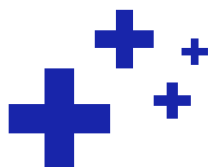


PLC 编程手册

安装使用本产品前请熟读本手册，并充分理解其内容。

请指定保管人员安全地保存在指定位置以便随时能阅读。



上海镓钠克数控科技有限公司

Shanghai Lynuc CNC Technology Co., Ltd.

概述

关于本说明书

- 手册名称 镓钠克数控装置 LYNUC-PLC 编程手册
- 文档类型 LYNUC PLC 的使用说明。
- 版本 V3.3

说明书使用

本说明书应在编程操作过程中使用。

本说明书的阅读对象

本说明书面向：

- 电气工程师
- 产品技术人员
- 技术服务人员
- 产品使用人员

操作前提

读者应：

- 熟悉编程手册中的相关概念
- 受过 PLC 编程操作方面的培训

参考信息

手册版本历史

版本	发布日期	修订说明
Ver 3.3	2020/2/20	SoftPLC 升级到 2.3.C.3 版本，修改以下内容： 1. 更新说明书里面所有的图片 2. 第四章 4.3.7 节“PLC 文件上传/下载”改名为“PLC 传输”，并增加 4.3.8 和 4.3.9 节。 3. 第十一章增加 11.4.14 节，马达正向运动禁止功能 4. 第十三章，删除编辑文件、系统远程升级、数据备份/恢复、NC 文件上传/下载等四个功能

目录

1 PLC 介绍	1
1.1 产品简介与规格.....	1
1.2 产品性能.....	1
2 SoftPLC 软件安装与运行	3
2.1 安装 SoftPLC.....	3
2.2 启动离线 SoftPLC 开发环境.....	3
3 PLC 编辑规范	6
3.1 设计编程内容及内容规范.....	6
3.2 变量使用.....	7
3.3 通讯资源.....	8
4 编程与使用	9
4.1 Soft PLC 界面构成	9
4.1.1 菜单栏.....	10
4.1.2 快捷工具栏.....	14
4.1.3 工程管理窗口.....	14
4.1.4 信息显示窗口.....	15
4.1.5 梯形图编辑区.....	16
4.2 工程的编辑和运行调试	18
4.2.1 创建新工程.....	18
4.2.2 编辑梯形图.....	21
4.2.3 保存工程.....	32
4.2.4 程序注释.....	33
4.2.5 生成程序文件.....	36
4.2.6 加载当前 PLC 程序.....	37
4.2.7 运行加载的 PLC 程序.....	39
4.2.8 查看 PLC 工程中的变量.....	42
4.2.9 数据采集.....	43
4.2.10 停止 PLC.....	48
4.3 其他功能介绍	50
4.3.1 查找功能.....	50
4.3.2 生成自动加载文件.....	54
4.3.3 交叉参考.....	56
4.3.4 清空无用 Lda 文件.....	57
4.3.5 清空磁盘.....	59
4.3.6 调试上电.....	60

4.3.7 PLC 传输.....	61
4.3.8 工程加密.....	65
4.3.9 工程解密.....	67
4.3.10 阅读模式.....	68
5 基本逻辑控制指令.....	70
5.1 语法参考.....	70
5.1.1 LD 语言.....	70
5.1.2 IL 语言.....	70
5.1.3 语句格式.....	71
5.2 位逻辑指令.....	71
5.2.1 常开触点/并联常开触点.....	72
5.2.2 常闭触点/并联常闭触点.....	73
5.2.3 普通线圈/复位线圈/置位线圈/反向线圈/报错线圈.....	74
5.3 逻辑运算指令.....	75
5.3.1 逻辑取反 NOT.....	75
5.3.2 逻辑与 AND.....	76
5.3.3 逻辑与非 ANDN.....	77
5.3.4 逻辑或 OR.....	78
5.3.5 逻辑异或 XOR.....	79
5.4 操作指令.....	80
5.4.1 上升沿.....	80
5.4.2 下降沿.....	81
5.4.3 跳变沿.....	82
5.4.4 自定义占空比方波发生器.....	83
5.5 计时器.....	84
5.5.1 指定编号的计时器.....	84
5.5.2 无指定编号的计时器.....	85
5.6 计数器.....	87
6 数值运算指令.....	89
6.1 整数运算.....	89
6.1.1 四则运算指令 ADD、SUB、MUL、DIV.....	89
6.1.2 取余取绝对值运算.....	90
6.1.3 自增一指令 INC、自减一指令 DEC.....	92
6.1.4 幂运算 POW.....	93
6.1.5 取自然对数取对数运算 LOGE、LOG2.....	94
6.1.6 循环自增一指令 INCC、循环自减一指令 DECC.....	96
6.2 三角函数运算.....	97
6.2.1 SIN、COS、TAN、CTAN 函数.....	97
6.2.2 ASIN、ACOS、ATAN 函数.....	99

7 传送与比较指令	101
7.1 赋值指令.....	101
7.1.1 赋值指令 MOVE	101
7.1.2 批量赋值 BLKMOVE.....	102
7.1.3 偏移地址赋值 MOVE_OFFEST	103
7.1.4 赋值偏移 MOVE_TO	104
7.2 比较指令.....	106
7.2.1 大于 GT/大于等于 GE.....	106
7.2.2 浮点数等于 DBEQ/浮点数不等于 DBNE.....	107
8 移位指令	109
9 数据处理指令	110
9.1 类型转换.....	110
9.1.1 转无符号整数 TO_U	110
9.1.2 转有符号整数 TO_S.....	111
9.1.3 转浮点数 TO_F.....	112
9.1.4 转双精度浮点数 TO_D.....	113
9.2 数据交换 XCH.....	114
9.3 ASCII 字符编码.....	114
10 程序流程控制指令.....	117
10.1 PLC 程序结构和程序流程.....	117
10.2 程序流程控制指令.....	117
11 CNC 扩展功能指令	119
11.1 控制指令.....	119
11.1.1 MCODE 指令.....	119
11.1.2 MFINISH 指令.....	120
11.1.3 MCALL 指令	121
11.1.4 T 指令获取 TGET.....	122
11.2 辅助功能指令	123
11.2.1 由刀具号获取刀套号 TOOLPOT	123
11.2.2 刀具类型 TOOLTYPE	124
11.2.3 由刀套获取刀具号 TOOLNUM.....	125
11.2.4 刀套类型 TOOLEXIS.....	126
11.2.5 旋转控制 ROT	127
11.2.6 调度内核 COMMAND_LINE.....	128
11.2.7 直接调度内核 S_CMD	129
11.3 主轴功能指令	130
11.3.1 主轴开环使能 SPOPEN.....	130
11.3.2 主轴输出 SPOUT.....	131

11.4 马达功能指令	132
11.4.1 马达回零 Home.....	132
11.4.2 马达回零完成标记 Home COMP.....	133
11.4.3 马达加速时间 JOGTA	134
11.4.4 马达移动速度 JOGV	135
11.4.5 马达移动位置 JOGA	136
11.4.6 马达移动距离 JOGI.....	137
11.4.7 Index 位置检测 IDXPOS.....	138
11.4.8 马达指令坐标 MTCMDP	139
11.4.9 杀掉马达 MTKILL	140
11.4.10 马达闭环 MTCLP	140
11.4.11 马达使能 MTENA	141
11.4.12 马达到位检测 MTINPOS.....	142
11.4.13 马达软限位设置 MTLMT	143
11.4.14 马达正向运动禁止	144
12 Examples	145
12.1 读写系统 IO	145
12.2 产生方波.....	147
12.3 调用子程序.....	148
13 配置管理功能	149
13.1 添加文件.....	149
13.2 删除文件.....	150
13.3 导入数据包.....	151
13.4 生成数据包.....	153
14 特殊功能和错误处理	156
15 PLC 接口详细	157
15.1 PLC 通知系统的外部输入指令	157
15.2 PLC 得到系统的内部输出指令	162
15.3 面板 PLC.....	166
15.4 主轴-PLC11.....	167
15.5 用户自定义-PLC13	170
15.6 PLC 执行 M 指令	172
15.7 其他 PLC.....	174

1 PLC 介绍

概述

PLC 是 Programmable Logic Controller 的简称，即可编程逻辑控制器。

可编程逻辑控制器是一种采用一类可编程的存储器，用于其内部存储程序，执行逻辑运算、顺序控制、定时、计数与算术操作等面向用户的指令，并通过数字或模拟式输入/输出控制各种类型的机械或生产过程。

1.1 产品简介与规格

概述

LYNUC PLC 为用户提供了一个 Ladder 图形化的编辑环境，用户在该环境下可以新建、编辑、保存 Ladder 文件；可以启动、加载、停止 PLC 程序；可以实时监控 PLC 的运行状态。

规格及参数：

表 1-1-1 产品规格

数量	性能
256 个 PLC	最小循环周期 1ms
最大 10000 个 rung	最大循环周期 1 秒
13*32 个输入	PLC 速度基本上取决于对数据的访问速度和次数。对共享内存访问速度最快时间为 60nm（大部分类型）~200nm（位移操作），不同控制器型号的访问速度有差异。 在 Lynuc 系统里，主要的操作是对共享内存的访问。
13*32 个输出	
1000 个扩展输入（32 位）	
1000 个扩展输出（32 位）	
5120 个 Q 变量（32 位）	
R 变量（参考宏编程手册）	

1.2 产品性能

概述

PLC 的控制系统性能指标主要有以下几种，现分别介绍：

①. I/O 点数：

PLC 可以接受的输入信号和输出信号的总和。表示 PLC 组成系统时可能最大的规模。

一般情况下，输入点数大于输出点数，而且输入点数与输出点数不能互相替代。

②. 存储容量：

用户程序存储器的容量。通常用 K 字（KW）、K 字节（KB）为单位，1K=1024B。

③. 扫描速度：

PLC 执行用户程序的速度。一般以执行 1000 条基本指令所需要的时间来衡量。单位为 ms/千步。一般逻辑指令和运算指令的平均执行时间有较大的差别，因而多数场合，扫描速度需要指定是执行哪类程序。

④. 编程语言：

编程语言是指用户与 PLC 进行信息交换的方法。目前已公布的 PLC 编程语言标准 IEC61131-3。详细说明了 PLC 可使用的 5 种编程语言:指令表（IL）、梯形图（LD）、顺序功能图（SFC）、功能图（FBD）和结构文本（ST）。目前指令表、梯形图、顺序功能图是使用最多的编程语言。特别是梯形图，几乎所有的 PLC 都支持该编程方法。但并不是所有厂家的 PLC 编程语言都可以兼容，即使梯形图语言相同，但指令表语言也不通用。

⑤. 指令的功能与数量：

编程指令的功能越强、数量越多，PLC 的处理能力和控制能力也越强，用户编程也越简单方便，越容易完成复杂的控制任务。

⑥. 内部元件的种类和数量：

元件的种类和数量越多，表示 PLC 的存储和处理各种信息的能力越强。

⑦. 可扩展能力：

PLC 可扩展能力包括 I/O 点数的扩展、存储容量的扩展、联网功能的扩展、各种功能模块的扩展等。

⑧. 特殊功能单元：

特殊功能单元种类越多，功能越来越强，使得 PLC 控制功能日益扩大。

2 SoftPLC 软件安装与运行

概述

PLC 可以脱离控制器，离线进行编辑。编辑完成后，导入到控制器即可运行（导入参见后面章节—导入控制器）。

2.1 安装 SoftPLC

离线 SoftPLC 编辑工具需要自行安装在 windows 系统中。在安装 SoftPLC 之前，必须确保安装环境可以正常使用。如果已经安装有老版本的 SoftPLC 和外设工具，则需要先卸载。

安装程序，请从上海铼钠克数控科技股份有限公司官网上下载。

2.2 启动离线 SoftPLC 开发环境

操作步骤

- ①. 双击程序软件，启动 SoftPLC。
- ②. 当启动后出现 PLC 的初始屏，如下所示：

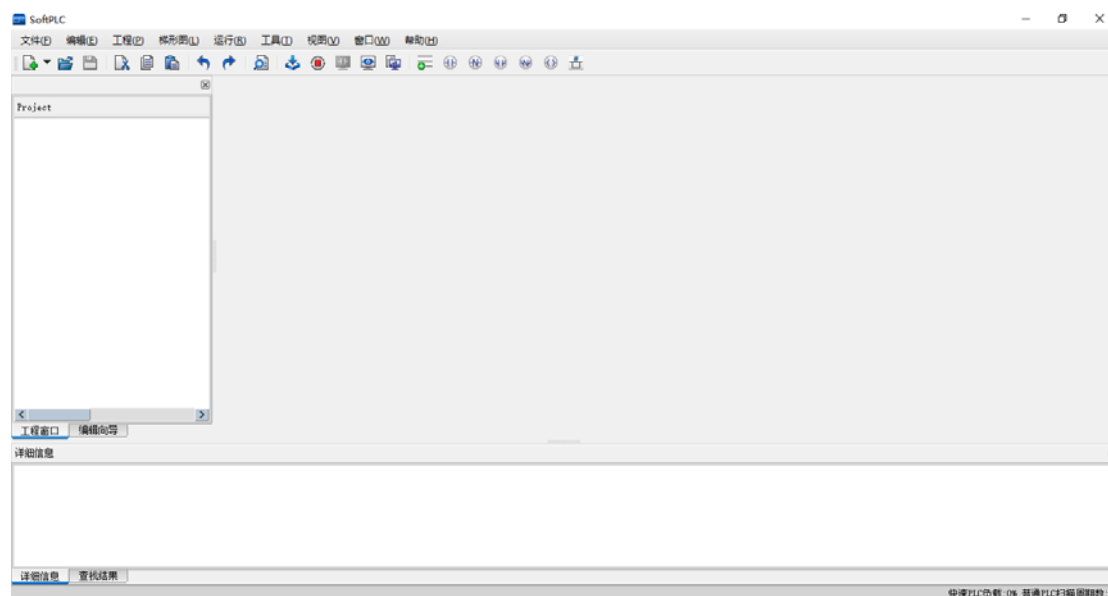


图 2-2-1 SoftPLC 启动界面

- ③. 在 SoftPLC 中点击工具栏的[文件]-[打开]按钮。

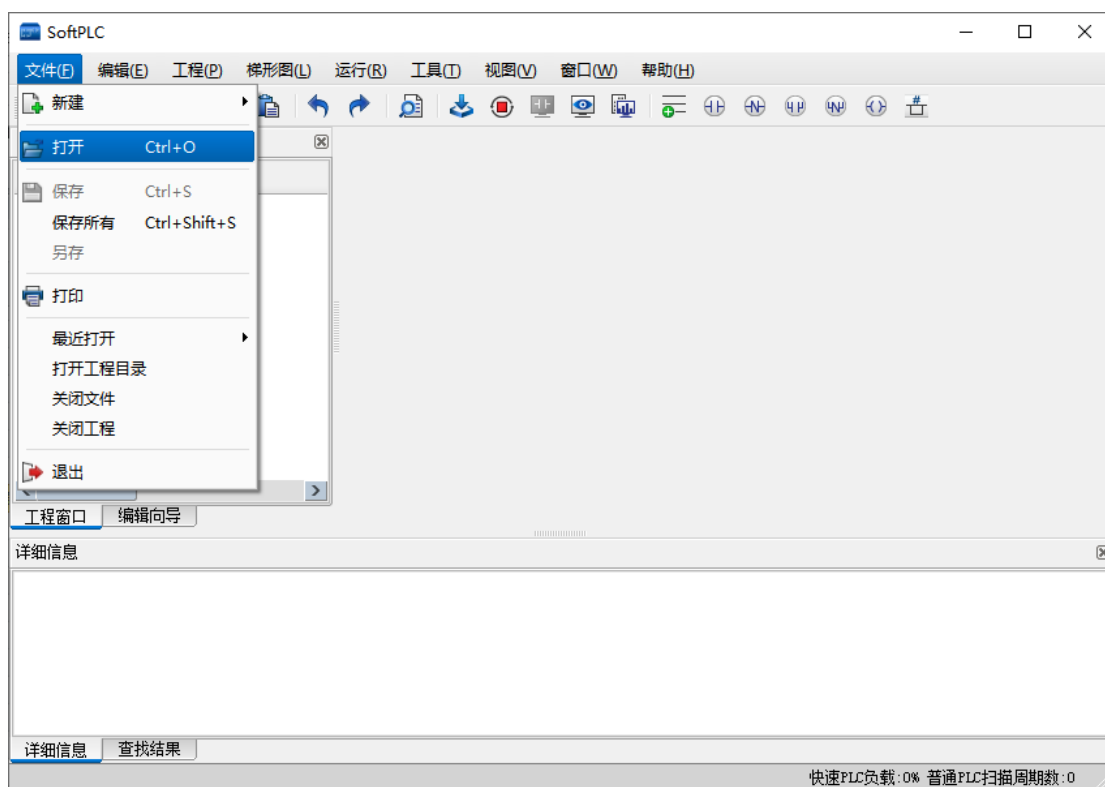


图 2-2-2 打开文件-1

④. 打开需要运行的 PLC 文件。

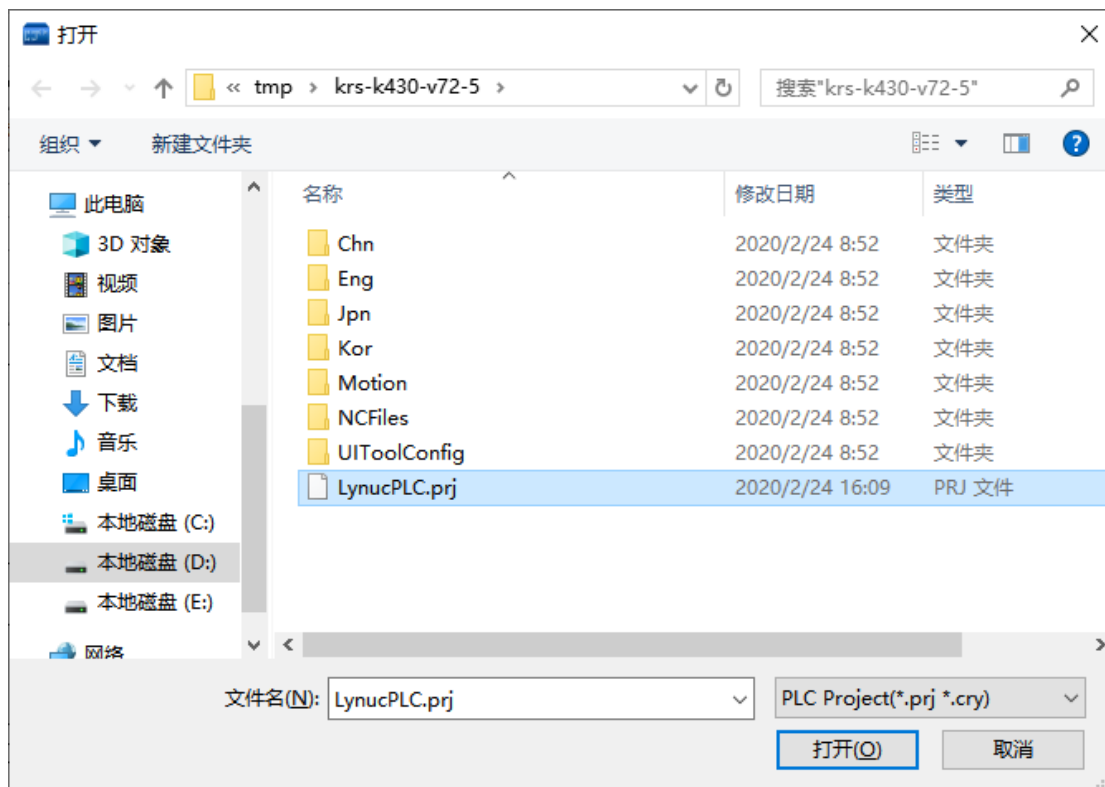


图 2-2-3 打开文件-2

⑤. 点击“打开”，启动了 PLC 离线运行的开发环境。

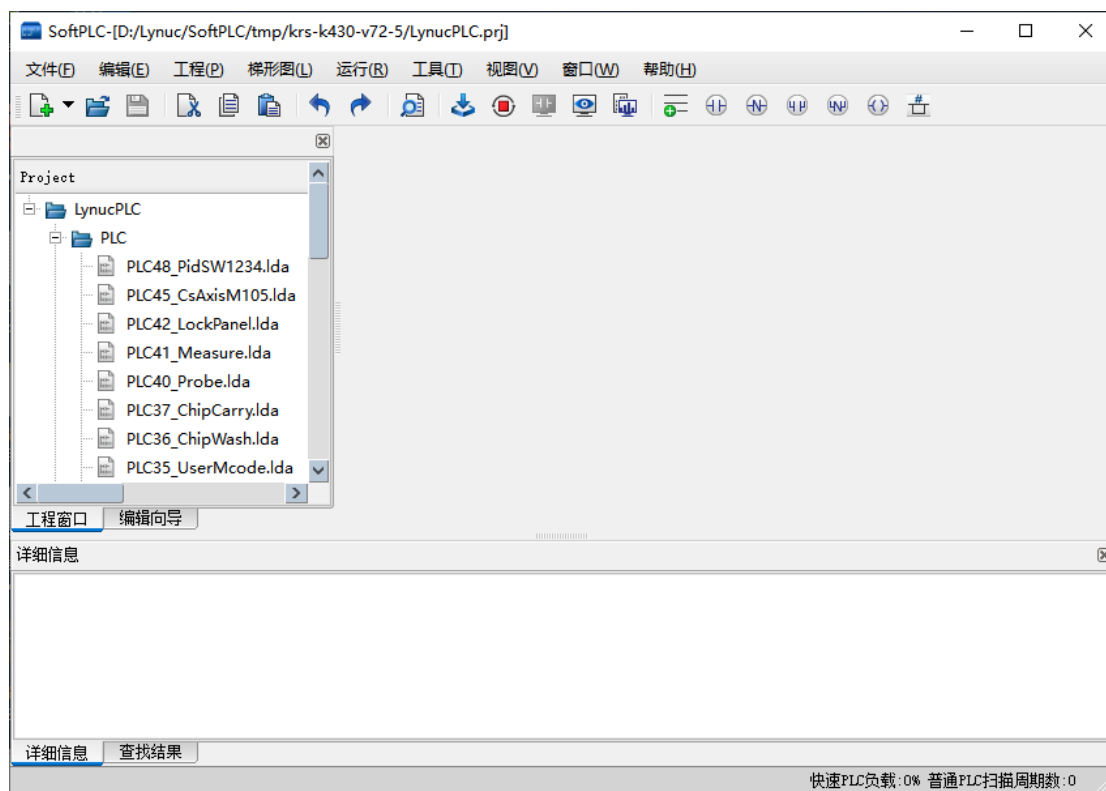


图 2-2-4 PLC 离线运行界面

说明

- 1) PPLC-是指反复执行的 PLC 程序，在一个扫描周期内结束，在下一个扫描周期重新执行。
- 2) SPLC-是指执行一次 PLC 程序，可以执行等待(一个信号)和 sleep (固定时间) 这两个特殊指令，所以不必在一个循环周期内结束。因为它只执行了一次，所以没有边沿触发、计时器和计数器指令。

3 PLC 编辑规范

概述

为规范项目自动化程序，将项目自动化程序做得更严谨高效，PLC 的编程方法需要更规范、统一，以方便程序的运行维护。

规范对象

本规范适用于技术中心、PLC 设计工作及其技术改进的控制，以及相关技术应用人员。

3.1 设计编程工程及内容规范

分析系统

分析被控对象的特点，提出被控对象对 PLC 控制系统的控制要求，确定控制方案，拟定详细的设计任务。

制定控制方案

用选择开关决定控制系统的全自动、半自动运行和手动调整方式；可根据需要调整控制方案。

系统配置及输入输出对照表

根据系统的控制要求，确定系统所需的全部输入设备（如：按钮、位置开关、转换开关及各种传感器等），以及输出设备（如：接触器、电磁阀、信号指示灯及其执行器等），从而确定与 PLC 有关的输入/输出设备，以确定 PLC 的 I/O 点数。

设计主电路及 PLC 外部接线图

列出需购部件单；分配 I/O 点数并画出 I/O 点与输入输出设备的连接图或对应关系表，设计外围硬件线路；画出系统其它部分的电气线路图，包括主电路和未进入系统的控制电路等。

程序编制

程序设计应严格按照功能流程图进行设计。在程序编制过程中，应及时在程序段中编写相应的注释语句。

程序调试

调试分为：模拟调试和现场调试。模拟调试：根据现场信号，采用硬件模拟法进行。输入信号可利用按钮开关和钮子开关来模拟，输出信号可直接用 IO 继电器板的输出信号灯进行观察。

现场调试则根据现场实际情况进行软件的调整。第一步：只连接输入设备进行调试；第二步：连接输入、输出设备调试；第三步：现场全部接上逐步进行调试。

3.2 变量使用

使用建议

- ①. 系统保留变量，禁止任何中间变量或机床用户使用的段：

MO 0~99, 1000~1023

MI 0~99, 1000~1023

- ②. PLC 中间临时变量：

每个 PLC 有 20 个 Q 变量，编号 0~19。

所有局部的中间变量，请尽量使用 Q 变量，避免 PLC 之间变量的干涉使用。

其中每个 Q0.0 用于空的输出节点。

- ③. 计时器、计数器：

请使用编号 TON、TOF 时，特别注意，避免编号有重复的情况，否则会引起逻辑错误，互相干扰。计时器提供有 TONN、TOFN 无需编号的模块，使用这两个模块时不会产生重复问题，推荐使用。

计数器 CTU、CTD 必须指定编号，请使用时特别注意，避免编号重复的情况。

- ④. R 变量：

R 变量是数控系统提供的宏变量，用于对数控系统内核数据的访问。

R300~R1999 范围的 R 变量属于掉电保存型，频繁的改写会引起系统频繁进行保存写操作的动作，会消耗 CPU 资源，甚至损坏数控系统的存储设备。请谨慎使用 R 变量！涉及频繁变化的变量，尽量用 Q 变量或 MO 变量。

3.3 通讯资源

概述

系统与 PLC 的通讯要求如下资源：

表 3-3-1 通讯资源

变量	定义
X0~X12	输入信号
Y0~Y12	输出信号
G 变量	从 PLC 侧到系统的信号
F 变量	从系统到 PLC 侧的信号
MO20	严重错误，系统错误编号 1001~1032
MO21~MO22	错误，系统错误编号 1033~1096
MO23	警告，系统错误编号 1097~1128
MO24	消息，系统错误编号 1129~1160

※ 注意

1. 本系统已经占用了 MO 变量以及 MI 变量的前 100 个（即 MO0~MO99、MI0~MI99），以及编号 1000 以后的变量（即 MO1000~MO1023、MI1000~MI1023），用户在编写 PLC 程序时，请避免使用这些变量。
2. 本系统占用了普通 PLC 的 1~10，用户可以使用 PLC0， PLC11 及以上的 PLC 程序。

4 编程与使用

概述

把用户程序写入 PLC 中必须使用专门的编程工具。

本节主要介绍与 PC 配合的编程及仿真软件的使用。

本节主要针对学习了 Lynuc PLC 基础指令后需要使用编程软件的用户。

4.1 Soft PLC 界面构成

软件启动后显示的主窗口界面如下图所示：

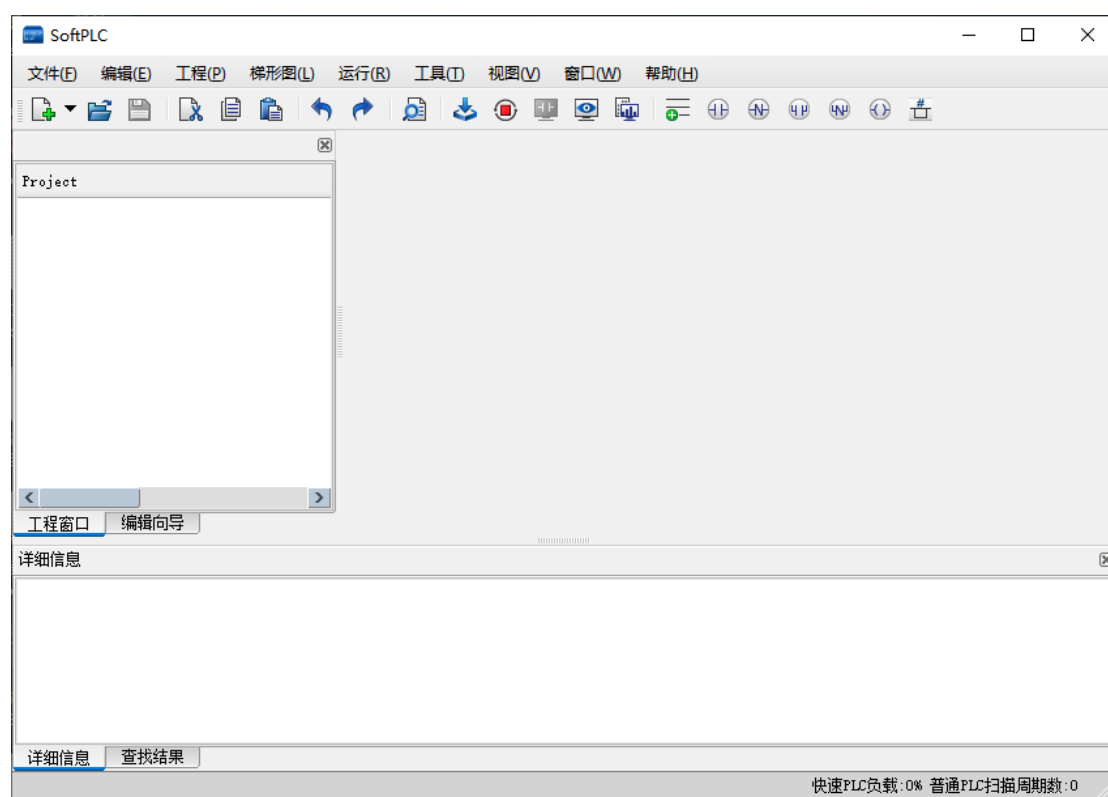


图 4-1-1 PLC 启动主界面

说明

主要分为以下几个区域：

表 4-1-1 SoftPLC 界面功能说明

名称	功能/内容
----	-------

菜单栏	共 9 个下拉菜单。
常用工具栏	点击常用图标选择功能。
工程管理窗口	包括工程窗口和编辑向导
信息显示窗口	显示详细信息和查找结果
梯形图编辑区	包括变量定义区和梯形图描画区。

4.1.1 菜单栏

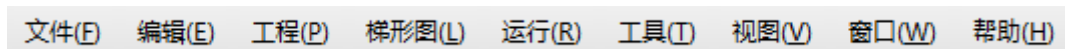


图 4-1-1-1 SoftPLC 菜单栏

表 4-1-1-1 SoftPLC 菜单栏选项

选项	子选项
文件	新建、打开、保存、保存所有、另存、打印、最近打开、关闭文件、关闭工程、退出
编辑	剪切、复制、粘贴、查找、查找上一个、查找下一个、比较、撤销、恢复、生成自动加载文档、交叉参考、类型检查、检查跳转、清空无用 Lda、清空磁盘
工程	添加文件、删除文件、导入数据包、生成数据包
梯形图	前面添加触点、添加节点、并联节点、添加输出、添加功能块、添加一行、删除节点、删除行、取消删除
运行	加载 PLC、运行、停止
工具	查看、数据采集、调试上电、系统远程升级、数据备份/恢复、NC 文件上传、NC 文件下载、PLC 文件上传/下载
视图	阅读模式、总览、详细信息、显示变量地址、显示别名、显示说明
窗口	全部关闭
帮助	帮助文档、关于

1. 文件

在文件菜单下包含了对文件进行操作的命令选项,主要是为用户提供了编辑梯形图文件的各种操作。

表 4-1-1-2 文件菜单栏选项

新建	该选项用于创建新的工程以及 PLC 文件。
打开	该选项用于打开已有的工程。
保存	该选项用于将当前窗口的文件内容保存为 LDA 文件。
保存所有	该选项用于保存所有窗口下的文件内容为 LDA 文件。
另存	该选项的功能与[保存]选项类似,也是保存最近编辑的文件,不同的是[另存]是指将打开的文件用新的文件名保存。
打印	该选项用于打印当前窗口中的内容。
最近打开	该选项用于显示最近打开的文件。
关闭文件	该选项用于关闭当前的文件。
关闭工程	该选项用于关闭当前的 PLC 工程
退出	该选项用于退出程序, 关闭软件。

2. 编辑

编辑菜单下包含了复制粘贴等快速操作方式,编辑中的各项功能是为了提高用户编写梯形图的效率,主要用于用户编写 PLC 的情况。这里主要介绍如下选项功能:

表 4-1-1-3 编辑菜单栏选项

复制/粘贴/剪切	快捷键分别为 Ctrl+C /Ctrl+V /Ctrl+X	-
查找	快捷键为 Ctrl+F, 该选项用于在程序的整个项目或当前的文件中查找某个变量、指令、注释、或地址的交叉引用等内容。	详见 4.3.1 节
查找上一个	快捷键为 F4, 与查找配合使用	
查找下一个	快捷键为 F3, 与查找配合使用	
比较	快捷键为 Ctrl+H, 现阶段不成熟	-
撤销/恢复	快捷键分别 Ctrl+Z /Ctrl+Y,该选项表示对编辑的内容进行修改	-
生成自动加载文件	该文件决定系统启动时自动加载的 PLC 文件	详见 4.3.2 节

交叉参考	查看某些地址在整个工程中交叉使用的情况	详见 4.3.3 节
类型检查	查看该 PLC 文件中组件的类型是否匹配	
检查跳转	PLC 文件加载后检查跳转是否正确	
清空无用 Lda	将工程树中不存在自动加载文件中的 PLC 清空并更新工程文件	详见 4.3.4 节
清空磁盘	将无用的 PLC 文件再显示在工程树中	详见 4.3.5 节

3. 工程

表 4-1-1-4 工程菜单栏选项

添加文件	工程中添加文件	详见 13 章-配置管理功能介绍
删除文件	删除工程文件	
导入数据包	由数据包解压生成工程	
生成数据包	由工程生成数据包	

4. 梯形图

表 4-1-1-5 梯形图菜单栏选项

前面添加触点	添加触点，选中后再点添加节点	
添加节点	包括添加常开触点、常闭触点	详见 4.2.2 节 编辑梯形图
并联节点	包括并联常开和并联常闭触点	
添加输出	包括各种输出线圈	
添加功能块	包含数十种功能指令	详见 5-11 节指令
增加一行	快捷键为 Alt+Ins	详见 4.2.2 节 编辑梯形图
删除节点	快捷键为 Del	
删除行	快捷键为 Shift+Del	
取消删除	快捷键为 Esc	

5. 运行

表 4-1-1-6 运行菜单栏选项

加载 PLC	快捷键为 Ctrl+D,	详见 4.2.6-4.2.7 节
运行	快捷键为 Ctrl+R	

停止	快捷键为 Ctrl+T	详见 4.2.10 节
----	-------------	-------------

6. 工具

表 4-1-1-7 工具菜单栏选项

查看	快捷键为 Ctrl+M, 查看某个变量值	详见 4.2.8-4.2.9 节
数据采集	采集数据	
调试上电	同 N3Config 中强制上强电的功能	详见 4.3.6 节
PLC 传输	上传或下载 PLC 文件	详见 4.3.7 节
工程加密		
工程解密		

7. 视图

表 4-1-1-8 视图菜单栏选项

阅读模式	快捷键为 F9, 在线时梯形图可显示全	详见 4.3.8 节
总览	设置工程管理窗口是否显示	
详细信息	设置信息显示窗口是否显示	
显示变量地址	快捷键为 Ctrl+F1	
显示别名	快捷键为 Ctrl+F2	
显示说明	快捷键为 Ctrl+F3, 显示组件的注释	

8. 窗口

表 4-1-1-9 窗口菜单栏选项

全部关闭	可将打开的 PLC 文件全部关闭	
------	------------------	--

9. 帮助

表 4-1-1-10 帮助菜单栏选项

帮助文档	快捷键为 F1	
关于	可查看软件版本信息, 输入密码后可查看系统 PLC 工程	

4.1.2 快捷工具栏



图 4-1-2-1 快捷工具栏

快捷工具栏子选项如表 4-1-2-1 所示：

表 4-1-2-1 快捷工具栏

选项	子选项	选项	子选项
	新建工程、PPLC 文件、SPLC 文件		监视
	打开文件		查看
	保存		数据采集
	剪切		增加一行
	复制		常开触点
	粘贴		常闭触点
	撤销		并联常开触点
	后退		并联常闭触点
	查找		普通线圈
	加载 PLC		赋值
	停止		

4.1.3 工程管理窗口

①. 工程窗口

工程窗口包含了各工程文件，可以查看不同的 PLC 文件。

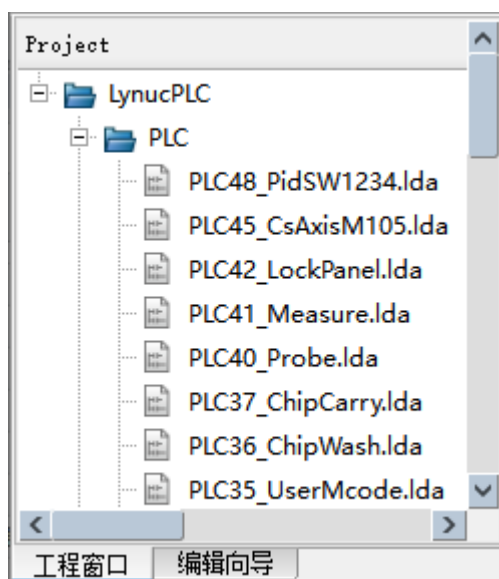


图 4-1-3-1 工程窗口

②. 编辑向导

包含了编制 PLC 的各种指令，如下图所示，具体各指令介绍请查阅相关章节。

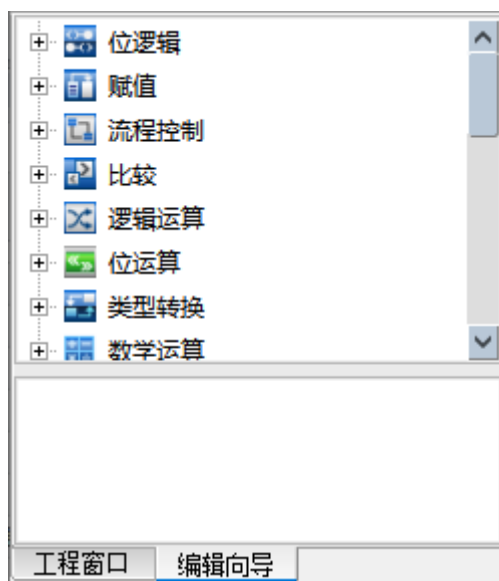


图 4-1-3-2 编辑向导

4.1.4 信息显示窗口

概述

包括详细信息和查找结果。在编辑梯形图的时候，若需了解编辑操作的提示信息或进行查找操作，则可在此处查看详细信息或查找结果。



图 4-1-4-1 信息显示窗口

4.1.5 梯形图编辑区

概述

包括变量定义区和梯形图描画区：

- 变量定义区窗口：相当于寄存器的编辑窗口，包括各变量的名称、别名、初值、是否常闭、注释各选项。

变量名称	别名	初值	常闭	注释
1 M020.0	E_PwAlm		<input type="checkbox"/>	(报警) 机床上电超时错误
2 Q10.0	Q_Pwsts		<input type="checkbox"/>	(用户接口) 强电输入信号 [根据QV_xMode参数选择强电状态来自于那个信号]
3 Q19.0	X_Pwsts		<input type="checkbox"/>	(IO信号) 强电输入信号 [如果没有IO输入, 则用Q19.02代替] [Q19.00] [根据QV_xMode参数选择]
4 Q19.1	Y_PwOff		<input type="checkbox"/>	(IO信号) CNC自动断电 [控制系统关机断电, PRIO有专用点] [Q19.01]
5 Y12.02	Y_zBreak		<input type="checkbox"/>	(IO信号) Z轴刹车输出 [Z刹车由PLC控制时才会用到] [Q19.02]
6 Q19.20	Y_Power		<input type="checkbox"/>	(IO信号) 上强电输出 [上强电的输出0点位, PRIO有专用点] [Q19.20]

