



三菱可编程控制器
培训教材

FX系列可编程控制器

(入门篇)

FX
S
M
E
L
E
M

● 安全注意事项 ●

(实习前请务必阅读本项注意事项)

在设计系统时，请务必阅读手册，并充分注意安全。
此外，在实习时，请充分注意下面几点，采用正确的操作。

【 实习时的安全注意事项 】

危险

- 为了不发生事故，请不要在通电中接触端子。
- 在打开安全罩时请切断电源，或在确认足够安全后进行操作。
- 请不要将手伸向可动部。

注意

- 请按照讲师的指示进行实习。
- 请不要擅自拆下实习机的模块或改变配线。
否则会导致故障、误操作、受伤、火灾。
- 在安装/卸下模块时，请在切断电源后进行。
若在通电中进行装卸，则会导致模块的故障或触电。
- 在实习机（X/Y平台等）产生异臭/异常噪声时，请立即切断电源开关。
- 若出现异常情况，则请立即联系讲师。

序 言

为了首次接触可编程控制器的人，本教材基于简单的例题对需要知道的顺序控制的基础知识进行介绍。

此外，本教材基于使用微型可编程控制器FX_{1S}-14MR型的教材FX-I/O演示机进行阐述。

有下列相关资料可供参考

- | | |
|---|-------------------|
| (1) FX _{1S} 袖珍手册 (使用说明书) | …………… JY992D83801 |
| (2) FX编程手册
(FX _{1S} 、FX _{1N} 、FX _{2N} 、FX _{1NC} 、FX _{2NC}) | …………… JY992D62001 |
| (3) FX-10P型操作手册 (操作要领书) | …………… JY992D33301 |

关于商标的说明

- Microsoft和Windows是美国微软公司 (Microsoft Corporation) 在美国和其它国家的注册商标或商标。
- 本书中记载的其它公司名称和商品名称是各相应公司的商标或注册商标。

本书的内容不构成对行使工业所有权等权利的保证，也不构成对行使权的许诺。另外，对于因本书中记载的内容的使用而引起的有关工业所有权的各种问题，本公司不负一切责任。

目 录

第1章 什么是顺序控制	1-1
1.1 什么是顺序控制	1-2
1.2 围绕顺序控制	1-4
1.3 顺序控制所需的	1-6
1.4 基于顺序进行配线实习	1-9
1.5 记住顺序符号	1-14
第2章 什么是可编程控制器	2-1
2.1 什么是可编程控制器	2-2
2.2 可编程控制器的构造	2-3
2.3 配线和程序	2-7
2.4 能使用可编程控制器的好处	2-8
第3章 关于顺序指令	3-1
3.1 记住指令	3-2
LD,LDI,OUT,END	
SET,RST	
AND,ANI	
OR,ORI	
ANB	
ORB	
NOP	
3.2 程序的顺序	3-12
3.3 关于定时器回路	3-13
3.4 关于计数器回路	3-14
3.5 关于自保持回路	3-15
第4章 程序练习	4-1
4.1 引入事例1 (电梯控制)	4-2
4.2 引入事例2 (供茶器控制)	4-4
4.3 引入事例3 (钻床控制)	4-6
4.4 引入事例4 (抢答显示控制)	4-8

附录1 GX Developer的操作附录.....1-1

附录1.1 操作GX Developer所需的基础知识附录.....1-2
附录1.2 启动GX Developer和新建项目附录.....1-6
附录1.3 生成回路附录.....1-9
附录1.4 向可编程控制器写入程序附录.....1-15
附录1.5 编辑回路附录.....1-21
附录1.6 保存生成的回路附录.....1-31
附录1.7 调试程序所需的操作附录.....1-34
附录1.8 注释输入附录.....1-42
附录1.9 列表程序的生成操作附录.....1-46

附录2 需要知道的SFC方式附录.....2-1

附录2.1 需要知道的SFC方式附录.....2-2

附录3 实习机的输入输出连接图附录.....3-1

附录3.1 实习机的输入输出连接图附录.....3-2

第1章 什么是顺序控制

了解顺序控制

下面向大家介绍的“可编程控制器”是用于进行“顺序控制”的装置。

那么，什么是“顺序控制”呢？

虽然平时听不到这个词，但实际上在我们周围有很多实例，谁都曾接触过。

例如全自动洗衣机，这也是出色的“顺序控制”。

在本章中，以我们身边存在的“顺序控制”为例，让我们思考什么是“顺序控制”。

1.1 什么是顺序控制

顺序是什么意思

“顺序控制”……虽然是个很听不惯的词语，但是在我们周围使用得极多，我们肯定曾经见过或碰过受其控制的东西。

首先，若在词典中查找顺序（Sequence）这个词语，则有

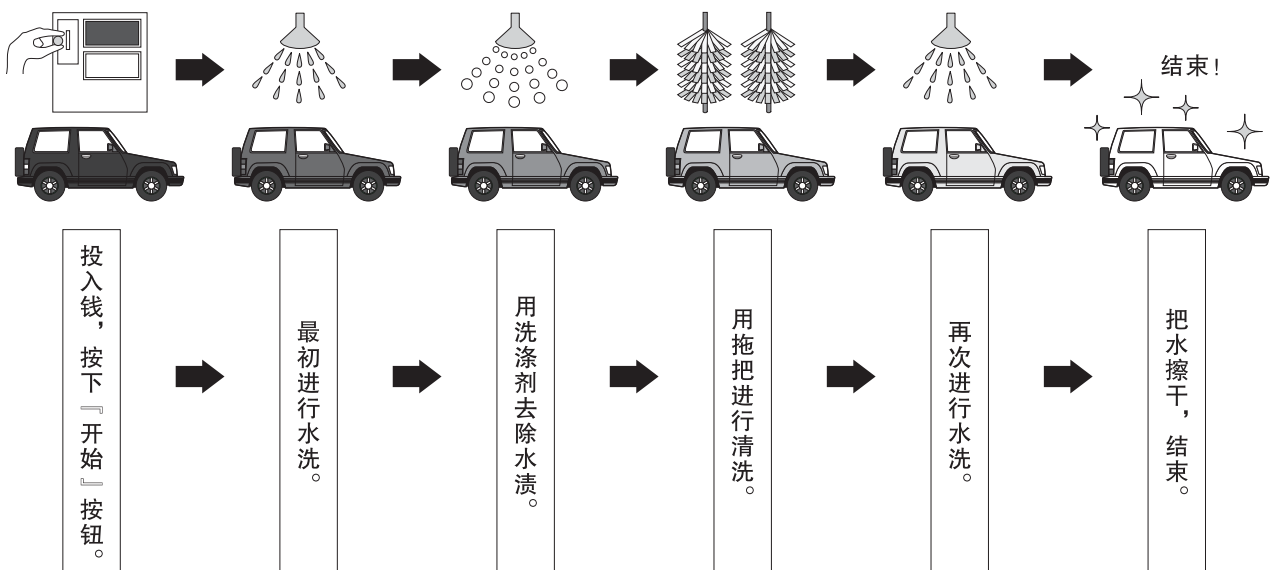
- ① 连续发生，连续、连发
- ② 系列事物，连续事物
- ③ 顺序，次序，依次
- ④ 后续发生的事件、余波、结果…等意思。

由此可知，顺序是指连续发生或现象发生的顺序。

“顺序控制”也是从“顺序”这个词语来的，是指按照预先确定的顺序进行动作。此外，使其按照希望的那样进行动作即为控制。

举例周围的事物……

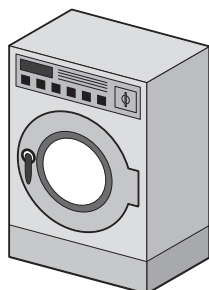
以在加油站看到的洗车机为例。



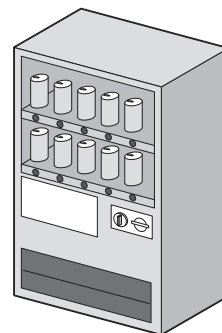
虽然上一页的洗车机的动作很粗略，但该作业顺序正是顺序的考虑方法，使该顺序按希望的那样无论几次都正确、自动地动作，这就是顺序控制。

此外，顺序控制在各方面、各领域广泛使用，其考虑方法不可或缺。

■ 业务用电器

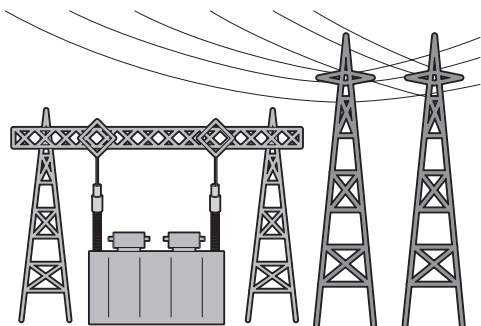


■ 自动售货机

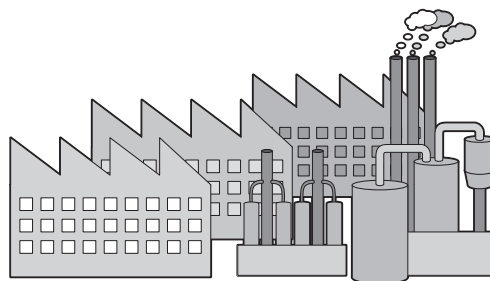


在各种领域中使用

■ 变电所



■ 工厂



就像这样，顺序控制并不难，它很贴近我们的生活。

参 考

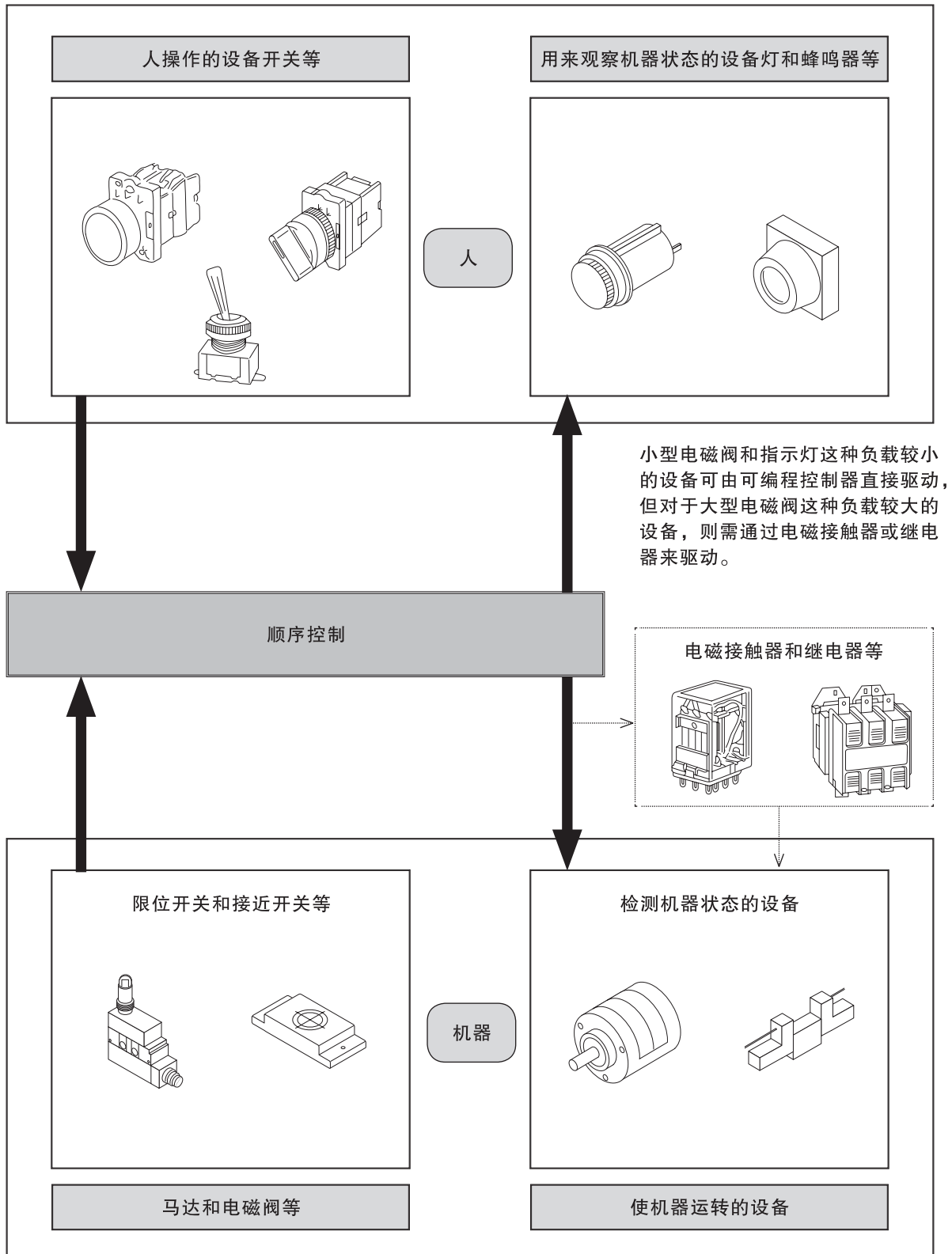
日本工业规格的定义

在日本工业规格（JIS）中，顺序控制如下定义。
“按照预先确定的顺序依次进行控制的各阶段的控制”

1.2 围绕顺序控制

构成顺序控制的设备

要进行顺序控制，则使用如下的设备。这些设备大致分为“人操作的设备”、“让人知道机器状态的设备”、“检测机器状态的设备”、“使机器运转的设备”。



图中只是一例，除此之外还有很多设备。

顺序控制在组合这种设备后使它们按照作业顺序进行动作。

此外，在这些设备中，“人操作的设备”、“检测机器状态的设备”在顺序控制中构成驱动用的条件。另外，“让人知道机器状态的设备”、“使机器运转的设备”构成根据该条件进行动作的设备。

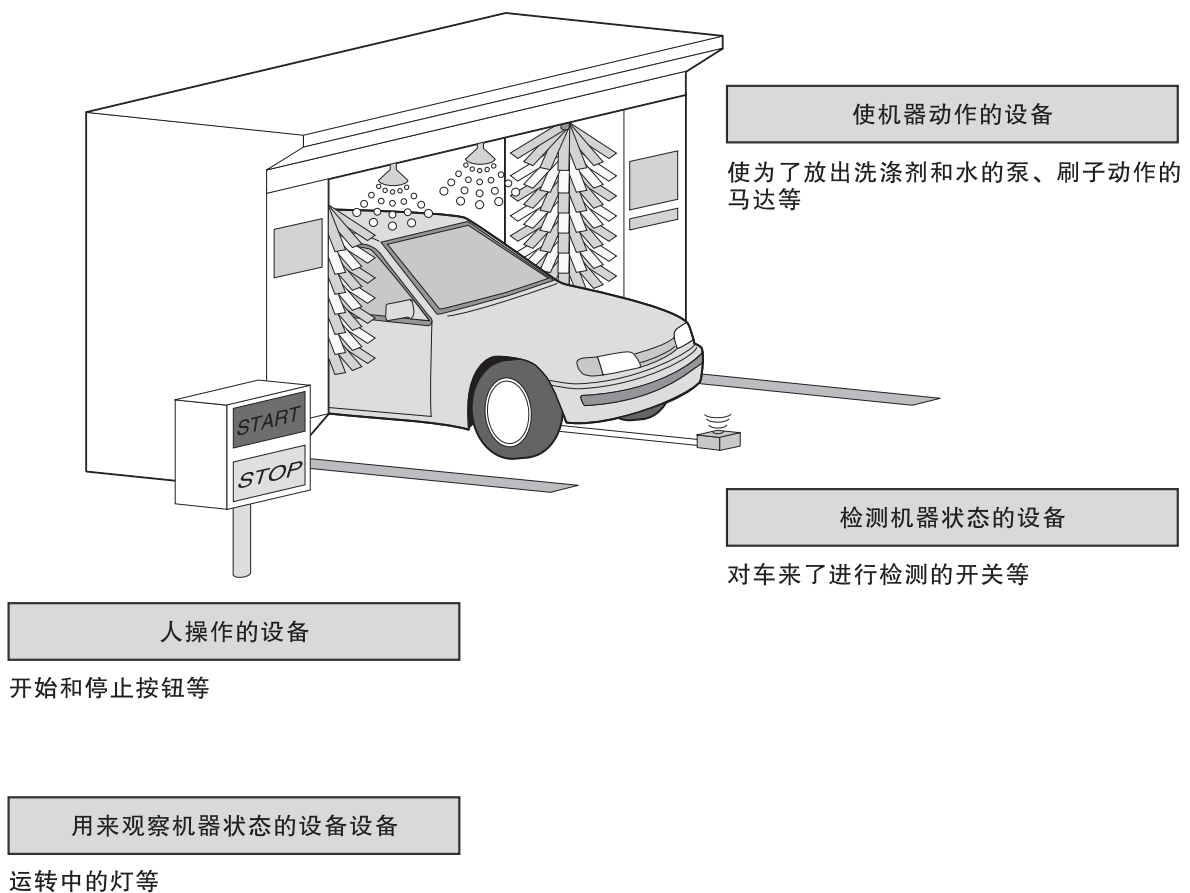
操作箱

安装有人操作的设备（按钮开关和选择开关等）、让人知道机器状态的设备（灯和数字显示器等）的箱

控制箱

安装有电磁接触器、继电器、可编程控制器等用于控制机器动作的设备的箱。

例如，洗车机也在组合各种设备后受到顺序控制。



1.3 顺序控制所需的

让我们来具体考虑顺序控制。

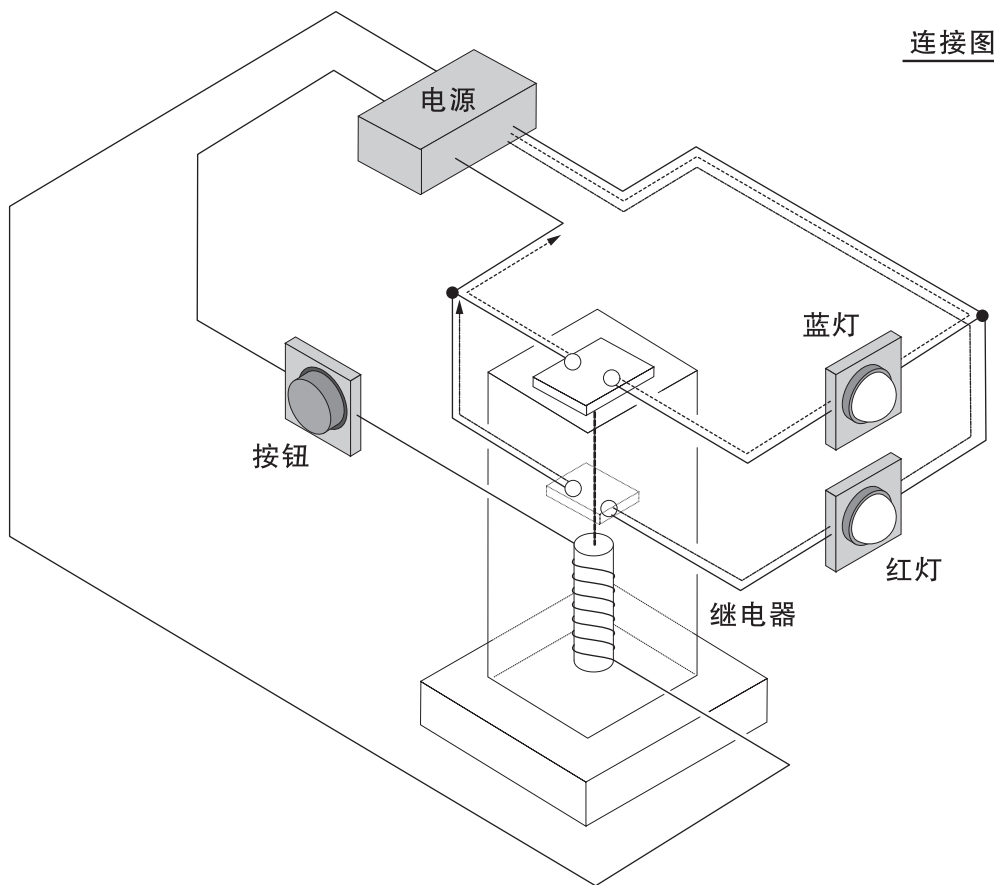
让我们根据以下连接图来具体考虑顺序控制。

此外，在此也对学习顺序控制需要知道的用语等进行说明。

例1：使用按钮、灯（蓝、红）、继电器来连接以下电气回路。

◎ 顺序控制内容

- ① 在未按下按钮的状态下，电流按线B的路径流动，蓝灯亮。
 - ② 若按下按钮，则电流按线A的路径流动，红灯亮。
 - ③ 若松开按钮，则又像①一样，蓝灯亮。
- ① ~ ③的动作构成顺序控制的一部分。



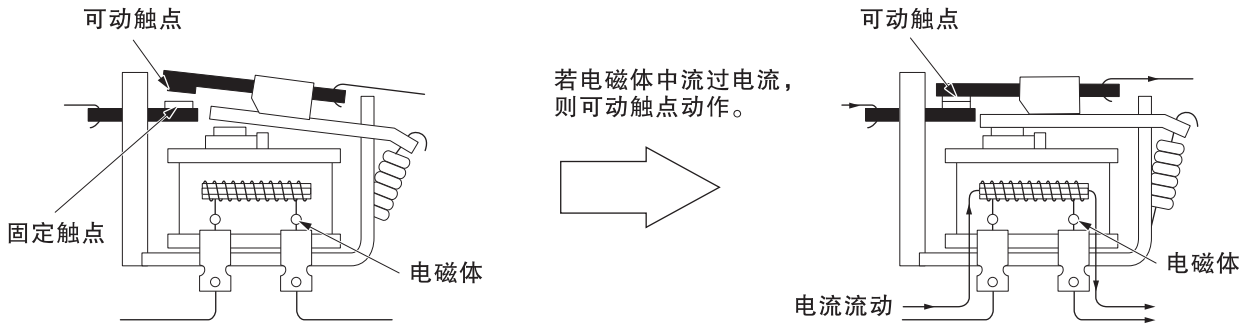
让我们学习新的用语。

对左页的继电器进行说明。

● 关于继电器

所谓继电器，犹如其名，具有中继的意思。

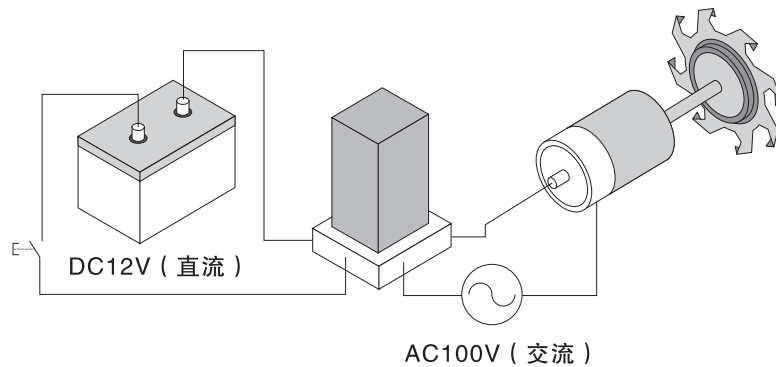
实际中，继电器内具有电磁体，通过该电磁体的作用吸引可动铁片，从而关闭或张开触点*1。



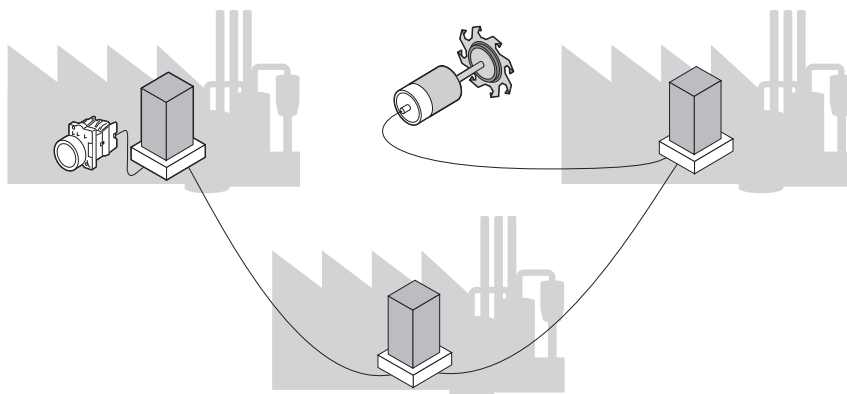
※1 触点所谓触点，是指进行开闭动作的接触部分，进行通电、断电。除继电器之外，开关、定时器、计数器等也有触点。此外，触点包括a触点、b触点等。（参照下一页）

☆ 为何需要继电器。

① 继电器可以用较小的信号来驱动较大的马达和放大器等。



② 可驱动远距离的马达和放大器等。



● 关于触点

触点通过开闭动作来切断、导通电流。

触点的基础是a触点、b触点。触点具有开关、继电器、定时器、计数器等。

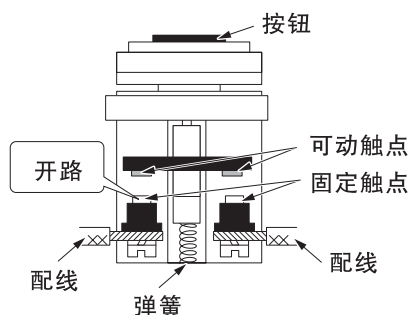
a触点

所谓a触点，是指常开触点，若收到指令*1，则触点关闭。

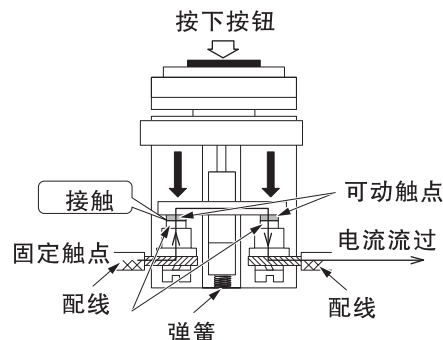
※1 所谓指令，意思是使动作、使变化，若是按钮，则按下按钮这一动作就是指令。

动作 按按钮开关时

在未按下按钮开关时，触点张开。若按下按钮，则触点关闭。



在按下按钮前的状态（复位）

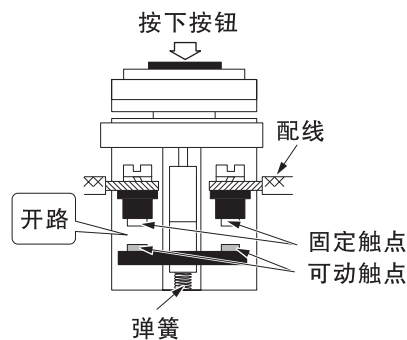
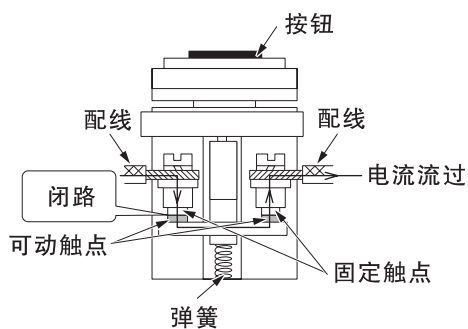


按下了按钮的状态（动作）

b触点

所谓b触点，是指常闭触点，若收到指令，则触点打开。

动作 按按钮开关时



参 考

a触点、b触点的名称由来

a触点、b触点的名称是从哪里来的呢？

a触点: arbeit contact ...工作触点
 b触点: break contact ...断开触点

取了它们的首字母

所以，a触点也叫常开触点，b触点也叫常闭触点。

1.4 基于顺序进行配线实习

在学习了新的用语的基础上

在学习了继电器触点的基础上，通过1-6页的例1进行实习。

◎继电器配线实习演示机概要

产品构成

电源电缆	1根
按钮	1个
蓝灯	1个
红灯	1个
继电器	1个
铅丝	(茶、红、橙、黄、绿、蓝、紫、灰)

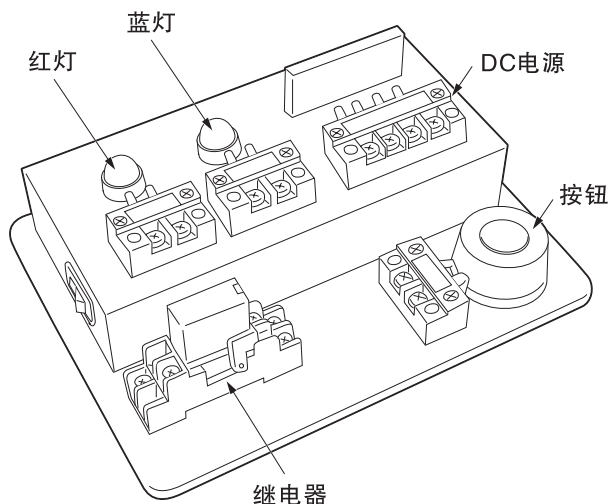


图1. 蓝灯

●配线吧。

1. 确认电源切断了。

2. 图1.进行蓝灯的配线。

7号端子间用紫线连接

4号端子间用黄线连接

8号端子间用灰线连接

开始接线

3. 图.进行红灯的配线。

5号端子间用绿线连接

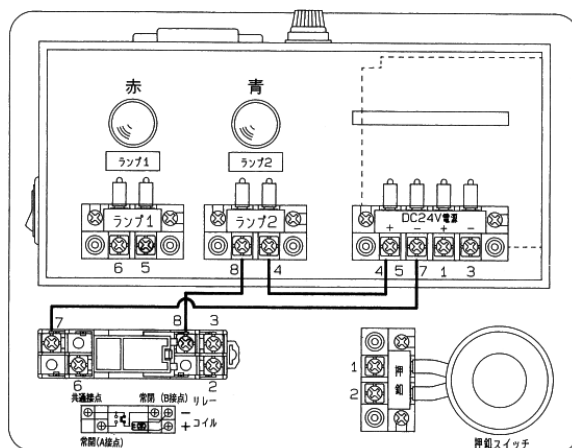
6号端子间用蓝线连接

1号端子间用茶线连接

2号端子间用红线连接

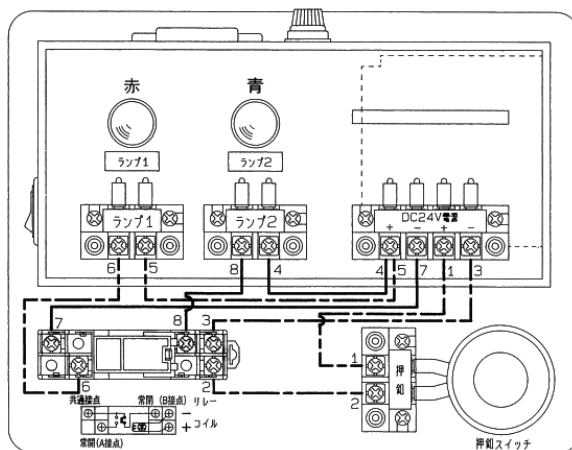
3号端子间用橙线连接

开始接线



实体配线图※2

图2. 红灯



注) 在本事例中，电源的4号、5号端子是共用端子。

※2 实体配线图…

是指以尽可能接近实物的形式来显示回路的连接和回路所使用的设备等的图。由于该图能正确地看到配线和设备的结构，因此在实际中制作装置或维护时很方便。

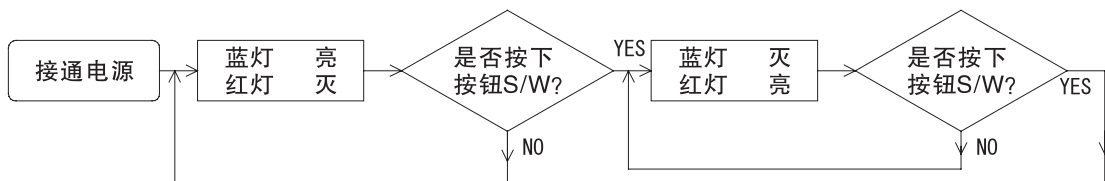
在学习了新的用语后

● 确认动作。

试着用从1-6页的顺序控制内容中学到的语句进行书写。

- ① 若接通电源，则以b触点、B路径通电，蓝灯亮。
- ② 之后，若按下按钮（按钮是a触点），则由于继电器的作用，a触点关闭，以A路径通电，红灯亮。
- ③ 若松开按钮，则蓝灯再次点亮。

