

密级：



项目名称：	BricksPro	项目编号：	
项目经理：		编 制：	
审 核：		批 准：	

ER 系列 RCS2 系统 PLC 软件安装及标准工程使用手册

V0.10

文档修订记录

序号	版本号	修订日期	修订概述	备注
1	V0.1	2020.06.28	新建文档	
2	V0.2	2021.02.05	添加部分库功能；添加新增固件库用法；对部分已有功能进行了调整(适配控制 RCS2_V1.22 版本)。	
3	V0.3	2021.08.04	部分库添加新功能；添加新用户库 ER_EtherNetIP（适配控制器 RCS2_V1.24）。	
4	V0.4	2021.12.30	添加 ModBusTCP 寄存器读写功能。	
5	V0.5	2022.04.11	修改 SIMDI/AI 功能块，支持读写	
6	V0.6	2023.07.10	新增 ER_JogMotion、ER_RobotMoveEnable 等功能块；对部分已有功能进行了调整(适配控制器 RCS2_V1.31 版本)。	
7	V0.7	2024.02.24	部分库添加新功能；添加新用户库 ER_MC_GroupControl；添加新固件库 ModbusTCP_Lib（适配控制器 RCS2_V1.33）。	
8	V0.8	2024.07.05	修改 3.1.13、3.1.27、3.1.28 小节功能块帮助内容；新增 3.1.33、3.1.34、3.1.35 小节；对 3.8 章节中 Modbus 读写寄存器索引号范围的变化添加说明；优化 3.9 章节中各运动功能块的帮助内容；新增 3.9.9、3.9.10 小节(适配控制器 RCS2_V1.34 版本)。	
9	V0.9	2025.01.16	修改了 3.1.24、3.1.25、3.1.33 小节，功能块帮助内容(适配控制器 RCS2_V1.35 版本)。	
10	V0.10	2025.05.26	新增功能块 ER_ExtSignalShielding、ER_StartBrakeTest、ER_GetConveyorPosition、ER_GetAreaActSts、ER_GetPolyAreaActSts	

目录

一、Multiprog 软件安装及环境配置	9
1.1 安装文件	9
1.1.0 安装 0 号文件夹《0_WinPacp_4_1_2》	9
1.1.1 安装 1 号文件夹《1_MULTIPROG 5.5 Express SP1 Acontis Build10228》 ..	12
1.1.2 安装 2 号文件夹《2_eCLR Addon》	18
1.1.3 安装 3 号文件夹《3_ECEngineer for MULTIPROG Express》	20
1.2 配置环境	23
二、模板工程使用	24
2.1 模板工程结构	24
2.2 逻辑 POU 简介	25
2.2.1 Demo 功能块	25
2.2.2 testFB 功能块	25
2.2.3 ER_UserControl、ER_PLCProgMain 程序	26
2.2.4 REAL_TO_DWORD、DWORD_TO_REAL 功能块	26
2.3 基于模板工程开发	27
2.3.1 添加任务 TASK	27
2.3.2 添加逻辑 POU	28
2.3.3 下载调试	29
2.5 其他注意事项	37
三、用户库功能块介绍	39
3.1 ER_PLC2CNT_Command: PLC 命令功能块	41
3.1.1 ER_GetAposData	42
3.1.2 ER_GetCposData	43
3.1.3 ER_GetERCVar	44
3.1.4 ER_KillProject	45
3.1.5 ER_LoadProgram	45
3.1.6 ER_ProgramPause	46
3.1.7 ER_ProgramStart	47
3.1.8 ER_ProgramStop	47

3.1.9 ER_Reset	48
3.1.10 ER_RobotPowerOff	48
3.1.11 ER_RobotPowerOn	49
3.1.12 ER_SetERCVar	50
3.1.13 ER_SetErrorLog	50
3.1.14 ER_SetOverride	51
3.1.15 ER_SetPayLoad.....	52
3.1.16 ER_SetProgPC	53
3.1.17 ER_SetProgRunMode	53
3.1.18 ER_SetRefCoord	54
3.1.19 ER_SetTool	55
3.1.20 ER_TeachPosVar	55
3.1.21 ER_SetAutoLoadPrg	56
3.1.22 ER_CancelAutoLoadPrg	57
3.1.23 ER_LoadAutoRunProg.....	57
3.1.24 ER_SetAposData.....	58
3.1.25 ER_SetCposData	59
3.1.26 ER_JogMotion.....	60
3.1.27 ER_RobotMoveDisable.....	61
3.1.28 ER_RobotMoveEnable.....	62
3.1.29 ER_SetRobJogCoord.....	62
3.1.30 ER_SetRobotMode.....	63
3.1.31 ER_SetVirtualSafetyDoor	64
3.1.32 ER_BackupRuntime	64
3.1.33 ER_GetPosInBatch.....	65
3.1.34 ER_SetJogMicroParam	66
3.1.35 ER_JogMicroMotion.....	67
3.1.36 ER_ExtSignalShielding	68
3.1.37 ER_StartBrakeTest	69
3.2 ER_CNT2PLC_Status: 获取状态功能块	70

3.2.1 ER_RemoteAutoLoadProg	71
3.2.2 ER_GetCurCposCfg	71
3.2.3 ER_GetCurCpos	72
3.2.4 ER_GetCurJpos	73
3.2.5 ER_GetGlobalSpeed.....	73
3.2.6 ER_GetProgRunMode.....	74
3.2.7 ER_GetCurLoadProg	75
3.2.8 ER_GetRobPowerOnState	75
3.2.9 ER_GetRobMovState	76
3.2.10 ER_GetProgRunState	76
3.2.11 ER_GetSystemState	77
3.2.12 ER_GetSystemMode	78
3.2.13 ER_GetSysError	78
3.2.14 ER_RobModbusCnctState.....	79
3.2.15 ER_GetRobJogCoord	79
3.2.16 ER_GetAxisServoError.....	80
3.2.17 ER_GetRobotHomeState.....	80
3.2.18 ER_GetRobSysTime	81
3.2.19 ER_GetAxisTorqueValue.....	81
3.2.20 ER_GetRobNearPath.....	82
3.2.21 ER_GetRobOnPath.....	82
3.2.22 ER_GetTool.....	83
3.2.23 ER_GetRefCoord	83
3.2.24 ER_GetMoveEnableState.....	84
3.2.25 ER_GetAxisTorqueNm	84
3.2.26 ER_GetCurTCPVelocity	85
3.2.27 ER_GetRobSysInfo	85
3.2.28 ER_GetRobRunningTime	86
3.2.29 ER_GetRobMaintainSts	86
3.2.30 ER_GetMultiRobColliSts.....	87

3.2.31 ER_GetAxisCurrent	87
3.2.32 ER_GetConveyorPosition	88
3.2.33 ER_GetAreaActSts.....	88
3.2.34 ER_GetPolyAreaActSts	89
3.3 ER_IO_LIB: IO 功能块.....	90
3.3.1 UPDATE_PHY_DI_aaa_bbb.....	90
3.3.2 UPDATE_PHY_DO_aaa_bbb	91
3.3.3 UPDATE_PHY_AI_aaa_bbb.....	92
3.3.4 UPDATE_PHY_AO_aaa_bbb	93
3.3.5 UPDATE_SIMDI_BYTE_aaaa.....	93
3.3.6 UPDATE_SIMDO_BYTE_aaaa.....	94
3.3.7 UPDATE_SIM_AI_aaa_bbb.....	94
3.3.8 UPDATE_SIM_AO_aaa_bbb	94
3.3.9 PHY_DI_PACK	94
3.3.10 PHY_DO_UNPACK.....	95
3.3.11 PHY_DO_PACK.....	96
3.4 ER_PLC_Share_VAR: PLC 变量功能块.....	97
3.4.1 PLCVAR_REAL_1_30.....	98
3.5 ER_MC_Control: 单轴控制功能块.....	100
3.5.1 MC_Home	100
3.5.2 MC_Power.....	101
3.5.3 MC_ReadStatus.....	102
3.5.4 MC_ReadActualPosition.....	102
3.5.5 MC_ReadAxisError.....	103
3.5.6 MC_Stop	104
3.5.7 MC_Reset.....	105
3.5.8 MC_SetOverride.....	105
3.5.9 MC_MoveAbsolute	106
3.5.10 MC_MoveRelative	107
3.5.11 MC_MoveVelocity.....	108

3.6 ER_Calculation: 矩阵运算功能块.....	109
3.6.1 矩阵加减.....	109
3.6.2 矩阵数乘.....	110
3.6.3 矩阵乘法.....	110
3.6.4 向量乘矩阵.....	110
3.6.5 矩阵求逆.....	111
3.6.7 角度弧度互转功能块.....	111
3.7 ER_EtherNetIP: EtherNet/IP 通讯配置功能块.....	112
3.7.1 ER_EIP_SetTrialRunMode.....	112
3.7.2 ER_EIP_GetProductCode.....	112
3.7.3 ER_EIP_SetLicenseCode.....	113
3.8 ER_ModbusTcp: ModbusTCP 功能库.....	114
3.8.1 ER_RobModbusCnctState.....	114
3.8.2 ReadModbusReg.....	114
3.8.3 WriteModbusReg.....	115
3.9 ER_MC_GroupControl: 机器人运动控制功能块.....	116
3.9.1 MC_GroupMoveDisable.....	116
3.9.2 MC_GroupMoveEnable.....	117
3.9.3 MC_GroupContinue.....	117
3.9.4 MC_GroupPause.....	118
3.9.5 MC_GroupStop.....	119
3.9.6 MC_MoveCircularAbsolute.....	119
3.9.7 MC_MoveDirectAbsolute.....	121
3.9.8 MC_MoveLinearAbsolute.....	122
3.9.9 MC_MoveArch.....	123
3.9.10 MC_MoveArchLinear.....	124
四、固件库功能块介绍.....	126
4.1 INI_File_WR.....	126
4.1.1 读取文件功能块.....	126
4.1.2 写入文件.....	127

4.2 TCP_IP_Lib.....	127
4.2.1 TcpClient	127
4.2.2 TcpServer	128
4.2.3 TcpRecvBuf.....	128
4.2.4 TcpSendBuf.....	129
4.2.3 TcpRecvStr.....	129
4.2.4 TcpSendStr	130
4.3 ModbusTCP_Lib	130
4.3.1 Modbus_Client	130
4.3.2 Modbus_ReadRegs.....	131
4.3.3 Modbus_WriteRegs.....	132
4.3.4 Modbus_WriteSingleReg	133

一、Multiprog 软件安装及环境配置

本软件推荐优先选择 Windows7、Windows8 及 Windows10 操作系统进行安装部署，以确保软件功能完整实现与运行稳定性。

1.1 安装文件

MULTIPROG Express5.5 软件的安装文件夹为 MULTIPROG Express Install。

名称	修改日期	类型	大小
MULTIPROG Express Install	2015/3/6 12:13	文件夹	

文件夹 MULTIPROG Express Install 包括三个文件夹，分别为安装文件夹（Install）、环境配置文件夹（Environment）和工程模板文件夹（Sample）。

名称	修改日期	类型
Environment	2018/6/6 20:52	文件夹
Install	2018/6/6 16:40	文件夹
Sample	2018/6/6 16:48	文件夹

安装文件夹内将需要安装的软件插件都按安装顺序进行了编号，如下图所示，按照编号顺序依次安装即可。

名称	修改日期	类型
0_WinPcap_4_1_2	2015/3/6 12:16	文件夹
1_MULTIPROG 5.5 Express SP1 Acontis Build10228	2015/3/6 12:46	文件夹
2_eCLR Addon	2015/3/6 2:07	文件夹
3_EC Engineer for MULTIPROG Express	2015/3/6 2:07	文件夹

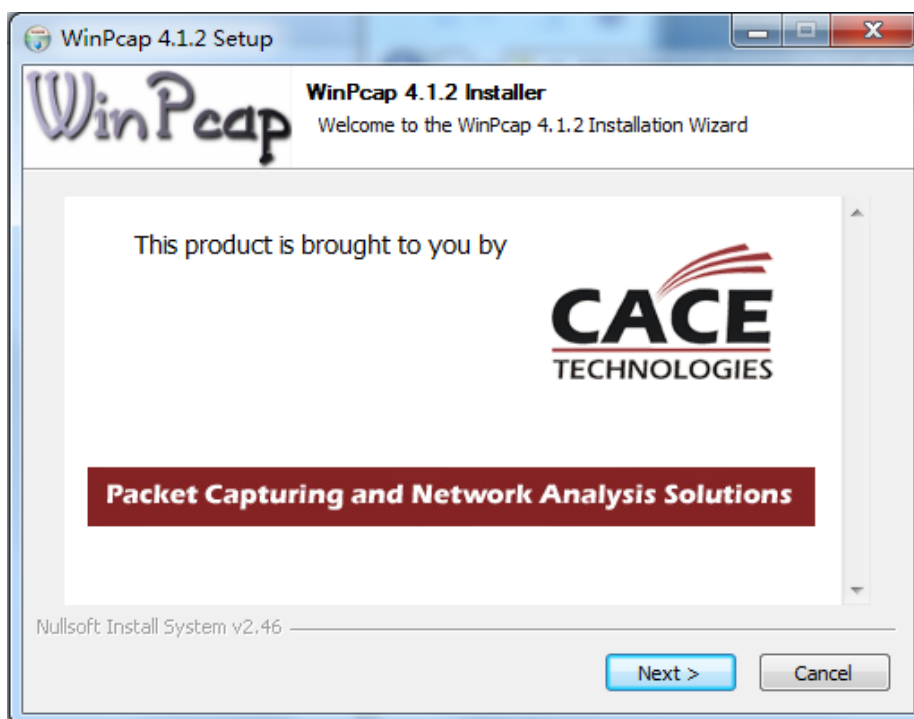
下面详细介绍各个文件夹安装步骤。

1.1.0 安装 0 号文件夹《0_WinPcap_4_1_2》

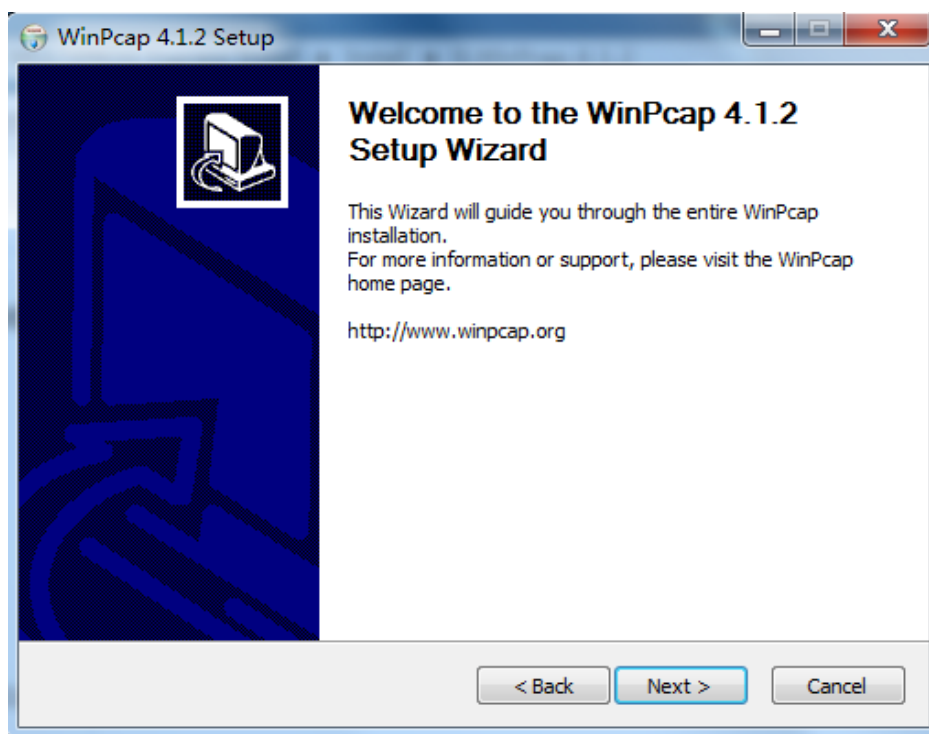
打开 0 号文件夹 0_WinPcap_4_1_2，双击安装包 WinPcap_4_1_2；

名称	修改日期	类型	大小
WinPcap_4_1_2	2011/12/18 23:45	应用程序	895 KB

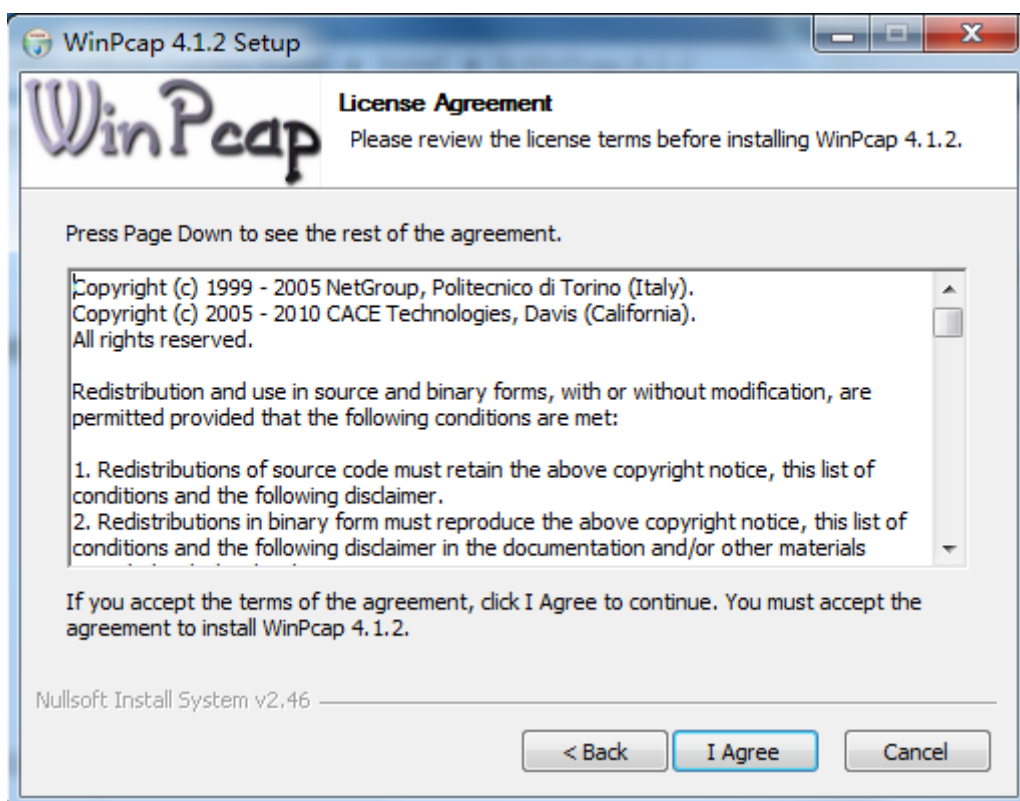
出现下图所示安装界面后单击“Next”；



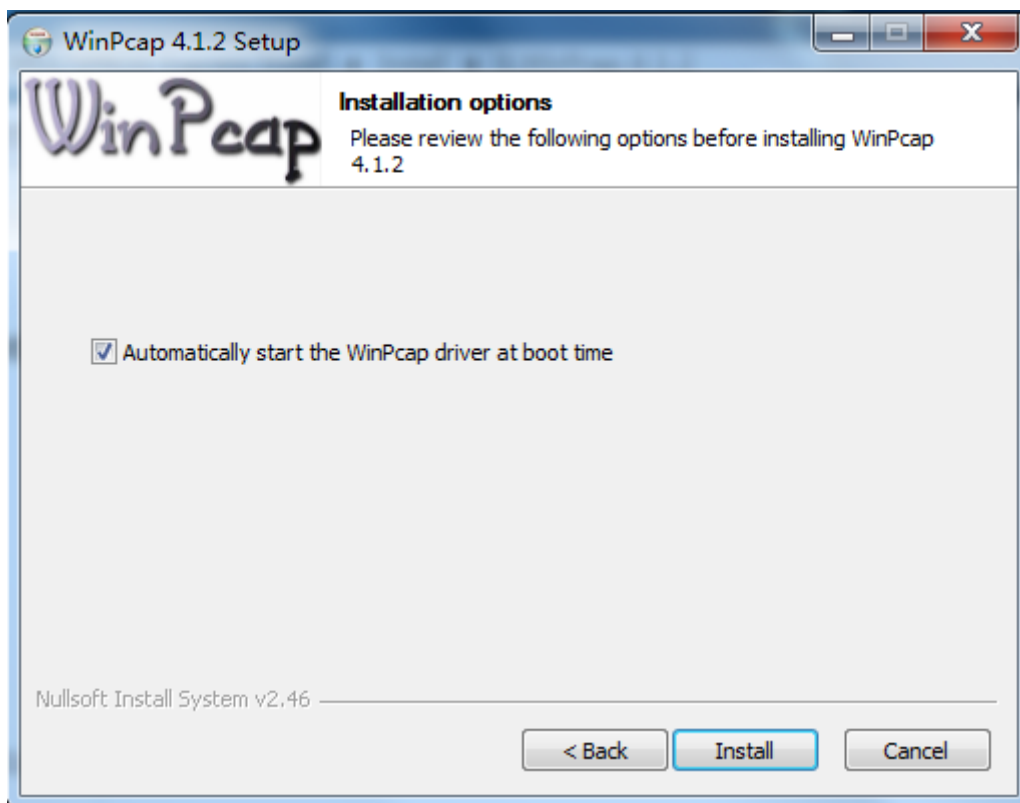
单击“Next”；



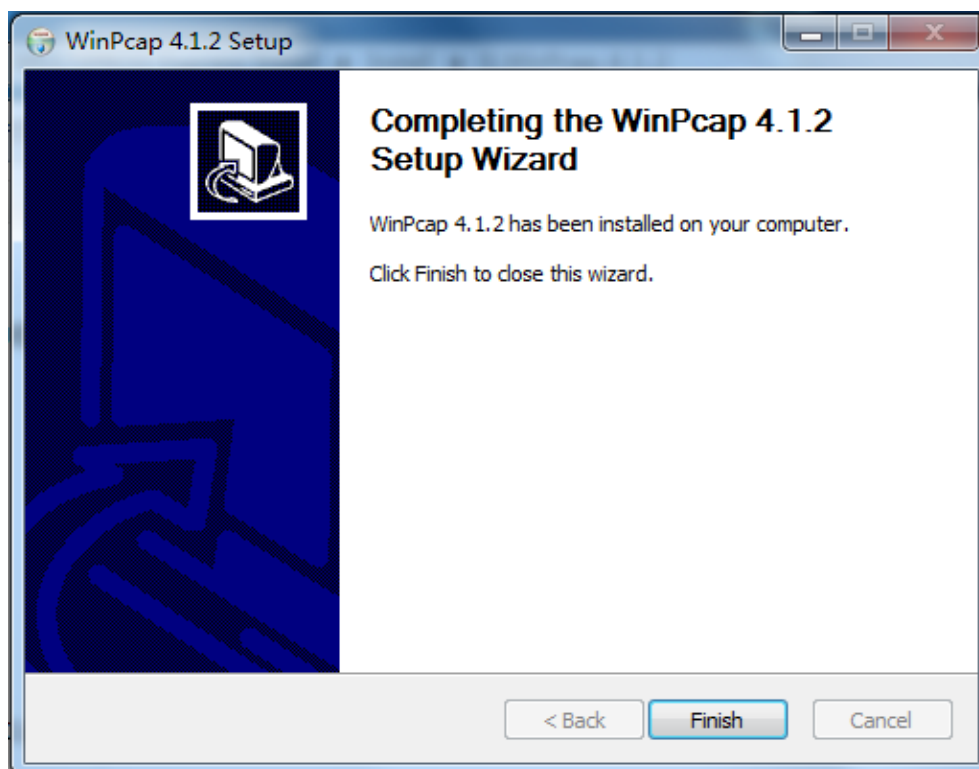
单击 “I Agree”;



默认勾选 “Automatically start the WinPcap driver at boot time”,
单击 “Install” 开始安装。



出现安装完成界面点击 “Finish”，退出安装程序。




1.1.1 安装 1 号文件夹《1_MULTIPROG 5.5 Express SP1 Acontis Build10228》

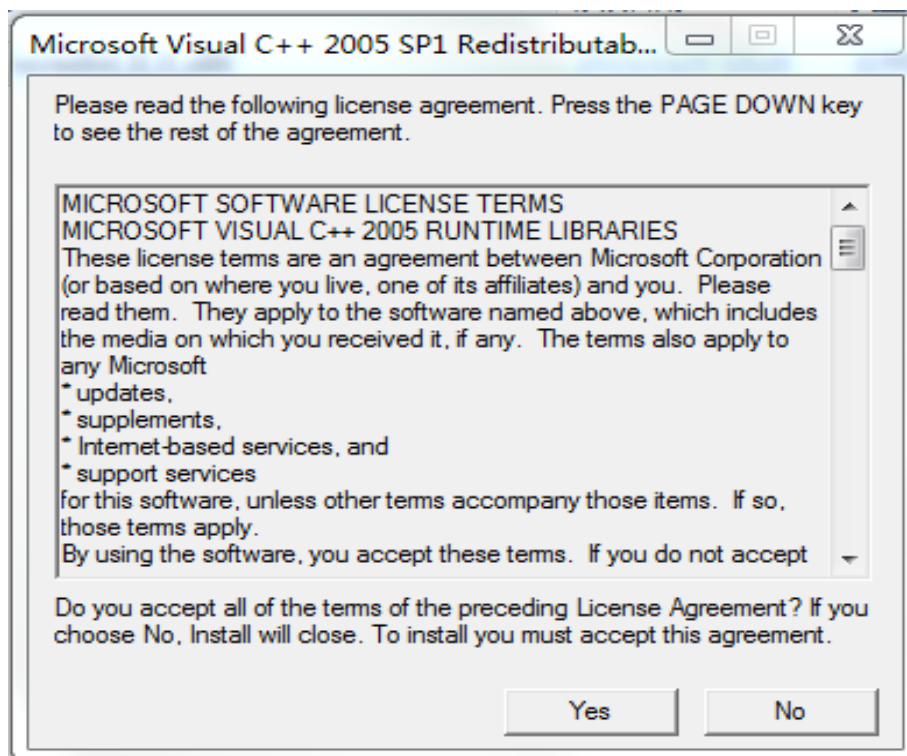
安装 1 号文件夹 1_MULTIPROG 5.5 Express SP1 Acontis Build10228 内程序。1 号文件夹内所需安装内容被编号为 0—3 号文件夹（文件），按编号顺序依次安装。

1.1.1.0 《0 号文件夹“0_vcredist_8_0_x86”》

打开 0_vcredist_8_0_x86 文件夹，双击文件 vcredist_8_0_x86；

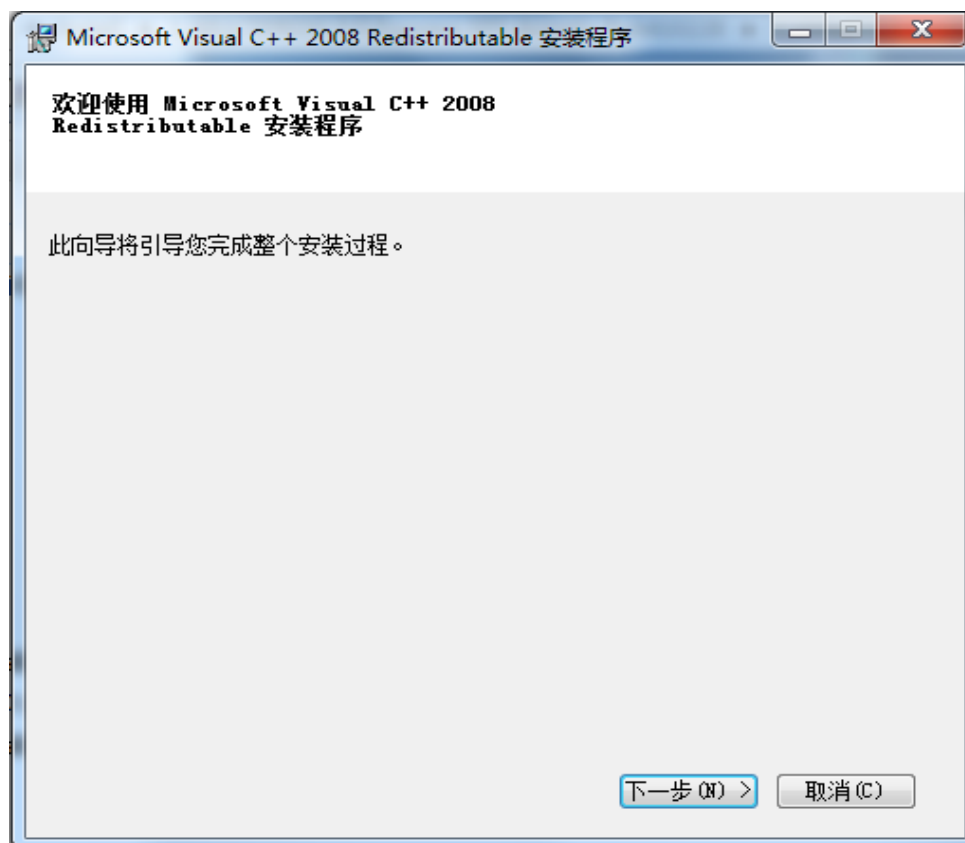
名称	修改日期	类型
 vcredist_8_0_x86	2015/11/3 13:28	应用程序

双击安装文件 vcredist_8_0_x86 后出现图示窗口，选择“**Yes**”确认安装。安装完成后不出现其他窗口。

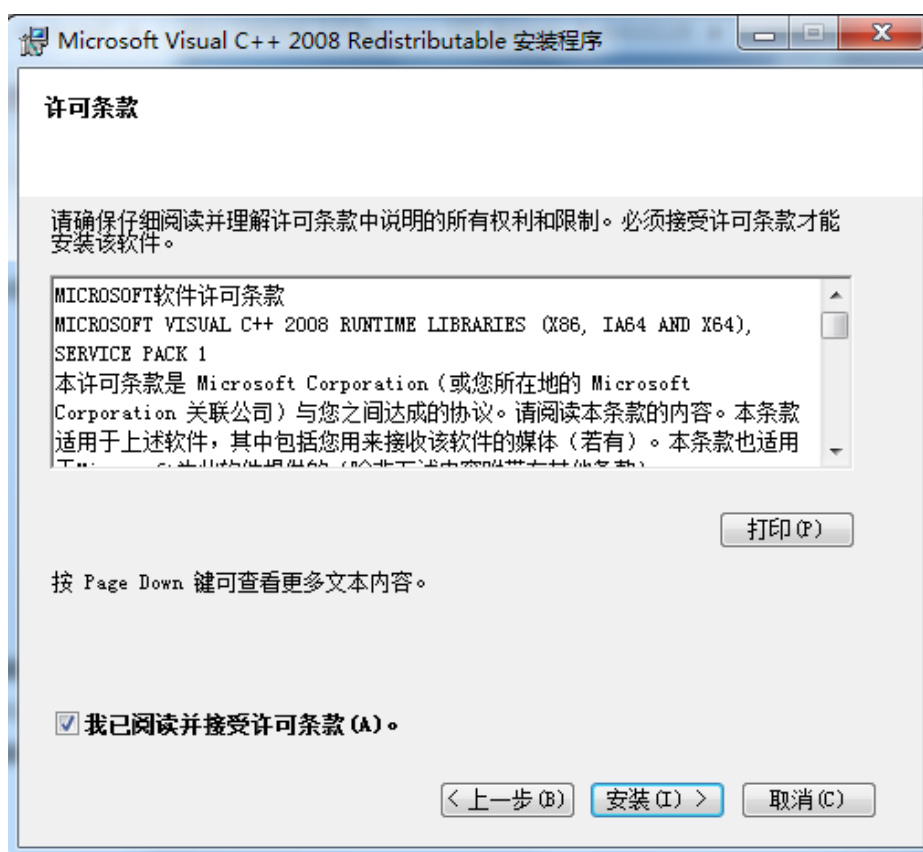


1.1.1.1 安装 1 号文件夹《1_vcrist_9_0_x86》

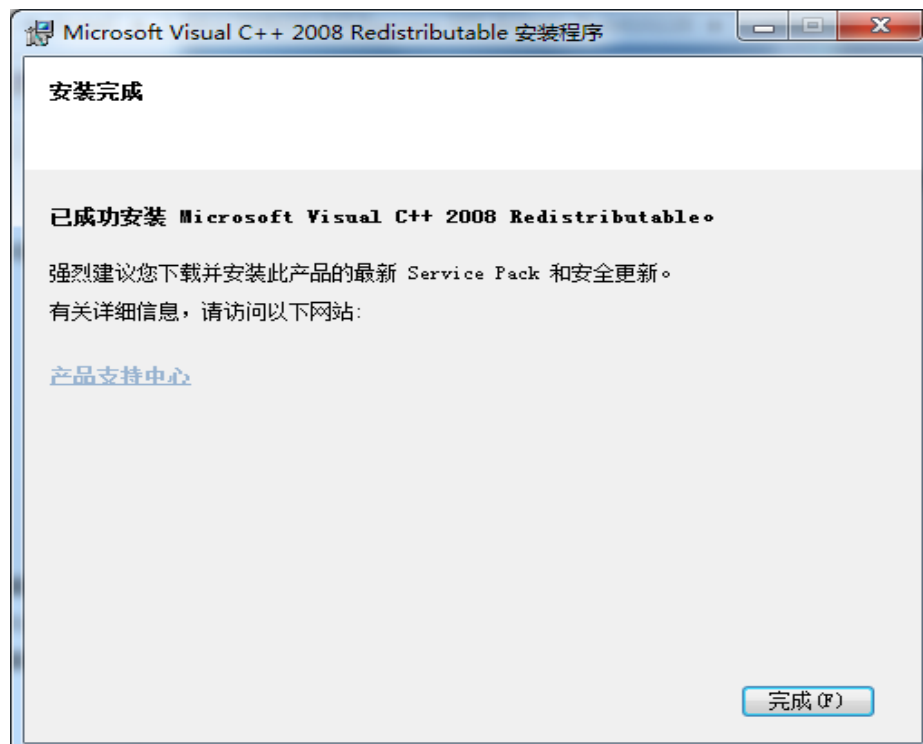
双击文件夹下安装文件“vcrist_9_0_x86”出现下图窗口后单击“下一步”；



