

## 单元描述:

编程工具PLC828用于创建PLC程序。

为了保证装有Sinumerik 828D控制系统的机床相互兼容，必须严格遵守PLC设计文档中的设计标准。提出这些要求是为了让PLC程序易于理解，这会简化故障查找过程，将因故障导致的停工时间减到最短。

本单元描述了PLC设计要求并说明了如何使用PLC828编程工具，包括上传/下载流程。

## 单元目标:

在完成本单元的学习之后，您将了解如何使用PLC828编程工具:

- 创建PLC程序
- 将PLC项目从控制系统上传至计算机
- 将PLC项目从计算机下载至控制系统

你还将了解推荐的PLC设计要求。

# Sinumerik

# 目录:

**B026**

PLC设计	→	第二节
工具箱示例	→	第三节
创建PLC程序	→	第四节
符号表	→	第五节
交叉引用	→	第六节
加载PLC程序	→	第七节

## PLC设计要求

必须落实PLC设计要求，保证装有828D控制系统的机床相互兼容。

下表列出了设计要求： -

	说明
1	除非没有别的选择，禁止使用传输变量。 <ul style="list-style-type: none"> <li>无法显示局部变量的状态。</li> <li>用户和经销商很难理解使用传输变量的程序块。</li> </ul>
2	组织段OB1只用于执行组织任务。 <ul style="list-style-type: none"> <li>有条件子程序调用</li> <li>无条件子程序调用</li> </ul>
3	程序控制逻辑不应该超出控制系统可显示区域的宽度。 <ul style="list-style-type: none"> <li>整个逻辑网络都应该可以显示在控制系统上。</li> </ul>
4	只有在没有其他选择的情况下，使用复杂指令。 <ul style="list-style-type: none"> <li>累加器/移位/旋转</li> </ul>
5	网络逻辑：长而简单的逻辑比短而复杂的逻辑好。
6	在程序中，一个输出只能被“使能/设置/重置”一次。
7	在程序中，一个过渡标记应该只有一个函数。
8	禁止使用间接编址，在程序中只使用操作数绝对名。
9	只在必要的时候执行伺服-同步任务。 <ul style="list-style-type: none"> <li>伺服任务不从OB1调用。因此，用户/经销商很难意识到它的存在。</li> </ul>
10	所有程序块和网络应带有注释。
11	符号可用于所有的外围信号，和电气图一样。
12	在中断或故障情况下，产生用户报警/信息。
13	在所有机床类型中，程序块的命名必须唯一。
14	Easy Extend应该用于机床制造商选项。
15	维护管理器应该用于机床维护时间间隔管理。

**PLC项目示例**

使用828D工具箱中的PLC828编程工具软件包创建PLC程序。

工具箱中也包括PLC项目示例。

这些示例没有满足PLC设计要求和机床测试协议。

如果需要使用这些示例，必须对其进行修改以满足PLC设计和机床测试协议文档的要求。

PLC示例描述:

D:\Tiger\828d\_toolb\_02060001\examples\PLC

- Emergency\_stop
- Sin828D\_MCP
- Sin828D\_interface\_symbols.ptp
- readme.txt

**急停示例:** 包括用于安全继电器的一个PLC项目和一个电气图。

- Safety\_relay.pdf
- Emergency\_stop\_relay.ptp

**MCP 示例:** 包括用于不同MCP类型和机床配置的PLC项目。

- Sin828D\_MCP310\_M.ptp
- Sin828D\_MCP310\_T\_b.ptp
- Sin828D\_MCP310\_T\_f.ptp
- Sin828D\_MCP483\_M.ptp
- Sin828D\_MCP483\_T\_b.ptp
- Sin828D\_MCP483\_T\_f.ptp
- readme.txt

使用MCP310的铣床  
 使用MCP310的车床，转塔刀库在主轴后面  
 使用MCP310的车床，转塔刀库在主轴前面  
 使用MCP483的铣床  
 使用MCP483的车床，转塔刀库在主轴后面  
 使用MCP483的车床，转塔刀库在主轴前面

调整行间距，使之对齐。

**接口符号:** 包括预制的符号表，表中包含用于系统接口信号和PP72/48 模块输入/输出的符号。可按照各机床要求，编辑符号表。

符号的分配用于操作数和特定功能的关联。示例：- I6.0用于通知气压正常。在该操作数中可使用符号，比如AIR\_PRESS\_OK。符号的使用有助于程序的编写和故障诊断。


