

Koyo

Value & Technology

步进马达轴定位模块 E—02PP

技术资料

[第一版]

光洋电子(无锡)有限公司

目 录

第一章 步进马达轴定位模块 E—02PP 简介	1
1. 1 E—02PP 模块的主要技术指标和特点:	1
1. 2 E—02PP 模块的应用	1
1. 3 系统结构图	2
1. 4 E—02PP 模块的端子排列	3
1. 5 步进马达和步进驱动器	3
1. 6 限位开关	3
1. 7 E—02PP 模块的速度控制信号	4
1. 8 E—02PP 模块的接线方法	5
1. 9 LED 的含义	5
第二章 轴定位工作原理	7
2. 1 位置、速度、加速度	7
2. 2 时间速度图	7
2. 3 位置控制的基本原理	8
第三章 E—02PP 模块命令	10
3. 1 命令一览表	10
3. 2 命令说明	12
3. 2. 1 参数设定类命令	12
3. 2. 2 调整类命令	24
3. 2. 3 子程序类和 MOV 类命令	34
3. 2. 4 第二轴控制用命令	44
3. 3 错误的发生和处理	46
第四章 编程方法	48
4. 1 编程注意	48
4. 1. 1 执行遍数和执行次序	48
4. 1. 2 F30~F33 指令的参数格式和选用方法	48
4. 1. 3 返回数据的用法	49
4. 1. 4 指令执行时间和对 SR21/SR22/SE-22 扫描时间的影响	49
4. 2 编程举例	49
4. 2. 1 例 1: 通过 PLC 指令使马达作调整运动动作	50
4. 2. 2 例题: 通过 PLC 指令使马达进行定位运动	50
4. 2. 3 例外: 用两个按钮和一个开关实现快速, 慢速调整	50
4. 3. 4 例行: 185, 186 指令的应用	51
4. 3. 5 例 5: 200, 201, 202 指令的应用	54
4. 3. 6 例 6: Y 轴运动举例	55
4. 3. 7 例 7: 一个系统中多个 E-02PP 同时动作的应用	56
4. 2. 8 例 8: 采用 E-02PP 模块控制磨床的磨头定位	56
4. 2. 9 例 9: 速度低于 1vu 时的处理	68
附录: 智能模块控制指令组说明	69

第一章 步进马达轴定位模块 E—02PP 简介

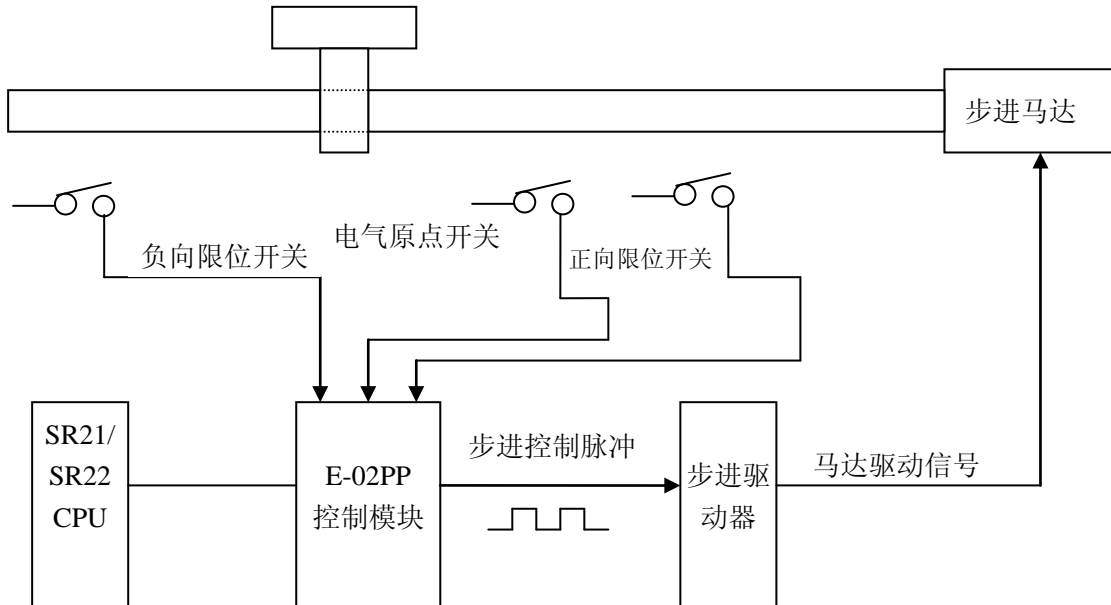
用于轴定位的马达通常有两类：一类为伺服马达、一类为步进马达。SR21/SR22/SE-22/SE-22 系列 PLC 具有两类轴定位控制模块，分别用于伺服马达和步进马达。E—02PP 是用于步进马达的开环轴定位模块。

1. 1 E—02PP 模块的主要技术指标和特点：

占用 I/O 点数	8 点
控制轴数	两轴
位置指定方式	绝对、相对、间接指定
位置指定范围	-8388608~8388607 CT
目标带宽度	0~255 可编程
速度控制脉冲	NPN 晶体管 OC 输出，耐压 48V，电流 200mA，3 种输出方式
可编程速度	1~60000vu，即 1.67Hz~100KHz
可编程加速度	1~500000au
运动方式	a 调整运动，b 自动找原点，c 定位运动，d 可变速动
限位开关输入	4 点，DC24V，源输入
定位命令存储器	可存放 256 条命令
子程序存储器	可存放 10 个子程序最多包含 512 条命令

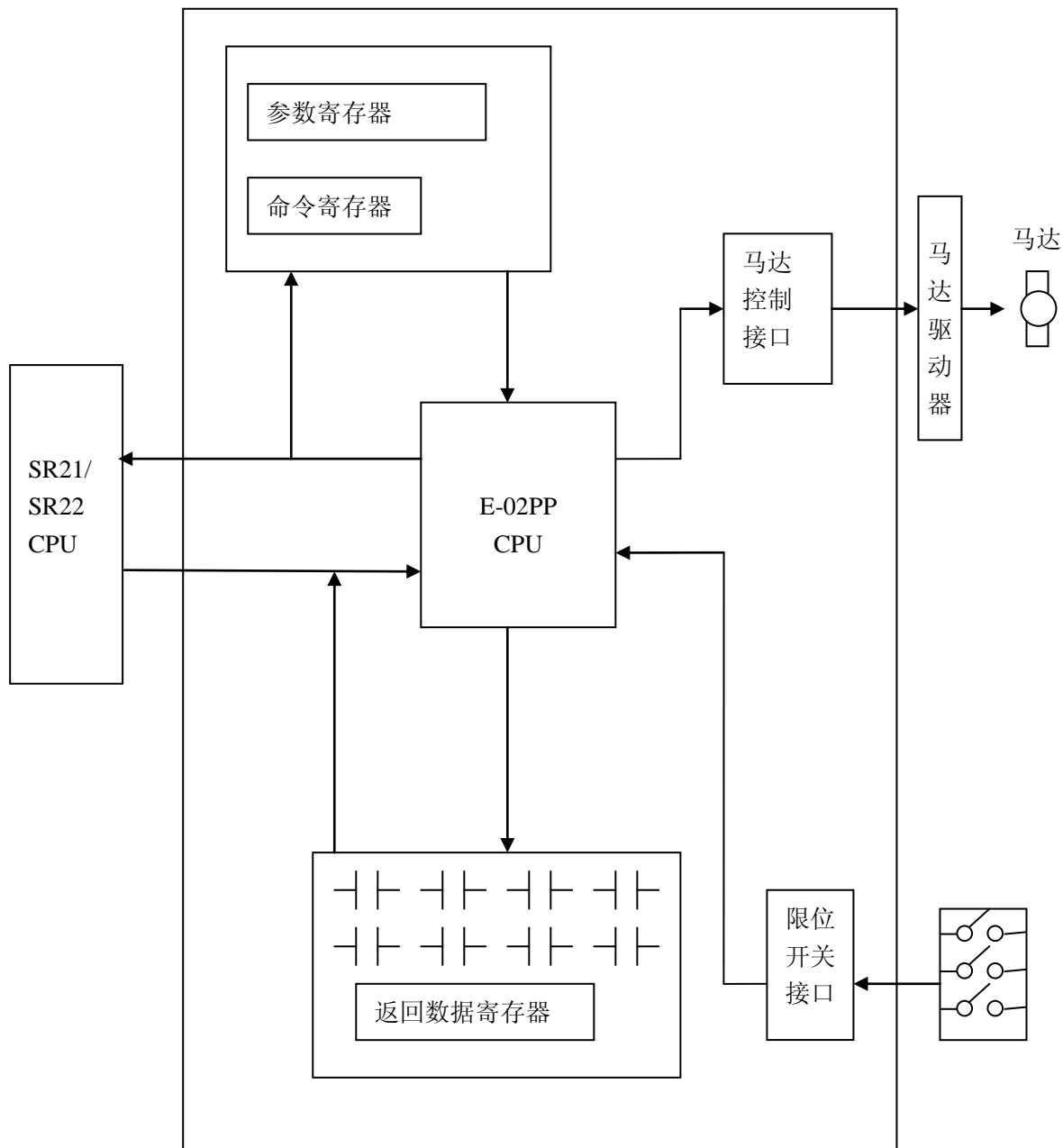
1. 2 E—02PP 模块的应用

E—02PP 步进马达控制模块的典型应用如下所示：



1. 3 系统结构图

将 E—02PP 装入 SR21/SR22/SE-22 电源机架的 I/O 槽上，并按要求连接限位开关和马达驱动器以后，SR21/SR22/SE-22 系列 PLC 与轴定位模块的关系如下所示：

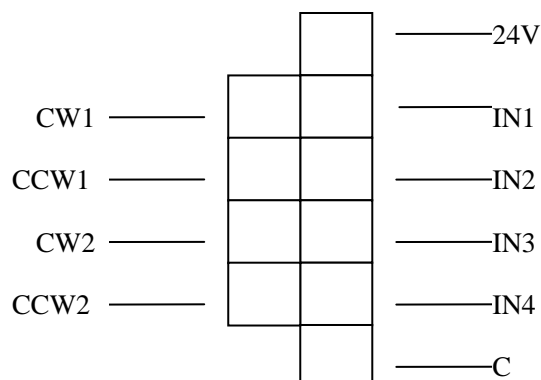


轴定位模块与 SR21/SR22/SE-22 的关系

SR21/SR22/SE-22 系列 PLC 可以通过 I/O 总线对 E—02PP 模块发出各种控制命令，E—02PP 模块接收返回信息。E—02PP 模块根据 PLC 的命令以及当时的运动状态，相应的速度控制信号，使马达按要求运转，达到位置控制目的。

1. 4 E—02PP 模块的端子排列

步进控制模块共有 10 个端子，端子的排列图如下：



各端子功能如下：

CW1、CCW1 为主控制轴的脉冲输出端子，其输出可在三种方式中选择一种。（详见 2 号命令说明）

CW2、CCW2 为辅助轴（以下辅助轴亦称 Y 轴）的脉冲输出端子，其输出同样可在三种方式中选择一种。

IN1~IN4 为 4 个输入端子，各路与 C 端子接通时为 ON，断开时为 OFF。

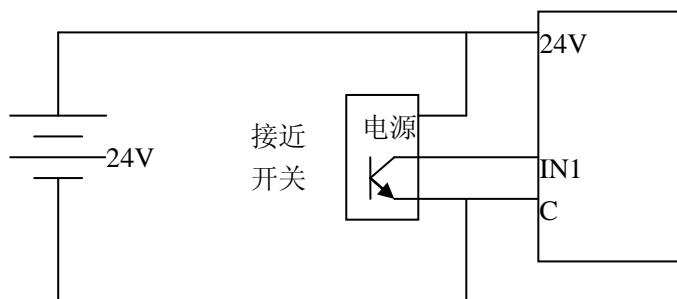
1. 5 步进马达和步进驱动器

E—02PP 模块对马达没有特殊的要求，只要求选择相应的步进驱动器。对步进驱动器的要求是输入信号电压阻抗与步进控制模块的输出信号相匹配。而且步进驱动器所能接收的步进控制脉冲相位关系必须满足步进模块的三种输出方式中的一种。（步进模块的三种输出方式详见第三章的 2 号命令说明）

1. 6 限位开关

限位开关接口上连接的限位开关，可以采用机械触点开关，但要求通断过程中触点抖动尽可能小的开关，推荐使用无触点接近开关。

如采用集电极开路的晶体管输出信号与 IN1 相连时，接线方法如下：

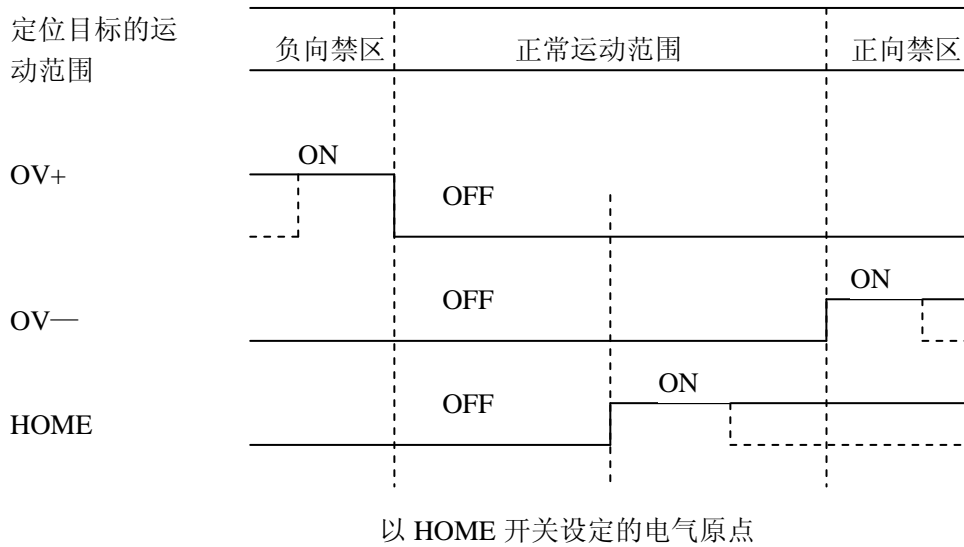


在 E—02PP 模块上可以接四个输入开关，从 IN1~IN4，如果输入开关被指定为限位开关或原点开关用时，IN1 为正向限位 OV+、IN2 为负向限位 OV-、IN3 为原点开关 HOME。限位开关安装满足下列要求：

在正常运动范围内，OV+ 和 OV- 开关的状态为 OFF，当定位目标越过正方向限定点时，OV+ 的状态为 ON，当定位目标越过负方向限定时，OV- 的状态为 ON。在定位目标的许可运动范围内

的一个设定点，HOME 开关有了一个从 OFF 到 ON 的变化，当定位目标在设定点的负方向时 HOME 开关为 OFF。

这三个限位开关状态和定位目标的位置关系如下所示，图中实线表示正常的状态，虚线表示许可的状态。



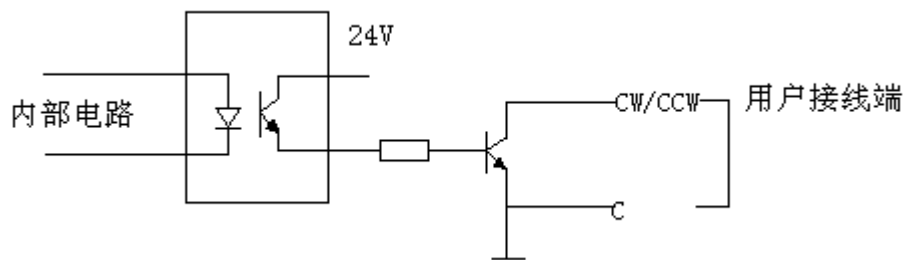
输入电路的电气特性如下：

- 输入电流：<math><18\text{mA}</math>
- 输入阻抗：>1.8K Ω
- ON 电平：<math><3\text{V}</math>
- OFF 电平：>18V

