

三菱 PLC 指令

主控指令 **MC**

主控复位指令 **MCR**

置位指令 **SET**

复位指令 **RST**

上升沿脉冲微分输出指令 **PLS**

下降沿脉冲微分输出指令 **PLF**

步进指令开始 **STL**

步进指令结束 **RET**

应用指令格式 **MEAN** 为例

三菱 FX 系列 PLC 的应用指令有连续执行型和脉冲执行型两种形式

连续执行型 **MOV**

脉冲执行型 **MOVP**

在应用指令前加 **D** 表示其处理数据为 32 位

16 位 **MOV**

32 位 **DMOV**

程序流程指令的功能是改变程序执行的顺序，主要包括条件跳转、中断、子程序调用、子程

序返回、主程序结束、警戒时钟、循环等指令。

条件跳转指令 **CJ**

子程序调用指令 **CALL**

子程序返回指令 **SRET**

中断返回指令 **IRET**

允许中断指令 **EI**

禁止中断指令 **DI**

主程序结束指令 **FEND**

刷新监视定时器指令 **WDT**

循环开始指令 **FOR**

循环结束指令 **NEXT**

传送与比较指令包括数据比较、传送、交换和变换指令，共 10 条，这些指令属于基本的

应用指令，使用较为广泛。

比较指令 **CMP**

区间比较指令 **ZCP**

传送指令 **MOV**

移位传送指令 **SMOV**

取反传送指令 **CML**

成批传送指令 **BMOV**

多点传送指令 **FMOV**

数据交换指令 **XCH**

BCD 码转换指令 **BCD**

BIN 码转换指令 **BIN**

四则运算与逻辑运算指令属于比较常用的应用指令，共有 10 条

二进制加法运算指令 **ADD**

二进制减法运算指令 **SUB**

二进制乘法运算指令 **MUL**

二进制除法运算指令 **DIV**

二进制加 1 运算指令 **INC**

二进制减 1 运算指令 **DEC**

逻辑与指令 **WAND**

逻辑或指令 **WOR**

逻辑异或指令 **WXOR**

逻辑求补指令 **NEG**

循环与移位指令有 10 条，功能号是 FNC30 ~ FNC39

循环右移指令 **ROR**

循环左移指令 **ROL**

带进位循环右移指令 **RCR**

带进位循环左移指令 **RCL**

位右移指令 **SFTR**

位左移指令 **SFTL**

字右移指令 **WSFR**

字左移指令 **WSFL**

移位写入指令 **SFWR**

移位读出指令 **SFRD**

数据处理指令有 10 条，功能号为 FNC40 ~ FNC49

成批复位指令 **ZRST**

解码指令 **DECO**

编码指令 **ENCO**

1 总数和指令 **SUM**

1 位判别指令 **BON**

平均值指令 **MEAN**

报警置位指令 **ANS**

报警复位指令 **ANR**

求平方根指令 **SQR**

二进制整数转换为浮点数指令 **FLT**

高速处理指令共有 10 条，功能号为 FNC50 ~ FNC59

输入/输出刷新指令 **REF**

输入滤波常数调整指令 **REFF**

矩阵输入指令 **MTR**

高速计数器置位指令 **HSCS**
高速计数器复位指令 **HSCR**
高速计数器区间比较指令 **HSZ**
速度检测指令 **SPD**
脉冲输出指令 **PLSY**
脉冲调制指令 **PWM**
可调速脉冲输出指令 **PLSR**

方便指令共有 10 条，功能号是 FNC60~FNC69

状态初始化指令 **IST**
数据查找指令 **SER**
绝对值式凸轮顺控指令 **ABSD**
增量式凸轮顺控指令 **INCD**
示教定时器指令 **TTMR**
特殊定时器指令 **STMR**
交替输出指令 **ALT**
斜坡信号输出指令 **RAMP**
旋转工作台控制指令 **ROTC**
数据排序指令 **SORT**

外围 I/O 设备指令共有 10 条，功能号为 FNC70~FNC79

十键输入指令 **TKY**
十六键输入指令 **HKY**
数字开关指令 **DSW**
七段译码指令 **SEGD**
带锁存的七段译码显示指令 **SEGL**
方向开关指令 **ARWS**
ASCII 码转换指令 **ASC**
ASCII 码打印输出指令 **PR**
读特殊功能模块指令 **FROM**
写特殊功能模块指令 **TO**

外围设备指令共有 8 条，用于对连接在串行通信口的外围设备进行控制，功能号是

FNC80~FNC86、FNC88

串行数据传送指令 **RS**
八进制位传送指令 **PRUN**
十六进制数转 ASCII 码指令 **ASCI**
ASCII 码转十六进制数指令 **HEX**
校验码指令 **CCD**
电位器模拟量读出指令 **VRRD**
电位器模拟量开关设定指令 **VRSC**
PID 运算指令 **PID**

浮点数（实数）运算指令

高低位变换指令只有一条，功能号为 FNC147

高低位变换指令 **SWAP**

时钟运算指令有 7 条，功能号为 160~163、166、167、169，其中 169 号指令仅适用

于 FX_{1S}、FX_{1N} 机型，不适用于 FX_{2N}、FX_{2NC} 机型。

时钟数据比较指令 **TCMP**

时钟数据区间比较指令 **TZCP**

时钟数据加法指令 **TADD**

时钟数据减法指令 **TSUB**

时钟数据读出指令 **TRD**

时钟数据写入指令 **TWR**

格雷码变换指令

有关格雷码的知识

二进制码（BIN 码）转换格雷码指令 **GRY**

格雷码转换二进制码（BIN 码）指令 **GBIN**

触点比较指令分为三类：LD*指令、AND*指令和 OR*指令

触点比较 LD*指令 **LD**

触点比较 AND*指令 **AND**

触点比较 OR*指令 **OR**

2 使用举例

MC、MCR 指令使用举例如图 4-8 所示。如果 X001 常开触点处于断开，MC 指令不执行，MC 到 MCR 之间的程序不会执行，即 0 梯级程序执行后会执行 12 梯级程序；如果 X001 触点闭合，MC 指令执行，MC 到 MCR 之间的程序会从上往下执行。

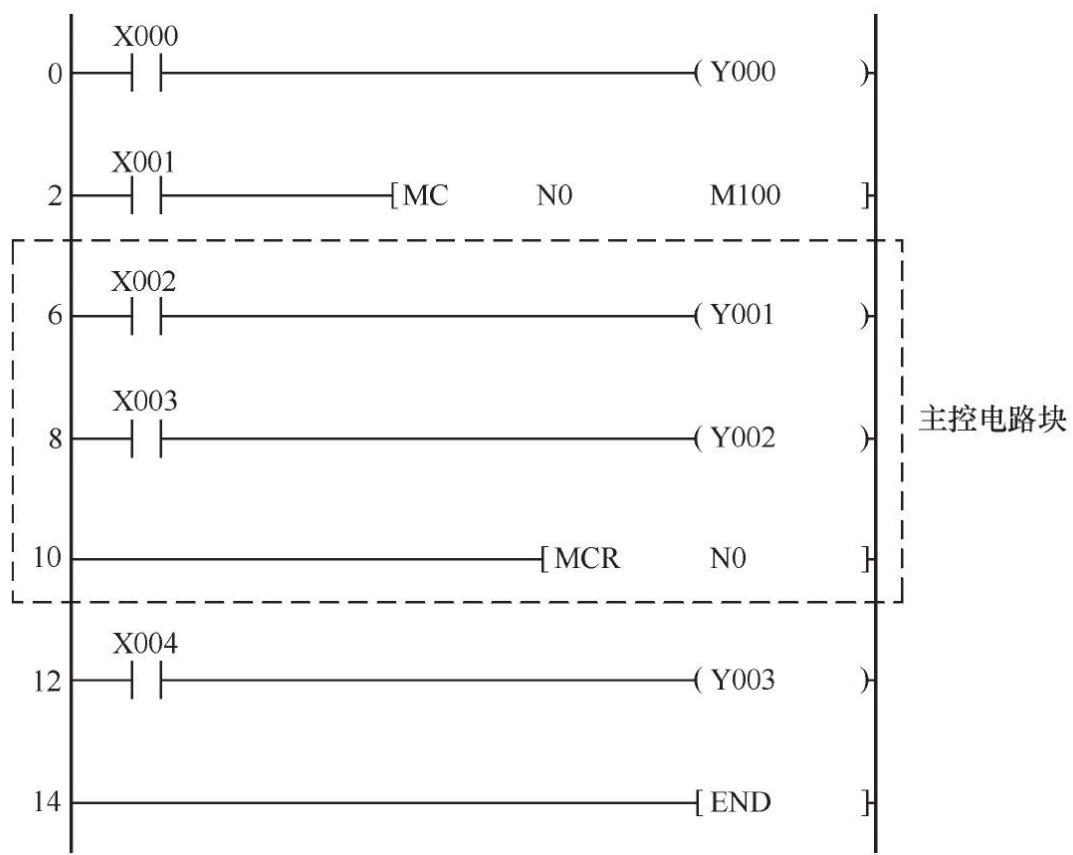


图 4-8 MC、MCR 指令使用举例

MC、MCR 指令可以嵌套使用，如图 4-9 所示，当 X001 触点闭合、X003 触点断开时，X001 触点闭合使[MC N0 M100]指令执行，N0 级电路块被启动，由于 X003 触点断开，使嵌在 N0 级内的[MC N1 M101]指令无法执行，故 N1 级电路块不会执行。

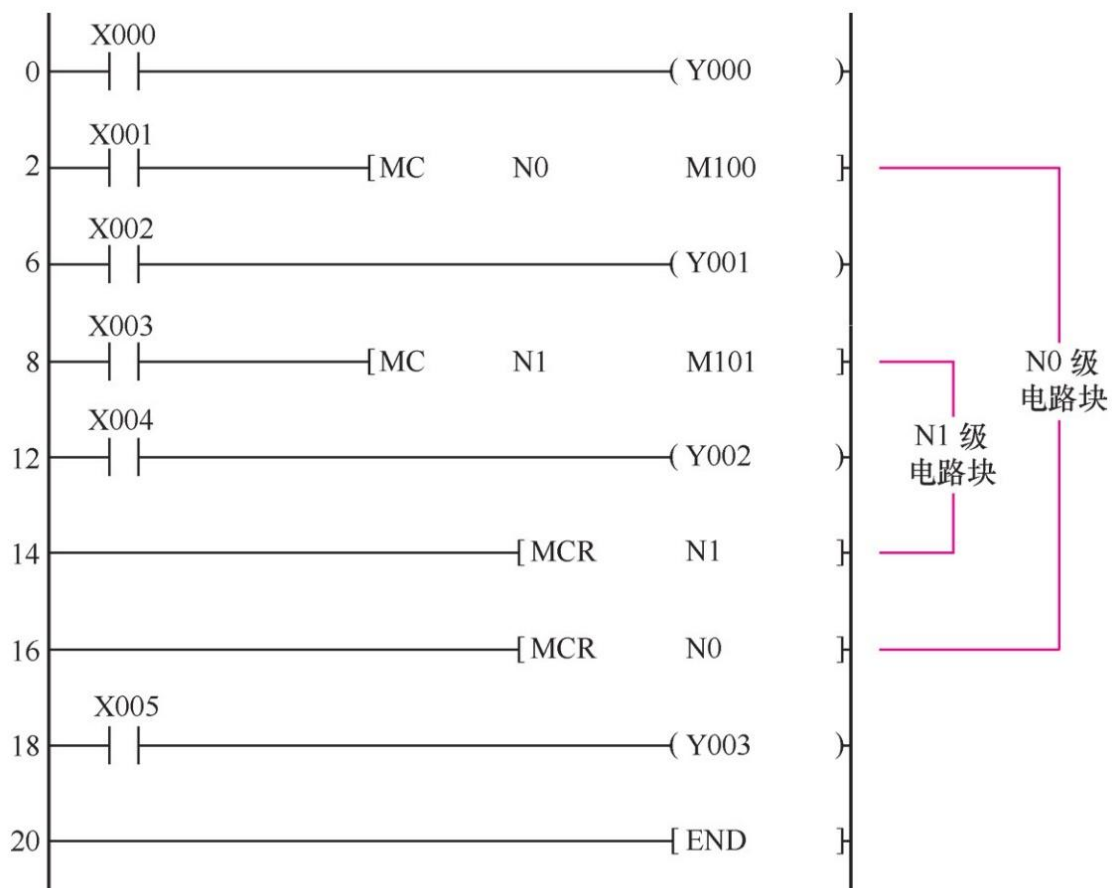


图 4-9 MC、MCR 指令的嵌套使用

如果 MC 主控指令嵌套使用，其嵌套层数允许最多 8 层（N0 ~ N7），通常按顺序从小到大使用，MC 指令的操作元件通常为输出继电器 Y 或辅助继电器 M，但不能是特殊继电器。MCR 主控复位指令的使用次数（N0 ~ N7）必须与 MC 的次数相同，在按由小到大顺序多次使用 MC 指令时，必须按由大到小相反的顺序使用 MCR 指令返回。

置位与复位指令使用举例如图 4-13 所示。

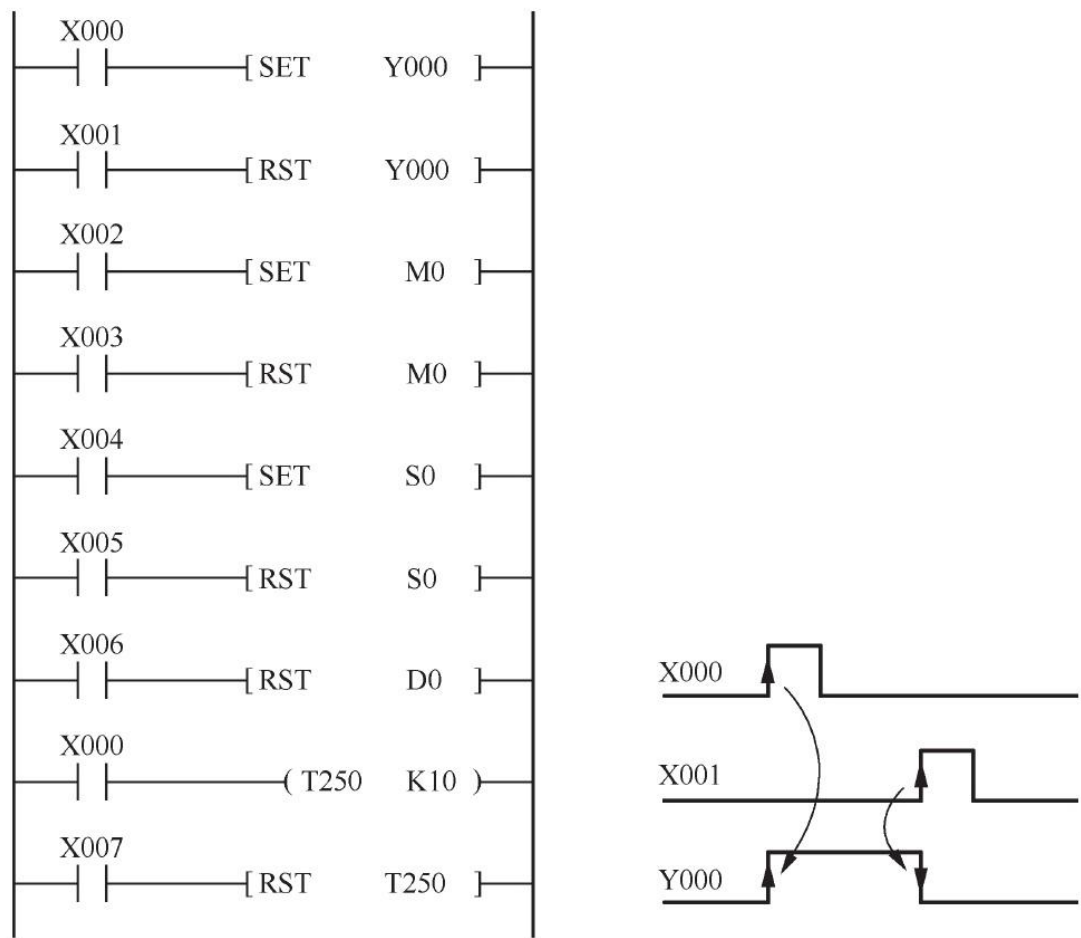


图 4-13 置位与复位指令使用举例

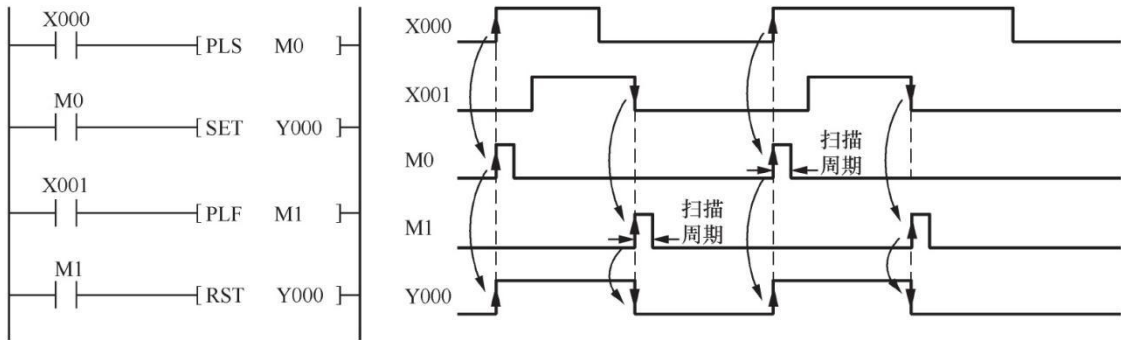
在图 4-13 中,当常开触点 X000 闭合后, Y000 线圈被置位 (状态变为 ON), X000 断开后, Y000 线圈仍维持 ON (通电) 状态;当常开触点 X001 闭合后, Y000 线圈被复位 (状态变为 OFF), X001 断开后, Y000 线圈维持 OFF (失电) 状态。

对于同一元件, SET、RST 指令可反复使用, 顺序也可随意, 但最后执行者有效。

指令名称及助记符	功能	对象软元件
上升沿脉冲微分输出指令 PLS	其功能是当检测到输入脉冲上升沿来时，使操作元件得电一个扫描周期	Y、M
下降沿脉冲微分输出指令 PLF	其功能是当检测到输入脉冲下降沿来时，使操作元件得电一个扫描周期	Y、M

2 使用举例

PLS、PLF 指令使用举例如图 4-14 所示。当常开触点 X000 由断开转为闭合时，一个上升沿脉冲加到[PLS M0]，指令执行，M0 线圈得电一个扫描周期，M0 常开触点闭合，[SET Y000]指令执行，将 Y000 线圈置位（即让 Y000 线圈得电并保持）；当常开触点 X001 由闭合转为断开时，一个脉冲下降沿加给[PLF M1]，指令执行，M1 线圈得电一个扫描周期，M1 常开触点闭合，[RST Y000]指令执行，将 Y000 线圈复位（即让 Y000 线圈失电）。



PLC 顺序控制需要用到步进指令,三菱 FX_{2N} 系列 PLC 有两条步进指令:STL 和 RET。

1 指令名称与功能

指令名称及功能如表 5-1 所示。

表 5-1 指令名称及功能

指令名称（助记符）	功能
STL	步进开始指令，其功能是将步进接点接到左母线，该指令的操作元件为状态继电器 S
RET	步进结束指令，其功能是将子母线返回到左母线位置，该指令无操作元件

2 使用举例

(1) STL 指令使用

STL 指令使用举例如图 5-2 所示，其中图 5-2（a）为梯形图，图 5-2（b）为其对应的指令语句表。状态继电器 S 只有常开触点，没有常闭触点，在绘制梯形图时，输入指令 [STL S20] 即能生成 S20 常开触点，S 常开触点闭合后，其右端相当于子母线，与子母线直接连接的线圈可以直接用 OUT 指令，相连的其他元件可用基本指令写出指令语句表，如触点用 LD 或 LDI 指令。

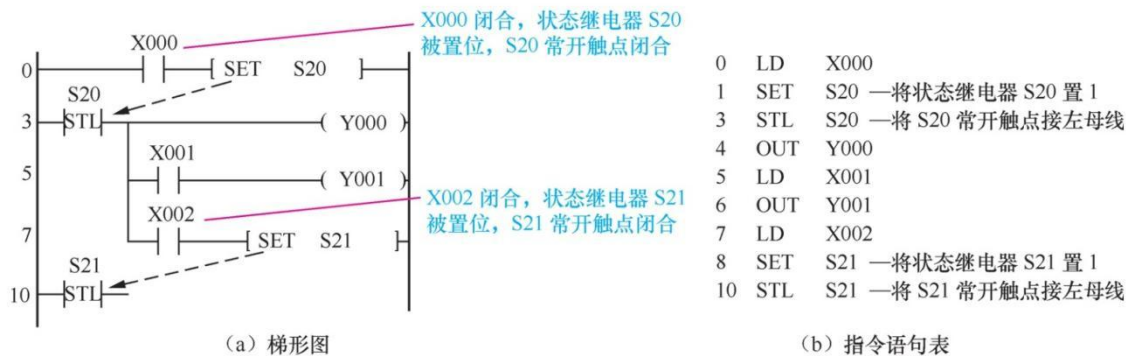


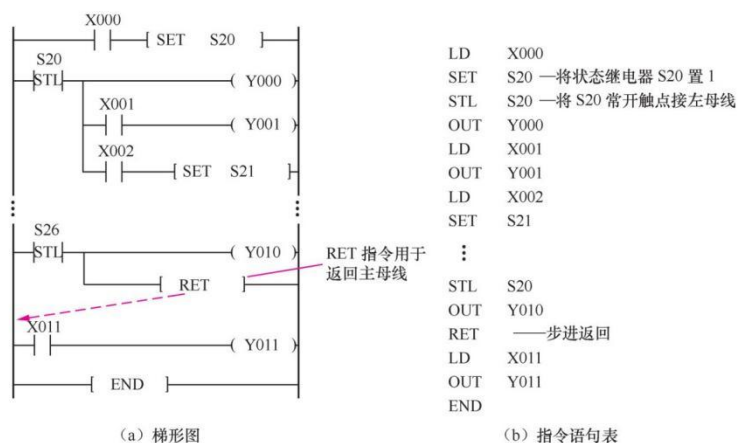
图 5-2 STL 指令使用举例

梯形图说明如下。

当 X000 常开触点闭合时 → 执行 [SET S20] 指令 → 状态继电器 S20 被置 1 (置位) → S20 常开触点闭合 → Y000 线圈得电; 若 X001 常开触点闭合, Y001 线圈也得电; 若 X002 常开触点闭合, 执行 [SET S21] 指令, 状态继电器 S21 被置 1 → S21 常开触点闭合。

(2) RET 指令使用

RET 指令使用举例如图 5-3 所示, 其中图 5-3 (a) 为梯形图, 图 5-3 (b) 为对应的指令语句表。RET 指令通常用在一系列步进指令的最后, 表示状态流程的结束并返回主母线。



状态转移图的分支方式主要有：单分支方式、选择性分支方式和并行分支方式。图 5-1

(b) 的状态转移图为单分支，程序由前往后依次执行，中间没有分支，不复杂的顺序控制常采用这种单分支方式。较复杂的顺序控制可采用选择性分支方式或并行分支方式。

1 选择性分支方式

选择性分支状态转移图如图 5-5 (a) 所示，在状态器 S21 后有两个可选择的分支，当 X1 闭合时执行 S22 分支，当 X4 闭合时执行 S24 分支，如果 X1 较 X4 先闭合，则只执行 X1 所在的分支，X4 所在的分支不执行。图 5-5 (b) 是依据图 5-5 (a) 画出的梯形图，图 5-5 (c) 则为对应的指令语句表。

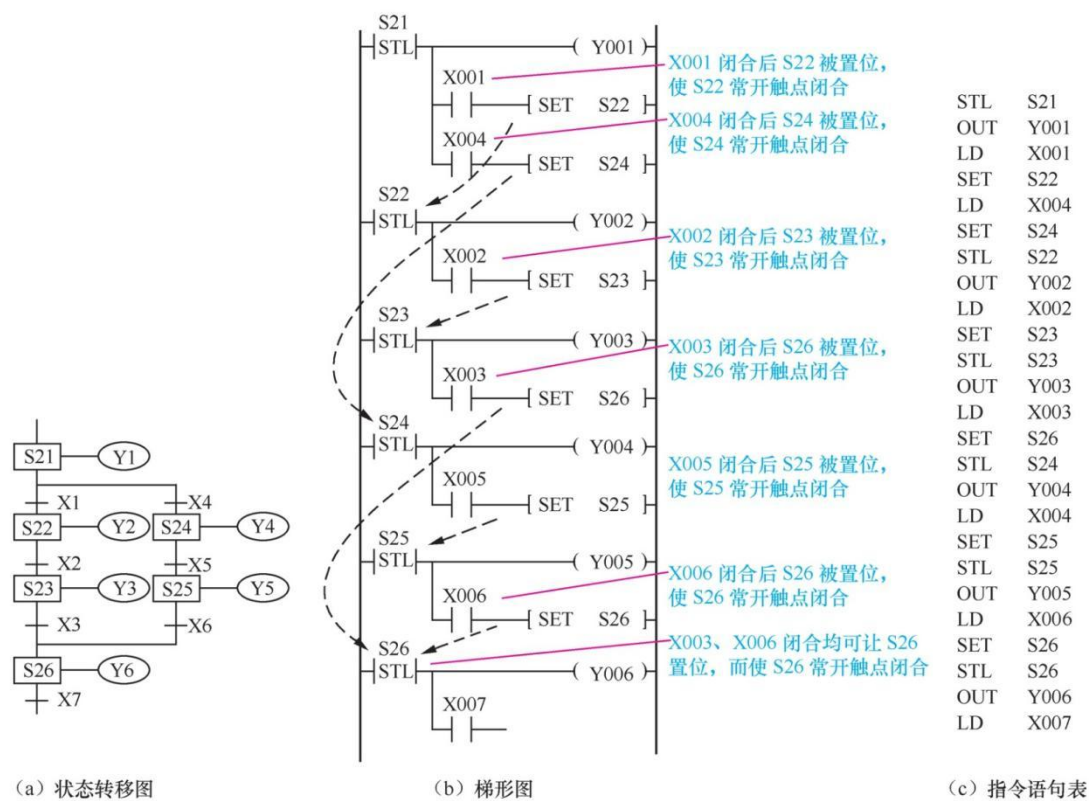


图 5-5 选择性分支方式

三菱 FX 系列 PLC 最多允许有 8 个可选择的分支。

2 并行分支方式

并行分支方式状态转移图如图 5-6 (a) 所示, 在状态器 S21 后有两个并行的分支, 并行分支用双线表示, 当 X1 闭合时 S22 和 S24 两个分支同时执行, 当两个分支都执行完成并且 X4 闭合时才能往下执行, 若 S23 或 S25 任一条分支未执行完, 即使 X4 闭合, 也不会执行到 S26。图 5-6 (b) 是依据图 5-6 (a) 画出的梯形图, 图 5-6 (c) 则为对应的指令语句表。

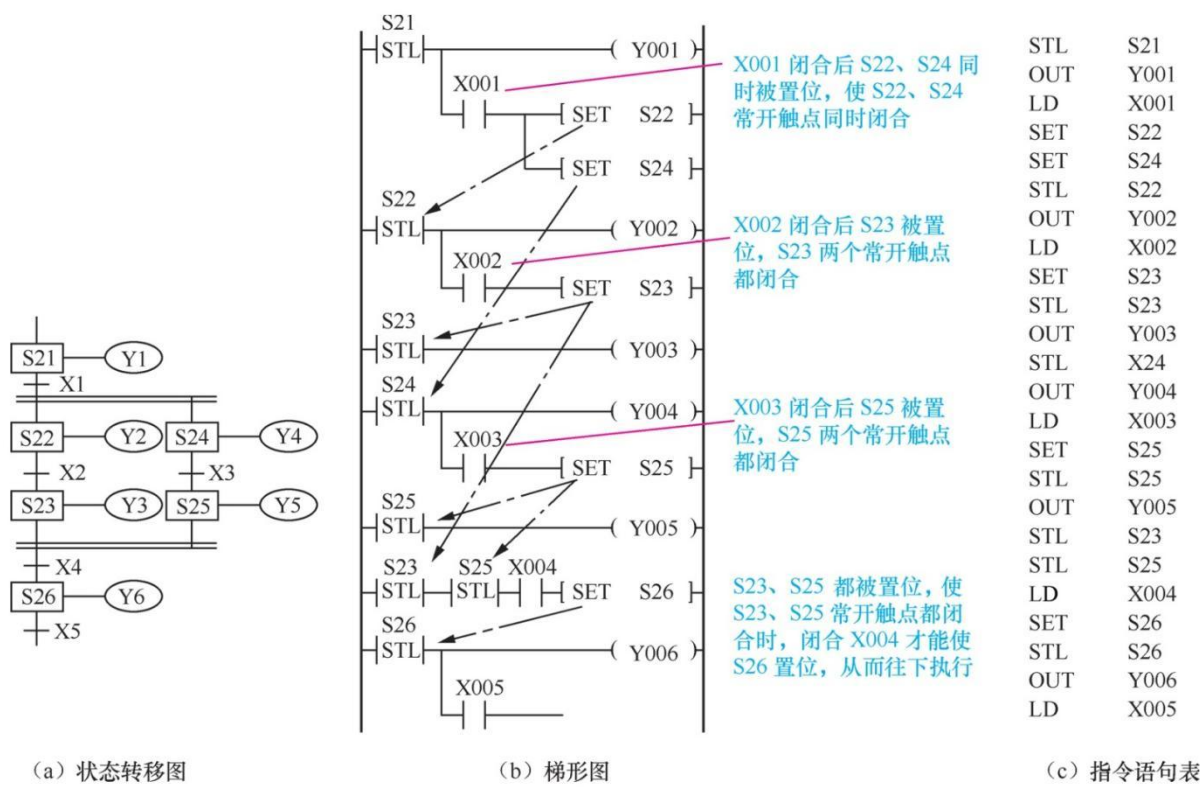


图 5-6 并行分支方式

三菱 FX 系列 PLC 最多允许有 8 个并行的分支。

5.1.5 用步进指令编程注意事项

在使用步进指令编写顺序控制程序时，要注意以下事项。

① 初始状态（S0）应预先驱动，否则程序不能向下执行，驱动初始状态通常用控制系统的初始条件，若无初始条件，可用 M8002 或 M8000 触点进行驱动。

② 不同步程序的状态继电器编号不要重复。

③ 当上一个步程序结束，转移到下一个步程序时，上一个步程序中的元件会自动复位（SET、RST 指令作用的元件除外）。

④ 在步进顺序控制梯形图中可使用双线圈功能，即在不同步程序中可以使用同一个输出线圈，这是因为 CPU 只执行当前处于活动步的步程序。

⑤ 同一编号的定时器不要在相邻的步程序中使用，不是相邻的步程序中则可以使用。

⑥ 不能同时动作的输出线圈尽量不要设在相邻的步程序中，因为可能出现下一步程序开始执行时上一步程序未完全复位，这样会出现不能同时动作的两个输出线圈同时动作，如果必须这样做，可以在相邻的步程序中采用软联锁保护，即给一个线圈串联另一个线圈的常闭触点。

⑦ 在步程序中可以使用跳转指令。在中断程序和子程序中也不能存在步程序。在步程序中最多可以有 4 级 FOR\NEXT 指令嵌套。

⑧ 在选择分支和并行分支程序中，分支数最多不能超过 8 条，总的支路数不能超过 16 条。

⑨ 如果希望在停电恢复后继续维持停电前的运行状态时，可使用 S500 ~ S899 停电保持型状态继电器。

应用指令由功能助记符、功能号和操作数等组成。应用指令的格式如表 6-1 所示（以平均值指令为例）。

表 6-1 应用指令的格式

指令名称	助记符	功能号	操作数		
			源操作数 (S)	目标操作数 (D)	其他操作数 (n)
平均值指令	MEAN	FNC45	KnX KnY KnS KnM T、C、D	KnX KnY KnS KnM T、C、D、V、Z	Kn 、 Hn $n=1 \sim 64$

应用指令格式说明如下。

(1) 助记符

助记符用来规定指令的操作功能，一般由字母（英文单词或单词缩写）组成。上面的「MEAN」为助记符，其含义是对操作数取平均值。

(2) 功能号

它是应用指令的代码号，每个应用指令都有自己的功能号，如 MEAN 指令的功能号为 FNC45，在编写梯形图程序时，如果要使用某应用指令，须输入该指令的助记符，而采用手持编程器编写应用指令时，要输入该指令的功能号。

(3) 操作数

操作数又称操作元件，通常由源操作数[S]、目标操作数[D]和其他操作数[n]组成。

操作数中的 K 表示十进制数，H 表示十六进制数，n 为常数，X 为输入继电器，Y 为输出继电器、S 为状态继电器，M 为辅助继电器，T 为定时器，C 为计数器，D 为数据寄存器，V、Z 为变址寄存器。

如果源操作数和目标操作数不止一个，可分别用[S1]、[S2]、[S3]和[D1]、[D2]、[D3]

表示。

举例：在图 6-1 中，指令的功能是在常开触点 X000 闭合时，将十进制数 100 送入数据寄存器 D10 中。



图 6-1 应用指令格式说明

三菱 FX 系列 PLC 的应用指令有连续执行型和脉冲执行型两种形式。图 6-2 (a) 中的 MOV 为连续执行型应用指令，当常开触点 X000 闭合后，[MOV D10 D12]指令在每个扫描周期都被重复执行。图 6-2 (b) 中的 MOVP 为脉冲执行型应用指令（在 MOV 指令后加 P 表示脉冲执行），[MOVP D10 D12]指令仅在 X000 由断开转为闭合瞬间执行（闭合后不执行）。

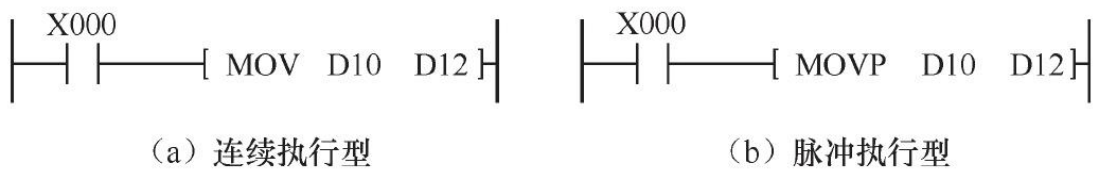


图 6-2 两种执行形式的应用指令

2 数据长度

应用指令可处理 16 位和 32 位数据。

(1) 16 位数据

数据寄存器 D 和计数器 C0~C199 存储的为 16 位数据，16 位数据结构如图 6-3 所示，其中最高位为符号位，其余为数据位，符号位的功能是指示数据位的正负，符号位为 0 表示数据位的数据为正数，符号位为 1 表示数据位的数据为负数。

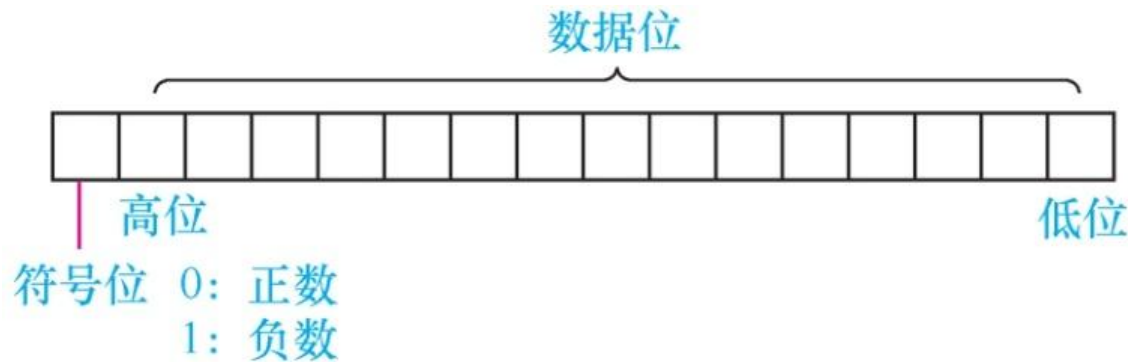


图 6-3 16 位数据结构

(2) 32 位数据

一个数据寄存器可存储 16 位数据，相邻的两个数据寄存器组合起来可以存储 32 位数据。32 位数据结构如图 6-4 所示。

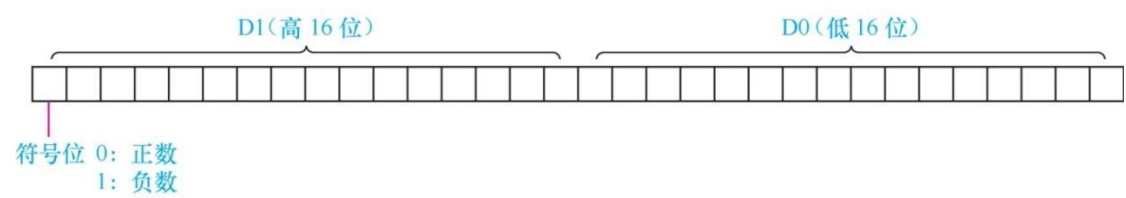


图 6-4 32 位数据结构

在应用指令前加 D 表示其处理数据为 32 位，在图 6-5 中，当常开触点 X000 闭合时，MOV 指令执行，将数据寄存器 D10 中的 16 位数据送入数据寄存器 D12，当常开触点 X001 闭合时，DMOV 指令执行，将数据寄存器 D20 和 D21 中的 16 位数据拼成 32 位送入数据寄存器 D22 和 D23，其中 D20→D22，D21→D23。脉冲执行符号 P 和 32 位数据处理符号 D 可同时使用。

