

第一章 可编程序控制器的结构及基本工作原理

第一节 PLC 的产生和特点及其应用方向

一、PLC 的发展史

可编程序控制器 (Programmable Logic Controller), 简称 PLC, 是在继电顺序控制基础上发展起来的以微处理器为核心的通用的工业自动化控制装置。

20 世纪 60 年代末期, 美国汽车制造工业竞争激烈, 为了适应生产工艺不断更新的需要, 在 1968 年美国通用汽车公司 (GM) 首先公开招标, 对控制系统提出的具体要求基本为: a. 它的继电控制系统设计周期短, 更改容易, 接线简单成本低。b. 它能把计算机的功能和继电器控制系统结合起来。但编程要比计算机简单易学、操作方便。c. 系统通用性强。1969 年美国数字设备公司 (DEC) 根据上述要求, 研制出世界上第一台 PLC, 并在 GM 公司汽车生产线上首次试用成功, 实现了生产的自动化。其后日本、德国等相继引入, 可编程序控制器迅速发展起来, 但是主要应用于顺序控制, 只能进行逻辑运算, 故称为可编程逻辑控制器, 简称 PLC。其定义: 可编程控制器是一种数字运算操作的电子系统, 专为在工业环境应用而设计的。它采用一类可编程的存储器, 用于其内部存储程序, 执行逻辑运算、顺序控制, 定时, 计数与算术操作等面向用户的指令, 并通过数字或模拟式输入/输出控制各种类型的机械或生产过程。可编程控制器及其有关外部设备, 都按易于与工业控制系统联成一个整体, 易于扩充其功能的原则设计。

随着电子技术和计算机技术的迅猛发展, PLC 的功能也越来越强大, 更多地具有计算机的功能, 所以又简称 PC (PROGRAMMABLE CONTROLLER), 但是为了不和 PERSONAL COMPUTER 混淆, 仍习惯称为 PLC。目前 PLC 已经在智能化、网络化方面取得了很好的发展, 并且现今已出现 SOFTPLC, 更是 PLC 领域无限的发展前景。

二、可编程序控制器的功能特点

1. 逻辑控制 PLC 具有逻辑运算功能, 能够进行与、或、非等逻辑运算, 可以代替继电器进行开关量控制, 故它可替代继电器进行开关量控制。
2. 定时控制 为满足生产控制工艺对时间的要求, PLC 一般提供时间继电器, 如 FX1S

提供 T0—T63 共 64 个计时器。并且计时时间常数在范围内用户编写程序时自己设定：接通延时、关断延时和定时脉冲等方式。并且在 PLC 运行中也可以读出、修改，使用方便。

3. 计数控制 为满足计数的需要，不同的 PLC 提供不同数量、不同类型的计数器。如 FX1S 提供 16 位增量计数 C0—C15（一般用）、C16—C31（保持用），32 位高速可逆计数器 C235—C245（单相单输入）、C246—C250（单相双输入）、C251—C255（双相双输入）共 26 个定时器。用脉冲控制可以实现加、减计数模式，可以连接码盘进行位置检测，且在 PLC 运行中也可以读出、修改，使用方便

4. 步进顺序控制 步进顺序控制是 plc 最基本的控制方式。是为有时间或运行顺序的生产过程专门设置的指令，在前道工序完成之后，就转入下一道工序，使一台 PLC 可作为多部步进控制器使用。

5. 对控制系统的监控 PLC 具有较强的监控能力，操作人员可以根据 PLC 的监控信息，通过监控命令，可以监视系统的运行状态，从而改变对异常值的设定。

6. 数据处理 PLC 具有较强的数据处理能力，随着 PLC 的发展，已经能对大量的数据进行快速处理。如数据采集、存储与处理功能。

7. 通信和联网 现代 PLC 大多数都采用了通信、网络技术，有 RS232 或 RS485 接口，可进行远程 I/O 控制，多台 PLC 可彼此间联网、通信，外部器件与一台或多台可编程控制器的信号处理单元之间，实现程序和数据交换，如程序转移、数据文档转移、监视和诊断。

通信接口或通信处理器按标准的硬件接口或专有的通信协议完成程序和数据转移。

在系统构成时，可由一台计算机与多台 PLC 构成“集中管理、分散控制”的分布式控制网络，以便完成较大规模的复杂控制。通常所说的 SCADA 系统，现场端和远程端也可以采用 PLC 作现场机。

8. 输入/输出接口调理功能 具有 A/D、D/A 转换功能，通过 I/O 模块完成对模拟量的控制和调节。位数和精度可以根据用户要求选择。具有温度测量接口，直接连接各种电阻或电偶。

9. 人机界面功能 提供操作者以监视机器、过程工作必需的信息。允许操作者和 PLC 系统与其应用程序相互作用，以便作出决策和调整。实现人机界面功能的手段：从

基层的操作者屏幕文字显示，到单机的 CRT 显示与键盘操作和用通信处理器、专用处理器、个人计算机、工业计算机的分散和集中操作与监视系统。

可编程序控制器是属于存储程序控制的一种装置，其控制功能是通过存放在存储器内的程序来实现的，若要对控制功能作修改，在很大程度上只须改变软件指令即可，使得硬件软件化。因此它在工业控制中的地位越来越高，占有极其重要的地位，最重要的原因是它具有如下独特的特点：

1. 可靠性高 PLC 是专门为工业控制设计的，在设计和制造过程中采取了多层次抗干扰、精选元件的措施，可在恶劣的工业环境下与强电设备一起工作，运行的稳定性和可靠性较高。PLC 是以集成电路为基本单元的电子设备，内部处理不依赖于接点，元件的寿命长，平均无故障工作时间高。

2. 编程简单易学 PLC 的最大特点之一，就是采用易学易懂的梯形图语言，它是以计算机软件技术构成人们惯用的继电器模型，形成一套独具风格的以继电器梯形图为基础的形象编程语言。方便电气人员在了解 PLC 工作原理和它的编程技术后，就可迅速地结合实际需要进行应用设计，进而将 PLC 用于实际控制系统中。

3. 通用性强，使用方便 由于 PLC 自身硬件特点，用户在进行控制系统的设计时，不需要自己设计和制作硬件装置，只需要根据控制要求进行模块的配置；用户所作的工作只是设计满足控制对象的控制要求的应用程序。对于一个控制系统，当控制要求改变时，只需修改程序，就能变更控制功能；与外围设备的连接方便，通讯协议标准。

4. 系统设计周期短 由于系统硬件的设计任务仅仅是根据对象的控制要求配置适当的模块，而不要设计具体的接口电路，同时软件设计和外围电路设计可以同时进行，这样大大缩短了整个系统设计的时间，加快了系统的设计周期。

5. 对生产工艺改变适应性强 其控制功能是通过软件编程来实现的，当生产工艺改变时，在很大程度上只需改变用户程序，这对现代化的小批量、多品种产品的生产尤其适合；现今 plc 已经朝着嵌入式系统发展，将进入日常生活中。

6. 安装简单、调试方便、维护工作量小 PLC 控制系统的安装接线工作量比继电器控制系统少得多，只需将现场的各种设备与 PLC 相应的 I/O 口相连。PLC 软件设计和调试大部分可以在实验室模拟进行，模拟调试好后再将 PLC 控制系统进行现场联机调试，方便省时。其本身可靠性高，有完善的自诊断能力和系统监控能力，方便迅速故障查明

和排除，维护的工作效率高。

7. 适应工业环境：PLC 的技术条件能在一般高温、振动、冲击和粉尘等恶劣环境下工作，能在强电磁干扰环境下可靠工作。这是 PLC 产品的市场生存价值。

三、PLC 的应用和发展前趋势

PLC 的应用是基于其以微处理器为核心，综合了计算机技术、自动控制技术和通信技术发展起来的一种通用的工业自动控制装置，它具有可靠性高、体积小、功能强、程序设计简单、灵活通用、维护方便等一系列优点，因而在制造、冶金、能源、交通、化工、电力等领域有着广泛的应用，成为现代工业控制的支柱之一。根据这些特点，可将其应用形式归纳为以下几种：开关量逻辑控制、模拟量控制、过程控制、定时和计数控制、顺序控制；、数据处理、通信和联网。

现代 PLC 的发展有两个主要趋势：其一是向体积更小、速度更快、功能更强和价格更低的微小型方面发展，即现今开始发展的嵌入式 PLC 控制方式；其二是向大型网络化、高可靠性、好的兼容性和多功能方面发展。

1、大型网络化 主要是朝 DCS 方向发展，使其具有 DCS 系统的一些功能。网络化和通信能力强是 PLC 发展的一个重要方面，向下可将多个 PLC、I/O 框架相连；向上与工业计算机、以太网、MAP 网等相连构成整个工厂的自动化控制系统。

2、多功能 随着自调整、步进电机控制、位置控制、伺服控制等模块的出现，使 PLC 控制领域更加宽广。

如研制出了多回路闭环控制模块、步进电机控制模块、仿真模块和通信处理模块等。并为用户提供了方便的人机界面，用户程序多级口令保护，极强的计算性能，完善的指令集，通过工业现场总线 PROFIBUS 以及以太网联网的网络能力，强劲的内部集成功能，全面的故障诊断功能；模块式结构可用于各处性能的扩展，脉冲输出晶闸管步进电机和直流电机；快速的指令处理大大缩短了循环周期，并采用了高速计数器，高速中断处理可以分别响应过程事件，大幅度降低了成本。

3、高可靠性 由于控制系统的可靠性日益受到人们的重视，一些公司已将自诊断技术、冗余技术、容错技术广泛应用到现有产品中，推出了高可靠性的冗余系统，并采用热备用或并行工作、多数表决的工作方式。PLC 即使在恶劣、不

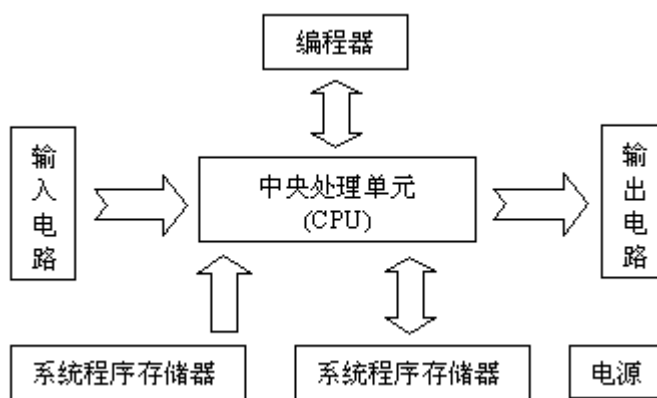
稳定的工作环境下，坚固、全密封的模板依然可正常工作，在操作运行过程中模板还可热插拔。

第二节 可编程控制器的结构和基本工作原理

PLC 由于其自身的特点，在工业生产的各个领域得到了愈来愈广泛的应用。而作为 PLC 的用户，要正确地应用 PLC 去完成各种不同的控制任务，首先应了解其组成结构和工作原理。

一、 PLC 的基本结构

可编程序控制器实施控制，其实质就是按一定算法进行输入输出变换，并将这个变换与以物理实现。输入输出变换、物理实现可以说是 PLC 实施控制的两个基本点，同时物理实现也是 PLC 与普通微机相区别之处，其需要考虑实际控制的需要，应能排除干扰信号适应于工业现场，输出应放大到工业控制的水平，能为实际控制系统方便使用，所以 PLC 采用了典型的计算机结构，主要是由微处理器（CPU）、存储器（RAM/ROM）、输入输出接口（I/O）电路、通信接口及电源组成。PLC 的基本结构如下图所示：



1、中央处理单元(CPU) 中央处理单元 (CPU) 是 PLC 的控制核心。它按照 PLC 系统程序赋予的功能：a. 接收并存储从用户程序和数据；b. 检查电源、存储器、I/O 以及警戒定时器的状态，并能诊断用户程序中的语法错误。当 PLC 投入运行时，首先它以扫描的方式采集现场各输入装置的状态和数据，并分别存入 I/O 映象寄存区，然后从用户程序存储器中逐条读取用户程序，经过命令解释后按指令的规定执行逻辑或算术运算并将结果送入 I/O 映象寄存区或数据寄存器内。等所有的用户程序执行完毕之后，最后将 I/O 映象寄存区的各输出状态或输出寄存器

内的数据传送到相应的输出装置，如此循环直到停止运行。为了提高 PLC 的可靠性，近年来对大型 PLC 还采用双 CPU 构成冗余系统，或采用三 CPU 的表决式系统。这样，即使某个 CPU 出现故障，整个系统仍能正常运行。

2、存储器 可编程序控制器的存储器分为系统程序存储器和用户程序存储器。存放系统软件（包括监控程序、模块化应用功能子程序、命令解释程序、故障诊断程序及其各种管理程序）的存储器称为系统程序存储器；存放用户程序（用户程序和数据）的存储器称为用户程序存储器，所以又分为用户存储器和数据存储器两部分。

PLC 常用的存储器类型：

（1）RAM（Random Access Memory）这是一种读/写存储器（随机存储器），其存取速度最快，由锂电池支持。

（2）EPROM（Erasable Programmable Read Only Memory）这是一种可擦除的只读存储器。在断电情况下，存储器内的所有内容保持不变。（在紫外线连续照射下可擦除存储器内容）。

（3）EEPROM（Electrical Erasable Programmable Read Only Memory）这是一种电可擦除的只读存储器。使用编程器就能很容易地对其所存储的内容进行修改。

PLC 存储空间的分配：虽然各种 PLC 的 CPU 的最大寻址空间各不相同，但是根据 PLC 的工作原理，其存储空间一般包括以下三个区域：

（1）系统程序存储区

（2）系统 RAM 存储区（包括 I/O 映象寄存区和系统软设备等）。

（3）用户程序存储区

系统程序存储区：在系统程序存储区中存放着相当于计算机操作系统的系统程序。包括监控程序、管理程序、命令解释程序、功能子程序、系统诊断子程序等。由制造厂商将其固化在 EPROM 中，用户不能直接存取。它和硬件一起决定了该 PLC 的性能。

系统 RAM 存储区：系统 RAM 存储区包括 I/O 映象寄存区以及各类软元件，如：逻辑线圈、数据寄存器、计时器、计数器、变址寄存器、累加器等存储器。

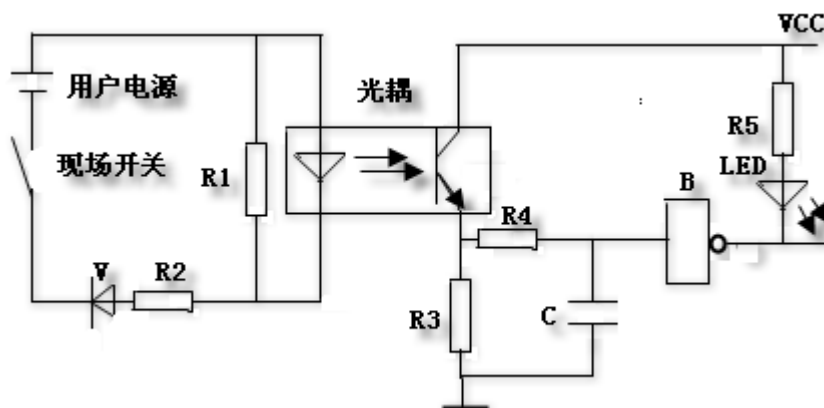
(1) I/O 映象寄存区：由于 PLC 投入运行后，只是在输入采样阶段才依次读入各输入状态和数据，在输出刷新阶段才将输出的状态和数据送至相应的外设。因此，它需要一定数量的存储单元 (RAM) 以存放 I/O 的状态和数据，这些单元称作 I/O 映象寄存区。一个开关量 I/O 占用存储单元中的一个位，一个模拟量 I/O 占用存储单元中的一个字。因此整个 I/O 映象寄存区可看作两个部分组成：开关量 I/O 映象寄存区；模拟量 I/O 映象寄存区。

(2) 系统软元件存储区：除了 I/O 映象寄存区以外，系统 RAM 存储区还包括 PLC 内部各类软元件（逻辑线圈、计时器、计数器、数据寄存器和累加器等）的存储区。该存储区又分为具有失电保持的存储区域和失电不保持的存储区域，前者在 PLC 断电时，由内部的锂电池供电，数据不会丢失；后者当 PLC 断电时，数据被清零。

(3) 用户程序存储区：用户程序存储区存放用户编制的用户程序。不同类型的 PLC，其存储容量各不相同。

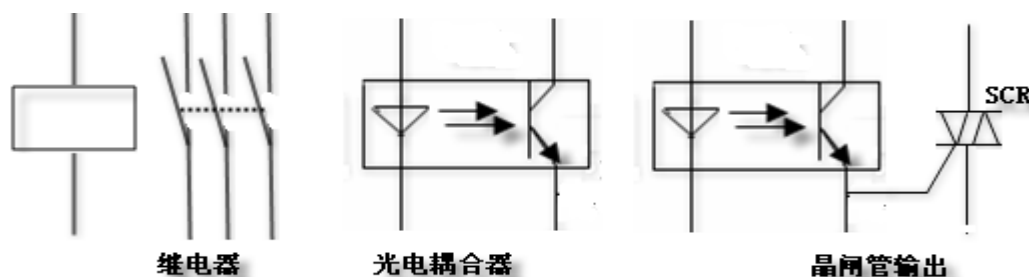
3. 输入接口电路 输入输出信号有开关量、模拟量、数字量三种，在我们实习室涉及到的信号当中，开关量最普遍，也是实验条件所限，在次我们主要介绍开关量接口电路。

可编程序控制器优点之一是抗干扰能力强。这也是其 I/O 设计的优点之处，经过了电气隔离后，信号才送入 CPU 执行的, 防止现场的强电干扰进入。如下图就是采用光电耦合器(一般采用反光二极管和光电三极管组成)的开关量输入接口电路：



4. 输出接口电路：可编程序控制器的输出有：继电器输出 (M)、晶体管输出 (T)、晶闸管输出 (SSR) 三种输出形式。

(1) 输出接口电路的隔离方式



(2) 输出接口电路的主要技术参数

a. 响应时间 响应时间是指 PLC 从 ON 状态转变成 OFF 状态或从 OFF 状态转变成 ON 状态所需要的时间。继电器输出型响应时间平均约为 10ms；晶闸管输出型响应时间为 1ms 以下；晶体管输出型在 0.2ms 以下为最快。

b. 输出电流 继电器输出型具有较大的输出电流，AC250V 以下的电路电压可驱动纯电阻负载 2A/1 点、感性负载 80VA 以下（AC100V 或 AC200V）及灯负载 100W 以下（AC100V 或 200V）的负载；Y0、Y1 以外每输出 1 点的输出电流是 0.5A，但是由于温度上升的原因，每输出 4 合计为 0.8A 的电流，输出晶体管的 ON 电压约为 1.5V，因此驱动半导体元件时，请注意元件的输入电压特性。Y0、Y1 每输出 1 点的输出电流是 0.3A，但是对 Y0、Y1 使用定位指令时需要高速响应，因此使用 10—100mA 的输出电流；晶闸管输出电流也比较小，FX1S 无晶闸管输出型。

c. 开路漏电流 开路漏电流是指输出处于 OFF 状态时，输出回路中的电流。继电器输出型输出接点 OFF 是无漏电流；晶体管输出型漏电流在 0.1mA 以下；晶闸管较大漏电流，主要由内部 RC 电路引起，需在设计系统时注意。

(3) 输出公共端（COM） 公共端与输出各组之间形成回路，从而驱动负载。

FX1S 有 1 点或 4 点一个公共端输出型，因此各公共端单元可以驱动不同电源电压系统的负载。

5. 电源

PLC 的电源在整个系统中起着十分重要得作用。如果没有一个良好的、可靠得电源系统是无法正常工作的，因此 PLC 的制造商对电源的设计和制造也十分重视。一般交流电压波动在+10%(+15%)范围内，可以不采取其它措施而将 PLC 直接连接到

交流电网上去。如 FX1S 额定电压 AC100V—240V，而电压允许范围在 AC85V—264V 之间。允许瞬时停电在 10ms 以下，能继续工作。

一般小型 PLC 的电源输出分为两部分：一部分供 PLC 内部电路工作；一部分向外提供给现场传感器等的工作电源。因此 PLC 对电源的基本要求：

- 1) 能有效地控制、消除电网电源带来的各种干扰；
- 2) 电源发生故障不会导致其它部分产生故障；
- 3) 允许较宽的电压范围；
- 4) 电源本身的功耗低，发热量小；
- 5) 内部电源与外部电源完全隔离；
- 6) 有较强的自保护功能。

二、 PLC 的工作原理

由于 PLC 以微处理器为核心，故具有微机的许多特点，但它的工作方式却与微机有很大不同。微机一般采用等待命令的工作方式，如常见的键盘扫描方式或 I/O 扫描方式，若有键按下或有 I/O 变化，则转入相应的子程序，若无则继续扫描等待。

PLC 则是采用循环扫描的工作方式。对每个程序，CPU 从第一条指令开始执行，按指令步序号做周期性的程序循环扫描，如果无跳转指令，则从第一条指令开始逐条执行用户程序，直至遇到结束符后又返回第一条指令，如此周而复始不断循环，每一个循环称为一个扫描周期。扫描周期的长短主要取决于以下几个因素：一是 CPU 执行指令的速度；二是执行每条指令占用的时间；三是程序中指令条数的多少。一个扫描周期主要可分为 3 个阶段。

1. 输入刷新阶段

在输入刷新阶段，CPU 扫描全部输入端口，读取其状态并写入输入状态寄存器。完成输入端刷新工作后，将关闭输入端口，转入程序执行阶段。在程序执行期间即使输入端状态发生变化，输入状态寄存器的内容也不会改变，而这些变化必须等到下一工作周期的输入刷新阶段才能被读入。

2. 程序执行阶段

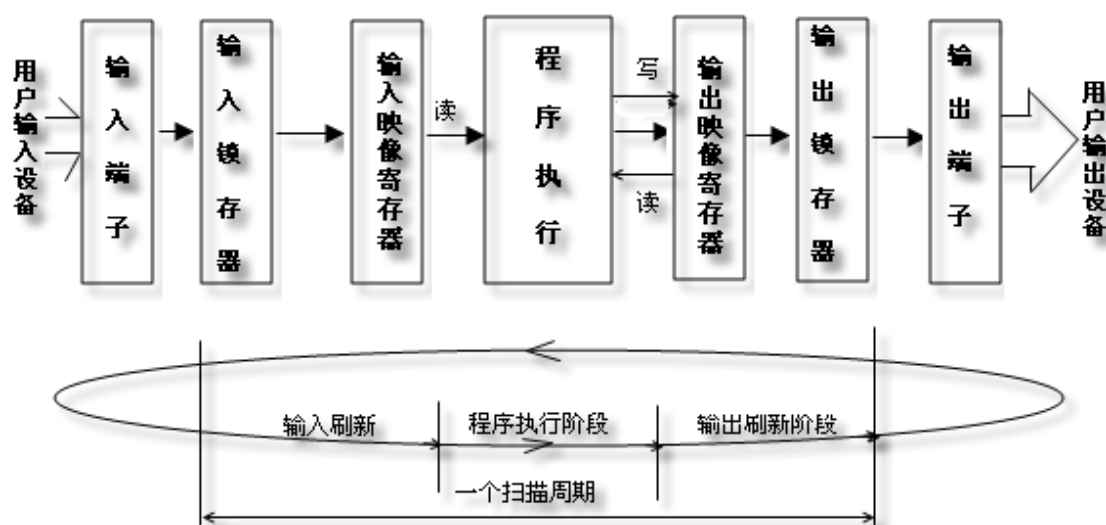
在程序执行阶段，根据用户输入的控制程序，从第一条开始逐步执行，并将相

应的逻辑运算结果存入对应的内部辅助寄存器和输出状态寄存器。当最后一条控制程序执行完毕后，即转入输入刷新阶段。

3. 输出刷新阶段

当所有指令执行完毕后，将输出状态寄存器中的内容，依次送到输出锁存电路（输出映像寄存器），并通过一定输出方式输出，驱动外部相应执行元件工作，这才形成 PLC 的实际输出。

由此可见，输入刷新、程序执行和输出刷新三个阶段构成 PLC 一个工作周期，由此循环往复，因此称为循环扫描工作方式。由于输入刷新阶段是紧接输出刷新阶段后马上进行的，所以亦将这两个阶段统称为 I/O 刷新阶段。实际上，除了执行程序 and I/O 刷新外，PLC 还要进行各种错误检测（自诊断功能）并与编程工具通讯，这些操作统称为“监视服务”，一般在程序执行之后进行。综上所述，PLC 的扫描工作过程如图 1—4 所示。



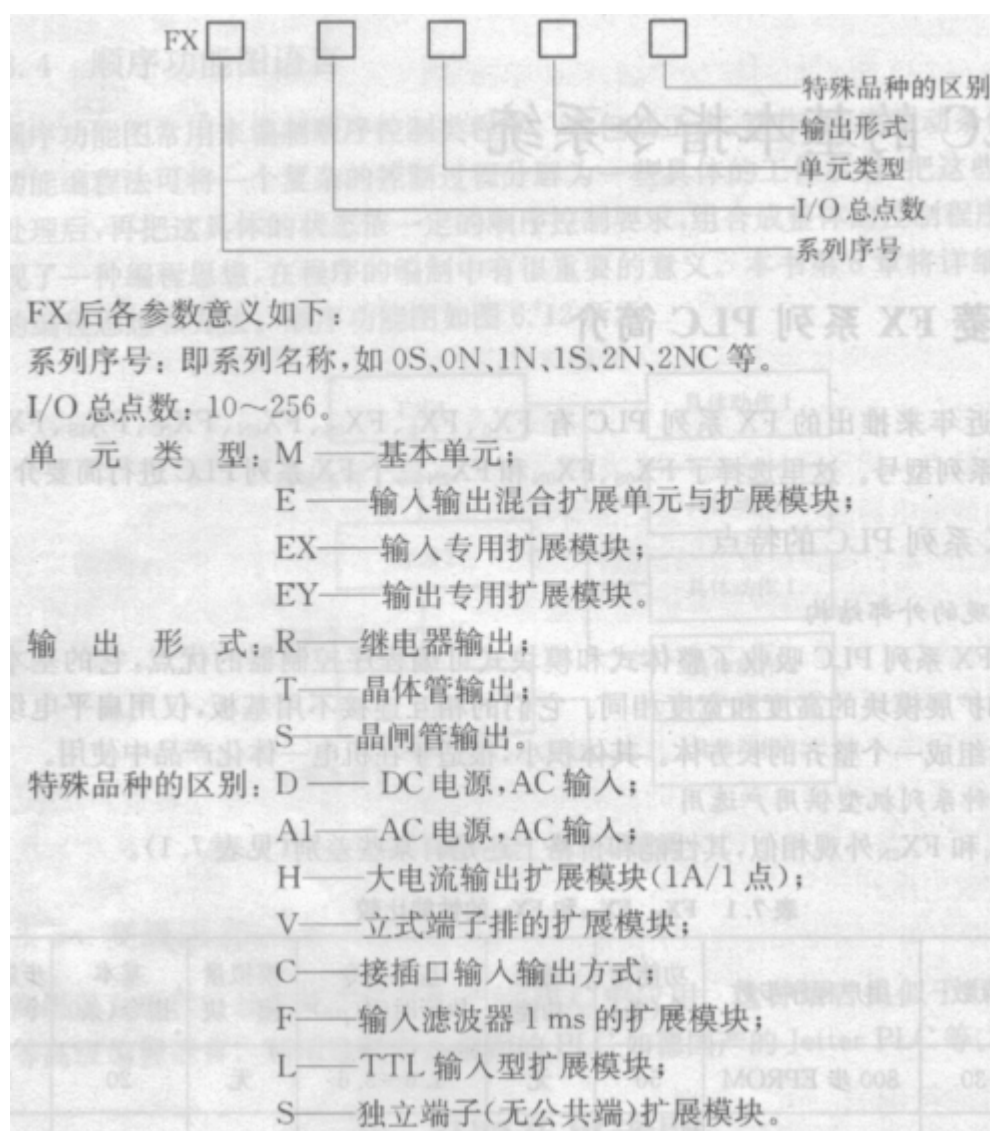
显然扫描周期的长短主要取决于程序的长短。扫描周期越长，响应速度越慢。由于每个扫描周期只进行一次 I/O 刷新，即每一个扫描周期 PLC 只对输入、输出状态寄存器更新一次，所以系统存在输入输出滞后现象，这在一定程度上降低了系统的响应速度。但是由于其对 I/O 的变化每个周期只输出刷新一次，并且只对有变化的进行刷新，这对一般的开关量控制系统来说是完全允许的，不但不会造成影响，还会提高抗干扰能力。这是因为输入采样阶段仅在输入刷新阶段进行，PLC 在一个工作周期的大部分时间是与外设隔离的，而工业现场的干扰常常是脉冲、短时间的，误动作将大大减小。但是在快速响应系统中就会造成响应滞后现象，这个一般 PLC 都

会采取高速模块。

总之，PLC 采用扫描的工作方式，是区别于其他设备的最大特点之一，我们在学习和使用 PLC 当中都应加强注意。

第三节 FX1S 的性能指标

Fx 系列 PLC 个部分含义：



若特殊品种缺省，通常指 AC 电源、DC 输入、横式端子排，其中继电器输出：2A/1 点；晶体管输出：0.5A/1 点；晶闸管输出：0.3A/1 点。

例如 FX2N---40MRD，其参数含义为三菱 FX2N PLC，有 40 个 I/O 点的基本单元，继电器输出型，使用 DC24V 电源。

FX1s 性能规格:

项目		规格	备注
运转控制方法		通过储存的程序周期运转	
I/O 控制方法		批次处理方法（当执行 END 指令时）	I/O 指令可以刷新
运转处理方法		基本指令：0.55 至 0.7 μ s 应用指令：3.7 至几百 μ s	
编程语言		逻辑梯形图和指令清单	使用步进梯形图能生成 SFC 类型程序
程式容量		内置 2K 步 EEPROM	存储盒（FX1n-EEPROM-8L）可选
指令数目		基本顺序指令：27 步进梯形指令：2 应用指令：85	最大可用 167 条应用指令，包括所有的变化
I/O 配置		最大总 I/O 由主处理单元设置	
辅助继电器 (M 线圈)	一般	384 点	M0 到 M383
	锁定	128 点（子系统）	M384 至 M511
	特殊	256 点	M8000 至 8255
状态继电器 (S 线圈)	一般	128 点	S0 至 S127
	初始	10 点（子系统）	S0 至 S9
定时器 (T)	100 毫秒	范围：0 至 3276.7 秒 63 点	T0 至 T55
	10 毫秒	范围：0 至 3276.7 秒 31 点	当特殊 M 线圈工作时 T32 到 T62
	1 毫秒	范围：0.001 至 32.767 秒 1 点	T163
计数器 (C)	一般	范围：1 至 32767 数 16 点	C0 至 C15 类型：16 位增计数器
	锁定	范围：1 至 32767 数 16 点	C16 至 C31 类型：16 位增计数器
高速计数器 (C)	单相	范围： -2147483648+2147483648 数	C235 至 C238 4 点（注意 C235 被锁定）
	单相 c/w 起始停止输入	Fxo:选择多达 4 个单相计数器，组合计数频率不大于 5KHz.或选择一个比相或 A/B 相计数器,组合计数频率不大于 2KHz.	C241（锁定上）C242 和 C244（锁定）3 点
	双相		C241、C247 和 C249（都锁定）3 点

	A/B 相	FXos:当使用多个单相计数器时,频率和必须不大于 14KHz.只允许单.双相高速计数器同时使用。当使用双相计数器时,最大遍数速度必须不大于 14KHz, 计算为(遍数边数为 5 时, 2ph 计数器速度)+1ph 计数器速度。	C251、C252 和 C254 (都锁定) 3 点
数据寄存器 (D)	一般	128 点	D0 至 D127 类型: 32 位元件的 16 位数据存储寄存器
	锁定	128 点	D128 至 255 类型: 32 位元件的 16 位数据存储寄存器
	外部调节	范围: 0 至 255 2 点	通过外部设置电位计间接输入 D8013 或 D8030&D8031 数据
	特殊	256 点 (包含 D8030, D8031)	从 D8000 至 D8255 类型: 16 位数据存储寄存器
	变址	16 点	V 和 Z 类型: 16 位数据存储寄存器
指标 (P)	用于 CALL	64 点	N0 至 P63
	用于中断	6 点	100*至 130* (上升触发*=1, 下降触发*=0)
嵌套层次		用于 MC 和 MRC 时 8 点	N0 至 N7
常数	十进位 K	16 位: -32768 至 32768 32 位: -2147483648 至 +2147483647	
	十六进位 H	16 位: 0000 至 FFFF 32 位: 00000000 至 FFFFFFFF	

习题:

1. 可编程序控制器的定义是什么?
2. 可编程序控制器有哪些主要特点?
3. 可编程序控制器的主要功能有哪些?
4. 可编程序控制器由哪几部分组成? 各有什么作用?
5. PLC 的工作方式是什么? 说明工作原理。

第二章 FX1S 的软元件及其编程软件

第一节 FX1S 的软元件地址号、错误代码介绍

一、FX1s 可编程控制器一般软元件的种类和编号如下所示，因为和其他 FX 系列可编程控制器的内容不同，请注意区别：

	FX1s-10M	FX1s-14M	FX1s-20M	FX1s-30M
输入继电器 X	X000~X005 6 点	X000~X007 8 点	X000~X013 12 点	X000~X017 16 点
输出继电器 Y	Y000~Y003 4 点	Y000~Y005 6 点	Y000~Y007 8 点	Y000~Y015 14 点

辅助继电器 M	M0~M383 384 点 一般用		【M384~M511】 128 点保持用	M8000~M8255 256 点 ※1 特殊用
状态 S	S0~S127 128 点保持用		【S0~S127】 128 点 保持用	
	初始化用 S0~S9 原点回归用 S10~S127			
定时器 T	T0~T31 32 点 100ms	T32~T62 31 点 10ms M8028 置 ON	【T63】 1 点 1ms 累计	内置电位器 2 点 VR1: D8030 VR2: D8031

	16 位增量记数		32 位高速可逆计数器 最大 6 点		
计数器 C	C0~C15 16 点 一般用	[C16~C31] 16 点 保持用	[C235~C245] 单相单输入	[C246~C250] 单相双输入	[C251~C255] 双相输入

数据寄存器 D, V, Z		D0~D127 128 点 一般用	[D128~D255] 保持用	[D1000~D2499] 1500 点 文件专用 文件用 参数设定, 可设定为文件寄存器	D8000~D82455 256 点 ※1 特殊用	V0~V7 Z0~Z7 16 点 变址用
嵌套指针		N0~N7 8 点 主控用		P0~P63 64 点 跳转指令、子程序用跳转 地址指针		100※~105※ 6 点 输入中断用指针
常数	K	16 位 -32768~32767			32 位 -2147483648~2147483647	

	H	16 位 0~FFFFH	32 位 0~FFFFFFFH
--	---	--------------	-----------------

【】内的软元件是停电保持区域（keep Area），保持区域的范围是不能变更的。

注 记：※ 1. 对 应 功 能 请 参 照 特 殊 软 元 件 编 号 一 览 表。

为了能可靠保持，可编程控制器连续通电时间必须在 5 分钟以上。

二、特殊软元件，FX1s 可编程控制器特殊软元件的种类及其功能如下：如[M][D]这样有[]括起的软元件和未使用的软元件，或没有记载的未定义的软元件，请不要对它们进行程序驱动或数据写入。

*1: RUN——STOP 时清除；*2: STOP——RUN 时清除；*3: 停电保持；*4: END 指令结束处理；*5: 22 (FX1s) 100 (版本号 1.00)；*6: 0002=2K 步；*7: 02H=存储盒 (PROTECT OFF) 0AH=存储盒 (PROTECT ON) 10H=可编程序控制器内置 EEPROM 存储器；*8: M8062 除外；*9: 用公历的后二位表示，也可以切换成公历四位表示，当用四位表示时可表示从 1980—2079 年为止；*10: 适用于 RS、ASCII、HEX、CCD 指令。

PC 状态：

编号	名称	备注	编号	名称	备注
[M8000]	RUN 监控	RUN 时常闭	D8000	监视定时器	初期值 200ms
[M8001]	RUN 监控	RUN 时常开	[D]8001	PC 类型和版本	*5
[M8002]	初始化脉冲	RUN 后输出一个扫描周期的 ON	[D]8002	存储器容量	*6
[M]8003	初始化脉冲	RUN 后输出一个扫描周期的 OFF	[D]8003	存储器种类	*7
[M]8004	出错发生	M8060~M8067 检知*8	[D]8004	出错特殊 M 的编号	M8060~M8067
[M]8005			[D]8005		
[M]8006			[D]8006		
[M]8007			[D]8007		
[M]8008			[D]8008		
[M]8009			[D]8009		

时钟

编号	名称	备注	编号	名称	备注
[M]8010		以 10ms 为周期振荡	[D]8010	扫描时间当前值 (单位 0.1ms)	含恒定扫描等待时间
[M]8011	10ms 时钟	以 100ms 为周期振荡	[D]8011	最小扫描时间 (单位 0.1ms)	
[M]8012	100ms 时钟	以 1s 为周期振荡	[D]8012	最大扫描时间 (单位 0.01ms)	
[M]8013	1s 时钟	以 1min 为周期振荡	D8013	0~59 秒预置值或当前值	时钟误差 ± 45 秒 / 月 (25℃) 有闰年修正.
[M]8014	1min 时钟		D8014	0~59 分预置值或当前值	
[M]8015	计时停止和预置		D8015	0~23 小时预置值或当前值	
[M]8016	停止显示时间		D8016	0~31 日	
[M]8017	± 30 秒修正		D8017	0~12 月预置值或当前值	
[M]8018	RTC 检出	常闭	D8018	公历年二位预置值或当前值	

				前值表示的	
[M]8019	RTC 出错		D8018	星期 0(一)-6(六预置值或当前值)	

D8013~D8019 是停电保持。D8018(年)也可以切换成公历 1980~2079 的 4 位表示。

第二节 三菱 PLC 编程软件简介

PLC 的程序输入通过手持编程器、专用编程器或计算机完成。手持编程器体积小，携带方便，在现场调试时优越性强，但在程序输入、阅读、分析时较繁琐；而专用编程器价格太贵，通用性差；计算机编程在教学中优势较大，且其通讯更为方便。因此也就有了相应的计算机平台上的编程软件和专用通讯模块，在这节当中我们重点介绍三菱 fx 系列编程软件的使用和操作。

三菱公司 fx 系列 plc 编程软件名称为 fxgpwin, 我们介绍版本为 SW0PC-FXGP/WIN-C Version3.00 Copyright (C) 1996 MITSUBISHI ELECTRIC CORPORATION, 其具体应用说明如下:

1. Fxgpwin 编程软件对 FX0/ FX0S、FX1S、FX1N、FX0N、FX1 FX2N / FX2NC 和 FX (FX2/FX2C) 系列三菱 plc 编程及其它操作。下图为软件的文件组成:



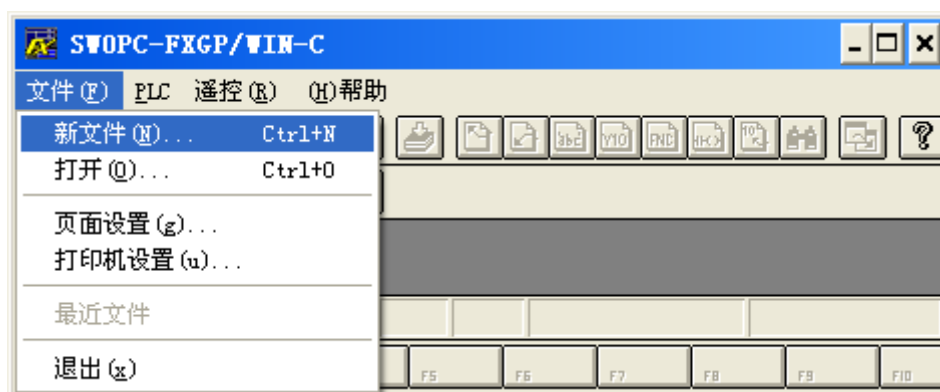
- 1) 进入 fxgpwin 的编程环境

双击桌面 fxgpwin 图标或按 table 键选择到图标 fxgpwin, 即可进入编程环境。

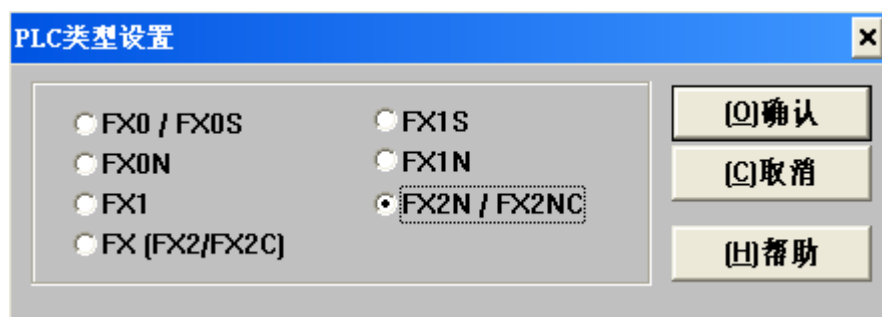
- 2) 编程环境如下图



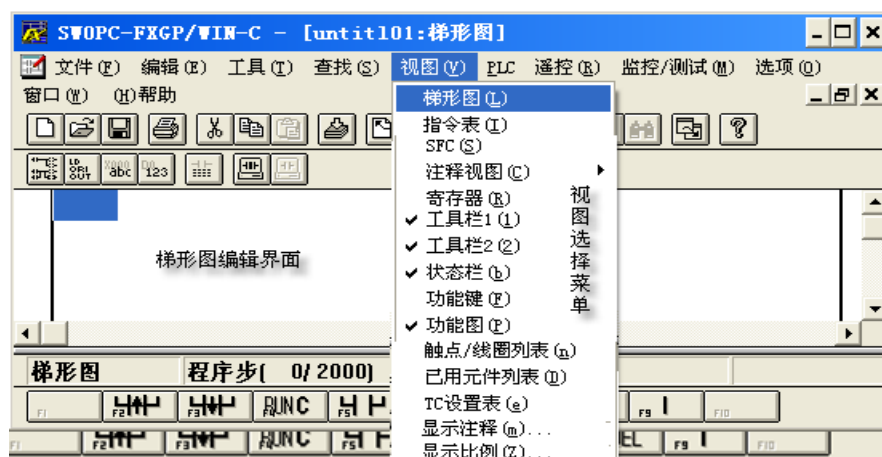
3) 编写新程序，新建文件



出现 PLC 选型界面



选择好 PLC 型号后按确认键即可进入编辑界面，在视图可以切换梯形图、指令表等

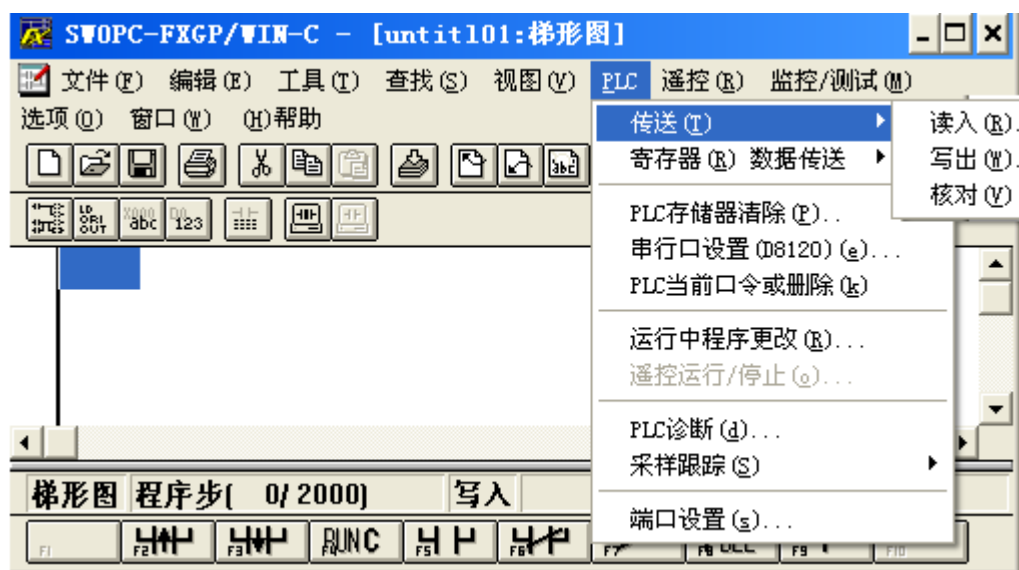


建立好文件后就可以在其中编写程序了。

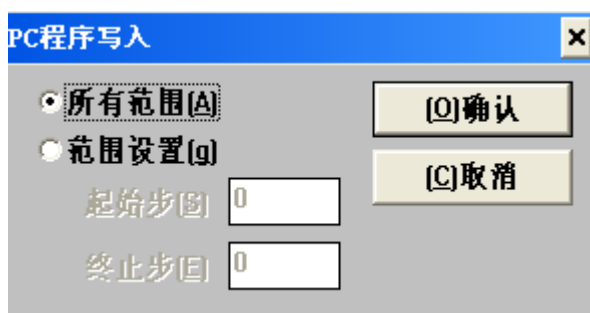
4) 程序的保存在“文件”菜单下的“另存为”下即可。

5) PLC 程序上载，传入 PLC。

当编辑好程序后可以就可以向 PLC 上载程序，方法是：首先必须正确连接好编程电缆，其次是 PLC 通上电源（POWER）指示灯亮，打开菜单“PLC”——“传送”——“读出”确认。

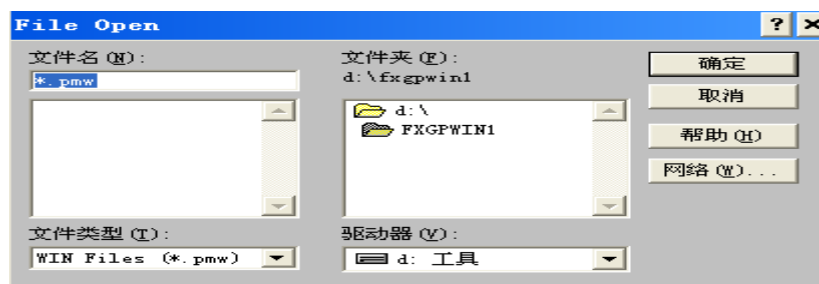


出现程序写入步数范围选择框图，确认后即可：



6) PLC 程序下载一样，在上述操作中选择“读入”，其他操作不变。

7) 程序打开 打开菜单“文件”——“打开”，出现界面，选择要打开的程序，确定即可。



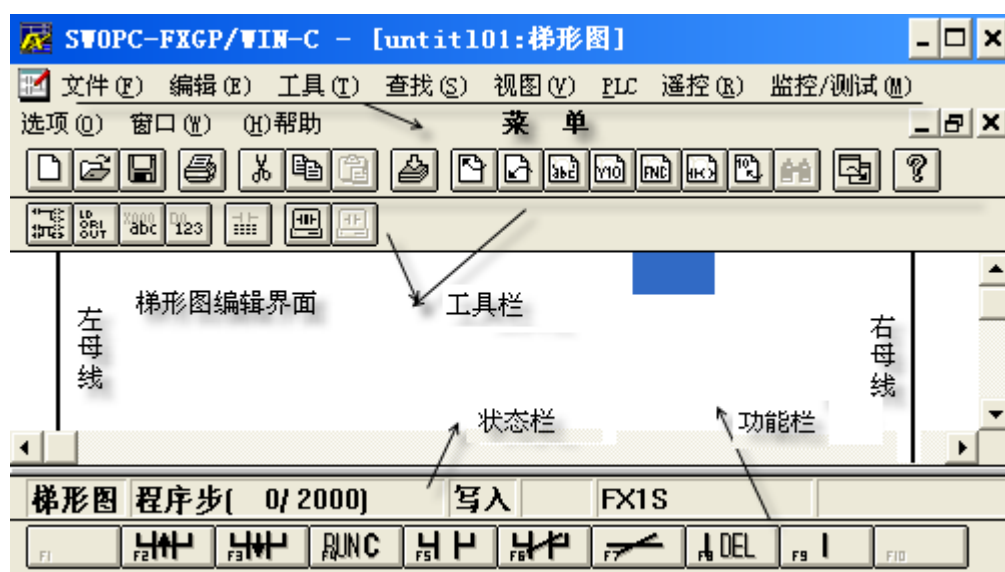
8) 退出主程序 ALT+F4 或点击文件菜单下的“退出”。

2. 程序的编写

1) 编程语言的选择

FXGPWIN 软件提供三种编程语言，分别为梯形图、指令表、SFC 状态流程图。打开“视图”菜单，选择对应的编程语言。

2) 梯形图编辑时如图



3) 编写程序可通过功能栏来选择，也可以直接写指令进行程序编写。主要是熟悉菜单下各功能子菜单。

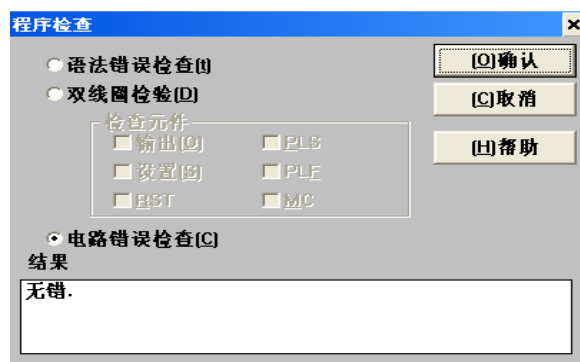
4) 梯形图编写需进行转换，在工具菜单下选择或按 F4 键，转换完毕即可进行上载



调试，注意端口设置。

5) 程序的检查

在“选项”菜单下的“程序检查”，即进入程序检查环境，可检查语法错误、双线圈、

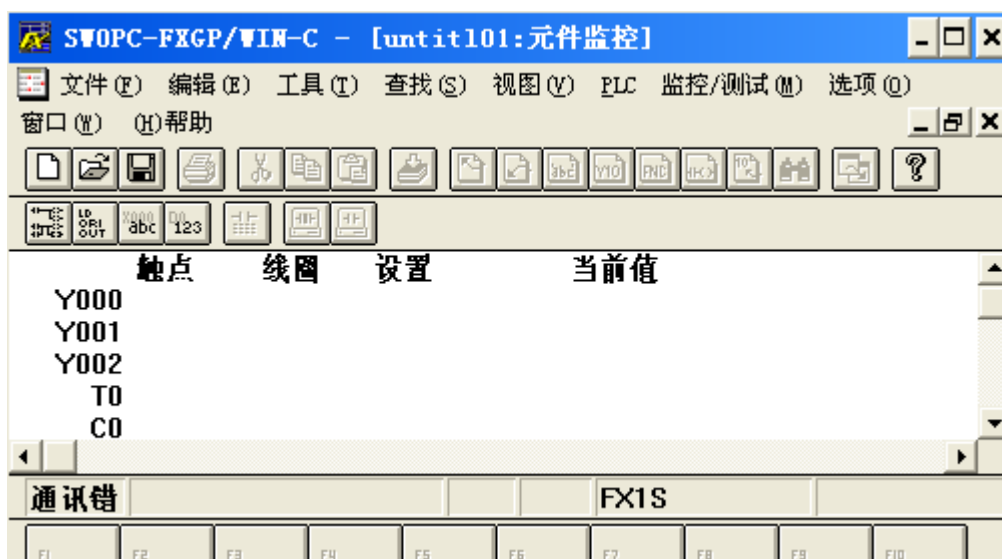


电路错误。

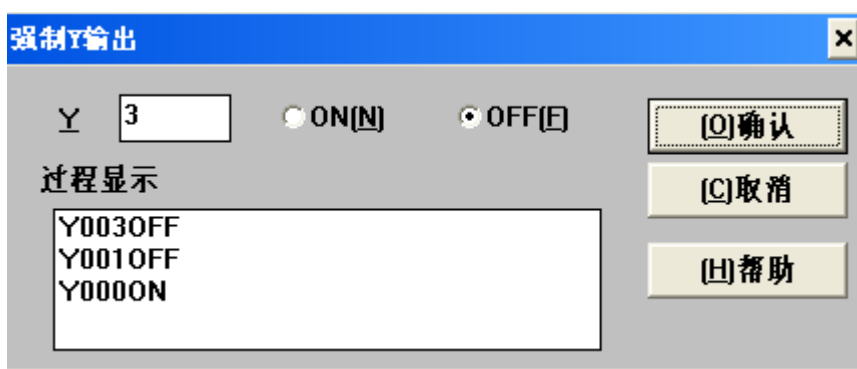
3. 软元件的监控和强制执行

在 FXGPWIN 操作环境下，可以监控各软元件的状态和强制执行输出等功能。

元件监控功能界面：



强制输出功能界面：

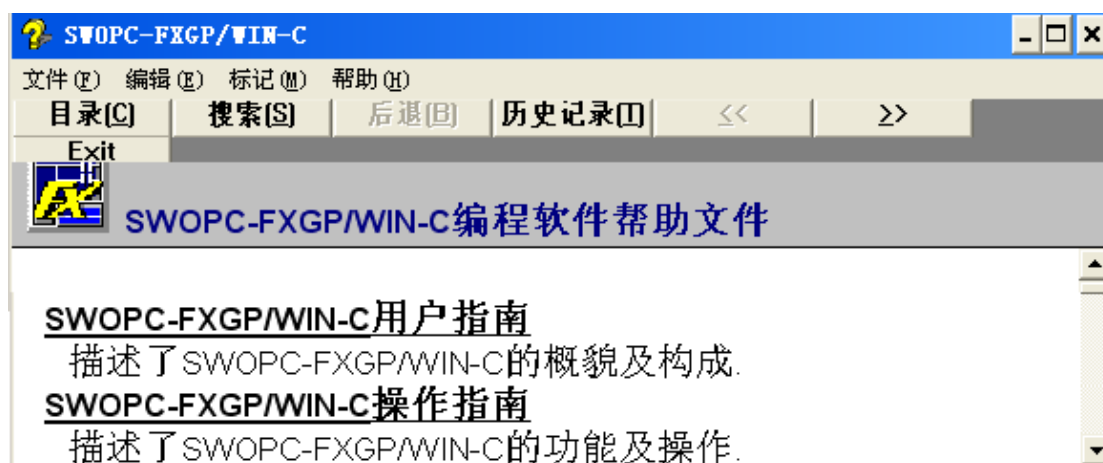


强制 ON/OFF 功能界面：



主要在“监控/测试”菜单中完成。

4. 其他各功能在操作过程中在帮助菜单中熟悉。



5. 梯形图常用项具体操作

- (1) 剪切 (梯形图编辑):

[编辑(Alt + E)] - [剪切(Alt + t)]

功能：将电路块单元剪切掉。

操作方法：通过[编辑] - [块选择]菜单操作选择电路块。 在通过[编辑] -

[剪切]菜单操作或[Ctrl] + [X]键操作,被选中的电路块被剪切掉。 被剪切的数据保存在剪切板中。

警告： 如果被剪切的数据超过了剪切板的容量,剪切操作被取消。

- (2) 粘贴 (梯形图编辑):

[编辑(Alt + E)] - [粘贴(Alt + P)]

功能：粘贴电路块单元。

操作方法：通过[编辑] - [粘贴] 菜单操作,或[Ctrl] + [V]键操作, 被选择

的电路块被粘贴上。 被粘贴上的电路块数据来自于执行剪切或拷贝命令时存储在剪切板上的数据。

通过[编辑] - [粘贴]菜单操作或[Ctrl] + [V]键操作,被选中的电路块被粘贴。 被粘贴的数据是在执行剪切或拷贝操作

时被保存在剪切板中的数据。

警告： 如果剪切板中的数据未被确认为电路块,剪切操作被禁止。

（3） 拷贝（梯形图编辑）：

[编辑(Alt + E)] - [拷贝(Alt + C)]

功能：拷贝电路块单元。

操作方法：通过[编辑] - [块选择]菜单操作选择电路块。 在通过[编辑] - [拷贝]菜单操作或[Ctrl] + [C]键操作,被选中的电路块数据被保存在剪切板中。

警告： 如果被拷贝的数据超过了剪切板的容量,拷贝操作被取消。

（4） 行删除（梯形图编辑）：

[编辑(Alt + E)] - [行删除(Alt + L)]

功能：在行单元中删除线路块。

操作方法：通过执行[编辑] - [行删除]菜单操作或[Ctrl]+[Delete]键盘操作,光标所在行的线路块被删除。

警告： 1.该功能在创建(更正)线路时禁用.需在完成线路变化后执行。

2.被删除的数据并未存储在剪切板中。

（5） 行删除（梯形图编辑）：

[编辑(Alt + E)] - [行删除(Alt + L)]

功能：在行单元中删除线路块。

操作方法： 通过执行[编辑] - [行删除]菜单操作或[Ctrl]+[Delete]键盘操作,光标所在行的线路块被删除。

警告 1. 该功能在创建(更正)线路时禁用.需在完成线路变化后执行。

2. 被删除的数据并未存储在剪切板中.

(6) 删除 (梯形图编辑):

[编辑(Alt + E)] - [删除(Alt + D)]

功能: 删除电路符号或电路块单元.

操作方法: 通过进行[编辑] - [删除]菜单操作或[Delete]键操作删除光标所在处的电路符号. 欲执行修改操作, 首先通过执行[编辑] - [块选择]菜单操作选择电路块. 在通过[编辑] - [删除]菜单操作或[Delete]键操作, 被选单元被删除.

警告 1. 被删除的数据并不在剪切板中.

(7) 行插入 (梯形图编辑):

[编辑(Alt + E)] - [行插入(Alt + I)]

功能: 插入一行.

操作方法: 通过执行[编辑] - [行插入]菜单操作, 在光标位置上插入一行.

(8) 触点

[工具(Alt + T)] - [触点(Alt + n)] - [-| |-...]

[工具(Alt + T)] - [触点(Alt + n)] - [-|/|-...]

[工具(Alt + T)] - [触点(Alt + n)] - [-|P|-...]

[工具(Alt + T)] - [触点(Alt + n)] - [-|F|-...]

功能: 输入电路符号中的触点符号.

操作方法: 在执行[工具] - [触点] - [-| |-] 菜单操作时, 选中一个触点符号, 显示元件输入对话框. 执行[工具] - [触点] - [-|/|-] 菜单操作选中B触点. 执行[工具] - [触点] - [-|P|-] 菜单操作选择脉冲触点符号, 或执行[工具] - [触点] - [-|F|-] 菜单操作选择下降沿触发触点符号. 在元件输入栏中输入元件, 按[Enter]键或确认按钮后, 光标所在处的便有一个元件被登录. 若点击参照按钮, 则显示元件说明对话框, 可完成更多的设置.

(9) 线圈

[工具(Alt + T)] - [线圈(Alt + o)]

功能: 在电路符号中输入输出线圈.

操作方法:在进行[工具] - [线圈] 菜单操作时,元件输入对话框被显示 t. 在输入栏中输入元件,按[Enter]键或确认按钮,于是光标所在地的输出线圈符号被登录. 点击参照按钮显示元件说明对话框,可进行进一步的特殊设置.

(10) 功能指令线圈:

[工具(Alt + T)] - [功能]

功能: 输入功能线圈命令等.

操作方法:在执行[工具] - [功能]菜单操作时,命令输入对话框显出. 在输入栏中输入元件,按[Enter]键或确认按钮,光标所在地的应用命令被登录. 再点击参照按钮,命令说明对话框被打开,可进行进一步的特殊设置.

(11) 连线

[工具(Alt + T)] - [连线(Alt + W)] - [|]

[工具(Alt + T)] - [连线(Alt + W)] - [-]

[工具(Alt + T)] - [连线(Alt + W)] - [- / -]

[工具(Alt + T)] - [连线(Alt + W)] - [| 删除]

功能: 输入垂直及水平线,删除垂直线.

操作方法:垂直线被菜单操作[工具] - [连线] - [|]登录,水平线被菜单操作[工具] - [连线] - [-]登录,翻转线菜单操作[被工具] - [连线] - [- / -]登录,垂直线被菜单操作[工具] - [连线] - [| 删除] 删除.

(12) 全部清除:

[工具(Alt + T)] - [全部清除(Alt + A)...]

功能: 清除程序区(NOP 命令).

操作方法: 点击[工具] - [全部清除] 菜单,显示清除对话框. 通过按[Enter]键或点击确认按钮,执行清除过程.

警告 1. 所清除的仅仅是程序区,而参数的设置值未被改变.

(13) 转换 (梯形图编辑):

[工具(Alt + T)] - [转换(Alt + C)]

功能：将创建的电路图转换格式存入计算机中。

操作方法：执行[工具] - [转换]菜单操作或按[转换]按钮(F4 键)。 在转换过程中,显示信息电路转换中。

警告 1. 如果在未完成转换的情况下关闭电路窗口,被创建的电路图被抹去。

（14） 梯形图监控：

[监控/测试(Alt + M)] - [开始监控(Alt + S)]

功能：在显示屏上监视可编程控制器的操作状态。 从电路编辑状态转换到监视状态,同时在显示的电路图中显示可编程控制器操作状态(ON/OFF)。

操作方法：激活梯形图视图,通过进行菜单操作进入[监控/测试]-[开始监控]。

警告 1. 在梯形图监控中,电路图中只有 ON/OFF 状态被监控。

2.当监控当前值以及设置寄存器,计时器,计数器数据时,应使用依据登录监控功能

（15） 程序传送：

[PLC] - [传送(Alt + T)]

功能：将已创建的顺控程序成批传送到可编程控制器中.传送功能包括[读入],[写出]及[校验]。

[读入]：将 PLC 中的顺控程序传送到计算机中。

[写出]：将计算机中的顺控程序发送到可编程控制器中。

[校验]：将在计算机及可编程控制器中顺控程序加以比较校验。

操作方法：由执行[PLC] - [传送] - [读入], - [写出], - [校验]菜单操作而完成。 当选择[读入]时,应在[PLC 模式设置]对话框中将已连接的 PLC 模式设置好。

警告 1.计算机的 RS232C 端口及 PLC 之间必须用指定的缆线及转换器连接。

2.执行完[读入]后,计算机中的顺控程序将被丢失,PLC 模式被改变成被设定的模式,现有的顺控程序被读入的程序替代。

3.在[写出]时, PLC 应停止运行,程序必须在 RAM 或 EE-PROM 内存保护关断的情况下写出。 然后机动进行校验。

（16） PLC 存储器清除：

[PLC] - [PLC 存储器清除(Alt +P)...]