1. 7 E—02PP 模块的速度控制信号

E—02PP 的速度控制输出是脉冲信号，该模块可以输出 3 种不同方式的脉冲，适应不同的步进马达驱动器。

E—02PP 的脉冲输出端子 CW1/CCW1，CW2/CCW2 的输出电路示意图如下：

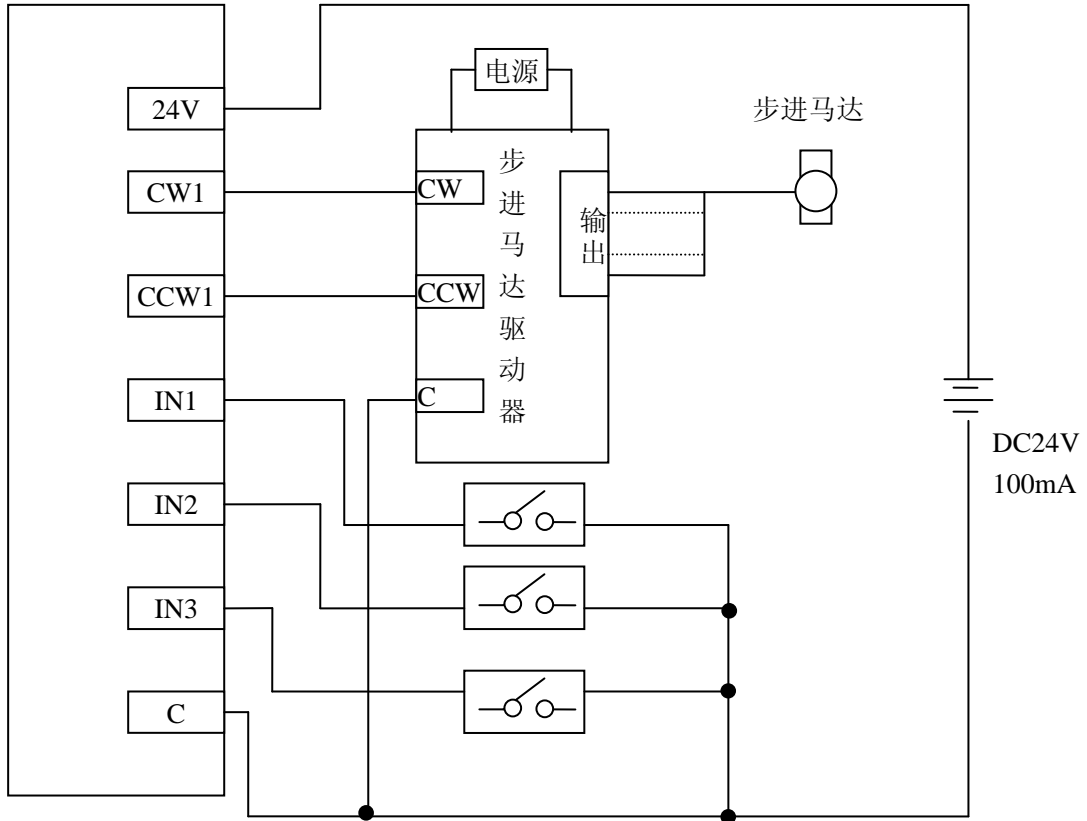


输出电路的电气特性如下：

- 耐压：48V（截止时， $I_C < 0.1\text{mA}$ ）
- 最大电流：200 mA（导通时， $V_0 < 1\text{V}$ ）

1. 8 E—02PP 模块的接线方法

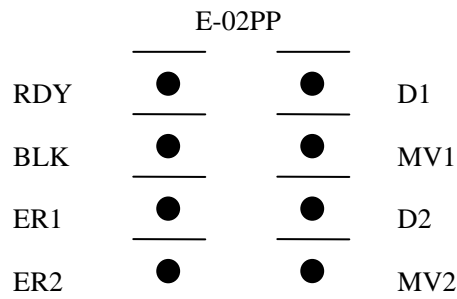
E—02PP 的接线方法如下图所示：



接在 E—02PP 模块上的 24V 电源要求稳压，在总容量不超过限制值的情况下，可使用电源框架上的 24V 电源端子上供电。

1. 9 LED 的含义

E—02PP 模块的面板上有 8 个 LED，用于表示模块的工作状态。LED 的排列如下图所示：



各 LED 的含义如下：

RDY：点亮时表示 E—02PP 可以接收来自 CPU 的命令。

BLK: 0.6 秒点亮，0.6 秒熄灭，表示模块无故障。若有故障，则变为常亮或常灭。


ER1: 表示主控制轴动作中发生错误，错误类型可以参考错误号。

ER2: 只在 PLC 未处于 RUN 状态时点亮。

D1: 当主控制轴正在正转时常亮；当主控制轴正在反转时闪烁；当主控制轴未进行动作时熄灭。

MV1: 当主控制轴正在进行动作时或等待、延时时点亮。

D2:  D2、MV2 为辅助轴的动作状态指示灯，含义与 D1 和 MV1 相似。

MV2: 

第二章 轴定位工作原理

2.1 位置、速度、加速度

【1】位置：

轴定位系统的主要对象是位置。一般情况下，马达的转动用传动机构转化为被控对象的直线运动或其它形式的运动，但单轴定位系统的位置可以用一维坐标来表示。

在 E-02PP 中，表示位置的最小单位是脉冲当量 CT，即步进马达的一步为一个 CT。在对 E-02PP 的位置参数进行编程时，实际尺寸例如 mm、英寸等应一律换算成 CT。

本模块支持位置的绝对表示、相对表示和间接表示法。绝对表示法是用坐标原点到指定点的 CT 数来表示一个位置。相对表示法用轴现在所处的位置到指定点的 CT 数来表示一个位置。间接表示法是指定一个 E-02PP 的内部寄存器，而寄存器的内容是用绝对表示法来表示位置。以下是采用不同位置表示法的例子：

- (1) PMOV8 1000 含义：向前运动，移动距离 1000CT 精确定位。
- (2) PMOV9 1000 含义：运动到坐标为 1000CT 的点精确定位。
- (3) PMOV0 含义：运动到 0 号位置寄存器规定的坐标点精确定位。

上面的 (1) 中，所用的位置参数为相对表示，无论轴现在处于什么位置，这个命令总是使轴向前移动 1000CT。而 (2) 的情况则不同，如果轴现在的位置坐标大于 1000，那么轴就反向运动；反之，轴正向运动；如果轴现在的位置坐标等于 1000，那么轴不作任何运动。

【2】速度：

E-02PP 是通过控制马达运动速度来进行定位的。本模块所用的表示速度的单位是 1 分钟内移动的位置 CT 数，采用的实际单位是 vu，1vu=100CT/分。被控对象的实际速度，例如 mm/秒，作为本模块的速度参数进行编程时，应换算成多少个 vu。

计算例：马达每圈步速=500CT/R，马达转速=1000R/M 时，
速度=1000 (R/M) × 500 (CT/R) =500000 (CT/M) =5000 (vu)

【3】加速度：

加速度是单位时间内的速度改变量，本模块所用的加速度表示单位是 au，1au=10vu/秒，即在 1 秒钟内改变 10 个 vu 时，加速度是一个 au。编程时的加速度一律换算成 au。

计算例：马达每圈步速=500CT/R，要求马达在 1 秒钟内速度从 0 增加到 300R/MIN 时：

$$\text{加速度} = \frac{300 \text{ (R/M)} \times 500 \text{ (CT/R)}}{1 \text{ (Sec)}} = 150000 [\text{CT}/(\text{M.S})] = 1500 \text{ (vu/s)} = 150 \text{ au}$$

2.2 时间速度图

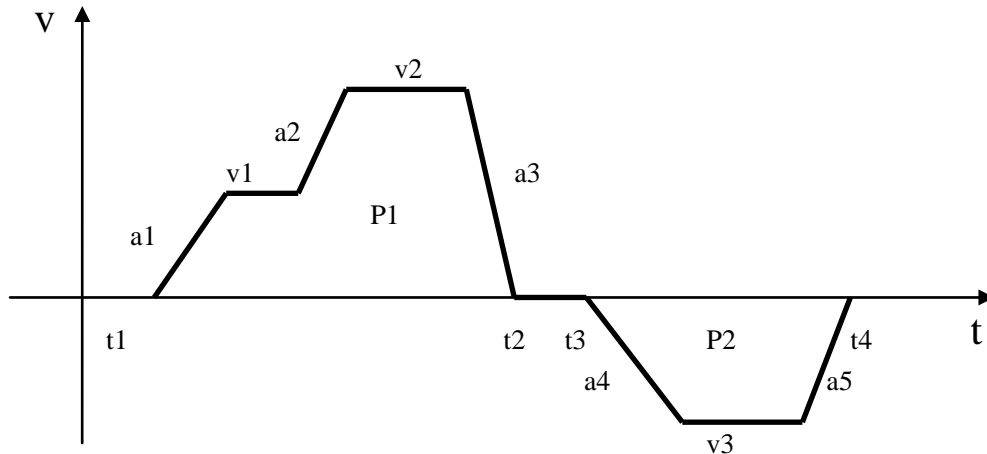
轴的运动涉及到时间、速度、加速度、位置等各种量，为了描述的便利，使用时间速度图来描述运动过程。在时间速度图上，上述各种量都可以直观的反映出来。

时间速度图采用直角坐标系，以横轴表示时间，纵轴表示速度，轴的运动过程可以用这个坐标系中的一条曲线来表示。曲线的纵坐标表示速度，如果曲线的纵坐标为正，则速度为正，表示正向

运动；如果曲线的纵坐标为负，则速度为负，表示反向运动。

根据加速度的定义， $Acc=dv/dt$ ，所以曲线斜率就是加速度。

在一定时间内移动的距离 $\Delta P = \int_{t_1}^{t_2} v dt$ 所以 t_1-t_2 之间运动的总位移量就是曲线到横轴之间所围的面积，这个面积处于横坐标之上时，就表示正向运动的位移；这个面积处于横坐标之下时，就表示负向运动的位移。



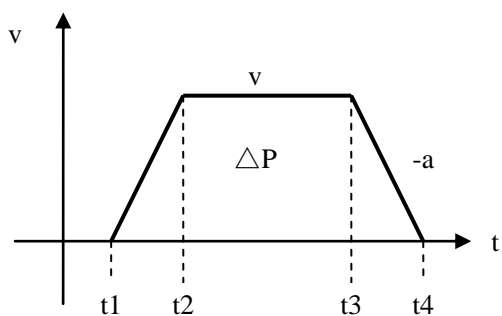
上图中：

- (1) t_1 、 t_2 、 t_3 、 t_4 表示四个时刻。
- (2) v_1 、 v_2 、 v_3 表示三个速度， v_2 比 v_1 大， v_3 是负速度，表示反向运动。
- (3) a_1 、 a_2 、 a_3 、 a_4 、 a_5 表示五个加速度。 a_1 、 a_2 为正加速度，此时速度变大， a_2 的加速度较大； a_3 负加速度，此时速度变小； a_4 、 a_5 是对反向运动的加速度， a_4 使速度的绝对值变大， a_5 使速度的绝对值变小。
- (4) 面积 P_1 表示 t_1-t_2 之间的位移是正向位移；面积 P_2 表示 t_3-t_4 之间的位移是负向位移； t_2-t_3 之间的速度为零，也就是没有位移。

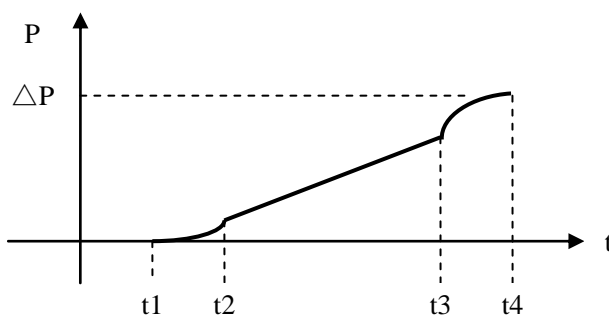
2. 3 位置控制的基本原理

送到轴定位模块的各种命令，是对运动状态的描述，模块以这些命令为依据进行速度、位置控制。

一般地，一个定位命令要求轴移动到一个新的位置时，模块先计算出理论的时间速度图，然后以这个时间速度图为准来控制轴，使之最后达到规定的位置。典型的速度时间图是一个梯形，也就是说，轴先以用户设定的加速度 a 加速到用户设定的速度 v ，然后定速运动一定的时间；再以用户设定的加速度 a 减速运动到速度变为 0。速度到 0 时，轴移动的距离正好是命令规定的距离置 ΔP ，如图所示：



时间速度图



时间位置图

根据这个时间速度图，相应地可以计算出一个时间位置图，E-02PP 模块就以时间速度图为依据输出速度命令脉冲。到 ΔP 个脉冲全部输出以后，模块停止输出脉冲。这时马达正好停在希望到达的位置。

第三章 E—02PP 模块命令

SR21/SR22/SE-22 通过向 E—02PP 发送各种命令来控制 E—02PP 模块的工作过程。E—02PP 则根据接收到的命令来控制轴的运动，各部分的关系如 1.3 节“系统结构简图”所述。本章将详细描述各命令的作用和使用方法。并说明返回数据的格式和错误码的含义。在进行编程之前，请仔细阅读本章。

3.1 命令一览表

参数设定类命令

命令编号	命令助记符	参数	参数长度	重复执行	子程序内许可	含义
00	RST	—	0	○	×	模块复位
01	INMODE	指定	4	○	×	IN1~IN4 用法指定
02	STEPMD	1-3	4	×	×	步进脉冲输出方式选择
03	PMAX	P	8	○	×	正向限位软开关设定
04	PMIN	P	8	○	×	反向限位软开关设定
05	PHOME	P	8	○	×	电气原点位置座标设定
06	RDATA	0-15	2	○	×	选择返回数据内容
09	STEPL	P	4	○	○	步进距离设定
10	VLMT	V	8	○	×	速度极限值设定
11	FJOGV	V	8	○	×	快速调整速度设定
12	SJOGV	V	8	○	×	慢速调整速度设定
13	JOGACC	A	8	○	×	调整加速度设定
14	PMUTI	指定	4	×	×	位置倍率设定
15	VMUTI	指定	4	×	×	速度倍率设定
16	PTELL	P	8	○	×	当前位置值设定
18	WATCH	时分秒	6	○	×	时钟值设定
19	ABTACC	A	8	○	×	出错停止加速度设定

(注) ○：使用许可；×：使用不可；参数长度为参数的 BCD 码位数

调整类命令

命令编号	命令助记符	参数	参数长度	重复执行	子程序内许可	含义
20	FJOGP	—	0	○	×	快速正向调整运动
21	FJOGN	—	0	○	×	快速负向调整运动
22	SJOGP	—	0	○	×	慢速成正向调整运动
23	SJOGN	—	0	○	×	慢速负向调整运动
24	JOGOFF	指定	2	○	○	停止调整/带参数调整
27	HOMEMOD	P	4	○	×	指定找原点方式
28	FHOME	—	0	○	×	找原点
29	RESTART	—	0	○	×	结束等待，继续运动
30	ABORT	—	0	○	×	中止运动取消所有命令
31	CLRERR	—	0	○	×	清除错误
33	VCHG	—	0	○	×	可变速成运动变速
34	PARSTOP	—	0	○	○	局部工序停止

子程序类命令

命令编号	命令助记符	参数	参数长度	重复执行	子程序内许可	含义
89	RET	—	0	○	○	子程序结束
90~99	SUBRi	—	0	○	×	子程序定义开始 (i=0~9)

(注) ○：使用许可；×：使用不可；参数长度为参数的BCD码位数

定位控制类（MOV类）命令

命令编号	命令助记符	参数	参数长度	重复执行	子程序内许可	含义
100-107	PMOVi	—	0	○	○	精确定位运动命令
108	PMOV8	△P	8	○	○	
109	PMOV9	P	8	○	○	
120-127	FMOVi	—	0	○	○	多段运动命令
128	FMOV8	△P	8	○	○	
129	FMOV9	P	8	○	○	
130-137	MVPMOVi	—	0	○	○	可变速精确定位运动命令
138	MVPMOV8	△P	8	○	○	
139	MVPMOV9	P	8	○	○	
140-147	DPTABi	P	8	○	○	位置寄存器设定
148	MVFP	△P	4	○	○	可变速定位保留距离

150-157	DPABADDi	P	4	○	○	位置寄存器正调整
160-167	PTABSUBi	P	4	○	○	位置寄存器负调整
170-177	DVTABi	V	8	○	○	可变速命令速度表设定
178	BASVEL	V	8	○	○	基础速度设定
179	MOVVEL	V	8	○	○	运动速度设定
180	MOVACC	A	8	○	○	运动加速度设定
181	DELAY	T	5	○	○	延时
182	STEP	步号	2	○	○	步号设定
183	WAIT	-	0	○	○	等待
185	PARBGN	-	0	○	○	局部可中止工序开始
186	PAREND	-	0	○	○	局部可中止工序结束
200	CONTON	0-8	2	○	○	接通指定的应答信号
201	CONTOFF	0-8	2	○	○	断开指定的应答信号
202	CONTSCN	0-8	2	○	○	使指定应答信号产生单脉冲
190-199	CALLi	REP	4	○	○	调用子程序

(注) ○：使用许可；×：使用不可；参数长度为参数的BCD码位数

第二轴控制命令

命令编号	命令助记符	参数	参数长度	重复执行	子程序内许可	含义
40	SECONDV	V	8	○	○	Y轴动作速度设定
41	SECONDP	P	8	○	○	Y轴动作距离设定
42	SECONDD	指定	4	○	○	Y轴动作方向设定
43	SECPOS	—	0	○	○	Y轴定位动作启动
44	SECVEL	—	0	○	○	Y轴定速动作启动
45	SECCONV	—	0	○	○	Y轴定速转定位控制

3. 2 命令说明

3. 2. 1 参数设定类命令

参数设定类命令是为 E-02PP 模块设定各种参数，对大部分参数来说，一经设定，模块就立即以设定的参数进行控制，但 FJOGV、SJOJV、JOGACC 命令设定的参数只对以后的调整命令起作用，已经在进行的调整运动的速度和加速度不再改变。

参数设定类命令的命令码在 20 以下。

RESET 复位命令

命令编码：0

参 数：无

允许执行条件：任何情况下均可执行。

执行后影响：中止所有处理，回复到初始状态。

SR21/SR22/SE-22 指令：

F30 0

该命令使 APM 模块复位，复位的过程与加电开机的效果相同，复位以后，至少要等待 1 秒时间，模块才能接收新的命令。模块接收到复位命令以后，所有的参数都变成默认值，所有的子程序都被取消；如果马达正在运转时进行复位，模块先以出错停止加速度减速到 0 后，再实施复位。