- Sin828D\_interface\_symbols.ptp

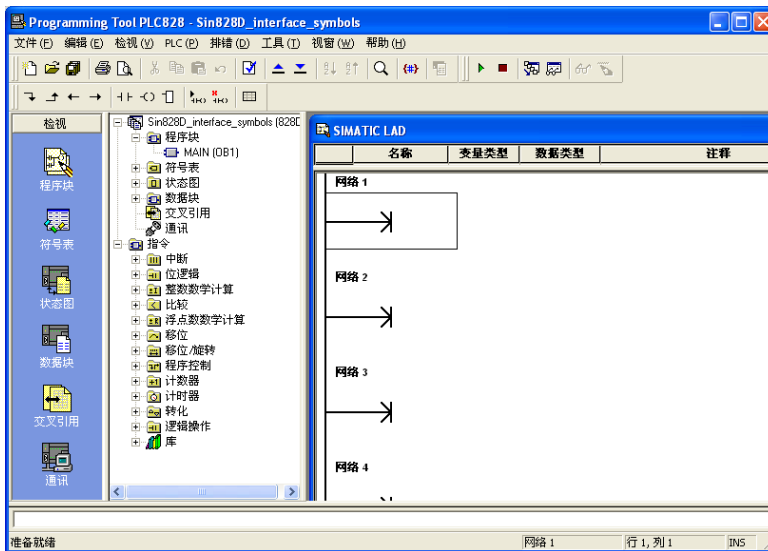
### 工具箱中的PLC项目示例

创建PLC程序时，建议使用项目示例：Sin828D\_interface\_symbols.ptp。该项目中没有任何PLC逻辑但包含预制的符号表，使用该符号表可以省时省力。