功能：为了初始化 PLC 中的程序及数据。以下三项将被清除。

[PLC 存储器]：顺控程序为 NOP,参数设置为缺省值。

[数据元件存储器]：数据文件缓冲器中数据置零。

[位元件存储器]：X, Y, M, S, T, C 的值被置零。

操作方法：执行[PLC] - [PLC 存储器清除]菜单操作,再在[PLC 存储器清除]中设置清除项。

警告 1.计算机的 RS232C 端口及 PLC 之间必须用指定的缆线及转换器连接。

2.特殊数据寄存器数据不被清除。

习题：

1. 列表写出 FX1s—20MR 的软元件种类及编号。
2. 特殊辅助继电器描述。

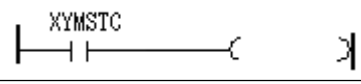
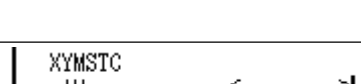
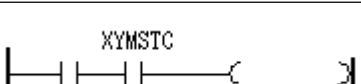
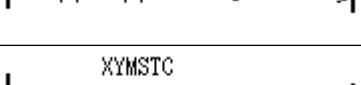
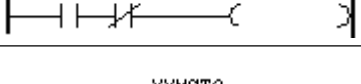
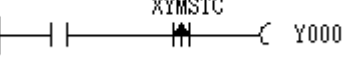
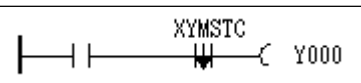
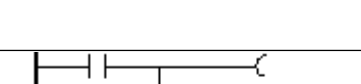
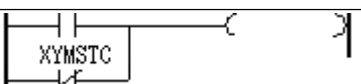
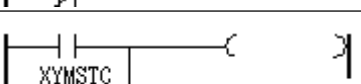
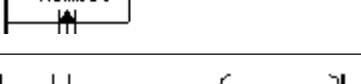
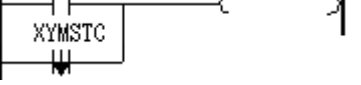
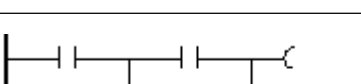
第三章 基本逻辑指令系统

可编程序控制器是按照用户的控制要求编写程序来进行控制的。程序的编写就是用一定的编程语言把一个控制任务描述出来。PLC 编程语言中，程序的表达方式有几种：梯形图、指令语句表、逻辑功能图和高级语言，但最常用的语言是梯形图语言和指令语句表。梯形图是一种图形语言，它沿用了传统的继电器控制系统的形式，读图方法和习惯也相同，所以梯形图比较形象和直观，便于熟悉继电器控制系统的技术人员接受。指令语句表一般由助记符和操作元件组成，助记符是每一条基本指令的符号，表示不同的功能；操作元件是基本指令的操作对象。本章内容主要是介绍 FX1S 的基本指令形式、功能和编程方法。

第一节 基本指令的类型

基本指令一览表：基本指令.步进梯形图指令 FX1S 可编程序控制器的基本顺控指令和步进梯形图指令的种类及其功能如下所示：

助记符	功 能	格式和操作软元件
-----	-----	----------

LD 取	常开触点逻辑运算起始(常开触点与左母线连接)	
LDI 取反	常闭触点逻辑运算起始(常闭触点与左母线连接)	
LDP 取脉冲 上升沿	上升沿检测(检测到信号的上升沿时闭合一个扫描周期)	
LDF 取脉冲 下降沿	下降沿检测(检测到信号的下降沿时闭合一个扫描周期)	
AND 与	串联连接(常开触点与其他触点或触点组串联连接)	
ANI 与非	串联连接(常闭触点与其他触点或触点组串联连接)	
ANDP 与脉冲 上升沿	上升沿串联连接(检测到位软元件上升沿信号时闭合一个扫描周期)	
ANDF 与脉冲 下降沿	下降沿串联连接(检测到位软元件下降沿信号时闭合一个扫描周期)	
OR 或	并联连接(常开触点与其他触点或触点组 并联连接)	
ORI 或非	并联连接(常闭触点与其他触点或触点组 并联连接)	
ORP 或脉冲 上升沿	脉冲上升沿检测并联连接(检测到位软元件上升沿信号时闭合一个扫描周期)	
ORF 或脉冲 下降沿	脉冲下降沿检测并联连接(检测到位软元件下降沿信号时闭合一个扫描周期)	
ANB 电路块与	并联电路块的串联连接(电路块与其他触点或触点组串联连接)	
ORB 电路块或	串联电路块的并联连接(电路块与其他触点或触点组并联连接)	
OUT 输出	线圈驱动	
SET 置 1	使线圈接通并保持动作	
RST 复零	使线圈断开,消除动作保持,寄存器清零	

PLS 上升沿 脉冲	上升沿微分输出(当检测到输入脉冲的上升沿时,指令的操作元件闭合一个扫描周期)	
PLF 下降沿 脉冲	下降沿微分输出(当检测到输入脉冲的下降沿时,指令的操作元件闭合一个扫描周期)	
MC 主控指令	公共串联接点的连接(将左母线临时移到一个所需位置,产生一临时左母线,形成主控电路块)	
MCR 主控复位	公共串联接点的消除(取消临时左母线,将左母线返回到原来的位置,结束主控电路块)	
MPS 进栈指令	进栈(将逻辑运算结果存入栈存储器,存储器中原来的存储结果依次向栈存储器下层推移)	
MRD 读栈指令	读栈(将存储器一号单元的内容读出,且栈存储器中的内容不发生变化)	
MPP 出栈指令	出栈 9 将存储器中一号单元的结果取出,存储器中其他单元的数据依次向上推移)	
INV 取反	运算结果取反	
NOP 空操作	无动作	
END 结束	输入输出处理以及返回到 0 步	
STL 步进接点	步进接点开始(将步进接点接到左母线)	
RET 步进结束	步进接点开始(使副母线返回到原来的左母线位置)	

第二节 基本指令介绍

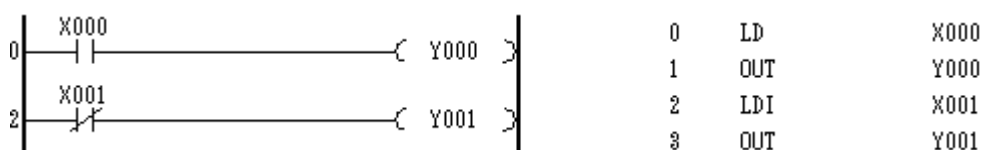
FX1S 的基本指令形式、功能和编程方法。基本指令是以位为单位的逻辑操作，是构成继电器控制电路的基础

一、LD、LDI、OUT 指令

符号名称	功能	操作元件
LD 取	常开触点逻辑运算起始	X、Y、M、S、T、C
LDI 取反	常闭触点逻辑运算起始	X、Y、M、S、T、C

OUT 输出	线圈驱动	Y、M、S、T、C
--------	------	-----------

1. 程序举例:



2. 例题解释: 1) 当 X0 接通时, Y0 接通;

2) 当 X1 断开时, Y1 接通。

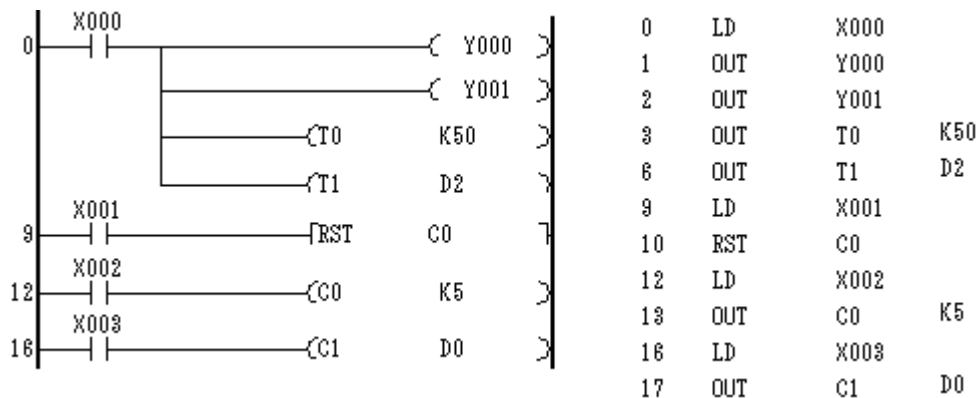
3. 指令使用说明:

1) LD 和 LDI 指令用于将常开和常闭触点接到左母线上;

2) LD 和 LDI 在电路块分支起点处也使用;

3) OUT 指令是对输出继电器、辅助继电器、状态继电器、定时器、计数器的线圈驱动指令, 不能用于驱动输入继电器, 因为输入继电器的状态是由输入信号决定的。

4) OUT 指令可作多次并联使用, 如下图。

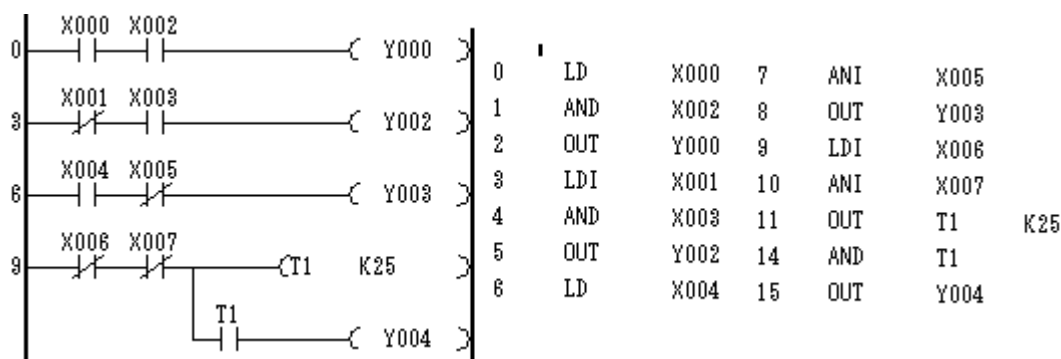


5) 定时器的计时线圈或计数器的计数线圈, 使用 OUT 指令后, 必须设定值 (常数 K 或指定数据寄存器的地址号), 如上图。

二、AND、ANI 指令

符号名称	功能	操作元件
AND 与	常开触点串联连接	X、Y、M、S、T、C
ANI 与非	常闭触点串联连接	X、Y、M、S、T、C

1. 程序举例:



2. 例题解释: 1) 当 X0 接通, X2 接通时 Y0 接通;

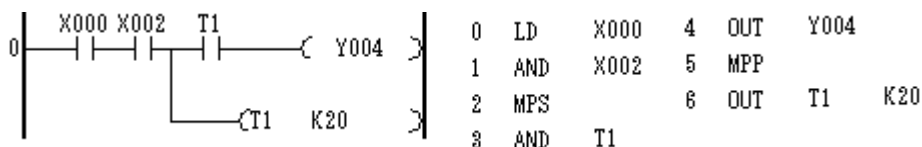
2) X1 断开, X3 接通时 Y2 接通;

3) 常开 X4 接通, X5 断开时 Y3 接通;

4) X6 断开, X7 断开, 同时达到 2.5 秒时间, T1 接通, Y4 接通。

3. 指令说明:

- 1) AND、ANI 指令可进行 1 个触点的串联连接。串联触点的数量不受限制, 可以连续使用;
- 2) OUT 指令之后, 通过触点对其他线圈使用 OUT 指令, 称之为纵接输出。这种纵接输出如果顺序不错, 可多次重复使用; 如果顺序颠倒, 就必须要用我们后面要学到的指令 (MPS/MRD/MPP) 如下图;



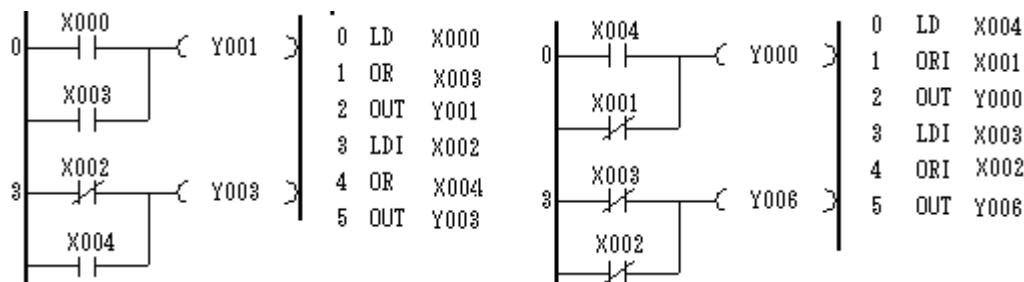
- 3) 当继电器的常开触点或常闭触点与其他继电器的触点组成的电路块串联时, 也使用 AND 指令或 ANI 指令。

电路块: 就是由几个触点按一定的方式连接的梯形图。由两个或两个以上的触点串联而成的电路块, 称为串联电路块; 由两个或两个以上的触点并联连接而成的电路块, 称为并联电路块; 触点的混联就称为混联电路块。

三、OR、ORI 指令

符号名称	功能	操作元件
OR 或	常开触点并联连接	X、Y、M、S、T、C
ORI 或非	常闭触点并联连接	X、Y、M、S、T、C

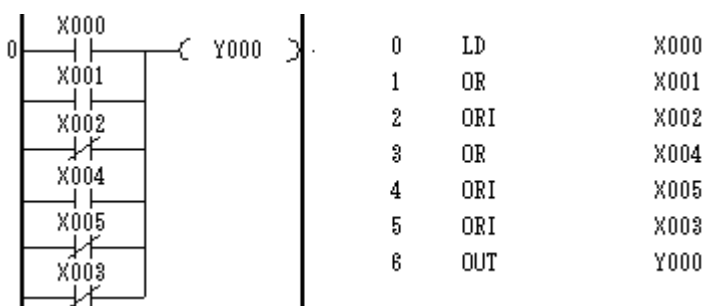
1. 程序举例:



2. 例题解释: 1) 当 X0 或 X3 接通时 Y1 接通;
- 2) 当 X2 断开或 X4 接通时 Y3 接通;
- 3) 当 X4 接通或 X1 断开时 Y0 接通;
- 4) 当 X3 或 X2 断开时 Y6 接通。

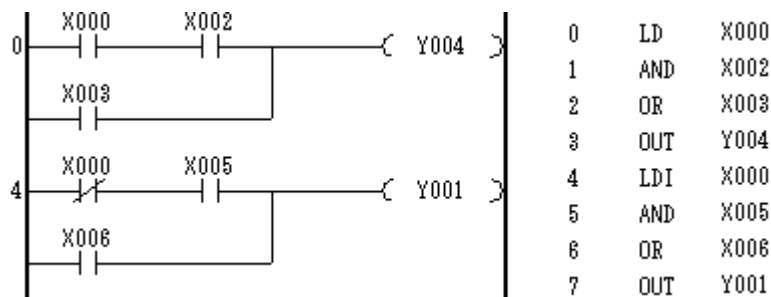
3. 指令说明:

- 1) OR、ORI 指令用作 1 个触点的并联连接指令。
- 2) OR、ORI 指令可以连续使用, 并且不受使用次数的限制;



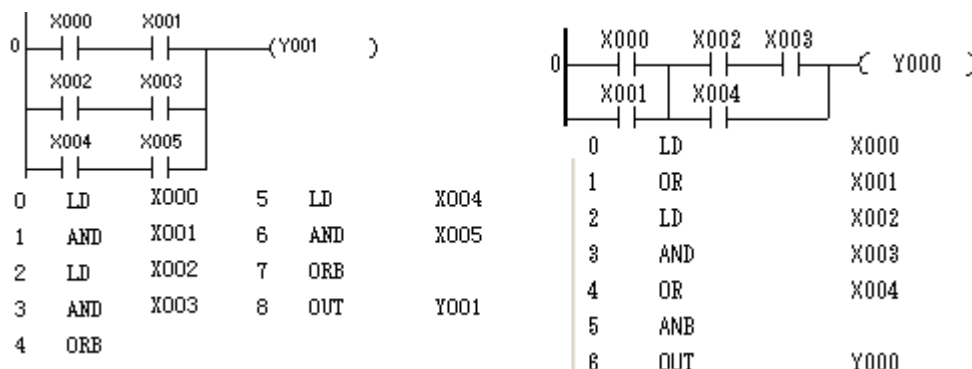
- 3) OR、ORI 指令是从该指令的步开始, 与前面的 LD、LDI 指令步进行并联连接。

- 4) 当继电器的常开触点或常闭触点与其他继电器的触点组成的混联电路块并联时，也可以用这两个指令。



四、串联电路块并联指令 ORB、并联电路块串联指令 ANB

1. 程序举例：

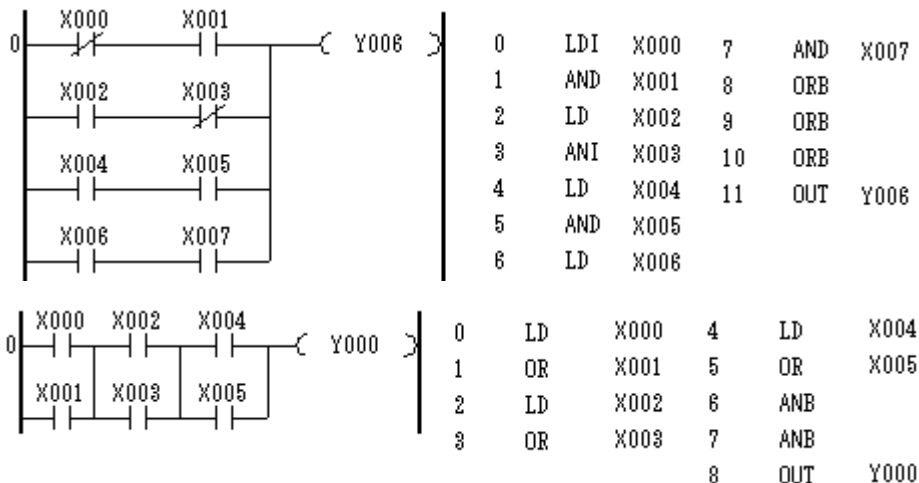


2. 例题解释：1) X0 与 X1、X2 与 X3、X4 与 X5 任一电路块接通，Y1 接通；

2) X0 或 X1 接通，X2 与 X3 接通或 X4 接通，Y0 都可以接通；

3. 指令说明：

- 1) ORB、ANB 无操作软元
- 2) 2 个以上的触点串联连接的电路称为串联电路块；
- 3) 将串联电路并联连接时，分支开始用 LD、LDI 指令，分支结束用 ORB 指令；
- 4) ORB、ANB 指令，是无操作元件的独立指令，它们只描述电路的串并联关系；
- 5) 有多个串联电路时，若对每个电路块使用 ORB 指令，则串联电路没有限制，如上举例程序；
- 6) 若多个并联电路块按顺序和前面的电路串联连接时，则 ANB 指令的使用次数没有限制；



- 7) 使用 ORB、ANB 指令编程时，也可以采取 ORB、ANB 指令连续使用的方法；但只能连续使用不超过 8 次，在此建议不使用此法。

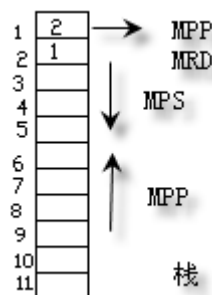
五、分支多重输出 MPS、MRD、MPP 指令

MPS 指令：将逻辑运算结果存入栈存储器；

MRD 指令：读出栈 1 号存储器结果

MPP 指令：取出栈存储器结果并清除；

用于多重输出电路；FX 的 PLC 有 11 个栈存储器，用来存放运算中间结果的存储区域称为堆栈存储器。使用一次 MPS 就将此刻的运算结果送入堆栈的第一段，而将原来的

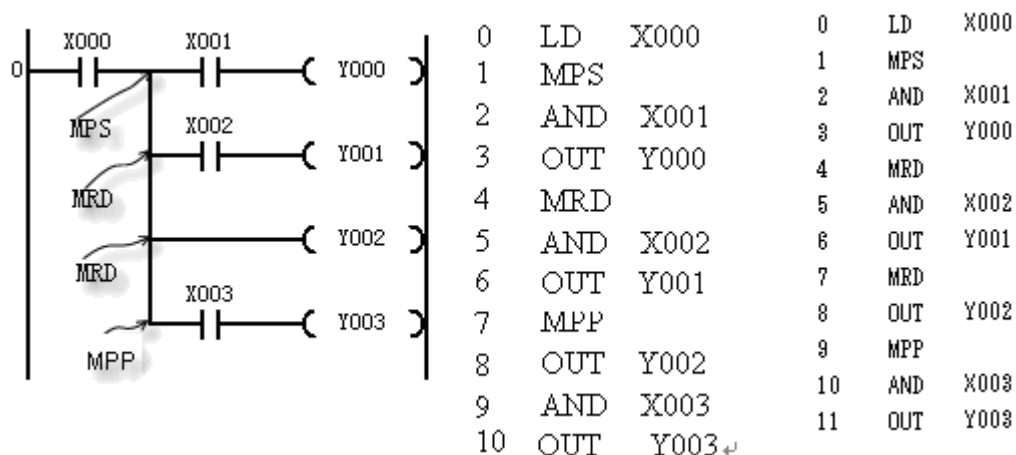


第一层存储的数据移到堆栈的下一段。

MRD 只用来读出堆栈最上段的最新数据，此时堆栈内的数据不移动。

使用 MPP 指令，各数据向上一段移动，最上段的数据被读出，同时这个数据就从堆栈中清除。

1. 程序举例：

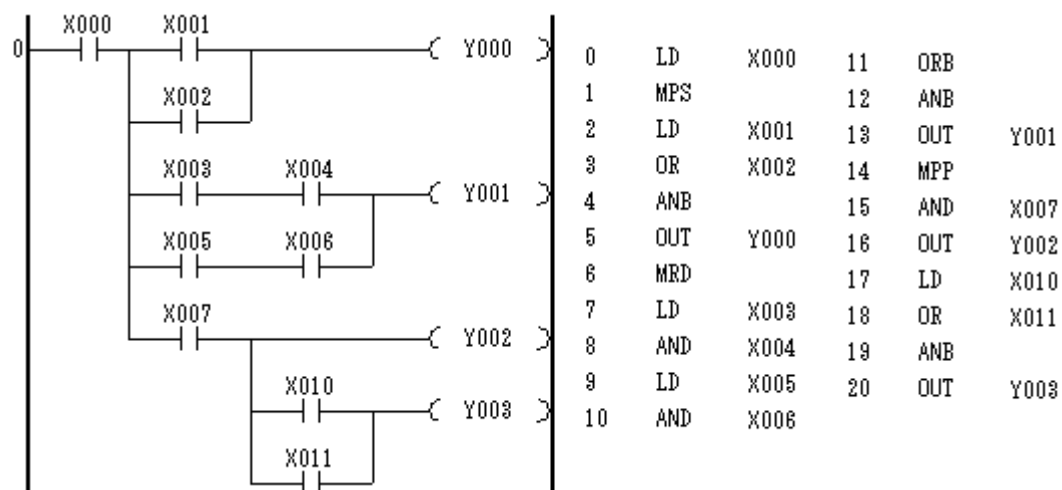


2. 例题解释：1) 当公共条件 X0 闭合时，X1 闭合则 Y0 接通；X2 接通则 Y1 接通；Y2 接通；X3 接通则 Y3 接通。

2) 上述程序举例中可以用两种不同的指令形式，这个地方应给学生明确解释。

3. 指令说明：

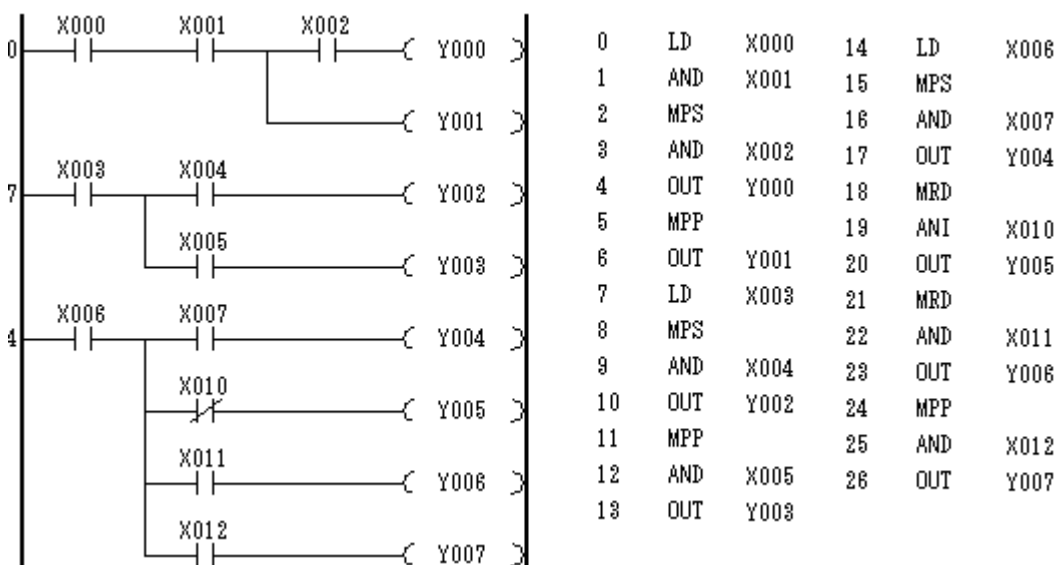
- 1) MPS、MRD、MPP 无操作软元件
- 2) MPS、MPP 指令可以重复使用，但是连续使用不能超过 11 次，且两者必须成对使用缺一不可，MRD 指令有时可以不用；
- 3) MRD 指令可多次使用，但在打印等方面有 24 行限制；
- 4) 最终输出电路以 MPP 代替 MRD 指令，读出存储并复位清零；
- 5) MPS、MRD、MPP 指令之后若有单个常开或常闭触点串联，则应该使用 AND 或 ANI 指令；
- 6) MPS、MRD、MPP 指令之后若有触点组成的电路块串联，则应该使用 ANB 指令；



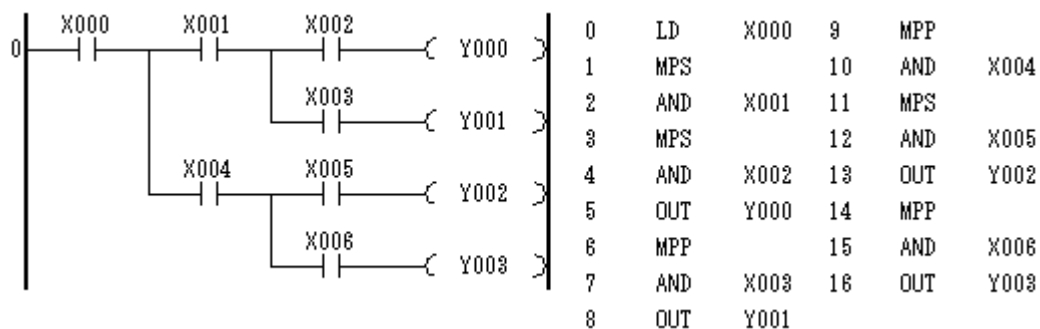
7) MPS、MRD、MPP 指令之后若无触点串联，直接驱动线圈，则应该使用 OUT 指令；

8) 指令使用可以有多层堆栈。

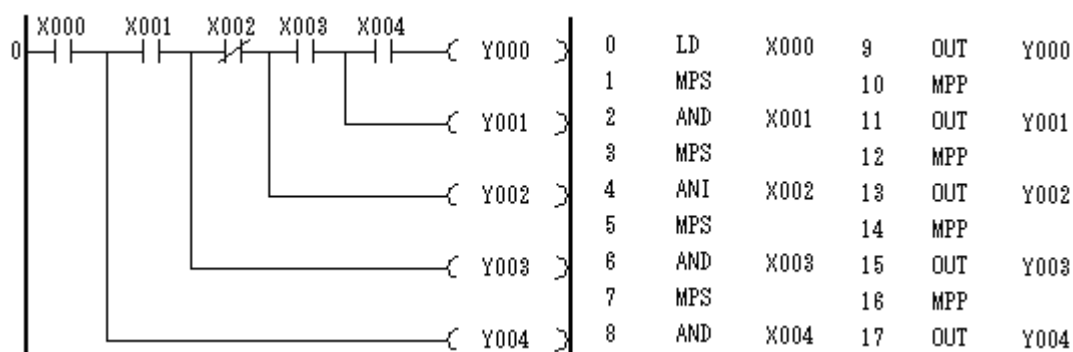
编程例一，一层堆栈：



编程例二，两层堆栈：



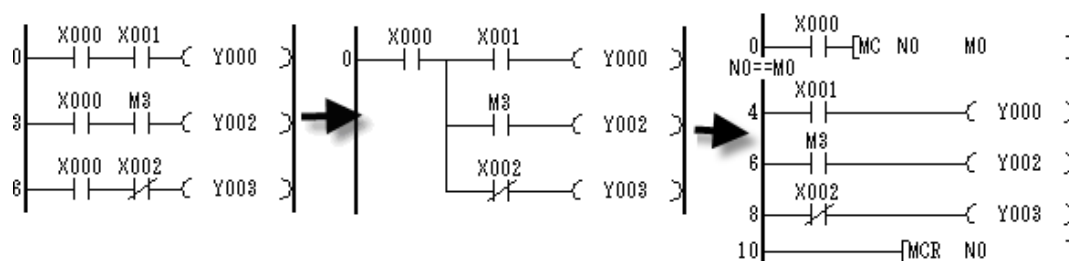
编程例三，四层堆栈：



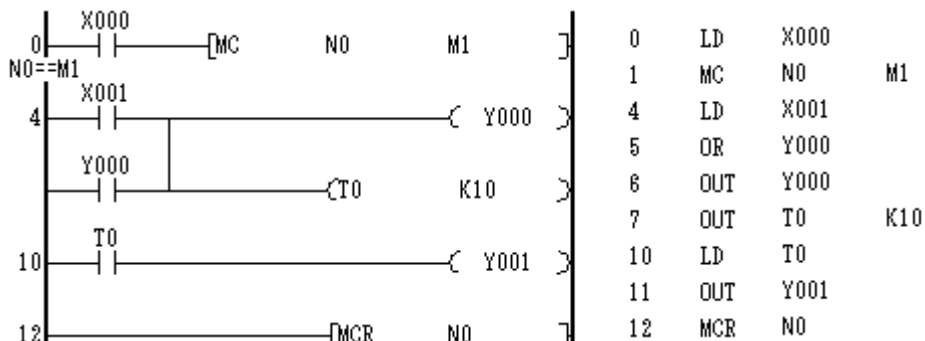
上面编程例三可以使用纵接输出的形式就可以不采用 MPS 指令了，请授课人员补充。

六、主控指令 MC、MCR

在程序中常常会有这样的情况，多个线圈受一个或多个触点控制，要是在每个线圈的控制电路中都要串入同样的触点，将占用多个存储单元，应用主控指令就可以解决这一问题，如下图。



1. 程序举例：



2. 例题解释：1) 当 X0 接通时，执行主控指令 MC 到 MCR 的程序；

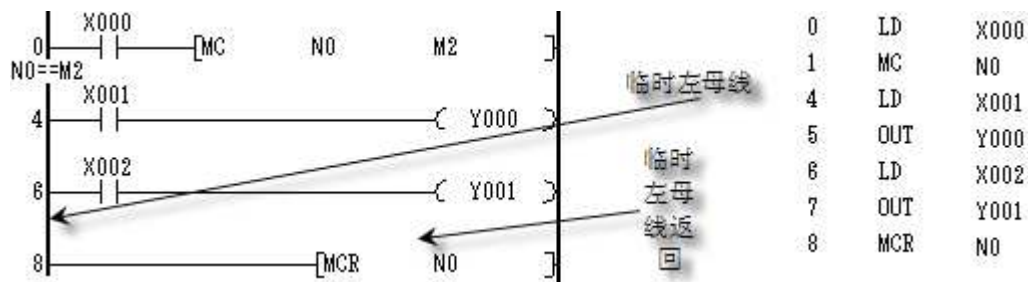
2) MC 至 MCR 之间的程序只有在 X0 接通后才能执行。

3. 指令说明：

- MC 指令的操作软元件 N、M
- 在上述程序中，输入 X0 接通时，直接执行从 MC 到 MCR 之间的程序；如果 X0 输入为断开状态，则根据不同的情况形成不同的形式：
保持当前状态：积算定时器（T63）、计数器、SET/RST 指令驱动的软元件；
断开状态：非积算定时器、用 OUT 指令驱动的软元件。
- 主控指令（MC）后，母线（LD、LDI）临时移到主控触点后，MCR 为其将临时母线返回原母线的位置的指令。
- MC 指令的操作元件可以是继电器 Y 或辅助继电器 M（特殊继电器除外）；
- MC 指令后，必须用 MCR 指令使临时左母线返回原来位置；
- MC/MCR 指令可以嵌套使用，即 MC 指令内可以再使用 MC 指令，但是必须使嵌套

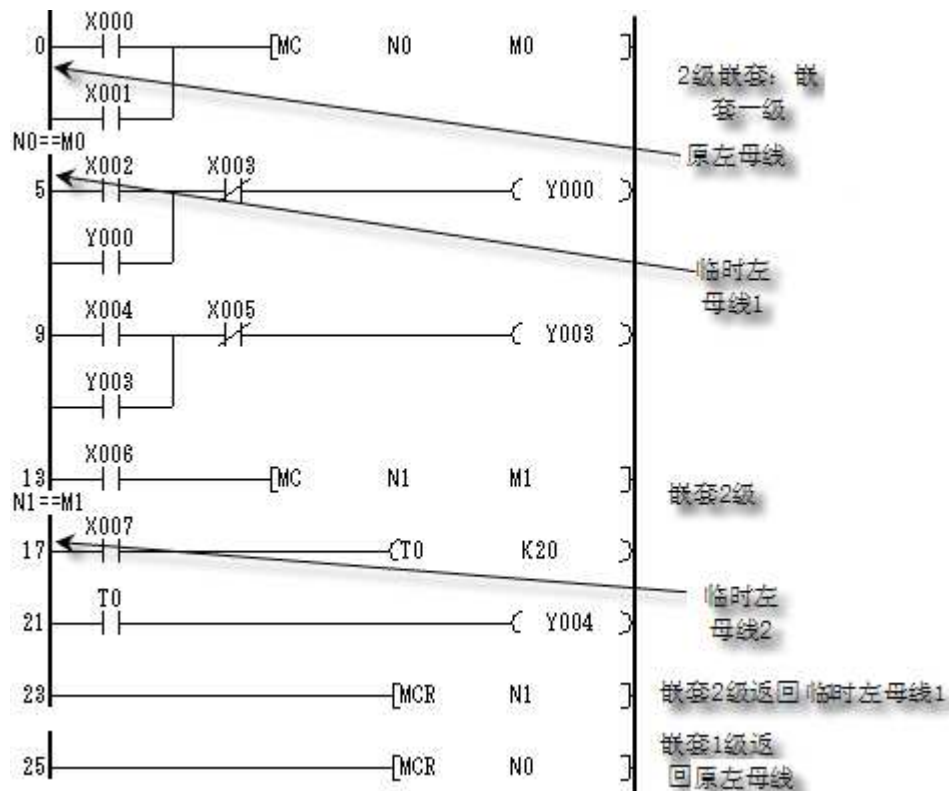
级编号从 N0 到 N7 按顺序增加，顺序不能颠倒；而主控返回则嵌套级标号必须从大到小，即按 N7 到 N0 的顺序返回，不能颠倒，最后一定是 MCR N0 指令；

