

生产线控制与维修技术

PLC 功能模块介绍、三菱 PLC 编程软件、指令系统和控制应用

学习任务：循环指令 FOR、NEXT；

时钟专用的特殊辅助继电器和特殊

数

据寄存器；

循环左移指令 ROL、循环右移指令

ROR、

适用年级：机电一体化专业 **适用专业**：机电一体化专业 **适用课程**：PLC 应用

SFTR；

4 学时

任课教师：姚贵发

SOC3：设计三菱 PLC 控制生产线输送产品，检测产品到达位置并自动将产品抓取放到另外一条生产线程序

成果形式：要有 I/O 分配、程序表、外部接线图和外形结构图。

考核方法：内容正确完整，符合设计目标要求 / 得
实平台提交

本次课学习重点与难点：

重点：循环指令 FOR、NEXT、特殊辅助继电器；
特殊数据寄存器；

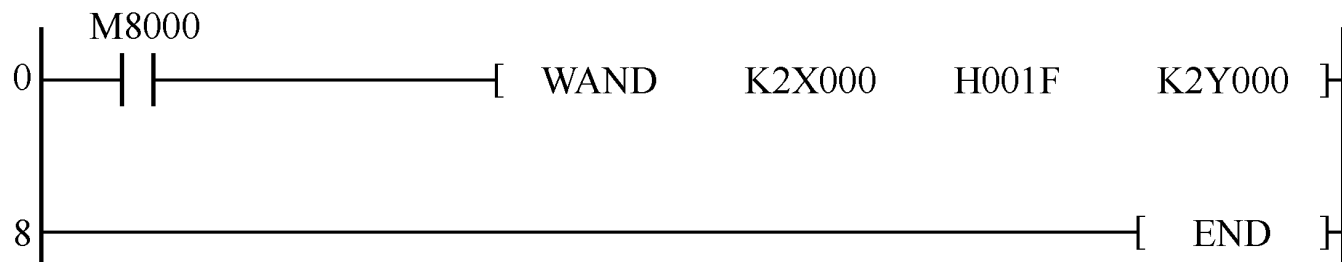
难点：循环左移指令 ROL、循环右移指令 ROR；



一、教学回顾

1. 逻辑字 “与” 指令 WAND

字 “与” 指令		操 作 数	
D	FNC26 WAND	S1 、 S2	K 、 H 、 KnX 、 KnY 、 KnM 、 KnS 、 T 、 C 、 D 、 V 、 Z
P		D	KnY 、 KnM 、 KnS 、 T 、 C 、 D 、 V 、 Z



	X7	X6	X5	X4	X3	X2	X1	X0
K2X0	1	0	1	1	1	0	1	0

H1F	0	0	0	1	1	1	1	1
-----	---	---	---	---	---	---	---	---

K2Y0	0	0	0	1	1	0	1	0
------	---	---	---	---	---	---	---	---

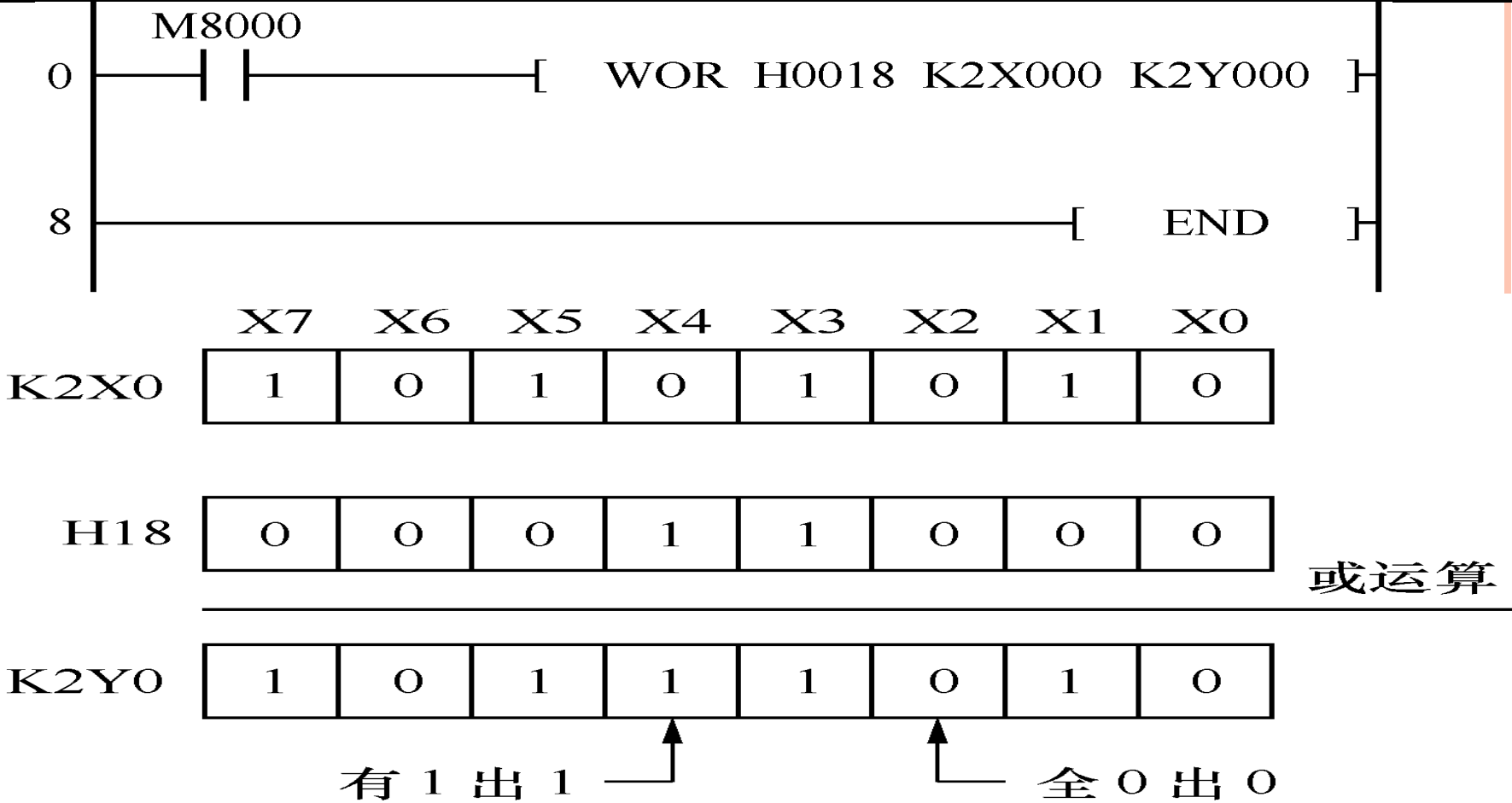
与运算

全 1 出 1 有 0 出 0

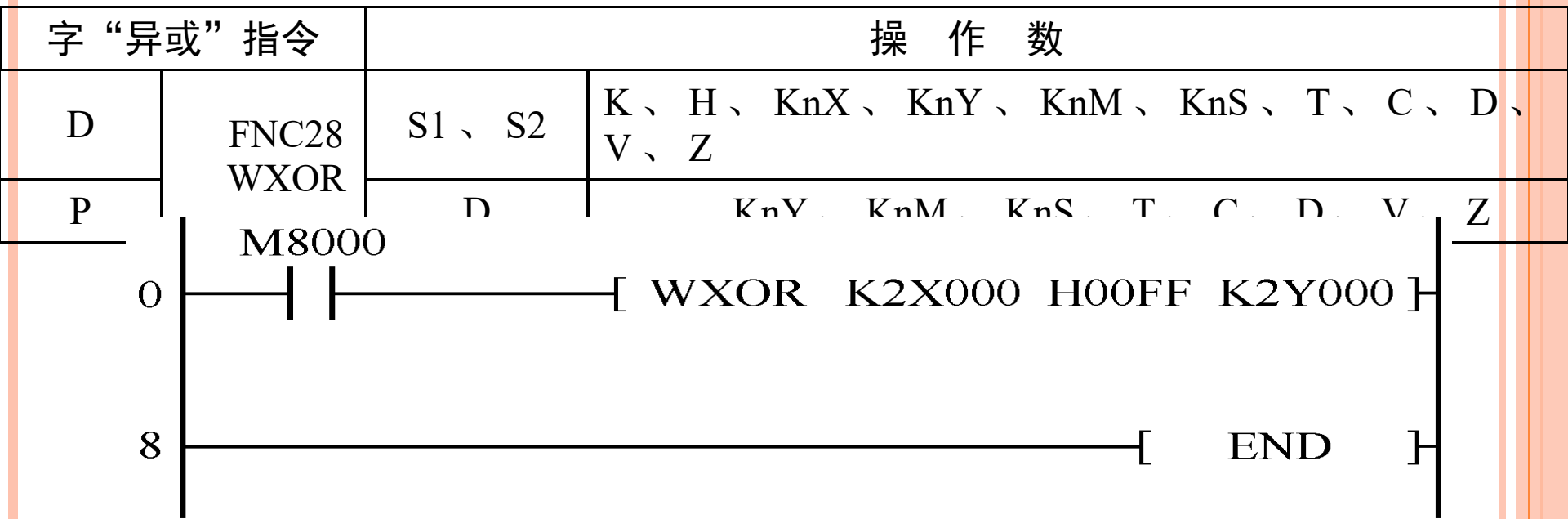


2. 逻辑字 “或” 指令 WOR

字 “或” 指令		操 作 数	
D	FNC27 WOR	S1 、 S2	K 、 H 、 KnX 、 KnY 、 KnM 、 KnS 、 T 、 C 、 D 、 V 、 Z
P		D	KnY 、 KnM 、 KnS 、 T 、 C 、 D 、 V 、 Z



3. 逻辑字 “异或” 指令 WXOR



	X7	X6	X5	X4	X3	X2	X1	X0
K2X0	1	0	1	0	1	0	1	0

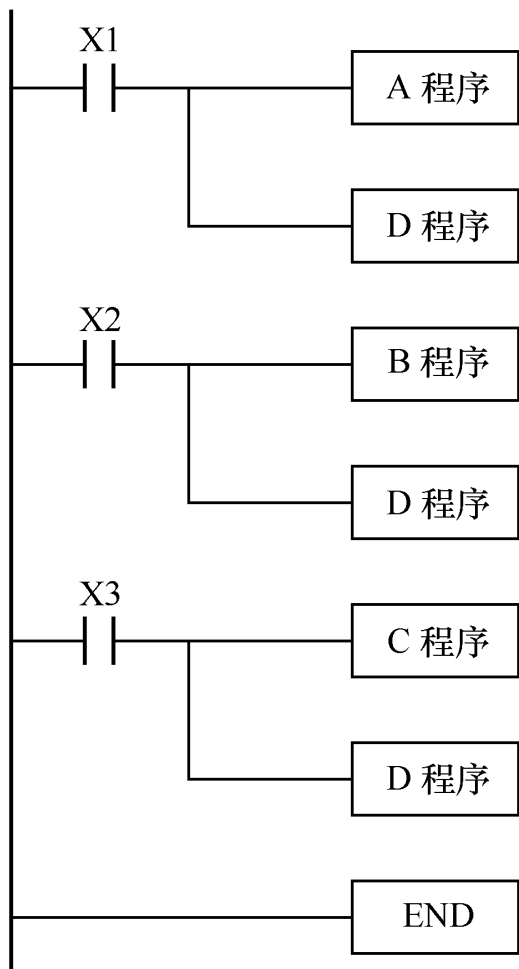
H0FF	1	1	1	1	1	1	1	1
------	---	---	---	---	---	---	---	---

