



 Version 1.3.1

Safety FA-AutoThink

用户手册





Safety FA-AutoThink

用户手册

V1.4 版

2021 年 01 月

版权声明

本手册内容，包括文字、图表、标志、标识、商标、产品型号、软件程序、版面设计及其它内容等，均受《中华人民共和国著作权法》、《中华人民共和国商标法》、《中华人民共和国专利法》及与之适用的国际公约中有关著作权、商标权、专利权或其他财产所有权法律的保护，为北京和利时智能技术有限公司专属所有或持有。

由于本手册中所描述的设备有多种使用方法，用户以及设备使用责任人必须保证每种方法的可容许性。对由使用或错误使用这些设备造成的任何直接或间接损失，北京和利时智能技术有限公司将不负法律责任。

由于实际应用时的不确定因素，北京和利时智能技术有限公司不承担直接使用本手册中提供的数据的责任。

本手册仅供商业用户阅读，在未得到北京和利时智能技术有限公司书面授权的情况下，无论出于何种目的和原因，不得以任何形式（包括电子、机械或其它形式）传播或复制本手册的任何内容。违者我公司将依法追究其相关责任。

已核对本手册中的内容、图表与所述硬件设备相符，但误差难以避免，并不能保证完全一致。同时，会定期对手册的内容、图表进行检查、修改和维护，恕不另行通知。

HollySys、和利时、 **HollySys** 的字样和徽标均为北京和利时智能技术有限公司的商标或注册商标。

手册中涉及到的其他商标或注册商标属于它们各自的拥有者。

北京和利时智能技术有限公司版权所有。

地址：北京经济技术开发区地盛中路 2 号院

邮编：100176

商务电话：010-5898 1588

产品咨询热线：4008-111-999

技术支持电话：010-58981514

技术支持手机：13611160213

传真：010-5898 1558

网址：<http://www.hollysys.com/>

Email：PLC@hollysys.com

新浪微博：<http://weibo.com/hollysysplc>

目录

第 1 章	关于手册	1
1.1	文档更新	1
1.2	文档用途	1
1.3	阅读对象	1
1.4	使用约定	2
1.4.1	菜单命令	2
1.4.2	鼠标指令	2
1.4.3	键盘指令	2
1.4.4	在线帮助	2
1.4.5	重要信息	2
1.5	产品文档目录	3
第 2 章	概述	5
第 3 章	安装卸载	7
3.1	安装软件	7
3.1.1	安装环境	7
3.1.2	安装步骤	7
3.2	卸载软件	13
第 4 章	软件启动	15
第 5 章	软件界面	17
5.1	标题栏	17
5.2	菜单栏	17
5.3	工具栏	18
5.4	工程管理	18
5.4.1	任务配置	18
5.4.2	用户程序	19
5.4.3	硬件配置	19
5.4.4	全局变量	19
5.4.5	Modbus 组态	19
5.4.6	监视列表	19
5.5	工作区域	19
5.6	信息窗口	20
5.7	状态栏	21

5.8	库管理器	21
5.8.1	库配置.....	21
5.8.2	查看.....	25
5.8.3	查找.....	25
5.8.4	库使用.....	28
5.9	设备库	28
第 6 章	工程管理	31
6.1	新建工程	31
6.1.1	工程.....	31
6.1.2	库.....	32
6.2	任务配置	32
6.2.1	程序调用.....	32
6.2.2	任务属性配置.....	34
6.3	设置工程安全属性.....	34
6.3.1	设置安全属性.....	35
6.3.2	工程配置信息.....	35
6.4	工程安全	36
6.4.1	新建用户	37
6.4.2	编辑用户	39
6.4.3	删除用户	39
6.4.4	使用时机.....	39
6.5	库安全	40
6.6	工程设置	40
6.6.1	配置.....	41
6.6.2	颜色.....	41
6.6.3	组态语言	42
6.6.4	重置当前页面.....	44
6.7	保存工程	44
6.7.1	保存.....	44
6.7.2	另存为.....	44
6.8	关闭工程	44
6.9	最近工程列表	45
6.10	上传工程文件	45
6.11	下载工程文件	45
6.12	窗口	45
6.13	帮助.....	46
第 7 章	硬件配置	47
7.1	控制器模块.....	47
7.2	组态 DP 协议	47

7.2.1	添加 DP 主站设备	47
7.2.2	添加通讯协议	48
7.2.3	添加 PROFI-safe/Profibus-DP 从站	49
7.2.4	修改从站地址	51
7.2.5	从站参数配置	52
7.3	组态 Modbus TCP 协议	56
7.3.1	Modbus TCP 主站协议	56
7.3.2	Modbus TCP 从站协议	63
7.4	系统诊断	67
7.5	硬件列表	70
第 8 章	程序组织单元	73
8.1	添加 POU	73
8.1.1	公共部分	74
8.1.2	差异部分	74
8.2	编辑 POU	75
8.2.1	变量定义	76
8.2.2	逻辑实现	76
8.3	POU 属性	76
8.4	POU 重命名	77
8.5	复制/粘贴 POU	77
8.6	删除 POU	78
8.7	缩放 POU	78
8.8	打印 POU	79
8.8.1	打印预览	79
8.8.2	打印	81
8.9	查找	81
8.10	替换	82
8.11	POU 调用	82
8.11.1	用户程序调用功能块	83
8.11.2	用户程序调用函数	83
8.11.3	程序间的调用	83
8.11.4	功能块间的调用	84
8.11.5	查看调用树	85
8.12	文件夹	85
8.12.1	添加文件夹	85
8.12.2	删除文件夹	86
8.12.3	文件夹重命名	86
第 9 章	变量	87
9.1	变量类型	87

9.1.1	从结构形式划分.....	87
9.1.2	按照变量的作用域划分.....	87
9.2	变量命名	87
9.3	变量声明	88
9.4	基本数据类型	89
9.5	复杂数据类型	90
9.5.1	数组(ARRAY)	90
9.5.2	功能块类型.....	93
9.6	变量监视	94
9.7	变量的其他操作	96
9.7.1	增加变量.....	96
9.7.2	插入变量.....	96
9.7.3	复制、粘贴.....	96
9.8	用户自定义变量组.....	97
9.8.1	新建.....	97
9.8.2	重命名.....	97
9.8.3	删除变量组.....	98
9.9	变量导入导出	98
9.9.1	导出全局变量.....	98
9.9.2	导入全局变量.....	98
9.9.3	导出硬件变量.....	99
9.9.4	导入硬件变量.....	100
9.10	变量组管理.....	100
9.10.1	新建文件夹.....	100
9.10.2	删除文件夹.....	101
9.10.3	重命名.....	101
9.11	变量的访问.....	101
9.12	未使用全局点回收.....	102
9.13	交叉引用表.....	103
9.14	变量调试回读	103
第 10 章	编程语言	105
10.1	FBD 编辑器	105
10.1.1	光标位置.....	106
10.1.2	网络线.....	107
10.1.3	输入元件.....	108
10.1.4	输出元件.....	108
10.1.5	块元件.....	109
10.1.6	跳转元件和标号元件	109
10.1.7	返回元件.....	111
10.1.8	注释元件.....	111
10.1.9	置位/复位	111

10.1.10 置反.....	112
10.1.11 元件对齐.....	113
10.1.12 执行顺序和编号.....	114
10.2 LD 编辑器	115
10.2.1 光标位置.....	116
10.2.2 LD 元素	119
10.2.3 节.....	120
10.2.4 串联触点.....	121
10.2.5 并联触点.....	122
10.2.6 线圈.....	122
10.2.7 块元件.....	123
10.2.8 置位/复位	124
10.2.9 置反.....	124
10.2.10 注释.....	124
10.2.11 跳转和返回	126
10.2.12 元件的移动.....	128
第 11 章 控制器操作.....	131
11.1 控制器操作.....	131
11.1.1 连接控制器.....	131
11.1.2 控制器信息.....	132
11.1.3 控制器锁.....	133
11.1.4 修改控制器 IP.....	135
11.1.5 配置路由.....	136
11.1.6 工程升级.....	138
11.1.7 固件升级.....	139
11.1.8 日志读取.....	140
11.1.9 扫描 IP.....	141
11.2 复位.....	144
11.3 冷复位	144
11.4 热复位	144
11.5 清空控制器.....	144
11.6 控制器校时.....	145
第 12 章 编译.....	147
12.1 增量编译	147
12.2 全编译	147
12.3 编译结果	148
12.4 编译问题分析	148
12.5 反编译	151
第 13 章 下装.....	155

13.1 通讯设置	155
13.2 下装操作	155
第 14 章 调试.....	157
14.1 仿真模式	157
14.2 在线操作	157
14.2.1 写变量.....	157
第 15 章 在线监视	161
15.1 运行	161
15.2 停止.....	161
15.3 暂停.....	161
15.4 监视.....	161
15.4.1 变量与程序监视.....	162
15.4.2 通道的在线值和质量位.....	162
15.5 查看操作日志	163
15.6 使能 SOE 功能.....	164
15.6.1 概述.....	164
15.6.2 要求.....	164
15.6.3 步骤.....	164
15.7 查看 SOE 信息.....	165
15.7.1 概述.....	165
15.7.2 要求.....	166
15.7.3 步骤.....	166
15.8 设置 NTP 校时	166
15.8.1 概述.....	166
15.8.2 要求.....	167
15.8.3 步骤.....	167

第1章 关于手册

1.1 文档更新

版本	日期	说明
V1.0	2017.08.30	创建
V1.1	2018.02.02	章节 6.3 设置工程安全属性 章节 7.2 组态 DP 协议，更新用户参数界面、DP_MASTER 界面 章节 7.4 系统诊断变量更新
V1.2	2018.11.30	章节 6.2 任务配置更新运行周期范围 新增 12.1 增量编译 6.6.1 配置增加硬件配置回读选项设置 13.2 下装操作增加硬件配置回读确认
V1.0	2019.06.13	版权信息更新
V1.1	2019.11.10	新增变量导入导出功能
V1.2	2020.01.17	新增以下功能： 1、控制器作 Modbus 主站时，可配置两个 IP 2、IP 地址自动检测功能 3、NTP 校时 4、全局变量 SOE 功能 5、新增 LK630S 模块 6、LK610S 更新滤波参数
V1.3	2020.08.14	软件版本升级
V1.4	2021.01.26	新增 LK611S/LK410/LK442/LK720 模块

1.2 文档用途

本文档详细介绍了 LKS 安全型控制系统组态软件（Safety FA-AutoThink）的菜单、功能、组态调试方法等功能。

1.3 阅读对象

LKS 系统的编程人员、操作人员以及系统维护人员。

使用本手册的人员应具备自动控制和 IEC 61131 编程的相关知识。

1.4 使用约定

1.4.1 菜单命令

本手册中菜单命令、菜单中的子命令、标签页名称用【】括注表示，如【文件】、【编辑】菜单。

1.4.2 鼠标指令

按下：按下鼠标左键一次并且不松开。

单击：按下鼠标左键一次并松开。

右击：按下鼠标右键一次并松开。

双击：快速按下鼠标左键两次并松开。

拖动：移动鼠标时按住鼠标左键不放。

1.4.3 键盘指令

本手册中涉及键盘上的按键时，统一用加粗表示，如 **Enter**。

1.4.4 在线帮助

在线帮助是对本手册的补充。目的是在使用该软件时提供详细的支持。

该帮助系统集成于软件中，通过以下两种方式打开：

- 菜单栏、设备库、库管理器、工程管理树的子节点按 **F1** 键准确定位到相关的帮助说明。
- 通过菜单栏中【帮助】—【内容】调用在线帮助，在【索引】、【搜索】标签页中输入关键字进行查找、定位。
 - 【索引】：遍历索引页面进行快速查找。
 - 【搜索】：全文搜索。

在线帮助中搜索关键字时，只检索页面当中包含该关键字的页面，对页面名称包含该关键字的页面不进行检索。

在线帮助中搜索关键字时，需要输入*关键字或关键字*来进行模糊搜索。如搜索 **Target** 指令，在搜索标签页中输入：***GET**，如搜索 **GetTickCnt**，在搜索标签页中输入：**GET***，即可搜索出相关的页面。

1.4.5 重要信息

文档中使用以下标识明确相应信息：



- 警告图标，标识该操作有造成软硬件设备故障或损坏的潜在威胁。



- 重要图标，标识需要理解的操作或功能的重要信息。



- 操作图标，标识该对象的打开或操作方法。

操作图标中的内容解读：

操作所发生的位置：鼠标指令+操作顺序。



- 菜单栏：单击【插入】—【输入元件】；
- 工具栏：；
- 快捷键：Ctrl+I；
- POU：右击空白区域，单击【输入元件】。

如上图中的第 1 条操作方法，在菜单栏处进行，依次单击【插入】菜单和子菜单项【输入元件】命令。

SEE ALSO

- 参考图标，标识理解该功能需要参考的页面或内容。

1.5 产品文档目录



Safety FA-AutoThink 用户手册



LKS 安全型控制系统指令手册



LKS 安全型控制系统产品手册

第2章 概述

LKS 是和利时公司 SIL2 级安全型控制系统。其中，Safety FA-AutoThink 是专门为 LKS 而自主研发的组态软件，属于和利时 AutoThink 软件家族。

Safety FA-AutoThink 具有以下特点：

- 采用国际标准 IEC61131-3 的编程架构，并建立丰富的应用指令集。
- 支持两种语言，梯形图（LD）、功能块图（FBD）。
- 支持中文、英文两种语系。
- 提供便利的查找/替换功能。
- 提供多层次密码及数据保护机制。
- 支持多种联机操作，如在线监视、运行、停止等。
- 支持仿真功能。

表 1 软件版本信息及部分功能使用限制

软件版本	部分功能使用限制条件	规避措施
Safety FA-AutoThink V1.0.0 / V1.0.1/V1.1.1/ V1.2.1/V1.2.2/ V1.3.1	打开的 POU 页面程序较多时，工程在线时，按住 Ctrl，滚动鼠标滚轴放大缩小页面，会导致通讯失败	在线监视状态下，当进行页面放大缩小操作时，请勿按住 Ctrl 键并长时间连续滚动鼠标滚轮；此种操作会引起退出监视
	硬件配置组态 DP 或 Modbus TCP，请不要使用替换功能	手动修改通道名或通道说明
	硬件配置组态 LK610S，全局变量组 Q_Profis_DPIO_Group 中生成中间变量。不支持将中间变量右键发送到【Modbus 组态】的操作方式	通过复制中间变量后引用成员变量的方式
	Safety FA-AutoThink V1.0.0 / V1.0.1 版本，全局变量和 PRG 中变量不支持导入、导出	通过复制粘贴变量的方式
	Safety FA-AutoThink V1.0.0 版本，【Modbus 组态】中的变量，工程监视状态下，不显示在线值	Modbus 中的变量，在线时可在变量定义的地方查看在线值，或者在监视列表中查看
	程序没有除零保护	用户组态时注意，且编译时给出警告提示
	程序没有数组越界保护	用户组态时注意，且编译时给出警告提示

表 2 软件工具列表

工具	版本
Safety FA-AutoThink	V1.0.0、V1.0.1、V1.1.1、V1.2.1、V1.2.2、V1.3.1
控制器操作	V1.0

第3章 安装卸载

3.1 安装软件

用户在组态之前，必须安装 Safety FA-AutoThink 软件。本节介绍该软件的安装环境及步骤。

3.1.1 安装环境

计算机配置如表 3 所示。

表 3 推荐使用的计算机配置

环境	类型	型号
硬件环境	显示器	彩色 CRT 或液晶屏
	输入输出	标准键盘、鼠标
	USB 接口	至少 1 个 USB 2.0 接口
	显卡	分辨率支持 1280×720
	CPU	Intel Pentium 2.4 GHz 以上
	内存	512 MB 以上
	硬盘	10 GB 以上
软件环境	操作系统	Windows 7 专业版 32 位 Windows 7 专业版 64 位 Windows 10 专业版 64 位
	工具软件	MS Excel 2003, 2007, 2010

工程中使用的计算机配置应与上述配置相当，或高于上述配置。

3.1.2 安装步骤

软件主要安装步骤：

第1步 选择安装语言

将本软件安装光盘插入光盘驱动器，弹出“选择安装语言”对话框，如图 1 所示。

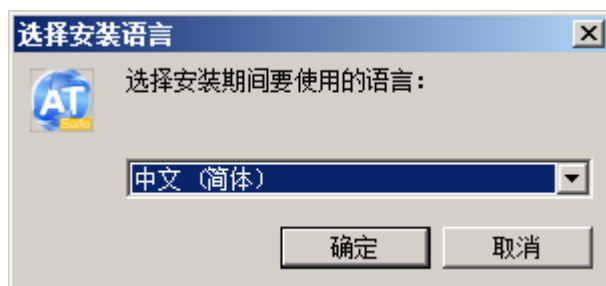


图 1 安装语言选择

第2步 启动安装向导

语言选择为中文（简体），单击确定。



图 2 安装向导

第3步 许可协议

单击下一步(N)，如图 3 所示。选择我接受协议(A)，然后单击下一步(N)。



图 3 许可协议

第4步 选择安装路径

如图 4 所示，通过单击**浏览(R)...**选择安装路径，单击**下一步(N)**。

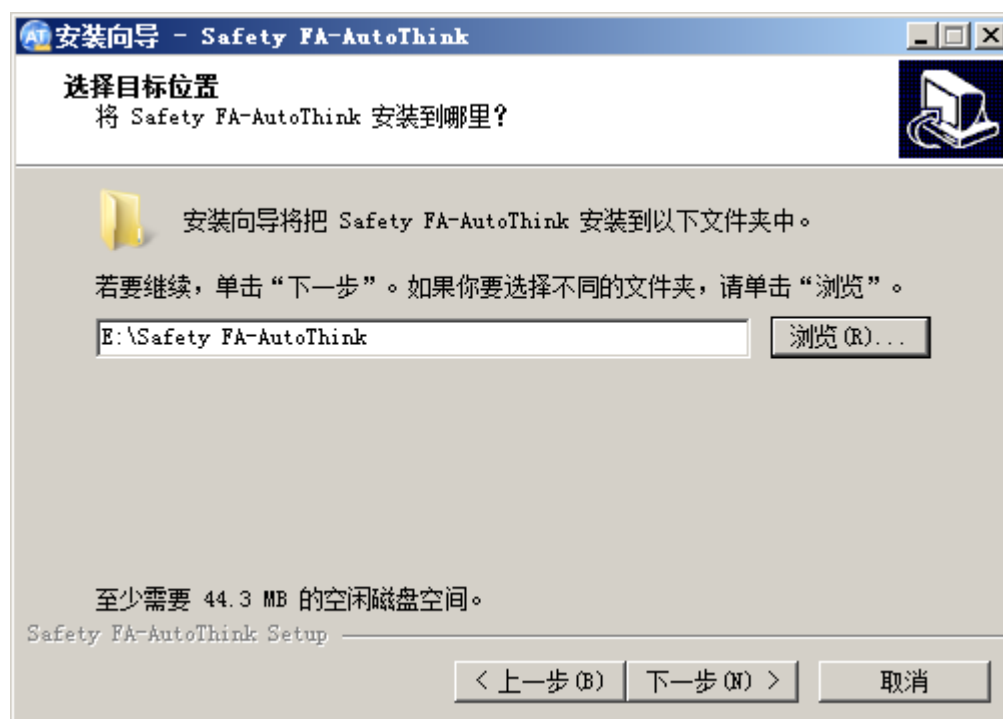


图 4 选择安装路径

第5步 选择开始菜单文件夹

如图 5 所示，通过单击**浏览(R)...**选择文件夹，单击**下一步(N)**。

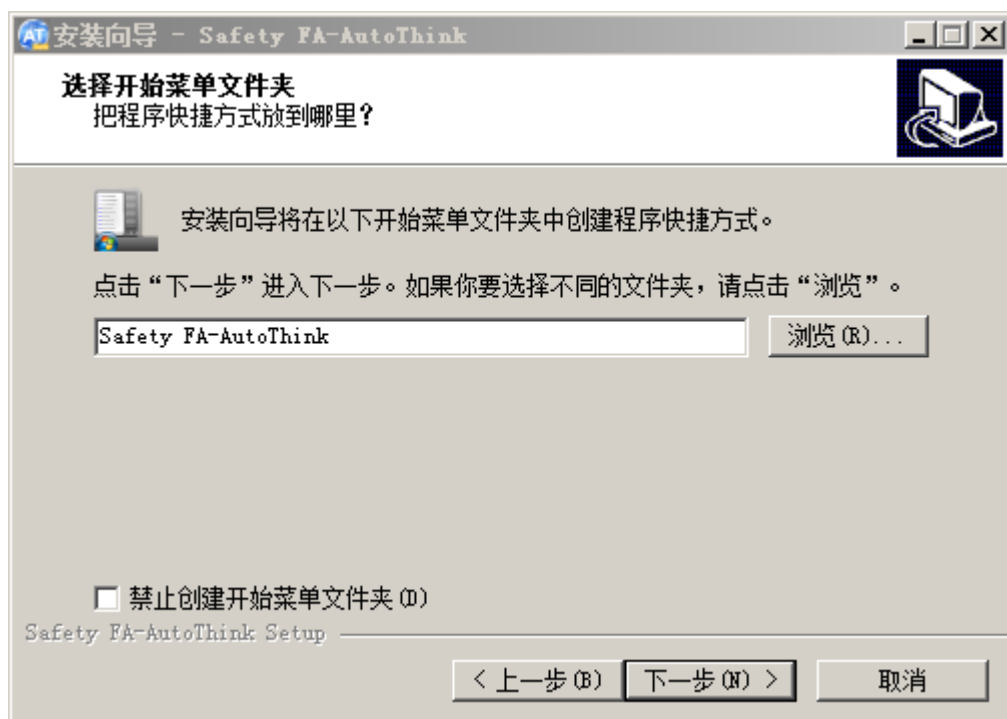


图 5 开始菜单文件夹名称

第6步 选择附加任务

如图 6 所示，在**附加图标**下勾选需要创建的图标项，单击**下一步(N)**。

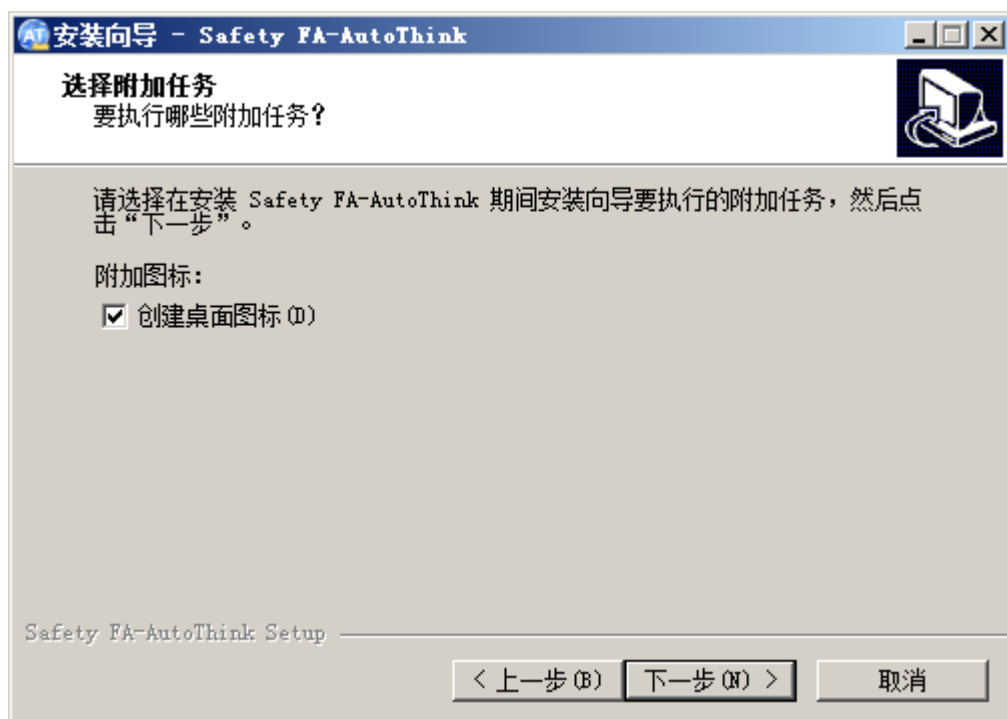


图 6 快捷方式创建

第7步 准备安装

如图 7 所示，确认安装信息。如果需要进行修改，则单击**上一步(B)**。无误后单击**安装**，开始软件安装。



图 7 准备安装

第8步 安装过程

如图 8 所示，会显示安装进度；若要取消安装，则单击**取消**。

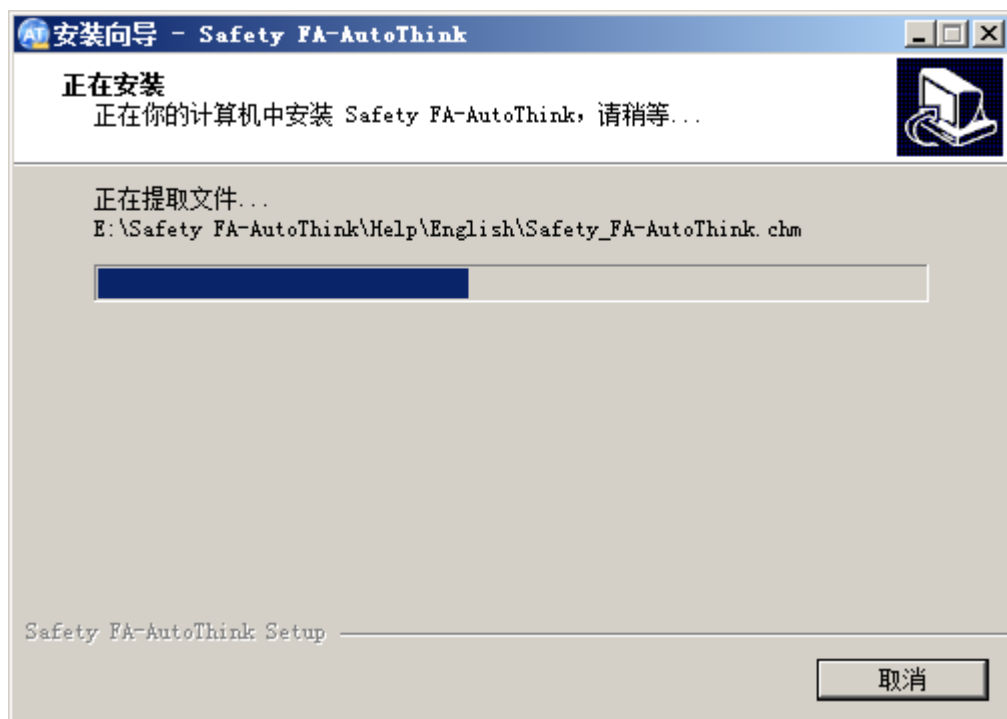


图 8 安装过程

第9步 安装完成提示

安装完毕，弹出“完成 Safety FA-AutoThink 安装”窗口。如图 9 所示。



图 9 完成安装

3.2 卸载软件

可通过控制面板或启动卸载程序卸载 Safety FA-AutoThink 软件。

通过启动卸载程序进行卸载操作：



- 桌面：单击【开始】菜单—【所有程序】—【Safety FA-AutoThink】—【卸载 Safety FA-AutoThink】。



图 10 卸载确认

单击**是(Y)**，开始卸载。

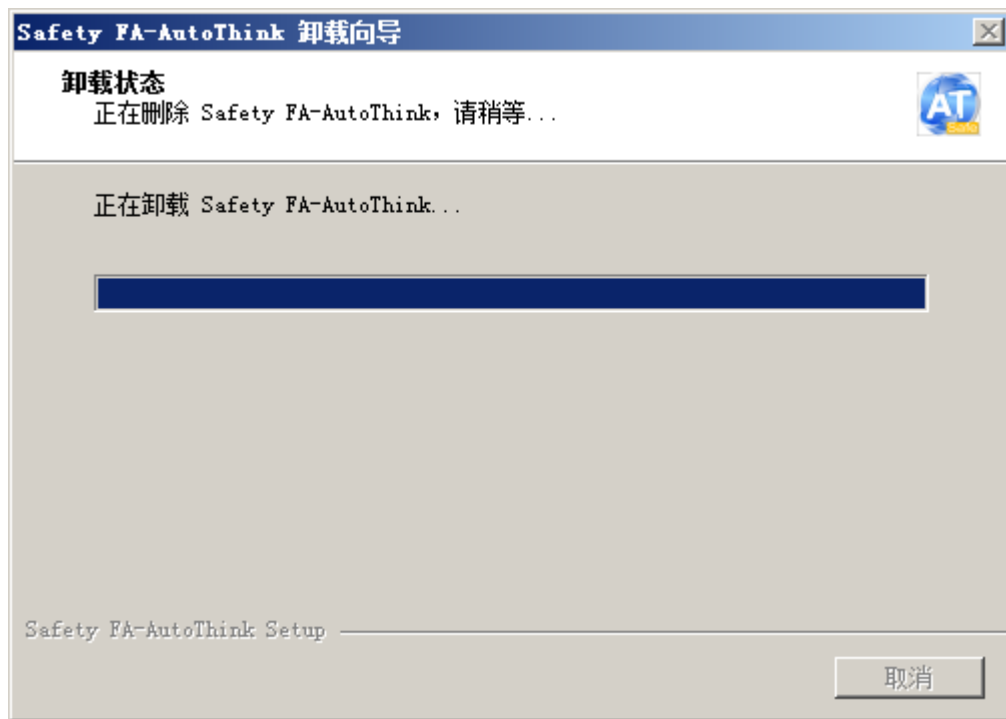


图 11 开始卸载

卸载完毕，弹出提示窗口，如图 12 所示。单击**确定**，完成软件的卸载。



图 12 卸载完成

第4章 软件启动



- 桌面：单击【开始】菜单—【所有程序】—【Safety FA-AutoThink】文件夹—【Safety FA-AutoThink】软件；

- 快捷图标：双击桌面 Safety FA-AutoThink 软件图标。

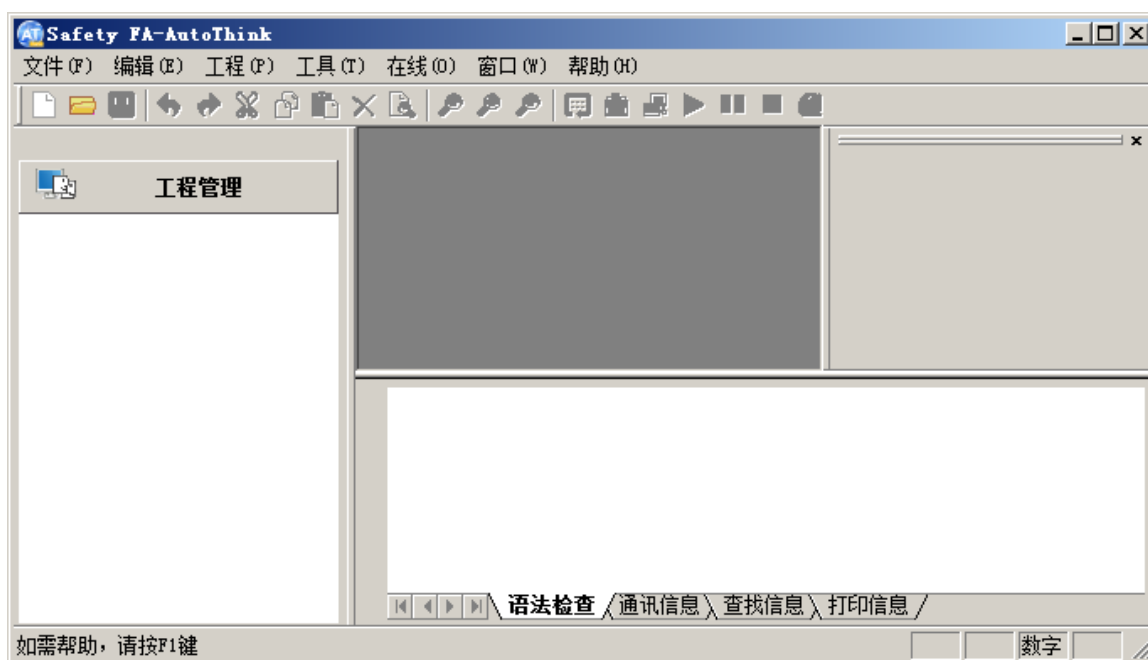


图 13 Safety FA-AutoThink 软件界面

打开已创建的工程时需要权限检查。

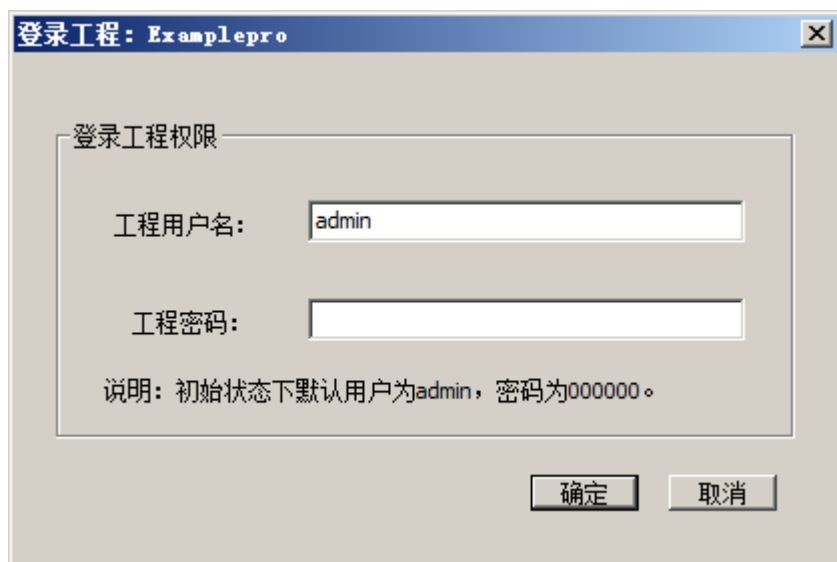


图 14 登录工程对话框

登录成功后，显示对应工程的 Safety FA-AutoThink 软件窗口，如图 15 所示。

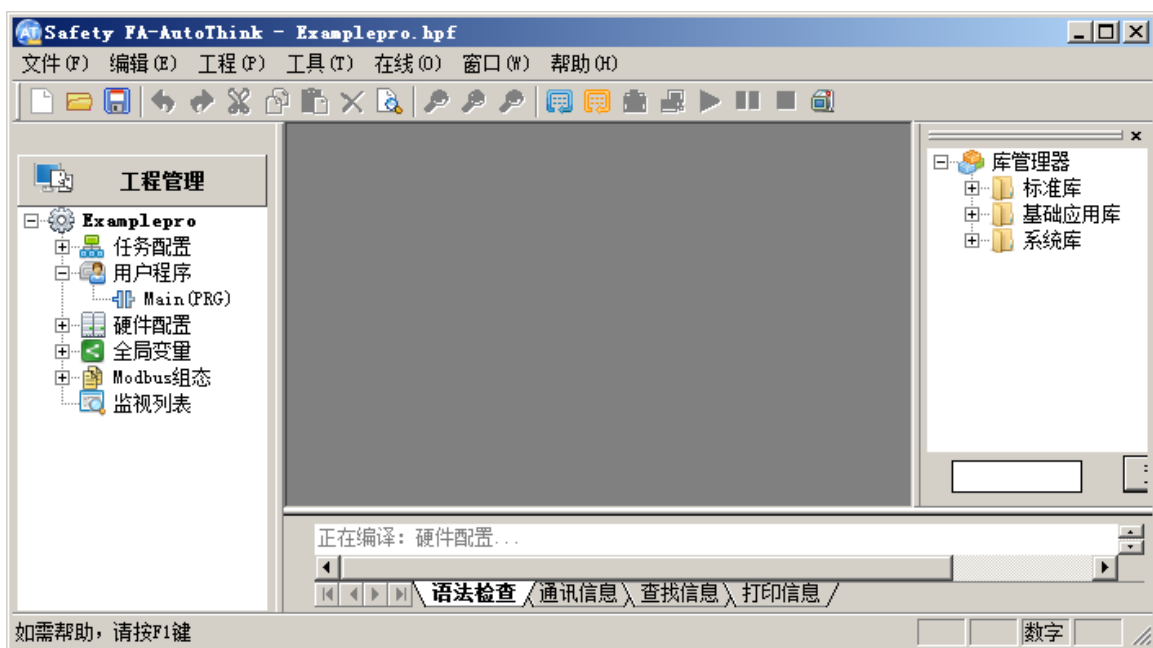


图 15 软件界面

工程打开后，如果需要变更登录的用户信息，可以通过【文件】—【以其他用户登录】命令实现。

第5章 软件界面

软件界面是整个软件的核心组成部分，用于组态工程算法、编译及下装工程至控制器。本节简要介绍了系统的软件界面组成以及各个部分的主要内容。

5.1 标题栏

位于窗口最顶部，用来显示相关内容的名称。如图 16 所示。



图 16 标题栏

- 1: 软件标识;
- 2: 工程名称;
- 3: 工程修改状态 (修改后未保存显示*);
- 4: 当前编辑内容;
- 5: 窗口控制按钮。

5.2 菜单栏

通过单击各个菜单名称，在下拉菜单中选择各个菜单命令项进行算法组态的相关操作。菜单栏根据工作区域编辑内容的不同，显示的菜单也略有不同，一般包括以下几组菜单，如图 17 所示。

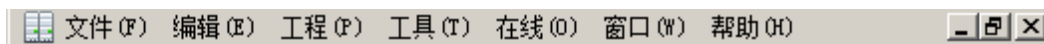


图 17 菜单栏

- 图标：显示当前打开的界面类型，包括： (FBD 语言)、 (LD 语言)  (硬件配置)、 (全局变量) 等。
- 文件：执行与工程文件有关的操作，例如打开、保存、关闭等。
- 编辑：对已有内容进行编辑、替换、查找等操作。
- 工程：对当前正在编辑的工程进行程序编译、设置、日志查看等操作。
- 插入：在组态程序块时显示该菜单，可以插入图形化编程语言的元件。
- 工具：提供控制器操作向导。
- 在线：提供工程在实际系统或仿真系统等操作的相关命令，如下装、监视、仿真等。
- 窗口：提供对当前打开窗口的排列方式、切换显示或关闭等操作的命令。

- 帮助：提供该软件的帮助信息。

5.3 工具栏

用户不仅可以通过菜单栏执行命令，也可以通过工具栏上的按钮执行命令，每个图标都是一个命令。当鼠标放在图标附近时，指针下部有相关内容提示。



图 18 工具栏

表 4 工具栏说明

工具栏	说明
工程工具栏	用于新建、打开、保存工程
编辑工具栏	用于编辑工程中的组态内容
显示调整工具栏	用于调整显示比例，操作对象程序组织单元
在线工具栏	在线操作命令，用于编译、下装、调试等
编程语言工具栏	用于离线组态，操作对象是程序组织单元，显示内容随着语言的不同而不同

5.4 工程管理

以树状图的形式管理当前工程，工程管理树的根节点名称为当前工程名称，包含任务配置、用户程序、硬件配置、全局变量、Modbus 组态、监视列表 6 个子节点。



图 19 工程管理

5.4.1 任务配置

在 Safety FA-AutoThink 中，用户程序的运行管理对应一个任务的执行，为该任务配置所需包含的用户程序、设置任务属性的过程称为任务配置，即任务配置用于引导用户程序的执行。

任务配置采用树状结构方式显示任务名称和任务中被调用程序，缺省添加任务“TASK1”，并且缺省添加 Main 程序的调用。

5.4.2 用户程序

用户程序用于实现控制策略的编程工作，在【用户程序】节点下，通过用户编程的基本单元 POU 来实现用户程序组态。在用户程序编程过程中，使用本软件提供的符合 IEC61131-3 要求的标准化编程语言 LD 和 FBD，POU 类型包括程序、功能块和函数。

在【用户程序】节点下缺省包含 Main (PRG)。通过【用户程序】节点的关联菜单可以进行创建、粘贴、分类管理 POU 的操作。

5.4.3 硬件配置

硬件配置用于配置工程的硬件设备结构，实现与工程数据的关联。

在硬件配置节点下完成系统硬件组态、设备参数设置、通讯参数设置等。设备库窗口中提供了工程可用的模块类型。

5.4.4 全局变量

全局变量下可以进行变量组添加、变量定义和诊断信息查看。包含 5 个变量组，分别为 DP 诊断变量组、全局变量组、Modbus 诊断变量组、通道信息变量组和系统诊断变量组。

5.4.5 Modbus 组态

在该变量组中定义 Modbus 通讯的变量。具体内容详见章节 [7.3.2.3 Modbus 变量组态](#)。

5.4.6 监视列表

监视列表即包含一组重要变量的列表，以便在监视模式下跟踪和观察各个变量的在线值变化情况。

一个工程中可以创建多个监视列表，分类管理所需监视的重要变量，对所添加的列表可以进行删除、重命名等操作，对列表中的变量可以进行增加或者删除操作。添加变量信息时，只需填写正确的变量名即可。

5.5 工作区域

在此区域内完成算法组态、硬件配置等内容的组态工作，如图 20 中红色框标识的位置。

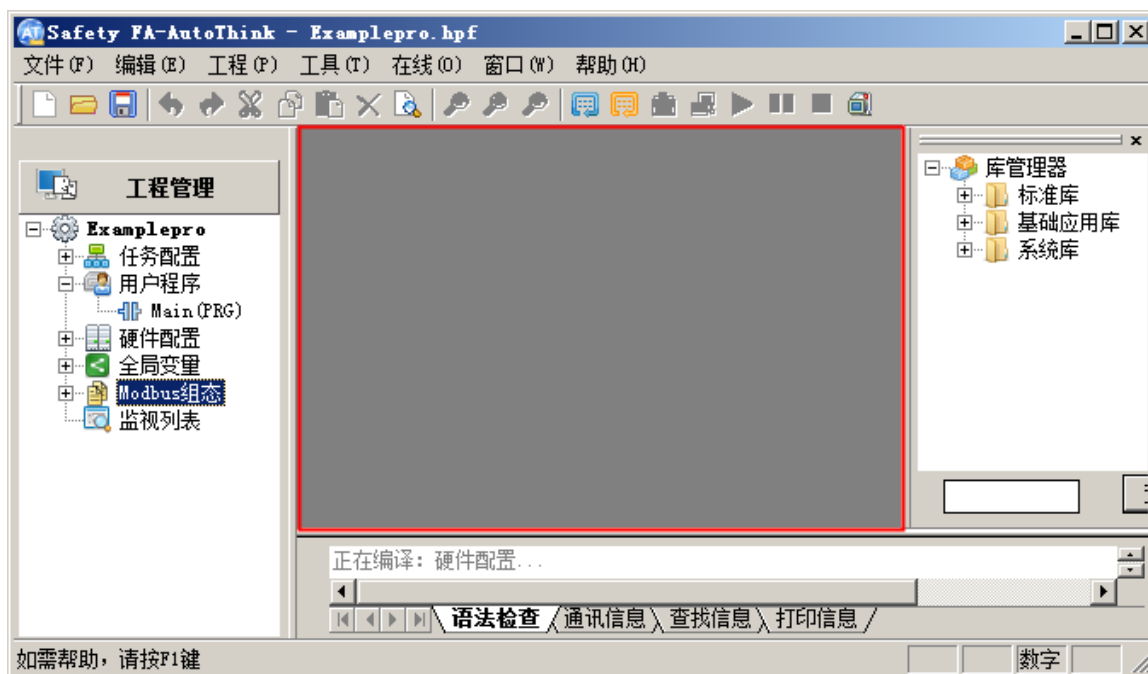


图 20 工作区域

工程管理树任意子节点的编辑窗口显示在工作区域。工作区域顶端以标签页的方式显示当前打开的编辑窗口，选择相应的标签页，则该窗口激活为活动窗口并且标签页标识为亮蓝色，可以进行编辑。也可以通过【窗口】菜单中当前打开的编辑窗口列表进行切换。当打开多个编辑器窗口时，单击工作区域右上角的翻页按钮，向前或向后切换到相应编辑窗口。