勾选“我已阅读并接受许可条款”，单击“安装”；



安装完成后单击“完成”退出安装程序。

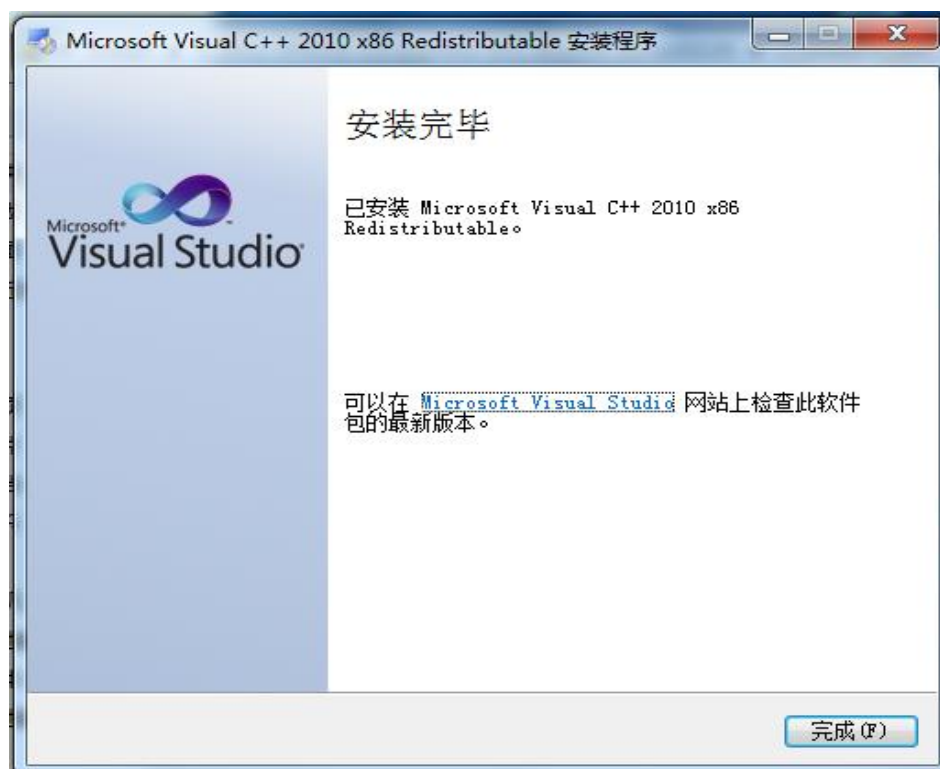


1.1.1.2 安装 2 号文件夹《2_vcrist_x86》

打开 2 号文件夹“2_vcrist_x86”，双击安装文件“vcrist_x86”，勾选“我已阅读并接受许可条款”后，点击安装；

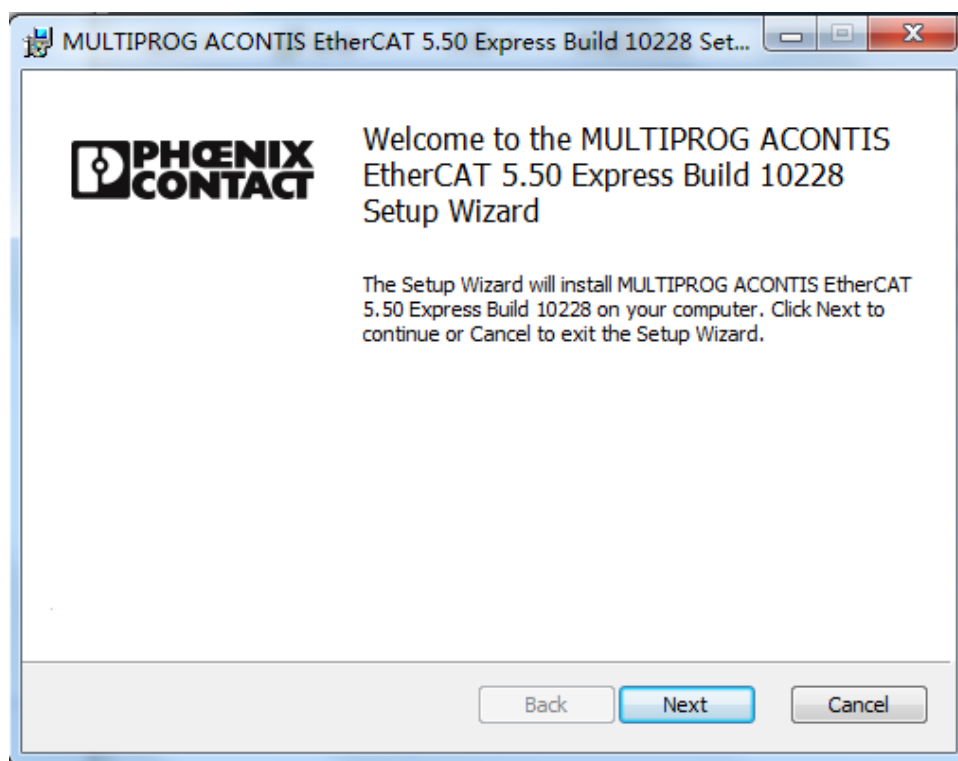


等待安装完成后点击“完成”退出安装程序。

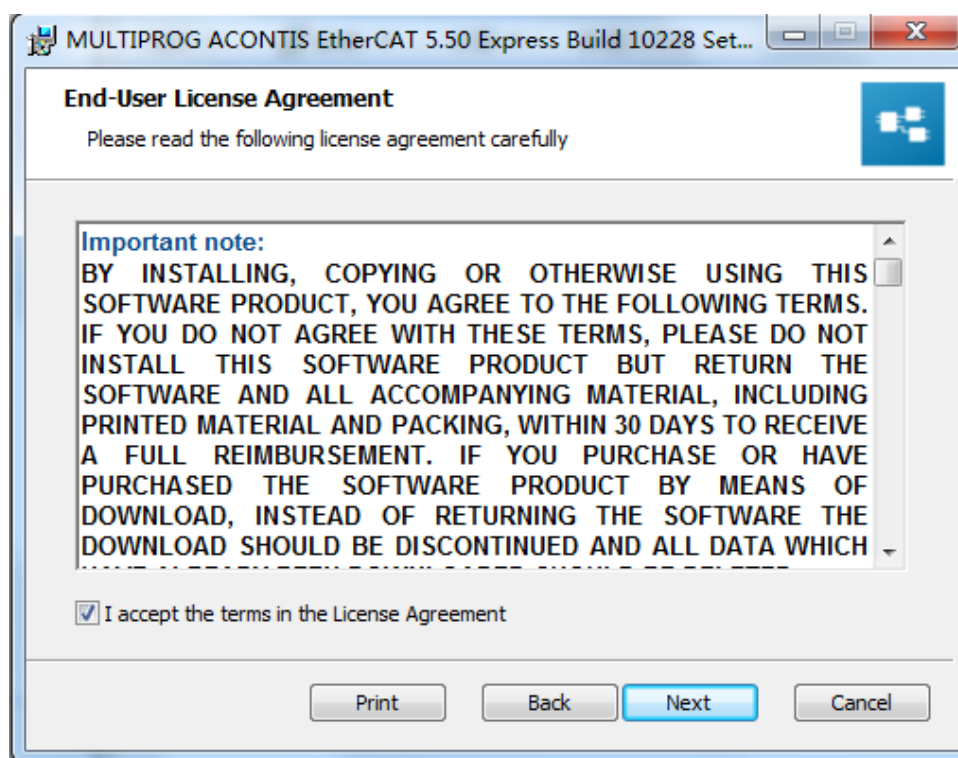


1.1.1.3 安装 3 号文件《3_MULTIPROG ACONTIS EtherCAT 5.50》

安装 3 号文件 3_MULTIPROG ACONTIS EtherCAT 5.50，双击后出现下图安装窗口，点击“Next”；

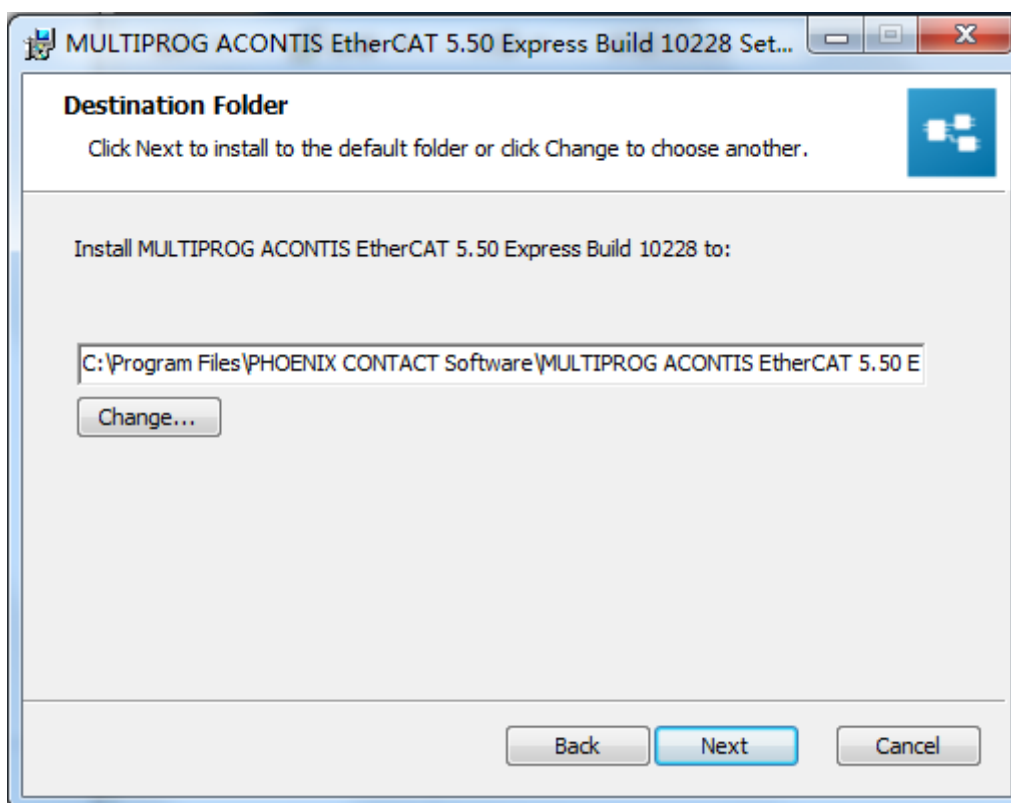


点击“Next”；

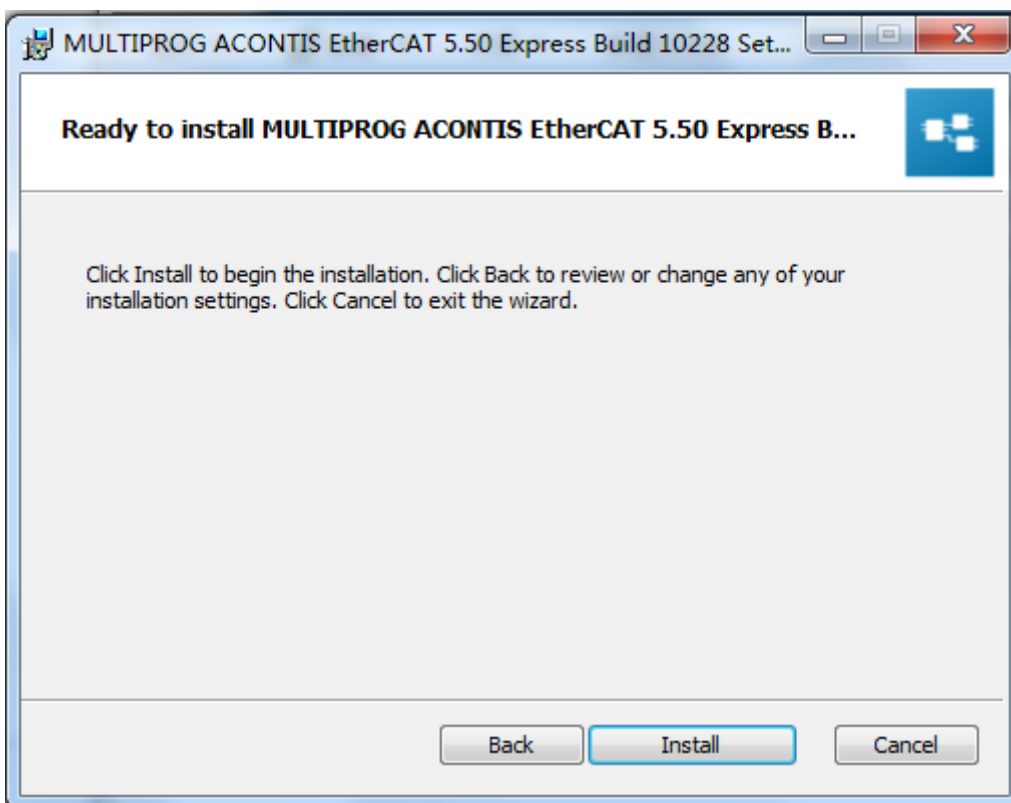


选择默认安装路径，勾选“I accept the terms in the License Agreement”后点击

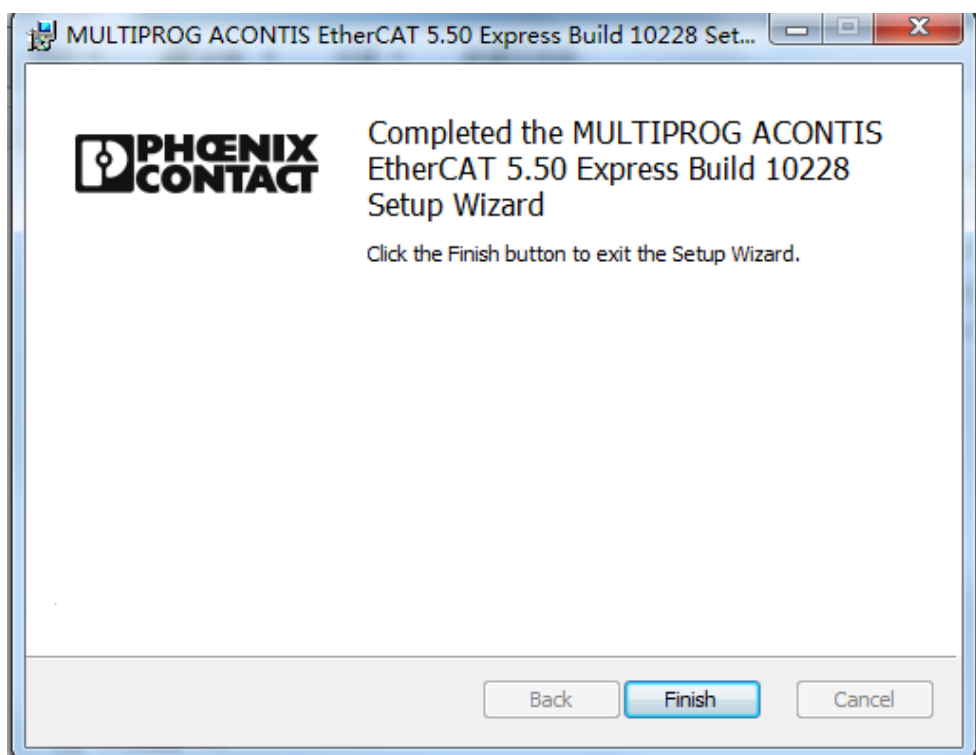
“Next”;



点击 “Install”



等待安装完成后点击 “Finish” 退出安装程序。



1.1.2 安装 2 号文件夹《2_eCLR Addon》

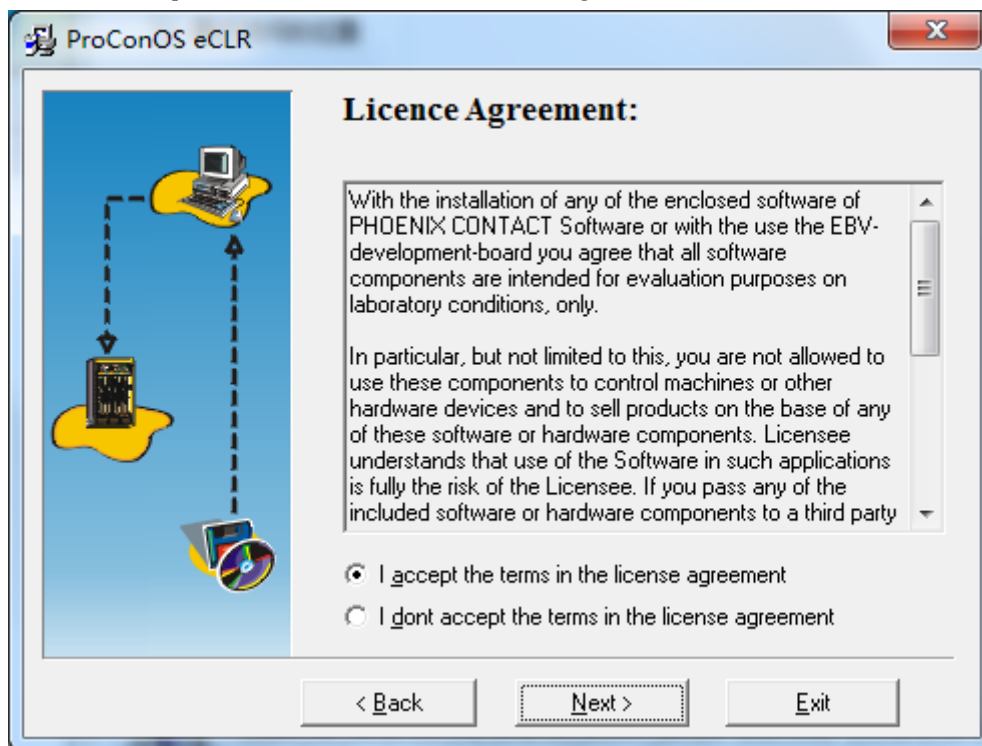
2 号文件夹内安装内容如下图所示：

名称	修改日期	类型
i486_LE_MSC12_ACONTIS_ECAT_eCLR	2018/6/26 13:53	应用程序

双击打开安装文件，出现下图所示安装窗口点击“Next”。



勾选 “I accept the terms in the License Agreement” 后点击 “Next”;

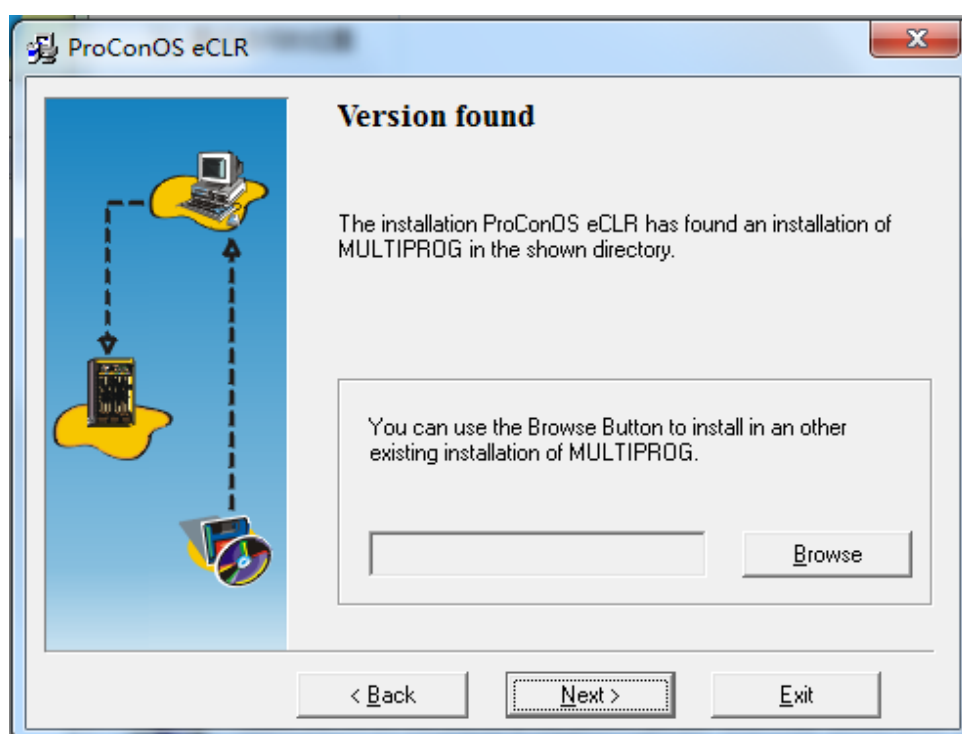


安装路径选择为:

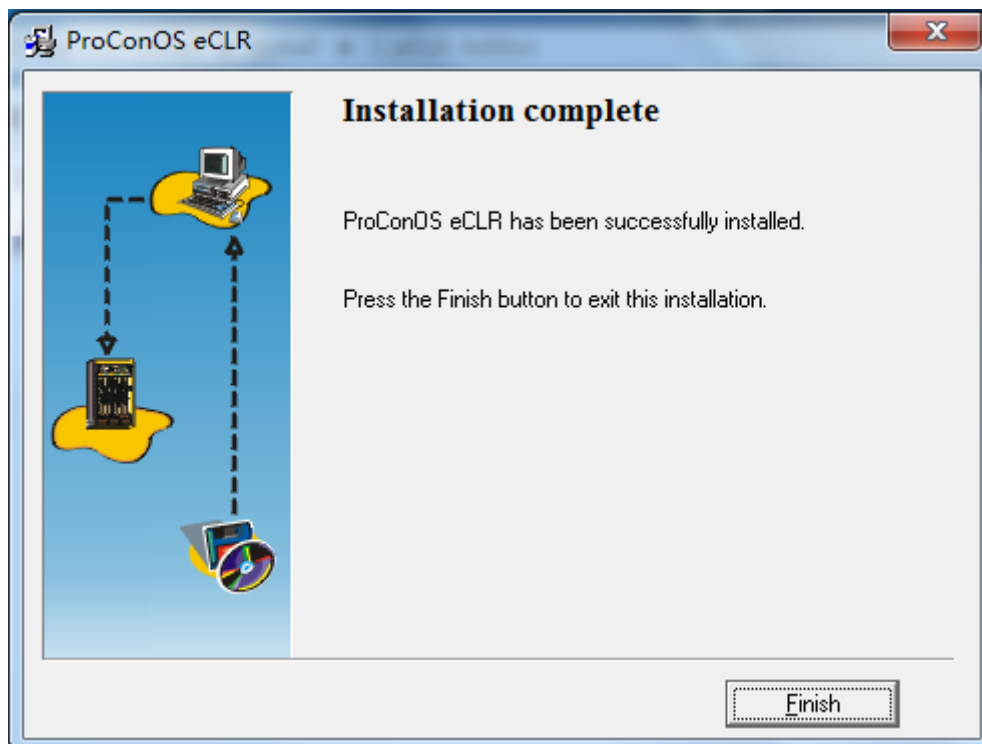
“C:\Program Files\PHOENIX CONTACT Software\MULTIPROG ACONTIS EtherCAT 5.50 Express Build 10228”

(注意: 选中 MULTIPROG ACONTIS EtherCAT 5.50 Express Build 10228 文件夹下的 plc 文件夹, 注意不要点开, 选中即可。)

选好安装路径后点击 “Next”, 等待安装完成。



安装完成后点击“Finish”退出安装程序。



1.1.3 安装 3 号文件夹《3_ECEngineer for MULTIPROG Express》

注意：若本机安装过“有狗”版本的 MULTIPROG，应先卸载本机上的 ECEngineer！

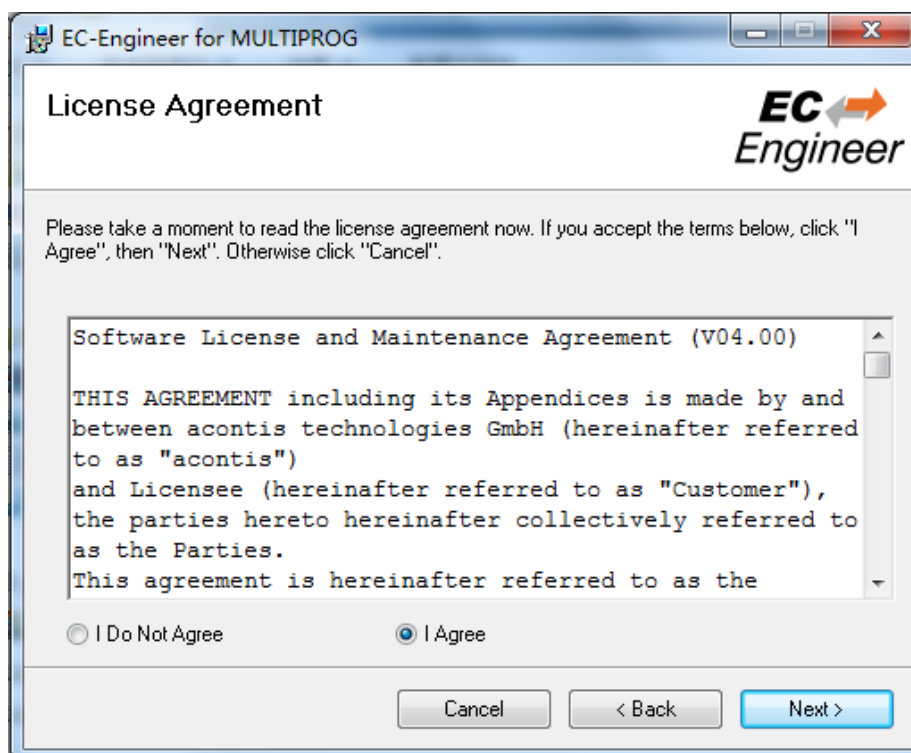
双击打开 3 号文件夹下的 ECEngineer 安装文件。

名称	修改日期	类型
 ECEngineer	2016/12/15 19:33	Windows Install...

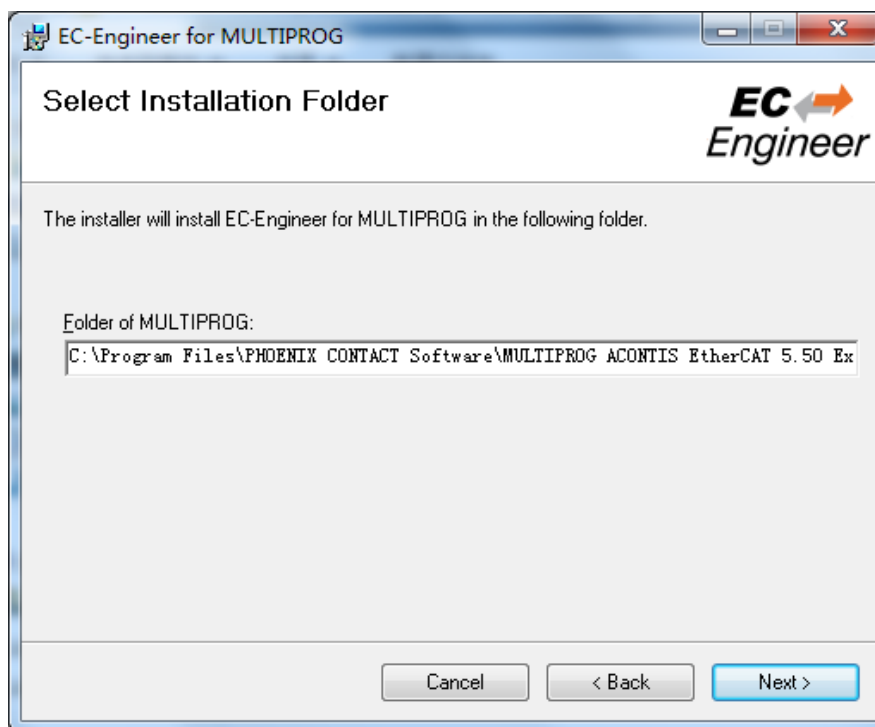
出现下图所示安装窗口后，点击“Next”；



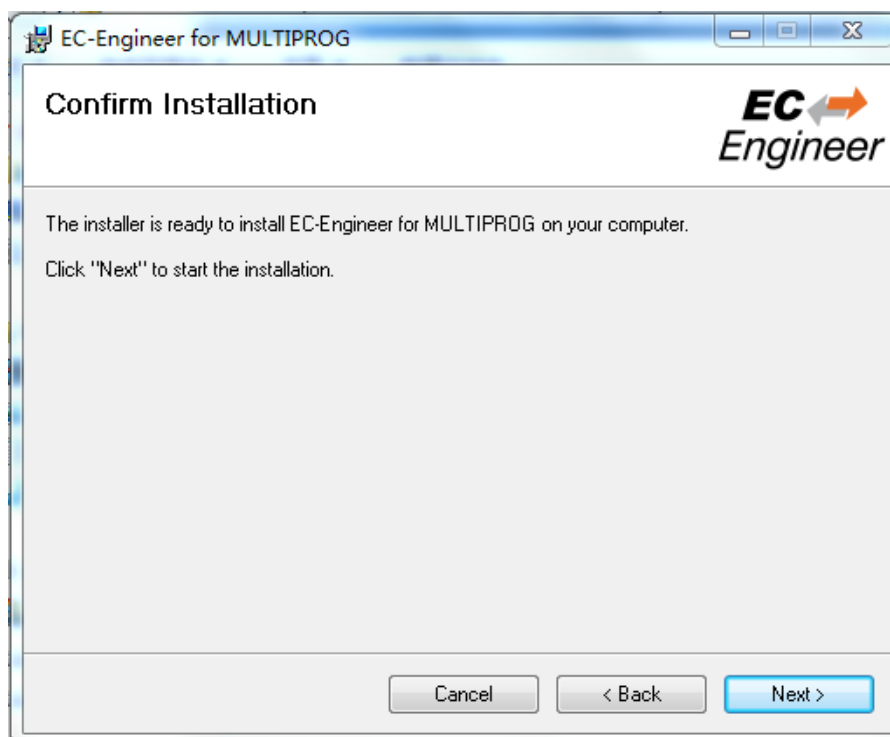
勾选 “I Agree” 后点击 “Next”;



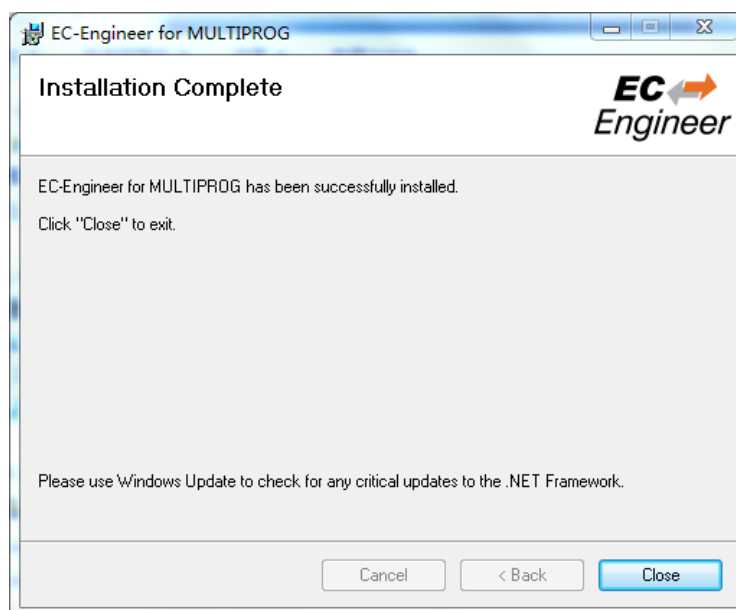
安装路径为默认安装路径即可，点击 “Next”;



点击“Next”安装，等待安装完成。



安装完成后点击“Close”退出安装程序。



至此完成 MULTIPROG Express 5.5 的安装。接下来需要按第二部分介绍配置环境。

1.2 配置环境

环境配置文件夹包括三个文件夹，用于配置环境。

名称	修改日期	类型
FW_LIB	2018/6/6 20:27	文件夹
I486_LE_MSC12_ACONTIS	2018/6/6 20:27	文件夹
Libraries	2018/6/6 20:27	文件夹

《FW_LIB》文件夹替换路径为：

C:\ProgramData\PHOENIX CONTACT Software\MULTIPROGExpress \5_50_10228\plc\
将路径内原有 FW_LIB 文件夹用此文件夹替换即可。

注意：ProgramData 为隐藏文件夹，需要设置显示隐藏的文件夹，若无法设置则直接输入路径。

《I486_LE_MSC12_ACONTIS》文件夹替换路径为：

C:\Program Files\PHOENIX CONTACT Software\MULTIPROG ACONTIS EtherCAT 5.50
Express Build 10228\plc\eCLR\
将路径下原有《I486_LE_MSC12_ACONTIS》文件夹替换即可。

《Libraries》文件夹配置路径为：

C:\Users\Public\Documents\MULTIPROG ACONTIS EtherCAT\Libraries

注意是将 Libraries 内的文件夹和文件全部粘贴到配置路径的“Libraries”文件夹内。

至此完成对软件环境的配置。

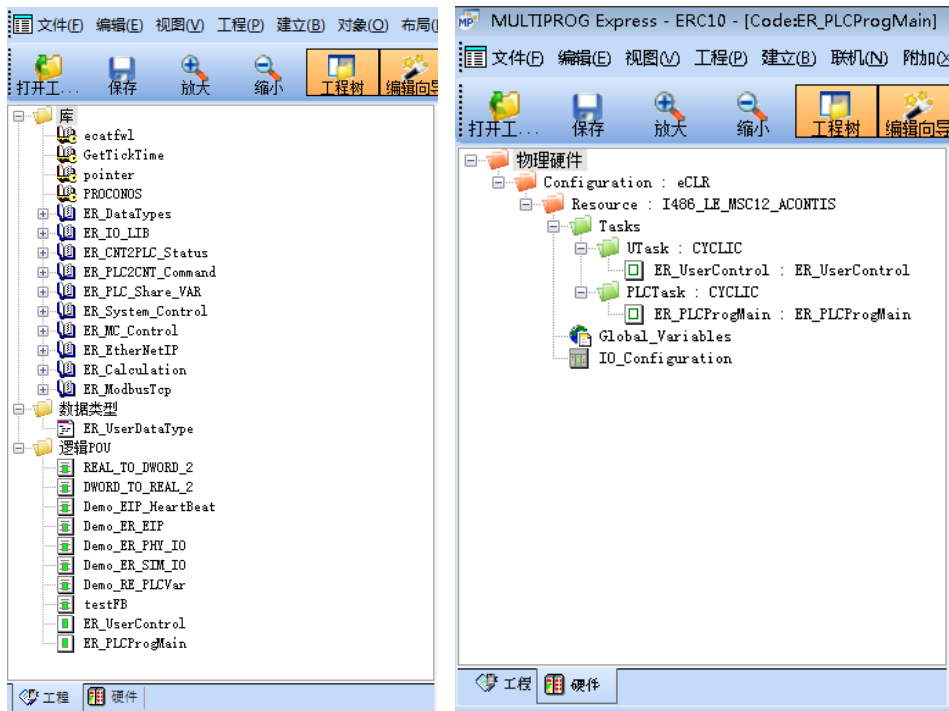
二、模板工程使用

用户需要用 PLC 进行相关开发时，需要在控制器研发团队提供的 Multiprog 标准工程上进行开发。当前 PLC 标准工程版本为 V0.0.26。

2.1 模板工程结构

用户功能开发必须基于此模板工程开发。须加载 Libraries 文件夹下的用户库 ER_DataTypes、ER_IO_LIB、ER_CNT2PLC_Status、ER_PLC2CNT_Command、ER_PLC_Share_Var、ER_System_Control、ER_MC_Control、ER_EtherNetIP、ER_Calculation、ER_ModbusTcp、ER_MC_GroupControl。

完整的基础工程结构图：



模板工程主要包含：

工程部分：

库 (Libraries)：可进行用户库、固件库添加或删除等操作。

数据类型 (Data Types)：用户自定义的数据类型（基于基本数据类型）。

逻辑 POU (Logical POU)：程序、功能块添加处。



硬件部分：

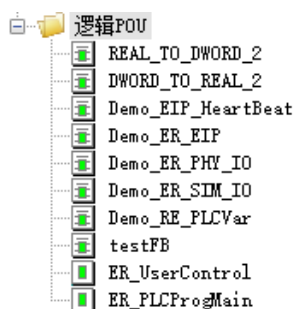
Tasks：可创建新任务，基础工程中已经创建了 2 个任务，2 个任务具体信息如下表：

Tasks(Task 类型)	任务用途	执行程序	周期(可否更改)	优先级
UTask(CYCLIC)	用于用户功能开发	ER_UserControl	10ms(可更改)	2
PLCTask(CYCLIC)	IO/PLC 共享变量控制任务	ER_PLCProgMain	4ms(不可更改)	1

Global_Variables: 工程中全局变量表。

2.2 逻辑 POU 简介

模板工程共有 10 个 POU，包括了 8 个功能块 ，2 个程序 。如下图：



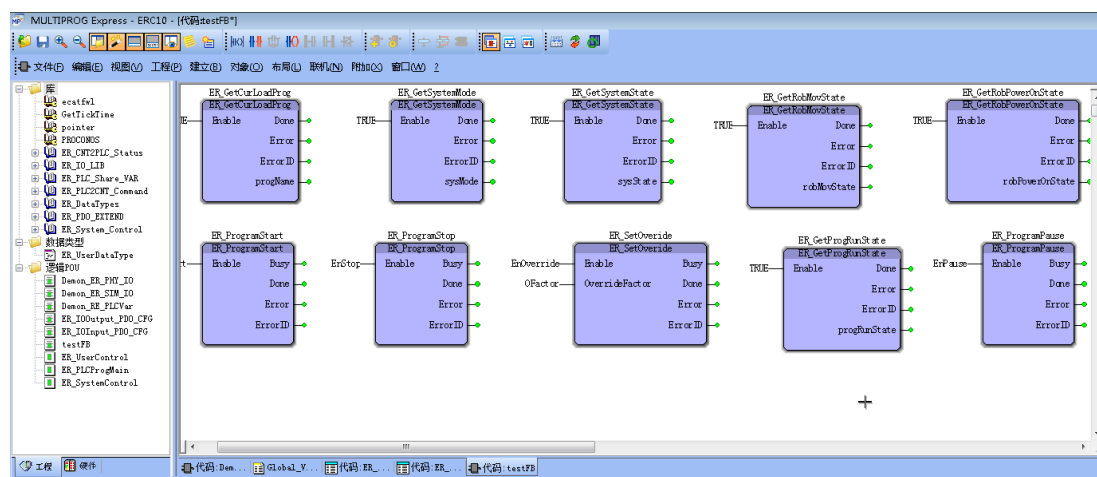
2.2.1 Demo 功能块

Demo_EIP_HeartBeat、Demo_ER_EIP、Demo_ER_PHY_IO、Demo_ER_SIM_IO、Demo_ER_PLCVar 为示例功能块，未被工程中任何程序调用，仅用作介绍相关功能块用法。

- 1、Demo_EIP_HeartBeat 为 EIP 通讯建立后，用于心跳计数的示例；
- 2、Demo_ER_EIP 介绍 EtherNet/IP 模块功能的使用方法；
- 3、Demo_ER_PHY_IO 介绍物理 IO 的用户使用方法；
- 4、Demo_ER_SIM_IO 介绍虚拟 IO 的用户使用方法；
- 5、Demo_ER_PLCVar 介绍 PLC 共享变量功能的使用方法；

2.2.2 testFB 功能块

此功能块可用于获取系统当前状态及简单控制命令下发。



此功能块内部调用的是用户库 ER_IO_LIB、ER_CNT2PLC_Status、ER_PLCCNT_Command 的功能块。具体每个小功能块含义与用法可以参考用户库内各个功能块的“FB/FU 帮助”信息，查看方法见章节 2.5。