异或运算

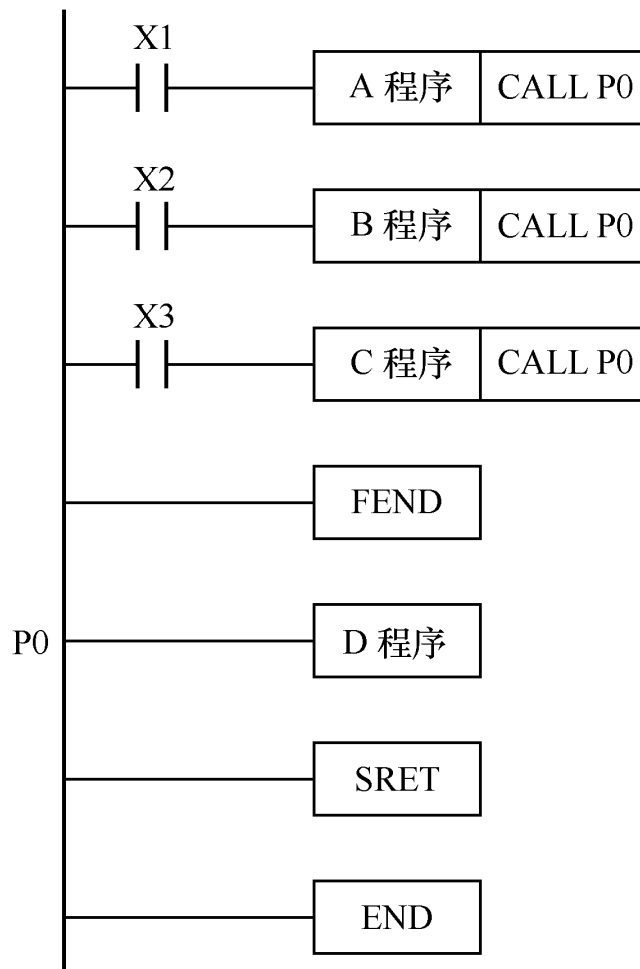
K2Y0	0	1	0	1	0	1	0	1
------	---	---	---	---	---	---	---	---

相异出 1 相同出 0

4. 子程序调用指令及应用



(a) 顺序控制程序

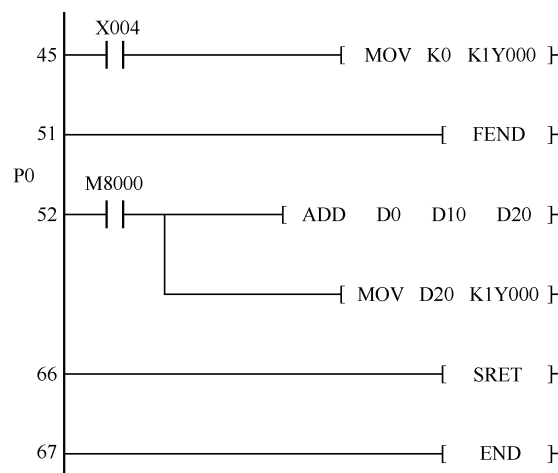
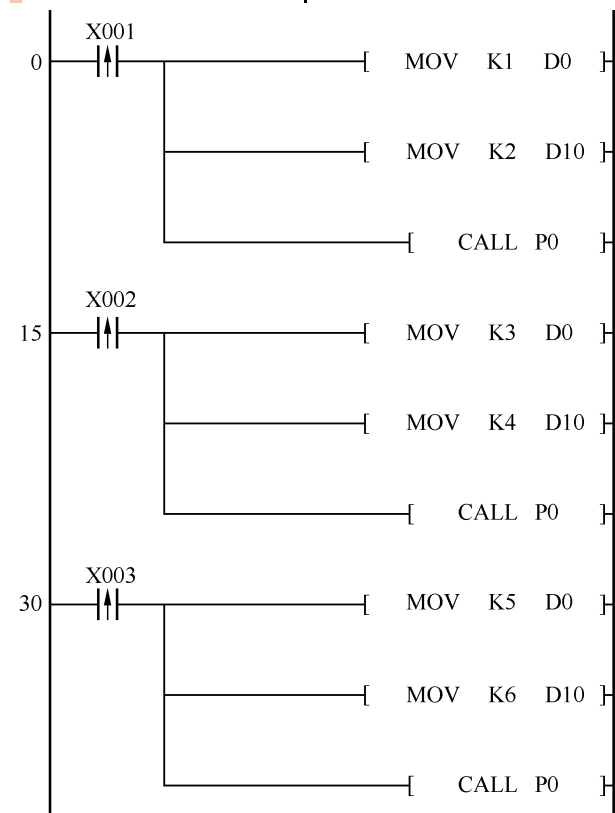


(b) 子程序调用程序



子程序指令 CALL、 SRET 与主程序结束指令 FEND

指令助记符		操 作 数	程 序 步
P	FNC1 CALL	标号 P0 ~ P62	CALL 3 步
		标号 P64 ~ P127	标号 P 1 步
	FNC2 SRET	无	SRET 1 步
	FNC6 FEND	无	FEND 1 步



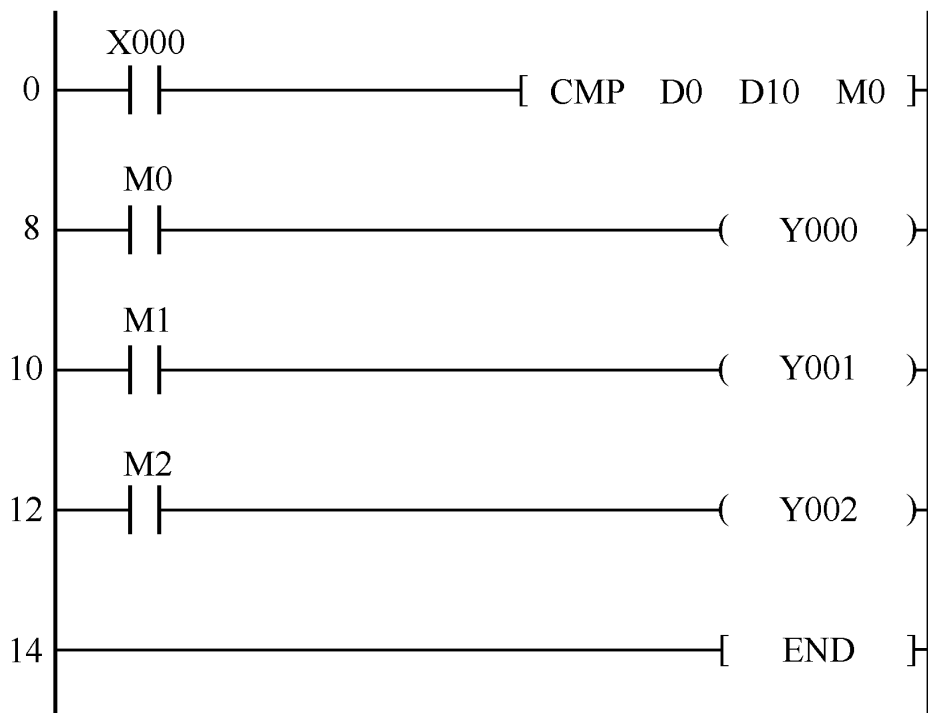
程序功能是：
X1、X2、X3 分别接通时，将相应的数据传送到 D0、D10，然后调用子程序；在子程序中，将 D0、D10 存储的数据相加，运算结果存储在 D20，用 D20 存储数据控制输出字元件 K1Y0。



5. 比较指令

组件比较指令 CMP

比 较 指 令		操 作 数	
D	FNC10 CMP	S1 、 S2	K、 H、 KnX、 KnY、 KnM、 KnS、 T、 C、 D、 V、 Z
P		D	Y、 M、 S



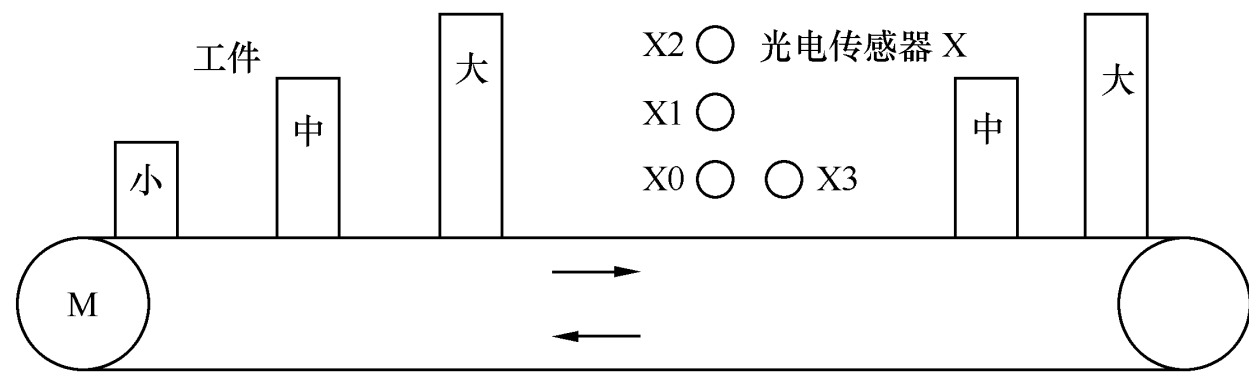
标志位的规则：

若 $(D0) > (D10)$ ，则
M0 置 1，M1、M2 为 0；

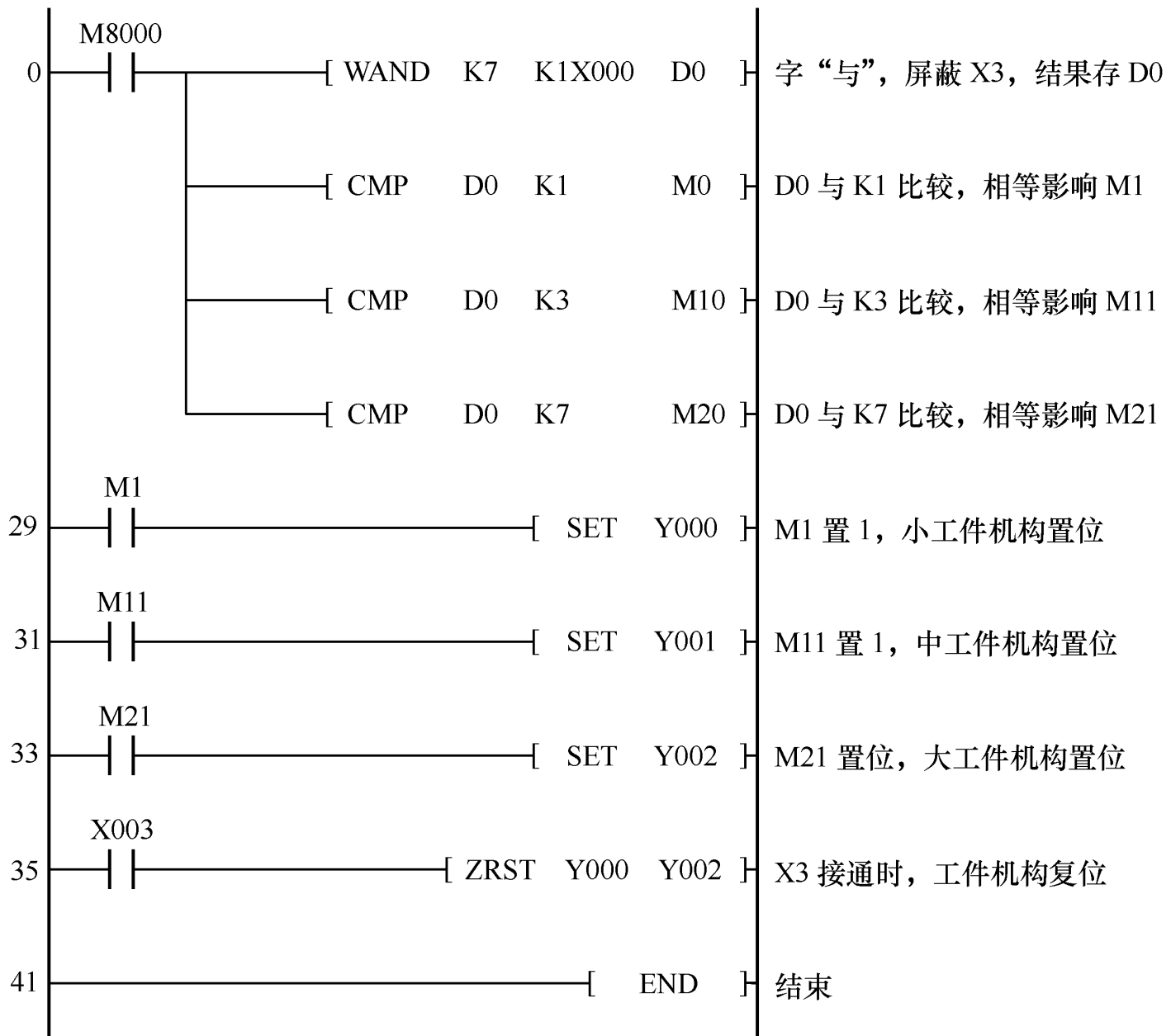
若 $(D0) = (D10)$ ，则
M1 置 1，M0、M2 为 0；

若 $(D0) < (D10)$ ，则
M2 置 1，M0、M1 为 0。

如图所示的传送带输送大、中、小三种规格的工件，用连接 X0、X1、X2 端子的光电传感器判别工件规格，然后启动分别连接 Y0、Y1、Y2 端子的相应操作机构；连接 X3 的光电传感器用于复位操作机构。用比较指令 CMP 编写工件规格判别程序。