基于流程图的顺序表示



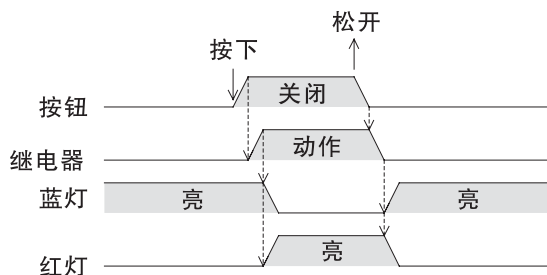
所谓流程图

顺序控制组合各种设备来构成回路。

因此，若详细地写下所组合的设备的动作顺序，则有时反而难以理解整体。

流程图是按顺序用四边形的图形和箭头简单地表示整体的相关动作的图。

基于时间图的顺序表示



所谓时间图

对应时间的变化使动作顺序易于理解的图。

在纵轴上写下要控制的设备，在横轴上表示时间的变化。

分别与哪个设备构成相关动作用虚线的箭头表示。

有时也用简化了箭头等的图进行表示。



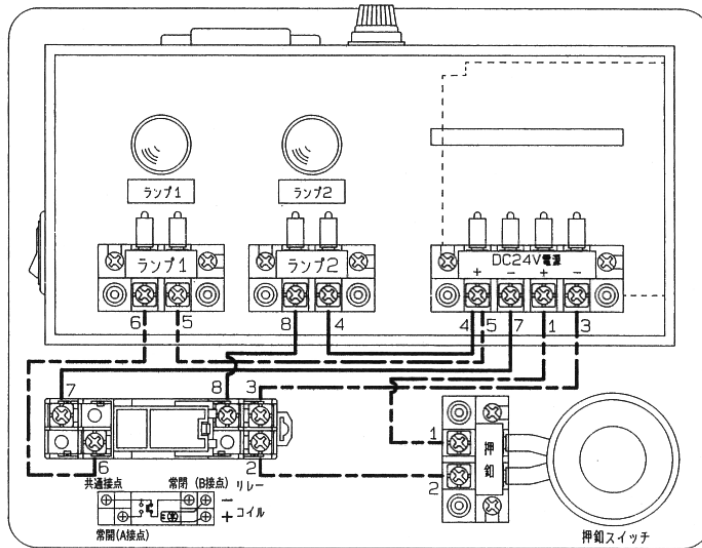
也可从流程图或时间图获悉顺序控制内容。

实体配线图和顺序图

实体配线图是容易理解设备的结构、配线等的作图方法，但对复杂的电气回路来说，动作顺序变得不易理解。

为了使其容易理解，试着将其转换成顺序图。

实体配线图



注) 在FX-I/O DEMO中，继电器和灯均由DC24V（直流）来供电，但一般来说，像下面的顺序图那样，继电器由DC24V来供电，而灯则由AC100V（交流）来供电。

顺序图



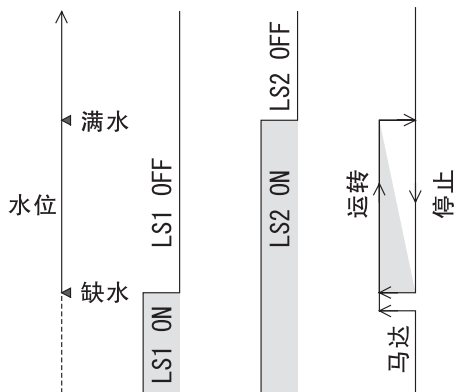
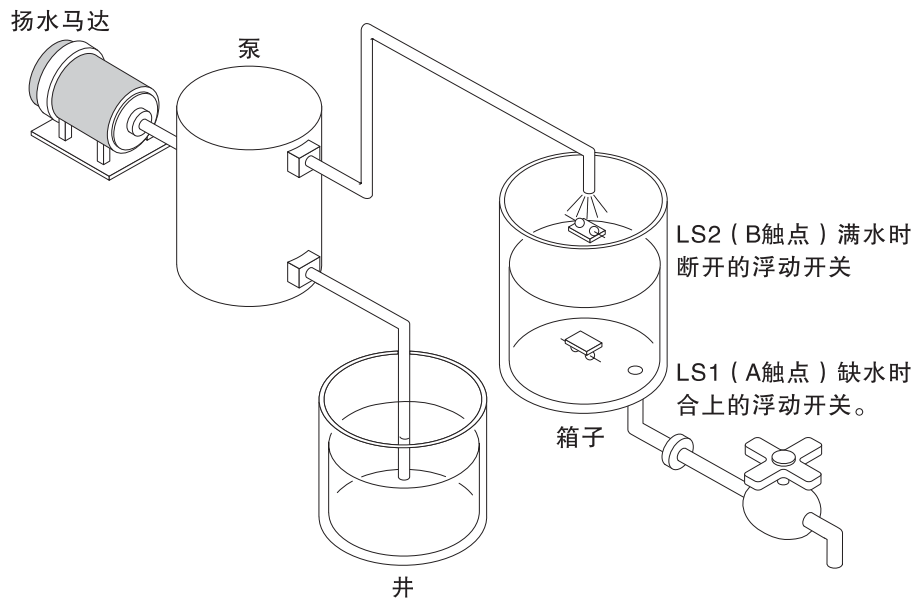
顺序图是将许多回路按照动作顺序、使动作内容变得容易理解的连接图。此外，由于该作图方法规格统一，因此第三者看起来也容易理解。

试着用其它例子来考虑。

例2：控制箱子的水位。

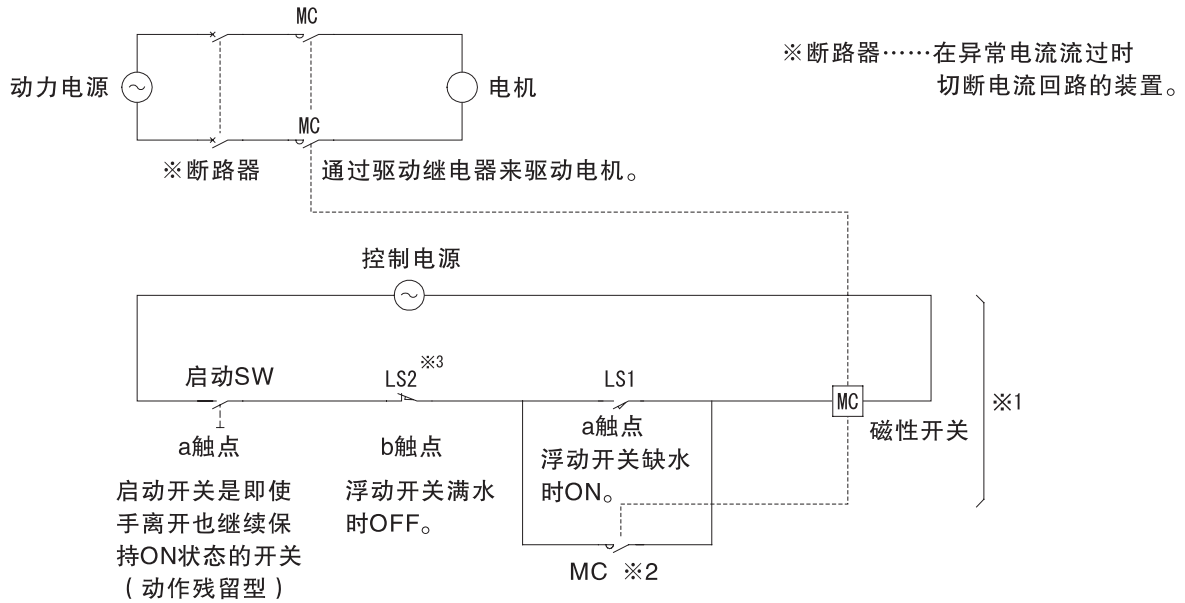
◎ 顺序控制内容

- ① 若关闭运转开关S/W，则缺水时浮动开关LS1合上，使驱动缺水用马达的磁性开关MC运转。即使达到中间水位，继电器MC也保持。
- ② 若变为满水，则浮动开关LS2打开，继电器MC的保持动作被解除，缺水用马达停止。
- ③ 若水位达到缺水平面，则马达再次自动地旋转。



对于该控制下的马达的动作，在缺水后的中间水位马达运转。在满水后的中间水位马达停止，虽然也是中间水位，但马达的运转状态不同。这种动作也叫滞后动作（延迟动作），可减小马达运转的停止频率。

● 顺序图



※1 一般将该部分的顺序转换为可编程控制器。

※2 关于自保持回路

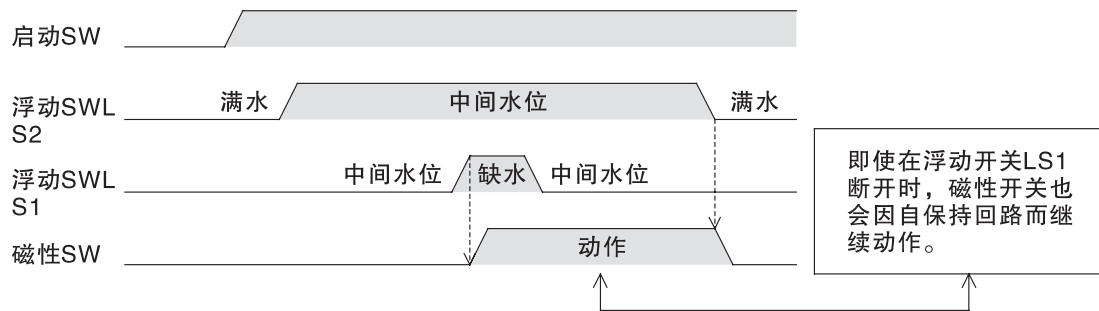
所谓自保持回路，是指利用继电器自身的触点将从外部付予继电器等的信号形成分路（旁路），从而生成动作回路。

此外，自保持回路也可以说是具有记忆功能就像即使是如果按下了按钮开关后再松开则会断开的回路也能连续地动作的回路。

在本例中，将驱动马达的继电器的触点与浮动开关LS1并联连接。该回路称作自保持回路。

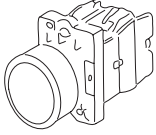




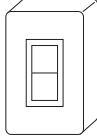



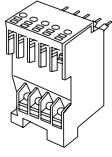



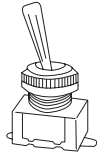


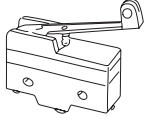








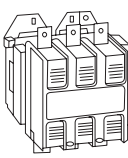




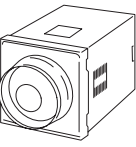




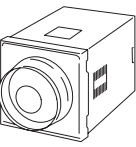




※3 LS2（浮动开关）具有切断※2自保持回路、停止供水用马达的作用，但若在B触点使用该LS2（浮动开关），则在开关接触故障或配线断裂时，开关断开，从而也可停止供水。

自保持回路的动作（时间图）



1.5 记住顺序符号

主要的顺序符号一览

按产品来分		按触点来分		a触点 (常开触点)		b触点 (常闭触点)		驱动源	
		横写	竖写	横写	竖写				
按钮开关 (自动复位型)						手动			
按钮开关 (动作残留型)						手动			
热继电器 (OCR)						加热器 动作			
开关(普通)						手动			
限位开关 (设备操作)						夹头 凸轮			
辅助开关 继电器触点						电磁线圈			
电磁接触器 (连接器)									
定时器 接通延时									
定时器 断开延时									

第2章 什么是可编程控制器

可编程控制器是……

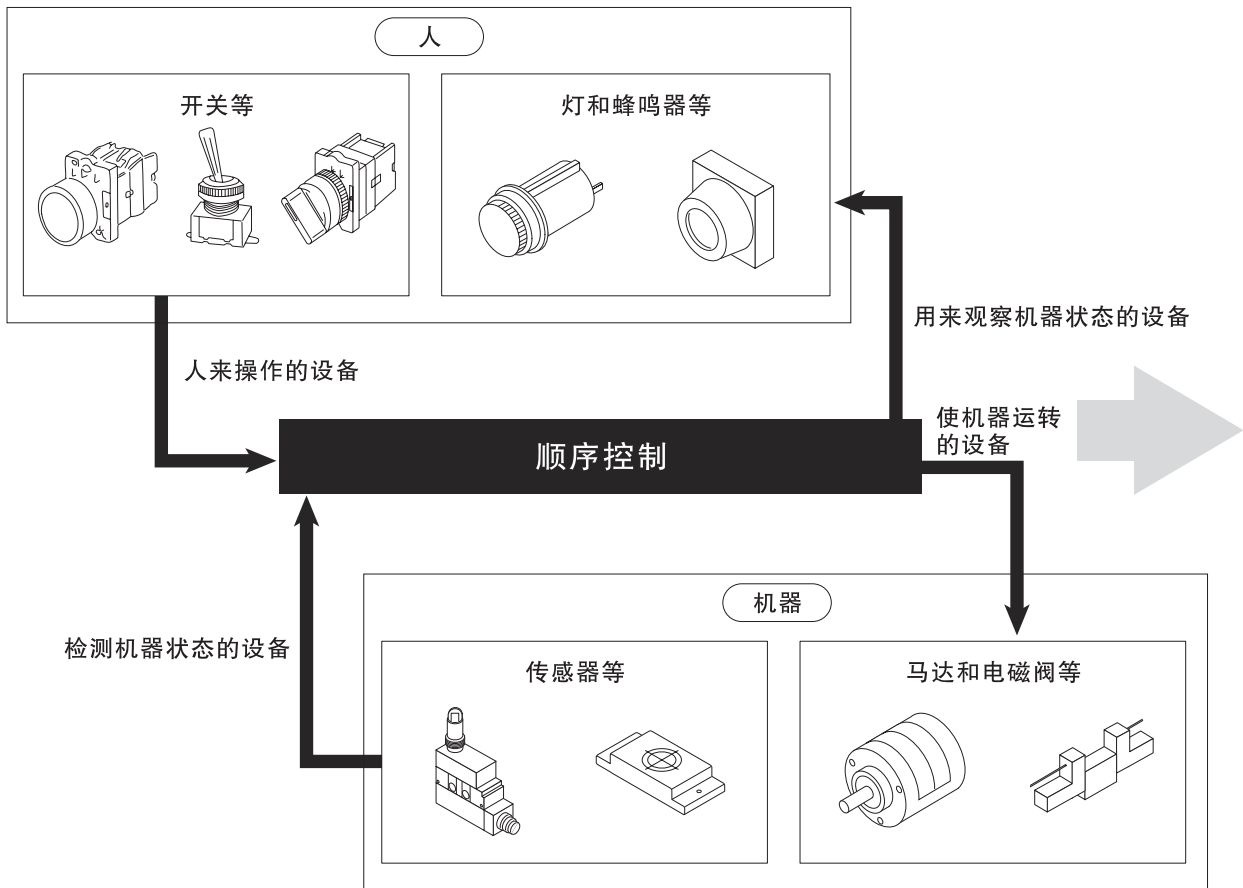
可编程控制器又被称作Programmable Controller (PC) , 也称作Sequence Controller (SC) 。它被定义为“通过输入输出部来控制各种装置、内置有用于存储编程指令的存储器的电子装置”。

在实际中……

请视作利用简单的程序来实现之前通过对继电器和定时器配线进行“顺序控制”的装置。

2.1 什么是可编程控制器

可编程控制器用来做什么



在顺序控制中，将“人操作的设备”、“检测机器状态的设备”称作指令信号或条件信号。此外，将“用来观察机器状态的设备”、“使机器运转的设备”称作负载。

可编程控制器相当于上图的“顺序控制”部分，起着控制这些设备的作用。可编程控制器正是用于进行顺序控制的装置。

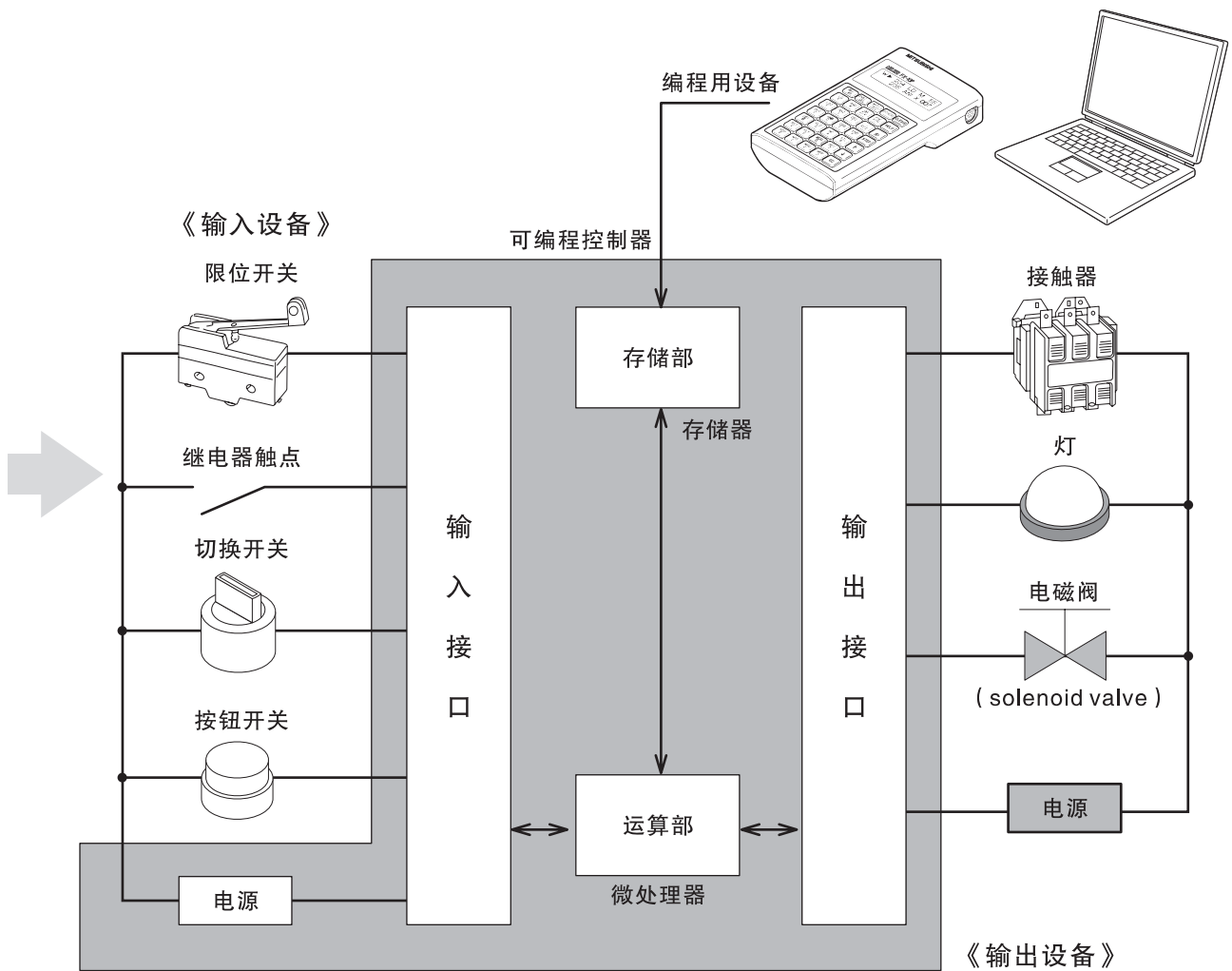
备考

可编程控制器是三菱电机创造的词语？

一般来说，“可编程控制器”这一称呼已被广泛使用。现在，日本电机工业会（JEMA）的正式名称是Programmable Controller（PC），但好像还是可编程控制器叫起来方便，也容易理解。本来，“可编程控制器”在PC诞生前好像就被部分使用过，但自从三菱电机用可编程控制器这一名称销售PC以来，是K系列和F系列使其变得普遍化的。

2.2 可编程控制器的构造

顺序控制怎么进行

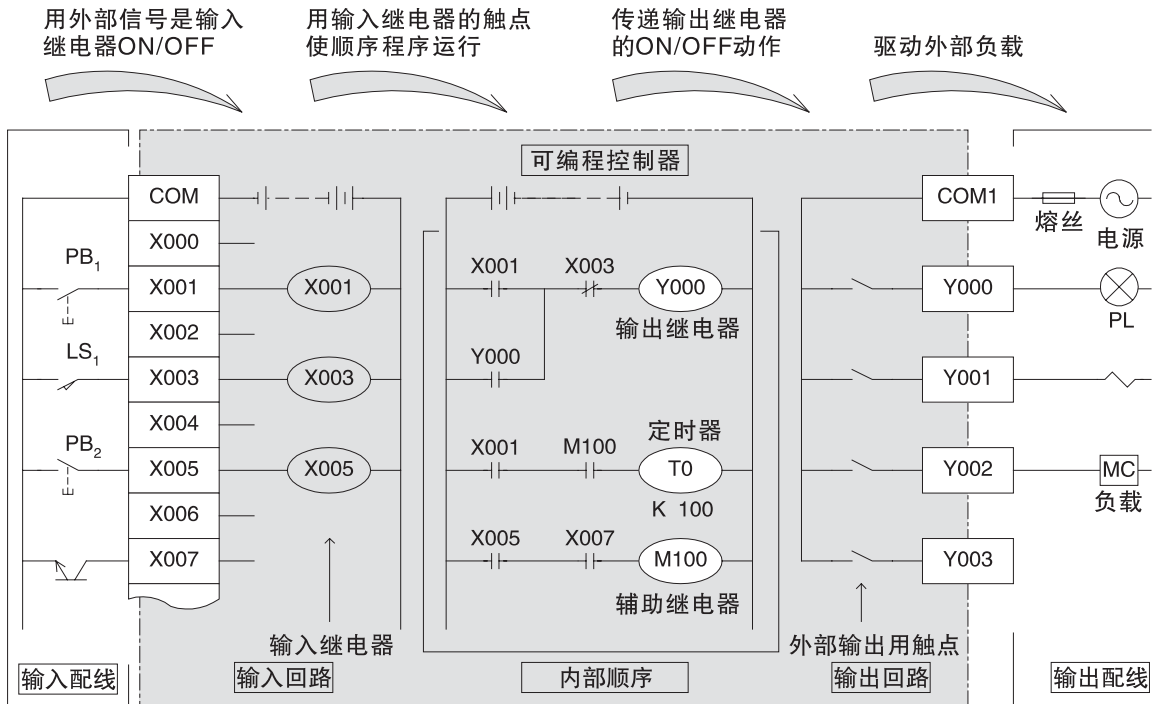


在可编程控制器上连接左页设备中的条件信号、指令信号、以及驱动负载。此外，将与输入侧相连的设备称作“输入设备”，将与输出侧相连的设备称作“输出设备”。在可编程控制器端子台上分别连接一个输入、输出设备。

只要像这样在可编程控制器上分别连接一个输入输出设备即可，用于进行顺序控制的连接在可编程控制器内部进行。

可编程控制器内部的连接用专用的顺序语言（指令）进行记述。该指令的组合称作顺序程序，根据该程序进行顺序控制。由此，无需在外部配线。

在实际中请视作继电器和定时器的集合体。



上图中包含了：输入设备、输出设备、顺序程序。

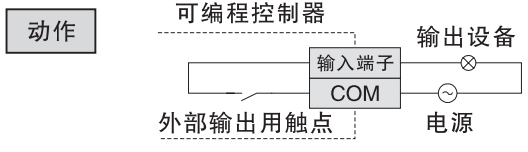
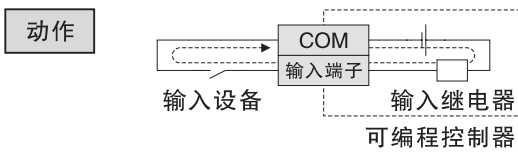
输入设备与可编程控制器的输入继电器相连，输出设备被外部输出用触点控制。

● 输入继电器

输入继电器起着将外部设备的信号转换成给可编程控制器的信号的作用。在上图中，输入设备只需连接到输入端子与COM端子之间即可动作。此外，通常继电器的触点再多也不过就几个，但在顺序程序中有着无数个触点。

● 输出用触点

输出用触点是利用顺序程序所驱动的输出继电器的触点来驱动外部负载的触点。外部输出用触点可用COM（公共端）来连接AC（交流）/DC（直流）的电源不同的设备。



在可编程控制器内部有输入继电器用的电源，若输入设备的触点导通，则电流像虚线那样流动，从而驱动输入继电器。像这样，在可编程控制器中，无论外部的输入设备是a触点还是b触点，若COM和输入端子导通则a触点闭合，若COM和输入端子导通则b触点打开。

若顺序程序中的输出继电器被驱动，则外部输出用触点合上。请将驱动输出设备的电源设置下可编程控制器的外部。

为了将与输入端子、输出端子相连的输入输出设备同顺序程序的输入继电器、输出继电器对照，在各端子上分配有软元件号（或I/O地址号）。

除了各端子号外，软元件号也付与可编程控制器内部具有的定时器和计数器。

- 软元件号：软元件号是表示软元件是什么的符号、以及对一个个进行区分的编号构成（软元件号也称作要素号）。

输入继电器 : X00 ~

是从可编程控制器外部的输入开关等接收信号的窗口，软元件符号用X表示。内置有与输入数（端子数）相应的输入继电器。

定时器 : T0 ~

是可编程控制器内部具有的定时器。定时器起着测量时间的作用，具有线圈和触点。若到达所设定的时间，则触点合上。

输出继电器 : Y00 ~

是驱动可编程控制器的外部负载的窗口，软元件符号用Y表示。内置有与输出数（端子数）相应的输出用触点。

计数器 : C0 ~

是可编程控制器内部具有的计数器。进行计数。若达到所设定的数值，则触点合上。

辅助继电器 : M0 ~

是可编程控制器内部具有的辅助性的继电器。（也称作内部继电器。）

- 输入继电器、输出继电器、辅助继电器、定时器、计数器等的使用数目因可编程控制器机种的不同而不同。

参 考

十进制数、八进制数、十六进制数

如下表所示，软元件号除了以十进制数来使用的外，也有八进制数和十六进制数的。

	输入继电器、输出继电器	辅助继电器、定时器、计数器
微型可编程控制器FX系列	八进制数	十进制数
通用可编程控制器Q/QnA/A系列	十六进制数	十进制数

所谓10进制数 就像一般所使用的0~9、10~19、20~29、…那样，是指以十个数为单位进一位的编号方式。

所谓8进制数 就像0~7、10~17、20~27、…那样，是指以八个数为单位进一位的编号方式。

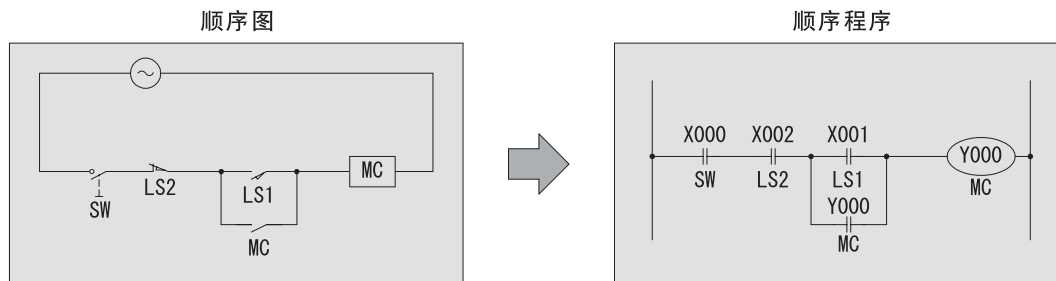
所谓16进制数 就像0~9、0A、0B、0C、0D、0E、0F、10~19、1A、1B、1C、1D、1E、1F、…那样，是指以十六个数为单位进一位的编号方式。

关于顺序图和顺序程序的表示

可编程控制器的顺序程序是转换成用于对配线到外部各端子上的输入设备、输出设备进行顺序控制用的回路的程序。…详细指令用语请参照第三章。

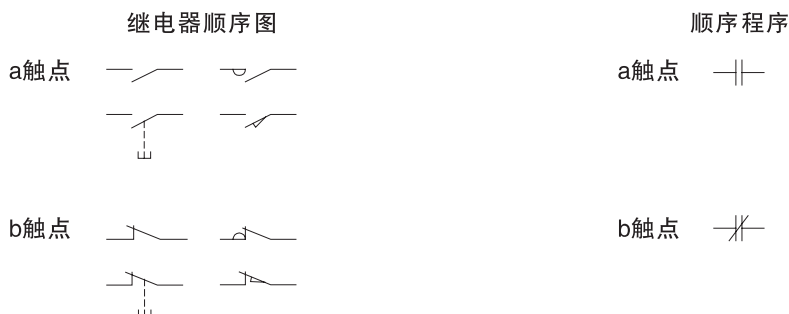
下面说明从顺序图到顺序程序的转换方法。

例子是将1-12页的水箱的水位控制转换成了顺序程序。



下面说明在转换时的几个不同点。

① a触点、b触点



② 不表示电源回路。

③ 在顺序程序中付有前页说明过的软元件号（要素号）。

重要内容

④ 虽然顺序图中的LS2是b触点，但在顺序程序中变为a触点。

首先，以上的顺序图中的LS2的作用是切断MC的自保持回路而停止MC，在回路上始终处于导通的状态。

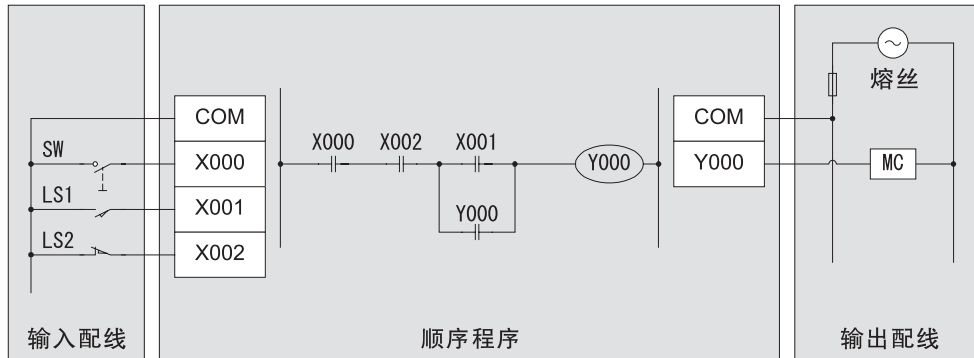
另外，考虑到2-4页的输入继电器的动作，在顺序程序中的a触点、b触点中，因外部配线中的触点的变化而通电时变为ON的是a触点，变为OFF的是b触点，因此为了获得相同的动作状态，在顺序程序中需使用a触点。