无嵌套：

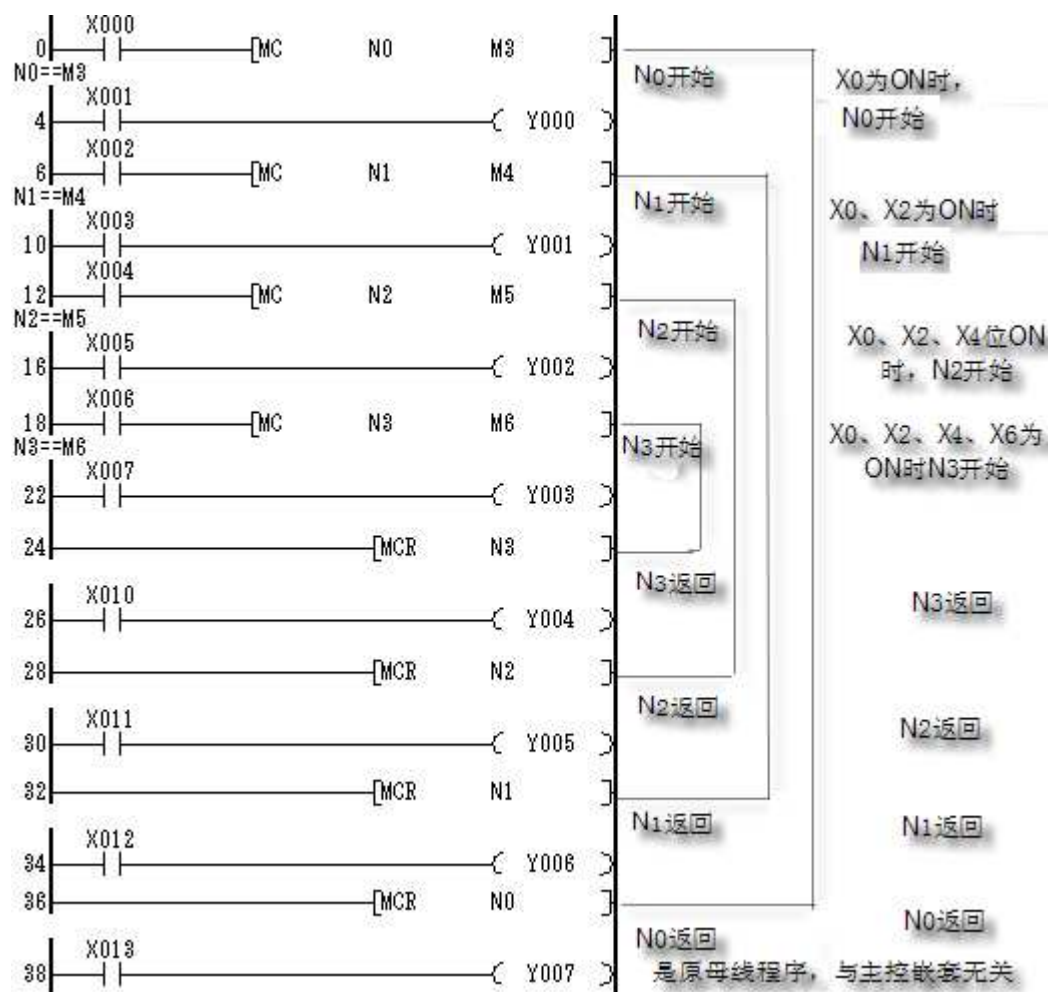


上述程序为无嵌套程序，操作元件 N 编程，且 N 在 N0—N7 之间任意使用没有限制；有嵌套结构时，嵌套级 N 的地址号增序使用，即 N0—N7。

有嵌套一：



有嵌套二：



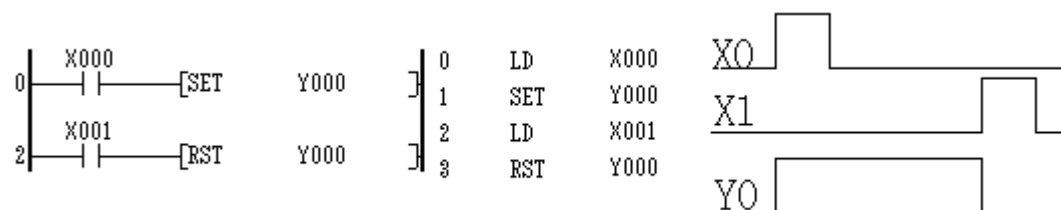
七、置 1 指令 SET、复 0 指令 RST

在前面的学习中我们了解到了自锁，自锁可以使动作保持。那么下面我们要学习的指令也可以做到自锁控制，并且在 PLC 控制系统中经常用到的一个比较方便的指令。

SET 指令称为置 1 指令：功能为驱动线圈输出，使动作保持，具有自锁功能。

RST 指令称为复 0 指令：功能为清除保持的动作，以及寄存器的清零。

1. 程序举例：



2. 例题解释：1) 当 X0 接通时，Y0 接通并自保持接通；

2) 当 X1 接通时，Y0 清除保持。

3. 指令说明：

- 在上述程序中，X0 如果接通，即使断开，Y0 也保持接通，X1 接通，即使断开，Y0 也不接通。
- 用 SET 指令使软元件接通后，必须要用 RST 指令才能使其断开。
- 如果二者对同一软元件操作的执行条件同时满足，则复 0 优先。
- 对数据寄存器 D、变址寄存器 V 和 Z 的内容清零时，也可使用 RST 指令。

5) 积算定时器 T63 的当前值复 0 和触点复位也可用 RST。

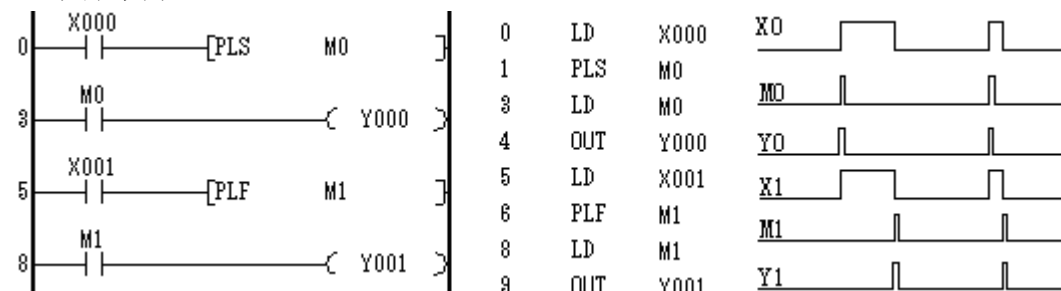
八、上升沿微分脉冲指令 PLS、下降沿微分脉冲指令 PLF

脉冲微分指令主要作为信号变化的检测，即从断开到接通的上升沿和从接通到断开的下降沿信号的检测，如果条件满足，则被驱动的软元件产生一个扫描周期的脉冲信号。

PLS 指令：上升沿微分脉冲指令，当检测到逻辑关系的结果为上升沿信号时，驱动的操作软元件产生一个脉冲宽度为一个扫描周期的脉冲信号。

PLF 指令：下降沿微分脉冲指令，当检测到逻辑关系的结果为下降沿信号时，驱动的操作软元件产生一个脉冲宽度为一个扫描周期的脉冲信号。

1. 程序举例：



2. 例题解释：1) 当检测到 X0 的上升沿时，PLS 的操作软元件 M0 产生一个扫描周期的脉冲，Y0 接通一个扫描周期。

2) 当检测到 X1 的上升沿时，PLF 的操作软元件 M1 产生一个扫描周期的脉冲，Y1 接通一个扫描周期。

3. 指令说明：

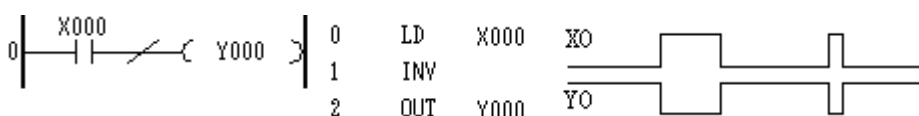
- 1) PLS 指令驱动的软元件只在逻辑输入结果由 OFF 到 ON 时动作一个扫描周期；
- 2) PLF 指令驱动的软元件只在逻辑输入结果由 ON 到 OFF 时动作一个扫描周期；
- 3) 特殊辅助继电器不能作为 PLS、PLF 的操作软元件。

九、INV 取反指令

INV 指令是将即将执行 INV 指令之前的运算结果反转的指令，无操作软元件。

INV 指令即将执行前的运算结果	INV 指令执行后的运算结果
OFF	ON
ON	OFF

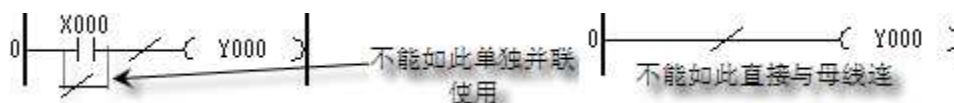
1. 程序举例：



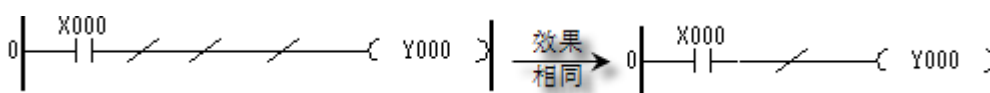
2. 例题解释：X0 接通，Y0 断开；X0 断开，Y0 接通。

3. 指令说明：

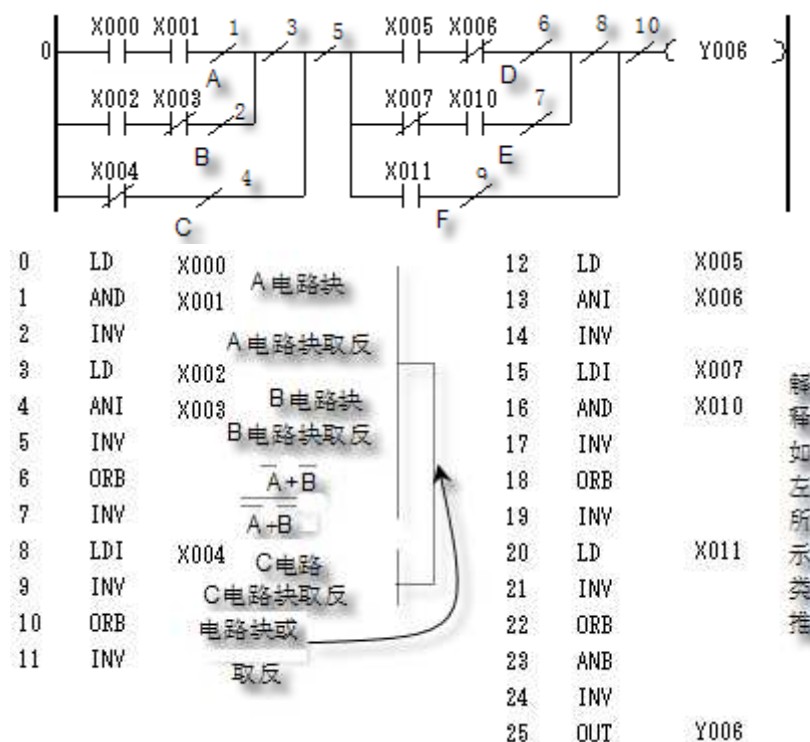
- 1) 编写 INV 取反指令需要前面有输入量，INV 指令不能直接与母线相连接，也不能如 OR、ORI、ORP、ORF 单独并联使用；



- 2) 可以多次使用，只是结果只有两个，要么通要么断；



3) INV 指令只对其前的逻辑关系取反。



如上图，在包含 ORB 指令、ANB 指令的复杂电路中使用 INV 指令编程时，INV 的取反动作如指令表中所示，将各个电路块开始处的 LD、LDI、LDP、LDF 指令以后的逻辑运算结果作为 INV 运算的对象。

十、空操作指令 NOP、结束指令 END

1. NOP 指令：称为空操作指令，无任何操作元件。其主要功能是在调试程序时，用其取代一些不必要的指令，即删除由这些指令构成的程序；另外在程序中使用 NOP 指令，可延长扫描周期。若在普通指令与指令之间加入空操作指令，可编程序控制器可继续工作，就如没有加入 NOP 指令一样；若在程序执行过程中加入空操作指令，则在修改或追加程序时可减少步序号的变化。
2. END 指令：称为结束指令，无操作元件。其功能是输入输出处理和返回到 0 步程序。
3. 指令说明：
 - 1) 在将程序全部清除时，存储器内指令全部成为 NOP 指令；
 - 2) 若将已经写入的指令换成 NOP 指令，则电路会发生变化；
 - 3) 可编程序控制器反复进行输入处理、程序执行、输出处理，若在程序的最后写入 END 指令，则 END 以后的其余程序步不再执行，而直接进行输出处理；
 - 4) 在程序中没 END 指令时，可编程序控制器处理完其全部的程序步；
 - 5) 在调试期间，在各程序段插入 END 指令，可依次调试各程序段程序的动作功能，确认后再删除各 END 指令；
 - 6) 可编程序控制器在 RUN 开始时首次执行是从 END 指令开始；
 - 7) 执行 END 指令时，也刷新监视定时器，检测扫描周期是否过长。

十一、LDP、LDF、ANDP、ANDF、ORP、ORF 指令

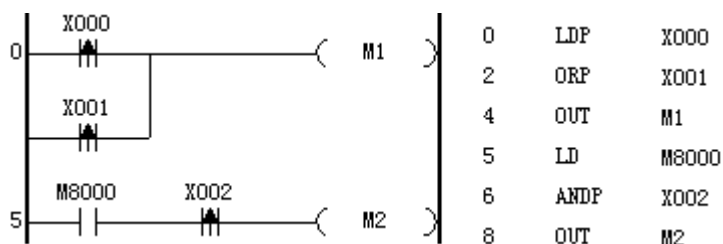
1. LDP：上升沿检测运算开始(检测到信号的上升沿时闭合一个扫描周期)。
- LDF：下降沿检测运算开始(检测到信号的下降沿时闭合一个扫描周期)
- ANDP：上升沿检测串联连接(检测到位软元件上升沿信号时闭合一个扫描周期)
- ANDF：下降沿检测串联连接(检测到位软元件下降沿信号时闭合一个扫描周期)

ORP: 脉冲上升沿检测并联连接(检测到位软元件上升沿信号时闭合一个扫描周期)

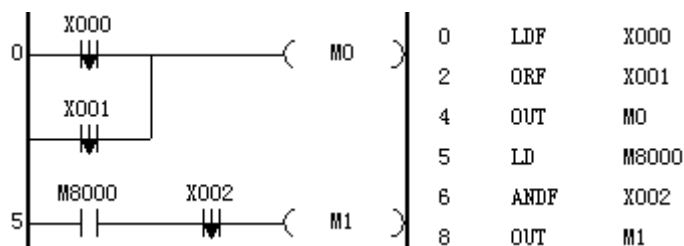
ORF: 脉冲下降沿检测并联连接(检测到位软元件下降沿信号时闭合一个扫描周期)

2. 上述 6 个指令的操作软元件都为 X、Y、M、S、T、C。

3. 程序举例:



在上面程序里, X0 或 X1 由 OFF——ON 时, M1 仅闭合一个扫描周期; X2 由 OFF——ON 时, M2 仅闭合一个扫描周期。



在上面程序里, X0 或 X1 由 ON——OFF 时, M0 仅闭合一个扫描周期; X2 由 ON——OFF 时, M1 仅闭合一个扫描周期。

所以上述两个程序都可以使用 PLS、PLF 指令来实现。

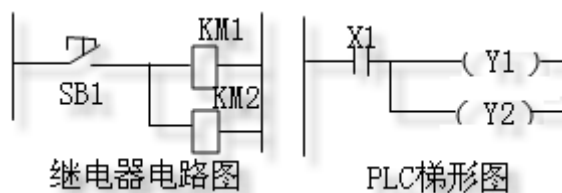
第三节 PLC 的编程及应用

一、PLC 编程特点

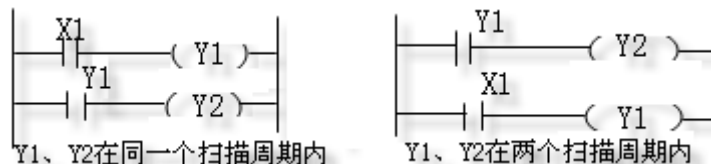
梯形图是 PLC 中最常用的方法, 它源于传统的继电器电路图, 但发展到今天两者之间已经有了极大的差别。

PLC 的梯形图有一条左母线, 相当于继电器电路的电源正极, 还有一条右母线, 相当于电源负极。

1. 程序执行顺序比较



2. PLC 程序的扫描执行结果



3. PLC 软件特性

PLC 在梯形图里可以无数次地使用其触点, 既可以是常闭也可以是常开。

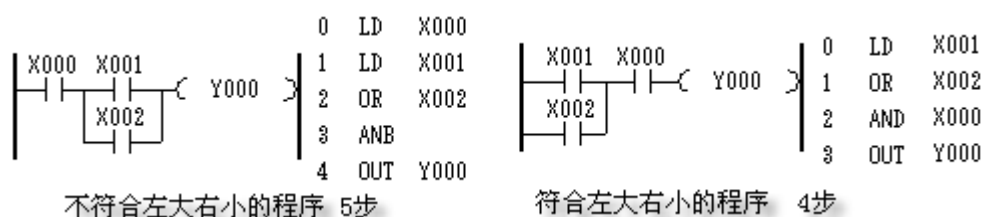
二、PLC 编程的基本规则

一) PLC 编程应遵循以下基本规则:

1. 输入/输出继电器、辅助继电器、定时器、计数器等软元件的触点可以多次重复使用，无需复杂的程序结构来减少触点的使用次数。
2. 梯形图每一行都是从左母线开始，线圈止于右母线。触点不能直接接右母线；线圈不能直接接左母线。
3. 在程序编写中一般不允许双重线圈输出，步进顺序控制除外。
4. 可编程控制器程序编写中所有的继电器的编号，都应在所选 PLC 软元件列表表范围内。
5. 梯形图中不存在输入继电器的线圈。

二) 合理设计梯形图

1. 程序的编写应按照自上而下、从左到右的方式编写。为了减少程序的执行步数，程序应“左大右小、上大下小”，尽量不出现电路块在左边或下边的情况。

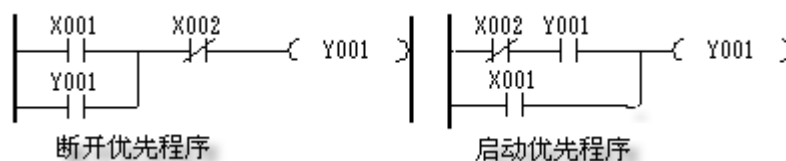


2. 依照扫描的原则，程序处理时尽可能让同时动作的线圈在同一个扫描周期内。

三、典型控制程序

1. 自保持程序

自保持电路也称自锁电路。常用于无机械锁定开关的启动停止控制中。如用无机械锁定功能的按钮控制电动机的启动和停止；并且分为启动优先和断开优先两种。



2. 互锁程序

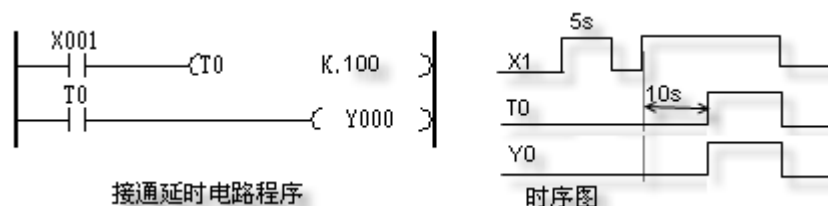
互锁电路用于不允许同时动作的两个或多个继电器的控制，如电动机的正反转控制。



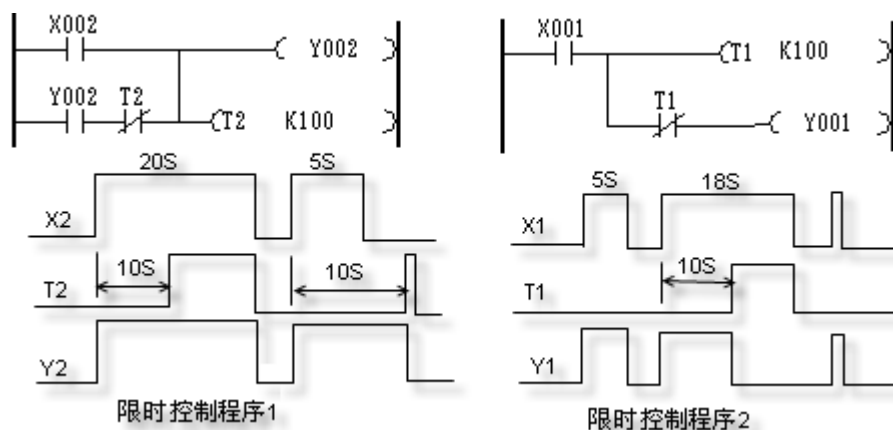
3. 时间电路程序

时间电路程序主要用于延时、定时和脉冲控制。时间控制电路，既可以用以用定时器实现也可以用标准时钟脉冲实现。在 FX1S 系列有 64 个定时器和四种标准时钟脉冲（1min、1S、100ms、10ms）可用于时间控制，编程时使用方便。

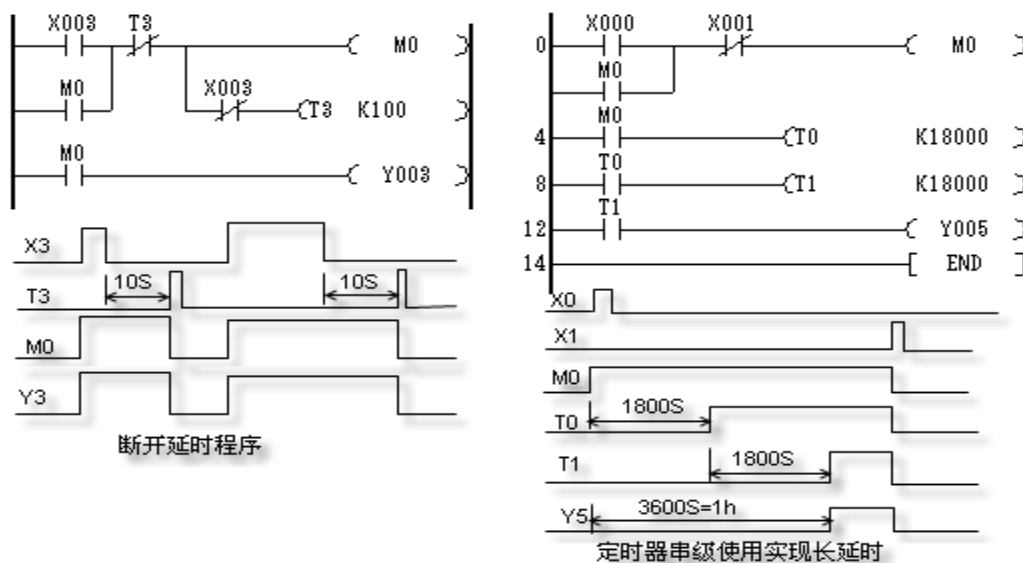
1) 接通延时



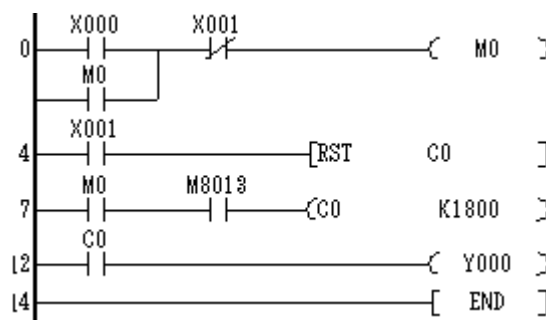
2) 限时控制程序



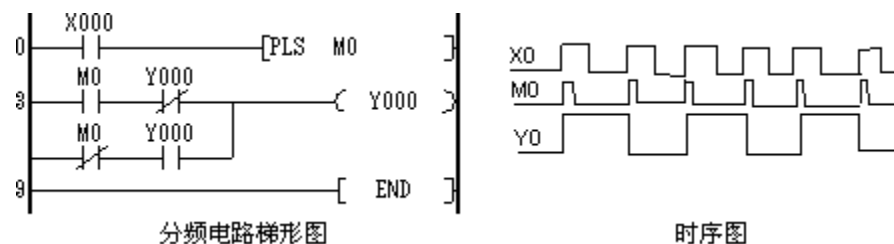
3) 断开延时和长延时



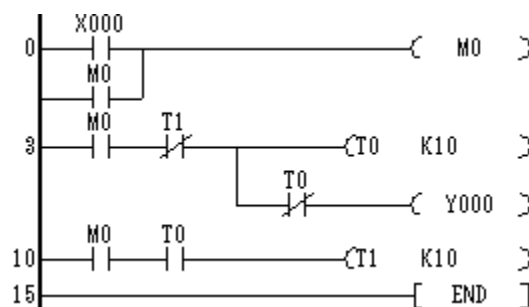
4) 计数器配合计时



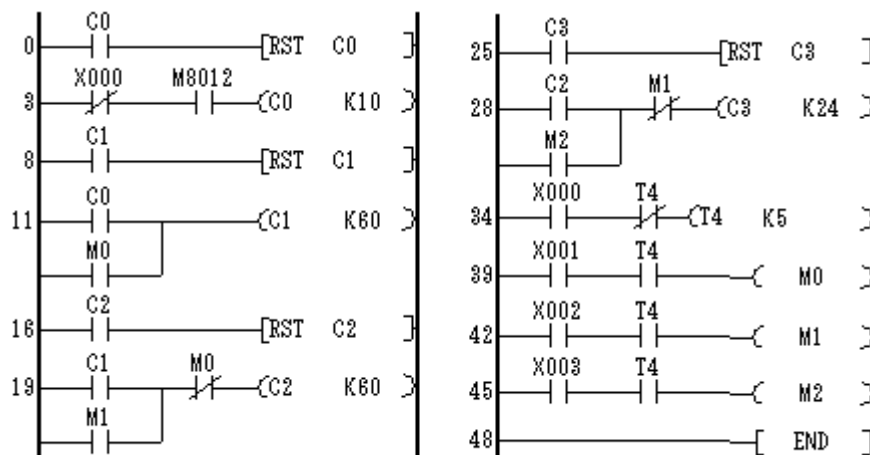
4. 分频电路程序



5. 振荡电路程序



6. 时钟电路程序



时钟电路程序

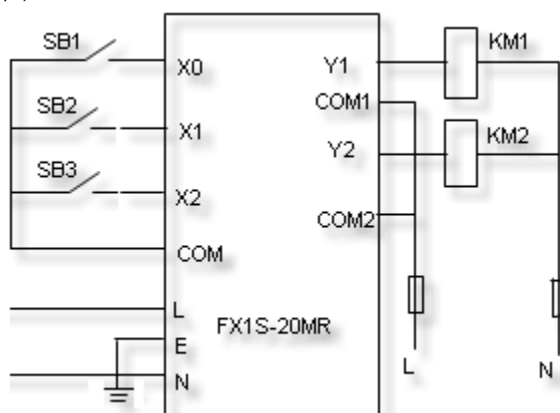
四、编程实例

一) 电动机正反转控制

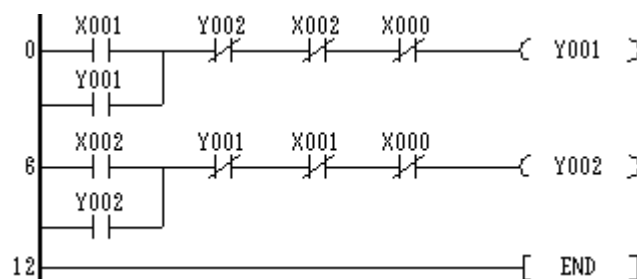
1. 分析工艺过程
2. PLC 的 I/O 点的确定和分配

输 入			输 出		
SB1	停止按钮	X0	KM1	接触器	Y1
SB2	正转按钮	X1	KM2	接触器	Y2
SB3	反转按钮	X2			

3. PLC 接线图



4. 程序编写



5. 调试。

二) 锅炉点火和熄火控制

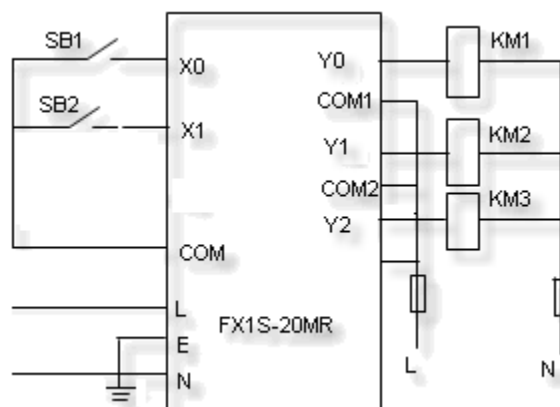
控制要求为：点火过程为先启动引风，5 分钟后启动鼓风，2 分钟后点火燃烧；熄火过程为先熄灭火焰，2 分钟后停止鼓风，5 分钟后停止引风。

1. 分析工艺过程

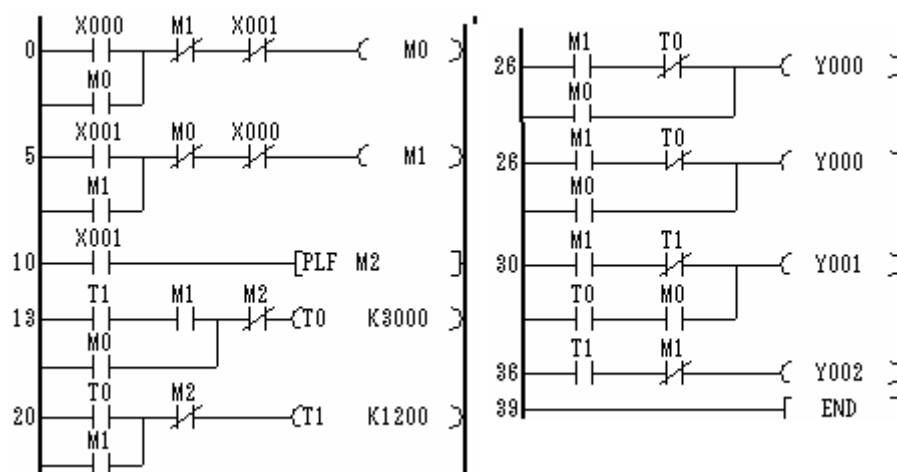
2. PLC 的 I/O 点的确定与分配

输 入		输 出		
点火信号	X0	控制引风	KM1	Y0
熄火信号	X1	控制鼓风	KM2	Y1
		控制点火开关	KM3	Y2

3. PLC 接线图



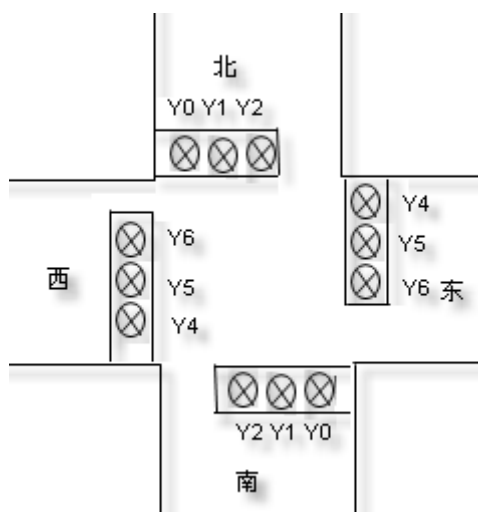
4. 控制程序编写



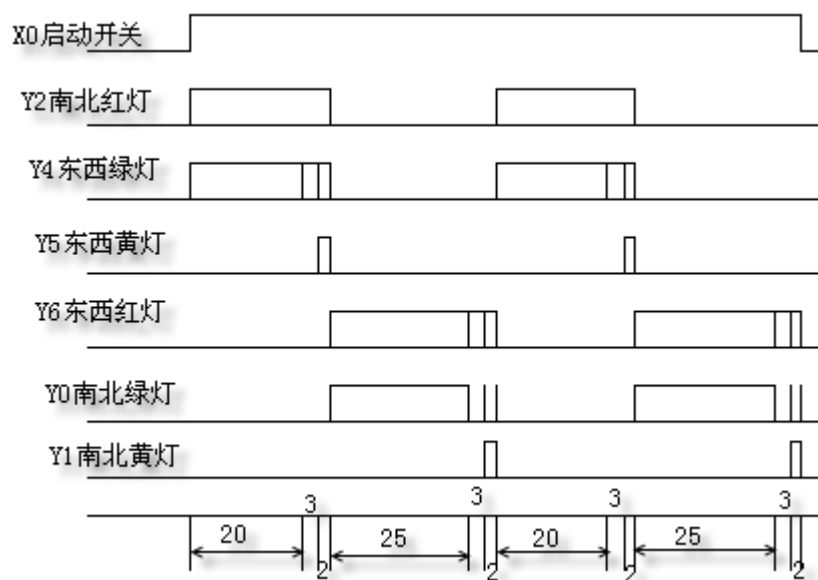
5. 调试。

三) 十字路口交通信号灯控制

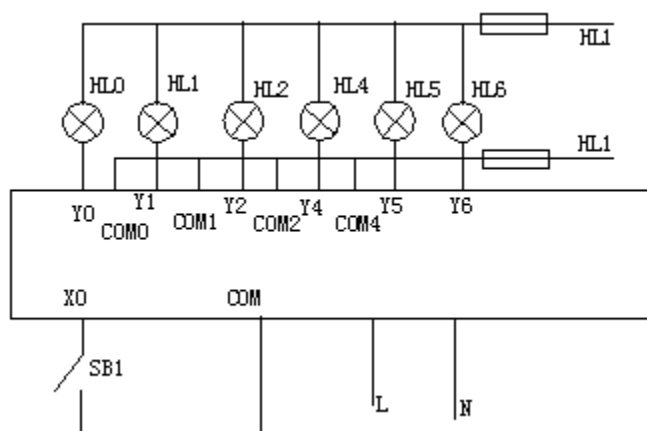
1. 分析工艺过程



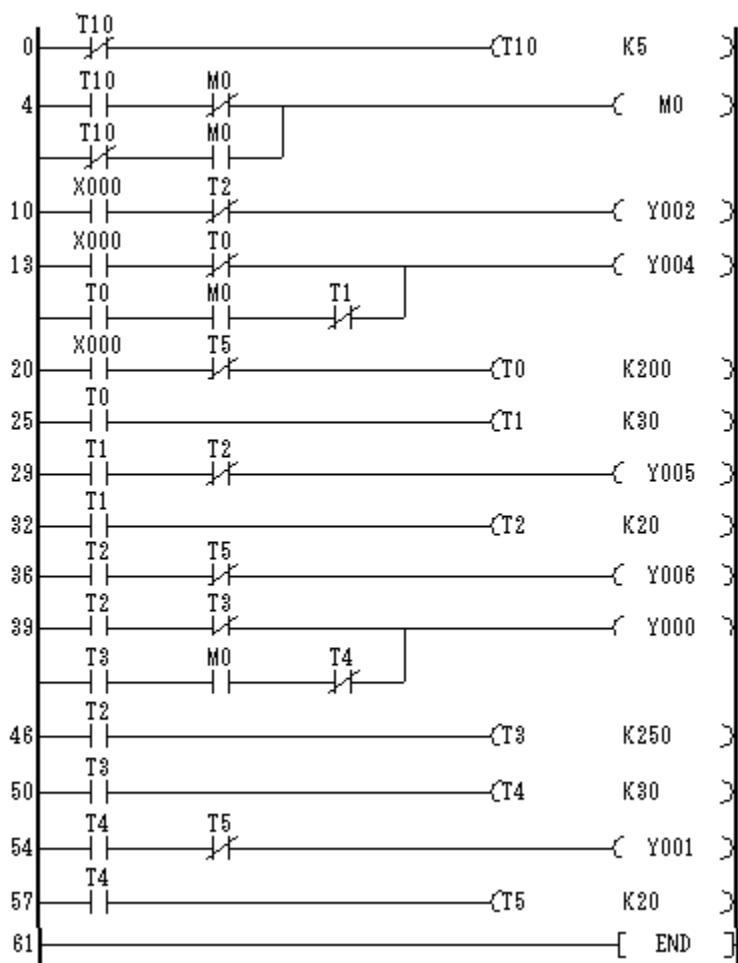
2. 十字路口交通信号灯控制时序图要求



3. I/O 分配如上图，PLC 接线图



4. 程序编写



5. 调试。

习题：

1. 基本逻辑指令都由哪几部分组成？各指令的功能？
2. 请画出以下指令表的梯形图。

0 LD X000 11

ORB

1 MPS 12

ANB

2 LD X001 13 0

UT Y001

3	OR	X002	14	M
	PP			
4	ANB		1	
5	AND	X007		
5	OUT	Y000	16	0
	UT	Y002		
6	MRD		1	
7	LD	X010		
7	LDI	X003	18	OR
	I	X011		
8	AND	X004	19	AN
	B			
9	LD	X005	20	
	OUT	Y003		
10	ANI	X006		