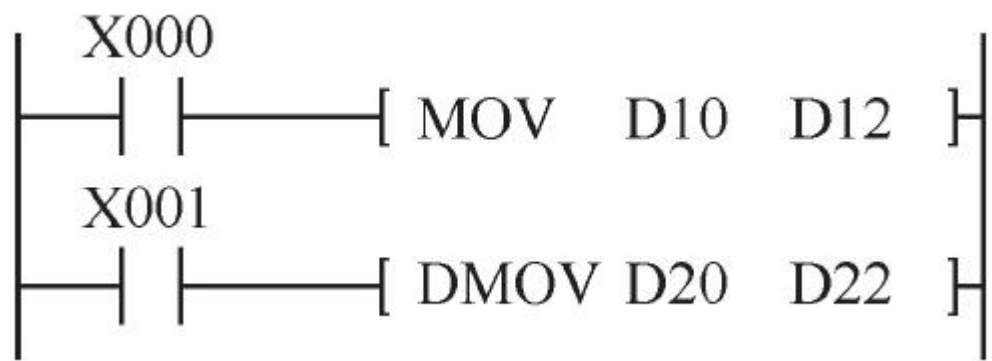


图 6-5 16 位和 32 位数据执行指令使用说明

(3) 字元件和位元件

字元件是指处理数据的元件，如数据寄存器和定时器、计数器都为字元件。位元件是指只有断开和闭合两种状态的元件，如输入继电器 X、输出继电器 Y、辅助继电器 M 和状态继电器 S 都为位元件。

多个位元件组合可以构成字元件，位元件在组合时通常 4 个元件组成一个单元，位元件组合可用 Kn 加首元件来表示， n 为单元数，如 K1M0 表示 M0~M3 4 个位元件组合，K4M0 表示位元件 M0~M15 组合成 16 位字元件（M15 为最高位，M0 为最低位），K8M0 表示位元件 M0~M31 组合成 32 位字元件。其他的位元件组成字元件如 K4X0、K2Y10、K1S10 等。

在进行 16 位数据操作时， n 为 1~3，参与操作的位元件只有 4~12 位，不足的部分用 0 补足，由于最高位只能为 0，所以意味着只能处理正数。在进行 32 位数据操作时， n 为 1~7，参与操作的位元件有 4~28 位，不足的部分用 0 补足。在采用「 Kn + 首元件编号」方式组合成字元件时，首元件可以任选，但为了避免混乱，通常选尾数为 0 的元件作首元件，如 M0、M10、M20 等。

不同长度的字元件在进行数据传递时，一般按以下规则。

- ① 长字元件 → 短字元件传递数据，长字元件低位数据传送给短字元件。
- ② 短字元件 → 长字元件传递数据，短字元件数据传送给长字元件低位，长字元件高位全部变为 0。

3 变址寄存器

三菱 FX 系列 PLC 有 V、Z 两种 16 位变址寄存器，它可以像数据寄存器一样进行读写操作。变址寄存器 V、Z 编号分别为 V0~V7、Z0~Z7，常用在传送、比较指令中，

用来修改操作对象的元件号。例如，在图 6-6 (a) 梯形图中，如果 V0=18（即变址寄存器 V 中存储的数据为 18）、Z0=20，那么 D2V0 表示 $D(2+V0) = D20$ ，D10Z0 表示 $D(10+Z0) = D30$ ，指令执行的操作是将数据寄存器 D20 中数据送入 D30 中，因此图 6-6 (a) 和图 6-6 (b) 中两个梯形图的功能是等效的。

变址寄存器可操作的元件有输入继电器 X、输出继电器 Y、辅助继电器 M、状态继电器 S、指针 P 和由位元件组成的字元件的首元件，如 $KnM0Z$ ，但变址寄存器不能改变 n 的值，如 $K2ZM0$ 是错误的。利用变址寄存器在某些方面可以使编程简化。图 6-6 (c) 中的程序采用了变址寄存器，在常开触点 X000 闭合时，先分别将数据 6 送入变址寄存器 V0 和 Z0，然后将数据寄存器 D6 中的数据送入 D16。

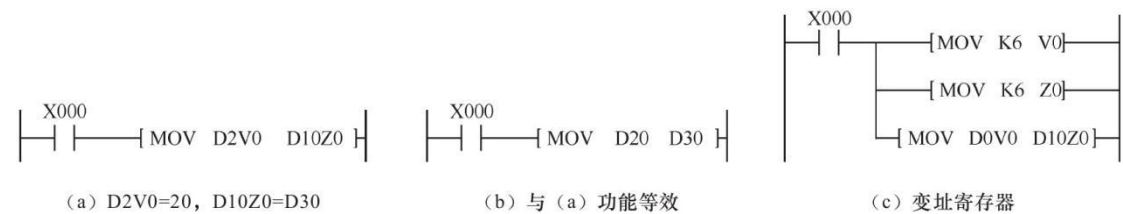


图 6-6 变址寄存器的使用说明

1 条件跳转指令 (CJ)

(1) 指令格式

条件跳转指令格式如表 6-2 所示。

表 6-2 条件跳转指令格式

指令名称	助记符 (功能号)	指令形式	操作数
			D
条件跳转指令	CJ (FNC00)		P0 ~ P63 (FX _{1S}) P0 ~ P127 (FX _{1N} \FX _{1NC} \FX _{2N} \FX _{2NC}) P0 ~ P2047 (FX _{3G}) P0 ~ P4095 (FX _{3U} \FX _{3UC})

(2) 使用说明

CJ 指令的使用说明如图 6-7 所示。在图 6-7 (a) 中，当常开触点 X020 闭合时，执行 [CJ P9] 指令，程序会跳转到 CJ 指令指定的标号（指针）P9 处，并从该处开始执行程序，跳转指令与标记之间的程序将不会执行，如果 X020 处于断开状态，程序则不会跳转，而是往下执行，当执行到常开触点 X021 所在行时，若 X021 处于闭合，CJ 指令执行会使程序跳转到 P9 处。在图 6-7 (b) 中，当常开触点 X022 闭合时，CJ 指令执行会使程序跳转到 P10 处，并从 P10 处往下执行程序。

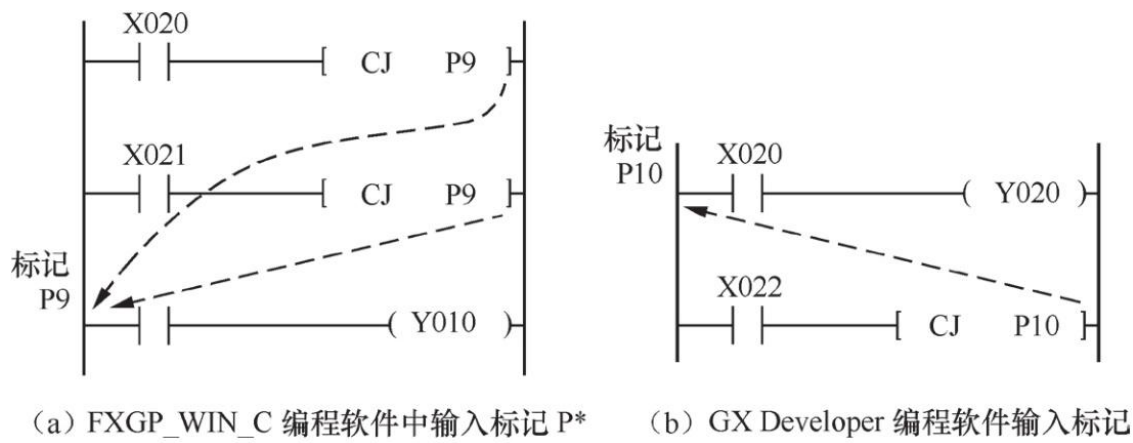
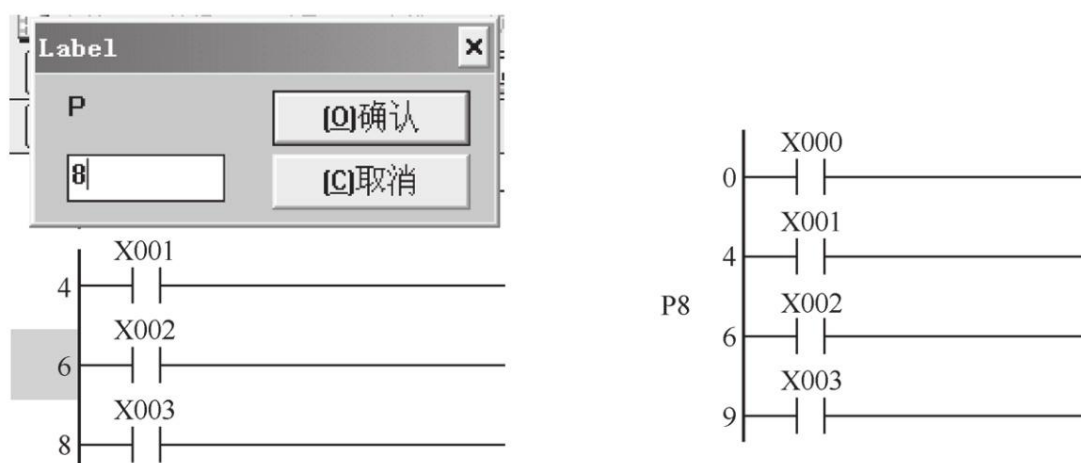
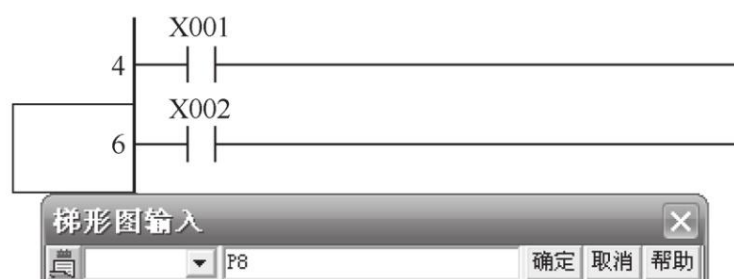


图 6-7 CJ 指令的使用说明

在 FXGP_WIN-C 编程软件中输入标记 P*的操作说明如图 6-8 (a) 所示，将光标移到某程序左母线步标号处，然后按键盘上的「P」键，在弹出的对话框中输入数字，单击「确定」按钮即输入标记。在 GX Developer 编程软件中输入标记 P*的操作说明如图 6-8 (b) 所示，在程序左母线步标号处双击，弹出「梯形图输入」对话框，输入标记号，单击「确定」按钮即可。



(a) 在 FXGP/WIN-C 编程软件中输入标记



(b) 在 GX Developer 编程软件中输入标记

2 子程序调用（CALL）和返回（SRET）指令

(1) 指令格式

子程序调用和返回指令格式如表 6-3 所示。

表 6-3 子程序调用和返回指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
子程序调用指令	CALL FNC01		P0 ~ P63 (FX _{1S}) P0 ~ P127 (FX _{1N} \FX _{1NC} \FX _{2N} \FX _{2NC}) P0 ~ P2047 (FX _{3G}) P0 ~ P4095 (FX _{3U} \FX _{3UC}) (嵌套 5 级)
子程序返回指令	SRET FNC02		无

(2) 使用说明

子程序调用和返回指令的使用如图 6-9 所示。当常开触点 X001 闭合时, 执行 [CALL P11] 指令, 程序会跳转并执行标记 P11 处的子程序 1, 如果常开触点 X002 闭合, 执行 [CALL P12] 指令, 程序会跳转并执行标记 P12 处的子程序 2, 子程序 2 执行到返回指令 [SRET] 时, 会跳转到子程序 1, 而子程序 1 通过其 [SRET] 指令返回主程序。从图 6-9 中可以看出, 子程序 1 中包含有跳转到子程序 2 的指令, 这种方式称为嵌套。

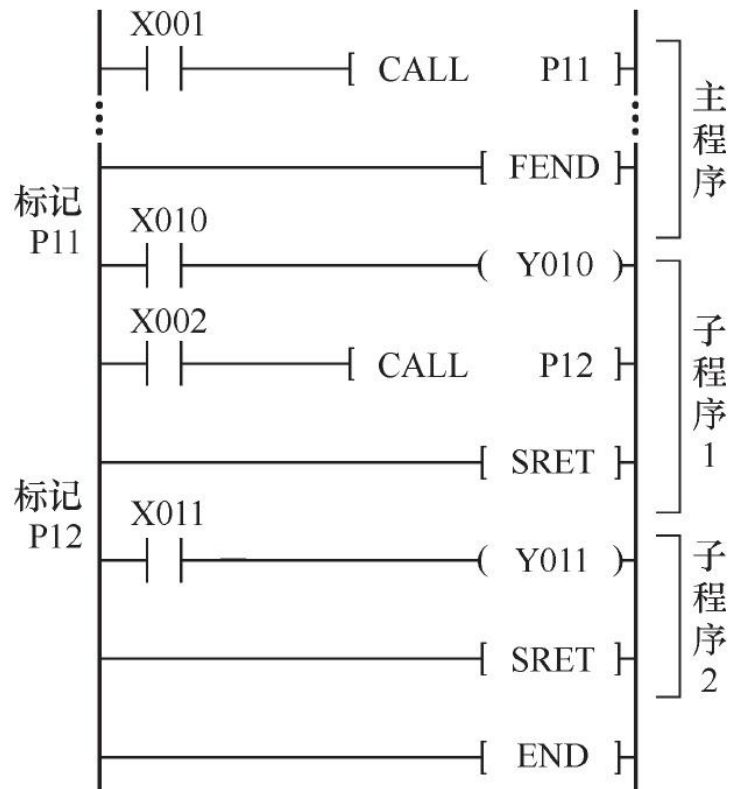


图 6-9 子程序调用和返回指令的使用

在使用子程序调用和返回指令时要注意以下几点。

- ① 一些常用或多次使用的程序可以写成子程序，然后进行调用。
- ② 子程序要求写在主程序结束指令 [FEND] 之后。
- ③ 子程序中可使用嵌套，嵌套最多为 5 级。
- ④ CALL 指令和 CJ 的操作数不能为同一标记，但不同嵌套的 CALL 指令可调用同一标记处的子程序。
- ⑤ 在子程序中，要求使用定时器 T192 ~ T199 和 T246 ~ T249

3 中断指令

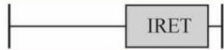
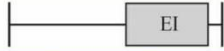
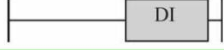
在生活中，人们经常会遇到这样的情况：当你正在书房看书时，突然客厅的电话响了，你就会停止看书，转而去接电话，接完电话后又接着去看书。这种停止当前工作，转而去做其他工作，做完后又返回来做先前工作的现象称为中断。

PLC 也有类似的中断现象，当 PLC 正在执行某程序时，如果突然出现意外事情（中断输入），它就需要停止当前正在执行的程序，转而去处理意外事情（即去执行中断程序），处理完后又接着执行原来的程序。

(1) 指令格式

中断指令有三条，其格式如表 6-4 所示。

表 6-4 中断三条指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
中断返回指令	IRET FNC03		无
允许中断指令	EI FNC04		无
禁止中断指令	DI FNC05		无

(2) 指令说明及使用说明

中断指令的使用如图 6-10 所示，下面对照该图来说明中断指令的使用要点。

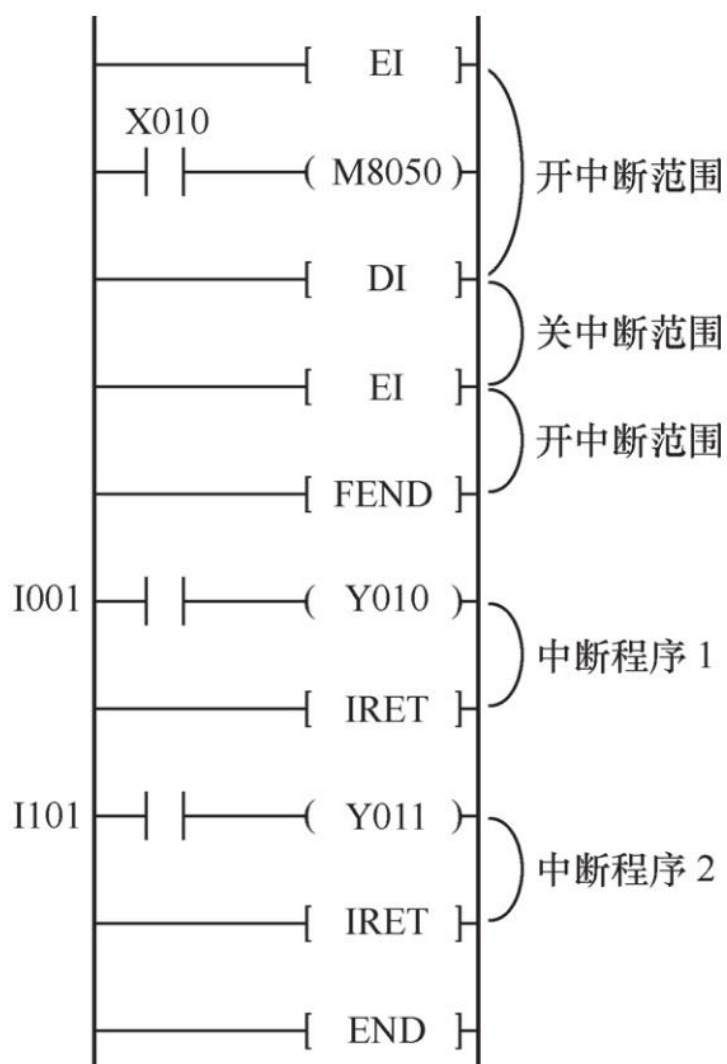
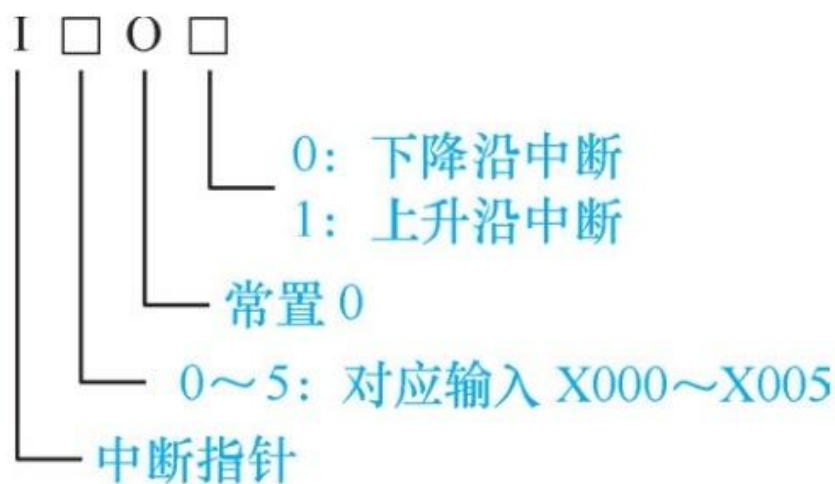


图 6-10 中断指令的使用

① 中断允许。EI 至 DI 指令之间或 EI 至 FEND 指令之间为中断允许范围，即程序运行到此处时，如果有中断输入，程序马上跳转执行相应的中断程序。

② 中断禁止。DI 至 EI 指令之间为中断禁止范围，当程序在此范围内运行时出现中断输入，不会马上跳转执行中断程序，而是将中断输入保存下来，等到程序运行完 EI 指令时才跳转执行中断程序。

③ 输入中断指针。图中标号处的 I001 和 I101 为中断指针，其含义如下。



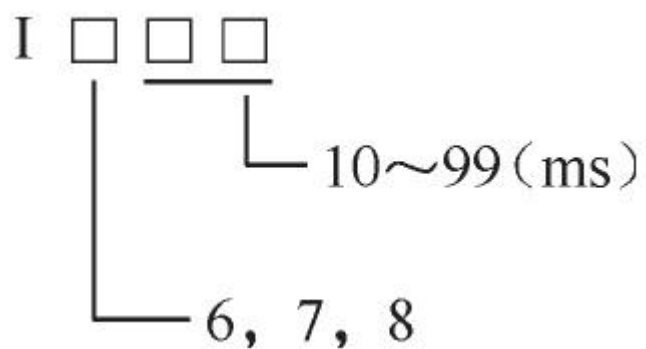
三菱 FX 系列 PLC 可使用 6 个输入中断指针，表 6-5 列出了这些输入中断指针编号和相关内容。

表 6-5 三菱 FX 系列 PLC 的中断指针编号和相关内容

中断输入	指针编号		禁止中断
	上升中断	下降中断	
X000	I001	I000	M8050
X001	I101	I100	M8051
X002	I201	I200	M8052
X003	I301	I300	M8053
X004	I401	I400	M8054
X005	I501	I500	M8055

对照表 6-5 不难理解图 6-10 梯形图的工作原理：当程序运行在中断允许范围内时，若 X000 触点由断开转为闭合 OFF→ON（如 X000 端子外接按钮闭合），程序马上跳转执行中断指针 I001 处的中断程序，执行到 [IRET] 指令时，程序又返回主程序；当程序从 EI 指令往 DI 指令运行时，若 X010 触点闭合，特殊辅助继电器 M8050 得电，则将中断输入 X000 设为无效，这时如果 X000 触点由断开转为闭合，程序不会执行中断指针 I100 处的中断程序。

④ 定时中断。当需要每隔一定时间就反复执行某段程序时，可采用定时中断。三菱 FX_{1S}\FX_{1N}\FX_{1NC} PLC 无定时中断功能，三菱 FX_{2N}\FX_{2NC}\FX_{3G}\FX_{3U}\FX_{3UC} PLC 可使用 3 个定时中断指针。定时中断指针含义如下。



定时中断指针 I6□□、I7□□、I8□□ 可分别用 M8056、M8057、M8058 禁止。

4 主程序结束指令（FEND）

主程序结束指令格式如表 6-6 所示。

表 6-6 主程序结束指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
主程序结束指令	FEND FNC06		无

主程序结束指令使用要点如下。

- ① FEND 表示一个主程序结束，执行该指令后，程序返回到第 0 步。
- ② 多次使用 FEND 指令时，子程序或中断程序要写在最后的 FEND 指令与 END 指令之间，且必须以 RET 指令（针对子程序）或 IRET 指令（针对中断程序）结束。

5 刷新监视定时器指令（WDT）

(1) 指令格式

刷新监视定时器指令格式如表 6-7 所示。

表 6-7 刷新监视定时器指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
刷新监视定时器指令	WDT FNC07		无

(2) 使用说明

PLC 在运行时，若一个运行周期（从 0 步运行到 END 或 FENT）超过 200ms 时，内部运行监视定时器会让 PLC 的 CPU 出错指示灯变亮，同时 PLC 停止工作。为了解决这个问题，可使用 WDT 指令对监视定时器进行刷新。WDT 指令的使用如图 6-11 (a) 所示，若一个程序运行需 240ms，可在 120ms 程序处插入一个 WDT 指令，将监视定时器进行刷新，使定时器重新计时。

为了使 PLC 扫描周期超过 200ms，还可以使用 MOV 指令将希望运行的时间写入特殊数据寄存器 D8000 中，如图 6-11 (b) 所示，该程序将 PLC 扫描周期设为 300ms。

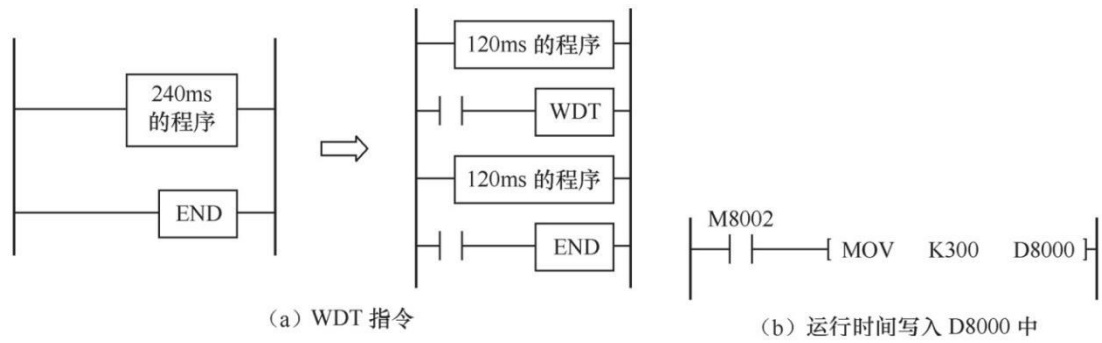


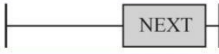
图 6-11 WDT 指令的使用

6 循环开始与结束指令

(1) 指令格式

循环开始与结束指令格式如表 6-8 所示。

表 6-8 循环开始与结束指令格式

指令名称	助记符与功能号	指令形式	操作数
			S
循环开始指令	FOR FNC08		K、H、KnX KnY、KnS KnM、 T、C、D、V、Z
循环结束指令	NEXT FNC09		无

(2) 使用说明

循环开始与结束指令的使用如图 6-12 所示，[FOR K4] 指令设定 A 段程序 (FOR ~ NEXT 之间的程序) 循环执行 4 次，[FOR D0] 指令设定 B 段程序循环执行 D0 (数据寄存器 D0 中的数值) 次，若 D0=2，则 A 段程序反复执行 4 次，而 B 段程序会执行 4×2=8 次，这是因为运行到 B 段程序时，B 段程序需要反复运行 2 次，然后往下执行，当执行到 A 段程序 NEXT 指令时，又返回到 A 段程序头部重新开始运行，直至 A 段程序从头到尾执行 4 次。

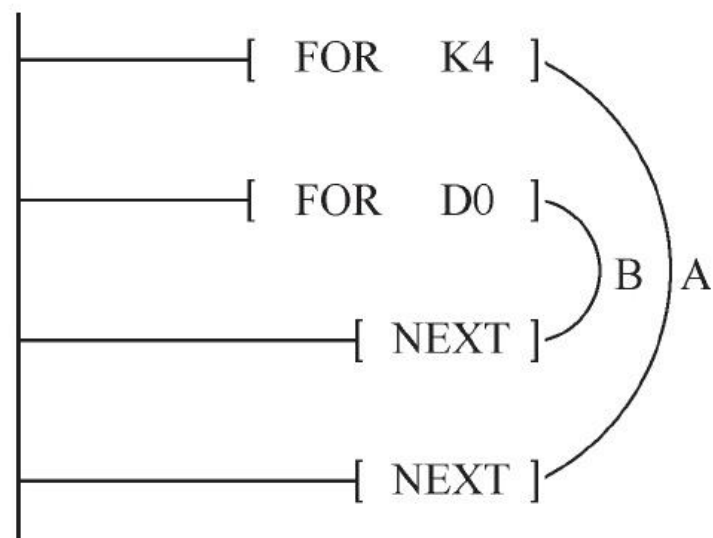


图 6-12 循环开始与结束指令的使用

FOR 与 NEXT 指令使用要点。

- ① FOR 与 NEXT 之间的程序可重复执行 n 次, n 由编程设定, $n=1 \sim 32767$ 。
- ② 循环程序执行完设定的次数后, 紧接着执行 NEXT 指令后面的程序步。
- ③ 在 FOR~NEXT 程序之间最多可嵌套 5 层其他的 FOR~NEXT 程序, 嵌套时应避免出现以下情况。
 - a. 缺少 NEXT 指令。
 - b. NEXT 指令写在 FOR 指令前。
 - c. NEXT 指令写在 FEND 或 END 之后。
 - d. NEXT 指令个数与 FOR 不一致。

1 比较指令

(1) 指令格式

比较指令格式如表 6-9 所示。

表 6-9 比较指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
比较指令	CMP FNC10		K、H KnX KnY、KnS KnM T、C、D、V、Z		
					Y、M、S

(2) 使用说明

比较指令的使用如图 6-13 所示。CMP 指令有两个源操作数 K100、C10 和一个目标操作数 M0（位元件），当常开触点 X000 闭合时，CMP 指令执行，将源操作数 K100 和计数器 C10 当前值进行比较，根据比较结果来驱动目标操作数指定的三个连号位元件，若 $K100 > C10$ ，M0 常开触点闭合；若 $K100 = C10$ ，M1 常开触点闭合，若 $K100 < C10$ ，M2 常开触点闭合。

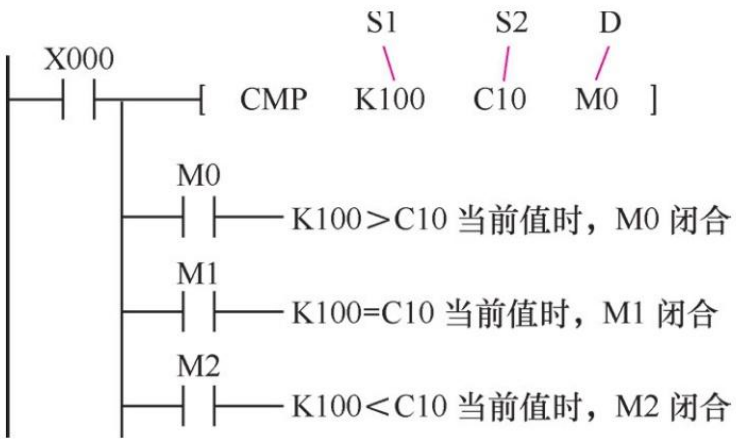


图 6-13 比较指令的使用

在指定 M0 为 CMP 的目标操作数时，M0、M1、M2 三个连号元件会被自动占用，在 CMP 指令执行后，这三个元件必定有一个处于 ON，当常开触点 X000 断开后，这三个元件的状态仍会被保存，要恢复它们的原状态，可采用复位指令。

2 区间比较指令

(1) 指令格式

区间比较指令格式如表 6-10 所示。

表 6-10 区间比较指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S1	S2	S3	D
区间比较指令	ZCP FNC11		K、H KnX KnY、KnS、KnM T、C、D、V、Z			Y、M、S

(2) 使用说明

区间比较指令的使用如图 6-14 所示。ZCP 指令有三个源操作数和一个目标操作数，前两个源操作数用于将数据分为三个区间，再将第三个源操作数在这三个区间进行比较，根据比较结果来驱动目标操作数指定的三个连号位元件，若 $C30 < K100$ ，M3 常开触点闭合，若 $K100 \leq C30 \leq K120$ ，M4 常开触点闭合，若 $C30 > K120$ ，M5 常开触点闭合。

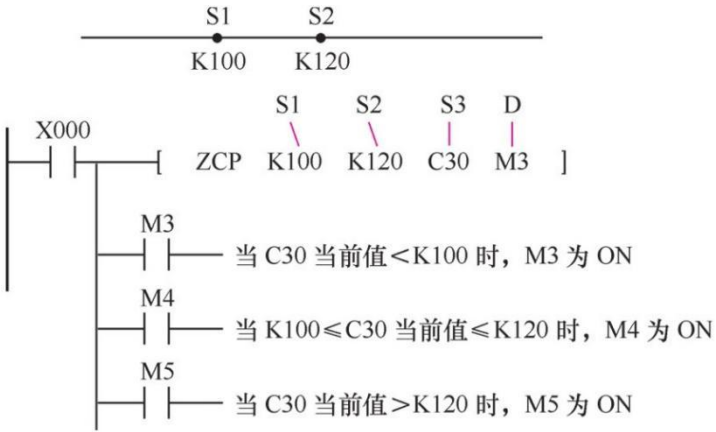


图 6-14 区间比较指令的使用

使用区间比较指令时，要求第一源操作数 S1 小于第二源操作数。

3 传送指令

(1) 指令格式

传送指令格式如表 6-11 所示。

表 6-11 传送指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
传送指令	MOV FNC12		K、H KnX、KnY、KnS、KnM T、C、D、V、Z	KnY、KnS、KnM T、C、D、V、Z

(2) 使用说明

传送指令的使用如图 6-15 所示。当常开触点 X000 闭合时，MOV 指令执行，将 K100（十进制数 100）送入数据寄存器 D10 中，由于 PLC 寄存器只能存储二进制数，因此将梯形图写入 PLC 前，编程软件会自动将十进制数转换成二进制数。



图 6-15 传送指令的使用

4 移位传送指令

(1) 指令格式

移位传送指令格式如表 6-12 所示。

表 6-12 移位传送指令格式

指令名称	助记符与功能号	指令形式	操作数				
			m1	m2	n	S	D
移位传送指令	SMOV FNC13		K、H			KnX 、 KnY 、 KnS 、 KnM T、C、D、V、Z	KnY 、 KnS 、 KnM T、C、D、V、Z

(2) 使用说明

移位传送指令的使用如图 6-16 所示。当常开触点 X000 闭合，SMOV 指令执行，首先将源数据寄存器 D1 中的 16 位二进制数据转换成四组 BCD 码，然后将这四组 BCD 码中的第 4 组 (m1=K4) 起的低 2 组 (m2=K2) 移入目标寄存器 D2 第 3 组 (n=K3) 起的低 2 组中，D2 中的第 4、1 组数据保持不变，再将形成的新四组 BCD 码还原成 16 位数据。例如，初始 D1 中的数据为 4567，D2 中的数据为 1234，执行 SMOV 指令后，D1 中的数据不变，仍为 4567，而 D2 中的数据将变成 1454。

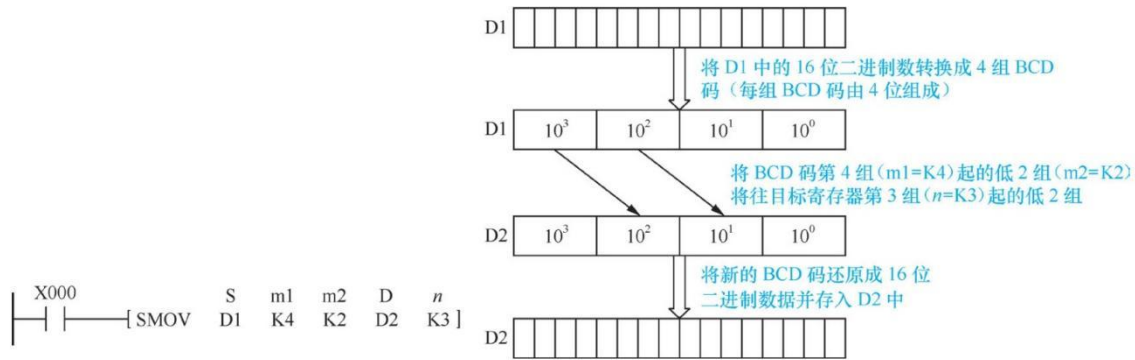


图 6-16 移位传送指令的使用

5 取反传送指令

(1) 指令格式

取反传送指令格式如表 6-13 所示。

表 6-13 取反传送指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
取反传送指令	CML FNC14		K、H KnX、KnY、KnS、KnM T、C、D、V、Z	KnY、KnS、KnM T、C、D、V、Z

(2) 使用说明

取反传送指令的使用如图 6-17 (a) 所示，当常开触点 X000 闭合时，CML 指令执行，将数据寄存器 D0 中的低 4 位数据取反，再将取反的数据按低位到高位顺序分别送入 4 个输出继电器 Y000~Y003 中，数据传送如图 6-17 (b) 所示。



图 6-17 取反传送指令的使用

6 成批传送指令

(1) 指令格式

成批传送指令格式如表 6-14 所示。

表 6-14 成批传送指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	<i>n</i>
成批传送指令	BMOV FNC15		<i>Kn</i> X、 <i>Kn</i> Y、 <i>Kn</i> S、 <i>Kn</i> M T、C、D	<i>Kn</i> Y、 <i>Kn</i> S、 <i>Kn</i> M T、C、D	K、H

(2) 使用说明

成批传送指令的使用如图 6-18 所示。当常开触点 X000 闭合时，BMOV 指令执行，将源操作元件 D5 开头的 *n* (*n*=3) 个连号元件中的数据批量传送到目标操作元件 D10 开头的 *n* 个连号元件中，即将 D5、D6、D7 3 个数据寄存器中的数据分别同时传送到 D10、D11、D12 中。

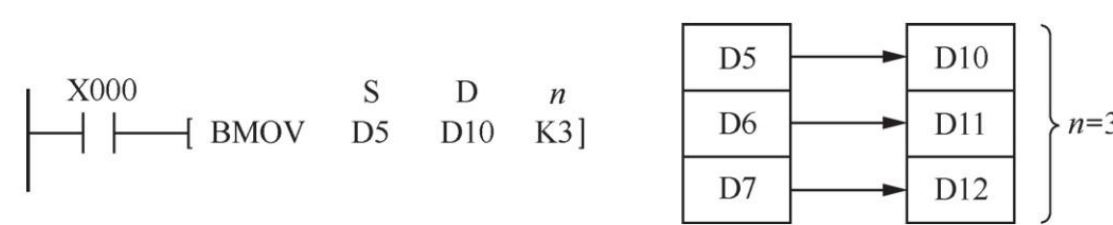


图 6-18 成批传送指令的使用

7 多点传送指令

(1) 指令格式

多点传送指令格式如表 6-15 所示。

表 6-15 多点传送指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	<i>n</i>
多点传送指令	FMOV FNC16		K、H KnX 、 KnY 、 KnS 、 KnM T、C、D、V、Z	KnY 、 KnS 、 KnM T、C、D	K、H