工 件 规 格	光电信号输入控制字 K1X0				光电转换数据
	X3	X2	X1	X0	
小	0	0	0	1	K1
中	0	0	1	1	K3
大	0	1	1	1	K7

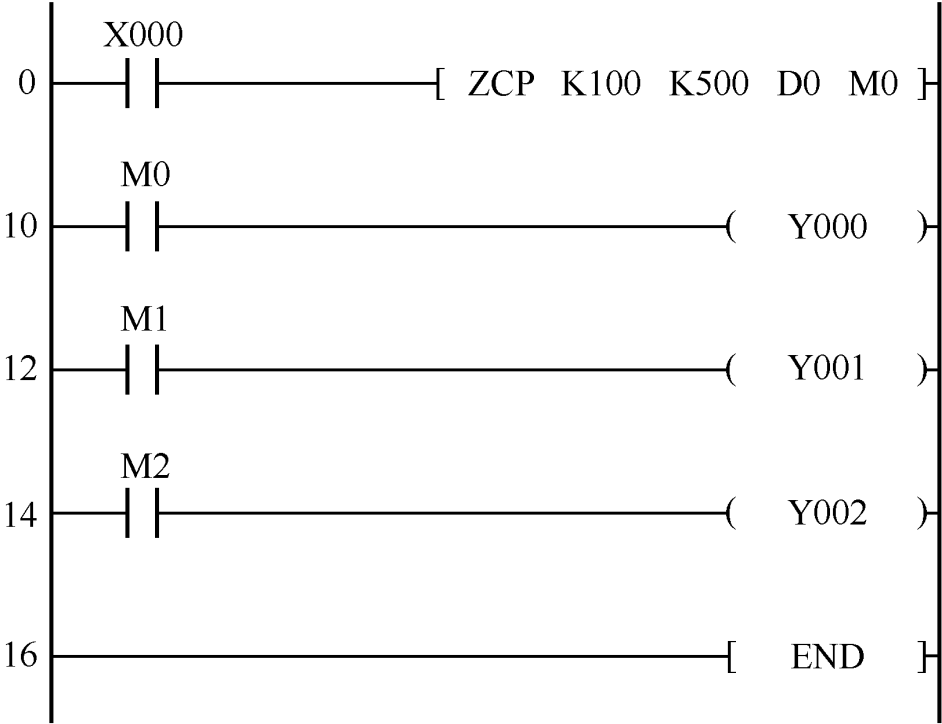


传送带工件规格判别程序



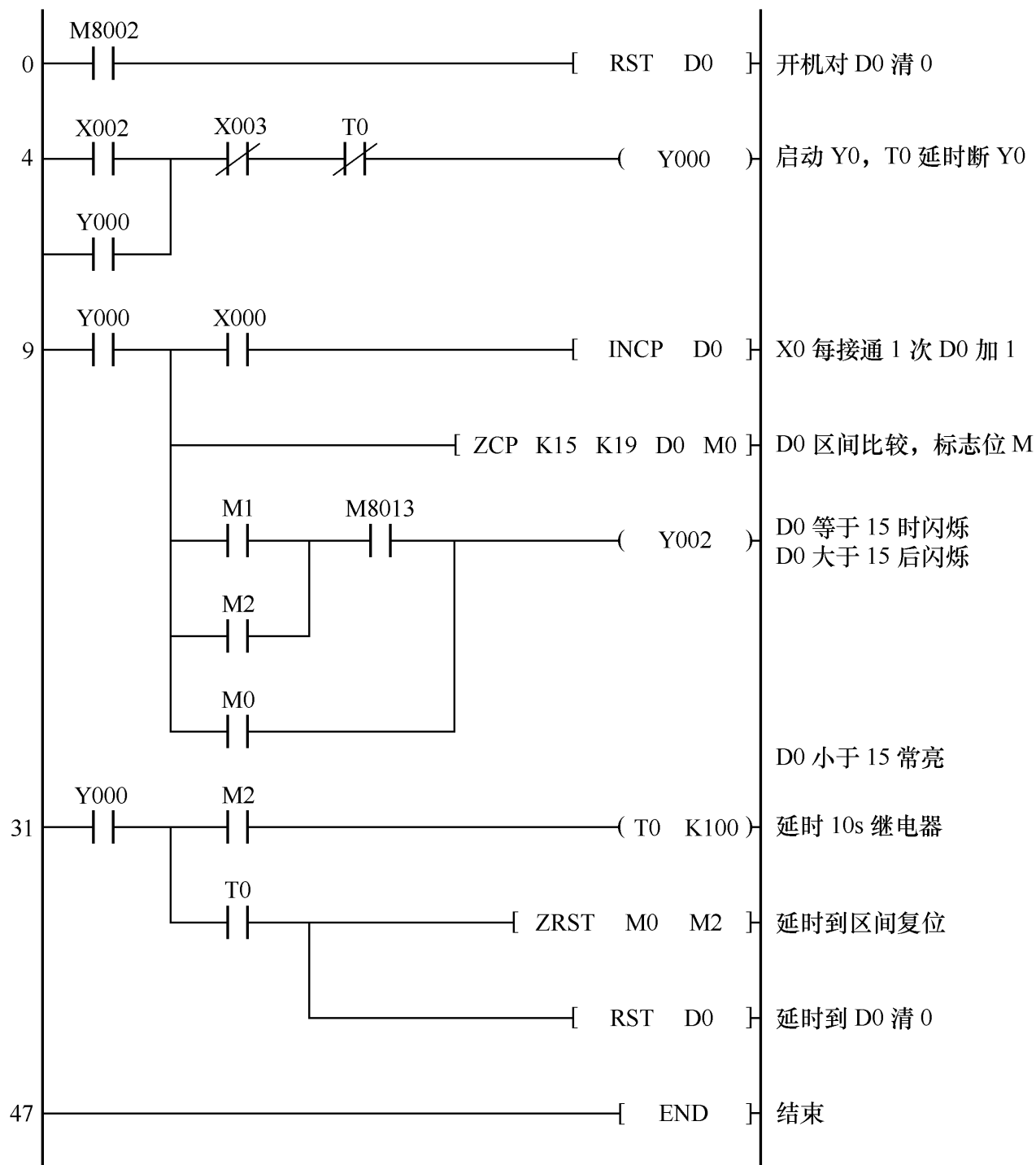
区域、区间比较指令

指令名称	助记符	指令代码 位数	操作数范围			程序步
			[S1 .]	[S2.]	[D.]	
区域比较	ZCP ZCP(P)	FNC11 (16/32)	K、H KnX、Kn Y、KnM 、KnS T、C、D 、V、Z	Y、 M、 S		ZCP、ZCPP ...9步 DZCP、DZC PP...17步



标志位的规则：
若 $K100 > (D0)$ ，则 M0 置 1，M1、M2 为 0；
若 $K100 \leq (D0) \leq K500$ ，则 M1 置 1，M0、M2 为 0；
若 $K500 < (D10)$ ，则 M2 置 1，M0、M1 为 0。





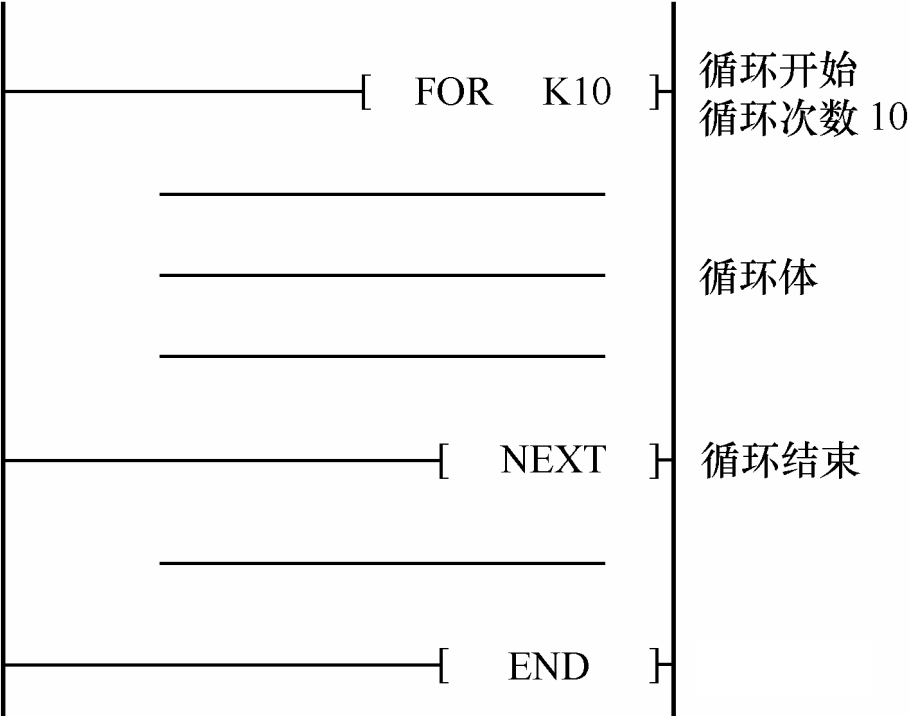
二、讲授新内容

1. 循环指令及应用

1.1 循环指令

FOR、NEXT

指令助记符		操 作 数	程序步
循环开始	FNC8 FOR	K、H、KnX、KnY、KnM、KnS、 T、C、D、V、Z	3
循环结束	FNC9 NEXT	无	1

