

三菱 FX 系列顺序控制编程
MITSUBISHI FX PLC STEP CONTROL SYSTEM
PROGRAMMING COURSE

培
訓
教
程



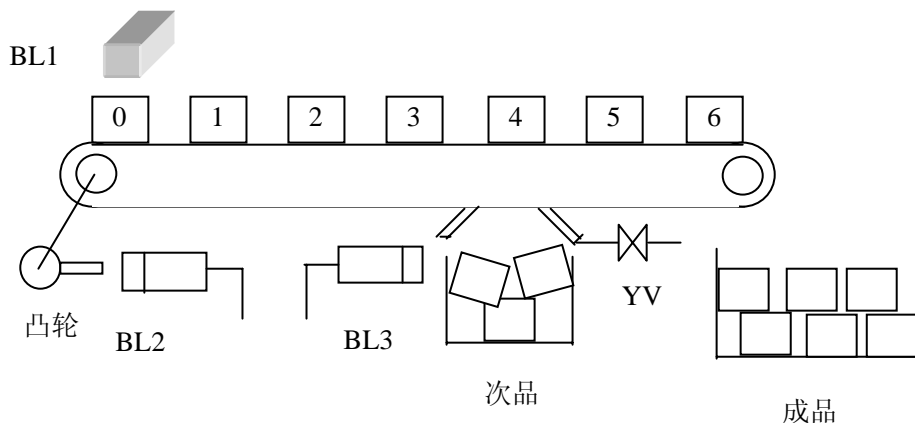
因为专业·所以优秀

第一讲 自动化概论

自动化是指：在没有人的直接参与下，机器设备或生产管理过程通过自动检测、信息处理、分析判断自动地实现预期的操作或某种过程。

下面请看一个实例：物流自动检测系统

- 1、BL1 为检测次品的传感器，当检测到次品后，会有一个 ON 信号输出
- 2、电机每转一圈，物品就右移一格，同时 BL2 检测到凸轮信号，并就发出一个脉冲
- 3、当次品移到第 4 位时，自动打开电磁阀 YV，使次品下落至次品箱里，这样就完成了挑选良品和次品的动作。BL3 为检测次品的数量



像这样一个自动检测，自动分析处理信号，并自动完成我们预期要求功能的系统，我们就称之为自动化控制系统。

在这个系统里，分别要完成三部分的工作

- 1、检测（控制系统输入）
- 2、分析处理（程序处理）
- 3、驱动执行机构（执行挑选良品和次品的动作）

其实，在所有的自动化控制系统里，无非也就是只有这三部分的工作。以人本身作为举例（人本身就是世界上最智能化的自动化系统）

A: 输入传感装置

用于将外界的信号输入到控制中心，由如接近开关、光电开关、温度传感器（相当人的眼睛，皮肤）

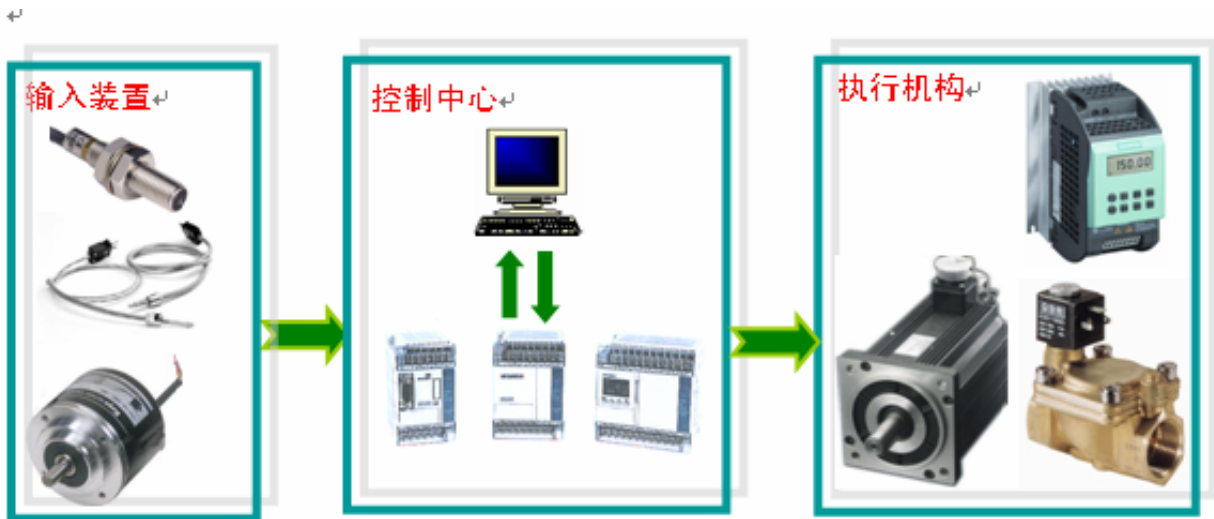
B: 中央处理单元

负责把从外界接收的信号经过处理运算之后，输出到执行机构（相当人的大脑）

C: 外部执行机构

将 CPU 输出的信号进行处理，完成特定的工作，例如加热器、电磁阀、定位装置等（相当人的手脚）

如下图：



而我们上面所看到的物流检测系统例子里，我们所看到的

BL1、BL2、BL3 为输入装置

传送带、马达、电磁阀为执行机构



噫~~，我们的控制中心呢？跑哪去了？别急，当然有，没有控制处理单元的系统是不叫自动化控制系统的，试问，没有大脑的人，能叫人吗？

一般控制系统都会安装在一个比较安全的单独的箱子里，一般我们叫这个箱子叫做控制柜，也叫电控柜。

哪~~，我们这个控制柜是怎么让这个系统动起来，怎么控制这个系统的呢？他们之间的关联又是什么呢？

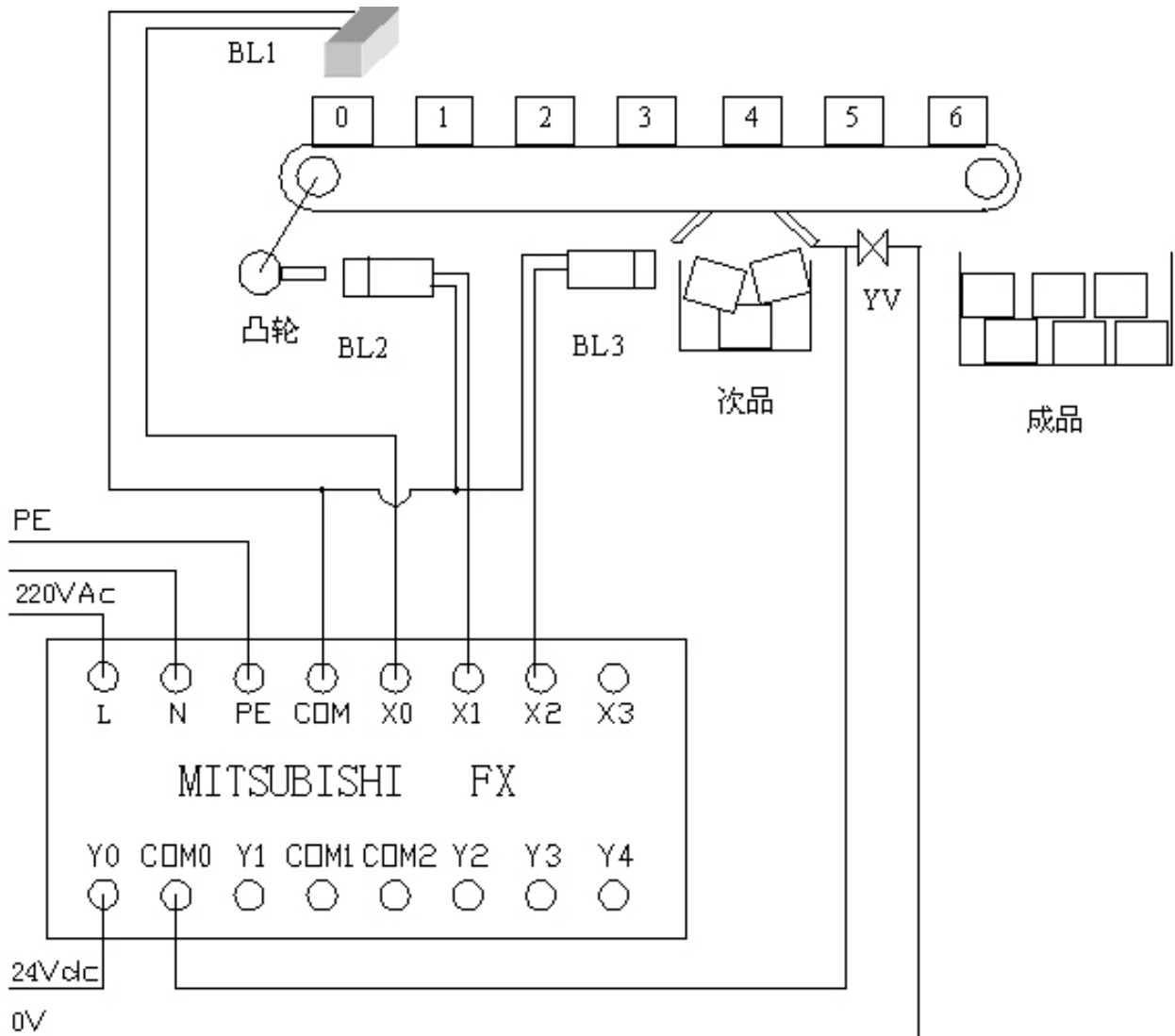
别急，请看第二讲，控制系统的硬件构成，即机械本体与控制中心的连接关系

因为专业·所以优秀

第二讲 控制系统的硬件构成 即机械本体与控制中心的连接关系

上一讲我们了解了整个自动化控制系统的构成，这一讲，我们就讲讲，这控制中心和机械本体他们的硬件连接关系又是怎么样。

我们知道，这机械本体上面的东西有 BL1、BL2、BL3(控制系统的输入)，传送带、马达、电磁阀(执行机构)。而这些东西和控制系统的通信连接关系又是怎么样呢，请看下图



因为专业·所以优秀

由上图，我们可以看到：

机械本体的输入(检测装置)和 PLC 的输入端连接(BL1 连到 X0，BL2 连到 X1，BL3 连到 X2)，这样，PLC 就可以实时地知道物流检测系统的当前状态，从而根据流水线的物品状态而做出相对应的控制处理。

同时，PLC 的弱电信号通过信号线，直接连接到执行机构(电磁阀 YV)，控制 YV 的开启/关闭。

而 PLC 又是怎么样实时地监控到机械本体的检测信号，同时又控制机械本体的执行机械(电磁阀 YV)的呢？

请学习第三讲，三菱 FX 系列 PLC 的硬件结构及选型



第三讲 三菱 FX 系列 PLC 的硬件结构及选型

请先直观认识一下三菱 FX 系列 PLC 的一些常用型号



FX1S



FX1N



FX2N



FX3U

下面是 FX2N 各部分的各部分指示

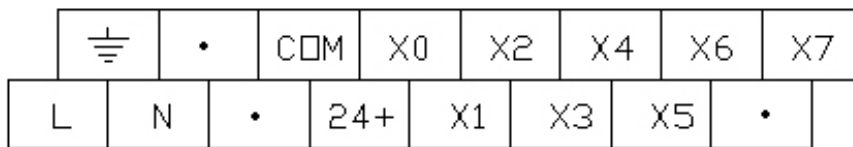


输入端子排



因为专业·所以优秀

输入端子排信号



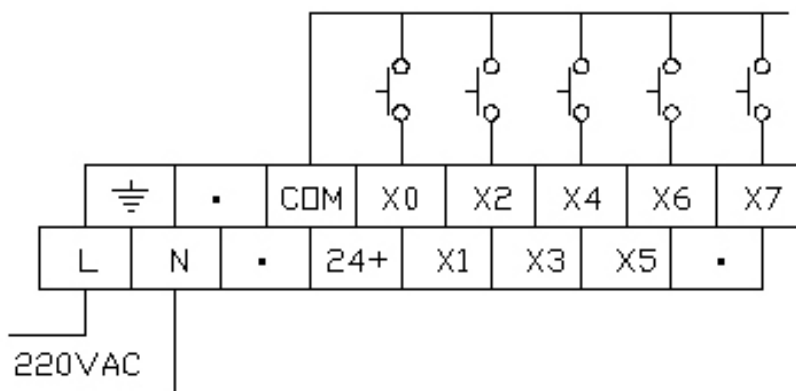
L: 输入电源火线

N: 输入电源零线

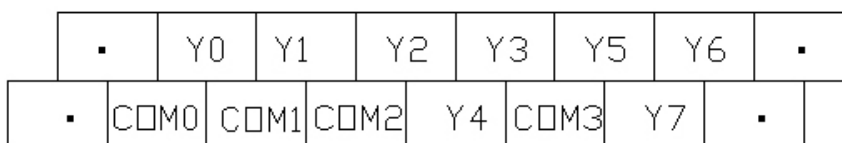
24+: PLC 自身 24V 电源输出正，可用于接小负载

COM: PLC 自身 24V 电源输出负 (0V)

