

PLC 可编程控制器

2010 年 8 月

目录

1.可编程控制器（PLC）概述.....	2
2.可编程控制器基本组成.....	5
3.PLC 的工作原理和系统构成.....	8
4.PLC 的编程元件.....	10
5.PLC 的编程语言和基本指令.....	16
6.PLC 的基本指令.....	22
7.PLC 程序设计方法一.梯形图经验设计法.....	30
8.PLC 程序设计方法二.继电器控制电路移植法.....	35
9.PLC 程序设计方法三 顺序控制设计法之功能表图.....	37
10.PLC 程序设计方法三 顺序控制设计法之单序列结构的编程方法.....	38
11.PLC 程序设计方法三 顺序控制功能图的编程方法选择单序列结构的编程方法.....	42
12.PLC 程序设计方法三顺序控制功能图的编程方法之并行单序列结构的编程方法.....	45
13.PLC 功能指令.....	49
14.PLC 功能指令 传送与比较指令.....	51
15.PLC 功能指令 算术和逻辑运算指令.....	56
16.PLC 功能指令 循环移位与移位指令.....	59
17.PLC 功能指令、方便指令和外部 I/O 设备指令.....	62
18.可编程控制器控制系统设计.....	68
19.GPP 软件简介.....	75

1.可编程控制器（PLC）概述

一、PLC 的产生及其特点

实物展示：

三菱 PLC 外形图：



Q 系列 PLC



FX2N 系列



PLC FX1N 系列 PLC

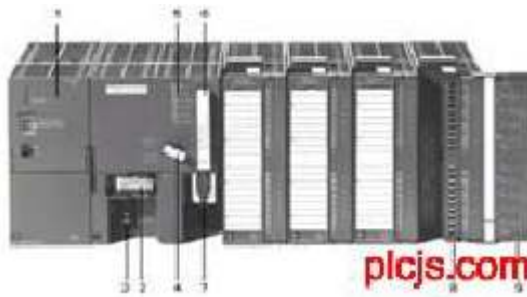


FX1SN 系列 PLC

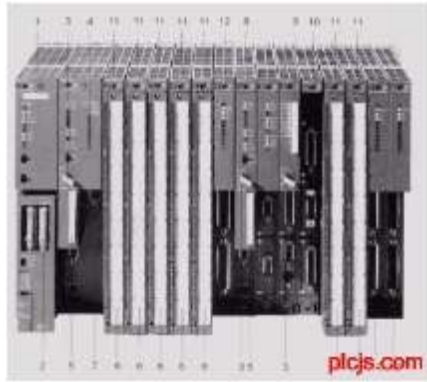
西门子 PLC 外形图：



S7-200 系列



PLC S7-300 系列 PLC



S7-400 系列 PLC

欧姆龙 PLC 外形图：



C200H 系列 PLC



CPM1A、CPM2A 系列 PLC

1、可编程控制器的名称演变

1969 年时被称为可编程逻辑控制器，简称 PLC (Programmable Logic Controller)。70 年代后期，随着微电子技术和计算机技术的迅猛发展，称其为可编程控制器，简称 PC (Programmable Controller)。但由于 PC 容易和个人计算机 (Personal Computer) 相混淆，故人们仍习惯地用 PLC 作为可编程控制器的缩写。

2、可编程控制器定义

(1987 年 国际电工委员会) 可编程序控制器是一种数字运算操作的电子系统，专为在工业环境下应用而设计。它采用了可编程序的存储器，用来在其内部存储执行逻辑运算、顺序控制、定时、计数和算术运算等操作的指令，并通过数字的，模拟的输入和输出，控制各种类型的机械或生产过程。

3、可编程控制器的产生

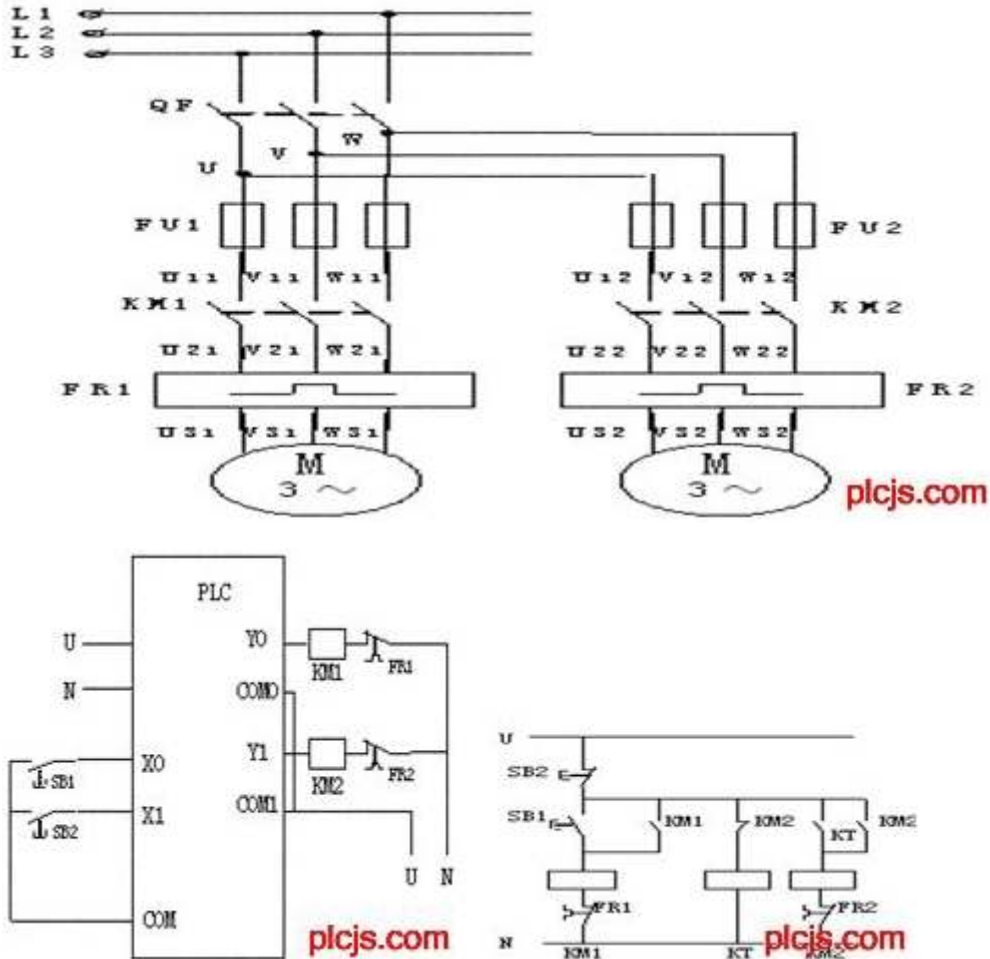
- 1) 1968 年，美国最大的汽车制造厂家通用汽车公司 (GM 公司) 提出设想。
- 2) 1969 年，美国数字设备公司研制出了世界上第一台 PC，型号为 PDP-14。
- 3) 第一代：从第一台可编程控制器诞生到 70 年代初期。其特点是：CPU 由中小规模集成电路组成，存储器为磁芯存储器；
- 4) 第二代：70 年代初期到 70 年代末期。其特点是：CPU 采用微处理器，存储器采用 EPROM ；
- 5) 第三代：70 年代末期到 80 年代中期。其特点是：CPU 采用 8 位和 16 位微处理器，有些还采用多微处理器结构，存储器采用 EPROM、EAROM、CMOSRAM 等 ；
- 6) 第四代：80 年代中期到 90 年代中期。PC 全面使用 8 位、16 位微处理芯片的位片式芯片，处理速度也达到 1 μ s/步 ；
- 7) 第五代：90 年代中期至今。PC 使用 16 位和 32 位的微处理器芯片，有的已使用 RISC 芯片。编程控制器运行演示展示板 PLC，时间继电器，继电器，直流电源，两个直流电动机，按钮，若干导线。

演示操作

可编程控制器控制电动机的顺序启动

方式一：按下启动按钮，由可编程控制器控制电动机 M1，M2 先后启动运行，按下停止按钮，两个电动机停止工作。

方式二：按下启动按钮，由可编程控制器控制电动机 M2，M1 先后启动运行，按下停止按钮，两个电动机停止工作。



问题提出：

传统的继电器接触控制系统，只能改变某些硬件接线，才能完成上述的两种控制方式，而可编程控制器控制系统可在不改变硬件接线的情况下，通过修改程序而实现控制顺序的变化。控制两个电动机的顺序运行，控制复杂程度不高，如用继电器接触控制系统已够费时的了，何况汽车生产流水线的控制系统？

二、可编程控制器的基本特点

- 1、灵活、实用
- 2、可靠性高、抗干扰能力强
- 3、编程简单、使用方便
- 4、接线简单
- 5、功能强
- 6、体积小、重量轻、易于实现自动化

三、可编程控制器的发展趋势

1. 向高速度、大存储容量方向发展 CPU 处理速度进一步加快，存储容量进一步扩大
2. 控制系统将分散化 分散控制、集中管理的原则。
3. 可靠性进一步提高 随着 PC 进入过程控制领域，对可靠性的要求进一步提高。硬件冗余的容错技术将进一步应用。
4. 控制与管理功能一体化 PC 将广泛采用计算机信息处理技术、网络通信技术和图形显示技术，使 PC 系统的生产控制功能和信息管理功能融为一体。

四、PC 的应用领域

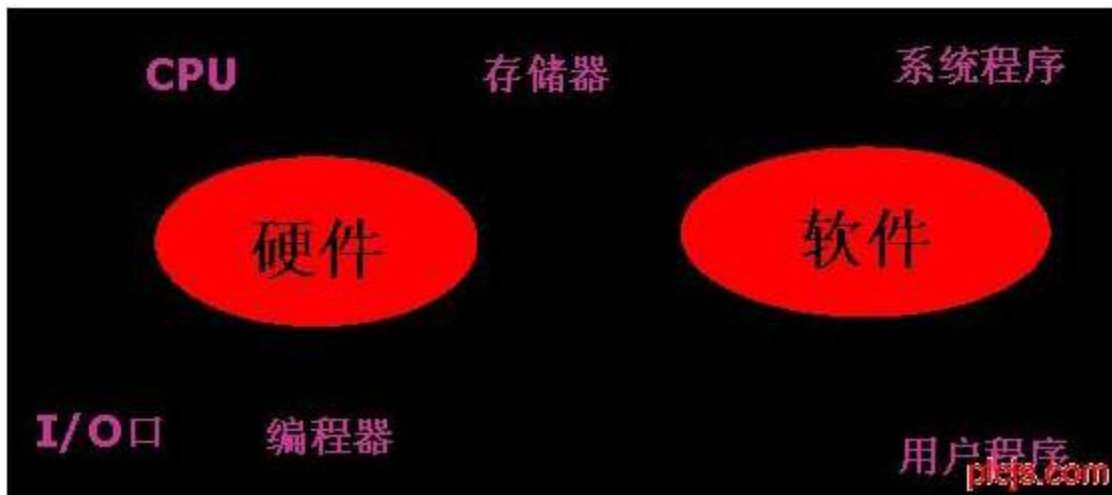
- 1、开关量逻辑控制
- 2、模拟量闭环控制
- 3、数据量的职能控制
- 4、数据采集与监控
- 5、通讯联网与集散控制

2.可编程控制器基本组成

问题：

- 1、可编程控制器由哪几部分组成？
- 2、可编程控制器怎样分类？

一、可编程控制器的组成



(一) 硬件构成

1、中央处理单元（CPU）

- (1)诊断 PLC 电源、内部电路的工作状态及编制程序中的语法错误。
- (2)采集现场的状态或数据，并送入 PLC 的寄存器中。
- (3)逐条读取指令，完成各种运算和操作。
- (4)将处理结果送至输出端。

(5)响应各种外部设备的工作请求。

2、存储器 (ROM/RAM)

(1) 系统程序存储器 (ROM) 用以存放系统管理程序、监控程序及系统内部数据, PLC 出厂前已将其固化在只读存储器 ROM 或 PROM 中, 用户不能更改。

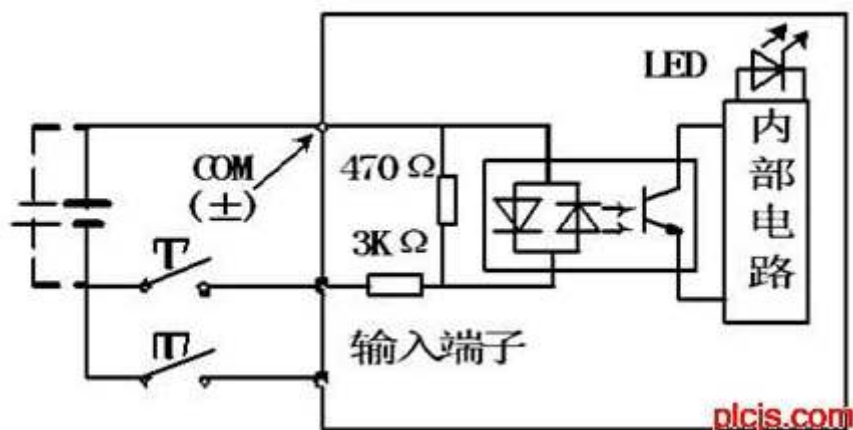
(2) 用户存储器 (RAM) 包括用户程序存储区和工作数据存储区。这类存储器一般由低功耗的 CMOS-RAM 构成, 其中的存储内容可读出并更改。掉电会丢失存储的内容, 一般用锂电池来保持。

注意: PLC 产品手册中给出的“存储器类型”和“程序容量”是针对用户程序存储器而言的

3、可编程控制器输入端口电路

开关量输入接口电路: 采用光电耦合电路, 将限位开关、手动开关、编码器等现场输入设备的控制信号转换成 CPU 所能接受和处理的数字信号。

PLC 的输入接口电路 (直流输入型)



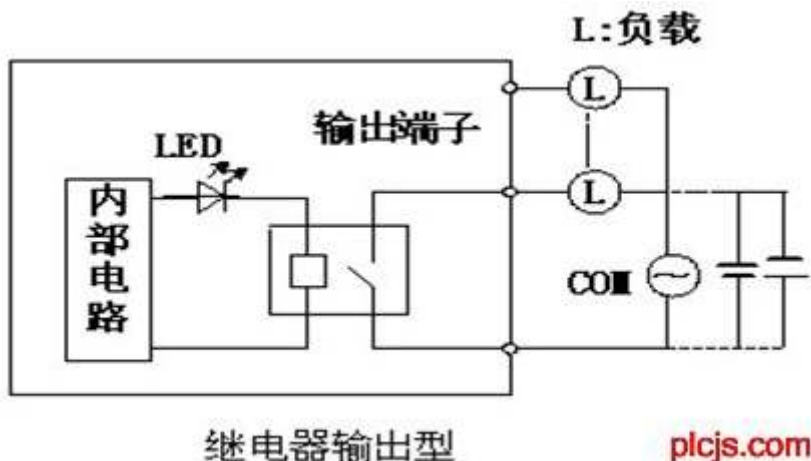
4、可编程控制器输出接口电路

开关量输出接口电路:

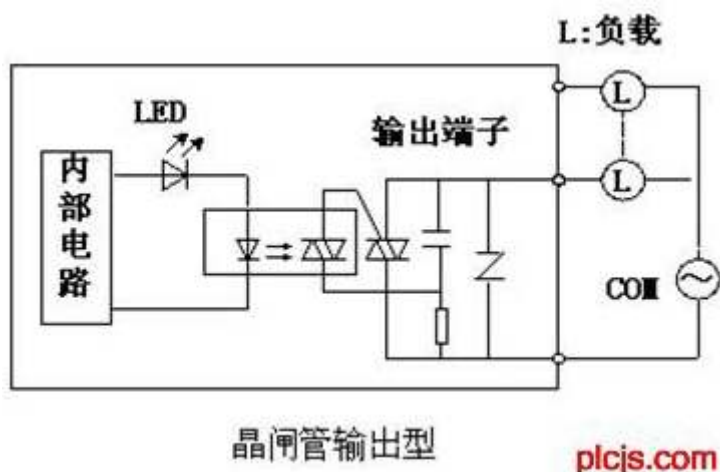
采用光电耦合电路, 将 CPU 处理过的信号转换成现场需要的强电信号输出, 以驱动接触器、电磁阀等外部设备的通断电。

有三种类型:

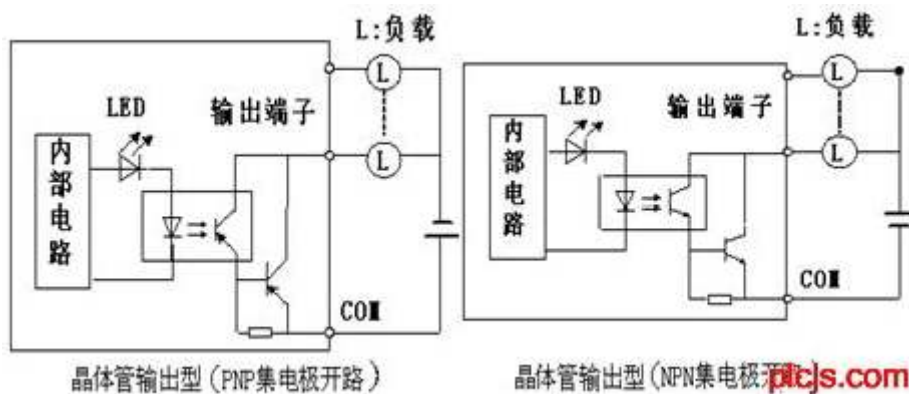
第一: 继电器输出型: 为有触点输出方式, 用于接通或断开开关频率较低的直流负载或交流负载回路。



第二：晶闸管输出型：为无触点输出方式，用于接通或断开开关频率较高的交流电源负载。



第三：晶体管输出型：为无触点输出方式，用于接通或断开开关频率较高的直流电源负载。



5、模拟量接口电路

(1) 模拟量输入接口：把现场连续变化的模拟量标准信号转换成适合 PLC 内部处理的有若干位二进制数字表示的信号标准

的模拟量信号：电流信号：4~20mA 电压信号：1~10V

(2) 模拟量输出接口：将 PLC 运算处理的若干位数字量信号转换为相应的模拟量信号输出，以满足生产过程现场连续控制的要求信号

(3) 智能输入输出接口：自带 CPU，由专门的处理能力，与主 CPU 配合共同完成控制任务，可减轻主 CPU 工作负担，又可提高系统的工作效率

6、电源

PLC 的电源是指将外部输入的交流电处理后转换成满足 PLC 的 CPU、存储器、输入输出接口等内部电路工作需要的直流电源电路或电源模块。许多 PLC 的直流电源采用直流开关稳压电源，不仅可提供多路独立的电压供内部电路使用，而且还可为输入设备（传感器）提供标准电源。

7、编程器

编程器：专用的手持式、台式；电脑+编程软件。

作用：编程，调试，监控

二、可编程控制器的分类

按硬件的结构类型分类：整体式、模块式、叠装式。

按 I/O 点数的多少分类：小型 PLC、中型 PLC、大型 PLC

3. PLC 的工作原理和系统构成

一、可编程控制器是如何工作的？

继电器控制系统：

硬逻辑并行运行的方式

计算机控制系统：

采用等待命令的工作方式，如键盘扫描方式或 I/O 扫描方式

可编程控制器控制系统：

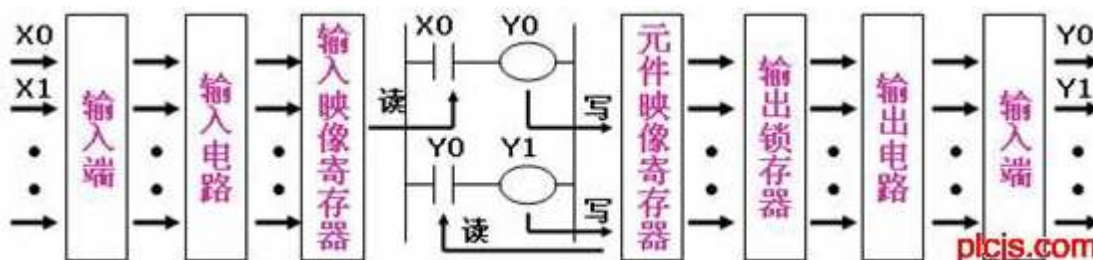
循环扫描工作方式，即系统工作任务管理及应用程序执行都是按循环扫描方式完成的

二、可编程控制器工作原理



可编程控制器在开机后，完成内部处理、通信处理、输入刷新、程序执行、输出刷新五个工作阶段，称为一个扫描周期。完成一次扫描后，又重新执行上述过程，可编程控制器这种周而复始的循环工作方式称为扫描工作方式。

1、信号传递过程(从输入到输出)



最终输出刷新：将输出映像寄存器的状态写入输出锁存电路，再经输出电路传递输出端子，从而控制外接器件动作。

2、扫描周期和 I/O 滞后时间

可编程控制器在运行工作状态时，执行一次扫描操作所需要的时间称为扫描周期。其典型值为 1~100ms。

I/O 滞后时间又称为系统响应时间，是指可编程控制器外部输入信号发生变化的时刻起至它控制的有关外部输出信号发生变化的时刻之间的间隔。

I/O 滞后现象的原因

- (1) 输入滤波器有时间常数
- (2) 输出继电器有机械滞后
- (3) PC 循环操作时，进行公共处理、I/O 刷新和执行用户程序等产生扫描周期
- (4) 程序语句的安排，也影响响应时间

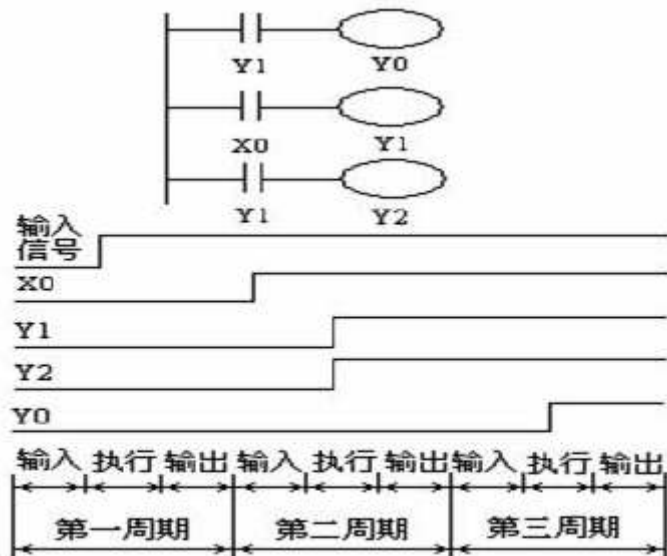
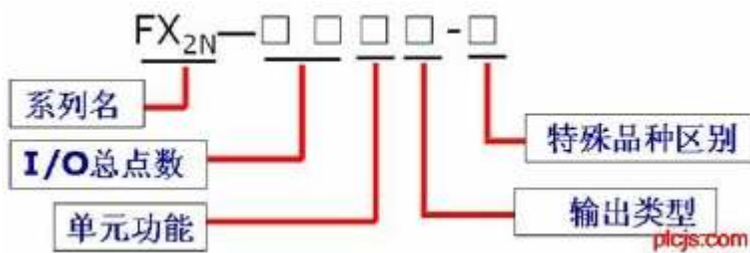


图 2.4 PLC 的 I/O 延迟 plcjs.com

三、可编程控制器的系统配置

1、FX2N 系列可编程控制器型号格式如下：



- M: 基本单元
- E: 输入输出混合扩展单元与扩展模块
- EX: 输入专用扩展模块
- EY: 输出专用扩展模块
- R: 继电器输出
- S: 晶体管输出
- T: 晶闸管输出

2、FX2N 系列可编程控制器系统配置

FX2N 是 FX 系列中功能最强、速度最高的微型可编程控制器。它的基本指令执行时间高达 0.08s，远远超过了很多大型可编程控制器。用户存储器容量可扩展到 16K 步，最大可以扩展到 256 个 I/O 点，有 5 种模拟量输入/输出模块、高速计数器模块、脉冲输出模块、4 种位置控制模块、多种 RS-232C/RS-422/RS-485 串行通信模块或功能扩展板，以及模拟定时器功能扩展板，使用特殊功能模块和功能扩展板，可以实现模拟量控制、位置控制和联网通信等功能。

3、可编程控制器的技术性能指标

- 1.输入/输出点数
- 2.存储容量
- 3.扫描速度
- 4.指令系统
- 5.可扩展性
- 6.通信功能

4.PLC 的编程元件

FX2N 系列 PLC 编程元件分类和编号

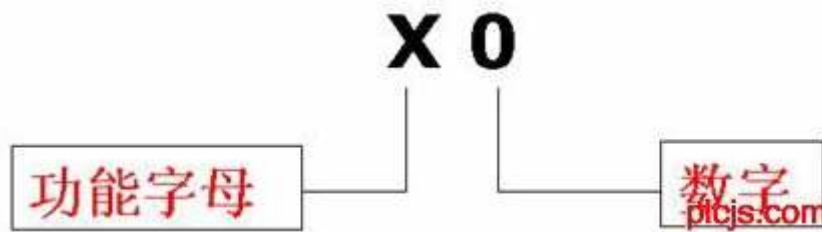
编程元件的基本特征

编程元件的使用

一、FX2N 系列 PLC 编程元件分类和编号

1、PLC 编程元件的物理实质：

电子电路及存储器。称“软继电器”



2、编程元件的基本特征

相同点	不同点
都具有线圈和常开常闭触点，触点的状态随着线圈的状态而变化，即当线圈被选中（通电）时，常开触点闭合，常闭触点断开，当线圈失去选中条件时，常闭接通，常开断开。	编程元件被选中，只是代表这个元件的存储单元置 1，失去选中条件只是这个元件的存储单元置 0；编程元件可以无限次地访问，可编程控制器的编程元件可以有无数多个常开、常闭触点。 plcjs.com

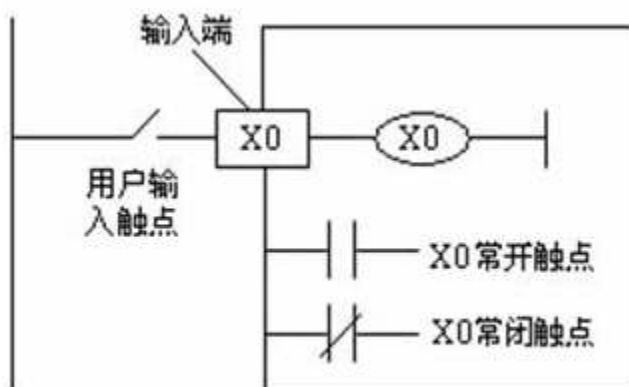
3、可编程控制器的编程元件

输入继电器 X	计数器 C
输出继电器 Y	数据寄存器 D
辅助继电器 M	变址寄存器 V/Z
状态器 S	指针 P/I
定时器 T	常数 (K/H)

plcjs.com

二、输入继电器 X

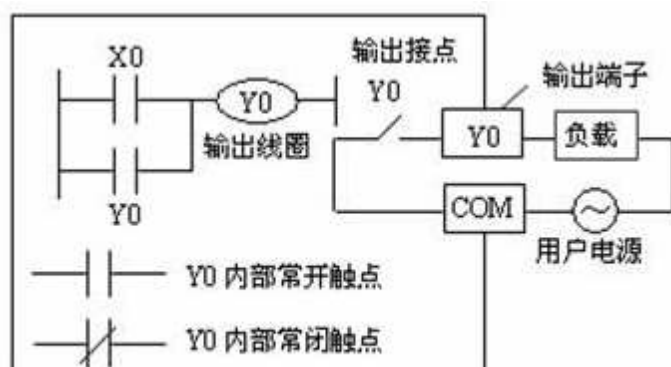
可编程控制器输入接口的一个接线点对应一个输入继电器。输入继电器的线圈只能由机外信号驱动，它可提供无数个常开触点、常闭触点供编程时使用。如图 3.1。FX2N 系列的输入继电器采用八进制地址编号，X0~X267 最多可达 184 点。



三、输出继电器 Y

PLC 输出接口的一个接线点对应一个输出继电器。输出继电器的线圈只能由程序驱动，每个输出继电器除了为内部控制电路提供编程用的常开、常闭触点外，还为输出电路提供一个常开触点与输出接线端连接。驱动外部负载的电源由用户提供。

如图 3.2 所示是输出继电器的等效电路。输出继电器的地址编号也是八进制，Y0~Y267，最多可达 184 点。



四、辅助继电器 M

PLC 内部有很多辅助继电器，和输出继电器一样，只能由程序驱动，每个辅助继电器也有无数对常开、常闭

接点供编程使用。其作用相当于继电器控制线路中的中间继电器。辅助继电器的接点在 PLC 内部编程时可以任意使用，但它不能直接驱动负载，外部负载必须由输出继电器的输出接点来驱动。

辅助继电器 M 分类

辅助继电器分以下三种类型：

通用辅助继电器

M0-M499，共 500 个点

断电保持辅助继电器

M500-M1023 及 M1024-M3071 共 2572 点。

特殊辅助继电器

M8000-M8255，共 256 个点。

特殊辅助继电器：

① 只能利用其接点的特殊辅助继电器。线圈由 PLC 自动驱动，用户只可以利用其接点。例如：

M8000 为运行监控用，PLC 运行时 M8000 接通。

M8002 为仅在运行开始瞬间接通的初始脉冲特殊辅助继电器。

② 可驱动线圈型特殊辅助继电器。用户激励线圈后，PLC 作特定动作。例如：

M8033 为 PLC 停止时输出保持特殊辅助继电器。

M8034 为禁止全部输出特殊辅助继电器。

M8039 为定时扫描特殊辅助继电器。

五、状态器 S

状态器 S 是构成状态转移图的重要软元件，它与后续的步进梯形指令配合使用。通常状态继电器软元件有下面五种类型

初始状态继电器 S0~S9 共 10 点。

回零状态继电器 S10~S19 共 10 点。

通用状态继电器 S20~S499 共 480 点。

停电保持状态器 S500~S899 共 400 点。

报警用状态继电器 S900~S999 共 100 点。

六、定时器 T

定时器作为时间元件相当于时间继电器，由设定值寄存器、当前值寄存器和定时器触点组成。在其当前值寄存器的值等于设定值寄存器的值时，定时器触点动作。故设定值、当前值和定时器触点是定时器的三要素。