这样，在使用了可编程控制器时，在顺序程序中，各触点可使用a触点和b触点。

2.3 配线和程序

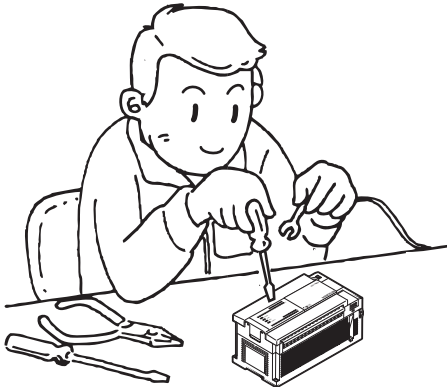
可编程控制器的配线和程序是怎么样的

下图对1-13页的回路图进行了分解。



可编程控制器的配线分为输入输出配线与内部配线进行处理。

输入输出配线需用钳子和起子进行以往那样的操作。



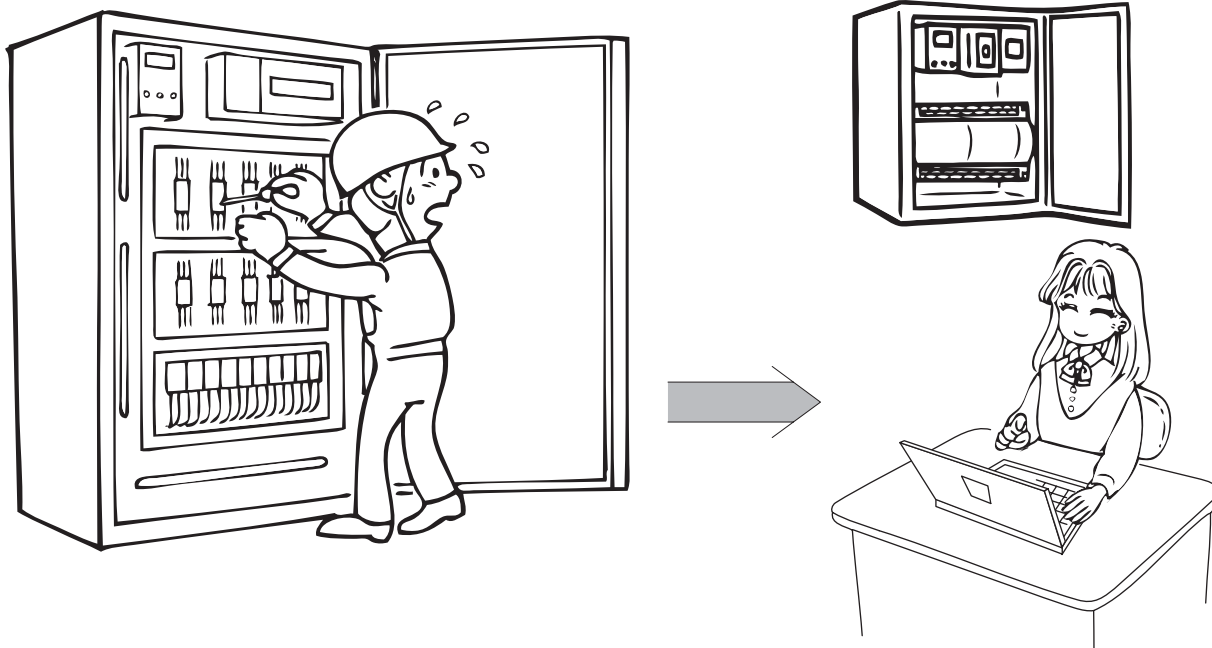
不过，复杂的内部配线（顺序程序）可利用编程面板通过按钮（键）操作容易地进行。



此外，输入端子与输入继电器线圈的连接以及输出继电器的外部输出用触点与输出端子的连接在出厂阶段就已完成。

2.4 能使用可编程控制器的好处

利用效果的好处



1

经济性

对于继电器约为10个以上的控制箱，一般可编程控制器价格较低。

2

设计省力

通过零件配线图的简化、顺序设计的简单化、试运转调整的简单化，设计大幅度省力。

3

操作工时的减少

通过准备零件的减少、对设备装置和控制箱的并行安排、规格变更的柔性、配线操作的简化，操作工时大幅度缩短。

4

小型化·标准化

与继电器箱相比被明显小型化，还可进行基于程序再利用的量产化。

5

可靠性提高

继电器、定时器故障减少，初始调整一完成即可放心使用。

6

维护性

提高寿命零件少，且附带故障诊断功能，因此维护也能方便地进行。

与继电器控制的比较

方式		继电器控制	可编程控制器控制
1	功能	只要使用许多继电器也可进行复杂的控制。	利用程序，无论多么复杂的控制都可以实现。
2	控制内容的可变性	只能采取更改配线。	只需更改程序即可，可自由地实现。
3	可靠性	通常使用中没问题，但存在接触故障和寿命限制。	核心部分全都是半导体，可靠性高。
4	通用性	制成的装置无法作他用。	根据程序，可用于任何控制。
5	装置的扩展性	需要追加、改造，困难。	只要有能力，可自由地扩展。
6	维护的容易性	需定期检修和更换寿命零件。	只需更换模块即可修理。
7	功能的丰富程度	仅有继电器控制	除顺序程序外，还可进行模拟和定位等的控制。
8	装置的尺寸	一般较大。	即使是很复杂的控制，也不会变大。
9	设计、制作时间	需要许多图，零件布置、装配试验需要时间。	即使是复杂的控制，由于设计简单，因此制作不费工夫。

参 考

可编程控制器补充内容

根据美国GM (General Motors) 公司的开放要求，1986年诞生了PC，并在1969年开始在美国销售。虽然1970年日本诞生了国产机，但通用机的出现则是在1976年以后。1977年，三菱电机开始销售通用机，并使单板形状的机器在市场上稳定了下来。之后，三菱电机在1980年发售了具有数值处理功能的通用机K系列，并终于在1981年发售了带编程器的破10万日元的微型可编程控制器F系列，从而可编程控制器正式开始普及。

备 忘 录

第3章 关于顺序指令

在之前的章节中……

已说明可编程控制器是许多继电器、定时器、以及计数器的集合体，其内部的顺序用于通过操作编程用的外围设备来进行编程。在该配线操作时，需要有与触点和线圈的连接方法、线圈的种类相应的规则，这就是指令。

指令的形式包括：指令用语 + 软元件号的指令形式，指令用语单独起作用的指令形式。

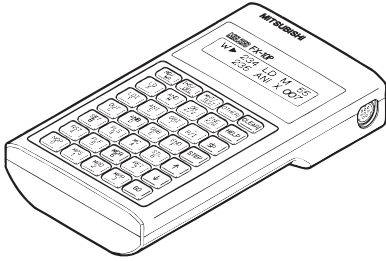
在本章中介绍几个运行可编程控制器的程序所需的基础指令。

请记住各个指令具有的含义。

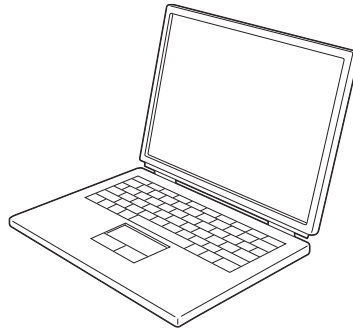
3.1 记住指令

指令和程序

用来输入顺序程序的外围设备（编程装置）包括：在屏幕上绘制顺序回路的设备、以及用指令用语来输入程序（列表程序）的设备。



用指令用语进行输入的设备



用回路图来输入的设备

下面表示的是可编程控制器具有的指令和各个指令所表示的回路图。

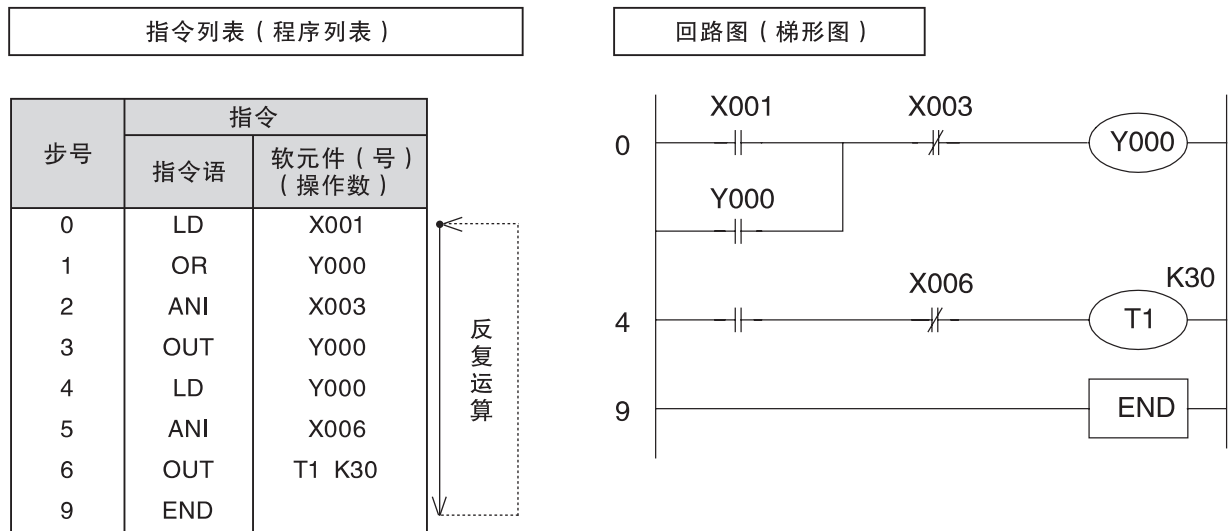
符号、称呼	功能	回路表示和对象软元件
LD 取	a触点 开始运算	
LDI 取反	b触点 开始运算	
AND 与	a触点 串联连接	
ANI 与非	b触点 串联连接	
OR 或	a触点 并联连接	
ORI 或非	b触点 并联连接	
ANB 与块	块之间 串联连接	
ORB 或块	块之间 并联连接	
OUT 输出	线圈驱动 指令	
SET 置位	动作保持 线圈指令	
RST 复位	动作保持 解除线圈指令	
NOP 空操作	无动作	删除程序或空格用
END 结束	程序结束	程序结束 返回0步

● 回路表示中使用的 —|— 、 —|/— 代表触点，因输入继电器、输出继电器、辅助继电器、定时器、计数器的 ON/OFF 而变为导通状态或不导通状态。

此外， —()— 、 —[SET]— 代表线圈的驱动。

程序的结构

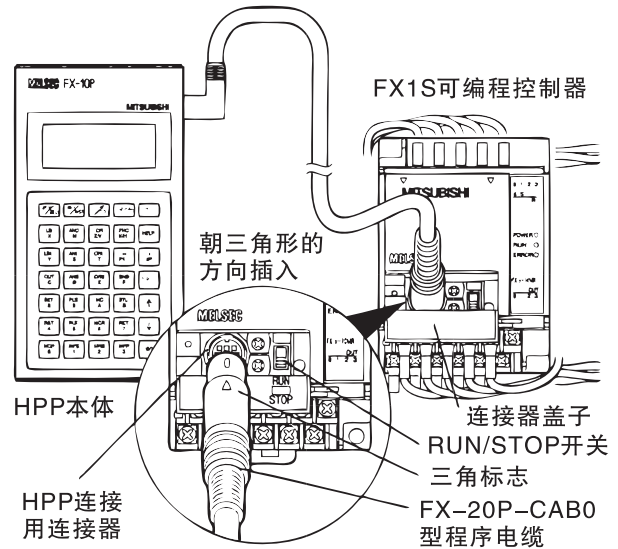
用于进行顺序控制的内部顺序作为顺序程序，作为指令列表（程序列表）或回路图（梯形图）生成。



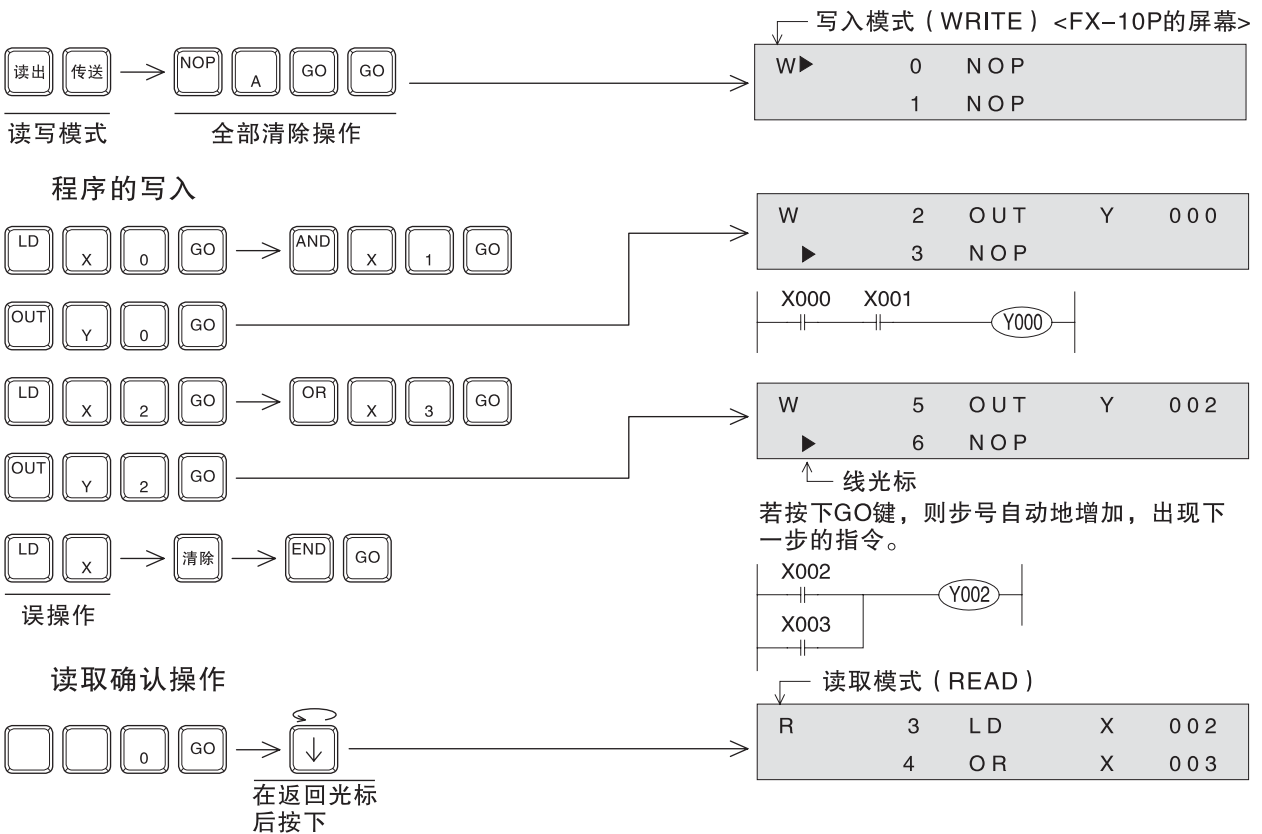
- 程序由许多指令语和软元件号（操作数）集合而成。
这些指令被依次赋予编号，这叫做步号。（步号自动地进行管理。）
- “指令”由“指令语 + 软元件”构成。但是，也有指令语后不带软元件的指令。
此外，有时仅由指令语构成指令。
- 程序的步数可在所使用的可编程控制器具有的“程序存储器容量”的范围内生成。
这叫做可编程控制器的程序容量。
例如，FX1S可编程控制器内置有“2000”步的程序存储器，FX1N、FX2N可编程控制器内置有“8000”步的程序存储器，FX3U可编程控制器内置有“64000”步的程序存储器。
- 指令从0步起到END指令为止反复执行。这叫做周期运算，循环一次所需的时间叫做运算周期（扫描时间）。
运算周期因程序内容和实际处理顺序的不同而发生变化，通常为几ms ~ 几十ms左右的值。
- 以回路图（梯形图）形式生成的顺序程序在可编程控制器内的程序存储器中也以“指令列表（程序列表）”形式进行存储。
在使用个人计算机的编程软件中进行着“指令列表（程序列表）”和回路图（梯形图）的相互转换。

程序的准备操作

1. 请打开可编程控制器上部的连接器盖子。
2. 将FX-20P-CABO型电缆的一端连接到可编程控制器的HPP连接用连接器上，将另一端连接到FX-10P的电缆连接用连接器上。
3. 使可编程控制器的RUN（运行）输入处于停止侧。
4. 打开可编程控制器的电源。由于HPP（编程装置）没有电源，因此利用可编程控制器经由程序电缆供给电源。
5. 按照下面的要领执行程序。



程序的全部清除



参 考



这些功能选择键以后一次按下的为优先，交替地切换。例如，若先按下 **读取/写入** 键，则进行读取，若按下两次，则进行写入，若按下三次，则又进行读取。

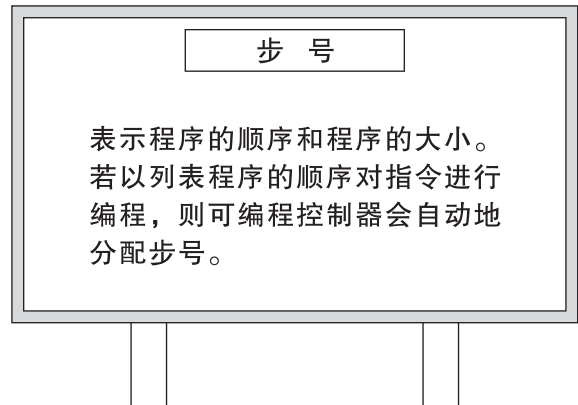
HPP的操作



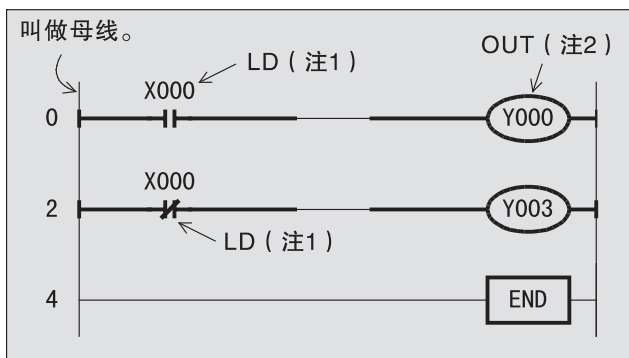
在这些键上，上部的符号为指令，下部的符号为软元件符号或数字，HPP根据操作顺序自动地进行判断。在改正误操作时，按下 **清除** 键。

记住指令语

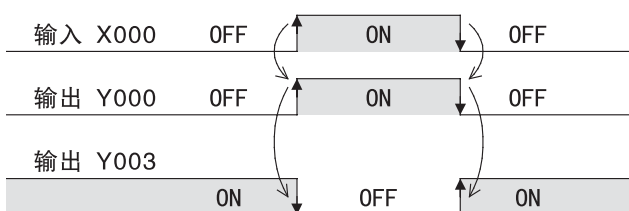
LD	取 a触点（常开触点） 运算开始
LDI	取反 b触点（常闭触点） 运算开始
OUT	输出 线圈驱动指令
END	结束 在结束程序时使用的指令



回路程序



在上面的程序中...



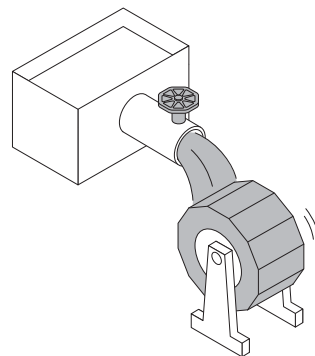
在输入X000为ON时，输出Y000为ON
在输入X000为OFF时，输出Y003为ON

列表程序

步	指令
0	LD X000
1	OUT Y000
2	LDI X000
3	OUT Y003
4	END

(注1) 相对于母线最先使用的a触点用LD（取）指令，b触点用LDI（取反）指令。LD和LDI等触点指令相对于输入继电器X、输出继电器Y、定时器T、计数器C、辅助继电器M等软元件进行使用。

(注2) OUT等线圈驱动指令用于输入继电器X之外的软元件。



SET

置位
动作保持输出指令

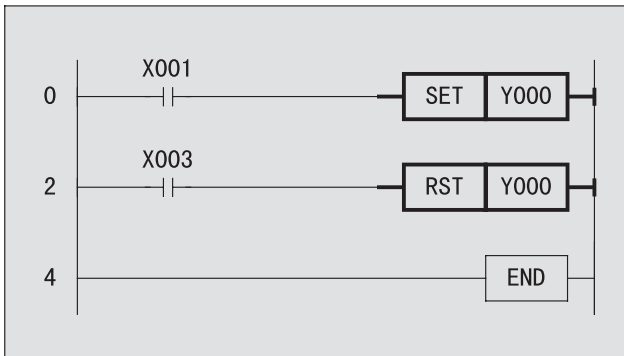
RST

复位
动作保持解除指令

SET/RST指令相对于输出继电器Y、辅助继电器M等使用。

另外，相对于计数器和乘法定时器也可使用RST指令。

回路程序

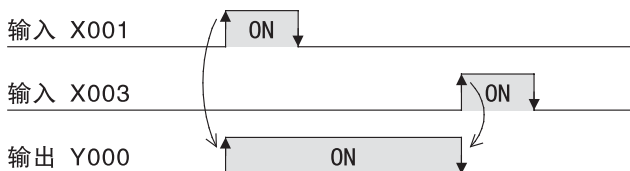


列表程序

步	指令
0	LD X001
1	SET Y000
2	LD X003
3	RST Y000
4	END

与OUT指令一样，是驱动线圈的指令。在OUT指令中，若驱动线圈的触点从ON变为OFF，则用OUT指令所写的线圈也变为OFF，但若使用SET（置位）指令，则即使触点从ON变为OFF，线圈仍然在ON的状态下保持动作。为了使由SET（置位）指令驱动的线圈从ON变为OFF，使用RST（复位）指令。

在上面的程序中...



若使输入X001变为ON，则输出Y000变为ON。之后即便使输入X001从ON变为OFF，Y000仍然在ON的状态下保持动作。
若使输入X003变为ON，则输出Y000变为OFF。

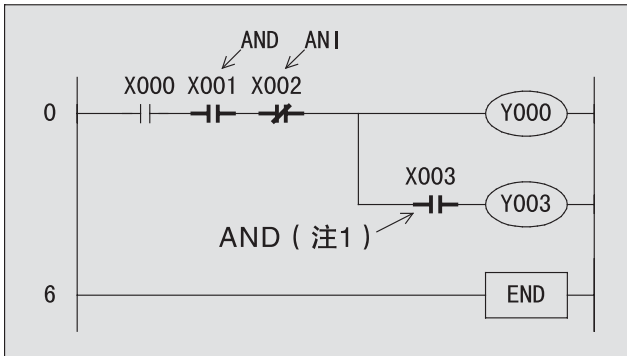
AND

与
a触点（常开触点）的
串联连接指令

ANI

与非
b触点（常闭触点）的
串联连接指令

回路程序



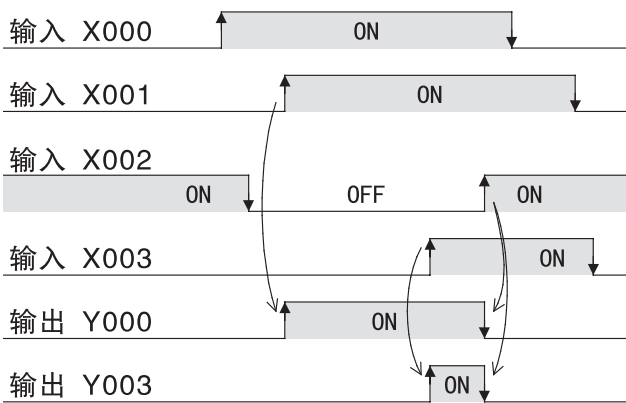
列表程序

步	指令
0	LD X000
1	AND X001
2	ANI X002
3	OUT Y000
4	AND X003
5	OUT Y003
6	END

在LD或LDI后串联连接的a触点使用AND（与）指令，b触点使用ANI（与非）指令。Y000和Y003在驱动其的串联触点全都导通时动作。

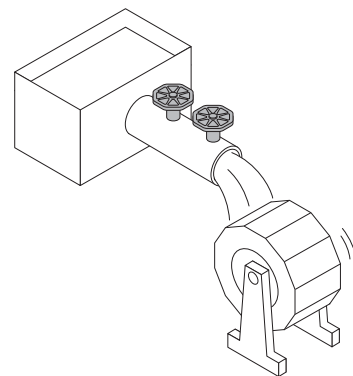
（注1）该位置的触点指令变为AND或ANI。请注意。

在上面的程序中...



在输入X000为ON、X001为ON、X002为OFF时，输出Y000变为ON。

在输入X000为ON、X001为ON、X002为OFF、X003为ON时，输出Y003变为ON。



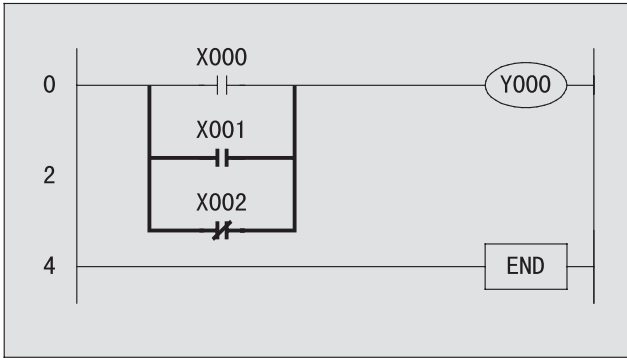
OR

或
a触点（常开触点）的
并联连接指令

ORI

或非
b触点（常闭触点）的
并联连接指令

回路程序



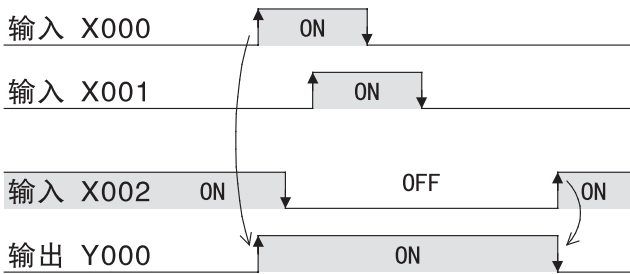
列表程序

步	指令
0	LD X000
1	OR X001
2	ORI X002
3	OUT Y000
4	END

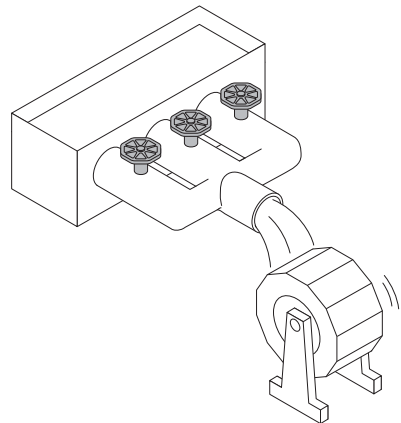
在LD或LDI后并联连接的a触点使用OR（或）指令，b触点使用ORI（或非）指令。

左图的输出Y000只要驱动其并联触点中有一个导通就动作。

在上面的程序中...



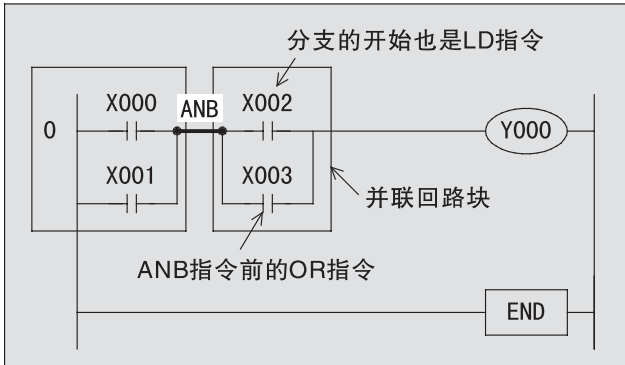
只要输入X000为ON、X001为ON、或X002为OFF中有一个条件满足，则输出Y000变为ON。



ANB

与块
并联回路块的
串联连接指令

回路程序



列表程序

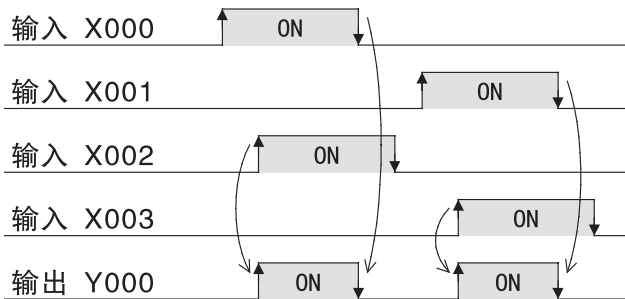
步	指令
0	LD X000
1	OR X001
2	LD X002
3	OR X003
4	ANB
5	OUT Y000
6	END

像左边那样，若串联的连接触点并联连接的回路，则使用ANB（与块）指令。ANB指令不带软元件号。

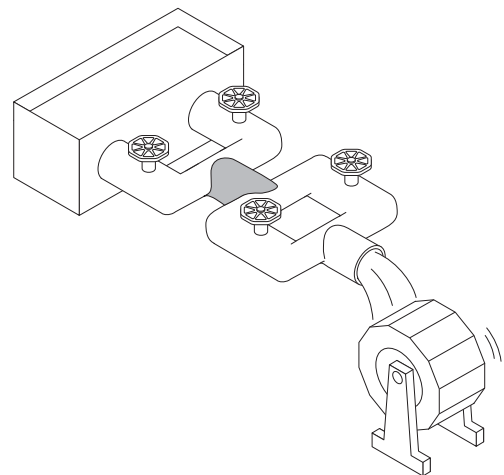
此外，在这种程序中，X002的触点也变为LD指令。

由于X001和X003是相对于作为LD触点的X000和X002并联连接的触点，因此构成OR指令。

在上面的程序中...