2.2.3 ER_UserControl、ER_PLCCProgMain 程序

程序 ER_UserControl 为空工作单，用于用户功能的自行开发。

程序 ER_PLCCProgMain 用于刷新和同步物理 IO、虚拟 IO、PLC 共享变量数据等功能。

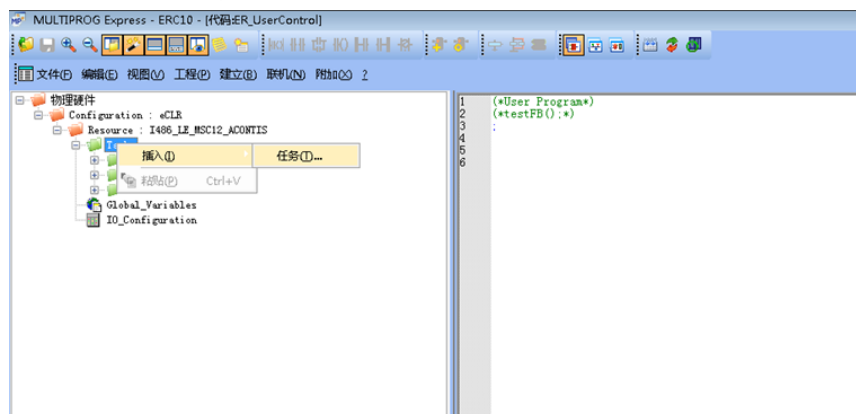
2.2.4 REAL_TO_DWORD、DWORD_TO_REAL 功能块

在使用系统自带的 REAL 与 DWORD 类型互转的功能块时，若发现小数点后精度丢失的问题，可使用这两个功能块解决。

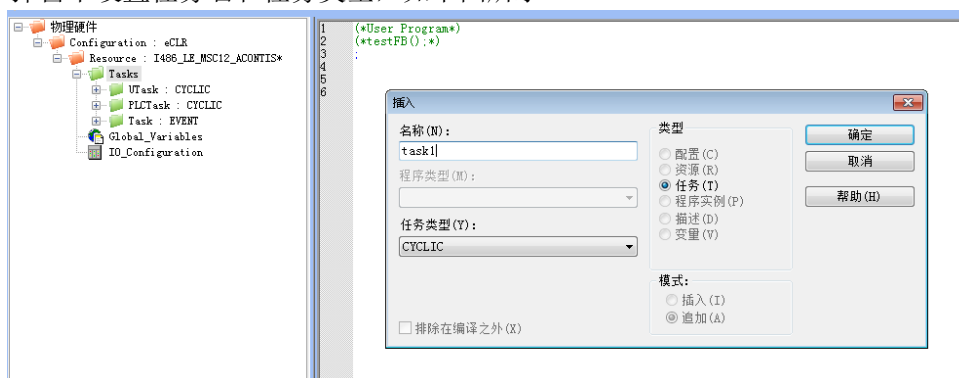
2.3 基于模板工程开发

2.3.1 添加任务 TASK

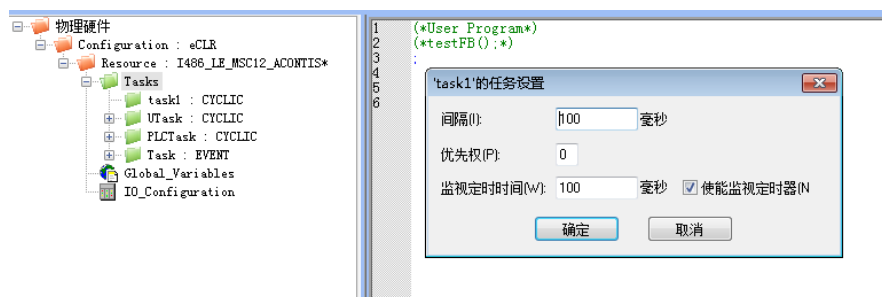
在 Task 上右键插入->任务，如图所示。



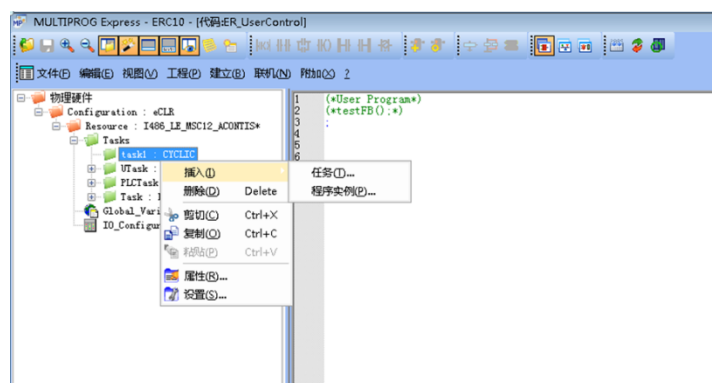
在弹窗中设置任务名和任务类型，如下图所示。



若选择 CYCLE 类型任务，将出现弹窗如下图：



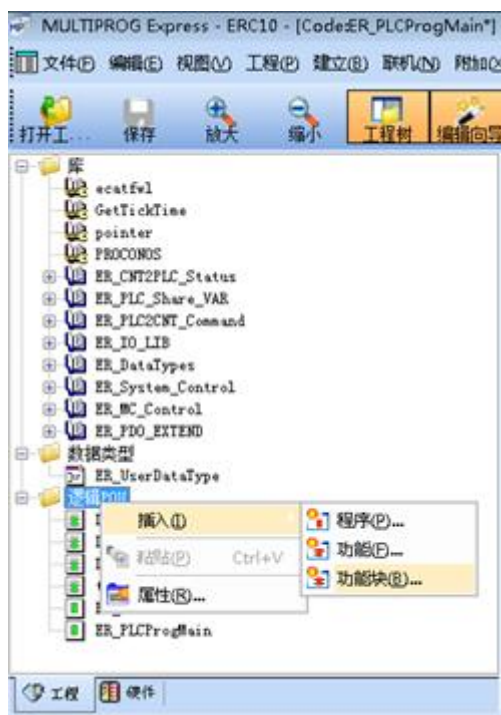
根据需求设置任务周期间隔、优先级等级、监视定时器设置等。点击确定即完成了任务的创建。创建任务后，根据需要在任务中添加任务或程序实例。



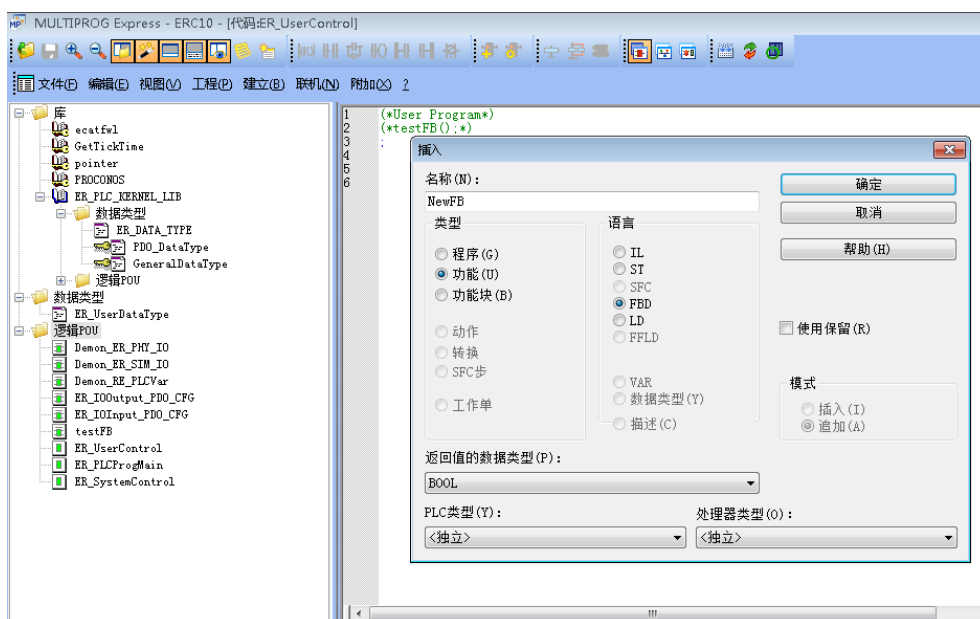
2.3.2 添加逻辑 POU

用户需要创建自己的功能、功能块或者程序时，都在逻辑 POU 内添加。

工程树窗口中右击鼠标，选择“插入”选项，根据需求选择所要创建的 POU 类型，如下图所示。



选择了 POU 类型后，在弹窗中设置该 POU 名称、类型、语言等并确认即可完成 POU 的添加。



2.3.3 下载调试

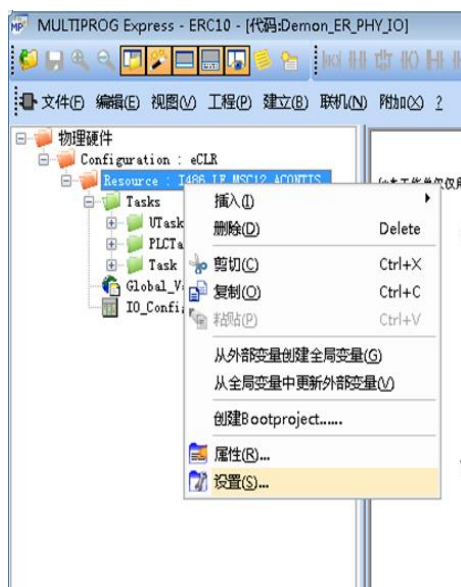
下载

使用网线与控制器的 CN2 网络口进行连接，直连或交换机连接。

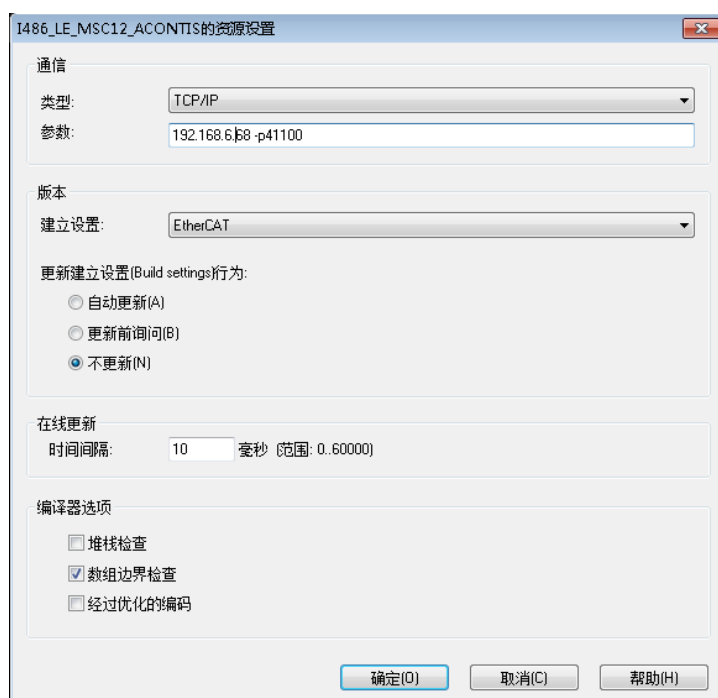
该 PLC 软件与控制器通过 TCP/IP 方式通信，进行程序的下载。如果控制器重启，需要等待控制器启动一段时间后，再去连接。

第一步：

设置 TCP/IP 的网络值，即控制器中 Intime 的网络 IP 值，右键 Resource: I486..._ECAT，选择设置。

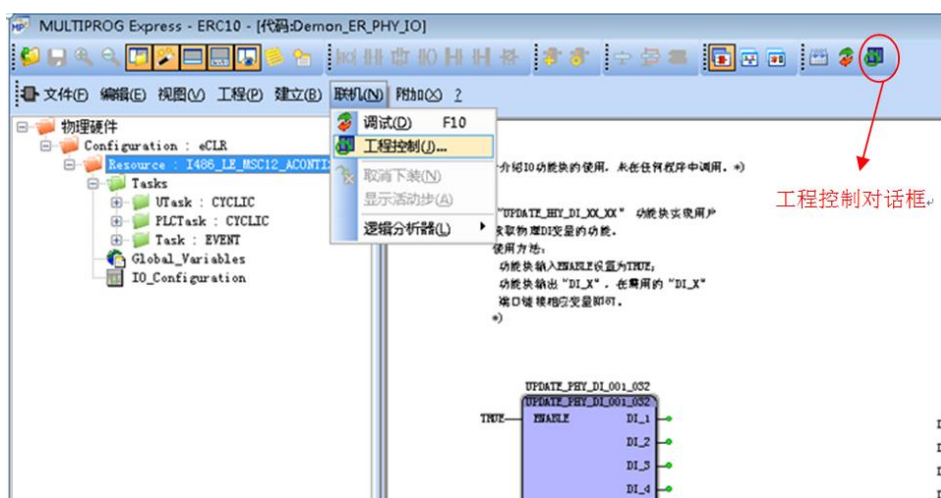


弹出如下设置框，如设置 IP 地址为：192.168.6.68，建立设置选择 RobotSimulation。

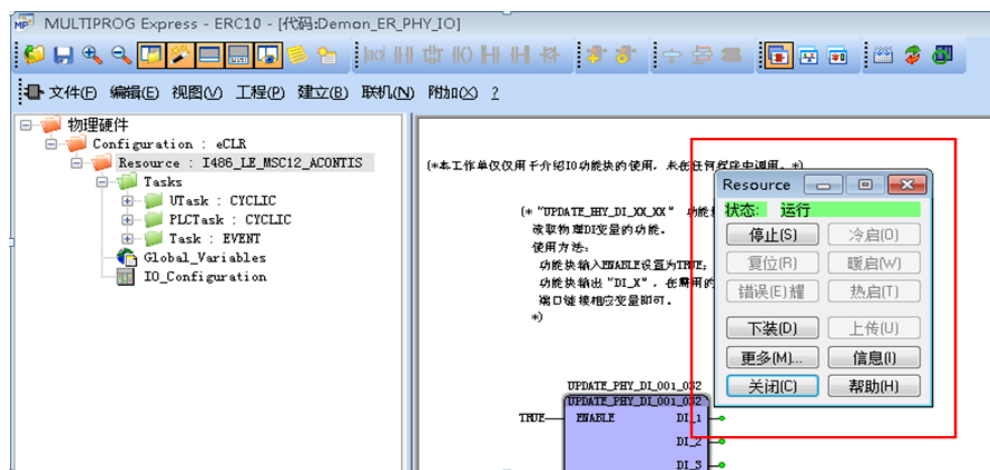


第二步：

点击菜单栏->联机下的工程控制，或者直接点击“工程控制对话框”按键

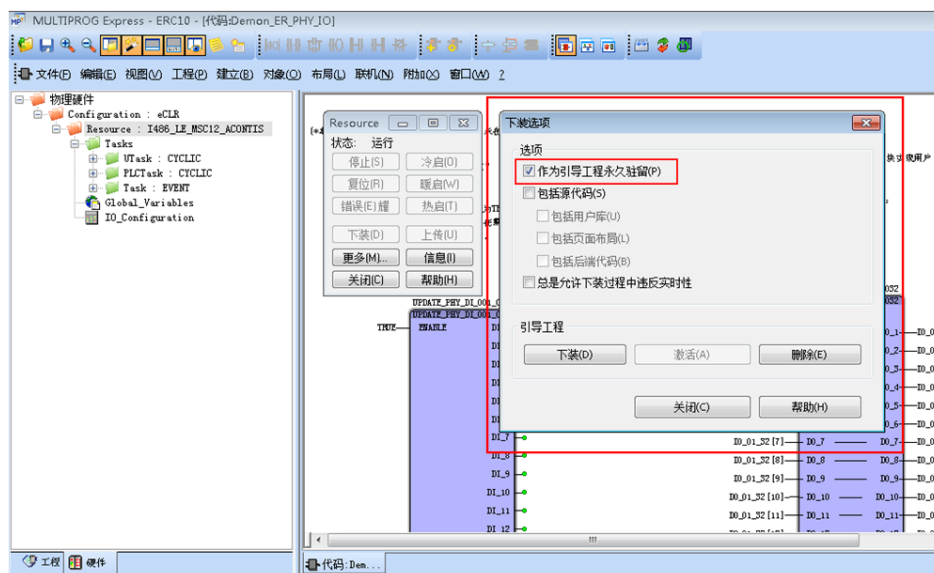


将会弹出如下窗口：



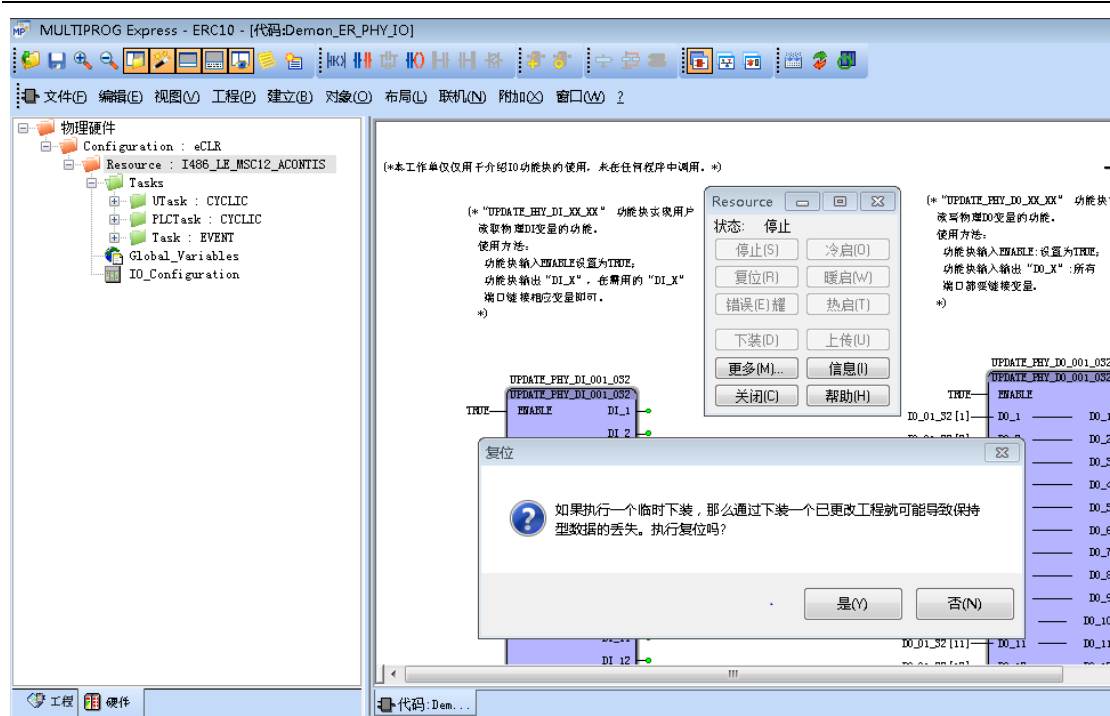
第三步

点击弹窗中“更多”按键，也会弹窗：如果第一项“作为引导工程永久驻留”没有勾选，勾选第一项。如下所示：

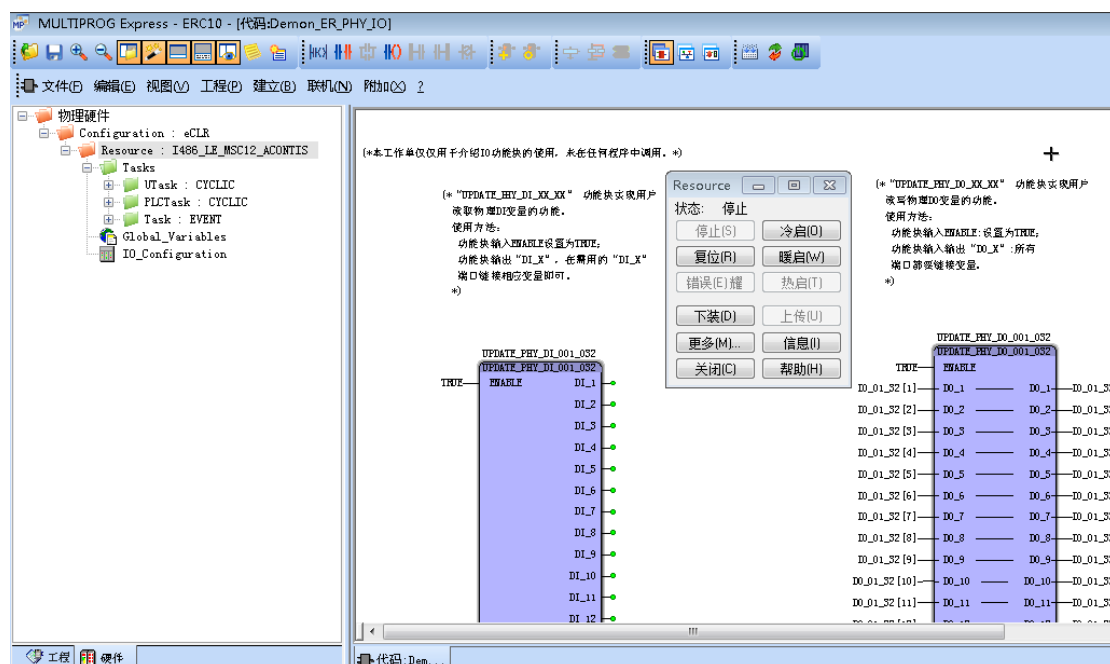


第四步：停止->复位->下装->冷启

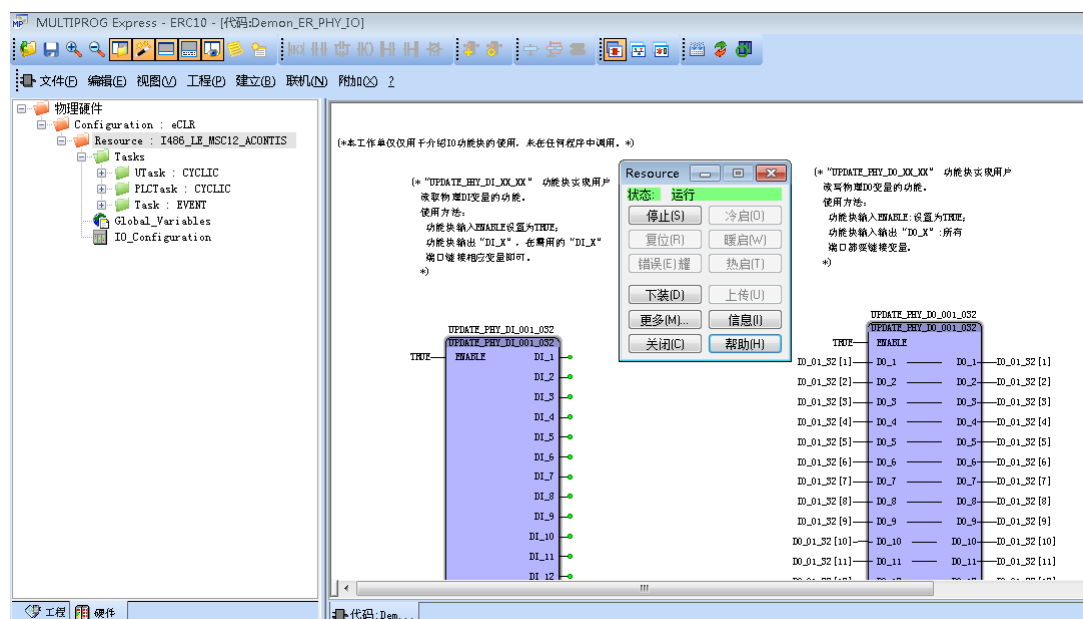
点击 Resource 弹窗中“停止”按钮，然后点击激活的“复位”按钮，并点击弹窗中的“是”，如下：



然后, 点击 Resource 弹窗中点击“下载”按钮, 等待进度条完成, 再点击“冷启”, 如下:



到此, 如果不报错, 显示如下, 即下载程序完成。

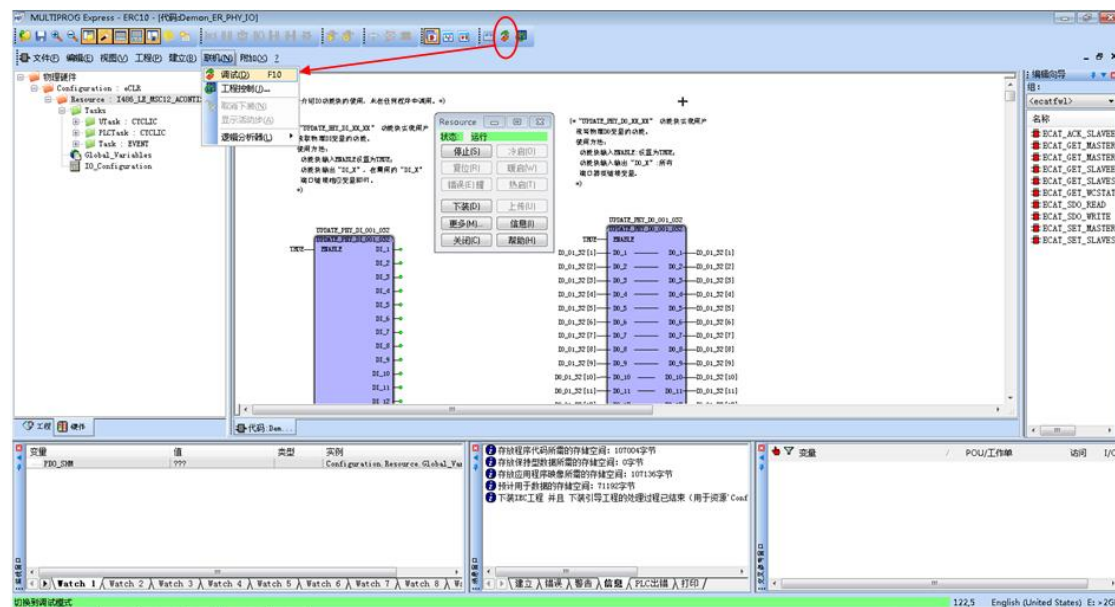


调试

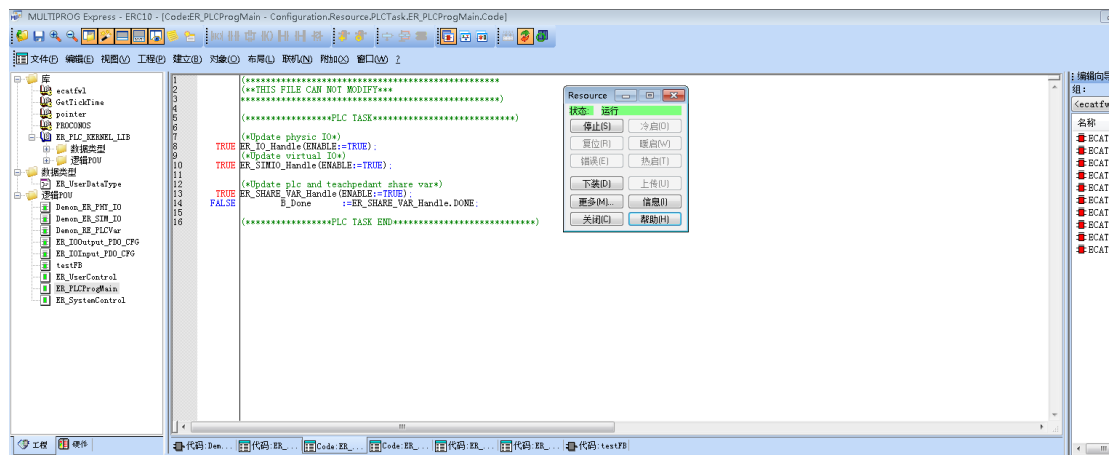
调试支持在线查看程序中变量的值，并且支持断点调试。

注意：调试前，保证已经与控制器建立了通信。如果没有建立，通过设置 IP 地址（如上节下载所示），建立连接。这里可以点击“工程控制对话框”按钮来确认是否建立了连接，弹窗显示状态为绿色运行即为建立了连接。

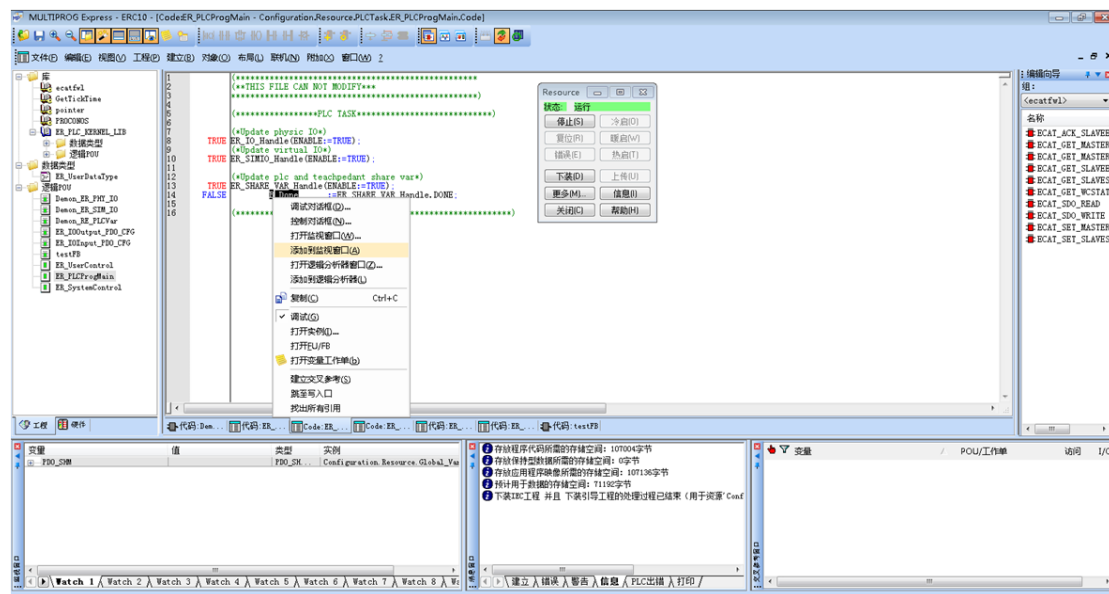
点击菜单栏->联机下的“调试”，或快捷菜单“调试”按钮如下所示：



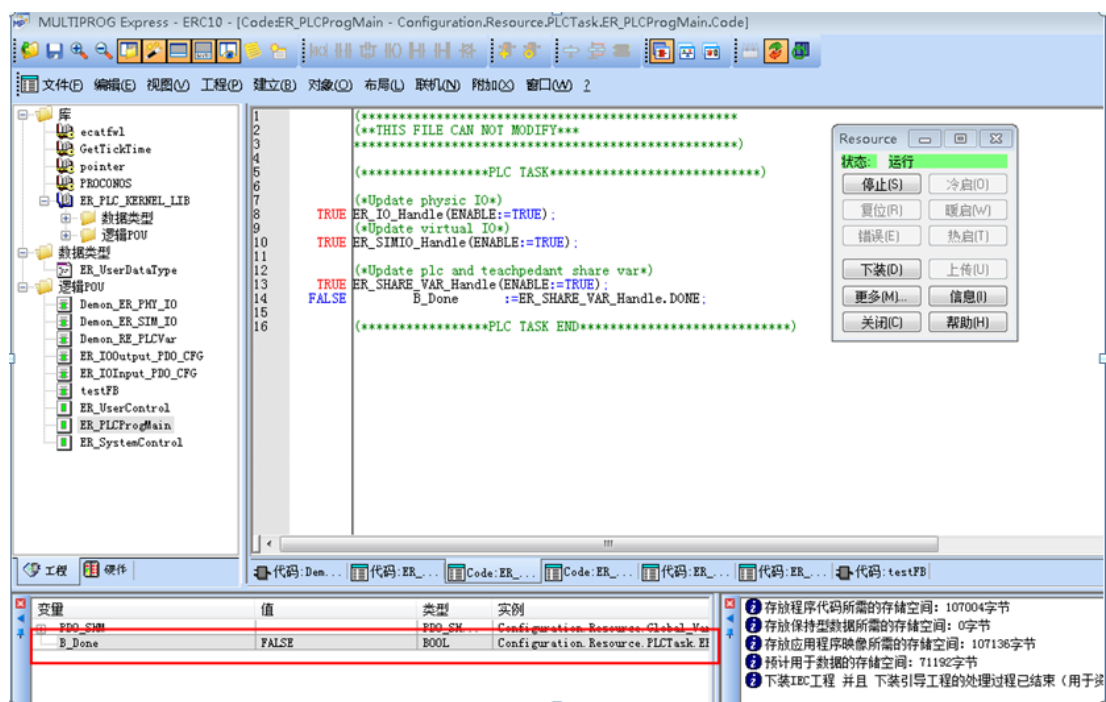
进入调试界面，此时仅可调试，不可编辑。如下：



可以将变量添加到监视窗口中，查看该变量的值。双击或选中变量名，点击右键，选择“添加到监视窗口”，如下：

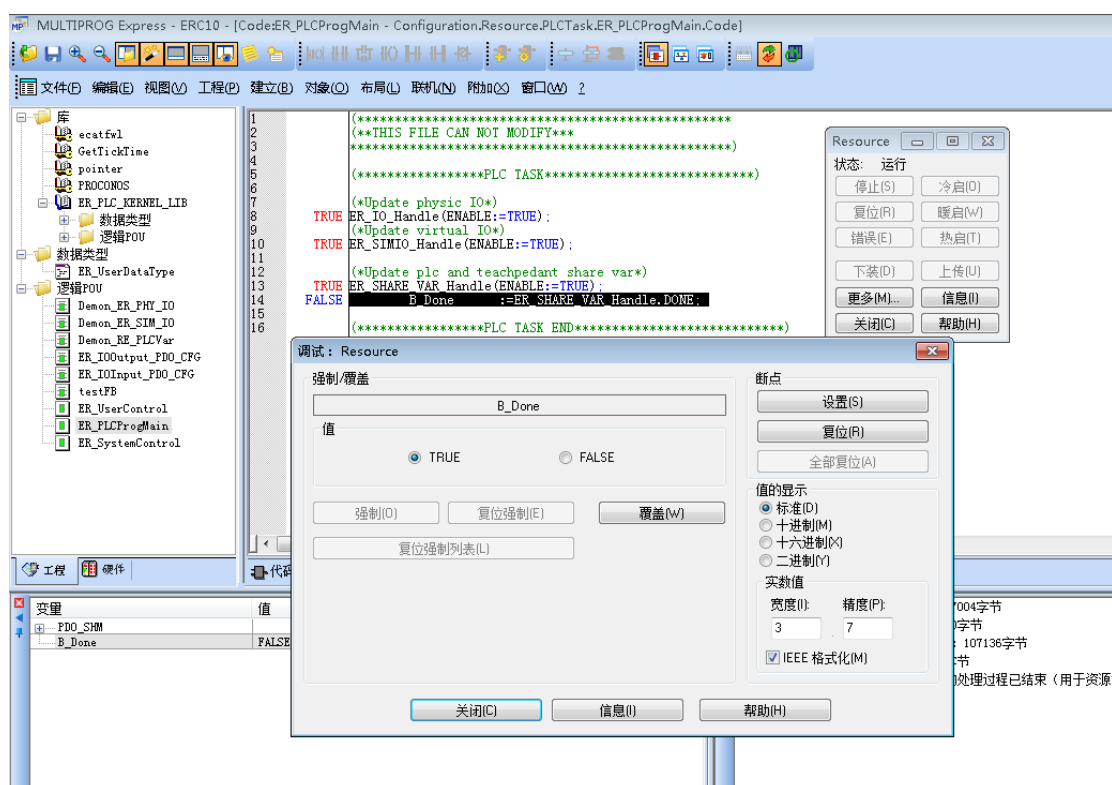


即可在“监视窗口”中，看到该变量的值。如下：



在线赋值: (慎用)

在所赋值变量处, 双击变量或者变量左侧 (竖线左侧), 即可弹出如下窗口:

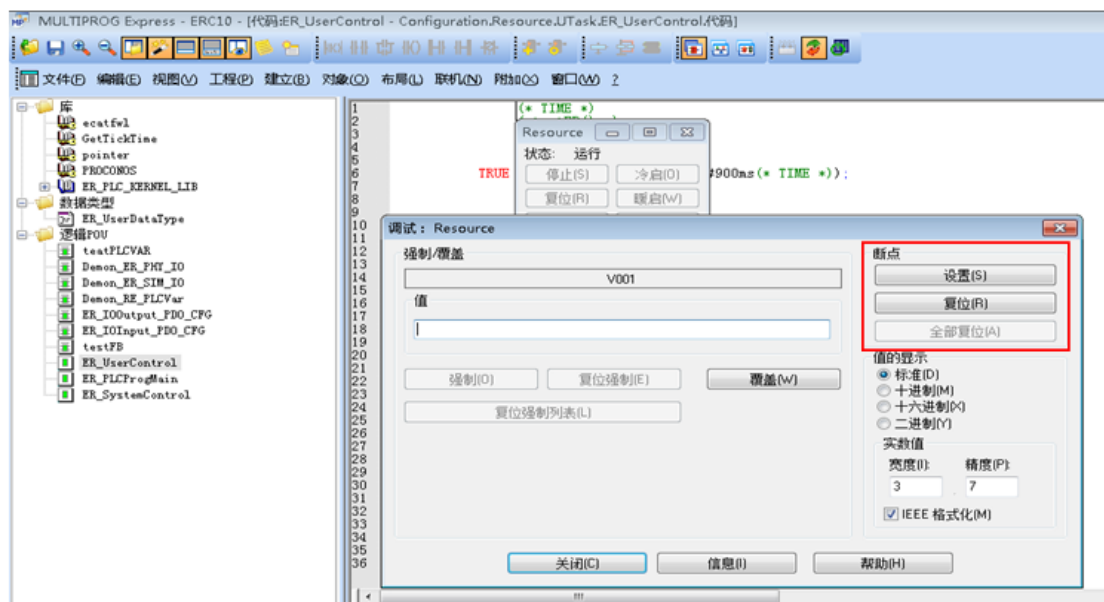


点击“覆盖”即可在线赋值。

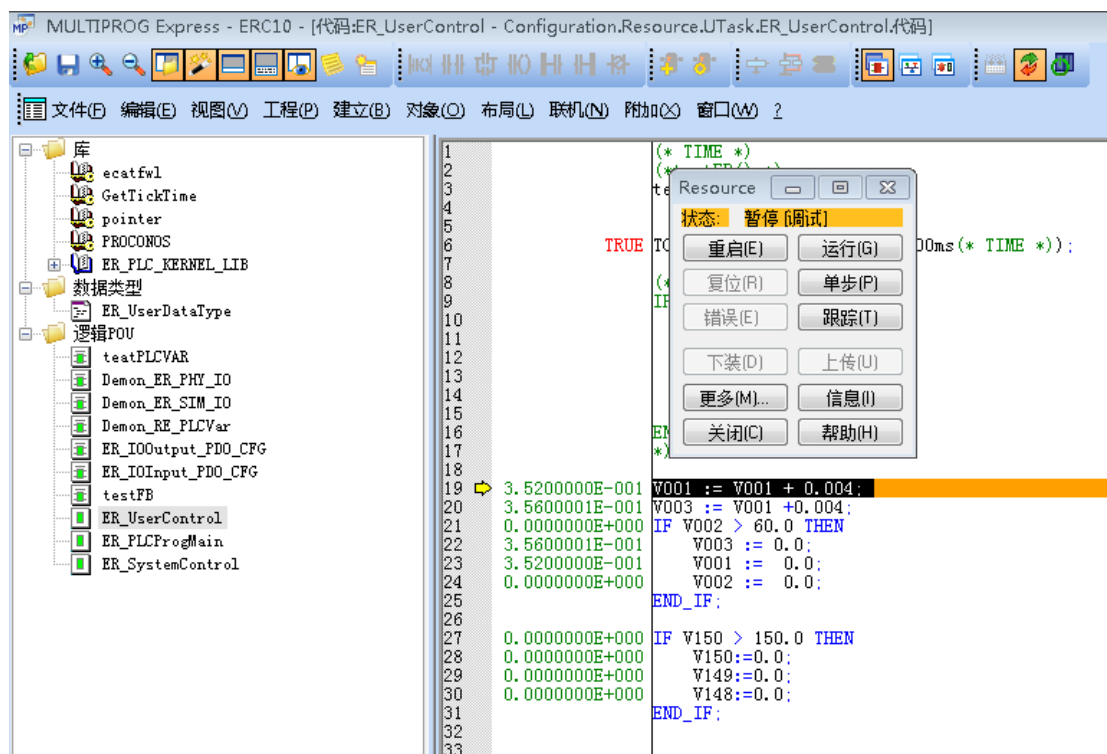
注意: 并非所有变量均可直接在线赋值。

断点设置调试:

