



—
AC500 PLC 系列

应用笔记

AC500 v3 Motion 入门。

AN500 型



EtherCAT® 

内容

1	介绍	2
2	伺服驱动器准备	3
2.1	设置 ABB 伺服驱动器以与 EtherCAT 主站配合使用	3
2.1.1	e180 和 e190 的 EtherCAT 设置过程	3
2.1.2	e530 EC 变频器的 EtherCAT 设置流程	4
2.2	获取 xml 文件	4
2.2.1	e180 和 e190 的过程 - 从驱动器导出 xml 文件。	4
2.2.2	从网站获取 e530 EC xml 文件的过程。	4
3	运动解决方案项目	5
3.1	在开始项目之前要考虑的项目。	5
3.1.1	选择正确的 PLC - PLC 功能简介	5
3.1.2	更新设备存储库中的设备描述文件	5
3.2	运动解决方案项目配置	6
3.2.1	创建新项目	6
3.2.2	将轴添加到项目中	7
3.2.3	配置 EtherCAT 主站	9
3.2.4	配置 EtherCAT 从站和轴	10
3.2.5	轴设置	11
3.2.6	重要 常规 PLC 设置	19
3.2.7	生成运动配置	19
3.2.8	任务配置	19
3.3	编写应用程序	22
3.3.1	设置编程介绍项目	22
3.3.2	程序编写	24
3.3.3	可视化（可选）	26
3.3.4	跟踪（可选）	29
3.3.5	检查 CPU 参数。	31
3.3.6	登录 CPU 并下载程序。	32
3.3.7	测试程序并使用程序控制轴。	33

2 介绍

介绍

本应用说明简介中详细介绍了如何使用 Automation Builder for v3 AC500 PLC ([AB v2.7.0 及更高版本](#)) 来定义适合单个 ABB 伺服驱动器的运动控制的硬件设置，以及如何编写一个简单的 AC500 v3 PLC 程序来在这些驱动器上执行运动。

这里有一个[更详细的文档](#)，其中更详细地介绍了提到的所有步骤，以及此处未涵盖的其他一些相邻功能和运动相关指南。

先决条件

- 最新的软件工具版本可以[在这里下载](#)
- 最新的硬盘固件可[在此处下载](#)
- PLC 程序是使用 **Automation Builder v2.7** 或更高版本编写的，可[在此处下载](#)
- ABB PLCopen v3 运动控制库的已安装（和许可）副本
- 以下 AC500 PLC 处理器之一：

AC500 运动套件：PM5630-2ETH、PM5650-2ETH、PM5670-2ETH、PM5675-2ETH

AC500eco 运动套件：PM5032-ETH、PM5052-ETH、PM5072-2ETH、PM5082-2ETH

- PLC 处理器应运行固件版本 3.2 或更高版本 – 如果 PLC 附带了 v3.0 固件，请首先使用 SD 卡，然后使用 Automation Builder 中的“更新固件”功能，以根据需要进一步更新处理器。
- 用于将 PLC EtherCAT 耦合器连接到驱动器的以太网电缆

运行此示例所需的 PLC 硬件 是 **ABB AC500 V3 PLC 系列**。它们有两种主要类型：AC500 和 AC500eco。所有 AC500 V3 CPU 都有一种名称，在代码的 PM 部分后有 4 个数字，例如 **PM5072-T-2ETH**，而上一代 CPU 只有 3 个数字，例如 PM564-ETH。

保修、责任：

用户应对本文件中描述的本产品的使用负全部责任。ABB 不作任何保证。ABB 对本文件中提供的产品或示例或文件中包含的文件的应用不承担任何责任，无论其法律依据如何。责任的免除不适用于故意或重大过失的情况。预先发送的声明应受瑞士法律管辖并按其解释，但不包括其法律冲突规则和《维也纳国际货物销售公约》（《销售公约》）。

1 伺服驱动器准备

1.一 设置 ABB 伺服驱动器以与 EtherCAT 主站配合使用

本部分假定您已经调试了驱动器。即，您已经完成了调试过程，并调整了驱动器的控制回路。



有关调试 e180 和 e190 变频器的详细信息，请参见应用笔记 AN250，可[在此处下载](#)。
e530 驱动器的类似应用说明正在开发中，并将很快[在此处提供](#)。

1.一.一 e180 和 e190 的 EtherCAT 设置过程

一. 这里的第一步是检查旋转开关是否设置正确。如下图所示，它们可以定义多种操作模式，对于 EtherCAT 模式，应设置为 00H

	Value	Mode
HI	00	EtherCAT slave mode
	01-EF	Ethernet POWERLINK CN mode: selected value is node ID
LO	F0-F1	Reserved
	F2	PROFINET slave mode
	F3-FF	Reserved

二. 接下来，我们必须定义正确的 ControlRefSourceStartUp，它定义了驱动器通电或重新启动时的初始操作模式。可以使用多种方法设置控制参考源启动。

- 参数查看器 - 参数系列 = '配置' > 参数 = 'CONTROL REF SOURCE STARTUP > 设置 = RT_ETHERNET_ (CiA402)
- 命令行 - using command = 'CONTROLREFSOURCESTARTUP (0) = crsRT_ETHERNET_402'
- 向导 - 从主视图中选择“操作模式”然后设置 “ControlRefSource = RT 以太网 (CiA 402)”： 按“完成”按钮



三. 最后一步始终是使用工具栏中的保存图标保存参数。

1.1.2 e530 EC 驱动器的 EtherCAT 设置流程

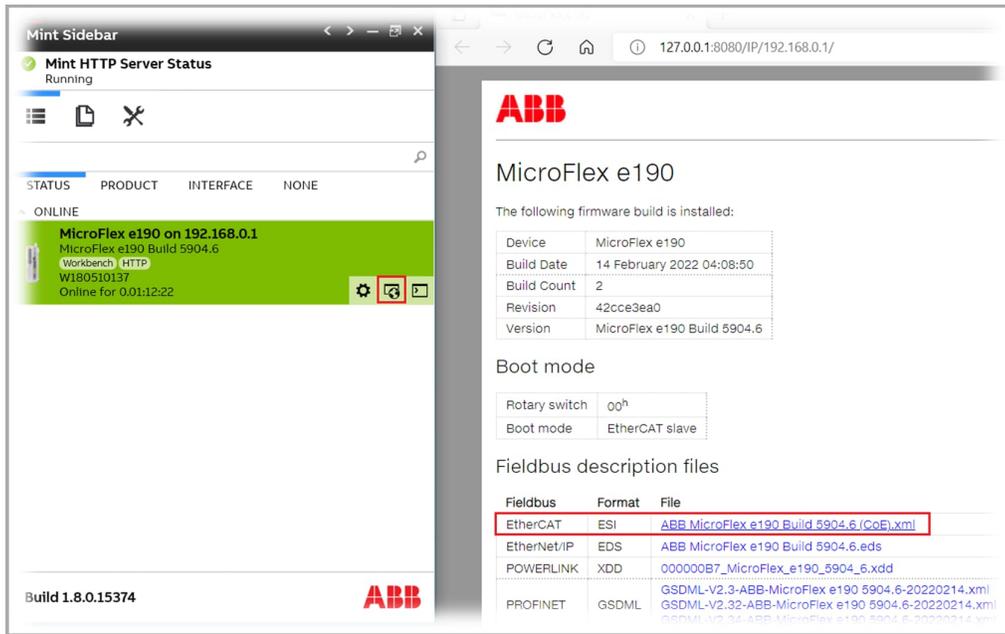
对于 e530 EC 驱动器，它们开箱即用，无需特殊的调试步骤即可为 EtherCAT 控制做准备。

1.2 获取 xml 文件

1.2.1 e180 和 e190 的过程 - 从驱动器导出 xml 文件。

在开始新的 PLC，项目和进行初始配置导航之前，我们需要确保我们需要的 E190 的相应.xml 文件已安装到“设备存储库”中，如果没有，请按照以下过程操作：

- 打开 Mint 侧边栏
- 连接到 e190，这将使它以绿色显示在列表中
- 然后单击“Web”图标。这将打开 Web 服务器
- 在主页底部显示与已安装固件版本关联的一系列文件
- 选择 EtherCAT 文件以启动.xml 文件下载到您的硬盘：



1.2.2 从网站获取 e530 EC xml 文件的过程。

访问 ABB 网站（[伺服产品 - 为机器创新提供动力 - 低压交流电 | ABB](#)）并下载最新版本。

2 运动解决方案项目

2.1 在开始项目之前要考虑的项目。

2.1.1 选择正确的 PLC - PLC 功能简介

在选择开始项目之前，您应该根据其功能选择正确的 PLC。以下是 AC500 和 AC500eco 产品的简要介绍。

2.1.1.1 PTO 操作

由于 CPU 类型的性能不同，每种 PLC 类型中可配置的最小循环时间限制也不同。

PLC 型	PM5032 型	PM5052 型	PM5072 型	PM5082 型
PTO 分析周期时间限制	5 毫秒	2 毫秒	1 毫秒*	1 毫秒

2.1.1.2 EtherCAT 操作限制

由于 CPU 类型的性能不同，EtherCAT 周期时间设置有不同的限制。可在每种 PLC 类型中配置最短 EtherCAT 循环时间。

PLC 型	PM5072 型	PM5082 型	PM5630 系列	PM5650 系列	PM5670 型
可配置的最小 EtherCAT 主循环时间 (ms)	2 毫秒	1 毫秒	2 毫秒	1 毫秒	1 毫秒

2.1.1.3 每种 PLC 类型中可配置轴的总最大数量。

除了对每种 PLC 类型的最短循环时间的限制外，根据配置的循环时间，对每种 PLC 类型可以连接的轴数也有一个链接配置限制。有关详细信息，请参阅帮助文件（使用 V3 CPU 的 PLC 自动化 >库和解决方案 >运动控制库 >使用库的前提条件），并相应地调整循环时间或 PLC 类型。

PLC 型	PM5072 型	PM5082 型	PM5630 系列	PM5650 系列	PM5670 型
同步器数量。轴在 0.5 ms 内完成	-	-	-	-	2
同步器数量。轴在 1 ms 内完成	-	4*	-	8	16
同步器数量。轴在 2 ms 内完成	4	8	4	16	32
同步器数量。轴在 4 ms 内完成	8	16	8	32	64

“运动解决方案”窗口可用于查看在 EtherCAT 主循环时间内支持的轴数以及特定 PLC 类型的已使用轴数，方法是选中运动解决方案向导概述页面底部的滑块：

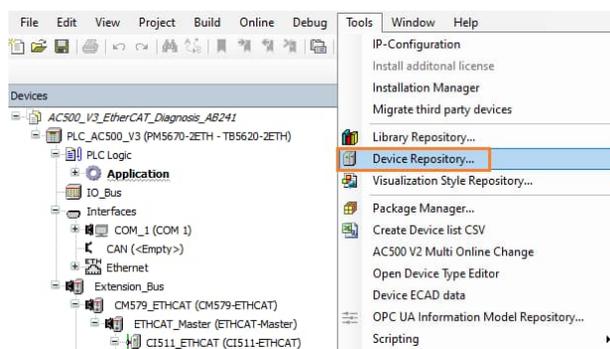


2.1.2 更新设备存储库中的设备描述文件

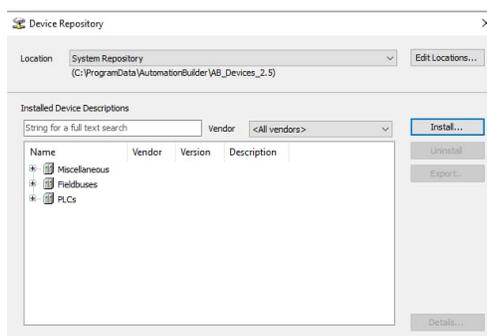
应考虑项目中的设备描述文件，如果它们与固件版本不匹配，则可能会发现缺少一些映射或功能。将 ABB 或第三方硬盘添加到 Automation Builder 设备存储库的过程是相同的。

若要安装设备描述文件，请按照下列步骤操作：

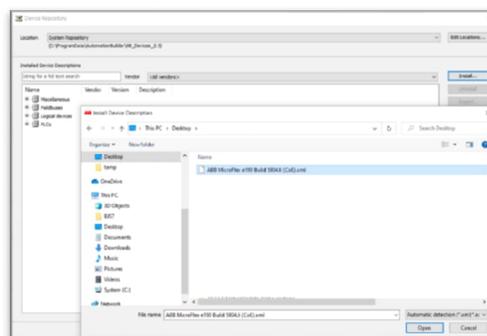
- 一. 单击“工具”菜单下的“设备存储库”。



- 二. 这将打开“设备存储库”窗口，然后单击“安装”并选择需要安装的设备描述文件。



- 三. 单击“安装...”，然后找到所需文件的存储位置。选择后，请单击“打开”。



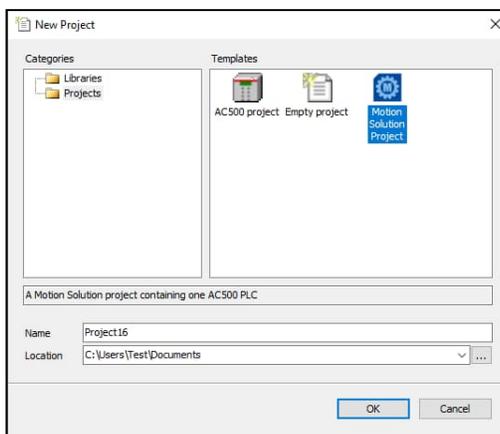
- 四. 文件安装的状态将显示在窗口底部。如果成功，设备将立即出现在已安装设备列表中，安装成功后，用户现在可以在 Automation Builder 中配置的相应协议下添加已安装的设备。