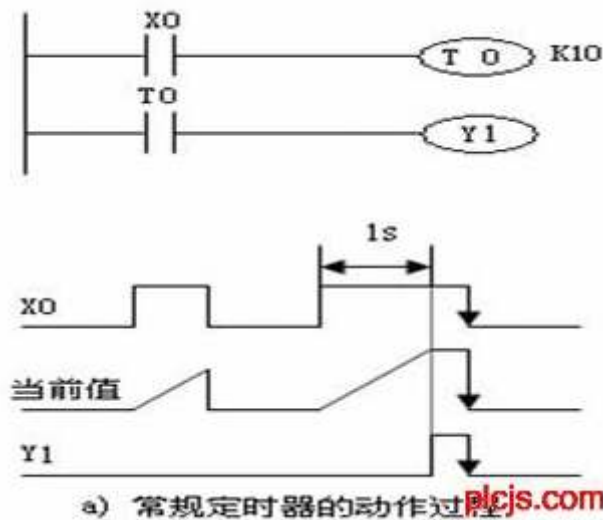
定时器累计 PLC 内的 1ms, 10ms, 100ms 等的时钟脉冲，当达到所定的设定值时，输出接点动作。定时器可以使用用户程序存储器内的常数 K 作为设定值，也可以用后述的数据寄存器 D 的内容作为设定值。这里的数据寄存器应有断电保持功能。

定时器可以分为：

常规定时器 T0~T245

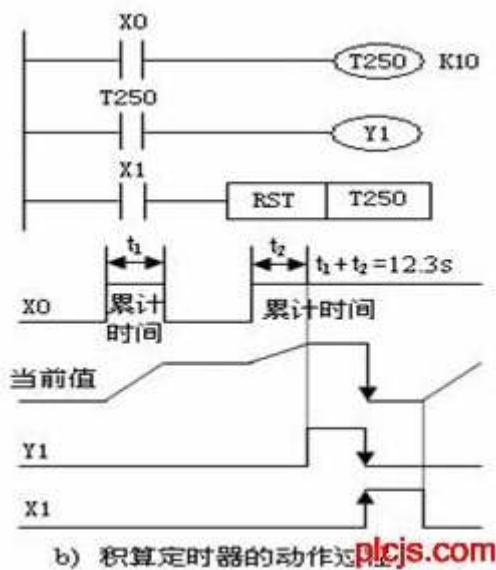
积算定时器 T246~T255

1、常规定时器的动作过程



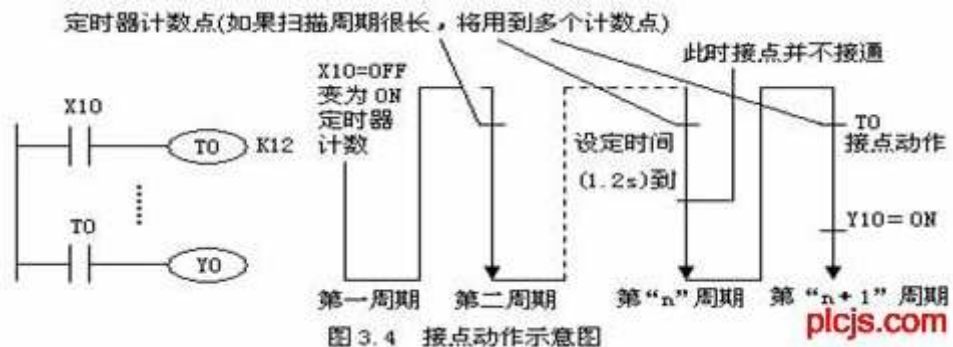
2、积算定时器 T246~T255

1ms 积算定时器 T246~T249 共四点，每点设定值范围 0.001s~32.767s；100ms 积算定时器 T250~T255 共 6 点，每点设定值范围 0.1s~3276.7s。如图所示，当定时器线圈 T250 的驱动输入 X1 接通时 T250 用当前值计数器累计 100ms 的时钟脉冲个数，当该值与设定值 K10 相等时，定时器的输出接点输出，当计数中间驱动输入 X0 断开或停电时，当前值可保持。输入 X1 再接通或复电时，计数继续进行，当累计时间为 $10 \times 0.1s = 1s$ 时，输出接点动作。当复位输入 X1 接通时，计数器就复位，输出接点也复位



3、接点的动作时序

接点动作时序如图所示。定时器在其线圈被驱动后开始计时，到达设定值后，在执行第一个线圈指令时，其输出接点动作。从驱动定时器线圈到其接点动作称为定时器接点动作精度时间 t ， $t = T + T0 - \alpha$ 。

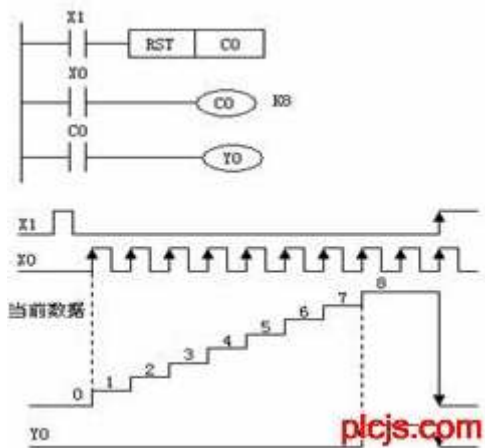


七、计数器 C

可编程控制器的计数器共有两种：内部信号计数器和高速计数器。内部信号计数器有分为两种：16 位递加计数器和 32 位增减计数器。

1、16 位递加计数器

设定值位 1~32767。其中，C0~C99 共 100 点是通用型，C100~C199 共 100 点是断电保持型。下图表示了递加计数器的动作过程。

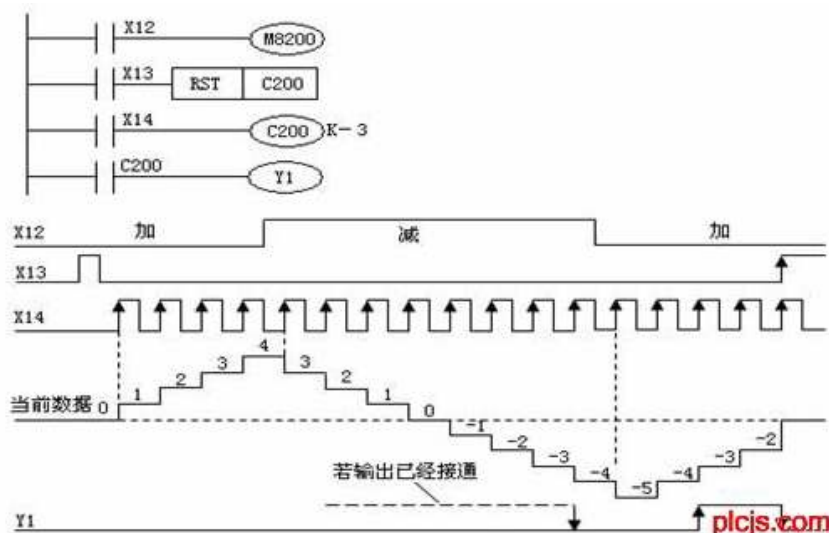


2、32 位增减计数器

设定值为-2147483648~+2147483647，其中 C200~C219 共 20 点是通用型，C220~C234 共 15 点为断电保持型计数器。

32 位双向计数器是递加型还是递减型计数由特殊辅助继电器 M8200~M8234 设定。特殊辅助继电器接通时（置 1）时，为递减计数；特殊辅助继电器断开（置 0）时，为递加计数。可直接用常数 K 或间接用数据寄存器 D 的内容作为设定值。间接设定时，要用器件号紧连在一起的两个数据寄存器。如图所示，用 X14 作为计数输入，驱动 C200 计数器线圈进行计数操作。当计数器的当前值由-4 到-3（增大）时，其接点接通（置 1）；当计数器的当前值由-3 到-4（减小）时，其接点

断开（置 0）。



八、数据寄存器 D

在进行输入输出处理、模拟量控制、位置控制时，需要许多数据寄存器存储数据和参数。数据寄存器为 16 位，最高位

为符号位，可用两个数据寄存器合并起来存放 32 位数据，最高位仍为符号位。

数据寄存器分成下面几类：

通用数据寄存器 D0~D199 共 200 点

断电保持/锁存寄存器 D200~D7999 共 7800 点

特殊数据寄存器 D8000~D8255 共 256 点

文件数据寄存器 D1000~D7999 共 7000 点

1、通用数据寄存器 D0~D199 共 200 点

一旦在数据寄存器写入数据，只有不再写入其他数据，就不会变化。但是当 PLC 由运行到停止或断电时，该类数据寄存

器的数据被清除为 0。但是当特殊辅助继电器 M8033 置 1，PLC 由运行转向停止时，数据可以保持。

2、断电保持/锁存寄存器 D200~D7999 共 7800 点

断电保持/锁存寄存器有断电保持功能，PLC 从 RUN 状态进入 STOP 状态时，断电保持寄存器的值保持不变。利用参数设

定，可改变断电保持的数据寄存器的范围。

3、特殊数据寄存器 D8000~D8255 共 256 点

这些数据寄存器供监视 PLC 中器件运行方式用。其内容在电源接通时，写入初始值（先全部清 0，然后由系统 ROM 安排

写入初始值）。例如，D8000 所存的警戒监视时钟的时间由系统 ROM 设定。若有改变时，用传送指令将目的的时间送入 D8000。

该值在 PLC 由 RUN 状态到 STOP 状态保持不变。未定义的特殊数据寄存器，用户不能用。

4、文件数据寄存器 D1000~D7999 共 7000 点

文件寄存器是以 500 点为一个单位，可被外部设备存取。文件寄存器实际上被设置为 PLC 的参数区。文件

寄存器与锁存

寄存器是重叠的，可保证数据不会丢失。FX2N 系列的文件寄存器可通过 BMOV（块传送）指令改写

九、变址寄存器（V/Z）

变址寄存器除了和普通的数据寄存器有相同的使用方法外，还常用于修改器件的地址编号。V、Z 都是 16 位的寄存器，可进行数据的读写。当进行 32 位操作时，将 V、Z 合并使用，指定 Z 为低位

十、指针（P/I）

分支指令用 P0~P62、P64~P127 共 127 点。指针 P0~P62、P64~P127 为标号，用来指定条件跳转，子程序调用

等分支指令的跳转目标。P63 为结束跳转用。中断用指针 I0□□~I8□□共 9 点。中断指针的格式表示如下：

(1) 输入中断 IΔ0□

□=0 表示为下降沿中断；□=1 表示为上升沿中断。

Δ表示输入号，取值范围为 0~5，每个输入只能用一次。

例如，I001 为输入 X0 从 OFF 到 ON 变化时，执行由该指令作为标号后面的中断程序，并根据 IRET 指令返回。

(2) 定时器中断 IΔ□□

Δ表示定时器中断号，取值范围为 6~8，每个定时器只能用 1 次。

□表示定时时间，取值范围为 10~99ms。

例如，I710，即每隔 10ms 就执行标号为 I710 后面的中断程序，并根据 IRET 指令返回。

十一、常数（K/H）

常数也作为器件对待，它在存储器中占有一定的空间，十进制常数用 K 表示，如 18 表示为 K18；十六进制常数用 H 表示，如 18 表示为 H12。

5. PLC 的编程语言和基本指令

一、可编程控制器的编程语言

不同厂家，不同型号的 PLC 的编程语言只能适应自己的产品。IEC 中的 PLC 编程语言标准中有五种编程语言：顺序功能

图编程语言、梯形图编程语言、功能块图编程语言、指令语句表编程语言、结构文本编程语言。最常用的就是梯形图编程语言和指令语句表编程语言。

1、梯形图编程语言

梯形图是在原继电器—接触器控制系统的继电器梯形图基础上演变而来的一种图形语言。它是目前用得最多的 PLC 编程语言。

注意：梯形图表示的并不是一个实际电路而只是一个控制程序，其间的连线表示的是它们之间的逻辑关系，即所谓“软接线”。

常开触点：

常闭触点：

线圈：

注意：它们并非是物理实体，而是“软继电器”。每个“软继电器”仅对应 PLC 存储单元中的一位。该位状态为“1”时，对应的继电器线圈接通，其常开触点闭合、常闭触点断开；状态为“0”时，对应的继电器线圈不通，其常开、常闭触点保持原态。

2、梯形图编程格式

(1) 梯形图按行从上至下编写，每一行从左往右顺序编写。PLC 程序执行顺序与梯形图的编写顺序一致。

(2) 图左、右边垂直线称为起始母线、终止母线。每一逻辑行必须从起始母线开始画起，终止于继电器线圈或终止母线（有些 PLC 终止母线可以省略）。

(3) 梯形图的起始母线与线圈之间一定要有触点，而线圈与终止母线之间则不能有任何触点。

3、指令语句表编程语言

助记符语言类似于计算机汇编语言，用一些简洁易记的文字符号表达 PLC 的各种指令。同一厂家的 PLC 产品，其助记符语言与梯形图语言是相互对应的，可互相转换。

助记符语言常用于手持编程器中，梯形图语言则多用于计算机编程环境中。

案例

在生产实践过程中，某些生产机械常要求既能正常起动，又能实现调整位置的点动工作。试用可编程控制器的基本逻辑指令来控制电动机的点动及连续运行。

1、异步电动机控制线路图

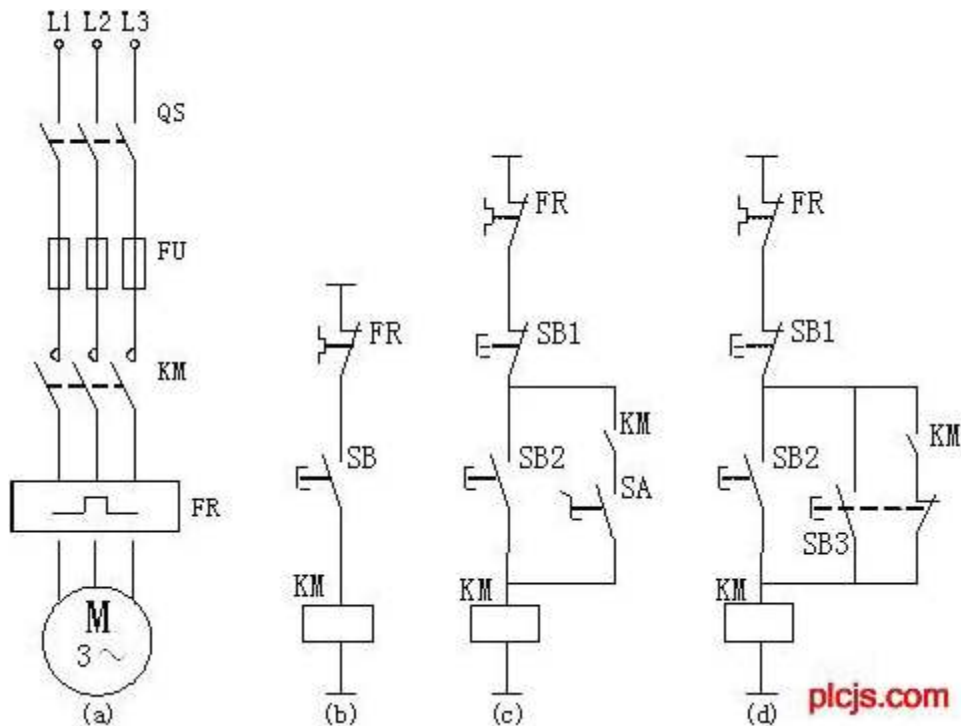


图 (a) 为主电路。工作时，合上刀开关 QS，三相交流电经过 QS，熔断器 FU，接触器 KM 主触点，热继电器 FR 至三相交流电动机。

图 (b) 为最简单的点动控制线路。起动按钮 SB 没有并联接触器 KM 的自锁触点，按下 SB，KM 线圈通电，松开按钮 SB 时，接触器 KM 线圈又失电，其主触点断开，电动机停止运转。

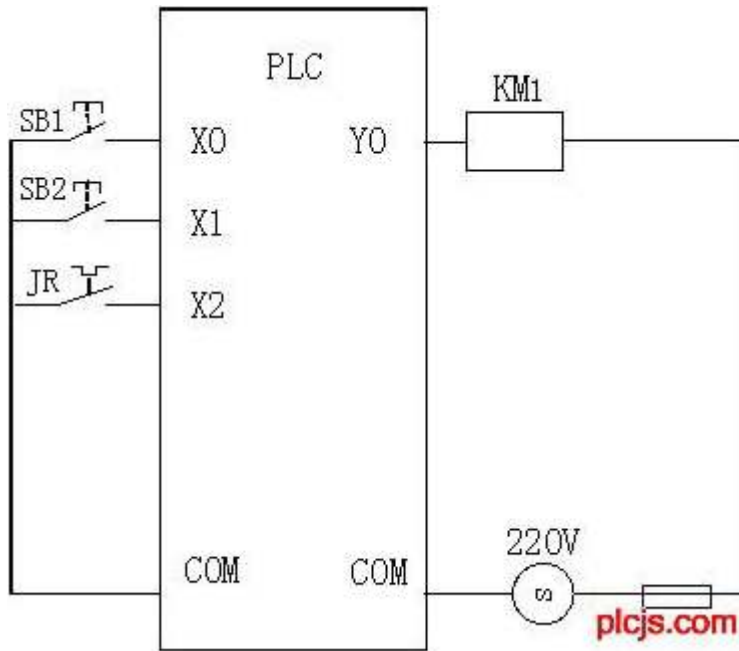
图 (c) 是带手动开关 SA 的点动控制线路。当需要点动控制时，只要把开关 SA 断开，由按钮 SB 2 来进行点动控制。当需要正常运行时，只要把开关 SA 合上，将 KM 的自锁触点接入，即可实现连续控制。

图 (d) 中增加了一个复合按钮 SB 3 来实现点动控制。需要点动运行时，按下 SB 3 点动按钮，其常闭触点先断开自锁电路，常开触发后闭合接通起动控制电路，KM 接触器线圈得电，主触点闭合，接通三相电源，电动机起动运转。当松开点动按钮 SB 3 时，KM 线圈失电，KM 主触点断开，电动机停止运转。

若需要电动机连续运转，由停止按钮 SB 1 及起动按钮 SB 2 控制，接触器 KM 的辅助触点起自锁作用。

2、可编程控制器的硬件连接

实现电动机的点动及连续运行所需的器件有：启动按钮 SB1，停止按钮 SB2，交流接触器 KM，热继电器 FR 及刀开关 QS 等。主电路的连接如图所示。



3、梯形图的设计

梯形图便是以图形符号及图形符号在图中的相互关系表示控制关系的编程语言，是从继电器电路图演变而来。两者部分符号对应关系如表所示。

符号名称	继电器电路符号	梯形图符号
常开触点		
常闭触点		
线圈		

根据输入输出接线圈可设计出异步电动机点动运行的梯形图如图 (a) 所示。工作过程分析如下：当按下 SB1 时，输入继电器 X0 得电，其常开触点闭合，因为异步电动机未过热，热继电器常开触点不闭合，输入继电器 X2 不接通，其常闭触点保持闭合，则此时输出继电器 Y0 接通，进而接触器 KM 得电，其主触点接通电动机的电源，则电动机起动运行。当松开按钮 SB1 时，X0 失电，其触点断开，Y0 失电，接触点 KM 断电，电动机停止转动，即本梯形图可实现点动控制功能。大家可能发现，在梯形图中使用的热继电器的触点为常开触点，如果要使用常闭触点，梯形图应如何设计？

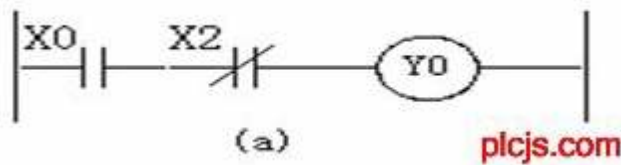
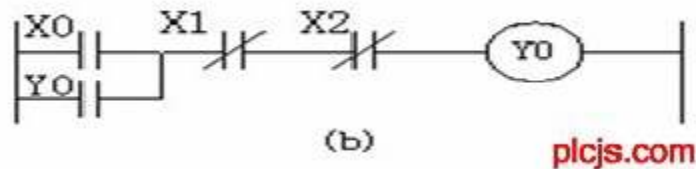


图 (b) 为电动机连续运行的梯形图，其工作过程分析如下：当按 SB1 被按下时 X0 接通，Y0 置 1，这时电动机连续运行。需要停车时，按下停车按钮 SB2，串联于 Y0 线圈回路中的 X1 的常闭触点断开，Y0 置 1，电机失电停车。



起--保--停电路

梯形图 (b) 称为启-保-停电路。这个名称主要来源于图中的自保持触点 Y0。并联在 X0 常开触点上的 Y0 常开触点的作用是当按钮 SB 1 松开，输入继电器 X0 断开时，线圈 Y0 仍然能保持接通状态。工程中把这个触点叫做“自保持触点”。

启-保-停电路是梯形图中最典型的单元，它包含了梯形图程序的全部要素。它们是：

- a、事件 每一个梯形图支路都针对一个事件。事件输出线圈（或功能框）表示，本例中为 Y0。
- b、事件发生的条件 梯形图支路中除了线圈外还有触点的组合，使线圈置 1 的条件既是事件发生的条件，本例中为起动按钮 X0 置 1。
- c、事件得以延续的条件 触点组合中使线圈置 1 得以持久的条件。本例中为与 X0 并联的 Y0 的自保持触点。
- d、使事件终止的条件 触点组合中使线圈置 1 中断的条件。本例中为 X1 的常闭触点断开。

4、语句表

点动控制即图 (a) 所使用到的基本指令有：从母线取用常开触点指令 LD；常闭触点的串联指令 ANI；输出继电器的线圈驱动指令 OUT。而每条指令占用一个程序步，语句表如下

程序步	指令	元件
0	LD	X0
1	ANI	X1
2	OUT	Y0

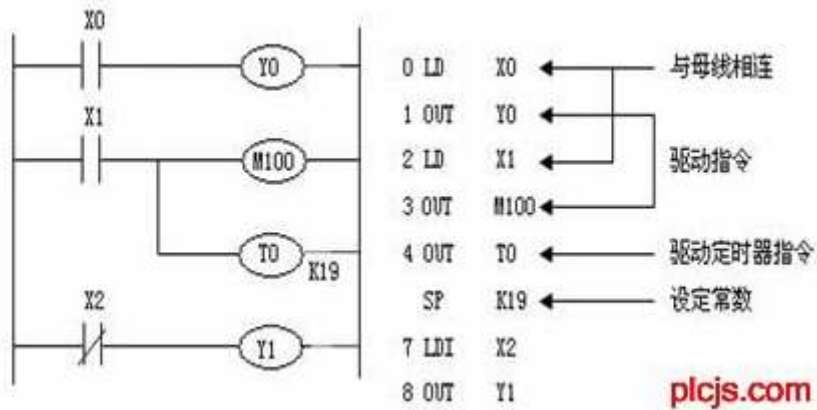
连续运行控制即图 (b) 所使用到的基本指令有：从母线取用常开触点指令 LD；常开触点的并联指令 OR；常闭触点的串联指令 ANI；输出继电器的线圈驱动指令 OUT。语句表如下：

程序步	指令	元件
0	LD	X0
1	OR	Y0
2	ANI	X1
3	ANI	X2
4	OUT	Y0

二、FX 系列可编程控制器的基本指令

FX2N 系列 PLC 共有 27 条基本指令，供设计者编制语句表使用，它与梯形图有严格的对应关系。

1、逻辑取及线圈驱动指令 LD、LDI、OUT

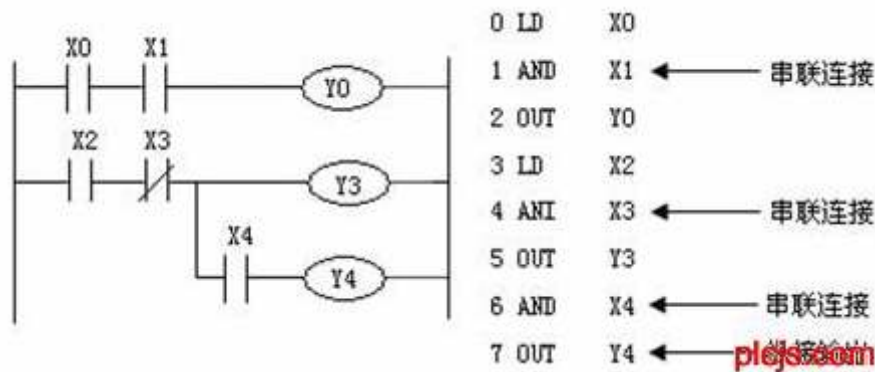


LD, 取指令。表示一个与输入母线相连的常开接点指令。

LDI, 取反指令。表示一个与输入母线相连的常闭接点指令。

OUT, 线圈驱动指令

2、接点串联指令 AND、ANI

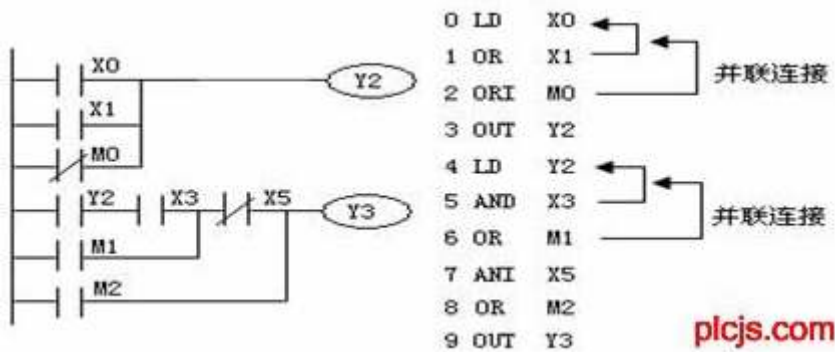


AND, 与指令。用于单个常开接点的串联。

ANI, 与非指令。用于单个常闭接点的串联。

OUT 指令, 通过接点对其它线圈使用 OUT 指令称为纵接输出或连续输出。

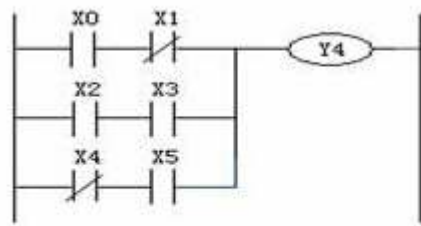
3、接点并联指令 OR、ORI



OR, 或指令, 用于单个常开接点的并联。

ORI, 或非指令, 用于单个常闭接点的并联。

4、串联电路块的并联连接指令 ORB

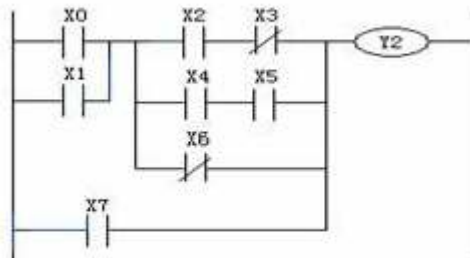


0 LD	X0	0 LD	X0
1 ANI	X1	1 ANI	X1
2 LD	X2	2 LD	X2
3 AND	X3	3 AND	X3
4 ORB		4 LDI	X4
5 LDI	X4	5 AND	X5
6 AND	X5	6 ORB	
7 ORB		7 ORB	
8 OUT	Y4	8 OUT	Y4

a) 语句表一 b) plcjs.com

两个或两个以上的接点串联连接的电路叫串联电路块。串联电路块并联连接时，分支开始用 LD、LDI 指令，分支结果用 ORB 指令。

5、并联电路块的串联连接指令 ANB



0 LD	X0	
1 OR	X1	
2 LD	X2	← 分支起点
3 ANI	X3	
4 LD	X4	← 分支起点
5 AND	X5	
6 ORB		← 串联电路块并联
7 ORI	X6	
8 ANB		← 与前面电路块串联
9 OR	X7	
10 OUT	Y2	

plcjs.com

两个或两个以上接点并联的电路称为并联电路块，分支电路并联电路块与前面电路串联连接时，使用 ANB 指令。分支的

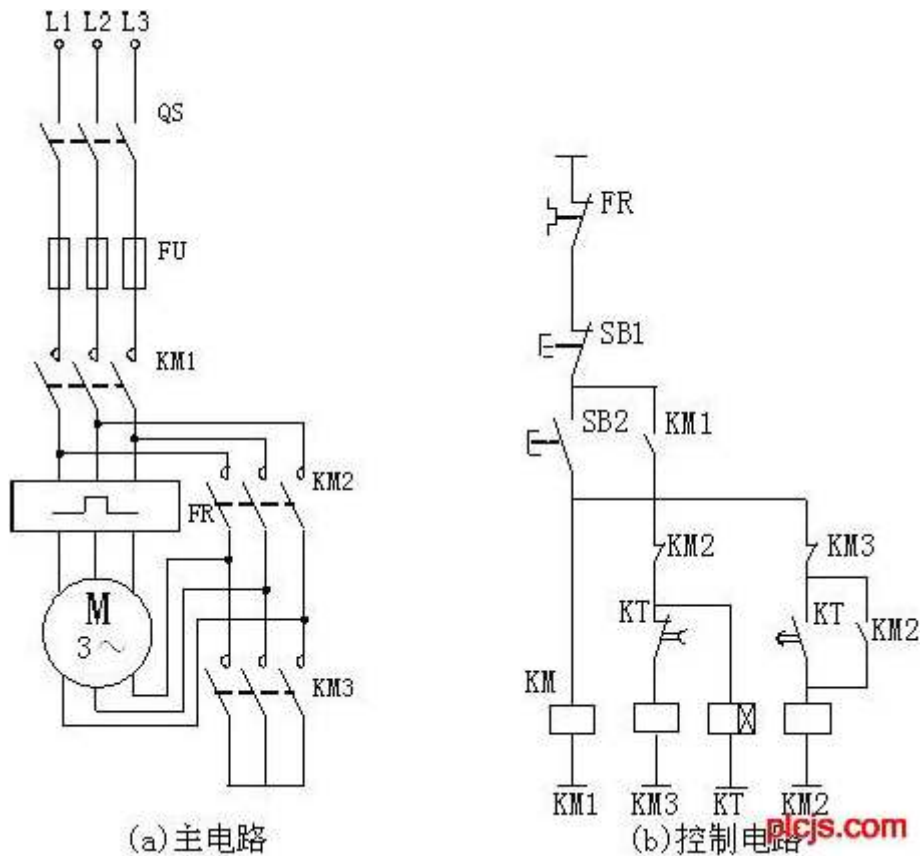
起点用 LD、LDI 指令。

6. PLC 的基本指令

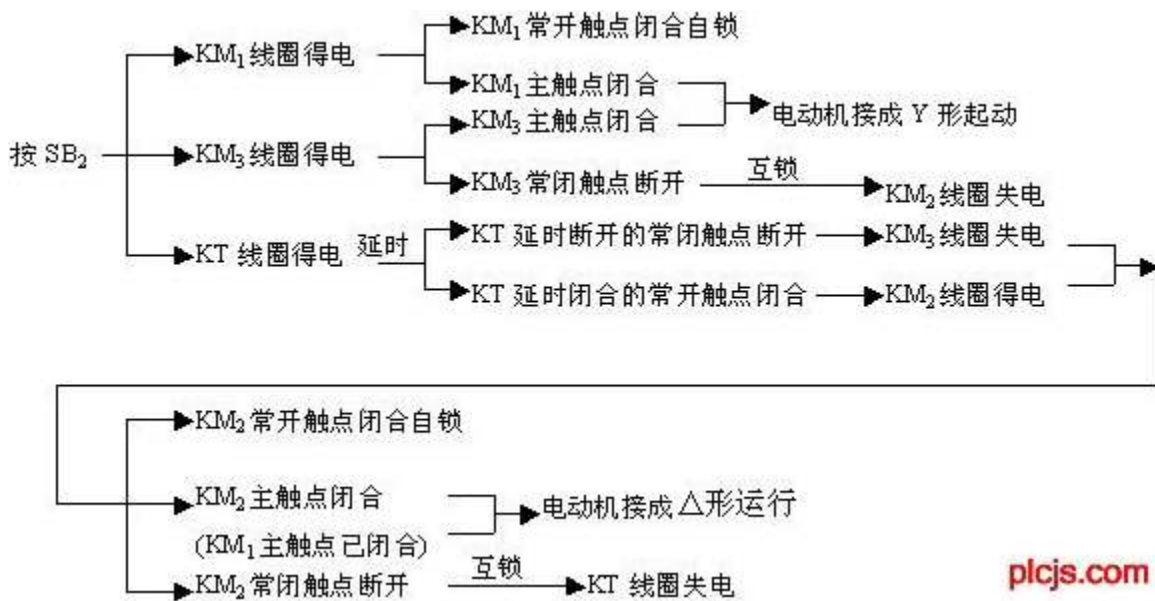
案例：

由电机及拖动基础可知，三相交流异步电动机起动时电流较大，一般是额定电流的（5 ~ 7）倍。故对于功率较大的电动机，应采用降压起动方式，Y/Δ降压起动是常用的方法之一。起动时，定子绕组首先接成星形，待转速上升到接近额定转速时，再将定子绕组的接线换成三角形，电动机便进入全电压正常运行状态。