在所要断点处，双击左侧（竖线左侧），即可弹出如赋值一样的弹窗，点击弹窗右侧“断点”的“设置”即可设置断点。具体效果，如下所示：



在选定行设置断点后，可以看到：**Resource** 弹窗变色，状态由运行转变为暂停模式，并且激活了 **Resource** 弹窗内的“单步”和“跟踪”等按钮，这里点击“单步”按钮即可单步执行程序，“跟踪”可用于进入调用的功能块或程序内。



取消断点时，同样双击断点处，点击弹窗断点的“复位”按钮。然后再点击 **Resource**

弹窗中的“运行”按钮即可。

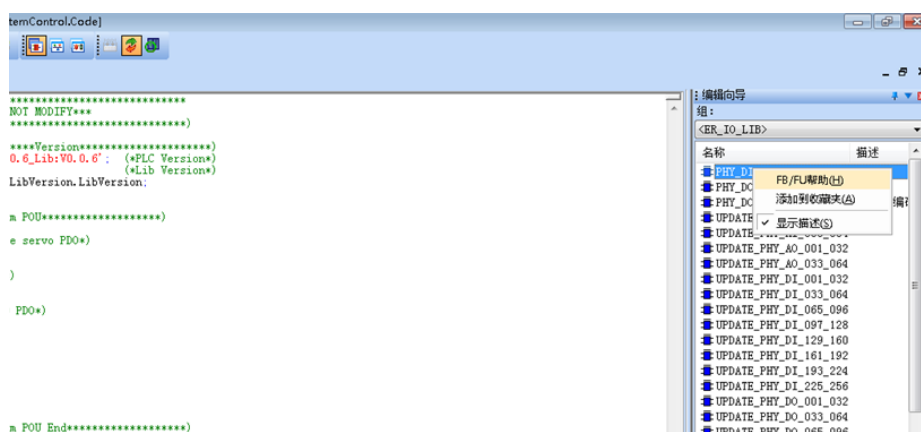
2.5 其他注意事项

软件使用过程中，应注意备份工程。若软件出现卡顿无响应，应选择等待程序响应，结束程序可能导致工程损坏。尽量避免使用过程中电脑断电情况，否则也可能导致工程损坏。

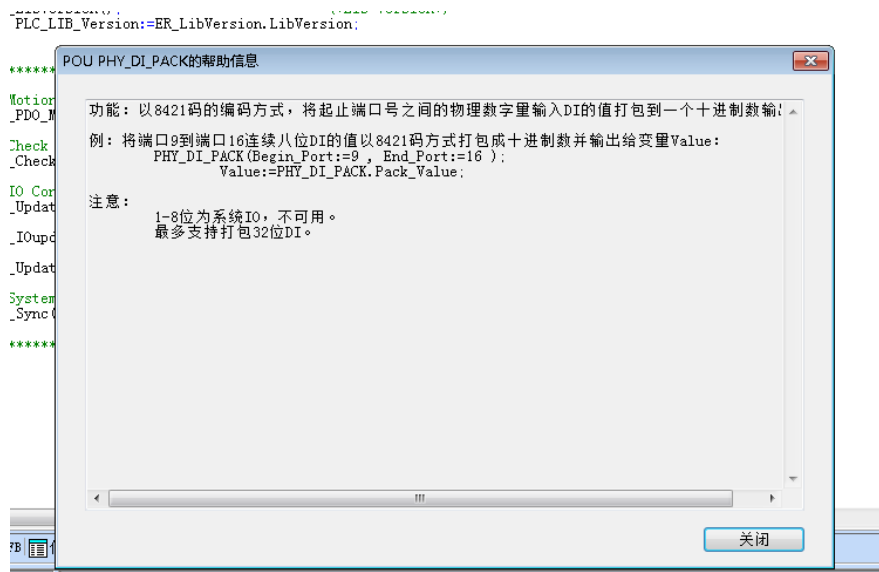
软件使用过程中，一些操作后软件没有体现的情况，关闭软件再打开，多数情况下可以解决。

替换用户库时，应先在工程中删除被替换的用户库，再到路径 C:\Users\Public\Documents\MULTIPROG ACONTIS EtherCAT\Libraries 下替换用户库文件（一个 MWT 文件，一个同名文件夹）。若先到该路径下替换文件，再在工程中删除该库，则可能替换不生效。

用户库功能块大部分有帮助信息，介绍功能块功能、用法及注意事项。查看方法：



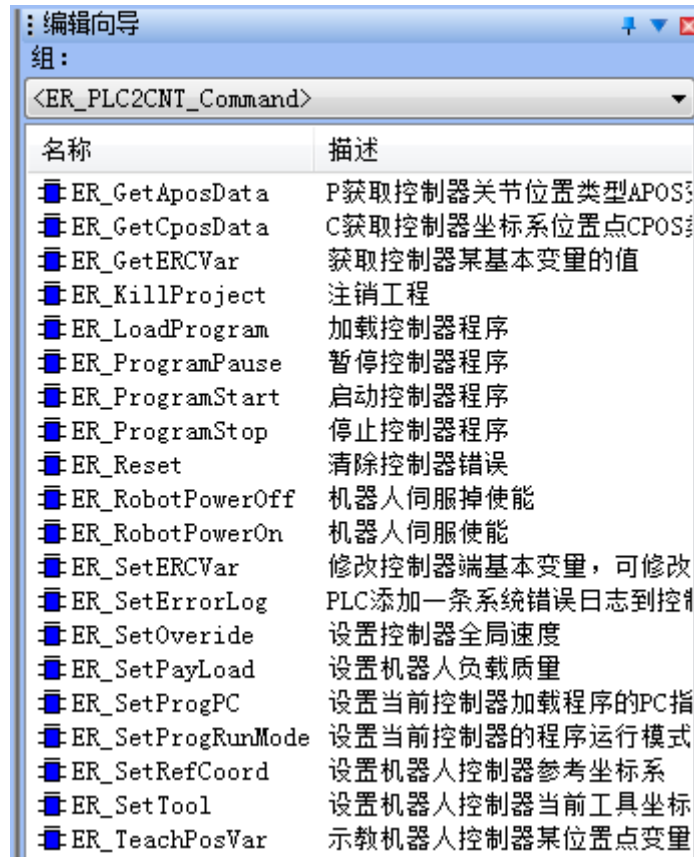
右侧编辑向导中，选中某用户库，选择需要了解的功能块右击鼠标，点击“FB/FU 帮助”即可弹出功能块说明信息如下图。



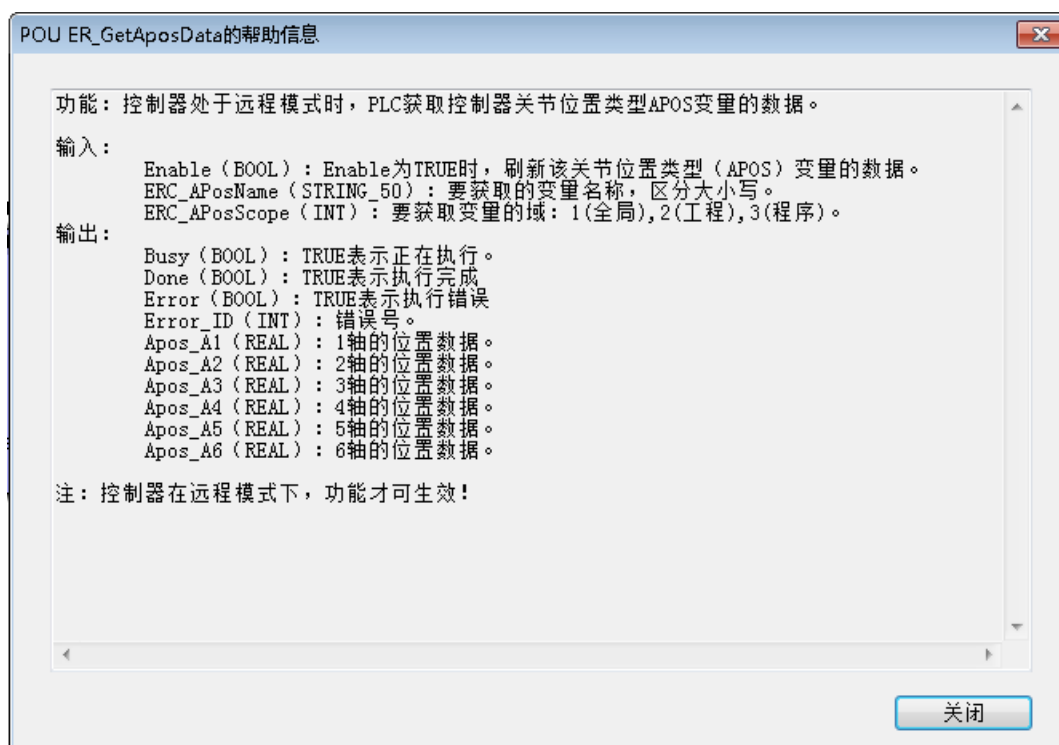
三、用户库功能块介绍

标准模板工程中，共加载了十一个用户库：ER_DataTypes、ER_IO_LIB、ER_CNT2PLC_Status、ER_PLC2CNT_Command、ER_PLC_Share_Var、ER_System_Control、ER_MC_Control、ER_EtherNetIP、ER_Calculation、ER_ModbusTcp、ER_MC_GroupControl。其中，ER_CNT2PLC_Status、ER_PLC2CNT_Command、ER_PLC_Share_Var、ER_IO_LIB、ER_MC_Control、ER_EtherNetIP、ER_Calculation、ER_ModbusTcp、ER_MC_GroupControl 为提供给用户使用的库。本章将介绍这些库中提供的功能块及其用法。

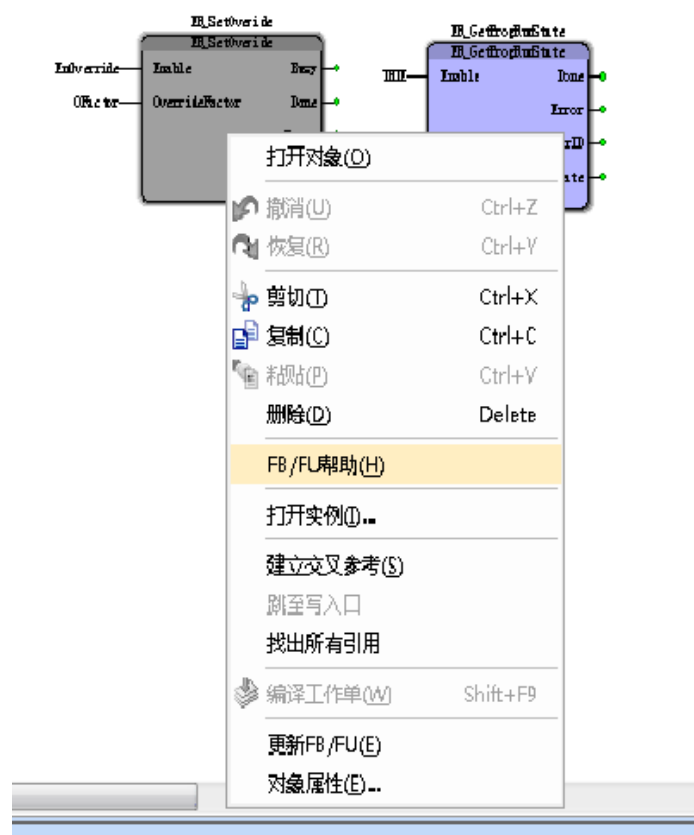
在使用 Multiprog 软件的过程中，这些用法说明可以通过如下方法获取到：在编辑向导界面，选择 ER_PLC2CNT_Command 用户库如下图：



以功能块 ER_GetAposData 为例，右击该功能块，选择“FB/FU 帮助”，即可显示该功能块帮助信息如下图所示：



或者在工作单中，右击功能块图（FDB）或者功能块名称（ST），点击“FB/FU 帮助”也可显示功能帮助信息。



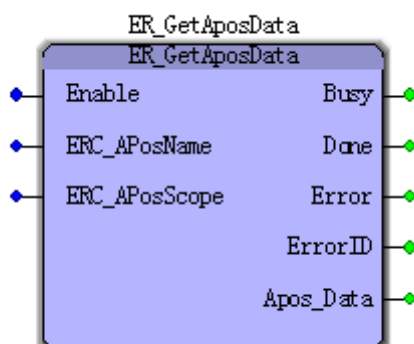
3.1 ER_PLC2CNT_Command: PLC 命令功能块

ER_PLC2CNT_Command 用户库提供了一系列用于控制控制器状态的功能块,具体开放的控制功能块见表 3.1。

表 3.1 PLC 控制命令接口统计

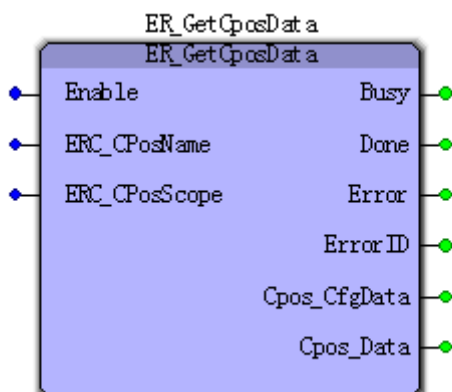
功能块名	实现功能
ER_GetAposData	用于 PLC 端获取控制器关节位置类型 APOS 变量数据。
ER_GetCposData	用于 PLC 端获取控制器坐标系位置类型 CPOS 变量数据。
ER_GetERCVar	用于 PLC 端获取控制器端某基本类型变量 (int、bool、real) 的值
ER_KillProject	用于 PLC 端注销控制器端工程
ER_LoadProgram	用于 PLC 端加载控制器端程序
ER_ProgramPause	用于 PLC 端暂停控制器端程序。
ER_ProgramStart	用于 PLC 端启动控制器端程序。
ER_ProgramStop	用于 PLC 端停止控制器端程序。
ER_Reset	用于 PLC 端清除控制器端错误报警。
ER_RobotPowerOff	用于 PLC 端使机器人掉伺服。
ER_RobotPowerOn	用于 PLC 端使机器人上伺服。
ER_SetERCVar	用于 PLC 端修改控制器端变量的值。
ER_SetErrorLog	用于 PLC 端给控制器端发送错误报警。
ER_SetOverride	用于 PLC 端设置全局速度。
ER_SetPayLoad	用于 PLC 端设置当前负载质量。
ER_SetProgPC	用于 PLC 端设置控制器当前加载程序的指针。
ER_SetProgRunMode	用于 PLC 端设置控制器的程序运行模式。
ER_SetRefCoord	用于 PLC 端设置控制器当前参考坐标系。
ER_SetTool	用于 PLC 端设置控制器当前工具坐标系。
ER_TeachPosVar	用于 PLC 端示教某一位置点变量。
ER_SetAutoLoadPrg	用于 PLC 端设置远程模式下自启动程序
ER_CancelAutoLoadPrg	用于 PLC 端取消某程序远程模式自动加载
ER_LoadAutoRunProg	用于 PLC 端加载设置为远程模式下自启动的程序
ER_SetAposData	用于 PLC 端修改 APOS 变量
ER_SetCposData	用于 PLC 端修改 CPOS 变量
ER_JogMotion	用于 PLC 端点动机器人
ER_RobotMoveDisable	用于 PLC 端设置机器人机械运动使能失效
ER_RobotMoveEnable	用于 PLC 端设置机器人机械运动使能
ER_SetRobJogCoord	用于 PLC 端设置当前点动坐标系类型
ER_SetRobotMode	用于 PLC 端设置机器人系统模式
ER_SetVirtualSafetyDoor	用于 PLC 端设置虚拟安全门停止方式,并停止机器人系统运行
ER_BackupRuntime	用于 PLC 端备份控制器中的 runtime 包,并拷贝到指定路径
ER_GetPosInBatch	用于 PLC 端批量获取名称连续的点位变量数据
ER_SetJogMicroParam	用于 PLC 端设置寸动的线性步长与关节步长
ER_JogMicroMotion	用于 PLC 端寸动机器人
ER_ExtSignalShielding	用于 PLC 端开启和关闭外部信号屏蔽功能
ER_StartBrakeTest	用于 PLC 端开启制动测试

3.1.1 ER_GetAposData



- 功能：控制器处于远程模式时，PLC 获取控制器关节位置类型 APOS 变量的数据。
 - 输入参数：
 - Enable (BOOL)：Enable 为上升沿获取该关节位置类型 (APOS) 变量的数据。
 - ERC_APosName (STRING_50)：要获取的变量名称，区分大小写。
 - ERC_APosScope (INT)：要获取变量的域：1(全局),2(工程),3(程序)。
 - 输出参数：
 - Busy (BOOL)：TRUE 表示正在执行。
 - Done (BOOL)：TRUE 表示执行完成
 - Error (BOOL)：TRUE 表示执行错误
 - Error_ID (INT)：执行成功时输出 0，执行出错时输出错误号。
 - Apos_Data(APOS_DATA)：APOS_DATA 数据类型是下标为 0 到 15 的 REAL 型数组。
 - 依次代表机器人不同轴的位置 (本体轴和附加轴)。
- 注：控制器在远程模式下，功能才可生效！
- Enable 上升沿获取变量数据。

3.1.2 ER_GetCposData



- 功能：控制器处于远程模式时，PLC 获取控制器坐标系位置点（CPOS）类型变量的数据。
- 输入参数：
 - Enable (BOOL)：Enable 为上升沿获取该坐标系位置类型（CPOS）变量的数据。
 - ERC_CPosName (STRING_50)：要获取的变量名称，区分大小写。
 - ERC_CPosScope (INT)：要获取变量的域：1(全局),2(工程),3(程序)。
- 输出参数：
 - Busy (BOOL)：TRUE 表示正在执行。
 - Done (BOOL)：TRUE 表示执行完成。
 - Error (BOOL)：TRUE 表示执行错误。
 - Error_ID (INT)：执行成功时输出 0，执行出错时输出错误号。
 - Cpos_CfgData (CPOS_CFG)：CPOS_CFG 数据类型为下标为 0 到 6 的七位 DINT 型数组，每一位对应含义如下：
 - Cpos_CfgData[0]:对应控制器“Cpos”变量“confdata”的 mode;
 - Cpos_CfgData[1]:对应控制器“Cpos”变量“confdata”的 cf1;
 - Cpos_CfgData[2]:对应控制器“Cpos”变量“confdata”的 cf2;
 - Cpos_CfgData[3]:对应控制器“Cpos”变量“confdata”的 cf3;
 - Cpos_CfgData[4]:对应控制器“Cpos”变量“confdata”的 cf4;
 - Cpos_CfgData[5]:对应控制器“Cpos”变量“confdata”的 cf5;
 - Cpos_CfgData[6]:对应控制器“Cpos”变量“confdata”的 cf6。
 - Cpos_Data (CPOS_DATA)：CPOS_DATA 数据类型为下标为 0 到 15 的十六位 REAL 型数组，每一位对应含义如下：
 - Cpos_Data[0]:对应控制器“Cpos”变量“x”;
 - Cpos_Data[1]:对应控制器“Cpos”变量“y”;
 - Cpos_Data[2]:对应控制器“Cpos”变量“z”;
 - Cpos_Data[3]:对应控制器“Cpos”变量“a”;
 - Cpos_Data[4]:对应控制器“Cpos”变量“b”;

Cpos_Data[5]:对应控制器“Cpos”变量“c”;
 Cpos_Data[6]:对应控制器“Cpos”变量“a7”，表示第 1 个附加轴位置。
 Cpos_Data[7]:对应控制器“Cpos”变量“a8”，表示第 2 个附加轴位置。

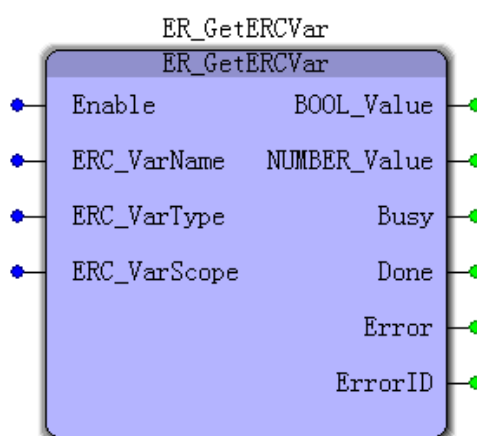
...

Cpos_Data[15]:对应控制器“Cpos”变量“a16”，表示第 16 个附加轴位置。

注：控制器在远程模式下，功能才可生效！

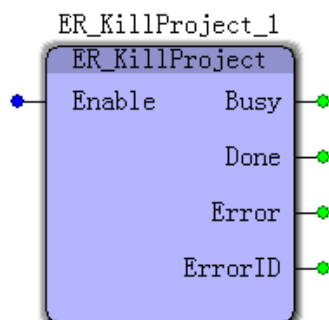
Enable 上升沿获取变量数据。

3.1.3 ER_GetERCVar



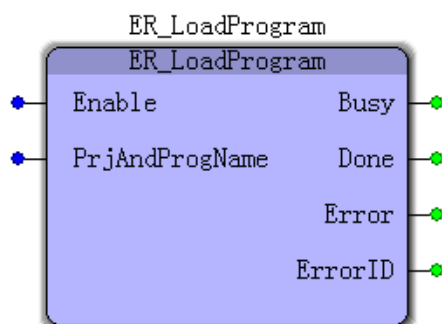
- 功能：控制器处于远程模式时，PLC 获取控制器某基本变量的值，仅支持 BOOL，INT，REAL 三种基本数据类型。
- 输入参数：
 - Enable (BOOL)：Enable 为 TRUE 时，刷新该坐标系位置类型 (CPOS) 变量的数据。
 - ERC_VarName (STRING_50)：要获取的变量名称，区分大小写。
 - ERC_VarType (INT)：要获取的变量类型：0(BOOL),1(INT),2(REAL)。
 - ERC_VarScope (INT)：要获取变量的域：1(全局),2(工程),3(程序)。
- 输出参数：
 - BOOL_Value (BOOL)：获取 BOOL 类型变量时，变量值从此输出。
 - NUMBER_Value (REAL)：获取 INT 或 REAL 变量时，变量值从此输出。
 - Busy (BOOL)：TRUE 表示正在执行。
 - Done (BOOL)：TRUE 表示执行完成
 - Error (BOOL)：TRUE 表示执行错误
 - Error_ID (INT)：执行成功时输出 0，执行出错时输出错误号。
- 注：控制器在远程模式下，功能才可生效！

3.1.4 ER_KillProject



- 功能： 控制器处于远程模式时，PLC 端注销控制器当前加载的工程。
- 输入参数：
 - Enable (BOOL)： 执行注销工程操作的使能信号，上升沿有效。
- 输出参数：
 - Busy (BOOL)： TRUE 表示正在执行。
 - Done (BOOL)： TRUE 表示执行完成
 - Error (BOOL)： TRUE 表示执行错误
 - Error_ID (INT)： 执行成功时输出 0，执行出错时输出错误号。
- 注： 控制器在远程模式下，功能才可生效！较上一版本，去掉了工程名的输入，直接注销当前加载的工程。

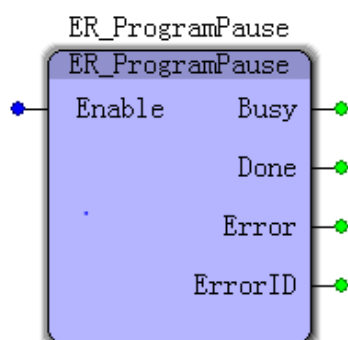
3.1.5 ER_LoadProgram



- 功能： 控制器处于远程模式时，加载控制器程序。
- 输入参数：
 - Enable (BOOL)： 执行加载程序操作的使能信号，上升沿有效。
 - PrjAndProgName (STRING_50)： 要加载的程序名，形式为“工程名.程序名”。

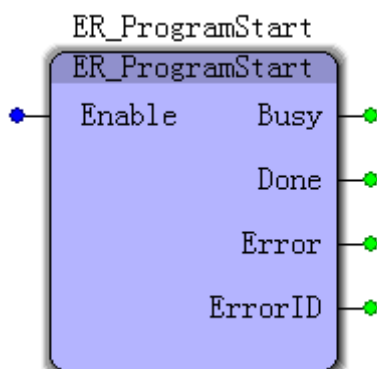
- 输出参数：
 - Busy (BOOL): TRUE 表示正在执行。
 - Done (BOOL): TRUE 表示执行完成
 - Error (BOOL): TRUE 表示执行错误
 - Error_ID (INT): 执行成功时输出 0, 执行出错时输出错误号。
- 注: 控制器在远程模式下, 功能才可生效!

3.1.6 ER_ProgramPause



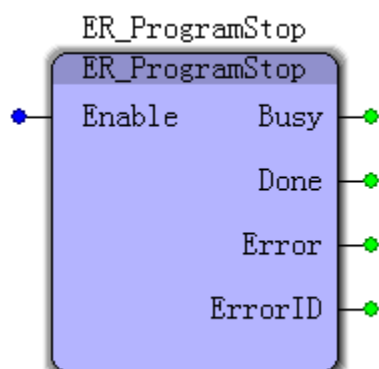
- 功能: 控制器处于远程模式时, 暂停运行程序。
- 输入参数:
 - Enable (BOOL): 执行暂停程序操作的使能信号, 上升沿有效。
- 输出参数:
 - Busy (BOOL): TRUE 表示正在执行。
 - Done (BOOL): TRUE 表示执行完成
 - Error (BOOL): TRUE 表示执行错误
 - Error_ID (INT): 执行成功时输出 0, 执行出错时输出错误号。
- 注: 控制器在远程模式下, 功能才可生效!

3.1.7 ER_ProgramStart



- 功能： 控制器处于远程模式时， 启动运行程序。
- 输入参数：
 - Enable (BOOL)： 执行启动程序操作的使能信号， 上升沿有效。
- 输出参数：
 - Busy (BOOL)： TRUE 表示正在执行。
 - Done (BOOL)： TRUE 表示执行完成
 - Error (BOOL)： TRUE 表示执行错误
 - Error_ID (INT)： 执行成功时输出 0， 执行出错时输出错误号。
- 注： 控制器在远程模式下， 功能才可生效！

3.1.8 ER_ProgramStop

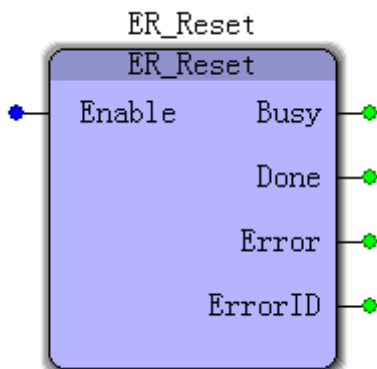


- 功能： 控制器处于远程模式时， 停止运行程序。
- 输入参数：
 - Enable (BOOL)： 执行停止程序操作的使能信号， 上升沿有效。
- 输出参数：

- Busy (BOOL): TRUE 表示正在执行。
- Done (BOOL): TRUE 表示执行完成
- Error (BOOL): TRUE 表示执行错误
- Error_ID (INT): 执行成功时输出 0, 执行出错时输出错误号。

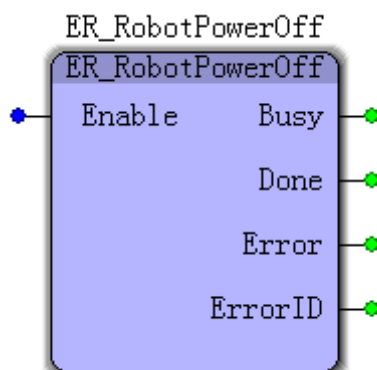
- 注: 控制器在远程模式下, 功能才可生效!

3.1.9 ER_Reset



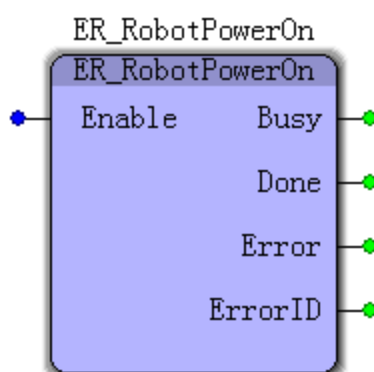
- 功能: 控制器处于远程模式时, 清除控制器错误报警。
- 输入参数:
 - Enable (BOOL): 执行清错操作的使能信号, 上升沿有效。
- 输出参数:
 - Busy (BOOL): TRUE 表示正在执行。
 - Done (BOOL): TRUE 表示执行完成
 - Error (BOOL): TRUE 表示执行错误
 - Error_ID (INT): 执行成功时输出 0, 执行出错时输出错误号。
- 注: 控制器在非远程模式下, 功能也可生效!

3.1.10 ER_RobotPowerOff



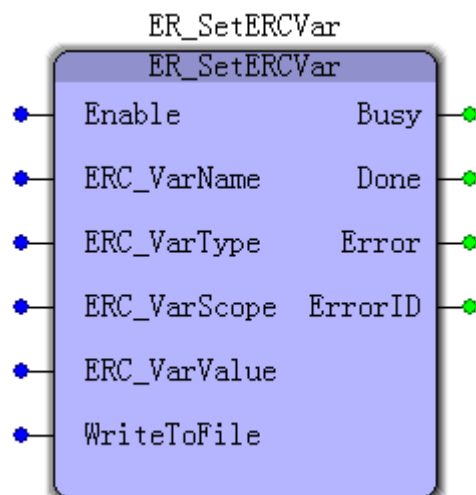
- 功能： 控制机器人掉伺服励磁。
- 输入参数：
Enable (BOOL): 执行下伺服操作的使能信号，上升沿有效。
- 输出参数：
Busy (BOOL): TRUE 表示正在执行。
Done (BOOL): TRUE 表示执行完成
Error (BOOL): TRUE 表示执行错误
Error_ID (INT): 执行成功时输出 0，执行出错时输出错误号。
- 注： 非远程模式下，功能也可生效！

3.1.11 ER_RobotPowerOn



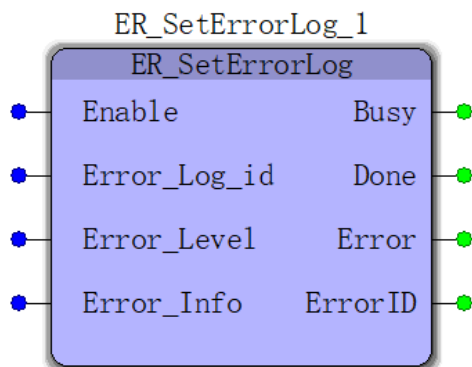
- 功能： 控制器处于远程模式时，使机器人上伺服。
- 输入参数：
Enable (BOOL): 执行上伺服操作的使能信号，上升沿有效。
- 输出参数：
Busy (BOOL): TRUE 表示正在执行。
Done (BOOL): TRUE 表示执行完成
Error (BOOL): TRUE 表示执行错误
Error_ID (INT): 执行成功时输出 0，执行出错时输出错误号。
- 注： 控制器在远程模式下，功能才可生效！（远程模式系统默认上励磁）

3.1.12 ER_SetERCVar



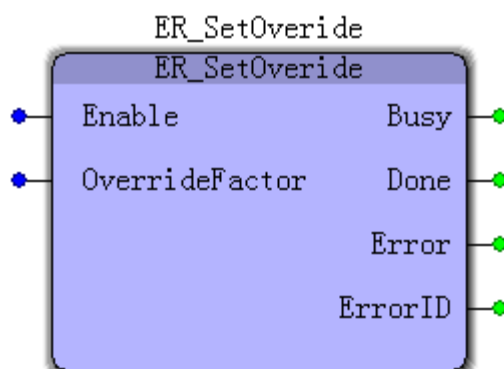
- 功能：控制器处于远程模式下，修改控制器端基本变量的变量值，可修改控制器端 BOOL、INT、REAL 三种类型。
- 输入参数：
 - Enable (BOOL)：操作使能信号，上升沿有效。
 - ERC_VarName (STRING_50)：要修改的控制器端变量的变量名，区分大小写。
 - ERC_VarType (INT)：该变量的类型：0(BOOL) 1(INT) 2(REAL)。
 - ERC_VarScope (INT)：该变量的变量域：1(全局变量) 2(工程变量) 3(程序变量)。
 - ERC_VarValue (REAL)：要修改的值。
 - BOOL 类型 1.0 代表 TRUE，0.0 代表 FALSE；
 - INT 类型 1.0 代表 1，2.0 代表 2，以此类推；
 - REAL 类型 正常赋值即可。
 - WriteToFile (BOOL)：是否将变化写入文件，若写入文件，那么该变量值将在系统重启、切换工程(工程变量或程序变量)或程序(程序变量)后得以保持。
- 输出参数：
 - Busy (BOOL)：TRUE 表示正在执行。
 - Done (BOOL)：TRUE 表示执行完成
 - Error (BOOL)：TRUE 表示执行错误
 - Error_ID (INT)：执行成功时输出 0，执行出错时输出错误号。
- 注：控制器在远程模式下，功能才可生效！

3.1.13 ER_SetErrorLog



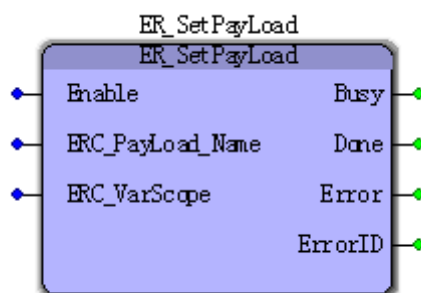
- 功能： PLC 添加一条系统错误日志到控制器。
- 输入参数：
 - Enable (BOOL)： 上升沿执行发送错误日志操作。
 - Error_Log_id (DINT)： 错误日志 id 号。
 - Error_Level (DINT)： 错误信息等级： 1 严重错误； 2 错误； 3 警告； 4 信息。
 - Error_Info (STRING)： 用户自定义错误信息。
- 输出参数：
 - Busy (BOOL)： TRUE 表示正在执行。
 - Done (BOOL)： TRUE 表示执行完成
 - Error (BOOL)： TRUE 表示执行错误
 - Error_ID (INT)： 执行成功时输出 0， 执行出错时输出错误号。
- 注：
 - 1、 错误等级 1、 2 会导致程序停止， 错误等级 3、 4 不会导致程序停止；
 - 2、 当错误等级输入其他值时， 一律按等级 4 处理；
 - 3、 错误日志 id 号注意与控制器端报警号作区分， 避免引起歧义；
 - 4、 当错误信息输入中文时， 需要用英文的单引号'' 将中文括起来， 否则会导致控制器系统运行异常。

3.1.14 ER_SetOverride



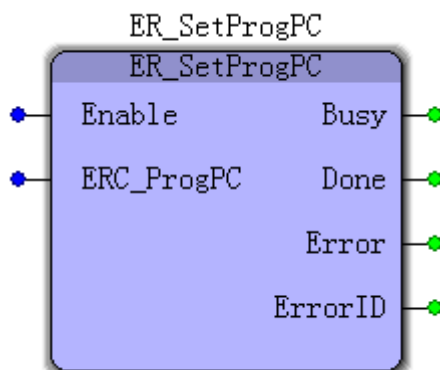
- 功能： 控制器处于远程模式时， 设置控制器端全局速度。
- 输入参数：
 - Enable (BOOL): 使能信号， 上升沿执行设置操作。
 - OverrideFactor (LREAL): 速度值
- 输出参数：
 - Busy (BOOL): TRUE 表示正在执行。
 - Done (BOOL): TRUE 表示执行完成
 - Error (BOOL): TRUE 表示执行错误
 - Error_ID (INT): 执行成功时输出 0， 执行出错时输出错误号。
- 注： 控制器在远程模式下， 功能才可生效！

3.1.15 ER_SetPayLoad



- 功能： 控制器处于远程模式时， 设置机器人负载质量。
- 输入参数：
 - Enable (BOOL): 上升沿执行设置负载质量操作。
 - ERC_PayLoadName (STR50): 要设置的负载变量名。
 - ERC_VarScope (INT) : 该 PAYLOAD 变量类型在控制器中的变量域：
 - 0:系统域
 - 1: 全局域
- 输出参数：
 - Busy (BOOL): TRUE 表示正在执行。
 - Done (BOOL): TRUE 表示执行完成
 - Error (BOOL): TRUE 表示执行错误
 - Error_ID (INT): 执行成功时输出 0， 执行出错时输出错误号。
- 注： 控制器在远程模式下， 功能才可生效！

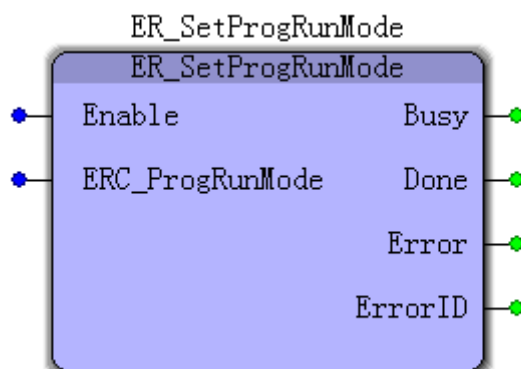
3.1.16 ER_SetProgPC



- 功能：控制器系统处于远程模式下，设置当前加载程序的 PC 指针。
- 输入参数：
 - Enable(BOOL)：使能信号，上升沿执行设置 PC 指针操作。
 - ERC_ProgPC(INT)：要设置的 PC 指针。
- 输出参数：
 - Busy(BOOL)：TRUE 表示正在执行；
 - Done(BOOL)：TRUE 表示执行完成；
 - Error(BOOL)：TRUE 表示执行出错；
 - ErrorID(INT)：执行成功时输出 0，执行出错时输出错误号。

注：控制器在远程模式下，功能才可生效！出于安全考虑，本功能将不在 PLC 端开放。

3.1.17 ER_SetProgRunMode

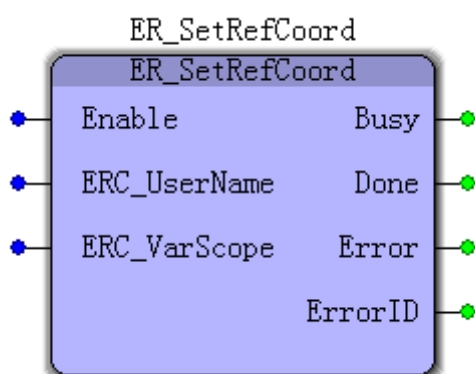


- 功能：控制器系统处于远程模式时，设置控制器程序运行模式。
- 输入参数：
 - Enable(BOOL)：使能信号，上升沿有效，执行设置控制器程序运行模式操作。

ERC_ProgRunMode(INT): 要设置的运行模式, 1(单步) 2(连续)。

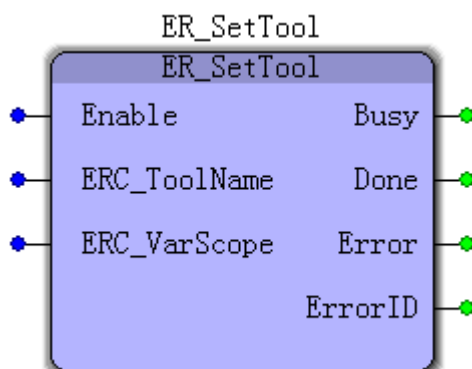
- 输出参数:
 - Busy(BOOL): TRUE 表示正在执行;
 - Done(BOOL): TRUE 表示执行完成;
 - Error(BOOL): TRUE 表示执行出错;
 - ErrorID(INT): 执行成功时输出 0, 执行出错时输出错误号。
- 注: 控制器在远程模式下, 功能才可生效!