SR21/SR22/SE-22 从 RUN 状态变为其它状态时，对 E-02PP 模块来说，都相当于接收到一个 RESET 命令。

INMODE 输入信号用法指定命令

命令编码：1

参 数：4 位指定数字

允许执行条件：ERROR 触点断开

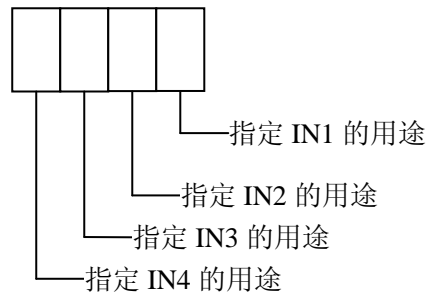
执行后影响：输入信号 IN1~IN4 作为不同用途的信号使用

SR21/SR22/SE-22 指令

F50 x x x x

F32 1

该命令的功能是指定 IN1~IN4 的用途，该命令应带一个 4 位数的参数，参数格式如下：



指定用途的数字分别为：

0：作限位开关或原点开关用，IN1~IN3 中选作 0 的信号含义如下：IN1：正向限位；N2：负向限位；IN3：原点开关。

1：作变速信号用，可以选择 1 个以上的变速信号。

5：作局部工序中止信号用，可以选择 1 个以上的中止信号。

例： F50 5501

 F32 0001

将 IN1 定为变速信号，将 IN2 定为负向限位开关，IN3 和 IN4 定为局部中止命令。

无论指定 IN1~IN4 为何种用途，这 4 个输入信号的状态都会在返回数据的 16 个开关量中反映出来。

请注意：2~4 和 6~9 是保留的数字，请不要使用。在轴处于停止状态时，才能对 IN1~IN4 进行重复设定。

SETPMD 步进马达脉冲信号输出方式选择

命令编码：2

参 数：1~3 或 101~103（默认值：3）

允许执行条件：只允许在复位后未执行过其它命令之前执行一次

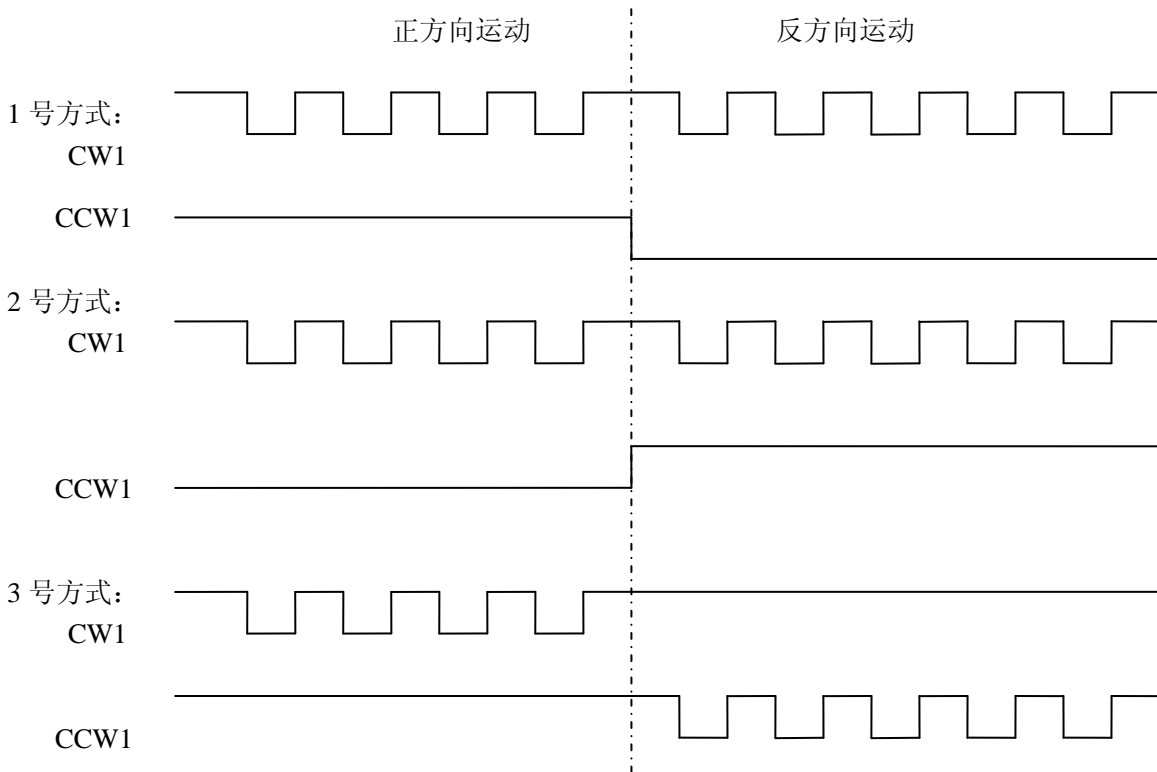
执行后影响：规定输出控制脉冲的相位关系

SR21/SR22/SE-22 指令：

F50 × × × ×

F32 2

该命令的作用是选择 E-02PP 的 CW1/CCW1, CW2/CCW2 端子上输出信号的相位关系。该命令允许带 4 位参数，4 位参数的高 2 位为 00 时，是对主控制轴的输出方式设定，参数可以是 1~3。此时，主控制轴的输出脉冲如下图所示：



（上图中低电平为输出端晶体管导通，即与 C 点接通）

当 4 位参数的高位不等于 0 时，允许的值为 101~103，分别对 Y 轴的输出方式设定为上图中的 3 种方式之一。

注意：该命令必须在执行除 RESET 以外的其它命令之前执行，而且方式一经设定，除非停电或 SR-21 退出 RUN 状态，否则不可改变。默认方式，对主控制轴和 Y 轴均为方式 3。

PMAX 正向限位软开关设定

命令编码：3

参 数：0~+8388607（默认值：+8388607）

参数类型：带符号整数。

单 位：CT

允许执行条件：ERROR 触点断开

执行后影响：正向限位监视起作用

SR21/SR22/SE-22 指令：

F50 ××××（高四位数字）

F60 R576

F50 ××××（低四位数字）

F33 3

该命令用于设定定位行程的正方向最大值。

如果模块对实际位置的跟踪有效，即 POK 触点接通时，一旦模块内部测到的实际位置大于所设定的值，就会发生 53 号错误。所谓位置跟踪有效的条件是：

- 1) 成功地完成了找原点 FHOME（命令码 28）命令。
- 2) 或者已执行过 PTELL（命令码 16）命令。
- 3) 而且在 1) 或 2) 的条件下，未发生过错误。

在 POK 线圈是断开时，即使内部的当前实际位置值大于设定值的话，也不会发生越界错误。如果参数设定为+8388607，则与不设定一样，正向限位软开关就不起作用。

PMIN 负向限位软开关设定

命令编码：4

参 数：-8388608~0（默认值：-8388608）

参数类型：带符号整数

单 位：CT

允许执行条件：ERROR 触点断开

执行后影响：负向限位监视起作用

SR21/SR22/SE-22 指令：

F50 ××××（高四位数字）

F60 R576

F50 ××××（低四位数字）

F33 4

该命令用于设定定位行程的负方向最大值。

如果模块对实际位置的跟踪有效即 POK 触点接通时，一旦模块内部检测到的实际位置小于所设定的值，就会发生 54 号错误。所谓位置跟踪有效的条件是：

- 1) 成功地完成了找原点 FHOME（命令码 28）命令。
- 2) 或者已执行过 PTELL（命令码 16）命令。
- 3) 而且在 1) 或 2) 的条件下，未发生过错误。

在 PLK 线圈是断开时，即使内部的当前实际位置值小于设定值的话，也不会发生越界错误。如果参数设为-8388608，则与不设定一样，负向限位软开关就不起作用。

注意：本命令设定的参数不能比 PMAX 命令的参数大。

PHOME 电气原点位置坐标

命令编码：5

参 数：-8388608~+8388607（默认值：0）

参数类型：带符号整数

单 位：CT

允许执行条件：ERROR 触点断开

执行后影响：找原点时，根据该值确定电气原点到机械原点之间的距离

SR21/SR22/SE-22 指令：

F50 × × × ×（高四位数字）

F60 R576

F50 × × × ×（低四位数字）

F33 5

该命令用于设定电气原点坐标，电气原点坐标的含义请参阅 FHOME 命令的说明。

RDATA 返回参数的选择

命令编码：6

参 数：0~18（默认值：0）

允许执行条件：ERROR 触点断开

执行后影响：返回数据格式切换

SR21/SR22/SE-22 指令：

F50 × ×（返回数据号 0~18）

F31 6

该命令用于选择返回数据。

返回数据，是 E—02PP 模块向 SR21/SR22/SE-22 CPU 报告模块的内部状态和定位目标目前所处的状态的数据。

用 SR21/SR22/SE-22 的 F40、F41、F42 指令，可以从 E—02PP 模块获取返回数据，E—02PP 的数据由两部分组成。

其中第一部分是 8 个开关量触点和一个 8 位 BCD 码的数据，8 个触点表示模块目前所处的状态，用户可以象使用一般的输入模块上的输入点一样来使用这些点。在一般的应用状态下，不要对这些点使用 OUT 那种会改变其状态的指令。这 8 触点与普通输入点的差别在于：普通输入点的状态更新是在 PLC 的扫描周期中自动完成的；但模块上的 8 个触点，只有在执行 F40、F41、F42 命令以后才更新。模块上的 8 个触点的定义号和模块的装置位置有关，此时可把 E—02PP 模块看作 8 点 I/O 模块来计算定义号。

这 8 个点的相对定义号为 0~7，各相对定义号对应的触点名称如下：

0	1	2	3	4	5	6	7
ERROR	APMRD	IP	EXEMOV	MOVING	P2ND	POK	RDOK

模块上 8 个状态触点的含义：

ERROR：模块发现某种错误或引起错误条件，此时，模块只能执行复位或 JOG 命令，错误类型在错误号上表示，该触点闭合时，模块面板上的 ERROR 也同时发光。

- APMRD:** 模块处于准备好状态，含义与模块面板上的 **RDY** 灯相同。只有该触点闭合时，模块才可以接收命令，闭合时，模块面板上的 **RDY** 灯同时发光。
- IP:** 轴被精确地定位在目标带内时闭合。
- EXEMOV:** 正在执行 **MOV** 类命令时闭合。
- MOVING:** 模块正在控制马达运转或正在执行 **MOV** 类指令，与模块面板上 **MOV** 灯的含义相同。
- P2ND:** 第二轴运动时点亮。
- POK:** 模块内部记忆的位置反映定位目标的物理位置，该触点在成功地找到原点或执行 **PTELL** 命令后闭合。在发生错误时断开。
- RDOK:** 返回数据正常，如果该触点断开，表示现在的返回数据并不反映真实信息，模块还未建立正确的返回数据，但 8 个触点的状态是正确的。一般在执行 **RDATA** 命令后，该触点就会在一段时间内断开，但断开的时间一般不大于 5ms。

第一部分的返回数据的其它部分是一个 8 位 BCD 码的数据。

用 **SR21/SR22/SE-22** 的 **F40** 指令时，这返回数据寄存器的内容不取到 **PLC** 的 **CPU** 内。

用 **F41** 命令时，只把返回数据寄存器的低半部分 4 位 BCD 码取到 **CPU** 的数据累加器之内。

用 **F42** 命令时，返回数据寄存器的高半部分 4 位 BCD 码取到 **CPU** 的辅助累加器内，即数据寄存器 **R576** 内；低半部分 4 位 BCD 码取到 **CPU** 的数据累加器之内。

数据寄存器的内容可根据需要进行不同的选择，各种选择的含义如下：

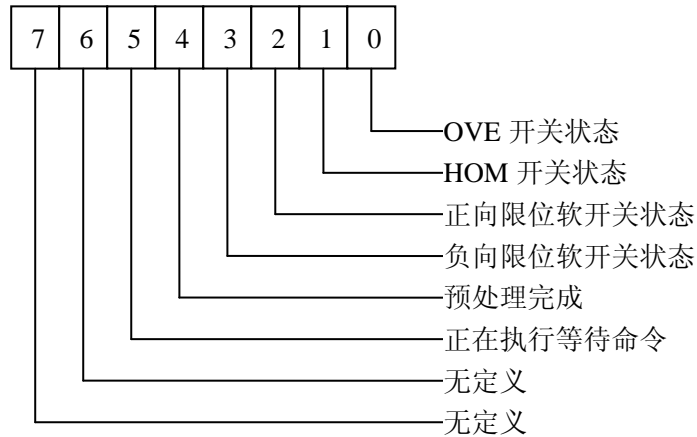
0 号	版本号	错误号	步号	状态触点	11 号	步号	S*	当前实际位置	
1 号	S	当前实际位置			12 号	步号	S*	当前理论位置	
2 号	S	当前理论位置			13 号	步号	S*	位置误差	
3 号	S	位置误差			14 号	步号	S*	当前实际速度	
4 号	S	当前实际速度			15 号	步号	S*	当前理论速度	
5 号	S	当前理论速度			18 号	步号	时	分	秒
8 号	时	分	秒	百分秒					

注：S 用 8 表示负数，0 表示正数。

S*以 BCD 码中权为 8 的 bit 为 1 表示负数，为 0 表示正数。

上述返回数据的含义，分别解释如下：

1) 软件版本号是二位数字，错误号的含义请参阅“错误的发生和处理”，步号是目前正在进行的定位命令的序号，根据序号可以知道现在正在进行何种动作。状态标志是 8 位二进制码，可以认为它也是 8 个触点，这 8 个触点的含义如下：



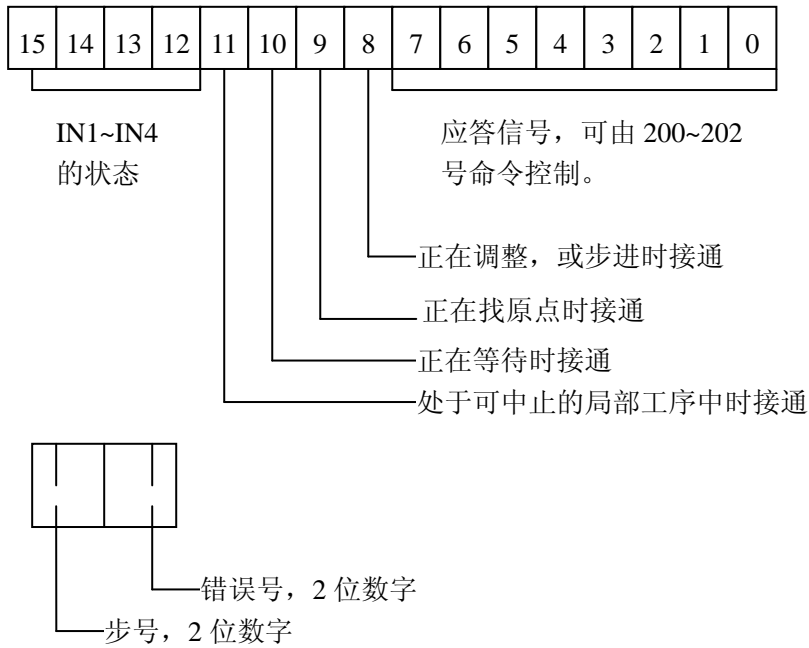
上述状态中，预处理完成的含义是：在 MOV 类命令缓冲区中的命令都已计算成速度时间图，MOV 类命令缓冲区中已空。

- 2) 当前理论位置：是根据理论速度时间图而求出的相应的位置时间图上现在的理论位置值。
- 3) 当前实际位置：这是根据模块自身发出的位置控制脉冲而得出的实际位置值。
- 4) 位置误差：是指令位置与实际位置之差。
- 5) 当前实际速度：是过去 0.6 秒的平均速度，该值每 150ms 计算一次，单位为 vu。
- 6) 当前理论速度：是理论速度时间图上，现在时刻的理论速度值（单位为 vu）。

在返回数据选择以后，立即进行返回数据的读入（SR21/SR33 的 F4×命令）的话，可能得到的返回数据并不是当前所指定的内容，而要延迟一段时间才能变为指定的内容，这个延迟在 5ms 以下，隔一个 PLC 的扫描周期以后再读入数据时，可以保证读出的就是指定的内容。

上述 1~5 号的返回数据，均为 8 位数字表示，但在某些定位系统中，可能速度、位置的实际应用范围要小得多，这时提供了一种用较少位数表示返回数据，同时又返回步号的选择。作这样的选择时，在 32bit 返回数据寄存器上，低 24bit 以 6 位 BCD 码表示相应的数值，高 8bit 的 2 位 BCD 码表示步号，6 位 BCD 码中最高一位 BCD 码要同时表示符号。这时，这个数字有可能不是 BCD 码，例如 -123456 则被表示为 923456，而 -223456 时，最高位上用 8+2 表示，这时变为 16 进数 A，这就需进行适当处理才可以使用，建议用这种表示方法时，表示范围不要超过 -99999~+99999，不得已时可用到 -799999~+799999，但如再超出范围，数字就不可能再表示了，仅仅只有低 5 位的尾数和符号位，还有参考价值，如 -1000000 则被表示为 800000。

如果需要上述格式的返回数据，只有在选择返回数据时将 1~5 和 8 的选择改为 11~15 和 18 即可。第二部分返回数据包括 16 个开关量和 4 位 BCD 码数据。



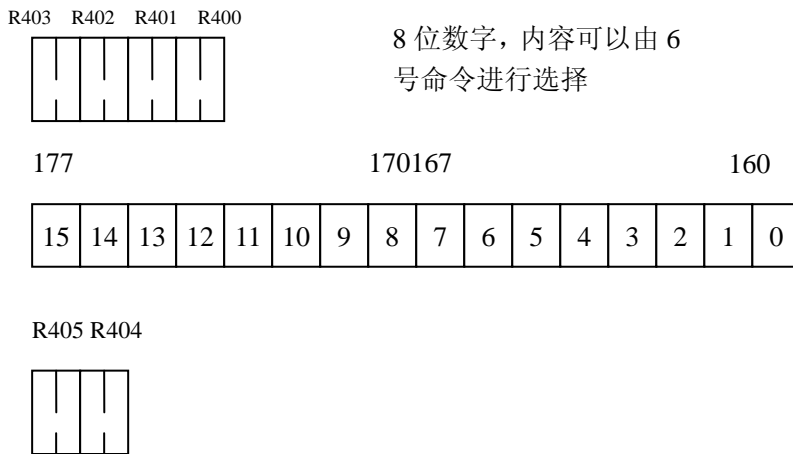
所以，取返回数据的程序应如下编制，并放在程序的最前面：

```

F42
F60 R400
F50 R576
F60 R402
F41
F60 R16
STR 0
F41
F60 R404
    
```