3. 画出题 3 指令语句表的梯形图。

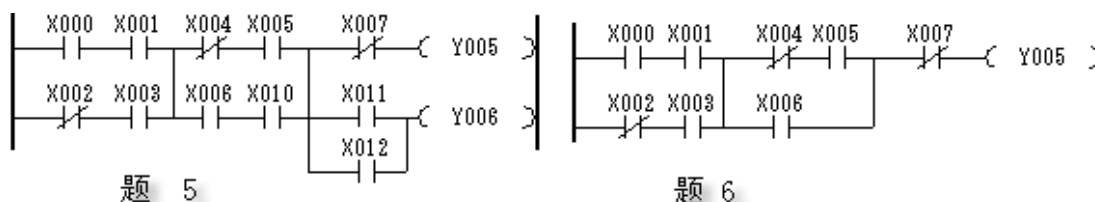
0	LD	X000	0	LD	X000	9	OUT	Y000
1	ANI	M0	1	MPS		10	MPP	
2	OUT	M0	2	AND	X001	11	OUT	Y001
3	LDI	X000	3	MPS		12	MPP	
4	RST	C0	4	AND	X002	13	OUT	Y002
6	LD	M0	5	MPS		14	MPP	
7	OUT	C0	6	AND	X003	15	OUT	Y003
10	LD	C0	7	MPS		16	MPP	
11	OUT	Y000	8	AND	X004	17	OUT	Y004
12	END					18	END	

题 3

题 4

4. 画出题 4 指令语句表的梯形图。

5. 写出题 5 梯形图的指令语句表。

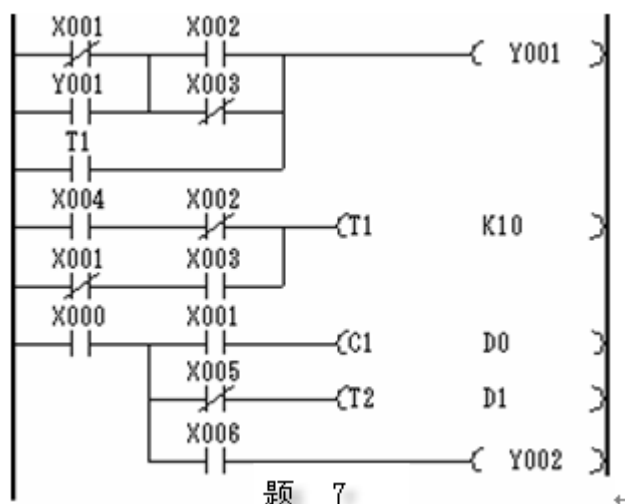


题 5

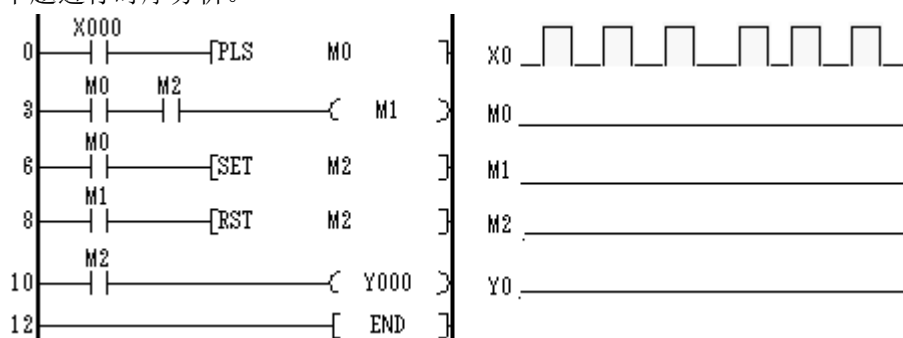
题 6

6. 写出上面题 6 梯形图的指令语句表。

7. 写出题 7 梯形图的指令语句表。

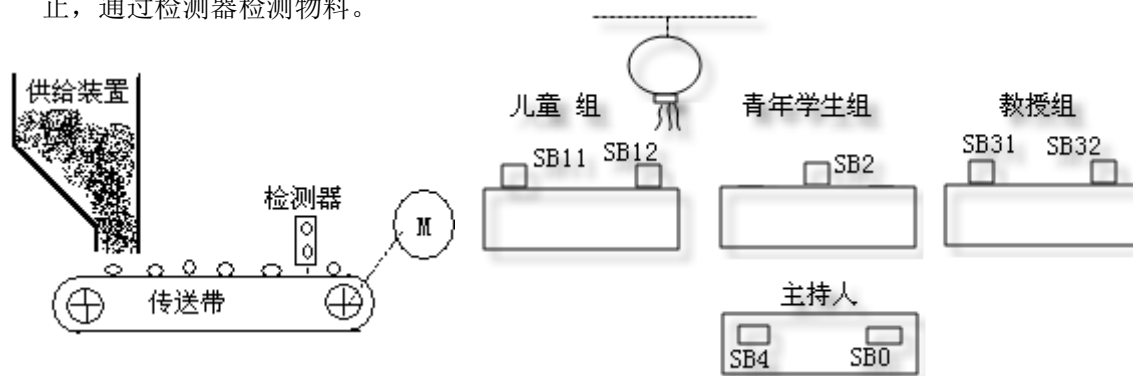


8. 对下题进行时序分析。



9. 试设计一个四分频的梯形图，并写出对应的指令语句表，画出输入信号及输出信号的状态时序图。

10. 运行过程中，若传送带上 15 秒无物料通过则报警，报警时间延续 30 秒后传送带停止，通过检测器检测物料。



11. 抢答器显示系统。请按系统设计步骤答题，控制要求：

- 1) 竞赛者若要回答主持人提出的问题，必须抢先按下按钮。
- 2) 指示灯亮后，需等到主持人按下复位按钮 SB4 后才熄灭，为了给参赛儿童一些优待，SB11 和 SB12 中任意一个按钮按下时，灯 L1 都亮；而为了对教授组做一定的限制，L3 只有在 SB31 和 SB32 都按下时才亮。
- 3) 如果竞赛者在主持人打开 SB0 开关的 10 秒钟内按下按钮，电磁阀将使彩球摇动，以示竞赛者得到一次幸运机会。
12. 有一个指示灯，控制要求为：按下启动按钮后，亮 5 秒灭 5 秒，重复 5 次后停止。

试设计梯形图。

13. 有两台三相异步电动机 M1 和 M2，要求：

- 1) M1 启动后，M2 才能启动；
- 2) M1 停止后，M2 延时 30 秒后才能停止；
- 3) M2 能点动调整。

试作出 PLC 输入输出分配接线图，并编写梯形图控制程序。

14. 设计两台电动机顺序控制 PLC 系统。控制要求：两台电动机相互协调运转，M1 运转 10S，停止 5S，M2 要求与 M1 相反，M1 停止 M2 运行，M1 运行 M2 停止，如此反复动作 3 次，M1 和 M2 均停止。

15. 根据时序图写出梯形图程序。



第四章 PLC 步进顺序控制系统

在前章的学习中我们了解了 PLC 的一些基本编程，用基本逻辑指令在做一些顺序控制，特别是较为复杂的顺序控制时，不很直观。因此 PLC 厂家开发出了专门用于顺序控制的指令，在三菱 FX 系列中为 STL、RET 一组指令，从而使得顺序控制变得直观简单。

PLC 是典型的开环顺序控制系统。我们在日常生活和工业生产中常常要求机器设备能实现某种顺序控制功能，即要求机器能按照某种预先规定的顺序、以及各种环境输入讯号来自动实现所期望的动作。比如一个配料系统，我们可能对其运转提出以下要求：

1. 先装入原料 A，直到液面配料桶容积的一半；
2. 再装入原料 B，直到液面配料桶容积的 75%；
3. 然后开始持续搅拌 20 秒；
4. 最后停止搅拌，开启出料阀，直到液位低于配料桶的 5%后再延时 2 秒，最后关闭出料阀；
5. 以上过程反复进行。

由此可见，顺序控制系统中的动作存在确定的先后关系，即顺序，且后面的动作必须根据前面的动



作情况来确定。本章主要介绍其指令及编程方法。

第一节 顺序控制及状态转移图

根据状态转移图，采用步进指令可对复杂的顺序控制进行编程。为了对步进指令灵活地运用，我们在此应对顺序控制和状态转移图的概念加强了解。

一、顺序控制

所谓顺序控制，就是按照生产工艺所要求的动作规律，在各个输入信号的作用下，根据内部的状态和时间顺序，使生产过程的各个执行机构自动地、有秩序地进行操作。

在顺序控制中，生产过程是按顺序、有秩序地连续工作。因此可以将一个较复杂的生产过程分解成若干步骤，每一步对应生产过程中的一个控制任务，即一个工步或一个状态。且每个工步往下进行都需要一定的条件，也需要一定的方向，这就是转移条件和转移方向。

二、状态继电器

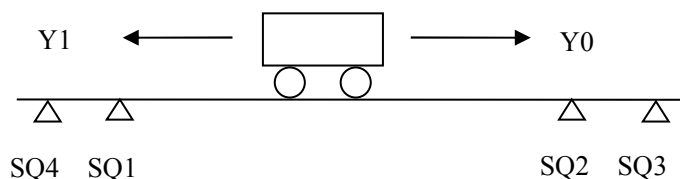
在状态转移图中，每个状态都分别采用连续的、不同的状态继电器表示。FX1S 系列 PLC 的状态继电器的分类、编号、数量及功能。

类 别	状态继电器编号	数 量	功能说明
初始化状态继电器	S0——S9	10 点	初始化
原点回归状态继电器	S10——S127	118 点	用 IST 指令时原点回归
掉电保持型继电器	S0——S127	128 点	具有停电记忆功能，停电后再启动，可继续执行

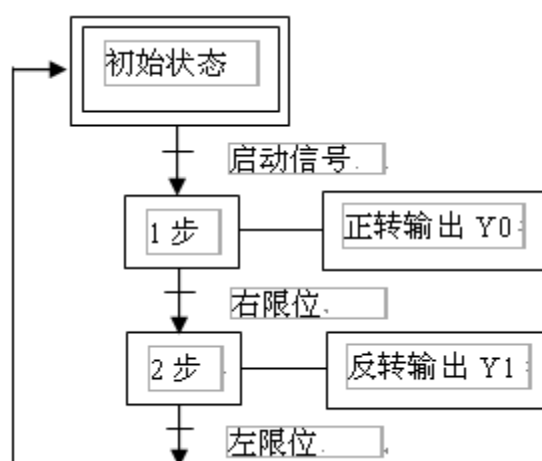
在用状态转移图编写程序时，状态继电器可以按顺序连续使用。但是状态继电器的编号要在指定的类别范围内选用；各状态继电器的触点可自由使用，使用次数无限制；在不用状态继电器进行状态转移图编程时，状态继电器可做为辅助继电器使用，用法和辅助继电器相同。

三、状态转移图的设计法

何谓状态转移图（系统状态）设计法，系统程序设计一般有两种思路：一是针对某一具体对象（输出）来考虑），另一种就是功能图设计法。它把整个系统分成几个时间段，在这段时间里可以有一个输出，也可有多个输出，但他们各自状态不变。一旦有一个变化，系统即转入下一个状态。给每一个时间段设定一个状态器（步进接点），利用这些状态器的组合控制输出。例如工作台自动往复控制系统,我们可以画出它的状态转移图：一工作台自动往复控制程序（1）要求：正反转启动信号 SB0、SB1，停车信号 SB2，左右限位开关 SQ1、SQ2，左右极限保护开关 SQ3、SQ4，输出信号 Y0、Y1。具有电气互锁和机械互锁功能。



状态转移图：



四、状态转移的实现

任何一个顺序控制过程都可分解为若干步骤，每一工步就是控制过程中的一个状态，所以顺序控制的动作流程图也称为状态转移图，状态转移图就是用状态（工步）来描述控制过程的流程图。

在状态转移图中，一个完整的状态必须包括：

- （1） 该状态的控制元件；
- （2） 该状态所驱动的对象；
- （3） 向下一个状态转移的条件；
- （4） 明确的转移方向。

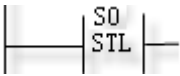
状态转移的实现，必须满足两个方面：一是转移条件必须成立，二是前一步当前正在进行。二者缺一不可，否则程序的执行在某些情况下就会混乱。

五、画状态转移图的一般步骤

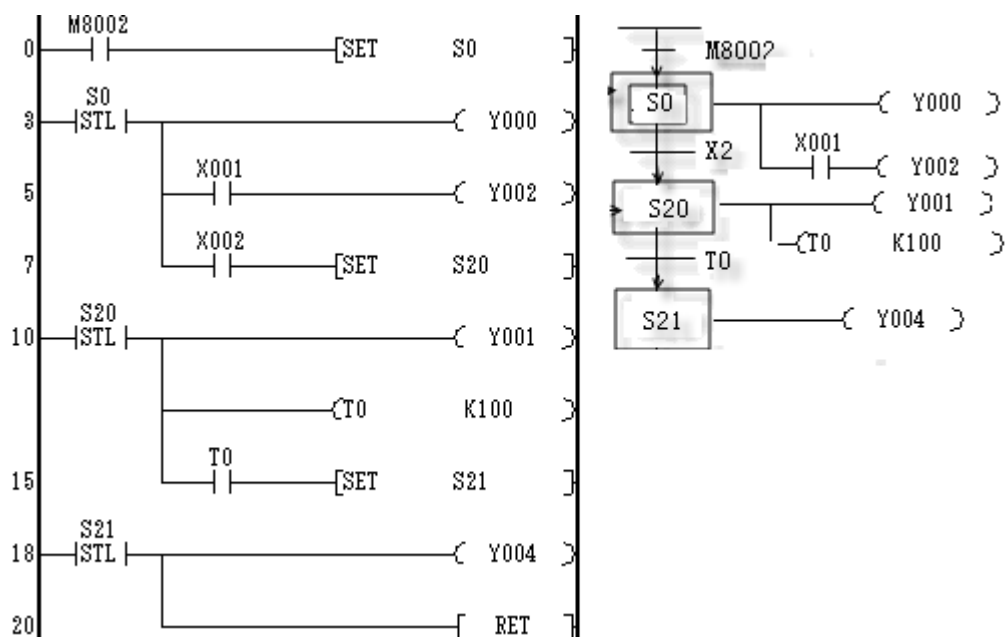
- (1) 分析控制要求和工艺流程，确定状态转移图结构（复杂系统需要）。
- (2) 工艺流程分解若干步，每一步表示一稳定状态。
- (3) 确定步与步之间转移条件及其关系。
- (4) 确定初始状态。（可用输出或状态器）
- (5) 解决循环及正常停车问题。
- (6) 急停信号的处理。

第二节 步进顺序控制指令

我们知道每一个状态都有一个控制元件来控制该状态是否动作，保证在顺序控制过程中，生产过程有秩序地按步进行，所以顺序控制也称为步进控制。FX1S 采用状态继电器作为控制元件，并且只利用其常开触点来控制步动作。控制状态的常开触点称为步进

接点，在梯形图中用符号  表示。

当利用 SET 指令将状态继电器置 1 时，步进接点闭合。此时，顺序控制就进入该步进接点所控制的状态。当转移条件满足时，利用 SET 指令将下一个状态控制元件（即状态继电器）置 1 后，上一个状态继电器（上一工步）自动复位，而不必采用 RST 指令复位。用梯形图表示：



状态转移图用梯形图表示的方法：

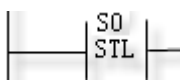
- (1) 控制元件：梯形图中画出状态继电器的步进接点；
- (2) 状态所驱动的对象：依照状态转移图画出；
- (3) 转移条件：转移条件用来 SET 下一个步进接点；
- (4) 转移方向：往哪个方向转移，就是 SET 置 1 的步进接点控制元件。

根据上述所学知识，我们学习步进指令(STL、RET)。

一、步进指令 STL、RET

1. STL 指令

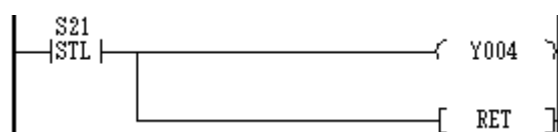
STL 指令称为“步进接点”指令。其功能是将步进接点接到左母线。

格式：

操作元件：状态继电器 S。

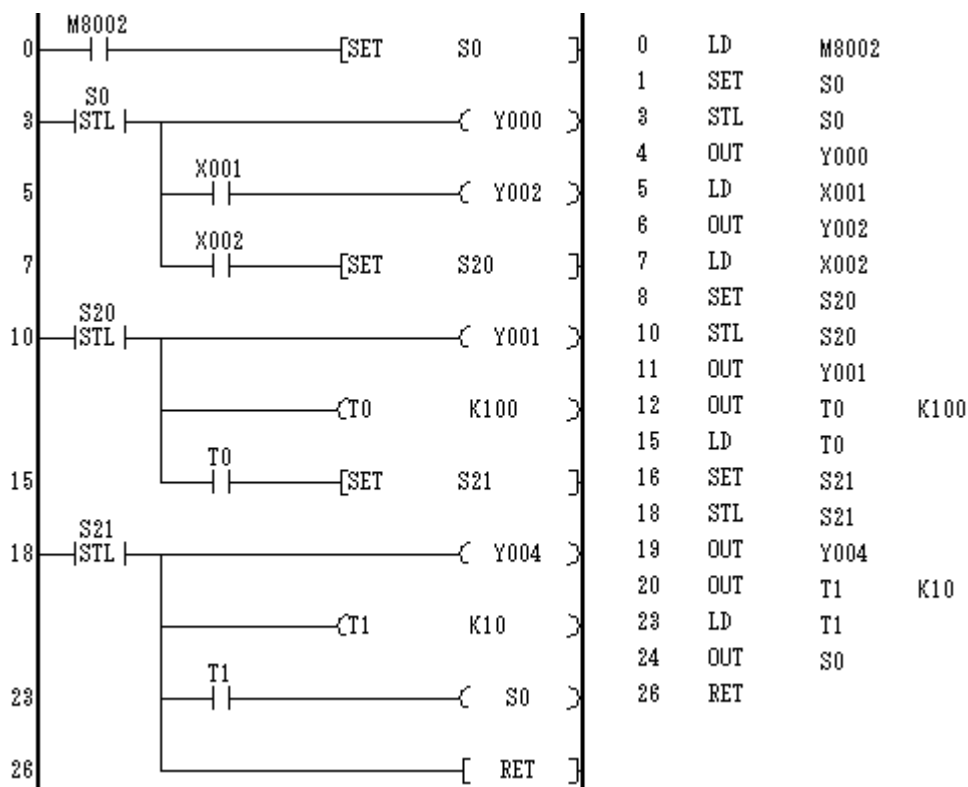
RET 指令称为“不仅返回”指令。其功能是使临时左母线回到原来左母线的位置。

格式：



操作元件：无。

程序举例：



步进接点只有常开触点，没有常闭触点。步进接通需要 SET 指令进行置 1，步进接点闭合，将左母线移动到临时左母线，与临时左母线相连的触点用 LD、LDI 指令，如上图。在每条步进指令后不必都加一条 RET 指令，只需在连续的一系列步进指令的最后一条的临时左母线后接一条 RET 指令返回原左母线，且必须有这条指令。

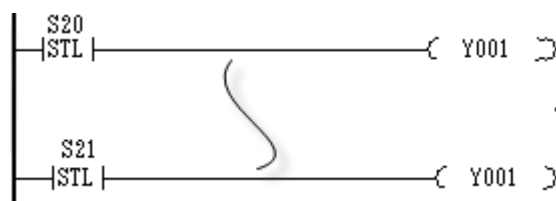
2. 指令说明：

- (1) 步进接点与左母线相连时，具有主控和跳转作用；
- (2) 状态继电器 S 只有在使用 SET 指令以后才具有步进控制功能，提供步进接点。
- (3) 在状态转移图中，会出现在一个扫描周期内两个或两个以上状态同时动作的可能因此在相邻的步进接点必须有联锁措施。
- (4) 状态继电器在不仅状态转移图中使用可以按编号顺序使用，也可以任意。但是建议按顺序。
- (5) 状态继电器可做辅助继电器使用，与辅助继电器 M 用法相同。
- (6) 步进接点后的电路中不允许使用 MC/MCR 指令。
- (7) 在状态内，不能从 STL 临时左母线位置直接使用 MPS/MRD/MPP。

二、编程与动作，步进梯形图

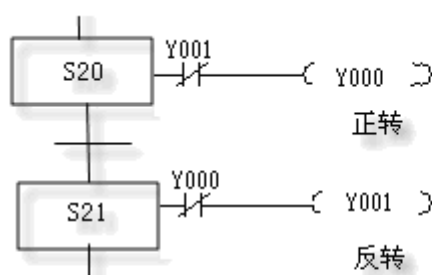
1. 状态的动作与输出的重复使用

- (1) 状态的地址号不能重复使用；
- (2) 如果 STL 触点接通，则与其相连的电路动作；如果 STL 触点断开，则与其相连的电路停止动作；
- (3) 如图所示，在不同的步之间可给同一软元件编程。

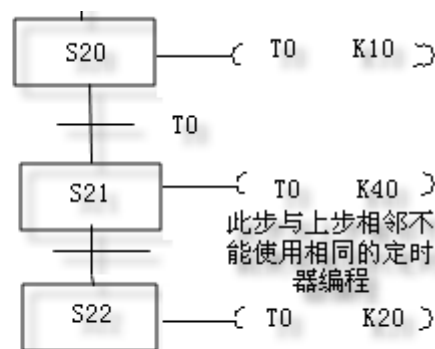


2. 输出的联锁

在状态转移过程中，仅在瞬间（一个扫描周期）两种状态同时接通，因此为了避免同时接通的一对输出同时接通，需要设置联锁；



输出的联锁程序



定时器相邻时不能编程

3. 定时器的重复使用

定时器线圈与输出线圈一样，也可对在不同的状态的同一软元件编程，但是在相邻的状态中不能编程。如果在相邻的状态下编程，则步进状态转移时定时器线圈不断开，当前值不能复位，如果不是相邻的两个状态则可以使用同一个定时器，如输出联锁图示。

三、单流程、多分支状态转移图的编程与梯形图的转换

1. 如何应用步进指令进行编程 采用步进指令编程时一般需要下面几个步骤：

- 1) 分析工艺过程；
- 2) 分配 I/O，列出输入输出分配表；

- 3) 画出 PLC 接线图;
 - 4) 根据工艺要求分析的结果, 画出顺序控制的状态转移图;
 - 5) 状态转移图转换成梯形图或指令语句表;
 - 6) 输入程序到 PLC;
 - 7) 运行调试。
2. 单流程: 没有分支的状态转移图称为单流程。

有一个机械动作如图所示。1) 按下启动按钮台车前进, 一直到限位开关 LS11 动作, 台车后退,

2) 台车后退时, 直到限位开关 LS12 动作, 停 5 秒后再前进, 直到限位开关 LS13 动作, 台车后退;

3) 不久限位开关再动作, 这时驱动台车的电机停止。

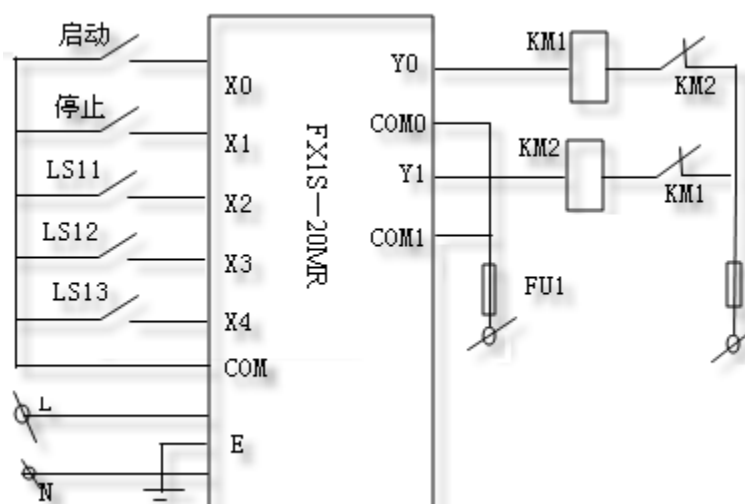


1) 解: 图中给出了台车机械动作的过程, 分作两次前进和后退, 进程长度不一样。

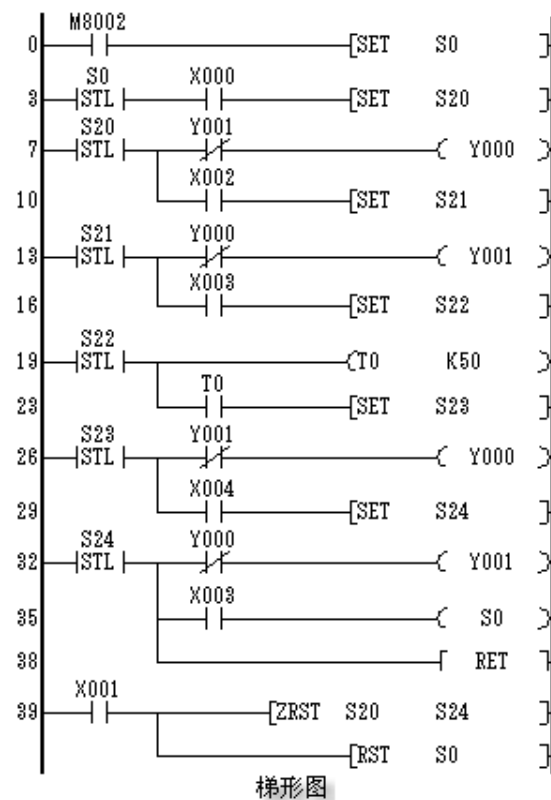
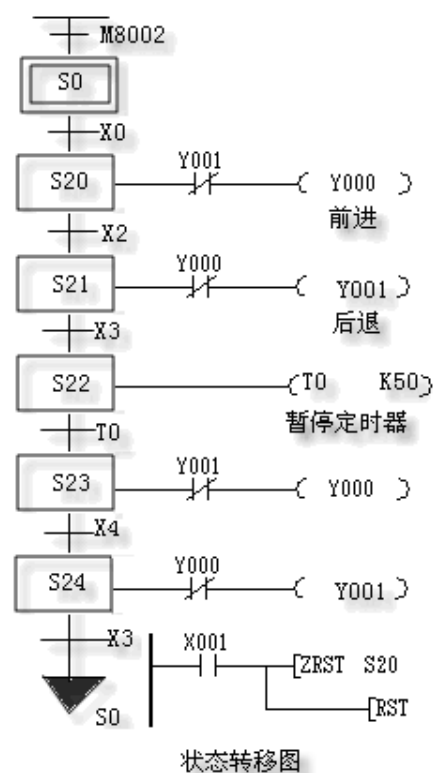
2) I/O 分配。

输 入		输 出	
启动按钮	X0	前进	Y0
停止按钮	X1	后退	Y1
开关 LS11	X2		
开关 LS12	X3		
开关 LS13	X4		

3) 画出 PLC 接线图



4) 状态转移图程序



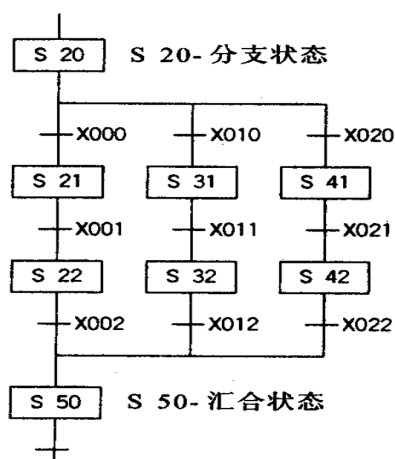
0	LD	M8002	20	OUT	T0	K50
1	SET	S0	23	LD	T0	
3	STL	S0	24	SET	S23	
4	LD	X000	26	STL	S23	
5	SET	S20	27	LDI	Y001	
7	STL	S20	28	OUT	Y000	
8	LDI	Y001	29	LD	X004	
9	OUT	Y000	30	SET	S24	
10	LD	X002	32	STL	S24	
11	SET	S21	33	LDI	Y000	
13	STL	S21	34	OUT	Y001	
14	LDI	Y000	35	LD	X003	
15	OUT	Y001	36	OUT	S0	
16	LD	X003	38	RET		
17	SET	S22	39	LD	X001	
19	STL	S22	40	ZRST	S20	S24
			45	RST	S0	

