

客服热线 400 - 820 - 9595

绵密网络 专业服务

中达电通已建立了 71 个分支机构及服务网点，并塑造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。

上海 电话 : (021)6301-2827	南昌 电话 : (0791)8625-5010	合肥 电话 : (0551)6281-6777	南京 电话 : (025)8334-6585	杭州 电话 : (0571)8882-0610
武汉 电话 : (027)8544-8475	长沙 电话 : (0731)8549-9156	南宁 电话 : (0771)5879-599	厦门 电话 : (0592)5313-601	广州 电话 : (020)3879-2175
济南 电话 : (0531)8690-7277	郑州 电话 : (0371)6384-2772	北京 电话 : (010)8225-3225	天津 电话 : (022)2301-5082	太原 电话 : (0351)4039-475
乌鲁木齐 电话 : (0991)4678-141	西安 电话 : (029)8836-0780	成都 电话 : (028)8434-2075	重庆 电话 : (023)8806-0306	哈尔滨 电话 : (0451)5366-0643
沈阳 电话 : (024)2334-16123	长春 电话 : (0431)8892-5060			



AX-3 系列快速入门手册

 **DELTA 台达**
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.deltagreentech.com.cn



扫一扫，关注官方微信

AX-3-0339310-04
2021/09/03

中达电通公司版权所有
如有改动,恕不另行通知

www.deltaww.com

 **DELTA**
Smarter. Greener. Together.

AX-3 系列快速入门手册

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	2020/08/21
第二版	1. 第 2 章：第 2.1.2 节更新 DIADesigner-AX 安装操作系统需求表 2. 第 3 章：第 3.2.2.1 节加批注说明、第 3.2.3.2 节更新软件图、第 3.3.1 节更新 ASDA-A2-M 讯号配置图、第 3.3.2.1 节增加批注说明及更新软件图、第 3.3.2.4 节更新项目建立流程说明及软件图、第 3.3.3 节更新软件图、第 3.5.2.1 节加批注说明、第 3.5.2.1 节更新软件图、第 3.5.3.2 节更新软件图	2021/03/29
第三版	1. 第 2 章：将范例機種改为 AX-324N 并更新相关软件画面	2021/09/3

AX-3 系列快速入门手册

目录

第 1 章 简介

1.1 手册内容简介	1-2
1.2 系统架构简介	1-3
1.3 主机运作介绍	1-4
1.3.1 工作 (Task)	1-4
1.3.2 工作 (Task) 类型	1-5
1.3.3 多工作 (Task) 执行	1-6
1.3.4 Watchdog 说明	1-7

第 2 章 建立一般标准 PLC 项目

2.1 基础工作准备	2-2
2.1.1 硬件	2-2
2.1.2 软件	2-3
2.1.3 工具与材料	2-3
2.2 安装	2-4
2.2.1 安装模块	2-4
2.2.2 安装脱落式端子	2-6
2.3 配线	2-8
2.3.1 PLC 电源模块配线	2-8
2.3.2 送电	2-9
2.4 软件基本操作说明	2-9
2.4.1 范例说明	2-9
2.4.2 建立项目	2-11
2.4.3 规划硬件架构	2-13
2.4.4 建立全局符号	2-17
2.4.5 建立功能块	2-20
2.4.6 建立主要程序	2-29

第 3 章 建立运动控制项目

3.1 基础工作准备	3-3
3.1.1 硬件	3-3
3.1.2 软件	3-4
3.1.3 工具与材料	3-4
3.2 EtherCAT 伺服轴项目建立	3-5
3.2.1 系统硬件配置	3-5
3.2.2 项目建立与程序编写	3-7
3.2.2.1 项目开启	3-7
3.2.2.2 轴参数设定	3-13
3.2.2.3 项目程序编写	3-20
3.2.3 程序监控	3-30
3.2.3.1 设备与计算机链接设定	3-30
3.2.3.2 程序监控流程	3-32
3.2.3.3 示波器监控	3-39
3.3 Pulse Out 伺服轴项目建立	3-47
3.3.1 系统硬件配置	3-47
3.3.2 项目建立与程序编写	3-49
3.3.2.1 项目开启	3-49
3.3.2.2 轴参数设定	3-53
3.3.2.3 设定伺服参数	3-56
3.3.2.4 项目程序编写	3-62
3.3.3 程序监控	3-75
3.3.3.1 设备与计算机链接设定	3-75
3.3.3.2 程序监控流程	3-77
3.3.3.3 示波器监控	3-81
3.4 虚拟轴作为同步主轴项目建立	3-89
3.4.1 系统硬件配置	3-89
3.4.2 项目建立与程序编写	3-90
3.4.2.1 项目开启	3-90
3.4.2.2 轴参数设定	3-97
3.4.2.3 项目程序编写	3-106
3.4.3 程序监控	3-117

3.4.3.1 设备与计算机链接设定	3-117
3.4.3.2 程序监控流程.....	3-119
3.4.3.3 示波器监控	3-126
3.5 内建编码器作为同步主轴项目建立	3-134
3.5.1 系统硬件配置	3-134
3.5.2 项目建立与程序编写	3-136
3.5.2.1 项目开启	3-136
3.5.2.2 轴参数设定	3-143
3.5.2.3 项目程序编写.....	3-154
3.5.3 程序监控	3-167
3.5.3.1 设备与计算机链接设定	3-167
3.5.3.2 程序监控流程.....	3-169
3.5.3.3 示波器监控	3-175

MEMO

第1章 简介

目录

1.1	手册内容简介	1-2
1.2	系统架构简介	1-3
1.3	主机运作介绍	1-4
1.3.1	工作 (Task)	1-4
1.3.2	工作 (Task) 类型	1-5
1.3.3	多工作 (Task) 执行	1-6
1.3.4	Watchdog 说明	1-7

1.1 手册内容简介

针对 AX 系列 PLC 的产品，台达依照不同的应用需求，分别为用户准备了不同的说明手册。

AX-3 系列操作手册：介绍硬件的配置、联机的设定、主机的运作及软件的设定操作等内容。

AX 系列运动控制指令手册：介绍 Motion FB 的内容。

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。

AS 系列模块手册：介绍各模块的规格、安装、设定及故障排除等内容。

DIADesigner-AX 使用手册：介绍编辑软件的详细使用，包含变量、联机、程序及功能块（FB）等内容。

AX-3 系列快速入门手册（本手册）：介绍如何让用户能先快速的建立并使用本系统。本手册除了介绍基本的系统架构外，也利用简单的实例来教导用户，如何一步一步的规划整个系统架构并编写程序，最后下载 PLC 程序并让主机运作，其中包含了变量、功能块（FB）等实际应用方式，让用户能快速的体验新功能所带来的便利性。若在系统运行时发生错误信息，请参考 AX-3 系列操作手册附录 A）。

手册中的图示说明

图示	说明
	单击鼠标左键
	单击鼠标右键
	双击鼠标左键
	按住鼠标左键后进行拖曳
	使用键盘输入
	操作顺序（当说明中需要特别表达操作顺序时会配合此标注，例如  、   ）
	配合图片说明的指示编号

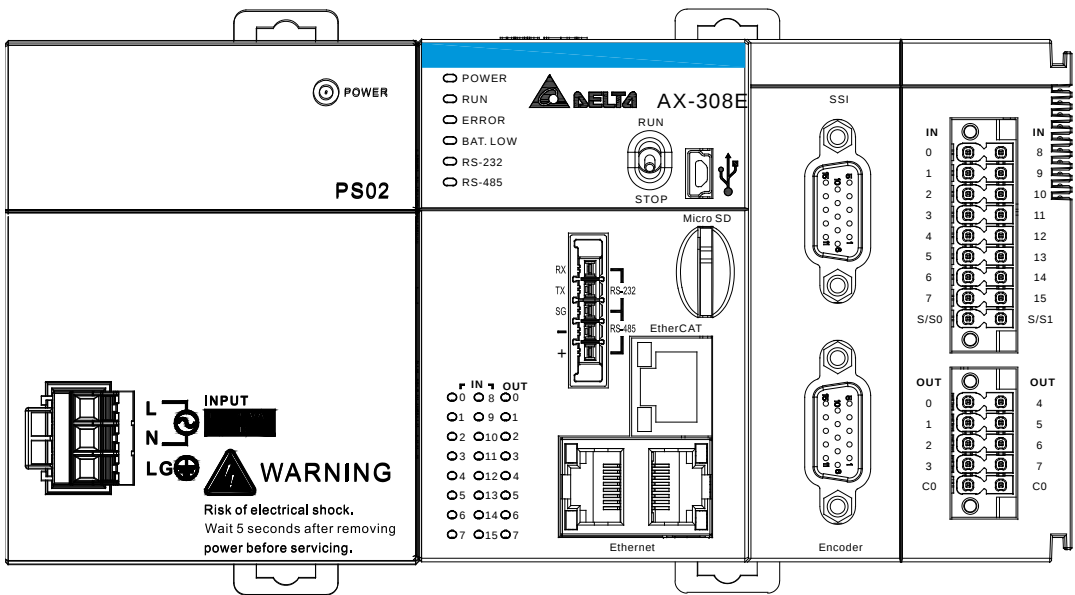
1.2 系统架构简介

AX-3 系列 PLC 为台达中小型的可程序化控制系统。除了执行速度与储存容量的提升之外，亦支持完整的功能块 (Function Block) 程序开发功能，同时为了符合用户更高端的应用需求，提供了更加灵活的系统扩展架构。在这样的一个系统架构下，用户不会因为系统点数过多或设备距离过远等问题，而必须将系统拆分成多个主机系统来控制，如此可以保留系统的完整性，也让用户在项目的开发过程中，能更加有效率。

AX-3 系列最小架构需求：

要架构一个 AX-3 系统，至少必须包含一台主机模块与一台电源模块，这样系统就可以进行规划与运作。

电源模块 + AX-3 主机 (AS-PS02 + AX308E-A)



AX-3 系列架构最大限制：

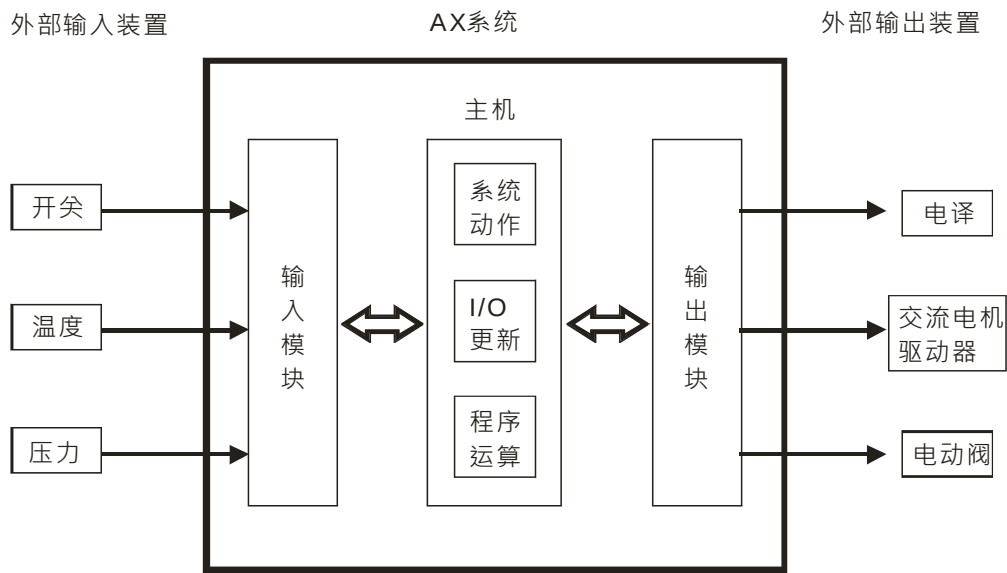
架构一个 AX-3 系统，其最大限制为下面条列三种限制状况，若超出任何一条限制条件，则主机将会发出警报信息。

限制一、扩展模块台数最多 32 台。(不含电源与主机)

限制二、IO 数据容量为 IN：8,192 bytes / OUT：8,192 bytes。

1.3 主机运作介绍

主机是整个 AX 系列的核心元件，其最主要的工作除了负责执行用户的逻辑程序之外，同时也负责所有 I/O 的数据收送与通讯数据的处理等工作。而主机与相关模块所建立起来的 AX 系统，与实际外部装置的关系可以简单表达如下：



以上为主机运作简单的表达方式，其中简化了初始化、诊断、通讯等系统面的流程，以及外部中断、时间中断等程序面的流程，若用户有兴趣想更深入的了解，可以参考完整的说明手册内容，下面对主机运作流程及工作相关进行说明。

1.3.1 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。

1.3.2 工作 (Task) 类型

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

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**

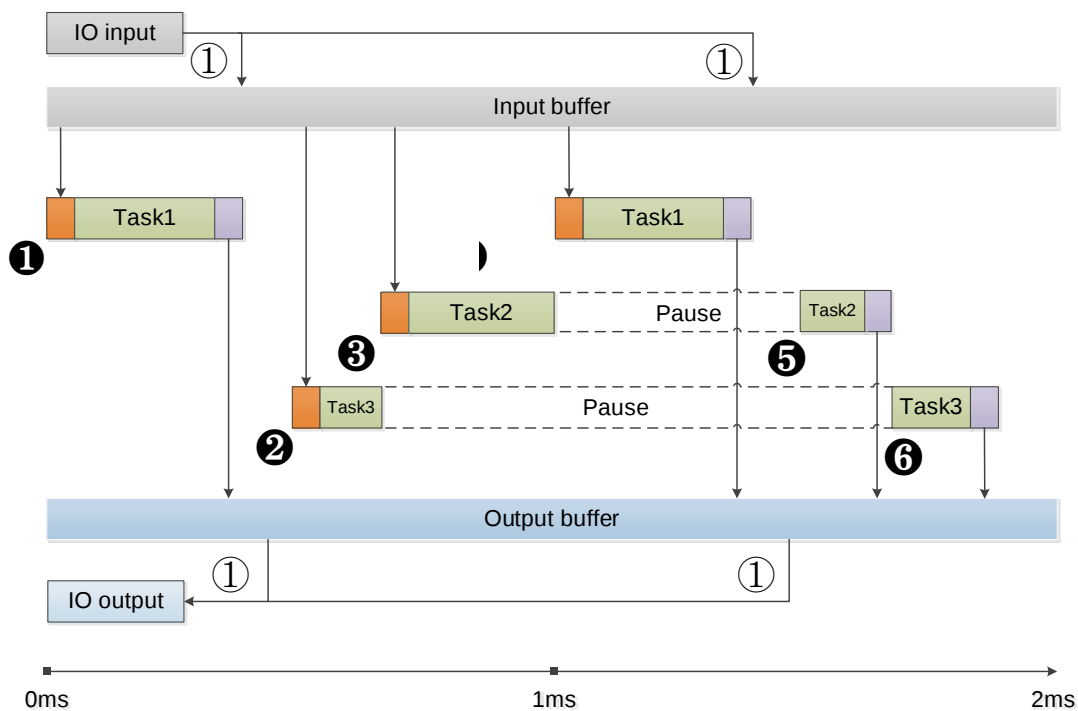
当 Status Task 内所定义的 Event 触发变量为「ON」时，则执行此 Status Task。

注 1：Event 触发变量必须为布尔变量。

1.3.3 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。

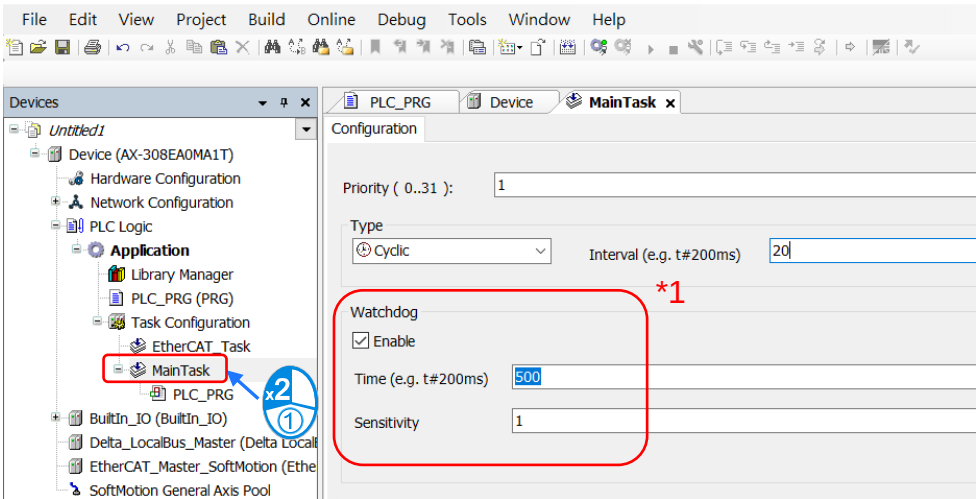
⑥：Task 2 执行完毕，Task 3 重新开始执行。

①：Bus cycle task。(注：在 Bus cycle 阶段时，会将 Input/output buffer 数据更新至实际的 IO 通道。)

1.3.4 Watchdog 说明

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。

MEMO

1

第2章 建立一般标准 PLC 项目

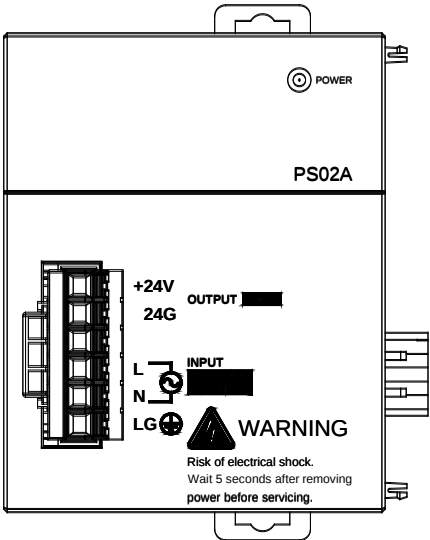
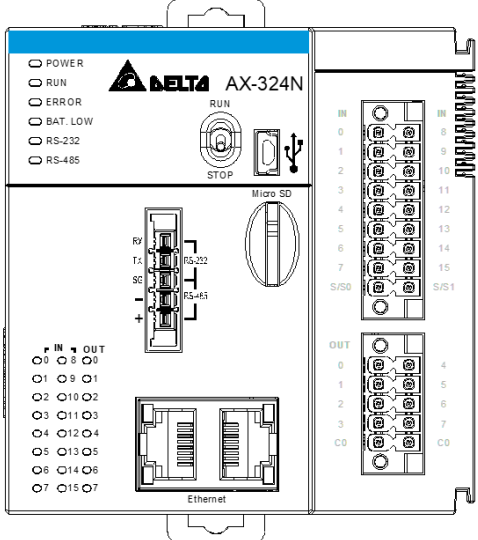
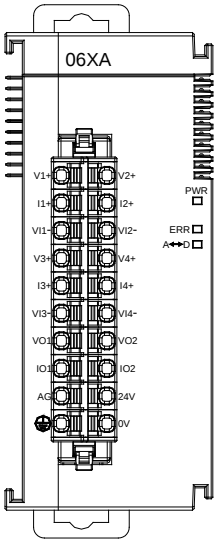
目录

2.1	基础工作准备	2-2
2.1.1	硬件	2-2
2.1.2	软件	2-3
2.1.3	工具与材料	2-3
2.2	安装	2-4
2.2.1	安装模块	2-4
2.2.2	安装脱落式端子	2-6
2.3	配线	2-8
2.3.1	PLC 电源模块配线	2-8
2.3.2	送电	2-9
2.4	软件基本操作说明	2-9
2.4.1	范例说明	2-9
2.4.2	建立项目	2-11
2.4.3	规划硬件架构	2-13
2.4.4	建立全局符号	2-17
2.4.5	建立功能块	2-20
2.4.6	建立主要程序	2-29

2.1 基础工作准备

2.1.1 硬件

范例所需要的硬件如下：

2 1. AS 系列电源模块 AS-PS02A x 1 	2. AX-3 系列主机（内建网络功能） AX-324N x 1 
3. AS 系列模拟输入输出混合模块（4AI+2AO 通道） AS06XA-A x 1 	

2.1.2 软件

范例所需要的软件如下：

DIADesigner-AX 安装操作系统需求表

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

2.1.3 工具与材料

范例所需要的工具与材料如下：

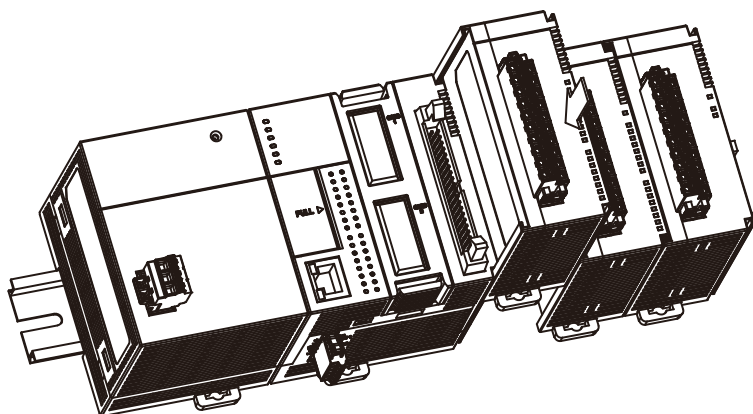
- 个人工作计算机（已安装上述需求之软件）x 1 台
- 100~240VAC 50/60Hz 的电源供应座 x 1 个
- 24VDC 电源供应器 x 1 个
- 电线 x 1 捆
- 针形端子压线工具 x 1 个
- 针形端子 x 20 个以上
- 一字螺丝起子 x 1 支
- USB 连接线或是网络线 x 1 条
- 若有需要可以准备开关与灯泡等配件（以便模拟外部设备动作）x 1 式

2.2 安装

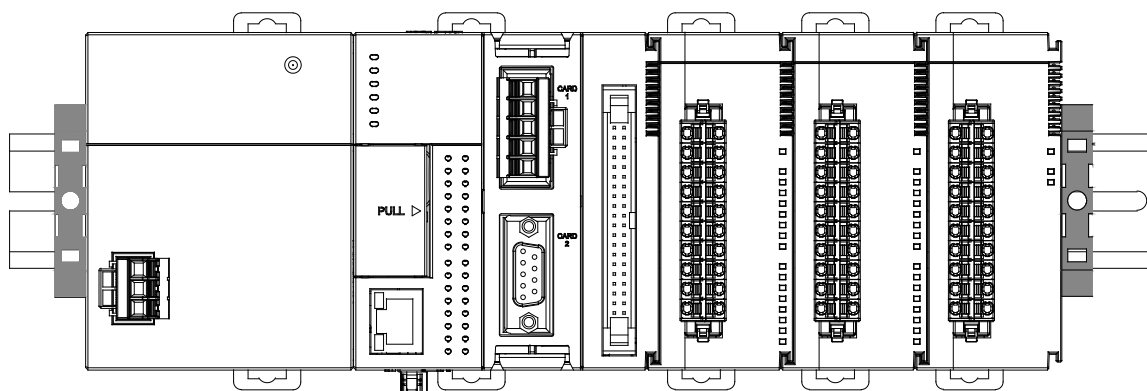
2.2.1 安装模块

模块安装的方式如下图标。

1. 将电源模块与主机结合之后，并将下方的卡勾先卡住，然后再卡上 DIN 轨 (DIN Rail) 上。
2. 对 IO 模块一台一台依序连接在主机右侧，并让下方卡勾先闭合，当模块推往 DIN 轨上卡住时，将会听到一声“喀”的音响，此时即表示模块已经卡上 DIN 轨，且与主机已经连接好了，如下图所示 (下方主机皆为参考 AS300 CPU 示意，并非 AX-324N)。

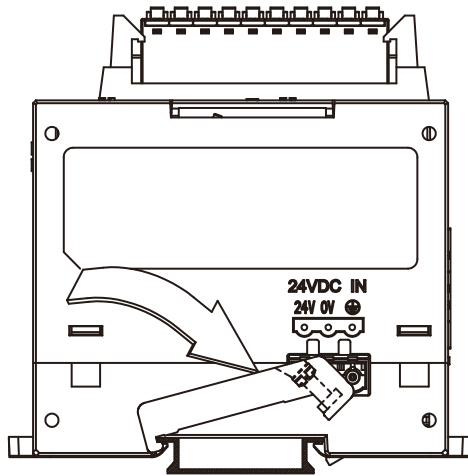


安装到位之后，若现场会有震动源，则建议在 AX-3 系列最左与最右两侧，补上挡板稳定所有模块。如下图所示最左与最右之灰色挡板。

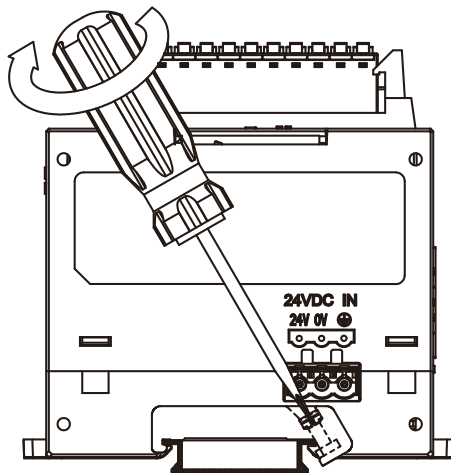


挡板安装如下：

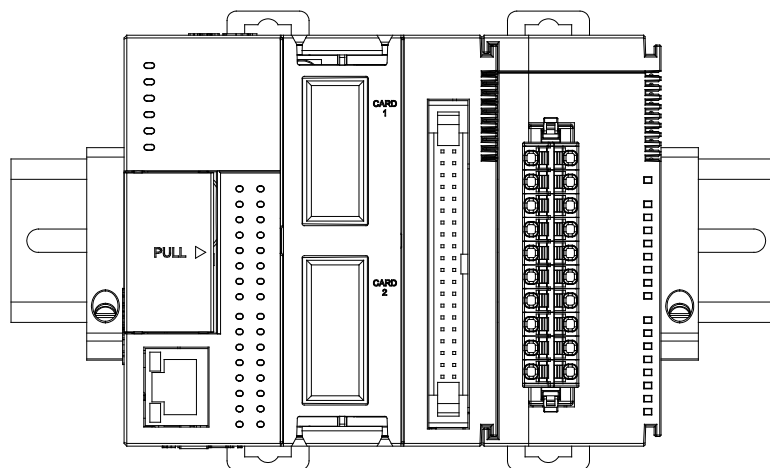
1. 将挡板扣住导轨（DIN Rail）往下压，如下图箭头所示。



2. 待挡板扣住导轨（DIN Rail）后，请使用螺丝起子将螺丝锁紧。



3. 安装完成后如下图所示。

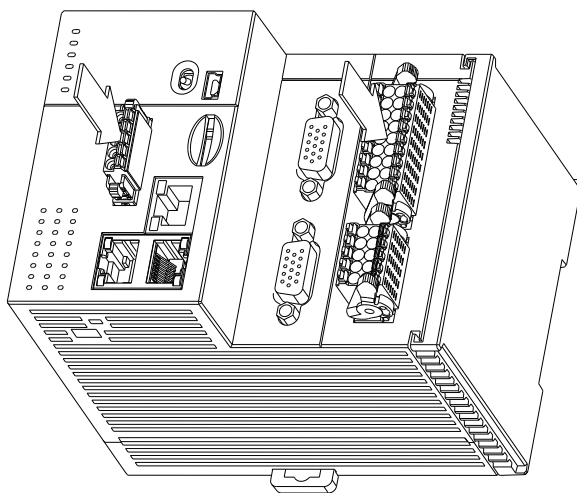


2.2.2 安装脱落式端子

将脱落式端子安装到主机上，安装方式说明如下。

● 脱落式端子安装方法

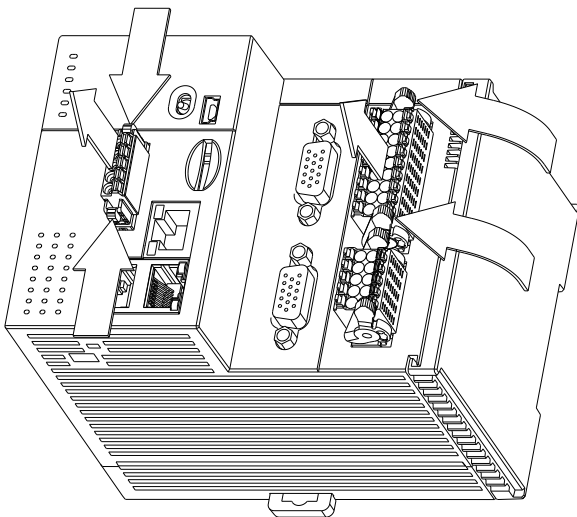
将端子下方卡槽对准端子座，直接往下压入，如下图所示。



● 脱落式端子取出方法

COM port 端子：两根手指往端子方向，按压端子的卡榫，并往上拉起，如下图左侧所示。

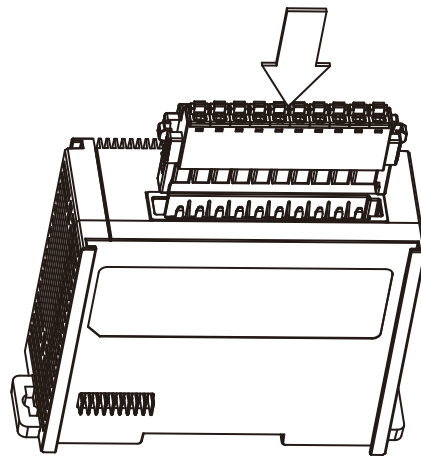
I/O 端子：两侧卡榫往原扣压的反方向拨动后往上拉起，如下图右侧所示。



将脱落式端子安装到模块上，安装方式说明如下。

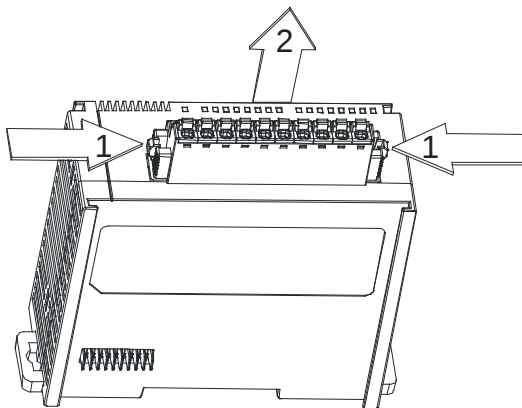
- 脱落式端子安装方法

将端子下方卡槽对准端子座，直接往下压入，如下图所示。



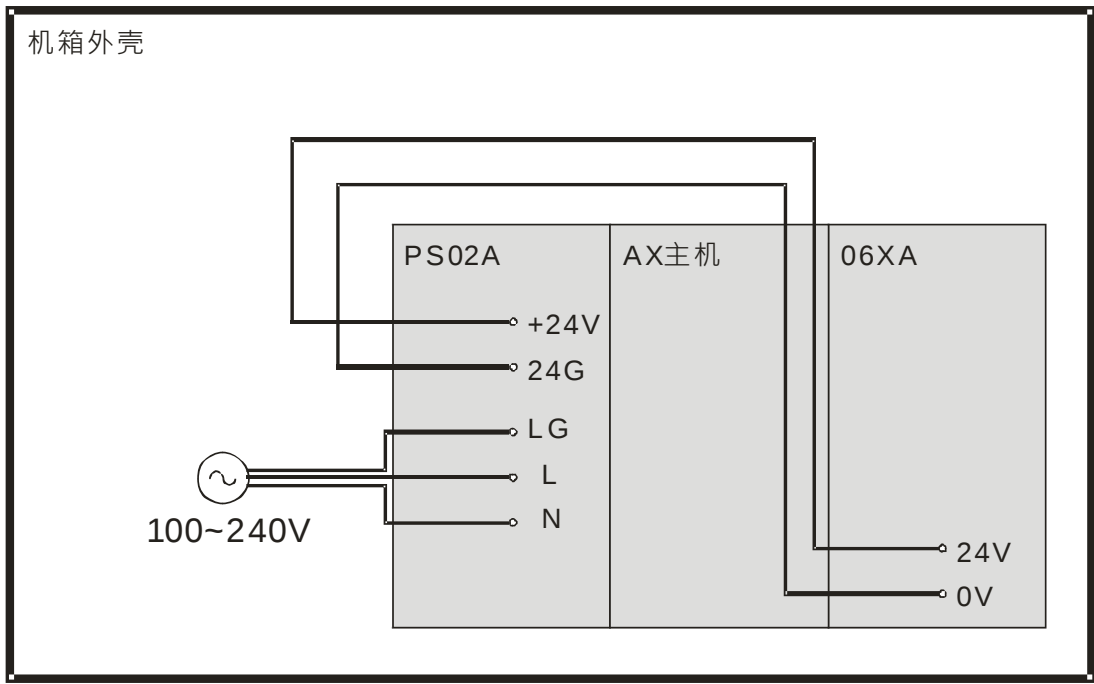
- 脱落式端子取出方法

两根手指往端子方向，按压端子的卡榫，并往上拉起，如下图所示。



2.3 配线

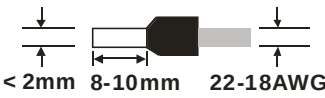
当安装好模块之后，接下来就可以进行模块的接线工作，为了后续范例的顺利进行，至少须将模块电源线的部份接上，也就是电源模块与模拟模块的电源部分，连接电源线的时候，请注意必须在断电的情况下操作以确保安全，而信号线的部份可依个人需求，看是否需要连接到开关与灯泡等元件，以增加操作仿真的便利性与真实感，大致的架构如下：



模块配线说明如下 (较完整的说明与注意事项请参阅 AS 系列模块手册)。

2.3.1 PLC 电源模块配线

- 交流电源输入电压，范围宽广 (100~240VAC)，电源请接在 L、N 两端，**如果将 AC110V 或 AC220V 接至 +24V 与 24G 输入端，将使 PLC 损坏，请用户特别注意。**
- 当停电时间低于 10ms 时，PLC 不受影响继续运转，当停电时间过长或电源电压下降将使 PLC 停止运转，输出全部 OFF，当电源恢复正常时，PLC 亦自动回复运转。(PLC 内部具有停电保持的变量设定功能，用户在作程序设计规划时应特别注意使用。)
- 电源输入端 (L、N) 及接地端 (LG) 请使用 22-18AWG 单蕊线或多蕊线，建议搭配孔径小于 2mm 的针形端子进行配线，规格如下图所示。只能使用 60/75°C 的铜导线。



2.3.2 送电

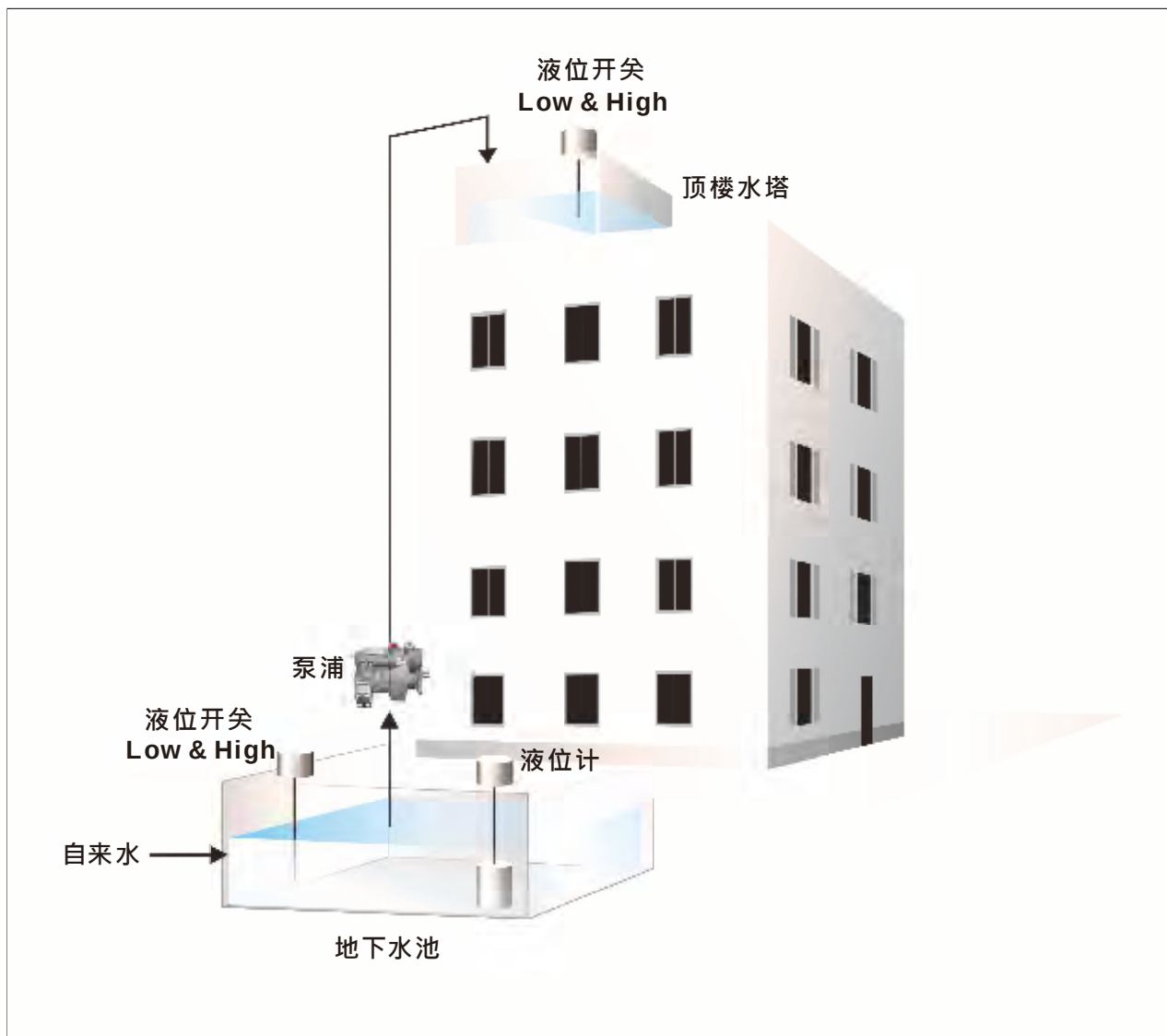
当所有配线确认无误完成之后，就可以准备送电，送电之前请先将主机的开关设置在 STOP 的位置，同时送电之后，主机会开始进行初始化的检查动作，当检查动作完成后，因为全新出厂的主机内部是没有任何程序与硬件配置，所以除了 POWER 灯号会亮灯以外不会有其它灯号亮灯，这是属于正常的现象。

2.4 软件基本操作说明

2.4.1 范例说明

当硬件安装好并完成配线与送电之后，接下来就可以开始准备进行程序编写的工作，在真正开始编写之前，为了让用户有一个明确的思考目标与方向，本手册准备了一个常用实例，并藉由一步一步的操作方式，将一个项目的完整流程，从建立新项目到下载至主机执行完整演练一遍，而以下是本范例的内容与架构说明。

● 系统架构示意



● 控制动作需求

本范例为一般大楼标准供水系统的基础设计，自来水会自动加入地下水池，而地下水池的水可经由泵浦输送至顶楼的水塔内，然后利用重力分送到各楼层使用，而泵浦的起停控制是依据地下水池与顶楼水塔的液位开关来动作的，同时为了监测自来水的供水情形，在地下水池部分另加装了压力式液位计，以便随时监视地下水池的蓄水容量。

2

PLC I/O 连接元件与装置：

1. 单点式液位开关 (A 接点) x 1

装置在地下水池，信号 (Low) 接点部份会连接至数字输入模块。

2. 两点式液位开关 (A 接点) x 1

装置在顶楼水塔，信号 (Low & High) 接点部份会连接至数字输入模块。

3. 泵浦 x 1

装置在地下水池附近，但实际 PLC 连接的装置并非为泵浦本体而是控制盘，而以一般的通用设计，通常为 3DI/1DO 的点数，远程控制 (Remote) x 1 (DI) & 运转 (Run) x 1 (DI) & 异常 (Trip) x 1 (DI) & 启动 (Start) x 1 (DO)，而这些信号会连接至数字输入与数字输出模块。

4. 压力式液位计 x 1

装置在地下水池，水位信号 (Level) 接点部份会连接至模拟输入模块 (0~10V 对应水位深度 0.0~10.0 公尺，0V 代表 0.0 公尺、10V 代表 10.0 公尺)。

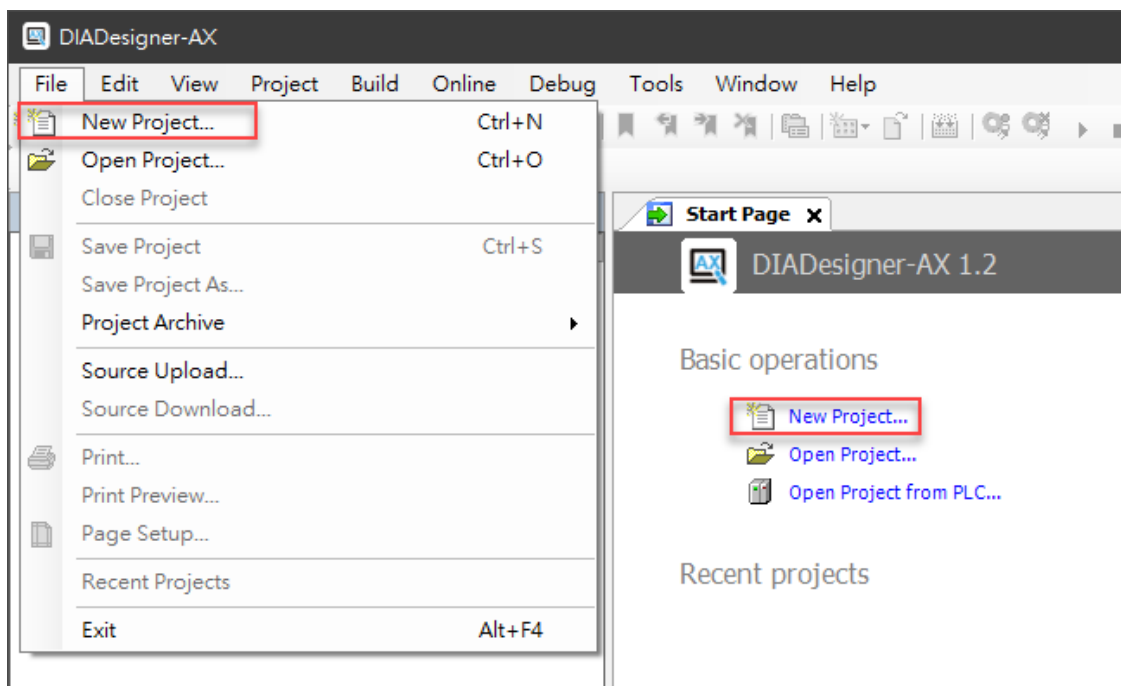
泵浦控制条件：

1. 启动：当顶楼水塔的液位 Low 触发时 (代表顶楼水不够)，且地下水池的液位 Low 并未触发时 (代表地下水池水足够，可以提供)，则启动泵浦进行补水。
2. 停止：当顶楼水塔的液位 High 触发时，或是地下水池的液位 Low 触发时，则停止泵浦运转。

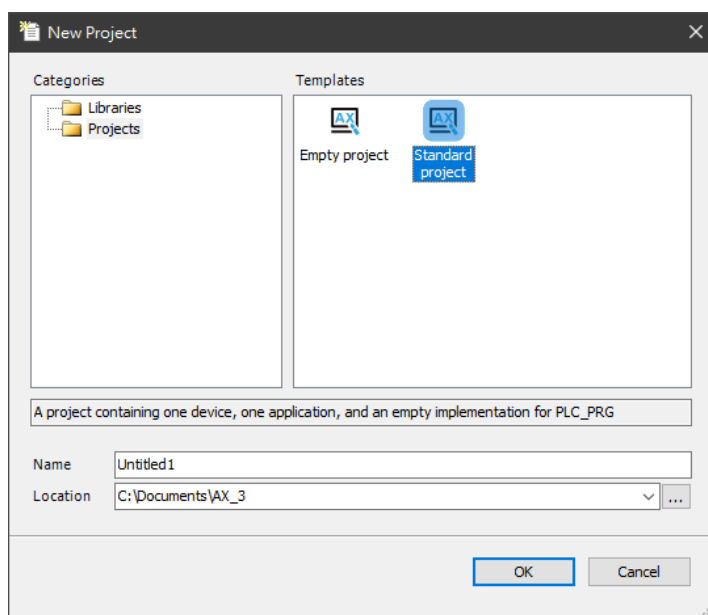
2.4.2 建立项目

当硬件安装完成并了解范例内的控制逻辑后，即可开始准备建立项目编写程序。

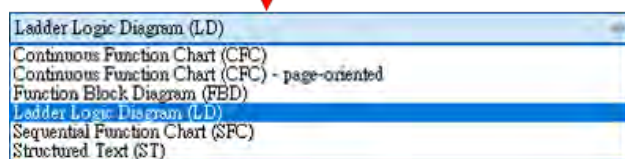
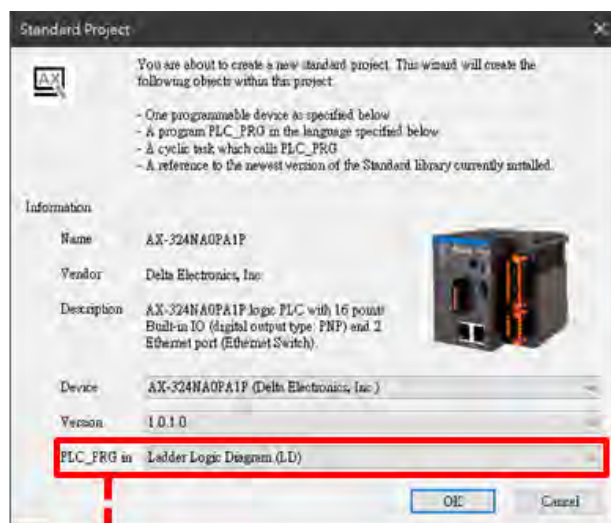
双击桌面编程软件图标  启动 DIA Designer-AX 后，在工具栏中单击「File」>「New Project (Ctrl+N)」或直接单击  图示；或在「Start Page」分页内单击  图示旁「New Project...」。



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」项目名称及「Location」文件路径，完成后按下「OK」确定。

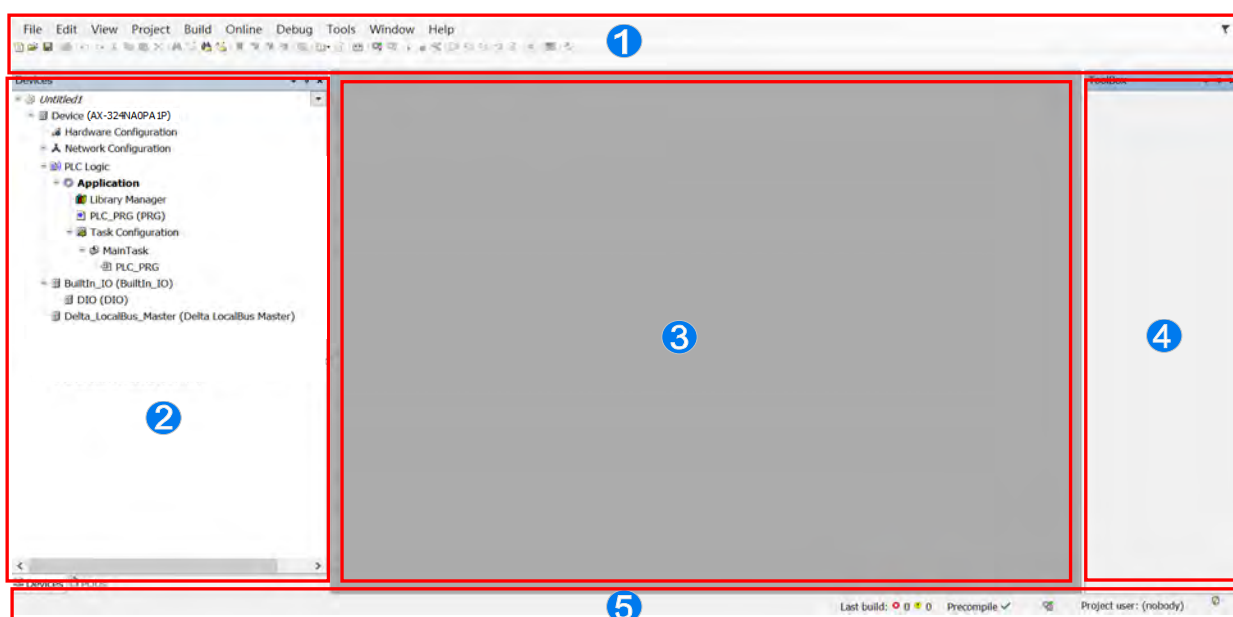


新建标准项目最后根据对话框提示，选择「Device」目标设备及「PLC_PRG in」主程序的编程语言，完成后按下「OK」确定，系统会自动生成一个默认为周期循环型的任务，任务中包含一个默认的程序 PLC_PRG。



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。



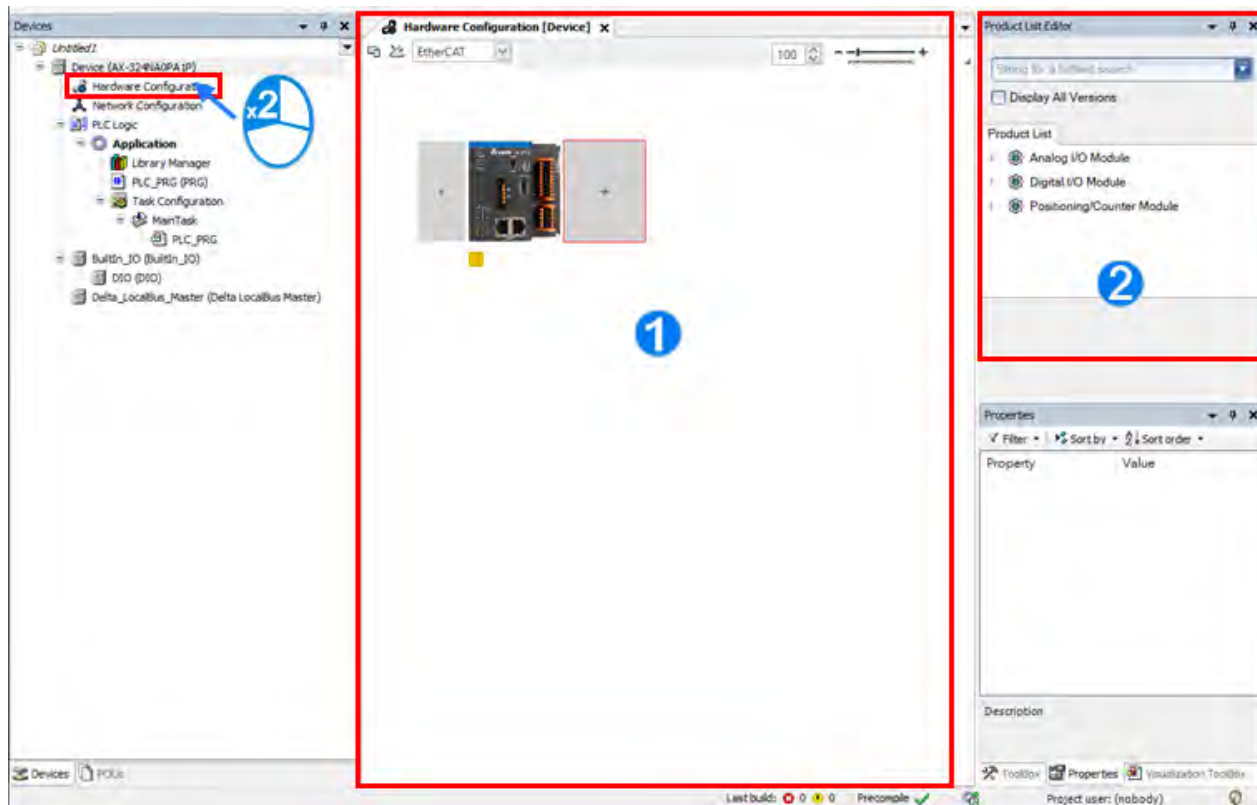
- ❶ 功能区：包含了整个编辑软件环境的主要功能，很多常用的功能都会放置在此区的快捷功能列上，而比较细项的功能则会整理在功能列里面。
- ❷ 项目管理区：这里展示了整个项目的主要对象架构，由这样的树状架构可以很容易的理解项目对象的关系，也加强了项目功能管理的效率。
- ❸ 工作区：大部分用户的编辑工作都会在此区。
- ❹ 工具区：在程序区编辑时，可在此处选择有显示可支持编写程序应用的相关对象。
- ❺ 状态区：显示目前项目的一些信息与状态。

2.4.3 规划硬件架构

当项目建立完成后，就可以进行硬件的规划工作，针对这个范例我们先假设以下的规划。

- 主机数字输入的部份 (16 点) 默认地址为%IX0.0 ~ %IX0.7 & %IX1.0 ~ %IX1.7
- 主机数字输出的部份 (8 点) 默认地址为%QX0.0 ~ %QX0.7
- 06XA 模拟输入的部份 (4 通道) 默认地址为%ID2 ~ %ID5
- 06XA 模拟输出的部份 (2 通道) 默认地址为%QD1 ~ %QD2
- 地下水池的 Low 点 → %IX0.0
- 顶楼水塔的 Low 点 → %IX0.1
- 顶楼水塔的 High 点 → %IX0.2
- 泵浦的远程控制 (Remote) 点 → %IX0.3
- 泵浦的运转 (Run) 点 → %IX0.4
- 泵浦的异常 (Trip) 点 → %IX0.5
- 泵浦的启动 (Start) 点 → %QX0.0
- 地下水池的 Level 值 → %ID2

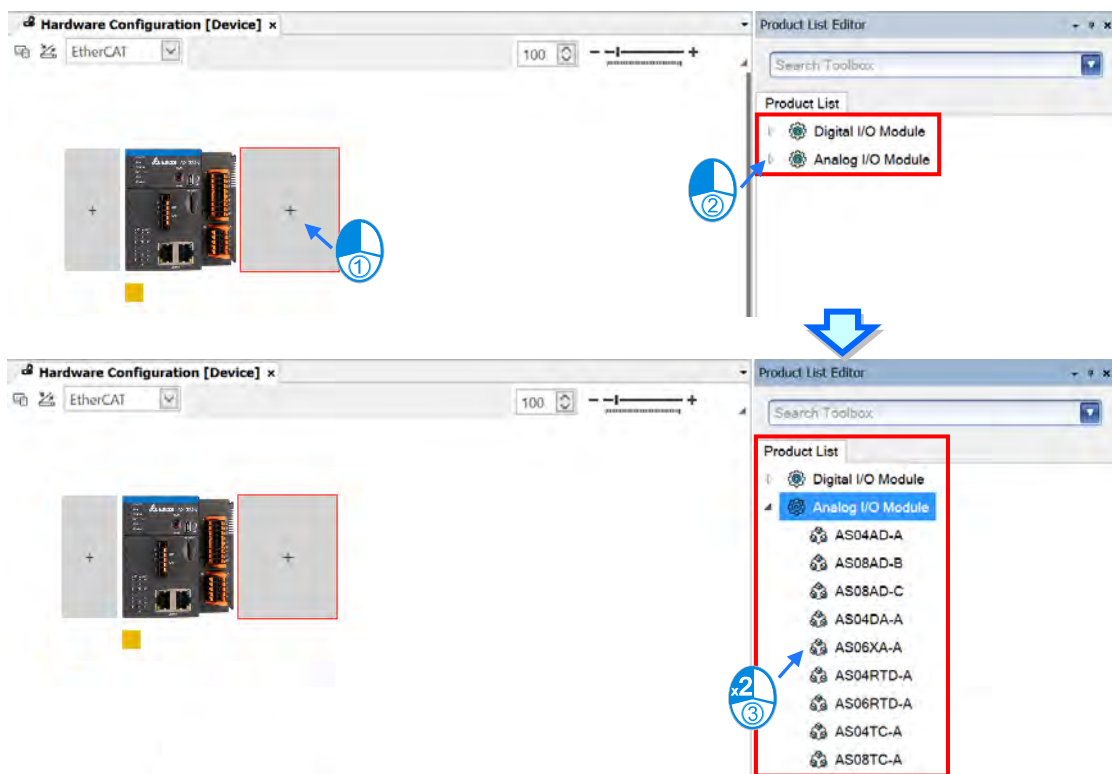
有了以上的信息后就可以实际进行规划了，在左侧「Devices」窗口中，使用鼠标左键双击 HWCONFIG 的图标与文字  Hardware Configuration，Hardware Configuration [Device]选项分页窗口会在设备列右侧打开，如下图所示。



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。

现在就可以依照之前所规划的范例信息开始进行规划，接下来可以进行模块的配置，先单击主机右侧“+”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。



若再在该模块名称上鼠标左键双击单击，就会直接将模块 AS06XA 置入原本“+”符号的位置，如下图所示。



在模拟模块的部份，必须还要确定设定信号类型与工程值转换关系，这样才算是完成基本设定，如果要进行这样的设定，可以直接连点该模块图标两下，系统会自动开启对应的参数设定窗口。

AS06XA_A x

AS06XA-A Parameters

AS06XA-A I/O Mapping

Status

Information

Parameter	Type	Value	Default Value	Unit	Description
Format	Enumeration of UINT	Integer	Integer		
Input CH1~CH4 Mode Setting					
Input CH1 Mode Setting	Enumeration of WORD	-10V~+10V	-10V~+10V		
Input CH2 Mode Setting	Enumeration of WORD	-10V~+10V	-10V~+10V		
Input CH3 Mode Setting	Enumeration of WORD	-10V~+10V	-10V~+10V		
Input CH4 Mode Setting	Enumeration of WORD	-10V~+10V	-10V~+10V		
Input CH1~CH4 Calibration					
Input CH1 Cal. Offset	INT(-32768..32767)	0	0		
Input CH2 Cal. Offset	INT(-32768..32767)	0	0		
Input CH3 Cal. Offset	INT(-32768..32767)	0	0		
Input CH4 Cal. Offset	INT(-32768..32767)	0	0		
Input CH1 Cal. Gain	INT(-32768..32767)	1000	1000		
Input CH2 Cal. Gain	INT(-32768..32767)	1000	1000		
Input CH3 Cal. Gain	INT(-32768..32767)	1000	1000		
Input CH4 Cal. Gain	INT(-32768..32767)	1000	1000		
Input Average Filter					
Input CH1 Average Times	WORD(1..100)	10	10		
Input CH2 Average Times	WORD(1..100)	10	10		
Input CH3 Average Times	WORD(1..100)	10	10		
Input CH4 Average Times	WORD(1..100)	10	10		
Input CH1 Filter Proportion	Enumeration of WORD	10%	10%		
Input CH2 Filter Proportion	Enumeration of WORD	10%	10%		
Input CH3 Filter Proportion	Enumeration of WORD	10%	10%		
Input CH4 Filter Proportion	Enumeration of WORD	10%	10%		
Input Sampling Time					
Input sampling time	Enumeration of WORD	2ms	2ms		
Input Channel Detect and Alarm Settings	WORD	0			

依照前述范例需求，调整信号为 0~10V 电压工作模式。

AS06XA_A x

AS06XA-A Parameters

AS06XA-A I/O Mapping

Status

Information

Parameter	Type	Value	Default Value	Unit	Description
Format	Enumeration of UINT	Integer	Integer		
Input CH1~CH4 Mode Setting					
Input CH1 Mode Setting	Enumeration of WORD	0V~+10V	-10V~+10V		
Input CH2 Mode Setting	Enumeration of WORD	-10V~+10V	-10V~+10V		
Input CH3 Mode Setting	Enumeration of WORD	-10V~+10V	-10V~+10V		
Input CH4 Mode Setting	Enumeration of WORD	-10V~+10V	-10V~+10V		

再单击分页「AS06XA-A I/O Mapping」，即可以看到各输入与输出通道自动配置的地址。

AS06XA_A x

AS06XA-A Parameters

AS06XA-A I/O Mapping

Status

Information

Find

Filter

Show all

Ad

Variable	Mapping	Channel	Address	Type	Unit	Description
		AS06XAIN	%ID1			
		Error code	%ID1	DWORD		
		CH1 Input	%ID2	DINT		
		CH2 Input	%ID3	DINT		
		CH3 Input	%ID4	DINT		
		CH4 Input	%ID5	DINT		
		AS06XAOUT	%QD1			
		CH1 Output	%QD1	DINT		
		CH2 Output	%QD2	DINT		

这样硬件架构部份规划完成，准备进行后续步骤。

2.4.4 建立全局符号

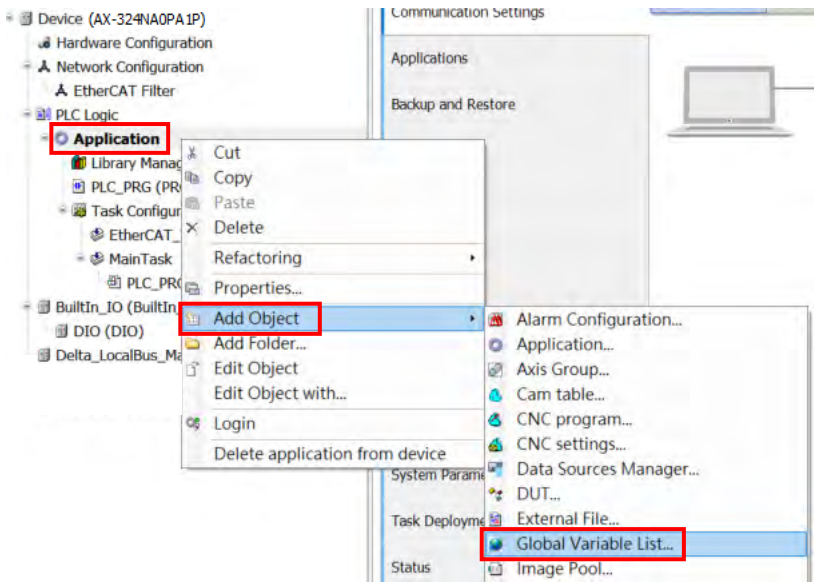
为了程序的可读性，一般会将实体 I/O 的地址部份再配置一个全局符号，这样在编写程序时，就可以使用容易理解的符号方式来编辑。

在这个例子我们会建立以下的全局符号：

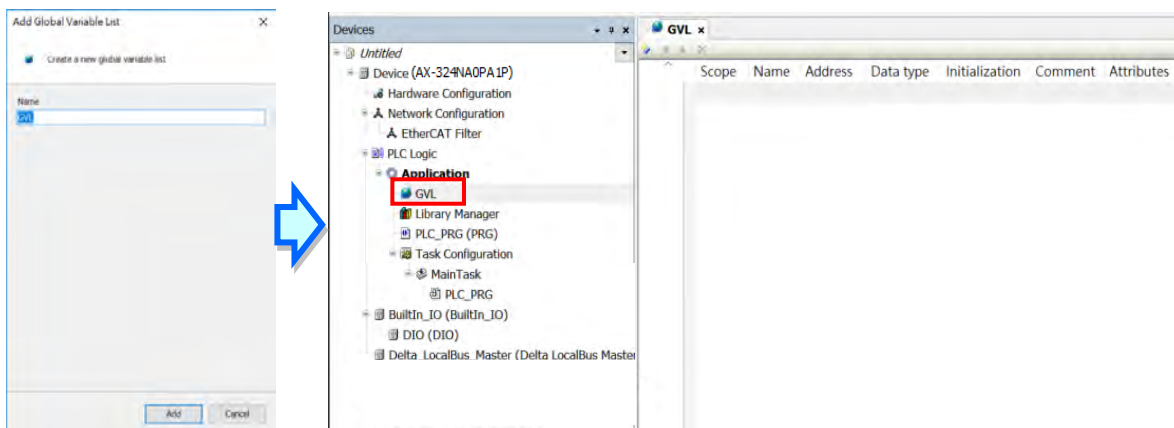
全局符号总表		
硬件点		
符号名称	地址	数据类型
Tank_B1F_LSW	%IX0.0	BOOL
Tank_RF_LSW	%IX0.1	BOOL
Tank_RF_HSW	%IX0.2	BOOL
SPP01_Remote	%IX0.3	BOOL
SPP01_Run	%IX0.4	BOOL
SPP01_Trip	%IX0.5	BOOL
SPP01_Start	%QX0.0	BOOL
Tank_B1F_LT	%ID2	DINT
软件点		
SPP01_Auto		BOOL
SPP01_Man_SW		BOOL

有了以上的信息后，我们可以准备开始建立全局符号，宣告要在整个应用程序中使用的变量，请先在「Application」下面建立一个全局变量列表（GVL）。

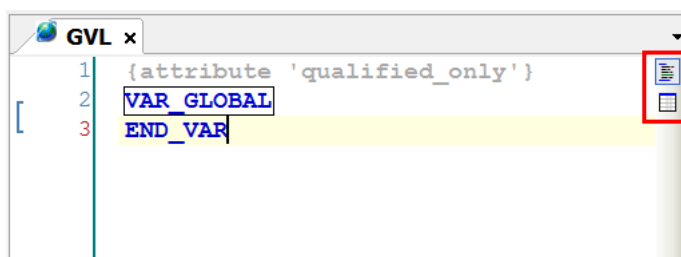
- 1. 鼠标右击「Application」，然后选择「Add Object」后再单击「Global Variable List」增加对象。



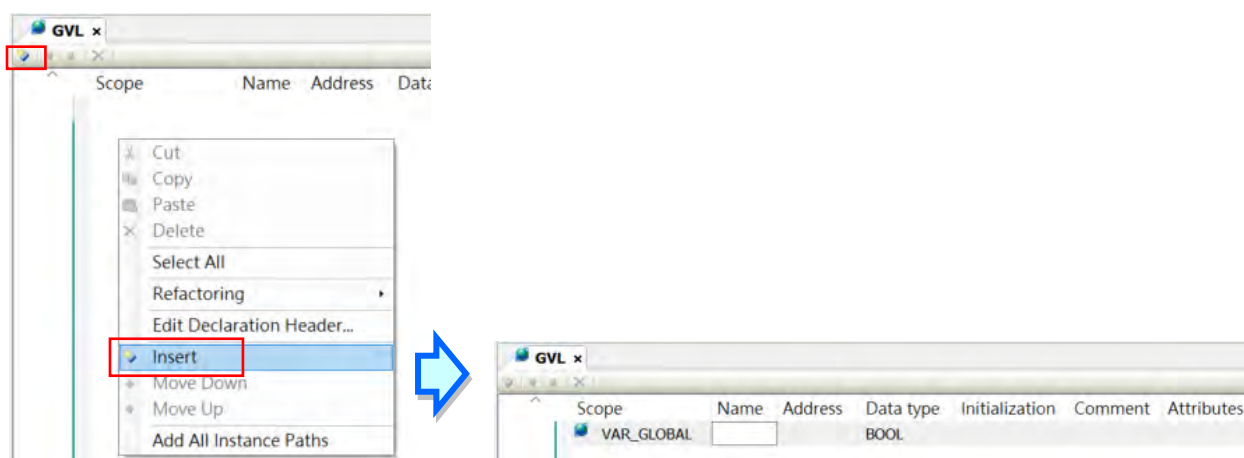
2. 此时跳出窗口「Add Global Variable List」会自动输入名称 GVL，然后单击「Add」确认新增，「GVL」对象会出现在「Application」下方，「GVL」编辑器将会在设备列的右侧打开。



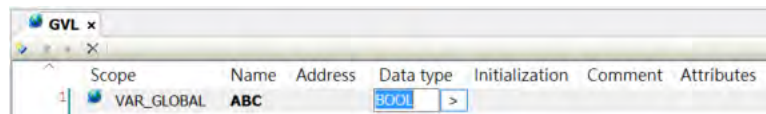
3. 当以表格方式出现时，将有一个空行且光标会在「Name」中。如果要切换成文本方式，可通过单击编辑器右侧字段中的按钮  来切换，当文本方式出现时，它已经包含关键词 VAR_GLOBAL 和 END_VAR。如要切换回表格方式可再单击右侧字段中的按钮  即可。



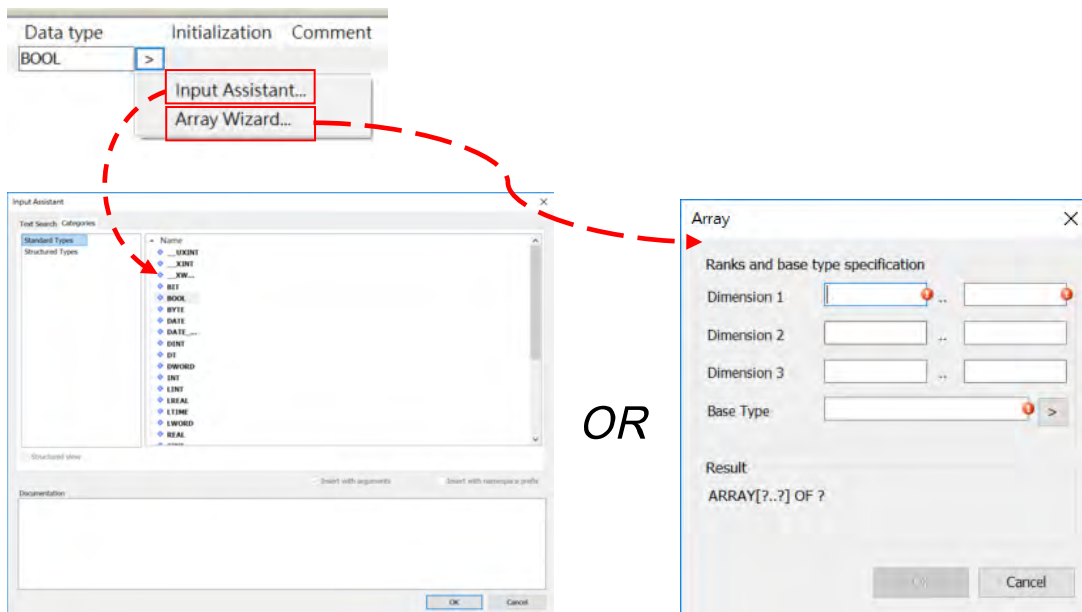
4. 在「GVL」窗口中空白处使用鼠标右击后，在选单中选择「Insert」命令或直接单击左上方 Insert 按钮 ，将会出现新的一行，且同时「Scope」为 VAR_GLOBAL 和「Data type」为 BOOL 的设定参数会自动输入到行中，只要再输入变量名称于 Name 即可。



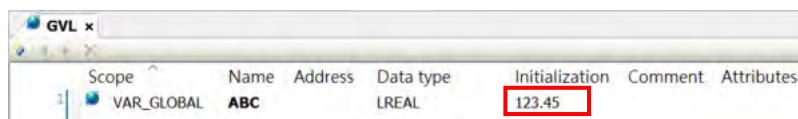
5. 鼠标左键双击「Data Type」列中的字段，就可以编辑该字段，并且在字段右侧会出现按钮 。



6. 此时鼠标左键单击按钮  可以获取更多设置，然后单击「Input Assistant」，再通过打开的窗口来设定变量类型；或是以单击「Array Wizard」之后出现的窗口来设定矩阵数据类型。 2



7. 变更过变量欲设定的数据类型后，可在「Initialization」列中输入一个数值来设定初始值。



8. 之后即可以相同的方式，将一开始叙述表格内所有的数据建立在全局符号表，并加入对应的 Address 实体 IO 位置。完成后应如下图所示。

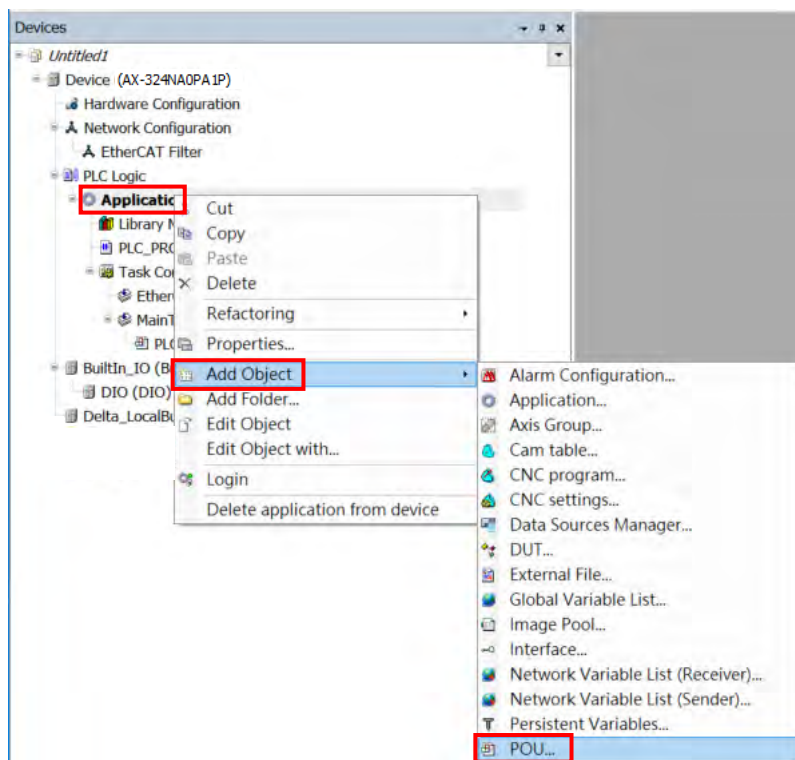
	Scope	Name	Address	Data type	Initialization	Comment
1	VAR_GLOBAL	Tank_B1F_LSW	%IX0.0	BOOL		Tank_B1F Level Switch-Low
2	VAR_GLOBAL	Tank_RF_LSW	%IX0.1	BOOL		Tank_RF Level Switch-Low
3	VAR_GLOBAL	Tank_RF_HSW	%IX0.2	BOOL		Tank_RF Level Switch-High
4	VAR_GLOBAL	SPP01_Remote	%IX0.3	BOOL		Supply Water Pump-SPP01 Remote
5	VAR_GLOBAL	SPP01_Run	%IX0.4	BOOL		Supply Water Pump-SPP01 Run
6	VAR_GLOBAL	SPP01_Trip	%IX0.5	BOOL		Supply Water Pump-SPP01 Trip
7	VAR_GLOBAL	SPP01_Start	%QX0.0	BOOL		Supply Water Pump-SPP01 Start
8	VAR_GLOBAL	Tank_B1F_LT	%ID2	DINT		Tank B1F Level Meter
9	VAR_GLOBAL	SPP01_Auto		BOOL		Supply Water Pump-SPP01 Auto Mode
10	VAR_GLOBAL	SPP01_Man_SW		BOOL		Supply Water Pump-SPP01 Manual Command

2.4.5 建立功能块

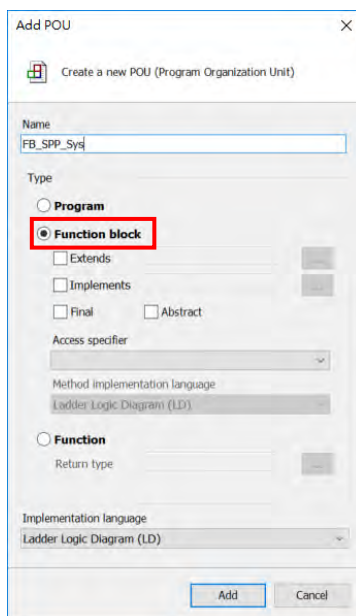
本范例的流程是先建立功能块的程序部份，当然也有人喜好先建立主流程的程序部份，这并没有绝对的关系，而使用的经验上，其实两者常常交互使用开发并无绝对的顺序，不过建议应当先开发功能块的部份会比较便利，尤其是常常重复使用的功能，就建议使用功能块节省程序开发时间。

以本范例来看，就会考虑把水池、水塔、与泵浦的控制关系写成功能块 (FB)，因为实际的应用上，一个大楼并不会只有一套这样的供水系统，通常会有两套以上的设计，如果使用功能块的方式，则在程序上只需要更换功能块 Input 与 Output 引脚的变量，就等于完成第二套系统的开发。

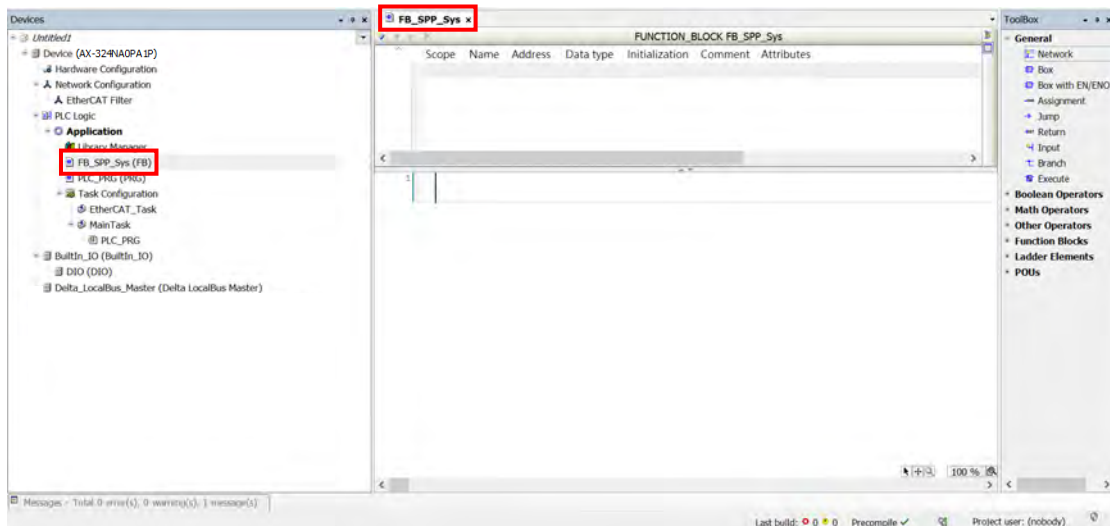
现在我们就开始建立功能块，目的就是把水池、水塔、与泵浦三者的控制关系包在一个功能块里面，首先在主画面左侧的窗口「Devices」内找到「Application」的图标与文字 **Application**，利用鼠标右键的方式选择「Add Object」，再使用左键选择「POU」新增。



可在「Name」下方框内填入新名称，并选择「Type」为 Function block，内容请参考下图，其余可保持默认值并按下「Add」。



这样在左侧设备列就会看到新增的 FB 名称，且同时主画面也会显示功能块的程序编辑窗口。

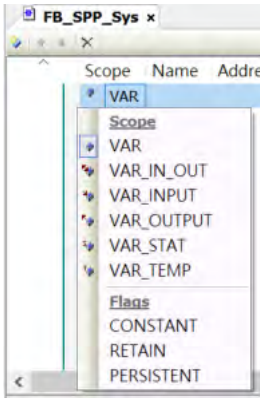


首先，先建立功能块的「区域符号」，这个部份就等于将来这个功能块的参数引入与引出（引脚）的部份，建立方式与之前提到的全局符号方式相同，当然在程序内部仍然可以使用全局符号来编辑程序，但是请注意这样使用时会降低这个功能块的可移植性与便利性。（区域符号与其它区域符号或是全局符号之间，其符号名称是可以重复的，当重复时系统会以程序或功能块本身的区域符号为识别优先）

以下为本范例需要建立的区域符号：

类别	符号	数据类型
VAR_INPUT	Tank_B_LSW	BOOL
VAR_INPUT	Tank_R_LSW	BOOL
VAR_INPUT	Tank_R_HSW	BOOL
VAR_INPUT	Pump_Remote	BOOL
VAR_INPUT	Pump_Run	BOOL
VAR_INPUT	Pump_Trip	BOOL
VAR_IN_OUT	Pump_Auto	BOOL
VAR_IN_OUT	Pump_Man_SW	BOOL
VAR_OUTPUT	Pump_Start	BOOL
VAR	Pump_Out	BOOL

在变量列表字段「Scope」下方新增字段 VAR 处使用鼠标左键双击，会显示菜单选项来变更变量引脚类型。



VAR_INPUT

程序执行时会先将外部变量的数值引入到内部变量里，当对应的内部变量数值有变更时，并不会再传送给外部变量，通常会使用在外部变量不应该被修改的情形，本范例中大多是由数字输入（DI）而来的状态，像这一类外部变量的状态是不应该被修改的，所以宣告成 VAR_INPUT 可避免程序误改到这类变量的数值，而影响后续其它程序或功能块的使用。

VAR_IN_OUT

程序执行时会先将外部变量的数值引入到内部变量里，当程序结束时会再传送给外部变量，通常会使用在变量需要被修改的情形，像本范例中 Pump_Auto 与 Pump_Man_SW 的两个符号变量，一般来说是给 SCADA 系统的用户操作使用，以决定泵浦的控制模式是要自动模式（依液位信息控制）或是手动模式（依照 Pump_Man_SW 的开关状态控制），看起来好像只要 VAR_INPUT 类别就可以，但是考虑到当泵浦发生故障（TRIP）时，为了保护与提醒作用，需要将泵浦的模式自动切换到手动且停止命令输出的状态，这样就必须要变更 Pump_Auto 与 Pump_Man_SW 这两个符号变量，所以需要将这两个变量宣告成 VAR_IN_OUT 的类别。

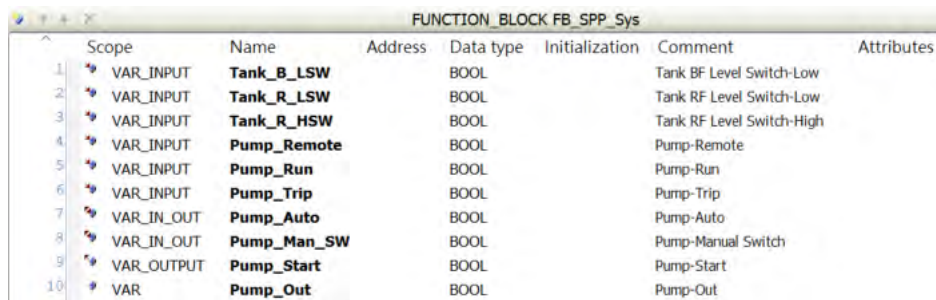
VAR_OUTPUT

程序执行时并不会先将外部变量的数值引入到内部变量里，而是直接使用之前内部储存的数值，而在程序结束时会将内容再传送给外部变量，一般来说此类变量在程序内使用时，都是落在指令的 OUT 位置，像本范例中的 Pump_Start 就属于这类的使用。

VAR

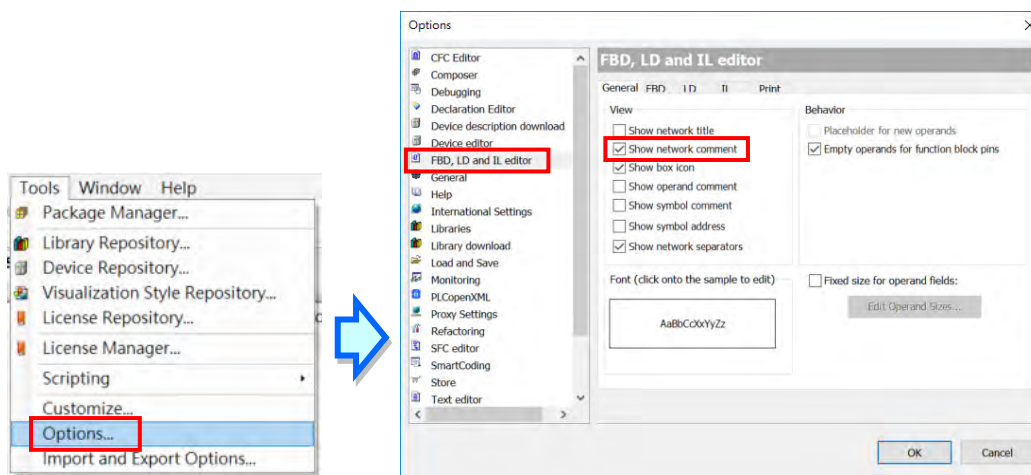
程序执行时会视为内部变量，与 VAR_OUTPUT 一样会使用之前内部储存的数值，没有引入与传送外部变量的功能，一般来说此类变量在程序内使用时，都是当作程序运算暂存使用，像本范例中的 Pump_Out 就属于这类的使用。

当建立完成后应如下图所示。



	Scope	Name	Address	Data type	Initialization	Comment	Attributes
1	VAR_INPUT	Tank_B_LSW		BOOL		Tank BF Level Switch-Low	
2	VAR_INPUT	Tank_R_LSW		BOOL		Tank RF Level Switch-Low	
3	VAR_INPUT	Tank_R_HSW		BOOL		Tank RF Level Switch-High	
4	VAR_INPUT	Pump_Remote		BOOL		Pump-Remote	
5	VAR_INPUT	Pump_Run		BOOL		Pump-Run	
6	VAR_INPUT	Pump_Trip		BOOL		Pump-Trip	
7	VAR_IN_OUT	Pump_Auto		BOOL		Pump-Auto	
8	VAR_IN_OUT	Pump_Man_SW		BOOL		Pump-Manual Switch	
9	VAR_OUTPUT	Pump_Start		BOOL		Pump-Start	
10	VAR	Pump_Out		BOOL		Pump-Out	

接着可以准备编写功能块的程序内容了，为了程序的浏览方便建议培养备注的编写习惯，若要编写程序区段备注请先开启区段备注的功能，上方工具栏「Tools」>「Options」>「FBD, LD and IL editor」>「General」内选项「Show network comment」打勾后按下「OK」即可。



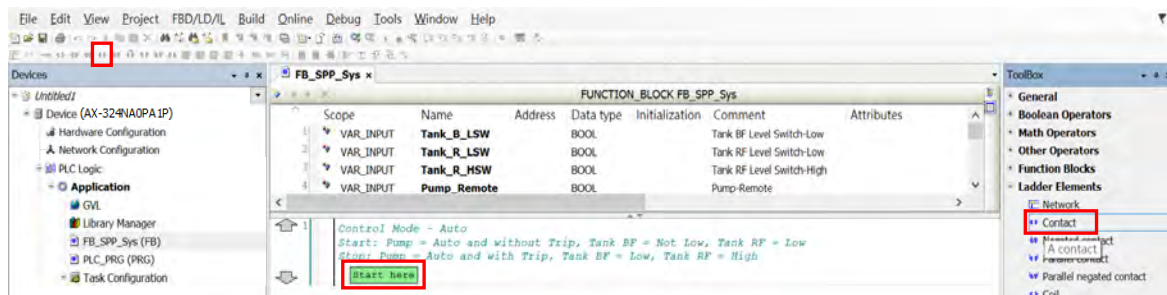
先在区段 1 的备注处填写说明。输入时，按下键盘的【Enter】即可换行。

```

1 | Control Mode - Auto
   | Start: Pump = Auto and without Trip, Tank BF = Not Low, Tank RF = Low
   | Stop: Pump = Auto and with Trip, Tank BF = Low, Tank RF = High

```

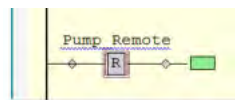
接着编辑程序代码，此处使用的语言为 LD，请按下上方的  图示或持续按压主画面右方「ToolBox」>「Ladder Elements」下的  **Contact**，将鼠标拖移到绿框位置显示「Start here」处，使光标出现+符号后，再放开鼠标左键放置。



接着用鼠标单击接点的???处，并填入「Pump_Remote」的名称，输入完成后按下输入键。



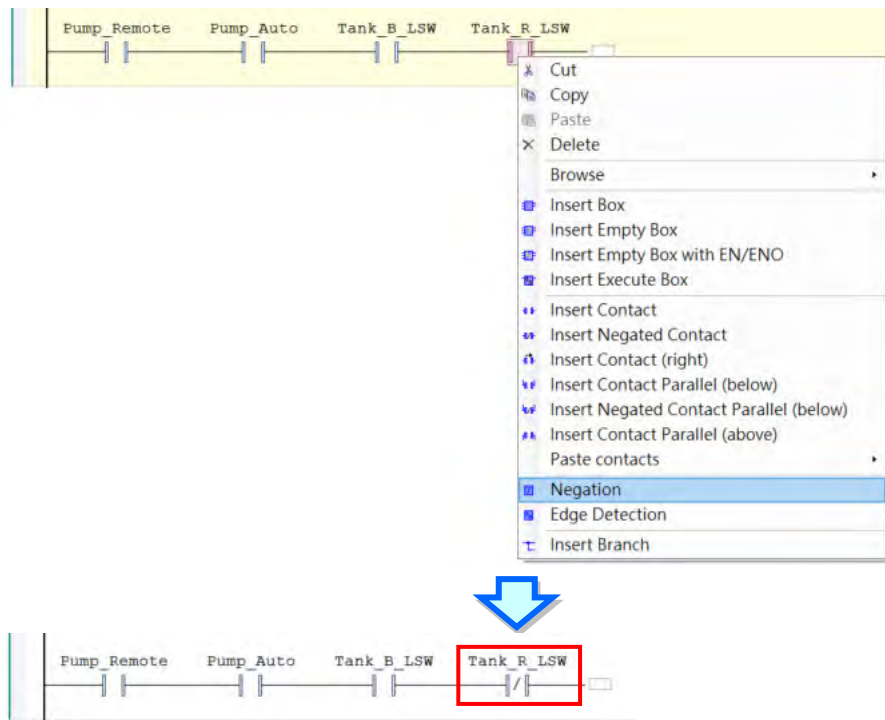
之后须要串接的对象  可拉至绿框位置处使光标出现+符号后放置或在鼠标左键单击区段 1 后按下快捷键 **Ctrl+K** 即可连续插入对象。



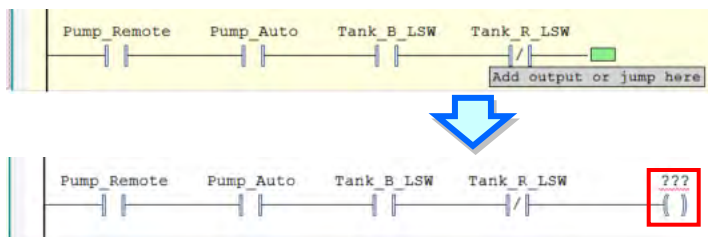
利用相同操作完成以下程序内容



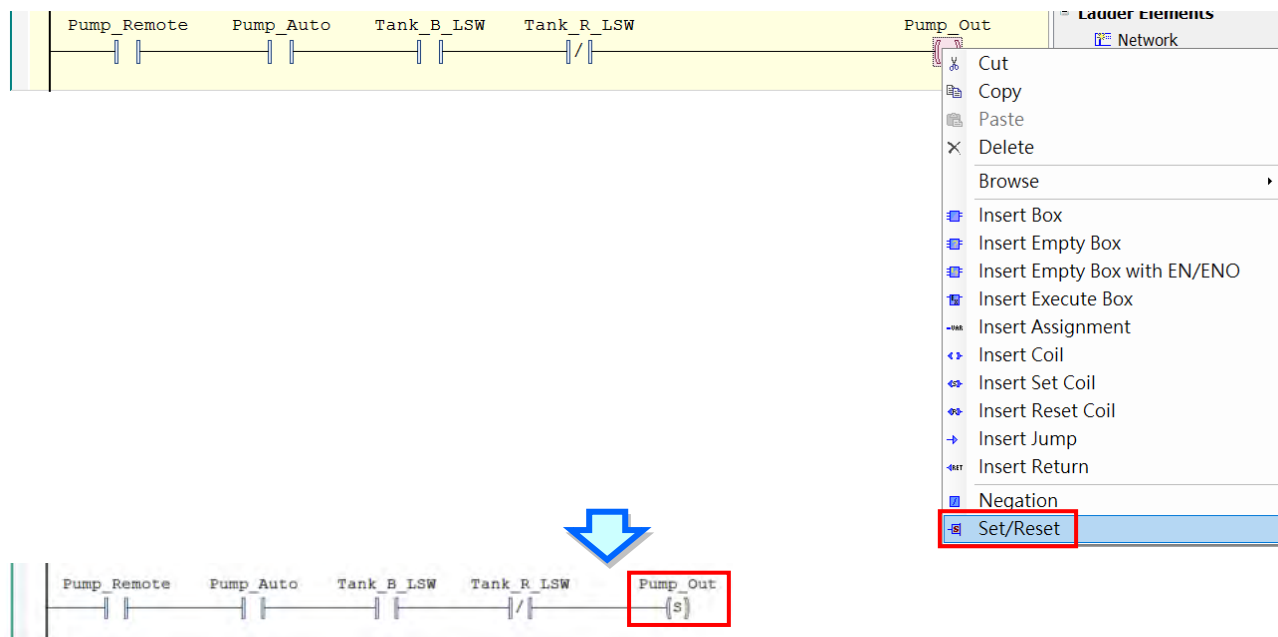
这时要地下水池有水而顶楼水塔没水的条件，所以 Tank_R_LSW 的状态必须为 False 才正确，这时可以使用鼠标右键单击 Tank_R_LSW 的接点图标，系统会显示接单击择的下拉选单，从里面选择我们需要的常闭接点型式 (Negation)。



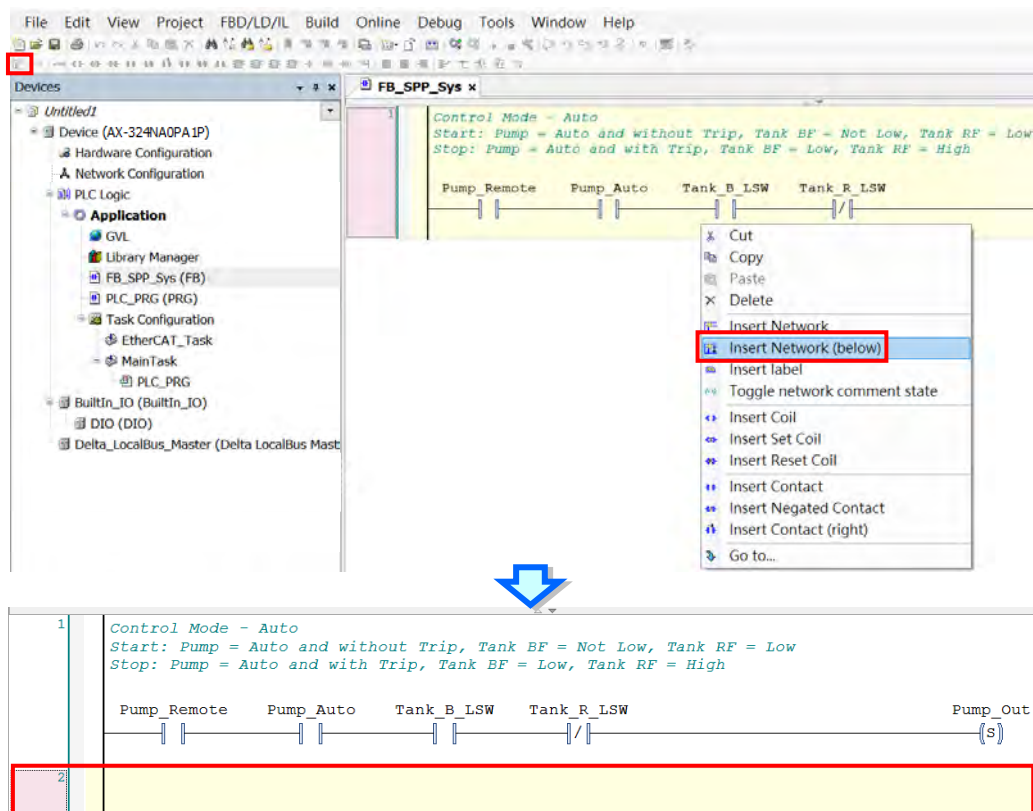
加入输出接点，一样按下上方的  图示或持续按压主画面右方「ToolBox」>「Ladder Elements」下的  Coil，将鼠标拖移到绿框位置，使光标出现+符号后，再放开鼠标左键放置。



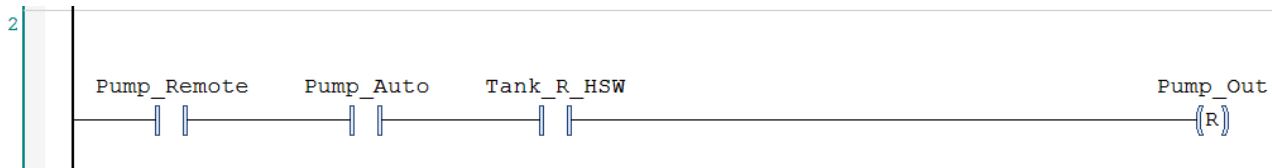
在接点的问号处填入「Pump_Out」，再使用鼠标右键单击该输出接点图标，系统会显示下拉选单，从里面选择我们需要的型式 (Set)，或直接由右侧「ToolBox」选择  Set coil



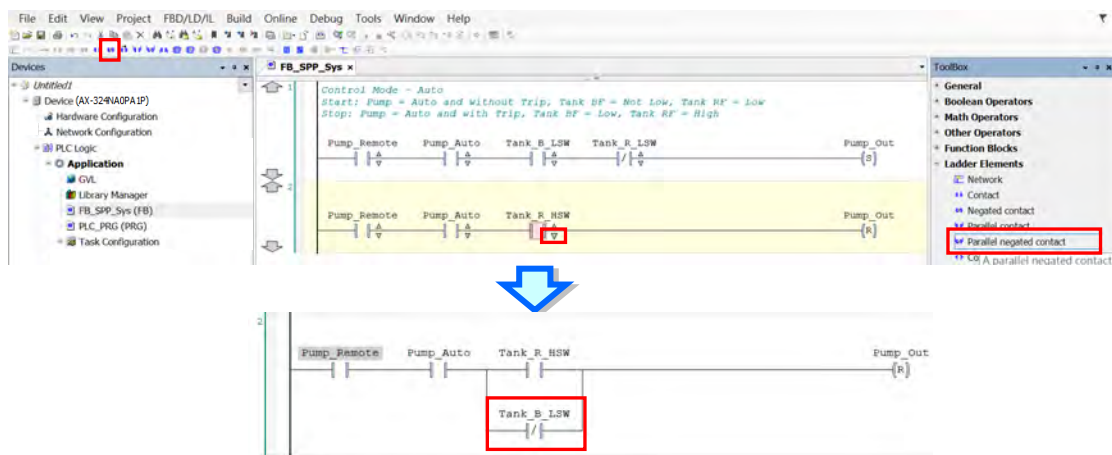
接着新增区段，可在上方找到  新增区段的图标，然后按下图标后新增，或在原区段 1 空白处按下鼠标右键，系统会显示下拉选单，从里面选择 Insert Network (below)。



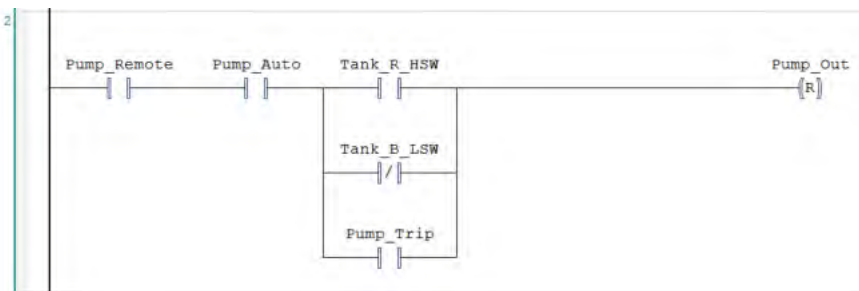
已新增区段 2 后，再依之前的方式建立以下程序内容。



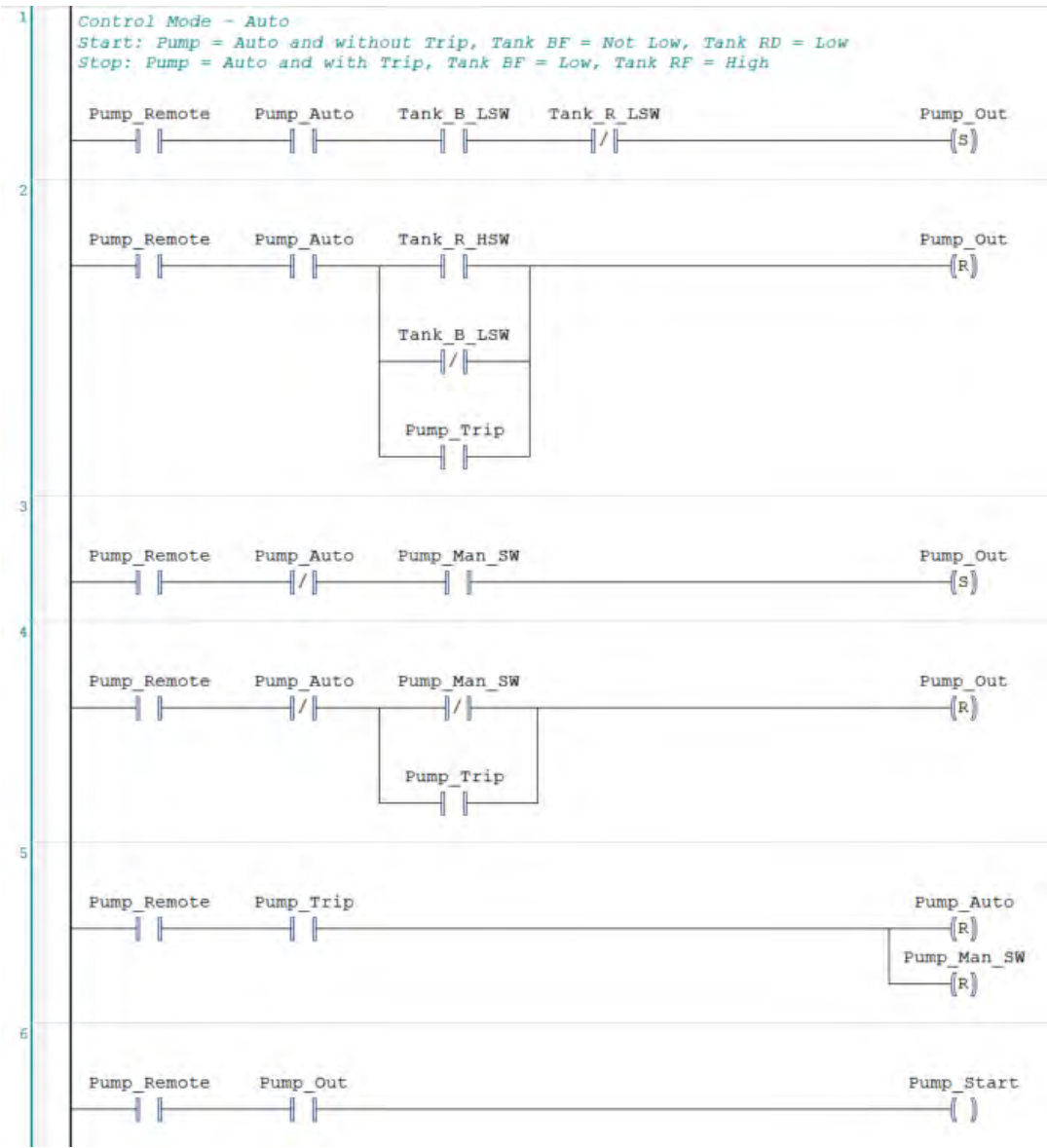
同时要考慮其它的停止條件，如地下水池沒水或泵浦發生故障，所以可先用鼠標左鍵单击欲并聯的接點，再单击上方  圖示來新增常閉接點，或持續按壓主畫面右方「ToolBox」>「Ladder Elements」下的  Parallel negated contact，將鼠標拖移到綠色倒三角形位置處，使光標出現+符號后，再放開鼠標左鍵放置新增，然後在接點的???處填入「Tank_B_LSW」的名稱。



依同樣的操作方式完成以下程序內容。



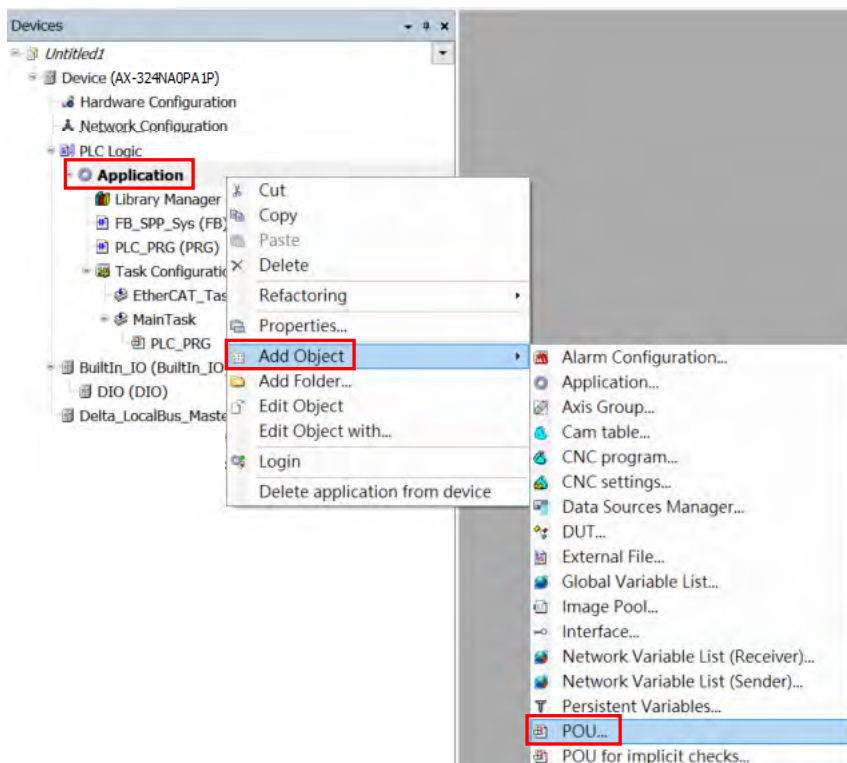
这样就算完成自动的程控部份，接下来请依照同样的操作技巧与方式，完成以下的程序内容。



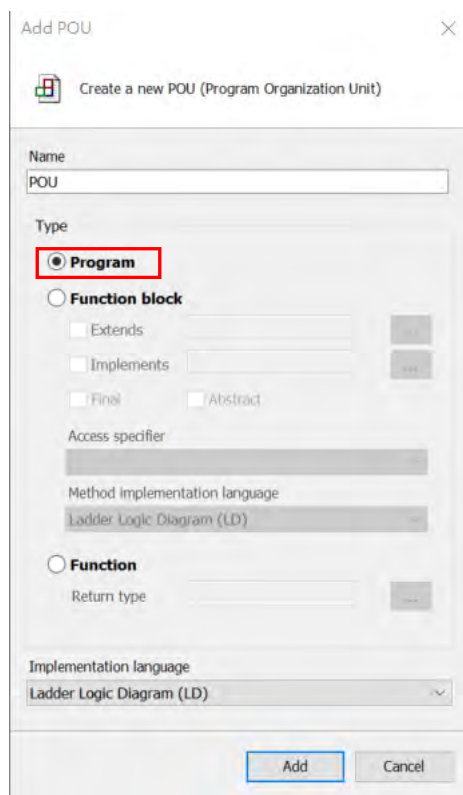
完成后存盘关闭这样就算完成了这个功能块（FB）的开发工作了。

2.4.6 建立主要程序

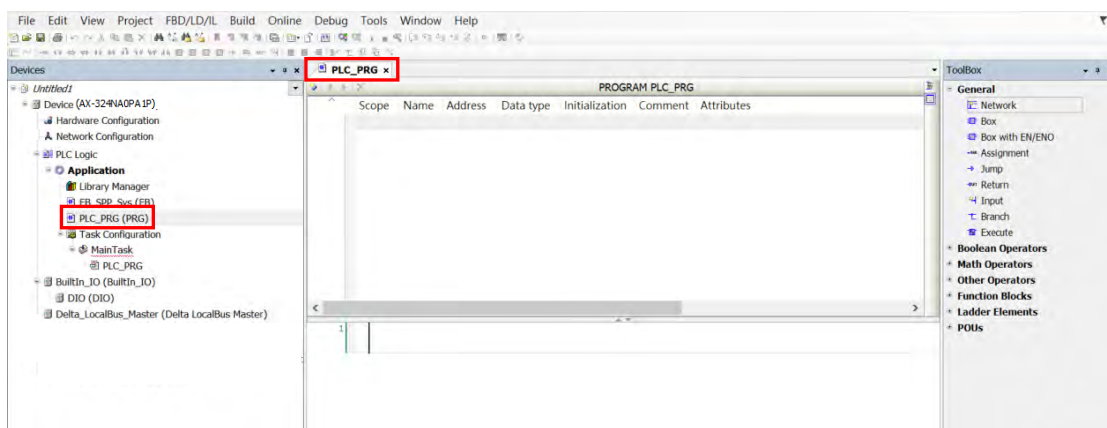
当功能块建立完成后就可以建立主程序了，可以使用建立项目时系统默认已先建立的程序 PLC_PRG，或在主画面左侧的窗口「Devices」内找到「Application」的图标与文字  **Application**，利用鼠标右键的方式选择「Add Object」，再使用左键选择「POU」新增。