异步电动机 Y/Δ降压起动控制电路



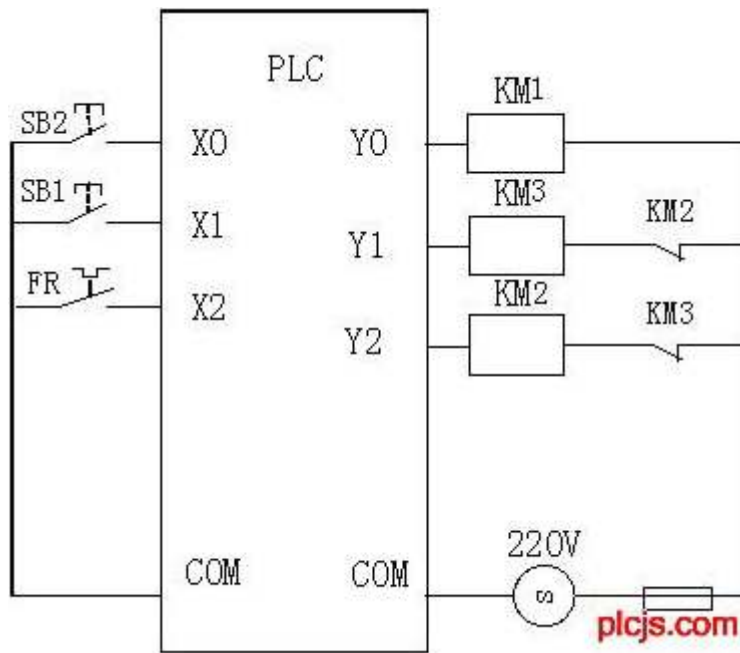
工作过程分析如下：



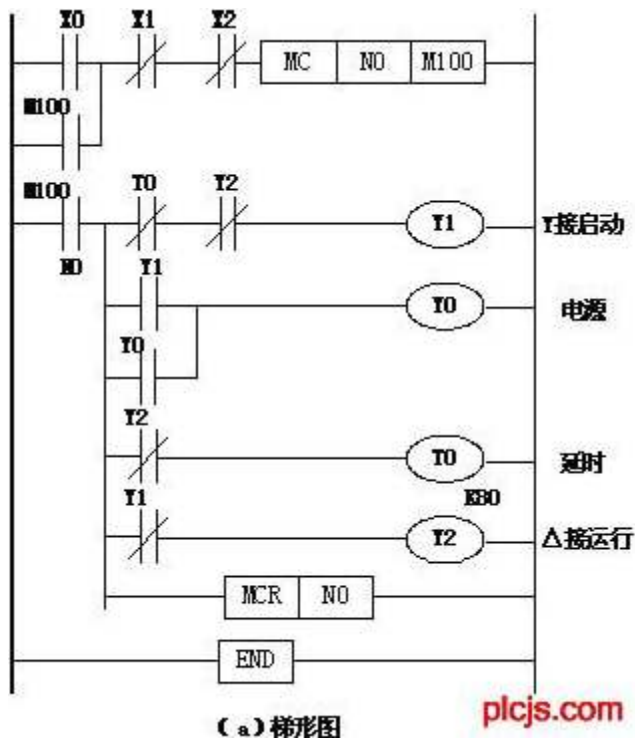
可编程控制器的硬件连接

本模块所需的硬件及输入 / 输出端口分配如图所示。由图可见：本模块除可编程控制器之外，还增添了部分器件，其中，SB1 为停止按钮，SB2 为启动按钮，FR 为热继电器的常开触点，KM1 为主电源接触器，KM2 为

Δ 形运行接触器，KM3 为 Y 形启动接触器。

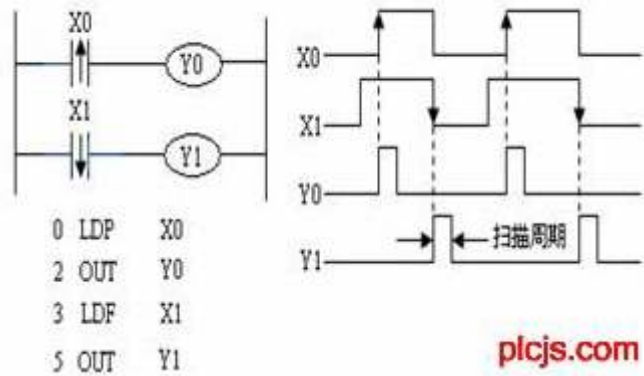


3、软件设计



知识点扩展：

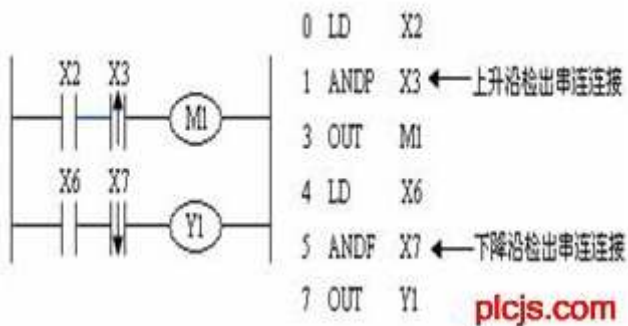
- 1、取脉冲指令 LDP、LDF



LDP 取脉冲上升沿，指在输入信号的上升沿接通一个扫描周期

LDF 取脉冲下降沿，指在输入信号的下降沿接通一个扫描周期

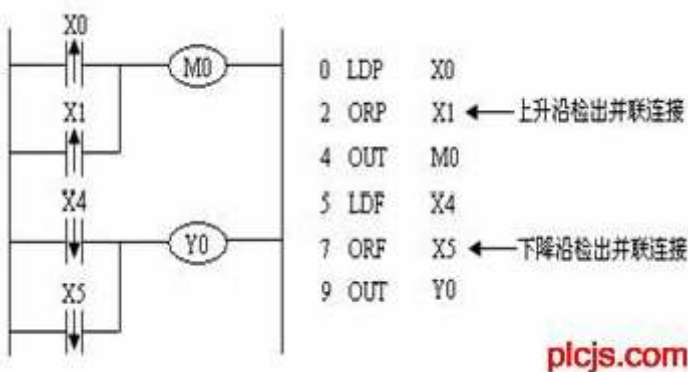
2、与脉冲指令 ANDP、ANDF



ANDP: 与脉冲上升沿

ANDF: 与脉冲下降沿

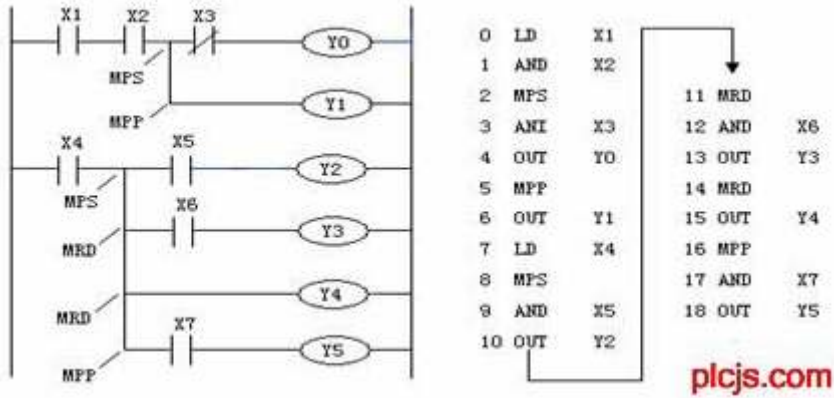
3、或脉冲指令 ORP、ORF



ORP: 或脉冲上升沿

ORF: 或脉冲下降沿

4、多重输出指令 MPS、MRD、MPP



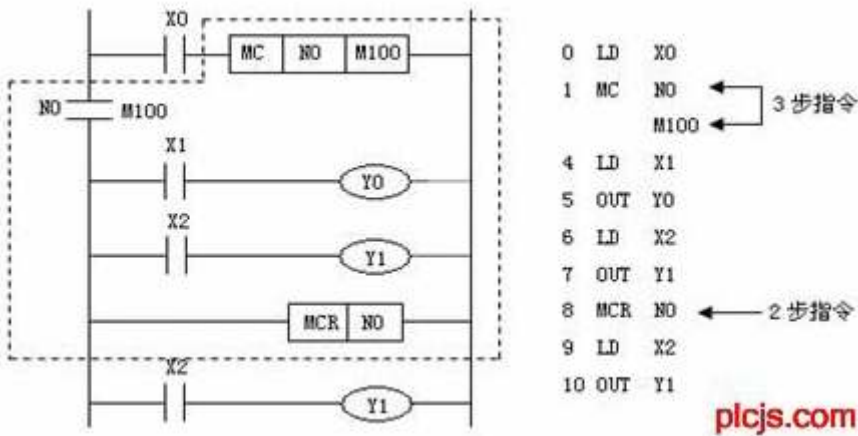
MPS, 进栈指令

MRD, 读栈指令

MPP, 出栈指令

这三条指令是无操作元件指令，都为一个程序步长。这组指令用于多输出电路。可将连接点先存储，用于连接后面的电路

5、主控及主控复位指令 MC、MCR



MC 为主控指令：用于公用串联接点的连接。

MCR 叫主控复位指令：MC 的复位指令。

在编程时，经常遇到多个线圈同时受一个或一组接点控制。如果在每个线圈的控制电路中都串入同样的接点，将多占用存储单元，应用主控指令可以解决这一问题。

案例一：

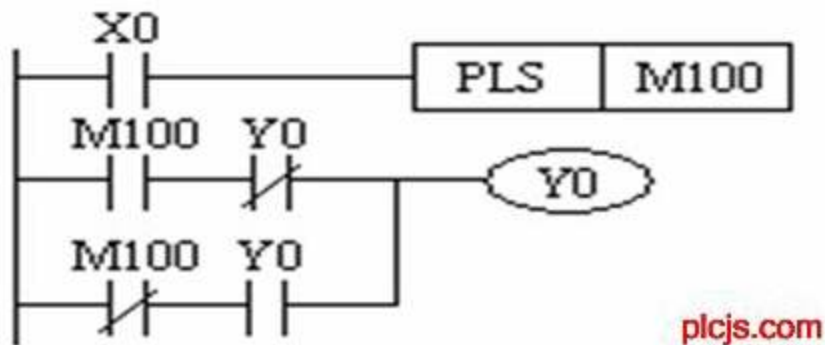
用 PLC 实现下图所示方波的 2 分频



1、可编程控制器的硬件连接

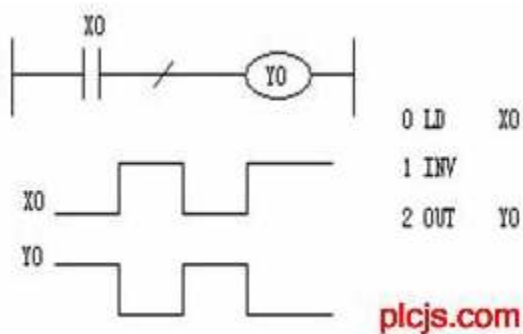
根据前面的学习，再依据图中的标示，同学们可以自己画出输入及输出端口的分配，在这不加深述。

2、软件设计



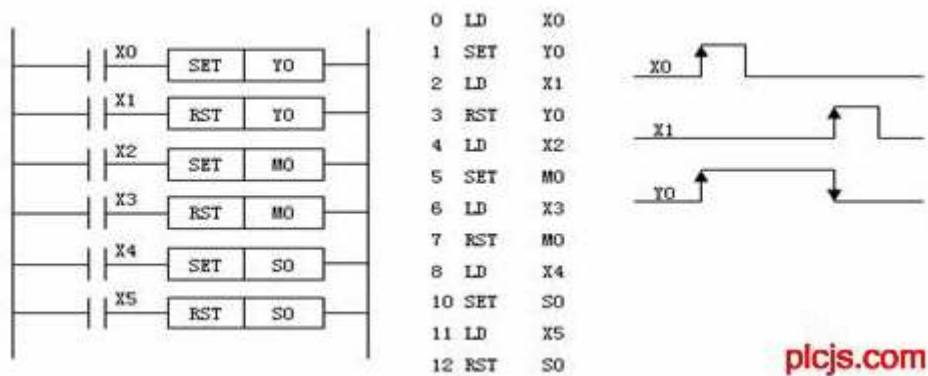
知识点链接：

1、取反指令 INV



该指令用于运算结果的取反。当执行该指令时，将 INV 指令之前存在的如 LD、LDI 等指令的运算结果反转

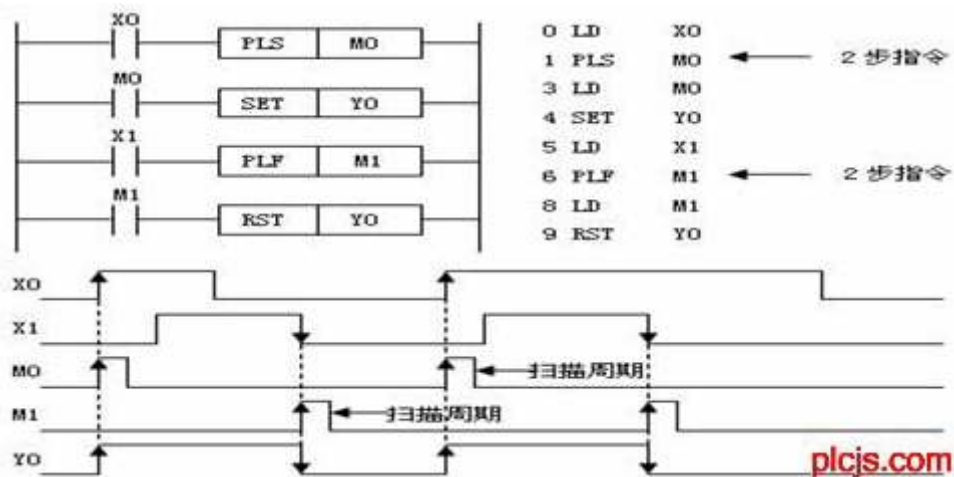
2、置位与复位指令 SET、RST



SET 为置位指令：使动作保持

RST 为复位指令：使操作保持复位

3、脉冲输出指令 PLS、PLF



PLS 指令：在输入信号上升沿产生脉冲输出

PLF 指令：在输入信号下降沿产生脉冲输出

4、空操作指令 NOP

NOP 为空操作指令，该指令是一条无动作、无目标元件占一个程序步的指令。空操作指令使该步序作空操作。用 NOP 指令替代已写入指令，可以改变电路。在程序中加入 NOP 指令，在改动或追加程序时可以减少步序号的变化。执行完清除用户存储器的操作后，用户存储器的内容全部变为空操作指令。

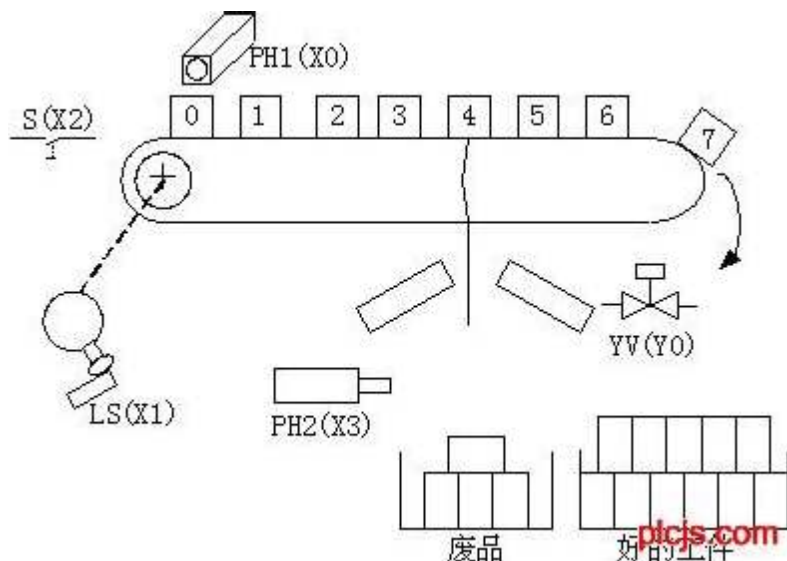
5、程序结束指令 END

END 是一条无目标元件占一个程序步的指令。PLC 反复进行输入处理、程序运算、输出处理，若在程序最后写入 END 指令，则 END 以后的程序步就不再执行，直接进行输出处理。在程序调试过程中，按段插入 END 指令，可以顺序扩大对各程序段动作的检查。采用 END 指令将程序划分为若干段，在确认处于前面电路块的动作正确无误之后，依次删去 END 指令。要注意的是在执行 END 指令时，也刷新监视时钟。

案例二：

现代工业生产广泛采用流水作业，对成品或半成品进行分检，排除残次品是必须的工序。在流水线上，成品或半成品往往要经过若干项检验，符合要求者得以通过，随着流水线进入下道生产工步。而不合格者必须在某处集中地予以排除，不得进入下道生产环节。因此，成品或半成品随传送带递进过程中，对其进行的检验结果也必须同步地向前移动。这样，当不合格者移动到规定的排除点时，才能正确地加以排除，从而实现质检与分检。利用 PC 提供的移位寄存器功能，可以很方便地做到这一点。

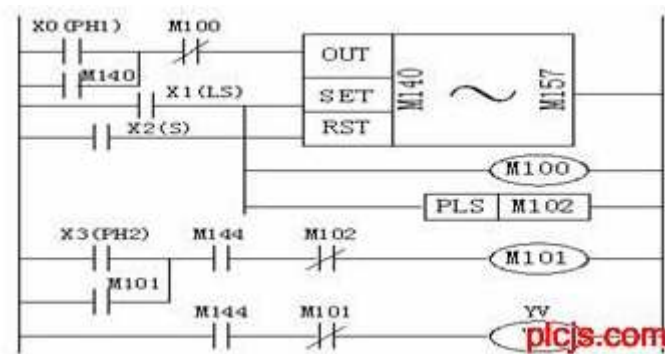
1、质量控制过程示意图



如图所示，工件在传送带上步行式前进。工件在 0 号站接受 PH1 光电检查。从 0 号站移到 4 号站要经过 4 次步进。移到 4 号站时，判断移到此站工件好还是坏。如果是废品则要求电磁阀 YV 打开，使废品掉下来。当 PH2 检测到废品已掉下来后重新将电磁阀关闭。PLC 的输入条件有以下几个：PH1 光电检测，当工件合格时，PH1 数据为“0”；当工件废品时，PH1 数据为“1”。PH2 光电检测，当数据为“1”表明废品已掉下来。LS 为行程开关，凸轮每转一次 LS 接通一次，工件步进一步。S 为外加复位按钮。PC 的输出点只有一个即电磁阀 YV。

2、硬件的连接

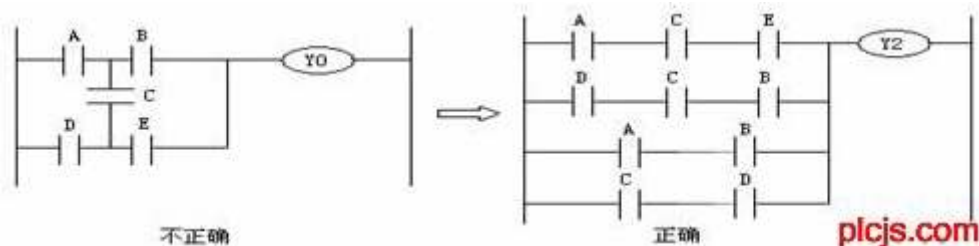
根据上述模块的学习，再依据图中的标示，同学们可以自己画出输入及输出端口的分配，在这不加深述。



3、系统的软件设计

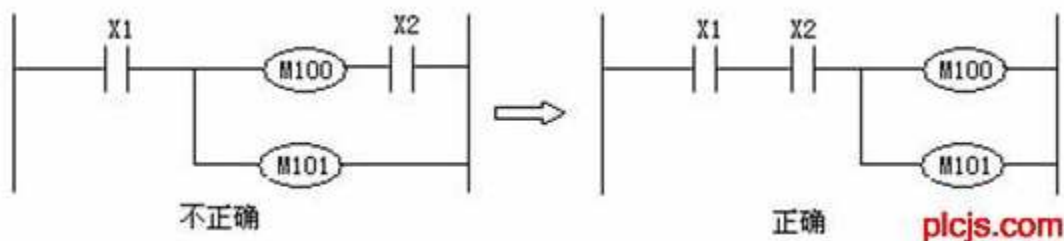
可编程控制器编程的基本原则：

1、水平不垂直



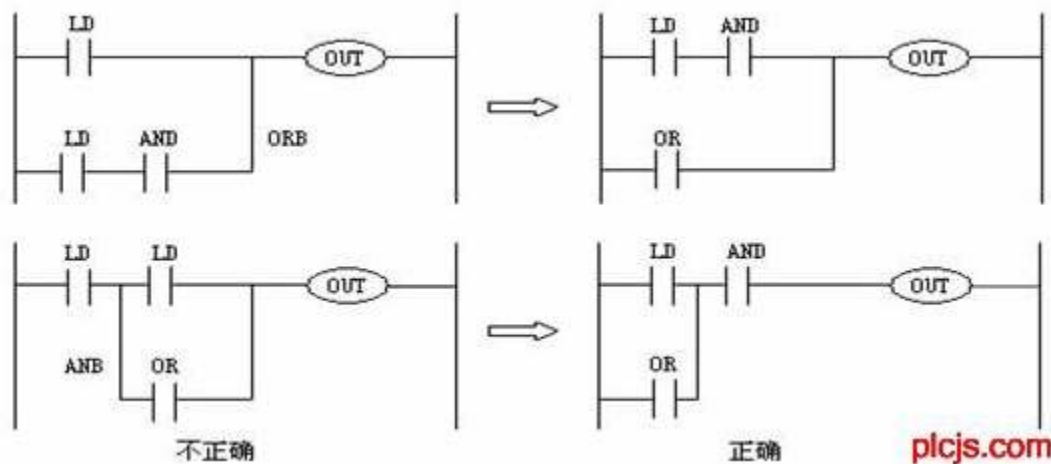
梯形图的接点应画在水平线上，不能画在垂直分支上

2、线圈右边无接点



不能将接点画在线圈右边，只能在接点的右边接线圈

3、左大右小，上大下小



有串联电路并联时，应将接点最多的那个串联回路放在梯形图最上面。有并联电路相串联时，应将接点最多的并联回路放在梯形图的最左边。

4、双线圈输出不可用



如果在同一程序中同一元件的线圈使用两次或多次，则称为双线圈输出。这时前面的输出无效，只有最后一次才有效，一般不应出现双线圈输出。

7.PLC 程序设计方法一.梯形图经验设计法

一、梯形图经验设计法

经验设计方法也叫试凑法，经验设计方法需要设计者掌握大量的典型电路，在掌握这些典型电路的基础上，充分理解实际的控制问题，将实际控制问题分解成典型控制电路，然后用典型电路或修改的典型电路进行拼凑梯形图。

二、梯形图经验设计法的步骤

分解梯形图程序

输入信号逻辑组合

使用辅助元件和辅助触点

使用定时器和计数器

使用功能指令

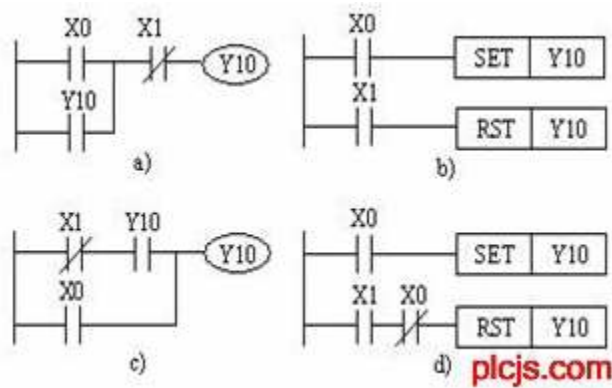
画互锁条件

画保护条件

三、常用基本环节梯形图程序

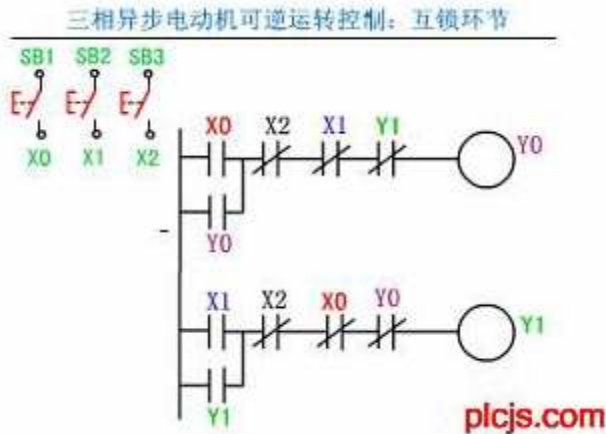
- | | |
|--------------------------|--|
| 1. <u>启动、保持和停止电路</u> | 6. <u>互锁控制电路</u> |
| 2. <u>三相异步电动机正反转控制电路</u> | 7. <u>顺序启动控制电路</u> |
| 3. <u>常闭触点输入信号的处理</u> | 8. <u>集中与分散控制电路</u> |
| 4. <u>多继电器线圈控制电路</u> | 9. <u>自动与手动控制电路</u> |
| 5. <u>多地控制电路</u> | 10. <u>闪烁电路</u> |
| | 11. <u>延合延分电路</u> |
| | 12. <u>定时范围扩展</u> plcjs.com |

1、启动、保持和停止电路



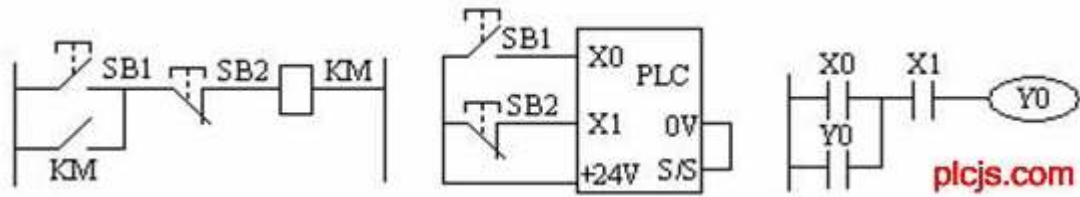
实现 Y10 的启动、保持和停止的四种梯形图如图所示。这些梯形图均能实现启动、保持和停止的功能。X0 为启动信号，X1 为停止信号。图 a、c 是利用 Y10 常开触点实现自锁保持，而图 b、d 是利用 SET、RST 指令实现自锁保持。

2、三相异步电动机正反转控制



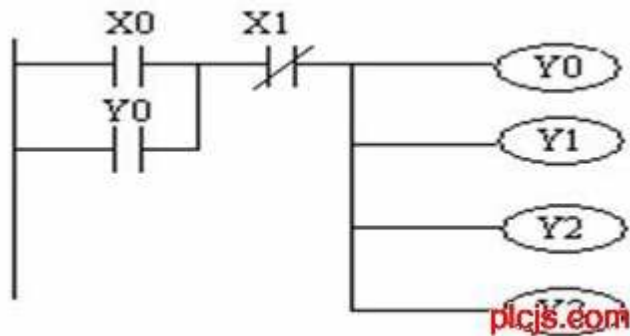
3、常闭触点输入信号的处理

如果输入信号只能由常开触点提供，梯形图中的触点类型与继电器电路的触点类型完全一致。如果接入 PLC 的是输入信号的常闭触点，这时在梯形图中所用的 X1 的触点的类型与 PLC 外接 SB2 的常开触点时刚好相反，与继电器电路图中的习惯也是相反的。建议尽可能采用常开触点作为 PLC 的输入信号。