图 4-1-5-1 变量定义窗口

- 梯形图描画区：在梯形图界面中需要用到菜单中的梯形图、快捷工具栏和编辑导向。

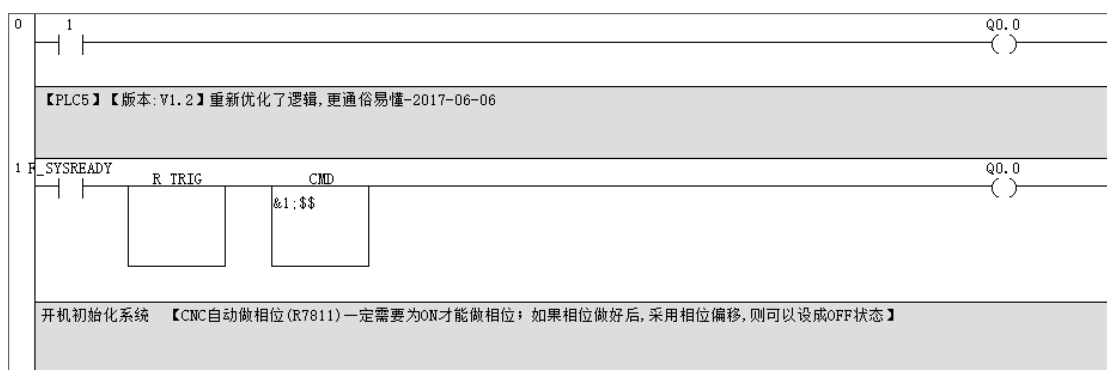


图 4-1-5-2 梯形图描画窗口

- ①. 菜单中的梯形图用于加入基本输入输出单元和选择各项功能模块。

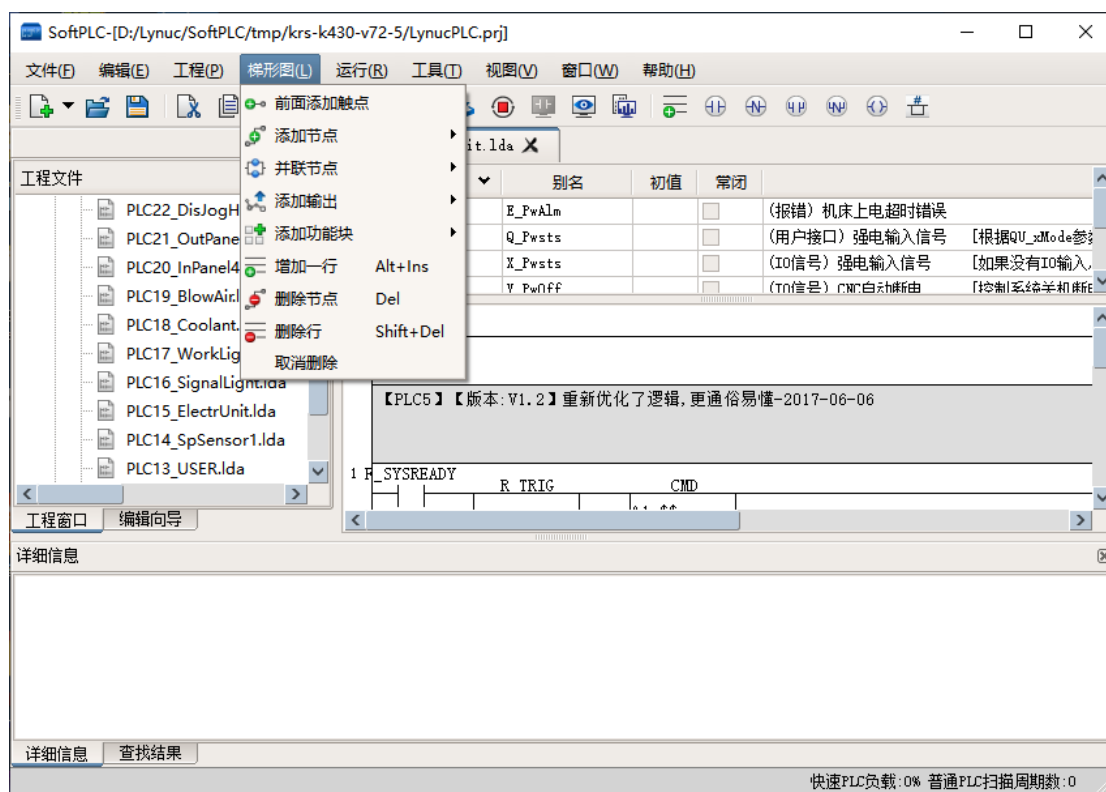


图 4-1-5-3 梯形图菜单栏

- ②. 快捷工具栏除了用于快捷的操作，如新建文件，复制粘贴、撤销恢复等菜单中的操作，还可以加入基本输入输出单元和选择各项功能模块。



图 4-1-5-4 快捷工具栏

- ③. 编辑向导菜单用于加入基本输入输出单元和选择各项功能模块。



图 4-1-5-5 编辑向导菜单栏

4.2 工程的编辑和运行调试

4.2.1 创建新工程

概述

进入编程画面后，画面仅显示菜单栏和工具栏以及工程管理窗口。

通过利用“新建工程”或“打开工程”，才能进入编程画面。

创建步骤

- ①. 点击工具栏[新建]-[工程]快捷图标，或选择菜单栏[文件]-[新建]-[工程]按钮。

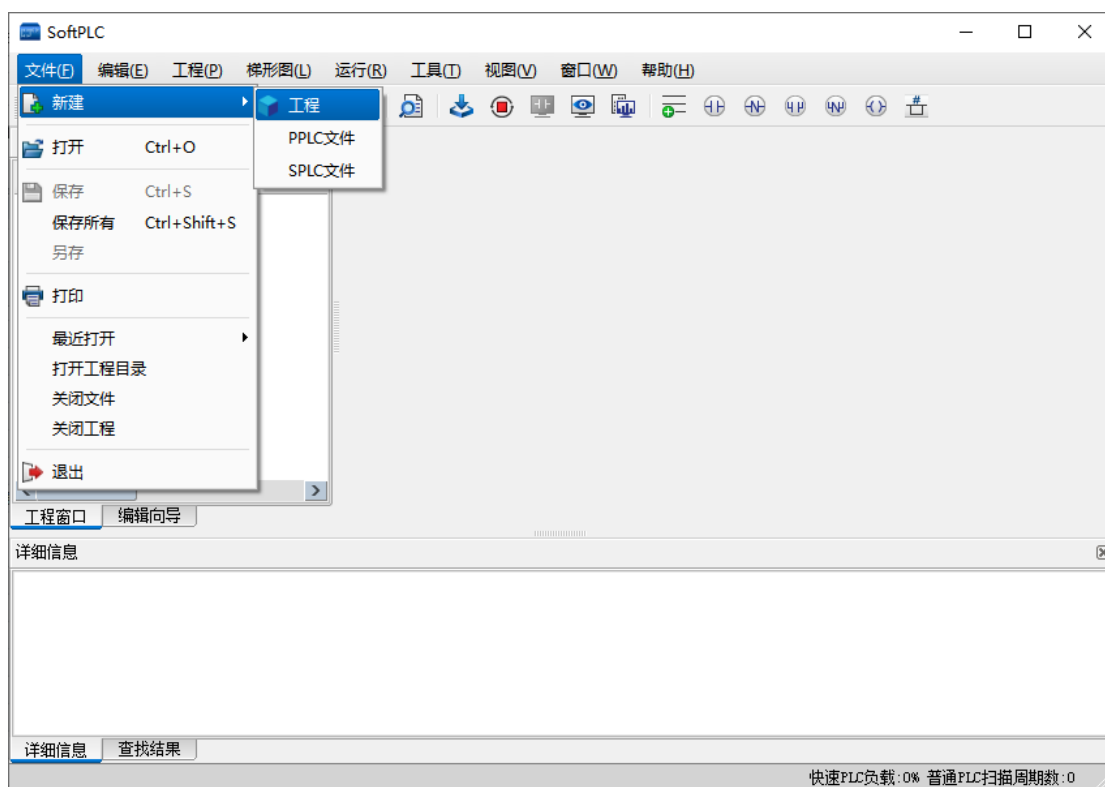


图 4-2-1-1 新建工程

②. 弹出窗口新建工程[文件名], 命名为“LynucPLC”并点击保存。

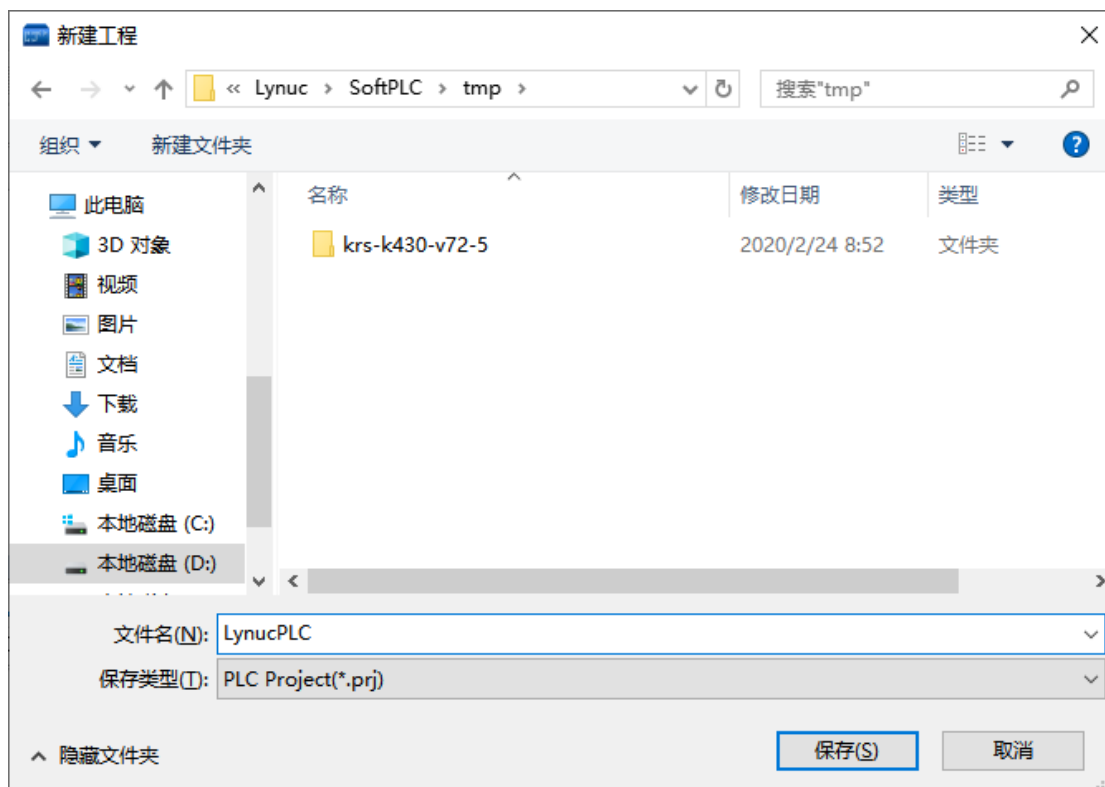


图 4-2-1-2 命名新建工程

③. 点击[保存]选项后，弹出新建工程的窗口界面，如下所示：

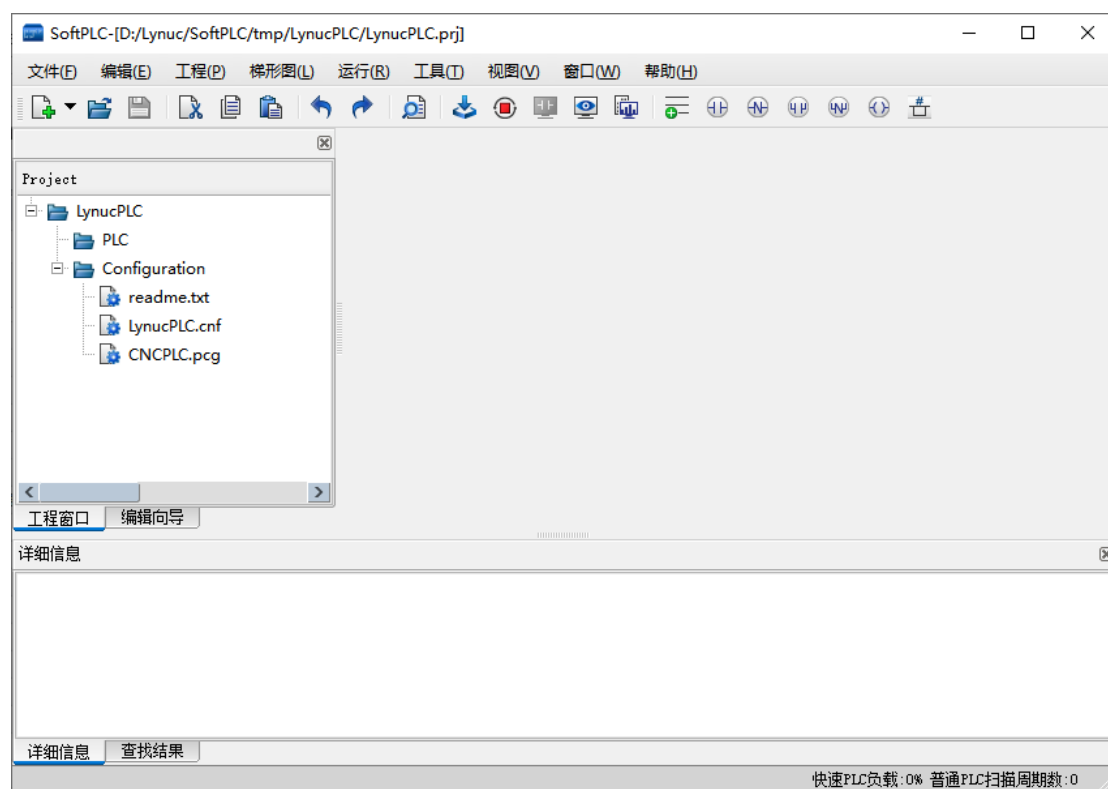


图 4-2-1-3 新建工程在界面中的显示

④. 将鼠标移动到[LynucPLC]工程下的[PLC]目录选项，右键单击出现窗口选项[新建]、[添加]、[删除]、[创建 PCG]。本节点点击[新建]-[PPLC 文件]选项创建文件。

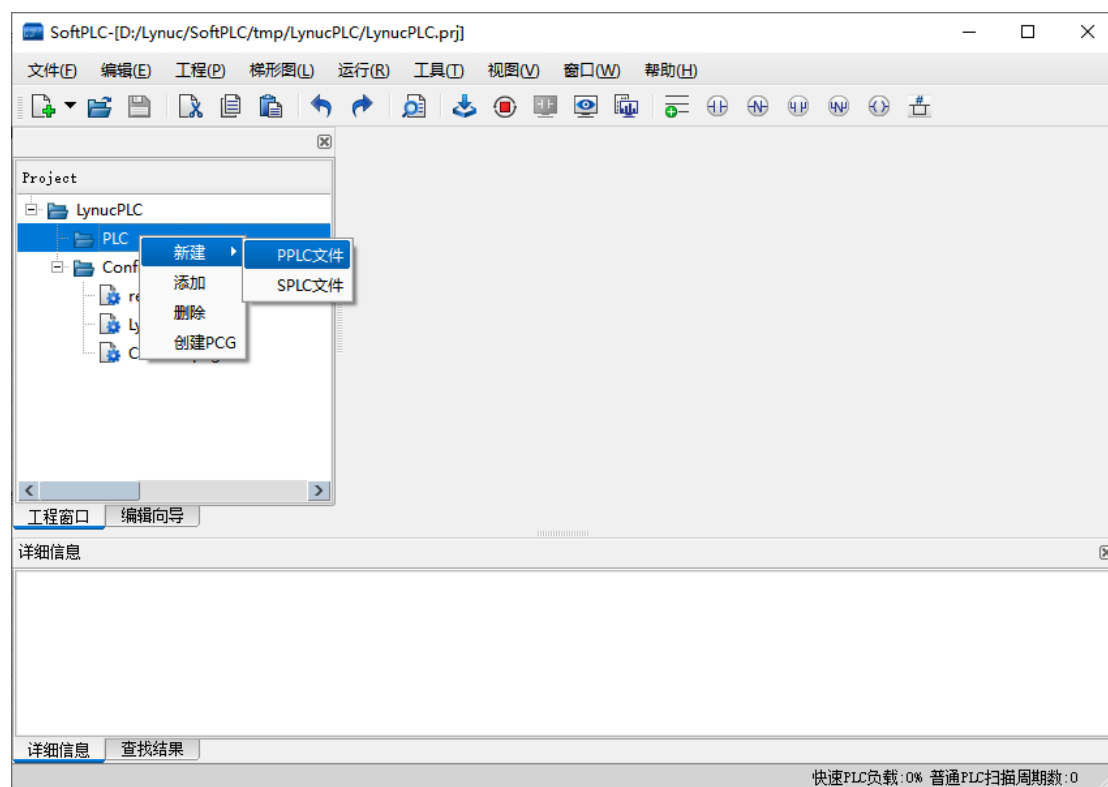


图 4-2-1-4 新建 PPLC 文件-1

⑤. 点击创建[PPLC 文件]后出现如下画面，即梯形图编辑区。

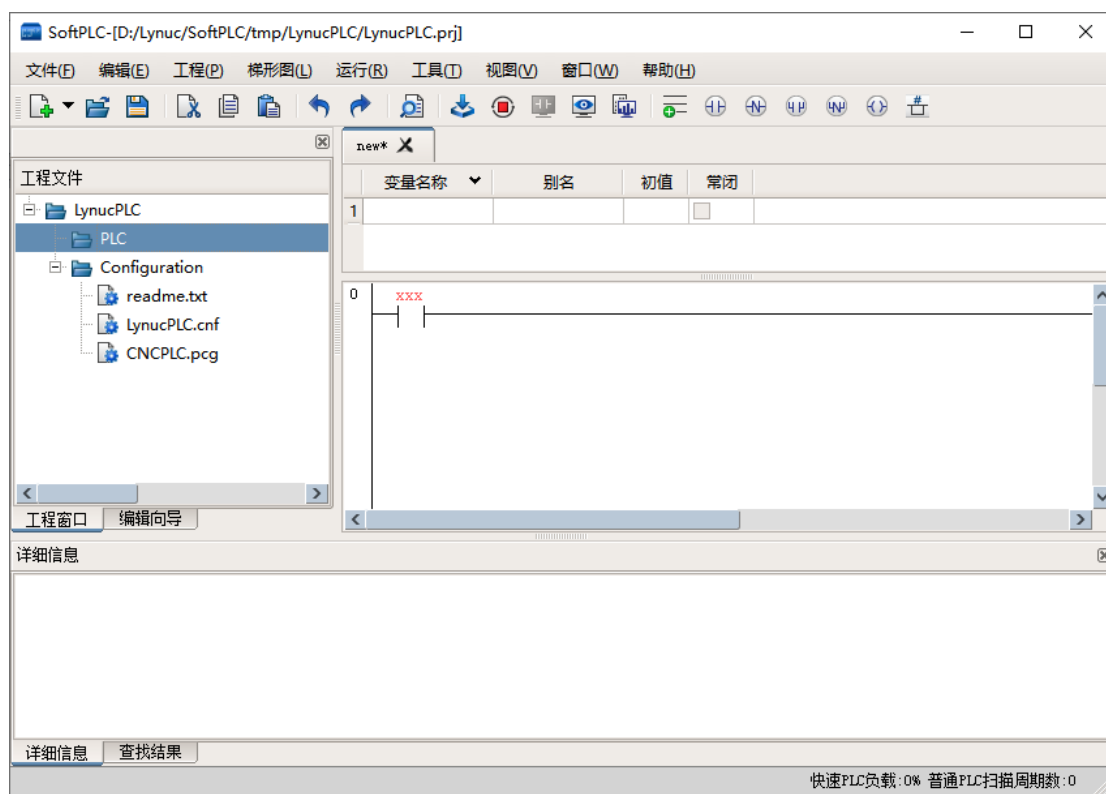


图 4-2-1-5 新建 PPLC 文件-2

⑥. 在程序编辑并保存后，所创建的新工程可以在下次重新启动 SoftPLC 后用打开程序的方式进行打开和编辑。

⑦. 打开工程文件的操作步骤即：在文件下拉菜单下选择[打开]选项。选择需要打开的工程文件。

4.2.2 编辑梯形图

概述

新工程被创建后，会出现梯形图编辑界面如图所示，画面左边为编辑向导窗口，可以编辑梯形图的各种功能指令；右边是梯形图描画编程区，用来描画组件和编程。

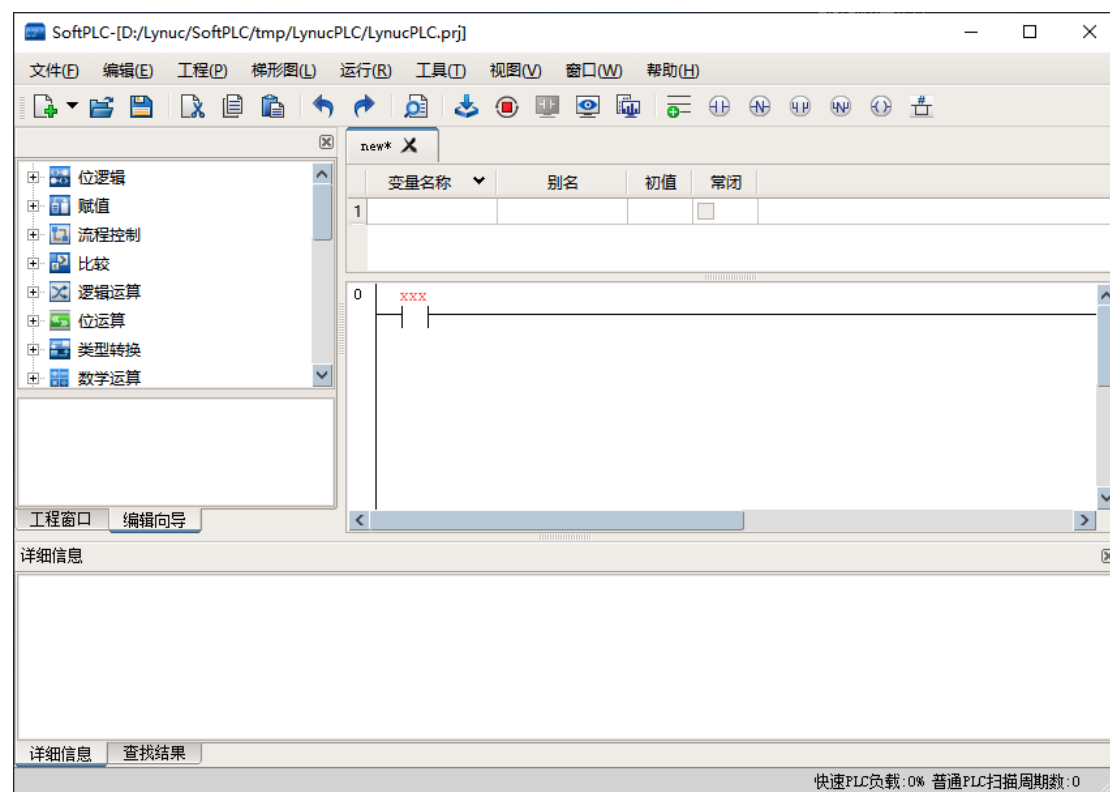


图 4-2-2-1 梯形图编辑界面


梯型图的输入方法有三种：

- 快捷图标输入
- 菜单栏输入
- 编辑向导输入

● 快捷图标输入

利用常用工具栏中的快捷图标进行梯形图编辑。工具条中各种快捷图标所表示的编辑含义如《4.1.2 快捷工具栏》所示。

操作步骤

- ①. 首先在快捷图标工具栏点击[常开触点] ，移动鼠标到梯形图描画区中需要输入的地方，出现黄色方块，表示待输入指令。

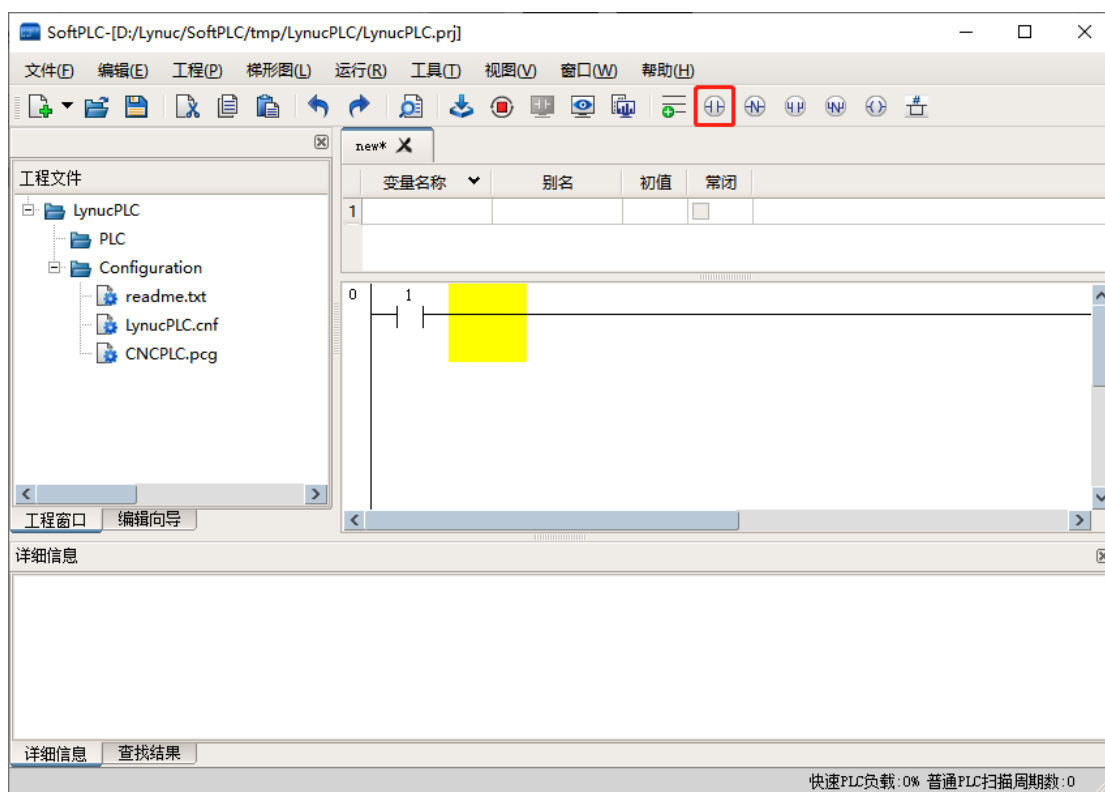


图 4-2-2-2 用快捷图标输入添加常开触点-1

②. 点击鼠标左键确定输入指令。

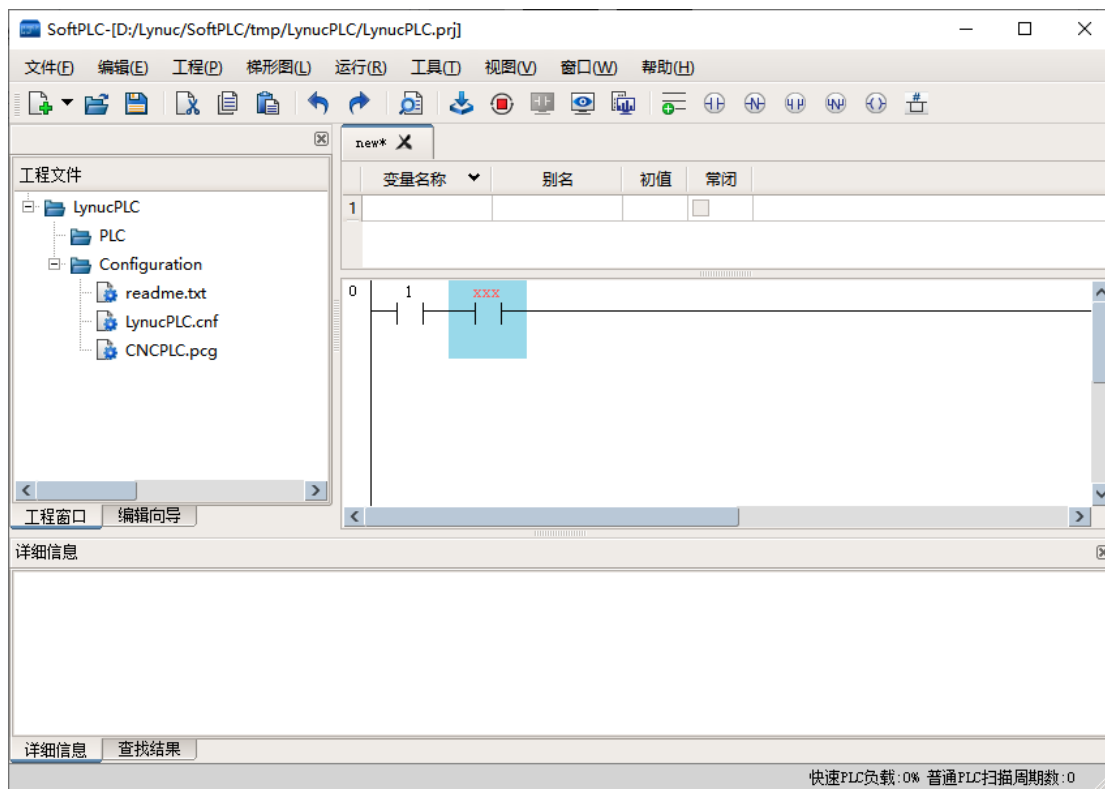


图 4-2-2-3 用快捷图标输入添加常开触点-2

- ③. 这里举例输入基本逻辑控制指令-[常开触点 Q0.1]。双击编辑指令，弹出如下对话框，在变量名栏中输入 Q0.1。

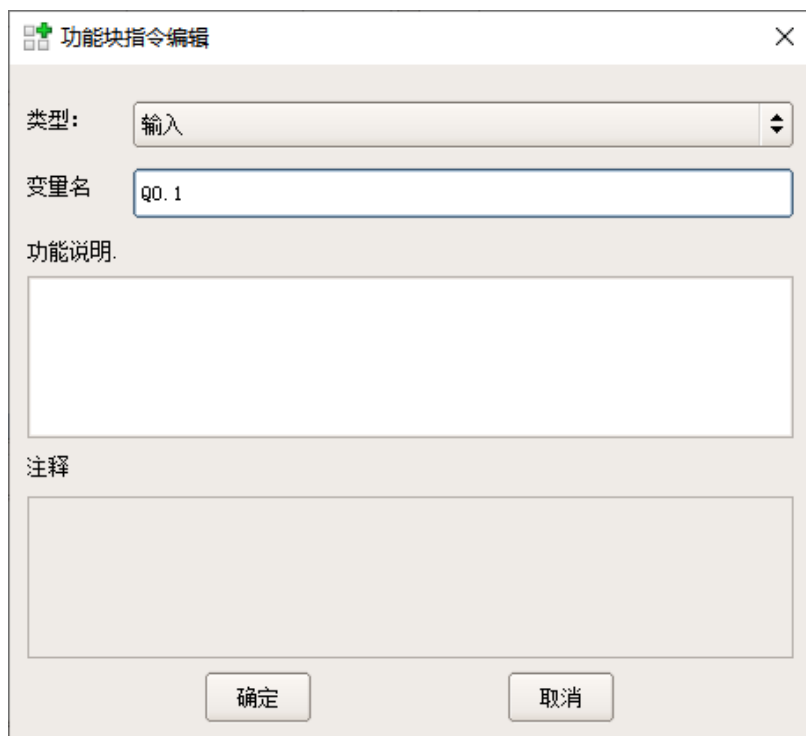


图 4-2-2-4 编辑常开触点对话框

- ④. 输入完成点击确定。梯形图编辑如下所示:

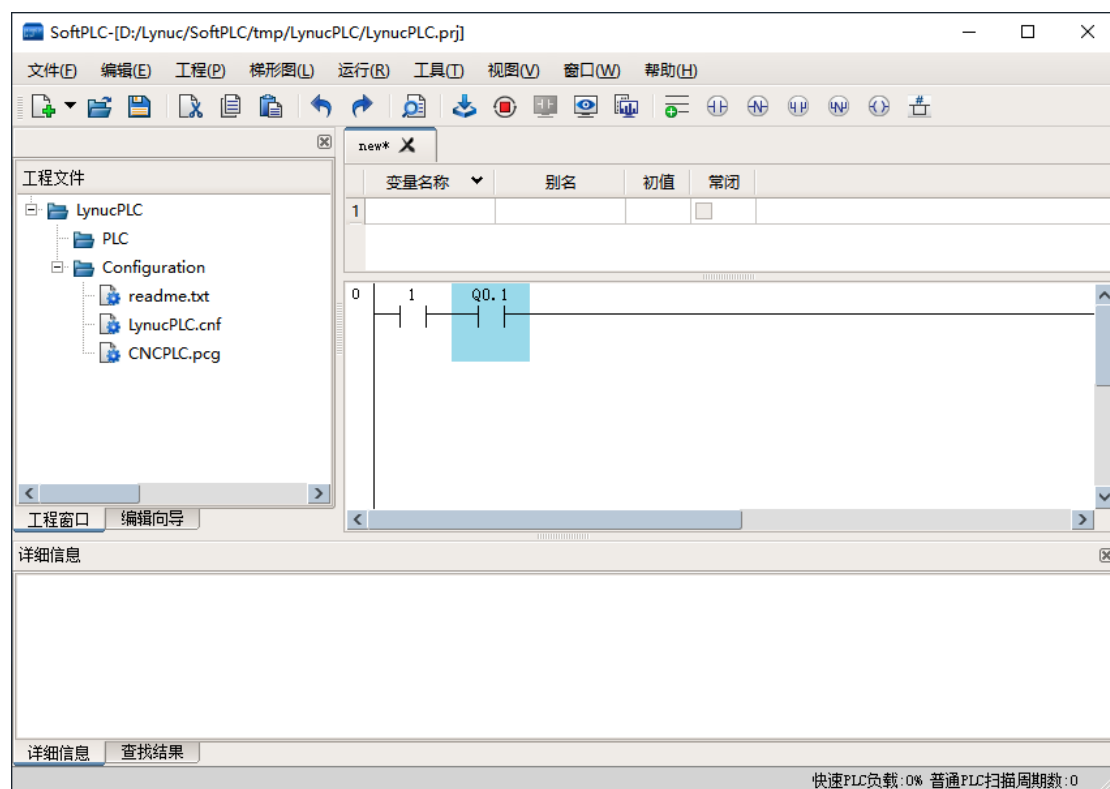


图 4-2-2-5 常开触点编辑完成

※ 注意

1. 如果要修改，则点击需修改的组件，然后右键出现选项如下：

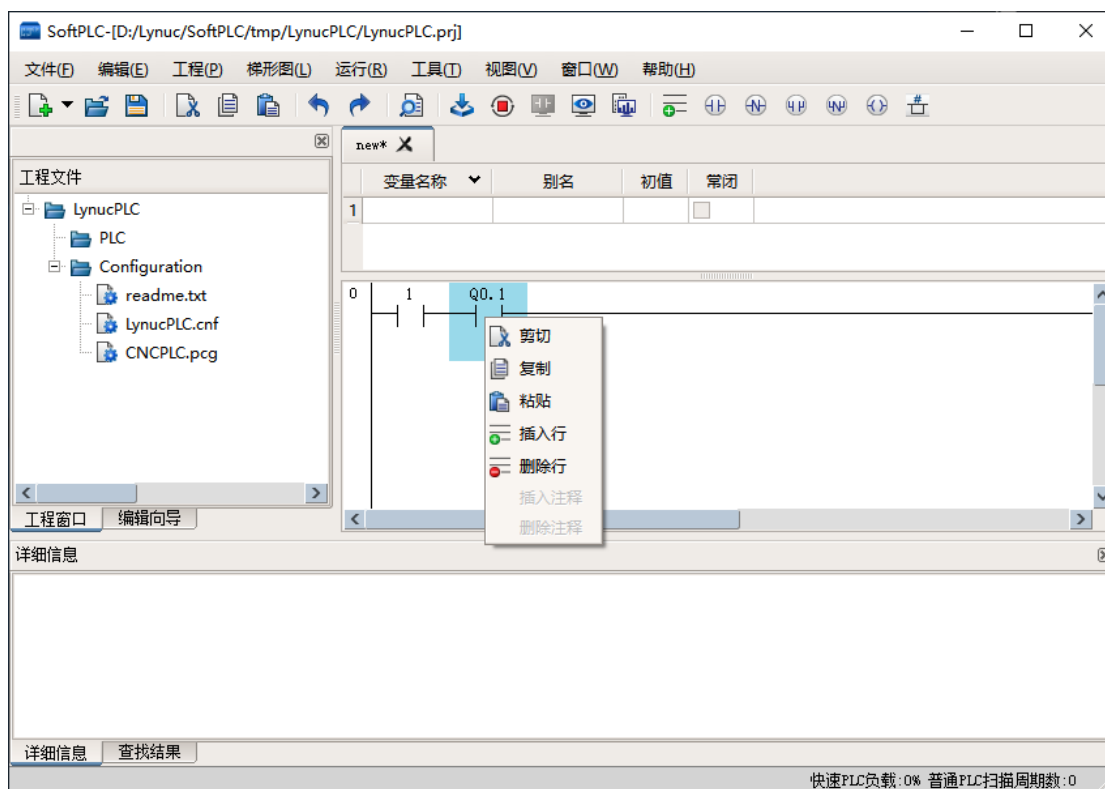


图 4-2-2-6 修改梯形图

2. 若要删除该组件（节点），则先点击 delete 按键，然后将光标移动到需要删除的地方。

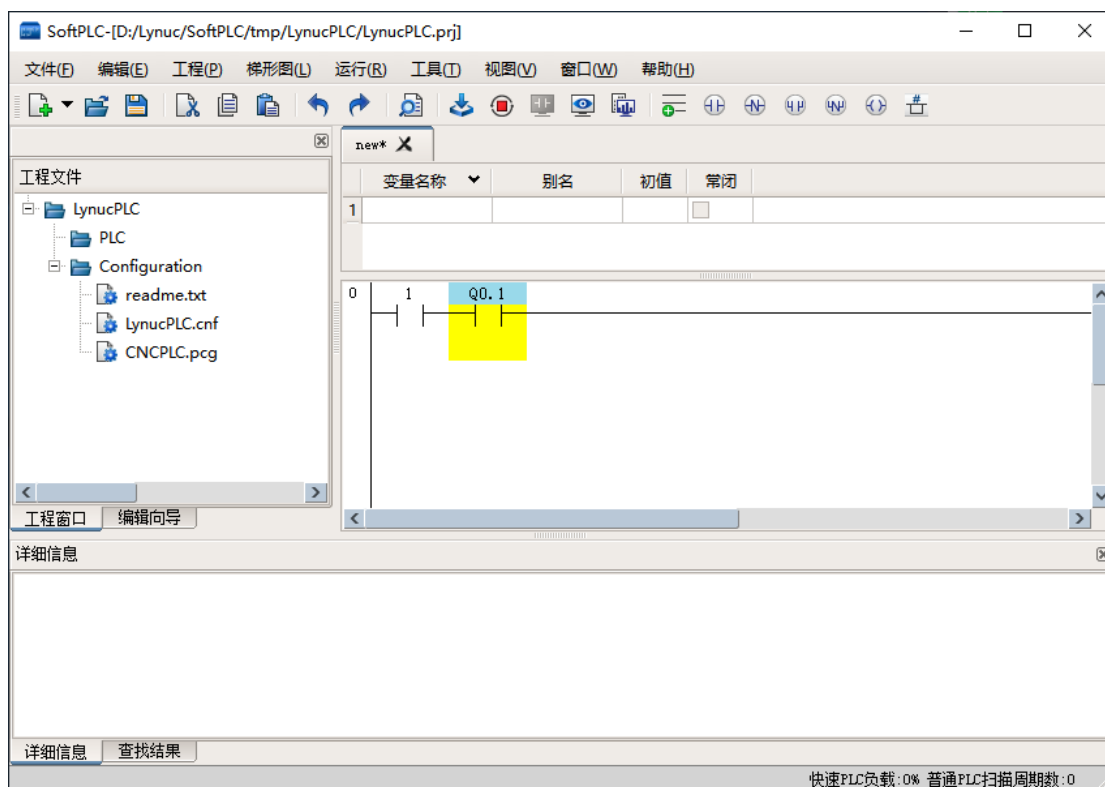


图 4-2-2-7 删除节点

3. 若要取消删除该组件（节点），则点击菜单梯形图-取消删除即取消删除该组件。
4. 快捷图标输入方式优点是工具化，简单便捷。一个对 PLC 不懂的人只要掌握快捷工具栏的各个图标使用，就可以把梯形图完整的输入到程序区。
5. 其他的触点、线圈、指令都可以通过单击相应的快捷图标进行编辑完成。一个完整的应用程序可以通过单击快捷图标来编辑完成。