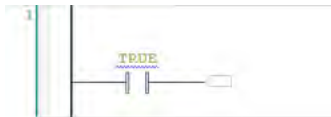
可在「Name」下方框内填入新名称，并选择「Type」为「Program」，内容请参考下图，其余可保持默认值并按下「Add」。



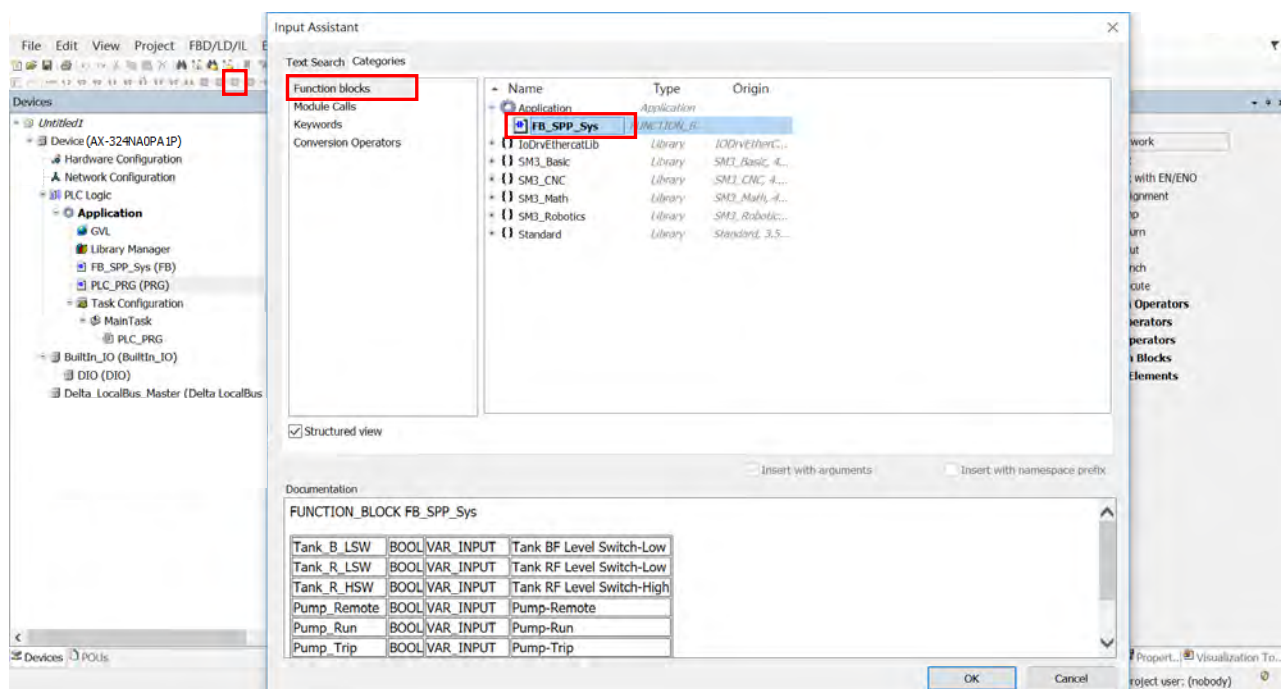
这样在主画面就可以看到主程序的程序编辑窗口。



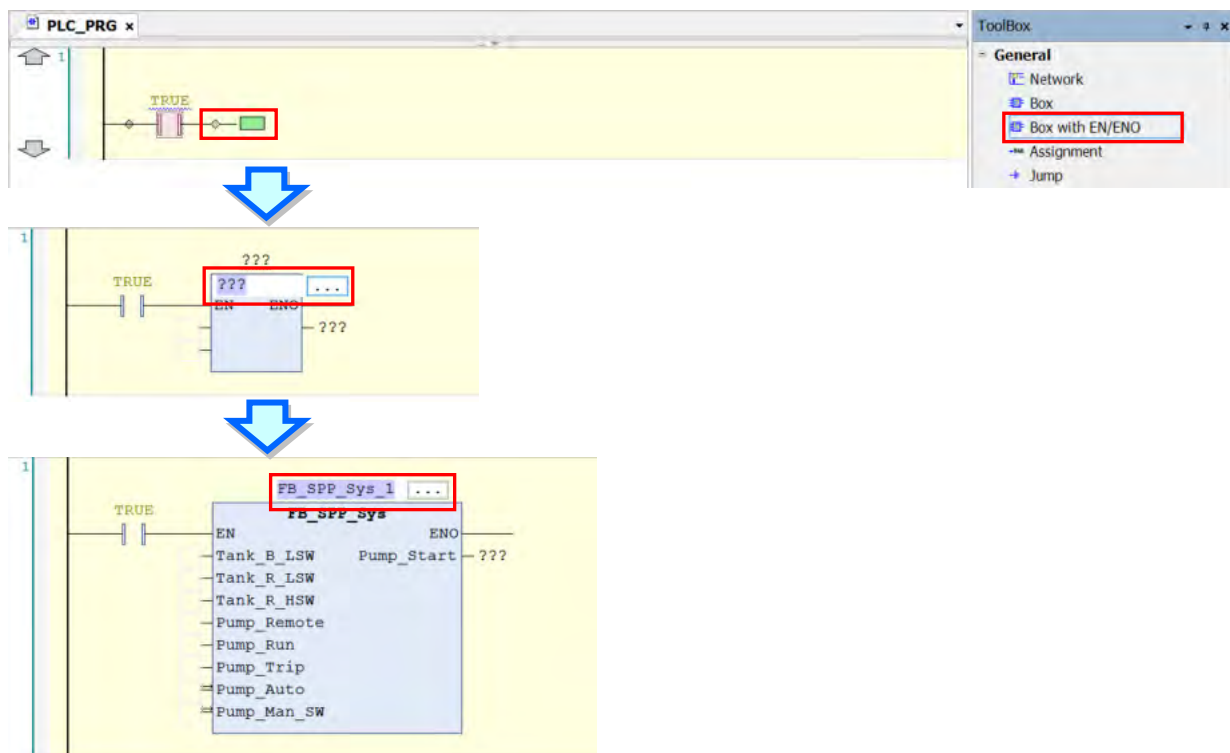
主程序的编辑环境与功能块一样，唯一的差别是功能块(FB)部份需要被调用才会执行，而程序(Program)部分是加入 Task 列表就会执行的，在本范例这里就不先规划主程序的区域符号了，所以请依照前面所学习的技巧，完成以下的程序内容。



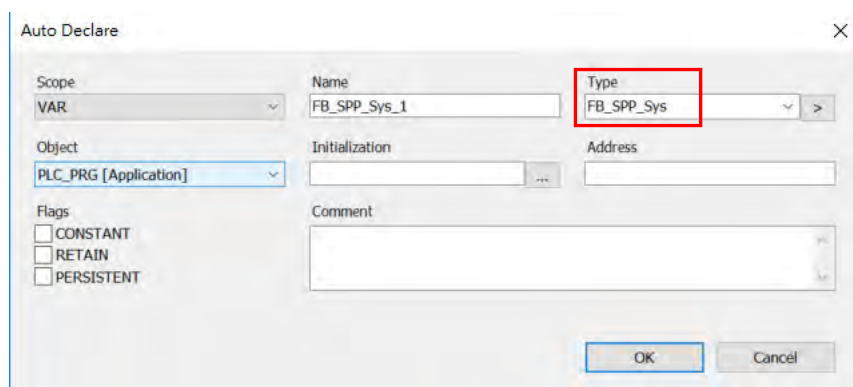
接着准备调用功能块(FB)使用，一样先在区段空白处使用鼠标左键单击一次，然后在上方工具栏找到 Insert Box with EN/ENO 的图示  并按下鼠标左键，接着就会跳出选择协助窗口，请在「Categories」下方左侧的地方选择「Function blocks」，并在右侧「Application」内选择「FB_SPP_Sys」，然后按下「OK」。



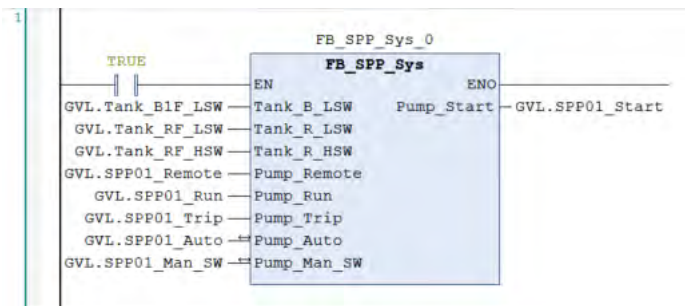
上述新增功能块动作也可由持续按压主画面右方「ToolBox」>「General」下的 **Box with EN/ENO**，将鼠标拖移到绿框位置，使光标出现+符号后，放开鼠标左键放置新增，再直接在问号处输入功能块名称，或按下图示 **...** 后会跳出输入协助窗口来选择功能块。



此时键盘按下 **Enter** 会自动宣告一个数据区块给这个功能块使用，这个数据区块可以建立在全局符号或是区域符号，在这里我们把它建立在主程序的区域符号内，而建立的方式与建立一般的符号一样，只是在数据类型的部份要指定「**FB_SPP_Sys**」的名称，此处系统会自动输入数据类型与功能块相同名称。

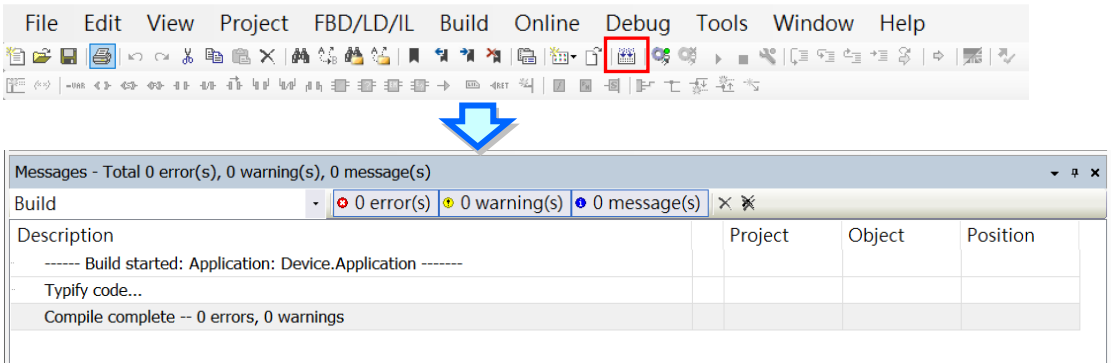


再请依照下面的内容，将对应的全局符号名称填入到引脚内。



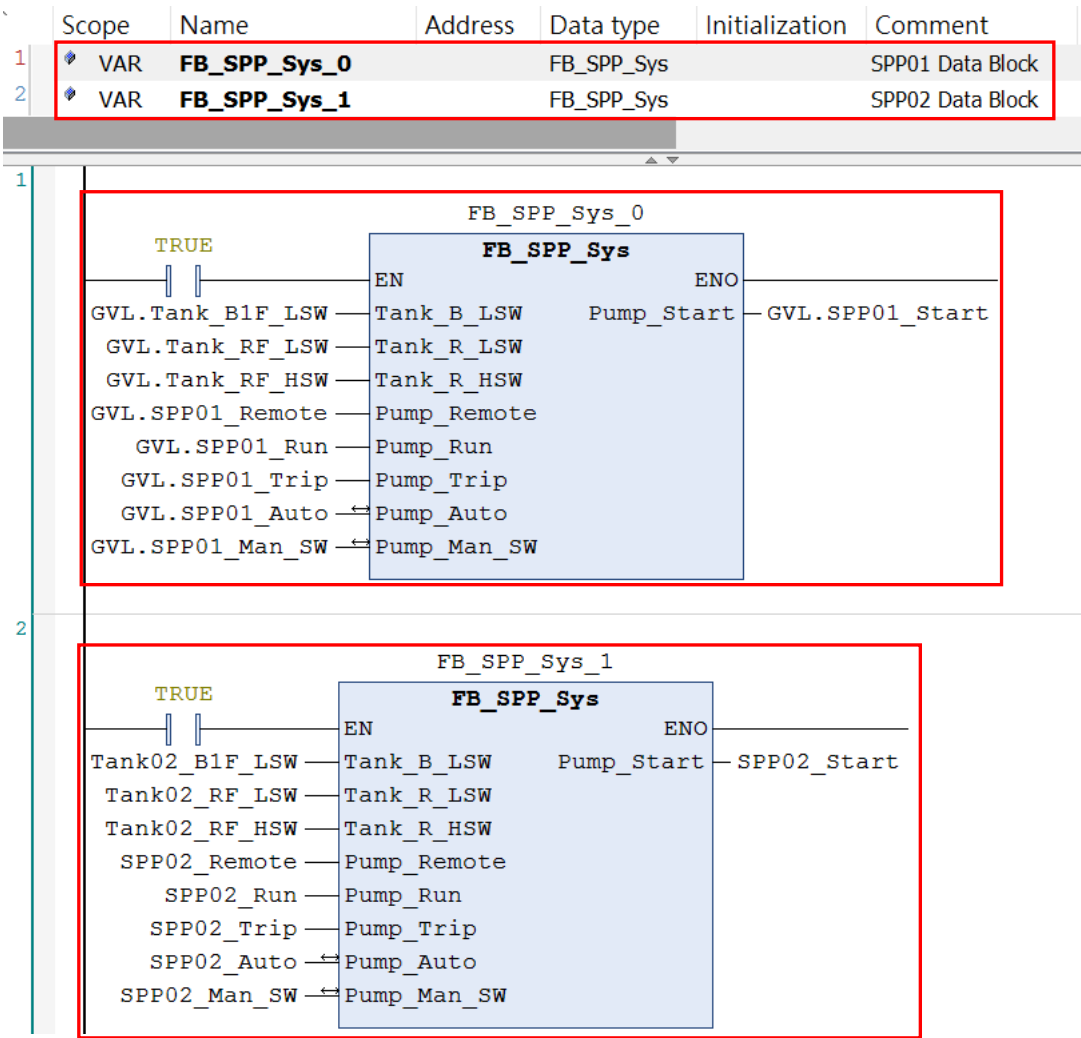
存盘后这样整个程序就算编写完成了。

如果要测试整个项目是否正确，则可以使用项目编译的按键来进行编译测试，项目编译包含了检查文法的功能同时编译范围包含所有的程序与功能块内容。



注：本范例的内容为了让用户容易了解，所以以较简易的程序内容来展现，真的控制程序内容其实是比较复杂的，因为还可能会加入启动失败的 Fail 状态判断功能、运转时间的记录功能...等，这时就可以拖曳主窗口右侧「ToolBox」窗口内或「Library」中的其它指令来帮助程序的编写。

另外，若有第二套系统要建立时，则只要再调用一次「FB_SPP_Sys」，并把第二套的符号名称对应填入，即完成整个程序的编写，马上可以体会到功能块的便利性，但请注意在数据区块的使用上请勿共享，除非用户认为功能块内 VAR_OUTPUT 与 VAR 的区域符号在数值共享上没有影响，否则请另外建立一个功能块类型的符号给第二套系统使用，如下图就是分开建立的应用。



第3章 建立运动控制项目

目录

3.1	基础工作准备	3-3
3.1.1	硬件	3-3
3.1.2	软件	3-4
3.1.3	工具与材料	3-4
3.2	EtherCAT 伺服轴项目建立	3-5
3.2.1	系统硬件配置	3-5
3.2.2	项目建立与程序编写	3-7
3.2.2.1	项目开启	3-7
3.2.2.2	轴参数设定	3-13
3.2.2.3	项目程序编写	3-20
3.2.3	程序监控	3-30
3.2.3.1	设备与计算机链接设定	3-30
3.2.3.2	程序监控流程	3-32
3.2.3.3	示波器监控	3-39
3.3	Pulse Out 伺服轴项目建立	3-47
3.3.1	系统硬件配置	3-47
3.3.2	项目建立与程序编写	3-49
3.3.2.1	项目开启	3-49
3.3.2.2	轴参数设定	3-53
3.3.2.3	设定伺服参数	3-56
3.3.2.4	项目程序编写	3-62
3.3.3	程序监控	3-75
3.3.3.1	设备与计算机链接设定	3-75
3.3.3.2	程序监控流程	3-77
3.3.3.3	示波器监控	3-81

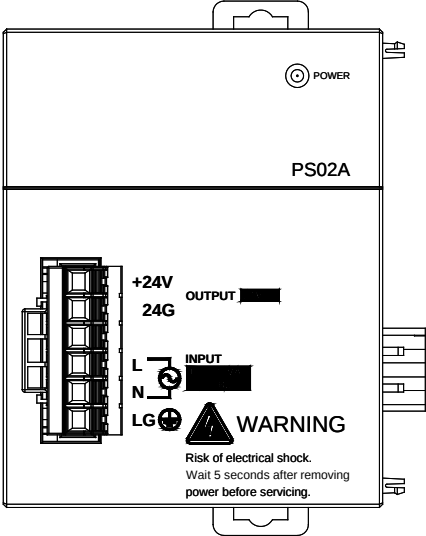
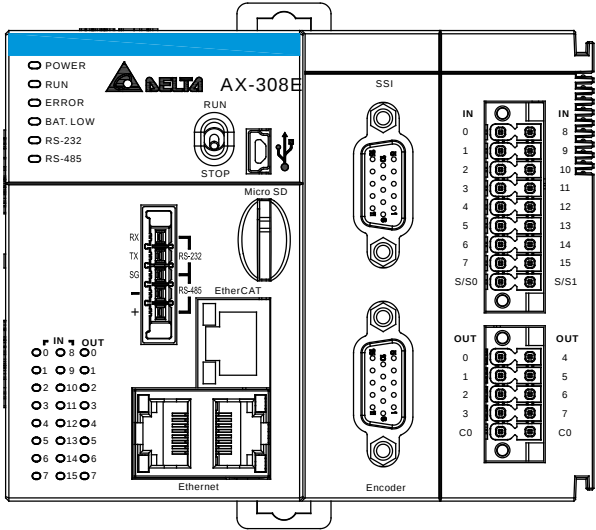
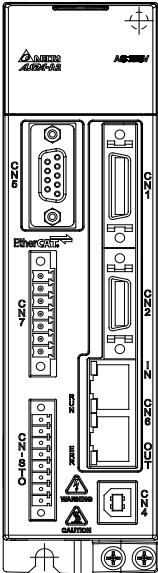
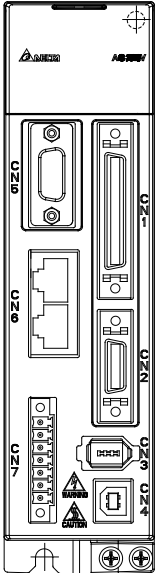
3.4	虚轴作为同步主轴项目建立	3-89
3.4.1	系统硬件配置	3-89
3.4.2	项目建立与程序编写	3-90
3.4.2.1	项目开启	3-90
3.4.2.2	轴参数设定	3-97
3.4.2.3	项目程序编写	3-106
3.4.3	程序监控	3-117
3.4.3.1	设备与计算机链接设定	3-117
3.4.3.2	程序监控流程	3-119
3.4.3.3	示波器监控	3-126
3.5	内建编码器作为同步主轴项目建立	3-134
3.5.1	系统硬件配置	3-134
3.5.2	项目建立与程序编写	3-136
3.5.2.1	项目开启	3-136
3.5.2.2	轴参数设定	3-143
3.5.2.3	项目程序编写	3-154
3.5.3	程序监控	3-167
3.5.3.1	设备与计算机链接设定	3-167
3.5.3.2	程序监控流程	3-169
3.5.3.3	示波器监控	3-175

3.1 基础工作准备

第三章中会说明机种常使用的运动控制范例，让用户依据范例内容建立基础运动控制项目。第 3.1 节会说明本范例所使用的软件与硬件需事前准备相关工具。

3.1.1 硬件

范例所需要的硬件如下：

1. AS 系列电源模块 (AS-PS02A) 	2. AX3 系列主机 (AX-308E) 
3. 通讯型伺服驱动器 (ASDA-A2-0421-E) 	4. 脉冲型伺服驱动器 (ASDA-A2-0421-M) 

3.1.2 软件

范例所需要的软件如下：

- DIADesigner-AX V1.0.0 以上

3.1.3 工具与材料

范例所需要的工具与材料如下：

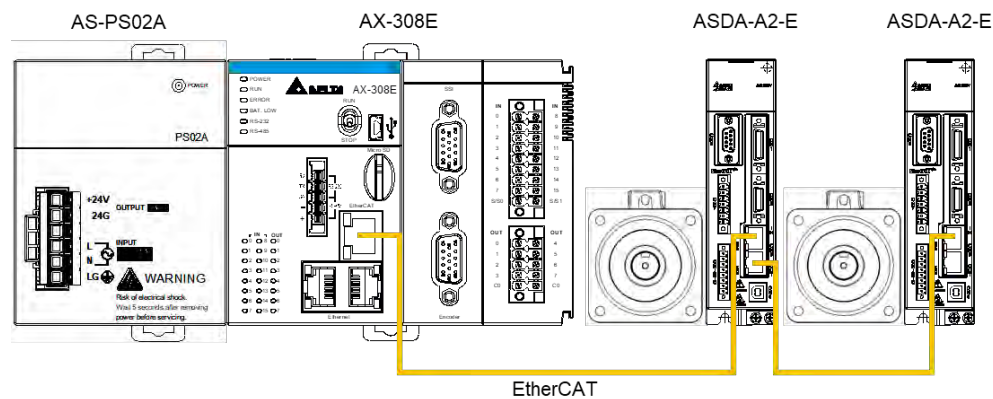
- 个人工作计算机 (已安装上述需求之软件) x 1 台
- 100~240VAC 50/60Hz 的电源供应座 x 1 个
- 24VDC 电源供应器 x 1 个
- 电线 x 1 捆
- 针形端子压线工具 x 1 个
- 针形端子 x 20 个以上
- 一字螺丝起子 x 1 支
- USB 连接线或是网络线 x 1 条

3.2 EtherCAT 伺服轴项目建立

此范例为两轴伺服控制并来回运动。

3.2.1 系统硬件配置

AX-308 通过 EtherCAT 端口口通讯连接 2 台伺服驱动器 (ASDA-A2-E)，控制两轴伺服电机定位。



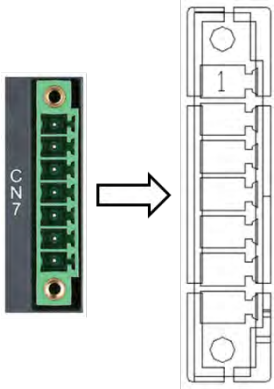
● 材料与工具

编号	名称	料号	数量
1	AS 系列电源模块	AS-PS02A	1
2	AX3 系列主机	AX-308EA0MA1T	1
3	电源供应器	DVP-PS02	1
4	通讯型伺服驱动器	ASDA-A2-0421-E	2
5	伺服电机	ECMA-C10604ES	2
6	伺服动力线	ASD-ABPW0003	2
7	伺服编码线	ASD-ABEN0003	2
8	EtherCAT 连接线 (1 米)	UC-EMC010-02A	2

● 硬件极限配置

配置硬件极限可配置于 CN1 与 CN7，此范例以 CN7 为主。

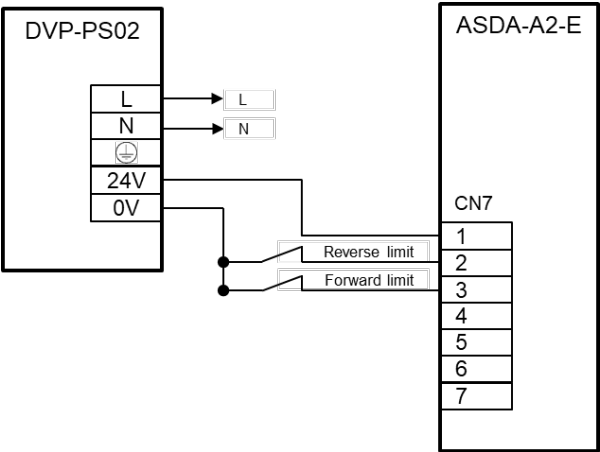
使用外部 DC24V 并配置反极限 (Reverse limit) 与正极限 (Forward limit)，配置完硬件极限，还须针对 ASDA-A2-E 的伺服参数设定 DI9~DI10 引脚功能。(两台极限配置相同)



■ CN7 DI 信号介绍

Pin No.	信号名称	端子记号	功能及说明
1	VDD 24 V 电源	COM+	VDD (24 V) 电源 · 与 CN1 的 Pin 相同
2	扩展 DI9	EDI9-	数字输入引脚 9-
3	扩展 DI10	EDI10-	数字输入引脚 10-
4	扩展 DI11	EDI11-	数字输入引脚 11-
5	扩展 DI12	EDI12-	数字输入引脚 12-
6	扩展 DI13	EDI13-	数字输入引脚 13-
7	扩展 DI14	EDI14-	数字输入引脚 14-

■ ASDA-A2-E 极限配置



注：ASDA-A2-E 信号配置详细信息可参考 **ASDA-A2 系列应用技术手册**

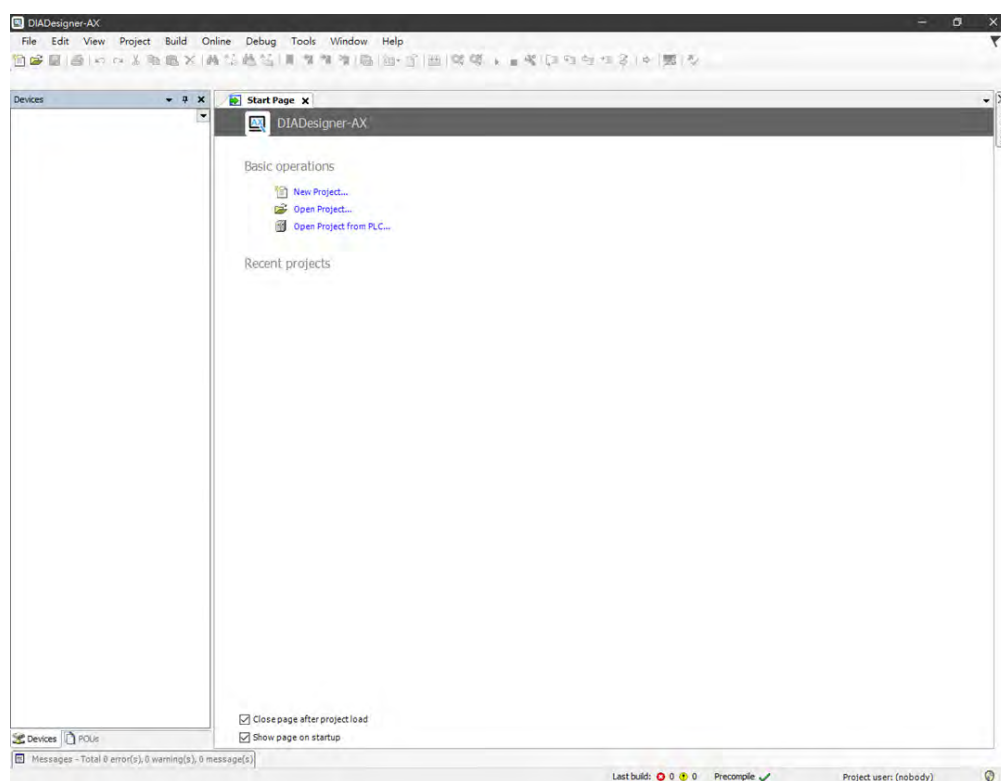
3.2.2 项目建立与程序编写

3.2.2.1 项目开启

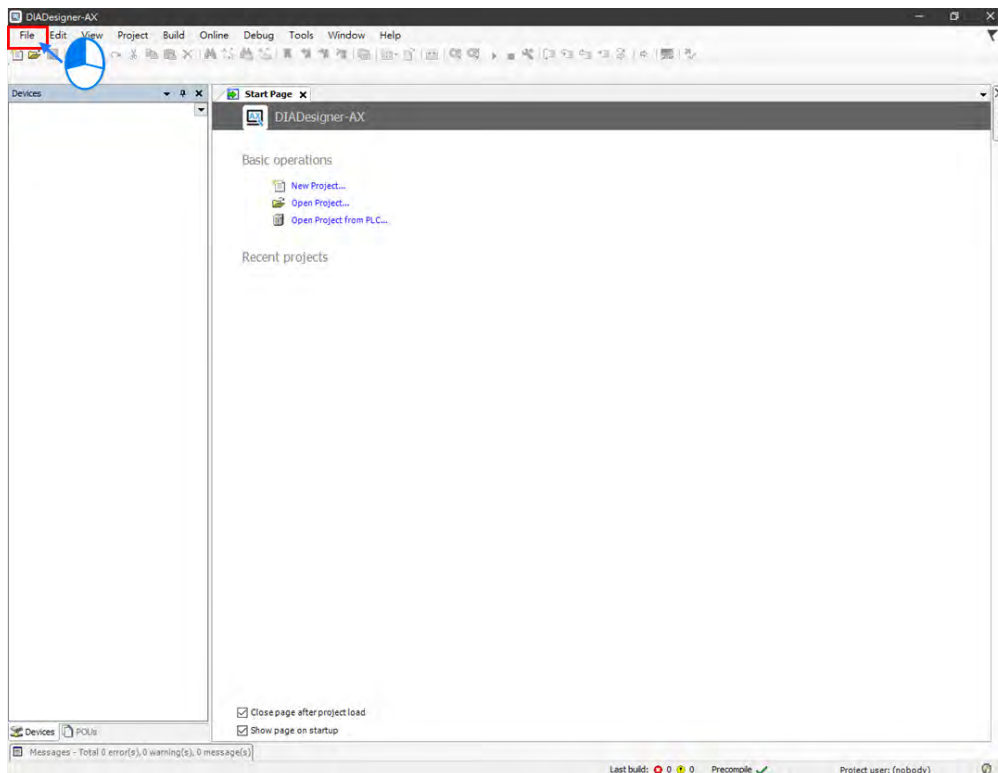
1. 首先 AX-308 编写程序软件为 DIADesigner-AX，所以需先启动软件。



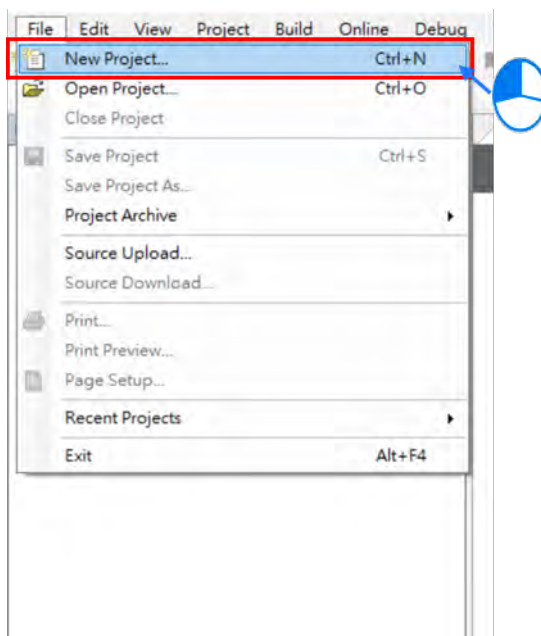
2. 启动 DIADesigner-AX 的画面如下



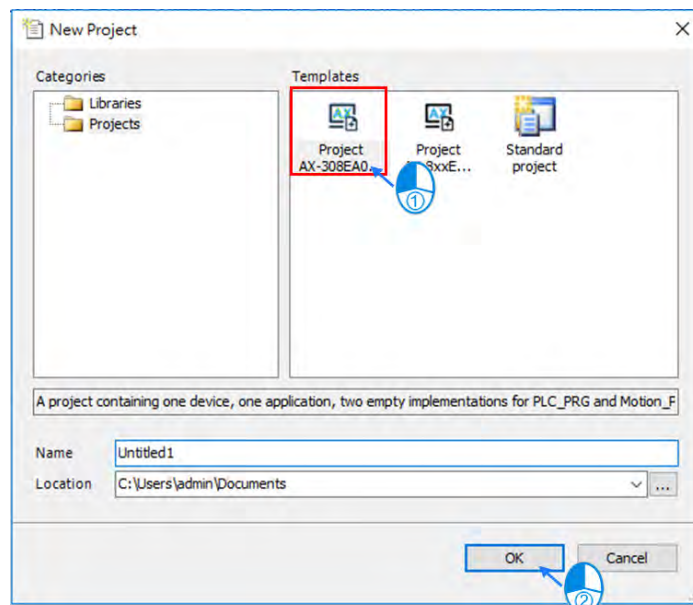
3. 单击「File」后双击鼠标左键



4. 在「File」下拉选单单击「New Project」以开启一个新项目。

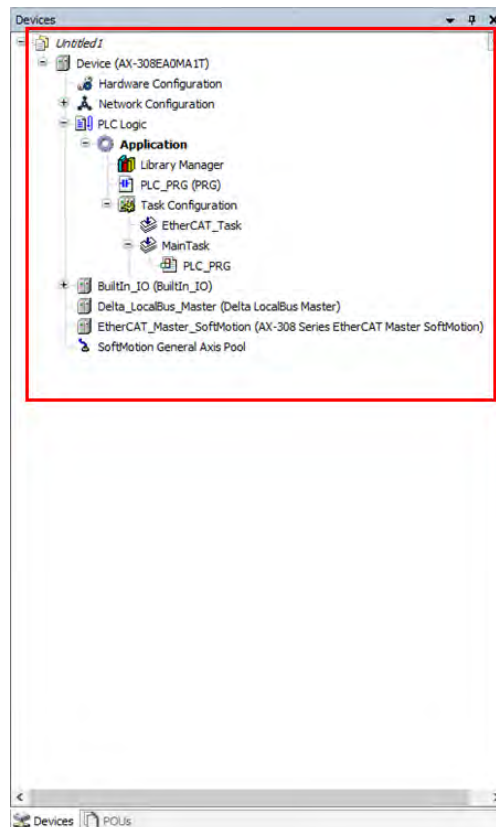


5. 进入「New Project」画面后选择「Project AX-308EA0MA1T*」，可在「Name」处输入项目名称，及「Location」处输入项目存取路径。

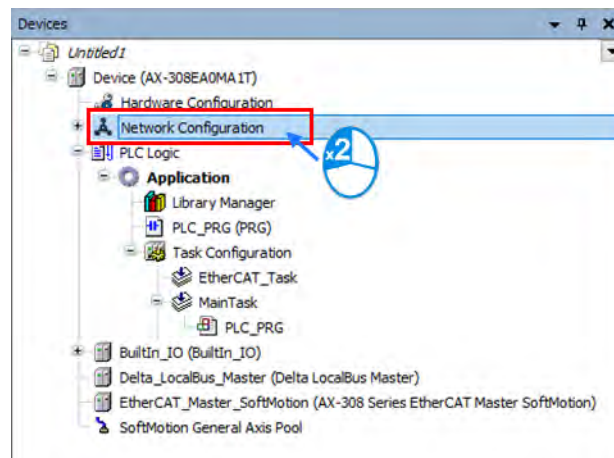


*注：DIADesigner-AX 版本于 V1.1.0 以上的操作为点选「Standard Project」后选择机种。

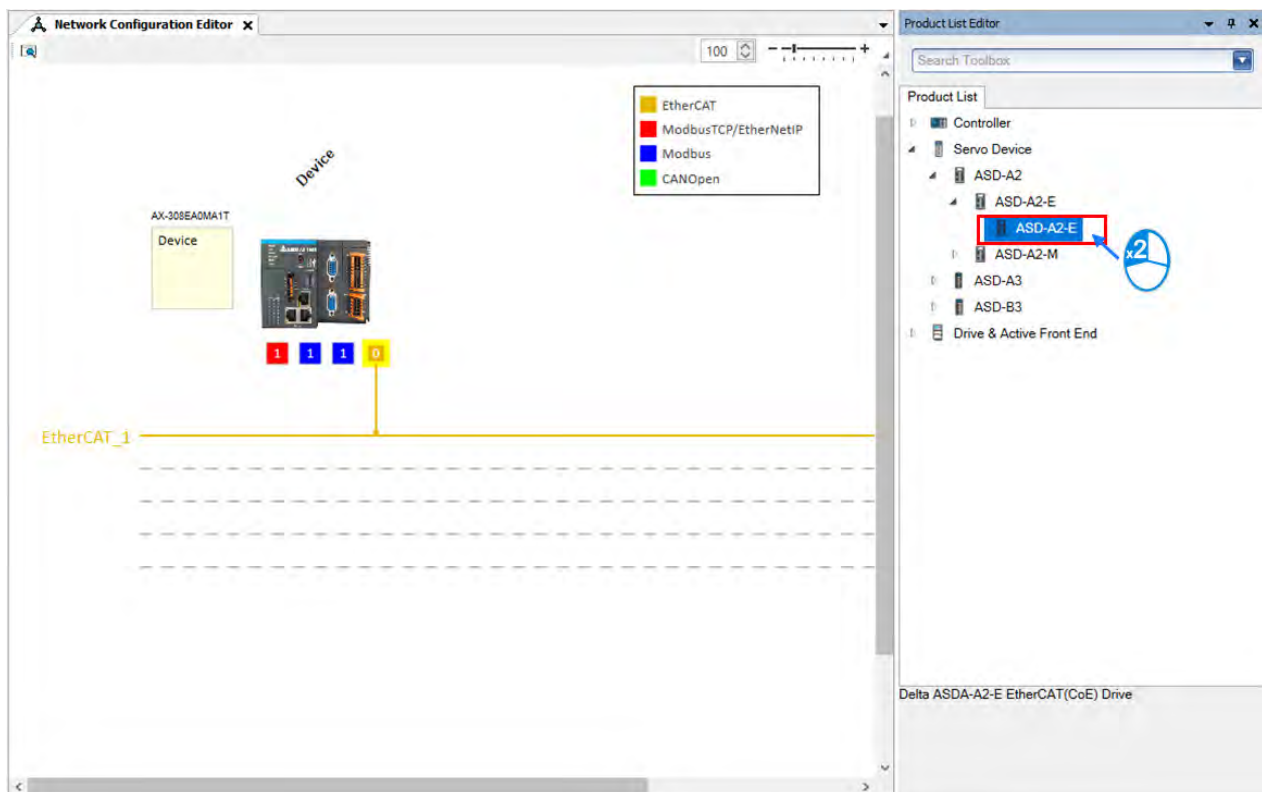
6. 进入项目画面如下，左边为项目树，可双击左键以开启程序画面或是模块相关设定。



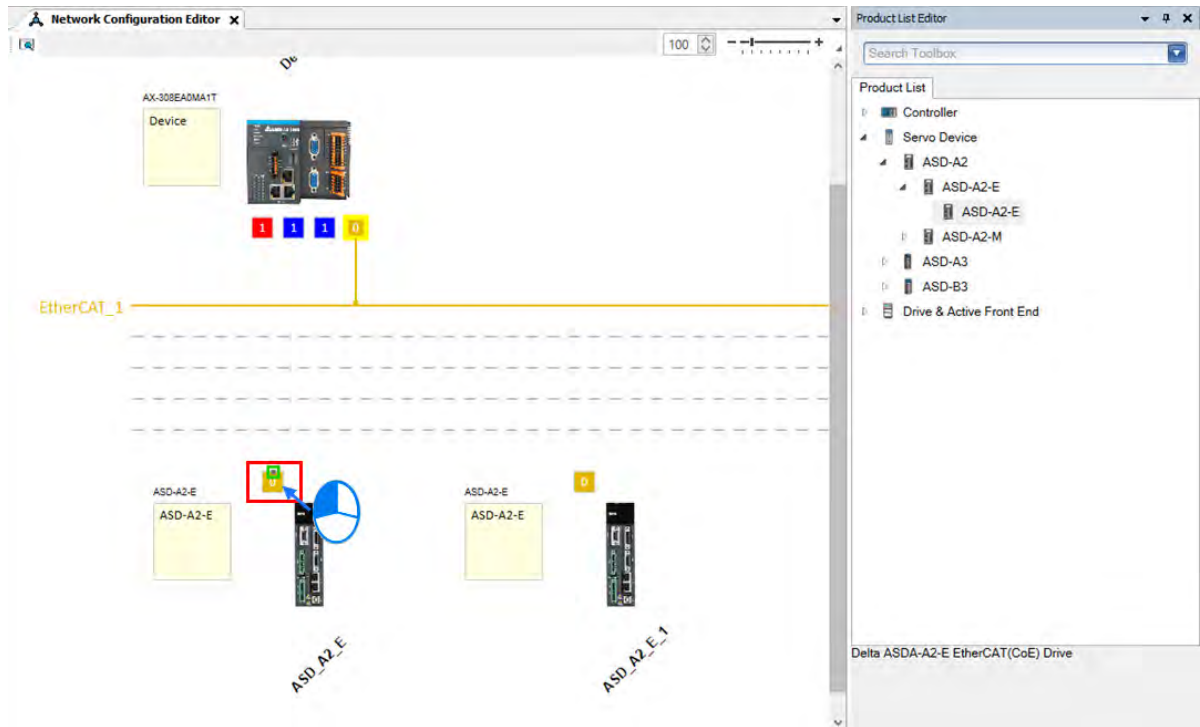
7. 先建立系统架构的伺服轴，双击「Network Configuration」。



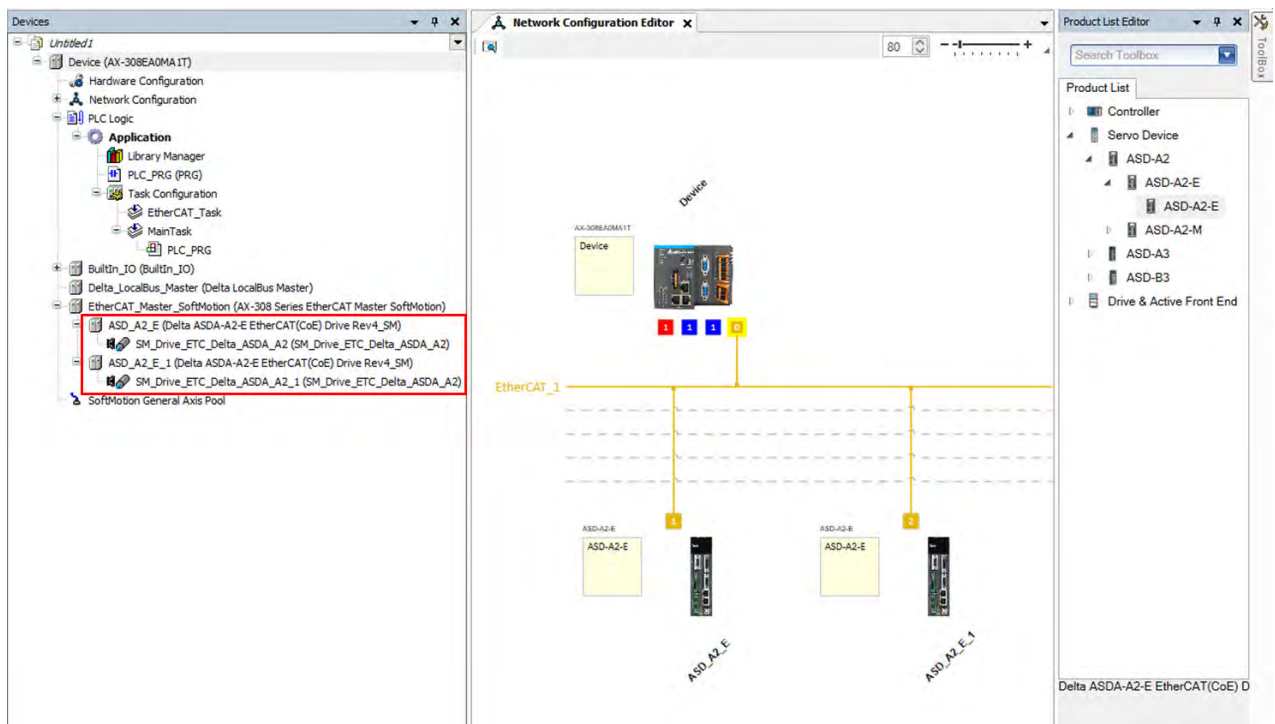
8. 进入「Network Configuration」画面后，选择 EtherCAT 所配置的 Slave 型号，双击左键以新增轴。



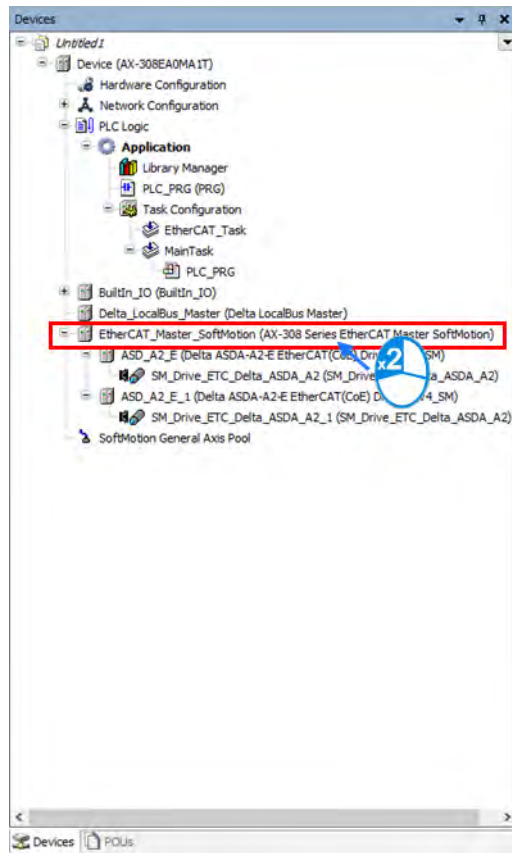
9. 将鼠标移至 0 上方，按压左键将黄色线连接至上方 EtherCAT_1。



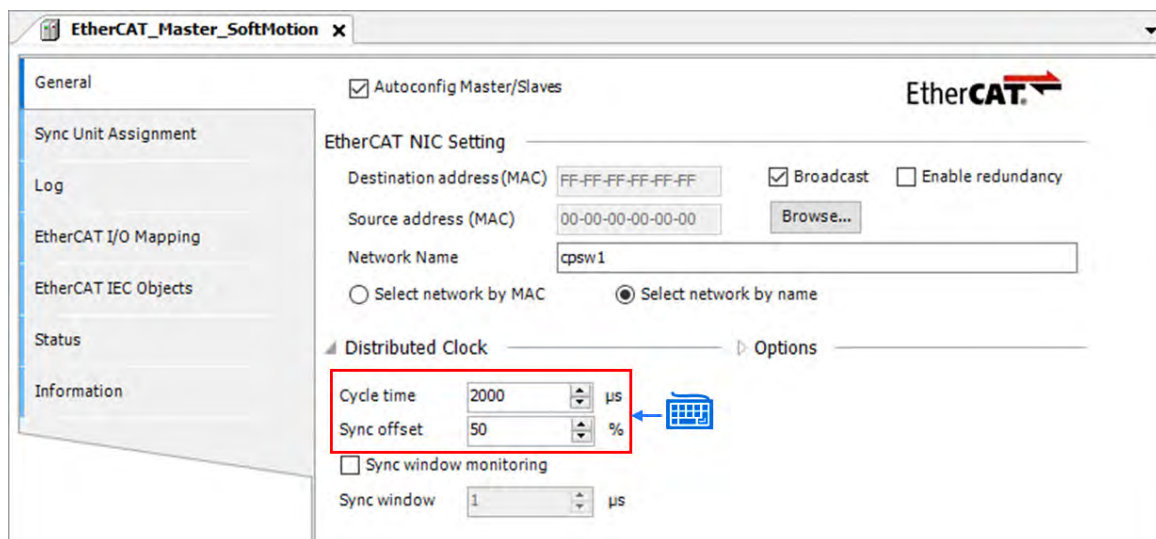
10. 将 Slave 都联机至 EtherCAT_1 后，左方树形图才会显示 Slave，代表此 Slave 有配置在 EtherCAT 通讯中。



11. 双击「EtherCAT_Master_SoftMotion」。

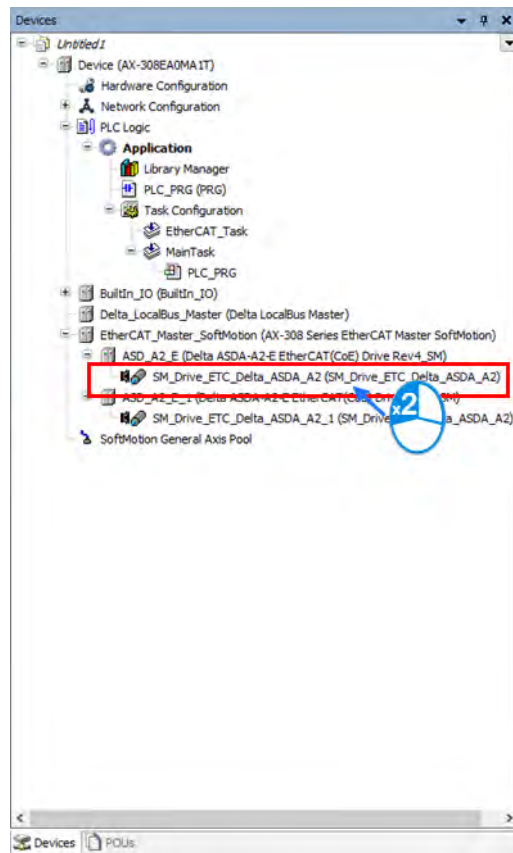


12. 进入「EtherCAT_Master_SoftMotion」画面后，设定「Cycle time」为 2ms 及「Sync offset」为 50。

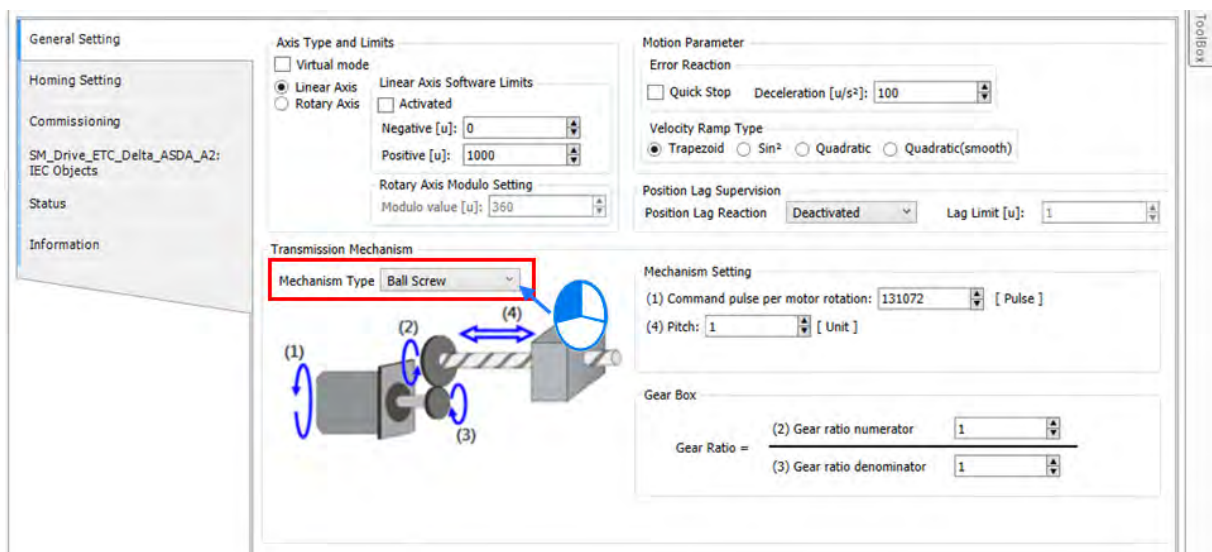


3.2.2.2 轴参数设定

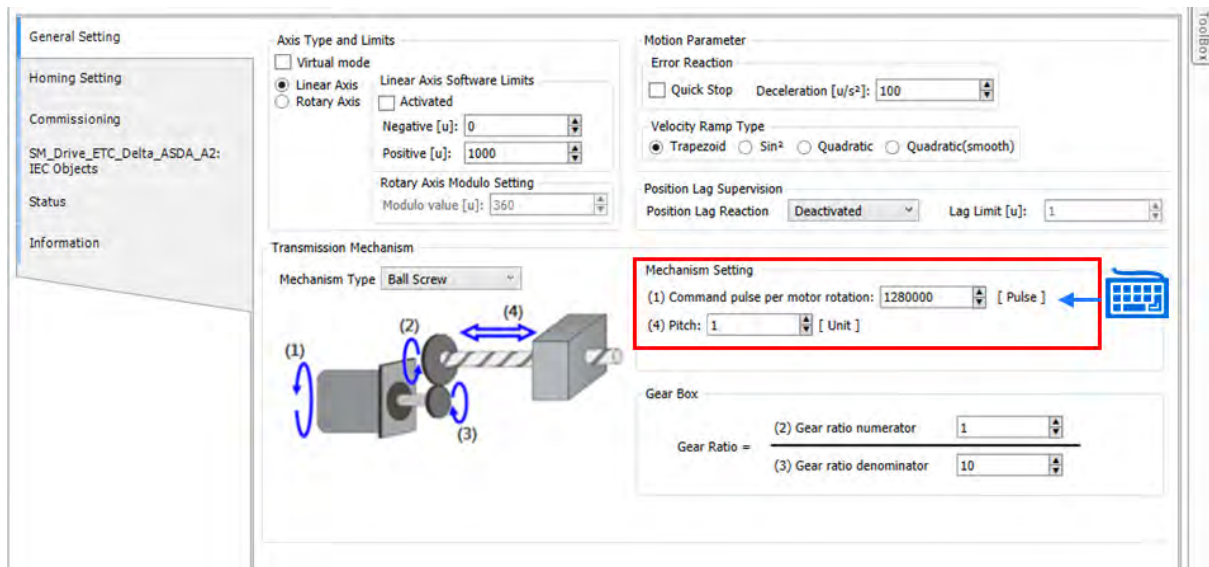
1. 首先选择第一轴「SM_Drive_ETC_Delta_ASDA_A2」后，单击左键两下。



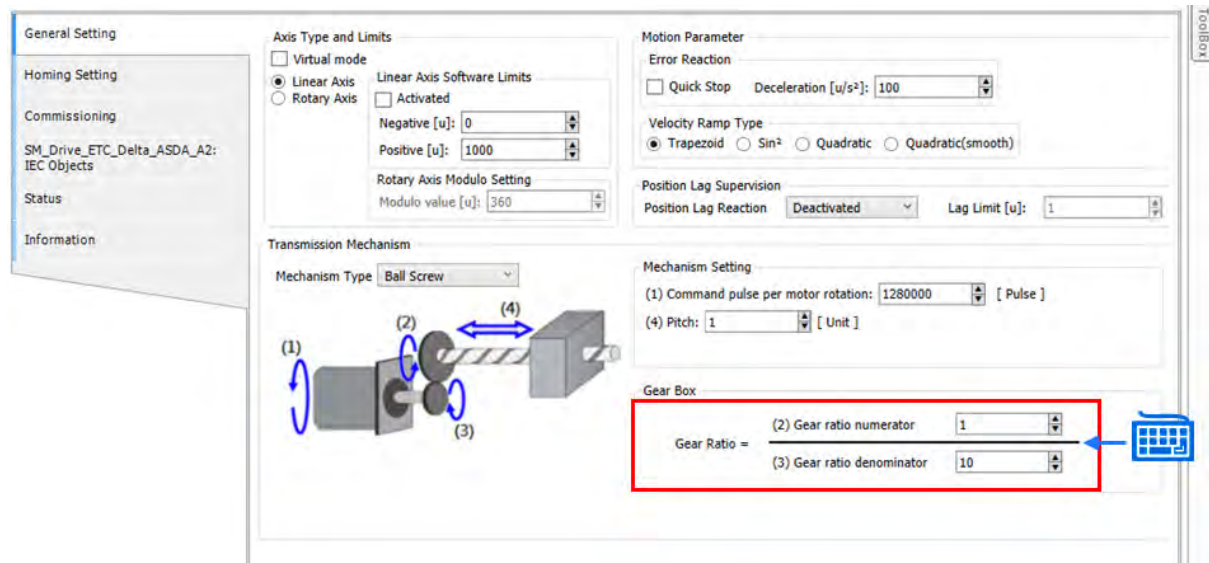
2. 进入「SM_Drive_ETC_Delta_ASDA_A2」画面后，此范例为螺杆机构，所以选择「Ball Screw」机构。



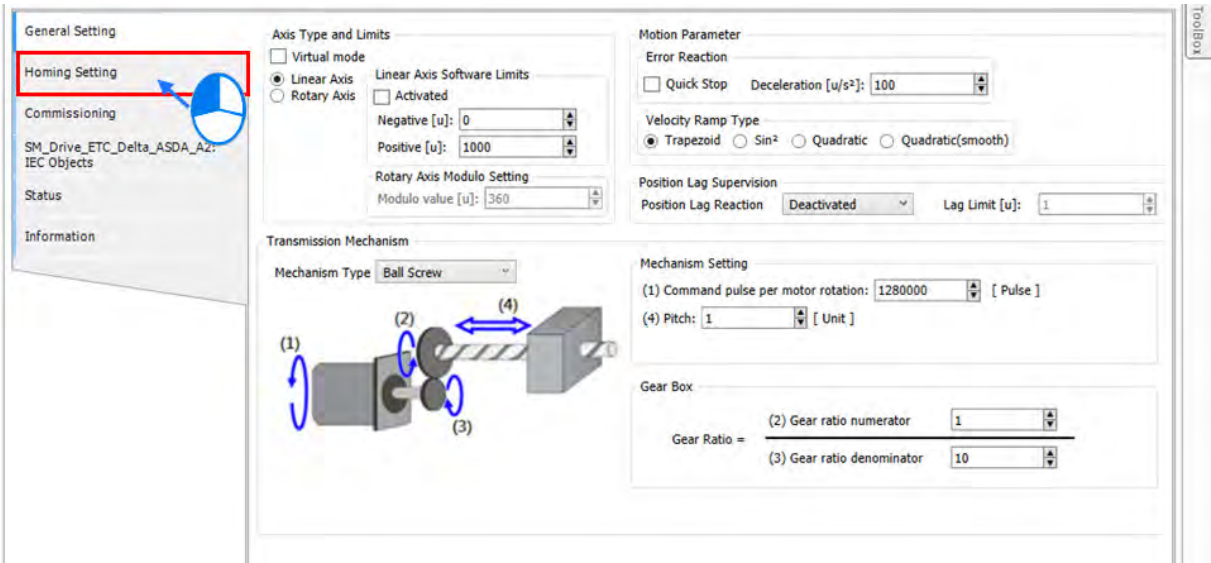
3. 实体机构选择完毕再设定齿比关系，一圈脉冲数（Command pulse per motor rotation）输入 1,280,000，螺杆菌一圈 Pitch 为 1mm，此时在 Pitch 输入 1（伺服端齿比关系 $\frac{P1-44}{P1-45} = \frac{\text{电机一圈脉冲数}}{\text{Command Pulse per rotation}}$ ）。



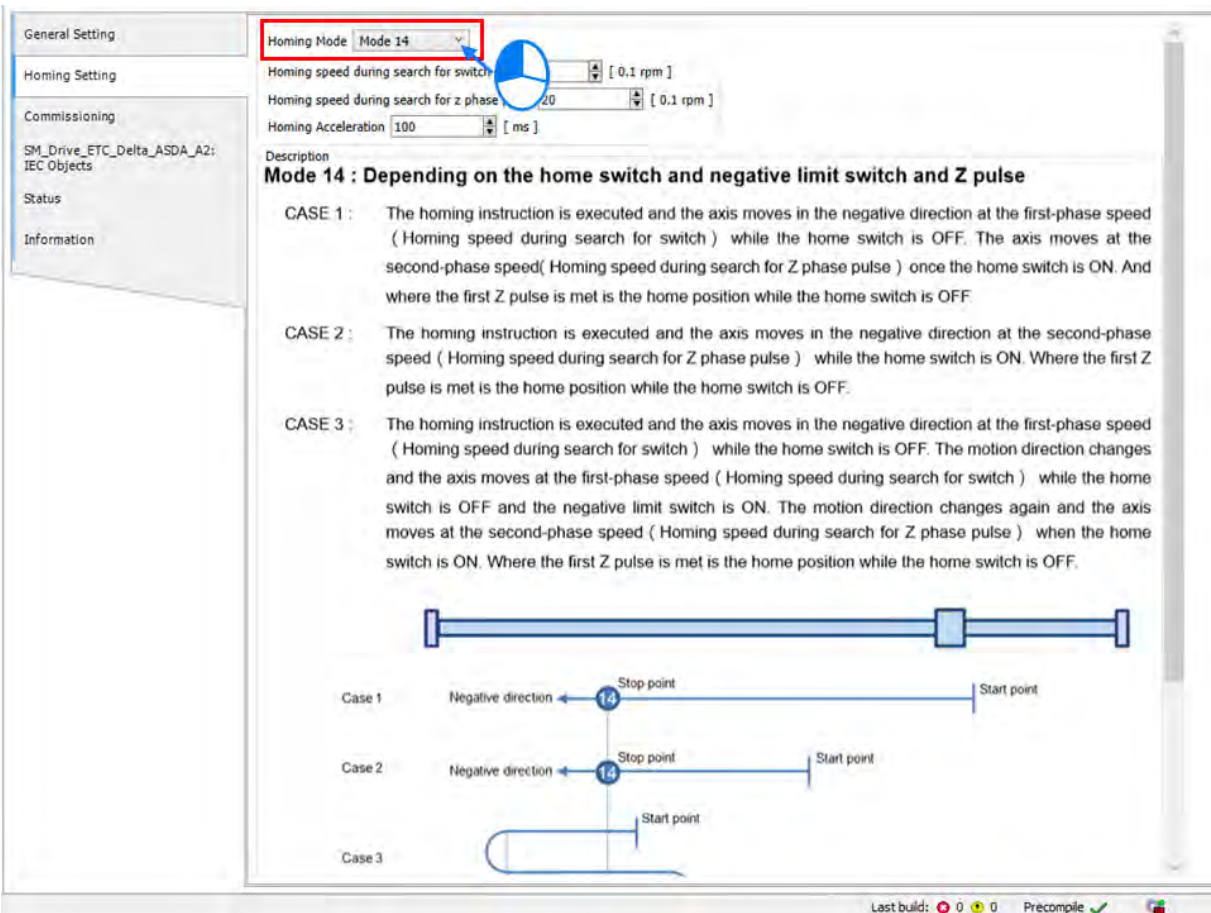
4. 因机构上装置减速机 10 : 1，所以在下方输入减速机的比例为 10 : 1。



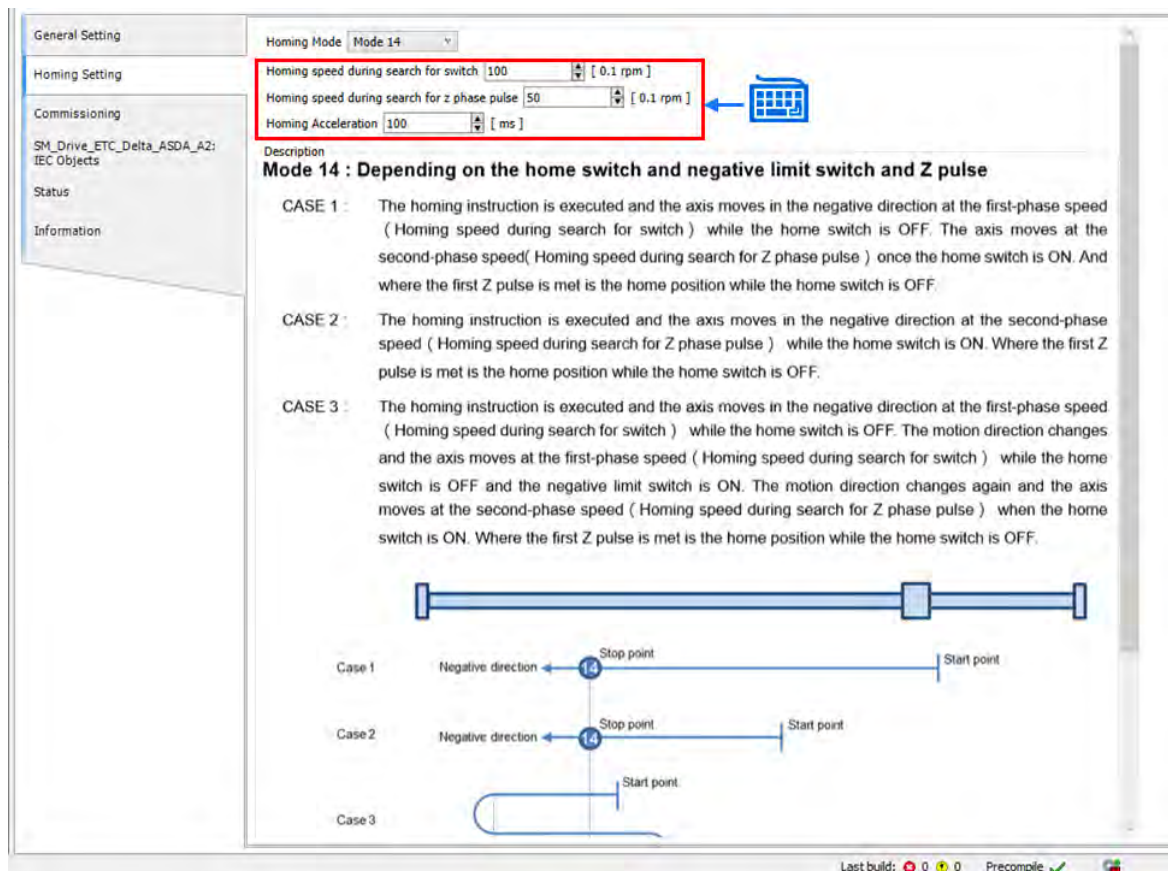
5. 将齿比关系设定完毕后，单击「Homing Setting」。



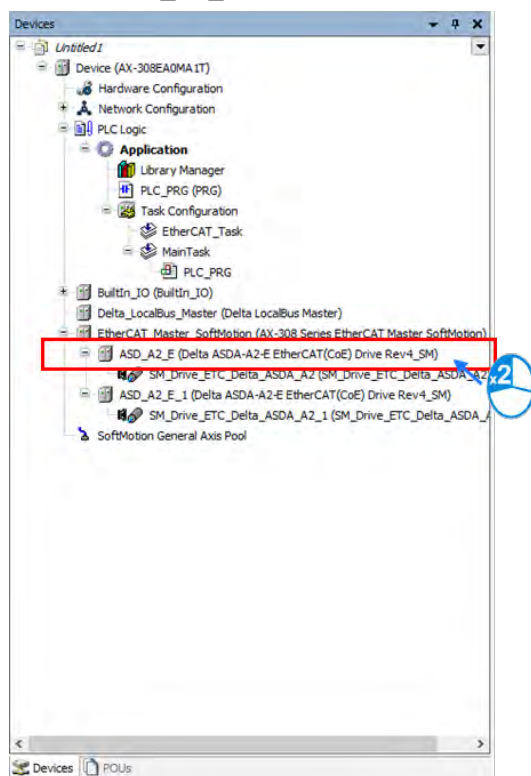
6. 依照硬件极限配置选择对应回原点模式，每种模式请参考下方详细说明。



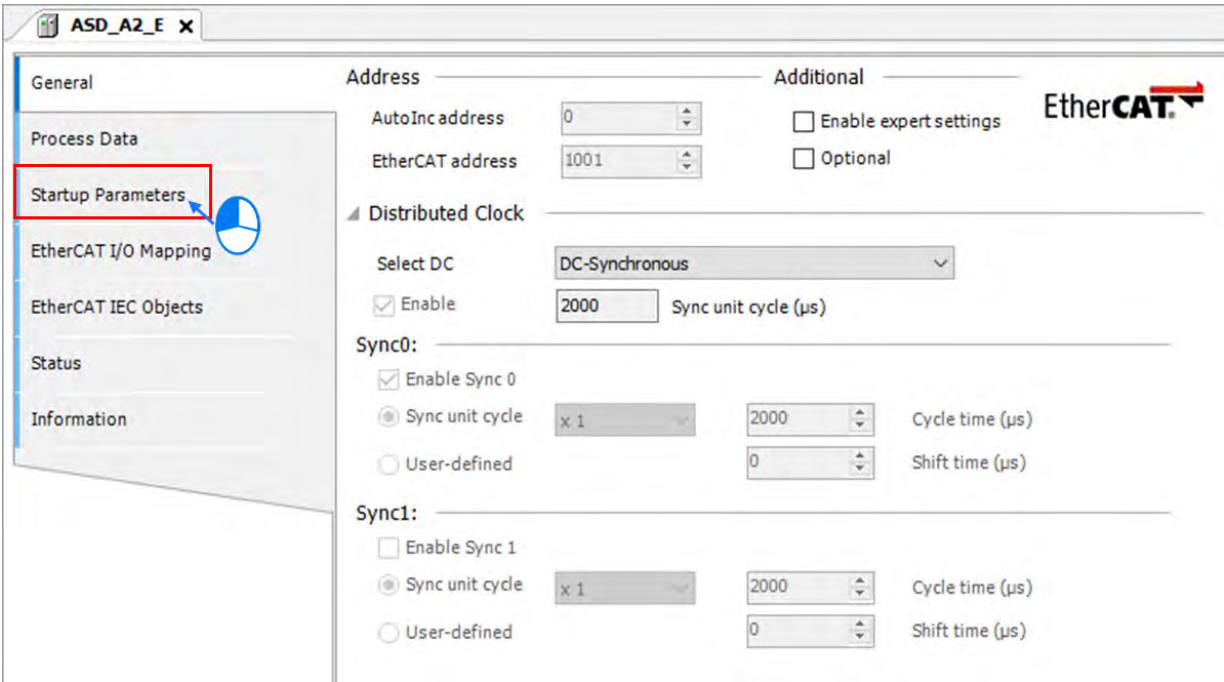
7. 再设定回原点的速度，此速度需搭配机构调整为适当速度，不可调整过快，以免超出硬件极限。



8. 将齿比与原点模式设定后，需再测试伺服端的参数，因硬件极限配置在伺服端的 DI9~DI10，所需参数设定 DI9 与 DI10 的引脚功能，此时双击「ASD_A2_E」。

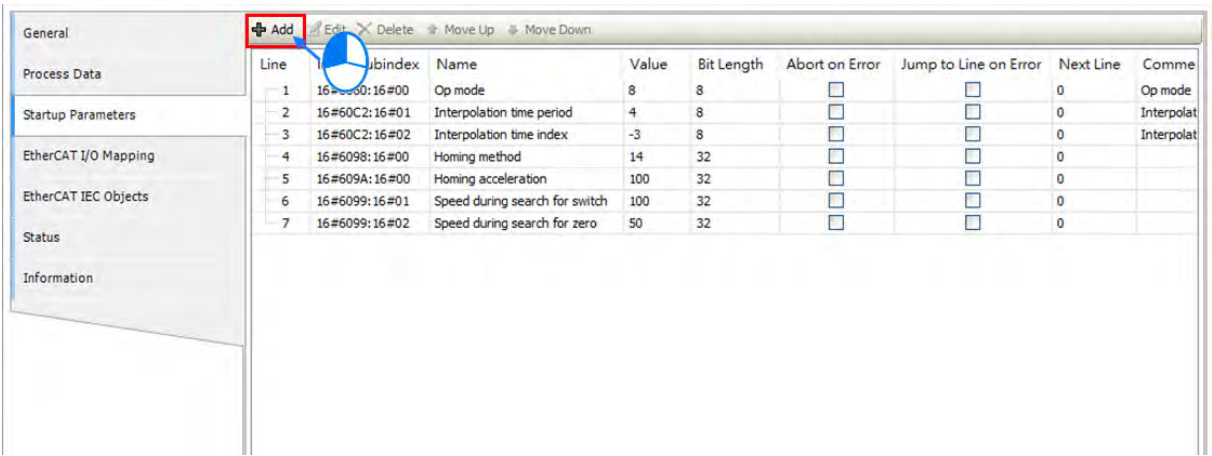


9. 进入「ASD_A2_E」画面再单击「Startup Parameters」。

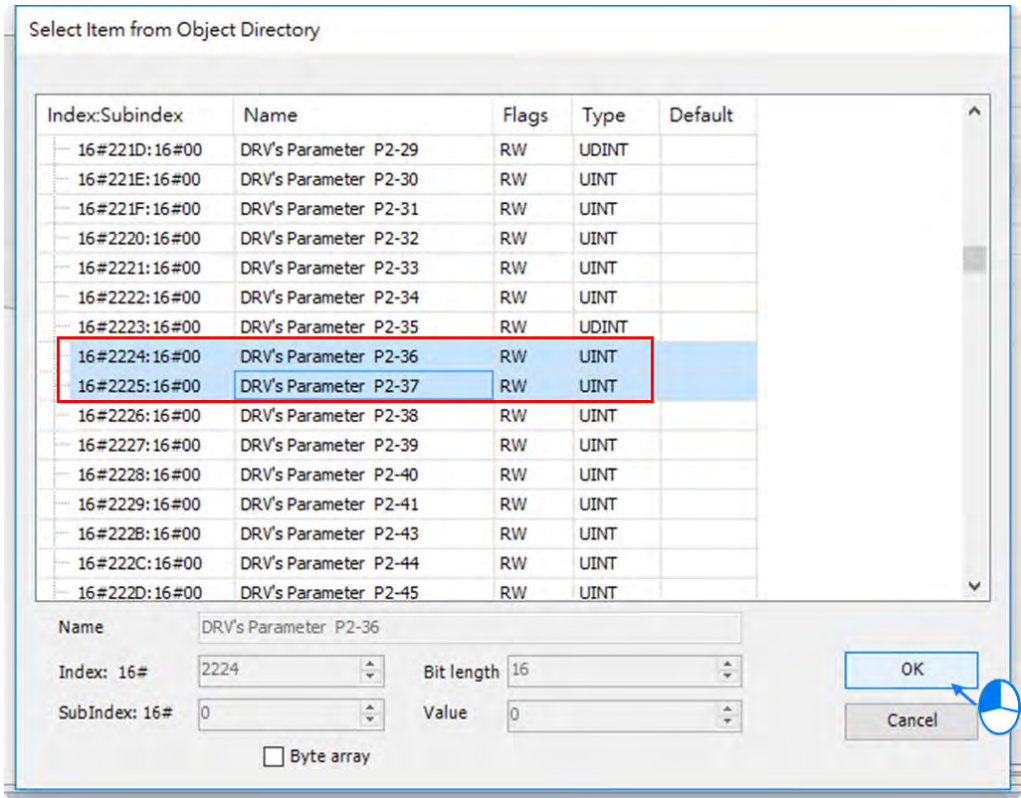


3

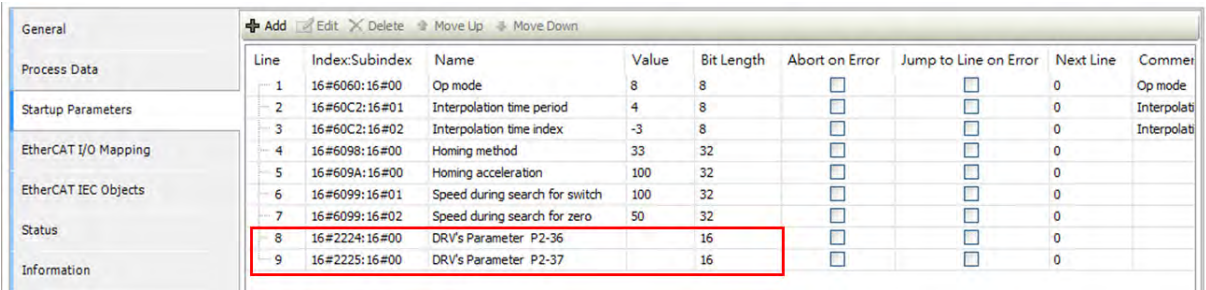
10. 在「Startup Parameters」画面中单击「Add」新增参数。



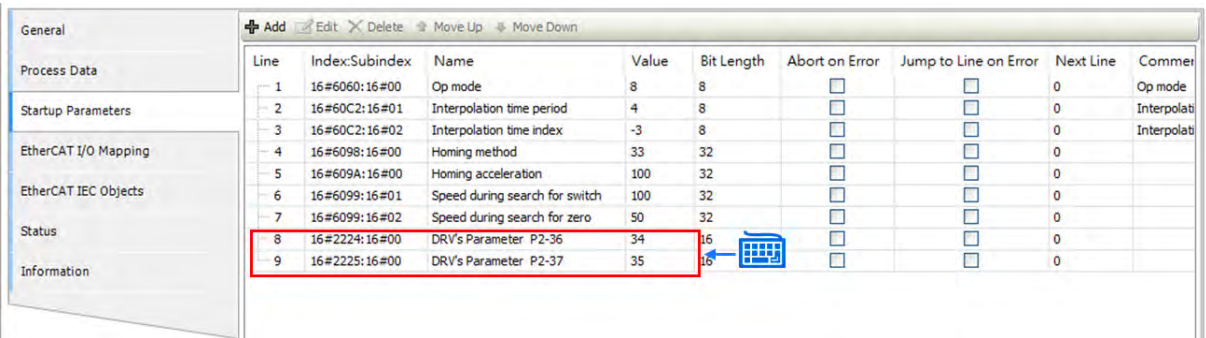
11. 单击 P2-36 后可搭配 Shift 按键，选取多个参数，确认后再单击「OK」即可新增。



12. 新增后参数表中会出现刚刚所选择的参数。



13. 将 P2-36 输入为 34 及 P2-37 设定 35。在此输入为 10 进位。(将 DI9 设定为负极线 (B 接点)); DI10 设定为正极限 (Normally Closed 接点))(在此画面的参数会在 EtherCAT 通讯建立完毕后下载所有设定值至伺服端)



14. 利用相同操作设定下列参数，将伺服端 DI4~DI7 功能关闭，并设定伺服端的齿轮比为 1：1。

General

Process Data

Startup Parameters

EtherCAT I/O Mapping

EtherCAT IEC Objects

Status

Information

+

Add

✎

Edit

✕

Delete

↕

Move Up

↕

Move Down

Line	Index:Subindex	Name	Value	Bit Length	Abort on Error	Jump to Line on Error	Next Line	Comn
1	16#6060:16#00	Op mode	8	8	☐	☐	0	Op moc
2	16#60C2:16#01	Interpolation time period	4	8	☐	☐	0	Interpc
3	16#60C2:16#02	Interpolation time index	-3	8	☐	☐	0	Interpc
4	16#6098:16#00	Homing method	33	32	☐	☐	0	
5	16#609A:16#00	Homing acceleration	100	32	☐	☐	0	
6	16#6099:16#01	Speed during search for switch	100	32	☐	☐	0	
7	16#6099:16#02	Speed during search for zero	50	32	☐	☐	0	
8	16#2224:16#00	DRV's Parameter P2-36	34	16	☐	☐	0	
9	16#2225:16#00	DRV's Parameter P2-37	35	16	☐	☐	0	
10	16#220D:16#00	DRV's Parameter P2-13	256	16	☐	☐	0	
11	16#220E:16#00	DRV's Parameter P2-14	256	16	☐	☐	0	
12	16#220F:16#00	DRV's Parameter P2-15	256	16	☐	☐	0	
13	16#2210:16#00	DRV's Parameter P2-16	256	16	☐	☐	0	
14	16#212C:16#00	DRV's Parameter P1-44	1	32	☐	☐	0	
15	16#212D:16#00	DRV's Parameter P1-45	1	32	☐	☐	0	
16	16#230C:16#00	DRV's Parameter P3-12	256	16	☐	☐	0	

15. 轴 1 与轴 2 设定相同，利用相同操作设定第二轴。

■ 伺服端参数

下列参数为此范例使用，请输入至 Startup Parameters 中。

参数	功能	设定值
P1-44	电子齿轮比分子	1
P1-45	电子齿轮比分母	1
P2-13	数字输入引脚 DI4 功能规划	256
P2-14	数字输入引脚 DI5 功能规划	256
P2-15	数字输入引脚 DI6 功能规划	256
P2-16	数字输入引脚 DI7 功能规划	256
P2-36	数字输入引脚 DI9 功能规划	34
P2-37	数字输入引脚 DI10 功能规划	35
P3-12	参数会维持驱动器断电前的参数	256

注：参数详细介绍可参考 ASDA-A2 系列应用技术手册

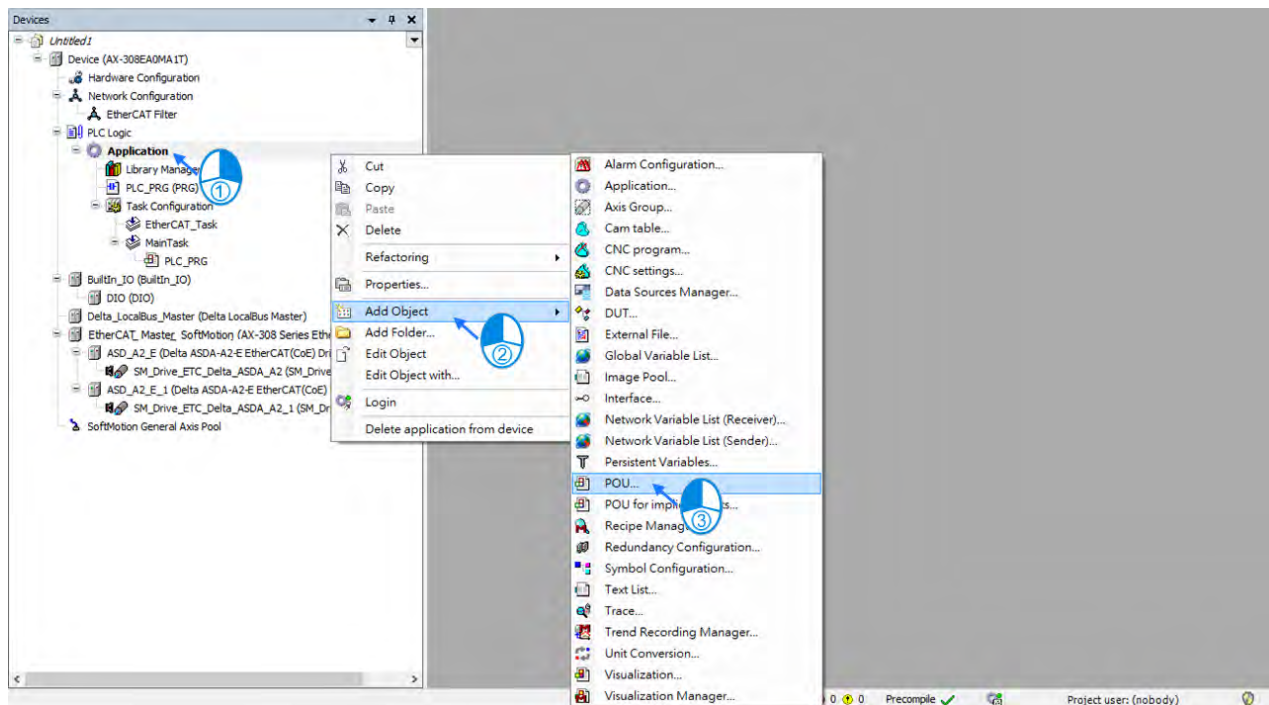
3.2.2.3 项目程序编写

- 程序执行流程

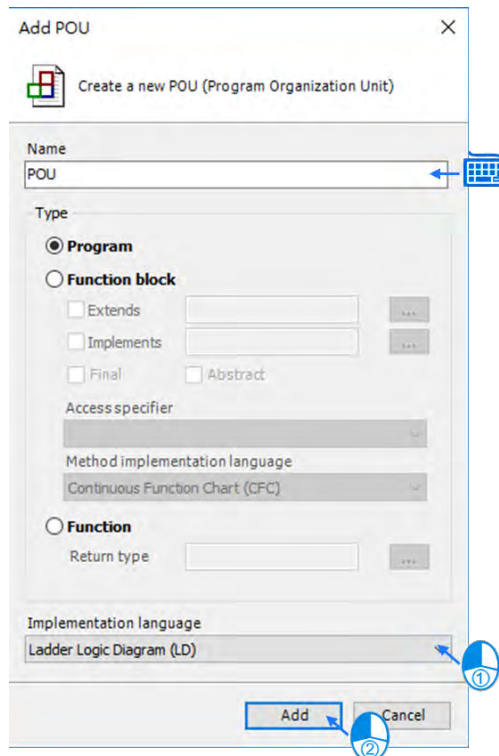
- 执行「Step1」后会触发「MC_Power」，让两轴 Servo ON。
- 执行「Step2」后会触发「MC_Home」，让两轴回原点。
- 执行「Step3」后会触发「MC_MoveRelative」，让两轴正反执行过 1 次 (即第一轴正转，当第一轴正转完毕再触发第二轴正转，当第二轴正转完毕再触发第一轴反转，当第一轴反转完毕再触发第二轴反转，直到反转完毕)，为一次自动执行，再将「Step3」关闭即可。
- 自动化执行一次，「Counter_1」会增加一次。

- 项目建立流程

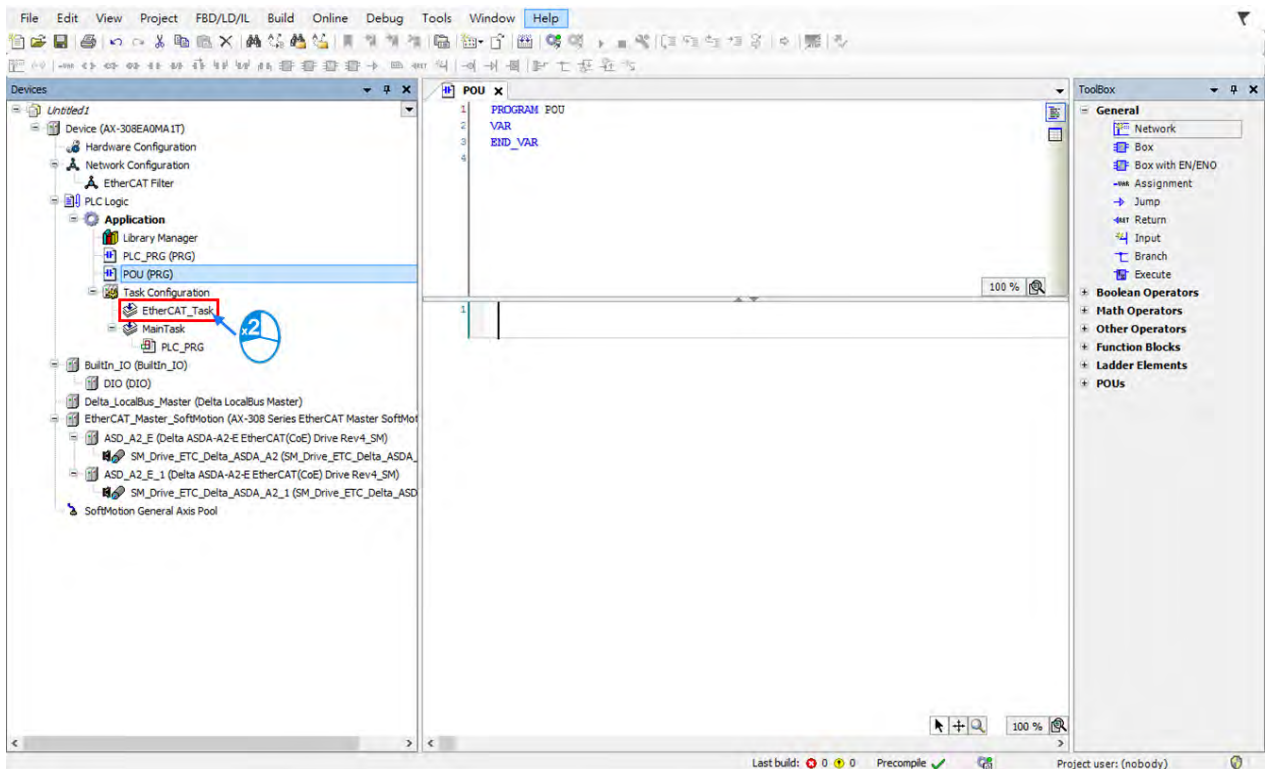
1. 首先建立新的 POU，选择「Application」单击右键，再选择「Add Object」 > 「POU」。



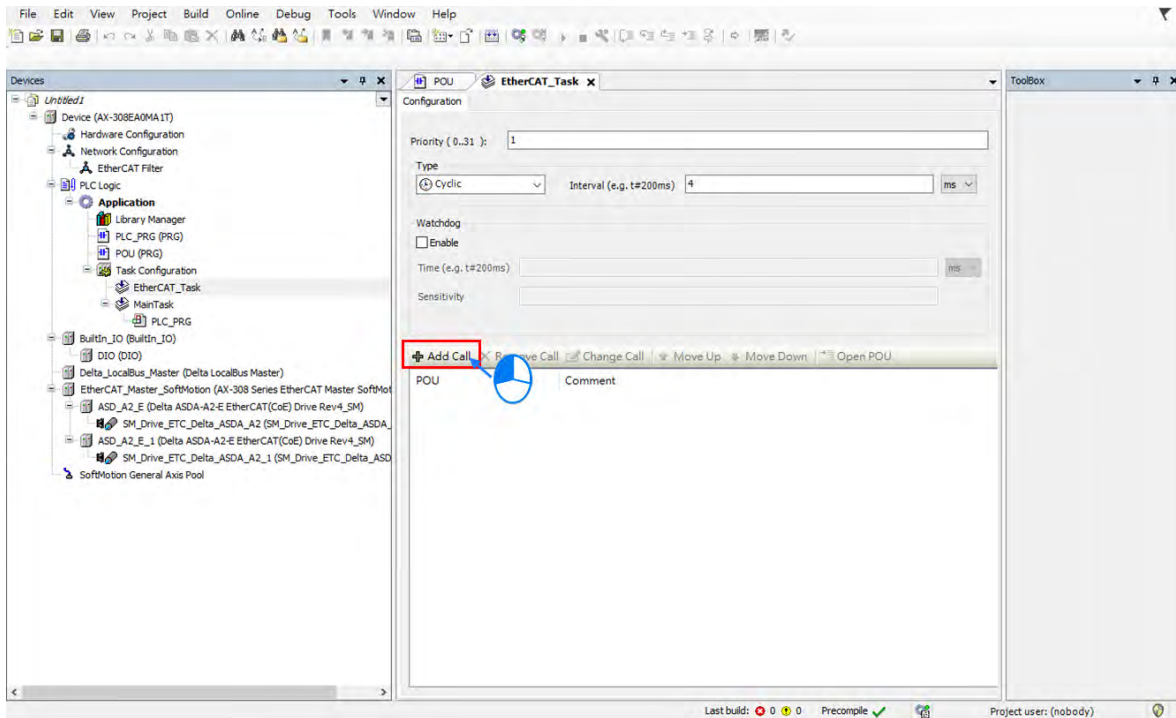
2. 新增画面中，「Name」可输入名称，在「Implementation language」可选择 LD 程序语言，完成后单击「Add」，新增 POU。



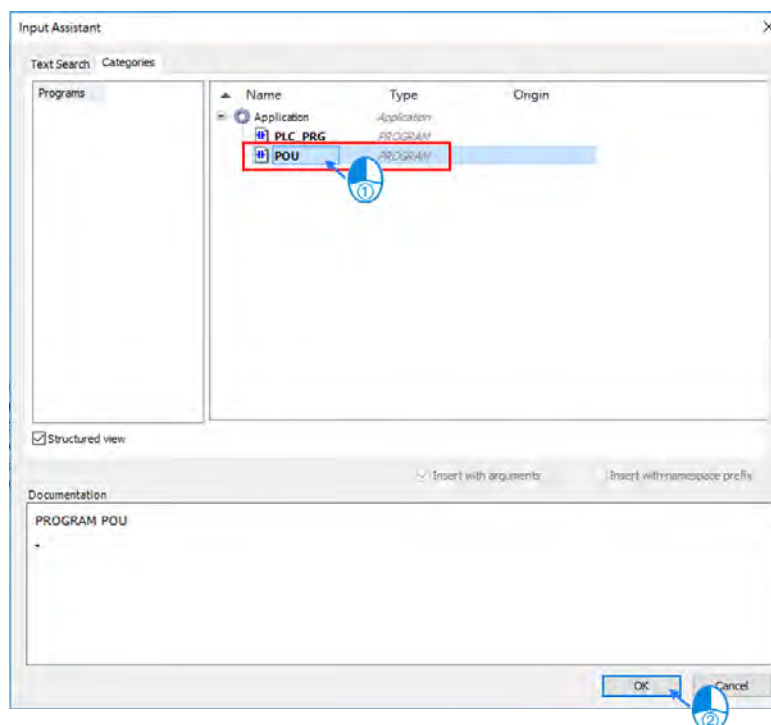
3. 新增的 POU 会出现在树状中，此时需将 POU 新增在 Task 中，单击「EtherCAT_Task」。(有使用轴运动相关 FB 都需建立在 EtherCAT_Task 中，以确保轴可正常运作)



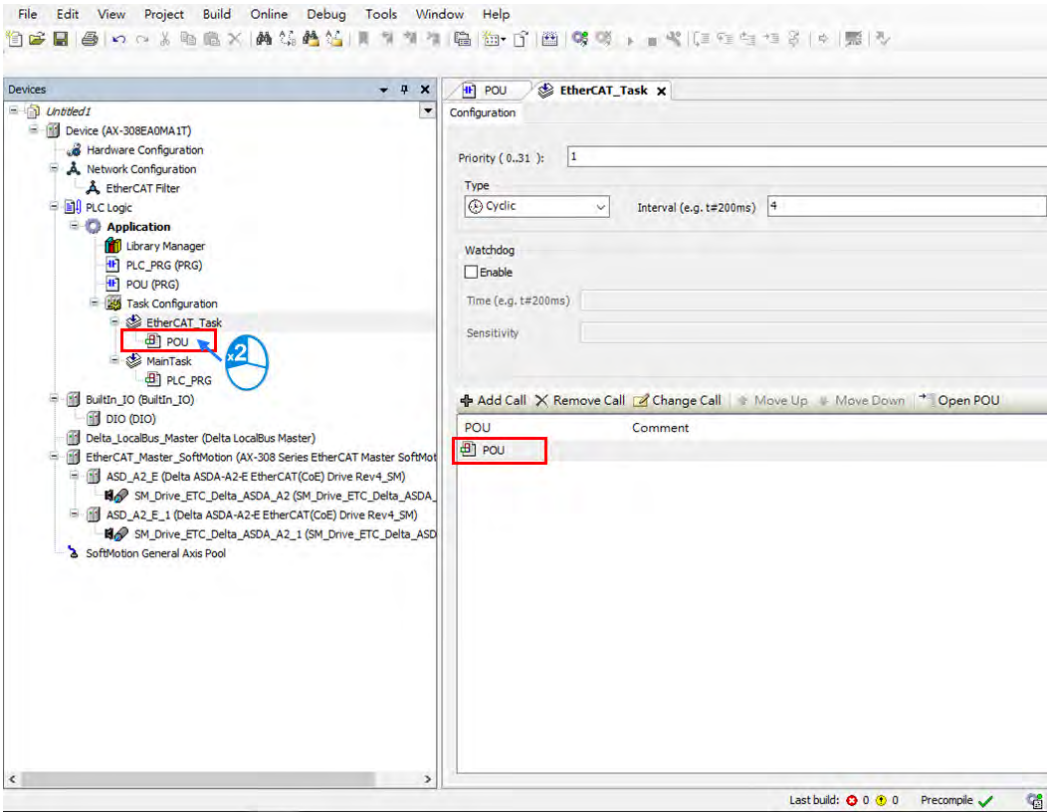
4. 在「EtherCAT_Task」中，单击「Add Call」。



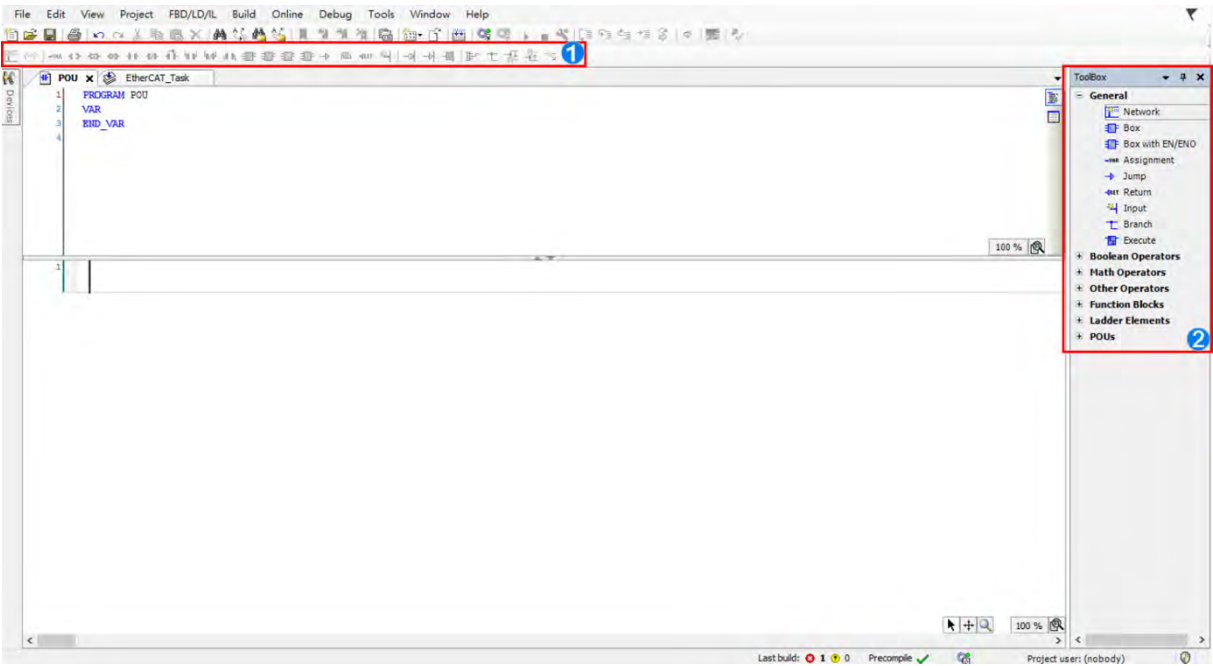
5. 选择刚所建立的「POU」后，单击「OK」。



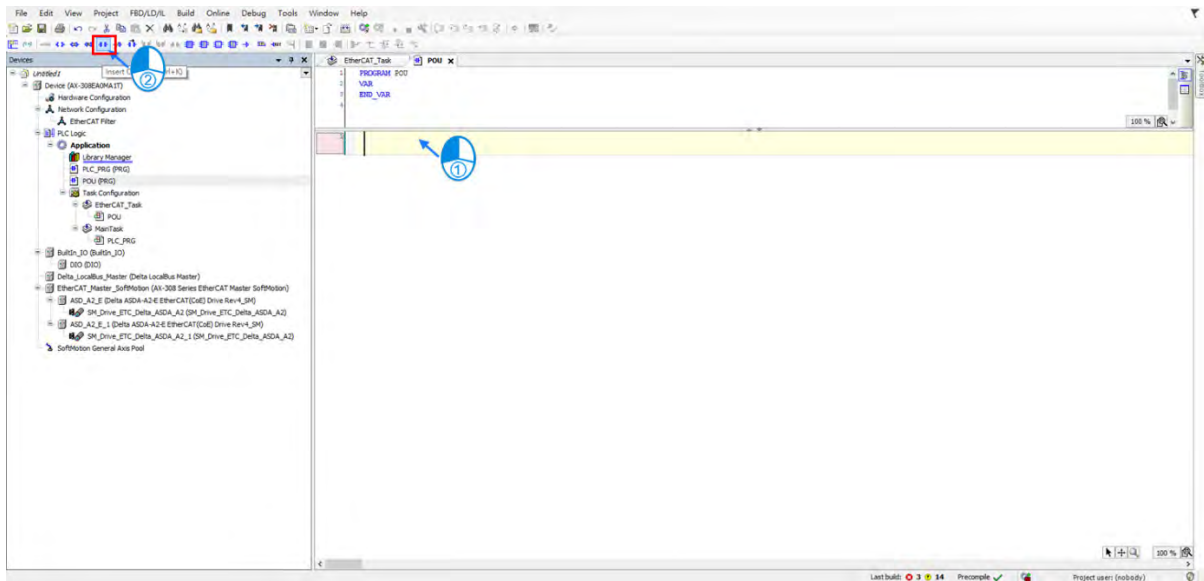
6. 在「EtherCAT_Task」新增完毕后，下方及树形图会出现「POU」，此时再双击树形图的「POU」。



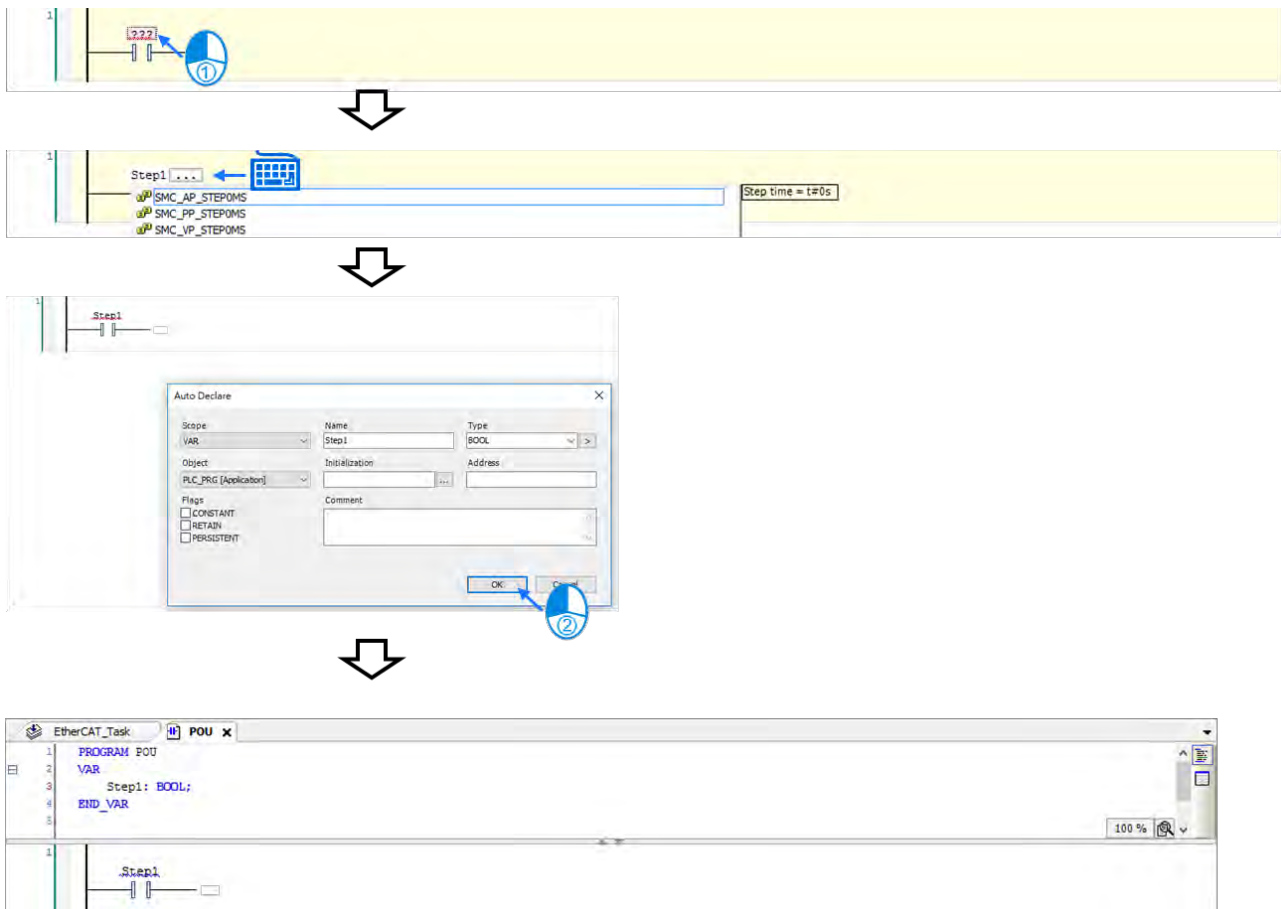
7. 使用 LD 语言编写程序，需新增指令可通过①添加，如需使用功能（Function）或者是功能块（Function Block）需通过②添加。



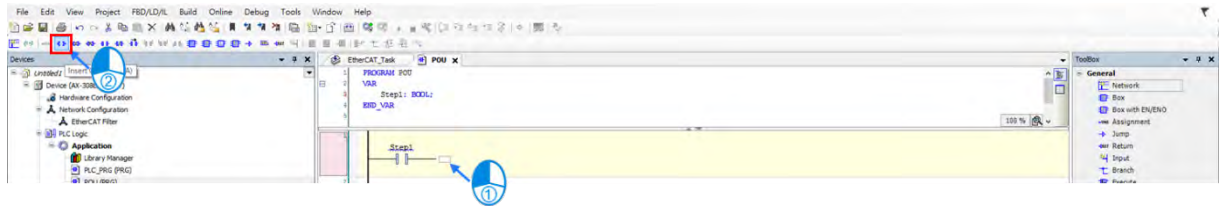
8. 在「POU」中单击欲新增指令的地方后，再单击  即可新增指令。



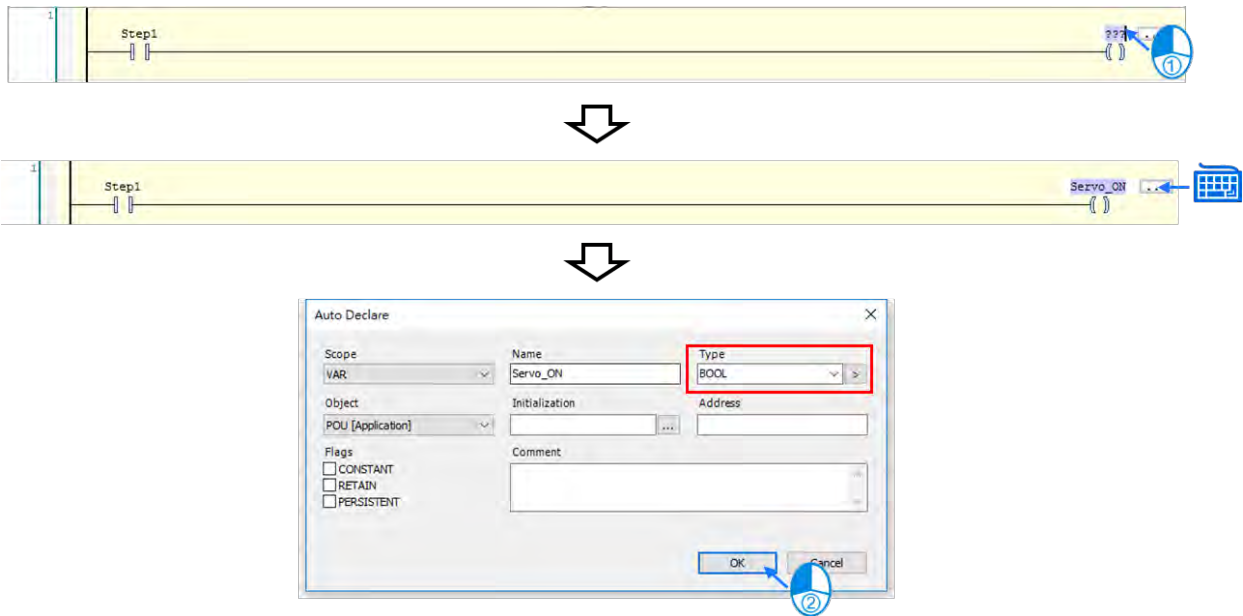
9. 「POU」中出现刚新增的指令后，单击「???」可输入指令变量名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



10. 再来新增输出指令，在 POU 中单击欲新增指令的地方后，再单击即可新增指令。



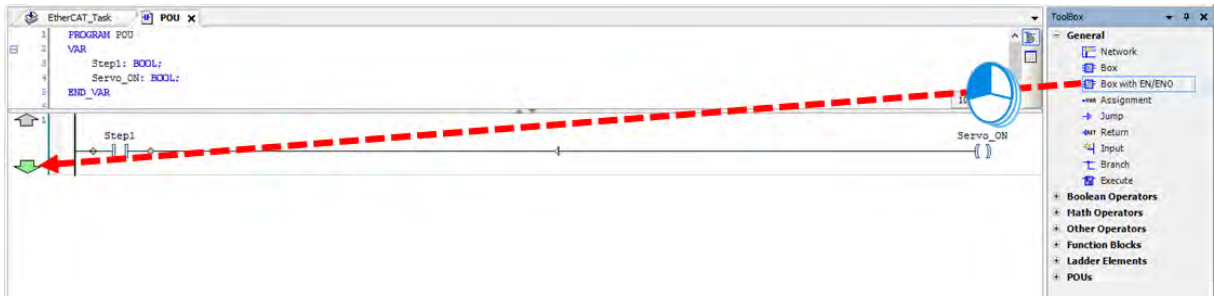
11. 单击「???'后可输入名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



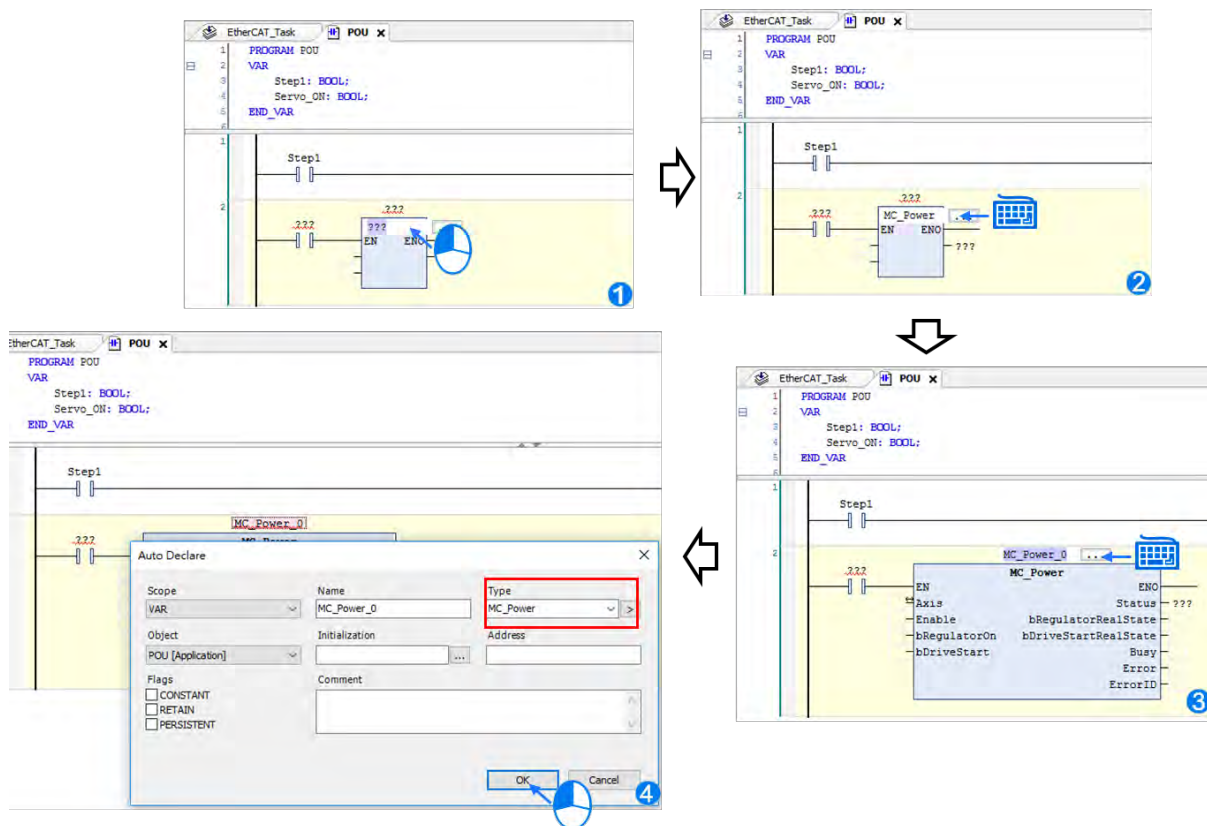
12. 先前新增的指令名称都会在上方便显示名称与类型，也可在上方直接新增指令名称与类型。(后续范例中使用指令方式都是相同操作)



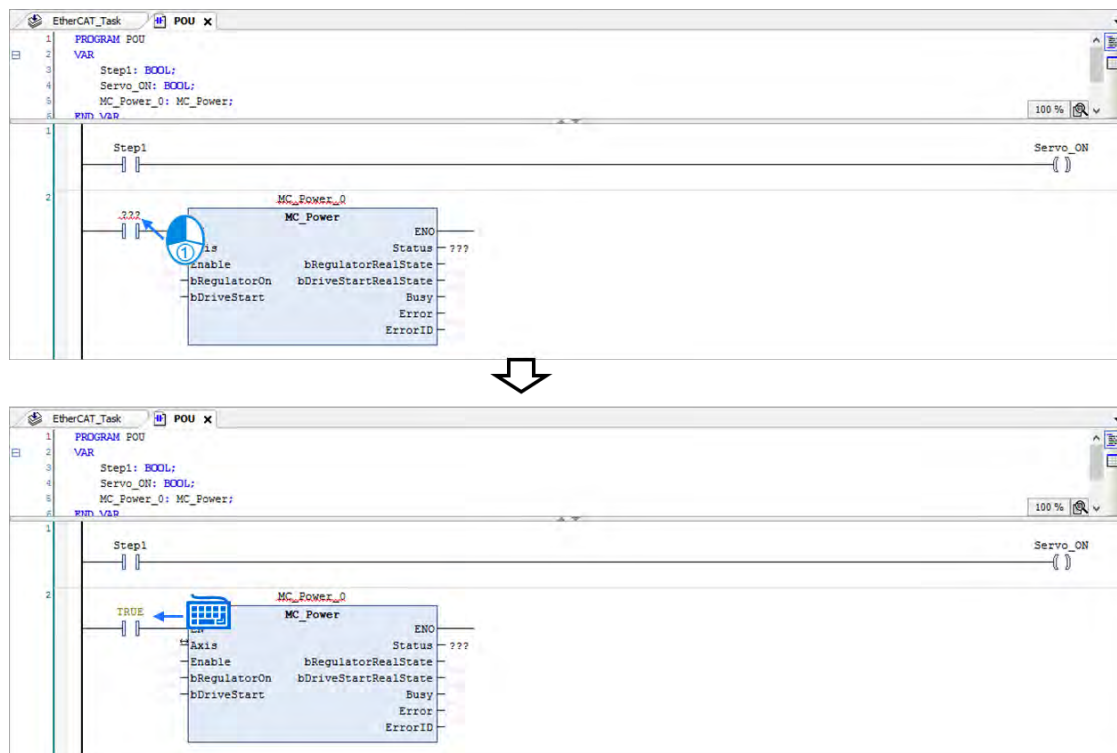
13. 再来新增功能块，单击「Box With EN / ENO」后拉至 POU 中的.