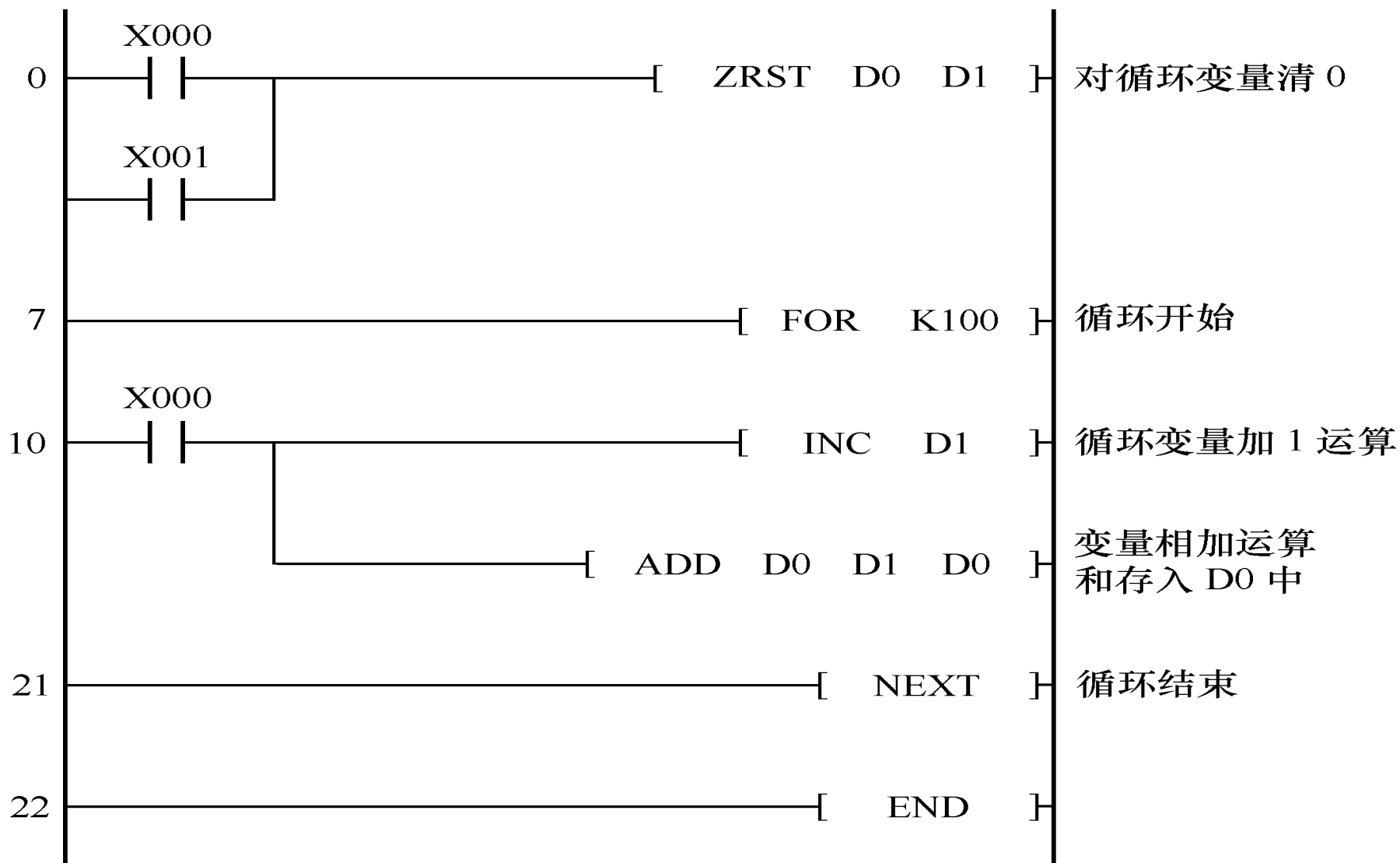


循环指令 FOR、NEXT 的说明
FOR、NEXT 指令必须成对出现，缺一不可。位于 FOR、NEXT 之间的程序称为循环体，在一个扫描周期内，循环体反复被执行。FOR 指令的操作数用于指定循环的次数，只有执行完循环次数后，才执行 NEXT 的下一条指令语句。循环指令的结构如图所示，图中指定循环次数为 10 次。