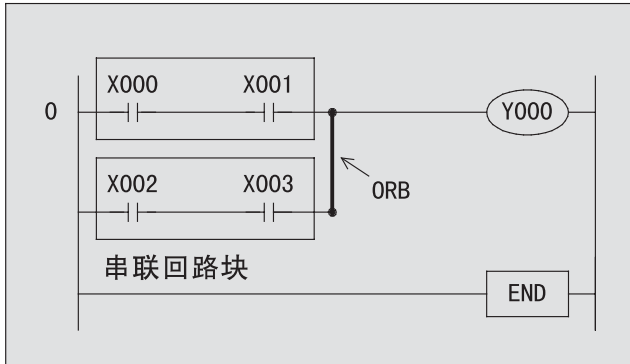
在输入X000或X001中的任一个和X002或X003中的任一个同时ON时，Y000动作。



ORB

或块
串联回路块的
并联连接指令

回路程序

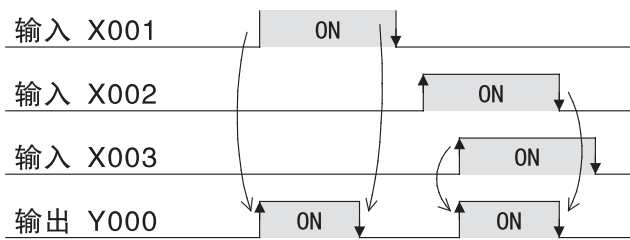
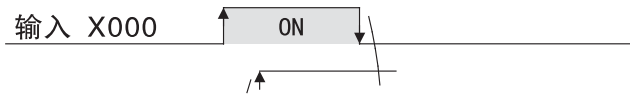


列表程序

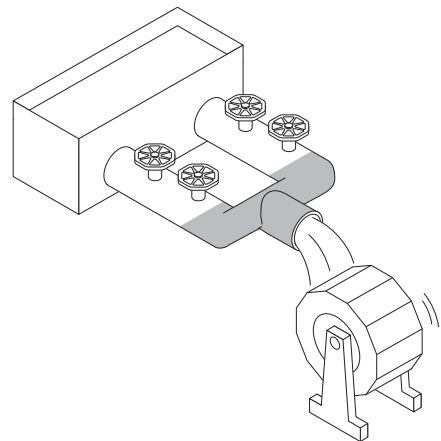
步	指令
0	LD X000
1	AND X001
2	LD X002
3	AND X003
4	ORB
5	OUT Y000
6	END

OR、ORI指令是将一个触点并联连接到紧靠其前的LD触点上的指令。然而，像左边那样，若并联地连接触点串联连接的回路，则使用ORB（或块）指令。此外，在这种程序中，X002的触点也构成LD指令。

在上面的程序中...



在输入X000和X001同时ON时，或在X002和X003同时ON时，输出Y000动作。



NOP

空（空操作）
误处理指令

在删除了全部程序时，所有指令变为NOP。

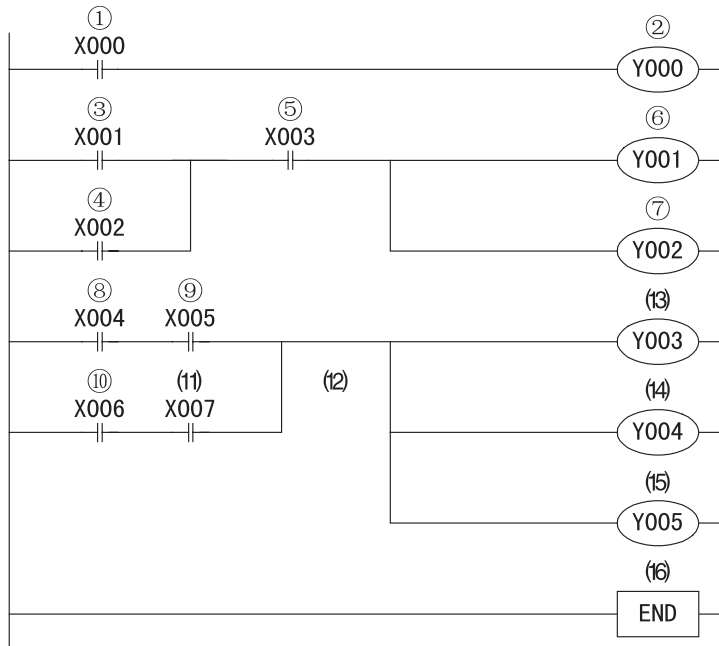
若在一般的指令与指令之间夹有NOP指令，则可编程控制器忽略其进行动作。

然而，由于需要多余的程序步、为了处理需要运算时间，因此请尽量删除。

3.2 程序的顺序

程序的顺序是什么

回路图的程序按从左到右、从上到下的顺序执行。



在如上的程序中，按①到(16)的顺序执行程序。
若改写成列表程序，则变为如下。

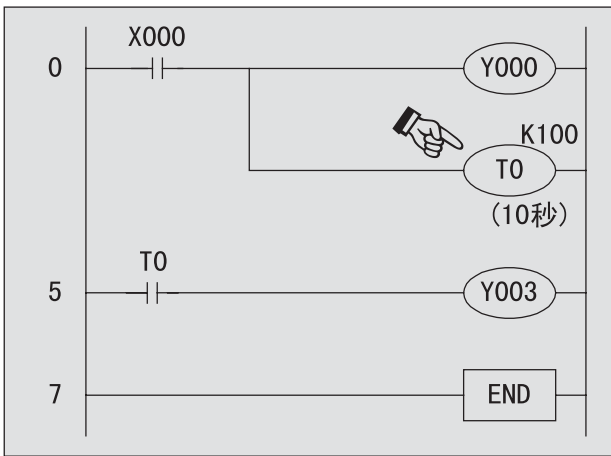
步	指令
0	LD X 000
1	OUT Y 000
2	LD X 001
3	OR X 002
4	AND X 003
5	OUT Y 001
6	OUT Y 002
7	LD X 004
8	AND X 005
9	LD X 006
10	AND X 007
11	ORB
12	OUT Y003
13	OUT Y004
14	OUT Y005
15	END

3.3 关于定时器回路

定时器

虽然存在各种各样的定时器，但在这里对微型可编程控制器具有的数字定时器的程序样式进行说明。

回路程序



试着使用监视功能!

键操作

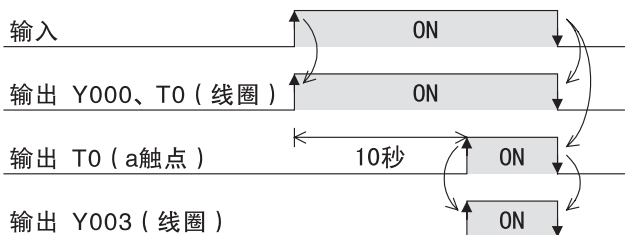


监视模式

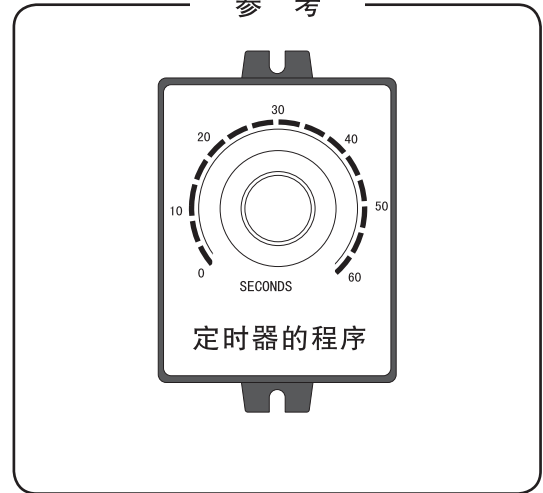


若选择监视模式，则在FX-10P屏幕的左上角显示M。

在上面的程序中...



参考



列表程序

步	指令
0	LD X000
1	OUT Y000
2	OUT T0
	SP K100
5	LD T0
6	OUT Y003
7	END

在个人计算机输入时为
[OUT T0 K100]

定时器的程序



设定值的输入

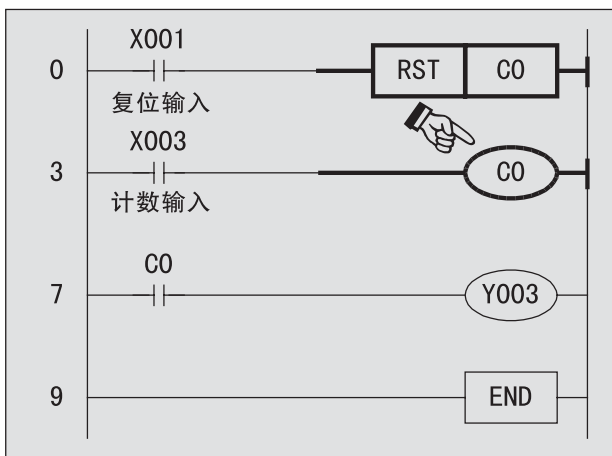
- 定时器触点在线圈通电后滞后规定时间进行动作。(通电延时定时器)
该规定时间称作设定值，用K表示。设定值K的值可设定为1~32,767。
例如，在K100时，构成10秒的定时器。
- 若在定时器驱动中使X000为OFF，则定时器的当前值变回0，定时器触点也变为OFF。

3.4 关于计数器回路

计数器

虽然存在着各种各样的计数器，但在这里对微型可编程控制器具有的普通计数器的程序样式进行说明。

回路程序



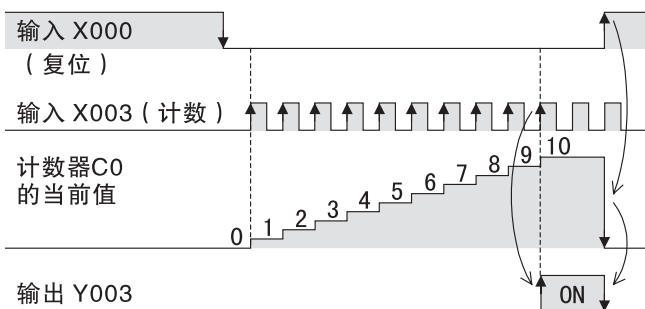
试着使用监视功能！

键操作

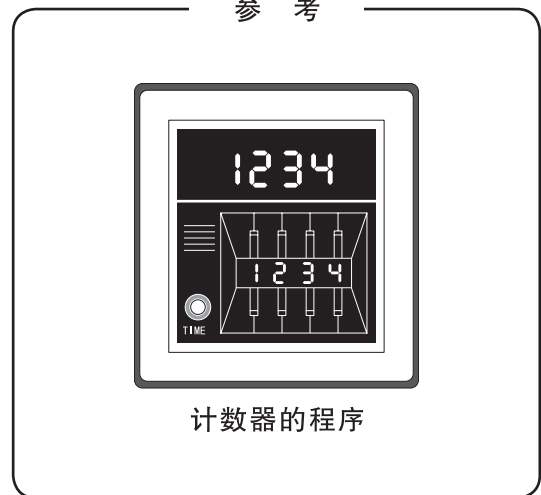


监视模式

在上面的程序中...



参考



列表程序

步	指令
0	LD X001
1	RST C0
2	LD X003
4	OUT C0 SP K10
7	LD C0
8	OUT Y003
9	END

在个人计算机输入时为
OUT C0 K10

计数器的程序



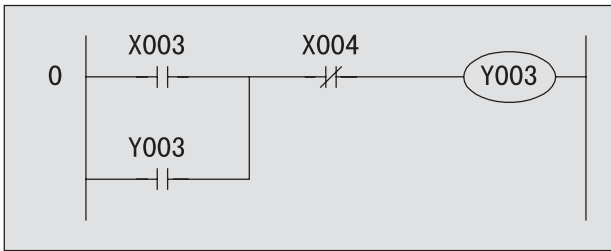
设定值的输入

- 计数器对触点 (X003) 从OFF变为ON的次数进行计数。
该触点 (X003) 称作计数输入，由计数器计得的数称作当前值。
此外，计数器触点在当前值达到规定数 (设定值) 时进行动作。
设定值可设定为1~32,767。
- 在计数到后，计数器的当前值不变化，输出触点也处于动作完的状态下。
- 若使复位输入变为ON，则计数器的当前值变为0，计数器触点也变为OFF。

3.5 关于自保持回路

利用顺序可实现输出的动作保持及其解除。

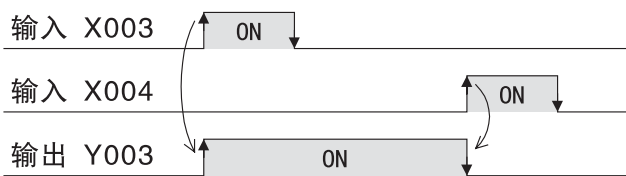
回路程序



列表程序

步	指令
0	LD X003
1	OR Y003
2	ANI X004
3	OUT Y003
4	END

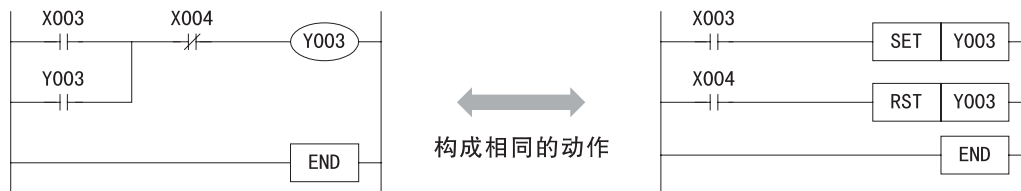
在上面的程序中...



- 在X003为ON、X004为OFF时，Y003变为ON。
- 即使使X003为OFF，Y003也继续动作（自保持动作），但若使X004为ON，则变为OFF。

参 考

两者都是相同的自保持回路



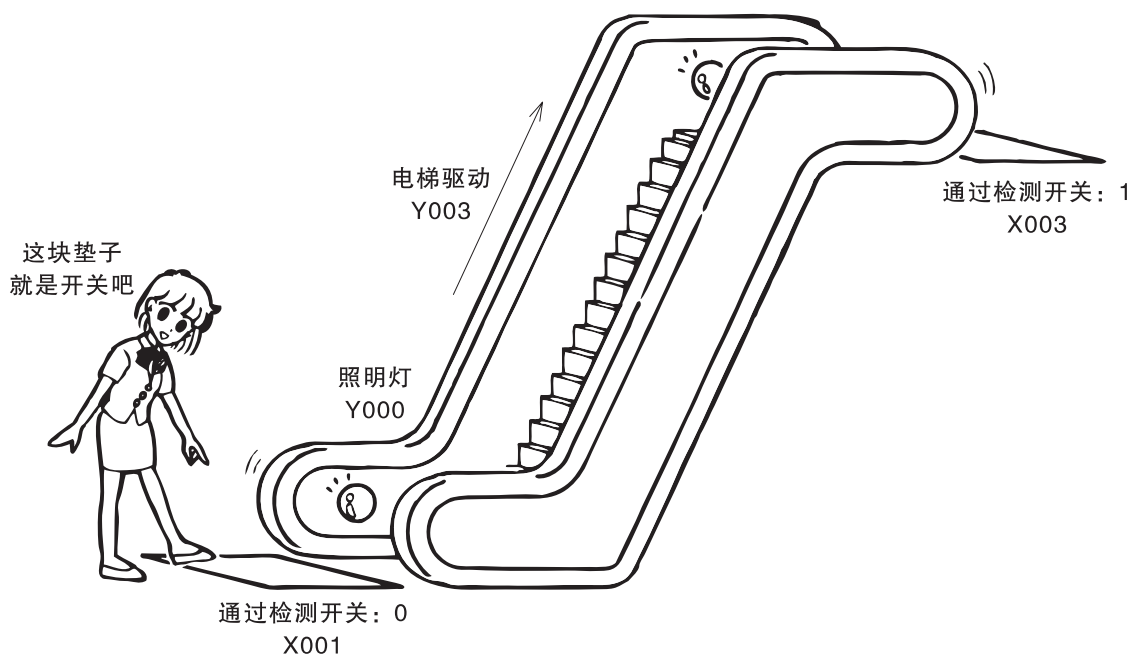
备 忘 录

第4章 程序练习

不需要太多的理论…

在本章中将基于简单的例题学习程序和监视等的整体操作。

4.1 引入事例1 (电梯控制)



让我们以电梯为例，考虑可编程控制器的程序。

《输入输出分配》

输入 通过检测开关: 0 X001
通过检测开关: 1 X003

输出 照明灯 Y000
电梯驱动输出 Y003

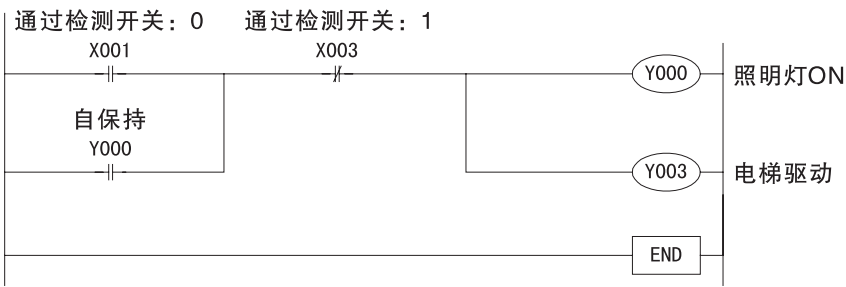
《动作》

- ① 当人靠近电梯前电梯不动作。
- ② 若人靠近电梯、使通过检测开关: 0变为ON, 则照明灯亮, 电梯动作。
(电梯仅上升)
- ③ 若人在被电梯运送到上面后从电梯上下来, 则通过检测开关: 1变为ON, 照明灯、电梯都返回不动作的状态。

在此, 请将通过检测开关0、1都视作非保持型开关。
此外, 设使用电梯的人并不连续地乘坐。

程序如下。

《回路程序》



《程序》

步	指令
0	LD X001
1	OR Y000
2	ANI X003
3	OUT Y000
4	OUT Y003
5	END

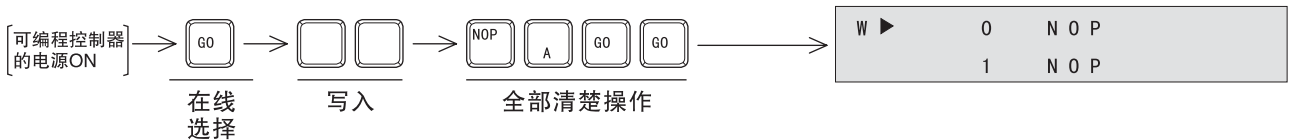
《试着确认动作》

试着将以上程序输入可编程控制器并进行动作的确认。

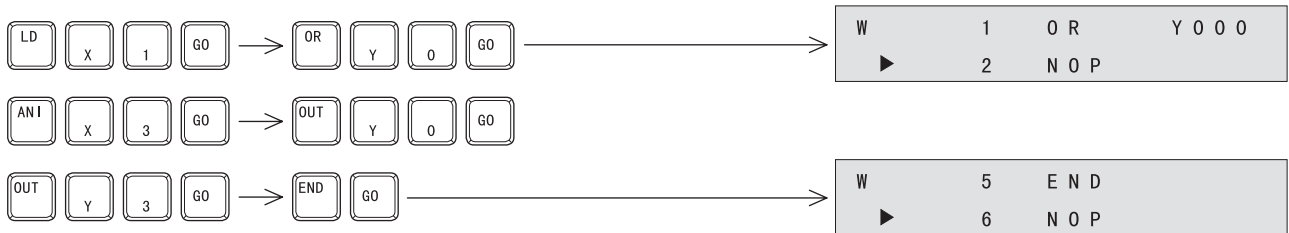
若使输入X001变为ON，则输出Y000和Y003动作。若接着使输入X003变为ON，则输出Y000和Y003不动作。

《键操作》

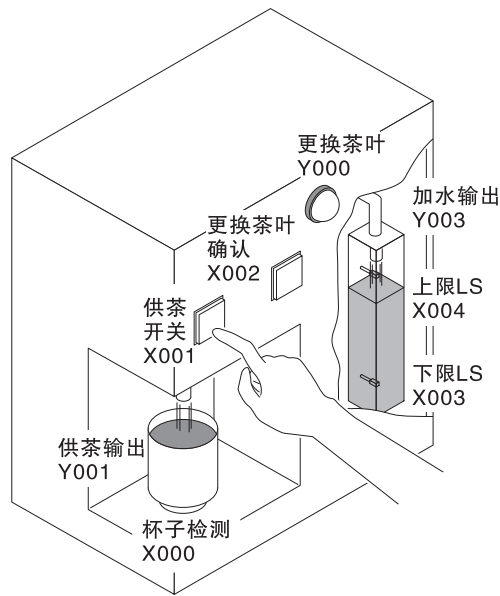
在进行了程序的准备操作（参照3-4页）后，请通过以下的键操作来执行程序。



程序的写入



4.2 引入事例2 (供茶器控制)



让我们以供茶器为例来考虑可编程控制器的程序。

《输入输出分配》

输入	杯子检测	X000 (在有杯子时变为ON)
	供茶按钮	X001
	茶叶确认按钮	X002
	加水箱下限LS	X003
	加水箱上限LS	X004

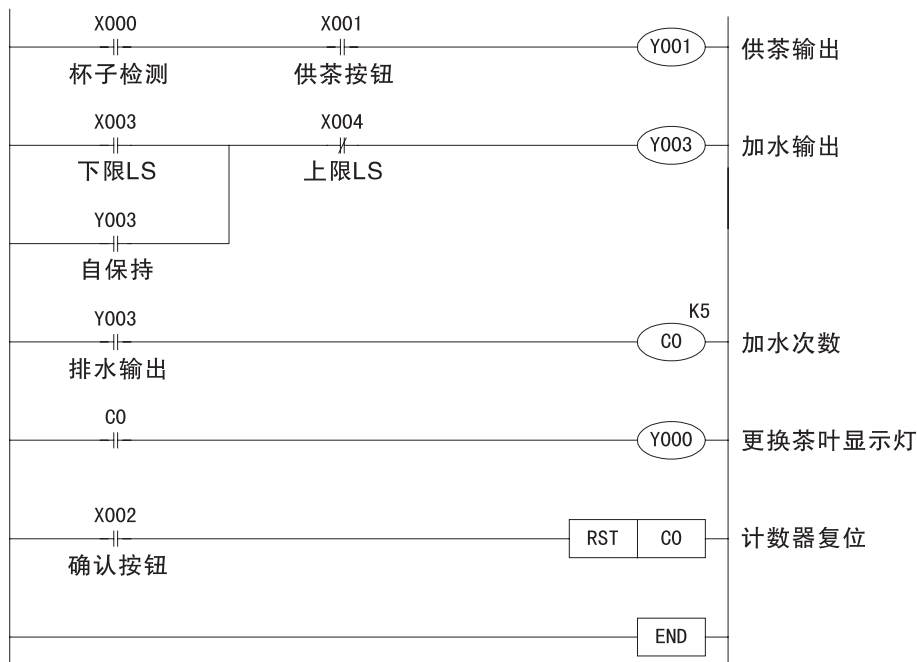
输出	茶叶更换显示灯	Y000
	供茶输出	Y001
	加水输出	Y003

《动作说明》

- ① 在杯子检测开关X000为ON时，若按下供茶按钮X001 (X001 ON)，则供茶输出Y001动作，将热水注入杯子中。热水仅在按下按钮时注入，在手离开按钮时停止。
在杯子检测X000为OFF时，即使按下供茶按钮X001，也不注入热水。
- ② 若加水箱的水变少，则下限LS X003变为ON，加水输出Y003动作。
若加水输出Y003动作、水被注入箱中，则最终上限LS X004变为ON，加水输出Y003变为不动作。
- ③ 若加水动作进行了5次，则更换茶叶显示灯亮。
- ④ 若按下确认按钮，则更换茶叶显示灯熄灭。

程序如下。

《回路程序》



《程序》

步	指令
0	LD X000
1	AND X001
2	OUT Y001
3	LD X003
4	OR Y003
5	ANI X004
6	OUT Y003
7	LD Y003
8	OUT C 0 SP K5
11	LD C 0
12	OUT Y000
13	LD X002
14	RST C 0
16	END

个人计算机
输入时为
[OUT C0 K5]

《试着确认动作》

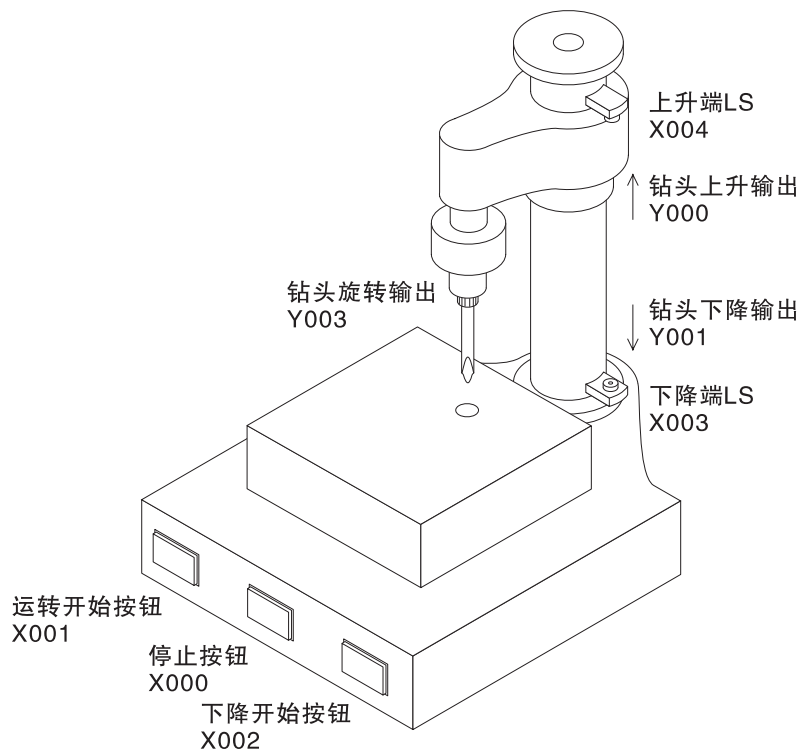
试着将以上程序输入可编程控制器中并确认动作。

- ① 在输入X000和X001都为ON时，输出Y001动作。
- ② 若输入X003变为ON，则输出Y003动作，若输入X004变为ON，则输出Y003变为不动作。
- ③ 每次输出Y003动作，计数器C0的值加1，若值变为5，则Y000动作。
- ④ 输入X002变为ON，则计数器C0的值变为0，输出Y000变为不动作。

(备考)

在FX-I/O演示机时，可调节输出Y001的蜂鸣器音量。若朝顺时针方向旋转蜂鸣器上面的音量，则音量变大，若朝逆时针方向旋转，则音量变小。

4.3 引入事例3 (钻床)



让我们以钻床的钻孔操作为例来考虑可编程控制器的程序。

《输入输出分配》

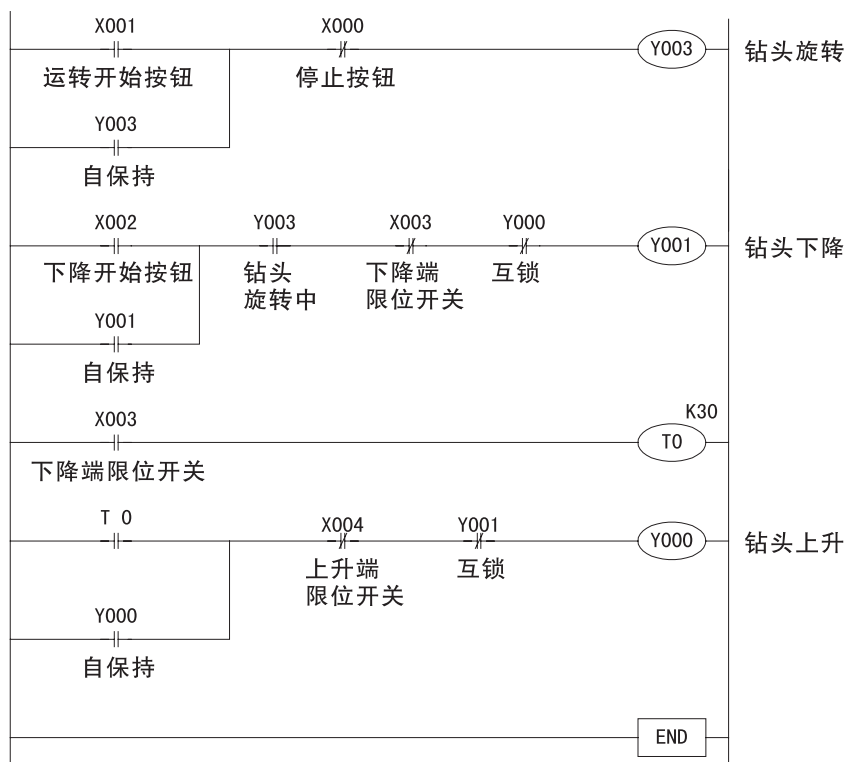
输入	停止按钮	X000	输出	钻头上升输出	Y000
	运转开始按钮	X001		钻头下降输出	Y001
	下降开始按钮	X002		钻头旋转输出	Y003
	(视作自动复位型按钮)				
	下降端限位开关	X003			
	上升端限位开关	X004			

《动作说明》

若按下运转开始按钮X001，则钻头旋转输入Y003动作，钻头开始旋转。
若按下停止按钮X000，则旋转输出Y003变为不动作，钻头的旋转停止。
在钻头旋转中，若按下下降开始按钮X002，则钻头下降输出Y001动作，钻头开始下降。
最后，若钻头到达下降端限位开关的位置，则下降端限位开关X003变为ON，钻头的下降动作停止。
在钻头的下降动作停止起3秒后，钻头的上升Y000动作，钻头上升。
若上升端限位开关X004变为ON，则开始上升的钻头结束上升动作。

顺序程序如下。

《回路程序》



《程序》

步	指令
0	LD X001
1	OR Y003
2	ANI X000
3	OUT Y003
4	LD X002
5	OR Y001
6	AND Y003
7	ANI X003
8	ANI Y000
9	OUT Y001
10	LD X003
11	OUT T0 SP K30
14	LD T0
15	OR Y000
16	ANI X004
17	ANI Y001
18	OUT Y000
19	END