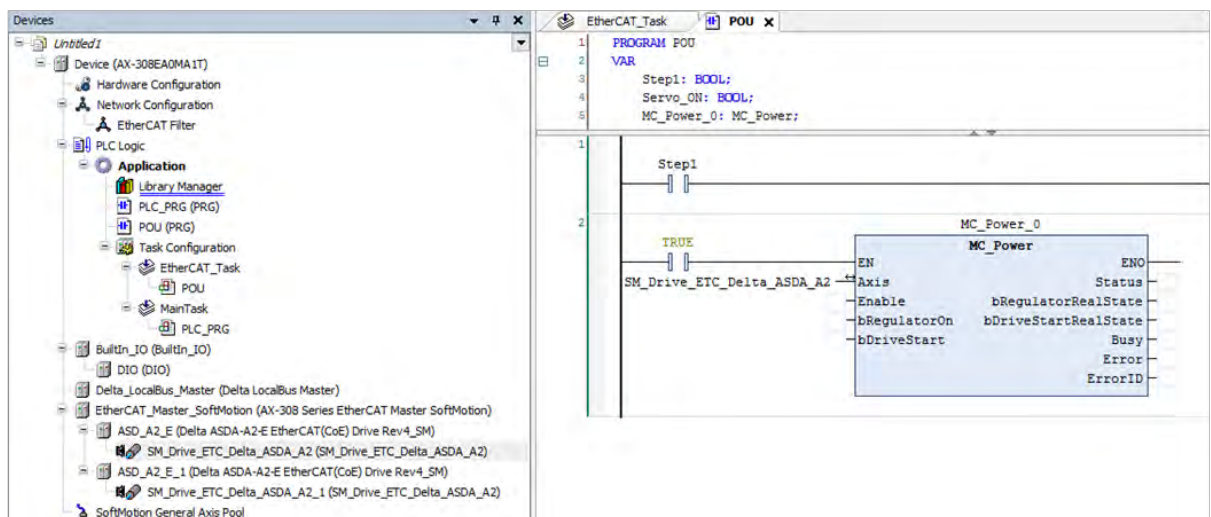
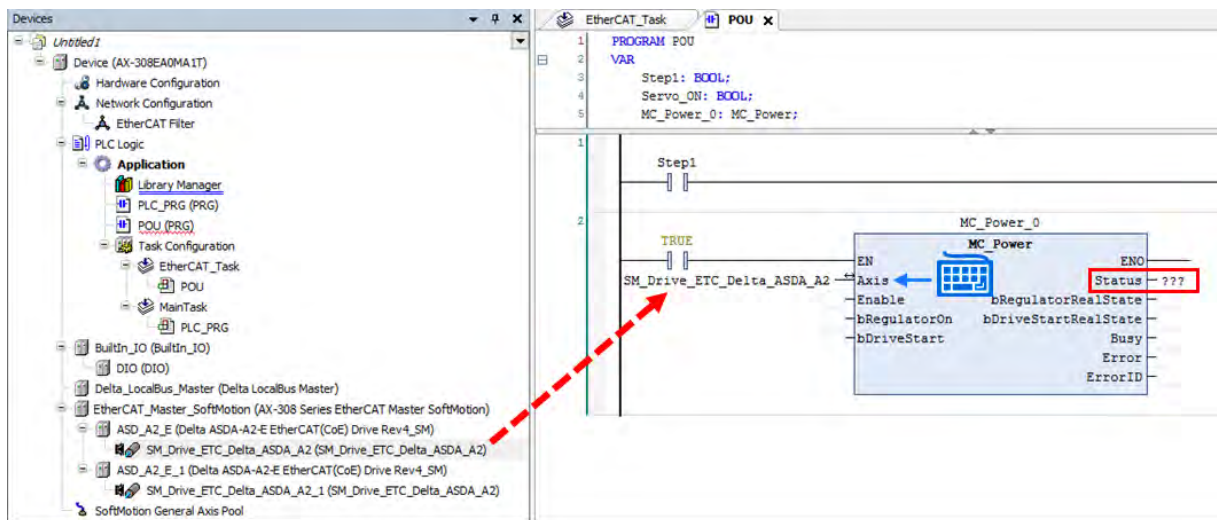
14. 单击「???」后输入功能块名称，输入完成后按下 Enter 键，此时功能块就会出现，再输入功能块名称后按下 Enter 键，就会出现「Auto Declare」，确认功能块「Type」正确后单击「OK」。



15. 在新增功能块，单击「???」输入 True 后单击 Enter 键。

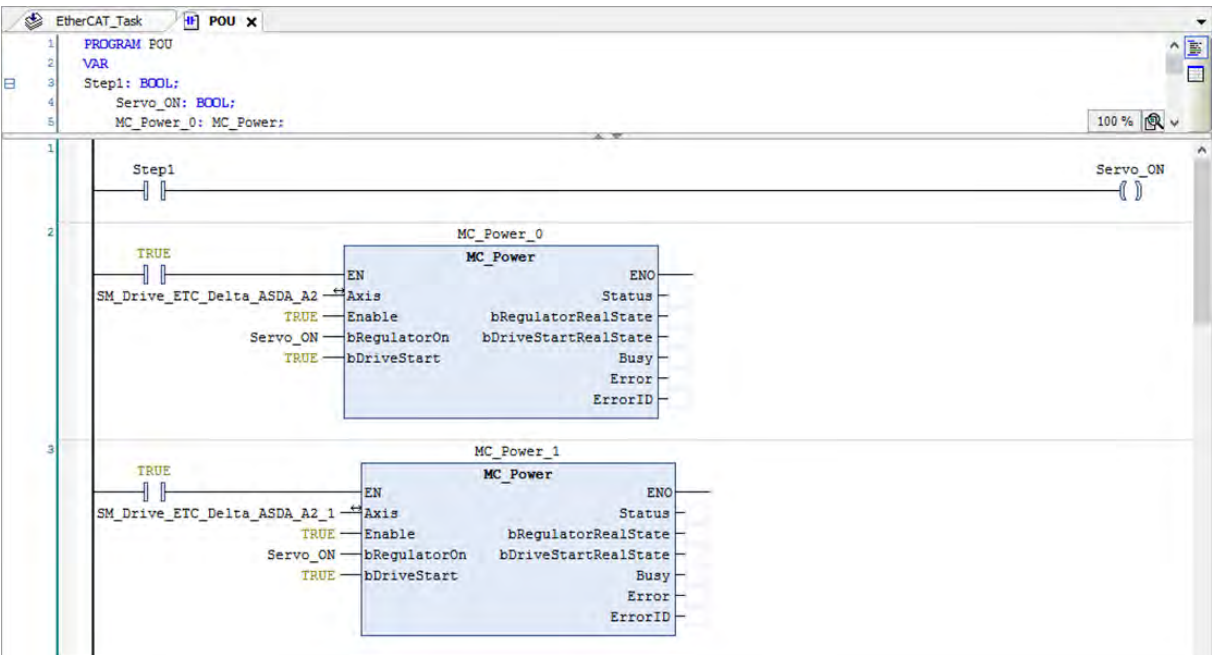


16. 将伺服名称输入至功能块中的 Axis 引脚中并删除引脚中的「???」，因「???」不为任何变数，在编译时会出错。

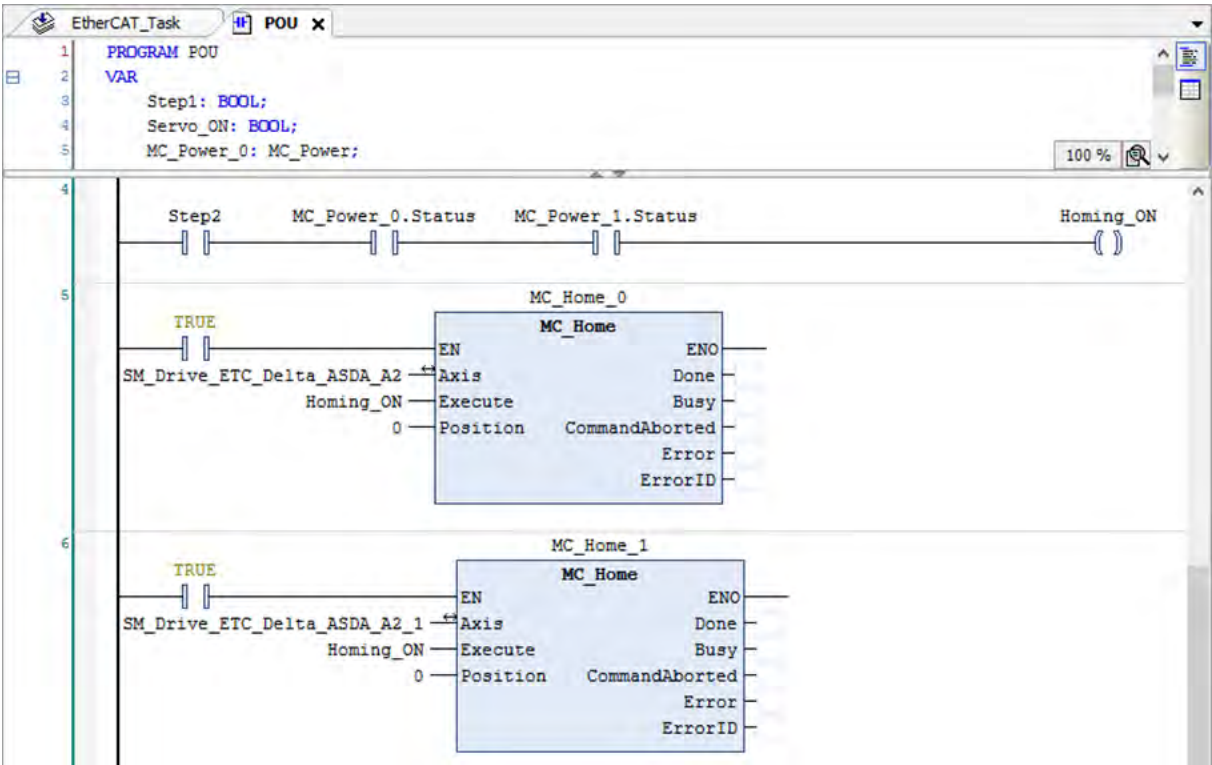


利用同样操作完成以下程序内容

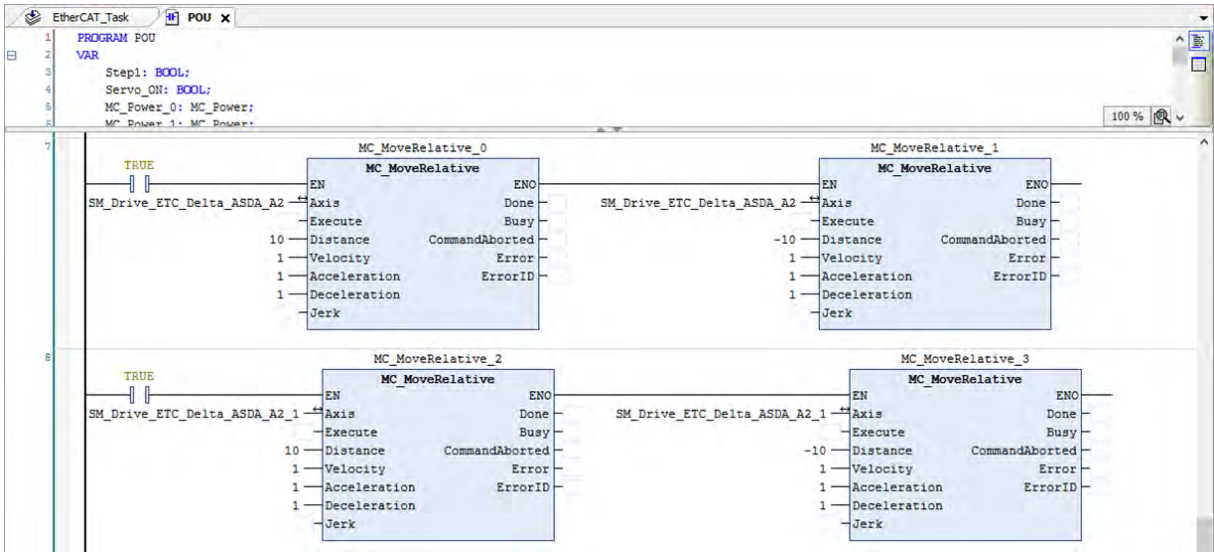
- 建立两轴「MC_Power」功能块



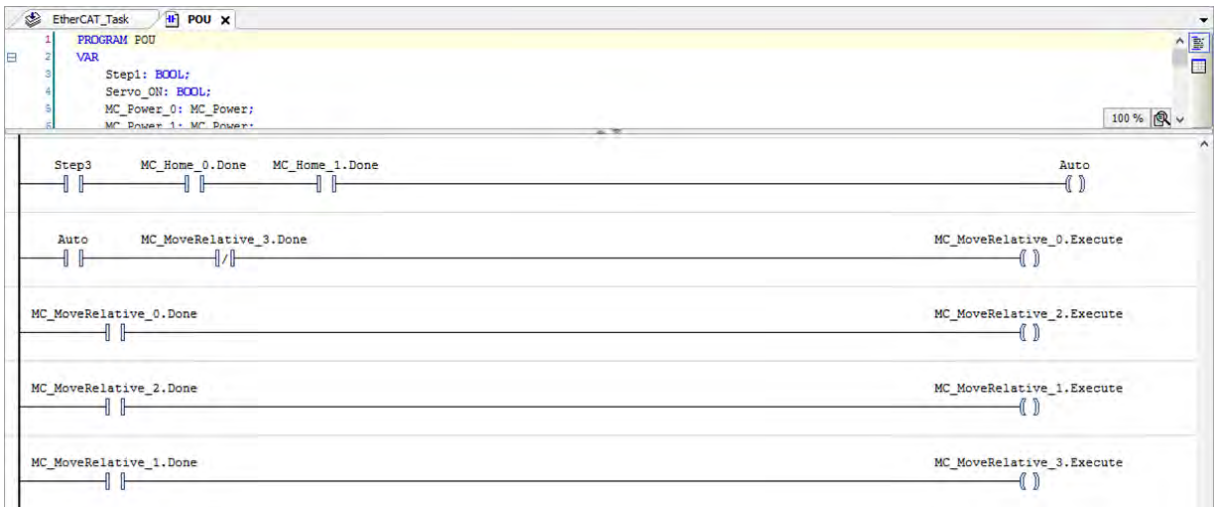
- 再建立两轴「MC_Home」功能块，并增加判断轴 1 与轴 2 是否有 Servo ON 后才可执行「MC_Home」。



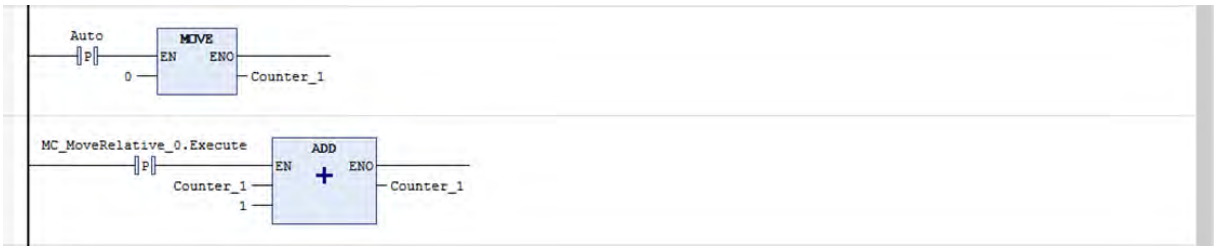
- 再建立「MC_MoveRelative」功能块，让两轴可以左右定位移动。



- 由于先前建立「MC_MoveRelative」功能块，再建立自动化程序让轴 1、2 执行正转后再执行轴 1、2 反转。



- 当执行「Auto」时，会将「Counter_1」数值清除，两轴正反执行过 1 次就会自动加 1。

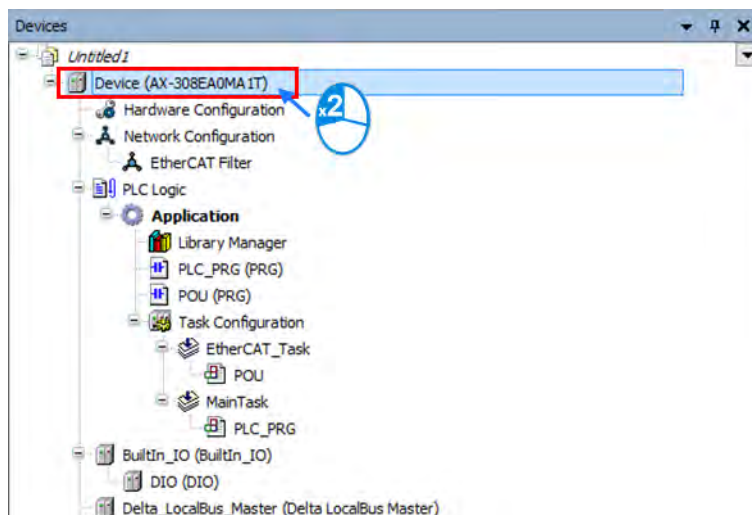


3.2.3 程序监控

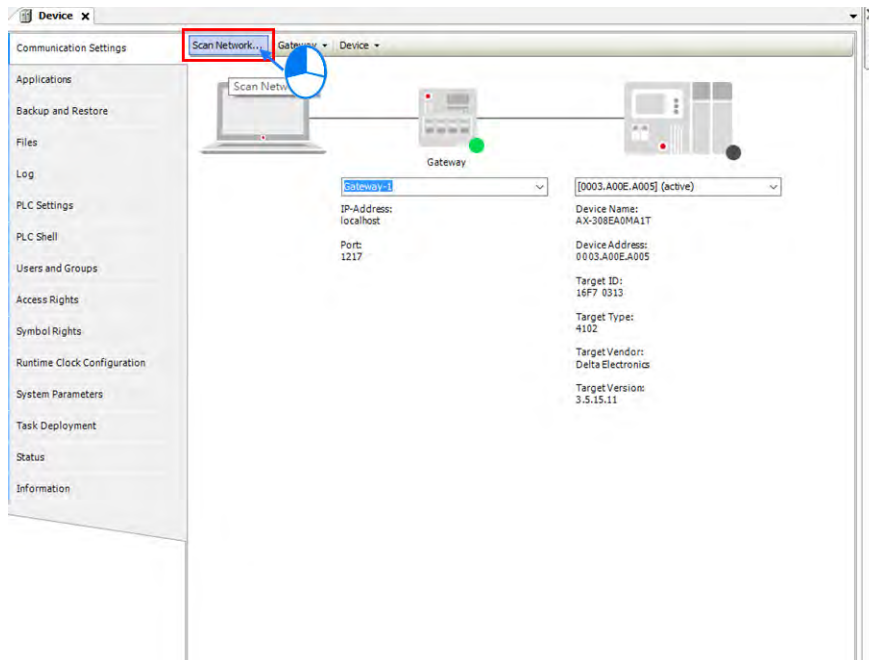
当程序运行的时候，用户可以通过监视的方式，来了解系统目前的逻辑控制状态，或是修改部分装置的数值来进行系统的测试，现在我们就来进行程序监视的操作。

3.2.3.1 设备与计算机链接设定

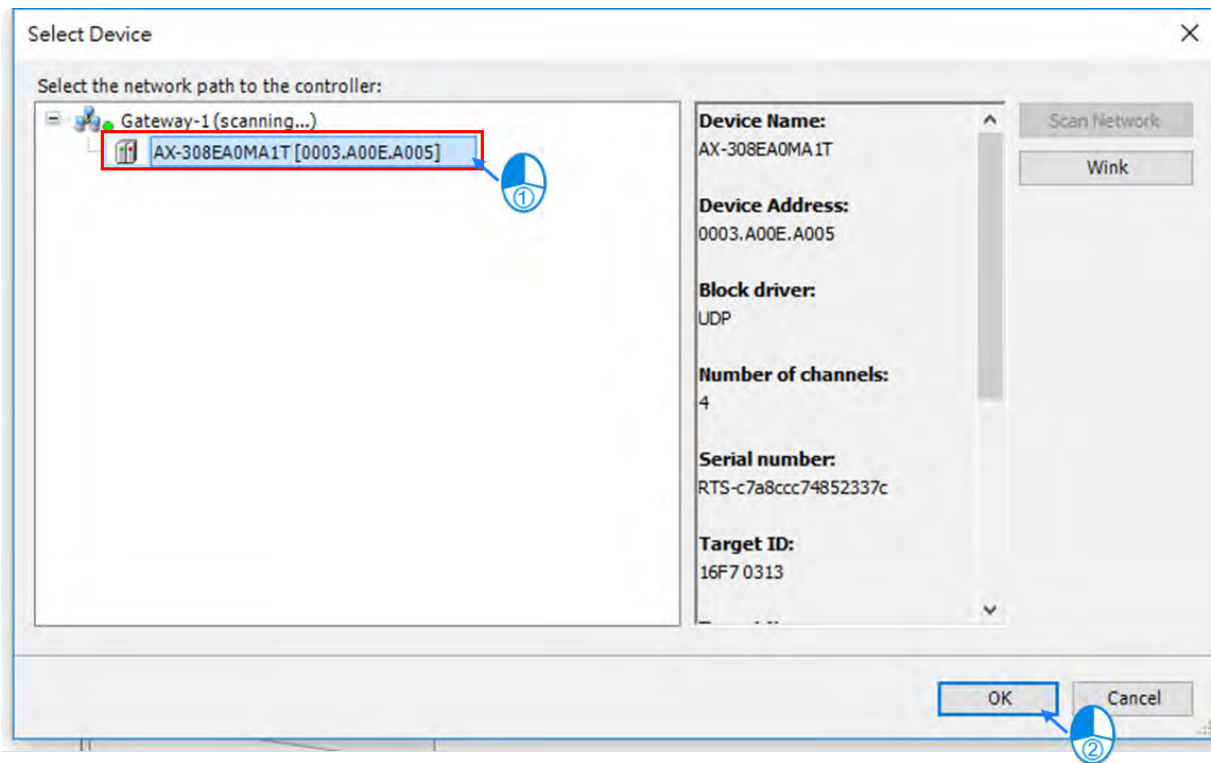
1. 单击「Device」后双击鼠标左键（AX-308E 预设 IP 为 192.168.1.5）



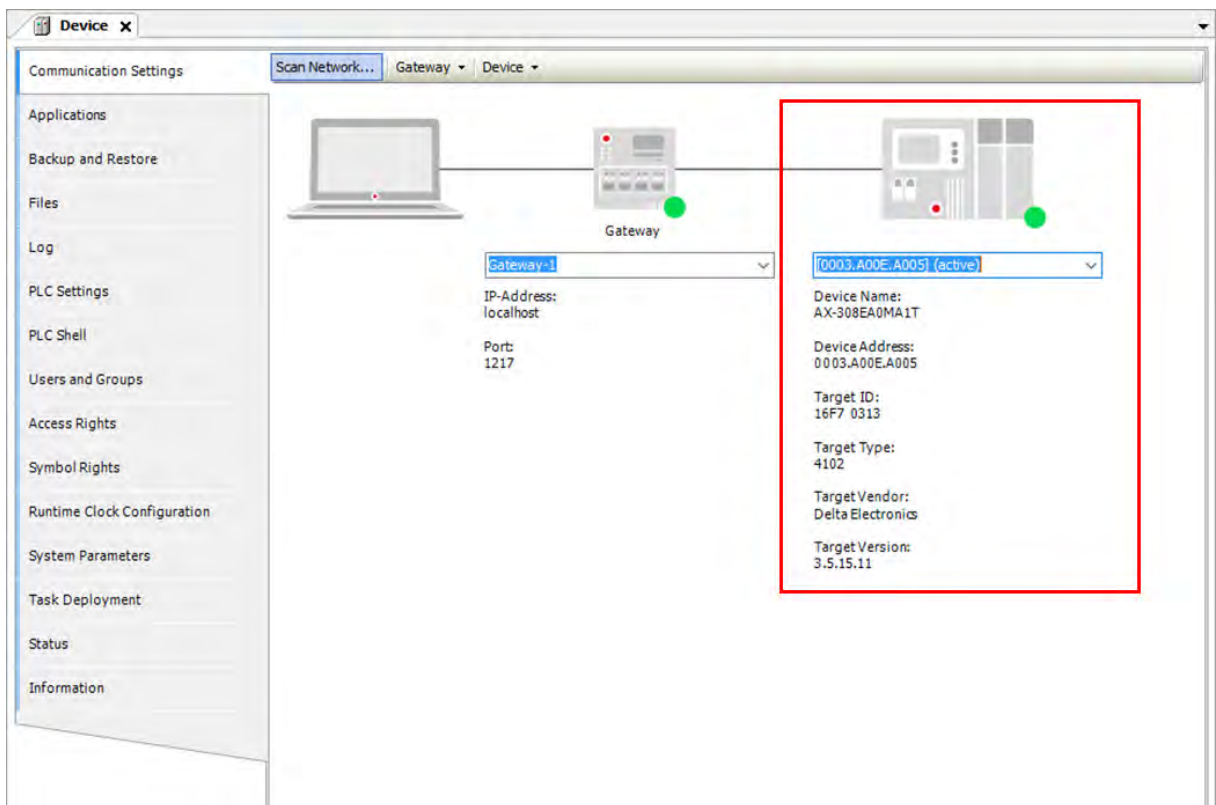
2. 进入「Device」画面后，选择「Scan Network」。



3. 在「Select Device」画面中，单击「AX-308E」>「OK」。

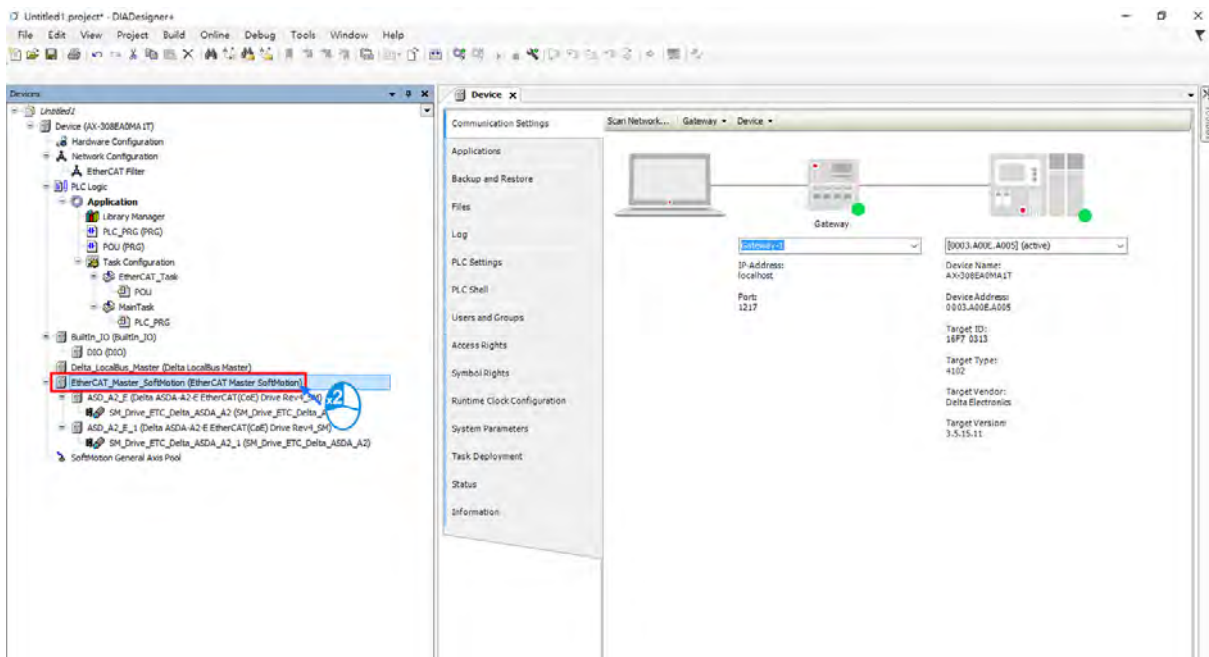


4. 当 PC 与设备连结后，会出现设备相关的信息，如下红框所示。

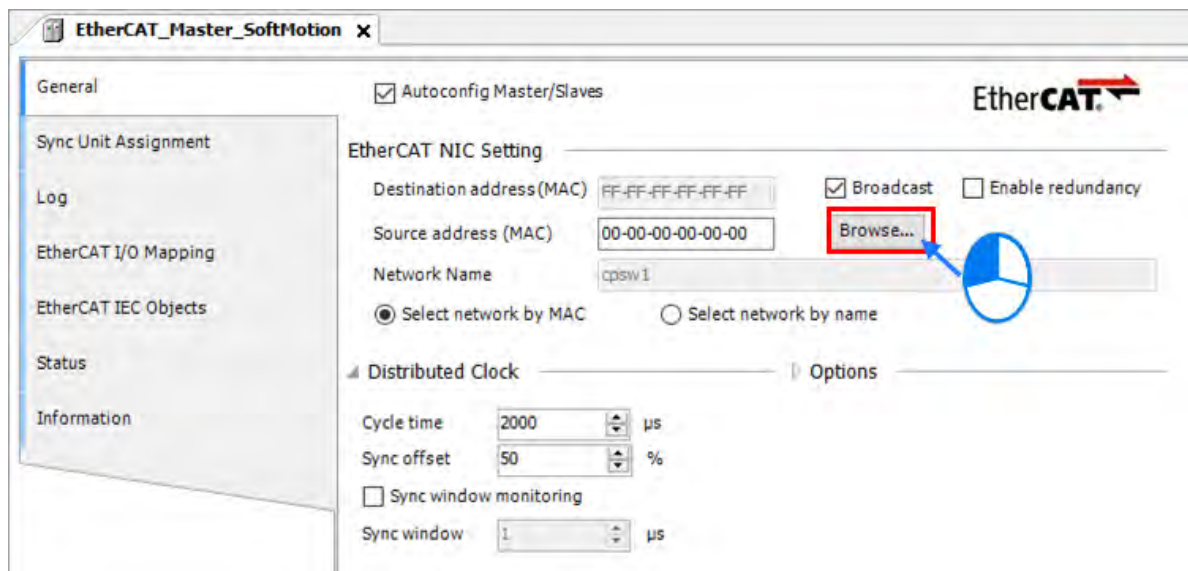


3.2.3.2 程序监控流程

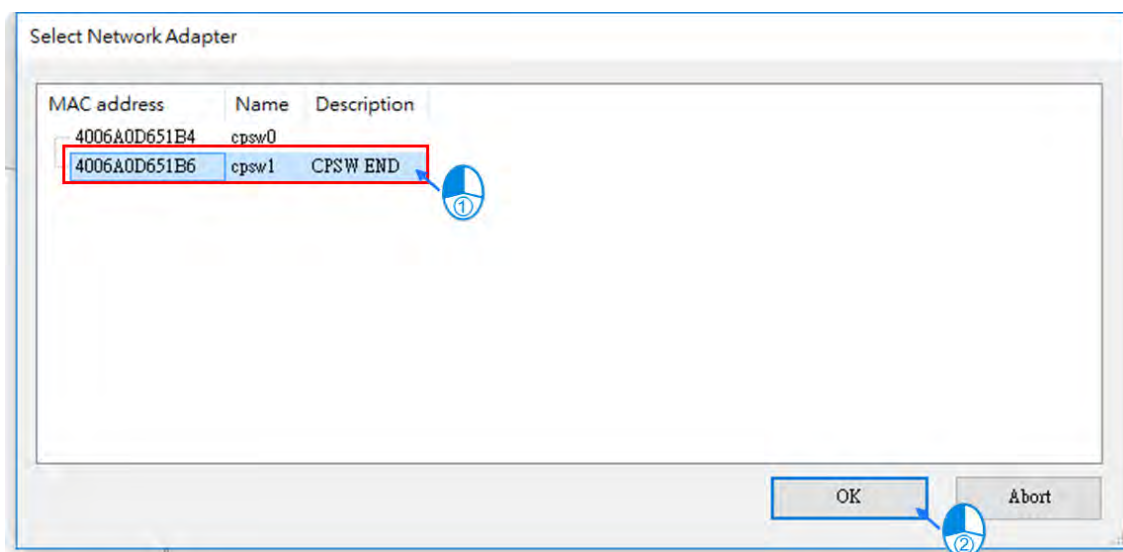
1. 监控程序前需确认 EtherCAT 通道是否建立正确，确认后再双击「EtherCAT_Master_SoftMotion」。



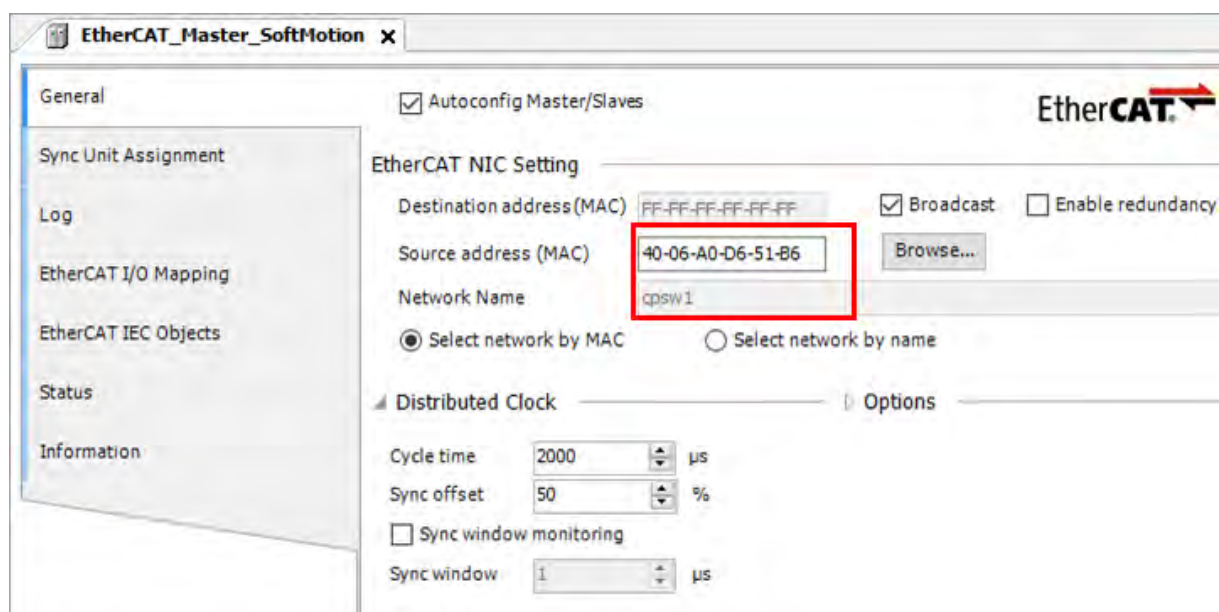
2. 进入「EtherCAT_Master_SoftMotion」画面后，单击「Browse」。



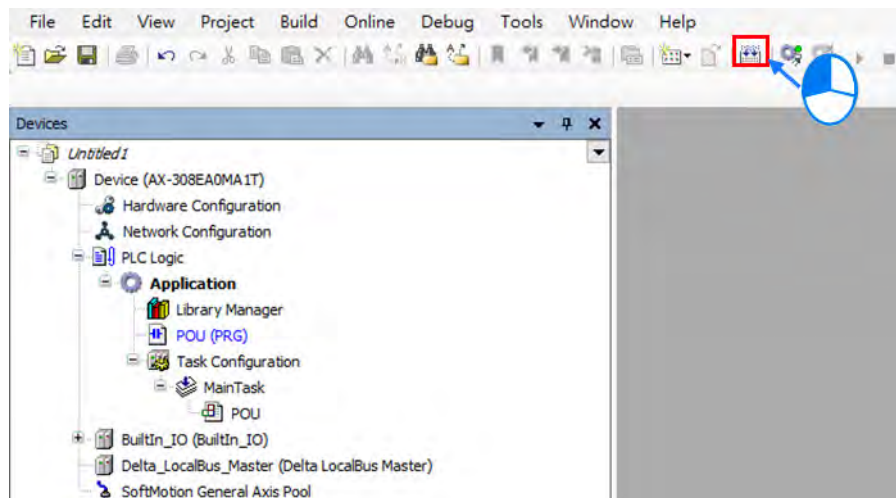
3. 需选择 EtherCAT 网络端口「cpsw1」后，单击「OK」。



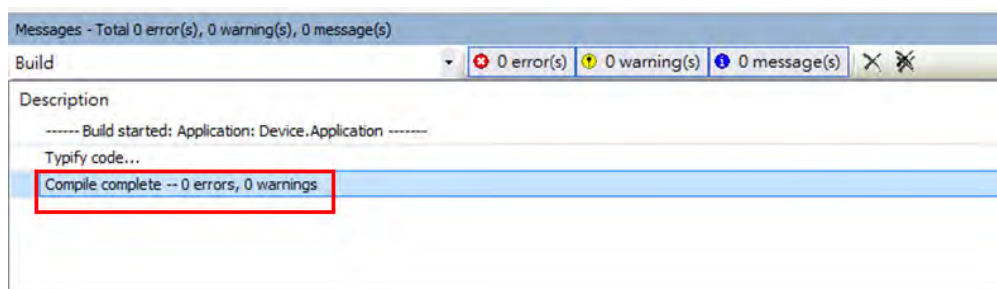
4. 选择完毕后，「MAC」会出现设定值。



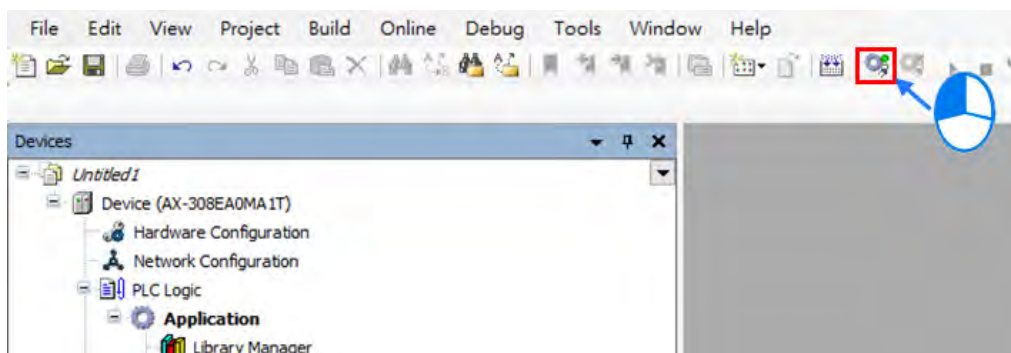
5. 监控前先单击  编译以确认程序是否正确。



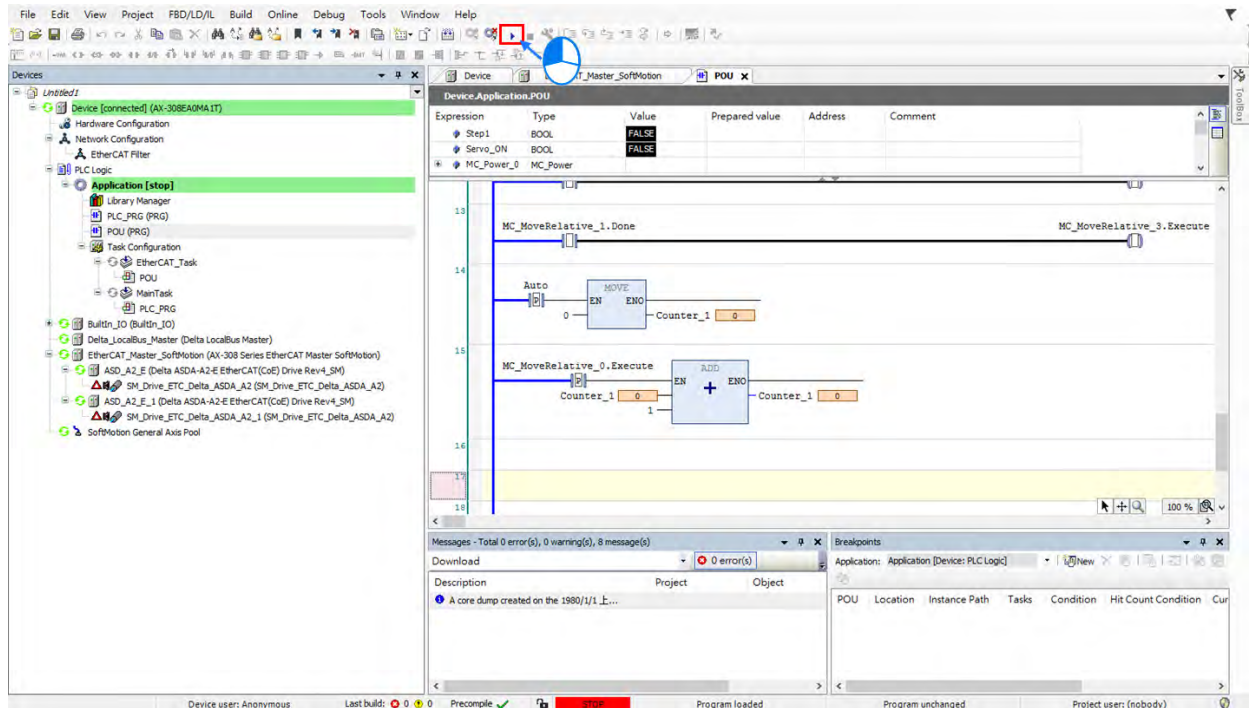
6. 编译完成后，可确认程序编写是否有错误。



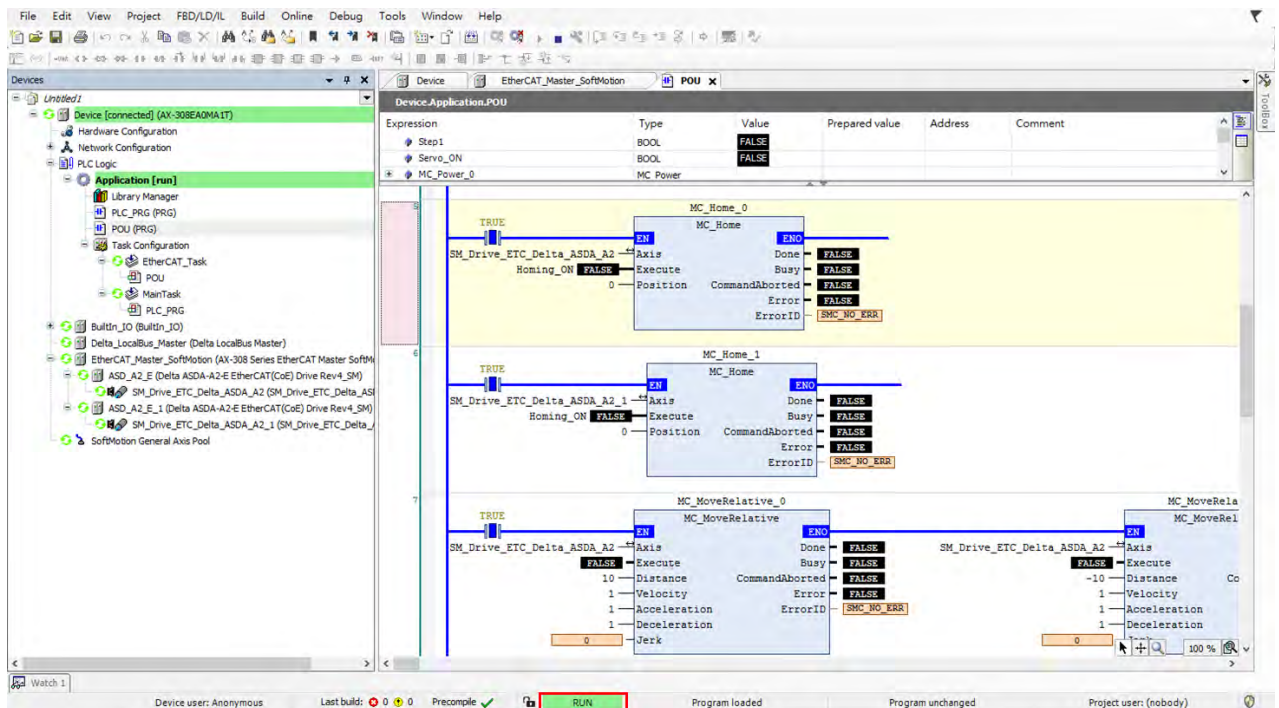
7. 当编译完成后，再单击  执行在线监控。



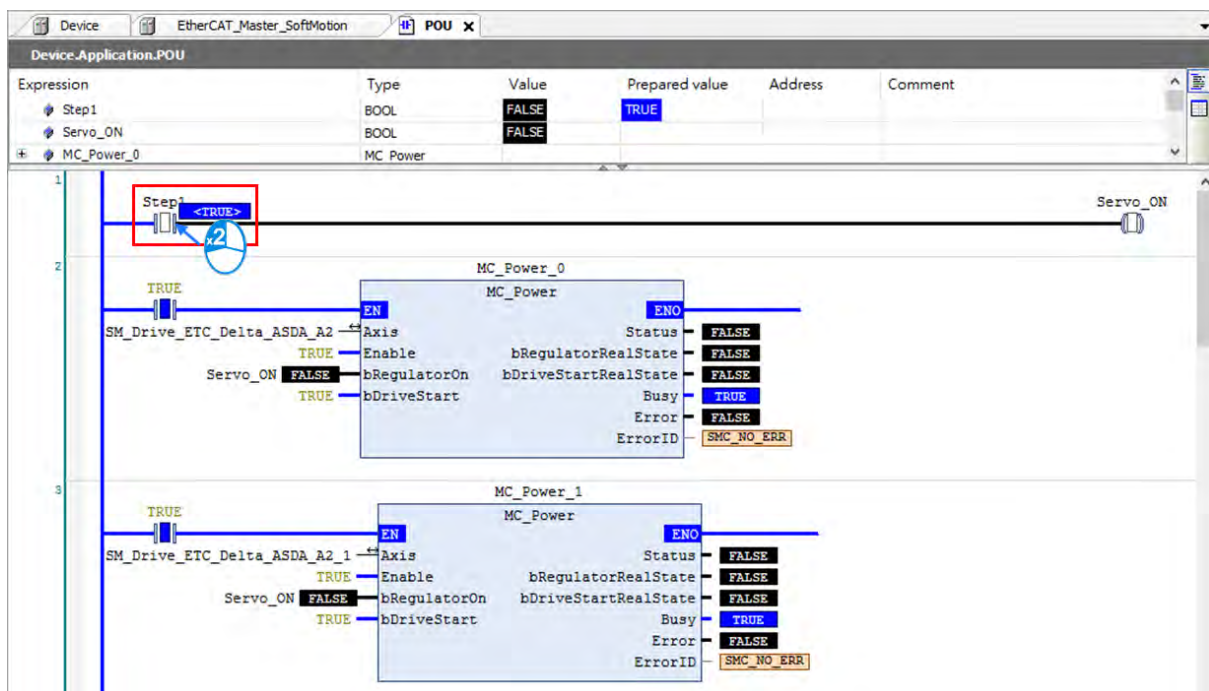
8. 执行在线监控后需下载程序，下载完成后 PLC 状态会 Stop，需单击 ► PLC 才会执行 Run。



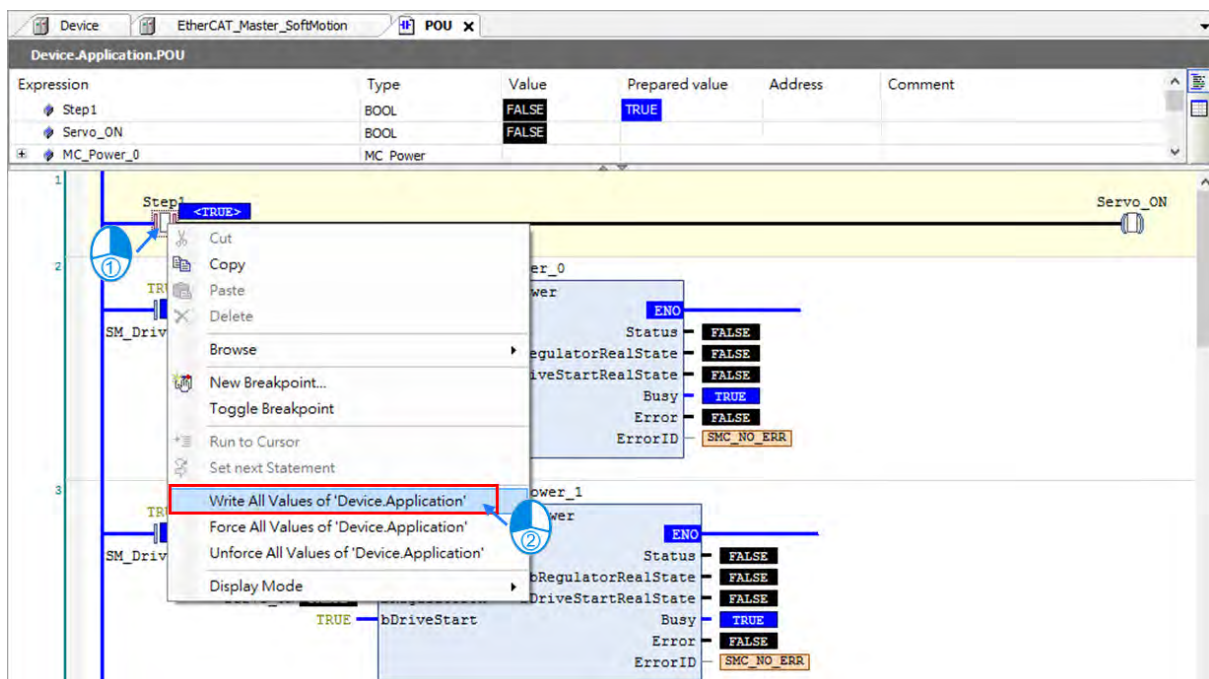
9. 当 PLC 开始 Run 时，下方的状态会改为 **RUN**，此时可开始执行程序。



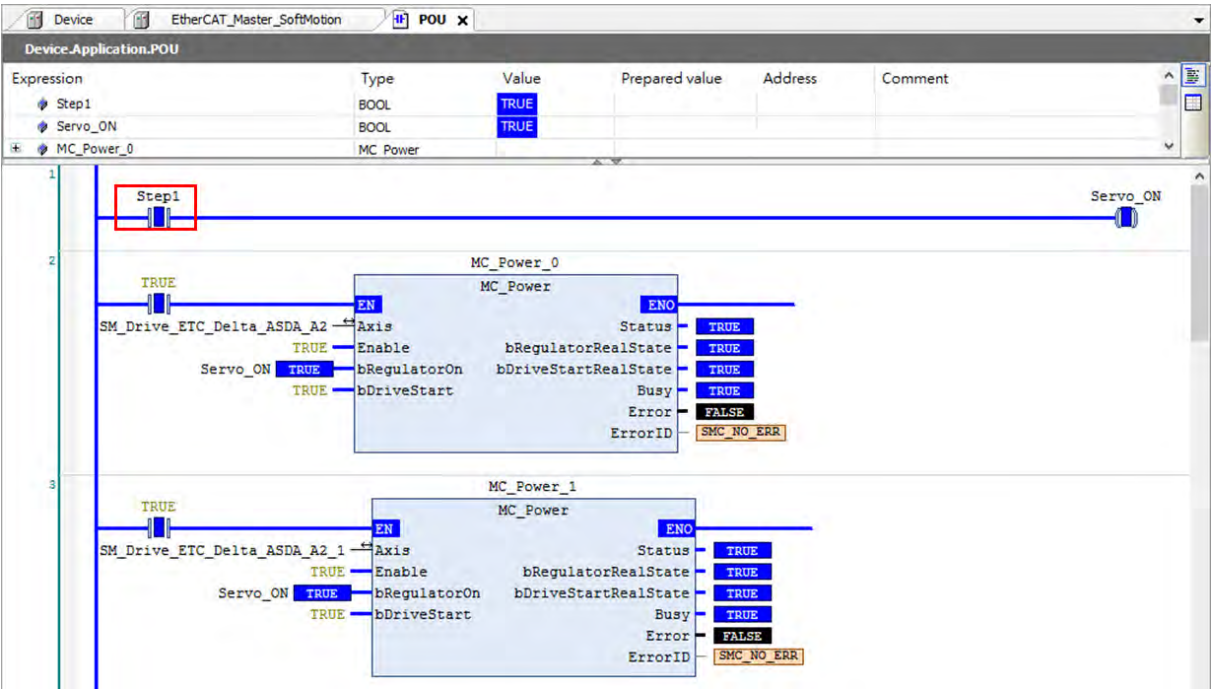
10. 在线监控后，双击「Step1」后会出现 **<TRUE>** 的图示。



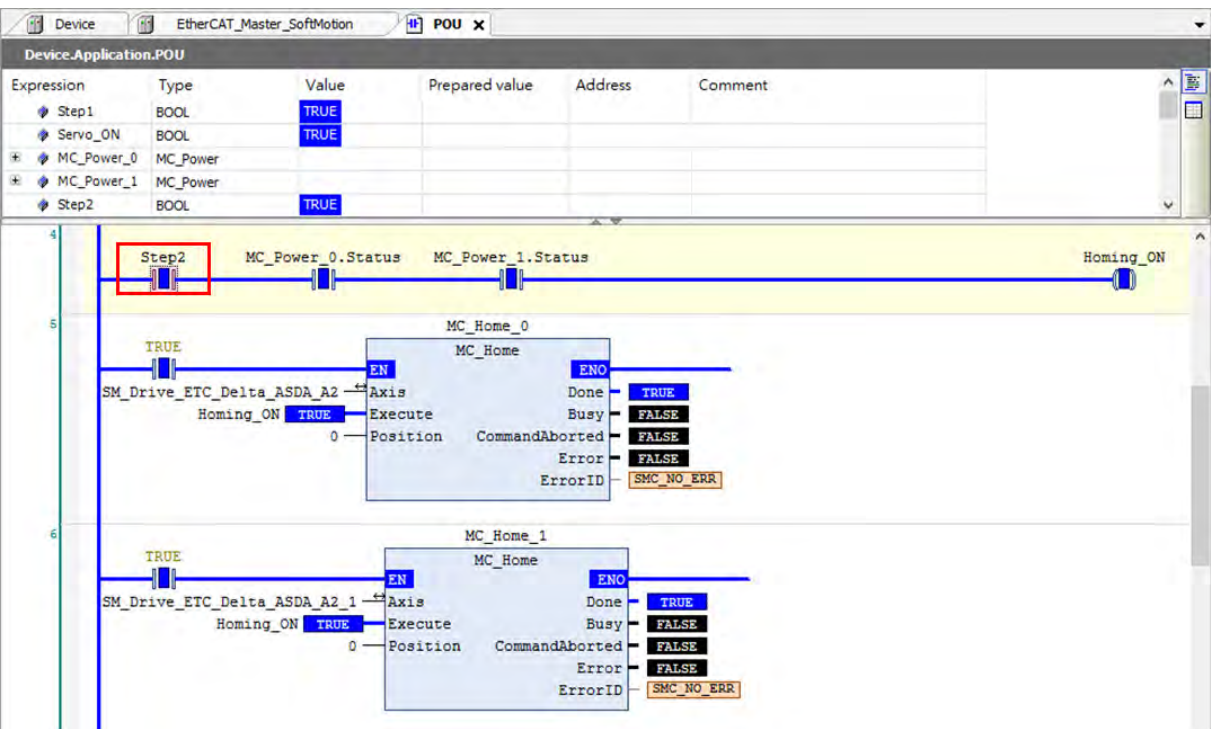
11. 单击「Step1」后右键单击，选择「Write All Values of "Device.Application"」，会写入程序中所有状态。（可使用快捷键 Ctrl + F7）



12. 启动「Step1」后，1、2 轴会 Servo ON。(以相同操作执行以下程序)

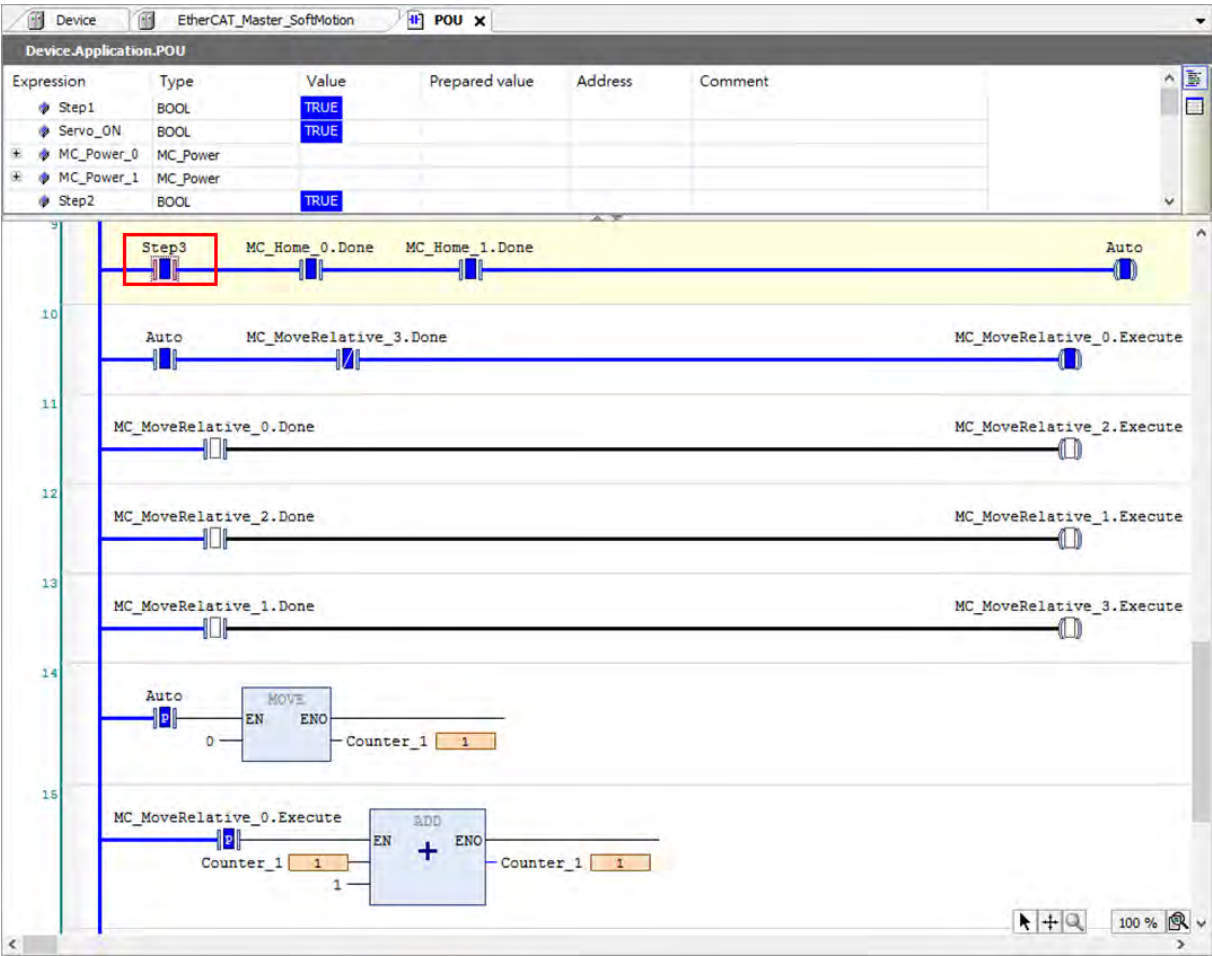


13. 启动「Step2」后使两轴回原点。



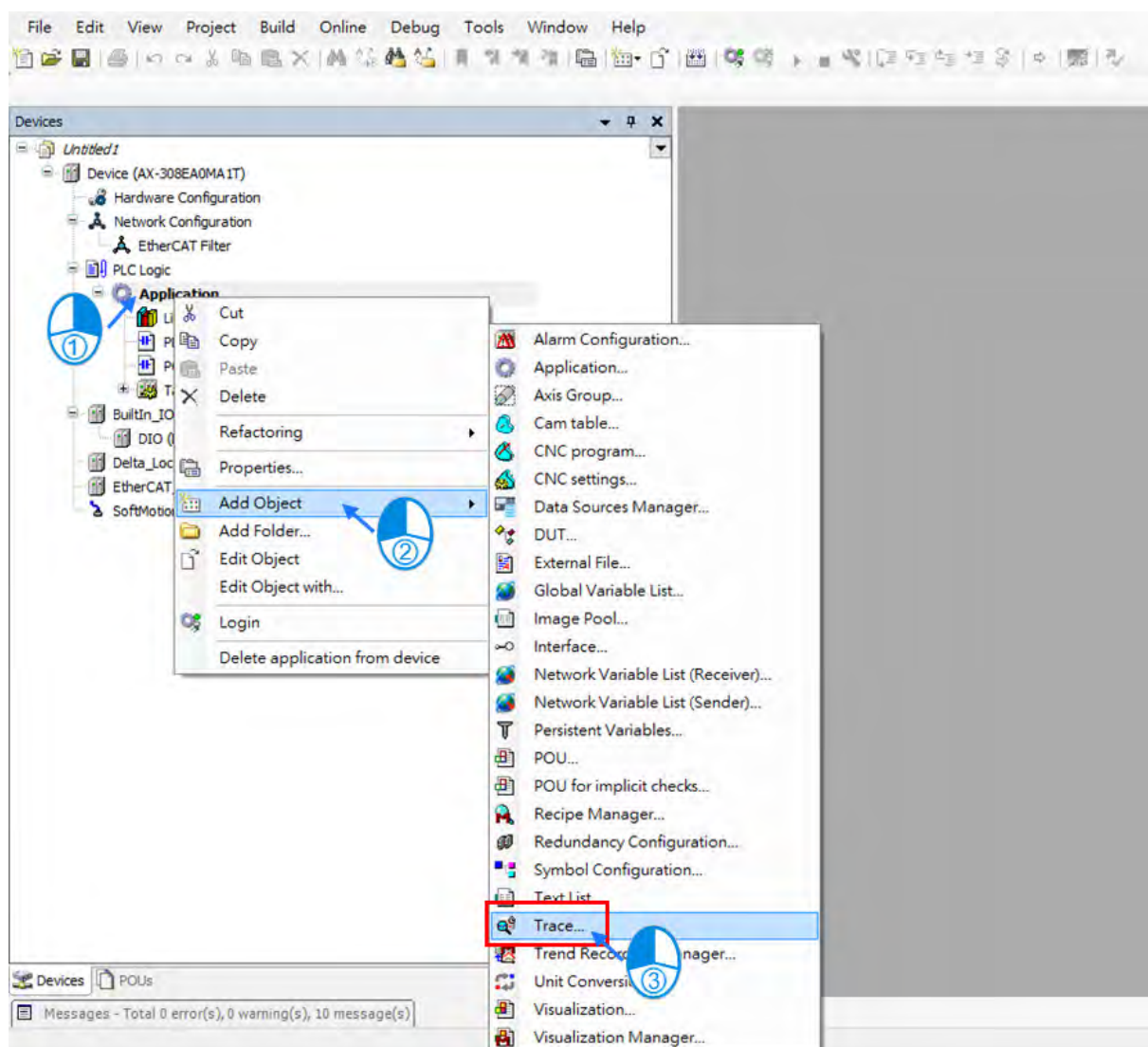
14. 启动「Step3」后，会以下列流程一轴一轴正转后反转，每正反转一次后「Counter_1」会+1，直到「Step3」关闭。

3

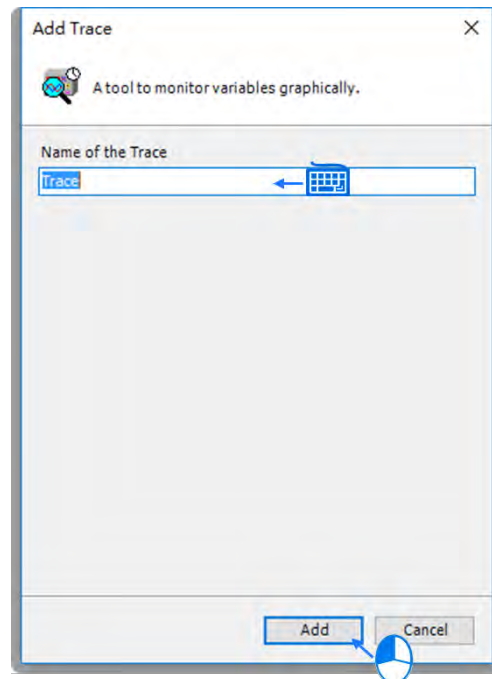


3.2.3.3 示波器监控

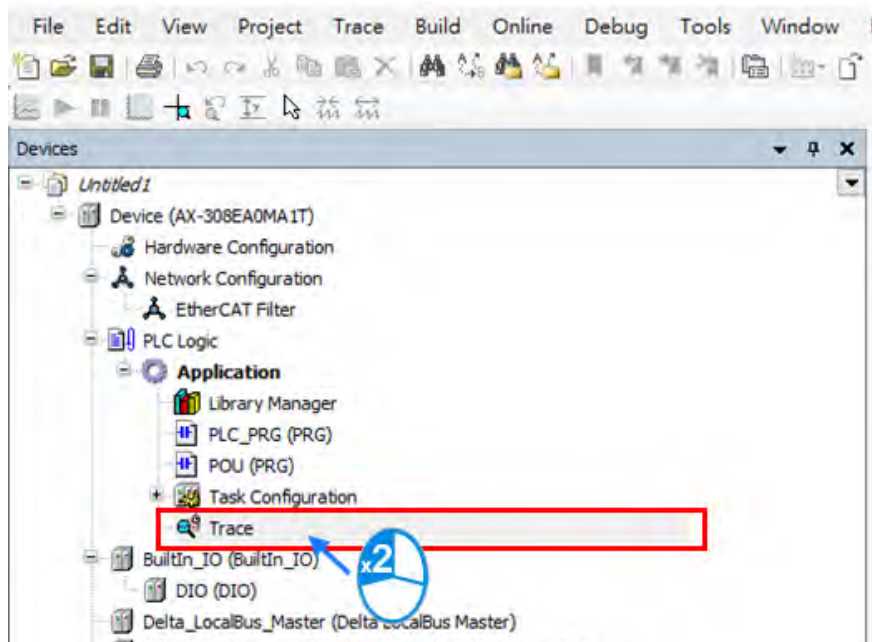
1. 当程序在运行过程中可以使用 Trace 监控的方式，确认引脚状态及轴状态。依序单击「Application」>「Add Object」>「Trace」。



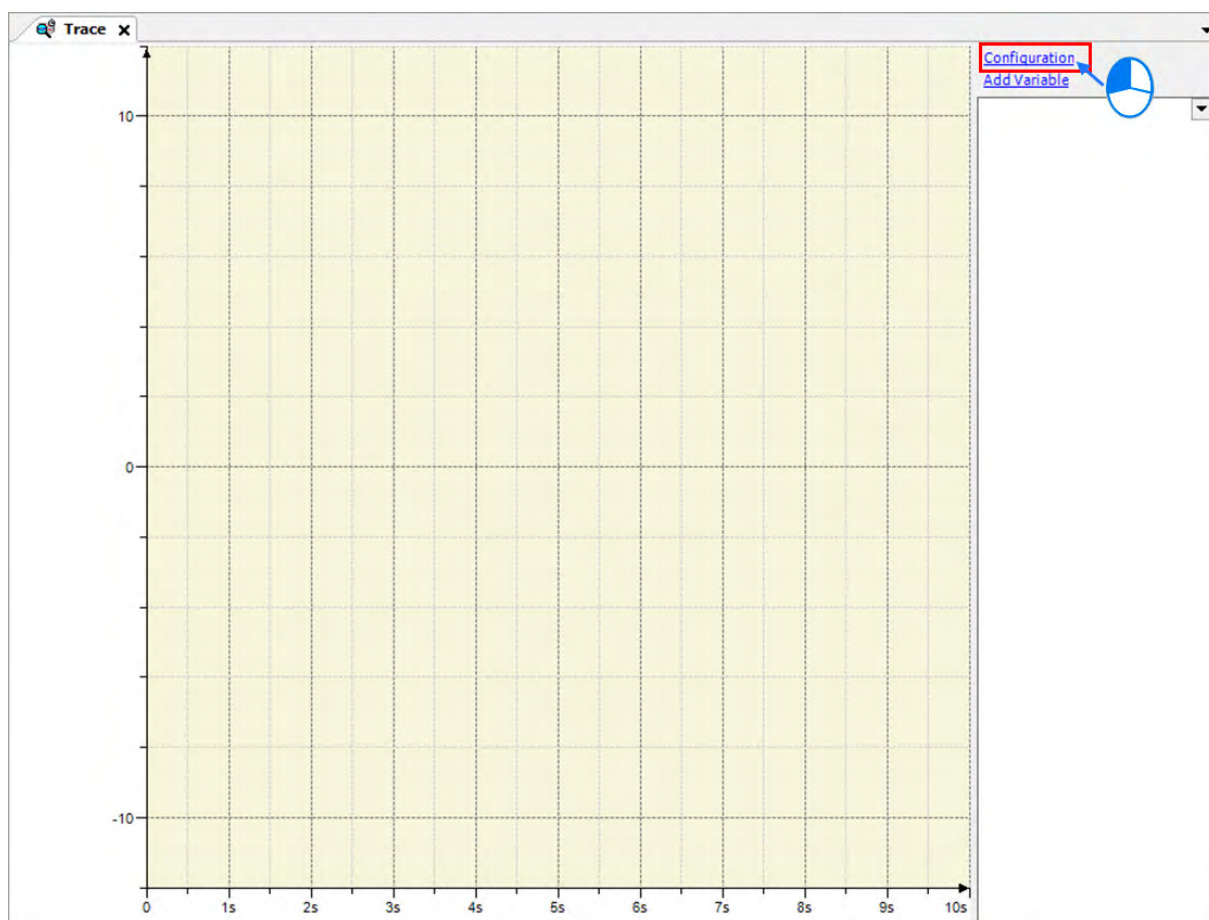
2. 进入「Add Trace」画面中，可输入 Trace 的名称后再单击「Add」。



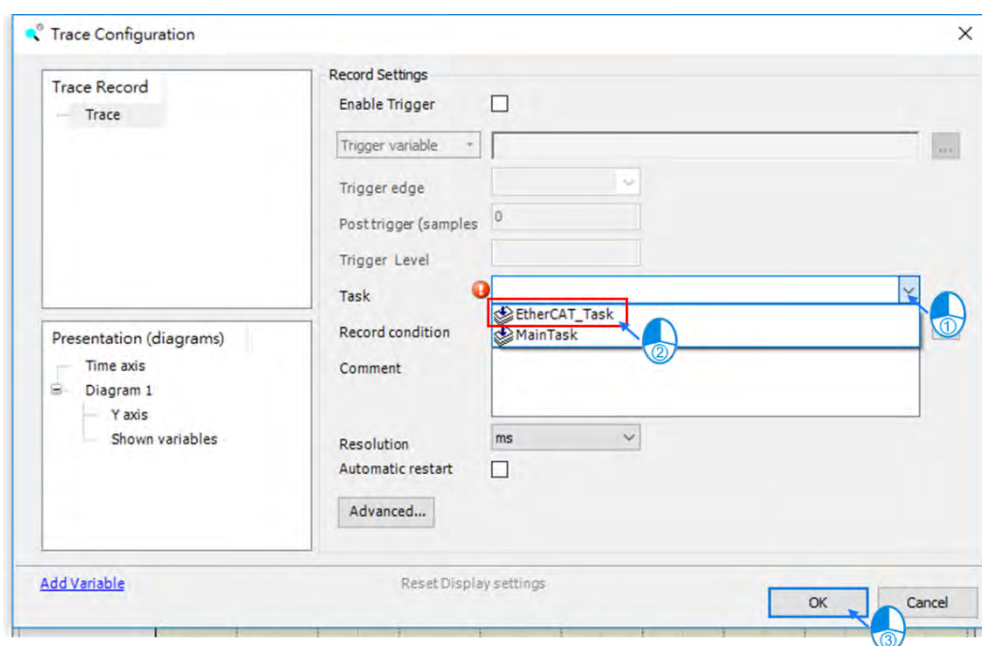
3. 新增完毕后，左边树形图会出现刚刚所新增的示波器，此时双击「Trace」。



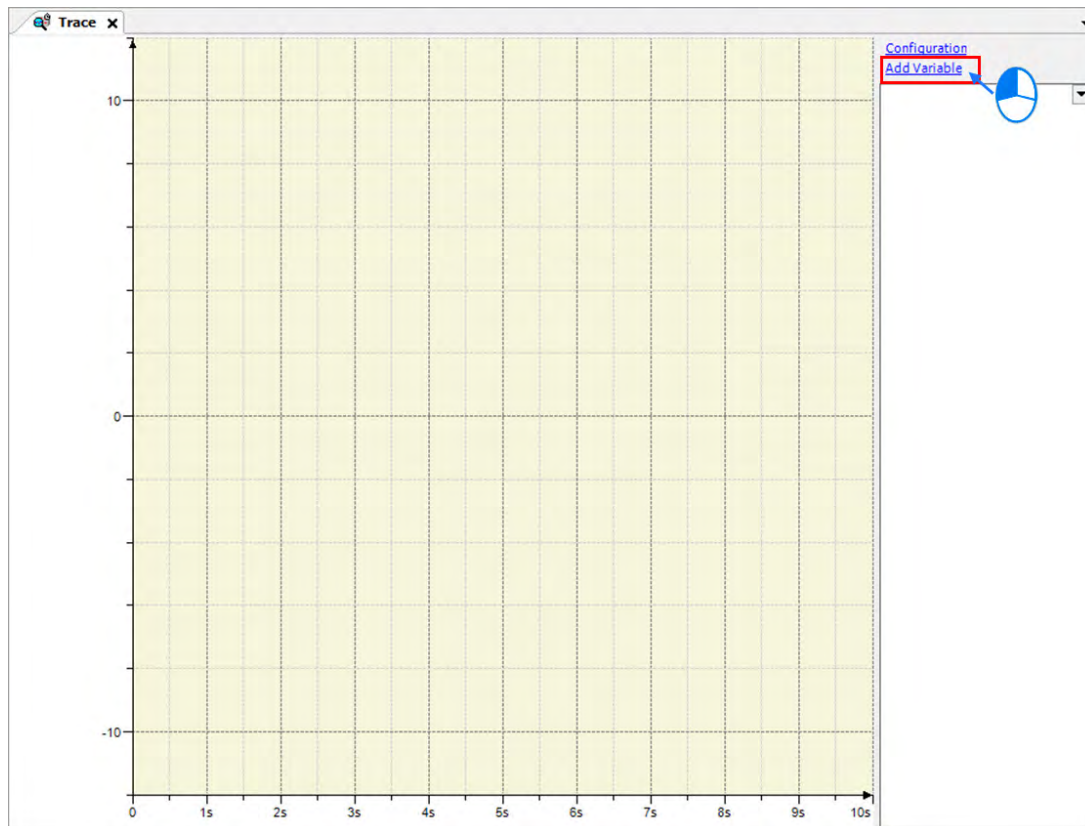
4. 进入「Trace」画面后，单击「Configuration」。



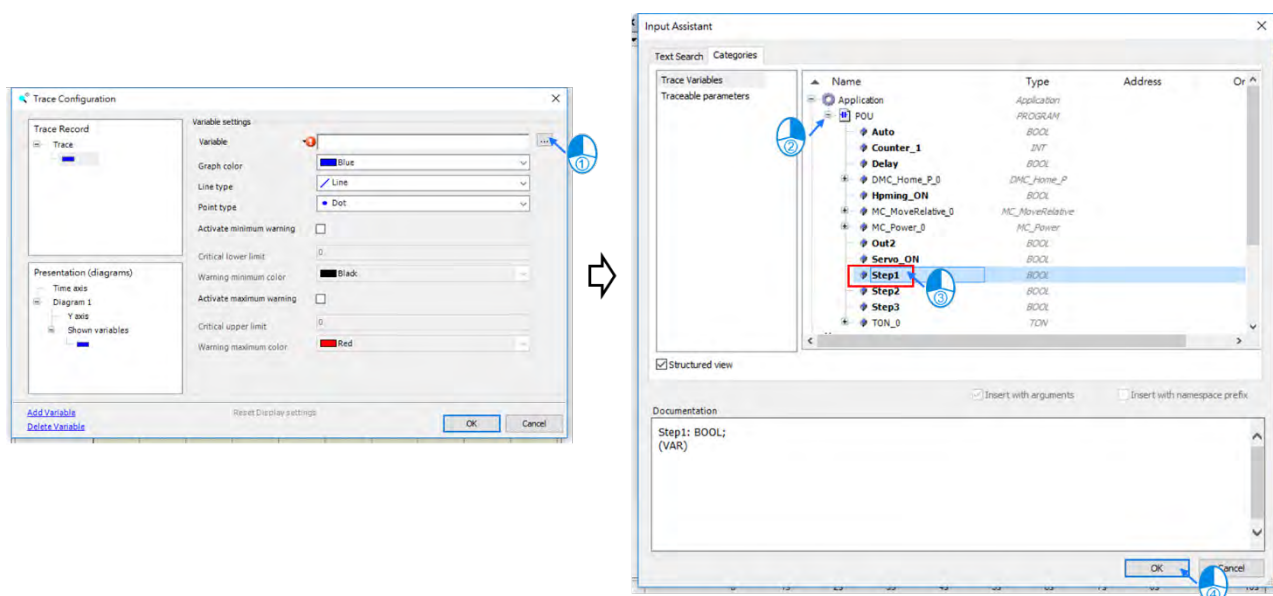
5. 在「Trace」设定画面中，选择「Task」下拉选单中的「EtherCAT_Task」。



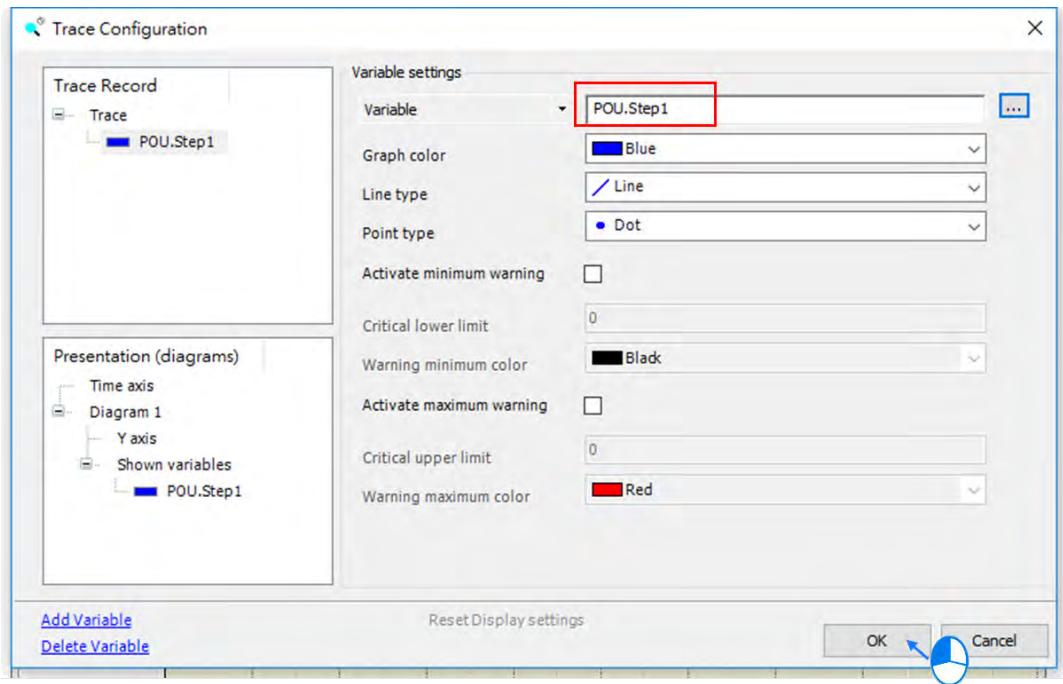
6. 设定完后单击「Add Variable」。



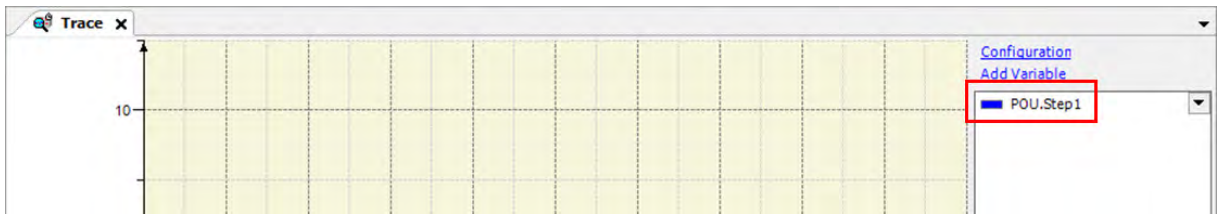
7. 进设定画面后，单击 添加需监控的参数，当进入添加画面后，单击 后选择「Step1」>「OK」。



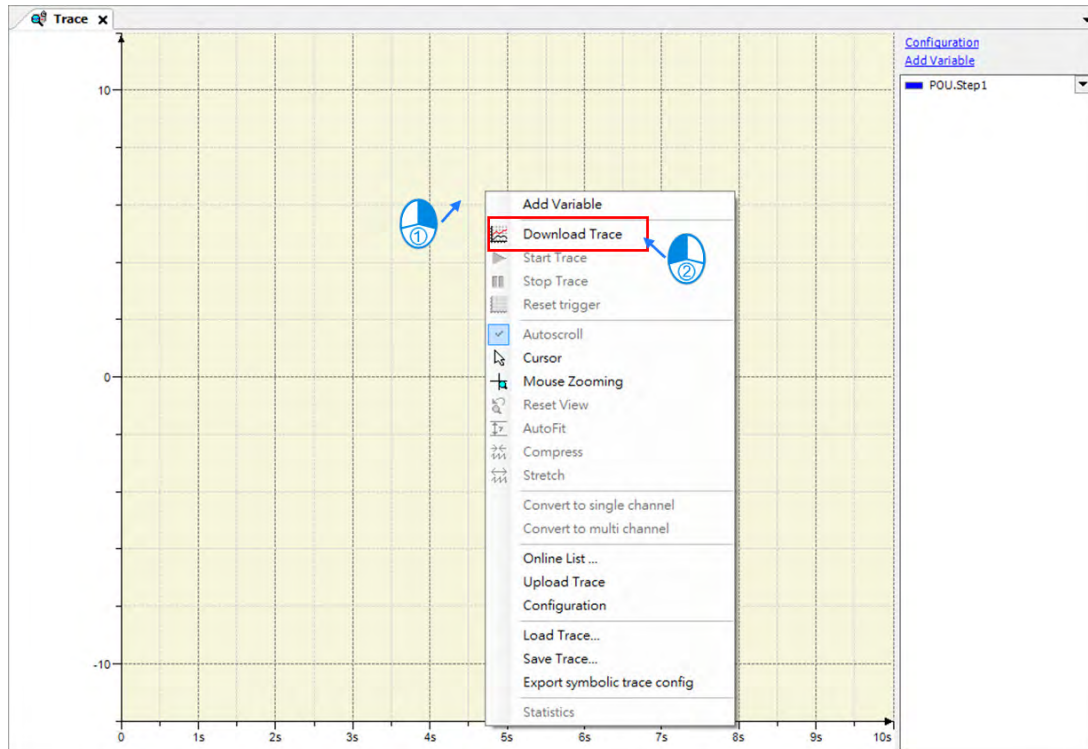
8. 选择好需监控的参数会出现在表格中再单击「OK」。



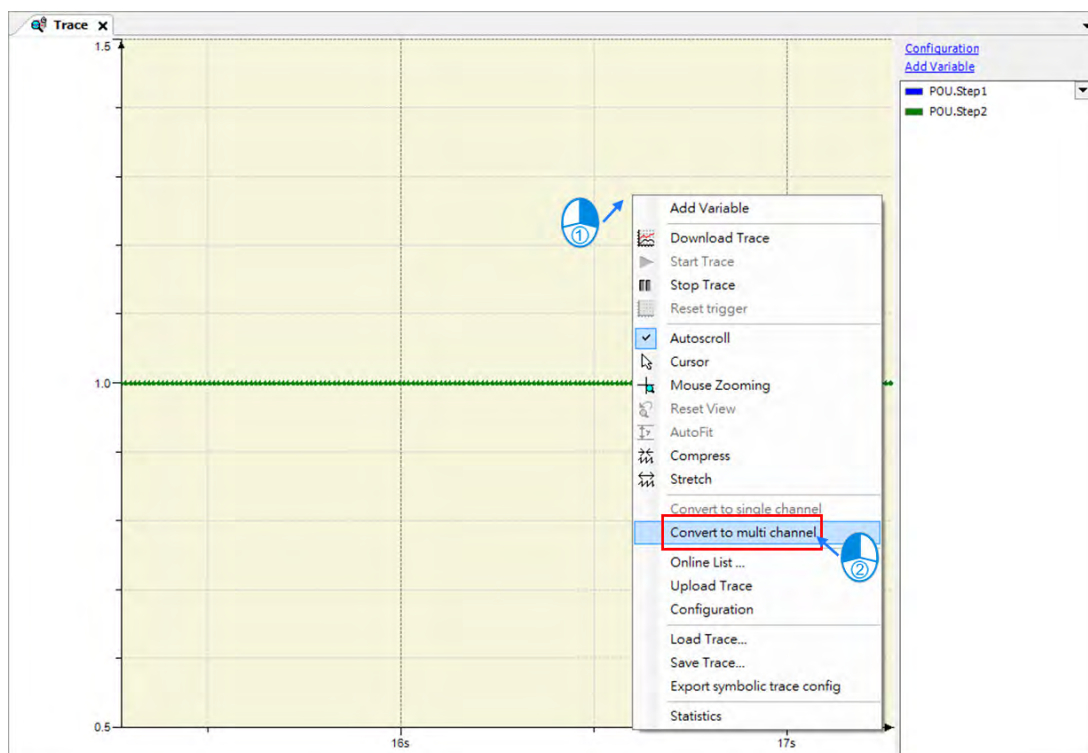
9. 添加完成后，在「Trace」画面右方会出现需监控的参数。



10. 在在线监控时单击「Trace」右键后，选择「Download Trace」就会开始监控。



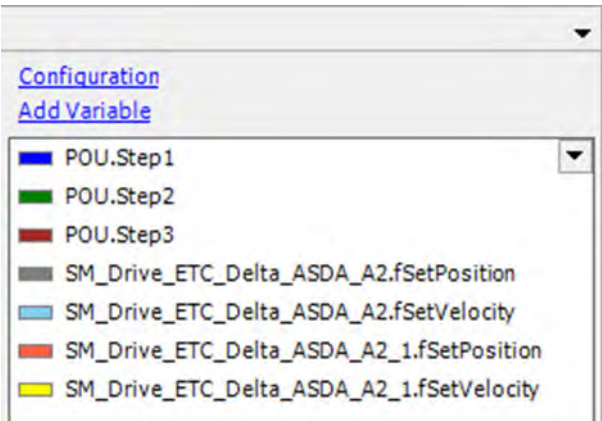
11. 当有两个参数需要监控时，可以在「Trace」画面单击右键，选择「Convert to multi channel」。



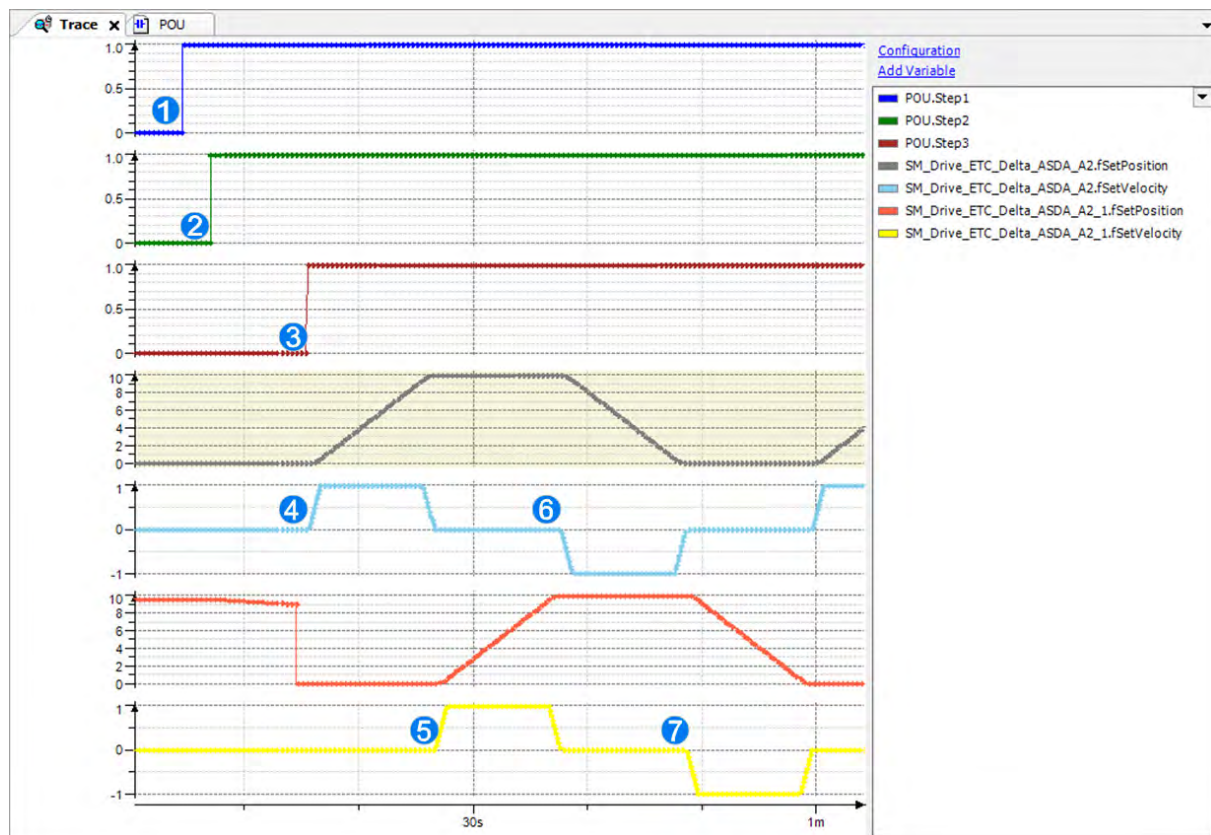
12. 「Convert to multi channel」选择后会以两格的方式显示。



13. 利用上方相同操作新增下列监控值。



14. 通过程序执行过程中，可使用示波器观察触发的开关及轴目前位置及速度。



示波器流程介绍：

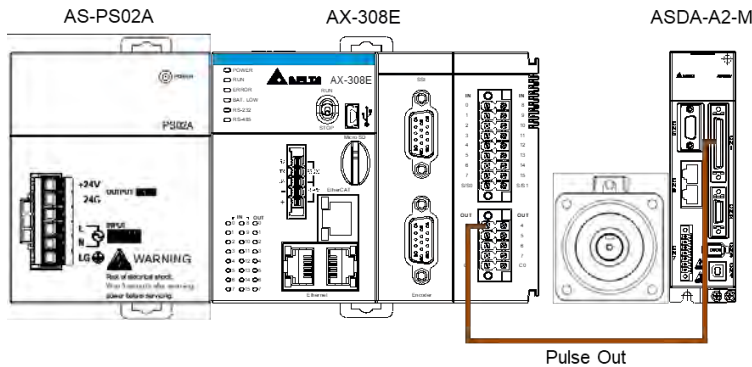
- ①启动「Step1」让 EtherCAT 轴启动。
- ②启动「Step2」让轴回原点。
- ③启动「Step3」后，触发「MC_MoveRelative」功能块让轴 1 与轴 2 正反转。
- ④轴 1 开始正转一段位置。
- ⑤轴 1 旋转完毕后触发轴 2 正转一段位置。
- ⑥轴 2 正转完毕后触发轴 1 反转一段位置。
- ⑦轴 1 反转完毕后触发轴 2 反转一段位置。

3.3 Pulse Out 伺服轴项目建立

此范例为单轴伺服控制并等速度运转后使用定位功能中断前一个指令。

3.3.1 系统硬件配置

AX-308 通过脉冲输出连接伺服电机 (ASDA-A2-M)，以脉冲方式控制伺服电机等速运转后定位。

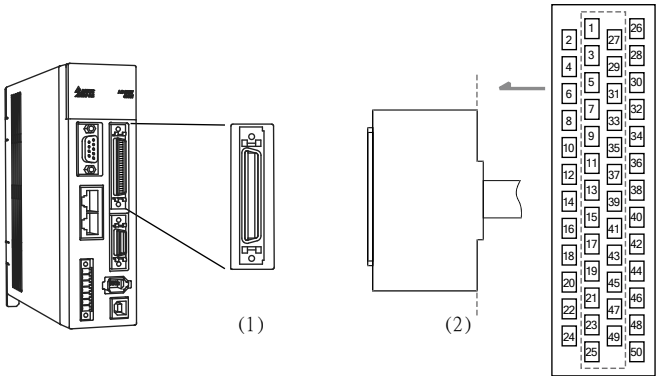


● 材料与工具

编号	名称	型号	数量
1	AS 系列电源模块	AS-PS02A	1
2	AX3 系列主机	AX-308EA0MA1T	1
3	电源供应器	DVP-PS02	1
4	脉冲型伺服驱动器	ASDA-A2-0421-M	1
5	伺服电机	ECMA-C10604ES	1
6	伺服动力线	ASD-ABPW0003	1
7	伺服编码线	ASD-ABEN0003	1
8	CN1 端子台	ASD-BM-50A	1

● 控制器搭配 ASDA-A2-M 硬件配置

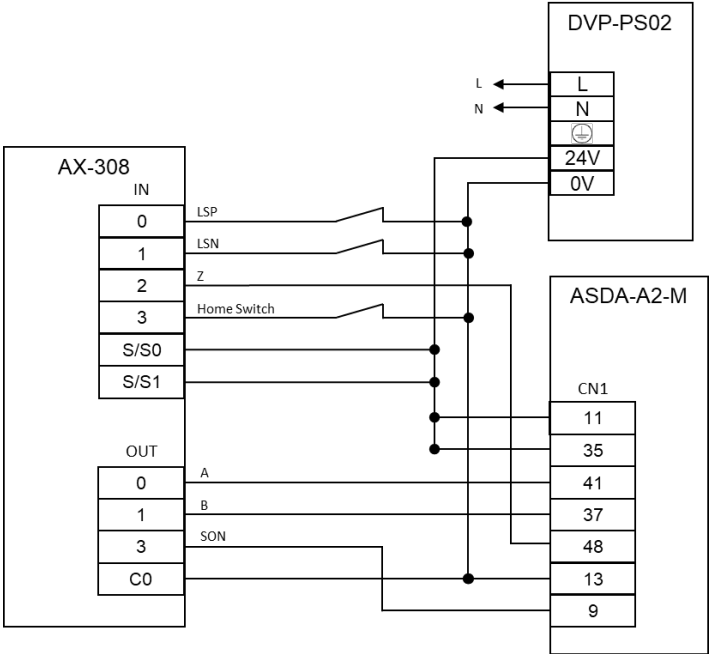
使用脉冲输出需配置硬件线路以及硬件极限需配置在控制器端。



(1) CN1 端子座图；(2) CN1 线端插头配线定义图

◆ CN1 信号介绍

信号名称		Pin No	功能
位置脉冲 命令 (输入)	PULSE	43	集极开路 (单相最高脉冲频率 200 KHz) 方式输入，命令的形式也可分成三种 (正逆转脉冲、脉冲与方向、AB 相脉冲)，可由参数 P1-00 来选择。 当位置脉冲使用集极开路方式输入时，必须将本端子连接至一外加电源，作为提升准位用。
	/PULSE	41	
	SIGN	36	
	/SIGN	37	
	PULL HI_P	39	
	PULL HI_S	35	
位置脉冲 命令 (输出)	OCZ	48	编码器 Z 相，开集极输出。
电源	VDD	17	VDD 是驱动器所提供的+24V 电源，用以提供 DI 与 DO 信号使用，可承受 500 mA。
	COM+ COM-	11 45 47 49	COM+是 DI 与 DO 的电压输入共同端，当电压使用 VDD 时，必须将 VDD 连接至 COM+。若不使用 VDD 时，必须由使用者提供外加电源 (+12 V ~ +24 V)，此外加电源的正端必须连至 COM+，而负端连接至 COM-。
	VCC	20	VCC 是驱动器所提供的+12V 电源，用以提供简易的模拟命令 (速度或扭矩) 使用，可承受 100 mA。
	GND	12 · 13 · 19 · 44	VCC 电压的基准是 GND。
DI1	SON	9	当 ON 时，伺服回路启动，电机线圈激磁。



注：ASDA-A2-M 信号配置详细信息可参考 **ASDA-A2 系列应用技术手册**

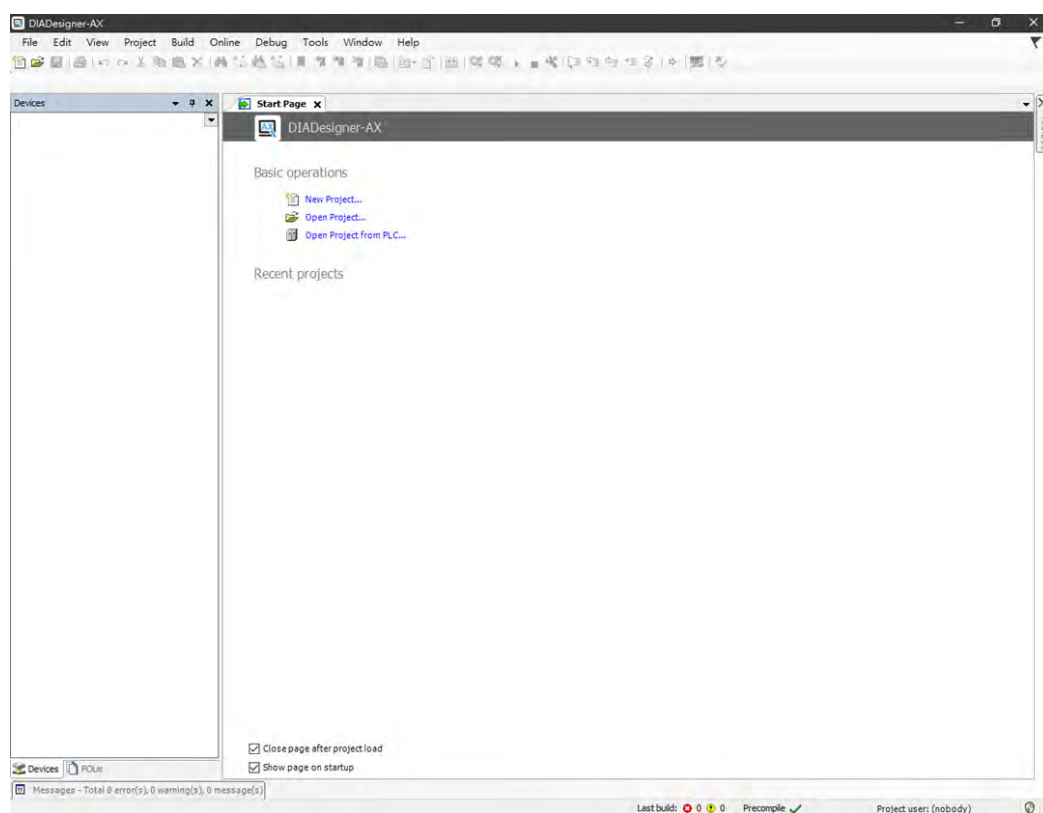
3.3.2 项目建立与程序编写

3.3.2.1 项目开启

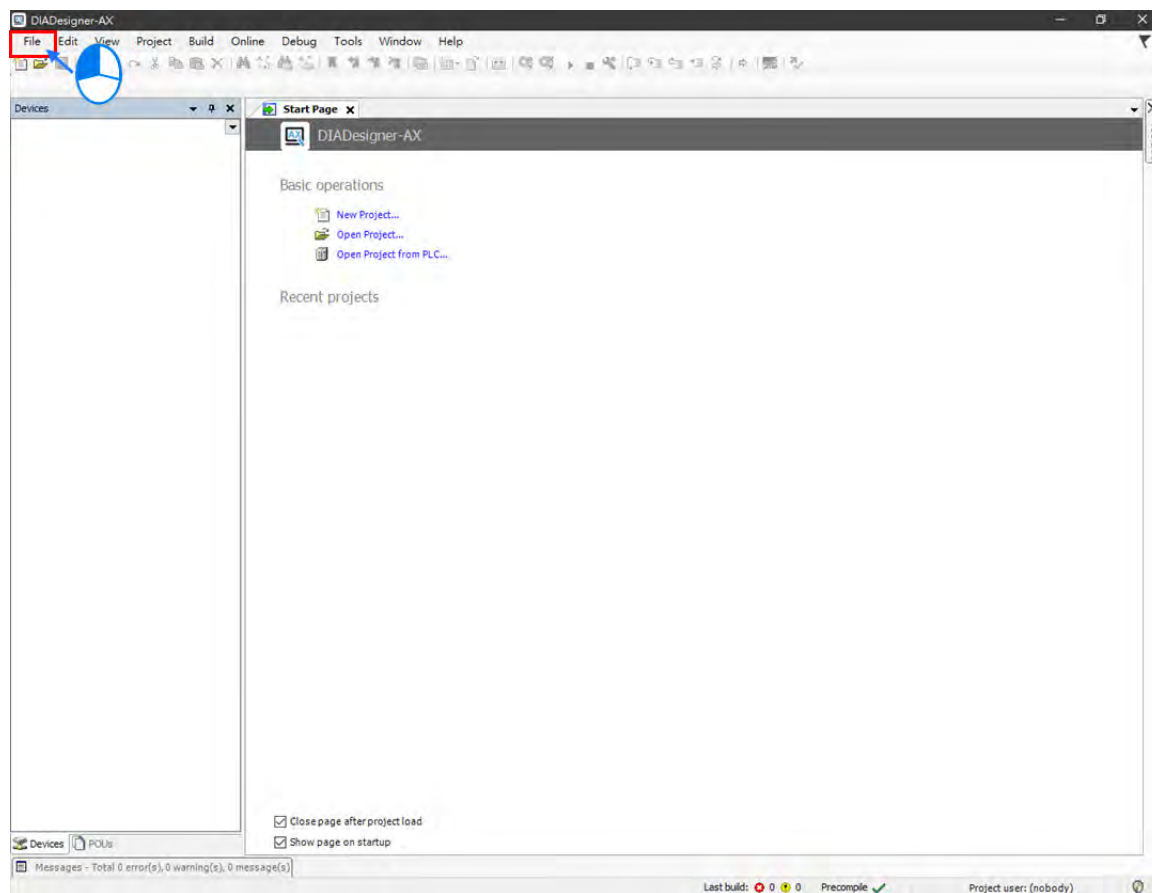
1. 首先 AX-308 编写程序软件为 DIADesigner-AX，所以需先启动软件。