X0-X7: PLC 输入信号端子，当 X 端子和 COM 短接时，视为接通（有信号输入），如下图



输出端子排信号



输出端子可以把它们看作是一组一组的开关，当某个输出点有输出时，相应的开关就会闭合，电路就会形成回路，从而就会驱动相应的负载（比如电磁阀，继电器等）。

例如下图，当 Y0 有输出时，PLC 内部触点闭合，电路导通，灯泡就亮。

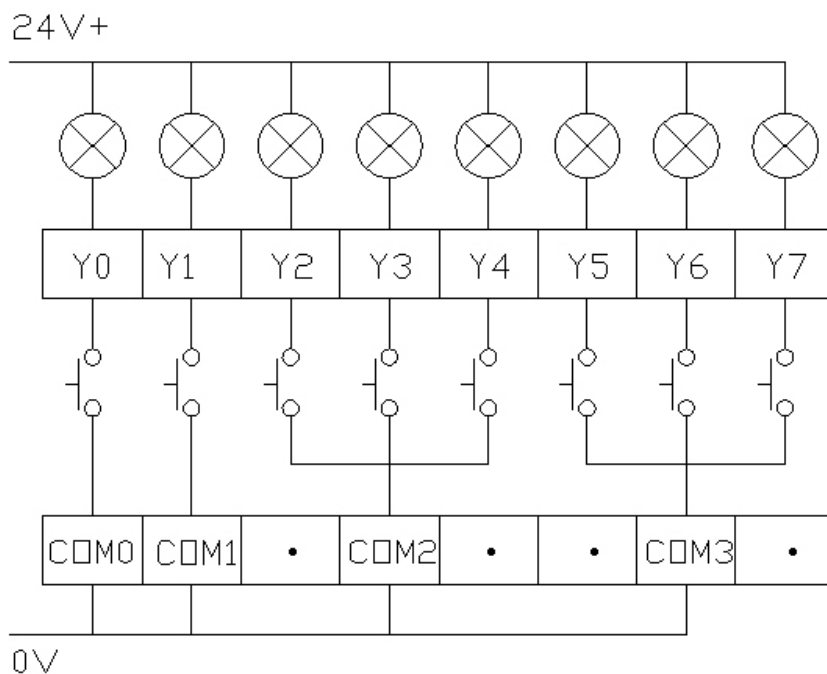
因为专业 · 所以优秀

注意：

三菱 PLC 的输出为无源输出触点，也就是说，如果没有外加电源，触点虽然闭合，一样也是没有电流流过。

请思考：

下图如果把 24V+ 去掉，当 Y0 有输出时，灯泡会不会亮？



上面我们了解了三菱 PLC 的硬件构成及输入输出端子排的定义及电气特性，下面我们就来总结一下 PLC 硬件选型时还应该注意哪些问题

1、电源：国内 PLC 的输入电源电压一般分为两种：直流 24V 和交流 220V，当然，国外还有一些输入电源电压标准，如 110VAC。

注意：在我们开始给 PLC 上电时，应该看清楚 PLC 的输入电源电压范围，以免把 PLC 烧掉。

2、输入 X 端子信号：常用的三菱 PLC 输入电信号为低压信号，一般为 X 和 COM 短接就视为接通，切记，不要输入交流高压信号，如果外部信号为高压信号的，应该通过中间继电器转换，或者转换成低压信号后再接入 PLC 的输入端子



3、输出：PLC 输出一般分为

1) 继电器输出

继电器类型输出的 PLC 优点是：

可以通过相对大的电流（约为 2A 左右，具体请查三菱手册），通过电流的电压可以是 250VAC 也可以是 24VDC 以下

缺点是：输出触点响应时间相对较慢，约为 10ms。在有需要高速脉冲输出控制（如需要控制步进或伺服系统时）时不能选用继电器输出类型的 PLC

2) 晶体管输出

晶体管类型输出的 PLC 优点是：

输出触点响应时间快，约为 0.2ms 以下，经常应用在需要高速输出响应的系统里，

缺点是：只能通过 30VDC 以下的电压，通过电流小（约为 500MA 左右）

注意：在需要用高速脉冲输出控制步进或伺服系统时，一定要选用晶体管输出类型

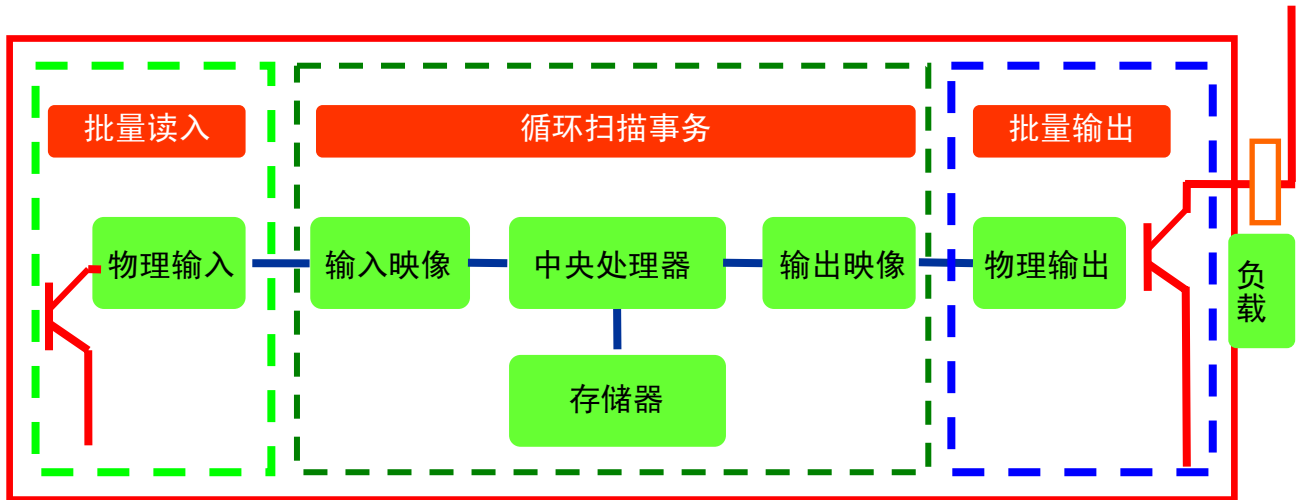
PLC 选型时需要注意：

- 1、输入电源电压：24VDC 还是 220VAC，根据供电系统电压来选择，一般来说，不建议直接从动力电源直接取电，最好用隔离变压器把动力电源和控制电源隔离开来，同时做好滤波等抗干扰措施。
- 2、输入输出点数：所谓的输入输出点数指的是整个控制系统需要用到的 PLC 的输入点个数和输出点个数。一般来说，当然是预先设计好整套系统方案，然后算出所有的输入输出点个数，如果是第一次设计的方案，建议预留点数
- 3、输出触点类型：是继电器输出类型还是晶体管输出类型。根据所控制的执行机构电源电压及功率来决定输出触点类型，如果系统里有需要用到高速脉冲输出控制步进电机或伺服系统时，一定要选择晶体管输出类型

在 PLC 选好了以后，接下来我们就要了解机械本体的各个检测和执行机构是怎么和 PLC 进行整体控制的

第四讲 PLC 的工作原理

我们先了解一下 PLC 的内部结构和外部元件的关系，看下图



PLC CPU单元的内部结构示意图

物理输入：指的是外部输入给 PLC 的信号（如上一节讲的 BL1、BL2、BL3 等外部传感器）

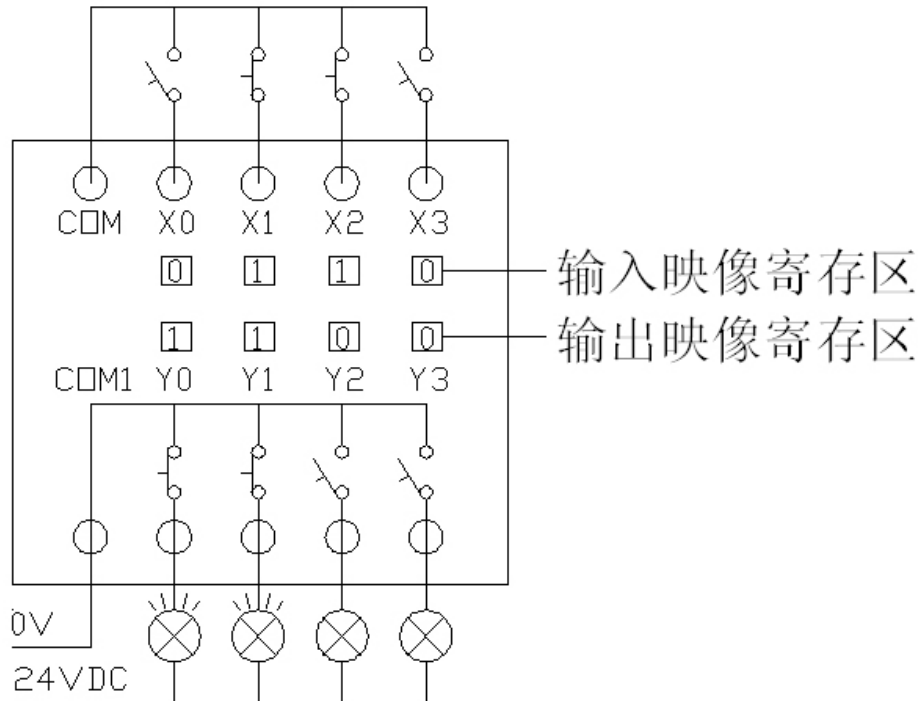
而我们知道，PLC 根本就不知道外部的传感器是什么东西。而它又是通过什么样的手段来知道外部传感器的状态的呢？这里，就引入了一个输入映像的概念。

这个**输入映像**就好像是外部输入端子的影子，当外部有信号输入时，它相对应的输入映像寄存区就为 1（在 PLC 里，1 表示真，也就是接通；0 表示假，也就是不通）。例如，当 X0 接通时，X0 的输入映像寄存区就的状态就为 1，不通时，为 0；从而，PLC 就可以直接通过扫描映像寄存区来知道外部端子的通断状态。

物理输出：指的是 PLC 输出去控制现场执行机构的端子输出信号（如上面例子的 YV 电磁阀）

同样，**输出映像**和物理输出也是对应的关系，当相对应的输出映像是 1 时，相对应的输出端了就接通，否则，就不通。如 Y0 映像寄存区为 1 时，Y0 输出端子接通（即 Y0 和 COM0 处导通状态）

请看下图，进一步理解映像寄存器的概念



输入：X1 和 X2 和 COM 接通，X1 和 X2 的输入映像寄存器都为 1，X0 和 X3 不通，映像寄存器为 0，PLC 就是通过扫描这些映像寄存器，从而知道各个输入端子的状态的。

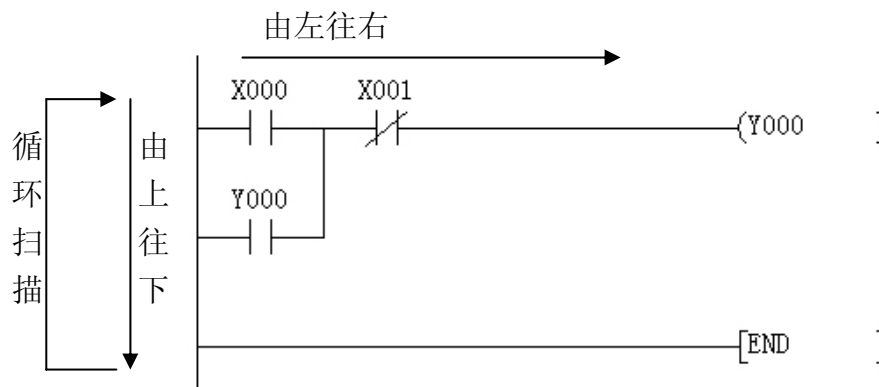
输出：Y0 和 Y1 有输入，映像寄存器为 1，Y0、Y1 和 COM1 导通，电路形成回路，灯泡亮，其它两个映像寄存器都为 0，不通，灯泡处于熄灭状态。

注意：

输入和输出没有必然的联系，PLC 仅仅只是通过扫描输入映像寄存器的状态，然后根据程序控制要求，输出控制执行机构

学完了映像寄存区的概念，下面了解下 PLC 的扫描机制及过程：

PLC只要一运行，就由左往右，由上往下，循环执行程序，并不停地刷新输入输出映像，如此循环运行不止，我们称之为扫描，把扫描执行一次需要的周期时间称为扫描周期。如下图



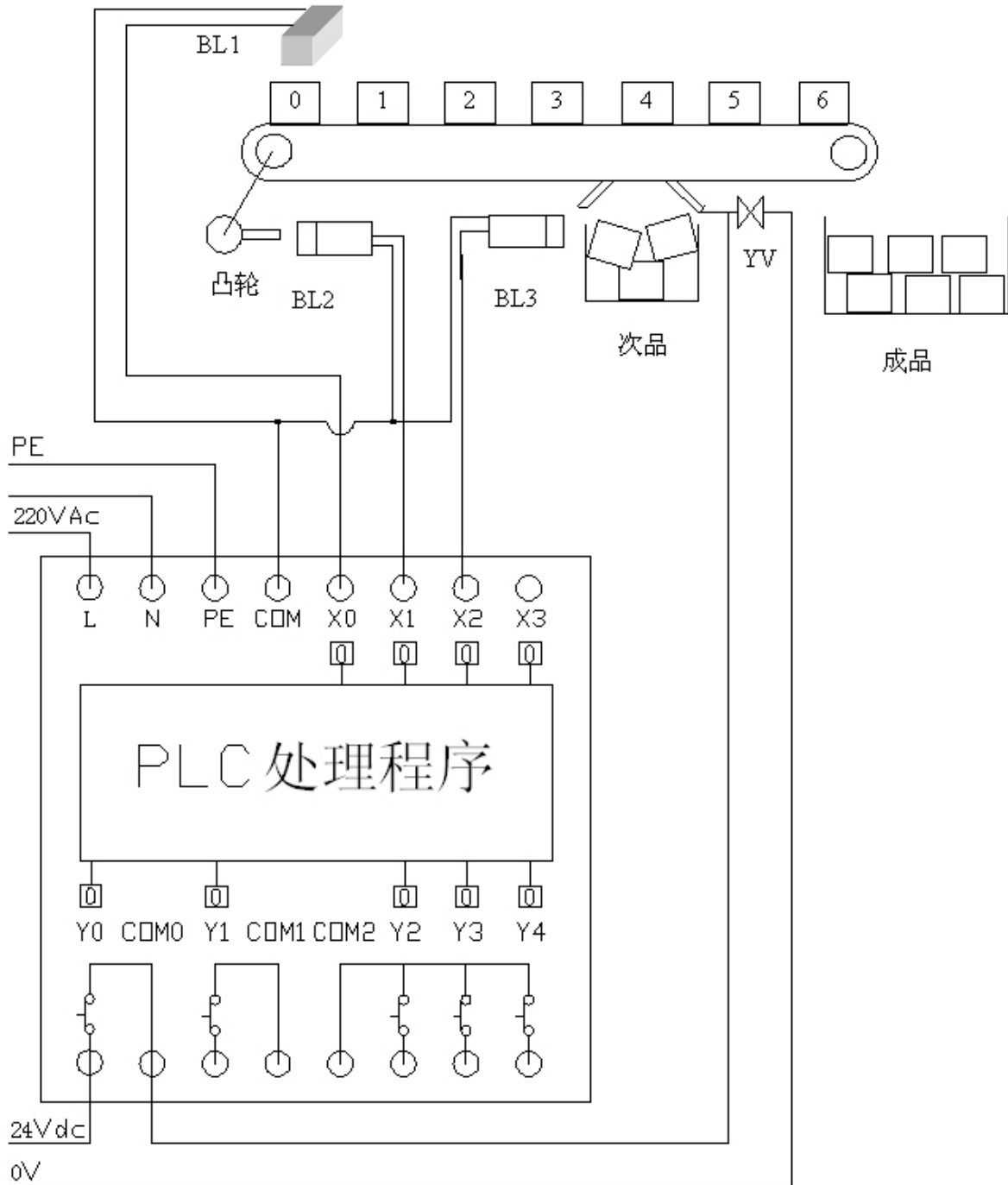
PLC 的扫描运行过程：

- 1、扫描到触点 X0，访问 X0 的映像寄存区，访问 Y0 映像寄存区，如果 X0 或 Y0 的映像寄存器其中一个或两个为 1，则导通
 - 2、接着扫描至 X1，访问 X1 的映像寄存区，如果 X1 的映像寄存区为 1，则触点动作，断开触点(X1 为常闭触点)，如果为 0，则保持原来的导通状态，
 - 3、这样，Y0 的线路就导通，给 Y0 的映像寄存器置 1，Y0 的输出端子就输出。
- 由此往复循环.....