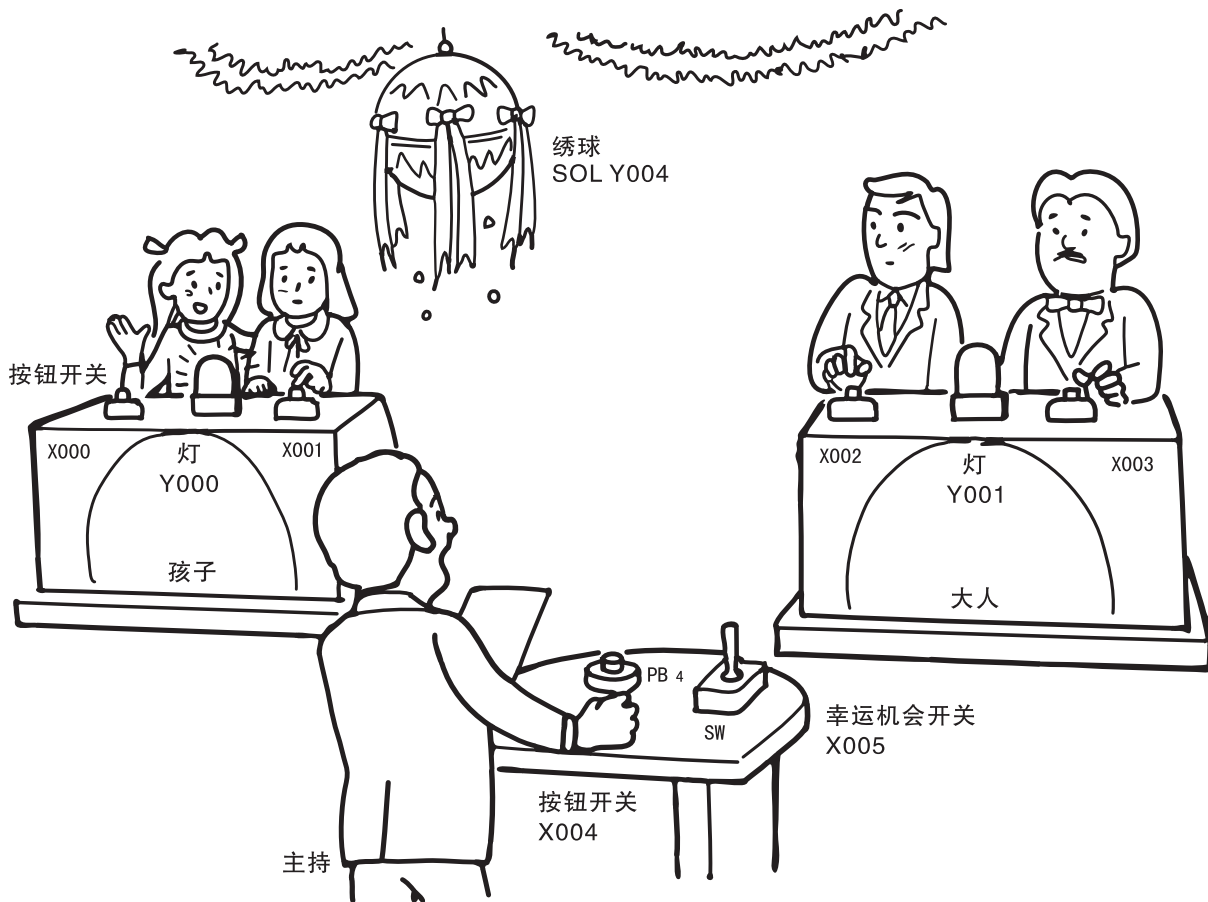
个人计算机
输入时为
[OUT T0 K30]

《试着确认动作》

试着将以上程序输入可编程控制器中并确认动作。

- ① 若使输入X001变为ON，则输出Y003动作。若输入X000变为ON，则输出Y003变为不动作。
- ② 在输出Y003动作时，若使输入X002变为ON，则输出Y001动作，若使输入X003变为ON，则输出Y001变为不动作。
- ③ 若使输入X003保持在ON的状态下，则定时器T0开始计数，3秒后T0触点变为ON。
- ④ 若定时器T0变为ON，则输出Y000动作，若输入X004变为ON，则输出Y000变为不动作。

4.4 引入事例4 (抢答显示控制)



让我们以答题节目为例来考虑可编程控制器的程序。

《输入输出分配》

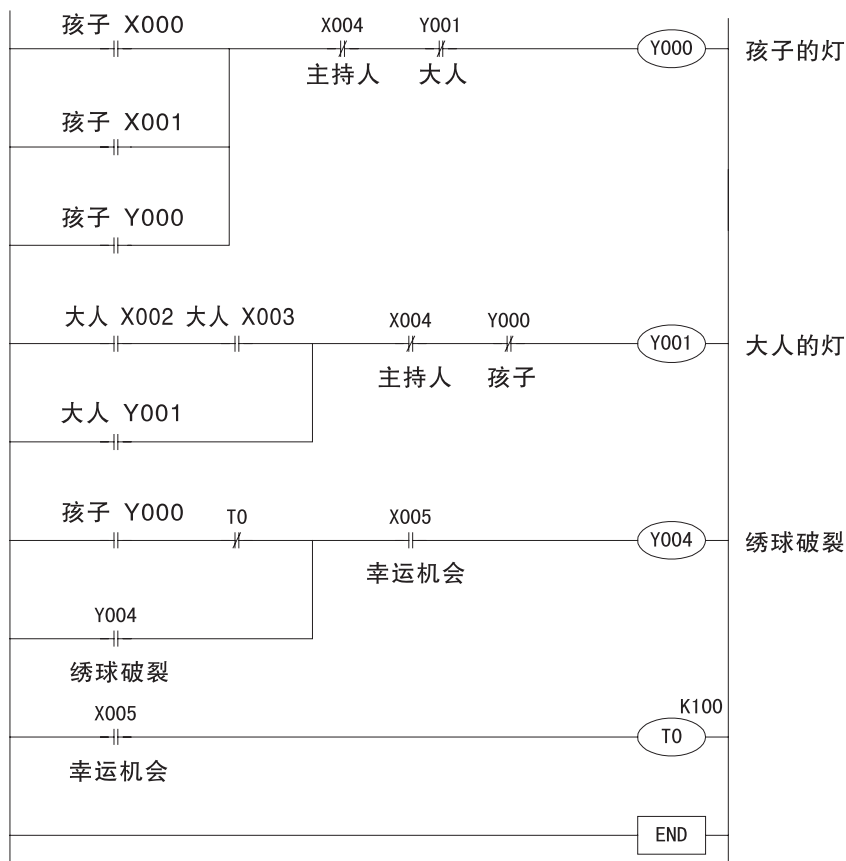
输入	孩子按钮	X000	X001	输出	孩子的灯	Y000
	大人按钮	X002	X003		大人的灯	Y001
	主持按钮	X004		输出		Y004
	幸运机会开关	X005				

《动作说明》

- ① 对于主持人所出的问题，使按钮先按下的桌子的灯亮。
- ② 灯在亮后直至主持人按下按钮X004为止一直亮着。但是，无论按下孩子一组的按钮X000和X001中的哪一个都能使孩子灯Y000亮，很有利。
大人一组若不按下X002和X003两者，则灯Y001不亮，不利。
- ③ 在主持人使幸运机会开关X005变为ON时，若孩子在10秒之内按下按钮，则绣球破裂，给予幸运机会。

程序如下。

《回路程序》



《程序》

步	指令	步	指令
0		11	
1		12	
2		13	
3		14	
4		15	
5		16	
6		17	
7		18	
8		19	
9		20	
10			

《试着确认动作》

试着将以上程序输入可编程控制器中并确认动作。

- ① 若X000或X001变为ON，则输出Y000动作并自保持。但是，若Y001先ON，则Y000不变为ON。
- ② 虽然Y001也作同样的动作，但若 X004变为OFF，则自保持解除（复位）。
- ③ 在幸运机会输入X005已是ON了时，若Y000变为ON，则绣球破裂输出Y004也变为ON进行自保持。但是，在 T0为OFF时不动作。若X005变为ON，则定时器T0开始动作，10秒后输出触点动作。

备 忘 录

附录1 GX Developer的操作

使用个人计算机来编制程序，程序的编制就变得很简单……

使用个人计算机软件GX Developer来编制程序，顺控程序的制作和编辑就变得有点象插图的感觉。

只要您掌握了基本操作，重复这些基本操作，就能够进行顺控程序的制作和编辑。

GX Developer是一种具有许多简便功能的软件，让我们从必要的操作入手来掌握使用方法吧。

顺利地进行起动和调整……

对程序而言。调试是必不可少的步骤。

因为在个人计算机的画面上能够监视PLC和程序的动作状态，所以可以尽早确认不动作部分并进行调整。

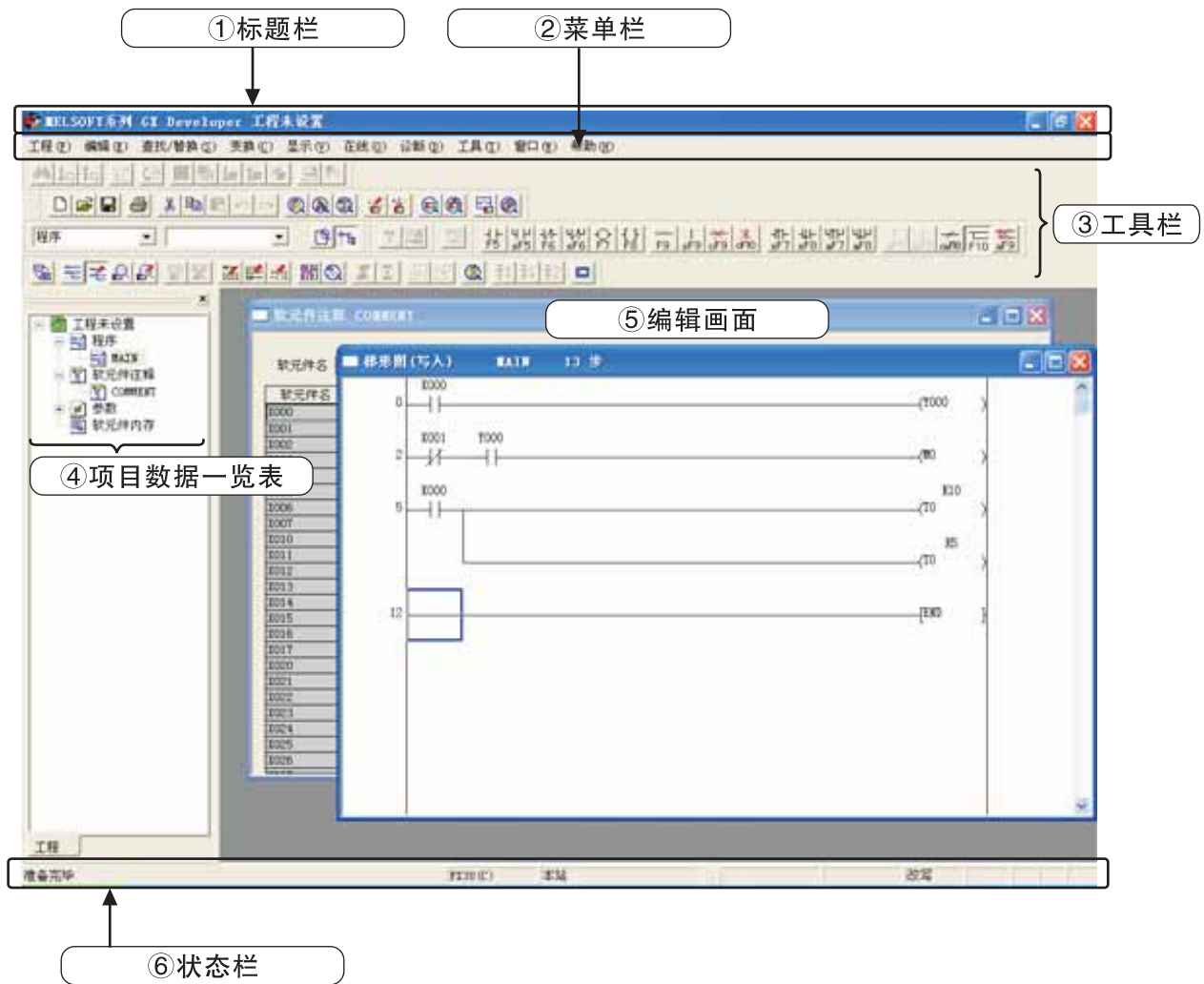
使程序的直观性变得更好……

GX Developer中配置有能够使顺控程序的直观性变得更好好的“注释输入功能”。

如果预先进行注释输入，就能提高顺控程序的制作和调试的效率。

附录1.1 GX Developer操作的基础知识

附录1.1.1 GX Developer画面的构成



① 标题栏

显示已经打开的项目名和窗口图标。

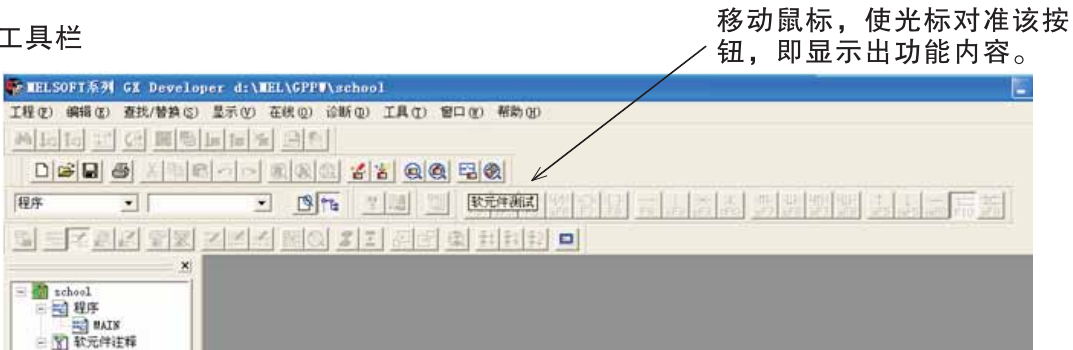


② 菜单栏



选择菜单，即显示出下拉菜单。

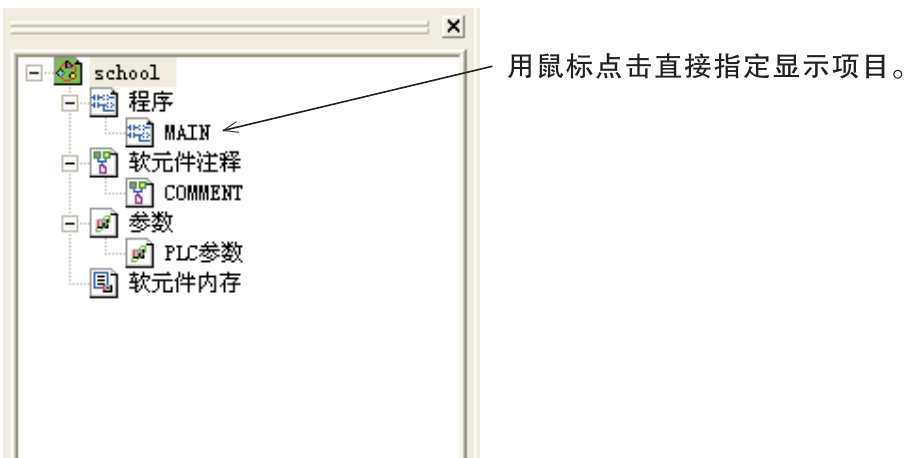
③ 工具栏



*: 工具栏的内容是可以移动和装卸的，所以，显示项目和配置因不同环境而异。

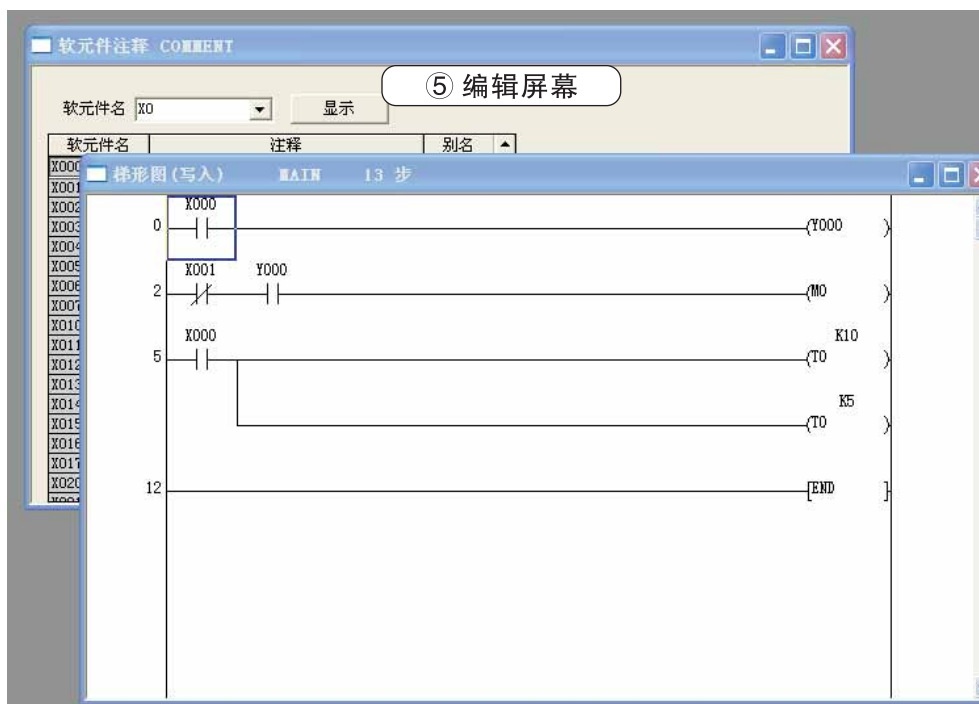
将使用频度高的功能配置为功能按钮，为了在菜单栏中进行选择，可以直接执行以比较为目的的功能。

④ 项目数据一览表



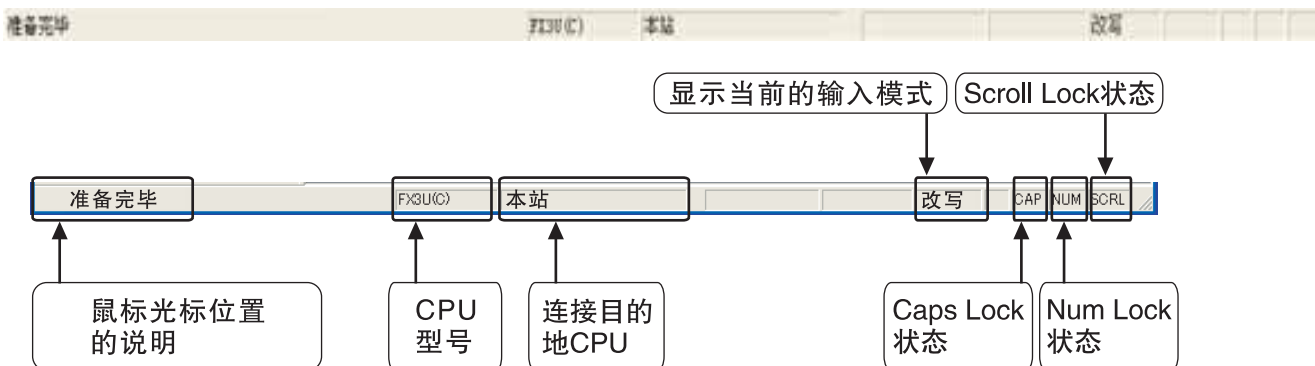
梯形图制作窗口和参数设置画面等的“树”显示。

⑤ 编辑屏幕



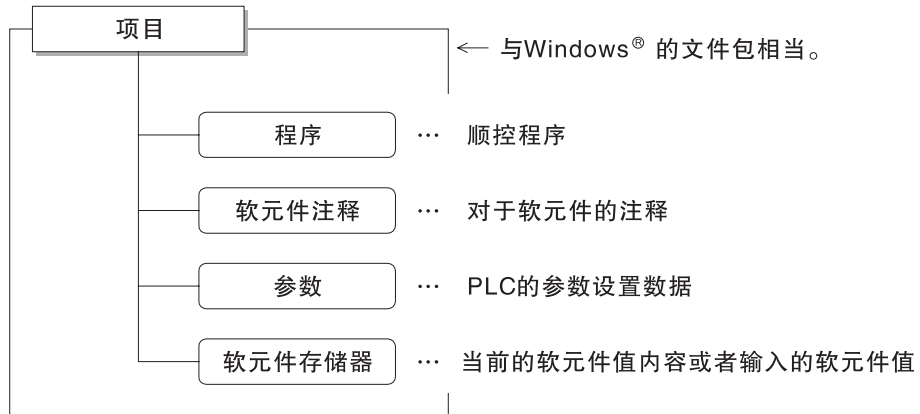
回路生成屏幕和监视屏幕等以窗口形式多个地显示。

⑥ 状态栏



附录1.1.2 项目

这里的所谓“项目”，是指程序、软元件注释、参数、软元件存储器的一种集合体。在GX Developer中，把一连串数据的集合体称之为“项目”，被当作Windows的文件包进行保存。

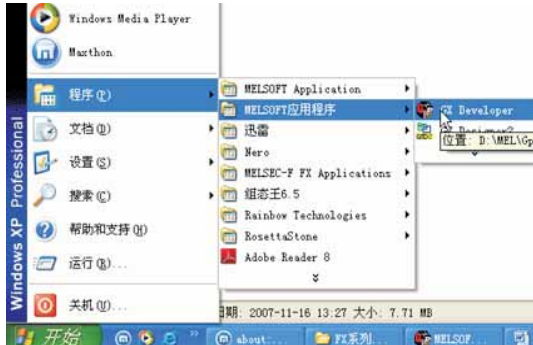


● 多个项目的编辑

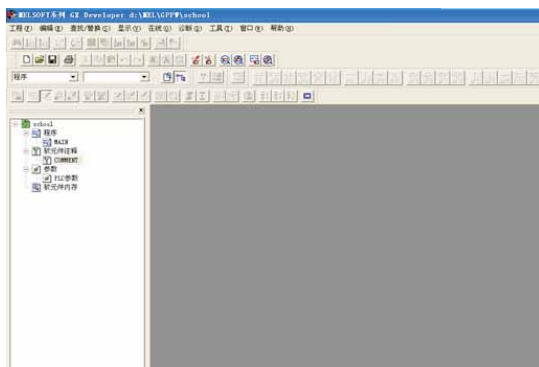
要使用GX Developer进行2个或2个以上项目的编辑时，应该起动多个GX Developer。

附录1.2 GX Developer的启动和新项目的制作

附录1.2.2 GX Developer的启动

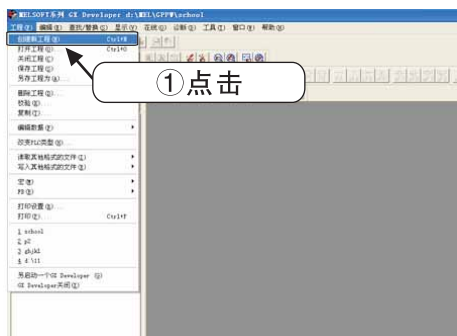





- ① 点击Windows的 **启动** 按钮。
依次进行下列选择：
[全部程序]
↓
[MELSOFT]
↓
[GX Developer]

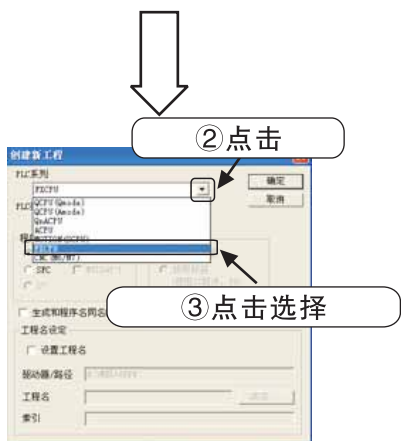


- ② GX Developer启动。

附录1.2.3 新项目的制作

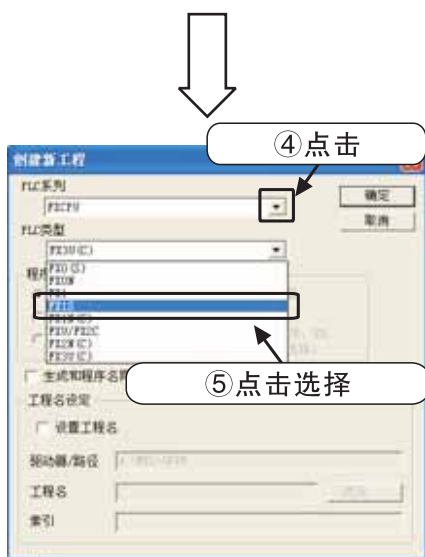


- ① 选择工具栏的 ，或者从菜单栏中选择 [项目]→[制作新项目] ( + )。



- ② 点击“PC系列”的[▼]按钮。

- ③ 选择“FXCPU”。

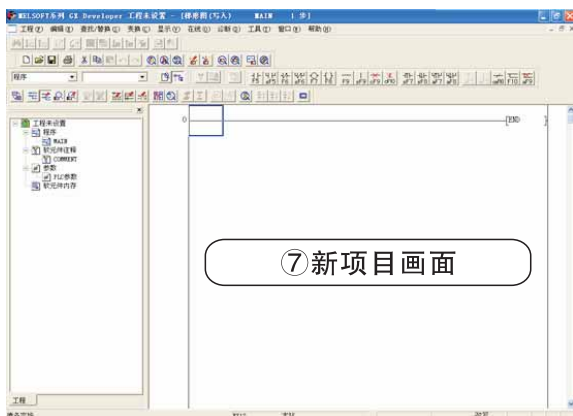
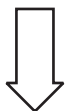


- ④ 点击“PC型”的[▼]按钮。

- ⑤ “FX1S”
(注) 应该选择实际使用的系列名。



⑥ 点击 按钮。



⑦ 显示出新项目画面，呈现可输入程序状态。

要点

- 关于参数设置

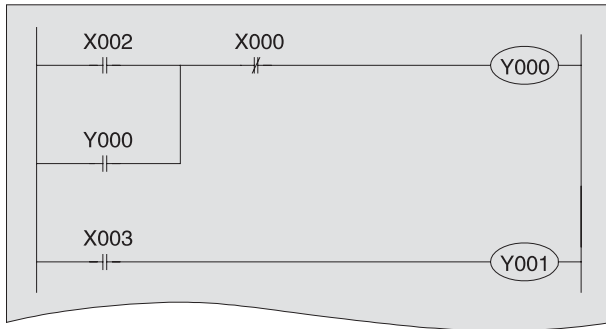
对于FX·PLC来说，如果不使用PLC中存储的注释设置，不使用文件寄存器，就不必进行参数设置。

关于参数内容的说明，请参见附录部分。

附录1.3 梯形图的制作

附录1.3.1 使用功能键制作梯形图

[制作的梯形图]

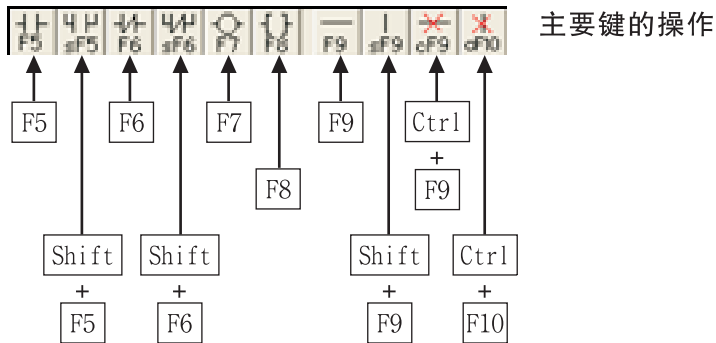


要点

本书中，用3位的“X000”和“Y000”表达输入继电器(X)和输出继电器(Y)的编号。在GX Developer画面上进行输入时，“X0”、“Y1”和左前方的“0”可省略，不输入。

要点

- 功能键和梯形图符号的关系显示在工具栏中的按钮上。

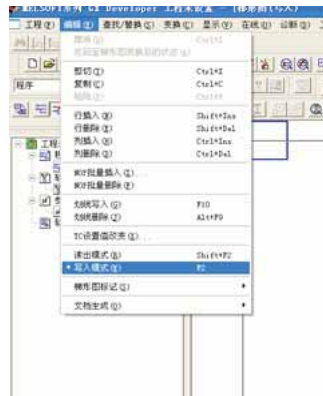


- 制作梯形图时，必须先设置在“写入模式”。

从工具栏中选择



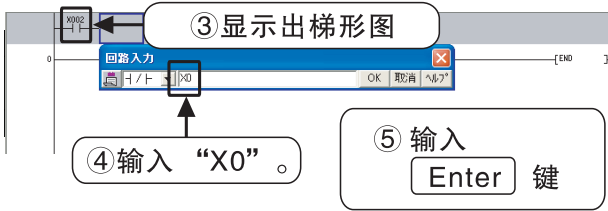
从菜单中选择 ([编辑]→[写入模式])



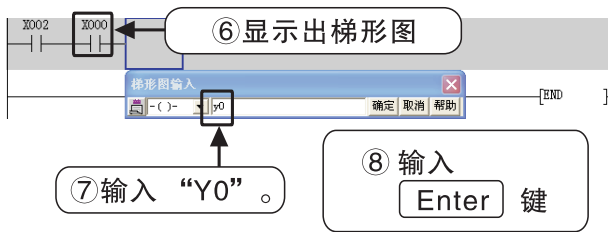
- 输入字符时要全部采用半角字符进行输入，不能采用全角字符。



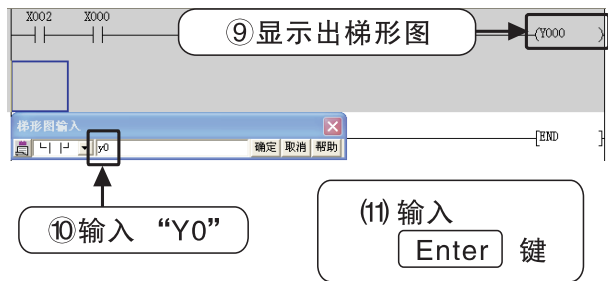
- ① 输入“X2”
- ② 输入 Enter 键



- ④ 输入“X0”。
- ⑤ 输入 Enter 键



- ⑦ 输入“Y0”。
- ⑧ 输入 Enter 键



- ⑩ 输入“Y0”
- (11) 输入 Enter 键

- ① 点击 F5 (⇧) 键。
输入“X2”。



用 ESC 或者[取消]键取消

- ② 用 Enter 键或者[OK]键确定。

- ③ 显示出输入的梯形图 (X²)。

- ④ 按 F6 (⇧) 键。
输入“X0”。

- ⑤ 用 Enter 键或者 [OK] 键确定。

- ⑥ 显示出梯形图 (X⁰)。

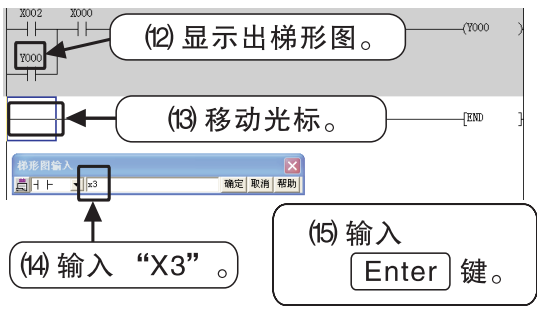
- ⑦ 按 F7 (⇧) 键。
输入“Y0”。

- ⑧ 用 Enter 键或者 [OK] 键确定。

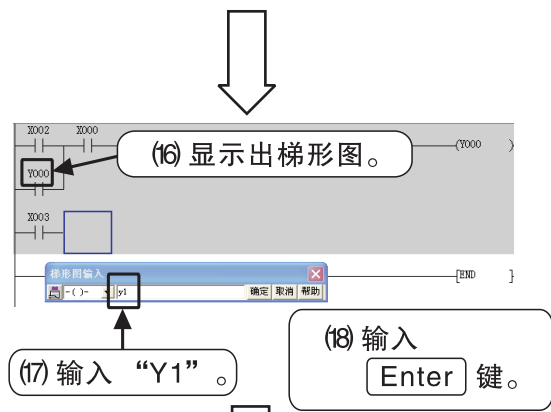
- ⑨ 显示出输入的梯形图 (-(Y⁰)-)。

- ⑩ 按 Shift + F5 (⇧) 键。
输入“Y0”。

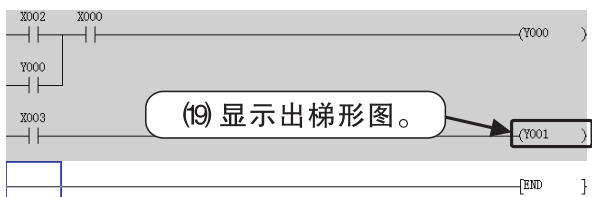
- (11) 用 Enter 键或者 [OK] 键确定。



- (12) 显示出输入的梯形图 ($\begin{matrix} Y0 \\ | \\ \text{---} \\ | \\ \text{---} \end{matrix}$)。
- (13) 将光标移动到下一行的开头处。
- (14) 按 **F5** ($\begin{matrix} + \\ | \\ \text{---} \\ | \\ \text{---} \end{matrix}$) 键。
输入“X3”。
- (15) 用 **Enter** 键或者 [OK] 键确定。



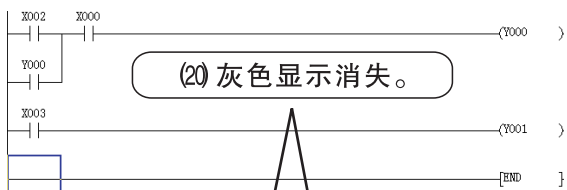
- (16) 输入显示出梯形图 ($\begin{matrix} X3 \\ | \\ \text{---} \\ | \\ \text{---} \end{matrix}$)。
- (17) 按 **F7** ($\begin{matrix} - \\ | \\ \text{---} \\ | \\ \text{---} \end{matrix}$) 键。
输入“Y1”。
- (18) 用 **Enter** 键或者 [OK] 键确定。




- (19) 显示出输入的梯形图 ($\begin{matrix} - \\ | \\ \text{---} \\ | \\ \text{---} \end{matrix} (Y1) \end{matrix}$)。
!! 梯形图制作结束!!