5.6 信息窗口

用于显示有关编译、错误、警告、通讯或查找的消息。

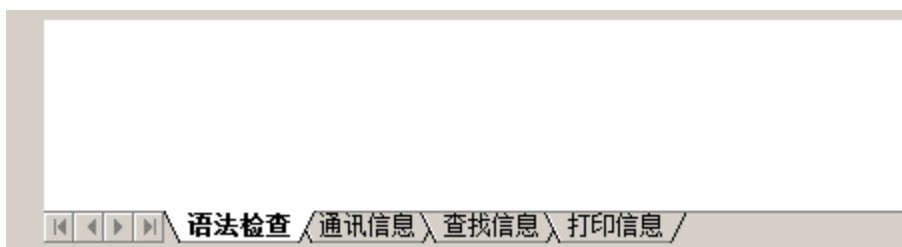


图 21 信息窗口

通过双击信息输出窗口中显示的编译错误信息、警告信息或者查找结果信息的所在行，可以自动定位到编辑器中的相关位置，以便查找相关信息。

在操作过程中出现异常情况时，会在信息输出窗口中以红色字体提示异常信息。

可通过信息窗口右键菜单进行信息的清除和复制。

5.7 状态栏

状态栏位于编程界面的最底端，用于显示与当前编辑窗口相关的信息。



图 22 状态栏

状态栏左侧显示帮助信息行，右侧的 4 个窗格分别指示键盘上的按键状态，从左至右依次是：预留窗格、大写（Caps Lock 键）、数字（Num Lock 键）、滚动（Scroll Lock 键）。

5.8 库管理器

在 Safety FA-AutoThink 系统中，库是指通用功能块（FB）和标准函数（FUN）的集合，实现基本的、常用的逻辑运算，以方便搭建更为复杂的控制策略。库文件保存格式为“库名.hlf”，例如 HS_Standard.hlf（标准库）。缺省的存放路径为“Safety FA_AutoThink\Target\Library”。

在 Safety FA-AutoThink 软件中，通过“库管理器”窗口实现对库的分类管理。如图 23 所示。

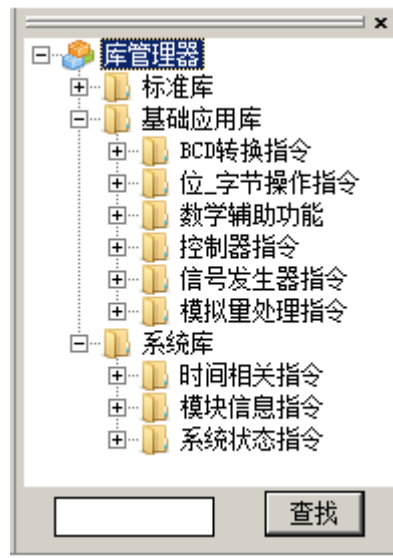


图 23 库管理器

5.8.1 库配置



- 库管理器：右击窗口空白处。

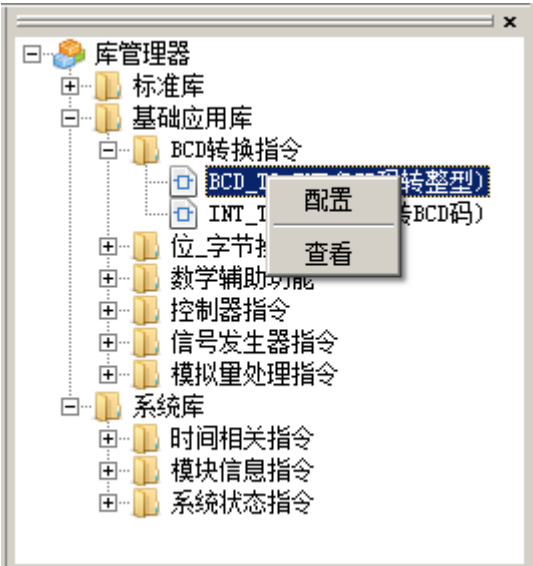


图 24 库管理器的快捷菜单

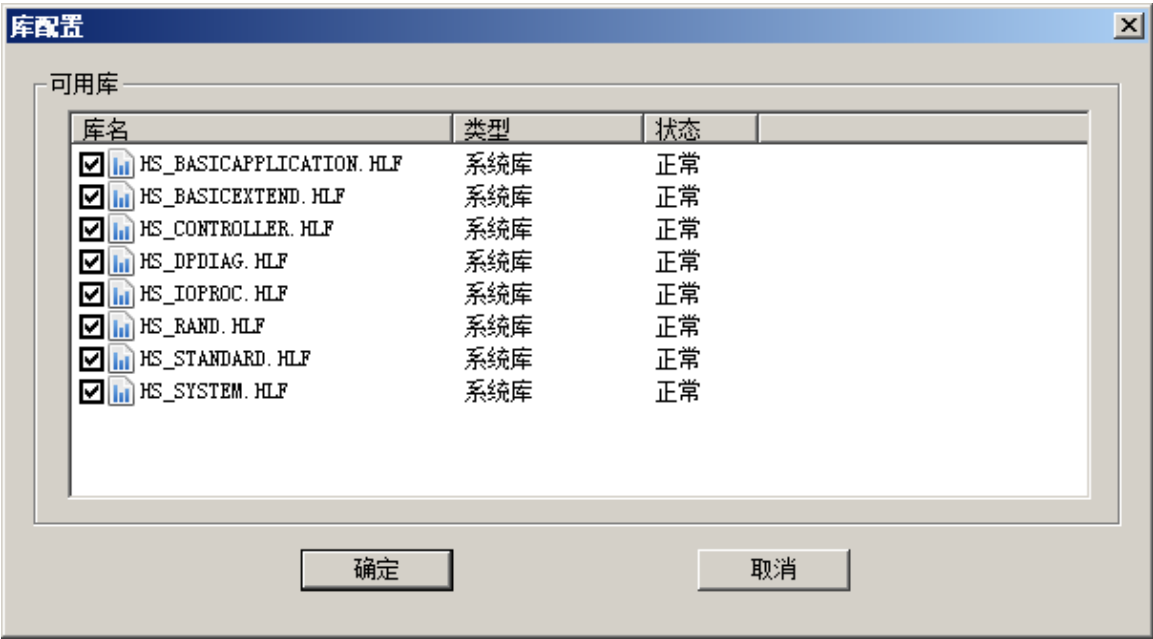


图 25 库配置

1. 相关操作

(1) 排序

通过单击列表头（库名、类型、状态）可以对内容按照字母顺序进行正序或倒序排列。

(2) 复选框

复选框处于勾选状态时，Safety FA-AutoThink 软件会调用对应的库文件。当某一库文件未被调用时，与其相关的算法块在库管理器窗口中标注成红色。

2. 类型

库的类型是按照库内容的实现方法进行分类的，对用户的使用没有任何影响。

系统库：系统缺省自带的库。

用户库：存放用户自定义功能块的库。

N/A：当库文件丢失、损坏时，系统无法判别其类型时用的标识符。

3. 状态

库的状态分为正常和异常两大类，异常状态的库都被标注为红色。具体包含以下 4 类：

(1) 版本低

表示库文件完好，但是版本低于当前软件版本。

■ 表现结果

状态栏会显示版本低。

■ 修正方法

对库进行升级，打开库文件，编译后保存即可。

(2) 损坏

表示库文件被破坏。

■ 表现结果

状态栏显示 N/A。

■ 修正方法

对其进行修补或更换，请与工程实施人员联系。

(3) 冲突

表示库内容有重复。

■ 表现结果

库内容重复时，信息窗口会显示相关提示。

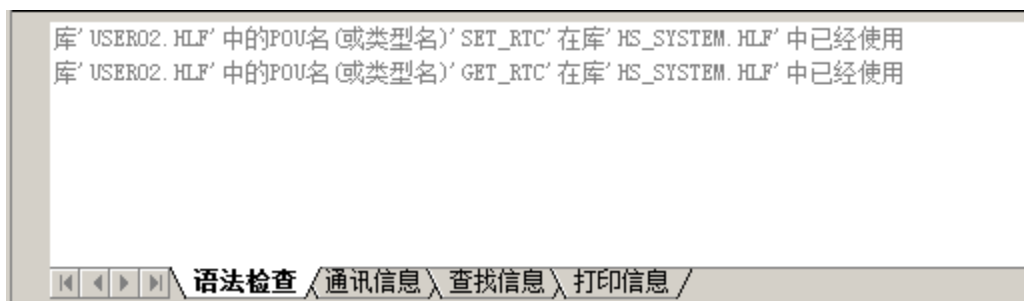


图 26 库内容重复时的提示

■ 修正方法

删除内容重复的库，保证库内容不重复。

(4) 丢失

表示库文件不存在。

■ 表现结果

- 当系统库丢失时，软件无法启动。
- 当用户自定义库丢失时，打开工程，会弹出提示框。



图 27 库文件丢失提示框

同时，库状态显示为丢失。

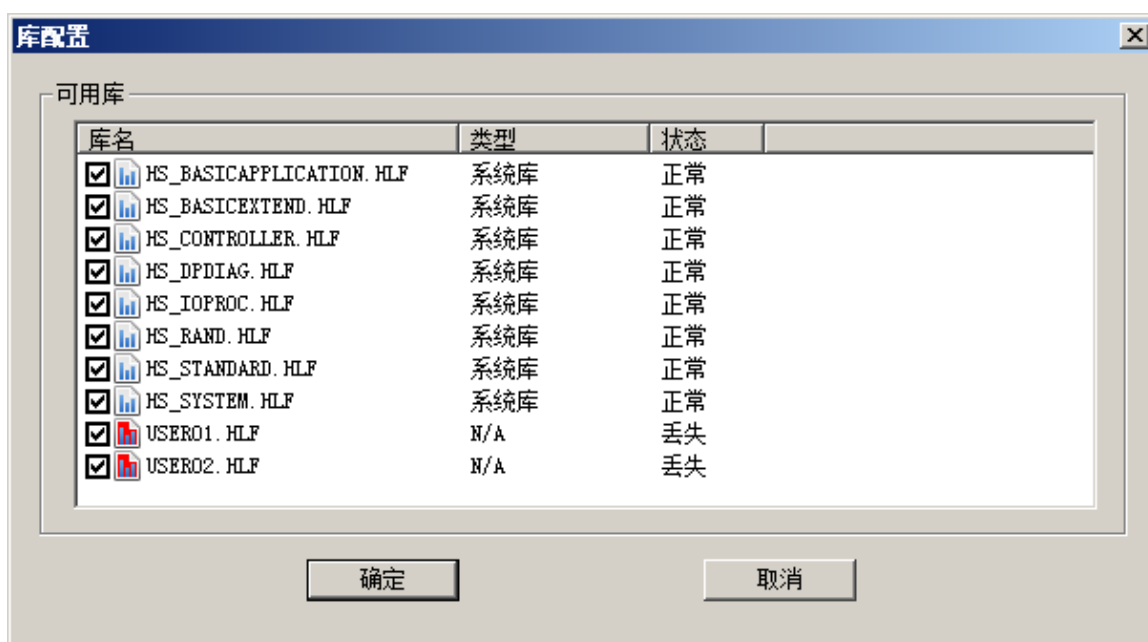


图 28 库配置显示异常

■ 修正方法

系统库丢失，请与工程实施人员联系。

自定义库丢失，需将丢失的库文件补充到目录下：Safety FA-AutoThink\Target\Library。



- 由于个别库之间存在依赖关系，当勾选其中一个时，在更新过程中，系统会自动勾选上与之有关的其他库；当只勾掉其中一个时，此操作无效。
- 如果异常状态的库被勾选，在工程编译时会有错误提示。

5.8.2 查看

通过此命令可以查看系统库和用户库的功能块的详细信息。

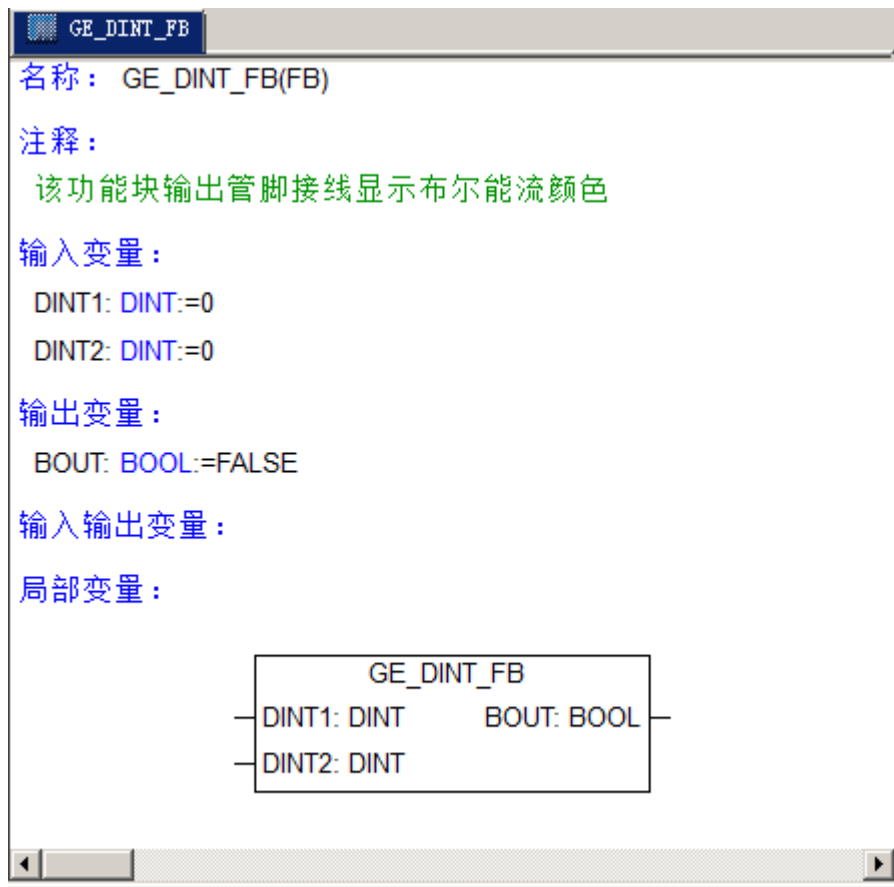


图 29 查看 GE_DINT_FB 块的详细信息

功能块的信息窗口通常由注释（仅支持布尔能流的功能块显示）、引脚、功能块外观图三部分组成。

5.8.3 查找

在库管理器窗口底端附有查找工具，方便用户快速查找。如图 30 所示。



图 30 查找窗口

在输入框中输入查找的条件。输入的内容不区分字母大小写，不限制特殊符号。

系统会以输入的内容作为字符串在库管理器窗口中进行匹配搜索，当指令名称中包含有输入的字符串时，系统视为其匹配成功。



- 两个字符串完全匹配的条件是：长度相等，并且各个对应位置上的字符都相同。

搜索结果显示在库列表框。

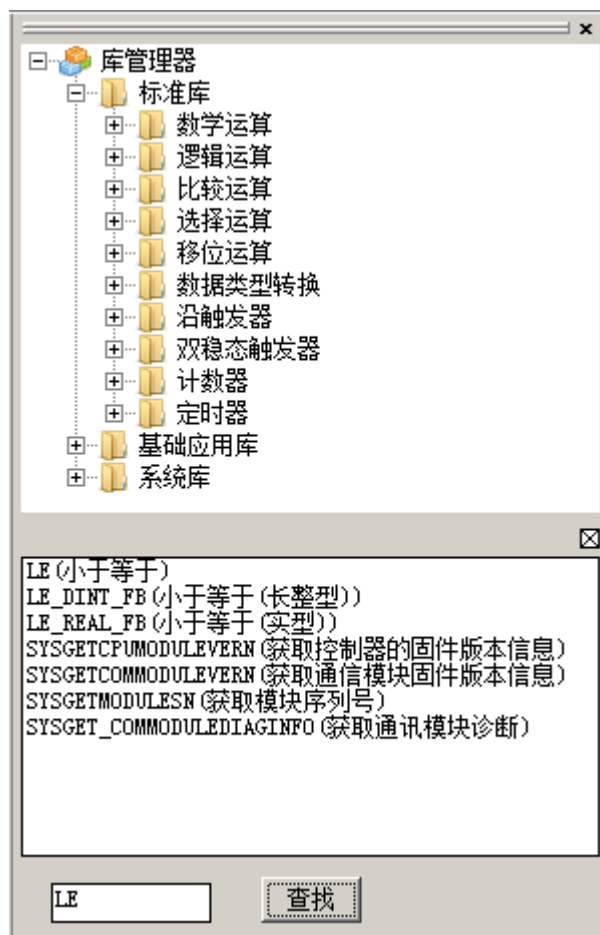


图 31 查找结果

双击查找结果，将自动定位到具体位置，并将其标识为亮蓝色以示区别。如图 32 所示。

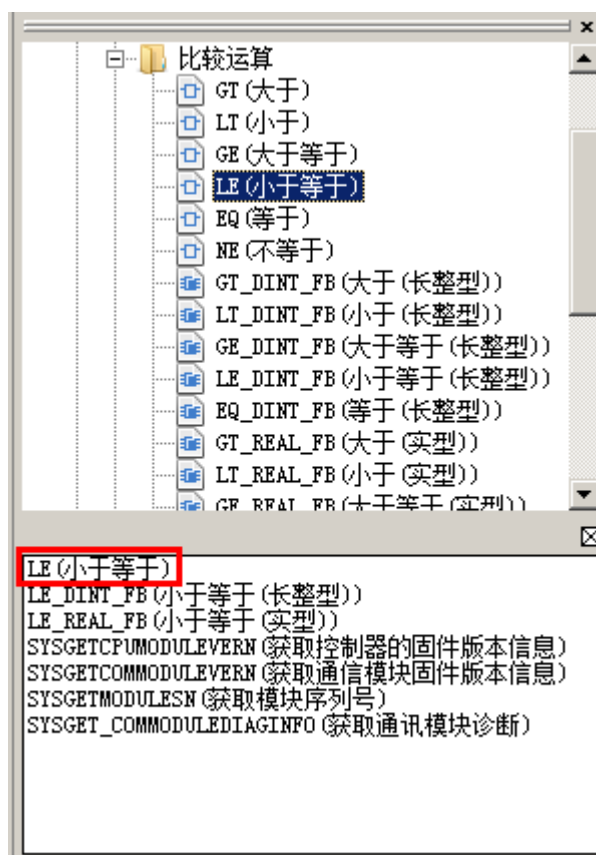


图 32 查找结果定位

5.8.4 库使用

使用“库管理器”中的功能块或函数时，可以将目标对象拖动到工作区域，待鼠标呈  形态时，释放鼠标即可完成添加操作；否则，如果鼠标呈  形态，表示该位置不能添加（该位置已有块元件，或者空白位置过小）。

库中功能块和函数的详细说明，请参见《LKS 安全型控制系统指令手册》。

5.9 设备库

该窗口同样采用树状结构管理方式。主要分为控制器、通信设备和协议。



图 33 设备库

■ 查找

在窗口底端有查找工具，方便用户快速查找。

在输入框中输入查找的条件。输入的内容不区分字母大小写，不限制特殊符号。

系统会以输入的内容作为字符串在设备窗口中进行匹配搜索，当模块名称中包含有输入的字符串时，系统视为其匹配成功。将匹配成功的结果显示在库列表框的中下方。

双击匹配结果中的任意条目，系统自动定位到选中内容的具体节点位置，并标识为亮蓝色，便于区分。



- 两个字符串完全匹配的条件是：长度相等，并且各个对应位置上的字符都相同。

第6章 工程管理

6.1 新建工程

此命令创建一个新的工程或库。



菜单栏：单击【文件】—【新建】；

工具栏：；

快捷键：Ctrl+N。

6.1.1 工程

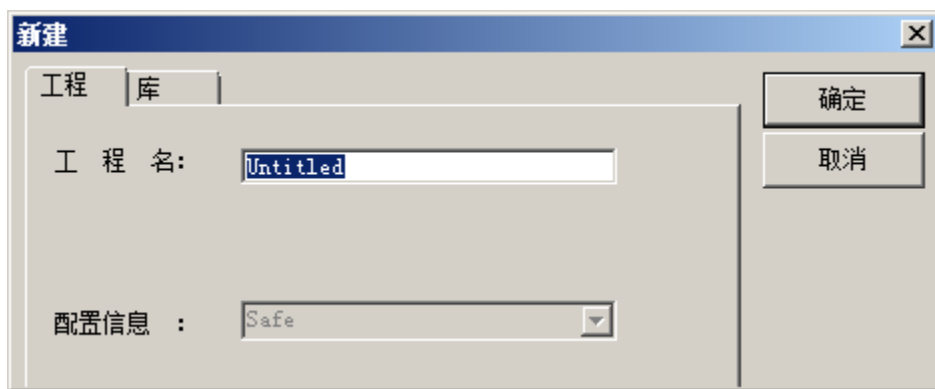


图 34 新建对话框—工程标签页

- 工程名：缺省为“Untitled”。
 - 工程名称中不能使用“\”、“/”、“:”、“*”、“?”、“””、“<”、“>”、“|”这9种字符。
 - 工程名称不可为空。
 - 工程名称长度最多不超过32字节，超出范围的部分无法输入。工程名无效时，**确定**按钮不可操作。
- 配置信息：Safe。

6.1.2 库

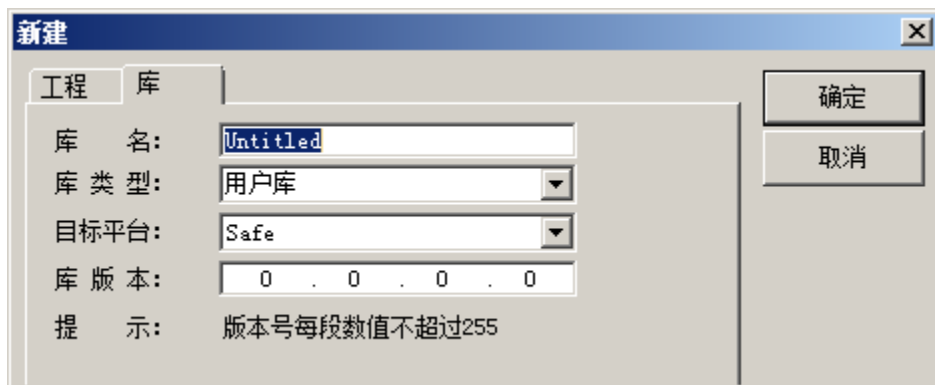


图 35 新建对话框—库标签页

- 库名：库名称缺省为“Untitled”，库的命名不能以“HS_”开头，其他命名规则与工程名的命名规则一致。
- 库类型：用户库。
- 目标平台：Safe。
- 库版本：采用四段式版本号命名方式，每段设置的数字范围为 0~255。用户借助它可以记录库变更的版本（记录变更发生的日期，如“20.15.01.20”）。

6.2 任务配置

用户程序只有被任务调用，才能执行。下面就任务配置节点的属性和如何在任务中添加或删除 POU 调用进行说明。

6.2.1 程序调用

【任务配置】节点下，缺省有 TASK1 任务节点，该任务节点缺省调用【用户程序】节点下已配置的程序 Main（PRG）。用户可通过任务节点的右键菜单添加程序调用。



- 工程管理：右击任务节点 TASK1，单击【添加程序调用】。



图 36 添加程序调用

弹出“添加程序调用”对话框，可添加 POU 列表框中显示【用户程序】节点下所有的 POU(PRG)，当前任务下 POU 列表框中显示该任务下已调用的 POU，如图 37 所示。选中需要调用的 POU，单击添加按钮，将该 POU 添加到当前任务下进行调用。每次添加后，左侧列表框中光标自动移到下一个 POU 位置。

在右侧列表框中选中 POU，单击**删除**按钮，将删除对该 POU 的调用。在添加和删除时，可通过 **Shift** 或 **Ctrl** 键进行多选。

单击**全选**，将可添加的 POU 全部添加到任务下进行调用。

单击**全删**，可删除任务下所有已调用的 POU。

单击**重置**，将 POU 的调用恢复到该窗口打开时的状态。

单击按钮 、，调整右侧列表框中 POU 的调用顺序。

编辑完成后，单击**确定**，关闭“添加程序调用”对话框。

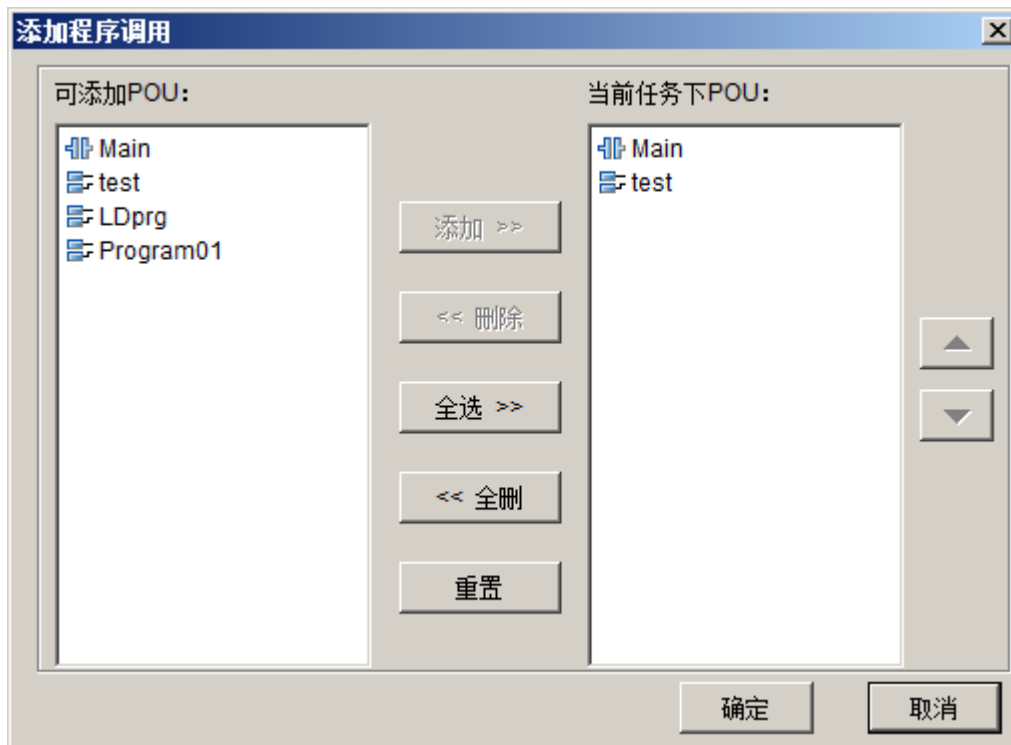


图 37 选择调用的 POU

任务中调用的 POU 按照从上到下的顺序被执行。

■ 重命名被调用的 POU

修改【用户程序】节点下 POU 的名字，任务节点下的同名 POU 同步被更新。



- 工程管理：右击【用户程序】节点下被调用的 POU，单击【重命名】。

■ 删除被调用的 POU



- 工程管理：右击任务节点下被调用的 POU，单击【删除】。

如果删除【用户程序】节点下被调用的 POU，则任务节点下的同名 POU 同时被删除。

6.2.2 任务属性配置



- 工程管理：右击任务节点 TASK1，单击【修改】。

弹出“编辑任务”对话框，如图 38 所示。

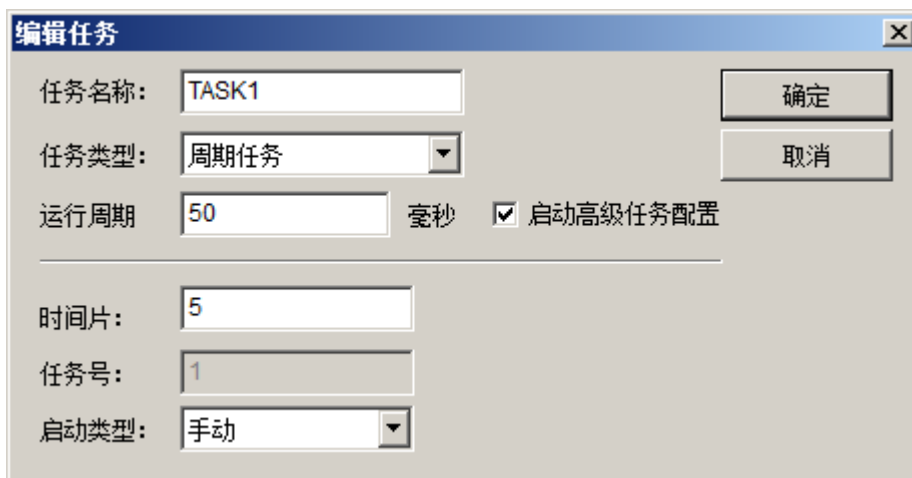


图 38 编辑任务

- 任务名称：名字由数字、字母、下划线组成，不能以数字开头。长度不超过 32 个字符。
- 任务类型：周期任务，任务根据设置的时间片和运行周期循环执行。
- 运行周期：周期性执行任务的时间间隔。默认为 50 毫秒，设置范围为 15~100,000 之间的整数。
- 时间片：是由系统划分的最小执行时间单元。用户可以设置时间片个数，默认为 1，可设置为 1~50 之间的整数。每个时间片默认为 62.5 μ s。
- 任务号：任务 ID。
- 启动类型：默认为手动，可以选择手动和自动。选择手动，工程下装后进入监视状态，任务是停止状态，需要手动单击运行按钮，任务才可以运行。选择为自动，工程下装后任务自动运行。

6.3 设置工程安全属性

右击【工程管理】根节点（即工程名称），单击属性，可以设置工程安全属性以及查看工程的配置信息。

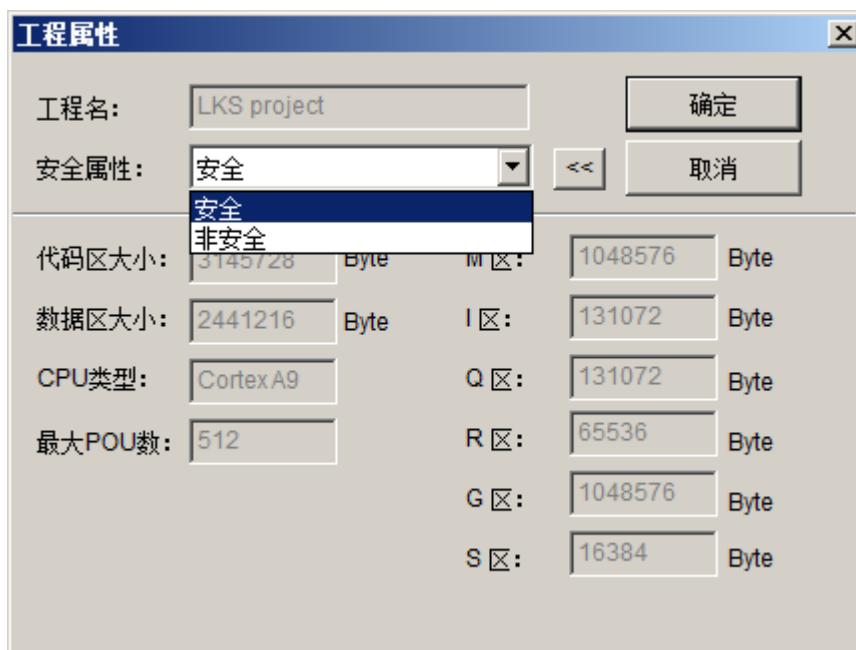


图 39 工程属性

6.3.1 设置安全属性

工程可以设置为安全或非安全两种属性，默认为安全型，根据实际系统要求进行设置。

在安全工程中，Modbus 通讯时，系统只能作从站，且从站为只读属性，不能进行写操作。在安全工程中，控制器钥匙开关拨到 RUN 位置，所有数据区数据（S 区除外）恢复为初始值。

6.3.2 工程配置信息

■ 代码区

存储用户组态的控制逻辑（PRG 类型 POU 中的逻辑实现）、自定义功能块（FB 类型中 POU 的逻辑实现）和函数（Function 类型中 POU 的逻辑实现）。

■ 数据区

存储用户自定义的变量，包括局部变量和全局变量。

■ 最大 POU 数

是用户 POU 和系统 POU 个数之和。

■ M 区

参数区，该区用来储存参数变量或常量。在运行时，定义在该区域中的变量不能被赋值，只能通过“写变量”操作改变变量的值。该区域中的变量具有同步特性。

■ I 区

输入区，在每个扫描周期的首端，CPU 对输入点进行采样，并将采样值存储到内存储器的输入区。可按位(BIT)、字节(BYTE)、字(WORD)、双字(DWORD)来存取输入区。

■ Q 区

输出区，在每个扫描周期的末端，CPU 将内存储器的输出区的数据传送到物理输出点上。可按位(BIT)、字节(BYTE)、字(WORD)、双字(DWORD)来存取输出区。

■ R 区

保持区，存储掉电保持变量，若将数据或变量定义为 R 区，则当 CPU 模块掉电时，系统会自动保存该数据，待重新上电后，自动恢复该数据。

■ G 区

普通变量区，也属于 PLC 的中间寄存器区，用于存储和管理中间过程产生的数据和状态。

■ S 区：特殊寄存器区，该区用来存储 Modbus TCP 和系统的诊断信息。

自定义变量在声明时，根据其用途设置其所属区域（G 区、M 区或者 R 区）。

工程编译通过后，各个区的大小及使用率会显示在信息窗口中。

6.4 工程安全

工程安全是指对访问工程的用户进行的权限管理，以保护工程的安全。不同的用户，根据其所属权限级别的不同，相应的对工程、控制器的操作和访问权限也不同。



- 菜单栏：单击【工程】—【工程安全】。



图 40 工程安全

新创建的工程默认为 Level0 级用户。

6.4.1 新建用户

创建新的用户，新用户的级别只能设置为 Level1 或 Level2。

新建用户信息

用户信息

用户名:

描述:

密码:

确认密码:

权限

☐ Level 0

☒ Level 1

☐ Level 2

确定

取消

图 41 新建用户信息

- 用户名：由字母、数字、下划线“_”任意组合，且必须以字母或者下划线开头，不能以数字开头，长度不超过 32 个字符；
- 描述：有关该用户的描述信息，可以设置为任何字符，长度不超过 256 个字符；
- 密码、确认密码：在此栏内输入设置的用户密码，可以设置为任何字符，长度范围为 6~12 个字符；
- 权限：设置用户的权限级别（Level0~2），权限依次降低。新建用户的级别要低于当前登录的用户级别。各级别具体的操作权限见表 5。

表 5 用户权限操作表

权限内容	Level0	Level1	Level2	说明
新建工程	√	×	×	
编辑修改用户名和密码	√	×	×	高级别用户可以编辑低级别用户的用户名、描述和密码。同级用户只能修改自己的用户名和描述，不能修改密码，也不能修改其他同级用户信息
创建、编辑或删除 Level 1 用户	√	×	×	
创建、编辑或删除 Level 2 用户	√	√	×	
离线组态	√	√	×	

权限内容	Level0	Level1	Level2	说明
修改保存工程	√	√	×	
下装	√	√	×	
运行、暂停、停止任务	√	√	×	
在线监视	√	√	√	
仿真	√	√	√	
调试变量	√	×	×	
上传工程文件	√	√	√	
下载工程文件	√	√	×	
变量回读	√	√	×	
清空控制器	√	√	×	
控制器校时	√	√	×	
复位、冷复位、热复位	√	√	×	
SOE 信息读取	√	√	√	
NTP 校时	√	√	×	



- 当 Level0 用户忘记密码后，无法对该工程进行最高权限操作，请联系厂商协助解决。

6.4.2 编辑用户

对已存在的用户信息进行变更，如密码、权限级别；对于 Level0 级别的用户，不能修改其权限级别。

6.4.3 删除用户

清除创建的用户。只能高级别的用户删除低级别的用户。

6.4.4 使用时机

在工程应用中，工程安全的主要使用情形如下：

■ 登录工程

打开已有工程时，需要进行登录工程权限验证；

■ 切换工程用户

使用【文件】—【以其他用户登录】命令来切换登录的工程用户时，需要进行登录工程权限验证。

■ 控制器密码设置

使用【工具】—【辅助工具】—【控制器操作】中的控制器锁来设置在线操作时的控制器登录密码，操作前，需要输入控制器锁密码。详见 [11.1.3 控制器锁](#)。

6.5 库安全

用户可以对自定义库进行密码保护，只有通过密码验证，才能打开库进行编辑。



- 菜单栏：单击【工程】—【库安全】。



图 42 设置库密码

密码可以设置为任何字符，长度范围为 6~12 个字符。

6.6 工程设置



- 菜单栏：单击【工程】—【选项】。

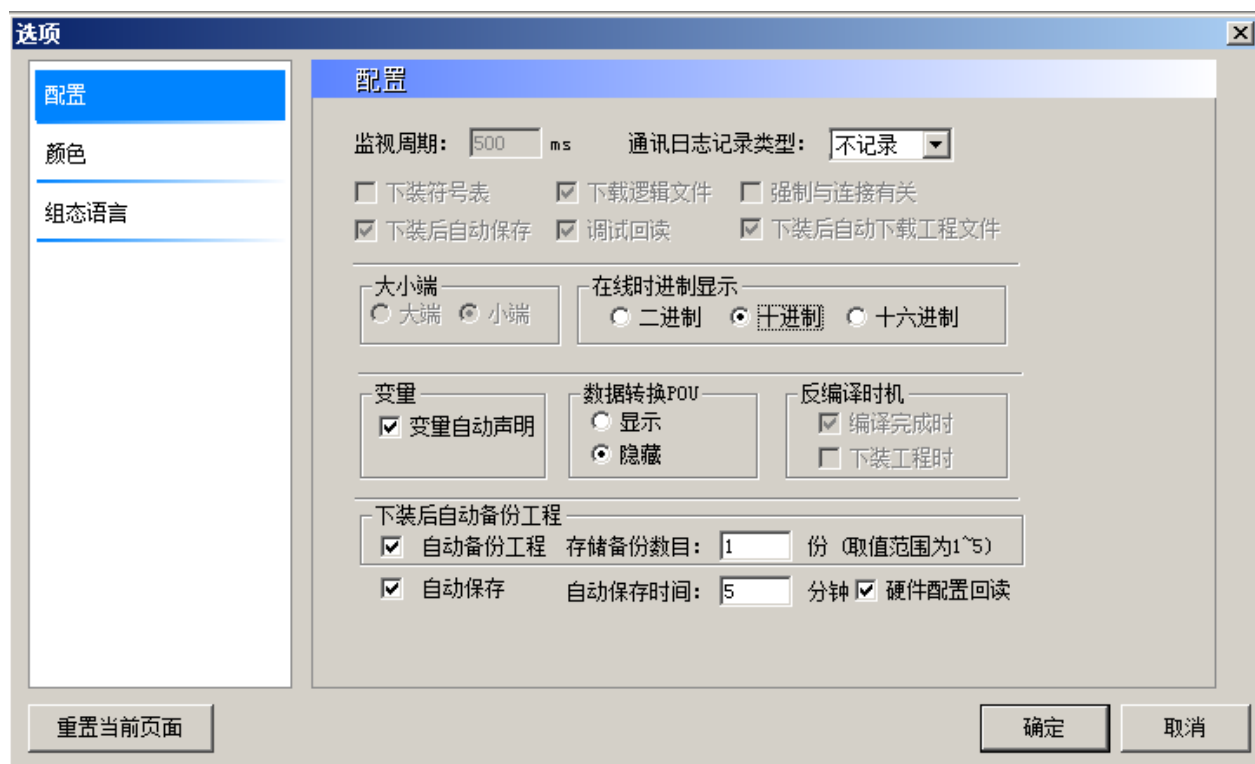


图 43 选项对话框

6.6.1 配置

- 监视周期: Safety FA-AutoThink 软件从控制器读取数据的间隔时间, 缺省为 500ms。
- 通讯日志记录类型: 设置通讯日志是否进行记录以及记录的格式, 可以选择: 不记录、精简模式、完全模式。
- 下装符号表: 将工程编译生成的符号表下装指定的控制站, 缺省选中。
- 下载逻辑文件: 将编译通过的可执行逻辑下装指定的控制站。
- 强制与连接有关: 设置强制功能是否与在线连接有关。
- 下装后自动保存: 设置工程下装后是否进行自动保存。
- 调试回读: 设置在调试时是否可以进行数据回读到工程中。
- 下装后自动下载工程文件: 设置下装成功后是否自动下载工程文件。
- 大小端: 指的是数据的存储方式, 数据的高位存在内存的高地址是小端, 数据的高位存在内存的低地址是大端。缺省采用小端方式。
- 在线时进制显示: 在线或仿真状态下, 设置在线显示的数据格式。缺省设置为十进制显示, 该设置可以修改。
- 变量自动声明: 在编写程序时, 如果创建了一个新变量, 是否自动弹出“变量自动声明”对话框进行声明。选中状态, 即会弹出对话框进行声明提示; 反之, 不弹出。
- 数据转换 POU: 用于设置任务 TASK1 中是否显示 HSGetQ POU。HSGetQ POU 中组态了 DP 从站诊断的相关逻辑, 该 POU 不支持用户编辑。
- 反编译时机: 设置反编译自动执行的时机, 缺省为编译完成时。
- 下装后自动备份工程: 在用户执行下装操作后, 设置系统是否自动备份当前工程。

系统自动备份工程时, 会将当前工程目录下所有工程文件进行备份。备份工程以“工程名称_备份时间(年-月-日 时-分-秒)”方式命名, 被保存在工程路径下的 BackUp_PRJ 文件夹。

当备份的工程数超过设置的最大值时, 会自动删除最早备份的工程文件夹。备份过程中出现异常情况(如磁盘空间不足)时, 系统会提示用户自行完成备份。



- 用户恢复备份时, 请先关闭工程组态窗口, 使用备份文件替换工程源文件。

■ 自动保存

以设置的自动保存时间为间隔, 对工程进行自动保存。

当工程异常关闭, 再次打开工程时, 用户可以选择打开系统自动保存的工程或者用户手动保存的工程。

■ 硬件配置回读

勾选硬件配置回读, 当编译通过后, 自动回读模块的用户参数配置, 供用户比对, 下装时进行回读信息确认。

6.6.2 颜色

该选项设置组态软件中各个编辑元素的颜色, 设置项的信息内容如图 44 所示。

在此对话框中显示了当前软件中缺省定义的各种缺省颜色，用于区分在程序定义时使用的不同对象颜色，使组态内容更易区分。可以通过下拉颜色菜单进行修改，也可以通过**重置当前页面**按钮将各项内容恢复到系统缺省设置。