- 五. 完成此步骤后，我们可以在新项目中使用此版本。

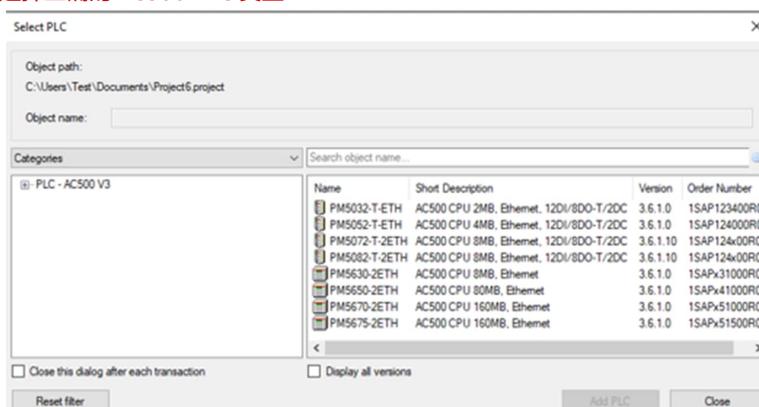
2.2 运动解决方案项目配置

2.2.1 创建新项目

- 一. 要启动 Automation Builder 项目，请打开 AB 并选择“新建项目”。
- 二. 选择“Motion Solution Project”图标，如下所示。单击“确定”按钮，将在指定位置创建一个具有指定名称的新项目。



三. 添加 PLC - Automation Builder 现在将弹出一个“选择 PLC”窗口，用户可以从这里选择一个 V3 PLC。有关正确的选择，请参阅以下部分：[选择正确的 AC500 PLC 类型](#)



四. 在这种情况下，我们将添加 **“PM5072-T-2ETH”**

五. 接下来，单击“添加 PLC”按钮将其添加到 Automation Builder 硬件树中。

六. 创建硬件树后，Automation Builder 将启动运动解决方案向导概述页。从这里，用户可以将轴添加到应用程序中。根据所选的 PLC 类型，用户无法将某些轴类型添加到项目中。



2.2.2 将轴添加到项目中

2.2.2.1 添加 PTO 轴

所有 AC500 eCo PLC 都通过板载 IO 通道支持最多 4 个 PTO 轴。

Motion axes cycle time : 4000 µs

Axis name	Axis type	HW Type	HW Info	Channel/Node	Status	Simulate	Generate
PTO_Axis	FiniteRotary	PTO	Onboard	PTO 0 (DO00-DO04)	Not generated	<input type="checkbox"/>	<input checked="" type="checkbox"/>
E530_EC_axis	FiniteRotary	EtherCAT	Onboard	1001	Not generated	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Hide generated code

Configured motion axes: 2/9

Revision 1 CPU

Mint sidebar

Add virtual axis

Add encoder axis

Add PTO axis

Add onboard EtherCAT axis

Add CM579-ETHCAT axis

要添加 PTO 轴，您可以使用上面的按钮。也可以通过单击“OnBoard_IO”对象并选择“添加对象”选项或使用“运动解决方案向导”概述页面中的“添加 PTO 轴”按钮来实现。

2.2.2.2 添加 EtherCAT 轴

从 AB2.7.0 开始，用户将有两种选择来实现 EtherCAT 网络;使用“AC500 和 CM579-ETH”解决方案，并可选择使用 AC500 eCo PM5072 和 PM5082 PLC（使用板载 ETH1 端口的 EtherCAT 主站）。

Motion axes cycle time : 4000 µs

Axis name	Axis type	HW Type	HW Info	Channel/Node	Status	Simulate	Generate
PTO_Axis	FiniteRotary	PTO	Onboard	PTO 0 (DO00-DO04)	Not generated	<input type="checkbox"/>	<input checked="" type="checkbox"/>
E530_EC_axis	FiniteRotary	EtherCAT	Onboard	1001	Not generated	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Hide generated code

Configured motion axes: 2/9

Revision 1 CPU

Mint sidebar

Add virtual axis

Add encoder axis

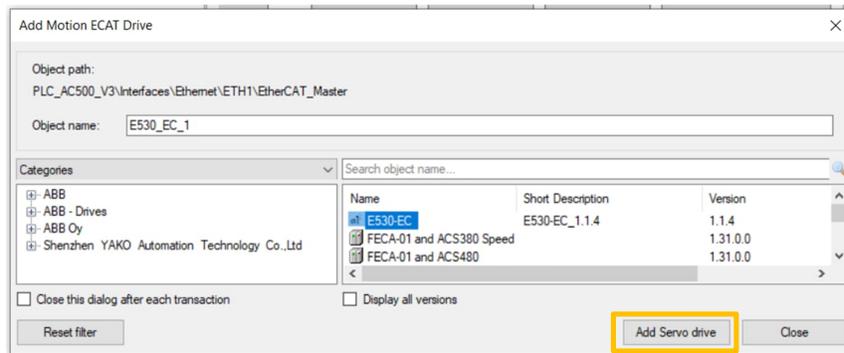
Add PTO axis

Add onboard EtherCAT axis

Add CM579-ETHCAT axis

要添加 ECAT 轴，您可以使用上面突出显示的按钮。也可以通过单击“EtherCAT_Master”对象并选择“添加对象”选项或使用“运动解决方案向导”概述页面中的“添加 PTO 轴”按钮来实现。

单击此按钮后，您将可以从设备存储库中选择驱动器类型。在这里，我将选择 e530-EC，然后单击“添加伺服驱动器”



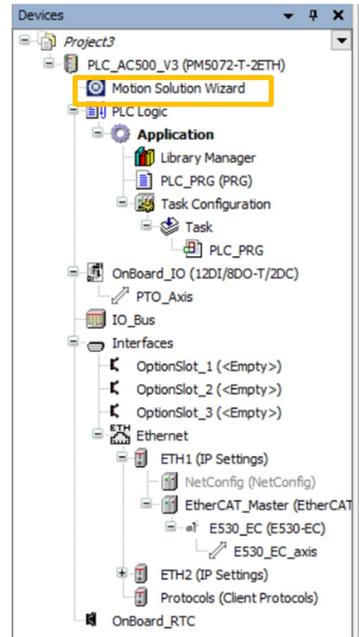
2.2.2.3 轴对象

Axis Objects 将作为列表显示在 Motion Solution Wizard 的菜单中，用户可以使用该列表跳转到其配置页面，并且它们也将显示在项目中它们所使用的硬件下。

嵌套在“OnBoard_IO”对象下方的是 PTO 轴对象。

同样嵌套在“EtherCAT 主站”对象下方的是 e530 EtherCAT 驱动器 Axis 对象。

在这里，用户可以根据应用程序要求更新设置。稍后下载应用程序时，向导将使用这些设置来定义此轴对象的操作和缩放。



2.2.3 配置 EtherCAT 主站

EtherCAT 主站可以通过 CM579-ETHCAT 耦合器在 AC500 CPU 中配置，也可以通过板载 ETH1 端口在 AC500 eCo Pro PLC 中配置。

一般来说，两种解决方案中的所有 EtherCAT 设置都是相似的，但在 CM579-ETHCAT 主配置器中，选项卡将比板载 EtherCAT 更多。根据使用的 EtherCAT 解决方案，用户还需要使用相关库进行诊断/SDO 处理等。

CM579-ETHCAT 系列

这些是特定于 CM579-ETHCAT 主站的设置，对板载 EtherCAT 无效。标签通常会“CM579_ETHCAT”，这可以由用户更改，但通常保留为默认值。如果是这样，名称将是 CM579_ETHCAT' (CM579-ETHCAT)。

Parameter	Type	Value	Description
Run on config fault	Enumeration of BYTE	Yes	Start PLC program even on configuration fault
Broken slave behaviour	Enumeration of DWORD	Leave all broken slaves down	Behaviour of broken slaves
Distributed clocks	Enumeration of DWORD	Active	Distributed clocks inactive or active
Bus Target State	Enumeration of BYTE	Operational, OP	Target state of the EtherCAT bus at application
Bus behavior	Enumeration of DWORD	Asynchronous (IEC bus cycle)	Sync mode 1 - minimum lag (1 bus cycle) between
Optimize I/O update	Enumeration of BYTE	On	Optimize I/O update

在大多数情况下，这些设置可以保留为默认值，但有时用户可能需要更改这些设置以满足应用程序要求。

- **在配置故障时运行**- 默认情况下，它设置为“是”，这意味着 CM579-ETHCAT 不会出错，如果从站丢失，PLC 程序不会停止，这是有益的，因为 PLC 仍然可以处理其他后续操作。如果用户想要缺少从站会阻止 PLC 运行的配置，则可以更改此配置。
- **分布式时钟**- 默认情况下，它设置为“活动”，如果您想在 EtherCAT 上使用同步运动，这是必须的。
- **优化 I/O 更新**- 默认情况下，它设置为“开”，这意味着连续的 I/O 合并到一个块中以优化性能。

ETHCAT_Master (EtherCAT-Master)

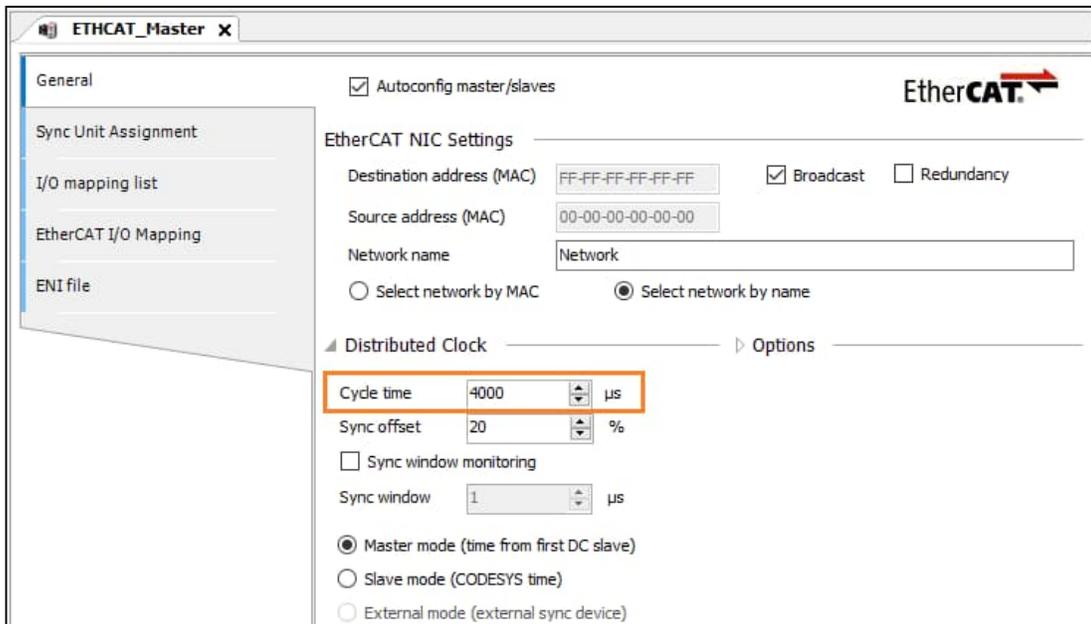
EtherCAT 主站的第二部分定义了定义 EtherCAT 操作行为的特定设置，从这里开始，CM579-ETHCAT 和板载 EtherCAT 的设置是相似的。



注意：板载 EtherCAT 只能在 PM5072 和 PM5082 ETH1 端口上配置。建议不要在 ETH1 上配置任何其他协议;EtherCAT 端口。

标签通常采用“[name] (ETHCAT-Master)”的格式，用户可以更改，但通常保留为默认值。如果是这样，则名称将为 **ETHCAT_Master** (EtherCAT-Master)。下面提到了最重要的设置：

- **自动配置主/从** - 默认情况下，这是激活的，这意味着参数会根据默认设置自动设置。除非用户非常熟悉 EtherCAT 网络的设置，否则建议使用此设置。
- **循环时间** - 默认 EtherCAT 循环时间为 4000 μ s，但根据应用要求，用户可以调整 EtherCAT 循环时间，如下所示。



请注意，EtherCAT 循环时间将直接影响 PLC 负载。如果您的 PLC 负载高于预期，请增加循环时间或升级 PLC 类型。

2.2.4 配置 EtherCAT 从站和轴

本节介绍如何配置 Comms Slave 模块的行为。从这里我们可以定义硬件的行为方式。重要的是要了解 EtherCAT 从站轴一旦添加，就会在项目树中分为两部分。下面将介绍这两个部分。

2.2.4.1 EtherCAT 从驱动器对象设置

标签通常采用“[驱动器名称] (驱动器类型)”的格式，用户可以更改，但通常保留为默认值。如果是这样，则名称可以 **MicroFlex_e190** (MicroFlex-e190)

在大多数情况下，这些设置可以保留为默认值，但有时用户可能需要更改这些设置以满足应用程序要求。有用的设置可能是启用专家设置以添加其他 PDO 映射、检查和设置映射或在联机后检查设备的状态。

2.2.4.2 PDO 映射

2.2.4.2.1 PDO 和启动参数 (SDO)

根据[运动轴]下的选择，>“控制类型”和“PDO”映射的映射，向导将从属轴对象更新“过程数据”选项卡，并为添加的每个对象分配自动生成的名称。



Object Name	Variable	Channel	Address	Type	Description
e530_ec	e530_ec_ControlWord	16#1600 Receive PDO Mapping 1 : Controlword	%QW2	UINT	Controlword
e530_ec	e530_ec_TargetPosition	16#1600 Receive PDO Mapping 1 : Target positie	%QD2	DINT	Target position
e530_ec	e530_ec_Touch_probe_function	16#1600 Receive PDO Mapping 1 : Touch probe f	%QW6	UINT	Touch probe function
e530_ec	e530_ec_Digital_outputs	16#1600 Receive PDO Mapping 1 : Digital output	%QD4	UDINT	Digital outputs
e530_ec	e530_ec_Error_code	16#1A00 Transmit PDO Mapping 1 : Error code	%IW2	UINT	Error code
e530_ec	e530_ec_StatusWord	16#1A00 Transmit PDO Mapping 1 : Statusword	%IW3	UINT	Statusword
e530_ec	e530_ec_ActualPosition	16#1A00 Transmit PDO Mapping 1 : Position actu	%ID2	DINT	Position actual value
e530_ec	e530_ec_Torque_actual_value	16#1A00 Transmit PDO Mapping 1 : Torque actu	%IW6	INT	Torque actual value
e530_ec	e530_ec_Following_error_actual_v...	16#1A00 Transmit PDO Mapping 1 : Following er	%ID4	DINT	Following error actual value
e530_ec	e530_ec_Touch_probe_status	16#1A00 Transmit PDO Mapping 1 : Touch probe	%IW10	UINT	Touch probe status
e530_ec	e530_ec_Touch_probe_pos1_pos...	16#1A00 Transmit PDO Mapping 1 : Touch probe	%ID6	DINT	Touch probe pos1 pos value
e530_ec	e530_ec_Touch_probe_pos2_pos...	16#1A00 Transmit PDO Mapping 1 : Touch probe	%ID7	DINT	Touch probe pos2 pos value
e530_ec	e530_ec_Digital_inputs	16#1A00 Transmit PDO Mapping 1 : Digital inputs	%ID8	UDINT	Digital inputs

