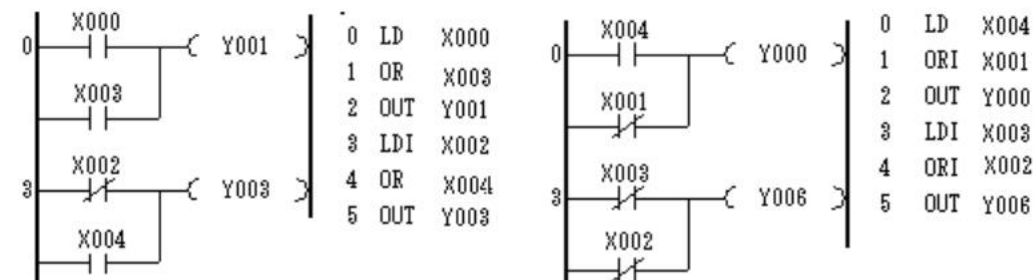


三、OR、ORI 指令

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

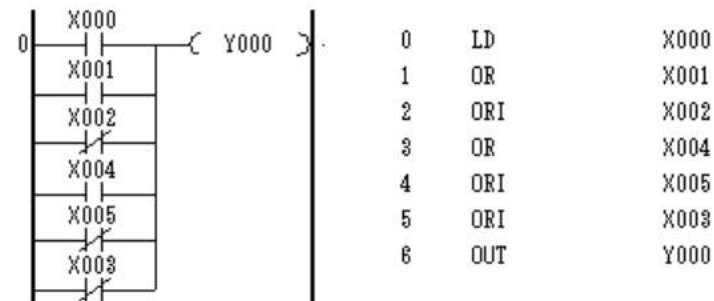
1. 程序举例：



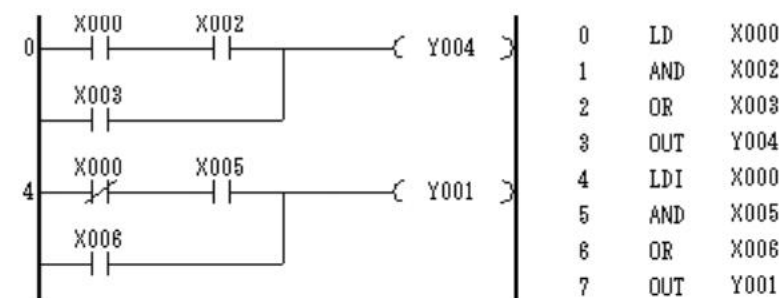
2. 例题解释：1) 当 X0 或 X3 接通时 Y1 接通；
 2) 当 X2 断开或 X4 接通时 Y3 接通；
 3) 当 X4 接通或 X1 断开时 Y0 接通；
 4) 当 X3 或 X2 断开时 Y6 接通。

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；

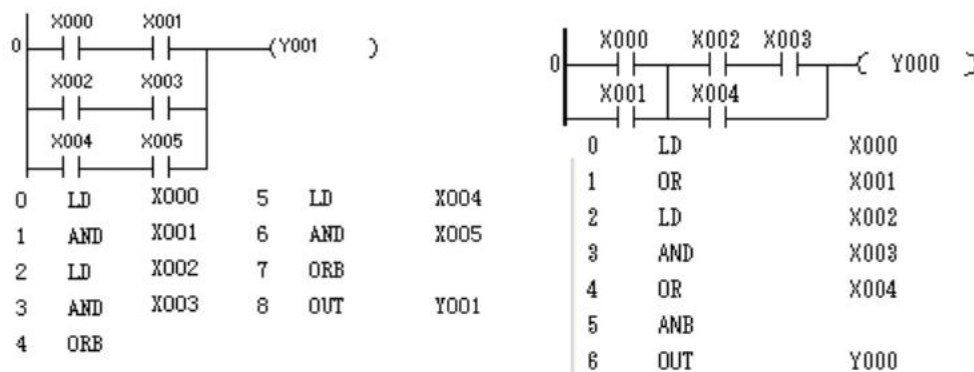


- 3) OR、ORI 指令是从该指令的步开始，与前面的 LD、LDI 指令步进行并联连接。
 4) 当继电器的常开触点或常闭触点与其他继电器的触点组成的混联电路块并联时，也可以用这两个指令。