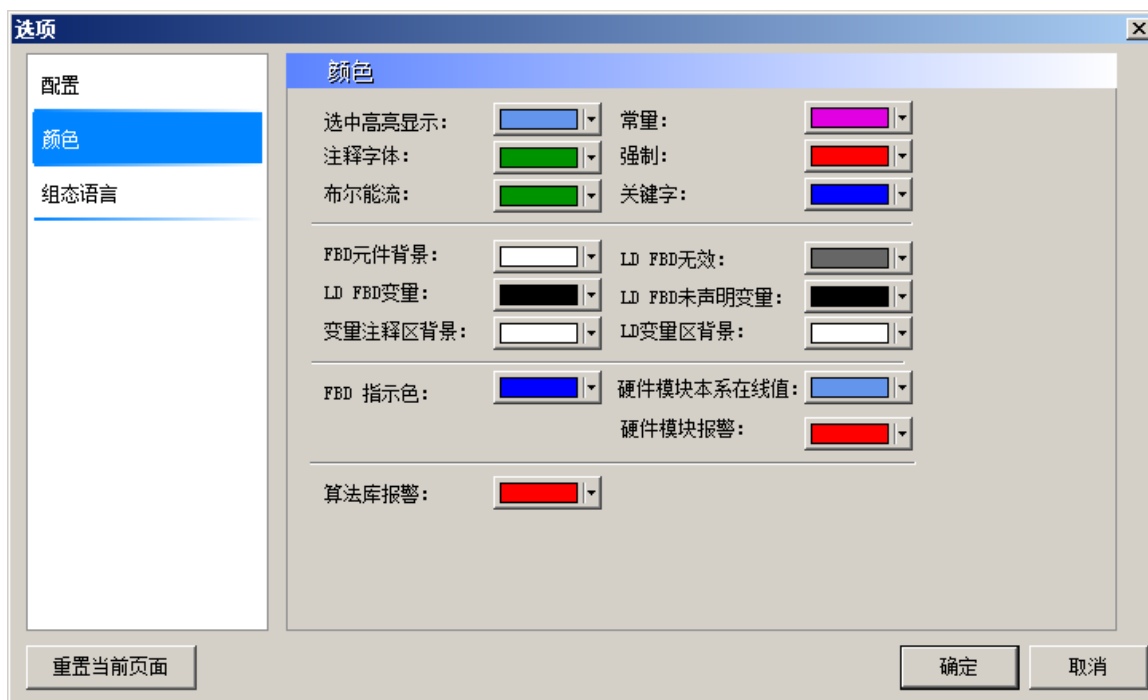


图 44 颜色对话框

- 布尔能流：用来设置 BOOL 类型输出引脚为 TRUE 状态时连线的颜色。默认为绿色（如图 45 所示），用户可以重新设置。

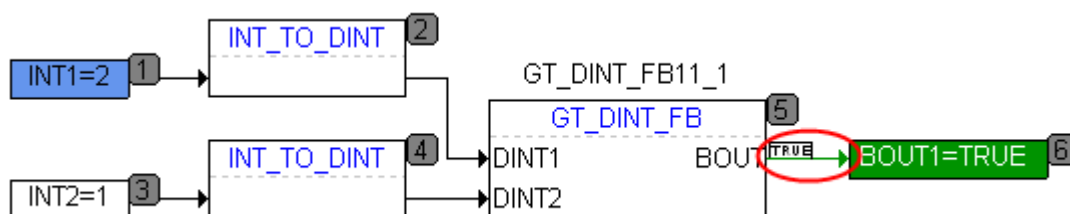


图 45 布尔能流



- 用户设置布尔能流可以影响“库管理器”窗口中逻辑运算（不包含 XOR）和比较运算（不包含 NE）功能块的输出引脚（BOOL 类型）线的颜色。

6.6.3 组态语言

该项设置 IEC 各种组态语言的显示特性，以及各个语言（LD、FBD）特有的显示特性，如图 46 所示。

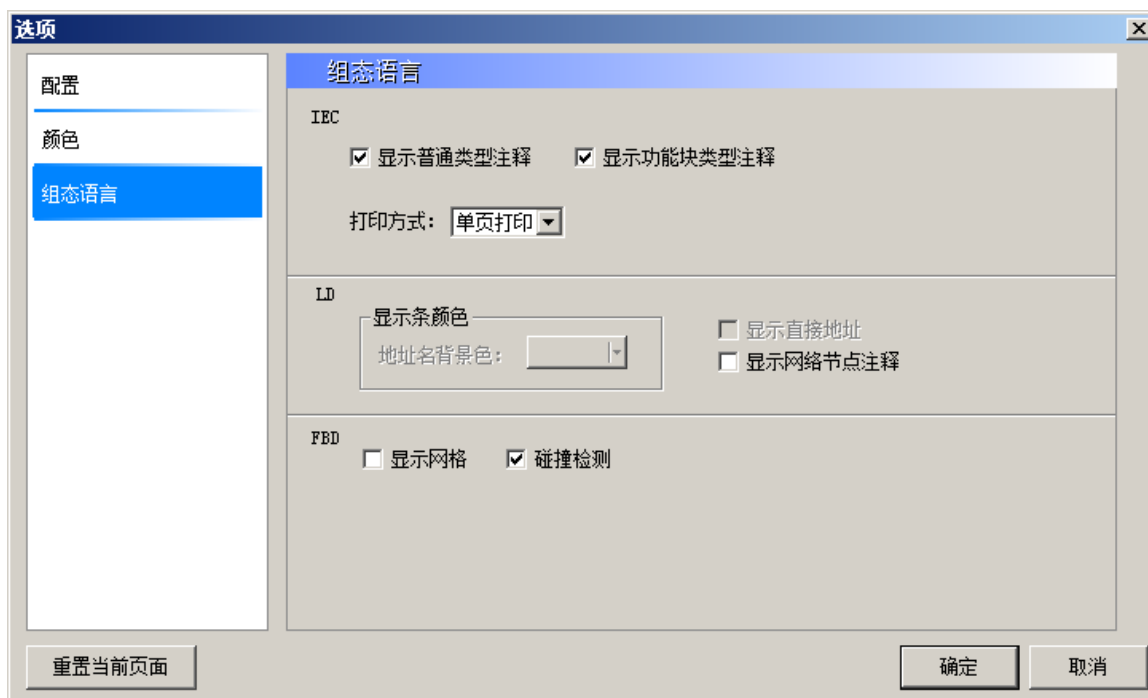


图 46 组态语言对话框

6.6.3.1 IEC

- 显示普通类型注释：显示普通类型变量的变量说明，缺省用 CMT 表示；编辑内容后，字体显示绿色（通过颜色可以修改）。
- 显示功能块类型注释：显示功能块类型变量的变量说明，缺省用 CMT 表示；编辑内容后，字体显示绿色（通过颜色可以修改）。
- 打印方式：设置 POU 页面的打印输出方式。缺省设置为单页打印。

- ☐ 分页打印：是按照打印机中设置的纸张规格打印 POU 页面内容。FBD 语言 POU 中按照打印机中设置的纸张规格将 POU 分成若干页，并用黑点线表示分页符。

这种打印方式下，每个 POU 打印预览时系统会自动添加页面编号，通常按照从左到右，从上到下的顺序编码：

LD 的编号形式为栅格的行号+列号，如 1-1、1-2、2-1。

FBD 无编号。

- ☐ 单页打印：是将 POU 页面的所有内容打印在一张纸上。如果 POU 的纸张类型和打印机中设置的纸张大小不一致，打印时系统会自动调整 POU 内容的显示比例，以确保一张纸上可以包含所有内容。

6.6.3.2 LD

显示与 LD 语言相关的参数设置。

显示条颜色：设置 LD 语言组态的用户程序中的地址名的背景颜色。

显示直接地址：设置添加一个触点或线圈后是否显示地址。缺省不显示。

显示网络节点注释：设置网络节点注释（Comment）是否显示，缺省不显示。图 47 是显示的效果。

0001	comment:

图 47 网络节点注释

6.6.3.3 FBD

显示与 FBD 语言相关的参数设置。

碰撞检测：检测在编写用户程序时，各个元件是否允许互相重叠。当不检测碰撞时，如果元件发生重叠时，软件不作处理；

显示网格：在编程区是否显示网格标记，不勾选，编程区显示为空白区域。

6.6.4 重置当前页面

用于将选项中所有参数的当前值恢复至缺省值。

6.7 保存工程

6.7.1 保存



- 菜单栏：单击【文件】—【保存】；
- 工具栏：；
- 快捷键：Ctrl+S

保存当前工程所做的修改。

6.7.2 另存为

用户变更当前工程或库的保存路径时，可以通过【另存为】命令实现。



- 菜单栏：单击【文件】—【另存为】；

6.8 关闭工程



- 菜单栏：单击【文件】—【关闭】。
- 标题栏：单击.

关闭当前编辑的工程。

此操作不退出 Safety FA-AutoThink 软件程序（Safety FA-AutoThink.exe）。

6.9 最近工程列表



- 菜单栏：单击【文件】—【最近工程列表】。

显示最近打开过的工程的列表，包括工程路径和工程名称。通过此命令可以快速切换工程。

6.10 上传工程文件



- 菜单栏：单击【在线】—【上传工程文件】。

使用该命令可以将工程文件从控制器上传至本地计算机。

该命令与【下载工程文件】命令是一对相互操作，是对当前的工程文件在本地和控制器间进行相互拷贝。

6.11 下载工程文件



- 菜单栏：单击【在线】—【下载工程文件】。

使用该命令可以将工程文件下载到控制器。

6.12 窗口

【窗口 (W)】菜单提供用于设置工作区各个窗口的布局方式的命令，隐藏或显示指定的窗口，且提供可以切换显示或最近打开的窗口命令，如图 48 所示，下面分别介绍具体功能。



图 48 “窗口”菜单

窗口菜单中的操作命令及其功能如表 6 所示。

表 6 窗口菜单项说明

菜单命令	说明
层叠	所有窗口在工作区串联排列，一个叠一个，依次错开
水平	所有窗口在工作区水平排列，不重叠，且充满整个工作区
垂直	所有窗口在工作区垂直排列，不重叠，且充满整个工作区
全部关闭	关闭工作区域当前打开的所有窗口

■ 常用窗口

该区域列出了常用的辅助窗口：**【信息窗口】**、**【库管理窗口】**、**【设备库窗口】**、**【状态栏】**，通过勾选来显示相应窗口。

在**【窗口】**下拉菜单的底端显示当前工作区域打开的各个编辑窗口名称，单击窗口名称可以切换为当前的活动窗口。

6.13 帮助

【帮助】菜单包含**【内容】**和**【索引】**2 个帮助窗口命令，选择其中一项打开在线帮助手册。

单击**【关于(A)...**显示软件产品商标，说明软件名称、标识、版本、版权所有者等信息。

第7章 硬件配置

硬件配置节点下完成硬件系统搭建、模块参数配置及通讯参数配置等。

7.1 控制器模块

在【硬件配置】节点下缺省有 LK220S 控制器模块，控制器默认添加以太网适配器。

通过右键菜单相关命令，可以查看控制器设备信息、进行重命名和设备添加。

7.2 组态 DP 协议

7.2.1 添加 DP 主站设备

通过 LK220S 右键菜单【添加设备】命令，添加 LK249S 安全型主站通信模块。

弹出“添加”对话框，如图 49 所示。选择 LK249S 进行添加。

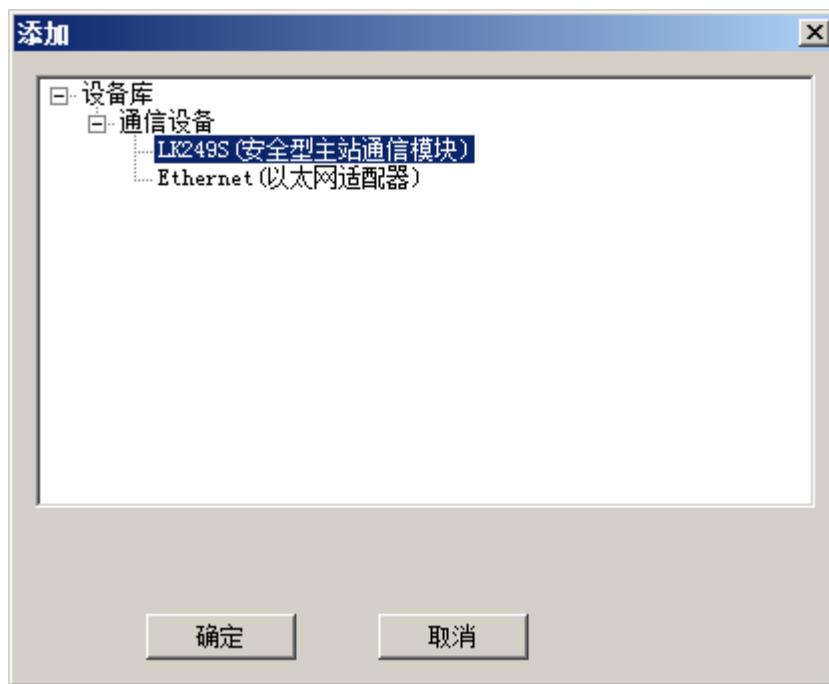


图 49 添加 LK249S 模块



- LK220S CPU 模块目前最多支持 1 个安全型 DP 主站通信模块和 1 个以太网适配器。

通过 LK249S 右键菜单，可以查看模块的设备信息、进行重命名、协议的添加和模块删除操作。

7.2.2 添加通讯协议

LK249S 可作为 PROFIsafe/Profibus-DP 通信主站，组态时，需要添加主站通信协议。在 LK249S 模块的右键菜单中选择【添加协议】命令，进行通讯协议的添加。

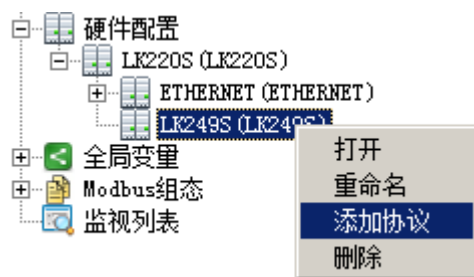


图 50 选择添加协议命令

弹出“添加”对话框，如图 51 所示。

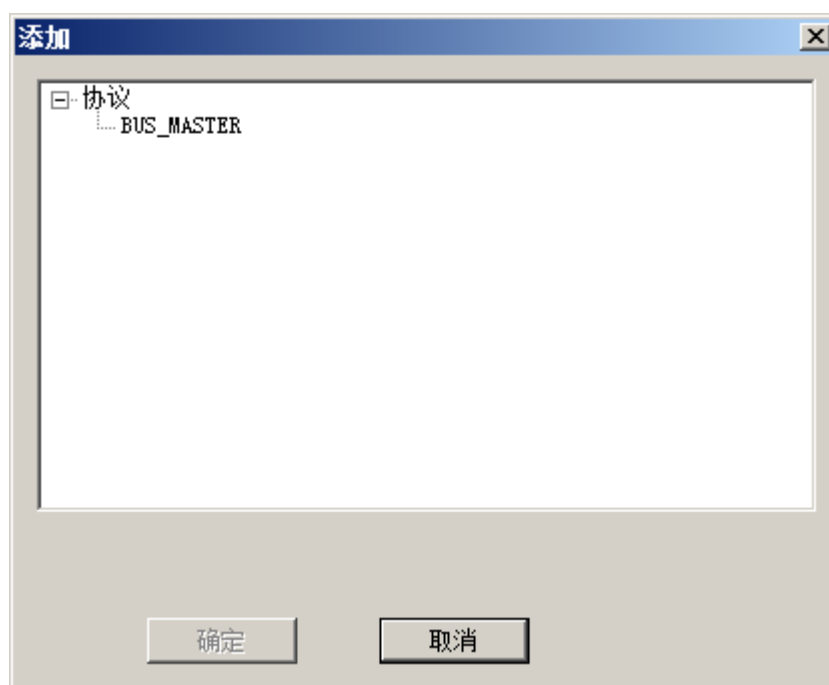


图 51 添加 BUS_MASTER 协议



- 一个 DP 主站通信设备下最多支持一个 BUS_MASTER 协议。

通过 BUS_MASTER 节点的右键菜单，可以查看模块的设备信息、重命名、添加设备以及模块删除操作。

双击 BUS_MASTER 节点，在右侧区域打开该模块的设备信息窗口。

BUS_MASTER (BUS_MASTER)	
设备信息	
项目	内容
协议名称	BUS_MASTER
端口	COM口
波特率	500.00
TSL (0~65535)	200
最小站延时 (0~65535)	11
最大站延时 (0~65535)	100
发送器失败/中继器切换时间 (0~255)	0
建立时间 (0~255)	1
目标循环时间	3416
GAP更新因数	10
重试最大次数 (0~255)	1
最小从站间隔 (0~65535)	50
主站对主站最大请求时间	500
全局总线状态在双口RAM中保存所需最小E	600
DP轮询周期	0
协议	Profibus-DP/PROFIsafe
从站地址范围	2~125
主/从站位置	Profibus-DP 主站/PROFIsafe主站

图 52 BUS_MASTER 信息窗口

在设备信息窗口中，显示通讯主站的参数信息。可以设置端口和波特率。端口选择为 COM 口。设置波特率时，选择不同的波特率，其它总线参数自动切换为最佳值。

7.2.3 添加 PROFIsafe/Profibus-DP 从站

安全型 DP 主站设备下可以添加 PROFIsafe 安全型从站设备也可以添加 Profibus-DP 非安全型从站设备。

7.2.3.1 导入设备描述文件

LK249S 支持第三方设备作为 DP 从站，组态第三方设备前，需要导入设备描述文件，在设备库中生成相应的设备供用户组态使用。

模块对应的设备描述文件均作为系统文件，用户不可任意修改或删除。

导入新的设备描述文件时，需要保证所导入文件的格式和内容正确无误。



- 菜单栏：单击【工程】—【导入设备描述文件...】。

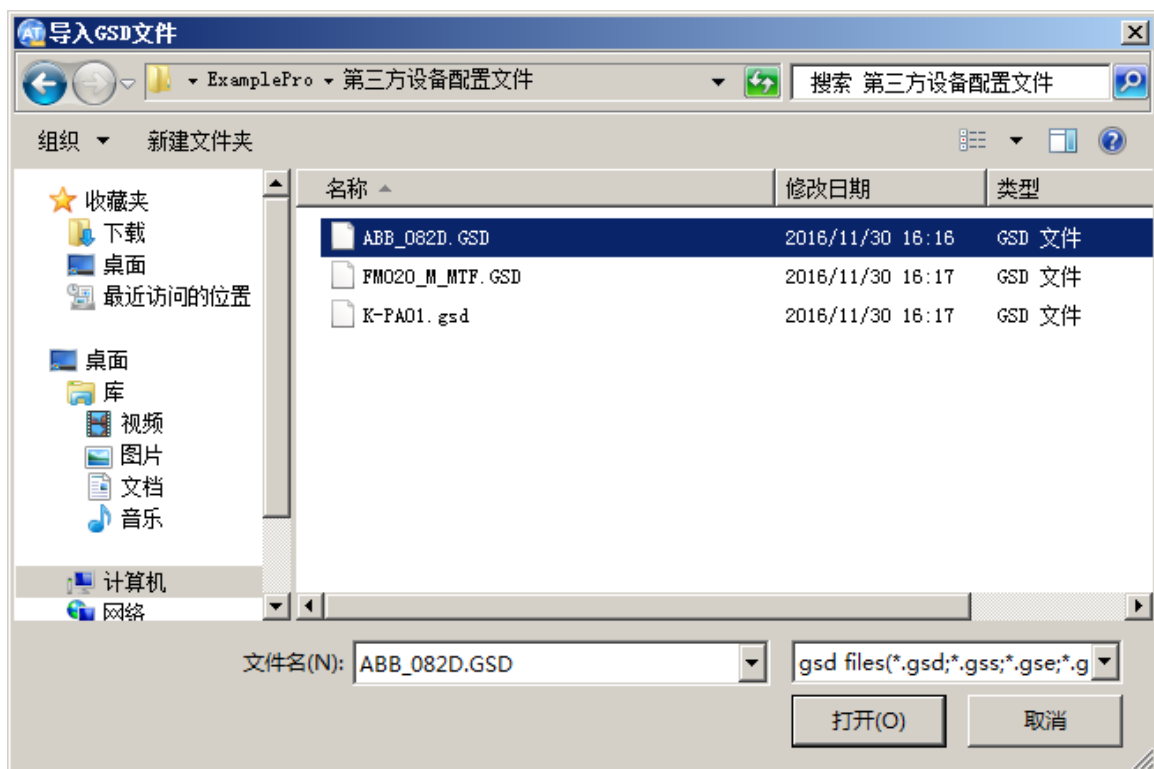


图 53 添加设备描述文件对话框

选择需要导入的 GSD 文件，单击**打开**，该设备被成功导入设备库。

7.2.3.2 添加 PROFIsafe/Profibus-DP 设备

通过【添加设备】命令，进行从站设备的添加，如图 54 所示。



图 54 选择添加设备命令

弹出“添加”对话框，如图 55 所示。Profibus-DP 协议为非安全协议，PROFIsafe 协议为安全协议，根据当前系统是否为安全系统来选择相应的协议模块。地址框中显示默认的从站地址，可进行修改。

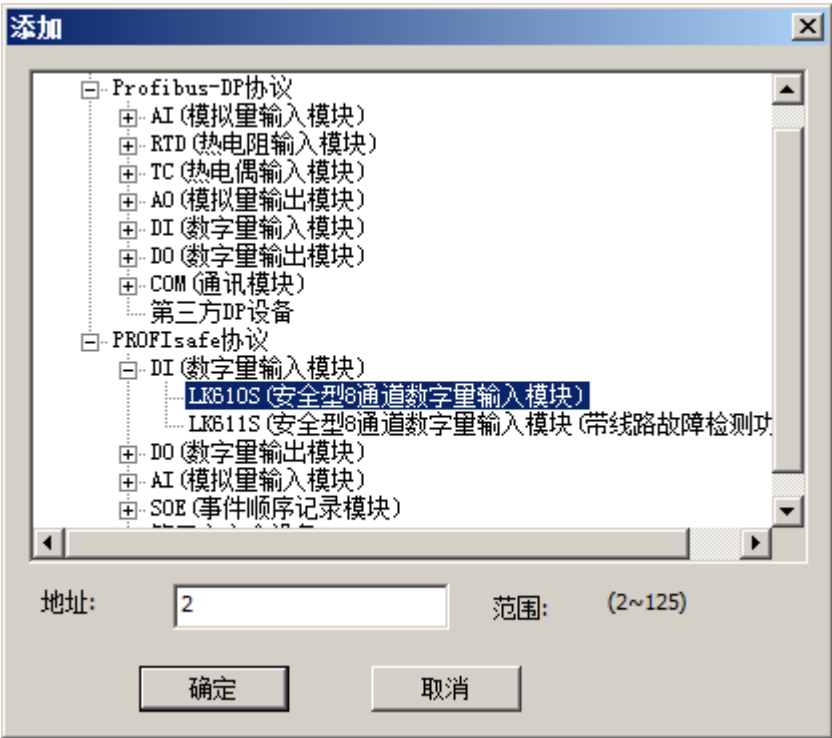


图 55 选择模块

从站命名组成：从站名称（从站地址：模块名称），对于同一模块重复添加时，会在从站名称的后面显示添加的序号。用户可通过右键菜单的【重命名】项修改从站名。

7.2.4 修改从站地址

在从站设备信息窗口中，双击设备地址项蓝色区域，打开地址设置对话框。如图 56 所示。

LK610S (2:LK610S)	
设备信息	
项目	内容
模块型号	LK610S
GSD文件名	LK610S.gsd
设备描述	安全型8通道数字量输入模块
设备地址	2 (双击配置)
是否组态冗余	否
输入起始地址	IB0
输出起始地址	未配置
设备属性	双击配置
通道数	8DI
最大功耗	50mA@24VDC
工作环境温度	0~60℃

(a)



(b)

图 56 修改从站地址

输入新地址，地址范围 2~125。输入的地址不合法或地址重复时，**确定**按钮不可用。

7.2.5 从站参数配置

双击设备属性项（图 56(a)中蓝色区域），打开“模块属性”对话框，如图 57 所示。

该对话框包含【输入/输出选择】和【用户参数】标签页。

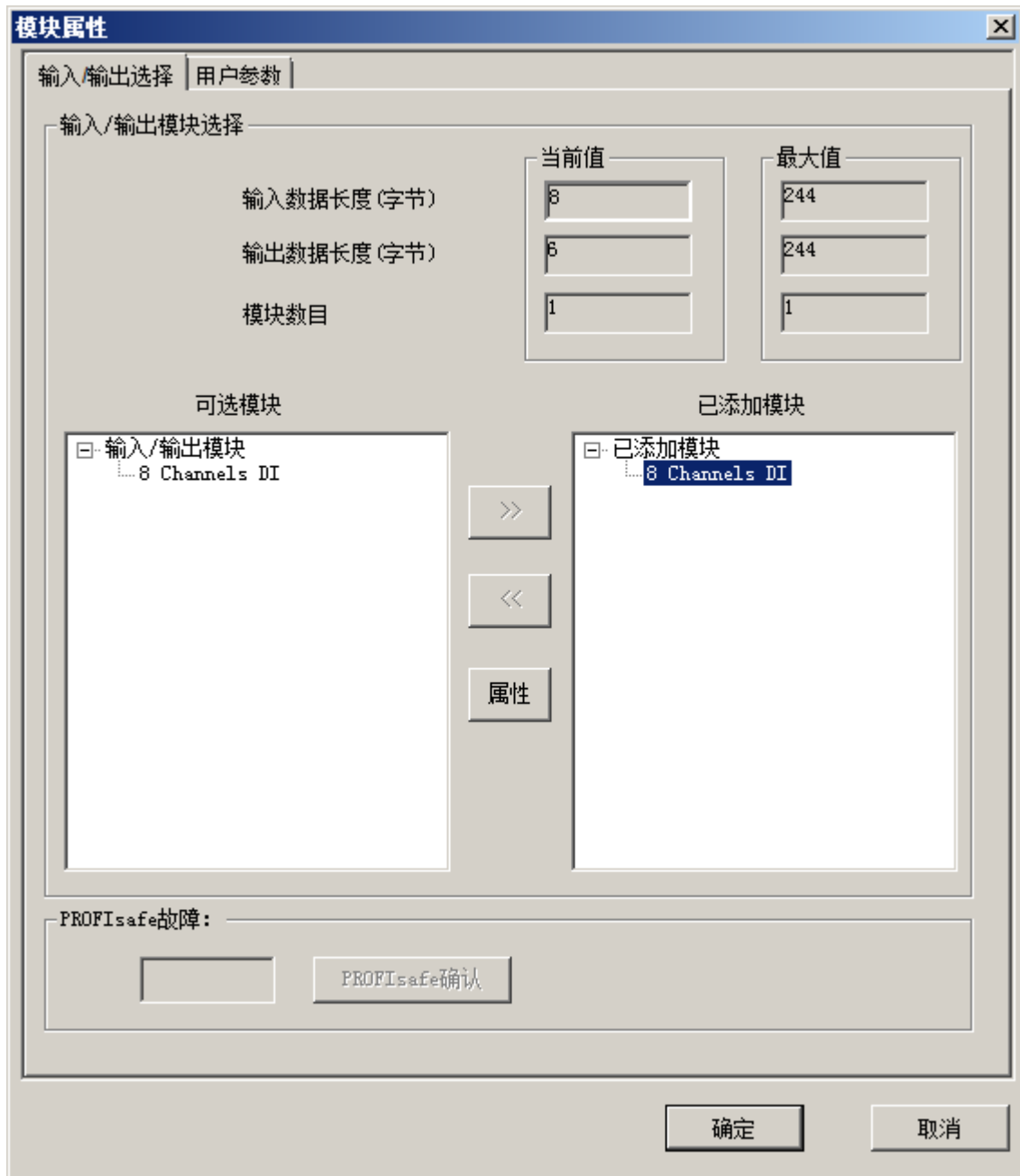


图 57 输入/输出选择标签页

7.2.5.1 子模块属性设置

通过属性查看和配置【已添加模块】中的子模块属性。



图 58 子模块属性窗口

详细参数设置请参见《LKS 安全型控制系统产品手册》第 7 章 IO 模块的参数说明。

7.2.5.2 PROFIsafe 故障状态

安全模块 LK610S、LK611S、LK710S、LK411S、LK630S 具有 PROFIsafe 故障确认功能。详细内容请参见《LKS 安全型控制系统产品手册》第 7 章的对应内容。

7.2.5.3 用户参数设置

【用户参数】标签页显示用户参数信息，用户可对参数进行配置。

各模块可配置的用户参数不同，这里以 LK610S 模块为例进行说明，其它模块用户参数配置请参见设备库各模块介绍。



图 59 用户参数设置

- CHn State: 通道使能状态显示，通过勾选**通道使能**项进行设置；

8 Channels DI							
通道号	通道名称	通道类型	通道地址	通道说明	通道使能	安全值模式	安全值
1	DPIO_2_1_2_1	BOOL	%IX0.0		☒	保持	
2	DPIO_2_1_2_2	BOOL	%IX0.1		☒	保持	
3	DPIO_2_1_2_3	BOOL	%IX0.2		☒	保持	
4	DPIO_2_1_2_4	BOOL	%IX0.3		☒	保持	
5	DPIO_2_1_2_5	BOOL	%IX0.4		☒	保持	
6	DPIO_2_1_2_6	BOOL	%IX0.5		☒	保持	
7	DPIO_2_1_2_7	BOOL	%IX0.6		☒	保持	
8	DPIO_2_1_2_8	BOOL	%IX0.7		☒	保持	

图 60 通道使能设置

- CHn Filter Time: 通道滤波时间。各模块滤波参数值不同，参见实际组态模块可选范围。
- Reserved User Parameters n: 预留参数。

7.3 组态 Modbus TCP 协议

7.3.1 Modbus TCP 主站协议

当控制器作 Modbus TCP 主站时，需要进行以下组态。

7.3.1.1 添加主站协议

在【ETHERNET】节点的右键菜单中选择【添加协议】命令，弹出“添加”对话框。选择 MODBUSTCP_MASTER 主站协议进行添加。

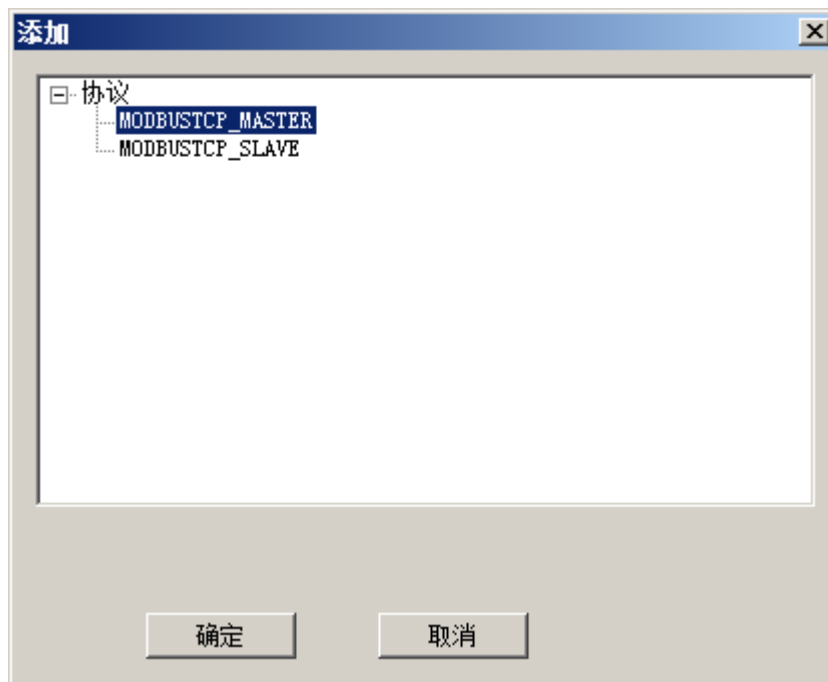


图 61 Modbus TCP 主站协议



- Modbus TCP 主站协议为非安全协议，仅在非安全属性工程中可用。

7.3.1.2 配置 Modbus TCP 主站参数

双击 MODBUSTCP_MASTER 节点打开主站配置窗口。

图 62 Modbus TCP 主站配置

表 7 主站参数

参数	参数值	默认值	说明
响应超时时间 (ms)	10~2,147,483,000	1000	主站发送请求帧后所允许的从站延时应答时间
Socket 超时时间 (ms)	10~2,147,483,000	100	TCP/IP 连接 Socket 超时时间
重试次数	0~10	0	应答异常后主站重新发送请求的次数
轮询间隔 (ms)	100~2,147,483,000 0	100	Modbus TCP 主站接收到从站应答帧后到发送下一个请求帧的间隔时间，如果上一帧从站应答超时，则主站可忽略该间隔时间直接发送请求帧

为保证轮询间隔的有效性，建议在组态中注意以下几点：

- 建议与从站通讯时，尽量使用少的指令个数读取从机数据，即每条读指令读取较多的数据。
- 如果组态了从站，请保证主站与从站的通讯链路正常，且从站可正常工作；若从站不存在，请在组态中删除关于此从站的组态。
- 轮询间隔最差情况下的理论计算公式为： $80\text{ms} \times \text{当前组态从站下的读指令条数}$ （注：此为最差值，若所有从站均正常工作情况下，不会达到此值）。

7.3.1.3 配置 Modbus TCP 从站参数

控制器为主站时，可以配置一个或多个从站进行数据通信。在 MODBUSTCP_MASTER 节点的右键菜单中选择【添加设备】命令，在弹出的对话框中选择从站设备，如图 63 所示。

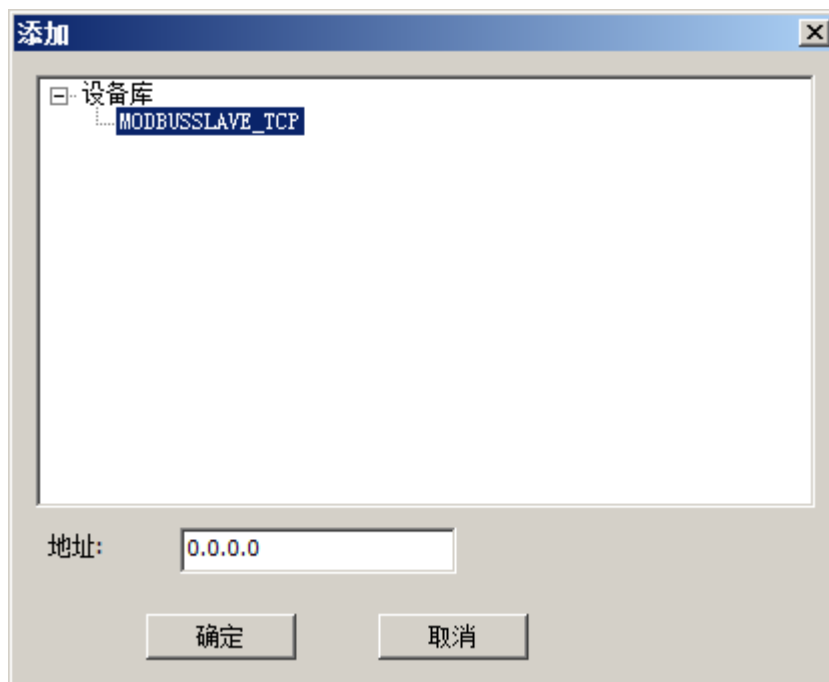


图 63 添加 Modbus TCP 从站

Modbus TCP 主站下可添加 32 个从站，添加的从站缺省显示：从站名（从站地址：设备名称）。

■ Modbus TCP 从站配置

双击 MODBUSSLAVE_TCP 节点打开从站配置窗口。

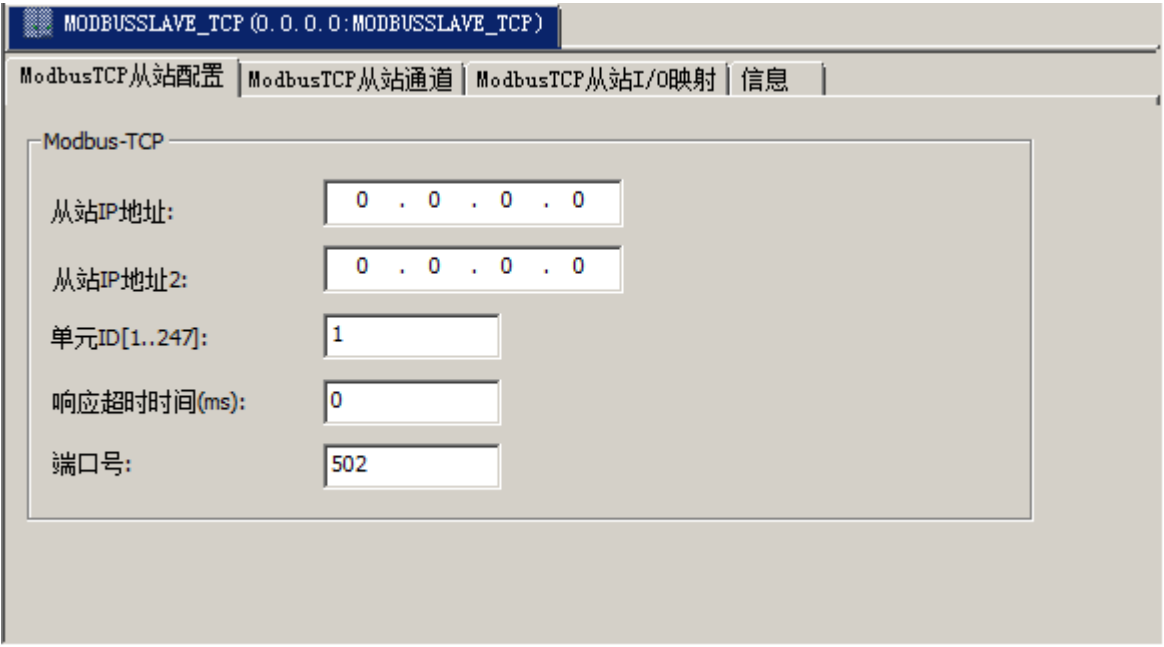


图 64 Modbus TCP 从站配置界面

表 8 从站通讯配置参数

参数	参数值	默认值	说明
从站 IP 地址	根据实际从站 IP 地址进行设置	0.0.0.0	主站请求的从站 IP 地址 可配置两路 IP 地址，选择一路进行连接，另一路作为备用，如第一路连接不成功，则尝试用第二路进行连接。需要注意的是，双路 IP 配置仅在 V1.2.1 及以上版本支持，同时，需要匹配 LK220S-A03 及以上固件版本，该功能才能正常使用
单元 ID	1~247	1	Modbus TCP 协议单元 ID
响应超时时间（ms）	0~2,147,483,000	0	Modbus TCP 主站发送请求帧后所允许的从站延时应答时间 默认情况：从站默认超时时间为零，此时以主站配置的“响应超时时间”为准；用户可单独配置某一个从站的响应超时时间，如果该参数配置大于零，则该从站的超时时间以当前从站配置为准
端口号	1~65,535	502	Modbus TCP 协议端口

■ 从站指令配置

在【Modbus TCP 从站通道】标签页中，可通过右键菜单的【添加】命令为从站添加指令，如图 65 所示。

MODBUS_SLAVE_TCP (0.0.0.0:MODBUS_SLAVE_TCP)						
Modbus TCP从站配置		Modbus TCP从站通道		Modbus TCP从站I/O映射		
Modbus TCP从站配置		Modbus TCP从站通道		Modbus TCP从站I/O映射		
名称	访问类型	错误处理	读偏移	读长度	写偏移	写长度
Channel 0	读线圈 (0xxxx, 01H)	保持	0	2		
Channel 1	读离散量输入 (1xxxx, 02H)	保持	0	3		
Channel 2	读保持寄存器 (4xxxx, 03H)	保持	0	3		
添加 删除 编辑						

(a)



(b)

图 65 添加从站指令

在**可选指令**列表框中选择指令，指令属性列表框中会显示相关参数，双击参数值项进行设置。最多可添加 32 个指令。

表 9 指令参数

参数	参数值	默认值	说明
错误处理	保持、清零	保持	保持：在应答异常后保持当前数据

参数	参数值	默认值	说明
			清零：在应答异常后将当前数据清零
读/写偏移	0~65,535	0	从站上对应的起始地址偏移值
读/写长度	1~2000	1	长度数值对应从站相应的通道数量

■ 从站 I/O 映射

配置指令后，在【Modbus TCP 从站 I/O 映射】标签页中会映射相应的 I/O 通道。

- 每条指令的 Modbus 起始地址=00001+指令读/写偏移地址。
- 通道名称：初始值为 TCPIO_设备序号_协议序号_四位从站 IP 地址_通道号；用户可以自定义，通过使用通道名称来访问 Modbus 变量。
- 通道地址：系统自动分配，不支持修改。

MODBUS_SLAVE_TCP (0.0.0.0:MODBUS_SLAVE_TCP)					
Modbus TCP从站配置 Modbus TCP从站通道 Modbus TCP从站I/O映射 信息					
通道号	Modbus地址	通道名称	通道类型	通道地址	通道说明
Channel 0					
1	000001	TCPIO_1_1_0_0_0_0_1	BOOL	%IX4.0	读线圈 (0xxxx, 01H)
2	000002	TCPIO_1_1_0_0_0_0_2_1	BOOL	%IX4.1	读线圈 (0xxxx, 01H)
Channel 1					
3	100001	TCPIO_1_1_0_0_0_0_3_1	BOOL	%IX8.0	读离散量输入 (1xxxx, 02H)
4	100002	TCPIO_1_1_0_0_0_0_4	BOOL	%IX8.1	读离散量输入 (1xxxx, 02H)
5	100003	TCPIO_1_1_0_0_0_0_5	BOOL	%IX8.2	读离散量输入 (1xxxx, 02H)
Channel 2					
6	400001	TCPIO_1_1_0_0_0_0_6	WORD	%IW12	读保持寄存器 (4xxxx, 03H)
7	400002	TCPIO_1_1_0_0_0_0_7	WORD	%IW14	读保持寄存器 (4xxxx, 03H)
8	400003	TCPIO_1_1_0_0_0_0_8	WORD	%IW16	读保持寄存器 (4xxxx, 03H)

图 66 从站 I/O 通道

7.3.1.4 Modbus 指令诊断

当组态了 Modbus 主站指令后，编译通过，在变量组 ModbusOrderDiagVar 中生成指令的诊断变量，如图 67 所示。该变量组参数不支持编辑。用户可以根据变量说明列显示的诊断信息描述来查找具体指令。