(2) 使用说明

多点传送指令的使用如图 6-19 所示。当常开触点 X000 闭合时，FMOV 指令执行，将源操作数 0 (K0) 同时送入以 D0 开头的 10 ($n=10$) 个连号数据寄存器中。

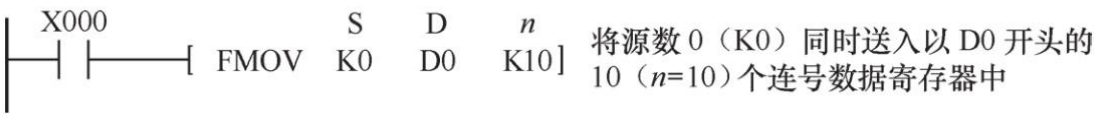


图 6-19 多点传送指令的使用

8 数据交换指令

(1) 指令格式

数据交换指令格式如表 6-16 所示。

表 6-16 数据交换指令格式

指令名称	助记符与功能号	指令形式	操作数	
			D1	D2
数据交换指令	XCH FNC17		KnY 、 KnS 、 KnM T、C、D、V、Z	KnY 、 KnS 、 KnM T、C、D、V、Z

(2) 使用说明

数据交换指令的使用如图 6-20 所示。当常开触点 X000 闭合时，XCHP 指令执行，两目标操作数 D10、D11 中的数据相互交换，若指令执行前 D10=100、D11=101，指令执行后，D10=101、D11=100，如果使用连续执行指令 XCH，则每个扫描周期数据都要交换，很难预知执行结果，所以一般采用脉冲执行指令 XCHP 进行数据交换。

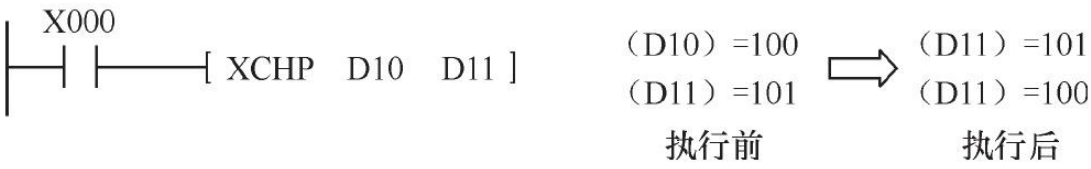


图 6-20 数据交换指令的使用

9 BCD 码转换指令

(1) 指令格式

BCD 码转换指令格式如表 6-17 所示。

表 6-17 BCD 码转换指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
BCD 码转换指令	BCD FNC18		KnX 、 KnY 、 KnS 、 KnM T、C、D、V、Z	KnY 、 KnS 、 KnM T、C、D、V、Z

(2) 使用说明

BCD 码转换指令的使用如图 6-21 所示。当常开触点 X000 闭合时, BCD 指令执行, 将源操作元件 D10 中的二进制数转换成 BCD 码, 再存入目标操作元件 D12 中。

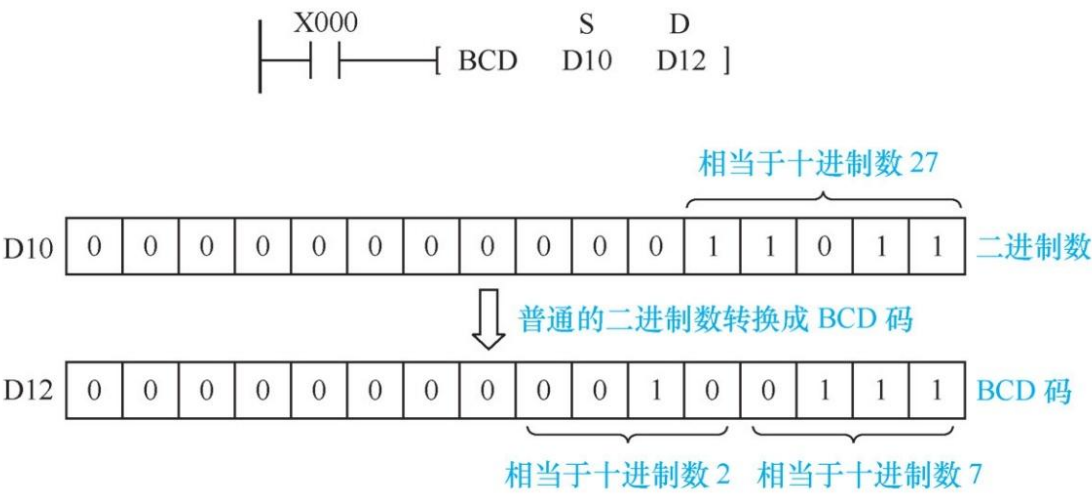


图 6-21 BCD 码转换指令的使用

三菱 FX 系列 PLC 内部在进行四则运算和增量、减量运算时, 都是以二进制方式进行的。

10 二进制码转换指令

(1) 指令格式

二进制码转换指令格式如表 6-18 所示。

表 6-18 二进制码转换指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
BIN 码转换指令	BIN FNC19		KnX 、 KnY 、 KnS 、 KnM T、C、D、V、Z	KnY 、 KnS 、 KnM T、C、D、V、Z

(2) 使用说明

二进制码转换指令的使用如图 6-22 所示。当常开触点 X000 闭合时，BIN 指令执行，将源操作元件 X000 ~ X007 构成的两组 BCD 码转换成二进制数码（BIN 码），再存入目标操作元件 D13 中。若 BIN 指令的源操作数不是 BCD 码，则会发生运算错误，如 X007 ~ X000 的数据为 10110100，该数据的前 4 位 1011 转换成十进制数为 11，它不是 BCD 码，因为单组 BCD 码不能大于 9，单组 BCD 码只能在 0000 ~ 1001 范围内。

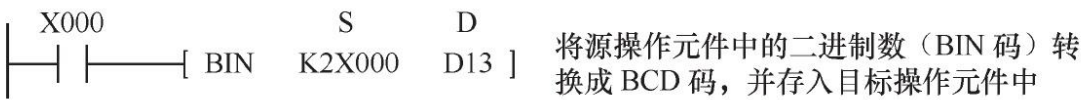


图 6-22 二进制码转换指令的使用

1 二进制加法运算指令

(1) 指令格式

二进制加法运算指令格式如表 6-19 所示。

表 6-19 二进制加法运算指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
二进制加法运算指令	ADD FNC20		K、H KnX、KnY、KnS、KnM T、C、D、V、Z		KnY、KnS、KnM T、C、D、V、Z

(2) 使用说明

二进制加指令的使用如图 6-23 所示。

① 在图 6-23 (a) 中, 当常开触点 X000 闭合时, ADD 指令执行, 将两个源操元件 D10 和 D12 中的数据进行相加, 结果存入目标操作元件 D14 中。源操作数可正可负, 它们是以代数形式进行相加, 如 $5 + (-7) = -2$ 。

② 在图 6-23 (b) 中, 当常开触点 X000 闭合时, DADD 指令执行, 将源操元件 D11、D10 和 D13、D12 分别组成 32 位数据再进行相加, 结果存入目标操作元件 D15、D14 中。当进行 32 位数据运算时, 要求每个操作数是两个连号的数据寄存器, 为了确保不重复, 指定的元件最好为偶数编号。

③ 在图 6-23 (c) 中, 当常开触点 X001 闭合时, ADDP 指令执行, 将 D0 中的数加 1, 结果仍存入 D0 中。当一个源操作数和一个目标操作数为同一元件时, 最好采用脉冲执行型加指令 ADDP, 因为若是连续型加指令, 每个扫描周期指令都要执行一次, 所得结果很难确定。

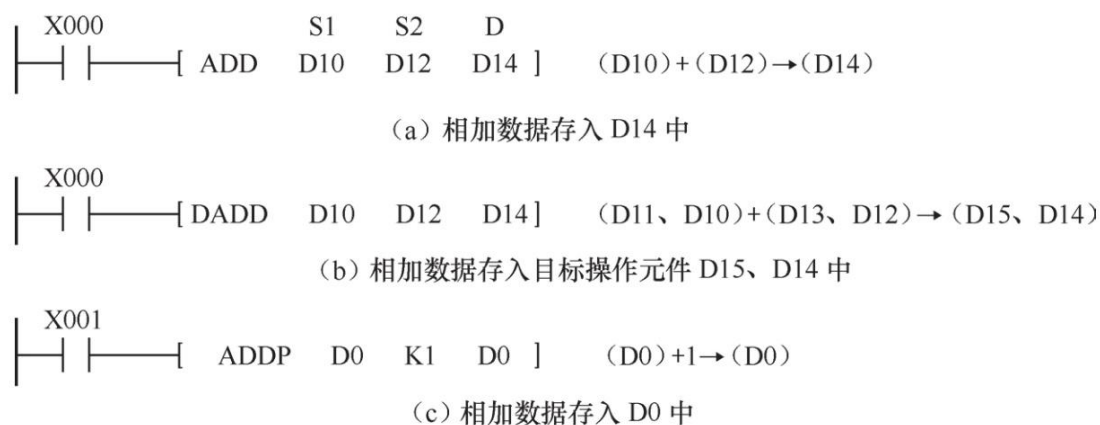


图 6-23 二进制加指令的使用

④ 在进行加法运算时，若运算结果为 0，0 标志继电器 M8020 会动作，若运算结果超出-32768~+32767（16 位数相加）或-2147483648~+2147483647（32 位数相加）范围，借位标志继电器 M8022 会动作。

2 二进制减法运算指令

(1) 指令格式

二进制减法运算指令格式如表 6-20 所示。

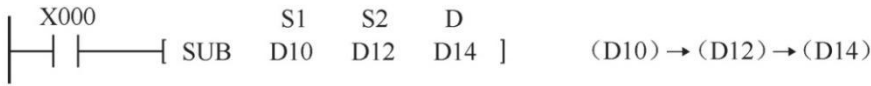
表 6-20 二进制减法运算指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
二进制减法运算指令	SUB FNC21		K、H KnX、KnY、KnS、KnM T、C、D、V、Z		KnY、KnS、KnM T、C、D、V、Z

(2) 使用说明

二进制减指令的使用如图 6-24 所示。

① 在图 6-24 (a) 中, 当常开触点 X000 闭合时, SUB 指令执行, 将 D10 和 D12 中的数据进行相减, 结果存入目标操作元件 D14 中。源操作数可正可负, 它们是以代数形式进行相减, 如 $5 - (-7) = 12$ 。



(a) 相减数据存储在 D14 中



(b) 相减数据存储在 D15、D14 中



(c) 相减数据存储在 D0 中

图 6-24 二进制减指令的使用

② 在图 6-24 (b) 中, 当常开触点 X000 闭合时, DSUB 指令执行, 将源操元件 D11、D10 和 D13、D12 分别组成 32 位数据再进行相减, 结果存入目标操作元件 D15、D14 中。当进行 32 位数据运算时, 要求每个操作数是两个连号的数据寄存器, 为了确保不重复, 指定的元件最好为偶数编号。

③ 在图 6-24 (c) 中, 当常开触点 X001 闭合时, SUBP 指令执行, 将 D0 中的数据减 1, 结果仍存入 D0 中。当一个源操作数和一个目标操作数为同一元件时, 最好采用脉冲执行型减指令 SUBP, 若是连续型减指令, 每个扫描周期指令都要执行一次, 所得结果很难确定。

④ 在进行减法运算时, 若运算结果为 0, 0 标志继电器 M8020 会动作, 若运算结果超出 $-32768 \sim +32767$ (16 位数相减) 或 $-2147483648 \sim +2147483647$ (32 位数相减) 范围, 借位标志继电器 M8022 会动作。

3 二进制乘法运算指令

(1) 指令格式

二进制乘法运算指令格式如表 6-21 所示。

表 6-21 二进制乘法运算指令

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
二进制乘法运算指令	MUL FNC22		K、H KnX、KnY、KnS、KnM T、C、D、V、Z		KnY、KnS、KnM T、C、D、V、Z (V、Z 不能用于 32 位)

(2) 使用说明

二进制乘法指令的使用如图 6-25 所示。在进行 16 位数乘积运算时，结果为 32 位，如图 6-25 (a) 所示；在进行 32 位数乘积运算时，乘积结果为 64 位，如图 6-25 (b) 所示；运算结果的最高位为符号位 (0：正；1：负)。



图 6-25 二进制乘法指令的使用

4 二进制除法运算指令

(1) 指令格式

二进制除法运算指令格式如表 6-22 所示。

表 6-22 二进制除法运算指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
二进制除法运算指令	DIV FNC23		K、H KnX、KnY、KnS、KnM T、C、D、V、Z		KnY、KnS、KnM T、C、D、V、Z (V、Z不能用于32位)

(2) 使用说明

二进制除法指令的使用如图 6-26 所示。在进行 16 位数除法运算时，商为 16 位，余数也为 16 位，如图 6-26 (a) 所示；在进行 32 位数除法运算时，商为 32 位，余数也为 32 位，如图 6-26 (b) 所示；商和余的最高位为用 1、0 表示正、负。



图 6-26 二进制除法指令的使用

在使用二进制除法指令时要注意。

- ① 当除数为 0 时，运算会发生错误，不能执行指令。
- ② 若将位元件作为目标操作数，无法得到余数。
- ③ 当被除数或除数中有一方为负数时，商则为负，当被除数为负时，余数则为负。

5 二进制加 1 运算指令

(1) 指令格式

二进制加 1 运算指令格式如表 6-23 所示。

表 6-23 二进制加 1 运算指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
二进制加 1 运算指令	INC FNC24		KnY 、 KnS 、 KnM T、C、D、V、Z

(2) 使用说明

二进制加 1 指令的使用如图 6-27 所示。当常开触点 X000 闭合时，INCP 指令执行，数据寄存器 D12 中的数据自动加 1。若采用连续执行型指令 INC，则每个扫描周期数据都要增加 1，在 X000 闭合时可能会经过多个扫描周期，因此最终结果很难确定，常采用脉冲执行型指令进行加 1 运算。



图 6-27 二进制加 1 指令的使用

6 二进制减 1 运算指令

(1) 指令格式

二进制减 1 运算指令格式如表 6-24 所示。

表 6-24 二进制减 1 运算指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
二进制减 1 运算指令	DEC FNC25		KnY 、 KnS 、 KnM T、C、D、V、Z

(2) 使用说明

二进制减 1 指令的使用如图 6-28 所示。当常开触点 X000 闭合时, DECP 指令执行, 数据寄存器 D12 中的数据自动减 1。为保证 X000 每闭合一次数据减 1 一次, 常采用脉冲执行型指令进行减 1 运算。



图 6-28 二进制减 1 指令的使用

7 逻辑与指令

(1) 指令格式

逻辑与指令格式如表 6-25 所示。

表 6-25 逻辑与指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
逻辑与指令	WAND FNC26		K、H KnX、KnY、KnS、 KnM T、C、D、V、Z		KnY、KnS、 KnM T、C、D、V、Z

(2) 使用说明

逻辑与指令的使用如图 6-29 所示。当常开触点 X000 闭合时，WAND 指令执行，将 D10 与 D12 中的数据「逐位进行与运算」，结果保存在 D14 中。

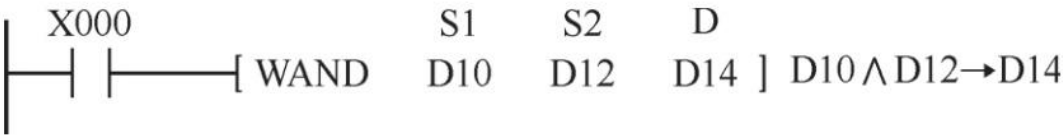


图 6-29 逻辑与指令的使用

与运算规律是「有 0 得 0，全 1 得 1」，具体为：0∧0=0，0∧1=0，1∧0=0，1∧1=1。

8 逻辑或指令

(1) 指令格式

逻辑或指令格式如表 6-26 所示。

表 6-26 逻辑或指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
逻辑或指令	WOR FNC27		K、H KnX、KnY、KnS、KnM T、C、D、V、Z		KnY、KnS、KnM T、C、D、V、Z

(2) 使用说明

逻辑或指令的使用如图 6-30 所示。当常开触点 X000 闭合时，WOR 指令执行，将 D10 与 D12 中的数据「逐位进行或运算」，结果保存在 D14 中。

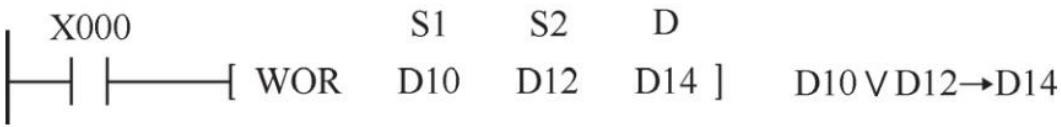


图 6-30 逻辑或指令的使用

或运算规律是「有 1 得 1，全 0 得 0」，具体为：0∨0=0，0∨1=1，1∨0=1，1∨1=1。

9 异或指令

(1) 指令格式

逻辑异或指令格式如表 6-27 所示。

表 6-27 逻辑异或指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
逻辑异或指令	WXOR FNC28		K、H KnX、KnY、KnS、KnM T、C、D、V、Z		KnY、KnS、KnM T、C、D、V、Z

(2) 使用说明

异或指令的使用如图 6-31 所示。当常开触点 X000 闭合时，WXOR 指令执行，将 D10 与 D12 中的数据「逐位进行异或运算」，结果保存在 D14 中。

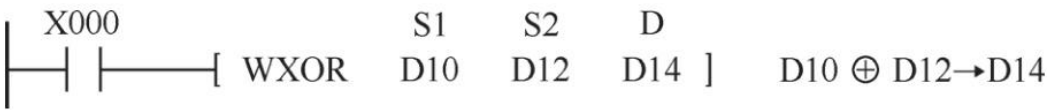


图 6-31 异或指令的使用

异或运算规律是「相同得 0，相异得 1」，具体为：0⊕0=0，0⊕1=1，1⊕0=1，1⊕1=0。

10 求补指令

(1) 指令格式

逻辑求补指令格式如表 6-28 所示。

表 6-28 逻辑求补指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
逻辑求补指令	NEG FNC29		KnY、KnS、KnM T、C、D、V、Z

(2) 使用说明

求补指令的使用如图 6-32 所示。当常开触点 X000 闭合时，NEGP 指令执行，将 D10 中的数据「逐位取反再加 1」。求补的功能是对数据进行变号（绝对值不变），如求补前 D10=+8，求补后 D10=-8。为了避免每个扫描周期都进行求补运算，通常采用脉冲执行型求补指令 NEGP。

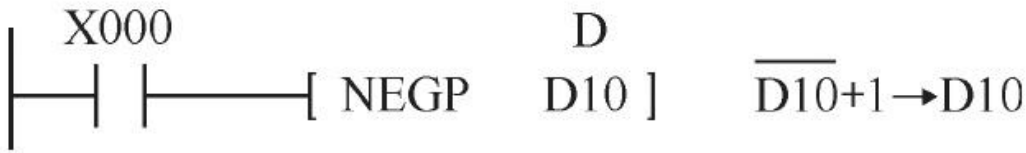


图 6-32 求补指令的使用

1 循环右移指令

(1) 指令格式

循环右移指令格式如表 6-29 所示。

表 6-29 循环右移指令格式

指令名称	助记符与功能号	指令形式	操作数	
			D	n (移位量)
循环右移指令	ROR FNC30		K、H KnY、KnS、KnM T、C、D、V、Z	K、H $n \leq 16$ (16 位) $n \leq 32$ (32 位)

(2) 使用说明

循环右移指令的使用如图 6-33 所示。当常开触点 X000 闭合时，RORP 指令执行，将 D0 中的数据右移（从高位往低位移）4 位，其中低 4 位移至高 4 位，最后移出的一位（即图中标有*号的位）除了移到 D0 的最高位外，还会移入进位标记继电器 M8022 中。

为了避免每个扫描周期都进行右移，通常采用脉冲执行型指令 RORP。

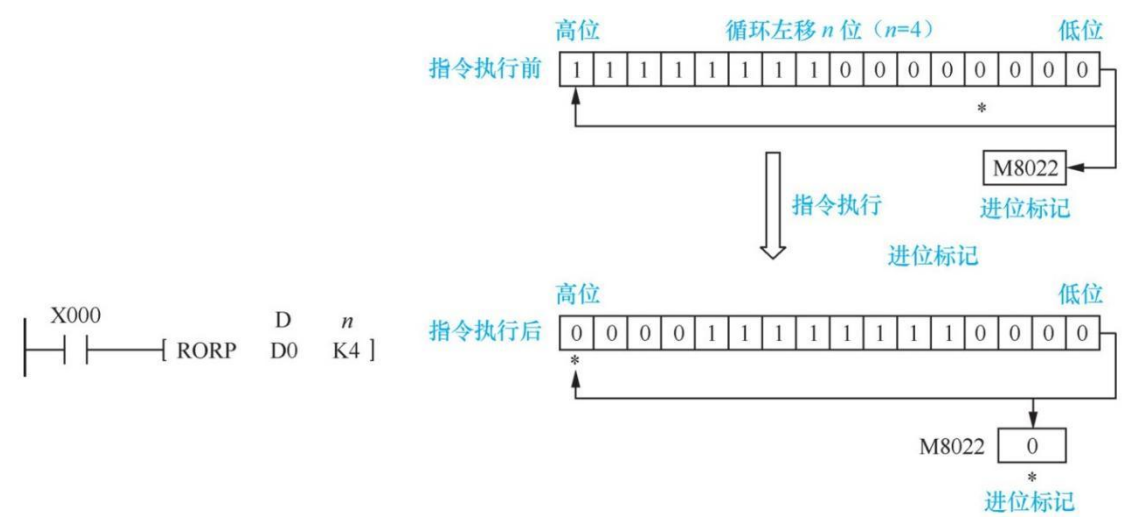


图 6-33 循环右移指令的使用

2 循环左移指令

(1) 指令格式

循环左移指令格式如表 6-30 所示。

表 6-30 循环左移指令格式

指令名称	助记符与功能号	指令形式	操作数	
			D	n (移位量)
循环左移指令	ROL FNC31		K、H KnY、KnS、KnM T、C、D、V、Z	K、H $n \leq 16$ (16 位) $n \leq 32$ (32 位)

(2) 使用说明

循环左移指令的使用如图 6-34 所示。当常开触点 X000 闭合时，ROLP 指令执行，将 D0 中的数据左移（从低位往高位移）4 位，其中高 4 位移至低 4 位，最后移出的一位（即图中标有*号的位）除了移到 D0 的最低位外，还会移入进位标记继电器 M8022 中。

为了避免每个扫描周期都进行左移，通常采用脉冲执行型指令 ROLP。

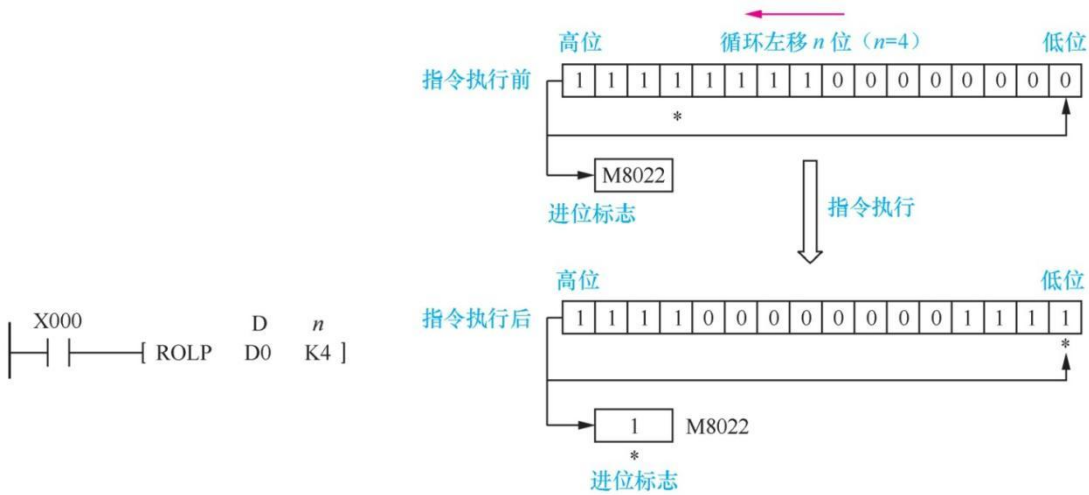


图 6-34 循环左移指令的使用

3 带进位循环右移指令

(1) 指令格式

带进位循环右移指令格式如表 6-31 所示。

表 6-31 带进位循环右移指令格式

指令名称	助记符与功能号	指令形式	操作数	
			D	n (移位量)
带进位循环右移指令	RCR FNC32		K、H KnY、KnS、KnM T、C、D、V、Z	K、H $n \leq 16$ (16 位) $n \leq 32$ (32 位)

(2) 使用说明

带进位循环右移指令的使用如图 6-35 所示。当常开触点 X000 闭合时，RCRP 指令执行，将 D0 中的数据右移 4 位，D0 中的低 4 位与继电器 M8022 的进位标记位（图中为 1）一起往高 4 位移，D0 最后移出的一位（即图中标有*号的位）移入 M8022。为了避免每个扫描周期都进行右移，通常采用脉冲执行型指令 RCRP。

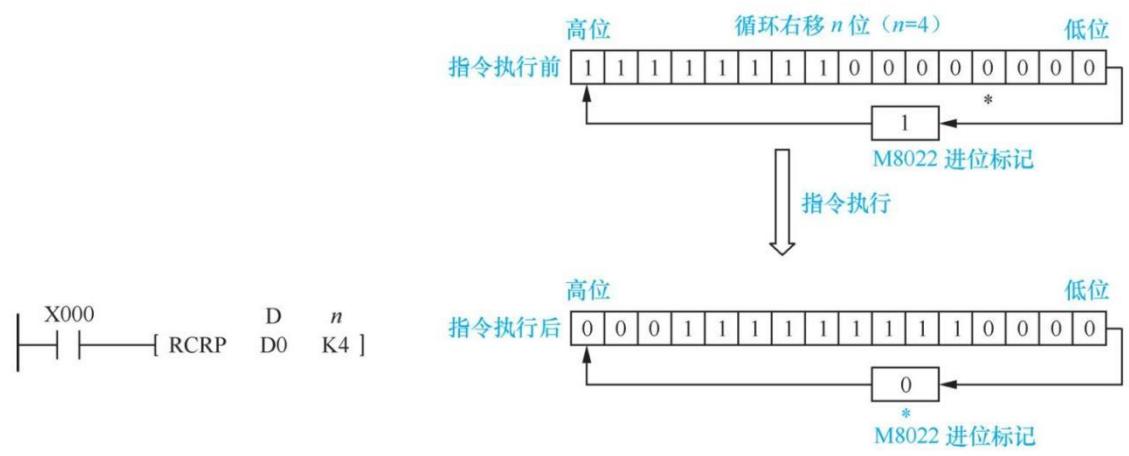


图 6-35 带进位循环右移指令的使用

带进位循环左移指令

(1) 指令格式

带进位循环左移指令格式如表 6-32 所示。

表 6-32 带进位循环左移指令格式

指令名称	助记符与功能号	指令形式	操作数	
			D	n (移位数)
带进位循环左移指令	RCL FNC33		K、H KnY、KnS、KnM T、C、D、V、Z	K、H $n \leq 16$ (16 位) $n \leq 32$ (32 位)

(2) 使用说明

带进位循环左移指令的使用如图 6-36 所示。当常开触点 X000 闭合时，RCLP 指令执行，将 D0 中的数据左移 4 位，D0 中的高 4 位与继电器 M8022 的进位标记位（图中为 0）一起往低 4 位移，D0 最后移出的一位（即图中标有*号的位）移入 M8022。为了避免每个扫描周期都进行左移，通常采用脉冲执行型指令 RCLP。

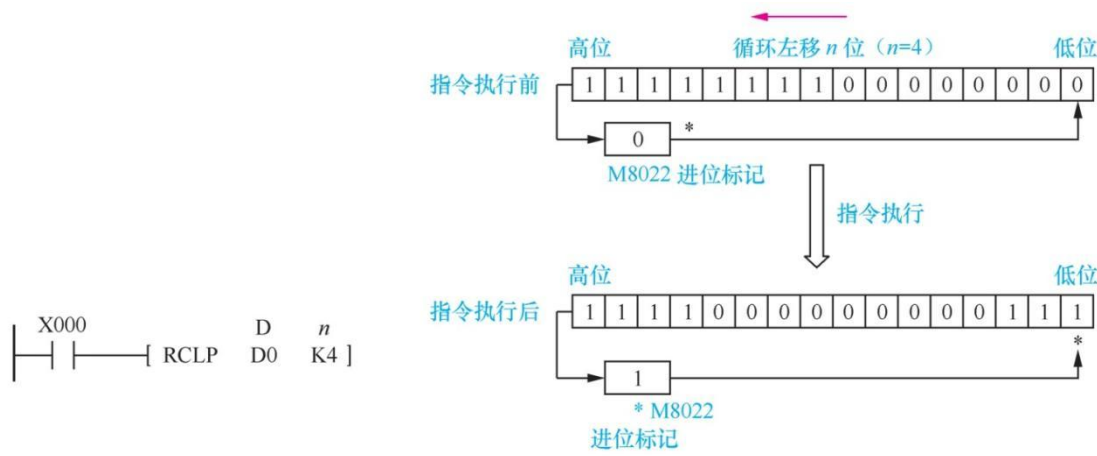


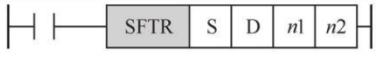
图 6-36 带进位循环左移指令的使用

5 位右移指令

(1) 指令格式

位右移指令格式如表 6-33 所示。

表 6-33 位右移指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S	D	n1 (目标位元件的个数)	n2 (移位量)
位右移指令	SFTR FNC34		X、Y M、S	Y、M、S	K、H $n2 \leq n1 \leq 1024$	

(2) 使用说明

位右移指令的使用如图 6-37 所示。在图 6-37 (a) 中，当常开触点 X010 闭合时，SFTRP 指令执行，将 X003~X000 四个元件的位状态（1 或 0）右移入 M15~M0 中，X000 为源起始位元件，M0 为目标起始位元件，K16 为目标位元件数量，K4 为移位量。SFTRP 指令执行后，M3~M0 移出丢失，M15~M4 移到原 M11~M0，X003~X000 则移入原 M15~M12，如图 6-37 (b) 所示。

为了避免每个扫描周期都移动，通常采用脉冲执行型指令 SFTRP。

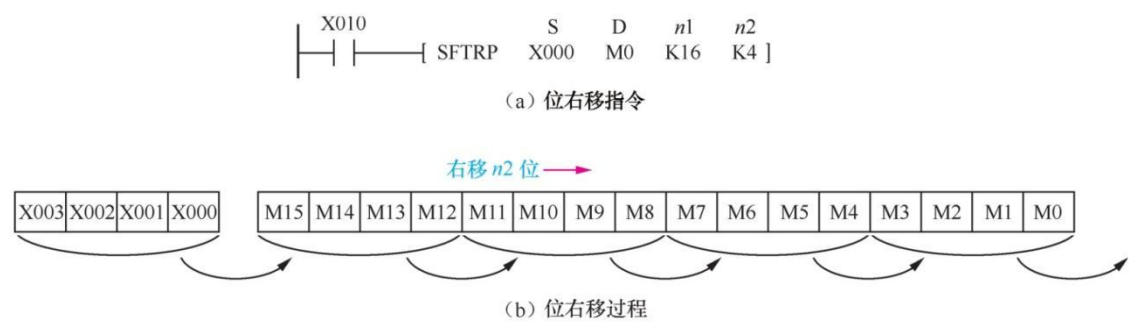


图 6-37 位右移指令的使用

6 位左移指令

(1) 指令格式

位左移指令格式如表 6-34 所示。

表 6-34 位左移指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S	D	n1 (目标位元件的个数)	n2 (移位量)
位左移指令	SFTL FNC35		X、Y M、S	Y、M、S	K、H $n2 \leq n1 \leq 1024$	

(2) 使用说明

位左移指令的使用如图 6-38 所示。在图 6-38 (a) 中，当常开触点 X010 闭合时，SFTLP 指令执行，将 X003~X000 四个元件的位状态（1 或 0）左移入 M15~M0 中，X000 为源起始位元件，M0 为目标起始位元件，K16 为目标位元件数量，K4 为移位量。SFTLP 指令执行后，M15~M12 移出丢失，M11~M0 移到原 M15~M4，X003~X000 则移入原 M3~M0，如图 6-38 (b) 所示。

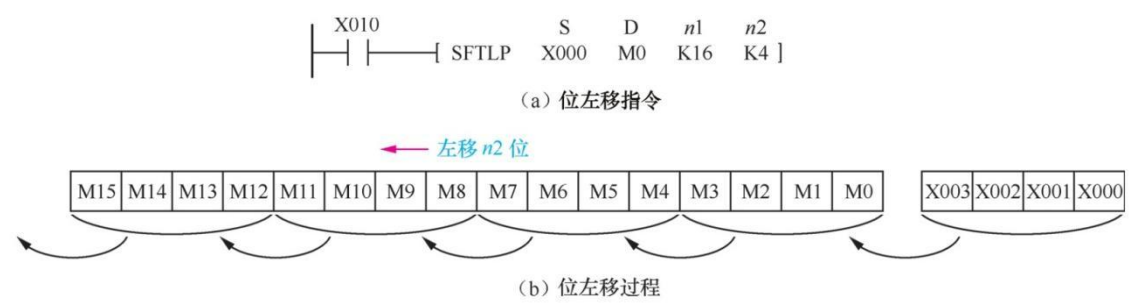


图 6-38 位左移指令的使用

为了避免每个扫描周期都移动，通常采用脉冲执行型指令 SFTLP。

7 字右移指令

(1) 指令格式

字右移指令格式如表 6-35 所示。

表 6-35 字右移指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S	D	n1 (目标位元件的个数)	n2 (移位量)
字右移指令	WSFR FNC36		KnX 、 KnY 、 KnS 、 KnM T、C、D、	KnY 、 KnS 、 KnM T、C、D、	K、H $n2 \leq n1 \leq 1024$	

(2) 使用说明

字右移指令的使用如图 6-39 所示。在图 6-39 (a) 中，当常开触点 X000 闭合时，WSFRP 指令执行，将 D3~D0 4 个字元件的数据右移入 D25~D10 中，D0 为源起始字元件，D10 为目标起始字元件，K16 为目标字元件数量，K4 为移位量。WSFRP 指令执行后，D13~D10 的数据移出丢失，D25~D14 的数据移入原 D21~D10，D3~D0 则移入原 D25~D22，如图 6-39 (b) 所示。

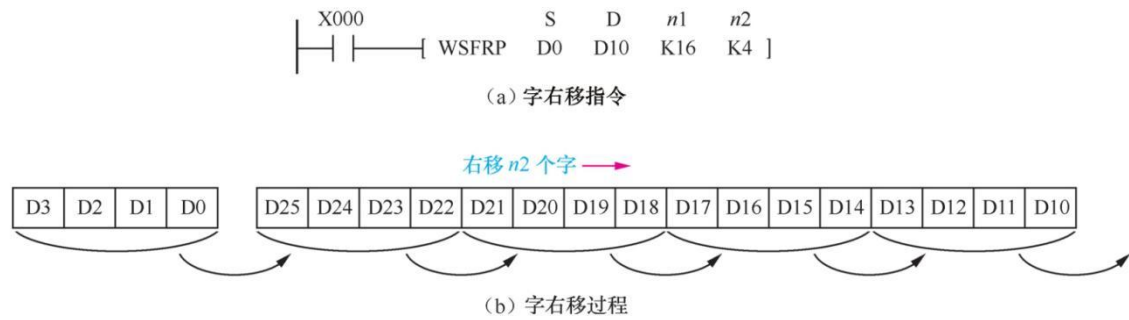


图 6-39 字右移指令的使用

为了避免每个扫描周期都移动，通常采用脉冲执行型指令 WSFRP。

8 字左移指令

(1) 指令格式

字左移指令格式如表 6-36 所示。

表 6-36 字左移指令格式

指令名称	助记符与 功能号	指令形式	操作数			
			S	D	n1 (目标位 元件的个数)	n2 (移位量)
字左移指令	WSFL FNC37		KnX 、 KnY 、 KnS 、 KnM T、C、D、	KnY 、 KnS 、 KnM T、C、D、	K、H $n2 \leq n1 \leq 1024$	

(2) 使用说明

字左移指令的使用如图 6-40 所示。在图 6-40 (a) 中，当常开触点 X000 闭合时，WSFLP 指令执行，将 D3 ~ D0 4 个字元件的数据左移入 D25 ~ D10 中，D0 为源起始字元件，D10 为目标起始字元件，K16 为目标字元件数量，K4 为移位量。WSFLP 指令执行后，D25 ~ D22 的数据移出丢失，D21 ~ D10 的数据移入原 D25 ~ D14，D3 ~ D0 则移入原 D13 ~ D10，如图 6-40 (b) 所示。