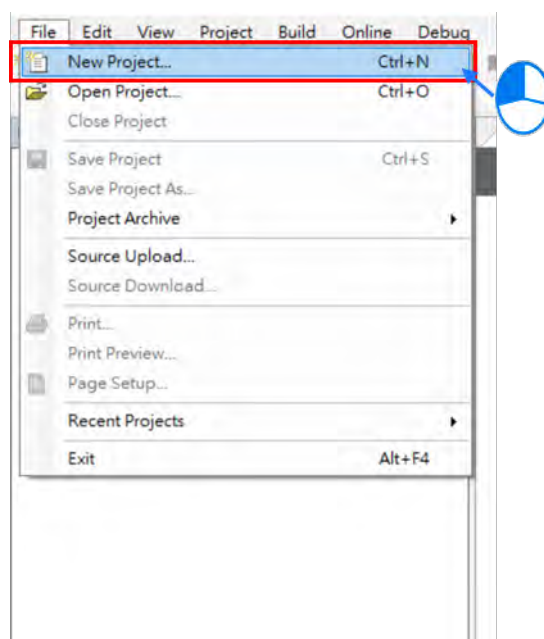
2. 启动 DIADesigner-AX 的画面如下



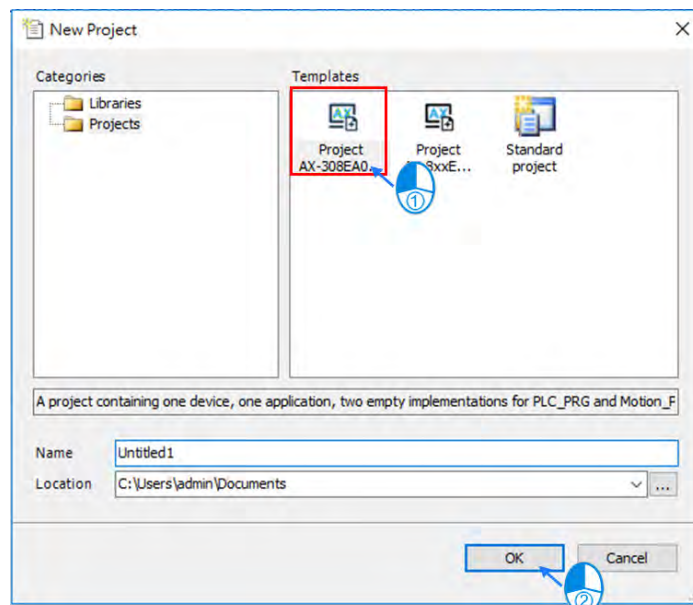
3. 单击「File」后双击鼠标左键



4. 在「File」下拉选单单击「New Project」以开启一个新项目。

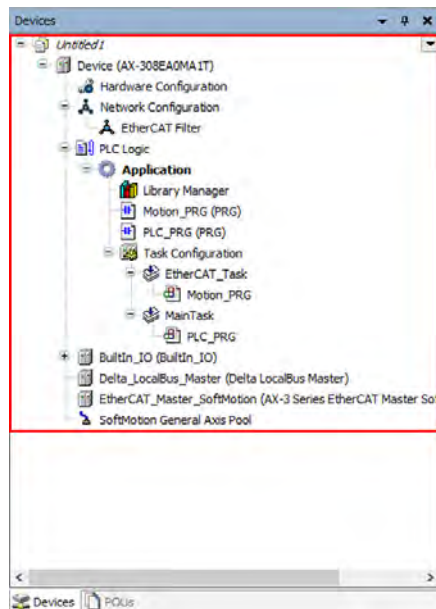


5. 进入「New Project」画面后选择「Project AX-308EA0MA1T*」，可在「Name」处输入项目名称，及「Location」处输入项目存取路径。

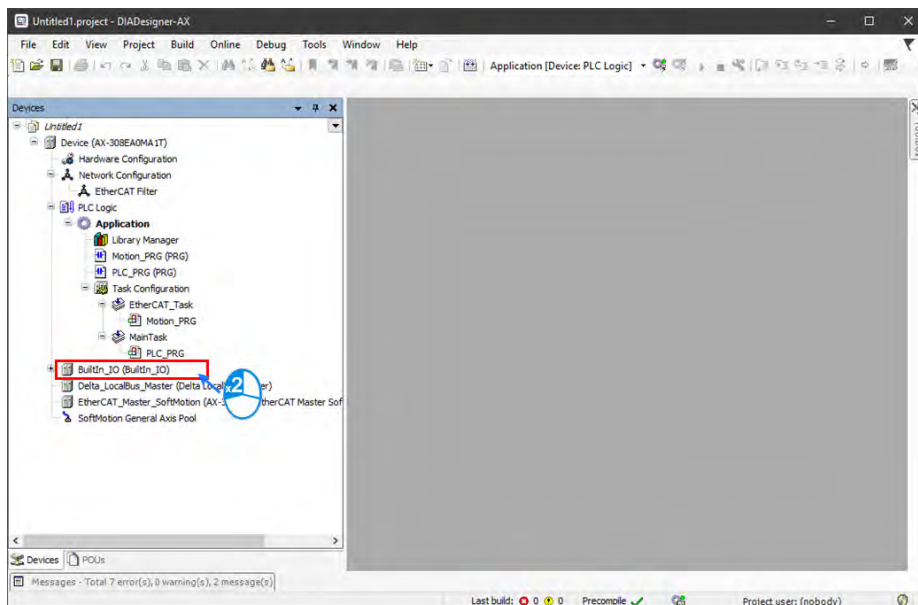


*注：DIADesigner-AX 版本于 V1.1.0 以上的操作为点选「Standard Project」后选择机种。

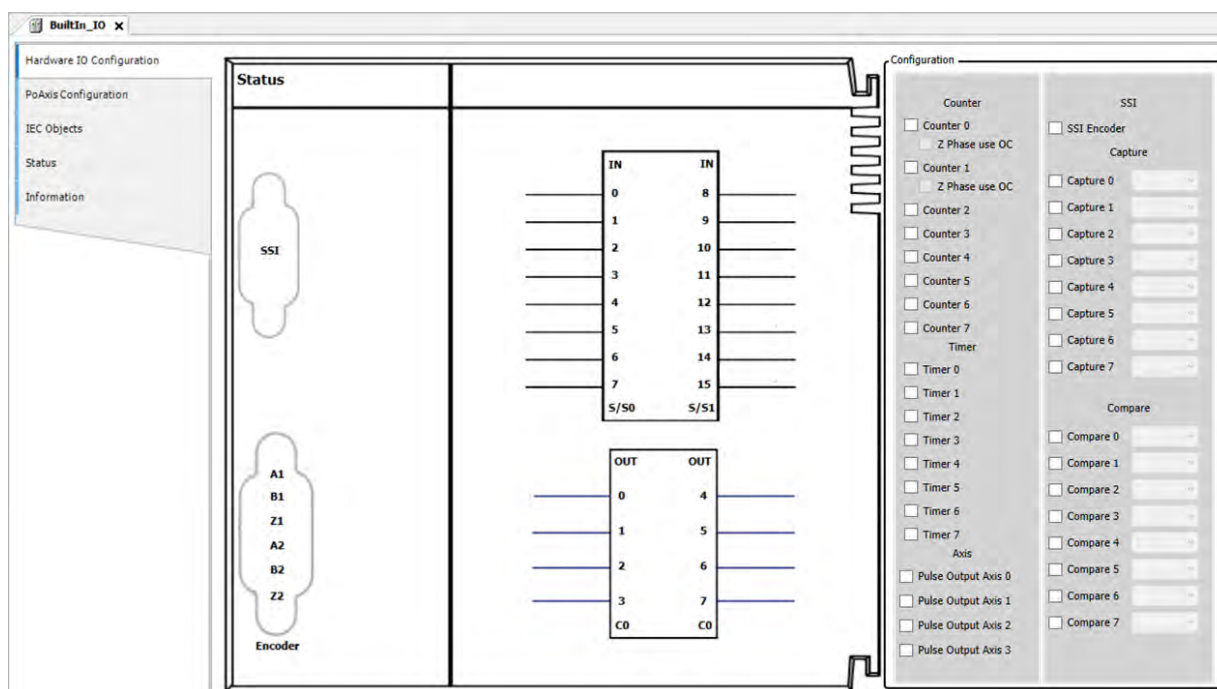
6. 进入项目画面如下，左边为项目树，可双击左键以开启程序画面或是模块相关设定。



7. 建立系统架构，首先双击「BuiltIn_IO」。

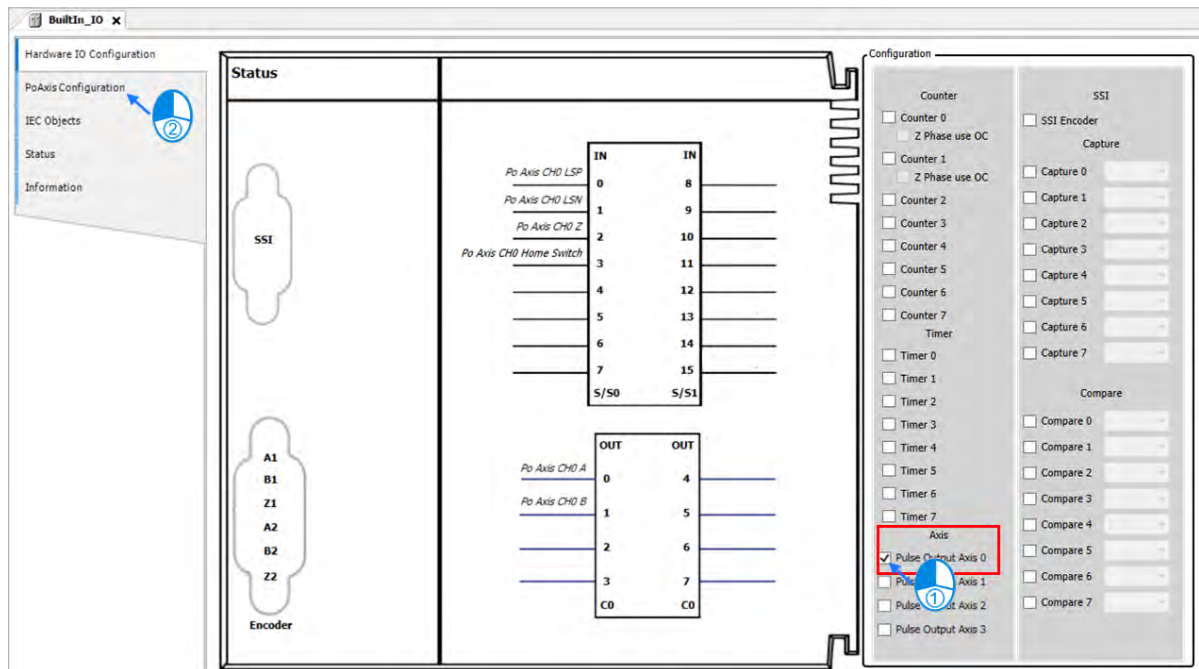


8. 进入「BuiltIn_IO」画面后，可设定 I / O 相关类型的功能。

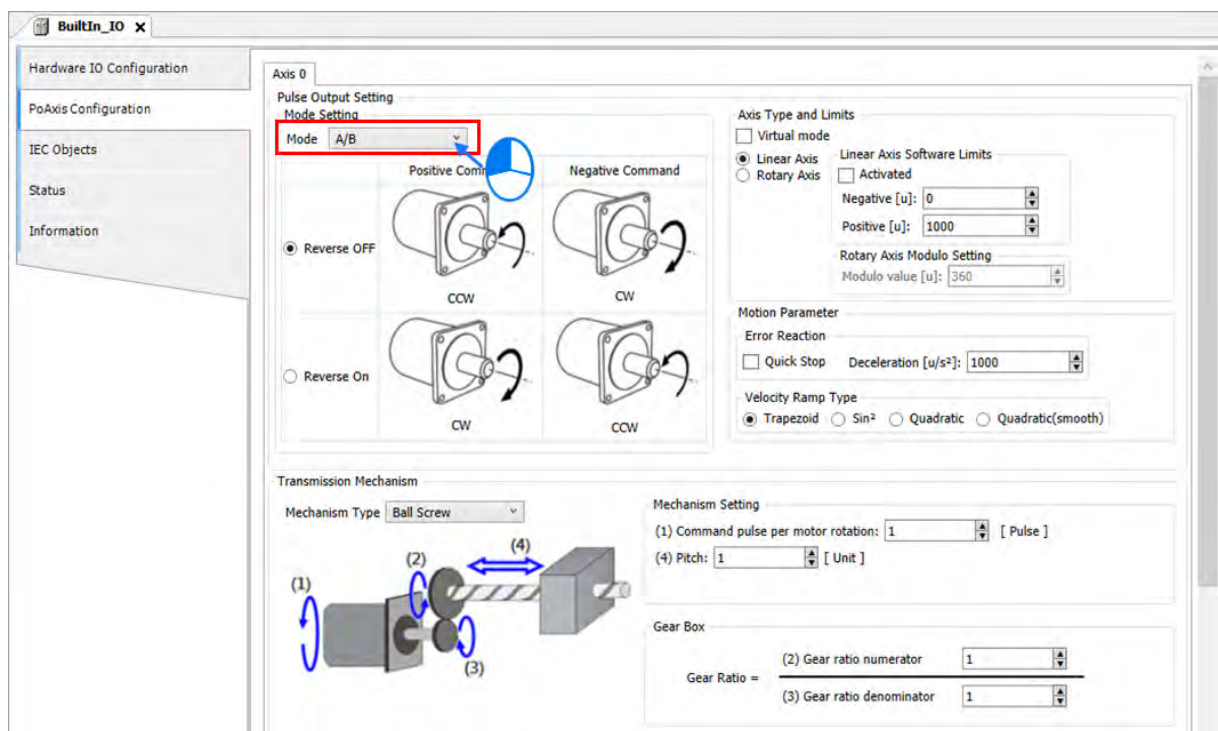


3.3.2.2 轴参数设定

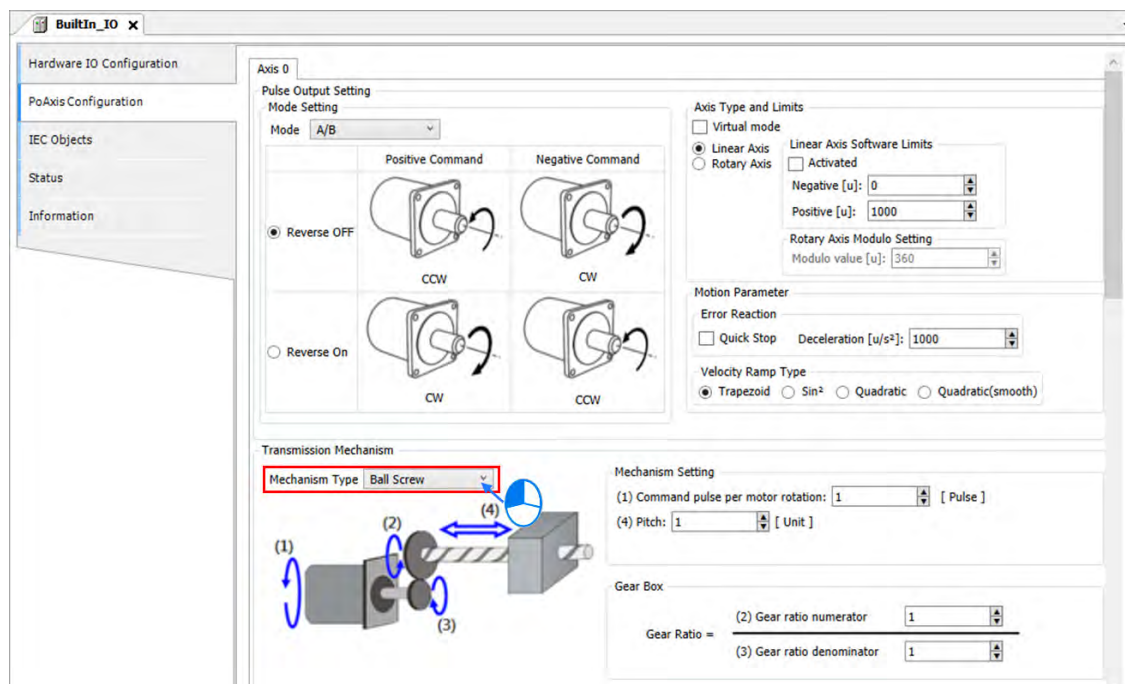
1. ①在右方「Configuration」下勾选「☐ Pulse Output Axis 0」后，②再单击左方的「PoAxis Configuration」。



2. 进入「PoAxis Configuration」分页后，选择「Mode」设定脉冲输出形式。



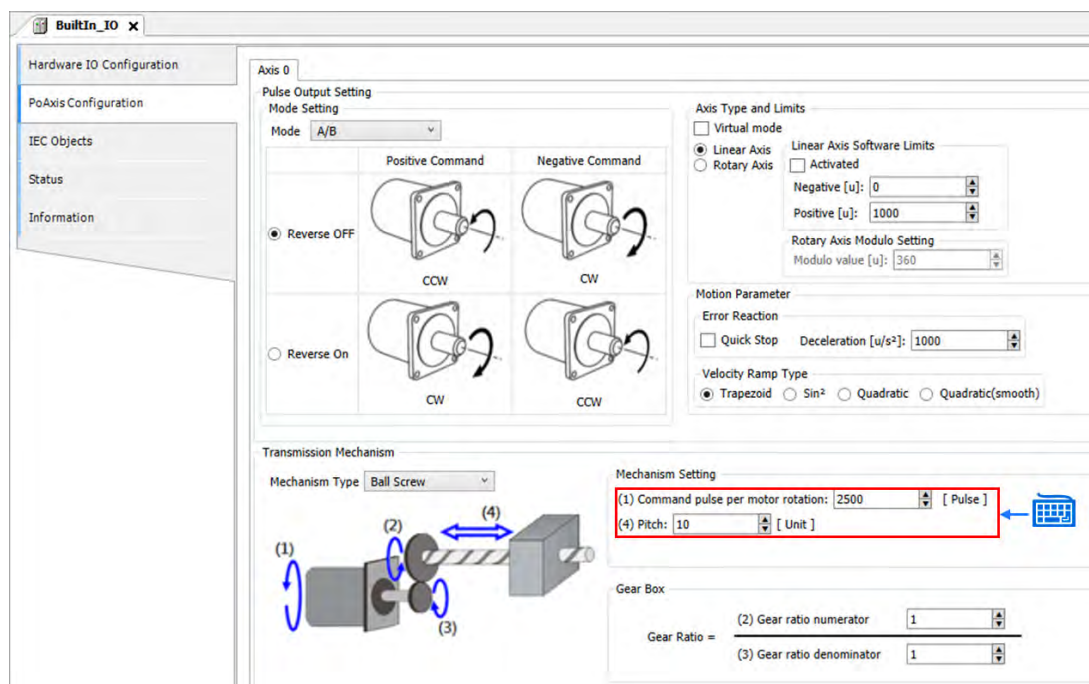
3. 实体机构类型，此范例为螺杆机构，所以选择「Ball Screw」机构。



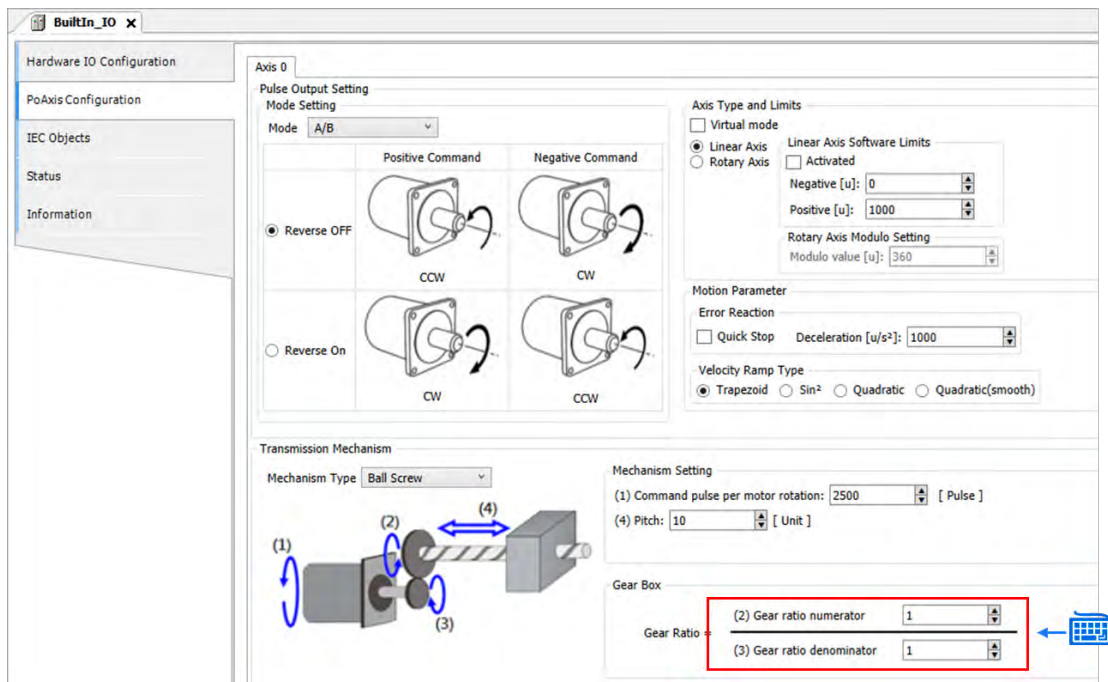
4. 实体机构选择完毕再设定齿比关系，一圈脉冲 (Command pulse per motor rotation) 输入 2,500 (此参数需设定 10,000 因伺服端 AB 相是 4 倍，所以 $10,000/4=2,500$)，螺杆间距为 10 mm，所以下方设定 10。(伺服端

P1-44 = 128、P1-45 = 1 需设定齿比) 因此范例用户单位 (User Unit) 为 10mm。

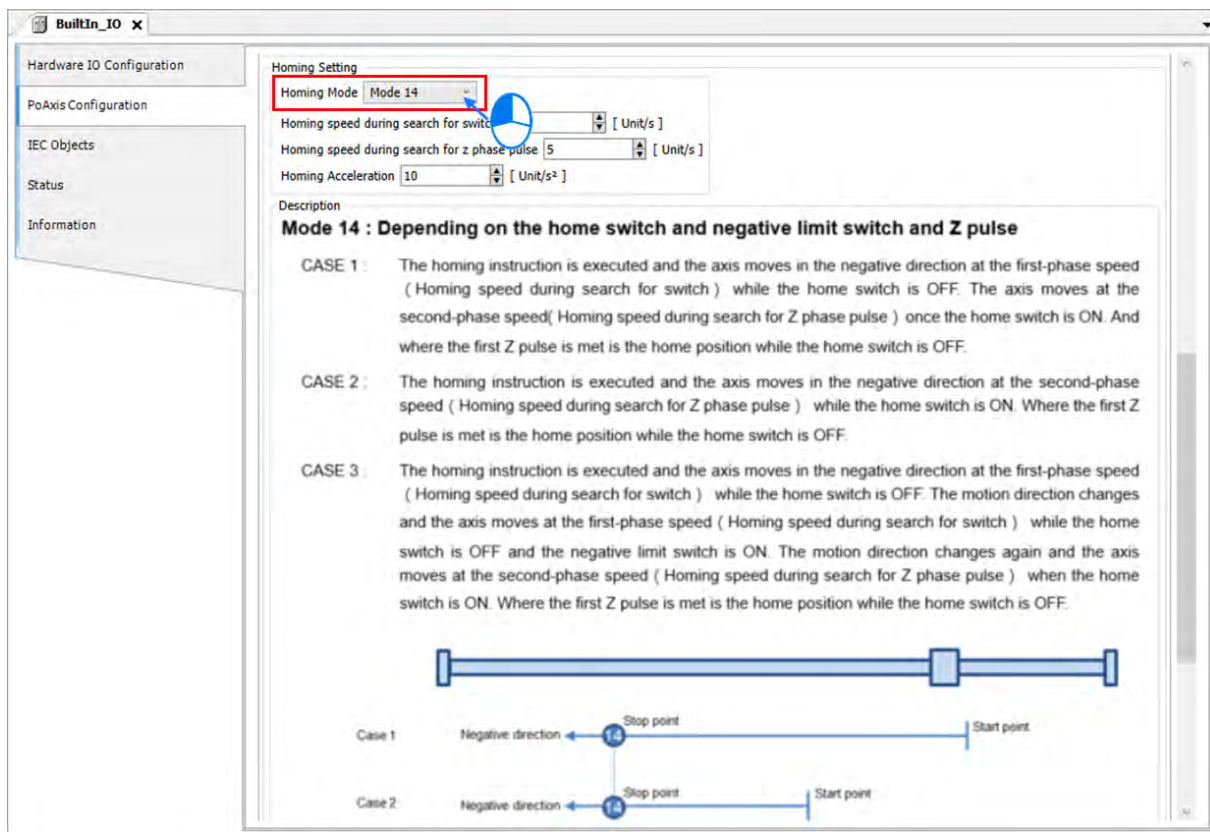
$$\left(\text{伺服端齿比关系} \frac{P1-44}{P1-45} = \frac{\text{电机一圈脉冲数}}{\text{Command Pulse per rotation}} \right)$$



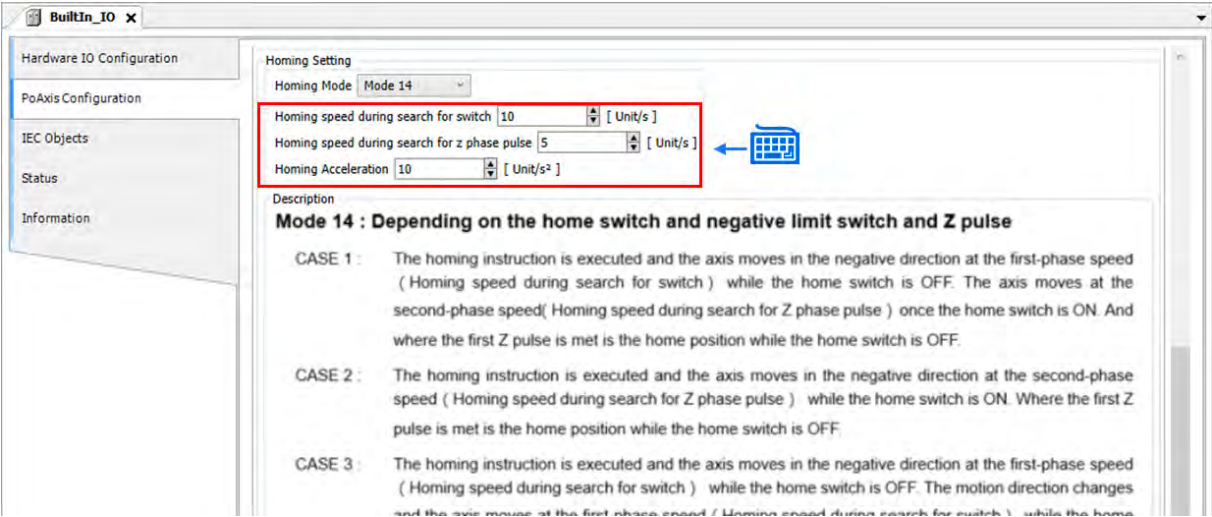
5. 因机构未装设减速机，所以在下方输入减速机的比例为 1 : 1。



6. 将伺服齿比参数设定完毕后，再设定轴回原点方式，回原点方式需搭配硬件如何配置，每种模式请参考下方详细说明。



7. 设定回原点速度，寻找原点开关速度设定 10 (Homing speed during search for switch)，寻找 Z 相速度设定 5mm/s(Homing speed during search for z phase pulse) 回原点加速度设定 10 mm/s(Homing Acceleration)，用户单位：mm。

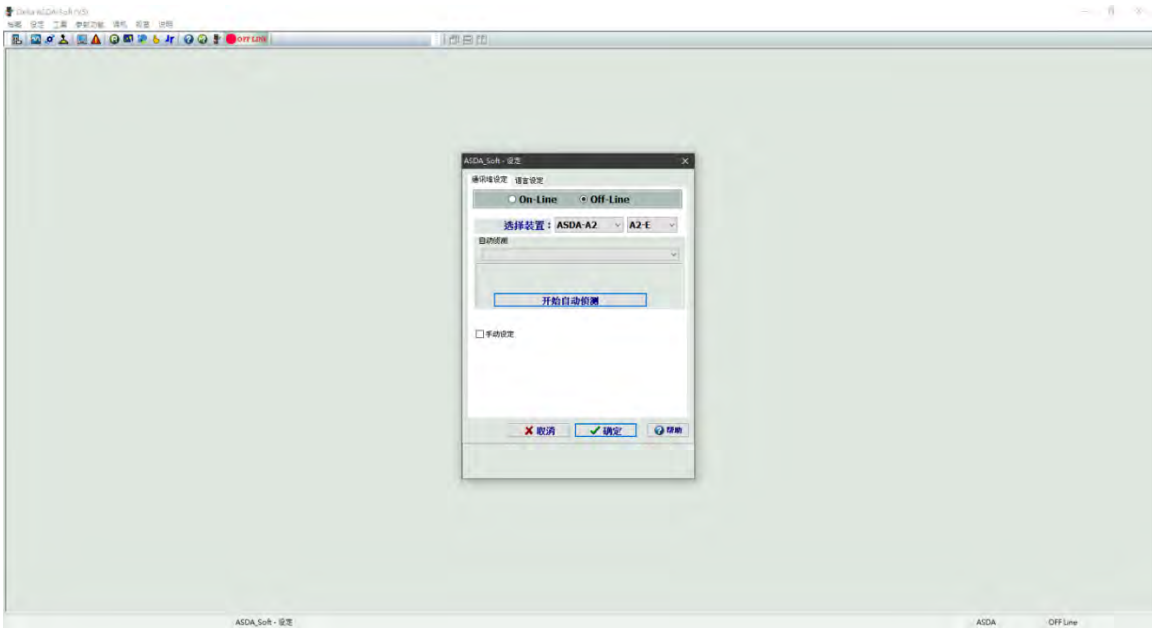


3.3.2.3 设定伺服参数

1. 设定伺服参数有两种方法，第一种通过伺服驱动器上的面板设定参数，第二种为软件设定「ASDA_SOFT」；此范例为软件操作，先开启软件「ASDA_SOFT」。



2. 启动「ASDA_SOFT」的画面如下



3. 在「On-Line」下选择连接伺服的 COM Port 后，再单击「开始自动侦测」。



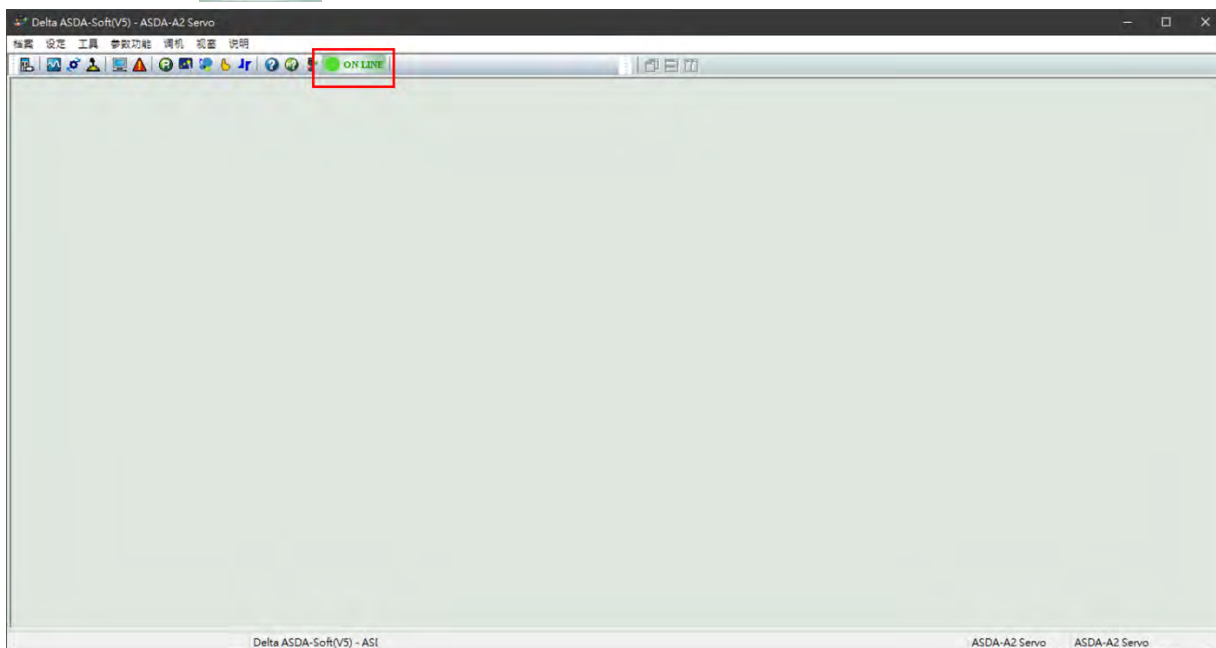
4. 当有侦测成功会跳出以下窗口。



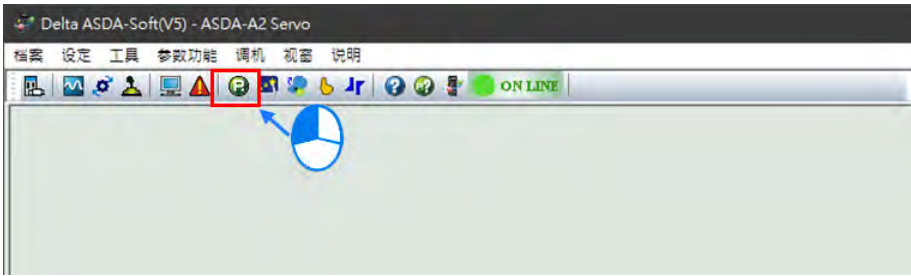
5. 确认通讯正常后，单击「确定」。



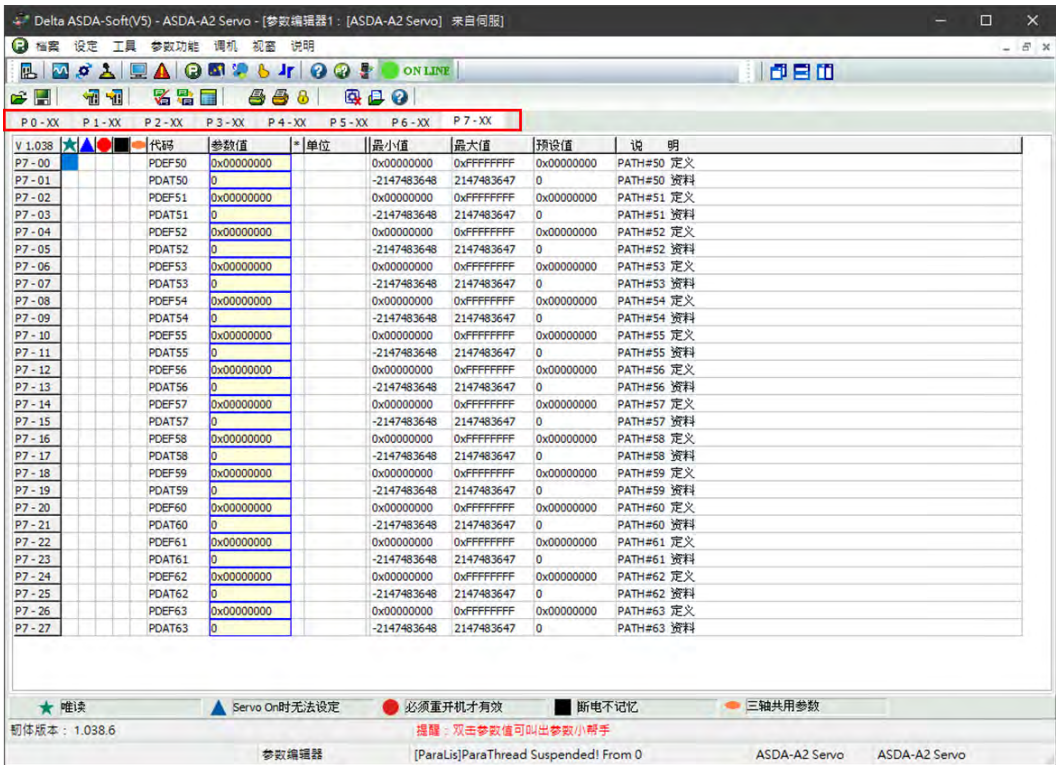
6. 此时状态会呈现 **ON LINE**，代表目前 PC 与伺服驱动的通信链接中。



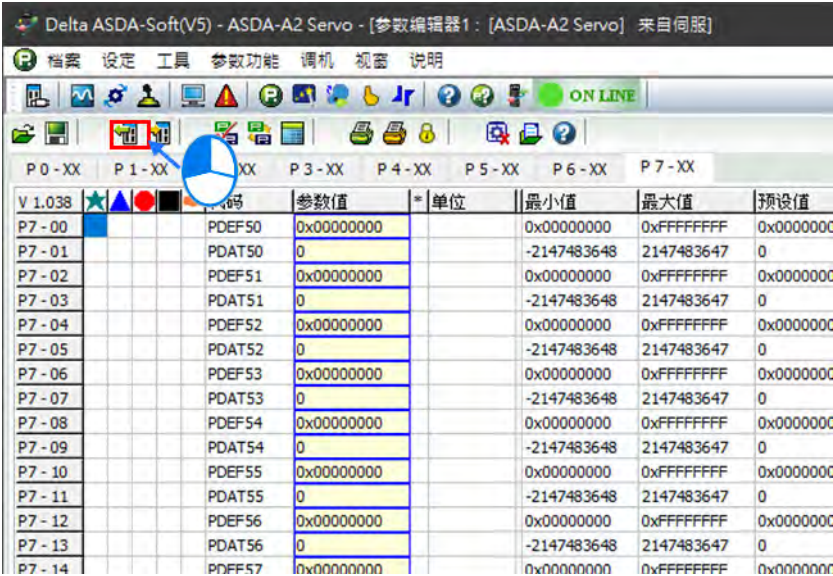
7. 通信链接中，鼠标单击。



8. 进入伺服参数画面后，上方是伺服所有参数，如下红框所示。



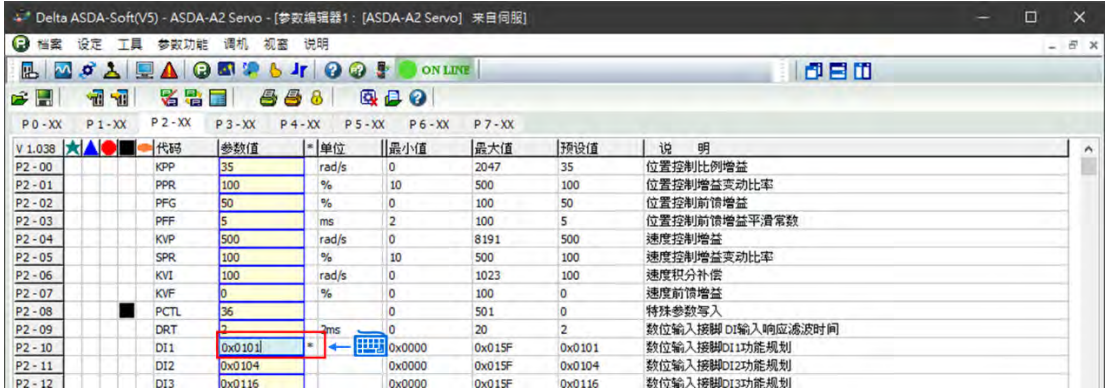
9. 单击 会将伺服所有参数上载。



10. 参数上载完毕后单击「P2-XX」，再单击 P2-10 参数的数值。



11. 此时可输入设定值，可直接输入或单击两下开启设定画面。



12. 进入设定画面后，可选择「功能选项」> 设定「接点极性」，设定完毕后，单击「OK」。



13. 当设定值有改变会出现 *，可单击设定值并按 Enter 键，将设定值输入至伺服驱动器中。

Delta ASDA-Soft(V5) - ASDA-A2 Servo - [参数编辑器1: [ASDA-A2 Servo] 来自伺服]

档案 设定 工具 参数功能 调机 视窗 说明

P 0 - XX	P 1 - XX	P 2 - XX	P 3 - XX	P 4 - XX	P 5 - XX	P 6 - XX	P 7 - XX		
V 1.038	★▲●■	代码	参数值	*	单位	最小值	最大值	预设值	说 明
P2 - 00		KPP	35		rad/s	0	2047	35	位置控制比例增益
P2 - 01		PPR	100		%	10	500	100	位置控制增益变动比率
P2 - 02		PFG	50		%	0	100	50	位置控制前馈增益
P2 - 03		PFF	5		ms	2	100	5	位置控制前馈增益平滑常数
P2 - 04		KVP	500		rad/s	0	8191	500	速度控制增益
P2 - 05		SPR	100		%	10	500	100	速度控制增益变动比率
P2 - 06		KVI	100		rad/s	0	1023	100	速度积分补偿
P2 - 07		KVF	0		%	0	100	0	速度前馈增益
P2 - 08		PCTL	36			0	501	0	特殊参数写入
P2 - 09		DRT	2		2ms	0	20	2	数位输入接脚 DI1 输入响应滤波时间
P2 - 10		DI1	0x0101	*		0x0000	0x015F	0x0101	数位输入接脚DI1功能规划
P2 - 11		DI2	0x0104			0x0000	0x015F	0x0104	数位输入接脚DI2功能规划
P2 - 12		DI3	0x0116			0x0000	0x015F	0x0116	数位输入接脚DI3功能规划

14. 当设定值输入至伺服驱动器后，* 就会消失，如果参数更改很多，可以通过将所有参数下载至伺服驱动器中。



■ 伺服端参数介绍

参数	功能	设定值
P1-00	外部脉冲列输入型式设定	0x0000
P1-44	电子齿轮比分子	128
P1-45	电子齿轮比分母	1
P2-10	数字输入引脚 DI1 功能规划	0x0101
P2-15	数字输入引脚 DI6 功能规划	0x0100
P2-16	数字输入引脚 DI7 功能规划	0x0100
P2-17	数字输入引脚 DI8 功能规划	0x0100

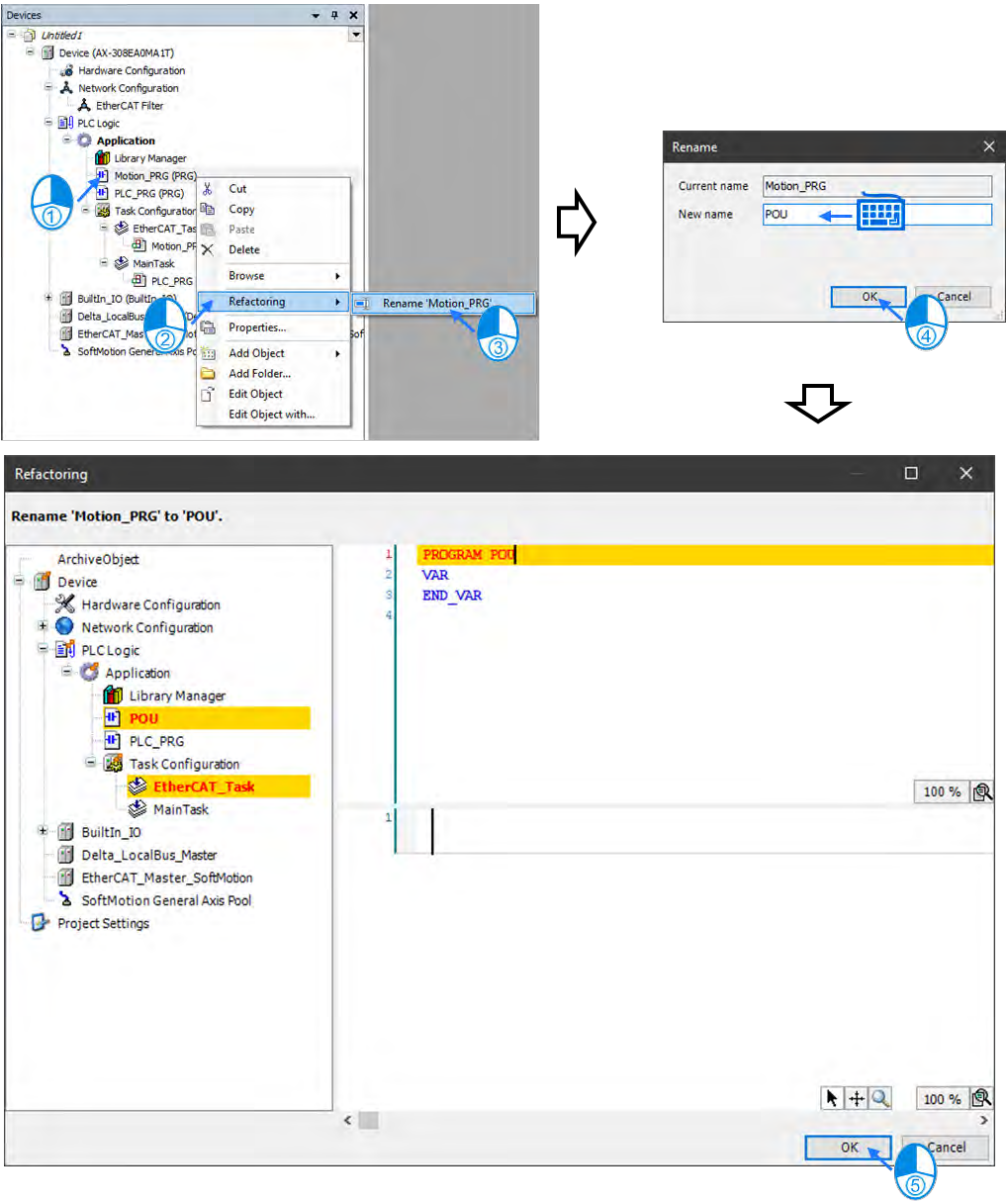
注：参数详细介绍可参考 **ASDA-A2 系列应用技术手册**

3.3.2.4 项目程序编写

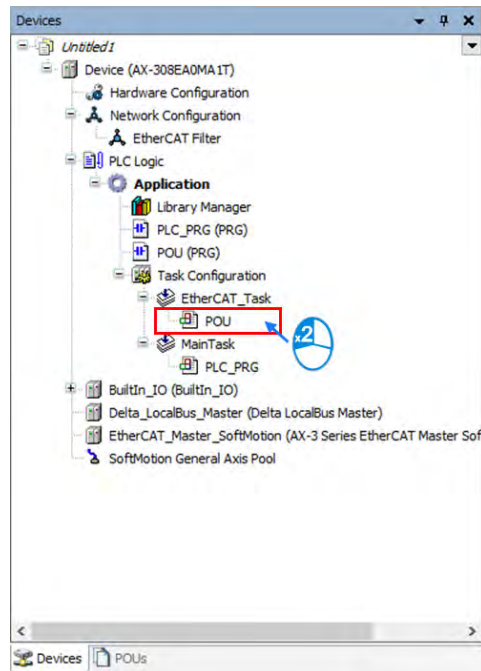
- 程序执行流程
 - 执行「Step1」后会触发「MC_Power」，让轴 Servo ON。
 - 执行「Step2」后会触发「MC_Home」，让轴回原点。
 - 执行「Step3」后会触发「MC_MoveVelocity」，当位置达到 300 以上后，会再触发「MC_MoveAbsolute」，将位置转至 0，此流程会重复执行，直到「Step3」接点关闭。
 - 自动化执行一次，「Counter_1」会增加一次。

● 项目建立流程

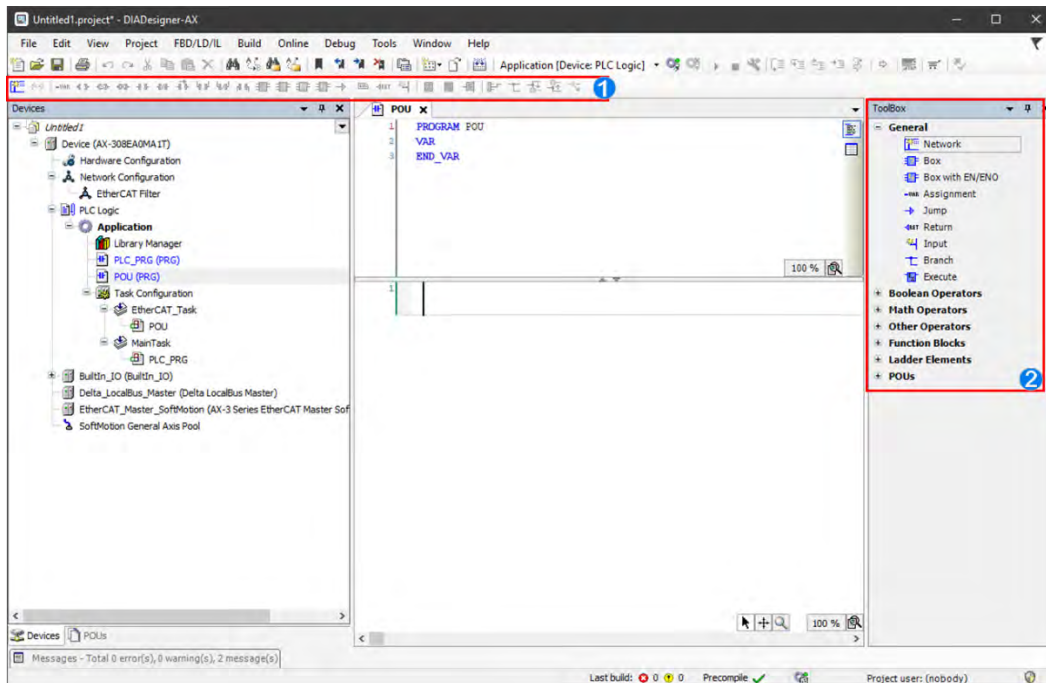
首先将「Motion_PRG」更名，选择「Motion_PRG」>「Refactoring」，在「Refactoring」画面中可将名称改为 POU 后点选「OK」，此时系统会提醒，是否将「Motion_PRG」相关名称改为 POU。



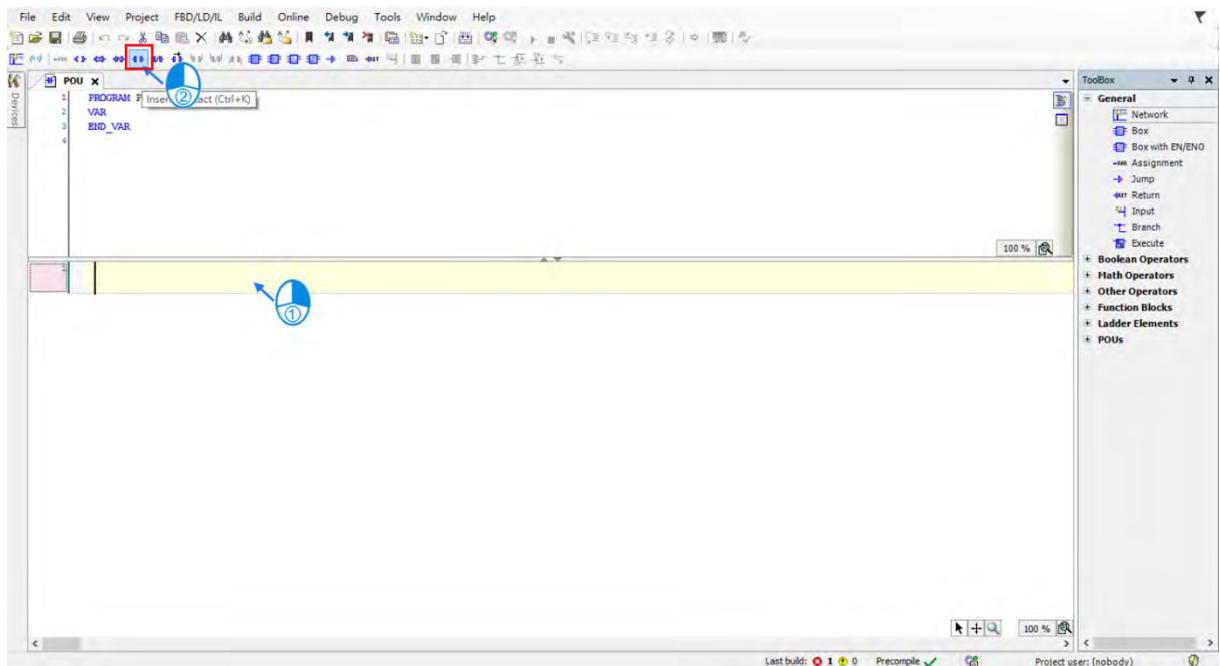
1. 双击改名后的「POU」



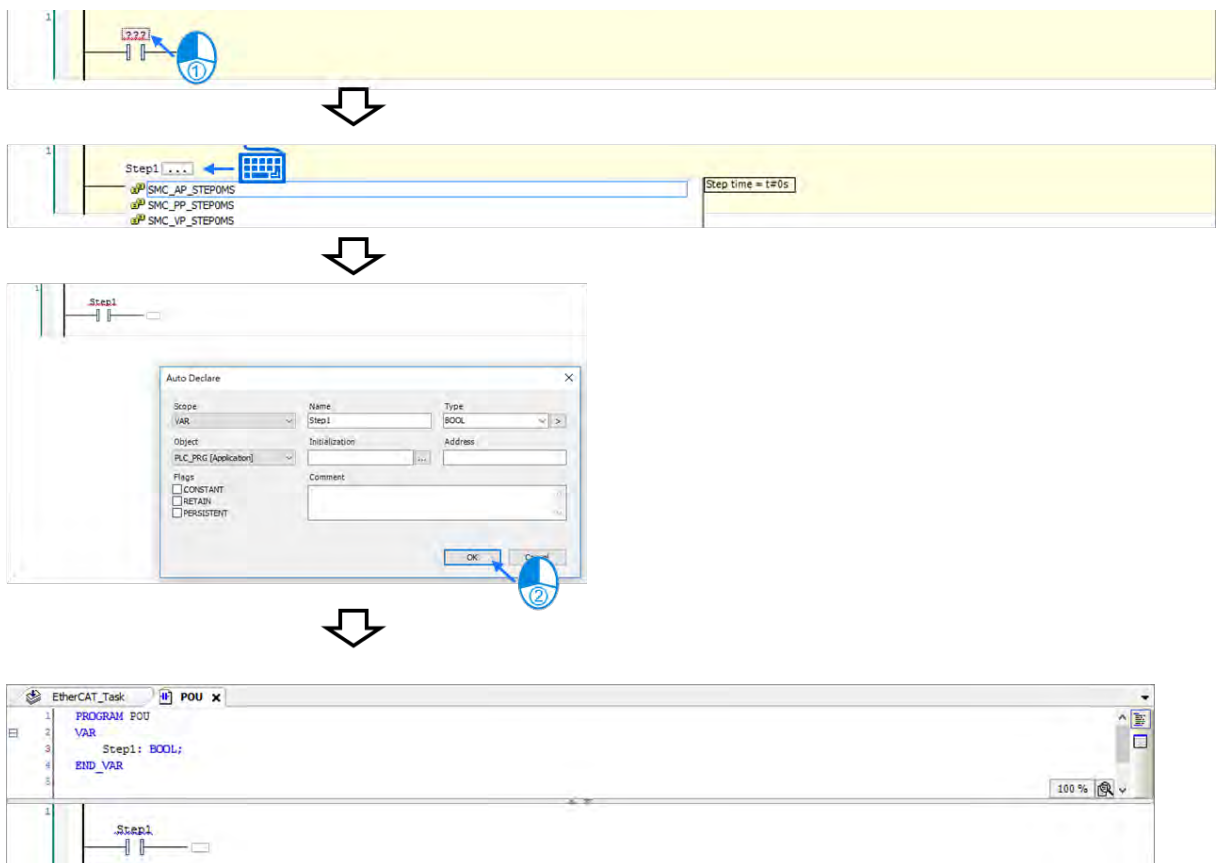
2. 使用 LD 语言编写程序，需新增指令可通过 ① 添加，如需使用功能 (Function) 或者是功能块 (Function Block) 需通过 ② 添加。



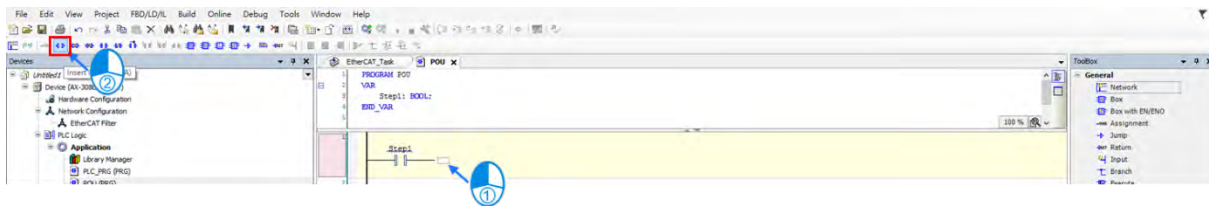
3. 在「POU」中单击欲新增指令的地方后，再单击  即可新增指令。



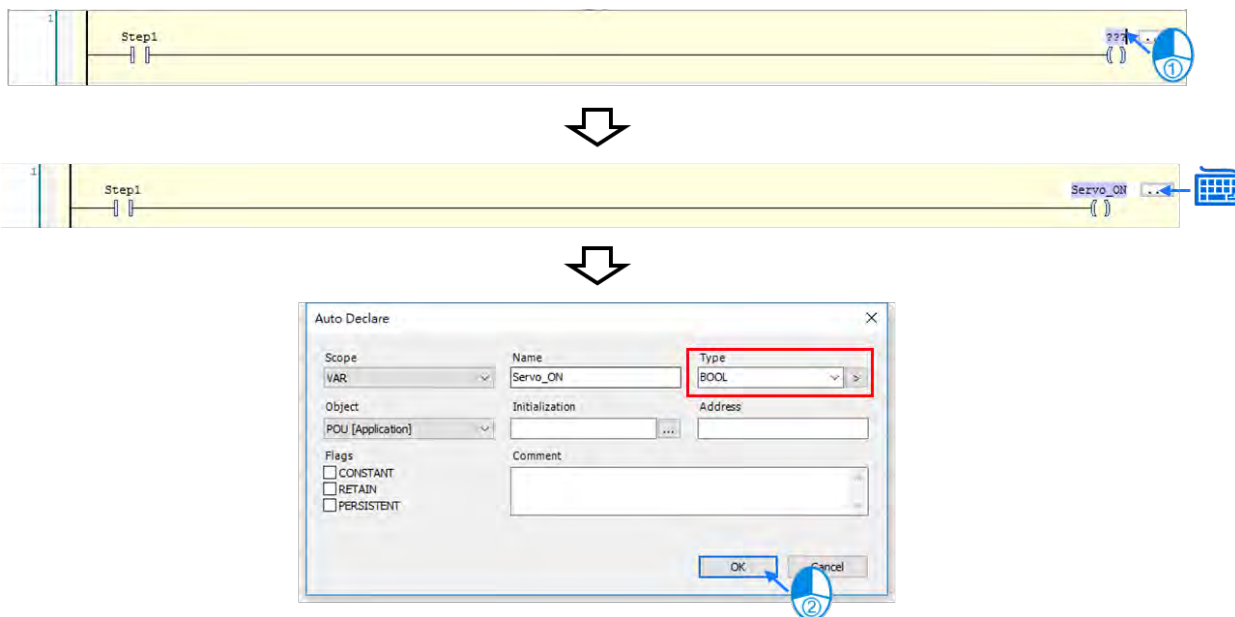
4. 「POU」中出现刚新增的指令后，单击「???」可输入指令变量名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



5. 再来新增输出指令，在「POU」中单击欲新增指令的地方后，再单击  即可新增指令。



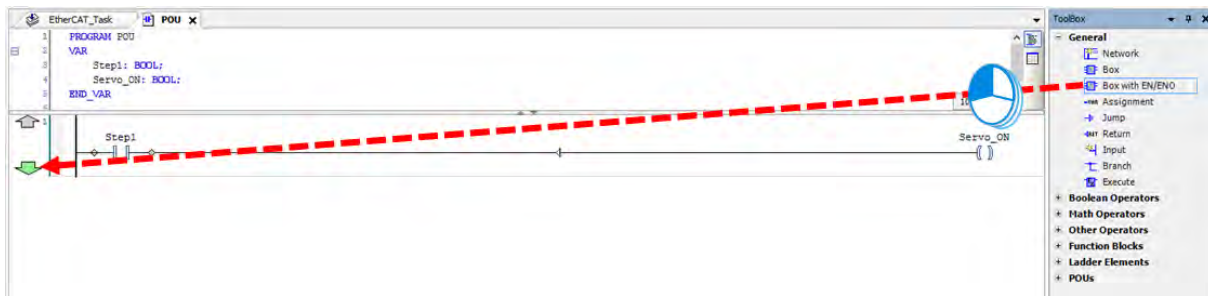
6. 单击「???」后可输入名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



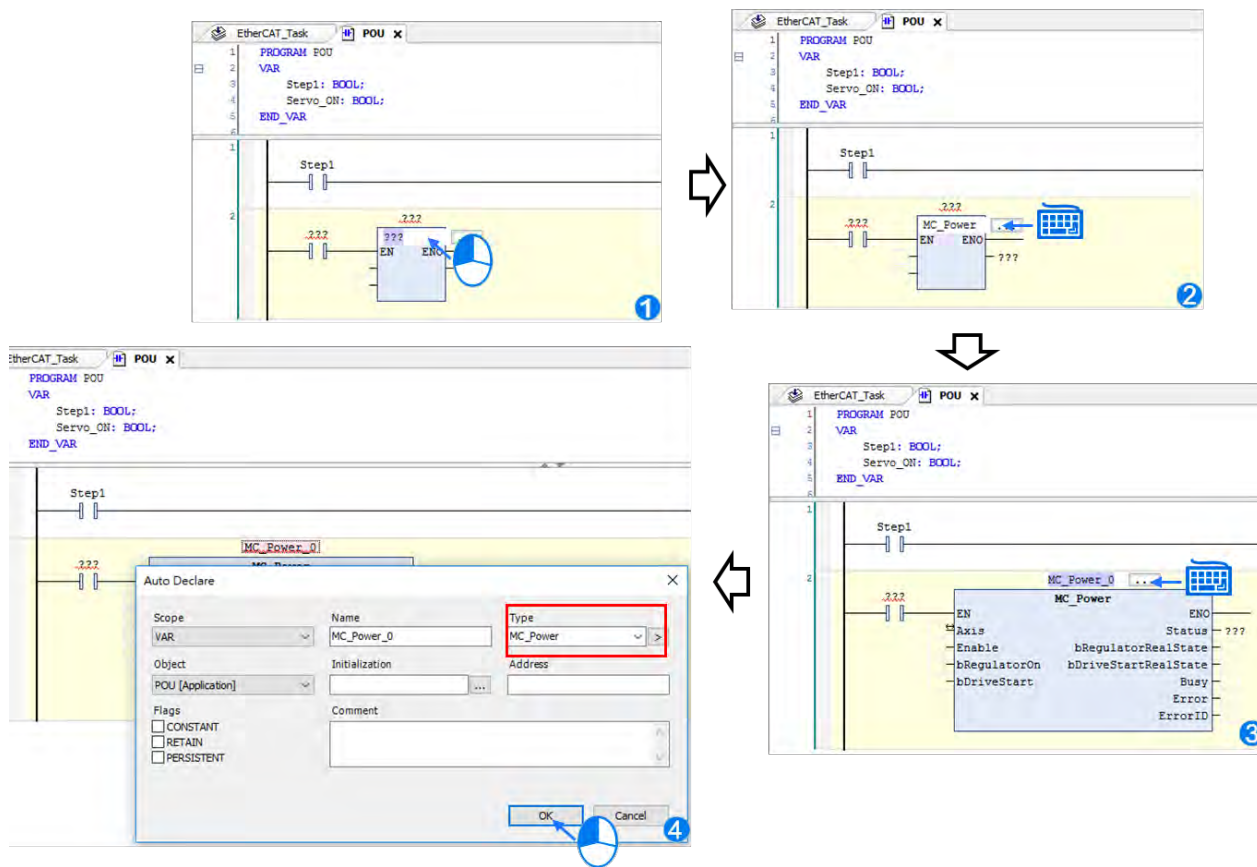
7. 先前新增的指令名称都会上方显示名称与类型，也可在上方直接新增指令名称与类型。(后续范例中使用指令方式都是相同操作)



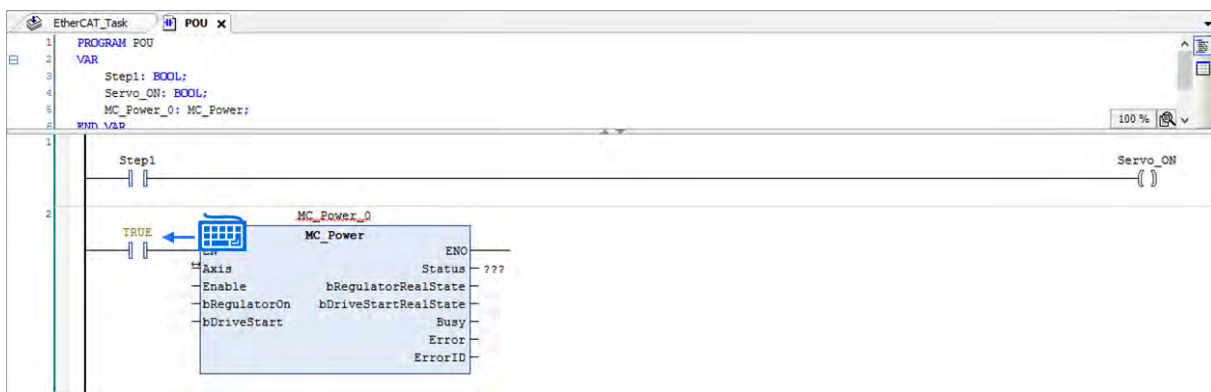
8. 再来新增功能块，单击「Box With EN / ENO」后拉至「POU」中的.



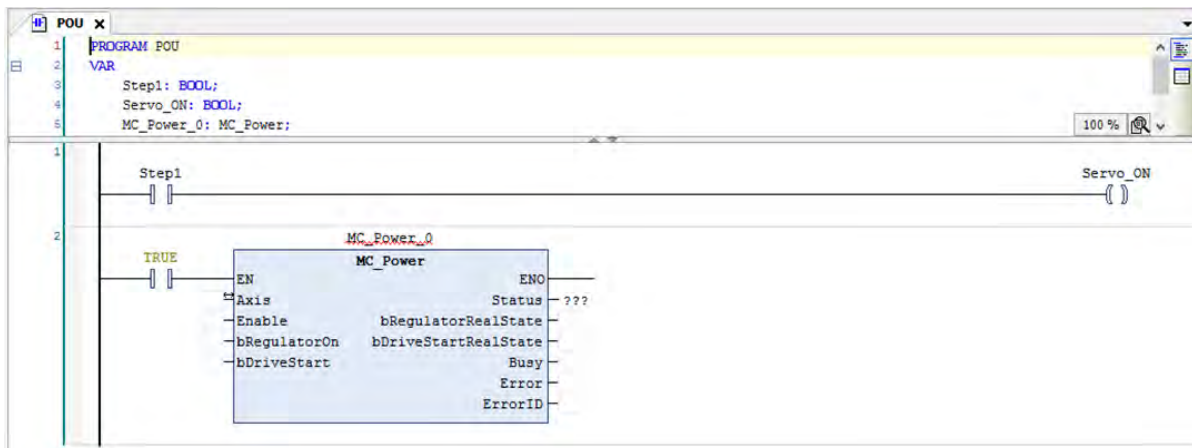
9. 单击「???」后输入功能块名称，输入完成后按下 Enter 键，此时功能块就会出现，再输入功能块名称后按下 Enter 键，就会出现「Auto Declare」，确认功能块「Type」正确后单击「OK」。



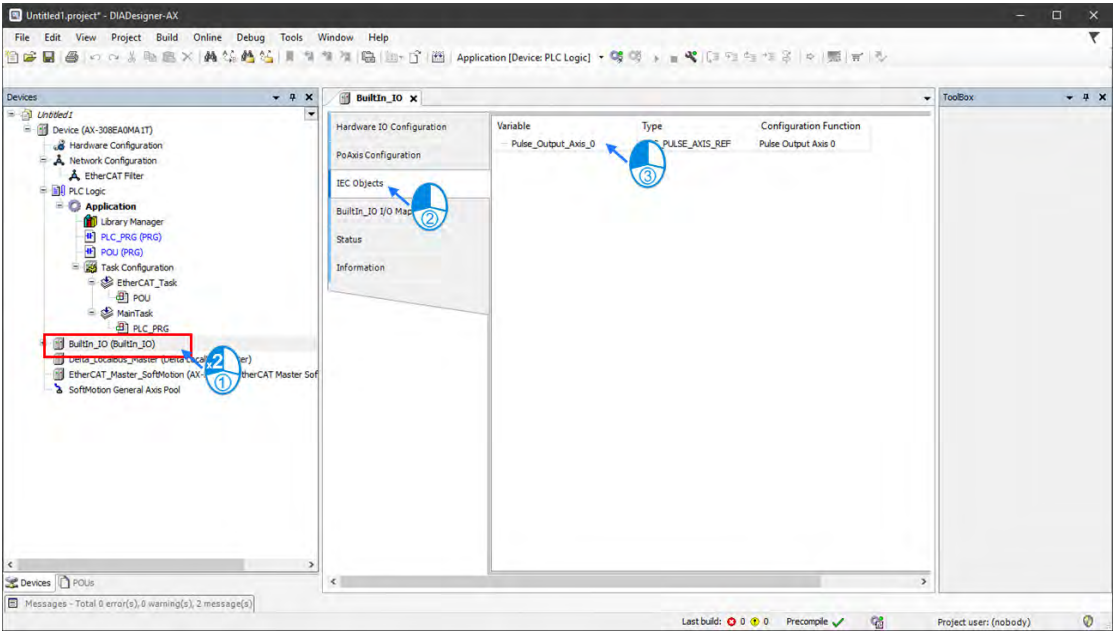
10. 在新增功能块，单击「???」输入 True 后单击 Enter 键。



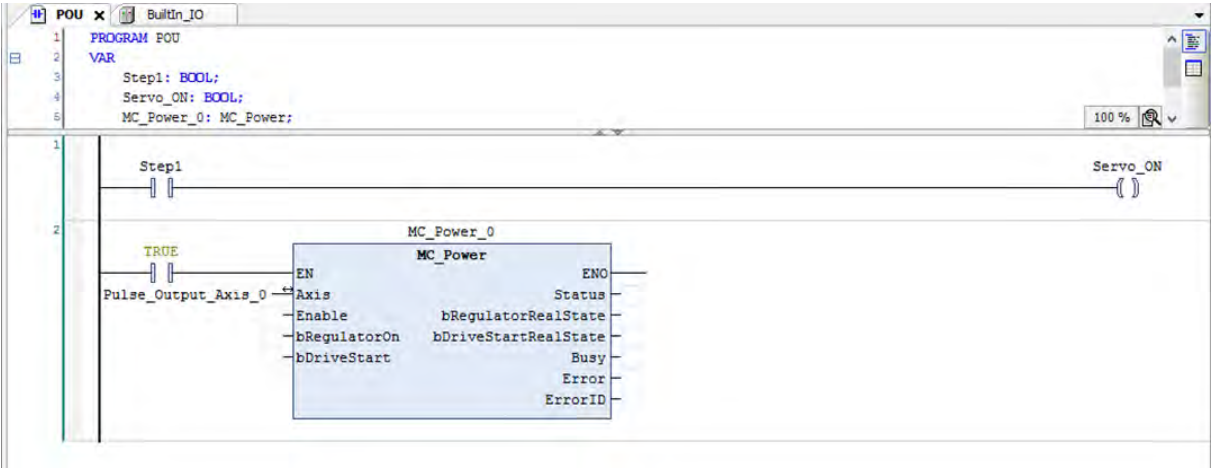
11. 需输入脉冲名称至功能块中的 Axis 引脚。



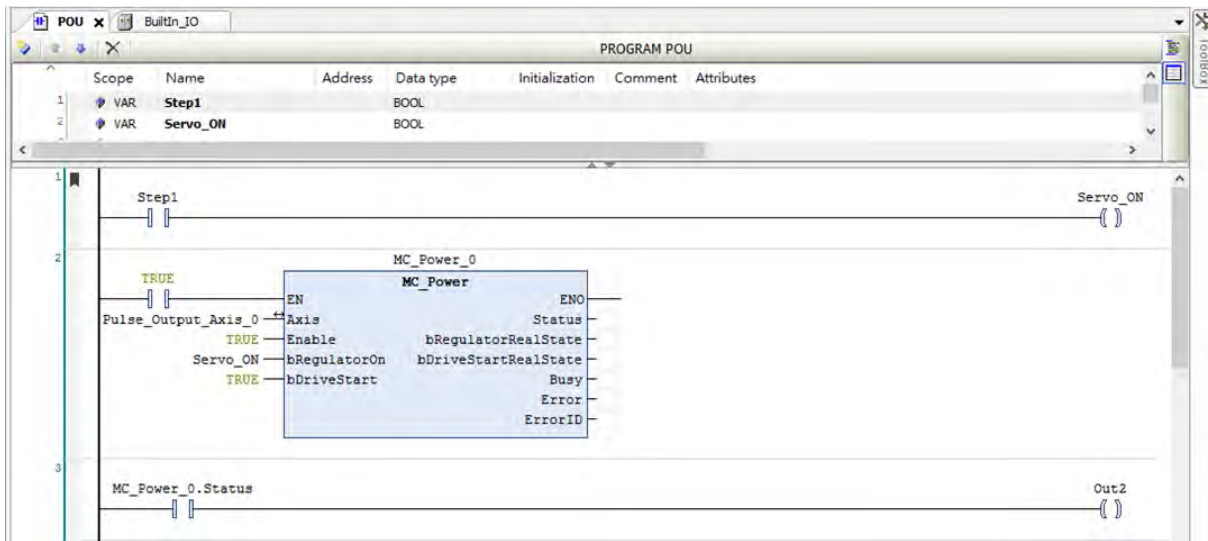
12. 脉冲轴名称需单击「BuiltIn_IO」>「IEC Objects」>「Variable」输入脉冲轴名称「Pulse_Output_Axis_0」，当新增脉冲轴名称都会在此显示。



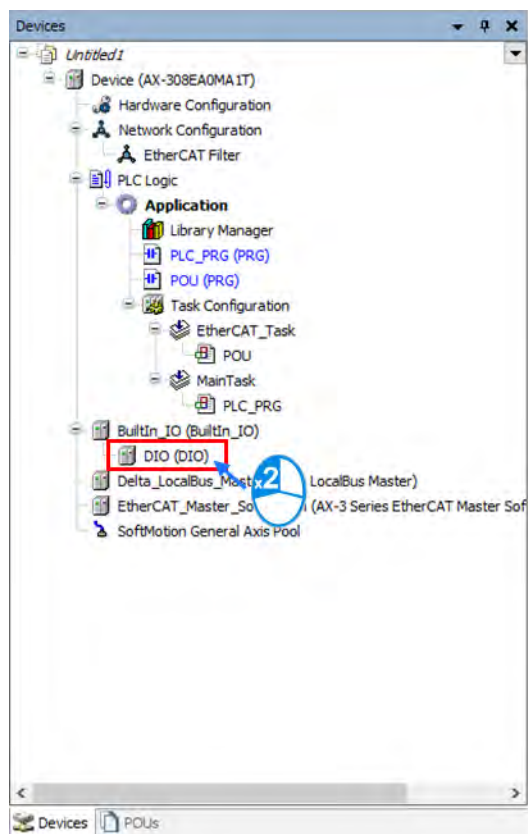
13. 将脉冲轴名称输入至功能块中的 Axis 引脚中并删除引脚中的「???」，因「???」不为任何变数，在编译时会出错。



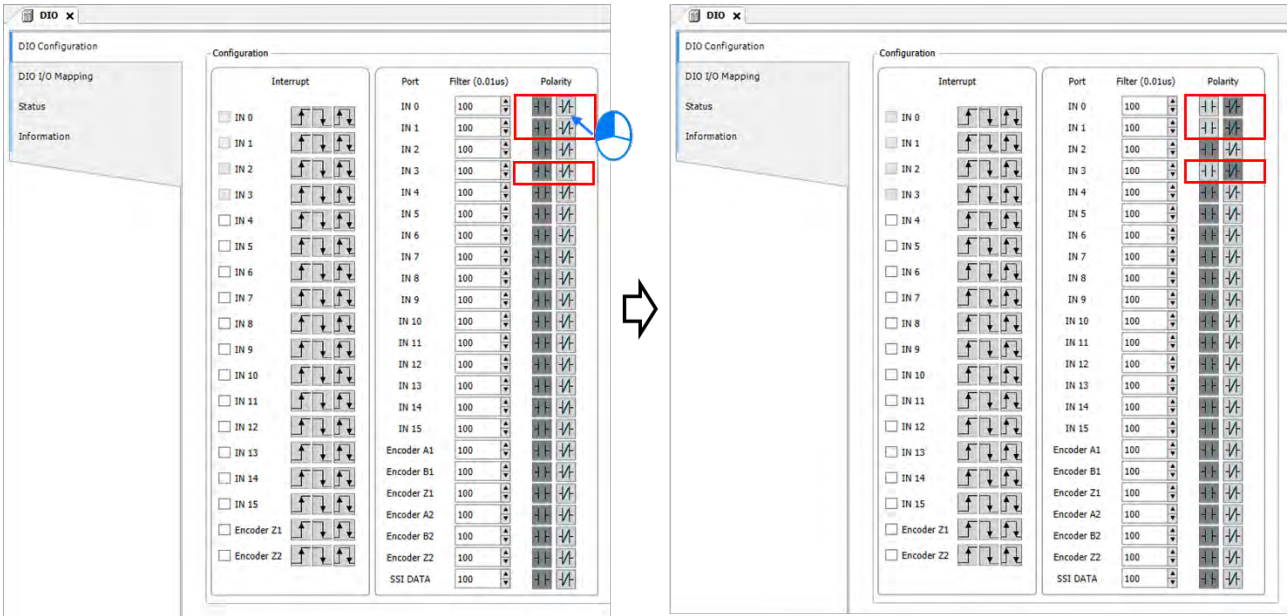
14. 建立功能块的状态输出至 Out2。(Out2 后续还需设定对应至外部硬件位置)



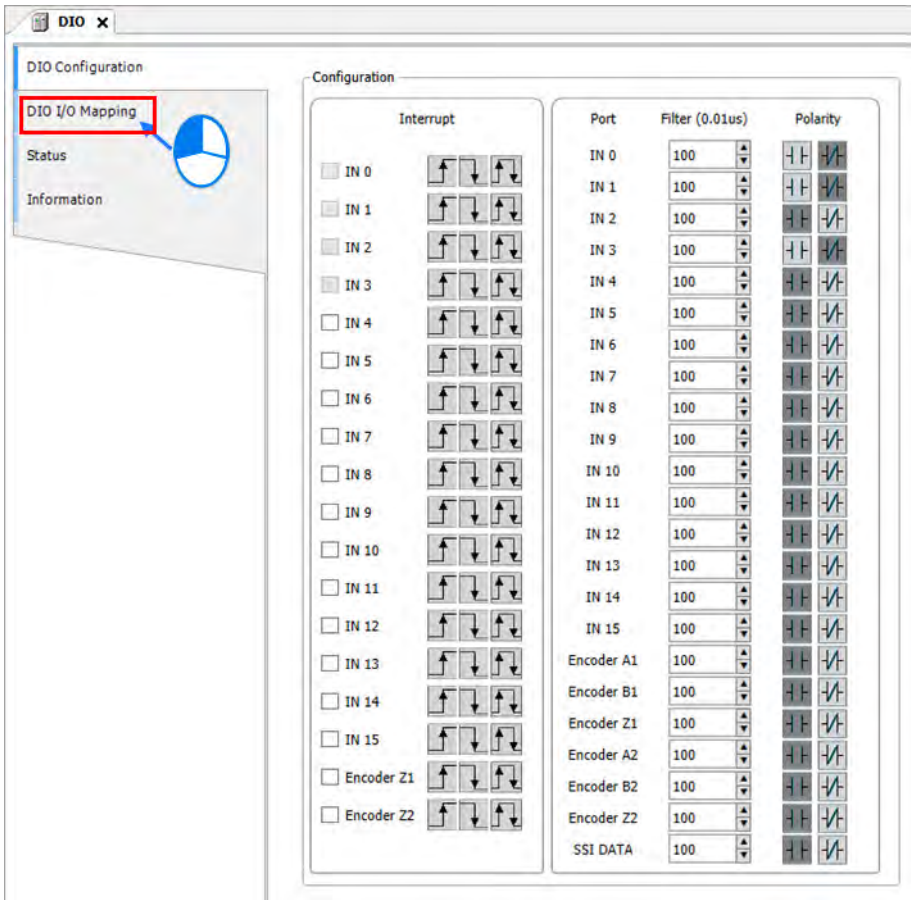
15. 再双击树状中的 DIO。



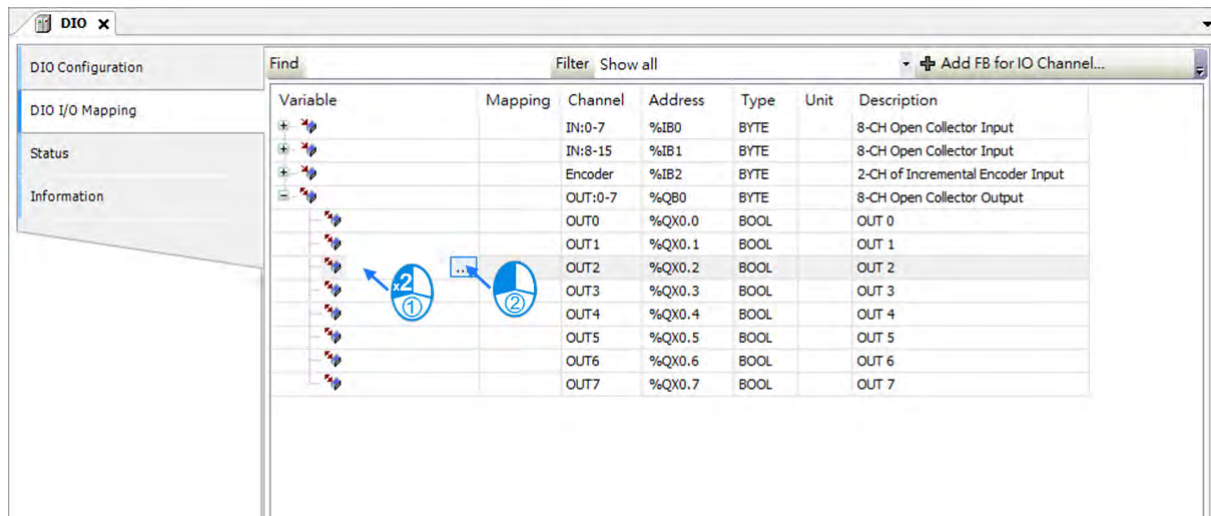
16. 进入「DIO」画面后将 IN0、IN1、IN3 的极性改为，因外部硬件为常闭接点，所以需要更改内部信号极性。



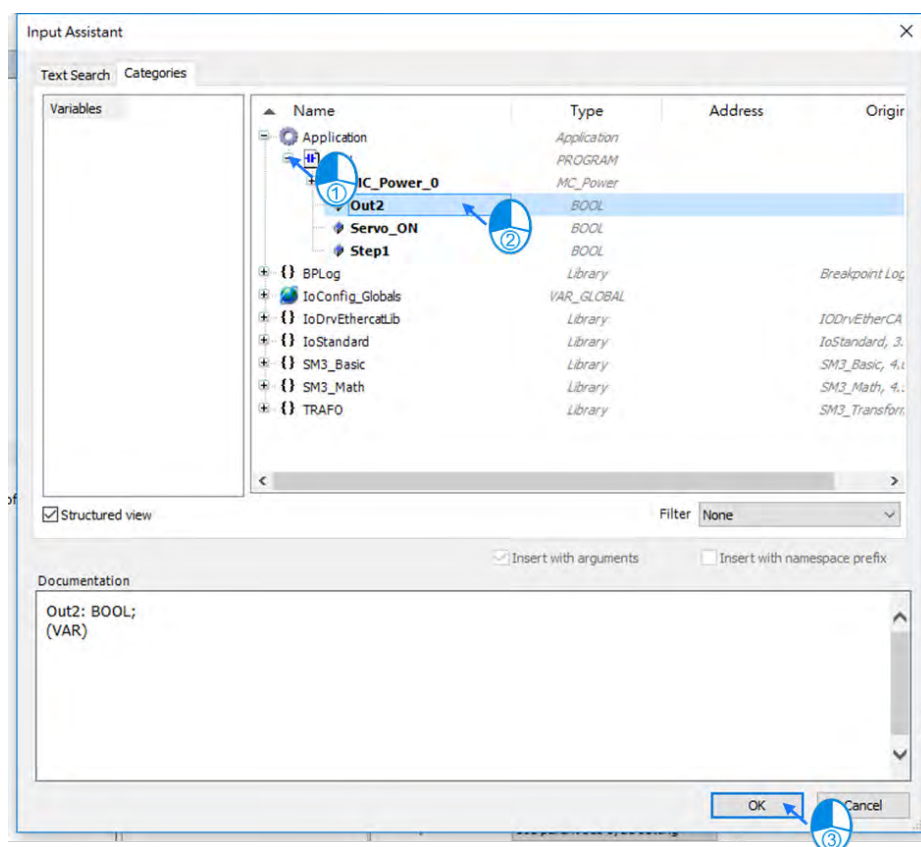
17. 极性设定完毕后，单击「DIO I / O Mapping」。



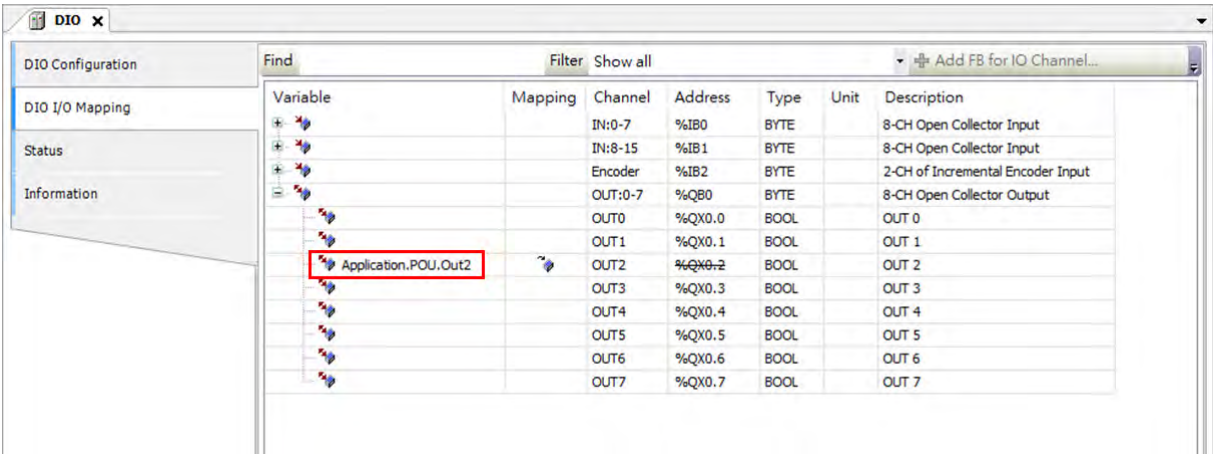
18. 在「DIO I / O Mapping」分页，双击「OUT2」字段后，单击 。



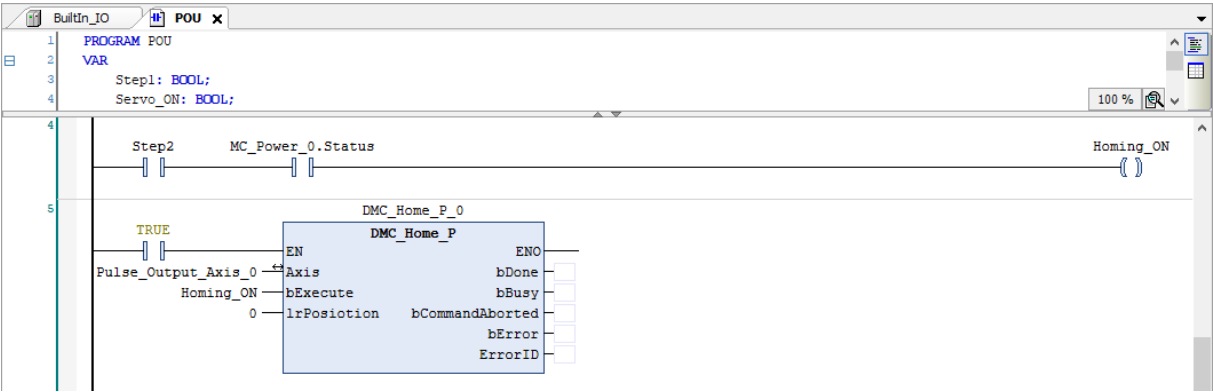
19. 此时需选择「POU」中的「Out2」接点映像至外部硬件位置（先单击  后，选择「Out2」 > 「OK」。）



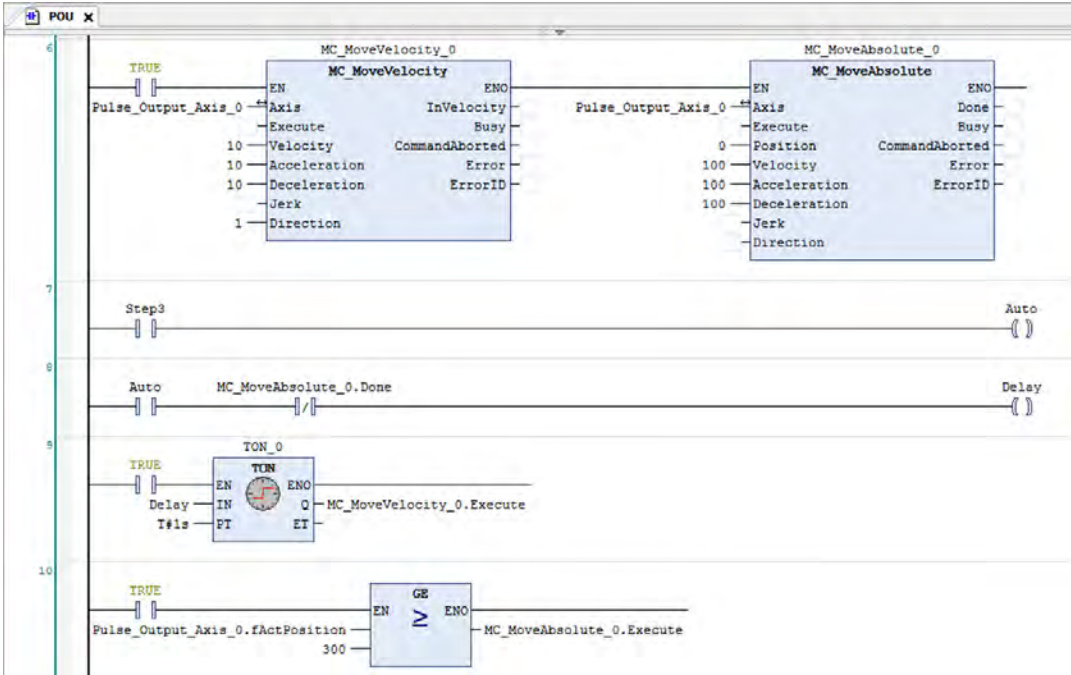
20. 配置完毕后，字段会出现「Application.POU.Out2」，当在 POU 有输出，Out2 就会对应到外部硬件 OUT2 的接点。(如需使用外部硬件信号需在此设定)



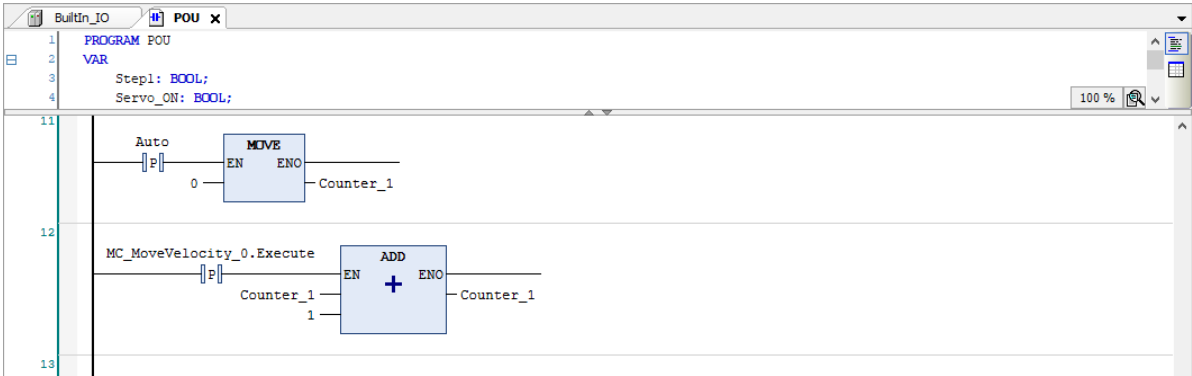
21. 利用上方操作方式建立「DMC_Home_P」，此功能主要为脉冲轴回原点使用。



22. 利用上方操作方式建立「MC_MoveVelocity」功能块与「MC_MoveAbsolute」，启动后延时 1 秒启动。



23. 当执行「Auto」时，会将「Counter_1」数值清除，每次触发就会自动加 1。

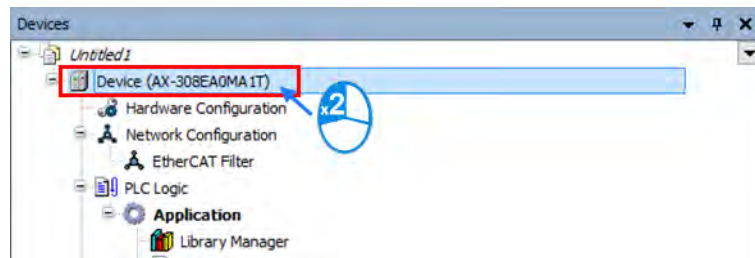


3.3.3 程序监控

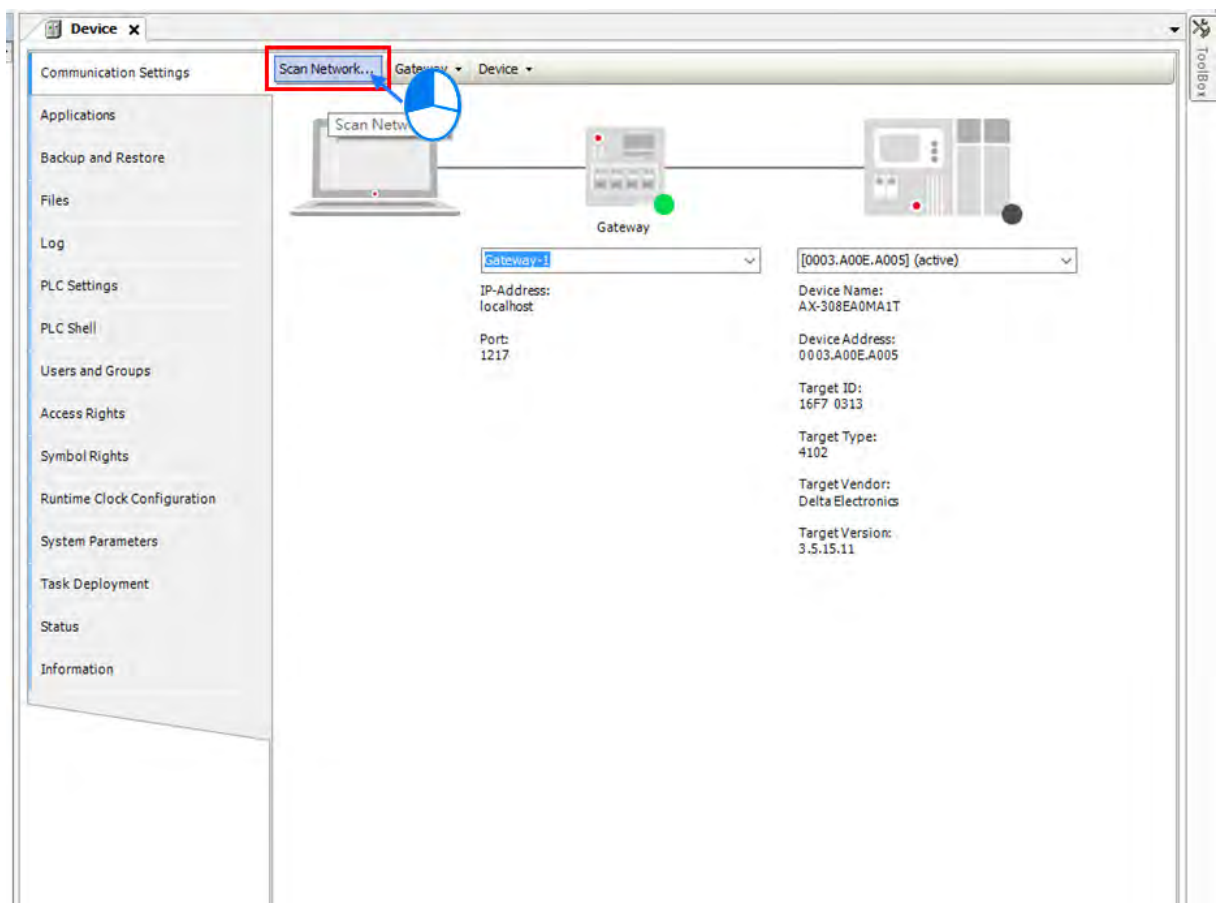
当程序运行的时候，用户可以通过监视的方式，来了解系统目前的逻辑控制状态，或是修改部分装置的数值来进行系统的测试，现在我们就来进行程序监视的操作。

3.3.3.1 设备与计算机链接设定

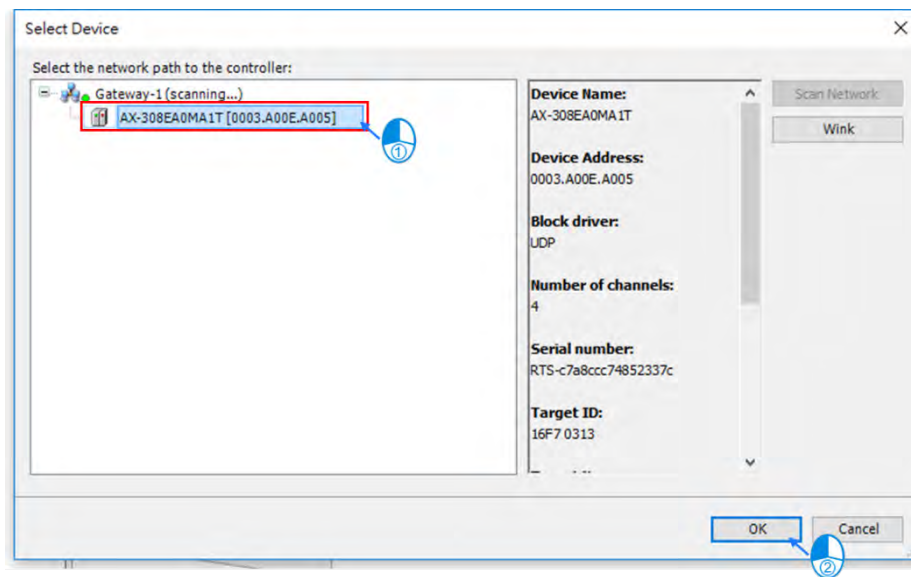
1. 单击「Device」后双击鼠标左键（AX-308E 预设 IP 为 192.168.1.5）



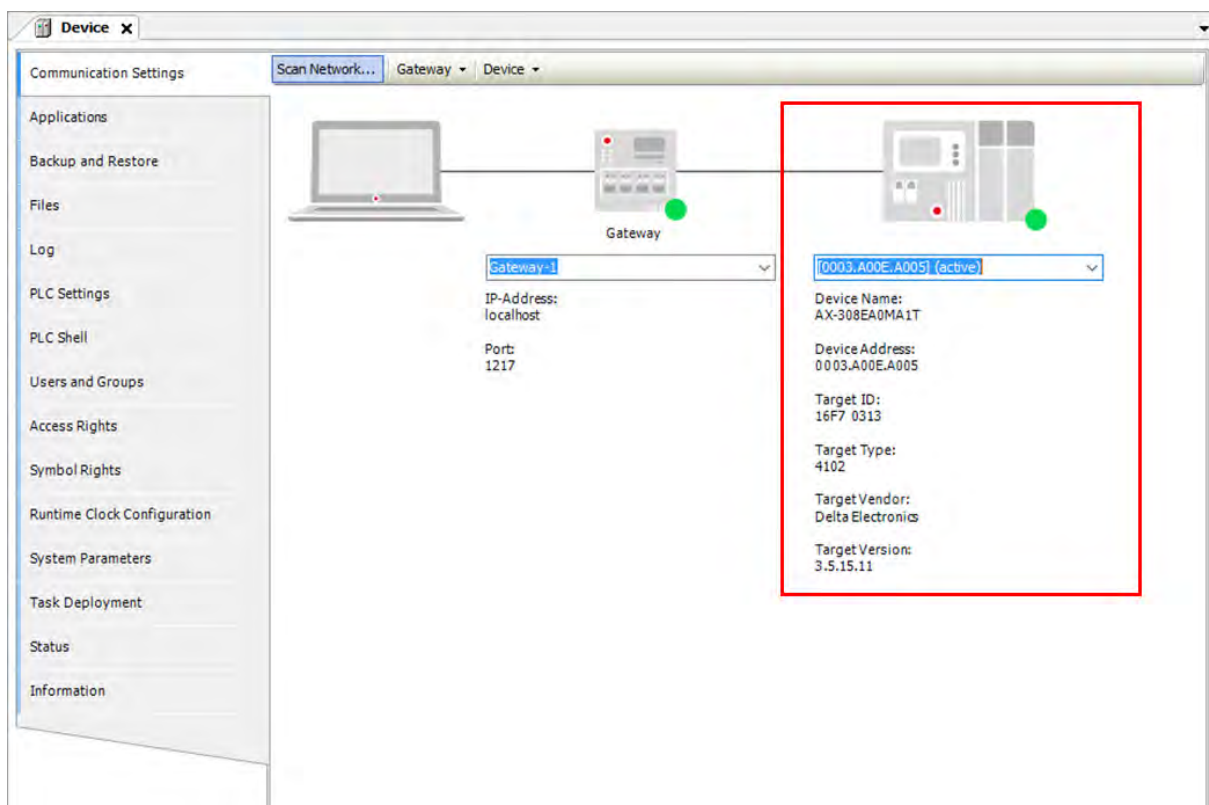
2. 进入「Device」画面后，选择「Scan Network」。



3. 在「Select Device」画面中，单击「AX-308E」 > 「OK」。

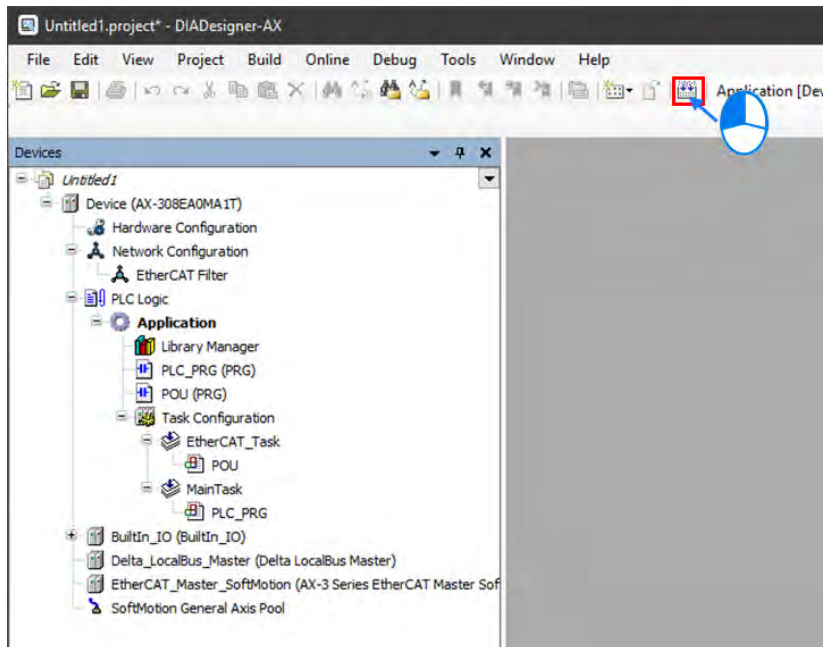


4. 当 PC 与设备连结后，会出现设备相关的信息，如下红框所示。

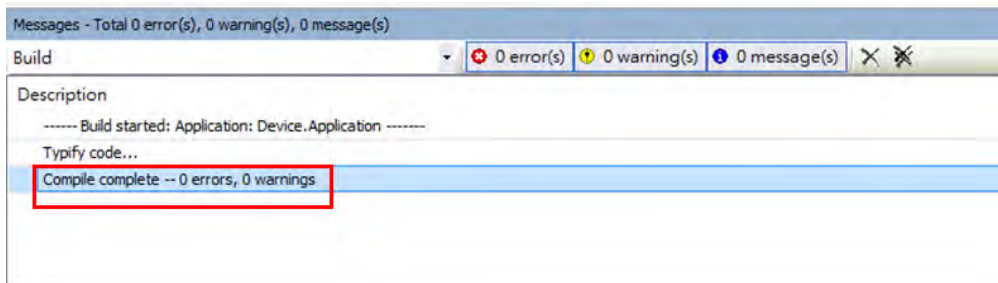


3.3.3.2 程序监控流程

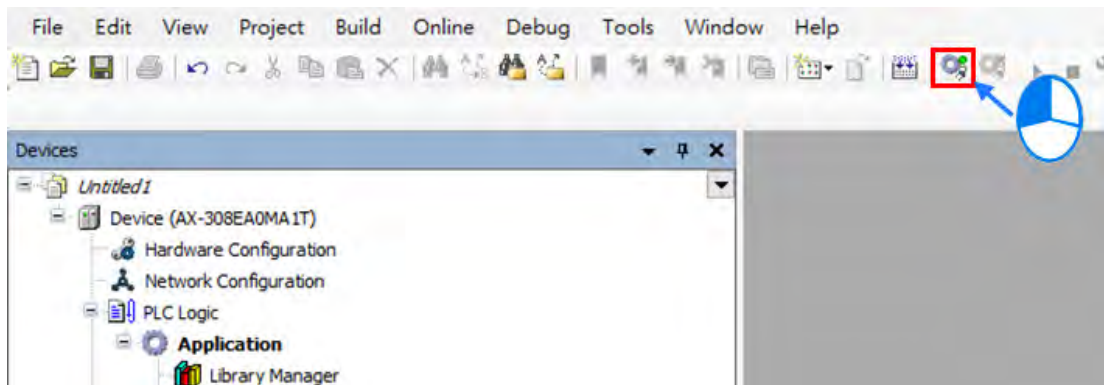
1. 监控前先单击  编译以确认程序是否正确。



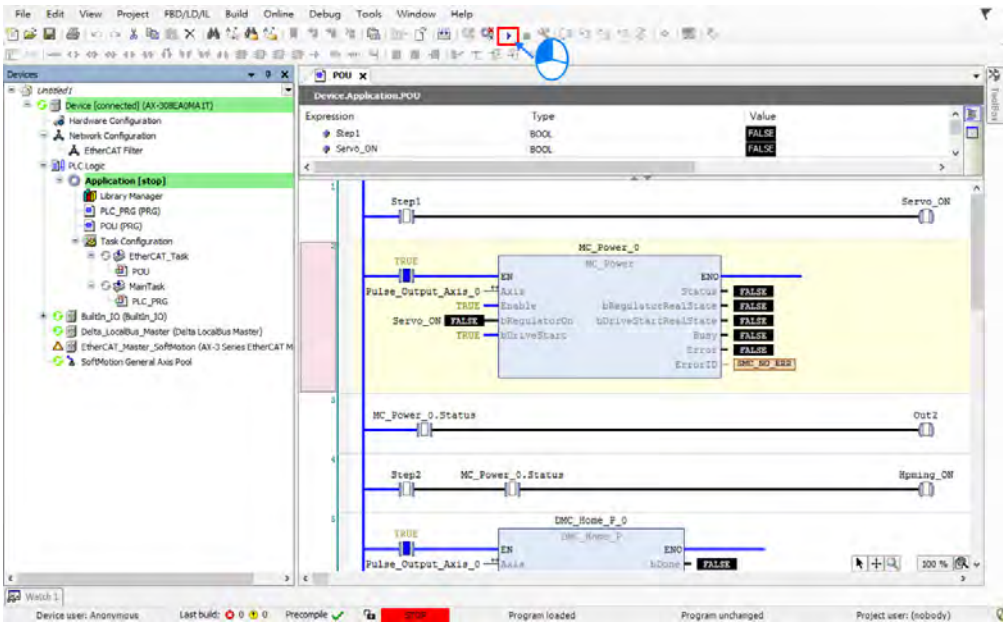
2. 编译完成后，可确认程序编写是否有错误。



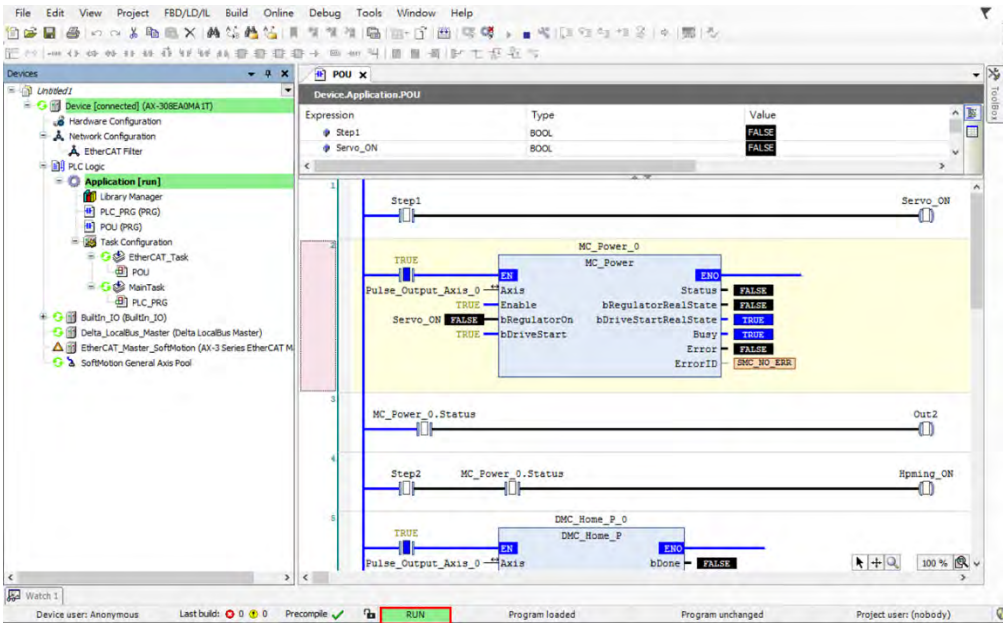
3. 完成后，再单击  执行在线监控。



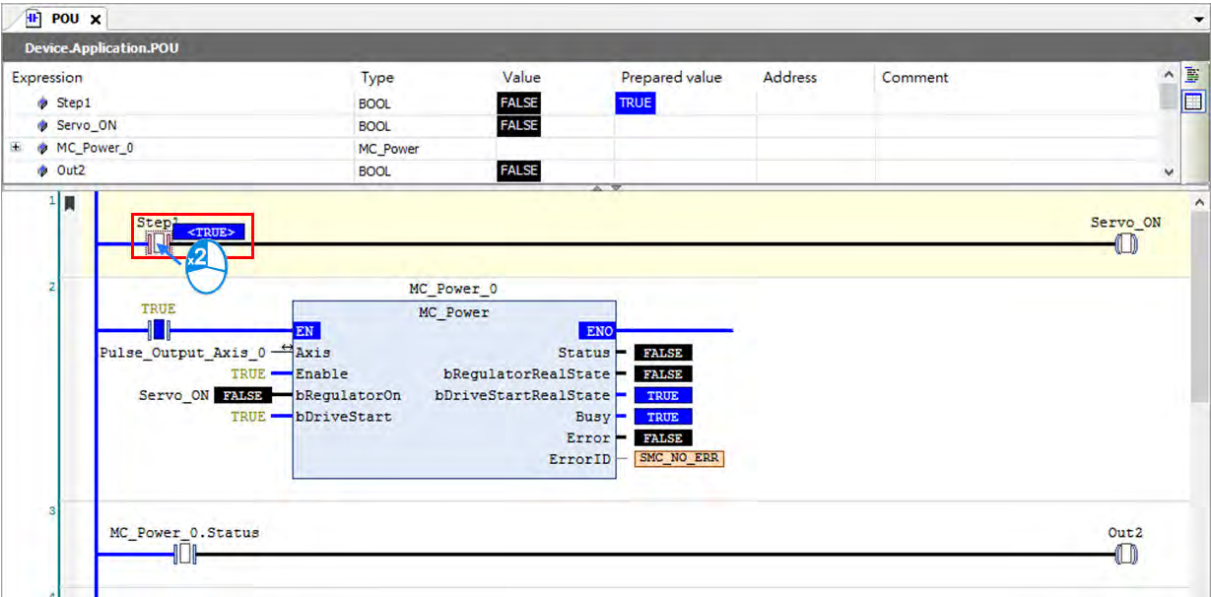
4. 执行在线监控后需下载程序，下载完成后 PLC 状态会 Stop，需单击 ▶ PLC 才会执行 Run。



