

通过 IEC 61131-3 和 PLCopen 功能块创建可重用的硬件无关运动控制应用程序

I. INTRODUCTION

基于应用要求和项目规格，工程师需要使用广泛的运动控制硬件。过去，这需要为每个应用创建独特的软件，即使功能相同。PLCopen 运动标准提供了一种方式，可以拥有可重用的标准应用程序库，以用于多个硬件平台。这降低了开发、维护和支持成本，同时消除了混淆。此外，工程变得更简单，培训成本降低，软件可重用。实际上，这种标准化是通过定义可重用组件的库来完成的。通过这种方式，编程与硬件的依赖性减少，应用软件的可重用性增加，培训和支持的成本降低，应用在不同的控制解决方案中变得可扩展。由于数据隐藏和封装，它可以用于不同的架构，从集中式到分布式，或从集成式到网络控制。它不是在不同领域的持续定义。因此，它对现有和未来的技术是开放的。

II. OVERVIEW OF THE RESULTS

目前，PLCopen 运动控制规范的套件由以下几个部分组成：

- 第 1&2 部分 - 基础 & 扩展合并为一
- 第 3 部分 - 用户指南和示例
- 第 4 部分 - 协调运动
- 第 5 部分 - 回原点程序
- 第 6 部分 - 液压动力的扩展

基本上，每个规范包含 3 个部分：

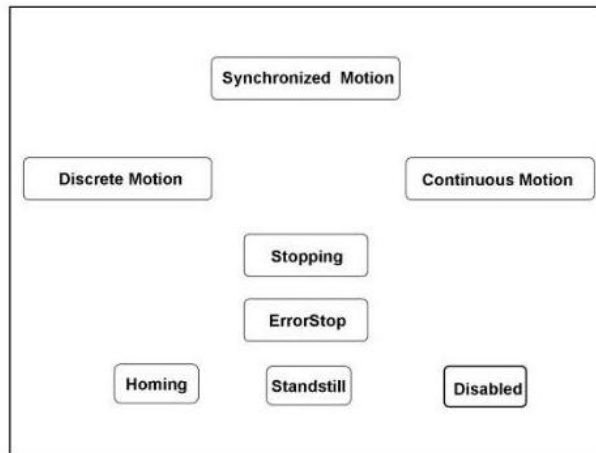
1. 状态机的定义
2. 单轴和多轴运动控制的基本功能块（FB）的定义
3. 合规规则和声明程序。

III. BASICS DEFINITION OF THE STATE MACHINE

轴始终处于定义的某种状态中（见图）。任何运动命令都是改变轴状态的过渡，因而改变当前运动的计算方式。

状态图规范性地定义了轴在高层次上的行为。该图对于构建更复杂的轮廓或在程序中处理例外情况非常有用。（在实际实现中，可能会定义更多的低级状态）。

如下图所示，共定义了八个状态：



正常程序从禁用状态开始。在此状态下，可以通过每个轴切换电源（通过命令 MC_Power），将相关轴转移到待机状态。从那里可以访问归位状态（通过对每个轴发出命令 Home），正常完成后返回到待机状态。从这里可以将轴转移到离散运动或连续运动中。从这些状态中，

可以通过发出 MC_GearIn 实现与主轴的耦合。然后，从动轴的结果状态为同步运动。发出单个轴移动命令将使轴恢复到离散或连续运动。通过停止状态，可以返回到待机状态。错误停止是轴在发生错误时转移到的状态。通过（手动）重置命令，可以返回到待机状态，机器可以再次移动到操作状态。请注意，状态定义了功能块的功能性。

IV. FUNCTION BLOCKS DEFINITIONS

A. AxisRef

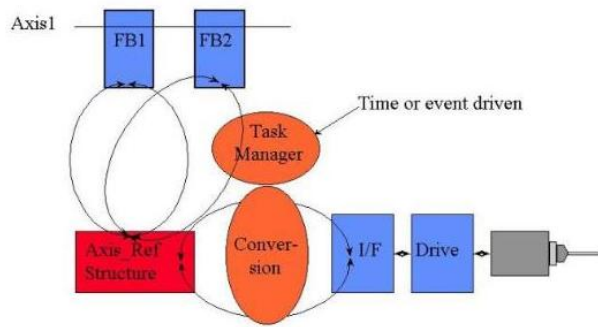
对轴的引用是通过派生数据类型 AXIS_REF 实现的。所有制造商都提供这种数据类型。它提供了与电机/驱动器本身的接口。真实接口的技术细节隐藏在结构和功能块内部。通过这种方式，从集中式到分布式和网络化的不同架构对用户来说看起来是一样的，同时可以访问所有相关参数。