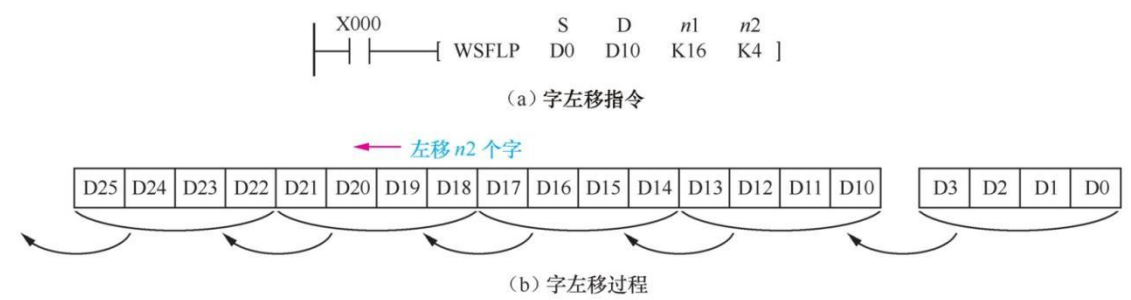


图 6-40 字左移指令的使用

为了避免每个扫描周期都移动，通常采用脉冲执行型指令 WSFLP。

9 移位写入指令

(1) 指令格式

移位写入指令格式如表 6-37 所示。

表 6-37 移位写入指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	n
移位写入指令	SFWR FNC38		K、H KnX、KnY、KnS、KM T、C、D、V、Z	KnY、KnS、 KnM T、C、D、	K、H $2 \leq n \leq 512$

(2) 使用说明

移位写入指令的使用如图 6-41 所示。当常开触点 X000 闭合时，SFWRP 指令执行，将 D0 中的数据写入 D2 中，同时作为指示器（或称指针）的 D1 的数据自动为 1，当 X000 触点第二次闭合时，D0 中的数据被写入 D3 中，D1 中的数据自动变为 2，连续闭合 X000 触点时，D0 中的数据将依次写入 D4、D5...中，D1 中的数据也会自动递增 1，当 D1 超过 $n-1$ 时，所有寄存器被存满，进位标志继电器 M8022 会被置 1。

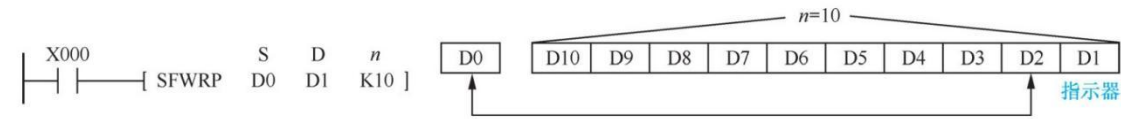


图 6-41 移位写入指令的使用

D0 为源操作元件，D1 为目标起始元件，K10 为目标存储元件数量。为了避免每个扫描周期都移动，通常采用脉冲执行型指令 SFWRP。

10 移位读出指令

(1) 指令格式

移位读出指令格式如表 6-38 所示。

表 6-38 移位读出指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	<i>n</i> (源操作元件数量)
移位读出指令	SFRD FNC39		K、H KnY、KnS、KnM T、C、D	KnY、KnS、KnM T、C、D、V、Z	K、H $2 \leq n \leq 512$

(2) 使用说明

移位读出指令的使用如图 6-42 所示。当常开触点 X000 闭合时，SFRDP 指令执行，将 D2 中的数据读入 D20 中，指示器 D1 的数据自动减 1，同时 D3 数据移入 D2 (即 D10 ~ D3→D9 ~ D2)。当连续闭合 X000 触点时，D2 中的数据会不断读入 D20，同时 D10 ~ D3 中的数据也会由左往右不断逐字移入 D2 中，D1 中的数据会随之递减 1，同时当 D1 减到 0 时，所有寄存器的数据都被读出，0 标志继电器 M8020 会被置 1。

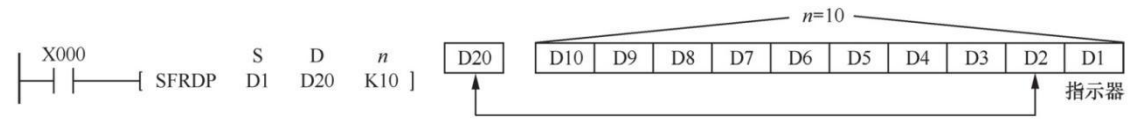


图 6-42 移位读出指令的使用

D1 为源起始操作元件，D20 为目标元件，K10 为源操作元件数量。为了避免每个扫描周期都移动，通常采用脉冲执行型指令 SFRDP。

1 成批复位指令

(1) 指令格式

成批复位指令格式如表 6-39 所示。

表 6-39 成批复位指令格式

指令名称	助记符与功能号	指令形式	操作数	
			D1	D2
成批复位指令	ZRST FNC40		Y、M、T、C、S、D (D1 ≤ D2, 且为同一系列元件)	

(2) 使用说明

成批复位指令的使用如图 6-43 所示。在 PLC 开始运行的瞬间, M8002 触点接通一个扫描周期, ZRST 指令执行, 将辅助继电器 M500~M599、计数器 C235~C255 和状态继电器 S0~S127 全部复位清 0。

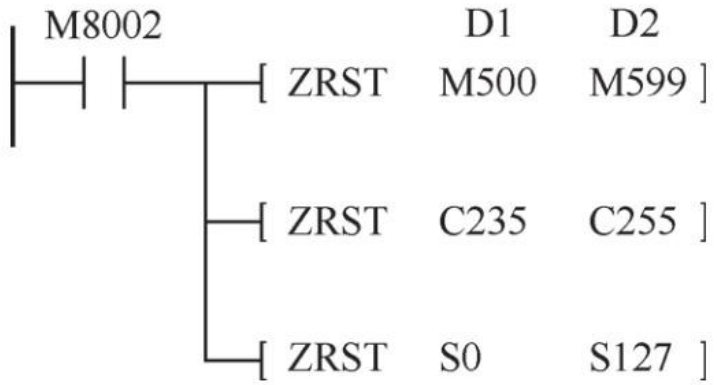


图 6-43 成批复位指令的使用

在使用 ZRST 指令时要注意, 目标操作数 D2 序号应大于 D1, 并且为同一系列元件。

2 解码指令

(1) 指令格式

解码指令格式如表 6-40 所示。

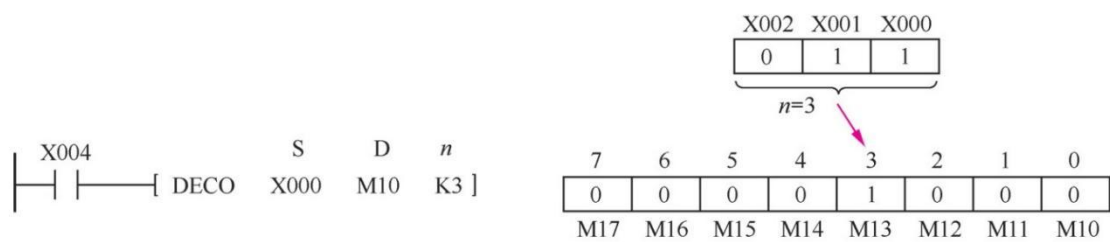
表 6-40 解码指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	n
解码指令	DECO FNC41		K、H X、Y、M、S、 T、C、D、V、Z	Y、M、S、 T、C、D	K、H n=1 ~ 8

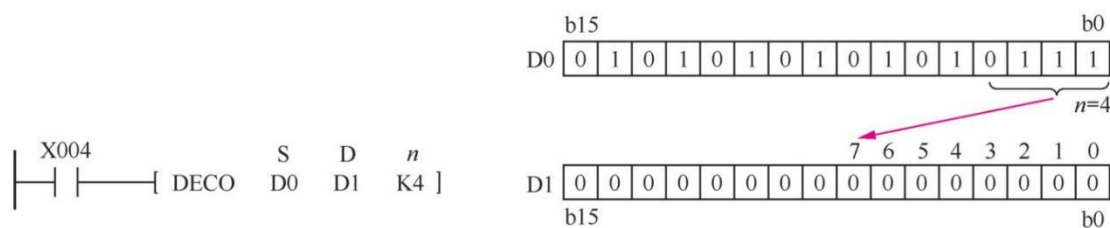
(2) 使用说明

解码指令的使用如图 6-44 所示，该指令的操作数为位元件，在图 6-44 (a) 中，当常开触点 X004 闭合时，DECO 指令执行，将 X000 为起始编号的 3 个连号位元件（由 n=3 指定）组合状态进行解码，3 位数解码有 8 种结果，解码结果存入在 M17 ~ M10（以 M10 为起始目标位元件）的 M13 中，因 X002、X001、X000 分别为 0、1、1，而 (011)₂=3，即指令执行结果使 M17 ~ M10 的第 4 位 M13=1。

图 6-44 (b) 的操作数为字元件，当常开触点 X004 闭合时，DECO 指令执行，对 D0 的低 4 位数进行解码，4 位数解码有 16 种结果，而 D0 的低 4 位数为 0111，(0111)₂=7，解码结果使目标字元件 D1 的低 8 位为 1，D1 的其他位均为 0。



(a) 解码指令使用说明



(b) 操作数为字元件的解码过程

图 6-44 解码指令的使用

当 n 在 K1 ~ K8 范围内变化时, 解码则有 2 ~ 255 种结果, 结果保存的目标元件不要在其他控制中重复使用。

3 编码指令

(1) 指令格式

编码指令格式如表 6-41 所示。

表 6-41 编码指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	<i>n</i>
编码指令	ENCO FNC42		X、Y、M、S、 T、C、D、V、Z	T、C、D、 V、Z	K、H <i>n</i> =1 ~ 8

(2) 使用说明

编码指令的使用如图 6-45 所示。图 6-45 (a)的源操作数为位元件,当常开触点 X004 闭合时, ENCO 指令执行, 对 M17 ~ M10 中的 1 进行编码 (第 5 位 M15=1) , 编码采用 3 位 (由 *n*=3 确定) , 编码结果 101 (即 5) 存入 D10 低 3 位中。M10 为源操作起始位元件, D10 为目标操作元件, *n*为编码位数。

图 6-45 (b) 的源操作数为字元件, 当常开触点 X004 闭合时, ENCO 指令执行, 对 D0 低 8 位中的 1 (b6=1) 进行编码, 编码采用 3 位 (由 *n*=3 确定) , 编码结果 110 (即 6) 存入 D1 低 3 位中。

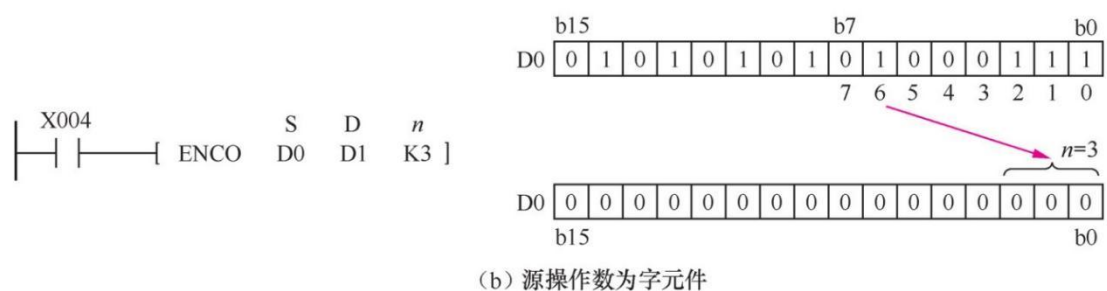
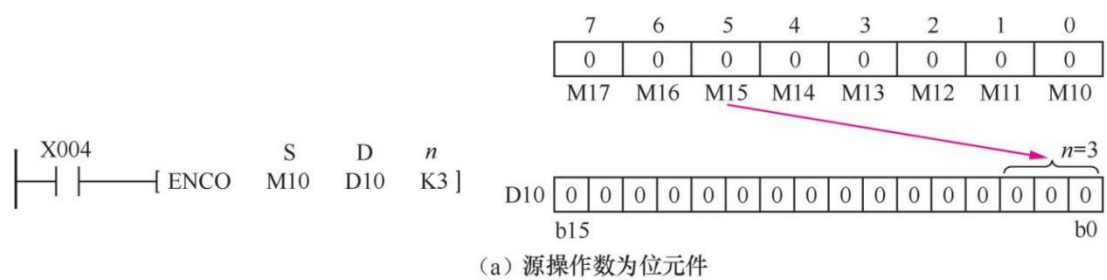


图 6-45 编码指令的使用

当源操作元件中有多个 1 时，只对高位 1 进行编码，低位 1 忽略。

4 1 总数和指令

(1) 指令格式

1 总数和指令格式如表 6-42 所示。

表 6-42 1 总数和指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
1 总数和指令	SUM FNC43		K、H KnX、KnY、KnM、KnS、 T、C、D、V、Z	KnY、KnM、KnS、 T、C、D、V、Z

(2) 使用说明

1 总数和指令的使用如图 6-46 所示。当常开触点 X000 闭合，SUM 指令执行，计算源操作元件 D0 中 1 的总数，并将总数值存入目标操作元件 D2 中，图中 D0 中总共有 9 个 1，那么存入 D2 的数值为 9（即 1001）。

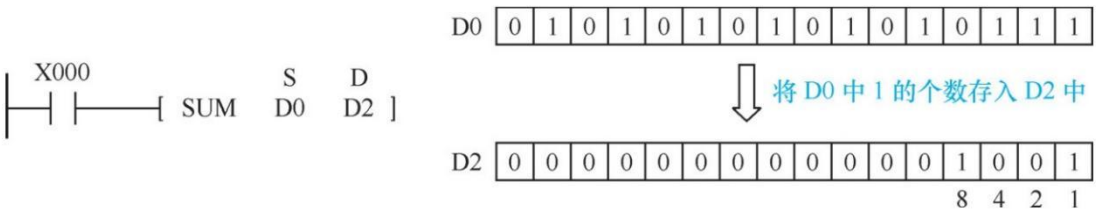


图 6-46 1 总数和指令的使用

若 D0 中无 1，0 标志继电器 M8020 会动作，M8020=1。

5 1 位判别指令

(1) 指令格式

1 位判别指令格式如表 6-43 所示。

表 6-43 1 位判别指令格式

指令名称	助记符与 功能号	指令形式	操作数		
			S	D	n
1 位判别指令	BON FNC44		K、H KnX、KnY、 KnM、KnS、 T、C、D、V、Z	Y、S、M	K、H n=0 ~ 15 (16 位操作) n=0 ~ 32 (32 位操作)

(2) 使用说明

1 位判别指令的使用如图 6-47 所示。当常开触点 X000 闭合，BON 指令执行，判别源操作元件 D10 的第 15 位 (n=15) 是否为 1，若为 1，则让目标操作位元件 M0=1，若为 0；M0=0。



图 6-47 1 位判别指令的使用

6 平均值指令

(1) 指令格式

平均值指令格式如表 6-44 所示。

表 6-44 平均值指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	<i>n</i>
平均值指令	MEAN FNC45		<i>Kn</i> X、 <i>Kn</i> Y、 <i>Kn</i> M、 <i>Kn</i> S、T、C、D	<i>Kn</i> Y、 <i>Kn</i> M、 <i>Kn</i> S、 T、C、D	K、H <i>n</i> =1 ~ 64

(2) 使用说明

平均值指令的使用如图 6-48 所示。当常开触点 X000 闭合时，MEAN 指令执行，计算 D0 ~ D2 中数据的平均值，平均值存入目标元件 D10 中。D0 为源起始元件，D10 为目标元件，*n*=3 为源元件的个数。

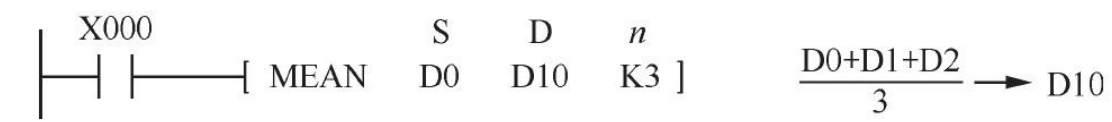


图 6-48 平均值指令的使用

7 报警置位指令

(1) 指令格式

报警置位指令格式如表 6-45 所示。

表 6-45 报警置位指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	n
报警置位指令	ANS FNC46		T (T0 ~ T199)	S (S900 ~ S999)	K n=1 ~ 32767 (100ms 单位)

(2) 使用说明

报警置位指令的使用如图 6-49 所示。当常开触点 X000、X001 同时闭合时，定时器 T0 开始 1s 计时（n=10），若两触点同时闭合时间超过 1s，ANS 指令会将报警状态继电器 S900 置位，若两触点同时闭合时间不到 1s，定时器 T0 未计完 1s 即复位，ANS 指令不会对 S900 置位。



图 6-49 报警置位指令的使用

8 报警复位指令

(1) 指令格式

报警复位指令格式如表 6-46 所示。

表 6-46 报警复位指令格式

指令名称	助记符与功能号	指令形式	操作数
报警复位指令	ANR FNC47		无

(2) 使用说明

报警复位指令的使用如图 6-50 所示。当常开触点 X003 闭合时，ANR 指令执行，将信号报警继电器 S900 ~ S999 中正在动作（即处于置位状态）的报警继电器复位，若这些报警器有多个处于置位状态，在 X003 闭合时小编号的报警器复位，当 X003 再一次闭合时，则对下一个编号的报警器复位。

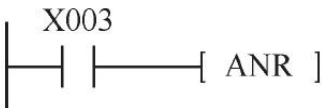


图 6-50 报警复位指令的使用

如果采用连续执行型 ANR 指令，在 X003 闭合期间，每经过一个扫描周期，ANR 指令就会依次对编号由小到大的报警器进行复位。

9 求平方根指令

(1) 指令格式

求平方根指令格式如表 6-47 所示。

表 6-47 求平方根指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
求平方根指令	SQR FNC48		K、H、D	D

(2) 使用说明

求平方根指令的使用如图 6-51 所示。当常开触点 X000 闭合时，SQR 指令执行，对源操作元件 D10 中的数进行求平方根运算，运算结果的整数部分存入目标操作元件 D12 中，若存在小数部分，将小数部分舍去，同时进位标志继电器 M8021 置位，若运算结果为 0，零标志继电器 M8020 置位。



图 6-51 求平方根指令的使用

10 二进制整数转换为浮点数指令

(1) 指令格式

二进制整数转换成浮点数指令格式如表 6-48 所示。

表 6-48 二进制整数转换成浮点数指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
二进制整数转换为浮点数指令	FLT FNC49		K、H、D	D

(2) 使用说明

二进制整数转换为浮点数指令的使用如图 6-52 所示。当常开触点 X000 闭合时, FLT 指令执行, 将源操作元件 D10 中的二进制整数转换成浮点数, 再将浮点数存入目标操作元件 D13、D12 中。

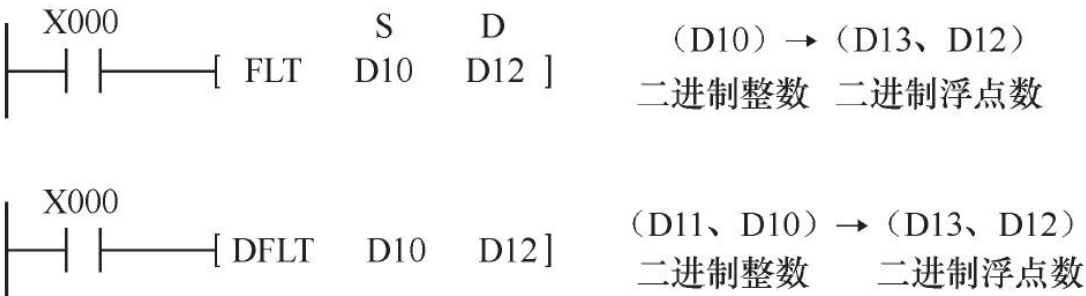


图 6-52 二进制整数转换为浮点数指令的使用

由于 PLC 编程很少用到浮点数运算, 读者若对浮点数及运算感兴趣, 可查阅有关资料, 这里不作介绍。

1 输入/输出刷新指令

(1) 指令格式

输入/输出刷新指令格式如表 6-49 所示。

表 6-49 输入/输出刷新指令格式

指令名称	助记符与功能号	指令形式	操作数	
			D	n
输入/输出刷新指令	REF FNC50		X、Y	K、H

(2) 使用说明

在 PLC 运行程序时，输入端子接收输入信号，PLC 通常不会马上处理输入信号，要等到下一个扫描周期才处理输入信号，这样从输入到处理有一段时间差。另外，PLC 在运行程序产生输出信号时，也不是马上从输出端子输出，而是等程序运行到 END 时，才将输出信号从输出端子输出，这样从产生输出信号到信号从输出端子输出也有一段时间差。如果希望 PLC 在运行时能即刻接收输入信号，或能即刻输出信号，可采用输入/输出刷新指令。

输入/输出刷新指令的使用如图 6-53 所示。图 6-53 (a) 为输入刷新，当常开触点 X000 闭合时，REF 指令执行，将以 X010 为起始元件的 8 个 ($n=8$) 输入继电器 X010 ~ X017 刷新，即让 X010 ~ X017 端子输入的信号能马上被这些端子对应的输入继电器接收。图 6-53 (b) 为输出刷新，当常开触点 X001 闭合时，REF 指令执行，将以 Y000 为起始元件的 24 个 ($n=24$) 输出继电器 Y000 ~ Y007、Y010 ~ Y017、Y020 ~ Y027 刷新，让这些输出继电器能即刻往相应的输出端子输出信号。

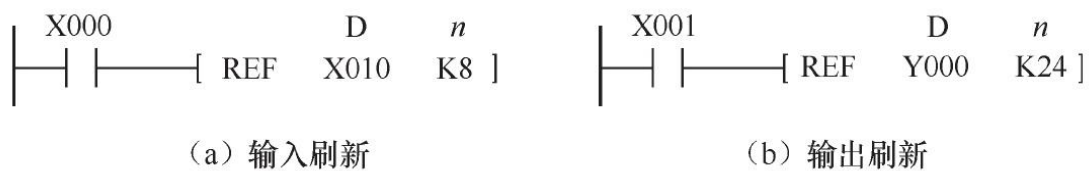


图 6-53 输入/输出刷新指令的使用

REF 指令指定的首元件编号应为 X000、X010、X020..., Y000、Y010、Y020..., 刷新的点数 n 就应是 8 的整数, 如 8、16、24 等。

2 输入滤波常数调整指令

(1) 指令格式

输入滤波常数调整指令格式如表 6-50 所示。

表 6-50 输入滤波常数调整指令格式

指令名称	助记符与功能号	指令形式	操作数
			n
输入滤波常数调整指令	REFF FNC51		K、H

(2) 使用说明

为了提高 PLC 输入端子的抗干扰性，在输入端子内部都设有滤波器，滤波时间常数在 10ms 左右，可以有效吸收短暂的输入干扰信号，但对于正常的高速短暂输入信号也有抑制作用，为此 PLC 将一些输入端子的电子滤波器时间常数设为可调。三菱 FX_{2N} 系列 PLC 将 X000 ~ X017 端子内的电子滤波器时间常数设为可调，调节采用 REFF 指令，时间常数调节范围为 0 ~ 60ms。

输入滤波常数调整指令的使用如图 6-54 所示。当常开触点 X010 闭合时，REFF 指令执行，将 X000 ~ X017 端子的滤波常数设为 1ms ($n=1$)，该指令执行前这些端子的滤波常数为 10ms，该指令执行后这些端子时间常数为 1ms，当常开触点 X020 闭合时，REFF 指令执行，将 X000 ~ X017 端子的滤波常数设为 20ms ($n=20$)，此后至 END 或 FEND 处，这些端子的滤波常数为 20ms。

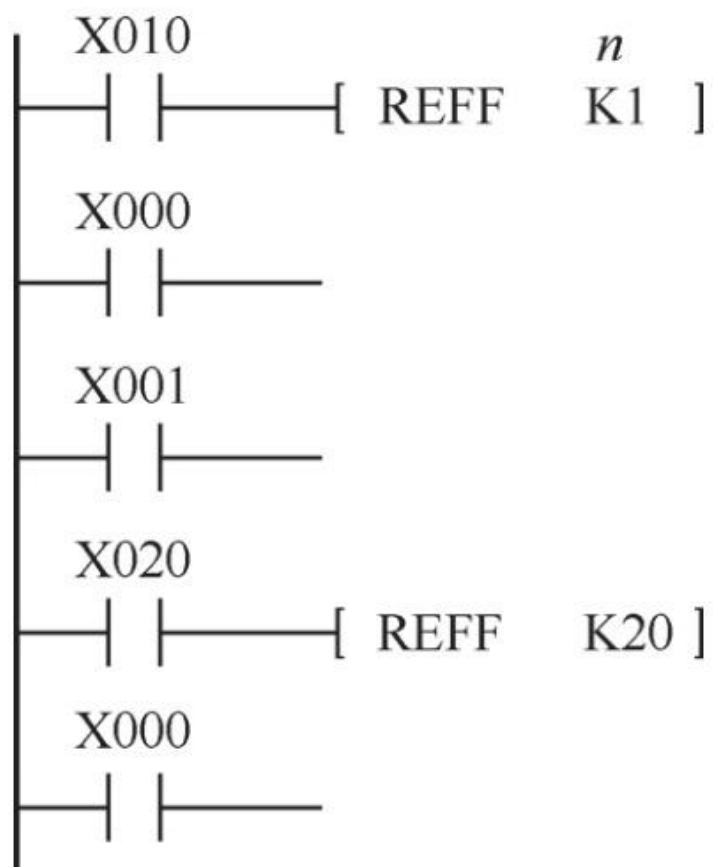


图 6-54 输入滤波常数调整指令的使用

当 X000 ~ X007 端子用作高速计数输入、速度检测或中断输入时，其输入滤波常数自动设为 50 μ s。

3 矩阵输入指令

(1) 指令格式

矩阵输入指令格式如表 6-51 所示。

表 6-51 矩阵输入指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S	D1	D2	n
矩阵输入指令	MTR FNC52		X	Y	Y、M、S	K、H n=2 ~ 8

(2) 矩阵输入电路

PLC 通过输入端子来接收外界输入信号，由于输入端子数量有限，若采用一个端子接收一路信号的普通输入方式，很难实现大量多路信号输入，给 PLC 加设矩阵输入电路可以有效解决这个问题。

图 6-55 (a) 所示是一种 PLC 矩阵输入电路，采用 X020 ~ X027 端子接收外界输入信号，外接 3 组由二极管和按键组成的矩阵输入电路，端子另一端则分别接 PLC 的 Y020、Y021、Y022 端子。在工作时，Y020、Y021、Y022 端子内部硬触点轮流接通，如图 6-55 (b) 所示，当 Y020 接通 (ON) 时，Y021、Y022 断开；当 Y021 接通时，Y020、Y022 断开；当 Y022 接通时，Y020、Y021 断开，然后重复这个过程，一个周期内每个端子接通时间为 20ms。

在 Y020 端子接通期间，若按下第一组输入电路中的某个按键，如 M37 按键，X027 端子输出的电流经二极管、按键流入 Y020 端子，并经 Y020 端子内部闭合的硬触点流到 COM 端，X027 端子有电流输出，相当于该端子有输入信号，该输入信号在 PLC 内部被转存到辅助继电器 M37 中。在 Y020 端子接通期间，若按下第二组或第三组中某个按键，

由于此时 Y021、Y022 端子均断开，故操作这两组按键均无效。在 Y021 端子接通期间，X020~X027 端子接收第二组按键输入，在 Y022 端子接通期间，X020~X027 端子接收第三组按键输入。

在采用图 6-55 (a) 形式的矩阵输入电路时，如果将输出端子 Y020~Y027 和输入端子 X020~X027 全部利用起来，则可以实现 $8 \times 8 = 64$ 个开关信号输入，由于 Y020~Y027 每个端子接通时间为 20ms，故矩阵电路的扫描周期为 $8 \times 20\text{ms} = 160\text{ms}$ 。对于扫描周期长的矩阵输入电路，若输入信号时间小于扫描周期，可能会出现输入无效的情况。例如，在图 6-55 (a) 中，若在 Y020 端子刚开始接通时按下按键 M52，按下时间为 30ms，再松开，由于此时 Y022 端子还未开始导通（从 Y020 到 Y022 导通时间间隔为 40ms），故操作按键 M52 无效，因此矩阵输入电路不适用于要求快速输入的场合。

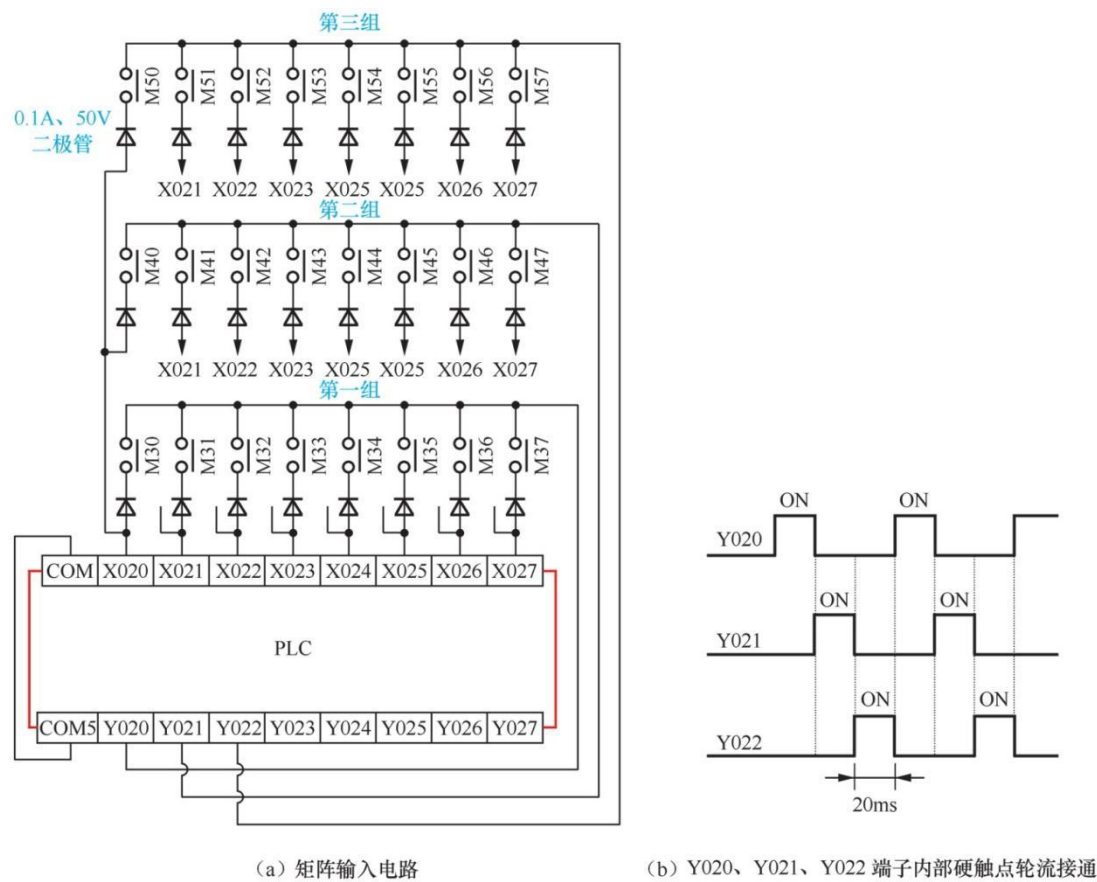


图 6-55 一种 PLC 矩阵输入电路

(3) 矩阵输入指令的使用

若 PLC 采用矩阵输入方式,除了要加设矩阵输入电路外,还须用 MTR 指令进行矩阵输入设置。矩阵输入指令的使用如图 6-56 所示。当触点 M0 闭合时, MTR 指令执行,将[S]X020 为起始编号的 8 个连号元件作为矩阵输入,将[D1]Y020 为起始编号的 3 个 ($n=3$) 连号元件作为矩阵输出,将矩阵输入信号保存在以 M30 为起始编号的 3 组 8 个连号元件 (M30~M37、M40~M47、M50~M57) 中。



图 6-56 矩阵输入指令的使用

4 高速计数器置位指令

(1) 指令格式

高速计数器置位指令格式如表 6-52 所示。

表 6-52 高速计数器置位指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
高速计数器置位指令	HSCS FNC53		K、H、 KnX、KnY、KnM、 KnS、T、C、D、V、 Z	C (C235 ~ C255)	Y、M、S

(2) 使用说明

高速计数器置位指令的使用如图 6-57 所示。当常开触点 X010 闭合时，若高速计数器 C255 的当前值变为 100 (99→100 或 101→100)，HSCS 指令执行，将 Y010 置位。



图 6-57 高速计数器置位指令的使用

5 高速计数器复位指令

(1) 指令格式

高速计数器复位指令格式如表 6-53 所示。

表 6-53 高速计数器复位指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
高速计数器复位指令	HSCR FNC54		K、H、 KnX、KnY、KnM、 KnS、T、C、D、V、Z	C (C235 ~ C255)	Y、M、S

(2) 使用说明

高速计数器复位指令的使用如图 6-58 所示。当常开触点 X010 闭合时，若高速计数器 C255 的当前值变为 100 (99→100 或 101→100)，HSCR 指令执行，将 Y010 复位。

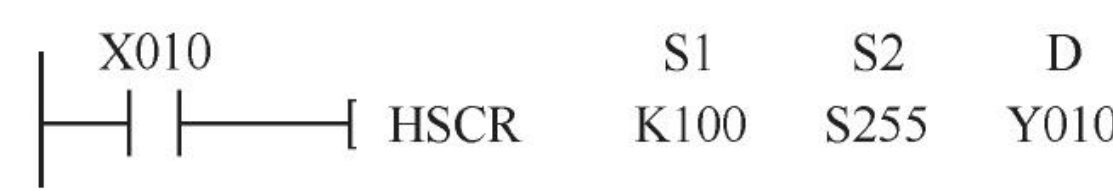


图 6-58 高速计数器复位指令的使用

6 高速计数器区间比较指令

(1) 指令格式

高速计数器区间比较指令格式如表 6-54 所示。

表 6-54 高速计数器区间比较指令格式

指令名称	助记符与 功能号	指令形式	操作数			
			S1	S2	S3	D
高速计数器区间 比较指令	HSZ FNC55		K、H、 KnX、KnY、KnM、 KnS、T、C、D、V、Z		C (C235 ~ C255)	Y、M、S (3个连号元件)

(2) 使用说明

高速计数器区间比较指令的使用如图 6-59 所示。在 PLC 运行期间，M8000 触点始终闭合，高速计数器 C251 开始计数，同时 HSZ 指令执行，当 C251 当前计数值 <K1000 时，让输出继电器 Y000 为 ON，当 $K1000 \leq C251$ 当前计数值 $\leq K2000$ 时，让输出继电器 Y001 为 ON，当 C251 当前计数值 >K2000 时，让输出继电器 Y002 为 ON。

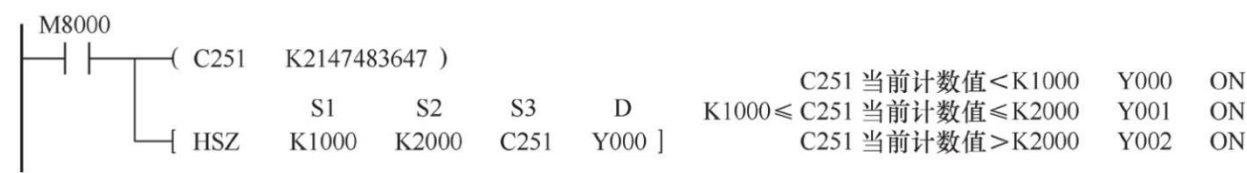


图 6-59 高速计数器区间比较指令的使用

7 速度检测指令

(1) 指令格式

速度检测指令格式如表 6-55 所示。

表 6-55 速度检测指令格式

指令名称	助记符与 功能号	指令形式	操作数		
			S1	S2	D
速度检测指令	SPD FNC56		X0 ~ X5	K、H、 KnX、KnY、KnM、KnS、 T、C、D、V、Z	T、C、D、V、Z

(2) 使用说明

速度检测指令的使用如图 6-60 所示。当常开触点 X010 闭合时，SPD 指令执行，计算 X000 输入端子在 100ms 内输入脉冲的个数，并将个数值存入 D0 中，指令还使用 D1、D2，其中 D1 用来存放当前时刻的脉冲数值（会随时变化），到 100ms 时复位；D2 用来存放计数的剩余时间，到 100ms 时复位。

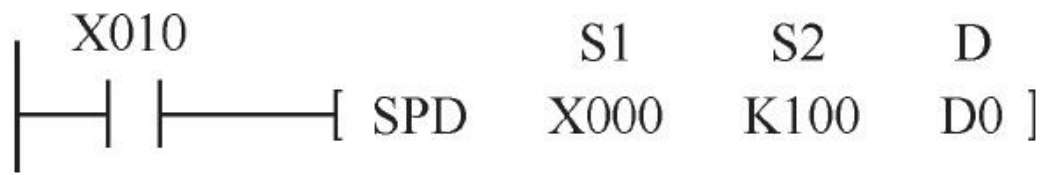


图 6-60 速度检测指令的使用

采用旋转编码器配合 SPD 指令可以检测电动机的转速。旋转编码器结构如图 6-61 所示，旋转编码器盘片与电动机转轴连动，在盘片旁安装有接近开关，盘片凸起部分靠近接近开关时，开关会产生脉冲输出， n 为编码器旋转一周输出的脉冲数。在测速时，先将测速

$$N = \left(\frac{60 \times [D]}{n \times [S2]} \times 10^3 \right) \text{ r/min}$$