至此，我们已经对 PLC 的整套硬件系统有个初步的了解，学至此，我们应该清楚地了解：

- 1、PLC 和机械本体之间的关系：它们之间的电气连接是什么样的
- 2、PLC 的选型：电源、输入、输出类型、所需要输入输出的点数
- 3、PLC 的输入信号与现场检测信号或其它输入信号的连线：X 与 COM 短接就表示有信号输入
- 4、PLC 应该怎么输出驱动现场的执行机构：Y 有输出，Y 内部触点就会导通
- 5、PLC 的输入信号，映像存器之间的关系：输入输出端子和映像寄存器为一一对应的关系
- 6、PLC 的扫描过程：由左往右，由上往下扫描

请看下一页的整套示意图



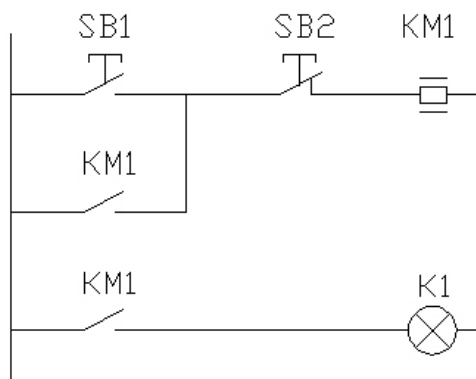
学完了整套硬件系统了以后，剩下的就是怎么把软件程序编写，然后怎么把这些软件装到 PLC 这个盒子里了，请接着学习第五讲，三菱 PLC 编程软件 GX Developer 应用

因为专业·所以优秀

第五讲 三菱 PLC 编程软件 GX Developer 应用

5. 1 梯形图与普通继电器电路的异同

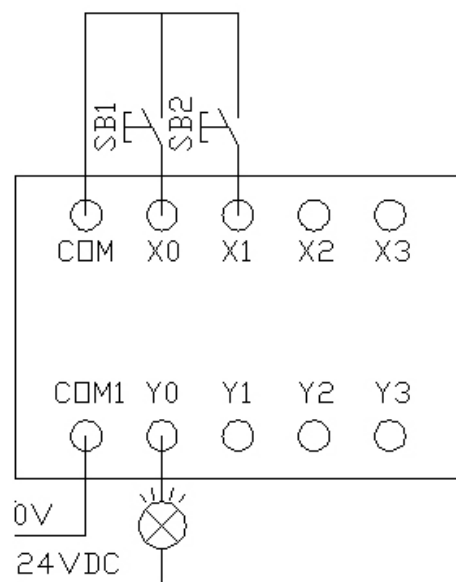
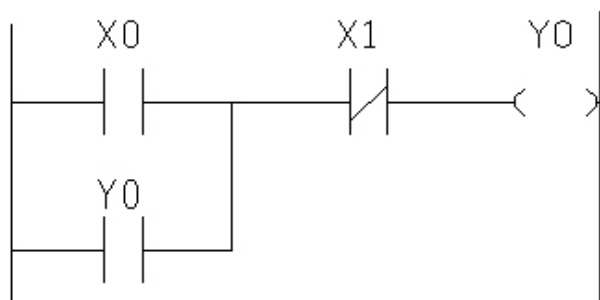
在开始编程时，先了解下我们后面要经常用到的梯形图语言。先看下下面的一个自锁继电器电路



SB1 按下，中间继电器 KM1 得电，相对应的触点 KM1 就会动作；

灯泡 K1 点亮，同时 KM1 闭合自导通，就使按钮 SB1 再放开，KM1 依然还是得电的，只有按下常闭开关 SB2 时，线圈 KM1 才会失电，灯泡才会熄灭。这就是继电器的自锁电路

如果我们现在要用 PLC 编程来替代，请看下面的梯形图



梯形图里的 X0 触点对应的就是端子 X0 的状态，当按钮 SB1 按下时，X0 和 COM 接通，X0 为有输入，梯形图里的 X0 触点动作，闭合梯形图里的 X1 触点

对应的就是端子 X1 的状态，当按钮 SB2 按下时，X1 和 COM 接通，X1 为有输入，梯形图里的 X1 触点动作，断开。

梯形图里的 Y0 相对应的就是 Y0 输出端子，当 Y0 导通输出时，Y0 和 COM1 导通，灯泡亮。

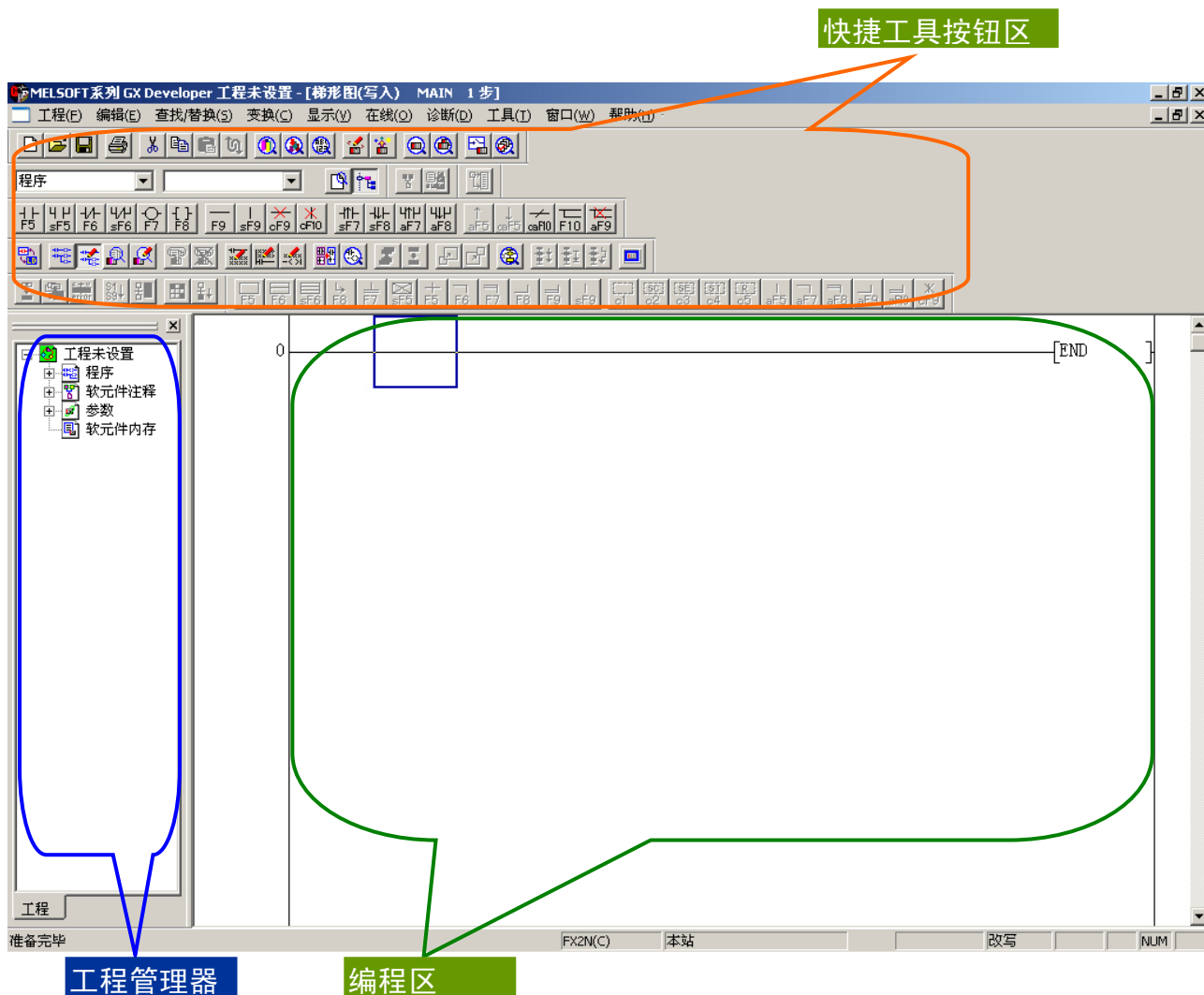
这个梯形图所实现的功能就是，当按钮 SB1 按下时，X0 端子接通，梯形图里的 X0 触点闭合，Y0 “得

电”，输出端子 Y0 输出，Y0 和 COM1 导通，灯泡亮；按下 SB2 时，Y0 “失电”，输出端子 Y0 没有输出，Y0 和 COM1 断路，灯泡熄灭。

由上面的例子可以得知，其实梯形图编程实际上就是从我们常用的继电器电路转换过来的，然后加了点计算机的功能，就成了我们现在所看到的 PLC 编程，下面就让我们开始我们的第一个程序。

5.2 新建一个新工程

先来了解下三菱编程软件 GX Developer 的界面



【设定目的】

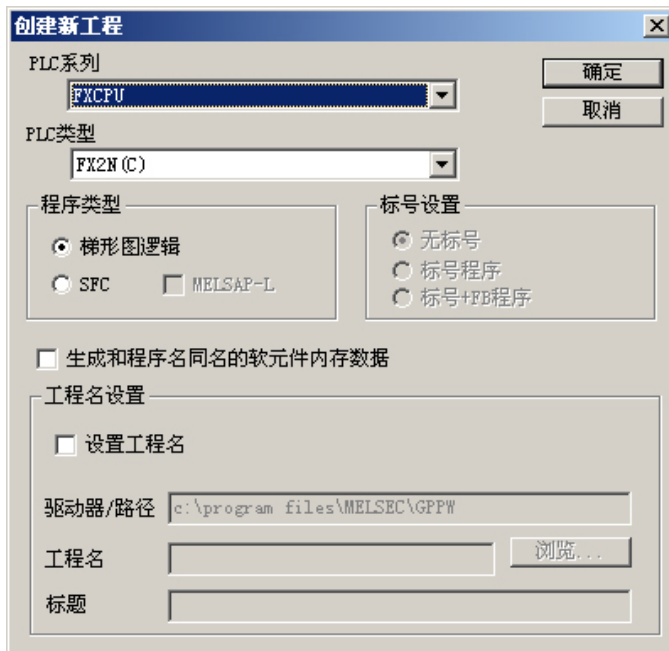
为了新建工程, 设定必要的PLC系列名PLC类型和工程名

【操作步骤】

[工程]->[新建工程]或者  (Ctrl + N), 弹出下面的对话框

因为专业·所以优秀

【设定画面】



第一步、选择PLC系列：选择三菱的PLC系列，在这里我们选择FXCPU

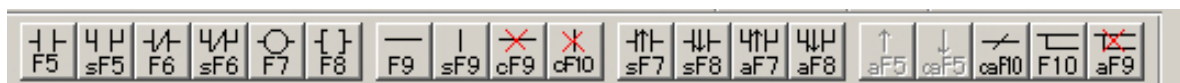
第二步、选择PLC类型：在这里我们选择FX2N(C)

第三步、选择程序类型：我们选择默认的梯形图，然后确定


注：其它的选项我们先不用管

5.3 常用工具按钮

我们平常所用到的逻辑梯形图都在常用工具栏里




先了解一下常用的梯形图操作

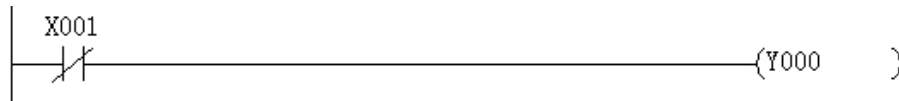
1、常开触点  F5：在正常情况下（没有信号输入）触点状态是断路状态，有信号输入时，触点闭合