第13页对符号表进行了说明。

定位并双击打开“Sin828D\_interface\_symbols.ptp”。

 Sin828D\_interface\_symbols.ptp

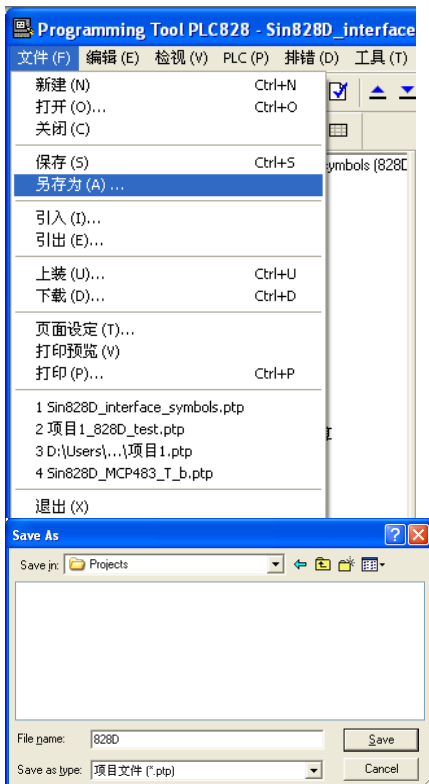


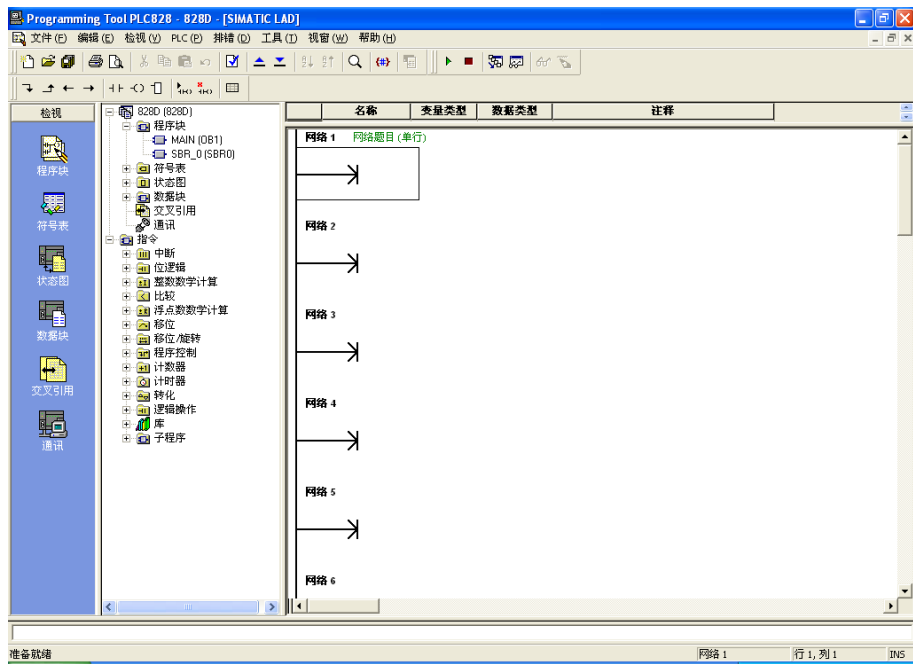
建议用合适的名称将项目另存。这样就可以保持示例不被修改，以后可以重复使用示例。

从下拉菜单中选择“另存为”保存项目。

输入项目名称并保存。

可以修改文件的目标目录。





右侧较大的窗口（SIMATIC LAD）用于插入PLC梯形逻辑。只允许使用梯形图编写程序。

空项目中默认包含两个空程序块：—

**MAIN (OB1)**是组织段（OB），可从OB中调用其他程序块。该程序块在每个PLC周期中运行一次。

**SBR\_0 (SBR0)**是空的子程序（SUBR）。所有的PLC逻辑都必须使用子程序。子程序必须被调用后才可执行其中的代码。

必须使用子程序正确构建程序。

可以使用单个的子程序，比如**排屑器**、轴使能、照明设备等。

### 子程序重命名

下例说明了如何重命名现有的默认子程序**SBR\_0**。本例中，该子程序将用于机床控制面板。

右键点击**SBR**，然后选择“重新命名”选项。输入合适的名称以体现该子例程的使用目的。



子例程被重命名为“MCP\_483”。程序块中将包含用于连接MCP483型机床控制面板的逻辑。

## 插入新的子程序

选择“编辑”菜单，然后选择“插入”命令插入新的子程序。



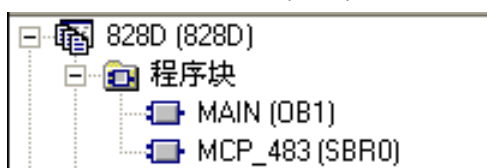
该子程序将使用下一个可用的编号。可根据需要修改子程序的编号和名称。可以添加注释和作者的相关信息。