与 PDO 映射一样，SDO 启动将根据控件类型和向导中选择的映射进行更新。

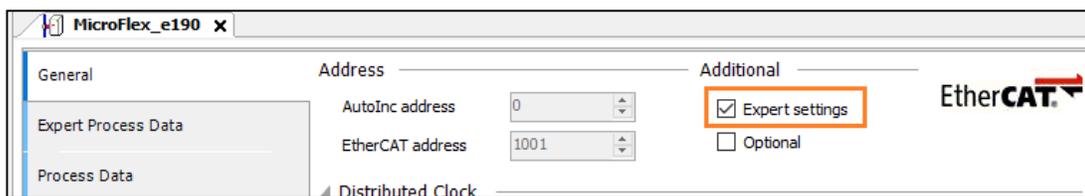
Line	Index:Subindex	Name	Value	Bit Length	Abort on Error	Jump to Line on Error	Next Line	Comment
1	16#6060:16#00	Modes of operation	8	8	<input type="checkbox"/>	<input type="checkbox"/>	0	Modes of operation



注意：根据应用要求，用户可以手动添加更多 PDO /启动参数。

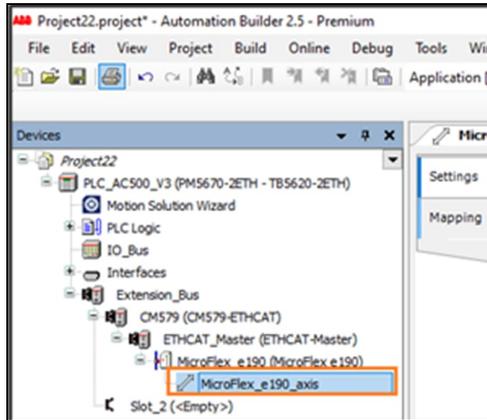
2.2.4.2.2 用户定义的 PDO 映射

如果用户想要添加额外的 PDO 映射对象（无法从[运动轴]>映射中选择），请先启用专家设置，然后用户将在常规选项卡下方获得一个名为“专家过程数据”的附加选项卡，在这里用户可以添加/编辑/删除映射。



2.2.5 轴设置

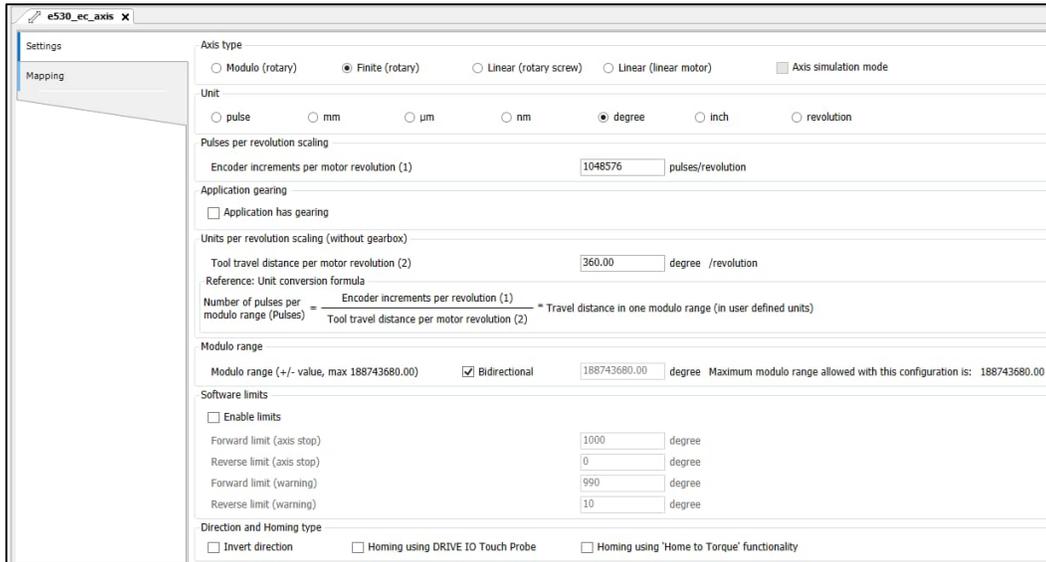
嵌套在 EtherCAT 从驱动器对象下方的是 Axis 对象，用户可以根据应用要求配置每个轴。



要打开它，请双击对象以打开配置页面。

2.2.5.1 所有类型通用的轴设置

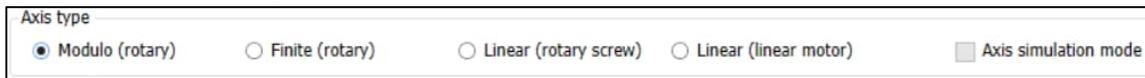
Axis 对象的主视图是“设置”选项卡，如果是 EtherCAT，轴也会显示“映射”选项卡。用户可以根据应用程序要求更新这两个页面。



稍后在“生成应用程序”过程中，向导将使用这些设置来定义此轴对象的操作和缩放。

轴类型

用户可以根据应用需求选择要配置的轴类型。



以下是用户可以从运动控制向导中配置的设置列表及其含义。

轴类型	细节	最大限制（计数）
模态（旋转）	向导中的默认设置。 通过选择模（旋转），您的轴将被配置为旋转轴，并且可以在下方配置所需的模范围。	2 ³¹ -1

有限（旋转）	您的轴将被配置为旋转轴，其中用户不可编辑模数范围，并根据“单位”选择、Inc_Per_R、U_Per_Rev_Nominator 和 U_Per_Rev_Denominator 设置进行计算。	2^30（单向） 2^39（双向）
线性（旋转螺杆）	当用户拥有具有线性运动（线性轴）的旋转电机时，需要对此进行配置。	2^31-1
线性（直线电机）	当用户轴是直线电机时，需要对此进行配置。	2^31-1
虚拟轴模式	此选项仅从此处读取，当用户配置轴但尚未拥有实际硬件时需要此选项。可以从运动解决方案向导概述页面中选择此选项。 有关虚拟轴配置，请参阅“添加和配置虚拟轴”一章。	那

单位

根据应用要求，用户可以在向导中选择所需的设备，向导会将后续参数更新为所选用户设备。从下图中，用户可以找到当前支持的单位格式。

Unit

pulse
 mm
 μm
 nm
 degree
 inch
 revolution

例如，当用户选择轴类型为模数（旋转）并将单位选择为度数时，向导会将后续参数更新为所选用户单位并填充默认值，例如：模范围 = 360 度（默认）。但是，请确保用户根据实际应用要求更新后续参数。

应用传动装置/每转单位数。

在这里，用户可以设置每电机转数的刀具行程距离，并定义与机械系统的齿轮关系。

默认情况下，“应用程序有齿轮”框处于未选中状态，这意味着应用程序没有齿轮，用户可以根据应用程序要求更新“每电机转数的刀具行程距离”：

Application gearing

Application has gearing

Units per revolution scaling (without gearbox)

Tool travel distance per motor revolution (2) degree /revolution

Reference: Unit conversion formula

$$\text{Number of pulses (pulse)} = \frac{\text{Encoder increments per revolution (1)}}{\text{Tool travel distance per motor revolution (2)}} * \text{Travel distance (in user defined units)}$$

如果应用程序有传动装置，用户可以选中“应用程序有传动装置”框，系统将提示他们提供传动装置详细信息。在生成应用程序期间，向导将相应地更新相同的内容。

Application gearing

Application has gearing

Units per revolution scaling (with gearbox)

Tool travel distance per Gearbox output side revolution (3) degree /revolution

Gearbox output turns: Tooling side (Numerator of reduction ratio) (4)

Gearbox input turns: Motor side (Denominator of reduction ratio) (5)

Unit conversion formula (modulo)

$$\text{Number of pulses (pulse)} = \frac{\text{Encoder increments per motor revolution (1)} * \text{Gear box output turns: Tooling side (4)}}{\text{Tool travel distance (3)} * \text{Gearbox input turns: Motor side (5)}} * \text{Travel distance (in user defined units)}$$

范围模块

用户可以在此处提供模范围。这是轴位置将折回零的值。仅当用户选择轴类型作为任何“旋转”轴时，此窗口才会处于活动状态。

Modulo range

Modulo range (0-value) degree

软限位

用户可以从向导本身配置一些常见的“软限位”。以下是用户可以从所选应用程序单元中的向导中配置的软限位列表。

默认情况下，向导中的软限位未启用，用户需要通过启用“软限位”复选框来启用相同的限制。

Software limits

Enable limits

Forward limit (axis stop) degree

Reverse limit (axis stop) degree

Forward limit (warning) degree

Reverse limit (warning) degree

方向和归位类型

反转方向

这需要为需要改变其真实方向与 PLC 程序内的轴关系的应用程序进行设置。默认情况下，该复选框将处于取消选中状态，并且方向将为正常。通过选中“反转方向”复选框，实际位置和参考位置都将反转，并且轴将向相反方向移动。

Direction and Homing type

Invert direction Homing using DRIVE IO Touch Probe

位置控制（循环同步模式）

在这里，用户可以配置与位置控制和监督相关的参数。每个参数的详细信息在系统技术章节“位置控制回路”和轴参数章节下的“监控”中得到了很好的解释 [ms-its:C:/Program Files \(x86\)/ABB/AutomationBuilder/Online Help/en/PLC-Automation-V3.chm::/d39017916e300679.htm](ms-its:C:/Program Files (x86)/ABB/AutomationBuilder/Online Help/en/PLC-Automation-V3.chm::/d39017916e300679.htm)。

Position control (cyclic sync mode)	
Control time	20 ms
Feed forward percentage (0-100%)	80 %
Following Error Fatal percentage (0-300%)	150 %
Delay time velocity check	100 ms
Position lag supervision	Activated ▾
Filter	Off ▾

- 关闭过滤器 - 不应用过滤。
- 速度过滤器 - 使用多个采样来平滑内核中计算的速度
- 位置过滤器 - 线性回归计算用于平滑内核读取的位置值。

动态限制

用户可以在此处更新最大限制。某些参数取决于驱动器设置，需要正确设置才能获得所需的结果。

用户可以将最大应用速度设置为所需值，以限制最大应用速度。

Dynamic limits	
Maximum application velocity	36000 pulses /sec
Maximum acceleration	10000 pulses /sec ²
Maximum deceleration	10000 pulses /sec ²
Maximum jerk	2000 pulses /sec ³

例：

如果用户使用的是 ABB 伺服驱动器，则 131072 编码器每转递增，最大速度为 6000 RPM

最大应用速度

= 最大应用速度（以 RPM 为单位）* 每转刀具行程距离 / 60 * 齿轮箱提名器/齿轮箱分母

= 6000 * 360 / 60 * 1 / 1 = 36000 度/秒



为了便于计算参数，用户可以使用 [C: \Users\Public\Documents\AutomationBuilder\Examples\PS5611-Motion\Documentation](C:\Users\Public\Documents\AutomationBuilder\Examples\PS5611-Motion\Documentation) 示例程序文件夹中的 excel“MC parameterization.xlsx AC500_V3_MotionControl_Startup 指南”。

基于驱动器的限制。

在这里，用户可以定义限制，这些限制将决定驱动器的预期行为。

Drive based limits	
Maximum speed (user defined)	6000 rpm
Maximum positive torque	300 %
Maximum negative torque	-300 %

建议在驱动器和 PLC 参数处保持相同的最大速度。

结果（计算结果）

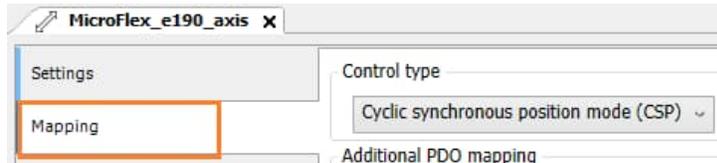
根据提供的输入，向导将计算结果，并可在配置页面末尾立即查看。

Results (calculated)		
Position resolution	364.0889	pulses/ degree
Maximum possible velocity	36000	degree /sec
Maximum allowed following error	3600	degree

2.2.5.2 特定于 EtherCAT 轴的轴对象设置

映射

如果用户要设置 CSP 以外的控制类型（默认）和非默认映射（控制字、设置位置、状态字、实际位置），则可以选择“映射和控制”



制类型”选项卡。可以在以下轴对象下找到它：

控件类型

默认情况下，为循环同步位置模式（CSP）选择向导。用户可以根据应用程序要求进行更改。支持的控制模式有：

- 循环同步位置模式（CSP）。
- 循环同步速度模式（CSV）。
- 用于负载控制（CSVL）的循环同步速度模式。
- 循环同步转矩模式（限定 CST）。



注意：CSVL 是 ABB 特有的模式，用于实现负载控制/分析。通过使用此模式，用户可以使用基于“PLCOpen Motion Part 6 – Fluid Power Extensions”实现的运动控制负载库。

其他 PDO 映射

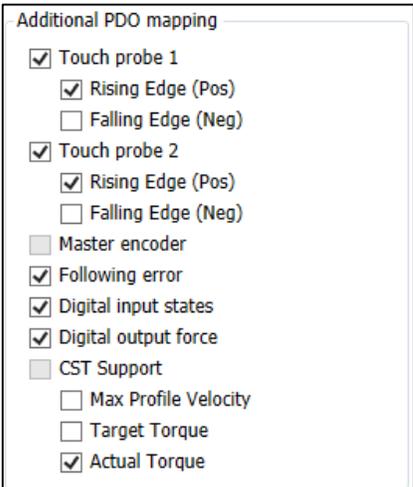
如果应用程序需要其他 PDO 映射，则向导可帮助用户添加最常用的 PDO 映射，只需在此处选择它们即可。