该程序假定 E-02PP 插在 0 号槽上，返回数据第一部分的 8 位数字放在 R400~R403 中，返回数据第二部分的 16 个开关量放在 160~177 的内部继电器中，4 位数字放在 R404~405 中，这 4 位数字只在出错时才记录。

这样，返回数据的结构如下：



以上程序中，STR 0 是因为 E-02PP 插在 0 号槽上，所以定义号 0 为 E-02PP 的出错信号。

STEPL 单次步进步长设定

命令编码：9

参 数：1~9999

参数类型：无符号整数

单 位：CT

允许执行条件：ERROR 触点断开

执行结果：建立新的单次步进步长

SR21/SR22/SE-22 指令：

F50 × × × ×

F32 9

该命令用于设置单次步进的步长，参数允许范围为 1~9999，关于单次步进的详细内容，见 24 号命令的说明。

VLMT 速度极限值设定

命令编码：10

参 数：1~60000（默认值：60000）

参数类型：无符号整数

单 位：vu

允许执行条件：ERROR 触点断开时

执行结果：建立新的速度限制值

SR21/SR22/SE-22 指令：

F50 × × × × （高位数字）

F60 R576

F50 × × × × （低四位数字）

F33 10

该命令为以后所有速度值设定命令建立一个极限检查值，一旦以后的速度设定超过这个值，就会出现相应的错误信息。如果这个参数设得大于模块最大允许值，模块就自动把速度限制值改为模块的最大允许值；如果把速度极限值设为 0 或负数，将会出现 31 号错误信息。模块最大允许值是 60000vu。

FJOGV 快速调整速度设定

命令编码：11

参 数：1~60000（默认值：5120）

参数类型：无符号整数

单 位：vu

允许执行条件：任何条件下均可执行

执行后影响：快速调整速度改变

SR21/SR22/SE-22 指令：

F50 × × × × （高位数字）

F60 R576

F50 × × × × （低四位数字）

F33 11

该命令用于设定快速调整速度，快速调整速度在 FJOGP（命令码 20）、FJOGN（命令码 21）、FHOME（命令码 28）命令中要使用。

SJOGV 慢速调整速度设定

命令编码：12

参数：1~60000（默认值：512）

参数类型：无符号整数

单位：vu

允许执行条件：任何条件下均可执行

执行后影响：慢速调整速度改变

SR21/SR22/SE-22 指令：

F50 × × × × （高位数字）

F60 R576

F50 × × × × （低四位数字）

F33 12

该命令用于设定慢速调整速度，慢速调整速度在 SJOGP（命令码）、SJOGN（命令码 23）、FHOME（命令码 28）命令中要使用。

JOGACC 调整加速度设定

命令编码：13

参 数：1~500000（默认值：500）

参数类型：无符号整数

单 位：au

允许执行条件：任何条件下均可执行

执行后影响：调整加速度改变

SR21/SR22/SE-22 指令：

F50 × × × × （高位数字）

F60 R576

F50 × × × × （低四位数字）

F33 13

该命令用于设定调整加速度，调整加速度在 FJOGP、FJOGN、SJOGP、SJOGN、FHOME 命令（命令码分别为 20、21、22、23、28）中要使用。

PMULTI 位置倍率设定

命令编码：14

参 数：指定

参数类型：无符号整数。

允许执行条件：ERROR 触点断开

执行后影响：位置倍率改变

SR21/SR22/SE-22 指令：

F50 ××××
F32 14

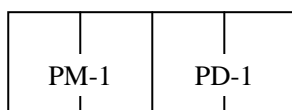
该命令用于设置位置的倍率，位置倍率的含义是编程时距离单位和马达动作的脉冲数的倍数关系，如下式所示：

动作脉冲数=设定距离×PM÷PD（PM：脉冲步数，PD：所定的量值）

例如设模块发 5 个 CT，轴走 3 个距离单位时，则 PM=5，PD=3，编程时指定动作 30 个距离单位时，动作的实际脉冲数为：30×5÷3=50 个脉冲。

计算中若出现小数，则小数点以下的数字舍去，例如动作 10 个距离单位时，则输出为：10×5÷3=16 个脉冲。而且舍去的部分在内部不记忆，所以使用时应注意防止产生积累误差。

PM 和 PD 的设定采用 4 位数字，格式如下：



例如上面要设 PM=5，PD=3 时，可以用如下命令：

F50 0402
F32 14
设 PM=4，PD=1 时用
F50 0300
F32 14

必须注意，设定时必须满足 $PM \geq PD$ 否则不予设定。采用 14 号命令指定位置倍率以后，凡是有关位置的设定，均通过倍率换算，属于位置值的返回数据，同样经过换算；但应注意：不能使经过换算以后的值超过规定的最大脉冲数，例如小于-8388608 或大于+8388607。

VMULTI 速度倍率设定

命令编码：15

参 数：指定

参数类型：无符号整数

允许执行条件：ERROR 触点断开

执行后影响：速度倍率改变

SR21/SR22/SE-22 指令：

F50 ××××
F32 15

该命令用于设定速度值的倍率，即速度值与 v_u 的倍数关系与 14 号命令类似，设定值与 v_u 的关系如下：

$v_u = \text{设定值} \times VW \div VD$

VW 与 VD 的指定方式与 PM、PD 类似，注意事项也一样请参考 14 号命令。

PTELL 设定当前位置值

命令编码：16

参 数：-8388608~+8388607

参数类型：带符号整数

单 位：CT

允许执行条件：IP 触点闭合，ERROR 触点断开，MOVING 触点断开

执行后影响：当前位置的坐标值改变

SR21/SR22/SE-22 指令：

F50 ×××× （高四位数字）

F60 R576

F50 ×××× （低四位数字）

F33 16

该命令用于告诉 E-02PP 模块，轴现在所处的实际位置的坐标。当定位系统不具备自动找原点条件时，可用其他方法先定位到一固定位置后，用本命令建立轴的基准坐标。所以，可以用该命令来实现手动找原点。在执行该指令以后，模块上的 POK 触点就接通，对于单向运动的定位系统来说，可以用这个命令修改轴的当前坐标值，防止轴因为重复单向运动而坐标无限制地增大。

WATCH 时钟实际值设定

命令编码：18

参 数：时：0-23，分：0-59，秒：0-59 （默认开始值：00：00：00）

允许执行条件：ERROR 触点断开

执行结果：内部时钟值改变

SR21/SR22/SE-22 指令：

F50 ×× （小时）

F60 R576

F50 ×××× （分，秒）

F33 18

该命令用于设定内部时钟的实际值，模块内部有一个以时分秒格式计时的实际时钟，在复位以后，这个时钟的值是 0 时 0 分 0 秒，为了使这个时钟与标准时间同步，可以用本命令设置实际时间。设置以后，时钟就可以表示实际时刻，时钟的内容可以用 8 号或 18 号返回数据取回到 SR21/SR22/SE-22 的 CPU 内。

ABTACC 出错停止加速度设定

命令编码：19

参 数：1~500000 （默认值：1500）

参数类型：无符号整数

单 位：au

允许执行条件：任何条件下均可执行

执行后影响：调整加速度改变

SR21/SR22/SE-22 指令：

F50 ××× （高位数字）

F60 R576

F50 ××× (低四位数字)
F33 19

该命令用于设定出错停止加速度，E-02PP 在发现错误时，要使马达停止以免发生事故，停止时使用该加速度。该加速度设得大一点，可以使出错停止过程进行得较快。

3. 2. 2 调整类命令

调整类命令可以使轴以几种方式运动，但一般不用于主要工作周期，而用于初始化（如找原点）、参数测试（如 JOG），以及定位过程的控制（如 RESTART 和 VCHG 命令）。调整类命令一接到，就立即执行，一般对于调整类命令中会使轴运动的命令，不应在执行 MOV 类命令时使用，也不能几个调整类的运动命令交错使用。如正在执行 FHOME 命令时，就不可执行 JOG 命令，调整类命令的命令码从 20 开始，在 40 以下。

FJOGP 快速正向调整命令

命令编码：20

参 数：无

允许执行条件：EXEMOV 触点断开，未执行找原点命令。

执行后影响：轴以快速调整速度正向运动。

SR21/SR22/SE-22 指令：

F30 20

该命令使轴以指定的快速调整速度正向运动。如果现在的速度和指定的速度不一样，就以调整加速度逐步变速，一直达到指定的速度，然后定速运动直至接收 FJOGP、FJOGN、SJOGP、SJOGN 中的一个命令或 JOG OFF 命令后，再改变速度，以新的速度运动。本命令可以在 ERROR 触点闭合的情况下执行，这样就可以定位目标超出限定位置时，通过调整来恢复正常的状态，但在定位目标越过限位开关的情况下，即 71、73 号错误发生时，本命令无效。

正在执行 MOV 类命令或正在找原点时，不可使用本命令。

FJOGN 快速负向调整命令

命令编码：21

参 数：无

允许执行条件：EXEMOV 触点断开，未执行找原点

执行后影响：轴以快速调整速度负向运动。

SR21/SR22/SE-22 指令：

F30 21

该命令使轴以指定的快速调整速度负向运动。如果现在的速度和指定的速度不一样，就以调整加速度逐步变速，一直达到指定的速度，然后定速运动直至接收到 FJOGP、FJOGN、SJOGP、SJOGN 中的一个命令或 JOG OFF 命令后，再改变速度，以新的速度运动。本命令可以在 ERROR 触点闭合的情况下执行，这样就可以在定位目标超出限定位置时，通过调整来恢复正常的状态，但在定位目标越过限位开关的情况下，即 72、73 号错误发生时，本命令无效。

正在执行 MOV 类命令或正在找原点时，不可使用本命令。

SJOGP 慢速正向调整命令

命令编码：22

参 数：无

允许执行条件：EXEMOV 触点断开，未执行找原点

执行后影响：轴以慢速调整速度正向运动

SR21/SR22/SE-22 指令：

F30 22

该命令使轴以指定的慢速调整速度正向运动，如果现在的速度和指定的速度不一样，就以调整加速度逐步变速，一直达到指定的速度，然后定速运动直至接收到 FJOGP、FJOGN、SJOGP、SJOGN 中的一个命令或 JOG OFF 命令后，再改变速度，以新的速度运动。本命令可以在 ERROR 触点闭合的情况下执行，这样就可以在定位目标超出限定位置时通过调整来恢复正常的状态，但在定位目标越过限位开关的情况下，即 71、73 号错误发生时，本命令无效。

正在执行 MOV 类命令或正在找原点，不可使用本命令。

SJOGN 慢速负向调整命令

命令编码：23

参 数：无

允许执行条件：EXEMOV 触点断开，未执行找原点

执行后影响：轴以慢速调整速度负向运动

SR21/SR22/SE-22 指令：

F30 23

该命令使轴以指定的慢速调整速度负向运动。如果现在的速度和指定的速度不一样，就以调整加速度逐步变速，一直达到指定的速度，然后定速运动直至接收到 FJOGP、FJOGN、SJOGP、SJOGN 中的一个命令或 JOG OFF 命令后，再改变速度，以新的速度运动。本命令可以在 ERROR 触点闭合的情况下执行，这样就可以在定位目标超出限定位置时通过调整来恢复正常的状态，但在定位目标越过限位开关的情况下，即 72、73 号错误发生时，本命令无效。

正在执行 MOV 类命令或正在找原点时，不可使用本命令。

JOGOFF 停止调整命令/带参数调整

命令编码：24

参 数：指定

参数类型：8 位二进制

允许执行条件：任何情况下均可执行。

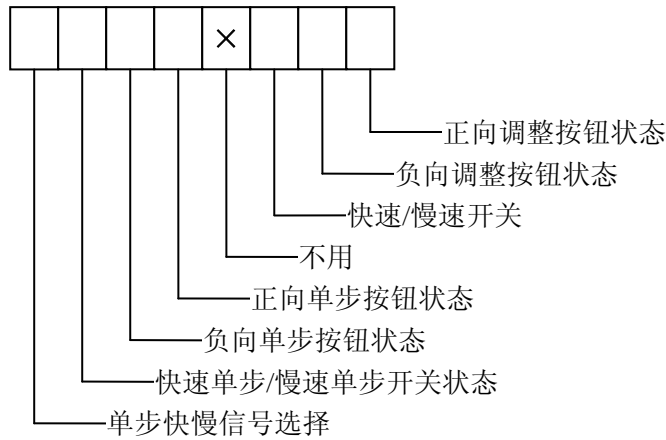
SR21/SR22/SE-22 指令：

F51 ××

F31 24

该命令的参数为 0 时，使正执行的调整运动（FJOGP、FJOGN、SJOGP、SJOGN 命令启动）停止，以调整加速度逐渐减速到 0。如果现在未进行调整，则该命令不起作用。

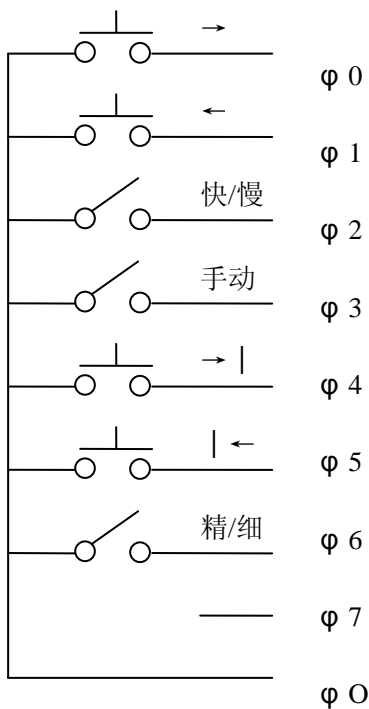
当该命令带参数时，参数格式如下：



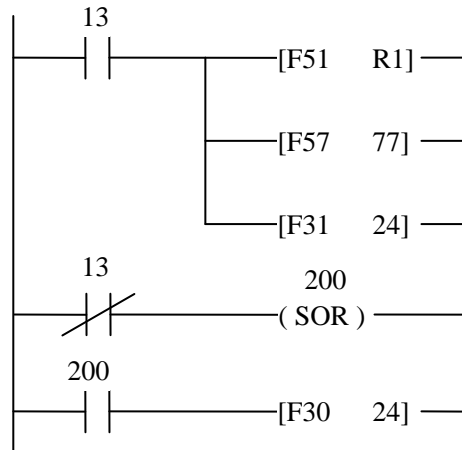
该命令的用法如下：

例：在定义号为 010 的槽上装有一个输入模块（如 E—01N 或 E—05N），并连接如下：

例 1 接线：

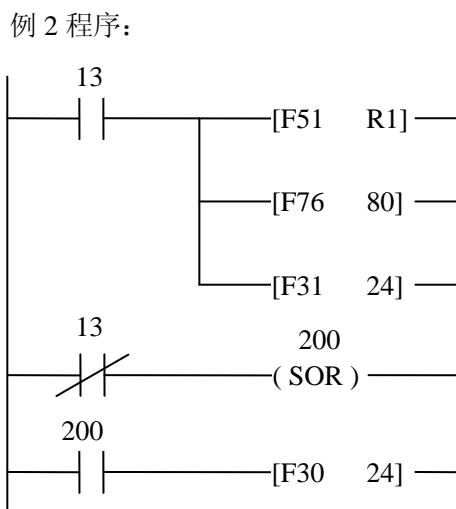
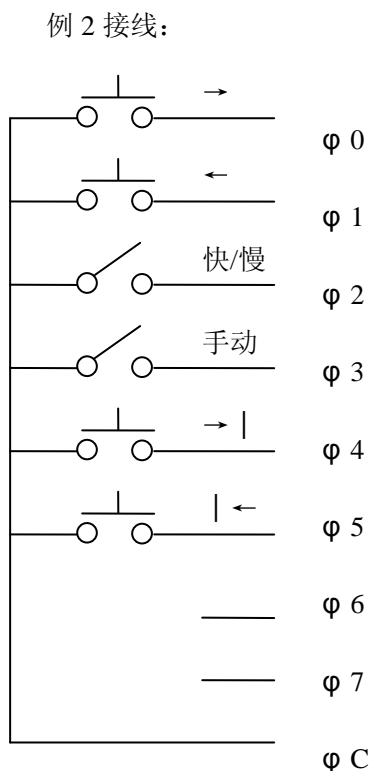


例 1 程序：



就可以完成调整单步等动作。→、←按钮按下时，分别进行调整，可用“快/慢”开关选择速度、开关按下为快速。用→|、|←按钮可以控制单步动作，用精细开关选择步长，开关按下为粗档。单步动作是新追加的，其动作是每按一次按钮就正向或负向动作规定的距离。这个距离是用 9 号命令设定的，当选择粗档时，步进的距离就扩大 10 倍。