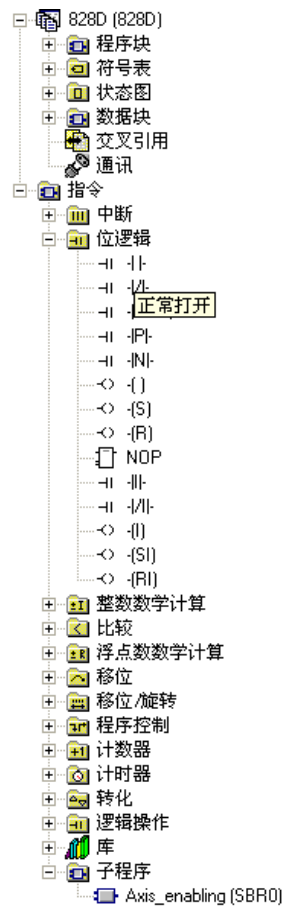
## 调用子程序

子程序必须被调用，否则其中的逻辑不被处理。

使用程序块“MAIN” (OB1) 调用子程序。打开该程序块之前，首先要打开“程序块”文件夹并且双击“MAIN (OB1)”程序块。

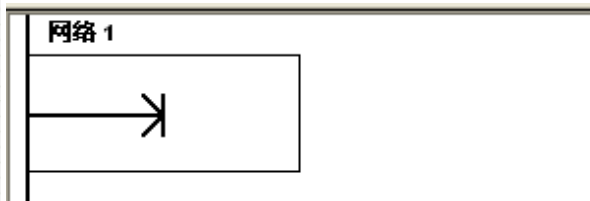


现在可以在OB1中输入对子程序的调用。  
 使用常开逻辑指令。可在“指令”文件夹中找到逻辑运算。

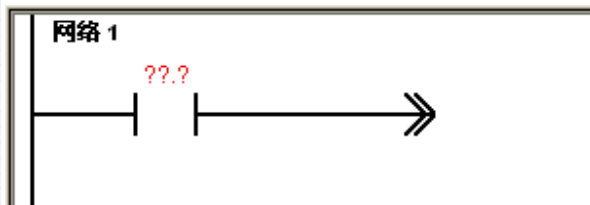


可从指令菜单中选择操作数。通过双击或拖放函数，选择所需操作数。  
 可使用各种数学函数和逻辑运算。当鼠标指针指向某个操作数/指令时，将出现简短的描述。

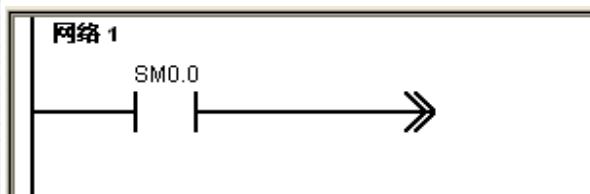
将矩形放在网络的开头并双击“正常打开”逻辑指令。



该指令将出现在网络中，输入操作数。

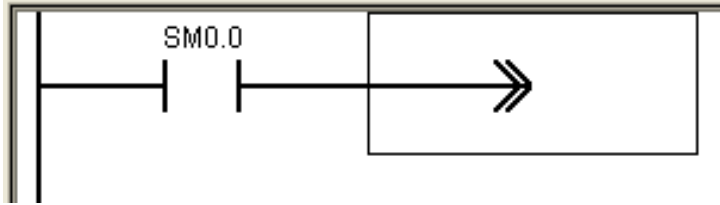


使用特殊标记SM0.0以无条件调用子程序。  
 SM0.0 常等于1。

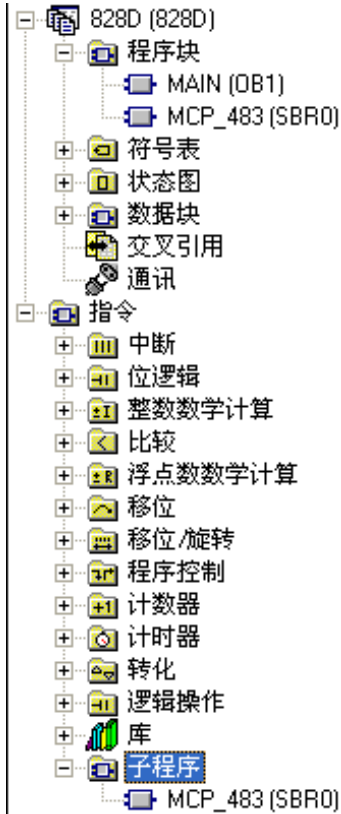




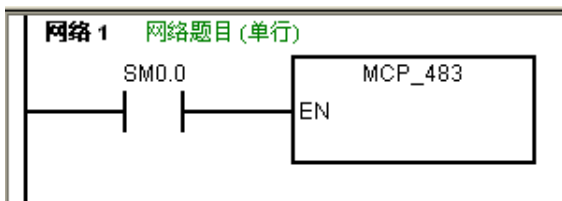
将矩形放在常开逻辑的右侧。



在矩形中输入子程序。可在“指令”文件夹中找到子程序。

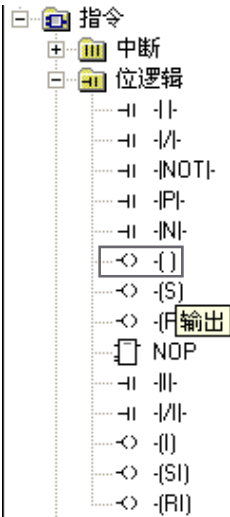


双击子程序将其输入网络。

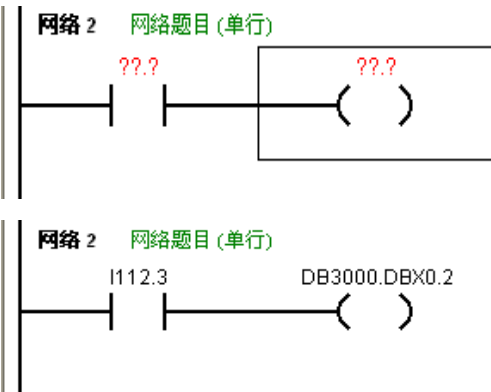


子程序将被不断的调用，其中的逻辑也将被不断处理。

一般，使用线圈/输出指令激活接口位和**输出点**等。



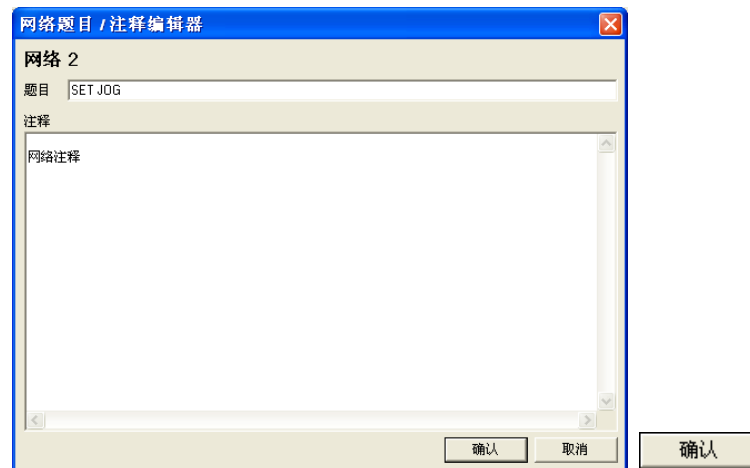
线圈的插入方法和常开触点的插入方法相同。必须为他们输入操作数。



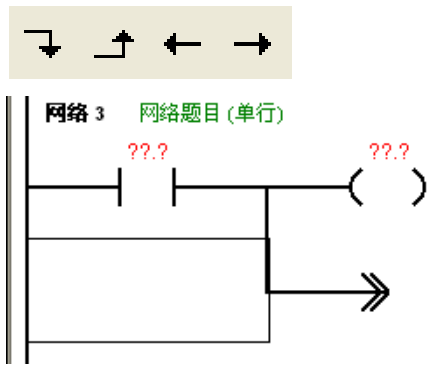
输入网络名称和注释



可以修改网络名，添加网络注释。



使用工具栏上的“线下”、“线上”、“线左”和“线右”创建网络分支。



**机床控制面板逻辑**

机床控制面板逻辑负责模式选择、轴选择、NC停止/启动等功能。