根据所选的控制类型，会自动生成一些必需的 PDO 映射，并且从向导中的其他映射区域中，用户可以找到最常见的 PDO 映射，并且用户可以根据应用程序要求添加相同的映射。

请注意，用户可以通过从设备常规配置页面（如前所述）启用专家设置来手动添加此处未列出的其他 PDO 映射。

SDO 启动参数映射

默认情况下，选择两个 SDO 启动参数，建议不要更改这些参数，除非用户对 DS 402 控制模式有专家级知识或打算进行非标准启动编码，因为这会改变轴在启动时的预期操作。



SDO startup parameter mapping

- Give EtherCAT control
- Operating mode
- Torque limits

用户可以选择扭矩限制，从“设置”页面设置的扭矩值将写入相应的从驱动器启动参数列表。

Drive based limits

Maximum speed (user defined) rpm

Maximum positive torque %

Maximum negative torque %

完成这些设置并执行生成代码后，您可以看到这些设置更改了驱动器的 EtherCAT 从站配置，如下图所示：

Line	Index:Subindex	Name	Value
1	16#5002:16#00	Give EtherCAT Control	1
2	16#6060:16#00	ModesOfOperation = 8 (Cyclic Synchronous Position Mode)	8
3	16#5023:16#00	Maximum Torque Limit	3000
4	16#5022:16#00	Minimum Torque Limit	-3000

设置 - 使用驱动器 IO 触摸探头归位。

选择此选项将允许轴使用 EtherCAT 连接到基于驱动器的触摸探头。

Direction and Homing type

Invert direction Homing using DRIVE IO Touch Probe Homing using 'Home to Torque' functionality

默认情况下，选择此选项也会为用户预先选择 PDO 映射，但以后可以根据应用程序需要进行更改。

Additional PDO mapping

- Touch probe 1
 - Rising Edge (Pos)
 - Falling Edge (Neg)
- Touch probe 2
 - Rising Edge (Pos)
 - Falling Edge (Neg)

设置 - 使用“原点扭矩”功能归位

选择此选项将允许轴回到终点停止或扭矩设定值。

Direction and Homing type

Invert direction Homing using DRIVE IO Touch Probe Homing using 'Home to Torque' functionality

这还需要将实际扭矩参数和设定值参数映射到轴参考。

- CST Support
 - Max Profile Velocity
 - Target Torque
 - Actual Torque

2.2.5.3 轴对象设置与 PTO 轴特别相关

与应用程序和特定轴相关的所有设置都将在此处完成，并且需要针对每个轴仔细更新。根据此处提供的输入，向导将编译并生成代码。

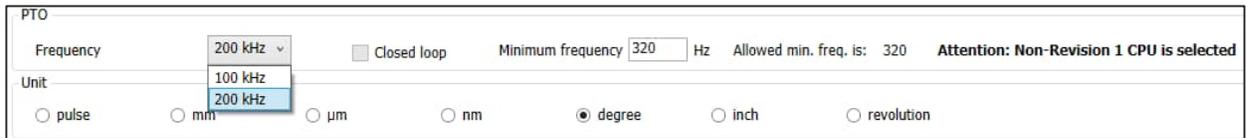
PTO (硬件设置)

使用版本 1 CPU 或更高版本，默认支持 4*200kHz PTO 轴，轴配置页面无需特殊设置。



PTO
Frequency 200 kHz Closed loop Minimum frequency 320 Hz Allowed min. freq. is: 320

如果使用修订版 0 CPU，用户可以使用如下所示的下拉列表将 PTO 轴频率设置为 100kHz 或 200kHz。根据 PTO 轴频率选择，板载 IO 配置将更新。



PTO
Frequency 200 kHz Closed loop Minimum frequency 320 Hz Allowed min. freq. is: 320 Attention: Non-Revision 1 CPU is selected
Unit
 pulse mm μm nm degree inch revolution

闭环

默认情况下，PTO 轴是开环的，但是用户也可以将最多两个 PTO 轴配置为闭环。要配置闭环轴，用户需要配置编码器轴 -> 编码器源 -> 编码器通道 -> 编码器用途作为 PTO 轴，该轴需要是闭环轴。此复选框将显示该设置的结果！

每转脉冲数缩放

当轴为开环时，用户可以使用每转步数更新“每转脉冲缩放”。



Pulses per revolution scaling
Steps per revolution (1) 2000 steps / revolution



注意：请注意，当 PTO 轴开环时，用户必须通过考虑 PTO 频率限制来设置每转步数和最大转速。

例如，对于 100KHz PTO 轴，如果每转步数为 2000，则该轴可以支持的最大 RPM 为 3000 RPM。 (= 2000 * 3000/60 = 100KHz)

当轴闭环时，用户可以使用电机转数的实际编码器增量来更新“每转脉冲数缩放”。



Pulses per revolution scaling
Encoder increments per motor revolution (1) 2000 pulses/revolution

基于驱动器的限制。

最大参考值是驱动器达到最大 RPM 的最大频率。在某些驱动器中，可能需要修改计算出的最大参考值，用户可以选择“修改”框并更改该值。请注意，根据配置，我们的硬件限制为 100kHz 或 200kHz，用户必须确保没有超过此限制。

建议使用最大速度以保持驱动器和 PLC 参数的最大速度相同。



Drive based limits
Maximum reference value 200000 Modify
Maximum speed (user defined) 6000 rpm Maximum speed allowed with this configuration is: 6000

目前，向导中的应用扭矩限制对 PTO 轴无效，这将被忽略。

	<p>注意：</p> <p>当 PTO 轴为开环时，用户必须通过考虑 PTO 频率限制来设置每转步数和最大转速。例如，对于 100Khz PTO 轴，如果每转步数为 2000，则该轴可以支持的最大 RPM 为 3000 RPM。（= 2000 * 3000/60 = 100Khz）</p> <p>闭环时，请设置提供最大 PTO 轴频率时轴达到的最大 RPM。例如，如果轴配置为 200Khz 且闭环，则设置达到 200Khz 频率时轴的最大 RPM。</p>
---	---

2.2.6 重要常规 PLC 设置

与标准 AC500 项目相比，当用户使用运动解决方案向导时，在代码生成过程中会更改一些默认设置。如果轴配置发生更改或添加了新轴，则每次都会覆盖这些设置。以下是由运动解决方案向导更新的设置：

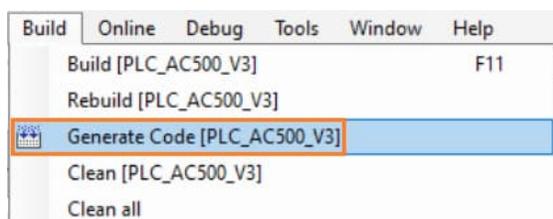
设置	项目类型	
	EtherCAT	仅限 PTO
IO 总线 - 在 congif 故障时运行	是	是
PLC > PLC 设置 > 始终更新变量	“已禁用”	“已禁用”
PLC > PLC 设置 > 总线循环选项	例如，“任务”——必须选择任何真正的慢速任务，并且不应将其设置为“未指定”，因为这可能会导致循环操作的负面影响。	
CPU-参数 > 检查电池	关闭	不适用
CM579-ETHERCAT > 在 congif 故障下运行	是	不适用
CM579-ETHERCAT > 总线行为	同步模式 1	不适用

建议用户根据实际应用要求手动更改上述设置。要手动更新这些设置，用户需要在完成所有轴更改后再更改设置，然后生成代码。

2.2.7 生成运动配置

完成所有配置后，用户可以生成应用程序（如编译），然后根据向导中提供的设置和参数自动更新其设置，PDO 和 SDO 映射以及 Motion 任务配置。

为此，请转到“生成”菜单，然后单击“生成代码”



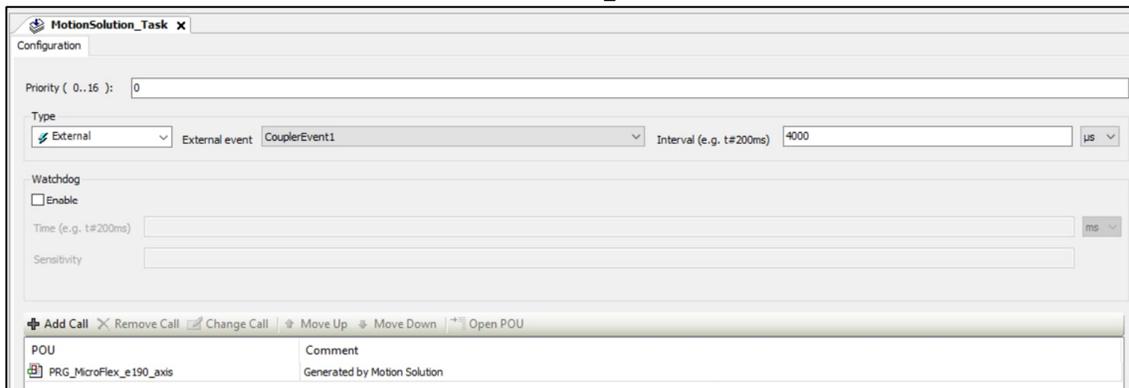
几秒钟后，Automation Builder 将生成轴配置。成功完成后，Automation Builder 将在消息窗口中显示消息“Motion Solution Generation successful”。这可能需要一些时间，具体取决于在项目中配置的轴数和 PDO 映射选择。

2.2.8 任务配置

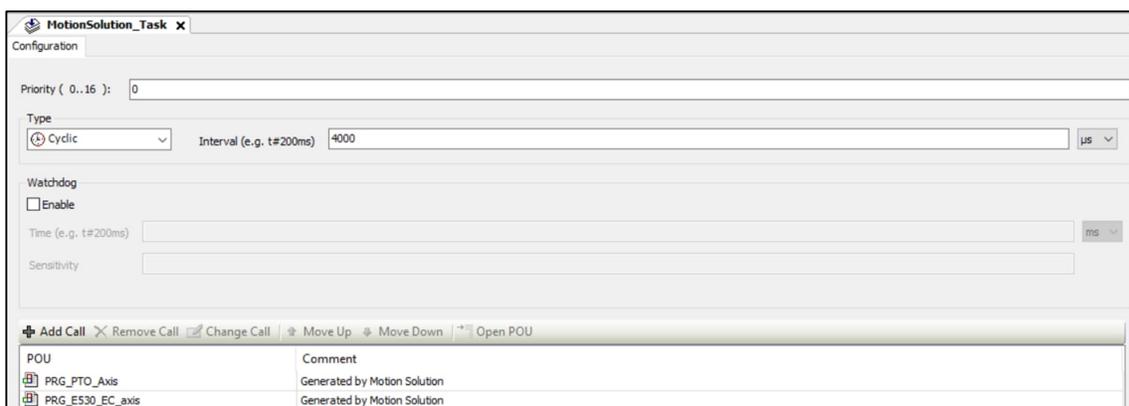
2.2.8.1 使用运动向导执行的任务

向导将自动创建“快速任务”和“慢速任务”，以帮助指导用户正确分发其代码。内核功能块和 EtherCAT（或其他硬件）的轴 IO 数据传输应该在更快的任务中处理。此任务应尽可能短且实时（如果是 EtherCAT）或尽可能快（如果是 PTO），以实现最佳运动控制性能。

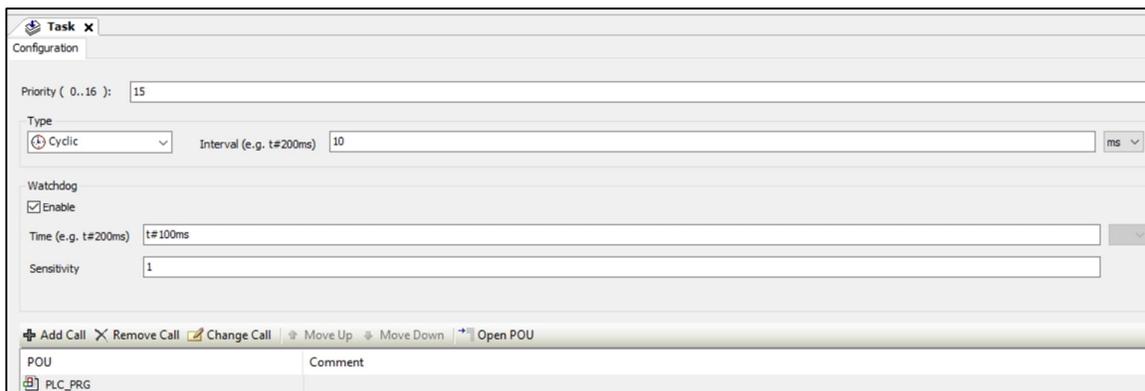
下图显示了 CM579- EtherCAT “快速任务” - MotionSolution_Task 的示例:



下图显示了板载 EtherCAT 的“快速任务”MotionSolution_Task 示例:



循环任务如下图所示:

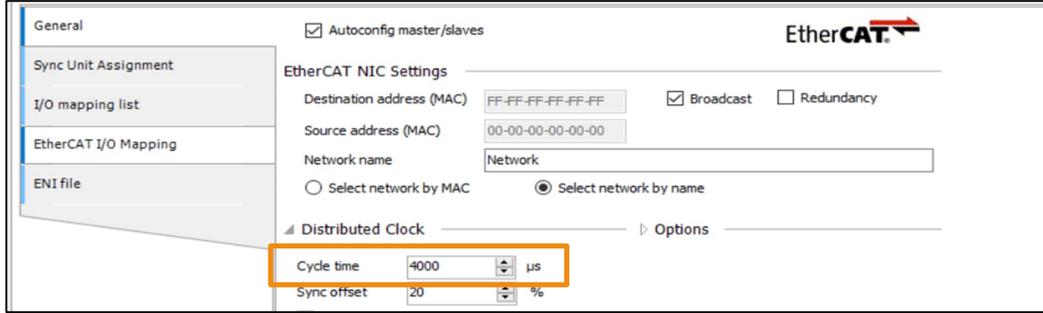


2.2.8.2 MotionSolution_Task

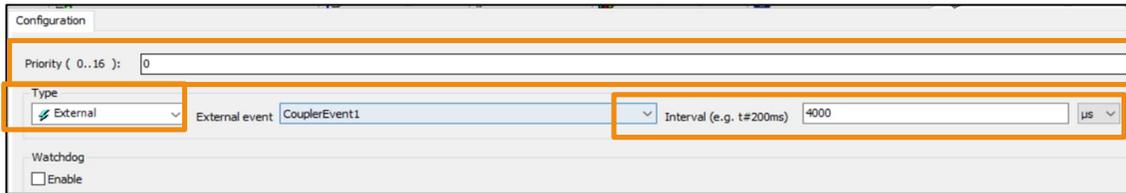
运动解决方案向导将自动创建一个名为“MotionSolution_Task”的快速任务。这将调用所有轴操作代码，并将根据配置进行配置。