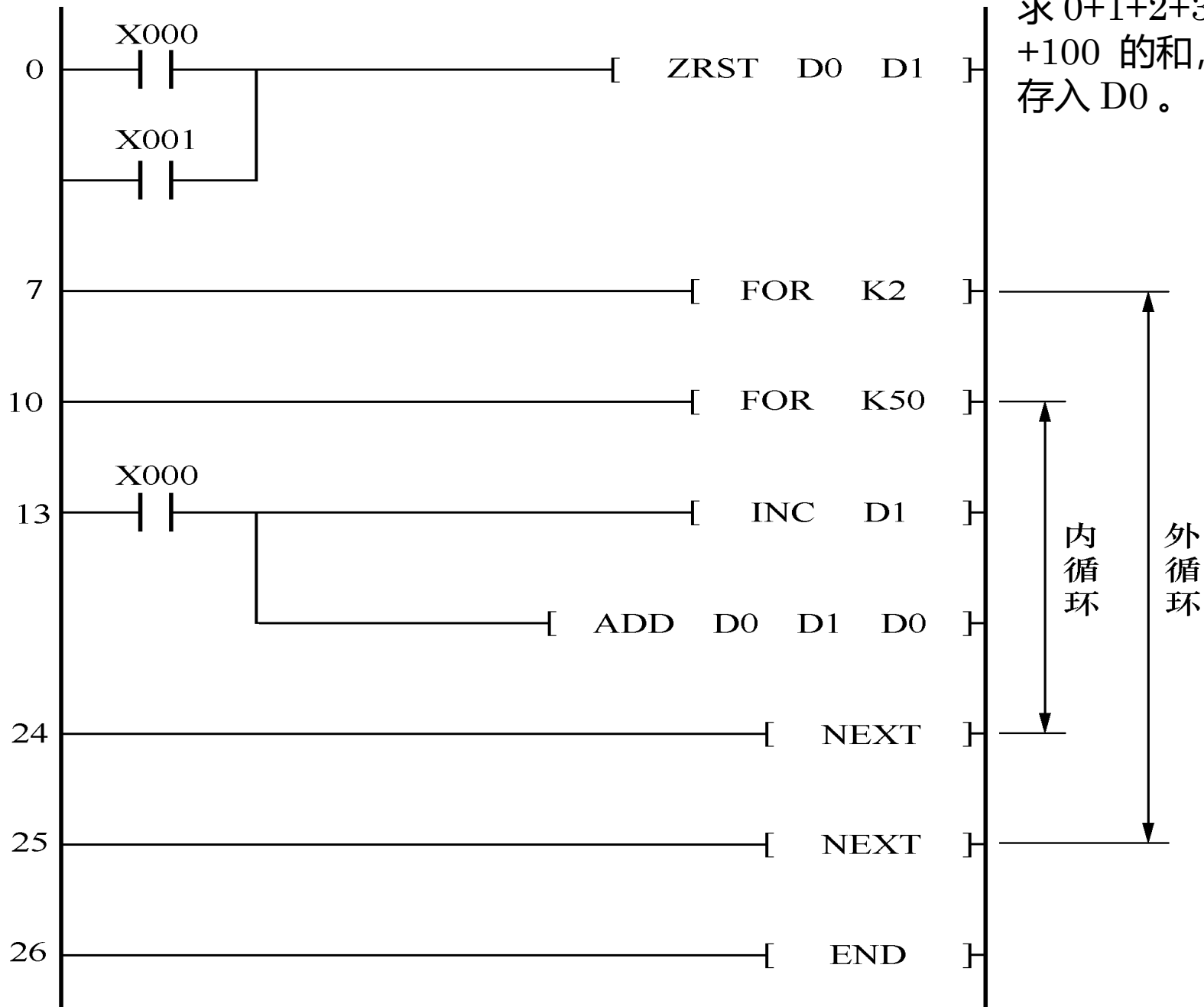
1.2 循环指令 FOR、NEXT 举例

例求 $0+1+2+3+\dots+100$ 的和，并将和存入 D0。

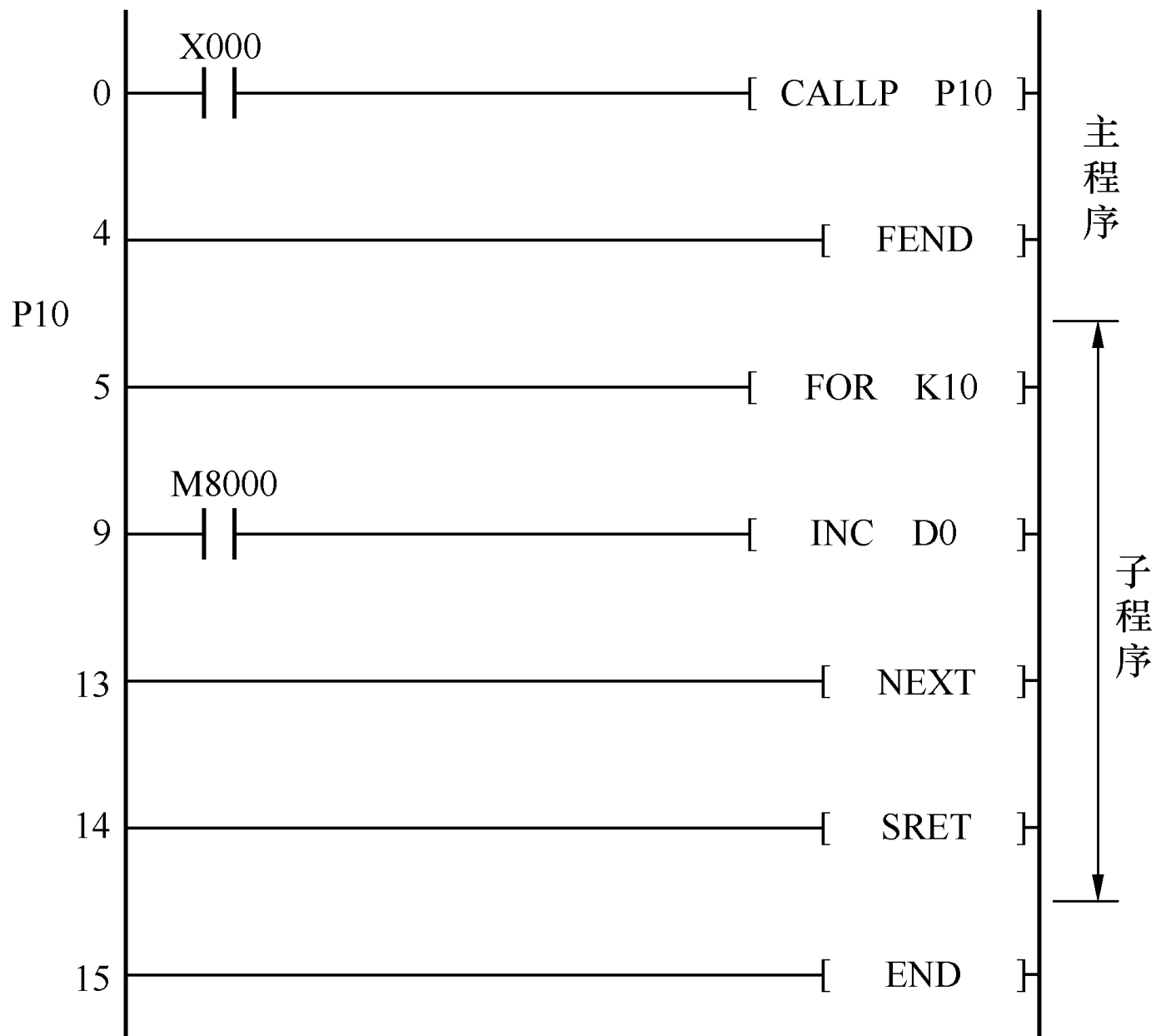
解：用循环指令编写的程序如图所示，D1 作为循环增量。



求 $0+1+2+3+\dots+100$ 的和, 并将和存入 D0。



1.3 循环指令的脉冲执行方式



在本例中，
每按下一次按钮
接通 X0 时，执行
一次循环指令，
数据寄存器 D0 中
存储的数据就增
加 10。

循环指令的脉冲
执行方式

2. 马路照明灯时钟控制程序

2.1 时钟专用的特殊辅助继电器和特殊数据寄存器

特殊辅助继电器功能

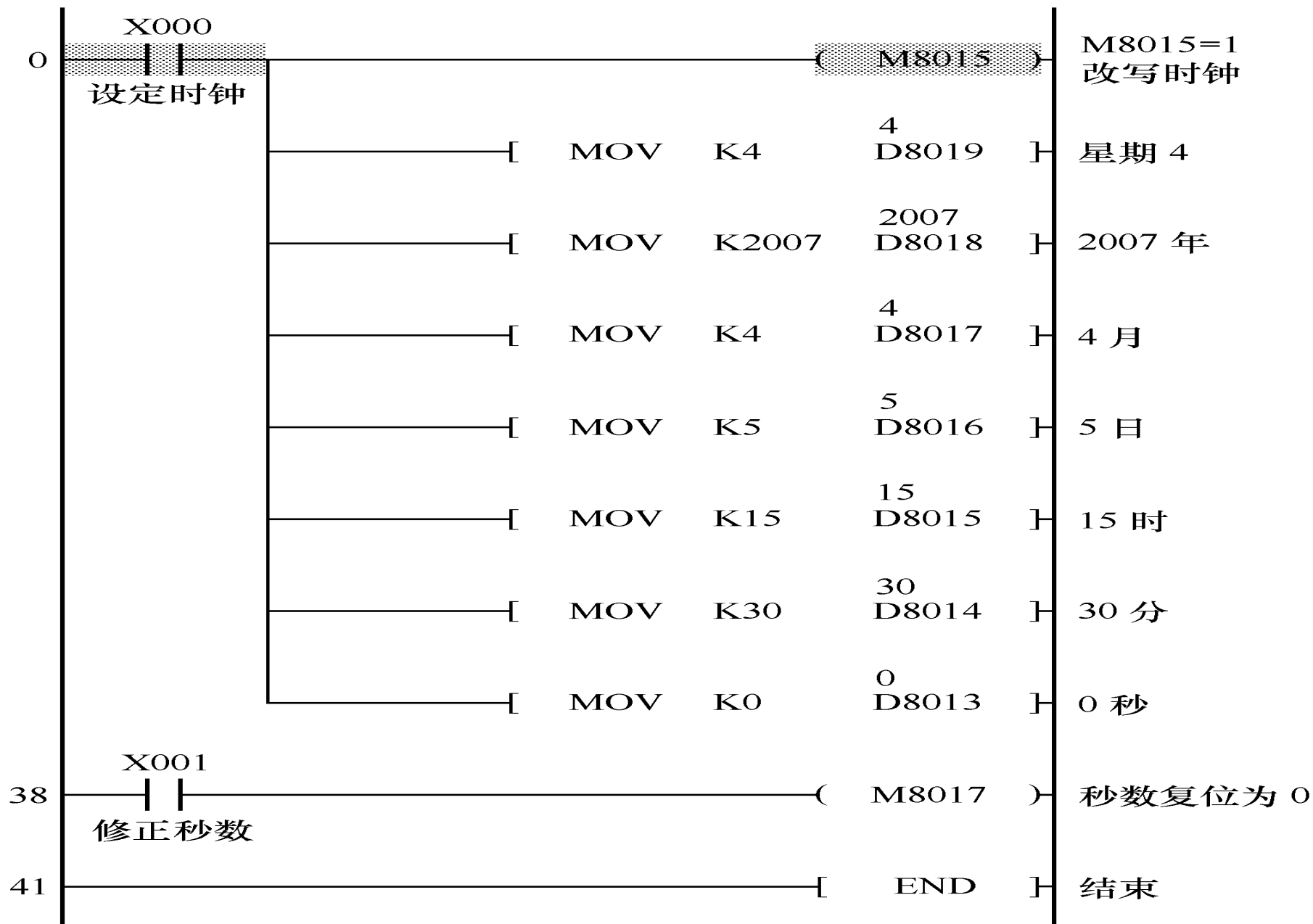
特殊辅助继电器	作 用	功 能
M8015	时钟停止和改写	=1 时钟停止，改写时钟数据
M8016	时钟显示停止	=1 停止显示
M8017	秒复位清 0	上升沿时修正秒数
M8018	内装 RTC 检测	平时为 1
M8019	内装 RTC 错误	改写时间数据超出范围时 =1

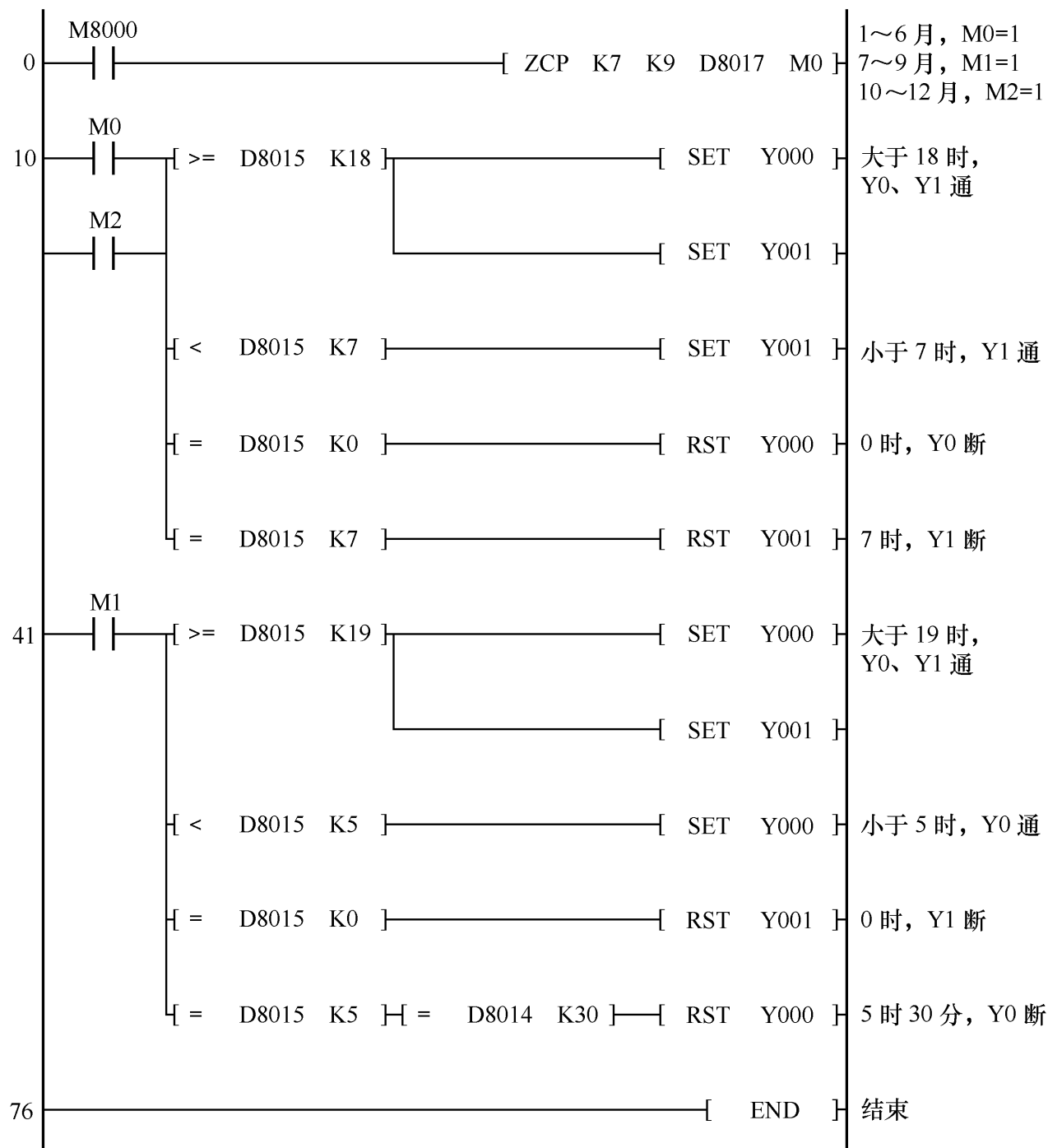
特殊数据寄存器功能

特殊数据寄存器	作 用	范 围
D8013	秒	0 ~ 59
D8014	分	0 ~ 59
D8015	时	0 ~ 23
D8016	日	1 ~ 31
D8017	月	1 ~ 12
D8018	年	公历 4 位
D8019	星期	0 ~ 6 （周日~周六）

2.2 设定时钟信息

设定时钟信息的程序





2.3 马路照明灯时钟控制程序

设马路照明灯由 PLC 输出端口 Y0、Y1 各控制一半。每年夏季（7~9月）每天 19 时 0 分至次日 0 时 0 分灯全部开，0 时 0 分至 5 时 30 分开一半灯。其余季节每天 18 时 0 分至次日 0 时 0 分灯全部开，0 时 0 分至 7 时 0 各开一半灯。



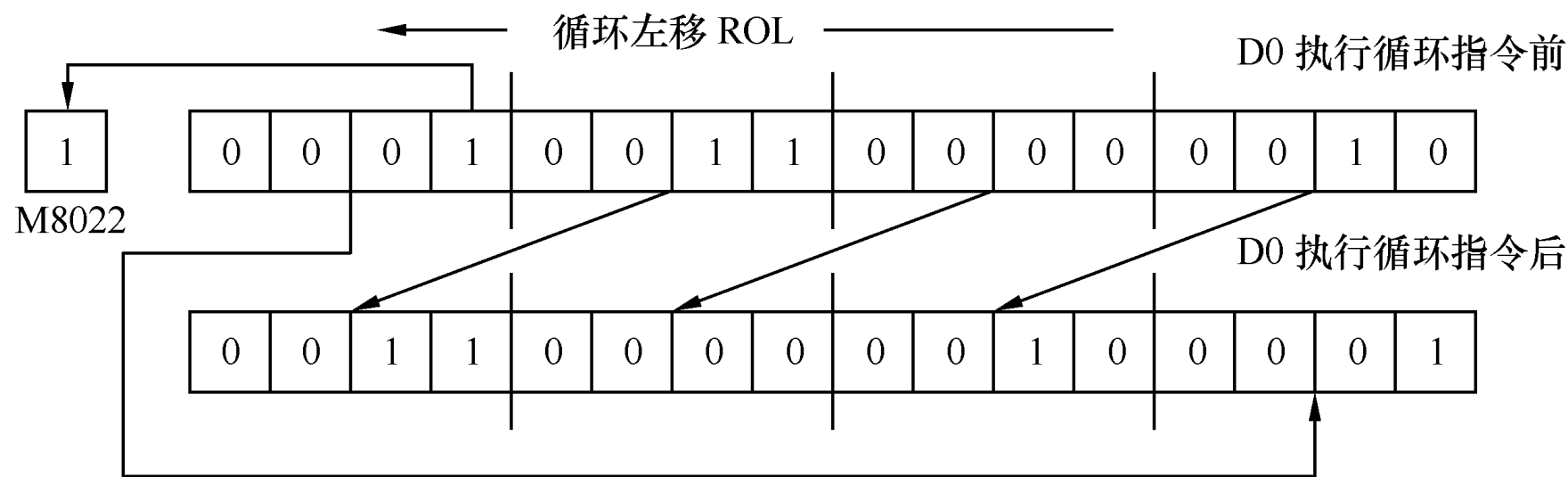
3. 循环移位指令及应用

3.1 循环左移指令 ROL

ROL 指令

循环左移指令		操 作 数	
D	FNC31 ROL	D	KnY、KnM、KnS、T、C、D、V、Z（Kn 位组件中 n = 4/8）
P		n	n≤16（16 位指令），n≤32（32 位指令）