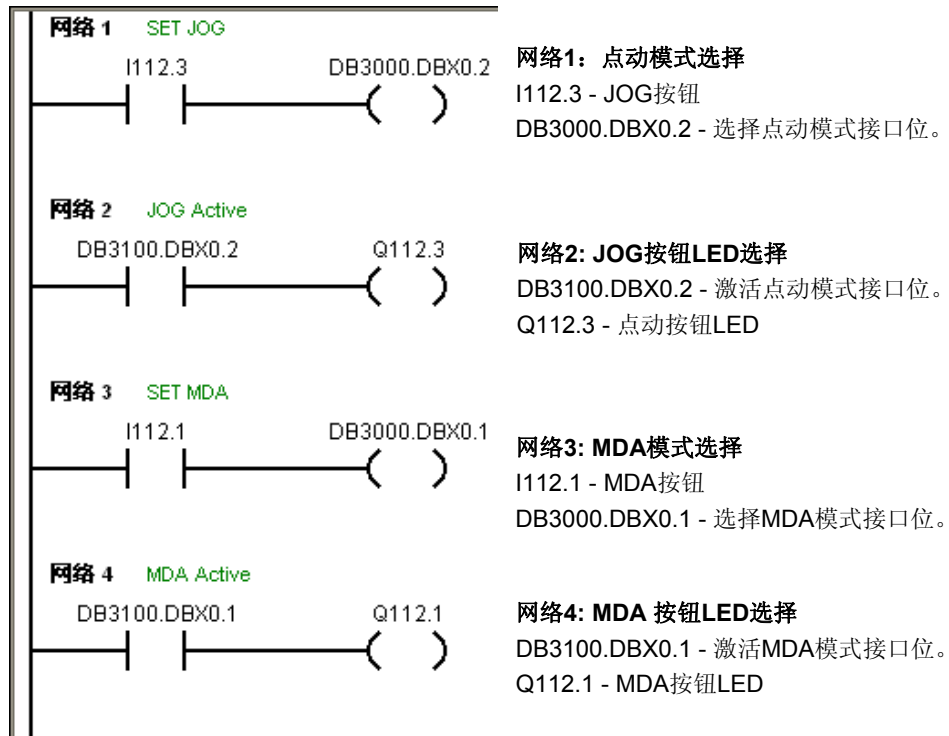
逻辑的编写必须满足第3页所述的PLC设计要求。

逻辑还需符合“机床测试协议”文档中的规范。比如：如果朝正方向点动轴的同时按下负方向点动键，轴停止运作。

可在单元B034中查看机床控制面板的接口信号。

使用开关为MCP编址，这将决定输入和输出的起始地址。为了满足兼容性要求，标准的起始地址为112，在单元B008中有相关的说明。

模式选择示例：-



## 符号表

## 符号表

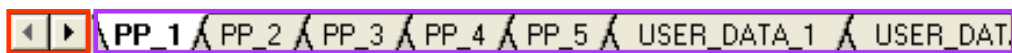
项目已经包含可编辑的符号表。  
选择左侧的“符号表”图标。



系统打开符号表并显示第一个PP72/48 模块的符号。

名称	地址	注释
		First PP module (PROFIBUS address: 9)
X111PP1		PP1/X111/T1: M (Ground of 24 VDC)
		PP1/X111/T2: 24VDC output
	I0.0	PP1/X111/T3:
	I0.1	PP1/X111/T4:
	I0.2	PP1/X111/T5:
	I0.3	PP1/X111/T6:
	I0.4	PP1/X111/T7:
	I0.5	PP1/X111/T8:
	I0.6	PP1/X111/T9:
	I0.7	PP1/X111/T10:
	I1.0	PP1/X111/T11:
	I1.1	PP1/X111/T12:
	I1.2	PP1/X111/T13:
	I1.3	PP1/X111/T14:
	I1.4	PP1/X111/T15:
	I1.5	PP1/X111/T16:
	I1.6	PP1/X111/T17:
	I1.7	PP1/X111/T18:
	I2.0	PP1/X111/T19:
	I2.1	PP1/X111/T20:
	I2.2	PP1/X111/T21:
	I2.3	PP1/X111/T22:
	I2.4	PP1/X111/T23:
	I2.5	PP1/X111/T24:
	I2.6	PP1/X111/T25:
	I2.7	PP1/X111/T26:
		PP1/X111/T27 is common ground of 24V supply (M)
		PP1/X111/T28 is common ground of 24V supply (M)