↓

F4 (转换)



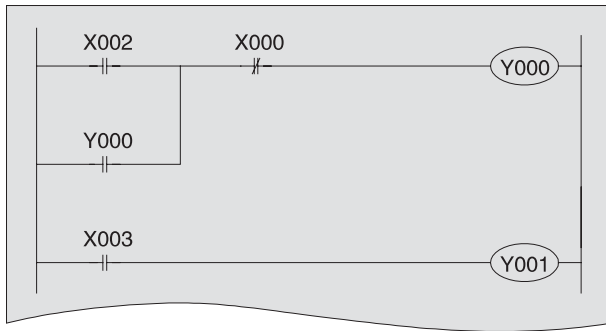
- (20) **梯形图转换操作【重要】**
为了确定尚未确定的梯形图 (灰色显示部分)，进行[转换]操作。

按 **F4** (转换) 键。
或者，选择工具栏中的  键。
或者，从菜单中选择“[转换]→[转换]”。

灰色显示消失，梯形图得到确定。
发生出错时，光标会移动到梯形图制作有问题的部分上。请进行梯形图修正。

附录1.3.2 使用工具按钮制作梯形图

[制作的梯形图]

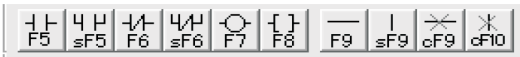


要点

本书中，用3位的“X000”和“Y000”表达输入继电器(X)和输出继电器(Y)的编号。在GX Developer画面上进行输入时，“X0”、“Y1”和左前方的“0”可省略，不输入。

要点


- 功能键和梯形图符号的关系显示在工具栏中的按钮上。



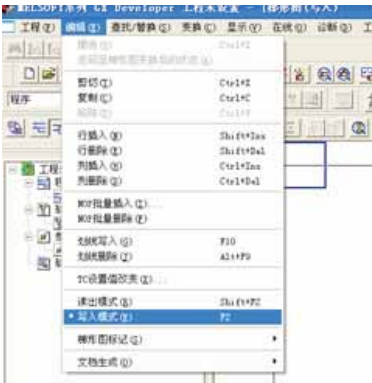
主要键的操作

- 制作梯形图时，必须先设置在“写入模式”。

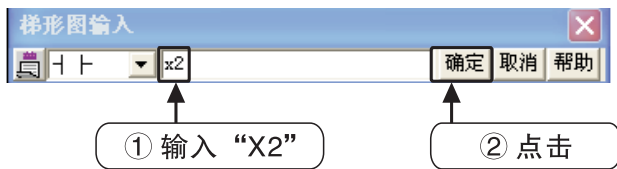
从工具栏中选择



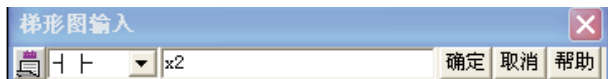
从菜单中选择 ([编辑]→[写入模式])



- 输入字符时要全部采用半角字符进行输入，不能采用全角字符。

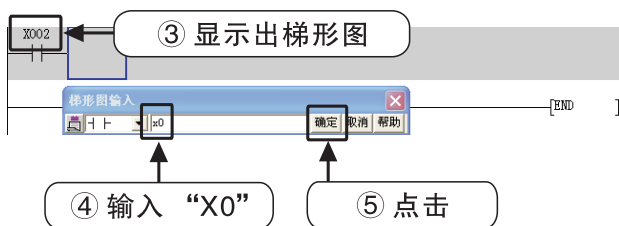


- ① 点击工具按钮 ($\frac{11}{F5}$)。
输入“X2”。



用 [ESC] 或者[取消]键取消

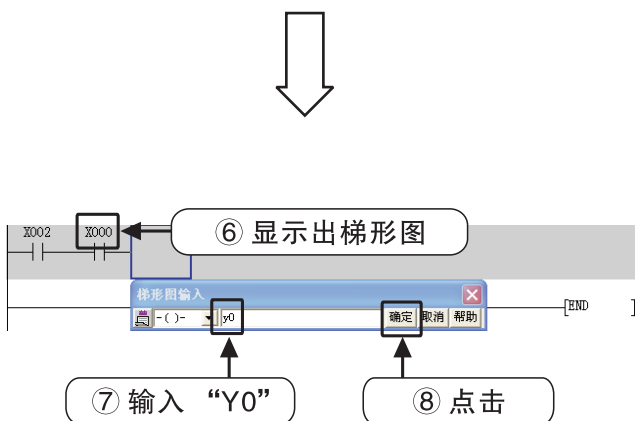
- ② 用 [Enter] 键或者 [OK] 键确定。



- ③ 显示出输入的梯形图 ($\frac{X2}{|}$)。

- ④ 点击工具按钮 ($\frac{11}{F6}$)。
输入“X0”。

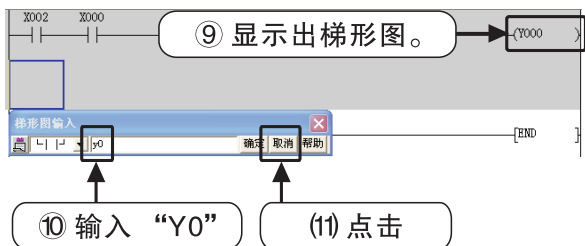
- ⑤ 用 [Enter] 键或者 [OK] 键确定。



- ⑥ 显示出梯形图 ($\frac{X0}{|}$)。

- ⑦ 点击工具按钮 ($\frac{11}{F7}$)。
输入“Y0”。

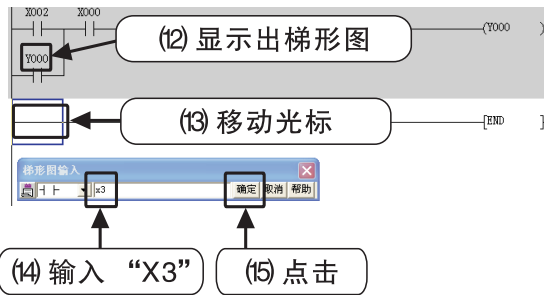
- ⑧ 用 [Enter] 键或者 [OK] 键确定。



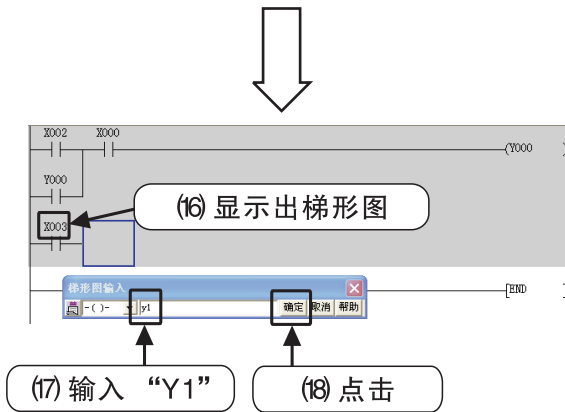
- ⑨ 显示出输入的梯形图 ($\frac{-Y0}{|}$)。

- ⑩ 点击工具按钮 ($\frac{41}{SF5}$)。
输入“Y0”。

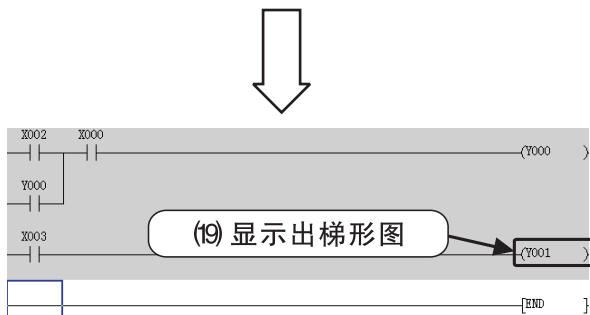
- (11) 用 [Enter] 键或者 [OK] 键确定。



- (12) 显示出输入的梯形图 ($\begin{matrix} Y0 \\ | \\ \hline \end{matrix}$)。
- (13) 将光标移动到下一行的开头处。
- (14) 点击工具栏中的按钮 $\begin{matrix} \text{X} \\ | \\ \hline \end{matrix}$ 。
- 输入“X3”。
- (15) 用 键或者 [OK] 键确定。

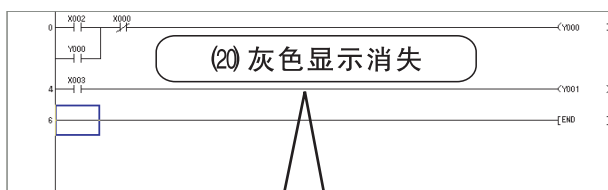


- (16) 显示出输入的梯形图 ($\begin{matrix} X3 \\ | \\ \hline \end{matrix}$)。
- (17) 点击工具栏中的按钮 $\begin{matrix} Y \\ | \\ \hline \end{matrix}$ 。
- 输入“Y1”。
- (18) 用 键或者 [OK] 键确定。



- (19) 显示出输入的梯形图 ($\begin{matrix} - \\ | \\ \hline \end{matrix} Y0$)。
- !! 梯形图制作结束!!

(转换)



- (20) 梯形图转换操作【重要】
- 为了确定尚未确定的梯形图(灰色显示部分), 进行[转换]操作。

按 (转换) 键。
 或者, 选择工具栏中的 $\begin{matrix} \text{X} \\ | \\ \hline \end{matrix}$ 键。
 或者, 从菜单中选择 “[转换]→[转换]”。

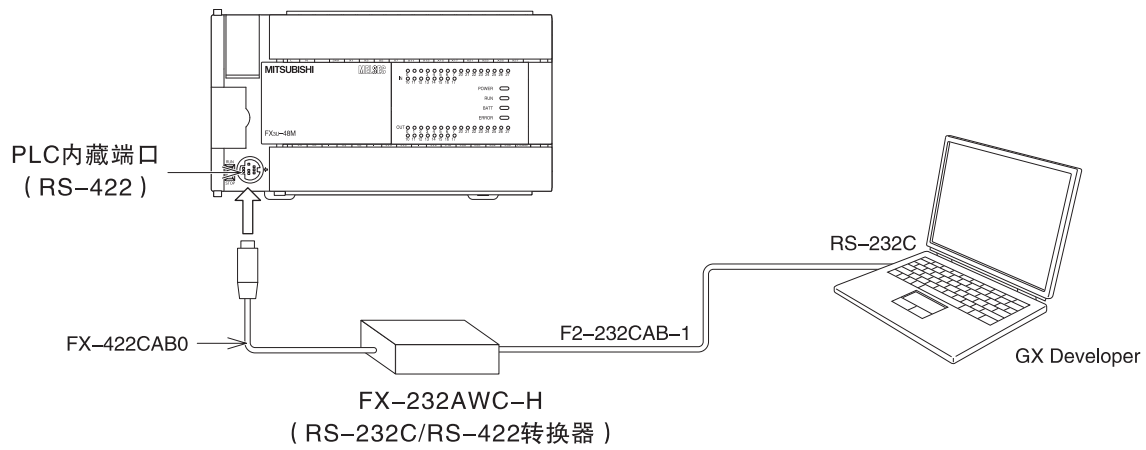
灰色显示消失, 梯形图得到确定。
 发生出错时, 光标会移动到梯形图制作有问题的部分上。请进行梯形图修正。

附录1.4 将程序写入PLC中

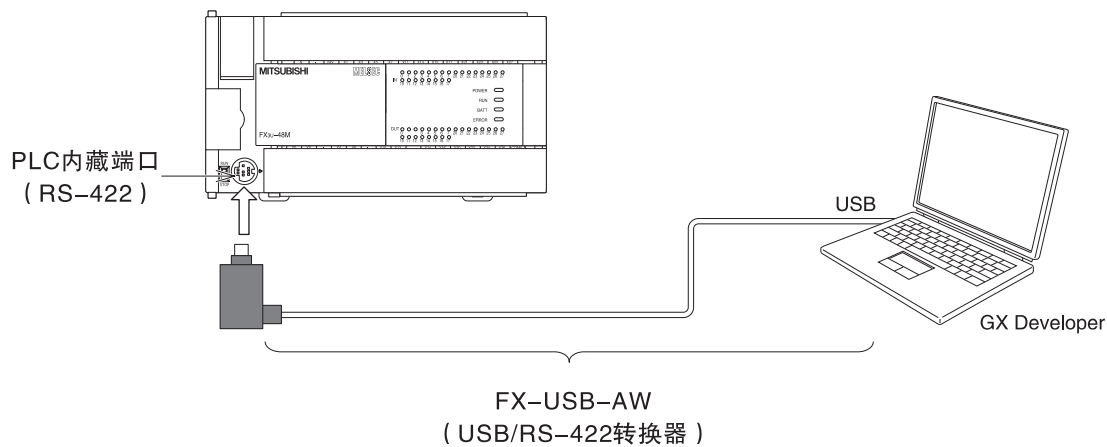
将制作成的顺控程序写入FX·PLC中。

附录1.4.1 与PLC的连接

① 连接例（个人计算机方面：RS-232C）

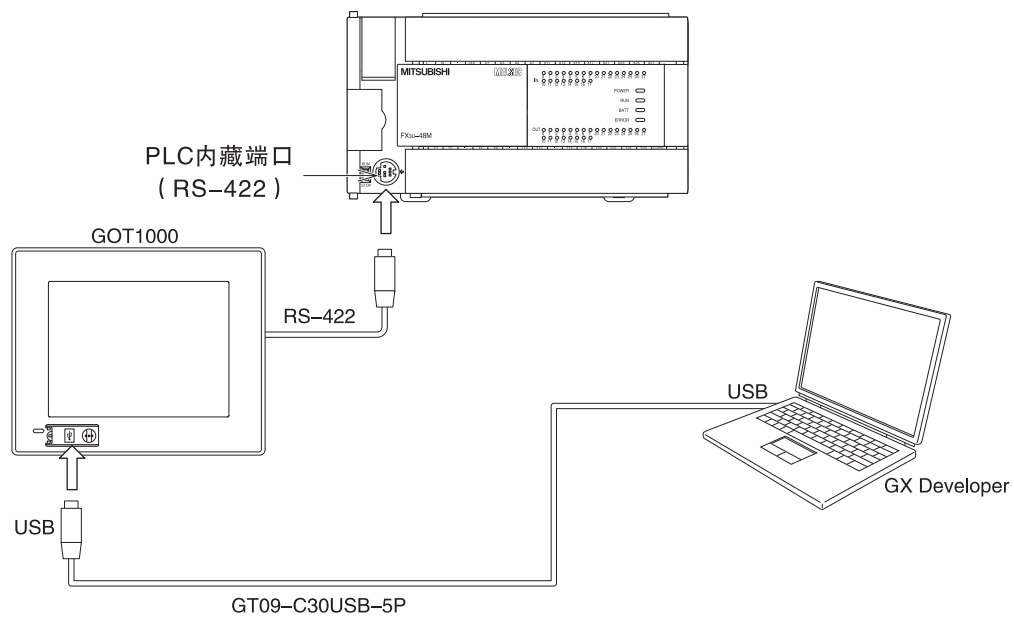


② 连接例（个人计算机方面：USB）



- 应该对FX-USB-AW的驱动器分配给个人计算机的COM端口编号进行确认。确认方法请参见FX-USB-AW的说明。

③ GOT1000的透明功能（个人计算机方面：USB）



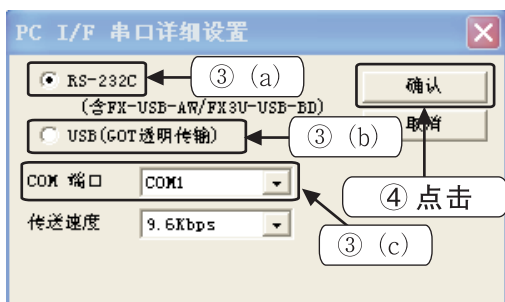
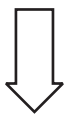
附录1.4.2 GX Developer的“连接对象设置”

为了GX Developer与PLC的通信而进行的设置。



① 进行“[在线]→[指定连接对象]”的菜单点击操作。

② 双击图标。



③ 设置个人计算机方面的通信端口。

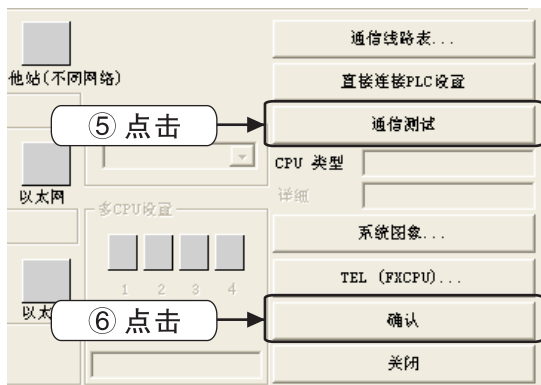
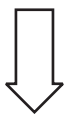
(a) 如果个人计算机方面的连接器是RS-232C连接器，或者，个人计算机方面的连接器是RS-USB连接器，使用的是FX-USB-AW，则应该选择“RS-232C”。

(b) 如果使用GOT1000的透明功能，个人计算机方面的连接器是USB连接器，则应该选择“USB (GOT透明)”。

(c) · 如果个人计算机方面的连接器是RS-232C连接器，通常选择COM1（也有因个人计算机而异的）。

· 使用FX-USB-AW时，由驱动器指定分配端口编号（参见3.4.1）。

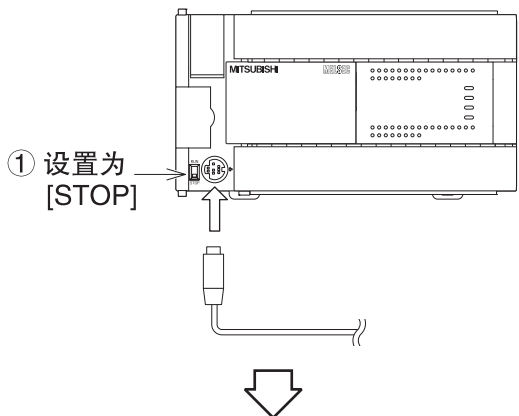
④ 设置后，点击[O]按钮。



⑤ 点击[通信试验]按钮，确认与PLC的通信。

⑥ 确认后，点击[OK]按钮，确定设置内容。

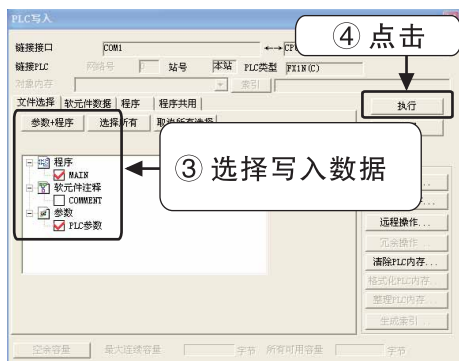
附录1.4.3 程序的写入



① 将PLC上的[RUN/STOP] 开关设置在 [STOP]位置上。



② 选择工具栏中的 , 或者, 在菜单栏中选择[在线]→[PC写入]。



③ 点击[参数+程序]。

④ 点击[执行]。

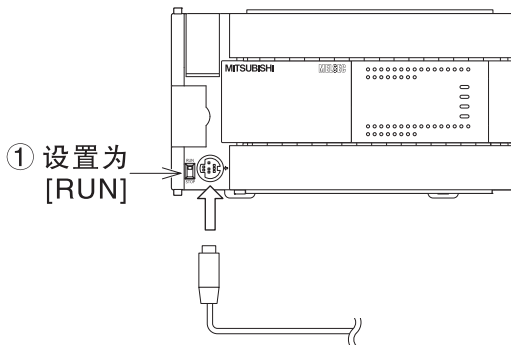


显示出表示写入经过的对话框。




⑤ 结束后, 点击[OK]按钮。

附录1.4.4 程序动作的监视



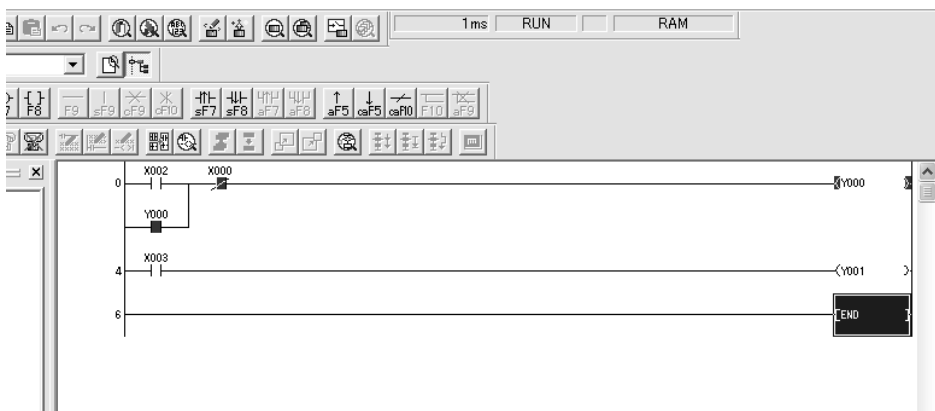
- ① 将PLC上的[RUN/STOP] 开关设置在“RUN”位置上。



- ② 选择工具栏中的 ，或者，在菜单栏中选择[在线]→[监视]→[监视模式]。

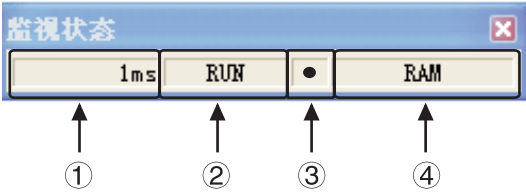
② 点击

用梯形图监视器进行动作确认。



- ① 在[开关X000 “OFF”]的状态下，如果[开关X002设置为“ON”]，则可以确认[开关Y000 “ON”]。
- ② 确认[开关X002 “OFF”]时，[输出Y000为“ON”]。
- ③ 确认[开关X000变为“ON”]时，[输出Y000变为“OFF”]。
- ④ 确认[输出Y001的“ON/OFF”]与[开关X003的“ON/OFF”]连动。

(1) 显示出监视状态对话框



- ① 扫描时间
显示顺控程序的最大扫描时间。
- ② PLC的状态。
显示PLC的状态。
- ③ 执行监视状态
正在执行监视中时闪烁。
- ④ 显示存储器内容
显示PLC的存储器内容。

(2) 怎样看梯形图监视器状态的显示内容

① 触点指令

输入触点 种类	X0: OFF	X0: ON
a触点	X000 — — 梯形图非导通	X000 —■— 梯形图导通
b触点	X000 —■— 梯形图导通	X000 — — 梯形图非导通

② 输出指令

驱动状态 种类	非执行·非驱动时	执行·驱动时
—()— OUT指令	—(Y000)—	—■Y000■—
—[]— SET指令等	—[SET M0]—	—■SET M0■—

RST指令用于监视显示复位软元件的ON/OFF状态。

软元件状态 种类	复位软元件 OFF时	复位软元件 ON时
—[]— RST指令	—■RST M0■—	—[RST M0]—

附录1.5 梯形图的编辑

附录1.5.1 梯形图的修正

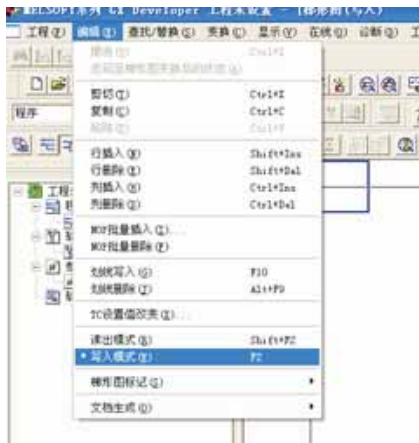
要点

- 要进行梯形图修正时，必须先设置在“写入模式”。

从工具栏中选择。



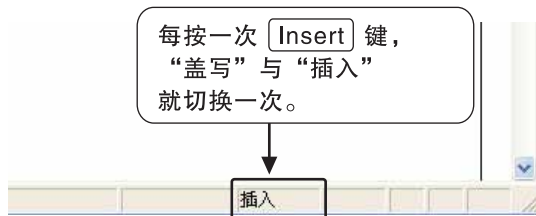
从菜单中选择（[编辑]→[写入]）。



- 输入字符时要全部采用半角字符进行输入，不能采用全角字符。

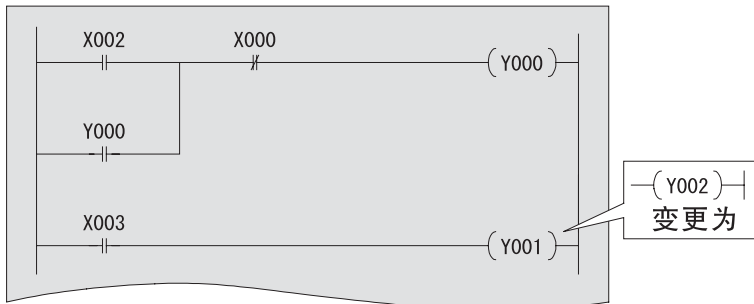
- “盖写”与“插入”的切换

- 要对现有的梯形图进行修正时，必须先设置在“盖写模式”。
- 如果设置为“插入”模式，可以增加插入另一个梯形图。

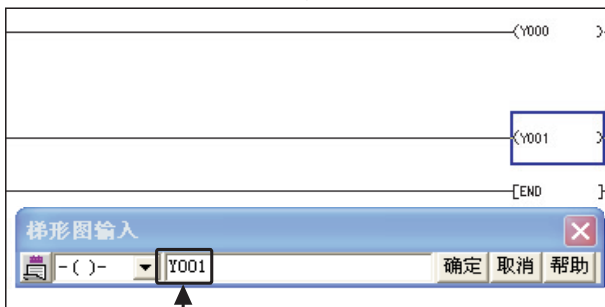
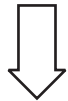


① 变更OUT 线圈和触点的编号

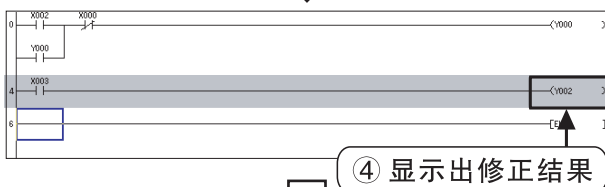
[要修正的梯形图]



① 双击要修正的部分。

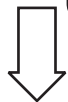


② 由“Y001”变更为“Y002”。

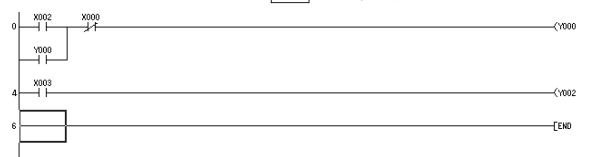


③ 用 Enter 键或者 [OK] 键确定。

④ 显示出修正结果，灰色部分即该梯形图块。



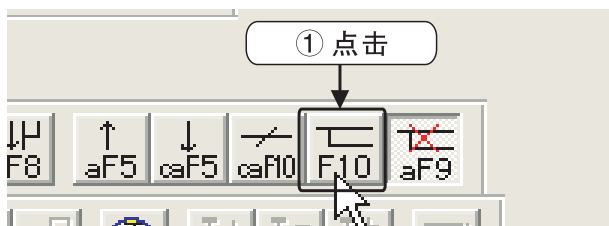
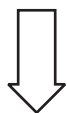
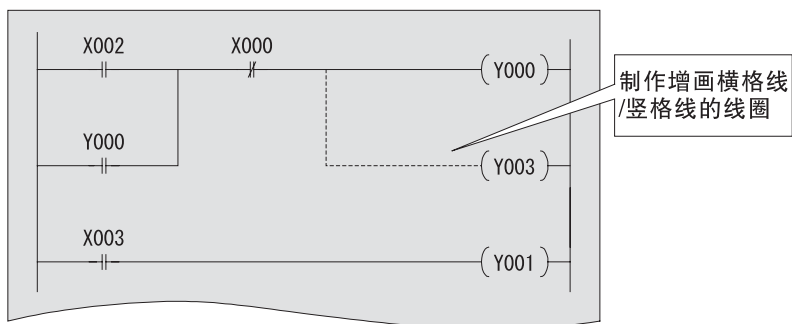
按 F4 (转换)



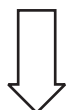
⑤ 按 F4 (转换) 键，确定变更内容。

② 增画格线

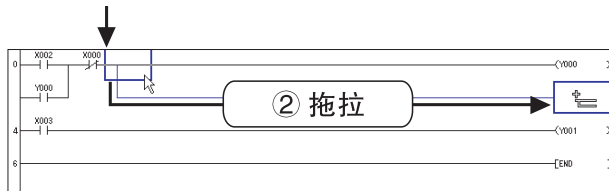
[为梯形图增画格线]



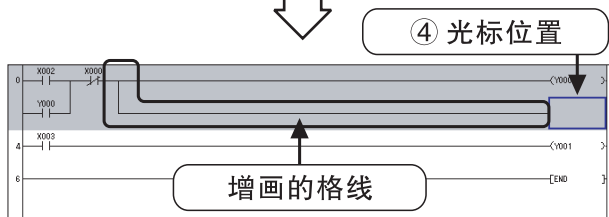
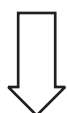
① 点击工具栏的 (F10)。



使光标停在竖格线
开始位置的右上角

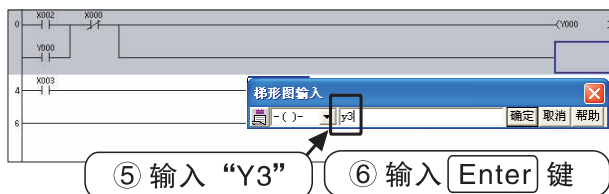
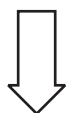