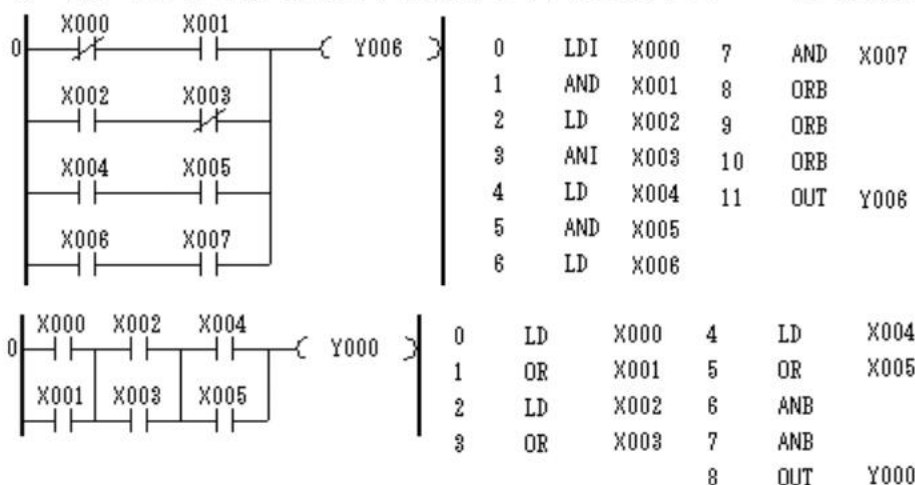


四、串联电路块并联指令 ORB、并联电路块串联指令 ANB

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。

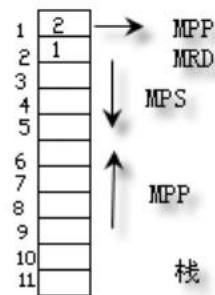
五、分支多重输出 MPS、MRD、MPP 指令

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

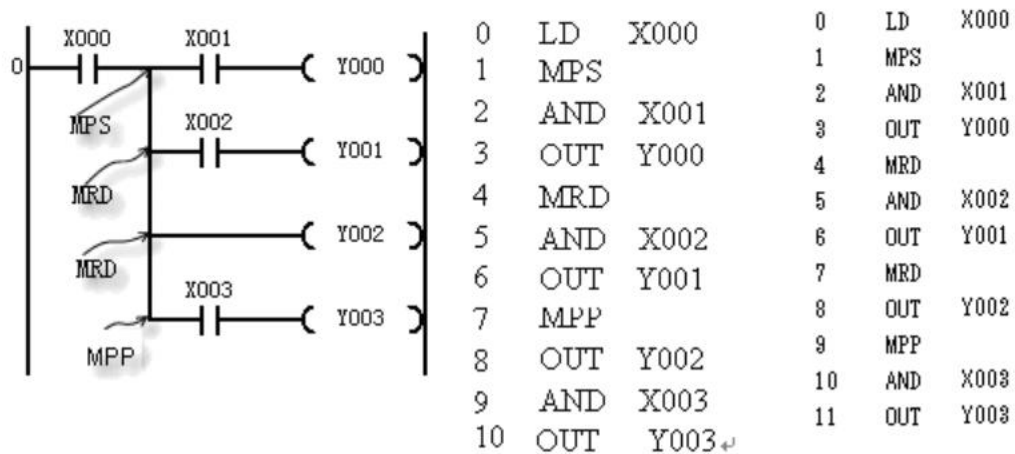


层存储的数据移到堆栈的下一段。

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：

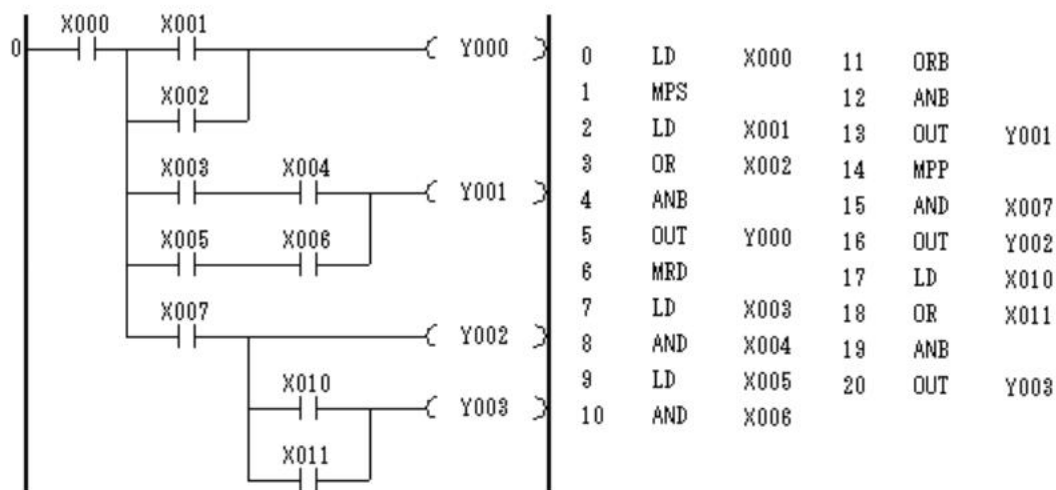


2. 例题解释：1) 当公共条件 X0 闭合时，X1 闭合则 Y0 接通；X2 接通则 Y1 接通；Y2 接通；X3 接通则 Y3 接通。

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：

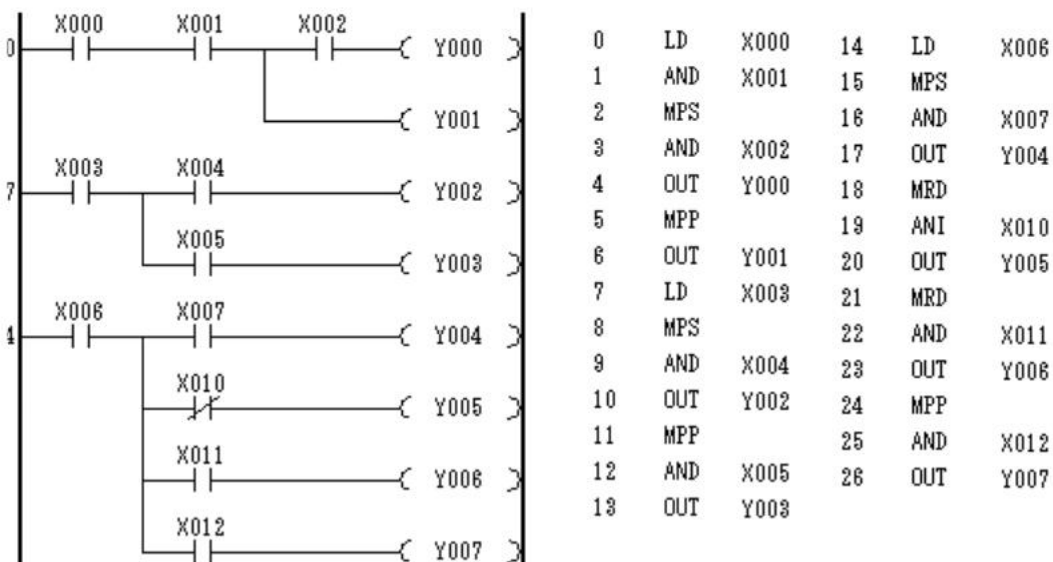
- 1) **MPS**、**MRD**、**MPP** 无操作软元件
- 2) **MPS**、**MPP** 指令可以重复使用，但是连续使用不能超过 11 次，且两者必须成对使用缺一不可，**MRD** 指令有时可以不用；
- 3) **MRD** 指令可多次使用，但在打印等方面有 24 行限制；
- 4) 最终输出电路以 **MPP** 代替 **MRD** 指令，读出存储并复位清零；
- 5) **MPS**、**MRD**、**MPP** 指令之后若有单个常开或常闭触点串联，则应该使用 **AND** 或 **ANI** 指令；
- 6) **MPS**、**MRD**、**MPP**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



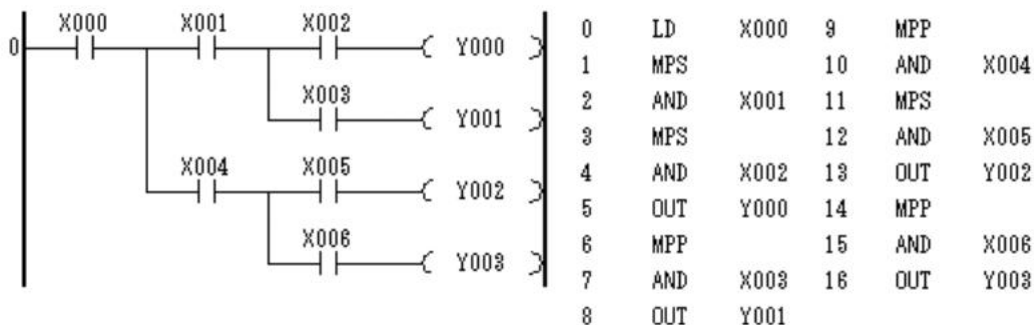
7) MPS、MRD、MPP 指令之后若无触点串联，直接驱动线圈，则应该使用 OUT 指令；

8) 指令使用可以有层堆栈。

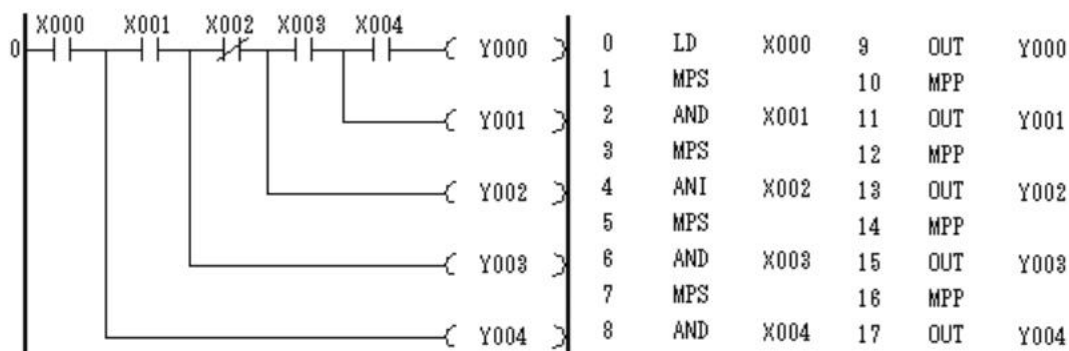
编程例一，一层堆栈：



编程例二，两层堆栈：



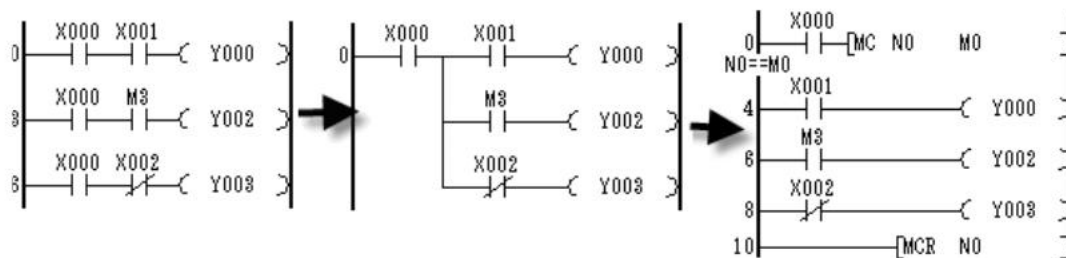
编程例三，四层堆栈：



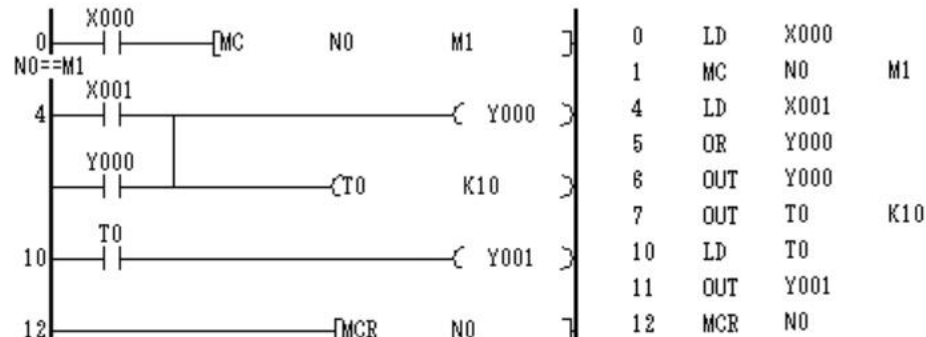
上面编程例三可以使用纵接输出的形式就可以不采用 MPS 指令了，请授课人员补充。