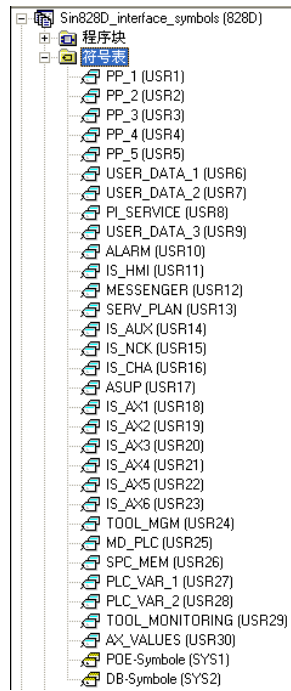
符号分别存放在不同的表中，每个表用于特定的功能，比如轴、刀具管理等。  
在符号表窗口的底部选择表格：



↑  
用于显示更多页面  
的按钮

↑  
符号表页面

也可以通过目录结构选择表格：

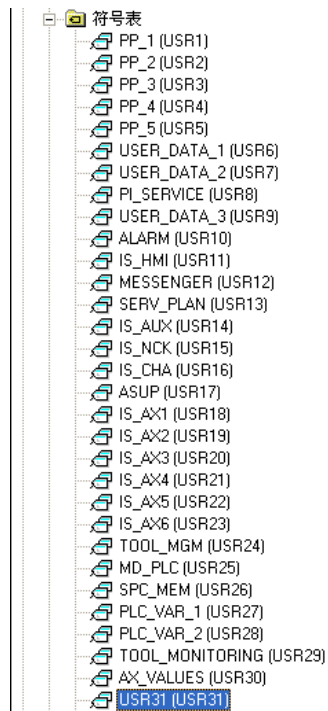


### 创建符号表

右键点击“符号表”文件夹并选择“插入符号表”创建符号表。



新的符号表将出现在现有符号表列表的底部，在本例中为“USR31 (USR31)”。



右键点击表格并选择“重新命名”重命名表格。



新插入的表格被重命名为“MCP\_483: -”



简单的编辑表格： -

符号表			
	名称	地址	注释
1	MCP_JOG_BUTTON	I112.3	MCP Jog button
2	JOG_MODE_SEL	DB3000.DBX0.2	Select Jog mode interface bit
3	JOG_MODE_ACT	DB3100.DBX0.2	Jog mode is active interface bit
4	MCP_JOG_LED	Q112.3	MCP Jog LED

在程序中选择符号信息： -

检视 (V) PLC (P) 排错 (D) 工具 (T)

STL (S)  
阶梯 (L)

---

符号表 (T)  
状态图 (C)  
数据块 (D)  
交叉引用 (R)  
通讯 (M)

---

符号寻址 (Y) Ctrl+Y  
 符合信息表 Ctrl+T

---

DB 地址显示 (B) Ctrl+B

---

顺序排列 (O)  
逆序排列 (N)

---

工具栏 ▶  
视窗 ▶

---

浏览栏 (V)  
 指令树 (I)  
 输出视窗 (U)

---

显示数据模块的数值 (D)

---

缩放图象 (Z)...

---

属性 (P)...

如果从“检视”菜单中选择了“符号寻址”或使用了“Ctrl+Y”快捷键，操作数将以文本格式显示。  
如果选择了“符合信息表”或使用了“Ctrl+T”快捷键，将在包含符号表信息的网络的底部显示表格。

含有所选择的符号地址和符号信息表的网络： -

没有符号寻址换图!!!