电动机的转速计算公式


$$N = \left(\frac{60 \times [D]}{n \times [S2]} \times 10^3 \right) \text{ r/min} = \left(\frac{60 \times [D]}{360 \times 100} \times 10^3 \right) \text{ r/min} = \left(\frac{5 \times [D]}{3} \right) \text{ r/min}.$$

程序段	指令	说明
0000	SPD X000 K100 D0	将 X000 端子在 100ms 内输入的脉冲个数存入 D0 中
0001	MUL D0 K5 D3	将 D0 的脉冲个数与 5 相乘，结果存入 D3 中
0002	DDIV D3 K3 D10	将 D3 中的数除以 3，结果存入 D10，D10 中的数值即为电动机转速值
0003	END	

图 6-62 电动机转速检测程序

8 脉冲输出指令

(1) 指令格式

脉冲输出指令格式如表 6-56 所示。

表 6-56 脉冲输出指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
脉冲输出指令	PLSY FNC57		K、H、 KnX、KnY、KnM、KnS、 T、C、D、V、Z		Y0 或 Y1

(2) 使用说明

脉冲输出指令的使用如图 6-63 所示。当常开触点 X010 闭合时，PLSY 指令执行，让 Y000 端子输出占空比为 50% 的 1000Hz 脉冲信号，产生脉冲个数由 D0 指定。



图 6-63 脉冲输出指令的使用

脉冲输出指令使用要点如下。

- ① [S1]为输出脉冲的频率，对于 FX_{2N} 系列 PLC，频率范围为 10~20kHz；[S2]为要求输出脉冲的个数，对于 16 位操作元件，可指定的个数为 1~32767；对于 32 位操作元件，可指定的个数为 1~2147483647，如指定个数为 0，则持续输出脉冲。[D]为脉冲输出端子，要求输出端子为晶体管输出型，只能选择 Y000 或 Y001。
- ② 脉冲输出结束后，完成标记继电器 M8029 置 1，输出脉冲总数保存在 D8037 (高位) 和 D8036 (低位) 。

③ 若选择产生连续脉冲，在 X010 断开后 Y000 停止脉冲输出，X010 再闭合时重新开始。

④ [S1]中的内容在该指令执行过程中可以改变，[S2]中的内容在指令执行时不能改变。

9 脉冲调制指令

(1) 指令格式

脉冲调制指令格式如表 6-57 所示。

表 6-57 脉冲调制指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
脉冲调制指令	PWM FNC58		K、H、 KnX、KnY、KnM、KnS、 T、C、D、V、Z		Y0 或 Y1

(2) 使用说明

脉冲调制指令的使用如图 6-64 所示。当常开触点 X010 闭合时，PWM 指令执行，让 Y000 端子输出脉冲宽度为[S1]D10、周期为[S2]50 的脉冲信号。

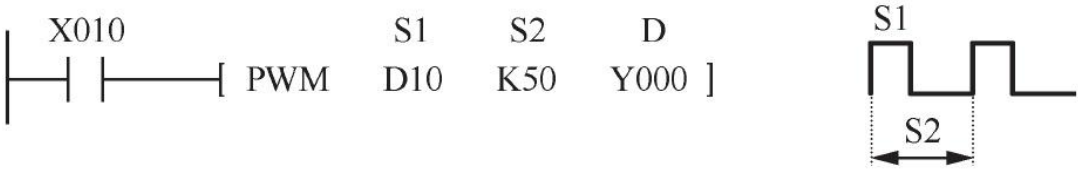


图 6-64 脉冲调制指令的使用

脉冲调制指令使用要点如下。

- ① [S1]为输出脉冲的宽度 t ， $t=0 \sim 32767\text{ms}$ ；[S2]为输出脉冲的周期 T ， $T=1 \sim 32767\text{ms}$ ，要求[S2]>[S1]，否则会出错；[D]为脉冲输出端子，只能选择 Y000 或 Y001。
- ② 当 X010 断开后，Y000 端子停止脉冲输出。

10 可调速脉冲输出指令

(1) 指令格式

可调速脉冲输出指令格式如表 6-58 所示。

表 6-58 可调速脉冲输出指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S1	S2	S3	D
可调速脉冲输出指令	PLSR FNC59		K、H、 KnX、KnY、KnM、KnS、 T、C、D、V、Z			Y0 或 Y1

(2) 使用说明

可调速脉冲输出指令的使用如图 6-65 所示。当常开触点 X010 闭合时，PLSR 指令执行，让 Y000 端子输出脉冲信号，要求输出脉冲频率由 0 开始，在 3600ms 内上升到最高频率 500Hz，在最高频率时产生 D0 个脉冲，再在 3600ms 内从最高频率下降到 0。



图 6-65 可调速脉冲输出指令的使用

可调速脉冲输出指令使用要点如下。

- ① [S1]为输出脉冲的最高频率，最高频率要设成 10 的倍数，设置范围为 10~20kHz。
- ② [S2]为最高频率时输出脉冲数，该数值不能小于 110，否则不能正常输出，[S2]的范围是 110~32767（16 位操作数）或 110~2147483647（32 位操作数）。

③ [S3]为加减速时间，它是指脉冲由 0 升到最高频率（或最高频率降到 0）所需的时间。输出脉冲的一次变化为最高频率的 1/10。加减速时间设置有一定的范围，具体可采用以下式子计算：

$$\frac{90000}{[S1]} \times 5 \leq [S3] \leq \frac{[S2]}{[S1]} \times 818$$

④ [D]为脉冲输出点，只能为 Y000 或 Y001，且要求是晶体管输出型。

⑤ 若 X010 由 ON 变为 OFF，停止输出脉冲，X010 再变为 ON 时，从初始重新动作。

⑥ PLSR 和 PLSY 两条指令在程序中只能使用一条，并且只能使用一次。这两条指令中的某一条与 PWM 指令同时使用时，脉冲输出点不能重复。

1 状态初始化指令

(1) 指令格式

状态初始化指令格式如表 6-59 所示。

表 6-59 状态初始化指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D1	D2
状态初始化指令	IST FNC60		X、Y、M、S (8 个连号元件)	S (S20 ~ S899)	

(2) 使用说明

状态初始化指令主要用于步进控制，且在需要进行多种控制时采用，使用这条指令可以使控制程序大大简化，如在机械手控制中，有 5 种控制方式：手动、回原点、单步运行、单周期运行（即运行一次）和自动控制。在程序中采用该指令后，只需编写手动、回原点和自动控制 3 种控制方式的程序即可。

状态初始化指令的使用如图 6-66 所示。当 M8000 由 OFF→ON 时,IST 指令执行，将 X020 为起始编号的 8 个连号元件进行功能定义（具体见后述），将 S20、S40 分别设为自动操作时的编号最小和最大状态继电器。

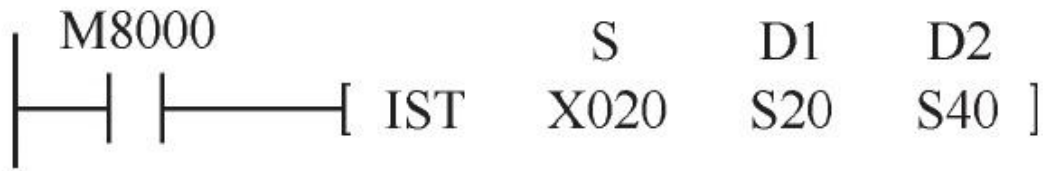


图 6-66 状态初始化指令的使用

状态初始化指令的使用要点如下。

- ① [S]为功能定义起始元件，它包括 8 个连号元件，这 8 个元件的功能定义如下。

X020: 手动控制	X022: 单步运行 控制	X024: 全自动运 行控制	X026: 自动运行 启动
X021: 回原点控 制	X023: 单周期运 行控制	X025: 回原点启 动	X027: 停止控制

其中 X020 ~ X024 是工作方式选择, 不能同时接通, 通常选用图 6-67 所示的旋转开关。

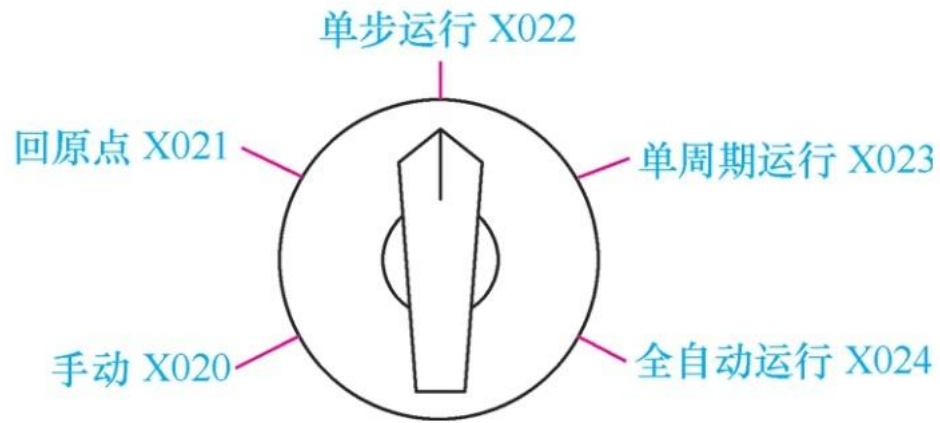


图 6-67 旋转开关

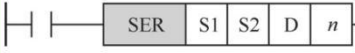
- ② [D1]、[D2]分别为自动操作控制时, 实际用到的最小编号和最大编号状态继电器。
- ③ **IST** 指令在程序中只能用一次, 并且要放在步进顺控指令 **STL** 之前。

2 数据查找指令

(1) 指令格式

数据查找指令格式如表 6-60 所示。

表 6-60 数据查找指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S1	S2	D	n
数据查找指令	SER FNC61		KnX 、 KnY 、 KnM 、 KnS 、 T 、 C 、 D	K 、 H 、 KnX 、 KnY 、 KnM 、 KnS 、 T 、 C 、 D 、 V 、 Z	KnY 、 KnM 、 KnS 、 T 、 C 、 D	K 、 H 、 D

(2) 使用说明

数据查找指令的使用如图 6-68 所示。当常开触点 X010 闭合时，SER 指令执行，从 [S1]D100 为首编号的 10 个连号元件 (D100~D109) 中查找与[S2]D0 相等的数 据，查找结果存放在[D]D10 为首编号的 5 个连号元件 D10~D14 中。



图 6-68 数据查找指令的使用

在 D10~D14 中，D10 存放数据相同的元件个数，D11、D12 分别存放数据相同的第一个和最后一个元件位置，D13 存放最小数据的元件位置，D14 存放最大数据的元件位置。例如，在 D100~D109 中，D100、D102、D106 中的数据都与 D10 相同，D105 中的数据最小，D108 中数据最大，那么 D10=3、D11=0、D12=6、D13=5、D14=8。

3 绝对值式凸轮顺控指令

(1) 指令格式

绝对值式凸轮顺控指令格式如表 6-61 所示。

表 6-61 绝对值式凸轮顺控指令格式

指令名称	助记符与 功能号	指令形式	操作数			
			S1	S2	D	n
绝对值式凸轮顺控 指令	ABSD FNC62		KnX 、 KnY 、 KnM 、 KnS 、 T、C、D	C	Y、M、S	K、H、 ($1 \leq n \leq 64$)

(2) 使用说明

ABSD 指令用于产生与计数器当前值对应的多个波形，其使用如图 6-69 所示。在图 6-69 (a) 中，当常开触点 X000 闭合时，ABSD 指令执行，将[D]M0 为首编号的 4 个连号元件 M0~M3 作为波形输出元件，并将[S2]C0 计数器当前计数值与[S1]D300 为首编号的 8 个连号元件 D300~D307 中的数据进行比较，然后让 M0~M3 输出与 D300~D307 数据相关的波形。

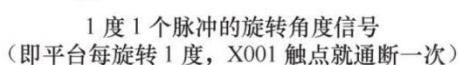
M0~M3 输出波形与 D300~D307 数据的关系如图 6-69 (b) 所示。D300~D307 中的数据可采用 MOV 指令来传送，D300~D307 的偶数编号元件用来存储上升数据点（角度值），奇数编号元件存储下降数据点。下面对照图 6-69 (b) 来说明图 6-69 (a) 梯形图的工作过程。

在常开触点 X000 闭合期间，X001 端子外接平台每旋转 1 度，该端子就输入一个脉冲，X001 常开触点就闭合一次（X001 常闭触点则断开一次），计数器 C0 的计数值就增 1。当平台旋转到 40 度时，C0 的计数值为 40，C0 的计数值与 D300 中的数据相等，ABSD 指令则让 M0 元件由 OFF 变为 ON；当 C0 的计数值为 60 时，C0 的计数值与

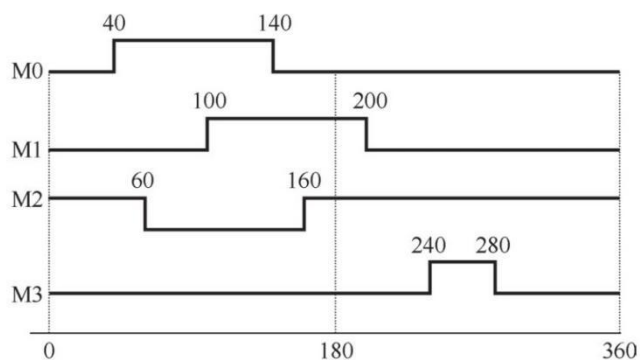
```

graph LR
    X000 --> ABSD
    S1[D300] --> ABSD
    S2[C0] --> ABSD
    D[M0] --> ABSD
    n[K4] --> ABSD
    ABSD --> C0
    C0 --> RST
    X001 --> RST
    RST --> C0
    X001 --> CO
    CO --> K360

```



上升数据点	下降数据点	输出元件
D300=40	D301=140	M0
D302=100	D303=200	M1
D304=160	D305=60	M2
D306=240	D307=280	M3



(b) 输出波形与数据的关系

图 6-69 ABSD 指令的使用

4 增量式凸轮顺控指令

(1) 指令格式

增量式凸轮顺控指令格式如表 6-62 所示。

表 6-62 增量式凸轮顺控指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S1	S2	D	n
增量式凸轮顺控指令	INCD FNC63		KnX 、 KnY 、 KnM 、 KnS 、 T 、 C 、 D	C (两个连号元件)	Y 、 M 、 S	K 、 H 、 ($1 \leq n \leq 64$)

(2) 使用说明

INCD 指令的使用如图 6-70 所示。INCD 指令的功能是将[D]M0 为首编号的 4 个连号元件 M0~M3 作为波形输出元件，并将[S2]C0 当前计数值与[S1]D300 为首编号的 4 个连号元件 D300~D303 中的数据进行比较，让 M0~M3 输出与 D300~D304 数据相关的波形。

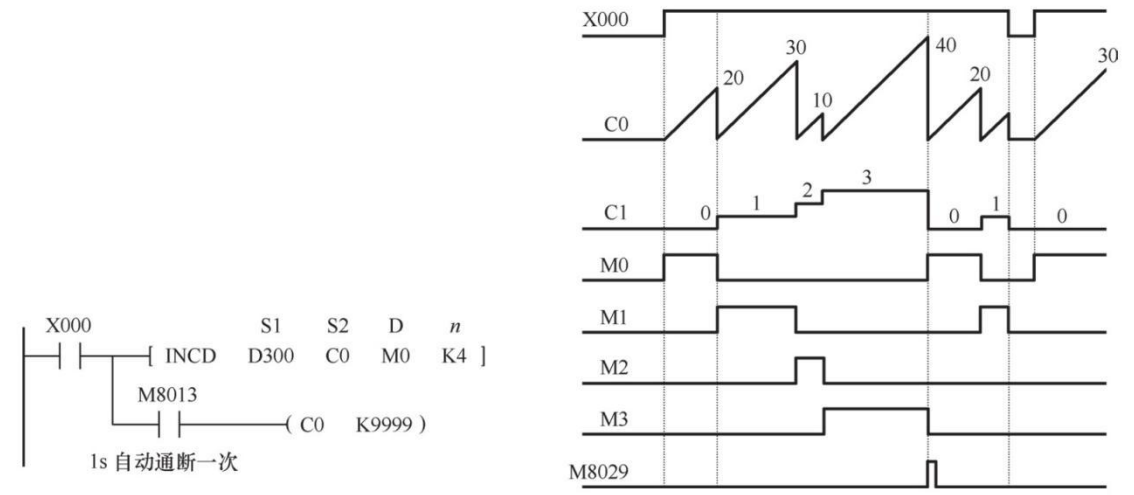


图 6-70 INCD 指令的使用

首先用 MOV 指令往 D300~D303 中传送数据，让 D300=20、D301=30、D302=10、D303=40。在常开触点 X000 闭合期间，时钟辅助继电器 M8013 触点每隔

1s 就通断一次（通断各 0.5s），计数器 C0 的计数值就计 1，随着 M8013 不断动作，C0 计数值不断增大。在 X000 触点刚闭合时，M0 由 OFF 变为 ON，当 C0 计数值与 D300 中的数据 20 相等时，C0 自动复位清 0，同时 M0 元件也复位（由 ON 变为 OFF）；然后 M1 由 OFF 变为 ON，当 C0 计数值与 D301 中的数据 30 相等时，C0 又自动复位，M1 元件随之复位；当 C0 计数值与最后寄存器 D303 中的数据 40 相等时，M3 元件复位，完成标记辅助继电器 M8029 置 ON，表示完成一个周期，接着开始下一个周期。

在 C0 计数的同时，C1 也计数，C1 用来计 C0 的复位次数，完成一个周期后，C1 自动复位。当触点 X000 断开时，C1、C0 均复位，M0~M3 也由 ON 转为 OFF。

5 示教定时器指令

(1) 指令格式

示教定时器指令格式如表 6-63 所示。

表 6-63 示教定时器指令格式

指令名称	助记符与功能号	指令形式	操作数	
			D	n
示教定时器指令	TTMR FNC64		D	K、H、 ($n=0 \sim 2$)

(2) 使用说明

TTMR 指令的使用如图 6-71 所示。TTMR 指令的功能是测定 X010 触点的接通时间。当常开触点 X010 闭合时，TTMR 指令执行，用 D301 存储 X010 触点当前接通时间 t_0 (D301 中的数据随 X010 闭合时间变化，再将 D301 中的时间 t_0 乘以 10^n ，结果存入 D300 中。当触点 X010 断开时，D301 复位，D300 中的数据不变。

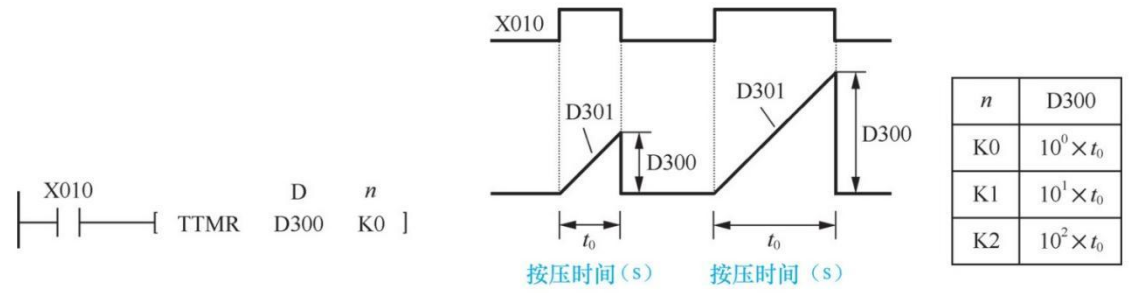


图 6-71 TTMR 指令的使用

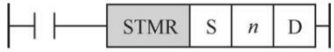
利用 TTMR 指令可以将按钮闭合时间延长 10 倍或 100 倍。

6 特殊定时器指令

(1) 指令格式

特殊定时器指令格式如表 6-64 所示。

表 6-64 特殊定时器指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	<i>n</i>	D
特殊定时器指令	STMR FNC65		T (T0 ~ T199)	K、H <i>n</i> =1 ~ 32767	Y、M、S (4 个连号)

(2) 使用说明

STMR 指令的使用如图 6-72 所示。STMR 指令的功能是产生延时断开定时、单脉冲定时和闪动定时。当常开触点 X000 闭合时,STMR 指令执行,让[D]M0 为首编号的 4 个连号元件 M0~M3 产生 10s 的各种定时脉冲,其中 M0 产生 10s 延时断开定时脉冲,M1 产生 10s 单定时脉冲,M2、M3 产生闪动定时脉冲(即互补脉冲)。

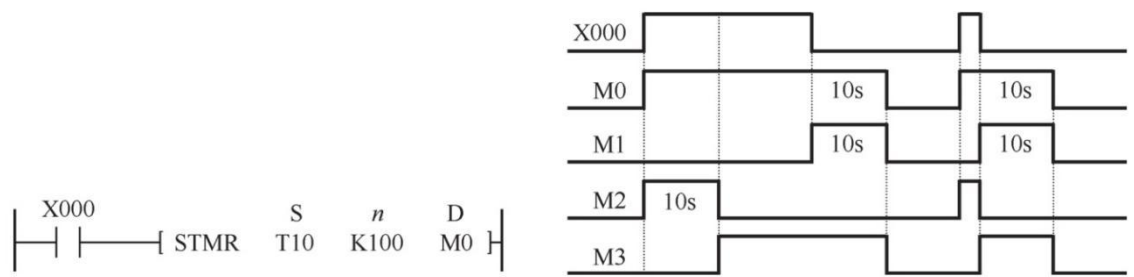


图 6-72 STMR 指令的使用

当触点 X010 断开时, M0~M3 经过设定的值后变为 OFF,同时定时器 T10 复位。

7 交替输出指令

(1) 指令格式

交替输出指令格式如表 6-65 所示。

表 6-65 交替输出指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
交替输出指令	ALT FNC66		Y、M、S

(2) 使用说明

ALT 指令的使用如图 6-73 所示。ALT 指令的功能是产生交替输出脉冲。当常开触点 X000 由 OFF→ON 时, ALTP 指令执行, 让[D]M0 由 OFF→ON, 在 X000 由 ON→OFF 时, M0 状态不变, 当 X000 再一次由 OFF→ON 时, M0 由 ON→OFF。若采用连续执行型 ALT 指令, 在每个扫描周期 M0 状态就会改变一次, 因此通常采用脉冲执行型 ALTP 指令。

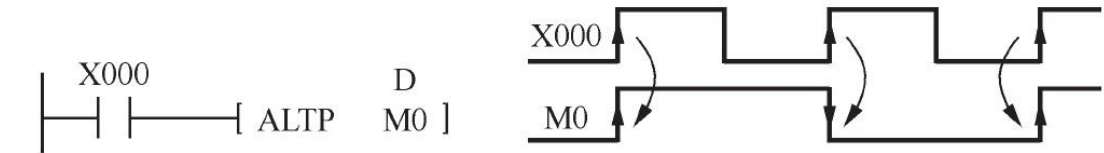


图 6-73 ALT 指令的使用

利用 ALT 指令可以实现分频输出, 如图 6-74 所示, 当 X000 按图示频率通断时, M0 产生的脉冲频率降低一半, 而 M1 产生的脉冲频率较 M0 再降低一半, 每使用一次 ALT 指令可进行一次二分频。

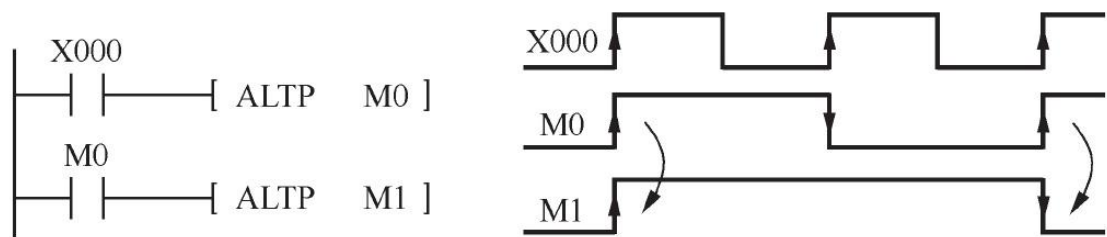


图 6-74 利用 ALT 指令实现分频输出

利用 ALT 指令还可以实现一个按钮控制多个负载启动/停止。如图 6-75 所示，当常开触点 X000 闭合时，辅助继电器 M0 由 OFF→ON，M0 常闭触点断开，Y000 对应的负载停止，M0 常开触点闭合，Y001 对应的负载启动，X000 断开后，辅助继电器 M0 状态不变；当 X000 第二次闭合时，M0 由 ON→OFF，M0 常闭触点闭合，Y000 对应的负载启动，M0 常开触点断开，Y001 对应的负载停止。

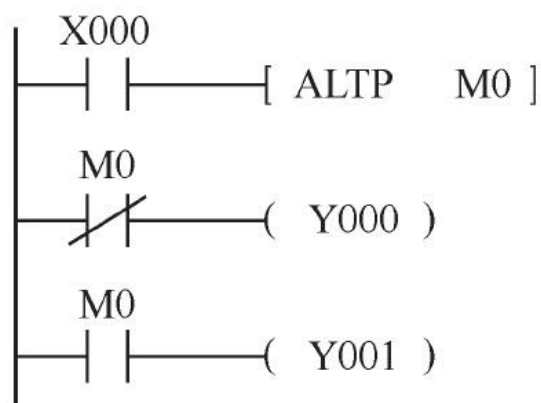


图 6-75 利用 ALT 指令实现一个按钮控制多个负载启动/停止

8 斜波信号输出指令

(1) 指令格式

斜波信号输出指令格式如表 6-66 所示。

表 6-66 斜波信号输出指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S1	S2	D	n
斜波信号输出指令	RAMP FNC67		D			K、H N=1 ~ 32767

(2) 使用说明

RAMP 指令的使用如图 6-76 所示。RAMP 指令的功能是产生斜波信号。当常开触点 X000 闭合时，RAMP 指令执行，让[D]D3 的内容从[S1]D1 的值变化到[S2]D2 的值，变化时间为 1000 个扫描周期，扫描次数存放在 D4 中。

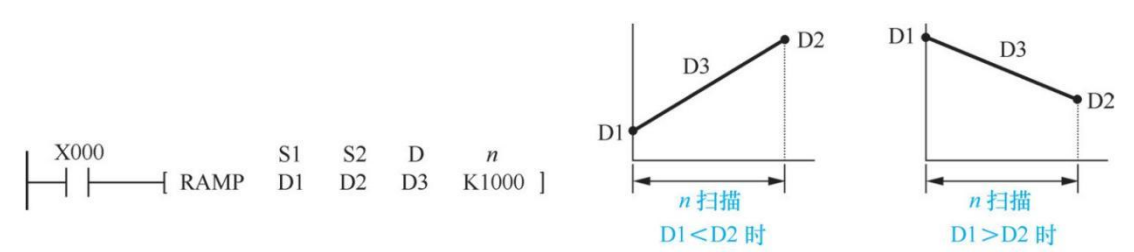


图 6-76 RAMP 指令的使用

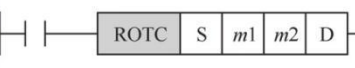
设置 PLC 的扫描周期可确定 D3 (值) 从 D1 变化到 D2 的时间。先往 D8039 (恒定扫描时间寄存器) 写入设定扫描周期时间 (ms) ，设定的扫描周期应大于程序运行扫描时间，再将 M8039 (定时扫描继电器) 置位，PLC 就进入恒扫描周期运行方式。如果设定的扫描周期为 20ms，那么图 6-76 的 D3 (值) 从 D1 变化到 D2 所需的时间应为 20ms×1000=20s。

9 旋转工作台控制指令

(1) 指令格式

旋转工作台控制指令格式如表 6-67 所示。

表 6-67 旋转工作台控制指令格式

指令名称	助记符与 功能号	指令形式	操作数			
			S	m1	m2	D
旋转工作台 控制指令	ROTC FNC68		D (3 个连号 元件)	K、H $m1=2 \sim 32767$	K、H $m2=0 \sim 32767$	Y、M、S (8 个连号 元件)
			$m1 \geq m2$			

(2) 使用说明

ROTC 指令的功能是对旋转工作台的方向和位置进行控制，使工作台上指定的工件能以最短的路径转到要求的位置。图 6-77 所示是一种旋转工作台的结构示意图，它由转台和工作手臂两大部分组成，转台被均分成 10 个区，每个区放置一个工件，转台旋转时会使检测开关 X000、X001 产生两相脉冲，利用这两相脉冲不但可以判断转台正转/反转外，还能检测转台当前旋转位置，检测开关 X002 用来检测转台的 0 位置。

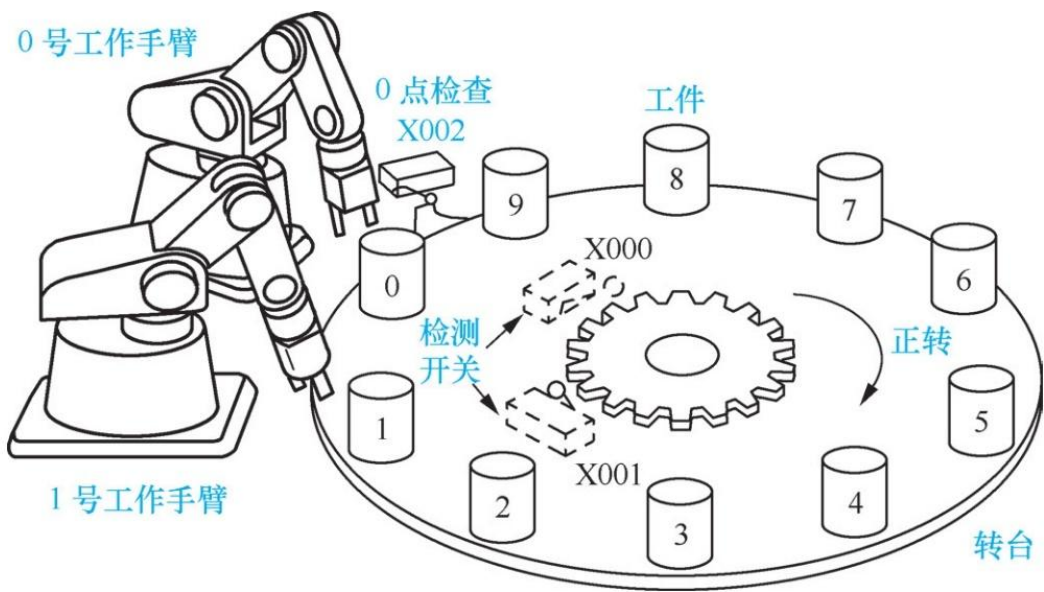


图 6-77 一种旋转工作台的结构示意图

ROTC 指令的使用如图 6-78 所示。



图 6-78 ROTC 指令的使用

在图 6-78 中，当常开触点 X010 闭合时，ROTC 指令执行，对操作数[S]、[m1]、[m2]、[D]的功能进行如下定义。

- [S] {
 - D200: 作为计数寄存器使用
 - D201: 调用工作手臂号
 - D202: 调用工件号} 用传送指令 MOV 设定
- [m1]: 工作台每转一周旋转编码器产生的脉冲数
- [m2]: 低速运行区域，取值一般为 1.5~2 个工件间距
- [D] {
 - M0: A 相信号
 - M1: B 相信号
 - M2: 0 点检测信号
 - M3: 高速正转
 - M4: 低速正转
 - M5: 停止
 - M6: 低速反转
 - M7: 高速反转}
 - 用输入 X（旋转编码器）来驱动，
X000→M0、X001→M1、X002→M2
 - 当 X010 置 ON 时，ROTC 指令执行，可以自动得到 M3~M7 的功能，当 X010 置 OFF 时，M3~M7 为 OFF

(3) ROTC 指令应用实例

有一个图 6-77 所示的旋转工作台，转台均分 10 个区，编号为 0~9，每区可放 1 个工件，转台每转一周两相旋转编码器能产生 360 个脉冲，低速运行区为工件间距的 1.5 倍，采用数字开关输入要加工的工件号，加工采用默认 1 号工作手臂。要求使用 ROTC 指令并将有关硬件进行合适的连接，让工作台能以最高的效率调任意一个工件进行加工。

① 硬件连接

旋转工作台的硬件连接如图 6-79 所示。4 位拨码开关用于输入待加工的工件号，旋转编码器用于检测工作台的位置信息，0 点检测信号用于告知工作台是否到达 0 点位置，启动按钮用于启动工作台运行，Y000 ~ Y004 端子用于输出控制信号，通过控制变频器来控制工作台电机的运行。

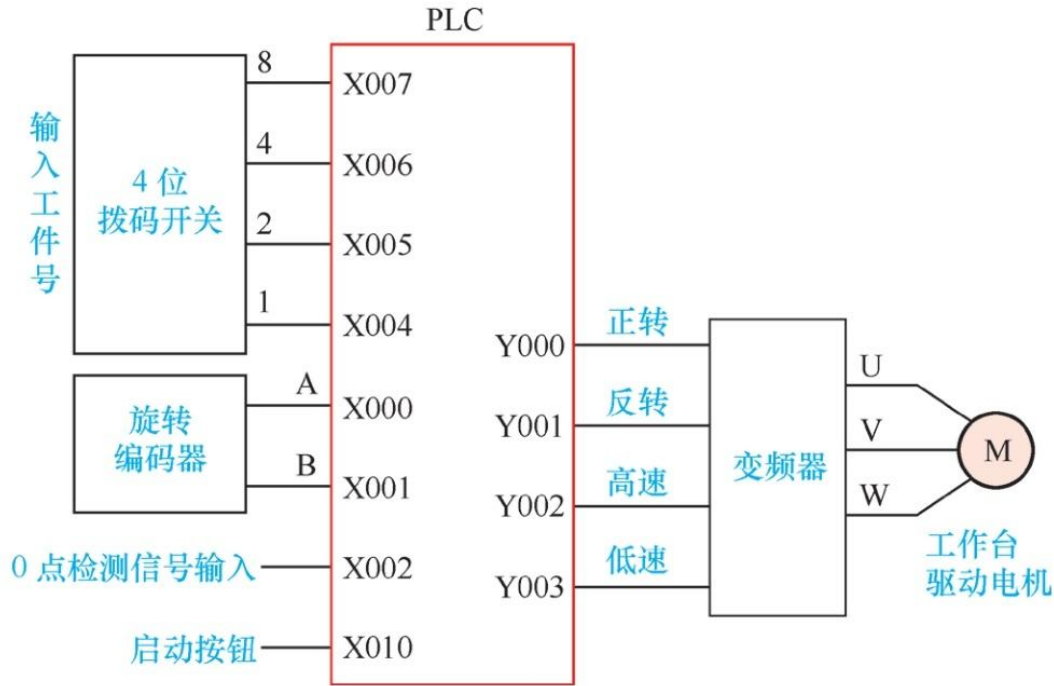


图 6-79 旋转工作台的硬件连接

② 编写程序

旋转工作台控制梯形图程序如图 6-80 所示。在编写程序时要注意，工件号和工作手臂设置与旋转编码器产生的脉冲个数有关，如编码器旋转一个工件间距产生 n 个脉冲，如 $n=10$ ，那么工件号 0~9 应设为 0~90，工作手臂号应设为 0，10。在本例中，旋转编码器转一周产生 360 个脉冲，工作台又分为 10 个区，每个工件间距应产生 36 个脉冲，因此 D201 中的 1 号工作手臂应设为 36，D202 中的工件号就设为「实际工件号 $\times 36$ 」。