例 2 的功能与例 1 相同，但单步的粗细与调整的快慢合用一个开关。



HOMEMOD 设置找原点动作方式

命令编码：27

参 数：0~3 （默认值：0）

参数类型：无符号整数

单 位：方式编号

允许执行条件：ERROR 触点断开

执行后影响：下次找原点时以设置的方式进行

SR21/SR22/SE-22 指令：

F50 × （找原点动作方式编号 0~3）

F31 27

HOMEMOD 命令用于设置找原点动作方式，E—02PP 支持 4 种找原点方式，如果不进行设置，E—02PP 将以方式 0 找原点，关于 4 种方式的说明见 FHOME 命令（命令码 28）的说明。

FHOME 找原点命令

命令编码：28

参 数：无

允许执行条件：ERROR 触点断开、MOVING 触点断开，未进行其他运动时

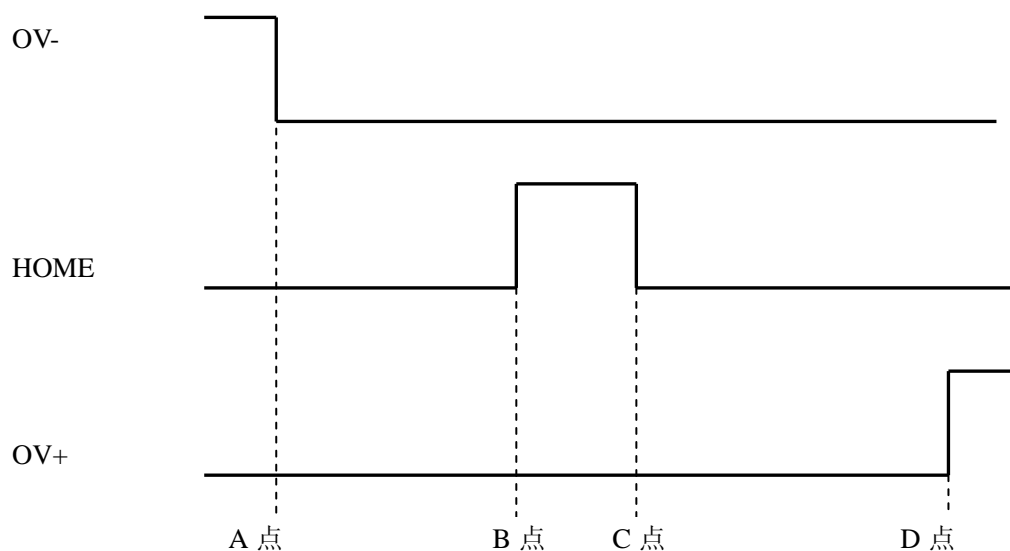
执行后影响：轴被定位到机械原点的位置

SR21/SR22/SE-22 指令：

F30 28

找原点，是为了使定位目标运动到定位系统的一个固定位置，使 E—02PP 模块内部记忆的坐标值与机械位置建立对应关系。所谓机械原点，即坐标为 0 的点；所谓电气原点，就是发出电气原点信号的点。找原点即找机械原点时要使用电气原点信号，电气原点和 0 坐标点可以不同，电气原点的坐标用 PHOME 命令进行设定。

找原点的动作，需要由接近开关确定电气原点，接近开关的安装要求如 1.5 节所述，用接近开关可以确定轴动作范围内的 A、B、C、D4 个点，如下图所示：



上述开关在 ON→OFF 或 OFF→ON 时触点抖动应在 30mS 内消失，超过此限时会影响找原点动作。

找原点时用到检索速度，其取值原则为：若慢速调整速度小于 128vu，则取慢速调整速度，不然则取 128vu。

找原点的 4 种动作方式说明如下：

方式 0:

该方式动作由 5 步构成，但因动作条件不同，有时可以减少步数。在该方式下，不接 OV+开关也可。

1) 如果 HOME 开关为 ON，直接进行第 3) 步；如果 HOME 开关为 OFF，以快速调整速度作负向运动，直至：HOME 开关变为 ON，即到达 C 点则进行第 3) 步；或 OV-开关变为 ON，即到达 A 点则减速停止后进行第 2) 步。

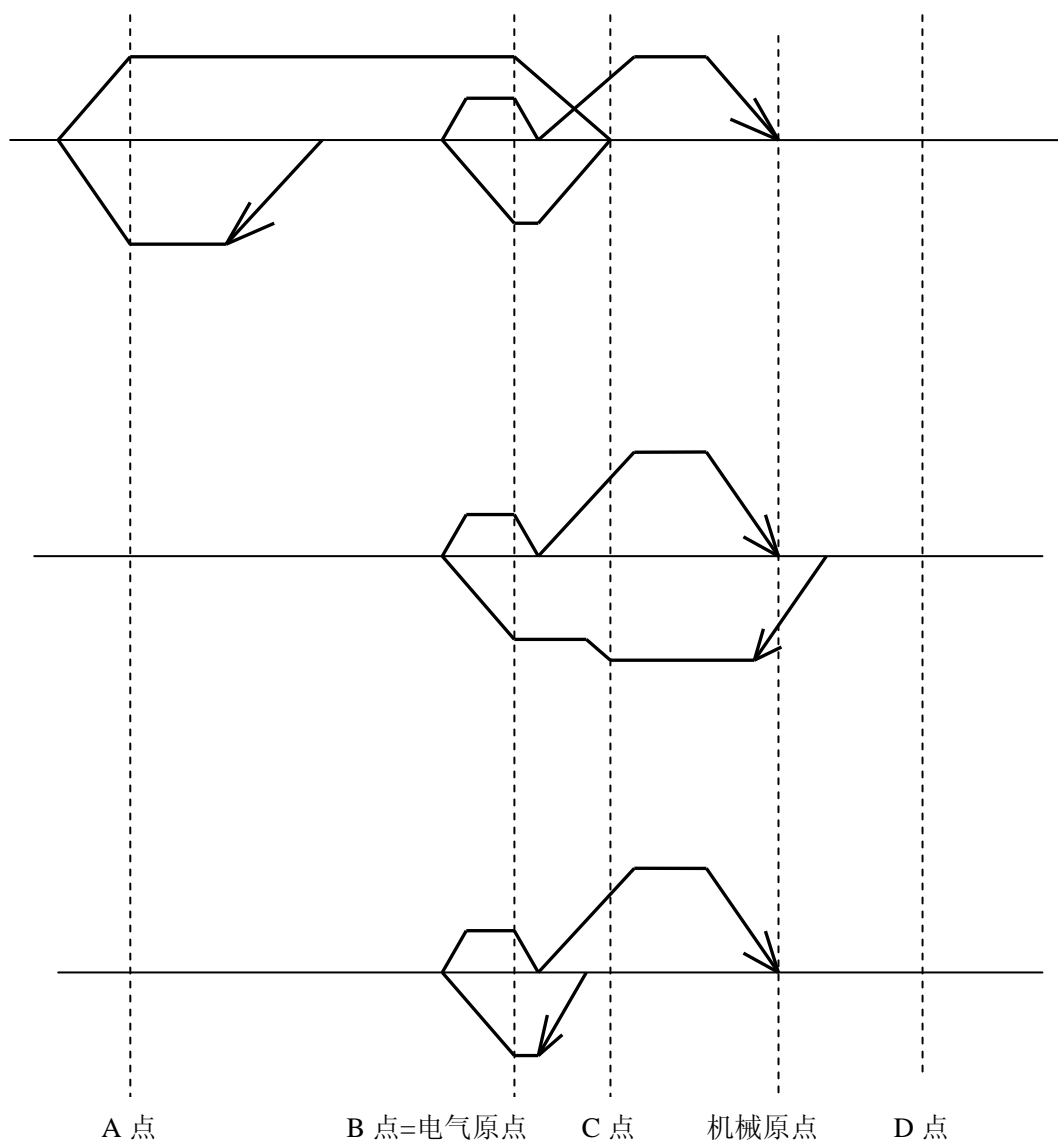
2) 以快速调整速度作正向运动，直至 HOME 开关变为 ON 即到达 B 点后减速停止。

3) 以慢速调整速度作负向运动，直至 HOME 开关由 ON 变为 OFF 即到达 B 点后减速停止。

4) 以检索速度作正向运动，在检测到 HOME 开关 OFF→ON 的变化点即 B 点时，记录当时的位置值作为电气原点，并减速停止，然后对模块内部的位置坐标进行修正。

5) 以快速调整速度定位到坐标为 0 的位置即机械原点。

以上动作图示如下：



方式 1:

该方式和方式 1 类似，但开始动作的方向不同，在该方式下，不接 OV-开关也可。

1) 如果 HOME 开关为 ON，直接进行第 3) 步；如果 HOME 开关为 OFF，以快速调整速度作正向运动，直至：HOME 开关变为 ON，即到达 B 点，则减速停止后进行第 3) 步；或 OV+开关变为 ON，即到达 D 点，则减速停止后进行第 2) 步。

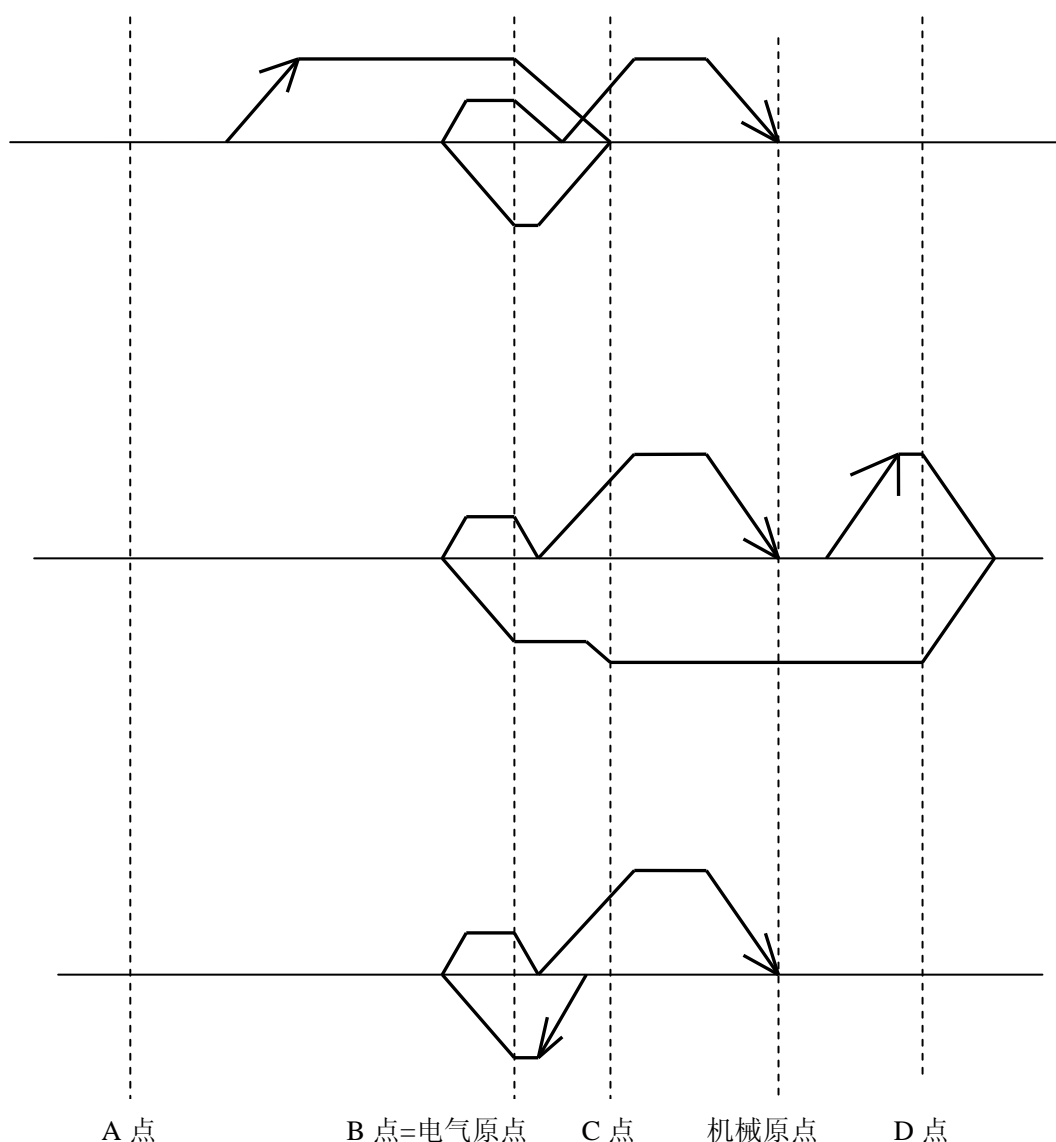
2) 以快速调整速度作负向运动，直至 HOME 开关变为 ON 即到达 C 点后进行第 3) 步。

3) 以慢速调整速度作负向运动，直至 HOME 开关由 ON 变为 OFF 即到达 B 点后减速停止。

4) 以检索速度作正向运动，在检测到 HOME 开关 OFF→ON 的变化点即 B 点时，记录当时的位置值作为电气原点，并减速停止，然后对模块内部的位置坐标进行修正。

5) 以快速调整速度定位到坐标为 0 的位置即机械原点。

以上动作图示如下：

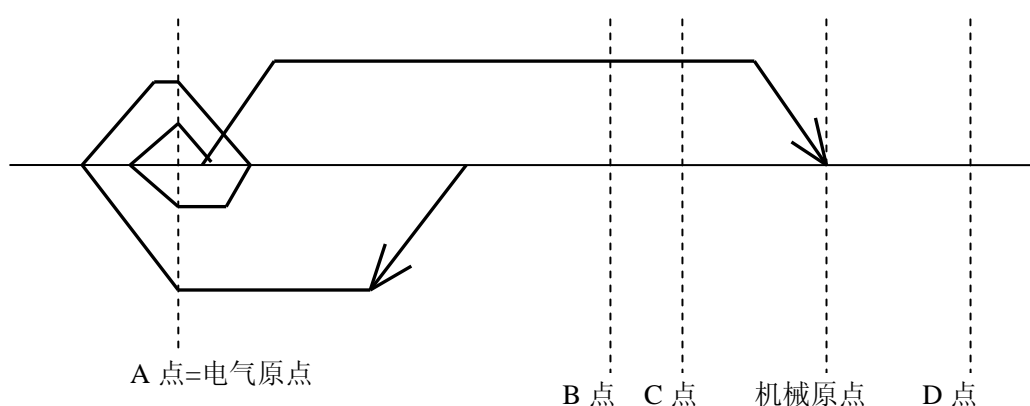


方式 2:

该方式动作由 5 步构成，在该方式下，不接 OV+和 HOME 开关也可。注意：在此方式下，电气原点坐标值应为负值。

- 1) 以快速调整速度作负向运动，直至 OV-开关变为 ON，即到达 A 点后减速停止。
- 2) 以慢速调整速度作正向运动，直至 OV-开关由 ON 变 OFF，即再次到达 A 点后减速停止。
- 3) 以检索速度作负向运动，直至 OV-开关 OFF→ON 的变化点即 A 点时，记录当时的位置值作为电气原点，并减速停止，然后对模块内部的位置坐标值进行修正。
- 4) 以慢速调整速度作正向运动，直至 OV-开关由 ON 变为 OFF，即再次到达 A 点后减速停止。
- 5) 以快速调整速度定位到坐标为 0 的位置即机械原点。

以上动作图如下:

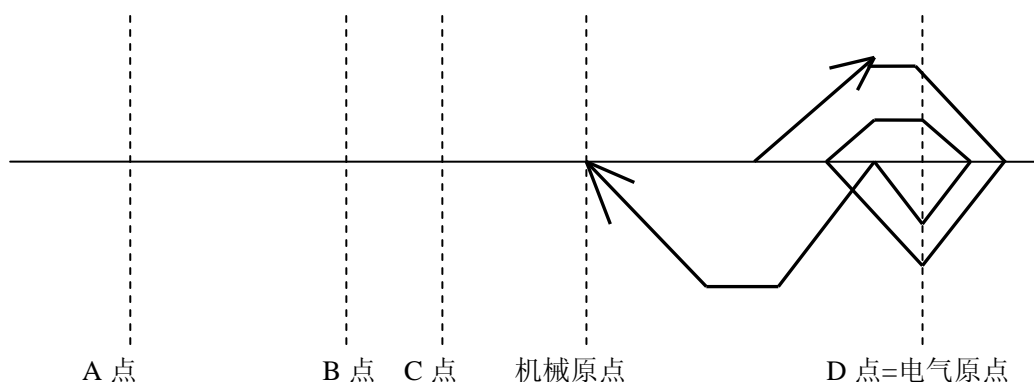


方式 3:

该方式和方式 3 类似，但开始动作的方向和检测的原点信号不同，在该方式下，不接 OV-和 HOME 开关也可，注意在此方式下，电气原点坐标值应为正值。

- 1) 以快速调整速度作正向运动，直至 OV+开关变为 ON 即到达 D 点后减速停止。
- 2) 以慢速调整速度作负向运动，直至 OV+开关由 ON 变为 OFF 即再次到达 D 点后减速停止。
- 3) 以检索速度作正向运动，直至 OV+开关 OFF→ON 的变化点即 D 点时，记录当时的位置值作为电气原点，并减速停止，然后对模块内部的位置坐标值进行修正。
- 4) 以慢速调整速度作负向运动，直至 OV+开关由 ON 变为 OFF 即再次到达 D 点后，减速停止。
- 5) 以快速调整速度定位到坐标为 0 的位置即机械原点。

以上动作图示如下:



RESTART 等待结束命令

命令编码：29

参 数：无

允许执行条件：无限制

执行后影响：正在执行的等待命令结束

SR21/SR22/SE-22 指令：

F30 29

本命令用于结束 MOV 类命令中的 WAIT（命令号 183）命令的执行，如果模块当前并未执行 WAIT 命令，则该命令不起作用。正在执行的 WAIT 命令被结束之后，就开始继续执行 WAIT 命令后面的 MOV 类命令。请参阅 WAIT 命令（命令码 183）的说明。