● 菜单栏输入

- ①. 单击菜单栏中的[梯形图]-[添加输出]-[普通线圈]。

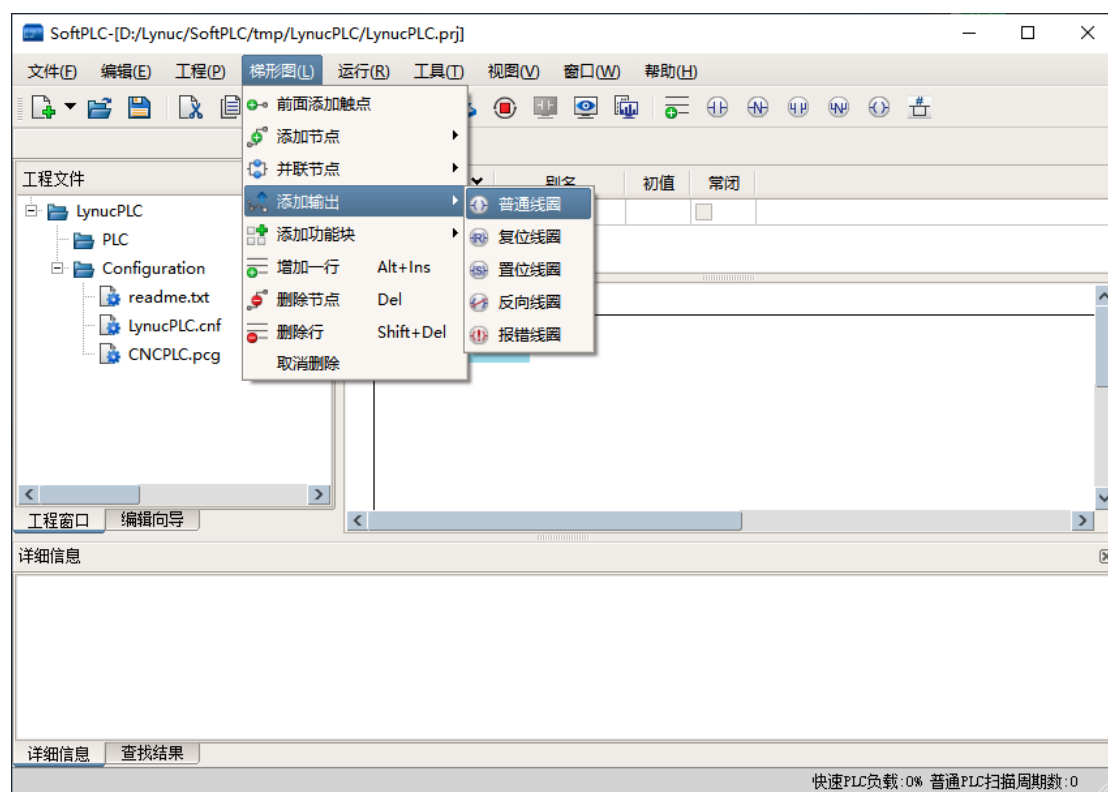


图 4-2-2-8 菜单栏输入添加普通线圈-1

- ②. 在梯形图描画区移动到需要添加的地方。添加完毕，显示如下：

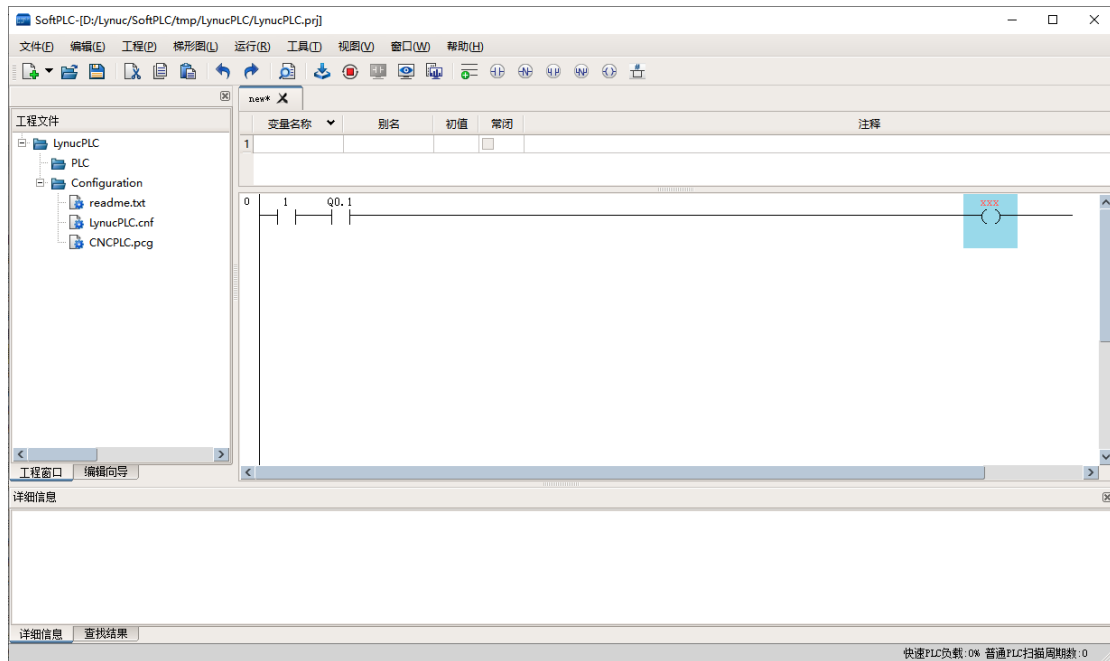


图 4-2-2-9 菜单栏输入添加普通线圈-2

③. 双击该线圈，可以对线圈进行编辑。这里列举输入线圈变量 Q0.0。

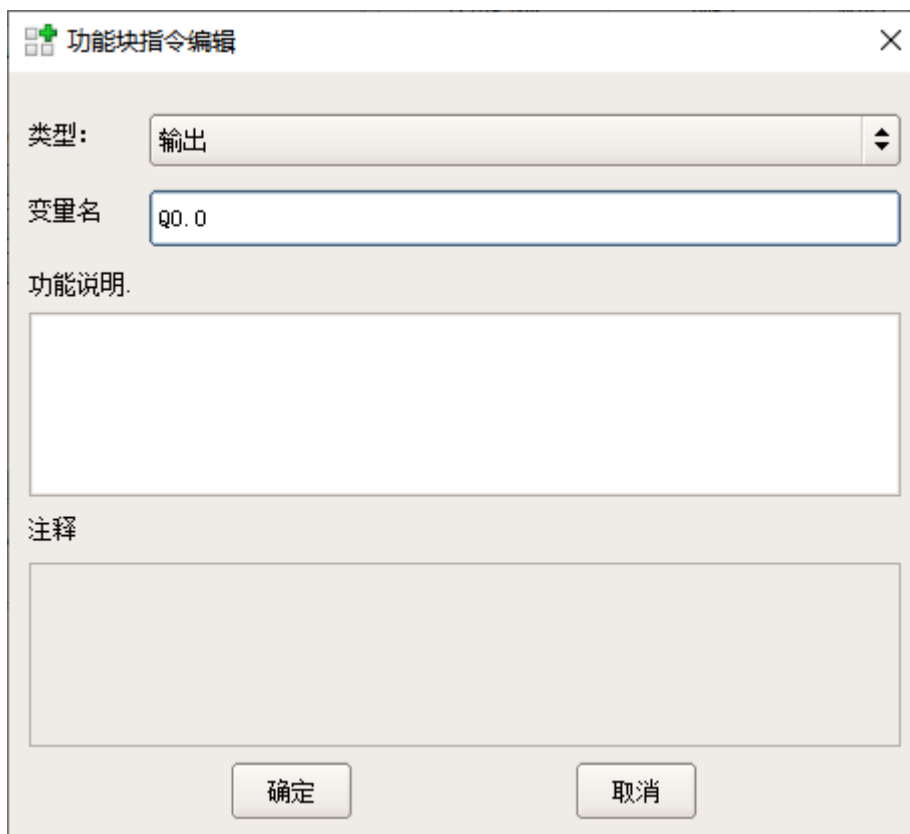


图 4-2-2-10 编辑线圈对话框

④. 编辑完成后单击确定，普通线圈就已经被编辑完成。

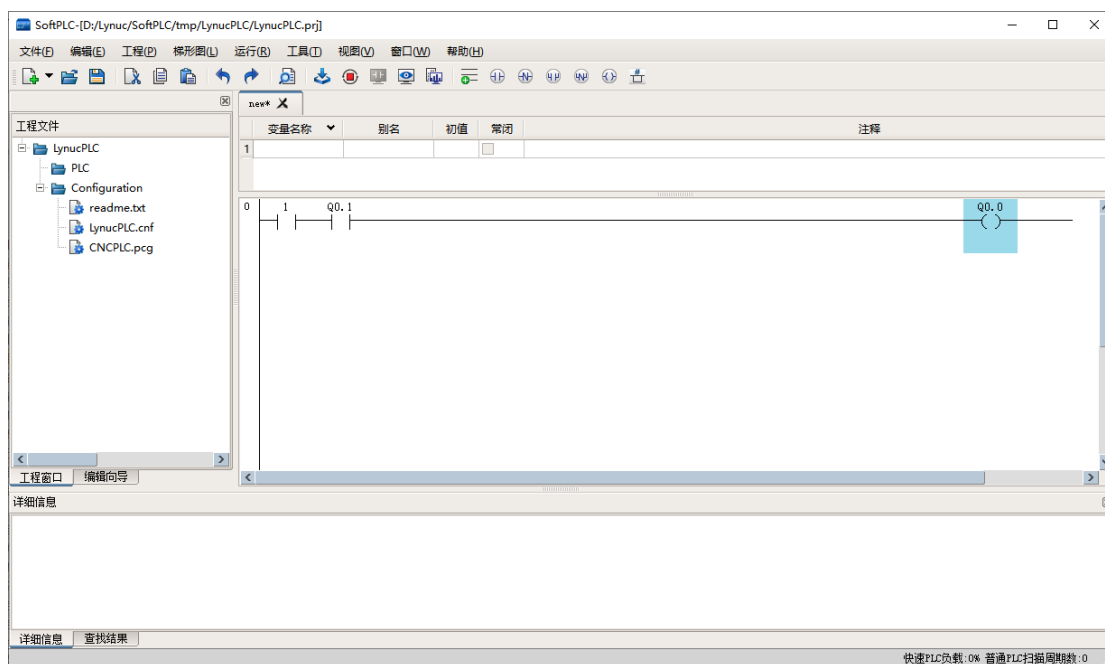


图 4-2-2-11 普通线圈编辑完成

● 编辑向导输入

- ①. 单击编辑向导栏中的[赋值]-[MOVE]功能指令。移动光标到需要添加该指令的地方，点击确定。出现如下所示界面：

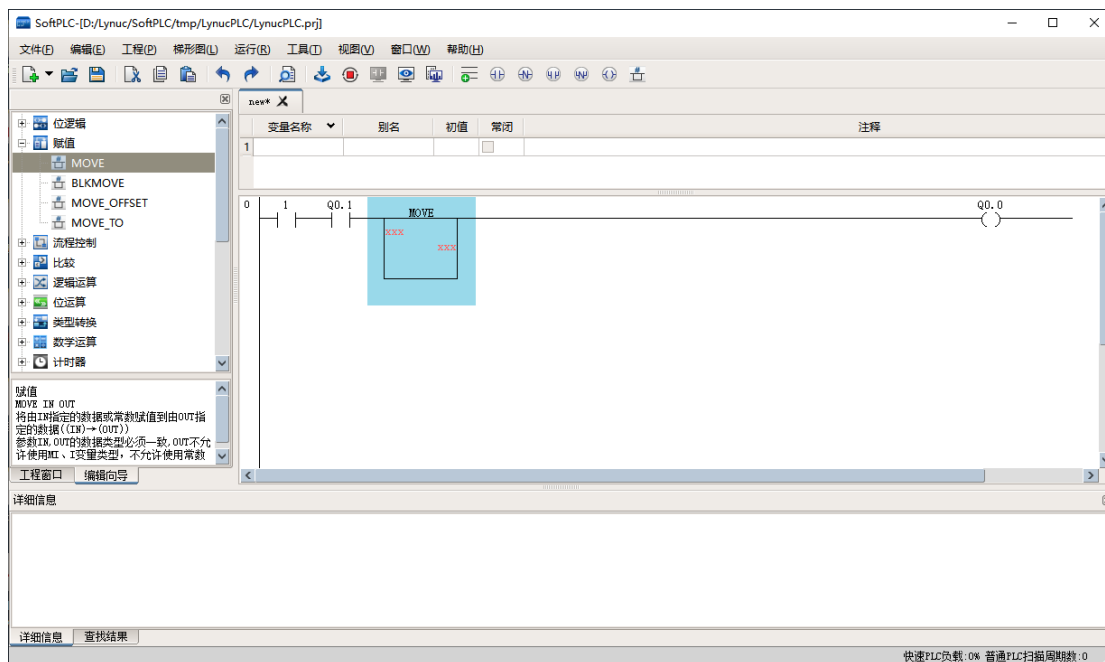


图 4-2-2-12 编辑向导输入 MOVE 功能指令

- ②. 双击该指令，可以对赋值指令进行编辑。这里列举输入指令的变量如下：

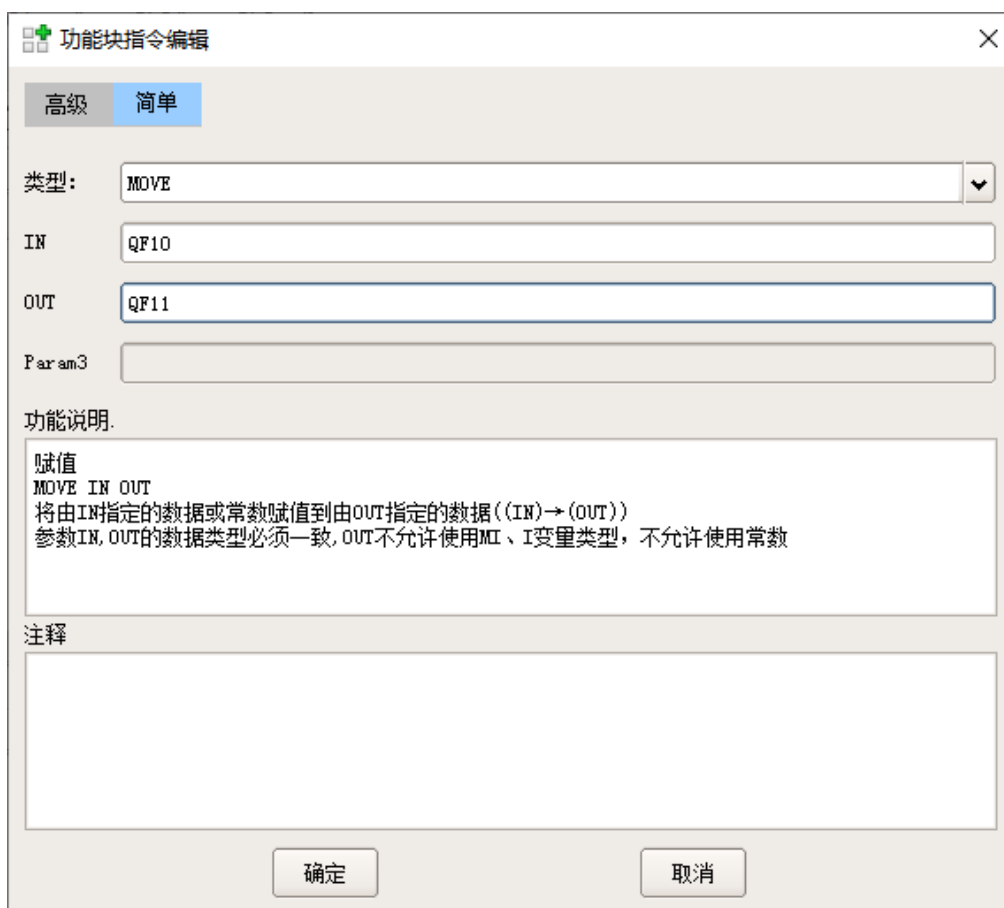


图 4-2-2-13 编辑 MOVE 功能指令对话框

③. 编辑完成后点击确定，则该赋值指令被添加完毕，出现如下界面：

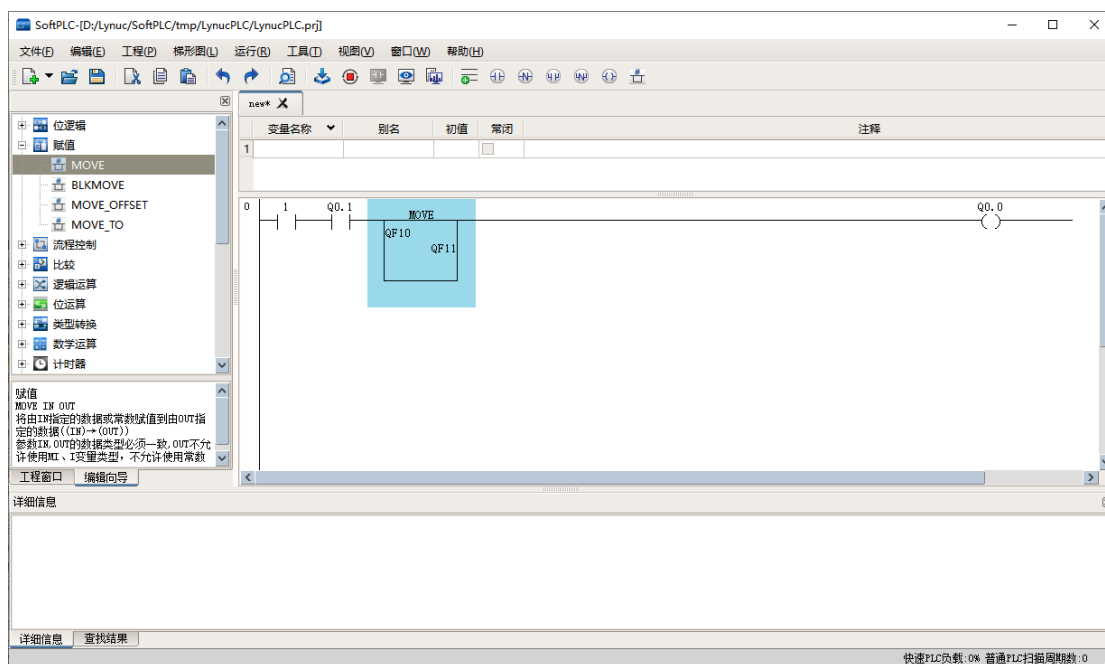


图 4-2-2-14 MOVE 功能指令添加完成

其他操作

1. 增加一行

①. 在菜单栏[梯形图]中，选择[增加一行](Alt+Ins) 选项。

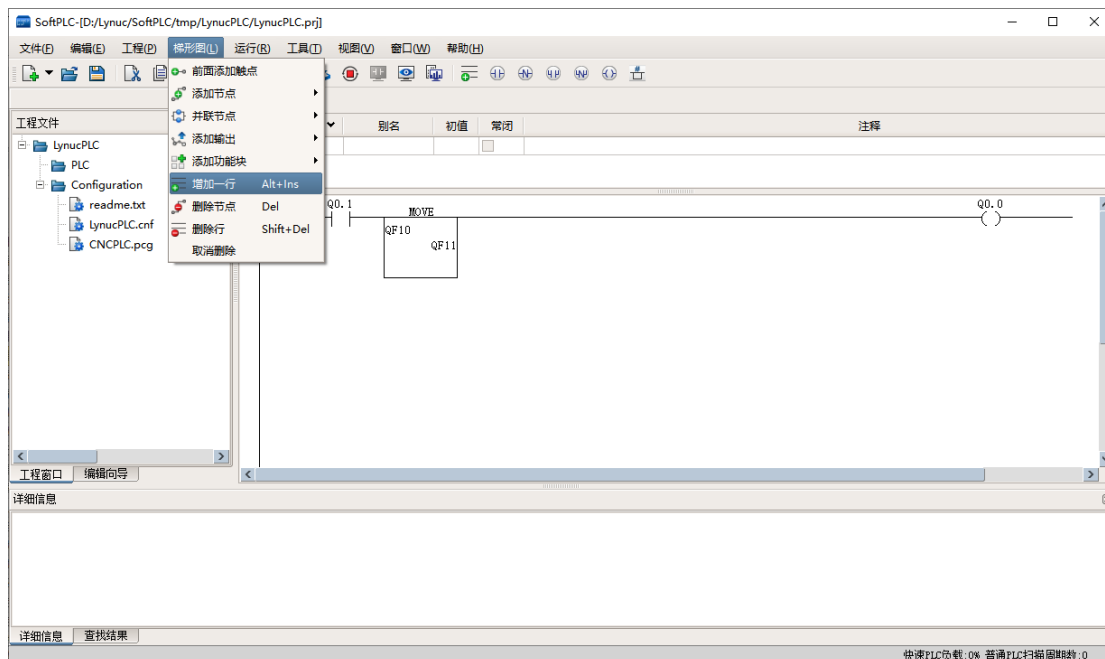


图 4-2-2-15 梯形图编辑：增加行-1

②. 移动光标到梯形图编辑区，确定需要增加一行的位置，显示黄色方块表示待添加行。

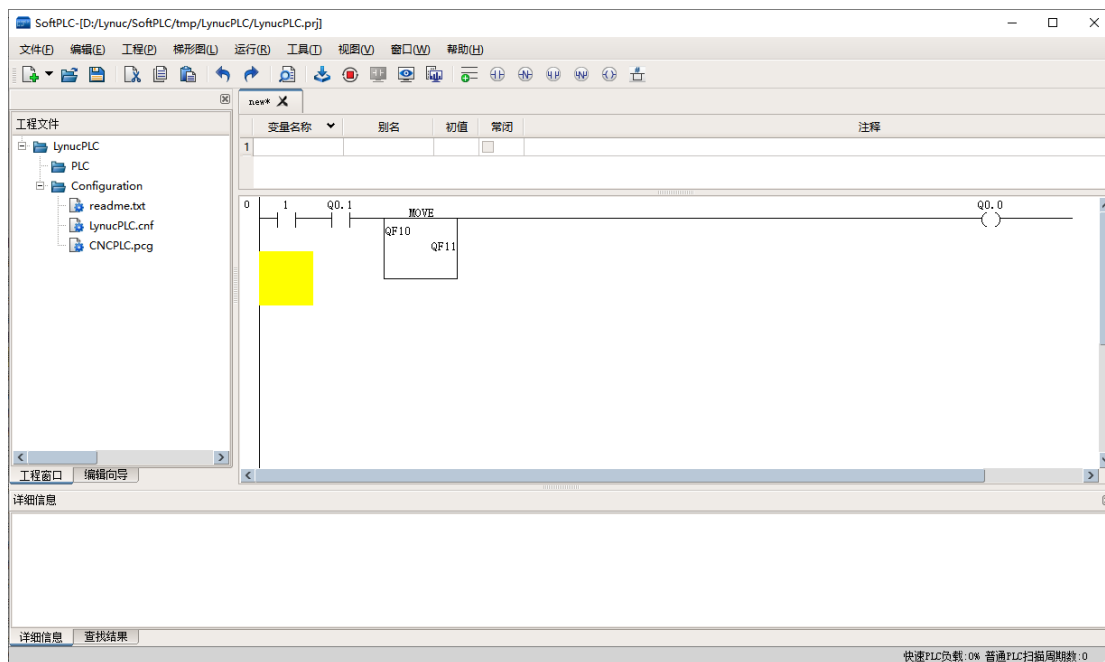


图 4-2-2-16 梯形图编辑：增加行-2

③. 右键确认点击增加一行，增加一行操作完毕，界面显示如下：

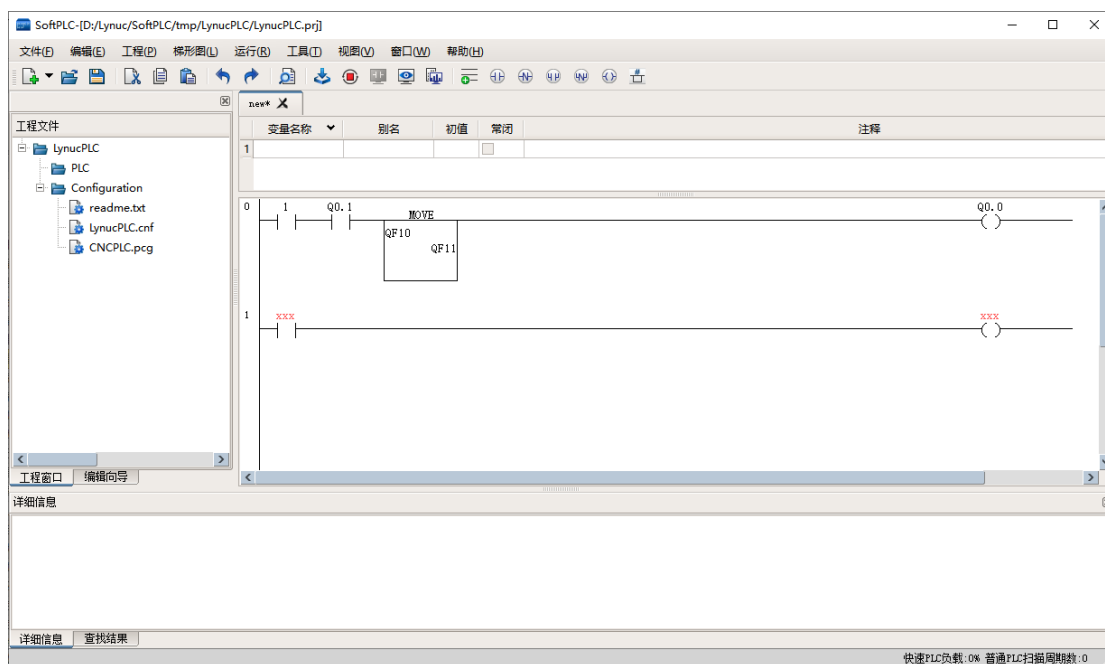


图 4-2-2-17 梯形图编辑：增加行-3

2. 删除行

①. 在菜单栏[梯形图]中，选择[删除行](Alt+Ins),选项。

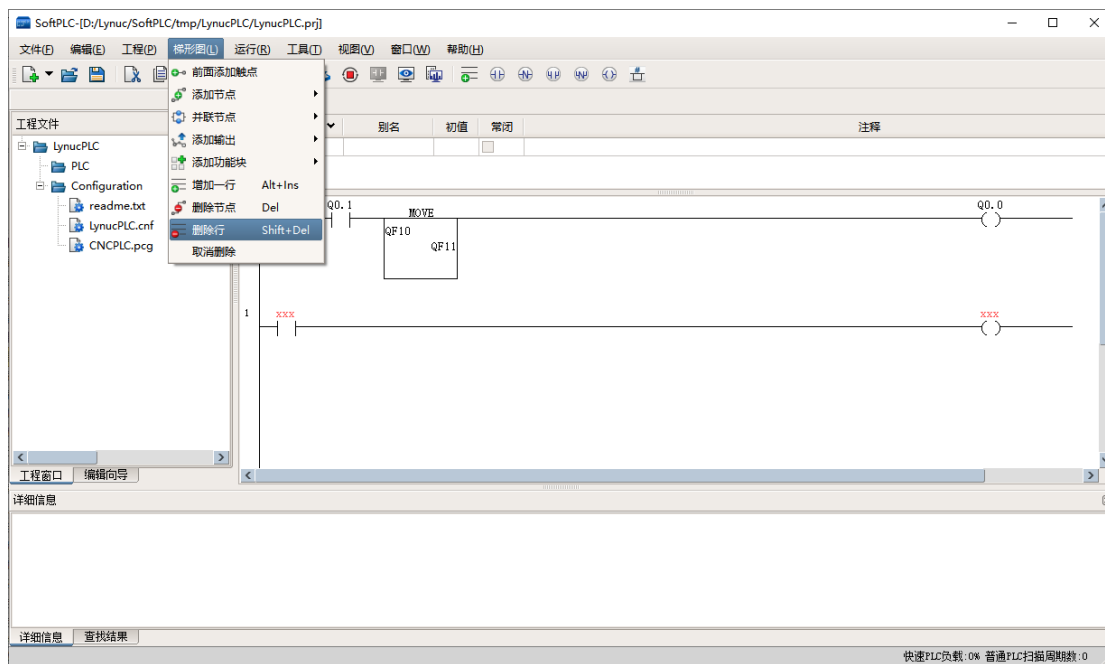


图 4-2-2-18 梯形图编辑：删除行-1

②. 移动光标到需要删除行的位置，黄色方块表示待删除该行。

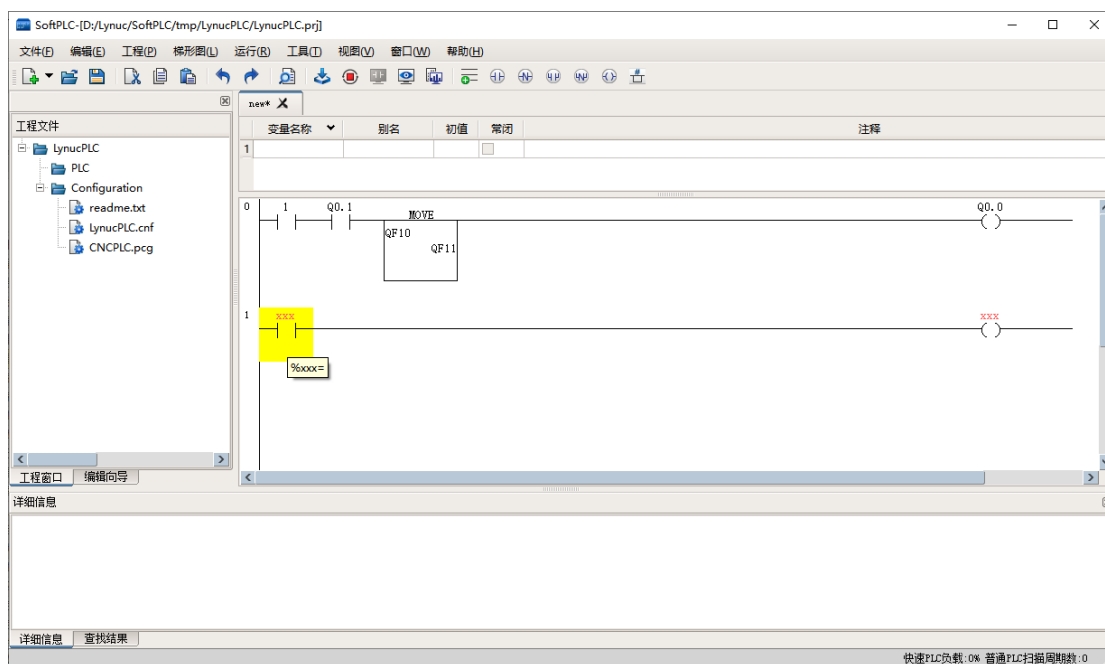


图 4-2-2-19 梯形图编辑：删除行-2

③. 右键单击确认删除该行，删除行的操作完毕，显示界面如下：

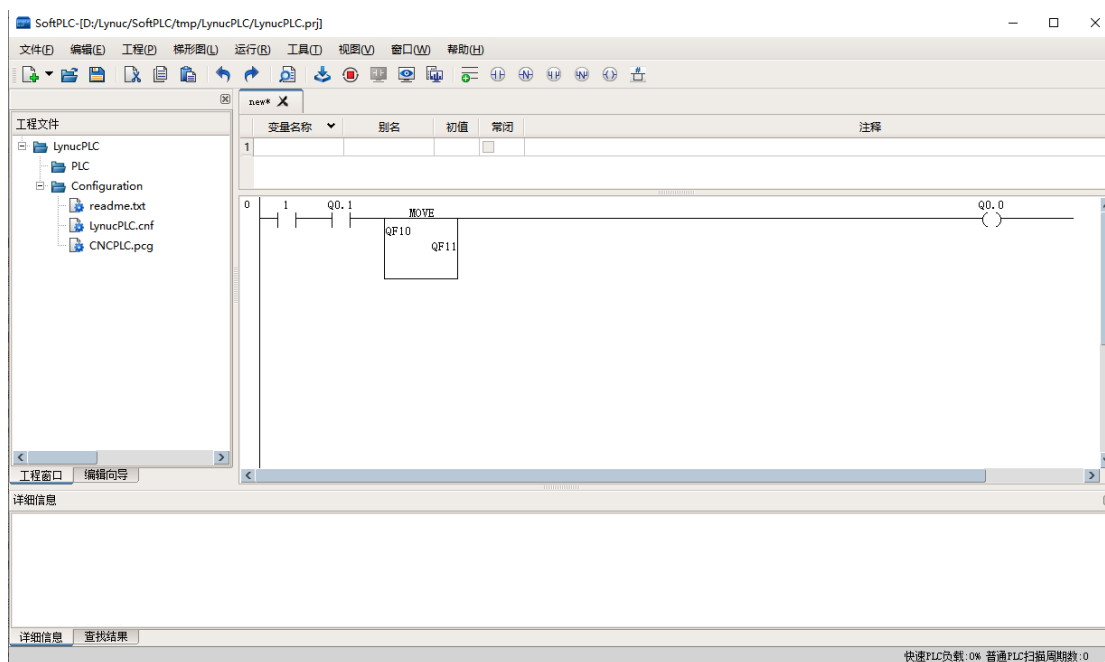


图 4-2-2-20 梯形图编辑：删除行-3

4.2.3 保存工程

概述

保存工程的操作和其他软件操作一样，分为首次新建保存和修改原有文件再保存。

- 新建保存

- ①. 在新建完 PLC 文件后，点击[快捷工具栏]或[菜单栏]的[保存]选项，出现对话框如图所示：

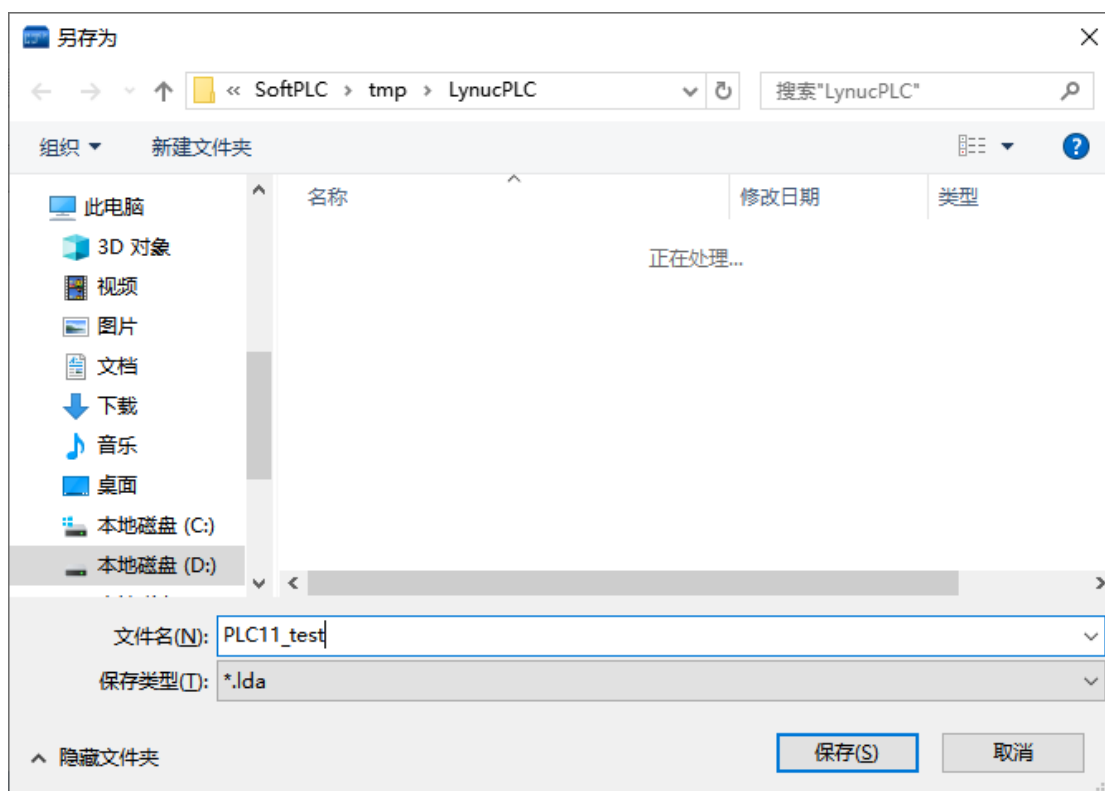


图 4-2-3-1 保存工程

- ②. 编辑[文件名]，单击[保存]按钮。完成梯形图文件的保存。

- 修改保存

直接点击快捷工具栏的[保存]按钮  即可。

4.2.4 程序注释

概述

梯形图描画完后，如果不加以注释，在一定时间后会很难理解。由于梯形图程序可读性较差，加上程序编制因人而异，完成同样的控制功能往往有多种不同的编程方法，所以给程序加上注释，可增加程序的可读性，方便交流和对程序的修改。

操作步骤

- ①. 首先打开某个 PLC 文件，然后点击需要添加注释的指令，显示蓝色方块。

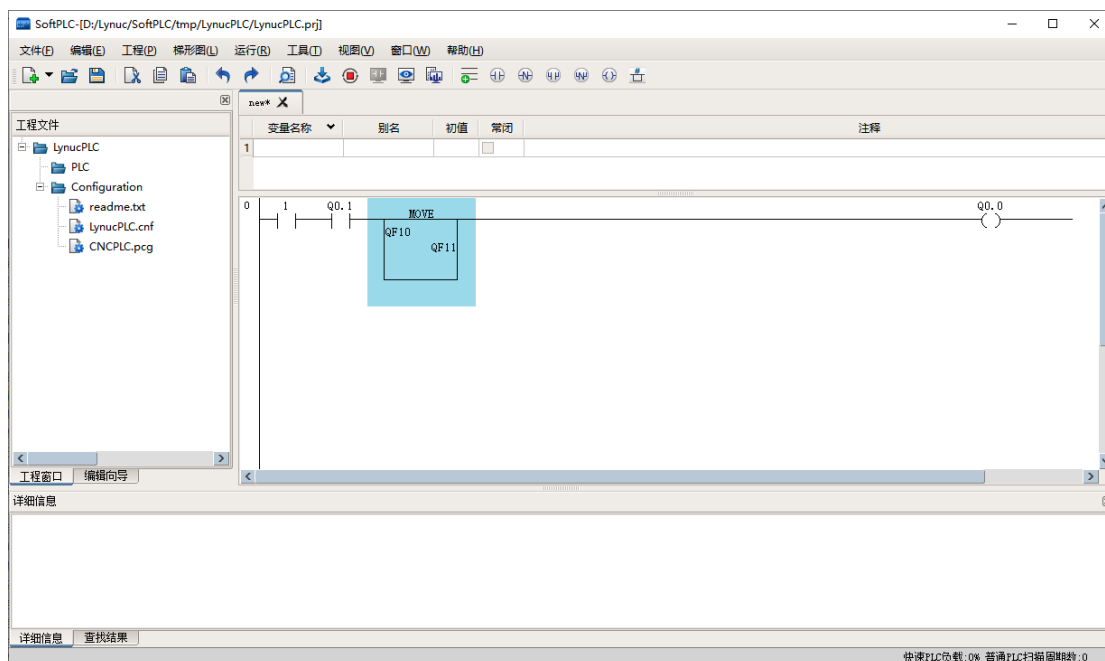


图 4-2-4-1 添加功能块指令注释

- ②. 双击该指令，出现如下窗口。在注释栏输入对该指令的注释。

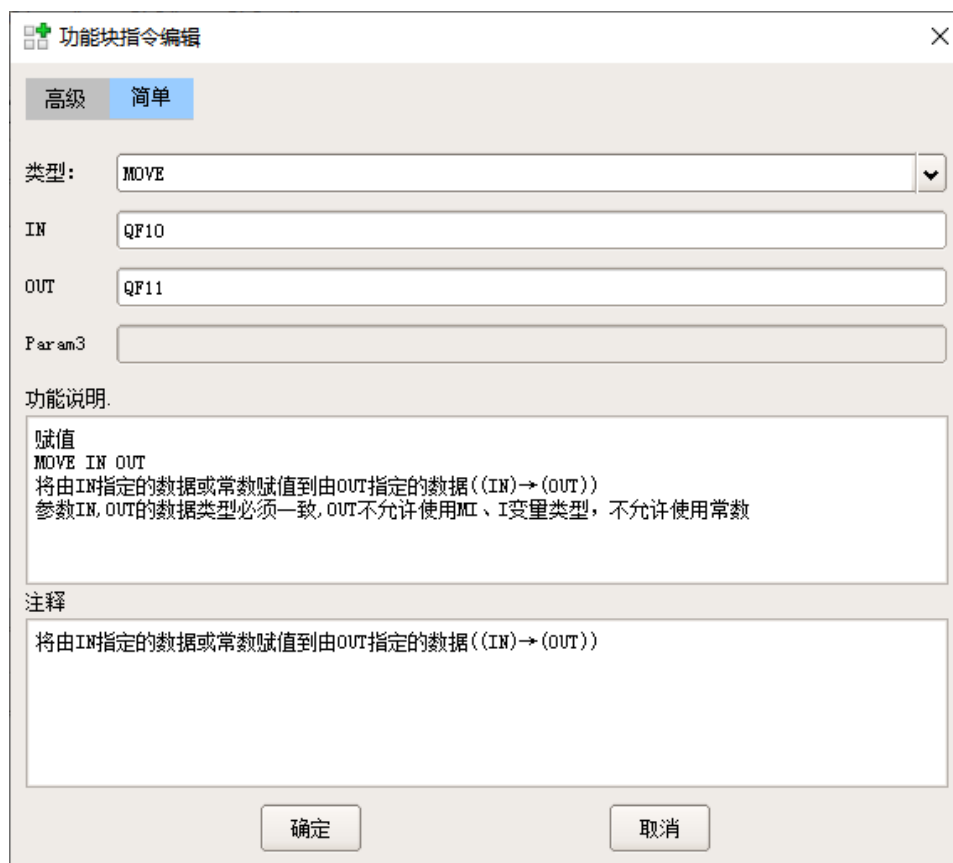


图 4-2-4-2 编辑功能块指令注释

③. 注释输入完毕后，点击确定，注释添加完毕，显示界面如下：

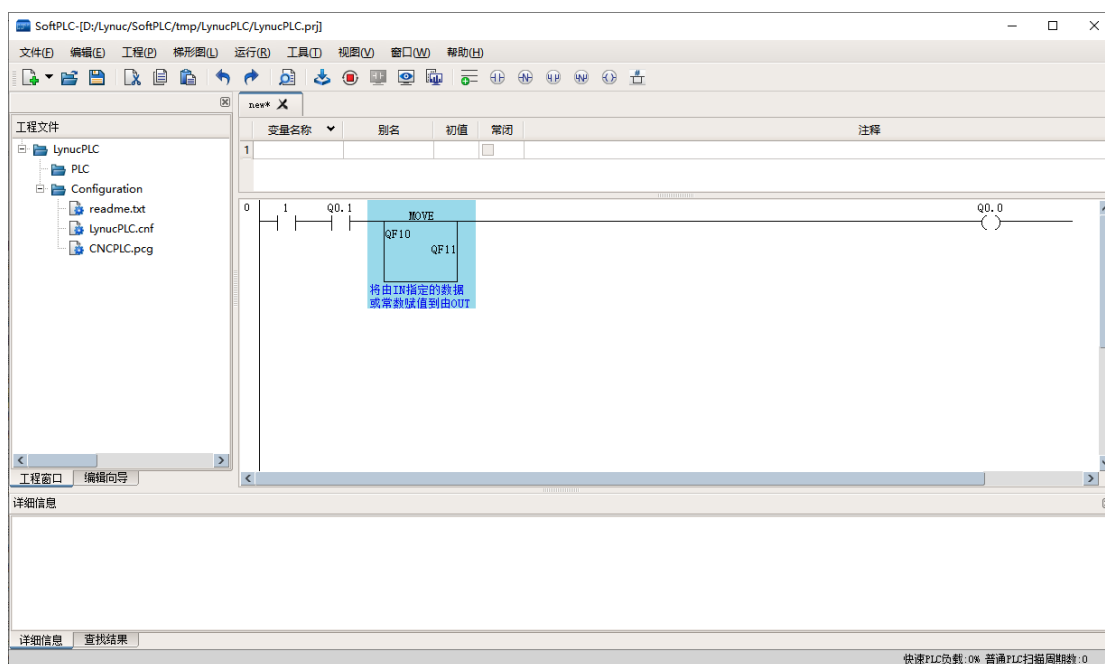


图 4-2-4-3 功能块指令注释添加完成

④. 一行梯形图也可以添加注释，首先右键选中一行，并弹出操作菜单如下图



图 4-2-4-4 添加梯形图注释

⑤. 选择插入注释，操作后如下图

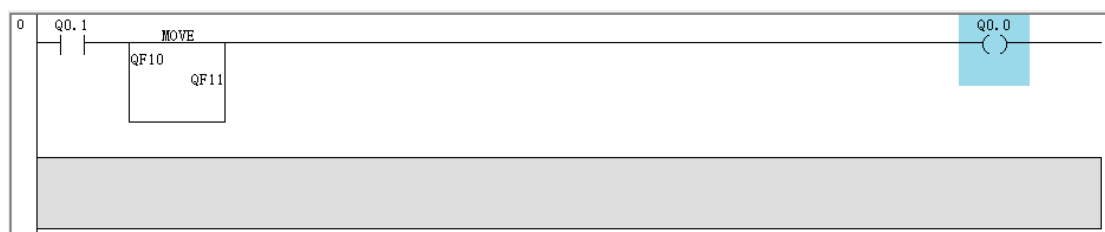


图 4-2-4-5 插入梯形图注释

⑥. 点击灰色区域，编辑注释内容

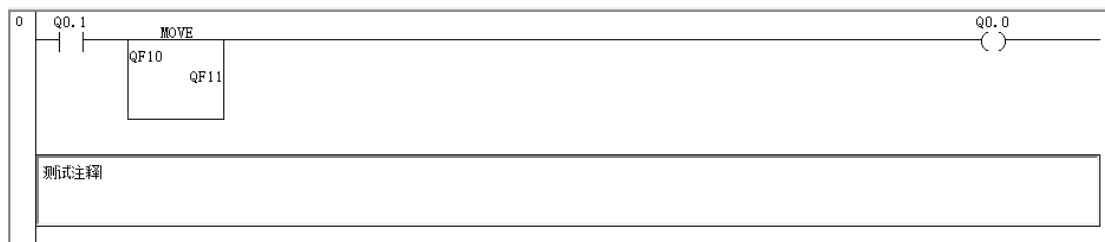


图 4-2-4-6 编辑梯形图注释

⑦. 点击其他地方，注释编辑完成，显示如下图

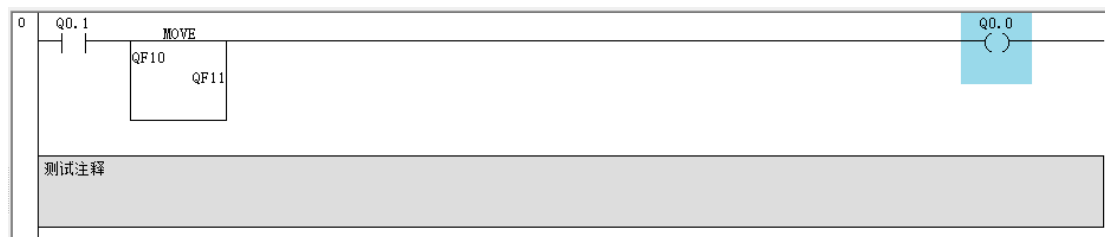


图 4-2-4-7 梯形图注释添加完成

4.2.5 生成程序文件

概述

编程语言是指用户与 PLC 进行信息交换的方法。在此介绍生成各种不同程序文件的编程格式。

- IL 文件格式（指令表）
- LDA 文件格式

表 4-2-5-1 程序文件说明

编程格式	说明
IL 文件格式	变量定义 VAR_INPUT END_VAB VAR_OUTPUT END_VAR VAR_IN_OUT END_VAR

	VAR fgh AT %LW0 : INT ; (*ghfghfh*) END_VAR 程序体是标准 IL 语言格式
LDA 文件格式	LDA 文件格式是指梯形图文件格式。

4.2.6 加载当前 PLC 程序

概述

点击加载 PLC 按钮或直接点击 Ctrl+D，界面弹出对话框询问用户是否想将当前 PLC 加载到第几个 PLC 中去运行，如下图：



图 4-2-6-1 加载 PLC 对话框

说明

- 1) 加载文件：表示所加载的文件名；
- 2) 加载序号：表示 PLC 文件扫描执行的顺序编号；
- 3) 加载类型：是否立即扫描；
- 4) 若勾选立即扫描，则出现如下窗口，点击确定则开始加载该 PLC。



图 4-2-6-2 加载 PLC 对话框-重新下载并运行

- 5) 若不勾选，选择加载，则出现如下窗口。



图 4-2-6-3 加载 PLC 对话框-加载前停止运行

6) 点击确定后,可以在工程窗口发现该PLC由绿色转变为黄色。表示目前没有运行该PLC。

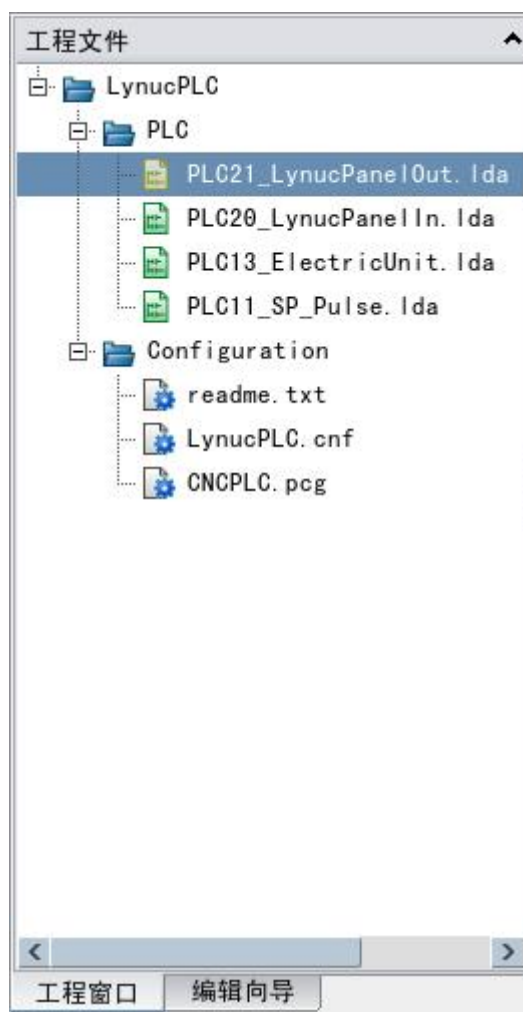


图 4-2-6-4 工程窗口: PLC 文件已加载但未运行

7) 扫描方式: 是否为快速 PLC 扫描。

若勾选快速 PLC, 则 PLC 在每个扫描周期都会扫描执行该 PLC 文件。

若不勾选快速 PLC, 则其按照普通 PLC 方式扫描执行。

- 8) 在 PLC 的每个扫描周期先扫描执行快速 PLC,快速 PLC 结束后再去扫描执行普通 PLC,每扫描完成一个普通 PLC 都会去判断当前已使用的时间是否在限定时间内,若在限定时间内则继续扫描执行下一个普通 PLC,否则停止扫描执行 PLC 文件,等下一个周期扫描完成快速 PLC 后继续扫描执行接下来的普通 PLC。若 PLC 文件过多或者某个文件逻辑过于复杂,所有的普通 PLC 扫描完成一遍可能需要多个 PLC 周期,因此在 softplc 界面的状态栏的右边显示有快速 PLC 负载百分比和普通 PLC 扫描周期数。

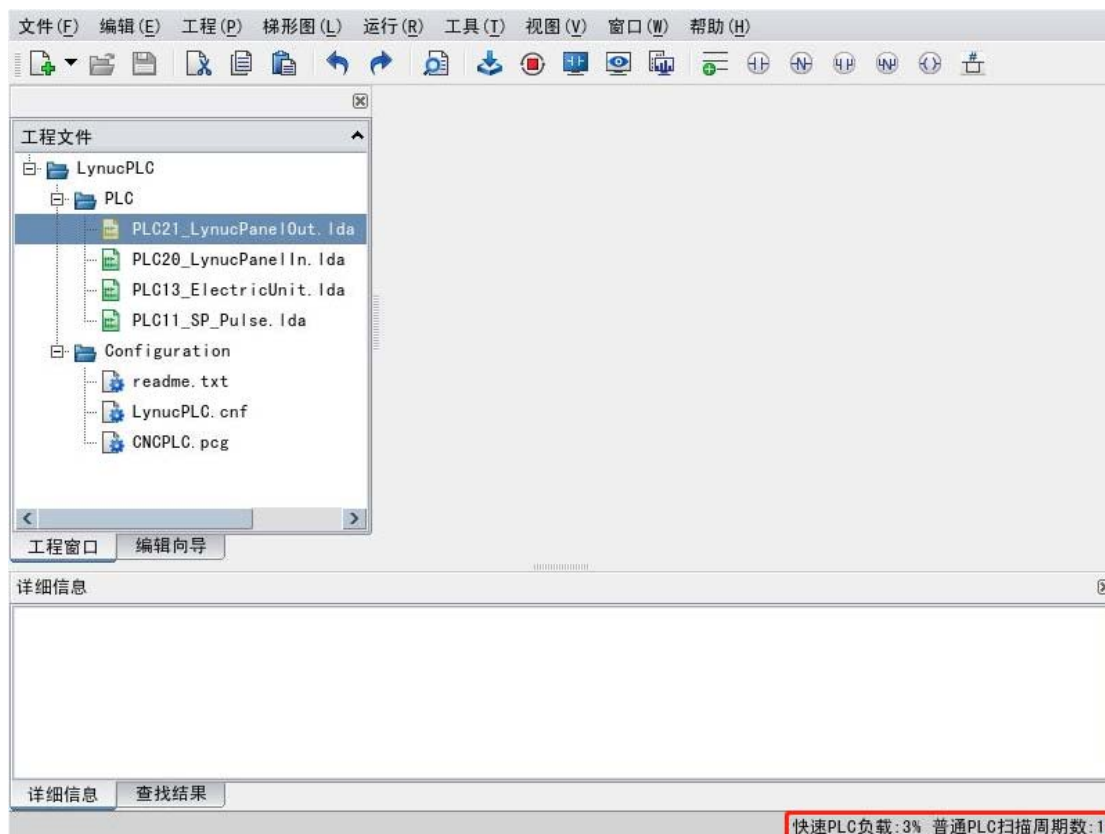


图 4-2-6-5 快速 PLC 负载百分比和普通 PLC 扫描周期数

- 9) 当快速 PLC 负载超过 100%时说明 CPU 占用率稍高,此时应减少快速 PLC 的个数,使得快速 PLC 负载百分比在 100%以下;同时要关注普通 PLC 扫描周期数,若普通 PLC 扫描周期数大于 1,则表明有些普通 PLC 需要大于等于两个 PLC 扫描周期才会扫描执行到。这时若普通 PLC 含有计时器或者计数器,对应的逻辑将不再准确,计时器会有延时,计数器则会乱掉。