5) 调试程序 可以在实习室试验操作时根据机械动作操作，观察结果。

3. 选择性分支与汇合

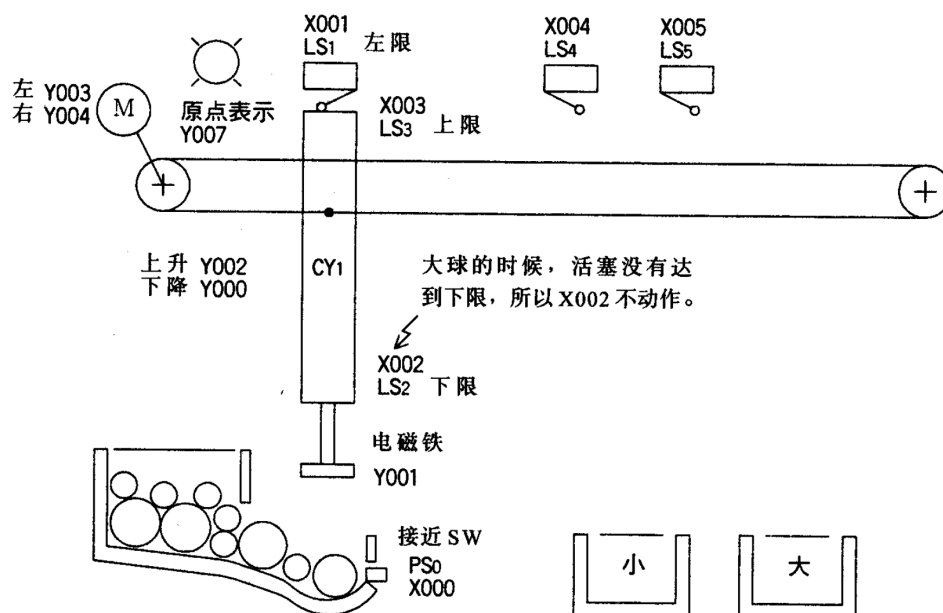
所谓选择性分支就是从多个流程中选择执行一个流程。最简单地就说就是前面有两条或则多条路，我们只能选择走一条，那么选择性分支也就是这个意思，每条路都通向某一个相同的地方，那么那个地方就是汇合处，例如抢答器就是这个原理，每次只有一个能抢答到。

选择性分支与汇合的状态转移图和梯形图之间的转换：

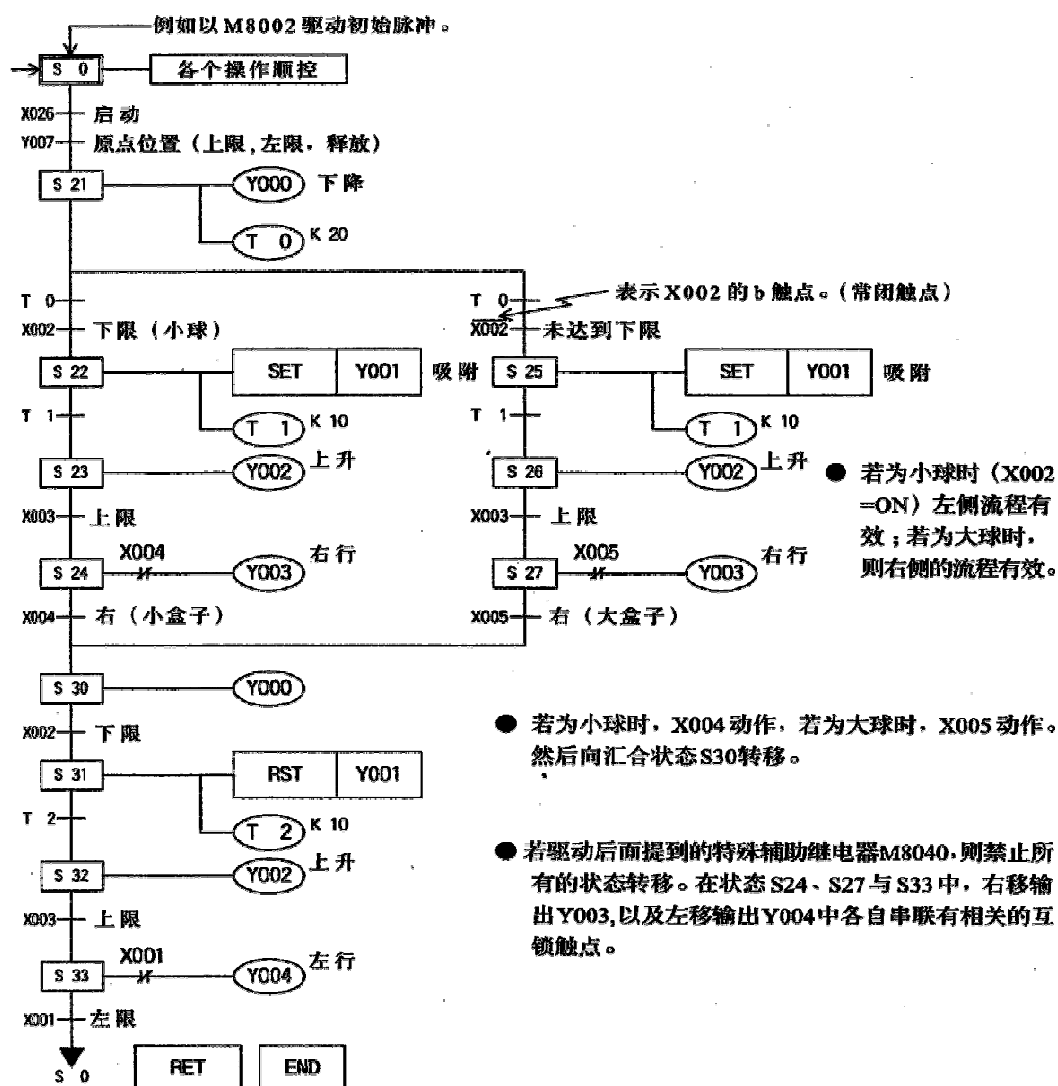


以左图为例，必须是 x000 , x010, x020 不同时接通。例如，在 S20 动作时，若 x000 接通，则动作状态就向 S21 转移，S20 变为不动作。因此，即使以后 X010 、 X020 动作， S31 、 S41 也不会动作。汇合状态 S50 ，可被 S22 、 S32 、 S42 中任意一个驱动。

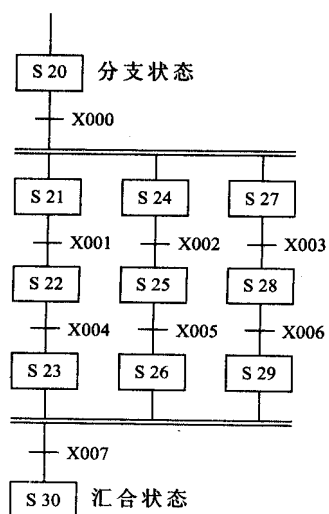
例题：下图为使用传送带，将大、小球分类选择传送的机械。左上方为原点，其动作顺序为下降、吸住、上升、右行、下降、释放、上升、左行。此外，机械臂下降，当电磁铁压着大球时，下限限位开关 LS2 断开；压着小球时，LS2 导通。



像这种大小分类选择或判别合格与否的 SFC 图，可用下图所示的选择性分支与汇合的 SFC 图表示。



4. 并行性分支与汇合

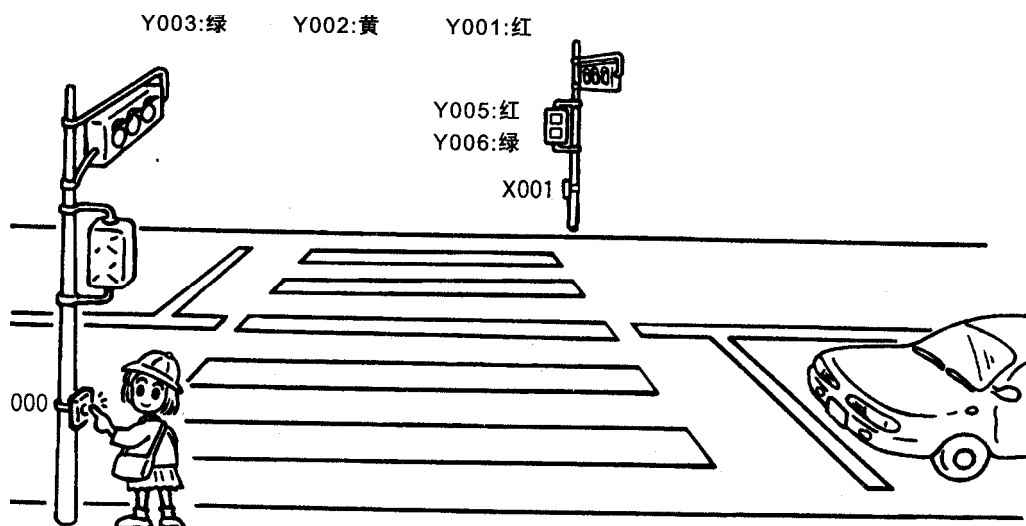


多个流程全部同时执行的分支被称为并行分支。以左

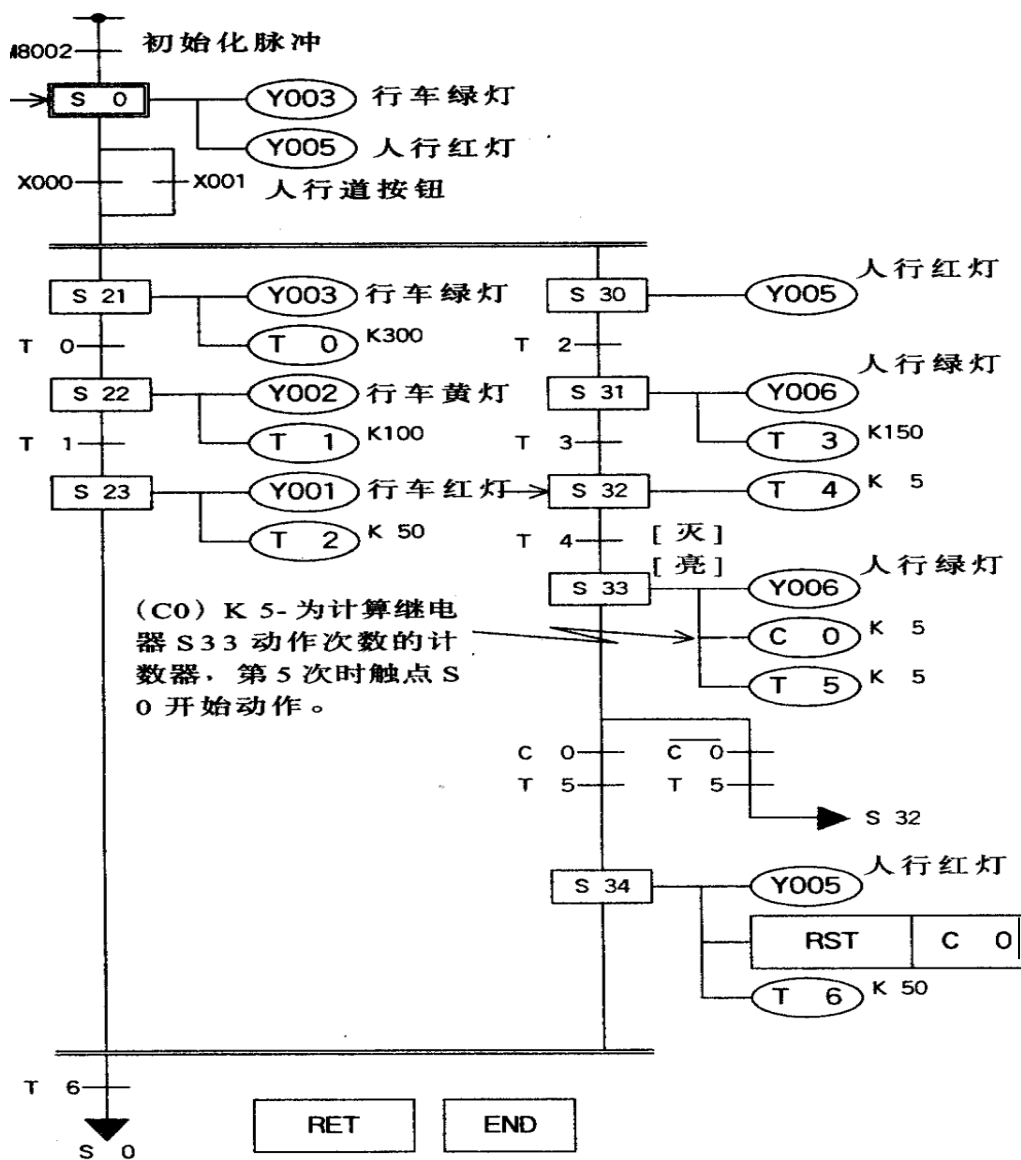
图为例,在 S20 动作时,若 X000 接通,则 S21、S24、S27 同时动作,各分支流程开始动作。当各流程动作全部结束时,若 X7 接通,则汇合状态 S30 开始动作,转移前的各状态 S23、S26、S29 全部变为不动作。这种汇合,有时又被称为等待汇合。(先完成的流程要等所有流程动作结束后,再汇合,继续动作。)

将零件 A、B、C 分别并行加工,零件加工后进行装配,这也是并行型分支与汇合流程。

例题: 下图以按钮式人行横道线为例,说明并行分支与汇合的流程。



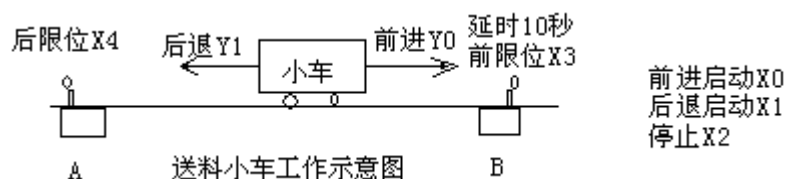
可编程控制器从 STOP——RUN 变换时,, 初始状态 S0 动作,通常车道信号灯为绿,而人行道信号灯为红。。按下人行道按钮 X000 或 X001,则状态 S21 为车道二绿;状态 S30 中的人行道信号已经为红色,此时状态无变化。30 秒钟后,车道信号为黄灯;再过 10 秒钟车道信号变为红灯。此后,定时器 T2(5 秒钟)启动,5 秒后人行道变为绿灯。15 秒后,人行道绿灯开始闪烁。(S32=暗, S33=亮)。闪烁中时 S32、S33 反复动作,计数器 C0 (设定值为 5 次)触点一接通,动作状态向 S34 转移,人行道变为红灯,5 秒后返回初始状态。。在动作过程中,即使按动人行道按钮 X000、X001 也无效。程序如下:

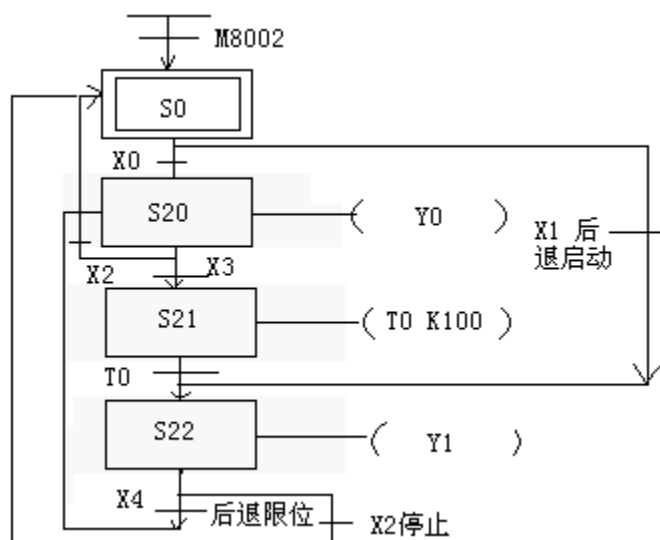


第三节 编程实例

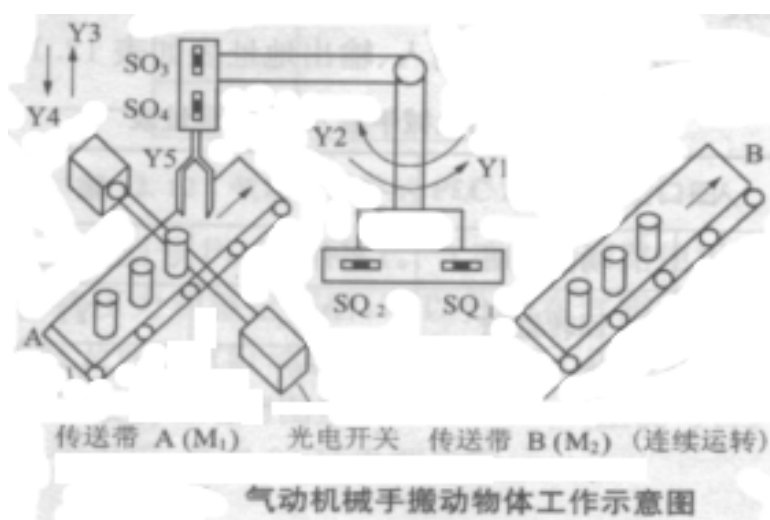
步进指令用于顺序控制具有极其方便独特之处。在这一节内容中我们通过实际例题进一步理解步进顺序控制。

- 一、某送料小车工作示意图如下, 小车可以在 A、B 之间正向启动前进和反向启动后退; 小车前进到 B 处停车, 延时 10 秒后返回, 后退至 A 处停车后立即返回; 按下停止按钮, 小车停在 A、B 之间任一位置。





二、气动机械手的任务大多数是搬运物品或器件，并且其快速准确。例如下图，将传送带 A 上的物品搬至传送带 B 上，它是设计把电子器件取来送至印刷电路板，按规定的动作和规律运行。



硬件系统如上图，控制要求：传送带 A 为步进式传送，每当机械手从传送带上取走一个物品时，该传送带向前步进一段距离，将使机械手在下一个工作循环取到物品。传送带 A、B 分别由电动机 M1、M2 驱动，机械手回转运动由气动阀 Y1、Y2 控制，机械手的上、下运动由气动阀 Y3、Y4 控制，机械手的夹紧与放松由气动阀 Y5 控制。右旋到位信号为 SQ1，左旋到位信号 SQ2，上升到位信号 SQ3，下降到位信号 SQ4。

- 1) 机械手在原始位置时 SQ1 动作，按下启动按钮，机械手松开，传送带 B 开始运动，机械手手臂开始上升。
- 2) 机械手上升到上限位置，SQ3 动作，上升动作结束，机械手开始左转。

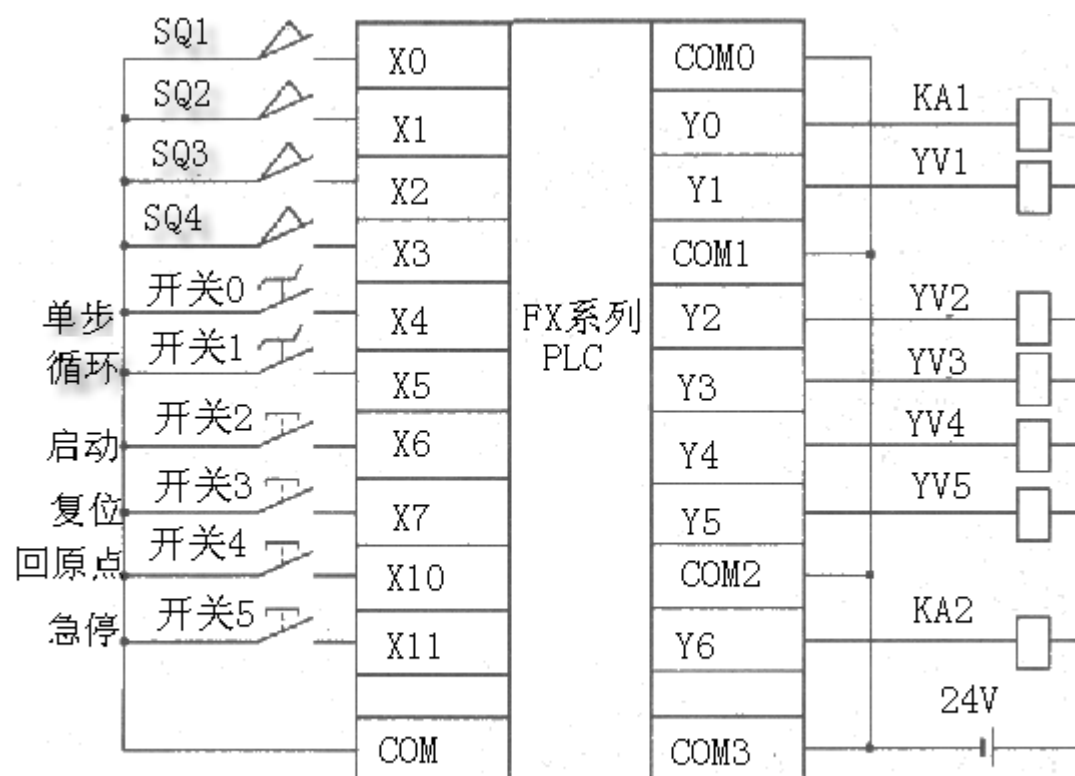
- 3) 机械手左旋到左限位, SQ2 动作, 左旋动作结束, 机械手开始下降。
- 4) 机械手下降到下限位置, SQ4 动作, 下降动作结束, 传送带 A 启动。
- 5) 传送带 A 向机械手方向前进一个物品的距离后停止, 机械手开始抓物。
- 6) 机械手抓物, 延时 1 秒钟时间, 机械手开始上升。
- 7) 机械手上升到上限位置, SQ3 动作, 上升动作结束, 机械手开始右旋。
- 8) 机械手右旋到右限位位置, SQ1 动作, 右旋动作结束, 机械手开始下降。
- 9) 机械手下降到下限位置, SQ4 动作, 机械手松开, 放下物品。
- 10) 机械手放下物品经过适当延时, 一个工作循环过程完毕。
- 11) 机械手的工作方式: 单步/循环。

解: 1. 分析工艺过程如控制要求。

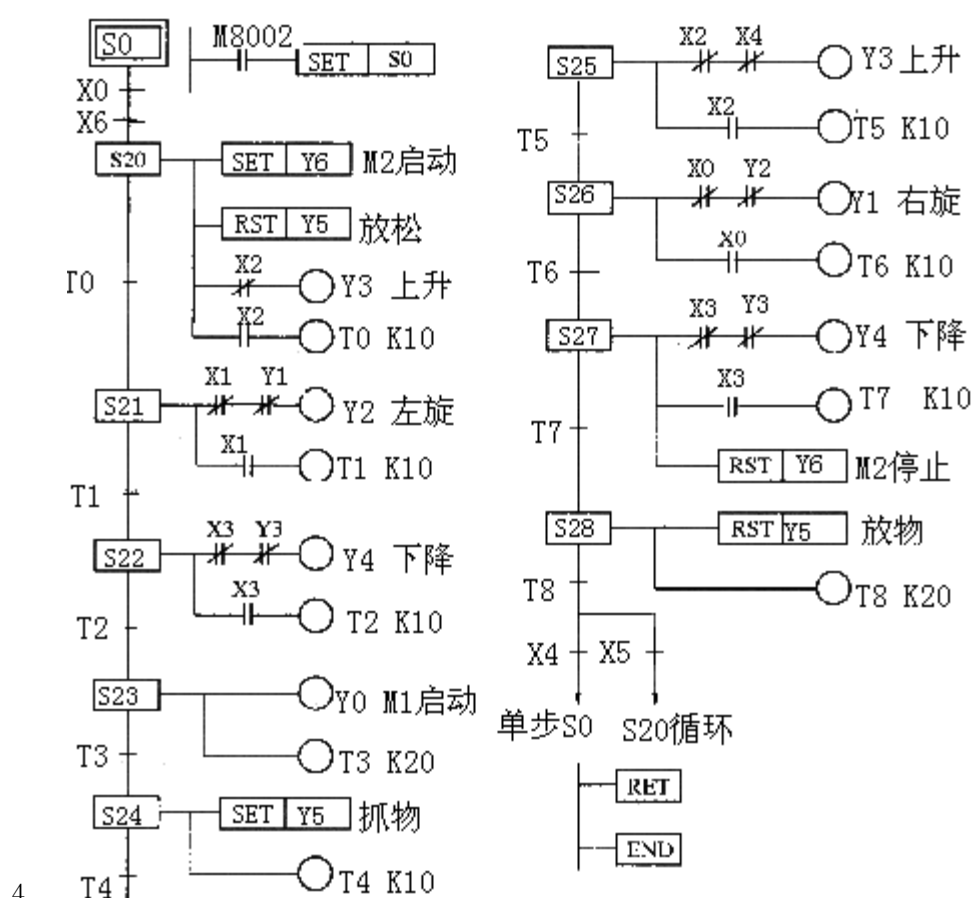
2. 确定 I/O 点数并分配。

输 入			输 出		
SQ1	X0	回转缸磁性开关	YV1	Y1	气缸右转
SQ2	X1		YV2	Y2	气缸左转
SQ3	X2	手臂上升/下降	YV3	Y3	手臂上升
SQ4	X3	气缸磁性开关	YV4	Y4	手臂下降
开关 0	X4	单步	YV5	Y5	夹紧与放松
开关 1	X5	循环	M1	Y0	传送带 A
开关 2	X6	手动启动	M2	Y6	传送带 B
开关 3	X7	手动复位			
开关 4	X10	手动到原点			
开关 5	X11	紧急停止			

3. PLC 接线图



气动机械手的PLC I/O电气接线图



4. 写 (参考)

程序编

LD	M8002	LD	X1	OUT	T4	STL	S27
SET	S0	OUT	T1		K10	LDI	X3
STL	S0		K10	LD	T4	ANI	Y3
LD	X0	LD	T1	SET	S25	OUT	Y4
AND	X6	SET	S22	STL	S25	LD	X3
SET	S20	STL	S22	LDI	X2	OUT	T7
STL	S20	LDI	X3	ANI	X4		K10
SET	Y6	ANI	Y3	OUT	Y3	RST	Y6
RST	Y5	OUT	Y4	LD	X2	LD	T7
LDI	X2	LD	X3	OUT	T5	SET	S28
OUT	Y3	OUT	T2		K10	STL	S28
LD	X2		K10	LD	T5	RST	Y5
OUT	Y3	LD	T2	SET	S26	OUT	T8
LD	X2	SET	S23	STL	S26		K20
OUT	T0	STL	S23	LDI	X0	LD	T8
	K10	OUT	Y0	ANI	Y2	AND	X4
LD	T0	OUT	T3	OUT	Y1	OUT	S0
SET	S21		K20	LD	X0	LD	T8
STL	S21	LD	T3	OUT	T6	AND	X5
LDI	X1	SET	S24		K10	OUT	S20
ANI	Y1	STL	S24	LD	T6	RET	
OUT	Y2	SET	Y5	SET	S27	END	

气动机械手控制程序指令语句表

5. 程序和控制系统调试。

第五章 可编程序控制器应用指令

本章论述可编程序控制器常用的应用指令的表示方法和规则。在使用应用指令时，需要了解指令的有关软元件的处理及其执行形式。

在基本逻辑指令的基础上，PLC 制造厂家开发了一系列完成不同功能的子程序，调用这些子程序的指令称为应用指令。FX 系列 PLC 的应用指令可分为程序控制、传送与比较、算术与逻辑运算、移位与循环等。

第一节 应用指令的基础知识

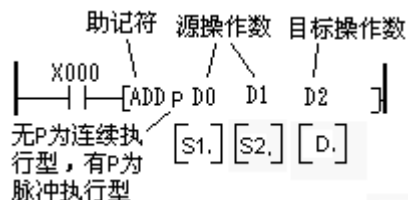
一、应用指令的基本格式

- 应用指令有三部分组成：功能编号 FNC---，助记符，操作数。

- 梯形图形式：

- 梯形图输入同一个应用指令：

- 应用指令的含义



二、应用指令的规则

- 应用指令操作数（软元件）的含义如表：

字 软 元 件	位 软 元 件
K：十进制整数	X：输入继电器
H：十六进制整数	Y：输出继电器
KnX：输入继电器 X 的位指定	M：辅助继电器
KnY：输出继电器 Y 的位指定	S：状态继电器
KnS：状态继电器 S 的位指定	
T：定时器 T 的当前值	
C：计数器 C 的当前值	

D: 数据寄存器	
V、Z: 变址寄存器	

2. 指令执行形式有连续执行和脉冲执行两种。
3. 位软元件 处理断开和闭合状态的元件为位软元件；

字软元件 处理数据的元件称字软元件。

由位软元件组合起来也可以构成字软元件，进行数据处理；每 4 个位软元件为一组，组合成一个单元，位软元件的组合由 Kn (n 在 1 至 7 之间) 加首元件来表示。如 KnY 、 KnX 等， $K1Y0$ 表示由 $Y0$ 、 $Y1$ 、 $Y2$ 、 $Y3$ 组成的 4 位字软元件； $K4M0$ 表示由 $M0$ —— $M15$ 组成的十六位字软元件。

变址寄存器都是十六位数据寄存器。表示时如果 $V=5$ ， $Z=10$ ，则 $D5V=D10$ ($5+5=10$)， $D5Z=D15$ ($5+10=15$)。32 位指令中 V 、 Z 是自动组对使用， V 作为高 16 位， Z 作为低 16 位，使用时只需编写 Z 。

第二节 应用指令及其应用

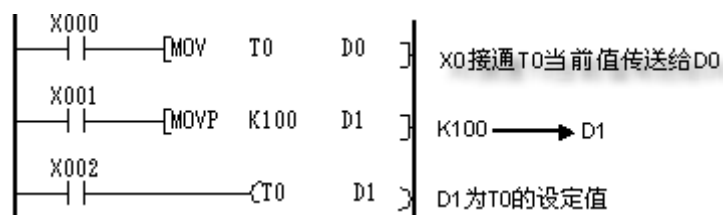
应用指令分为程序流程控制、传送与比较、数据处理等，在这节里我们对常用应用指令作一简单介绍。

一、传送与比较指令

1. MOV 指令

功 能 编 号	助 记 符	功能	操作软元件		D 连续 执行	P 脉冲 执行
			S	D		
12	MOV	将源操作元件的数据传送到指定的目标操作元件	K、H、 KnX 、 KnY 、 KnM 、 KnS 、T、C、D、V、Z	KnY 、 KnM 、 KnS 、T、C、D、V、Z	+	+

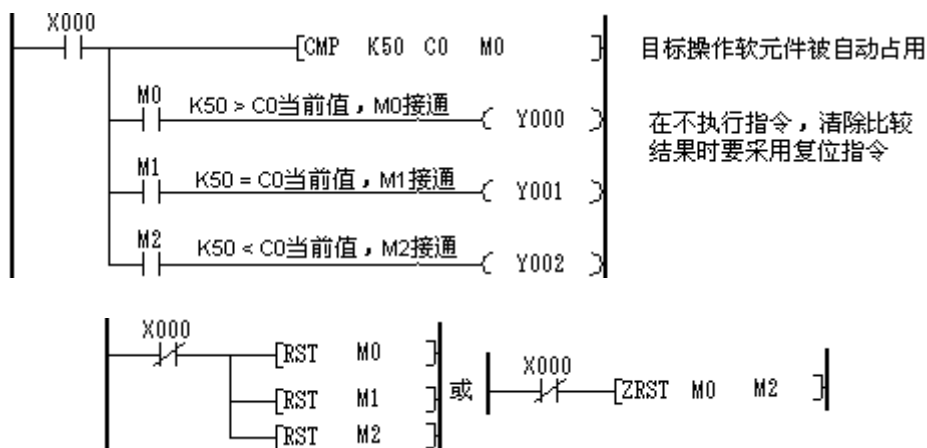
程序举例：



2. 比较指令 CMP(FNC10)、区间比较指令 ZCP(FNC11)

功 能 编 号	助 记 符	功能	操作软元件			
			S1.	S2.	S.	D.
10	CMP	将源操作软元件 S1 与 S2 的内容比较	K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z			
11	ZCP	S 与 S1、S2 区间比较				

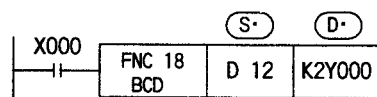
程序举例：



3. BCD(FNC18)转换指令、二进制转换指令 BIN(FNC19)

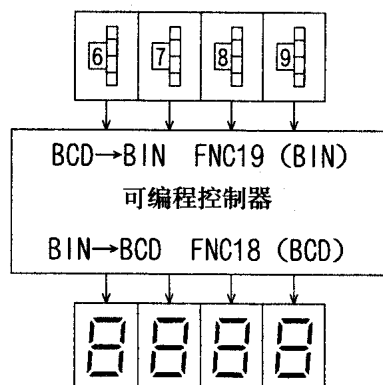
功 能 编 号	助 记 符	功能	操作软元件		D	P
			S	D		
18	BC D	将源操作软元件的二进制数据转换成 BCD 码传送到指定的目标操作元件中	KnX 、 KnY 、 KnM、 KnS、 T、 C、 D、 V、 Z	KnY、 KnM、 KnS、 T、 C、 D、 V、 Z	+	+
19	BIN	将源操作元件的 BCD 码转换成二进制数据传送到指定的目标操作元件中	KnX 、 KnY 、 KnM、 KnS、 T、 C、 D、 V、 Z	KnY、 KnM、 KnS、 T、 C、 D、 V、 Z	+	+