正常情况下，X0 触点断路，Y0 状态为 0，即端子 Y0 没输出

当 X0 有输入时（X0 和 COM 短接），X0 闭合，Y0 状态为 1，即端子 Y0 输出

2、常闭触点 ：在正常情况下（没有信号输入）触点状态是通路状态，有信号输入时，触点断路




正常情况下，X1 触点处闭合状态，Y0 状态为 1，即端子 Y0 有输出

当 X1 有输入时（X1 和 COM 短接），X1 断开，Y0 状态为 0，即端子 Y0 没输出



驱动线圈：对输出继电器（例如 Y0）、中间继电器（例如 M0）、定时器（例如 T0）、计数器（例如 C0）进行驱动输出。

- a. 输出继电器 Y (Y0, Y1...) 就是直接和实际的物理端子输出进行连接，当 Y 有输出时，实际端子就有输出
- b. 中间继电器 M (M0, M1...) 仅仅只作为中间转换使用的断电器，并不作为实际的物理输出，在程序编写过程中一般做中间转换使用
- c. 定时器 (T0, T1...) 做定时用，和我们平常的时间继电器一样的功能，后面会再进一步详细讲解
- d. 计数器 (C0, C1...) 计数用，后面也会再讲解

3、划线输入 ：用来创建连线用，创建方法如下，创建完成后请弹出划线按钮（再点击一下即可）

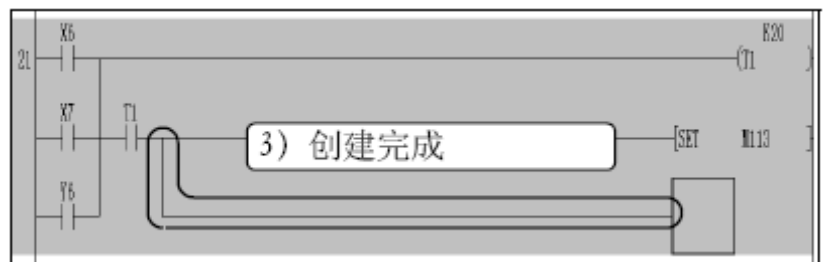


1) 单击工具栏  按钮。


2) 从开始位置向结束位置拖动鼠标。



3) 释放鼠标左键。规则线被创建。



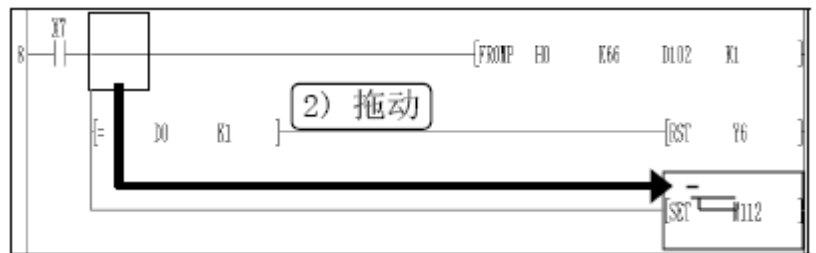
因为专业·所以优秀

4、划线删除 : 用来删除多余的连线用, 创建完成后请弹出删除划线按钮 (再点击一下即可)

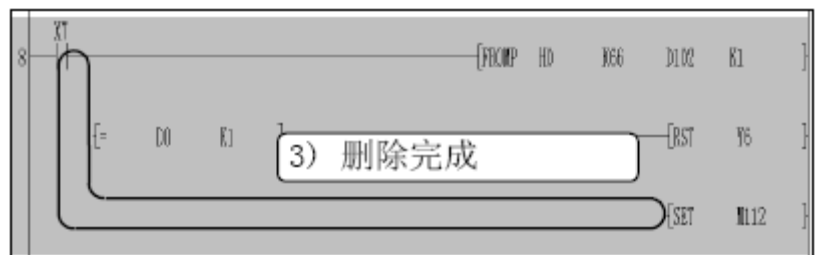


1) 单击工具栏  按钮。

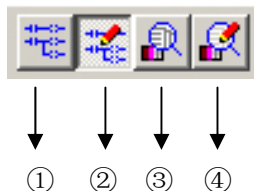
2) 从开始位置向结束位置拖动鼠标。



3) 释放鼠标左键, 删除完成。



5、程序装态



- ① 读出模式: 在该状态下, 只能查看或查找程序, 不能对程序做任何的修改或编辑, 单击该按钮就选中该状态
- ② 写入模式: 在该状态下, 才能修改和编辑程序, 单击该按钮就选中该状态
- ③ 监控模式: 在该状态下, 才能对程序进行状态监控, 但不能编辑程序, 单击该按钮就选中该状态
- ④ 监控 (写入模式): 在该状态下, 才能对程序进行状态监控, 也可以编辑程序, 单击该按钮就选中该状态


注意: 上载的程序一般为读出状态, 如果要对程序进行修改, 一定要选择写入模式

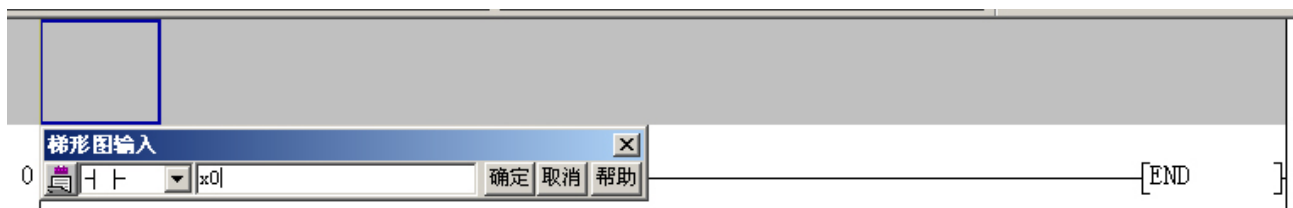
5.4 创建一个新的梯形图


了解了这几个常用的基本梯形指令操作后，下面我们就动手把上面我们了解到的自锁程序写到 PLC 里去。

第一步：插入一行，在编程空间里，点击鼠标右键，选择“行插入”，或 Shift + Inset




第二步：输入X0常开触点：单击工具栏  按钮，或按F5键，打开程序输入窗口，键入“X0”，确定




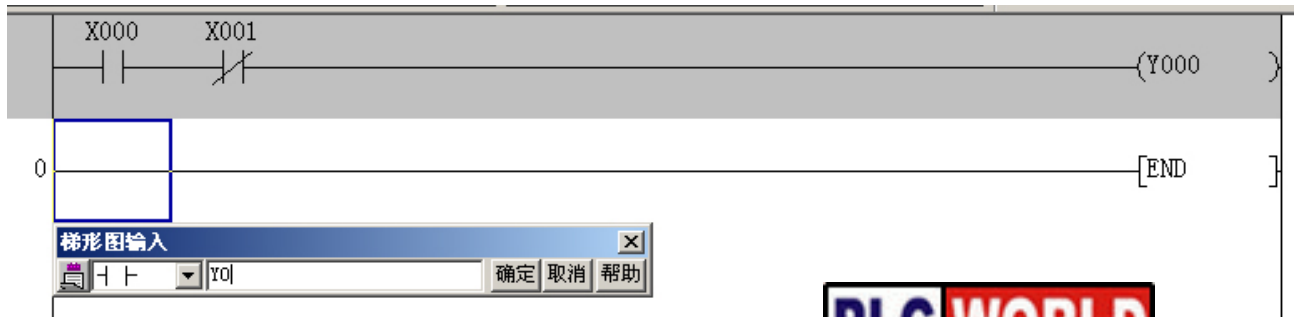
第三步：输入X1常闭触点：单击工具栏  按钮，或按F6键，打开程序输入窗口，键入“X1”，确定




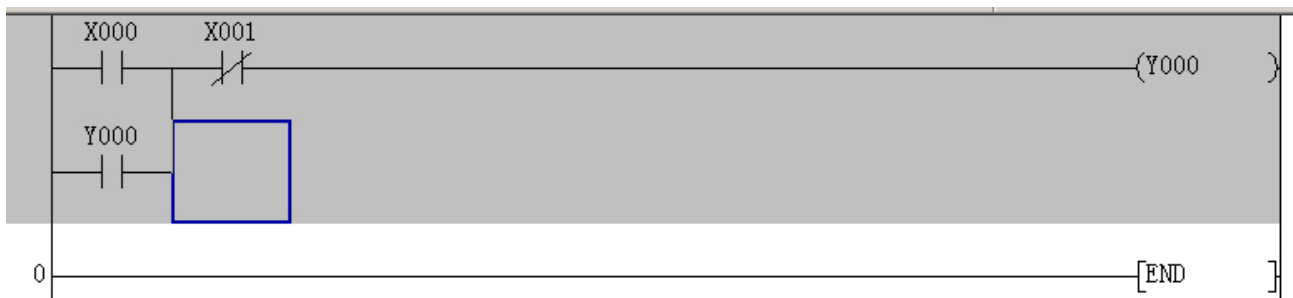
第四步：输入Y0输出线圈：单击工具栏  按钮，或按F7键，打开程序输入窗口，键入“Y0”，确定





第五步：输入 Y0 常开触点：单击工具栏  按钮，或按 F5 键，打开程序输入窗口，键入“Y0”，确定

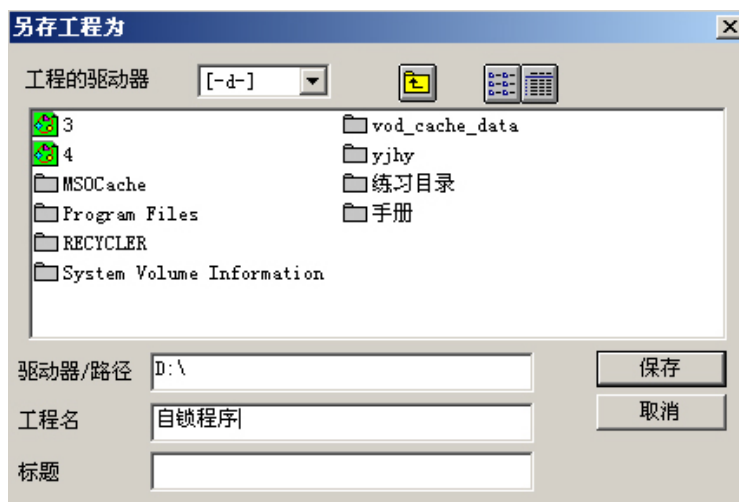


第六步：并联 X0 和 Y0 触点（使用画线功能  连线）



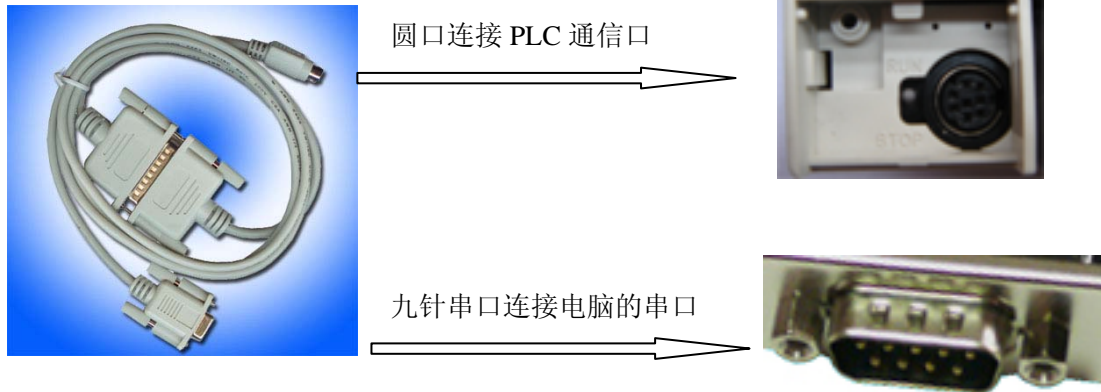
第七步：转换梯形图（把梯形图编译成可以下载到 PLC 的代码），点击  ，或按下 F4

第八步：程序存盘：点击  或者 点击下接菜单 工程 → 保存工程 ，使用 Ctrl + S 也行
选择存盘路径，输入要保存的工程名字，例如：自锁程序。



第九步，连接 PLC 并检查通信。

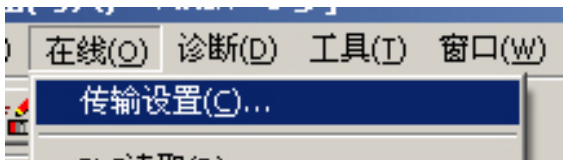
1) 首先，需要一条三菱 PLC 下载线，用来连接电脑与 PLC。



2) 设置好 PLC 与电脑通信的端口号

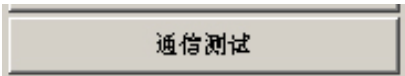
(电脑的串行通信端口号查看方法：点击我的电脑—点击鼠标右键—选择属性—选择硬件页面—选择设备管理器—点开通信端口，就可查看到当前的串行通信口编号是多少)