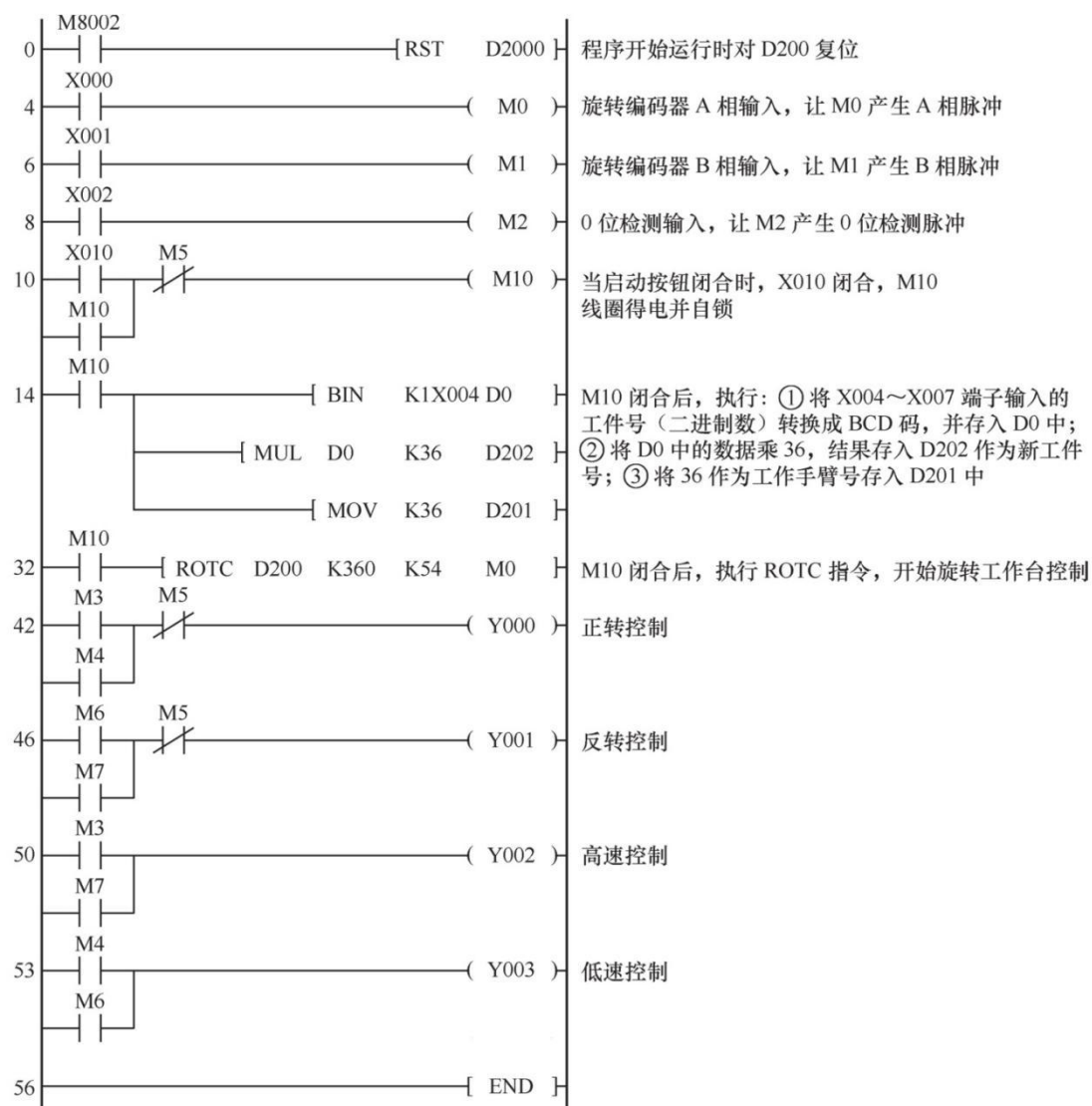


图 6-80 旋转工作台控制梯形图程序

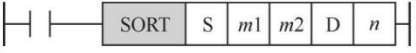
PLC 在进行旋转工作台控制时,在执行 ROTC 指令时,会根据有关程序和输入信号(输入工件号、编码器输入、0 点检测输入和启动输入)产生控制信号(高速、低速、正转、反转),通过变频器来对旋转工作台电动机进行各种控制。

10 数据排序指令

(1) 指令格式

数据排序指令格式如表 6-68 所示。

表 6-68 数据排序指令格式

指令名称	助记符与 功能号	指令形式	操作数				
			S	m1	m2	D	n
数据排序指令	SORT FNC69		D (连号 元件)	K、H $m1=2 \sim 32$ $m1 \geq m2$	K、H $m2=1 \sim 6$	D (连号 元件)	D

(2) 使用说明

SORT 指令的使用如图 6-81 所示。SORT 指令的功能是将[S]D100 为首编号的 [m1]5 行[m2]4 列共 20 个元件（即 D100~D119）中的数据进行排序，排序以 D0 指定的列作为参考，排序按由小到大进行，排序后的数据存入[D]D200 为首编号的 $5 \times 5=20$ 个连号元件中。

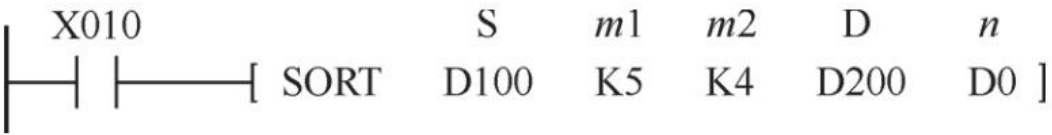


图 6-81 SORT 指令的使用

表 6-69 为排序前 D100~D119 中的数据，若 D0=2，当常开触点 X010 闭合时，SORT 指令执行，将 D100~D119 中的数据以第 2 列作为参考进行由小到大排列，表 6-70 为排序后 D200~D219 中的数据。

表 6-69 排序前 D100~D119 中的数据

行号 \ 列号	1	2	3	4
	人员号码	身长	体重	年龄
1	D100 1	D105 150	D110 45	D115 20
2	D101 2	D106 180	D111 50	D116 40
3	D102 3	D107 160	D112 70	D117 30
4	D103 4	D108 100	D113 20	D118 8
5	D104 5	D109 150	D114 50	D119 45

表 6-70 排序后 D200~D219 中的数据

行号 \ 列号	1	2	3	4
	人员号码	身长	体重	年龄
1	D200 4	D205 100	D210 20	D215 8
2	D201 1	D206 150	D211 45	D216 20
3	D202 5	D207 150	D212 50	D217 45
4	D203 3	D208 160	D213 70	D218 30
5	D204 2	D209 180	D214 50	D219 40

1 十键输入指令

(1) 指令格式

十键输入指令格式如表 6-71 所示。

表 6-71 十键输入指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D1	D2
十键输入指令	TKY FNC70		X、Y、M、S (10 个连号元件)	KnY、KnM、KnS、 T、C、D、V、Z	X、Y、M、S (11 个连号元件)

(2) 使用说明

TKY 指令的使用如图 6-82 所示。在图 6-82 (a) 中，TKY 指令的功能是将以[S]为首编号的 X000 ~ X011 10 个端子输入的数据送入[D1]D0 中，同时将[D2]为首地址的 M10 ~ M19 中相应的位元件置位。

使用 TKY 指令时，可在 PLC 的 X000 ~ X011 10 个端子外接代表 0 ~ 9 的 10 个按键，如图 6-82 (b) 所示，当常开触点 X030 闭合时，如果依次操作 X002、X001、X003、X000，就往 D0 中输入数据 2130，同时与按键对应的位元件 M12、M11、M13、M10 也依次被置 ON，如图 6-82 (c) 所示，当某一按键松开后，相应的位元件还会维持 ON，直到下一个按键被按下才变为 OFF。该指令还会自动用到 M20，当依次操作按键时，M20 会依次被置 ON，ON 的保持时间与按键的按下时间相同。

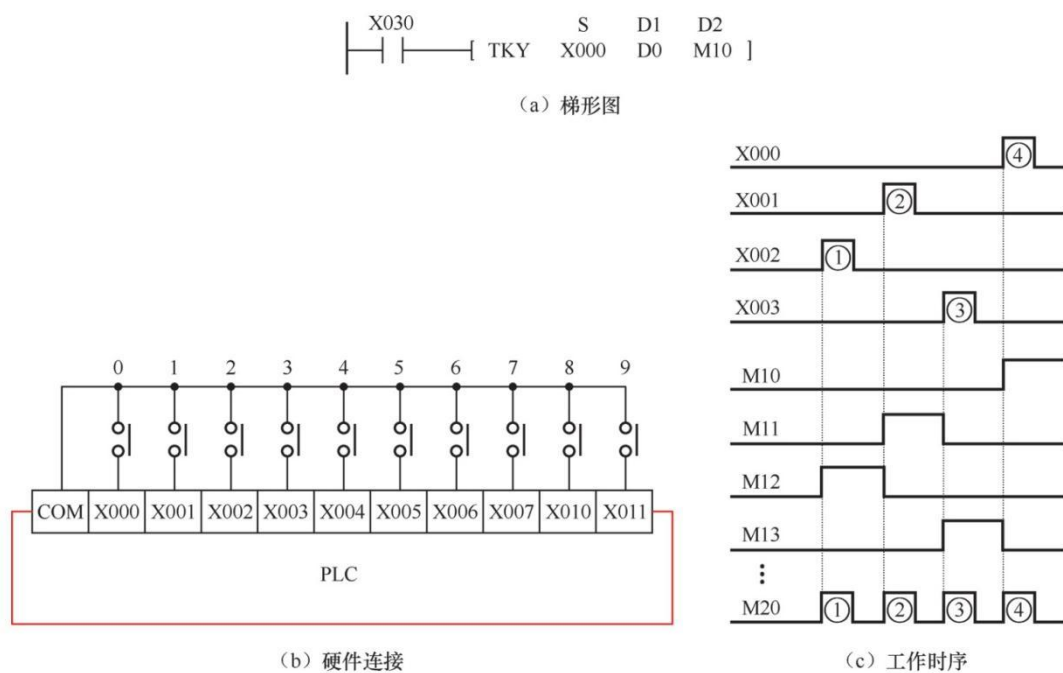


图 6-82 TKY 指令的使用

十键输入指令的使用要点如下。

- ① 若多个按键都按下，先按下的键有效。
- ② 当常开触点 X030 断开时，M10 ~ M20 都变为 OFF，但 D0 中的数据不变。
- ③ 在做 16 位操作时，输入数据范围为 0 ~ 9999，当输入数据超过 4 位，最高位数（千位数）会溢出，低位补入；在做 32 位操作时，输入数据范围为 0 ~ 99999999。

2 十六键输入指令

(1) 指令格式

十六键输入指令格式如表 6-72 所示。

表 6-72 十六键输入指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S	D1	D2	D3
十六键输入指令	HKY FNC71		X (4个连号元件)	Y	T、C、D、 V、Z	Y、M、S (8个连号元件)

(2) 使用说明

HKY 指令的使用如图 6-83 所示。在使用 HKY 指令时，一般要给 PLC 外围增加键盘输入电路，如图 6-83 (b) 所示。HKY 指令的功能是将[S]为首编号的 X000~X003 4 个端子作为键盘输入端，将以[D1]为首编号的 Y000~Y003 4 个端子作为 PLC 的扫描键盘输出端，[D2]指定的元件 D0 用来存储键盘输入信号，[D3]指定的以 M0 为首编号的 8 个元件 M0~M7 用来响应功能键 A~F 输入信号。

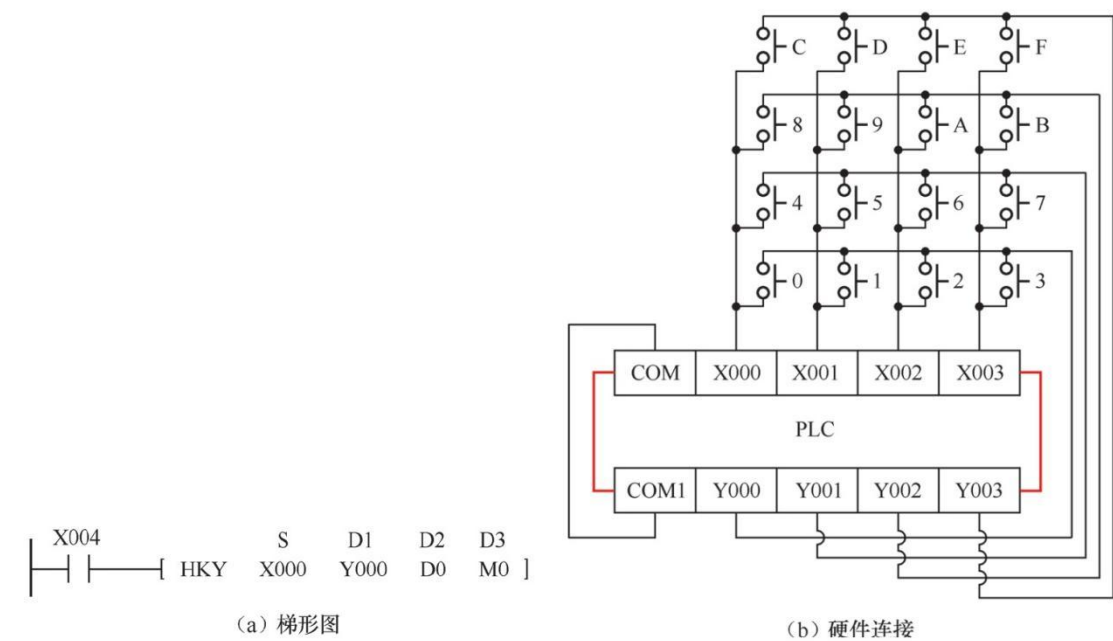


图 6-83 HKY 指令的使用

十六键输入指令的使用要点如下。

① 利用 0~9 数字键可以输入 0~9999 数据，输入的数据以 BIN 码（二进制数）形式保存在[D2]D0 中，若输入数据大于 9999，则数据的高位溢出。若使用 32 位操作 DHKY 指令时，可输入 0~99999999，数据保存在 D1、D0 中。按下多个按键时，先按下的键有效。

② Y000~Y003 完成一次扫描工作后，完成标记继电器 M8029 会置位。

③ 当操作功能键 A~F 时，M0~M7 会有相应的动作，A~F 与 M0~M5 的对应关系如图 6-84 所示。

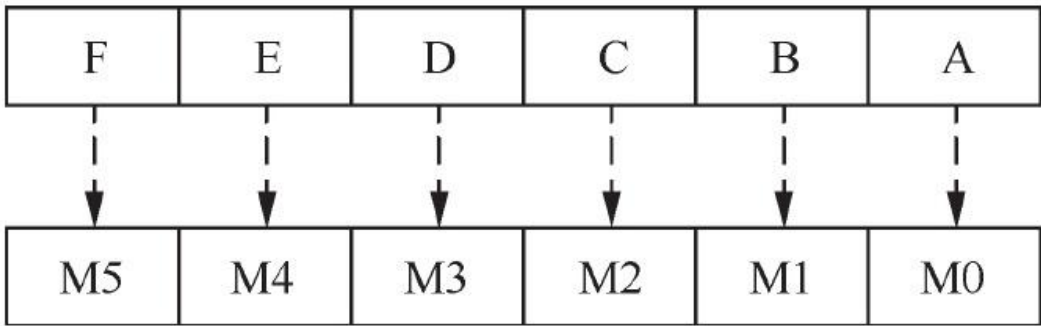


图 6-84 A~F 与 M0~M5 的对应关系

如按下 A 键时，M0 置 ON 并保持，当按下另一键时，如按下 D 键，M0 变为 OFF，同时 D 键对应的元件 M3 置 ON 并保持。

④ 在按下 A~F 某键时，M6 置 ON（不保持），松开按键 M6 由 ON 转为 OFF；在按下 0~9 某键时，M7 置 ON（不保持）。当常开触点 X004 断开时，[D2]D0 中的数据仍保存，但 M0~M7 全变为 OFF。

⑤ 如果将 M8167 置 ON，那么可以通过键盘输入十六进制数并保存在[D2]D0 中。

如操作键盘输入 123BF，那么该数据会以二进制形式保存在[D2]中。

⑥ 键盘一个完整扫描过程需要 8 个 PLC 扫描周期，为防止键盘输入滤波延时造成存储错误，要求使用恒定扫描模式或定时中断处理。

3 数字开关指令

(1) 指令格式

数字开关指令格式如表 6-73 所示。

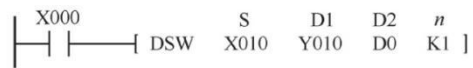
表 6-73 数字开关指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S	D1	D2	n
数字开关指令	DSW FNC72		X (4个连号元件)	Y	T、C、D、 V、Z	K、H n=1、2

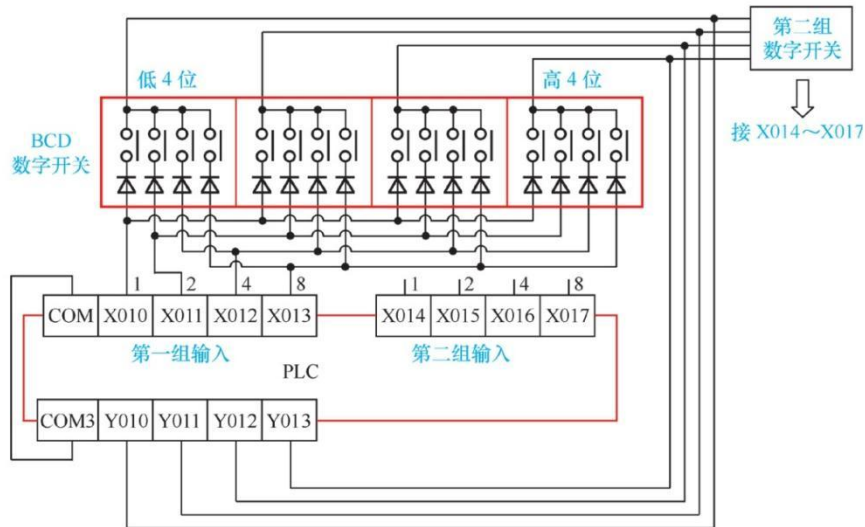
(2) 使用说明

DSW 指令的使用如图 6-85 所示。DSW 指令的功能是读入一组或两组 4 位数字开关的输入值。[S]指定键盘输入端的首编号，将首编号为起点的 4 个连号端子 X010~X013 作为键盘输入端；[D1]指定 PLC 扫描键盘输出端的首编号，将首编号为起点的 4 个连号端子 Y010~Y013 作为扫描输出端；[D2]指定数据存储元件；[n]指定数字开关的组数，n=1 表示一组，n=2 表示两组。

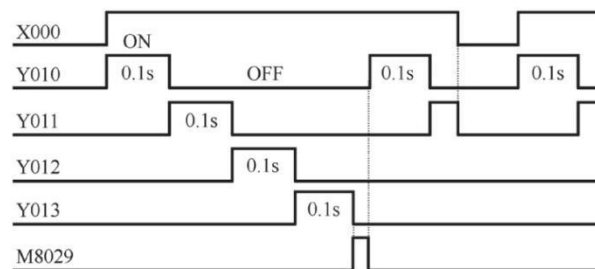
在使用 DSW 指令时，须给 PLC 外接相应的数字开关输入电路。PLC 与一组数字开关连接电路如图 6-85 (b) 所示。在常开触点 X000 闭合时，DSW 指令执行，PLC 从 Y010~Y013 端子依次输出扫描脉冲，如果数字开关设置的输入值为 1101 0110 1011 1001 (数字开关某位闭合时，表示该位输入 1)。当 Y010 端子为 ON 时，数字开关的低 4 位往 X013~X010 输入 1001，1001 被存入 D0 低 4 位；当 Y011 端子为 ON 时，数字开关的次低 4 位往 X013~X010 输入 1011，该数被存入 D0 的次低 4 位，一个扫描周期完成后，1101 0110 1011 1001 全被存入 D0 中，同时将标志继电器 M8029 置 ON。



(a) 梯形图



(b) 硬件连接



(c) 工作时序

图 6-85 DSW 指令的使用

如果需要使用两组数字开关，可将第二组数字开关一端与 X014~X017 连接，另一端则和第一组一样与 Y010~Y013 连接，当将 n 设为 2 时，第二组数字开关输入值通过 X014~X017 存入 D1 中。

4 七段译码指令

(1) 指令格式

七段译码指令格式如表 6-74 所示。

表 6-74 七段译码指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
七段译码指令	SEGD FNC73		K、H、KnY、KnM、KnS、T、C、D、V、Z	KnY、KnM、KnS、T、C、D、V、Z

(2) 使用说明

SEGD 指令的使用如图 6-86 所示。SEGD 指令的功能是将源操作数[S]D0 中的低 4 位二进制数（代表十六进制数 0~F）转换成七段显示格式的数据，再保存在目标操作数 [D]Y000~Y007 中，源操作数中的高位数不变。4 位二进制数与七段显示格式数对应关系见表 6-75。



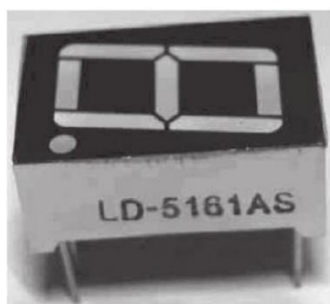
图 6-86 SEG D 指令的使用

表 6-75 4 位二进制数与七段显示格式数对应关系

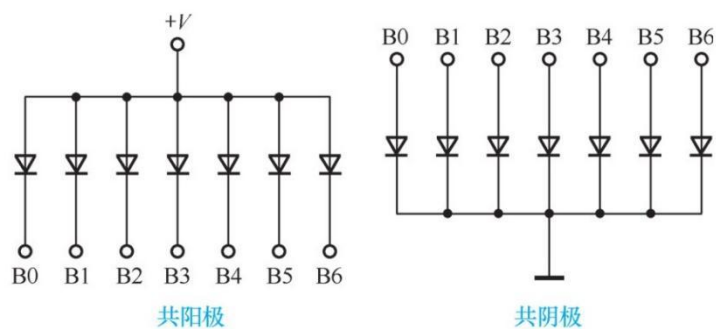
[S]		七段码构成	[D]								显示数据
十六进制	二进制		B7	B6	B5	B4	B3	B2	B1	B0	
0	0000		0	0	1	1	1	1	1	1	0
1	0001		0	0	0	0	0	1	1	0	1
2	0010		0	1	0	1	1	0	1	1	2
3	0011		0	1	0	0	1	1	1	1	3
4	0100		0	1	1	0	0	1	1	0	4
5	0101		0	1	1	0	1	1	0	1	5
6	0110		0	1	1	1	1	1	0	1	6
7	0111		0	0	1	0	0	1	1	1	7
8	1000		0	1	1	1	1	1	1	1	8
9	1001		0	1	1	0	1	1	1	1	9
A	1010		0	1	1	1	0	1	1	1	A
B	1011		0	1	1	1	1	1	0	0	b
C	1100		0	0	1	1	1	0	0	1	c
D	1101		0	1	0	1	1	1	1	0	d
E	1110		0	1	1	1	1	0	0	1	e
F	1111		0	1	1	1	0	0	0	1	F

(3) 用 SEGD 指令驱动七段码显示器

利用 SEGD 指令可以驱动七段码显示器显示字符，七段码显示器外形与结构如图 6-87 所示，由 7 个发光二极管排列成「8」字形，根据发光二极管共用电极不同，可分为共阳极和共阴极两种。PLC 与七段码显示器的连接如图 6-88 所示。在图 6-86 所示的梯形图中，设 D0 的低 4 位二进制数为 1001，当常开触点 X000 闭合时，SEGD 指令执行，1001 被转换成七段显示格式数据 01101111，该数据存入 Y007 ~ Y000，Y007 ~ Y000 端子输出 01101111，七段码显示管 B6、B5、B3、B2、B1、B0 段亮（B4 段不亮），显示十进制数「9」。



(a) 外形



(b) 结构

图 6-87 七段码显示器外形与结构

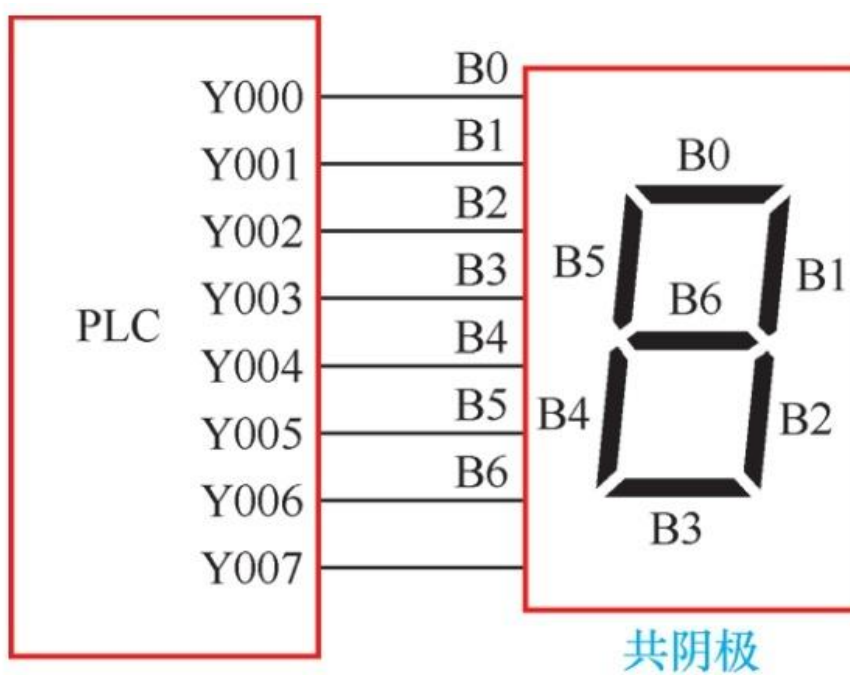


图 6-88 PLC 与七段码显示器的连接

5 带锁存的七段码显示指令

(1) 关于带锁存的七段码显示器

普通的七段码显示器显示一位数字需用到 8 个端子来驱动，若显示多位数字时要用到大量引线，很不方便。采用带锁存的七段码显示器可实现用少量几个端子来驱动显示多位数字。带锁存的七段码显示器与 PLC 的连接如图 6-89 所示。下面以显示 4 位十进制数 1836 为例来说明电路工作原理。

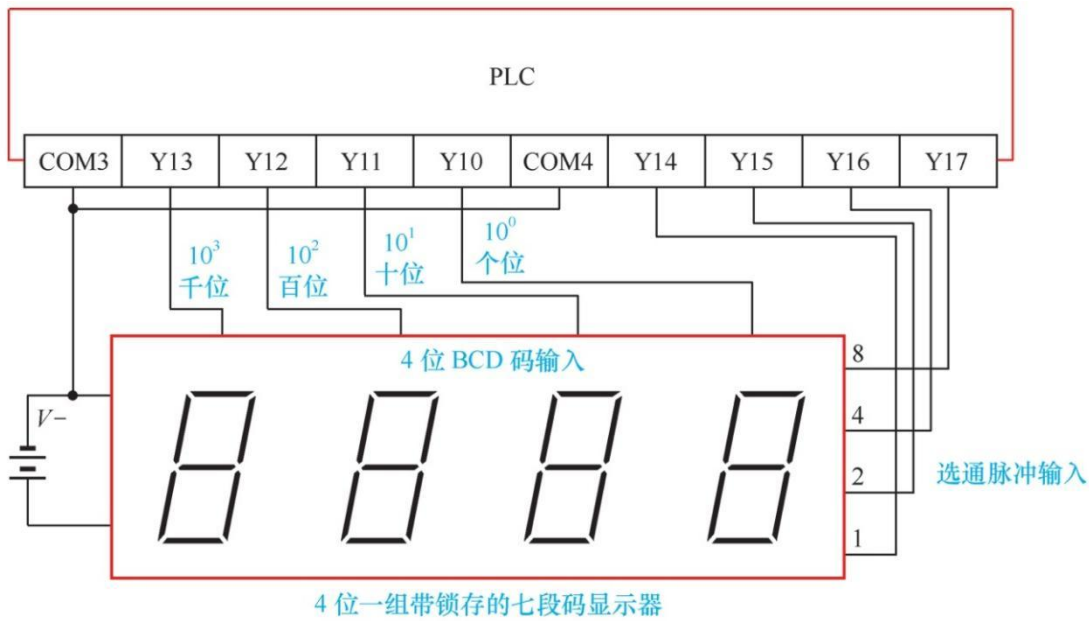


图 6-89 带锁存的七段码显示器与 PLC 的连接

首先 Y13、Y12、Y11、Y10 端子输出 6 的 BCD 码 0110 到显示器，经显示器内部电路转换成 6 的七段码格式数据 01111101，与此同时 Y14 端子输出选通脉冲，该选通脉冲使显示器的个位数显示有效（其他位不能显示），显示器个数显示 6；然后 Y13、Y12、Y11、Y10 端子输出 3 的 BCD 码 0011 到显示器，给显示器内部电路转换成 3 的七段码格式数据「01001111」，同时 Y15 端子输出选通脉冲，该选通脉冲使显示器的十位数显示有效，显示器十位数显示 3；在显示十位的数字时，个位数的七段码数据被锁存

下来，故个位的数字仍显示，采用同样的方法依次让显示器百、千位分别显示 8、1，结果就在显示器上显示出 1836。

(2) 带锁存的七段码显示指令格式

带锁存的七段码显示指令格式如表 6-76 所示。

表 6-76 带锁存的七段码显示指令格式

指令名称	助记符与 功能号	指令形式	操作数		
			S	D	n
带锁存的七段码 显示指令	SEGL FNC74		K、H、KnY、KnM、 KnS、T、C、D、V、Z	Y	K、H (一组时 n=0 ~ 3, 两组时 n=4 ~ 7)

(3) 使用说明

SEGL 指令的使用如图 6-90 所示，当 X000 闭合时，SEGL 指令执行，将源操作数 [S]D0 中数据 (0 ~ 9999) 转换成 BCD 码并形成选通信号，再从目标操作数[D]Y010 ~ Y017 端子输出，去驱动带锁存功能的七段码显示器，使之以十进制形式直观显示 D0 中的数据。

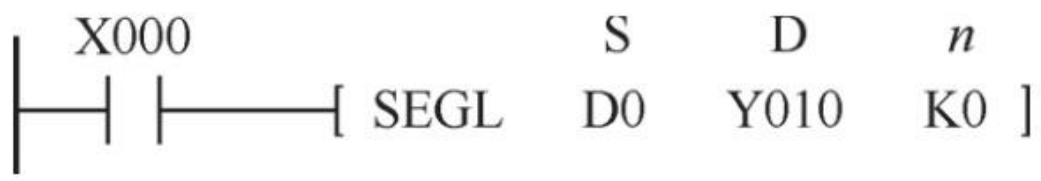


图 6-90 SEGL 指令的使用

指令中[n]的设置与 PLC 输出类型、BCD 码和选通信号有关，具体见表 6-77。例如，PLC 的输出类型 = 负逻辑(即输出端子内接 NPN 型三极管)、显示器输入数据类型 = 负逻辑 (如 6 的负逻辑 BCD 码为 1001，正逻辑为 0110)、显示器选通脉冲类型 = 正

逻辑（即脉冲为高电平），若是接 4 位一组显示器，则 $n=1$ ；若是接 4 位两组显示器，则 $n=5$ 。

表 6-77 PLC 输出类型、BCD 码、选通信号与[n]的设置关系

PLC 输出类型		显示器数据输入类型		显示器选通脉冲类型		n 取值	
PNP	NPN	高电平有效	低电平有效	高电平有效	低电平有效	4 位一组	4 位两组
正逻辑	负逻辑	正逻辑	负逻辑	正逻辑	负逻辑		
	√	√		√		3	7
	√	√			√	2	6
	√		√	√		1	5
	√		√		√	0	4
√		√		√		0	4
√		√			√	1	5
√			√	√		2	6
√			√		√	3	7

(4) 4 位两组带锁存的七段码显示器与 PLC 的连接

4 位两组带锁存的七段码显示器与 PLC 的连接如图 6-91 所示，在执行 SEGL 指令时，显示器可同时显示 D10、D11 中的数据，其中 Y13~Y10 端子所接显示器显示 D10 中的数据，Y23~Y20 端子所接显示器显示 D11 中的数据，Y14~Y17 端子输出扫描脉冲（即依次输出选通脉冲），Y14~Y17 完成一次扫描后，标志继电器 M8029 会置 ON。Y14~Y17 端子输出的选通脉冲是同时送到两组显示器的，如 Y14 端输出选通脉冲时，两显示器分别接收 Y13~Y10 和 Y23~Y20 端子送来的 BCD 码，并在内部转换成七段码格式数据，再驱动各自的各位显示数字。

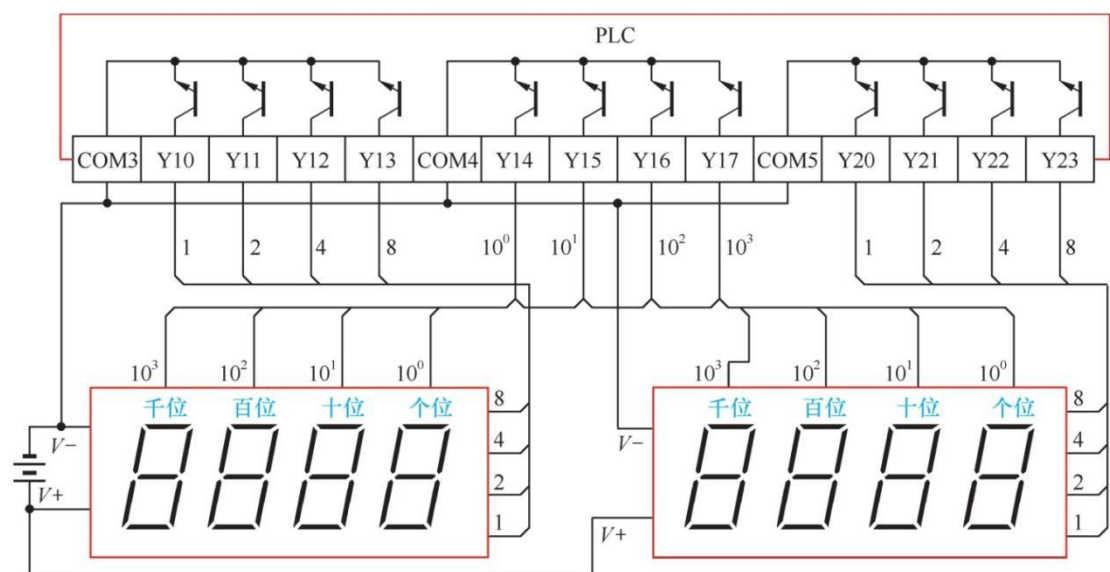


图 6-91 4 位两组带锁存的七段码显示器与 PLC 的连接

6 方向开关指令

(1) 指令格式

方向开关指令格式如表 6-78 所示。

表 6-78 方向开关指令格式

指令名称	助记符与 功能号	指令形式	操作数			
			S	D1	D2	n
方向开关指令	ARWS FNC75		X、Y、M、S	T、C、D、 V、Z	Y	K、H (n=0~3)

(2) 使用说明

ARWS 指令的使用如图 6-92 所示。ARWS 指令不但可以像 SEGL 指令一样，能将 [D1]D0 中的数据通过[D2]Y000~Y007 端子驱动带锁存的七段码显示器显示出来，还可以利用[S]指定的 X010~X013 端子输入来修改[D]D0 中的数据。[n]的设置与 SEGL 指令相同，见表 6-76。

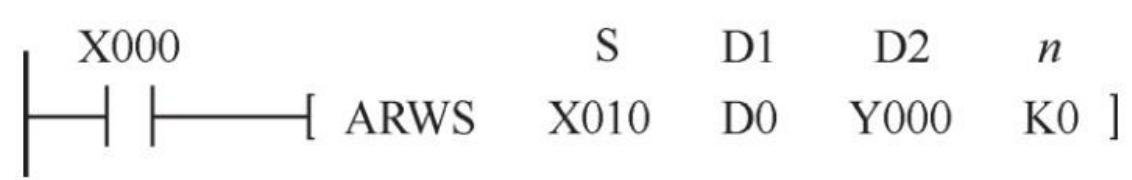


图 6-92 ARWS 指令的使用

利用 ARWS 指令驱动并修改带锁存的七段码显示器与 PLC 的连接电路如图 6-93 所示。当常开触点 X000 闭合时，ARWS 指令执行，将 D0 中的数据转换成 BCD 码并形成选通脉冲，从 Y0~Y7 端子输出，驱动带锁存的七段码显示器显示 D0 中的数据。

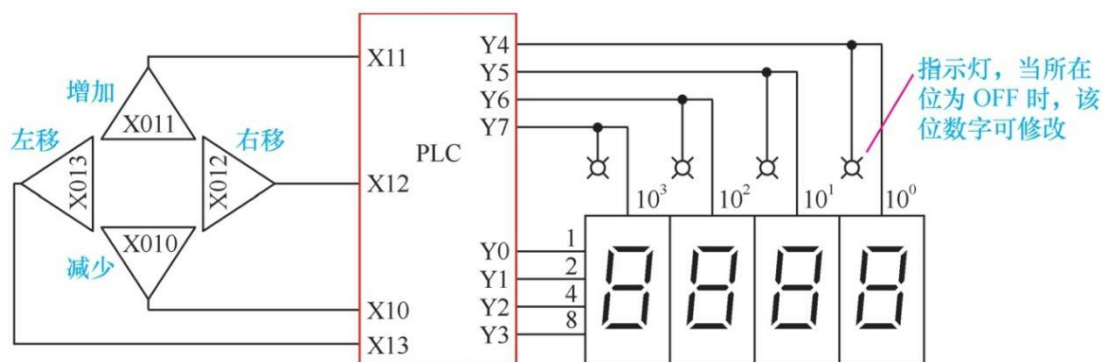


图 6-93 利用 ARWS 指令驱动并修改带锁存的七段码显示器与 PLC 的连接电路

如果要修改显示器显示的数字（即修改 D0 中的数据），可操作 X10~X13 端子外接的按键。显示器千位默认是可以修改的（即 Y7 端子默认处于 OFF），按压增加键 X11 或减小键 X10 可以将数字调大或调小，按压右移键 X12 或左移键 X13 可以改变修改位，连续按压右移键时，修改位变化为 $10^3 \rightarrow 10^2 \rightarrow 10^1 \rightarrow 10^0$ ，当某位所在的指示灯为 OFF 时，该位可以修改。

ARWS 指令在程序中只能使用一次，且要求 PLC 为晶体管输出型。

7 ASCII 码转换指令

(1) 指令格式

ASCII 码转换指令格式如表 6-79 所示。

表 6-79 ASCII 码转换指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
ASCII 码转换指令	ASC FNC76		8 个以下的字母或数字	T、C、D