B. AxisRefas Var_In_Out

Axis_Ref 被用作 Var_In_Out，在函数块的图形表示中，表示为一个输入和一个通过一条水平线连接的输出。在 Axis_Ref 中使用的变量，作为输入和输出参数，同时可以在函数块内进行修改，并从外部变量接收值。然而，它们存储在 FB 外部，因此不需要复制结构。

作为这种操作的一个例子：假设一个程序包含几个函数块，彼此串联（从左到右），并通过 Axis_Ref 都引用相同的轴。第一个 FB 在完成其执行后读取最新的值。然后下一个 FB 启动并读取 Axis_Ref 中的更新值，从而使用最新的值。这些值内部与电动机本身耦合。同样，控制架构在不同的系统中可能会有很大的不同。

人们可以利用这个引用定义一个或多个虚拟轴，从某种意义上说，它存在作为一个数据结构，但并不与物理驱动和/或电动机耦合。

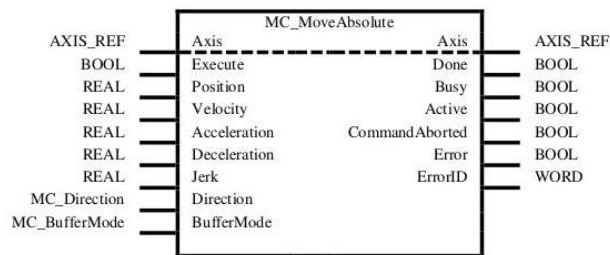


V. THE FUNCTION BLOCKS FOR SINGLE AXIS MOTION CONTROL

在第一部分和第二部分中都定义了与运动相关和管理功能模块。第一个运动功能模块在这里显示得更详细：

FB-Name	MC_MoveAbsolute
This function block commands a controlled motion at a specified absolute position.	

在图形表示中看起来像：



在第 1 部分和第 2 部分中定义的其他单轴功能块在此简要列出：

- MC_MoveRelative - 使轴相对于执行时的实际位置移动指定距离；
- MC_MoveAdditive - 在离散运动状态下，相对于原始命令位置指定额外的相对距离。在连续运动状态下，指定的相对距离被加到执行时的实际位置上；
- MC_MoveSuperimposed - 为现有运动指定额外的相对距离。现有运动不会中断，但会被附加运动叠加；
- MC_HaltSuperimposed - 命令停止所有叠加的轴运动。基础运动不会被中断；
- MC_MoveVelocity - 在指定速度下进行永无止境的受控运动；
- MC_MoveContinuousAbsolute & MC_MoveContinuousRelative - 命令受控运动到指定的绝对或相对位置，最后以指定速度结束；
- MC_TorqueControl - 持续施加指定幅度的扭矩或力，采用定义的斜坡接近，并在达到扭矩水平时设置 “InTorque” 输出；
- MC_SetPosition - 通过操控轴的设定点位置和实际位置的相同值，不造成任何运动来移动轴的坐标系；

- MC_SetOverride - 为整个轴及所有在该轴上工作的功能设置覆盖值；
- MC_TouchProbe - 在触发事件时记录轴的位置；
- MC_AbortTrigger - 用于中止连接到触发事件的功能块（例如 MC_TouchProbe）；
- MC_DigitalCamSwitch - 提供与电机轴上的开关类比的功能：它控制一组离散输出位以对应应该轴连接的一套机械凸轮控制开关。允许前后移动；
- MC_Home - 命令轴执行 \hat{n} 寻回零点 \hat{z} 序列。该序列的细节依赖于制造商，并可以通过轴参数以及第 5 部分 - 回零程序中定义的功能块进行设置；
- MC_Stop - 命令进行控制运动停止，并将轴转移到“停止”状态。它中止任何正在进行的功能块执行。设置“完成”输出后，状态转换为“静止”。在轴处于“停止”状态时，无法在同一轴上执行其他功能块的运动；
- MC_Halt - 命令进行控制运动停止。它中止任何正在进行的功能块执行。轴移动到“离散运动”状态，直到速度为零。设置“完成”输出后，状态转换为“静止”；
- MC_Power - 开关电源级的开或关；
- MC_ReadStatus - 详细返回所选轴状态图的状态；
- MC_ReadMotionState - 详细返回轴当前进行中的运动状态；
- MC_ReadAxisInfo - 读取与轴直接相关的模式、输入和某些轴状态信息；
- MC_ReadAxisError - 指示与功能块无关的错误；
- MC_Reset - 通过重置所有内部轴相关的错误并清除待处理命令,使状态从'ErrorStop'过渡到'StandStill'；
- MC_ReadParameter & MC_ReadBoolParameter - 返回特定于厂商的参数值；
- MC_WriteParameter & MC_WriteBoolParameter - 修改特定于厂商的参数值；
- MC_ReadActualPosition - 返回实际位置；
- MC_ReadDigitalInput - 提供由 INPUT_REF 引用的数字输入的值；
- MC_ReadDigitalOutput - 提供由 OUTPUT_REF 引用的数字输出的值；
- MC_WriteDigitalOutput - 将一个值写入由参数'Output'引用的输出一次；
- MC_ReadActualVelocity&MC_ReadActualTorque - 只要启用，返回实际速度或扭矩的值；
- MC_PositionProfile, MC_VelocityProfile & MC_Acceleration-Profile - 指令不同的锁定运动轨迹：时间-位置、时间-速度或时间-加速度轨迹。