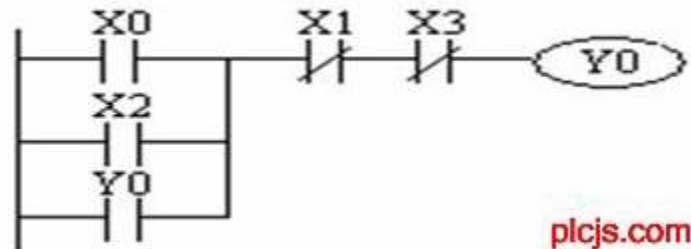
4、多继电器线圈控制电路

下图是可以自锁的同时控制 4 个继电器线圈的电路图。其中 X0 是起动按钮，X1 是停止按钮。



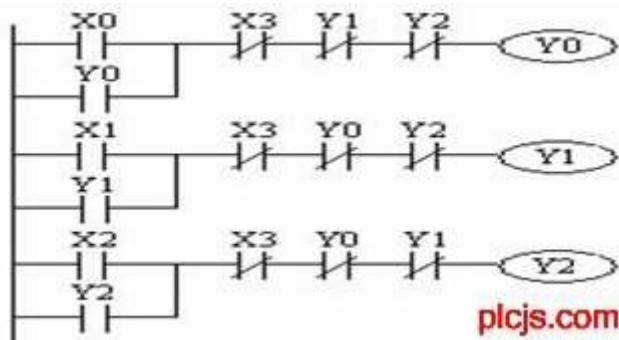
5、多地控制电路

下图是两个地方控制一个继电器线圈的程序。其中 X0 和 X1 是一个地方的起动和停止控制按钮，X2 和 X3 是另一个地方的起动和停止控制按钮。



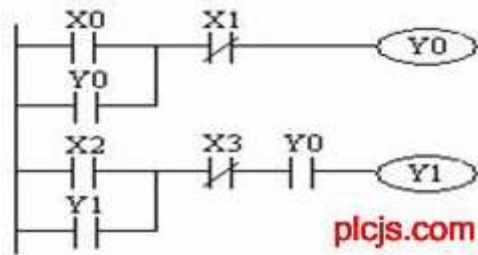
6、互锁控制电路

下图是 3 个输出线圈的互锁电路。其中 X0、X1 和 X2 是起动按钮，X3 是停止按钮。由于 Y0、Y1、Y2 每次只能有一个接通，所以将 Y0、Y1、Y2 的常闭触点分别串联到其它两个线圈的控制电路中。



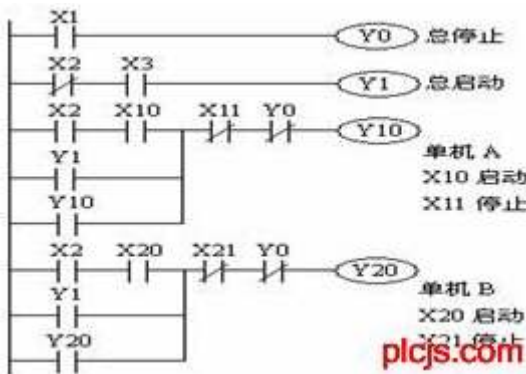
7、顺序起动控制电路

如图所示。Y0 的常开触点串在 Y1 的控制回路中，Y1 的接通是以 Y0 的接通为条件。这样，只有 Y0 接通才允许 Y1 接通。Y0 关断后 Y1 也被关断停止，而且 Y0 接通条件下，Y1 可以自行接通和停止。X0、X2 为起动按钮，X1、X3 为停止按钮。



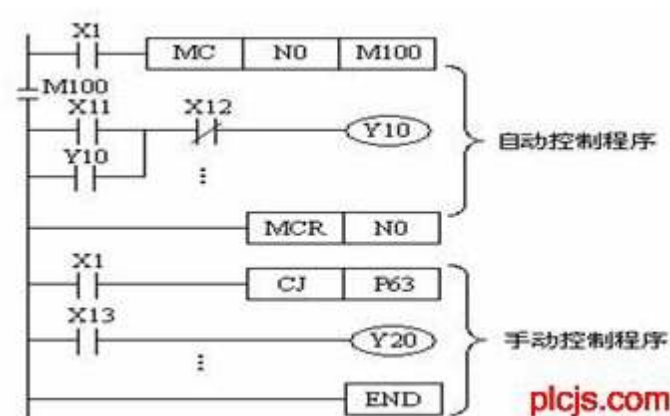
8、集中与分散控制电路

在多台单机组成的自动线上，有在总操作台上的集中控制和在单机操作台上分散控制的联锁。集中与分散控制的梯形图如图所示。X2 为选择开关，以其触点为集中控制与分散控制的联锁触点。当 X2 为 ON 时，为单机分散起动控制；当 X2 为 OFF 时，为集中总起动控制。在两种情况下，单机和总操作台都可以发出停止命令。



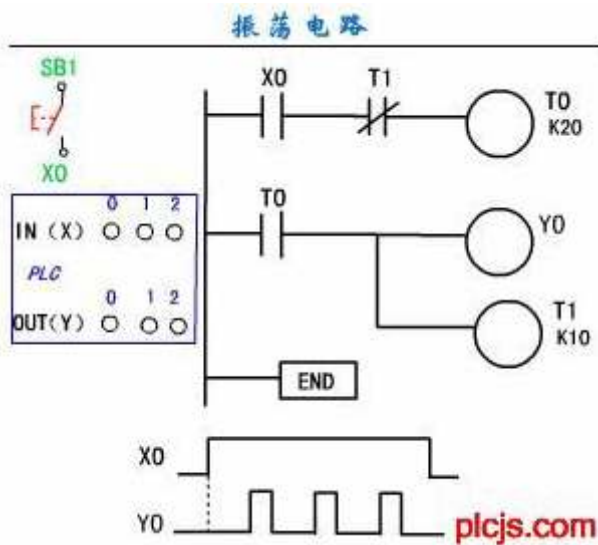
9、自动与手动控制电路

在自动与半自动工作设备中，有自动控制与手动控制的联锁，如图所示。输入信号 X1 是选择开关，选其触点为联锁型号。当 X1 为 ON 时，执行主控指令，系统运行自动控制程序，自动控制有效，同时系统执行功能指令 CJ? P63，直接跳过手动控制程序，手动调整控制无效。当 X1 为 OFF 时，主控指令不执行，自动控制无效，跳转指令也不执行，手动控制有效。



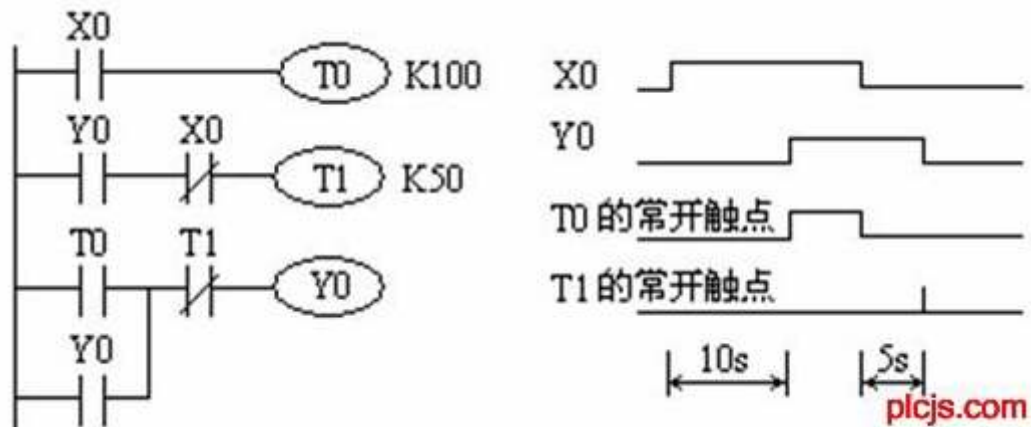
10、闪烁电路

当拨动开关将 X0 接通，启动脉冲发生器。延时 2s 后 Y0 接通，再延时 1s 后 Y0 断开。这一过程周期性地重复。Y0 输出一系列脉冲信号，其周期为 3s，脉宽为 1s。



11、延合延分电路

如图所示用 X0 控制 Y0，当 X0 的常开触点接通后，T0 开始定时，10s 后 T0 的常开触点接通，使 Y0 变为 ON。X0 为 ON 时其常闭触点断开，使 T1 复位，X0 变为 OFF 后 T1 开始定时，5s 后 T1 的常闭触点断开，使 Y0 变为 OFF，T1 也被复位。Y0 用起动、保持、停止电路来控制。

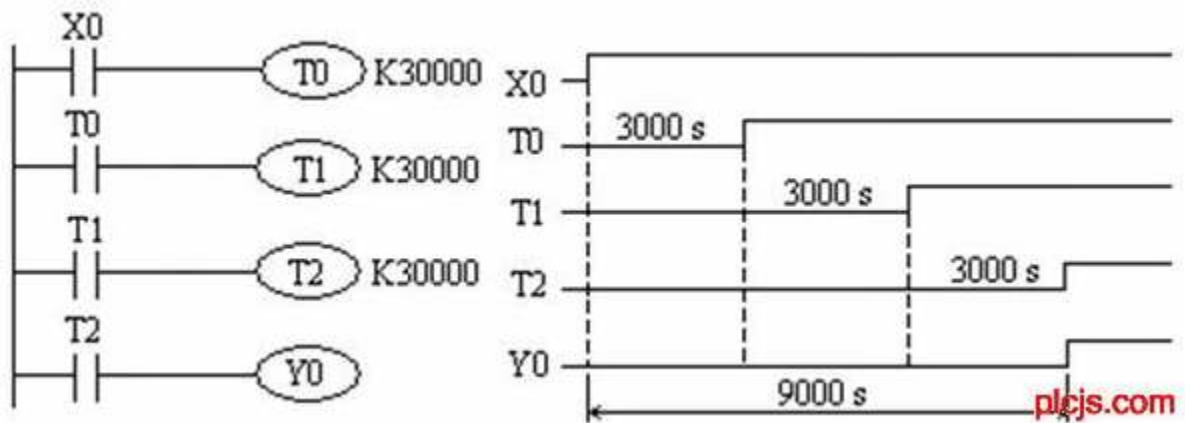


12、定时范围扩展电路

FX2N 系列 PLC 定时器的最长定时时间为 3276.7s，如果需要更长的定时时间，可以采用以下方法以获得较长延时时间。

1、多个定时器组合电路

如图所示。当 X0 接通，T0 线圈得电并开始延时，延时到 T0 常开触点闭合，又使 T1 线圈得电，并开始延时，当定时器 T1 延时到，其常开触点闭合，再使 T2 线圈得电，并开始延时，当定时器 T2 延时到，其常开触点闭合，才使 Y0 接通。因此，从 X0 为 ON 开始到 Y0 接通共延时 9000s。



2、定时器和计数器组合

当 X1 为 ON 时，T1 开始定时，0.6s 后 T1 定时时间到，其常闭触点断开，使它自己复位，复位后 T1 的当前值变为 0，同时它的常闭触点接通，使它自己的线圈重新通电，又开始定时。T1 将这样周而复始地工作，直至 X1 变为 OFF。从分析中可看出，最上面一行电路是一个脉冲信号发生器，脉冲周期等于 T1 的设定值。产生的脉冲列送给 C0 计数，计满 3 个数后，C0 的当前值等于设定值，它的常开触点闭合，Y0 开始输出。

8.PLC 程序设计方法二.继电器控制电路移植法

一、继电器控制电路移植法设计梯形图

用 PLC 改造继电器控制系统时，因为原有的继电器控制系统经过长期的使用和考验，已被证明能够完成系统要求的控制功能，而且继电器电路图与梯形图在表示方法和分析方法上有很多相似之处，因此可以根据继电器电路图设计梯形图，即将继电器电路图转换为具有相同功能的 PLC 外部硬件接线图和梯形图。

二、继电器控制电路移植法设计梯形图设计步骤

1. 了解和熟悉被控设备的工艺过程和机械的动作情况。
2. 确定 PLC 的输入信号和输出负载，画出 PLC 外部接线图。
3. 确定与继电器电路图的中间继电器、时间继电器对应的梯形图中的辅助继电器 M 和定时器 T 的元件号。
4. 根据上述对应关系画出梯形图

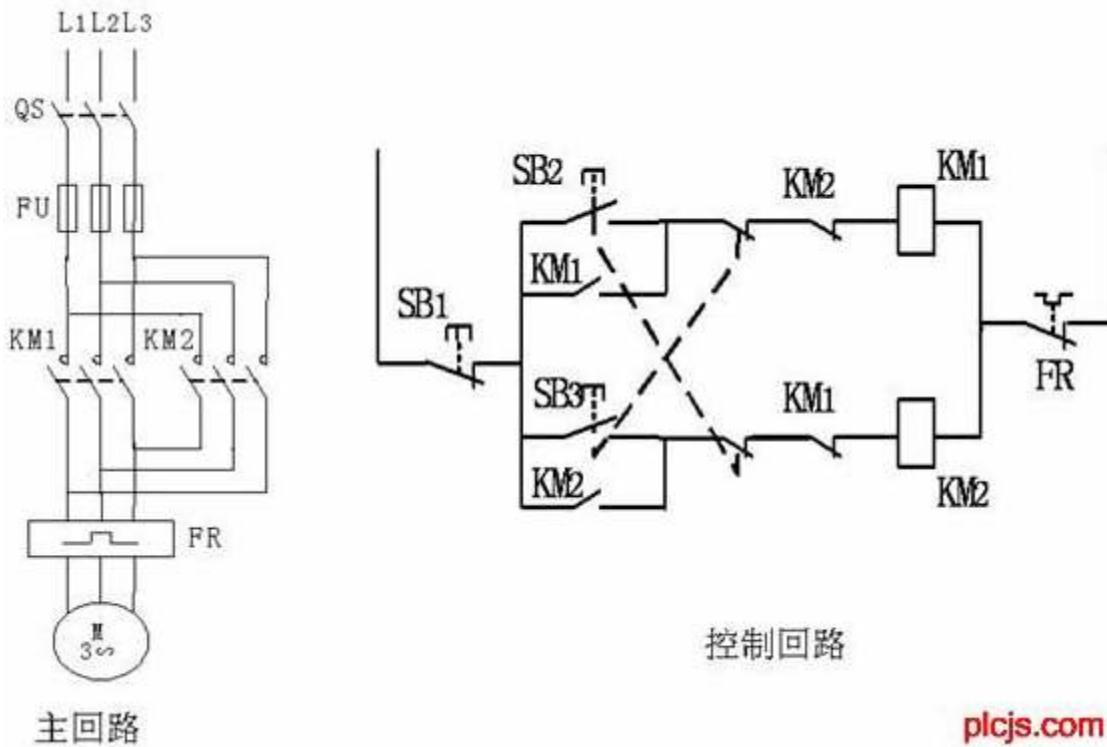
案例：

PLC 控制的电动机正反转

利用 PLC 控制一台异步电动机的正反转。

输入端直流电源 E 由 PLC 内部提供，可直接将 PLC 电源端子接在开关上。交流电源则是由外部供给。

- 1、熟悉电动机的正反转控制电路

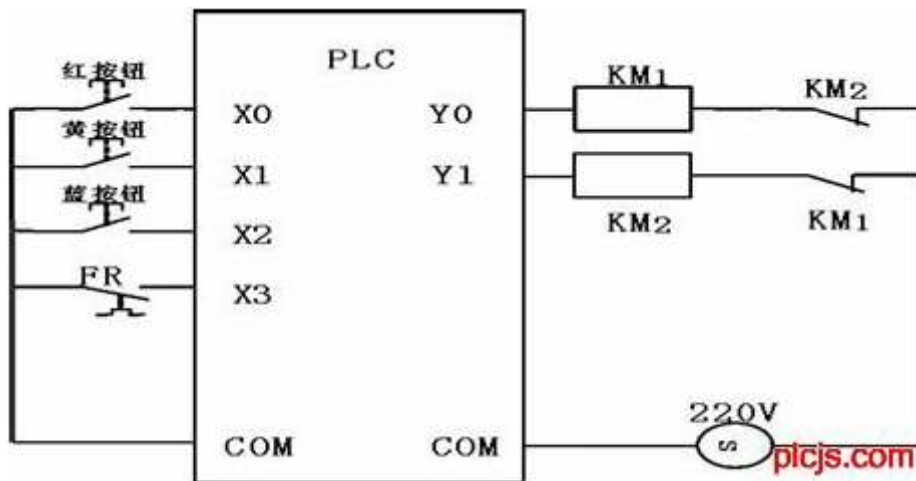


2、PLC 的 I/O 点的确定与分配

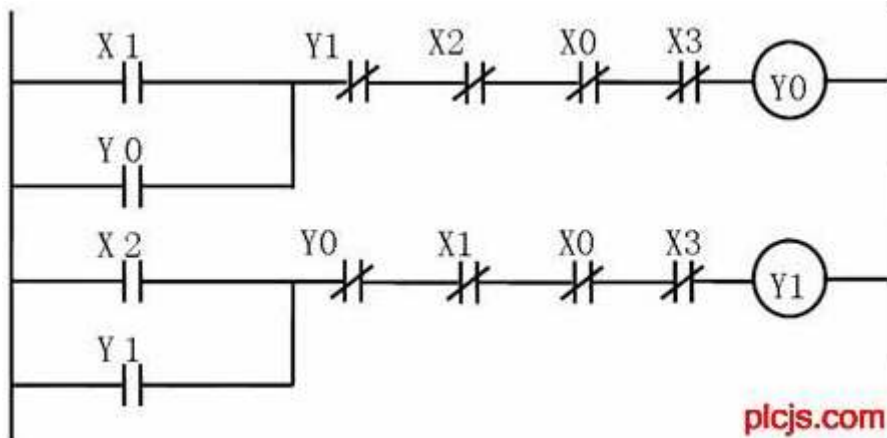
电机正反转控制PLC的I/O点分配表		
PLC点名称	连接的外部设备	功能说明
X0	SB1	停止命令
X1	SB2	电机正转命令
X2	SB3	电机反转命令
X3	FR常开	电动机过载保护
Y0	KM1	控制电机正转
Y1	KM2	控制电机反转

plcjs.com

3.PLC 控制电动机正反转外部接线图



4、程序编制



9.PLC 程序设计方法三 顺序控制设计法 之功能表图

一、顺序控制设计法

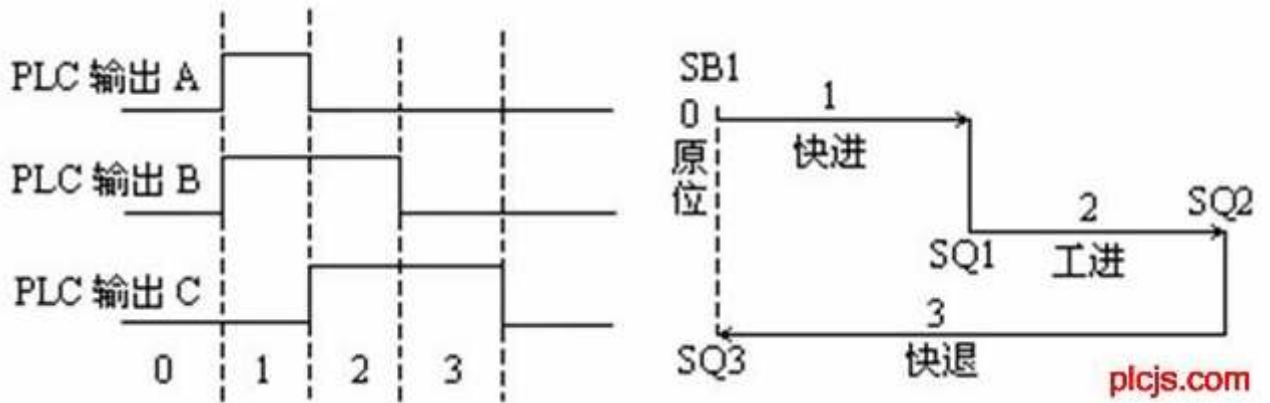
顺序控制就是按照生产工艺预先规定的顺序,在各个输入信号的作用下,根据内部状态和时间的顺序,在生产过程中各个执行机构自动地有序地进行工作。使用顺序控制设计法时首先根据系统的工艺过程,画出顺序功能图,然后根据顺序功能图画出梯形图。

顺序控制设计法是一种先进的设计方法,很容易被初学者接受,程序的调试、修改和阅读也很容易,并且大大缩短了设计周期,提高了设计效率。

二、顺序控制设计法的设计基本步骤

1.步的划分

分析被控对象的工作过程及控制要求,将系统的工作过程划分成若干个阶段,这些阶段称为“步”。步是根据 PLC 输出量的状态划分的,只要系统的输出量状态发生变化,系统就从原来的步进入新的步。在每一步内 PLC 各输出量状态均保持不变,但是相邻两步输出量总的状态是不同的。



2.转换条件的确定

转换条件是使系统从当前步进入下一步的条件。常见的转换条件有按钮、行程开关、定时器和计数器的触点的动作（通/断）等。

3.顺序功能图的绘制

根据以上分析画出描述系统工作过程的顺序功能图。这是顺序功能设计法中最关键的一个步骤。绘制顺序功能图的具体方法将在下节介绍。

4.梯形图的绘制

根据顺序功能图，采用某种编程方式设计出梯形图。常用的设计方法有三种：起一保一停电路设计法、以转换为中心设计法、步进顺控指令设计法。

三、顺序功能图的组成要素

顺序功能图主要由步、有向连线、转换、转换条件和动作（或命令）等要素组成。步与步之间实现转换应同时具备两个条件：

- (1) 前级步必须是活动步。
- (2) 对应的转换条件成立。



10.PLC 程序设计方法三 顺序控制设计法之单序列结构的编程方法

一、使用起保停电路的编程方法

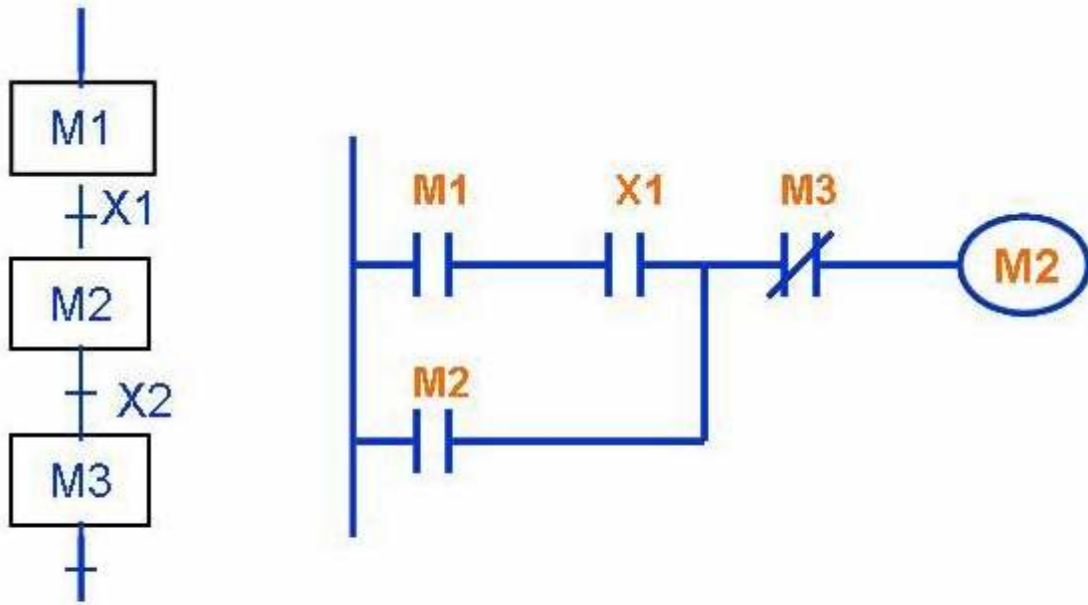
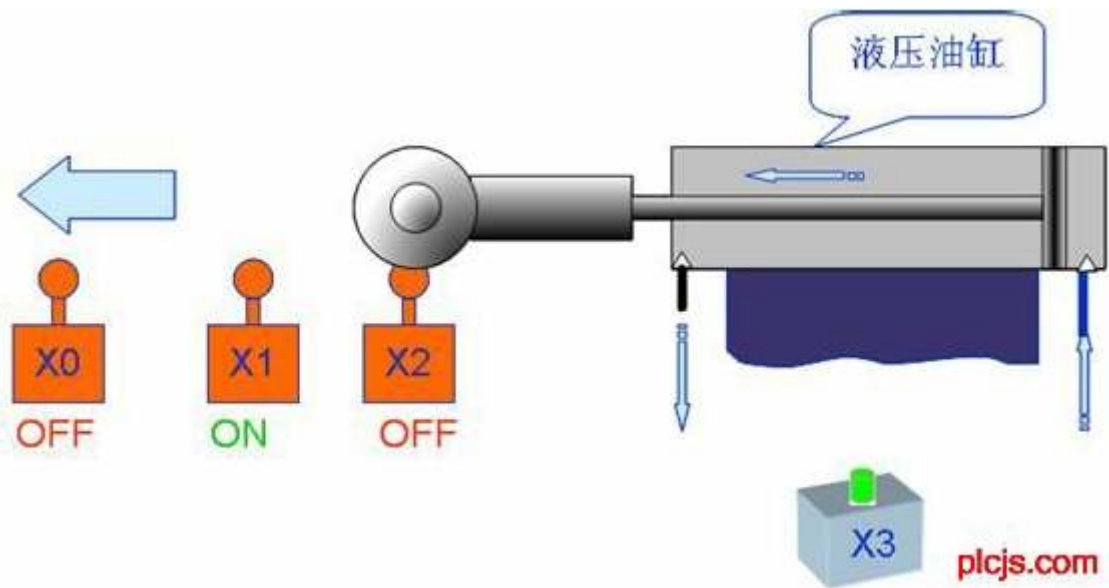


图4.28 使用起保停电路控制步

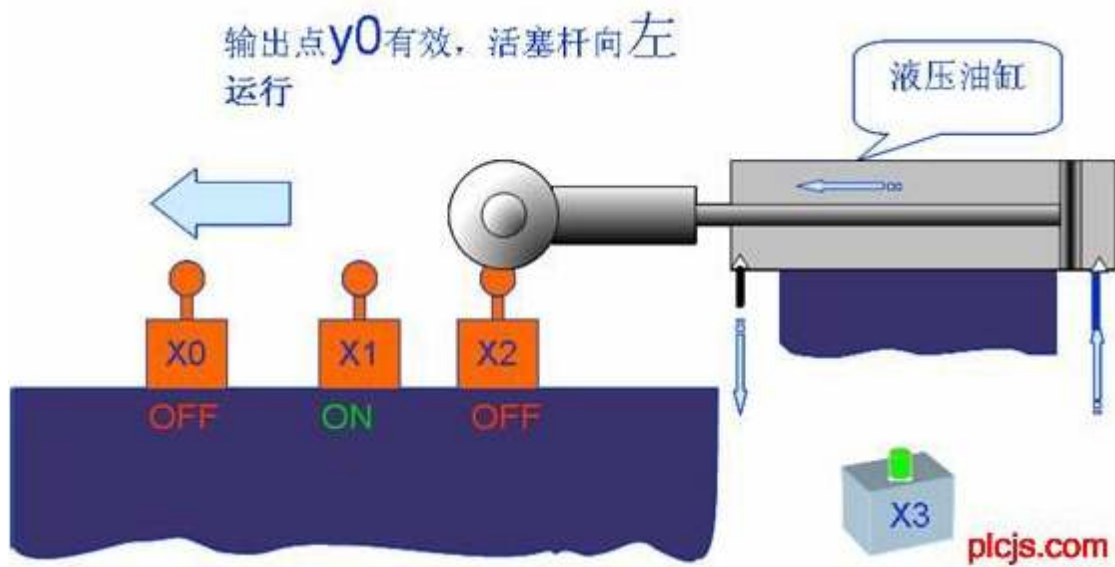
plcjs.com

案例：

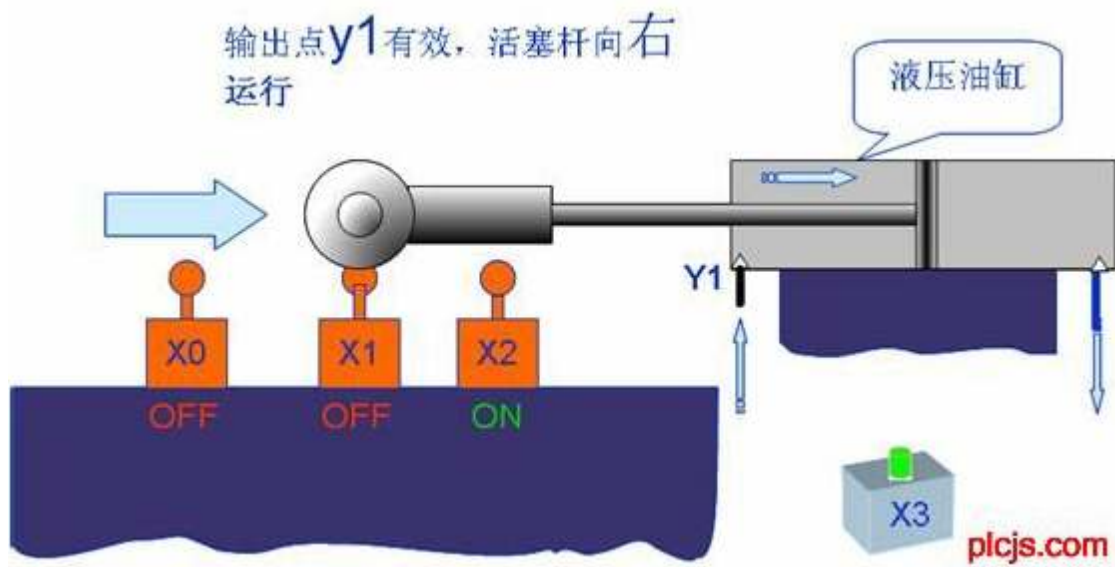
液压进给装置运动控制：



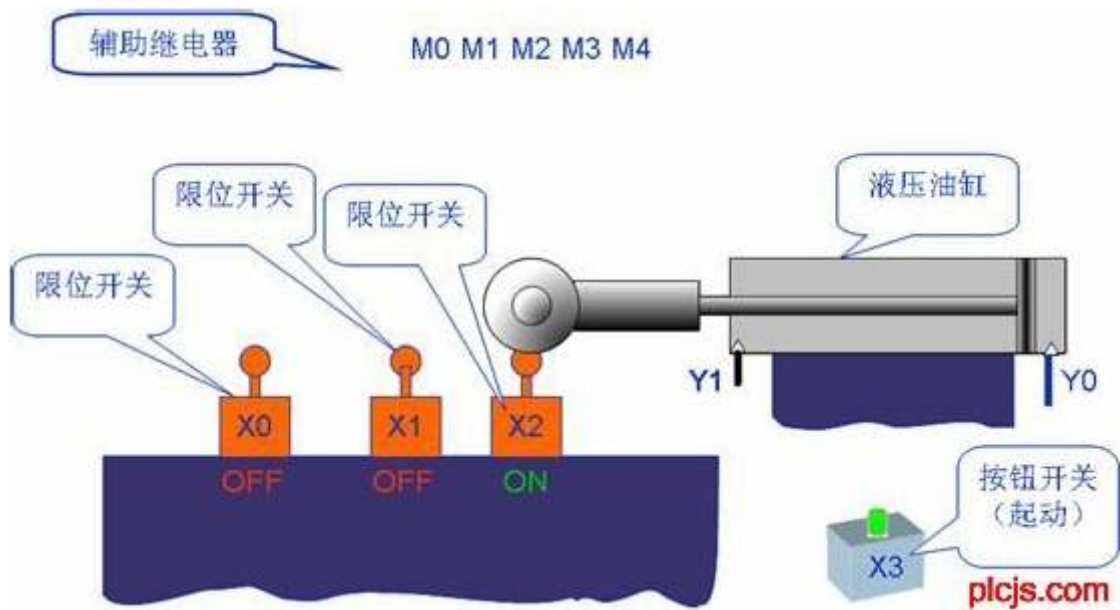
左行示意：



右行示意:



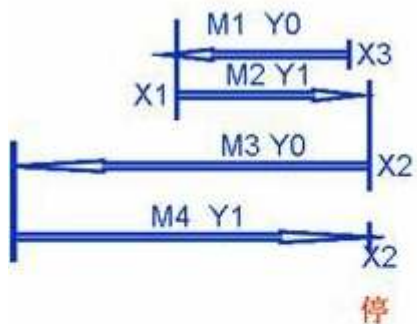
控制开关 (转换条件):



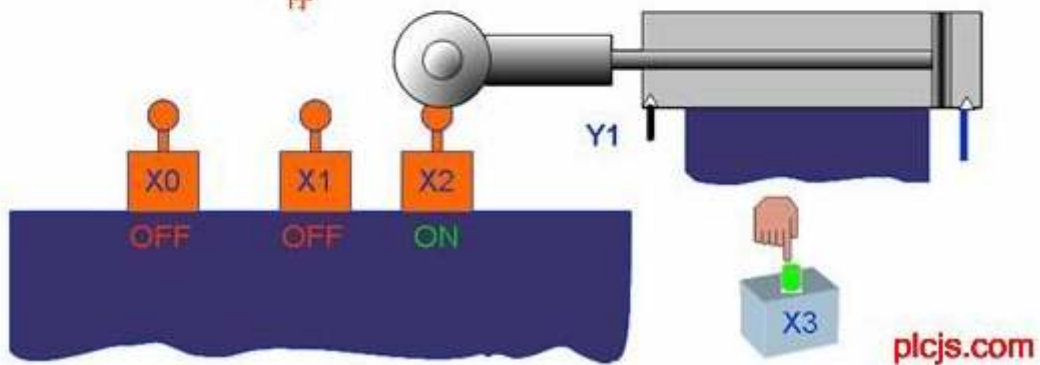
动作要求:

③进给装置顺序动作要求

★初始状态: 活塞杆置右端, 开关X2为ON, 辅助继电器M0为ON。

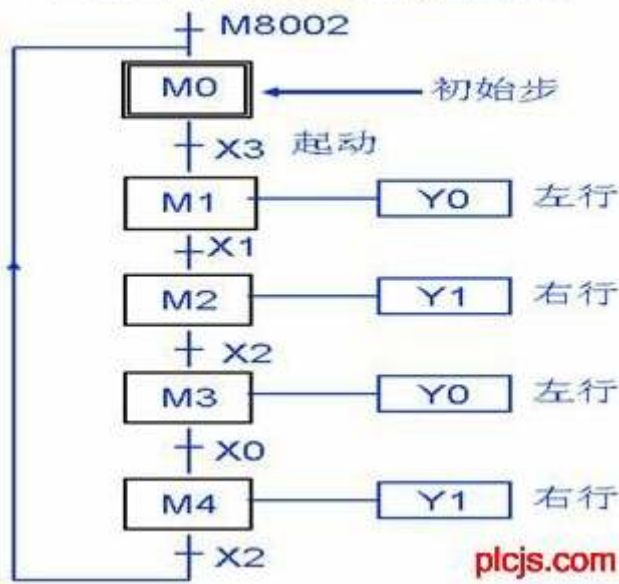


- ★1. 按下启动按钮X3, 开关Y0、M1为ON, 左行。
- ★2. 碰到限位开关X1时, M2、Y1为ON, 右行。
- ★3. 碰到限位开关X2时, M3、Y0为ON, 左行。
- ★4. 碰到限位开关X0时, M4、Y1为ON, 右行。
- ★5. 碰到限位开关X2时, 停止。



顺序功能图绘制:

单序列结构顺序功能图绘制



梯形图绘制:

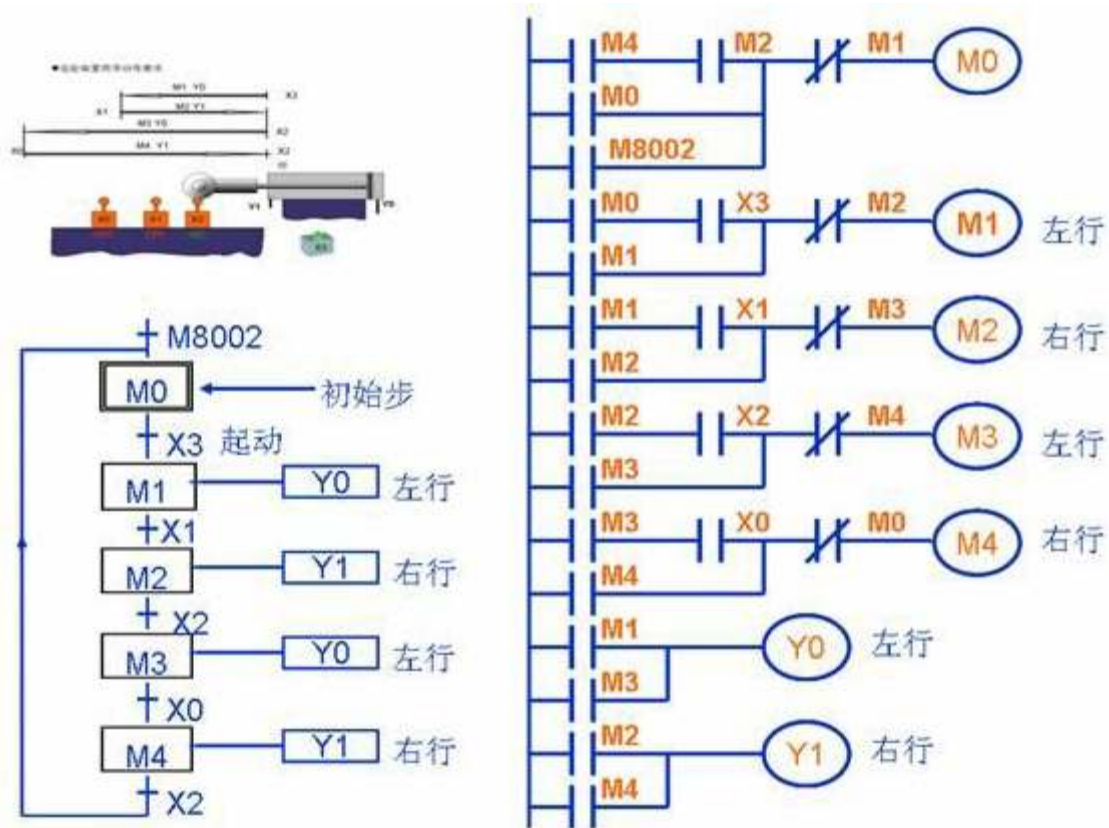


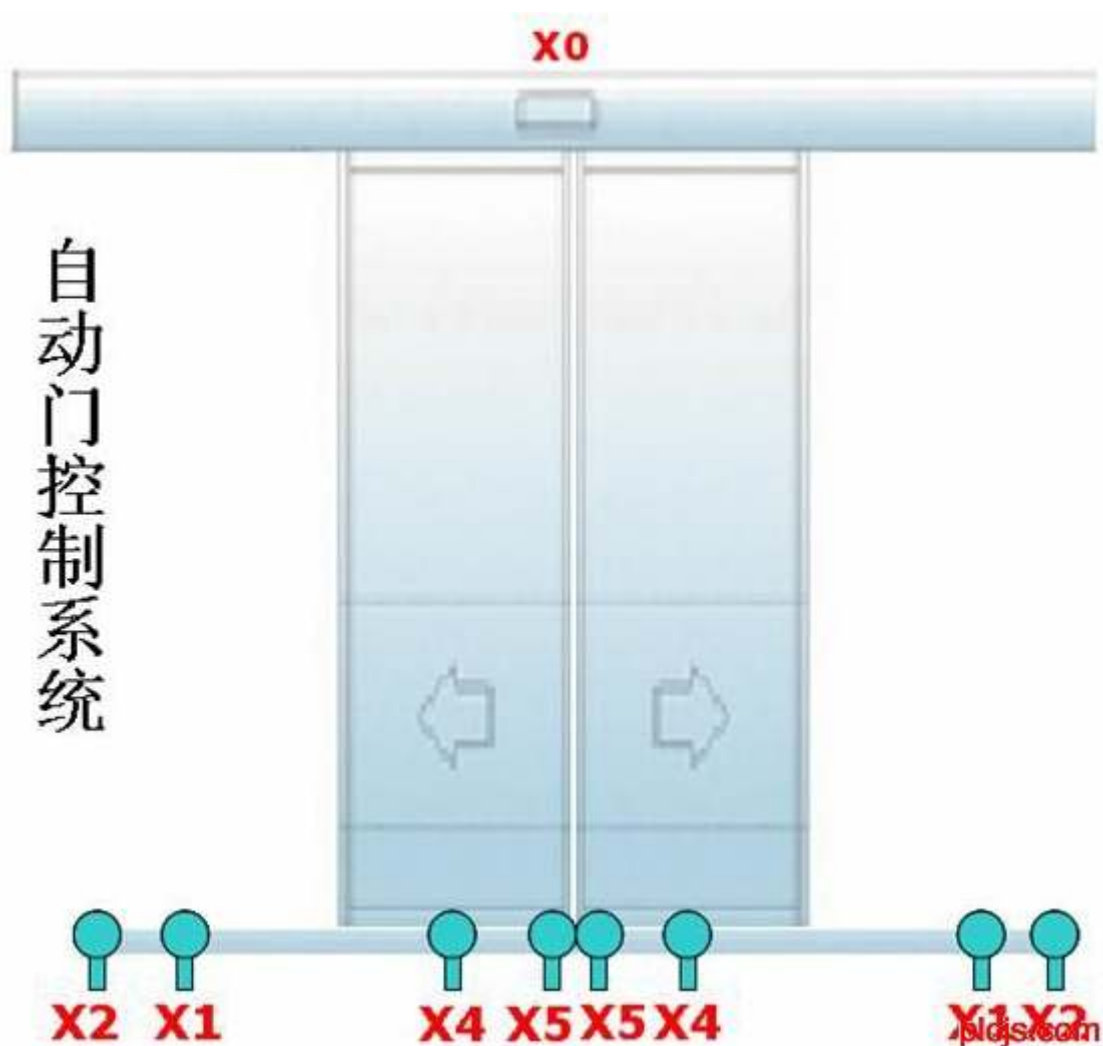
图4.20(b) 单序列结构顺序功能图

图4.20(c) 单序列结构梯形图

11.PLC 程序设计方法三 顺序控制功能图的编程方法选择单序列结构的编程方

法

案例:



一、动作分析

人靠近自动门时，感应器 X0 为 ON，Y0 驱动电动机高速开门，碰到开门减速开关 X1 时，变为低速开门。碰到开门极限

开关 X2 时电动机停转，开始延时。若在 0.5s 内感应器检测到无人，Y2 起动电动机高速关门。碰到关门减速开关 X4 时，

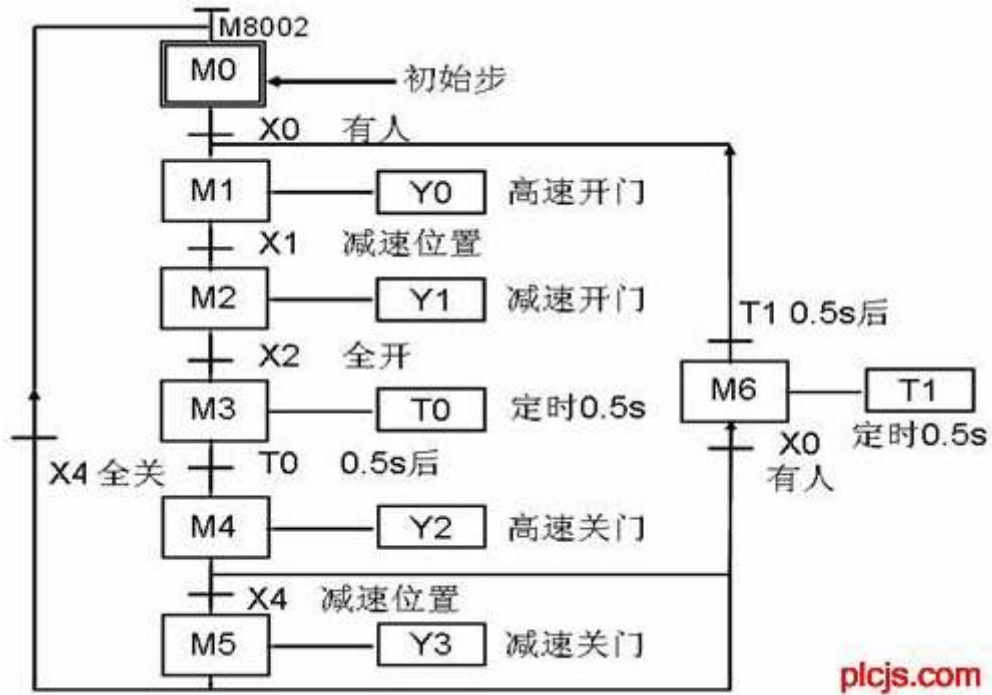
改为低速关门，碰到关门极限开关 X5 时电动机停转。在关门期间若感应器检测到有人，停止关门，T1 延时 0.5s 后自动转

换为高速开门。

二、硬件设计

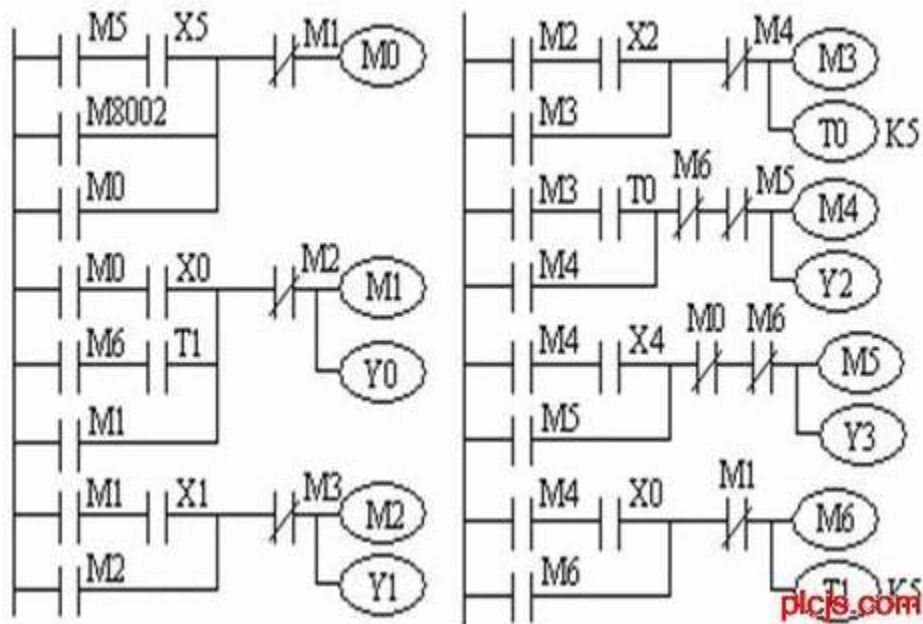
根据前面的学习，再依据图中的标示，同学们可以自己画出输入及输出端口的分配，在这不加深述

三、顺序功能图的绘制

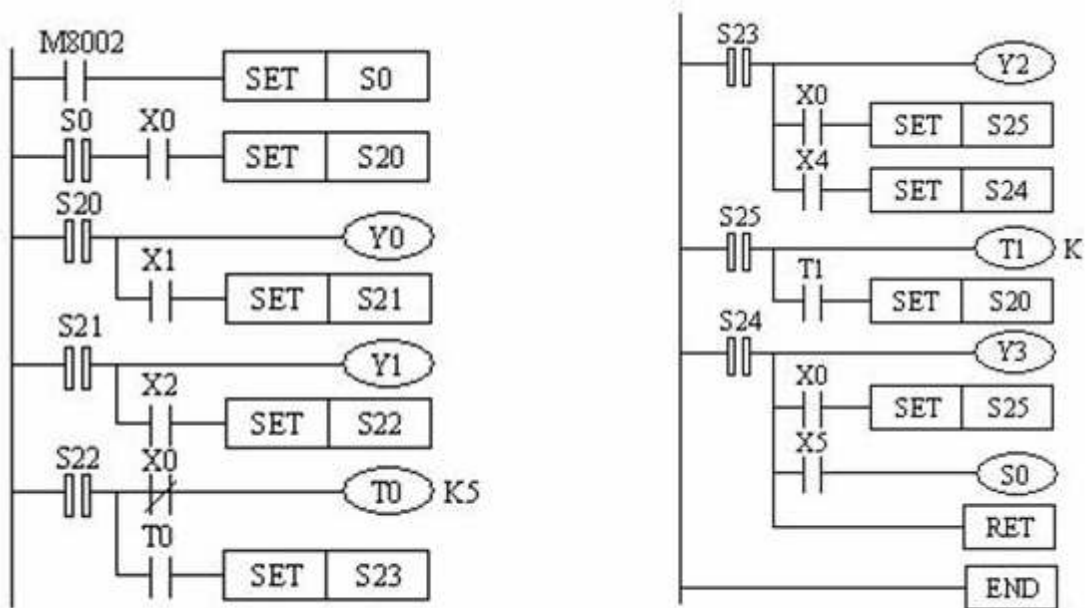


四、编程

1、选择序列起-保-停电路编程



2、选择序列结构的步进指令编程



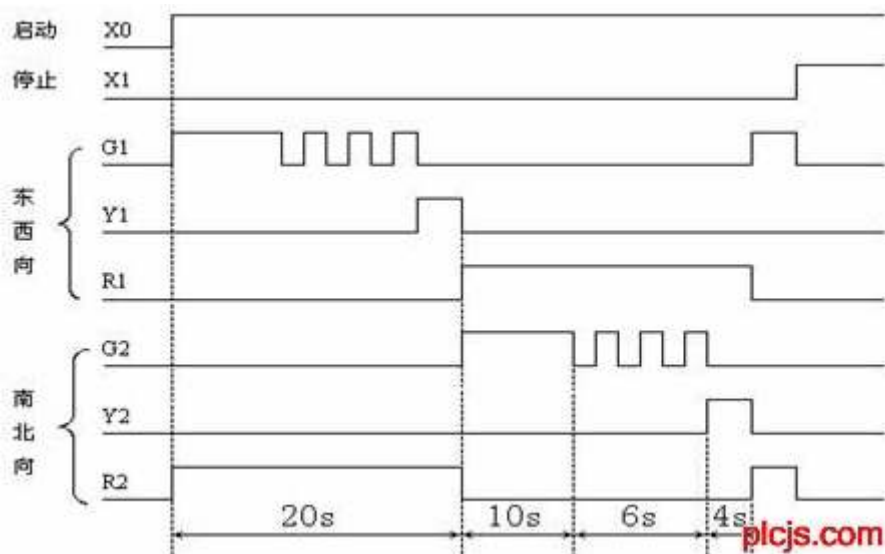
plcjs.com

12.PLC 程序设计方法三顺序控制功能图的编程方法之并行单序列结构的编程方法

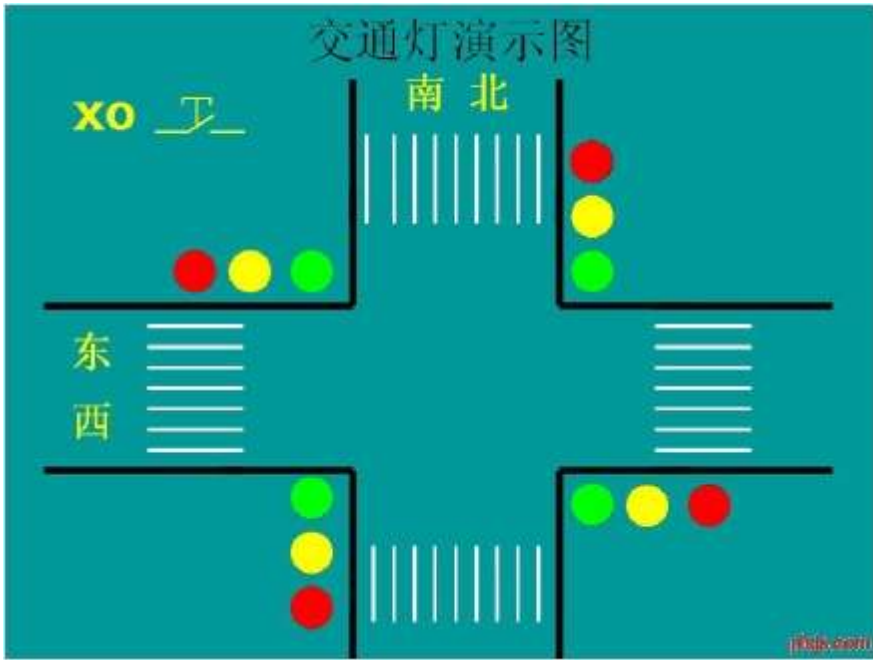
案例：

交通信号灯控制系统

1、十字路口双向交通灯自动控制系统动作要求如时序图所示。



plcjs.com



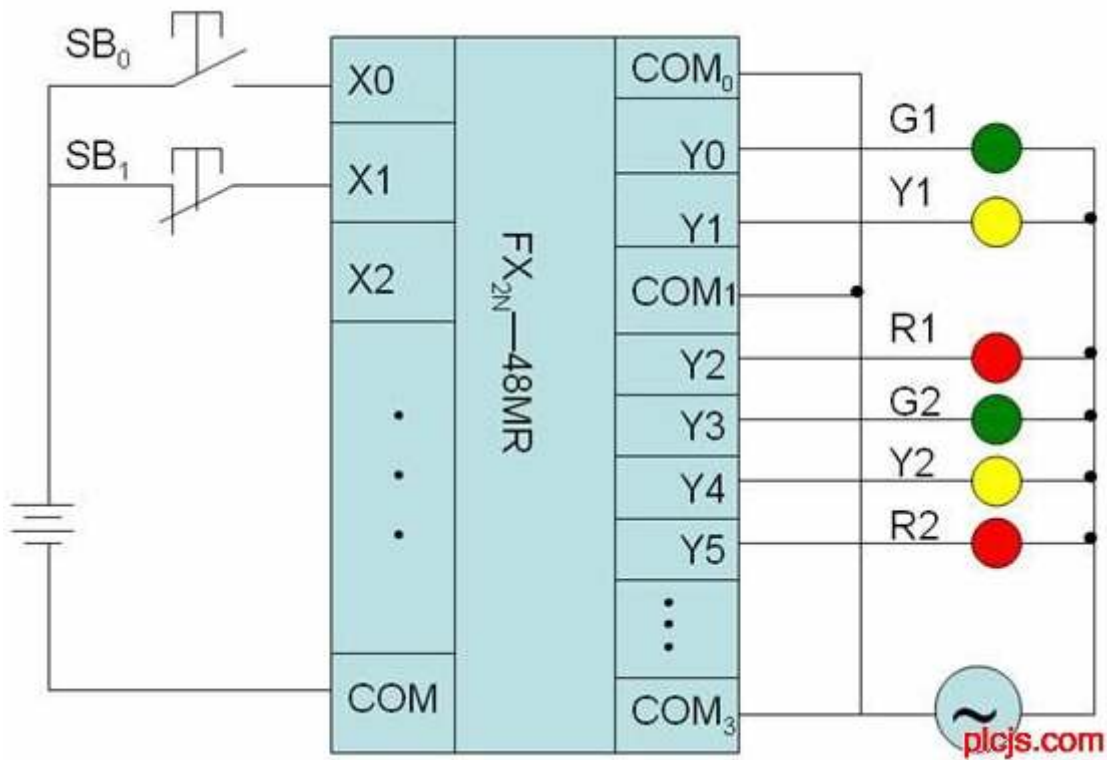
2、思考问题

1. 该控制系统是否属于顺序控制？是哪种结构？
1. 该控制系统工作步如何划分？共有多少工作步？
2. 输入/输出信号有那些？

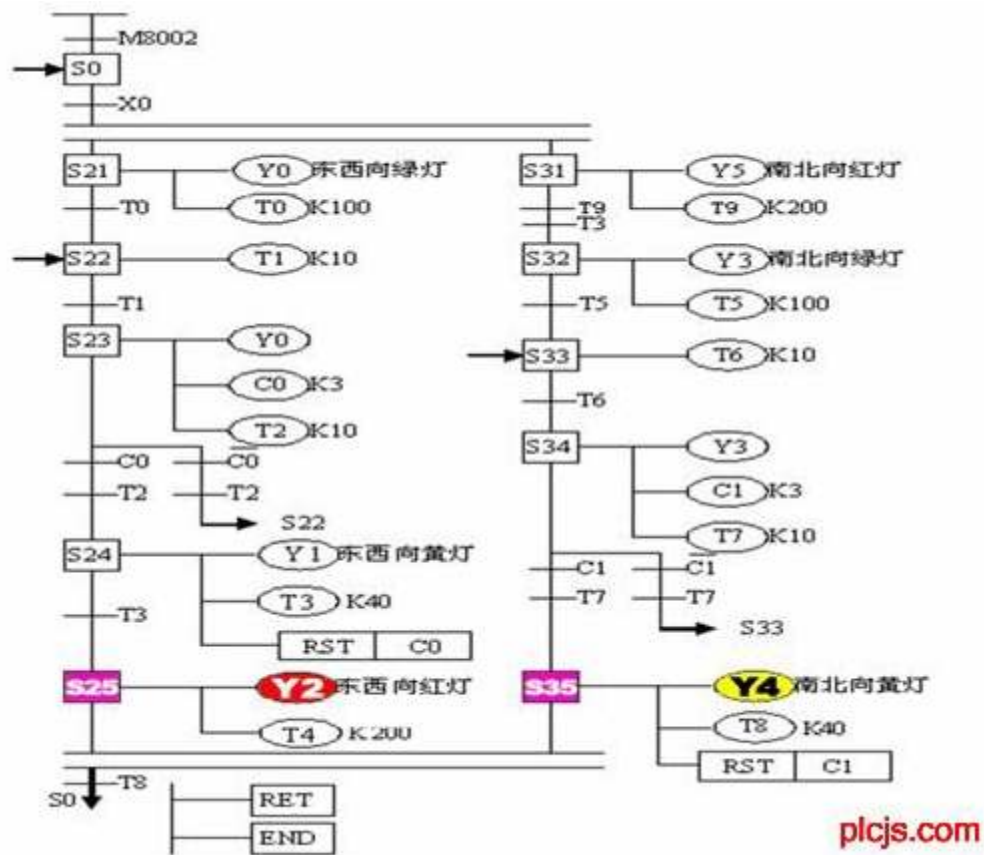
3、信号分配

输 入			输 出		
器件	器件号	功能说明	器件	器件号	功能说明
0	X0	启动按钮	G1	Y0	东西向绿灯
1	X1	停止按钮	Y1	Y1	东西向黄灯
			R1	Y2	东西向红灯
			G2	Y3	南北向绿灯
			Y2	Y4	南北向黄灯
			R2	Y5	南北向红灯

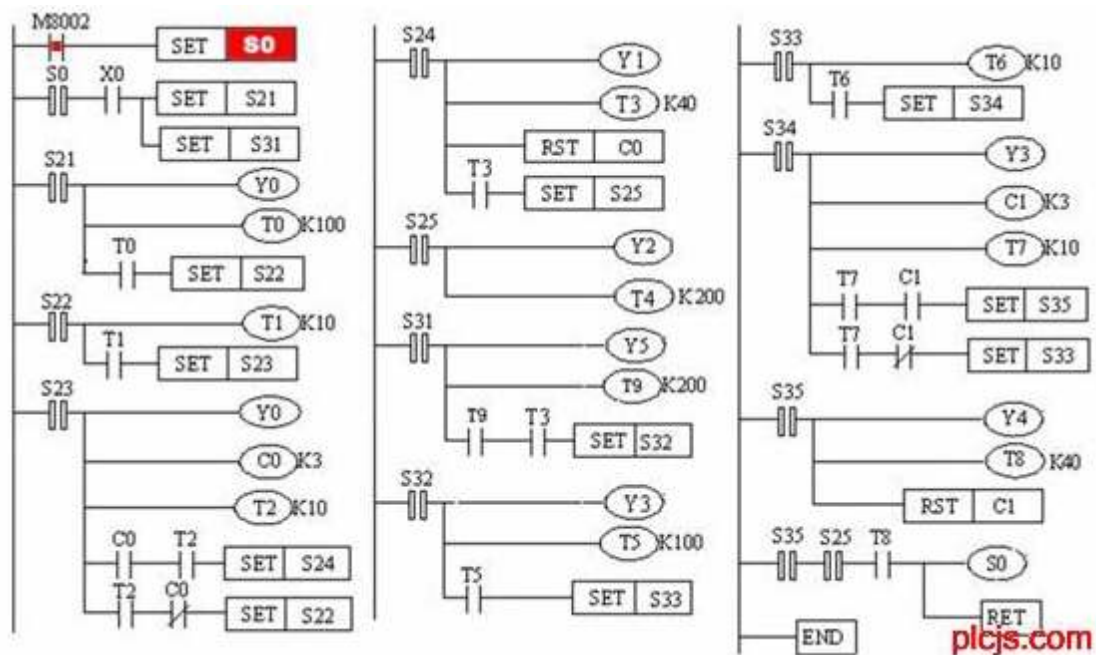
4、硬件设计



5、功能图的绘制



6、梯形图程序



7、指令程序

```

LD M8002      OUT Y0
SET S0        OUT C0
STL S0        K3
LD X0         OUT T2
SET S21       K10
SET S31       LD C0
STL S21       AND T2
OUT Y0        SET S24
OUT T0        LDI C0
              AND T2
LD T0         SET S22
SET S22       STL S24
STL S22       OUT Y1
OUT T1        OUT T3
              K10
LD T1         OUT T3
              K40
SET S23       RST C0
STL S23
LD T3         K100
SET S25       LD T5
STL S25       SET S33
OUT Y2        SET S33
OUT T4        STL S33
              OUT T6
              K10
              OUT Y4
STL S31       LD T6
              OUT T8
              K200
              SET S34
              K40
OUT T9        STL S34
              RST C1
              K200
              OUT Y3
LD T9         STL S35
              OUT C1
              K100
              STL S25
AND T3        LD T8
SET S32       OUT S0
STL S32       RET
OUT Y3        LD T7
              K10
OUT T5        AND C1
              END
    
```