ModbusOrderDiagVar					
序号	变量名	变量说明	变量类型	初始值	区域
0001	SYS_OrderDiag_0	第1个通信设备 第1条协议 地址0.0.0.0 第1条指令的状态诊断	WORD	0	S区
0002	SYS_OrderDiag_2	第1个通信设备 第1条协议 地址0.0.0.0 第2条指令的状态诊断	WORD	0	S区
0003	SYS_OrderDiag_4	第1个通信设备 第1条协议 地址0.0.0.0 第3条指令的状态诊断	WORD	0	S区

图 67 Modbus 变量诊断信息

在线时，通过指令状态的码值可以诊断指令状态。

表 10 Modbus TCP 主站指令状态码值定义

指令状态码值	含义
--------	----

指令状态码值	含义
0	无错误
2	超时
4	功能码错误
16	单元 ID 不匹配（从站地址错误）
32	TCP 连接失败
33	发送请求报文失败
34	接收确认报文失败
64	发送的数据与接收的数据不匹配
128	单元 ID 不匹配
128+1	从站不支持的功能码
128+2	数据地址溢出
128+3	数据范围溢出
128+4	从站设备故障
128+6	从站设备忙
128+15	从站其它故障

7.3.2 Modbus TCP 从站协议

在安全属性工程中，LK220S 只能做 Modbus TCP 从站。

7.3.2.1 添加从站协议

右击 ETHERNET 树节点，选择【添加协议】命令，弹出协议添加对话框，如图 68 所示。选择从站协议 MODBUSTCP_SLAVE 进行添加。

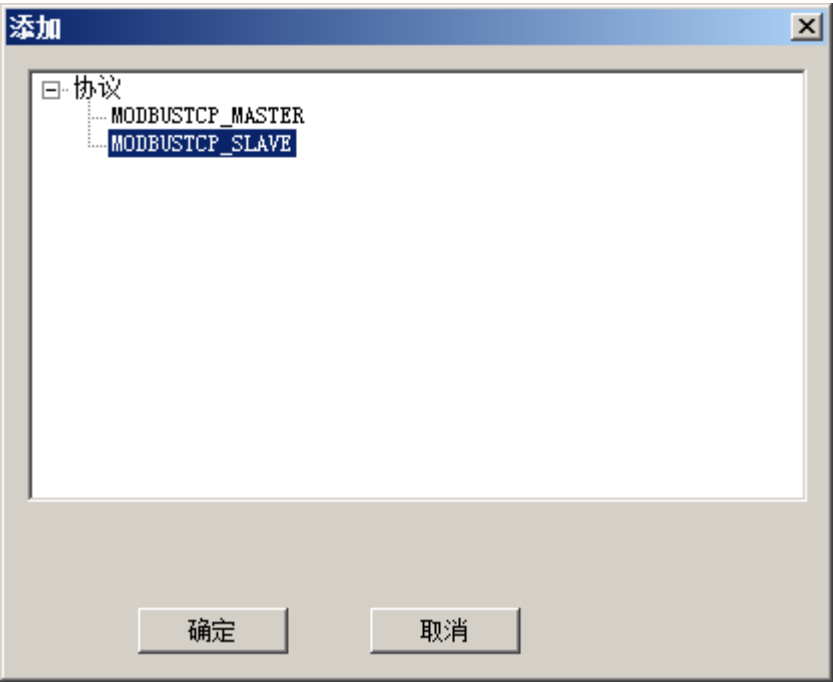


图 68 添加从站协议

7.3.2.2 从站参数配置

双击添加的协议节点，打开从站参数配置界面。



图 69 从站配置界面

表 11 配置参数

参数	参数值	默认值	说明
超时时间（ms）	500~2,147,483,000	2000	从站没有接收到主站发送数据的时间间隔，超过设定的时间，则从站断开与主站的通讯链接

参数	参数值	默认值	说明
从站端口	1~65,535	502	Modbus TCP 协议端口号
单元 ID	1~247	1	Modbus TCP 协议单元 ID

■ RUN 状态时 Modbus 读写属性设置

- ☐ 只读：安全系统中默认为只读。
- ☐ 读写：为非安全设置，可对从站 Modbus 数据进行读写操作。

7.3.2.3 Modbus 变量组态

进行 Modbus 通讯的变量需要在【Modbus 组态】中进行定义，如图 70 所示。变量定义分为四个组：线圈、输入离散量、输入寄存器和保持寄存器，各个组中关联的变量类型也不尽相同。



图 70 Modbus 变量区

在【Modbus 组态】节点下，不同变量组中添加不同数据类型的变量，地址对应关系和填写范围见表 12。

表 12 Modbus 通讯变量设置限制

存储区域	数据类型	存取权限	区内偏移	Modbus 地址
线圈	BOOL	读写	1~65535	000001~065535
输入离散量	BOOL	只读	1~65535	100001~165535
输入寄存器	WORD\DINT\DWORD\REAL	只读	1~65535	300001~365535
保持寄存器	WORD\DINT\DWORD\REAL	读写	1~65535	400001~465535

双击【Modbus 组态】的子节点，打开变量组对话框。如图 71 所示。

线圈						
序号	变量组名	变量名	区内偏移	Modbus 地址 (0xxxxx)	变量类型	在线值
0001	Main	g1	12	000012	BOOL	
0002	Main	g2	13	000013	BOOL	
0003	Main	g3	14	000014	BOOL	

图 71 Modbus 变量编辑窗口

Modbus 变量的添加方式有四种：

(1) 手动添加

通过编辑窗口右键菜单中的【增加变量】命令进行添加。

(2) 通过命令添加

通过【发送到】命令进行添加，对于 PRG 类型 POU 中的变量、【全局变量】组中的变量、通道变量都可以通过变量右键菜单中的【发送到】命令，将变量添加到对应的 Modbus 变量组中。

(3) 复制变量到 Modbus 变量组

复制 PRG 或者【全局变量】组中的变量，粘贴到 Modbus 变量组。

Modbus 变量组中对应的变量属性如下：

■ 变量组名：发送过来的变量组名显示如下。

POU 局部变量：变量所在的 POU（PRG）名。

全局变量组中的变量：【全局变量】节点下的变量组名。

通道变量：显示 Physical。

■ 变量名：

发送过来的变量显示原变量名。

功能块的 BOOL/WORD 类型点项：功能块名称.项名。

■ 区内偏移：按照主站设置的从站访问地址及寄存器类型，在对应的变量组中定义区内偏移，主站通过 Modbus 地址访问从站的 Modbus 通讯数据。对从站进行读写操作。

设置说明：

☐ DWORD、DINT、REAL 型变量的区内偏移地址长度为 2 个字，设置区内偏移地址：n+2，n 为当前已占用地址。如：REAL 型变量 g1、g2，设置 g1 偏移地址为 1，则 g2 地址设置为 3。

☐ DWORD、DINT、REAL 型变量的区内偏移地址不能设置为 65535。

■ Modbus 地址：由寄存器类型和区内偏移组成。

寄存器类型分为以下四种：

☐ 0：线圈

☐ 1：输入离散量

☐ 3：输入寄存器

☐ 4：保持寄存器

(4) 导入和导出 Modbus 变量

在首次批量导入 Modbus 变量之前，需先从软件中导出 Modbus 变量的模板表格。

■ 导出 Modbus 变量



- 菜单栏：单击【工程】—【导出 Modbus 变量】。



图 72 模板表格

■ 导入 Modbus 变量

将需要进行 Modbus 通讯的变量填入表格后，通过【导入 Modbus 变量】命令将变量表导入到工程中。

填写表格时，每个 sheet 页中的**变量组名**、**变量名**、**区内索引**必填项。变量导入时是清空式导入，即软件会先清空原有的 Modbus 变量，后导入新变量。



- 菜单栏：单击【工程】—【导入 Modbus 变量】。

导入结果会在【信息窗口】中显示。

7.4 系统诊断

在线时，用户可以通过【全局变量】节点下的变量组 SysDiagVar 查看双机运行状态以及系统故障诊断。

序号	变量名	变量说明	变量类型	初始值	区域
0001	sys_LocalMSState	本机主从状态	WORD	0	S区
0002	sys_LocalRSSState	本机单双机状态	WORD	0	S区
0003	sys_LocalABState	本机AB机状态	WORD	0	S区
0004	sys_LocalRedLink	本机冗余模块工作网状态	WORD	0	S区
0005	sys_LocalTaskState	本机任务运行状态	WORD	0	S区
0006	sys_LocalKeyState	本机钥匙开关状态	WORD	0	S区
0007	sys_RemoteMSState	对方机主从状态	WORD	0	S区
0008	sys_RemoteRSSState	对方机单双机状态	WORD	0	S区
0009	sys_RemoteABState	对方机AB机状态	WORD	0	S区
0010	sys_RemoteRedLink	对方机冗余模块工作网状态	WORD	0	S区
0011	sys_RemoteTaskState	对方机任务运行状态	WORD	0	S区
0012	sys_RemoteKeyState	对方机钥匙开关状态	WORD	0	S区
0013	sys_TaskFirstRun	下装后任务第一次运行状态	WORD	0	S区
0014	sys_ModuleState	本机模块存在状态	WORD	0	S区
0015	sys_TaskCycleAbnormal	任务周期设定与当前组态工程不匹配标识	WORD	0	S区
0016	sys_CFUModuleFatalErr1	CFU模块严重性故障1	WORD	0	S区
0017	sys_CFUModuleFatalErr2	CFU模块严重性故障2	WORD	0	S区
0018	sys_CFUModuleMinorErr1	CFU模块一般性故障1	WORD	0	S区
0019	sys_CFUModuleMinorErr2	CFU模块一般性故障2	WORD	0	S区
0020	sys_REDModuleFatalErr1	RED模块严重性故障1	WORD	0	S区
0021	sys_REDModuleFatalErr2	RED模块严重性故障2	WORD	0	S区
0022	sys_REDModuleMinorErr1	RED模块一般性故障1	WORD	0	S区
0023	sys_REDModuleMinorErr2	RED模块一般性故障2	WORD	0	S区
0024	sys_DFMModuleFatalErr1	DF模块严重性故障1	WORD	0	S区
0025	sys_DFMModuleFatalErr2	DF模块严重性故障2	WORD	0	S区
0026	sys_DFMModuleMinorErr1	DF模块一般性故障1	WORD	0	S区
0027	sys_DFMModuleMinorErr2	DF模块一般性故障2	WORD	0	S区
0028	sys_ReservedDiag	系统保留诊断信息	ARRAY[0..31] OF WORD		S区

图 73 SysDiagVar 诊断变量组

表 13 各变量诊断信息说明

变量	变量说明	上报的诊断值
sys_LocalMSState	本机主从状态	0: 未知态 1: 初始态 2: 硬件 ready 态 3: 双机 ready 态 4: 单机 ready 态 5: 故障态 6: 错误态 7: 工程冗余 8: 工程验证 9: 从机 10: 主机 11: 不合格主机 12: 等待进入故障态 13: 等待进入错误态
sys_LocalRSSState	本机单双机状态	0: 未知态

变量	变量说明	上报的诊断值
		1: 单机 2: 双机
sys_LocalABState	本机 AB 机状态	0: 未知态 1: A 机 2: B 机
sys_LocalRedLink	本机冗余模块工作网状态	0: 未知态 1: 第一路光纤 2: 第二路光纤
sys_LocalTaskState	本机任务运行状态	每一个任务的状态使用 2 bit 位表示 0: 未知态 1: 正在运行 2: 已停止 3: 挂起
sys_LocalKeyState	本机钥匙开关状态	0: UNKNOWN 1: RUN 2: REMOTE 3: PRG
sys_RemoteMSState	对方机主从状态	同本机上报值
sys_RemoteRSSState	对方机单双机状态	同本机上报值
sys_RemoteABState	对方机 AB 机状态	同本机上报值
sys_RemoteRedLink	对方机冗余模块工作网状态	同本机上报值
sys_RemoteTaskState	对方机任务运行状态	同本机上报值
sys_RemoteKeyState	对方机钥匙开关状态	同本机上报值
sys_TaskFirstRun	下装后任务第一次运行状态	每一个任务的状态使用 1 bit 位表示 0: 非第一次运行 1: 第 1 次运行
sys_ModuleState	本机模块存在状态	(通信模块在线状态) bit0~bit3: 槽位 2 模块 ID bit4~bit7: 槽位 3 模块 ID bit8~bit11: 槽位 4 模块 ID, 预留 bit12~bit15: 槽位 5 模块 ID, 预留 模块 ID 为 0: 代表该槽位未插入模块 模块 ID 为 7: LK240S 在线 模块 ID 为 8: LK249S 在线
sys_TaskCycleAbnormal	任务周期设定与当前组态工程不匹配标识	0: 任务周期设定与当前组态工程匹配 1: 任务周期设定与当前组态工程不匹配
sys_CPUModuleFatalErr1	CPU 模块严重性故障 1	Bit2=1: 双以太网故障 Bit7=1: 冗余超时故障 Bit8=1: 冗余状态故障 (针

变量	变量说明	上报的诊断值
		对冗余配置下 240S 离线情况)
sys_CPUModuleFatalErr2	CPU 模块严重性故障 2	预留
sys_CPUModuleMinorErr1	CPU 模块一般性故障 1	Bit0=1: 历史不可运行故障 Bit6=1: RTC 供电电压异常 Bit7=1: 任务周期超时故障
sys_CPUModuleMinorErr2	CPU 模块一般性故障 2	预留
sys_REDModuleFatalErr1	RED 模块严重性故障 1	Bit0=1: 自检故障
sys_REDModuleFatalErr2	RED 模块严重性故障 2	预留
sys_REDModuleMinorErr1	RED 模块一般性故障 1	Bit0=1: 第 1 条光纤链路故障 Bit1=1: 第 2 条光纤链路故障 Bit2=1: 双光纤链路故障 Bit3=1: AB 系开关冲突 Bit5=1: 光纤通信故障
sys_REDModuleMinorErr2	RED 模块一般性故障 2	预留
sys_DPMModuleFatalErr1	DP 模块严重性故障 1	Bit0=1: 检测到 DP 链路存在主机无法进入工作主站故障 Bit4=1: DP1、DP2 网故障
sys_DPMModuleFatalErr2	DP 模块严重性故障 2	预留
sys_DPMModuleMinorErr1	DP 模块一般性故障 1	Bit3=1: DP 通信故障 Bit4=1: DP1 网故障 Bit5=1: DP2 网故障
sys_DPMModuleMinorErr2	DP 模块一般性故障 2	预留
sys_ReservedDiag	系统保留诊断信息	预留

7.5 硬件列表

表 14 硬件版本与软件版本对应关系

模块型号	模块名称	对应的软件版本
LK130-B	4 槽本地背板模块	V1.0.0
LK117-B	11 槽 LK 系列 PLC 扩展背板	
LK118-B	5 槽 LK 系列 PLC 扩展背板	
LK220S-A01	安全型主控模块	
LK240S-A01	安全型冗余通讯模块	
LK249S-A01	安全型主站通信模块	
LK232S-A01	安全型总线重复器模块	

模块型号	模块名称	对应的软件版本
LK610S-A01	安全型 8 通道数字量输入模块	
LK710S-A01	安全型 8 通道数字量输出模块	
LKA104-B	Profibus-DP 总线连接器	
LK411S-A01	安全型 8 通道电流型模拟量输入模块	V1.0.1
LK220S-A02	安全型主控模块	
LK249S-A02	安全型主站通信模块	
LK610S-A02	安全型 8 通道数字量输入模块	
LK710S-A02	安全型 8 通道数字量输出模块	
LK630S-A01	安全型 8 通道数字量输入（带 SOE 功能）模块	V1.2.1
LK220S-B01	安全型主控模块	V1.2.2
LK249S-A03	安全型主站通信模块	
LK610S-A04	安全型 8 通道数字量输入模块	
LK710S-A05	安全型 8 通道数字量输出模块	
LK630S-A02	安全型 8 通道数字量输入（带 SOE 功能）模块	
LK611S-A01	安全型 8 通道数字量输入模块（带线路故障检测功能）	V1.3.1

第8章 程序组织单元

程序组织单元 POU（Program Organization Unit）是 Safety FA-AutoThink 软件中重要的、唯一的基本编程单位。POU 可以分为程序（Program）、功能块（Function Block）、函数（Function）。

程序 Program：是 Safety FA-AutoThink 软件的逻辑执行主体；是为了完成某项任务而编写的语句序列或一组指令的集合。

一般在程序中定义普通全局变量、映射硬件地址全局变量。通过程序间的调用实现应用逻辑。

功能块 Function Block：是面向对象特性的基本算法单元，功能块有内部状态。

功能块执行时产生一个或多个值。可创建一个功能块的多个命名实例，每个实例应具有一个相关的示例名称和包含其输出和内部变量的一个数据结构，以及与实例有关的输入变量值或输入变量的引用。输出变量和这个数据结构必要的内部变量的所有值，应从此功能块的一次执行保持到下一次执行；因此，输入变量相同的功能块的不一定总是产生相同的输出值。如流量累积。

函数 Function：是有至少一个输入变量、没有内部状态、有一个返回值的基本算法单元。

函数在使用时无实例名，其输出值只跟同一周期的输入值和自身的逻辑有关。如加法。

本章主要介绍 POU 的类型，如何创建 POU 以及 POU 之间的调用关系。

8.1 添加 POU



- 工程管理：右击【用户程序】，单击【添加 POU】。

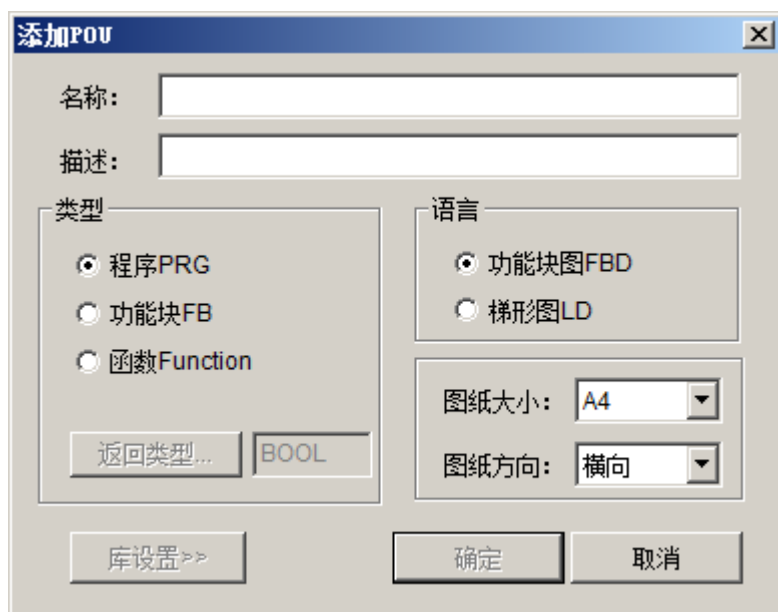


图 74 添加 POU

这个窗口按结构由上到下分为 3 部分：名称、类型、语言。

8.1.1 公共部分

首先，说明三种类型在这个窗口中的公共部分。

8.1.1.1 名称

- 名称：POU 的名称。必填项。

POU 名只能包含字母、数字、下划线“_”，不能以“AT_”、“SYS_”开始，且长度不超过 32 个字符，超出部分无法输入。

POU 名不能与变量名、变量组名、数据类型（自定义或系统缺省的）、关键字、指令库名或功能块名重名。

POU 名不能为 Windows 系统保留的设备名称：CON、PRN、AUX、NUL、COM0~9、LPT0~9。

只有名称符合要求时，**确定**可用；反之，不可用。

- 描述：POU 的注释说明。最多可用 512 字符表述。

8.1.1.2 类型

POU 类型：包括程序块、函数、功能块类型。

8.1.1.3 语言

选择编程语言。

8.1.1.4 图纸大小

确定编辑时的纸张尺寸。缺省为 A4，可选择 A0、A3、A4、B5、Ax（尺寸大小是 4 倍的 A0）。

当纸张类型确定后，系统会自动按照标准尺寸划分编辑区域。

图纸方向：选择图纸的打印方向，缺省为横向打印。

8.1.2 差异部分

当 POU 类型为**函数**时，需要通过**返回类型**设置返回值的数据类型。

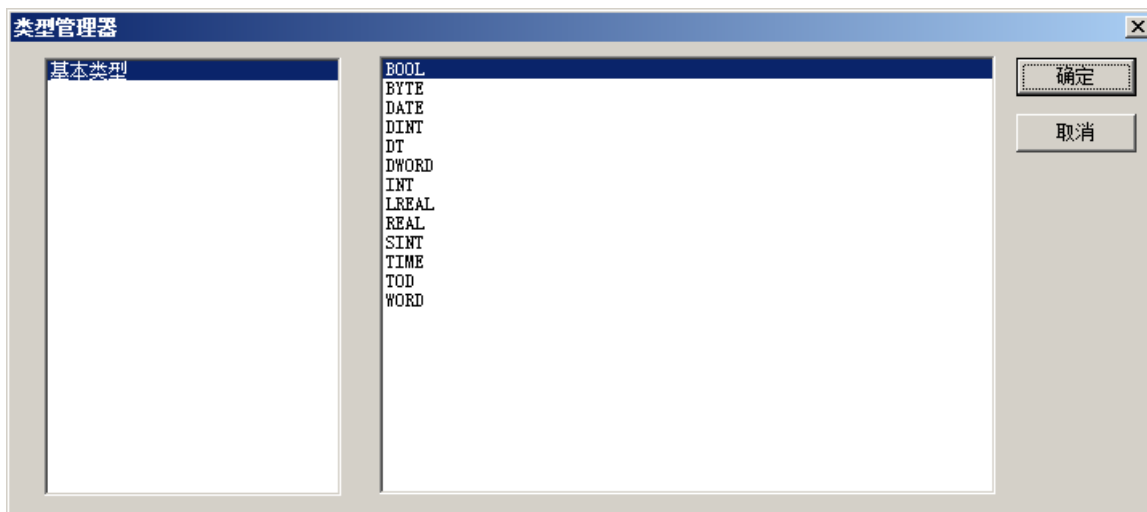


图 75 类型管理器对话框

8.2 编辑 POU



- 工程管理：右击【用户程序】节点下 POU，单击【编辑】。
- 工程管理：双击【用户程序】节点下 POU。

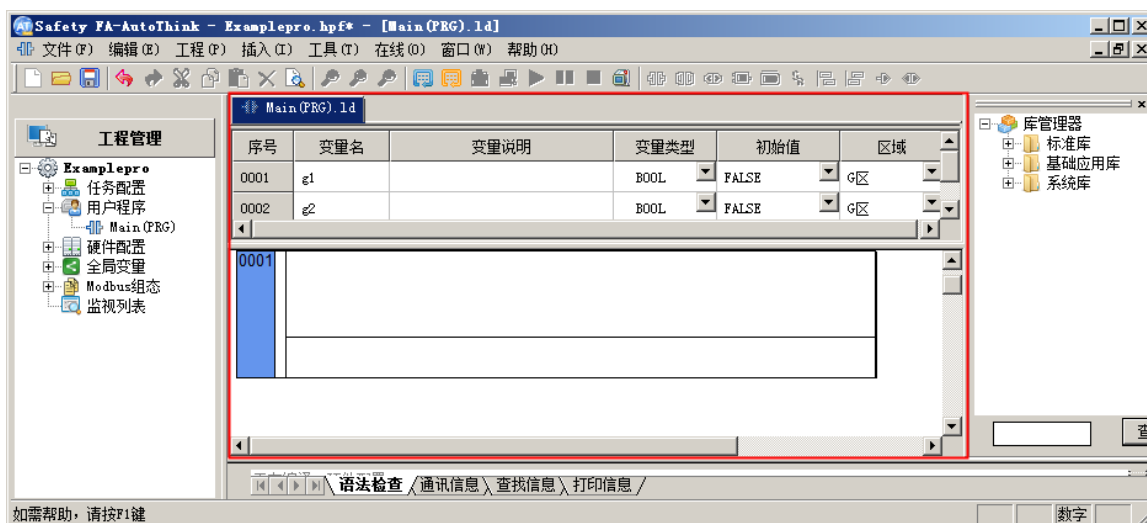


图 76 POU 编辑窗口

POU 包含变量定义和逻辑实现两部分。逻辑实现可以由图形编辑器组成；变量定义通常是表格编辑器。

变量定义和逻辑实现部分通过能拖动的屏幕分割器来分开，通过用鼠标点击它并朝上或朝下移动它到希望的位置。

8.2.1 变量定义

Safety FA-AutoThink 软件提供了丰富的基本变量类型，包括 BOOL、WORD、BYTE、INT、SINT、TIME、DATE、ARRAY、DINT、DT、DWORD、LREAL、REAL、TOD 等等，这些基本的数据类型极大地满足工业控制行业的需求。

在 POU 中，按照功能类型划分，一般包括如下 5 类变量：VAR_POU、VAR_INPUT、VAR_OUTPUT、VAR_IN_OUT、VAR_GLOBAL。

不同 POU 所支持的变量类型及其功能说明见表 15。

表 15 不同 POU 所支持的变量类型及其功能说明

变量类型	功能说明	读写权限		应用范围		
		外部	内部	程序 Program	功能块 Function Block	函数 Function
VAR_POU	局部变量 仅在 POU 内部使用，外部不能访问	不允许	读写	支持	支持	支持
VAR_INPUT	输入变量，可作为 POU 的外部调用参数	读写	读	不支持	支持	支持
VAR_OUTPUT	输出变量，一般作为 POU 的返回值	读	读写	不支持	支持	不支持
VAR_IN_OUT	输入输出变量，兼有输入变量和输出变量的特性	读写	读写	不支持	支持	不支持
VAR_GLOBAL	全局变量	读写	读写	支持	支持	不支持

8.2.2 逻辑实现

Safety FA-AutoThink 软件支持功能强大的 2 种编程语言，且都是图形化的编程界面，工程使用极为方便，思路清晰。

8.3 POU 属性

POU 属性窗口一方面帮助用户了解具体 POU 的基本信息（如使用环境和用途）；另一方面方便用户修改部分内容的设置。

POU 属性窗口可以通过下面方法打开：



- 工程管理：右击【用户程序】下的子 POU，单击【POU 属性】。

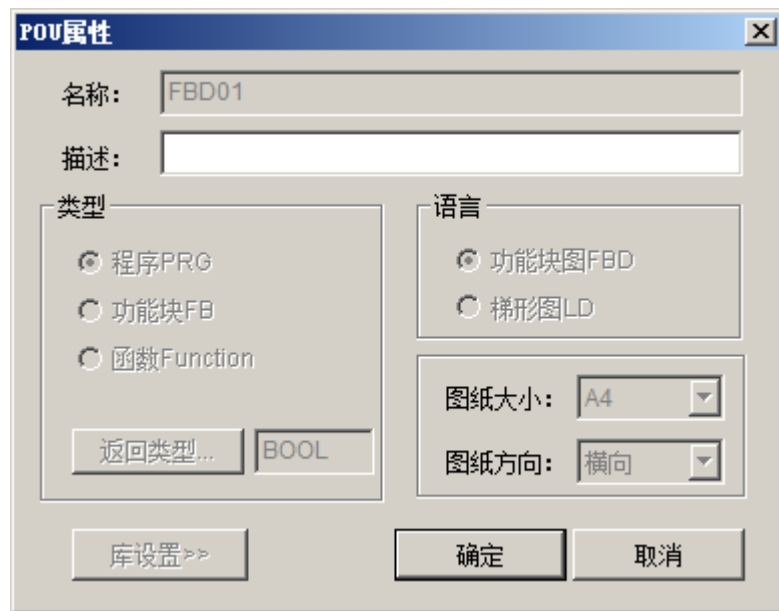


图 77 FBD 语言 POU (PRG) 属性对话框

这个对话框与“添加 POU”对话框的内容结构完全相同，只是其中一些项的可操作权限有差异。相关参数的介绍，请详见 8.1 添加 POU。

8.4 POU 重命名



- 工程管理：右击【用户程序】节点下的 POU，单击【重命名】。

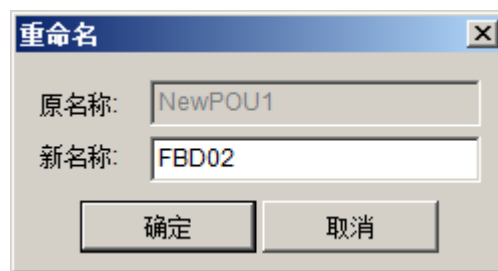


图 78 重命名对话框

在编辑框中输入新的 POU 名称，单击**确定**完成修改。POU 名称的命名规则请详见 8.1 添加 POU。

8.5 复制/粘贴 POU

粘贴结果显示为“NewPOU*”，“*”用正整数表示，当复制的 POU 名在粘贴时会发生重名，“*”的值累积递增。操作说明如下：

右击期望复制的 POU 节点，选择【复制 POU】，如图 79 所示。

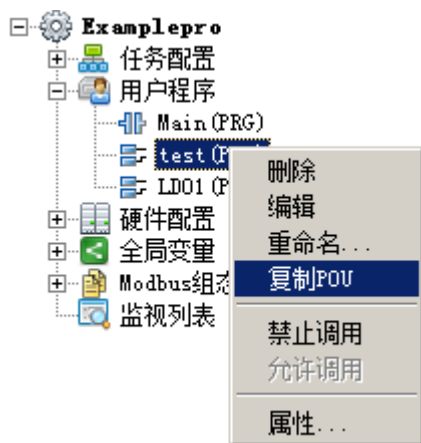


图 79 复制 POU

然后在【用户程序】节点的右键菜单中选择【粘贴 POU】。

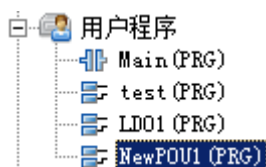


图 80 粘贴

8.6 删除 POU



- 菜单栏：单击【编辑】—【删除 (D)】；
- 工具栏：
- 快捷键：Delete；
- 工程管理：右击【用户程序】下 POU 节点，单击【删除】。

删除的 POU 不可恢复，请操作时注意。

8.7 缩放 POU



- 工具栏：

无论是在线，还是离线状态下，POU 编辑窗口中内容的显示比例都可以调整。

工程关闭时，缩放比例会自动恢复至 100%。

8.8 打印 POU

8.8.1 打印预览

对当前工作区域显示内容的打印效果进行预览。

工程中只能对 POU 进行预览。



- 菜单栏：单击【文件】—【打印预览】。

预览结果会因打印方式的不同而不同。

POU 页面的打印方式包含分页打印和单页打印两种形式。默认是单页打印，设置位置在【工程】—【选项】—【组态语言】。

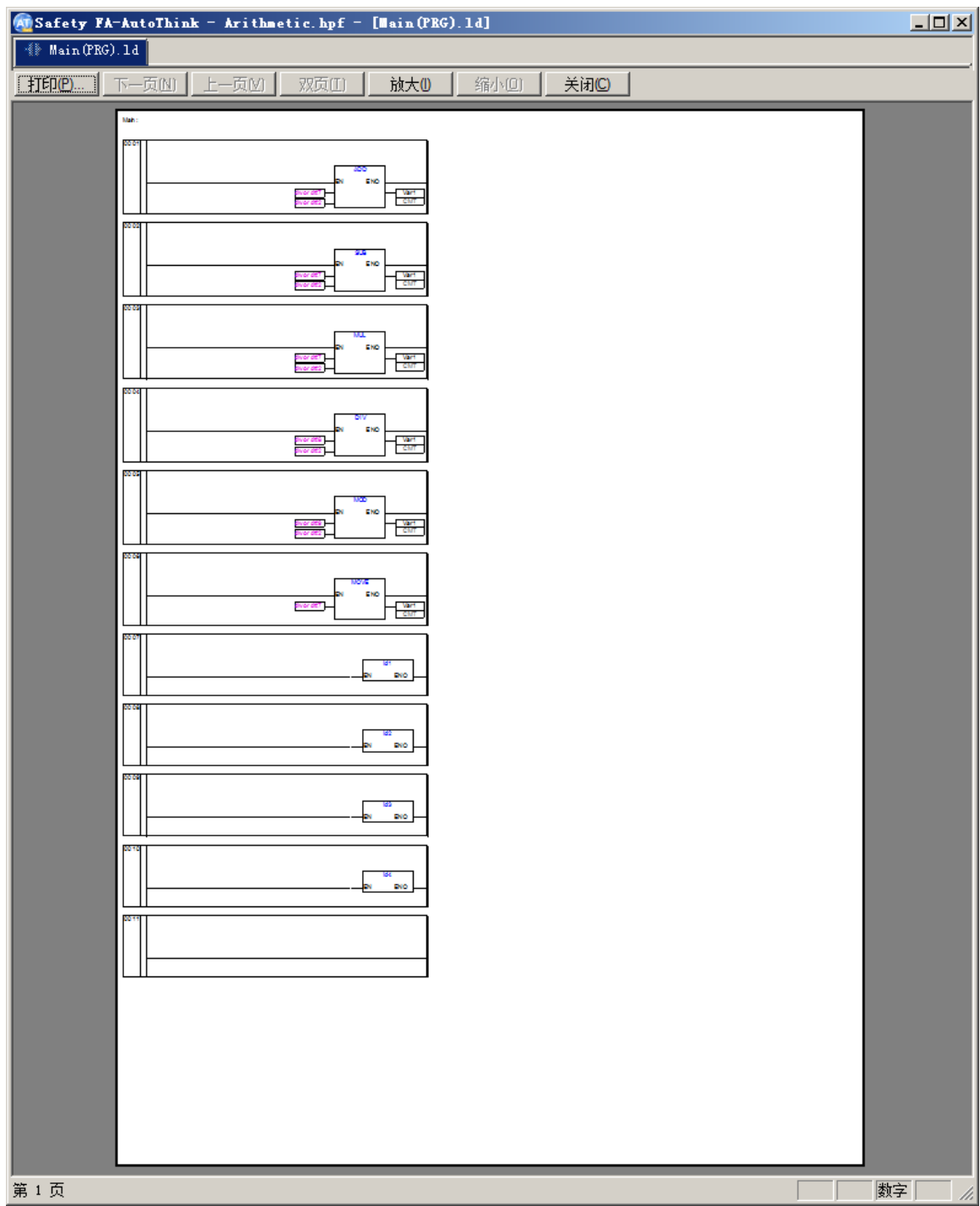


图 81 预览结果

- 打印：用来设置打印条件。
- 上一页和下一页：用来对预览结果进行翻页，当选择单页打印时，该按钮变灰。
- 双页或单页：用来对预览结果进行分栏显示。当预览内容只有一页时，该按钮变灰。
- 放大和缩小：用来调整预览结果的显示比例。
- 关闭：用来退出这个功能。

8.8.2 打印



- 菜单栏：单击【文件】—【打印】；
- 快捷键：Ctrl+P。

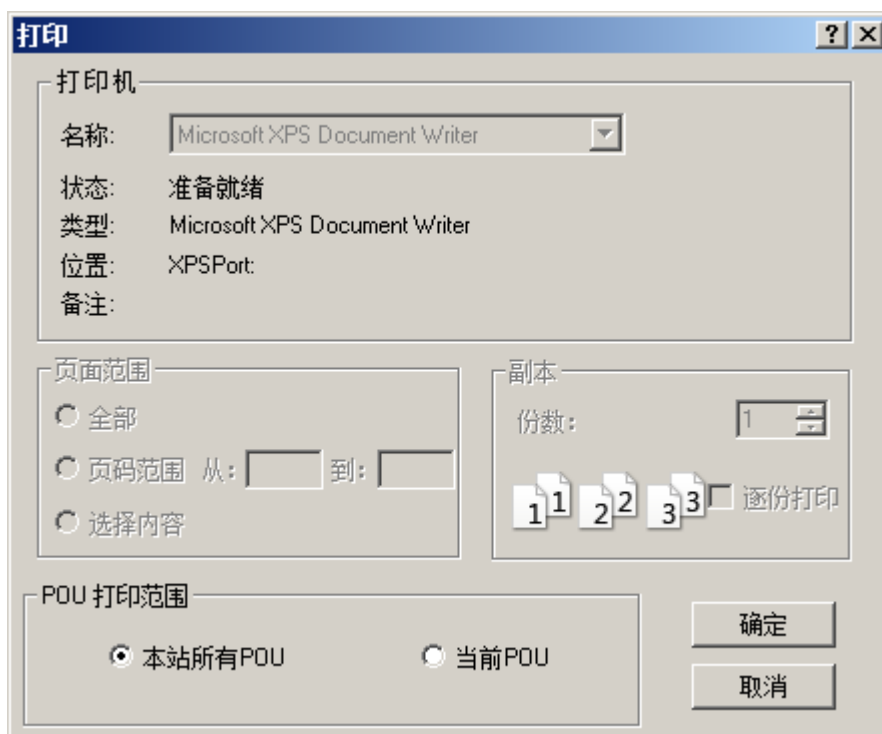


图 82 打印对话框

8.9 查找



- 菜单栏：单击【编辑】—【查找】；
- 工具栏：；
- 快捷键：Ctrl+F。

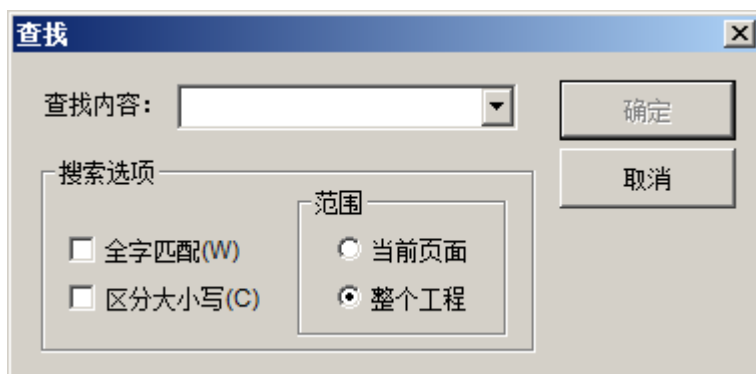


图 83 查找对话框

此命令可以在当前编辑器中查找某一文本。

在“查找内容”输入框中输入要查找的字符序列；设置搜索选项：是否全字匹配、是否区分大小写，搜索整个工程或是当前页面。

单击**确定**按钮后开始查找，查找均从当前页面开始。查找结果显示在信息窗口的【查找信息】栏中。

8.10 替换



- 菜单栏：单击【编辑】—【替换】；
- 快捷键：Ctrl+H。

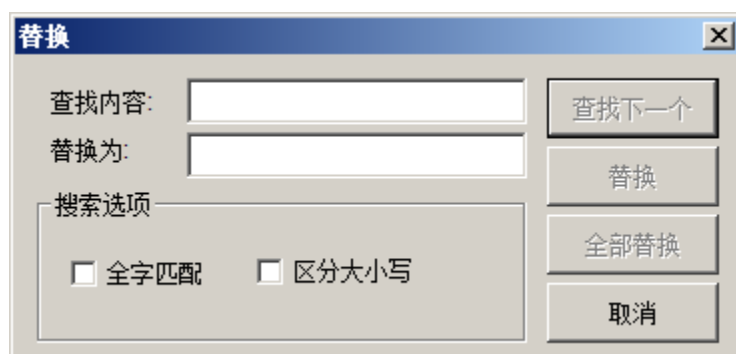


图 84 替换对话框

使用此命令可以快速将一个内容替换另一个内容，如算法中的变量名修改。

这个对话框的使用与【查找】对话框相同。操作方法请详见 [8.9 查找](#)。

8.11 POU 调用

任何一个 POU 只有经过调用才能够开始运算，POU 的调用关系如图 85 所示。

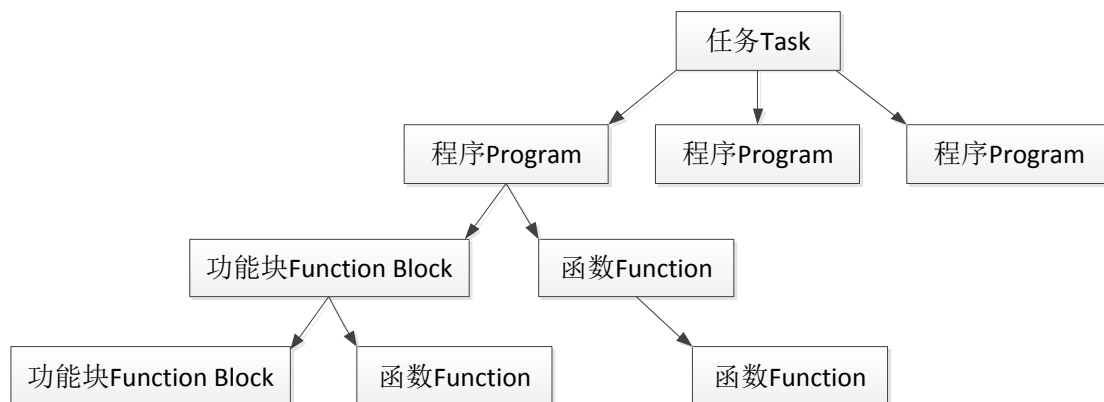


图 85 POU 调用规则

用户可以通过【查看调用树】来了解任务与其他 POU（PRG）之间的关系。

8.11.1 用户程序调用功能块

8.11.1.1 添加块元件

在 LD、FBD 语言编辑环境下，在插入点选择块元件。添加结束后，需要对块元件进行实例声明（FB），在缺省文本 ??? 中输入实例名。

8.11.1.2 引用已声明的功能块（FB）

在同一工程中，一个程序（PRG）引用另一个程序中已声明的功能块时，引用格式：程序名.实例名。

8.11.2 用户程序调用函数

函数的调用方法与用户程序调用功能块的方法相同。只是函数不需要定义实例名。



- 调用函数时，必须将函数的返回值赋值给其他变量；否则提示编译错误。

8.11.3 程序间的调用

程序这里着重指用户程序。

程序间的调用分为任务调用和程序调用。

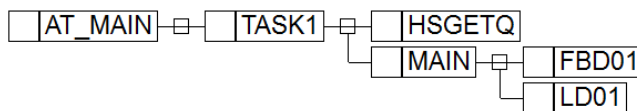


图 86 任务与程序间的调用关系

用户程序必须被任务调用，才能被执行。调用方法请参见章节 6.2.1 程序调用。

用户程序相互调用的操作方法：

选中节，从【用户程序】节点下拖动被调用的程序到调用的位置。以 Main 调用 LD01 程序为例。如图 87 所示。

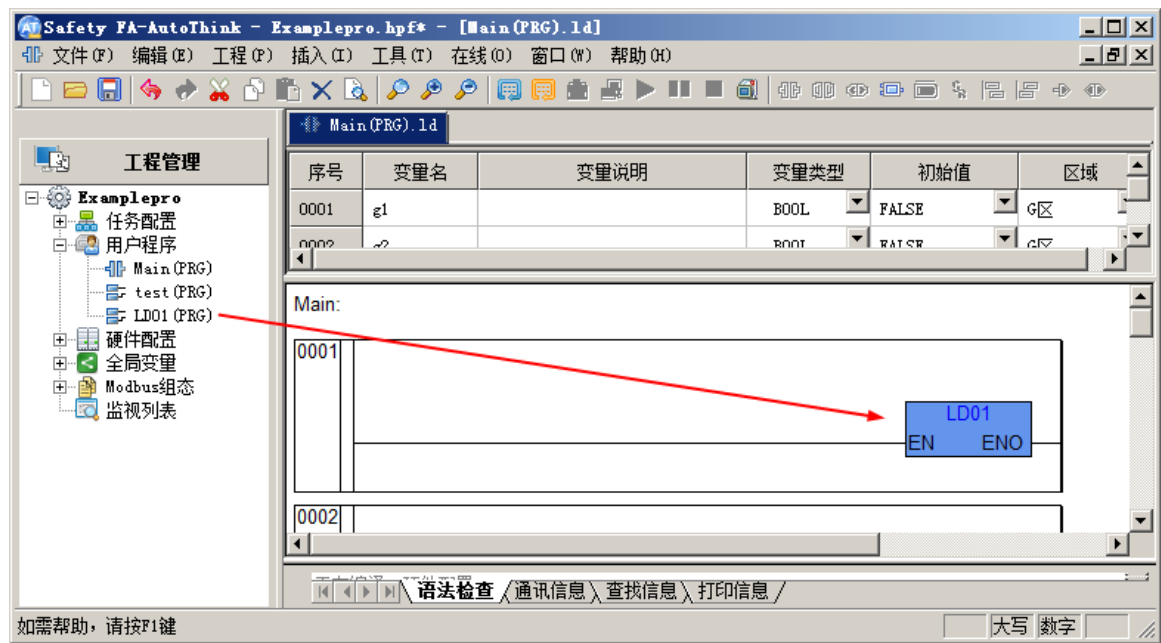


图 87 拖动方式调用

8.11.4 功能块间的调用

功能块间的调用方法与章节 8.11.1 用户程序调用功能块完全相同，只是显示结果有差异。

功能块拖动到 POU（FB）时，系统会自动分配实例名（PRG 名+FB 类型名+ _已声明的同一类型 FB 的序号），并显示变量声明框，如图 88 所示。

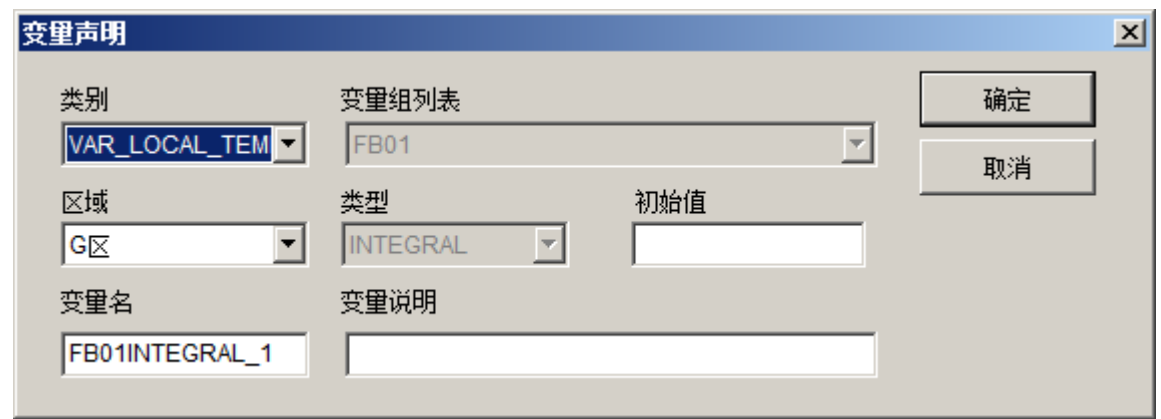


图 88 变量声明对话框

类别：选择变量的属性。包含全局变量和局部变量，类别的意义详见 9.1.2 按照变量的作用域划分。

其他项与 9.3 变量声明中同名项相同。

8.11.5 查看调用树

用树状结构图说明当前工程中程序的相互调用关系，可以层层展开或关闭。工程编译成功后，才能查看调用树。



- 菜单栏：单击【工程】—【查看调用树】；

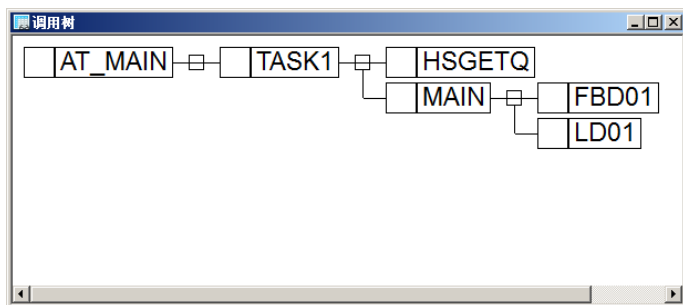


图 89 调用树窗口

8.12 文件夹

8.12.1 添加文件夹

用户可以依据 POU 的组态内容对其进行分类，方便管理。



- 工程管理：右击【用户程序】，单击【添加文件夹】。



图 90 文件夹命名

在编辑框中输入文件夹名称，文件夹命名要求：

1. 不能包含符号（英文字符）\、/、:、*、?、“”、<>、|；
2. 不能以“用户程序”命名；
3. 可以与 POU 名称同名；
4. 名称长度最大为 32 个字符。

文件夹创建后，图标显示 .