VI. COMMON SET OF MULTI-AXES FUNCTION BLOCKS

对于多轴协调运动，定义了一小组集合。该集合将通过额外的应用特定库进行扩展。当前定义的功能块如下：

- CamTableSelect - 通过设置指针到相关表来选择 CAM 表；

- CamIn - 启动 CAM;
- CamOut - 立即将从轴从主轴脱离;
- GearIn - 命令从轴和主轴之间的速度比;
- GearOut - 将从轴从主轴脱离;
- MC_GearInPos - 命令从同步点开始从轴和主轴之间的位置比;
- MC_PhasingAbsolute & MC_Phasing Relative - 在从轴的主位置上创建绝对或相对的相位偏移;
- MC_CombineAxes - 将两个轴的运动组合成一个第三轴, 并选择组合方法。

A. Aborting, merging, and blending

多个功能块有一个输入, 用于设置不同的操作模式, 并结合一个输出以进行信号传递。通过这个输入, 功能块可以在“非缓冲模式”(默认行为)或“缓冲模式”下工作。这两种模式之间的区别在于何时开始其操作:

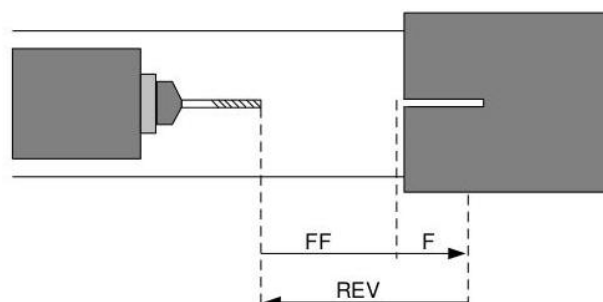
- 非缓冲模式下的命令会立即执行, 即使这会中断另一项动作;
- 缓冲模式下的命令会等到当前功能块完成后(通过相应的输出或通过位置、速度或类似输出进行信号传递)。

已识别出以下模式:

- 中止 - 默认模式, 不进行缓冲。下一个功能块中止正在进行的运动, 命令会立即影响轴;
- 缓冲 - 下一个功能块将在前一个运动完成后(信号为“完成”)影响轴。没有混合;
- 混合低速 - 下一个功能块在前一个功能块完成后控制轴(等同于缓冲), 但轴在运动之间可能不会停止。速度以两个命令(1 和 2)中最低的速度在第一个末端位置(1)进行混合;
- 混合之前 - 在功能块 1 的末端位置以功能块 1 的速度进行混合;
- 混合下一个 - 在功能块 1 的末端位置以功能块 2 的速度进行混合;
- 混合高速 - 在功能块 1 的末端位置以功能块 1 和功能块 2 的最高速度进行混合。

VII. AN EXAMPLE

以下示例是一个简单钻孔单元的例子:



我们在这里使用顺序功能图来描述这个钻孔示例的不同步骤。

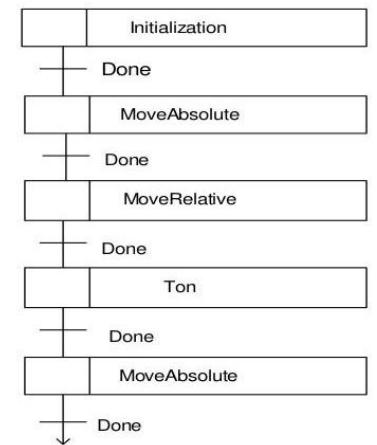
步骤 1：初始化，例如在上电时；

步骤 2：向前移动到钻孔位置并开始钻头旋转：这样在到达位置之前就会完全正常工作；然后检查两个动作是否完成；

步骤 3：钻孔；

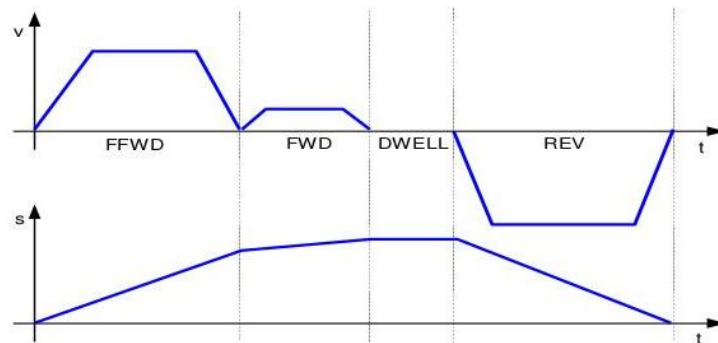
步骤 4：钻孔后，我们必须等待步骤链序列完成，使孔内没有任何可能卡住的东西；

步骤 5：将钻头移动回起始位置并关闭主轴。结合倒退移动的完成和停止主轴，我们向步骤链发出信号以重新开始。

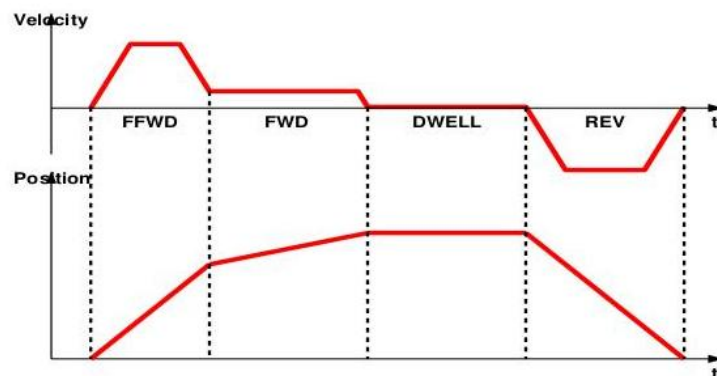


presentation of the drilling example in SFC

运动的相应时序图取决于所选模式。例如：



钻孔 - 中止模式的时序图



钻孔 - 混合模式的时序图

VIII. PART 3 USER GUIDELINES

在第 3 部分，用户示例被解释。这些示例展示了如何基于规格中定义的功能块创建实际应用部分。通过这一点，用户可以创建自己的功能块集，从而创建专门用于他们典型应用领域的自己的库。

举个例子：在第 2 部分，未定义注册功能。为此，功能块 TouchProbe 是预定的。通过一个示例，展示了用户如何基于功能块 TouchProbe 以及已经定义的功能块创建自己的功能块 Registration。通过这种方式，创建了一个特定于应用的功能块，该功能块也可以跨平台使用。以这种方式，可以使用来自传感器的不同输入，其中位置可能未知（直接或联网）和/或需要补偿。这个用户派生的功能块可以添加到公司的自有库中。这个注册功能现在可以在公司范围内使用，源代码可在不同平台上使用。这在下一台机器上节省了时间和金钱。

这一部分是一个持续的规范，随着新示例的添加而发布。有关最新信息，请查看 www.PLCopen.org。

IX. PART 4 COORDINATED MOTION

第 4 部分专注于 3D 空间中的协调多轴运动，以满足大多数用户在该领域的应用需求。第 1 部分和第 2 部分涉及主/从运动控制，这是一种协调运动控制，在这种控制中，主轴位置用于生成一个或多个从轴位置命令。对于多维运动，可以通过将一组轴进行分组而超越这一点，而不使用主轴。这是通过定义一组具有相关协调运动功能的功能块以及一个更高级别的状态图来实现的，该状态图连接组内的单轴状态图。通过这种方式，可以实现更好的轨迹规划。此外，当前的主/从轴可能存在的问题是，如果发生错误，其他轴对此没有了解，并且继续移动。通过将轴组合在一个组中，可以提前知道涉及哪些轴，并为更好的错误处理奠定基础。