**网络 1 SET JOG**

JOG_MODE_SEL	V30000000.2	Select Jog mode interface bit
MCP_JOG_BUTTON	I112.3	MCP Jog button

**网络 2 JOG Active**

JOG_MODE_ACT	V31000000.2	Jog mode is active interface bit
MCP_JOG_LED	Q112.3	MCP Jog LED

### 交叉引用

可以创建交叉引用表，用于列出单个操作数的信息，比如在故障定位过程中使用了哪些程序块以及哪些程序块可以很好的协助维护人员。



**A compile must be performed to display cross reference.**

选择工具栏上的编译按钮完成编译。



编译过程的结果显示在窗口底部。

```
正在编译数据模块...  
DB_9000 (DB9000)  
已编译数据模块的数量 = 1, 0 错误  
  
正在编译程序块...  
MAIN (OB1)  
MCP_483 (SBR0)  
  
联接程序块...  
MAIN (OB1)  
MCP_483 (SBR0)  
  
程序段内包含11指令, 使用了6281符号  
编译的程序段尺寸 = 9% (使用内存器的百分比), 0 出错  
  
总错误数 0
```



如果程序没有错误，系统将显示交叉引用表。  
双击一个元素将打开使用该元素的程序块。

	元素	块	位置	上下文
1	I112.1	MCP_483 (SBR0)	网络 3	- -
2	MCP_JOG_BUT	MCP_483 (SBR0)	网络 1	- -
3	Q112.1	MCP_483 (SBR0)	网络 4	-()
4	MCP_JOG_LED	MCP_483 (SBR0)	网络 2	-()
5	N_C_MDAMOD	MCP_483 (SBR0)	网络 3	-()
6	JOG_MODE_SEL	MCP_483 (SBR0)	网络 1	-()
7	P_C_MDAMOD	MCP_483 (SBR0)	网络 4	- -
8	JOG_MODE_ACT	MCP_483 (SBR0)	网络 2	- -

之前描述的“符号寻址”选项仍然可以用于交叉引用表。

	元素	块	位置	上下文
1	I112.1	MCP_483 (SBR0)	网络 3	- -
2	MCP_JOG_BUT	MCP_483 (SBR0)	网络 1	- -
3	Q112.1	MCP_483 (SBR0)	网络 4	-()
4	MCP_JOG_LED	MCP_483 (SBR0)	网络 2	-()
5	N_C_MDAMOD	MCP_483 (SBR0)	网络 3	-()
6	JOG_MODE_SEL	MCP_483 (SBR0)	网络 1	-()
7	P_C_MDAMOD	MCP_483 (SBR0)	网络 4	- -
8	JOG_MODE_ACT	MCP_483 (SBR0)	网络 2	- -