3.1.18 ER_SetRefCoord



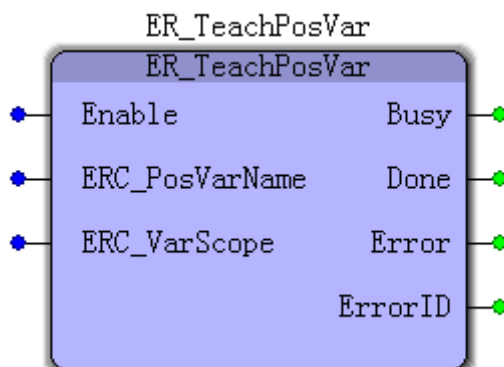
- 功能: 机器人控制器处于远程模式下, 设置参考坐标系。
- 输入参数:
 - Enable(BOOL): 使能信号, 上升沿有效, 执行设置参考坐标系操作。
 - ERC_UserName(String_50): 要设置的参考坐标系, 区分大小写, 注意与控制器端参考坐标系的名称统一。
 - ERC_VarScope(INT): 该参考坐标系的域
 - 0: 系统域(仅支持 World 坐标系)
 - 1: 全局域(用户坐标系)
- 输出参数:
 - Busy(BOOL): TRUE 表示正在执行;
 - Done(BOOL): TRUE 表示执行完成;
 - Error(BOOL): TRUE 表示执行出错;
 - ErrorID(INT): 执行成功时输出 0, 执行出错时输出错误号。
- 注: 控制器在远程模式下, 功能才可生效!

3.1.19 ER_SetTool



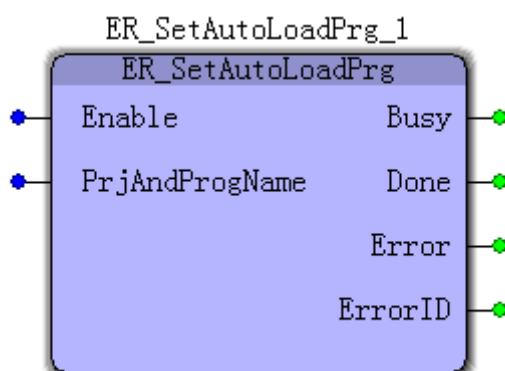
- 功能：机器人控制器处于远程模式时，设置工具坐标系。
- 输入参数：
 - Enable(BOOL): 使能信号，上升沿有效，执行设置参考坐标系操作。
 - ERC_ToolName(String_50): 要设置的工具坐标系名称，区分大小写，注意与控制器端工具名称统一。
 - ERC_VarScope(INT): 该工具坐标系的域
 - 0: 系统域(仅支持 nullTool)
 - 1: 全局域(用户自定义工具)
- 输出参数：
 - Busy(BOOL): TRUE 表示正在执行；
 - Done(BOOL): TRUE 表示执行完成；
 - Error(BOOL): TRUE 表示执行出错；
 - ErrorID(INT): 执行成功时输出 0，执行出错时输出错误号。
- 注：控制器在远程模式下，功能才可生效！

3.1.20 ER_TeachPosVar



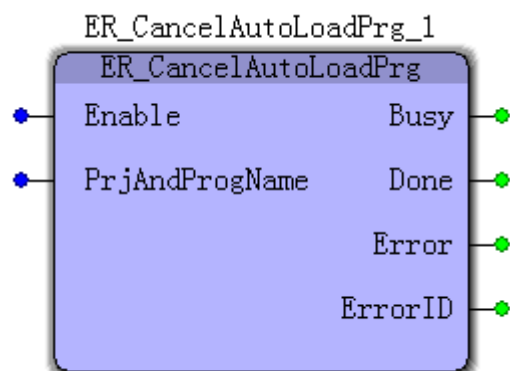
- 功能：系统处于远程模式时，示教位置点变量。
- 输入参数：
 - Enable(BOOL)：使能信号，上升沿有效，执行示教位置点操作。
 - ERC_PosVarName(STRING_50)：要示教的控制器端的位置点变量名。
 - ERC_VarScope(INT)：要示教的位置点变量的域(1:全局 2: 工程 3: 程序)。
- 输出参数：
 - Busy(BOOL)：TRUE 表示正在执行；
 - Done(BOOL)：TRUE 表示执行完成；
 - Error(BOOL)：TRUE 表示执行出错；
 - ErrorID(INT)：执行成功时输出 0，执行出错时输出错误号。
- 注：控制器在非远程模式下，功能也可生效！

3.1.21 ER_SetAutoLoadPrg



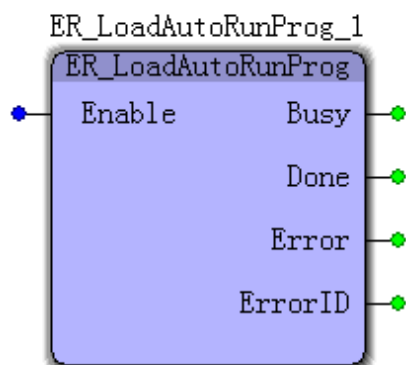
- 功能：设置远程模式下自启动程序。
- 输入参数：
 - Enable(BOOL)：使能信号，上升沿执行设置操作。
 - PrjAndProgName(STRING50)：“工程名.程序名”，要设置的自启动程序。
- 输出参数：
 - Busy(BOOL)：输出 TRUE 表示命令执行中。
 - Done(BOOL)：输出 TRUE 表示命令执行完成。
 - Error(BOOL)：输出 TRUE 表示命令执行出错。
 - ErrorID(INT)：错误代码。

3.1.22 ER_CancelAutoLoadPrg



- 功能：远程模式下，取消某程序远程模式自动加载，功能块图如下。
- 输入参数：
 - Enable(BOOL)：使能信号，上升沿执行取消自启动操作，远程模式有效。
 - PrjAndProgName(STRING50)：“工程名.程序名”，要取消自启动的程序。
- 输出参数：
 - Busy(BOOL)：输出 TRUE 表示命令执行中。
 - Done(BOOL)：输出 TRUE 表示命令执行完成。
 - Error(BOOL)：输出 TRUE 表示命令执行出错。
 - ErrorID(INT)：错误代码。

3.1.23 ER_LoadAutoRunProg



- 功能：远程模式下，加载设置为远程模式下自启动的程序。

- 输入参数:

Enable(BOOL): 上升沿执行加载自启动程序的操作, 远程模式下有效。

- 输出参数:

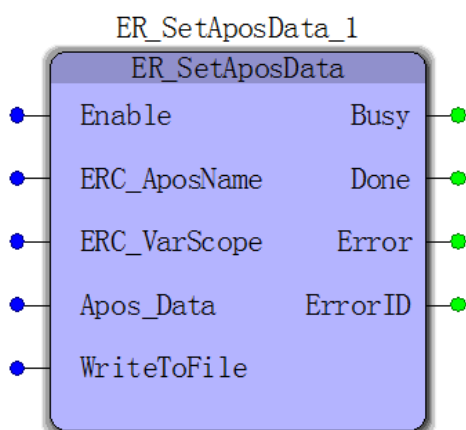
Busy(BOOL): 输出 TRUE 表示命令执行中。

Done(BOOL): 输出 TRUE 表示命令执行完成。

Error(BOOL): 输出 TRUE 表示命令执行出错。

ErrorID(INT): 错误代码。

3.1.24 ER_SetAposData



- 功能: 修改控制器端 APOS 类型变量。

- 输入参数:

Enable(BOOL): 使能信号, 上升沿生效, 执行修改变量操作。

ERC_AposName(STRING_50) : 待修改的 APOS 类型变量名, 区分大小写。

ERC_VarScope(INT) : 该变量的变量域: 1(全局变量), 2(工程变量), 3(程序变量)。

Apos_Data(APOS_DATA) : 要修改的值。

WriteToFile(BOOL) : 是否将变化写入文件, 可缺省, 默认为 FALSE 不写文件, 为 TRUE 时写文件。

- 输出参数:

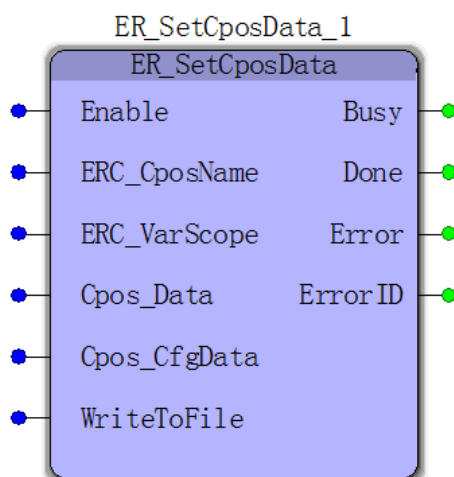
Busy(BOOL): 输出 TRUE 表示命令执行中。

Done(BOOL): 输出 TRUE 表示命令执行完成。

Error(BOOL): 输出 TRUE 表示命令执行出错。

ErrorID(INT): 错误代码。

3.1.25 ER_SetCposData



- 功能：修改控制器端 CPOS 类型变量。

- 输入参数：

Enable(BOOL): 使能信号，上升沿生效，执行修改变量操作。

ERC_CposName(String_50) : 要修改的 CPOS 类型变量名，区分大小写。

ERC_VarScope(INT) : 该变量的变量域：1(全局变量)，2(工程变量)，3(程序变量)。

Cpos_Data(CPOS_DATA) : 要修改的 CPOS 位置值。

Cpos_CfgData (CPOS_CFG): 要修改的 CPOS 位置姿态配置。

WriteToFile(BOOL) : 是否将变化写入文件，可缺省，默认为 FALSE 不写文件，为 TRUE 时写文件。

- 输出参数：

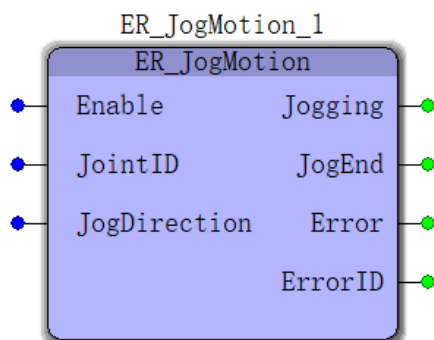
Busy(BOOL): 输出 TRUE 表示命令执行中。

Done(BOOL): 输出 TRUE 表示命令执行完成。

Error(BOOL): 输出 TRUE 表示命令执行出错。

ErrorID(INT): 错误代码。

3.1.26 ER_JogMotion



- 功能：手动模式下，点动机器人。
- 输入参数：

Enable(BOOL): 点动使能，为 TRUE 时执行点动，变 FALSE 时停止点动。

JointID(USINT) : 关节坐标系下为点动关节轴号，取值范围 1-16;

笛卡尔坐标系下表示 XYZ 平移和 ABC 旋转,取值范围 1-6。

JogDirection(INT) : 点动方向: 1 正向、-1 反向。

- 输出参数：

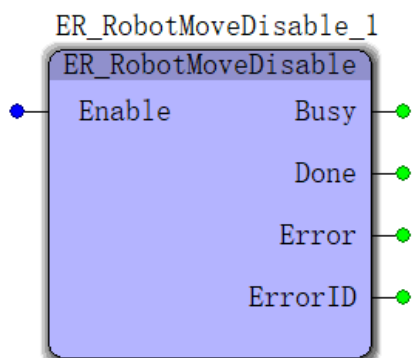
Jogging(BOOL): 输出 TRUE 表示点动中。

JogEnd(BOOL): 输出 TRUE 表示点动停止。

Error(BOOL): 输出 TRUE 表示命令执行出错。

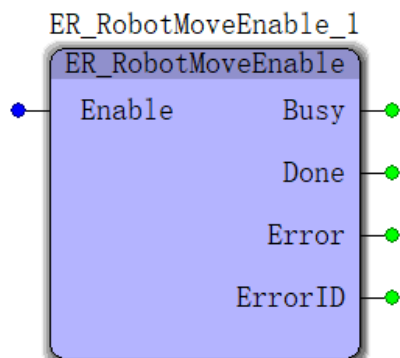
ErrorID(INT): 错误代码。

3.1.27 ER_RobotMoveDisable



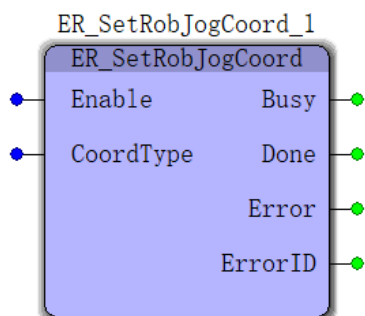
- 功能：远程模式下，设置机器人机械运动掉使能。
- 输入参数：
 - Enable(BOOL)：上升沿执行机器人机械运动掉使能操作。
- 输出参数：
 - Busy(BOOL)：输出 TRUE 表示正在执行。
 - Done(BOOL)：输出 TRUE 表示执行完成。
 - Error(BOOL)：输出 TRUE 表示命令执行出错。
 - ErrorID(INT)：错误代码。
- 注：
 - 控制器在远程模式下，功能才可生效！
 - 当机械运动掉使能时，远程模式下机器人禁止运动，手动/自动模式不受限制。

3.1.28 ER_RobotMoveEnable



- 功能：远程模式下，设置机器人机械运动上使能。
- 输入参数：
 - Enable(BOOL)：上升沿执行机器人机械运动上使能操作。
- 输出参数：
 - Busy(BOOL)：输出 TRUE 表示正在执行。
 - Done(BOOL)：输出 TRUE 表示执行完成。
 - Error(BOOL)：输出 TRUE 表示命令执行出错。
 - ErrorID(INT)：错误代码。
- 注：
 - 控制器在远程模式下，功能才可生效！
 - 当机械运动使能时，远程模式下机器人允许运动。

3.1.29 ER_SetRobJogCoord



- 功能：用于设置机器人点动坐标系类型。
- 输入参数：

Enable(BOOL)：上升沿执行设置操作。

CoordType(USINT)：点动坐标系类型：1 关节 2 世界 3 工具 4 用户。
- 输出参数：

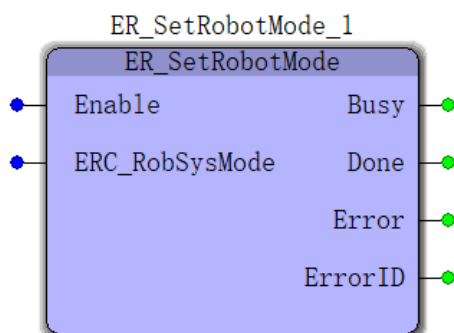
Busy(BOOL)：输出 TRUE 表示正在执行。

Done(BOOL)：输出 TRUE 表示执行完成。

Error(BOOL)：输出 TRUE 表示命令执行出错。

ErrorID(INT)：错误代码。

3.1.30 ER_SetRobotMode



- 功能：用于设置机器人系统模式。
- 输入参数：

Enable(BOOL)：上升沿执行设置操作。

ERC_RobSysMode(INT)：机器人系统模式：1 手动 2 自动 3 远程
- 输出参数：

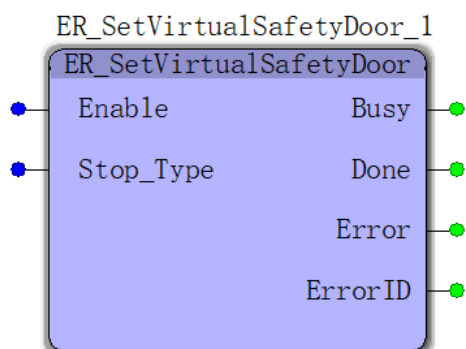
Busy(BOOL)：输出 TRUE 表示正在执行。

Done(BOOL)：输出 TRUE 表示执行完成。

Error(BOOL)：输出 TRUE 表示命令执行出错。

ErrorID(INT)：错误代码。

3.1.31 ER_SetVirtualSafetyDoor



- 功能：用于设置虚拟安全门停止方式，并停止机器人系统运行。

- 输入参数：

Enable(BOOL): 上升沿生效，依据输入参数，执行不同的停止方式。

Stop_Type(DINT): 机器人停止方式：0 延时减速停止，1 紧急停止(输入其他值时，按 0 处理)。

- 输出参数：

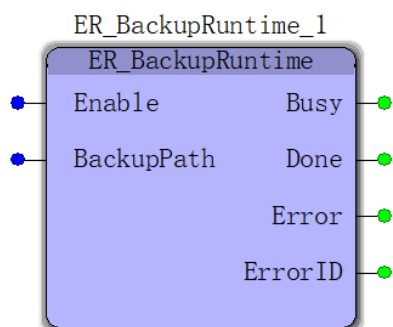
Busy(BOOL): 输出 TRUE 表示正在执行。

Done(BOOL): 输出 TRUE 表示执行完成。

Error(BOOL): 输出 TRUE 表示命令执行出错。

ErrorID(INT): 错误代码。

3.1.32 ER_BackupRuntime



- 功能：备份控制器中的 runtime 包，并拷贝到指定路径。

- 输入参数:

Enable(BOOL): 上升沿执行备份操作。

BackupPath(INT): 备份路径选择:

0: 往共享文件夹拷贝;

1: 往 FTP 服务器拷贝;

具体路径在示教器端设置(高级设置->控制器备份恢复->远程备份设置)。

- 输出参数:

Busy(BOOL): 输出 TRUE 表示正在执行。

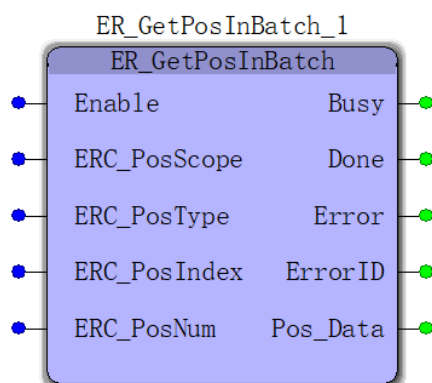
Done(BOOL): 输出 TRUE 表示执行完成。

Error(BOOL): 输出 TRUE 表示命令执行出错。

ErrorID(INT): 执行成功时输出 0, 执行出错时输出错误号:

注: 每次备份操作需要 5 分钟。

3.1.33 ER_GetPosInBatch



- 功能: 批量获取控制器端名称连续的点位变量数据。

- 输入参数:

Enable(BOOL): 上升沿执行操作;

ERC_PosScope (INT): 要获取变量的域: 1(全局),2(工程),3(程序);

ERC_PosType (INT): 要获取点变量的类型: 0(APOS),1(CPOS);

ERC_PosIndex (INT): 要获取的 P 变量的起始序号, 如输入为 1, 表示从 P1 开始获取;

ERC_PosNum (INT): 要获取的 P 变量的数量, 最大支持获取 10 个变量, 当该值为 0 或超范围时默认按 1 处理。

- 输出参数:

Busy(BOOL): 输出 TRUE 表示正在执行。

Done(BOOL): 输出 TRUE 表示执行完成。

Error(BOOL): 输出 TRUE 表示命令执行出错。

ErrorID(INT): 执行成功时输出 0, 执行出错时输出错误号:

1: 内存上锁, 命令下发失败;

2: 输入类型转换不成功, 命令下发失败;

3: 获取点位失败, 请检查名称连续的点位变量是否都存在。

Pos_Data (POS_ARRAY_IN_BATCH):

POS_ARRAY_IN_BATCH 数据类型为二维数组变量,

第一个下标表示获取的第几个变量, 范围 0-9, 最大支持获取 10 个变量;

第二个下标表示获取点位变量的具体值, 范围 0-22;

当获取 APOS 类型时, 以第一个变量为例:

Pos_Data[0][0] - Pos_Data[0][6]为空;

Pos_Data[0][7] - Pos_Data[0][12]分别对应一轴到六轴的关节位置;
其余存放附加轴位置。

当获取 CPOS 类型时, 以第一个变量为例:

Pos_Data[0][0] - Pos_Data[0][6]表示 confdata 的值,

Pos_Data[0][7]:表示“CPOS”变量的“x”;

Pos_Data[0][8]:表示“CPOS”变量的“y”;

Pos_Data[0][9]:表示“CPOS”变量的“z”;

Pos_Data[0][10]:表示“CPOS”变量的“a”;

Pos_Data[0][11]:表示“CPOS”变量的“b”;

Pos_Data[0][12]:表示“CPOS”变量的“c”;

其余存放附加轴位置。

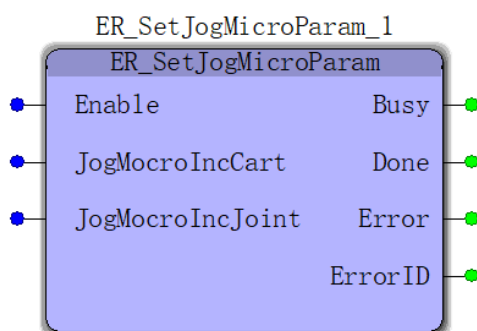
注 1:

1、默认获取的变量名以 P 开头, 依据输入的 ERC_PosIndex 来组成变量名, 如输入 ERC_PosIndex 为 1, 输入 ERC_PosNum 为 10, 则最终会获取示教器上 P1,P2,P3...P10 的值。

2、输入 Enable 上升沿获取变量数据, 下降沿复位变量数据!

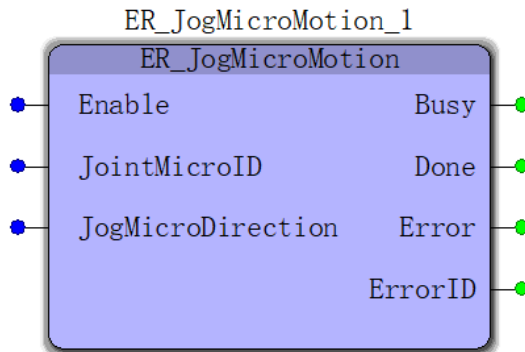
注 2: 该功能块为选配功能, 默认不开放, 如需使用请联系厂家人员!

3.1.34 ER_SetJogMicroParam



- 功能：用于设置 MultiProg 端寸动的线性步长与关节步长。
- 输入参数：
 - Enable(BOOL)：上升沿执行设置操作。
 - JogMocroIncCart (LREAL)：线性步长，范围 0.0-10.0mm，当该值为 0.0 或超范围时默认按 1.0 处理；
 - JogMocroIncjoint (LREAL)：关节步长，范围 0.0-1.0°，当该值为 0.0 或超范围时默认按 0.5 处理；
- 输出参数：
 - Busy(BOOL)：输出 TRUE 表示正在执行。
 - Done(BOOL)：输出 TRUE 表示执行完成。
 - Error(BOOL)：输出 TRUE 表示命令执行出错。
 - ErrorID(INT)：执行成功时输出 0，执行出错时输出错误号：
 - 1：内存上锁，命令下发失败；
 - 2：输入类型转换不成功，命令下发失败；
 - 3：控制器执行错误或超时。

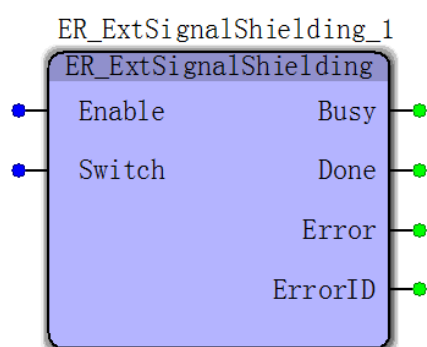
3.1.35 ER_JogMicroMotion



- 功能：控制机器人寸动。
- 输入参数：
 - Enable(BOOL)：上升沿执行寸动操作。
 - JointMicroID (USINT)：关节坐标系下为寸动关节轴号，取值范围 1-16；
笛卡尔坐标系下表示 XYZ 平移和 ABC 旋转，取值范围 1-6；
 - JogMicroDirection (INT)：寸动方向：1 正向、-1 反向。
- 输出参数：
 - Busy(BOOL)：输出 TRUE 表示正在执行。
 - Done(BOOL)：输出 TRUE 表示执行完成。
 - Error(BOOL)：输出 TRUE 表示命令执行出错。
 - ErrorID(INT)：执行成功时输出 0，执行出错时输出错误号：
 - 1：内存上锁，命令下发失败；

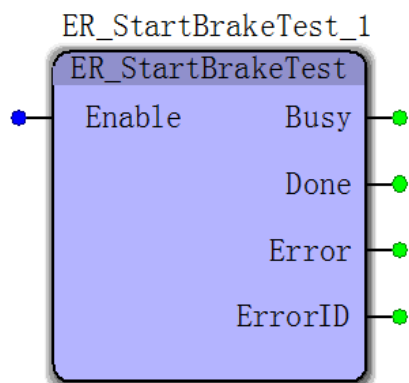
- 2: 输入类型转换不成功, 命令下发失败;
- 3: 控制器执行错误或超时;
- 4: 参数错误, 请检查输入参数;
- 5: 机器人未上励磁;
- 6: 机器人处于运动状态, 不响应寸动命令;
- 7: 当前系统模式不在手动模式;
- 8: 控制器程序处于运行状态, 禁止寸动;
- 9: 控制器系统处于错误状态;
- 10: 正在响应点动功能块, 寸动命令下发失败。

3.1.36 ER_ExtSignalShielding



- 功能: 开启和关闭外部信号屏蔽功能。
- 输入参数:
 - Enable(BOOL): 上升沿执行设置操作。
 - Switch(INT): 外部信号屏蔽开关: 0 关闭 1 开启
- 输出参数:
 - Busy(BOOL): 输出 TRUE 表示正在执行。
 - Done(BOOL): 输出 TRUE 表示执行完成。
 - Error(BOOL): 输出 TRUE 表示命令执行出错。
 - ErrorID(INT): 错误代码。

3.1.37 ER_StartBrakeTest



- 功能：开启制动测试功能。
- 输入参数：
 - Enable(BOOL)：上升沿执行开启制动测试操作。
- 输出参数：
 - Busy(BOOL)：输出 TRUE 表示开启动作正在执行。
 - Done(BOOL)：输出 TRUE 表示开启动作执行完成。
 - Error(BOOL)：输出 TRUE 表示命令执行出错。
 - ErrorID(INT)：错误代码。

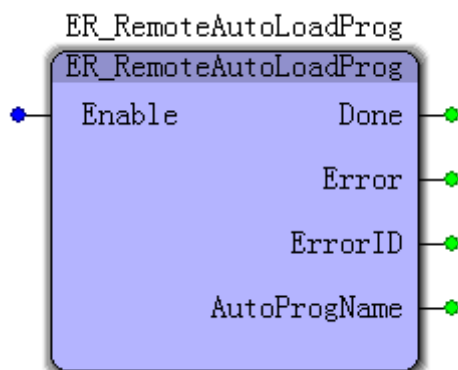
3.2 ER_CNT2PLC_Status: 获取状态功能块

ER_CNT2PLC_Status 用户库提供了一系列获取与控制器相关的状态的功能块，供用户在 PLC 程序中获取所需的控制器信息。开放的所有获取控制器状态功能块见表 3.2。

表 3.2 控制器状态获取接口统计

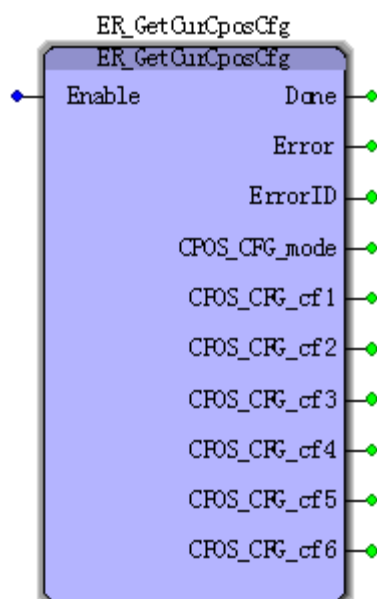
功能块名	实现功能
ER_RemoteAutoLoadProg	获取机器人远程模式下，设置为自动加载的程序名。
ER_GetCurCposCfg	获取当前机器人笛卡尔坐标系数据。
ER_GetCurCpos	获取机器人在笛卡尔坐标系中的当前位置。
ER_GetCurJpos	获取机器人在关节空间坐标系中的当前位置。
ER_GetGlobalSpeed	获取机器人当前的全局速度。
ER_GetProgRunMode	获取当前控制器程序运行模式。
ER_GetCurLoadProg	获取当前控制器加载程序的程序名。
ER_GetRobPowerOnState	获取当前机器人是否在励磁状态。
ER_GetRobMovState	获取当前机器人是否在运动的状态。
ER_GetProgRunState	获取当前程序运行状态。
ER_GetSystemState	获取当前控制器系统状态。
ER_GetSystemMode	获取当前控制器系统模式。
ER_GetSysError	获取控制器报警数和报警代码
ER_RobModbusCnctState	获取标准 ModbusTCP 连接状态
ER_GetRobJogCoord	获取当前点动坐标系类型
ER_GetAxisServoError	获取伺服轴是否有伺服报警
ER_GetRobotHomeState	获取伺服轴是否丢零点
ER_GetRobSysTime	获取机器人控制器系统时间
ER_GetAxisTorqueValue	获取机器人每个轴的力矩值(百分比)
ER_GetRobNearPath	获取机器人末端是否在编程规划的轨迹附近
ER_GetRobOnPath	获取机器人末端是否在编程规划的轨迹上
ER_GetTool	获取系统加载的工具坐标系信息
ER_GetRefCoord	获取系统加载的参考坐标系信息
ER_GetMoveEnableState	获取机器人当前机械使能状态
ER_GetAxisTorqueNm	获取机器人每个轴的力矩值(牛顿·米)
ER_GetCurTCPVelocity	获取机器人末端 TCP 的速度
ER_GetRobSysInfo	获取机器人系统信息
ER_GetRobRunningTime	获取机器人系统运行时间(励磁、运动、程序运行)
ER_GetRobMaintainSts	获取机器人维保状态(本体、电池)
ER_GetMultiRobColliSts	获取双机碰撞状态
ER_GetAxisCurrent	获取机器人每个轴的电流值(安培)
ER_GetConveyorPosition	获取传送带当前位置
ER_GetAreaActSts	获取区域激活状态
ER_GetPolyAreaActSts	获取多边形区域激活状态

3.2.1 ER_RemoteAutoLoadProg



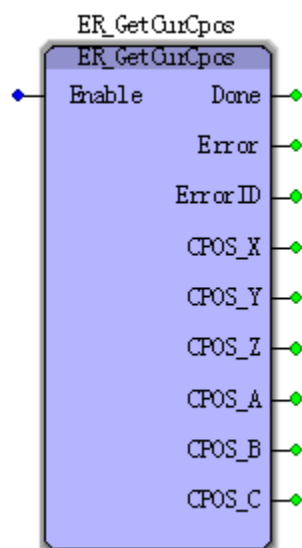
- 功能：获取机器人远程模式设置为自动加载的程序。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，远程自动加载程序名实时刷新。
- 输出参数：
 - Done(BOOL): TRUE 表示获取成功；
 - Error(BOOL): TRUE 表示获取出错；
 - ErrorID(INT): 获取成功时输出 0，出错时输出错误号。
 - AutoProgName(String_50): 以“工程名.程序名”形式输出远程模式自动加载的程序名。

3.2.2 ER_GetCurCposCfg



- 功能：获取当前机器人笛卡尔空间下，基于用户坐标系的位置对应的空间形态。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，输出对应空间形态信息实时刷新。
- 输出参数：
 - Done(BOOL): TRUE 表示获取完成；
 - Error(BOOL): TRUE 表示获取出错；
 - ErrorID(INT): 获取成功时输出 0，获取出错时输出错误号。
 - CPOS_CFG_mode(DINT): 当前位置与姿态对应的运动学 mode 值。
 - CPOS_CFG_cf1(DINT): 当前位置的 1 轴角度所在象限。
 - CPOS_CFG_cf2(DINT): 当前位置的 2 轴角度所在象限。
 - CPOS_CFG_cf3(DINT): 当前位置的 3 轴角度所在象限。
 - CPOS_CFG_cf4(DINT): 当前位置的 4 轴角度所在象限。
 - CPOS_CFG_cf5(DINT): 当前位置的 5 轴角度所在象限。
 - CPOS_CFG_cf6(DINT): 当前位置的 6 轴角度所在象限。

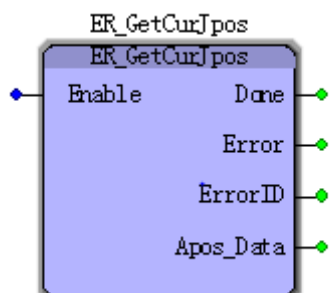
3.2.3 ER_GetCurCpos



- 功能：获取当前机器人在笛卡尔坐标系中的当前位置。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，输出笛卡尔坐标位置数据实时刷新。
- 输出参数：
 - Done(BOOL): TRUE 表示获取完成；
 - Error(BOOL): TRUE 表示获取出错；
 - ErrorID(INT): 获取成功时输出 0，获取出错时输出错误号。
 - CPOS_X(LREAL): TCP 点在参考坐标系上 x 方向的坐标。

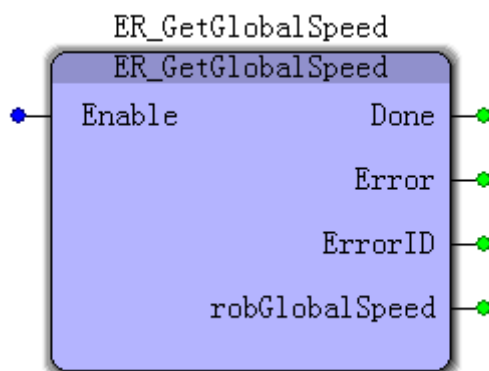
- CPOS_Y(LREAL): TCP 点在参考坐标系上 y 方向的坐标。
 CPOS_Z(LREAL): TCP 点在参考坐标系上 z 方向的坐标。
 CPOS_A(LREAL): TCP 点相对于参考坐标系 z 轴旋转的欧拉角。
 CPOS_B(LREAL): TCP 点相对于参考坐标系 z 轴旋转的欧拉角。
 CPOS_C(LREAL): TCP 点相对于参考坐标系 x''轴旋转的欧拉角。

3.2.4 ER_GetCurJpos



- 功能：获取机器人在关节空间中的当前位置。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，输出关节空间位置数据实时刷新。
- 输出参数：
 - Done(BOOL): TRUE 表示获取完成；
 - Error(BOOL): TRUE 表示获取出错；
 - ErrorID(INT): 获取成功时输出 0，获取出错时输出错误号。
 - Apos_Data(APOS_DATA): **APOS_DATA** 数据类型是下标为 0 到 15 的 REAL 型数组,依次代表机器人不同轴的位置（包括本体轴和附加轴）。