2.2.8.2.1 EtherCAT 应用中的 MotionSolution_Task

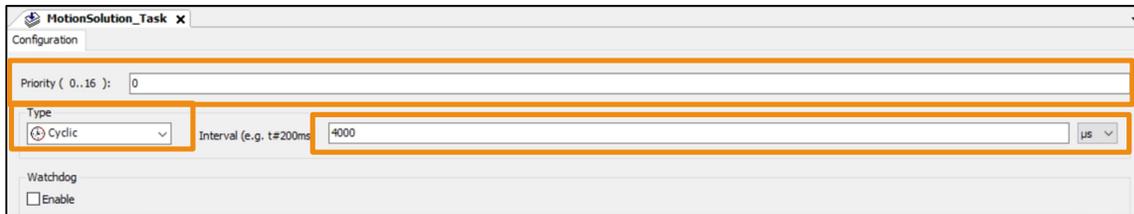
在同步 EtherCAT 应用的情况下，循环时间的设置在 EtherCAT 主站中定义；



一旦生成解决方案，它将定义一个任务，该任务的类型为外部，周期时间与 EtherCAT 主站中配置的时间相匹配，优先级设置为“0”。

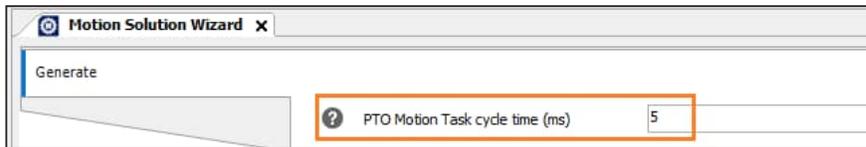


对于板载 EtherCAT，任务类型为循环，周期时间与 EtherCAT 主站中配置的时间相匹配，优先级设置为“0”。

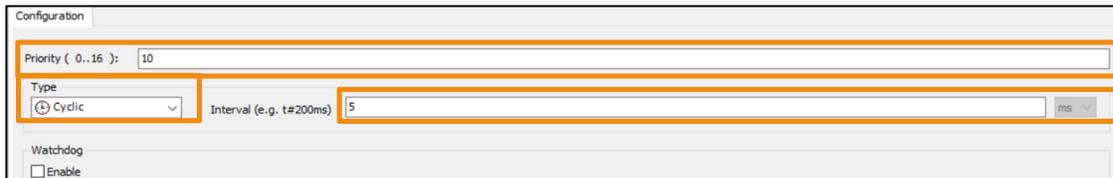


2.2.8.2.2 仅 PTO 应用中的 MotionSolution_Task

对于仅 PTO 应用，请在运动解决方案向导中设置循环时间，



生成解决方案后，这将定义一个任务，该任务的类型为“循环”，周期时间与解决方案向导中配置的时间相匹配，并且优先级设置为“10”



2.2.8.2.3 必须在运动解决方案任务中调用功能块

如前所述，“MotionSolution_task”是 CPU 中速度最快、优先级最高的任务，因此当然也是最耗费资源的任务。用户可以在此“MotionSolution_task”中调用运动应用程序的某些部分，但是此任务中过多的代码或结构不佳的代码可能会导致 CPU 过载。

尽管有这些限制，但某些功能块必须与内核功能块（MotionSolution_Task）相同的任务中调用。下面列出了这些：

- MC_CombineAxes

- MCA_MoveByExternalReference
- MCA_MoveByExtRefRelative
- MCA_DigitalCamSwitch
- MCA_GetTappetValue
- ECAT_CiA402 Touchprobe_App
- ECAT_HomingOnTouchprobe
- ECAT_CiA402ParameterHoming



注意：对于板载 EtherCAT 运动轴项目，建议在一个任务中调用所有 EtherCAT FB，对于耦合器 EtherCAT 来说，这无关紧要。

2.2.8.3 循环任务和创建的任务“任务”

默认情况下，向导还将创建一个名为“Task”的 10 毫秒循环任务，用户可以使用该任务调用非关键运动功能块：

也可以添加其他任务，但为了避免复杂化，最好不要添加太多任务，并避免冲突的优先级（例如使所有任务都相同）或干预时间，例如 10 毫秒、20 毫秒、40 毫秒，其中每 x 毫秒将调用几个任务。

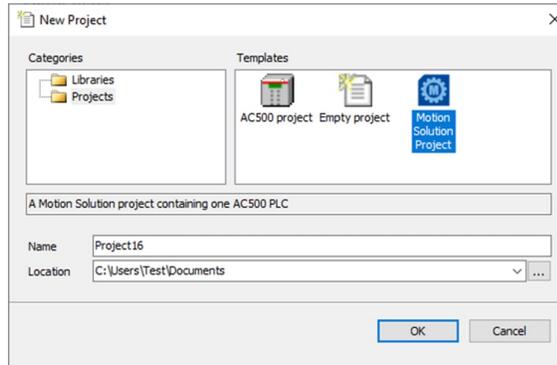
2.3 编写应用程序

到目前为止，我们遵循了第 3.2 节，我们已经完成了选择 PLC、更新设备文件、设置项目以及了解影响行为的重要设置的步骤。在本节中，我们将介绍项目的最后部分，编程和下载。

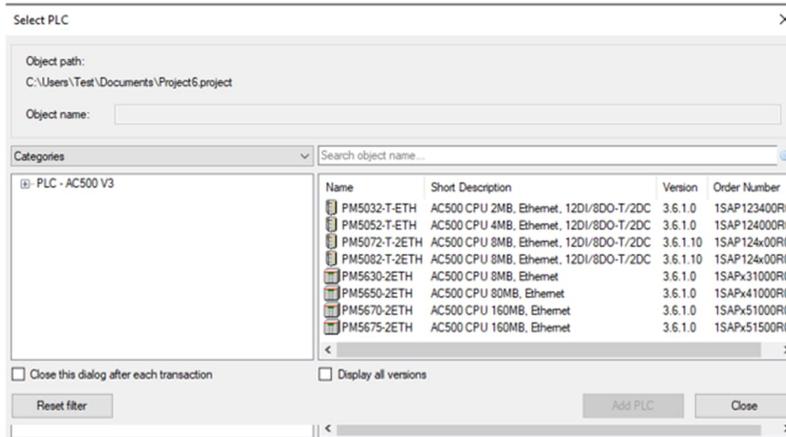
2.3.1 设置编程介绍项目

现在我们应该了解设置项目的步骤，我将简要介绍这些步骤，以与我们将在程序中引用的设置保持一致。

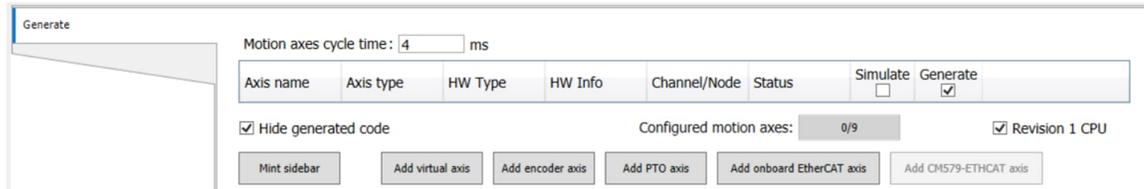
- 一. 创建程序的第一步是打开 Automation Builder 并选择“新建项目”。
- 二. 然后选择“运动解决方案项目”图标，如下所示。单击“确定”按钮，将在指定位置创建一个具有指定名称的新项目。



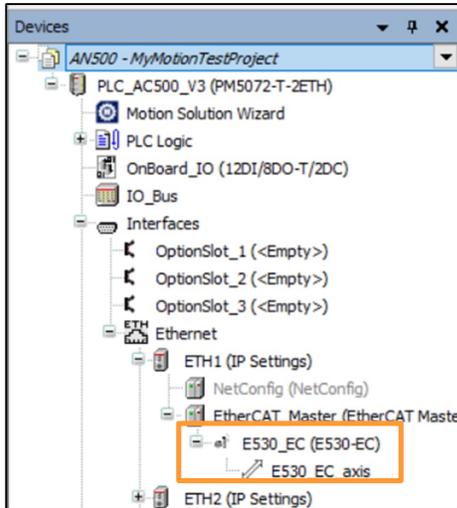
- 三. 添加 PLC - Automation Builder 现在将弹出一个“选择 PLC”窗口，用户可以从这里选择一个 V3 PLC。
 注意：有关正确的选择，请参阅以下部分：[选择正确的 AC500 PLC 类型](#)



- 四. 在这种情况下，我们将添加“PM5072-T-2ETH”
 五. 接下来，单击“添加 PLC”按钮将其添加到 Automation Builder 设备树中。
 六. 然后，在创建设备树后，Automation Builder 将启动运动解决方案向导概述页面。从这里，用户可以将轴添加到应用程序中。根据所选的 PLC 类型，用户无法将某些轴类型添加到项目中：



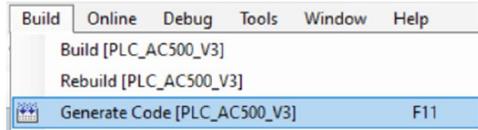
- 七. 现在，在这种情况下，我们可以通过选择“添加板载 EtherCAT 轴”并从“添加运动 ECAT 驱动器”弹出窗口中选择“e530-EC”来添加一个 EtherCAT 轴，然后将在 ETH1 下创建 am TherCAT_Master 并在其下方添加这个新轴：



八. 在此阶段，用户可以通过单击“E530_EC_axis”来更改轴设置，但现在我们将这些设置保留为默认设置。关键设置如下：

- 控制类型：CSP
- 映射：CWord、Sword、Apos、TPos、TP1Pos、TP2Pos
- 轴类型：有限
- 缩放比例：度数

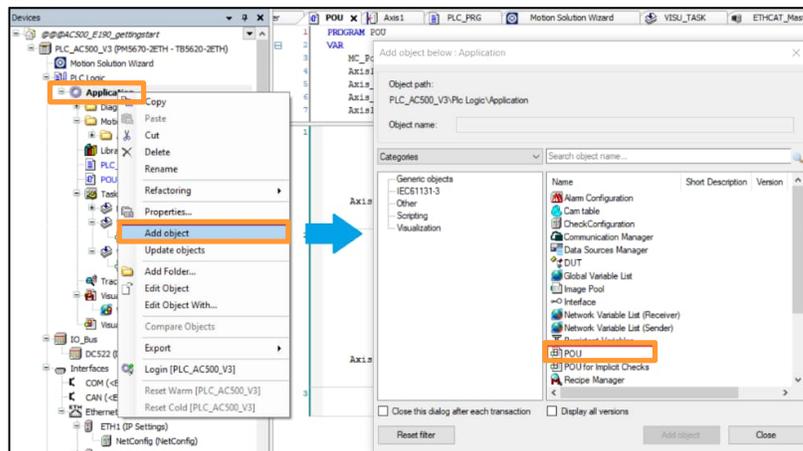
九. 在此之后，我们可以生成代码来构建 axis_ref 和其他关键设置，以便在程序中使用：



注意：如果此时存在任何构建错误，用户必须重新执行，然后才能继续操作。

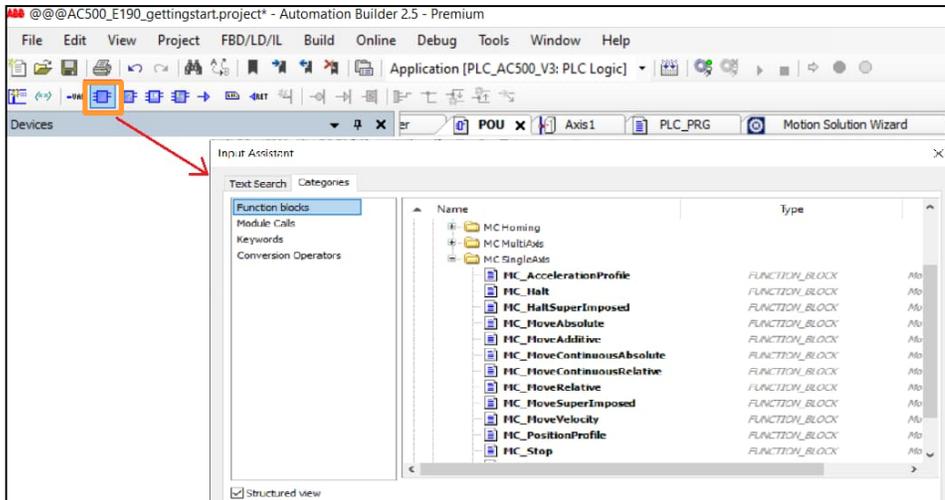
2.3.2 程序编写

十. 接下来，我们可以添加 POU（程序组织单元），其中包含我们的 PLCOpen Motion 程序。为此，请右键单击“应用程序”>“> POU 添加对象”

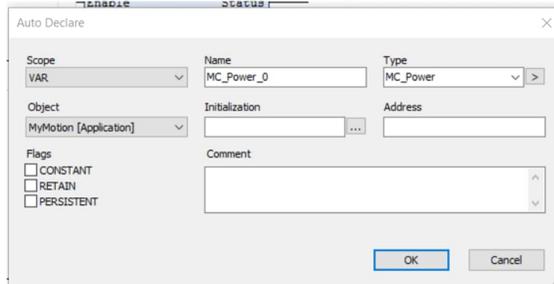


十一. 接下来，您可以编辑名称，我们可以向上显示“MyMotion”，作为语言，我们可以选择“FBD”，然后单击“添加”将 POU 添加到程序中。

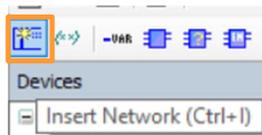
十二. 现在我们可以开始插入功能块了。为此，请单击此图标将 PLC OPEN 模块插入 POU。