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：

1) MC 指令的操作软元件 N、M

2) 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。

3) 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。

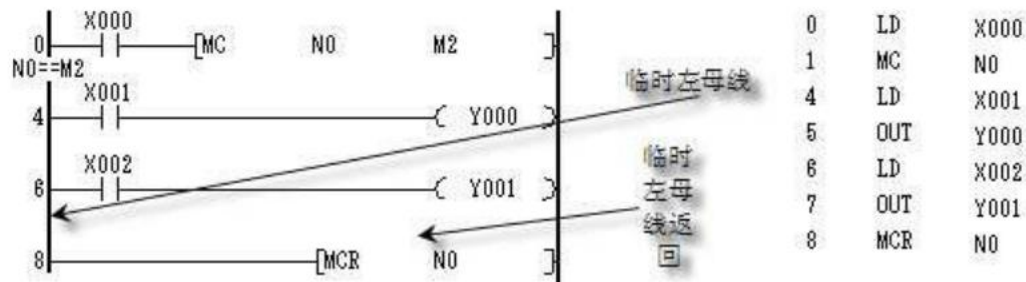
4) MC 指令的操作元件可以是继电器 Y 或辅助继电器 M（特殊继电器除外）；

5) MC 指令后，必须用 MCR 指令使临时左母线返回原来位置；

6) 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

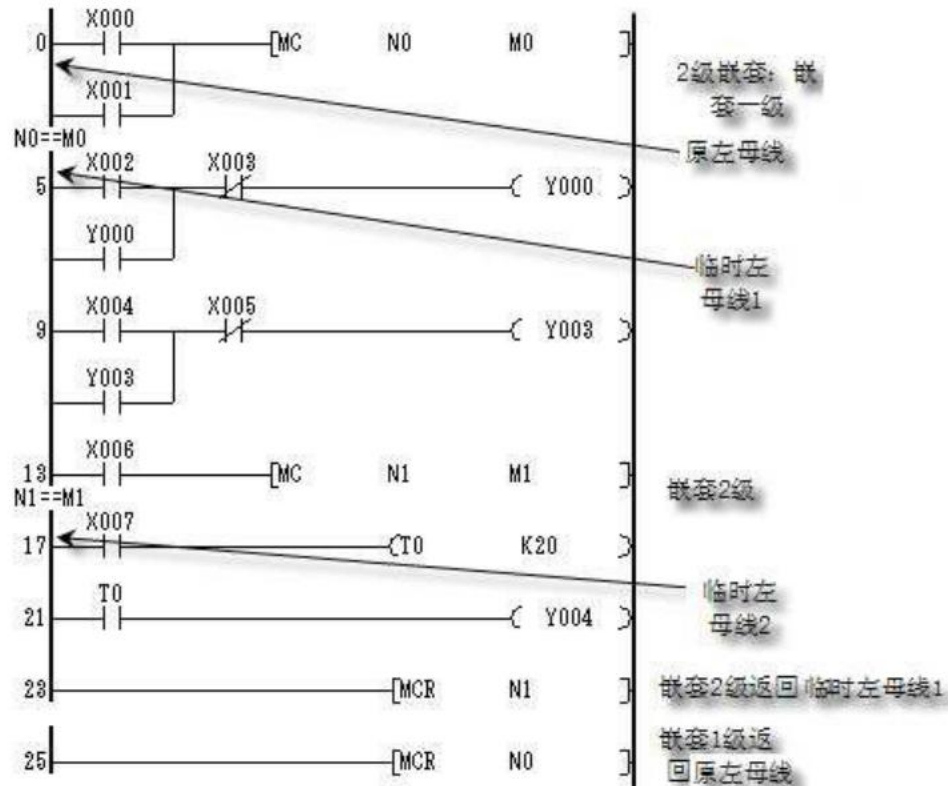
编号从 N0 到 N7 安顺序增加，顺序不能颠倒；而主控返回则嵌套级标号必须从大到小，即按 N7 到 N0 的顺序返回，不能颠倒，最后一定是 MCR N0 指令；

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：

5) 积算定时器 T63 的当前值复 0 和触点复位也可用 RST。

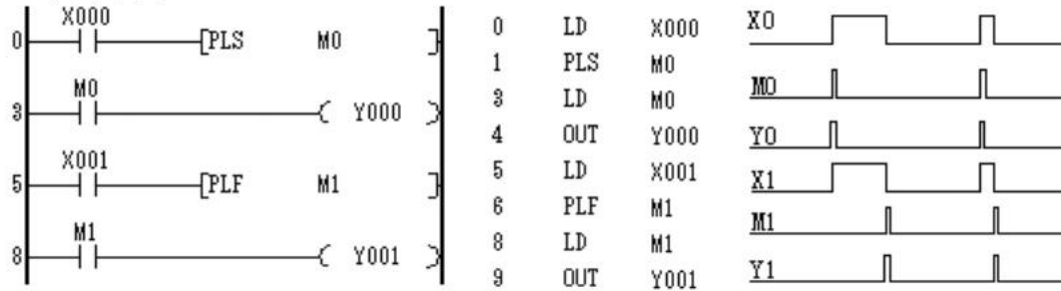
八、上升沿微分脉冲指令 PLS、下降沿微分脉冲指令 PLF

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：

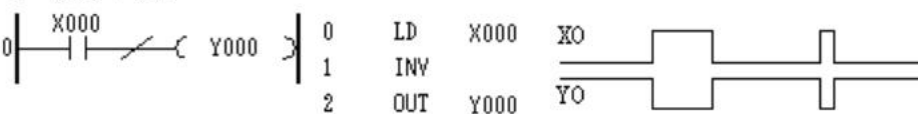
- 1) PLS 指令驱动的软元件只在逻辑输入结果由 OFF 到 ON 时动作一个扫描周期；
- 2) PLF 指令驱动的软元件只在逻辑输入结果由 ON 到 OFF 时动作一个扫描周期；
- 3) 特殊辅助继电器不能作为 PLS、PLF 的操作软元件。