点击下拉菜单 在线—传输设置，弹出传输设置对话框




双击  弹出串口设置对话框，设置好 COM 端口号，波特率选择 9.6Kbps 或 19.2Kbps 都可

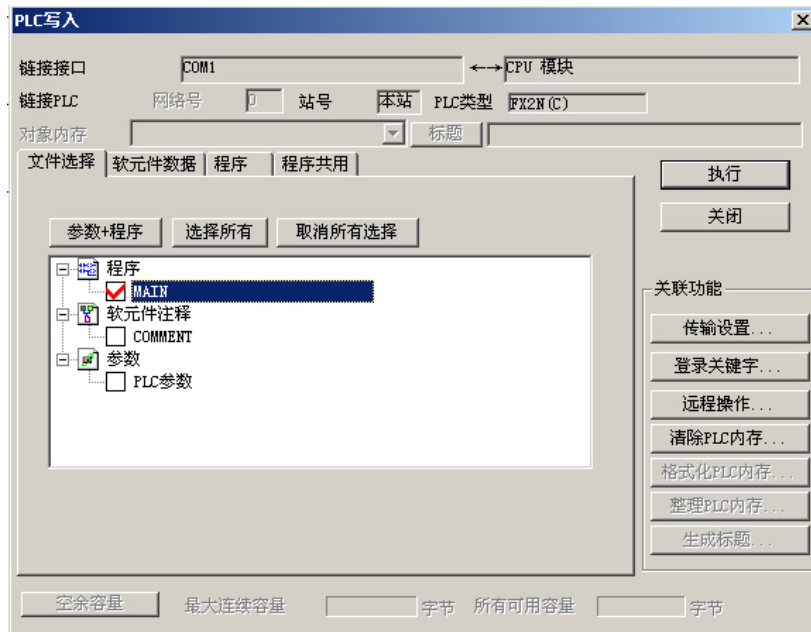


确认后，测试通信下，点击  如果通信成功，则弹出成功对话框，否则，请再检查通信电缆和通信串口号。

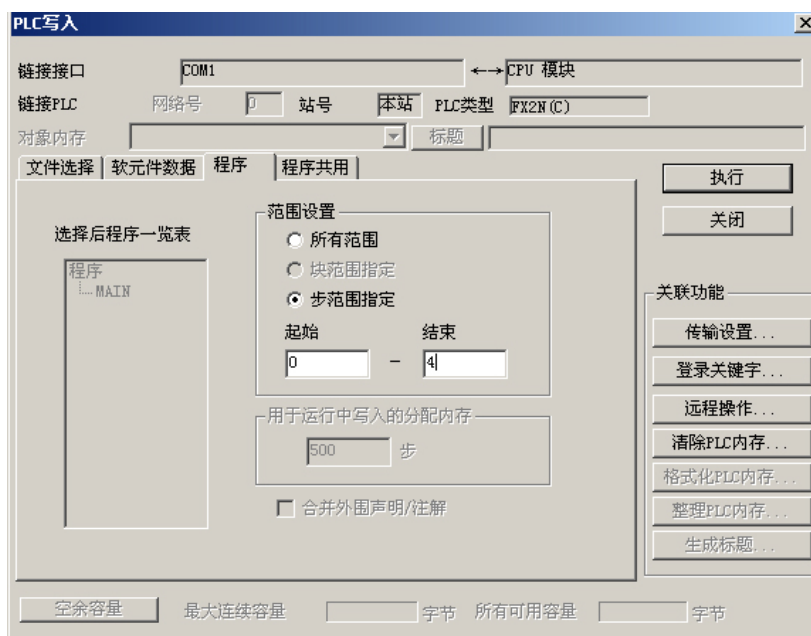
因为专业·所以优秀

第十步，把编写编译转换完的程序下载到 PLC 里

1) 点击  或选择 在线—PLC 写入，弹出写入对话框，这里我们只选择写入程序，把 MAIN 的对话框打上勾



2) 选择程序页面，输入要下载的程序步数，在范围设置里，选择“步范围指定”，指定好步数



程序的步数为：页面顶上的步数减去 1，如下图，步数为 $5 - 1 = 4$ 步



指定好步数后，点击执行按钮，一路选择“是”，程序就下载完毕。

因为专业·所以优秀

第十一步，运行并监控程序状态

1) 把 PLC 的状态打到 RUN 位置



2) 把程序状态打到监控模式



这样，就实时地监控到 PLC 程序运行的状态，如下图

A. SB1、SB2 在不接通状态



B. SB1 接通、SB2 不接通



C. SB1 接通以后，Y0 自锁




D. SB2 接通，Y0 断路

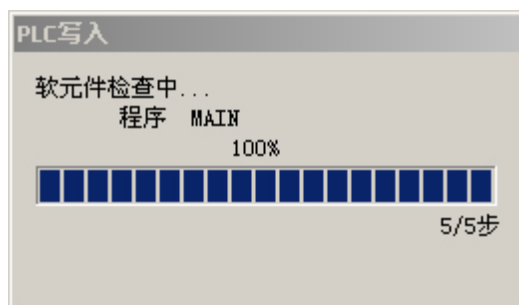
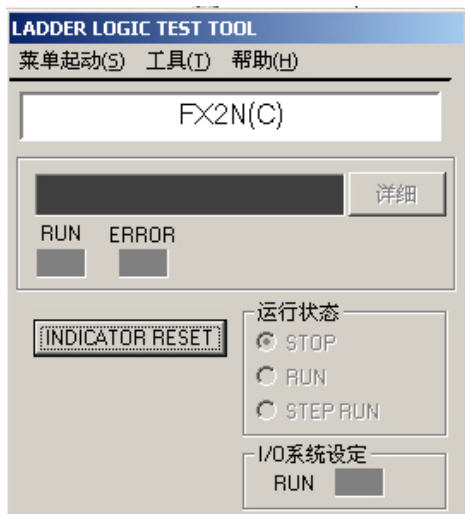


以上就完成了—个 PLC 程序从 新建—输入梯形图—PLC 连接—下载—监控的过程。

不过有时候我们没有 PLC 的时候怎么办？请接着看下一节：使用 GX Simulator 模拟代替 PLC

5. 5 使用 GX Simulator 模拟

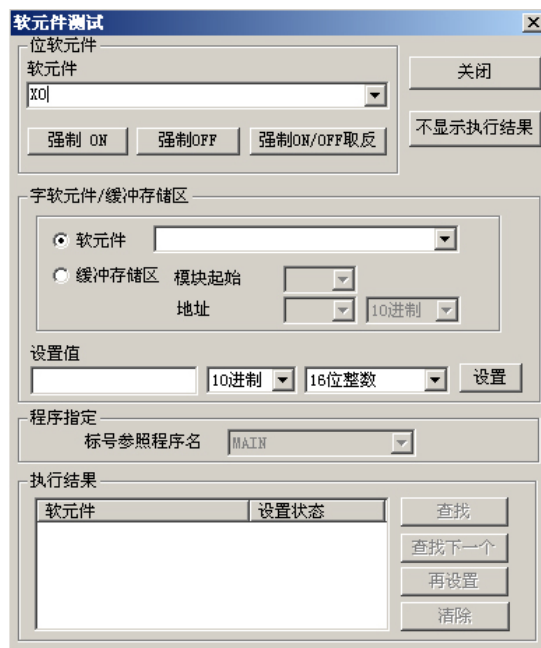
程序编写转换完毕后（即上一节的第七步完成），点击 （需要安装完毕模拟软件才能使用，否则图标是灰色的），运行模拟，如下图



转入程序监控状态



我们要给接通 X0（就是给 X0 赋值），点击  在位软元件栏里输入 X0，点击“强制 ON”按钮，X0 就强制接通，如果要断开，就点击“强制 OFF”按钮。



因为专业·所以优秀

我们除了可以直接在程序里监控各软元件的状态，还可以进入软元件登录页面查看

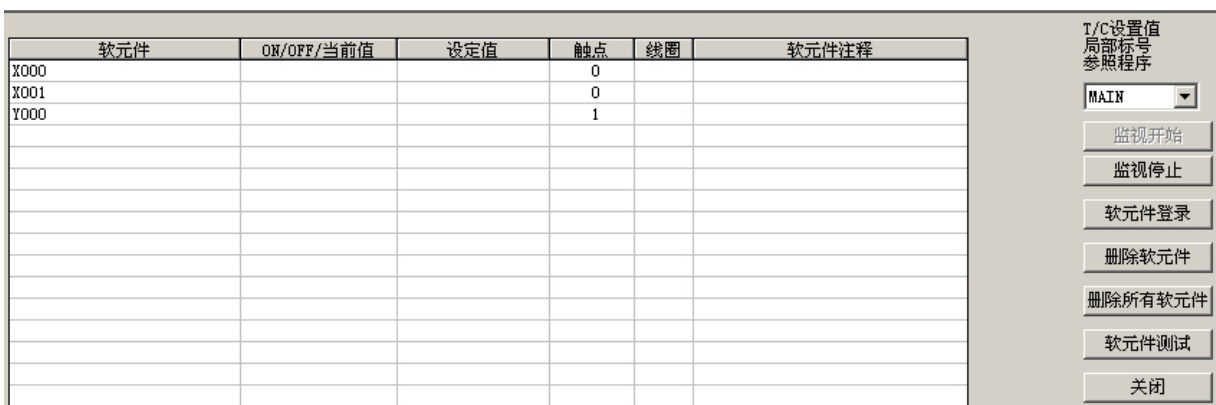
1、点击 ，进入软元件登录页面



2、点击“软元件登录”按钮，输入要监视的软元件，点击登录



3、点击“监视开始”按钮，就开始监视各软元件的状态，如需要关闭该页面，请点击“关闭”按钮



注：双击软元件可以弹出“软元件测试”对话框，然后可以给各软元件赋值

第六讲 位逻辑操作

位逻辑指的就是数字量的操作，前面我们讲了常开、常闭，下面，我们接着讲

上升沿脉冲 $\text{—}\uparrow\text{—}$ 例： $\text{—}\uparrow\text{—}$ ^{X0}

下降沿脉冲 $\text{—}\downarrow\text{—}$ 例： $\text{—}\downarrow\text{—}$ ^{X0}

上升沿脉冲指令 PLS 和下降沿脉冲指令 PLF

置位 SET 例：SET Y0

复位 RST 例：RST Y0

区间复位 ZRST 例：ZRST Y0 Y5

运算结果取反 $\text{—}/\text{—}$

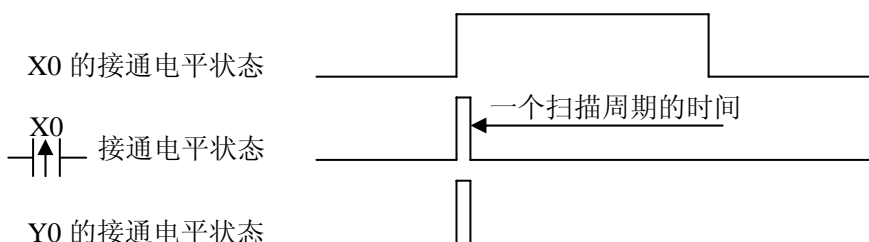
6.1 上升（下降）沿脉冲

上升沿：上升沿指的就是信号从 OFF 转向 ON 的那一瞬间

下降沿：下降沿指的就是信号从 ON 转向 OFF 的那一瞬间

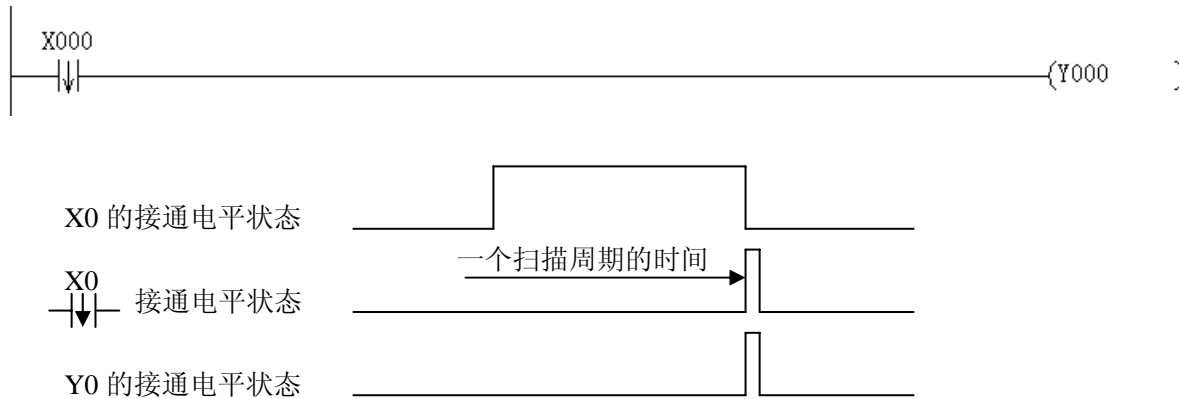


上升沿脉冲 $\text{—}\uparrow\text{—}$ ：指的是当 PLC 检测到有上升沿信号时，保持一个扫描周期的高电平，如



因为专业·所以优秀

下降沿脉冲 $\text{—}\downarrow\text{—}$ ：指的是当 PLC 检测到有下降沿信号时，保持一个扫描周期的高电平，如



由上面的例子可以看到，如果使用了上升（或下降）沿脉冲，Y0 输出只会在 X0 上升（或下降）沿的瞬间接通一个扫描周期（也就是只会接通一次的意思，不管 X0 接通的时间有多长），至下个扫描周期，上升沿触点就会断开。

上升（或下降）沿一般都是使用在外部信号接通（或断开）瞬间，只需要执行一次程序的场合

上升（或下降）沿触点在上升（或下降）沿的那一瞬间，只保持一个扫描周期的高电平（接通）状态

注意：

因为 PLC 的扫描速度很快，一般的按钮按下至松开的时间，不管你按的速度多快，都会比一个扫描周期长很多，有的可达到几百、几千个扫描周期，也就是说按钮按下至松开的这段时间，PLC 已经循环扫描执行程序几百、几千次了。

所以，在只需要程序执行一次的场合，必须要加上上升（或下降）沿脉冲指令

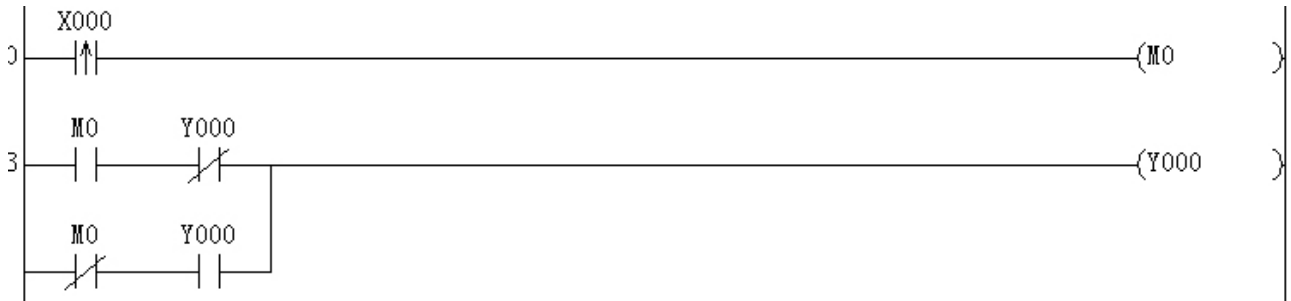
下面请看一个单按钮启/停的例子：

按下按钮，接通 X0，Y0 输出，再按下 X0，Y0 熄灭

请注意分析每个扫描周期的各个寄存器及触点的状态

因为专业·所以优秀

程序:

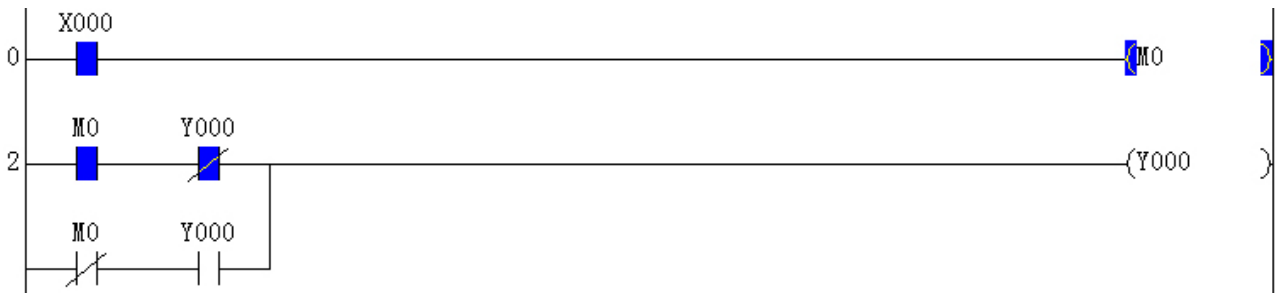


当 X0 按下接通瞬间

第一个扫描期各寄存器状态:

X0 上升沿脉冲接通, M0 线圈得电

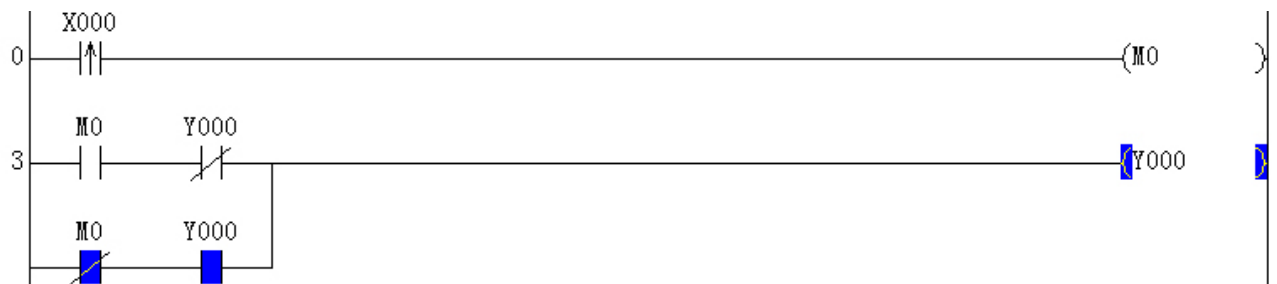
当 PLC 由上升下扫描至 M0 触点时, 因为 M0=1, 所以 M0 触点动作, 常开触点闭合, 常闭触点断开, 接通 Y0, Y0=1, 如下图



第二个扫描周期

X0 上升沿脉冲不再接通, M0 线圈失电

当 PLC 扫描至 M0 触点时, 因为 M0=0, 所以 M0 触点复位, 常开触点断开, 常闭触点接通, 而上个扫描周期的 Y0=1, 触点动作, 常开触点接通, 常闭触点断开, Y0 线圈自锁, 如下图



第三以后的扫描周期, Y0 依然保持自锁状态, 如果再按下按钮, 接通 X0

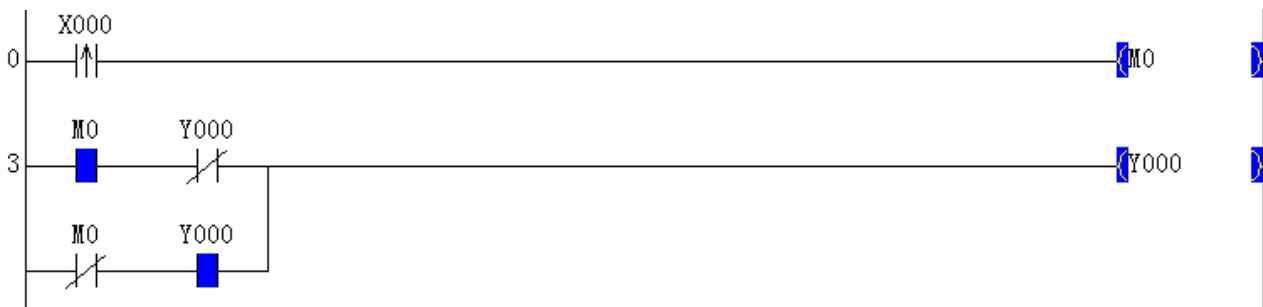
因为专业·所以优秀

当 X0 按下接通瞬间

第一个扫描期各寄存器状态：

X0 上升沿脉冲接通，M0 线圈得电

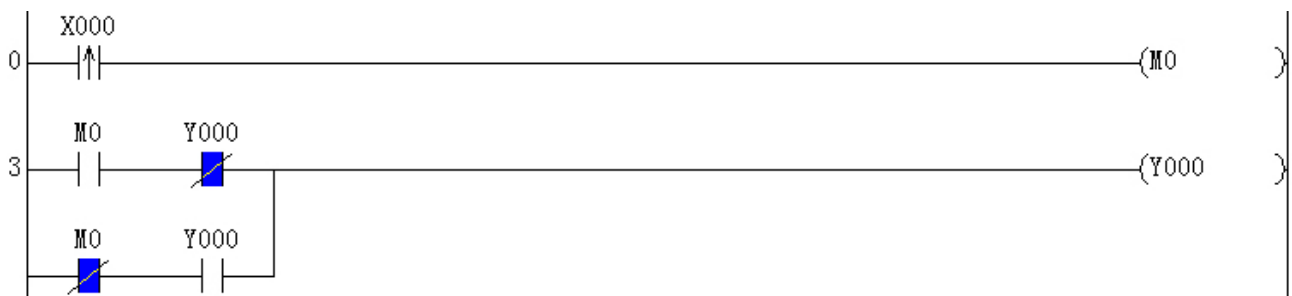
当 PLC 由上往下扫描至 M0 触点时，因为 M0=1，所以 M0 触点动作，常开触点闭合，常闭触点断开，而此时的 Y0 状态为 1 也就是接通状态（因为之前它都在自锁接通状态），所以相对应的 Y0 触点也是常开的闭合，常闭的断开，结果是 Y0 线圈失电，如下图



第二个扫描周期

X0 上升沿脉冲不再接通，M0 线圈失电

当 PLC 扫描至 M0 触点时，因为 M0=0，所以 M0 触点复位，常开触点断开，常闭触点接通，而上个扫描周期的 Y0=0，触点复位，常开触点断开，常闭触点接通，Y0 线圈依然在失电状态，如下图



接下来的扫描周期，Y0 依然保持失电状态，如果 X0 再次接通的话，那就重复上面接通的过程

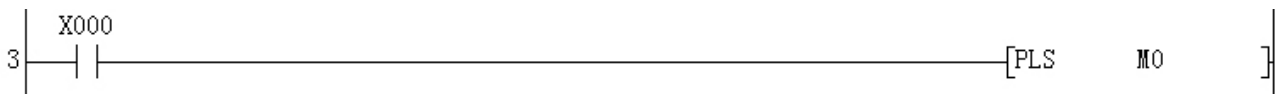


PLS 和 PLF 指令

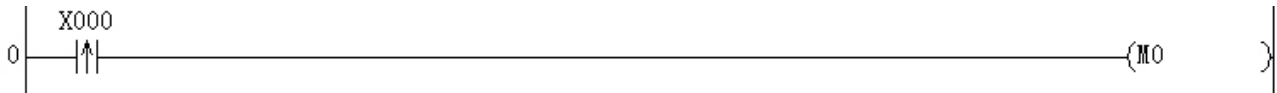
PLS 和 PLF 指令其实和我们前面学到的上升沿脉冲 \uparrow 和下降沿脉冲 \downarrow 一样。

PLS 是上升沿，表示的是在驱动输入为 ON 的瞬间，保持一个扫描周期的高电平

例：表示在 X0 接通的瞬间，M0 动作（接通）一个扫描周期，下个扫描周期后状态就不再为 1（接通）



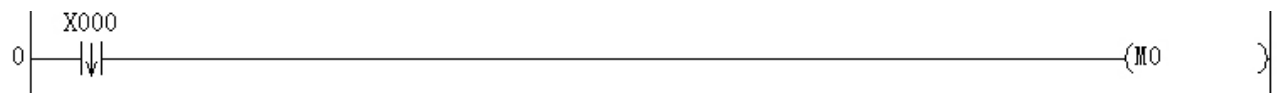
上面的程序和下面的执行结果是一样的



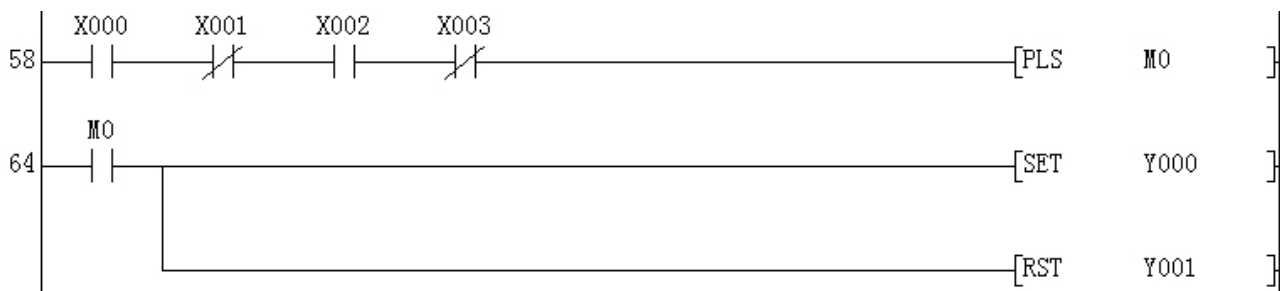
PLF 是下降沿，表示的是在驱动输入为 OFF 的瞬间，保持一个扫描周期的高电平



和下面的程序效果一样



PLS 和 PLF 更多时候会用在很多触点条件之后，表示在众多条件满足接通 ON 或 OFF 的瞬间执行某个动作，如：



当 X0, X1, X2, X3 四个触点条件满足（即 X0 和 X2 接通输入，X1 和 X3 不接通）的瞬间，接通 M0，M0 接通，执行置位 Y0，复位 Y1 的指令，下个扫描周期，M0 不再接通

因为专业 · 所以优秀

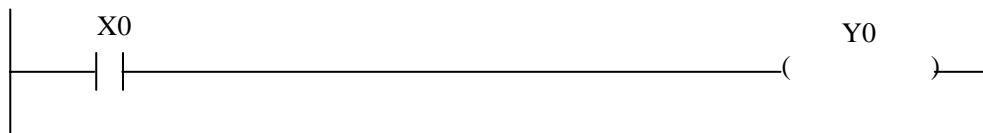
6.2 置位、复位指令

置位：SET

复位：RST

在这之前我们知道通过线圈直接可以输出使灯泡点亮；

然后我们又知道了三菱 PLC 的 Y 输出与否是和其相对应的输出映像寄存器有关，当其相对应的映像寄存器为 1 时，相对应的输出端子就接通，请看下面的例子：



PLC 扫描到 X0 触点，然后访问 X0 的映像寄存器，如果 X0 的映像寄存器的状态为 1（即 X0 端子和 COM 短接），则认为 Y0 的线圈线路是接通的，PLC 把 Y0 的映像寄存器的状态置 1，然后接着往下扫描，如此循环。

在每一个扫描周期里，PLC 都会访问相应的 X0 映像，同时刷新 Y0 的映像寄存器。

而 Y0 的物理端子到底有没有接通输出，由 Y0 的映像寄存器的状态决定。

当 X0 接通，Y0 线圈得电，Y0 的映像寄存器的状态置 1，Y0 的物理端子就输出，如接上外围电路连通灯泡，灯泡就亮。

反之，Y0 的映像寄存器的状态为 0，Y0 的物理端子就没有输出，灯泡不会亮。

由此可见

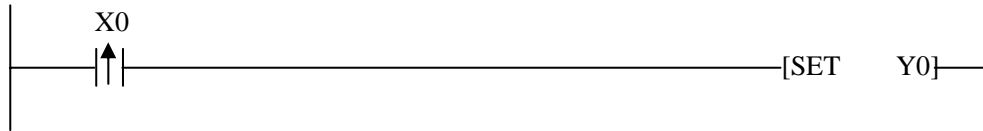
**PLC 的物理端子有没有输出，是由相对应的映像寄存器的状态决定的！
相对应的映像寄存器状态为 1，有输出；为 0，无输出**

而我们看到上面的 Y0 线圈得电失电，相对应的物理端子有输出和无输出的，

实质就是：PLC 根据线圈的状态不断地对 Y0 的映像寄存器进行置位（设 1）和复位（复 0）的结果。

了解了 PLC 输出的实质以后，我们再来看置位 SET 和复位 RST

置位 SET：PLC 一执行置位指令，就是对相对应的位寄存器进行置位（设 1）的动做。例：



当 X0 上升沿来时，接通 [SET Y0] 这条指令，把 Y0 映像寄存器的状态置 1，Y0 就有输出。下个扫描周期来时，X0 上升沿的指令触点变为断路状态（上升沿指令只保持一个扫描周期的高电平），不再执行 [SET Y0]，请思考一下，Y0 端子的状态会是什么样？

很显然，Y0 端子还是处于输出导通状态，道理很简单，因为 Y0 的映像寄存器的状态依然还是 1。这时可能会有人问，那为什么线圈输出的 Y0，线路断的时候，Y0 就会失电，Y0 端子就不再有输出？这是因为线圈输出的执行过程是：