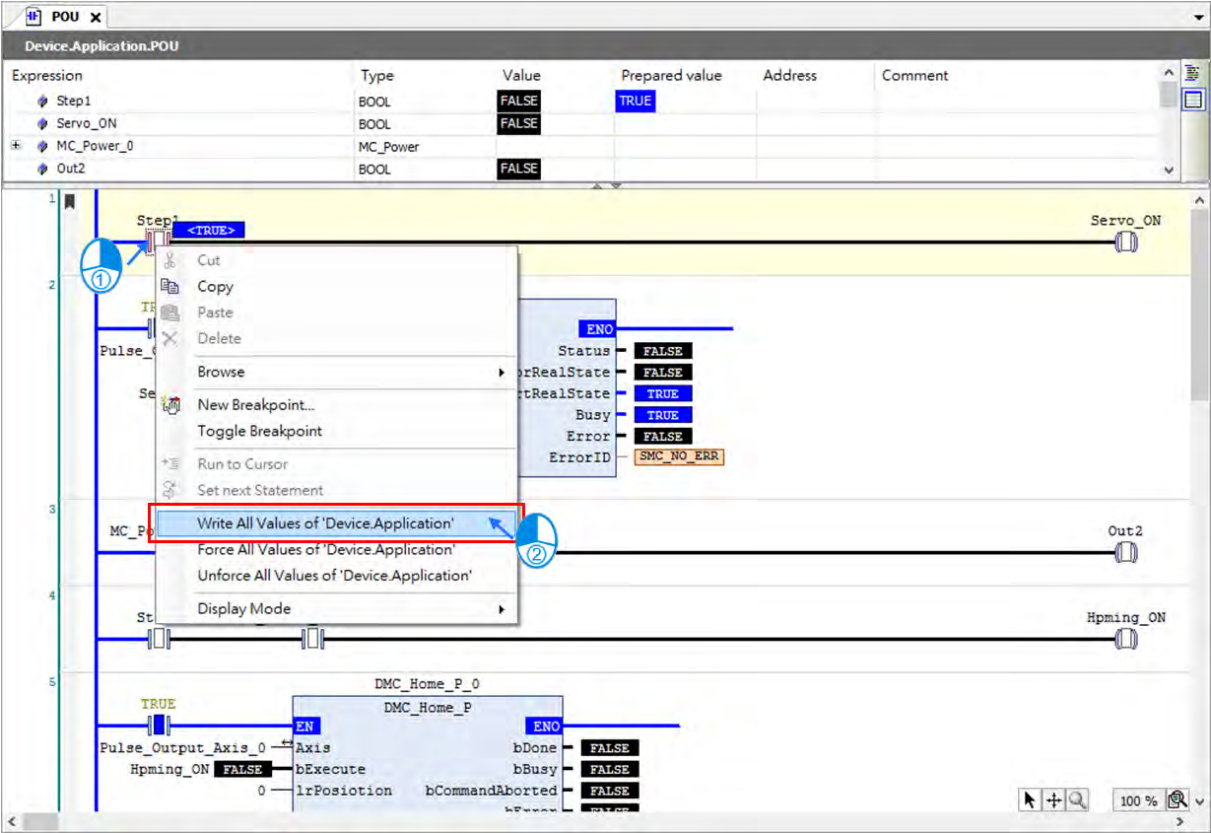
5. 当 PLC 开始 Run 时，下方的状态会改为 **RUN**，此时可开始执行程序。



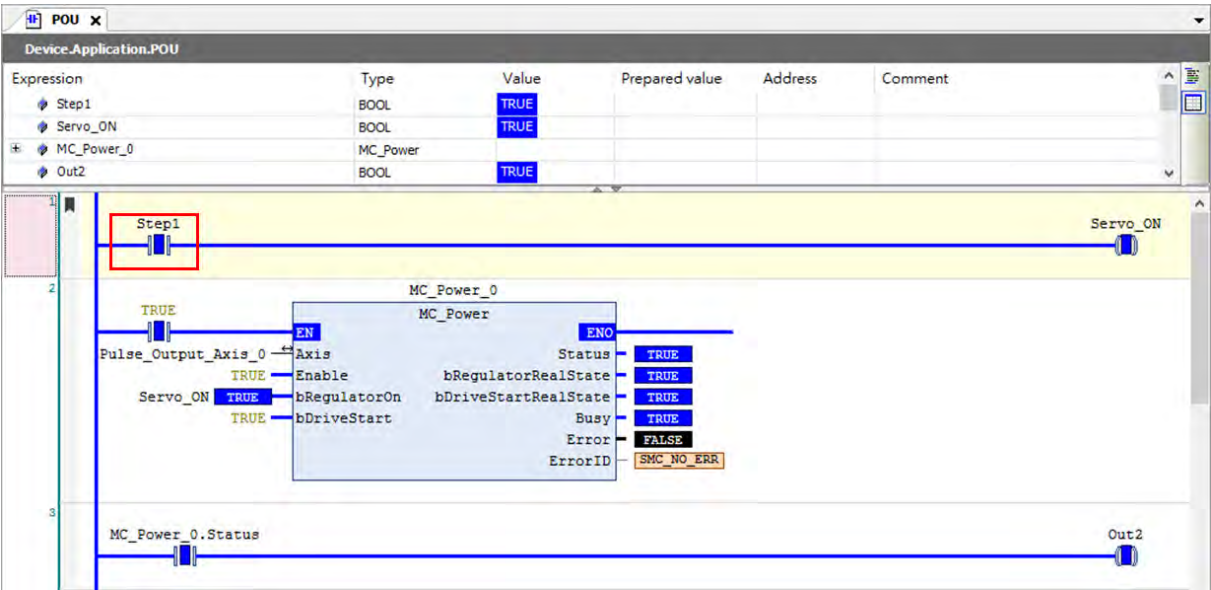
6. 在线监控后，双击「Step1」后会出现<TRUE>的图示。



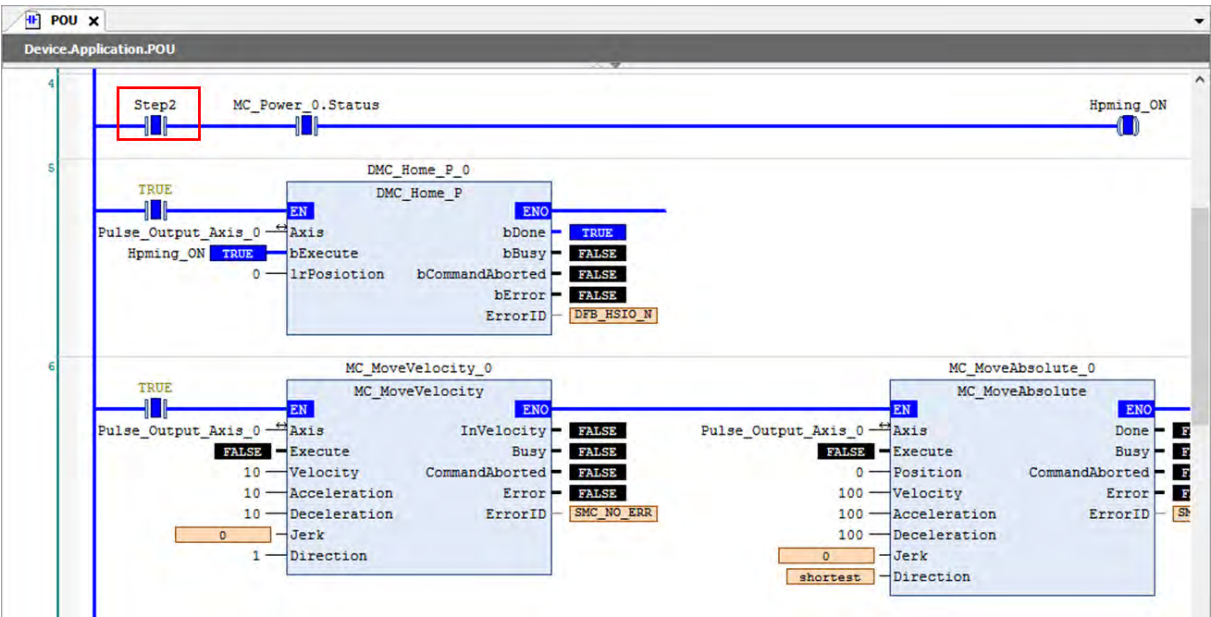
7. 单击「Step1」后右键单击，选择「Write All Values of "Device.Application")」，会写入程序中所有状态。（可使用快捷键 Ctrl + F7）



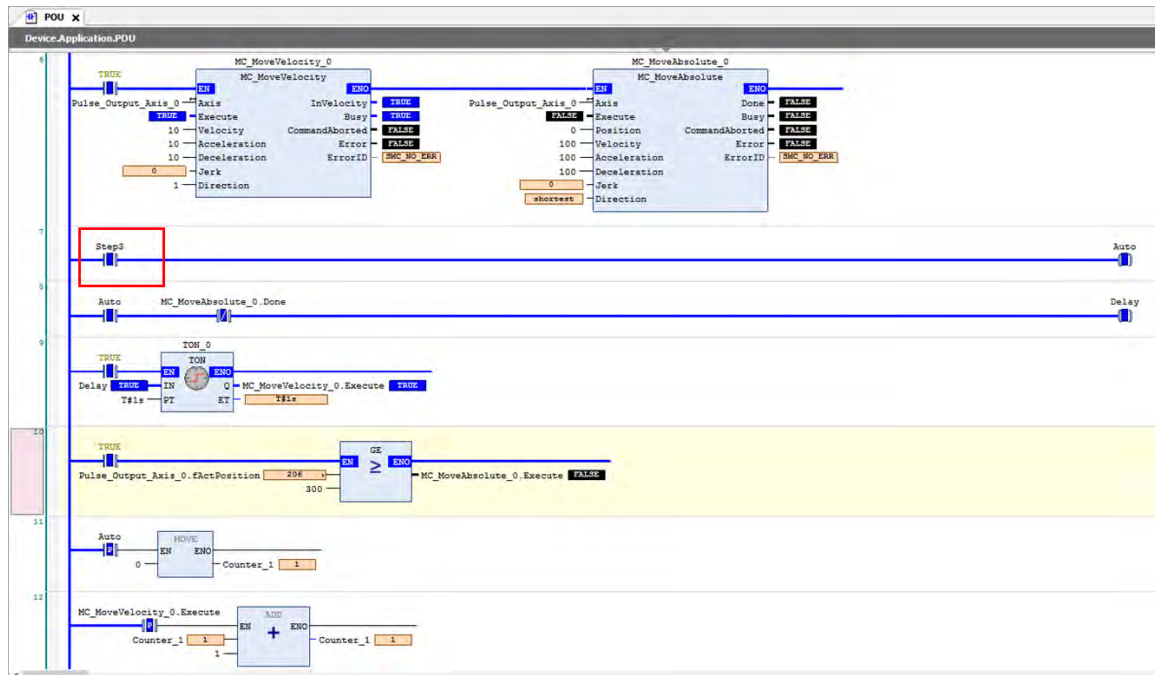
8. 启动「Step1」后，就会 Servo ON。(以相同操作执行以下程序)



9. 启动「Step2」后使轴回原点

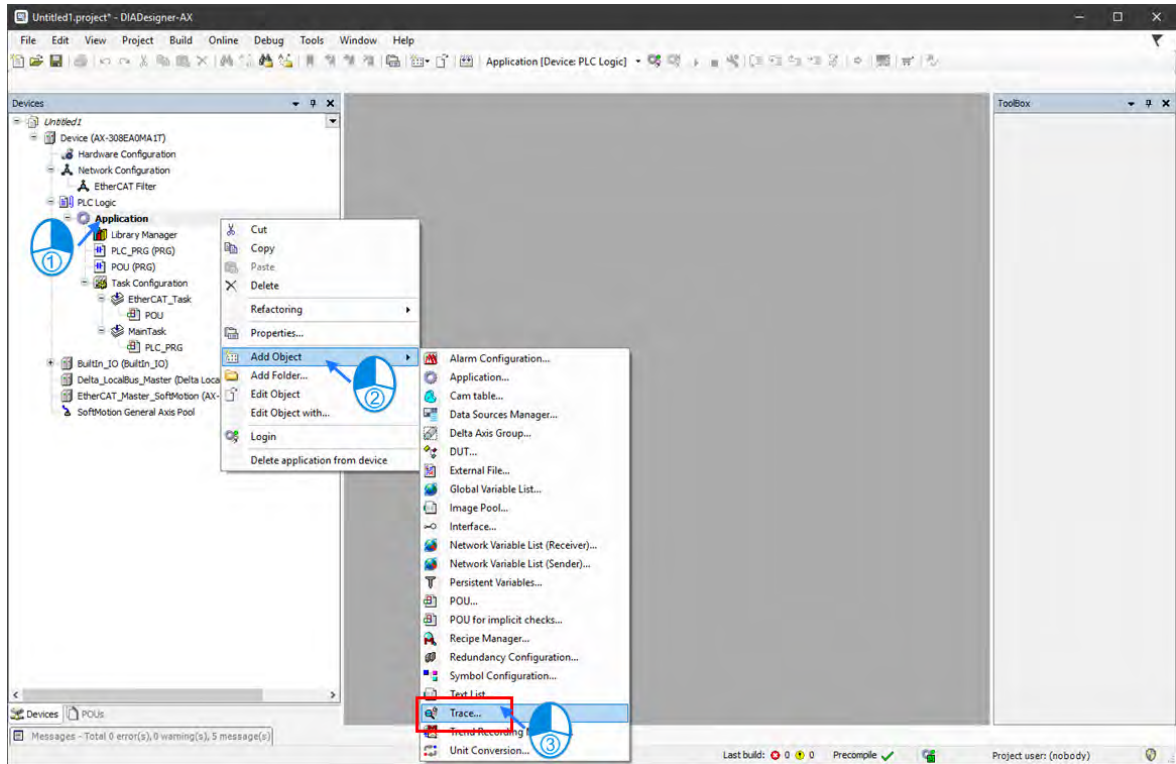


10. 启动「Step3」后，轴会先以「MC_MoveVelocity」等速运转至 300mm 后，再执行「MC_Absolute」，反转至 0 的位置，每旋转一次，「Counter_1」就会+1，直到「Step3」关闭。

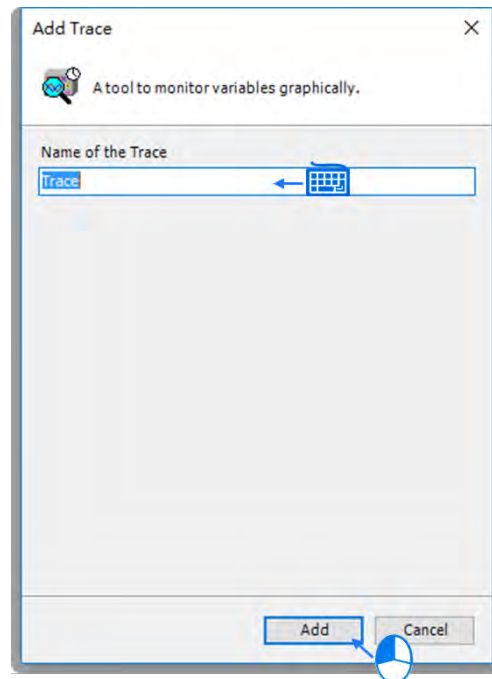


3.3.3.3 示波器监控

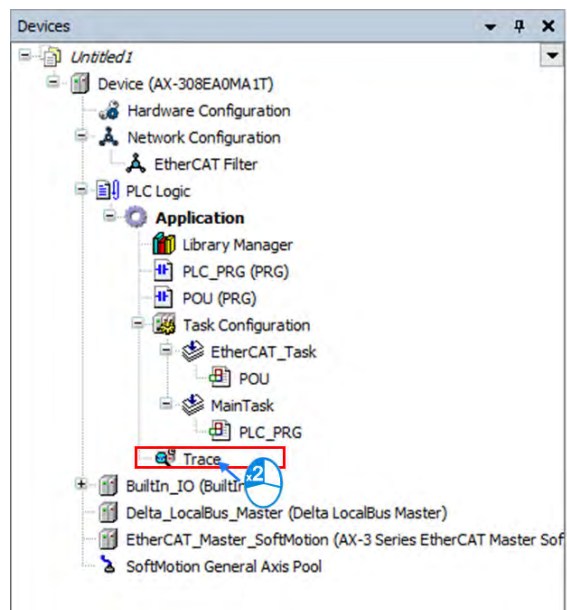
1. 当程序在运行过程中可以使用 Trace 监控的方式，确认引脚状态及轴状态。依序单击「Application」>「Add Object」>「Trace」。



2. 进入「Add Trace」画面中，输入 Trace 的名称后再单击「Add」。



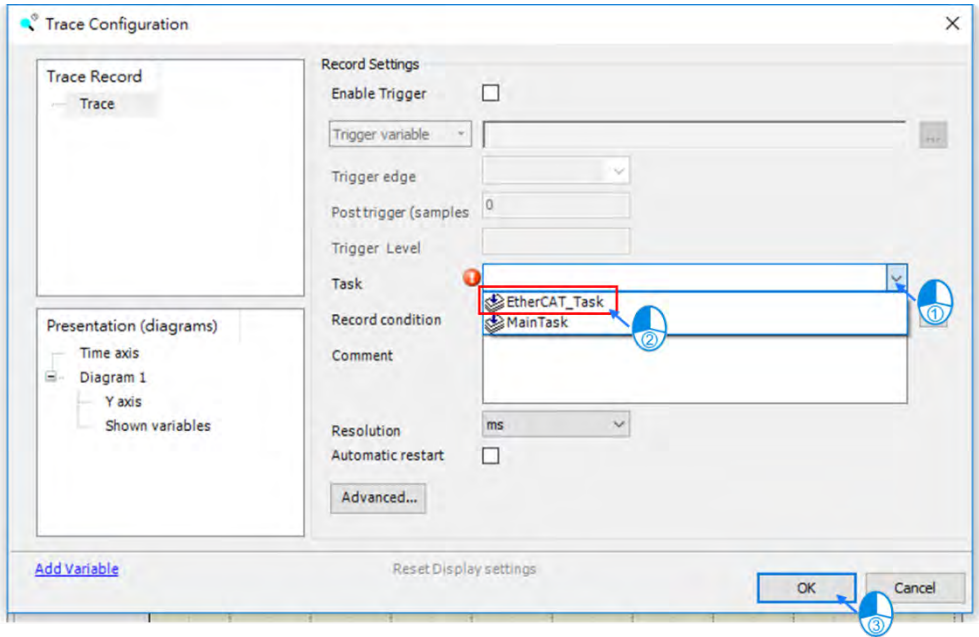
3. 新增完毕后左边树形图会出现刚刚所新增的示波器，此时双击「Trace」。



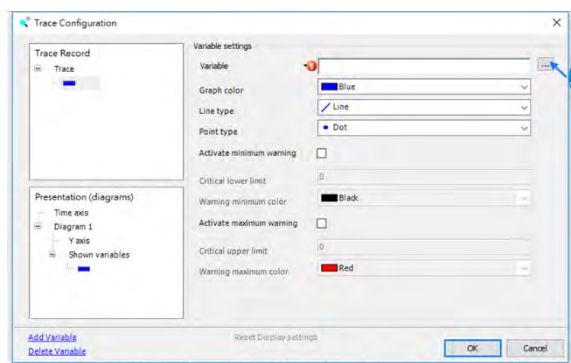
4. 进入「Trace」画面后，单击「Configuration」。



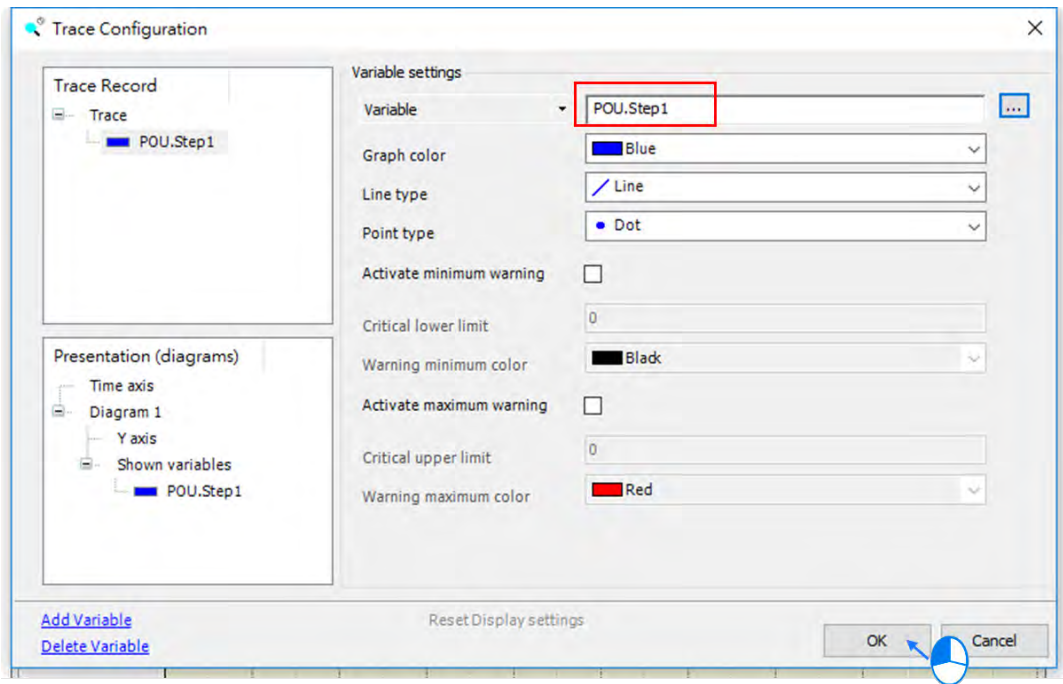
5. 在「Trace」设定画面中，选择「Task」下拉选单中的「EtherCAT_Task」。



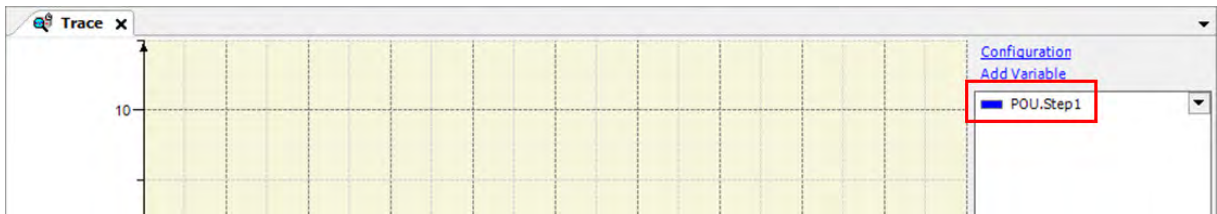
7. 进设定画面后，单击添加需监控的参数，当进入添加画面后，单击后选择「Step1」>「OK」。



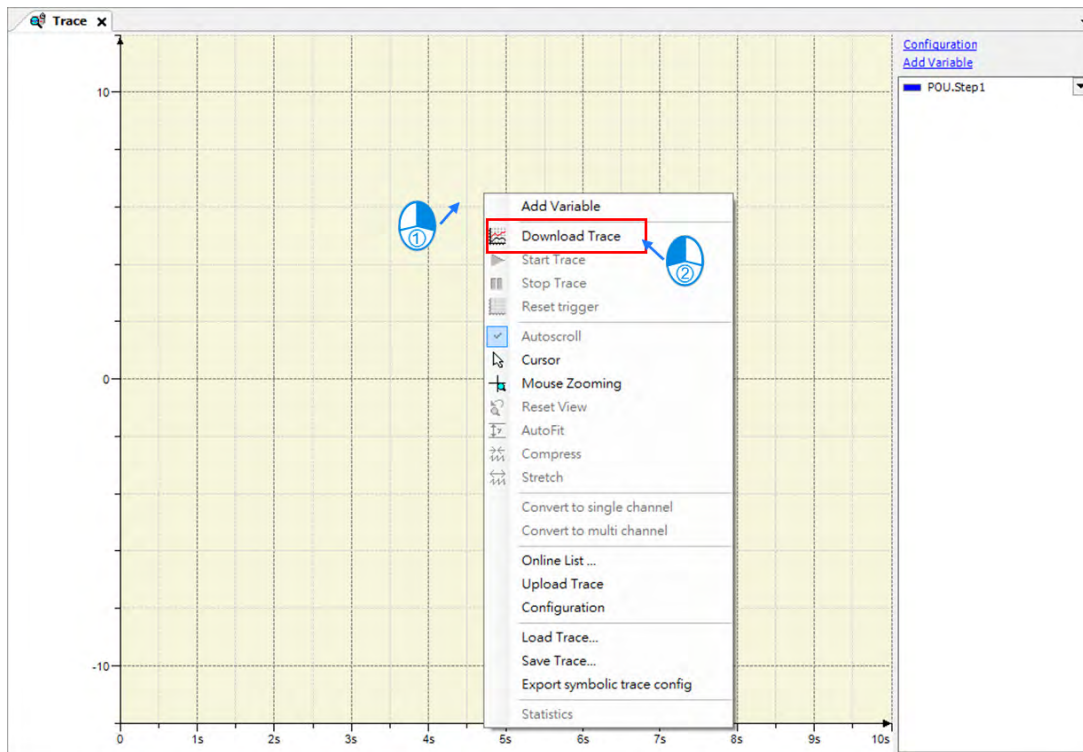
8. 选择好需监控的参数会出现在表格中再单击「OK」。



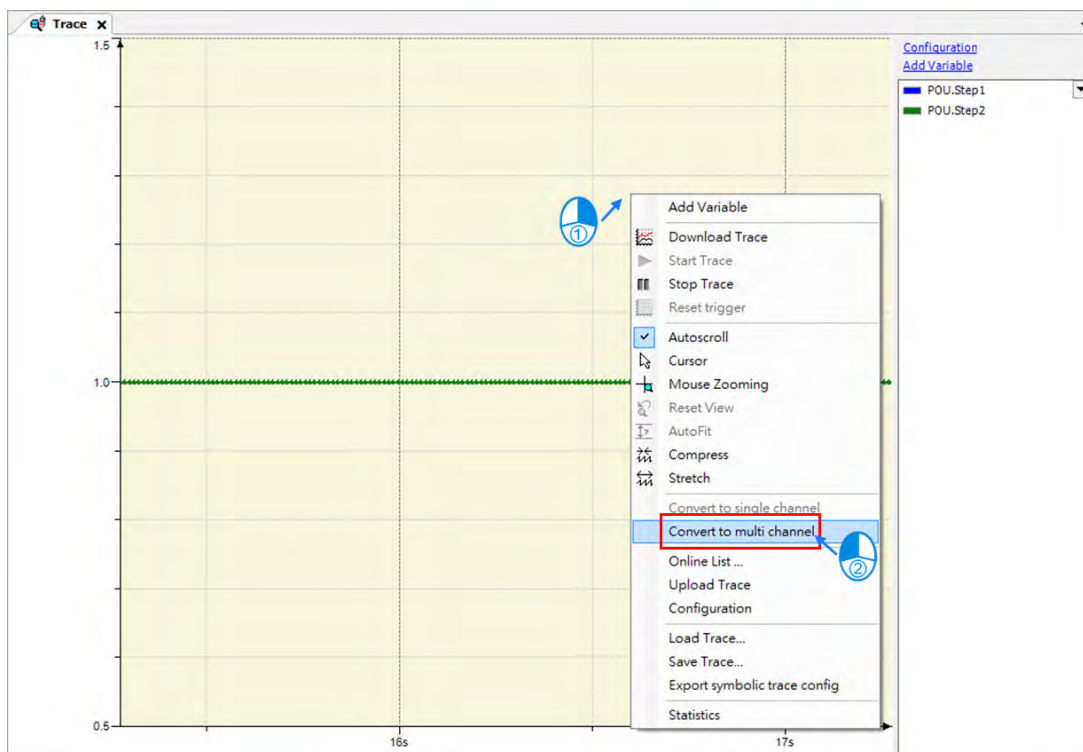
9. 添加完成后，在「Trace」画面右方会出现需监控的参数。



10. 在在线监控时单击「Trace」右键后，选择「Download Trace」就会开始监控。



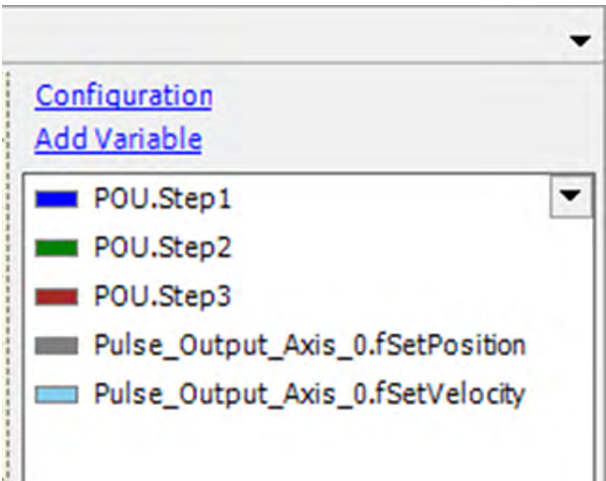
11. 当有两个参数需要监控时，可以在「Trace」画面单击右键，选择「Convert to multi channel」。



12. 「Convert to multi channel」选择后会以两格的方式显示。



13. 利用上方相同操作新增下列监控值。



14. 通过程序执行过程中，可使用示波器观察触发的开关及轴目前位置及速度。



示波器流程介绍：

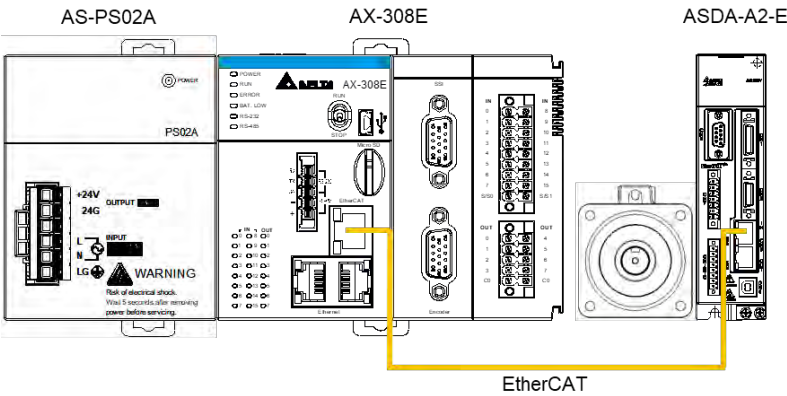
- ①启动「Step1」让脉冲轴启动。
- ②启动「Step2」让脉冲轴回原点。
- ③启动「Step3」后，等待 1 秒后，触发「MC_MoveVelocity」正向旋转。
- ④当脉冲轴旋转位置超过 300 后，会在触发「MC_MoveAbsolute」中断前一个功能块，将脉冲轴位置反转至 0 点。

3.4 虚轴作为同步主轴项目建立

此范例是虚轴为主轴搭配伺服电机追随动作。

3.4.1 系统硬件配置

AX-308 通过 EtherCAT 端口口通讯连接 1 台伺服驱动器 (ASDA-A2-E)，控制伺服电机定位跟随虚轴。



● 材料与工具

编号	名称	料号	数量
1	AS 系列电源模块	AS-PS02A	1
2	AX3 系列主机	AX-308EA0MA1T	1
3	电源供应器	DVP-PS02	1
4	通讯型伺服驱动器	ASDA-A2-0421-E	1
5	伺服电机	ECMA-C10604ES	1
6	伺服动力线	ASD-ABPW0003	1
7	伺服编码线	ASD-ABEN0003	1
8	EtherCAT 连接线 (1 米)	UC-EMC010-02A	1

● ASDA-A2-E 硬件极限配置

此范例为旋转轴无须硬件极限

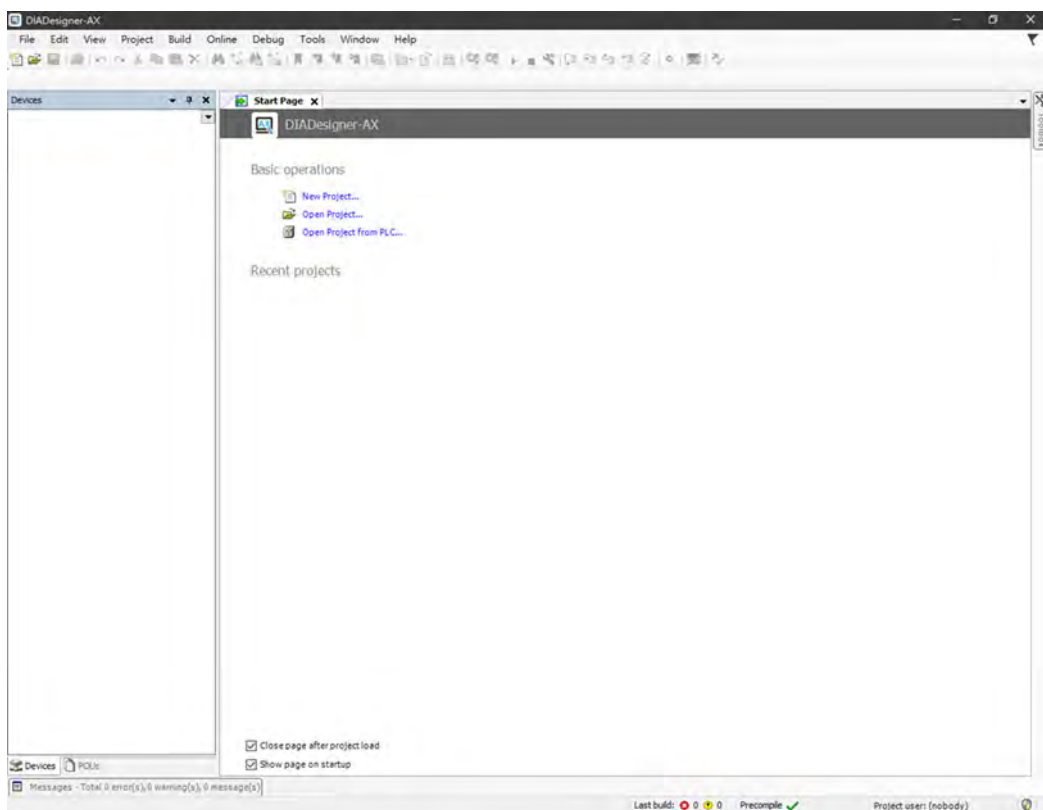
3.4.2 项目建立与程序编写

3.4.2.1 项目开启

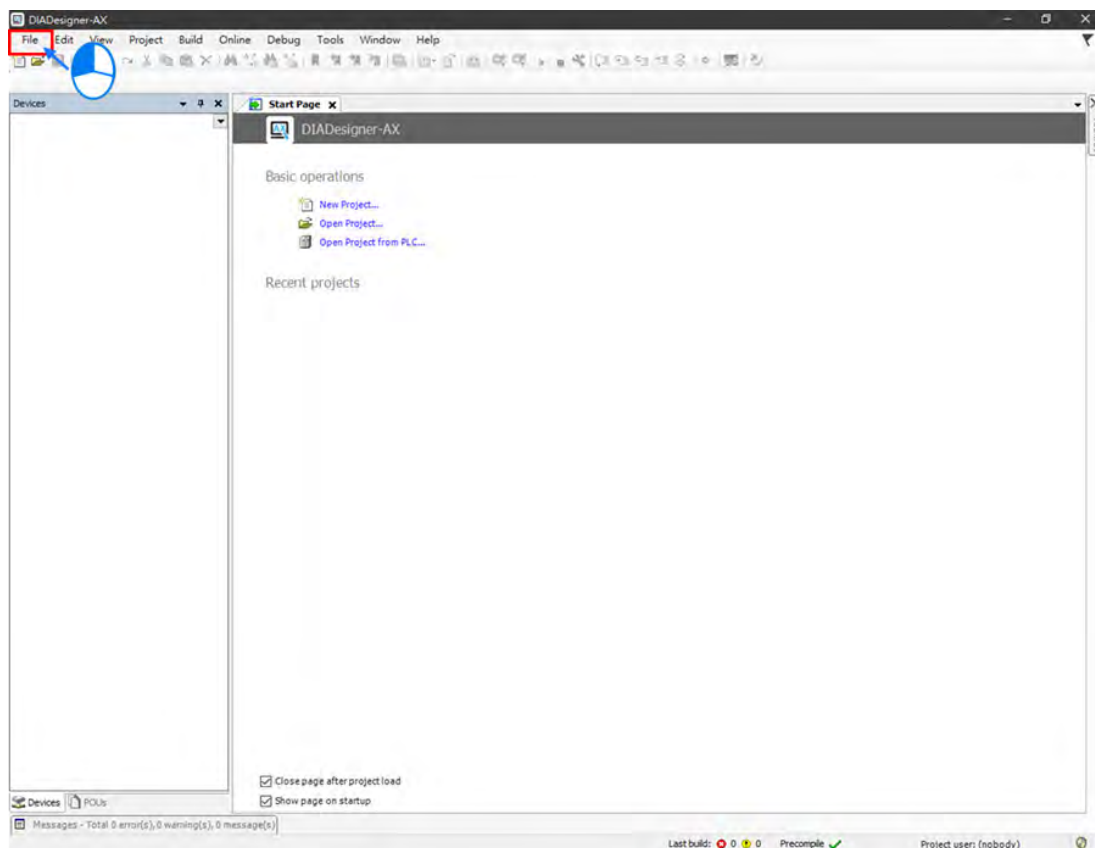
1. 首先 AX-308 编写程序软件为 DIADesigner-AX，所以需先启动软件。



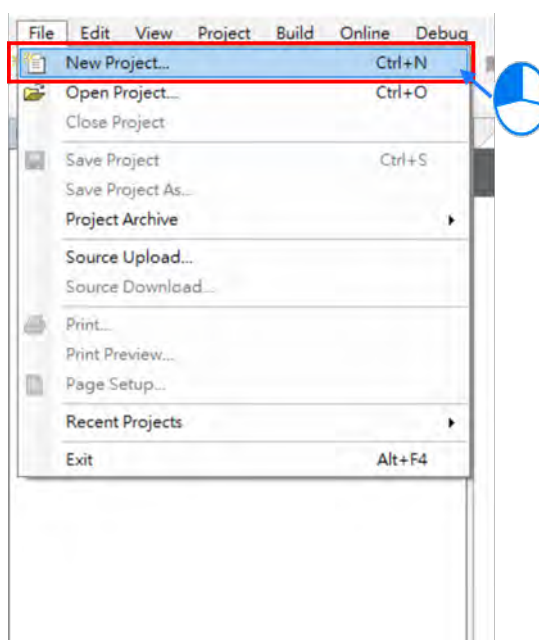
2. 启动 DIADesigner-AX 的画面如下



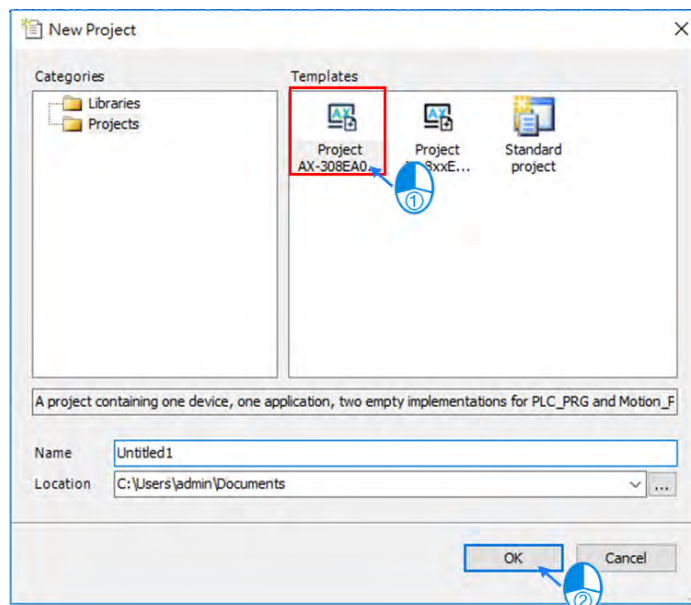
3. 单击「File」后双击鼠标左键



4. 在「File」下拉选单单击「New Project」以开启一个新项目。

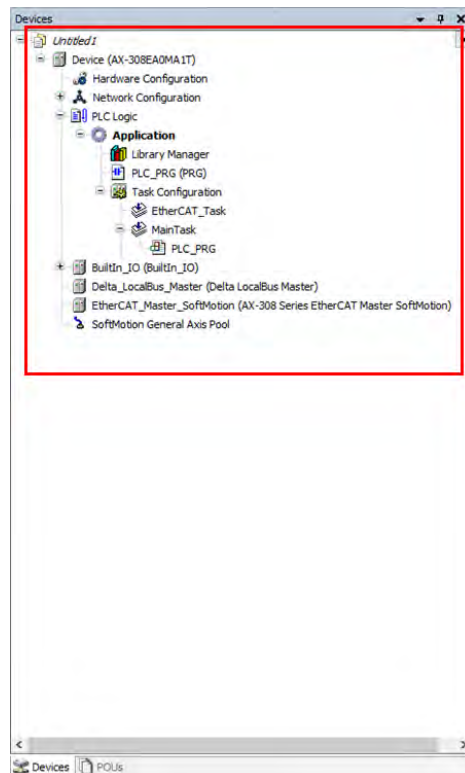


5. 进入「New Project」画面后选择「Project AX-308EA0MA1T*」，可在「Name」处输入项目名称，及「Location」处输入项目存取路径。

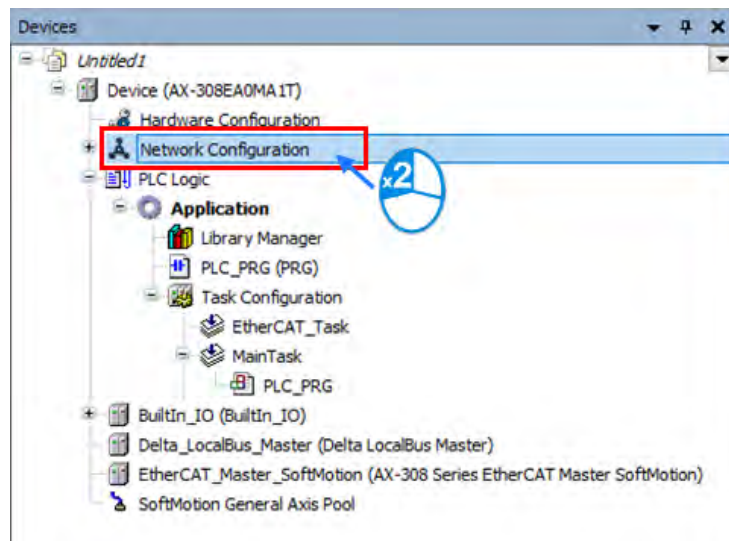


*注：DIADesigner-AX 版本于 V1.1.0 以上的操作为点选「Standard Project」后选择机种。

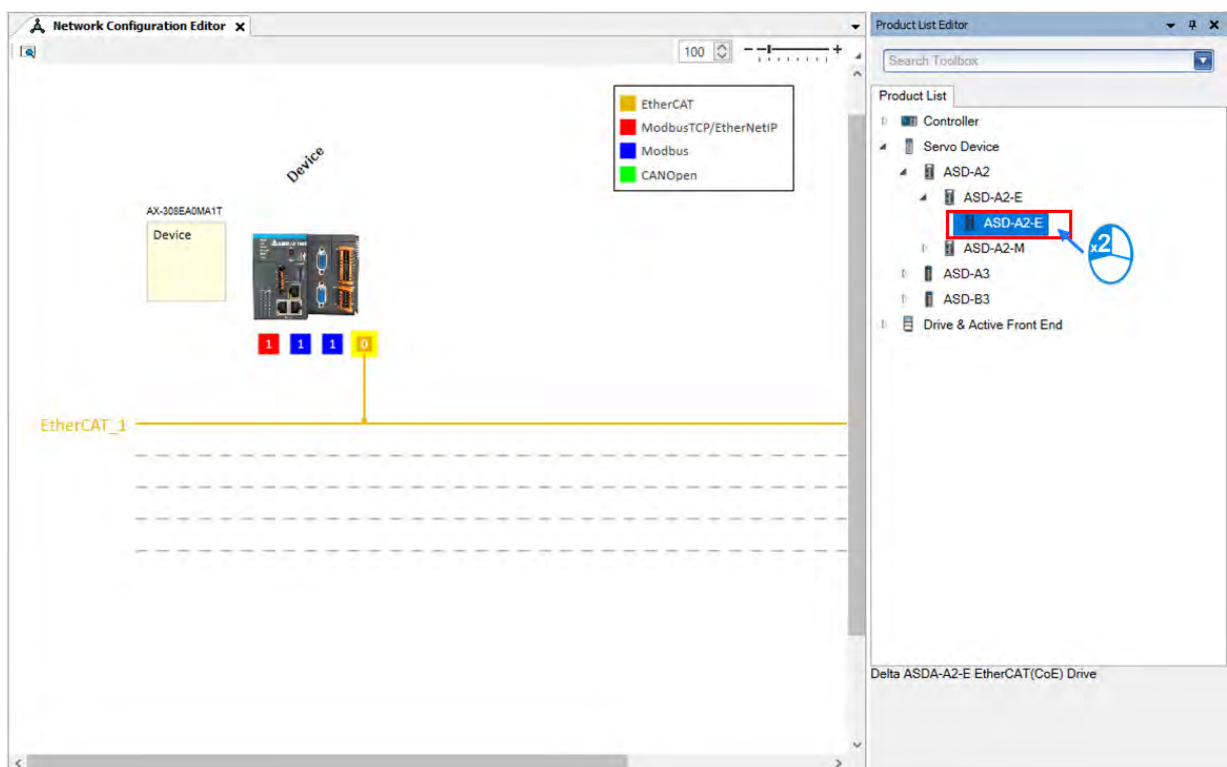
6. 进入项目画面如下，左边为项目树，可双击左键以开启程序画面或是模块相关设定。



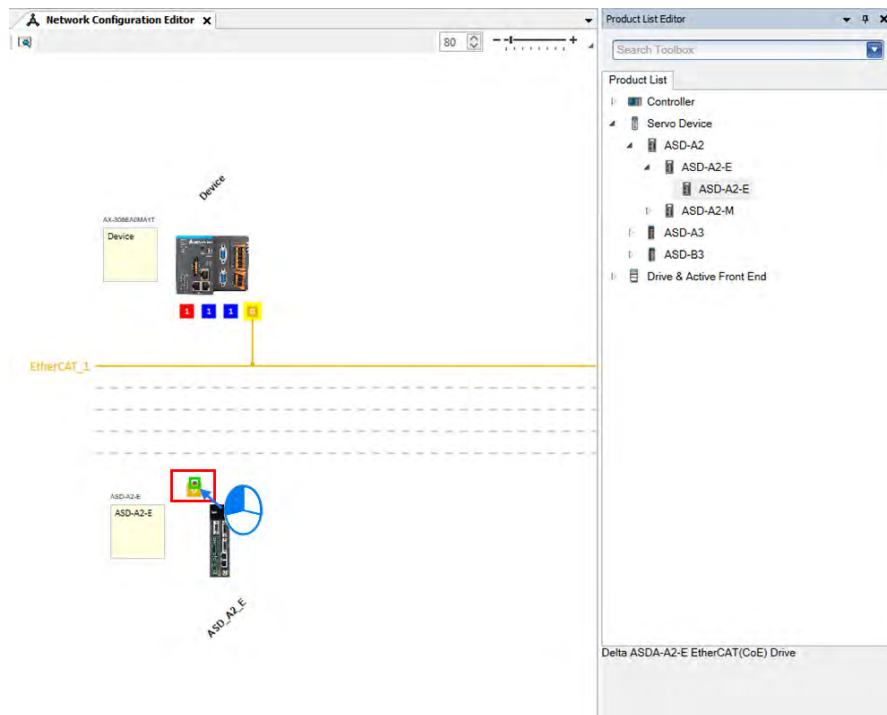
7. 先建立系统架构的伺服轴，双击「Network Configuration」。



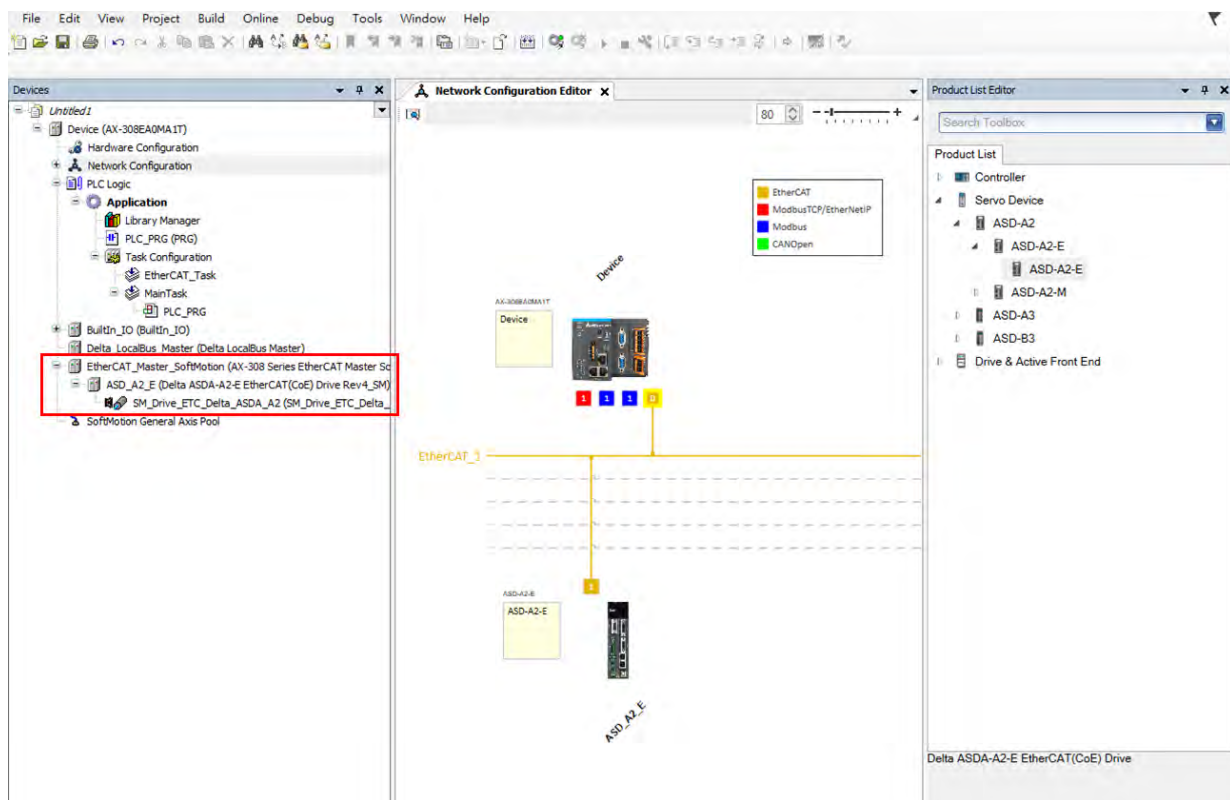
8. 进入「Network Configuration」画面后，选择 EtherCAT 所配置的 Slave 型号，双击左键以新增轴。



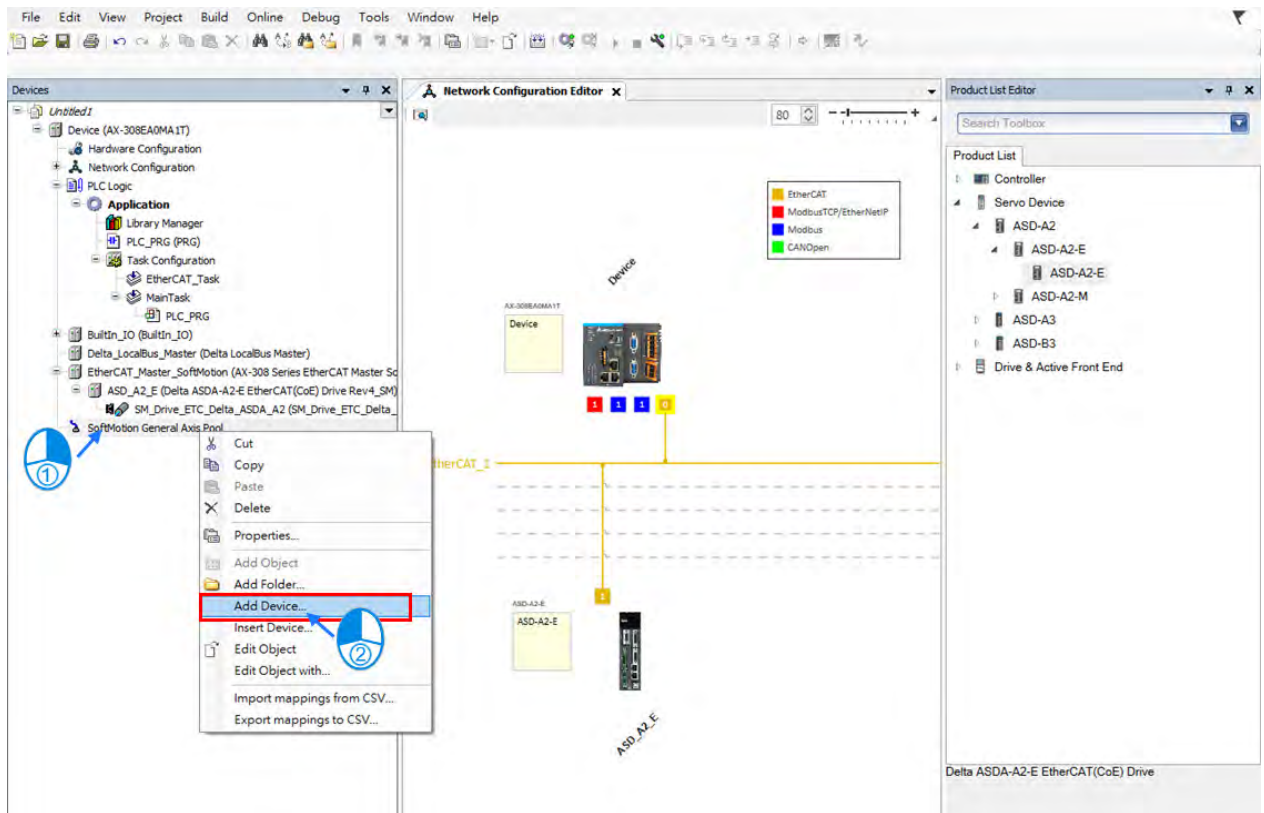
9. 将鼠标移至 **0** 上方，按压左键将黄色线连接至上方 EtherCAT_1。



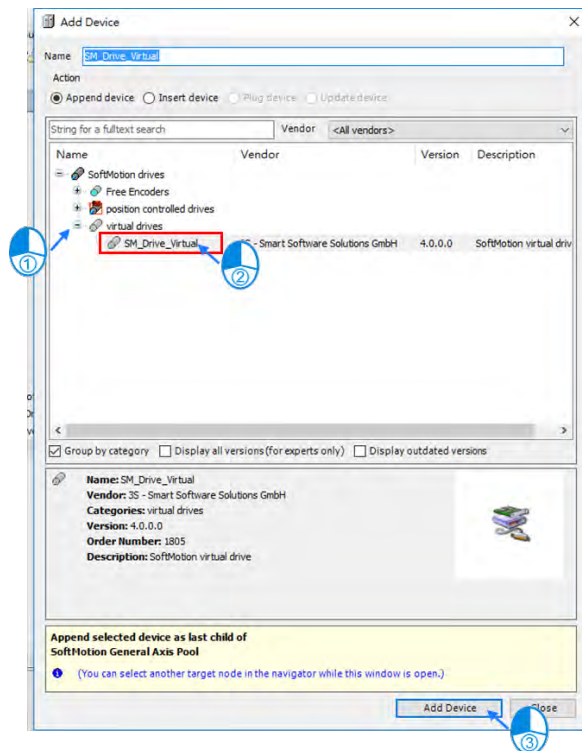
10. 将 Slave 都联机至 EtherCAT_1 后，左方树形图才会显示 Slave，代表此 Slave 有配置在 EtherCAT 通讯中。



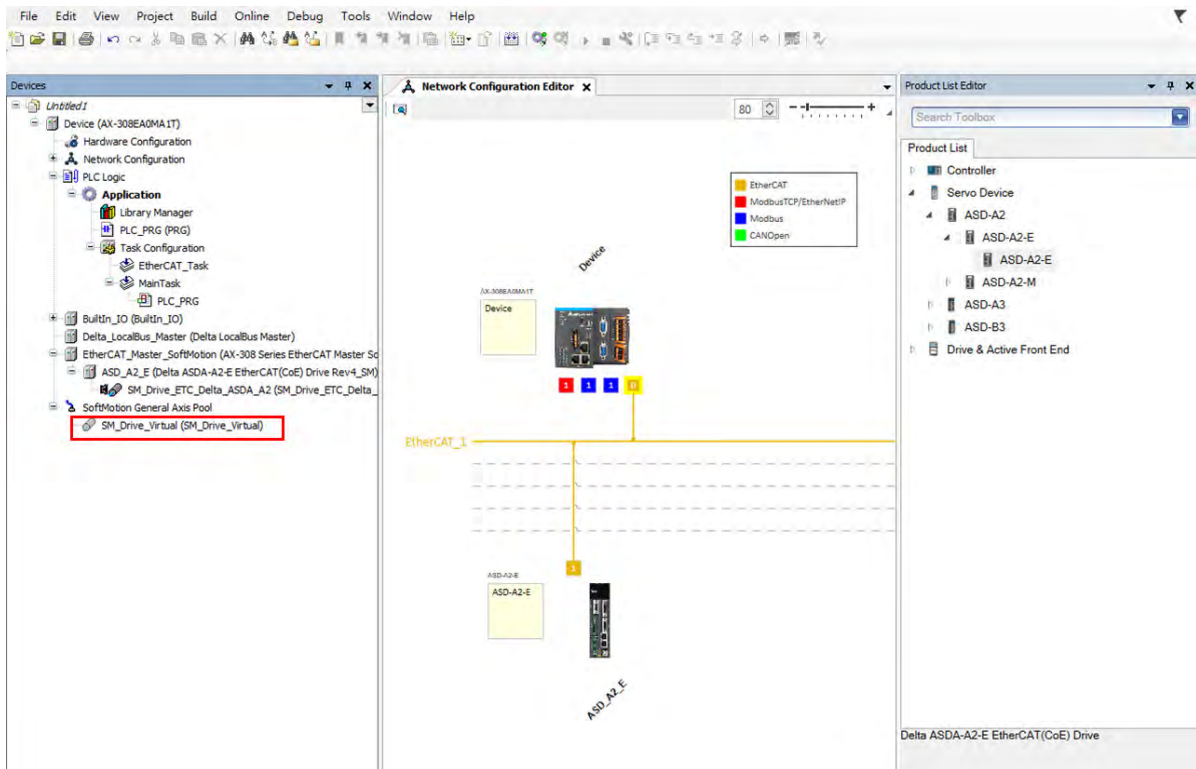
11. 建立好实轴后，接下来进行虚轴建立。先单击「SoftMotion General Axis Pool」右键，选择「Add Device」。



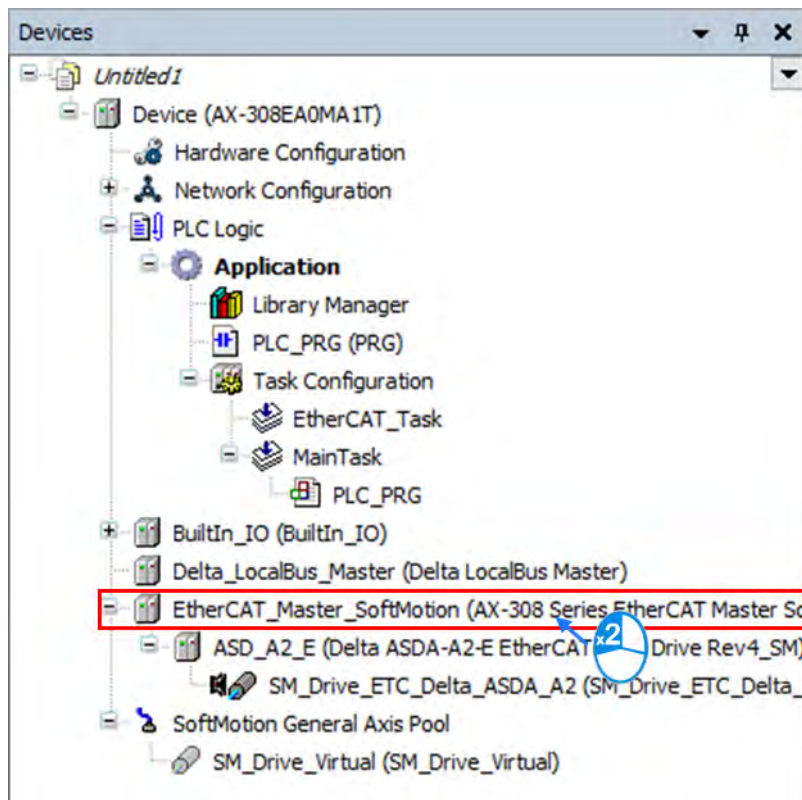
12. 进入「Add Device」画面中，将「Virtual drives」点开，选择「SM_Drive_Virtual」 > 「Add Device」。



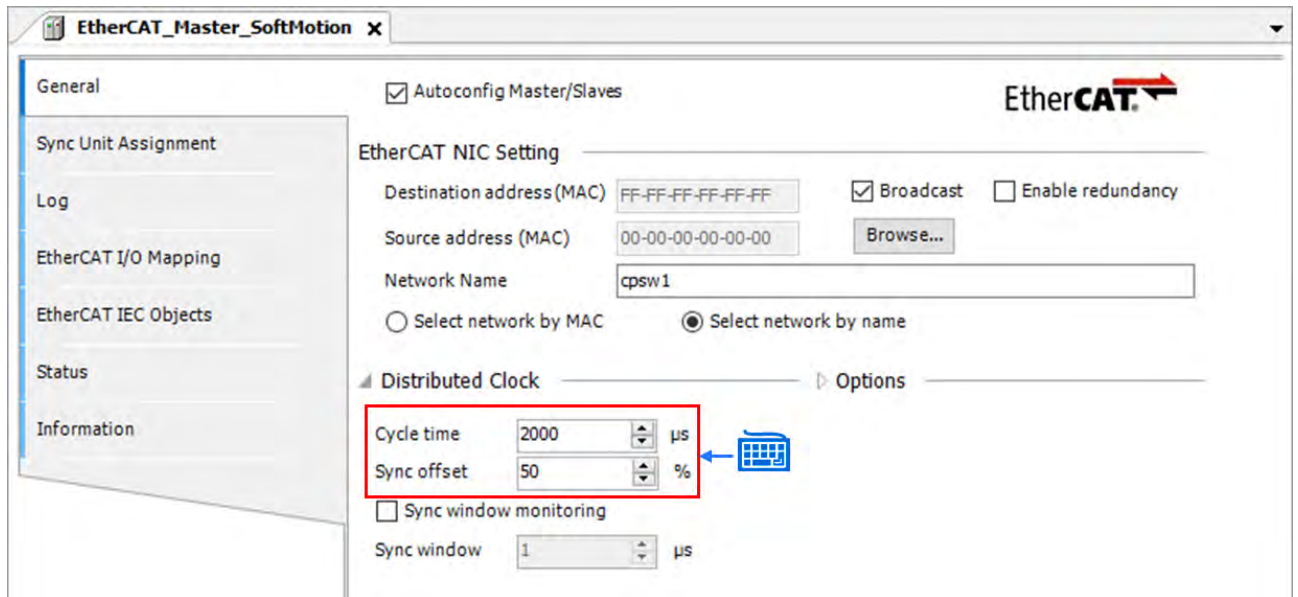
13. 建立好虚轴后，树形图才会出现「SM_Drive_Virtual」。



14. 双击「EtherCAT_Master_SoftMotion」。

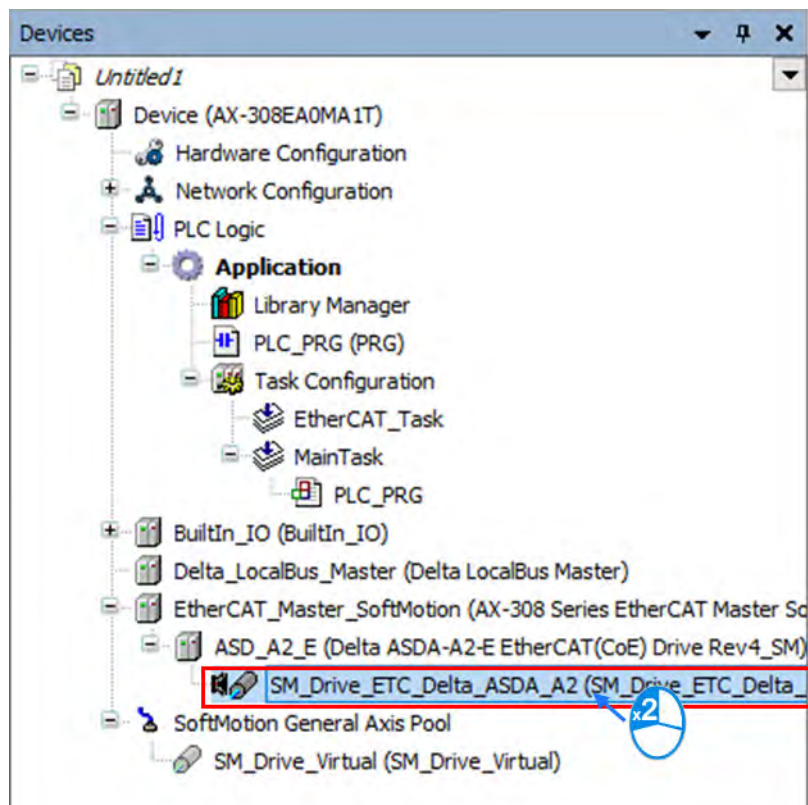


15. 进入「EtherCAT_Master_SoftMotion」画面后，设定「Cycle time」为 2ms 及「Sync offset」为 50。

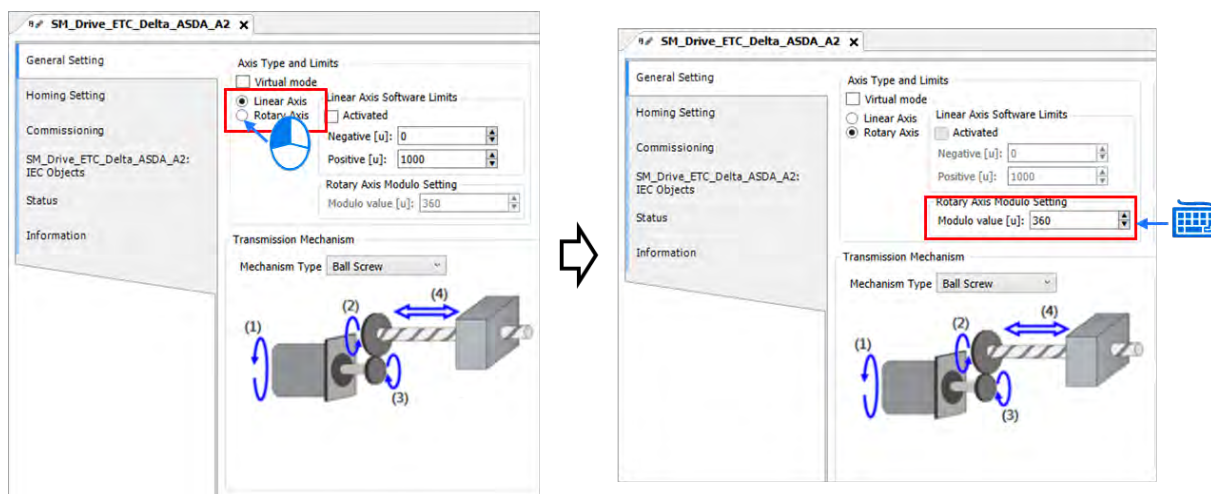


3.4.2.2 轴参数设定

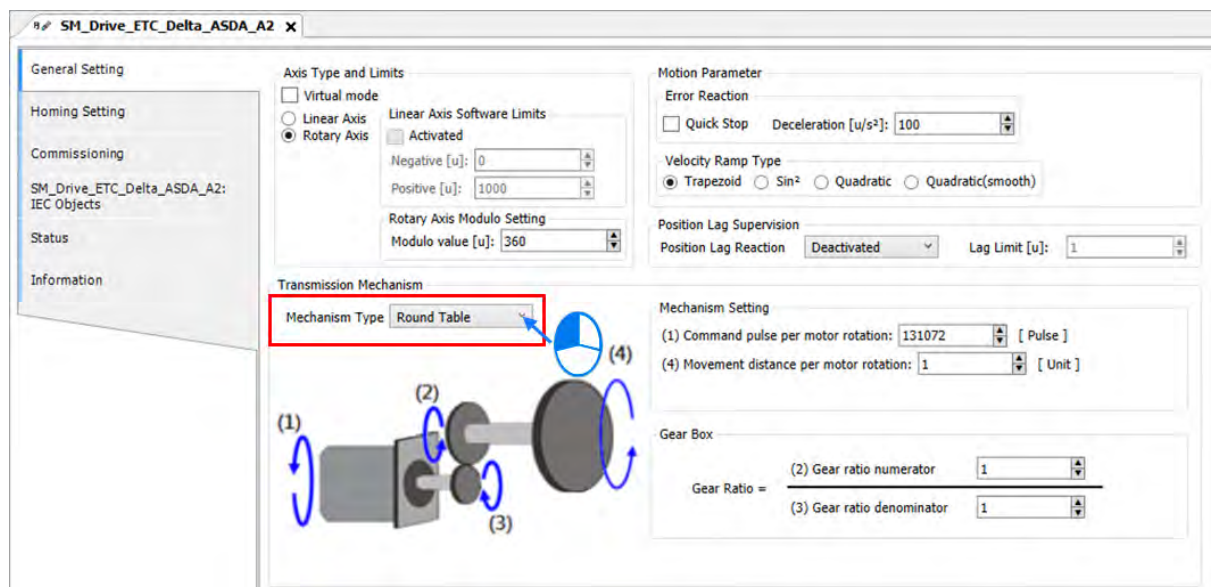
1. 首先选择第一轴「SM_Drive_ETC_Delta_ASDA_A2」后，单击左键两下。



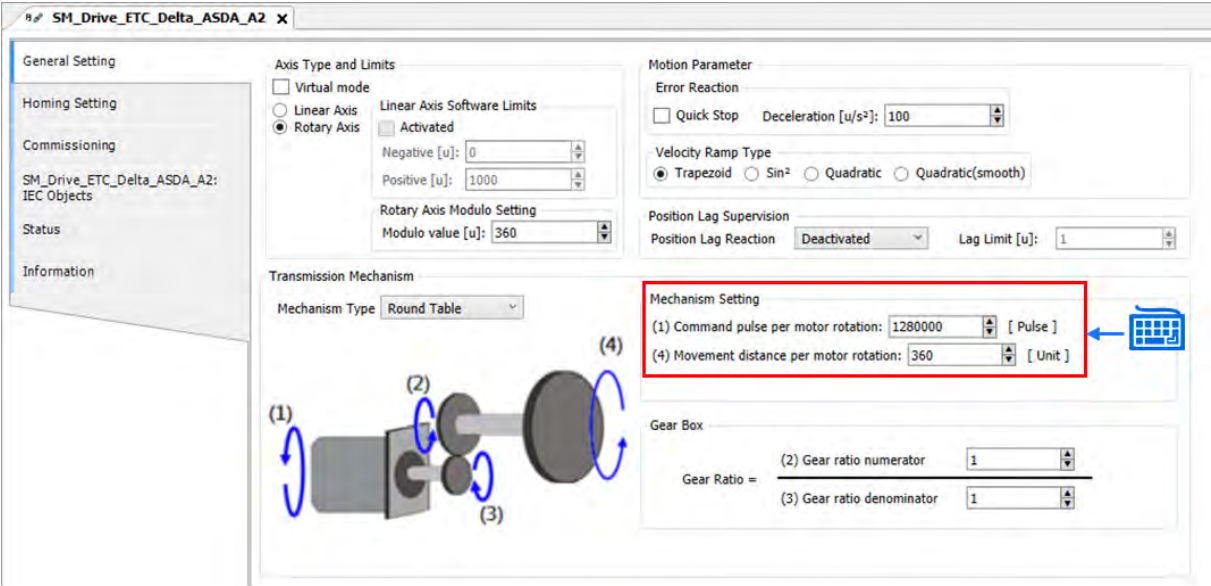
2. 进入「SM_Drive_ETC_Delta_ASDA_A2」画面后，机构为旋转轴，需将直线轴 (Linear Axis) 改为旋转轴 (Rotary Axis)，旋转轴可设定旋转轴最大范围，此范例为 360 度。



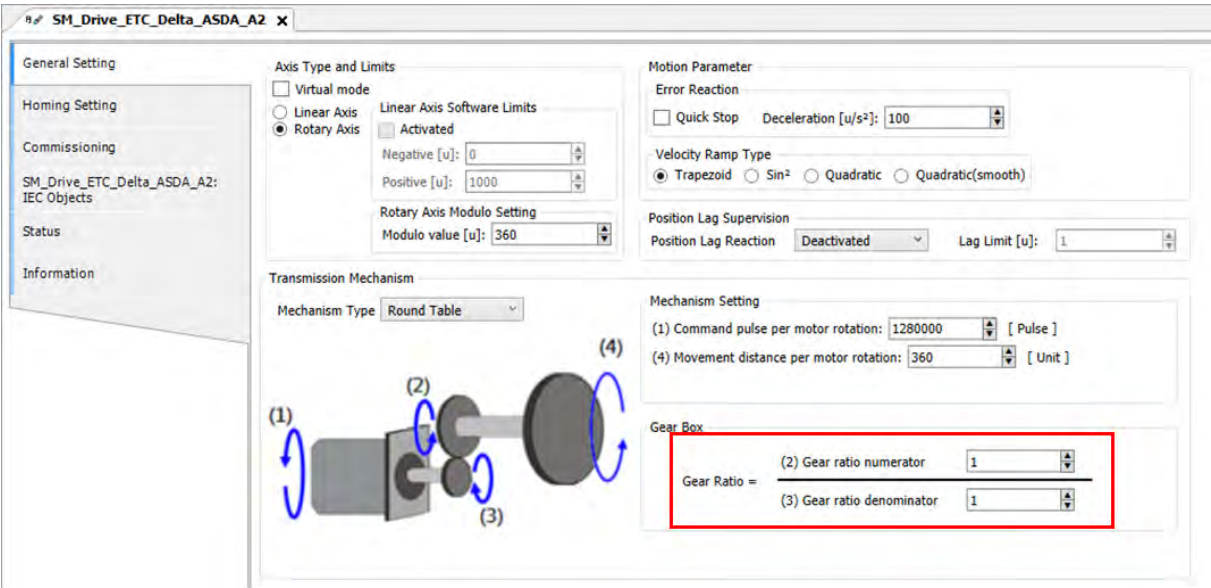
3. 此范例机构为旋转轴，所以选择「Round Table」。



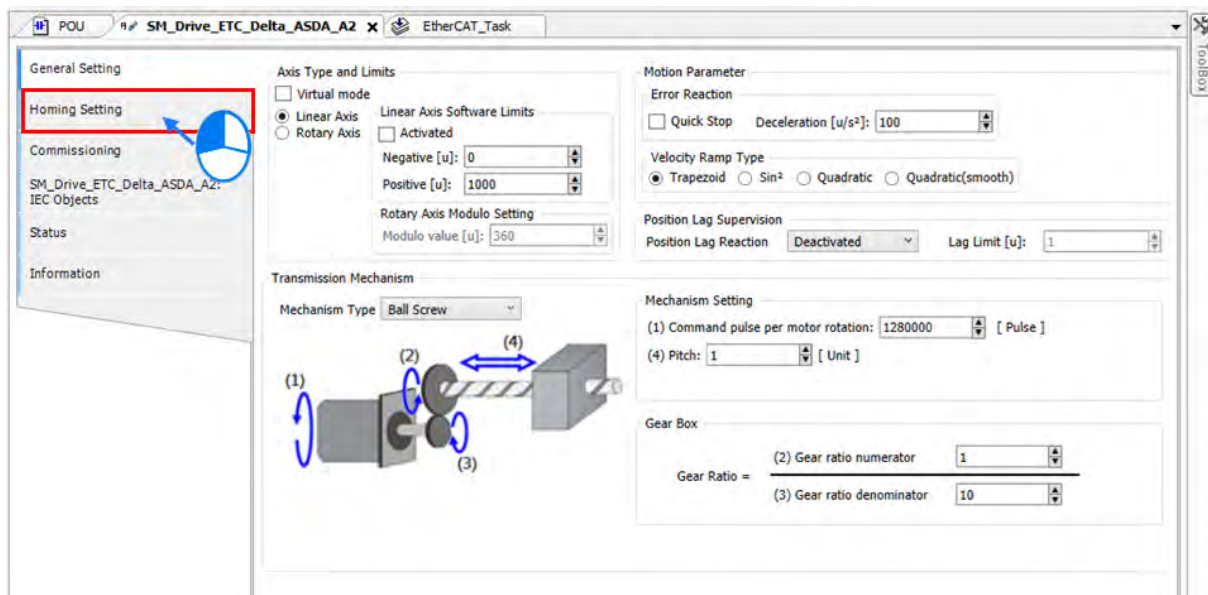
4. 实体机构选择完毕再设定齿比关系，在一圈脉冲（Command pulse per motor rotation）输入 1,280,000，用户单位为 360 度。（伺服端齿比关系 $\frac{P1-44}{P1-45} = \frac{\text{电机一圈脉冲数}}{\text{Command Pulse per rotation}}$ ）



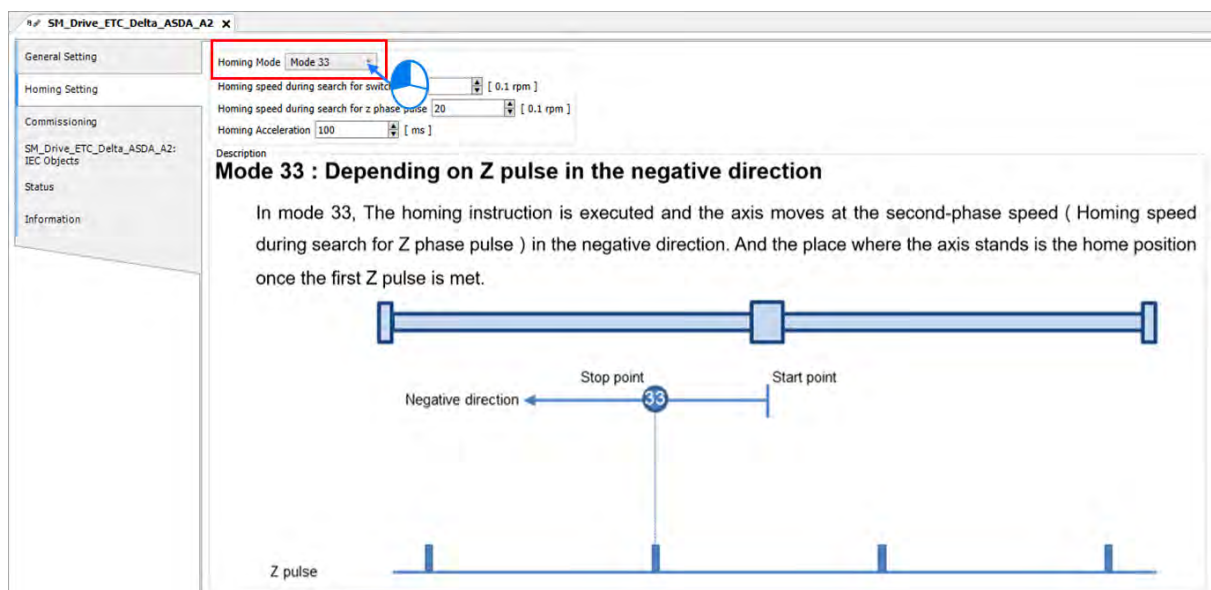
5. 实体机构未配减速机所以为 1 : 1。



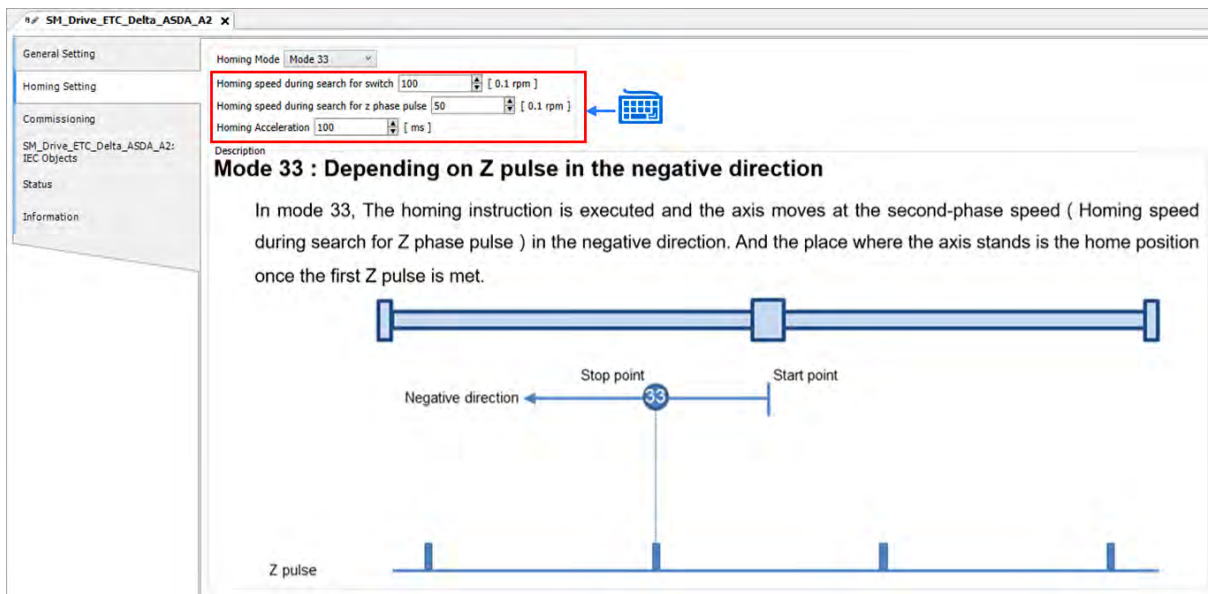
6. 将齿比关系设定完毕后，单击「Homing Setting」。



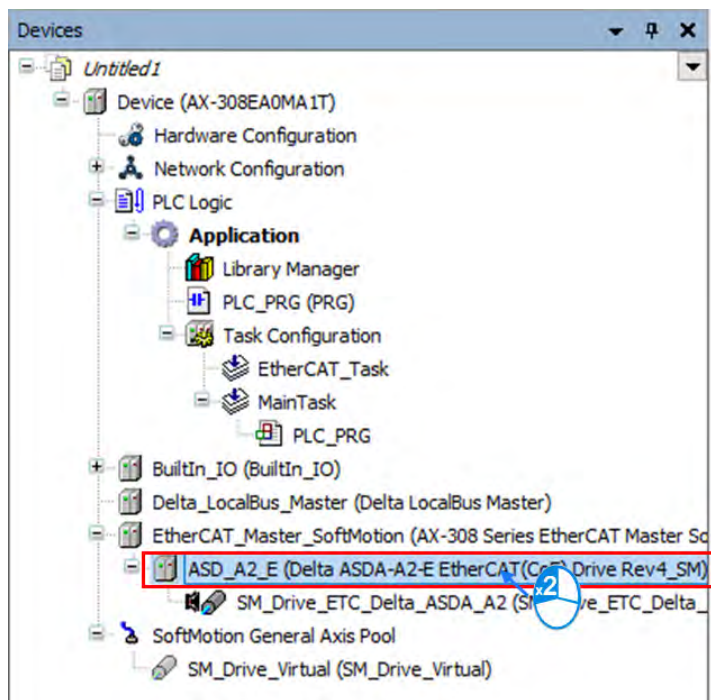
7. 因机构为旋转轴，所以选择 Mode 33，以 Z 相方式回原点模式，每种模式请参考下方详细说明。



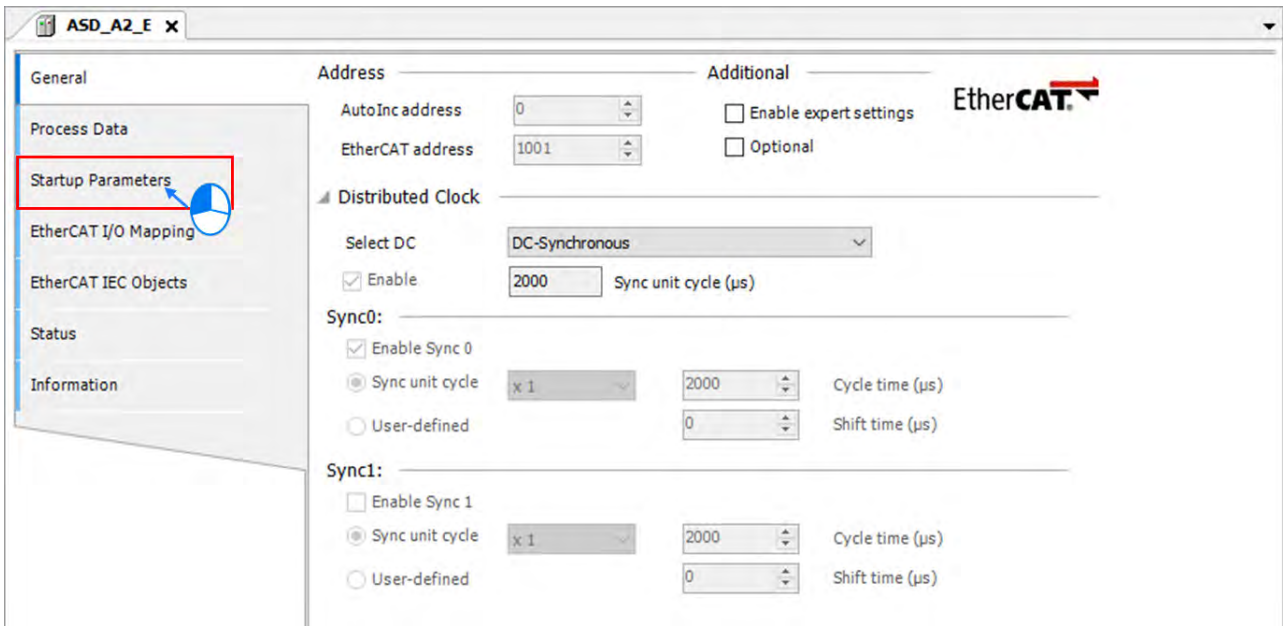
8. 再设定回原点的速度，此速度需搭配机构调整为适当速度。



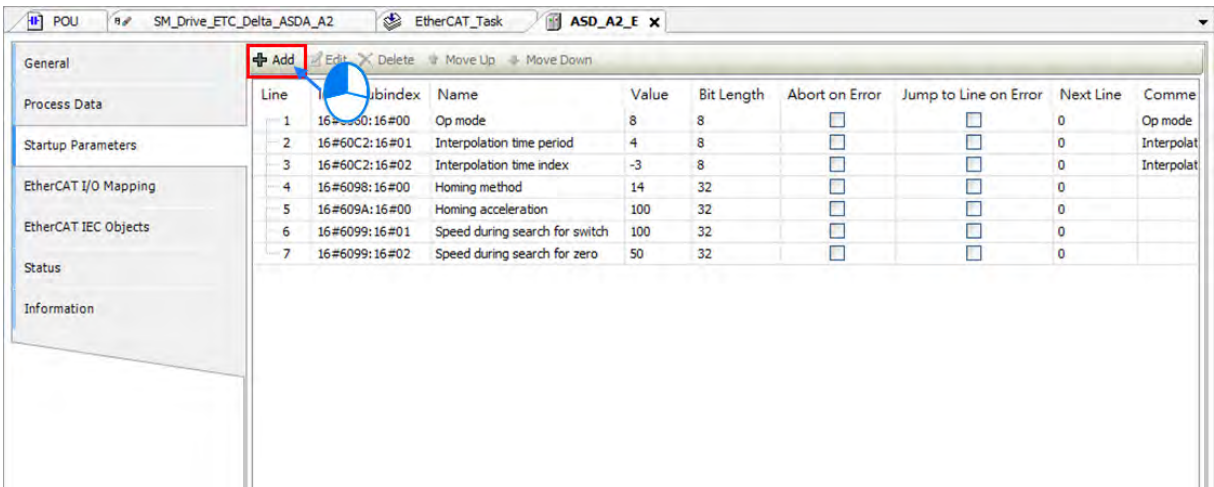
9. 将齿比与原点模式设定后，需在测试伺服端的参数，因硬件极限配置在伺服端的 DI9~DI10，所需参数设定 DI9 与 DI10 的引脚功能，此时单击「ASD_A2_E」。



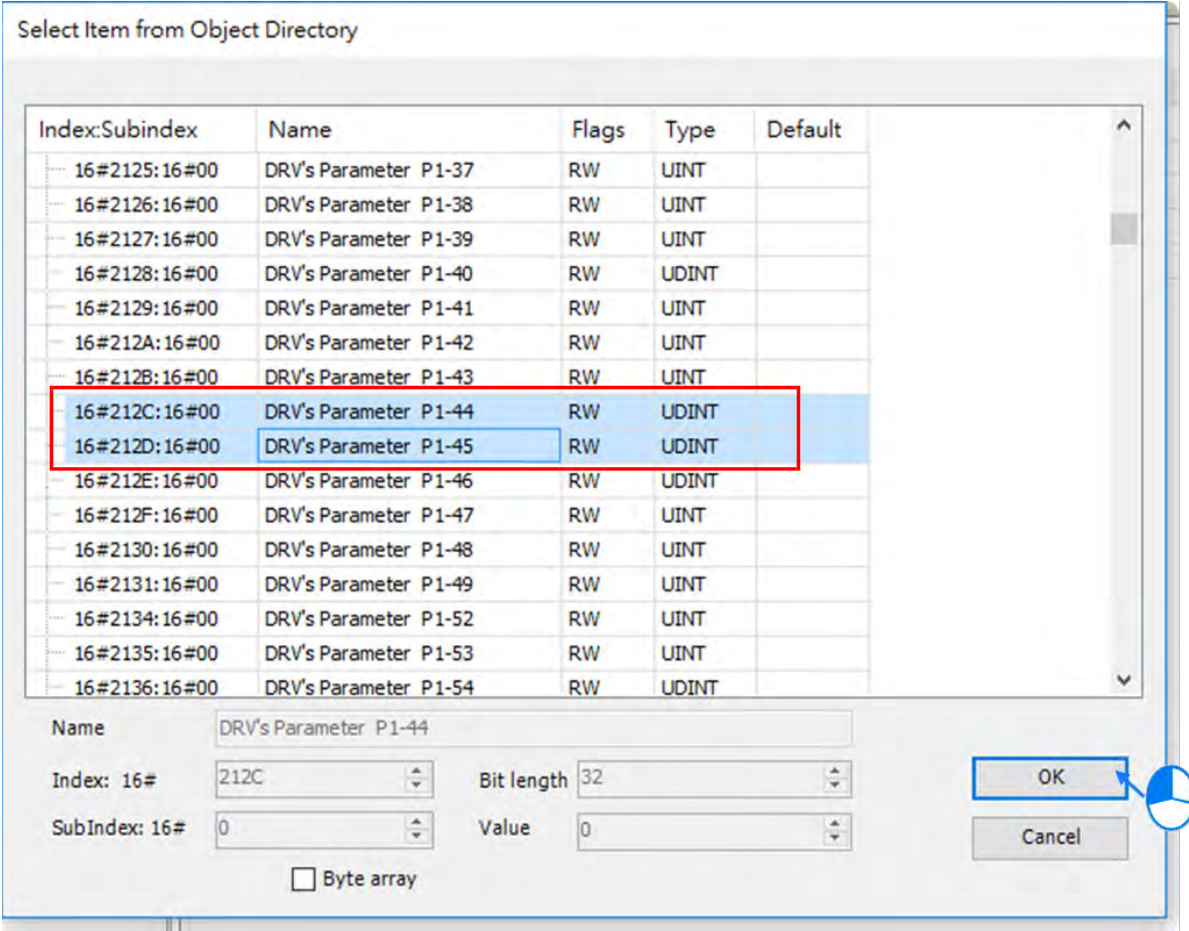
10. 进入「ASD_A2_E」画面再单击「Startup Parameters」。



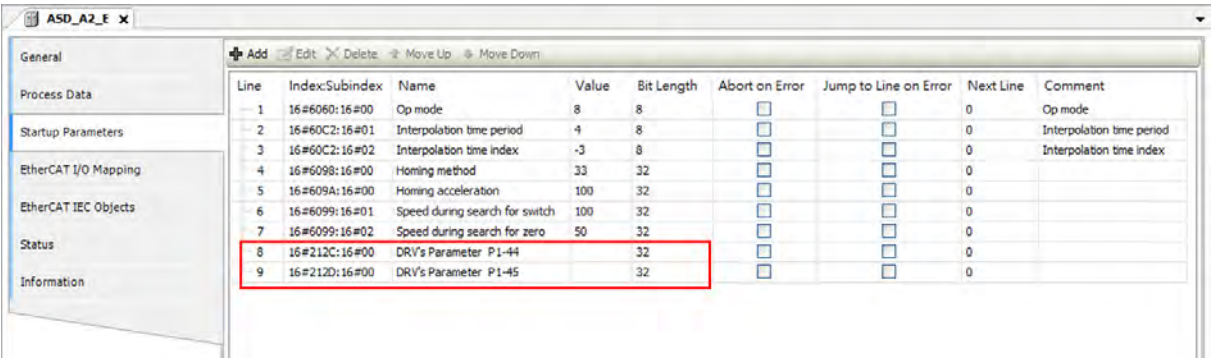
11. 在「Startup Parameters」画面中，单击「Add」新增参数。



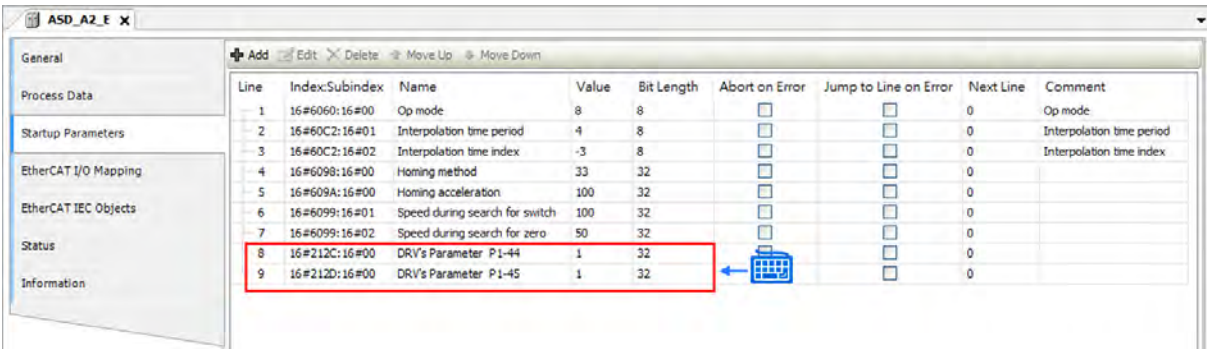
12. 单击 P1-44 后可搭配 Shift 按键，选取多个参数，确认后再单击「OK」即可新增。



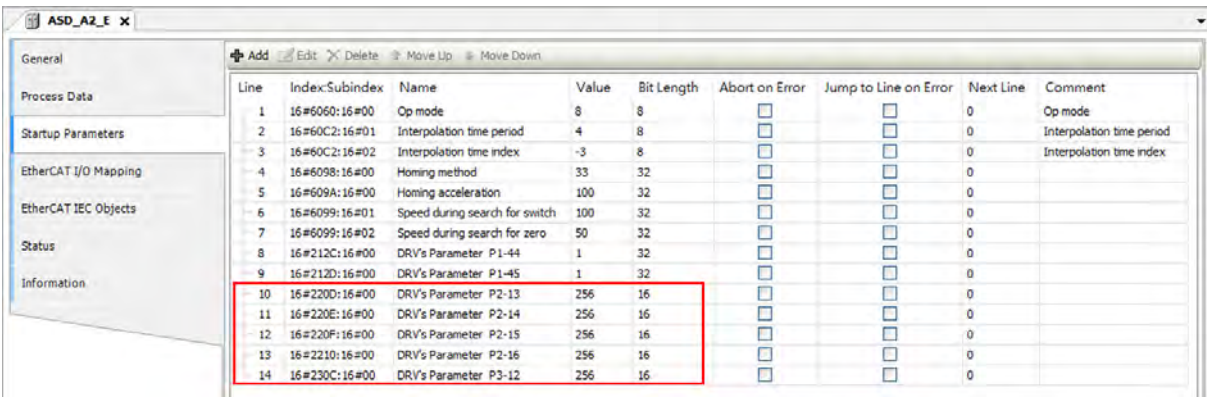
13. 新增后参数表中会出现刚刚所选择的参数。



14. 将 P1-44 输入为 1 及 P1-45 设定 1。(此设定为伺服端的齿比需搭配轴参数中的齿比)(在此画面的参数会在 EtherCAT 通讯建立完毕后，下载所有设定值于伺服端)



15. 利用相同操作设定下列参数，将伺服端 DI4~DI7 功能关闭，并设定断电保持参数。



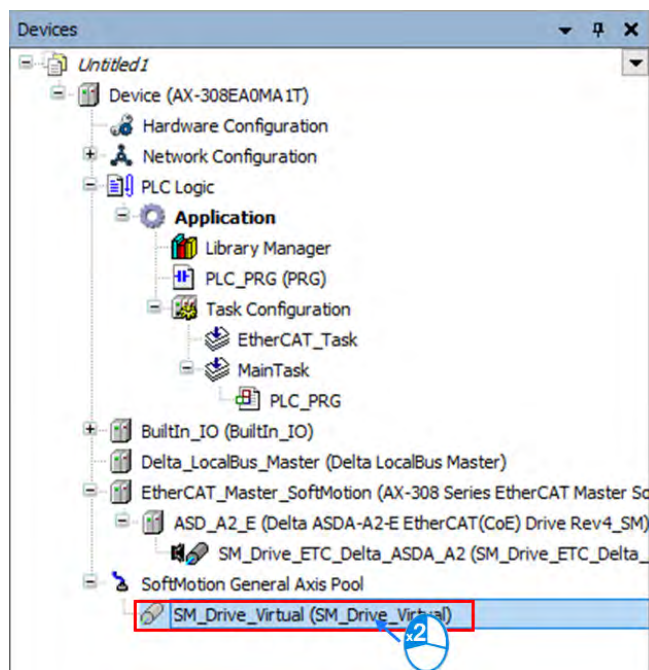
■ 伺服端参数

下列参数为此范例使用，请输入至「Startup Parameters」中。

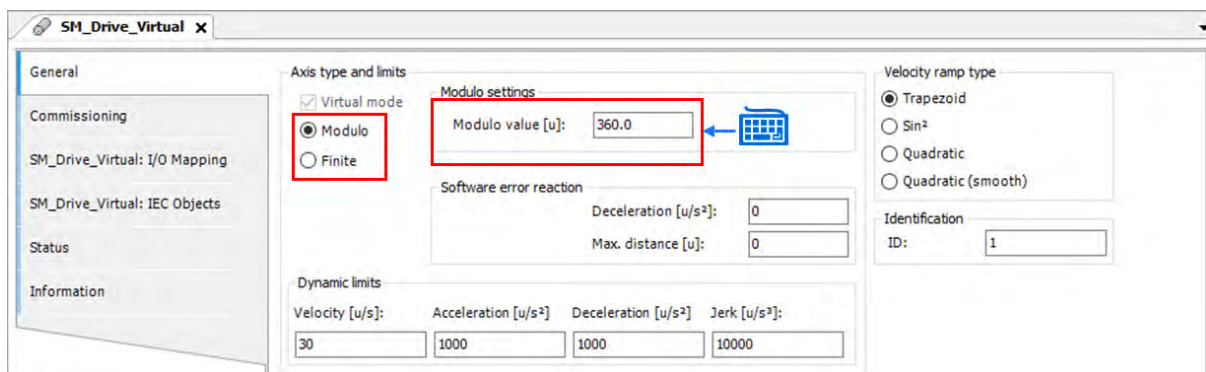
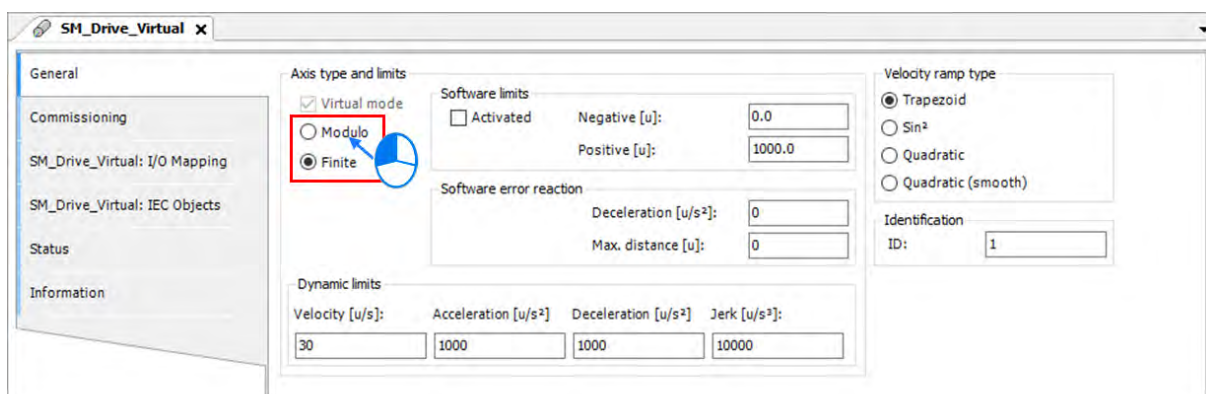
参数	功能	设定值
P1-44	电子齿轮比分子	1
P1-45	电子齿轮比分母	1
P2-13	数字输入引脚 DI4 功能规划	256
P2-14	数字输入引脚 DI5 功能规划	256
P2-15	数字输入引脚 DI6 功能规划	256
P2-16	数字输入引脚 DI7 功能规划	256
P3-12	参数会维持驱动器断电前的参数	256