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

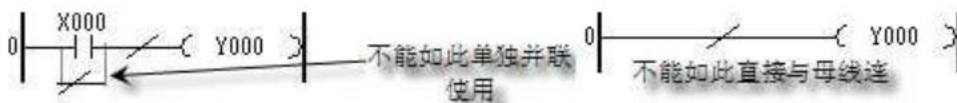
1. 程序举例：



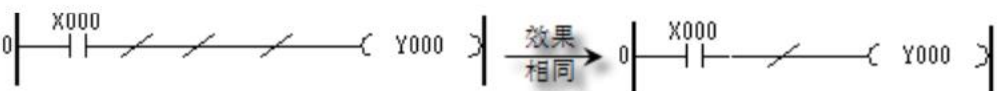
2. 例题解释：X0 接通，Y0 断开；X0 断开，Y0 接通。

3. 指令说明：

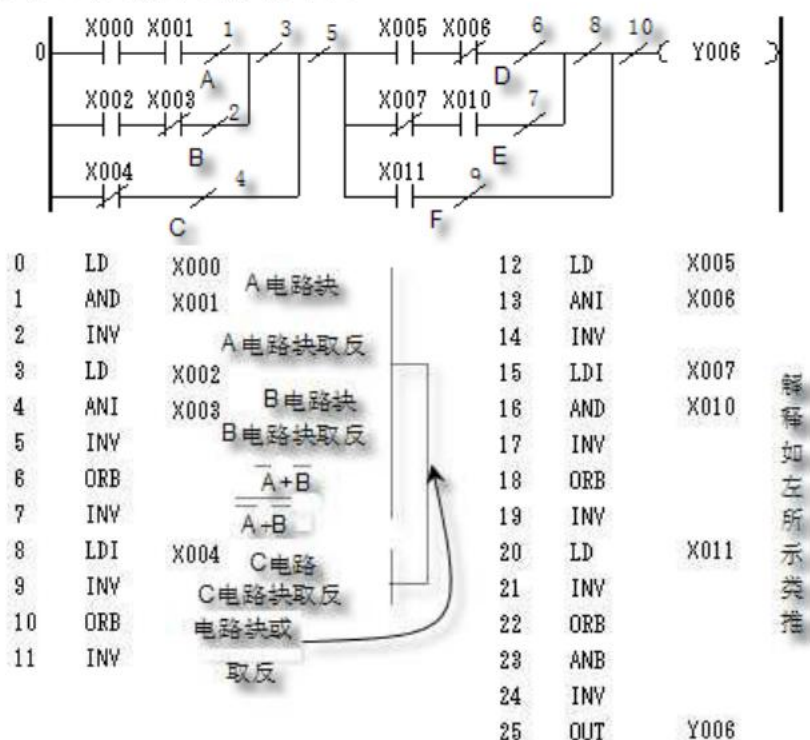
- 1) 编写 INV 取反指令需要前面有输入量，INV 指令不能直接与母线相连接，也不能如 OR、ORI、ORP、ORF 单独并联使用；



- 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)

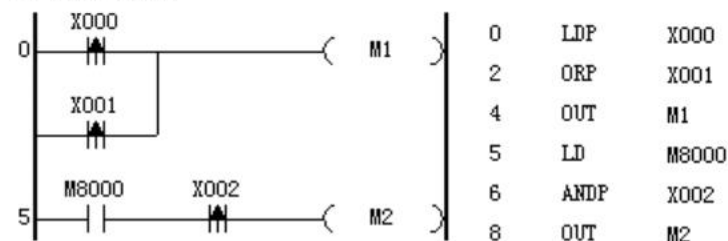
ANDF：下降沿检测串联连接(检测到位软元件下降沿信号时闭合一个扫描周期)

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)

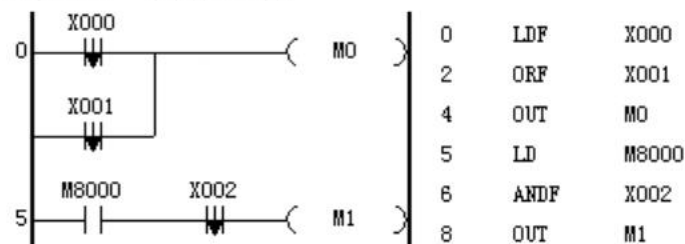
ORF: 脉冲下降沿检测并联连接(检测到位软元件下降沿信号时闭合一个扫描周期)

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。

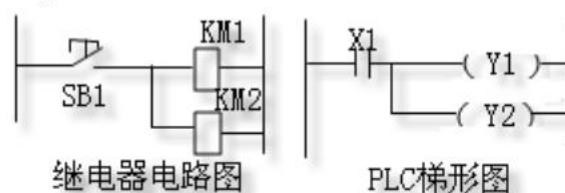
第三节 PLC 的编程及应用

一、 PLC 编程特点

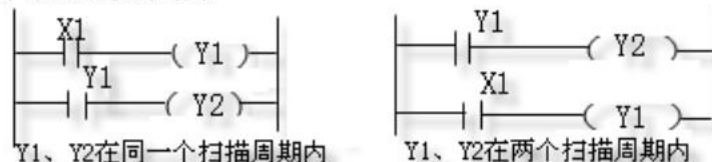
梯形图是 PLC 中最常用的方法,它源于传统的继电器电路图,但发展到今天两者之间已经有了极大的差别。

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

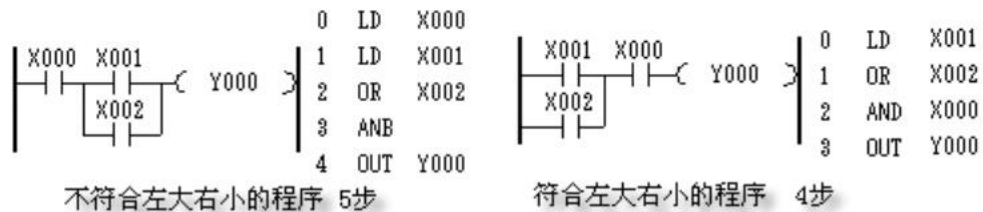
一) PLC 编程应遵循以下基本规则:

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. 可编程序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。