(2) 使用说明

ASC 指令的使用如图 6-94 所示。当常开触点 X000 闭合时，ASC 指令执行，将 ABCDEFGH 这 8 个字母转换成 ASCII 码并存入 D300~D303 中。如果将 M8161 置 ON 后再执行 ASC 指令，ASCII 码只存入[D]低 8 位（要占用 D300~D307）。M8161 处于不同状态时，ASCII 码存储位置如图 6-95 所示。

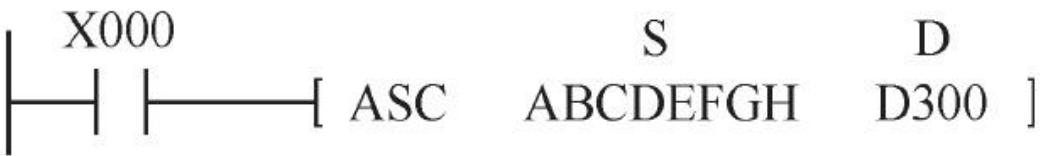
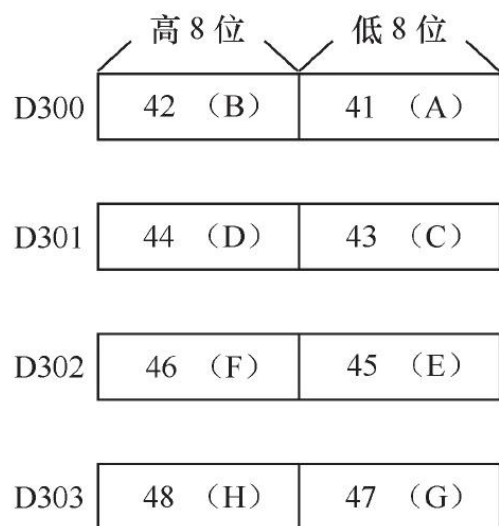


图 6-94 ASC 指令的使用



(a) M8161=OFF

	高 8 位	低 8 位	
D300	00	41	A
D301	00	42	B
D302	00	43	C
D303	00	44	D
D304	00	45	E
D305	00	46	F
D306	00	47	G
D307	00	48	H

(b) M8161=ON

图 6-95 M8161 处于不同状态时 ASCII 码的存储位置

8 ASCII 码打印输出指令

(1) 指令格式

ASCII 码打印输出指令格式如表 6-80 所示。

表 6-80 ASCII 码打印输出指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
ASCII 码打印输出指令	PR FNC77		8 个以下的字母或数字	T、C、D

(2) 使用说明

PR 指令的使用如图 6-96 所示。当常开触点 X000 闭合时, PR 指令执行, 将 D300 为首编号的几个连号元件中的 ASCII 码从 Y000 为首编号的几个端子输出。在输出 ASCII 码时, 先从 Y000~Y007 端输出 A 的 ASCII 码 (由 8 位二进制数组成), 然后输出 B、C...H, 在输出 ASCII 码的同时, Y010 端会输出选通脉冲, Y011 端输出正在执行标志, 如图 6-96 (b) 所示, Y010、Y011 端输出信号去 ASCII 码接收电路, 使之能正常接收 PLC 发出的 ASCII 码。

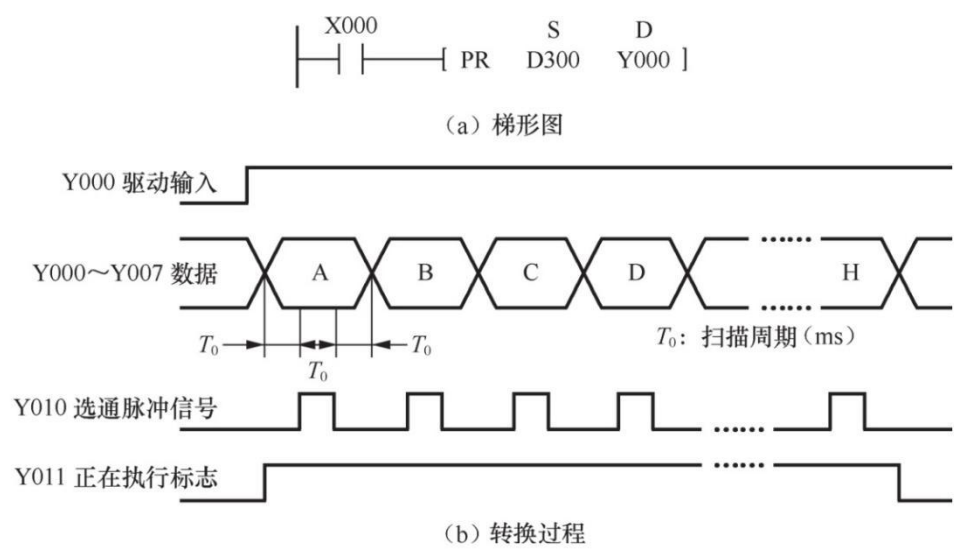


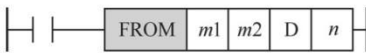
图 6-96 PR 指令的使用

9 读特殊功能模块指令

(1) 指令格式

读特殊功能模块指令格式如表 6-81 所示。

表 6-81 读特殊功能模块指令格式

指令名称	助记符与 功能号	指令形式	操作数			
			<i>m1</i>	<i>m2</i>	D	<i>n</i>
读特殊功能 模块指令	FROM FNC78		K、H <i>m1</i> =0 ~ 7	K、H <i>m2</i> =0 ~ 32767	<i>KnY</i> 、 <i>KnM</i> 、 <i>KnS</i> 、T、C、 D、V、Z	K、H <i>n</i> =0 ~ 32767

(2) 使用说明

FROM 指令的使用如图 6-97 所示。当常开触点 X000 闭合时，FROM 指令执行，将[*m1*]单元号为 1 的特殊功能模块中的[*m2*]29 号缓冲存储器（BFM）中的[*n*]16 位数据读入 K4M0（M0 ~ M16）。

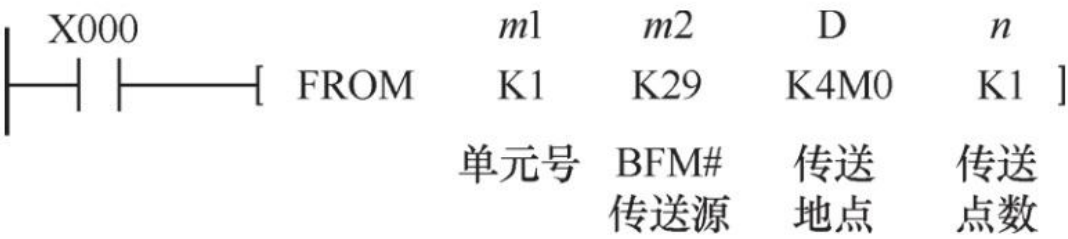


图 6-97 FROM 指令的使用

当 X000=ON 时，执行 FROM 指令；当 X000=OFF 时，不传送数据，传送地点的数据不变。脉冲指令执行也一样。

10 写特殊功能模块指令

(1) 指令格式

写特殊功能模块指令格式如表 6-82 所示。

表 6-82 写特殊功能模块指令格式

指令名称	助记符与功能号	指令形式	操作数			
			<i>m1</i>	<i>m2</i>	D	<i>n</i>
写特殊功能模块指令	TO FNC79		K、H <i>m1</i> =0 ~ 7	K、H <i>m2</i> =0 ~ 32767	<i>KnY</i> 、 <i>KnM</i> 、 <i>KnS</i> 、T、C、 D、V、Z	K、H <i>n</i> =0 ~ 32767

(2) 使用说明

TO 指令的使用如图 6-98 所示。当常开触点 X000 闭合时，TO 指令执行，将[D]D0 中的[*n*]16 位数据写入[*m1*]单元号为 1 的特殊功能模块中的[*m2*]12 号缓冲存储器 (BFM) 中。

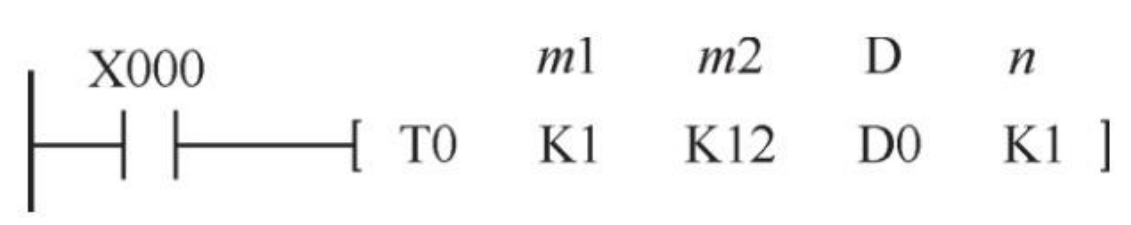


图 6-98 TO 指令的使用

1 串行数据传送指令

(1) 指令格式

串行数据传送指令格式如表 6-83 所示。

表 6-83 串行数据传送指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S	<i>m</i>	D	<i>n</i>
串行数据传送指令	RS FNC80		D	K、H、D	D	K、H、D

(2) 使用说明

① 指令的使用形式

利用 RS 指令可以让两台 PLC 之间进行数据交换, 首先使用 FX_{2N}-485-BD 通信版将两台 PLC 连接好, 如图 6-99 所示。RS 指令的使用形式如图 6-100 所示, 当常开触点 X000 闭合时, RS 指令执行, 将[S]D200 为首编号的[*m*]D0 个寄存器中的数据传送给 [D]D500 为首编号的[*n*]D1 个寄存器。

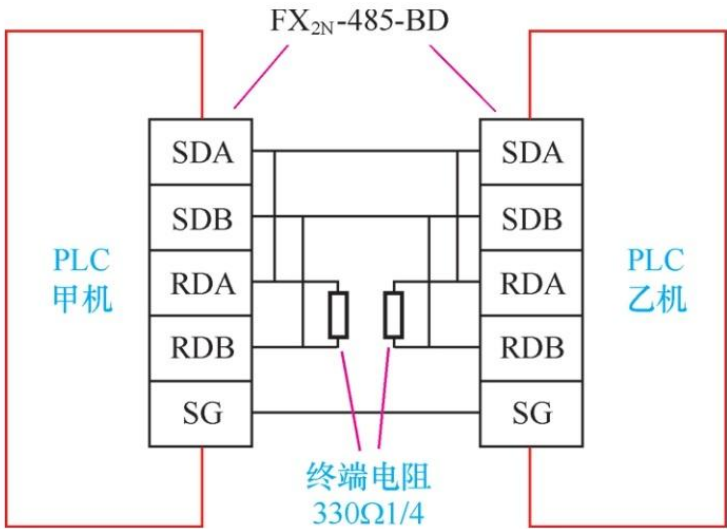


图 6-99 利用 RS 指令通信时的两台 PLC 的连接

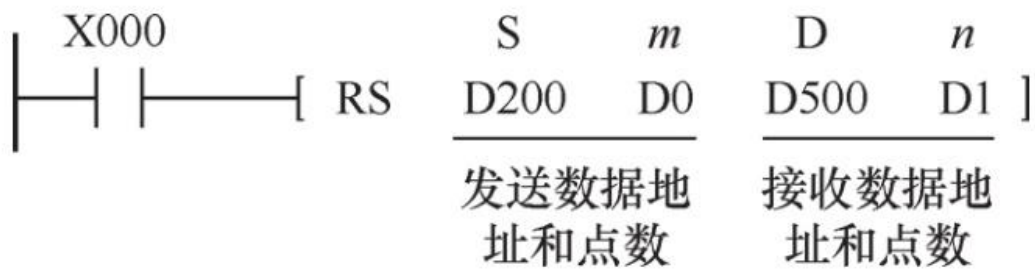


图 6-100 RS 指令的使用形式

② 定义发送数据的格式

在使用 RS 指令发送数据时，先要定义发送数据的格式，设置特殊数据寄存器 D8120 的各位可以定义发送数据格式。D8120 的各位与数据格式关系见表 6-84。例如，要求发送的数据格式为：数据长 =7 位、奇偶校验 = 奇校验、停止位 =1 位、传输速度 =19200、无起始和终止符。D8120 的各位应进行如下设置。

	b15		b12 b11				b8 b7		b4 b3				b0		
D8120	0		0	0	0	0	0	1	0	0	1	0	0	1	0
↓	0		0				9		2						
D8120 = 0092H															

表 6-84 D8120 的各位与数据格式的关系

位号	名称	内容	
		0	1
b0	数据长	7 位	8 位
b1 b2	奇偶校验	b2, b1 (0, 0): 无校验 (0, 1): 奇校验 (1, 1): 偶校验	
b3	停止位	1 位	2 位
b4 b5 b6 b7	传送速率 (bit/s)	b7, b6, b5, b4 (0, 0, 1, 1): 300 (0, 1, 0, 0): 600 (0, 1, 0, 1): 1,200 (0, 1, 1, 0): 2,400 (0, 0, 1, 1): 4,800 (1, 0, 0, 0): 9,600 (1, 0, 0, 1): 19,200	
b8	起始符	无	有 (D8124)
b9	终止符	无	有 (D8125)
b10 b11	控制线	通常固定设为 00	
b12	不可使用（固定为 0）		
b13	和校验	通常固定设为 000	
b14	协议		
b15	控制顺序		

要将 D8120 设为 0092H，可采用图 6-101 所示的梯形图程序，当常开触点 X001 闭合时，MOV 指令执行，将十六进制数 0092 送入 D8120 中（指令会自动将十六进制数 0092 转换成二进制数，再送入 D8120 中）。



图 6-101 将 D8120 设为 0092H 的梯形图程序

③ 指令的使用说明

图 6-102 所示为一个典型的 RS 指令使用程序。

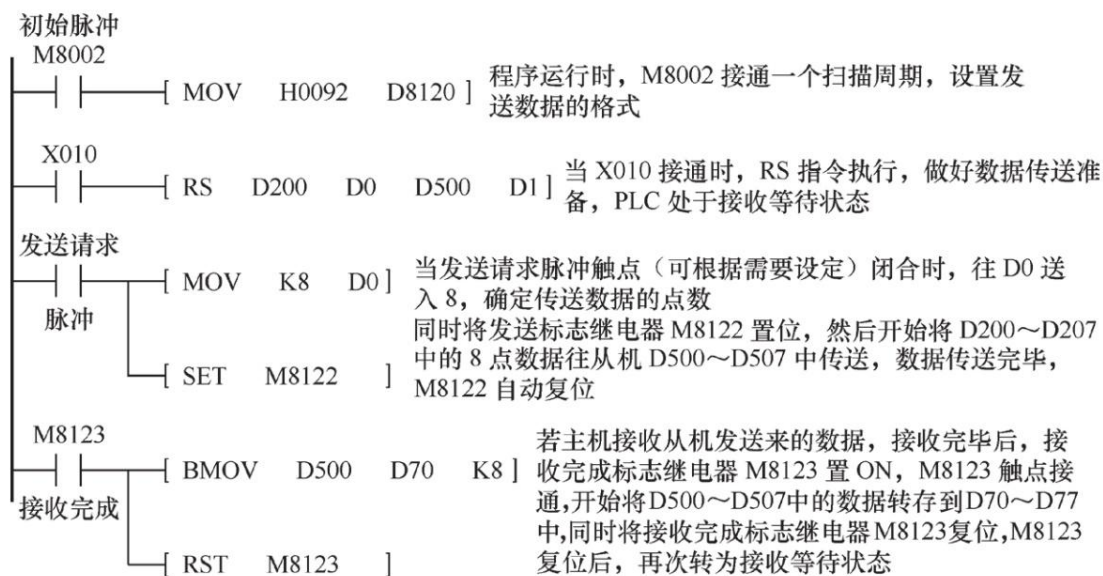


图 6-102 一个典型的 RS 指令使用程序

2 八进制位传送指令

(1) 指令格式

八进制位传送指令格式如表 6-85 所示。

表 6-85 八进制位传送指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
八进制位传送指令	PRUN FNC81		K_nX 、 K_nM ($n=1 \sim 8$, 元件最低位要为 0)	K_nX 、 K_nM ($n=1 \sim 8$, 元件最低位要为 0)

(2) 使用说明

PRUN 指令的使用如图 6-103 所示，以图 6-103 (a) 为例，当常开触点 X030 闭合时，PRUN 指令执行，将[S]位元件 X000~X007、X010~X017 中的数据分别送入[D]位元件 M0~M7、M10~M17 中，由于 X 采用八进制编号，而 M 采用十进制编号，尾数为 8、9 的继电器 M 自动略过。

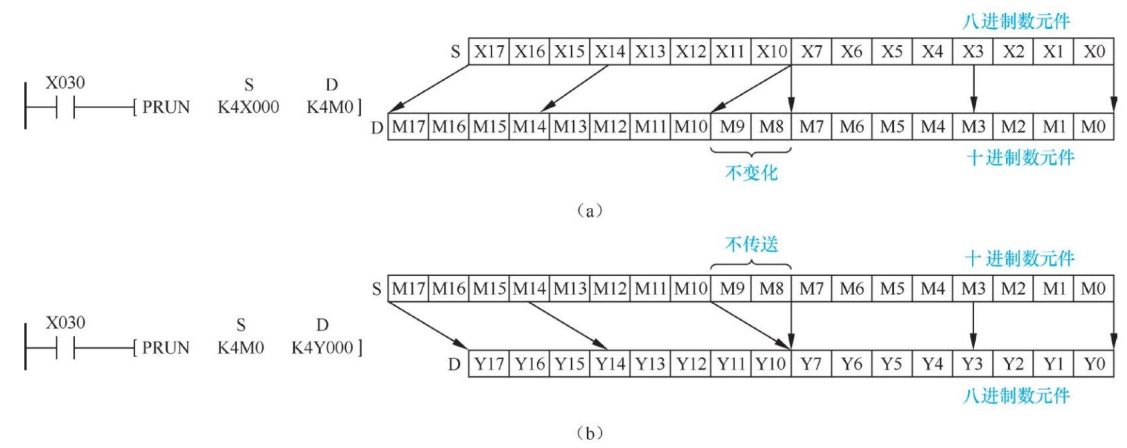


图 6-103 PRUN 指令的使用

3 十六进制数转 ASCII 码指令

(1) 关于 ASCII 码知识

ASCII 码又称美国标准信息交换码，它是一种使用 7 位或 8 位二进制数进行编码的方案，最多可以对 256 个字符（包括字母、数字、标点符号、控制字符及其他符号）进行编码。ASCII 编码表见表 6-86。计算机采用 ASCII 编码方式，当按下键盘上的 A 键时，键盘内的编码电路就将该键编码成 1000001，再送入计算机进行处理。

表 6-86 ASCII 编码表

$b_7b_6b_5$ $b_4b_3b_2b_1$	000	001	010	011	100	101	110	111
0000	nul	dle	sp	0	@	P	\	p
0001	soh	dc1	!	1	A	Q	a	q
0010	stx	dc2	"	2	B	R	b	r
0011	etx	dc3	#	3	C	S	c	s
0100	eot	dc4	\$	4	D	T	d	t
0101	enq	nak	%	5	E	U	e	u
0110	ack	svn	&	6	F	V	f	v
0111	bel	etb	'	7	G	W	g	w
1000	bs	can	(	8	H	X	h	x
1001	ht	em	)	9	I	Y	i	y
1010	lf	sub	*	:	J	Z	j	z
1011	vt	esc	+	;	K	[	k	{
1100	ff	fs	,	<	L	\	l	
1101	cr	gs	-	=	M	]	m	}
1110	so	rs	.	>	N	^	n	~
1111	si	us	/	?	O	_	o	del

(2) 十六进制数转 ASCII 码指令格式

十六进制数转 ASCII 码指令格式如表 6-87 所示。

表 6-87 十六进制数转 ASCII 码指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	n
十六进制数转 ASCII 码指令	ASCII FNC82		K、H、 KnX、KnY、KnM、 KnS、T、C、D	KnX、KnY、KnM、 KnS、T、C、D	K、H n=1 ~ 256

(3) 使用说明

ASCII 指令的使用如图 6-104 所示。在 PLC 运行时，M8000 常闭触点断开，M8161 失电，将数据存储设为 16 位模式。当常开触点 X010 闭合时，ASCII 指令执行，将[S]D100 存储的[n]4 个十六进制数转换成 ASCII 码，并保存在[D]D200 为首编号的几个连号元件中。



图 6-104 ASCII 指令的使用

当 8 位模式处理辅助继电器 M8161=OFF 时，数据存储形式是 16 位，此时[D]元件的高 8 位和低 8 位分别存放一个 ASCII 码，如图 6-105 所示，D100 中存储十六进制数 0ABC，执行 ASCII 指令后，0、A 被分别转换成 0、A 的 ASCII 码 30H、41H，并存入 D200 中；当 M8161=ON 时，数据存储形式是 8 位，此时[D]元件仅用低 8 位存放一个 ASCII 码。

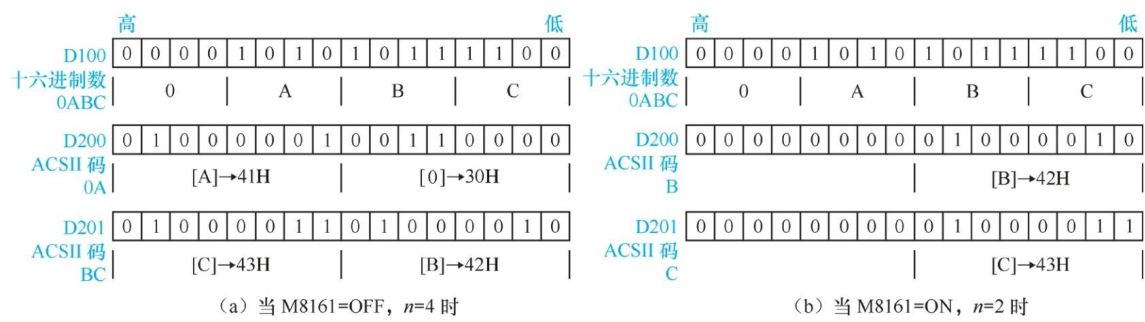


图 6-105 M8161 处于不同状态时 ASC 指令的使用

4 ASCII 码转十六进制数指令

(1) 指令格式

表 6-88 ASCII 码转十六进制数指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	n
ASCII 码转十六进制数指令	HEX FNC83		K、H、 KnX、KnY、 KnM、KnS、 T、C、D	KnX、KnY、 KnM、KnS、 T、C、D	K、H n=1 ~ 256

(2) 使用说明

HEX 指令的使用如图 6-106 所示。在 PLC 运行时, M8000 常闭触点断开, M8161 失电, 将数据存储设为 16 位模式。当常开触点 X010 闭合时, HEX 指令执行, 将[S]D200、D201 存储的[n]4 个 ASCII 码转换成十六进制数, 并保存在[D]D100 中。



图 6-106 HEX 指令的使用

如图 6-107 所示, 当 M8161=OFF 时, 数据存储形式是 16 位, [S]元件的高 8 位和低 8 位分别存放一个 ASCII 码; 当 M8161=ON 时, 数据存储形式是 8 位, 此时[S]元件仅低 8 位有效, 即只用低 8 位存放一个 ASCII 码。

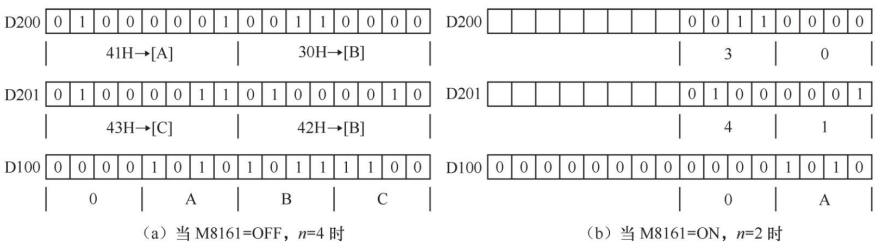


图 6-107 M8161 处于不同状态时 HEX 指令的使用

5 校验码指令

(1) 指令格式

校验码指令格式如表 6-89 所示。

表 6-89 校验码指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S	D	n
校验码指令	CCD FNC84		KnX 、 KnY 、 KnM 、 KnS 、 T 、 C 、 D	KnY 、 KnM 、 KnS 、 T 、 C 、 D	K 、 H $n=1 \sim 256$

(2) 使用说明

CCD 指令的使用如图 6-108 所示。在 PLC 运行时，M8000 常闭触点断开，M8161 失电，将数据存储设为 16 位模式。当常开触点 X010 闭合时，CCD 指令执行，将[S]D100 为首编号元件的[n]10 点数据（8 位为 1 点）进行求总和，并生成校验码，再将数据总和及校验码分别保存在[D]、[D]+1（D0、D1）中。



图 6-108 CCD 指令的使用

数据求总和及校验码生成如图 6-109 所示。在求总和时，将 D100～D104 中的 10 点数据相加，得到总和为 1091（二进制为 10001000011）。生成校验码的方法是：逐位计算 10 点数据中每位 1 的总数，每位 1 的总数为奇数时，生成的校验码对应位为 1；每位 1 的总数为偶数时，生成的校验码对应位为 0，图 6-109 中 D100～D104 中的 10 点数据的最低位 1 的总数为 3，是奇数，故生成校验码对应位为 1，10 点数据生成的校验码为 1000101。数据总和存入 D0 中，校验码存入 D1 中。

S	数据内容
D100 低	K100 = 0 1 1 0 0 1 0 0
D100 高	K111 = 0 1 1 0 1 1 1 1
D101 低	K100 = 0 1 1 0 0 1 0 0
D101 高	K98 = 0 1 1 0 0 0 1 0
D102 低	K123 = 0 1 1 1 1 0 1 1
D102 高	K66 = 0 1 0 0 0 0 1 0
D103 低	K100 = 0 1 1 0 0 1 0 0
D103 高	K95 = 0 1 0 1 1 1 1 1
D104 低	K210 = 1 1 0 1 0 0 1 0
D104 高	K88 = 0 1 0 1 1 0 0 0
合计	K1091
校验	1 0 0 0 0 1 0 1

1 的个数是奇数，校验为 1
1 的个数是偶数，校验为 0

D0
0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 1
← 1091

D1
0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1
← 校验

图 6-109 数据求总和及校验码生成

校验码指令常用于检验通信中数据是否发生错误。

6 电位器模拟量读出指令

(1) 指令格式

电位器模拟量读出指令格式如表 6-90 所示。

表 6-90 电位器模拟量读出指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
电位器模拟量读出指令	VRRD FNC85		K、H 变量号 0 ~ 7	KnY、KnM、KnS、 T、C、D、V、Z

(2) 使用说明

VRRD 指令的功能是将模拟量调整器[S]号电位器的模拟值转换成二进制数 0~255，并存入[D]元件中。模拟量调整器是一种功能扩展板，FX_{1N}-8AV-BD 和 FX_{2N}8AV-BD 是两种常见的调整器，安装在 PLC 的主单元上，调整器上有 8 个电位器，编号为 0~7，当电位器阻值由 0 调到最大时，相应转换成的二进制数由 0 变到 255。

VRRD 指令的使用如图 6-110 所示。当常开触点 X000 闭合时，VRRD 指令执行，将模拟量调整器的[S]0 号电位器的模拟值转换成二进制数，再保存在[D]D0 中，当常开触点 X001 闭合时，定时器 T0 开始以 D0 中的数作为计时值进行计时，这样就可以通过调节电位器来改变定时时间，如果定时时间大于 255，可用乘法指令 MUL 将[D]与某常数相乘而得到更大的定时时间。

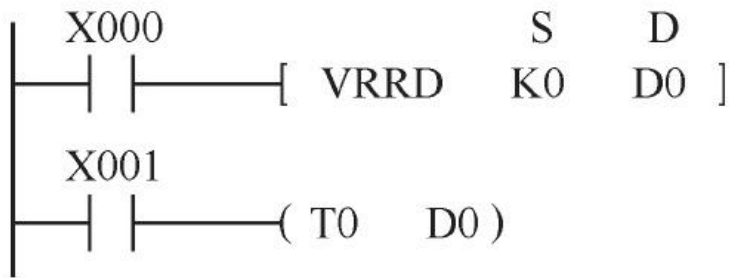


图 6-110 VRRD 指令的使用

7 电位器模拟量开关设定指令

(1) 指令格式

电位器模拟量开关设定指令格式如表 6-91 所示。

表 6-91 电位器模拟量开关设定指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
电位器模拟量 开关设定指令	VRSC FNC86		K、H 变量号 0 ~ 7	KnY、KnM、KnS、 T、C、D、V、Z

(2) 使用说明

VRSC 指令的功能与 VRRD 指令类似，但 VRSC 指令是将模拟量调整器[S]号电位器均分成 0~10 部分（相当于 0~10 挡），并转换成二进制数 0~10，再存入[D]元件中。电位器在旋转时是通过四舍五入化成整数值 0~10。

VRSC 指令的使用如图 6-111 所示。当常开触点 X000 闭合时，VRSC 指令执行，将模拟量调整器的[S]1 号电位器的模拟值转换成二进制数 0~10，再保存在[D]D1 中。



图 6-111 VRSC 指令的使用

利用 VRSC 指令能将电位器分成 0~10 共 11 挡，可实现一个电位器进行 11 种控制切换，程序如图 6-112 所示。当常开触点 X000 闭合时，VRSC 指令执行，将 1 号电位器的模拟量值转换成二进制数（0~10），并存入 D1 中；当常开触点 X001 闭合时，DECO（解码）指令执行，对 D1 的低 4 位数进行解码，4 位数解码有 16 种结果，解码

结果存入 M0~M15 中，设电位器处于 1 挡，D1 的低 4 位数则为 0001，因 (0001)2=1，解码结果使 M1 为 1 (M0~M15 其他的位均为 0)，M1 常开触点闭合，执行设定的程序。

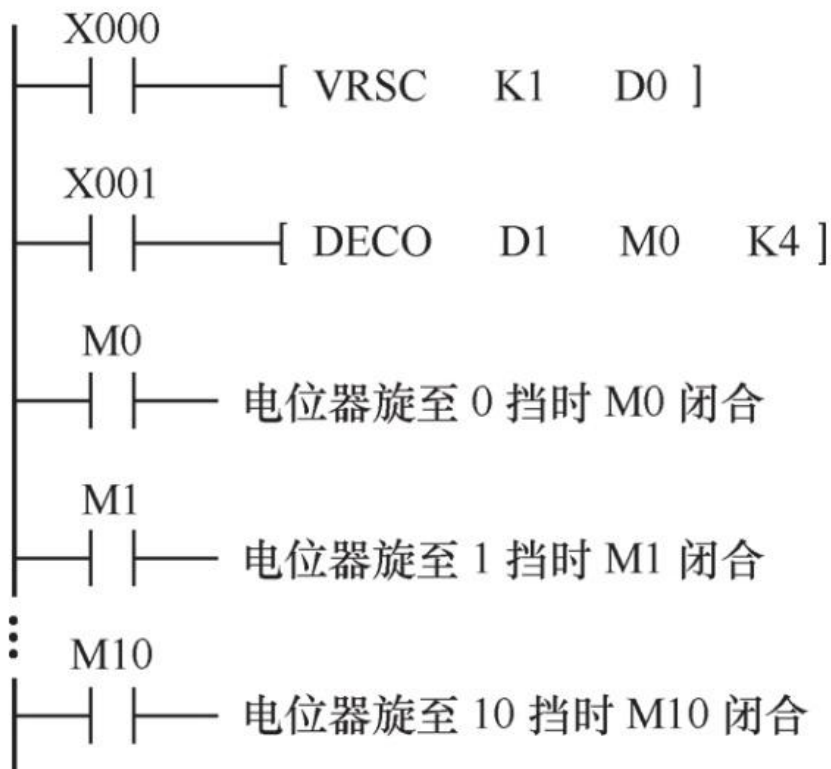


图 6-112 利用 VRSC 指令将电位器分成 11 挡的程序

8 PID 运算指令

(1) 关于 PID 控制

PID 控制又称比例微积分控制，是一种闭环控制。以图 6-113 所示的恒压供水系统为例来说明 PID 控制原理。

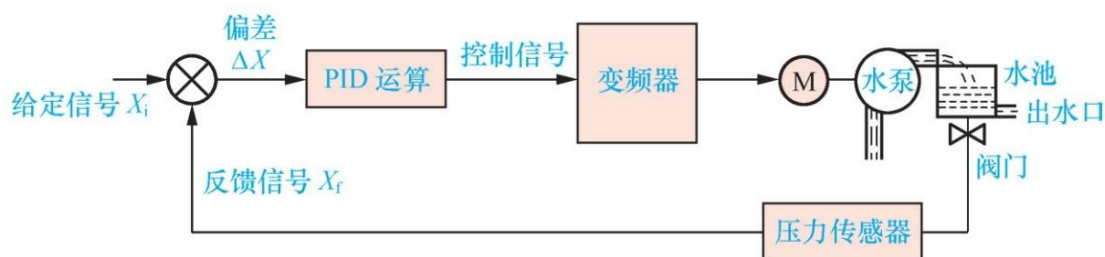


图 6-113 恒压供水系统

电动机驱动水泵将水抽入水池，水池中的水除了经出水口提供用水外，还经阀门送到压力传感器，传感器将水压大小转换成相应的电信号 X_f ， X_f 反馈到比较器与给定信号 X_i 进行比较，得到偏差信号 ΔX ($\Delta X = X_i - X_f$)。

若 $\Delta X > 0$ ，表明水压小于给定值，偏差信号经 PID 运算得到控制信号，控制变频器，使之输出频率上升，电动机转速加快，水泵抽水量增多，水压增大。

若 $\Delta X < 0$ ，表明水压大于给定值，偏差信号经 PID 运算得到控制信号，控制变频器，使之输出频率下降，电动机转速变慢，水泵抽水量减少，水压下降。

若 $\Delta X = 0$ ，表明水压等于给定值，偏差信号经 PID 运算得到控制信号，控制变频器，使之输出频率不变，电动机转速不变，水泵抽水量不变，水压不变。

由于控制回路的滞后性，会使水压值与给定值有偏差。例如，当用水量增多水压下降时，电路需要对有关信号进行处理，再控制电动机转速变快，提高水泵抽水量。从压力传感器检测到水压下降，使控制电动机转速加快，提高水泵抽水量，恢复水压需要一定时间。通过提

高电动机转速恢复水压后，系统又要将电动机转速调回正常值，这也要一定的时间，在这段回调时间内水泵抽水量会偏多，导致水压又增大，又需进行反调。这样的结果是水池水压会在给定值上下波动（振荡），即水压不稳定。

采用了 PID 运算可以有效减小控制环路滞后和过调问题（无法彻底消除）。PID 运算包括 P 处理、I 处理和 D 处理。P（比例）处理是将偏差信号 ΔX 按比例放大，提高控制的灵敏度；I（积分）处理是对偏差信号进行积分处理，缓解 P 处理比例放大量过大引起的超调和振荡；D（微分）处理是对偏差信号进行微分处理，以提高控制的迅速性。

(2) PID 运算指令格式

PID 运算指令格式如表 6-92 所示。

表 6-92 PID 运算指令格式

指令名称	助记符与功能号	指令形式	操作数			
			S1	S2	S3	D
PID 运算指令	PID FNC88		D	D	D	D

(3) 使用说明

① 指令的使用形式

PID 指令的使用形式如图 6-114 所示。当常开触点 X000 闭合时，PID 指令执行，将[S1]D0 设定值与[S2]D1 测定值之差按[S3]D100 ~ D124 设定的参数表进行 PID 运算，运算结果存入[D]D150 中。

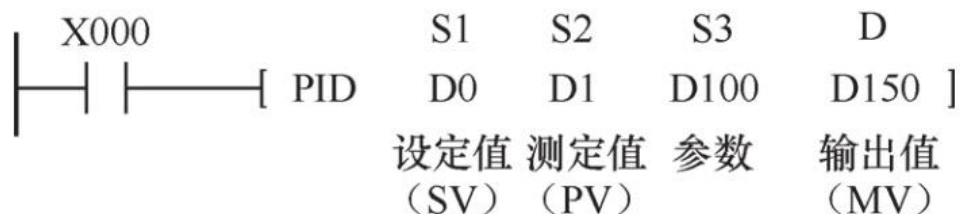


图 6-114 PID 指令的使用形式

② PID 参数设置

PID 运算的依据是[S3]指定首地址的 25 个连号数据寄存器保存的参数表。参数表一部分内容必须在执行 PID 指令前由用户用指令写入（如用 MOV 指令），一部分留作内部运算使用，还有一部分用来存入运算结果。[S3] ~ [S3]+24 保存的参数表内容见表 6-93。