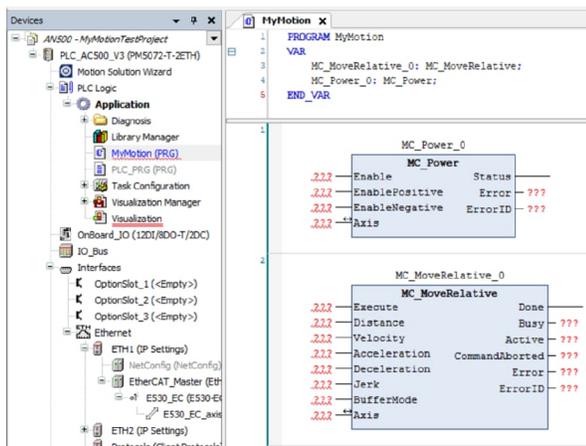
- 十三. 然后，您将看到一个弹出窗口，您可以在其中手动找到库，也可以通过“文本搜索”快速找到所需的 MC 块。
- 十四. 首先，我们可以选择功能块 MC_Power 然后单击“确定”来接受它
- 十五. 默认情况下，实例名称将在结构“[FB_name] [运行编号]”中给出，我们只需单击“确定”即可接受



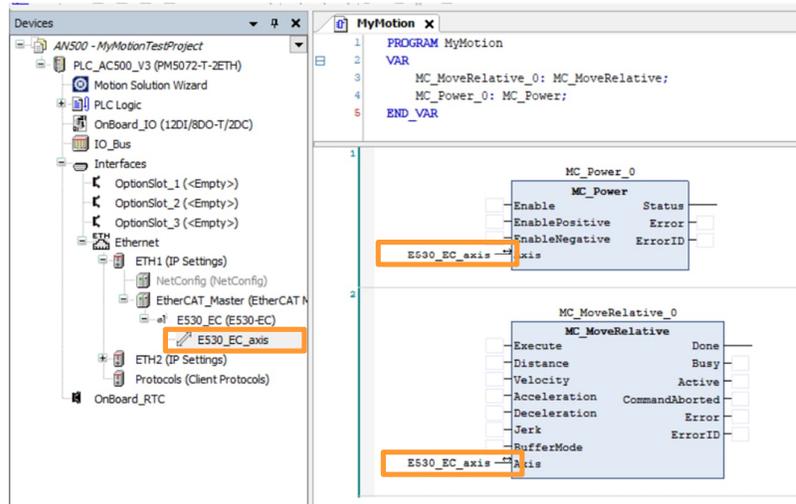
- 十六. 在此之后，我们可以使用此图标在下面插入一个网络：



- 十七. 然后我们可以按照前面的 steps 插入 MC_MoveRelative 的实例。完成这些步骤后，我们应该看到以下内容：

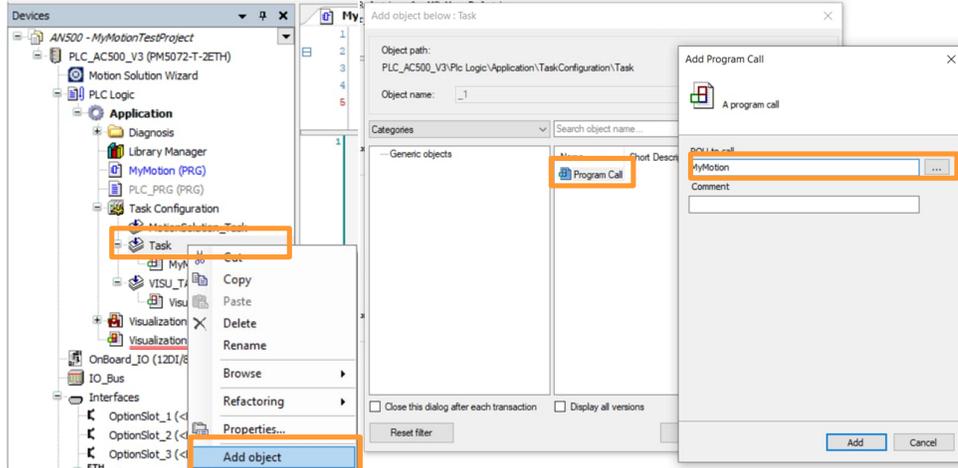


- 十八. 每个功能块接口的唯一必需变量是“轴”接口。Axis 接口的变量名称必须与树中的 Axis 名称相同。

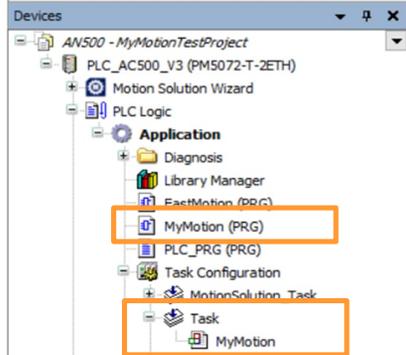


注意：所有其他输入都可以硬编码、附加到变量或强制在线，但在这种情况下，我们将使用可视化来填充并使用这些功能块。

十九. 现在，我们已经可以使用该程序了，我们必须通过从 Task 调用程序来确保对其进行处理。如前所述，为了节省 CPU 过载，建议考虑代码的正确分发。在这种情况下，我们将应用程序的代码放在“常规”任务“任务”中，该任务的周期时间为 10 毫秒，任务优先级低于“MotionSolution_Task”。为此，请右键单击“任务”>“添加对象”，然后选择“程序调用”并输入 POU 的名称“MyMotion”：



二十. 之后，您将在任务下看到 POU 调用

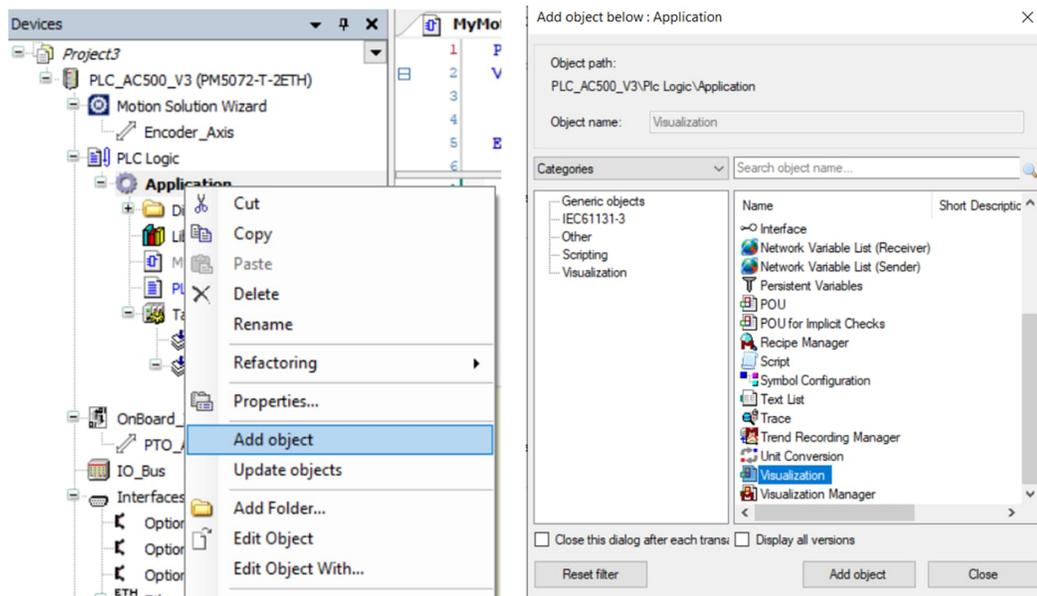


二十一. 该程序现在可以下载了，但有一些工具可以使测试更容易。

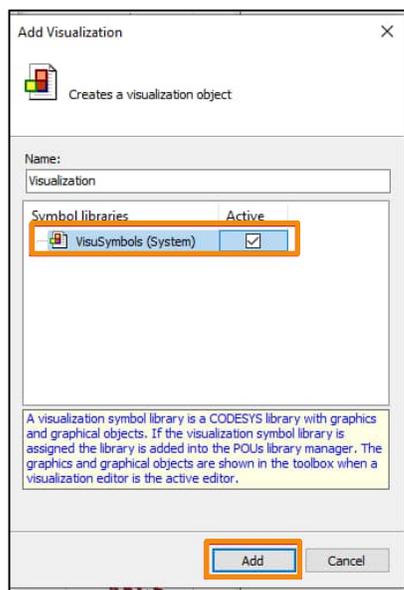
2.三.三 可视化（可选）

可视化可用于创建快速图形界面来测试程序。运动控制库有一系列预制接口可供使用。

- 一. 第一步是添加 Visualisation，右键单击“应用程序”，然后单击“添加对象”和“可视化”，然后单击“添加对象”和“添加”：

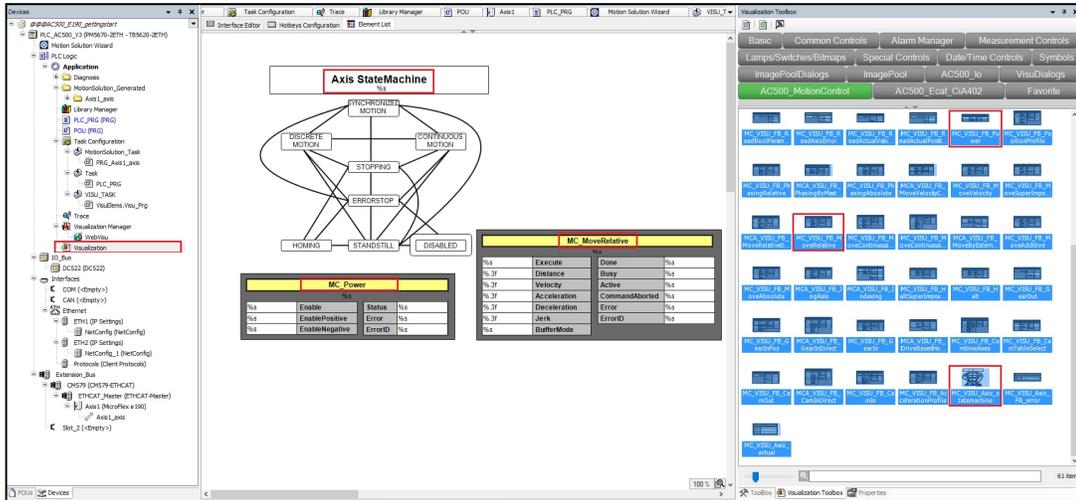


- 二. 然后选择“活动”VisuSymbols，然后选择“添加”



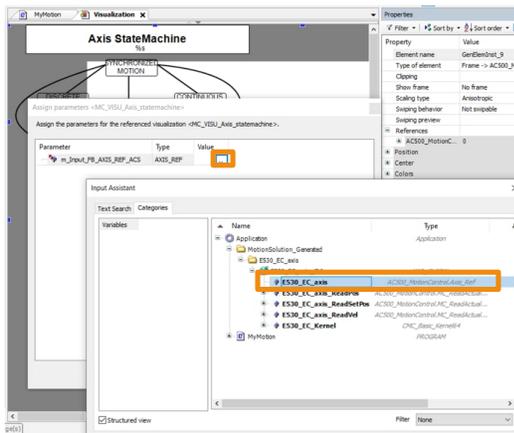
- 三. 接下来，打开“可视化工具箱”，然后从“元素”列表中拖放。

- 四. 在这里，我们选择 *MC_VISU_FB_POWER*、*MC_VISU_FB_MoveRelative* 和 *MC_VISU_Axis_statemachine*：

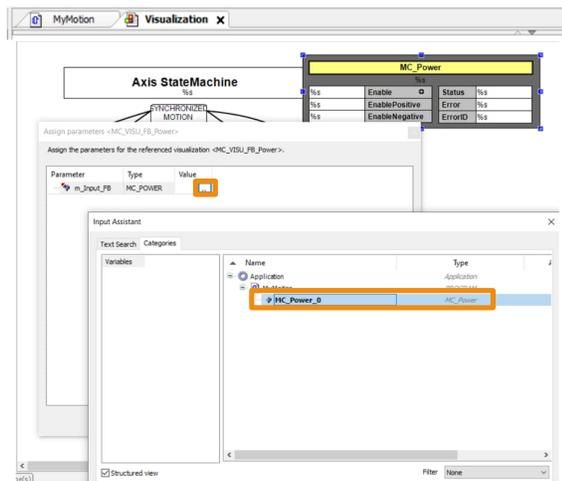


五. 将元素向下拖动到页面中后, 将弹出窗口, 以允许将可视化与功能块的实例相关联。

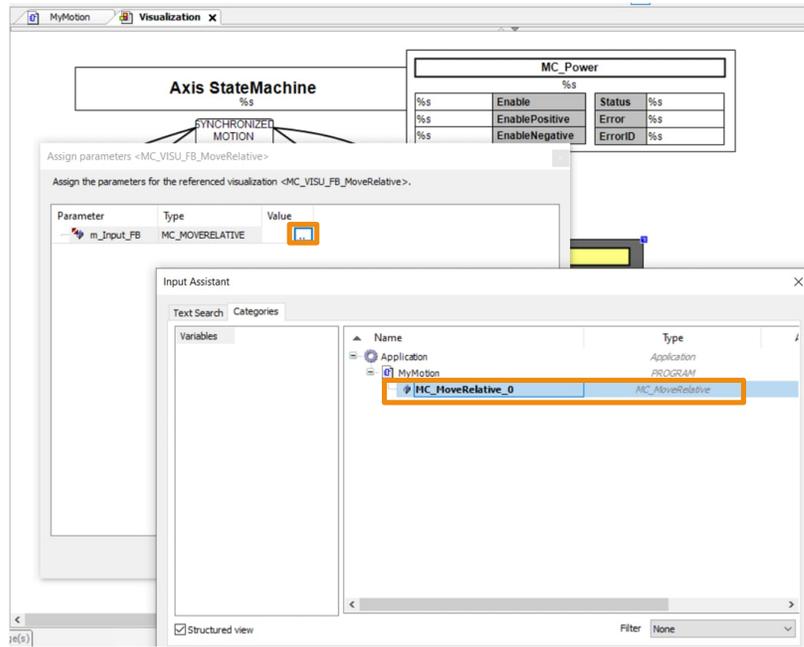
六. 一个接一个地, 单击“值”项, 然后选择相应的轴, 在此应用程序中 Axis_Ref Axis StateMachine, 然后单击“确定”