4.2.7 运行加载的 PLC 程序

概述

在 4-2-6 章节中加载完当前的 PLC 程序后,底部的信息窗口会弹出信息提示运行加载的 PLC 程序。

- ①. 点击常用工具栏 Run 或 Ctrl+R 出现对话框如下所示, 显示了所有可运行的 PLC 程序。



图 4-2-7-1 PLC 状态显示对话框-1

- ②. 选中 PLC 程序并点击 OK, 则选中的 PLC 开始运行。



图 4-2-7-2 PLC 状态显示对话框-2

- ③. 当开始运行加载的程序后，工程窗口的 PLC 由黄色转变为绿色。表示目前正在运行加载的 PLC 程序。

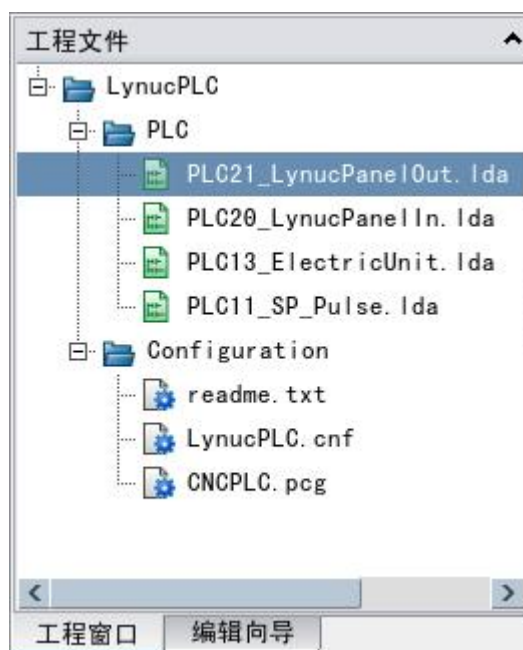


图 4-2-7-3 工程窗口：正在运行加载的 PLC 程序

4.2.8 查看 PLC 工程中的变量

概述

在线运行 PLC 工程后，为了便于检查和调试 PLC 逻辑，可以查看变量的值。

操作步骤

- ①. 点击菜单栏[工具]-[查看]选项。

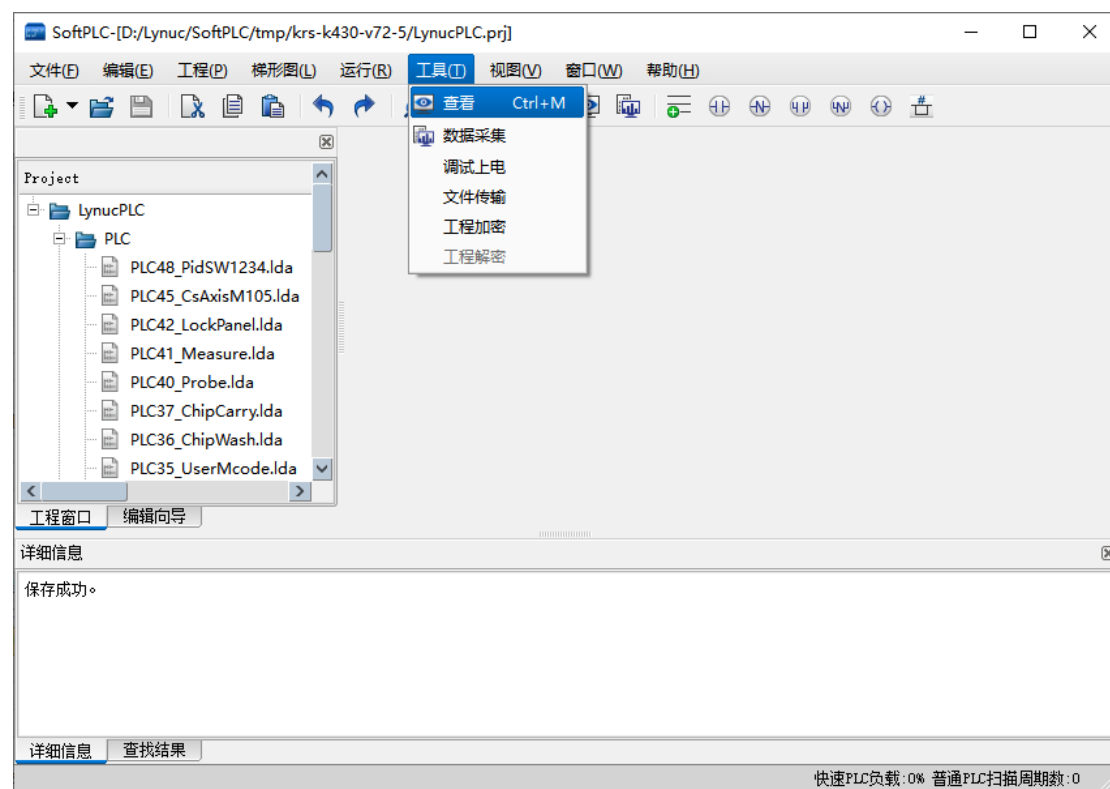


图 4-2-8-1 查看 PLC 工程中的变量

- ②. 出现界面如下，编辑需要查看的变量。



图 4-2-8-2 查看变量对话框

- ③. 添加变量是指将变量加入到查看列表中，删除变量是指从查看列表中删除变量，刷新是使查看的变量的值每 0.5s 刷新一次
- ④. 若需退出查看 PLC 工程界面，点击[关闭]即可。
- ⑤. 查看对话框下面显示当前打开的 PLC 文件以及该文件所加载的序号，便于使用者核实所查看的局部变量，例如 Q 变量是每个 PLC 程序对应 20 个，不同的程序即使相同的 Q 变量编号，其实际的地址也是不同的。

4.2.9 数据采集

概述

数据采集功能是指将变量在一段时间内的值描画成一条曲线，通常在变量有一个突变又回到原值，查看无法发现时使用，具体的操作步骤如下。数据采集已经实现循环采集，每次更新时显示的曲线是根据变量在更新前 8 秒内的值描画出来的。

操作步骤

①. 点击菜单栏[工具]-[数据采集]选项。

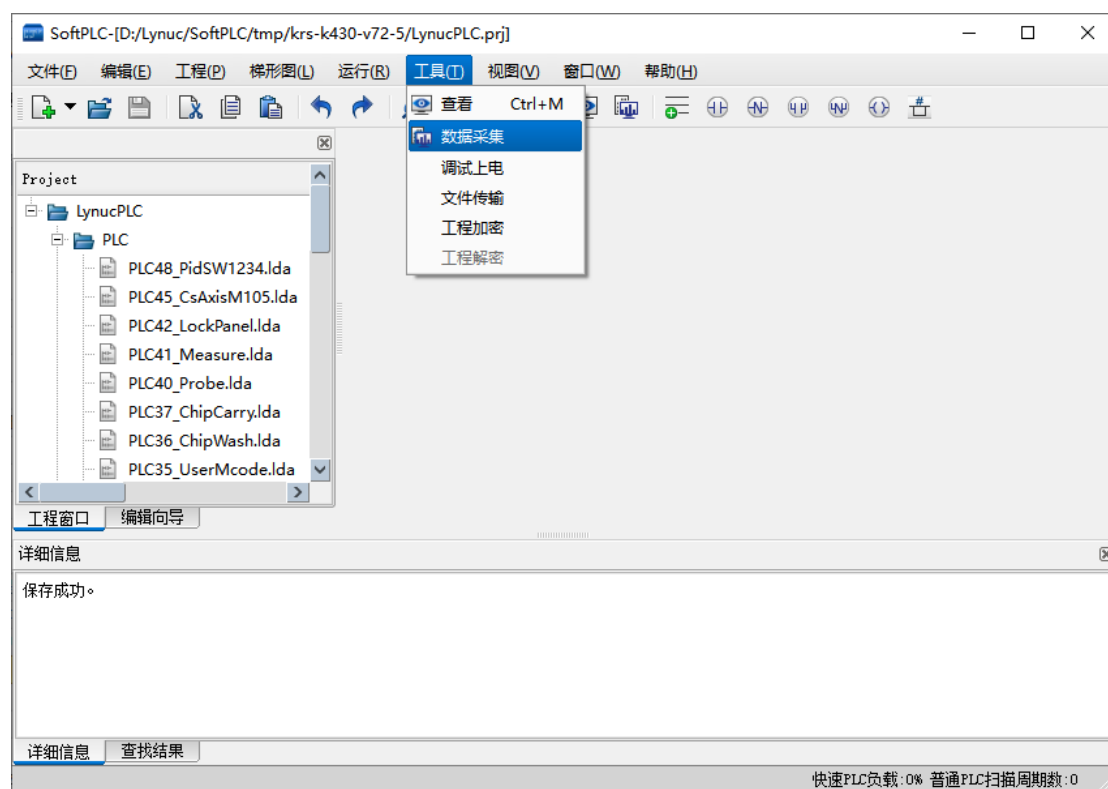


图 4-2-9-1 打开数据采集

②. 出现如下数据采集界面。



图 4-2-9-2 数据采集对话框

- ③. 该对话框可以选择读取内存来采集数据也可以从本地文件中显示曲线，点击读取文件，弹出文件选择对话框，选取对应的 cvs 文件点击打开按钮，文件的全路径名称显示在下面的文本框内，点击更新，会显示文件内的变量对应的曲线。以下步骤介绍读取内存的信息采集步骤。



图 4-2-9-3 数据采集-读取文件

- ④. 勾选需要填写变量名称对应的选项框，输入需要采集的变量名称，依次按步骤点击[加载变量]-[开始采集]-[停止采集]-[更新]按钮。对话框最下面会有每次按下按钮后的消息提示。



图 4-2-9-4 数据采集-加载变量

⑤. 最后点击更新后数据采集结果界面如下图。

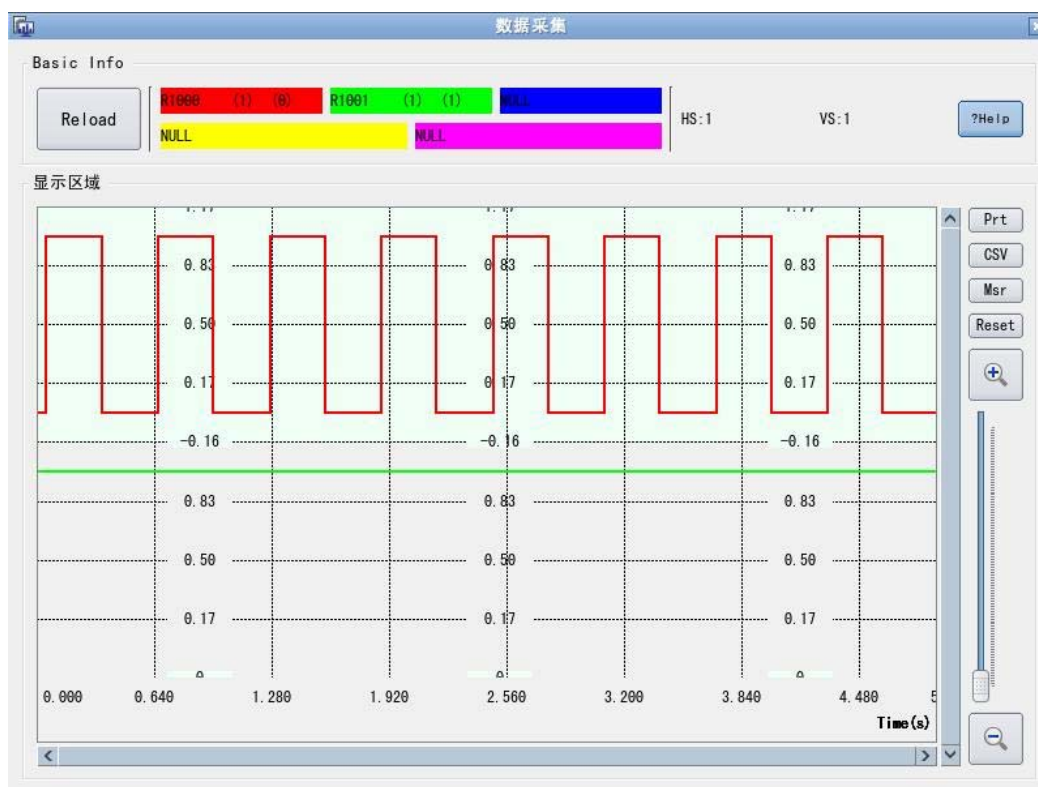


图 4-2-9-5 数据采集结果界面

※ 注意

1. 数据采集界面上面区域是基本信息，Reload 按钮可以调出刚才填写变量的对话框，右边是所查找变量以及变量对应的最大值和最小值，再右边 HS 为水平拉伸比例，VS 为垂直拉伸比例，最右边是帮助按钮。点击 Reload 按钮弹出如下界面：

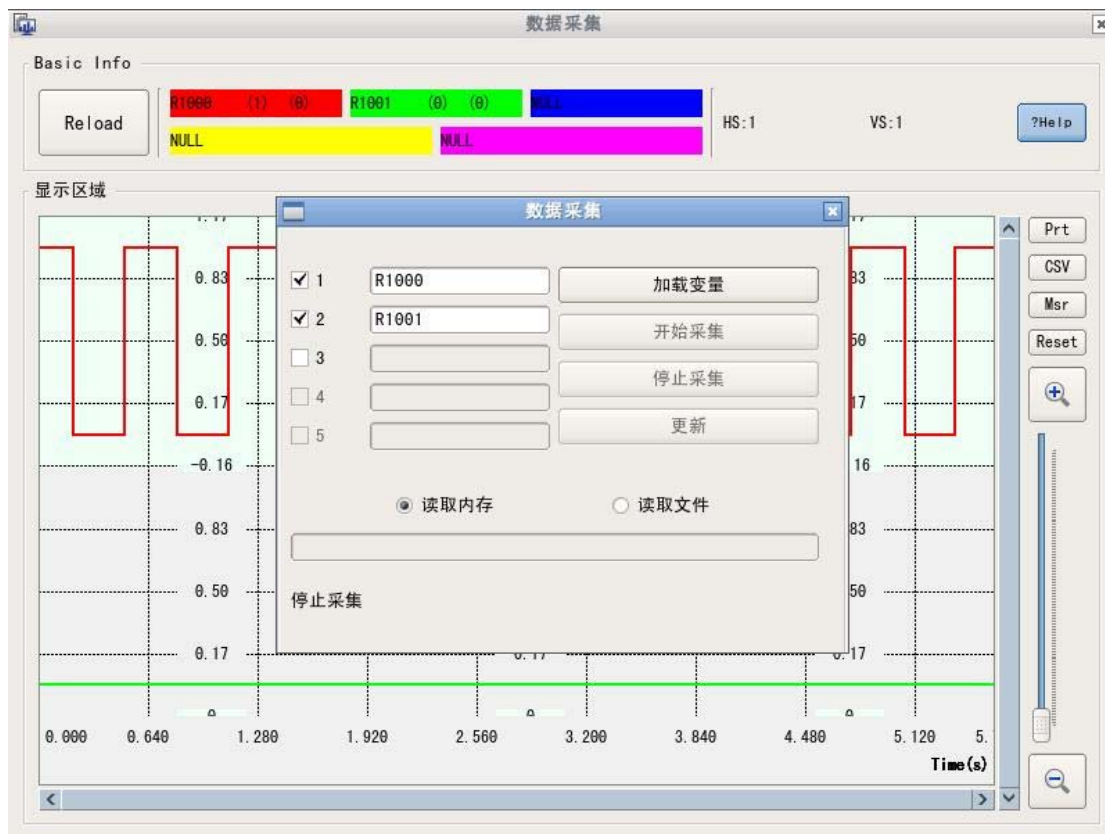


图 4-2-9-6 数据采集-重新加载对话框-

2. 点击帮助按钮弹出帮助对话框。

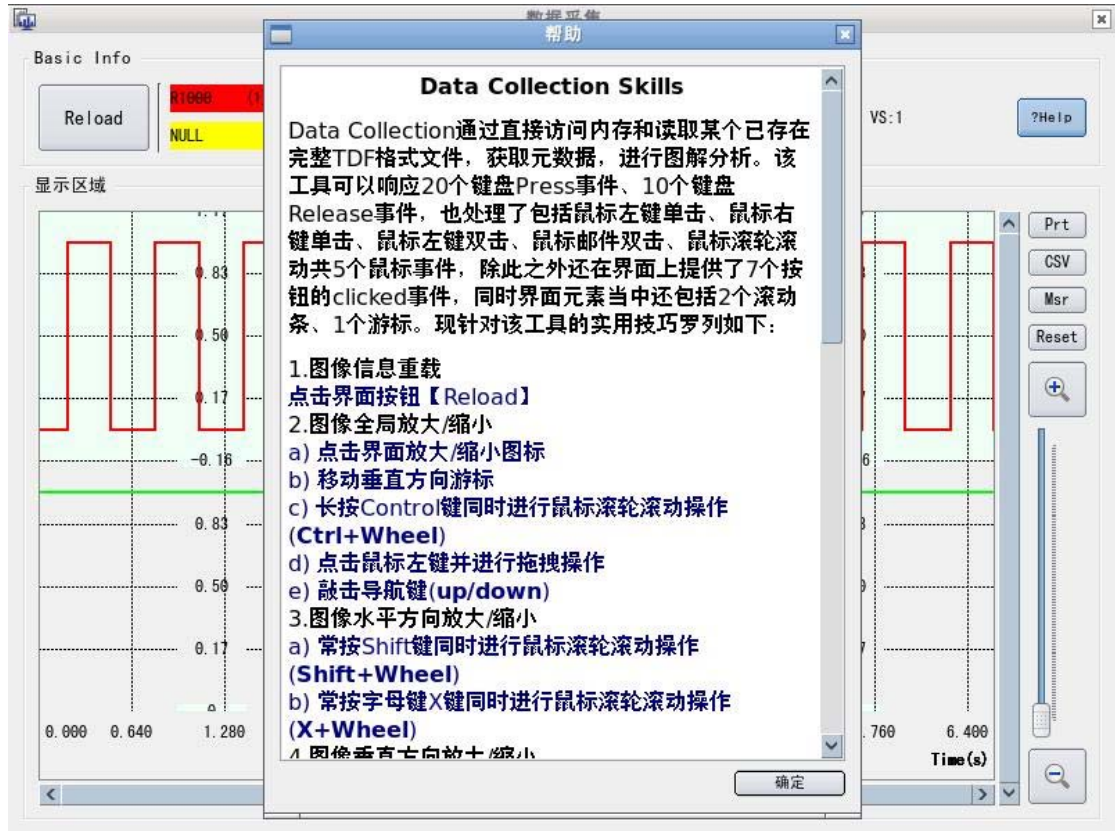


图 4-2-9-7 数据采集-帮助对话框

3. 信息采集界面下面区域为曲线描画操作区, 点击 **Prt** 可保存当前界面对应的图片文件; 点击 **CVS** 可保存当前曲线对应的数据表格; 点击 **Msr** 进入测量模式; 点击 **Reset** 按钮将曲线回到最初始的状态; 还可以点击放大缩小; 移动鼠标滚轮进行放大缩小还有其他的快捷键操作, 给出的帮助文档都有提示。

4.2.10 停止 PLC

操作步骤


- ①. 当需要停止当前正在运行的 PLC 时, 可以点击快捷工具栏的[停止]  或 **Ctrl+T** 快捷键, 则会弹出一个对话框, 显示当前正在运行的 PLC。该对话框中的表格显示所有运行的 PLC 文件, 第一列用于选择是否停止, 第二列显示 PLC 文件加载的序号, 第三列显示 PLC 文件名称, 第四列显示 PLC 文件加载后的扫描执行的类型是否为快速 PLC。



图 4-2-10-1 PLC 状态显示对话框

- ②. 选中需要停止的程序，点击 OK 按钮，则被选中的 PLC 将停止运行。



图 4-2-10-2 选中需要停止的 PLC 程序

- ③. 如需停止所有的 PLC 程序，可点击全部选择按钮，再点击 OK 按钮，所有的 PLC 程序将被停止。

4.3 其他功能介绍

概述

SoftPLC 软件在编辑调试梯形图时还可以使用如下功能：剪切、复制、粘贴、查找、撤销、恢复、生成自动加载文件、交叉参考、类型检查、检查跳转、清空无用 Lda 文件、清空磁盘、调试上电、PLC 文件上传/下载，阅读模式等操作。

※ 注意：剪切、复制操作同文档操作相同，执行粘贴操作前需要选定粘贴的位置；

撤销、恢复操作只针对梯形图描画区有效，对于变量定义区无效；

检查跳转是软件开发者使用，此处不做介绍。

4.3.1 查找功能

概述

查找功能可以分为全局查找和快速查找。

- 全局查找不仅可以查找组件、指令、字符和触点线圈在梯形图中的位置，还可以查找其在变量定义表中的位置以及其在项目配置文件(.cnf)中的位置；快速查找将查找内容直接定位到梯形图描画区。查找既可以从起始位置开始查找，也可以从光标所在位置向上或向下查找。全局查找时将查找结果按顺序逐步显示在界面下方的信息显示窗口，查找完毕后关闭查找对话框；

- 快速查找仅查找当前文件，直接将查找结果显示在梯形图描画区。

查找功能可以用三种方式选择：①在菜单[编辑]-[查找]中选择。②快捷键 Ctrl+F。③在梯形图编辑区单击右键选择。离线时选择查找直接显示的是全局查找，在线时选择查找直接显示的是快速查找。

操作步骤

- ①. 点击菜单栏[编辑]-[查找]或快捷键 Ctrl+F。

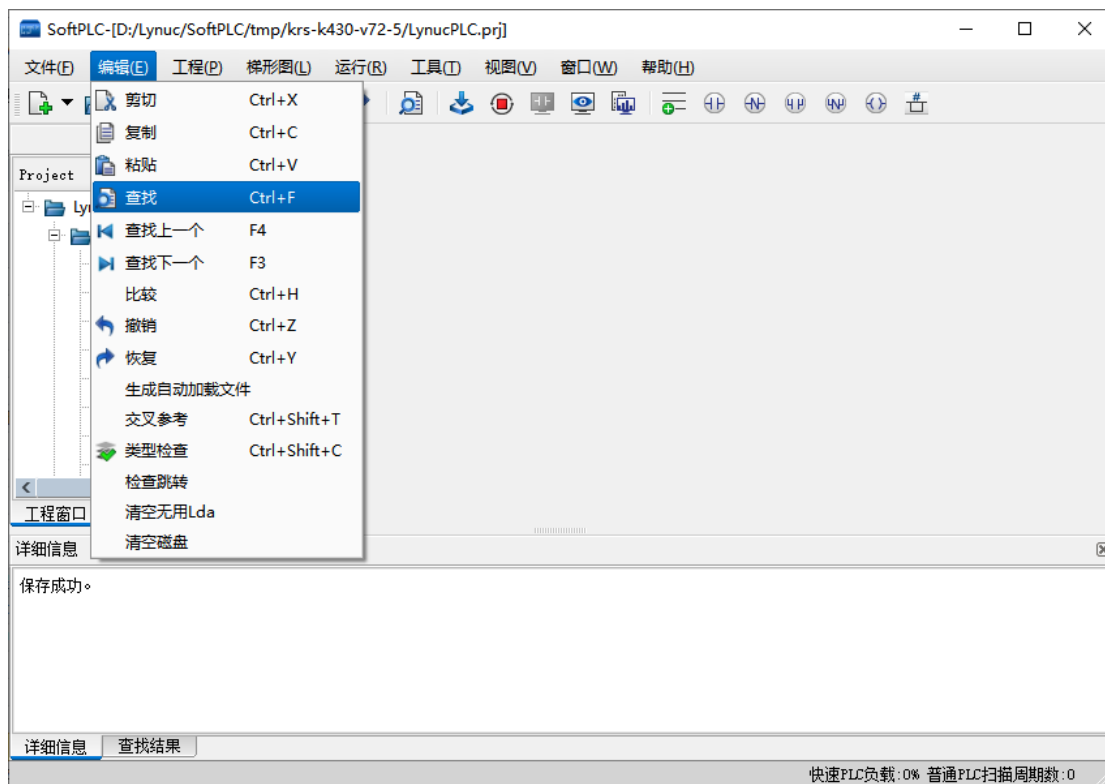


图 4-3-1-1 菜单栏开启查找功能

确认查找后，出现界面如下：



图 4-3-1-2 查找对话框

查找可分为全局查找和快速查找，全局查找如上图所示，主要包括：

表 4-3-1-1 全局查找内容

查找内容	可记录查找历史，可直接输入查找内容，也可匹配全局变量和别名，若查找历史中含有需要查找的内容，可直接下拉菜单选择变量
查找类型	全部、变量、指令、注释、交叉引用。(请正确选择查找类型，例如不可

	在注释的查找类型下去查找指令)
查找范围	整个项目、当前文件
※ 注意	其他选项还有查找全局变量、区分大小写、全字匹配和正则表达式

查找全局变量：选中后会查找全局变量，默认是不选中状态。

区分大小写：选中后会区分大小写查找，默认是不选中状态。

全字匹配：选中后只查找到全字匹配的变量，默认是不选中状态。

正则表达式：选中后按照正则表达式进行查找，默认是不选中状态。

举例：若想查找 MO20.0, MOU20.0, MOS20.0, 可输入 MO[US]?20.0 查找

若想查找宏变量，可输入 R\d+

②. 输入相关查找内容，例如查找字符串 start。在查找结果出现如下界面。

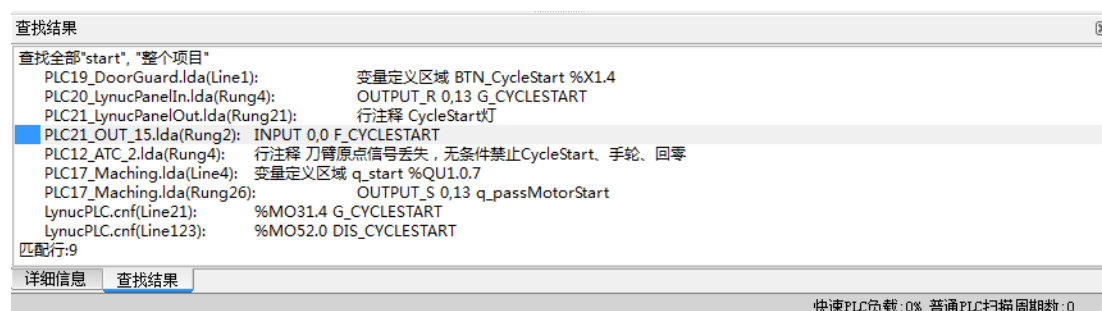


图 4-3-1-4 查找结果界面

③. 在查找结果界面中，双击所要显示的行会直接定位到内容查找结果有几种分类：

- 1) lda 文件中的变量定义区域，双击后打开对应的 lda 文件，变量定义区域显示查找结果对应的行。
- 2) lda 文件中的梯形图描画区域，双击后打开对应的 lda 文件，梯形图描画区域显示查找对应的触点、功能块、线圈、行注释。
- 3) 系统配置文件 LynucPLC.cnf 文件，双击后打开该配置文件，如下图所示：也是直接显示到查找对应的行。



图 4-3-1-5 系统配置文件窗口

- ④. 双击定位一次后就可以采用快捷键进行向上向下查找，F3 为查找下一个，F4 为查找上一个，一直点击可实现循环查找定位。
- ⑤. 在线时直接显示的是快速查找，如下图：



图 4-3-1-6 快速查找对话框

快速查找主要包括：

表 4-3-1-2 快速查找内容

查找内容	可记录查找历史，可直接输入查找内容，若查找历史中含有需要查找的内容，可直接下拉菜单选择变量
------	---

查找下一个	点击后直接定位到当前 lda 文件的梯形图描画区第一个具有查找字符的组件，一直点击会一直往下面定位，查找完一遍后再从文件开始位置查找
查找上一个	点击后直接定位到当前 lda 文件的梯形图描画区最后一个具有查找字符的组件，一直点击会一直往上面定位，查找完一遍后再从文件结束位置查找

4.3.2 生成自动加载文件

概述

自动加载文件可以决定在系统开机启动时需要扫描执行的 PLC 文件，自动加载文件在普通 PLC 工程中是 CNCPLC.pcg 文件和 CNCPLC_qck.pcg 文件。其中前一个文件决定了哪些 PLC 文件在开机启动时加载执行，后一个文件决定 PLC 文件加载执行的类型，是快速 PLC 还是普通 PLC。

操作步骤

- ①. 点击菜单[编辑]-[生成自动加载文件]或者双击工程树中的 CNCPLC.pcg 文件。

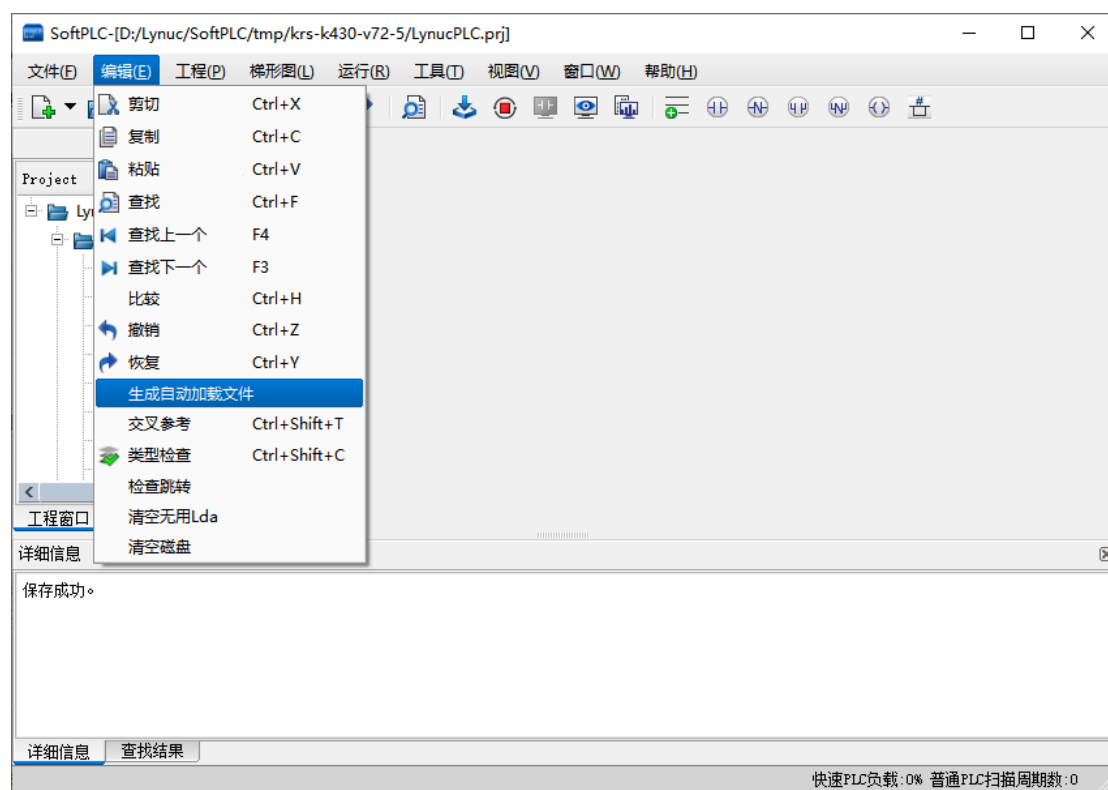


图 4-3-2-1 生成自动加载文件-1

②. 点击确认后出现如下界面，包含有 5 列选项，分别说明如下：

第一列“Include”，决定所在行的 PLC 文件是否自动加载。选中即为开机自动加载，不选中则为开机不自动加载；

第二列“File Name”，显示的 PLC 文件名称；

第三列“PLC ID”，为 PLC 文件的加载的程序号，为必填项，若不填系统开机会报 PLC 加载失败；

第四列“Auto Start”，为 PLC 文件加载完成后是否立即执行。选中即为加载后立即执行，不选中加载后不会立即执行。当程序加载完没有执行时，PLC 文件所编写的逻辑并不生效，需要手动运行起来，方法是选择菜单运行->运行，选中需要运行的 PLC 文件后点击 OK 按钮即可；

第五列“Quick PLC”，为决定 PLC 文件的执行类型。选中为快速 PLC，每个 PLC 周期都会扫描执行；不选中为普通 PLC，不能保证每个 PLC 周期都会扫描执行。详见 4.2.6 节中详细说明。



图 4-3-2-2 自动加载文件对话框

③. 选择需要自动加载的 PLC 文件，在对应的行的 Include 选中，填写 PLC ID，选择是否立即执行和是否为快速 PLC，最后点击 Create Pcg 按钮即可，下次开机时所选中的 PLC 文件便可实现自动加载。

4.3.3 交叉参考

概述

交叉参考是指可查看变量地址(Y、MO、S、R、Q)所使用的位置包括（触点、线圈、功能块）及次数。

操作步骤

- ①. 点击菜单栏[编辑]-[交叉参考]。

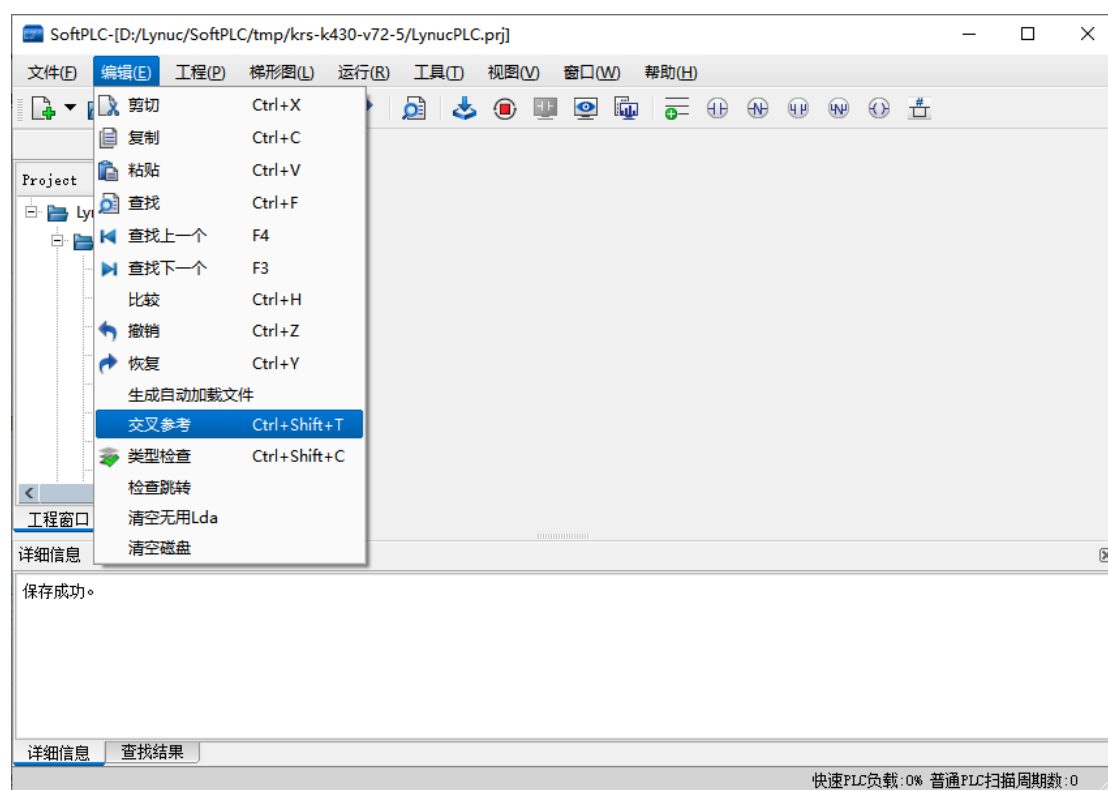


图 4-3-3-1 开启交叉参考

- ②. 确认后出现界面如下：备注明确表明了变量的使用次数及使用位置。

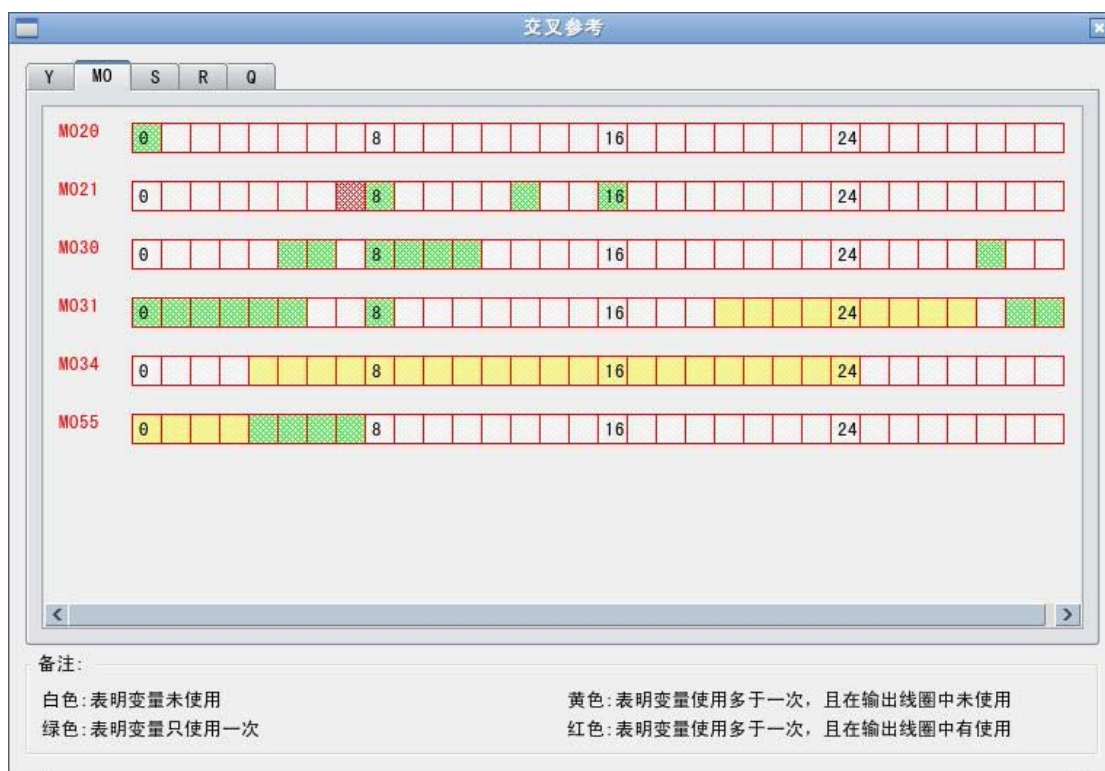


图 4-3-3-2 交叉参考对话框

4.3.4 清空无用 Lda 文件

概述

在 PLC 工程打开后,若不查看除自动加载文件指定的 PLC 文件的其他文件,可选择清空无用 lda 文件,选择后 PLC 工程中只显示自动加载的 PLC 文件,工程文件也会更新为只包含自动加载的 PLC 文件。

操作步骤

- ①. 点击菜单栏[编辑]-[清空无用 Lda]。

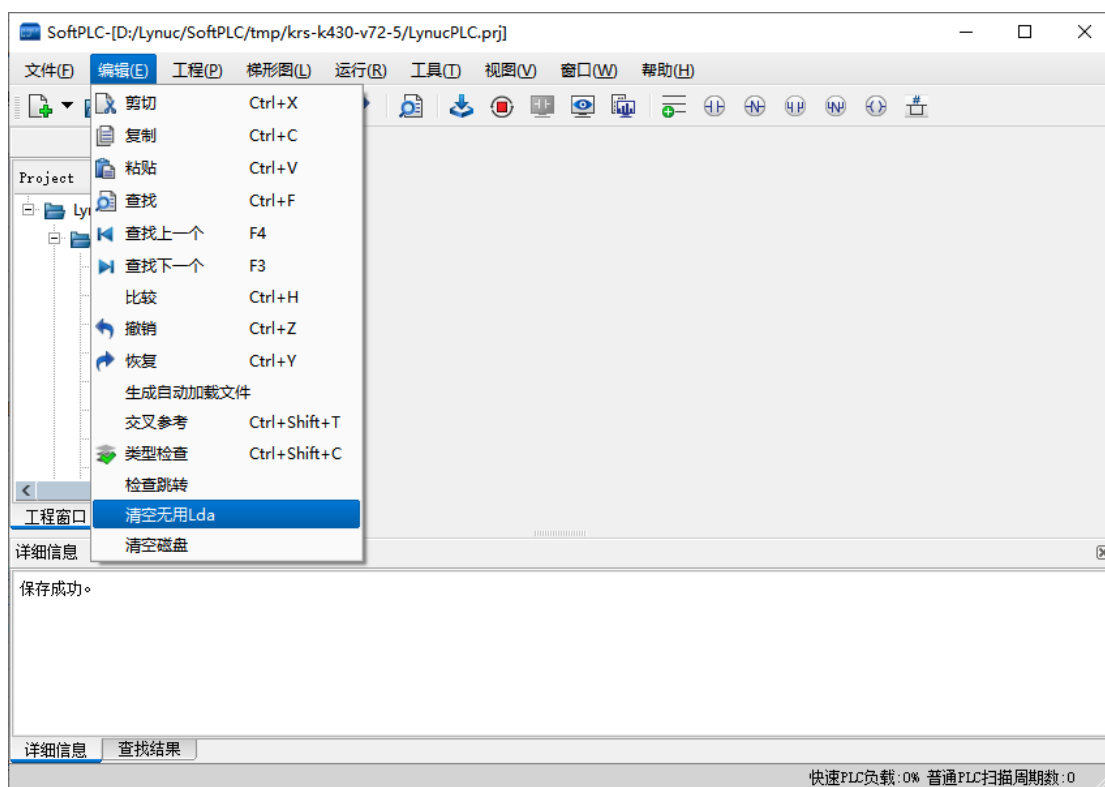


图 4-3-4-1 清空无用 Lda 文件

②. 点击后 PLC 工程文件被清空，工程文件也会更新，界面如下：

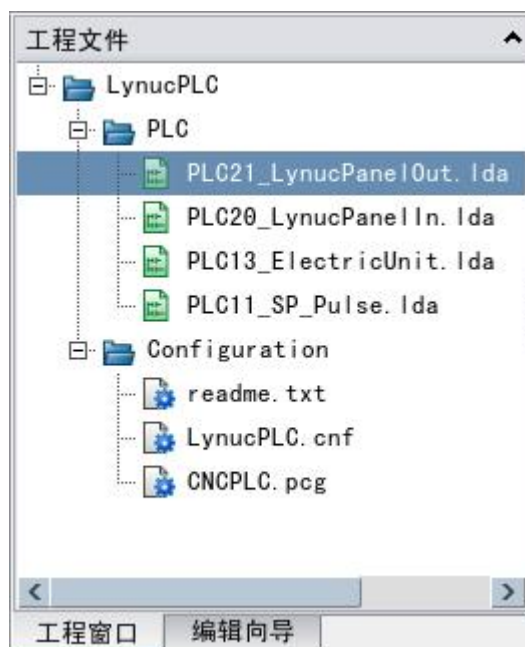


图 4-3-4-2 完成清空无用 Lda 文件

4.3.5 清空磁盘

概述

PLC 在线使用的过程中，清空磁盘功能可将 PLC 工程目录下没有存在于工程项目中的 PLC 文件从磁盘内清除。可在“清空无用 Lda”操作后进行。

操作步骤

- ①. 点击菜单栏[编辑]-[清空磁盘]即可。

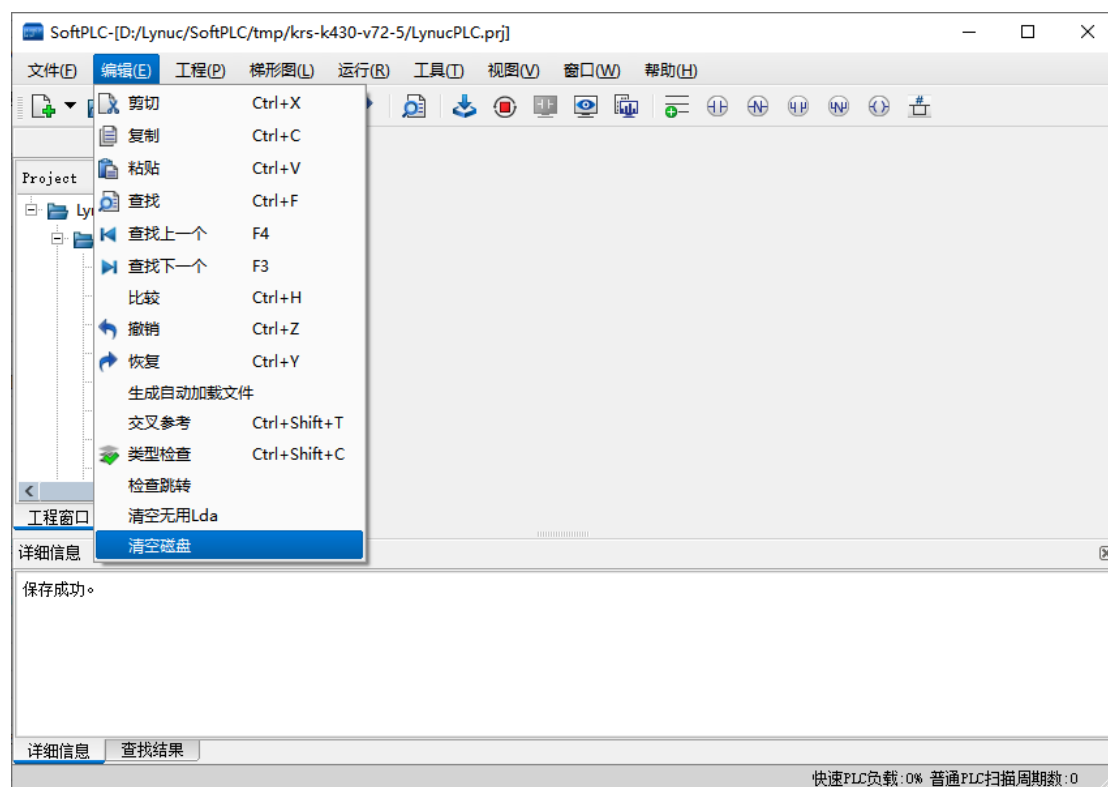


图 4-3-5-1 清空磁盘

- ②. 点击后弹出确认窗口，点击 OK 将执行清空磁盘操作，点击 Cancel 将不执行清空磁盘操作。



图 4-3-5-2 清空磁盘提示框

4.3.6 调试上电

概述

使用调试上电功能可对电流控制，速度控制，位置控制等进行优化，合理提高伺服的增益，并保证合理的加减速，实现伺服的高速度、高精度。

操作步骤

- ①. 点击菜单栏[工具]-[调试上电]。

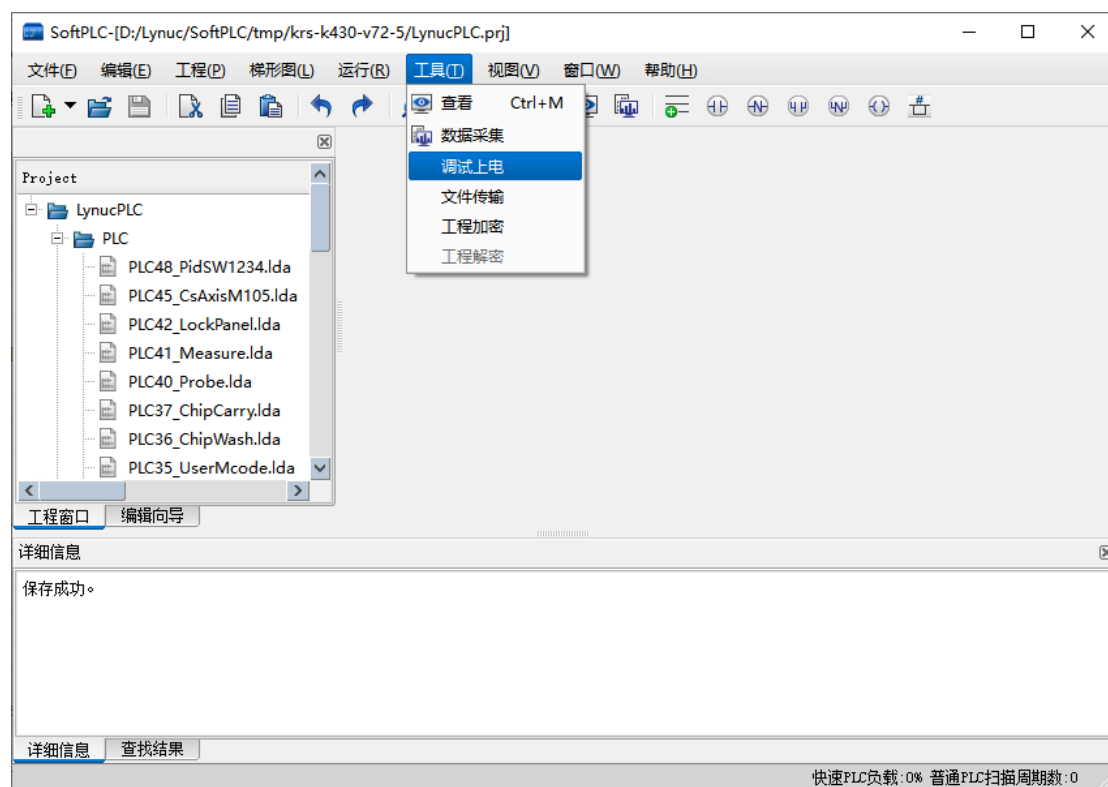


图 4-3-6-1 调试上电

- ②. 系统弹出调试上电确认窗口，由于调试上电非常危险，避免引起误操作，需要输入密码：123456 后点击“上电”按钮，系统便可强制上强电进行调试。



图 4-3-6-2 调试上电确认窗口

调试上电完成后，可重复①②步骤，在调试上电窗口中点击“断电”完成调试断电操作。

4.3.7 PLC 传输

概述

使用此功能可对控制器进行 PLC 文件上传或下载。

操作步骤

- ①. 点击菜单栏[工具]-[PLC 传输]