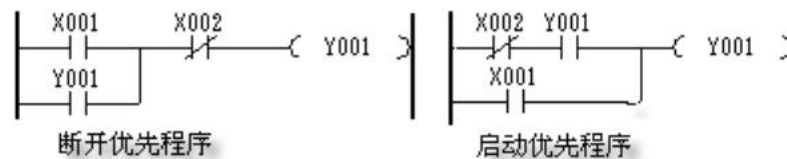


2. 依照扫描的原则，程序处理时尽可能让同时动作的线圈在同一个扫描周期内。

三、典型控制程序

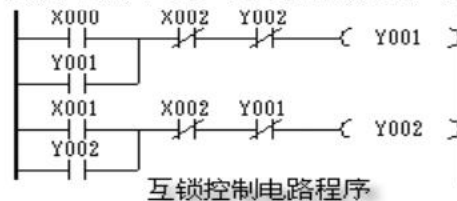
1. 自保持程序

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

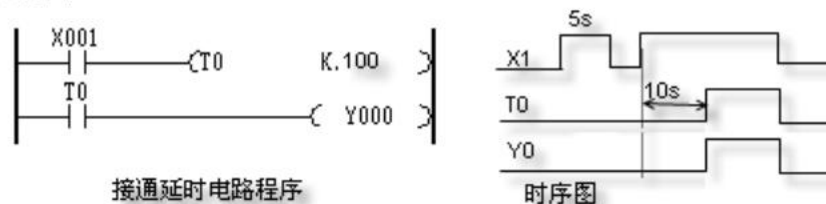
互锁电路用于不允许同时动作的两个或多个继电器的控制，如电动机的正反转控制。



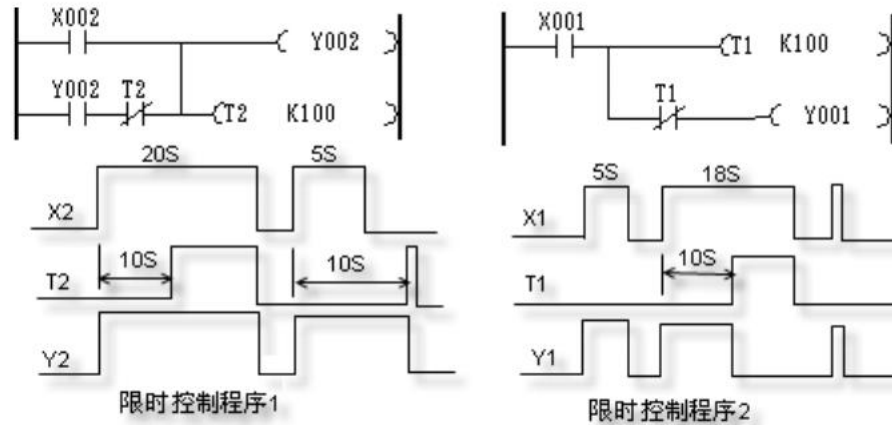
3. 时间电路程序

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。

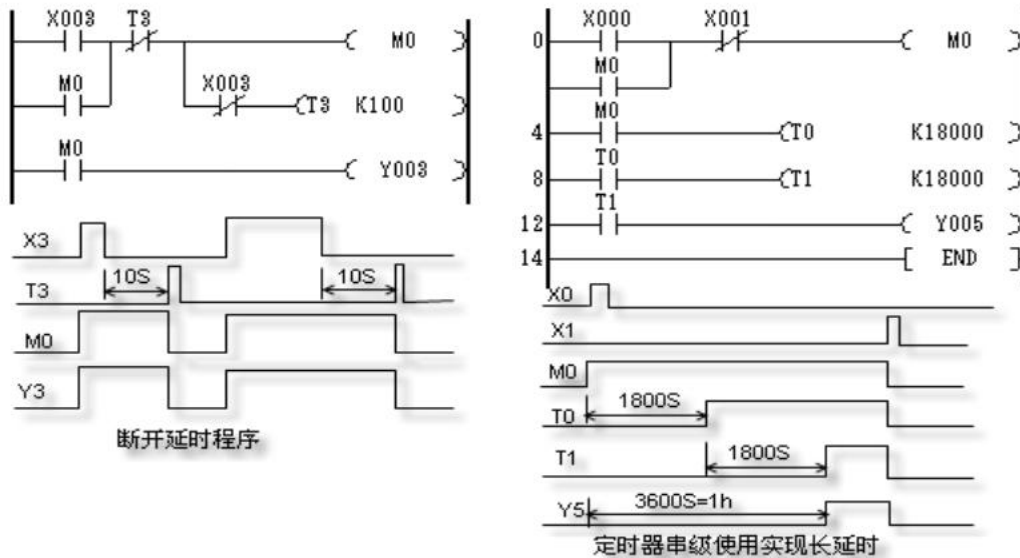
1) 接通延时



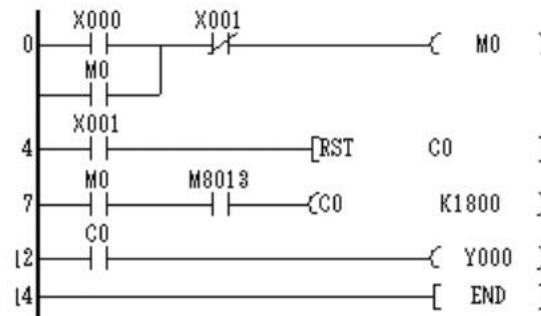
2) 限时控制程序



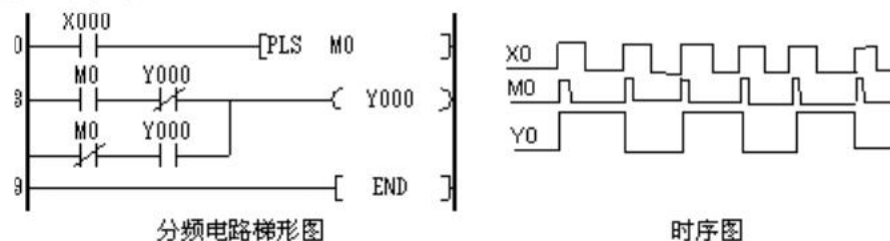
3) 断开延时和长延时



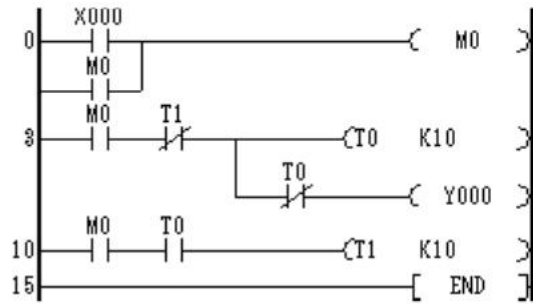
1) 计数器配合计时



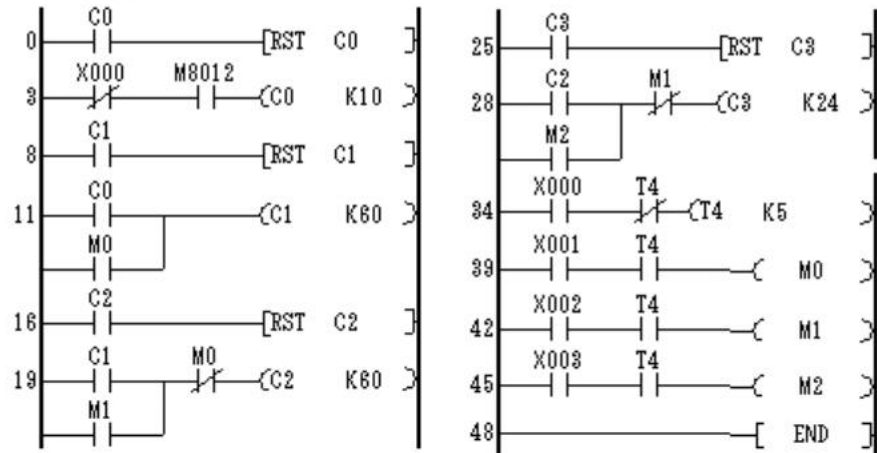
4. 分频电路程序



5. 振荡电路程序



6. 时钟电路程序



时钟电路程序

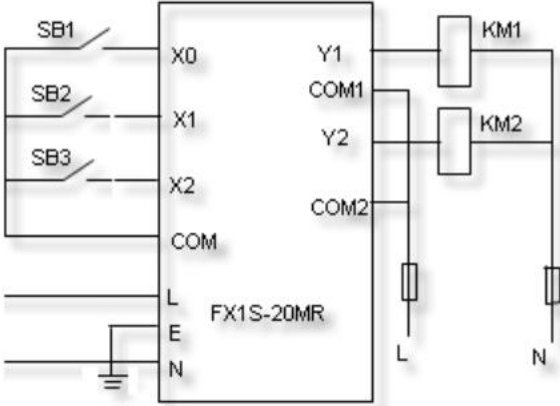
四、 编程实例

一) 电动机正反转控制

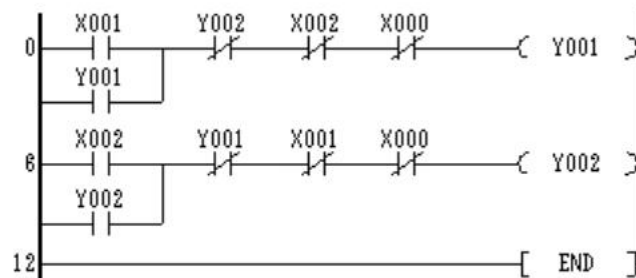
1. 分析工艺过程
2. PLC 的 I/O 点的确定和分配

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

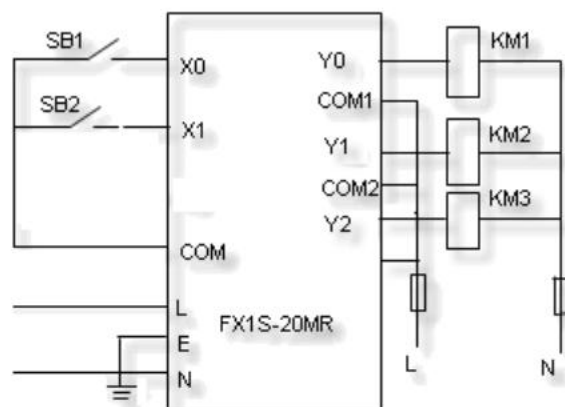
控制要求为：点火过程为先启动引风，5 分钟后启动鼓风，2 分钟后点火燃烧；熄火过程为先熄灭火焰，2 分钟后停止鼓风，5 分钟后停止引风。