② 使光标停在欲增加竖格线的部分右上位置，拖拉至结束位置后放开。



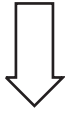
③ 一直到拖拉结束位置为止，显示出增画的竖格线。

④ 使光标停在欲增加OUT线圈的位置上，点击工具栏的 。

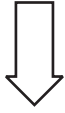


⑤ 输入“Y3”。

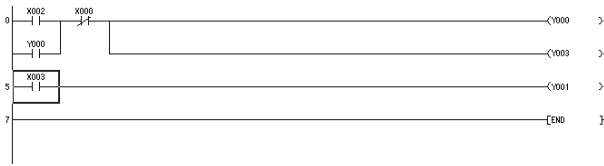
⑥ 用 键或者 [OK] 键确定




⑦ 增画的梯形图完成，增画的梯形图块显示为灰色。



F4 (转换)

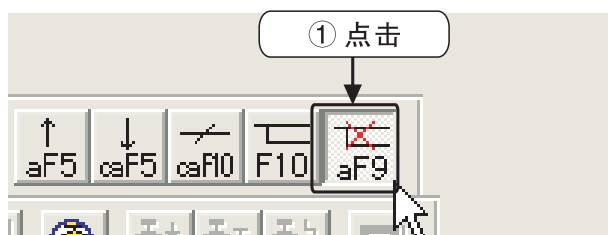
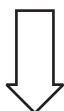
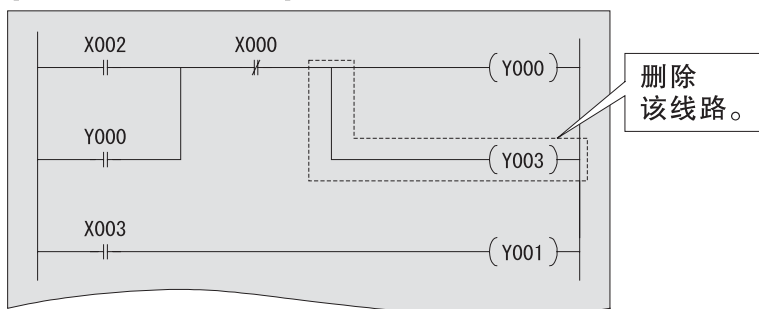



⑧ 按 **F4** (转换) 键，确定变更内容。

- 操作结束时，再一次点击工具栏的 。

③ 格线的删除

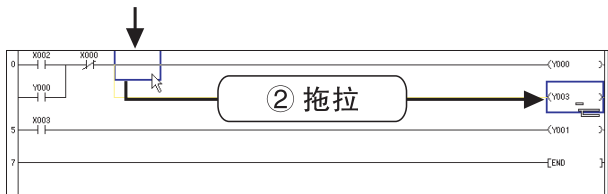
[要删除格线的梯形图]



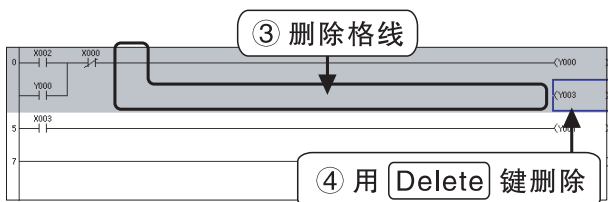
① 点击工具栏的  (Alt + F9)。



光标停在开始删除位置的右上角



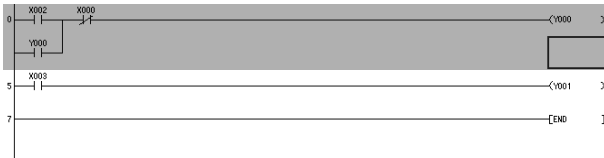
② 使光标停在欲删除部分右上位置，拖拉至结束位置后放开。



③ 格线即被删除。

④ 用 Delete 键删除OUT线圈。

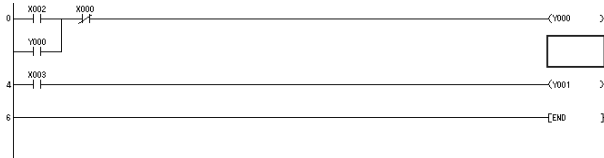





⑤ 删除的线路图块显示为灰色。



F4 (转换)

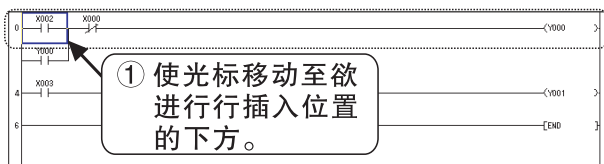
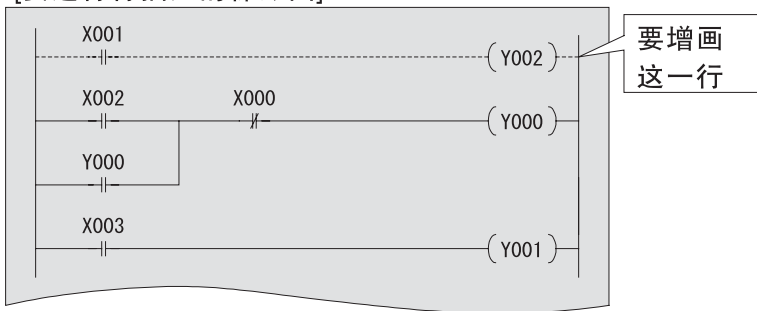


⑥ 按 **F4** (转换) 键设置变更内容。
 • 操作结束时，再一次点击工具栏的 。

附录1.5.2 行的插入和删除

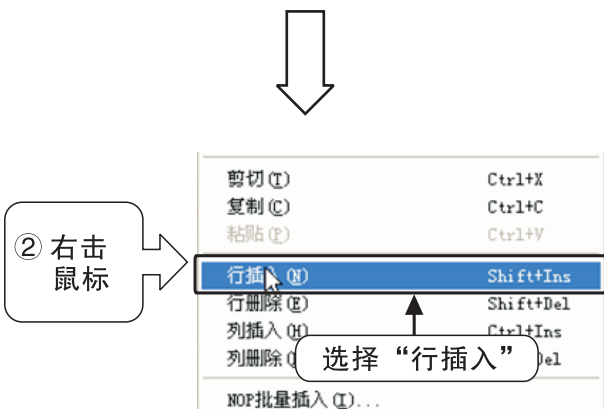
① 行的插入

[要进行行插入的梯形图]

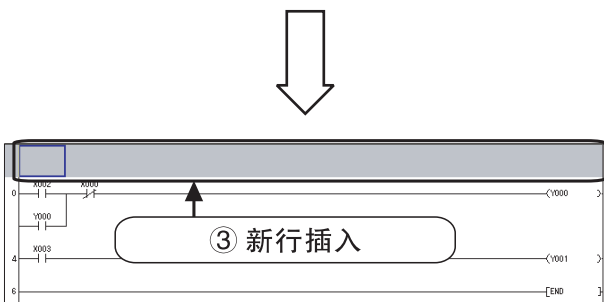


行被插入光标停留位置的上方。

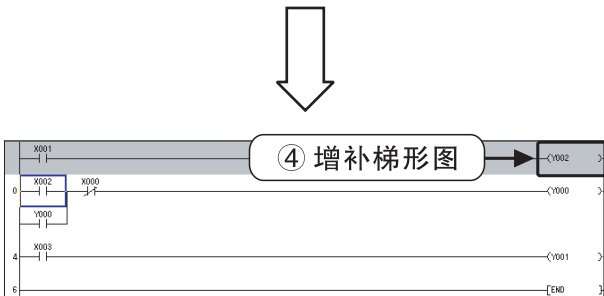
① 使光标停在欲进行行插入位置的下一行。



② 在任意位置上右击鼠标，选择[行插入]。



③ 行插入结束。



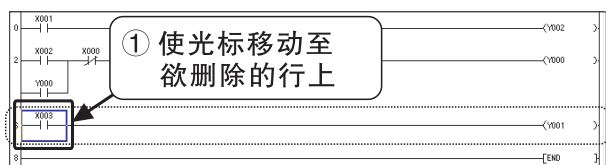
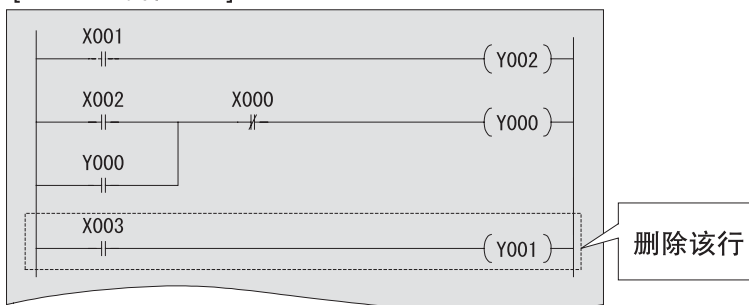
④ 将程序增补在插入行位置。



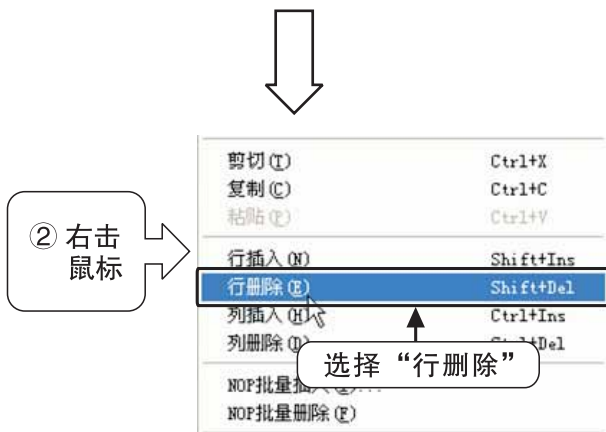
⑤ 按 **转换** (F4) 键，确定变更内容。

② 行的删除

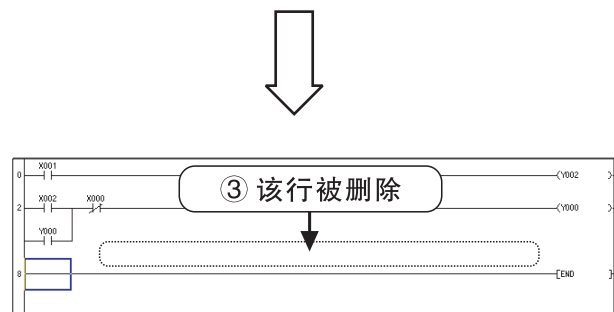
[要删除的梯形图]



① 使光标移动至欲删除的行上。



② 在任意位置右击鼠标，选择[删除行]。



③ 行被删除。

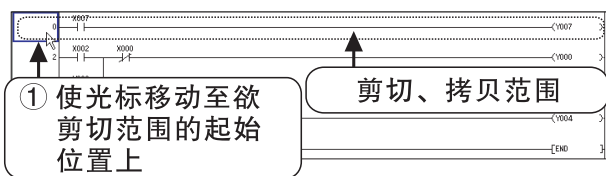
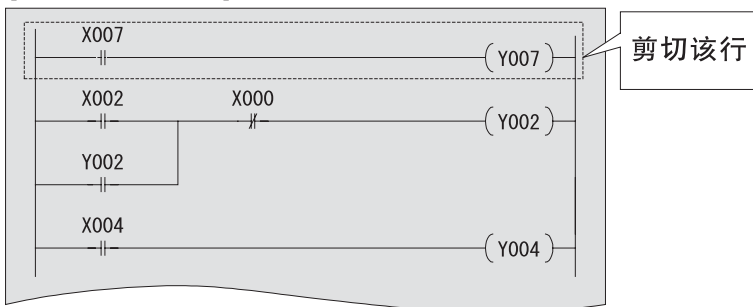
要点

虽然行被删除的位置并不显示为灰色，但是，仍要按 **转换** (F4) 键，确定变更内容。

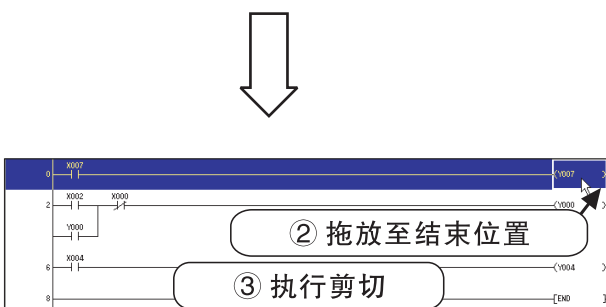
附录1.5.3 梯形图的剪切和拷贝（粘贴）

① 剪切


[要编辑的梯形图]

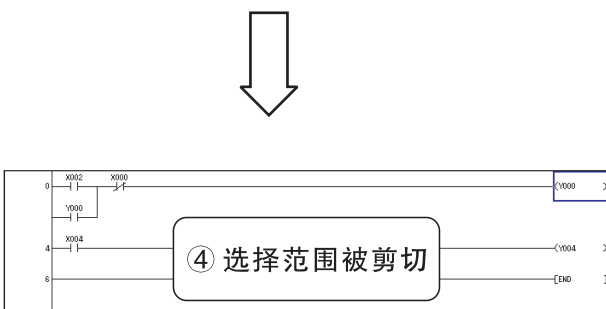


① 使光标移动至欲剪切梯形图的起始位置上。



② 拖放至结束位置。

③ 在工具栏中选择 、或者在菜单栏中选择“[编辑]→[剪切] (Ctrl + X)”，执行剪切。

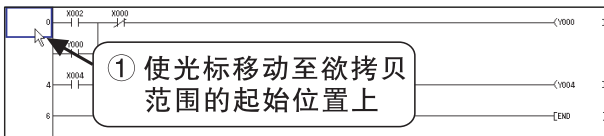
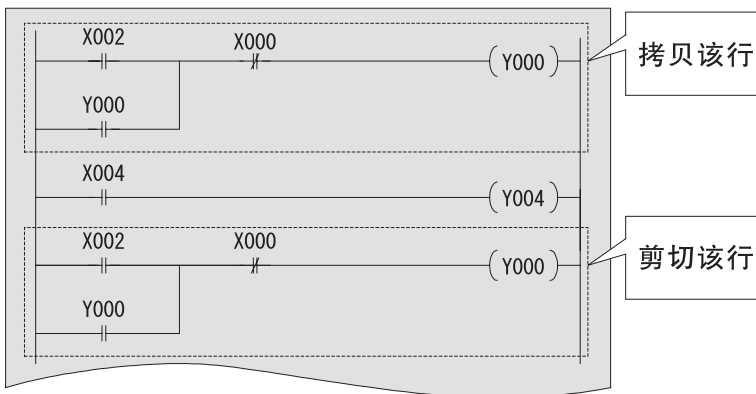


④ 选择范围被剪切

部分梯形图被剪切时，会留下灰色显示。在修正梯形图后，按 **F4**（转换）键，确定变更内容。

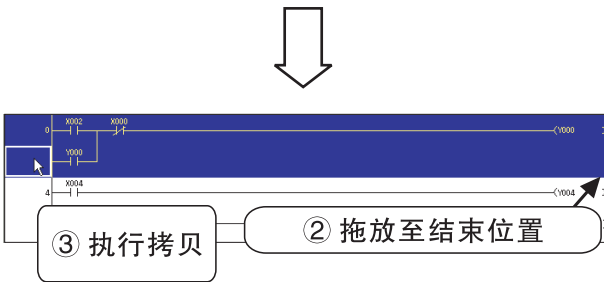
② 复制（粘贴）

[要复制（粘贴）的回路]



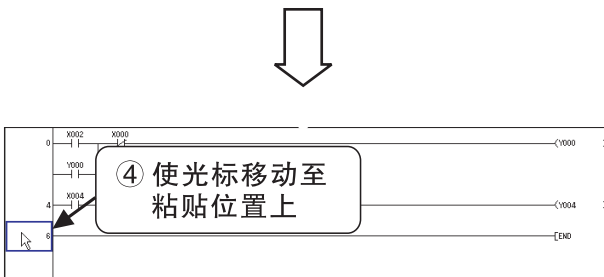
继续编辑上一节中“剪切”的梯形图。

① 使光标移动至欲拷贝梯形图的起始位置上。



② 拖放至结束位置。

③ 在工具栏中选择 、或者在菜单栏中选择“[编辑]→[拷贝] (Ctrl + C)”。



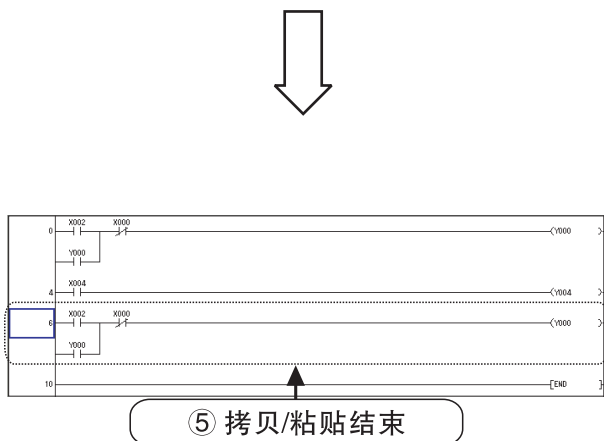
④ 使光标移动至欲粘贴位置上。

要点

用 **Insert** 键切换：

“盖写”模式：用盖写方法粘贴在光标位置上。

“插入”模式：插入至光标位置上侧。



⑤ 在工具栏中选择 、或者在菜单栏中选择“[编辑]→[粘贴] (Ctrl + V)”。


粘贴部分梯形图时，会留下灰色显示。在修正梯形图后，按 **F4** (转换) 键，确定变更内容。

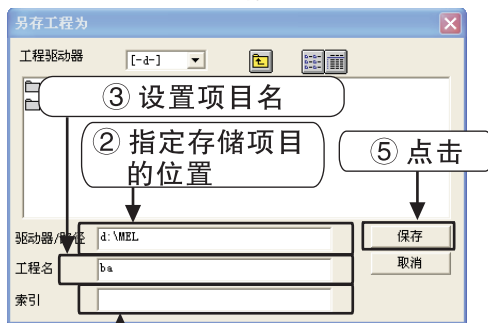
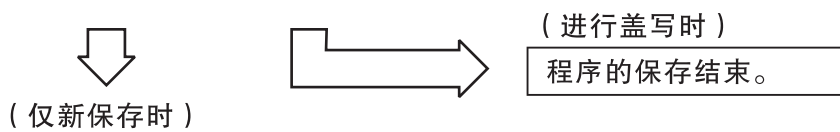
附录1.6 制作成的梯形图的保存

附录1.6.1 新保存·盖写保存

要 点
程序中如果有未转换的梯形图，进行转换 [F4] 键操作结束后，要进行保存操作。

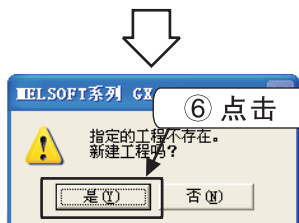


- ① 在工具栏中选择 、或者在菜单栏中选择“[项目]→[项目的盖写保存] (**Ctrl** + **S**)”。



- ② 指定存储项目的位置。
 ③ 设置项目名。
 ④ 可以设置显示程序内容等的关键字 (任意)。
 ⑤ 点击 **保存** 。

④ 设置关键字 (任意)



- ⑥ 点击确认对话框中的 **是 (Y)** ，结束保存。

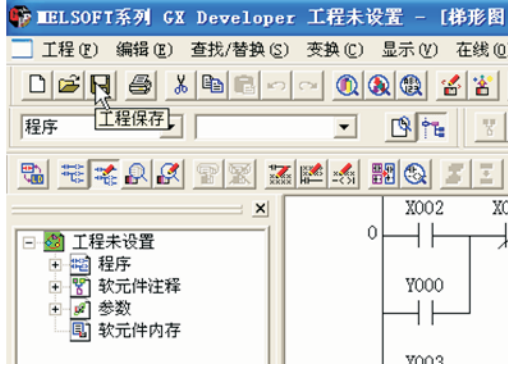
保存在软盘中时，如果容量不足，可暂时保存在硬盘中，然后将项目移存到别的软盘中。

参 数
<ul style="list-style-type: none"> ● 项目名中不可使用下列字符： /、¥、>、<、*、?、”、“、 、:、; (; 和 ¥ 仅可用于指定驱动器)。 另外，项目名的最后不可使用句号。 ● 如果用GX Developer (SW6D5-GPPW以上版本) 设置字符数 ≥ 8 的项目名， 而用GX Developer (SW6D5-GPPW以下版本) 进行读出，则第8个字符起无显示。 ● 项目路径+项目名的字符数不得超过半角150个字符 (全角75个字符)。 ● 关键字的字符数不得超过半角32个字符 (全角16个字符) ● 项目路径和项目名中包含有空位时，即使在EXPLORER画面上双击GPPW.gpj,**gps文件， 也无法正常起动GX Developer。 项目路径和项目名中包含有空位时，可在起动GX Developer后进行“[项目]→[打开项目]” 的菜单点击操作，打开项目。

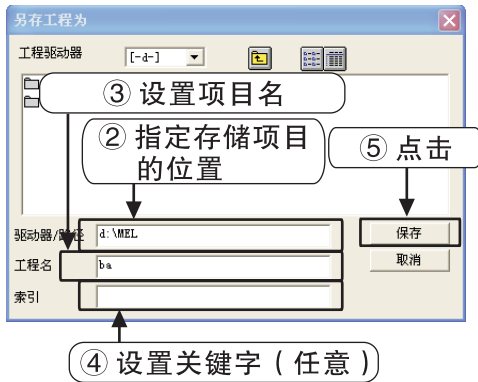
附录1.6.2 给项目命名后保存

要点

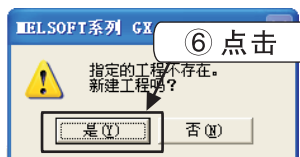
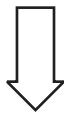
程序中如果有未转换的梯形图，进行转换 **[F4]** 键操作结束后，要进行保存操作。



- ① 在菜单栏中选择“[项目]→[给项目命名后保存] (Ctrl + S)”。



- ② 指定存储项目的位置。
- ③ 设置项目名。
- ④ 可以设置显示程序内容等的关键字（任意）。
- ⑤ 点击 **保存**。



- ⑥ 点击确认对话框中的 **是(Y)**，结束保存。

关于驱动器/路径名和项目名的使用，请参见上一頁的内容。

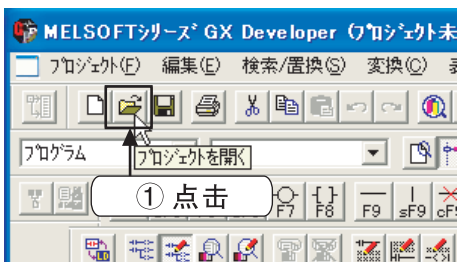
保存在软盘中时，如果容量不足，可暂时保存在硬盘中，然后将项目移存到别的软盘中。


附录1.6.3 项目的读出

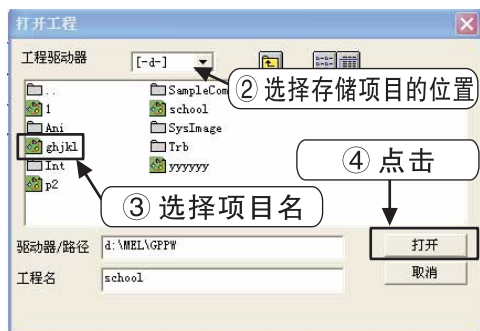
参 考

进行读出操作时，如果有另一个项目处于打开状态，则该项目将被关闭。

如果项目中有尚未转换的梯形图，或者未作保存，则将显示出相应的警告信息。



- ① 在工具栏中选择 、或者在菜单栏中选择“[项目]→[打开项目] (Ctrl + O)”。



- ② 选择存储项目的位置。
- ③ 选择要读出的项目。
- ④ 点击 **打开**，读出项目。


附录1.7 为程序调试所必需的操作

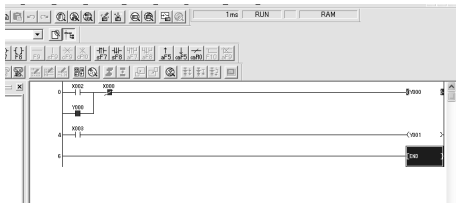
关于与PLC的连接以及程序的写入方面的说明，请参见“3.4 将程序写入PLC中”。


附录1.7.1 梯形图的监视

在显示梯形图的同时监视触点的导通状态和线圈的驱动状态。

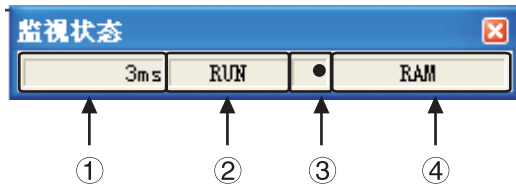


- ① 在工具栏中选择 、或者在菜单栏中选择“[在线]→[监视]→[监视模式]”。



- ② 梯形图监视窗口中显示出梯形图的ON/OFF状态和字软元件（定时器、计数器、数据寄存器）的当前值。
- ③ 用结束梯形图监视时，可在窗口上右击鼠标，选择[停止监视]。
- ④ 为了进行程序的修正和写入，在工具栏中选择 、或者在菜单栏中选择“[编辑]→[写入模式]”。

(1) 显示监视状态对话



- ① 扫描时间
显示顺控程序的最大扫描时间。
- ② PLC的状态
显示PLC的状态。
- ③ 执行监视状态
正在执行监视中时闪烁。
- ④ 显示存储器的内容
显示PLC的存储器的内容。

(2) 怎样看梯形图监视状态的显示内容

① 触点指令

输入触点 种类	X0: OFF	X0: ON
a触点	X000 — — 梯形图非导通	X000 —■— 梯形图导通
b触点	X000 —■— 梯形图导通	X000 — — 梯形图非导通

② 输出指令

驱动状态 种类	非执行·非驱动时	执行·驱动时
—()— OUT指令	—(Y000)—	—■Y000■—
—[]— SET指令等	—[SET M0]—	—■SET M0■—

RST指令用于监视显示复位软元件的ON/OFF状态。

软元件状态 种类	复位软元件 OFF时	复位软元件 ON时
—[]— RST指令	—■RST M0■—	—[RST M0]—

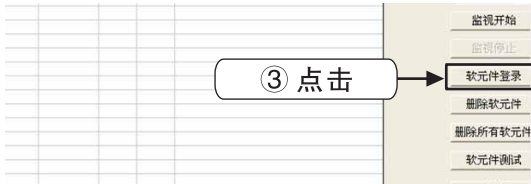
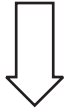
附录1.7.2 软元件登录的监视

① 任意软元件的登录

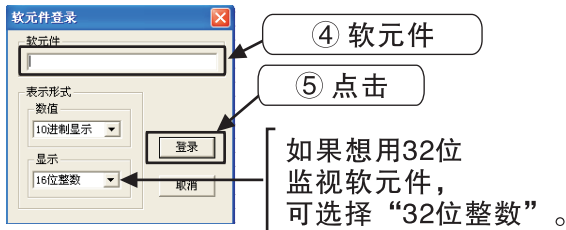


① 设置为梯形图监视状态（参见附录1.7.1）。

② 在菜单栏中选择“[监视]→[软元件登录]”，或者在梯形图窗口上右击鼠标，选择[软元登录]。

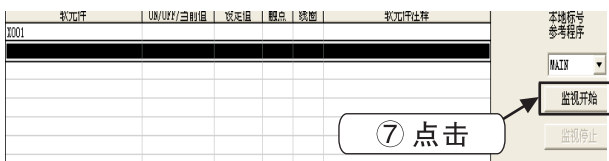


③ 点击[软元件登录监视]窗口中的[软元登录]。



④ 将软元件编号输入软元件登录窗口中。

⑤ 点击[登录]。

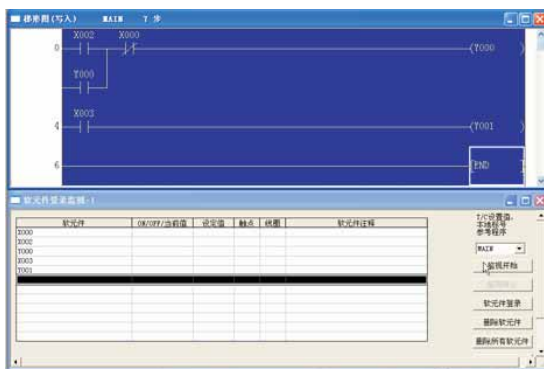
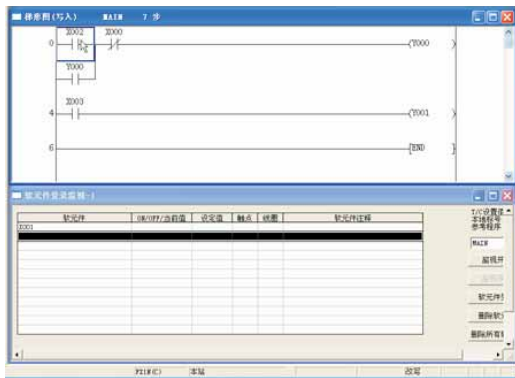
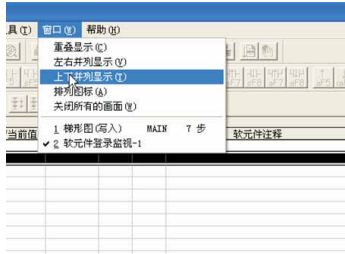


⑥ 软元件被登录到监视窗口中。

⑦ 点击[监视开始]，就会根据软元件的动作显示软元件值的内容和触点以及线圈的ON/OFF状态。

② 登录正在梯形图监视显示中的软元件

在梯形图监视窗口中指定梯形图的范围，对该部分中的软元件进行成批软元件登录。



① 设置为梯形图监视状态（参见附录1.7.1）。

② 在菜单栏中选择“[监视]→[软元件登录]”，或者在梯形图窗口上右击鼠标，选择[软元件登录]（参见上一页的内容）。

③ 在菜单栏中选择“[窗口]→[上下并列显示]”，并列显示出“梯形图窗口”和“软元件登录监视窗口”（“软元件登录监视窗口”要设置在停止监视状态）。

④ 并列显示出“梯形图窗口”和“软元件登录监视窗口”。

⑤ 点击梯形图的起始点。

⑥ 在按住[Shift]键的同时，点击终点，选择范围。

⑦ 用鼠标将所选择的范围拖拉到“软元件登录监视窗口”上。

⑧ 软元件被登录到监视窗口中。

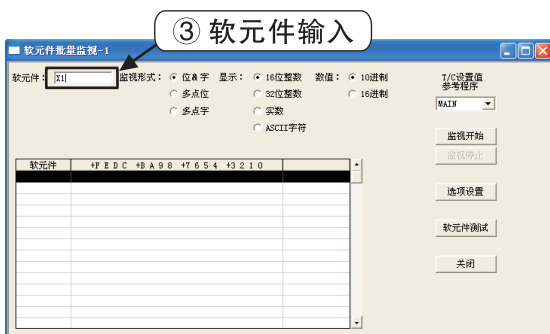
⑨ 点击[监视开始]，就会根据软元件的动作显示软元件值的内容和触点以及线圈的ON/OFF状态。