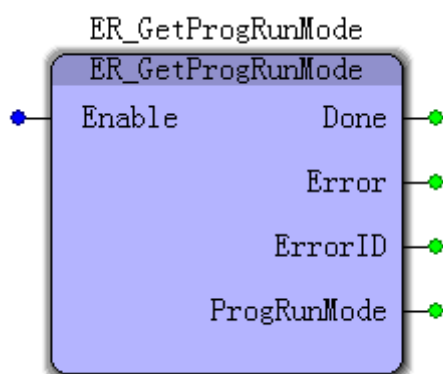
3.2.5 ER_GetGlobalSpeed



- 功能：获取机器人全局速度。

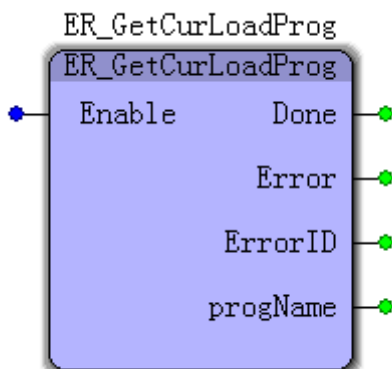
- 输入参数：
Enable (BOOL): Enable 为 TRUE 时，全局速度数据实时刷新。
- 输出参数：
Done(BOOL): TRUE 表示获取成功；
Error(BOOL): TRUE 表示获取出错；
ErrorID(INT): 获取成功时输出 0，出错时输出错误号。
robGlobalSpeed(LREAL): 控制器当前全局速度值。

3.2.6 ER_GetProgRunMode



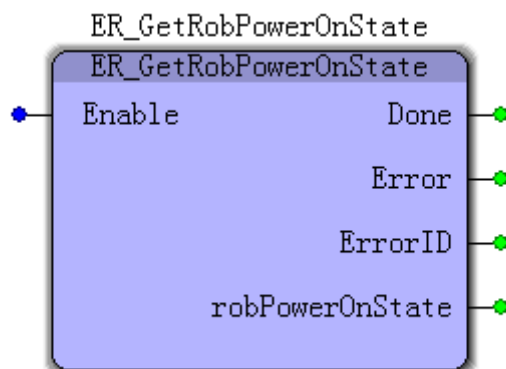
- 功能：获取当前控制器程序运行模式。。
- 输入参数：
Enable (BOOL): Enable 为 TRUE 时，运行模式实时刷新。
- 输出参数：
Done(BOOL): TRUE 表示获取成功；
Error(BOOL): TRUE 表示获取出错；
ErrorID(INT): 获取成功时输出 0，出错时输出错误号。
ProgRunMode(INT): 1 表示单步模式； 2 表示连续模式。

3.2.7 ER_GetCurLoadProg



- 功能：获取当前控制器加载程序的程序名。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，当前加载程序名实时刷新。
- 输出参数：
 - Done(BOOL): TRUE 表示获取成功；
 - Error(BOOL): TRUE 表示获取出错；
 - ErrorID(INT): 获取成功时输出 0，出错时输出错误号。
 - progName(String_50): 以“工程名.程序名”形式输出当前加载的程序名。

3.2.8 ER_GetRobPowerOnState



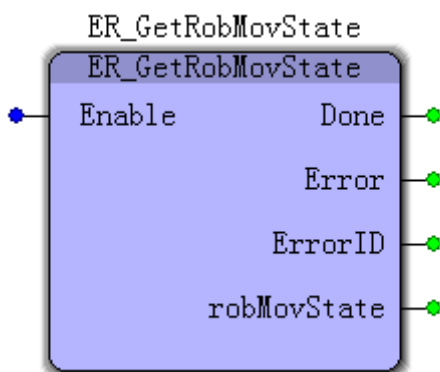
- 功能：获取当前机器人伺服励磁状态。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，励磁状态实时刷新。
- 输出参数：
 - Done(BOOL): TRUE 表示获取成功；

Error(BOOL): TRUE 表示获取出错;

ErrorID(INT): 获取成功时输出 0, 出错时输出错误号。

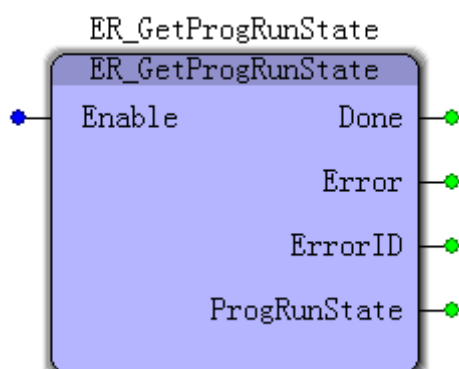
robPowerOnState (BOOL): 机器人伺服励磁状态, TRUE 表示在励磁状态。

3.2.9 ER_GetRobMovState



- 功能：获取当前机器人运动状态。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时, 运动状态实时刷新。
- 输出参数：
 - Done(BOOL): TRUE 表示获取成功;
 - Error(BOOL): TRUE 表示获取出错;
 - ErrorID(INT): 获取成功时输出 0, 出错时输出错误号。
 - robMovState (BOOL): 机器人运动状态, TRUE 表示机器人正在运动。

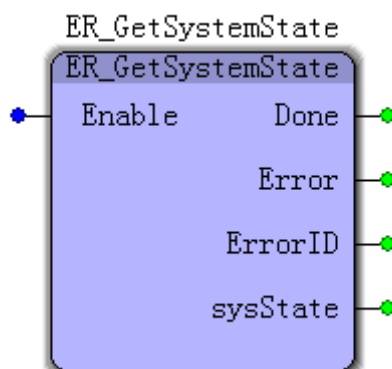
3.2.10 ER_GetProgRunState



- 功能：获取机器人当前程序运行状态。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时, 程序运行状态实时刷新。

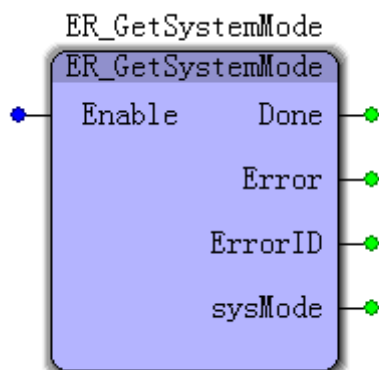
- 输出参数:
 - Done(BOOL): TRUE 表示获取成功;
 - Error(BOOL): TRUE 表示获取出错;
 - ErrorID(INT): 获取成功时输出 0, 出错时输出错误号。
 - ProgRunState (INT): 机器人运动状态, 0:初始化 1:正在运行 2:暂停 3:停止 4:错误。

3.2.11 ER_GetSystemState



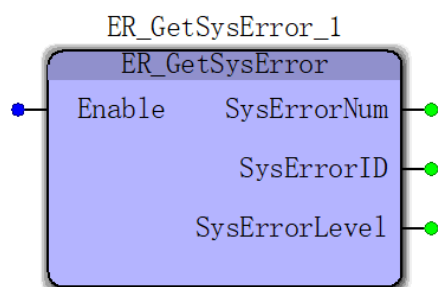
- 功能: 获取机器人控制系统当前状态。
- 输入参数:
 - Enable (BOOL): Enable 为 TRUE 时, 系统状态实时刷新。
- 输出参数:
 - Done(BOOL): TRUE 表示获取成功;
 - Error(BOOL): TRUE 表示获取出错;
 - ErrorID(INT): 获取成功时输出 0, 出错时输出错误号。
 - sysState (INT): 机器人系统状态, 1:初始化 2:正常运行 3:错误状态 4:系统错误。

3.2.12 ER_GetSystemMode



- 功能：获取机器人控制系统当前模式。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，系统模式实时刷新。
- 输出参数：
 - Done(BOOL): TRUE 表示获取成功；
 - Error(BOOL): TRUE 表示获取出错；
 - ErrorID(INT): 获取成功时输出 0，出错时输出错误号。
 - sysMode(INT): 机器人系统模式，0:手动模式 1:自动模式 2:远程模式。

3.2.13 ER_GetSysError

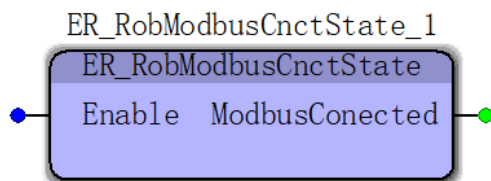


- 功能：获取控制器报警数和报警代码及报警等级。
- 输入参数：
 - Enable(BOOL):为 TRUE 时，实时获取控制器报警。
- 输出参数：
 - SysErrorNum(UINT): 当前系统报警数。
 - SysErrorID (ERC_DINT_ARRAY_10): 当前系统报警码，可存储十个（对应控

制器)。

SysErrorLevel (ERC_DINT_ARRAY_10): 当前系统报警等级, 可存储十个

3.2.14 ER_RobModbusCnctState



- 功能: 获取控制器 ModbusTcp 连接状态。

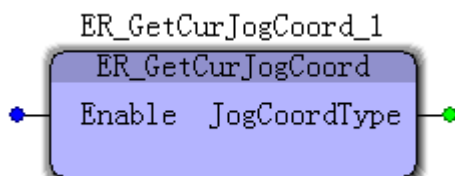
- 输入参数:

Enable(BOOL):为 TRUE 时, 实时获取 ModbusTcp 连接状态。

- 输出参数:

ModbusCnected(BOOL): 为 TRUE 表示已经连接, FALSE 表示未连接。

3.2.15 ER_GetRobJogCoord



- 功能: 获取当前点动坐标系类型。

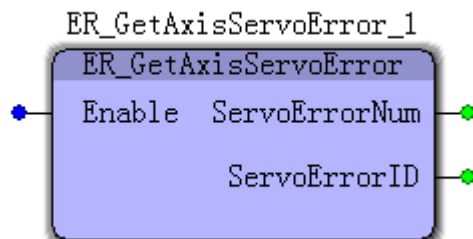
- 输入参数:

Enable (BOOL): 为 TRUE 期间获取当前点动坐标系类型。

- 输出参数:

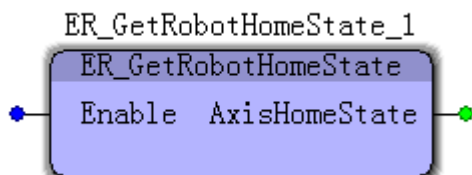
JogCoordType (INT): 1:关节 2:世界 3:工具 4:用户, -1 表示获取失败。

3.2.16 ER_GetAxisServoError



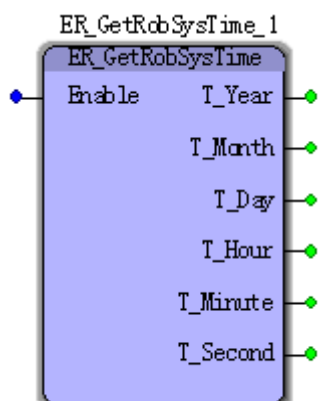
- 功能：获取伺服轴是否有伺服报警。
- 输入参数：
 - Enable(BOOL)：使能信号，为 TRUE 期间持续获取伺服错误状态。
- 输出参数：
 - ServoErrorNum(INT)：有报警的伺服轴数。
 - ServoErrorID(ERC_DINT_ARRAY)：存储对应轴当前的伺服报警号。

3.2.17 ER_GetRobotHomeState



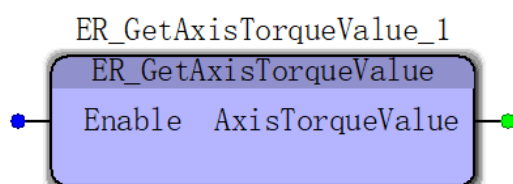
- 功能：获取伺服轴是否丢零点，功能块示意图如下。
- 输入参数：
 - Enable(BOOL)：使能信号，为 TRUE 期间持续获取机器人轴回零状态。
- 输出参数：
 - AxisHomeState(INT)：1:正常， -1 有轴丢零点。

3.2.18 ER_GetRobSysTime



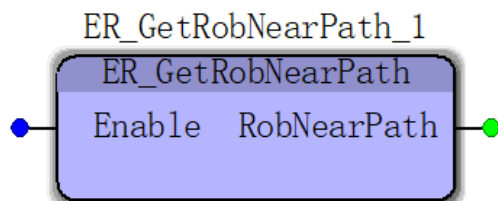
- 功能：获取机器人控制器系统时间。
- 输入参数：
Enable(BOOL)：为 TRUE 时持续输出系统时间。
- 输出参数：
T_Year (UINT) : 年;
T_Month (UINT) : 月;
T_Day (UINT) : 日;
T_Hour (UINT) : 时;
T_Minute (UINT): 分;
T_Second (UINT): 秒;

3.2.19 ER_GetAxisTorqueValue



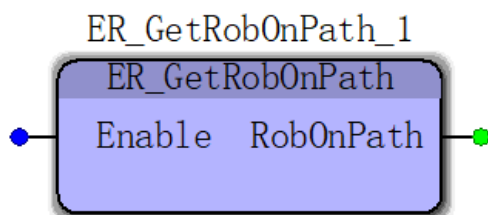
- 功能：获取机器人每个轴的力矩值(百分比)。
- 输入参数：
Enable(BOOL): Enable 为 TRUE 时, 机器人每个轴的力矩值(百分比)实时刷新。
- 输出参数：
AxisTorqueValue(ERC_REAL_ARRAY): **ERC_REAL_ARRAY** 数据类型是下标为 0 到 15 的 REAL 型数组, 依次代表机器人的各个轴号(包括本体轴和附加轴)。

3.2.20 ER_GetRobNearPath



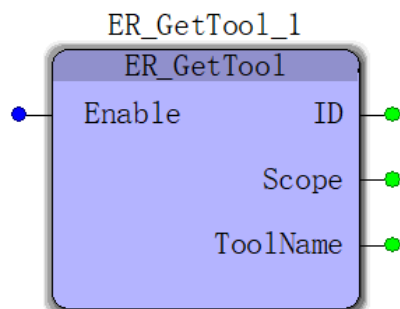
- 功能：获取机器人末端是否在编程规划的轨迹附近。
- 输入参数：
Enable(BOOL)：为 TRUE 时，机器人末端是否在编程规划的轨迹附近的状态实时刷新。
- 输出参数：
RobNearPath(BOOL)：获取机器人末端是否在编程规划的轨迹附近，为 TRUE 时表示在附近，为 FALSE 时表示超出阈值。

3.2.21 ER_GetRobOnPath



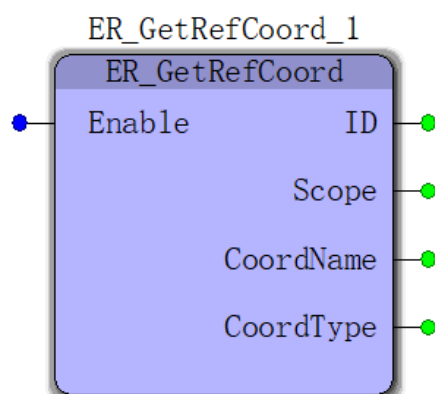
- 功能：获取机器人末端是否在编程规划的轨迹上。
- 输入参数：
Enable(BOOL)：为 TRUE 时，机器人末端是否在编程规划的轨迹上的状态实时刷新。
- 输出参数：
RobOnPath(BOOL)：获取机器人末端是否在编程规划的轨迹上，为 TRUE 时表示在轨迹上，为 FALSE 时表示不在轨迹上。

3.2.22 ER_GetTool



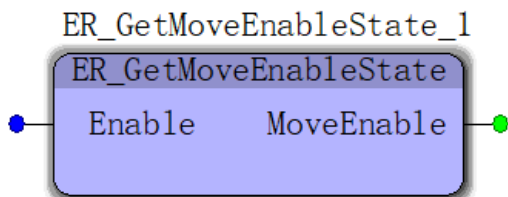
- 功能：获取系统加载的工具坐标系信息。
- 输入参数：
Enable(BOOL): Enable 为 TRUE 时，实时获取当前加载的工具坐标系信息。
- 输出参数：
ID(DINT): 工具坐标系 ID 号；
Scope(DINT): 工具坐标系所属的域：1:系统域，2:全局域；
ToolName(String_50): 工具坐标系名称。

3.2.23 ER_GetRefCoord



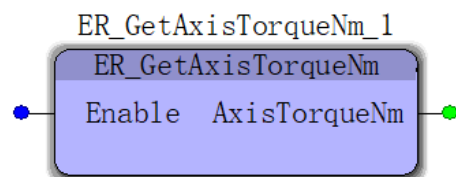
- 功能：获取系统加载的参考坐标系信息。
- 输入参数：
Enable(BOOL): Enable 为 TRUE 时，实时获取当前加载的参考坐标系信息。
- 输出参数：
ID(DINT): 坐标系 ID 号；
Scope(DINT): 坐标系所属的域：1:系统域，2:全局域；
CoordName(String_50): 坐标系名称；
CoordType(DINT): 坐标系类型：1:用户坐标系，2:外部 TCP 坐标系，3:变位机坐标系，4:动坐标系。

3.2.24 ER_GetMoveEnableState



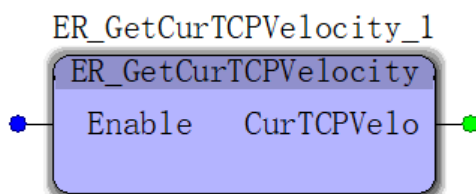
- 功能：获取机器人当前机械使能状态。
- 输入参数：
Enable(BOOL): Enable 为 TRUE 时，机器人机械使能状态实时刷新。
- 输出参数：
MoveEnable(BOOL): 为 TRUE 时表示允许机器人机械运动，为 FALSE 则不允许。
注：此版本 MoveEnable 信号只在远程模式下生效

3.2.25 ER_GetAxisTorqueNm



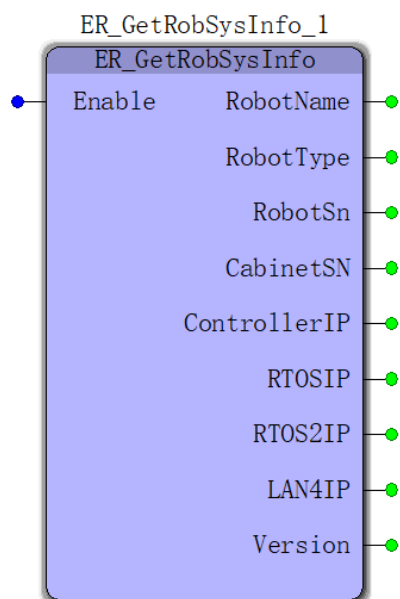
- 功能：获取机器人每个轴的力矩值(牛顿·米)。
- 输入参数：
Enable(BOOL): Enable 为 TRUE 时，机器人每个轴的力矩值(牛顿·米)实时刷新。
- 输出参数：
AxisTorqueNm(ERC_REAL_ARRAY): 获取到的机器人每个轴的力矩值(牛顿·米)。
注：输入 Enable 为 TRUE 时，各轴力矩值(牛顿·米)实时刷新。

3.2.26 ER_GetCurTCPVelocity



- 功能：获取机器人末端 TCP 的速度。
- 输入参数：
 - Enable(BOOL): Enable 为 TRUE 时, 实时获取机器人末端 TCP 的速度(mm/s)。
- 输出参数：
 - CurTCPVelo(LREAL): 获取到的机器人末端 TCP 速度。

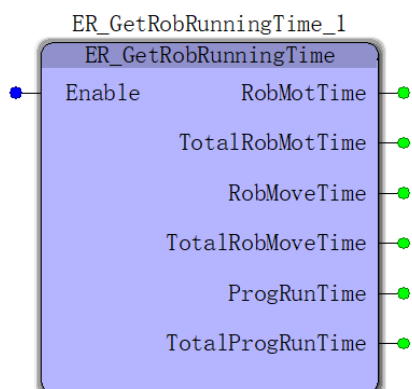
3.2.27 ER_GetRobSysInfo



- 功能：获取机器人系统信息。
- 输入参数：
 - Enable(BOOL): Enable 为 TRUE 时, 获取系统信息。
- 输出参数：
 - RobotName(String_50): 机器人名称;
 - RobotType(String_50): 机器人类型;
 - RobotSn(String_50): 机器人本体序列号;
 - CabinetSN(String_50): 电柜序列号;

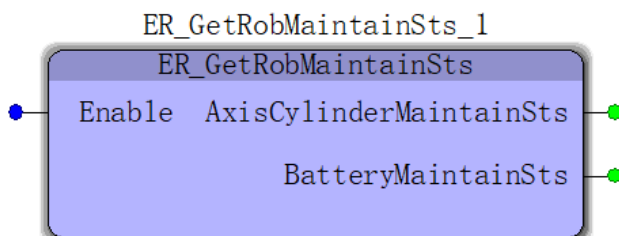
ControllerIP(String_50): 控制器(LAN2)IP;
 RTOSIP(String_50): 实时系统(LAN2)IP;
 RTOS2IP(String_50): 实时系统(LAN3)IP;
 LAN4IP(String_50): 控制器(LAN4)IP;
 Version(String_50): 版本信息。

3.2.28 ER_GetRobRunningTime



- 功能：获取机器人系统运行时间(励磁、运动、程序运行)。
- 输入参数：
 - Enable(BOOL): Enable 为 TRUE 时，实时获取系统运行参数。
- 输出参数：
 - RobMotTime(REAL): 机器人系统启动后励磁时间，断电重启复位；
 - TotalRobMotTime(REAL): 累计励磁时间，掉电保持；
 - RobMoveTime(REAL): 机器人系统启动后运动时间，断电重启复位；
 - TotalRobMoveTime(REAL): 累计运动时间，掉电保持；
 - ProgRunTime(REAL): 机器人系统启动后程序运行时间，断电重启复位；
 - TotalProgRunTime(REAL): 累计程序运行时间，掉电保持。
 注：以上时间单位：小时。

3.2.29 ER_GetRobMaintainSts

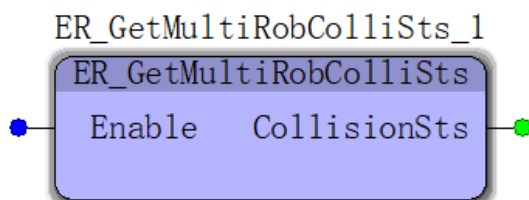


- 功能：获取机器人维保状态(本体、电池)。
- 输入参数：

Enable(BOOL): Enable 为 TRUE 时, 实时获取维保状态。

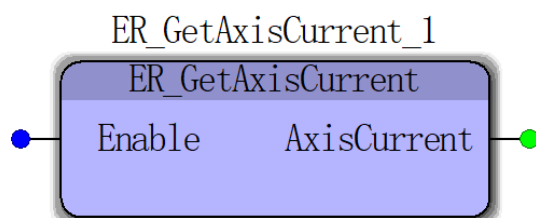
- 输出参数:
 - AxisCylinderMaintainSts(BOOL): 本体是否需要保养(减速机、丝杆和氮气缸), 为 TRUE 时表示需要保养;
 - BatteryMaintainSts(BOOL): 电池是否需要保养, 为 TRUE 时表示需要保养。

3.2.30 ER_GetMultiRobColliSts



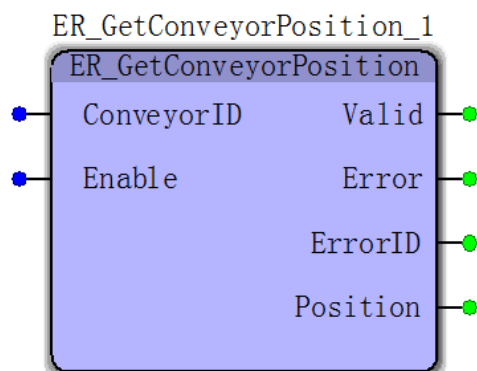
- 功能: 获取双机碰撞状态。
- 输入参数:
 - Enable(BOOL): Enable 为 TRUE 时, 实时获取双机碰撞状态。
- 输出参数:
 - CollisionSts(UINT): 当前双机碰撞状态, 0:无碰撞, 1:有碰撞。

3.2.31 ER_GetAxisCurrent



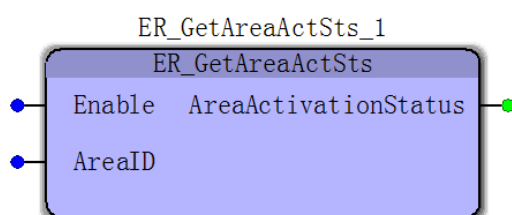
- 功能: 获取机器人每个轴的电流值(安培)。
- 输入参数:
 - Enable(BOOL): Enable 为 TRUE 时, 机器人每个轴的电流值实时刷新。
- 输出参数:
 - AxisCurrent(ERC_REAL_ARRAY): 获取到的机器人每个轴的电流值。
注: 输入 Enable 为 TRUE 时, 各轴电流值实时刷新。

3.2.32 ER_GetConveyorPosition



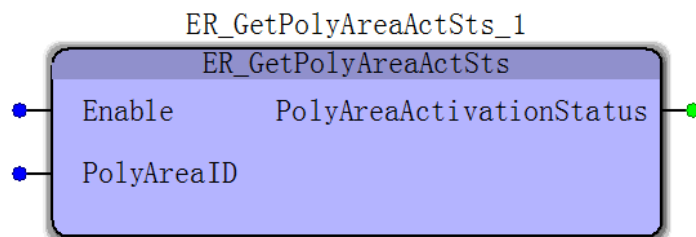
- 功能：获取传送带当前位置。
- 输入参数：
 - ConveyorID (USINT)：传送带 ID 号，ID 号从 1 开始依次递增，最大不超过 4。
 - Enable (BOOL)：使能期间持续获取传送带实际位置。
- 输出参数：
 - Valid (BOOL)：输出 TRUE 表示获取到的位置值有效；
 - Error (BOOL)：获取位置错误；
 - ErrorID (UDINT)：错误 ID；
 - Position (REAL)：获取到的传送带位置。

3.2.33 ER_GetAreaActSts



- 功能：获取区域激活状态。
- 输入参数：
 - Enable (BOOL)：Enable 为 TRUE 时，实时获取区域激活状态。
 - AreaID (UINT)：区域 ID 号，与示教器上区域变量的 ID 号对应，取值范围为 0-31。
- 输出参数：
 - AreaActivationStatus (BOOL)：为 TRUE 时，表示对应区域已激活。

3.2.34 ER_GetPolyAreaActSts



- 功能：获取多边体区域激活状态。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，实时获取多边体区域激活状态。
 - PolyAreaID (UINT): 多边体区域 ID 号，与示教器上多边体区域变量的 ID 号对应，取值范围为 1-4。
- 输出参数：
 - PolyAreaActivationStatus(BOOL): 为 TRUE 时，表示对应多边体区域已激活。

3.3 ER_IO_LIB: IO 功能块

IO 功能块在用户库“ER_IO_LIB”中。该库功能块用于提供用户在 PLC 端对控制系统的物理和虚拟输入输出进行读写和刷新操作。

对于物理数字量 IO 的刷新，最大可支持到 256 路数字量输入和数字量输出，每 32 路为一组。例如 UPDATE_PHY_DI_001_032 代表该功能块实现的是系统第 1 路到第 32 路物理数字量输入 DI 信号的刷新功能。此类功能块的功能以此类推。

对于虚拟数字量 IO 的刷新，最大可支持到 2048 路数字量输入和数字量输出，每 256 路为一组。例如 UPDATE_SIMDI_BYTE_0256 代表该功能块实现的是系统第 1 路到第 256 路虚拟数字量输入 DI 信号的刷新功能。此类功能块的功能以此类推。

IO 功能块统计见表 3.3:

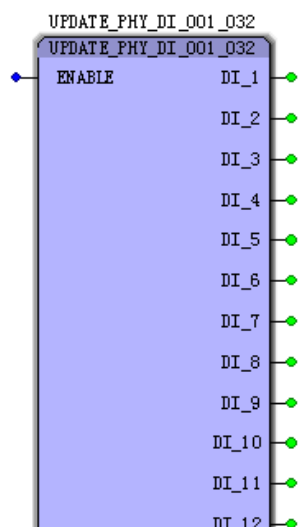
表 3.3 控制器状态获取接口统计

功能块名	实现功能
UPDATE_PHY_DI_aaa_bbb	获取并刷新对应端口的数字量输入。
UPDATE_PHY_DO_aaa_bbb	读写对应端口的数字量输出。
UPDATE_PHY_AI_aaa_bbb	获取并刷新对应端口的模拟量输入。
UPDATE_PHY_AO_aaa_bbb	读写对应端口的模拟量输出。
UPDATE_SIMDI_BYTE_aaaa	读写对应端口的虚拟数字量输入。
UPDATE_SIMDO_BYTE_aaaa	读写对应端口的虚拟数字量输出。
UPDATE_SIM_AI_aaa_bbb	读写对应端口的虚拟模拟量输入。
UPDATE_SIM_AO_aaa_bbb	读写对应端口的虚拟模拟量输出。
PHY_DI_PACK	将一组连续的数字量输入端口的状态以十进制数输出。
PHY_DO_UNPACK	用十进制数设置一组连续的数字量输出端口。
PHY_DO_PACK	将一组连续的数字量输出端口的状态以十进制数输出。

3.3.1 UPDATE_PHY_DI_aaa_bbb

UPDATE_PHY_DI_aaa_bbb 是指 UPDATA_PHY_DI_001_032、UPDATA_PHY_DI_033_064、...、UPDATA_PHY_DI_225_256 功能块的统称，用于刷新对应端口的数字量输入信号。

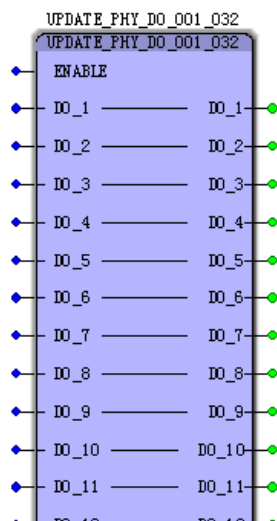
刷新 IO 类输入信号功能块的使用方法，在本小节以功能块 UPDATE_PHY_DI_001_032 为例详细介绍，其他类似功能块均可参考本小节的使用方法。



- 功能：刷新第 1 到第 32 号物理 DI 端口的数字量输入信号。
- 输入参数：
Enable (BOOL): Enable 为 TRUE 时，输出端口的状态实时刷新。
- 输出参数：
DI_1(BOOL): 输出 1 号 DI 端口的状态；
DI_2(BOOL): 输出 2 号 DI 端口的状态；
...
(DI_3-DI_31)
DI_32(BOOL): 输出 32 号 DI 端口的状态；
- 注意：
用户若要在调用本功能块以外的工作单中（程序或功能块）使用某一数字量输入端口做判断条件，则本功能块中该端口对应的输出应链接到一个全局变量，目的是将该端口状态映射到该全局变量以供其他工作单（POU）使用。

3.3.2 UPDATE_PHY_DO_aaa_bbb

UPDATE_PHY_DO_aaa_bbb 是指 UPDATA_PHY_DO_001_032、UPDATA_PHY_DO_033_064、...、UPDATA_PHY_DO_225_256 功能块的统称，用于刷新对应端口的数字量输出信号。



- 功能：刷新第 1 到第 32 号物理 DO 端口的数字量输出信号，通过修改与输入输出变量 DO_1 到 DO_32 连接的变量值，可实现对对应端口状态进行修改。
- 输入参数：
 - Enable (BOOL): Enable 为 TRUE 时，DO 端口的状态实时刷新到与输入输出变量 DO_1-DO_32 链接的变量上；也可将所链接变量的变化输出到对应端口。
- 输入输出参数：
 - DO_1(BOOL): 所链接变量无变化时，输出 1 号 DO 端口的状态到该变量；链接的变量有变化时，将变化输出到对应 DO 端口。
 - DO_2(BOOL): 所链接变量无变化时，输出 1 号 DO 端口的状态到该变量；链接的变量有变化时，将变化输出到对应 DO 端口。
 - ... (DO_3-DO_31)
 - DO_32(BOOL): 所链接变量无变化时，输出 32 号 DO 端口的状态到该变量；链接的变量有变化时，将变化输出到对应 DO 端口。
- 注意：
 - 用户若要在调用本功能块以外的工作单中（程序或功能块）操作某一数字量输出端口，则本功能块中该端口对应的输入输出接口应链接到一个全局变量。
 - 链接到同一端口的输入输出变量必须为同一变量。
 - 其他刷新 IO 类输出信号功能块的使用方法，均可参考本小节的使用方法。

3.3.3 UPDATE_PHY_AI_aaa_bbb

- 功能：用于刷新第 aaa 号到第 bbb 号端口的模拟量输入信号。
- 参数说明与使用方法与功能块 UPDATE_PHY_DI_aaa_bbb 类似（仅输出参数类型不同），参考章节 [3.3.1](#)。

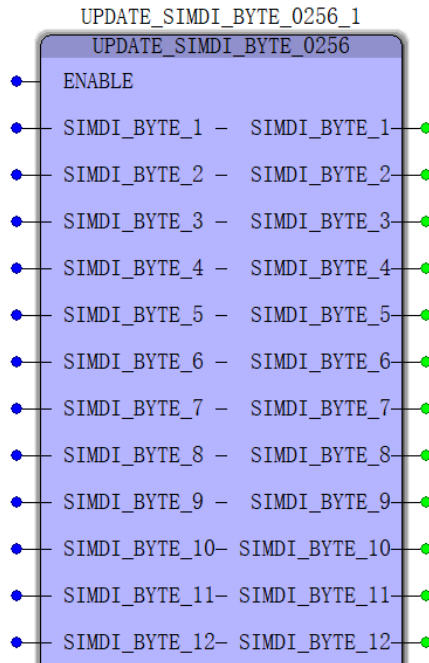
3.3.4 UPDATE_PHY_AO_aaa_bbb

- 功能：用于读写第 aaa 号到第 bbb 号端口的模拟量输出信号。
- 参数说明与使用方法与功能块 UPDATE_PHY_DO_aaa_bbb 类似（仅输入输出参数类型不同），参考章节 [3.3.2](#)。

3.3.5 UPDATE_SIMDI_BYTE_aaaa

UPDATE_SIMDI_BYTE_aaaa 是指 UPDATE_SIMDI_BYTE_0256、UPDATE_SIMDI_BYTE_0512、...、UPDATE_SIMDI_BYTE_2048 功能块的统称，用于刷新对应端口的数字量输入信号。

刷新 IO 类输入信号功能块的使用方法，在本小节以功能块 UPDATE_SIMDI_BYTE_0256 为例详细介绍，其他类似功能块均可参考本小节的使用方法。



- 功能：刷新第 1 到第 256 号虚拟 DI 端口的数字量输入信号，通过修改与输入输出变量 SIMDI_BYTE_1 到 SIMDI_BYTE_32 连接的变量值，可实现对对应端口状态进行修改。（注：该功能块中，每个端口表示 8 路输入）
- 输入参数：

Enable (BOOL): Enable 为 TRUE 时，输入端口的状态的状态实时刷新到与输入输出变量 SIMDI_BYTE_1- SIMDI_BYTE_32 链接的变量上；也可将所链接变量的变化输出到对应端口。
- 输入输出参数：

SIMDI_BYTE_1 (BYTE): 所链接变量无变化时，输出 1-8 号 SIMDI 端口的状态到该变量；链接的变量有变化时，将变化输出到对应 1-8 号 SIMDI 端口；

SIMDI_BYTE_2(BYTE): 所链接变量无变化时, 输出 9-16 号 SIMDI 端口的状态到该变量; 链接的变量有变化时, 将变化输出到对应 9-16 号 SIMDI 端口;

...