设（D0）循环前为 H1302，则执行“ROLP D0 K4”指令后，（D0）为 H3021，进位标志位（M8022）为 1。执行过程如图所示。

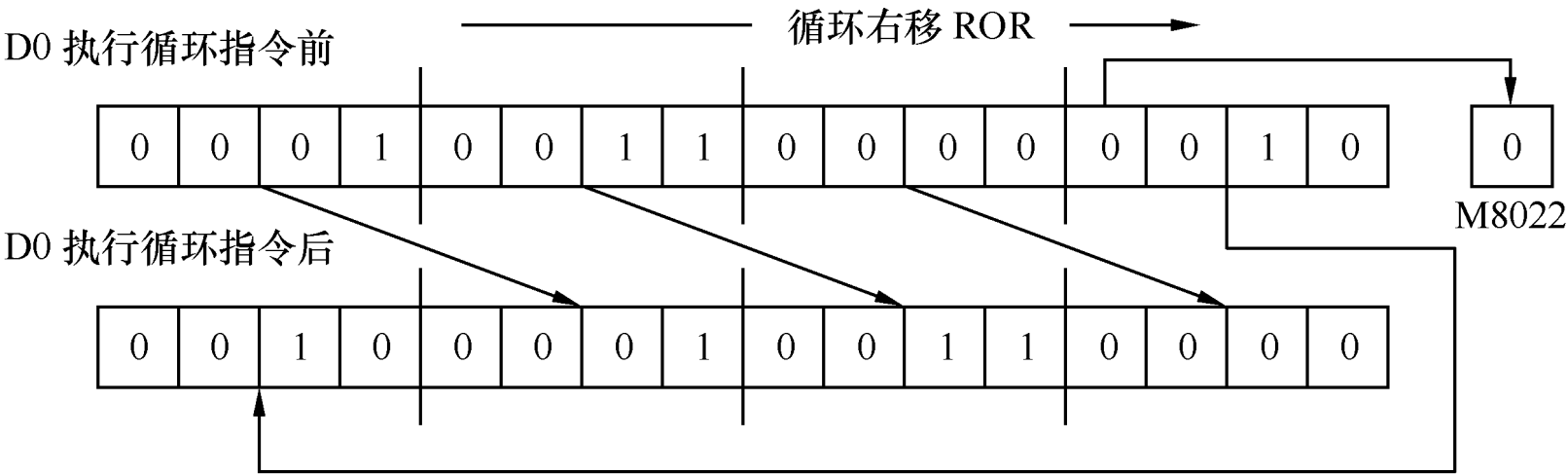


循环左移指令 ROL 执行过程

3.2 循环右移指令 ROR

ROR 指令

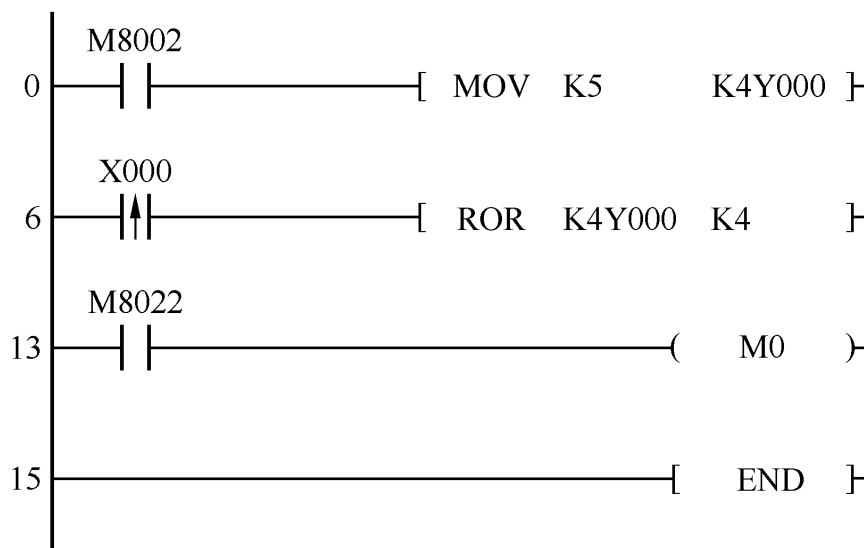
循环右移指令		操 作 数	
D	FNC30 ROR	D	K_nY 、 K_nM 、 K_nS 、 T 、 C 、 D 、 V 、 Z （ K_n 位组件中 $n = 4/8$ ）
P		n	$n \leq 16$ （16 位指令）， $n \leq 32$ （32 位指令）



循环右移指令 ROR 执行过程



例：循环右移指令 ROR 的应用举例如图所示。求输出位组件 K4Y0 在一个循环周期中各位状态的变化。



各位状态的变化

[illegible]

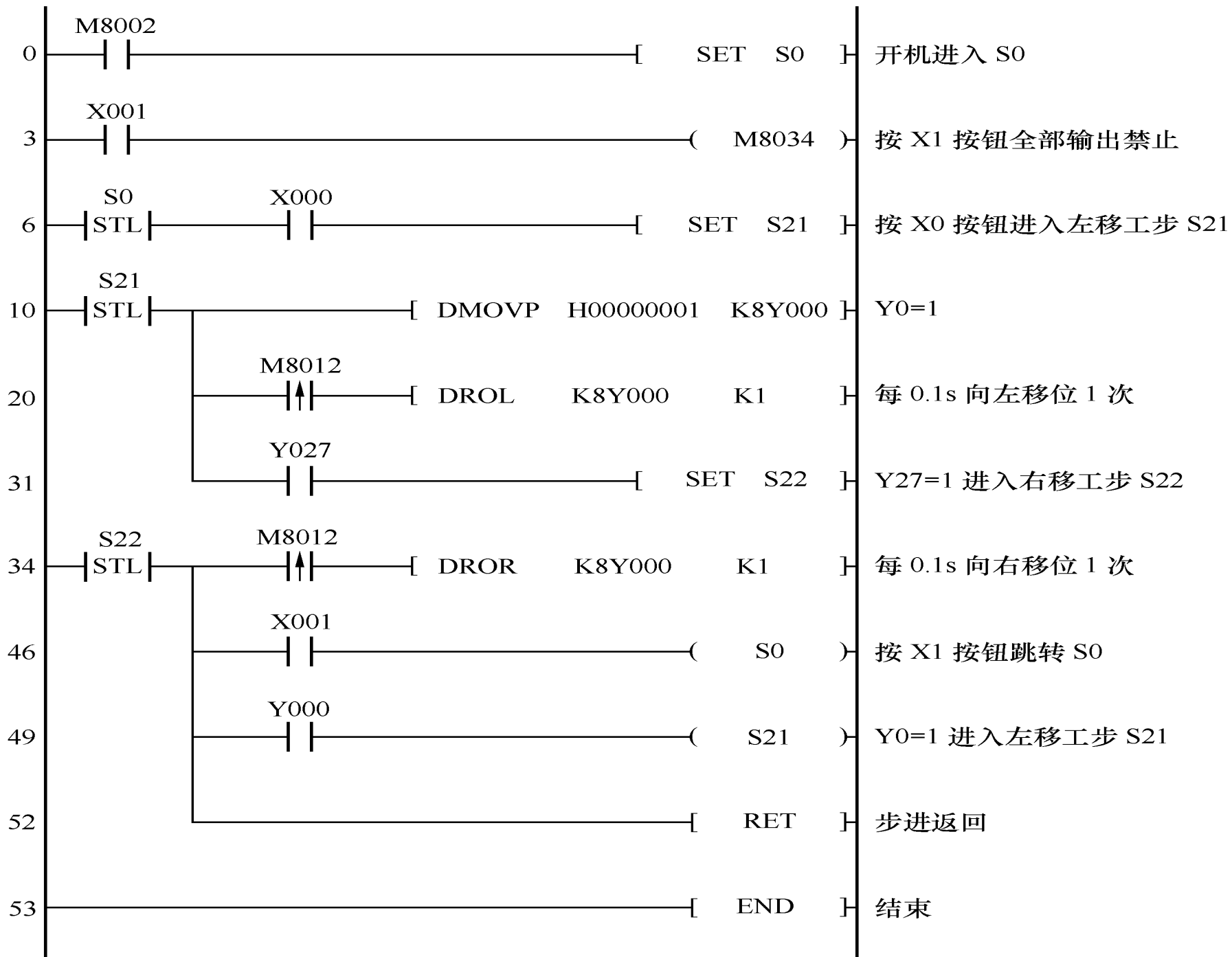
例：利用 **PLC** 实现流水灯控制。某灯光招牌有 **24** 个灯，要求按下启动按钮 **X0** 时，灯以正、反序每 **0.1s** 间隔轮流点亮；按下停止按钮 **X1** 时，停止工作。

解：由于输出动作频繁，应选择晶体管或晶闸管输出类型的 **PLC** 。流水灯控制需要 **2** 个输入端口， **24** 个输出端口。输入、输出端口的分配如表所示。

输入 / 输出端口分配表

输 入			输 出	
输入继电器	输 入 元 件	作 用	输出继电器	控 制 对 象
X0	SB1	启动按钮	Y7 ~ Y0	HL8 ~ HL1
X1	SB2	停止按钮	Y17 ~ Y10	HL16 ~ HL9
			Y27 ~ Y20	H24 ~ HL17





3.3 位左移指令 SFTL

SFTL 指令

循环左移指令		操 作 数			
P	FNC35 SFTL	S	X、Y、M、S	n1、n2	K、H
		D	Y、M、S		1≤n2≤n1≤1024

位左移指令 SFTL 的说明

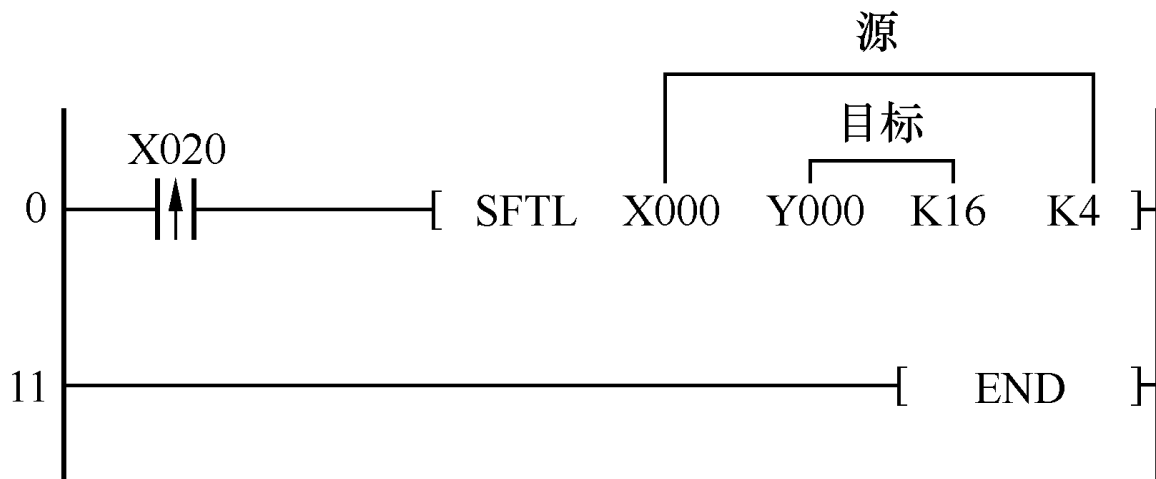
（1）S 为移位的源操作数的最低位，D 为被移位的目标操作数的最低位。n1 为目标操作数个数，n2 为源操作数个数。

（2）位左移就是源操作数从目标操作数的低位移入 n2 位，目标操作数各位向高位方向移 n2 位，目标操作数中的高 n2 位溢出。源操作数各位状态不变。

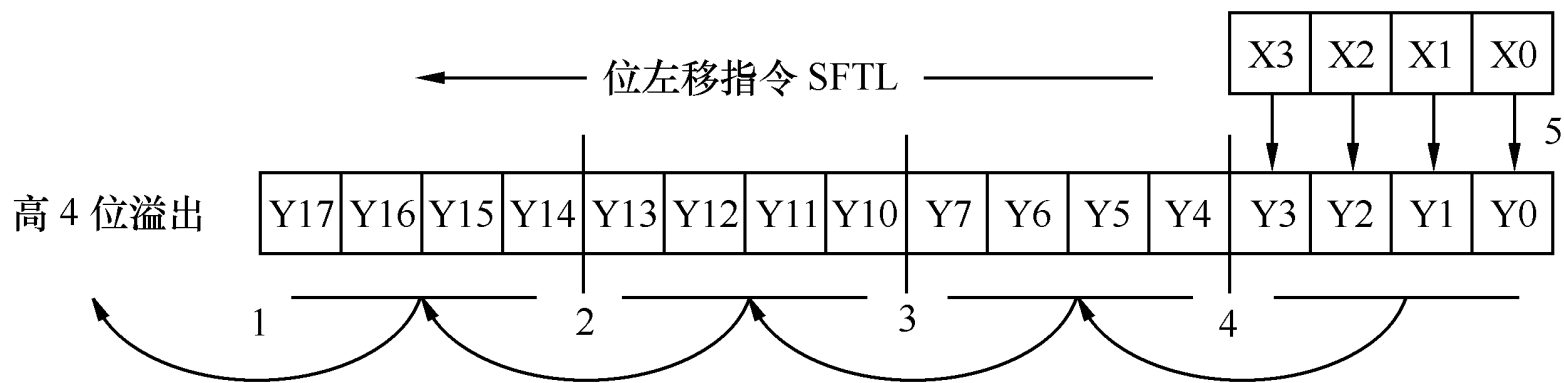
（3）在指令的连续执行方式中，每一个扫描周期都会移位一次。在实际控制中，常采用脉冲执行方式。