PLC 程序设计方法比较

下面将经验设计法和顺序控制设计法进行适当比较,以便设计时选用。

采用经验设计法设计梯形图时, 不可能找出一种简单通用的设计方法。顺序控制设计法将整个程序分成了控制程序和输出程序两个部分。由于步是根据输出 Y 的状态划分的, 所以 M 和 Y 之间具有很简单的逻辑关系, 输出程序的设计极为简单。而代表步的辅助继电器或状态继电器的控制程序, 不管多么复杂, 其设计方法都是相同的, 并且很容易掌握。



13. PLC 功能指令

一、FX 系列可编程控制器功能指令概述

功能指令的通用表达形式

功能指令的表达形式如下表所示：

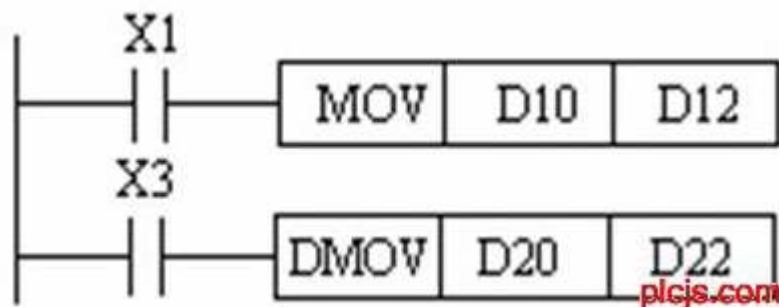
功能指令按功能号 FNC00~FNC249 编排。每条功能指令都有一个指令助记符。

MEAN FNC45 (P) (16) (Mean)	操作元件							
	[S]							
	K, H	KnX	KnY	KnM	KnS	T	C	D
n	[D]							
n=1~64	参数 MEAN: 7步							

数据长度

功能指令可处理 16 位数据和 32 位数据。

如下图中的第一条指令是将 D10 中的数据送到 D12 中，处理的是 16 位数据。第二条指令是将 D21 和 D20 中的数据送到 D23 和 D22 中，处理的是 32 位数据。



功能指令类型



FX 系列 PLC 的功能指令有连续执行型和脉冲执行型两种形式。

如图左中程序是连续执行方式的例子。当 X2 为 ON 状态时上述指令在每个扫描周期都被重复执行。图右程序是脉冲执行方式，该指令仅在 X1 由 OFF 转为 ON 时有效。

位元件

位元件：只处理 ON/OFF 状态的元件称为位元件。

字元件：处理数据的元件称为字元件。

位元件的组合：由位元件也可构成字元件进行数据处理，位元件组合由 Kn 加首元件号来表示。4 个位元件为一组组合成单元，KnM0 中的 n 是组数。

变址寄存器 V、Z

变址寄存器在传送、比较指令中用来修改操作对象的元件号。其操作方式与普通数据寄存器一样。在[D.]中的(.)表示可以加入变址寄存器。对 32 位指令，V 作高 16 位，Z 作低 16 位。32 位指令中用到变址寄存器时只需指定 Z，这时 Z 就代表了 V 和 Z。

二、程序流程控制功能指令

条件跳转指令 CJ

子程序调用指令 CALL 与返回指令 SRET

中断返回指令 IRET、允许中断指令 EI 与禁止中断指令 DI

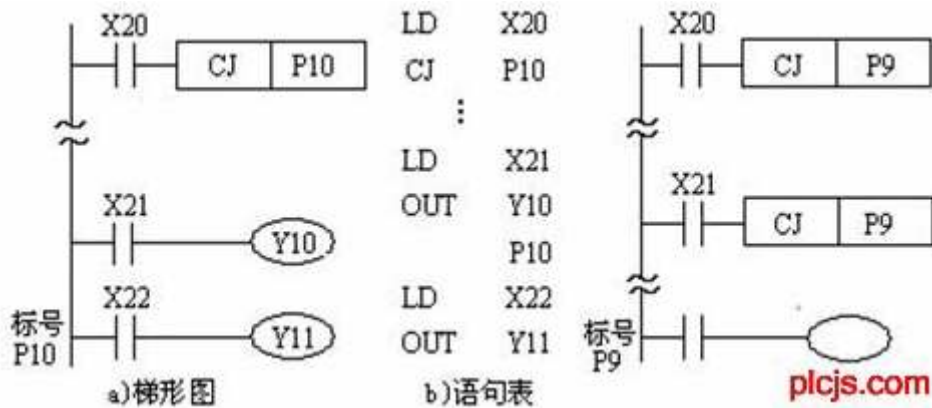
主程序结束指令 FEND

监视定时器刷新指令 WDT

循环开始指令 FOR 与循环结束指令 NEXT

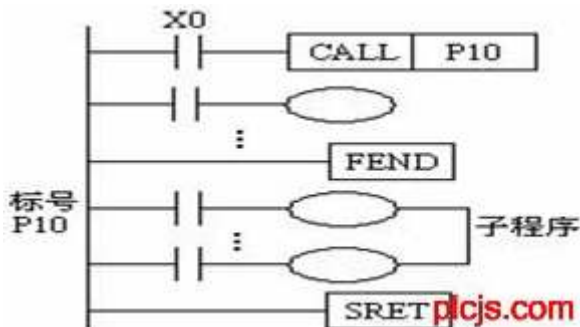
1、条件跳转指令 CJ

CJ、CJP 指令用于跳过顺序程序某一部分的场合，以减少扫描时间。条件跳转指令 CJ 应用说明如图



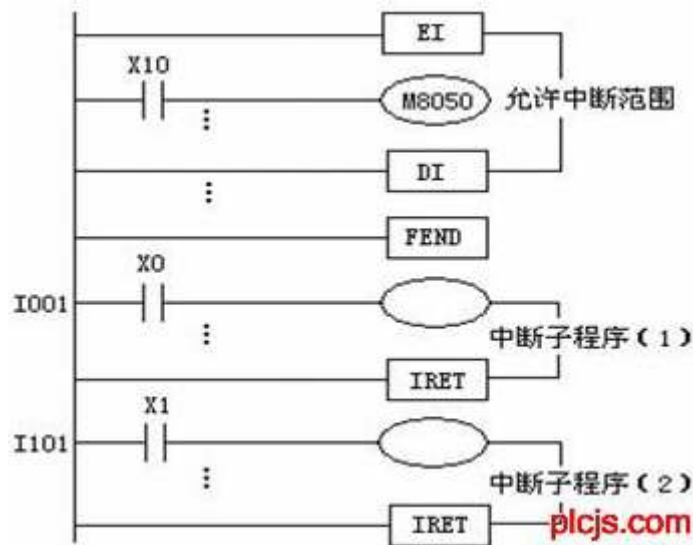
2、子程序调用指令 CALL 与返回指令 SRET

子程序应写在主程序之后，即子程序的标号应写在指令 FEND 之后，且子程序必须以 SRET 指令结束



3、中断返回指令 IRET、允许中断指令 EI 与禁止中断指令 DI

PLC 一般处在禁止中断状态。指令 EI~DI 之间的程序段为允许中断区间，而 DI~EI 之间为禁止中断区间。当程序执行到允许中断区间并且出现中断请求信号时，PLC 停止执行主程序，去执行相应的中断子程序，遇到中断返回指令 IRET 时返回断点处继续执行主程序。



4、主程序结束指令 FEND

FEND 指令表示主程序的结束，子程序的开始。程序执行到 FEND 指令时，进行输出处理、输入处理、监视定时器刷新，完成后返回第 0 步。

FEND 指令通常与 CJ-P-FEND、CALL-P-SRET 和 I-IRET 结构一起使用 (P 表示程序指针、I 表示中断指针)。CALL 指令的指针及子程序、中断指针及中断子程序都应放在 FEND 指令之后。CALL 指令调用的子程序必须以子程序返回指令 SRET 结束。中断子程序必须以中断返回指令 IRET 结束。

5、监视定时器刷新指令 WDT

如果扫描时间（从第 0 步到 END 或 FEND）超过 100ms，PLC 将停止运行。在这种情况下，应将 WDT 指令插到合适的程序步（扫描时间不超过 100ms）中刷新监视定时器。

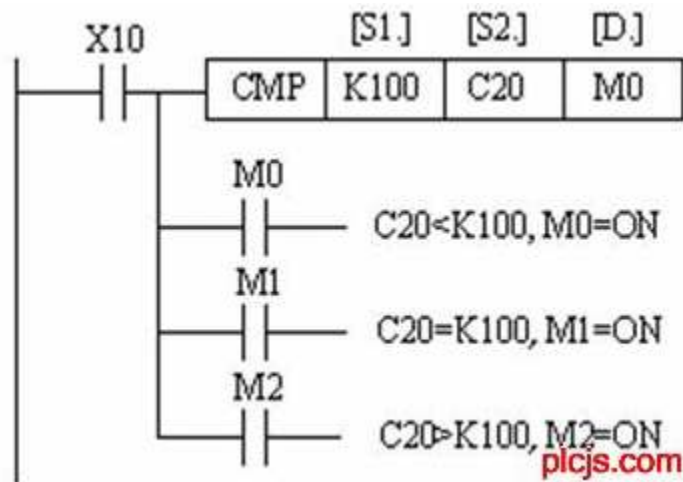
6、循环开始指令 FOR 与循环结束指令 NEXTFOR~NEXT 之间的程序重复执行 n 次（由操作数指定）后再执行 NEXT 指令后的程序。循环次数 n 的范围为 1~32767。若 n 的取值范围为-32767~0，循环次数作 1 处理。

FOR 与 NEXT 总是成对出现，且应 FOR 在前，NEXT 在后。FOR~NEXT 循环指令最多可以嵌套 5 层，利用 CJ 指令可以跳出 FOR~NEXT 循环体。

14.PLC 功能指令 传送与比较指令

1、比较指令 CMP

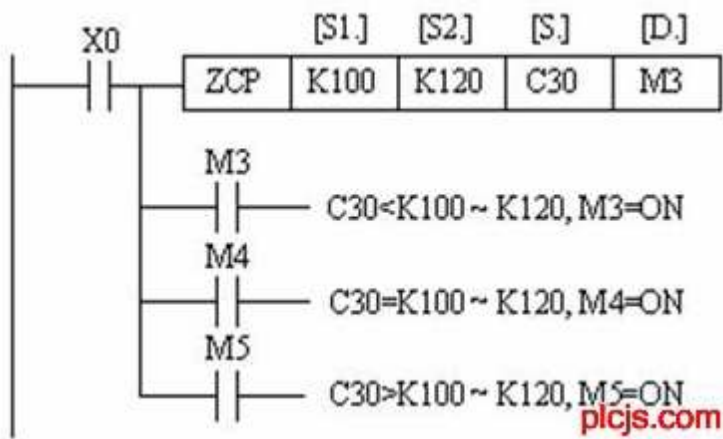
CMP 指令有三个操作数：两个源操作数[S1.]和[S2.]，一个目标操作数[D.]，该指令将[S1.]和[S2.]进行比较，结果送到[D.]中。CMP 指令使用说明如图所示。



2、区间比较指令 ZCP

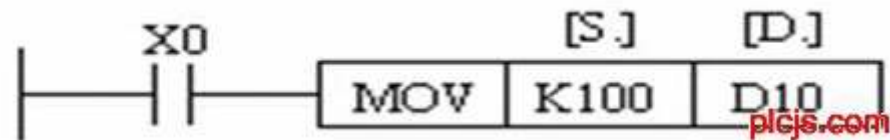
ZCP 指令是将一个操作数[S.]与两个操作数[S1.]和[S2.]形成的区间比较，且[S1.]不得大于[S2.]，结果送到[D.]中。
ZCP

指令使用说明如图所示。



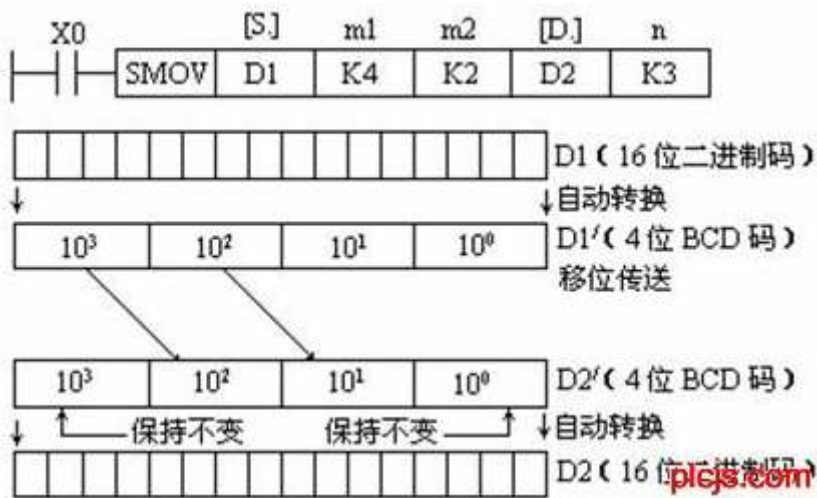
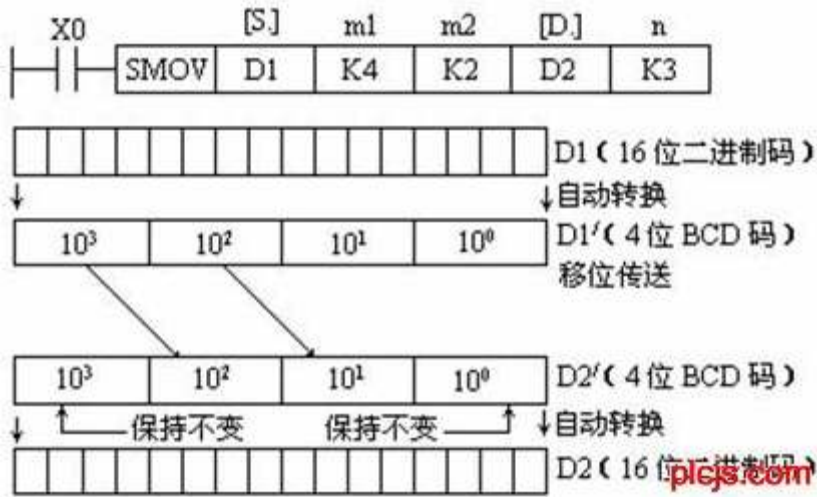
3、传送指令 MOV

MOV 指令将源操作数的数据传送到目标元件中，即[S.]→[D.]。MOV 指令的使用说明如图所示。当 X0 为 ON 时，源操作数[S.]中的数据 K100 传送到目标元件 D10 中。当 X0 为 OFF，指令不执行，数据保持不变。



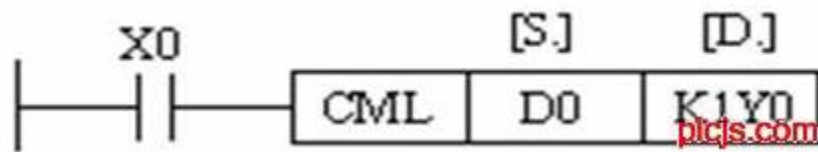
4、移位传送指令 SMOV

首先将二进制的源数据 (D1) 转换成 BCD 码，然后将 BCD 码移位传送，实现数据的分配、组合。源数据 BCD 码右起从第 4 位 (m1=4) 开始的 2 位 (m2=2) 移送到目标 D2/的第 3 位 (n=3) 和第 2 位，而 D2/的第 4 和第 1 两位 BCD 码不变。然后，目标 D2/中的 BCD 码自动转换成二进制数，即为 D2 的内容。BCD 码值超过 9999 时出错。



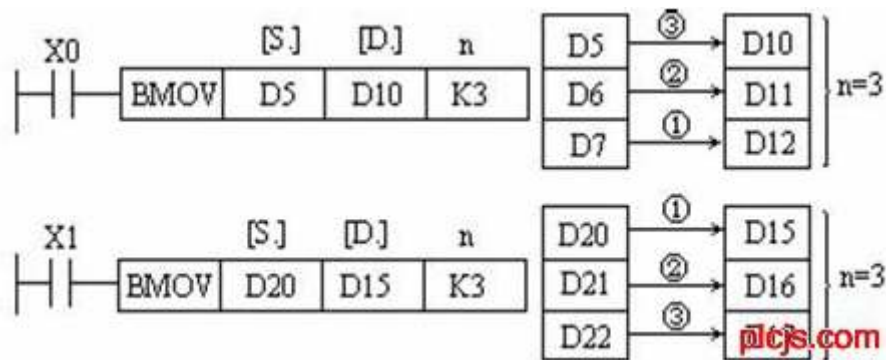
5、取反传送指令 CML

CML 指令使用说明如图所示。将源操作数中的数据（自动转换成二进制数）逐位取反后传送。



6、块传送指令 BMOV

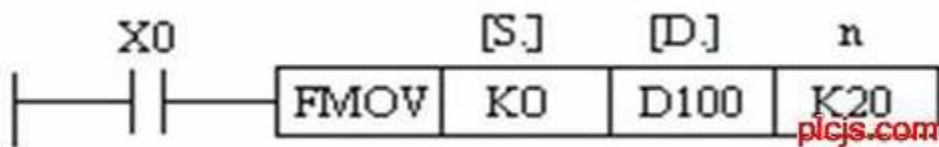
BMOV 指令是从源操作数指定的元件开始的 n 个数组成的数据块传送到指定的目标。如果元件号超出允许的元件号范围，数据仅传送到允许的范围。BMOV 指令的使用说明如图所示。



7、多点传送指令 FMOV

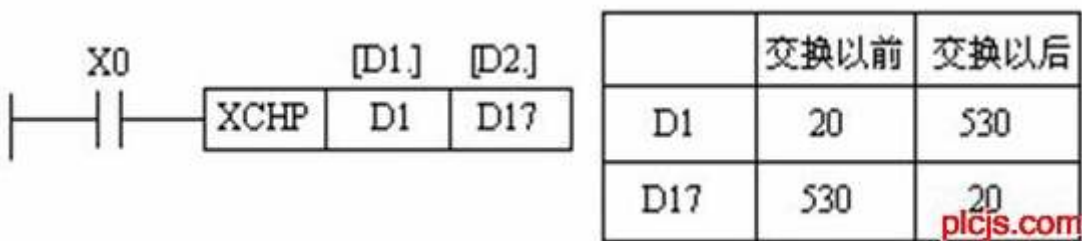
FMOV 指令是将源元件中的数据传送到指定目标开始的 n 个目标元件中，这 n 个元件中的数据完全相同。FMOV 指令使用

说明如图所示。



8、数据交换指令 XCH

XCH 指令是将两个目标元件 D1 和 D2 的内容相互交换。使用说明如图所示。



9、BCD 变换、BIN 变换指令

BCD 是将源元件中的二进制数转换为 BCD 码送到目标元件中。对于 16 位或 32 位二进制操作数，若变换结果超出 0-9999 或 0-99999999 的范围就会出错。

BCD 指令常用于将 PLC 中的二进制数变换成 BCD 码输出以驱动 LED 显示器。

BIN 是将源元件中的 BCD 码转换为二进制数送到目标元件中。常数 K 不能作为本指令的操作元件。如果源操作数不是 BCD 码就会出错。BIN 指令常用于将 BCD 数字开关的设定值输入到 PLC 中。

案例：货料小车控制

一、控制要求

某车间有 6 个工作台，送料车往返于工作台之间送料，每个工作台设有一个到位开关（SQ）和一个呼吸按钮（SB）。

具体控制要求如下：

- （1）送料车开始应能停留在 6 个工作台中任意一个到位开关的位置上。
- （2）设送料车现暂停于 m 号工作台（SQ_m 为 ON）处，这时 n 号工作台呼叫（SQ_n 为 ON），若：

(a) $m > n$, 送料车左行, 直至 SQ_n 动作, 到位停车。即送料车所停位置 SQ 的编号大于呼叫按钮 SB 的编号时, 送料车往左行运行至呼叫位置后停止。

(b) $m < n$, 送料车右行, 直至 SQ_n 动作, 到位停车。即送料车所停位置 SQ 的编号小于呼叫按钮 SB 的编号时, 送料车往右运行至呼叫位置后停止。

(c) $m = n$, 送料车原位不动。即送料车所停位置 SQ 的编号与呼叫按钮 SB 的编号相同时, 送料车不动。

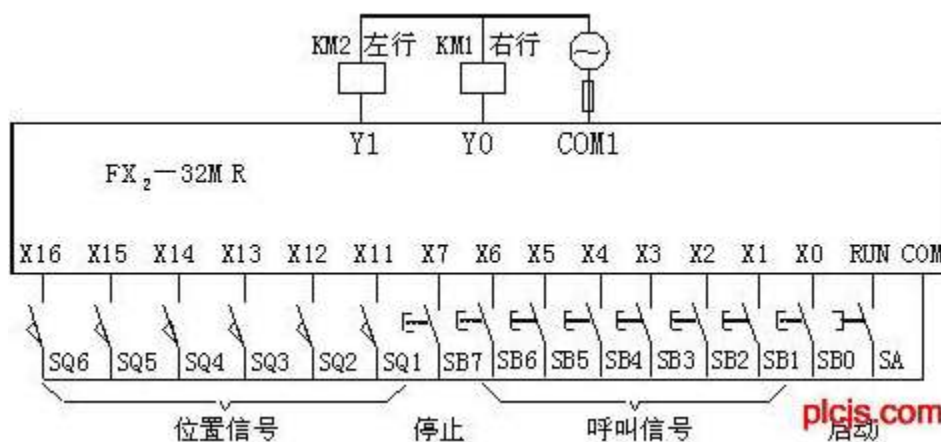
二、PLC 硬件的实现

1、I/O 的分配表

输入		功能说明	输出		功能说明
SB0	X0	启动	KM1	Y0	左行
SB1	X1	呼叫1	KM2	Y1	右行
SB2	X2	呼叫2			
SB3	X3	呼叫3			
SB4	X4	呼叫4			
SB5	X5	呼叫5			
SB6	X6	呼叫6			
SB7	X7	呼叫7			
SQ1	X11	限位1			
SQ2	X12	限位2			
SQ3	X13	限位3			
SQ4	X14	限位4			
SQ5	X15	限位5			
SQ6	X16	限位6			

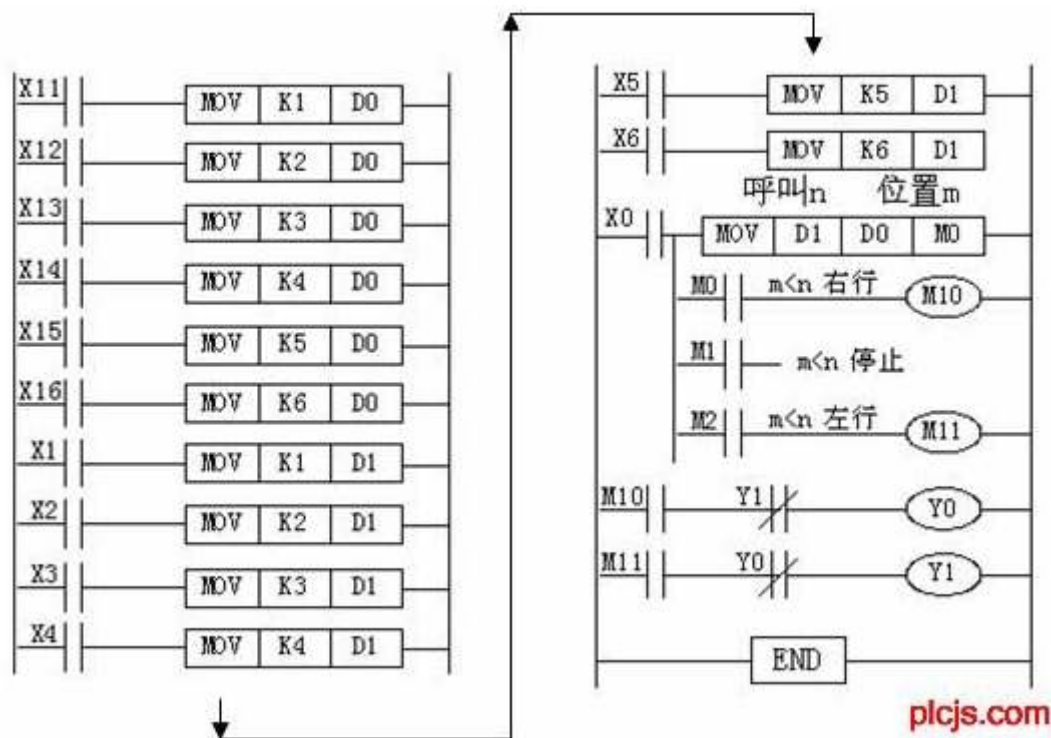
plcjs.com

2、I/O 的外部接线



plcjs.com

三、PLC 软件的实现



图中将送料车当前位置送到数据寄存器 D0 中，将呼叫工作台号送到数据寄存器 D1 中，然后通过 D0 与 D1 中数据的

比较，决定送料车的运行方向和到达的目标位置。

15. PLC 功能指令 算术和逻辑运算指令

算术运算和逻辑运算指令

加法指令 ADD、减法指令 SUB

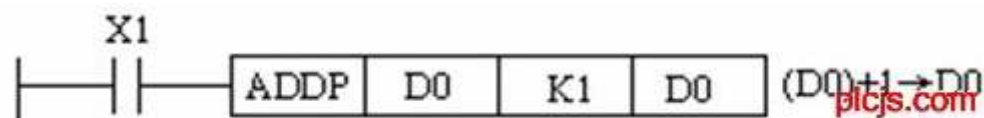
乘法指令 MUL、除法指令 DIV

加 1 指令 INC、减 1 指令 DEC

字逻辑运算指令(FNC26~FNC29)

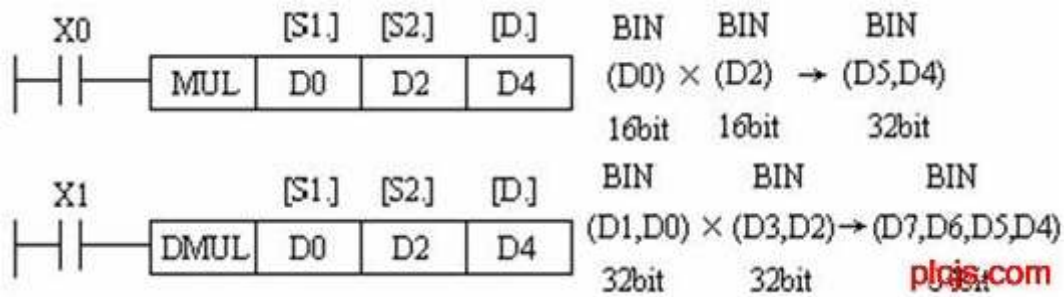
1、加法指令 ADD、减法指令 SUB

ADD 指令是将指定的源元件中的二进制数相加,结果送到指定的目标元件中去。每个数据的最高位作为符号位(0 为正, 1 为负), 运算是二进制代数运算。减法指令 SUB 与 ADD 指令类似。



2、乘法指令 MUL、除法指令 DIV

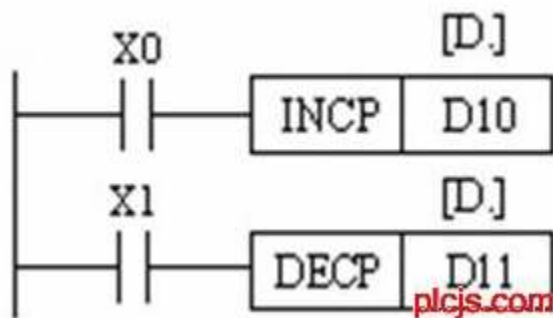
MUL 指令是将两个源元件中的数据的乘积送到指定目标元件。如果为 16 位数乘法, 则乘积为 32 位, 如果为 32 位数乘法, 则乘积为 64 位, 如图 5.25 所示。数据的最高位是符号位。



3、加 1 指令 INC、减 1 指令 DEC

INC、DEC 指令操作数只有一个，且不影响零标志、借位标志和进位标志。

在 16 位运算中，32767 再加 1 就变成了-32768。32 位运算时，2147483647 再加 1 就变成-2147483648。DEC 指令与 INC 指令处理方法类似。



4、字逻辑运算指令(FNC26~FNC29)

字逻辑运算指令包括 WAND(字逻辑与)、WOR（字逻辑或）、WXOR（字逻辑异或）和 NEG（求补）指令。使用方法如图所示。



案例一：

假设有一汽车停车场，最大容量只能停车 50 辆，为了表示停车场是否有空位，试用 PLC 来实现控制。

一、硬件的实现

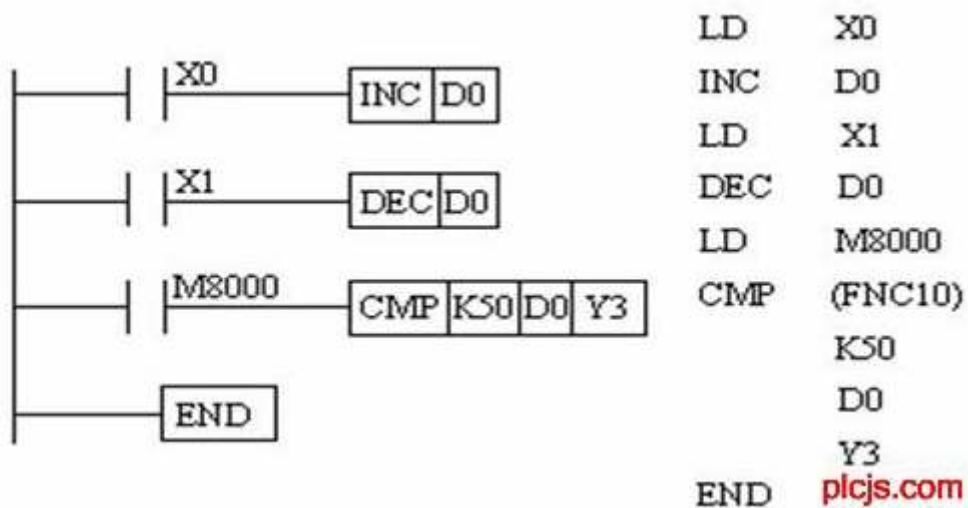
1、I/O 分配表

器件	PLC地址	功能说明
HL1	Y4、Y5	停车场已满
HL2	Y3	停车场有空位
开关0	X0	车已进入停车场信号
开关1	X1	车已离开停车场信号
	D0	停车场车辆数（最大50辆）