七. 指定相应的 POU, MC_Power_0 MC_VISU_FB_Power 然后单击“确定”, 如下所示



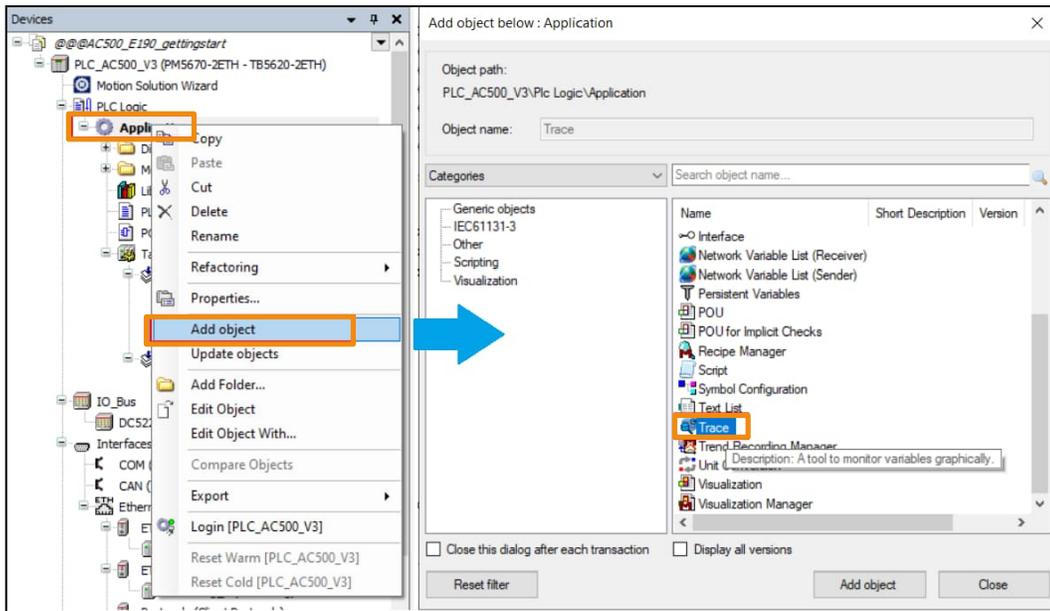
八. 指定相应的 POU, MC_MoveRelative_0 MC_VISU_FB_MoveRelative, 方法如下:



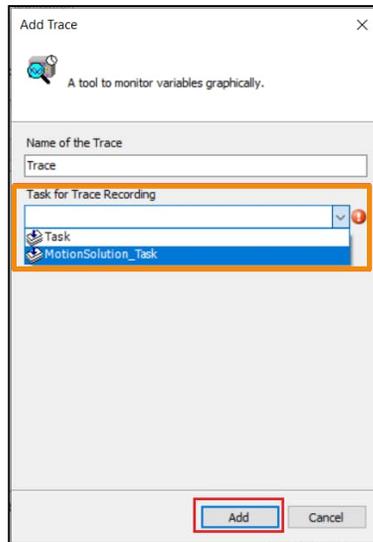
2.三四 跟踪（可选）

在运动应用程序中，跟踪函数通常用于在调试期间向用户展示运动状态。以下步骤说明如何向项目添加跟踪函数。

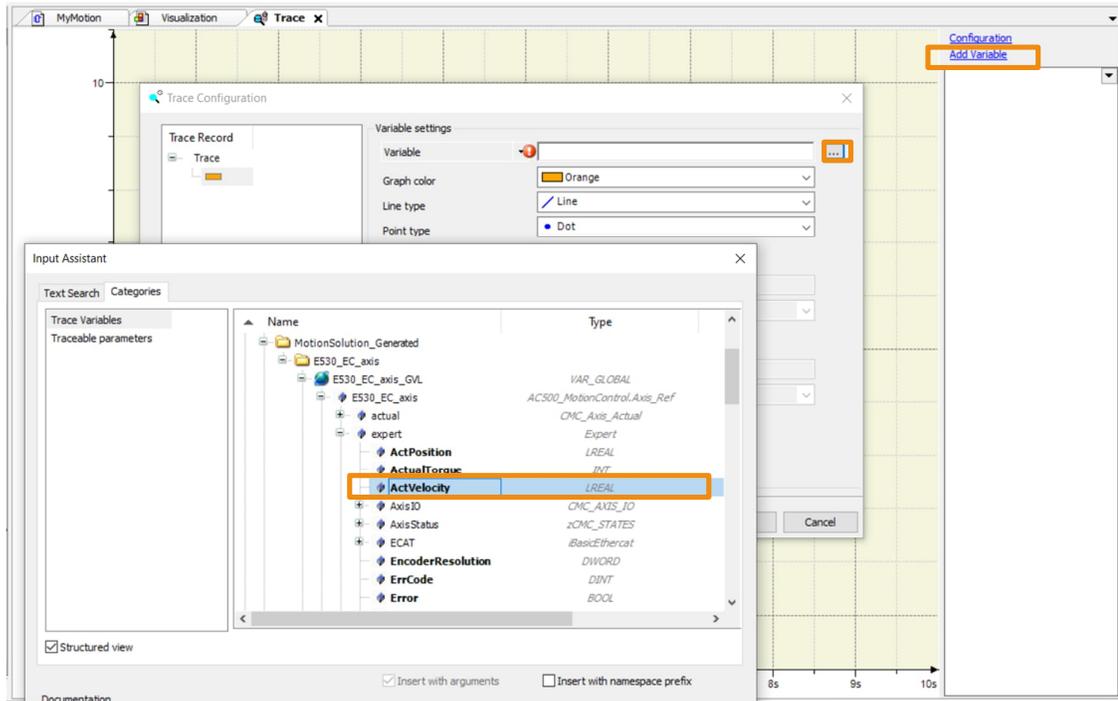
- 一. 要再次向项目添加跟踪，请右键单击“应用程序”>“添加对象”，然后选择“跟踪”，然后选择“添加对象”。



- 二. 下一步是选择用户要运行跟踪的任务。根据用户想要的响应（与 POU 调用方式相同），他们可以选择 Motion 任务或常规任务。运动任务速度较快，但会增加更多 CPU 负载，常规任务较慢，但对 CPU 负载的影响较小。



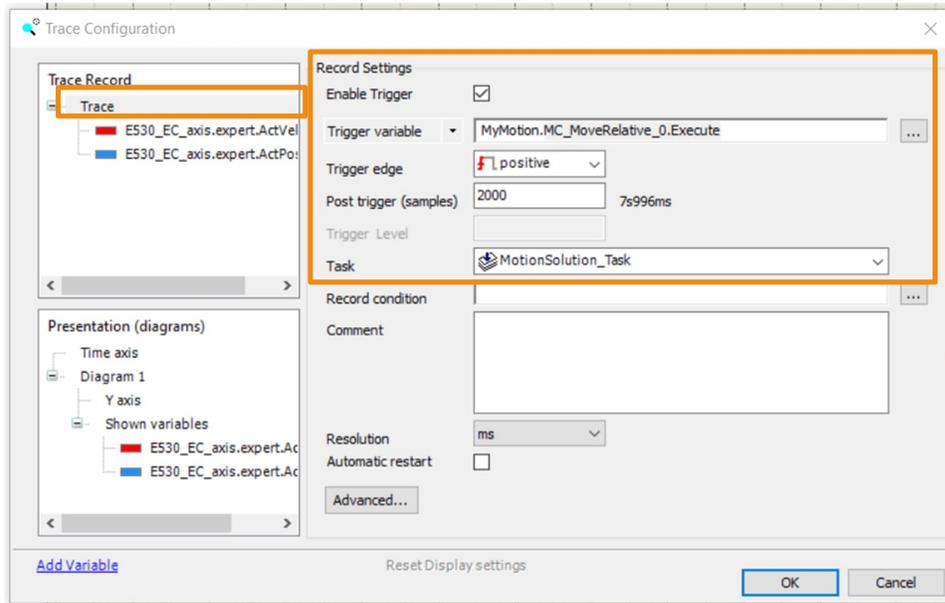
三. 添加跟踪后，用户可以开始将他们想要的任何变量映射到他们的项目。



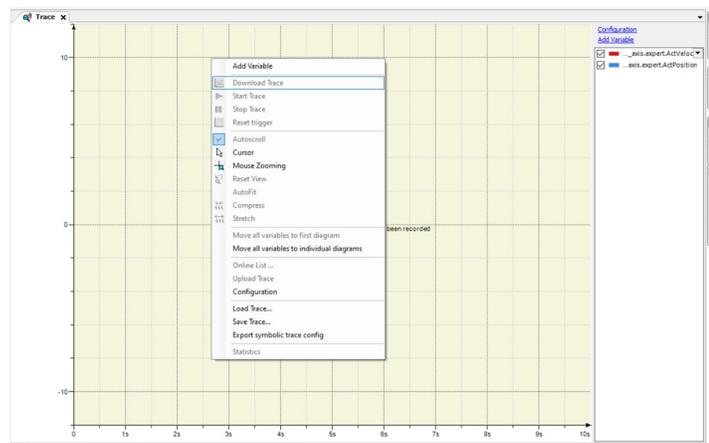
为了便于使用，可以从 Axis_Ref 结构的 [axis_ref].expert 部分获得一些实际信号。在这里，我们将使用：

- E530_EC_axis.expert.ActVelocity
- E530_EC_axis.expert.ActPosition

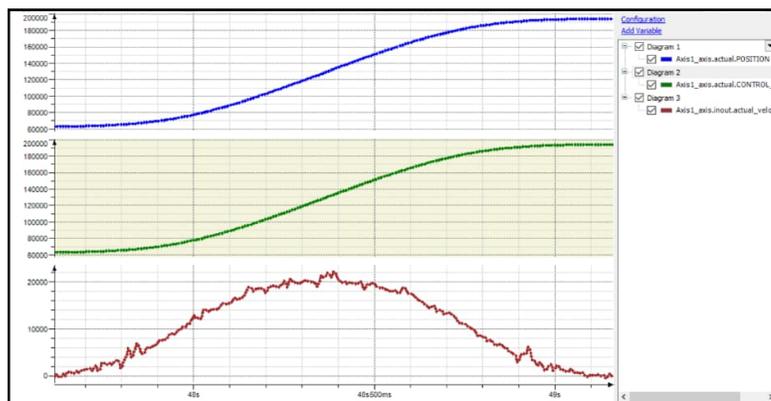
四. 它还可以设置触发条件，这意味着一旦下载，跟踪就不会触发，直到满足条件。在这种情况下，这将处于 MC_MoveRelative_0 的上升边缘。执行。



五. 现在，跟踪已准备好下载。要下载它，它必须是在线的，所以这个步骤将在后面完成。



六. 下载后，我们可以看到所有映射变量的范围跟踪：



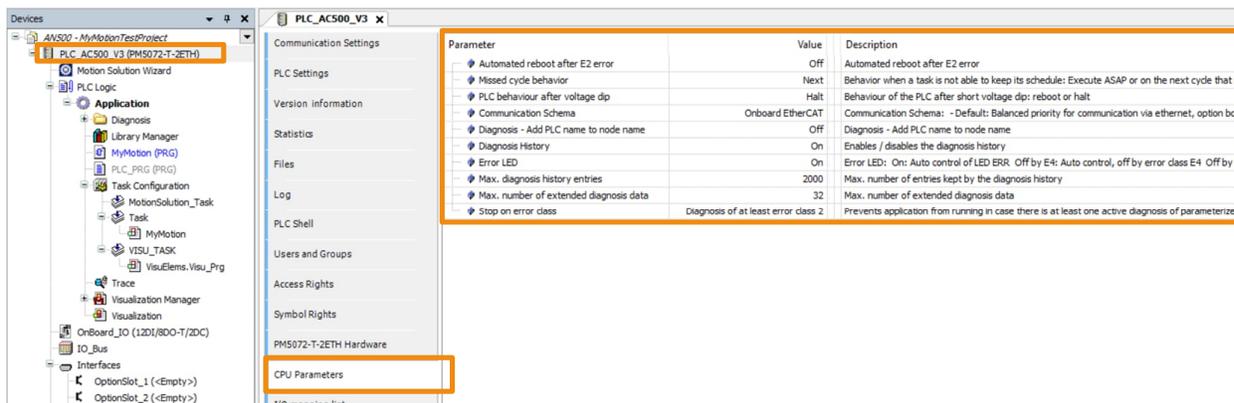
2.3.5 检查 CPU 参数。

在配置 Automation Builder 项目时，应考虑 CPU 参数设置，至少对于默认值不合适的苛刻情况。请根据您的硬件设置和系统要求检查并更新参数设置。访问这些设置。