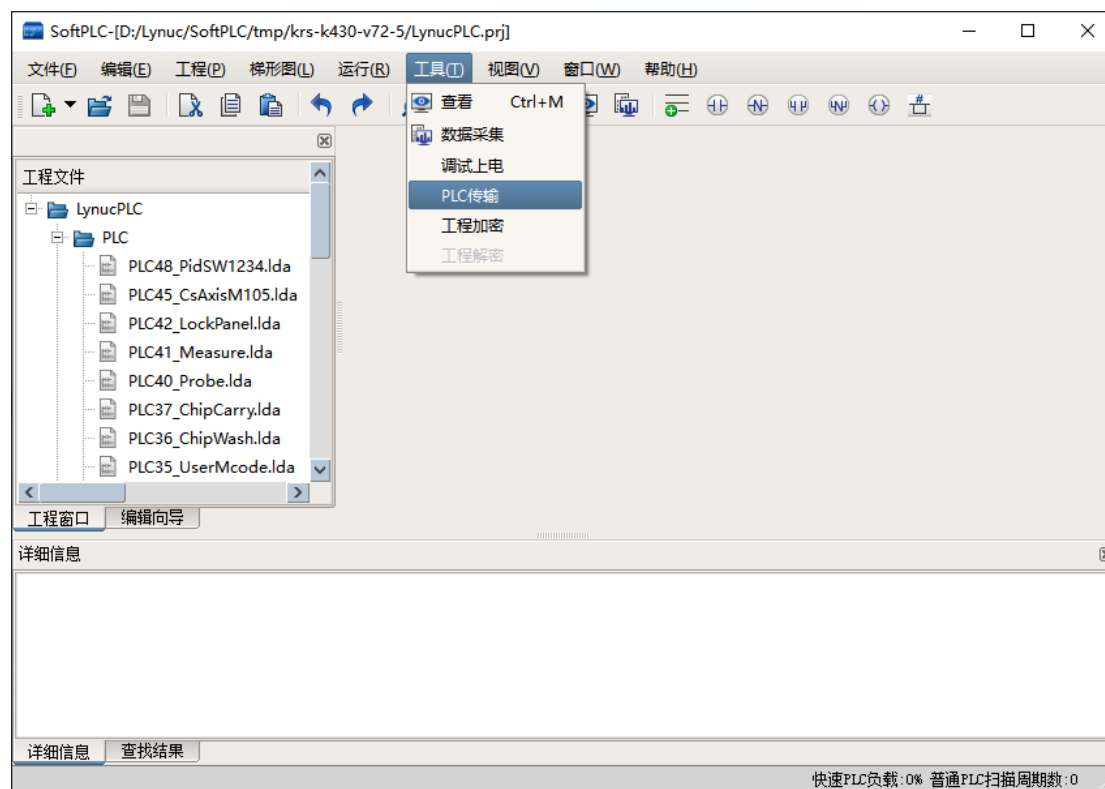


图 4-3-7-1 开启 PLC 传输

②. 弹出“PLC 传输”对话框：

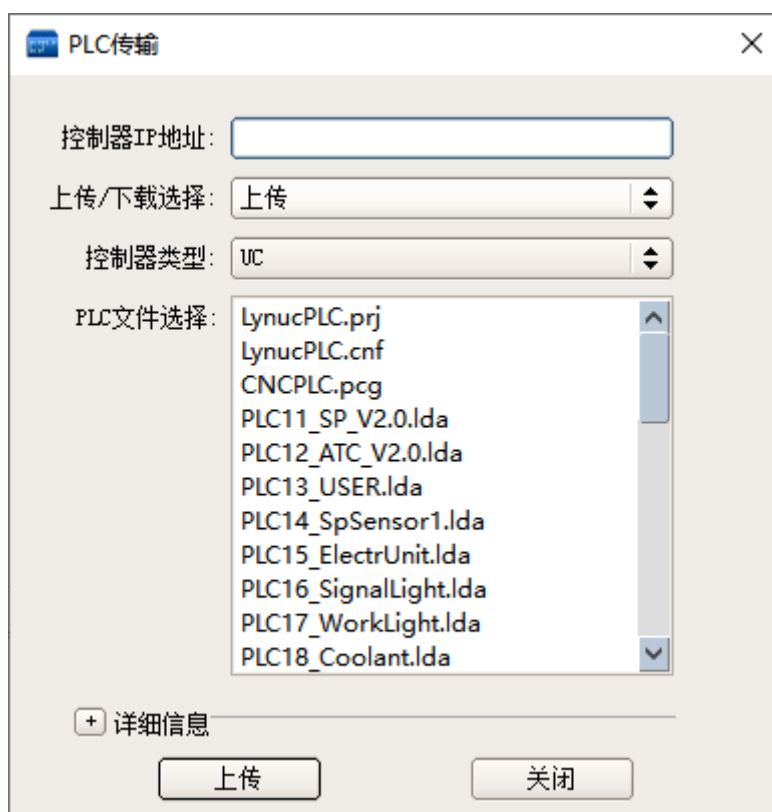


图 4-3-7-2 PLC 文件上传/下载对话框

- “控制器 IP 地址”：输入需要进行 PLC 文件更新的控制器的 IP 地址；
- “上传/下载选择”：可选择“上传”或“下载”；
- “控制器类型”：选择对应的 LYNUC 控制器类型，分别为 UC、N3、N5；
- “PLC 文件选择”：可同时选择多个需要上传的 PLC 文件。

a. 上传

输入控制器 IP 地址后，在“上传/下载选择”下拉框中选择“上传”，然后选择对应的控制器类型并选中需要上传的文件后，点击“上传”按钮：

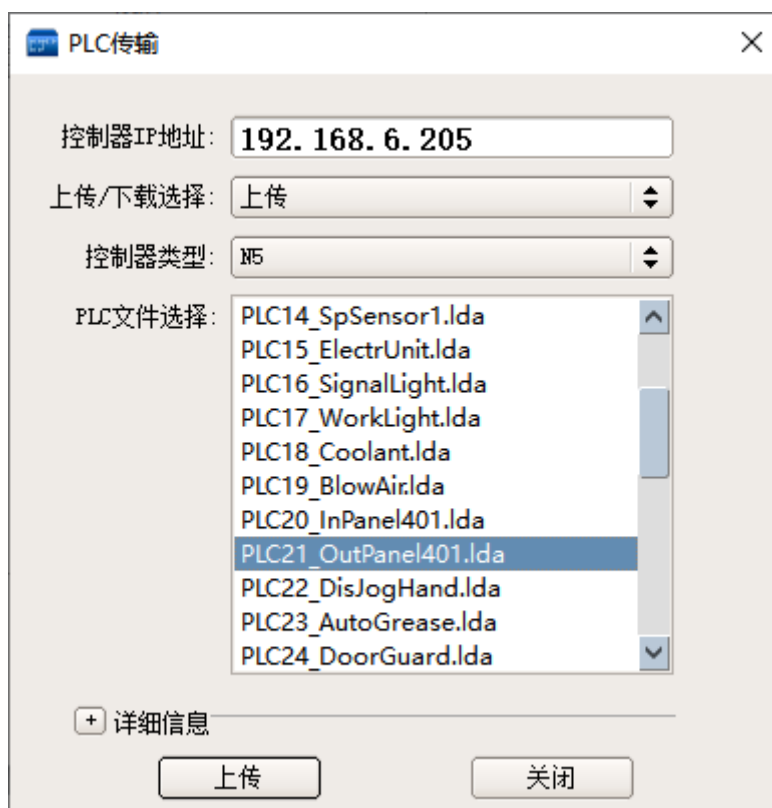


图 4-3-7-3 PLC 文件上传对话框

上传完成后，弹出 PLC 工程文件上传成功提示对话框，点击“OK”按钮后完成。

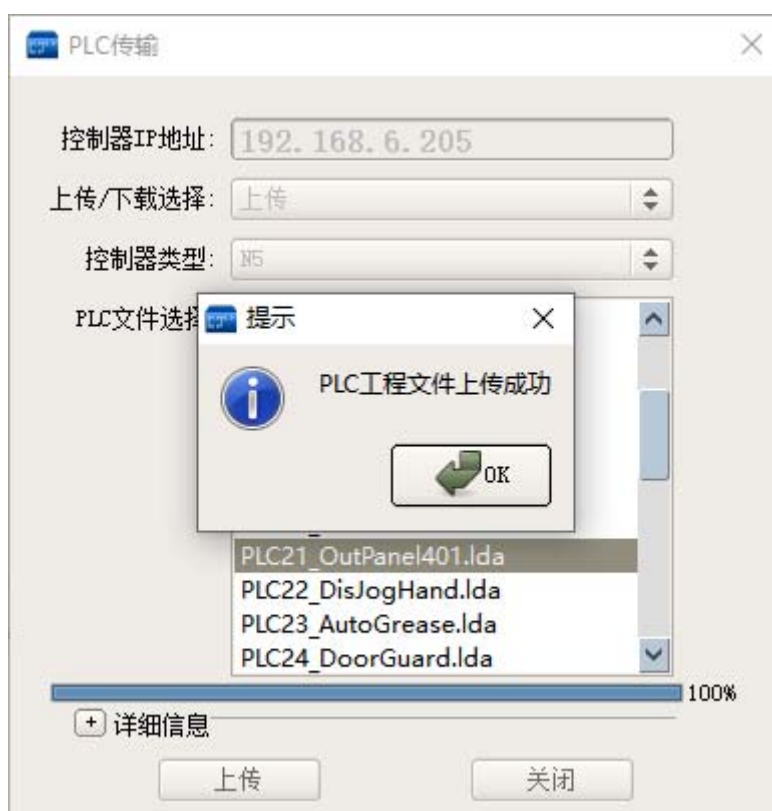


图 4-3-7-4 PLC 文件上传成功对话框

b. 下载

输入控制器 IP 地址后，在“上传/下载选择”下拉框中选择“下载”，然后选择对应的控制器类型，点击“刷新”后选中需要下载的文件后，点击“下载”按钮：

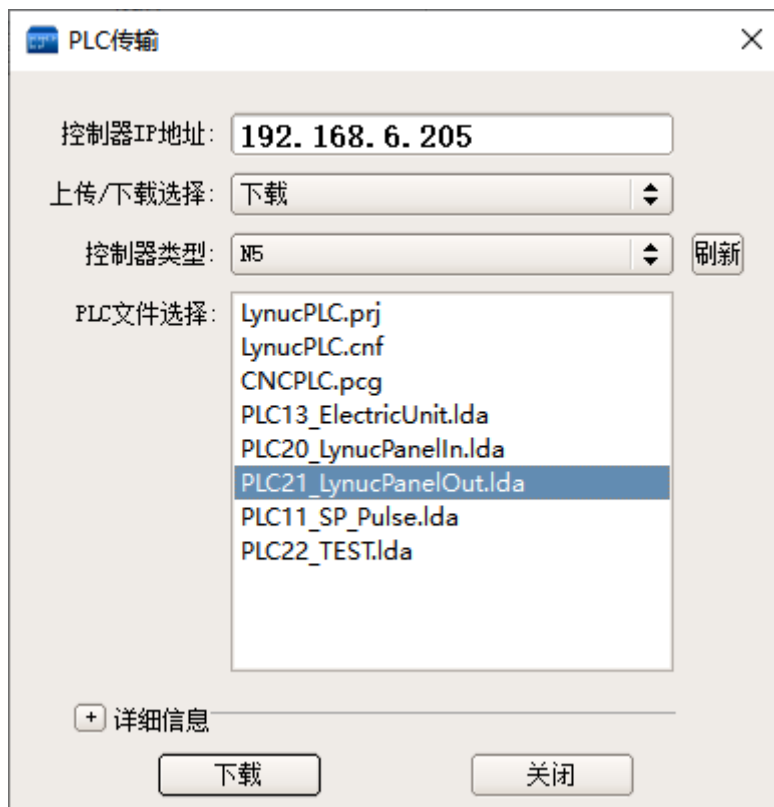


图 4-3-7-5 PLC 文件下载对话框

下载完成后，弹出 PLC 工程文件下载成功提示对话框，点击“OK”按钮后完成。

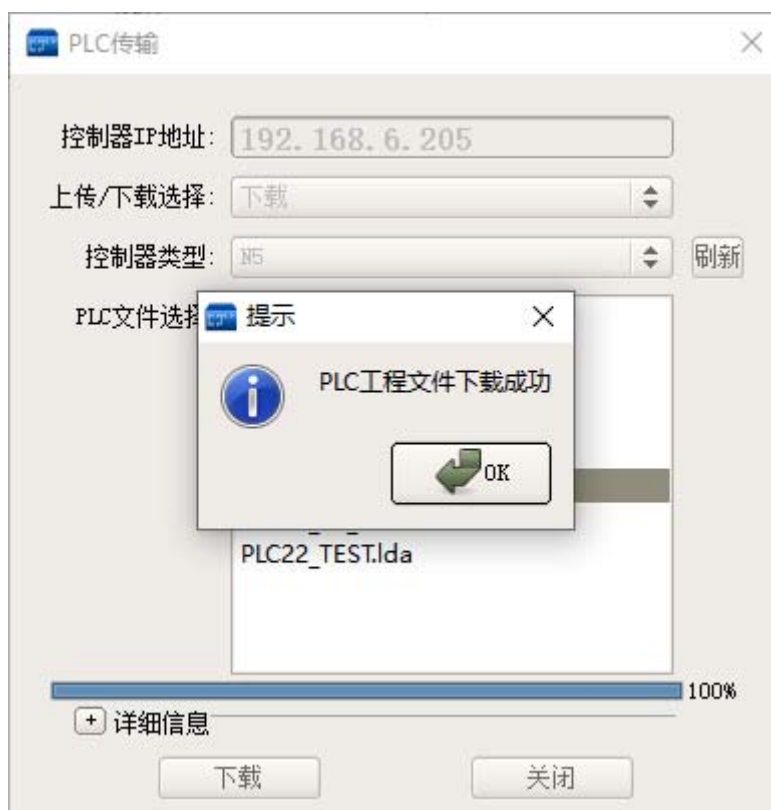


图 4-3-7-6 PLC 文件下载成功对话框

4.3.8 工程加密

概述

PLC 使用的过程中，为了防止客户随意修改 PLC，更好的维护和修改配置，我们可以选择把整个 PLC 工程进行加密。

操作步骤

- ①. 点击菜单栏[工具]-[工程加密]。

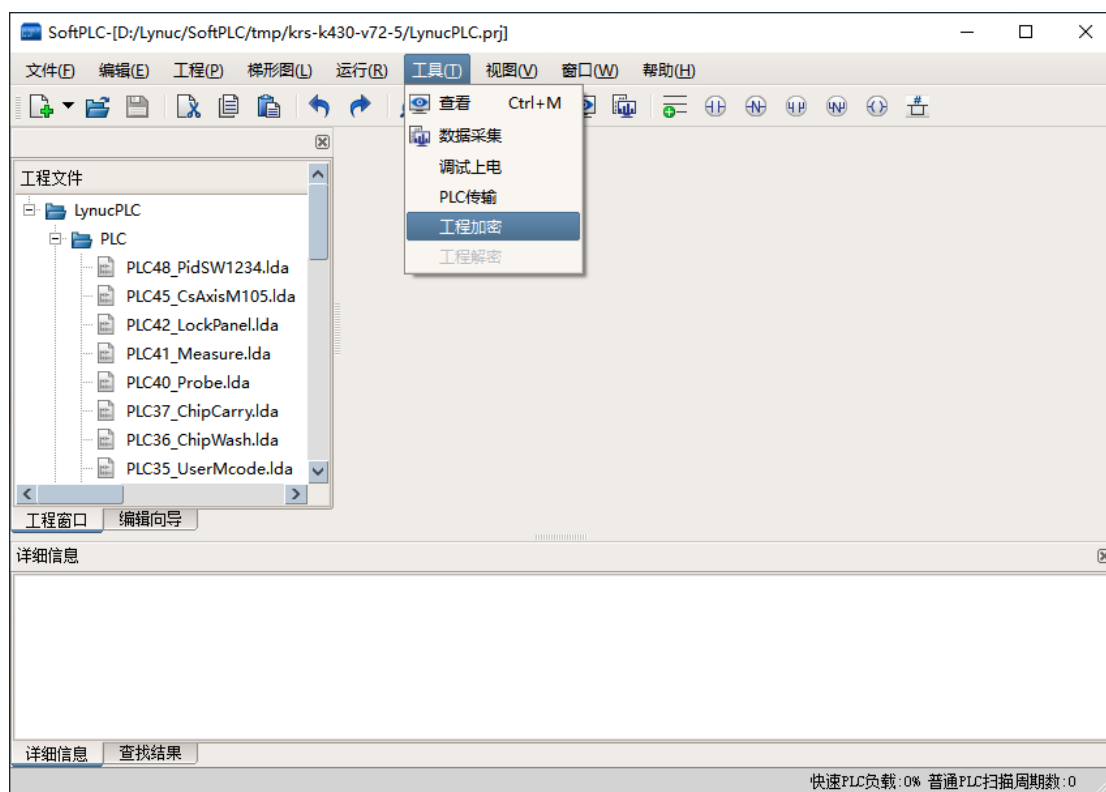


图 4-3-8-1 工程加密

②. 点击后弹出输入窗口，输入密码并确认密码后点击确定。

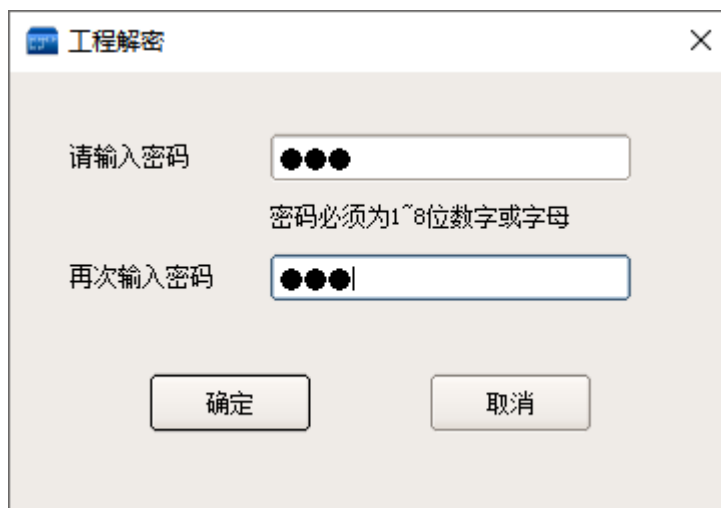


图 4-3-8-2 工程加密输入框

③. 如果两次密码输入一致，则加密成功并弹出提示框。

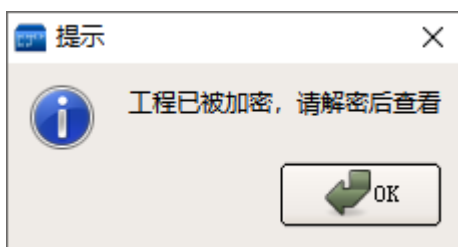


图 4-3-8-3 加密成功提示框

4.3.9 工程解密

概述

PLC 使用的过程中, 如果打开一个加密的 PLC 工程, 需要进行解密操作才能查看其详细信息。

操作步骤

①. 点击菜单栏[工具]-[工程解密]。

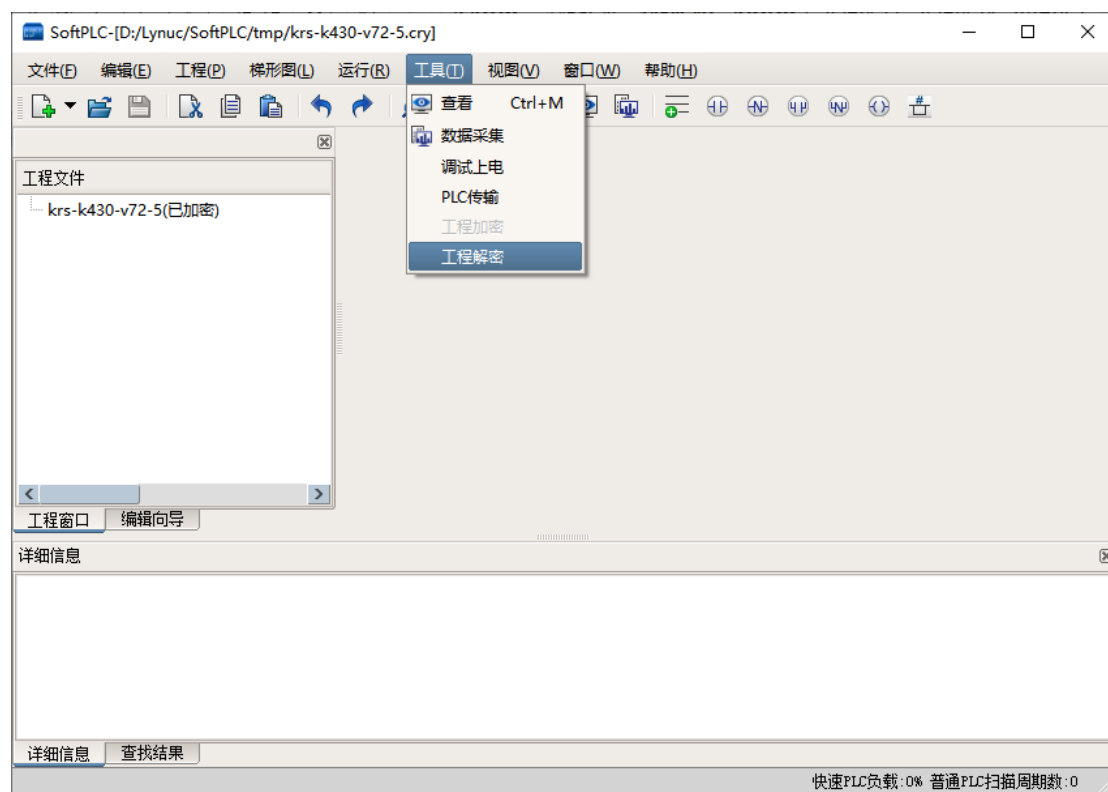


图 4-3-8-1 工程解密

②. 点击后弹出窗口, 输入对应正确密码后点击确定。



图 4-3-8-2 工程解密输入框

③. 如果密码输入正确，则解密成功并弹出提示框。

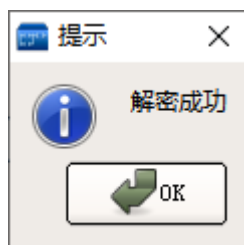


图 4-3-8-2 工程解密提示框

4.3.10 阅读模式

概述

使用阅读模式可以隐藏 PLC 界面左侧的工程窗口/编辑向导，以及下方的详细信息区域，在打开 LDA 文件的状态下最大化显示梯形图，方便查看，使 PLC 梯形图区域显示全面。

操作步骤

①. 点击菜单栏[视图]-[阅读模式]，或按下快捷键 F9。

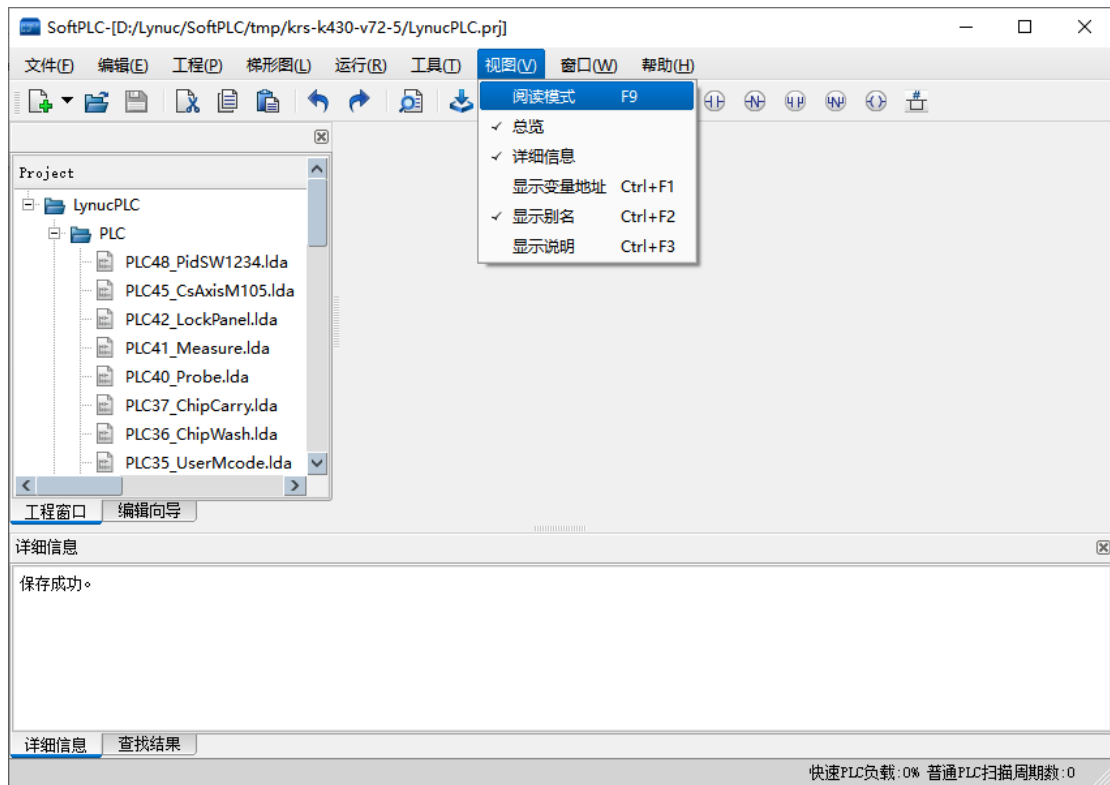


图 4-3-8-1 开启阅读模式

②. 阅读模式视图如下图显示:

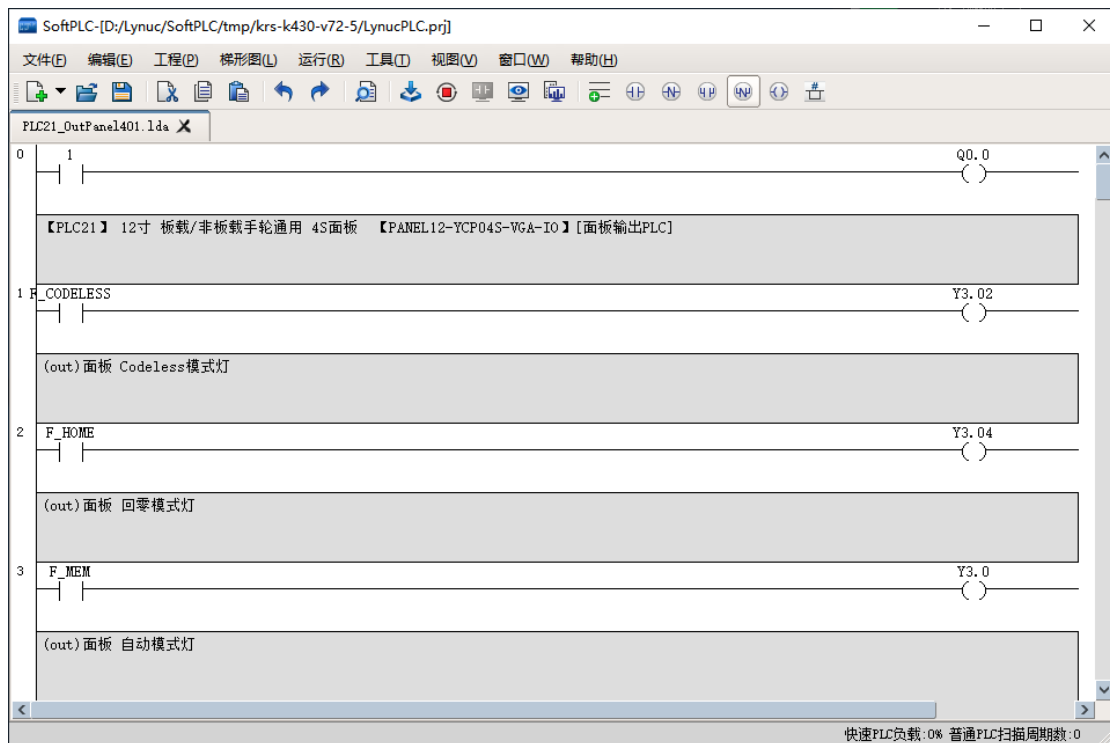


图 4-3-8-2 阅读模式界面

5 基本逻辑控制指令

概述

基本指令系统、定时器和计数器的知识应用是学习 PLC 程序控制和其他功能指令的基础。这里对基本指令及定时器、计数器作一般性介绍。更进一步的详解请参考其他书籍。

5.1 语法参考

概述

这里的语法参考主要是指编程语言，现介绍如下语言：LD 语言（梯形图语言）、IL 语言（指令表语言）。此处强调一下，Q0.0 属于无效变量，不可作为输入触点变量。

5.1.1 LD 语言

概述

LD 语言（梯形图语言）是 IEC61131-3 标准中规定的 5 种编程语言之一，是 PLC 编程中最为广泛使用的一种图形化语言。

它源于机电一体化领域中图形表示的继电器逻辑，用它编写的程序与实际电气操作原理图像对应，非常直观，易于学习和掌握。

5.1.2 IL 语言

概述

IL 语言是一种低级语言，与汇编语言非常相似，是在借鉴世界著名 PLC 厂商的指令表语言的基础上形成的一种标准语言。

在其他语言进行编译时，IL 是作为其中的公用中间语言。因此通常情况下，与其他语言相比，使用 IL 编写的程序效率更高。

※ 注意

IL 中提供了一个“当前结果”（CR，Current Result），可理解为电路中的电流，在 CR 中存储了用户程序的当前执行结果。用户程序中的每一行语句执行之后，CR 值都会被“刷新”。CR 值可以是编程软件支持的任意数据类型，则取决于刚刚执行完的语句。依据后续语句的不同，

CR 值可能作为下一条语句的执行条件,也可能作为下一条语句的操作数之一。操作数不同,执行之后对 CR 值的影响也不同。下表根据对 CR 值不同的影响将操作符进行了初步的分组。更详细的描述参见下一节。

表 5-1-2-1 操作符对 CR 值的影响

操作符	分组	对 CR 值的影响
LD, LDN	C	CR 重新建立
逻辑比较, 比较指令等	P	CR 值被更新为操作结果
ST, R, S, JMP 等	U	CR 值保持不变

5.1.3 语句格式

概述

IL 程序主要面向程序行,每行语句只能对应一条指令。IL 程序允许有空白存在,其每个语句基本格式如下:

<p>标号:</p> <p>操作符/功能/功能块 操作数 (表) (*注释)</p>
--

表 5-1-3-1 语句格式说明

格式	说明
[标号]	可选。使用标号的目的在于实现程序的跳转。
[操作符/功能/功能块]	参见下一节对各操作符、功能、功能块的详细说明。
[操作数/表]	表示操作符/功能/功能块对应的操作数。
[注释]	可选。由 (**) 进行界定,每行只允许一个注释,注释不允许嵌套。

5.2 位逻辑指令


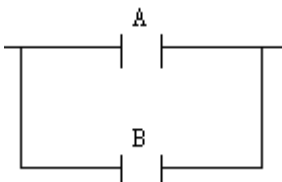
概述

基本逻辑指令包括位逻辑指令和逻辑运算指令。这些指令是 PLC 的基本逻辑指令,加上定时器和计数器的综合应用,基本上可以实现继电器控制系统的程序编制。在程序中,这部分指令用触点、线圈及连线可以很方便的在梯形图中表示。

5.2.1 常开触点/并联常开触点

格式

表 5-2-1-1 常开触点/并联常开触点指令格式

语法	名称	指令格式	影响 CR 值
LD	常开触点		
	并联常开触点		
IL	装载	LD bit	C
	与	AND bit	P
	或	OR bit	
操作数	输入/输出	数据类型	允许的变量类型
Bit	输入 A	Bit	I, MI, MO, Q, 常量

功能说明

当在指定地址中的位的值为“1”时。常开触点闭合。当触点闭合时。信号流将通过该点。

举例说明



图 5-2-1-1 常开触点/并联常开触点指令梯形图


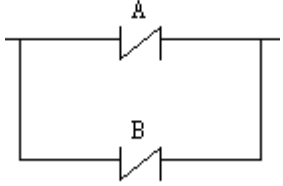
描述：

当变量 Q1.0 值为“1”或者变量 Q0.3 值为“1”，并且在变量 Q0.2 值为“1”时，则电流导通，输出变量 Q1.0 值为 1，否则输出变量 Q1.0 值为 0。

5.2.2 常闭触点/并联常闭触点

格式

表 5-2-2-1 常闭触点/并联常闭触点指令格式

语法	名称	指令格式	影响 CR 值
LD	常闭触点		
	并联常闭触点		
IL	取反装载	LDN bit	C
	与非	ANDN bit	P
	或非	ORN bit	
操作数	输入/输出	数据类型	允许的变量类型
Bit	输入 A	Bit	I, MI, MO, Q, 常量

功能说明

当保存在地址中的位的值为“0”时，常闭触点开启。当触点开启时，信号流将通过该接点。

举例说明



图 5-2-2-1 常闭触点/并联常闭触点指令梯形图






描述：

当变量 Q0.1 值为“0”或者变量 Q0.3 值为“0”，并且在变量 Q0.2 值为“0”时，则电流导通，输出变量 Q1.0 值为 1，否则输出变量 Q1.0 值为 0。

5.2.3 普通线圈/复位线圈/置位线圈/反向线圈/报错线圈

格式

表 5-2-3-1 PLC 线圈指令格式

语法	名称	指令格式	影响 CR 值
LD	普通线圈		
	复位线圈		
	置位线圈		
	反向线圈		
	报错线圈		
IL	赋值	ST bit	U
	复位	R bit	
	置位	S bit	
	反向	STN bit	
	报错	ERROR N 注：N 为常数，范围 1~160	
操作数	输入/输出	数据类型	允许的变量类型
Bit	输出 A	Bit	O, MO, Q

功能说明

PLC 线圈是指 PLC 的输出点，在 PLC 中称为线圈，叫做输出继电器，当然，也有内部使用的辅助继电器线圈。具体作用见下表所示：

表 5-2-3-2 PLC 线圈的具体作用

语法	指令	作用
LD	普通线圈	将线圈左侧的电流值赋值给 bit。
	复位线圈	若线圈左侧电流值为 1，则 bit 值被置为 0，否则 bit 值不变。
	置位线圈	若线圈左侧电流值为 1，则 bit 值被置为 1，否则 bit 值不变。
	反向线圈	将线圈左侧的电流值取反赋值给 bit。
	报错线圈	若线圈左侧电流值为 1，会报相应编号的 PLC 错误。

举例说明



图 5-2-3-1 线圈指令梯形图

描述:

当变量 X3.03 值为 1 时，电流导通，输出变量 Y3.03 值为 1，否则输出变量 Y3.03 值为 0。

5.3 逻辑运算指令

概述

逻辑运算指令主要针对变量进行按位操作，主要包括逻辑取反、逻辑与、逻辑与非、逻辑或、逻辑异或操作，这些指令的参数要求数据类型必须一致。

5.3.1 逻辑取反 NOT

格式

表 5-3-1-1 逻辑取反 NOT 指令格式

语法	名称	指令格式	影响 CR 值
LD	按位取反		
IL	按位取反	NOT A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	unsigned, signed, bit	所有类型
OUT	输出 B	unsigned, signed, bit	O, MO, Q

IN、OUT 的数据类型必须一致。

功能说明

逻辑取反：将 IN 端指定的数据相对应的位置按位取反，即 1 变 0，0 变 1，然后将结果存入 OUT 指定的单元中。

例如：

$$\begin{array}{r} \text{NOT} \quad 0010 \\ \hline 1101 \end{array}$$

举例说明



图 5-3-1-1 逻辑取反 NOT 指令梯形图

描述：

当变量 Q0.1 的值为 1 时，将地址 Q0.2 的值进行逻辑取反操作后输出到 Q0.3 中。

5.3.2 逻辑与 AND

格式

表 5-3-2-1 逻辑与 AND 指令格式

语法	名称	指令格式	影响 CR 值
LD	按位与		
IL	按位与	AND A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1, IN2	输入 A B	unsigned, signed, bit	所有类型
OUT	输出 C	unsigned, signed, bit	O, MO, Q

IN1, IN2, OUT 的数据类型必须一致。

功能说明

逻辑与：将 IN1 和 IN2 端指定的数据相对应的位置按位与运算，如果对应的两位都为 1，则该位的结果为 1，否则为 0，然后将结果存入 OUT 指定的单元中。

例如：

	0010
AND	1110
	0010

举例说明

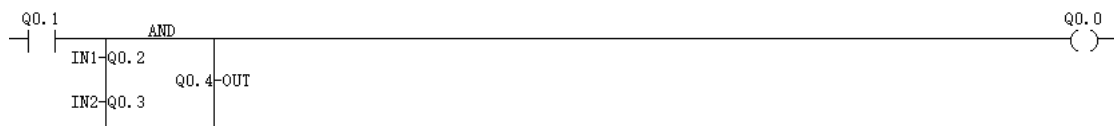


图 5-3-2-1 逻辑与 AND 指令梯形图

描述:

当 Q0.1 为 1 时，将地址 Q0.2 的值和 Q0.3 的值进行逻辑与操作后输出到 Q0.4 中。

5.3.3 逻辑与非 ANDN

格式

参照上节逻辑与 AND 指令。

功能说明

逻辑与非：将 IN1 和 IN2 端指定的数据相对应的位置先进行逻辑与运算，然后再将所得结果进行取反运算，将结果存入 OUT 指定的单元中。

例如:

	0010
ANDN	1110
	1101

举例说明



图 5-3-3-1 逻辑与非 ANDN 指令梯形图

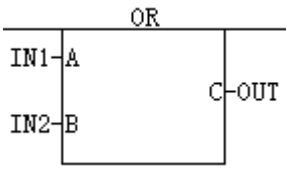
描述:

当 Q0.1 为 1 时, 将地址 Q0.2 的值和 Q0.3 的值进行逻辑与非操作后输出到 Q0.4 中。

5.3.4 逻辑或 OR

格式

表 5-3-4-1 逻辑或 OR 指令格式

语法	名称	指令格式	影响 CR 值
LD	按位或		
IL	按位或	OR A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1, IN2	输入 A B	unsigned, signed, bit	所有类型
OUT	输出 C	unsigned, signed, bit	O, MO, Q

IN1, IN2, OUT 的数据类型必须一致。

功能说明

逻辑或: 将 IN1 和 IN2 端指定的数据相对应的位置进行逻辑或运算, 如果相对应的两位都为 0, 则该位的结果值为 0, 否则为 1, 将结果存入 OUT 指定的单元中。

例如:

$$\begin{array}{r}
 0010 \\
 \text{OR } 1110 \\
 \hline
 1110
 \end{array}$$

举例说明



图 5-3-4-1 逻辑或 OR 指令梯形图

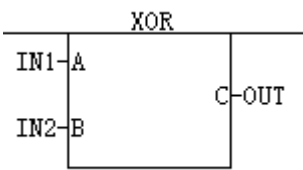
描述:

当 Q0.1 为 1 时, 将地址 Q0.2 的值和 Q0.3 的值进行逻辑与非操作后输出到 Q0.4 中。

5.3.5 逻辑异或 XOR

格式

表 5-3-5-1 逻辑异或 XOR 指令格式

语法	名称	指令格式	影响 CR 值
LD	按位与		
IL	按位与	XOR A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1, IN2	输入 A B	unsigned, signed, bit	所有类型
OUT	输出 C	unsigned, signed, bit	O, MO, Q

IN1, IN2, OUT 的数据类型必须一致。

功能说明

逻辑异或: 将 IN1 和 IN2 端指定的数据相对应的位置进行逻辑异或运算, 如果相对应的两位不同则为 1, 否则为 0, 将结果存入 OUT 指定的单元中。

例如:

$$\begin{array}{r}
 0010 \\
 \text{XOR } 1110 \\
 \hline
 1100
 \end{array}$$

举例说明



图 5-3-5-1 逻辑异或 XOR 指令梯形图

描述:

当 Q0.1 为 1 时, 将地址 Q0.2 的值和 Q0.3 的值进行逻辑异或操作后输出到 Q0.4 中。

5.4 操作指令

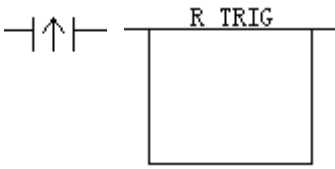
概述

操作指令主要是针对电流的边沿跳变执行相关操作, 类似于电路中的边沿触发方式, 另外还有高电平触发, 这类操作指令包括上升沿、下降沿、跳变沿、自定义占空比方波发生器, 这类指令只适用于 PPLC。