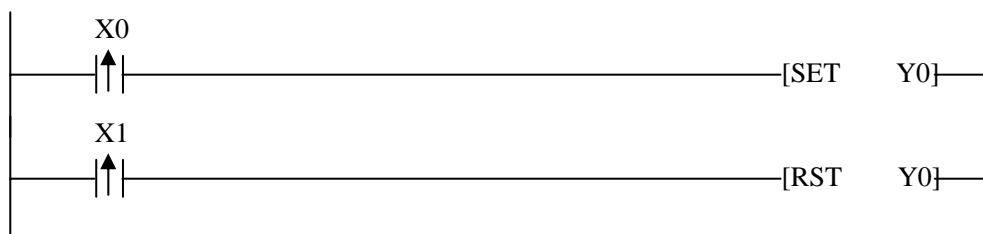
当通路时，Y0 线圈得电，即 PLC 会把 Y0 映像寄存器置 1；断路时，Y0 线圈失电，即 PLC 会把 Y0 映像寄存器复 0。

线圈断路时 Y0 不再有输出，不是 PLC 不执行指令，而是 PLC 执行了把 Y0 映像寄存器复 0 的动作。

而指令 [SET Y0] 的功能仅仅只是把 Y0 的映像寄存器置 1。

虽然由于断路了，PLC 不再执行该程序，但 Y0 的映像寄存器依然还是 1；就好像你去把灯打开了，你虽然不再按着开关，灯依然还是亮的一样。

那如果我们想要灯灭掉，即 Y0 不再有输出怎么办？这里就有了复位指令 RST，RST 指令和 SET 指令一样，只不过它是对状态位置 0 的。如



按下 X0，Y0 有输出（灯亮），按下 X1，Y0 没输出（灯灭）

请思考，如果要求，用置位复位指令实现：按下 X0，Y0 有输出（灯亮），松开 X0，Y0 没输出（灯灭）应该怎么实现？

上面我们学习了单个的置位和复位，但有的时候，我们想一下子把好多点都复位时怎么办？这里就引进了区域复位的概念

区域复位：ZRST Y0 Y5 执行指令结果是把 Y0—Y5（Y0、Y1、Y2、Y3、Y4、Y5）全都置 0

例：



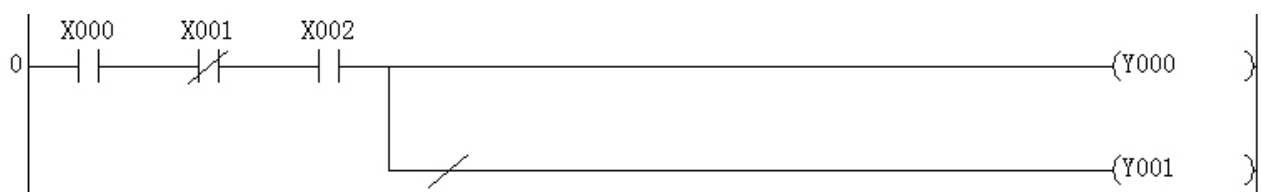
按下 X0，置位 Y0

按下 X1，置位 Y1

按下 X2，置位 Y2

按下 X3，Y0、Y1、Y2 都复位

运算结果取反 



当 X0、X1、X2 三个触点串联的通路导通，则 Y0 输出，因为 Y1 前面加了运算结果取反指令，所以 Y1 无输出

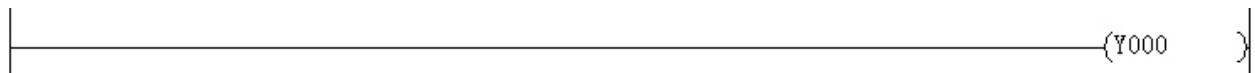
当 X0、X1、X2 三个触点串联的通路不导通，则 Y0 无输出，因为 Y1 前面加了运算结果取反指令，所以 Y1 有输出

学到这里，我们总结一下，我们都学习了什么内容，应该掌握哪些要点

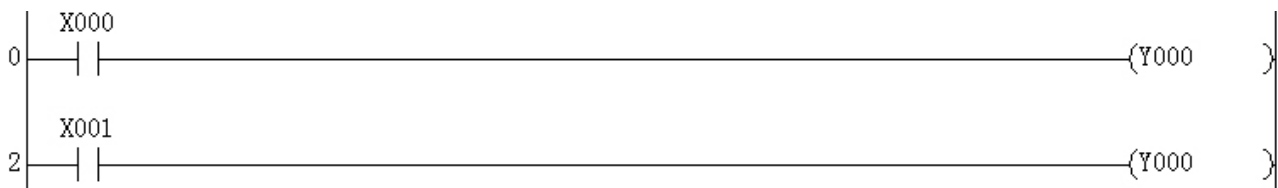
- 1、自动化系统的概念，PLC 和外围设备的关系
- 2、PLC 和外部设备是怎么连接的，输入输出关系是什么样的？
- 3、三菱 PLC 的选型
- 4、三菱 PLC 的 X 输入端子怎么样接通输入，Y 输出端子怎么样接通输出控制执行机构
- 5、PLC 扫描机制，扫描周期，映像寄存器的实质
- 6、三菱编程软件的应用，新建工程，编写程序，修改程序，下载程序，模拟器应用等
- 7、常开触点、常闭触点、输出线圈、中间继电器 M
- 8、上升/下降沿脉冲
- 9、置位/复位指令应用及和线圈输出的本质
- 10、区域复位指令
- 11、运算结果取反指令

这里，在编程时还要注意以下几个问题

- 1、输出线圈或指令前一定要连有触点条件，像下面的程序是不对的，编译不了的



- 2、同一个程序里不要出现两个以上相同名字的线圈，如



为什么呢？假设 X0 接通时，第一条程序就接通了 Y0，而 PLC 扫描到第二条程序时，如果 X1 是断路的，那 PLC 又把 Y0 给复位了，所以，第一条程序和第二条程序是冲突的。

如果出现这种情况，PLC 的执行输出结果是最后那一条程序的执行结果

这里，再学习两个常用的特殊中间继电器

M8000：当 PLC 运行（RUN）状态，其触点会处于接通状态

M8002：在 PLC 从停止状态转至运行（RUN）状态时，接通一个扫描周期，相当于



该中间继电器经常用在程序初始化，这在以后再详细讲解

下面，进行初步的逻辑程序工程设计讲解。

案例分析

1、试使用 PLC 控制两台三相异步电动机，要求如下

- (1) 两台电动机互不影响地独立操作启动与停止（即可通过两个独立的按钮分别控制电机 M1 和电机 M2 的启停）
- (2) 能同时控制两台电动机的停止（即通过一个按钮同时控制电机 M1 和电机 M2 的启停）

要求设计出连接电机的电气连接图，及编写正确的 PLC 程序，输入输出端子可任意选择。或按下面的端子分配

- X0: 系统启动
- X1: 系统停止
- X2: M1 电机启动
- X3: M1 电机停止
- X4: M2 电机启动
- X5: M2 电机停止
- X6: M1 和 M2 电机同时启动
- X7: M1 和 M2 电机同时停止
- Y0: 控制 M1 电机接触器的启停（得电启动，失电停止）
- Y1: 控制 M2 电机接触器的启停（得电启动，失电停止）

设计过程：

一、输入输出分析：输入点数 8 位，输出点数 2 位

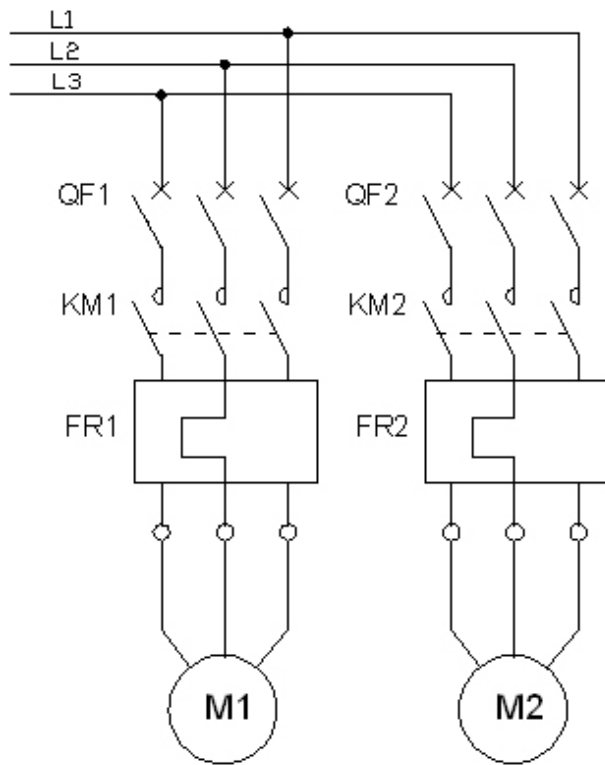
二、PLC 选型：选择 FX1N24MR 220Vac 输入，14 路输入，10 路继电器输出类型

三、设计电气图纸，面板图如下

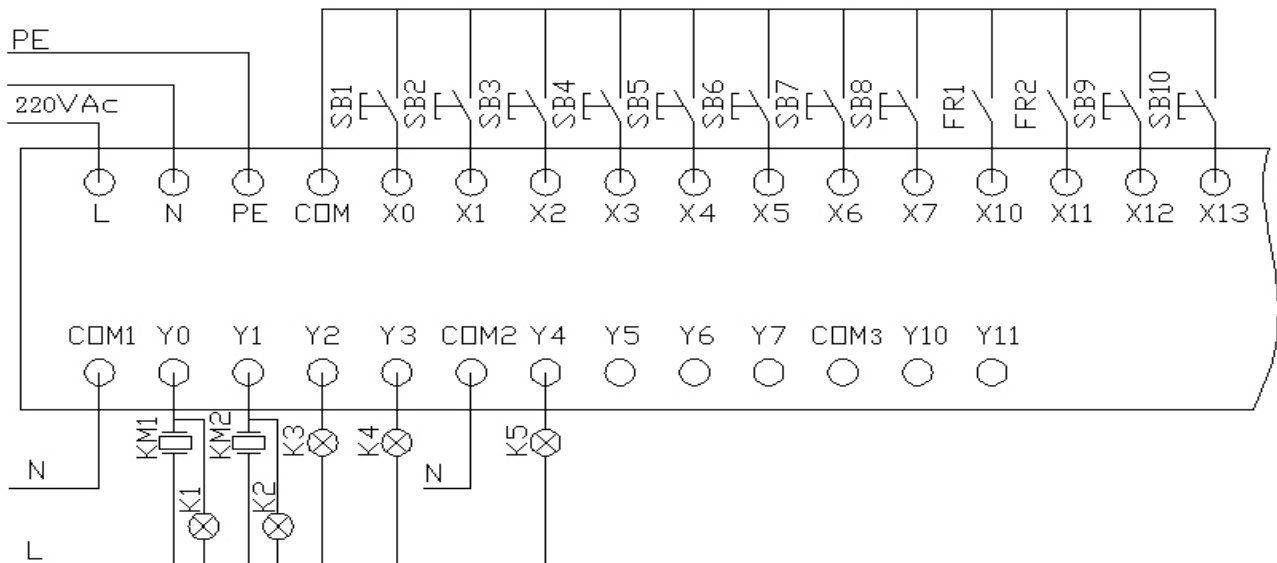


因为专业·所以优秀

主要的电气图设计



- QF: 空气开关
- KM: 接触器
- FR: 热继电器
- SB1(X0): 系统启动按钮
- SB2(X1): 系统停止
- SB3(X2): M1 电机启动
- SB4(X3): M1 电机停止
- SB5(X4): M2 电机启动
- SB6(X5): M2 电机停止
- SB7(X6): M1 和 M2 电机同时启动
- SB8(X7): M1 和 M2 电机同时停止
- FR1(X10): M1 热继电器过载信号
- FR2(X11): M2 热继电器过载信号
- SB9(X12): 故障复位
- SB10(X13): 紧急停止
- K1: M1 电机启动信号灯
- K2: M2 电机启动信号灯
- K3: 系统启动信号灯
- K4: M1 电机过载信号灯
- K5: M2 电机过载信号灯

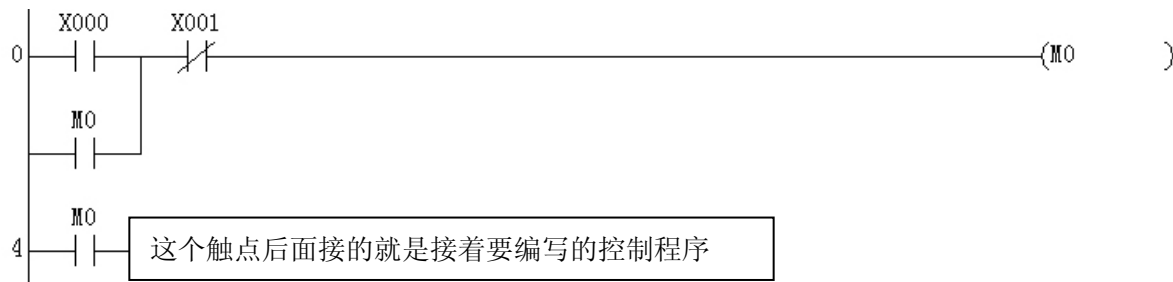


因为专业 · 所以优秀

PLC 编程思路及步骤:

第一步: 系统启停按钮功能: 即按下 X0 系统启动 (接通程序), 按下 X1 系统停止 (即断开程序)