1. 分析工艺过程

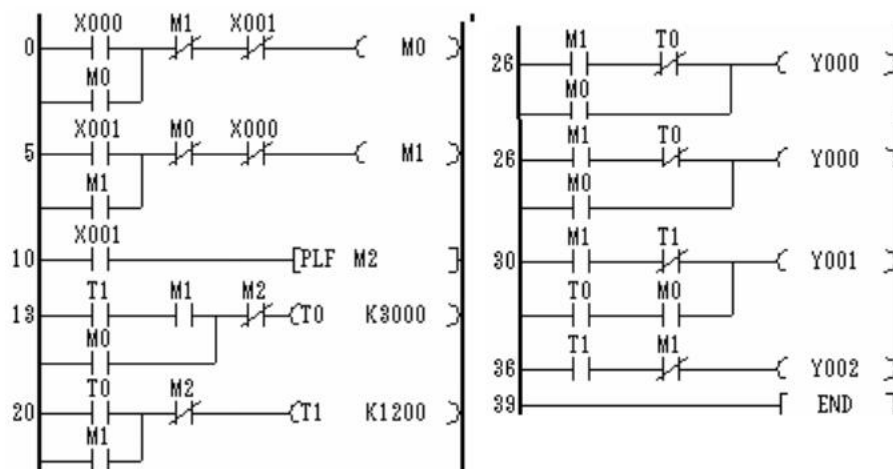
2. PLC 的 I/O 点的确定与分配

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



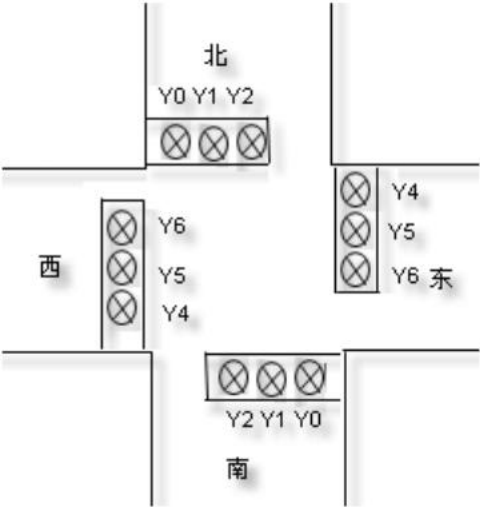
4. 控制程序编写



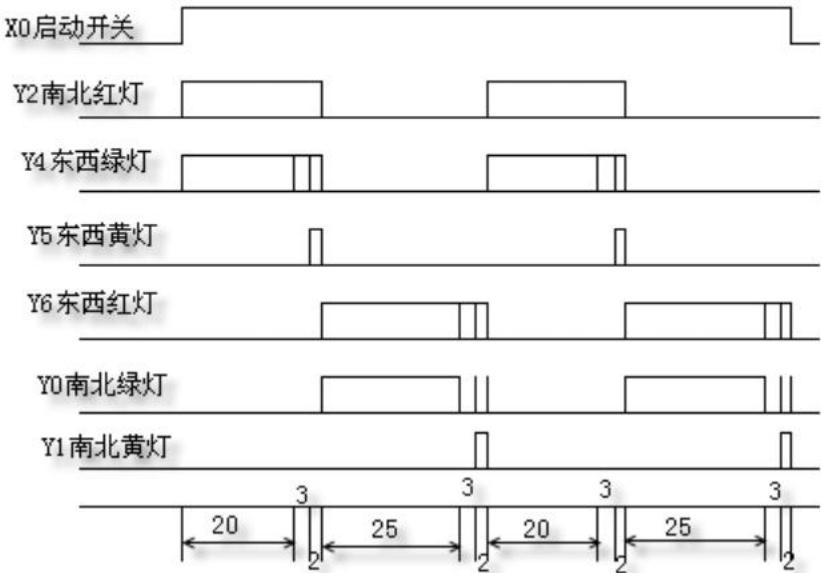
5. 调试。

三) 十字路口交通信号灯控制

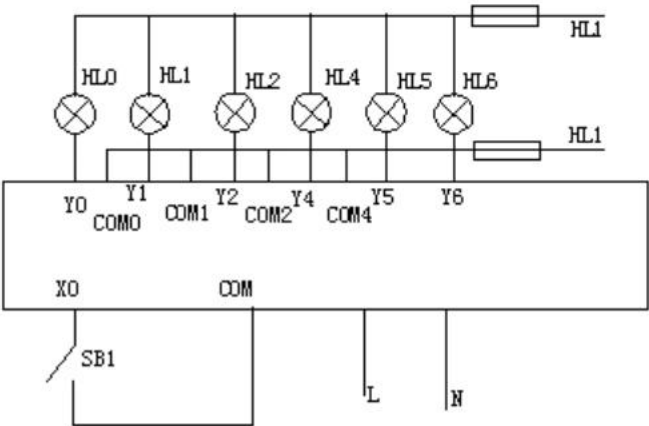
1. 分析工艺过程



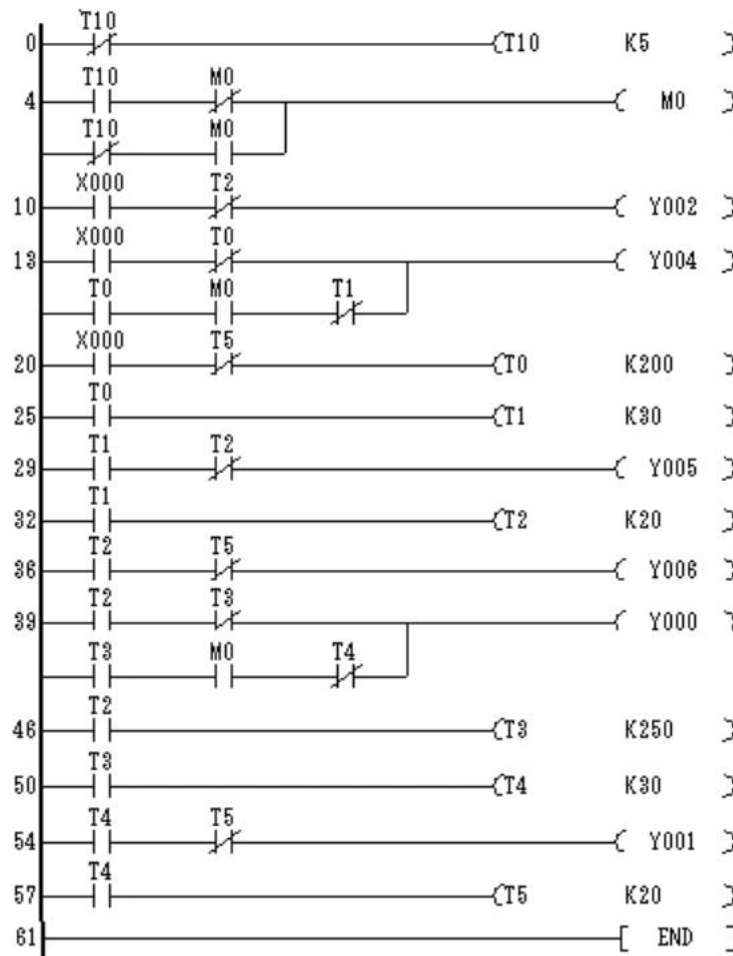
2. 十字路口交通信号灯控制时序图要求



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	OUT Y001
3	OR	X002	14	MPP
4	ANB		15	AND X007
5	OUT	Y000	16	OUT Y002
6	MRD		17	LD X010

7	LDI	X003	18	ORI	X011
8	AND	X004	19	ANB	
9	LD	X005	20	OUT	Y003
10	ANI	X006			

3. 画出题 3 指令语句表的梯形图。

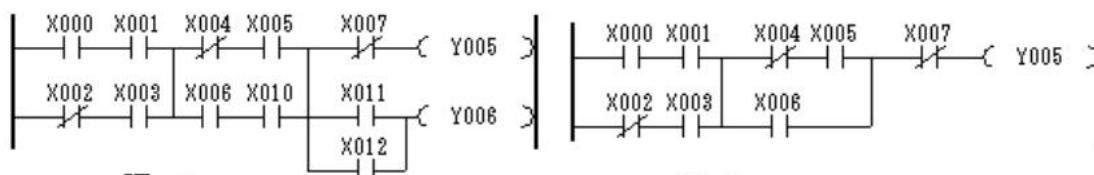
0	LD	X000	0	LD	X000	9	OUT	Y000
1	ANI	M0	1	MPS		10	MPP	
2	OUT	M0	2	AND	X001	11	OUT	Y001
3	LDI	X000	3	MPS		12	MPP	
4	RST	C0	4	AND	X002	13	OUT	Y002
6	LD	M0	5	MPS		14	MPP	
7	OUT	C0	6	AND	X003	15	OUT	Y003
10	LD	C0	7	MPS		16	MPP	
11	OUT	Y000	8	AND	X004	17	OUT	Y004
12	END					18	END	