注：参数详细介绍可参考 ASDA-A2 系列应用技术手册

16. 实轴设定完毕后，再设定虚轴，单击左键两下。



17. 进入虚轴设定画面后，将直线轴 (Finite) 改为旋转轴 (Modulo)，移动一圈最大值为 360。



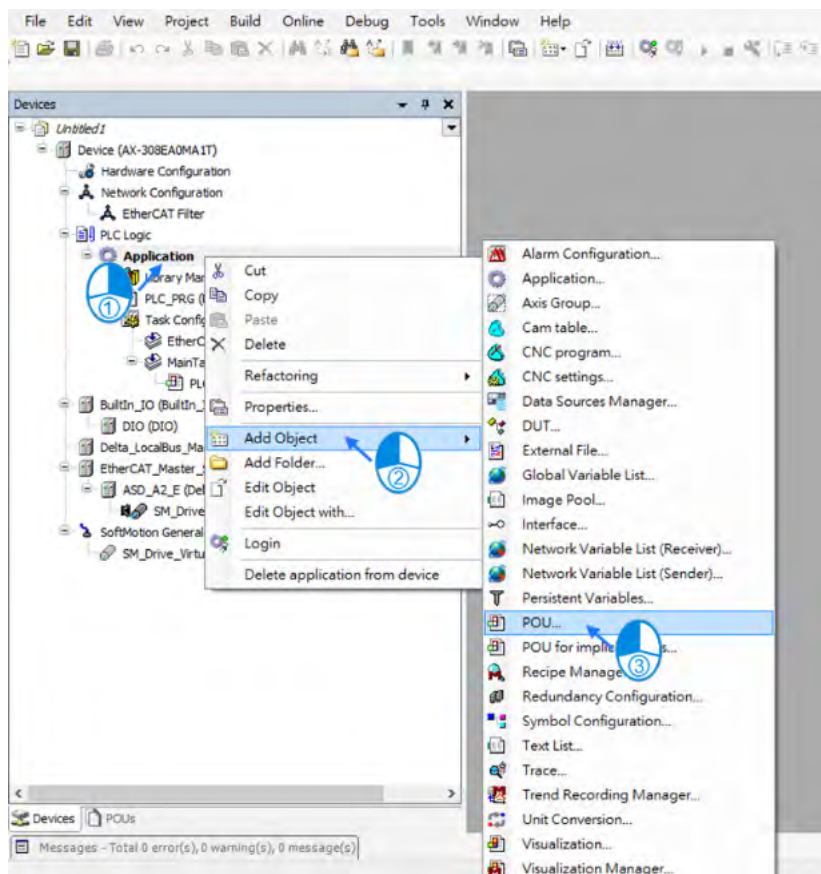
3.4.2.3 项目程序编写

● 程序执行流程

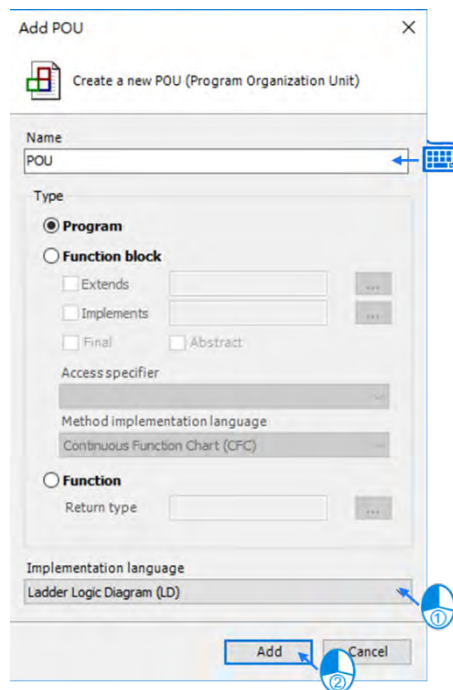
- 执行「Step1」后会触发「MC_Power」，让两轴 Servo ON。
- 执行「Step2」后会触发「MC_Home」，让实轴回原点，虚轴需要使用「MC_SetPosition」将位置改为 0。
- 执行「Step3」后会触发「MC_Jog」让虚轴开始转动，此时会一起触发「MC_GearIn」，从轴会开始跟随，当「Step3」关闭后「MC_Jog」会停止触发，此时会一起触发「MC_GearOut」「MC_GearOut」的 Done 启动后，会再触发「MC_Stop」功能，让从轴停止运转，此流程会重复执行直到「Step3」接点关闭。
- 自动化执行一次，「Counter 1」会增加一次。

● 项目建立流程

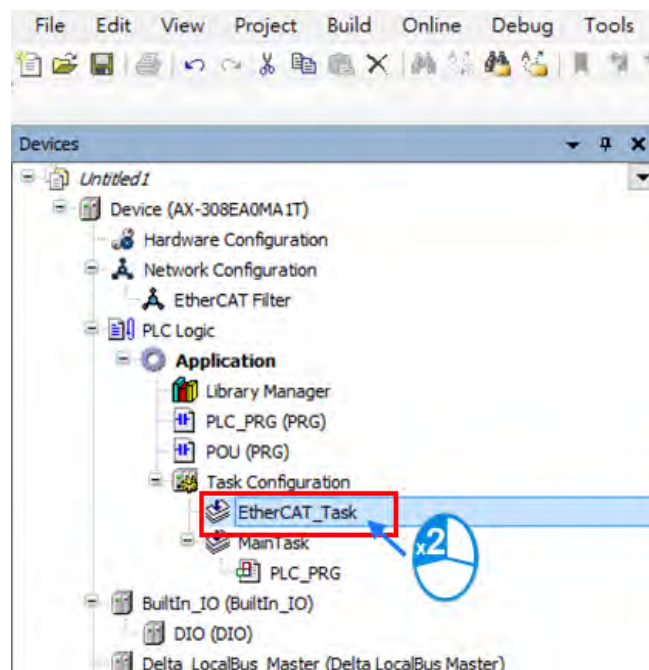
1. 首先建立新的 POU，选择「Application」单击右键，再选择「Add Object」 > 「POU」。



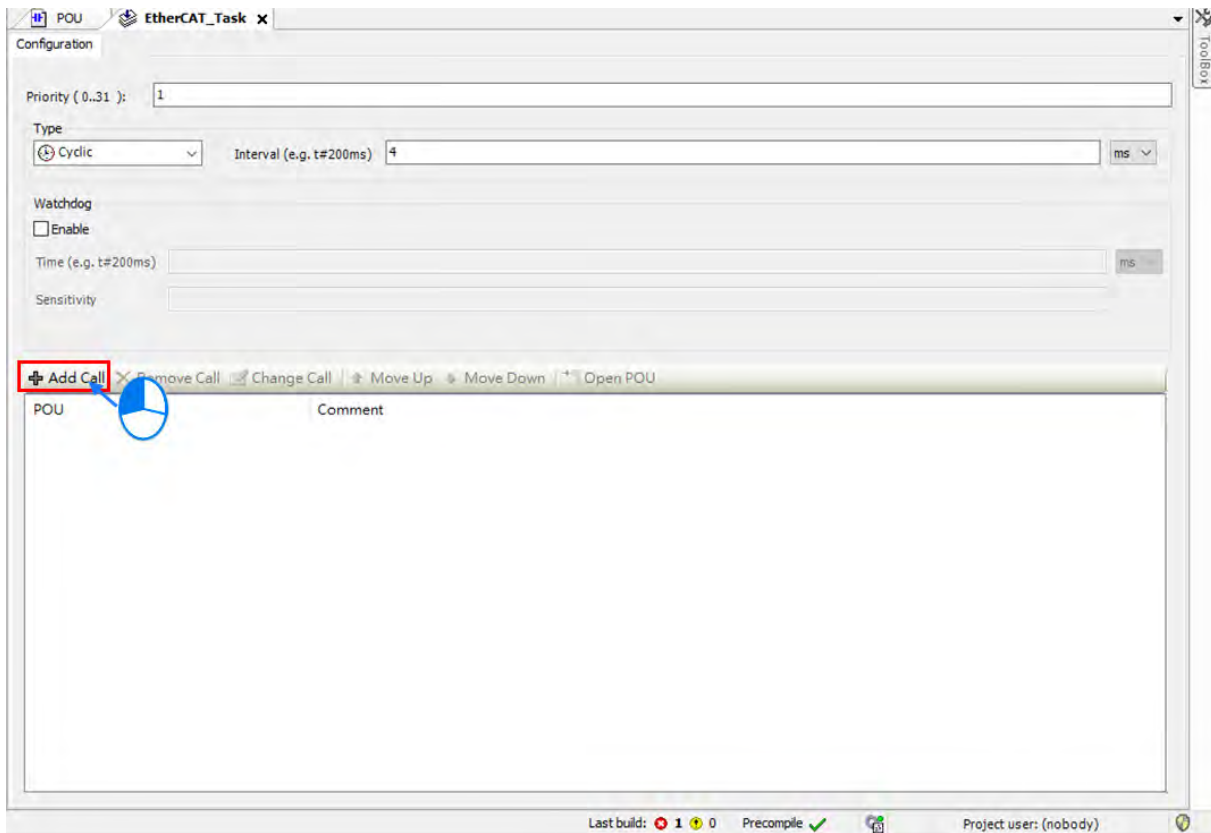
2. 新增画面中，「Name」可输入名称，在「Implementation language」可选择 LD 程序语言，完成后单击「Add」，新增 POU。



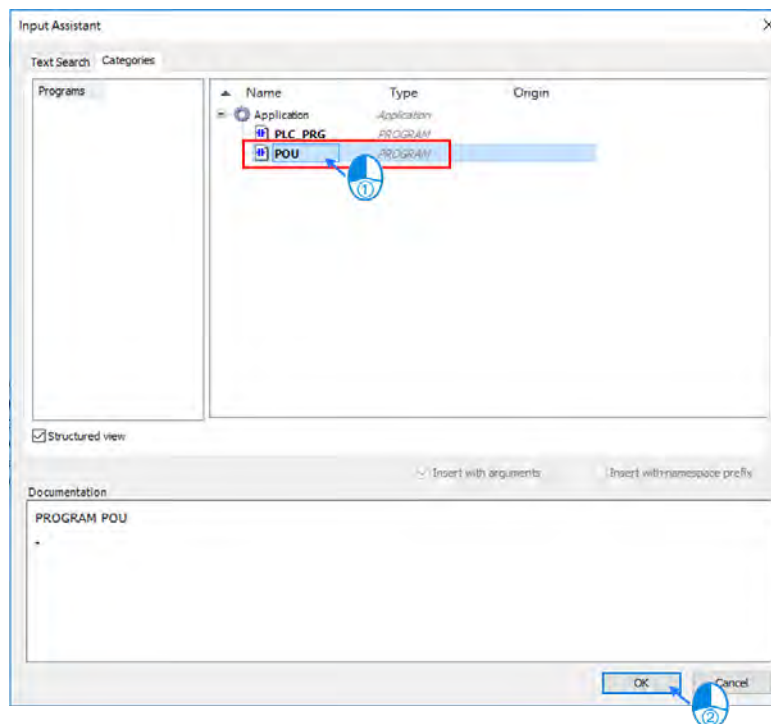
3. 新增的 POU 会出现在树状中，此时需将 POU 新增在 Task 中，单击「EtherCAT_Task」。(有使用轴运动相关 FB 都需建立在 EtherCAT_Task 中以确保轴可正常运作)



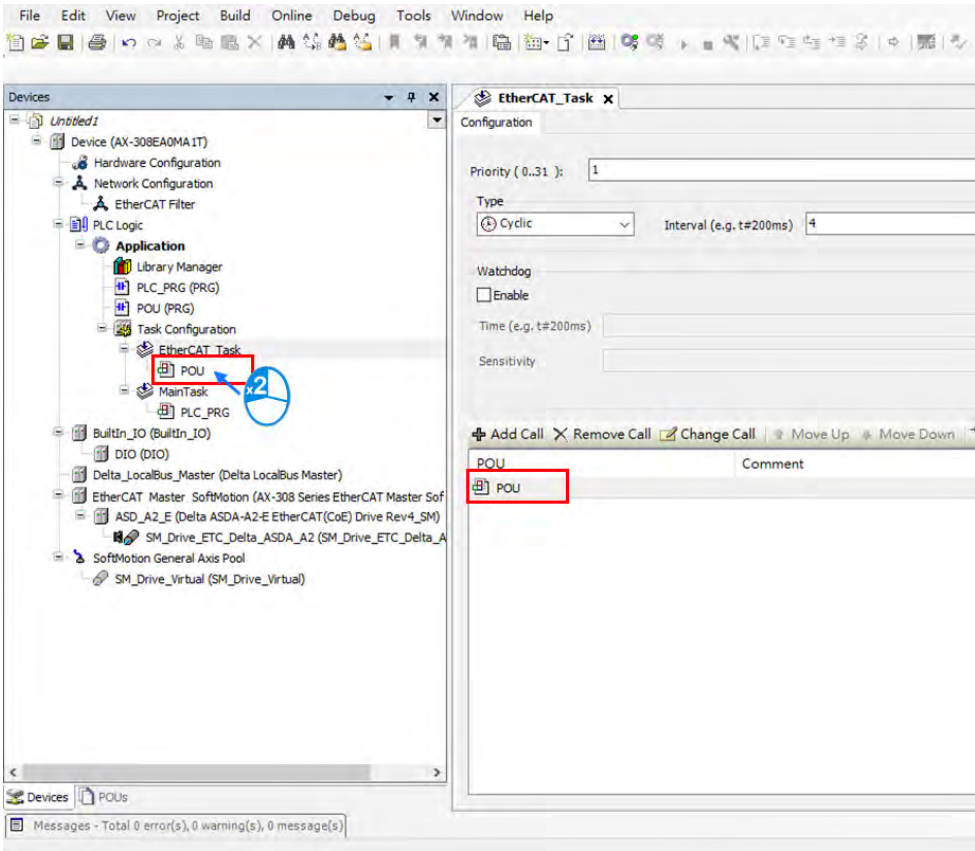
4. 在「EtherCAT_Task」中，单击「Add Call」



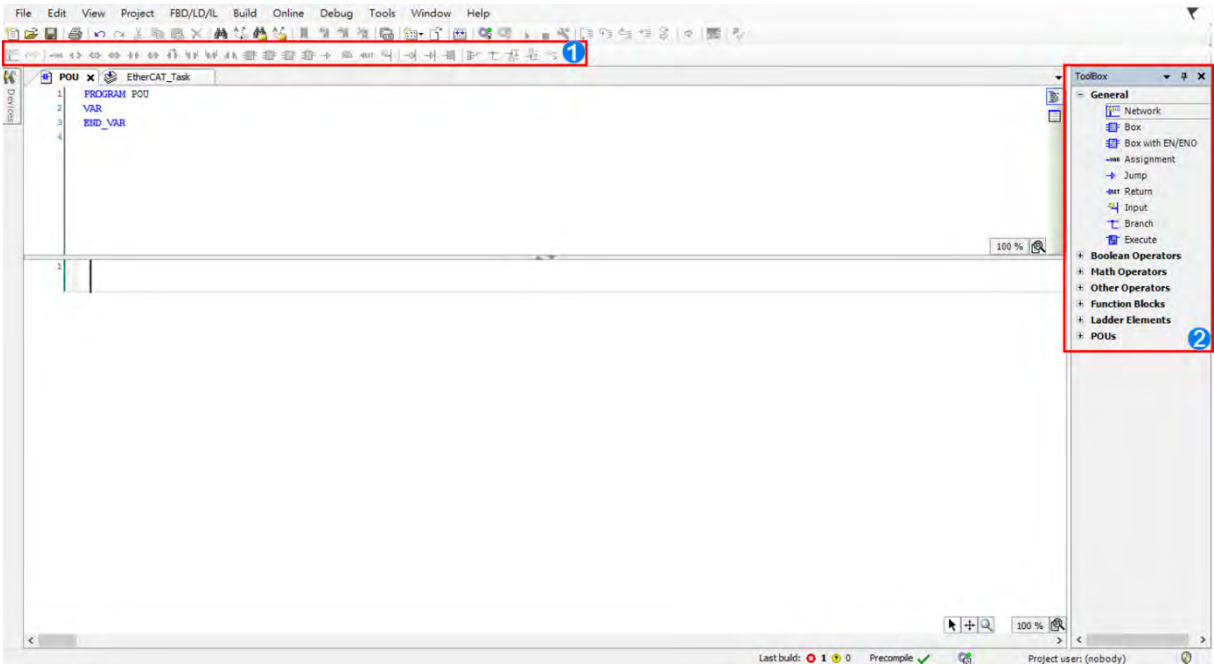
5. 选择刚所建立的「POU」后，单击「OK」。



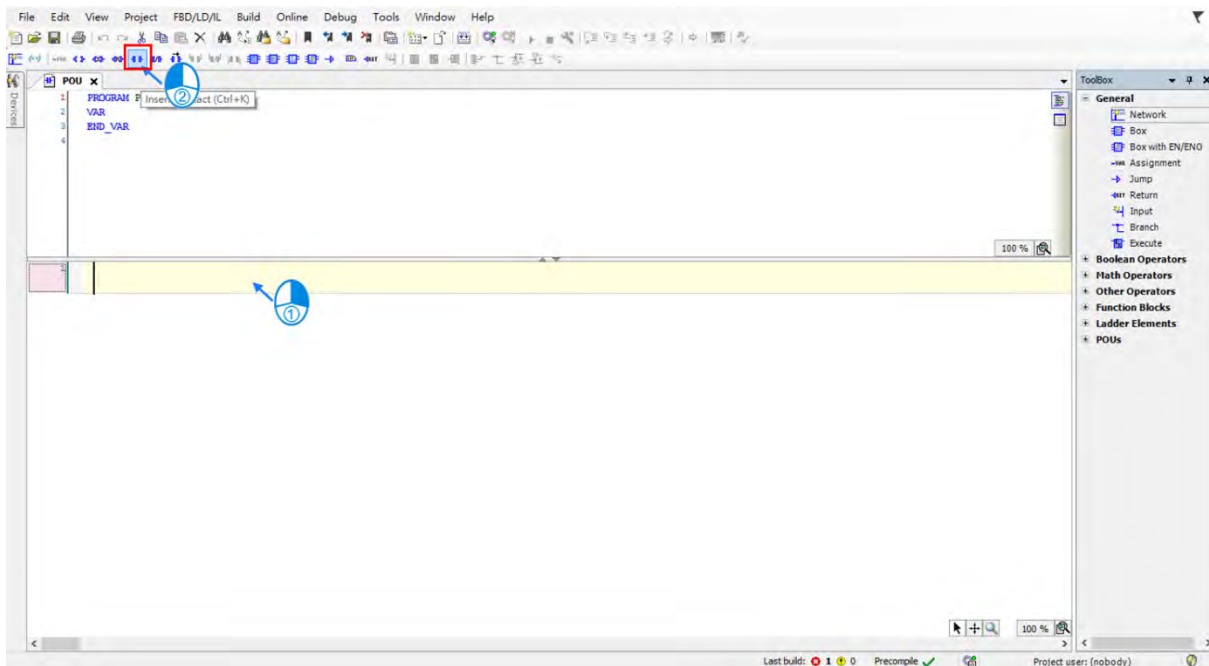
6. 在「EtherCAT_Task」新增完毕后，下方及树形图会出现「POU」，此时再双击树形图的「POU」。



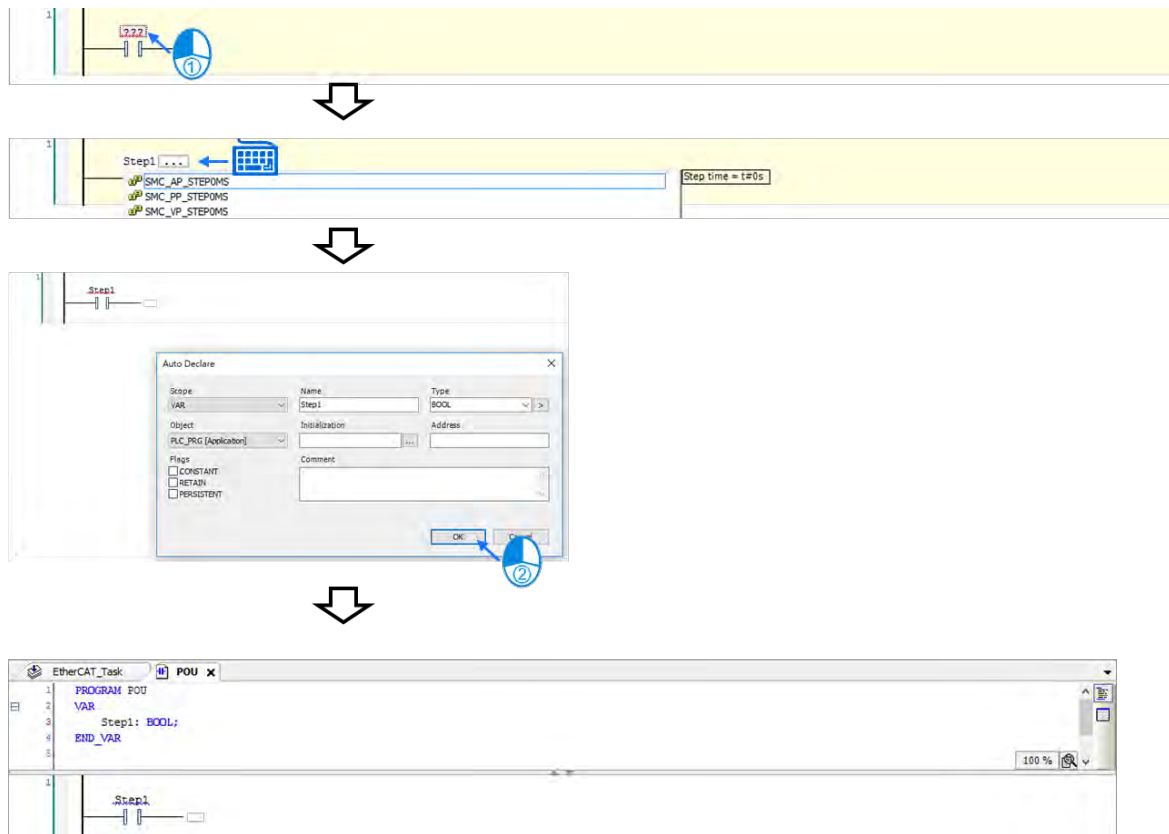
7. 使用 LD 语言编写程序，需新增指令可通过①添加，如需使用功能（Function）或者是功能块（Function Block）需通过②添加。



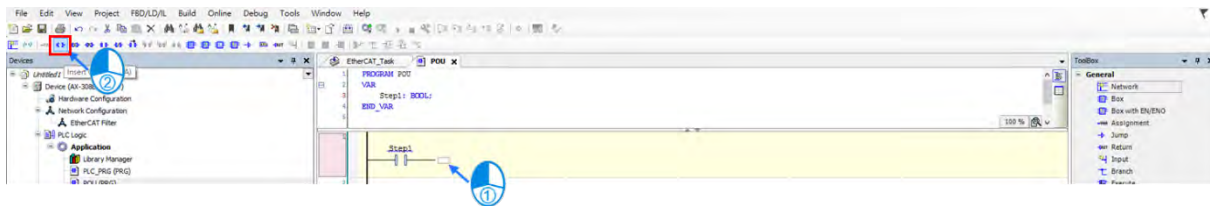
8. 在「POU」中单击欲新增指令的地方后，再单击  即可新增指令。



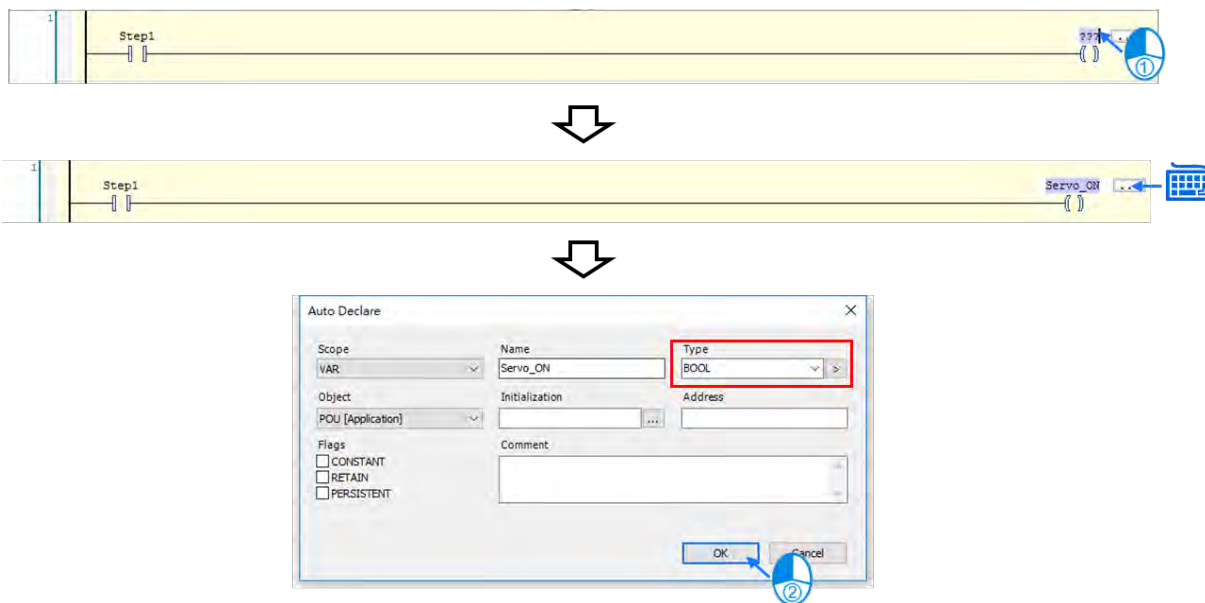
9. 「POU」中出现刚新增的指令后，单击「???」可输入指令变量名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



10. 再来新增输出指令，在 POU 中单击欲新增指令的地方后，再单击  即可新增指令。



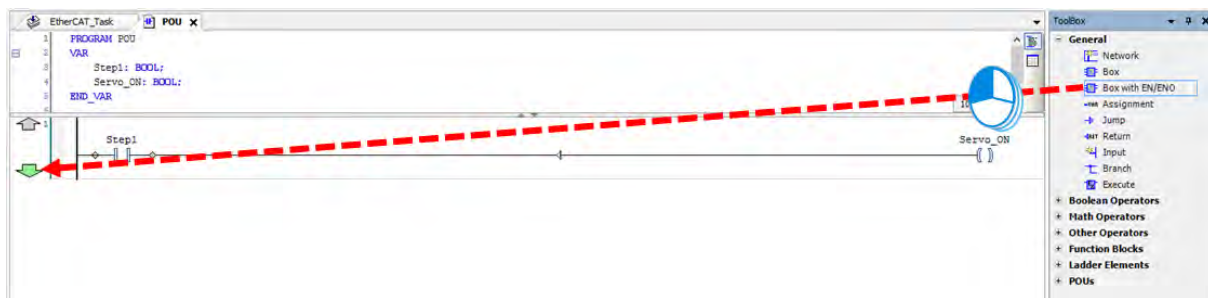
11. 单击「???」后可输入名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



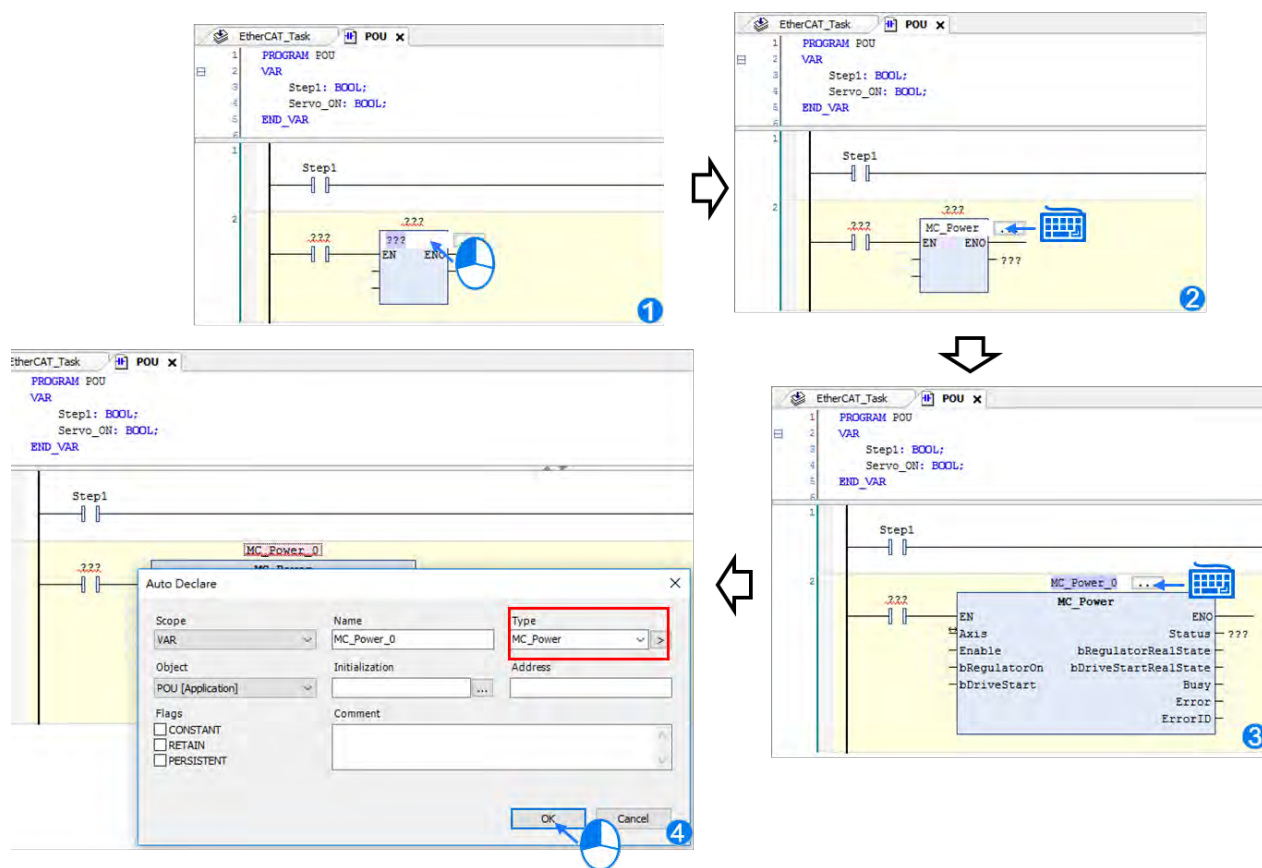
12. 先前新增的指令名称都会在上方显示名称与类型，也可在上方直接新增指令名称与类型。(后续范例中使用指令方式都是相同操作)



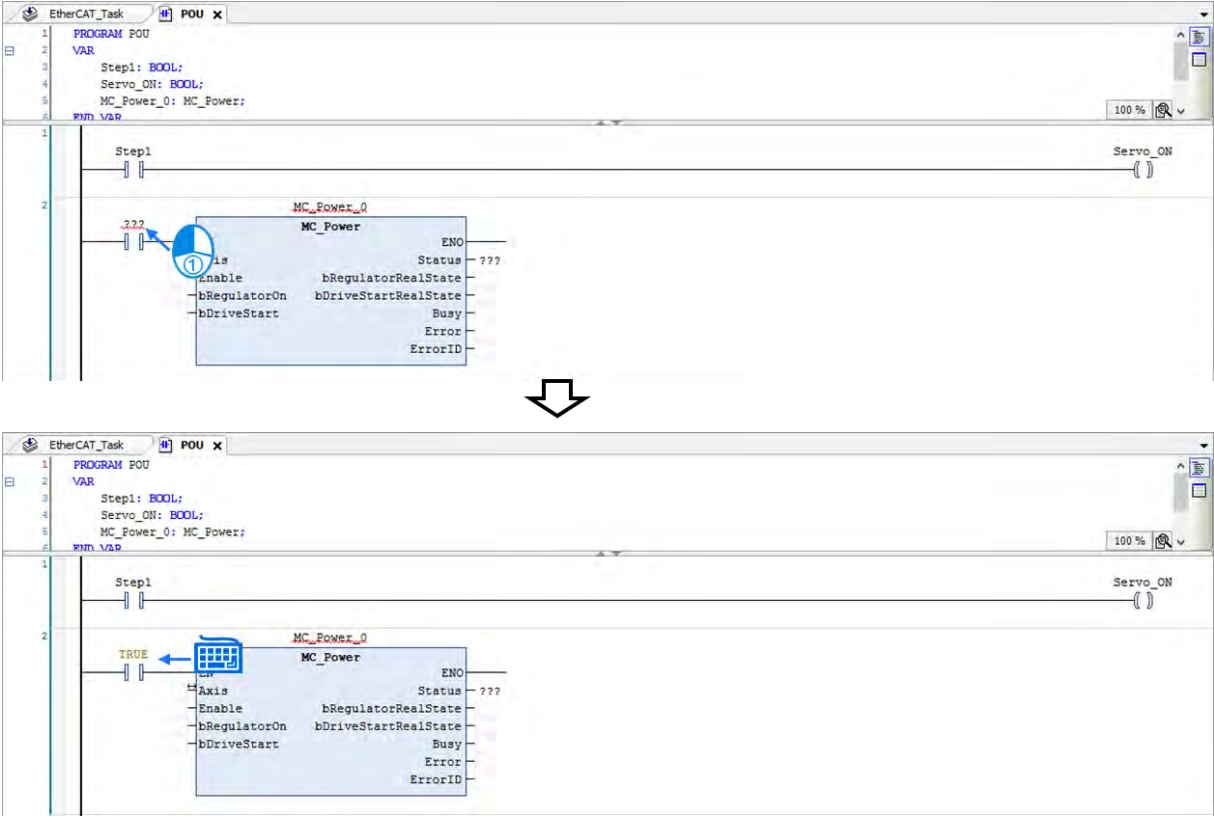
13. 再来新增功能块，单击「Box With EN / ENO」后拉至 POU 中的 。



14. 单击「???」后输入功能块名称，输入完成后按下 Enter 键，此时功能块就会出现，再输入功能块名称后按下 Enter 键，就会出现「Auto Declare」，确认功能块「Type」正确后单击「OK」。

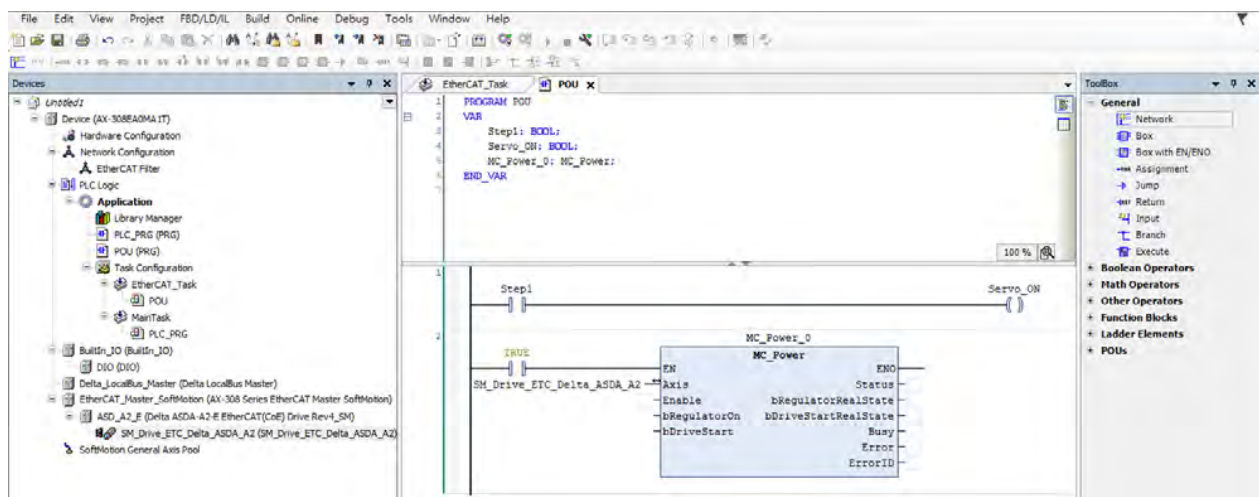


15. 在新增功能块，单击「???」输入 True 后单击 Enter 键。



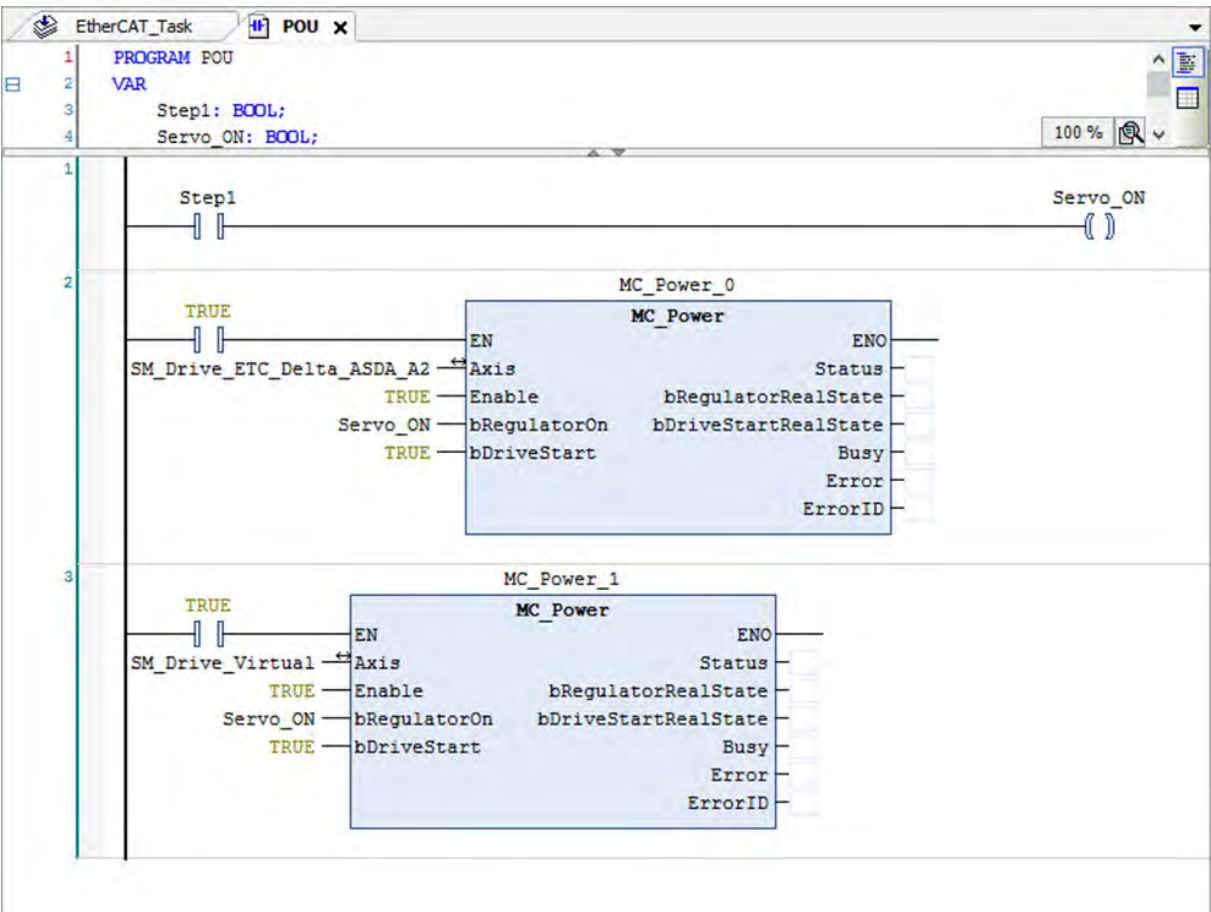
3

16. 将伺服名称输入至功能块中的 Axis 引脚中并删除引脚中的「???」，因「???」不为任何变数，在编译时会出错。

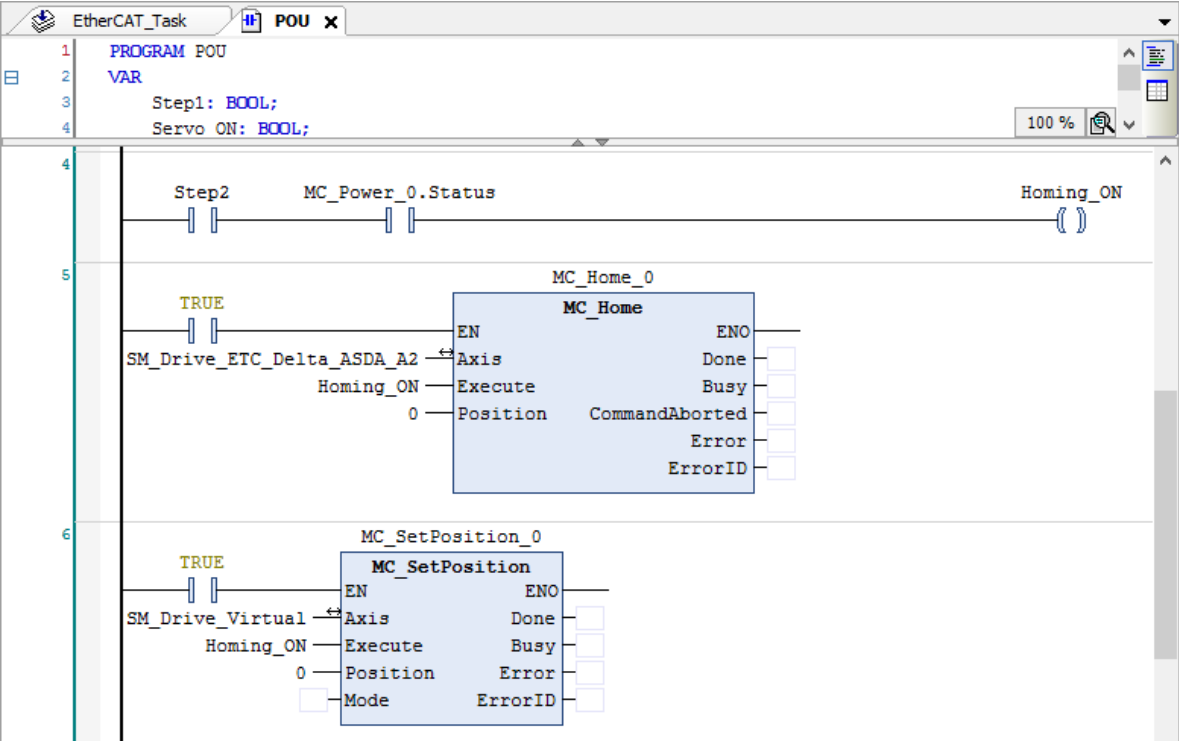


利用同样操作完成以下程序内容

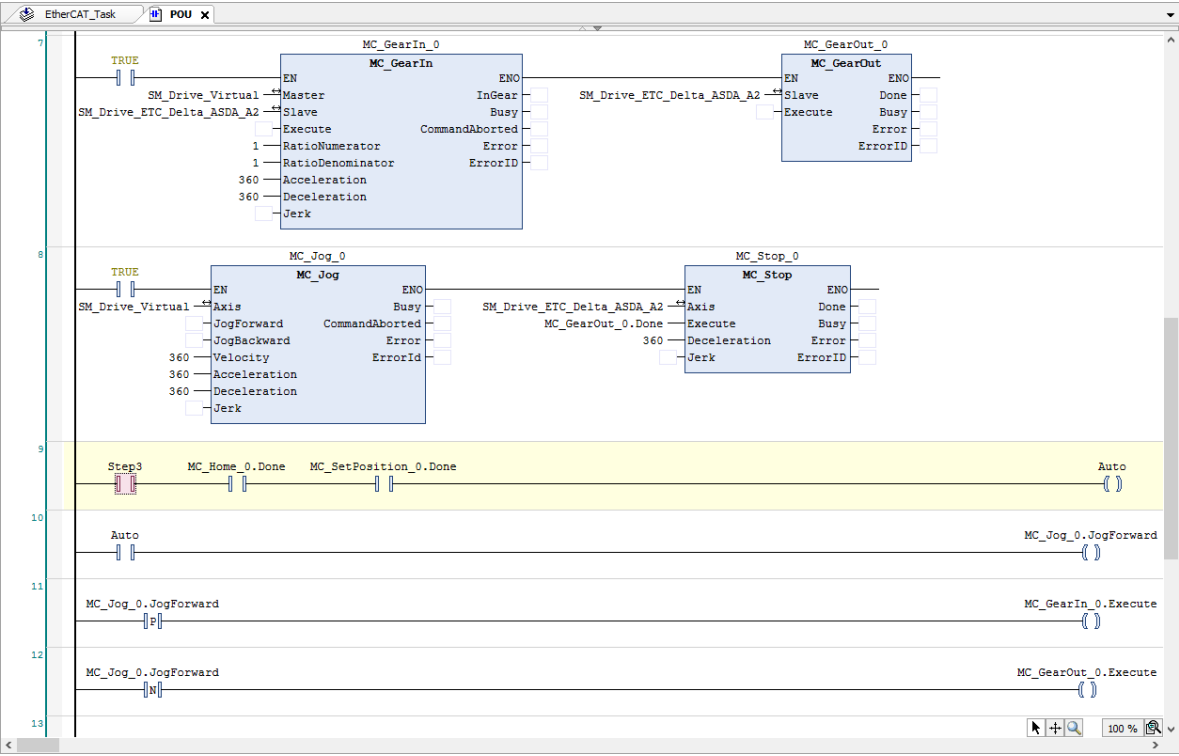
- 建立两轴「MC_Power」功能块



- 虚轴无法使用「MC_Home」功能块，只能使用「MC_SetPosition」功能块，并建立实轴回原点功能。



- 建立「MC_GearIn」实轴追随虚轴，并搭配「MC_Jog」触发虚轴运转，当关闭虚轴运转时，「MC_GearOut」会一起执行，当脱离追随后，从轴会等速运转，需使用「MC_Stop」停止运转。

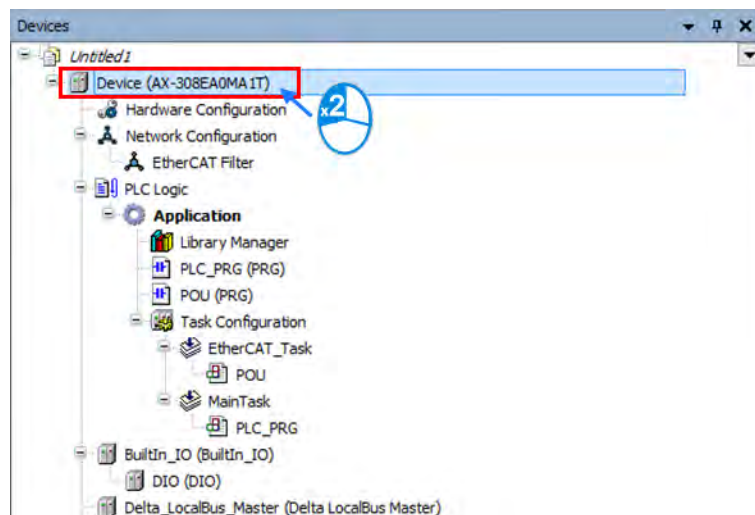


3.4.3 程序监控

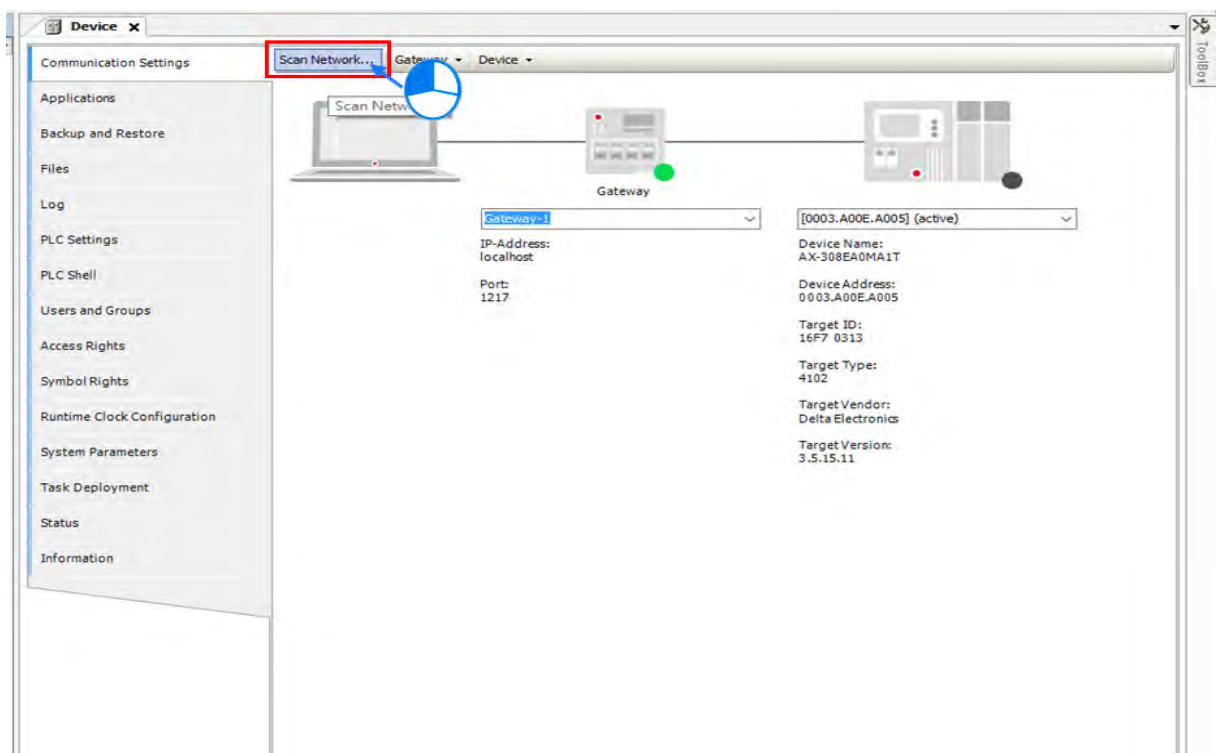
当程序运行的时候，用户可以通过监视的方式，来了解系统目前的逻辑控制状态，或是修改部分装置的数值来进行系统的测试，现在我们就来进行程序监视的操作。

3.4.3.1 设备与计算机链接设定

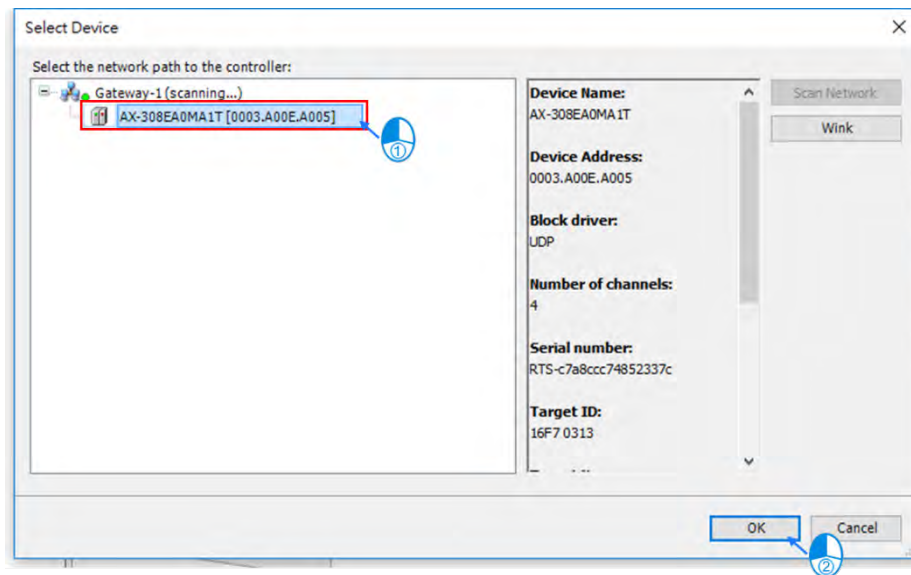
1. 单击「Device」后双击鼠标左键 (AX-308E 预设 IP 为 192.168.1.5)



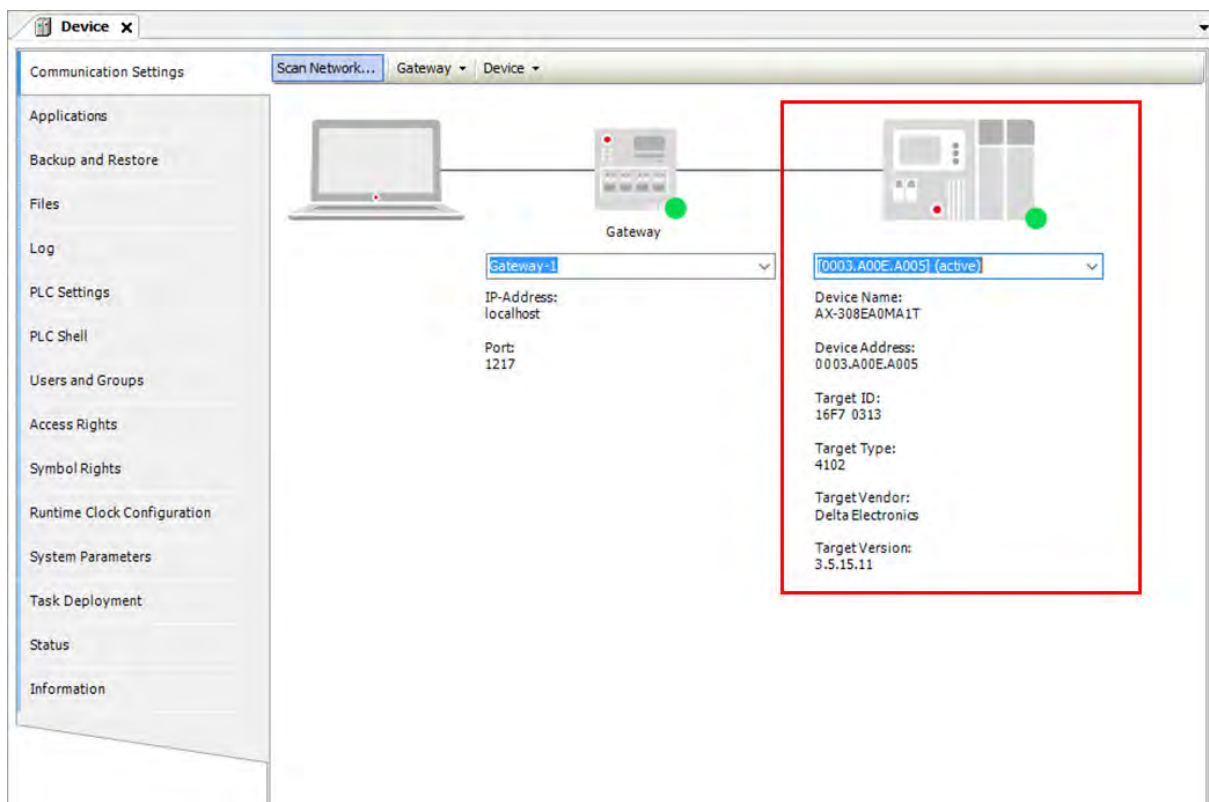
2. 进入「Device」画面后，选择「Scan Network」。



3. 在「Select Device」画面中，单击「AX-308E」 > 「OK」。

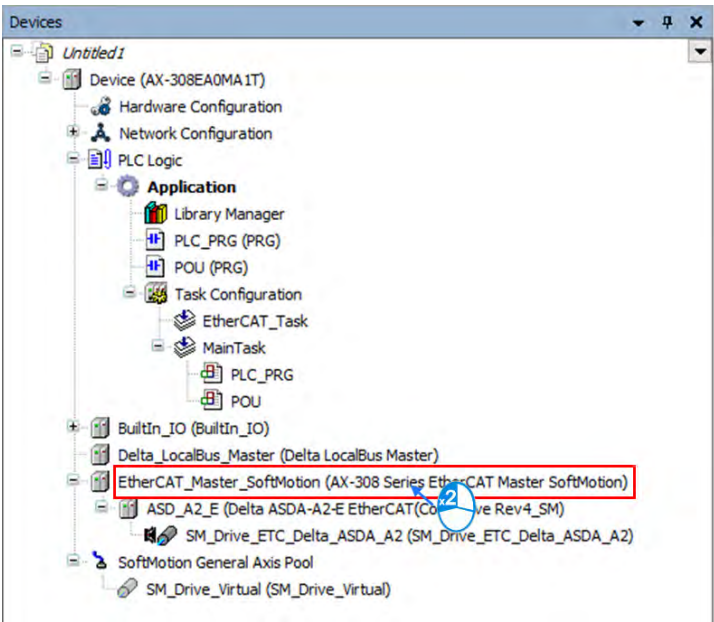


4. 当 PC 与设备连结后，会出现设备相关的信息，如下红框所示。

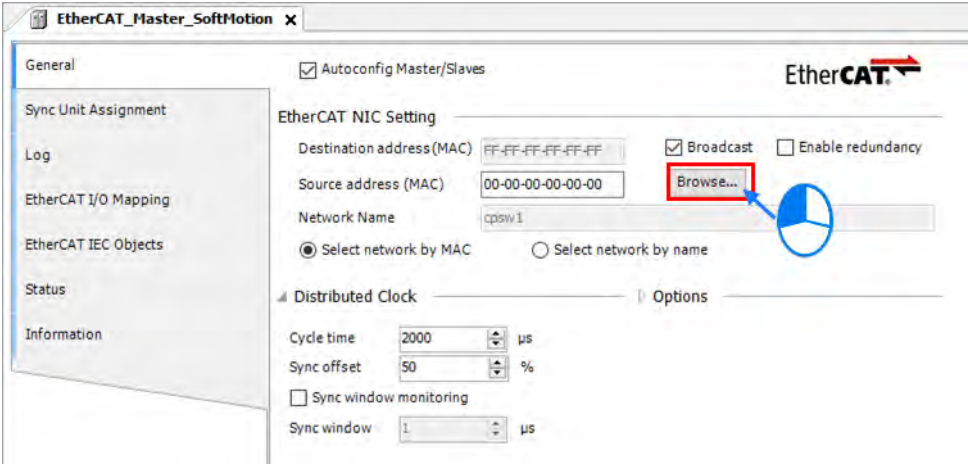


3.4.3.2 程序监控流程

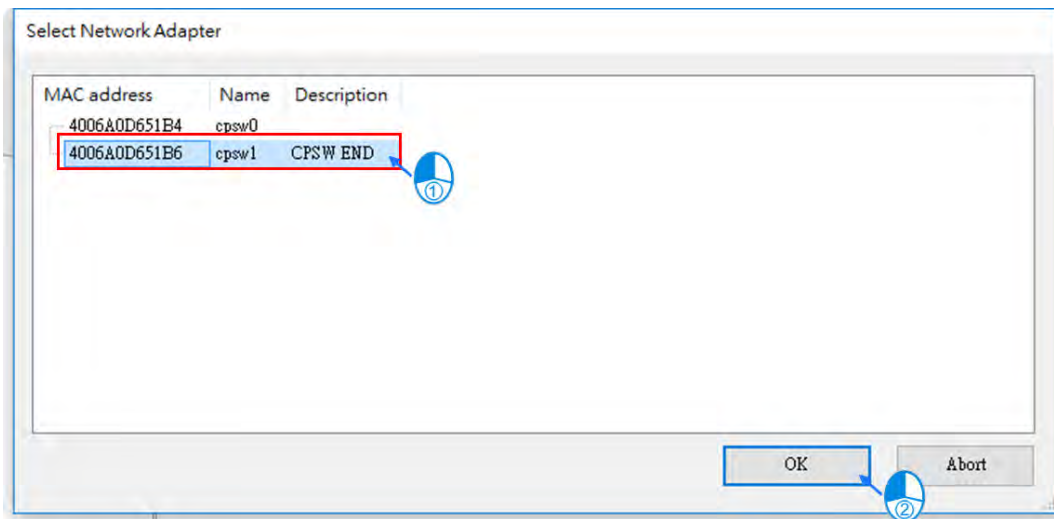
1. 监控程序前需确认 EtherCAT 通道是否建立正确，确认后再双击「EtherCAT_Master_SoftMotion」。



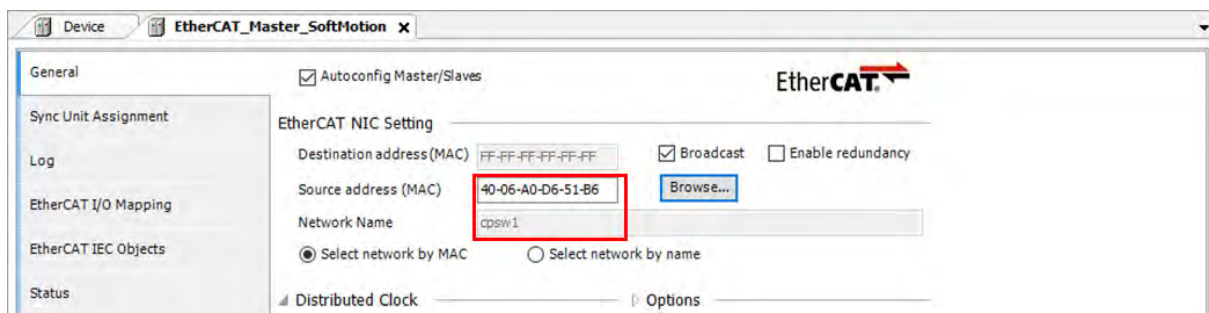
2. 进入「EtherCAT_Master_SoftMotion」画面后，单击「Browse」。



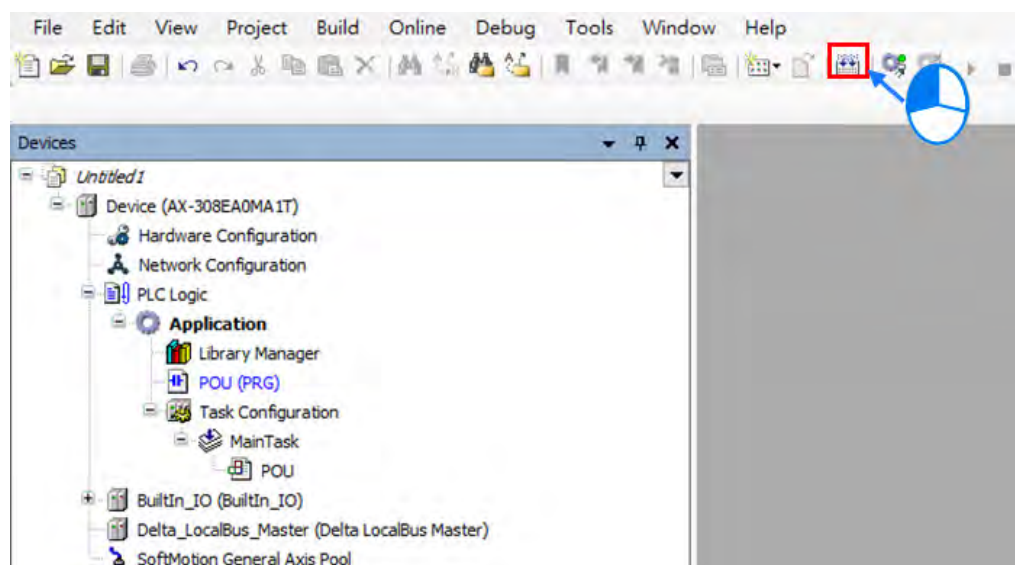
3. 需选择 EtherCAT 网络端口「cpsw1」后单击「OK」。



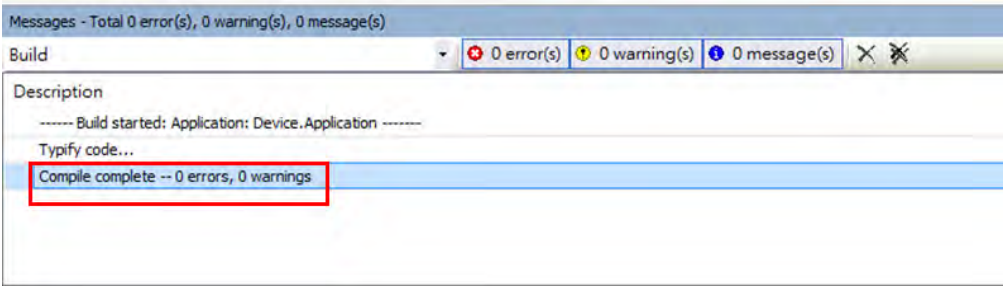
4. 选择完毕后，「MAC」会出现设定值。



5. 监控前先单击  编译以确认程序是否正确。

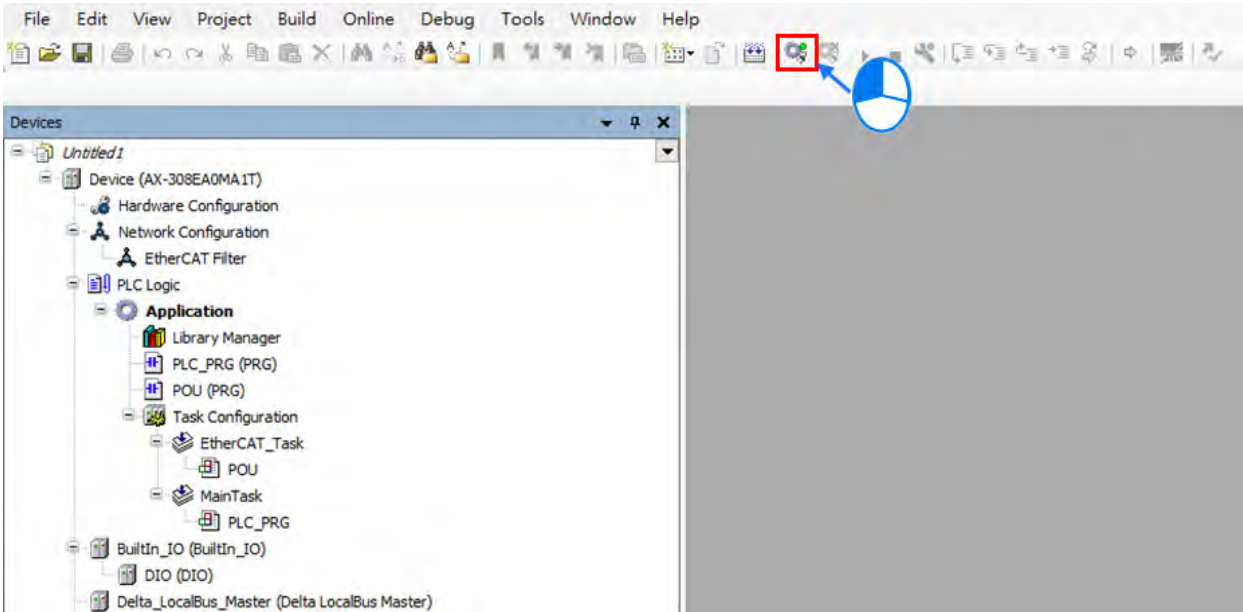


6. 编译完成后，可确认程序编写是否有错误。

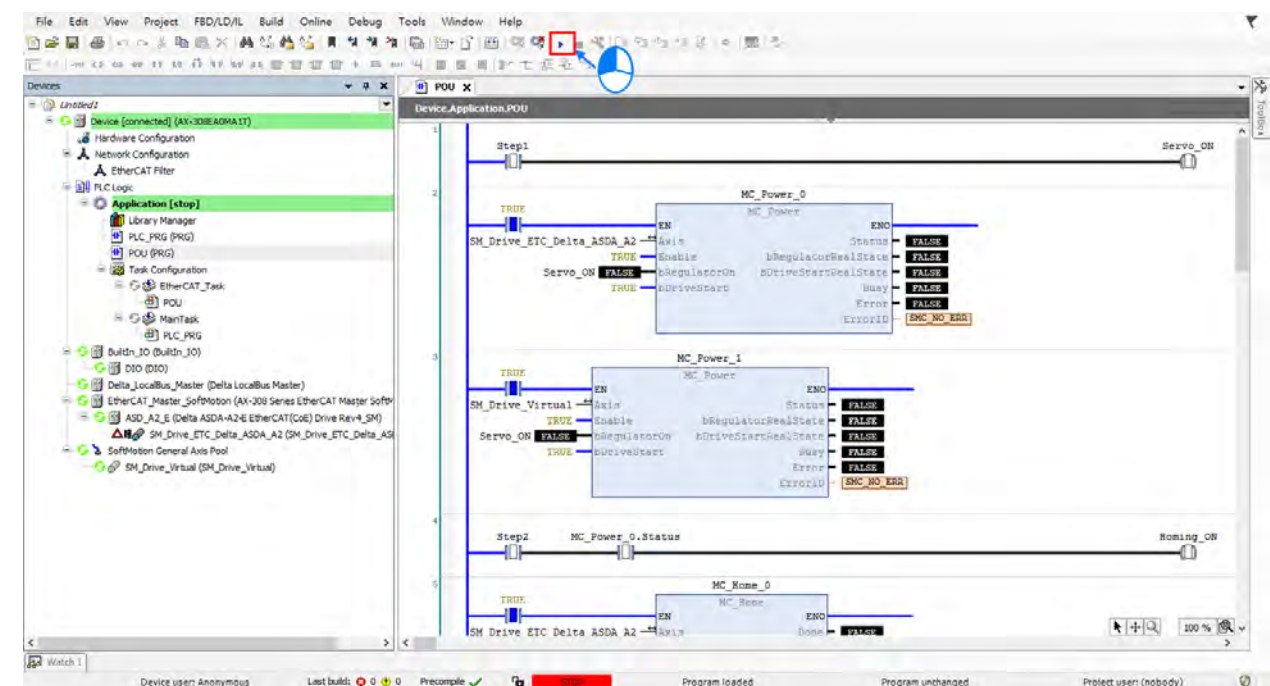


7. 当编译完成后，再单击  执行在线监控。

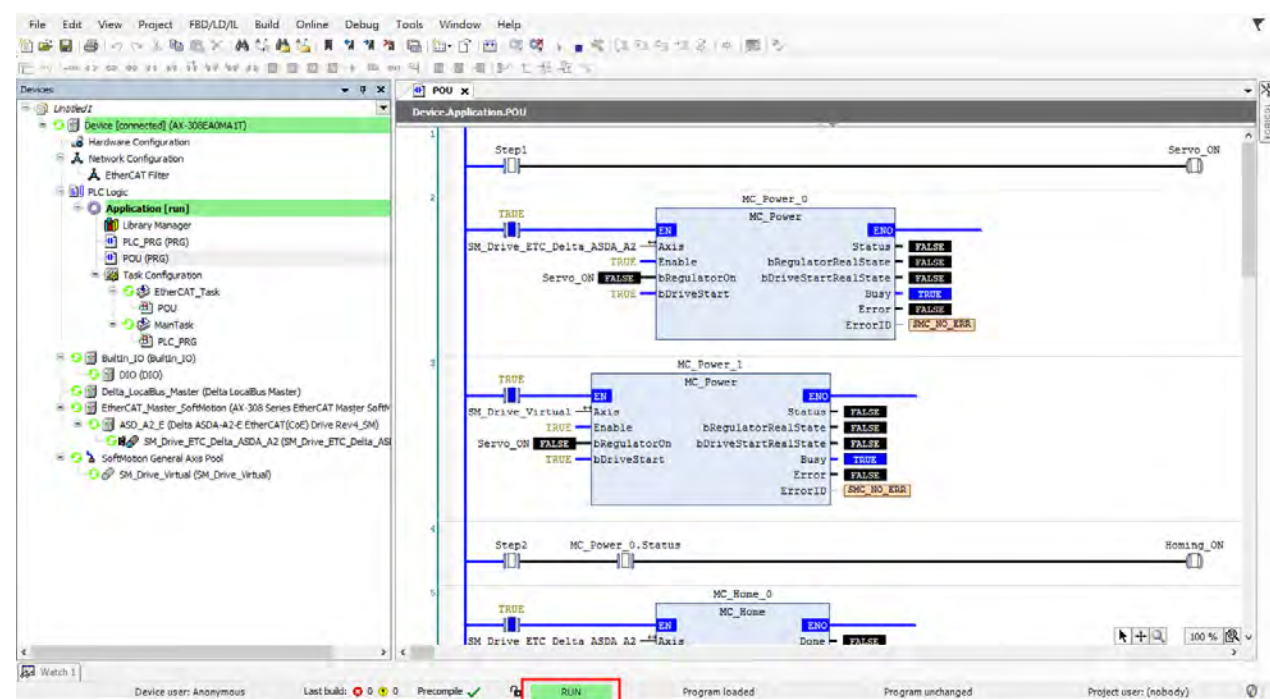
3



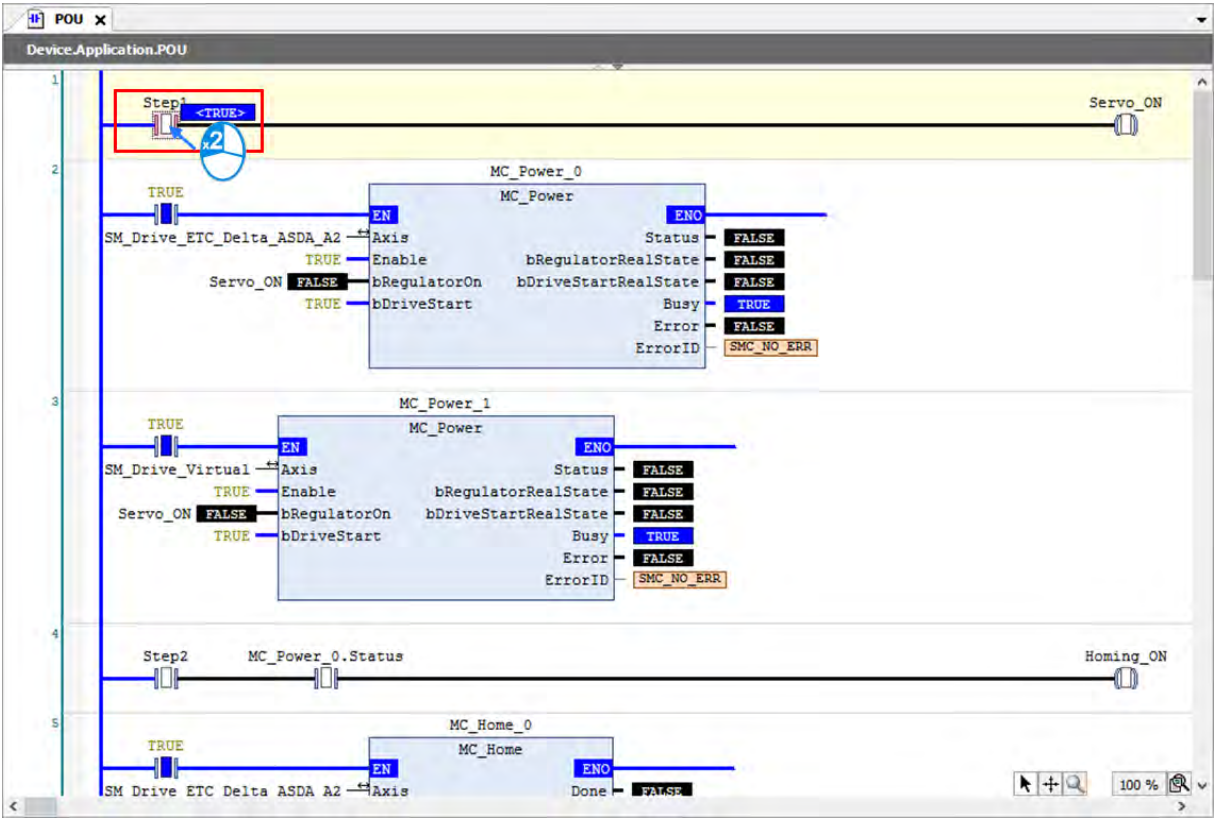
8. 执行在线监控后需下载程序，下载完成后 PLC 状态会再 Stop，需单击 ▶ PLC 才会执行 Run。



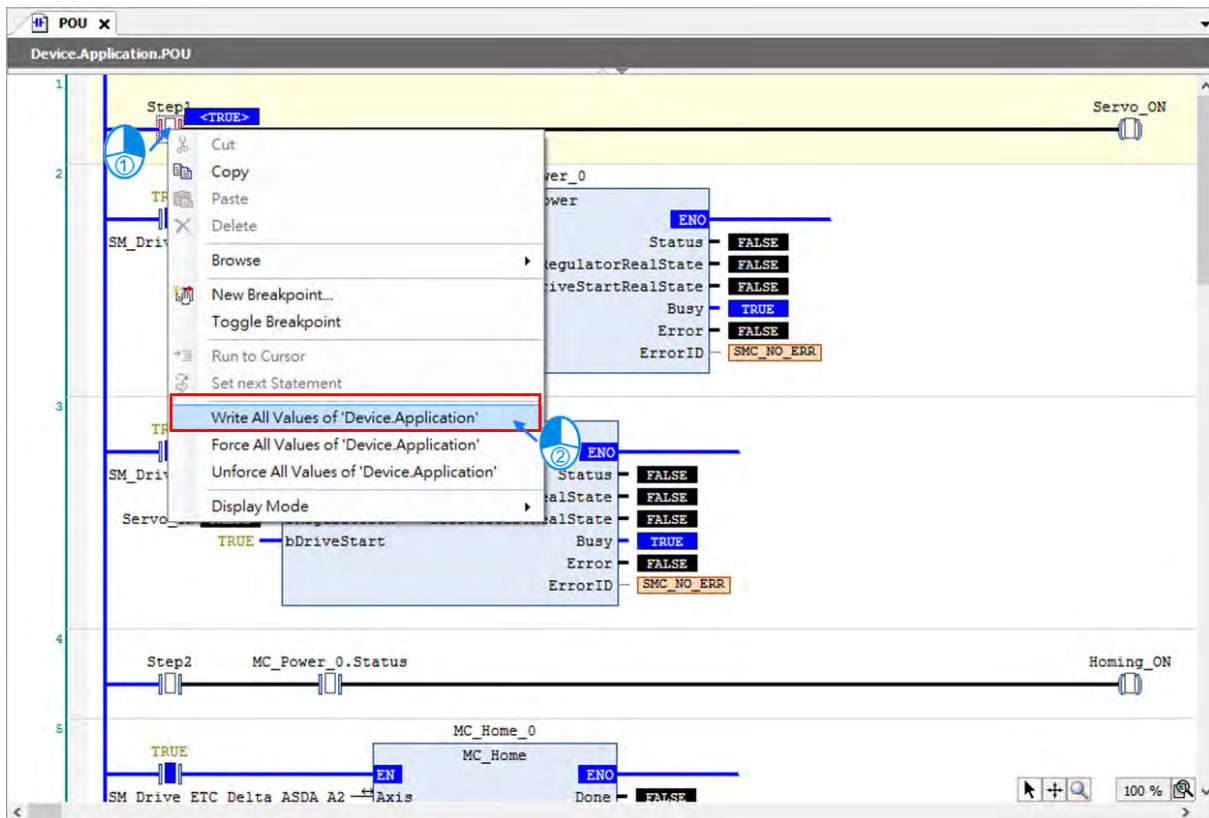
9. 当 PLC 开始 Run 时下方的状态会改为 **RUN**，此时可开始执行程序。



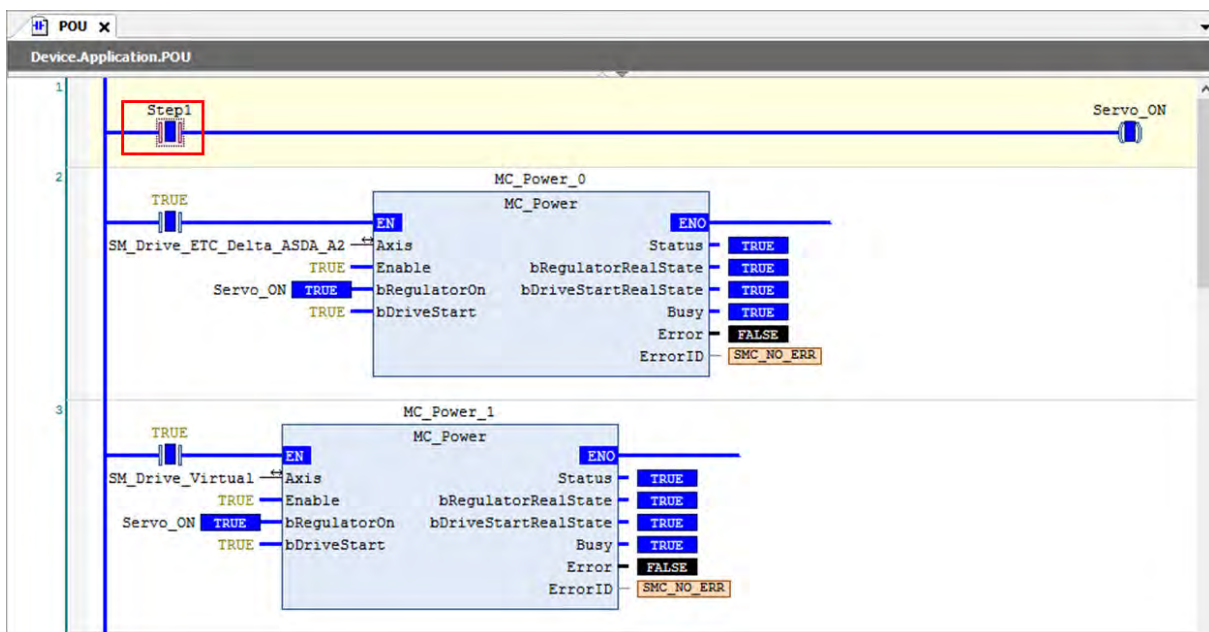
10. 在线监控后，双击「Step1」后会出现 **<TRUE>** 的图示。



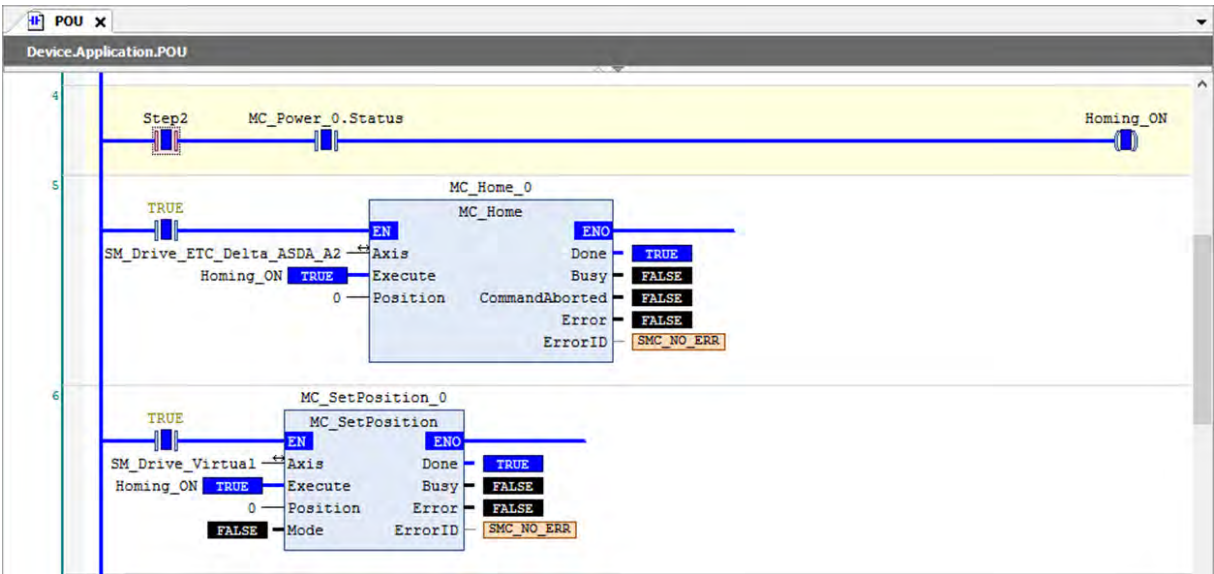
11. 单击「Step1」后右键单击后，选择「Write All Values of “Device.Application”」，会写入程序中所有状态写入。(可使用快捷键 Ctrl + F7)



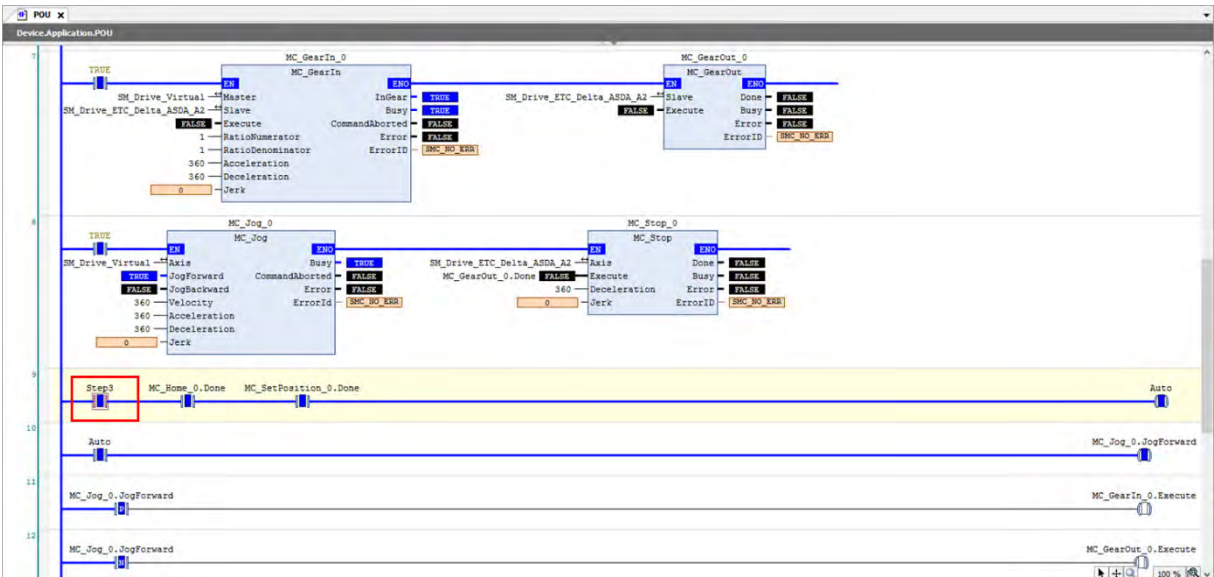
12. 启动「Step1」后就会 Servo ON。(以相同操作执行以下程序)



13. 启动「Step2」后使虚轴与实轴回原点

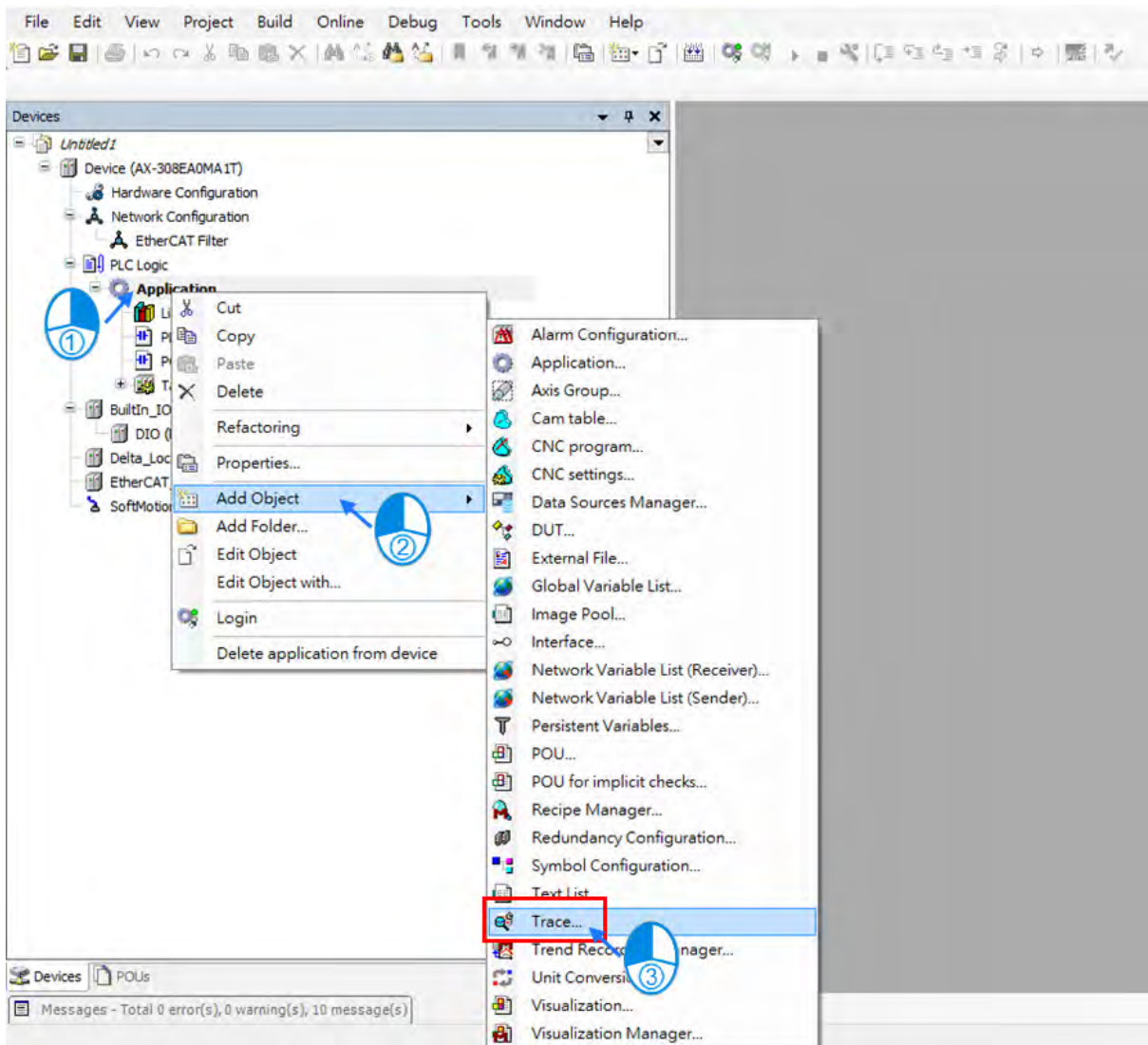


14. 启动「Step3」后，先触发「MC_Jog」让虚轴旋转，此时一起触发「MC_GearIn」从轴追踪动作。当「Step3」关闭接点「MC_Jog」与「MC_GearOut」会同时触发，虚轴停止转动，当脱离追踪时，从轴会等速运转后，再执行「MC_Stop」停止轴运转。

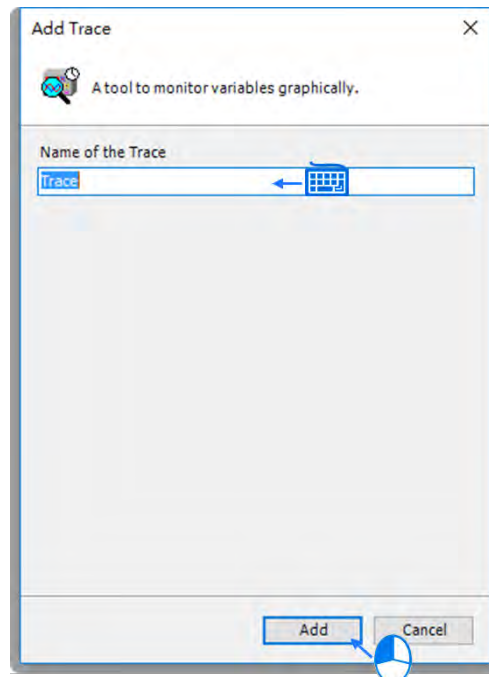


3.4.3.3 示波器监控

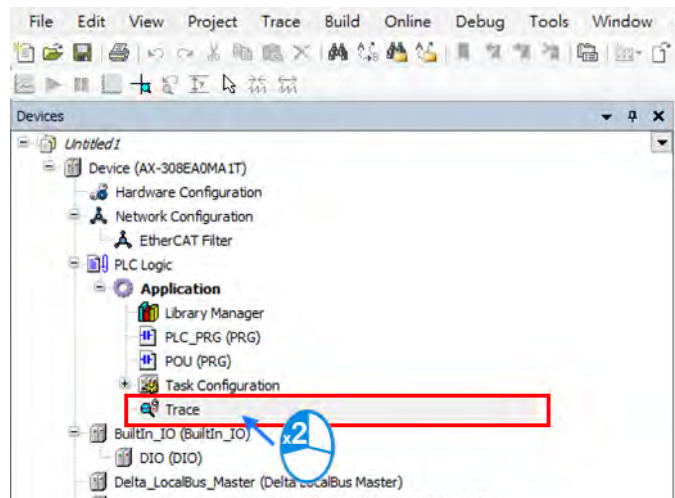
1. 当程序在运行过程中可以使用 Trace 监控的方式，确认引脚状态及轴状态。依序单击「Application」>「Add Object」>「Trace」。



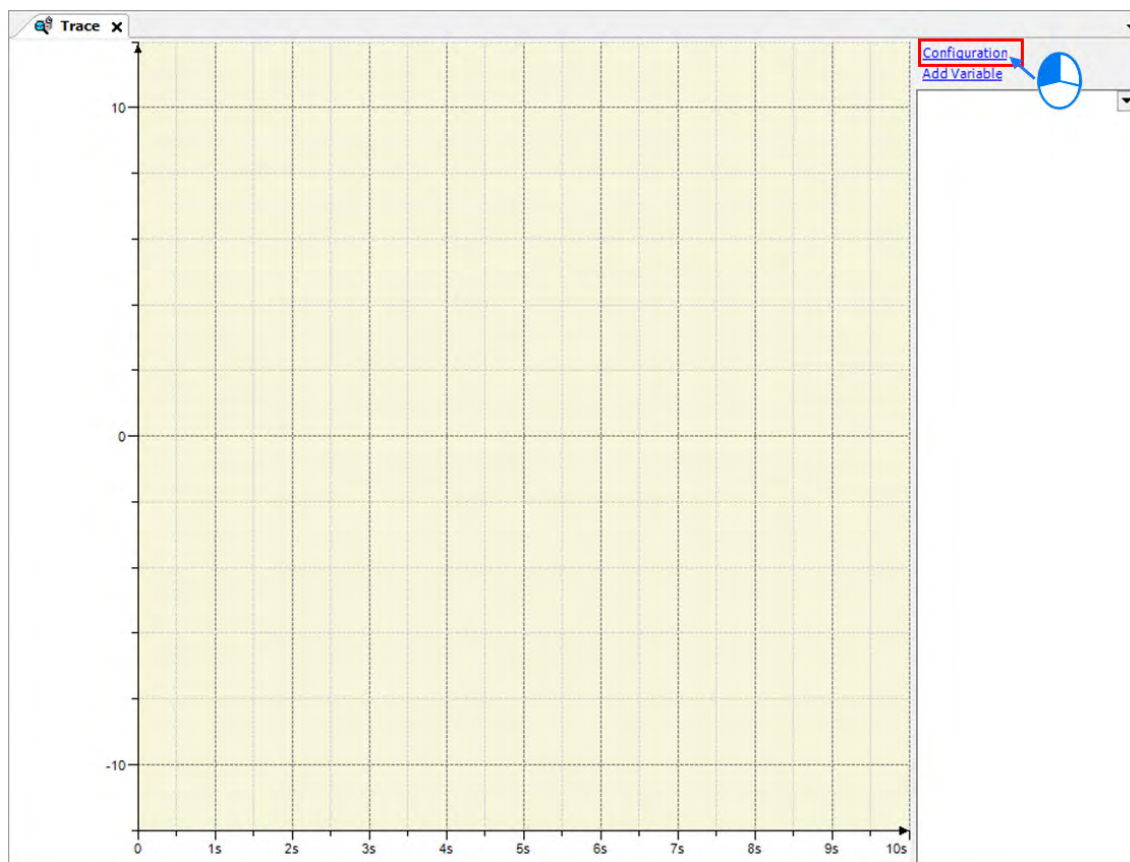
2. 进入「Add Trace」添加画面中，可以输入 Trace 的名称后再单击「Add」。



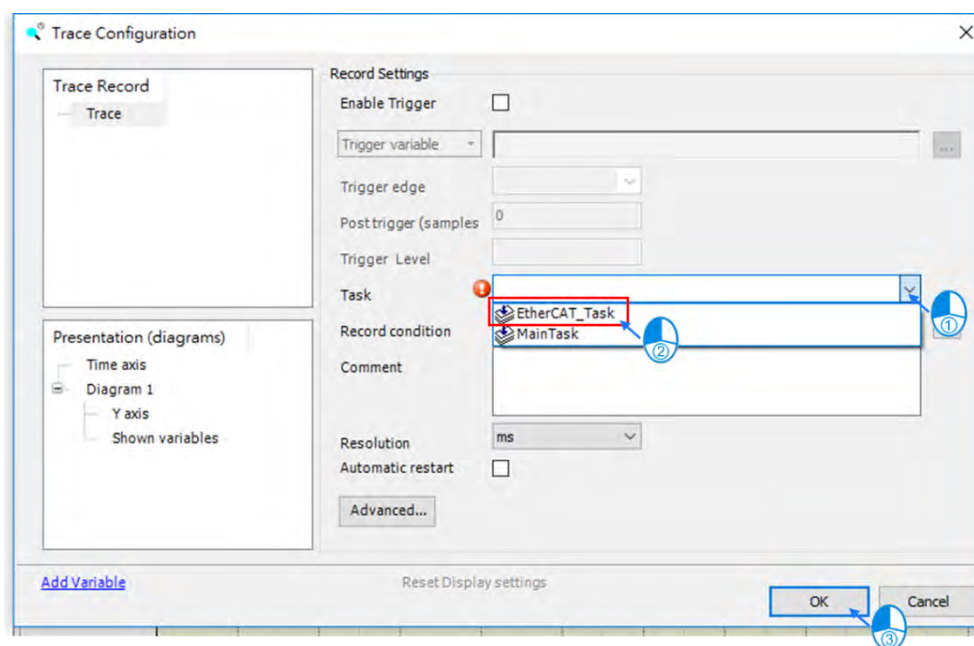
3. 新增完毕后左边树形图会出现刚刚所新增的示波器，此时双击「Trace」。



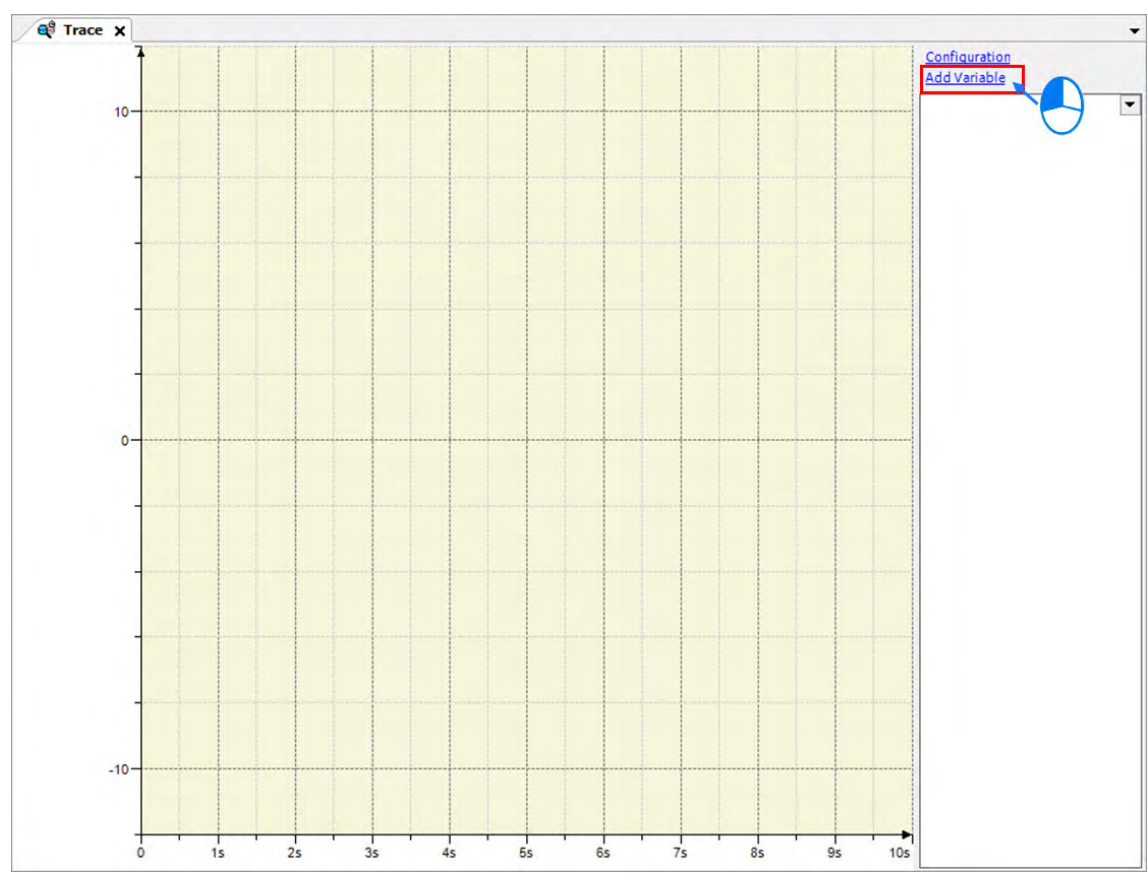
4. 进入「Trace」画面后，单击「Configuration」。



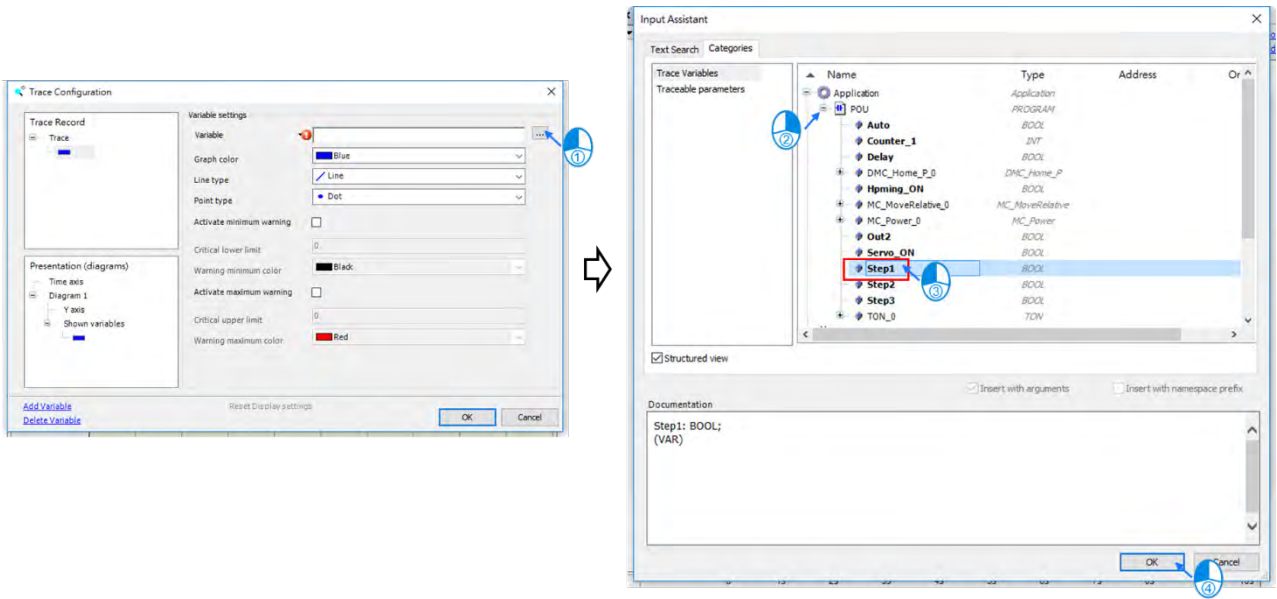
5. 在「Trace」设定画面中，选择「Task」下拉选单中的「EtherCAT_Task」。



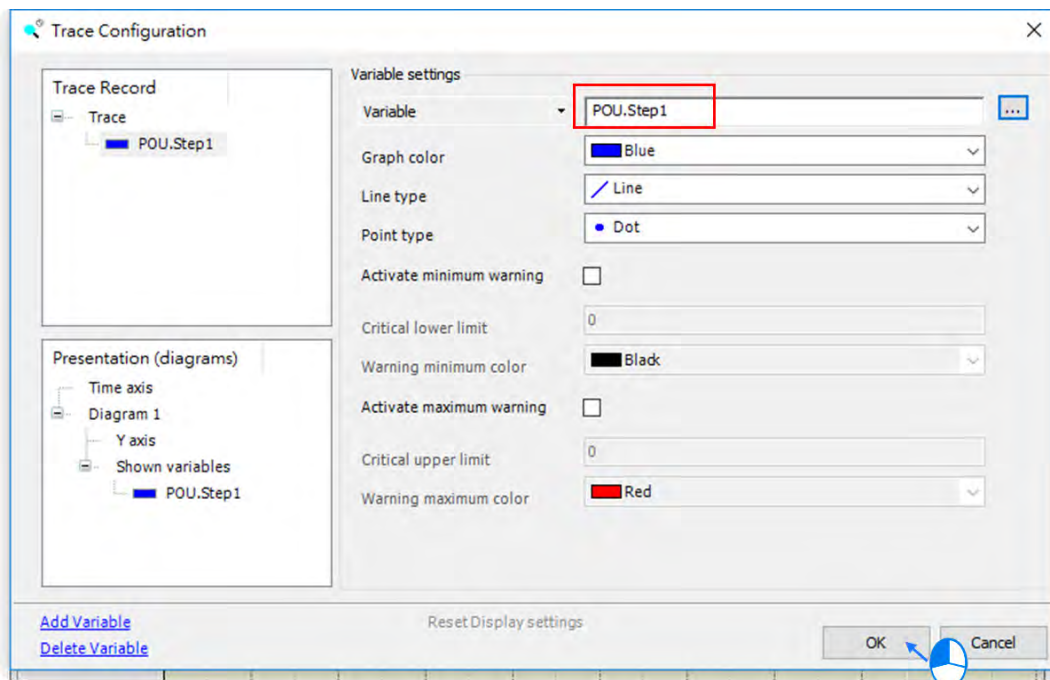
6. 设定完后单击「Add Variable」。



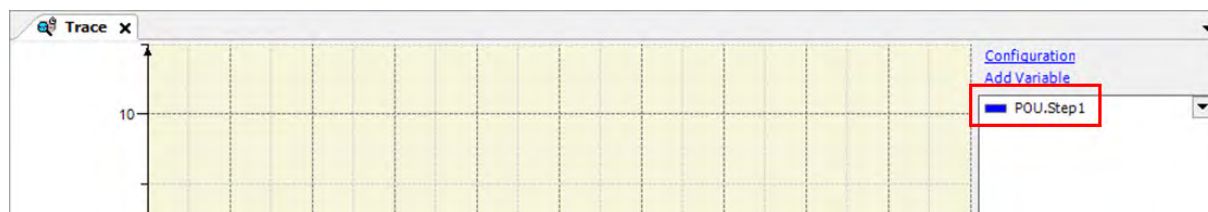
7. 进设定画面后，单击 添加需监控的参数，当进入添加画面后，单击 后选择「Step1」>「OK」。