位左移指令 SFTL 的应用示例梯形图如图所示。



位左移指令 SFTL 示例梯形图



位左移指令 SFTL 示例过程

位左移指令 SFTL 举例

例： 位左移指令 SFTL 的程序梯形图如图 5.53 所示。设 Y17 ~ Y0 的初始状态为 0， X3 ~ X0 的位状态为 1011。求数次执行位左移指令 SFTL 后， Y17 ~ Y0 各位状态的变化。

解：Y17 ~ Y0 各位状态的变化如表 5.36 所示。第一次执行左移指令 SFTL 后，(K4Y0) = H0B，第二次执行左移指令 SFTL 后，(K4Y0) = H0BB，依次类推。

例：各位状态的变化

[illegible]

3.4 位右移指令 SFTR

SFTR 指令

循环左移指令		操 作 数			
P	FNC34 SFTR	S	X、Y、M、S	n1、n2	K、H
		D	Y、M、S		$1 \leq n2 \leq n1 \leq 1024$

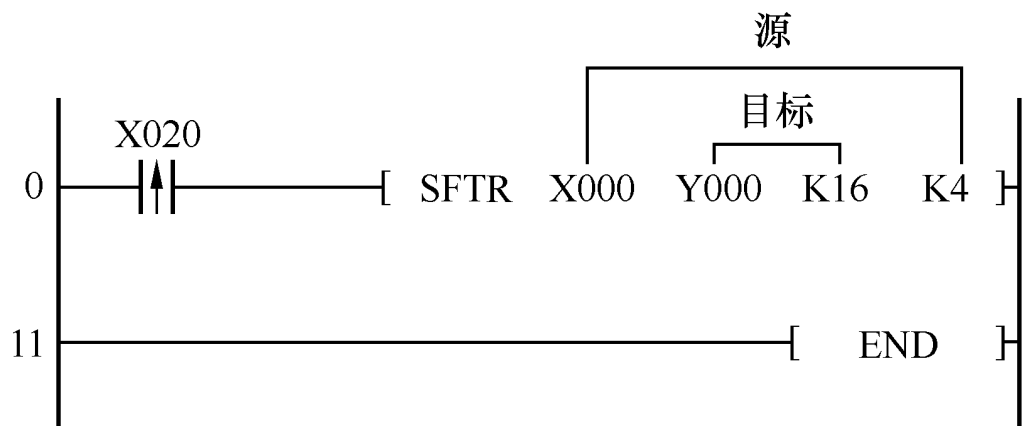
位右移指令 SFTR 的说明

（1）S 为移位的源操作数的最低位，D 为被移位的目标操作数的最低位。n1 为目标操作数个数，n2 为源操作数个数。

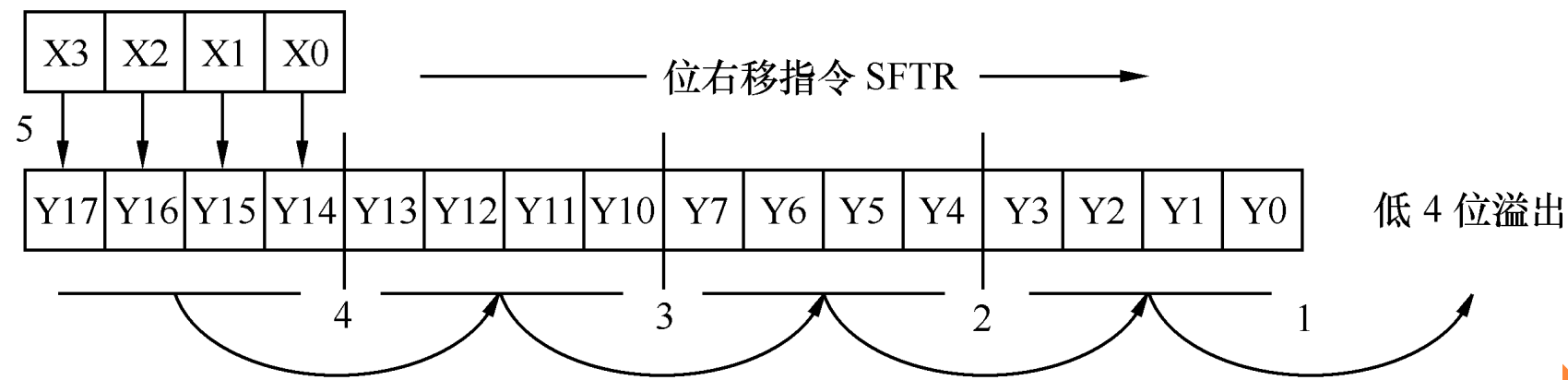
（2）位右移就是源操作数从目标操作数的高位移入 n2 位，目标操作数各位向低位方向移 n2 位，目标操作数中的低 n2 位溢出。源操作数各位状态不变。



位右移指令 SFTR 的应用示例梯形图如图所示。



位右移指令 SFTR 示例梯形图



位右移指令 SFTR 示例过程

位右移指令 SFTR 举例

例：位右移指令 SFTR 的程序梯形图如图所示。设 Y17 ~ Y0 的初始状态为 0，X3 ~ X0 的位状态为 1011。求数次执行位右移指令 SFTR 后，Y17 ~ Y0 各位状态的变化。

解：Y17 ~ Y0 各位状态的变化如表所示。在未执行位右移指令 SFTR 前，(K4Y0) = 0，第一次执行左移指令 SFTR 后，(K4Y0) = H0B000，第二次执行左移指令 SFTR 后，(K4Y0) = H0BB00，依次类推。

例：各位状态的变化

[illegible]

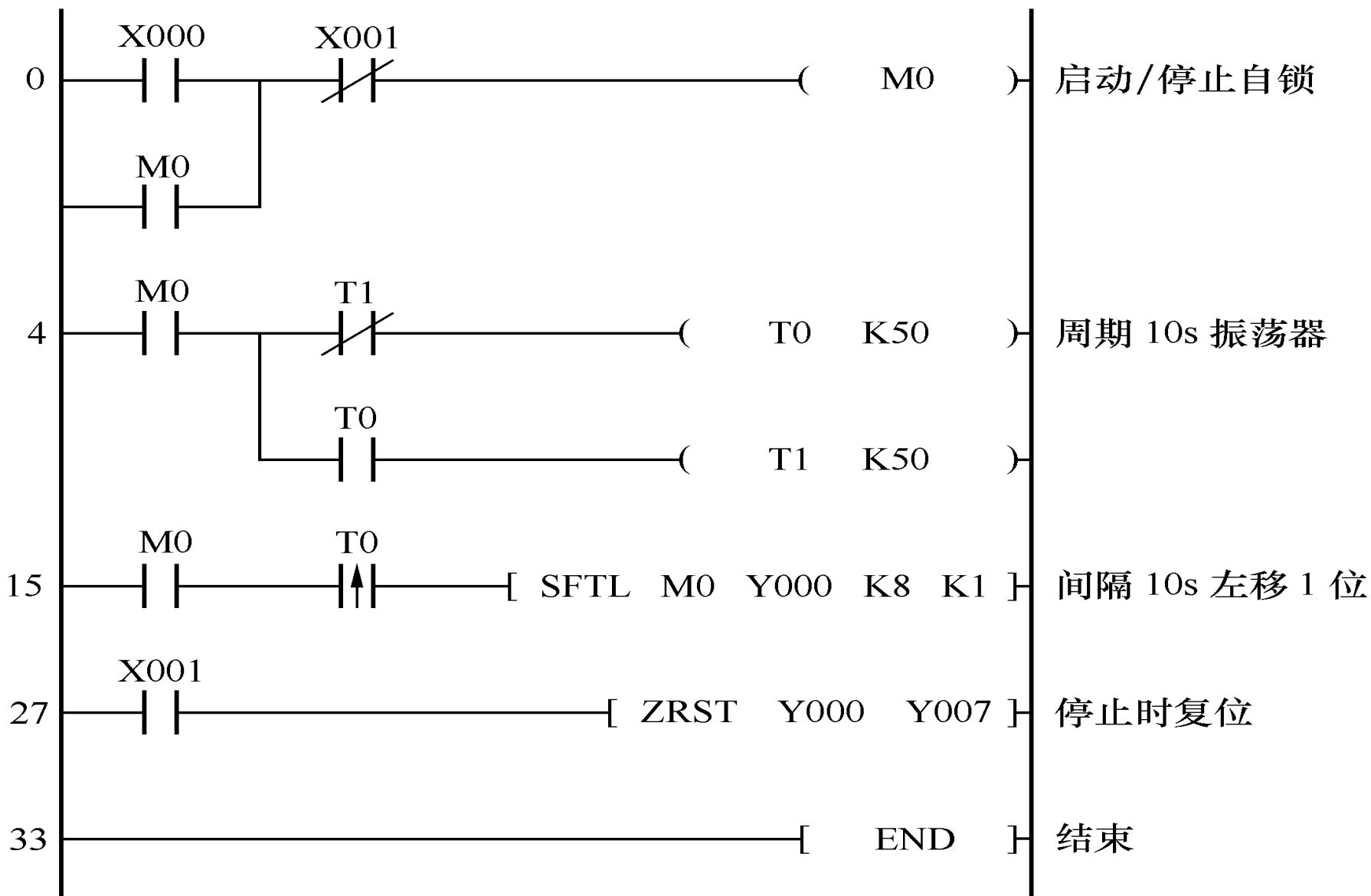
例：某台设备有 8 台电动机，为了减小电动机同时启动对电源的影响，利用位移指令实现间隔 10s 的顺序通电控制。按下停止按钮时，同时停止工作。

解： 控制线路需要 2 个输入端口， 8 个输出端口。输入、输出端口的分配如表所示。

输入 / 输出端口分配表

输 入			输 出	
输入继电器	输 入 元 件	作 用	输出继电器	控 制 对 象
X0	SB1	启动按钮	Y7 ~ Y0	8 个接触器
X1	SB2	停止按钮		





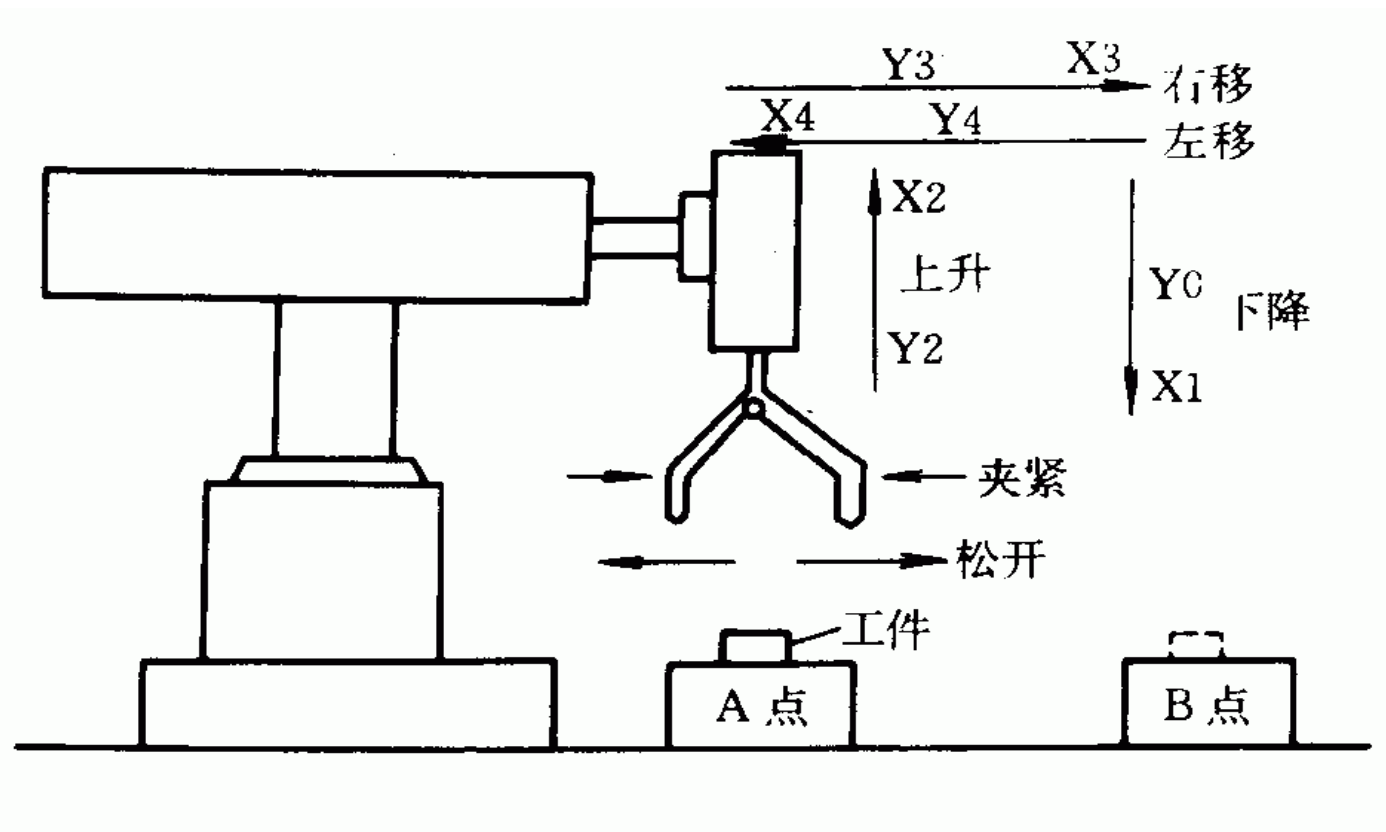
三、本次课小结

:

1. 循环指令 FOR 、 NEXT ；
2. 时钟专用的特殊辅助继电器和特殊数据寄存器；
3. 循环左移指令 ROL 、 循环右移指令 ROR;
4. 位左移指令 SFTL 、 位右移指令 SFTR ；
5. 数码显示及应用；

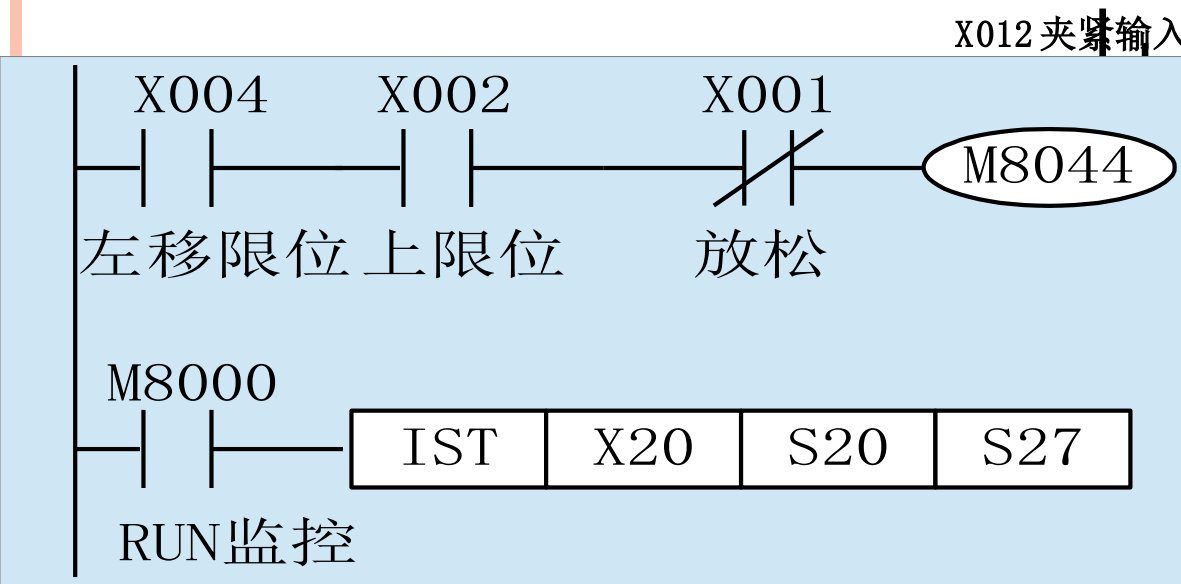


四、布置作业



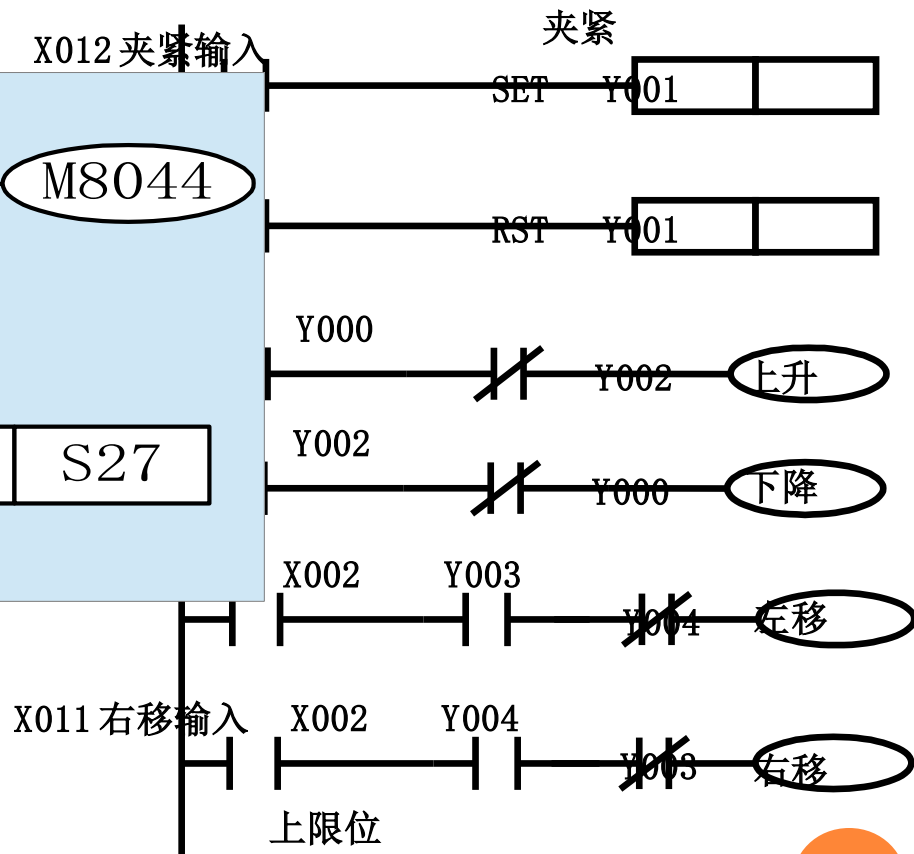
解析提示如下：





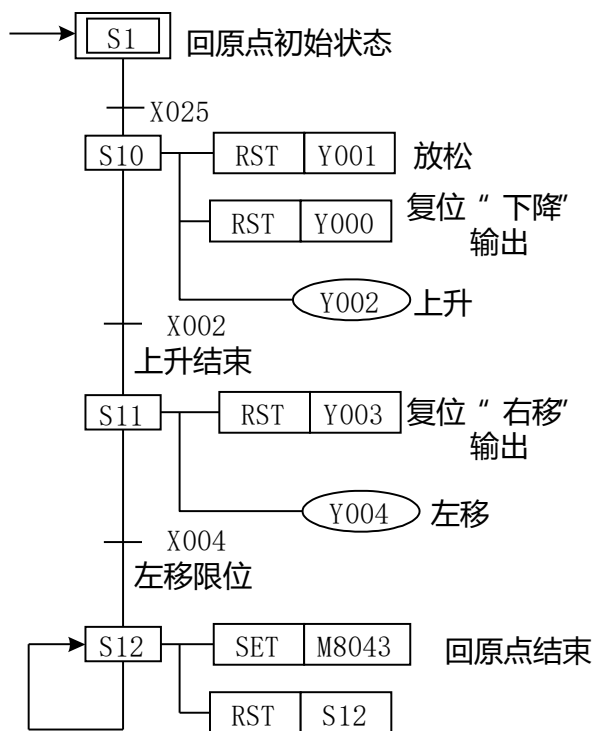
该系统的初始化程序

手动方式程序

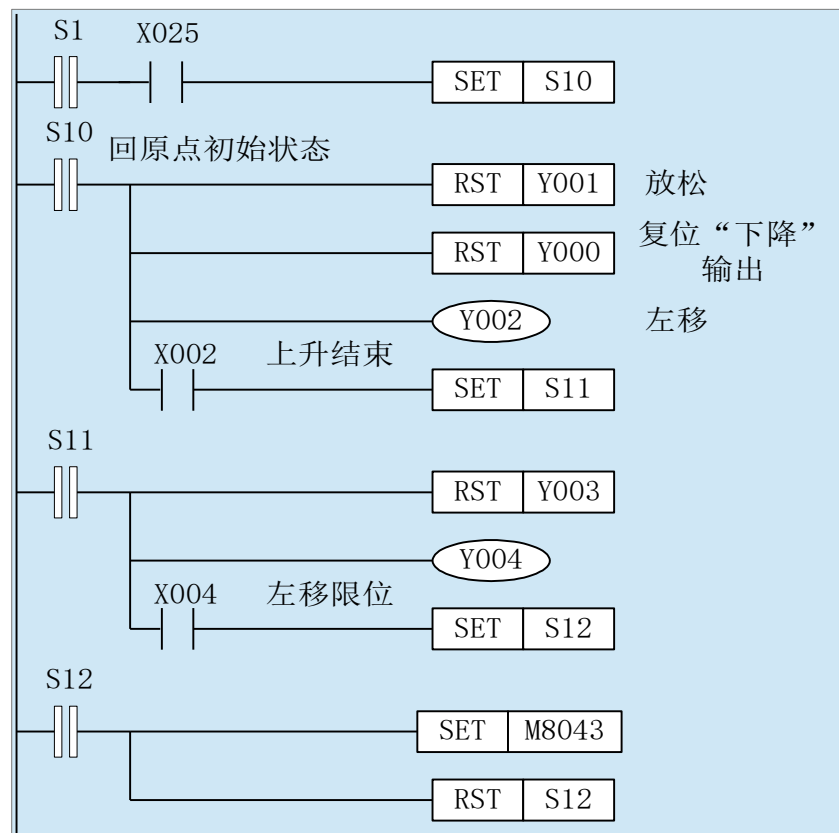


手动控制程序

○ 原点方式程序

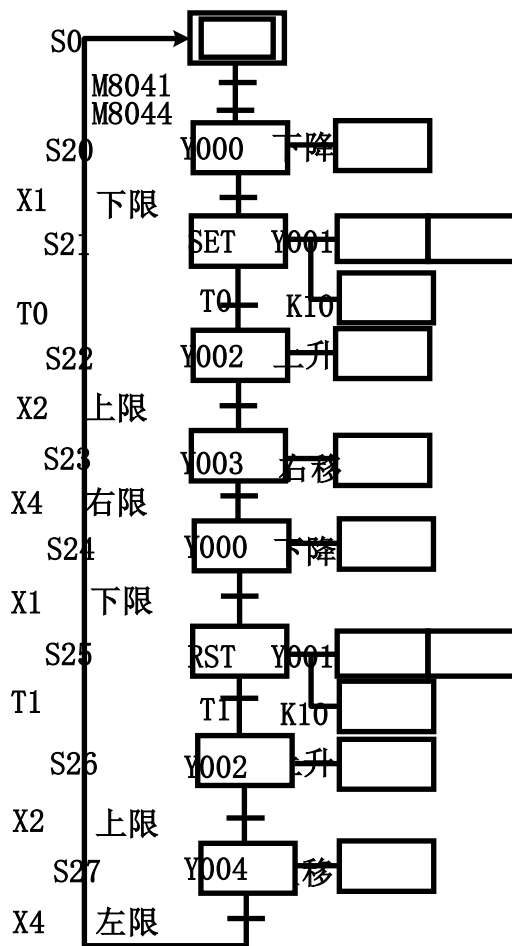


(a) 功能图



(b) 指令表

○ 自动方式程序



**谢谢观看
!**