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。

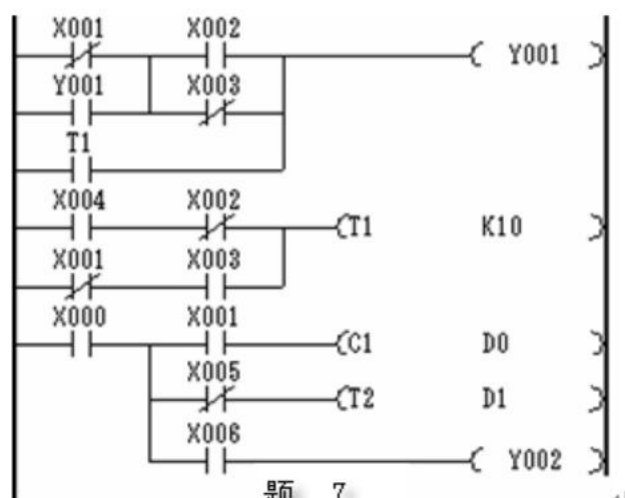


题 5

题 6

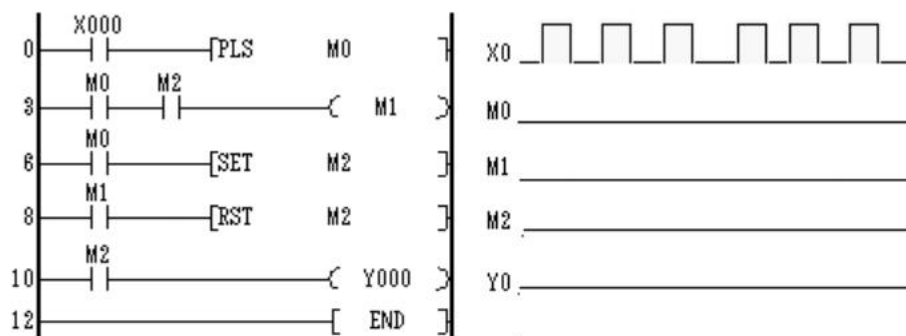
1. 写出上面题 6 梯形图的指令语句表。

2. 写出题 7 梯形图的指令语句表。

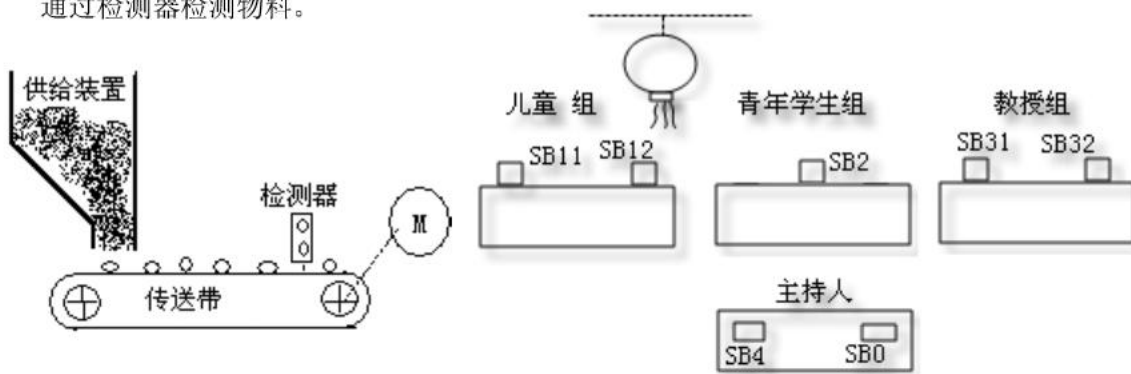


题 7

3. 对下题进行时序分析。



4. 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。
5. 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。



6. 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。
7. 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。
8. 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。

9. 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。
10. 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

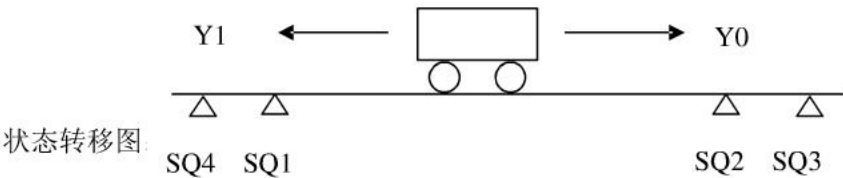
在状态转移图中，每个状态都分别采用连续的、不同的状态继电器表示。FX1S 系列 PLC 的状态继电器的分类、编号、数量及功能。

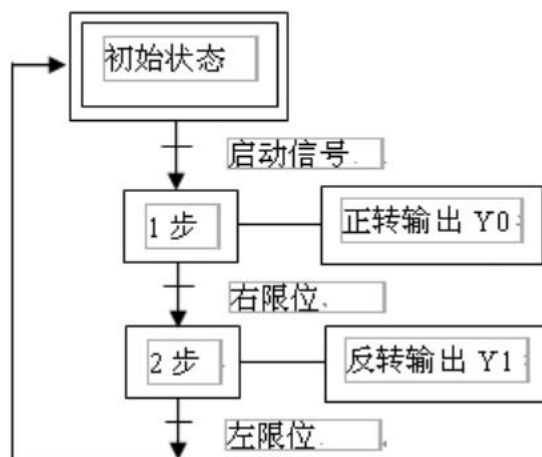
类 别	状态继电器编号	数 量	功能说明
初始化状态继电器	S0——S9	10 点	初始化
原点回归状态继电器	S10——S127	118 点	用 IST 指令时原点回归
掉电保持型继电器	S0——S127	128 点	具有停电记忆功能，停电后再启动，可继续执行

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。





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

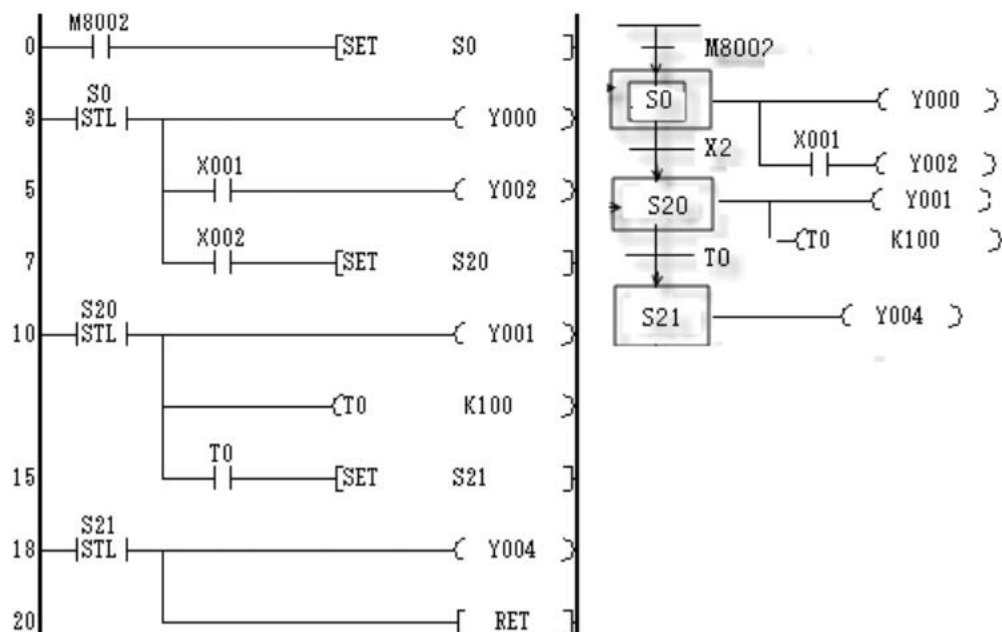
- (1) 分析控制要求和工艺流程，确定状态转移图结构（复杂系统需要）。
- (2) 工艺流程分解若干步，每一步表示一稳定状态。
- (3) 确定步与步之间转移条件及其关系。
- (4) 确定初始状态。（可用输出或状态器）
- (5) 解决循环及正常停车问题。
- (6) 急停信号的处理。