2、外部连线

根据上述模块的学习，再依据图中的标示，同学们可以自己画出输入及输出端口的分配，在这不加深述。

二、软件的编程



案例二：

某控制程序中要进行以下算式的运算： $38X/255+2$ 式中“X”代表输入端口 K2X0 送入的二进制数，运算结果需送输出口 K2Y0；X020 为起停开关。

一、硬件的实现

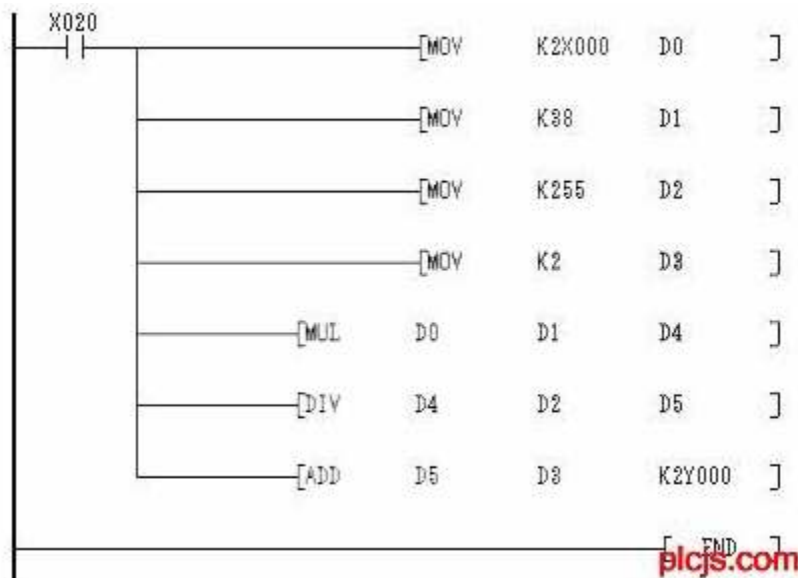
1、I/O 的分配

输入		功能说明	输出		功能说明
K2X0	X0	二进制数输入	K2Y0	Y0	二进制数输出
	X1			Y1	
	X2			Y2	
	X3			Y3	
	X4			Y4	
	X5			Y5	
	X6			Y6	
	X7			Y7	
	X20	启动			plcjs.com

2、硬件连线

根据上述模块的学习，再依据图中的标示，同学们可以自己画出输入及输出端口的分配，在这不加深述。

二、软件的实现



16.PLC 功能指令 循环移位与移位指令

循环移位与移位指令

右循环移位指令 ROR、左循环移位指令 ROL

带进位循环右移指令 RCR、带进位循环左移指令 RCL

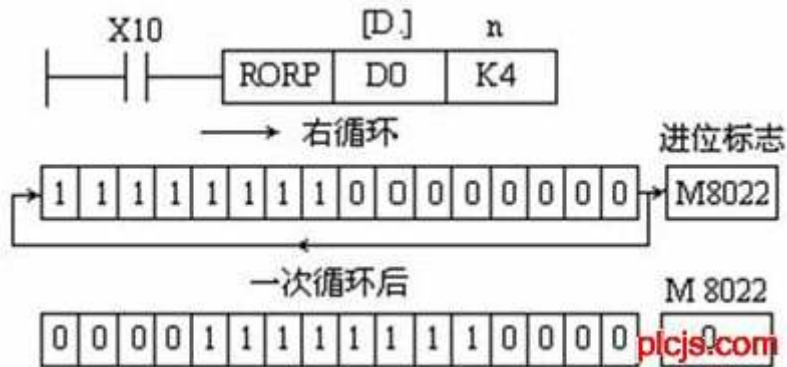
位右移位指令 SFTR、位左位指令 SFTL

字右移位指令 WSFR、字左移位指令 WSFL

先入先出(FIFO)写入指令 SFWR、读出指令 SFRD

1、右循环移位指令 ROR、左循环移位指令 ROL

助记符	功能	操作数		程序步
		[D.]	n	
ROR FNC30 循环右移	把目标元件的位循环右移n次	KnY、 KnM、 KnS、T、 C、D、 V、Z	K、H 16位操作 : n ≤ 16 32位操作 : n ≤ 32	ROR、RORP、 ROL、ROLP: 5步 DROR、DRORP、 DROL、DROLP: 9步 plcjs.com
ROL FNC31 循环左移	把目标元件的位循环左移n次			

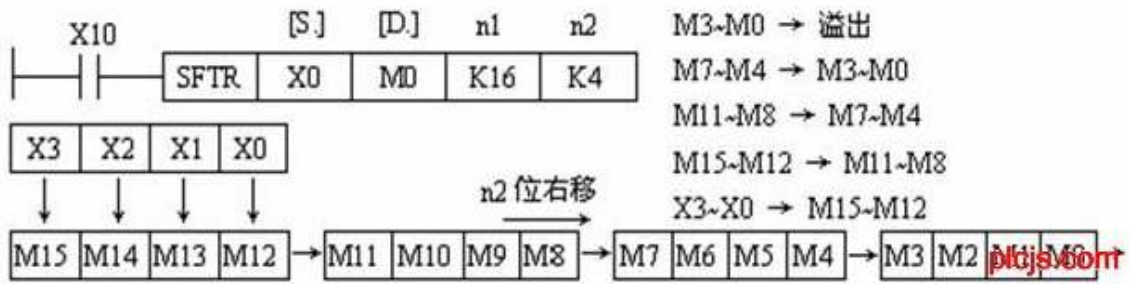


2、带进位循环右移指令 RCR、带进位循环左移指令 RCL

助记符	功能	操作数		程序步
		[D.]	n	
RCR FNC32 带进位右移	把目标元件的位和进位一起右移n位	KnY、 KnM、 KnS、T、 C、D、 V、Z	K、H 16位操作 : n ≤ 16 32位操作 : n ≤ 32	RCR、RCRP、 RCL、RCLP: 5步 DRCR、DRCRP、 DRCL、DRCLP: 9步 plcjs.com
RCL FNC33 带进位左移	把目标元件的位和进位一起左移n位			

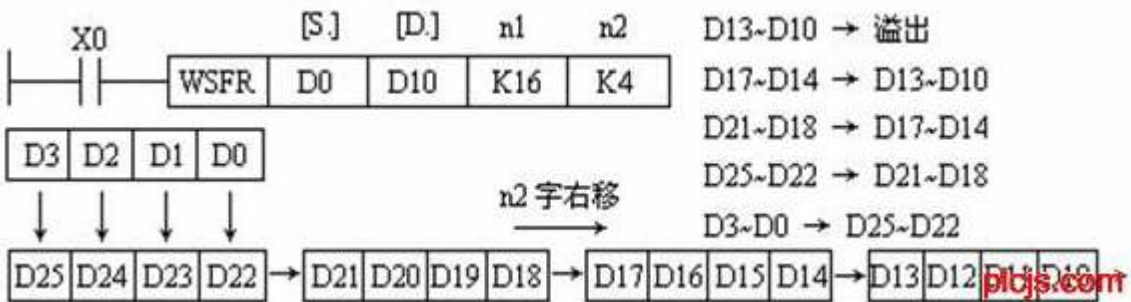
执行 RCR、RCL 指令时，各位的数据与进位位 M8022 一起（16 位指令时一共 17 位）向右（或向左）循环移动 n 位。循环中移出的位送入进位标志，后者又被送回到目标操作元件的另一端。

助记符	功能	操作数				程序步
		[S.]	[D.]	n1	n2	
SFTR FNC34 带进位右移	把源元件状态存放到堆栈中，堆栈右移	X Y M S	Y M S	K、H n2 ≤ n1 ≤ 1024		SFTR、 SFTRP、 SFTL、 SFTLP: 9步 plcjs.com
SFTL FNC35 带进位左移	把源元件状态存放到堆栈中，堆栈左移					



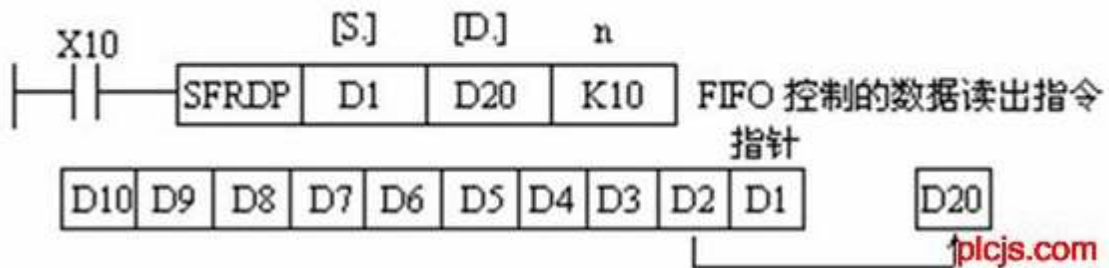
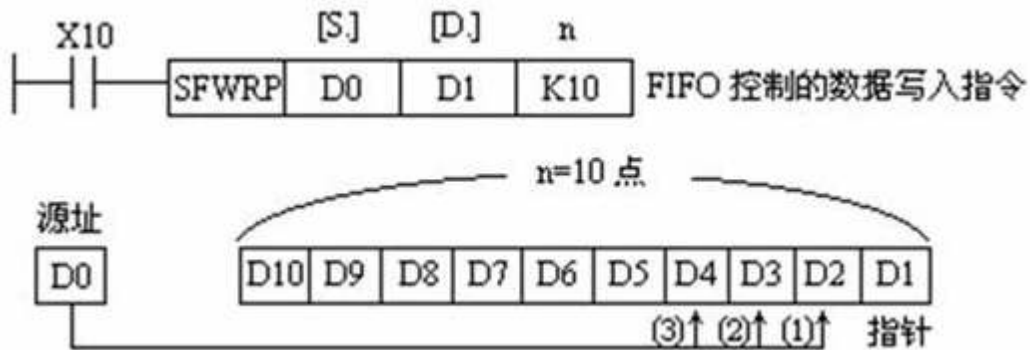
4、字右移位指令 WSFR、字左移位指令 WSFL

助记符	功 能	操 作 数				程 序 步
		[S.]	[D.]	n1	n2	
WSFR FNC36 字右移	把源元件状态存放到字栈中，堆栈右移	KnX 、 KnY 、 KnM	KnY 、 KnM 、 KnS	K、H $n2 \leq n1 \leq 512$		WSFR、 WSFRP、 WSFL、 WSFLP: 9步
WSFL FNC37 字左移						



5、先入先出(FIFO)写入指令 SFWR、读出指令 SFRD

助记符	功能	操作数			程序步
		[S.]	[D.]	n	
SFWR FNC38 FIFO写入	创建长度为n位FIFO堆栈，与SFRD指令一起使用	K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z	KnY、KnM、KnS、T、C、D	K、H $2 \leq n \leq 5$ 12	SFWR、SFWRP、SFRD、SFRDP: 7步 plcjs.com
SFRD FNC39 FIFO读出	读FIFO，长度减1，与SFWR指令一起使用	KnY、KnM、KnS、T、C、D	KnY、KnM、KnS、T、C、D、V、Z		



17.PLC 功能指令、方便指令和外部 I/O 设备指令

方便指令

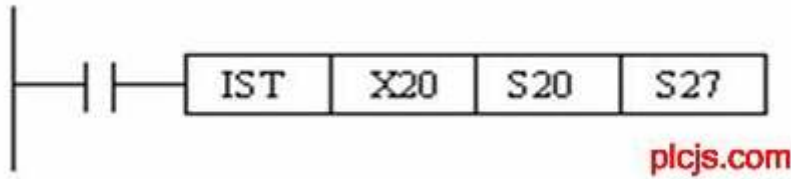
初始状态指令 IST

特殊定时器指令 STMR

交替输出指令 ALT

一、初始状态指令 IST

状态初始化指令 IST 与 STL 指令一起使用，用于自动设置多种工作方式的系统的顺序功能图。IST 指令只能使用一次，它应放在程序开始的地方，被它控制的 STL 电路应放在它的后面。



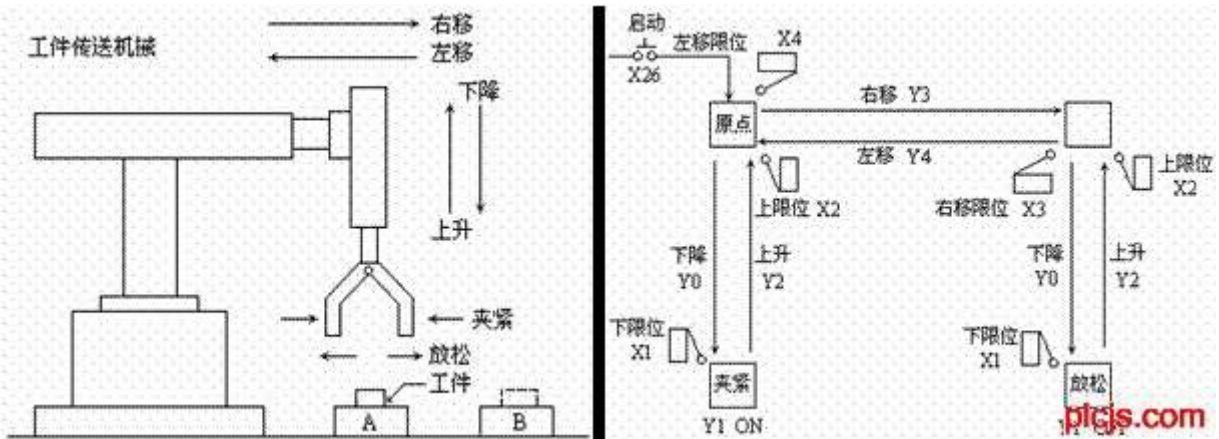
IST 中的源操作数可取 X、Y 和 M，图中 IST 指令的源操作数 X20 用来指定与工作方式有关的输入继电器的首元件，它实际上指定从 X20 开始的 8 个输入继电器，这 8 个输入继电器的意义如下表。

输入继电器X	功 能	输入继电器X	功 能
X20	手动	X24	连续运行
X21	回原点	X25	回原点启动
X22	单步运行	X26	自动启动
X23	单周期运行	X27	停止

IST 指令的执行条件满足时，初始状态继电器 S0~S2 和下列特殊辅助继电器被自动指定为以下功能(如下表)，以后即使 IST 指令的执行条件变为 OFF，这些元件的功能仍保持不变。

特殊辅助继电器M	功 能	状态继电器S	功 能
M8040	禁止转换	S0	手动操作初始状态继电器
M8041	转换启动	S1	回原点初始状态继电器
M8042	启动脉冲	S2	自动操作初始状态继电器
M8043	回原点完成		
M8044	原点条件		
M8047	STL监控有效		

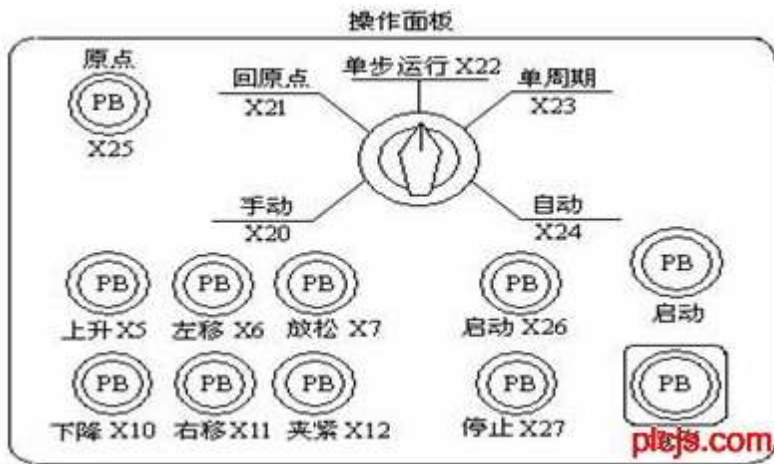
案例 机械手控制



1、系统工作过程分析

机械手将工件从 A 点向 B 点传送。机械手工作原点在左上方，按下降、夹紧、上升、右移、下降、松开、上升、左移的顺序依次运动。它有手动、单步、一个周期和连续工作（自动）四种操作方式。

简易机械手的操作面板如图。



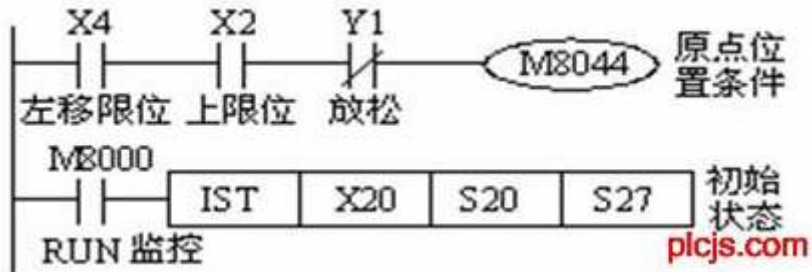
2、硬件设计

略

3、软件设计

1) 初始化程序

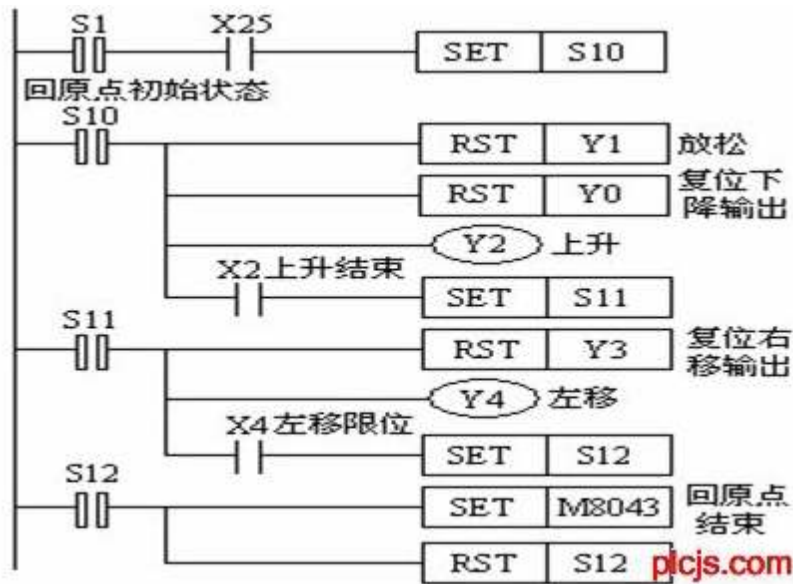
设置初始状态和原点位置条件



2) 手动方式程序



3) 回原点方式程序

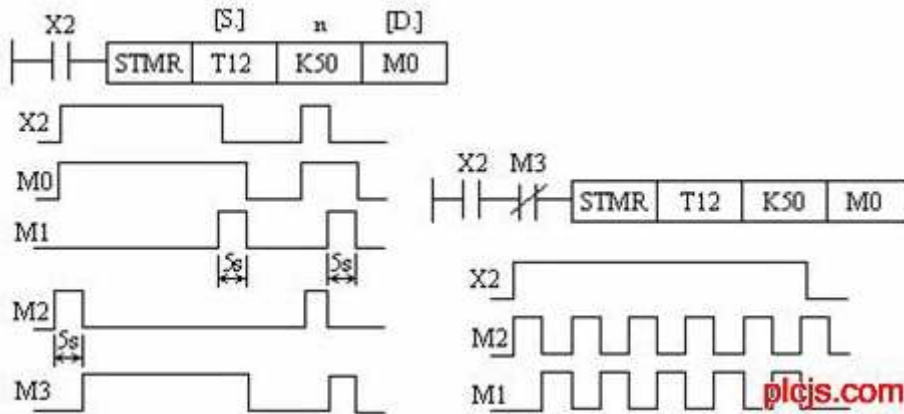


4) 自动方式程序

请同学们自行设计

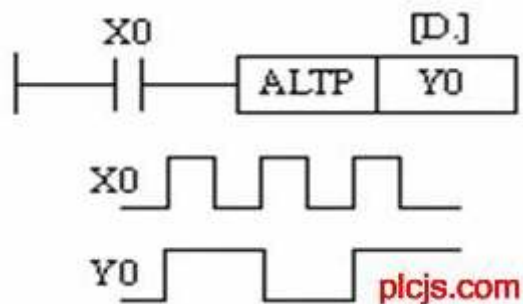
二、特殊定时器指令 STMR

特殊定时器指令用来产生延时断开定时器、脉冲定时器和闪烁定时器。该指令使用说明如图所示。n 用来指定定时器的设定值，图中 T12 的设定值为 5s(n=50)。图中的 M0 是延时断开定时器，M1 是 X2 由 ON→OFF 的单脉冲定时器，M2 和 M3 是为闪动而设的。



三、交替输出指令 ALT

ALT 指令使用说明如图所示。X0 由 OFF 变为 ON 时，Y0 的状态改变一次，若不用脉冲执行方式，每个扫描周期 Y0 的状态都要改变一次。ALT 指令具有分频器的效果，使用 ALT 指令，用 1 只按钮 X0 就可以控制 Y0 对应的外部负载的起停。



外部 I/O 设备指令

一、七段译码指令 SEGD

SEGD 指令助记符、功能、操作数、程序步如表所示：

助记符	功能	操作数		程序步
		[S.]	[D.]	
SEGD	十六进制数译为七段显示代码	K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z	KnY、KnM、KnS、T、C、D、V、Z	SEGD、SEGD P: 5步 plcjs.com



注：将 S 指定的元件低 4 位中的十六进制数译码后送给 7 段显示器，译码信号存于 D 指定的元件中，输出要占 7 个输出点

案例

用按钮 SB0 控制 LED 数码管顺序显示数字 0 到 F。

在系统刚开始运行时,LED 数码管显示数字“0”；

第 1 次闭合按钮 SB0 时,LED 数码管显示数字“1”；

第 2 次闭合按钮 SB0 时,LED 数码管显示数字“2”；

.....

第 10 次闭合按钮 SB0 时,LED 数码管显示数字“A”；

.....

第 15 次闭合按钮 SB0 时,LED 数码管显示数字“F”；

此后再闭合按钮 SB0,LED 数码管显示数字“F”不变；

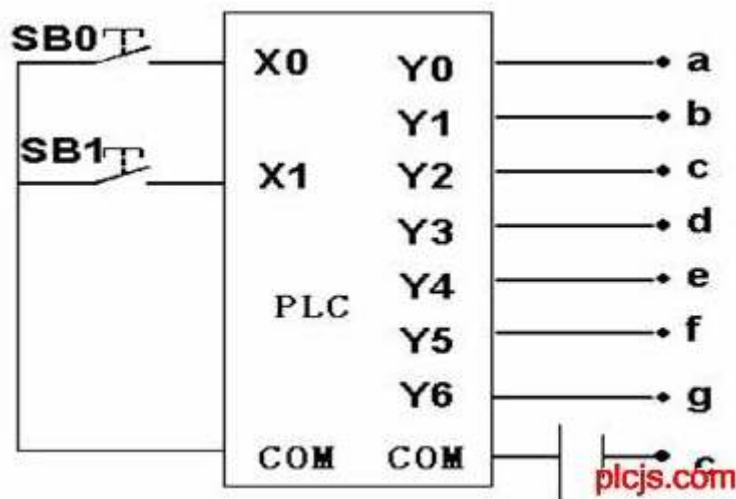
按钮 SB1 做复位用，闭合 SB1,LED 数码管显示数字“0”。

断开 SB1,再闭合 SB0 时，LED 数码管显示“1”

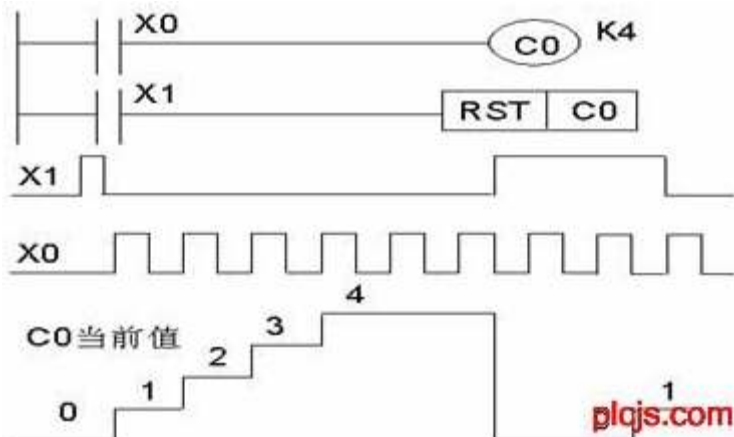
.....

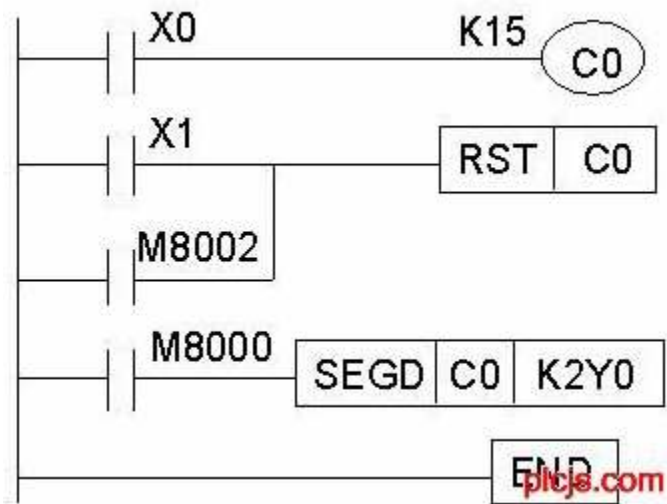
1、硬件设计

LED显示控制PLC的I/O点分配表		
PLC点名称	连接的外部设备	功能说明
X0	SB0	控制信号
X1	SB1	复位信号
Y0	LED数码管a段	使LED数码管a段亮
Y1	LED数码管b段	使LED数码管b段亮
Y2	LED数码管c段	使LED数码管c段亮
Y3	LED数码管d段	使LED数码管d段亮
Y4	LED数码管e段	使LED数码管e段亮
Y5	LED数码管f段	使LED数码管f段亮
Y6	LED数码管g段	使LED数码管g段亮



2、软件

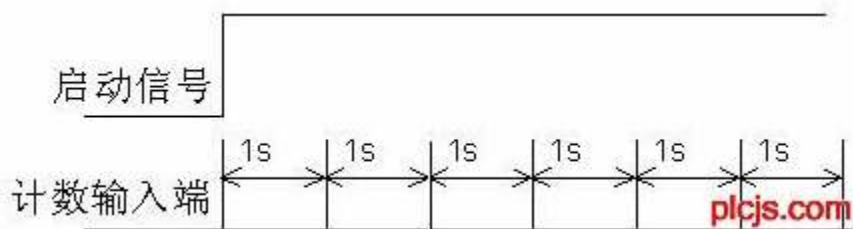




思考题:

如果让你实现下面的控制该如何实现?