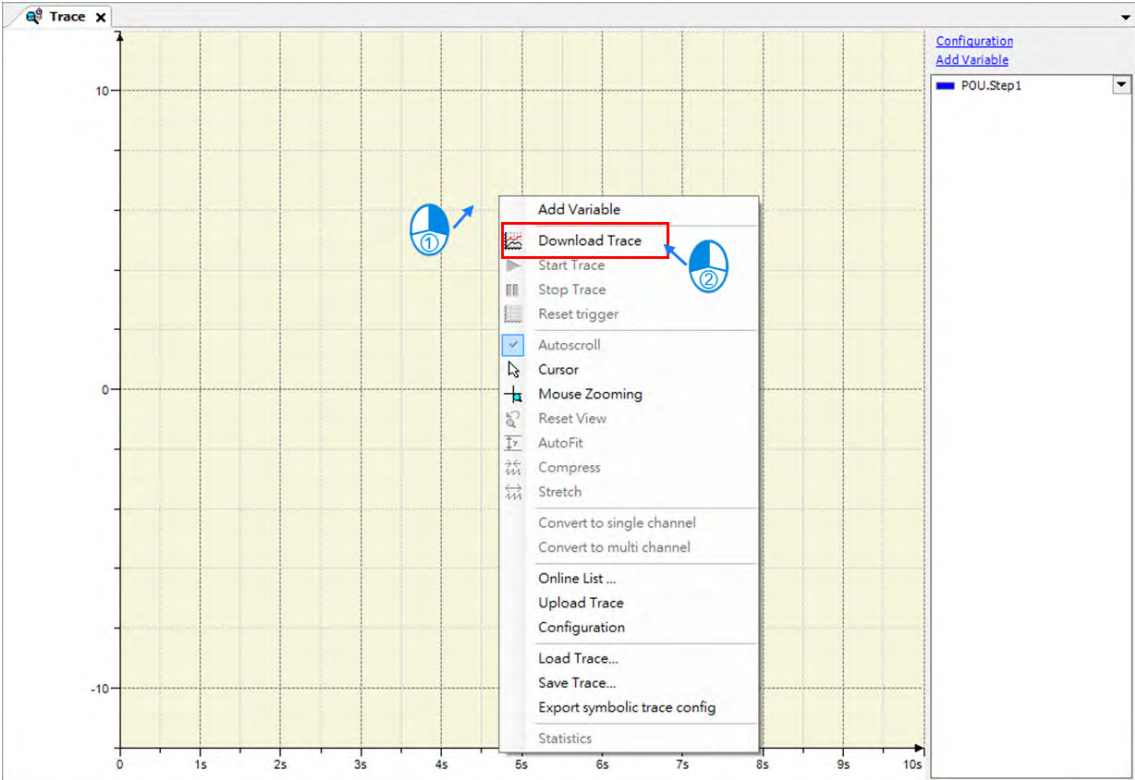
8. 选择好需监控的参数会出现在表格中再单击「OK」。



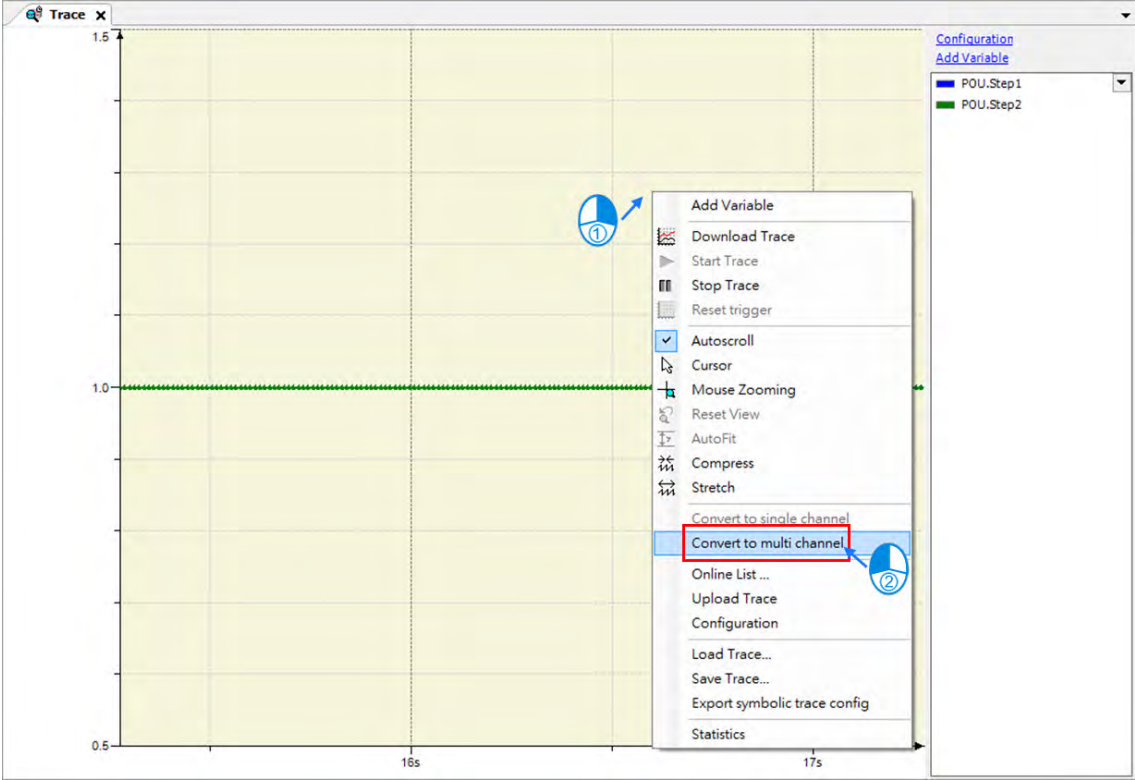
9. 添加完成后，在「Trace」画面右方会出现需监控的参数。



10. 在在线监控时单击「Trace」右键后，选择「Download Trace」就会开始监控。



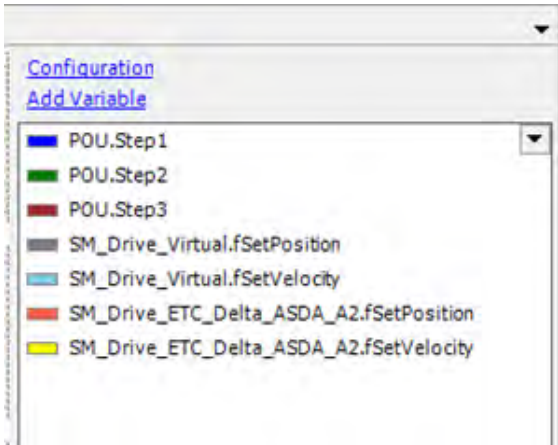
11. 当有两个参数需要监控时，可以在「Trace」画面单击右键，选择「Convert to multi channel」。



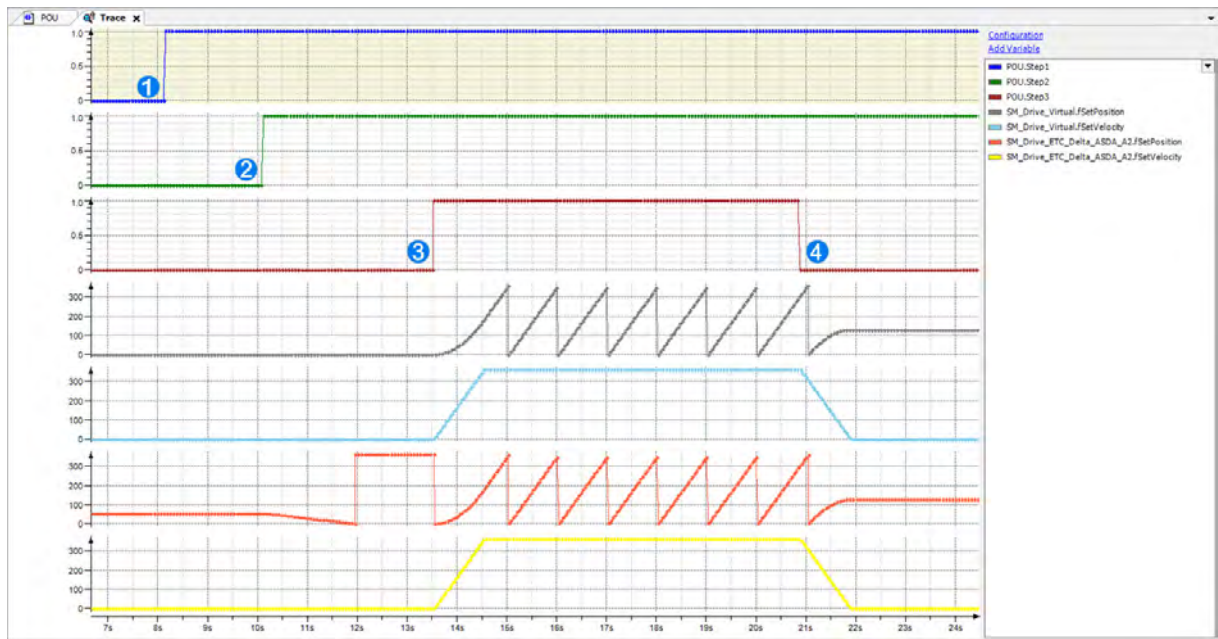
12. 「Convert to multi channel」选择后会以两格的方式显示。



13. 利用上方相同操作新增下列监控值。



14. 通过程序执行过程中，可使用示波器观察触发的开关及轴目前位置及速度。



示波器流程介绍：

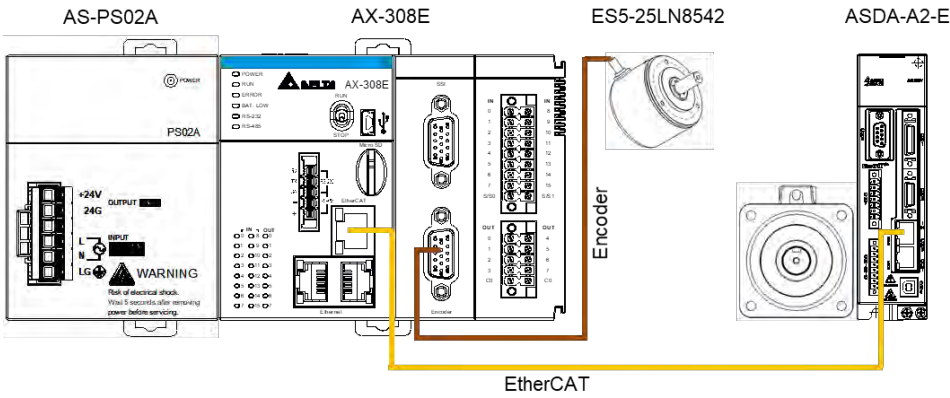
- ①启动「Step1」让实轴与虚轴启动。
- ②启动「Step2」让轴回原点。
- ③启动「Step3」后，「MC_Jog」启动主轴旋转，此时「MC_GearIn」也会同时启动，从轴就跟随主轴旋转。
- ④关闭「Step3」后，「MC_Jog」关闭主轴旋转，此时「MC_GearOut」也会同时启动，从轴就会等速运转，同时再触发「MC_Stop」让从轴停止运转。

3.5 内建编码器作为同步主轴项目建立

此范例是内建编码器轴为主轴搭配伺服电机跟随动作。

3.5.1 系统硬件配置

AX-308 通过 EtherCAT 端口口通讯连接 1 台伺服驱动器 (ASDA-A2-E)，控制伺服电机定位跟随虚轴。



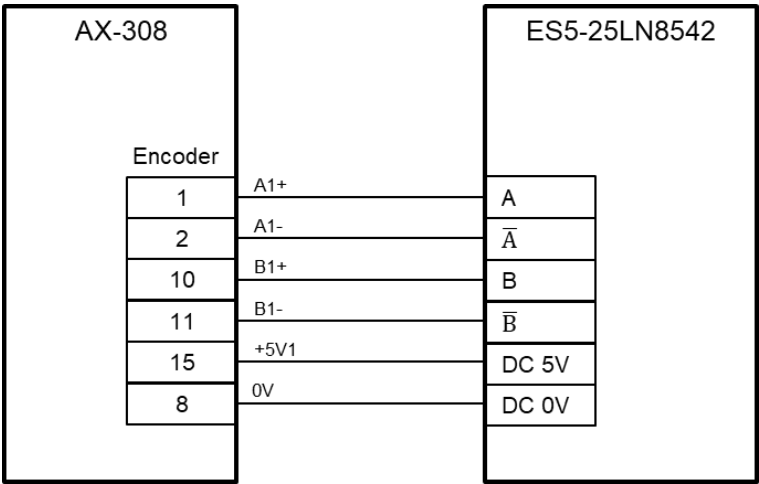
● 材料与工具

编号	名称	料号	数量
1	AS 系列电源模块	AS-PS02A	1
2	AX3 系列主机	AX-308EA0MA1T	1
3	编码器	ES5-25LN8542	1
4	电源供应器	DVP-PS02	1
5	通讯型伺服驱动器	ASDA-A2-0421-E	1
6	伺服电机	ECMA-C10604ES	1
7	伺服动力线	ASD-ABPW0003	1
8	伺服编码线	ASD-ABEN0003	1
9	EtherCAT 连接线(1 米)	UC-EMC010-02A	1

- 控制器配置编码器
 - ◆ 编码器功能介绍

编号	线材颜色	功能
1	Brown	5V
2	Blue	0V
3	Black	A
4	Black/Red	$\overline{A}$
5	White	B
6	White/Red	$\overline{B}$
7	Orange	Z
8	Orange/Red	$\overline{Z}$

- ◆ 编码器配置于控制器方法



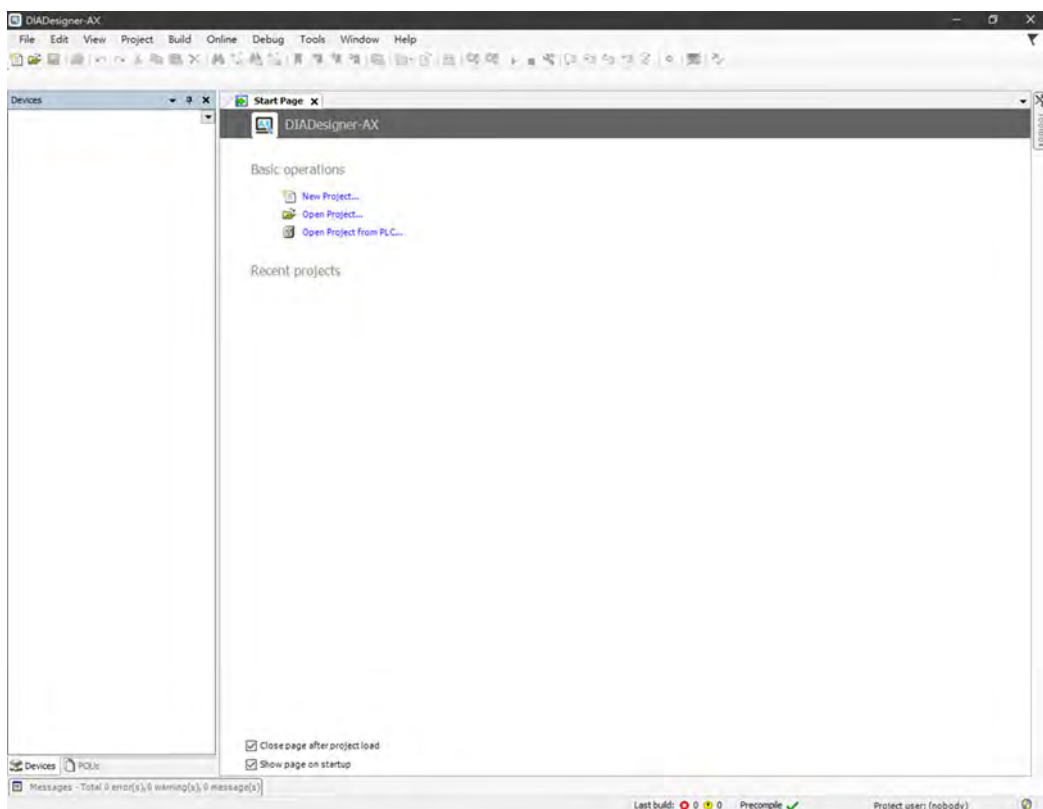
3.5.2 项目建立与程序编写

3.5.2.1 项目开启

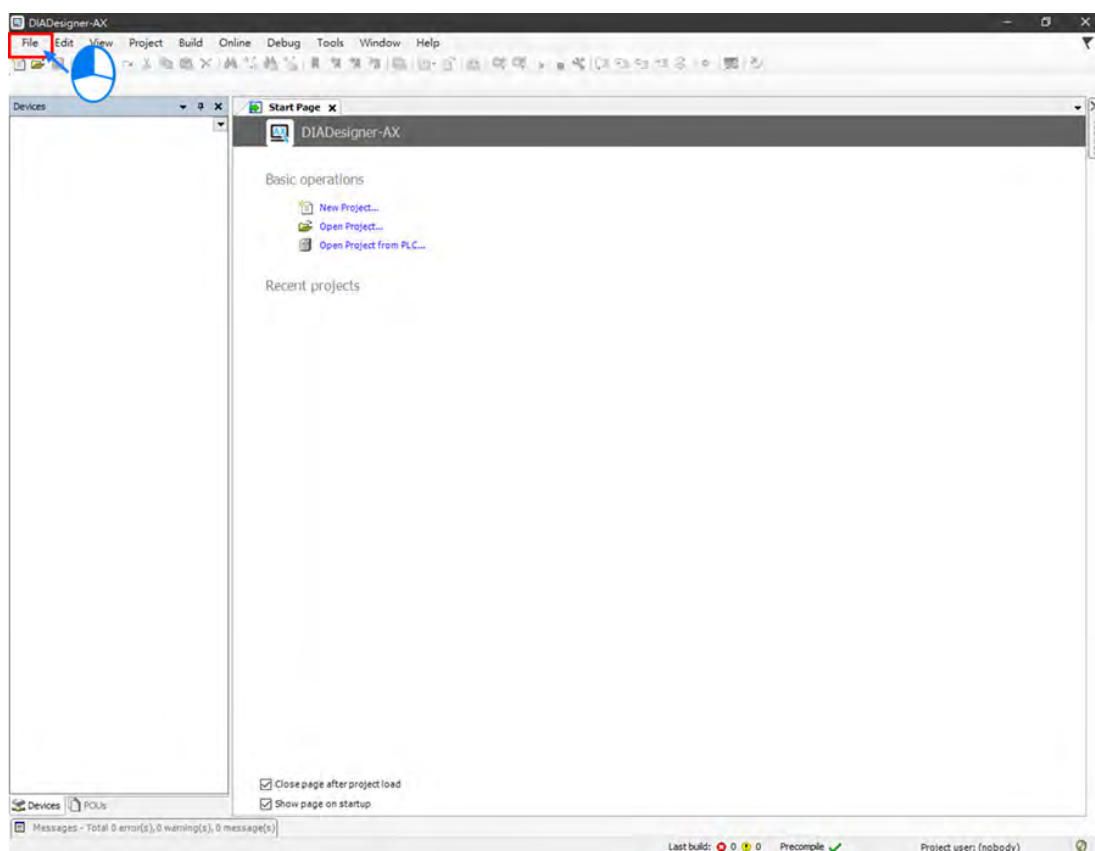
1. 首先 AX-308 编写程序软件为 DIADesigner-AX，所以需先启动软件。



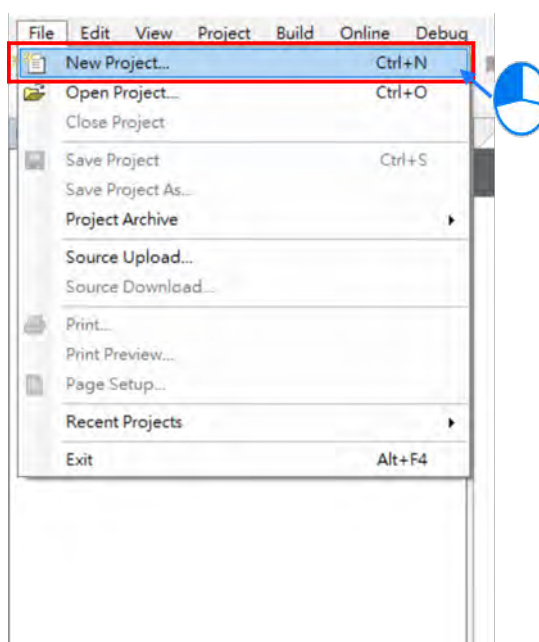
2. 启动 DIADesigner-AX 的画面如下



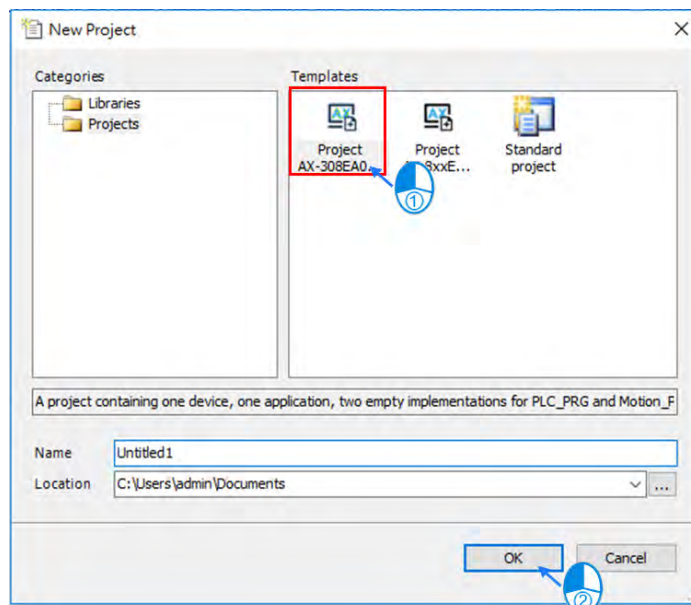
3. 单击「File」后双击鼠标左键



4. 在「File」下拉选单单击「New Project」以开启一个新项目。

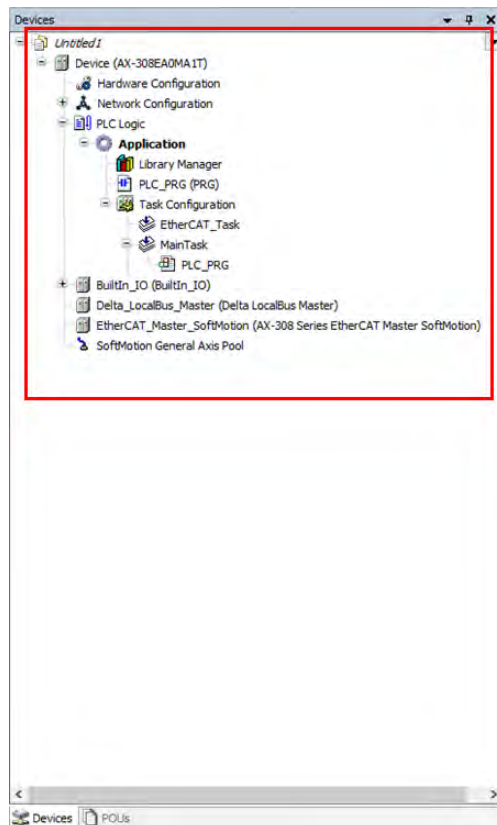


5. 进入「New Project」画面后选择「Project AX-308EA0MA1T*」，可在「Name」处输入项目名称，及「Location」处输入项目存取路径。

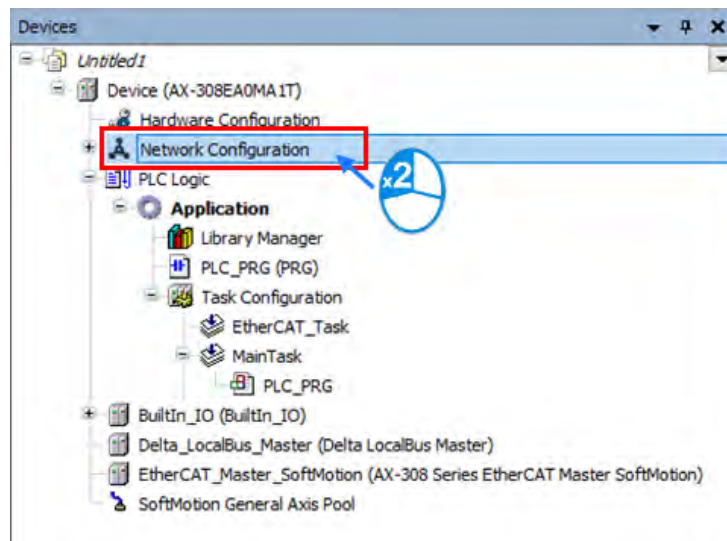


*注：DIADesigner-AX 版本于 V1.1.0 以上的操作为点选「Standard Project」后选择机种。

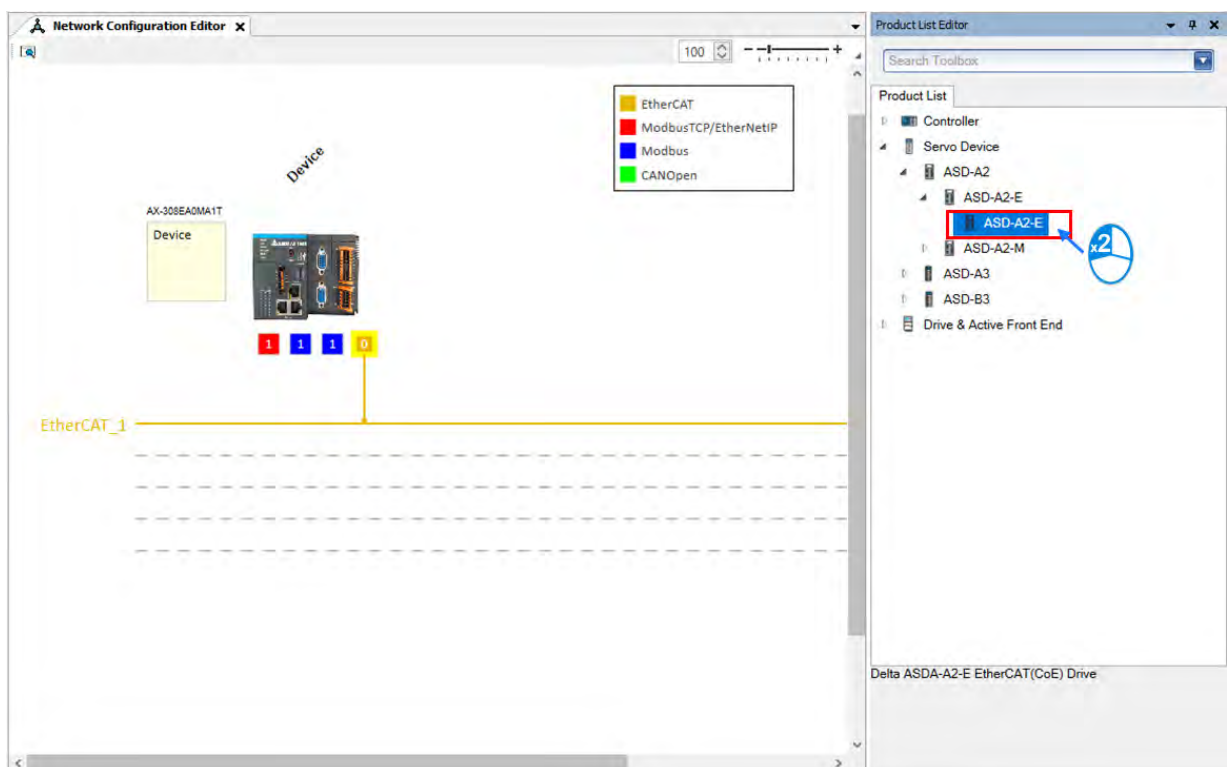
6. 进入项目画面如下，左边为项目树，可双击左键以开启程序画面或是模块相关设定。



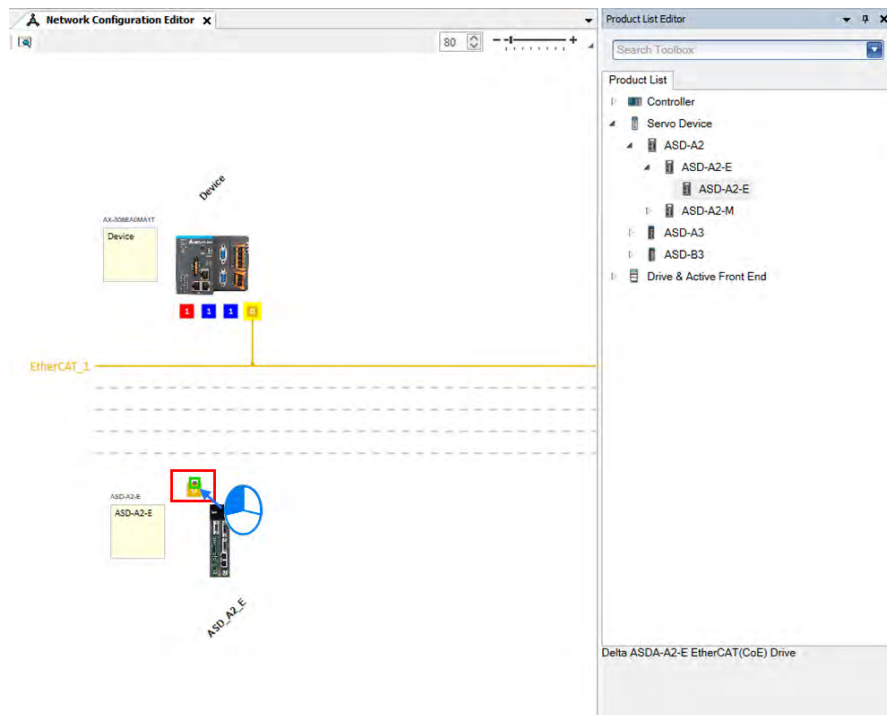
7. 先建立系统架构的伺服轴，双击「Network Configuration」。



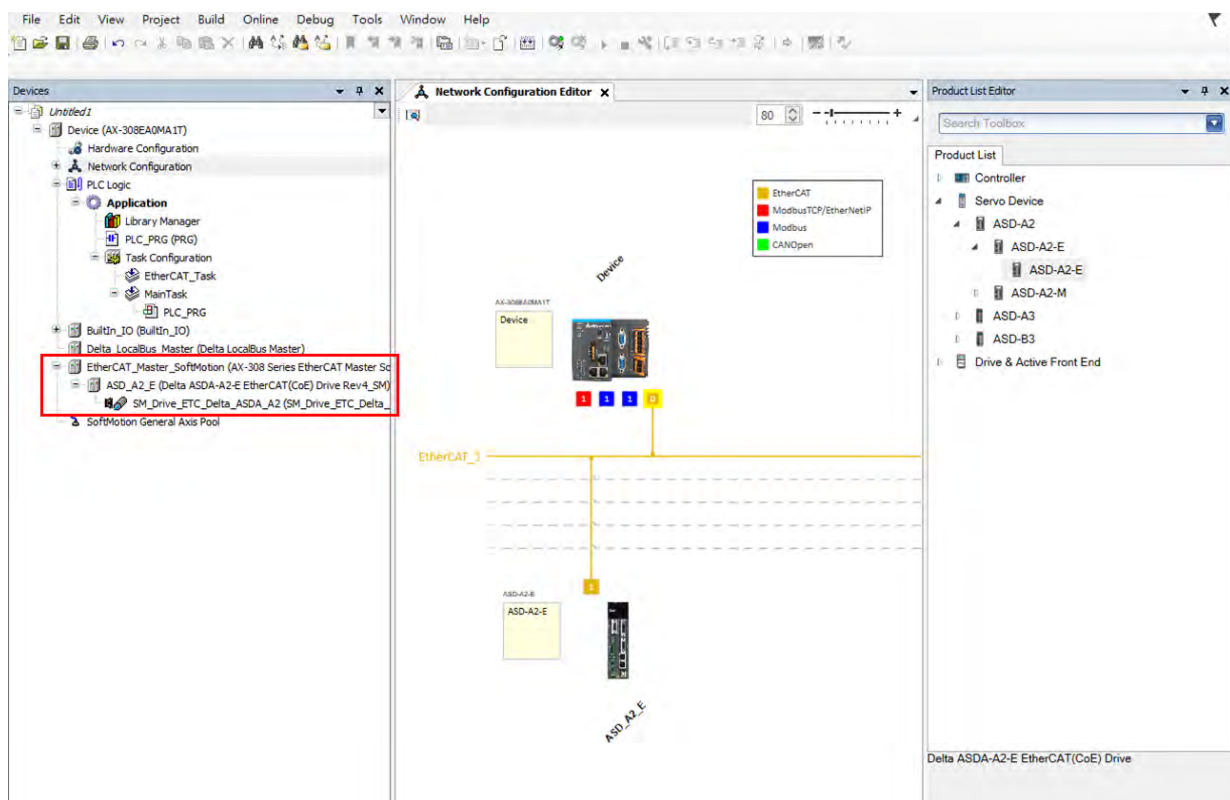
8. 进入「Network Configuration」画面后，选择 EtherCAT 所配置的 Slave 型号，双击左键以新增轴。



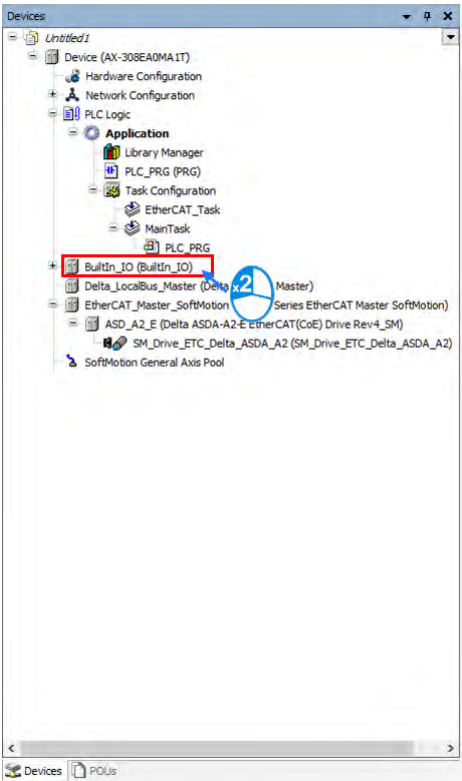
9. 将鼠标移至 **0** 上方，按压左键将黄色线连接至上方 EtherCAT_1。



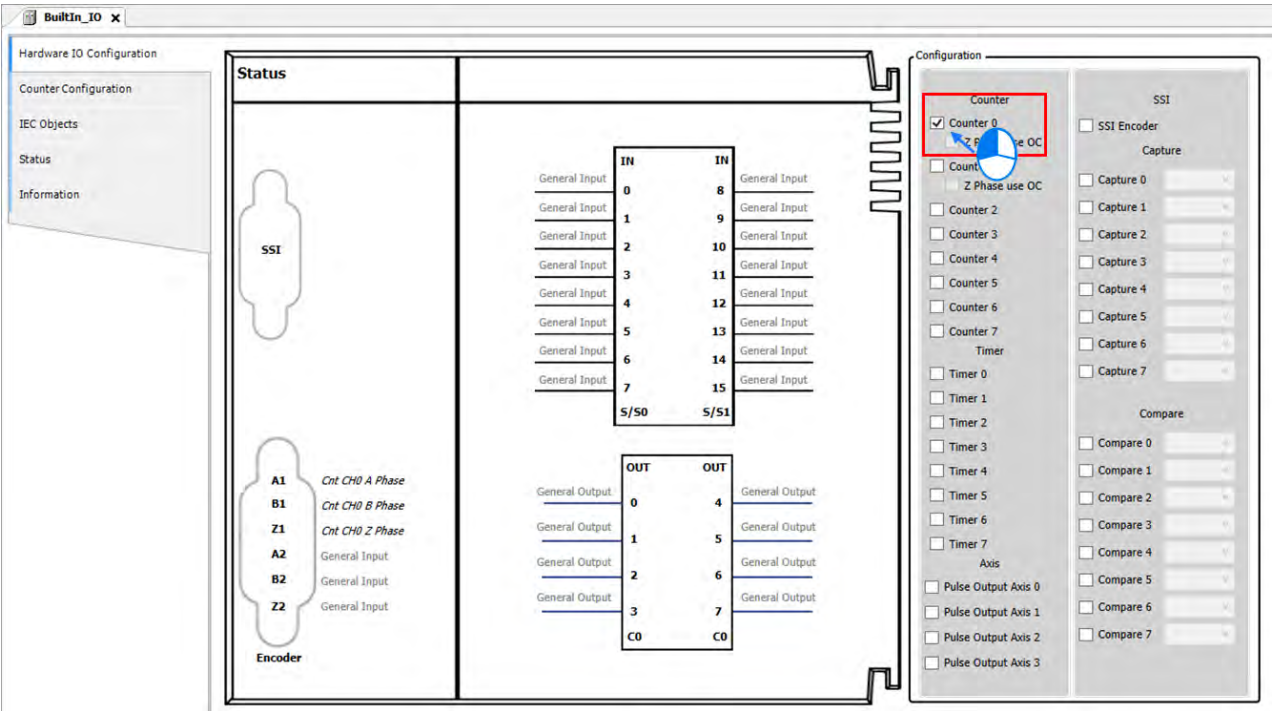
10. 将 Slave 都联机至 EtherCAT_1 后，左方树形图才会显示 Slave，代表此 Slave 有配置在 EtherCAT 通讯中。



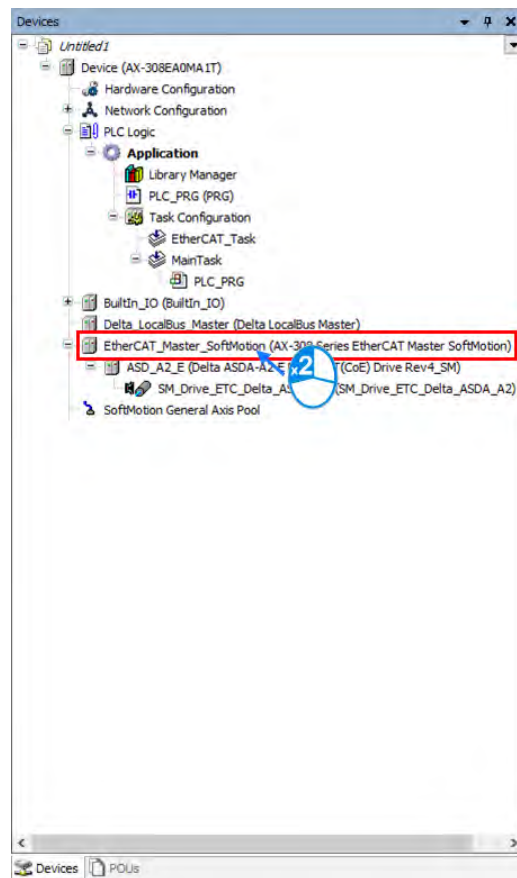
11. EtherCAT 轴建立完毕后，再单击「BuiltIn_IO」。



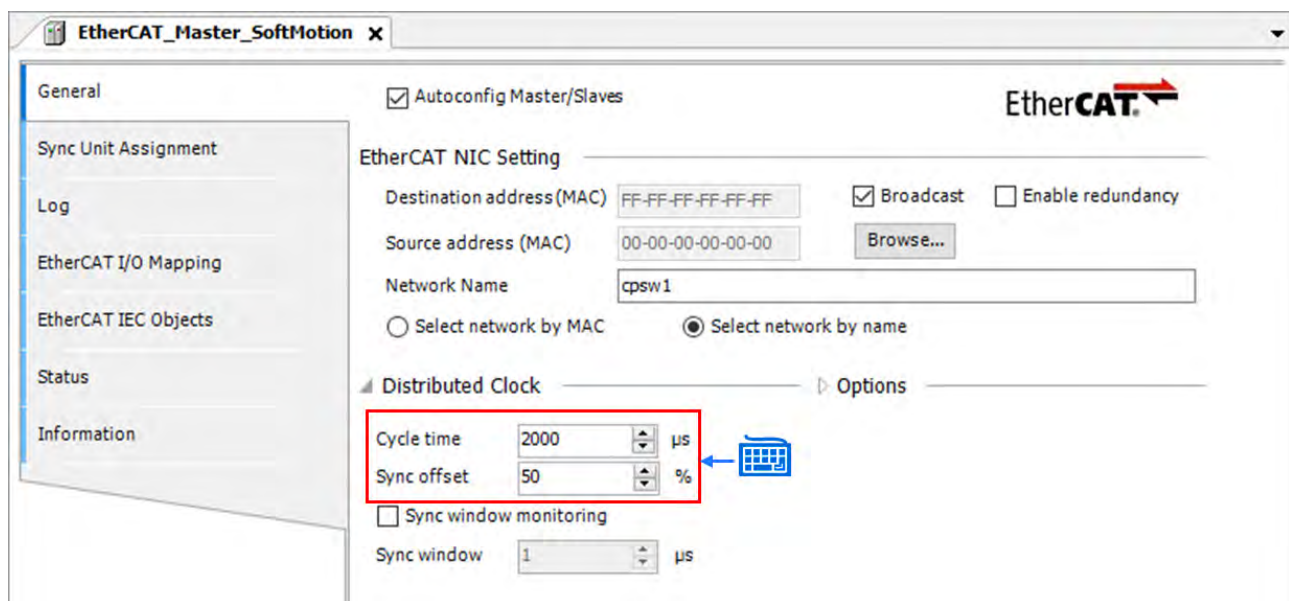
12. 进入「BuiltIn_IO」画面后，此范例为第一组 Counter，所以单击「Counter 0」，启动编码器需对应外部硬件配置。



13. 双击「EtherCAT_Master_SoftMotion」。

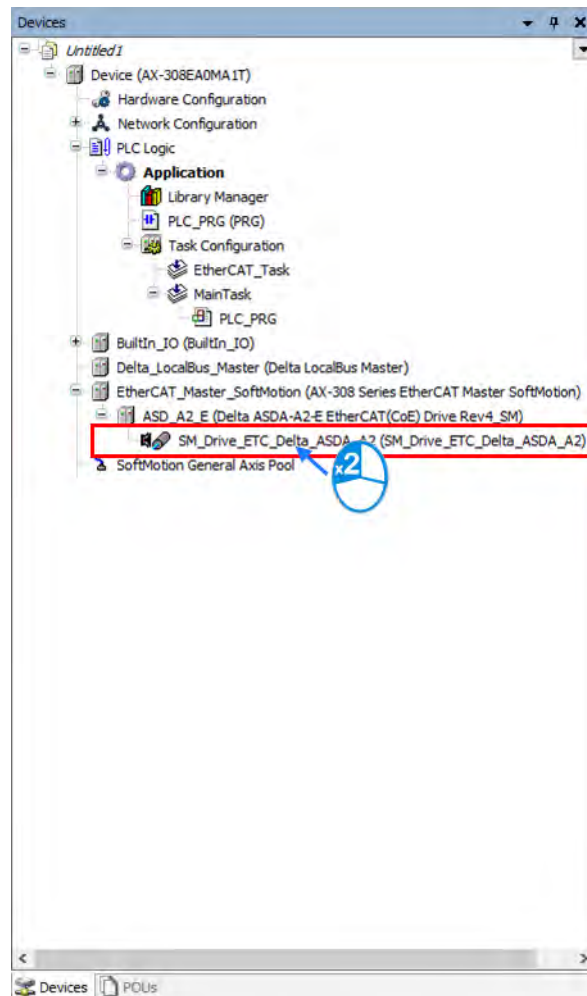


14. 进入「EtherCAT_Master_SoftMotion」画面后，设定「Cycle time」为 2ms 及「Sync offset」为 50。

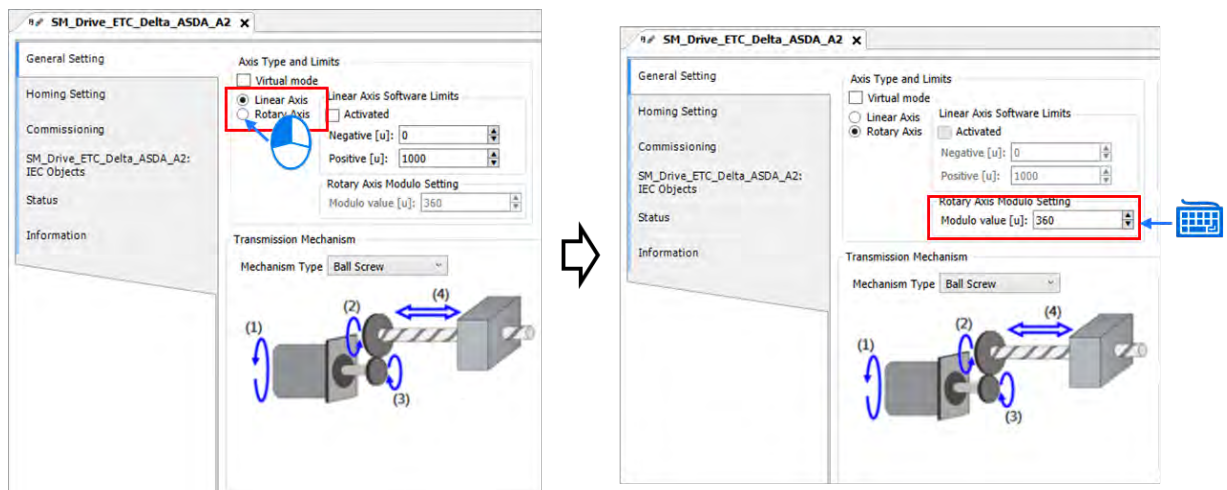


3.5.2.2 轴参数设定

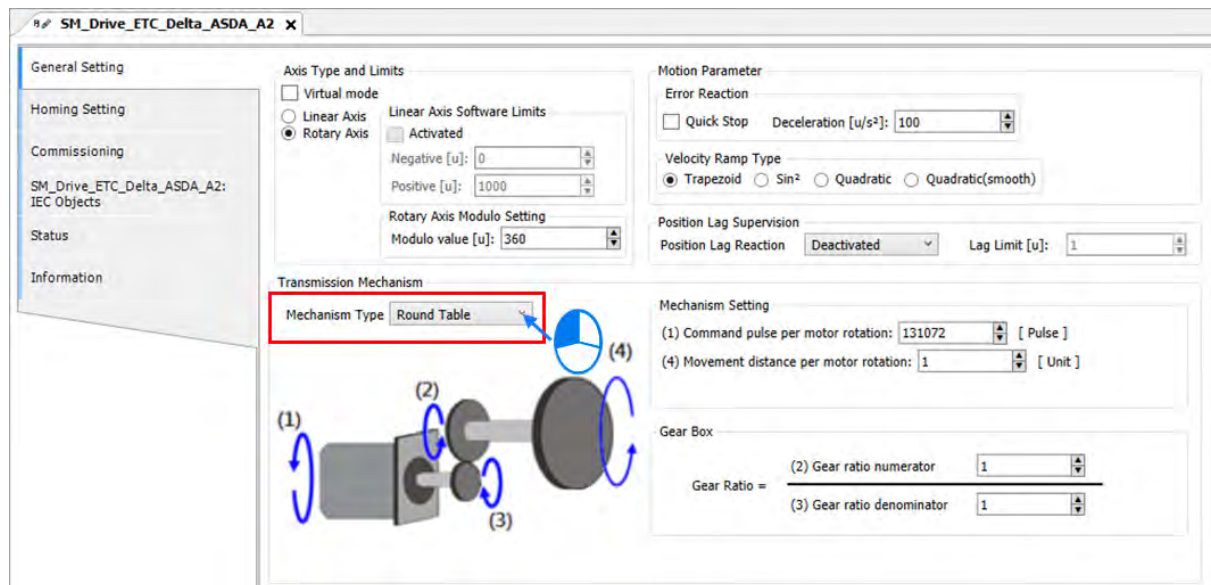
1. 首先选择第一轴「SM_Drive_ETC_Delta_ASDA_A2」后，单击左键两下。



2. 进入「SM_Drive_ETC_Delta_ASDA_A2」画面后，机构为旋转轴，需将直线轴 (Linear Axis) 改为旋转轴 (Rotary Axis)，旋转轴可设定旋转轴最大范围，此范例为 360 度。

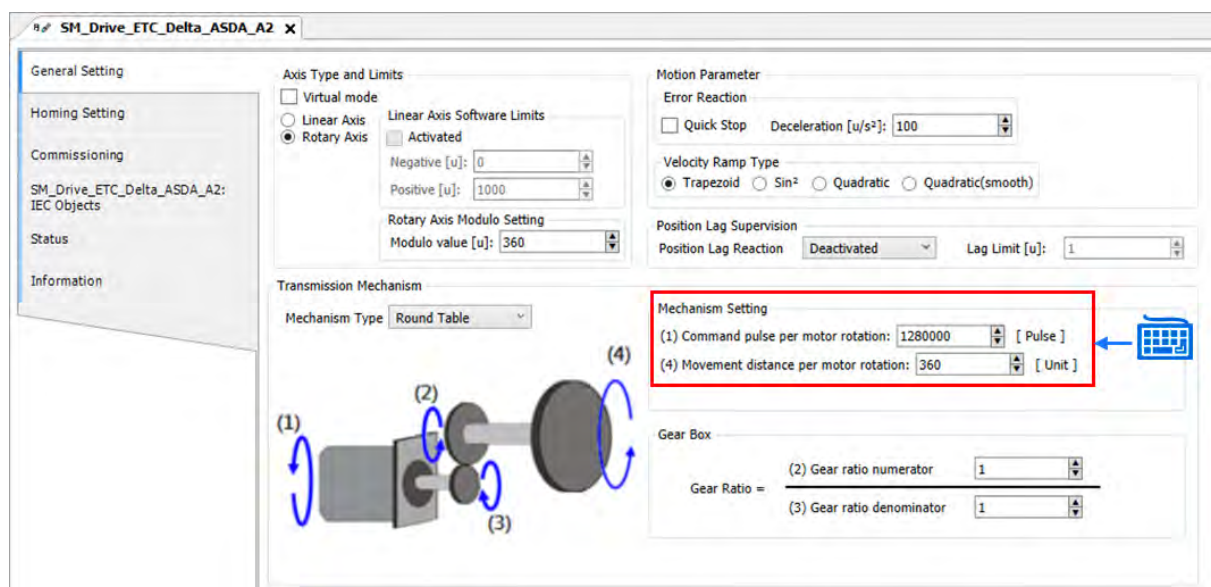


3. 此范例机构为旋转轴所以选择「Round Table」。

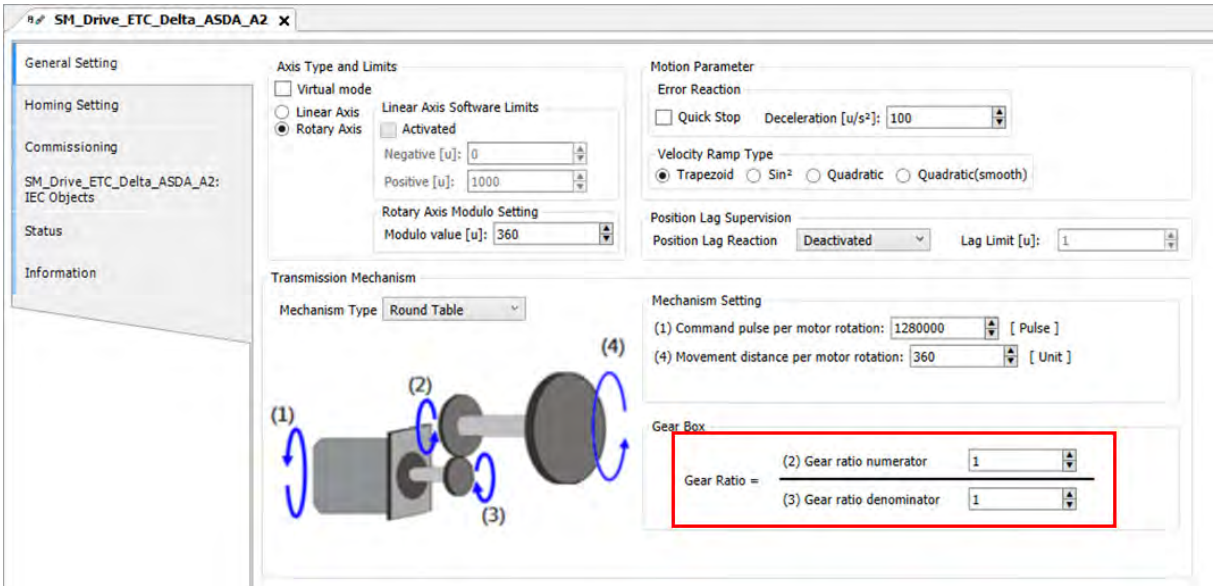


4. 实体机构选择完毕再设定齿比关系，在一圈脉冲（Command pulse per motor rotation）输入 1,280,000。

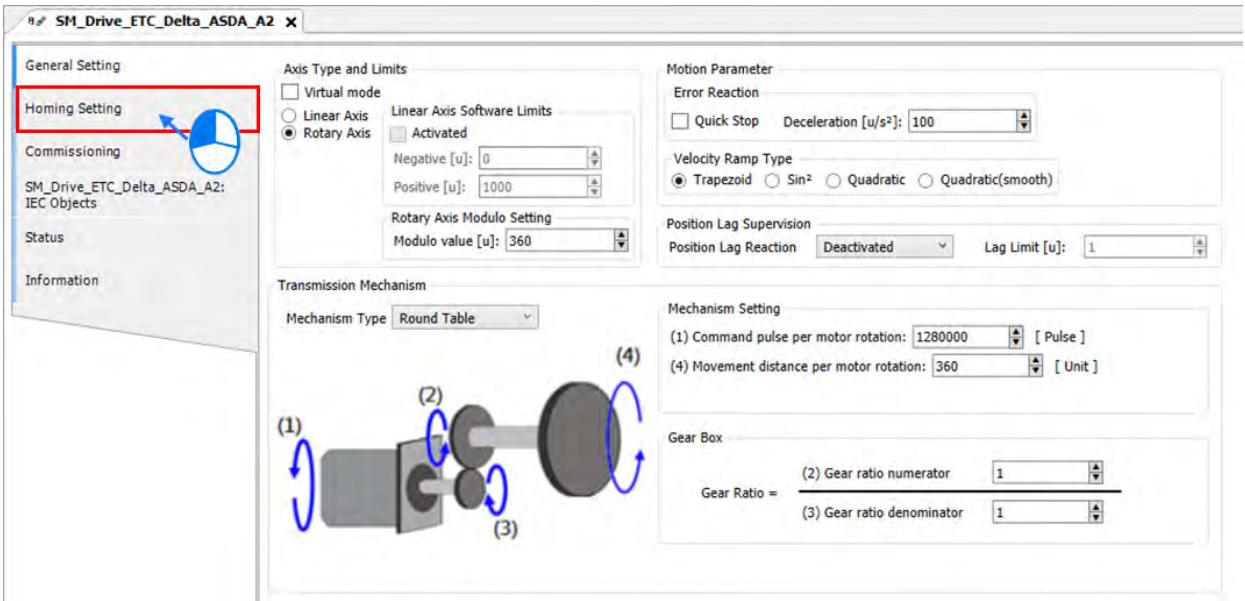
用户单位为 360 度。（伺服端齿比关系 $\frac{P1-44}{P1-45} = \frac{\text{电机一圈脉冲数}}{\text{Command Pulse per rotation}}$ ）



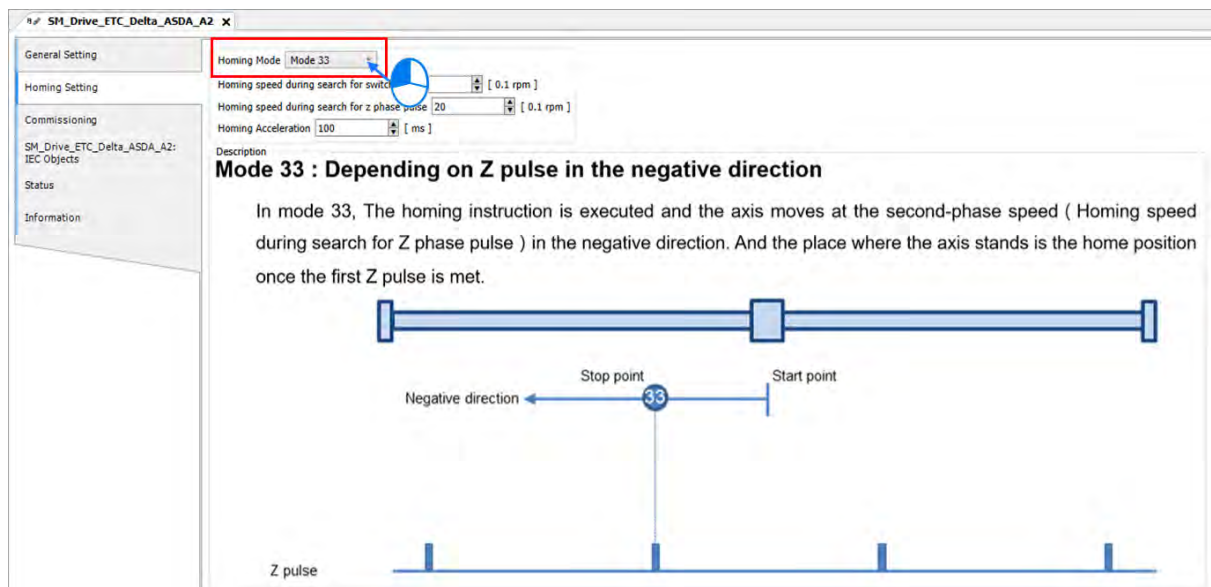
5. 实体机构未配减速机所以为 1 : 1。



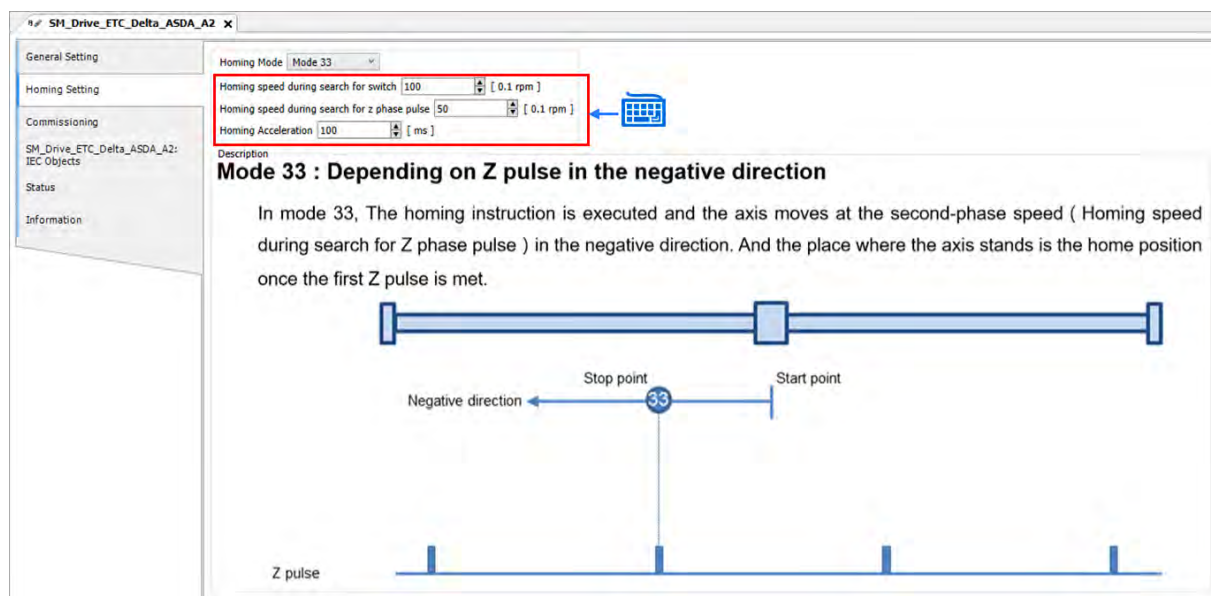
6. 将齿比关系设定完毕后，单击「Homing Setting」。



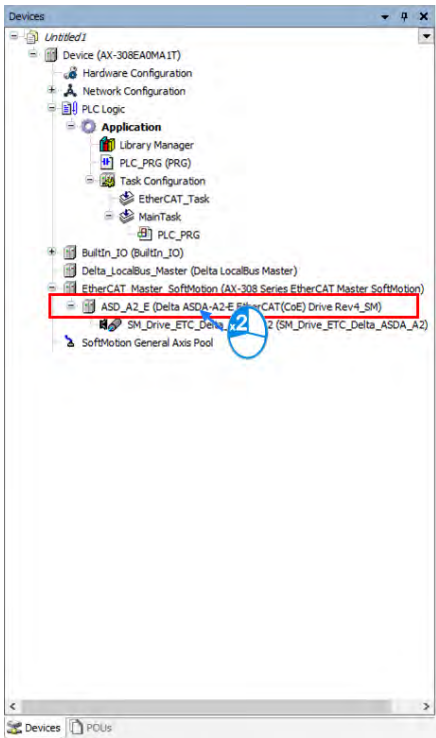
7. 因机构为旋转轴所以选择 Mode 33 以 Z 相方式回原点模式，每种模式请参考下方详细说明。



8. 再设定回原点的速度，此速度需搭配机构调整为适当速度。

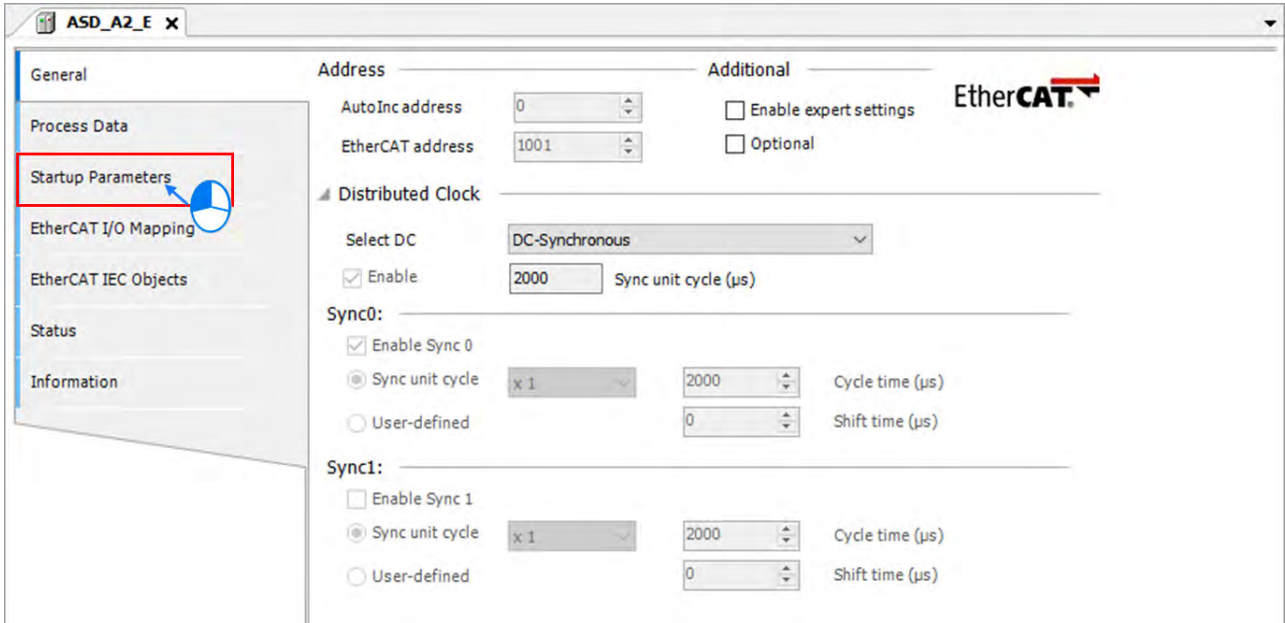


9. 将齿比与原点模式设定后，需在测试伺服端的参数，因硬件极限配置在伺服端的 DI9~DI10，所需参数设定 DI9 与 DI10 的引脚功能，此时单击「ASD_A2_E」。

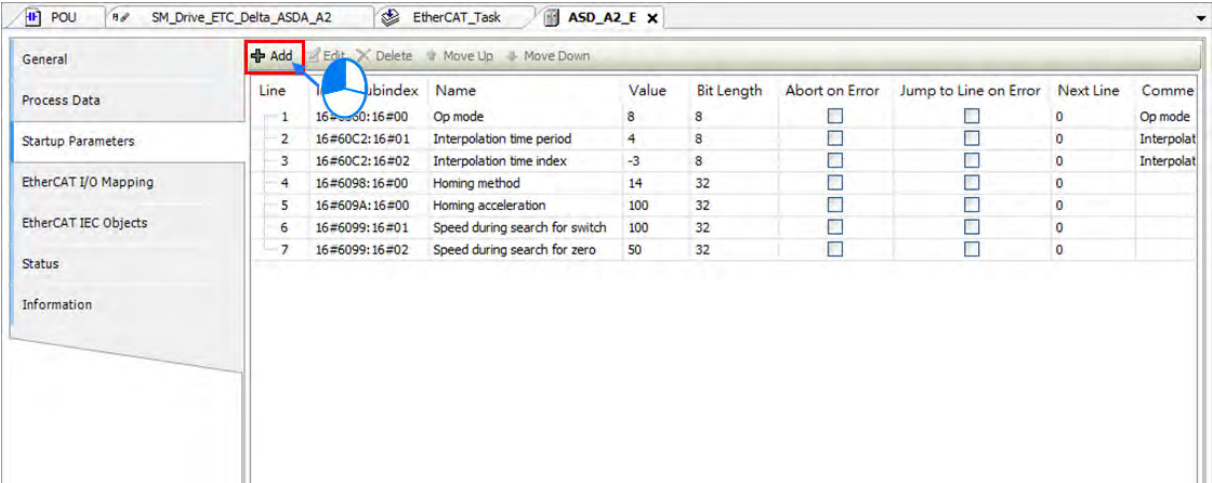


3

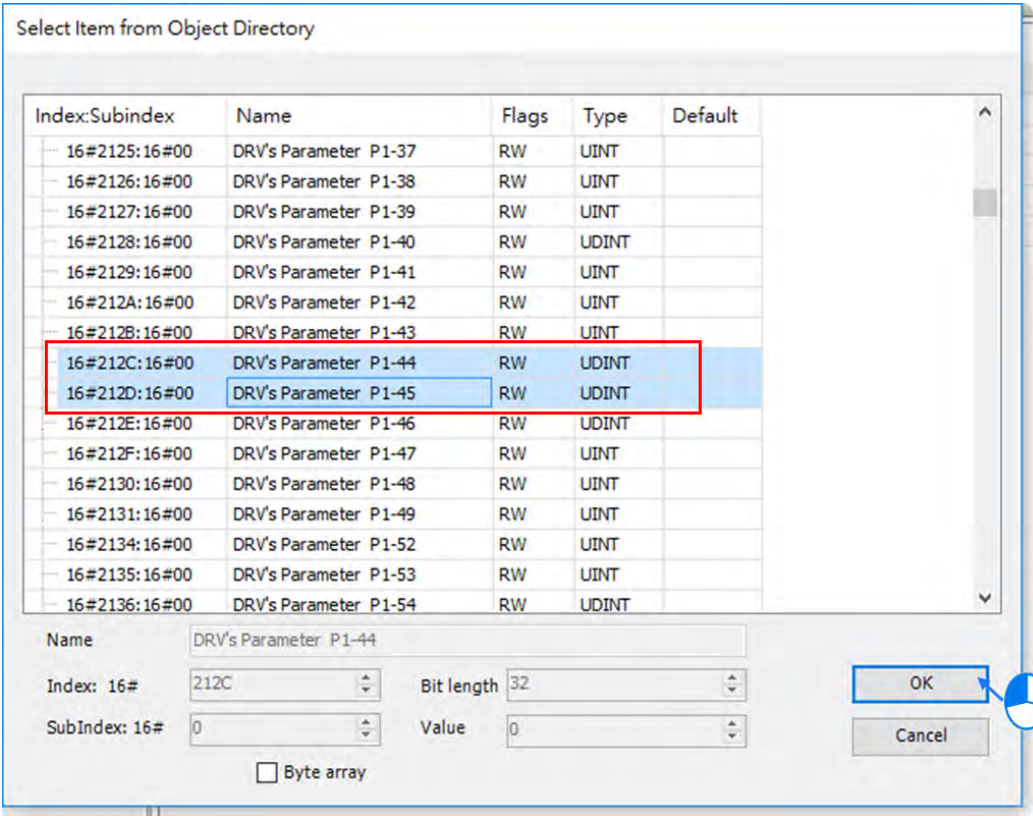
10. 进入「ASD_A2_E」画面，单击「Startup Parameters」。



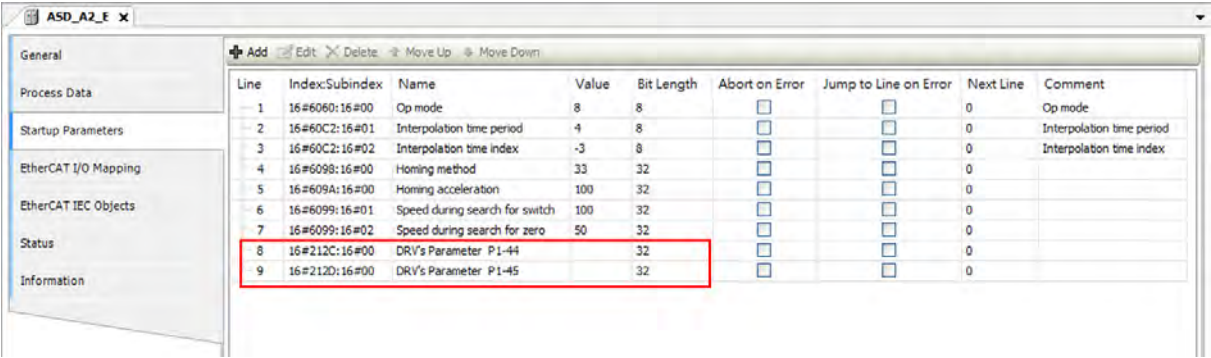
11. 在「Startup Parameters」画面中，单击「Add」新增参数。



12. 单击 P1-44 后可搭配 Shift 按键，选取多个参数，确认后再单击「OK」即可新增。

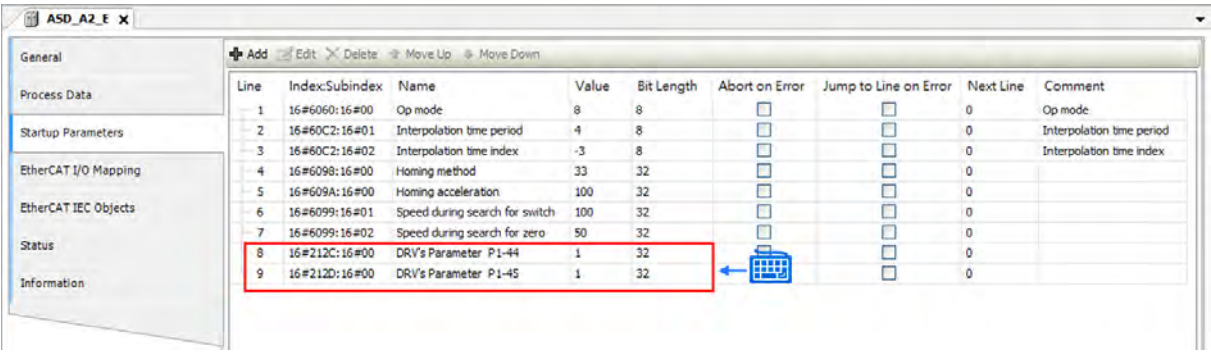


13. 新增后参数表中会出现刚刚所选择的参数。

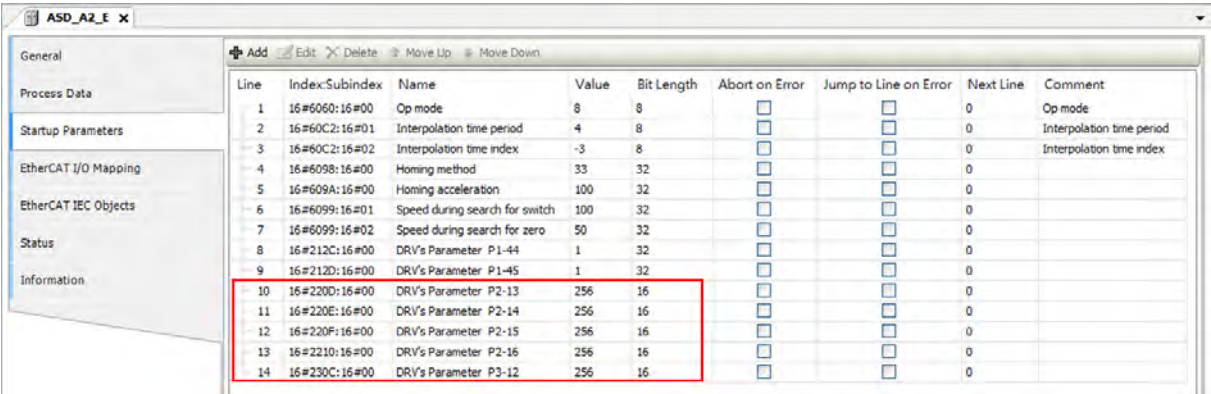


14. 将 P1-44 输入为 1 及 P1-45 设定 1。(此设定为伺服端的齿比需搭配轴参数中的齿比)(在此画面的参数会在 EtherCAT 通讯建立完毕后下载所有设定值于伺服端)

3



15. 利用相同操作设定下列参数，将伺服端 DI4~DI7 功能关闭，并设定断电保持参数设定。



■ 伺服端参数

下列参数为此范例使用，请输入至 Startup Parameters 中

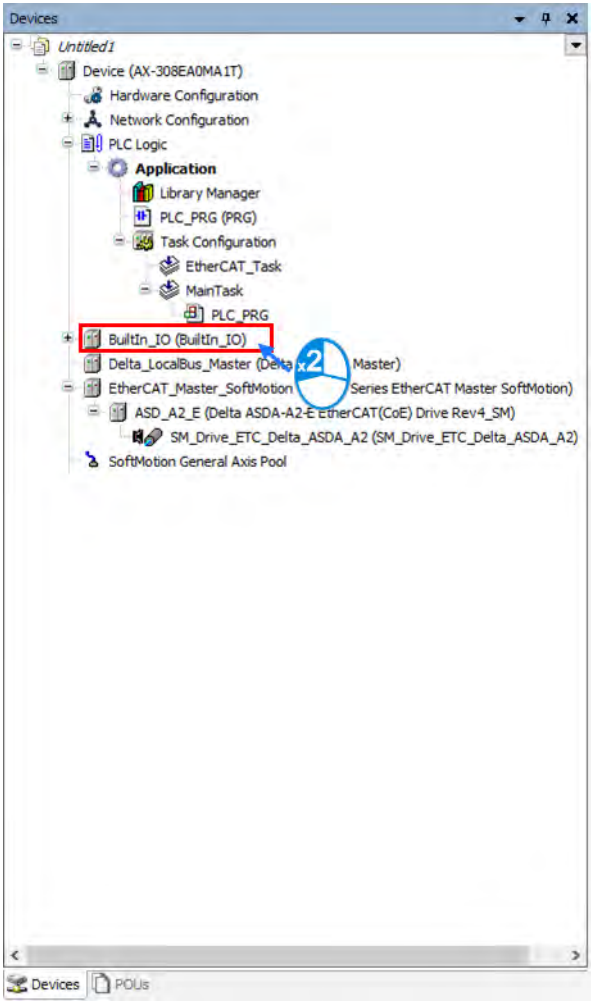
参数	功能	设定值
P1-44	电子齿轮比分子	1
P1-45	电子齿轮比分母	1
P2-13	数字输入引脚 DI4 功能规划	256
P2-14	数字输入引脚 DI5 功能规划	256
P2-15	数字输入引脚 DI6 功能规划	256

参数	功能	设定值
P2-16	数字输入引脚 DI7 功能规划	256
P3-12	参数会维持驱动器断电前的参数	256

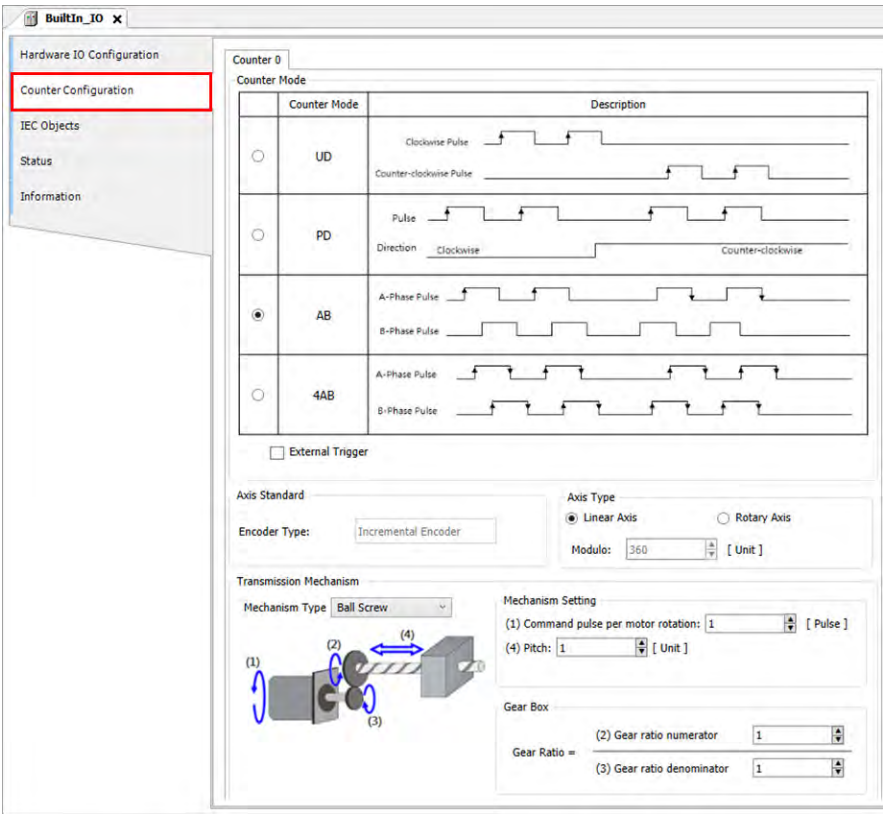
注：参数详细介绍可参考 **ASDA-A2 系列应用技术手册**

16. 实轴设定完毕后，再设定编码器轴，双击「BuiltIn_IO」。

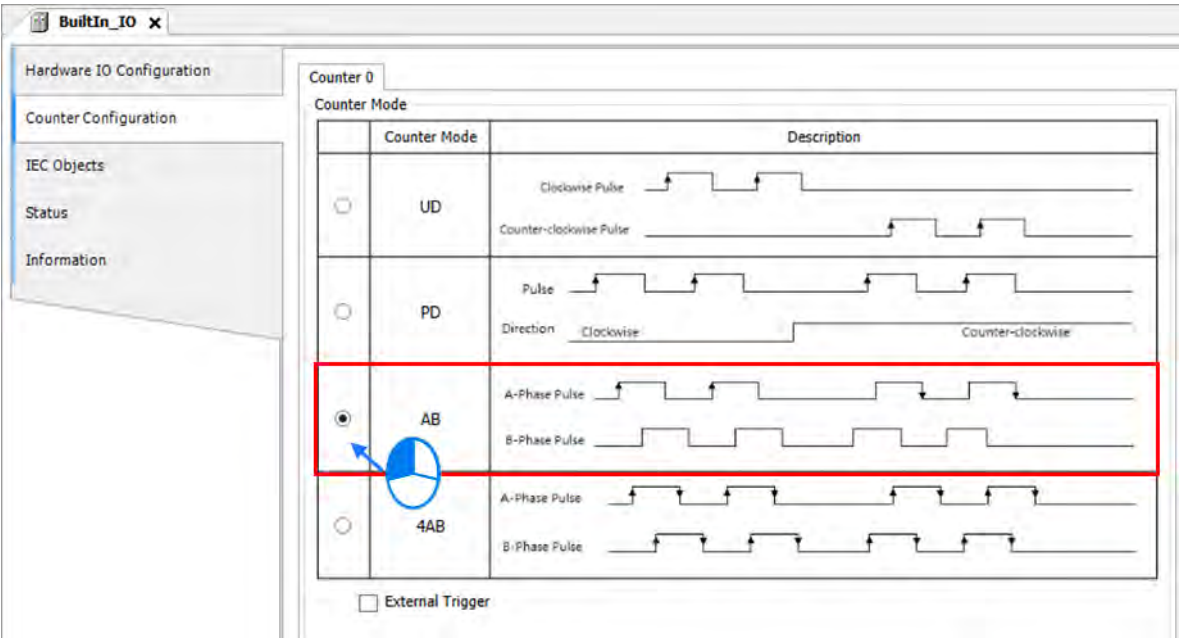
3



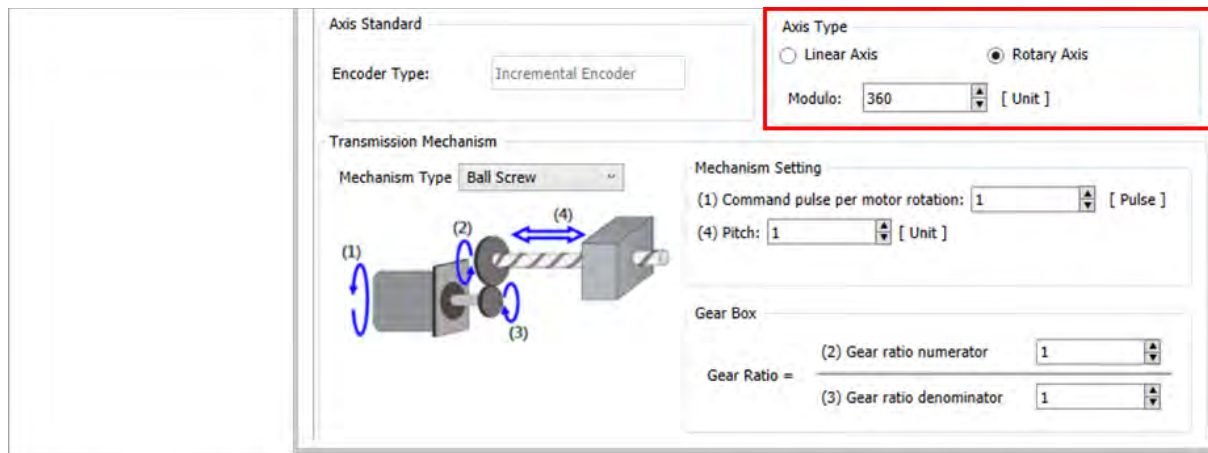
17. 进入「BuiltIn_IO」画面后，单击「Counter Configuration」。



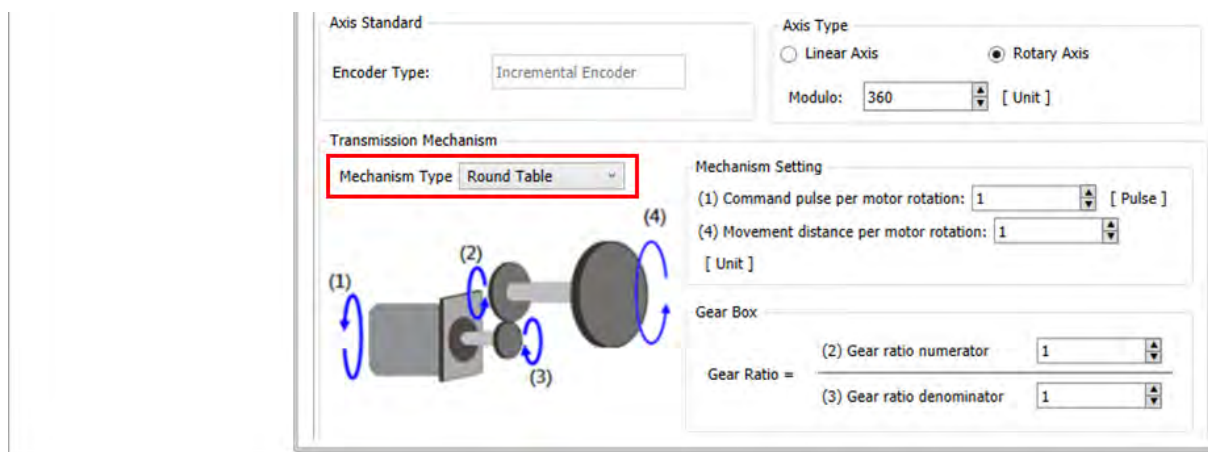
18. 设定编码器信号类型，范例使用的编码器为 AB 相信号输入。



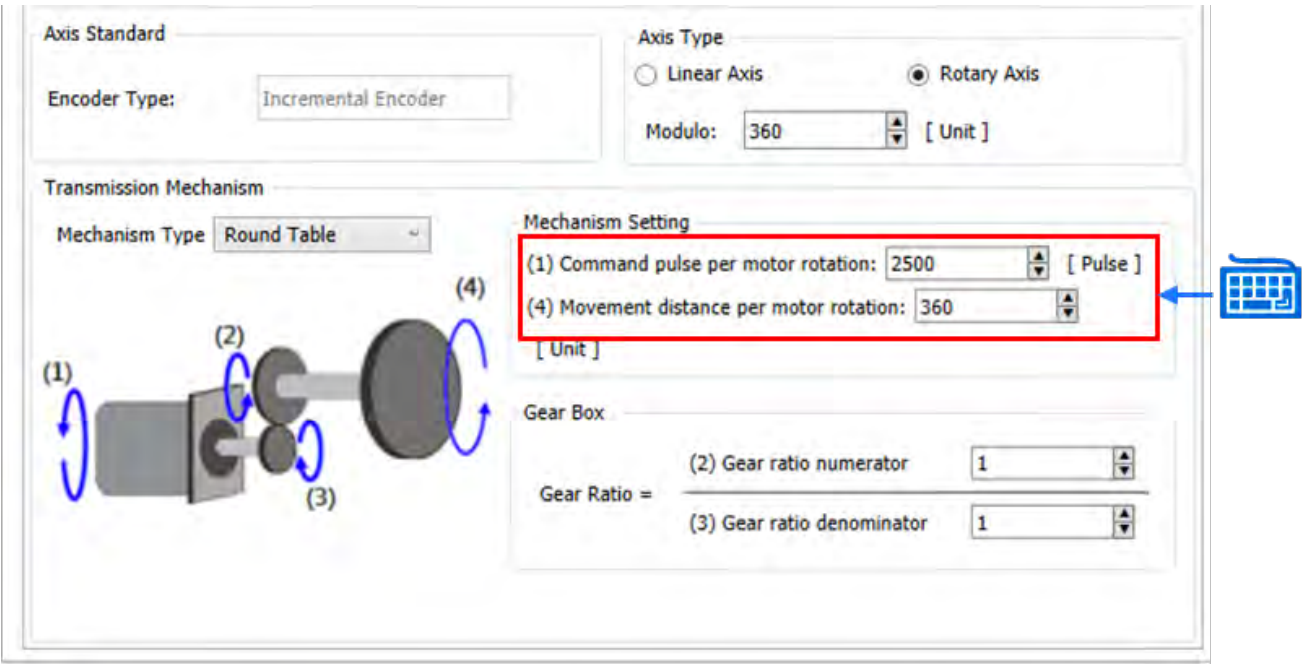
19. 将编码器设定为旋转轴 (Rotary Axis)，一圈单位为 360 度。



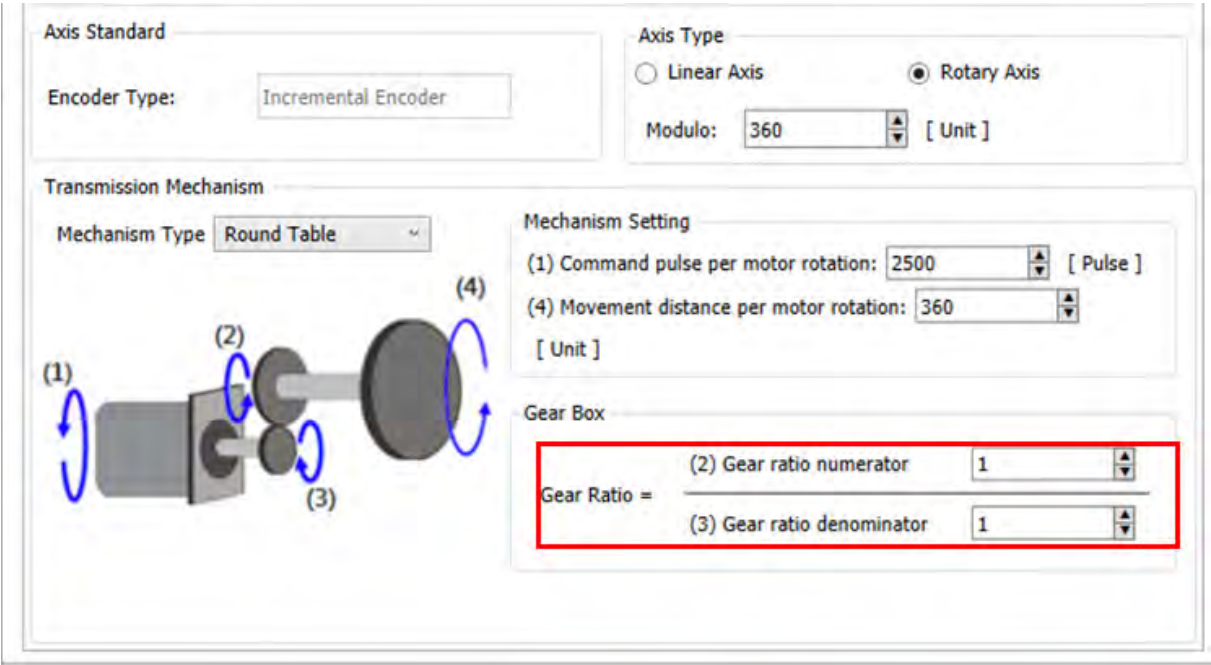
20. 范例使用的机构类型为 Round Table



21. 实体机构选择完毕再设定齿比关系，在一圈脉冲（Command pulse per motor rotation）输入 2500（编码器一圈为 2500），用户单位为 360 度。



22. 实体机构未配减速机所以为 1 : 1。



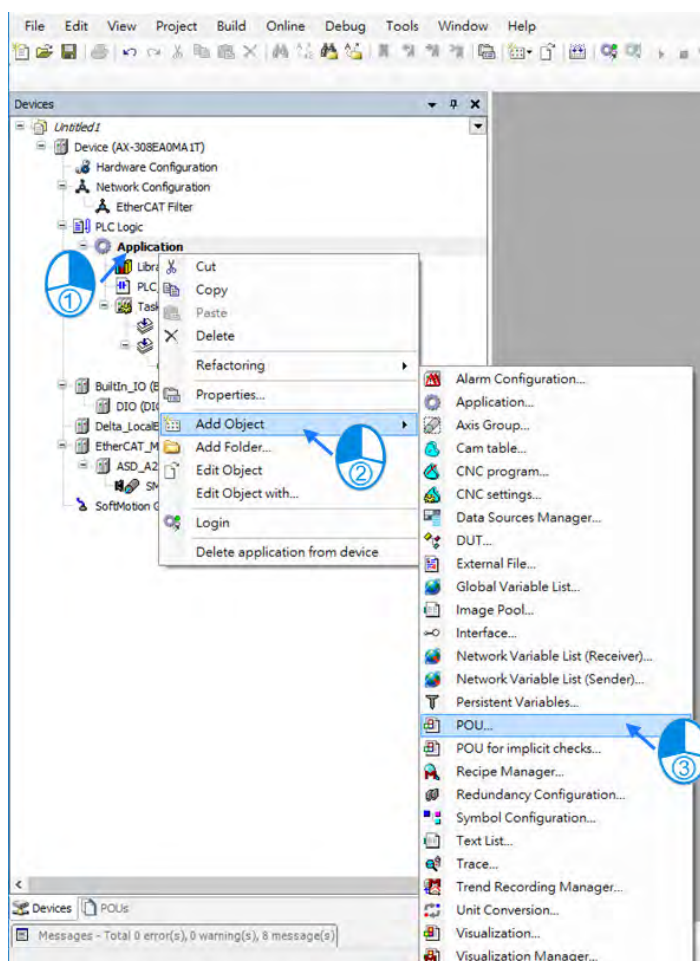
3.5.2.3 项目程序编写

- 程序执行流程

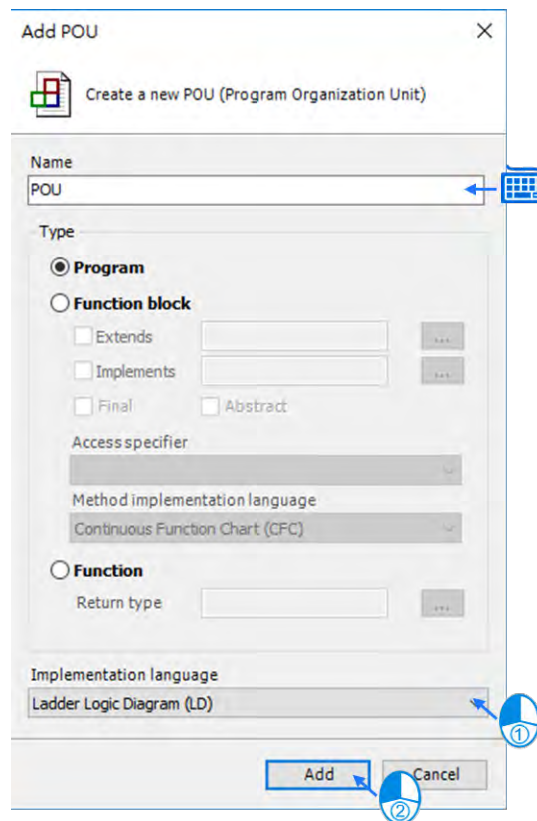
- 执行「Step1」后会触发「MC_Power」与「DFB_HCnt」的功能块(Counter 功能需要透过 DFB_HCnt 才可启动功能)。
- 执行「Step2」后会触发「MC_Home」与「MC_SetPosition」, 让实轴回原点及编码器轴位置为 0。
- 执行「Step3」后会触发「MC_GearIn」, 从轴会开始跟随, 当「Step3」关闭后会触发「MC_GearOut」, 「MC_GearOut」的 Done 启动后, 会再触发「MC_Stop」功能让从轴停止运转。
- 自动化执行一次, 「Counter_1」会增加一次。

- 项目建立流程

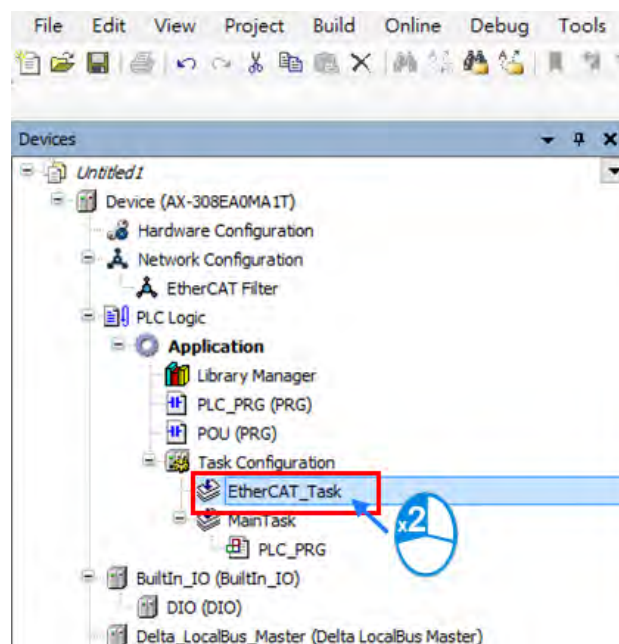
1. 首先建立新的 POU, 选择「Application」单击右键, 再选择「Add Object」 > 「POU」。



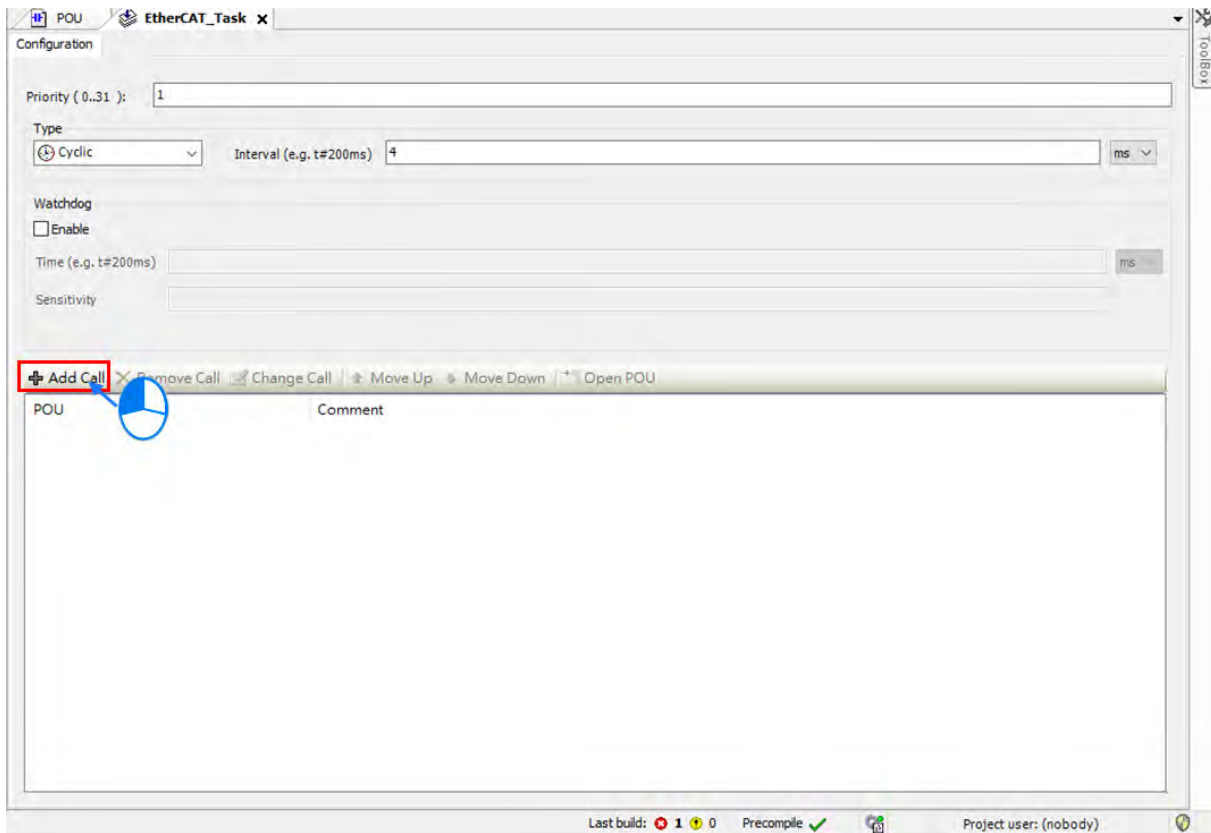
2. 新增画面中，「Name」可输入名称，再「Implementation language」可选择 LD 程序语言，完成后点「Add」，新增 POU。



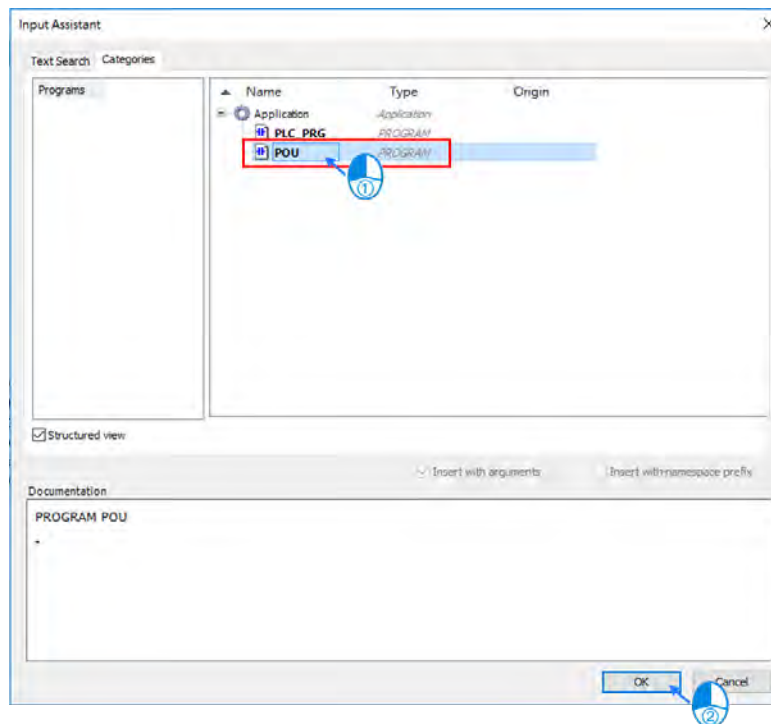
3. 新增的 POU 会出现在树状中，此时需将 POU 新增在 Task 中，单击「EtherCAT_Task」。(有使用轴运动相关 FB 都需建立在 EtherCAT_Task 中，以确保轴可正常运作)



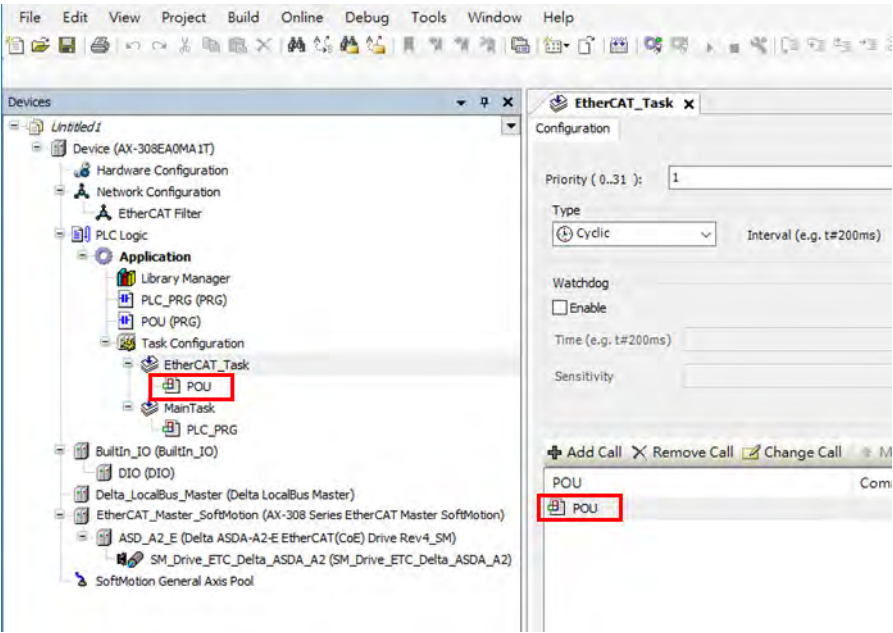
4. 在「EtherCAT_Task」中，单击「Add Call」。



5. 选择刚所建立的「POU」后，单击「OK」。

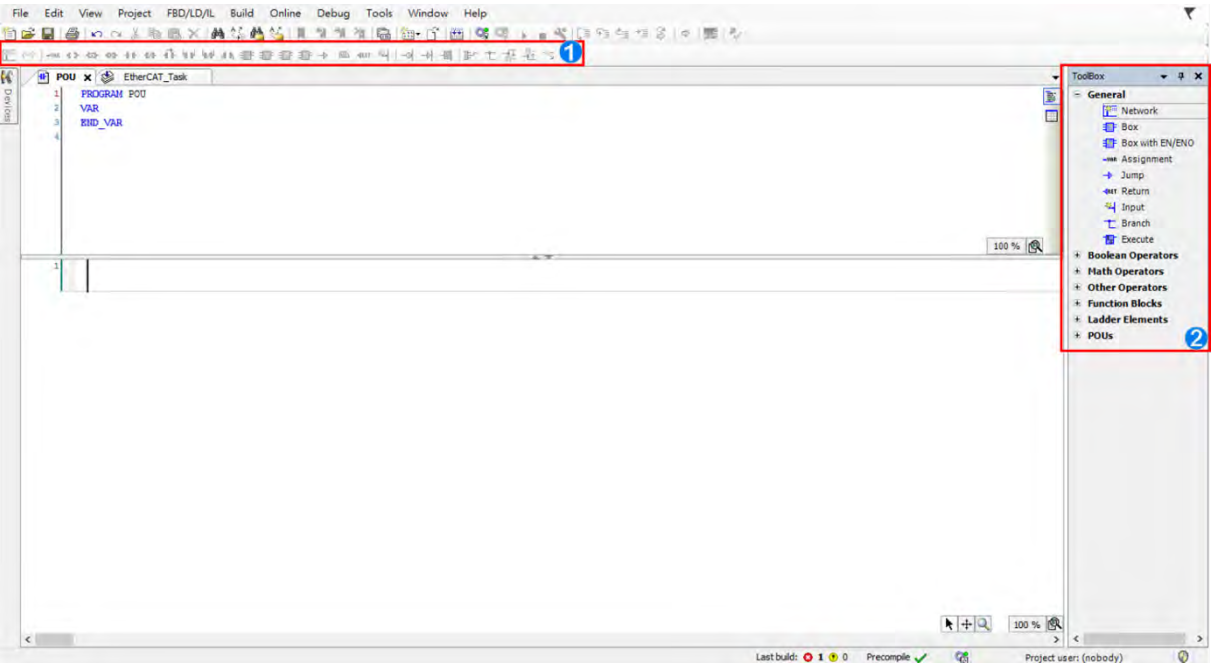


6. 在「EtherCAT_Task」新增完毕后，下方及树形图会出现「POU」，此时再双击树形图的「POU」。

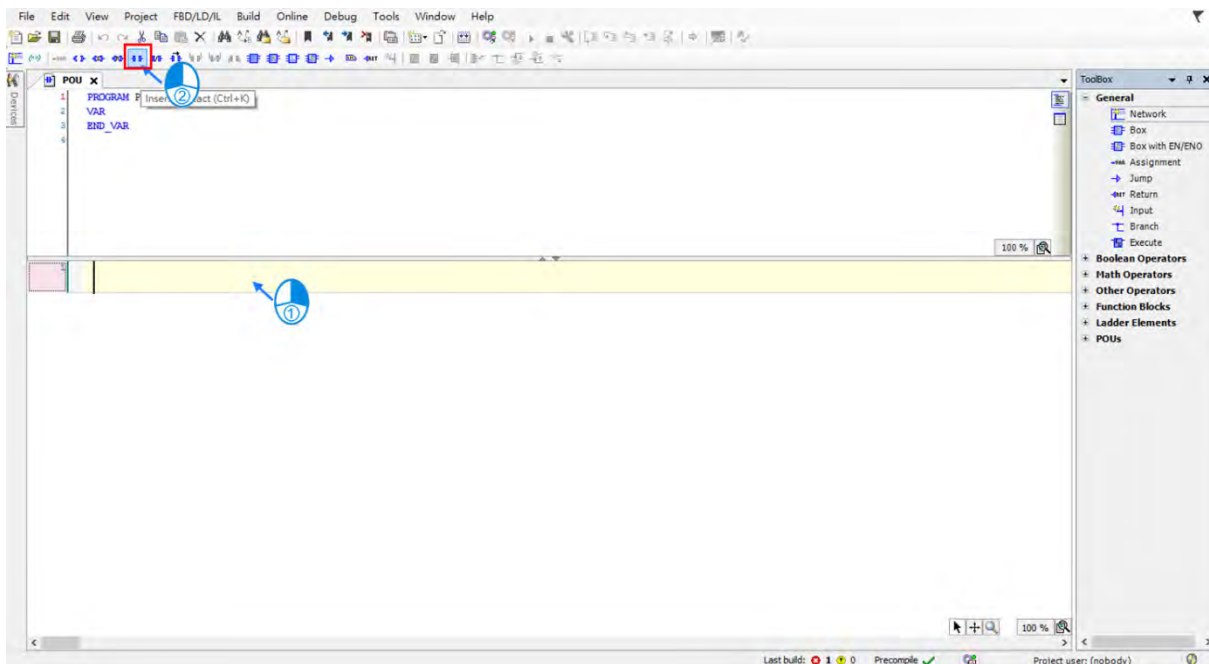


3

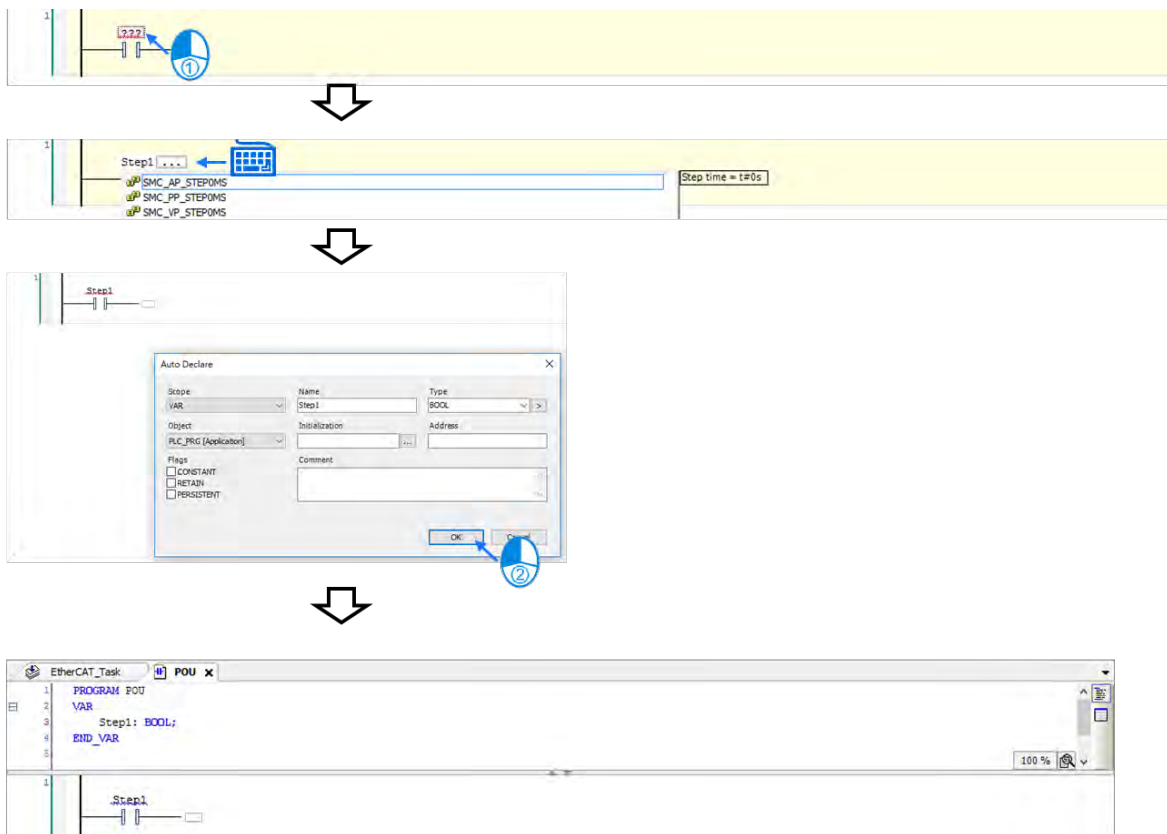
7. 使用 LD 语言编写程序，需新增指令可通过①添加，如需使用功能（Function）或者是功能块（Function Block）需通过②添加。