附录1.7.3 软元件的成批监视

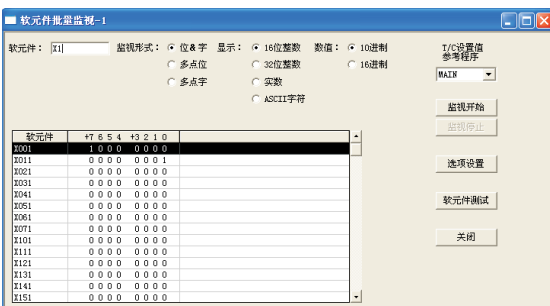
指定起始软元件编号，成批监视从起始号起的连号软元件。

① 设置为梯形图监视状态（参见附录1.7.1）。

② 在菜单栏中选择“[监视]→[软元件批量]”，或者在梯形图窗口上右击鼠标，选择[软元件成批]。



③ 将要监视的软元件的起始编号输入到“软元件成批监视”窗口中，按 **Enter** 键，或者点击[监视开始]。



④ 根据软元件的动作显示软元件值的内容和触点以及线圈的ON/OFF状态。

附录1.7.4 软元件试验

① 强制ON/OFF

使PLC的位软元件（M、Y、T、C等）强制性地ON/OFF（不能使用X的强制ON/OFF）。PLC正在运行时，仅可以进行一个运算周期的ON/OFF动作，以顺控程序驱动的动作作为优先。进行输出确认等时，要先将PLC设置在STOP状态。



① 设置为梯形图监视状态（参见附录1.7.1）。

② 在菜单栏中选择“[监视]→[调试]→[软元件测试]”，或者在梯形图窗口上右击鼠标，选择[软元件测试]。



③ 输入强制ON/OFF的软元件编号。

- ④ • [强制ON]：使软元件变为ON。
- [强制OFF]：使软元件变为OFF。
- [强制ON/OFF反转]：每按一次反复进行软元件的ON/OFF。

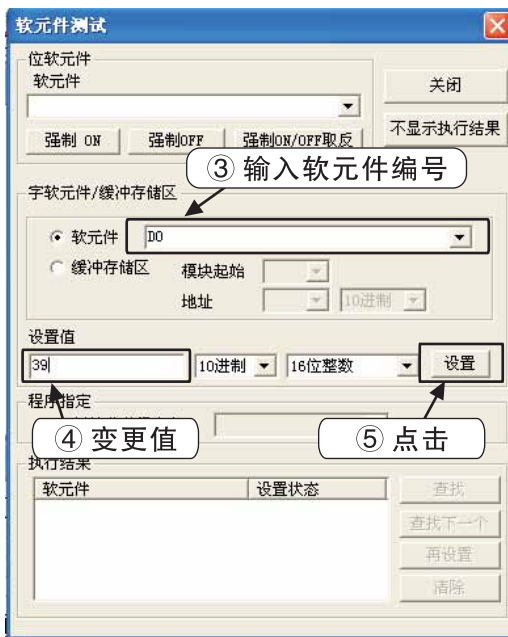
参 考

强制ON/OFF（梯形图监视窗口）

在按住[Shift]的同时，双击“梯形图监视窗口”上的任意位ICE（触点、线圈），就能够使指定软元件强制性ON/OFF。

② 变更字软元件的当前值

使PLC的字软元件（T、C、D等）当前值变更为指定值。



① 设置为梯形图监视状态（参见附录1.7.1）。

② 在菜单栏中选择“[监视]→[调试]→[软元件测试]”，
或者在梯形图窗口上右击鼠标，选择[软元件测试]。

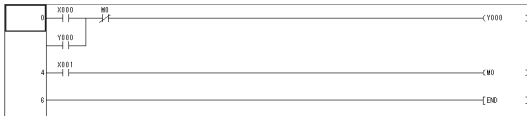
③ 输入要变更的软元件编号。


④ 输入变更值。

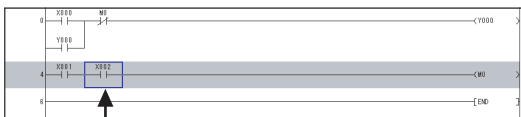
⑤ 点击 [设置] 按钮。

附录1.7.5 运行中写入程序

在PLC运行中时，仅将经过修正的梯形图部分写入PLC中。
因为不需要传输整个程序，程序的写入能够在短时间内完成。




① 以在左图所示的梯形图上增加触点为例来进行说明。
显示出梯形图，设置为写入模式 ()。

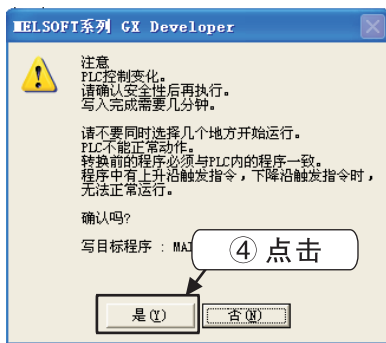


② 增加触点

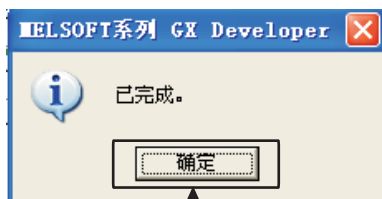
② 增加触点。
梯形图块显示为灰色。

运行中写入  [Shift] + [F4]

③ [Shift] + [F4] 或者，在菜单栏中选择 “[变换] → [变换 (运行中写入)]”。



④ PLC的控制因程序变更而产生变化，要注意确认无安全方面的问题等的提醒信息内容，点击 [是] 按钮。



⑤ 点击

⑤ 会显示出“运行中写入处理已结束”字样的信息，点击 [OK] 按钮。

注意

如果PLC中修正前的程序与GX Developer中的程序不一致，则不能进行写入。如果事前不知道是否一致，应该事前进行核对，也可以用[PC写入]作成批传输。

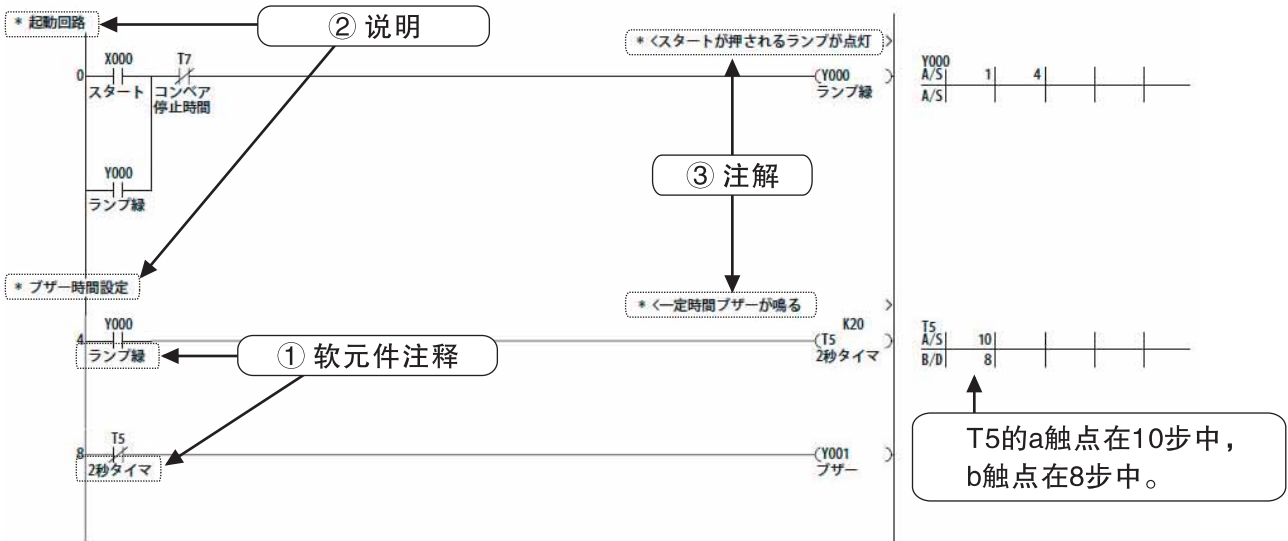
附录1.8 注释的输入

附录1.8.1 注释的种类

可以输入下表中列出的三种注释：

种类	目的	字符数（全角）	备考
① 软元件注释	表示每一个软元件的作用和用途。	16	对PLC进行写入时，需要参数的“注释容量设置”。另外，还需要写入的“注释范围设置”。
② 说明	表示对梯形图块的作用和用途。	32	仅GX Developer方面的注释（外围）（不进入PLC中）。
③ 注解	表示对输出指令的作用和用途。	16	仅GX Developer方面的注释（外围）（不进入PLC中）。

[注释例]



要点

显示注释的方法

- 在菜单中选择“[显示]→[显示注释]”，显示注释。
- 要中止注释显示时，可再次进行上述操作。

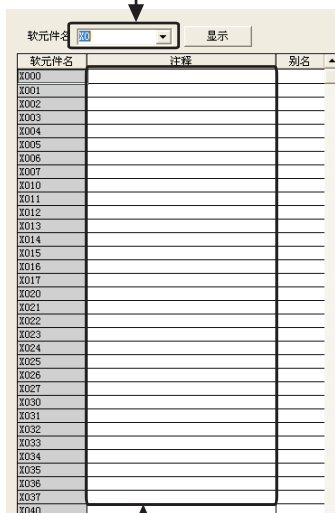
附录1.8.2 制作软元件注释的操作

① 用一览表进行输入的方法



- ① 在项目一览表中进行“[软元件注释]→[COMMENT]”菜单点击操作。

② 输入软元件编号



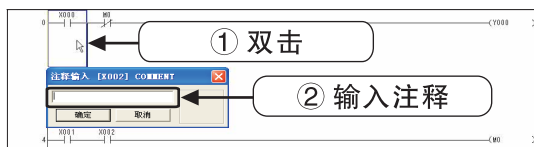
- ② 将起始编号输入到要制作注释的软元件的"软元件名"中。


- ③ 将注释输入到“注释”串中。

- 输入其它软元件的注释时，用“②”的操作方法再次输入软元件编号。


③ 输入注释

② 用梯形图进行输入的方法



- ① 单击菜单栏中的 ，双击输入注释的梯形图符号。

- ② 将注释输入到“输入注释”窗口中。

- 要结束操作时，可再一次单击工具栏中的 。

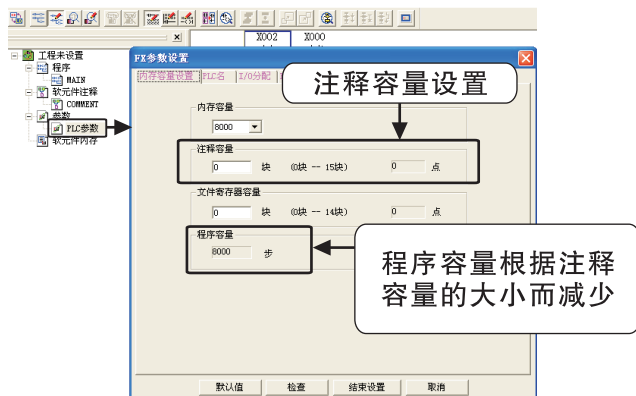
要点

将软元件注释写入PLC中的设置

为了将软元件注释写入PLC中，必须进行“参数设置”和“注释范围设置”。

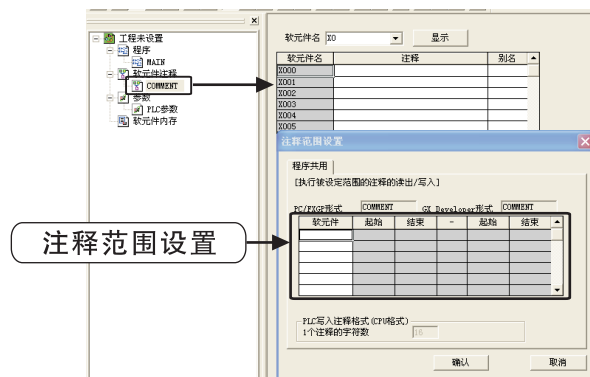
① 设置参数

- 进行“[参数]→[PC参数]”选择。
 - 在“注释容量”栏上设置“块数”。
- 在程序存储器中占用500步空间，相当于每块50点的注释。



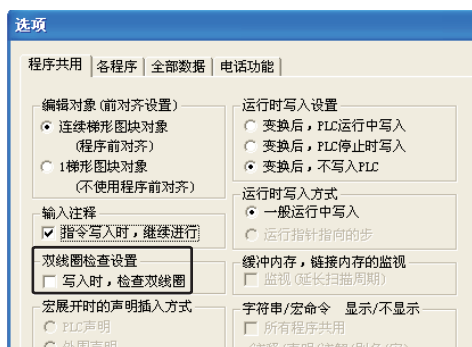
② 设置注释范围

- 进行“[软元件注释]→[COMMENT]”选择，显示出注释输入画面。
- 在菜单上进行“[编辑]→[注释范围设置]”选择。
- 在注释范围设置对话框中设置要写入PLC中的软元件的种类和范围。



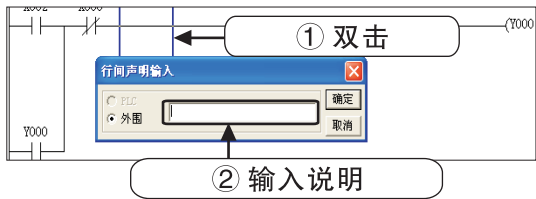
参考


制作梯形图时输入注释的方法




在菜单上进行“[工具]→[选项]”选择，点击“程序共用”标签，在[注释输入]栏中选择“写入指令时，继续进行”（打勾）。进行该项设置后，进行制作梯形图时的梯形图输入操作，即显示出上述②的“输入注释”窗口。

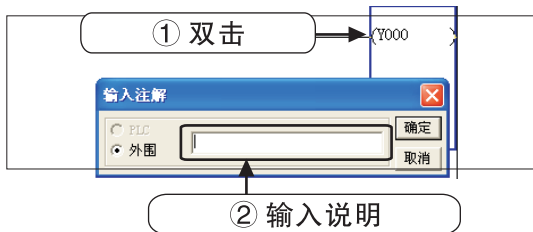
附录1.8.3 制作说明的操作步骤





- ① 单击菜单栏中的 ，双击输入说明的梯形图块中的任意部分。
- ② 将说明输入到“行间说明输入”窗口中，点击[OK]按钮。

• 要结束操作时，可再一次单击工具栏中的 。

附录1.8.4 制作注解的操作步骤



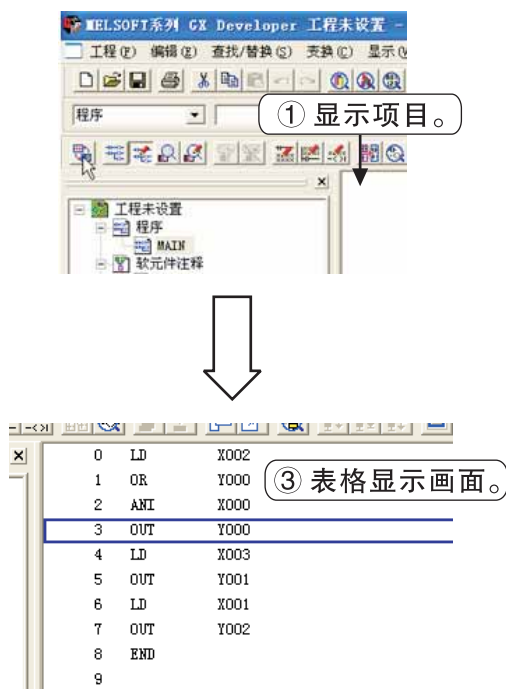
- ① 单击菜单栏中的 ，双击输入注解的输出指令符号。
- ② 将注解输入到“注解输入”窗口中，点击 OK 按钮。

• 要结束操作时，可再一次单击工具栏中的 。

附录1.9 制作表格程序的操作步骤

可以用GX Developer制作表格方式的程序。


附录1.9.1 显示表格编辑画面



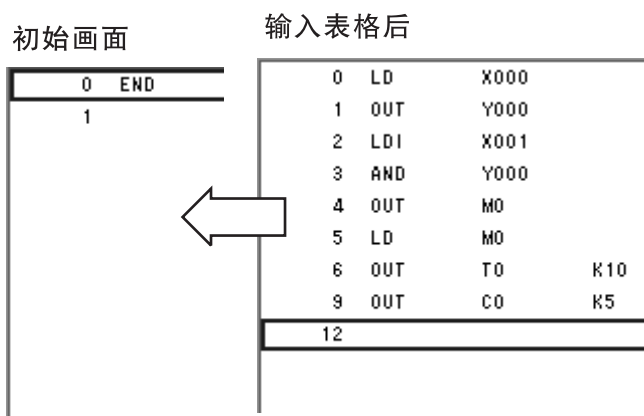
① 制作新项目（参见3.2.3）。或者，以梯形图形式显示现有的项目。

② 在工具栏中选择 ，或者在菜单上进行“[显示]→[表格显示]”选择操作。

③ 显示出表格编辑画面。

如果要返回梯形图显示，可再一次单击工具栏中的 ，也可以在菜单上进行“[显示]→[表格显示]”选择操作。

附录1.9.2 指令的输入方法

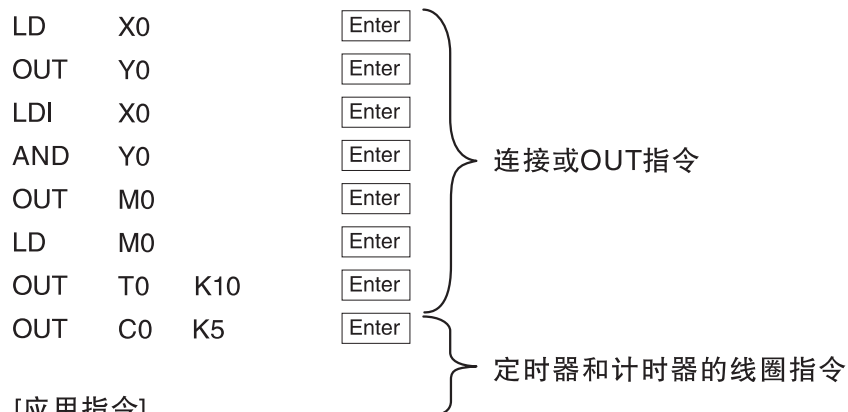


① 从0步起依次输入指令语言。每次进行输入时都自动增大显示的步号（输入方法请参见下一页的内容）。

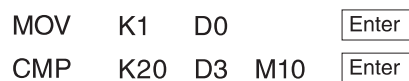
● 基本指令和应用指令的输入方法

用“空间”分段输入指令语言、软元件编号和操作数。

[基本指令举例]



[应用指令]



参 考

输入/编辑时的键操作

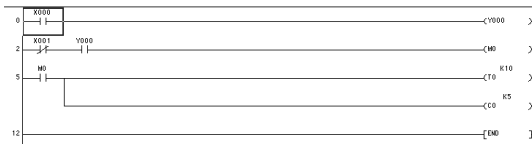
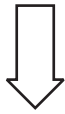
- 每按一次 键，[盖写]/[插入]模式就进行一次切换。
- 每按一次 键，就能够删除一个指令。
- 右击鼠标，即可进行“行插入”和“行删除”操作。

附录1.9.3 表格输入内容的确认

使表格输入的内容作梯形图显示，确认无出错。

0	LD	X000	
1	OUT	Y000	
2	LDI	X001	
3	AND	Y000	
4	OUT	M0	
5	LD	M0	
6	OUT	T0	K10
9	OUT	C0	K5
12	END		
13			

- ① 在工具栏中选择 ，或者在菜单上进行“[显示]→[梯形图显示]”选择操作。



- ② 确认表格输入的内容已经显示为梯形图。



- ③ 如果在菜单上进行“[工具]→[程序检查]”选择操作、进行程序选择，就可以确认有无出错、出错的是哪一步。

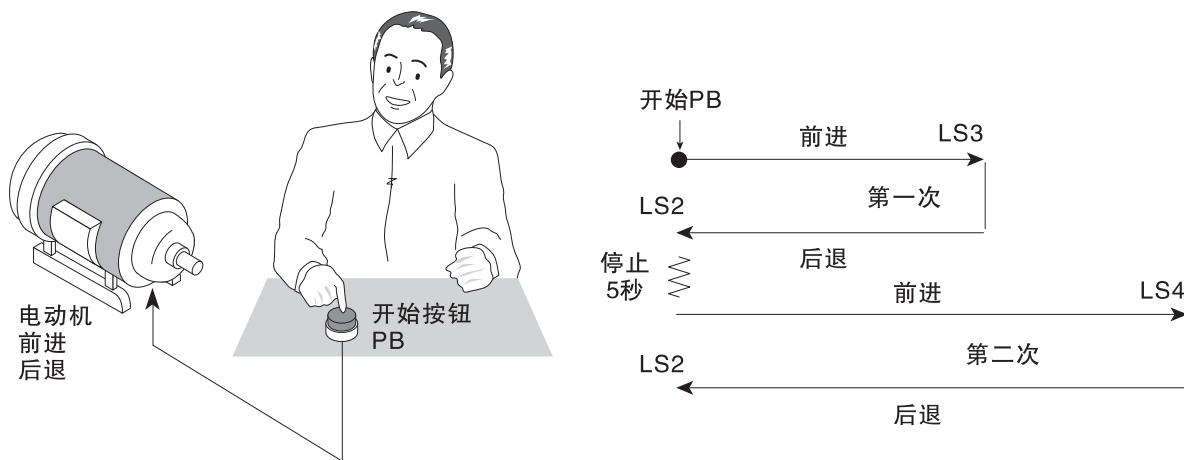
附录2 需要知道的SFC方式

附录2.1 需要知道的SFC方式 (序列 · 功能 · 图表 = 状态转移图)

在顺序中一般使用继电器顺序，但由于它是基于电气动作来表示的，因此很多情况下，在设计和维护中需要熟练的技术人员。解决它的方式是称作SFC的回路方式。SFC图基于时间的过去来表示机械的动作，也称作状态迁移图。即使不进行复杂的顺序设计，可编程控制器的程序也可简单实现。

《SFC方式的主要特点》

- ① 用工程步进动作来表现时刻变化的顺序。
- ② 无需互锁回路和双重输出等的回路设计。(可编程控制器自动处理)
- ③ 整体的控制内容和动作容易理解。
- ④ 利用动态监视功能，解决问题无需熟练技术人员。



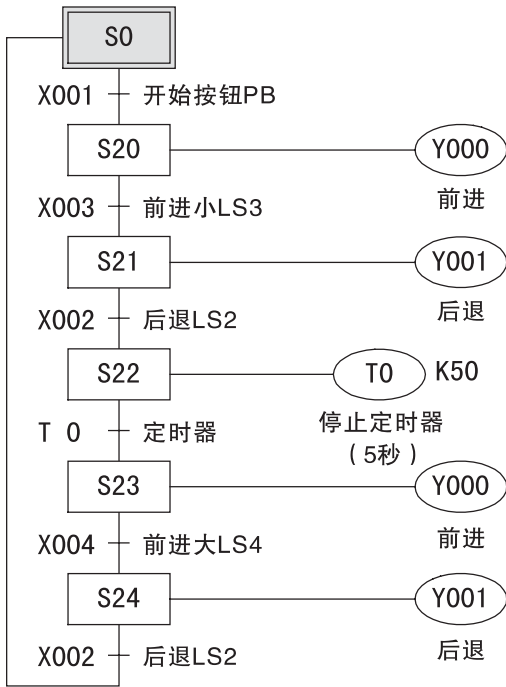
《动作》

- ① 若按下开始按钮PB，则台车前进，在限位开关LS3动作后立即后退。
- ② 若通过后退而使限位开关LS2动作，则停止5秒后继续前进，在限位开关LS4动作后立即后退。
- ③ 若最终限位开关LS2动作，则台车驱动电动机停止。

若用SFC来表示这样的动作，则如下。

此外，编程可使用方便的编程面板和个人计算机等进行。

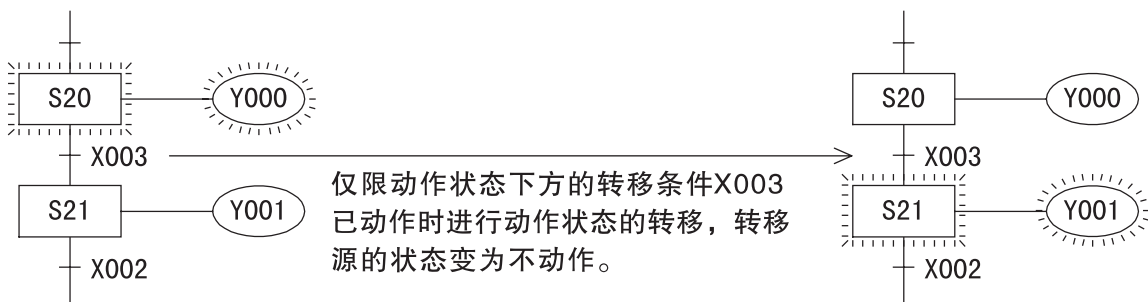
《微型可编程控制器的SFC图的例子》



- S0、S20…称作状态，在SFC中起着辅助继电器的作用。
- 最初的状态S0称作初始状态，因运行脉冲信号而变为ON。（若ON，则保持ON状态）
- 仅限下方的开始信号X001（转移条件）变为ON时，S20变为ON（保持）。（同时，原来的S0变为OFF·自动处理）其结果是，前进Y000变为ON。
- 接着若前进小X003变为ON，则S21变为ON，后退Y001变为ON。（同时S20、Y000变为OFF·自动处理）
- 之后同样地，若转移条件变为ON，则下一个状态分别变为ON。
- 程序可基于指令语，也可基于SFC图。

参 考

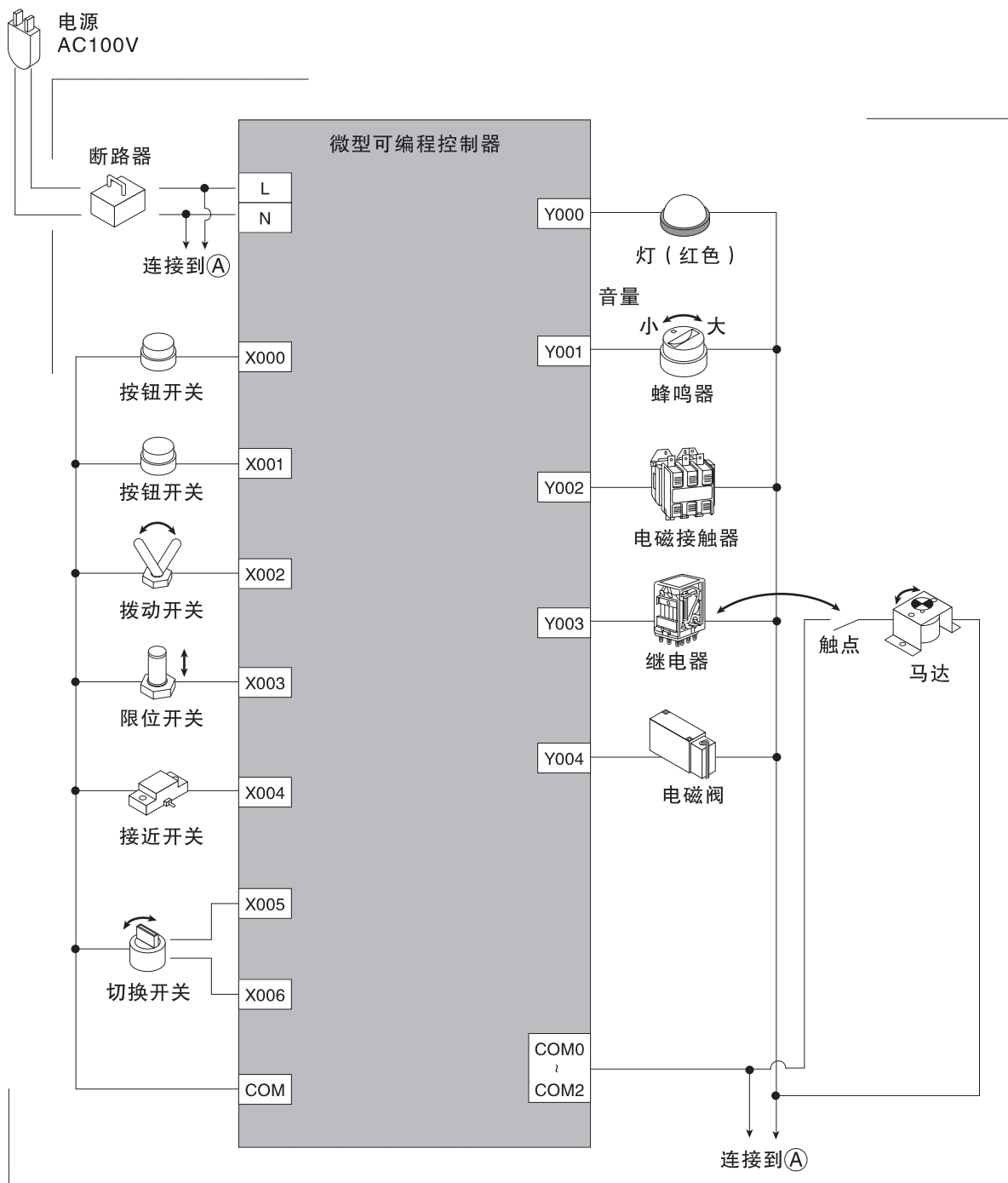
- 所谓初始状态…是指SFC图的开始状态。
- 所谓RUN脉冲…是指可编程控制器变为RUN状态时产生的信号。
- 所谓转移条件…是指状态之间所使用的转移信号。
- SFC的动作…如下。



备 忘 录

附录3 实习机的输入输出连接图

附录3.1 实习机的输入输出连接图



FX-I/O DEMO型实习机

手册修订历史

完成日期	副编号	内 容
1997年4月	D	<ul style="list-style-type: none"> · 将页的设置改为章、节 · P2-5 追加十进制数、八进制数、十六进制数说明 · P2-8 追加可编程控制器使用效果 · P3-14 追加自保持回路例 · 4-4节 追加抢答节目事例 · 5-1、5-2节 机种的维护实施 · 5-3节 外围设备的维护实施 · 5-4节 追加SFC介绍 · 删除学校介绍 · 附录1 追加演示机的输入输出连接图
1997年12月	E	<ul style="list-style-type: none"> · 追加“序言” · 2章 改变输入配线与可编程控制器的触点（a触点\leftrightarrowb触点） · 3章 将顺序例的输出Y1（蜂鸣器）变更为Y3（马达） · 5章 将“SFC图的例子”变更为“基于微型可编程控制器的SFC图的例子”
2002年3月	F	<ul style="list-style-type: none"> · 引入JIS电气用图符号 · 将FX0-14MR机种变更为FX1S-14MR · 5章 删除产品介绍
2006年9月	G	<ul style="list-style-type: none"> · 重新编辑所有章节的布置 · 追加GX Developer操作说明