按下启动按钮后, 自动循环显示 0 到 F. 显示间隔 1s。



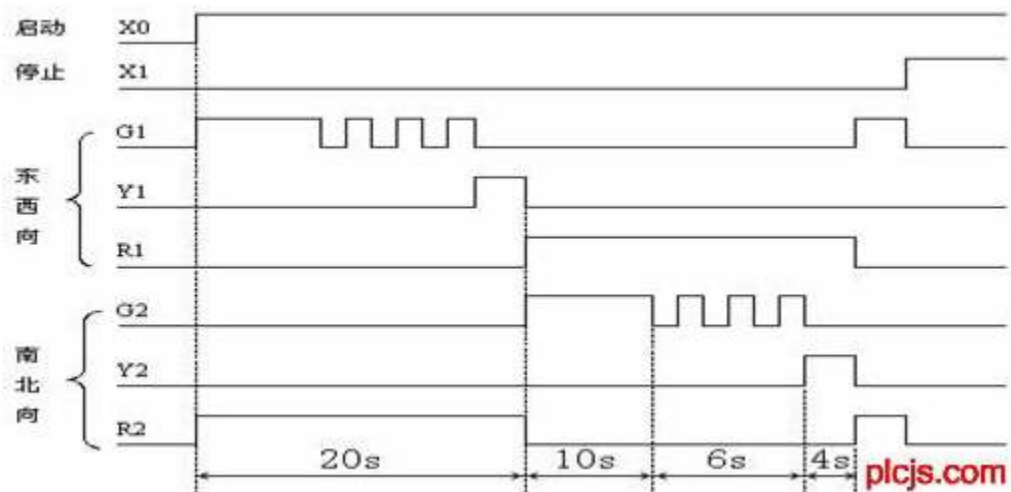
18. 可编程控制器控制系统设计

可编程控制器控制系统设计

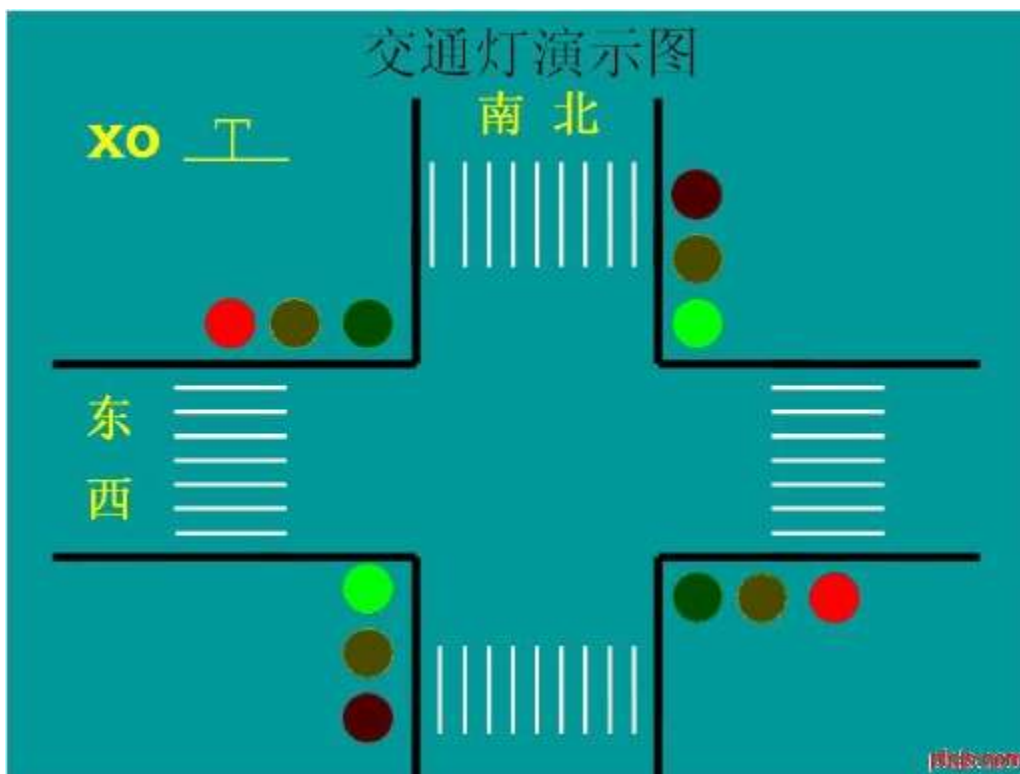
一、 可编程控制器系统设计的步骤和内容

案例: 交通信号灯的自动控制系统设计

1、 信号转换关系



2、交通等演示图

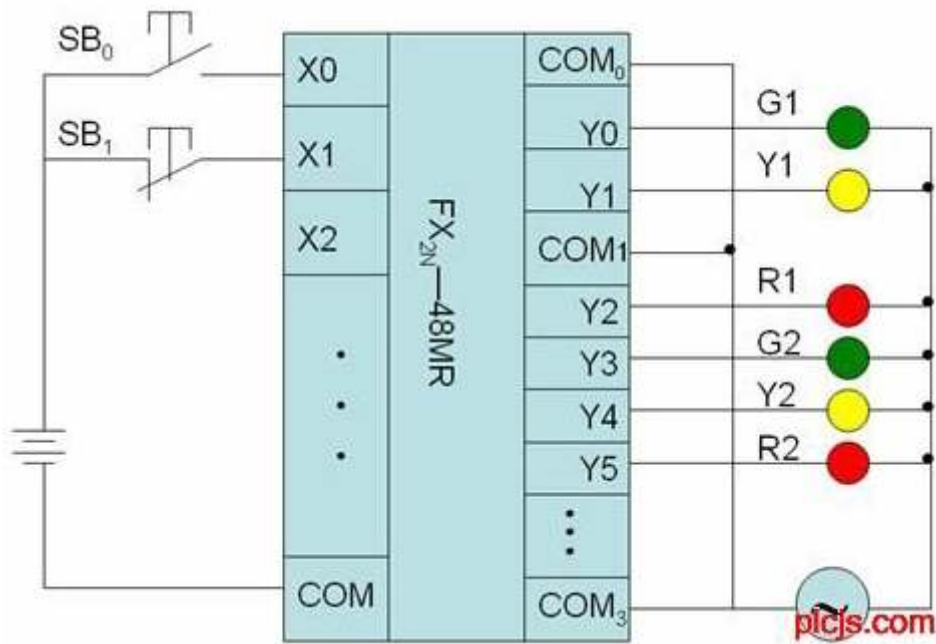


3、分析工况，思考问题：

1. 该控制系统是否属于顺序控制？是哪种结构？
2. 该控制系统工作步如何划分？共有多少工作步？
3. 输入/输出信号有那些？
4. PLC 的 I/O 分配

输 入			输 出		
器件	器件号	功能说明	器件	器件号	功能说明
0	X0	启动按钮	G1	Y0	东西向绿灯
1	X1	停止按钮	Y1	Y1	东西向黄灯
			R1	Y2	东西向红灯
			G2	Y3	南北向绿灯
			Y2	Y4	南北向黄灯
			R2	Y5	南北向红灯

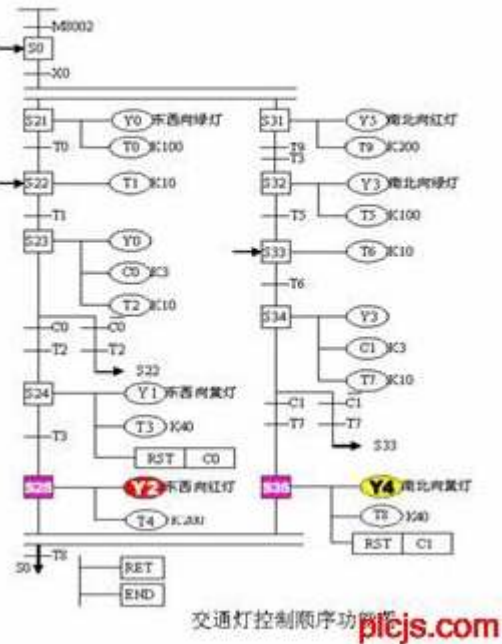
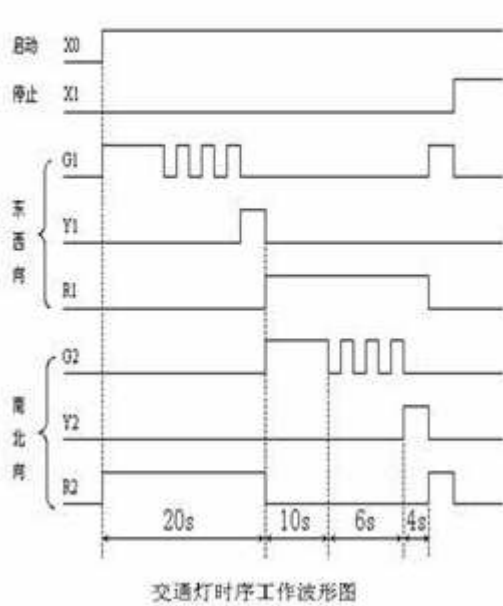
5、PLC 硬件设计



6、PLC 顺序功能图设计

交通信号灯的自动控制系统设计

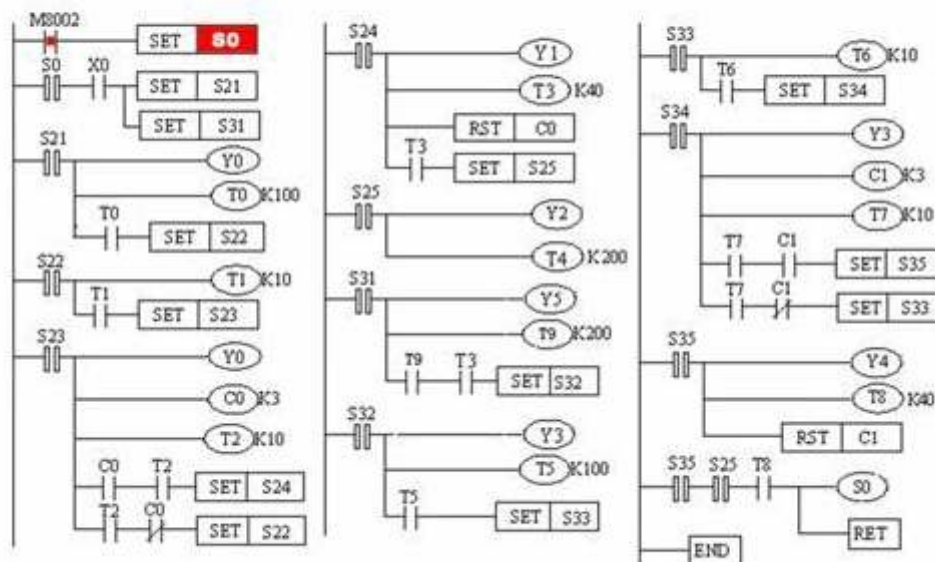
—— 顺序功能图的设计



7、PLC 梯形图程序设计

交通信号灯的自动控制系统设计

—— 步进梯形图设计



plcjs.com

8、综述 PLC 系统设计的步骤

- 1) 熟悉控制对象的工艺要求（工况分析）
- 2) 电器控制线路的设计
- 3) 程序设计
- 4) 控制系统模拟调试
- 5) 现场调试
- 6) 技术文件整理

二、 可编程控制器的选型与硬件配置

（一）选择合适的可编程控制器类型

1、可编程控制器选择

三菱 FX1S 系列可编程控制器是一种卡片大小的 PLC，适合在小型环境中进行控制。它具有卓越的性能、串行通讯功能以及紧凑的尺寸，这使得它们能用在以前常规可编程控制器无法安装的地方。三菱 FX1N 系列可编程控制器是一种普遍选择方案，最多可达 128 点控制。由于 FX1N 系列具有对于输入/输出、逻辑控制以及通讯 / 链接功能的可扩展性，因此它对普遍的顺控解决方案有广泛的适用范围，并且能增加特殊功能模块或扩展板。

三菱 FX2N 系列可编程控制器是 FX 系列中最高级的模块。它拥有无以匹及的速度、高级的功能、逻辑选件以及定位控制等特点，FX2N 是从 16 到 256 路输入 / 输出的多种应用的选择方案。三菱 FX2NC 系列可编程控制器在保留其原有的强大功能特色的前提下实现了极为可观的规模缩小，I / O 型连接口降低了接线成本并节省了时间。对于开关量控制的系统，当控制速度要求不高时，一般的小型整体机 FX1S 就可以满足要求。对于以开关量控制为主，带有部分模拟量控制的应用系统，应选择具有所需功能的可编程控制器主机，如用 FX1N 或 FX2N 型整体机。另外还要根据需要选择相应的模块，例如开关量的输入/输出模块、模拟量输入/输出模块、配接相应的传感器及变送器和驱动装置等。

2、I/O 点数的确定

一般的讲，可编程控制器控制系统的规模的大小是用输入、输出的点数来衡量的。我们在设计系统时，应准确统计被控对象的输入信号和输出信号的总点数并考虑今后调整和工艺改进的需要，在实际统计 I/O 点数基础上，一般应加上 10%—20% 的备用量。对于整体式的基本单元，输入输出点数是固定的，不过三菱的 FX 系列不同型号输入/输出点数的比例也不同，根据输入/输出点数的比例情况，可以选用输入/输出点都有的扩展单元或模块，也可以选用只有输入（输出）点的扩展单元或模块。

3、用户存储器容量的估算

根据经验，对于开关量控制系统，用户程序所需存储器的容量等于 I/O 信号总数乘以 8。对于有模拟量输入输出的系统，每一路模拟量信号大约需 100 存储器容量。如果使用通信接口，那么每个接口需 300 存储器容量。一般估算时根据算出存储器的总字数再加上一个备用量。

4、可编程控制器的处理速度应满足实时控制的要求

可编程控制器是采用顺序扫描的工作方式，其顺序扫描工作方式使它不能可靠地接收持续时间小于 1 个扫描周期的输入信号。为此，对于快速反映的信号需要选取扫描速度高的机型关于可编程控制器的选型问题，当然还应考虑到它的的联网通信功能、价格等因素。系统可靠性也是考虑的重要因素。

（二）开关量输入输出模块及扩展的选择

开关量输入模块的输入电压一般为 DC24V 和 AC220V 两种。直流输入可以直接与接近开关、光电开关等电子输入装置连接，三菱 FX 系列直流输入模块的公用端已经接在内部电源的 0V，因此直流输入不需要外接直流电源，有些类型的可编程控制器输入的公用端要另接电源，对初学者应该注意。交流输入方式的触点接触可靠，适合于在有油雾、粉尘的恶劣环境下使用。我们最常用的还是直流输入模块。

开关量输出模块有继电器输出、晶体管输出及可控硅输出。继电器型输出模块的触点工作电压范围广，导通压降小，承受瞬时过电压和过电流的能力较强，但是动作速度较慢，寿命(动作次数)有一定的限制。一般控制系统的输出信号变化不是很频繁，我们优先选用继电器型，并且继电器输出型价格最低，也容易购买。晶体管型与双向可控硅型输出模块分别用于直流负载和交流负载，它们的可靠性高，反应速度快，寿命长，但是过载能力稍差。选择时应考虑负载电压的种类和大小、系统对延迟时间的要求、负载状态变化是否频繁等，还应注意同一输出模块对电阻性负载、电感性负载和白炽灯的驱动能力的差异。

（三）编程器和外围设备的选择

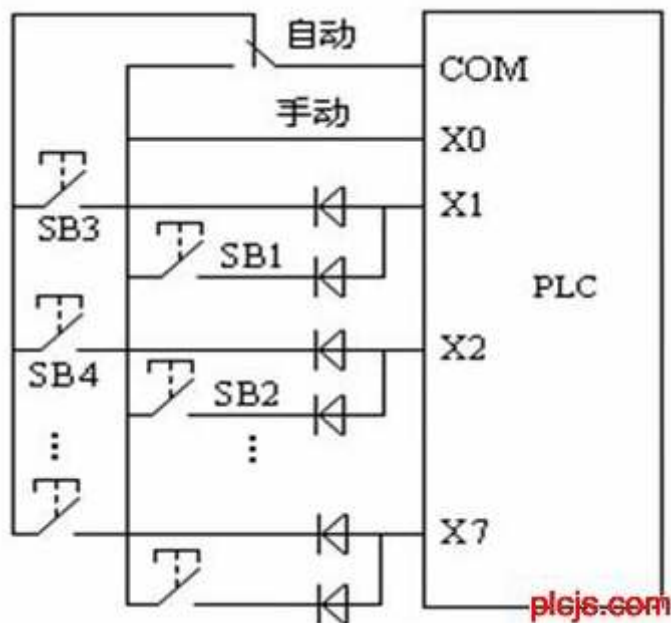
早期的小型可编程控制系统，通常都选用价格便宜的简易编程器。如果系统较大，可编程控制器多，可以选用一台功能强、编程方便的图形编程器；随着科技的发展，个人计算机的使用越来越普及，编程软件包的出现，在个人计算机上安装的编程软件包配上通信电缆，也可取代原编程器。

三、节省 I/O 点数的方法

（一）减少所需输入点数的方法

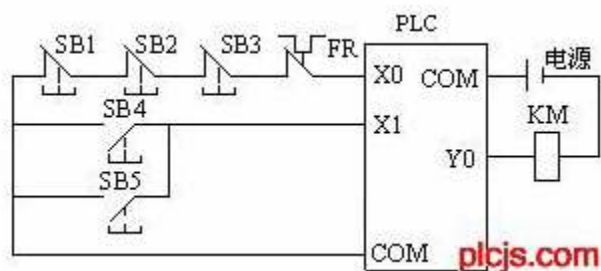
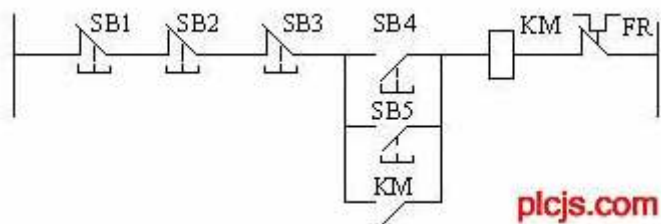
1、分组输入

多设备都有自动控制和手动控制两种状态，自动程序和手动程序不会同时执行，把自动和手动信号叠加起来，按不同控制状态要求分组输入到可编程控制器，可以节省输入点数。例如：电梯轿箱内的操纵箱内一般都设有检修运行的手动上、下按钮，也有自动运行的选层按钮，现在很多电梯在设计时就是利用最底层选层按钮和最顶层的选层按钮取代检修手动上、下按钮，这样不仅节省了输入点，同时还减少了两个按钮，进一步降低了成本。



2、触点合并式输入

修改外部电路，将某些具有相同功能的输入触点串联或并联后再输入可编程控制器，这些信号就只占用一个输入点了。串联时，几个开关同时闭合有效。并联时，其中任何一个触点闭合都有效。例如一般设备控制时都有很多保护开关，任何一个开关动作都要设备停止运行，这样在设计时就可以将这些开关串联在一起，用一个输入点。对同一台设备的多点控制一般将多点的控制按钮并联在一起，用一点输入，如图。



3、矩阵式输入

当可编程控制器有两个以上富余的输出端点时，可将二极管开关矩阵的行、列引线分别接到 I/O 端点上。这样，当矩阵为 n 行 m 列时，可以得到 $n \times m$ 个输入信号供可编程控制器组成的控制系统使用。对于 FX2N 系列，使用矩阵输入指令 MTR，只用 8 个输入点和 8 个输出点，就可以输入 64 个输入点的状态。

4、充分利用可编程控制器的内部功能

使用 KEY 指令，只需 4 个输入点，4 个输出点就可以输入 10 个数字键和 6 个功能键；使用 DSW 指令，只需 4 个或 8 个输入点，4 个输出点就可以读入一个或两个 4 位 BCD 码数字开关信息。利用转移指令，在一个输入端上接一开关，作为自动、手动工作方式转换开关，用转移指令，可将自动和手动操作加以区别。利用计数器计数，或利用移位寄存器移位，也可以利用交替输出指令实现单按钮的启动和停止。

（二）减少所需输出点数的方法

1. 通断状态完全相同的负载并联后，可以共用 PLC 的一个输出点，即一个输出点带多个负载，如果多个负载的总电流超出输出点的容量，可以用一个中间继电器在控制其它负载。
2. 在采用信号灯做负载时，采用数码管做指示灯可以减少输出点数。例如电梯的楼层指示，如果使用信号灯，则一层就要一个输出点，楼层越高占用输出点越多，现在很多电梯使用数字显示器显示楼层就可以节省输出点，常用的是用 BCD 码输出，9 层站以下仅用 4 个输出点，10 到 19 层仅用 5 个输出点。FX2N 系列 7 段译码指令 SEGD 可把十六进制数译为七段显示器所需的代码，直接控制一只七段显示器，用 7 个输出点；还有一些数字显示的指令，可以减少输出点的数量。

四、 可编程控制器应用中需注意的若干问题

可编程控制器是专门为工业生产环境设计的控制装置，一般不需要采取什么特殊措施便可直接用于工业环境，但是，如果环境过于恶劣，电磁干扰特别强烈，或安装使用不当，都不能保证系统的正常安全运行，为了保证其正常安全运行和提高系统

的可靠性和稳定性，我们在应用可编程控制器时还要注意以下问题。

（一）工作环境

1. 温度

一般情况下可编程控制器的四周环境温度不应低于 0 或高于 60，最好不高于 45，否则应采取通风或其他保温措施。

2. 湿度

为了保证可编程控制器的绝缘性能，其周围的湿度应保持在 35~80%RH 范围内。

3. 振动

可编程控制器不应在具有频繁振动、连续振动(频率为 10~55Hz，振幅大于 0.5mm)或超过 10g 的冲击加速度的环境下工作，否则应采取防振或减振措施。

4. 介质

可编程控制器不应安装在充满导电尘埃、油物或有机溶剂、腐蚀性气体的环境下工作，否则应将控制柜做成封闭结构或对柜内气体采取净化措施。

（二）安装布线

可编程控制器在安装时应注意以下事项：

1. 为了提供足够的通风空间，保证 PLC 正常的工作温度，基本单元与扩展单元之间要留 30mm 以上间隙，各 PLC 单元与其他电器元件之间要留 100mm 以上间隙，以避免电磁干扰；
2. 安装时远离高压电源线和高压设备，它们之间要留 200mm 以上间隙，高压线、动力线等应避免与输入输出线平行布置；
3. 安装时远离加热器、变压器、大功率电阻等发热源，必要时安装风扇；
4. 远离产生电弧的开关、继电器等设备；控制柜内部的布线，主要是指 PLC 的电源、接地、输入、输出、通信等接线端子到各输出端子板或柜内其他电器元件之间的连接。布线时应该注意：各种类型的电源线、控制线、信号线、输入线、输出线都应各自分开，最好采用线槽走线；信号线与电源线应尽量不要平行敷设；所有导线要分类编号，排列整齐；可编程控制器的所有接线端子最好采用标准接插件统一连接到端子上，以便于检修；不同的接线端子，其接线还应遵循各自的接线特点。

（三）日常维护

日常维护工作主要包含以下内容。

1. 日常清洁与巡查

经常用干抹布和皮老虎为可编程控制器的表面及导线间除尘除污,以保持工作环境的整洁和卫生;经常巡视、检查工作环境、工作状况、自诊断指示信号、编程器的监控信息及控制系统的运行情况,并做好记录,发现问题及时处理。

2. 定期检查与维修

在日常检查、记录的基础上,每隔半年(可根据实际情况适当提前或推迟)应对控制系统做一次全面停机检查,项目应包括工作环境、安装条件、电源电压、使用寿命和控制性能等方面。重点检查温度、湿度、振动、粉尘、干扰是否符合标准工作环境;接线是否安全、可靠;螺丝、连线以及接插头是否有松动;电气、机械部件是否有锈蚀和损坏等;检查电压大小、电压波动是否在允许范围内;检查导线及元件是否老化、锂电池寿命是否到期、继电器输出型触点开合次数是否已经超过规定次数(如 35 VA 以下为 300 万次)、金属部件是否锈蚀等。

(四) 故障诊断

可编程控制系统的常见故障,一方面可能来自于外部设备,如各种开关、传感器、执行机构和负载等;另一方面也可能来自于系统内部,如 CPU、存储器、系统总线、电源等。大量的统计分析与实践经验已经证明:可编程控制器本身一般是很少发生故障的,控制系统故障主要发生在各种开关、传感器、执行机构等外部设备。因此,当系统发生故障时首先检查外部设备。

在检查时根据可编程控制器使用手册上给出的诊断方法、诊断流程图和错误代码表,根据它们可很容易检查出 PLC 的故障。另外,利用 FX 系列 PLC 基本单元上 LED 指示灯诊断故障的方法。PLC 电源接通,电源指示灯(POWER)LED 亮,说明电源正常;若电源指示灯不亮,说明电源不通,应按电源检查流程图检查。当系统处于运行或监控状态,若基本单元上的 RUN 灯不亮,说明基本单元出了故障。锂电池(BATTERY)灯亮,应更换锂电池。若一路输入触点接通,相应的 LED 灯不亮;或者某一路未输入信号但是这一路对应的 LED 灯亮,可以判断是输入模块出了问题。输出 LED 灯亮,对应的硬输出继电器触点不动作,说明输出模块出了故障。基本单元上 CPU ERROR 灯 LED 闪亮,说明 PLC 用户程序的内容因外界原因发生改变所致。可能的原因有:锂电池电压下降;外部干扰的影响和 PLC 内部故障;写入程序时的语法错误也会使它闪亮。基本单元上 CPU ERROR 灯 LED 常亮,表示 PLC 的 CPU 误动作后,监控定时器使 CPU 恢复正常工作。这种故障可能由于外部干扰和 PLC 内部故障引起,应查明原因,对症采取措施。

19.GPP 软件简介

一、基本概况

SW3D5-GPPW-E 是三菱电气公司开发的用于可编程控制器的编程软件,可在 Windows3.1 及 Windows 95 下运行,适用于 IBM PC/AT (兼容)其 CPU 为 i486SX 或更高,内存需 8 兆或更高(推荐 16 兆以上)。该程序可在串行系统中可与可编程控制器进行通讯,文件传送,操作监控以及各种测试功能。

在 GPP 软件中,你可通过线路符号,助记符来创建顺控指令程序,建立注释数据及设置寄存器数据,并可将其存储为文件,用打印机打印。在 PLC 与 PC 之间必须有接口单元及缆线。

接口单元:

FX-232AWC 型 RS-232C/RS-422 转换器(便携式)、FX-232AW 型 RS-232C/RS-422 转换器(内置式)缆线:FX-422CAB 型 RS-422 缆线 [用于 FX1, FX2, FX2C 型可编程控制器, 0.3 米]; FX-422CAB-150 型 RS-422 缆线 [用于 FX1, FX2, FX2C 型可编程控制器, 1.5 米]。

二、用 GPP 编写梯形图

GPP 软件使用起来灵活、简单、方便,我们把它安装在程序中,使用时只要进入程序,选中 MELSEC Applications → 在 WINDOWS 下运行的 GPP,打开工程,选中新建,出现如下图 19-1 画面,先在 PLC 系列中选出你所使用的程控器的 CPU 系列,如在我们的实验中,选用的是 FX 系列,所以选 FXCPU, PLC 类型是指选机器的型号,我们实验用 FX2N 系列,所以选中 FX2N (C),确定后出现如图 19-2 画面,在画面上我们清楚地看到,最左边是根母线,兰色框表示现在可写入区域,上方有菜单,你只要任意点击其中的组件,就可得到你所要的线圈、触点等。



图 19-1



图 19-2


如你要在某处输入 X000，只要把蓝色光标移动到你所需要写的地方，然后在菜单上选中  触点，出现如图 19-3 画面



图 19-3

再输入 X000，即可完成写入 X000。

如要输入一个定时器,先选中线圈，再输入一些数据，数据的输入标准在第三章中已提过，图 19-4 显示了其操作过程。



图 19-4

对于计数器，因为它有时要用到两个输入端，所以在操作上既要输入线圈部分，又要输入复位部分，其操作过程如图 19-5、19-6 所示。



图 19-5

注意，在图 19-5 中的箭头所示部分，它选中的是应用指令，而不是线圈。



图 19-6

计数器的使用方法及计数范围在第三章中已讲过，同学们可自己查阅。图 19-7 是一个简单的计数器显示形式。

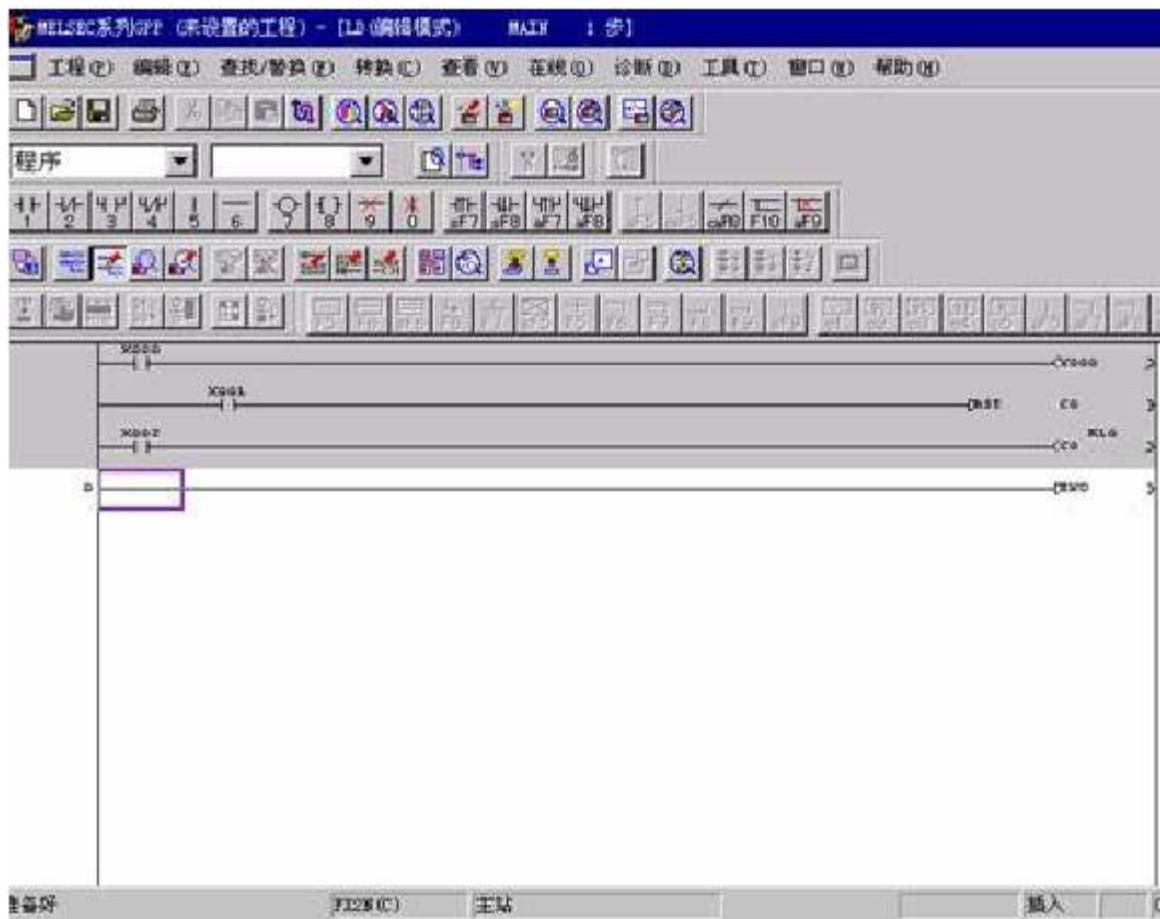


图 19-7

通过上面的举例，同学们就明白了，如果你需要画梯形图中的其它一些线、输出触点、定时器、定时器、辅助继电器等，在菜单上都能方便地找到，再输入组件编号即可。在图 5-6 的上方还有其它的一些功能菜单，如果你把光标指向菜单上的某处，在屏幕的左下角就会显示其功能，或者打开菜单上的“帮助”，你可找到一些快捷键列表、特殊继电器/寄存器等信息，同学们可自己边学习边练习。

三、传输、调试

当你写完梯形图，最后写上 END 语句后，必须进行程序转换，转换功能键有两种，在下图 19-8 的箭头所示位置。



图 19-8

在程序的转换过程中，如果程序有错，它会显示，也可通过菜单“工具”，查询程序的正确性。只有当梯形图转换完毕后，才能进行程序的传送，传送前，必须将 FX2N 面板上的开关拨向 STOP 状态，再打开“在线”菜单，进行传送设置，如下图 19-9 所示：



图 19-9

根据图标，你必须确定你的 PLC 与计算机的连接是通过 COM1 口还是 COM2 口连接，在实验中我们已统一将 RS-232 线连在了计算机的 COM1 口，你在操作上只要进行设置选择。写完梯形图后，在菜单上还是选择“在线”，选中“写入 PLC (W)”，就出现如图 19-9



从图上可看出，在执行读取及写入前必须先选中 MAIN、PLC 参数，否则，不能执行对程序的读取、写入，然后点击“开始执行”即可。