PLCopen 运动控制功能块的级别被指定到一个级别，以便用户能够快速识别功能块的功能以及如果它被激活或连接到其他块的运动命令序列中会发生什么。这一 PLCopen 倡议将 CNC 和机器人领域中已知的功能转换到 PLC 领域。

总体而言，定义了一组功能块来实现这一点，如下所示，但不详细说明：有关详细信息，请从网站 www.PLCopen.org 下载规范。

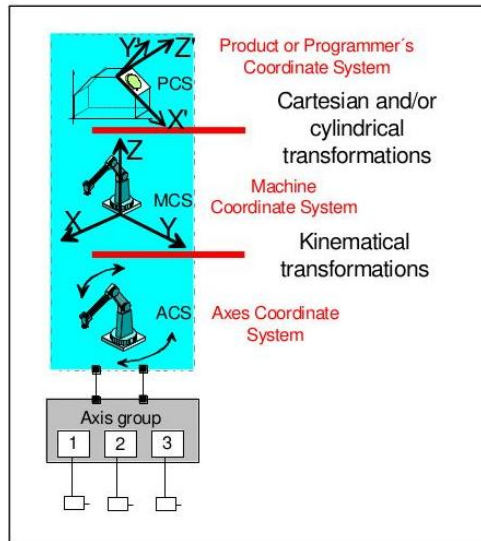
第一步是分组轴。为此定义了以下功能块：

- MC_AddAxisToGroup
- MC_RemoveAxisFromGroup
- MC_UngroupAllAxes
- MC_GroupReadConfiguration
- MC_GroupEnable
- MC_GroupDisable

下一步是链接转换：

- MC_SetKinTransform
- MC_SetCartesianTransform
- MC_SetCoordinateTransform
- MC_ReadKinTransform
- MC_ReadCartesianTransform

- MC_ReadCoordinateTransform
- MC_SetDynCoordTransform



坐标系统和变换的概述

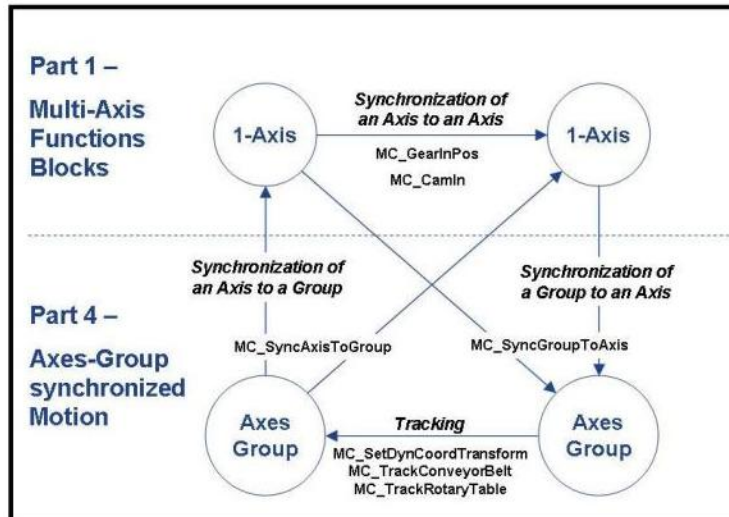
- ACS 轴相关
- MCS 机器相关
- PCS 产品或工件相关

对于协调移动，定义了以下区块：

- MC_GroupHome - MC_MoveLinearRelative
- MC_GroupStop - MC_MoveCircularAbsolute
- MC_GroupHalt - MC_MoveCircularRelative
- MC_GroupInterrupt - MC_MoveDirectAbsolute
- MC_GroupContinue - MC_MoveDirectRelative
- MC_MoveLinearAbsolute - MC_MovePath

Axes Group Synchronized Motion

以下功能块处理单个或一组轴与单个或一组轴之间的主/从关系，用于协调目的。从编程的角度和运动控制本身的实现来看，需要区分两种协调运动。这两种模式在这里通过其名称进行识别：同步和跟踪。两者之间的差异及相关的功能块在下图中展示：



协调的图形说明

然后定义了其他功能块，大多数没有运动：

- MC_GroupSetPosition - MC_GroupReadError
- MC_GroupReadActualPosition - MC_GroupReset
- MC_GroupReadActualVelocity - MC_PathSelect
- MC_GroupReadActualAcceleration - MC_GroupSetOverride
- MC_GroupReadStatus