当输入名称无效时，**确定**不可选。

8.12.2 删除文件夹



- 工程管理：右击【用户程序】下的文件夹，单击【删除文件夹】。

删除【用户程序】节点下的文件夹时均会弹出系统提示，以确认删除操作。文件夹删除后，不可恢复，请操作时引起注意。

8.12.3 文件夹重命名

对文件夹进行重命名操作。



- 工程管理：右击【用户程序】下的文件夹，单击【文件夹重命名】。

第9章 变量

数据是 Safety FA-AutoThink 中控制和运算的操作对象,我们把变化的数据称为变量,变量存于 CPU 模块的内存储器。为了在数据运算过程中使用、标识这些变量,每个变量都需要有一个名字,即变量名。控制算法组态中所用到的变量的名称和数据类型必须是事先明确的,这就需要首先对用到的变量进行定义,这个定义过程叫做变量声明。

每个变量都必须声明,本节将着重介绍变量的基本声明格式、以及不同类型变量的具体声明方法。

9.1 变量类型

9.1.1 从结构形式划分

9.1.1.1 简单型变量

简单型变量:指单一的变化量,能被赋予一个明确的数值。一个简单型变量仅代表一个意义。如 BOOL、REAL 类型的变量。

9.1.1.2 功能块实例

一个功能块实例由一组特定的变量组成。具体是怎样的一组特定变量,则取决于该功能块实例的类型。所以功能块实例接近于数据结构的概念。

9.1.2 按照变量的作用域划分

全局变量 (VAR_GLOBAL): 是使用范围最广的变量,一个全局变量可以在工程的多个地方同时使用(如图形、算法),但一个工程中的全局变量不允许重名。

局部变量:相对全局变量而言,使用范围比较有限,只在所属 POU 中使用有效;在一个工程中可以重名。

局部变量作用域按照用途可分为输入变量、输出变量、输入输出变量和局部变量。

输入变量 (Input_Variable): 变量是从外部输入的,本功能块(或函数)对该变量只可读。

输出变量 (Output_Variable): 变量是向外部输出的,本功能块对该变量可读写;其它功能块只可读。

输入输出变量 (Input/Output_Variable): 兼有输入变量和输出变量的特性,其它功能块和本功能块对该变量都可读写。

局部变量 (Local_Variable): 是对具体的功能块(或函数)而言,是块实现逻辑运算时所需要的临时变量;该变量不与外部连接,只在它所属的块中有效。在功能块查看信息中被称为参数 (VAR_TEMP)。

9.2 变量命名

变量名只可以由字母、下划线或数字组成,但以数字开头的点名后面必须接上字母或下滑线,不能全部都是数字,若全部为数字系统会默认为常量类型。

变量名长度不超过 32 个字符。

变量名识别下划线，例如 AB_CD 和 ABC_D 被认为是两个不同的变量名。

变量名不区分大小写。例如 VAR1、Var1 和 var1 表示相同的变量。

变量名不能为空，且不能包含空格。例如 AB CD 是错误的变量名。

变量名中不能包含特殊字符，如中划线“-”和加号“+”等。例如 AB-CD 和 AD+CD 是错误的变量名。

变量名不能与类型名（包括自定义类型）、POU 名、枚举名或类型转换函数名重名。

变量名不能以 sys_、AT_ 开头。

变量名不能与关键字相同。

变量名在整个工程保证唯一。

9.3 变量声明

当【工程】—【选项】—【配置】中的变量自动声明项处于勾选状态：新建变量时，会自动弹出变量声明对话框。

在 POU 中新建变量时，输入有效的变量名后，按 **Enter** 键，会弹出“变量声明”对话框，如图 91 所示。

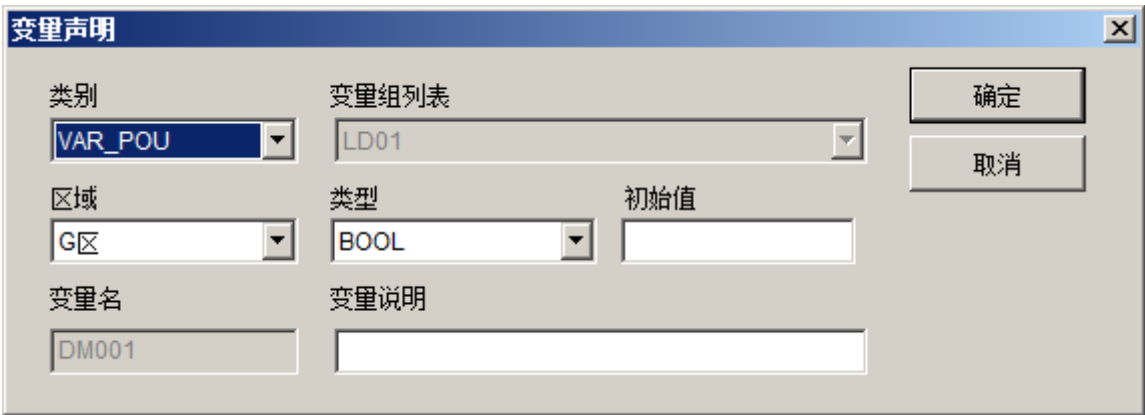


图 91 变量声明

类别：定义变量的作用域。请详见 [9.1.2 按照变量的作用域划分](#)。

不同类型的 POU 中，可以添加的变量不同。

一个 POU 中可以添加的变量最大数见表 16。

表 16 一个 POU 支持的变量数量

作用域	POU (PRG)	POU (FB)	POU (FUN)
全局变量 (VAR_GLOBAL)	无限制	/	/
输入变量 (Input_Variable)	/	64	28
输出变量 (Output_Variable)	/	64	/

作用域	POU (PRG)	POU (FB)	POU (FUN)
输入输出变量 (Input/Output_Variable)	/	32	/
局部变量 (Local_Variable)	/	128	128

区域：参见章节 6.3 设置工程安全属性。

变量名：显示输入的变量名。

变量组列表：在 POU (PRG) 类型中，用来显示 POU 名称。

类型：设置该变量的数据类型，默认为 BOOL；可从下拉菜单中选择合适的数据类型。

初始值：设置该变量的初始值，当变量首次下装时，该值被下装到控制器中参与运算。

变量说明：对变量进行描述，最多不能超出 64 个字符。

9.4 基本数据类型

在变量声明中为了保证类型选择的正确性，您可以通过表 17 先对基本类型有所了解。

表 17 数据类型

序号	关键字	数据类型	取值范围	存储位数	备注
1	INT	整数	-32,768~32,767	16 bit	
2	SINT	短整数	-128~127	8 bit	
3	DINT	双整数	-2,147,483,648~2,147,483,647	32 bit	
4	WORD	字类型	0~65,535	16 bit	该类型的点还可以用于位运算
5	DWORD	双字类型	0~4,294,967,295	32 bit	该类型的点还可以用于位运算
6	BYTE	字节类型	0~255	8 bit	该类型的点还可以用于位运算
7	BOOL	布尔型	FALSE (0) TRUE (1)	1 bit	
8	REAL	实数 / 单精度浮点型	-3.402823466E+38~-1.175494351E-38 1.175494351E-38~3.402823466E+38	32 bit	
9	LREAL	双实数 / 双精度浮点型	-1.7976931348623158e+308~-2.2250738585072014e-308 2.2250738585072014e-308~1.7976931348623158e+308	64 bit	
10	TIME	时间	总是由首字母“t”或“T” (“time”或“TIME”) 和一个数字“#”组成。随之便是包含日 (显示为“d”)，时 (显示为“h”)，分 (显示为“m”)，秒 (显示为“s”)，和毫秒 (显示为“ms”) 的确切的时间声明。请注意，时间项必须按时间顺序排列 (d 在 h 前，h 在 m 前，m 在 s 前，s 在 ms 前)，但无须包含所有时间部分。最大值：49 天 17 时 2 分 47 秒 295 毫秒 (4294967295 毫秒)。		正确形式如： T#14ms T#100S12ms (最高成分允许超过其界限) T#12h23m50s 错误形式如： T#5m68s (在较低的成分中界限被超过的情况) T15m (#丢失) T#14s12d (时间顺序错误)

序号	关键字	数据类型	取值范围	存储位数	备注
11	DATE	日期	1970-01-01 至 2106-02-07		一个日期常数以 “d”， “D”， “DATE” 或 “date” 开始，后接 “#” 号。然后便可按年月日的格式键入日期。 如 DATE#1996-05-06 d#1972-03-29
12	TOD	时刻	00: 00: 00~23: 59: 59.999		用来存储一天的时间，以 “tod#”， “TOD#”， “TIME_OF_DAY#” 或 “time_of_day#” 开始，接着是以时分秒格式出现的时间。秒可作为真正的数字键入，或以秒的分数形式键入。 如 tod#00:00:00
13	DT	日期时刻	1970-01-01-00:00:00 2106-02-07-06:28:15	至	日期与时间的合并：以 “dt#”， “DT#”， “DATE_AND_TIME” 或 “date_and_time” 起始。日期与时间之间用连字符连接。 如 dt#1972-03-29-00:00:00

9.5 复杂数据类型

除基本数据类型外，本软件提供的自定义的复杂数据类型，这些复杂数据类型是由一系列基本数据类型组合完成的，这里是指数组和功能块类型。

9.5.1 数组(ARRAY)

为了将同种类型的数据元素有序的组合成一个有机的整体，以便于引用，可以定义数组(ARRAY)以实现此功能。

根据基本数据类型来定义一维数组，数组是将同一类型的基本数据有序的组合在一起。数组内每个元素有固定的编号(或下标)。

可以初始化数组中所有元素或者不初始化。

例如：为一维数组 ARR1 赋初始值，ARR1 数组下标从 0~6，变量类型为 REAL，如图 92 所示：

序号	变量名	变量说明	变量类型	初始值	区域
0001	ARR1	数组1	ARRA...		G区

■ain. ARR1

序号	变量名	变量说明	变量类型	初始值	区域
0001	ARR1[0]		REAL	0	G区
0002	ARR1[1]		REAL	0	G区
0003	ARR1[2]		REAL	0	G区
0004	ARR1[3]		REAL	0	G区
0005	ARR1[4]		REAL	0	G区
0006	ARR1[5]		REAL	0	G区
0007	ARR1[6]		REAL	0	G区

图 92 为数组元素赋初始值

在 LD 语言编辑器中，对上述数组元素赋值，采用的 MOVE 块进行赋值，格式如图 93 所示：

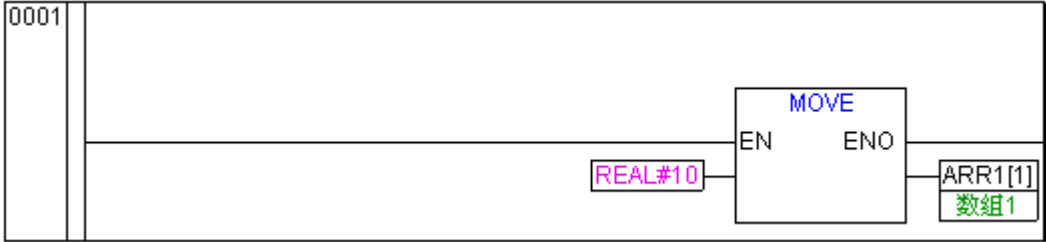


图 93 为数组元素赋值



- 在程序区中使用常数时，必须指明常数的数据类型，采用格式如下“数据类型+ ‘#’ + 数值”。

■ 定义数组

在变量声明对话框中，我们可以定义一维数组型变量。添加变量后，在该变量行中选择“变量类型”单元，单击下拉箭头，从数据类型的下拉列表中选择“ARRAY”项，弹出“数组”定义对话框：

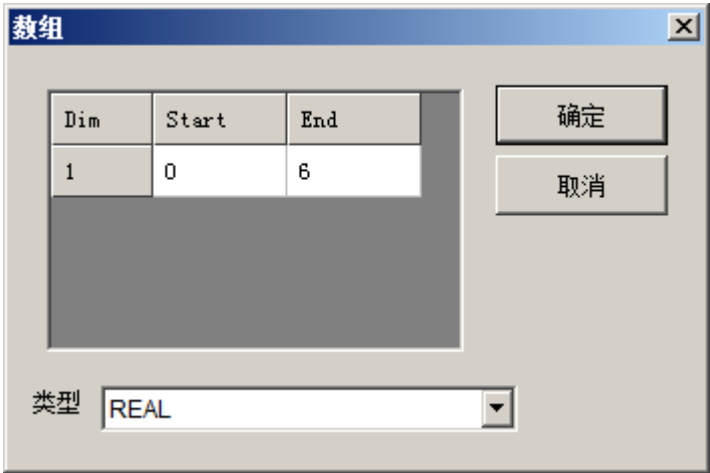


图 94 定义数组

“Dim”列表示定义的数组维数，本软件仅支持一维数组；“Start”列表示数组起始下标，“End”列表示终止下标。在类型下拉菜单中可选择数组元素的类型，可定义的类型如变量的数据类型所示。单击**确定**按钮完成定义。



- 数组的下标可输入的数字范围为 0~ 2147483646，缺省从 0 开始，也可以根据需要修改起始下标的开始位数。
- 一个变量定义的数组元素总数不能超过 8000 个。

选中变量表中的数组变量点所在行，双击“序号”或选择右键菜单中的【详细】命令，弹出该变量各个子项的定义窗口，包括：子项变量名、变量说明、变量类型、初始值、区域等属性，在该窗口中修改各个子项的详细信息。

序号	变量名	变量说明	变量类型	初始值	区域
0001	ARR1	数组1	ARRAY[0..6] OF REAL		G区

Main. ARR1

序号	变量名	变量说明	变量类型	初始值	区域
0001	ARR1[0]		REAL	100	G区
0002	ARR1[1]		REAL	0	G区
0003	ARR1[2]		REAL	2	G区
0004	ARR1[3]		REAL	5	G区
0005	ARR1[4]		REAL	40	G区
0006	ARR1[5]		REAL	70	G区
0007	ARR1[6]		REAL	6	G区

图 95 数组元素的各个子项

■ 引用数组元素

程序中可引用数组元素，例如引用一维数组元素 ArrayVar[2](一维数组元素[2])，以 FBD 语言为例：



图 96 数组元素赋值

9.5.2 功能块类型

与数组概念类似，功能块也是由一组数据元素构成的，但是这些元素是为了某一特定功能而定义在一起的，这样的一组有序的能完成特定功能的数据元素组合的有机整体，称之为功能块。功能块中的各个变量的元素可以由不同的基本数据类型组成的。

本软件可以使用系统提供的功能块（库中自定义类型）和自定义功能块（工程中的自定义类型），如下图所示。系统提供的功能块可以在库管理器中进行查阅。

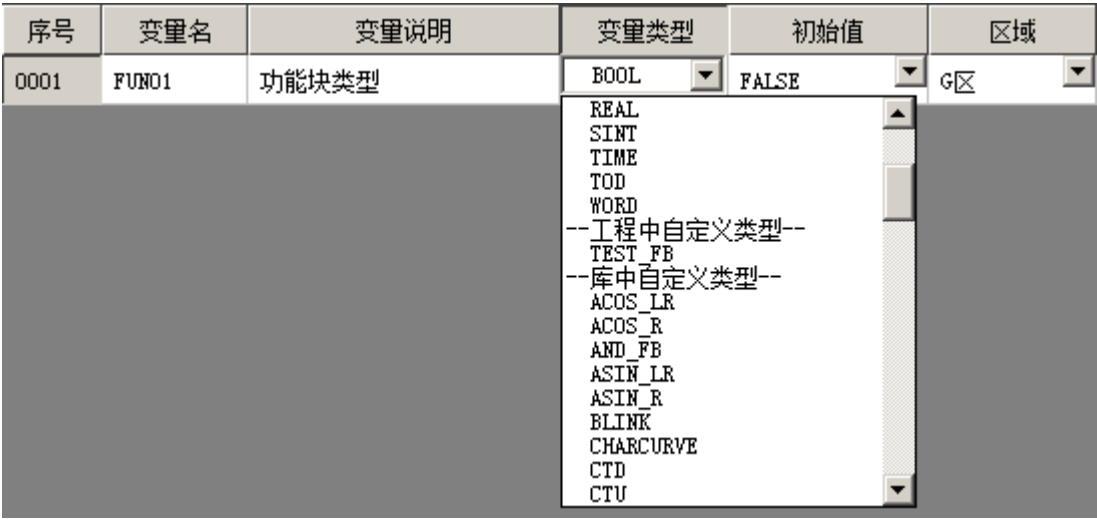


图 97 自定义类型

以系统提供的功能块 ACOS_LR 类型为例，声明的功能块实例 FUN01 如图 98 所示：

序号	变量名	变量说明	变量类型	初始值	区域
0001	FUN01	功能块类型	ACOS_LR		G区
Main.FUN01					
序号	变量名	变量说明	变量类型	初始值	区域
0001	IN		LREAL	0	G区
0002	OUT		LREAL	0	G区
0003	ERR		BOOL	FALSE	G区

图 98 功能块各个子项

■ 引用功能块元素

程序中可引用功能块的元素，例如引用 ACOS_LR 的 ERR 端子(引用功能块的元素)，以 FBD 语言为例：

序号	变量名	变量说明	变量类型	初始值	区域
0001	VarBool01	变量1	BOOL	FALSE	G区
0002	FUN01	功能块类型	ACOS_LR		G区

00001

功能块类型 FUN01.ERR 0 → VarBool01 变量1 0

图 99 引用功能块的子项



- 数组和功能块类型的变量，不能直接修改其初始值，且只能修改其元素的初始值。

9.6 变量监视

监视列表用于对重要的变量进行分类管理，便于监视模式下对其数值变化的掌握和跟踪。通过工程树中的【监视列表】节点可创建多个监视列表。



- 工程管理：右击【工程管理】—【监视列表】，单击【添加列表】。



图 100 添加监视列表对话框

在编辑框中输入监视列表的名称，该名称可以使用任意字符，最长不超过 32 个字符。

创建的监视列表会以【监视列表】的子节点形式显示。如图 101 所示。

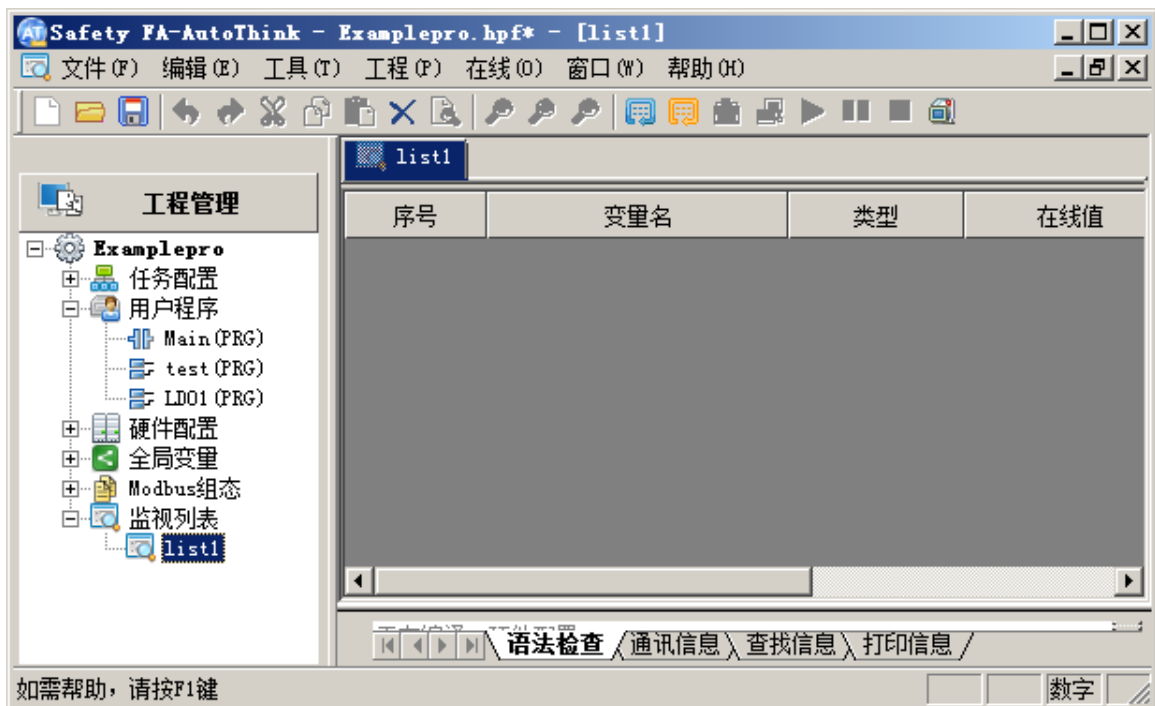


图 101 “监视列表”编辑窗口

在编辑区通过功能菜单可以增加或删除变量。如图 102 所示。

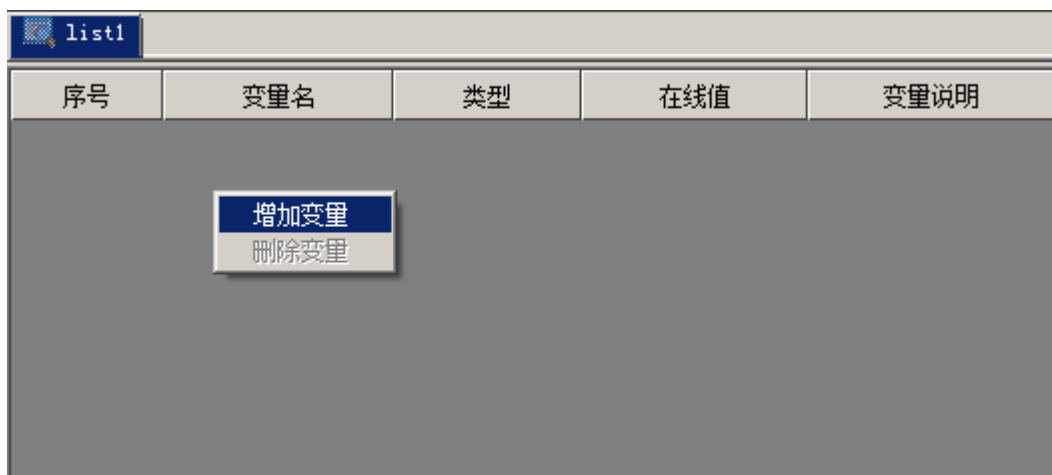


图 102 编辑区的功能菜单

【增加变量】命令每执行一次可以添加一个变量行，变量行中除了**变量名**是必填项外，其他项是非必填项，因为当软件在监视模式下时，软件会自动读取变量的相关内容并显示。不过您填写的变量名必须是已声明成功的。

监视列表中可以添加全局变量，或者添加用户程序（必须为 PRG 类型）中的局部变量，添加局部变量时，格式为**程序名.变量名**，如 **TEST1.VAR1**。

9.7 变量的其他操作

9.7.1 增加变量

在变量组中通过【增加变量】命令，增加一条变量记录。有关添加变量的方法请详见变量声明的内容。

新增加的变量总是显示在变量组的最后一行。

9.7.2 插入变量

选中变量组中的一个或多个变量，然后执行【插入变量】命令，新增加的变量总是显示在选中的序号的最小的变量之前。每次只能插入一个变量。

9.7.3 复制、粘贴

剪切变量，如果该变量名称与新的变量表中的变量无重名，则粘贴后变量名等信息均保持不变；如果与新的变量表中的变量发生重名，则粘贴后变量名自动按照系统命名规则重新命名，其他信息保持不变。

复制的情况和结果与剪切相同。

粘贴、粘贴在上方、粘贴在下方：三者的区别在于粘贴后的位置不同。

- 粘贴：粘贴的对象增加到变量表后面。
- 粘贴在上方：粘贴的对象插入到选中行的上方。
- 粘贴在下方：粘贴的对象插入到选中行的下方。

9.8 用户自定义变量组

下面介绍有关用户如何在工程树的【全局变量】节点下自定义全局变量组的操作：

9.8.1 新建



图 103 自定义变量组



图 104 设置变量组名

输入完成后，单击**确定**完成变量组的添加。

新增变量组后，双击该变量组名，打开的变量列表窗口对变量的进行增加、插入、删除、复制、剪切等操作，详见 [9.7 变量的其他操作](#)。

9.8.2 重命名



图 105 选择重命名命令

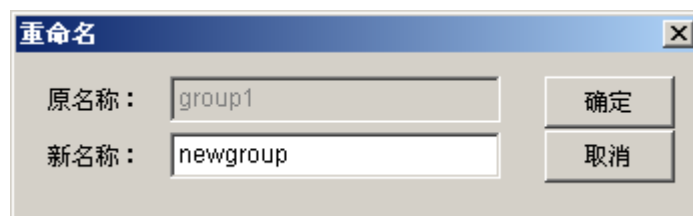


图 106 输入新名称

在新名称处符合要求的组名，单击**确定**结束重命名操作。

9.8.3 删除变量组

用户可以删除非系统定义的变量组。

在需要移除的变量组的右键菜单中选择【删除组】命令。



- 删除的变量组不能恢复。
- 全局变量组命名规则与 9.2 变量命名相同，此处不再赘述。

9.9 变量导入导出

9.9.1 导出全局变量

1. 概述

如果需要对工程中的全局变量进行批量编辑、或导入到其他工程，您需要先导出工程中的所有全局变量。

可导出的全局变量包括全局变量组和程序（PRG）中的变量。全局变量以.xls 变量表形式被导出，在变量表中按标签页分类。

2. 要求

工程已打开

3. 步骤

要导出全局变量，请按以下步骤操作：

第1步 单击【工程】菜单。

第2步 选择【导出全局变量】命令。

将弹出“另存为”对话框。

第3步 输入文件名，并选择保存路径。

默认以工程名保存在“Project”文件夹下。

第4步 单击保存按钮。

9.9.2 导入全局变量

1. 概述

您可以将编辑后的变量表或其他工程导出的变量表导入到当前工程。可导入变量包括全局变量组中的变量和程序（PRG）中的变量。

■ 导入格式要求

- ☐ 变量表名称无要求，后缀为.xls。
- ☐ 全局变量组表格第一行格式必须为：“变量组名” + （COMMON）。
- ☐ 程序（PRG）变量表格第一行格式必须为：“程序名” + （PRG）。
- ☐ 变量间不能有空行。

- ☐ 变量各属性项不能有空缺。
- ☐ 变量名要求请参见 9.2 变量命名。
- 导入规则
 - ☐ 支持导入新增变量组，不支持导入新增程序变量。
 - ☐ 未定义数据区的变量，导入时默认为 G 区。
 - ☐ 工程中已定义同名变量不支持导入。
 - ☐ 工程中的变量组，被再次导入时，组中的同名变量被覆盖。

2. 要求

变量表已编辑完整

3. 步骤

要导入全局变量，请按以下步骤操作：

第1步 单击【工程】菜单。

第2步 选择【导入全局变量】命令。

将弹出“打开”对话框。

第3步 选择要导入的变量表。

第4步 单击**打开**按钮。

变量表被导入，在【语法检查】标签中显示导入结果。

9.9.3 导出硬件变量

1. 概述

您可以导出 I/O 模块的通道变量，进行批量修改，可编辑通道名称和通道说明。

编辑规则：

- 表格中，仅允许修改 Channel Name 和 Channel Comment 列的内容。
- 通道名称命名规则请参见变量命名规则。
- 通道名称不能同名。

2. 要求

I/O 模块已添加

3. 步骤

要导出通道变量，请按以下步骤操作：

第1步 单击【工程】菜单。

第2步 选择【导出硬件点】命令。

将弹出提示确认对话框。

第3步 单击**确定**按钮。

将弹出“另存为”对话框。

第4步 输入文件名，并选择保存路径。

默认以“工程名 Channel”保存在“Project”文件夹下。

第5步 单击**保存**按钮。

9.9.4 导入硬件变量

1. 概述

导出的通道变量编辑好后，再导入到工程。

2. 要求

通道变量表已编辑完成

3. 步骤

要导入通道变量，请按以下步骤操作：

第1步 单击**【工程】**菜单。

第2步 选择**【导入硬件点】**命令。

将弹出提示确认对话框。

第3步 单击**确定**按钮。

将弹出“打开”对话框。

第4步 选择通道变量表。

第5步 单击**打开**按钮。

开始导入通道变量，导入完成后，在**【语法检查】**标签中显示导入结果。

9.10 变量组管理

9.10.1 新建文件夹

1. 概述

全局变量组可通过文件夹进行分类管理。

您可以在“全局变量”节点下创建多个文件夹进行分类，也可以在文件夹下分层级创建文件夹进行分类。

2. 要求

工程已打开

3. 步骤

要新建文件夹，请按以下步骤操作：

第1步 右击“全局变量”树节点。

第2步 在右键菜单中，选择**【添加文件夹】**命令。

将弹出添加对话框。

第3步 输入文件夹名称。

- ☐ 不能包含符号（英文字符）\、/、:、*、?、“”、|
- ☐ 不能以“全局变量”命名
- ☐ 不能与变量组名同名
- ☐ 名称长度最大为 32 个字符

第4步 单击**确定**按钮。

当名称无效时，确定按钮不可用。

9.10.2 删除文件夹

1. 概述

当删除文件夹时，文件夹下的所有文件夹和变量组全部被删除，删除前请确认是否删除全部文件夹。

2. 要求

有一个文件夹

3. 步骤

要删除文件夹，请按以下步骤操作：

第1步 右击文件夹名称。

第2步 在右键菜单中，选择【删除】命令。

将弹出删除确认对话框。

第3步 单击**确定**按钮。

9.10.3 重命名

1. 要求

已创建文件夹

2. 步骤

要重命名文件夹，请按以下步骤操作：

第1步 右击文件夹名称。

第2步 在右键菜单中，选择“重命名”命令。

将弹出“重命名”对话框。

第3步 输入新名称。

命名规则请参见 [9.10.1 新建文件夹](#)。

第4步 单击**确定**按钮。

当名称无效时，确定按钮不可用。

9.11 变量的访问

“访问”即“使用”，运算过程中对变量的访问包括：读取变量值和给变量赋值。

- 访问一个**简单型**变量，直接输入变量名即可。
- 访问一个**功能块实例**的项，书写格式为：**功能块变量名.项名**
如：R_TRIG01.Q(取 R_TRIG01 的设定值项,前提已经声明了 R_TRIG01 的类型为 R_TRIG。
- 访问一个**局部变量**（其他程序进行引用），书写格式为：**程序名.变量名**
如：VAR1(POU1 中定义的局部变量)，在 POU2 中的引用格式为：POU1.VAR1。
- 快速访问变量
如果您想快速了解程序中当前变量的类别，那么选中变量名，按下 **F2** 功能键，会弹出该变量的“变量声明”对话框，快速显示当前选中的变量的属性窗口。

9.12 未使用全局点回收

对工程中用户自定义的但未使用（指未参与算法组态或 Modbus 通讯）的全局变量进行回收。
工程编译通过后，该命令可用。



- 菜单栏：单击【工程】—【未使用全局点回收】。



图 107 未使用全局点回收

窗口中显示已定义但未使用的全局变量,包括 PRG 类型 POU 和用户自定义全局变量组中的变量。
手动勾选需要删除的变量，或通过单击**全部勾选**框快速全选，单击**回收**，进行删除。



- 使用这个命令删除的逻辑点，不可恢复。

9.13 交叉引用表

工程编译成功后，可以查看整个工程的交叉引用的变量和程序。

选择【工程】—【查看交叉引用表】命令，显示“交叉引用表”对话框，示例如图 108 所示。显示的引用信息列表中包含如下信息：变量名、POU 名和位置，双击信息行可以打开相应的交叉引用程序的编辑窗口，并且光标定位到引用位置处。

序号	变量名	POU名	位置
0001	hsgetq.bjmpctrld	HSGetQ (PRG).ld	0050
0002	hsgetq.bjmpctrld	HSGetQ (PRG).ld	0051
0003	hsgetq.bjmpctrld	HSGetQ (PRG).ld	0062
0004	hsgetq.bjmpctrld	HSGetQ (PRG).ld	0063
0005	q_dpio_11_6_all_b	HSGetQ (PRG).ld	0040
0006	s_dpio_11_9_a_end	HSGetQ (PRG).ld	0030
0007	s_dpio_11_9_a_end	HSGetQ (PRG).ld	0033
0008	s_dpio_11_6_c_end	HSGetQ (PRG).ld	0014
0009	s_dpio_11_6_c_end	HSGetQ (PRG).ld	0017
0010	q_dpio_11_6_all_c	HSGetQ (PRG).ld	0040
0011	b_bus	HSGetQ (PRG).ld	0037
0012	b_bus	HSGetQ (PRG).ld	0046
0013	b_bus	HSGetQ (PRG).ld	0058
0014	s_dpio_11_6_b_end	HSGetQ (PRG).ld	0013
0015	s_dpio_11_6_b_end	HSGetQ (PRG).ld	0016

图 108 交叉引用表

9.14 变量调试回读



- 菜单栏：单击【在线】—【变量调试回读】。

工程退出监视状态后，执行该命令可以将 M 区变量的在线值赋给初始值。

第10章 编程语言

POU 可以用两种语言编写：FBD 和 LD。它们的特点如下：

FBD：功能块图，是一种类似于数字逻辑电路结构的编程语言，是一种使用布尔代数的图形逻辑符号来表示的控制逻辑，一些复杂的功能用指令框表示，适合于有数字电路基础的编程人员使用。

LD：梯形图语言，是在原电气控制系统中常用的接触器、继电器梯形图基础上演变而来的，它与电气操作原理图相呼应，形象、直观并且实用，为广大继电器控制的电气人员所熟知，特别适合于数字量逻辑控制，是使用最多的 PLC 编程语言，但不适合于编写大型控制程序。

10.1 FBD 编辑器

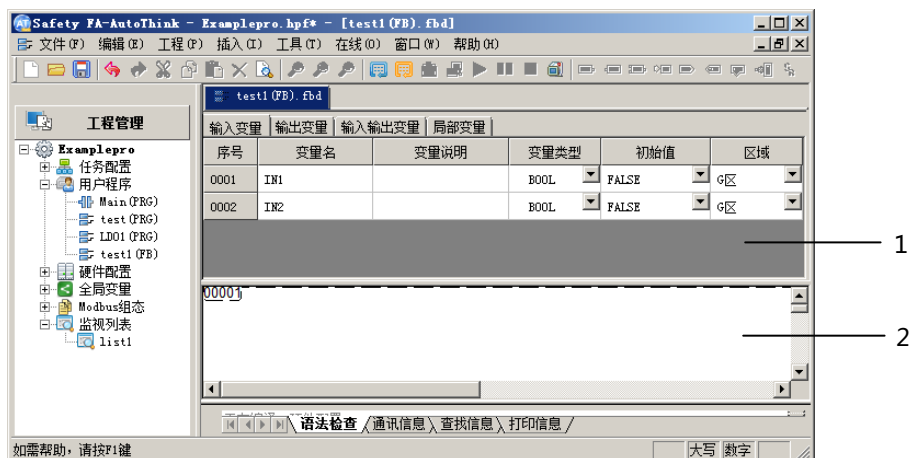


图 109 FBD 语言编辑的 POU 页面

1: 变量区

2: 编程区

变量区位于 FBD 编辑器的上部，它用于显示当前 POU 已声明的局部变量的详细信息。局部变量的分类会根据 POU 类型的不同而有所差异。

编程区是用来对功能块和连接线进行编程的。

编程区的网格是为了定位块元件，当检测到最小距离时，用户只能将块的角和信号流线放在网格线上，变量和常数能自由放置，他们在矩形框中显示和编辑。网格的显示与隐藏可以进行设置。

当 POU 文件设置为分页打印时，每个用 FBD 语言编辑的 POU 被划分成了若干页，将完整的 POU 页面划分成了栅格。通过水平和垂直的滚动条可以接触到每个独立的页面。独立页间的分界线用黑点线表示在屏幕上。打印出的每一页和编程区的独立页是完全对应的。

为了保证打印的完整性，建议在排版时尽可能地将程序放在黑点线以内。

编程区中没有使用捕捉栅格，因此元素可以任意放置。连续处理的元素包括块、输入、输出、跳转和返回等。块元素与一个元素的输出引脚和另一个的输入引脚连接。这里连接形式用信号流线表示。信号流线的编辑通过拖动完成，信号流线自动画出。

10.1.1 光标位置

在 FBD 编辑窗口中，当光标在元件上移动时，单击各个元件或元件的引脚，光标位于选中的对象上，显示为蓝色；双击元件上的文本域位置，显示为蓝色，可以输入变量名或说明文字等。