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

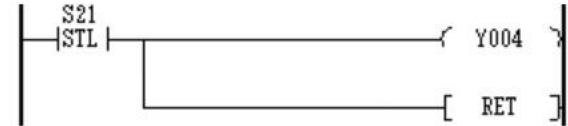
STL 指令称为“步进接点”指令。其功能是将步进接点接到左母线。



操作元件：状态继电器 S。

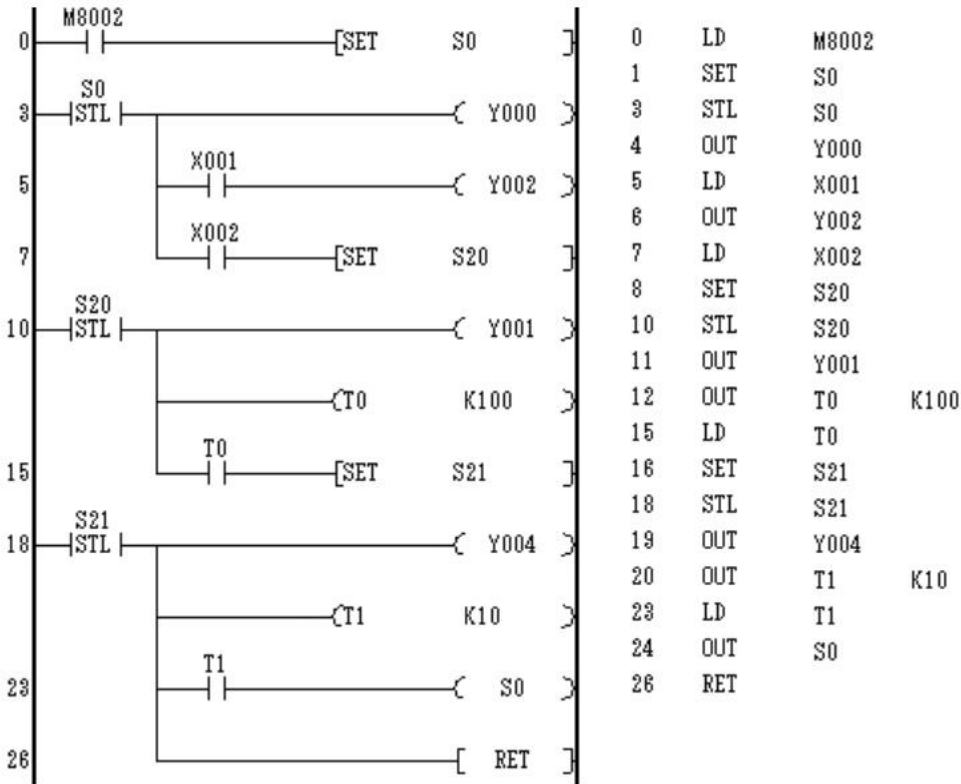
RET 指令称为“不仅返回”指令。其功能是使临时左母线回到原来左母线的位置。

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轮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：

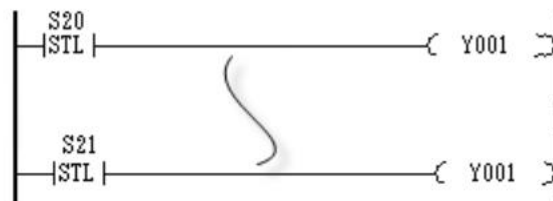
- (1) 步进接点与左母线相连时，具有主控和跳转作用；
- (2) 状态继电器 S 只有在使用 SET 指令以后才具有步进控制功能，提供步进接点。

- (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

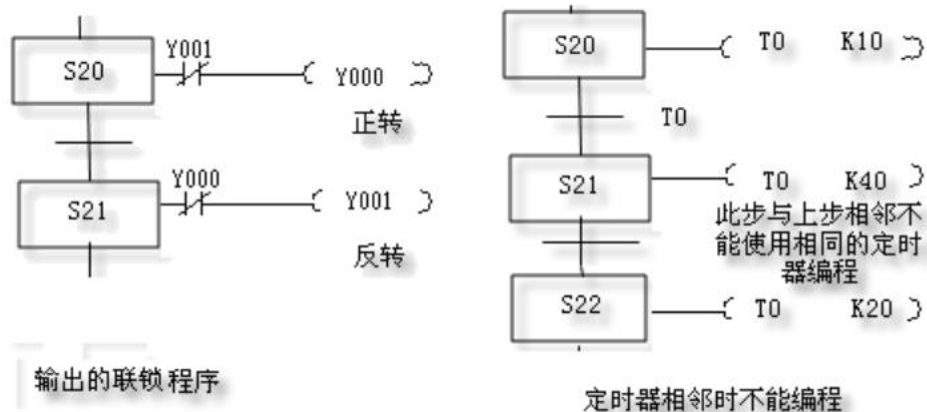
1. 状态的动作与输出的重复使用

- (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；



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。

有一个机械动作作为图所示. 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。



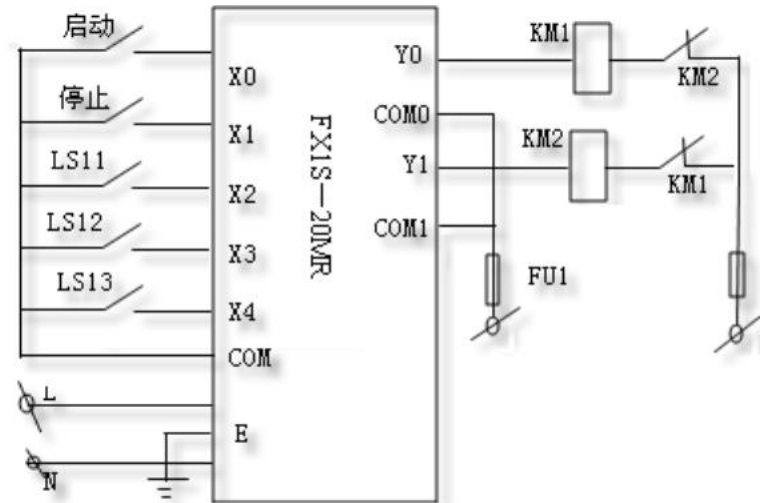
1) 解：图中给出了台车机械动作的过程，分作两次前进和后退，进程长度不一样。

2) I/O 分配。

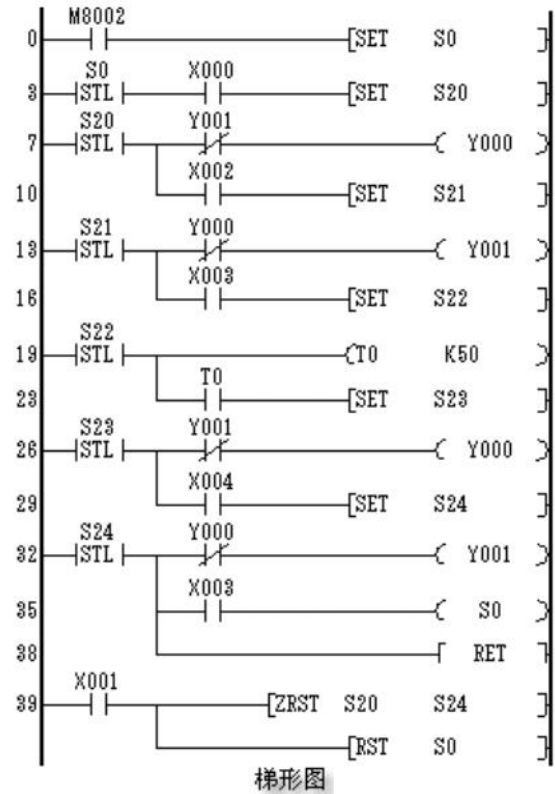
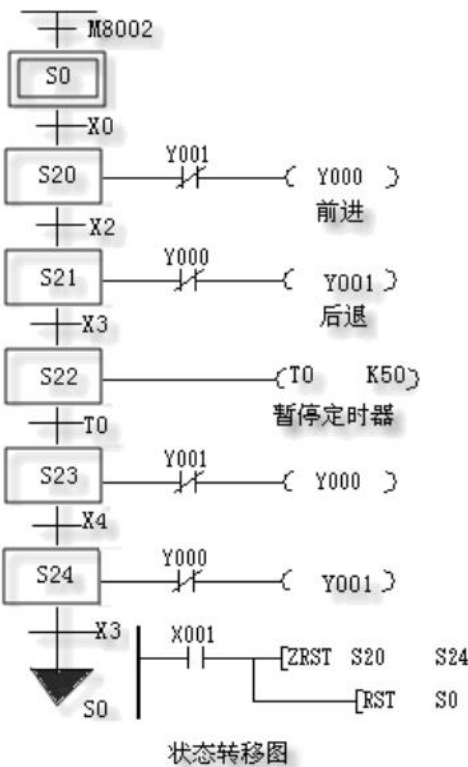
输 入		输 出	
启动按钮	X0	前进	Y0
停止按钮	X1	后退	Y1
开关 LS11	X2		
开关 LS12	X3		

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	