5.4.1 上升沿

格式

表 5-4-1-1 上升沿指令格式

语法	名称	指令格式	影响 CR 值
LD	上升沿		
IL	上升沿	R_TRIG	P

现在的上升沿有两种样式, 新版本两种样式都支持, 老版本不支持新的样式(带有箭头的)。

功能说明

指令改变 CR 值, 当指令前面的 CR 值由 0 变为 1 的时刻, 通过该指令后的 CR 值为 1, 否则通过该指令后的 CR 值为 0。

举例说明



图 5-4-1-1 上升沿指令梯形图

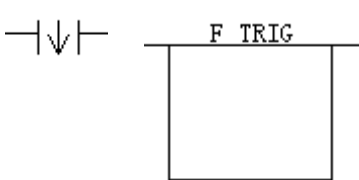
描述:

当变量 X3.03 由 0 变为 1 时, 变量 Y3.03 的值为 1, 若变量 X3.03 一直为 1 或 0, 变量 Y3.03 的值为 0。即 Y3.03 只会在 X3.03 上升沿时是 1, 并保持 1 个 PLC 周期, 其余时间都是为 0。

5.4.2 下降沿

格式

表 5-4-2-1 下降沿指令格式

语法	名称	指令格式	影响 CR 值
LD	下降沿		
IL	下降沿	F_TRIG	P

现在的下降沿有两种样式, 新版本两种样式都支持, 老版本不支持新的样式(带有箭头的)。

功能说明

指令改变 CR 值, 当指令前面的 CR 值由 1 变为 0 的时刻, 通过该指令后的 CR 值为 1, 否则通过该指令后的 CR 值为 0。

举例说明



图 5-4-2-1 下降沿指令梯形图

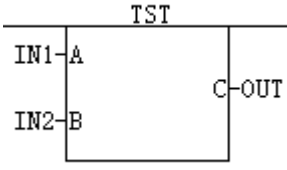
描述:

当变量 X3.03 由 1 变为 0 时, 变量 Y3.03 的值为 1, 若变量 X3.03 一直为 1 或 0, 变量 Y3.03 的值为 0。即 Y3.03 只会在 X3.03 下降沿时是 1, 并保持 1 个 PLC 周期, 其余时间都是为 0。

5.4.3 跳变沿

格式

表 5-4-3-1 跳变沿指令格式

语法	名称	指令格式	影响 CR 值
LD	跳变沿		
IL	跳变沿	TST A B C	P
操作数	输入/输出	数据类型	允许的变量类型
IN1	上升沿时的输入 A	unsigned,signed,float,double,bit	MO、Q、常数
IN2	下降沿时的输入 B	unsigned,signed,float,double,bit	MO、Q、常数
OUT	跳变沿满足时的输出 C	unsigned,signed,float,double,bit	MO、Q

IN1, IN2, OUT 的数据类型必须一致。

功能说明

指令改变 CR 值并且给输出赋值，当指令前面的 CR 值由 0 变为 1 产生一个上升沿，通过该指令后的 CR 值为 1，且将 IN1 的值赋给 OUT；当指令前面的 CR 值由 1 变为 0 产生一个下降沿，通过该指令后的 CR 值为 1，且将 IN2 的值赋给 OUT。

举例说明

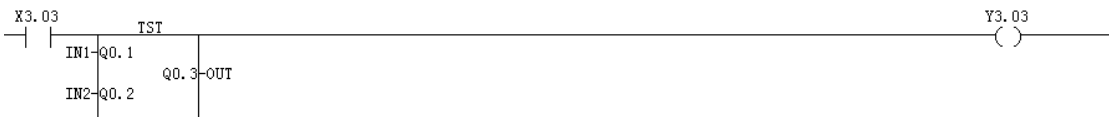


图 5-4-3-1 跳变沿指令梯形图

描述：

当变量 X3.03 由 0 变为 1 为上升沿，变量 Q0.1 的值赋给变量 Q0.3，变量 Y3.03 的值为 1；

当变量 X3.03 由 1 变为 0 为下降沿，变量 Q0.2 的值赋给变量 Q0.3，变量 Y3.03 的值为 1；

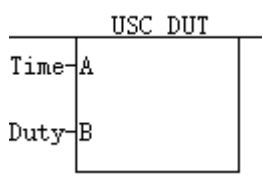
若变量 X3.03 一直为 1 或 0 时，不会给变量 Q0.3 赋值，变量 Y3.03 的值为 0。

即 Y3.03 只会在 X3.03 上升沿或下降沿时是 1，并保持 1 个 PLC 周期，其余时间都是为 0。

5.4.4 自定义占空比方波发生器

格式

表 5-4-4-1 自定义占空方波发生器指令格式

语法	名称	指令格式	影响 CR 值
LD	自定义占空方波发生器		
IL	自定义占空比方波发生器	USC_DUT A B	P
操作数	输入/输出 (A/B)	数据类型	允许的变量类型
IN1	周期, 单位 ms A	Unsigned	常数
IN2	占空比, 范围 (0,1) B	float, double	常数

功能说明

指令改变 CR 值，使通过指令后的 CR 值呈现出方波或矩形波。

举例说明

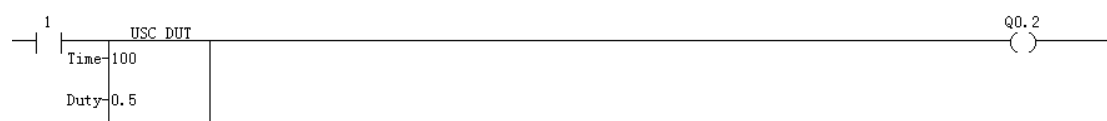


图 5-4-4-1 自定义占空方波发生器指令梯形图

描述:

由于输入节点为 1，一直导通，经过该指令后输出一个周期为 100ms 的方波，Q0.2 的值呈现出一个周期为 100ms 的方波。

5.5 计时器

概述

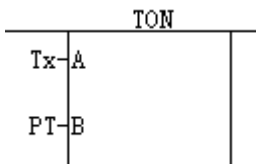
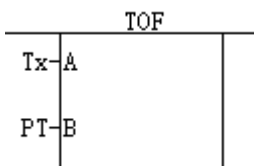
计时器和计数器是两个非常主要的编程元件。对其学习的关键在于其应用的设定值。掌握一些计时器和计数器常用基本控制电路，对在程序中套用是非常有帮助的。

PLC 中计时器相当于继电控制系统中的时间继电器。它在程序中的基本功能是延时控制，但利用计时器可以组成丰富多彩的时序逻辑电路。

5.5.1 指定编号的计时器

格式

表 5-5-1-1 指定编号的计时器指令格式

语法	名称	指令格式	影响 CR 值
LD	延时导通计时器		
	延时断开计时器		
IL	延时导通计时器	TON A B	P
	延时断开计时器	TOF A B	
参数	说明	数据类型	允许的变量类型
Tx	表明计时器的编号 A	Unsigned (0~1023)	常数
PT	延时时间(单位: ms) B	unsigned	所有类型

※ 注意

1. 在计时器工作时，时间从指定时间递减，到剩余时间为 0 结束。这期间可以改变 PT 值，如果改变后的 PT 值小于剩余时间，剩余时间=PT；PT 值大于剩余时间，剩余时间不变；
2. 计时器的精度等于 PLC 的循环时间；

3. 计时器只能用于 PPLC;
4. 同一个工程中不能重复使用同一编号的计时器, 否则容易产生逻辑混乱;

功能说明

TON: 对应编号的计时器延时 PT 时间后电流导通, TON 前的 CR 值为 0 时无任何作用, 当 TON 前的 CR 值从 0 变为 1 时, 经过 TON 后 CR 值仍然保持为 0, 经过 PT 时间后, 经过 TON 后 CR 值变为 1;

TOF: 对应编号的计时器延时 PT 时间后电流关断, TOF 前的 CR 值为 1 时无任何作用, 当 TOF 前的 CR 值从 1 变为 0 时, 经过 TOF 后 CR 值仍然保持为 1, 经过 PT 时间后, 经过 TOF 后 CR 值变为 0。

举例说明



图 5-5-1-1 指定编号延时导通计时器指令梯形图

描述:

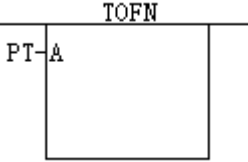
变量 F_SYSREADY 值为 1, 变量 F_PWOFF 值为 0 时, 开始变量 G_PWSTS 值为 0, 延时 500ms 后变量 G_PWSTS 值为 1。

5.5.2 无指定编号的计时器

格式

表 5-5-2-1 无指定编号的计时器指令格式

语法	名称	指令格式	影响 CR 值
LD	延时导通计时器		

	延时断开计时器		
IL	延时导通计时器	TONN A	P
	延时断开计时器	TOFN A	
参数	说明	数据类型	允许的变量类型
PT	延时时间(单位: ms) A	unsigned	所有类型

※ 注意

1. 在计时器工作时, 时间从指定时间递减, 到剩余时间为 0 结束。这期间可以改变 PT 值, 如果改变后的 PT 值小于剩余时间, 剩余时间=PT; PT 值大于剩余时间, 剩余时间不变。计时器的精度等于 PLC 的循环时间。
2. 计时器只能用于 PPLC。
3. 无指定编号的计时器, 避免了编程时计时器重复的问题, 不会产生意想不到的逻辑错乱。推荐使用无指定编号的计时器。

功能说明

TONN: 延时 PT 时间后电流导通, TON 前的 CR 值为 0 时无任何作用, 当 TON 前的 CR 值从 0 变为 1 时, 经过 TON 后 CR 值仍然保持为 0, 经过 PT 时间后, 经过 TON 后 CR 值变为 1;。

TOFN: 延时 PT 时间后电流关断, TOF 前的 CR 值为 1 时无任何作用, 当 TOF 前的 CR 值从 1 变为 0 时, 经过 TOF 后 CR 值仍然保持为 1, 经过 PT 时间后, 经过 TOF 后 CR 值变为 0。

举例说明



图 5-5-2-1 无指定编号延时导通计时器指令梯形图

描述:

变量 F_SYSREADY 值为 1, 变量 F_PWOFF 值为 0 时, 开始变量 G_PWSTS 值为 0, 延时 500ms 后变量 G_PWSTS 值为 1。

5.6 计数器

格式

表 5-6-1 计数器指令格式

语法	名称	指令格式	影响 CR 值
LD	增计数器		
	减计数器		
IL	增计数器	CTU A, B, C	P
	减计数器	CTD A, B, C	
参数	说明	数据类型	允许的变量类型
Cx	表明计数器的编号 A	unsigned (0~255)	常数,
R	计数器复位信号 B	bit	I, O, MI, MO, Q
PV	计数器预置值 C	unsigned	所有类型

※ 注意

在继电器控制电路中, 计数器是作为一种仪表在电路中使用的, 其基本功能是对输入开关量信号进行计数。这里的计数器只能用于 PPLC, 是统计其前面的跳变沿的次数, 而不是高电平的次数, 这点必须要注意。

功能说明

CTU: 增计数器, 在复位信号为 0 时, 计时器前面上跳沿的次数由 0 累加到预置值后, 经过 CTU 后 CR 值为 1。

CTD: 减计数器, 在复位信号为 0 时, 计时器前面上跳沿的次数由预置值减少到 0 后, 经过 CTD 后 CR 值为 1。

举例说明

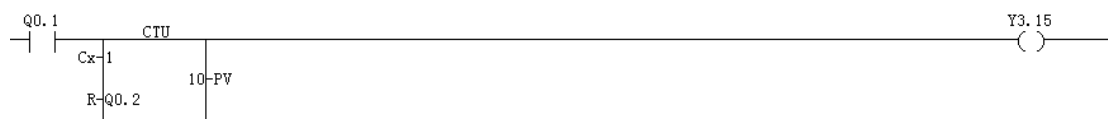


图 5-6-1 增计数器指令梯形图

描述:

当变量 Q0.2 的值为 0 时, 也就是没有复位信号, 变量 Q0.1 值由 0 变为 1 的次数达到 10 次时, 变量 Y3.15 的值为 1。并且由于计数已达到, 将会一直保持 CTU 后的 CR 值为 1, 即 Y3.15 可以保持为 1, 直到复位信号 Q0.2 为 1 时, 计数器 CTU 复位, Y3.15 的值为 0。

6 数值运算指令

概述

本节介绍数值运算指令包括两种运算：整数运算和小数运算。

6.1 整数运算

概述

包括四则运算指令 ADD、SUB、MUL、DIV；取绝对值运算、取余运算；自增一指令 INC、自减一指令 DEC；幂运算 POW、取自然对数 LOGE、取对数 LOG2、循环自增一 INCC、循环自减一 DECC。

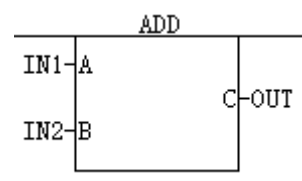
6.1.1 四则运算指令 ADD、SUB、MUL、DIV

指令格式

四则运算指令有四种：加、减、乘、除。指令格式表如下：

- 以加法（ADD）运算为例：

表 6-1-1-1 四则运算指令格式

语法	名称	指令格式	影响 CR 值
LD	加法		
IL	加法	ADD A B C	U
操作数	输入/输出	数据类型	允许的变量类型
INI	输入 A	unsigned, signed, float, double, int64	所有类型
IN2	输入 B	unsigned, signed, float, double, int64	所有类型
OUT	输出 C	unsigned, signed, float, double, int64	O, MO, Q

※ 注意

其中输入输出必须为同一数据类型。

减法（SUB）、乘法（MUL）、除法（DIV）都同加法（ADD）。

功能说明

IN1 和 IN2 相加得到 OUT1。

举例说明

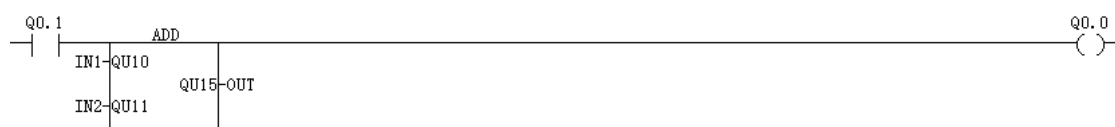


图 6-1-1-1 加法 ADD 运算指令梯形图

描述：

当 Q0.1 为 1 时，将地址 QU10 和 QU11 的值进行加法运算操作后输出到 QU15 指定的地址。

6.1.2 取余取绝对值运算

格式

● 取余（MOD）

表 6-1-2-1 取余（MOD）运算指令格式

语法	名称	指令格式	影响 CR 值
LD	求余		
IL	求余	MOD A B C	U
操作数	输入/输出	数据类型	允许的变量类型
INI	输入 A	unsigned	所有类型
IN2	输入 B	unsigned	所有类型
OUT	输出 C	unsigned	O, MO, Q

其中输入输出必须为同一数据类型。

功能说明

IN1 除 IN2 余数得到 OUT。

例如：

IN1=10, IN2=3

OUT=(IN1) % (IN2)=10%3=1

举例说明



图 6-1-2-1 取余 (MOD) 运算指令梯形图

描述：

当 Q0.1 为 1 时，将地址 QU10 的值和 QU11 的值进行取余操作后输出到 QU15 指定的地址。

● 取绝对值 (ABS)

表 6-1-2-2 取绝对值 (ABS) 运算指令格式

语法	名称	指令格式	影响 CR 值
LD	绝对值		
IL	绝对值	ABS A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	unsigned, signed, float, double, int64	所有类型
OUT	输出 B	unsigned, signed, float, double, int64	O, MO, Q

其中输入输出必须为同一数据类型。

功能说明

IN 取绝对值得到 OUT。

举例说明



图 6-1-2-2 取绝对值运算指令梯形图

描述:

当 Q0.1 为 1 时，将地址 QU10 的值进行取绝对值操作后输出到 QU11 指定的地址。

6.1.3 自增一指令 INC、自减一指令 DEC

格式

● 自增一指令 INC

表 6-1-3-1 自增一指令 INC 格式

语法	名称	指令格式	影响 CR 值
LD	每周期加		
IL	每周期加	INC A	U
操作数	输入/输出	数据类型	允许的变量类型
INI	输入/输出 A	unsigned, signed,	O, MO, Q

功能说明

IN 自加 1。

举例说明



图 6-1-3-1 加 1 指令 INC 梯形图

描述:

当 Q0.1 为 1 时, 将地址 QS10 的值进行加 1。如果 Q0.1 一直保持为 1, 则每扫描一次, 都会进行一次 QS10 的加 1 操作。

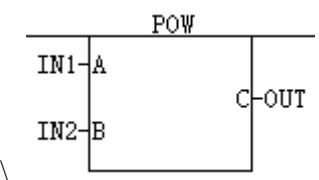
● 自减一指令 DEC

指令格式同自增一指令 INC。

6.1.4 幂运算 POW

格式

表 6-1-4-1 幂运算 POW 指令格式

语法	名称	指令格式	影响 CR 值
LD	幂		
IL	幂	POW A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1	输入 A	float, double	所有类型
IN2	输入 B	float, double	所有类型
OUT	输出 C	float, double	O, MO, Q

其中输入输出必须为同一数据类型。

功能说明

IN1 的 IN2 次幂得到 OUT1。

举例:

IN1=10, IN2=2

OUT=IN1^{IN2}=10²=100

举例说明



图 6-1-4-1 幂运算 POW 指令梯形图

描述:

当 Q0.1 为 1 时,将地址 QF10 的 QF11 次幂进行幂运算操作后输出到 MOF200 指定的地址。

6.1.5 取自然对数取对数运算 LOGE、LOG2

格式

● 取自然对数 LOGE

表 6-1-5-1 取自然对数运算 LOGE 指令格式

语法	名称	指令格式	影响 CR 值
LD	自然对数		
IL	自然对数	LOGE A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	float	所有类型
OUT	输出 B	float	O, MO, Q, S

其中输入输出必须为同一数据类型。

功能说明

IN1 取自然对数得到 OUT1。

例如:

IN1=20

OUT=LOGE(IN1)=LOGE(20)= 1.301

举例说明

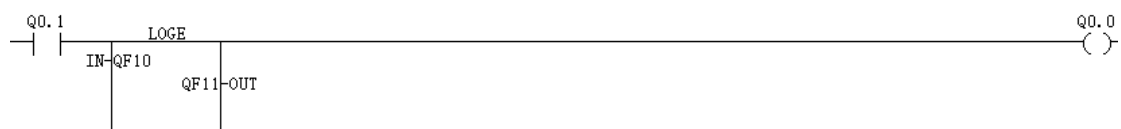


图 6-1-5-1 取自然对数运算 LOGE 指令梯形图

描述:

当 Q0.1 为 1 时，将地址 QF10 的值进行取自然对数运算后输出到 QF11 指定的地址。

● 取对数

表 6-1-5-2 取对数运算 LOG2 指令格式

语法	名称	指令格式	影响 CR 值
LD	对数		
IL	对数	LOG2 A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	Float	所有类型
OUT	输出 B	Float	O, MO, Q, S

其中输入输出必须为同一数据类型。

功能说明

IN1 取对数得到 OUT1。

例如:

IN1=20

OUT= LOG2(IN1)=LOG2(20)= 4.322

举例说明

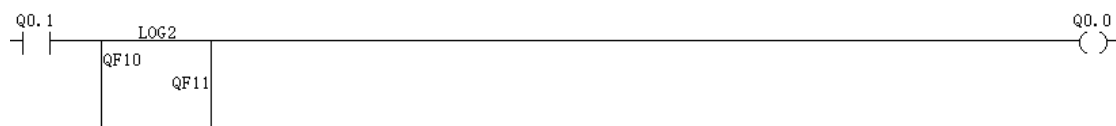


图 6-1-5-2 取对数运算 LOG2 指令梯形图

描述:

当 Q0.1 为 1 时，将地址 QF10 的值进行取对数运算后输出到 QF11 指定的地址。

6.1.6 循环自增一指令 INCC、循环自减一指令 DECC

格式

● 循环自增一指令 INCC

表 6-1-6-1 循环自增一 INCC 指令格式

语法	名称	指令格式	影响 CR 值
LD	每周期加		
IL	每周期加	INCC A B	P
操作数	输入/输出	数据类型	允许的变量类型
INI	自增一起始值 A	unsigned	MO, Q
IN2	自增一最大值 B	unsigned, signed	MO, Q

功能说明

IN1: 进行自增一的起始值;

IN2: 进行自增一的最大值;

从起始值开始每导通一次进行增一操作, 当增加到最大值后, 电流导通, IN1 赋值为起始值, 下一周期再进行增一操作。

举例说明



图 6-1-6-1 循环自增一 INCC 指令梯形图

描述:

当 Q0.1 为 1 时，从起始值 QU10 开始每导通一次进行增一操作，当增加到 QU12 的值后，电流导通，QU10 的值重新从起始值开始，进行下一周期的增一操作。仅当 QU10=QU12 时，电流才导通，其余时刻电流不导通。

- 循环自减一指令 DECC

指令格式同加 1 指令 INCC。

6.2 三角函数运算

概述

三角函数指令也是一类较为常见的指令，包括正弦、余弦、正切、余切，反正弦、反余弦、反正切等指令。

6.2.1 SIN、COS、TAN、CTAN 函数

格式

- SIN 函数

表 6-2-1-1 SIN 函数指令格式

语法	名称	指令格式	影响 CR 值
LD	正弦		
IL	正弦	SIN A B	U
操作数	输入/输出	数据类型	允许的变量类型

IN	输入 A	float, double	所有类型
OUT	输出 B	float, double	O, MO, Q

其中输入输出必须为同一数据类型。

其他的三角函数指令 COS、TAN、CTAN 同 SIN 函数。

功能说明

IN1 取正弦操作后得到 OUT1。

例如：

如下图所示坐标系，单位圆半径为 r ，在三角形中即为其斜边。

- SIN 函数：IN 所对应的角的对边比斜边，即 OUT 为 y/r ；
- COS 函数：IN 所对应的角的邻边比斜边，即 OUT 为 x/r ；
- TAN 函数：IN 所对应的角的对边比邻边，即 OUT 为 y/x ；
- CTAN 函数：IN 所对应的角的邻边比对边，即 OUT 为 x/y ；

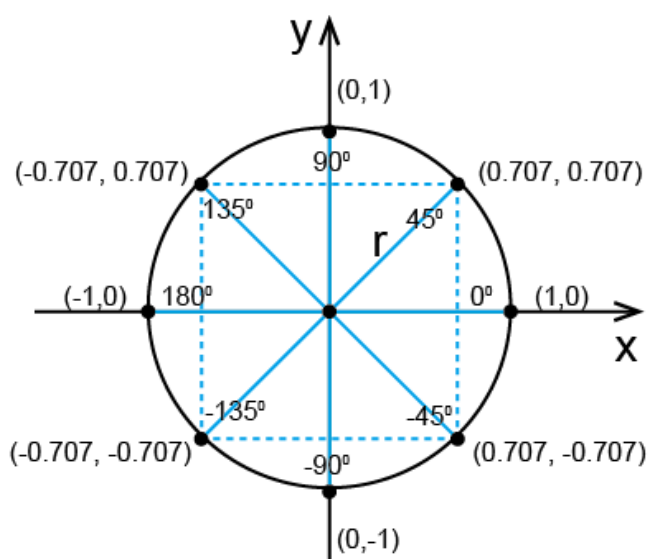


图 6-2-1-1 单位圆

角度不能直接作为输入值，必须先转换成弧度后才能进行运算，公式为：弧度=角度* π /180。

对照上图，下面列出各象限常见角度的三角函数对照表：

表 6-2-1-1 常见三角函数对照表

角度	IN (弧度)	OUT			
		SIN	COS	TAN	CTAN
0°	0	0	1	0	
45°	0.785	0.707	0.707	1	1
90°	1.571	1	0		0
135°	2.356	0.707	-0.707	-1	-1
180°	3.142	0	-1	0	
-135°	-2.356	-0.707	-0.707	1	1
-90°	-1.571	-1	0		0
-45°	-0.785	-0.707	0.707	-1	-1

举例说明

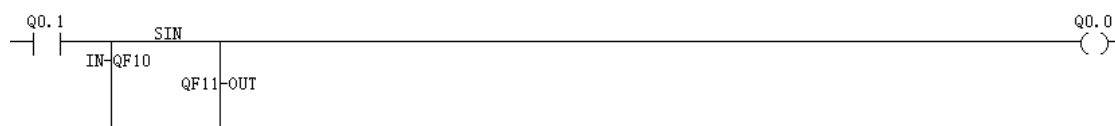


图 6-2-1-1 正弦 SIN 函数指令梯形图

描述:

当 Q0.1 为 1 时，将地址 QF10 的值进行正弦运算操作后输出到 QF12 指定的地址。

6.2.2 ASIN、ACOS、ATAN 函数

格式

● ASIN 函数

表 6-2-2-1 ASIN 函数指令格式

语法	名称	指令格式	影响 CR 值
LD	反正弦		
IL	反正弦	ASIN A B	U
操作数	输入/输出	数据类型	允许的变量类型

IN	输入 A	float, double	所有类型
OUT	输出 B	float, double	O, MO, Q

其中输入输出必须为同一数据类型。其他的反三角函数指令 ACOS、ATAN 均同 ASIN 函数。

功能说明

IN1 反正弦操作后得到 OUT1。

例如：

OUT 为 IN 通过反函数运算后所得到的弧度，通过公式换算可得到对应角度：角度=弧度

* $180 \div \pi$ 。对照图 6-2-1-1，下表列出各象限常见反三角函数对照表：

表 6-2-2-1 常见反三角函数对照表

IN	OUT					
	ASIN		ACOS		ATAN	
	弧度	对应角度	弧度	对应角度	弧度	对应角度
0	0	0°	1.571	90°	0	0°
0.707	0.785	45°	0.785	45°		
1	1.571	90°	0	0°		
-0.707	-0.785	-45°	2.356	135°		
-1	-1.571	-90°	3.142	180°		

举例说明



图 6-2-2-1 反正弦 ASIN 函数指令梯形图

描述：

当 Q0.1 为 1 时，将地址 QF11 的值进行反正弦运算操作后输出到 QF12 指定的地址。

7 传送与比较指令

概述

传送和比较指令是功能指令中最常用的指令，在应用程序中使用十分频繁。可以说，这些指令是功能指令中的基本指令。其主要功能是对元件的读写、清零、比较、交换等。

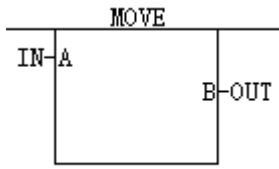
这些指令是 PLC 进行各种数据处理和数值运算的基础，而其本身的应用也可以使一些逻辑运算控制程序得到简化和优化。

7.1 赋值指令

7.1.1 赋值指令 MOVE

格式

表 7-1-1-1 赋值 MOVE 指令格式

语法	名称	指令格式	影响 CR 值
LD	赋值		
IL	赋值	MOVE A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	unsigned,signed,float,double, bit	所有类型
OUT	输出 B	unsigned,signed,float,double,bit	O, MO, Q

其中输入输出必须为同一数据类型。

功能说明

将由 IN 指定的数据或常数赋值到由 OUT 指定的数据(IN→OUT)。

举例说明

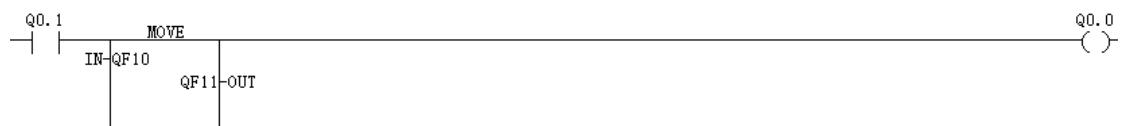


图 7-1-1-1 赋值 MOVE 指令梯形图

描述:

当 Q0.1 为 1 时，将地址 QF10 的值进行赋值操作后输出到 QF11 指定的地址。

7.1.2 批量赋值 BLKMOVE

格式

表 7-1-2-1 批量赋值 BLKMOVE 指令格式

语法	名称	指令格式	影响 CR 值
LD	批量赋值		
IL	批量赋值	BLKMOVE A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	unsigned, signed, float, double	所有类型
LEN	输入 B	unsigned	所有类型
OUT	输出 C	unsigned, signed, float, double	O, MO, Q

功能说明

将从 IN 指定的地址开始、LEN 长度（按照 IN 指定的数据类型计算）的数据一并传输到以 OUT 指定地址开始的地址。

举例说明



图 7-1-2-1 批量赋值 BLKMOVE 指令梯形图

描述:

当 Q0.1 为 1 时, 将地址 QF11 开始的 3 个长度 (这里按照 float 类型计算长度, 每个长度是 4Byte) 的数据进行批量赋值到 OF15 开始的地址中, 即把 QF11、QF12、QF13 的数值, 批量赋值到 QF15、QF16、QF17。

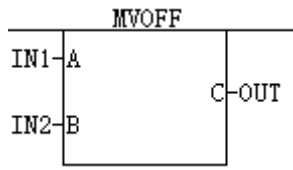
※ 注意

1. 参数 IN 和 OUT 的数据类型必须一致。
2. 因为赋值是按顺序执行的, 如果源地址和目标地址有重合的部分, 那么重合的部分会被前面的数据影响。
3. 例如: MOU1.0.31 = 20, MOU2.0.31=21, MOU3.0.31=22。如果执行 BLKMOVE
MOU1.0.31 3 MOU2.0.31
4. 结果是: MOU1.0.31 = 20, MOU2.0.31=20, MOU3.0.31=20, MOU4.0.31=20。
而不是: MOU1.0.31 = 20, MOU2.0.31=20, MOU3.0.31=21, MOU4.0.31=22。

7.1.3 偏移地址赋值 MOVE_OFFEST

格式

表 7-1-3-1 偏移地址赋值 MOVE_OFFEST 指令格式

语法	名称	指令格式	影响 CR 值
LD	偏移地址赋值		
IL	偏移地址赋值	MVOFF A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1	输入 A	unsigned,signed,float, double,bit	所有类型
IN2	输入 B	unsigned	所有类型
OUT	输出 C	unsigned, signed,float,double, bit	O, MO, Q

其中 IN1,OUT 必须为同一数据类型。

功能说明

把寄存器 IN1 的地址加上偏移量 IN2 后的地址中的值赋给 OUT。

举例说明



图 7-1-3-1 偏移地址赋值 MOVE_OFFSET 指令梯形图

描述:

当 Q0.1 为 1 时，将地址 R1000 加上偏移量 3 后的地址内的值赋给 R1500，即将 R1003 的值赋给 R1500。

例如:

```
MVOFF R1000 3 R1500
```

```
R1003=10
```

执行后 $R1500 = R(1000+3) = 10$ ，R1000 保持不变。

7.1.4 赋值偏移 MOVE_TO

格式

表 7-1-4-1 赋值偏移 MOVE_TO 指令格式

语法	名称	指令格式	影响 CR 值
LD	赋值偏移		
IL	赋值偏移	MVTO A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1	输入 A	unsigned,signed, float, double,int64,bit	所有类型
IN2	输入 B	unsigned	所有类型
OUT	输出 C	unsigned, signed, float, double,int64, bit	O, MO, Q

其中 IN1,OUT 必须为同一数据类型。

功能说明

把寄存器 IN1 后的值赋给 OUT 再加上偏移量 IN2 后的地址。

举例说明

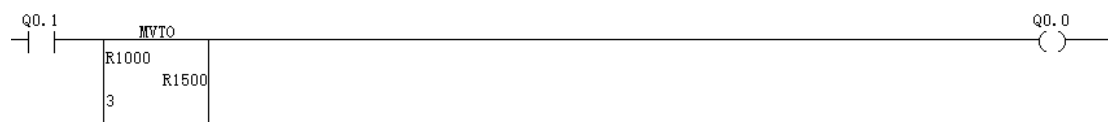


图 7-1-4-1 赋值偏移 MOVE_TO 指令梯形图

描述:

当 Q0.1 为 1 时，将地址 R1000 的值输出到 R1500 地址加上偏移量 3 后生成的地址中，即将 R1000 的值赋给 R1503。

例如:

```
MVTO R1000 3 R1500
```

R1000=10

执行后 $R(1500+3) = R1000 = 10$ ，R1500 保持不变。

7.2 比较指令

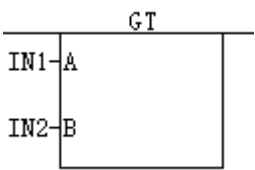
概述

比较指令也是 PLC 中比较常见的指令，包括整数的比较以及浮点数的比较。

7.2.1 大于 GT/大于等于 GE

格式

表 7-2-1-1 大于 GT/大于等于 GE 指令格式

语法	名称	指令格式	影响 CR 值
LD	大于		
IL	大于	GT A B	P
操作数	输入/输出	数据类型	允许的变量类型
IN1	输入 A	unsigned,signed,float,double	所有类型
IN2	输入 B	unsigned,signed,float,double	所有类型

其中 IN1,IN2 必须为同一数据类型。

功能说明

比较两个参数的大小， $IN1 > IN2$ 经过 GT 的 CR 值为 1，否则 CR 值为 0。

举例说明



图 7-2-1-1 大于 GT 指令梯形图

描述:

当 Q0.1 为 1 时，将输入参数变量 QU10 的值与参数变量 QU11 的值进行比较，若 QU10 的值大于 QU11 的值，Q0.2 的值为 1；否则 Q0.2 的值为 0。

例如：

大于等于 GE，小于等于 LE，小于 LT，等于 EQ，不等于 NE，参照上述指令表。

7.2.2 浮点数等于 DBEQ/浮点数不等于 DBNE

格式

表 7-2-2-1 浮点数等于 DBEQ/浮点数不等于 DBNE 指令格式

语法	名称	指令格式	影响 CR 值
LD	DBEQ		
	DBNE		
IL	DBEQ	DBEQ A B C	P
	DBNE	DBNE A B C	P
操作数	输入/输出	数据类型	允许的变量类型
IN1	比较数 1 A	unsigned,signed,float,double	所有类型
IN2	比较数 2 B	unsigned,signed,float,double	所有类型
Limits	条件：判断范围 C	unsigned,signed,float,double	所有类型

其中输入输出必须为同一数据类型。

功能说明

- ①. DBEQ：比较两个参数的大小，若 IN1 的值减去 IN2 的值的绝对值小于等于 Limits 则经过 DBEQ 的 CR 值为 1，否则 CR 值为 0。
- ②. DBNE：比较两个参数的大小，若 IN1 的值减去 IN2 的值的绝对值大于 Limits 则经过 DBEQ 的 CR 值为 1，否则 CR 值为 0。

举例说明



图 7-2-2-1 浮点数等于 DBEQ 指令梯形图

描述:

当 Q0.1 为 1 时，将参数变量 QF10 的值减去参数变量 QF11 的值的绝对值后，若小于等于 0.0001 则 Y3.15 的值为 1，若大于 0.0001 则 Y3.15 的值为 0。

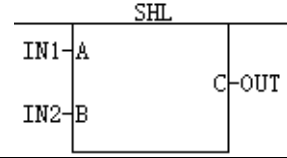
8 移位指令

概述

移位指令的功能是对数据进行左、右移动。其中有对组件组合进行左右移位的指令 SHL 和 SHR。

• 位左移指令 SHL

表 8-1 位左移 SHL 指令格式

语法	名称	指令格式	影响 CR 值
LD	左移		
IL	左移	SHL A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1	输入 A	unsigned, signed	所有类型
IN2	输入 B	unsigned	所有类型
OUT	输出 C	unsigned, signed	所有类型

其中参数 IN1 和 OUT 必须为同一数据类型。

功能说明

IN1 左移 IN2 位后的值赋给 OUT。

举例说明



图 8-1 位左移 SHL 指令梯形图

描述:

当 Q0.1 为 1 时，将参数 QU10 左移 1 位的结果输出到 QU11 的地址中。

※ 注意

位右移指令 SHR 请参考位左移指令 SHL。

9 数据处理指令

概述

广义的数据处理是指针对数据的采集、存储、检索、变换、传送显示及数表的处理。实际上，PLC 的控制功能就是对控制系统的的功能。因此可以说全部功能指令都是数据处理指令。

本章节讲的数据处理指令相对狭义，包括类型转换和 AscII 字符编码。

9.1 类型转换

概述

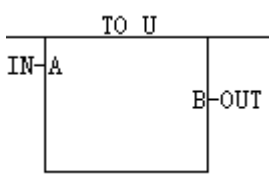
计算机里的数是用二进制表示的，最左边的这一位一般用来表示这个数是正数还是负数，这样的话这个数就是有符号整数。如果最左边这一位不用来表示正负，而是和后面的连在一起表示整数，那么就不能区分这个数是正还是负，就只能是正数，这就是无符号整数。

在这里介绍转无符号整数 TO_U、转有符号整数 TO_S、转浮点数 TO_F、转双精度浮点数 TO_D。

9.1.1 转无符号整数 TO_U

格式

表 9-1-1-1 转无符号整数 TO_U 指令格式

语法	名称	指令格式	影响 CR 值
LD	转无符号整数		
IL	转无符号整数	TO_U A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	signed, float, double	所有类型
OUT	输出 B	unsigned	O, MO, Q

功能说明

IN 转成无符号整数赋给 OUT。

指令说明

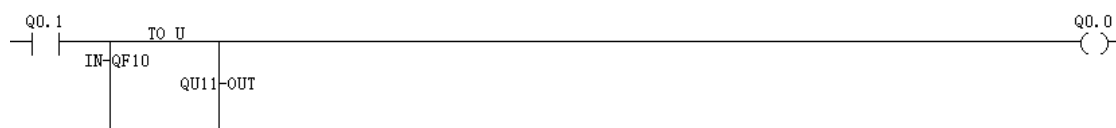


图 9-1-1-1 转无符号整数 TO_U 指令梯形图

描述：

当 Q0.1 为 1 时，将参数 QF10 的值转为无符号整数后输出到 QU11 的地址中。

9.1.2 转有符号整数 TO_S

格式

表 9-1-2-1 转有符号整数 TO_S 指令格式

语法	名称	指令格式	影响 CR 值
LD	转有符号整数		
IL	转有符号整数	TO_S A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	unsigned, float, double	所有类型
OUT	输出 B	signed	O, MO, Q

功能说明

IN 转换成有符号整数赋给 OUT。

举例说明



图 9-1-2-1 转有符号整数 TO_S 指令梯形图

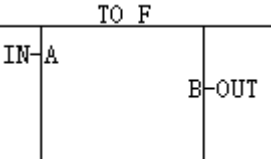
描述:

当 Q0.1 为 1 时, 将参数 QF10 的值转为有符号整数后输出到 QS11 的地址中。

9.1.3 转浮点数 TO_F

格式

表 9-1-3-1 转浮点数 TO_F 指令格式

语法	名称	指令格式	影响 CR 值
LD	转浮点数		
IL	转浮点数	TO_F A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	unsigned, signed, double	所有类型
OUT	输出 B	float	O, MO, Q

功能说明

IN 转换成浮点数后赋给 OUT。

举例说明



图 9-1-3-1 转浮点数 TO_F 指令梯形图

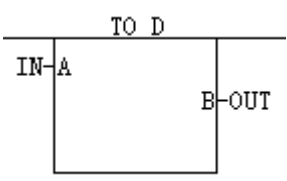
描述:

当 Q0.1 为 1 时, 将参数 QU10 通过转浮点型操作后输出到 QF11 的地址中。

9.1.4 转双精度浮点数 TO_D

格式

表 9-1-4-1 转双精度浮点数 TO_D 指令格式

语法	名称	指令格式	影响 CR 值
LD	转双精度浮点型		
IL	转双精度浮点型	TO_D A B	U
操作数	输入/输出	数据类型	允许的变量类型
IN	输入 A	unsigned, signed, float, int64	所有类型
OUT	输出 B	double	O, MO, Q

功能说明

IN 转换成双精度浮点数后赋给 OUT。

举例说明



图 9-1-4-1 转双精度浮点数 TO_D 指令梯形图

当 Q0.1 为 1 时，将参数 QU10 通过转双精度浮点型操作后输出到 QD11 的地址中。

9.2 数据交换 XCH

格式

表 9-2-1 数据交换 XCH 指令格式

语法	名称	指令格式	影响 CR 值
LD	XCH	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> XCH Q0.2 Q0.3 </div>	
IL	XCH	XCH A B	U
参数	说明	数据类型	允许的变量类型
IN1	参数 1	unsigned,signed,float,double,bit	MO, Q
IN2	参数 2	unsigned,signed,float,double,bit	MO, Q

功能说明

数据交换：互换两个输入参数的数值，两个输入参数的数据类型必须一致

举例说明

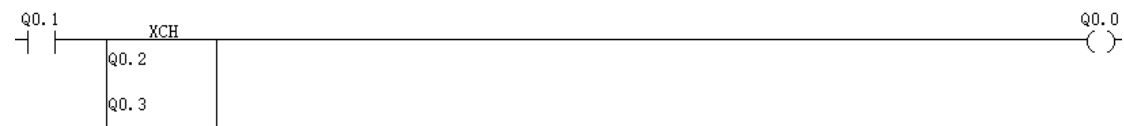


图 9-2-1 数据交换 XCH 指令梯形图

描述：

当 Q0.1 为 1 时，将 Q0.2 的值与 Q0.3 的值进行互换。

9.3 ASCII 字符编码

概述

ASCII 编码是美国国家标准学会制定的信息交换标准代码，它包括 10 个数字，26 个大写字母、26 个小写字母及大约 25 个特殊符号和一些控制码。ASCII 码规定用七位或者八位二进制数组合来表示 128 种或 256 种的字符及控制码。标准 ASCII 码是用七位二进制组合来表示数字、字母、符号和控制码。

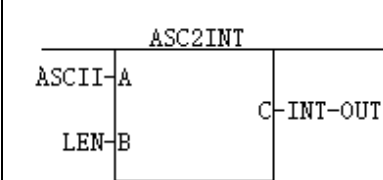
下面介绍 ASCII 码和整数之间的转换：

ASCII2INT：是指将 ASCII 码转换成整数的操作。

INT2ASCII：是指将整数转换成 ASCII 码的操作。

● ASCII 码转成整数

表 9-3-1 ASCII 码转成整数指令格式

语法	名称	指令格式	影响 CR 值
LD	ASCII 码转成整数		
IL	ASCII 码转成整数	ASC2INT A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1	输入 A	字符串	MO, Q
IN2	输入[0,10] B	unsigned	常数
OUT	输出 C	unsigned, signed	MO, Q

功能说明

将由 IN1 开始的 IN2 个数由 ASCII 码转为整数赋给 OUT。最大长度 10。

举例说明



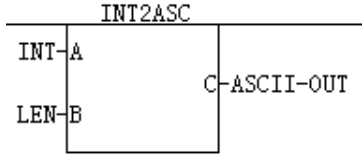
图 9-3-1 ASCII 码转成整数指令梯形图

描述：

当 Q0.1 为 1 时，将字符串“6534”由 ASCII 码转换成整数输出到 QU11，QU11=6534。

● 整数转成 ASCII 码

表 9-3-2 ASCII 码转成整数指令格式

语法	名称	指令格式	影响 CR 值
LD	整数转成 ASCII 码		
IL	整数转成 ASCII 码	INT2ASC A B C	U
操作数	输入/输出	数据类型	允许的变量类型
IN1	输入 A	字符串	MO, Q
IN2	输入 [0,10] B	unsigned	常数
OUT	输出 C	unsigned, signed	O, MO, Q

功能说明

将由 IN1 开始的 IN2 个数由整数转为 ASCII 码赋给 OUT。最大长度 10。

举例说明



图 9-3-2 ASCII 码转成整数指令梯形图

描述:

当 Q0.1 为 1 时，将地址 QU1 的值得前 4 位由整数转换成 ASCII 码输出到 QU2 的地址中。

10 程序流程控制指令

概述

在 PLC 程序控制流程中，需要用到程序的跳转、调用、控制等多种指令。

10.1 PLC 程序结构和程序流程

概述

在 Soft PLC 程序中，主要的程序流程分为两种：从上到下，从左到右的顺序控制；另外是程序会发生跳转，即当跳转条件成立时，扫描会跳过这一部分程序，向前或向后转移到指令的位置继续扫描。

10.2 程序流程控制指令

概述

现介绍如下指令：

- Lab-标签
- J-跳转
- CAL-调用子程序
- PLOG-流程控制

1、Lab-标签

格式

表 10-2-1 Lab 标签指令格式

语法	名称	指令格式	影响 CR 值
LD	跳转	--(lab)	
IL	跳转	Label Name	U

※ 注意

Label 在 Ladder 中独占一个 Rung。Label 名称是任意小于 31 个字符的字符串。

2、J-跳转

格式

表 10-2-2 J-跳转指令格式

语法	名称	指令格式	影响 CR 值
LD	跳转	--(J)	
IL	跳转	JMP LABEL	U
参数		说明	
Lbl		Label 名(参见 lable)。当有电流通过(CR 为 TRUE)程序跳转到 label 后面的 rung 执行, 如果 label 后面没有 rung, 那么 PLC 执行结束。	

※ 注意

使用跳转指令可能会引起死循环, 导致 PLC 内核超时而停止运行, 这时必须重启系统。

3、CALL-调用子程序

格式

表 10-2-3 CALL-调用子程序指令格式

语法	名称	指令格式	影响 CR 值
LD	调用 PPLC 子程序, 同步调用, 程序结束后返回	--(CAL)	
	调用 SPLC 子程序, 异步调用, 立刻返回		
IL	调用 PPLC 子程序	CALL N	U
	调用 SPLC 程序/子程序, 异步调用, 立刻返回	CALL N	