10.1.1.1 选中块、输入、输出、跳转、标签、返回和注释元件

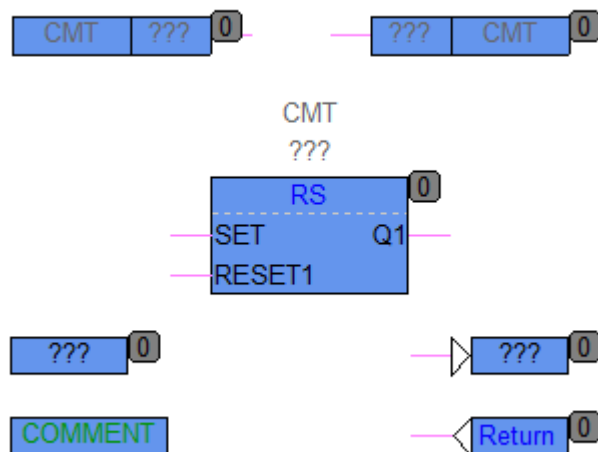


图 110 选中元件

10.1.1.2 选中块、输入、输出、跳转、标签、和注释元件的文本区域

单击文本区域，即可选中。

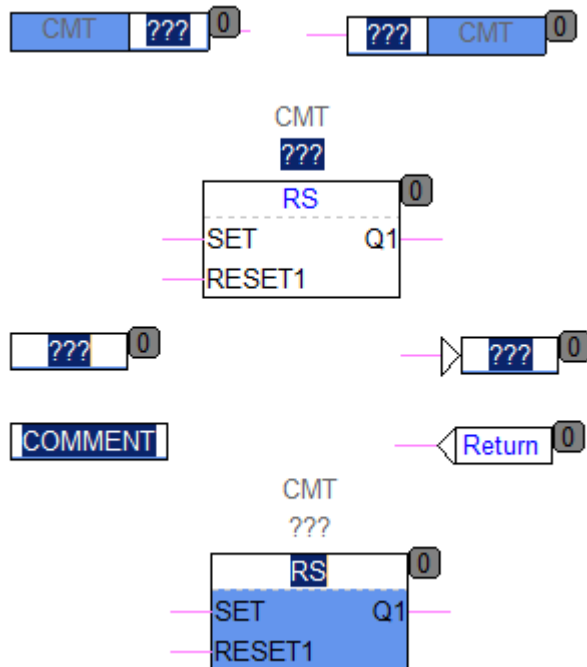


图 111 选中元件的文本区域

10.1.1.3 块、输入、输出、跳转、标签、返回这些元件的输入端

单击输入端，即可选中（呈蓝色），一次只能选中一个。

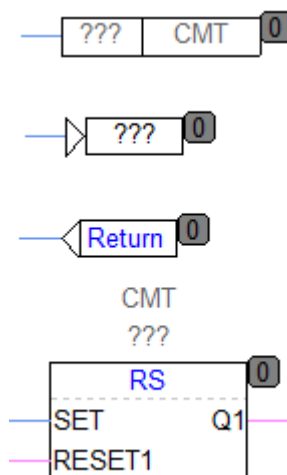


图 112 选中输入端

10.1.1.4 块和输入元件的输出端

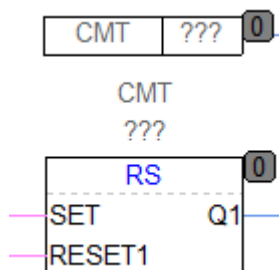


图 113 选中输出端

元件选中状态标识的颜色可以在【工程】—【选项】—【颜色】中进行修改。

需要同时选中多个元件时，可以按下 **Ctrl** 键的同时，选择需要的元件；也可以按下鼠标左键，在编辑器中拖动鼠标画矩形区域来选中其中几个元件。

使用 **Ctrl+A** 组合键可以选中当前编辑器中的所有元件。

10.1.2 网络线



- 菜单栏：单击【插入】—【网络线】；
- 快捷键：Ctrl+W；
- POU：右击空白区域，单击【网络线】。

在 FBD 语言环境中，可以利用网络线将程序区划分成一个个区域，这样可以设置每段程序的执行次序。



- 用户如果在网络线内移动元件位置时，不会触发工程编译；如果跨网络线移动时，则会触发工程编译。
- 创建每个 FBD 类型的 POU 时，系统自动在编辑窗口的最上端添加一条编号为 00001 的网络线，只有该网络线不可以拖动。

10.1.3 输入元件



- 菜单栏：单击【插入】—【输入元件】；
- 工具栏：；
- 快捷键：Ctrl+I；
- POU：右击空白区域，单击【输入元件】。

这个命令用来插入一个输入，出现的文本“???”可以被选中并被变量或常量替代。

新插入的输入元件随鼠标移动，移动到合适的位置单击，插入成功。如图 114 所示。

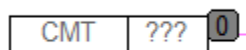


图 114 输入元件

在编程区添加输入元件时，缺省输入元件显示变量名和注释；在 6.6.3 组态语言中设置后，可以隐藏注释内容。

10.1.4 输出元件



- 菜单栏：单击【插入】—【输出元件】；
- 工具栏：；
- 快捷键：Ctrl+U；
- POU：右击空白区域，单击【输出元件】。

这个命令用来插入一个输出，可以选中出现的文本“???”并且可以用变量替代，与这个输出的输入相关的值分配给这个变量。

执行此命令，可以插入输出元件。新插入的输出元件随鼠标移动，移动到合适的位置单击，插入成功。如图 115 所示。

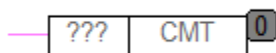


图 115 输出元件

输出元件与输入元件一样，在输入 FB 型变量名后，系统自动为其添加默认的项名。

10.1.5 块元件



- 菜单栏：单击【插入】—【块元件】；
- 工具栏：；
- 快捷键：Ctrl+B；
- POU：右击空白区域，单击【块元件】。



图 116 块元件

双击“AND”可以将其改为其它有效的功能块、函数、程序名。

添加块元件还可以从“库管理器”窗口中将期望的功能块（如“RS”），拖动至目标位置，当鼠标呈状态，释放左键，块元件自动添加到目标位置。

如果是拖动到 POU（FUN/FB）中，参见 8.11.3 程序间的调用。

10.1.6 跳转元件和标号元件

在 FBD 语言环境中，跳转和标号是配合使用的，下面分别说明如何添加跳转和标号。

10.1.6.1 跳转元件



- 菜单栏：单击【插入】—【跳转元件】；
- 工具栏：；
- 快捷键：Ctrl + J；
- POU：右击空白区域，单击【跳转元件】。

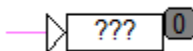


图 117 跳转元件

新插入的跳转随鼠标移动，移动到合适的位置单击，插入成功。

文本域“???”中输入要跳转到的标号名称，该名称必须与标号定义的名称一致。

跳转条件必须为 BIT 或 BOOL 类型（包括可以转换成 BOOL 的类型）的变量。

10.1.6.2 标号元件



- 菜单栏：单击【插入】—【标号元件】；
- 工具栏：；
- 快捷键：**Ctrl + L**；
- POU：右击空白区域，单击【标号元件】。

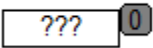


图 118 标号元件

新插入的标号元件随鼠标移动，移动到合适的位置单击，插入成功。

文本域“???”中输入标号的标签名。该名称必须与跳转名一致。

当跳转条件为 **TRUE** 时，将跳过跳转标号之前的程序段，执行跳转标号位置的程序行。如图 119 所示，当程序段 1 中的“VAR1”为 **TRUE** 时，跳过程序段 2，执行标号 Label0 所在的第 3 段程序。

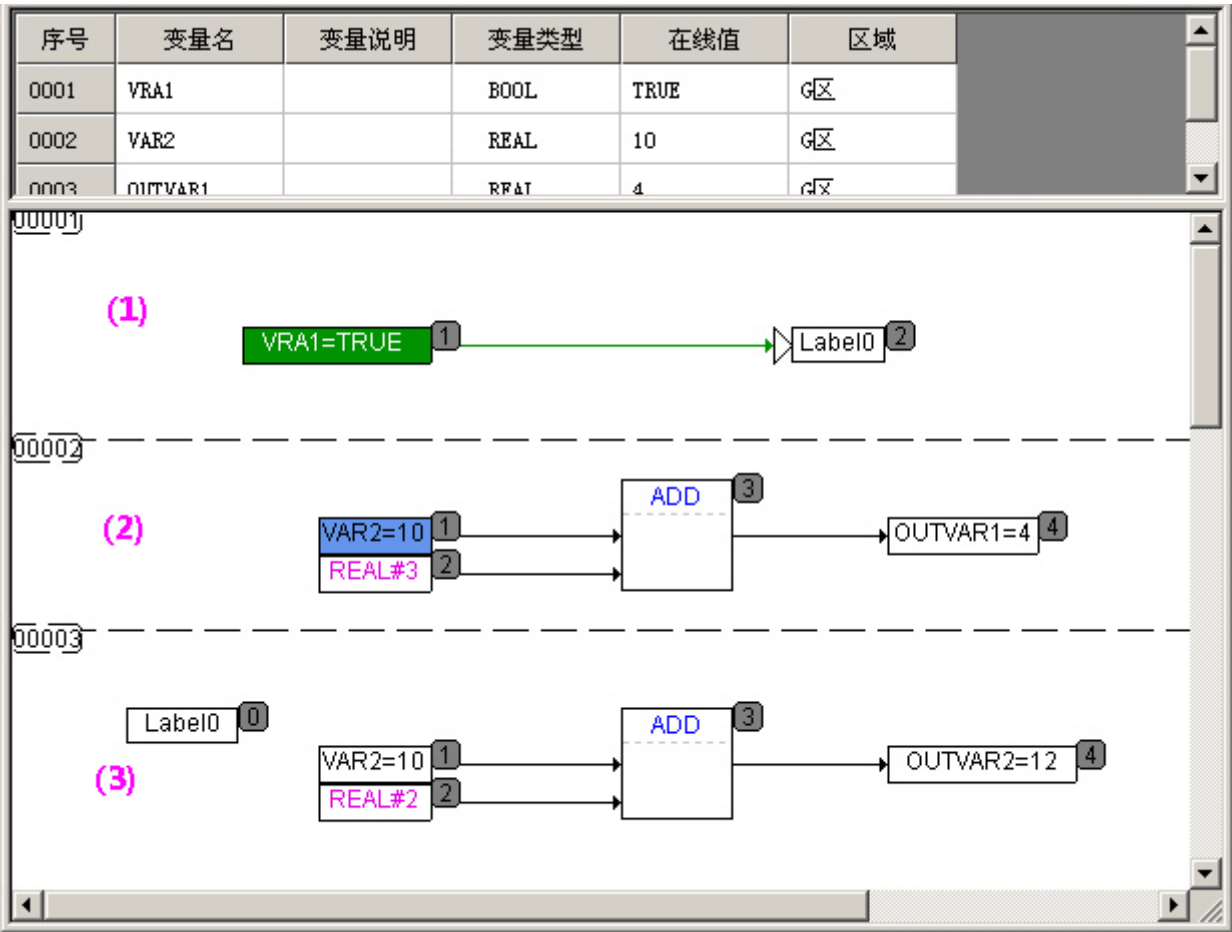


图 119 添加跳转和标号示例

10.1.7 返回元件



- 菜单栏：单击【插入】—【返回元件】；
- 工具栏：；
- 快捷键：Ctrl+R；
- POU：右击空白区域，单击【返回元件】。

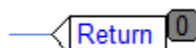


图 120 返回元件

用来提供从函数、功能块或程序中提前退出的功能。

新插入的返回随鼠标移动，移动到合适的位置单击，插入成功。

返回元件只能连接 BIT 或 BOOL 类型（包括可以转换成 BOOL 的类型）的变量。

10.1.8 注释元件



- 菜单栏：单击【插入】—【注释元件】；
- 工具栏：；
- 快捷键：Ctrl+K；
- POU：右击空白区域，单击【注释元件】。

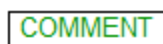


图 121 注释元件

为程序添加注释，可以增强控制逻辑的可读性。

新插入的注释随鼠标移动，移动到合适的位置单击，插入成功。

缺省文本为“comment”，双击该文本，修改注释内容。

10.1.9 置位/复位



- 菜单栏：单击【插入】—【置位/复位】；
- 工具栏：；
- 快捷键：Ctrl+T；
- POU：右击输出元件，单击【置位/复位】。

选中一个输出元件，使用【置位/复位】命令，为该元件设置置位或复位功能，如图 122 所示。

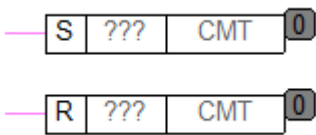


图 122 置位和复位功能

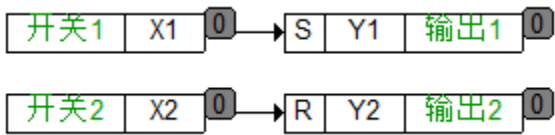


图 123 使用示例

当 X1=TRUE, Y1=TRUE; Y1 保持这个值, 即使当 X1=FALSE;
当 X2=TRUE, Y2=FALSE; Y2 保持这个值, 即使当 X2=FALSE;
重复激活这个命令可以使输出在置位、复位和正常状态之间转换。

10.1.10 置反



- 菜单栏: 单击【插入】—【置反】;
- 工具栏: ;
- 快捷键: **Ctrl+Q**;
- POU: 右击元件引脚, 单击【置反】。

这个命令用来对输入、输出、跳转、返回命令取反, 在连接上用一个圆圈表示否定的符号。
块元件、输出、跳转或返回当它们被选中时, 它们的输入否定。
块元件的输出或输入当它被选中时被否定。
通过再次否定可以取消原来的否定。
选中某一元件的引脚, 使用【置反】命令, 为该引脚的值取反。

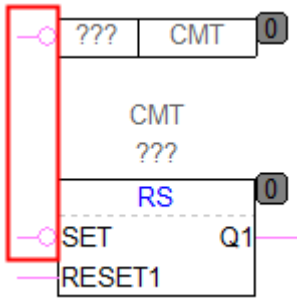


图 124 置反样式



- 置反功能仅适用于 BOOL、BYTE、WORD、DWORD 和 BIT 类型。

10.1.11 元件对齐

该功能可以帮助用户对 POU 中的元件进行快速排版。支持元件的排列方式有水平左/右对齐和垂直对齐。

用户选中 POU 中两个及其以上元件后，在右键菜单中可以选择相关命令。如图 125 所示。

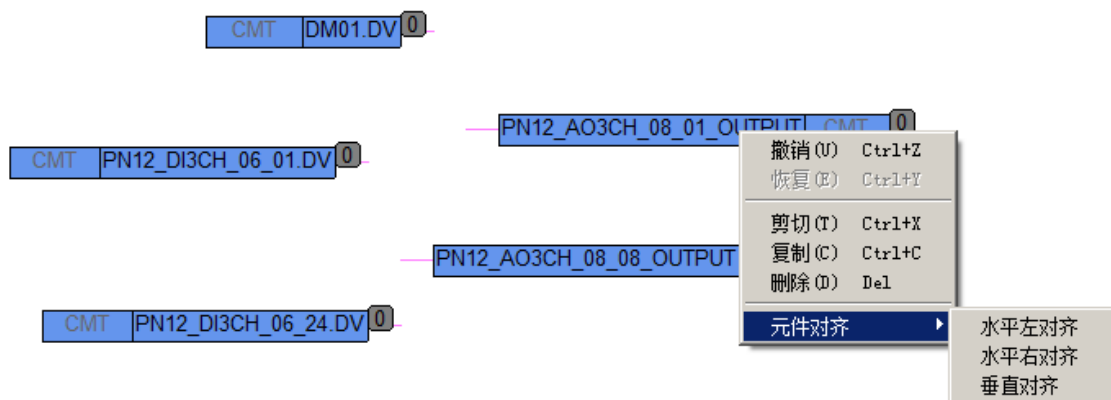


图 125 选择对齐方式

水平左/右对齐：以选中元件中最左/右边位置的元件为基准，选中的其他元件在水平方向上与基准对齐。如图 126 和图 127 所示。

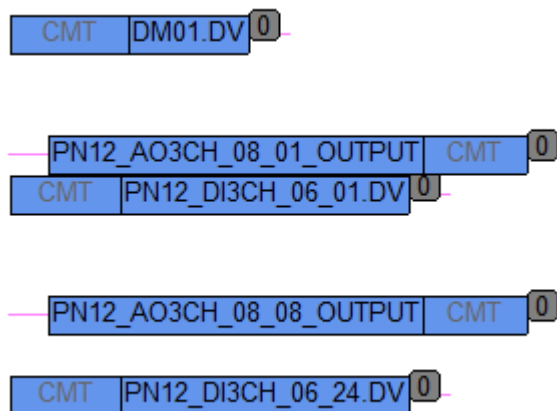


图 126 左对齐排版的结果

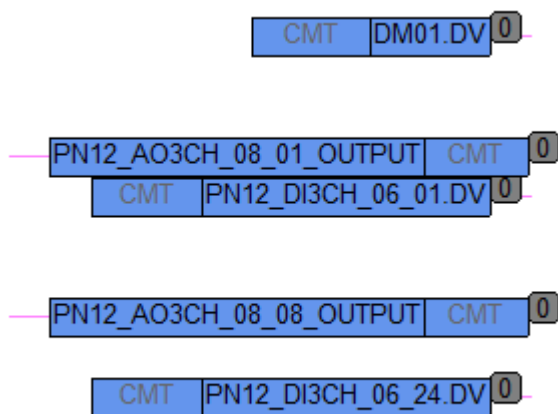


图 127 右对齐排版的结果



- 用户执行元件对齐命令，并保存对齐结果后，会触发当前工程编译；如果用户在保存之前想放弃对齐结果，可以使用撤销命令还原。
- 当元件对齐操作会使某元件移动到边界线（黑点线）外时，该元件及其后面的元件均不会移动。

10.1.12 执行顺序和编号

除注释元件外，其他元件的右上角均自动添加了执行序号`0`，该序号用于确定在线模式或仿真模式下 FBD 中元件的执行顺序。

添加一个新元件时，该元件将自动获得一个执行序号，缺省序号为 `0`。移动元件时，该元件所在的执行序号仍保持不变。

当工程编译通过时，FBD 按照一定的排序原则生成各个元件的计算或执行顺序。

执行流程从执行序号为 `1` 的元件开始。

在本软件中，当执行编译操作时，FBD 按照一定的排序原则生成各个元件的计算或执行顺序，下面就排序规则进行说明。

首先介绍“终点元件”的概念。所谓“终点元件”是指：在合法网络中，没有右引脚或者右引脚悬空的元件。

- 输出元件、跳转元件、返回元件、PRG 只能作为终点元件；
- FB 可以是终点元件，也可以不是；
- 输入元件、指令块和 Function 一定不是终点元件。

以下述图 128 所示为例，说明基于上述终点元件的定义，FBD 合法网络中元件的排序原则为：

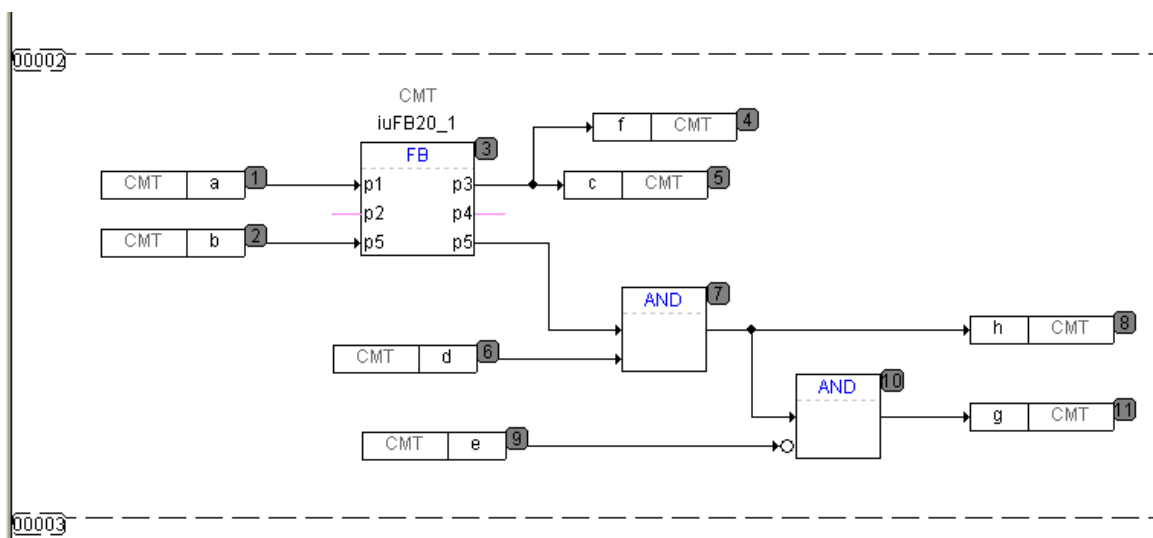


图 128 执行顺序说明示例

- (1) 初始时，所有元件的执行序号均为 0，编译后排序时，第一个序号为 1；
- (2) 各个终点元件的执行顺序应该按照几何位置：先上后下，先左后右。例如上述例子中，终点元件为：f，c，h，g。根据几何位置关系，这四个元件的执行顺序应为 f，c，h，g；
- (3) 按照 (2) 中的顺序处理每一个终点元件，从终点元件开始按照原则 (4) 排序；
- (4) 任何一个元件如果要得到执行序号，除非它的所有左引脚连接的元件全都获得了执行序号，否则按照引脚顺序，先处理这些左引脚连接的元件，处理原则仍为 (4)。

10.2 LD 编辑器

LD 是梯形图 (Ladder Diagram) 是一种基于 IEC61131-3 标准的图形化编程语言。

LD 语言是源自现场的电磁继电器，用来表明控制器的 POU 的独立节点当前通过的电源流。

当 POU 文件设置为分页打印时，每个用 LD 语言编辑的 POU 的工作区由若干个页面组成。通过水平和垂直的滚动条可以接触到每个独立的页面。一个隐形的栅格覆盖了整个工作区。一个 POU 文件打印输出的每一页内容和独立页的是完全对应的。

LD 的 POU 也有标题栏，位置和作用与 FBD 的相同。

一个 LD 网络节点被叫做电源轨线的垂直线左右界定。左电源轨线的状态被视为总是 ON (接通)。

■ 一个 LD 网络节点由下列图形元素组成：

- (1) 电源流线
- (2) 变量和常数
- (3) 触点
- (4) 线圈
- (5) 条件和非条件跳转
- (6) 函数和功能块

一个 LD POU 中可以有几个网络节点，它们是按照从上到下的顺序进行运算，除非这里有跳转元素。

■ 一个 LD POU 是根据下面规则进行运算：

- (1) 输入端运算状态完成时，网络节点才可能运算；
- (2) 输出端的运算状态完成时，网络节点才运算结束。
- (3) 所有的输出端运算状态完成时，网络节点才运算结束，即使网络节点的前面或后面包含跳转。
- (4) 网络节点的运算过程是从上到下。

但是，规则 4 也是依靠 POU 的电源流，而规则 1-3 是必须遵循。为了说明执行顺序，下面运算法则用来确定顺序的：

- 第 1 步 所有网络节点元素的执行顺序是从上到下，从左到右；
- 第 2 步 寻找第一个网络节点中的所有输入再运算；
- 第 3 步 这个网络节点元素运算；
- 第 4 步 如果这里没有其它网络节点元素计算，返回第 2 步。

与 FBD 语言不同，这里的块和输出元素上没有被指定明确的执行顺序，但是 LD 是通过 POU 的结构体现的，反馈信息是不允许的，否则就违背了规则 1。

在分析梯形图中的逻辑关系时，可以想象两条垂直电源轨线之间有从左向右流动的直流电。如图 129 所示。

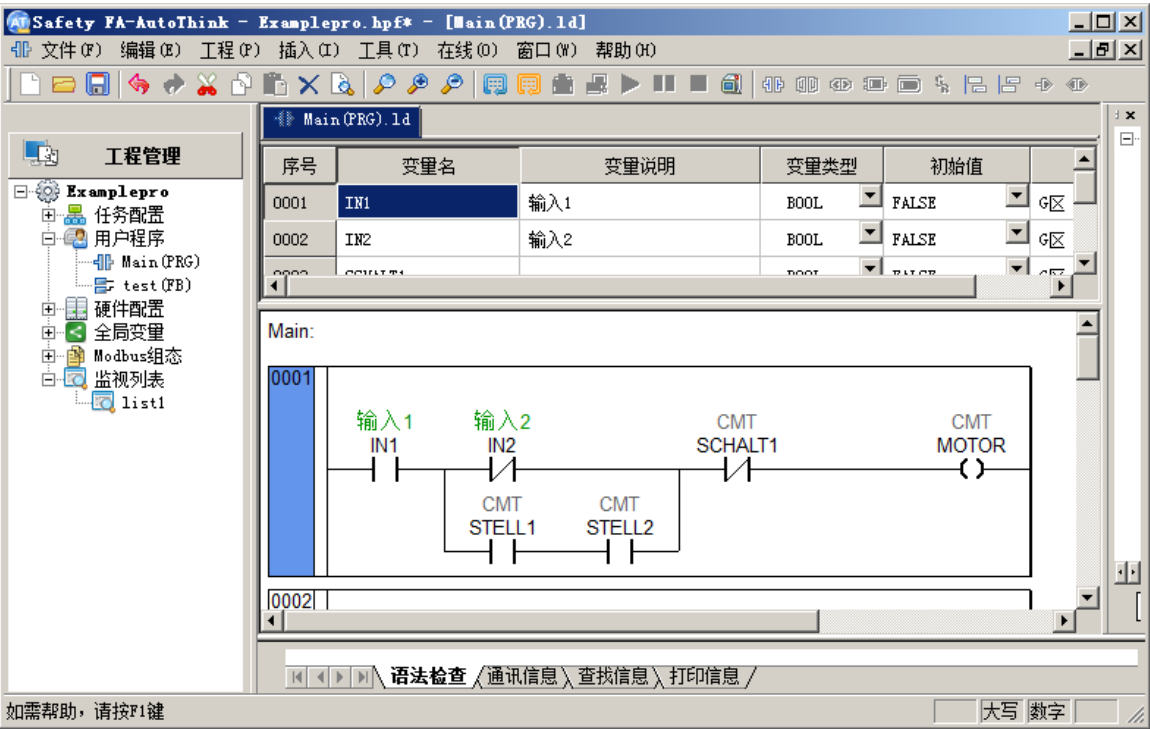


图 129 LD 编辑器

10.2.1 光标位置

选中的内容通过标记蓝色背景来显示，下面是光标可能位置的举例。

■ 节

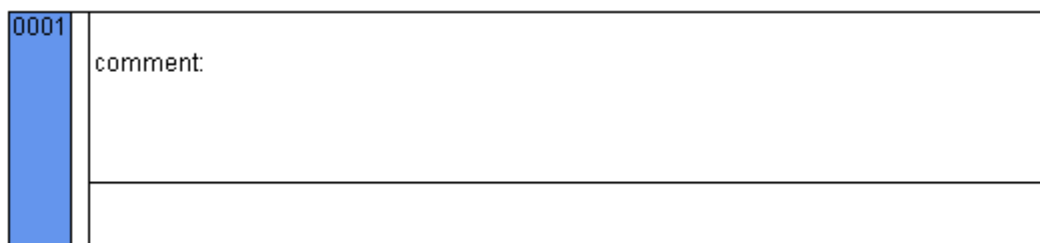


图 130 选中节

■ 触点

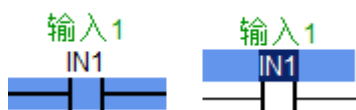


图 131 选中触点及其文本

■ 线圈

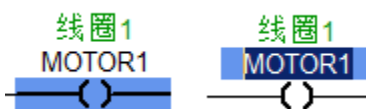


图 132 选中线圈及其文本

■ 块元素

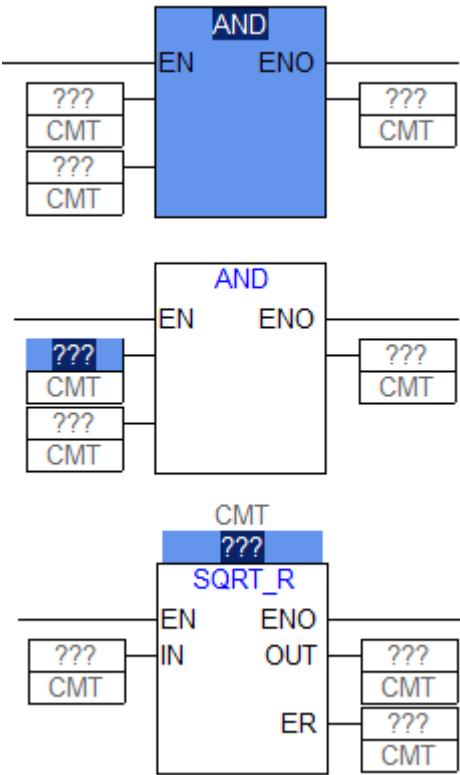


图 133 选中块元素及其文本

■ 返回

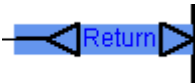


图 134 选中返回元素

■ 跳转

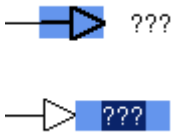


图 135 选中跳转及其文本

■ 注释和标号

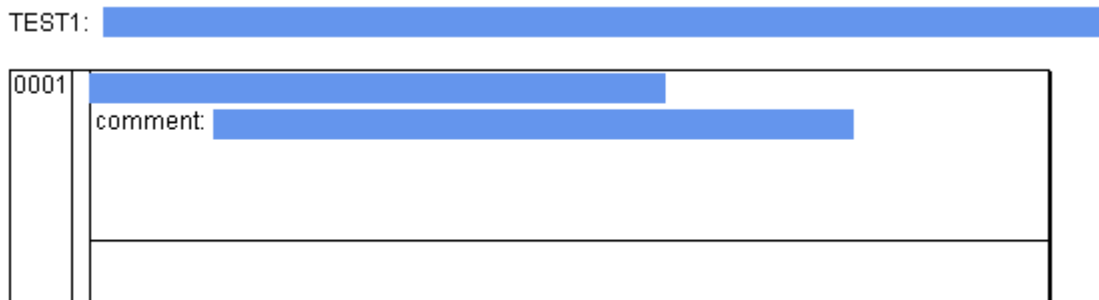


图 136 选中注释和标号文本框

10.2.2 LD 元素

1. 节

“节”是 Safety FA-AutoThink 中的一个重要概念，是 LD 程序的基本单元，每个 LD 语言编写的 POU 都是由“节”组成。节号缺省从 0001 开始，每个 POU 中最多能添加 999 个节。

2. 触点

在梯形图中的每一个网络图的左边都有触点（触点是两个平行线||来表示）。触点代表逻辑输入条件，如开关、按钮和内部条件等；它用来表示电路的“开”“关”状态。

这些状态与布尔变量 TRUE(ON)和 FALSE(OFF)相一致。布尔变量属于每一个触点。如果变量值为 TRUE，那么状态可以通过连接线从左边传到右边。否则，右边接收到的是“断开”。

触点可以并联使用，其中的一个并联分支必须传递“开”状态时，并联分支才能传递“开”。或者触点串联连接，此时，触点必须传递“开”状态时，最后的触点才传递“开”，这些与串并联连接的电路一致。

：常开触点，如果其变量值为 ON 时，左侧链路的状态会被传递到右侧链路；反之，右侧链路为 OFF。

3. 线圈

在梯形图网络图的右边有一些所谓的线圈，线圈通常表示逻辑运算输出结果，用来控制外部的指示灯、接触器和内部的输出条件等，用 () 表示并且只能通过水平线来连接。线圈传递从左到右的连接状态，并且复制状态到布尔变量中，可以描述入口线的状态为“开”（对应布尔变量的 TRUE）或者“关闭”状态（对应布尔变量的 FALSE）。

：线圈，线圈将其左边链接的状态无修改地复制到其右边的链接，并把左边链接的状态或转换的适当功能存储到有关的布尔变量。

4. 块元件

可以在网络图中添加块元件和程序，但它们必须具有布尔型值的输入和输出并且可以像触点那样用在梯形图的左边。

在图形表示中，基本指令块用包含多个输入和多个输出的块结构表示。输入始终表示在块结构的左侧，而输出始终表示在块结构的右侧。指令块的名称（即功能块类型）显示在块结构的中央。实例名称显示在块结构的上方。

块元件包括功能块、函数和程序。

5. 图例说明

梯形图包含了一系列的节，左右两边各有一个垂直的电流线（能量线）限制其范围，在中间是由触点、线圈、连接线组成的电路图。如图 137 所示。

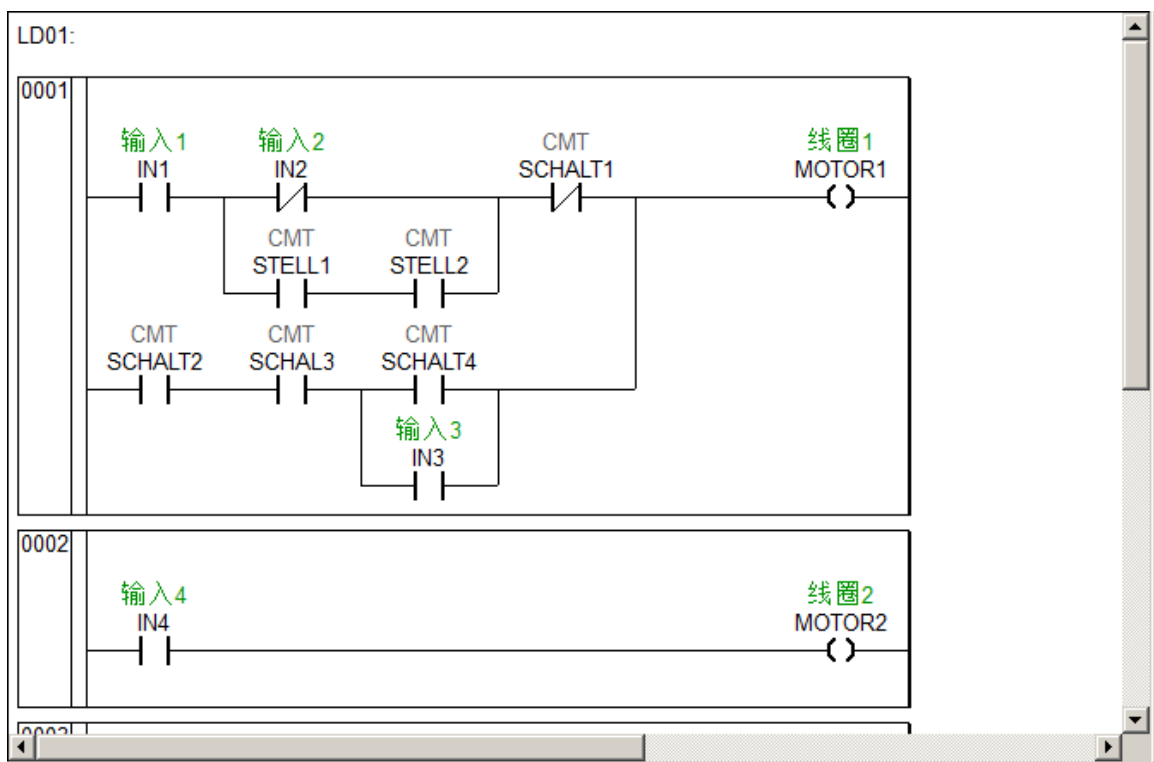


图 137 LD 语言编程元素图例

10.2.3 节

10.2.3.1 添加节



- 菜单栏：单击【插入】—【前节】/【后节】；
- 工具栏： / ；
- POU：右击空白区域，单击【前节】/【后节】。

此时，会在 LD 编辑器当前选中节的前面或后面添加一个新节。



- 在 LD 语言编程区的每个节中，横向元素个数不超过 32 个，纵向元素个数不超过 16 个。

10.2.3.2 删除

选中要删除的节，直接使用 **Delete** 键。

10.2.3.3 复制和粘贴节

当光标位于节时，可以对选中的节进行在同一 POU 或不同 POU 中复制、粘贴等操作。

■ 复制



- 菜单栏：单击【编辑】—【复制】；
- POU：右击空白区域，单击【复制】。

复制单个节：单击期望被复制的节，选择【复制】命令进行拷贝。

复制多个节：当需要复制多个连续的节时，先单击第一个，然后按下 **Shift** 键，单击最后一个节，则在这两个节之间的节全部选中。选择【复制】命令进行拷贝。

■ 粘贴



- 菜单栏：单击【编辑】—【粘贴】；
- POU：右击空白区域，单击【粘贴】。

将剪切板的节粘贴到当前选中节的前面/后面。

10.2.4 串联触点



- 菜单栏：单击【插入】—【串联触点】；
- 工具栏：；
- 快捷键：Alt+I+F；
- POU：右击节/触点，单击【插入触点】/【追加触点】。

在当前光标位置前串联插入一个触点。

以下两个位置可以使【串联触点】命令有效：

若光标位于节上，则新触点会添加到程序行首位置。

若光标位于程序行的触点 S1 上，则新触点插入到它之后；若使用【插入触点】，则插入到它之前，如图 138 触点 S2。

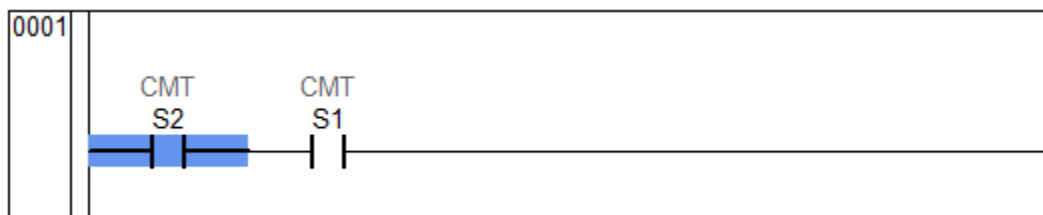


图 138 追加触点



- 一个节中如果触点个数大于 1，则本节中只能添加一个线圈。

10.2.5 并联触点



- 菜单栏：单击【插入】—【并联触点】；
- 工具栏：；
- 快捷键：Alt+I+P。

添加一个触点与已有触点建立并联的电路关系。

若是单个触点对象建立并联关系，只需选中触点，执行【并联触点】命令即可。

若是多个串联触点对象共同与建立并联关系，则需要通过 **Shift** 键将相关的串联触点选中，然后执行【并联触点】命令；如果希望实现图 139 中添加的并联触点（蓝色背景），则添加时需要选中 IN1 和 IN2。

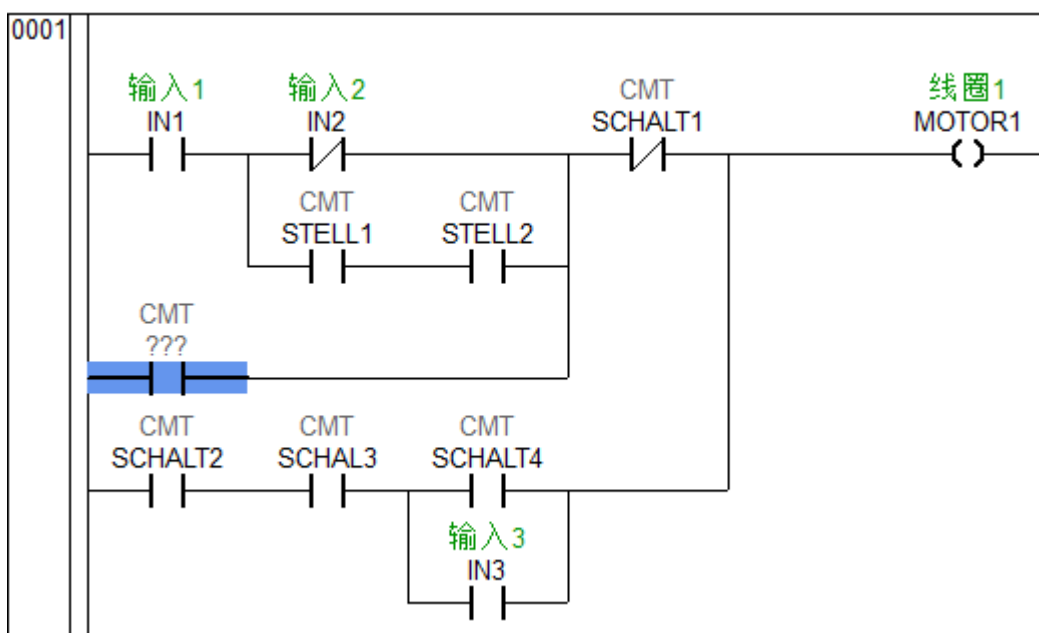


图 139 多个对象添加并联触点

10.2.6 线圈



- 菜单栏：单击【插入】—【线圈】；
- 工具栏：；
- 快捷键：Alt+I+L；
- POU：右击节，单击【线圈】。

在光标所在的节上添加一个输出线圈。

以下两个位置可以使【线圈】命令有效：

若光标位于节上，且该节对应的程序行无线圈，则新线圈会添加到程序行尾端上。

若光标位于线圈上，则新线圈插入到原线圈的下面。如图 140 所示。

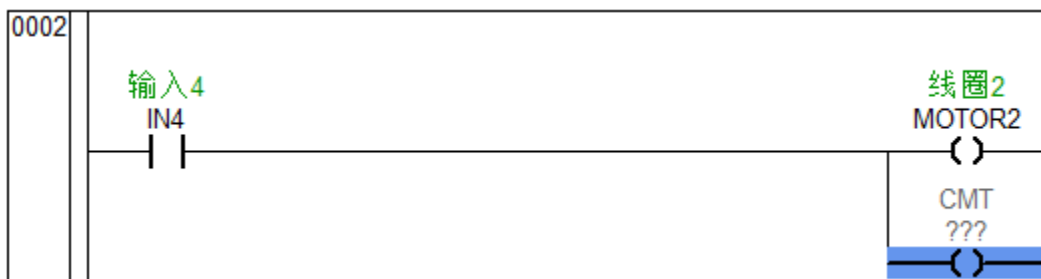


图 140 添加线圈的有效位置



- 一个节只有一个触点时，本节中的线圈可以建立并联关系。
- 一个节不支持线圈与块元件、跳转元件或返回元件同时存在。

10.2.7 块元件



- 菜单栏：单击【插入】—【块元件】；
- 工具栏：；
- 快捷键：Alt+I+B；
- POU：右击节，单击【块元件】。

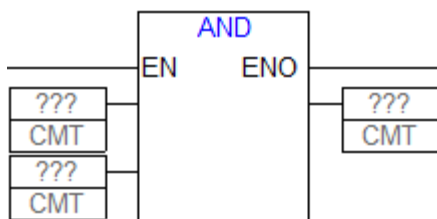


图 141 块元件

双击“AND”可以将其改为有效的功能块、函数、程序名。

多个块元件间只能建立并联关系，不支持串联使用。



- 一个节中添加块元件后，不允许再添加线圈或跳转元件或返回元件。

10.2.8 置位/复位



- 菜单栏：单击【插入】—【置位/复位】；
- 工具栏：；
- 快捷键：Ctrl+T；
- POU：右击线圈，单击【置位/复位】。

如果在线圈上执行这个命令，会得到一个设定线圈。

线圈可以定义成置位或者复位状态。

：置位线圈，通过置位线圈，左侧链路的 ON 状态被复制给右侧链路，则相关的 BOOL 型变量被设置为 ON；并保持该状态直到由复位线圈复位。

：复位线圈，通过复位线圈，左侧链路为 ON 状态时，它的相反状态被复制到右侧链路的布尔变量，即为 OFF 状态；并保持该状态直到由置位线圈置位。

如果您重复执行这个命令，线圈将在设置位、复位和常态之间切换。

10.2.9 置反



- 菜单栏：单击【插入】—【置反】；
- 工具栏：；
- 快捷键：Ctrl+G；
- POU：右击触点/线圈引脚，单击【置反】。

使用这个命令来对触点和线圈进行否定操作。

在线圈的圆括号或触点的直线之间，出现一个斜线（或）。

：反向线圈，通过反向线圈，左侧链路的状态将复制到右侧链路中，左侧链路的相反状态将复制到相关的 BOOL 型变量。如果左侧链路为 OFF，那么右侧链路也为 OFF，相关的 BOOL 变量为 ON。