(SIMDI_BYTE_3- SIMDI_BYTE_31)

SIMDI_BYTE_32(BYTE): 所链接变量无变化时, 输出 249-256 号 SIMDI 端口的状态到该变量; 链接的变量有变化时, 将变化输出到对应 249-256 号 SIMDI 端口。

- **注意:**

用户若要在调用本功能块以外的工作单中(程序或功能块)使用某一虚拟数字量输入端口做判断条件, 则本功能块中该端口对应的输入输出应链接到一个全局变量, 目的是将该端口状态映射到该全局变量以供其他工作单(POU)使用。

3.3.6 UPDATE_SIMDO_BYTE_aaaa

- 功能: 用于读写到 aaaa 号端口的虚拟数字量输出信号。
- 参数说明与使用方法与功能块 UPDATE_SIMDI_BYTE_aaaa 类似, 参考章节 [3.3.5](#)。

3.3.7 UPDATE_SIM_AI_aaa_bbb

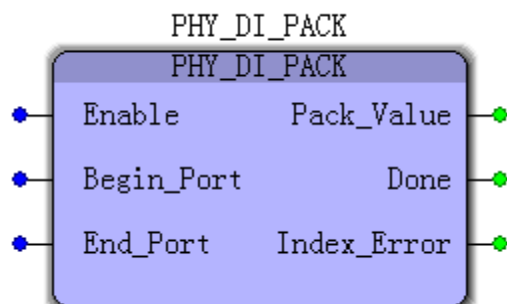
- 功能: 用于刷新第 aaa 号到第 bbb 号端口的虚拟模拟量输入信号。
- 参数说明与使用方法与功能块 UPDATE_PHY_DI_aaa_bbb 类似(仅输出参数类型不同), 参考章节 [3.3.1](#)。

3.3.8 UPDATE_SIM_AO_aaa_bbb

- 功能: 用于读写第 aaa 号到第 bbb 号端口的模拟量输出信号。
- 参数说明与使用方法与功能块 UPDATE_PHY_DO_aaa_bbb 类似(仅输入输出参数类型不同), 参考章节 [3.3.2](#)。

3.3.9 PHY_DI_PACK

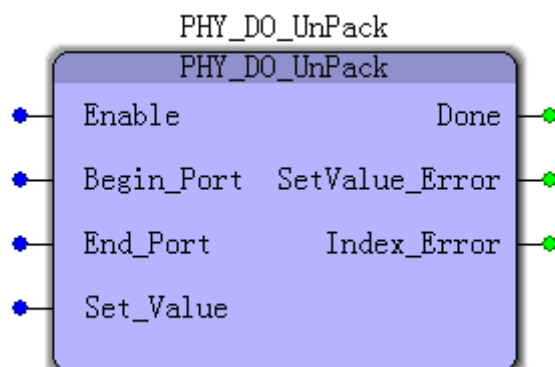
- 功能: 此功能块用于将一组连续的数字量输入端口的状态值视为二进制数(端口数小的端口值视为二进制数低位), 将该二进制数转换为十进制数输出。



- 输入参数：
 - Enable (BOOL): 功能块使能信号, 上升沿触发功能块执行操作。
 - Begin_Port(INT): 起始端口号, 对应二进制数最低位。
 - END_PORT(INT): 结束端口号, 对应二进制数最高位。
- 输出参数：
 - Pack_Value(DINT): 指定连续 DI 端口值转换为十进制后的数值。
 - Done(BOOL): 执行完成标志。
 - Index_Error(BOOL): 输入端口错误标志, 输入 Begin_Port 和 End_Port 有错误时为 TRUE。
- 注意：
 - 最多支持 32 位 DI 的打包输出。
 - 起始端口值应小于结束端口。
 - 1 到 8 位为系统 DI, 不可进行此操作。

3.3.10 PHY_DO_UNPACK

- 功能: 此功能块用于将一个十进制数转化为二进制数后, 用该二进制数设置一组连续的数字量输出端口 (端口数小的端口对应二进制数的低位)。



- 输入参数：
 - Enable (BOOL): 功能块使能信号, 上升沿触发功能块执行操作。
 - Begin_Port(INT): 起始端口号, 对应二进制数最低位。

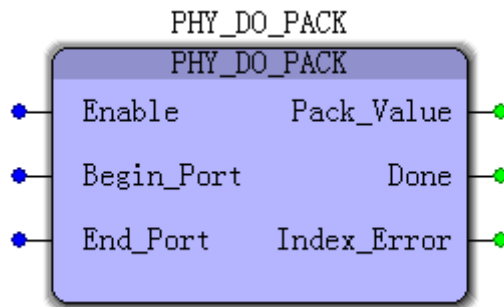
END_PORT(INT):结束端口号，对应二进制数最高位。

Set_Value(UDINT):将要设置的该组 DO 值视为二进制数(端口号小的视为低位)，转成十进制后的数值。

- 输出参数:
 - Done(BOOL):执行完成标志。
 - SetValue_Error(BOOL):输入参数 Set_Value 不合法时为 TRUE。
 - Index_Error(BOOL):输入端口错误标志，输入 Begin_Port 和 End_Port 有错误时为 TRUE。
- 注意:
 - 最多支持 32 位 DO 的打包输出。
 - 起始端口值应小于结束端口。
 - 1 到 8 位为系统 DO，不可进行此操作。

3.3.11 PHY_DO_PACK

- 功能: 此功能块用于将一组连续的数字量输出端口的状态值视为二进制数（端口数小的端口值视为二进制数低位），将该二进制数转换为十进制数输出。



- 输入参数:
 - Enable (BOOL): 功能块使能信号，上升沿触发功能块执行操作。
 - Begin_Port(INT):起始端口号，对应二进制数最低位。
 - END_PORT(INT):结束端口号，对应二进制数最高位。
- 输出参数:
 - Pack_Value(DINT):指定连续 DI 端口值转换为十进制后的数值。
 - Done(BOOL):执行完成标志。
 - Index_Error(BOOL):输入端口错误标志，输入 Begin_Port 和 End_Port 有错误时为 TRUE。
- 注意:
 - 最多支持 32 位 DI 的打包输出。
 - 起始端口值应小于结束端口。
 - 1 到 8 位为系统 DO，不可进行此操作。

3.4 ER_PLC_Share_VAR: PLC 变量功能块

用户库 ER_PLC_Share_VAR 提供了 PLC 与控制器共享变量功能。共享变量包括 4 中类型，分别是 BOOL、REAL、INT、DINT，每种最多支持 300 个变量，提供如下图所示功能块，用户根据需要，选择相应功能块进行调用。

BOOL 类型

- PLCVAR_BOOL_1_30 更新ID为1到30的BOOL类型PLC共享变量。
- PLCVAR_BOOL_121_150 更新ID为121到150的BOOL类型PLC共享变量。
- PLCVAR_BOOL_151_180 更新ID为181到210的BOOL类型PLC共享变量。
- PLCVAR_BOOL_181_210 更新ID为181到210的BOOL类型PLC共享变量。
- PLCVAR_BOOL_211_240 更新ID为211到240的BOOL类型PLC共享变量。
- PLCVAR_BOOL_241_270 更新ID为241到270的BOOL类型PLC共享变量。
- PLCVAR_BOOL_271_300 更新ID为271到300的BOOL类型PLC共享变量。
- PLCVAR_BOOL_31_60 更新ID为31到60的BOOL类型PLC共享变量。
- PLCVAR_BOOL_61_90 更新ID为61到90的BOOL类型PLC共享变量。
- PLCVAR_BOOL_91_120 更新ID为91到120的BOOL类型PLC共享变量。

INT 类型

- PLCVAR_INT_1_30 更新ID为1到30的INT类型PLC共享变量。
- PLCVAR_INT_121_150 更新ID为121到150的INT类型PLC共享变量。
- PLCVAR_INT_151_180 更新ID为151到180的INT类型PLC共享变量。
- PLCVAR_INT_181_210 更新ID为181到210的INT类型PLC共享变量。
- PLCVAR_INT_211_240 更新ID为211到240的INT类型PLC共享变量。
- PLCVAR_INT_241_270 更新ID为241到270的INT类型PLC共享变量。
- PLCVAR_INT_271_300 更新ID为271到300的INT类型PLC共享变量。
- PLCVAR_INT_31_60 更新ID为31到60的INT类型PLC共享变量。
- PLCVAR_INT_61_90 更新ID为61到90的INT类型PLC共享变量。
- PLCVAR_INT_91_120 更新ID为91到120的INT类型PLC共享变量。

DINT 类型

- PLCVAR_DINT_1_30 更新ID为1到30的DINT类型PLC共享变量。
- PLCVAR_DINT_121_150 更新ID为121到150的DINT类型PLC共享变量。
- PLCVAR_DINT_151_180 更新ID为151到180的DINT类型PLC共享变量。
- PLCVAR_DINT_181_210 更新ID为181到210的DINT类型PLC共享变量。
- PLCVAR_DINT_211_240 更新ID为211到240的DINT类型PLC共享变量。
- PLCVAR_DINT_241_270 更新ID为241到270的DINT类型PLC共享变量。
- PLCVAR_DINT_271_300 更新ID为271到300的DINT类型PLC共享变量。
- PLCVAR_DINT_31_60 更新ID为31到60的DINT类型PLC共享变量。
- PLCVAR_DINT_61_90 更新ID为61到90的DINT类型PLC共享变量。
- PLCVAR_DINT_91_120 更新ID为91到120的DINT类型PLC共享变量。

REAL 类型

- PLCVAR_REAL_1_30 更新ID为1到30的REAL类型PLC共享变量。
- PLCVAR_REAL_121_150 更新ID为121到150的REAL类型PLC共享变量。
- PLCVAR_REAL_151_180 更新ID为151到180的REAL类型PLC共享变量。
- PLCVAR_REAL_181_210 更新ID为181到210的REAL类型PLC共享变量。
- PLCVAR_REAL_211_240 更新ID为211到240的REAL类型PLC共享变量。
- PLCVAR_REAL_241_270 更新ID为241到270的REAL类型PLC共享变量。
- PLCVAR_REAL_271_300 更新ID为271到300的REAL类型PLC共享变量。
- PLCVAR_REAL_31_60 更新ID为31到60的REAL类型PLC共享变量。
- PLCVAR_REAL_61_90 更新ID为61到90的REAL类型PLC共享变量。
- PLCVAR_REAL_91_120 更新ID为91到120的REAL类型PLC共享变量。

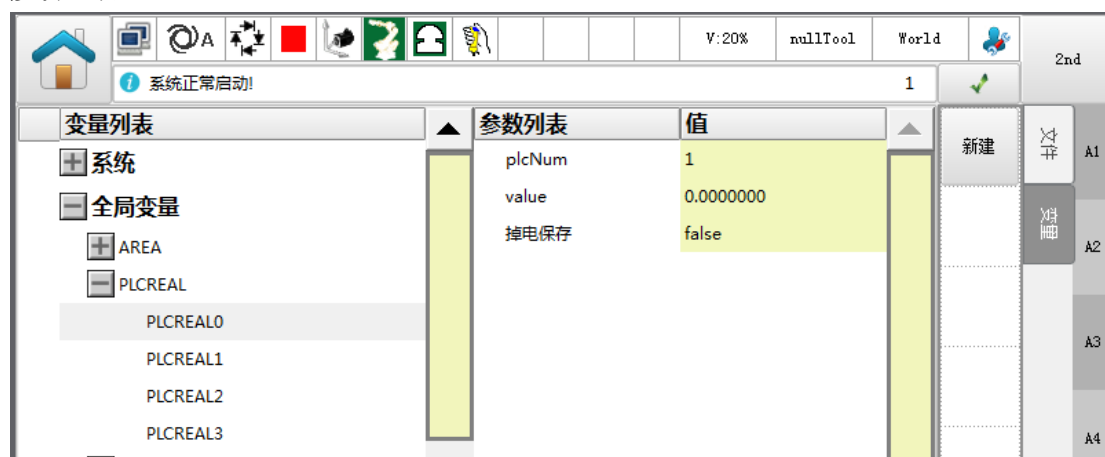
3.4.1 PLCVAR_REAL_1_30

以刷新 REAL 型 1 到 30 号变量功能块“PLCVAR_REAL_1_30”为例，介绍共享变量功能块用法，其他功能块用法以此类推。

功能：刷新同步机器人示教器上创建的 PLCREAL 型变量，根据示教器端 PLCREAL 类型的 plcNum 参数值与对应功能块上的输入输出端口绑定。

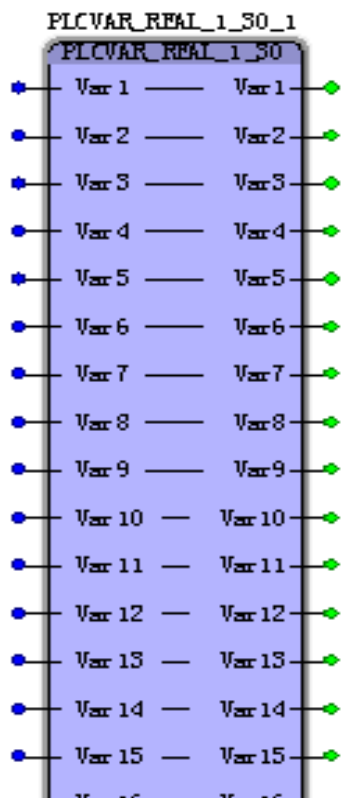
“PLC 变量”具体使用步骤如下：

1、 控制器端创建：首先在控制器端创建需要与 PLC 端共享变量绑定的“PLCREAL”类型变量，并设定好该变量的索引号“plcNum”（当索引号改变后，控制器会提示重启，使修改生效）。



该变量“PLCREAL0”创建后，其 plcNum 参数值修改为 1，重启控制器后，该变量与 PLC 端功能块“PLCVAR_REAL_1_30”的 Var_1 同步。

当这个变量被 PLC 修改时，变化会同步到控制器端对应的“PLCREAL”变量的“value”值；当控制器端对应的“PLCREAL”变量的参数 value 值被修改时，PLC 端对应端口变量的值也会随之改变。



- 输入输出参数：
 - Var_1 (REAL): 对应机器人控制器端 PLCNUM=1 的 PLCREAL 型变量。
 -
 -
 -
 - Var_30 (REAL): 对应机器人控制器端 PLCNUM=30 的 PLCREAL 型变量。
- 注意：

控制器端“PLC 实数变量”创建或属性修改后，需重启控制器后生效。

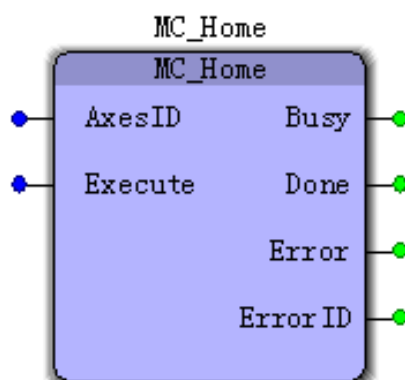
3.5 ER_MC_Control: 单轴控制功能块

ER_MC_Control 用户库提供一系列功能块，用于控制除机器人本体轴以及附加轴以外的单轴运动及状态转换。功能列表如下表。

表 3.4 单轴控制接口统计

功能块名	实现功能
MC_Home	单轴回零。
MC_Power	控制单轴励磁状态。
MC_ReadStatus	获取单轴状态
MC_ReadActualPosition	获取单轴实际位置
MC_ReadAxisError	获取单轴错误信息
MC_Stop	控制单轴停止运动
MC_Reset	清除单轴错误状态
MC_SetOverride	设置单轴速度倍率
MC_MoveAbsolute	控制单轴进行绝对位置运动
MC_MoveRelative	控制单轴进行相对位置运动
MC_MoveVelocity	控制单轴以指定速度运动

3.5.1 MC_Home



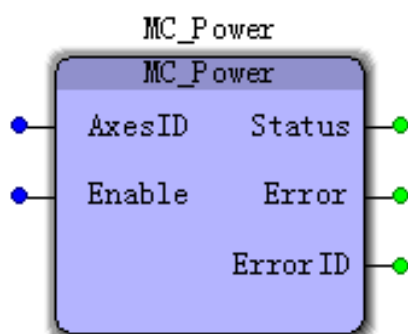
- 功能：单轴回零。
- 输入参数：
 - AxesID (USINT)：轴 ID 号。和伺服从站配置相关，除本体轴及附加轴以外，配置的轴 ID 号从 1 开始依次递增，本节后续功能块不再赘述。
 - Execute (BOOL)：Execute 上升沿触发功能块。
- 输出参数：
 - Busy(BOOL)：执行过程中输出 TRUE。
 - Done(BOOL)：回零完成输出 TRUE。

Error(BOOL):回零出错、失败输出 TRUE。

ErrorID(UDINT): 回零失败错误代号, 代号释义如下

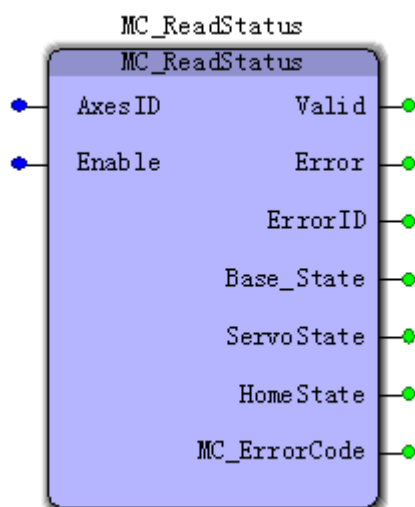
- 1: 伺服未励磁
- 2: 轴运动中
- 3: 控制器读取命令超时
- 4: 控制器端执行命令超时
- 6: 轴状态不满足回零操作, 可能是轴未励磁
- 9: 该轴 ID 号超出限制 (1—12)
- 40002: 控制器端执行失败

3.5.2 MC_Power



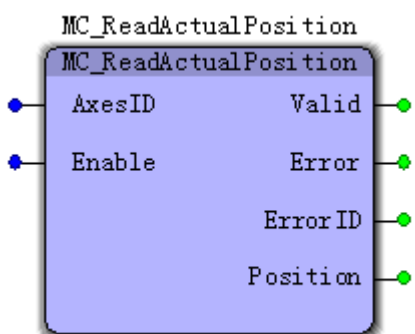
- 功能： 控制单轴励磁状态。
- 输入参数：
 - AxesID (USINT): 该轴 ID 号。
 - Enable (BOOL): 上升沿执行上励磁操作, 下降沿执行掉励磁操作。
- 输出参数：
 - Status(BOOL): 在励磁状态输出 TRUE, 否则 FALSE。
 - Error(BOOL):执行励磁操作出错输出 TRUE。
 - ErrorID(UDINT): 励磁操作失败错误代号, 代号释义如下
 - 2: 轴运动中
 - 3: 控制器读取命令超时
 - 4: 控制器端执行命令超时
 - 9: 该轴 ID 号超出限制 (1—12)
 - 40002: 控制器端执行失败

3.5.3 MC_ReadStatus



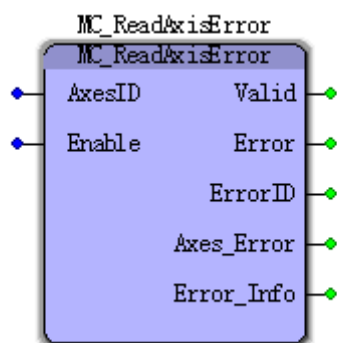
- 功能：读取单轴状态。
- 输入参数：
 - AxesID (USINT)：该轴 ID 号。
 - Enable (BOOL)：使能期间持续获取该轴相关状态。
- 输出参数：
 - Valid (BOOL)：输出 TRUE 表示获取到的状态信息有效。
 - Error (BOOL)：获取状态错误。
 - ErrorID (UDINT)：错误 ID。
 - Base_State (UINT)：轴基本状态，输出 2 表示可执行运动。
 - ServoState (UINT)：轴励磁状态，1 表示励磁，0 表示未励磁。
 - HomeState (UINT)：回零状态，1 表示回零成功，0 表示未回零。
 - MC_ErrorCode (UDINT)：轴错误代号。

3.5.4 MC_ReadActualPosition



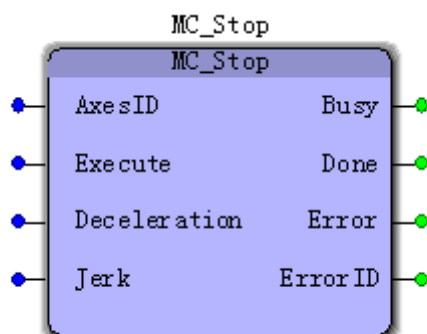
- 功能： 获取轴当前实际位置。
- 输入参数：
 - AxesID (USINT)： 该轴 ID 号。
 - Enable (BOOL)： 使能期间持续获取该轴实际位置。
- 输出参数：
 - Valid (BOOL)： 输出 TRUE 表示获取到的位置值有效。
 - Error (BOOL)： 获取位置错误。
 - ErrorID (UDINT)： 错误 ID。
 - Position (LREAL)： 获取到的该轴实际位置。

3.5.5 MC_ReadAxisError



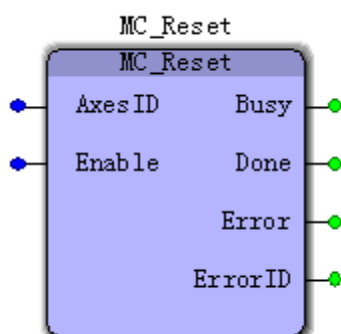
- 功能： 读取轴错误
- 输入参数：
 - AxesID (USINT)： 该轴 ID 号。
 - Enable (BOOL)： 使能期间持续获取轴错误状态。
- 输出参数：
 - Valid (BOOL)： 输出 TRUE 表示获取到的位置值有效。
 - Error (BOOL)： 输出 TRUE 表示获取错误信息出错。
 - ErrorID (UDINT)： 错误 ID。
 - Axes_Error (UDINT)： 该轴错误代号。
 - Error_Info (STRING)： 轴错误信息，仅支持英文显示。

3.5.6 MC_Stop



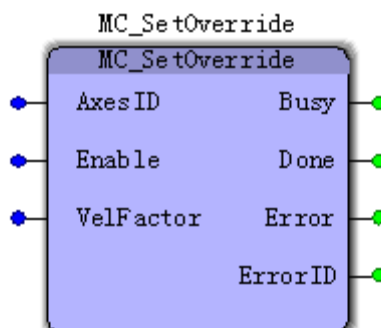
- 功能： 停止轴运动。
- 输入参数：
 - AxesID (USINT)： 该轴 ID 号。
 - Execute (BOOL)： 上升沿执行停止操作。
 - Deceleration (REAL)： 停止减速度，设定值越小，减速越慢，停止用时越长，不输入则使用系统默认值。
 - Jerk (REAL)： 停止加加速度，设定值越小，减速越慢，停止用时越长。不输入则使用系统默认值。
- 输出参数：
 - Busy (BOOL)： 执行过程中输出 TRUE。
 - Done (BOOL)： 执行完成输出 TRUE。
 - Error (BOOL)： 执行失败输出 TRUE。
 - ErrorID (UDINT)： 执行出错的错误代号：
 - 3： 控制器读取命令超时
 - 4： 控制器端执行命令超时
 - 9： 该轴 ID 号超出限制（1—12）
 - 40002： 控制器端执行失败

3.5.7 MC_Reset



- 功能：清除轴错误。
- 输入参数：
 - AxesID (USINT)：该轴 ID 号。
 - Enable (BOOL)：上升沿执行清除错误操作。
- 输出参数：
 - Busy (BOOL)：执行过程中输出 TRUE。
 - Done (BOOL)：执行完成输出 TRUE。
 - Error (BOOL)：执行失败输出 TRUE。
 - ErrorID (UDINT)：执行出错的错误代号：
 - 3：控制器读取命令超时
 - 4：控制器端执行命令超时
 - 9：该轴 ID 号超出限制（1—12）
 - 40002：控制器端执行失败。

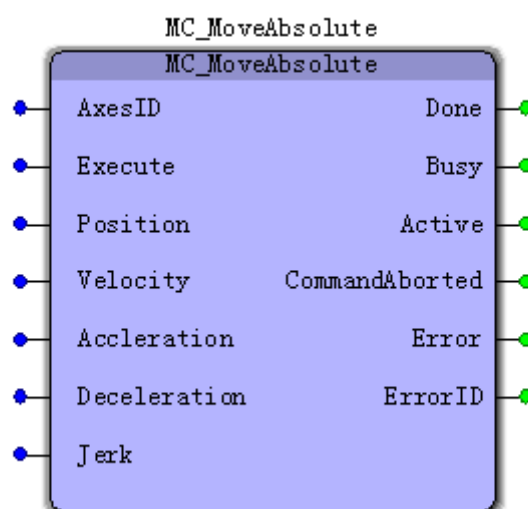
3.5.8 MC_SetOverride



- 功能：设置轴速度倍率

- 输入参数：
 - AxesID (USINT)：该轴 ID 号。
 - Enable (BOOL)：上升沿执行设置操作。
 - VelFactor (REAL)：待设置的倍率值，范围 1.0—100.0，倍率越大，速度越快。
- 输出参数：
 - Busy (BOOL)：执行过程中输出 TRUE。
 - Done (BOOL)：执行完成输出 TRUE。
 - Error (BOOL)：执行失败输出 TRUE。
 - ErrorID (UDINT)：执行出错的错误代号：
 - 3：控制器读取命令超时
 - 4：控制器端执行命令超时
 - 9：该轴 ID 号超出限制（1—12）
 - 40002：控制器端执行失败

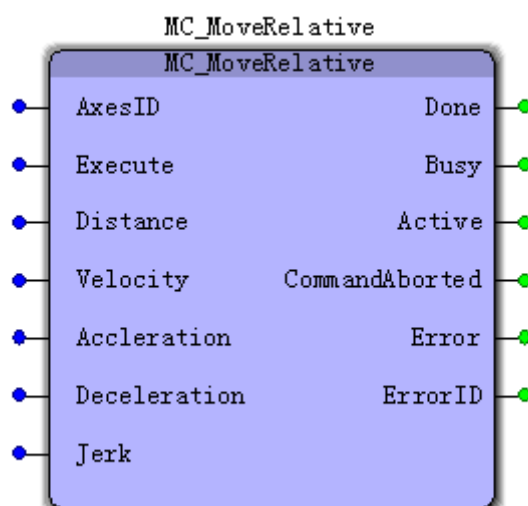
3.5.9 MC_MoveAbsolute



- 功能：轴绝对运动。
- 输入参数：
 - AxesID (USINT)：该轴 ID 号。
 - Execute (BOOL)：使能信号，上升沿功能块开始执行。
 - Position (LREAL)：目标位置,根据给定正负值区别运动方向。
 - Velocity (LREAL)：运动到目标位置的速度值，单位为 “°/s”。
 - Acceleration (LREAL)：加速度，无输入则采用默认值
 - Deceleration (LREAL)：减速度，无输入则采用默认值。
 - Jerk (LREAL)：加加速度，无输入采用默认值。
- 输出参数：

- Done (BOOL): 运动执行完成标志。
- Busy (BOOL): 运动指令开始处理。
- Active (BOOL): 运动指令开始执行。
- CommandAborted (BOOL): 运动指令中途被打断或停止。
- Error (UDINT): 运动过程中该轴出错。
- ErrorID (UDINT): 错误 ID 号。执行出错的错误代号释义如下:
- 8:轴未励磁
 - 9: 轴 ID 号超出限制 (1—12)
 - 10: 运动内核状态不对
 - 40001: 控制器端执行失败

3.5.10 MC_MoveRelative

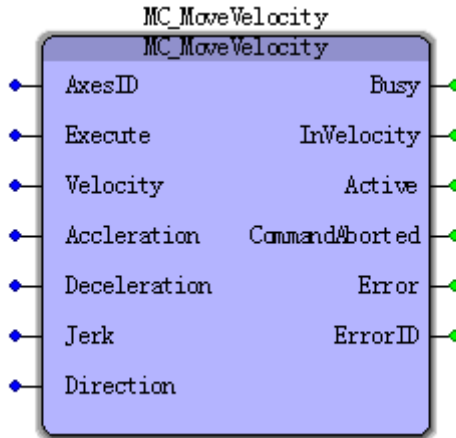


- 功能: 轴相对运动。
- 输入参数:
 - AxesID (USINT): 轴 ID 号。
 - Execute (BOOL): 使能信号, 上升沿功能块开始执行。
 - Distance (LREAL): 目标距离, 根据给定正负值区别运动方向。
 - Velocity (LREAL): 运动到目标距离的速度值, 单位为 “°/s”。
 - Acceleration (LREAL): 加速度, 无输入则采用默认值
 - Deceleration (LREAL): 减速度, 无输入则采用默认值。
 - Jerk (LREAL): 加加速度, 无输入采用默认值。
- 输出参数:
 - Done (BOOL): 运动执行完成标志。
 - Busy (BOOL): 运动指令开始处理。
 - Active (BOOL): 运动指令开始执行。
 - CommandAborted (BOOL): 运动指令中途被打断或停止。
 - Error (BOOL): 运动过程中轴状态出错。

ErrorID (UDINT): 错误 ID 号, 执行出错的错误代号释义如下:

- 8: 轴未励磁
- 9: 轴 ID 号超出限制 (1—12)
- 10: 运动内核状态不对
- 40001: 控制器端执行失败

3.5.11 MC_MoveVelocity



- 功能: 控制轴按照指定速度持续运动 (仅限连续旋转轴)。

- 输入参数:

AxesID(USINT): MC 轴 ID 号

Execute(BOOL): 触发信号, 上升沿触发运动。

Velocity(LREAL): 目标速度值, 大于 0.0(Target velocity)

Acceleration(LREAL): 最大加速度, 大于或等于 0.0(缺省则使用系统默认值)

Deceleration(LREAL): 最大减速度, 大于或等于 0.0(缺省则使用系统默认值)

Jerk(LREAL): 最大加加速度, 大于或等于 0.0(缺省则使用系统默认值)

Direction(USINT): 运动方向 1(正向)、-1(反向), 不接受其他输入值

- 输出参数:

InVelocity(BOOL): 到达指定速度。(本功能块被打断或者 Execute 下降沿, 将被复位)

Busy(BOOL): 功能块正在执行。

Active(BOOL): 轴在本功能块的控制下运动。

CommandAborted(BOOL): 被其他命令中断时输出 TRUE, 目前仅 MC_Stop 可以触发。

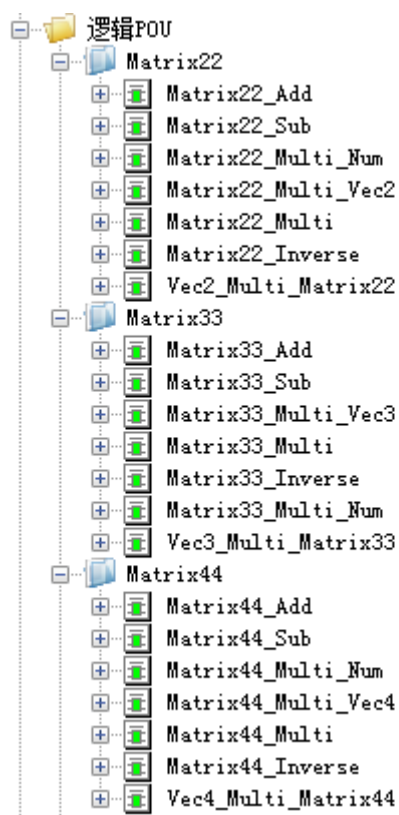
Error(BOOL): 功能块发生错误。

ErrorID(UDINT) : 错误码:

- 5: Acceleration、Deceleration 或 Jerk 输入值不得小于 0.0。
- 6: 不接受的 Direction 参数值, 只能为 -1 或 1。
- 8: 轴在运动状态不允许调用 MC_MoveVelocity 功能。
- 9: MC 轴 ID 号超出限制 (1—12)。
- 10: 运动内核状态异常。

3.6 ER_Calculation: 矩阵运算功能块

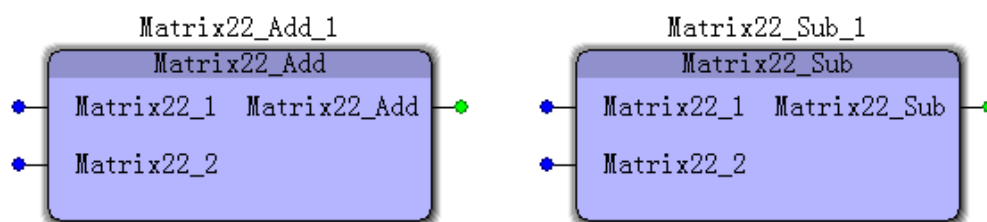
矩阵运算共涉及二、三、四阶的矩阵运算，功能块列表如下：



本章以二阶矩阵运算为例介绍功能来用法，三阶四阶参考二阶。

3.6.1 矩阵加减

计算两个矩阵相加减，功能块如下图：



参数说明（以加法为例）：

输入：

Matrix22_1 (Matrix22): 二阶矩阵 1。

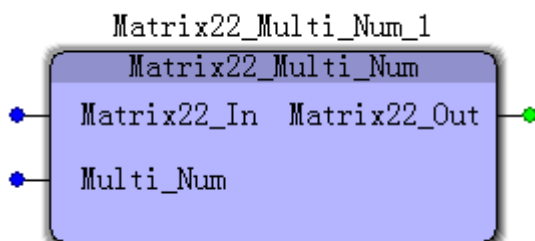
Matrix22_2 (Matrix22): 二阶矩阵 2。

输出：

Matrix22_Add (Matrix22): 结果二阶矩阵。

3.6.2 矩阵数乘

计算一个矩阵乘一个常数，功能块如下图：



参数说明：

输入：

Matrix22_IN (Matrix22)： 输入二阶矩阵。

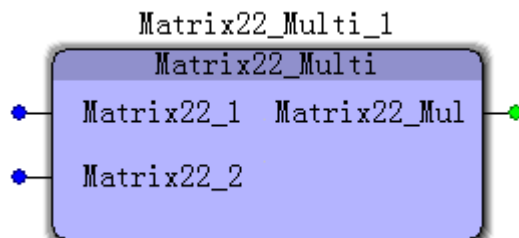
Multi_Num (REAL)： 乘数。

输出：

Matrix22_Out (Matrix22)： 结果二阶矩阵。

3.6.3 矩阵乘法

计算两个矩阵相乘，功能块如下图：



参数说明：

输入：

Matrix22_1 (Matrix22)： 输入矩阵 1。

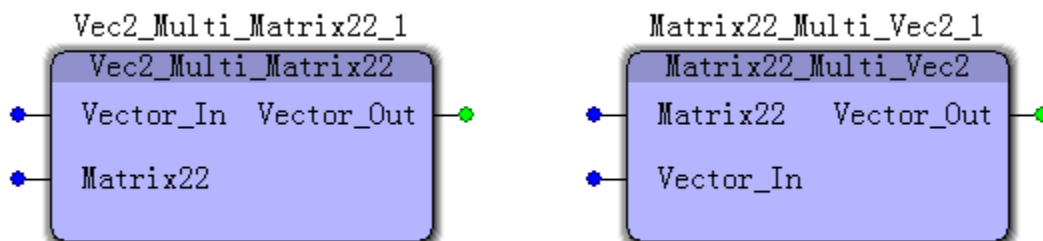
Matrix22_2 (Matrix22)： 输入矩阵 2。

输出：

Matrix22_Mul (Matrix22)： 结果二阶矩阵。

3.6.4 向量乘矩阵

计算向量与矩阵相乘，分为向量右乘矩阵和向量左乘矩阵两个功能块，功能块如下图（输入在上面的操作数在乘号左侧）。



参数说明（以 Matrix22_Multi_Vec2 为例）：

输入：

Matrix22 (Matrix22)： 输入二阶矩阵。

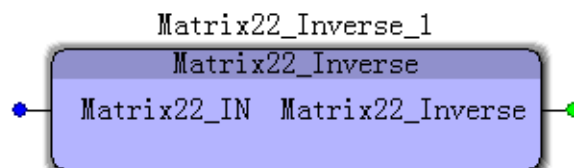
Vector_In (Vector2)： 输入二维向量。

输出：

Vector_Out (Vector2)： 结果二维向量。

3.6.5 矩阵求逆

计算二阶矩阵的逆矩阵，功能块如下图。



参数说明：

输入：

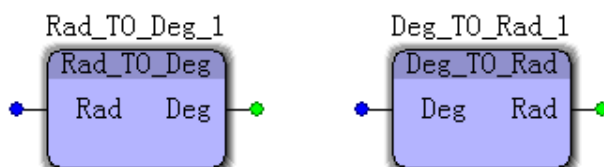
Matrix22_IN (Matrix22)： 输入矩阵。

输出：

Matrix22_Inverse (Matrix22)： 逆矩阵。

3.6.7 角度弧度互转功能块

角度弧度互转功能块如下图所示。



参数说明（以 Rad_TO_Deg 为例）：

输入：

Rad (LREAL)： 弧度值。

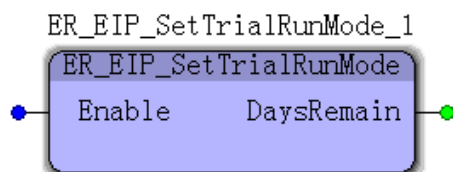
输出：

Deg (LREAL)： 角度值。

3.7 ER_EtherNetIP: EtherNet/IP 通讯配置功能块

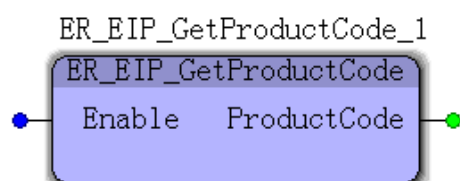
ER_EtherNetIP 功能库用于 PLC 和外部设备通过 EtherNet/IP 协议进行通讯。本功能具体使用方法请参考手册《ER 系列工业机器人 EtherNet/IP 调试手册》，本小节仅介绍功能块用法。

3.7.1 ER_EIP_SetTrialRunMode



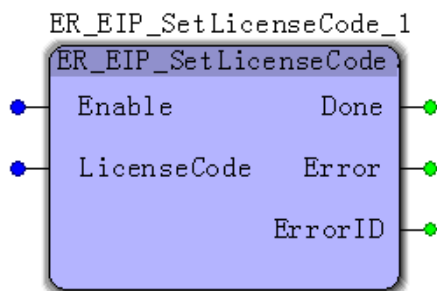
- 功能：用于开启试用版 EtherNet/IP 功能。试用期限为 30 天（从第一次调用过此功能块开始计时），超过期限将自动关闭 EtherNet/IP 功能。
- 输入参数：
 - Enable(BOOL)：为 TRUE 开启试用，并输出试用剩余天数。
- 输出参数：
 - DaysRemain(UINT)：输入 Enable 为 TRUE 时，输出试用剩余天数

3.7.2 ER_EIP_GetProductCode



- 功能：获取产品序列号，需要将此序列号发送至厂家，以获取相应的 LicenseCode（许可码）。
- 输入参数：
 - Enable(BOOL)：输入为 TRUE 时获取产品。
- 输出参数：
 - ProductCode(DINT)：输出产品序列号，发送至厂家以获取相应的 LicenseCode。

3.7.3 ER_EIP_SetLicenseCode



- 功能：将由厂家获取的产品许可码注册到 EtherNet/IP 模块中，注册成功后重新启动控制器生效。

- 输入参数：

Enable(BOOL)：输入为 TRUE 时注册 EtherNet/IP 模块。

LicenseCode(DINT)：输入由厂家获取到的产品许可码。

- 输出参数：

Done(BOOL):输出 TRUE 表示注册成功。

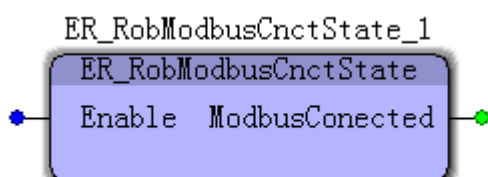
Error(BOOL):错误，输出 TRUE 表示激活失败。

ErrorID(UDINT):错误码。

3.8 ER_ModbusTcp: ModbusTCP 功能库

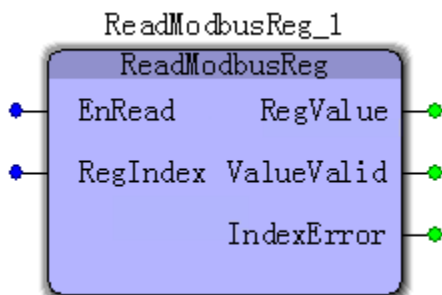
ER_ModbusTcp 功能库提供了读写 Modbus 寄存器的功能。可读写索引号为 101-1500 的寄存器的值(控制器 V1.33 及以上版本, 通用读写区域索引号范围改为 501-1500, 详细可见 ModBusTCP 接口调试手册), 可用于与机器人控制器、外部 Modbus 设备进行数据交互。