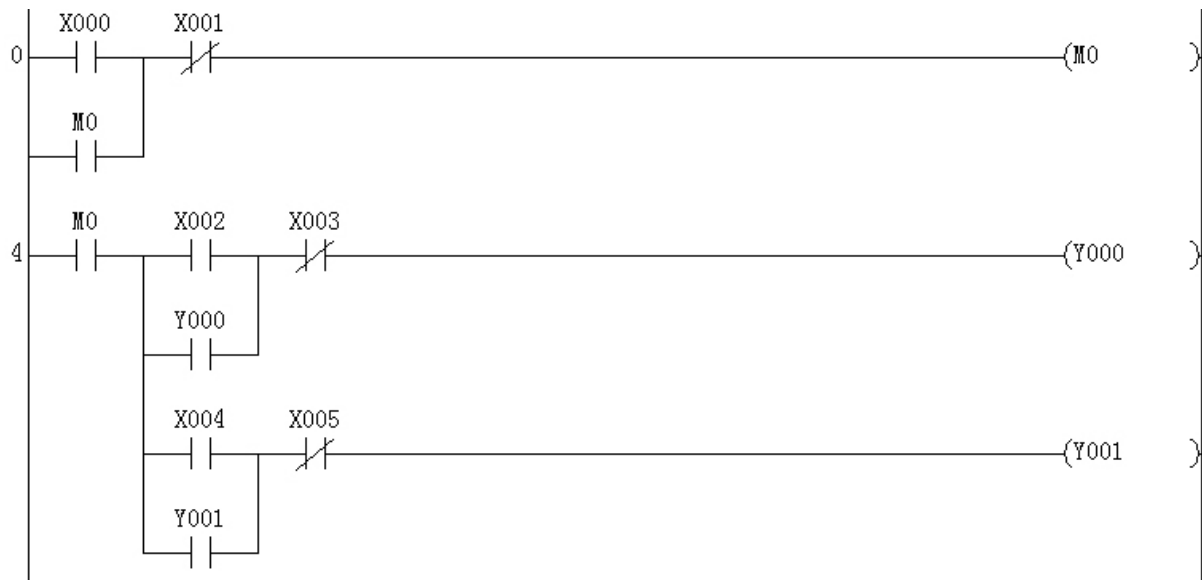
这里我们用一个中间继电器 M0 来当系统开关, 如下:



按下 SB1, 接通 X0, M0 得电并锁住, M0 触点接通, 才能执行后面的程序

按下 SB2, 接通 X1, 断开 M0, M0 触点失电断开, 切断程序

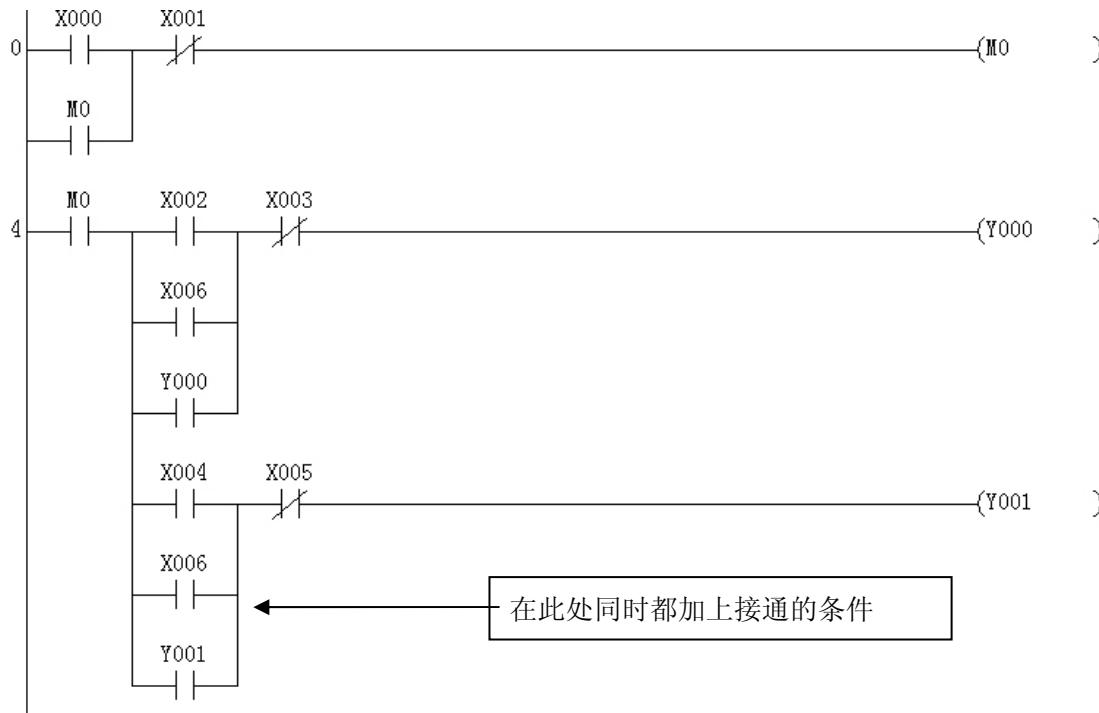
第二步, 编写 M1 电机和 M2 电机独立运行的程序, 即 Y0、Y1 输出的自锁程序, 如下:



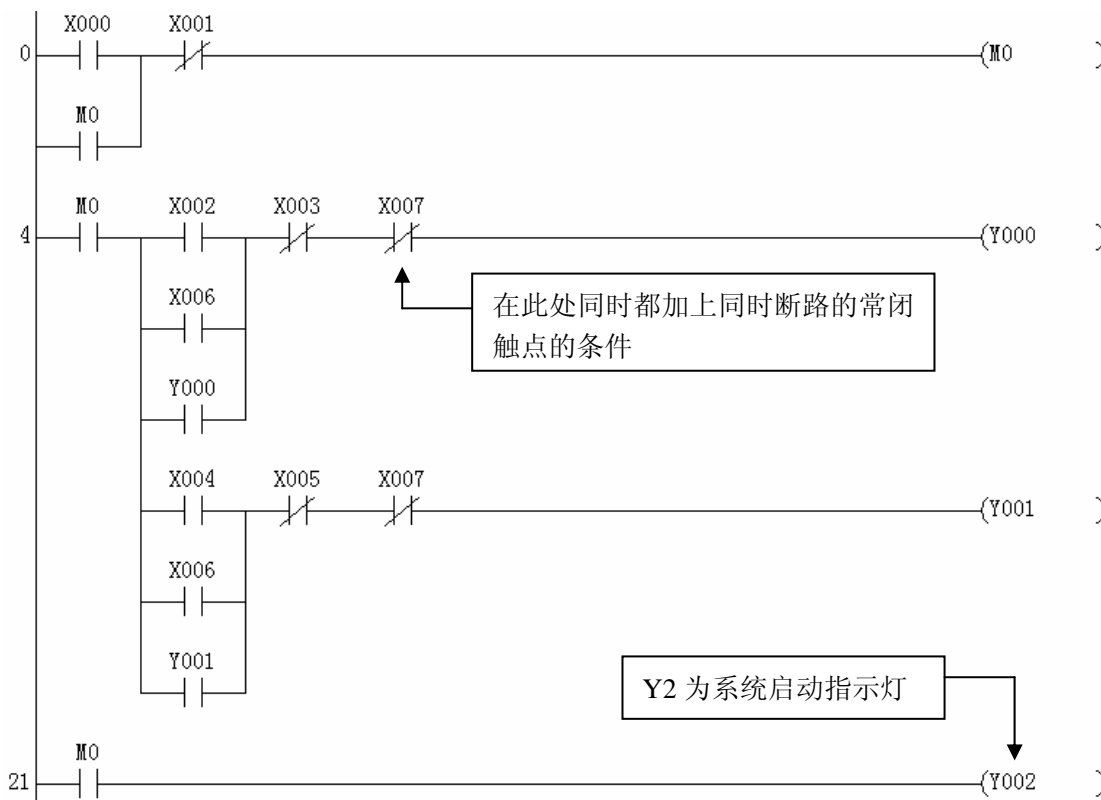
第三步, 编写按下两个电机同时启动和停止的梯形图, 可分小两步

1、按下 SB7 (接通 X6) 两个电机同时启动。实际上就是在 Y0 和 Y1 的接通触点处都加上 X6 的接通条件, 如下:

因为专业 · 所以优秀



2、按下 SB8 (X7) 两个电机同时停止。实际上就是在 Y0 和 Y1 的通路上都加上 X7 的常闭触点，当 SB8 一按下，X7 得电，由常闭导通变成断开，Y0 和 Y1 就同时失电，如下：



至此，所有程序的设计基本全部完成

从上面的程序设计过程来看

所有的程序都是由所有的小步骤一点一点组合起来的，程序设计时把整个过程思路理清，那再复杂的程序设计就也不会是难事

接下来，请把下面的要求在原来的程序上往上添加新功能

2、在上面的基础上再加上一个保护功能：当其中任一台电动机发生过载时（热继电器动作时），两台电动机均停止。

端子可分配：

X10: M1 电动机的热继电器过载

X11: M2 电动机的热继电器过载

注意：三菱 PLC 的输入端子没有 X8、X9，X7 后面紧跟着就是 X10；X17 后面就是 X20；依此类推。输出端子也是一样，Y7 后面紧跟着就是 Y10

3、请加上输出过载报警功能

当 M1 或 M2 有过载情况时，报警输出，等待处理，处理完毕以后，按下故障复位按钮（SB9），报警才会解除

端子可分配：

X12: 故障复位按钮

Y3: M1 电动机的热继电器过载报警输出

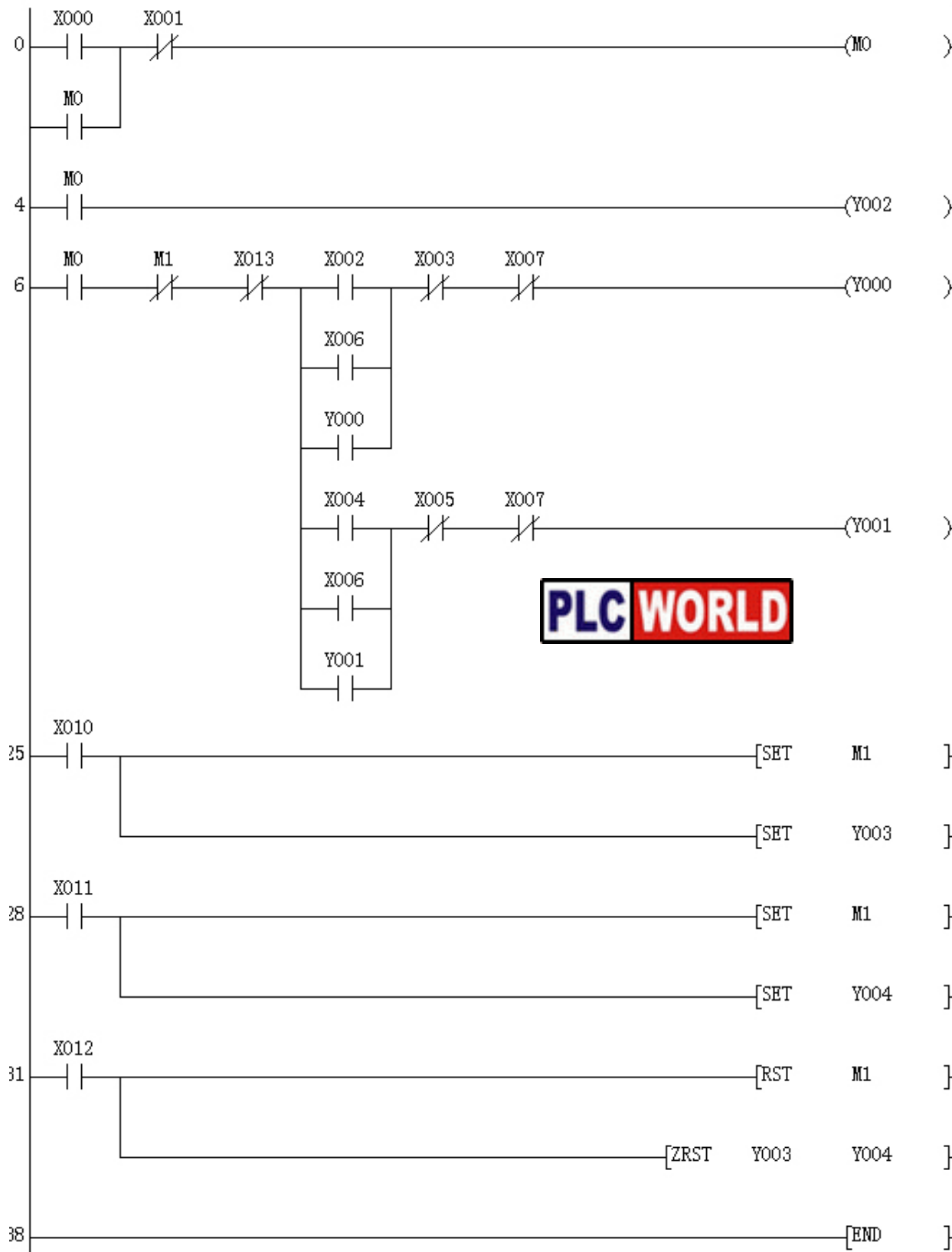
Y4: M2 电动机的热继电器过载报警输出

4、请再加入急停按钮：

按下急停时，M1 和 M2 都要停止，但系统不停止。

端子可分配：X13

请顺着上面的设计完成 2、3、4 的设计。下面是完整的程序



如果以上的程序，全部都用置位和复位指令去完成，应该怎么做？请思考，并独立编写程序

因为专业·所以优秀

练习

一、试设计一个工作台前进——退回的控制线路

工作台由电动机 M 拖动，行程开关 SQ1、SQ2 分别装在工作台的原位和终点。要求：

- 1) 有手动/自动功能
- 2) 自动时，按下启动按钮，实现前进—至终点—返回—停止到原位；
- 3) 手动时，工作台可以前后点动移动（有点动按钮）
- 4) 有原点复位功能
- 5) 有电机过载保护、并且报警功能
- 6) 有报警复位功能
- 7) 有急停功能。

请设计其电气方案图纸，及 PLC 编程，可以自由往上添加新功能。

可参照上面的例子

二、设计一个汽车库自动门控制系统

具体控制要求是：当汽车到达车库门前，超声波开关接收到车来的信号，开门上升，当升到顶点碰到上限开关，门停止上升，当汽车驶入车库后，光电开关发出信号，门电动机反转，门下降，当下降碰到下限开关后门电动机停止。

请设计其电气方案图纸，及 PLC 编程，可以自由往上添加新功能。