表 6-93 [S3]~[S3]+24 保存的参数表内容

元件	功能		
[S3]	采样时间 (T_s) 1 ~ 32767 (ms) （但比运算周期短的时间数值无法执行）		
[S3]+1	动作方向 (ACT)	bit0 0: 正动作（如空调控制）	1: 逆动作（如加热炉控制）
		bit1 0: 输入变化量报警无	1: 输入变化量报警有效
		bit2 0: 输出变化量报警无	1: 输出变化量报警有效
		bit3 不可使用	
		bit4: 自动调谐不动作	1: 执行自动调谐
		bit5: 输出值上下限设定无	1: 输出值上下限设定有效
		bit6 ~ bit15 不可使用	
		另外, 请不要使 bit5 和 bit2 同时处于 ON	
[S3]+2	输入滤波常熟 (α)	0 ~ 99[%]	0 时没有输入滤波
[S3]+3	比例增益 (K_P)	1 ~ 32767[%]	
[S3]+4	积分时间 (T_i)	0 ~ 32767($\times 100$ ms)	0 时作为 ∞ 处理（无积分）
[S3]+5	微分增益 (K_D)	0 ~ 100[%]	0 时无积分增益
[S3]+6	微分时间 (T_D)	0 ~ 32767($\times 10$ ms)	0 时无微分处理
[S3]+7 ~ [S3]+19	PID 运算的内部处理占用		
[S3]+20	输入变化量（增侧）报警设定值	0 ~ 32767（[S3]+1<ACT> 的 bit1=1 时有效）	
[S3]+21	输入变化量（减侧）报警设定值	0 ~ 32767（[S3]+1<ACT> 的 bit1=1 时有效）	
[S3]+22	输出变化量（增侧）报警设定值	0 ~ 32767（[S3]+1<ACT> 的 bit2=1, bit5=0 时有效）	
	另外, 输出上限设定值	-32768 ~ 32767（[S3]+1<ACT> 的 bit2=0, bit5=1 时有效）	
[S3]+23	输出变化量（减侧）报警设定值	0 ~ 32767（[S3]+1<ACT> 的 bit2=1, bit5=0 时有效）	
	另外, 输出下限设定值	-32768 ~ 32767（[S3]+1<ACT> 的 bit2=0, bit5=1 时有效）	
[S3]+24	报警 输出	bit0 输入变化量（增侧）溢出	（[S3]+1<ACT> 的 bit1=1 或 bit2=1 时有效）
		bit1 输入变化量（减侧）溢出	
		bit2 输出变化量（增侧）溢出	
		bit3 输出变化量（减侧）溢出	

③ PID 控制应用举例

在恒压供水 PID 控制系统中, 压力传感器将水压大小转换成电信号。该信号是模拟量, PLC 无法直接接收, 需要用电路将模拟量转换成数字量, 再将数字量作为测定值送入 PLC, 将它与设定值之差进行 PID 运算, 运算得到控制值, 控制值是数字量, 变频器无法直接接收, 需要用电路将数字量控制值转换成模拟量信号, 去控制变频器, 使变频器根据控制信号来调制泵电机的转速, 以实现恒压供水。

三菱 FX_{2N} 型 PLC 有专门配套的模拟量输入/输出功能模块 FX_{0N}-3A, 在使用时将它用专用电缆与 PLC 连接好, 如图 6-115 所示, 再将模拟输入端接压力传感器, 模拟量输出端接变频器。在工作时, 压力传感器送来的反映压力大小的电信号进入 FX_{0N}-3A 模块转换成数字量, 再送入 PLC 进行 PID 运算, 运算得到的控制值送入 FX_{0N}-3A 转换成模拟量控制信号, 该信号去调节变频器的频率, 从而调节泵电机的转速。

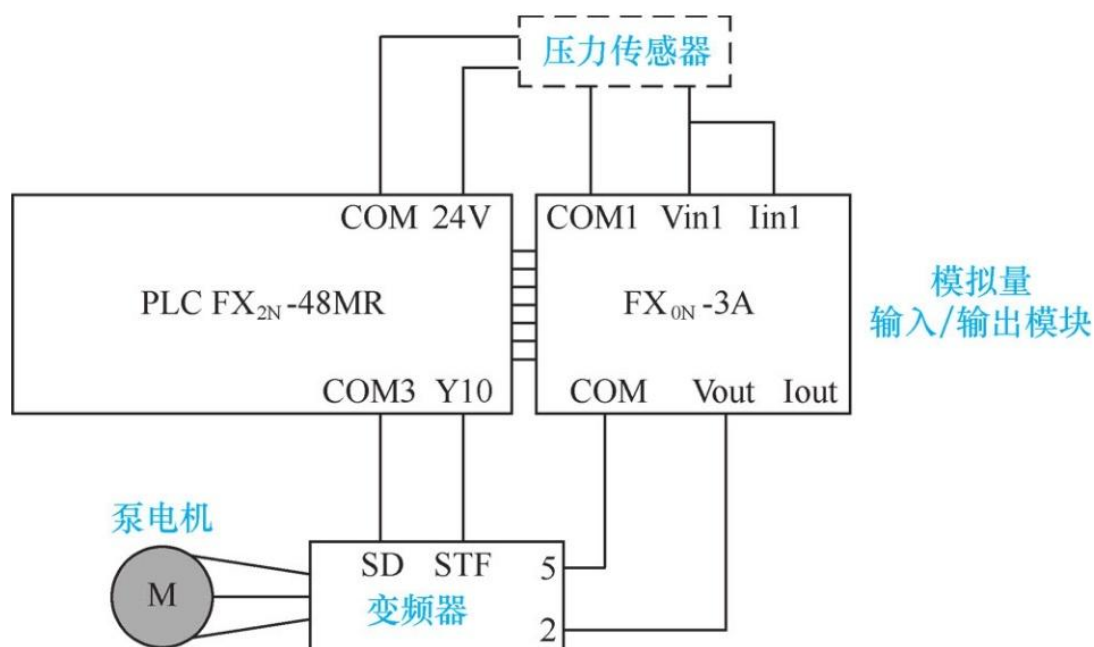


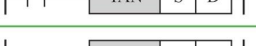
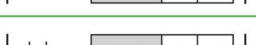
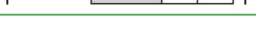
图 6-115 PID 控制恒压供水的硬件连接

浮点数即实数，是指带小数点的数。浮点数运算指令包括浮点数比较、变换、四则运算、开平方和三角函数等指令。这些指令的使用方法与二进制数运算指令类似，但浮点运算都是 32 位。大多数情况下，很少用到浮点运算指令。浮点运算指令格式见表 6-94。

表 6-94 浮点运算指令格式

指令名称	助记符与功能号	指令形式
二进制浮点数比较	ECMP FNC110	
二进制浮点数区间比较	EZCP FNC111	
二进制浮点数数据传送	EMOV FNC112	
二进制浮点数→字符串	ESTR FNC116	
字符串→二进制浮点数	EVAL FNC117	
二进制浮点数→十进制浮点数	EBCD FNC118	
十进制浮点数→二进制浮点数	EBIN FNC119	
二进制浮点数加法运算	EADD FNC120	
二进制浮点数减法运算	ESUB FNC121	

续表

指令名称	助记符与功能号	指令形式
二进制浮点数乘法运算	EMUL FNC122	
二进制浮点数除法运算	EDIV FNC123	
二进制浮点数指数运算	EXP FNC124	
二进制浮点数自然对数运算	LOGE FNC125	
二进制浮点数常用对数运算	LOG10 FNC126	
二进制浮点数开方运算	ESQR FNC127	
二进制浮点数字号翻转运算	ENEG FNC128	
二进制浮点数 → BIN 整数	INT FNC129	
二进制浮点数 SIN 运算	SIN FNC130	
二进制浮点数 COS 运算	COS FNC131	
二进制浮点数 TAN 运算	TAN FNC132	
二进制浮点数 $\sin^{-1}$ 运算	ASIN FNC133	
二进制浮点数 $\cos^{-1}$ 运算	ACOS FNC134	
二进制浮点数 $\tan^{-1}$ 运算	ATAN FNC135	
二进制浮点数角度 → 弧度	RAD FNC136	
二进制浮点数弧度 → 角度	DEG FNC137	

1 高低位变换指令格式

(1) 指令格式

高低位变换指令格式如表 6-95 所示。

表 6-95 高低位变换指令格式

指令名称	助记符与功能号	指令形式	操作数
			S
高低位变换指令格式	SWAP FNC147		KnY、KnM、KnS、 T、C、D、V、Z

(2) 使用说明

高低位变换指令的使用如图 6-116 所示, 图 6-116 (a) 中的 SWAPP 为 16 位指令, 当常开触点 X000 闭合时, SWAPP 指令执行, D10 中的高 8 位和低 8 位数据互换; 图 6-116 (b) 的 DSWAPP 为 32 位指令, 当常开触点 X001 闭合时, DSWAPP 指令执行, D10 中的高 8 位和低 8 位数据互换, D11 中的高 8 位和低 8 位数据互换。

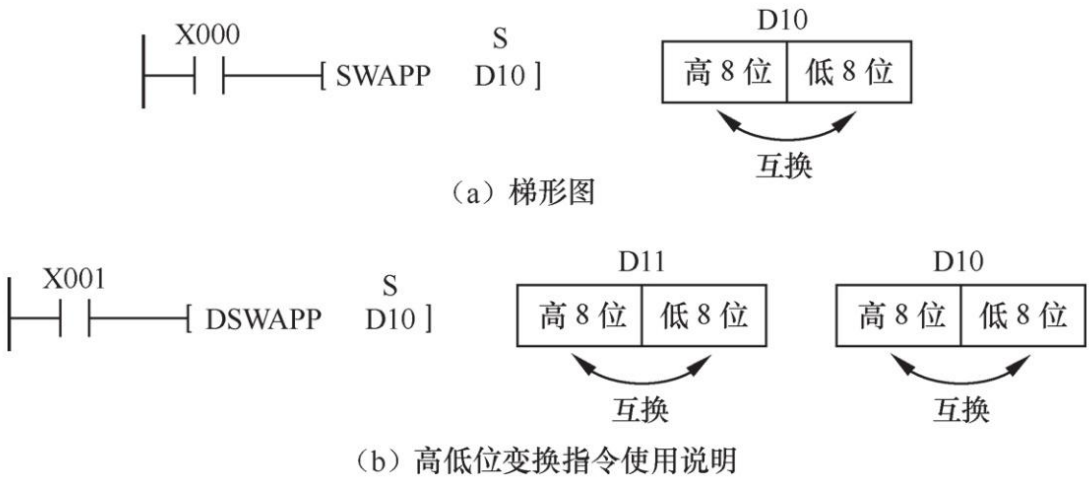


图 6-116 高低位变换指令的使用

1 时钟数据比较指令

(1) 指令格式

时钟数据比较指令格式如表 6-96 所示。

表 6-96 时钟数据比较指令格式

指令名称	助记符与功能号	指令形式	操作数				
			S1	S2	S3	S	D
时钟数据比较指令	TCMP FNC160		K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z			T、C、D (占3个连续元件)	Y、M、S (占3个连续元件)

(2) 使用说明

TCMP 指令的使用如图 6-117 所示。[S1]为指定基准时间的小时值（0~23），[S2]为指定基准时间的分钟值（0~59），[S3]为指定基准时间的秒值（0~59），[S]指定待比较的时间值，其中[S]、[S]+1、[S]+2 分别为待比较的小时、分、秒值，[D]为比较输出元件，其中[D]、[D]+1、[D]+2 分别为 >、=、< 时的输出元件。

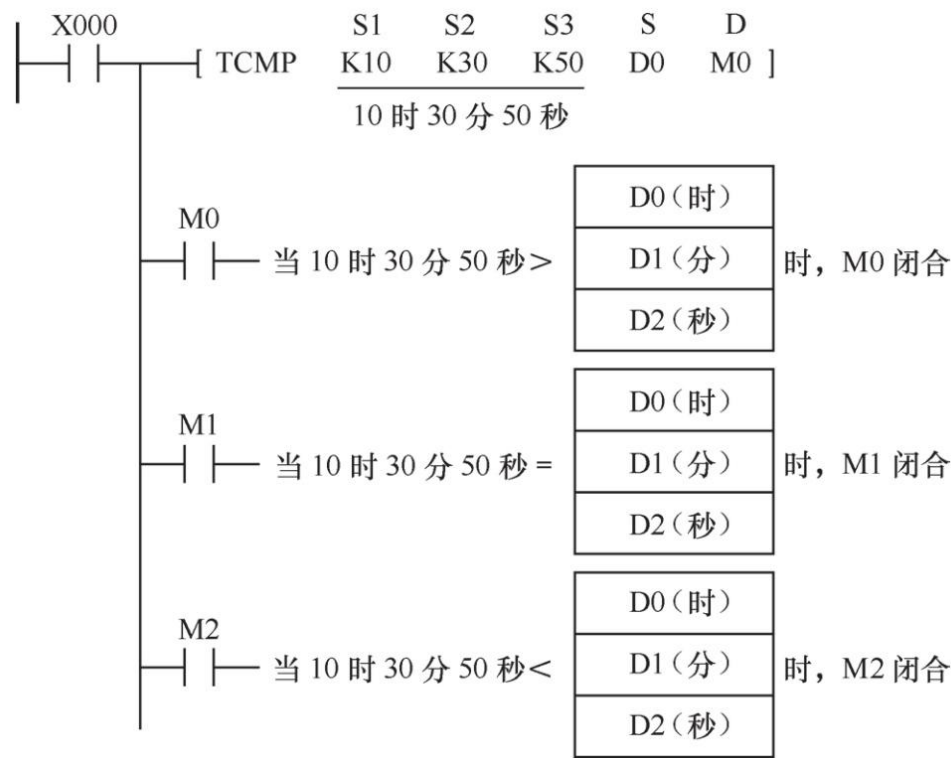


图 6-117 TCMP 指令的使用

当常开触点 X000 闭合时, TCMP 指令执行, 将时间值「10 时 30 分 50 秒」与 D0、D1、D2 中存储的小时、分、秒值进行比较, 根据比较结果驱动 M0~M2, 具体如下。

若「10 时 30 分 50 秒」大于「D0、D1、D2 存储的小时、分、秒值」, M0 被驱动, M0 常开触点闭合。

若「10 时 30 分 50 秒」等于「D0、D1、D2 存储的小时、分、秒值」, M1 被驱动, M1 常开触点闭合。

若「10 时 30 分 50 秒」小于「D0、D1、D2 存储的小时、分、秒值」, M2 被驱动, M2 常开触点闭合。

当常开触点 X000=OFF 时, TCMP 指令停止执行, 但 M0~M2 仍保持 X000 为 OFF 前时的状态。

2 时钟数据区间比较指令

(1) 指令格式

时钟数据区间比较指令格式如表 6-97 所示。

表 6-97 时钟数据区间比较指令格式

指令名称	助记符与 功能号	指令形式	操作数			
			S1	S2	S	D
时钟数据区间 比较指令	TZCP FNC161		T、C、D [S1] ≤ [S2] (3 个连续元件)		T、C、D	Y、M、S (占 3 个连续元件)

(2) 使用说明

TZCP 指令的使用如图 6-118 所示。[S1]指定第一基准时间值（小时、分、秒值），[S2]指定第二基准时间值（小时、分、秒值），[S]指定待比较的时间值，[D]为比较输出元件，[S1]、[S2]、[S]、[D]都需占用 3 个连号元件。

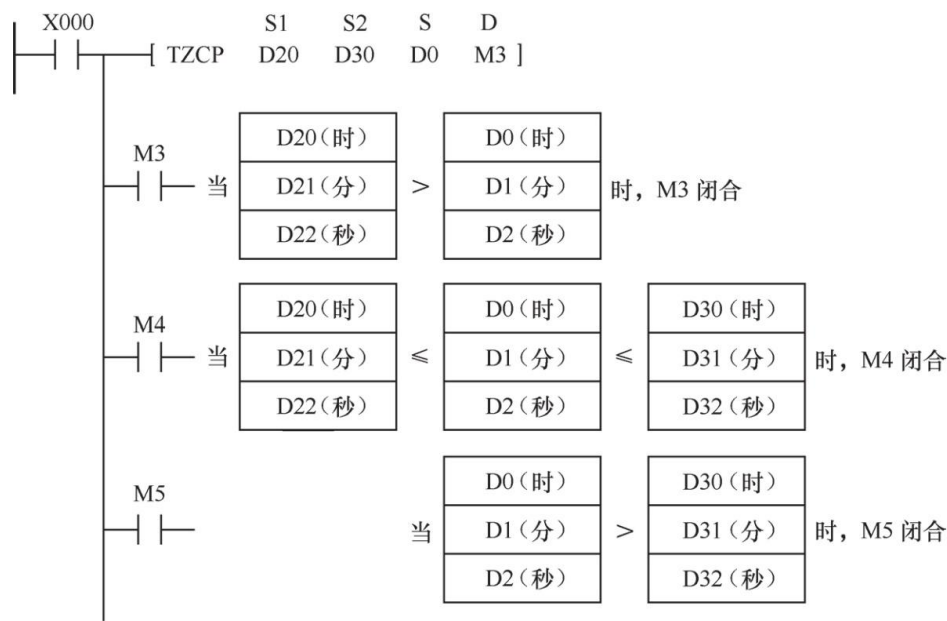


图 6-118 TZCP 指令的使用

当常开触点 X000 闭合时, TZCP 指令执行, 将「D20、D21、D22」「D30、D31、D32」中的时间值与「D0、D1、D2」中的时间值进行比较, 根据比较结果驱动 M3~M5, 具体如下。

若「D0、D1、D2」中的时间值小于「D20、D21、D22」中的时间值, M3 被驱动, M3 常开触点闭合。

若「D0、D1、D2」中的时间值处于「D20、D21、D22」和「D30、D31、D32」时间值之间, M4 被驱动, M4 常开触点闭合。

若「D0、D1、D2」中的时间值大于「D30、D31、D32」中的时间值, M5 被驱动, M5 常开触点闭合。

当常开触点 X000=OFF 时, TZCP 指令停止执行, 但 M3~M5 仍保持 X000 为 OFF 前时的状态。

3 时钟数据加法指令

(1) 指令格式

时钟数据加法指令格式如表 6-98 所示。

表 6-98 时钟数据加法指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
时钟数据加法指令	TADD FNC162		T、C、D		T、C、D

(2) 使用说明

TADD 指令的使用如图 6-119 所示。[S1]指定第一时间值（小时、分、秒值），[S2]指定第二时间值（小时、分、秒值），[D]保存[S1]+[S2]的和值，[S1]、[S2]、[D]都需占用 3 个连号元件。

当常开触点 X000 闭合时，TADD 指令执行，将「D10、D11、D12」中的时间值与「D20、D21、D22」中的时间值相加，结果保存在「D30、D31、D32」中。

如果运算结果超过 24 小时，进位标志位会置 ON，将加法结果减去 24 小时再保存在[D]中，如图 6-119 (b) 所示。如果运算结果为 0，零标志位会置 ON。

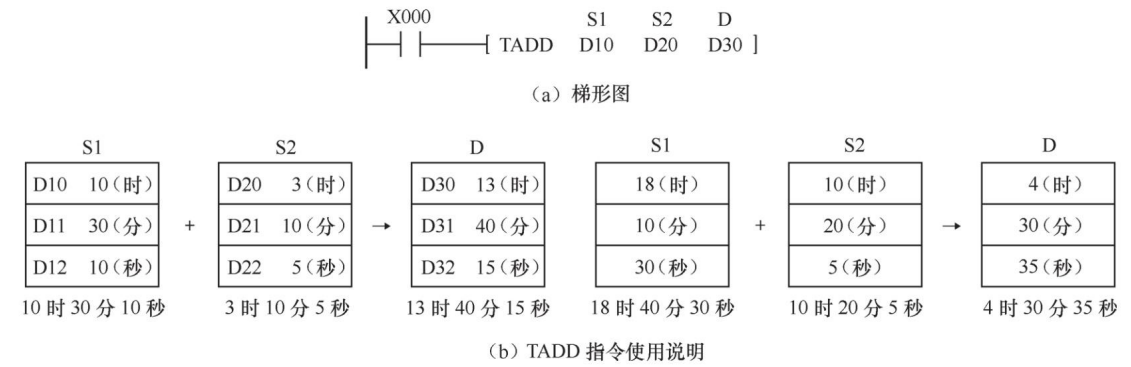


图 6-119 TADD 指令的使用

4 时钟数据减法指令

(1) 指令格式

时钟数据减法指令格式如表 6-99 所示。

表 6-99 时钟数据减法指令格式

指令名称	助记符与功能号	指令形式	操作数		
			S1	S2	D
时钟数据减法指令	TSUB FNC163		T、C、D		T、C、D

(2) 使用说明

TSUB 指令的使用如图 6-120 所示。[S1]指定第一时间值（小时、分、秒值），[S2]指定第二时间值（小时、分、秒值），[D]保存[S1]-[S2]的差值，[S1]、[S2]、[D]都需占用 3 个连号元件。

当常开触点 X000 闭合时，执行 TSUB 指令，将「D10、D11、D12」中的时间值与「D20、D21、D22」中的时间值相减，结果保存在「D30、D31、D32」中。

如果运算结果小于 0 小时，借位标志位会置 ON，将减法结果加 24 小时再保存在[D]中，如图 6-120 所示。

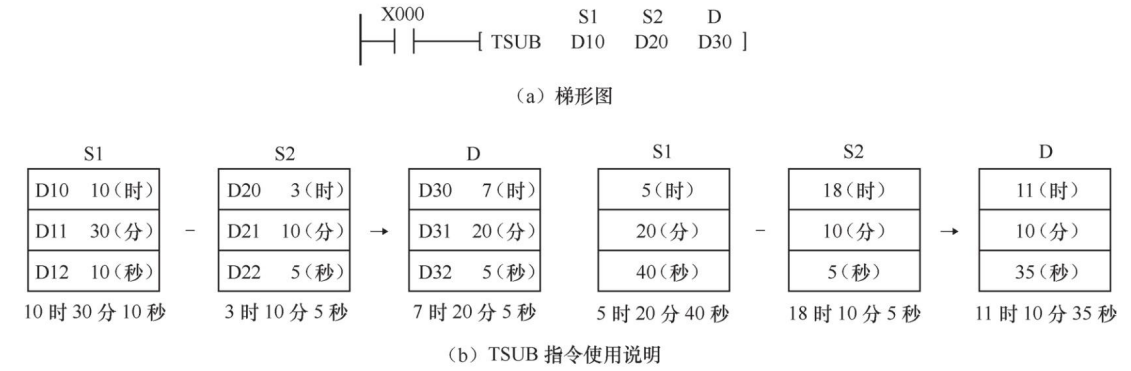


图 6-120 TSUB 指令的使用

5 时钟数据读出指令

(1) 指令格式

时钟数据读出指令格式如表 6-100 所示。

表 6-100 时钟数据读出指令格式

指令名称	助记符与功能号	指令形式	操作数
			D
时钟数据读出指令	TRD FNC166		T、C、D (7 个连号元件)

(2) 使用说明

TRD 指令的使用如图 6-121 所示。TRD 指令的功能是将 PLC 当前时间（年、月、日、时、分、秒、星期）读入[D]D0 为首编号的 7 个连号元件 D0~D6 中。PLC 当前时间保存在实时时钟用的特殊数据寄存器 D8013~D8019 中，这些寄存器中的数据会随时间变化而变化。D0~D6 和 D8013~D8019 的内容及对应关系如图 6-121（b）所示。



图 6-121 TRD 指令的使用

当常开触点 X000 闭合时，TRD 指令执行，将「D8018~D8013、D8019」中的时间值保存到（读入）D0~D6 中，如将 D8018 中的数据作为年值存入 D0 中，将 D8019 中的数据作为星期值存入 D6 中。

6 时钟数据写入指令

(1) 指令格式

时钟数据写入指令格式如表 6-101 所示。

表 6-101 时钟数据写入指令格式

指令名称	助记符与功能号	指令形式	操作数
			S
时钟数据写入指令	TWR FNC167		T、C、D (7 个连号元件)

(2) 使用说明

TWR 指令的使用如图 6-122 所示。TWR 指令的功能是将[S]D10 为首编号的 7 个连号元件 D10~D16 中的时间值（年、月、日、时、分、秒、星期）写入特殊数据寄存器 D8013~D8019 中。D10~D16 和 D8013~D8019 的内容及对应关系如图 6-121 (b) 所示。



图 6-122 TWR 指令的使用

当常开触点 X001 闭合时，TWR 指令执行，将「D10~D16」中的时间值写入 D8018~D8013、D8019 中，如将 D10 中的数据作为年值写入 D8018 中，将 D16 中的数据作为星期值写入 D8019 中。

(3) 修改 PLC 的实时时钟

PLC 在出厂时已经设定了实时时钟，以后实时时钟会自动运行，如果实时时钟运行不准确，可以采用程序修改。图 6-123 所示为修改 PLC 实时时钟的梯形图程序，利用它可以将实时时钟设为 05 年 4 月 25 日 3 时 20 分 30 秒星期二。

在编程时，先用 MOV 指令将要设定的年、月、日、时、分、秒、星期值分别传送给 D0 ~ D6，然后用 TWR 指令将 D0 ~ D6 中的时间值写入 D8018 ~ D8013、D8019 中。在进行时钟设置时，设置的时间应较实际时间晚几分钟，当实际时间到达设定时间后让 X000 触点闭合，程序就将设置的时间写入 PLC 的实时时钟数据寄存器中，闭合触点 X001，M8017 置 ON，可对时钟进行 $\pm 30s$ 修正。

PLC 实时时钟的年值默认为两位（如 05 年），如果要改成 4 位（2005 年），可给图 6-123 程序追加图 6-124 所示的程序，在第二个扫描周期开始年值就为 4 位。

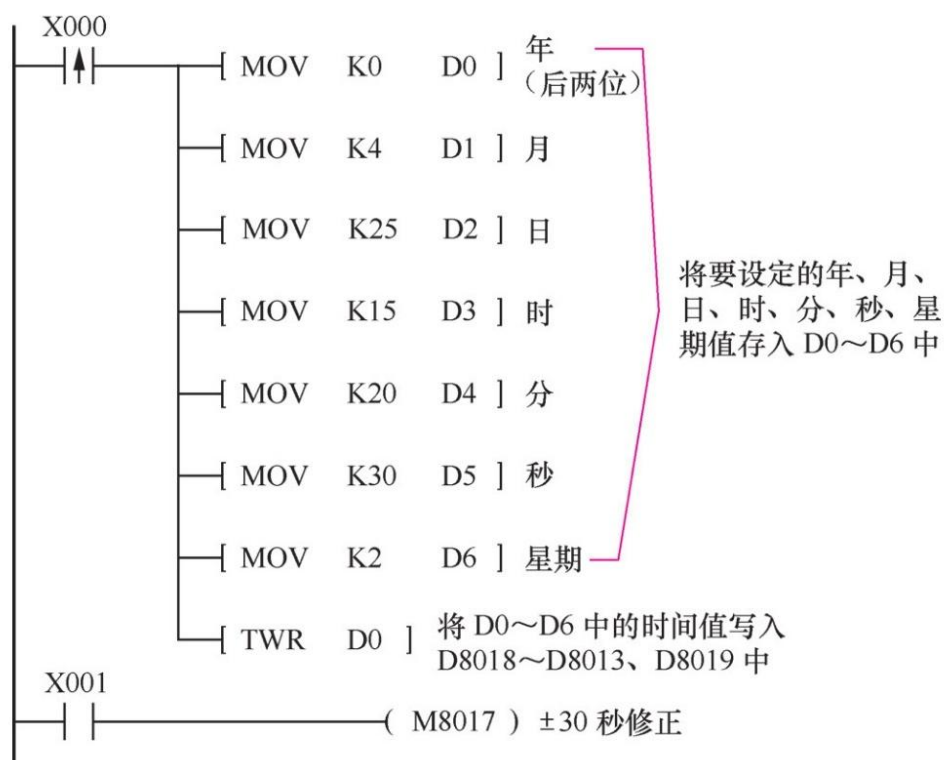


图 6-123 修改 PLC 实时时钟的梯形图程序



图 6-124 将年值改为 4 位需增加的梯形图程序

1 有关格雷码的知识

两个相邻代码之间仅有一位数码不同的代码称为格雷码。十进制数、二进制与格雷码的对应关系如表 6-102 所示。

表 6-102 十进制数、二进制数与格雷码的对应关系

十进制数	二进制数	格雷码	十进制数	二进制数	格雷码
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

从表 6-102 中可以看出，相邻的两个格雷码之间仅有一位数码不同，如 5 的格雷码是 0111，它与 4 的格雷码 0110 仅最后一位不同，与 6 的格雷码 0101 仅倒数第二位不同。二进制数在递增或递减时，往往多位发生变化，3 的二进制数 0011 与 4 的二进制数 0100 同时有三位发生变化，这样在数字电路处理中很容易出错，而格雷码在递增或递减时，仅有一位发生变化，这样不容易出错，所以格雷码常用于高分辨率的系统中。

2 二进制码（BIN 码）转格雷码指令

(1) 指令格式

二进制码转格雷码指令格式如表 6-103 所示。

表 6-103 二进制码转格雷码指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
二进制码转格雷码指令	GRY FNC170		K、H、 KnX、KnY、KnM、KnS、 T、C、D	KnY、KnM、KnS、 T、C、D

(2) 使用说明

GRY 指令的使用如图 6-125 所示。GRY 指令的功能是将[S]指定的二进制码转换成格雷码，并存入[D]指定的元件中。当常开触点 X000 闭合时，执行 GRY 指令，将「1234」的二进制码转换成格雷码，并存入 Y23~Y20、Y17~Y10 中。

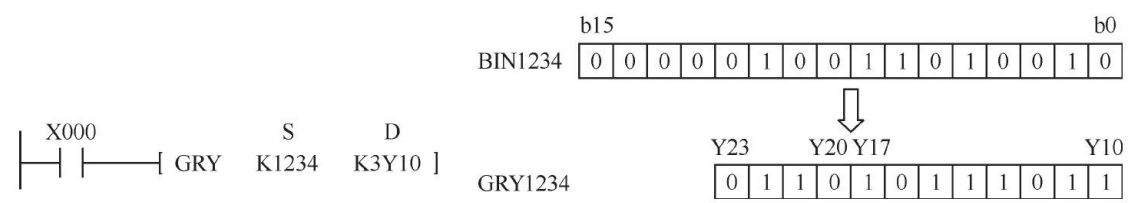


图 6-125 GRY 指令的使用

3 格雷码转二进制码指令

(1) 指令格式

格雷码转二进制码指令格式如表 6-104 所示。

表 6-104 格雷码转二进制码指令格式

指令名称	助记符与功能号	指令形式	操作数	
			S	D
格雷码转二进制码指令	GBIN FNC171		K、H、 KnX、KnY、KnM、KnS、 T、C、D	KnY、KnM、KnS、 T、C、D

(2) 使用说明

GBIN 指令的使用如图 6-126 所示。GBIN 指令的功能是将[S]指定的格雷码转换成二进制码，并存入[D]指定的元件中。当常开触点 X020 闭合时，GBIN 指令执行，将 X13 ~ X10、X7 ~ X0 中的格雷码转换成二进制码，并存入 D10 中。

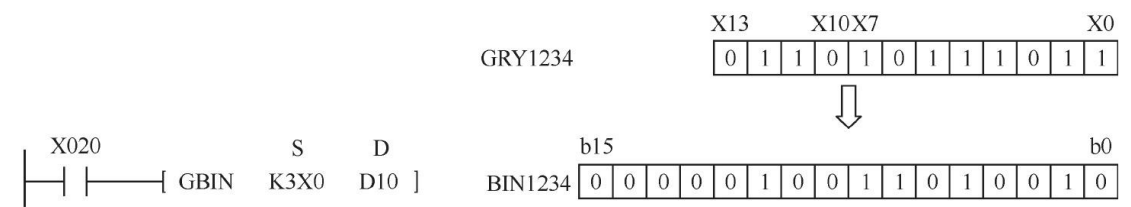


图 6-126 GBIN 指令的使用

1 触点比较 LD*指令

LD*指令是连接左母线的触点比较指令，其功能是将[S1]、[S2]两个源操作数进行比较，若结果满足要求则执行驱动。LD*为 16 位指令，LDD*为 32 位指令。

(1) 指令格式

触点比较 LD*指令格式如表 6-105 所示。

表 6-105 触点比较 LD*指令格式

指令助记符与功能号	指令形式	导通条件	操作数	
			S1	S2
LD=FNC224		S1=S2 时导通	K、H、 KnX、KnY、 KnM、KnS、 T、C、D、V、Z	K、H、 KnX、KnY、 KnM、KnS、 T、C、D、V、Z
LD > FNC225		S1>S2 时导通		
LD < FNC226		S1<S2 时导通		
LD <> FNC228		S1 ≠ S2 时导通		
LD ≤ FNC229		S1 ≤ S2 时导通		
LD ≥ FNC230		S1 ≥ S2 时导通		

(2) 使用说明

LD*指令的使用如图 6-127 所示。当计数器 C10 的计数值等于 200 时,驱动 Y010;当 D200 中的数据大于-30 并且常开触点 X001 闭合时,将 Y011 置位;当计数器 C200 的计数值小于 678493 时,或 M3 触点闭合时,驱动 M50。

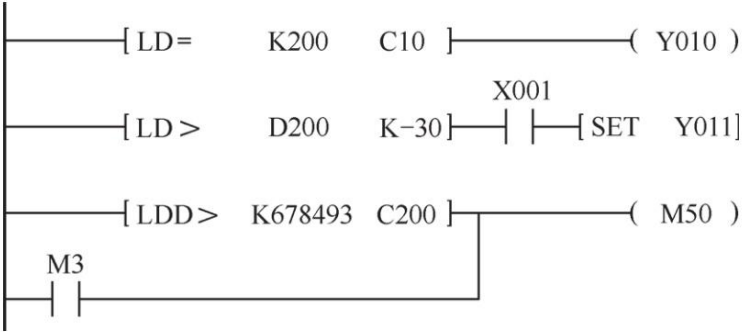


图 6-127 LD*指令的使用

2 触点比较 AND*指令

AND*指令是串联型触点比较指令，其功能是将[S1]、[S2]两个源操作数进行比较，若结果满足要求则执行驱动。AND*为 16 位指令，ANDD*为 32 位指令。

(1) 指令格式

触点比较 AND*指令格式如表 6-106 所示。

表 6-106 触点比较 AND*指令格式

指令助记符与功能号	指令形式	导通条件	操作数	
			S1	S2
AND=FNC232		S1=S2 时导通	K、H、 KnX、KnY、 KnM、KnS、 T、C、D、V、Z	K、H、 KnX、KnY、 KnM、KnS、 T、C、D、V、Z
AND > FNC233		S1>S2 时导通		
AND < FNC234		S1<S2 时导通		
AND <> FNC236		S1 ≠ S2 时导通		
AND ≤ FNC237		S1 ≤ S2 时导通		
AND ≥ FNC238		S1 ≥ S2 时导通		

(2) 使用说明

AND*指令的使用如图 6-128 所示。当常开触点 X000 闭合且计数器 C10 的计数值等于 200 时，驱动 Y010；当常闭触点 X001 闭合且 D0 中的数据不等于-10 时，将 Y011 置位；当常开触点 X002 闭合且 D10、D11 中的数据小于 678493 时，或者触点 M3 闭合时，驱动 M50。

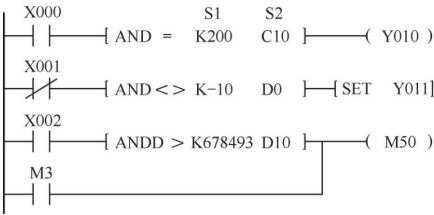


图 6-128 AND*指令的使用

3 触点比较 OR*指令

OR*指令是并联型触点比较指令，其功能是将[S1]、[S2]两个源操作数进行比较，若结果满足要求则执行驱动。OR*为 16 位指令，ORD*为 32 位指令。

(1) 指令格式

触点比较 OR*指令格式如表 6-107 所示。

表 6-107 触点比较 OR*指令格式

指令助记符与功能号	指令形式	导通条件	操作数	
			S1	S2
OR=FNC240		S1=S2 时导通	K、H、 KnX、KnY、 KnM、KnS、 T、C、D、V、Z	K、H、 KnX、KnY、 KnM、KnS、 T、C、D、V、Z
OR > FNC241		S1>S2 时导通		
OR < FNC242		S1<S2 时导通		
OR <> FNC244		S1 ≠ S2 时导通		
OR ≤ FNC245		S1 ≤ S2 时导通		
OR ≥ FNC246		S1 ≥ S2 时导通		

(2) 使用说明

OR*指令的使用如图 6-129 所示。当常开触点 X001 闭合时, 或者计数器 C10 的计数值等于 200 时, 驱动 Y000; 当常开触点 X002、M30 均闭合, 或 D100 中的数据大于或等于 100000 时, 驱动 M60。

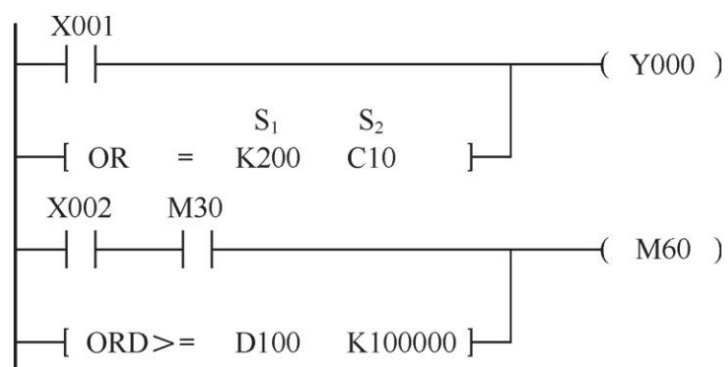


图 6-129 OR*指令的使用