3.8.1 ER_RobModbusCnctState



- 功能: 获取标准 ModbusTCP 连接状态。
- 输入参数:
 - Enable(BOOL): 为 TRUE 期间持续获取 ModbusTcp 连接状态。
- 输出参数:
 - ModbusConcted(BOOL): 为 TRUE 表示 Modbus 已连接。

3.8.2 ReadModbusReg

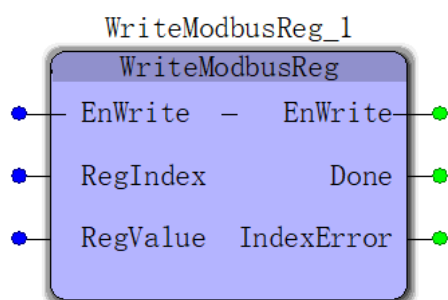


- 功能: 读取指定 Modbus 寄存器的值。
- 输入参数:
 - EnRead (BOOL): 为 TRUE 期间持续获取寄存器的值。
 - RegIndex (INT): 寄存器 ID 号, 支持 101-1500 范围内输入值。
- 输出参数:
 - RegValue (WORD): 寄存器的值。

ValueValid (BOOL): 为 TRUE 表示输出值有效。

IndexError (BOOL): 为 TRUE 表示索引号超范围。

3.8.3 WriteModbusReg



- 功能： 写指定 Modbus 寄存器的值。
- 输入参数：
 - RegIndex (INT): 要写入的寄存器 ID, 支持 101-1500 范围输入值。
 - RegValue (WORD): 待写入寄存器的值。
- 输出参数：
 - Done (BOOL): 为 TRUE 表示写入完成。
 - IndexError (BOOL): 为 TRUE 表示索引号超范围。
- 输入输出参数：
 - EnWrite (BOOL): 上升沿执行将值写入指定寄存器操作。

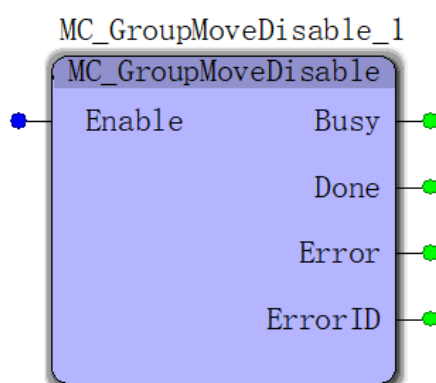
3.9 ER_MC_GroupControl: 机器人运动控制功能块

ER_MC_GroupControl 用户库提供一系列功能块，用于控制机器人本体轴运动。功能列表如下表。

表 3.5 运动功能控制接口统计

功能块名	实现功能
MC_GroupMoveDisable	关闭机器人运动指令功能
MC_GroupMoveEnable	开启机器人运动指令功能。
MC_GroupContinue	继续机器人运动
MC_GroupPause	暂停机器人运动
MC_GroupStop	停止机器人运动
MC_MoveCircularAbsolute	机器人圆弧运动到指定位置(MovC)
MC_MoveDirectAbsolute	机器人关节运动到指定位置(MovJ)
MC_MoveLinearAbsolute	机器人直线运动到指定位置(MovL)
MC_MoveArch	机器人点到点拱形运动到指定位置(MovArch)
MC_MoveArchLinear	机器人直线拱形运动到指定位置(MovLArch)

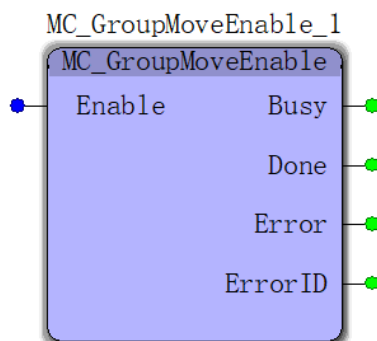
3.9.1 MC_GroupMoveDisable



- 功能：关闭机器人运动指令功能。
 - 输入参数：
 - Enable (BOOL)：上升沿有效，执行功能块。
 - 输出参数：
 - Busy (BOOL)：为 TRUE 表示正在执行。
 - Done (BOOL)：为 TRUE 表示执行完成。
 - Error (BOOL)：为 TRUE 表示执行出错。
 - ErrorID (INT)：正常为 0，出错时输出错误 ID 号。
- 错误代号：
- 1：内存上锁，命令执行失败。

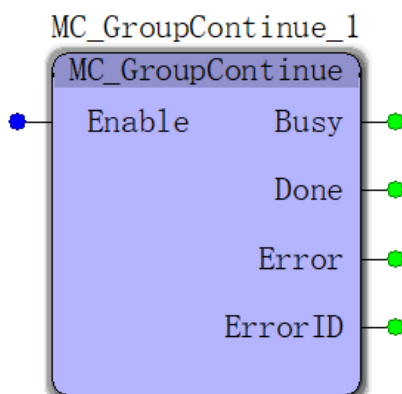
- 2: 输入类型转换不成功, 命令下发失败。
- 3: 控制器执行错误或超时。

3.9.2 MC_GroupMoveEnable



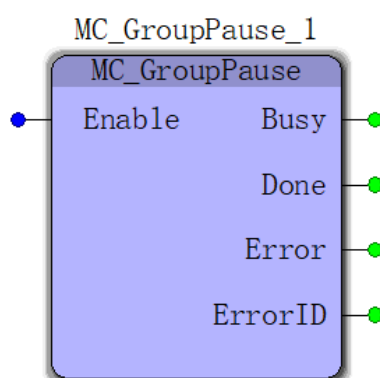
- 功能: 开启机器人运动指令功能。
 - 输入参数:
 - Enable (BOOL): 上升沿有效, 执行功能块。
 - 输出参数:
 - Busy (BOOL): 为 TRUE 表示正在执行。
 - Done (BOOL): 为 TRUE 表示执行完成。
 - Error (BOOL): 为 TRUE 表示执行出错。
 - ErrorID (INT): 正常为 0, 出错时输出错误 ID 号。
- 错误代号:
- 1: 内存上锁, 命令执行失败。
 - 2: 输入类型转换不成功, 命令下发失败。
 - 3: 控制器执行错误或超时。

3.9.3 MC_GroupContinue



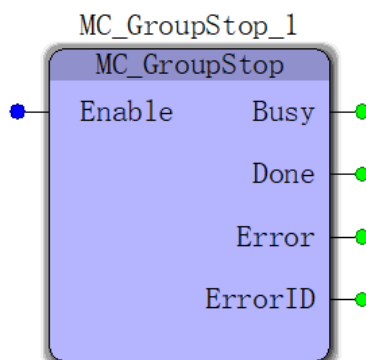
- 功能：继续机器人运动。
- 输入参数：
 - Enable (BOOL)：上升沿有效，执行功能块。
- 输出参数：
 - Busy (BOOL)：为 TRUE 表示正在执行。
 - Done (BOOL)：为 TRUE 表示执行完成。
 - Error (BOOL)：为 TRUE 表示执行出错。
 - ErrorID (INT)：正常为 0，出错时输出错误 ID 号。错误代号：
 - 1：内存上锁，命令执行失败。
 - 2：输入类型转换不成功，命令下发失败。
 - 3：控制器执行错误或超时。
 - 4：运动功能未使能。

3.9.4 MC_GroupPause



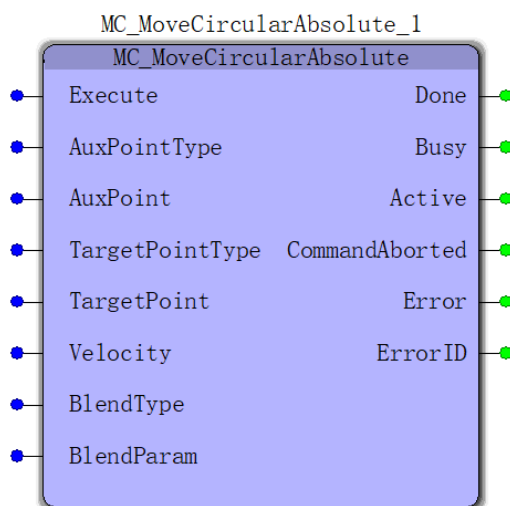
- 功能：暂停机器人运动。
- 输入参数：
 - Enable (BOOL)：上升沿有效，执行功能块。
- 输出参数：
 - Busy (BOOL)：为 TRUE 表示正在执行。
 - Done (BOOL)：为 TRUE 表示执行完成。
 - Error (BOOL)：为 TRUE 表示执行出错。
 - ErrorID (INT)：正常为 0，出错时输出错误 ID 号。错误代号：
 - 1：内存上锁，命令执行失败。
 - 2：输入类型转换不成功，命令下发失败。
 - 3：控制器执行错误或超时。
 - 4：运动功能未使能。

3.9.5 MC_GroupStop



- 功能：停止机器人运动。
 - 输入参数：
 - Enable (BOOL)：上升沿有效，执行功能块。
 - 输出参数：
 - Busy (BOOL)：为 TRUE 表示正在执行。
 - Done (BOOL)：为 TRUE 表示执行完成。
 - Error (BOOL)：为 TRUE 表示执行出错。
 - ErrorID (INT)：正常为 0，出错时输出错误 ID 号。
- 错误代号：
- 1：内存上锁，命令执行失败。
 - 2：输入类型转换不成功，命令下发失败。
 - 3：控制器执行错误或超时。
 - 4：运动功能未使能。

3.9.6 MC_MoveCircularAbsolute



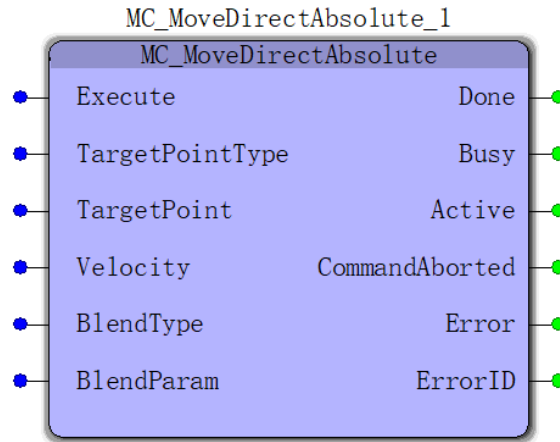
- 功能：机器人圆弧运动到指定位置(MovC)。
- 输入参数：
 - Execute (BOOL)：上升沿执行运动功能。
 - AuxPointType(UINT)：圆弧中间点类型，0：APOS，1：CPOS (当输入其他值时，按 APOS 处理)。
 - AuxPoint (POS_DATA_ARRAY)：圆弧中间点，缺省时点位值默认都按 0 处理。(大小为 23 的数组，0-6:表示 confdata 值，7-22:表示点位值，当为 APOS 时仅 7-22 有效)。
 - TargetPointType(UINT)：目标点类型，0：APOS，1：CPOS (当输入其他值时，按 APOS 处理)。
 - TargetPoint (POS_DATA_ARRAY)：目标点，缺省时点位值默认都按 0 处理。(大小为 23 的数组，0-6:表示 confdata 值，7-22:表示点位值，当为 APOS 时仅 7-22 有效)。
 - Velocity (SPEED)：运动速度，缺省时采用默认值。
 - BlendType (UDINT)：过渡类型，0：无过渡，1：相对过渡，2：绝对过渡，缺省时无过渡。
 - BlendParam (ZONE)：过渡参数，缺省时无过渡。

注：目标点类型、过渡类型与过渡参数变量的结构体定义，可在该库下的数据类型 DataTypes 中查看。
- 输出参数：
 - Done (BOOL)：为 TRUE 表示运动执行到位完成。
 - Busy (BOOL)：为 TRUE 表示正在等待响应运动中。
 - Active (BOOL)：为 TRUE 表示正在执行运动中。
 - CommandAborted (BOOL)：为 TRUE 表示运动指令中途被命令或错误打断。
 - Error (BOOL)：为 TRUE 表示执行出错。
 - ErrorID (INT)：正常为 0，出错时输出错误 ID 号。

错误代号：

 - 1：机器人未上励磁
 - 2：运动内核有报警，命令执行失败
 - 3：启动内核运行失败
 - 4：运动段等待下发过程中产生报警
 - 5：运动段等待下发过程中内核状态异常
 - 6：当前运动功能未使能或处理运动段过程中产生报警
 - 7：等待运动到位过程中产生报警
 - 8：暂停状态中产生报警
 - 9：机器人运动功能未使能
 - 10：运动内核状态异常，非 idle 状态
 - 11：启动运动内核超时

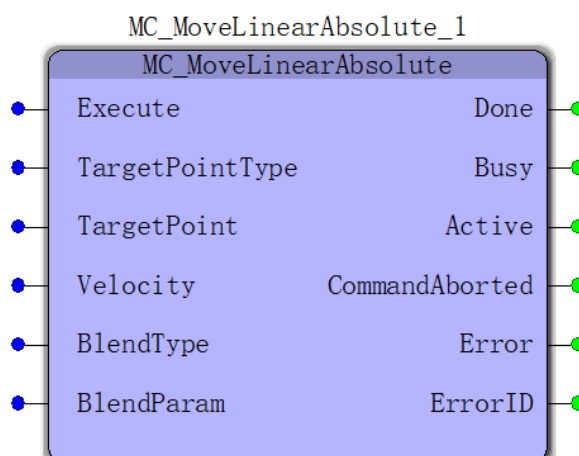
3.9.7 MC_MoveDirectAbsolute



- 功能：机器人关节运动到指定位置(MovJ)。
- 输入参数：
 - Execute (BOOL)：上升沿执行运动功能。
 - TargetPointType(UINT)：目标点类型，0: APOS, 1: CPOS (当输入其他值时，按 APOS 处理)。
 - TargetPoint (POS_DATA_ARRAY)：目标点，缺省时点位值默认都按 0 处理。(大小为 23 的数组，0-6:表示 confdata 值，7-22:表示点位值，当为 APOS 时仅 7-22 有效)。
 - Velocity (SPEED)：运动速度，缺省时采用默认值。
 - BlendType (UDINT)：过渡类型，0: 无过渡，1: 相对过渡，2: 绝对过渡，缺省时无过渡。
 - BlendParam (ZONE)：过渡参数，缺省时无过渡。
- 输出参数：
 - Done (BOOL)：为 TRUE 表示运动执行到位完成。
 - Busy (BOOL)：为 TRUE 表示正在等待响应运动中。
 - Active (BOOL)：为 TRUE 表示正在执行运动中。
 - CommandAborted (BOOL)：为 TRUE 表示运动指令中途被命令或错误打断。
 - Error (BOOL)：为 TRUE 表示执行出错。
 - ErrorID (INT)：正常为 0，出错时输出错误 ID 号。
 错误代号：
 - 1: 机器人未上励磁
 - 2: 运动内核有报警，命令执行失败
 - 3: 启动内核运行失败
 - 4: 运动段等待下发过程中产生报警
 - 5: 运动段等待下发过程中内核状态异常
 - 6: 当前运动功能未使能或处理运动段过程中产生报警
 - 7: 等待运动到位过程中产生报警

- 8: 暂停状态中产生报警
- 9: 机器人运动功能未使能
- 10: 运动内核状态异常, 非 idle 状态
- 11: 启动运动内核超时

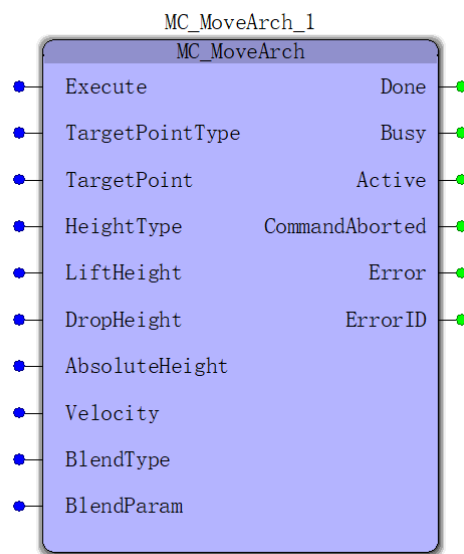
3.9.8 MC_MoveLinearAbsolute



- 功能：机器人直线运动到指定位置(MovL)。
- 输入参数：
 - Execute (BOOL)：上升沿执行运动功能。
 - TargetPointType(UINT)：目标点类型，0: APOS, 1: CPOS (当输入其他值时，按 APOS 处理)。
 - TargetPoint (POS_DATA_ARRAY)：目标点，缺省时点位值默认都按 0 处理。(大小为 23 的数组，0-6:表示 confdata 值，7-22:表示点位值，当为 APOS 时仅 7-22 有效)。
 - Velocity (SPEED)：运动速度，缺省时采用默认值。
 - BlendType (UDINT)：过渡类型，0: 无过渡，1: 相对过渡，2: 绝对过渡，缺省时无过渡。
 - BlendParam (ZONE)：过渡参数，缺省时无过渡。
- 输出参数：
 - Done (BOOL)：为 TRUE 表示运动执行到位完成。
 - Busy (BOOL)：为 TRUE 表示正在等待响应运动中。
 - Active (BOOL)：为 TRUE 表示正在执行运动中。
 - CommandAborted (BOOL)：为 TRUE 表示运动指令中途被命令或错误打断。
 - Error (BOOL)：为 TRUE 表示执行出错。
 - ErrorID (INT)：正常为 0，出错时输出错误 ID 号。
 错误代号：
 - 1: 机器人未上励磁
 - 2: 运动内核有报警，命令执行失败

- 3: 启动内核运行失败
- 4: 运动段等待下发过程中产生报警
- 5: 运动段等待下发过程中内核状态异常
- 6: 当前运动功能未使能或处理运动段过程中产生报警
- 7: 等待运动到位过程中产生报警
- 8: 暂停状态中产生报警
- 9: 机器人运动功能未使能
- 10: 运动内核状态异常, 非 idle 状态
- 11: 启动运动内核超时

3.9.9 MC_MoveArch



- 功能: 机器人点到点拱形运动到指定位置(MovArch)。
- 输入参数:
 - Execute (BOOL): 上升沿执行运动功能。
 - TargetPointType(UINT): 目标点类型, 0: APOS, 1: CPOS (当输入其他值时, 按 APOS 处理)。
 - TargetPoint (POS_DATA_ARRAY): 目标点, 缺省时点位值默认都按 0 处理。(大小为 23 的数组, 0-6:表示 confdata 值, 7-22:表示点位值, 当为 APOS 时仅 7-22 有效)。
 - HeightType (UINT): 高度类型, 0: 相对, 1: 绝对 (当输入其他值时, 按相对处理)。
 - LiftHeight (REAL): 抬升高度 (当高度类型选择相对时有效)。
 - DropHeight (REAL): 下降高度 (当高度类型选择相对时有效)。
 - AbsoluteHeight (REAL): 绝对高度 (当高度类型选择绝对时有效)。
 - Velocity (SPEED): 运动速度, 缺省时采用默认值。
 - BlendType (UDINT): 过渡类型, 0: 无过渡, 1: 相对过渡, 2: 绝对过渡,

缺省时无过渡。

BlendParam (ZONE): 过渡参数, 缺省时无过渡。

- 输出参数:

Done (BOOL): 为 TRUE 表示运动执行到位完成。

Busy (BOOL): 为 TRUE 表示正在等待响应运动中。

Active (BOOL): 为 TRUE 表示正在执行运动中。

CommandAborted (BOOL): 为 TRUE 表示运动指令中途被命令或错误打断。

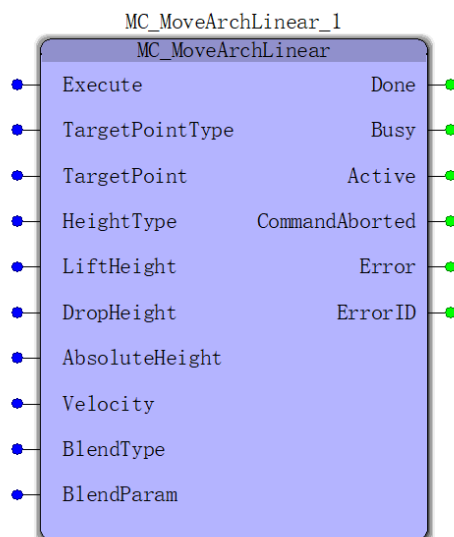
Error (BOOL): 为 TRUE 表示执行出错。

ErrorID (INT): 正常为 0, 出错时输出错误 ID 号。

错误代号:

- 1: 机器人未上励磁
- 2: 运动内核有报警, 命令执行失败
- 3: 启动内核运行失败
- 4: 运动段等待下发过程中产生报警
- 5: 运动段等待下发过程中内核状态异常
- 6: 当前运动功能未使能或处理运动段过程中产生报警
- 7: 等待运动到位过程中产生报警
- 8: 暂停状态中产生报警
- 9: 机器人运动功能未使能
- 10: 运动内核状态异常, 非 idle 状态
- 11: 启动运动内核超时

3.9.10 MC_MoveArchLinear



- 功能: 机器人直线拱形运动到指定位置(MovLArch)。

- 输入参数:

Execute (BOOL): 上升沿执行运动功能。

TargetPointType(UINT): 目标点类型, 0: APOS, 1: CPOS (当输入其他值时,

按 APOS 处理)。

TargetPoint (POS_DATA_ARRAY): 目标点, 缺省时点位值默认都按 0 处理。(大小为 23 的数组, 0-6:表示 confdata 值, 7-22:表示点位值, 当为 APOS 时仅 7-22 有效)。

HeightType (UINT): 高度类型, 0: 相对, 1: 绝对 (当输入其他值时, 按相对处理)。

LiftHeight (REAL): 抬升高度 (当高度类型选择相对时有效)。

DropHeight (REAL): 下降高度 (当高度类型选择相对时有效)。

AbsoluteHeight (REAL): 绝对高度 (当高度类型选择绝对时有效)。

Velocity (SPEED): 运动速度, 缺省时采用默认值。

BlendType (UDINT): 过渡类型, 0: 无过渡, 1: 相对过渡, 2: 绝对过渡, 缺省时无过渡。

BlendParam (ZONE): 过渡参数, 缺省时无过渡。

- 输出参数:

Done (BOOL): 为 TRUE 表示运动执行到位完成。

Busy (BOOL): 为 TRUE 表示正在等待响应运动中。

Active (BOOL): 为 TRUE 表示正在执行运动中。

CommandAborted (BOOL): 为 TRUE 表示运动指令中途被命令或错误打断。

Error (BOOL): 为 TRUE 表示执行出错。

ErrorID (INT): 正常为 0, 出错时输出错误 ID 号。

错误代号:

- 1: 机器人未上励磁
- 2: 运动内核有报警, 命令执行失败
- 3: 启动内核运行失败
- 4: 运动段等待下发过程中产生报警
- 5: 运动段等待下发过程中内核状态异常
- 6: 当前运动功能未使能或处理运动段过程中产生报警
- 7: 等待运动到位过程中产生报警
- 8: 暂停状态中产生报警
- 9: 机器人运动功能未使能
- 10: 运动内核状态异常, 非 idle 状态
- 11: 启动运动内核超时

四、固件库功能块介绍

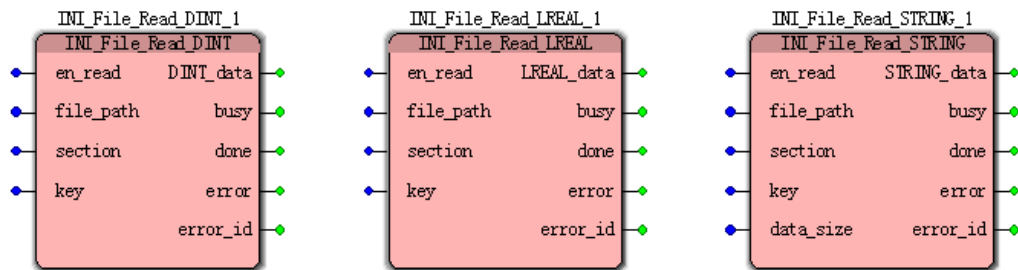
本章介绍部分固件库中功能块的用法。

4.1 INI_File_WR

固件库 INI_File_WR 提供了读写 ini 文件的功能。

4.1.1 读取文件功能块

读取文件数据根据读取数据类型不同,提供三个功能块,分别是 **INI_File_Read_DINT**、**INI_File_Read_LREAL**、**INI_File_Read_STRING**, 功能块示意图如下图。



读取功能的使用方法以读取 **STRING** 类型为例介绍,其他功能块使用方法类似。

INI_File_Read_STRING

```
[MBTCPCtrl_PARAM]
listenPort = 502
registLen = 300
timeout = 1000
cmdexe timeout = 5000    (示例 ini 文件)
```

输入参数:

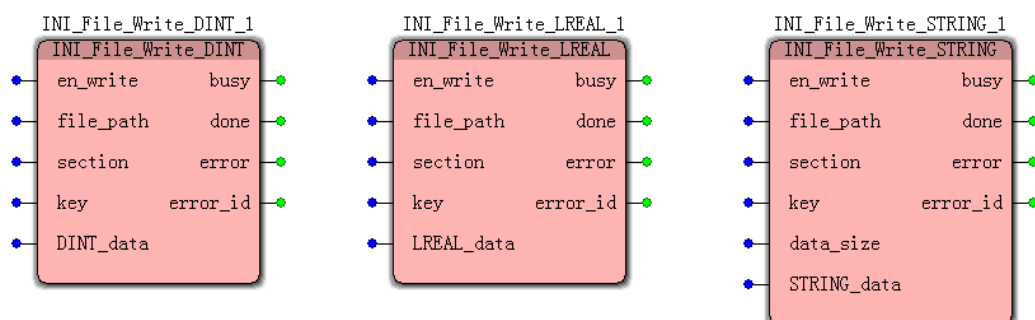
- en_read (STRING): 触发信号, 上升沿触发执行读取操作。
- file_path (STRING): “ini 文件路径\文件名”, 文件名需要加上后缀 “.ini”。
- lable_1 (STRING): 中括号内的字符串内容例如上图中的 “MBTCPCtrl_PARAM”。
- lable_2 (STRING): 需要写的项目索引, 例如上图中的 “listenPort”。
- data_size (INT): 需要读取的长度, 最大支持读取 80 字节长度的字符串。

输出参数:

- STRING_data (STRING): 读取到的字符串, 最大支持到 80 字节长度。
- busy (BOOL): 输出 TRUE 表示正在执行。
- done (BOOL): 输出 TRUE 表示执行完成或出错。
- error (BOOL): 输出 TRUE 表示执行出错失败。
- errir_id (INT): 执行失败时输出错误 ID。

4.1.2 写入文件

写入文件数据根据写入数据类型不同, 提供三个功能块, 分别是 **INI_File_Write_DINT**、**INI_File_Write_LREAL**、**INI_File_Write_STRING**, 功能块示意图如下图。使用方法类似 **INI_File_Read_STRING**。

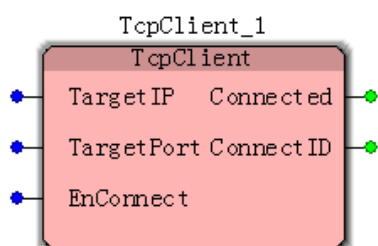


4.2 TCP_IP_Lib

本功能库提供 TCP/IP 通讯功能, 可作服务器或者客户端使用。

4.2.1 TcpClient

功能: 用于创建 TCP/IP 客户端, 连接至指定服务器, 功能块释义图如下图。



参数说明:

输入:

TargetIP (STRING): 目标服务器 IP, 例如 “192.168.60.200”。

TargetPort (DINT): 目标端口。

EnConnect (BOOL): 连接服务器使能, 上升沿执行。

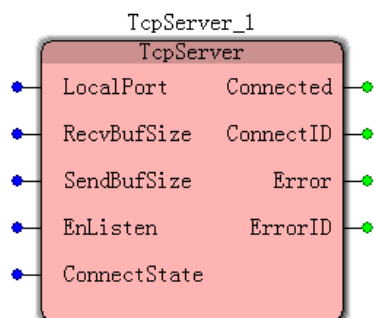
输出:

Connected (BOOL): 连接状态, TRUE 表示已连接服务器。

ConnectID (DINT): 连接 ID 号, 用于发送和接收功能块。

4.2.2 TcpServer

功能：用于创建 TCP/IP 服务器，功能块示意图如下图。



参数说明：

输入：

LocalPort (DINT)：本地端口号。

RecvBufSize (INT)：接收 Buffer 长度，最大支持到 256byte。

SendBufSize (INT)：发送 Buffer 长度，最大支持到 256byte。

EnListen (BOOL)：监听使能信号，上升沿开始执行。

ConnectState (DINT)：控制与客户端连接状态 0：正常连接 2：主动断开 4：错误状态主动断开。

注：本输入端口应与对应 ConnectID 的 TcpRecv 功能块的输出“RecvState”相连接。

输出：

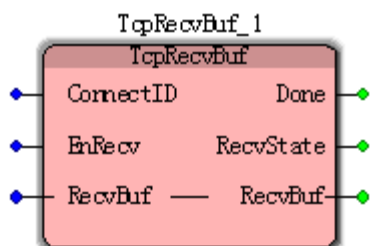
Connected (BOOL)：输出 TRUE 表示客户端已连接。

ConnectID (DINT)：连接 ID 号，用于发送和接收功能块。

Error (BOOL)：输出 TRUE 表示连接出错。

ErrorID (DINT)：错误 ID。

4.2.3 TcpRecvBuf



功能：用于接收（客户端/服务器）发送的数据，功能块示意图如下图。

参数说明：

输入：

ConnectID (DINT)：连接 ID 号，对应于服务器或客户端功能块的 ConnectID 输出。

EnRecv (BOOL)：接收使能，为 TRUE 期间持续接收数据。

输出：

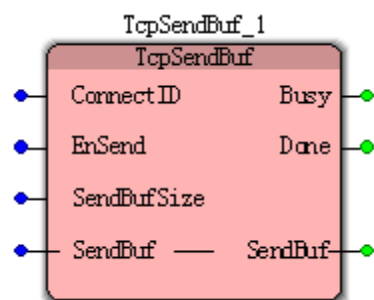
Done (BOOL): 功能块持续接收, Done 不输出信号

RecvState (DINT): 实时输出当前 ConnectID 对应连接的连接状态, 0: 正常 2: 断开 4: 错误

输入输出:

RecvBuf (ANY): 接收到的数据 Buffer。

4.2.4 TcpSendBuf



功能: 用于发送数据, 功能块示意图如下图。

参数说明:

输入:

ConnectID (DINT): 连接 ID 号, 对应于服务器或客户端功能块的输出 ConnectedID

ENSend (BOOL): 发送使能, 上升沿发送 SendBuffer 中的内容。

SendBufSize (DINT): 发送 Buffer 的字节长度。

输出:

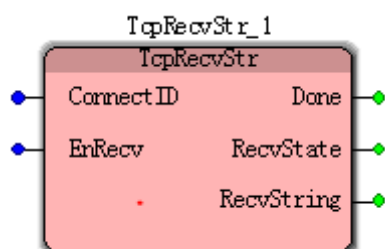
Busy (BOOL): 发送中

Done (BOOL): 发送完成后输出一个脉冲。

输入输出:

SendBuf (ANY): 待发送的 Buffer。

4.2.3 TcpRecvStr



功能: 用于接收 (客户端/服务器) 发送的字符串, 功能块示意图如下图。

参数说明:

输入:

ConnectID (DINT): 连接 ID 号, 对应于服务器或客户端功能块的 ConnectID 输出。

EnRecv (BOOL): 接收使能, 为 TRUE 期间持续接收数据。

输出:

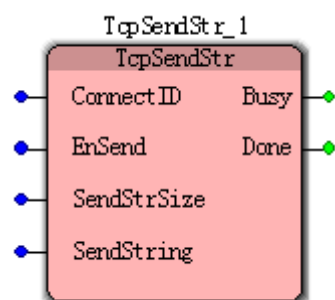
Done (BOOL): 功能块持续接收, Done 不输出信号

RecvState (DINT): 实时输出当前 ConnectID 对应连接的连接状态, 0: 正常 2: 断开 4: 错误

输入输出:

RecvStr (STRING): 接收到的字符串, 最大长度 80。

4.2.4 TcpSendStr



功能: 用于发送字符串, 功能块示意图如下图。

参数说明:

输入:

ConnectID (DINT): 连接 ID 号, 对应于服务器或客户端功能块的输出 ConnectedID

ENSend (BOOL): 发送使能, 上升沿发送 SendBuffer 中的内容。

SendBufSize (DINT): 发送 Buffer 的字节长度。

SendStr (STRING): 待发送的字符串, 最大长度 80。

输出:

Busy (BOOL): 发送中

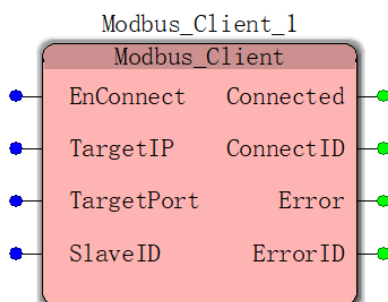
Done (BOOL): 发送完成后输出一个脉冲。

4.3 ModbusTCP_Lib

固件库 ModbusTCP_Lib 提供了 ModbusTCP 主站、读写寄存器功能。

4.3.1 Modbus_Client

功能: 用于创建 ModbusTCP 主站(客户端), 用于连接从站建立 Modbus 通信, 功能块释义图如下图。



参数说明:

输入:

EnConnect(BOOL): 为 TRUE 时, 一直尝试连接。

TargetIP(String): 目标服务器的 IP 地址, 例如 “192.168.60.200”。

TargetPort(DINT): 目标服务器的端口号。

SlaveID(UINT): 从站 ID 号, 用于 RTU 网络, 使用 TCP 通信时可缺省(默认为 1)

输出:

Connected(BOOL): 为 TRUE 时, 表示已经建立连接。

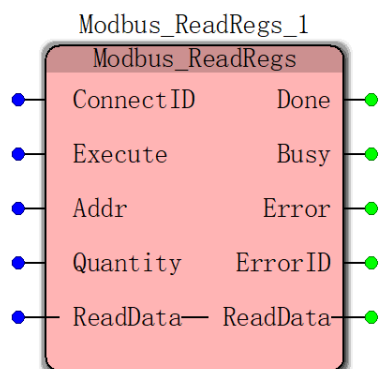
ConnectID(DINT): 建立连接后的通讯实例, 用于发送请求。

Error(BOOL): 功能块执行出错。

ErrorID(DINT): 执行错误 ID 号。

4.3.2 Modbus_ReadRegs

功能: 用于读取指定寄存器中的数据, 保存至本地变量。



参数说明:

输入:

ConnectID(DINT): 用于绑定 Modbus_Client 的输出 ConnectID, 表示指定客户端发送读写请求。

Execute(BOOL): 上升沿发送读取请求。

Addr(UINT): 待读取的寄存器的起始地址。

Quantity(UINT): 待读取的寄存器的数量。

输出:

Done(BOOL): 功能块执行成功。

Busy(BOOL): 功能块执行中。

Error(BOOL): 功能块执行出错。

ErrorID(DINT): 执行错误 ID 号。

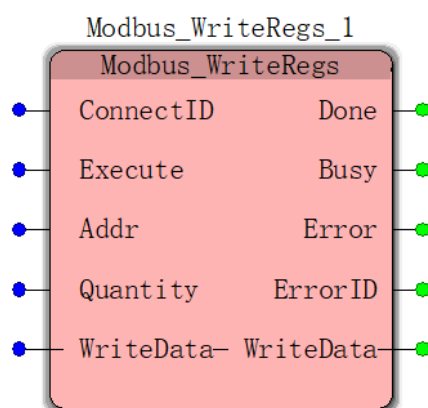
输入输出:

ReadData(MODBUS_READ_ARRAY): 读取到的数据保存到本地的变量, 最大为 125 个 WORD。

注: MODBUS_READ_ARRAY 结构体定义可在 ER_UserDataType 中查看。

4.3.3 Modbus_WriteRegs

功能: 用于将数据写入指定寄存器中。



参数说明:

输入:

ConnectID(DINT): 用于绑定 Modbus_Client 的输出 ConnectID, 表示指定客户端发送读写请求。

Execute(BOOL): 上升沿发送写入请求。

Addr(UINT): 待写入的寄存器的起始地址。

Quantity(UINT): 待写入的寄存器的数量。

输出:

Done(BOOL): 功能块执行成功。

Busy(BOOL): 功能块执行中。

Error(BOOL): 功能块执行出错。

ErrorID(DINT): 执行错误 ID 号。

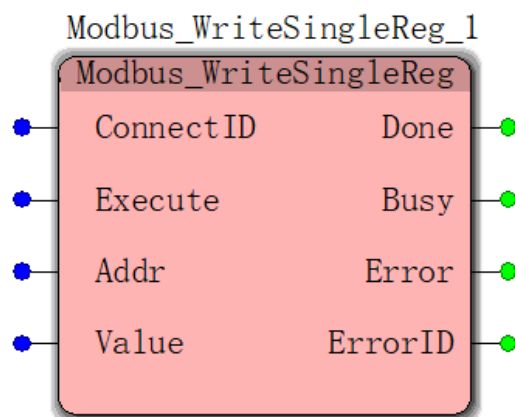
输入输出:

WriteData(MODBUS_WRITE_ARRAY): 待写入的数据, 本地的变量, 最大为 123 个 WORD。

注: MODBUS_WRITE_ARRAY 结构体定义可在 ER_UserDataType 中查看。

4.3.4 Modbus_WriteSingleReg

功能：用于将数据写入单个寄存器中。



参数说明：

输入：

ConnectID(DINT)：用于绑定 Modbus_Client 的输出 ConnectID，表示指定客户端发送读写请求。

Execute(BOOL)：上升沿发送写入请求。

Addr(UINT)：待写入的寄存器的起始地址。

Value(WORD)：待写入的寄存器的值。

输出：

Done(BOOL)：功能块执行成功。

Busy(BOOL)：功能块执行中。

Error(BOOL)：功能块执行出错。

ErrorID(DINT)：执行错误 ID 号。