- 一. 双击“PLC_AC500_V3”。

二. 在编辑器视图中将打开一个选项卡。

三. 选择“CPU-参数参数”。



在大多数情况下，这些可以保留为默认值。

2.3.5.1 检查程序大小和配置轴的数量

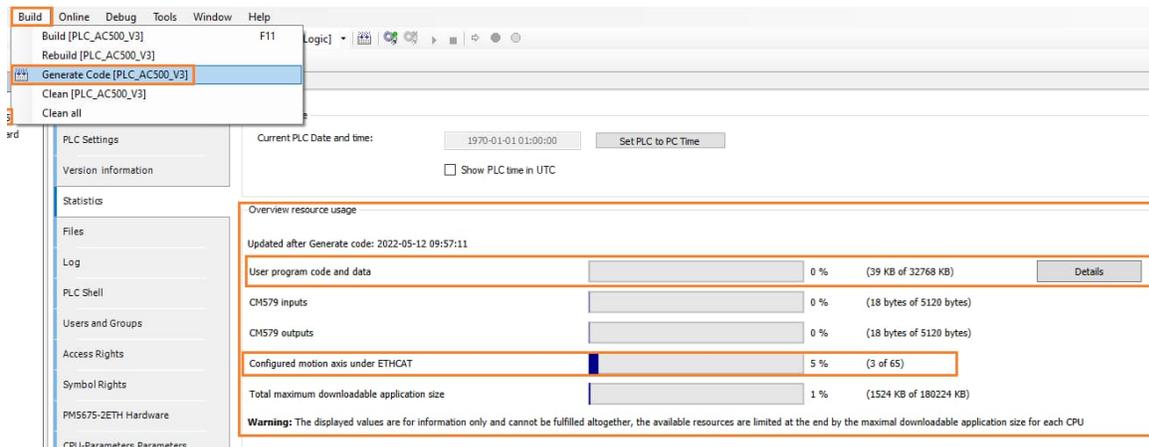
用户可以从“统计”选项卡中检查程序大小以及已配置和支持的轴数。访问此页面。

一. 要更新统计信息以匹配最新代码，用户首先需要从“生成”菜单执行“生成代码”

二. 双击“PLC_AC500_V3”。

三. 在编辑器视图中将打开一个选项卡。

四. 选择“Statistics”。



对于“用户程序代码和数据”的限制，可以使用[详细信息]按钮。单击此按钮将打开一个窗口，显示内存使用情况的更详细视图

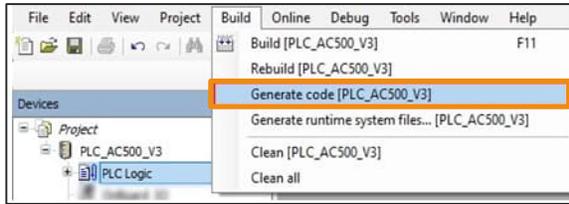
2.3.6 登录 CPU 并下载程序。

登录 CPU 时，项目将加载到 AC500 V3 CPU 中。首次登录还将加载硬件设置。

2.3.6.1 准备代码

在登录 CPU 之前，您需要编译代码，不要有任何错误。

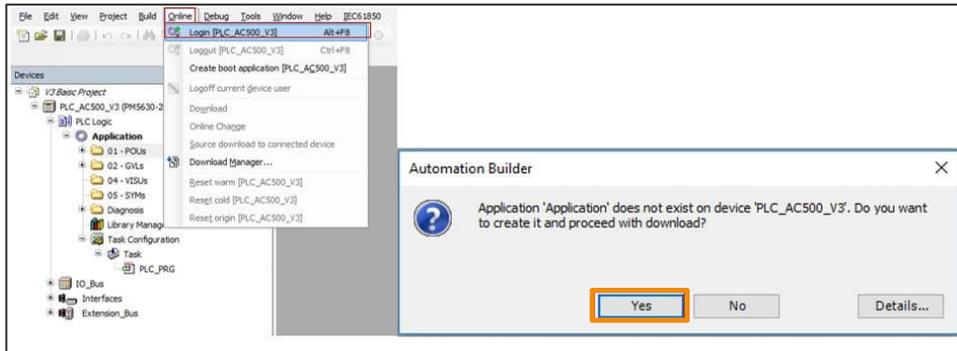
- 为此，请选择菜单“生成 -> 生成代码”。编译结果显示在屏幕底部的“消息”字段中。



- 如果跳过编译并选择“登录”，Automation Builder 将自动触发编译提前登录。

2.3.6.2 下载到单个 CPU

- 一. 在 Automation Builder 菜单中，选择“Online -> Login [PLC_AC500_V3]”。
- 二. 将出现一个弹出窗口，选择“是”将应用程序下载到 AC500V3 CPU。



- 三. PLC 将处于 STOP 模式，如窗口底部的状态栏所示。

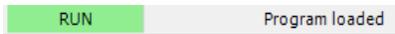


- 四. 默认情况下，下载会在 CPU 中生成以下操作：

- 一. 项目存储在 RAM 内存中。
- 二. 如果创建了引导应用程序，则项目存储在闪存 EEPROM 中。要强制执行此操作，请选择菜单“在线 -> 创建启动应用程序”。

2.3.6.3 将程序设置为“运行”

菜单选择“Debug -> Start [PLC_AC500_V3]”以运行 PLC。窗口底部的状态栏将显示此状态更改

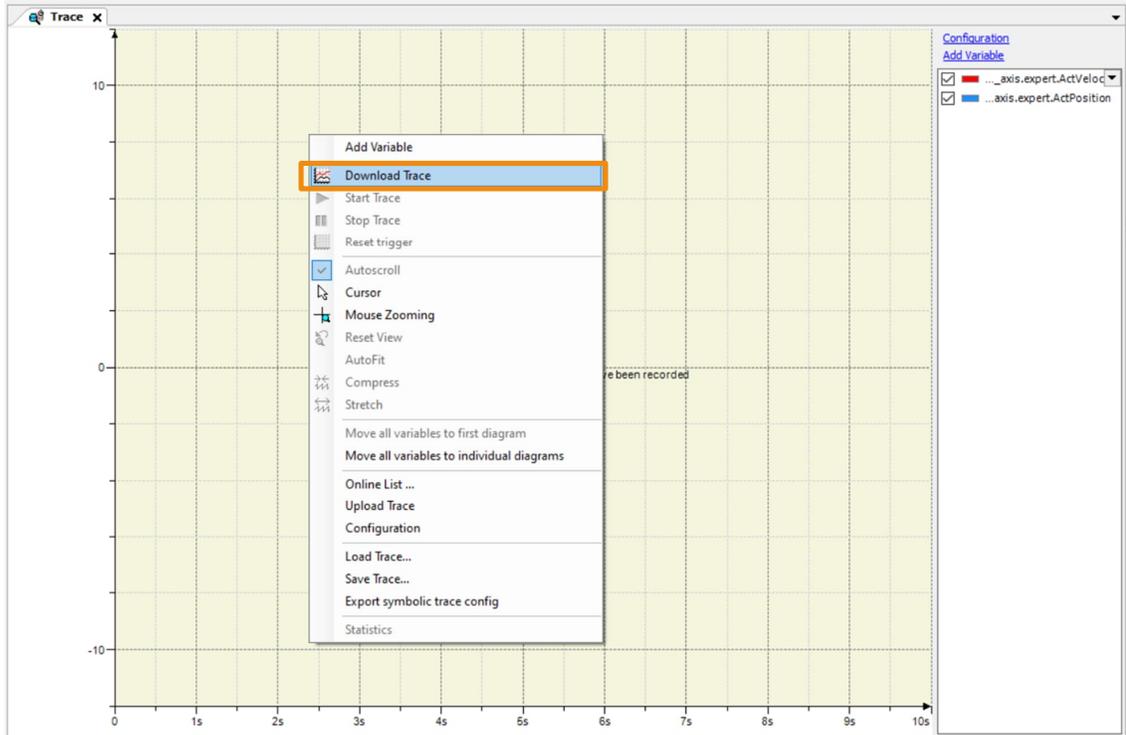


2.3.7 测试程序并使用程序控制轴。

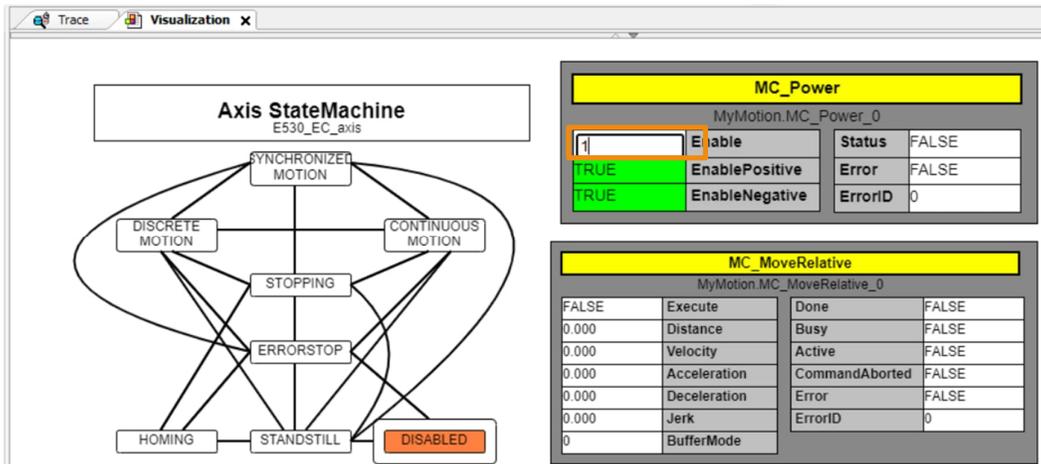
现在您的项目已上线，我们可以测试您编写的项目。为此，请执行以下操作：

- 一. 导航到 Visulaisation 页面。这应该具有与以前相同的布局，但现在会更加丰富多彩，根据用户输入在每个框中显示真实值，但在我们使用它之前，我们必须下载跟踪。

二. 为此，我们可以导航到“跟踪”选项卡，右键单击背景，然后选择“下载”：



三. 然后，我们可以通过单击 MC_Power – 启用输入并将其从 False 更改为 1/True 来重新打开 Visualisation 选项卡并启用驱动器：



四. 现在轴已启用 (MC_Power.Status = TRUE, 轴状态机显示 Standstill), 我们已准备好加载测试移动。如前所述, 在我们以度为单位缩放轴单位之前, 我们在这里的设置必须反映:

The screenshot shows the 'Axis StateMachine' diagram for 'E530_EC_axis'. The states include SYNCHRONIZED MOTION, DISCRETE MOTION, CONTINUOUS MOTION, STOPPING, ERRORSTOP, HOMING, STANDSTILL, and DISABLED. The 'STANDSTILL' state is highlighted in green. To the right, two tables show parameter settings:

MC_Power			
MyMotion.MC_Power_0			
TRUE	Enable	Status	TRUE
TRUE	EnablePositive	Error	FALSE
TRUE	EnableNegative	ErrorID	0

MC_MoveRelative			
MyMotion.MC_MoveRelative_0			
1	Execute	Done	FALSE
3600.000	Distance	Busy	FALSE
720.000	Velocity	Active	FALSE
360.000	Acceleration	CommandAborted	FALSE
360.000	Deceleration	Error	FALSE
0.000	Jerk	ErrorID	0
0	BufferMode		

五. 当我们准备好时, 我们可以选择 Execute, 对于大多数块, 当它们正在处理时, 它们将显示 Busy 和 or Active 为 TRUE (并且 Axis Statemachine 显示 Discrete Motion):

The screenshot shows the 'Axis StateMachine' diagram with 'DISCRETE MOTION' highlighted in green. The parameter tables are updated as follows:

MC_Power			
MyMotion.MC_Power_0			
TRUE	Enable	Status	TRUE
TRUE	EnablePositive	Error	FALSE
TRUE	EnableNegative	ErrorID	0

MC_MoveRelative			
MyMotion.MC_MoveRelative_0			
TRUE	Execute	Done	FALSE
3600.000	Distance	Busy	TRUE
720.000	Velocity	Active	TRUE
360.000	Acceleration	CommandAborted	FALSE
360.000	Deceleration	Error	FALSE
0.000	Jerk	ErrorID	0
0	BufferMode		

然后

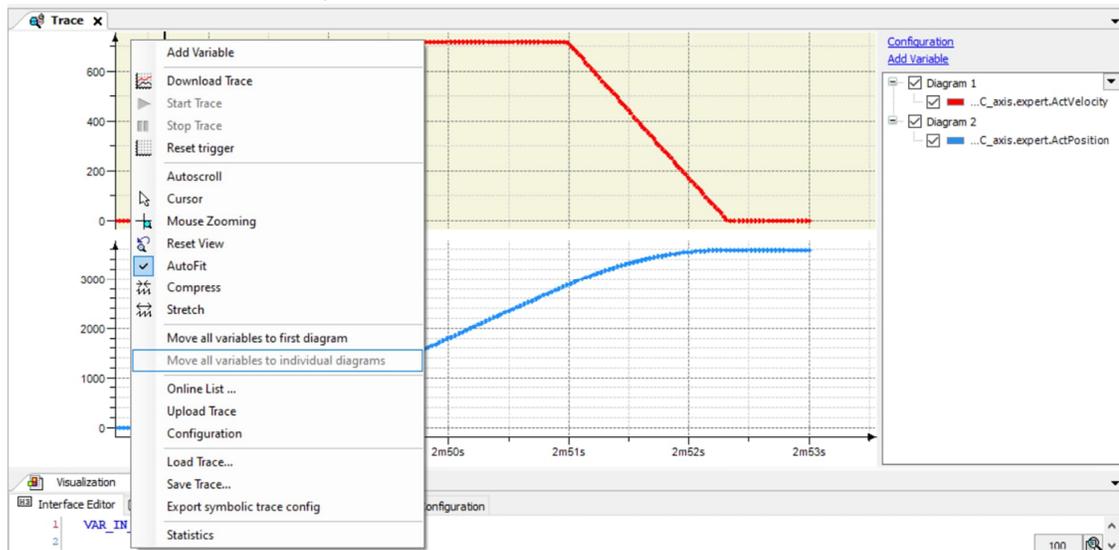
完成为 TRUE (并且 Axis Statemachine 显示 Standstill)。

六. 现在, 如果我们转到“跟踪”选项卡, 我们可以看到数据已被捕获:



这里的黑线显示触发条件, 然后之后的数据显示之前配置的定义点数 (7.996 秒数据)

七. 右键单击还提供了所有关键选项，用于操作（放大、缩小、堆叠等）、导入或导出“跟踪”选项卡中的数据：



八. 现在轴配置已经完成，我们可以禁用并注销或做进一步的测试。

联系我们

欲了解更多信息，请联系您的

当地 ABB 代表或以下人员之一：

new.abb.com/drives/low-voltage-ac/servo-products

new.abb.com/drives

new.abb.com/drivespartners

new.abb.com/PLC

© 版权所有 2022 ABB。

保留所有权利。

规格如有更改，恕不另行通知。