ABORT 中止运动

命令编码：30

参 数：无

允许执行条件：无限制

执行后影响：正在执行的运动被中止，使轴以调整加速度减速到 0

SR21/SR22/SE-22 指令：

F30 30

本命令可用于任何运动的中止，任何一类运动不想再继续进行，都可以用本命令中止，一旦执行 ABORT 命令，轴就以调整加速度到 0；同时，所有已接收到但未执行的命令都被取消。ABORT 命令不但能中止使轴实际运动的命令，例如 JOG、FHOME 等，对于 DELAY 和 WAIT 这样并不引起轴实际运动的命令，同样可以中止，但 ABORT 命令不取消已定义的子程序。

CLRERR 清除错误命令

命令编码：31

参 数：无

允许执行条件：无限制

执行命令后影响：错误码被清除，ERROR 触点断开

SR21/SR22/SE-22 指令：

F30 31

该命令用于模块报告错误信息时，在确认错误以后清除错误码，使模块可以接收并执行在 ERROR 触点闭合时不允许执行的命令，以便调整轴的状态，为恢复正常进行建立条件。但希望用户不要用这个命令清除错误码以后，继续进行正常工作，而应该在查清错误码发生的原因，排除故障以后重新开始新的工作周期。因为错误码产生以后，模块内部有很多状态发生了变化，许多参数都不再是用户所希望的值，再继续工作下去，可能会发生不正常的情况，所以应在用本命令清除错误码以后，只进行排除故障所必需的工作，请参阅“错误发生过程”的说明。

VCHG 变速命令

命令编码：33

参 数：无

允许执行条件：无限制

执行后影响：正在执行的 MVPMOV 运动改变速度

SR21/SR22/SE-22 指令：

F30 33

本命令只在轴目前正在按 MVPMOV 命令运动时起作用，MVPMOV 的运动采用的速度是速度表中的 V0, V1, ……Vn，一般先从 0 加速到 V0，然后定速运动；如果在此期间没有 VCHG 命令，则一直以 V0 运动，直到接近目标位置时，减速到 Vn，最后以 Vn 运动规定的保留距离，然后减速到 0，定位到目标位置。如果在定速运动阶段，接收到一个 VCHG 命令，运动的速度就以规定的加速度改变一次，如正以 V0 运动时，就变到 V1；以 V1 运动时，就变到 V2；以此类推，这样的过程一直可以进行到 Vn，以后再也不允许变速，一直到最后运动到目标位置。所以，通过 VCHG 命令可以根据外界的条件变化，使轴的运动速度随时变小，请参阅 MVPMOV 命令的说明。如果模块的输入点 IN1~IN4 中的参数有被指定为 1 时，那么这个输入点接通一次，则相当于执行一次 VCHG 指令。

PARSTOP 局部中止命令

命令编码：34

参 数：无

SR21/SR22/SE-22 指令：

F30 34

本命令对已定义的局部可中止程序进行中止(参看 186 命令的说明)。如果模块的输入点 IN1~IN4 中的参数有被指定为 5 时，那么这个输入点接通一次，则相当于执行一次 PARSTOP 指令。

3. 2. 3 子程序类和 MOV 类命令

以上说明的命令属于参数设定类、调整类，以下对子程序定义类和定位控制类（简称 MOV 类）命令进行说明，这两类命令有许多共同点，为了有效地利用这些命令，对这两类命令的共同点进行说明。

MOV 类命令的执行不象前两类命令接收到命令后就立即执行，而是先存入缓冲区再依次执行。模块内部有一个可以存放 256 条 MOV 类命令的缓冲区和可以存放 512 条命令的子程序缓冲区，MOV 类命令按输入的次序一个接一个地执行，前一个完成以后才继续执行下一条命令，执行完毕的命令在 MOV 类命令缓冲区中去除。对于所定义的程序中包含的 MOV 类指令先存入子程序缓冲区，只有以后调用这个子程序时，该命令才会被执行。

某些 MOV 类命令的执行将引起步号加 1，这些命令是 PMOV、FMOV、MVPMOV、DELAY、WAIT。同时，这些命令又是需要花较长时间才能执行完毕的命令，在执行这些命令的时候，模块上的 EXEMOV 触点是闭合的。一般来说，如果送给模块的所有 MOV 类命令都执行完以后，没有新 MOV 类命令送到模块时，EXEMOV 才会断开。

MOV 类命令的执行条件：

MOV 类命令是指命令编码在 100~199 范围内的所有命令，E-02PP 模块满足下列条件时才可以接收这类命令：

1. ERROR 触点断开。
2. 模块未执行调整类运动命令（如 JOG、FHOME 等）。
3. MEMFULL 触点断开（如果该触点闭合以后再传送 10 条以上命令给 E-02PP 模块的话，就会发生 13 号错误。

定位命令参数指定方法：

PMOV、FMOV、MVPMOV 这三种定位命令都需要指定一个位置参数，这个参数用于指明轴应通过运动到达的新的位置。

位置参数指定的方法有三种：

1) 绝对指定法

采用绝对指定法时，上述三种命令表示为 PMOV9、FMOV9、MVPMOV9，这时命令带有参数，参数的范围为-8388608~+8388607，参数所表示的含义是使轴运动到由参数所指定的新坐标。

例如，开始时轴处于坐标为 10000 的位置，执行 PMOV9 13000 时，轴就向正方向运动移过距离 3000 以后定位到坐标为 13000 的位置。如果开始时轴处于坐标为 20000 的位置而执行 PMOV9 13000 时，轴就向负方向运动移过距离 7000 以后定位到坐标为 13000 的位置。如果开始时轴正好处于 13000 的位置，执行 PMOV9 13000 时，轴就不进行任何运动。

应注意，采用绝对指定法时，参数值和轴的初始位置值之差即命令所需要的位移量的范围，应在-8388608~+8388607 之间。

用绝对指定法时 SR21/SR22/SE-22 指令：

F50 ××××（高四位数字）

F60 R576

F50 ××××（低四位数字）

F33 ××× （命令码）

2) 相对指定法:

采用相对指定法时,上述三种命令表示为 **PMOV8**、**FMOV8**、**MVPMOV8**,这时命令带有参数,参数的范围为-8388608~+8388607,参数所表示的含义是从轴开始执行命令的当前位置起,移动指定的距离,参数为负时,表示向负方向移动。

例如,开始时轴处于坐标为 10000 的位置,执行 **PMOV8 3000** 后,轴就应定位到坐标为 13000 的位置。反之,执行 **PMOV8 3000** 后,轴应定位到 7000 的位置。

应注意轴的现在位置值加上位移值以后得出的新坐标值,不应超过-8388608~+8388607 的范围。用相对指定法时 **SR21/SR22/SE-22** 指令:

```

F50  ×××× (高四位数字)
F60  R576
F50  ×××× (低四位数字)
F33  ×××   (命令码)

```

3) 间接指定法:

这种方法在命令中并不出现参数,而是在命令码中规定新位置的坐标值存放的寄存器号。E-02PP 模块内有 8 个位置寄存器用于存放位置值,这 8 个寄存器的内容可以用三组命令 **DPTABi**, **PTABDDi**, **PTABSUBi** 来进行修改。其中 **DPTABi** 命令为第 i 号位置寄存器一个坐标值, i 可以从 0 到 7。值的范围应在-8388608~+8388607 之间。**PTABDDi** 和 **PTABSUBi** 是在第 i 号位置寄存器的内容上加上或减去一个 0000~9999 的数。一般地用 **DPTABi** 为第 i 号位置寄存器置数,用 **PTABDDi** 和 **PTABSUBi** 修正设定的值。

采用间接指定法时, **PMOVi** 的含义就是在 **PMOV** 时运动到第 i 号位置寄存器所规定的位置。即寄存器所表示的值是绝对表示方式,位置寄存器中的值,也和绝对指定法一样要注意范围。对 **FMOV**、**MVPMOV** 命令也和 **PMOV** 命令相同的原则使用这 8 个位置寄存器。

用间接指定法时 **SR21/SR22/SE-22** 指令:

```

F30  ×××   (命令码)

```

子程序定义类命令的命令码在 89~99 之间, **MOV** 类命令的命令码在 100~199 之间。

SRBRi 子程序定义命令**RET** 子程序结束命令

命令编码: **SRBR0~SRBR9** 90~99

```

RET 89

```

参 数: 无

允许执行条件: **ERROR** 触点断开, 未执行其他运动命令。

SR21/SR22/SE-22 指令:

```

SRBR0~SUBR9      F30  90~99
RET                F30  89

```

子程序定义和子程序结束这一组命令用于为 E-02PP 模块定义子程序。子程序是一系列 **MOV** 类命令组成的一个控制过程,这个控制过程一旦定义好以后,就可以象使用一条命令一样来执行这一过程。对于重复进行一系列规定动作的定位系统来说,预先将一些重复进行的规定动作作为子程序定义好以后,实际需要执行时,只要在 **SR21/SR22/SE-22** 之间进行很少的信息传输,就可以完成比较复杂的控制。

在命令的输送过程中，送了一条 **SRBRi** ($i=0\cdots\cdots 9$) 以后，一直到送 **RET** 命令之前的 **MOV** 类命令都放在相应的子程序存储器中，这些命令并不马上执行，只有以后输送一个 **CALL** 命令时，就把这些命令依次执行一遍。在 **SUBRi** 和 **RET** 命令之间，可以送 **MOV** 类命令以外的命令到 E-02PP 模块，但这些命令并不被存放到子程序存储器中，而是立即执行。所以建议在 **SR21/SR22/SE-22** 上编程时，**SUBRi** 和 **RET** 命令之间，子程序的内容一次或几次集中传送以免发生混乱或错误。

用 **SUBR0~SUBR9**，一共可以定义 10 个子程序，每个子程序的大小无具体限制；但是，10 个子程序内最多只能包含 512 条 **MOV** 类命令，如果超过这个限制，将会发生 57 号错误。

子程序的定义必须按 **SUBR0→SUBR9** 的次序进行。定义的时候，一旦定义 **SUBR0**，那么，**SUBR1~SUBR9** 就会全部被清除，以次类推。编号 i 的子程序定义的时候，编号 $i+1\sim 9$ 的子程序全部被清除。

在子程序内部可以包含调用其他子程序的命令，但是不可以调用自身。例如，**SUBR0** 内不允许有 **CALL0** 命令，也不允许调用其他包含有 **CALL0** 的程序，即不允许递归或间接递归调用。如果从一个子程序执行开始，到最后要调用到自身的话，就会产生 60 号错误。此外，在子程序内部也不可以定义别的子程序，其实这样做也没有必要。一旦出现这样的情况，就会产生 62 号错误。反之如果一个 **RET** 命令没有 **SUBRi** 命令与之对应的話，也会产生 62 号错误。

PMOV 精确定位命令

命令编码：100~109

参 数：参阅“定位命令参数指定方法”

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

参阅“定位命令参数指定方法”

PMOV 是精确定位的命令，在执行 **PMOV** 时，先以运行加速度 **MOVACC** 变速到运行速度，然后定速运动；到一定时候开始减速，一直减速到 0，最后正好使轴的位置停止在定位目标带之内，本命令才算执行完毕。由于理论速度与实际速度的不一致，轴停在定位目标带以内的时间要比理论值滞后一点。定位完成后，若还有其他命令需要执行的话，就开始执行下一命令。

FMOV 多段运动命令

命令编码：120~129

参 数：参阅“定位命令参数指定方法”

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

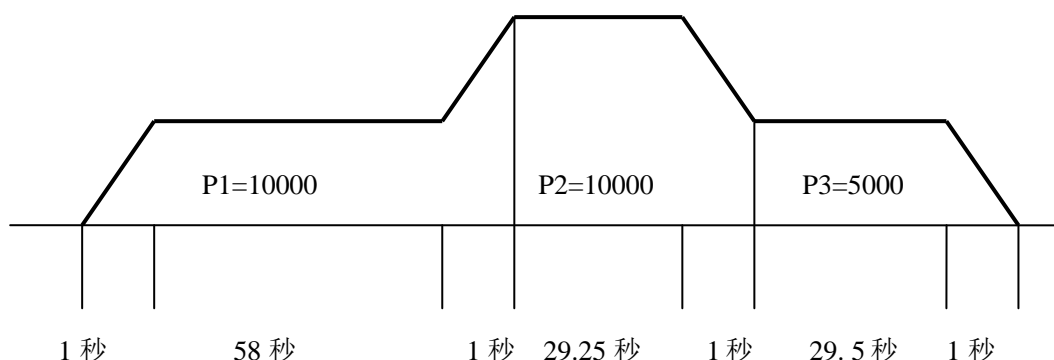
参阅“定位命令参数指定方法”

FMOV 命令和 **PMOV** 不一样，**FMOV** 指定一个中间过程的位置，轴在经过所定义的位置时速度不为零。但在理论上，轴经过该点时速度正好达到下一条 **FMOV** 或 **PMOV** 命令所要求的速度。**FMOV** 命令后面必须跟 **PMOV** 或别的 **FMOV** 命令，但最后必须以 **PMOV** 命令结束。只有接收到下一个 **FMOV** 或 **PMOV** 命令时，以前的 **FMOV** 命令才会开始执行。

举例：

MOVACC	10
MOVVEL	100
FMOV8	10000
MOVVEL	200
FMOV8	10000
MOVVEL	100
PMOV8	5000

对应的理论时间速度图：



FVPMOV 可变速精确定位命令

命令编码：130~139

参 数：参阅“定位命令参数指定方法”

允许执行条件：参阅“MOV类命令执行条件”

SR21/SR22/SE-22 指令：

参阅“定位命令参数指定方法”

本命令的运动过程中，可以在接收到变速命令时改变运动速度。该命令运行时所采用的速度不是 MOVVEL，而是 VTAB 中的速度。典型的执行过程是先从 0 加速到 V_0 ，然后定速运动，在定速过程中可以接收变速命令。接收到变速命令后，速度就变为 V_1 ，然后以 V_1 定速运动，此时又可接收变速命令。这个过程可以一直重复到速度变为 V_n ，最后定位到目标位置。

如果在定速运动阶段没有接收到变速命令，则一直到离目标位置的一定距离就开始减速，保证减速到 0 时定位到目标。

与本命令有关的命令还有 MVFP 为本命令定义一个参数，这个参数是一个距离值，称为保留距离。它规定 MVOMOV 命令执行过程中，在到达目标位置附近的保留距离内，速度不得超过 V_n ，这种情况下，MVPMOV 的运动过程中，即使没有接收到变速命令，也会在一定的時候减速，保证以不超过 V_n 的速度进行一定的距离，最后定位到目标位置。

本命令执行时，要检查速度表的内容，如果速度表的速度不按 $V_0 > V_1 > \dots > V_n$ 的递降规律排列的话，就会产生 67 号错误。

速度表中速度个数即 n 值以下列原则确定，速度表中为 0 的速度之前的速度为 V_n ，但 n 应 ≥ 1 。

例如： $V_0=10000$ ， $V_1=3000$ ， $V_2=500$ ， $V_3=0JF$ ，不论 $V_4\sim V_7$ 为何值， n 就等于 2，此时 MVPMOV 命令只能变速二次，整个运行过程中采用三种速度。如果 $V_0\sim V_7$ 全不为 0，则 $n=7$ 。

如果本命令的移动距离比较短，尚未达到 V_0 就应开始减速，则不能再接收变速命令。这时执行命令的过程中先从 0 加速到某一速度就开始减速，但在保留距离内的速度不超过 V_n 这一点仍能得到保证。

MVFP 可变速运动保留距离设定

命令编码：148

参 数：0~9999 （默认值：0）

参数类型：无符号整数。

单 位：CT

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

F50 $\times\times\times\times$ （保留距离）

F32 148

本命令用于设定可变速运动 MVPMOV 命令中定速运动的距离，设定这个距离之后，MVPMOV 命令执行时以速度表的最后一个速度至少要运动这段规定的距离，这个值可以设为 0。

DPTABi 位置寄存器设定 ($i=0\sim 7$)

命令编码：140~147

参 数：-8388608~+8388607 （默认值：无）

参数类型：带符号整数

单 位：CT

允许执行条件：参阅“MOV 类命令执行条件”。

SR21/SR22/SE-22 指令：

F50 $\times\times\times\times$ （高四位数字）

F60 R576

F50 $\times\times\times\times$ （低四位数字）

F33 140 (~147)

本命令为 i 号位置寄存设定一个值，该值的用法请参阅“定位命令的参数指定方法”。

PTABADDi 位置寄存器值增加 ($i=0\sim 7$)

命令编码：150~157

参 数：0~9999

参数类型：无符号整数

单 位：CT

允许执行条件：参阅“MOV 类命令执行条件”。

SR21/SR22/SE-22 指令:

F50 ×××× (位置寄存器递增)
F32 150 (~157)

本命令在 i 号寄存器原来值的基础上加上一个值。位置寄存器中值的用法请参阅“定位命令的参数指定方法”。

PTABSUBi 位置寄存器值减 (i=0~7)

命令编码: 160~167
参 数: 0~9999
参数类型: 无符号整数
单 位: CT
允许执行条件: 参阅“MOV 类命令执行条件”。

SR21/SR22/SE-22 指令:

F50 ×××× (位置寄存器递减)
F32 160 (~167)

本命令在 i 号寄存器原来值的基础上减去一个值。位置寄存器中值的用法请参阅“定位命令的参数指定方法”。

DVTABi 速度表内容设定命令= (i=0~7)