X. PART 5 HOMING PROCEDURES AND BLOCKS

A. Homing Procedures

归位功能最初被视为机器或轴启动阶段的一个独立序列。对于一个轴的序列是：开机，归位，然后移动“某个东西”。整个归位过程本身对用户是隐藏的。然而，一些用户需要对归位功能本身有更多的控制。为此，已经识别了一组构建块，以便能够定义专用的归位程序。该过程可以封装在一个专用的归位功能块中，该功能块可以添加到库中并用于这个特定的程序。作为示例，定义了几种可能的归位程序：

- HomeAbsSwitch - 绝对开关归位加限位开关；
- HomeLimitSwitch - 对限位开关进行归位；
- HomeBlock - 针对阻碍运动的硬件部件进行归位；
- HomeRefPulse - 使用编码器参考脉冲“零标记”进行归位；
- HomeRefPulseSet - 使用一组编码器参考脉冲“零标记”进行归位；
- HomeDirect - 静态归位，强制用户参考位置；
- HomeAbsolute - 静态归位，强制绝对编码器的位置。

B. Homing Step Function Blocks

为了将这些程序提供给用户，定义了一个工具包，其中包含额外的功能模块（Function Blocks, FBs）。通过使用一个或多个连接的功能模块，可以创建复杂的归位序列。这样，用户可以更详细地创建自己的归位程序，甚至可以创建用户特定的归位程序。此外，这些功能模块还包括高级归位过程错误报告，评估时间、距离和扭矩限制。

以下定义的 FBs 对应于归位程序，并使状态更改为“归位”（Homing）状态。

- MC_StepAbsSwitch - 绝对开关归位加限位开关；
- MC_StepLimitSwitch - 反向限位开关归位；
- MC_StepBlock - 反向硬件部件阻碍运动的归位；
- MC_StepRefPulse - 使用编码器参考脉冲“零标记”的归位。

通过这种方式，可以实现任何组合序列，而无需预先定义数百种归位方法。扭矩限制、时间限制和距离限制的单独调整提供了对序列错误条件的控制。

以下 FBs 导致最终位置并将归位状态更改为“静止”（StandStill）：

- MC_HomeDirect - 强制用户参考位置的静态归位；
- MC_HomeAbsolute - 强制绝对编码器位置的静态归位；
- MC_FinishHoming - 完成归位程序，可以相对移动到操作区域。

此外，在机器操作时也需要归位功能，即不处于“归位”（Homing）状态。这种“飞行归位”称为被动归位。它们对状态图没有影响，就像管理 FBs 一样，可以在任何运动状态下调用。它们包括：

- MC_StepReferenceFlyingSwitch
- MC_StepReferenceFlyingRefPulse
- MC_AbortPassiveHoming

XI. PART 6 FLUID POWER

这一部分的目标是通过定义使用 PLCopen 标准化和模块化方法的功能块来优化流体动力设备和系统的编程与集成，基于现有的集合，并导致数量有限的额外功能块。

主要目标是降低市场响应的限制，并在不考虑特定技术构造的情况下提供问题解决方案。标准化的接口将允许电力和流体动力资源池之间的交叉融合，并推动最佳解决方案配置。

定义的功能块包括：

- MC_LimitLoad - MC_LoadControl
- MC_LimitMotion - MC_LoadSuperImposed
- MC_LoadProfile

XII. CERTIFICATION

包含在文档中的合规和认证规则。基本上，这是一个自我认证，供应商的结果会公布在 PLCopen 网站 www.PLCopen.org 上。获得认证的公司可以使用下面的徽标，并附上额外的编号、日期和所支持的合规功能块的数量：



XIII. CONCLUSION

规范的发布和第一次实现清楚地表明，多轴实现确实符合 IEC 61131-3 标准所定义的框架。随着许多实现的可用，运动控制将永远不再相同：更大的硬件独立性为用户降低了培训成本，并且更容易创建可用于多个不同目标的应用软件，同时这些目标可以在整个机器开发周期的后期阶段选择。这带来了更高的效率，并在开发、维护和培训期间降低了成本。

有关最新信息、规范本身以及 PLCopen 在安全、通信和交换方面的贡献，请查阅 www.PLCopen.org。

警告：该 PDF 由 302.AI 调用大语言模型一键生成，版权归原文作者所有。翻译内容可靠性无保障，请仔细鉴别并以原文为准。当前大语言模型: gpt-4o-mini-2024-07-18 为了防止大语言模型的意外谬误产生扩散影响，禁止移除或修改此警告。