### 上传至外界计算机

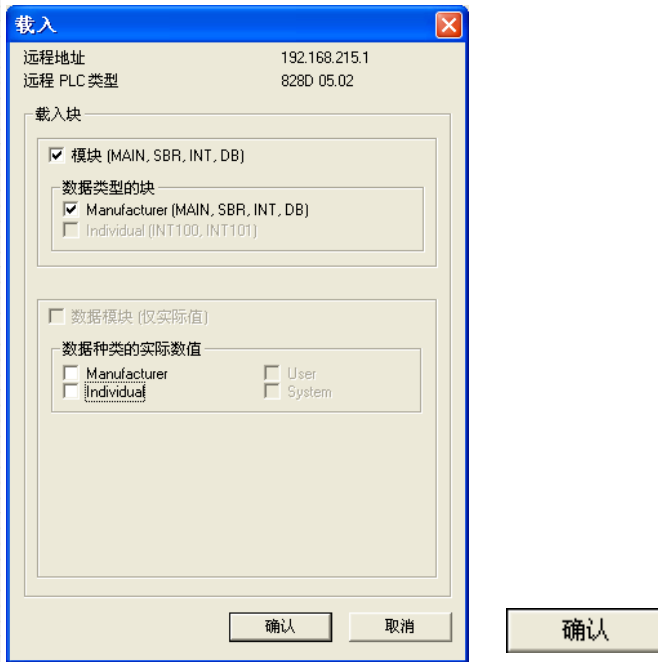
在继续操作之前，必须与控制系统建立在线连接。

可能需要将现有的程序从控制系统保存至计算机。这种情况下，在 PLC828D 编程工具中打开一个新的项目。

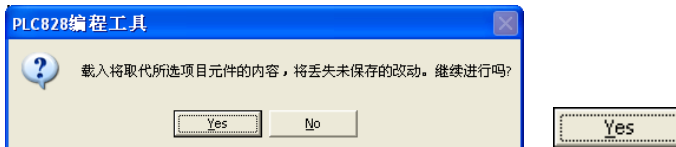
点击工具栏上的“上传”按钮。



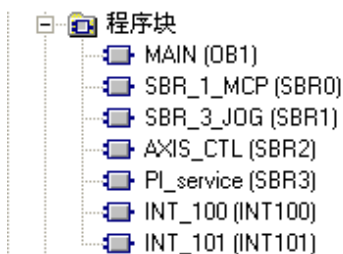
选择待上传的数据级程序块。完全备份时应选择所有的程序块。



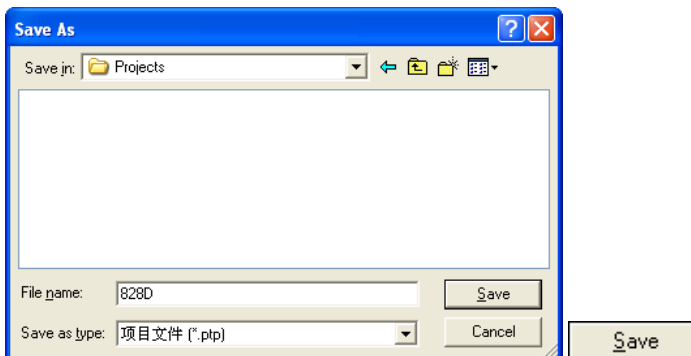
系统将显示警告-该步骤将覆盖所有现有的项目组件。



从控制系统上传的程序块将显示在“程序块”文件夹中。



使用合适的名称保存上传的文件：

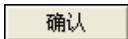
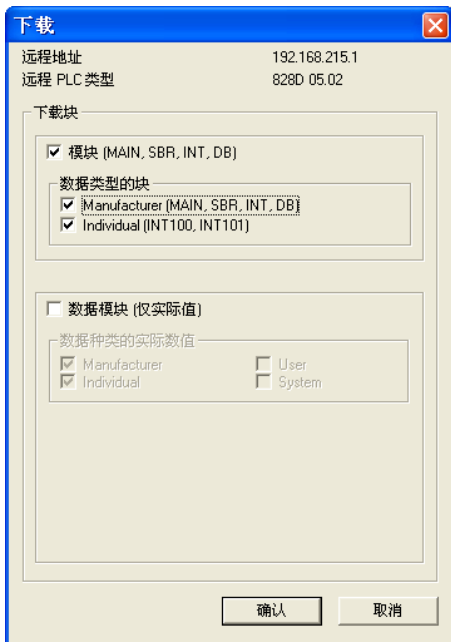


### 下载程序

在工具条中选择“下载”按钮。

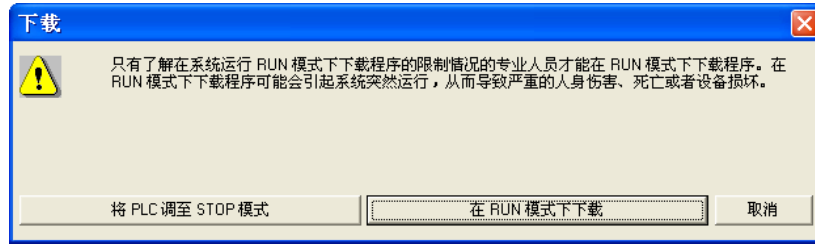
确保“模块 (MAIN, SBR, INT, DB)”选项被选中。可以选择数据级程序块。完整下载时两项都应勾选。

选择“确认”键继续。



对话框警告，在运行模式中下载时可能存在危险。  
在某些情况下，比如下载数据块时，系统只允许在停止模式中下载。

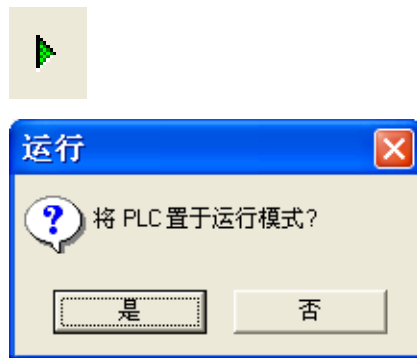
选择“将 PLC调至STOP模式”选项。



程序被下载至控制系统。下载完成后，系统弹出对话框。按“OK”按钮确认。



PLC仍然处于停止模式。选择工具栏上的“运行”按钮，重启PLC。在弹出的对话框中选择“是”按钮。



控制系统将执行热重启，热重启结束后，PLC不应该有错误。  
现在可以测试PLC程序的功能了。