11 CNC 扩展功能指令

概述

CNC 扩展功能指令主要包括控制指令、辅助功能指令、主轴功能指令、马达功能指令。

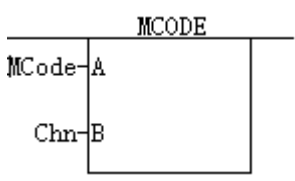
11.1 控制指令

控制指令主要包括系统执行 M 指令，调用 NC 文件以及获取 T 指令编号等功能。

11.1.1 MCODE 指令

格式

表 11-1-1-1 MCODE 指令格式

语法	名称	指令格式	影响 CR 值
LD	MCODE		
IL	MCODE	MCODE A B	P
参数	说明	数据类型	允许的变量类型
MCode	指令的 M 指令编号[0,9999] A	Unsigned	常数
Chn	路径号(1 或 2) B	Unsigned	常数

功能说明

M 指令执行条件

Mcode: M 指令编号

Chn: 路径号

当系统执行到 M 指令时，会对 MCODE 模块导通一次，后自动断开。

举例说明



图 11-1-1-1 MCODE 指令梯形图

描述:

当 Q0.1 为 1 时，当系统执行 M01 时，电流导通，Y3.15 的值为 1，否则 Y3.15 的值为 0。

※ 注意:

每一个 MCODE 都需要匹配的 MFINISH 指令，否则系统会因为等不到 M 指令的结束信号而一直等待，不能继续往下运行。

11.1.2 MFINISH 指令

格式

表 11-1-2-1 MFINISH 指令格式

语法	名称	指令格式	影响 CR 值
LD	MFINISH		
IL	MFINISH	MFINISH A	P
参数	说明	数据类型	允许的变量类型
Chn	路径号(1 或 2) A	unsigned	常数

功能说明

M 指令执行完成：该组件直接导通，返回给系统 M 指令执行完成信号。

举例说明



图 11-1-2-1 MFINISH 指令梯形图

描述:

当 Q0.1 为 1 时，通过 MFINISH 后返回系统 M 指令完成信号，NC 程序中的 M 指令会结束等待，继续执行后面的 NC 指令。

11.1.3 MCALL 指令

格式

表 11-1-3-1 MCALL 指令格式

语法	名称	指令格式	影响 CR 值
LD	MCALL		
IL	MCALL	MCALL A B	P
参数	说明	数据类型	允许的变量类型
Sub	指令的 M 指令编号 A	unsigned,signed	常数,MO,Q
Chn	路径号(1 或 2) B	unsigned	常数

功能说明

调用指定的 NC 文件。

Sub: 调用的 NC 文件编号。

Chn: 路径号，路径号为 1 时，调用/home/Lynuc/Users/NCFiles/路径下的文件；路径号为 2 时，调用/home/Lynuc/Users/NCFiles/Chn/路径下的文件。

举例说明



图 11-1-3-1 MCALL 指令梯形图

描述:

当 Q0.1 为 1 时，系统会调用/home/Lynuc/Users/NCFiles/Chn/路径下的 O0001.NC 文件。

11.1.4 T 指令获取 TGET

格式

表 11-1-4-1 T 指令获取 TGET 指令格式

语法	名称	指令格式	影响 CR 值
LD	TGET		
IL	TGET	TGET A B	U
参数	说明	数据类型	允许的变量类型
IN1	路径号(1,2)	Unsigned,	常数, MO, Q
OUT	T 指令编号	Unsigned	MO, Q

功能说明

T 指令获取：通过输入的路径号获取 T 指令编号（刀具号）。

举例说明



图 11-1-4-1 T 指令获取 TGET 指令梯形图

描述:

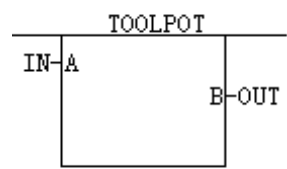
当 Q0.1 为 1 时, 系统依据 QU11 所输入的路径号获取 T 指令编号, 即当前加工的刀具号, 并存放在 QU12 中。

11.2 辅助功能指令

11.2.1 由刀具号获取刀套号 TOOLPOT

格式

表 11-2-1-1 由刀具号获取刀套号 TOOLPOT 指令格式

语法	名称	指令格式	影响 CR 值
LD	TOOLPOT		
IL	TOOLPOT	TOOLPOT A B	U
参数	说明	数据类型	允许的变量类型
IN1	刀具号	unsigned	常数, MO, Q
OUT	刀套号	unsigned	MO, Q

功能说明

刀具对应刀套号: 由刀具号获取到对应的刀套号。

举例说明



图 11-2-1-1 由刀具号获取刀套号 TOOLPOT 指令梯形图

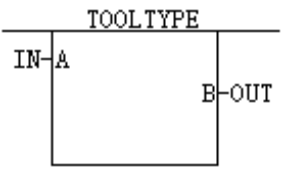
描述:

当 Q0.1 为 1 时, 系统依据 Q11 所输入的刀具号获取相应的刀套号并存放在 QU12 中。

11.2.2 刀具类型 TOOLTYPE

格式

表 11-2-2-1 刀具类型 TOOLTYPE 指令格式

语法	名称	指令格式	影响 CR 值
LD	TOOLTYPE		
IL	TOOLTYPE	TOOLTYPE A B	U
参数	说明	数据类型	允许的变量类型
IN1	刀具号	unsigned	常数, MO, Q
OUT	刀具类型	Bit (0, 小刀; 1, 大刀)	MO, Q

功能说明

刀具类型：由刀具号获取对应的刀具类型(1 为大刀,0 为小刀)。

举例说明

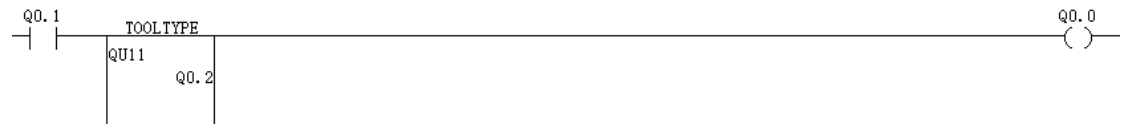


图 11-2-2-1 刀具类型 TOOLTYPE 指令梯形图

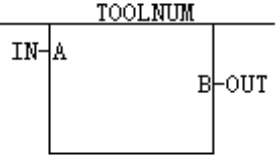
描述：

当 Q0.1 为 1 时，系统依据 QU11 所输入的刀具号获取相应的刀具类型(1 为大刀,0 为小刀)并存放在 Q0.2 中。

11.2.3 由刀套获取刀具号 TOOLNUM

格式

表 11-2-3-1 由刀套获取刀具号 TOOLNUM 指令格式

语法	名称	指令格式	影响 CR 值
LD	TOOLNUM		
IL	TOOLNUM	TOOLNUM A B	U
参数	说明	数据类型	允许的变量类型
IN1	刀套号	unsigned	MO, Q
OUT	刀具号	unsigned	MO, Q

功能说明

刀套对应刀具号：由刀套号获取对应的刀具号。

举例说明



图 11-2-3-1 由刀套获取刀具号 TOOLNUM 指令梯形图

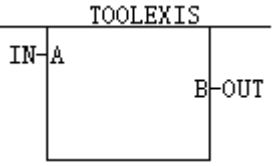
描述：

当 Q0.1 为 1 时，系统依据 QU11 所输入的刀套号获取相应的刀具号并存放在 QU12 中。

11.2.4 刀套类型 TOOLEXIS

格式

表 11-2-4-1 刀套类型 TOOLEXIS 指令格式

语法	名称	指令格式	影响 CR 值
LD	TOOLEXIS		
IL	TOOLEXIS	TOOLEXIS A B	U
参数	说明	数据类型	允许的变量类型
IN1	刀套号	unsigned	常数, MO, Q
OUT	刀套类型	bit	MO, Q

功能说明

刀套类型：由刀套号获取刀套类型，有刀为 1，无刀为 0。

举例说明

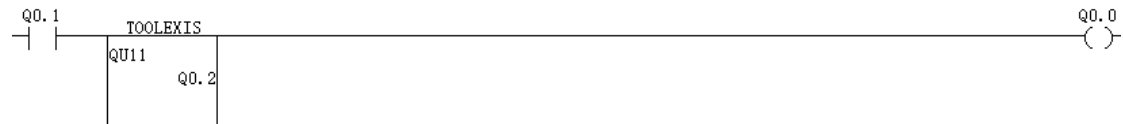


图 11-2-4-1 刀套类型 TOOLEXIS 指令梯形图

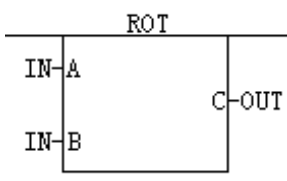
描述：

当 Q0.1 为 1 时，系统依据 QU11 所输入的刀套号获取相应的刀套类型并存放在 Q0.2 中。

11.2.5 旋转控制 ROT

格式

表 11-2-5-1 旋转控制 ROT 指令格式

语法	名称	指令格式	影响 CR 值
LD	ROT		
IL	ROT	ROT A B C	P
参数	说明	数据类型	允许的变量类型
IN1	目标刀套号 A	Unsigned	常数, MO,Q
IN2	最大刀套号 B	Unsigned	常数, MO,Q
OUT	刀库需要旋转的刀套个数 C	Unsigned	MO,Q

功能说明

旋转控制：刀库旋转个数及方向，电流输出 0 为正向旋转，电流输出 1 为反向旋转。

举例说明



图 11-2-5-1 旋转控制 ROT 指令梯形图

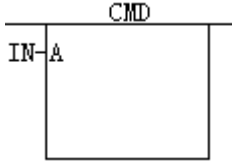
描述：

当 Q0.1 为 1 时，系统依据 QU11 所输入的目标刀套号和 QU12 所输入的最大刀套号计算出刀库需要旋转的刀套个数并存放在 QU13 中。当 Q0.2 为 0 时，后续逻辑实现刀库正向旋转，反之，当 Q0.2 为 1 时，刀库反向旋转。

11.2.6 调度内核 COMMAND_LINE

格式

表 11-2-6-1 调度内核 COMMAND_LINE 指令格式

语法	名称	指令格式	影响 CR 值
LD	CMD		
IL	CMD	CMD % A	U
参数		说明	
IN A		命令字符串	

功能说明

命令行

调度内核的在线命令

举例说明



图 11-2-6-1 调度内核 COMMAND_LINE 指令梯形图

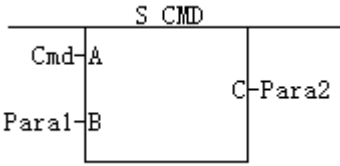
描述:

当变量 F_PWSTS 值为 0 时，延时 200ms 后，恰好有一个上升沿，执行命令 CMD，杀掉 6 号马达。

11.2.7 直接调度内核 S_CMD

格式

表 11-2-7-1 直接调度内核 S_CMD 指令格式

语法	名称	指令格式	影响 CR 值
LD	S_CMD		
IL	S_CMD	S_CMD % A B C	U
参数		说明	
IN	A	命令字符串	
	Para1 B	参数 1	
	Para2 C	参数 2	

功能说明

用来直接调度内核的在线命令。可用来实现部分高级功能，变量参数可以灵活给定。

举例说明

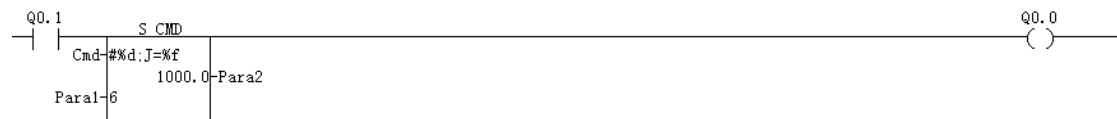


图 11-2-7-1 直接调度内核 S_CMD 指令梯形图

描述:

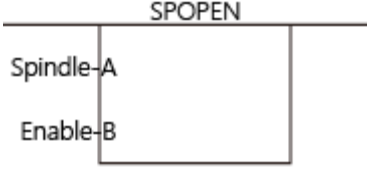
当 Q0.1 为 1 时，执行命令 S_CMD，使 6 号马达到达 1000.0 的目标位置。即“#6;J=1000.0”。

11.3 主轴功能指令

11.3.1 主轴开环使能 SPOPEN

格式

表 11-3-1-1 主轴开环使能 SPOPEN 指令格式

语法	名称	指令格式	影响 CR 值
LD	SPOPEN		
IL	SPOPEN	SPOPEN A B	U
参数	说明	数据类型	允许的变量类型
IN1	主轴编号	unsigned	常数, MO, Q
IN2	使能标记 (1,0)	bit	MO, Q

功能说明

主轴开环使能：设置主轴状态，使能或被杀掉（IN2： 1 代表使能主轴， 0 代表杀掉主轴）。

举例说明

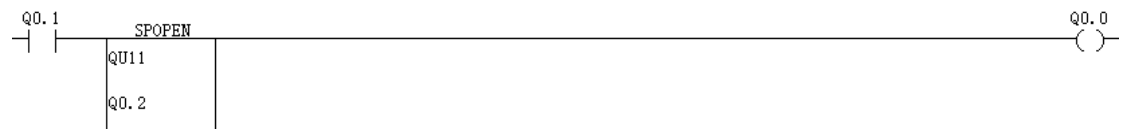


图 11-3-1-1 主轴开环使能 SPOPEN 指令梯形图

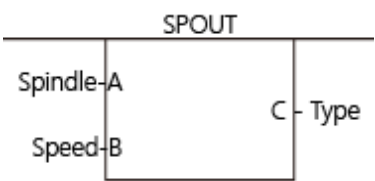
描述：

当 Q0.1 为 1 时，系统依据所输入的主轴编号 QU11 和使能标记 Q0.2 设置主轴状态。当 Q0.2 的值为 1 时，表示指定编号的主轴使能；当 Q0.2 的值为 0 时，表示指定编号的主轴被杀掉。

11.3.2 主轴输出 SPOUT

格式

表 11-3-2-1 主轴输出 SPOUT 指令格式

语法	名称	指令格式	影响 CR 值
LD	SPOUT		
IL	SPOUT	SPOUT A B C	U
参数	说明	数据类型	允许的变量类型
IN1	主轴编号	unsigned	常数, MO, Q
IN2	转速	signed	MO, Q
IN3	设备类型 (1,0)	bit	MO, Q

功能说明

主轴输出：将指定的主轴按照给定的转速及设备类型（脉冲型为 0，总线型为 1）进行旋转。
指定的转速单位是 RPM（转每分钟）。

举例说明



图 11-3-2-1 主轴输出 SPOUT 指令梯形图

描述：

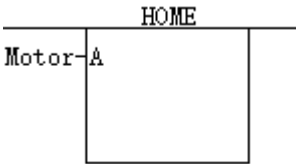
当 Q0.1 为 1 时，系统依据所输入的主轴编号 QU11、转速 QU12 及设备类型（1 为总线型）进行旋转。

11.4 马达功能指令

11.4.1 马达回零 Home

格式

表 11-4-1-1 马达回零 Home 指令格式

语法	名称	指令格式	影响 CR 值
LD	HOME		
IL	HOME	HOME A	P
参数	说明	数据类型	允许的变量类型
Motor	马达编号 A	unsigned	常数,MO,Q

功能说明

执行马达回零动作：

Motor：马达编号，

HOME 需要一直导通才能进行回零动作，一旦断开，回零动作会立即停止；直到回零动作完成后，HOME 模块后面的逻辑会导通一个 PLC 周期，后又断开。

举例说明



图 11-4-1-1 马达回零 Home 指令梯形图

描述：

当 Q0.1 为 1 时，将指令 1 执行马达回零操作。

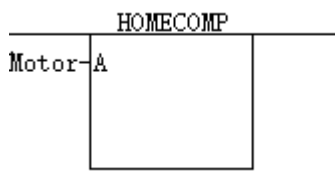
※ 注意

HOME 动作完成后，已经自动设置了回零完成标记，不需要再执行 HOMECOMP。

11.4.2 马达回零完成标记 Home COMP

格式

表 11-4-2-1 马达回零完成标记 Home COMP 指令格式

语法	名称	指令格式	影响 CR 值
LD	HOMECOMP		
IL	HOMECOMP	HOMECOMP A	U
参数	说明	数据类型	允许的变量类型
Motor	马达编号	Unsigned	常数,MO,Q

功能说明

设置马达回零完成标记:

IN: 马达编号。

可用于无回零动作,如串口读取;或不需回零的马达,通过该模块设置马达的回零完成标记。

举例说明

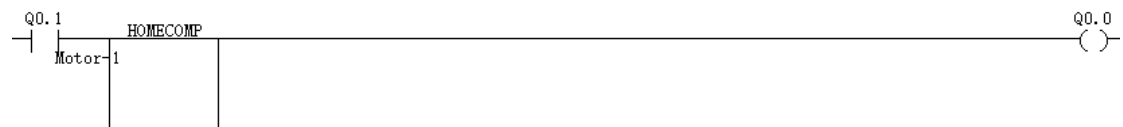


图 11-4-2-1 马达回零完成标记 Home COMP 指令梯形图

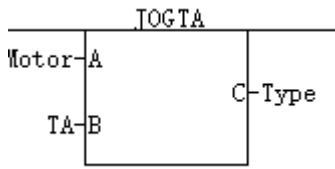
描述:

当 Q0.1 为 1 时,将#1 马达的回零完成标记设为 1,表示回零已完成,#1 零位已确定。

11.4.3 马达加速时间 JOGTA

格式

表 11-4-3-1 马达加速时间 JOGTA 指令格式

语法	名称	指令格式	影响 CR 值
LD	JOGTA		
IL	JOGTA	JOGTA A B C	U
参数	说明	数据类型	允许的变量类型
Motor	指令的马达编号 A	unsigned	常数, MO, Q
TA	加速时间 B	double、ms 单位	常数, R, MO, Q
Type	0: S 型加速 1: 梯形加速 C	unsigned	常数 (0、1), MO, Q

功能说明

指定马达移动的加速时间，以及加减速类型。

Motor: 指令的马达编号

TA: 加速时间

Type 0: S 性加速 1: 梯形加速。

举例说明



图 11-4-3-1 马达加速时间 JOGTA 指令梯形图

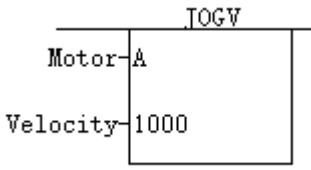
描述:

当 Q0.1 为 1 时，1 号马达进行加速时，保证至少 50ms 的 S 型加速曲线。如果移动距离较大，Jog 加速度相对偏小，则会有匀加速段，整体加速时间 TA 会超出设定时间。

11.4.4 马达移动速度 JOGV

格式

表 11-4-4-1 马达移动速度 JOGV 指令格式

语法	名称	指令格式	影响 CR 值
LD	JOGV		
IL	JOGV	JOGV A B	U
参数	说明	数据类型	允许的变量类型
Motor	指令的马达编号 A	unsigned	常数,MO,Q
Velocity	移动速度 B	double 、 mm/min 单位	R,MI,MO,Q, 常数

功能说明

指定马达移动的速度：

IN1：指令的马达编号，

IN2：移动速度，mm/min 单位。

举例说明



图 11-4-4-1 马达速度 JOGV 指令梯形图

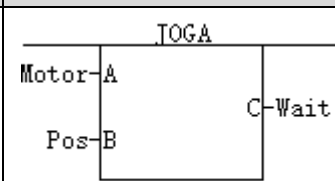
描述：

当 Q0.1 为 1 时，设定 1 号马达的 Jog 速度为 1000mm/min。

11.4.5 马达移动位置 JOGA

格式

表 11-4-5-1 马达移动位置 JOGA 指令格式

语法	名称	指令格式	影响 CR 值
LD	JOGA		
IL	JOGA	JOGA A B C	P
参数	说明	数据类型	允许的变量类型
Motor	指令的马达编号 A	unsigned	常数, MO, Q
Pos	目标位置 B	double 、 mm 单位	R, MI, MO, Q
Wait	是否等待移动结束 C	unsigned	常数 (0、1)

功能说明

指定马达移动到目标位置:

Motor: 指令的马达编号;

Pos: 目标位置 单位 mm;

Wait: 是否等待移动结束。

举例说明

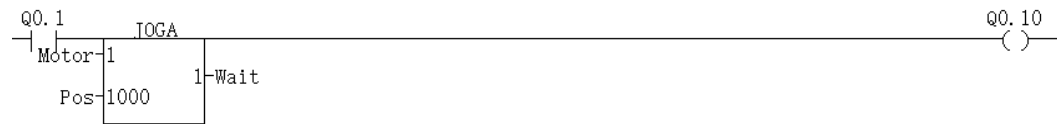


图 11-4-5-1 马达移动位置 JOGA 指令梯形图

描述:

当 Q0.1 为 1 时, 1 号马达移动到目标位置 1000mm, 并等待移动结束后, JOGA 模块后的 CR 值变为 1, 变量 Q0.10 的值为 1。

11.4.6 马达移动距离 JOGI

格式

表 11-4-6-1 马达移动距离 JOGI 指令格式

语法	名称	指令格式	影响 CR 值
LD	JOGI		
IL	JOGI	JOGI A B C	P
参数	说明	数据类型	允许的变量类型
Motor	指令的马达编号 A	unsigned	常数, MO, Q
Pos	移动距离 B	double 、 mm 单位	R, MI, MO, Q
Wait	是否等待移动结束 C	unsigned	常数 (0、1)

功能说明

指定马达移动的距离:

Motor: 指令的马达编号;

Pos: 移动距离 单位 mm;

Wait: 是否等待移动结束。

举例说明



图 11-4-6-1 马达移动距离 JOGI 指令梯形图

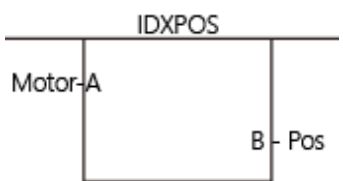
描述:

当 Q0.1 为 1 时, 1 号马达向正向移动距离 1000mm, 并等待移动结束后, JOGI 模块后的 CR 值变为 1, 变量 Q0.10 的值为 1。

11.4.7 Index 位置检测 IDXPOS

格式

表 11-4-7-1 Index 位置检测 IDXPOS 指令格式

语法	名称	指令格式	影响 CR 值
LD	IDXPOS		
IL	IDXPOS	IDXPOS A B	U
参数	说明	数据类型	允许的变量类型
IN1	指令的马达编号 A	unsigned	常数, MO, Q
OUT	将 Index 位置输出到给定变量中	Signed 单位:毫米(mm)	MO, Q

功能说明

Index 位置检测: 获取指令编号马达的 Index 位置并输出到给定变量中。

举例说明



图 11-4-7-1 Index 位置检测 IDXPOS 指令梯形图

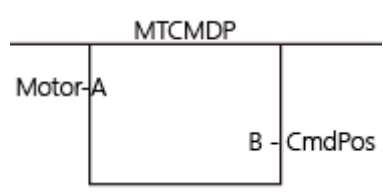
描述:

当 Q0.1 为 1 时, 获取 QU11 所指定编号马达的 Index 位置, 并输出到 QS12 中。

11.4.8 马达指令坐标 MTCMDP

格式

表 11-4-8-1 马达指令坐标 MTCMDP 指令格式

语法	名称	指令格式	影响 CR 值
LD	MTCMDP		
IL	MTCMDP	MTCMDP A B	U
参数	说明	数据类型	允许的变量类型
IN1	马达编号 A	unsigned	常数, MO, Q
OUT	将指令坐标输出到给定变量中 B	Double 单位:毫米(mm)	MO, Q

功能说明

马达指令坐标：获取指定编号马达的指令坐标，并将其输出到给定变量中。

举例说明



图 11-4-8-1 马达指令坐标 MTCMDP 指令梯形图

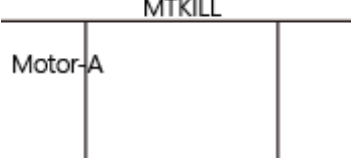
描述：

当 Q0.1 为 1 时，获取 QU11 所指定编号马达的指令坐标，并输出到 QD12 中。

11.4.9 杀掉马达 MTKILL

格式

表 11-4-9-1 杀掉马达 MTKILL 指令格式

语法	名称	指令格式	影响 CR 值
LD	MTKILL		
IL	MTKILL	MTKILL A	U
参数	说明	数据类型	允许的变量类型
IN1	马达编号 A	unsigned	常数, MO, Q

功能说明

杀掉马达：杀掉指定编号的马达。

举例说明



图 11-4-9-1 杀掉马达 MTKILL 指令梯形图

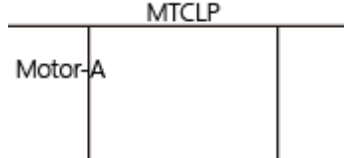
描述：

当 Q0.1 为 1 时，杀掉 QU11 的值所对应编号的马达。

11.4.10 马达闭环 MTCLP

格式

表 11-4-10-1 马达闭环 MTCLP 指令格式

语法	名称	指令格式	影响 CR 值
LD	MTCLP		

IL	MTCLP	MTCLP A	U
参数	说明	数据类型	允许的变量类型
IN1	马达编号 A	unsigned	常数, MO, Q

功能说明

马达闭环：闭环指定编号的马达。

举例说明



图 11-4-10-1 马达闭环 MTCLP 指令梯形图

描述：

当 Q0.1 为 1 时，闭环控制 QU11 所指定编号的马达。

11.4.11 马达使能 MTENA

格式

表 11-4-11-1 马达使能 MTENA 指令格式

语法	名称	指令格式	影响 CR 值
LD	MTENA		
IL	MTENA	MTENA A	U
参数	说明	数据类型	允许的变量类型
IN1	马达编号 A	unsigned	常数, MO, Q

功能说明

马达使能：使能指定编号的马达。

举例说明



图 11-4-11-1 马达使能 MTENA 指令梯形图

描述:

当 Q0.1 为 1 时, 使能 QU11 所指定编号的马达。

11.4.12 马达到位检测 MTINPOS

格式

表 11-4-12-1 马达到位检测 MTINPOS 指令格式

语法	名称	指令格式	影响 CR 值
LD	MTINPOS		
IL	MTINPOS	MTINPOS A	P
参数	说明	数据类型	允许的变量类型
IN1	马达编号 A	unsigned	常数, MO, Q

功能说明

马达到位检测: 马达到位后电流导通, 否则电流不导通。

举例说明



图 11-4-12-1 马达到位检测 MTINPOS 指令梯形图

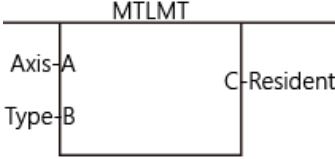
描述:

当 Q0.1 为 1 时, 驱动条件成立, 判断 QU11 所指定编号的马达是否到位, 若到位则电流导通, 否则电流不导通。

11.4.13 马达软限位设置 MTLMT

格式

表 11-4-13-1 马达软限位设置 MTLMT 指令格式

语法	名称	指令格式	影响 CR 值
LD	MTLMT		
IL	MTLMT	MTLMT A B C	P
参数	说明	数据类型	允许的变量类型
IN1	指令的轴号 A	unsigned	常数, MO, Q
IN2	软限位类型 (0,1) B	unsigned	常数, MO, Q
IN3	生效方式 (0,1) C	unsigned	常数, MO, Q

功能说明

马达软限位设置:

IN1: 轴号;

IN2: 软限位类型, 第一软限位请设 0, 第二软限位请设 1;

IN3: 生效方式, 如果需要驻留请设 0, 不仅仅在一个程序循环内部生效, 在手动操作也生效; 如果不需要驻留请设 1, 仅在一个程序循环内部临时生效。

举例说明



图 11-4-13-1 马达软限位设置 MTLMT 指令梯形图

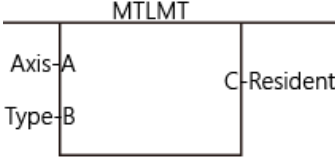
描述:

当 Q0.1 为 1 时, 设置 QU11 对应轴的软限位类型为第二软限位, 仅在当前程序循环内部临时生效。

11.4.14 马达正向运动禁止

格式

表 11-4-14-1 马达正向运动禁止设置 MLIM 指令格式

语法	名称	指令格式	影响 CR 值
LD	P_MLIM		
IL	P_MLIM	MLIM A B	P
参数	说明	数据类型	允许的变量类型
IN1	指令的马达号 A	unsigned	常数, MO, Q
IN2	是否有效 (0,1) B	unsigned	常数, MO, Q

功能说明

正向运动禁止设置:

IN1: 马达号;

IN2: 是否有效, 有效请设 1, 无效请设 0;

举例说明



图 11-4-13-1 马达软限位设置 MTLMT 指令梯形图

描述:

当 Q0.1 为 1 时, 设置 QU11 对应马达禁止正向运动, 仅在当前程序循环内部临时生效。

注: 负向运动禁止类同。

12 Examples

概述

本节介绍了在 PLC 操作中的读写 YIO、产生方波、调用子程序的相关内容。

12.1 读写系统 IO

概述

本 PLC 提供了 X、Y 变量来访问系统 IO 信号。

具体对应的地址表如下：

表 12-1-1 YIO 地址

板卡	输入地址	输出地址
YIO-1	X0	Y0
YIO-2	X1	Y1
YIO-3	X2	Y2
YCP	X3\X4\X5	Y3\Y4\Y5

表 12-1-2 U3/U5 LIO 地址

LIO 输入地址	PLC 地址	LIO 输出地址	PLC 地址
CH1-正限位	X12.0	BRK1	Y12.0
CH1-HOME	X12.1	BRK2	Y12.1
CH1-负限位	X12.2	BRK3	Y12.2
CH1-User1	X12.3	BRK4	Y12.3
CH2-正限位	X12.4	BRK5	Y12.4
CH2-HOME	X12.5	BRK6	Y12.5
CH2-负限位	X12.6	USR4	Y12.6
CH2-User2	X12.7	USR5	Y12.7
CH3-正限位	X12.8	USR6	Y12.8
CH3-HOME	X12.9		
CH3-负限位	X12.10		

CH3-User3	X12.11		
CH4-正限位	X12.12		
CH4-HOME	X12.13		
CH4-负限位	X12.14		
CH5-正限位	X12.15		
CH5-HOME	X12.16		
CH5-负限位	X12.17		
CH6-正限位	X12.18		
CH6-HOME	X12.19		
CH6-负限位	X12.20		

表 12-1-3 N3B LIO 地址

LIO 输入地址	PLC 地址	LIO 输出地址	PLC 地址
IN_0	X12.22	OUT_0	Y12.10
IN_1	X12.23	OUT_1	Y12.11
IN_2	X12.24	OUT_2	Y12.12
IN_3	X12.25	OUT_3	Y12.13
IN_4	X12.26	OUT_4	Y12.14
IN_5	X12.27	OUT_5	Y12.15
IN_6	X12.28	OUT_6	Y12.16
IN_7	X12.29	OUT_7	Y12.17
IN_8	X12.30		
IN_9	X12.31		

说明

以上对应关系是 LYNUC 系统约定的对应关系。

12.2 产生方波

概述

方波是一种非正弦曲线的周期波形，通常会于电子和讯号处理时出现。波形为矩形的电流即为方波电流。在一个波形周期内，高电平所在的时间比率为占空比，比如高电平所占时间为 0.5 个周期，则该方波的占空比为 0.5。

利用计时器很容易产生方波，而且所占空比也可以随意调节。

程序图例

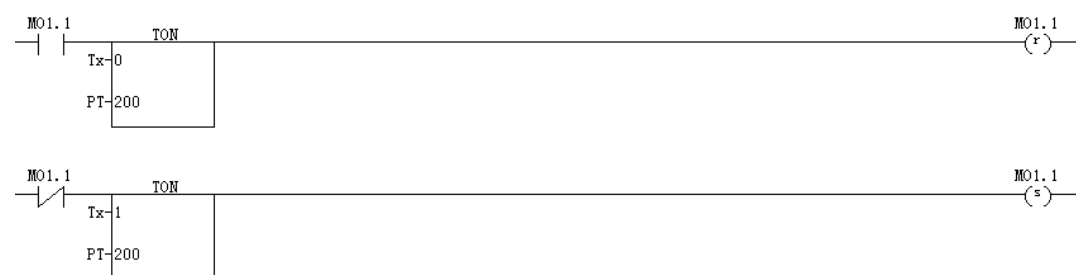


图 12-2-1 计时器梯形图程序示例

描述:

示例程序在 MO1.1 上产生了一个周期为 400 毫秒，占空比为 0.5 的方波。

另外新增了一个自定义占空比脉冲发生器，可产生任意占空比的方波，矩形波，举例如下：



图 12-2-2 占空比脉冲发生器梯形图示例

以上梯形图也可在 MO1.1 产生一个周期为 400 毫秒，占空比为 0.5 的方波，如下图所示：

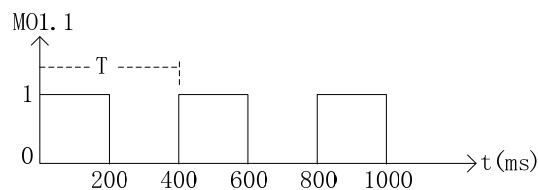


图 12-2-3 方波图例

12.3 调用子程序

概述

子程序是一种重要的程序结构。在实际应用中，经常会遇到需要反复多次执行同一程序段的情形，为减少重复编写的工作量和程序存储空间，常常把功能完整、意义明确并被多次使用的程序段从原来的程序（主程序）中分离出来独立编写，即子程序。主程序可以根据需要多次调用子程序。

举例说明

1. 调用固定子程序

假设某个 PPLC 已经 download 到内存的 PLC1 位置，在其他某个 PLC 中可以使用以下语句调用这个 PLC：



图 12-3-1 调用子程序梯形图示例

2. 调用的子程序号由变量指定：

假设有若干个 PPLC 已经 download 到内存的 PLC2~5 位置上，在其他某个 PLC 中可以使用以下语句调用这些 PLC，PLC 号根据变量 MOU1.0.7 指定：

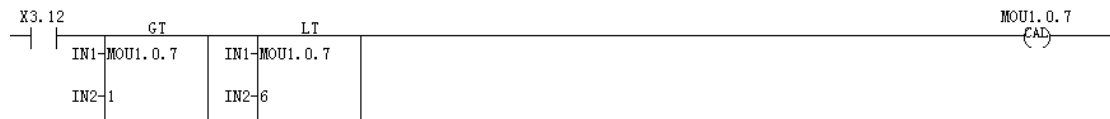


图 12-3-2 调用的子程序号由变量指定梯形图示例

13 配置管理功能

概述

本节涉及到的 PLC 管理功能包括：添加文件、删除文件、编辑文件、导入数据包、生成数据包。详细内容请参照以下各小节。

13.1 添加文件

操作步骤

- ①. 在 PLC 界面菜单栏选择【工程】下的【添加文件】。弹出对话框选择添加的文件类型。

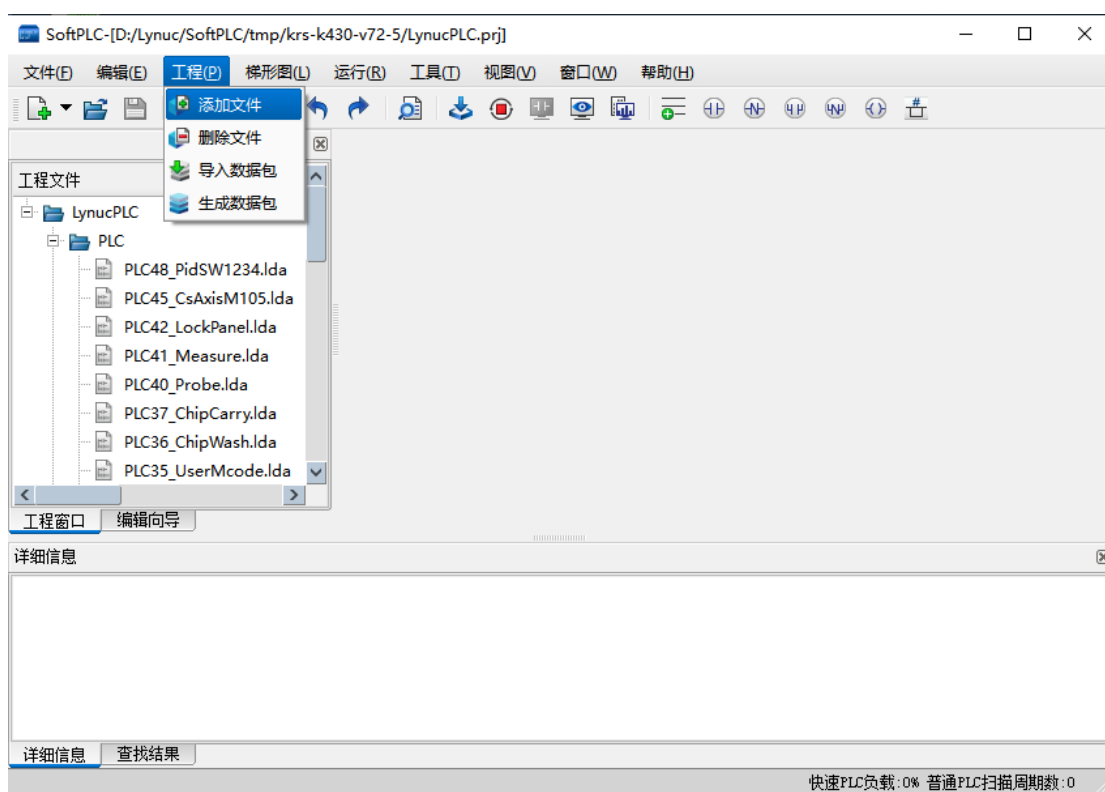


图 13-1-1 添加文件-1

- ②. 然后找到文件所在的路径，选择相应文件，点击确定，即完成了增加该文件到 PLC 工程中的操作。

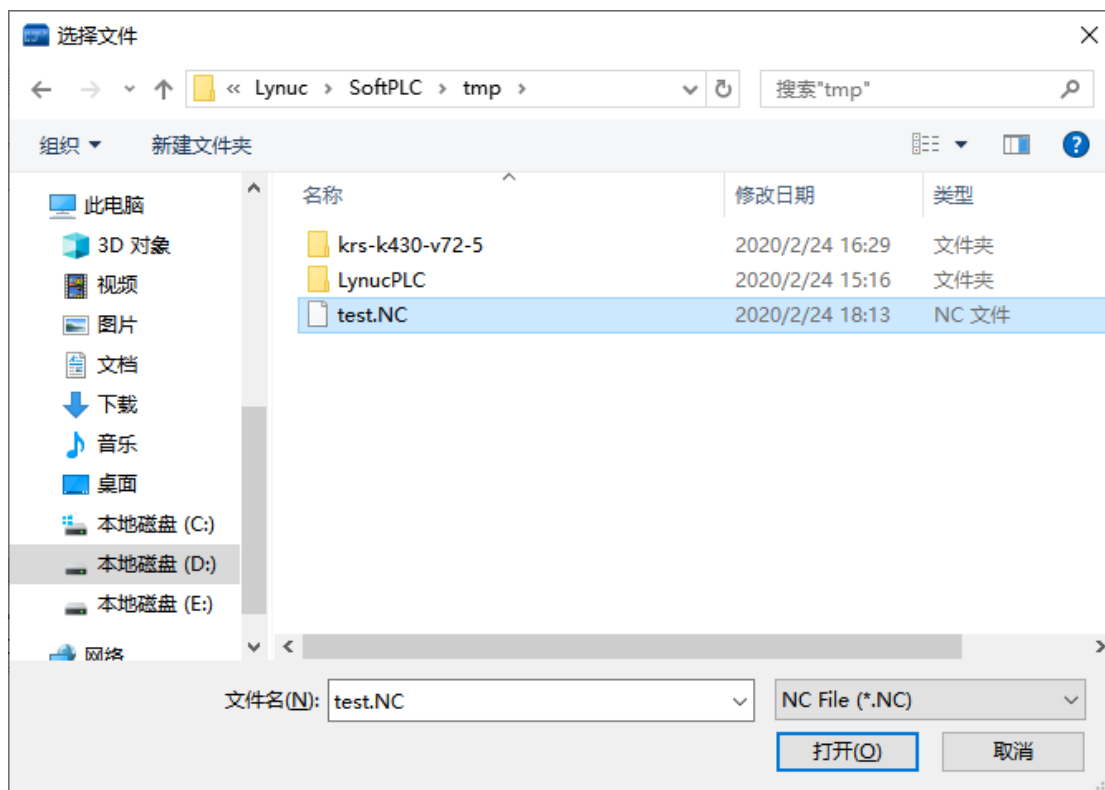


图 13-1-2 添加文件-2

※ 注意

很多配置文件是严格按文件名匹配的，而且这些文件在工程中只能有一个；只有 PLC 和 Motion 文件是根据后缀选择的，可以添加多个。

13.2 删除文件

操作步骤

- ①. 首先打开一个需要删除的文件或者在左边工程树中选中需要删除的文件名，点击菜单栏【工程】下的【删除文件】。

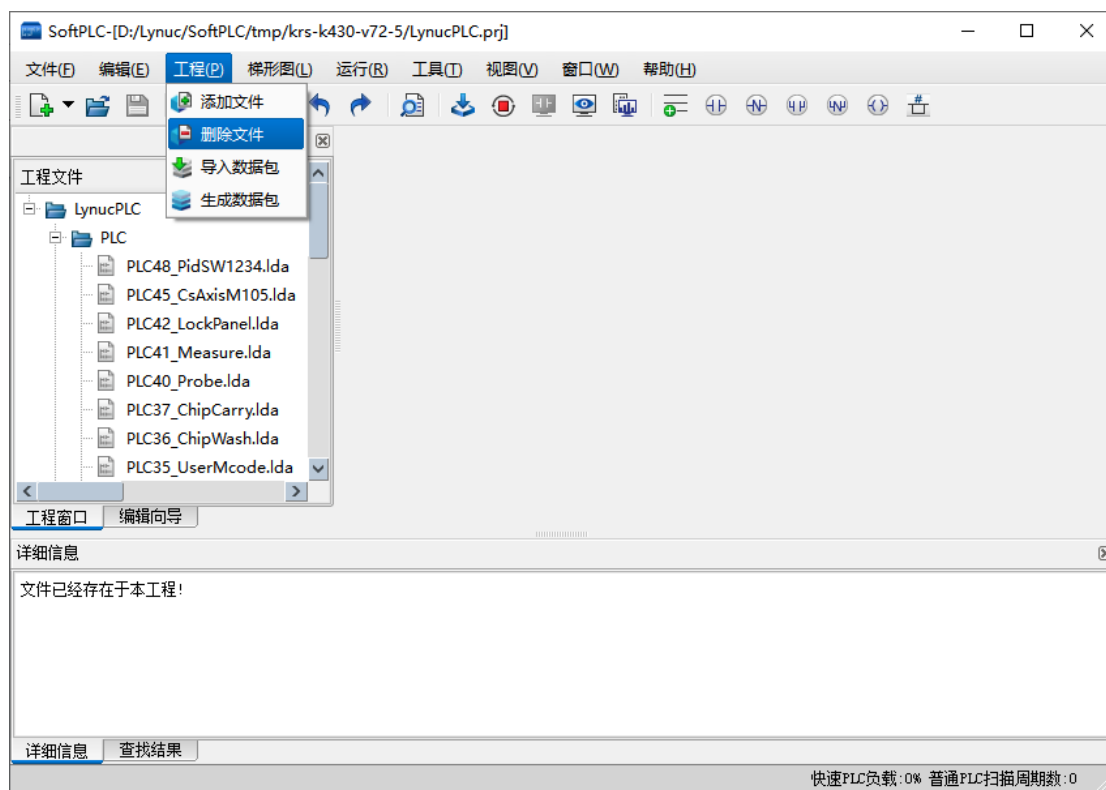


图 13-2-1 删除文件

②. 确认后出现界面如下:

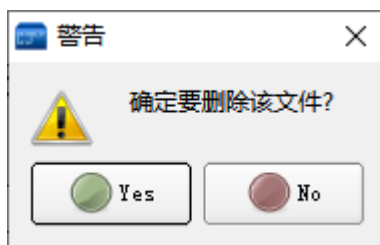


图 13-2-2 删除文件确认窗口

③. 点击确认键，该程序文件被删除。

13.3 导入数据包

操作步骤

①. 在菜单栏【工程】选项中点击【导入数据包】，弹出如下窗口：

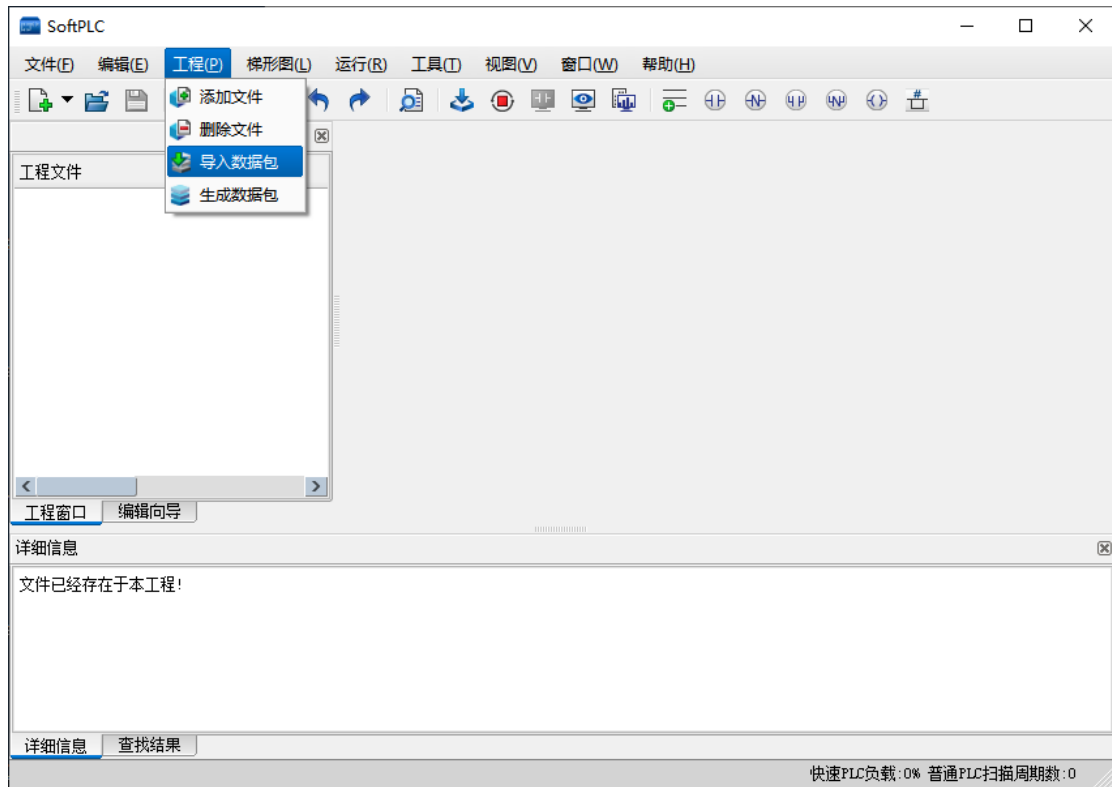


图 13-4-1 导入数据包

②. 选择对应的数据包文件，点击打开，然后选择工程存储路径。

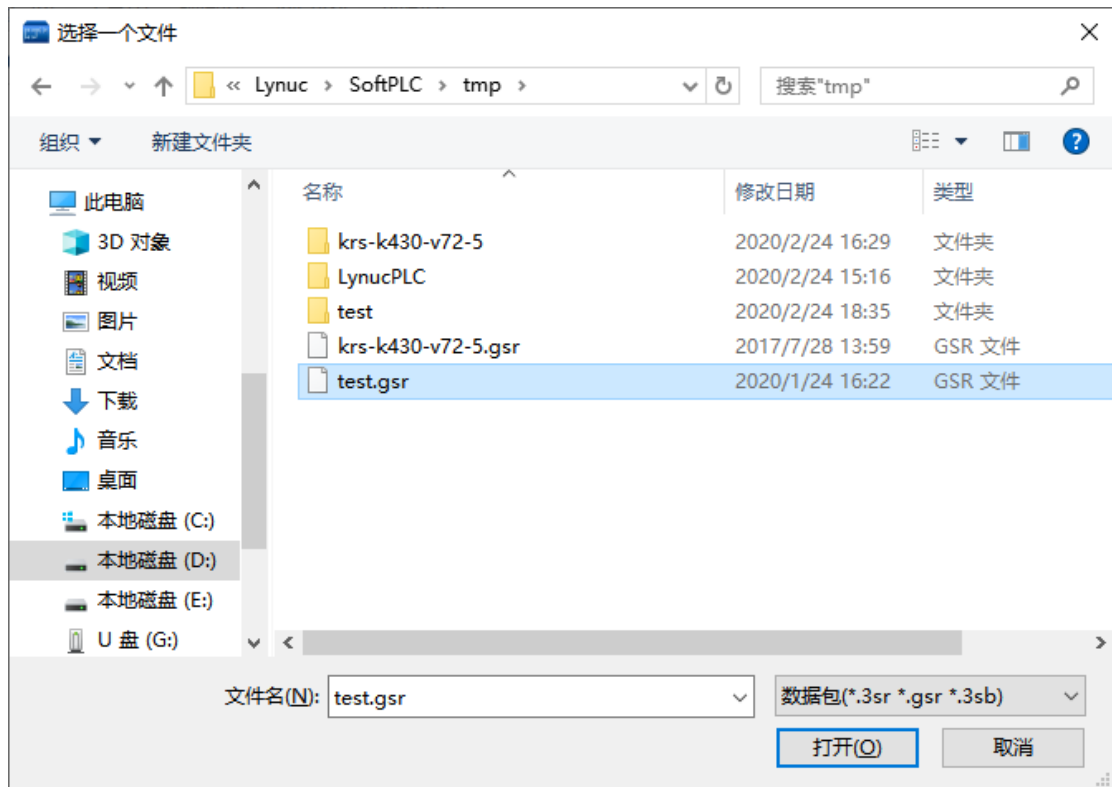


图 13-4-2 选择需要导入的数据包

- ③. 点击选择文件夹按钮后，弹出该工程的版本信息，如下图：



图 13-4-3 工程文件版本信息

- ④. 点击加载按钮，工程直接生成，左边的工程树显示工程中包含的所有文件。

13.4 生成数据包

操作步骤

- ①. 在菜单栏【工程】选项中点击【生成数据包】，弹出如下窗口：



生成数据包

第一步: 请输入相关信息

文件名: LynucPLC

Version:

Machine ID:

Fit Version:

Machine type:

描述:

Next Cancel

图 13-5-1 生成数据包-1

说明

Version 版本项是指配置文件的版本，仅对 1.8 以后的版本有效。该版本和下面的 Fit Version 无关。

Machine ID 指机器的 ID（可空白）。

Fit Version 是指在哪个 lynuc 软件版本上可用。

Description 描述了输入对数据包的说明，可以在 infopub 或 1.8 中预览。

Machine type 表示机床类型（可空白）。

②. 按【Next】选择需要打包的文件类型，这里 PLC 和 Motion 作为一个整体选择。



图 13-5-2 生成数据包-2

- ③. 若选择了 PLC、Motion 或 data 文件，则会出现类似的对话框需选择单个的文件或数据项。
- ④. 选择结束后点击 finish 生成数据包到软件安装目录下的 install 目录中。

14 特殊功能和错误处理

概述

本节介绍了 MO1023、MO1022 等特殊功能指令的定义说明和错误处理。

表 14-1 特殊功能指令定义说明和错误处理

指令		说明
MO1023	0-11	预留
	12	定义的功能模块不能被识别,产生的原因是接口和内核的版本不一致
	13	除零错误,在除法模块中,除数为零。下次执行除法时如果除数不为零则清除
	14	调用子程序失败,产生原因是 CALL 模块中指定的被调用 PLC 程序不可运行。下次调用程序成功则清除
	15	定义的 BUFF 溢出,产生于 MVOFF 模块,表示定义的 BUFFER 大小超过系统允许。下次调用成功时清除
	28	用于 MVTO 功能块(Move To 输出地址偏移),此错误位只针对 R 变量有效果。如果此位为 1,则表示偏移后的 R 变量为只读不可写 R 变量,写操作不执行。下次调用成功时清除
	29	全局 PLC 错误标记,若发生任何错误类型中的一种,则此位变 1。注意此位在标记为 1 后会一直保持,直到用户手动清零
	30	当偏移后的 Macro 类型同输出变量的类型不匹配时,此位为 1 代表类型错误。下次调用成功时清除
MO1022	31	Macro 变量在 Move offset 时,偏移后的 Macro 号不存在 1---代表发生错误。下次调用成功时清除
	0	内核产生的方波,频率为 100hz,占空比为 1
	1	内核产生的方波,频率为 10hz,占空比为 1
	2	内核产生的方波,频率为 1hz,占空比为 1
MO1000~MO1014	-	输出信号
MI1000~MI1014	-	输入信号

※ 注意

错误信息若不在执行完成后立即检查,则有可能会正在下次正常执行后被清零。第 12, 29 位除外。

15 PLC 接口详细

概述

本节介绍了 PLC 的系统信号定义、面板 PLC、主轴-PLC11、用户自定义-PLC13、PLC 执行 M 指令、PLC 外部设备信号等相关内容。详见以下内容。

15.1 PLC 通知系统的外部输入指令




指令定义


PLC => 系统信号定义：由 PLC 通知系统的外部输入命令。




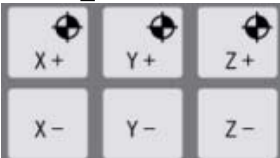
表 15-1-1 PLC 通知系统的外部输入指令定义

名称	功能	取值与用法	NickName
输入系统	路径 1 选择	1: 选中 0: 取消 系统默认选中路径 1	G_CHAN1
	路径 2 选择	1: 选中 0: 取消	G_CHAN2
		预留	
	数据保护	0: 取消保护 1: 打开保护	G_DATAPROTECT
	急停	0: 急停按下 1: 急停松开	G_EMGSTOP
	主电源关闭	0: 打开主电源 1: 关闭主电源 仅仅是断开三相强电，不是关机！	G_PWOFF
	机床上电	0: POWER 断开 1: POWER 闭合 一般连接主电源接触器的辅助触点，或者是伺服器的 ServoReady 信号	G_PWSTS
	系统关机	系统关机信号，常开 置 1 将关闭控制器	G_HALTSYS
	OT2	0: 硬限位保护 1: 取消硬限位保护 马达超程时使用	G_OT2
	自动模式手轮干预	G_TTFUNC=0: 上升沿有效； G_TTFUNC=1: 0: 禁止手轮干预	G_AUTOHW

	1: 允许手轮干预	
用户自定义手轮模块	0: 使用系统默认的手轮模块 1: 不使用系统默认的手轮, 支持任意的手轮接入方式, 需要编写用户 PLC	G_USRMPG
JOG 运动模式	0: Jog 1: INC/寸动 2: RAPID	G_JOGMODE
选择第 2 手轮	0: 默认第一手轮 1: 选择第二手轮	G_MPG2
选择下面接口的有效模式: G_AUTOHW G_SINGLE G_OPSTOP G_BLKSKIP G_DRYRUN G_MPGTRACE G_SINGLE2 G_OPSTOP2 G_BLKSKIP2 G_DRYRUN2 G_MPGTRACE2	0: 上升沿有效 1: 电平有效 默认是 0, 即上升沿有效	G_TTFUNC
选择系统提供的回零方法	仅当 G_USERHM=0 时有效 0: 使用 G181 指令归零 1: 使用系统的 PLC 指令归零 通过各轴的 J+接口调入该回零方法;	G_SELHM
编写用户 PLC 实现回零	0: 使用系统默认的回零方法, 具体的方法参见 G_SELHM 接口; 1: 禁用系统默认的回零方法, 此时 G181 功能是无法启用的。只能编写用户 PLC 实现回零;	G_USERHM
模块选择	1: POS/位置 9: OFFSET/偏置 4: EDIT/编辑 2: RUN/程序 6: SYSTEM/系统 7: INFO/信息 8: CUSTUM/用户	G_MODULE
F 键	F1~F10 对应 LCD 下方的 10 个功能键 F11 左侧向上 F12 左侧向下 F13 右侧向上 F14 右侧向下	G_FKEY
JOG 接口模式选择	0: X+, X-...	G_JOGSELECT

		 <p>这种面板能够支持多轴同时 Jog 移动； 1: XYZ, J+, J-</p>  <p>系统默认是 0；</p> 	
		预留	
	第 1 主轴	1: 反转 2: 停止 3: 正转 4: 定位	G_SELSP
	第 2 主轴		G_SELSP2
	第 3 主轴		G_SELSP3
	第 4 主轴		G_SELSP4
	第 5 主轴		G_SELSP5
	第 6 主轴		G_SELSP6
	第 7 主轴		G_SELSP7
	第 8 主轴		G_SELSP8
输入到路径 1	单步	G_TTFUNC=0: 上升沿有效； G_TTFUNC=1: 0: 单步无效 1: 单步有效	G_SINGLE
	任意停止	G_TTFUNC=0: 上升沿有效； G_TTFUNC=1: 0: M01 选停无效 1: M01 选停有效	G_OPSTOP
	选择跳跃	G_TTFUNC=0: 上升沿有效； G_TTFUNC=1: 0: “/”跳过无效 1: “/”跳过有效	G_BLKSKIP
	干加工	G_TTFUNC=0: 上升沿有效； G_TTFUNC=1: 0: 干加工无效 1: 干加工有效	G_DRYRUN

程序启动	CycleStart 信号，上升沿有效 只在 Mem、MDI、CodeLess 模式有效 程序运行后，产生 F_CYCLESTART 的状态	G_CYCLESTART
程序暂停	STOP 信号，上升沿有效 只在 CycleStart 运行状态时有效；可以通过 CycleStart 信号继续执行被暂停的程序 程序暂停后，产生 F_FEEDHOLD 状态	G_FEEDHOLD
程序停止	程序停止信号，上升沿有效 在 CycleStart 运行状态、程序暂停状态时有效； 停止后产生 F_PROGSTOP 状态，只能通过 G_RESET 清除	G_PROGSTOP
复位	系统复位信号	G_RESET
手轮模拟	G_TTFUNC=0: 上升沿有效； G_TTFUNC=1: 0: 手轮模拟无效 1: 手轮模拟有效	G_MPGTRACE
快速复位	复位系统，在任意时刻，都可以直接将系统复位到就绪状态	G_FRESET
W 轴控制有效	W 轴控制，上升沿有效	G_WAXIS
Rtcp 手轮	上升沿有效	G_RTCPHW
	预留	
模式选择	1: Codeless/无代码 2: MEM/自动 4: Handle/手轮 5: Jog 6: Home/原点复归 8: MDI	G_MODE
Jog/HM 轴选	1~15: XYZABCUVW..... 针对 G_JOGSELECT = 1 的模式 	G_AXIS
	预留	
J+	J+ 信号，必须先选择要移动的轴 针对 G_JOGSELECT = 1 的模式	G_JOGPLUS 通道一、二共用

			
J-	J- 信号, 必须先选择要移动的轴 针对 G_JOGSELECT = 1 的模式		G_JOGMINUS 通道一、二共用
Jog+;16 轴	J+ 信号 针对 G_JOGSELECT = 0 的模式		G_JMPLUS G_JXPLUS G_JYPLUS G_JZPLUS G_JAPLUS
Jog-;16 轴	J- 信号 针对 G_JOGSELECT = 0 的模式		G_JMMINUS G_JXMINUS G_JYMINUS G_JZMINUS G_JAMINUS
输入到路径 2	单步	参见路径一!	G_SINGLE2
	任意停止		G_OPSTOP2
	选择跳跃		G_BLKSKIP2
	干加工		G_DRYRUN2
	程序启动		G_CYCLESTART2
	程序暂停		G_FEEDHOLD2
	程序停止		G_PROGSTOP2
	复位		G_RESET2
	手轮模拟		G_MPGTRACE2
	快速复位		G_FRESET2
	手轮 RTCP		G_RTCPHW2
	模式选择		G_MODE2
	Jog/HM 轴选		G_AXIS2
J+	G_JOGPLUS2		
J-	G_JOGMINUS2		
	G_JMPLUS2		
Jog+;16 轴	G_JXPLUS2 G_JYPLUS2 G_JZPLUS2 G_JAPLUS2		
	G_JMMINUS2		
Jog-;16 轴	G_JXMINUS2 G_JYMINUS2		

			G_JZMINUS2 G_JAMINUS2
输入 到系 统	手轮轴选	0: 无轴选 1~15: XYZABCUVW.....	G_HWAXIS
	Jog 倍率	0~15 档, 每档对应的倍率值可通过系统-> 参数进行配置	G_JOVRD
	Jog 寸动倍率	0~15 档, 每档对应的倍率值可通过系统-> 参数进行配置	G_JINC_OVRD
	Rapid 倍率	0~15 档, 每档对应的倍率值可通过系统-> 参数进行配置	G_ROVRD
	Feed 倍率	0~31 档, 每档对应的倍率值可通过系统-> 参数进行配置	G_FOVRD
	主轴倍率	0~15 档, 每档对应的倍率值可通过系统-> 参数进行配置	G_SOVRD
	手轮倍率	0~7 档, 每档对应的倍率值可通过系统-> 参数进行配置	G_HWOVRD
		预留	
	路径 2 手轮轴选	0: 无轴选 1~15: XYZABCUVW.....	G_HWAXIS2
	路径 2 手轮倍率	0~7 档, 每档对应的倍率值可通过系统-> 参数进行配置	G_HWOVRD2
		预留	
	禁止程序启动		DIS_CYCLESTART
	禁止 Jog 移动		DIS_JOG
	禁止手轮移动		DIS_HANDLE
	预留		

15.2 PLC 得到系统的内部输出指令

指令定义

系统 => PLC 信号定义: 由 CNC 返回 PLC 的状态信息

表 15-2-1 PLC 得到系统的内部输出指令定义

名称	功能	取值与用法	NickName
系统 返回 状态	路径 1 选择	1: 路径 1 被选中 0: 路径 1 未选中	F_CHAN1
	通道 2 选择	1: 路径 2 被选中 0: 路径 2 未选中	F_CHAN2
		预留	
	数据保护	0: 数据保护解除 1: 数据保护有效	F_DATAPROTECT
	急停	0: 急停松开 1: 急停	F_EMGSTOP
	主电源关闭	0: 允许上主电源 1: 断开主电源	F_PWOFF

		一般后接主电源接触器的线圈输出	
	机床上电	0: 机床主电源未吸合 1: 机床主电源已吸合 当 F_PWSTS=1 时, 马达闭环 当 F_PWSTS=0 时, 马达自由	F_PWSTS
	系统关机	关机信号!后接关机继电器的输出。 仅当 F_PWSTS=0 后, 才会输出本信号	F_HALTSYS
	OT2	0: 未忽略硬限位 1: 已忽略硬限位	F_OT2
	自动模式手轮使能	0: 未使能 1: 已使能	F_AUTOHW
	CI 就绪状态	系统启动完成, 已正常工作	F_SYSREADY
	JOG 移动模式	0: Jog 1: INC/寸动 2: RAPID	F_JOGMODE
	第 2 手轮选择	0: 选中第 1 手轮 1: 选中第 2 手轮	F_MPG2
		预留	
	模块选择	1: POS/位置 9: OFFSET/偏置 4: EDIT/编辑 2: RUN/程序 6: SYSTEM/系统 7: INFO/信息 8: CUSTUM/用户	F_MODULE
		预留	
		预留	
路径 1 的返回状态	单步	0: 单步无效 1: 单步有效	F_SINGLE
	任意停止	0: M01 停止无效 1: M01 停止有效	F_OPSTOP
	选择跳跃	0: “/”跳过无效 1: “/”跳过有效	F_BLKSKIP
	干加工	0: 干加工无效 1: 干加工有效	F_DRYRUN
	程序启动	程序启动状态, 正处于加工中, 可以一直保持, 直到加工被停止, 或者结束。	F_CYCLESTART
	程序暂停	程序暂停状态, 处于暂停中, 可以一直保持, 直到重新开始加工, 或停止。	F_PROGHOLD
	程序停止	程序停止状态, 程序已被停止。可以一直保持, 直到系统复位。	F_PROGSTOP
	复位	系统复位中, 脉冲信号, 每次复位产生一次。 G RESET/G FRESET 都会产	F_RESET

		生同样的该脉冲信号。	
手轮模拟		0: 手轮模拟无效 1: 手轮模拟有效	F_MPGTRACE
快速复位		同 F_RESET	F_FRESET
W 轴控制有效		W 轴控制有效	F_WAXIS
手轮 rtcw			F_RTCPHW
		预留	
模式选择		1: Codeless/无代码 2: MEM/自动 4: Handle/手轮 5: Jog 6: Home/原点复归 8: MDI	F_MODE
Jog/HM 轴选		1~15: XYZABCUVW.....	F_AXIS
		预留	
运行状态		系统运行中,包括:程序加工、偏置登陆、手轮、Jog、原点复归	F_RUNNING
暂停状态		只针对程序加工的程序一时停止	F_HOLDING
停止状态		只针对 MEM/MDI 模式下的程序停止	F_STOPPED
就绪状态		系统就绪时,可以进行机床移动操作	F_READY
CodeLess 模式			F_CODELESS
Jog 模式			F_JOG
Handle 模式			F_HANDLE
原点复归模式		REF	F_HOME
MDI 模式			F_MDI
MEM 模式			F_MEM
M30		MEM、MDI 模式下,系统-> PLC 的 M30 信号	F_M30
M02		MEM、MDI 模式下,系统-> PLC 的 M02 信号	F_M02
通道 2 输出	单步	参见路径 1。	F_SINGLE2
	任意停止		F_OPSTOP2
	选择跳跃		F_BLKSKIP2
	干加工		F_DRYRUN2
	程序启动		F_CYCLESTART2
	程序暂停		F_PROGHOLD2
	程序停止		F_PROGSTOP2
	复位		F_RESET2

	手轮模拟		F_MPGTRACE2
	快速复位		F_FRESET2
	手轮 rtcp		F_RTCPHW2
	模式选择		F_MODE2
	Jog/HM 轴选		F_AXIS2
	运行状态		F_RUNNING2
	暂停状态		F_HOLDING2
	停止状态		F_STOPPED2
	就绪状态		F_READY2
	CodeLess 模式		F_CODELESS2
	Jog 模式		F_JOG2
	Handle 模式		F_HANDLE2
	原点复归模式		F_HOME2
	MDI 模式		F_MDI2
	MEM 模式		F_MEM2
	M30		F_M30_2
	M02		F_M02_2
系统 输出	手脉轴选	0: 空轴选 1~15: XYZABCUVW.....	F_HWAXIS
	Jog 倍率		F_JOVRD
	Jog 寸动倍率		F_JINC_OVRD
	Rapid 倍率		F_ROVRD
	Feed 倍率		F_FOVRD
	主轴倍率		F_SOVRD
	手轮倍率		F_HWOVRD
		预留	
	路径 2 手轮轴选	0: 空轴选 1~15: XYZABCUVW.....	F_HWAXIS2
	路径 2 手轮倍率		F_HWOVRD2
		预留	
	内部 CycleStart	*****	INNER_CYCLESTART
	位置模块	POS	F_POS
	编辑模块	EDIT	F_EDIT
	偏置模块	OFFSET	F_OFFSET
	程序模块	RUN	F_PROG
	系统模块	SYSTEM	F_SYSTEM
	信息模块	INFO	F_INFOMATION
	用户模块	CUSTOM	F_USERS
	内部启动 2	*****	INNER_CYCLESTART2
	系统严重错误		F_FATALERR
	系统停止错误		F_STOPERR
	系统错误	包括严重错误、停止错误	F_CIERR
	系统警告		F_WARN

程序停止	程序加工时被异常停止	F_STOP_PROG
蜂鸣器		F_BUZZER
	预留	
第 1 主轴状态	主轴状态： 1: 反转 2: 停止 3: 正转 4: 定位	F_SELSP
第 2 主轴状态		F_SELSP2
第 3 主轴状态		F_SELSP3
第 4 主轴状态		F_SELSP4
第 5 主轴状态		F_SELSP5
第 6 主轴状态		F_SELSP6
第 7 主轴状态		F_SELSP7
第 8 主轴状态		F_SELSP8

15.3 面板 PLC

概述

针对用户自定义的面板，只需要将 X 输入对应到 G 变量、F 变量对应到 Y 输出即可。

例：X 输入对应到 G 变量。

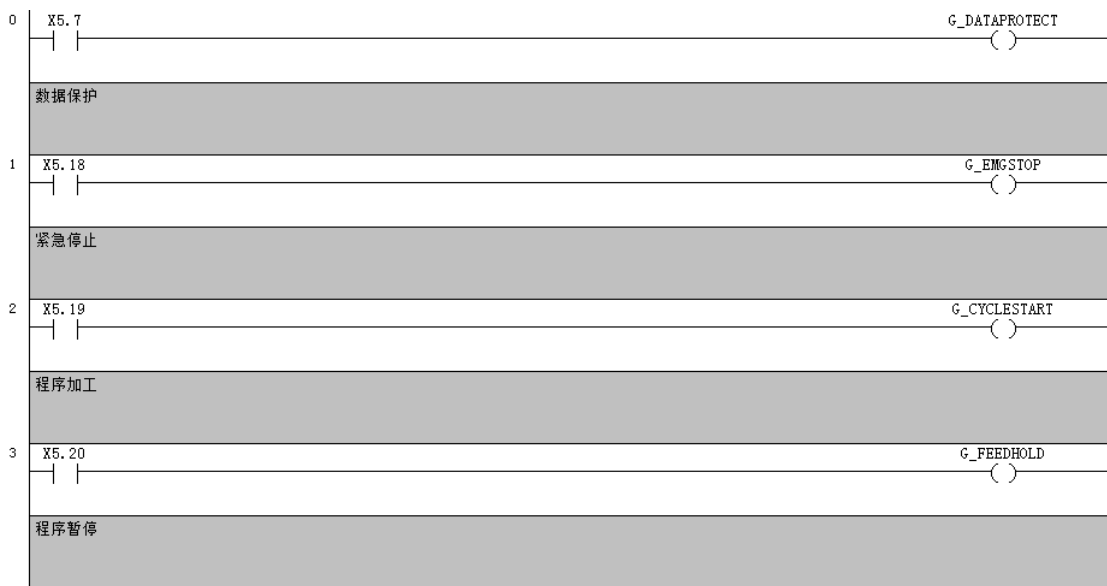


图 15-3-1 X 输入对应到 G 变量

例：F 变量对应到 Y 输出。

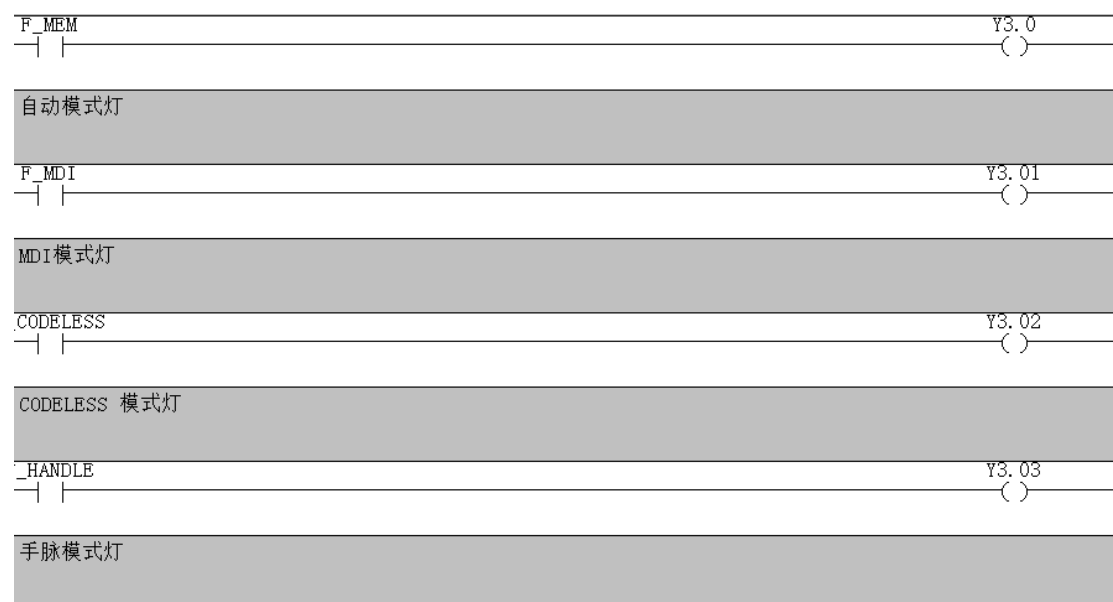


图 15-3-2 F 变量对应到 Y 输出

15.4 主轴-PLC11

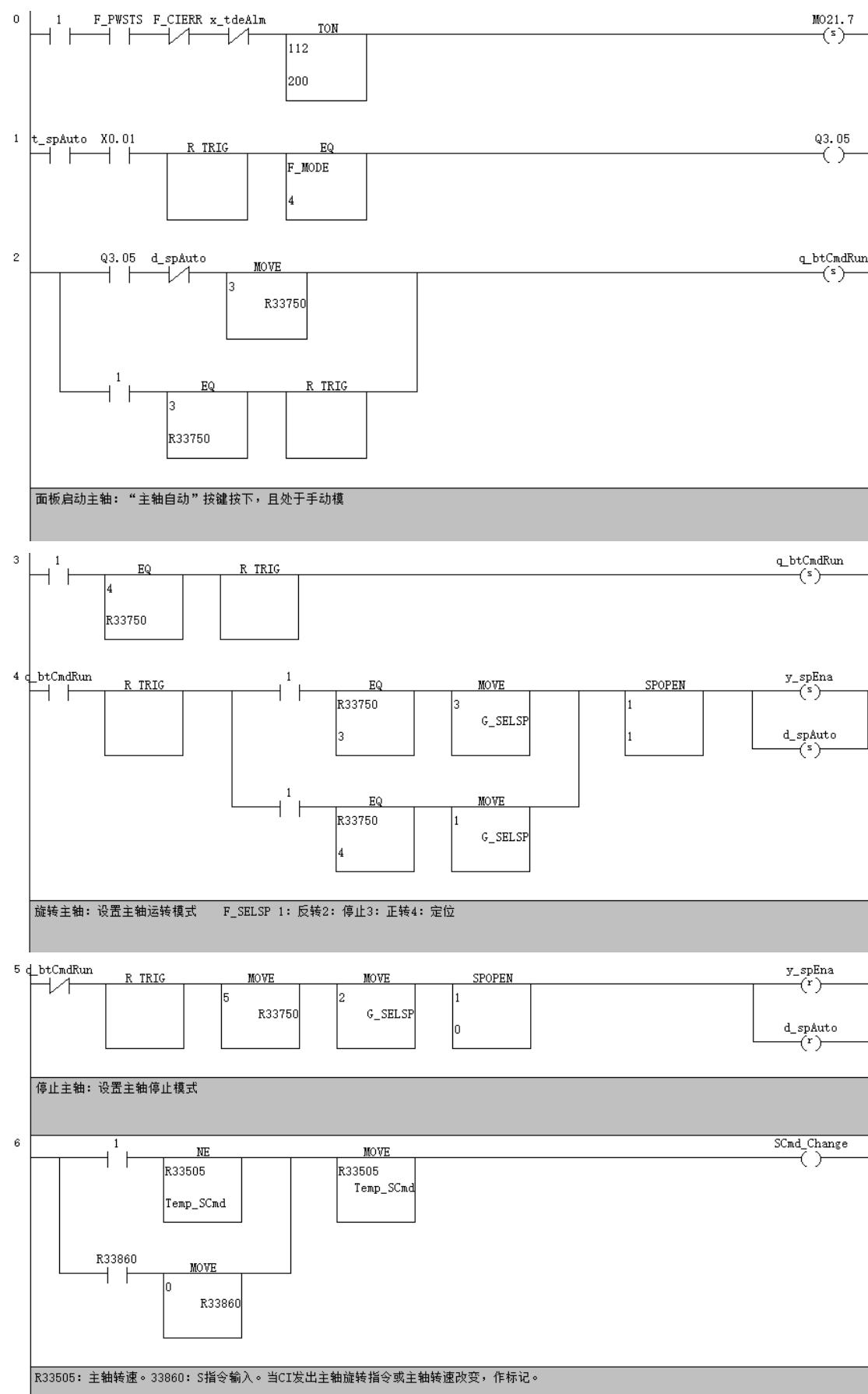
概述

PLC11 可以实现主轴的起停、正反转、设置转速等功能，并记录出错、复位、停止加工时复位输出及状态记录。

变量定义表

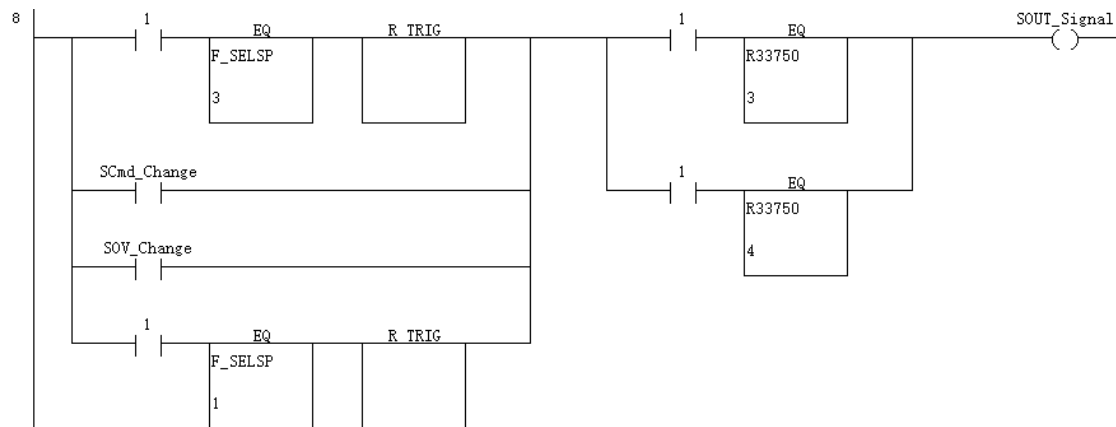
	变量名称 /	别名	初值	常闭	注释
1	MO210.28	t_spAuto		<input type="checkbox"/>	面板“主轴自动”按键
2	MO212.28	d_spAuto		<input type="checkbox"/>	面板“主轴自动”按键指示灯
3	Q0.03	q_btCmdRun		<input type="checkbox"/>	记录“主轴自动”按钮的命令 1：启动运动 0：停止运动
4	Q0.10	SCmd_Change		<input type="checkbox"/>	s code change flag
5	Q0.11	SOUT_Signal		<input type="checkbox"/>	S Cmd Output Signal
6	Q0.12	SOV_Change		<input type="checkbox"/>	
7	Q0.14	q_spWait	0	<input type="checkbox"/>	
8	QD10	q_spVel	0	<input type="checkbox"/>	
9	QD12	q_actVel		<input type="checkbox"/>	
10	QF1	Temp_SCmd		<input type="checkbox"/>	bak s for check s code change
11	QF2	TempF_SPOV		<input type="checkbox"/>	
12	X0.24	x_tdeAlm		<input type="checkbox"/>	TDE变频器报警检出
13	Y0.05	y_spEna		<input type="checkbox"/>	主轴使能
14	QF3	Temp_SVal		<input type="checkbox"/>	
15	QD4	TempD_SCmd		<input type="checkbox"/>	

梯形图





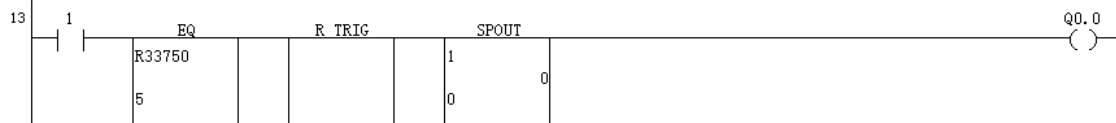
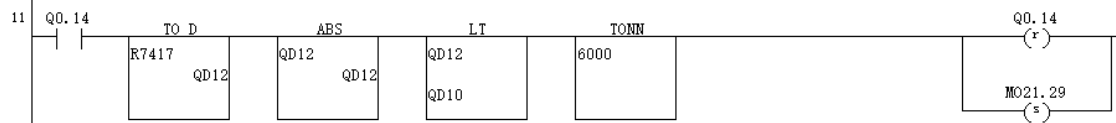
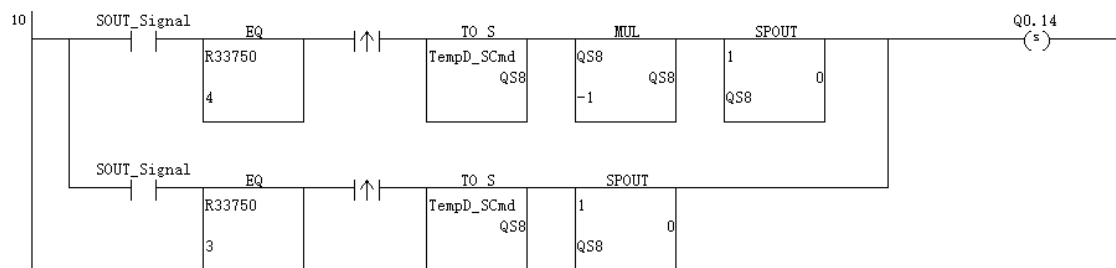
R33518: 主轴倍率。主轴倍率改变标记



当主轴的转速，倍率，启停状态被改变，并且是M03模式或主轴旋转状态，设置输出标记



R33505: 主轴指令转速。如果输出标记为1，计算出要输出的转速



当M指令模式不是M03，主轴状态非旋转时，输出转速0到SI

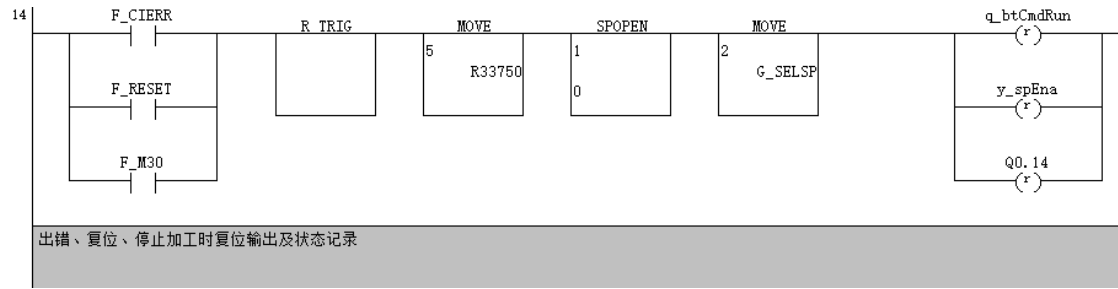


图 15-4-1 主轴-PLC11

15.5 用户自定义-PLC13

举例说明

- 以工作液冷却机为例
 1. I/O 点
 - 水泵供电 Y0.10
 - 水泵过载报警 X0.11
 - 面板按钮-启停工作液 X3.4
 - 中间变量-工作液冷却状态 Q0.2
 - 面板指示灯-工作液是否已开启 Y3.8
 2. NC 指令
 - M08 启动工作液
 - M09 停止工作液
 3. 报错信息
 - 水泵过载报警信息
 4. 异常的处理
 - M30F_M30
 - M02F_M02
 - 加工被异常停止 F_STOP_PROG
 - 急停 F_EMGSTOP
 - 复位 F_RESET
- 面板按钮、M08/M09 的工作液冷却功能实现：

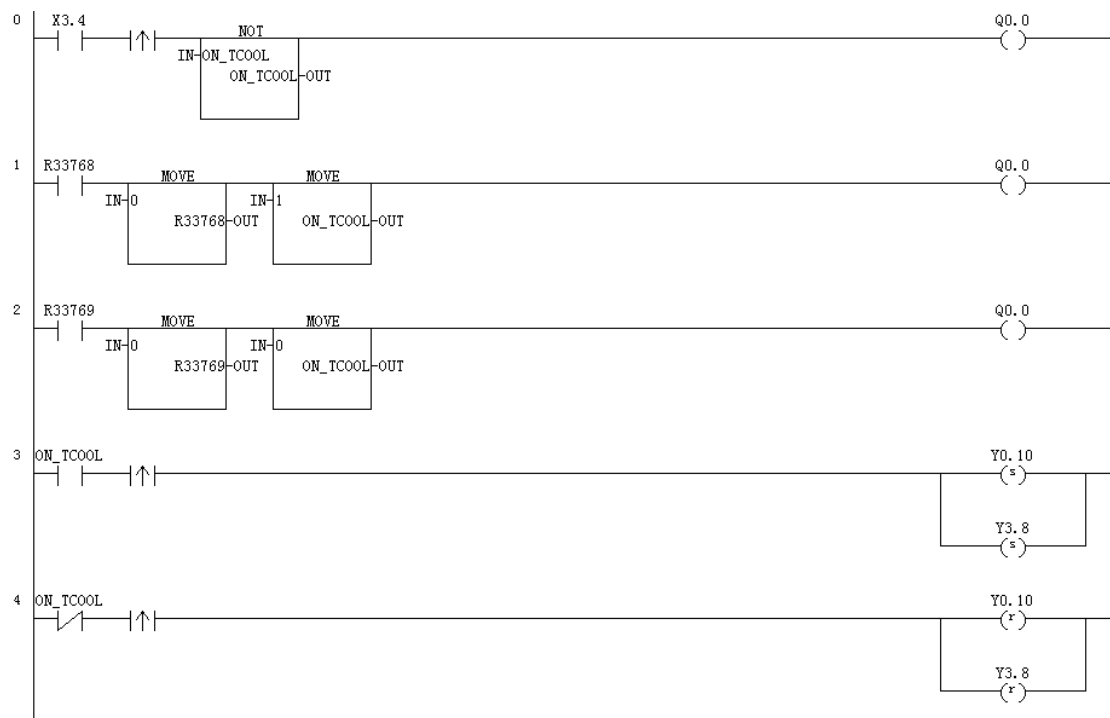


图 15-5-1 面板按钮、M08/M09 的工作液冷却功能实现

● 定义外部错误，取 MO21.1

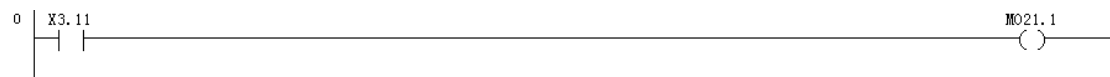


图 15-5-2 定义外部错误，取 MO21.1

定义错误信息，MO21.1 对应的系统错误编号 1034

1034=工作液冷却泵过载。

● 异常处理



图 15-5-3 异常处理

15.6 PLC 执行 M 指令

概述

本系统提供了“M 指令等待完成信号”的设定 (ON/OFF)。下面列举了实现 M 指令的三种方式，其中第三种方式可以通过 PLC 独立执行 M 指令并等待完成信号，第一种和第二种必须依赖 M 宏程序才能实现等待完成信号的功能。

举例说明

①. PLC 执行 M 指令方式 1

应用范围 M00~M99，对应的 MACRO 变量#33760~#33859。

特点：

只能执行编号为 0~99 的 M 指令，如要等待完成信号，则必须同时添加相应的 M 宏程序。

※ 注意

如果在一个 PLC 周期内指定了多个相同 M 指令，则只会执行一次该 M 指令。

例 M08，实现如下：



图 15-6-1 执行 M 指令方式 1

②. PLC 执行 M 指令方式 2

应用范围 M00~M9999，对应的 MACRO 变量#33862 (M 指令执行信号)、#33863 (M 指令编号)

特点

能执行编号为 0~9999 的 M 指令，如要等待完成信号，则必须同时添加相应的 M 宏程序。

※ 注意：

同行最多只能输入 1 个 M 指令，第二个以后的 M 指令 (包括第二个) 会被系统自动忽略。一个 PLC 周期内只能执行 1 个 M 指令。

例 M103，实现如下：



图 15-6-2 执行 M 指令方式 2

③. PLC 执行 M 指令方式 3

应用范围 M00~M9999, PLC 组件: MCODE、MFINISH。

特点:

当系统设定“M 指令等待完成信号”为 OFF 时, 实现情况与前面两种相同, 但可以同时执行 3 个 M 指令。

例如 M08, 实现如下: MCODE 组件(IN1: 要执行的 M 指令编号 IN2:所在通道号 1~2)。



图 15-6-3 执行 M 指令方式 3-1

当系统设定“M 指令等待完成信号”为 ON 时, 系统会先停止加工, 等待 PLC 执行 M 指令, 直到 M 指令执行完后, 继续加工。PLC 最多只能同时执行 3 个 M 指令, 即每行 NC 最多只能同时指定 3 个 M 指令。

例 M03, 实现如下: MFINISH 组件(IN1:所在通道号 1~2), Q8.0 为约定好的局部变量, FINISH 为宏程序结束后的标记变量。

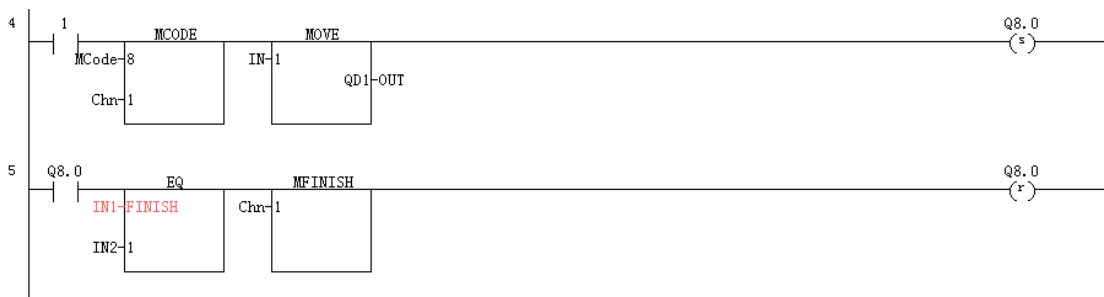


图 15-6-4 执行 M 指令方式 3-2

※ 注意:

MCODE 对一个 M 指令仅触发一次, 一旦导通后, 下一次 PLC 周期将会自动断开。MFINISH 属于上跳沿组件, 如果 M 指令完成信号未满足或一直保持时, MFINISH 不被导通。

15.7 其他 PLC

概述

用户可以自行增加 PLC10 以上的 PLC，完成特殊功能，例如 ATC 逻辑，安全门锁等等。

Lynuc

上海铼钠克数控科技有限公司

地址：中国上海市闵行区都会路 2338 弄 30-31 号

邮编：201108

电话：+86 21 61837766

传真：+86 21 60720487

网址：<http://www.lynuc.cn>