命令编码: 170~177
参 数: 0~最大允许值 (默认值: V0=1500, V1=500, V2=0, V3~V7 无定义)
参数类型: 无符号整数
单 位: vu
允许执行条件: 参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令:

F50 ×××× (高位数字)
F60 R576
F50 ×××× (低四位数字)
F33 170 (~177)

本命令用于设定速度表中的 V0~V7, V0~V7 是为 MVPMOV 命令规定运动速度变化规律用的。MVPMOV 命令最多允许 8 个速度, 可以接收 7 次变速命令, 也可以用少于 8 个速度。如果希望定义少于 8 个速度时, 只需在最后一个速度之后定义下一个速度值为 0。例如希望定义少于 8 个速度时, 只需在最后一个速度之后定义下一个速度值为 0。例如希望定义四个速度时, 在设定 V0, V1, V2, V3 以后应定义 V4 为 0。这时把 V3 称为 Vn。

注意: V0 和 V1 不允许定义为 0, 否则将会产生 64 号错误。

如果定义的速度不满足 V0>V1>……>Vn, 则在执行 MVPMOV 命令时将会产生 67 号错误。在执行 ABORT 命令以后, 速度表中内容又变为默认值。

BASVEL 基础速度设定命令

命令编码：178

参 数：0~60000 （默认值：0）

参数类型：无符号整数

单 位：vu

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

F50 ×× （高位数字）

F60 R576

F50 ×××× （低四位数字）

F33 178

本命令用于设定基础速度。所谓基础速度是指马达在此速度以下的范围内改变速度时，不必以规定的加速度变速而可以直接跳变速度。如果马达确实具有这种性能的话，选择适当的基础速度可以缩短加速或减速时间。

MOVVEL 运动速度设定命令

命令编码：179

参 数：1~60000 （默认值：512）

参数类型：无符号整数

单 位：vu

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

F50 ×× （高位数字）

F60 R576

F50 ×××× （低四位数字）

F33 179

本命令为以后 PMOV、FMOV 设定运行速度，在此以前的上述命令的运行速度不受影响。注意：这个速度不可为 0，也不可设得大于由 VLMT 设定的速度极限。

MOVACC 运动加速度设定命令

命令编码：180

参 数：1~500000 （默认值：500）

参数类型：无符号整数

单 位：au

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

F50 ×× （高位数字）

F60 R576

F50 ×××× （低四位数字）

F33 180

本命令为以后 PMOV、FMOV 设定运行加速度，在此以前的上述命令的运行速度不受影响。注意：这个速度不可为 0，也不可设得大于 500000。

DELAY 延时命令

命令编码：181

参 数：0~99999

参数类型：无符号整数

单 位：10ms

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

F50	×
F60	R576
F50	××× (参数值)
F33	181

执行本命令时，必须是轴的运动速度已为 0（即不是在执行 FMOV 命令之后），本命令使轴保持不动，延迟规定的时间。例如选择参数为 200 时，DELAY200 命令使用轴保持不动，过 2 秒钟，然后继续执行以后的命令（如果还有命令的话）。本命令虽然不引起轴的运动，但也可以象 PMOV 等命令那样用 ABORT 命令来中止它。可以把 DELAY 命令看作是速度等于 0 移动距离等于 0 的运动命令。

STEP 设定步号

命令编码：182

参 数：00~99

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

F50	×× (步号值)
F31	182

执行本命令将使步号设定为所希望的值。步号是在运动过程中各个工序的标识号，在初始状态下，步号是 0；以后在执行 PMOV、FMOV、MVPMOV、DELAY、WAIT 命令之前步号加 1，如果步号到 99 再加工，又重新变为 0。一般对重复循环进行的定位系统来说，在每次循环开始之前，先将步号设为一个固定数字，则以后就可以根据步号判断定位系统目前正在进行那一道工序。SR21/SR22/SE-22 可以根据从模块返回的数据中的步号进行工序管理，利用其他 I/O 模块对各工序进行控制。

WAIT 等待命令

命令编码：183

参 数：无

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

F30	183
-----	-----

本命令与 DELAY 命令类似，使轴处于不动的状态下。唯一的区别在于命令的结束条件不同，

DELAY 命令到了规定的时间就结束，而 WAIT 命令只有在接收到 RESTART 命令之后才结束。

PARBGN 局部可中止开始

命令编码：185

参 数：无

SR21/SR22/SE-22 指令：

F30 185

本命令是定义程序中的局部命令可中止开始，和 186 命令一起使用，详细用法参看 186 命令。

PAREND 局部可中止结束

命令编码：186

参 数：无

SR21/SR22/SE-22 指令：

F30 186

本命令是定义程序中的局部命令可中止结束，和 185 命令一起使用。这两个命令是使 MOV 类命令的执行顺序发生改变。例如：

185 命令	当 MOV 类命令正常执行时，执行的次序是
MOV 命令 1	命令 1、命令 2……依次执行；但程序段
MOV 命令 2	用 185、186 命令定义以后，在程序正常的
。	执行中，如果执行了 34 号命令，那么此时
。	模块会以最快的速度中止正在执行的程序，
。	开始执行 186 命令以后的程序 MOV 命令 A。
MOV 命令 n	如果 186 命令的后面已没有命令，那么模块
186 命令	就停止运动。
MOV 命令 A	
。	
。	
。	

CALLi 调用子程序命令

命令编码：190~199

参 数：0~9999

参数类型：无符号整数

允许执行条件：参阅“MOV 类命令执行条件”

SR21/SR22/SE-22 指令：

F50 ×××× （重复调用次数）

F32 190 (~199)

本命令是调用以前定义好的子程序的命令。调用子程序的效果相当于子程序内的命令被重新传送一遍，也就是说，定义在子程序内的所有命令被重新执行一遍。使用子程序的优点是编程过程可以简化，而且 SR21/SR22/SE-22 与模块之间的命令传送次数可以大幅度地减少，有利于提高 PLC 的扫描速度。对于重复同一动作的循环作业定位系统来说，可以将循环动作定义成几个子程序，调用

预先定义好的子程序进行工作。子程序就象用户自己为模块定义好的新命令一样，可用来完成各种各样复杂的功能。

CALL 命令所带的参数是子程序的重复次数，例如 CALL0 100 就相当于调用 0 号子程序 100 次。如果把重复次数设为 0，模块就把它看作 1。

CONTON 接通应答信号

命令编码：200

参数：00~08

SR21/SR22/SE-22 指令：

F50 ××

F31 200

本命令使返回数据第二部分的低八位（bit）返回信号接通，如果参数是 00~07，则接通相应的 1 位（bit）；如果参数是 08，则 8 位（bit）信号全部接通。例如：假设返回数据放在 R70 中，执行了 F50 2、F31 200 后，则触点 702 接通；执行了 F50 8、F31 200 后，则触点 700~707 全部接通。

CONTOFF 断开应答信号

命令编码：201

参 数：00~08

SR21/SR22/SE-22 指令：

F50 ××

F31 201

本命令使返回数据第二部分的低八位（bit）返回信号断开，如果参数是 00~07，则断开相应的 1 位（bit）；如果参数是 08，则 8 位（bit）信号全部断开。例如：假设返回数据放在 R70 中，执行了 F50 2、F31 201 后，则触点 702 断开；执行了 F50 8、F31 200 后，则触点 700~707 全部断开。

CONTSCAN 使应答信号产生一个单脉冲

命令编码：202

参 数：00~08

SR21/SR22/SE-22 指令：

F50 ××

F31 202

本命令使返回数据第二部分的低八位（bit）返回信号产生一个扫描周期的单脉冲，如果参数是 00~07，则在相应的 1 位（bit）上产生一个单脉冲；如果参数是 08，则 8 位（bit）信号全部产生一个单脉冲。例如：假设返回数据放在 R70 中，执行了 F50 2、F31 202 后，则触点 702 上产生了一个单脉冲，执行了 F50 8、F31 200 后，则触点 700~707 全部产生了一个单脉冲。

3. 2. 4 第二轴控制用命令

第二轴（以下亦称 Y 轴）只能进行一些简单的定位、定速运动，而且运动中速度的改变为直跳式，所以一般应在马达的突跳频率之内进行动作。命令 40~45 用于第二轴控制。命令 01 用于 Y 轴的输出脉冲方式进行选择。

40~45 号命令的编程应遵循下列原则：

- ① 40~42 号命令应在 43~45 号命令前执行。
- ② 根据需要用 43 或 44 号命令启动 Y 轴。
- ③ 45 号命令在 44 号命令执行中根据需要执行。
- ④ 当 Y 轴启动以后，在规定的动作完成以前，40~44 号命令无效。
- ⑤ Y 轴正在运动中，模块的 号状态触点为接通状态。
- ⑥ Y 轴的命令与定位控制类命令的性质不同，所以不能用于 185~186，200~202 号命令。

SECONDV Y 轴动作速度设定

命令编码：40

参 数：36~60000vu （默认值为 36）

SR21/SR22/SE-22 指令：

```
F50  ×
F60  R576
F50  ××××
F33  40
```

本命令用于设定 Y 轴的运动速度值。

SECONDP Y 轴运动距离设定

命令编码：41

参 数：0~65535CT （默认值为 1）

SR21/SR22/SE-22 指令：

```
F50  ×
F60  R576
F50  ××××
F33  41
```

SECONDD Y 轴运动方向的设定

命令编码：42

参 数：0：正向，1：负向（默认值为 0）

SR21/SR22/SE-22 指令：

```
F50  ×
F31  42
```

本命令用于 Y 轴动作方向的设定。

SECPOS Y 轴定位动作启动

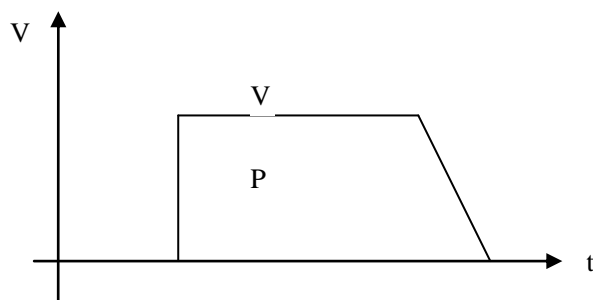
命令编码：43

参 数：无

SR21/SR22/SE-22 指令：

F30 40

本命令使 Y 轴以设定的速度 V 开始运动，移动设定的距离 P 后停止。速度时间图如下：

**SECVEL Y 轴定速启动**

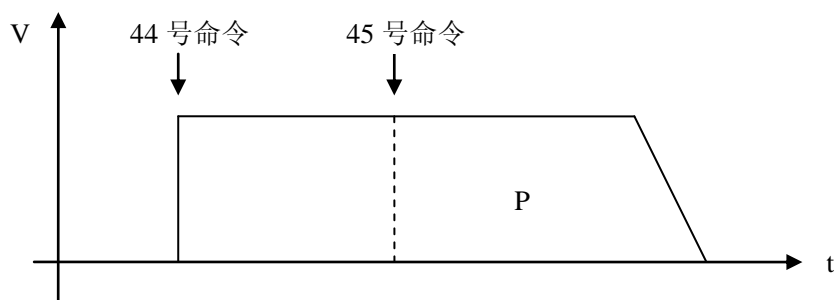
命令编码：44

参 数：无

SR21/SR22/SE-22 指令：

F30 44

本命令执行后，Y 轴以设定的速度 V 开始运动，直到接收到 45 号命令以后，开始计算位置，直到 P 以后停止。速度时间图如下：

**SEC CONV Y 轴定速转定位控制**

命令编码：45

参 数：无

SR21/SR22/SE-22 指令：

F30 45

本命令执行后，使正在作定速运动的轴再运动所设定的距离 P 后停止（参看上面的速度时间图）。作为一种特例，当 P 设为零时，一旦接收到 45 号命令，Y 轴就立即停止运动。

3. 3 错误的发生和处理

E-02PP 模块在运转的过程中，一旦检测到外界的错误条件，例如 OVER 开关接通或接收到的命令不符合要求时，就会进入出错状态。

进入出错状态时，则以出错停止加速度减速后停止。

1. 如果轴正在转动，则以出错停止加速度减速后停止。
2. 取消 MOV 类命令缓冲区中尚未执行的命令，但已定义的子程序不取消。
3. 返回数据中的 ERROR 触点闭合，POK 触点断开，APMRD 触点断开。
4. APMRD 触点重新闭合，表示又可以接收命令。
5. 在错误被清除之前，除了 FJOGP、SJOGP、FJOGN、SJOGN、JOGOFF、RESET、CLRERR 命令以外，其他命令不再执行。

进入出错状态以后，用户应查清出错原因，排除故障以后，重新复位再开始新的工作周期。

为了便于用户排除故障，E-02PP 模块在检测到错误条件时，返回一个错误类型编号，指出错误原因。这些错误编码的含义说明如下：

- 07 非法命令码，即从 SR21/SR22/SE-22 送来的命令码在命令表规定的命令码之外。
- 10 送到步进控制模块的步进马达相位分配方式错。
- 13 MOV 类命令缓冲区溢出。
- 15 正在找原点时，接收到 MOV 类命令。
- 16 正在进行其他运动时，接收到找原点命令。
- 17 正在进行其他运动时，接收到调整命令。
- 19 正在运动时，接收到 PTELL 命令。
- 20 PTELL 命令的参数越过硬限位开关。
- 22 快速调整速度超过最大允许值。
- 23 慢速调整速度超过最大允许值。
- 27 调整加速度超过最大允许值。
- 29 PHOME 命令设定的原点位置值越过硬限位开关或正在找原点时进行原点位置坐标值设定。
- 30 正在进行其他运动时，接收到强制定位命令。
- 31 VLMT 命令设定的速度极限值为 0。
- 32 不允许有负参数的命令中出现负参数。
- 33 正在进行调整运动时，接收到 MOV 类命令。
- 35 在执行其他命令后接收到 STEPMD 命令。
- 42 速度值超过最大允许值。
- 43 速度值被设为 0。
- 44 加速度值超过最大允许值。
- 45 加速度值被设为 0。
- 46 基础速度超过最大允许值。
- 47 在 FMOV 命令之后接收到 DELAY 命令。
- 48 在 FMOV 命令之后接收到 WAIT 命令。
- 49 FMOV 命令之后的命令企图使轴改变运动方向。

- 50 第一个 FMOV 命令的距离太短，以规定的加速度从 0 达到最终速度所经过的距离大于命令规定的距离。
- 51 中间过程中的 FMOV 命令的距离太短，以规定的加速度从现在的速度达到新的速度所经过的距离大于命令规定的距离。
- 52 FMOV 以后的 MOV 或 PMOV 命令的距离太短，以规定的加速度从现在的速度减速到 0 所经过的距离大于命令规定的距离。
- 53 企图或已经使轴越过正向限位软开关。
- 54 企图或已经使轴越过负向限位软开关。
- 55 MOV 类命令的定位距离或坐标值超过内部可以表示的范围。或企图执行需要 655 分钟以上才能执行完毕的命令。
- 56 PMIN 命令的参数为正，或 PMAX 命令的参数为负。
- 57 子程序缓冲区溢出，即所有子程序中命令条数之和超过 512。
- 59 企图调用未定义的子程序。
- 60 子程序之间出现递归调用现象。
- 62 在子程序内部定义子程序或在子程序之外用 RET 命令。
- 63 在 FMOV 命令之后以 MVPMOV 结尾。
- 64 速度表中 V0 或 V1 被定义为 0。
- 65 速度表中速度设定超过最大允许值。
- 67 在执行 MVPMOV 命令时，发现速度表中速度不按递减规律排列。
- 69 FMOV 命令的执行时间过短。FMOV 命令已执行完毕，但后继的命令尚未接收到或还在进行预处理。有两个 FMOV 连续执行而前面一个 FMOV 的执行时间短于 200ms 时可能会出现该错误。
- 71 OV+开关接通，被认为正向越界。
- 72 OV-开关接通，被认为负向越界。
- 73 OV+和 OV-同时接通，或在找原点时发现 OV+、OV-和 HOME 开关的关系不正确。
- 75 控制失调，即位置误差超过位置误差极限值部分达到 1024CT。
- 80~98 这类错误码为模块内部的软件自身的故障。
- 90 硬件诊断错误码。当模块对自身的硬件诊断发现错误时将会报告 99 号错误。当发生 99 号错误时，返回数据寄存器中原来为步号的位置返回的硬件诊断号，诊断号有很多情况。对于硬件的故障或 80 号以上的错误号（即软件故障），请将出错时的使用情况说明，PLC 的程序清单和出错的情况说明一起交给光洋电子（无锡）有限公司检查维修。

第四章 编程方法

轴定位模块的各类命令，是通过 SR21/SR22/SE-22 的梯形图程序发送到轴定位模块，同时轴定位模块的返回数据也要通过梯形图程序接收到 SR21/SR22/SE-22 的 CPU 内部，完成这些功能的 SR21/SR22/SE-22 指令称为智能模块控制指令。

智能模块控制指令包括 F29, F30, F31, F32, F33, F40, F41, F42, 这些指令的含义和用法请参阅“智能模块控制指令组说明”。

4. 1 编程注意

4. 1. 1 执行遍数和执行次序

智能模块控制指令的操作码均为 Fxx, 和它的 Fxx 指令一样, 当连接到这些指令的逻辑行 ON 时, 该指令的动作就执行, 否则就不执行。大部分情况下, F29、F30~F33 指令的执行只在需要时执行一遍, 而不是重复执行很多遍。所以使用时一定要注意: 使这些指令只在需要时接通一个扫描周期, 即应用 SR21/SR22/SE-22 的 SET OU RET 指令来执行轴定位命令。