：常闭触点，如果其变量值为 OFF 时，左侧链路的状态会被传递到右侧链路；反之，右侧链路为 OFF。

通过双重否定可以得到肯定状态。

10.2.10 注释

为了增加程序的可读性，需要给程序、节、变量添加注释，还可以输入跳转标签，指定程序行的跳转方向。

注释具体分类如下：

10.2.10.1 程序注释

对 POU 进行程序注释，输入当前 POU 的执行说明等内容，如图 142 所示。

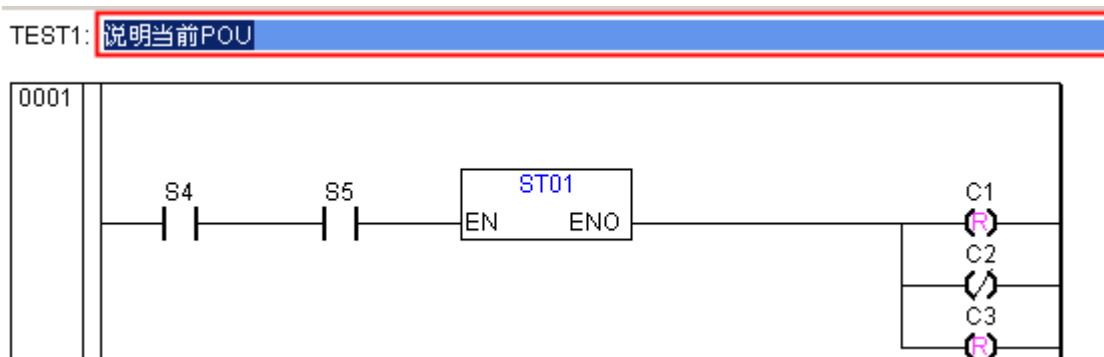


图 142 程序注释

10.2.10.2 网络节点注释

对当前节进行说明。

首先需要勾选【工程】—【选项】—【组态语言】—【LD】中的显示网络节点注释项。

然后在程序区缺省显示该行注释文本为“comment”，单击注释后面显示为蓝色矩形框的注释区，在其中输入注释内容，单击编辑窗口其他位置，可确认输入完成。如图 143 所示。

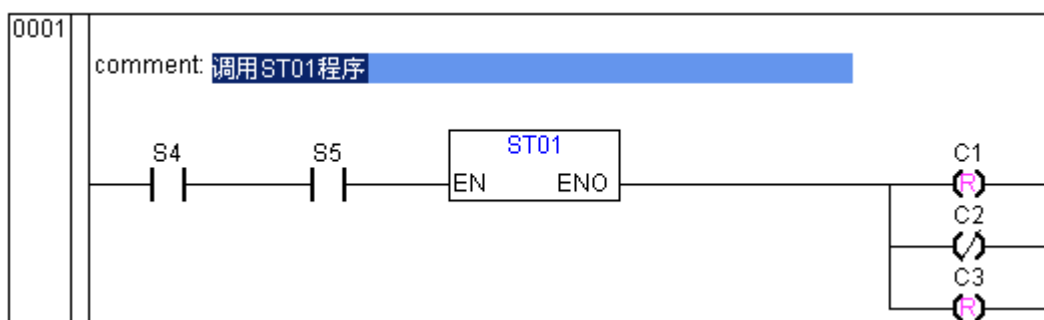


图 143 程序行注释

10.2.10.3 变量注释

允许对变量名进行注释，只有触点、线圈、功能块实例名对应的变量说明会显示在 LD 编程窗口中。在定义变量时，变量注释或变量说明可以在“变量声明”对话框中直接进行注释，也可以在【变量定义】对应的变量组中修改相应变量的说明，如图 144 所示。

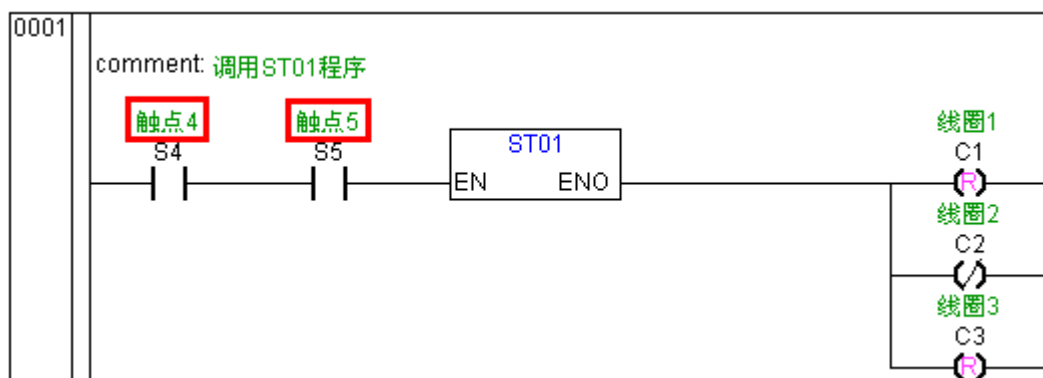


图 144 变量注释

如果希望达到图 144 中的显示效果，请先勾选【工程】—【选项】—【组态语言】—【IEC】中的显示普通类型注释和显示功能块类型注释项。

10.2.10.4 跳转标号

每节都有一个标号，位于此节的首行，缺省为空。鼠标直接单击该节的首行位置，即显示为亮蓝色的矩形输入框，能且只能输入跳转标号。当条件满足后，则跳转至指定标号的相应节。注意标号名称要与跳转元件后的名称一致。如图 145 所示。

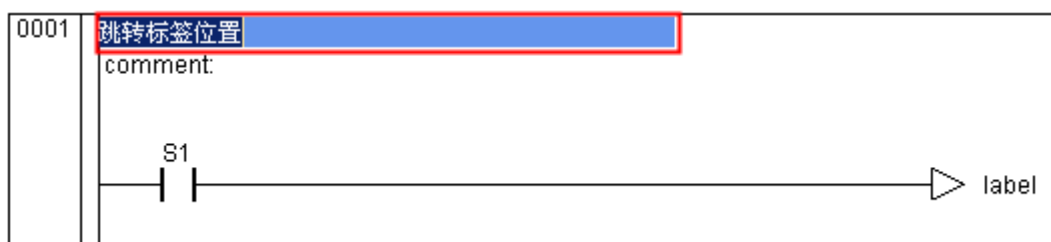


图 145 跳转标号位置

如果注释需要分行显示，用 **Ctrl+Enter** 进行换行。

10.2.11 跳转和返回

跳转和返回都是对程序执行顺序的改变。在正常情况下，控制器将根据主程序中节的顺序进行执行。

10.2.11.1 跳转



- 菜单栏：单击【插入】—【跳转 (J)】；
- 快捷键：Alt+I+J；
- 工具栏：；
- POU：右击节号，单击【跳转 (J)】。

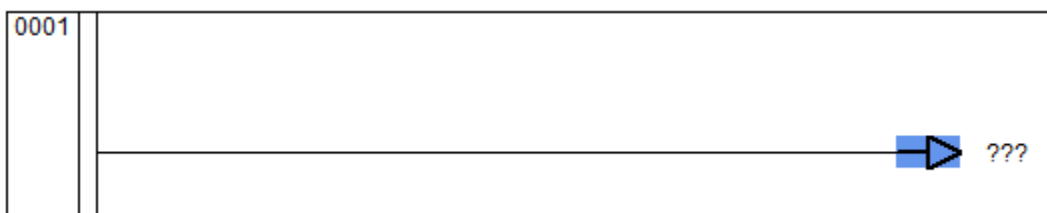


图 146 跳转

插入跳转后，需要输入跳转标号（缺省为“???”）。跳转标号用于识别跳转目的地，如图 147 所示，当跳转条件满足后，直接跳转至 0003 节，不再执行 0002 节的程序。

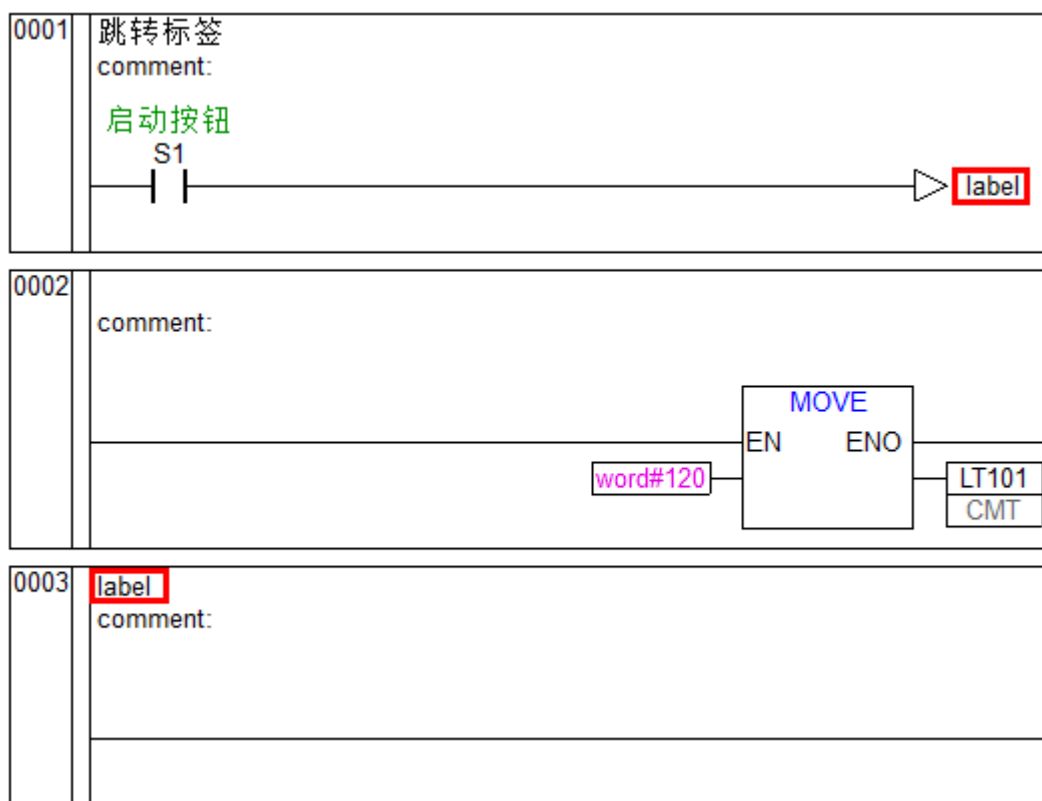


图 147 跳转标号



- 当一个节中添加了块元件或线圈后，不允许再插入跳转元件。

10.2.11.2 返回



- 菜单栏：单击【插入】—【返回（R）】；
- 快捷键：Alt+I+U；
- 工具栏：；
- POU：右击节号，单击【返回（R）】。

当调用 POU 时，可以利用【返回 (R)】命令完成：当条件满足后，被调用的 POU 不再继续执行，而返回到调用的 POU 中。

光标位于节，插入“返回”元素，如图 148 所示。

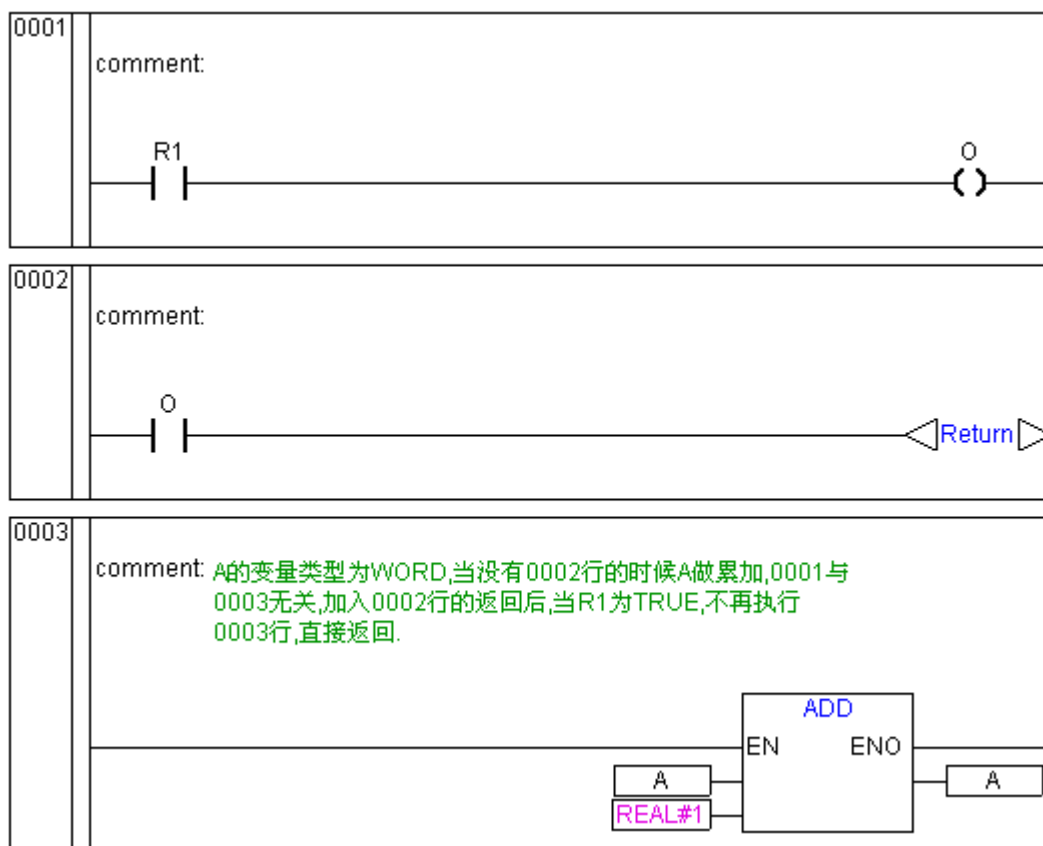


图 148 返回应用实例



- 当一个节中添加了块元件或线圈后，不允许再插入返回元件。

10.2.12 元件的移动

在 LD 的 POU 中可以通过拖放来移动元素到一个不同的位置。

选择期望的元素（触点、线圈、功能模块）拖动它离开当前的位置，在 POU 的所有可以移动到的位置，将会通过蓝色矩形■来指示，移动元素到这些位置中的一个（显示■）并释放鼠标键，元素将插入到新位置。如图 149 所示。

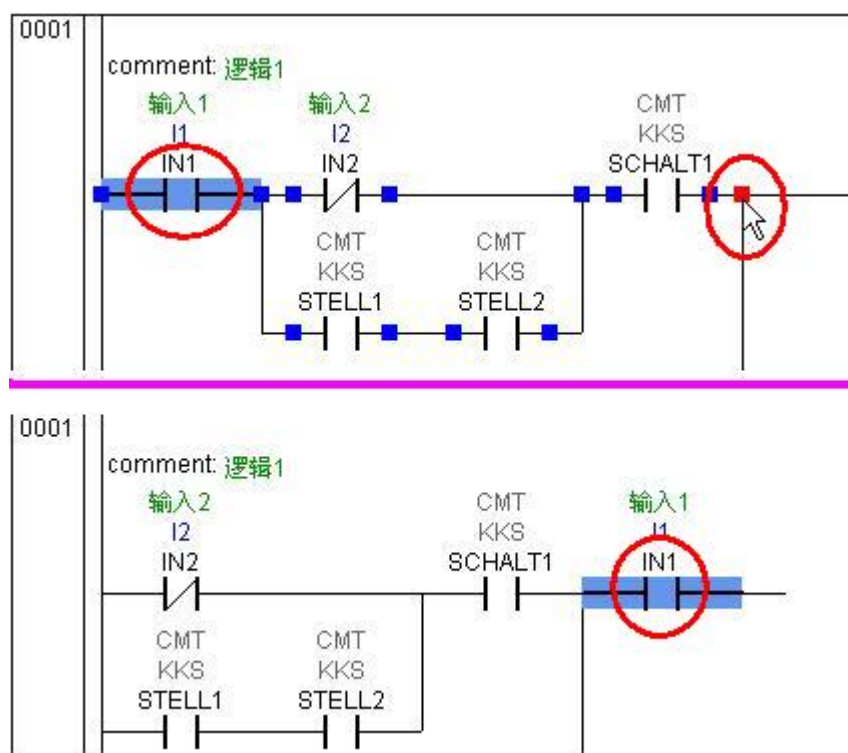


图 149 移动元素

第11章 控制器操作

11.1 控制器操作

控制器操作包含【控制器信息】、【控制器锁】、【网络配置】、【工程升级】、【固件升级】、【日志读取】、【IP 扫描】。

11.1.1 连接控制器

在各标签页进行相关操作前，必须要设置控制器 IP 地址，并成功连接到控制器。否则在各标签页进行操作时会提示设置控制器 IP。

IP 设置界面如图 150 所示。

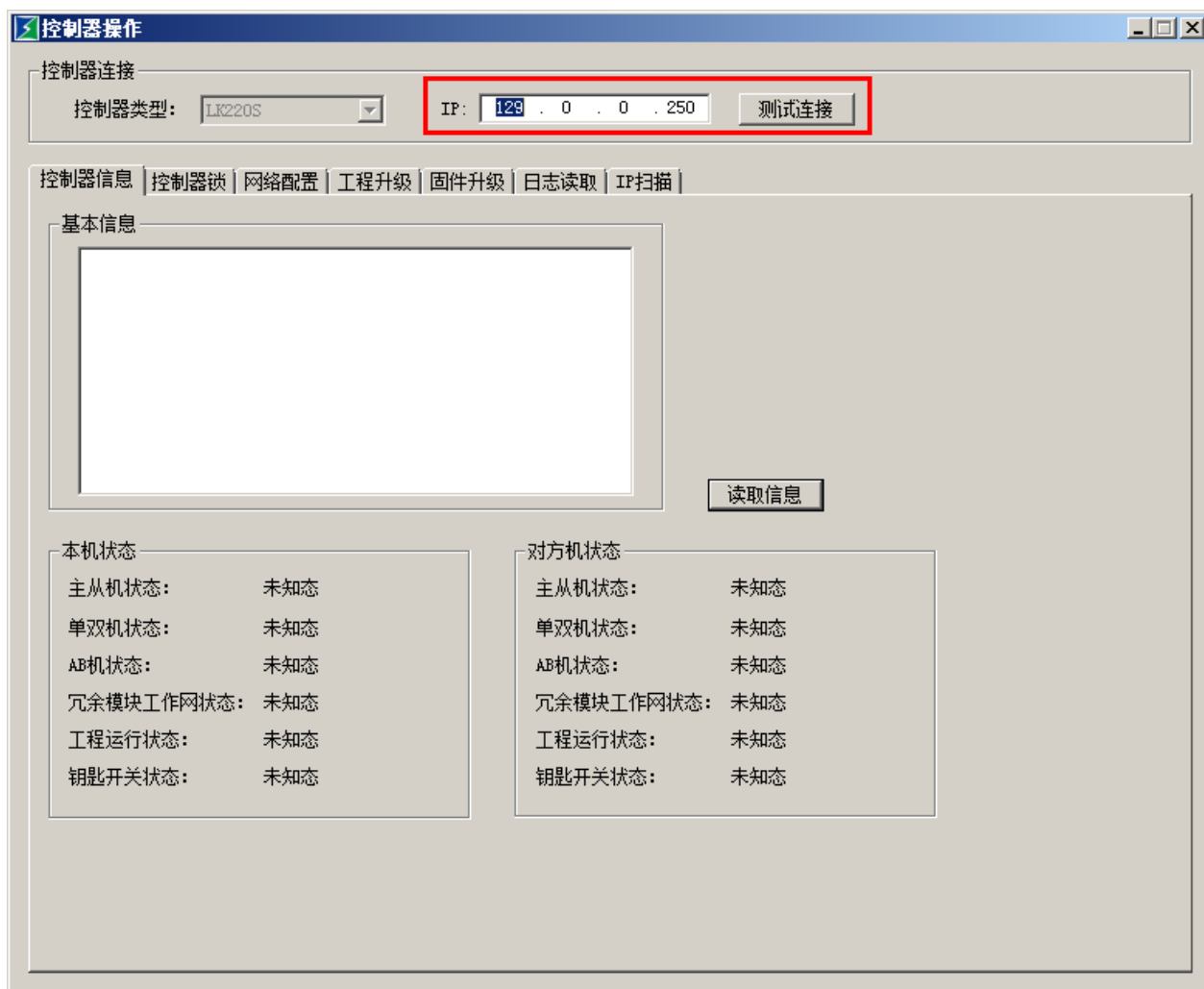


图 150 设置控制器 IP

在 IP 地址编辑框中输入当前控制器的 IP 地址，单击**测试连接**按钮，弹出“控制器连接成功!”的提示框，单击**确定**后，显示当前控制器的加锁状态。此时，则可以在各标签页进行相关操作。

11.1.2 控制器信息

在【控制器信息】标签页中可以读取控制器的版本信息，以及获取当前控制器的主从机状态。

在【控制器信息】标签页中，单击**读取信息**按钮，可以获取当前控制器的版本信息，显示在白色窗口中。同时弹出“读取版本信息成功！”的提示框，如图 151 所示。



图 151 读取控制器版本信息

在【本机状态】和【对方机状态】栏中显示控制器冗余状态信息。各状态的显示信息如表 18 所示。

表 18 控制器冗余状态信息说明

控制器状态	状态信息	说明
主从机状态	未知态	只有当本机的 主从机状态为不合格主机时，需要单击清除按钮进行清除
	初始态	
	硬件 Ready 态	
	双机 Ready 态	
	单机 Ready 态	
	故障态	
	错误态	

控制器状态	状态信息	说明
	工程冗余 工程验证 从机 主机 不合格主机	
单双机状态	未知态 单机 双机	若【本机状态】为单机，则【对方机状态】为 unknown
AB 机状态	未知态 A 机 B 机	显示当前系为 A 机还是 B 机
冗余工作网状态	未知态 第一路光纤 第二路光纤	此状态值显示当前工作的是 A 网还是 B 网
工程运行状态	未知态 正在运行 已停止 挂起	显示工程的运行状态
钥匙开关状态	UNKNOWN RUN REMOTE PRG	显示当前 A、B 机的钥匙开关状态

11.1.3 控制器锁

在【控制器锁】标签页中，实现对当前控制器的加解锁操作。

通过对控制器进行加锁操作，离线状态下禁止对控制器进行相关操作，包括下载工程文件、下装、上传工程文件、监视、清空控制器、IP 修改、工程升级、固件升级。

控制器锁窗口如图 152 所示。

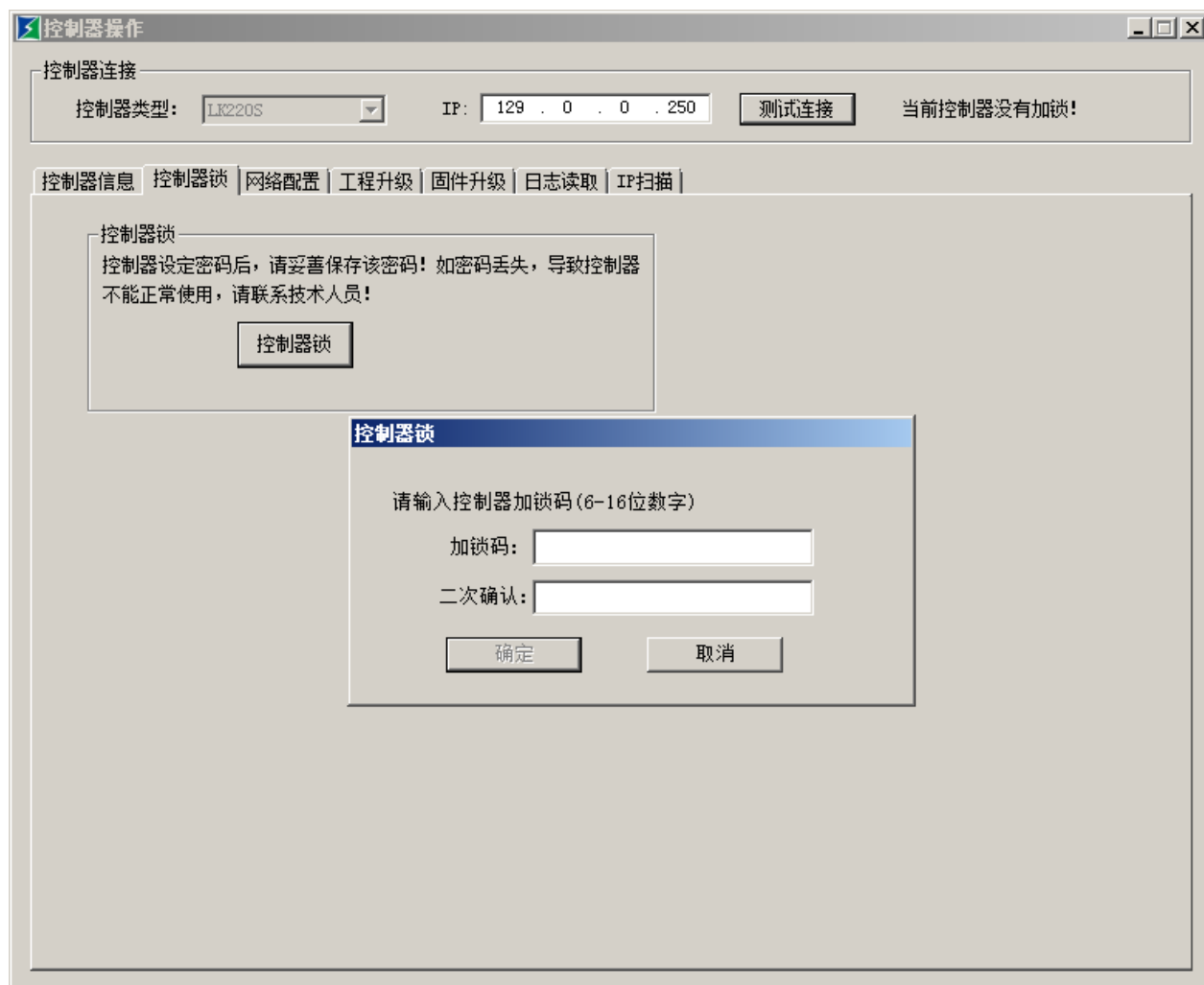


图 152 控制器加锁

单击**控制器锁**按钮，弹出控制器加锁密码输入对话框。输入两次相同的 6-16 位数字**加锁码**后，单击**确定**，完成对控制器加锁操作，并提示控制器加锁成功。

加锁后对控制器进行相关操作时，操作无法完成，并在信息栏出现“当前控制器已加锁，请解锁后重试”。

加锁后，再次单击**控制器锁**按钮可进行解锁。解锁对话框如图 153 所示。输入之前设置的**加锁码**，单击**确定**，完成控制器解锁操作，并提示控制器解锁成功。

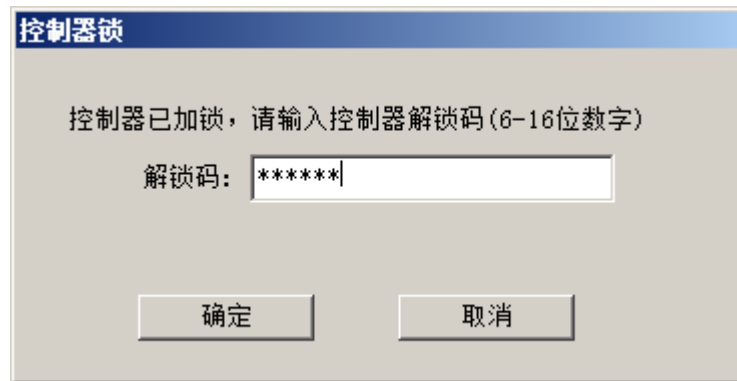


图 153 控制器解锁

11.1.4 修改控制器 IP

通过小工具读取控制器 IP，修改后，再写入控制器。执行该操作前，请确保小工具与控制器建立正常连接。

修改控制器 IP 前，请确保控制器在单机模式，并且工程处于停止状态。

第1步 读取 IP

单击 IP 读取按钮，从控制器读取到两个网口的 IP 地址和掩码。

第2步 修改 IP

修改 IP 地址时，勾选以太网口，输入新的 IP 地址和掩码，单击 IP 修改按钮。

控制器为冗余配置时，请将 A、B 机 IP 地址设置为相同值。当冗余主从机建立后，主机 IP 地址为 IP 设置值，从机的 IP 地址末位为设置值加 1。例如，将 A、B 机两个网口 IP 地址均设置为 128.0.0.250 和 129.0.0.250，当冗余主从机建立后，主机 IP 地址为 128.0.0.250 和 129.0.0.250，从机 IP 地址为 128.0.0.251 和 129.0.0.251，此时，即使发生主从切换，新主机 IP 地址仍然为 128.0.0.250 和 129.0.0.250，从机 IP 地址为 128.0.0.251 和 129.0.0.251。



- 当控制器为冗余配置时，主机 IP 地址的第四字段不能设置为 254，否则，无法与从机正常通讯。
- 当控制器为单机配置时，128.0.0.254 和 129.0.0.254 是控制器保留地址，请勿使用。
- 同一网段内，多套控制器的 IP 地址不能连续。

第3步 重启控制器

修改 IP 地址后，需要重启控制器才能生效。

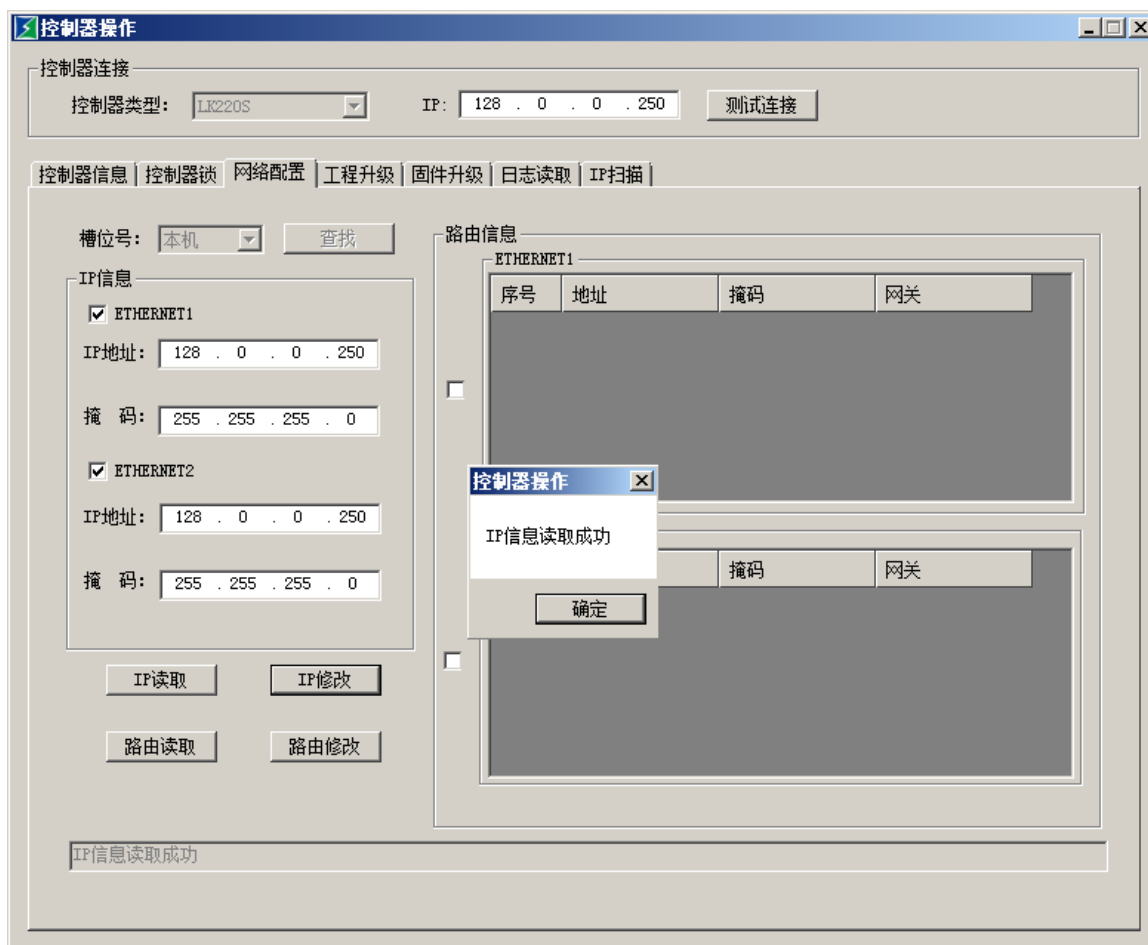


图 154 网络配置对话框

11.1.5 配置路由

执行该操作前，请确保小工具与控制器建立正常连接。

11.1.5.1 添加路由

为控制器网口配置路由信息。

添加路由前，请确保控制器在单机模式，并且工程处于停止状态。

第1步 勾选需要配置的以太网口。

第2步 右击路由表区域，选择**添加**命令，在添加对话框中输入 IP 地址、子网掩码、网关信息，单击**确定**。

第3步 单击**路由修改**按钮。

第4步 重启控制器，路由生效。

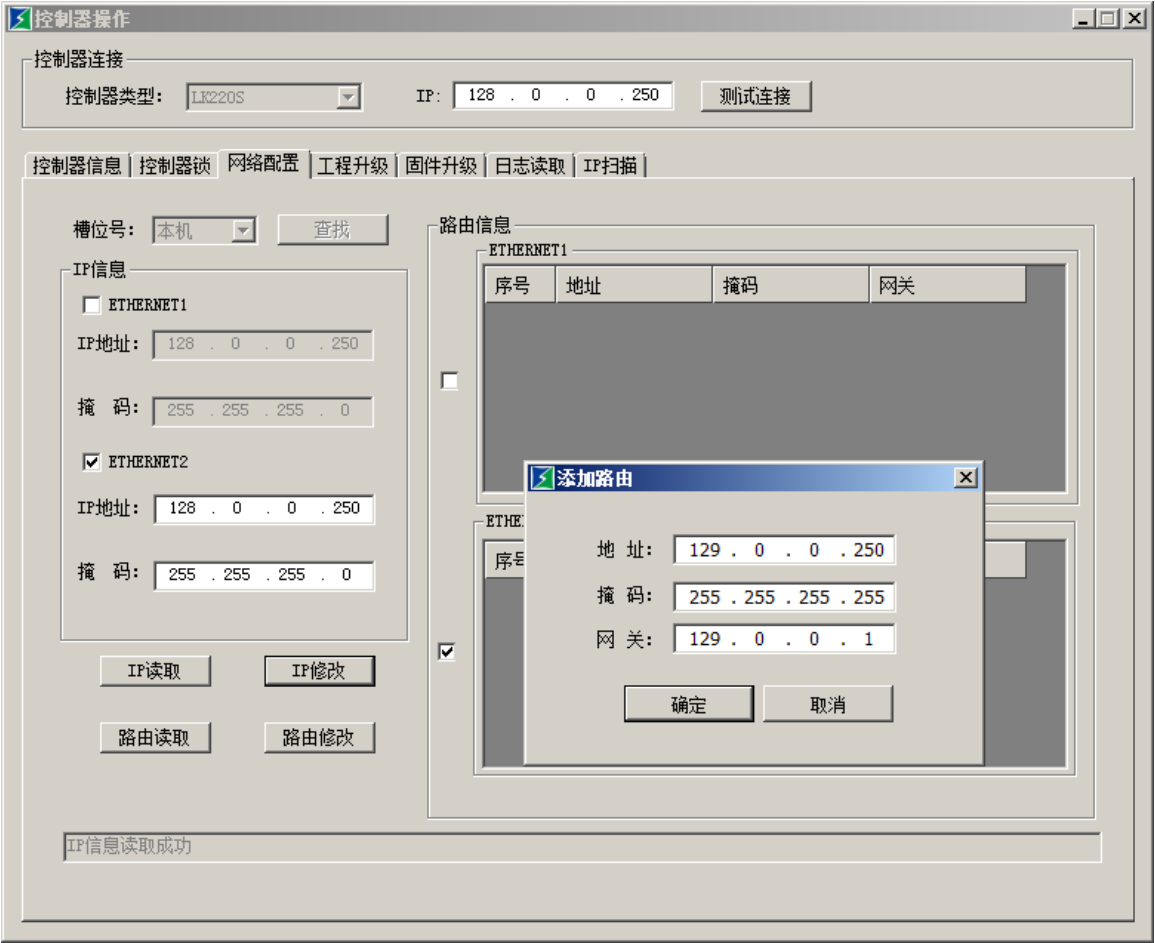


图 155 添加路由

ETHERNET2			
序号	地址	掩码	网关
0001	129.0.0.250	255.255.255.255	129.0.0.1
0002	128.0.0.250	255.255.255.255	128.0.0.1

图 156 路由信息表

11.1.5.2 修改路由

第1步 读取路由

单击路由读取按钮，获取两个以太网口的路由信息，包含 IP 地址、子网掩码和网关。

第2步 修改路由

勾选需要修改的以太网口，输入新的路由信息，单击路由修改按钮。

第3步 重启控制器

修改路由后，需要重启控制器才能生效。



- 路由设置规则同 PC 机路由设置一致。

11.1.6 工程升级

11.1.6.1 概述

当您在没有用户源工程的情况下，如果想升级控制器工程，可通过辅助工具将编译后的工程文件传输到控制器，进行工程更新。

工程升级时，将“.at”工程编译文件和“.SDB”系统文件传输到控制器。其他需要同时具备的文件根据工程实际组态情况确定。

工程升级规则：

- 工程升级时，需要有“.SDB”系统文件，否则，无法进行工程升级。
- 如果工程组态了 SOE 功能，升级时，需要有“.soe”文件，否则，SOE 功能和 NTP 功能均不生效。
- 如果工程组态了 Modbus 功能，升级时，需要有“.Mod”文件，否则，Modbus 通讯无法正常工作。

“.at”和“.soe”文件由工程全编译生成。“SDB”和“.Mod”文件工程下装后生成。升级时，请拷贝对应文件即可。

11.1.6.2 要求

- 准备“.at”和“.SDB”文件
- 辅助工具与控制器已建立连接
- 控制器不能在被监视状态（通过软件或辅助工具）
- 任务处于停止状态

11.1.6.3 步骤

要进行工程升级，请按以下步骤操作：

1. 在“工程升级”标签窗口中，单击按钮.
2. 选择.at 文件，单击升级按钮。
弹出升级确认提示框。
3. 单击是。
开始向控制器上传.at 文件，传输完成后，弹出确认提示框。
4. 单击确定。

11.1.7 固件升级

此标签页实现控制器固件升级和固件文件的备份。

11.1.7.1 控制器固件升级

(1) 选择固件文件

在【升级】标签页中，单击按钮 ，选择控制器固件.bin 文件后，单击**升级**按钮，弹出升级确认提示框，单击**是**，固件文件被上传到控制器中。

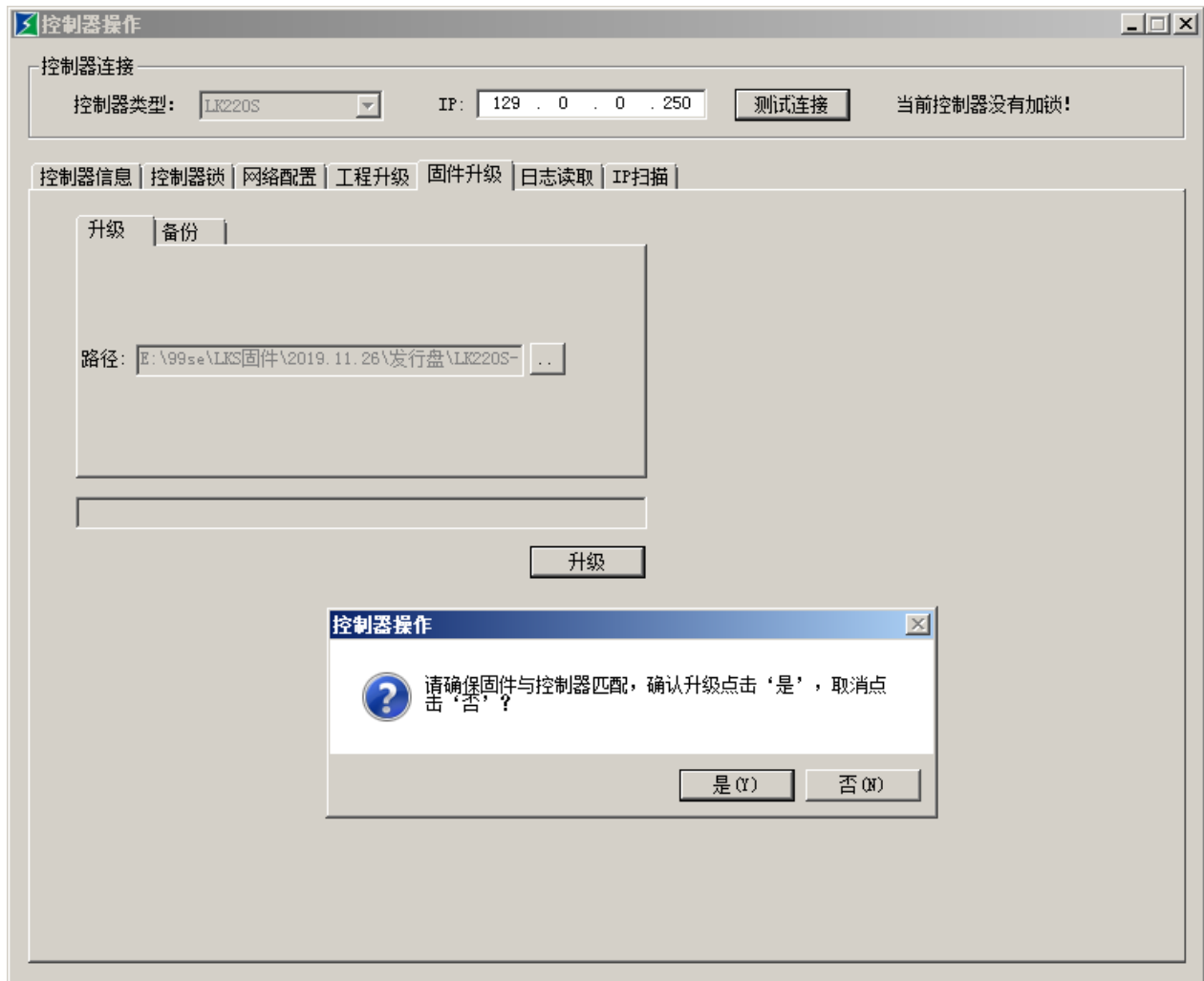


图 157 固件升级对话框

(2) 开始升级

控制器接收到固件文件后开始升级，此时会弹出如图 158 所示的提示框。单击**确定**，等待控制器完成升级即可。



图 158 固件文件传输提示框

(3) 确认固件版本

固件升级完成后，可通过读取控制器信息查看新固件版本是否正确。

11.1.7.2 备份固件文件

在【备份】标签页中，可将控制器中的固件文件备份到本地路径下。用于专业人员分析问题使用。

单击**路径**栏的按钮，选择备份路径，单击**备份**按钮进行备份。完成后提示备份成功。

11.1.8 日志读取

通过此标签页，可以读取控制器中的日志文件，保存到本地路径下，供专业人员分析问题用。

单击**路径**栏的按钮，选择备份路径，单击**读取**按钮，控制器日志文件被下载到本地备份路径下，读取成功时会弹出提示框。

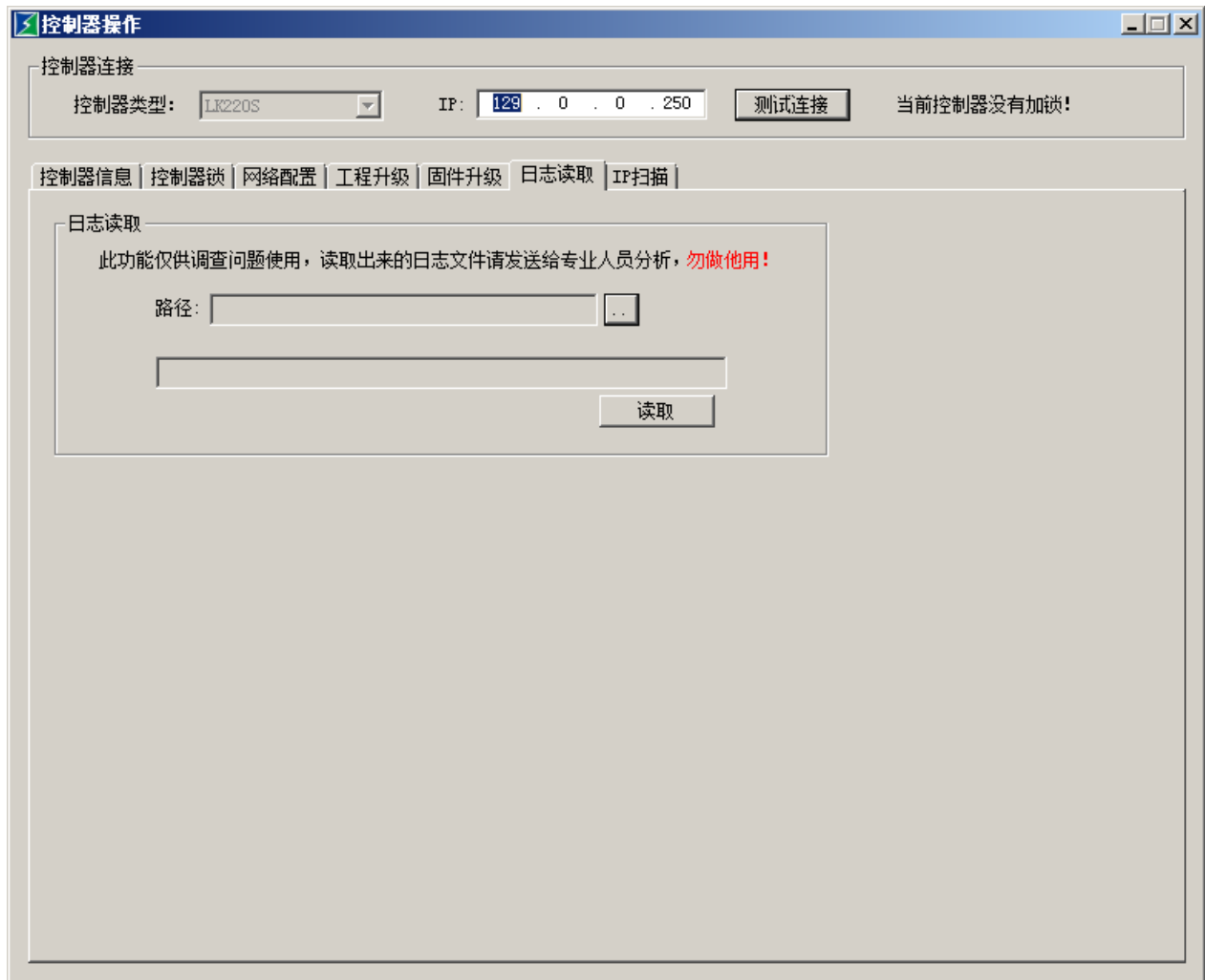


图 159 日志读取对话框

11.1.9 扫描 IP

11.1.9.1 扫描 IP

1. 概述

通过小工具的 IP 扫描功能，可以查找局域网内所有控制器的 IP 地址。需要注意的是，IP 扫描不支持路由级联，如果局域网中接入了路由器，则路由器接入的设备 IP 无法扫描到。