程序举例 1：



源 (BIN) → 目标 (BCD) 的转换传送指令。

接线形式如下图：



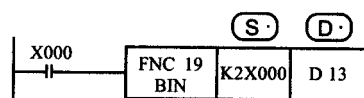
四则运算 (+ - × ÷) 与增量指令、减量指令等编程控制器内的运算都用 BIN 码进行。因此可编程控制器获取 BCD 的数字开关信息时要使用 FNC19 (BCD → BIN) 转换传送指令。另外向 BCD 的七段显示器输出时请使用 FNC18 (BIN → BCD) 转换传送指令。但是 一些特殊指

令能自动地进行 BCD / BIN 转换。

指令说明 1:

- 1) 使用 BCD、BCD (P) 指令时, 如 BCD 转换结果超出 0 — 9999 范围会出错。
- 2) 当使用 (D) BCD 、 (D) BCDP 指令时, 如 BCD 转换结果超出 0 — 99999999 范围会出错 。
- 3) 将可编程序控制器内的二进制数据变为七段显示等的 BCD 码而向外部输出时使用该指令。

程序举例 2:



数值范围 : 0-9,999 或 0-99,999,999 有效。
源 (BCD) → 目标 (BIN) 的转换传送指令

指令说明 2:

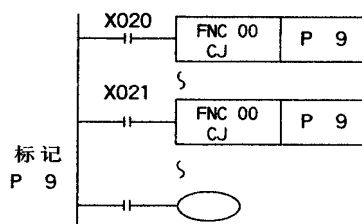
- 1) 可编程控序制器获取 BCD 数字开关的设定值时使用。
- 2) 源数据不是 BCD 码时, 会发生 M8067 (运算错误), M8068 (运算错误锁存) 将不工作。
- 3) 因为常数 K 自动地转换成二进制数, 所以不成为这个指令适用软元件。

二、程序流程控制指令

功能编号	功能助记符	指令名称及功能
00	CJ	条件跳转，程序跳到P指针指定处，P63为END
01	CALL	子程序调用，指定P指针，可嵌套5层以下
02	SRET	子程序返回，从子程序返回，与CALL配对
06	FEND	主程序结束
07	WDT	定时器刷新
08	FOR	重复循环开始，可嵌套5层
09	NEXT	重复循环结束

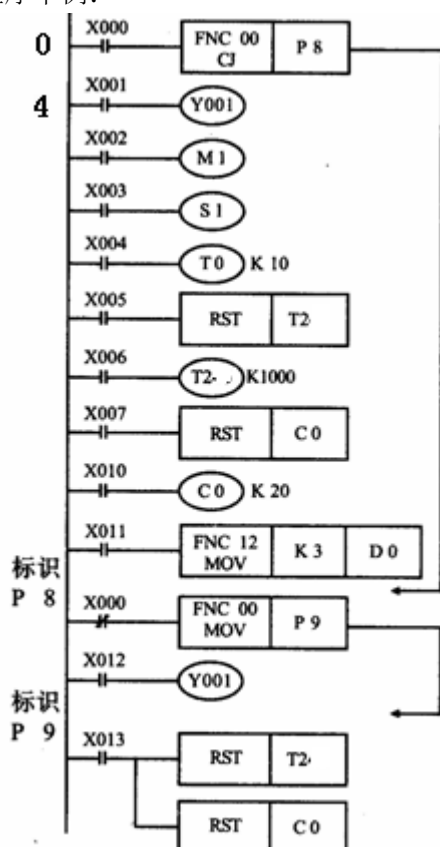
1. 条件跳转指令 CJ

格式：



操作软件：P0～P63 共 64 个。

程序举例：



跳转过程中发生变化会产生这样的结果！

类型	跳转前的接点状态	跳转中的接点动作	跳转中的线圈动作
Y,M,S	X001,X002, X003 OFF	X001,X002, X003 ON	Y001,M1, S1 OFF
	X001,X002, X003 ON	X001,X002, X003 OFF	Y001,M1, S1 ON
10ms, 100ms 定时器	X004 OFF	X004 ON	定时器不动作
	X004 ON	X004 OFF	时钟中断, X000 "OFF" 后继续
1ms 定时器	X005 OFF	X006 ON	定时器不动作, 时钟继续运行,
	X005 OFF	X006 OFF	X000 "OFF" 后触点动作
计数器	X007 OFF	X010 ON	计数不工作
	X007 OFF	X010 OFF	计数中断, X000 "OFF" 后继续
应用指令	X011 OFF	X011 ON	跳转过程中不执行 FNC 指令。
	X011 ON	X011 OFF	但是, FNC52 ~58 继续工作。

作为执行序列的一部分指令，有 CJ、CJP 指令，可以缩短运算周期及使用双线圈。

在左图的示例中，如果 X0 "ON"，则从 1 步跳转到 标识 P8 的后一步。X0 = OFF

时, 不进行跳转, 从 0 步向 4 步移动, 不执行跳转指令。Y1 变成双线圈, 但是, X0 = OFF 时采用 X1。x000 = ON 时采用 X12 动作。因条件跳转, 即使是分段的程序, 在跳转内或跳转外将同一线圈编成 2 个以上程序时, 也当作一般的双线圈对待。

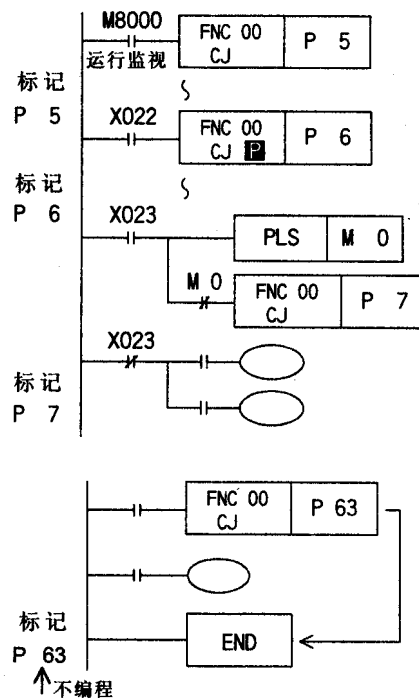
累计定时器及计数器的复位指令在跳转外时, 计时线圈及跳转的计数线圈复位(接点恢复及当前值的清除)有效。

指令说明:

1) 如举例程序中 X0=ON 时跳转到程序 P8 称为有条件转移, 而如下程序则为无条件跳转。

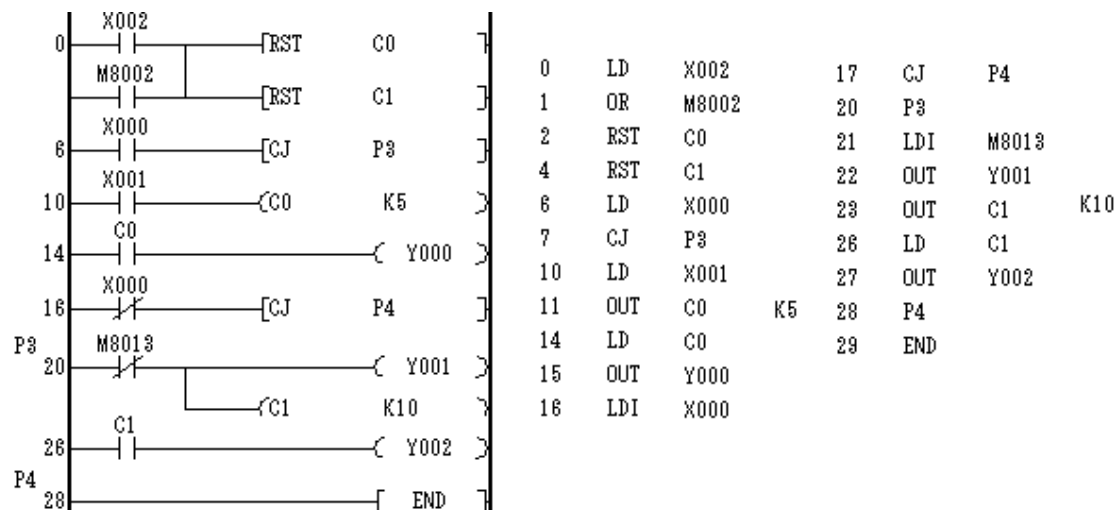
2) 一个标号只能出现一次, 多于一次则会出错; 两条或多条跳转指令可以使用同一标号, 如上格式例。

3) 如左图, 编程时标号占一行, 对有意为向 END 步跳转的指针 P63 编程时, 请不要对标记 P63 编程, 给标记 P63 编程时, 可编程控制器显示出错码 6507 (标记定义不正确) 并停止。



应用举例:

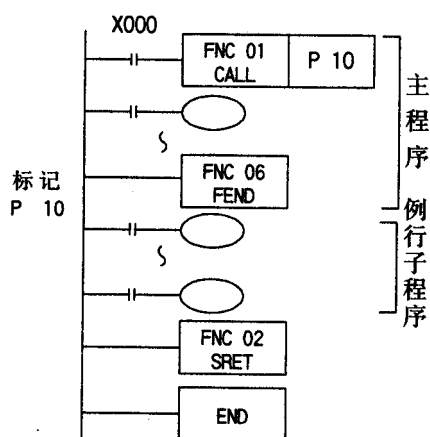
在工业控制当中经常采用手动和自动两种方式以确保生产控制的安全性和连续性。



X0=OFF手动 X0=ON自动

2. 子程序调用指令 CALL、子程序返回指令 SRET、主程序结束指令 FEND。

程序格式：



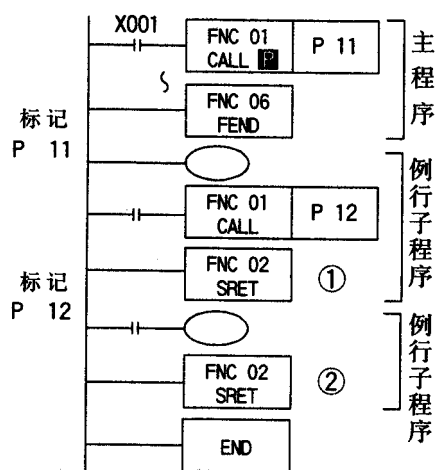
如果 X000 = ON, 则执行调用指令跳转到标记 P10 步。在这里, 执行子程序后, 通过执行 SRET 指令返回原来的步即 CALL 指令之后的步。

操作软元件：CALL P0 — P62

SRET、FEND 无操作软元件

X001=OFF 到 ON 后, 只执行 CALLP P11 指令 1 次后向标记 P11 跳转, 即脉冲形式。

在执行 P11 的子程序的过程中, 如果执行 P12 的调用指令, 则执行 P12 的子程序、用 SRET 指令向 P11 的子程序跳转。

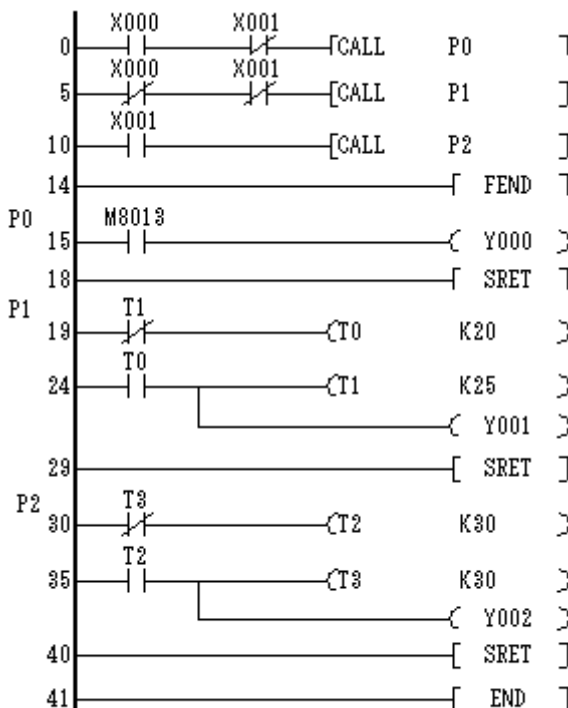


第一个 SRET 返回主程序, 第二个 SRET 返

回第一个子程序。这样, 在子程序内最多可以允许有 4 次调用指令, 整体而言可做 5 层嵌套。

应用举例：

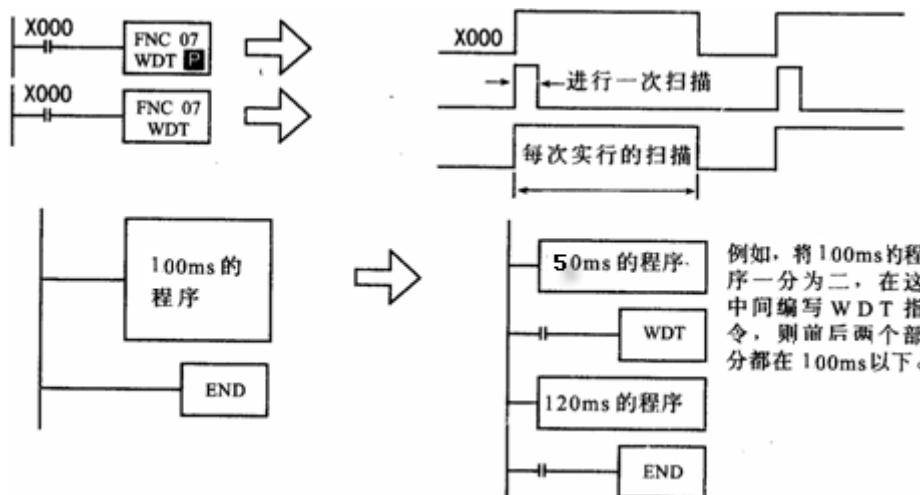
应用子程序调用指令, 可以优化程序结构, 提高编写程序的效果。如下实例：当 X1 为 OFF、X0 为 ON 时, 调用 P0 子程序；若 X0 为 OFF 时, 调用 P1 子程序；当 X1 为 ON 时, 不能调用 P0、P1 子程序, 而调用 P2 子程序。



子程序调用实例

3. 监视定时器刷新指令 WDT

WDT 指令是在 PLC 顺序执行程序中，进行监视定时器刷新的指令。WDT(P) 为连续/脉冲执行型指令。无操作软元件。

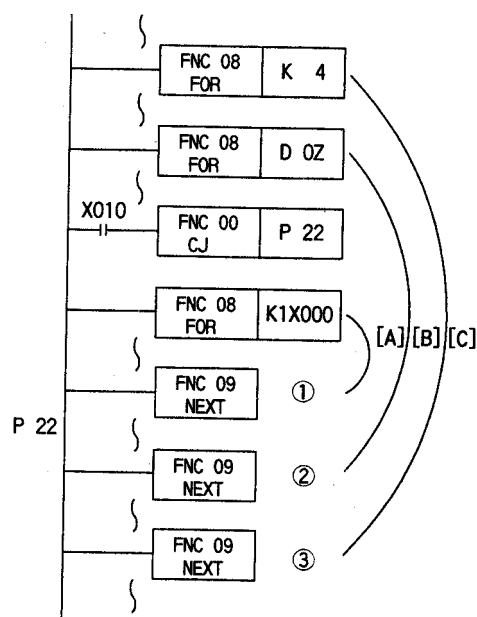


4. 循环指令 FOR、NEXT 指令说明

只在 FOR 到 NEXT 指令之间的处理（利用源数据指定的次数）执行几次后，才处理 NEXT 指令以后的步。 $n=1\sim 32,767$ 时有效，在指定了 $-32,767\sim 0$ 时，被当作 $n=1$ 处理。

[C] 的程序执行 4 次后向 NEXT 指令 (3) 以后的程序转移。

若在 [C] 的程序执行一次的过程中，数据寄存器 D02 的内容为 6，则 [B] 的程序执行 6 次。因此 [B] 的程序合计一共被执行了 24 次。



若不想执行 FOR ~NEXT 间的程序时，利用 CJ 指令，使之跳转。（ $X10 = ON$ ）当 $x10$ 为 OFF 时，例如， $K1X000$ 的内容为 7，则在 [B] 的程序执行一次的过程中，[A] 被执行了 7 次。总计被执行了 $4 \times 6 \times 7 = 168$ 次，这样一共可以嵌套 5 层。循环次数多时扫描周期会延长，有可能出现监视定时器错误，请务必注意。

NEXT 指令在 FOR 指令之前或 FOR 指令与 NEXT 或无 NEXT 指令，或在 FEND，END

指令以后有 NEXT 指令的个数不一致时等等，都会出错。

操作软元件：K、H、KnH、KnY、KnM、KnS、T、C、D、V、Z；NEXT 无操作软元件。

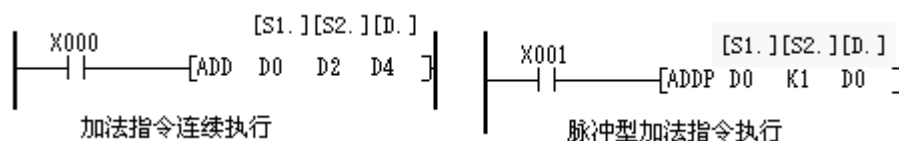
三、四则运算指令

功能编号	助记符	指令名称及功能	D	P
20	ADD	二进制加法指令	0	0
21	SUB	二进制减法指令	0	0
22	MUL	二进制乘法指令	0	0
23	DIV	二进制除法指令	0	0
24	INC	加1指令	0	0
25	DEC	减1指令	0	0

1. 加法指令

功能：加法指令时将指定的源操作软元件[S1]、[S2]中二进制数相加，结果送到指定的目标操作软元件[D]中。

格式：

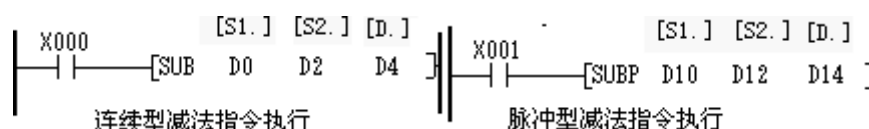


指令说明：

- 1) 操作软元件：[S] K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z
[D] KnY、KnM、KnS、T、C、D、V、Z
- 2) 当执行条件满足时，(S1)+(S2)的结果存入(D)中，运算为代数运算。
- 3) 加法指令操作时影响三个常用标志，即 M8020 零标志、M8021 借位标志、M8022 进位标志。运算结果为零则 M8020 置 1，超过 32767 进位标志 M8022 置 1，小于-32767 则借位标志 M8021 置 1。（以上都为 16 位时）

2. 减法指令

功能：减法指令是将指定的操作软元件[S1]、[S2]中的二进制数相减，结果送到指定的目标操作软元件[D]中。



格式:

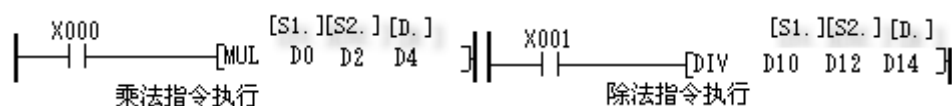
指令说明:

- 1) 操作软元件也和加法指令一样。
- 2) 当执行条件满足时, (S1)-(S2)的结果存入(D)中, 运算为代数运算。
- 3) 各种标志的动作和加法指令一样。

3. 乘法指令

功能: 乘法指令是将指定的源操作软元件[S1]、[S2]的二进制数相乘, 结果送到指定的目标操作软元件[D]中。

格式:



指令说明: 1) 操作软元件同减法指令一样。

2) $[S1] * [S2]$ 存入[D]中, 即[D0] * [D2]结果存入[D5] [D4]中。

3) 最高位为符号位, 0 正 1 负。

4. 除法指令

功能: 除法指令是将源操作软元件[S1]、[S2]中的二进制数相除, [S1]为被除数, [S2]为除数, 商送到指定的目标操作软元件[D]中。

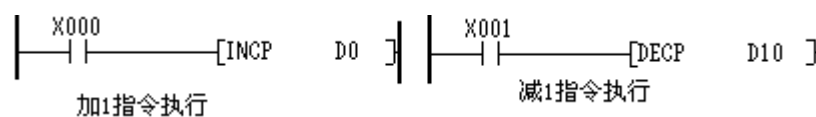
指令说明: 1) 格式如上。

2) 操作软元件通加法指令。

5. 加 1 指令/减 1 指令

功能: 目标操作软元件[D] 中的结果加 1/目标操作软元件[D]中的结果减 1。

格式:



指令说明: 1) 若用连续指令时, 每个扫描周期都执行, 须注意。

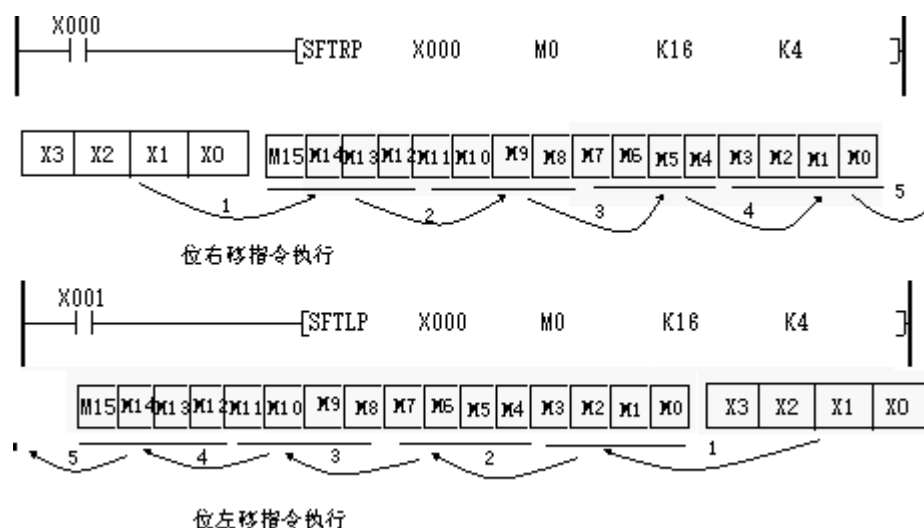
2) 脉冲执行型只在脉冲信号时执行一次。

四、移位指令

功能编号	助记符	指令名称及功能	操作软元件			
			[S.]	[D.]	n1	n2
34	SFTR(p)	位右移	X、Y、M、S	Y、M、S	K、Hn2<=n1<=1024	
35	SFTL(p)	位左移				

功能：两条指令是使位软元件中的状态向右/向左移位，n1 指定位软元件长度，n2 指定移位的位数。

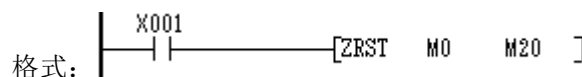
格式：



五、数据处理：批复位指令 ZRST

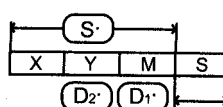
功能编号	助记符	操作软元件	
		[D1.]	[D2.]
40	ZRST	Y、M、S、T、C、D (D1 <= D2)	

功能：区间批复位。



六、状态初始化

操作软元件：

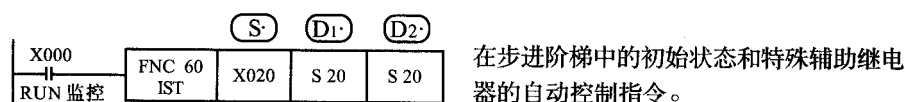


指令使用次数:1次但索引可修辞

(D1') (D2') :S20~S899

指令 (D1') < (D2')

格式：



指定区[S.]运行模式起始输入：

X020 : 手动操作 X021 : 原点回归 X022 : 单步 X23 : 循环运行一次(单周期)
X024 : 连续运行 X025 : 原点回归开始 X026 : 自动运行启动
X027 : 停止

[D1]指定自动操作模式中，实际用到状态的最小序号。

[D2] 指定自动操作模式中，实用状态的最大序号。

如果驱动该指令，下列元件被自动切换控制。但是如果驱动输入处于 OFF 状态，则不变化。

M8040 : 转移禁止 S0 : 手动操作的初始状态

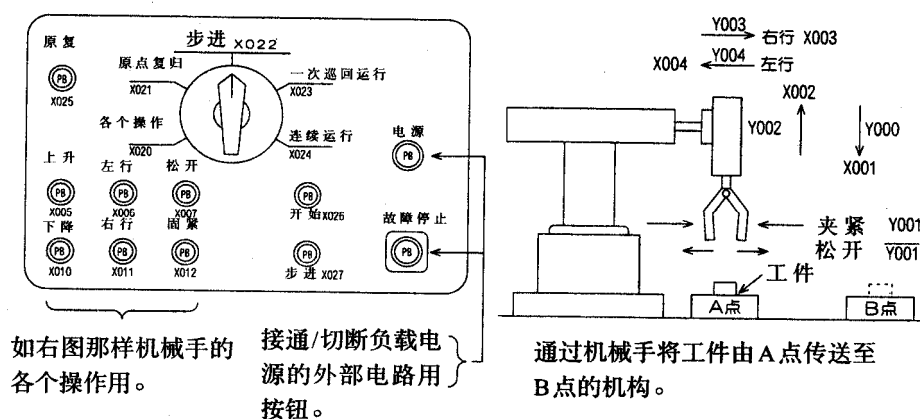
M8041 : 转移开始 S1 : 原点回归的初始状态

M8042 : 启动脉冲 S2 : 自动运行的初始状态

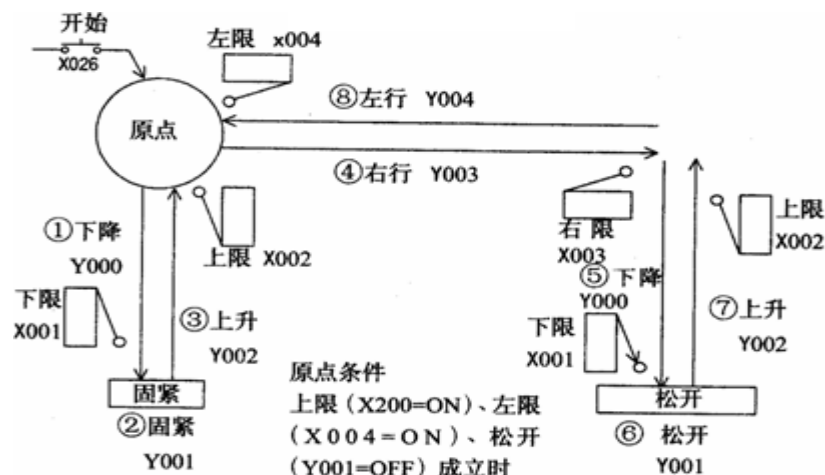
M8047 : STL 监控有效。

如使用这个指令，则 S10 — S19 可作为原点回归用。因此，在编程中请勿将这些状态作为普通状态使用。另外，S0 — S9 作为初始状态处理，S0 — S2 作为如上述的手动操作，原点回归以及自动运行使用。关于 S3 — S9 可以自由地使用。这个指令必须比状态 S0 — S2 等一系列的 STL 电路优先编程。为了防止上例的 X20 — X24 同时处于 ON 状态，必须用旋转开关。原点回归完成 (M8043) 未动作时，如果在各个 (X20)，原点回归 (X21)，自动 (X22、X23、X24) 之间进行切换时，则所有输出进入 OFF 状态。并且，自动运行在原点回归结束后，才可以再次驱动。

机械手工件传送例：

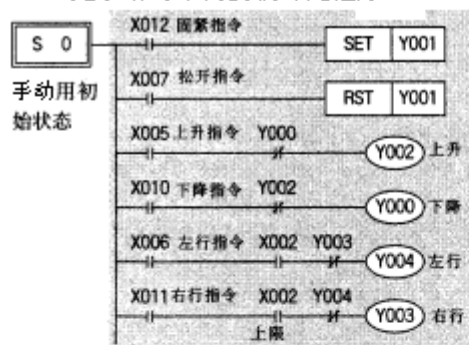


- 手动 — 各个操作 : 用单个按钮接通或切断各负载的模式。
 原点复归 : 按下原点复归按钮时, 使机械自动复归原点的模式。
- 自动 — 单步 : 每次按起动按钮, 前进一个工序。
 循环运行一次 : 在原点位置上按起动按钮时, 进行一次循环的自动运行并在原点停止。途中按停止按钮, 其工作停止, 若再按起动按钮, 在此继续动作至原点自动停止。
 连续运行 : 在原点位置上按起动按钮, 开始连续运行。若按停止按钮, 则运转至原点位置后停止。

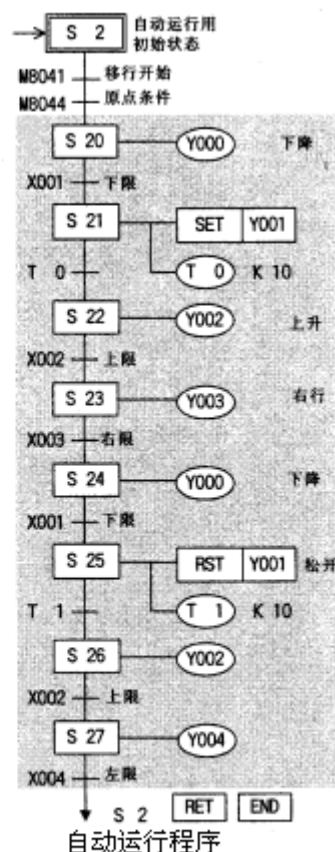
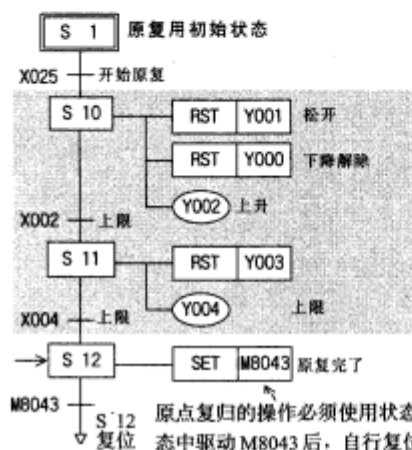


左上为原点, 按下降、夹紧、上升、右行、下降、松开、上升、左行的顺序从左向右传送。下降/上升、左行/右行使用的是双电磁阀 (驱动/非驱动2个输入), 夹紧使用的是单电磁阀 (只在通电时动作)。

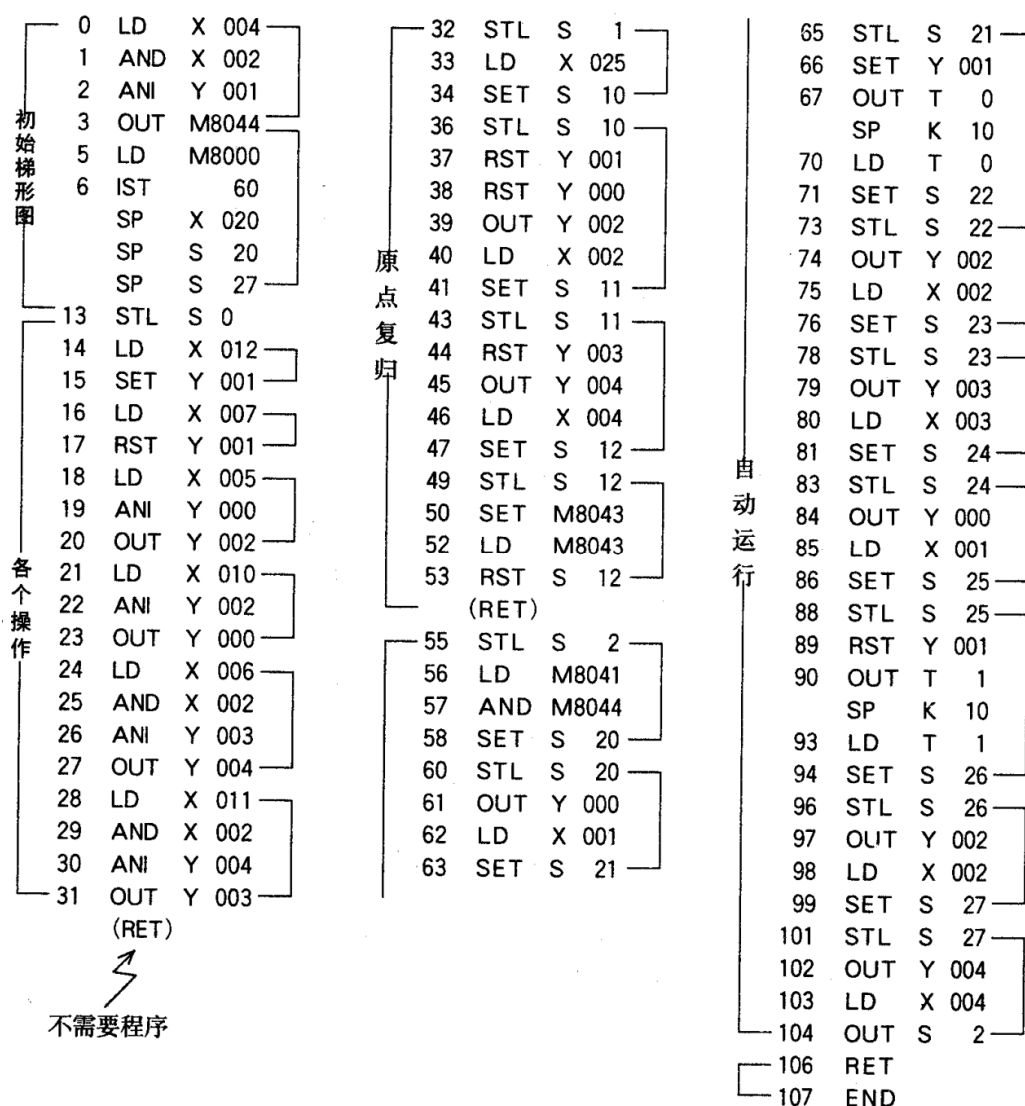
无手动时不需要编写此程序



《原点归》没有原复模式时, 不需要该程序。但在自动运行前需要置位一次 M8043。(原复完毕)。



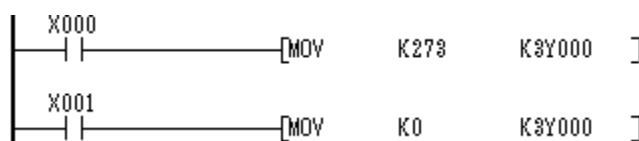
指令语句表:



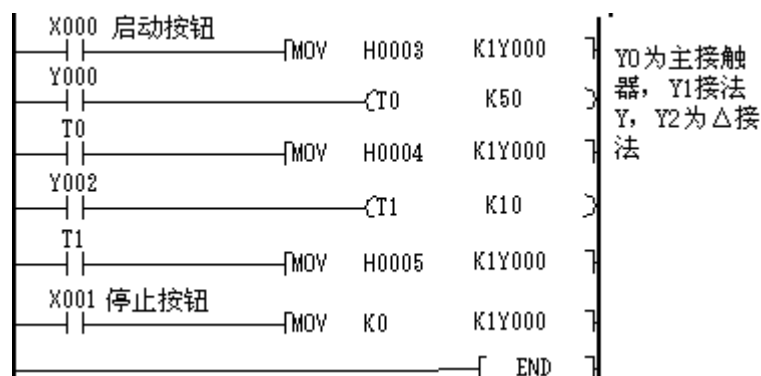
第三节 指令应用

应用指令的例程程序编写，也因为应用比较方便，在实际系统当中使用较频繁，在此举例。

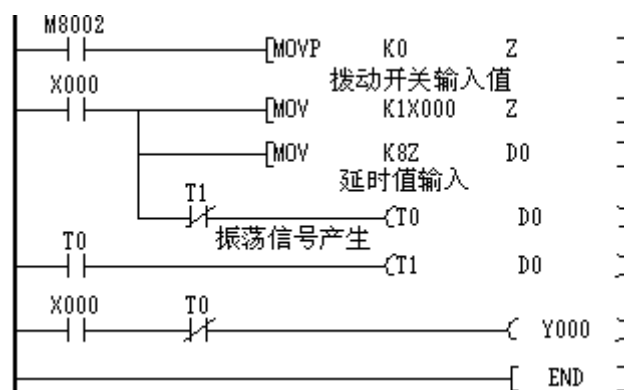
1. 应用传送指令，使分别接在 Y0、Y4、Y10 三个输出端的灯亮和熄灭。



2. 电动机的 Y/△启动控制程序



3. 多谐振荡电路



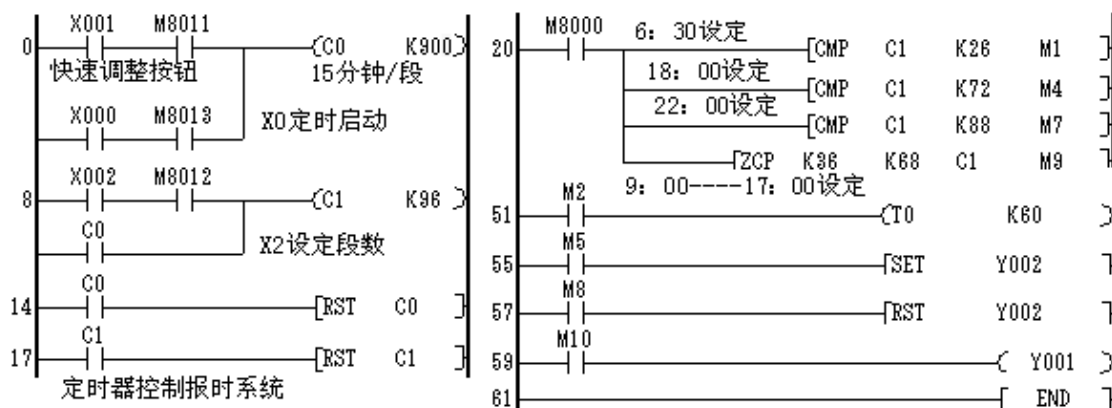
4. 定时报时器控制程序

应用计数器与比较指令构成 24 小时可设定定时时间的控制器，每 15 分钟为一设定单位，工 96 个时间单位。

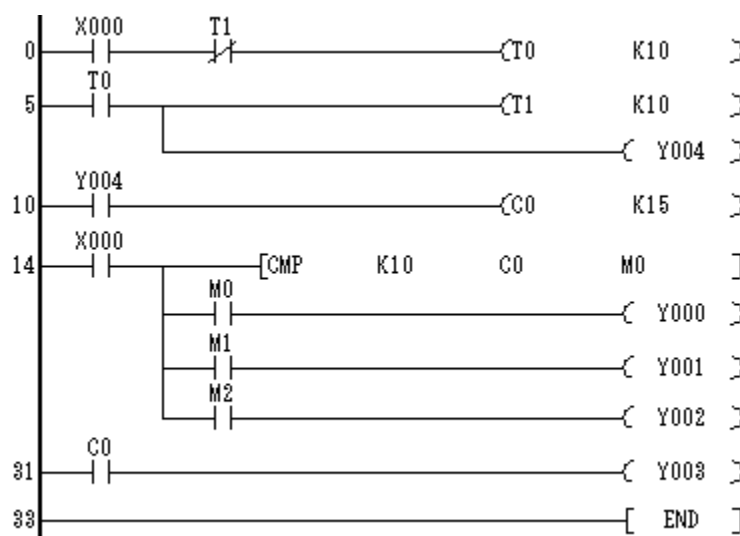
现控制实现如下：1) 6: 30 电铃 Y0 每秒响一次，6 次后自动停止；

2) 9: 00——17: 00，启动校园报警系统 Y1；3) 18: 00 开校内照明 Y2；

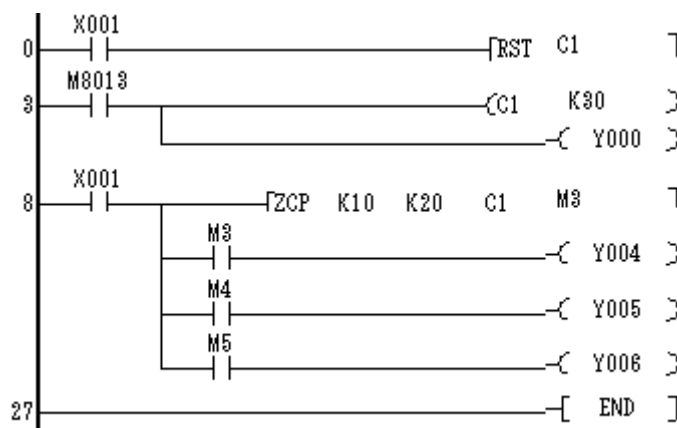
4) 22: 00 关校园内照明 Y2。



5. CMP、ZCP 应用

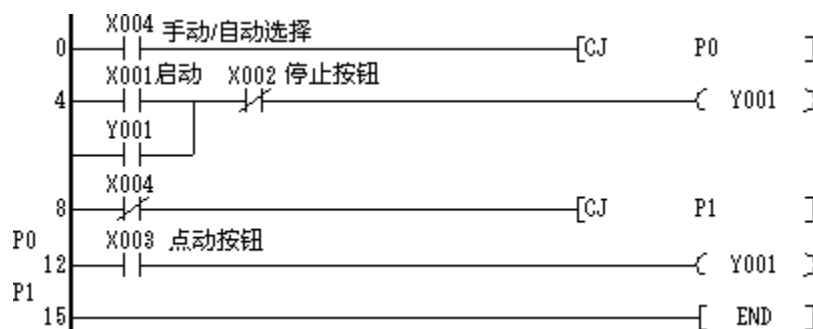


CMP指令应用



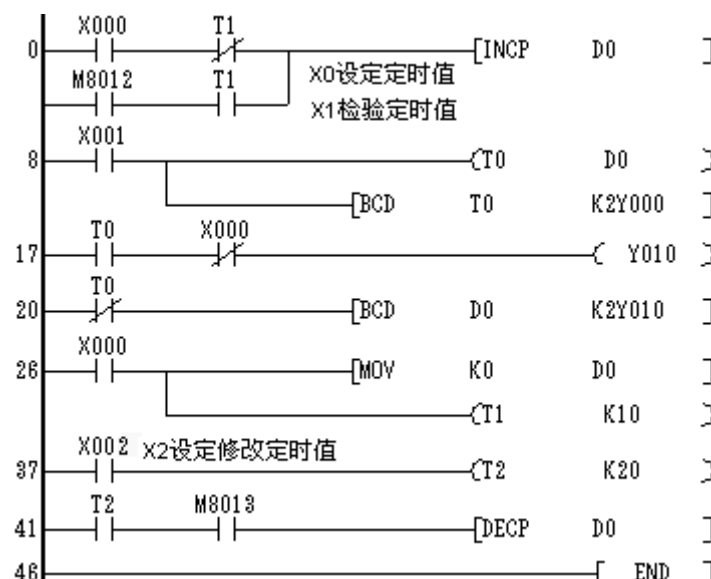
ZCP指令应用

6. 三相异步电动机技能实现连续运行有能实现点动控制。

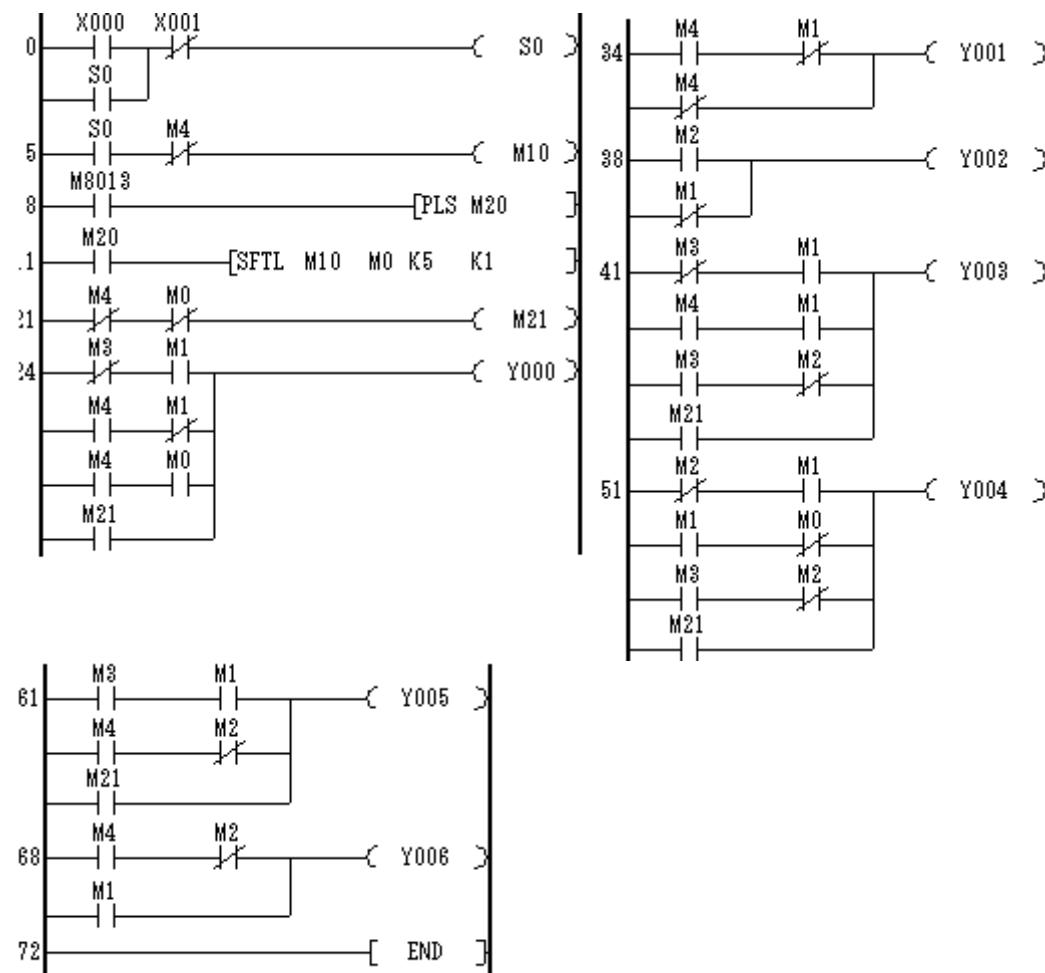


7. 用一个按钮任意改变定时器的定时值。

按下 X0 不超过 1 秒，每按一次加 1，从而改变 D0 中数据。或按下 X01 以上，每隔 0.1 秒 D0 中数据加 1，松开则不改变。按下 X2 按钮 3 秒后，每隔 1 秒 D0 数据减 1。从而可以方便地设定定时值。



8. 0——9 数码显示



第六章 PLC 控制系统设计与习题

第一节 控制系统设计概要

我们在学习了 PLC 的大量的相关知识后，要能够把其运用在实际训练当中。在这章学习 PLC 控制系统设计的基本规则、基本内容和步骤以及全书内容的相关习题。当然要设计经济、可靠、简洁的 PLC 控制系统，需要丰富的专业知识和实际的工作经验。

一、PLC 控制系统设计的基本原则

- 1) 最大限度地满足被控对象的控制要求。
- 2) 保证控制系统的高可靠、安全。
- 3) 满足上面条件的前提下，力求使控制系统简单、经济、实用和维修方便。
- 4) 选择 PLC 时，要考虑生产和工艺改进所需的余量。

二、PLC 控制系统设计的基本内容

- 1) 选择合适的用户输入设备、输出设备以及输出设备驱动的控制对象。
- 2) 分配 I/O，设计电气接线图，考虑安全措施。
- 3) 选择适合系统的 PLC。
- 4) 设计程序
- 5) 调试程序，一个是模拟调试，一个是联机调试。
- 6) 设计控制柜，编写系统交付使用的技术文件，说明书、电气图、电气元件明细表。
- 7) 验收、交付使用。

三、PLC 控制系统设计的一般步骤

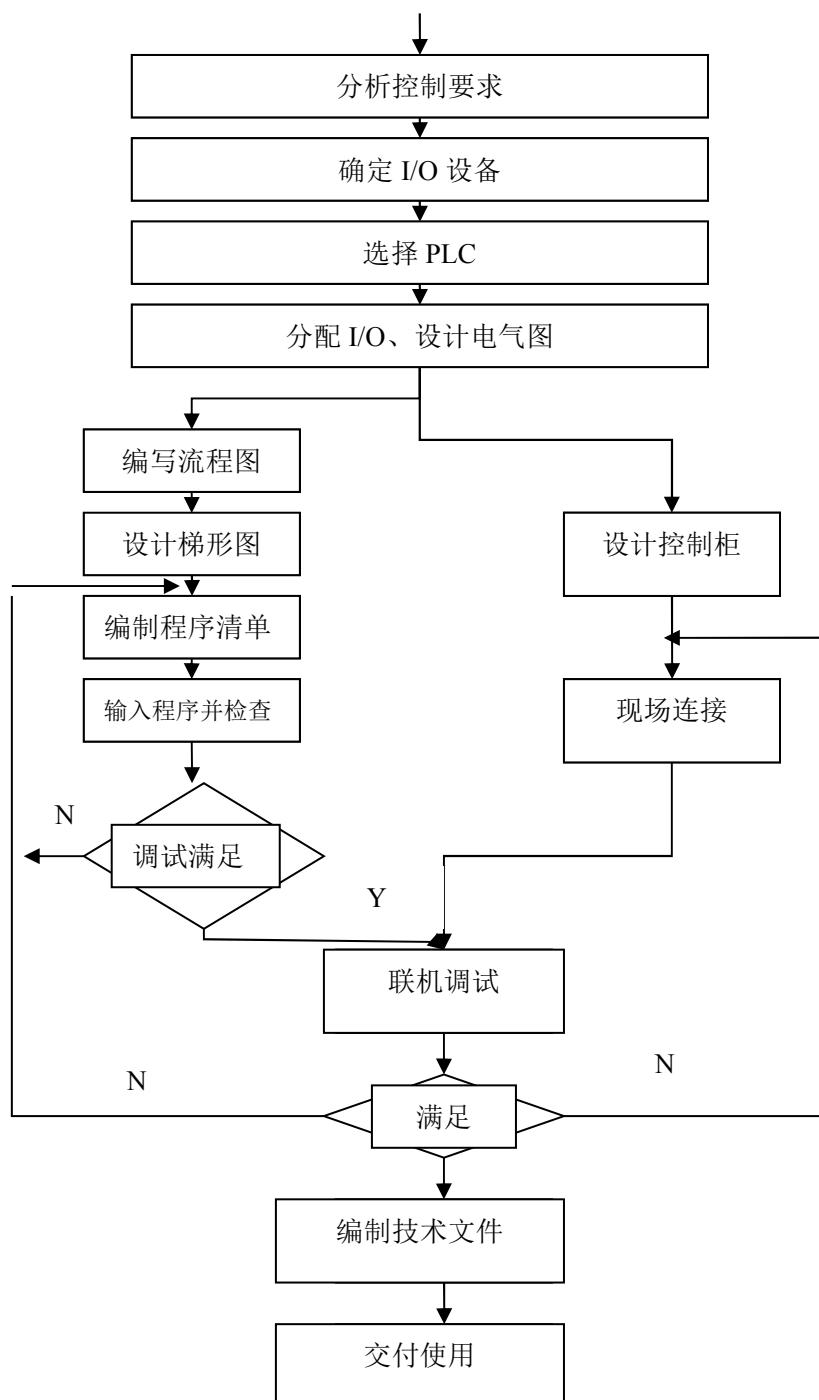
1. 流程图功能说明

- 1) 分析生产工艺过程。
- 2) 根据控制要求确定所需的用户输入、输出设备，分配 I/O。
- 3) 选择 PLC。
- 4) 设计 PLC 接线图以及电气施工图。
- 5) 程序设计和控制柜接线施工。

2. PLC 程序设计的步骤

- 1) 对于复杂的控制系统，最好绘制编程流程图，相当于设计思路。

- 2) 设计梯形图。
 - 3) 程序输入 PLC 模拟调试, 修改, 直到满足要求为止。
 - 4) 现场施工完毕后进行联机调试, 直至可靠地满足控制要求。
 - 5) 编写技术文件
 - 6) 交付使用。
3. 设计步骤框图如下:



第二节 设计题练习

利用一个章节专门作习题，可以按照指导老师的作业布置完成习题，也可以自己根据学习情况选择训练，内容不详细区分。

第一部分：基础设计练习

1. 有一个指示灯。控制要求为：按下启动按钮后，亮 2 秒灭 3 秒，重复 5 次后自动停止工作，请按设计步骤作题。
2. 有四台电动机。控制要求：按下启动按钮后，每隔 1 分钟启动一台，按下停止按钮后逆序每隔 30 秒停止，请按设计步骤作题。
3. 有一台电动机。控制要求，既可以连续运行又可以电动运行，请按设计步骤作题。
4. 设计一个报警器。控制要求，保安系统被侵入时，蜂鸣器报警，同时报警灯连续闪烁，每次亮 1 秒灭 2 秒，直至保安系统复位，请按设计步骤作题。
5. 设计一电动机过载保护系统。控制要求：电动机过载时，能自动停止运转，并发出报警信号，请按设计步骤作题。
6. 污水处理。控制要求：
 - 1) 一个污水池，有两台污水泵实现污水排放处理。两台泵每隔 5 分钟换泵循环进行；
 - 2) 当其中一台泵在工作期间出现故障时，另一台泵投入全程运行；
 - 3) 当污水液位达到超高液位时，两台泵同时投入运行；
 - 4) 污水池液位在高液位时，系统自动开启污水泵，污水池液位在低液位时系统自动关闭污水泵，污水液位达到超高液位时，系统自动开启两台泵；
 - 5) 污水出现超低液位时，液位报警灯以 0.5 秒的周期闪烁，污水池出现超高液位时，液位报警灯以 0.1 秒的周期闪烁。请按设计步骤作题。以上为自动控制，如果要手动控制如何解决。
7. 设计一个灯光控制程序。控制要求：

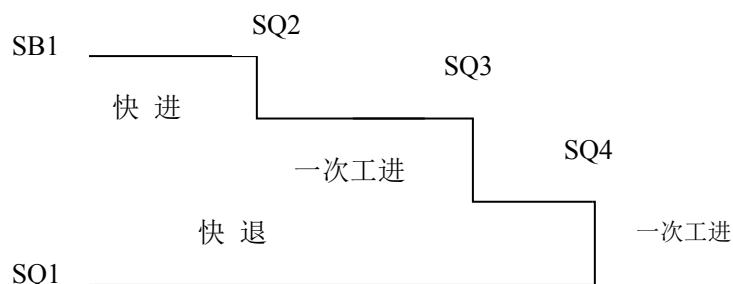
- 1) 首先 HL1 亮;
- 2) 1 秒后, HL1 熄灭, HL2 亮;
- 3) 1 秒后, HL2 熄灭, HL3 亮;
- 4) 1 秒后, HL3 熄灭;
- 5) 1 秒后, 三个灯一起亮;
- 6) 1 秒后, 三个灯一起灭;
- 7) 1 秒后, 三个灯一起亮;
- 8) 1 秒后, 三个灯一起灭;
- 9) 循环 1) ——8) 直至按下开关。

请按设计步骤作题。

8. 什么叫状态转移图? 一个完整的状态转移图包含那些内容?
9. 有四台电动机要求顺序启动逆序停止(任意时刻按下停止按钮都要逆序停止), 间隔时间自定; 急停按钮启动则全部停止。
10. 有三台电动机, 控制要求为:
 - 1) 按下启动按钮, M1 启动; 10 分钟后, M2 自行启动; 10 分钟后, M3 自行启动。
 - 2) 按下停止按钮, M3 停止运转; 8 分钟后, M2 自行停止运转; 6 分钟后 M1 自行停止运转。

请按设计步骤作题。

11. 设计一个液压动力台二次工进的控制程序。如下图:



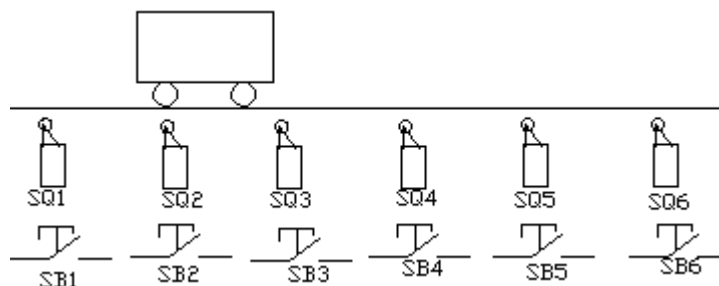
电磁阀动作表:

动作/机构	YV1	YV2	YV3	YV4
快进	+	—	+	—
一次工进	+	—	—	—
二次工进	+	—	—	+
快退	—	+	—	—
停止	—	—	—	—

请按设计步骤作题。

12. 脉冲指令和连续指令的区别。
13. 什么叫字软元件？什么叫位软元件？如何表示。
14. 说明下列所给位软元件有哪些个元件组合，表示多少位数据。
K1X0 K2Y4 K4M10 K4S20 K5X0 K6Y10 K7M0
15. 说明变址寄存器 V 和 Z 的作用。当 V=10 时，说明以下组合的含义。
K20V D5V Y4V K4X5V K4Y0V
16. 设计一个程序。控制要求：按下启动开关，依次将计数器 C0-----C9 的当前值转换成 BCD 码后，传送到输出元件 K2Y0。
17. 设计程序，对 X0 输入的脉冲信号计数，当累计到 50 个脉冲时，使输出 Y0 接通，然后继续计数 50 次后，使输出 Y0 断开。
18. 小车的控制要求如下：
 - 1) 当小车所停位置 SQ 的编号大于呼叫的 SB 的编号时，小车往左运行至呼叫的 SB 位置后停下。
 - 2) 当小车所停位置 SQ 的编号小于呼叫的 SB 的编号时，小车往右运行至呼叫的 SB 位置后停下。
 - 3) 当小车所停位置 SQ 的编号等于呼叫的 SB 编号时，小车不动。

小车运行的示意图如下,请按设计要求，遵循设计步骤作题。



第二部分：综合设计练习(给出一种思考题目,可根据实际情况选择讲学和自行设计模型)

机械手操作控制装置

一、课题内容：

用 PLC 对机械手（实物模型）操作进行控制，设机械手起始点位于乙地，工作时可将工件从甲地取起，将工件翻转后放至丙地，周而复始完成每个工件的翻转，机械手的动作步骤如下：

乙地→甲地（左转）→抓取工件→丙地（右转）→放置工件→返回乙地（左转）

输入输出点的分配：

X1——左转限位 E1 Y1——左转 M1
 X2——右转限位 E2 Y2——右转 M1，
 X3——抓手开放度限位 E3 Y3——抓取 M2
 X4——抓手抓取计数 E4 Y4——放开 M2，

二、课题要求：

1. 根据题意，画出 PLC 的硬件连接图，按图接线，编制控制程序，并设计程序。
2. 完成系统调试，实现控制要求。
3. 完成课程设计说明书

三、答辩问题：

1. 利用 PLC 软件编程后，向 PLC 传送，但 PLC 没有响应，一般是何原因？试举出 2~3 例可能的原因。
2. 可编程序控制器的控制程序为串行工作方式，继电器接触器控制线路为并行工作方式，相比之下，可编程序控制器的控制结果有什么特殊性？
3. 动作之间的顺序转换，若硬件上没有限位功能，可采用哪些编程指令实现控制功能？

工业铲车操作控制

一、课题内容：

用 PLC 对工业铲车操作进行控制，设铲车可将货物铲起或放下，并能作前进、后退、左转、右转的操作，要求动作过程如下：

铲起 → 向前 0.5 米 → 左转 90 度后向前 0.5 米 → 右转 90 度后向前 0.5 米 → 右转 90 度后后退 0.5 米 → 放下。

二、课题要求：

1. 根据题意，画出 PLC 的硬件连接图，按图接线，编制控制程序，并设计程序。
2. 完成系统调试，实现控制要求。
3. 完成课程设计说明书。

三、答辩问题：

为什么 PLC 具有很强的抗干扰能力？试例举软件和硬件方面的措施予以说明。

车辆出入库管理

一、课题内容：

编制一个用 PLC 控制的车辆出入库管理梯形图控制程序，控制要求如下：

1. 入库车辆前进时，经过 1# → 2#传感器后计数器加 1，后退时经过 2# → 1#传感器后计数器减 1，单经过一个传感器则计数器不动作。
2. 出库车辆前进时经过 2# → 1#传感器后计数器减 1，后退时经过 1# → 2#传感器后计数器加 1，单经过一个传感器则计数器不动作。
3. 设计一个由两位数码管及相应的辅助元件组成的显示电路，显示车库内车辆的实际数量。

X1—启动 X6—1#传感器 X2—停止 X7—2#传感器

X3—清零 Y1—仓库空显示 Y4— = “1” 入库操作

X4 = “0” 出库操作 X5— = “1” 前进操作

X5— = “0” 后退操作

二、课题要求：

1. 根据题意设计显示电路图，并按图连接。
2. 画出 PLC I/O 端口接线图，并按图接线。

3. 编制控制程序，并画出梯形逻辑图。

4. 完成系统调试，实现控制要求。

5. 完成课程设计说明书。

三、答辩问题：

1. PLC 中的“软继电器”与实际继电器相比，有哪些特点？

2. PLC 的扫描周期包括哪些时间？其长短主要受哪些因素影响？

自动门控制装置

一、 课题内容：

1. 自动门控制装置的硬件组成：

自动门控制装置由门内光电探测开关 K1、门外光电探测开关 K2、开门到位限位开关 K3、关门到限位开关 K4、开门执行机构 KM1（使直流电动机正转）、关门执行机构 KM2（使直流电动机反转）等部件组成。

2. 控制要求：

- 1) 当有人由内到外或由外到内通过光电检测开关 K1 或 K2 时，开门执行机构 KM1 动作，电动机正转，到达开门限位开关 K3 位置时，电机停止运行。
- 2) 自动门在开门位置停留 8 秒后，自动进入关门过程，关门执行机构 KM2 被起动，电动机反转，当门移动到关门限位开关 K4 位置时，电机停止运行。
- 3) 在关门过程中，当有人员由外到内或由内到外通过光电检测开关 K2 或 K1 时，应立即停止关门，并自动进入开门程序。
- 4) 在门打开后的 8 秒等待时间内，若有人员由外至内或由内至外通过光电检测开关 K2 或 K1 时，必须重新开始等待 8 秒后，再自动进入关门过程，以保证人员安全通过。

三、答辩问题：

1. 若开门等待时间改为 10 秒，梯形图作何改动？

2. 如光电检测开关改为有光断开、无光导通，则梯形图作何修改？

3. 试分析开门与关门为何不能同时动作？假如开、关门时同时动作会出现什么现象？

4. 为防止开、关门同时动作，试在电路中设计一个保护环节。

全自动洗衣机 PLC 控制

一、课题内容:

全自动洗衣机运行框图及梯形图控制程序的编制,并画出硬件接线图。

二、控制要求:

- (1) 按下启动按钮及水位选择开关,开始进水直到高(中、低)水位,关水
- (2) 2 秒后开始洗涤
- (3) 洗涤时,正转 30 秒,停 2 秒,然后反转 30 秒,停 2 秒
- (4) 如此循环 5 次,总共 320 秒后开始排水,排空后脱水 30 秒
- (5) 开始清洗,重复(1)~(4),清洗两遍
- (6) 清洗完成,报警 3 秒并自动停机
- (7) 若按下停车按钮,可手动排水(不脱水)和手动脱水(不计数)

I/O 分配:

输入: 启动 X1 低水位检测 X11 停止 X2 手动排水 X12 高水位 X3
手动脱水 X13 中水位 X4 低水位 X5 排空检测 X6 高水位检测 X7 中
水位检测 X10

输出: 启动洗衣机 Y1 进水阀 Y2 正转及脱水 Y3 反转 Y4 排水 Y5 报警 Y6

饮料罐装生产流水线的 PLC 控制

一、课题内容:

饮料罐装生产流水线梯形图控制程序设计并画出硬件接线图

二、控制要求:

- (1) 系统通过开关设定为自动操作模式,一旦启动,则传送带的驱动电机启动并一直保持到停止开关动作或罐装设备下的传感器检测到一个瓶子时停止;瓶子装满饮料后,传送带驱动电机必须自动启动,并保持到又检测到一个瓶子或停止开关动作。
- (2) 当瓶子定位在罐装设备下时,停顿 1 秒,罐装设备开始工作,罐装过程为 5 秒钟,罐装过程应有报警显示,5 秒后停止并不再显示报警
- (3) 用两个传感器和若干个加法器检测并记录空瓶数和满瓶数,一旦系统启动,必须记录空瓶数和满瓶数,设最多不超过 32767 瓶
- (4) 可以手动对计数值清零(复位)

三、 课题要求：

1. 根据课题的控制要求完成设计并调试成功
2. 完成课程设计说明书

八层电梯楼层显示

控制要求：

- 1 要求用行程开关为电梯作楼层定位
- 2 用 9 段数码管作十进制数显示。9 段数码管显示要求如下：



自动售货机

自动售货机采用 P L C 控制要求如下：

- 1) 售货机可投入 1 角、 5 角、 1 元硬币。
- 2) 当投入的硬币总值超过 2 元时，汽水按钮指示灯亮；当投入的总值超过 3 元时，汽水及可乐按钮指示灯亮；
- 3) 当汽水按钮指示灯亮时，按汽水按钮，则汽水排出， 8 秒后，自动停止；这段时间内汽水指示灯以 0 . 2 秒闪烁；
- 4) 当可乐按钮指示灯亮时，按可乐按钮，则可乐排出， 8 秒后，自动停止，这段时间内可乐指示灯以 0 . 2 秒闪烁；
- 5) 当投入硬币的总值超过所选货物所需的钱数时，找钱指示灯亮，按下退钱按钮后停止。

请根据控制要求设计系统程序。