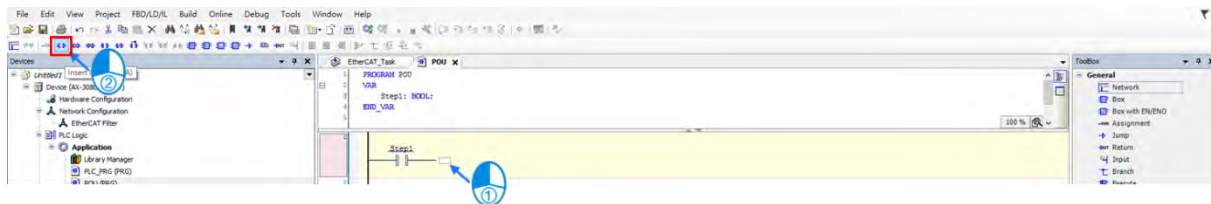
8. 在 POU 中单击欲新增指令的地方后，再单击  即可新增指令。



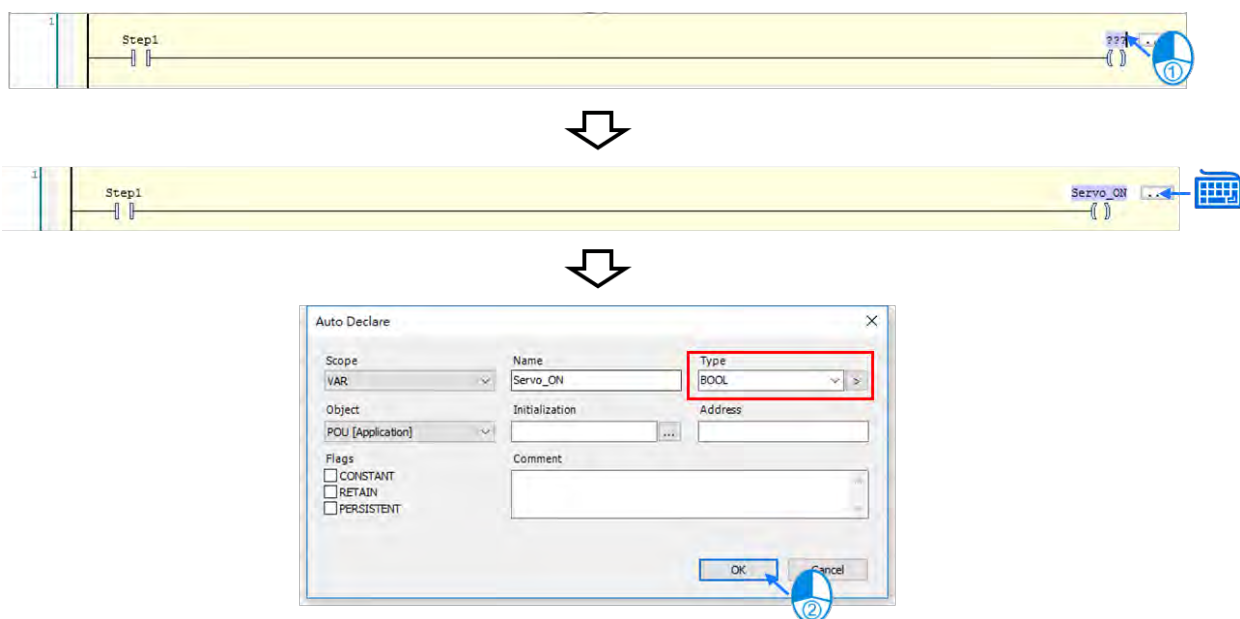
9. 「POU」中出现刚新增的指令后，单击「???」可输入指令变量名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



10. 再来新增输出指令，在 POU 中单击欲新增指令的地方后，再单击  即可新增指令。



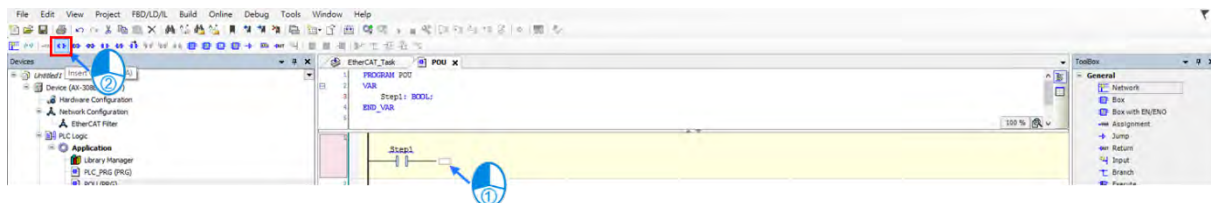
11. 单击「???」后可输入名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



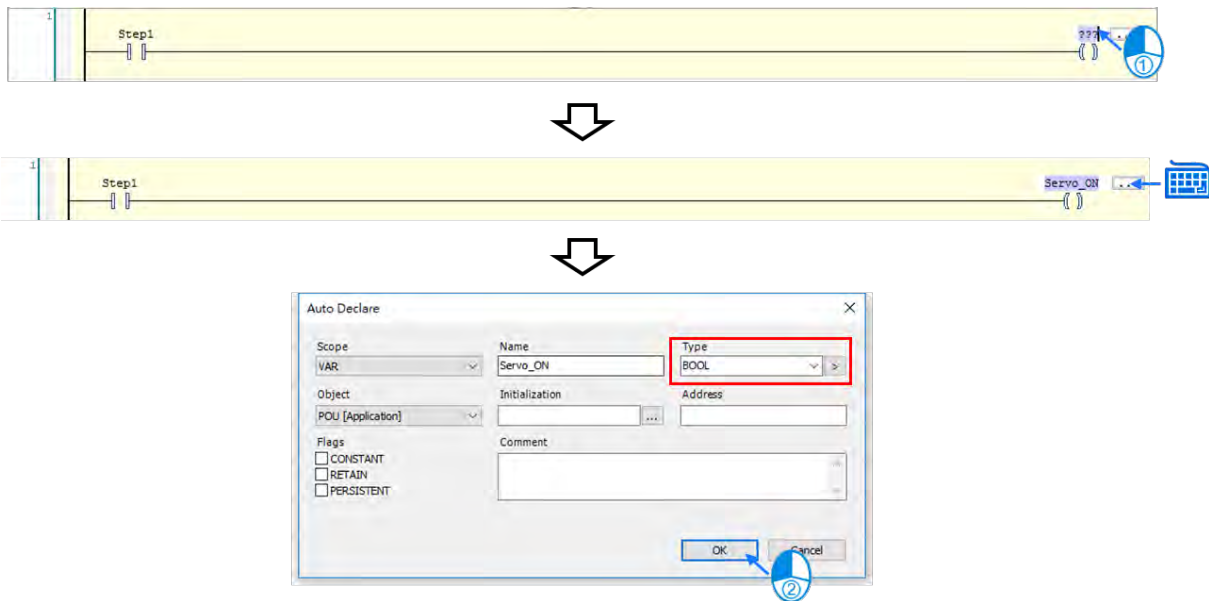
12. 先前新增的指令名称都会在上方便显示名称与类型，也可在上方直接新增指令名称与类型。(后续范例中使用指令方式都是相同操作)



13. 再来新增输出指令，在 POU 中单击欲新增指令的地方后，再单击  即可新增指令。



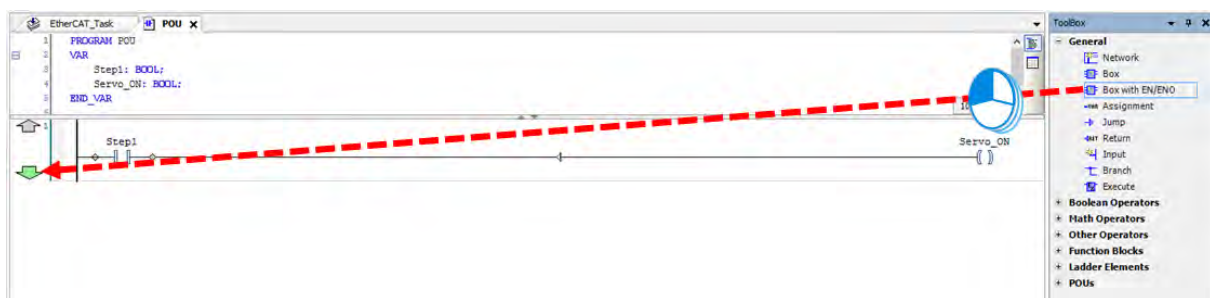
14. 单击「???'」后可输入名称，输入完成后按下 Enter 键，就会出现「Auto Declare」画面，可确认「Type」是否正确后再单击「OK」。



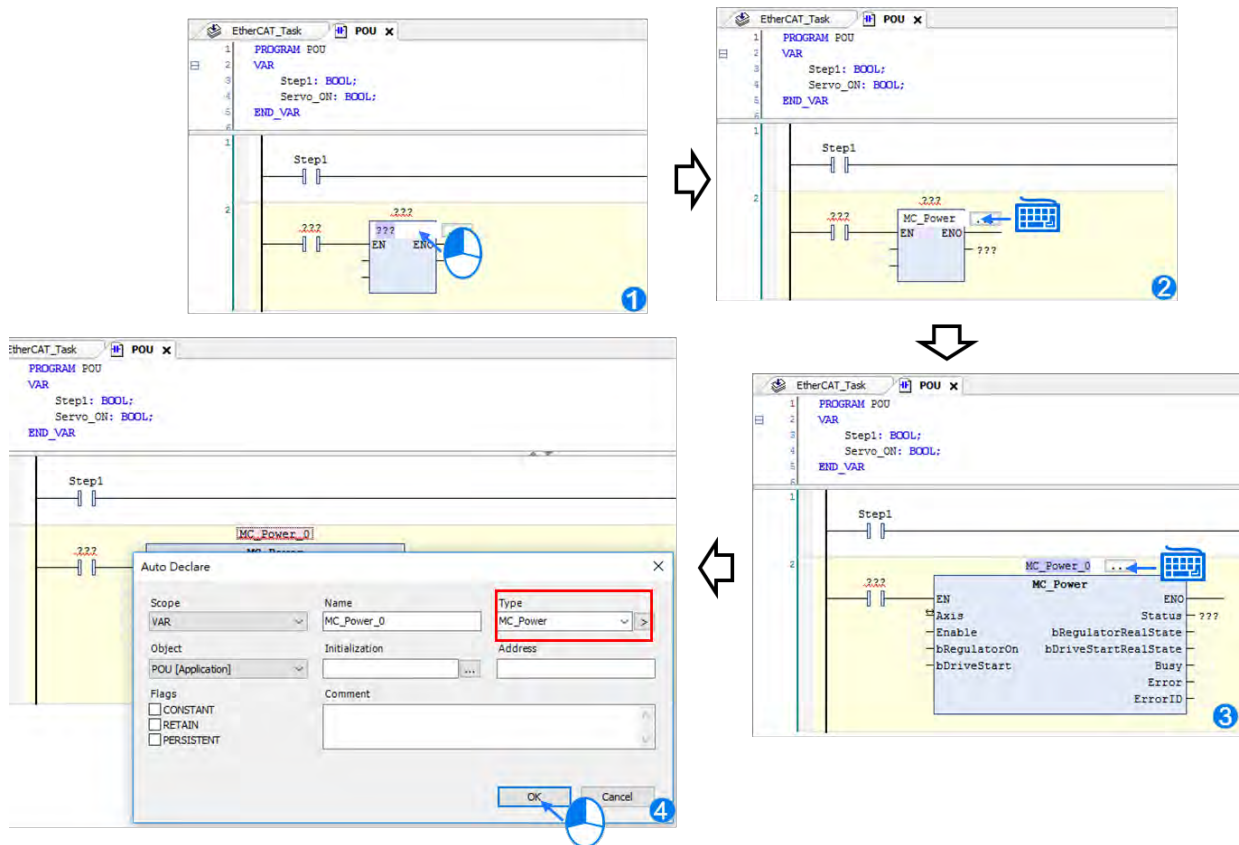
15. 先前新增的指令名称都会上方显示名称与类型，也可在上方直接新增指令名称与类型。(后续范例中使用指令方式都是相同操作)



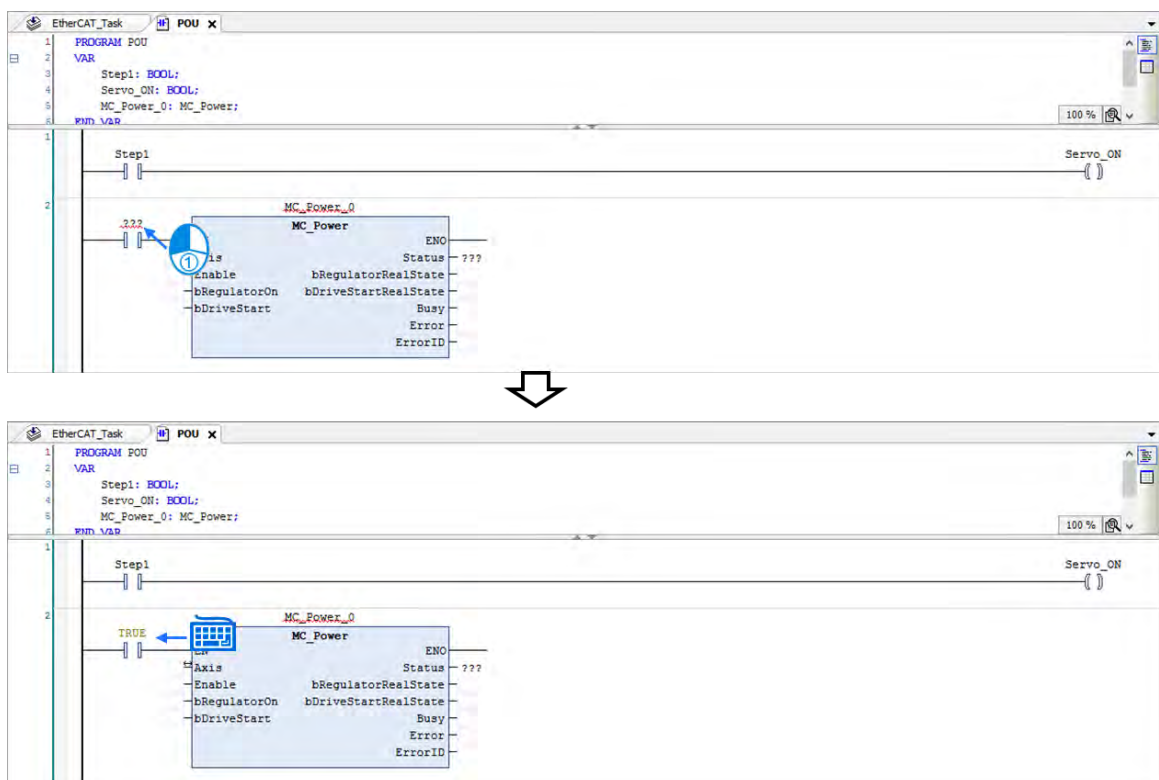
16. 再来新增功能块，单击「Box With EN / ENO」后拉至 POU 中的 。



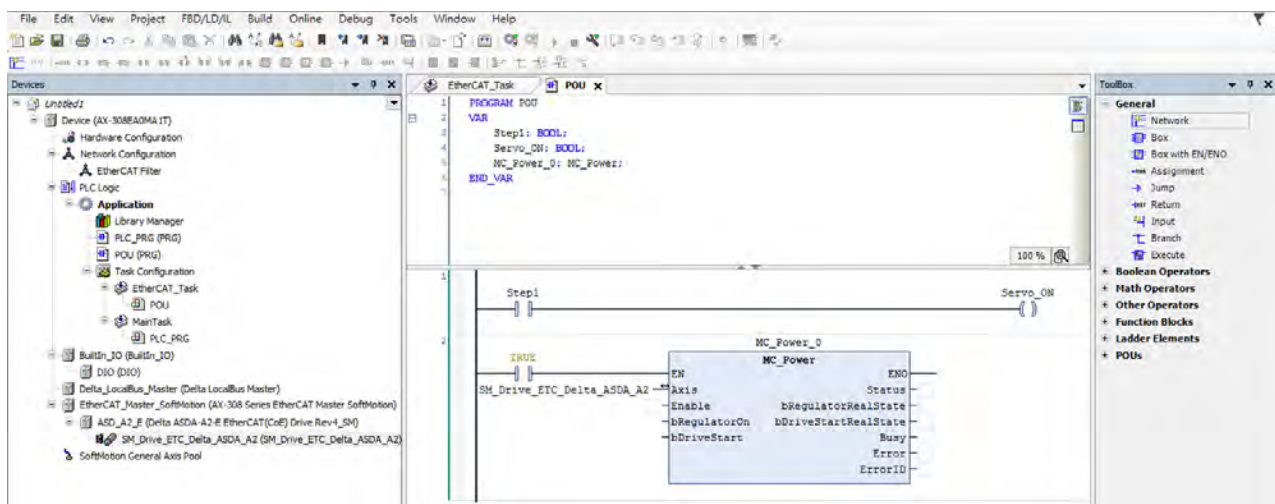
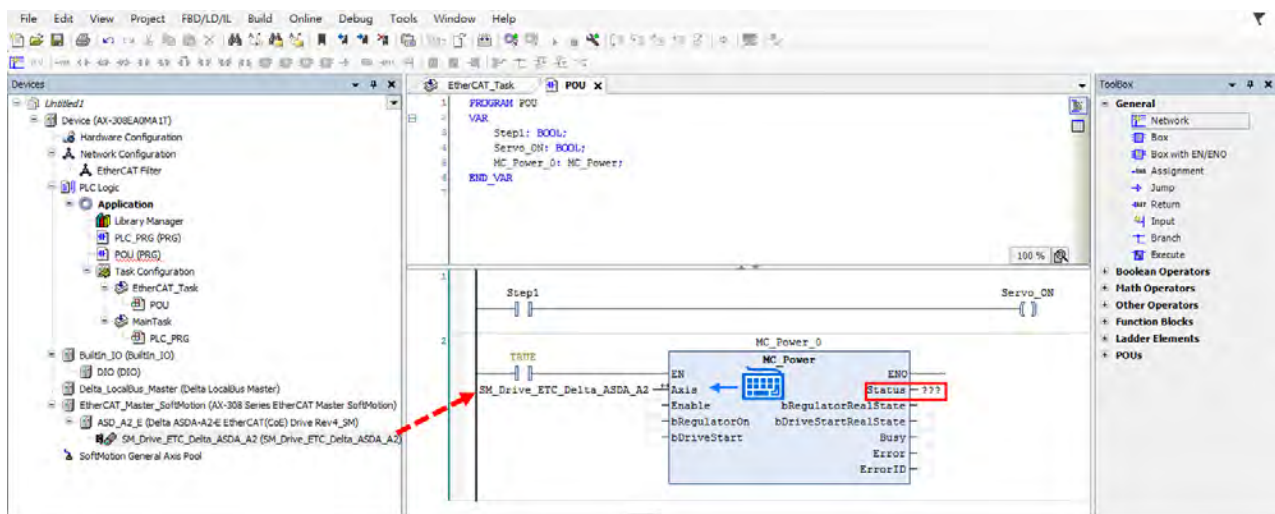
17. 单击「???'」后输入功能块名称，输入完成后按下 Enter 键，此时功能块就会出现，再输入功能块名称后按下 Enter 键，就会出现「Auto Declare」，确认功能块「Type」正确后单击「OK」。



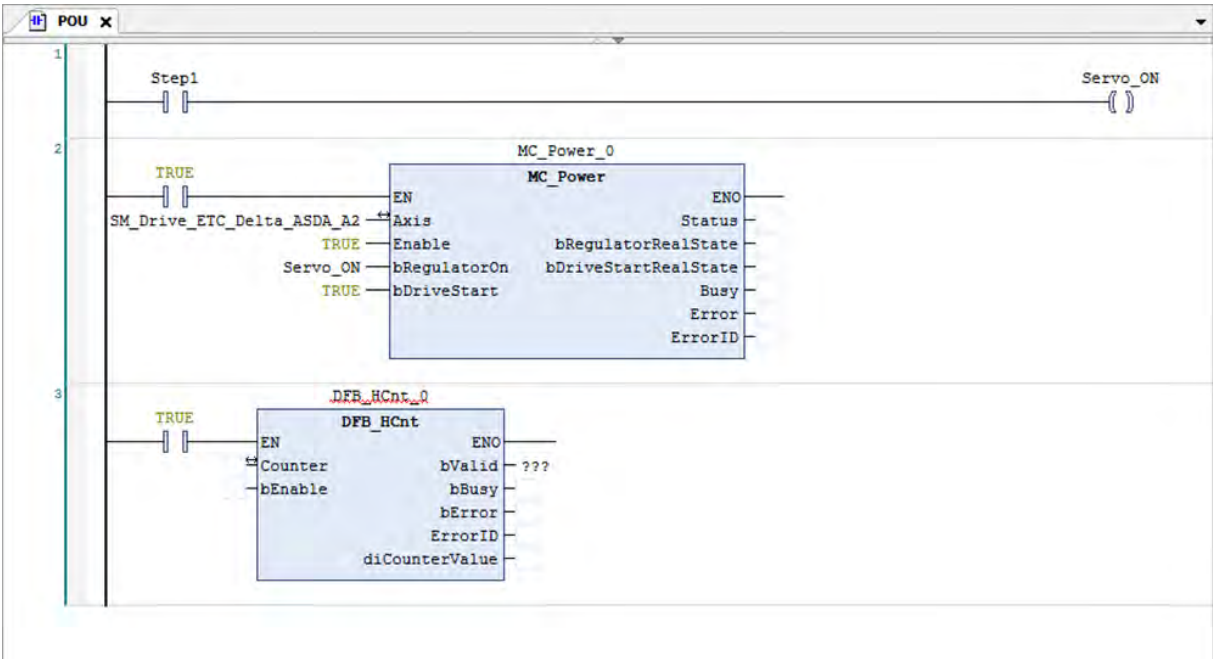
18. 功能块新增好，单击「???」输入 True 后单击 Enter 键。



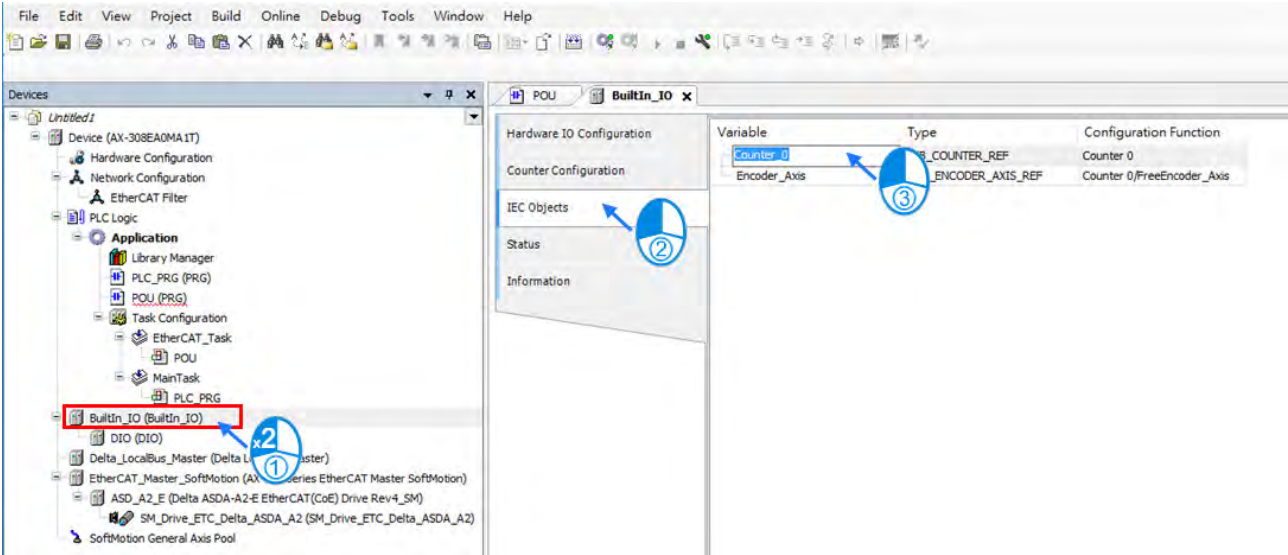
19. 将伺服名称输入至功能块中的 Axis 引脚中并删除引脚中的「???」，因「???」不为任何变数，在编译时会出错。



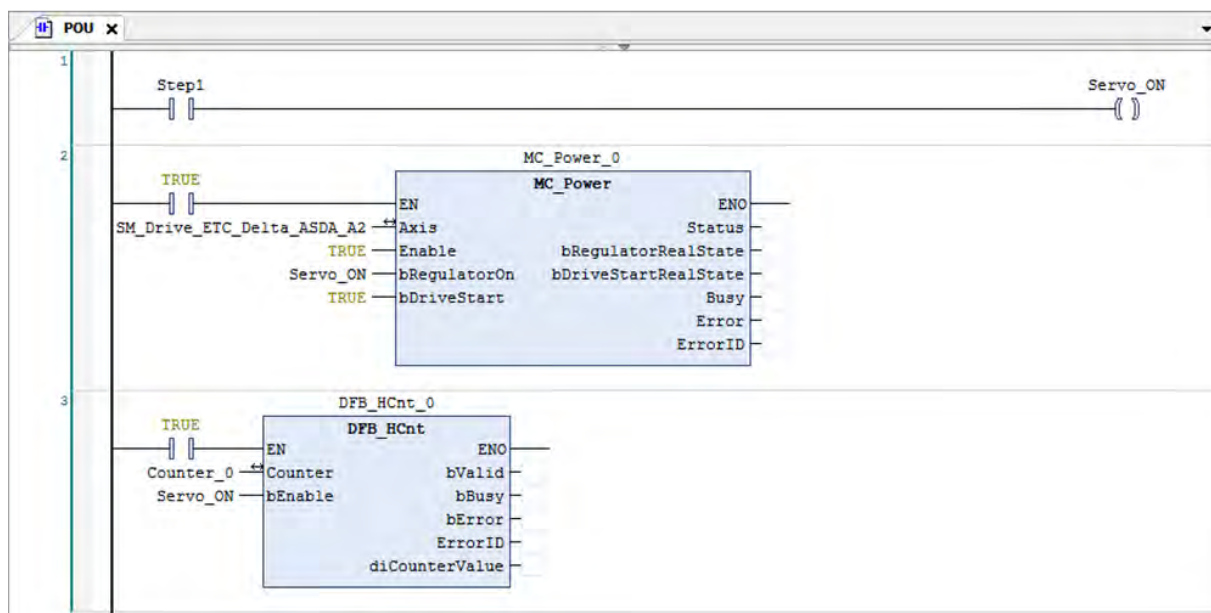
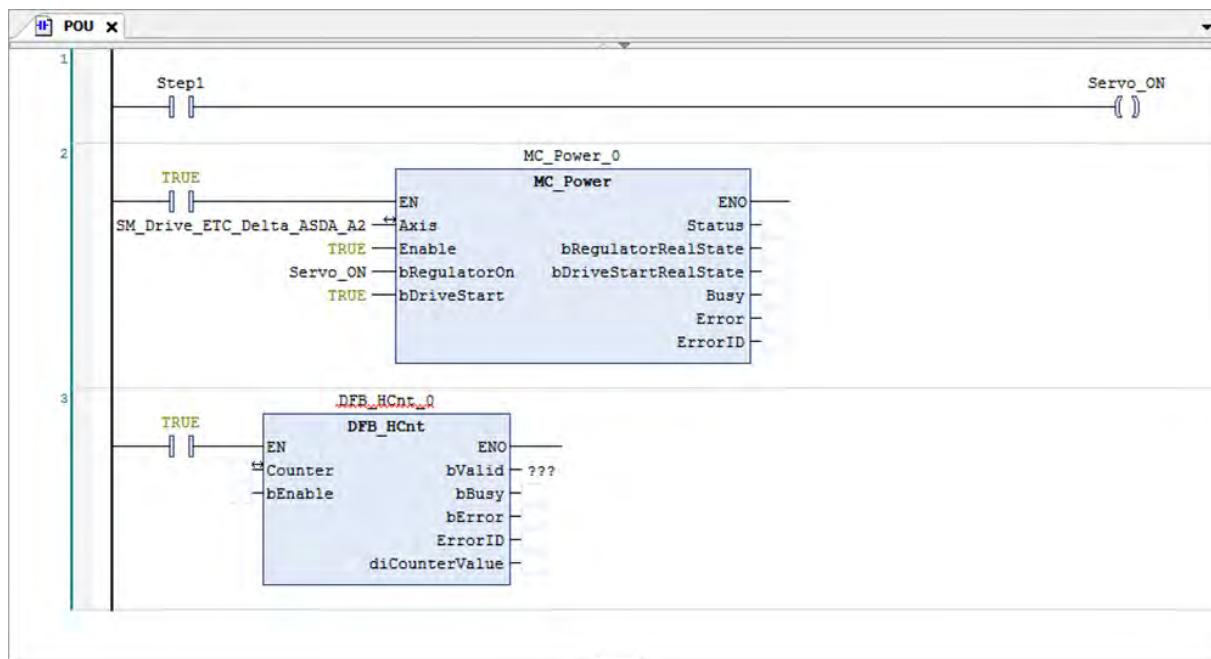
20. 利用上方的操作建立「MC_Power」后，再建立「DFB_HCnt」功能块。



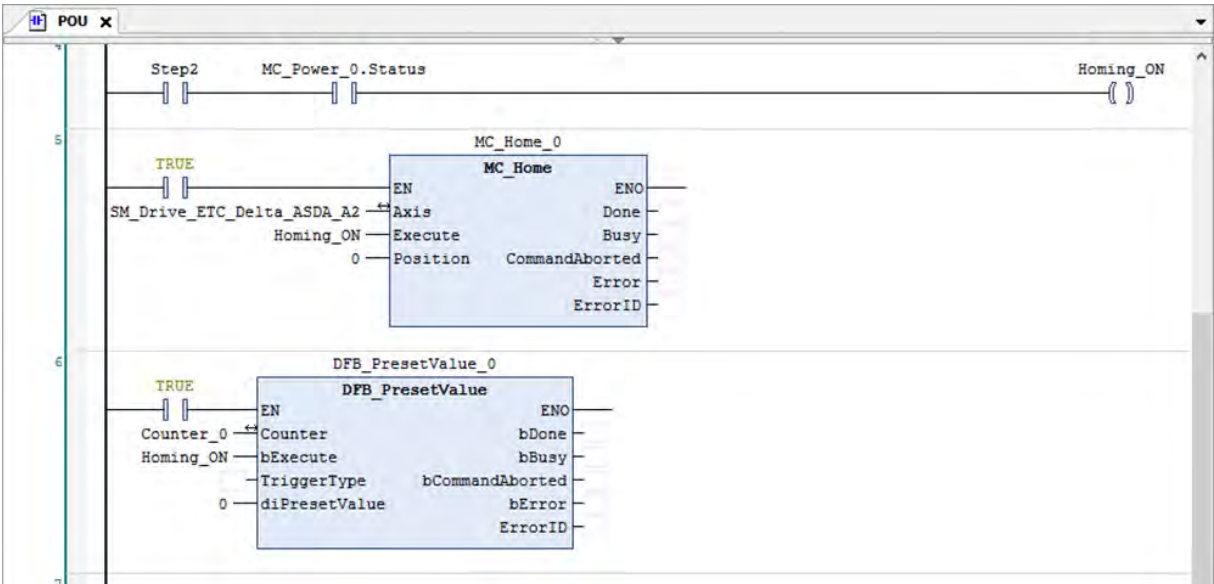
21. 编码器名称需选「BuiltIn_IO」>「IEC Objects」>「Variable」输入编码器的名称为「Counter_0」，当新增脉冲轴名称都会在此显示。



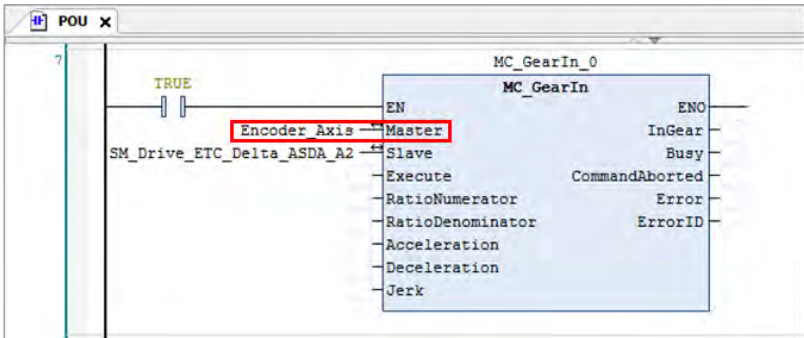
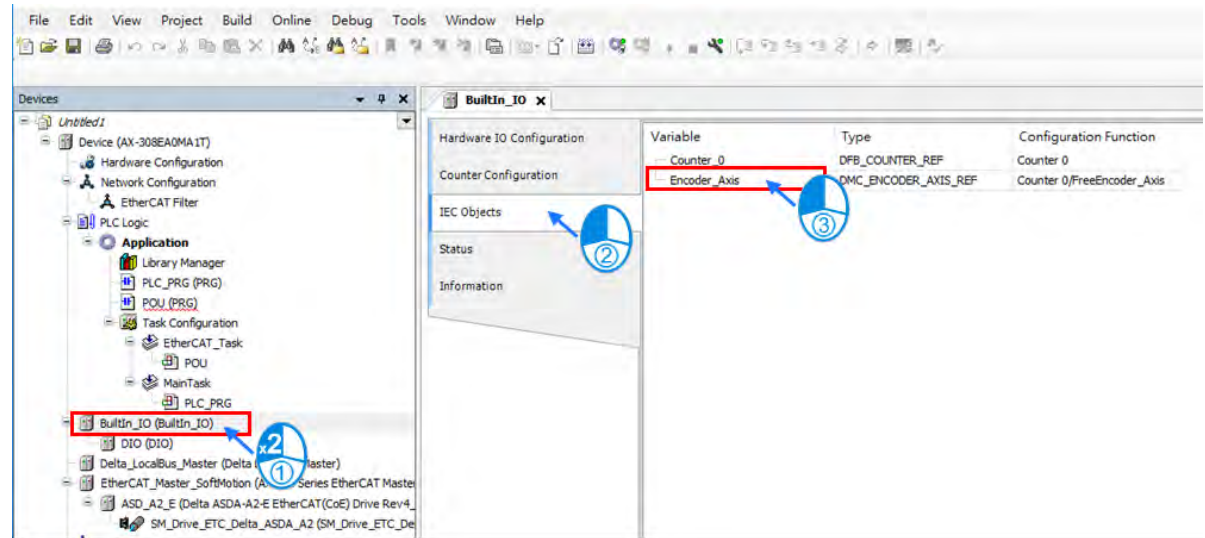
22. 将编码器名称填入「DFB_HCnt」的 Counter 引脚中并删除引脚中的「???」，因「???」不为任何变数，在编译时会出错。(编码器需要使用 DFB_HCnt 功能块启动计数功能)



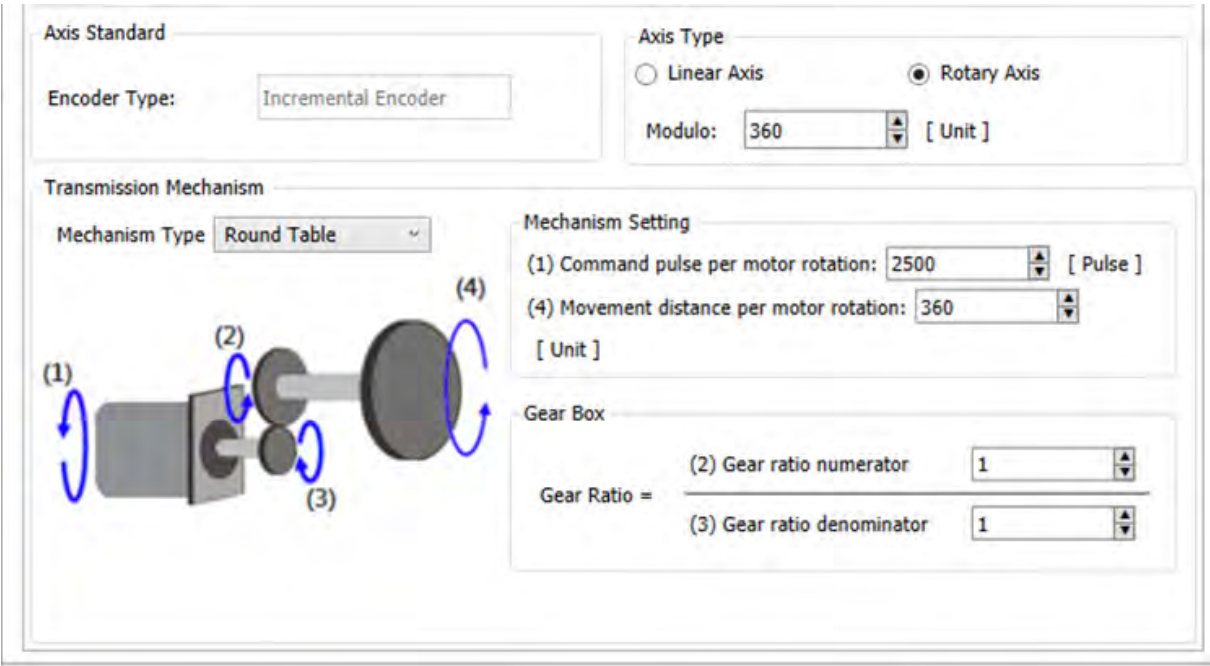
23. 建立「MC_Home」功能块，让实轴回原点，编码器需要通过「MC_SetPosition」清除计数值。



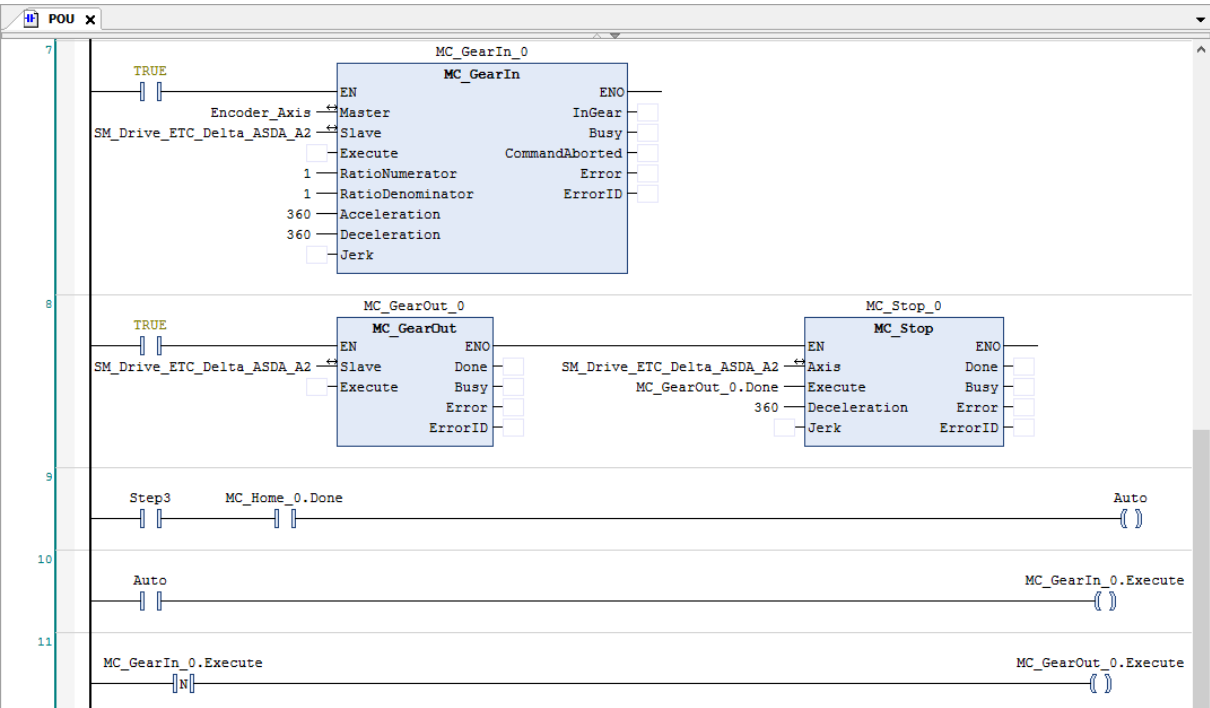
24. 以编码器当主轴时，需要将 Encoder_Axis 输入至「MC_GearIn」的 Master。



25. DFB_HCnt 计数的数值会通过 Encoder_Axis 所设定的参数，如下图所示，进行转换为用户单位。



26. 以「MC_GearIn」让实轴追随编码器，当关闭「Step3」时「MC_GearOut」会一起执行，当脱离追随后，从轴会等速运转，需使用「MC_Stop」停止运转。

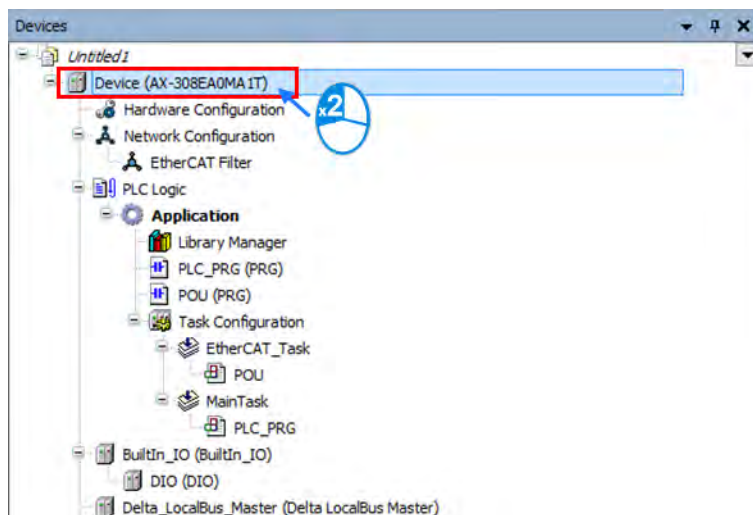


3.5.3 程序监控

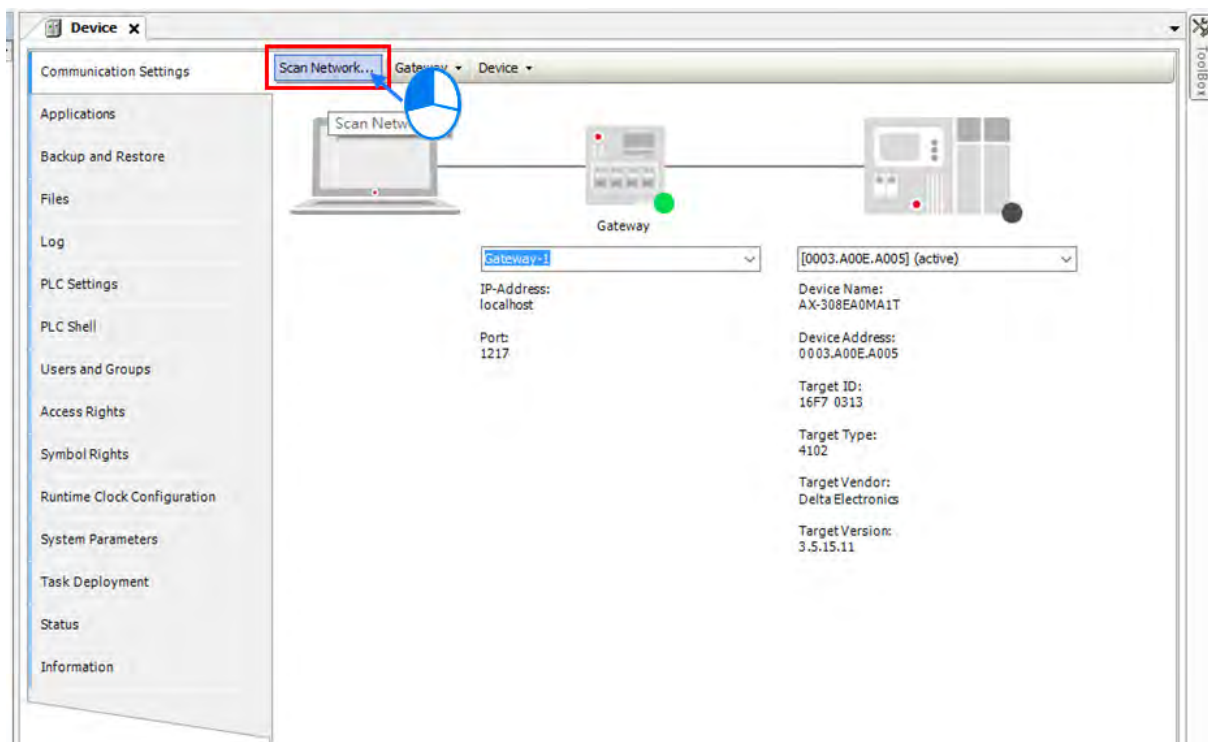
当程序运行的时候，用户可以通过监视的方式，来了解系统目前的逻辑控制状态，或是修改部分装置的数值来进行系统的测试，现在我们就来进行程序监视的操作。

3.5.3.1 设备与计算机链接设定

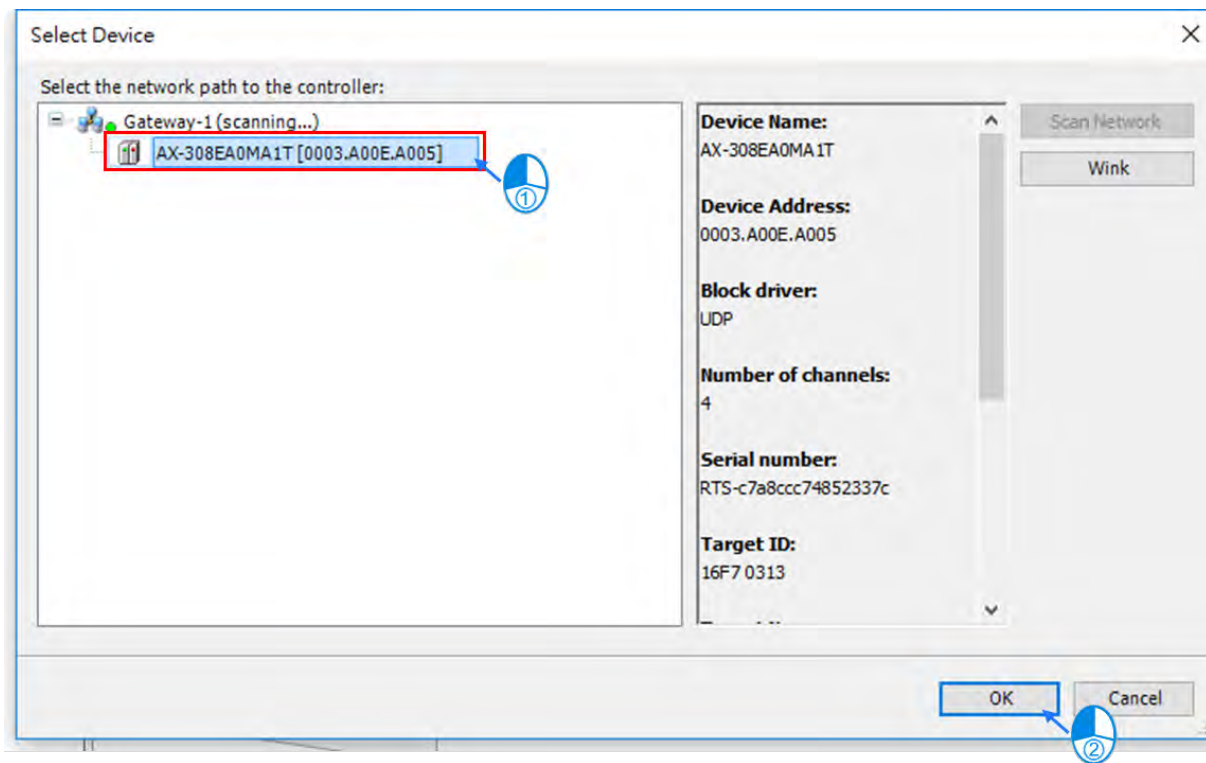
1. 单击「Device」后双击鼠标左键（AX-308E 预设 IP 为 192.168.1.5）



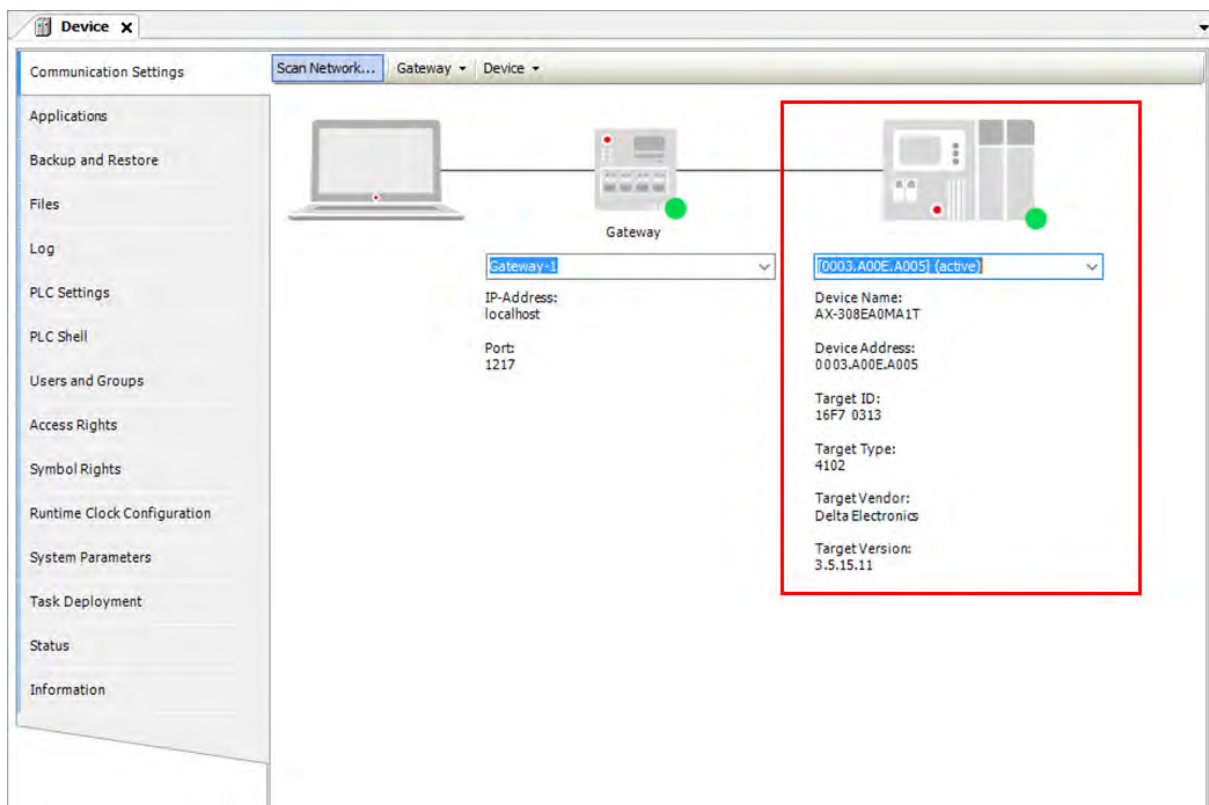
2. 进入「Device」画面后选择「Scan Network」。



3. 在「Select Device」画面中，单击「AX-308E」 > 「OK」。

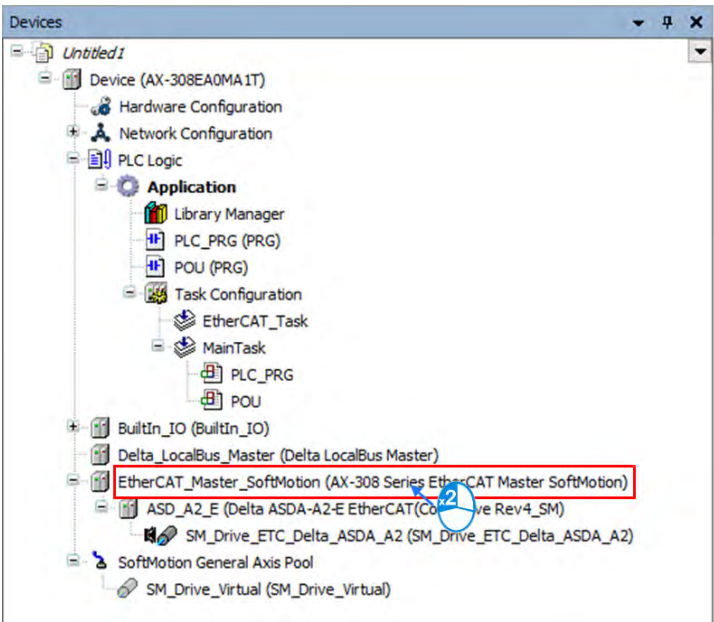


4. 当 PC 与设备连结后，会出现设备相关的信息，如下红框所示。

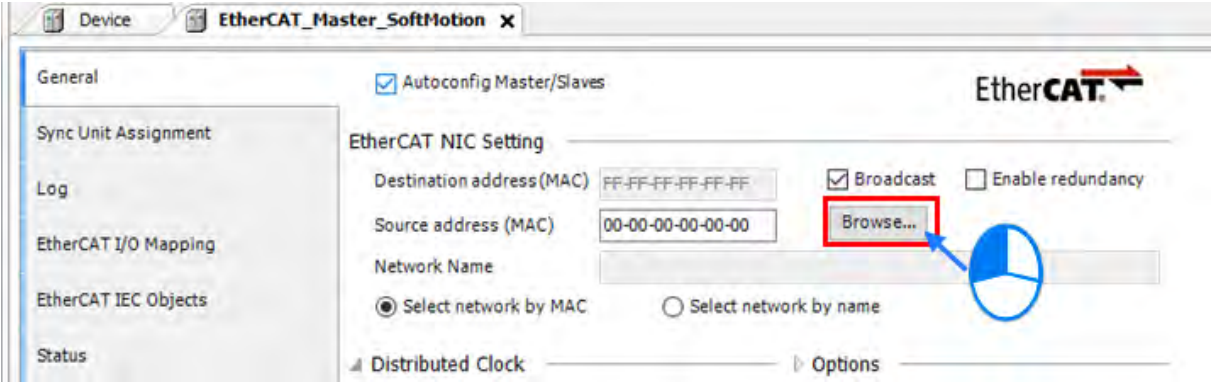


3.5.3.2 程序监控流程

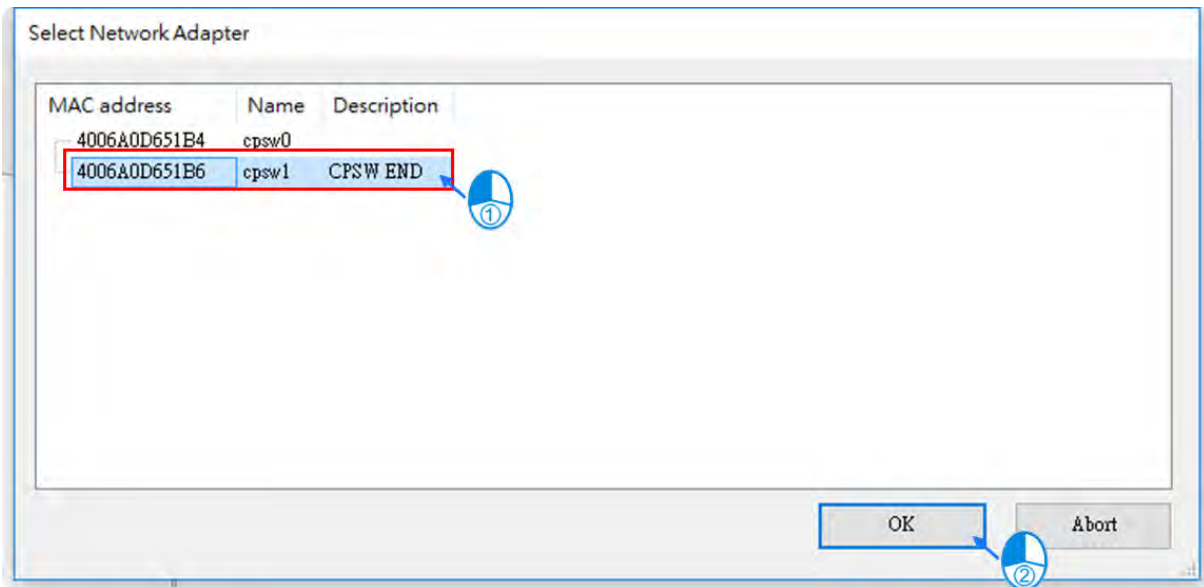
1. 监控程序前需确认 EtherCAT 通道是否建立正确，确认后再双击「EtherCAT_Master_SoftMotion」。



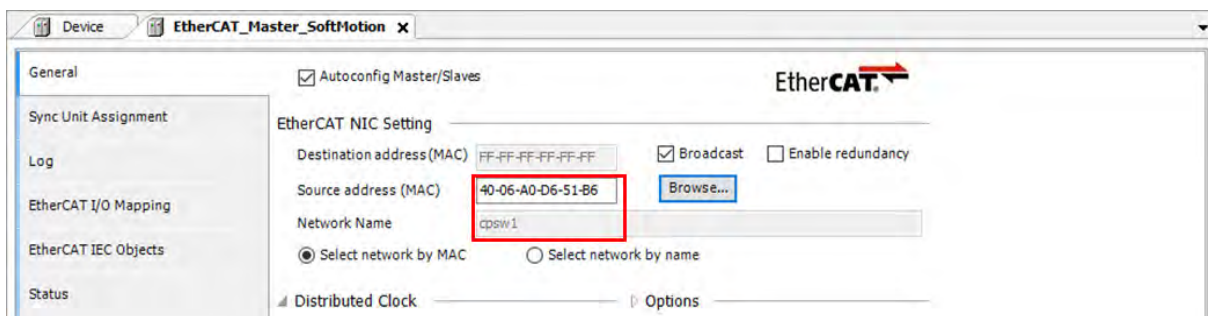
2. 进入「EtherCAT_Master_SoftMotion」画面后，单击「Browse」。



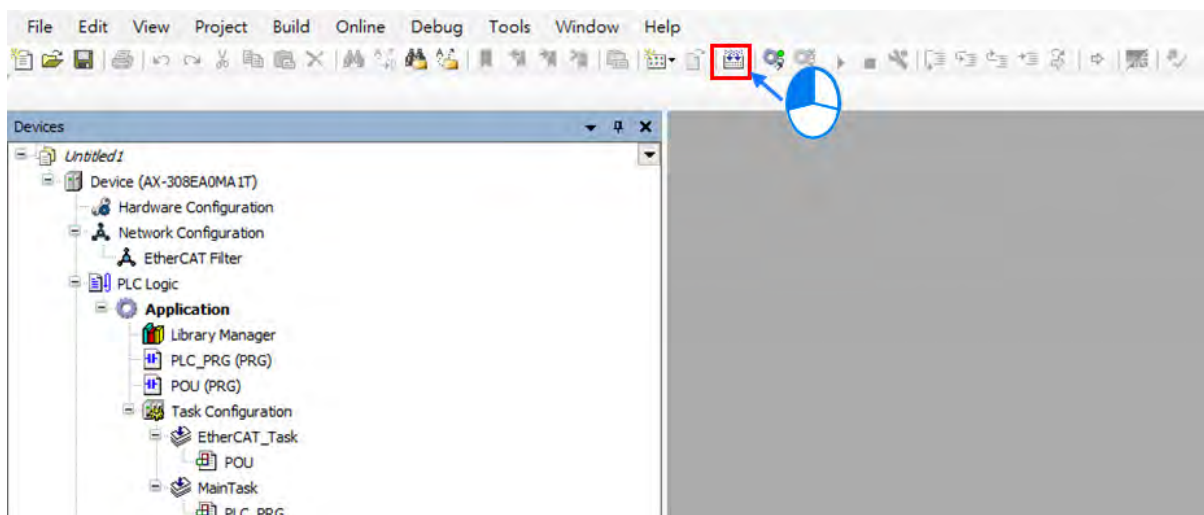
3. 需选择 EtherCAT 网络端口「cpsw1」后，单击「OK」。



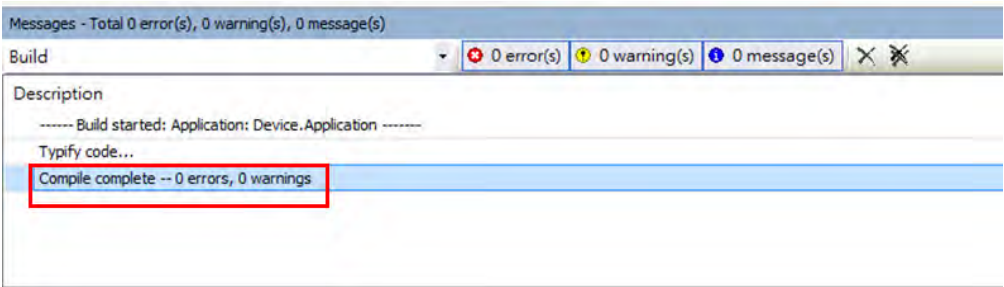
4. 选择完毕后，「MAC」会出现设定值。



5. 监控前先单击  编译以确认程序是否正确。

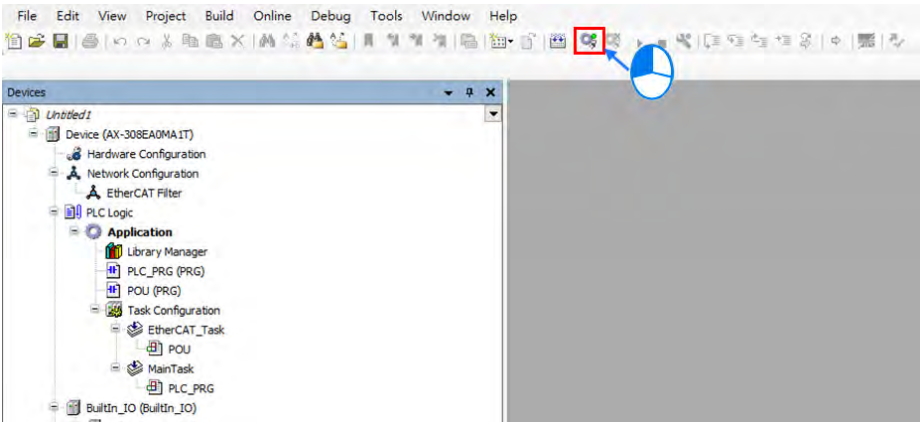


6. 编译完成后，可确认程序编写是否有错误。

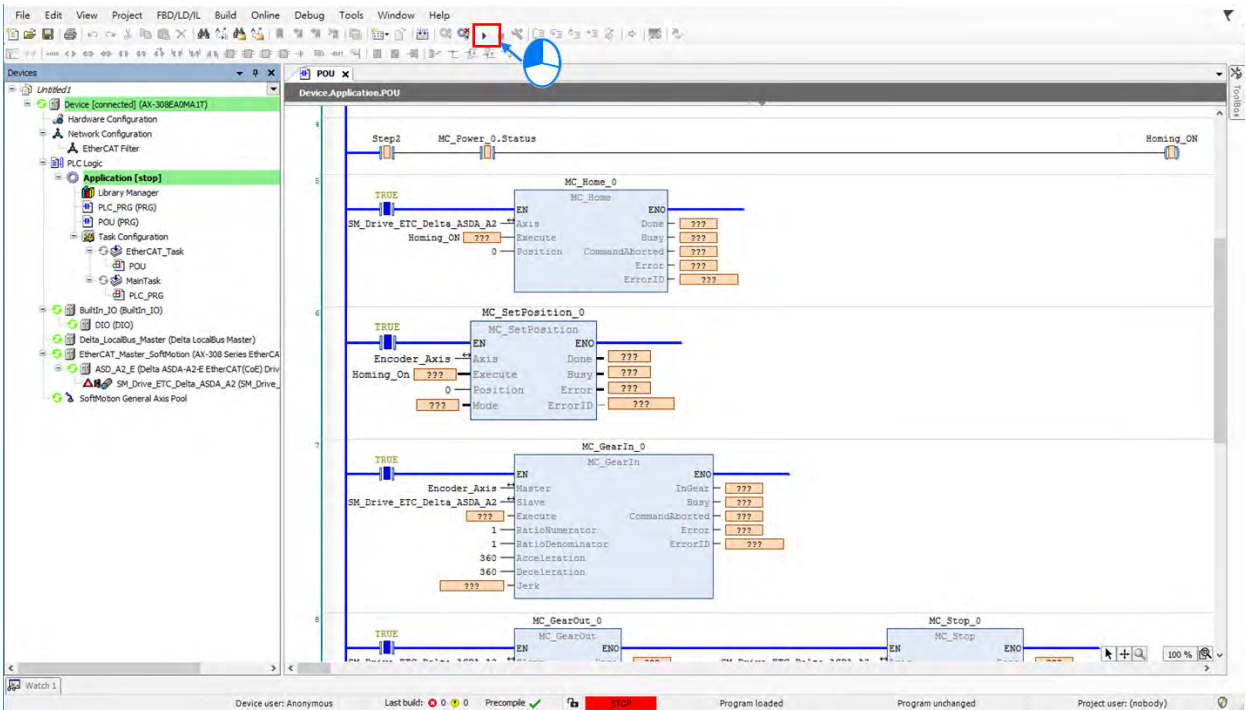


7. 当编译完成后，再单击  执行在线监控。

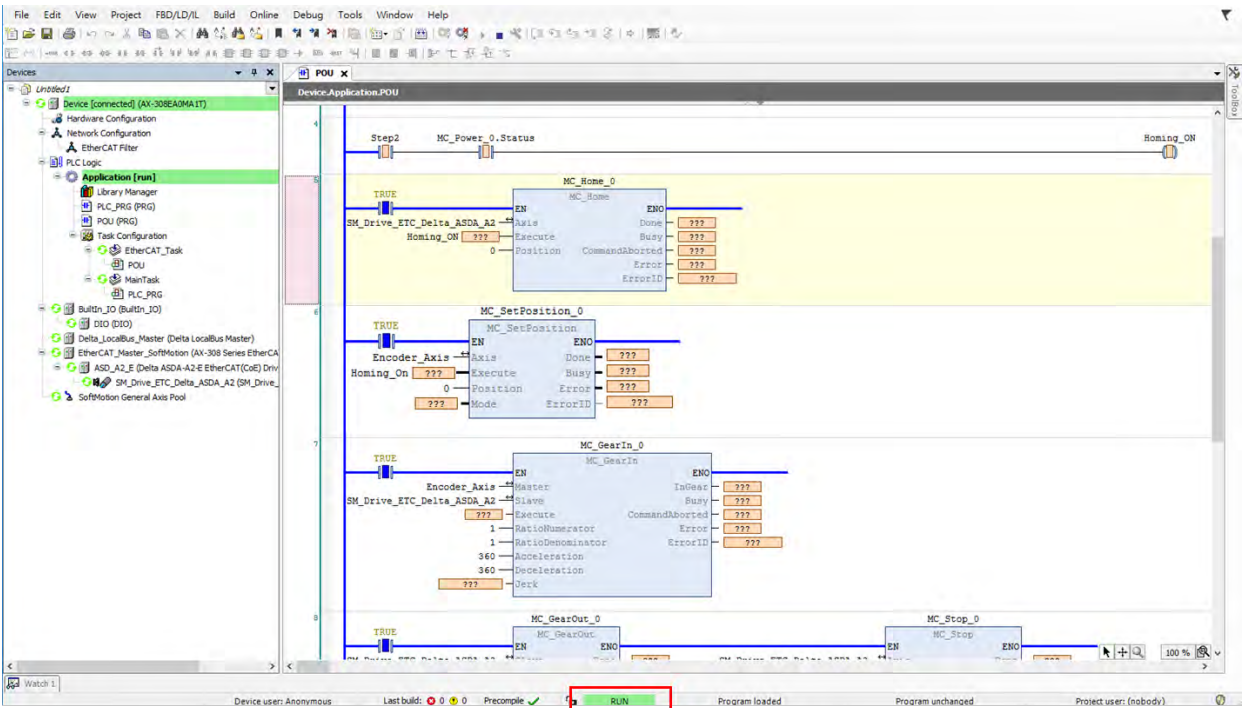
3



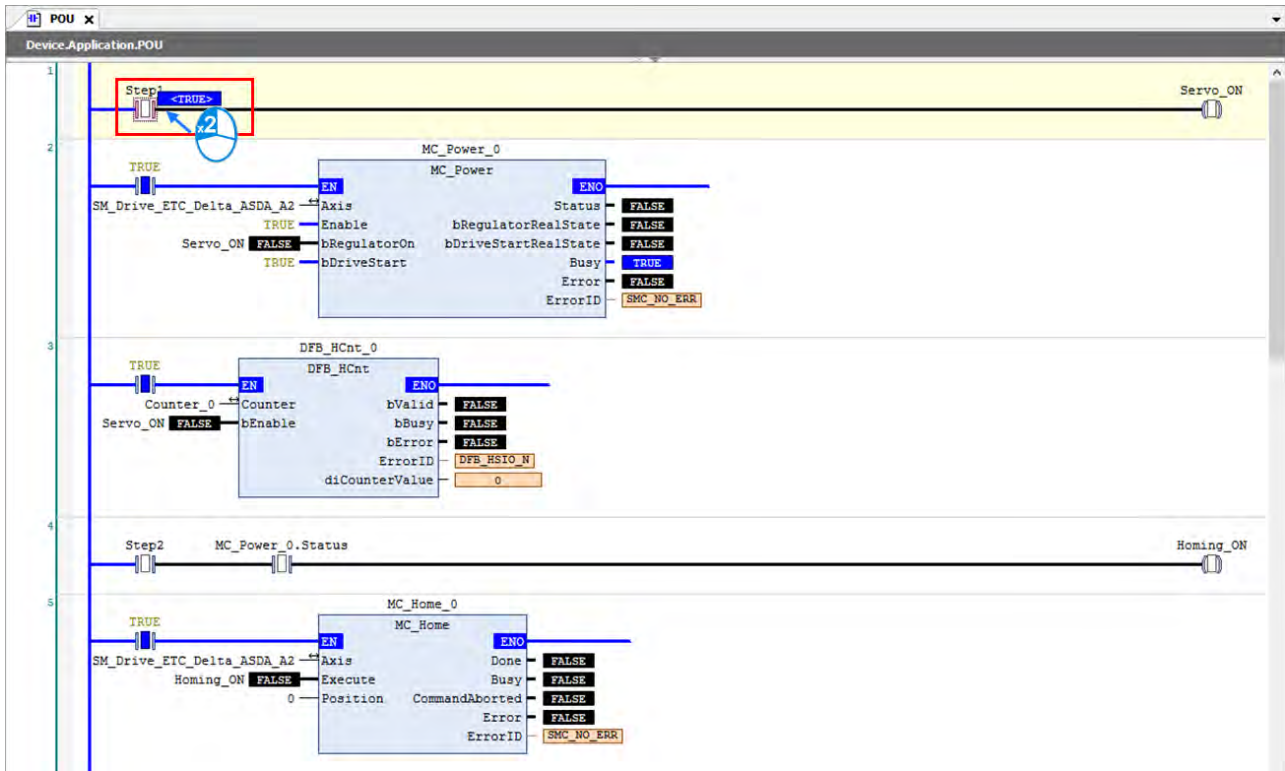
8. 执行在线监控后需下载程序，下载完成后 PLC 状态会再 Stop，需单击  PLC 才会执行 Run。



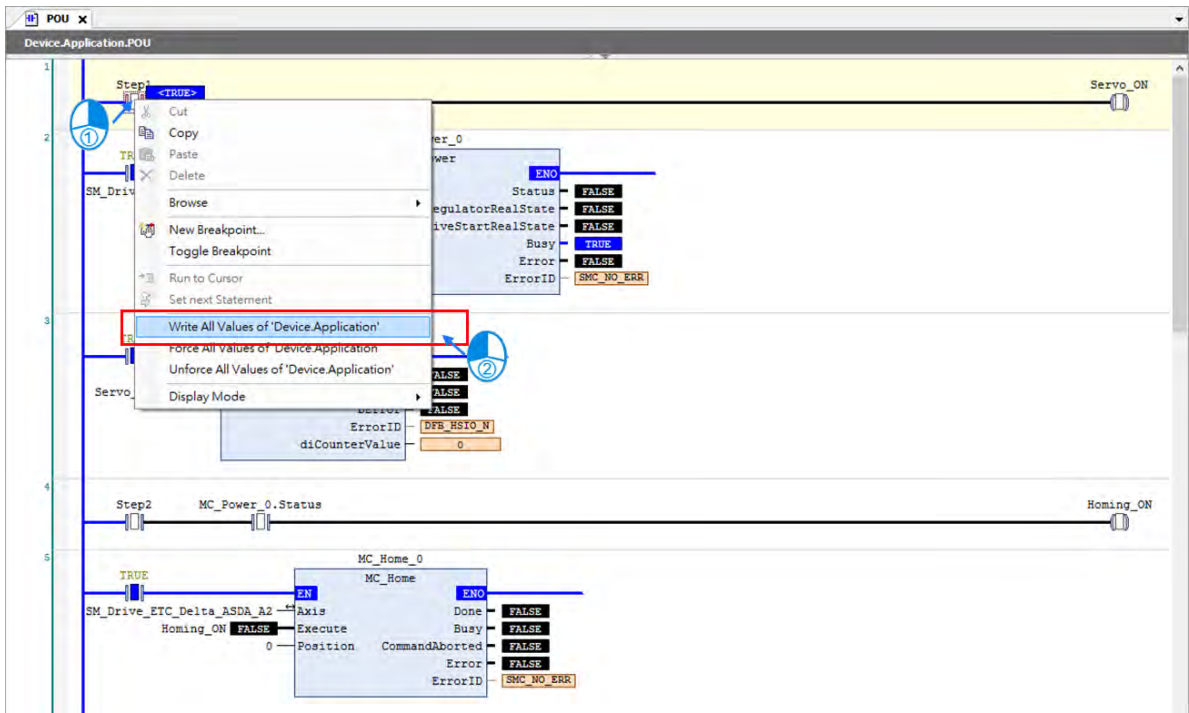
9. 当 PLC 开始 Run 时下方的状态会改为 **RUN**，此时可开始执行程序。



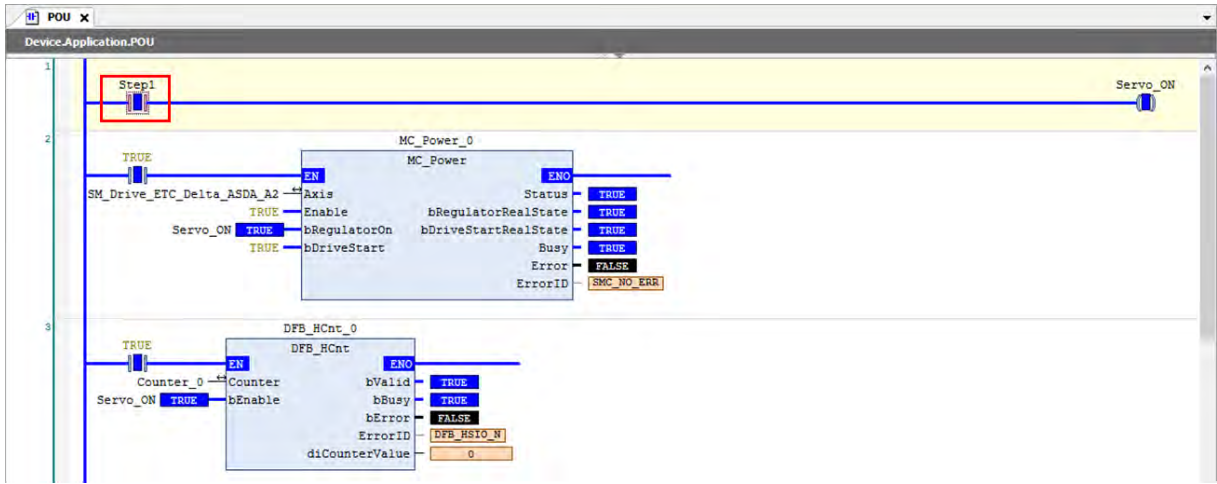
10. 在线监控后，双击「Step1」后会出现 **<TRUE>** 的图示。



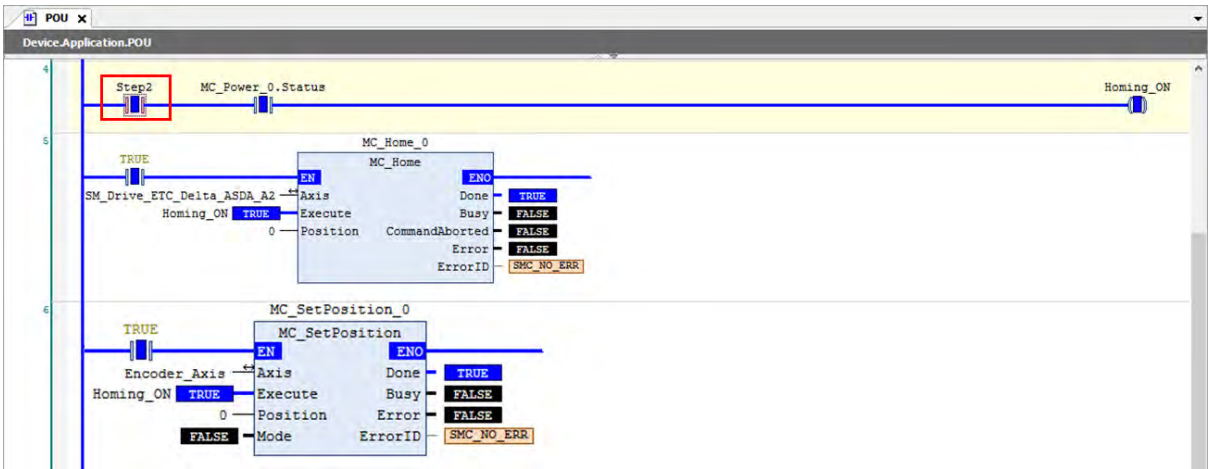
11. 单击「Step1」后右键单击后，选择「Write All Values of “Device.Application”」，会写入程序中所有状态。(可使用快捷键 Ctrl + F7)



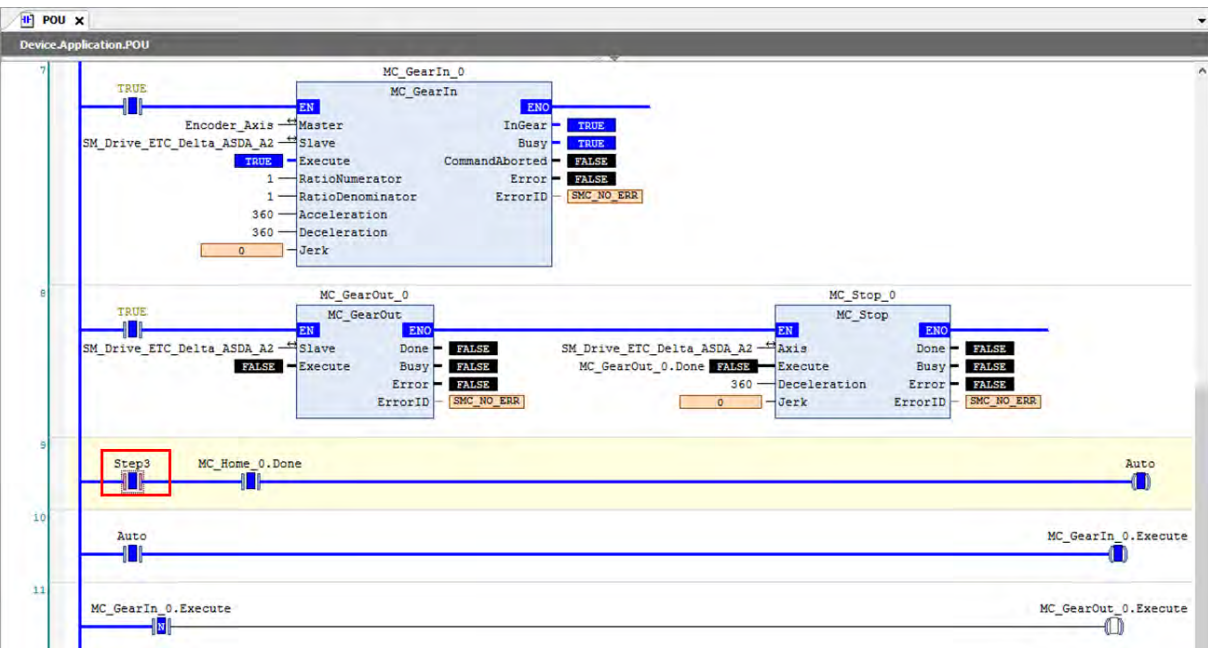
12. 启动「Step1」后，就会 Servo ON 并启动编码器功能。(以相同操作执行以下程序)



13. 启动「Step2」后使轴回原点并将编码器轴数值改为 0。

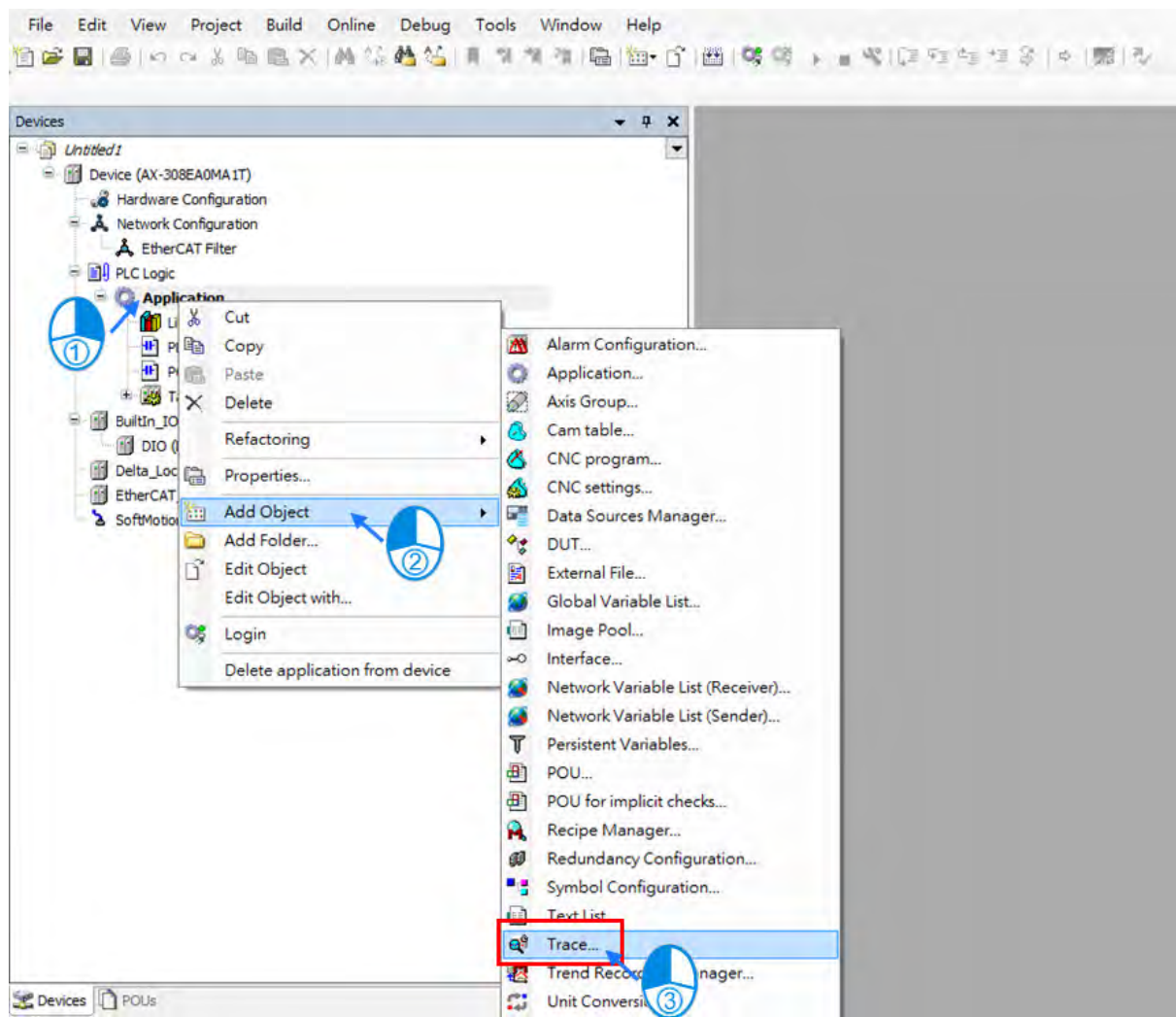


14. 启动「Step3」后，启动从轴追随主轴，此时只要转动编码器，从轴就会跟随；如将「Step3」关闭后，会触发「MC_GearOut」，当「MC_GearOut」的 Done 为 True 时，再触发「MC_Stop」。(当主从轴脱离从轴会以当下速度运转，需以「MC_Stop」停止从轴运转)

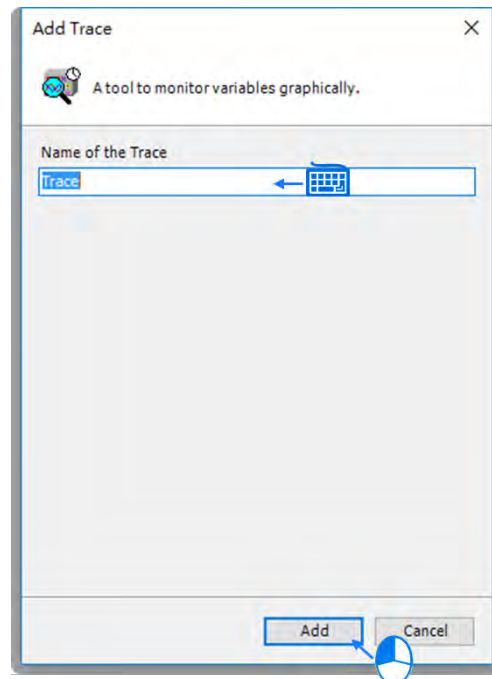


3.5.3.3 示波器监控

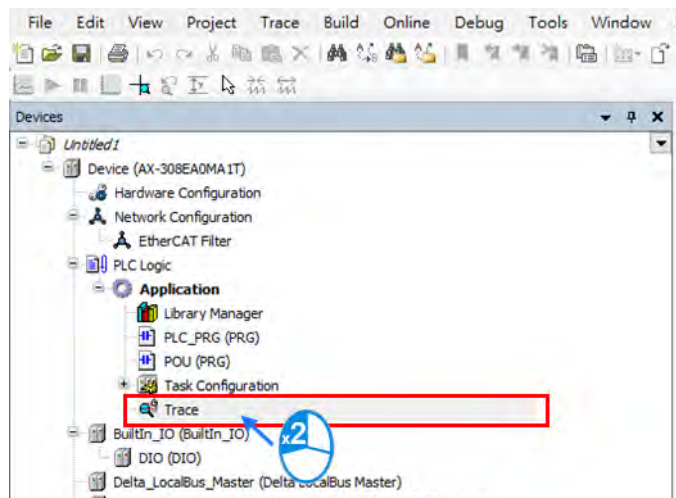
1. 当程序在运行过程中可以使用 Trace 监控的方式，确认引脚状态及轴状态。依序单击「Application」>「Add Object」>「Trace」。



2. 进入「Add Trace」画面中，可以输入 Trace 的名称后再单击「Add」。



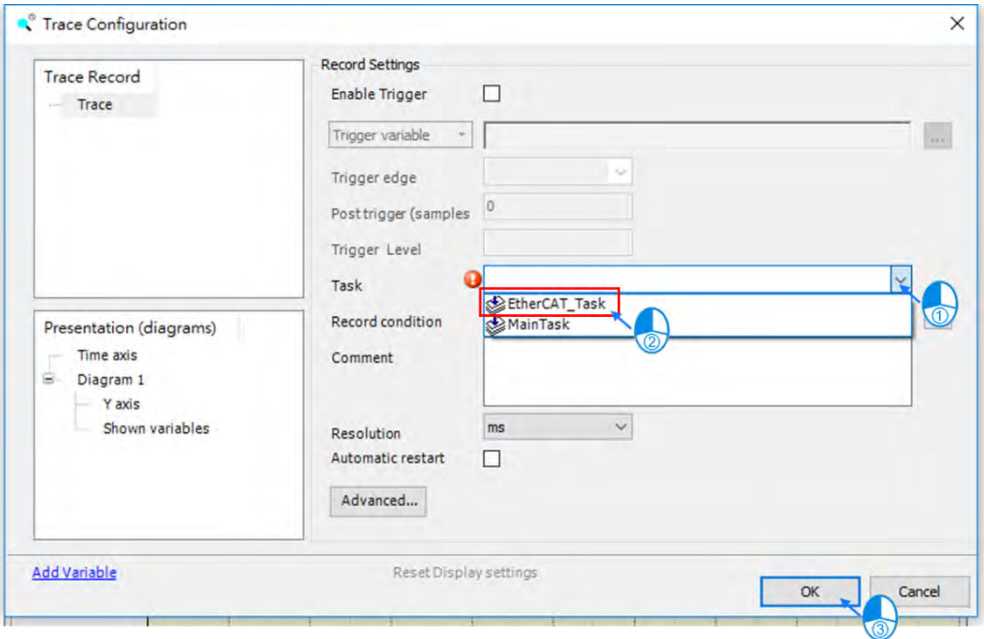
3. 新增完毕后，左边树形图会出现刚刚所新增的示波器，此时双击「Trace」。



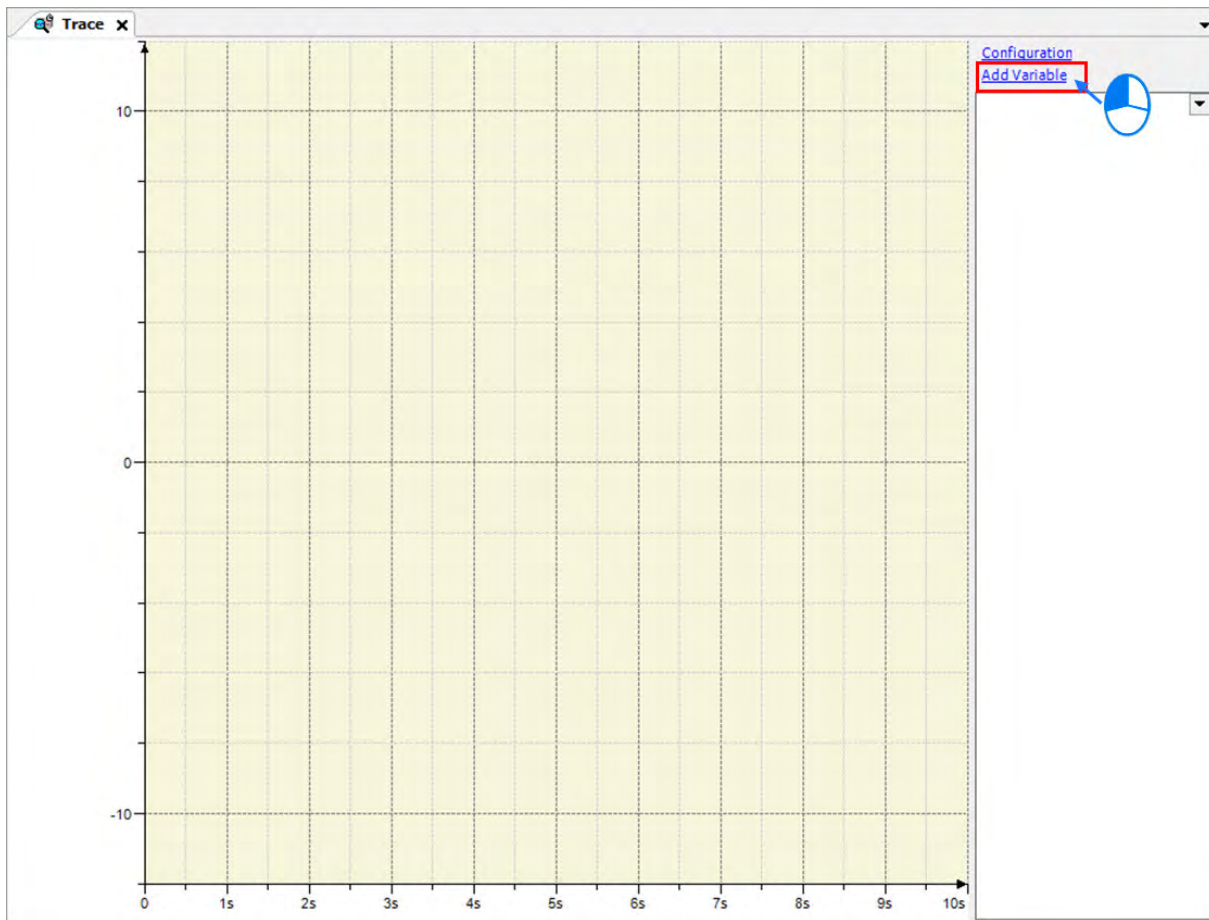
4. 进入「Trace」画面后，单击「Configuration」。



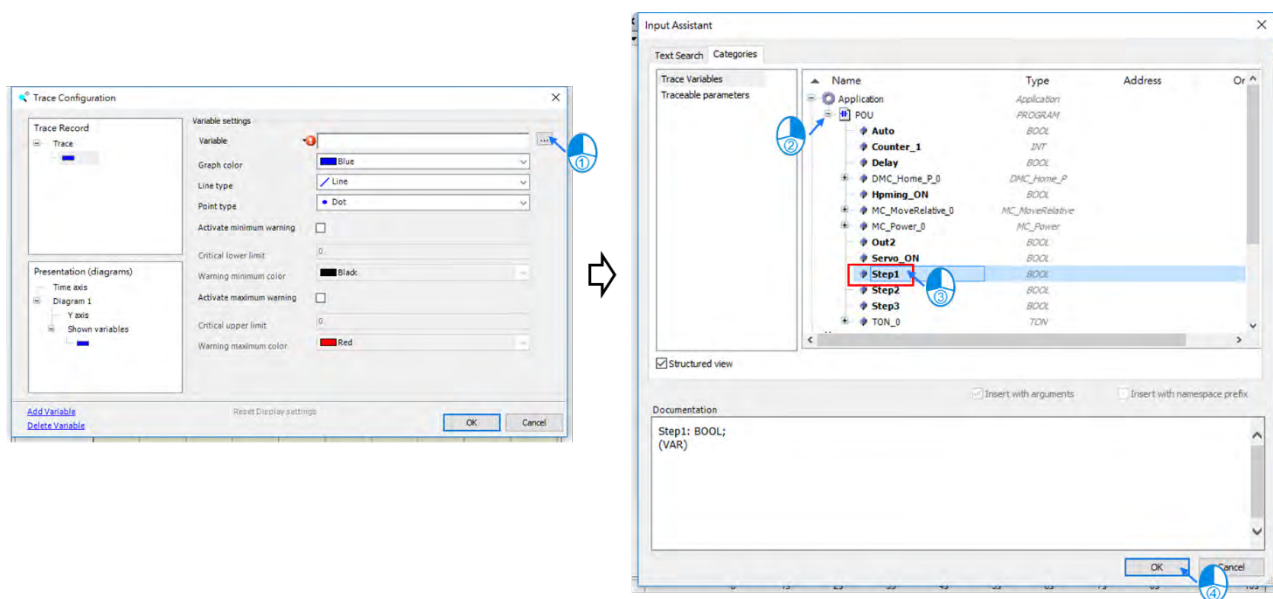
5. 在「Trace」设定画面中，选择「Task」下拉选单中的「EtherCAT_Task」。



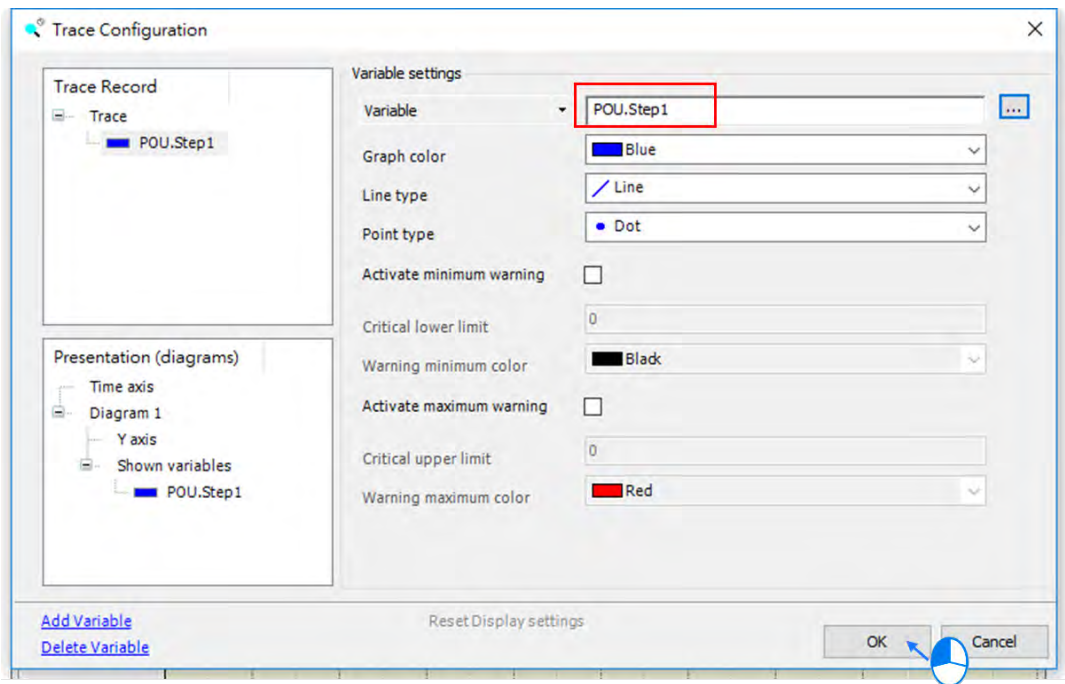
6. 设定完后单击「Add Variable」。



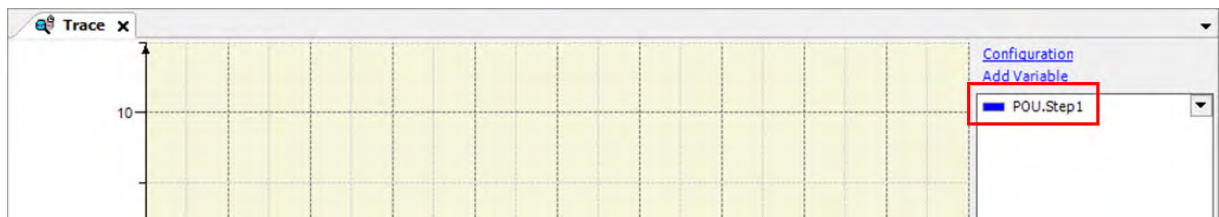
7. 进设定画面后，单击 [...] 添加需监控的参数，当进入添加画面后，单击 [+] 后选择「Step1」>「OK」。



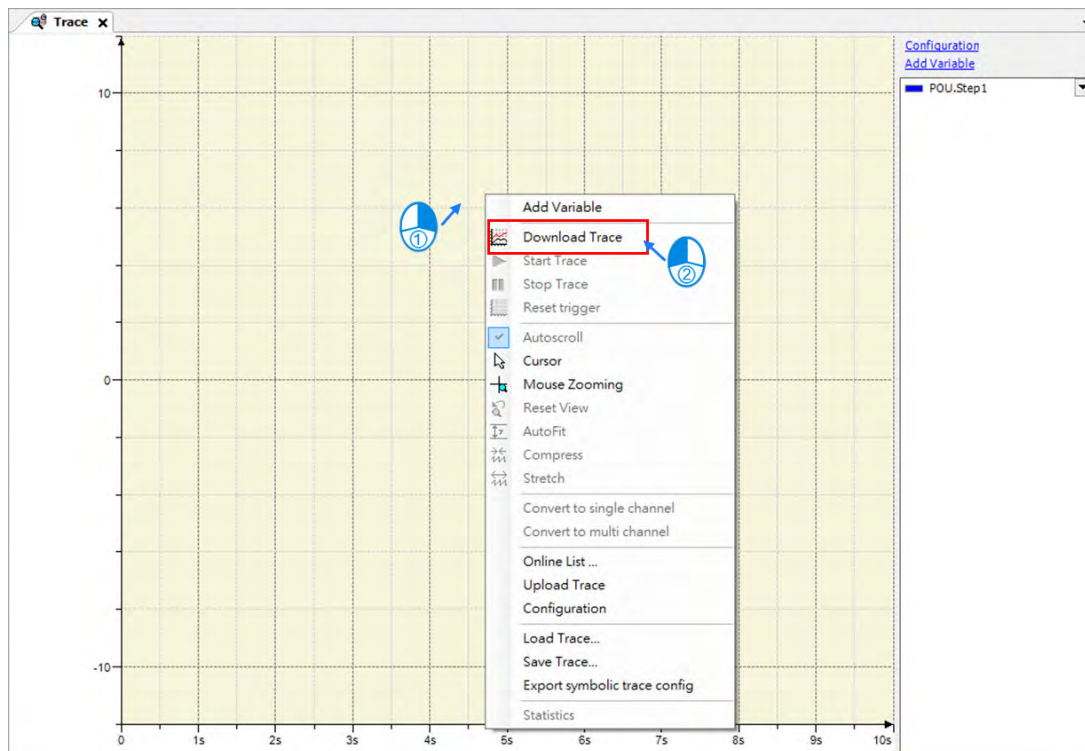
8. 选择好需监控的参数会出现在表格中再单击「OK」。



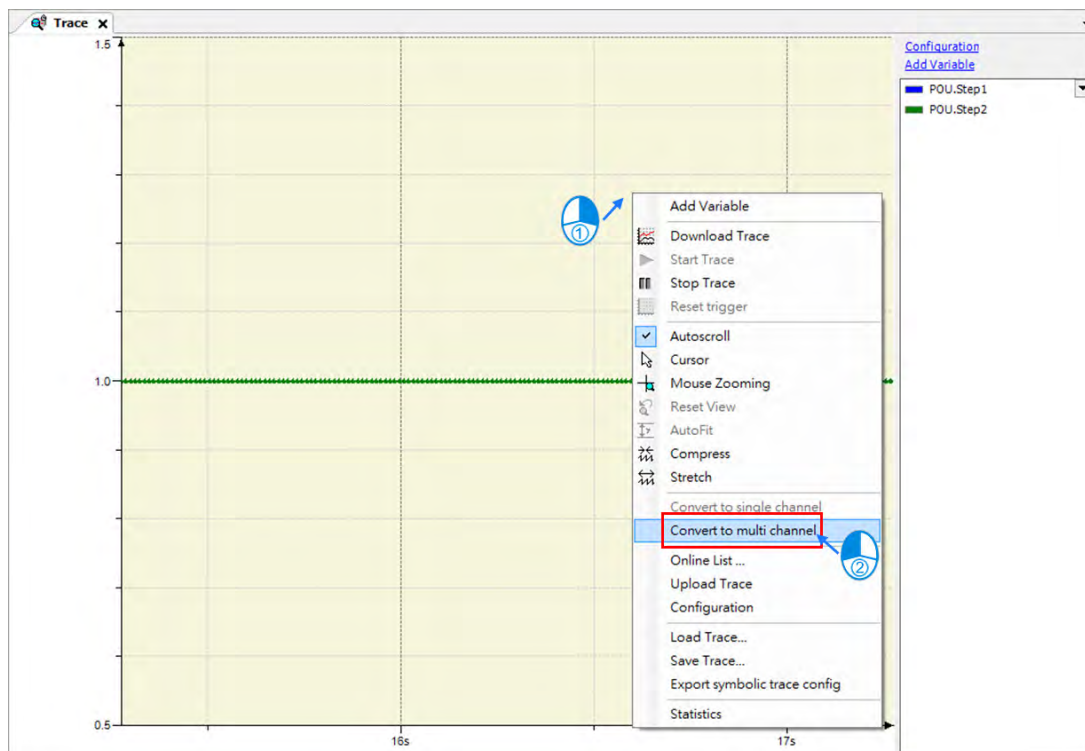
9. 添加完成后，在「Trace」画面右方会出现需监控的参数。



10. 在在线监控时单击「Trace」右键后，选择「Download Trace」就会开始监控。



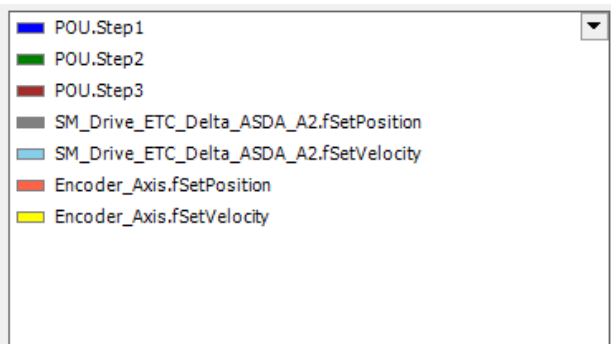
11. 当有两个参数需要监控时，可以在「Trace」画面单击右键，选择「Convert to multi channel」。



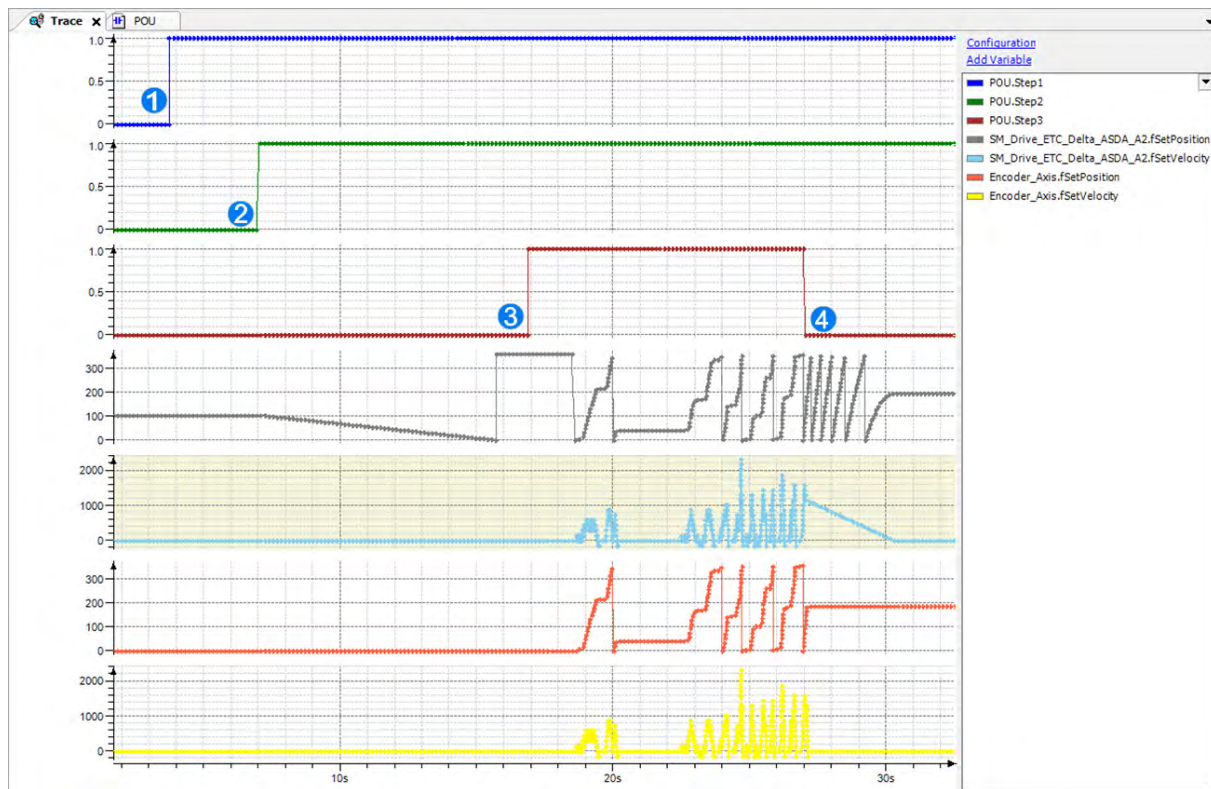
12. 「Convert to multi channel」选择后会以两格的方式显示。



13. 利用上方相同操作新增下列监控值。



14. 通过程序执行过程中，可使用示波器观察触发的开关及轴目前位置及速度。



示波器流程介绍：

- ①启动「Step1」让轴及编码器启动。
- ②启动「Step2」让轴回原点并将编码器轴数值清除。
- ③启动「Step3」后会触发「MC_GearIn」，让「EtherCAT」轴追随编码器轴，此时旋转主轴时，从轴会追随。
- ④关闭「Step3」后会触发「MC_GearOut」，当「MC_GearOut」输出引脚 Done 为 True 时，触发「MC_Stop」让从轴减速停止运转。