当扫描到 IP 相同的设备时，您可以通过 MAC 地址识别具体设备。再通过修改 IP 地址、子网掩码来重置 IP，解决 IP 重名冲突。

如果 PC 机为多网卡，您需要在扫描前，检查接入网络的网卡型号，并在小工具中进行配置，以确保通讯口正确。

2. 要求

- 已确认网卡型号
- “控制器操作”窗口已打开

3. 步骤

扫描局域网内 IP 地址，请按以下步骤操作：

第1步 单击“IP 扫描”标签页。

第2步 “通讯接口”下拉框中，选择网卡型号。

第3步 单击**查找 CPU** 按钮。

开始扫描，窗口中显示已扫描到的控制器及 IP 地址。

所有控制器查找结束后，提示扫描完成。

第4步 单击**确定**按钮。

4. 下一步

扫描完成后，您可以选择 IP 地址，检测其所属的控制器，或者设置 IP 地址和站名称。

5. 参考信息

[11.1.9.2 修改 IP](#)

[11.1.9.3 检测设备](#)

[11.1.9.4 修改设备名称](#)

11.1.9.2 修改 IP

1. 概述

您可以选择扫描到的 IP 地址，进行重置。

控制器为冗余配置时，请将 A、B 机 IP 地址设置为相同值。当冗余主从机建立后，主机 IP 地址为 IP 设置值，从机的 IP 地址为设置值加 1。

2. 要求

■ “控制器操作”工具已打开

■ IP 地址已被读取

3. 步骤

要修改 IP 地址，请按以下步骤操作：

第1步 在“IP 扫描”标签页中，选择 IP 地址。

第2步 单击**编辑**按钮。

IP 地址栏变为可编辑状态。

第3步 输入 IP 地址、子网掩码。

第4步 单击**下发**按钮。

将弹出提示对话框。

第5步 单击**是**按钮。

控制器 IP 地址被修改，同时，弹出确认对话框。

第6步 单击**确定**按钮。

11.1.9.3 检测设备

1. 概述

您可以通过点亮指示灯的方式，识别控制器的位置。点亮指示灯时，通过 MAC 地址识别所属控制器。

2. 要求

- “控制器操作”工具已打开
- IP 地址已被读取

3. 步骤

要检测控制器，请按以下步骤操作：

第1步 在“IP 扫描”标签页中，选择 IP 地址。

第2步 单击**闪烁指示灯**按钮。

此时，IP 所属控制器的 RUN 灯、ERR 灯、FRC 灯、BAT 灯同时闪烁，30s 后自动停止闪烁。

您也可以在确认控制器后，单击**熄灭**按钮使其停止闪烁。

11.1.9.4 修改设备名称

1. 概述

您可以对局域网内扫描到的控制器进行重命名。

2. 要求

- “控制器操作”工具已打开
- 已查找到控制器

3. 步骤

要修改控制器名称，请按以下步骤操作：

第1步 在“IP 扫描”标签页中，选择控制器一路 IP 地址。

第2步 单击**编辑**按钮。

站名称栏变为可编辑状态。

第3步 输入新名称。

- ☐ 名称为 a~z、A~Z、0~9、“-”、“.” 字符的组合。
- ☐ 名称不得超过 20 个字符。

第4步 单击**下发**按钮。

名称被修改。

11.2 复位



菜单栏：单击【在线】—【复位】。

复位是模拟掉电再上电后的控制器状态，控制器需要将掉电保持区的变量恢复为掉电前的数值，而其他内存区的变量恢复为初始化值。

11.3 冷复位



菜单栏：单击【在线】—【冷复位】。

冷复位是模拟工程下装后的状态，当现场控制器下装程序后，需要把各内存区的数据恢复到初始值时启用冷复位功能。

11.4 热复位



菜单栏：单击【在线】—【热复位】。

热复位是模拟工程重启后各数据区保持重启前的状态。启用热复位功能后，重新加载用户工程逻辑，各内存区的数据保持不变。

11.5 清空控制器

在离线模式下有效，清空 PLC 中的工程，重新初始化 PLC 系统。



菜单栏：单击【在线】—【清空控制器】。

此功能用于清空控制器指定目录，弹出对话框如图 160 所示。

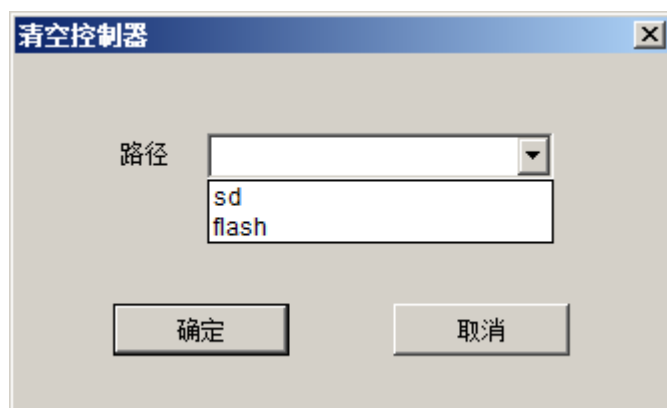


图 160 清空控制器

- 选择 SD 卡路径，则删除 SD 目录下的所有文件，删除后不能恢复。

- 选择 Flash 路径，则删除运行程序、用户程序和用户文件，删除后不能恢复。

选择路径后，单击**确定**，会弹出删除提示框。



- 工程执行清空控制器后，需要对控制器进行复位或重启。

11.6 控制器校时



菜单栏：单击**【在线】—【控制器校时】**。

该功能在线状态下可用。在弹出的窗口中，上方显示的是控制器当前时间，下方为设置控制器新时间。新时间可手动输入，然后单击**设置**按钮生效。或者直接单击**同步 PC 时钟**按钮，将控制器时间设置为 PC 时间。控制器校时界面如图 161 所示。

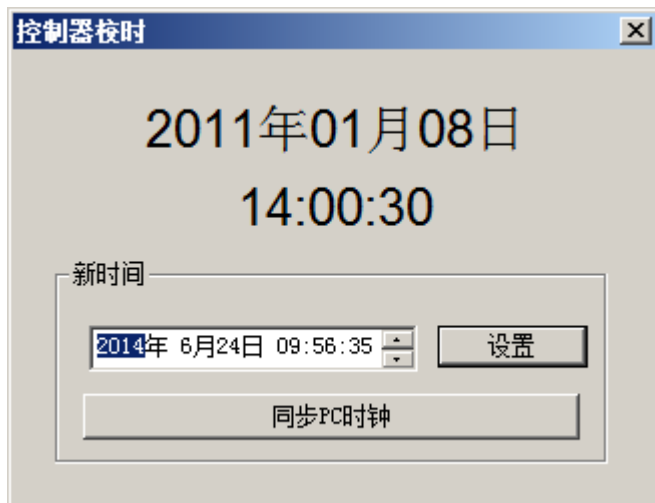


图 161 控制器校时

支持的时间范围为：

- ☐ 年：1971~2036
- ☐ 月：1~12
- ☐ 日：1~31
- ☐ 时：0~23
- ☐ 分：0~59
- ☐ 秒：0~59

第12章 编译

工程组态完成后,要进行编译操作,目的是检查组态是否存在错误,验证程序的正确性和可执行性,将用户组态的程序转换成可下装到控制器中的文件,并在“信息窗口”中显示编译结果。

编译分为全编译和增量编译两种方式。

12.1 增量编译

- 增量编译只对修改和追加的内容进行编译。以下操作将触发增量编译：POU 的修改（增加或删除 POU、修改已有 POU 中的逻辑）。
- 变量的修改（增加或删除变量定义、修改已有变量定义）。
- 任务属性修改：增加调用 POU、任务属性值改变。

通过以下方式执行增量编译：



- 菜单栏：单击【工程】—【编译】；
- 工具栏：；
- 快捷键：**F11**。

12.2 全编译

全编译时硬件配置信息重新生成，所有变量重新分配，POU 重新翻译等，生成新的组态逻辑信息和硬件配置信息。以下操作将触发全编译：

- 新建工程后首次执行【编译】操作。
- 编译时，.iec 或.tmp 文件不完整或者损坏。
- 打开旧版本工程文件引起 TrgVersion 文件升级。
- 下装控制器时，.iec 或者.tmp 文件不完整。
- 直接执行【全编译】命令。
- 硬件配置改变，执行【编译】操作。

通过以下方式执行全编译：



- 菜单栏：单击【工程】—【全编译】；
- 工具栏：；

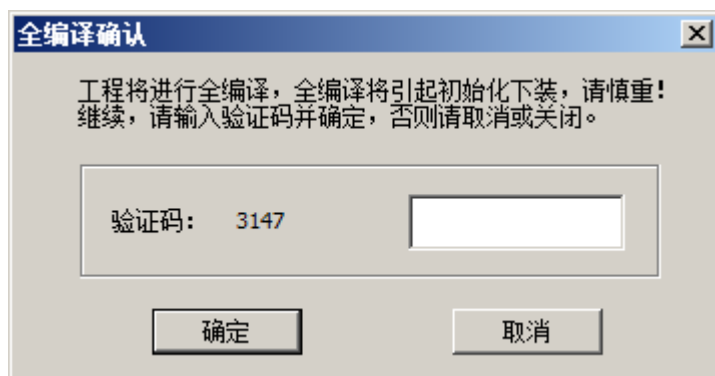


图 162 全编译确认

12.3 编译结果

编译结果显示在信息窗口中，红色字体标注编译错误的信息，双击可直接定位错误进行修改。

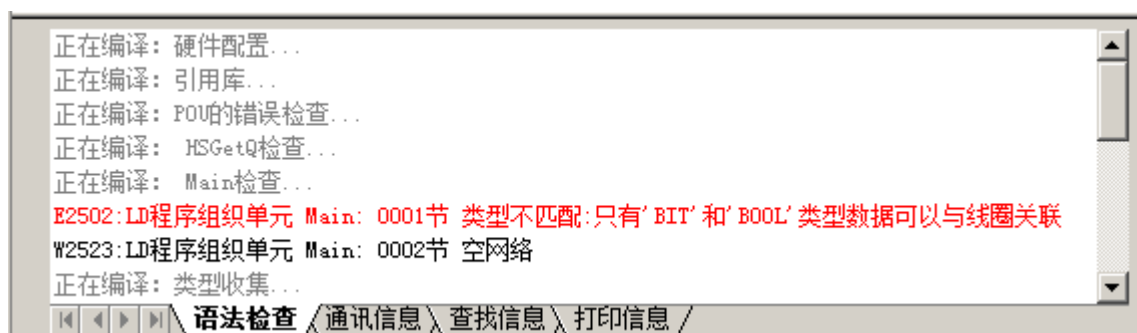


图 163 编译结果

12.4 编译问题分析

1. 只有 BIT 和 BOOL 类型数据可以...

原因：“置反”、“置位/复位”的对象类型以及一些特殊元件（例如跳转元件）或 LD 语言中的输入必须为 BIT 或 BOOL 型。

示例：编辑区程序为 , 其中输入变量 W 为 DINT 类型，后接跳转元件，编译出现错误提示**类型不匹配:只有' BIT '和' BOOL '类型数据可以作为输入**。

2. 标识符'%s'未定义

分析：“%s”所代表变量没有定义，需补充变量定义。

3. '%s'不是类型'%s'的成员

分析：程序中引用了非某类型的成员变量。

示例：

程序如图 164 所示，EXAMPLE 包含成员变量 a、g、w，网络线下的程序段引用了非 EXAMPLE 成员变量的 e，编译时提示：**关联数据: 'e' 不是类型' EXAMPLE '的成员**。

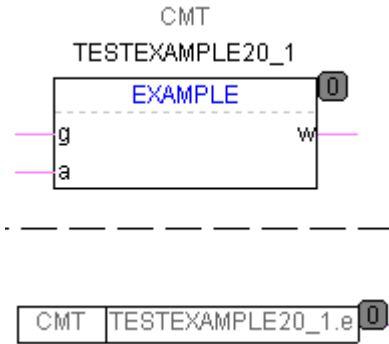


图 164 错误实例

4. 只有...变量后面可以接 ...

分析：特殊符号例如 ‘.’ 和 ‘[’，只可跟在特殊类型的变量后，使用时请注意。

5. 跳转位置（或标号）为空

分析：跳转位置和标号需要有明确标识，不可为空。

示例：程序区标号元件，未填写标识内容，编译后提示 **关联数据: 标号为空**。

6. ...后面不能出现...

分析：某些符号，常量或变量后不能连接其他信息或特定字符。

示例：编辑区程序 `CMT 5TES`，编译出现错误提示 **关联数据: 常量'5' 后面不能出现其他信息**。

7. ...超出...范围或（警告）使用变量下标访问数组元素，有可能导致数组越界，引起控制器异常。 强烈建议在程序中增加数组越界保护逻辑！

分析：编辑程序时，请注意数据类型、数组下标的范围。在编写逻辑时，可以使用常量下标或者变量下标来访问数组元素。这两种下标都有可能出现数组越界。对于常量下标越界，在执行编译操作后，系统会给出错误信息 **关联数据: 下标:'11' 超过对应数组定义范围**；对于变量下标，在执行编译操作后，系统会给出警告信息。

使用变量下标访问数组元素，有可能导致数组越界，引起控制器异常。强烈建议在程序中增加数组越界保护逻辑！。
编辑程序时，请注意数组下标可用范围。

示例 1:



图 165 警告实例

8. 缺少与‘[’匹配的‘]’或只有无符号整型数据可以作为数组下标‘%s’

分析：[]需成对使用，数组下标必须为整型。

示例：数组元素

CMT	g[1	1
CMT	g[2.1]	0

 作为输入，编译提示 **关联数据:缺少与‘[’匹配的‘]’** 和 **关联数据:只有无符号整型数据可以作为数组下标‘2.1’**。

9. 不能将‘%s’类型转换为‘%s’类型

分析：变量类型不匹配

示例：程序

CMT	w	1
-----	---	---

CMT	
TESTEXAMPLE20_1	
EXAMPLE	
g	3
f	

，定义功能块引脚 g 为 DWORD 类型，

实际输入 w 为 BOOL 类型，则编译提示**类型不匹配:不能将‘BOOL’类型转换为‘DWORD’类型**。

10. 无效数据 ‘%s’ 或‘%s’类型不能参与‘%s’操作

分析：数据为无意义符号或常量，提示无效数据。某些操作对于输入数据类型有确定要求，非要求类型数据不能参与操作。

示例：幂操作程序

CMT	g	0
CMT	1~~	0

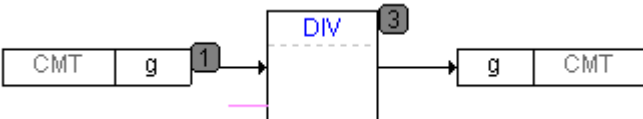
EXPT		0
------	--	---

，g 为 INT 型数据，编译提示错误

关联数据:无效数据‘1~~’ 以及 **类型不匹配:‘INT’类型不能参与‘EXPT’操作**。

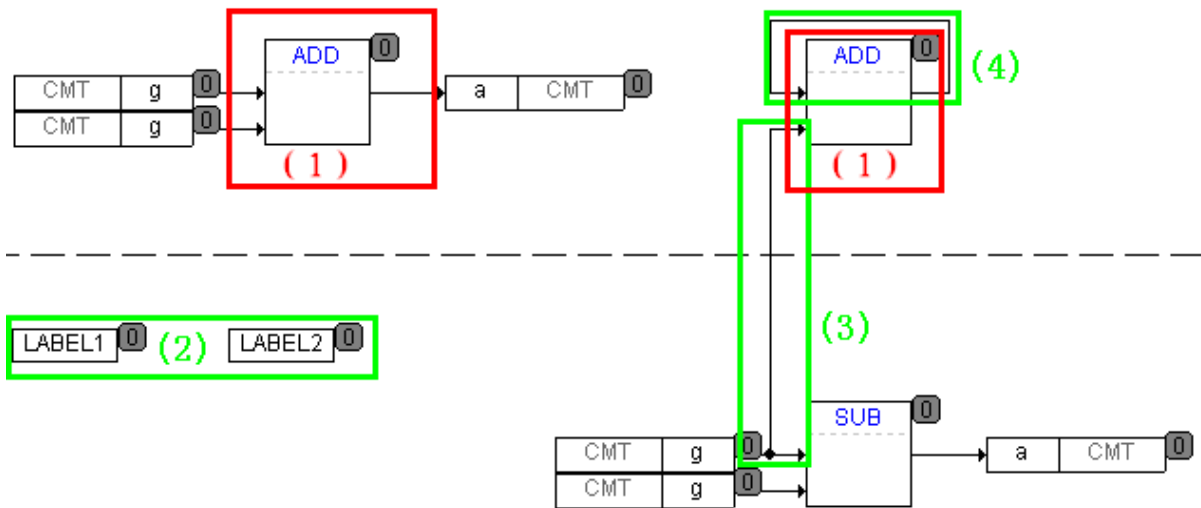
11. 只读数据‘%s’不能被赋值或缺少数据

分析：INPUT 类型变量为只读变量，不能对其进行赋值等写入操作。

示例：程序 ，g 为 INPUT 变量，并悬空一

输入引脚，编译提示 **类型不匹配:缺少数据** 和 **只读数据'g'不能被赋值**。

12. 网络中的...个数不能超过一个



13. 网络中存在元件跨网络、块元件不支持环路或未关联'%s'类型功能块变量

分析：'%s'代表的功能块未和该类型的变量关联。

12.5 反编译

反编译技术是 Safety FA-AutoThink 组态软件的关键技术之一。在 Safety FA-AutoThink 中，通过反编译技术，实现对目标代码的逆向翻译并经过复杂的重构，还原组态逻辑，再与源工程进行比对，从而保证下装到控制器的目标代码和源工程保持一致；如果不一致，反编译失败，无法下装。因此，下装到控制器的过程是安全的。

该功能由 Safety FA-AutoThink 软件提供，且在增量编译或全编译完成后自动进行反编译。在执行反编译操作后，系统将源程序和反编译生成的程序展示给用户，然后供用户检查程序的正确性、完整性和一致性，即进行“组态验证”。

反编译的具体过程如下：

1. Safety FA-AutoThink 反编译

Safety FA-AutoThink 读取编译后的工程文件，进行反编译。反编译成功时，会提示用户进行检查。如图 166 所示。

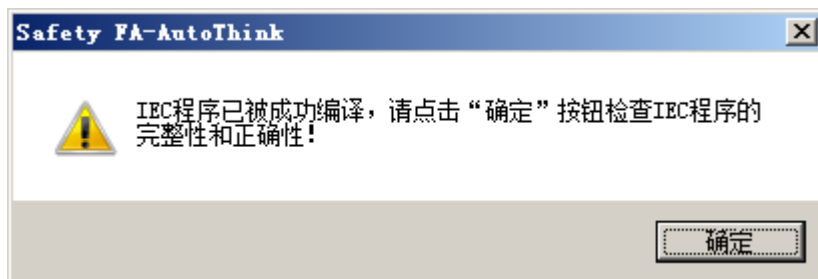


图 166 检查提示框

用户看到这个提示后，选择**确定**按钮。进行下一步操作。

2. 用户确认

在【组态逻辑验证程序】节点下为通过 Safety FA-AutoThink 反编译的各个程序子节点，用户可双击任意程序子节点在工作区域中打开其反编译结果，显示与本地工程相比较的程序对照窗口，以便确认二者是否一致。

例如图 167 所示，在工作区域中，如果原程序的编辑窗口处于打开状态，则双击【组态逻辑验证程序】树中的相应的 POU 子节点，其反编译结果显示在原程序的程序区的右侧部分。关闭该 POU 编辑窗口，此对照结果也关闭。



图 167 反编译结果对照窗口

通过：当原程序与反编译生成的程序一致时，且用户希望继续下装时，选择此按钮。

未通过：当原程序与反编译生成的程序不一致，或用户希望放弃本次下装操作时，选择此按钮。

编译通过后，软件读取模块的用户参数信息供用户确认，下装时，需进行回读信息确认。



- 工程投入使用前，请进行调试，以排除人为错误或非安全环境引发的其它错误。

第13章 下装

工程成功编译后，就可以下装到控制器了。下装前，需要建立工程师站和控制器的连接。

13.1 通讯设置

单击【在线】-【通讯设置】命令，即可设置当前下装控制器的 IP 地址，心跳时间、重连时间和端口号均采用默认值，不允许修改，最后单击**确定**按钮，完成通讯设置，如图 168 所示。



图 168 通讯设置

13.2 下装操作

上述下装准备完成后，可以进行下装操作。

下装前请检查拨码开关位置，如果在 RUN 状态，禁止下装。

下列任意项发生时，都会触发下装。

- 控制器无工程
- 控制器工程与本地工程不一致
- 工程编译



- 菜单栏：单击【在线】—【下装】；
- 工具栏：；
- 快捷键：**F5**。

增量编译后单击【下装】，将执行增量下装，全编译后单击【下装】，将执行初始化下装。下装时，需要确认回读的模块用户参数是否验证正确，单击“是”，确认通过，继续执行下装操作。单击“否”，需检查模块用户参数是否配置正确，无误后重新编译。

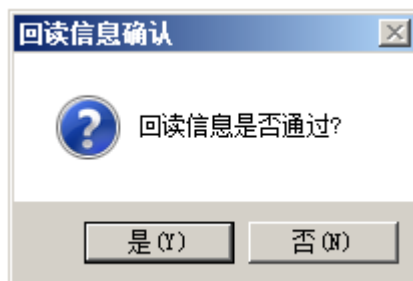


图 169 模块用户参数回读确认

初始化下装过程中，安全 IO 全部导向安全侧，下装完成，控制器运行后，所有 IO 恢复正常值。下装过程中请关注【信息窗口】的相关提示。



- 下装前，确保没有工程文件被打开，否则，下装时备份工程文件失败，此时，用户需要手动拷贝工程文件进行备份。

第14章 调试

14.1 仿真模式

仿真模式主要目的是利用实验环境来模拟调试用户程序逻辑的正确性，方便了联锁逻辑的修改和调试，为生产现场的联机调试提供了前提。在仿真期间，程序是在本地运行的仿真软件中执行运算过程的，与控制器实物不连接。



- 菜单栏：单击【在线】—【仿真模式】；
- 工具栏：。

进入仿真模式后，程序自动运行，仿真界面类似于操作模式下监视状态界面。

在仿真模式下，用户可查看功能块、数组类型变量对应成员的变量值，周期读取仿真软件中变量的值，功能块实例界面各参数也根据仿真软件中的值变化。双击功能块实例所在区域，弹出变量在线值列表如图 170 所示，还可以通过双击复杂变量打开其监视窗口。软件每 500ms 读取一次仿真软件中变量的值，并实时显示到表中。

test1.test1CTU10_1					
序号	变量名	变量说明	变量类型	在线值	区域
0001	CU		BOOL	FALSE	G区
0002	RESET		BOOL	FALSE	G区
0003	PV		WORD	0	G区
0004	Q		BOOL	TRUE	G区
0005	CV		WORD	12666	R区
0006	M		BOOL	FALSE	R区

图 170 变量在线值查看

14.2 在线操作

14.2.1 写变量

写变量是指对监视状态的程序中的变量、通道变量进行写入新数值的操作。

仅 Level 0 级用户有该权限。POU 示例的编辑窗口如图 171 所示，简单变量在线值在变量区及程序区直接显示；复杂变量可通过双击其变量名或程序区的块元件，打开成员变量对话框如图 172 所示，若其成员变量为简单变量，即可查看其成员变量的在线值。

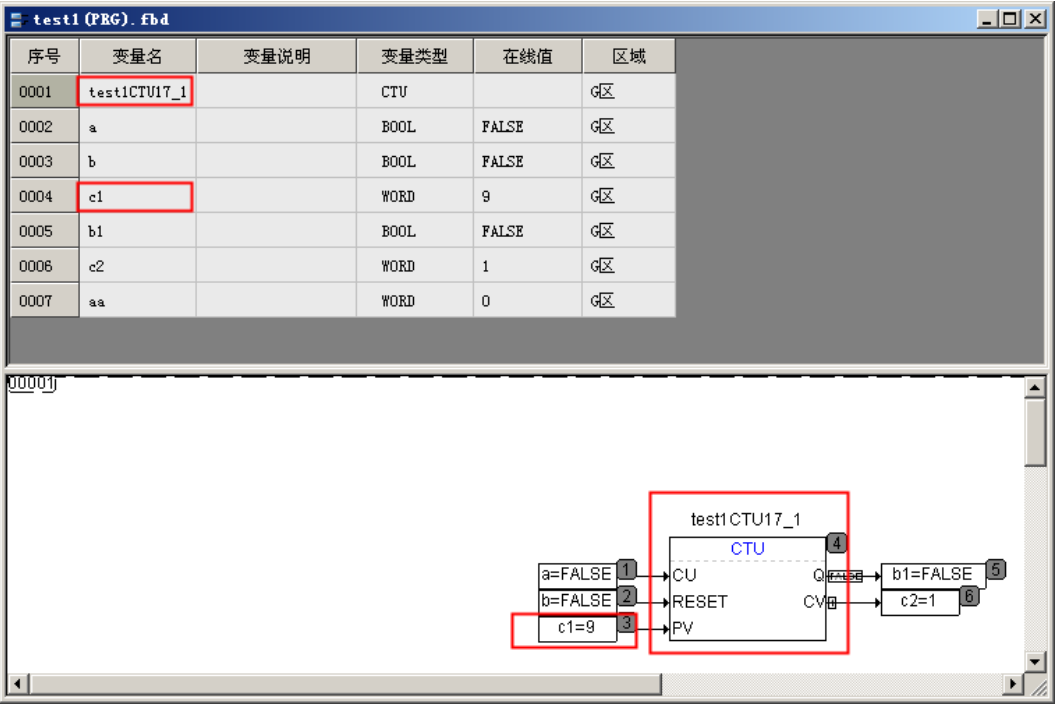


图 171 操作模式



图 172 复杂变量在线值

14.2.1.1 变量值单个写入

在 POU 或变量列表中，双击目标变量名，可以打开“调试变量”对话框。您可以在输入变量值处输入新值，通过写入使新值生效。如图 173 所示的实例中，test1CTU17_1 变量 PV 项的新值 1000 欲将被写入到控制器中。

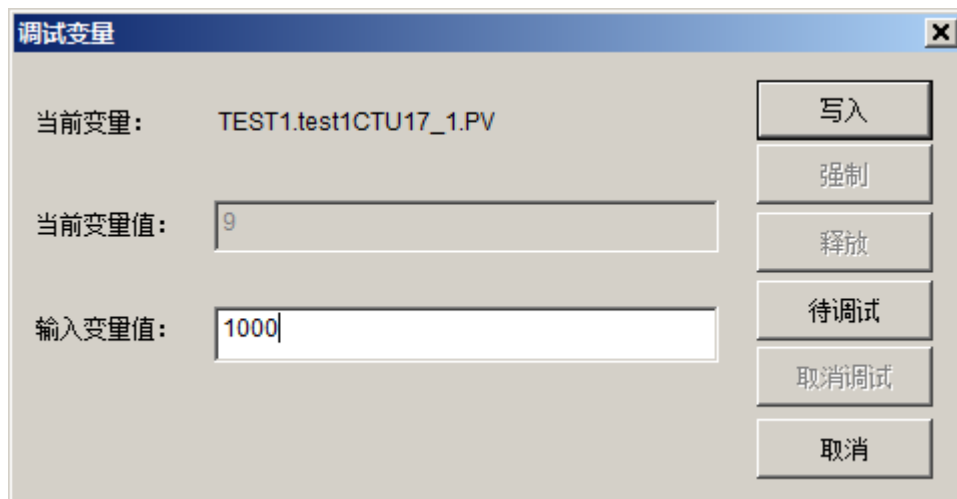


图 173 写入变量实例

14.2.1.2 变量值批量写入

批量写入与单个写入的操作顺序及内容基本一致，唯一不同之处在于，批量写入在新值输入完成后，不是直接选择**写入**，而是选择**待调试**，直到所有变量的新值都输入完成后，才通过**写入**使得待调试的值同时生效。

TEST1.test1CTU17_1					
序号	变量名	变量说明	变量类型	在线值	区域
0001	CU		BOOL	FALSE	Q区
0002	RESET		BOOL	FALSE	Q区
0003	PV		WORD	9 < 1000 >	Q区
0004	Q		BOOL	FALSE < TRUE >	Q区
0005	CV		WORD	1	R区
0006	M		BOOL	FALSE < TRUE >	R区

图 174 变量处于待调试状态

在单击**写入**之前，如果想放弃对某个变量的调试，可以在该变量的“调试变量”对话框中，通过**取消调试**解除它的待调试状态。



- 若输入的在线值非法，例如类型不匹配等，则**写入**与**待调试**按钮失效。
- 写入变量值仅一次有效，写入后的值会随着程序的周期运行被改变。
- 控制器处于 RUN 状态时，在线写入变量，信息输出窗口提示不允许写入。

第15章 在线监视

下装后，系统可以进行在线监视。、在线监视界面如下：

变量变化：进入监视状态后，Safety FA-AutoThink 软件周期地读取控制器中变量的值，监视变量在控制器中的运行情况。软件默认监视周期为 500ms，即每 500ms 读取一次控制器中变量的值，并按新值更新监视界面。可以创建监视列表对重要的变量值变化进行集中监视。

硬件配置：在硬件配置窗口中，可以监视通道的采集值。

程序运算：在 POU 窗口可以跟踪程序的运算状况。

15.1 运行



- 菜单栏：单击【在线】—【运行】；
- 工具栏：

监视状态下，使用该命令进入在线运行。如果任务属性的启动类型选择为自动，编译下装后任务自动运行，无需单击运行。

15.2 停止



- 菜单栏：单击【在线】—【停止】。
- 工具栏：

停止程序的运算状态，各个变量保持运算停止前的值，当再次执行时，从头开始执行逻辑程序。

15.3 暂停

暂停程序的运算状态，各个变量保持运算暂停前的值，当再次执行时，从逻辑暂停的地方继续执行。



- 工具栏：

15.4 监视

可以通过【监视】命令对运算的逻辑进行监视。



- 菜单栏：单击【在线】—【监视】；
- 工具栏：

监视状态进入和退出的操作命令完全相同。需要退出监视状态时，将上面的命令复选一次即可。

15.4.1 变量与程序监视

Safety FA-AutoThink 周期读取控制器中变量的值，软件工作区域中打开的各个 POU 编辑窗口中的各变量也显示根据实际控制器中的变化值，如图 175 所示，变量的实时值反映在变量表的“在线值”单元格中和程序区的输入输出端子处。

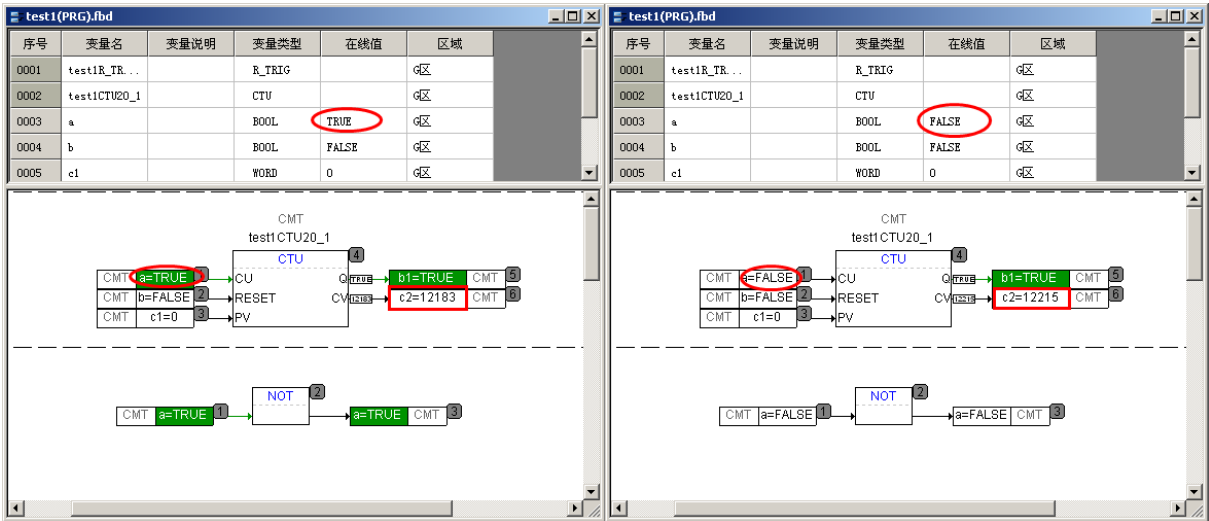


图 175 功能块实例界面参数变化图

监视状态下，可以查看功能块、数组类型变量对应成员变量的在线值。此处以功能块为例，双击功能块实例所在区域，弹出变量在线值表如图 176 所示，软件每 500ms 读取一次控制器中变量的值，并实时显示到表中。同时在线状态可以查看此变量值。

test1.test1CTU10_1					
序号	变量名	变量说明	变量类型	在线值	区域
0001	CU		BOOL	FALSE	Q区
0002	RESET		BOOL	FALSE	Q区
0003	PV		WORD	0	Q区
0004	Q		BOOL	TRUE	Q区
0005	CV		WORD	12666	R区
0006	M		BOOL	FALSE	R区

图 176 变量在线值查看

若列表中有复杂型变量，则可通过双击变量名，使其成员变量信息显示。

15.4.2 通道的在线值和质量位

1~8 为通道在线值，显示通道实时数据。9~16 为通道质量位，显示通道的好坏，TRUE 为通道坏，FALSE 为通道正常。

通道号	通道名称	通道类型	通道地址	通道说明
1	DP10_2_1_2_1	BOOL	%IX0.0	
2	DP10_2_1_2_2	BOOL	%IX0.1	
3	DP10_2_1_2_3	BOOL	%IX0.2	
4	DP10_2_1_2_4	BOOL	%IX0.3	
5	DP10_2_1_2_5	BOOL	%IX0.4	
6	DP10_2_1_2_6	BOOL	%IX0.5	
7	DP10_2_1_2_7	BOOL	%IX0.6	
8	DP10_2_1_2_8	BOOL	%IX0.7	
9	Q_DP10_2_1_2_1	BOOL	%IX1.0	
10	Q_DP10_2_1_2_2	BOOL	%IX1.1	
11	Q_DP10_2_1_2_3	BOOL	%IX1.2	
12	Q_DP10_2_1_2_4	BOOL	%IX1.3	
13	Q_DP10_2_1_2_5	BOOL	%IX1.4	
14	Q_DP10_2_1_2_6	BOOL	%IX1.5	
15	Q_DP10_2_1_2_7	BOOL	%IX1.6	
16	Q_DP10_2_1_2_8	BOOL	%IX1.7	

图 177 通道参数的在线值

15.5 查看操作日志

操作日志用于记录工程师在调试过程中的所有操作。



- 菜单栏：单击【工程】—【查看操作日志】。

窗口中可以记录程序的在线操作信息：操作时间、操作内容，并统计操作总次数。日志实例如图 178 所示。



图 178 “操作日志”窗口

每个操作日志文件最多可以存放 5000 条信息，当日志信息到达 5000 条，最新记录的日志信息将存储到另一个新的操作日志文件中。

日志文件名称为“工程名+日期+空格+时间+.log”，例如：ExamplePro2017-8-31 14-18-30.log。日志文件的保存路径与该工程文件的保存路径一致。

15.6 使能 SOE 功能

15.6.1 概述

控制器支持 SOE 功能，您可以通过使能变量的 SOE 功能，实时获取变量历史事件信息。目前，SOE 事件变量包括 PRG 变量、全局变量以及 LK630S 通道变量。

15.6.2 要求

已创建变量

15.6.3 步骤

使能变量 SOE 功能，请按以下步骤操作：

第1步 在变量定义区或模块“用户参数”，使能 SOE 属性。

Main (PRG).ld						
序号	变量名	变量说明	变量类型	初始值	区域	SOE使能
0001	g1		BOOL	FALSE	G区	☒
0002	g2		BOOL	FALSE	G区	☒
0003	g3		BOOL	FALSE	G区	☒

图 179 示例_PRG 变量 SOE 使能



图 180 示例_通道变量 SOE 使能

第2步 下装工程。

15.7 查看 SOE 信息

15.7.1 概述

控制器实时记录已使能 SOE 功能的 BOOL 型变量的历史事件信息。每当值从 TRUE 变为 FALSE，或从 FALSE 变为 TRUE，均会记录一次值跳变，可保存 6000 条事件信息。

当您需要查看历史信息时，可从控制器读取 SOE 信息。SOE 信息将以 “.xls” 格式被保存在本地路径。

事件信息详见下表。

事件信息	信息说明
跳变时间	事件发生时间，精确到毫秒
变量名	变量名或通道名
变量组名	变量所属的 POU 名、模块名或全局变量组或库文件名
变量说明	同变量定义
跳变类型	变量值的变化状态，从 0 变为 1 或从 1 变为 0

控制器工程修改引起的历史事件不可追溯说明：

- 变量被删除后，之前的历史事件不可追溯。
- 变量重命名后，之前的历史事件显示为当前最新变量名。

15.7.2 要求

- 变量已使能 SOE 功能
- 当前工程已下装
- Safety FA-AutoThink 离线状态
- 控制器固件版本为 LK220S-A03 及以上版本

15.7.3 步骤

要查看 SOE 信息，请按以下步骤操作。

- 第1步 单击【在线】菜单。
选择【SOE 信息读取】命令。
- 第2步 将弹出确认对话框。
- 第3步 单击确定按钮。
将弹出“另存为”对话框。
- 第4步 输入文件名并选择保存路径。
- 第5步 单击保存按钮。

15.8 设置 NTP 校时

15.8.1 概述

当控制器通过 NTP 服务器校时时，您需要使能 NTP、设置服务器 IP 地址以及校时周期。

勾选 NTP 使能后，开启校时功能。控制器通过服务器 IP 地址建立校时连接，周期性发送校时请求，使其与服务器时间保持同步。

15.8.2 要求

Safety FA-AutoThink 离线状态

15.8.3 步骤

要设置 NTP 校时，请按以下步骤操作：

第1步 单击【在线】菜单。

第2步 选择【NTP 设置】命令。

将弹出设置对话框。

第3步 勾选“NTP 使能”选项。

第4步 输入 NTP 校时服务器的 IP 地址。

第5步 请选中小时、分钟、或秒任一时间值，单击调节按钮增加或减少该值。

可设置范围 00:01:00~23:59:59。

第6步 设置完 NTP 参数后，请下载装到控制器。

此时，NTP 校时功能生效。



和利时集团

HollySys Group

地址：北京经济技术开发区地盛中路2号院

邮编：100176

电话：010-58981000

传真：010-58981100

<http://www.hollysys.com>