当 SR21/SR22/SE-22 的一个 CPU 之下连接多台智能模块时, 要对不同的智能模块进行选择, 选择的指令 F29 在何时执行, 这一点非常重要, 一定要注意: 在 F30~F33 或 F40~F42 指令执行的结果是针对指定的智能模块的, 如果选择错误将会引起混乱。但 SR21/SR22/SE-22 的一个 CPU 之下只连接了一个智能模块时, 就不必担心混乱, 那时不必要用 F29 指令来指明操作的模块, CPU 也能自动判别模块的 I/O 定义号。

4. 1. 2 F30~F33 指令的参数格式和选用方法

F30~F33 指令的作用是将一个命令码和一个参数送到智能模块, 轴定位模块对每个规定的命令码都有相应的功能与之对应。不同的命令码对其所带的参数有不同的要求, 第三章在每个命令的说明中已对参数的要求进行了说明, 并列出了在 SR21/SR22/SE-22 上用来传送该命令和参数的指令序列, 这里说明一下指令的选用原则和活用方法。

一般说来, 命令需要的参数长度 (见命令一览表), 应和 F30~F33 命令所传送的参数长度一致, 不过在所传送的实际参数为正数时, 某一参数的长度比规定的最大参数位数小时, 可以选用传送参数位数少的指令, 但是参数为负数时, 则必须以 8 位 BCD 码进行传送 (用 F33 命令), 负数的表示方法是在 8 位 BCD 码的最高位上放一个 8。F30~F33 指令的参数传送能力和适用范围如下表所示:

指令	参数 BCD 码位数	参数范围
F30	0	0
F31	2	1~99
F32	4	100~9999
F33	8	10000 以上或负数

在执行了 F31~F33 指令之前, 必须先将要传送的参数装入 SR21/SR22/SE-22 的累加器中, 其中辅助累加器中的是高四位部分。如果传送的 BCD 码不超过四位时, 不需要使用辅助累加器。

4. 1. 3 返回数据的用法

返回数据的两部分由 F40~F42 指令取到 SR21/SR22/SE-22 的 CPU 内,其中 F40 不取数据寄存器内容, F41 不取数据寄存器的高 16 位 (bit), F42 取入所有内容。8 个状态触点取入后,存放在轴定位模块所插的槽内相应的 I/O 触点状态表中,用户可象输入模块上的点一样来对待这些触点。但是这些触点不会象输入点一样自动更新状态,而只在执行过 F40~F42 指令后,才会更新一次状态。返回的内容大多数情况下也是 BCD 码,只有 0 号方式中有一部分二进制的状态。需注意:只有在读入的状态中的 RDOK 触点闭合时,取来的内容才是有效的。如果用取来的数据运算后进行控制,必须不在刚改变返回数据格式以后就立即取数据,对某些随时变化的数据还应注意返回的数据存在一定的滞后,在模块的内部,每 9.4ms 刷新一次返回数据。如果只把返回数据用来显示的话,则可以不考虑数据的有效性和滞后特性,只需在 PLC 的每个扫描周期中取一次返回数据送到显示器即可。

4. 1. 4 指令执行时间和对 SR21/SR22/SE-22 扫描时间的影响

智能模块专用指令组中,除了 F29 以外,其它指令的执行都要和智能模块进行一定的信息交换,而且这些指令往往需要一些数据处理指令与之配合,指令的执行时间比较长。

如果在 PLC 的一个扫描周期中,执行这类指令过多的话,将会使 PLC 的扫描周期变长。为了使扫描周期不超过规定的上限 200ms,就应合理安排程序,将参数设定和子程序定义这类只需要在初始化时执行一遍,但命令较多的部分安排在开始的几个扫描周期内完成。

作为合理安排程序的参考,将 F30~F33 和 F40~F42 指令的执行时间列表如下:

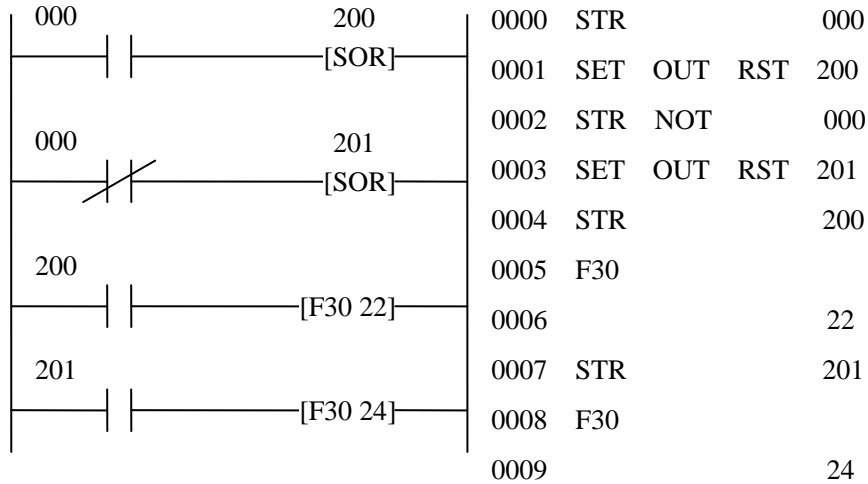
命 令		执行时间	
		有效时	无效时
F30	F40	约 0.25ms	0.01ms
F31		约 0.35ms	0.01ms
F32	F41	约 0.45ms	0.01ms
F33	F42	约 0.65ms	0.01ms

4. 2 编程举例

因为以下例子中经常要用到 SR21/SR22/SE-22 的指令,阅读以下例子以前,请先阅读《SR20/21/22 用户手册》(CKER20101)和本说明书的附录《智能模块控制指令组说明》。

4. 2. 1 例 1：通过 PLC 指令使马达作调整运动动作

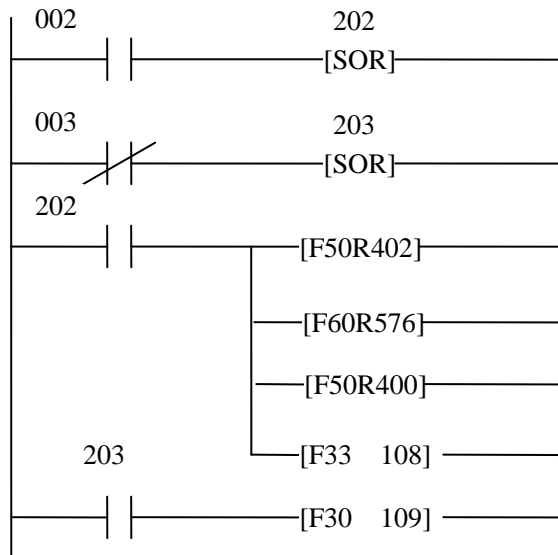
用一个开关控制马达，开关接通，马达运动；开关断开，马达停止。马达的运动是慢速正向调整。实现以上功能的梯形图和程序如下：



4. 2. 2 例题：通过 PLC 指令使马达进行定位运动

用两个开关控制马达，2号开关每接通一次，马达正向运动N个CT，3号开关接通后，马达回到机械零点，即座标为零的点。N的值存放在数据寄存器R400~R403内。注意：本例中未列出SR21；/SR22的助记符命令行，用户可自行将梯形图改写成PLC命令。

实现以上功能的梯形图如下：



4. 2. 3 例外：用两个按钮和一个开关实现快速，慢速调整

I/O 定义号为 10, 11 的两个输入点为按钮，其作用如下：

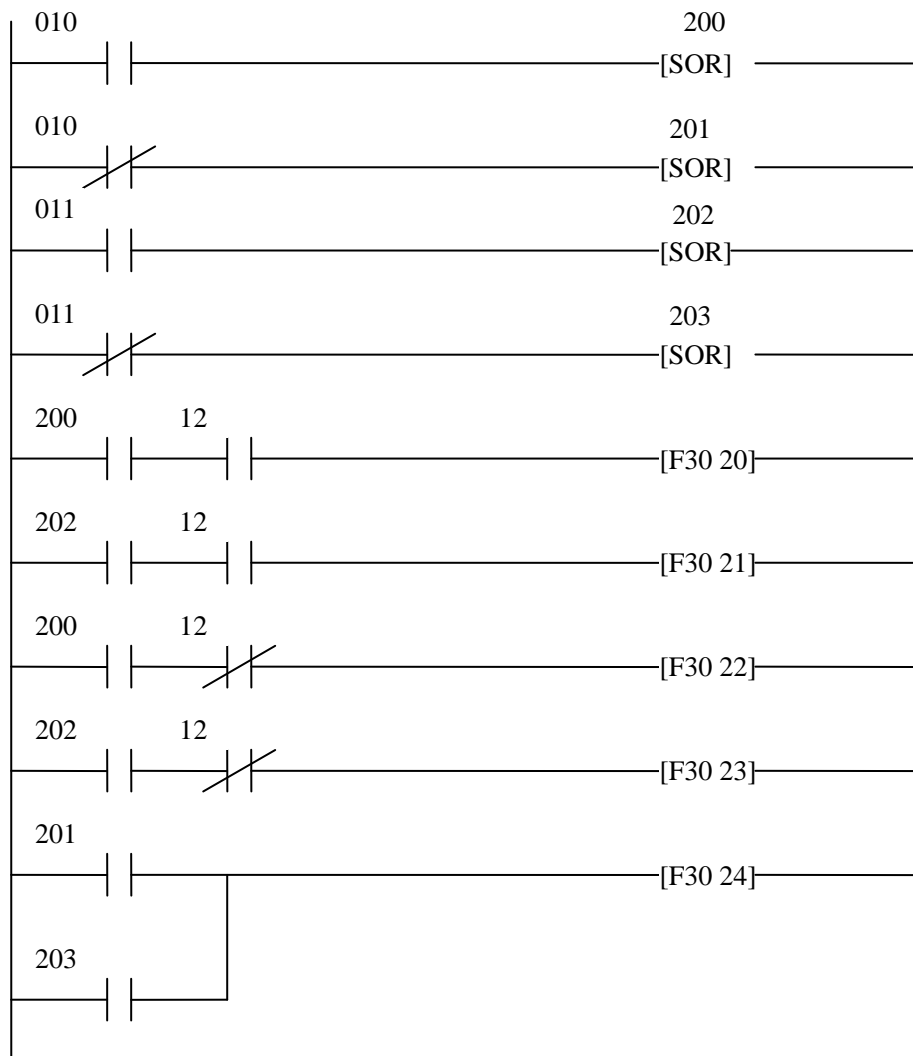
10 号点：ON 时为正向调整；

11 号点：ON 时为负向调整。

I/O 定义号为 12 的输入点为钮子开关，其作用如下：

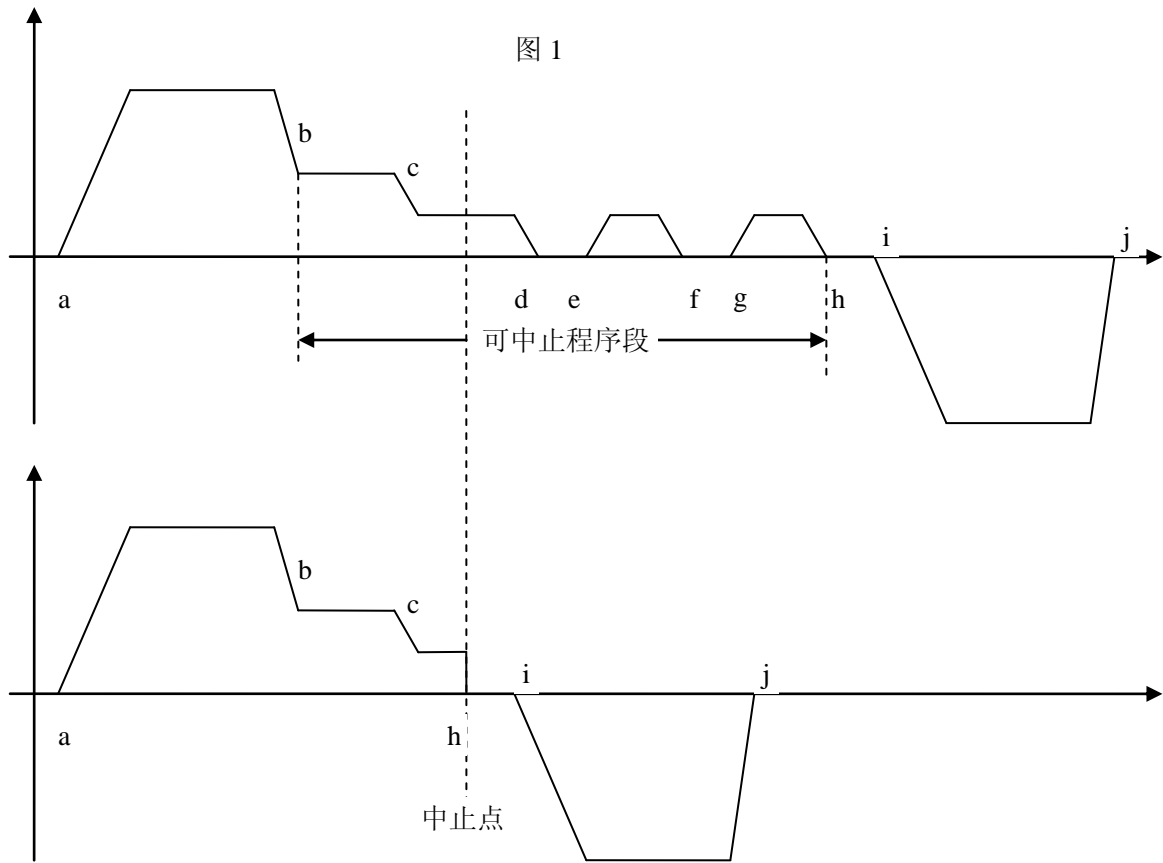
12 号点：ON 时为快速调整；OFF 时为慢速调整。

实现以上功能的梯形图如下：

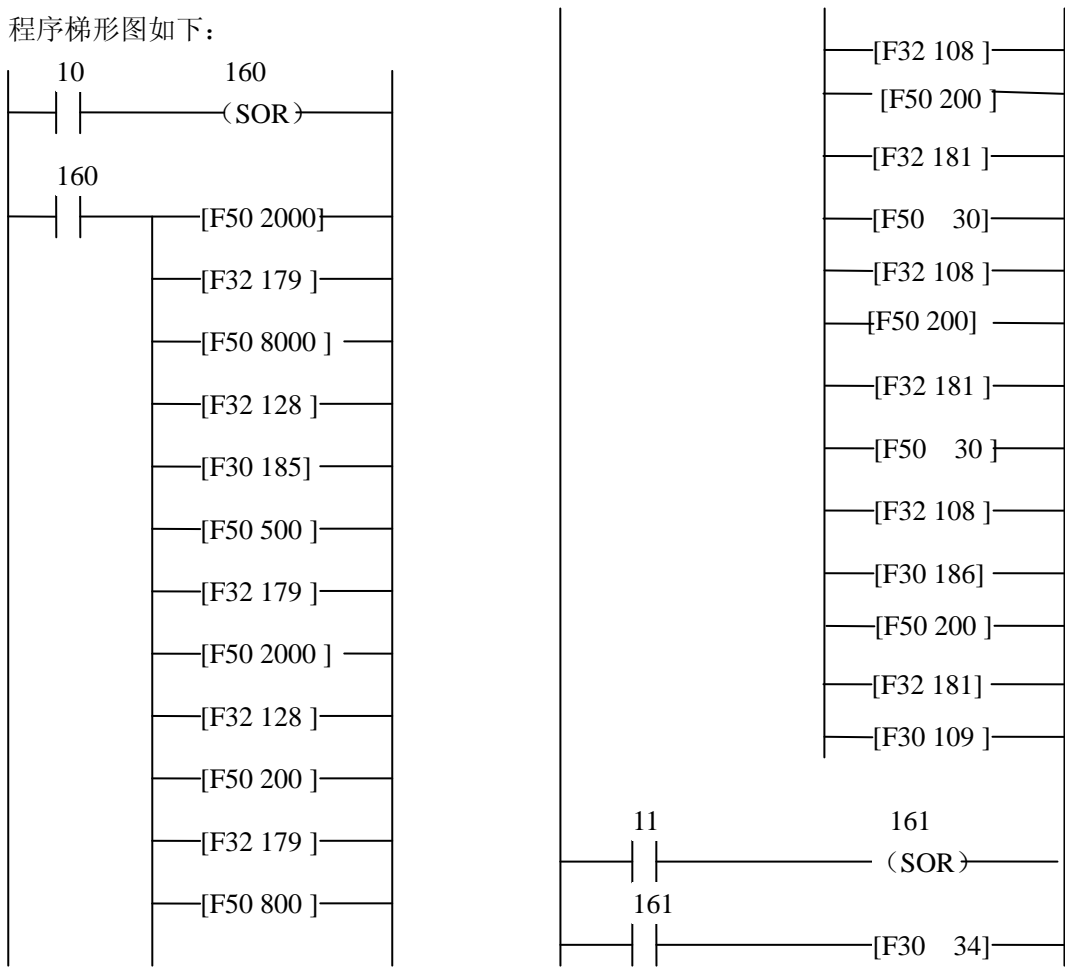


4. 3. 4 例行：185，186 指令的应用

本例当 10 号开关接通时，使轴以时间速度图 1 运动；次序是 a, b, c, d, e, f, g, h, i, j。
当 11 号开关接通时，以时间速度图 2 运动，使轴在用 185, 186 命令定义的程序段中止运动，然后执行 186 以后的命令，则次序是 a, b, c, h, i, j。

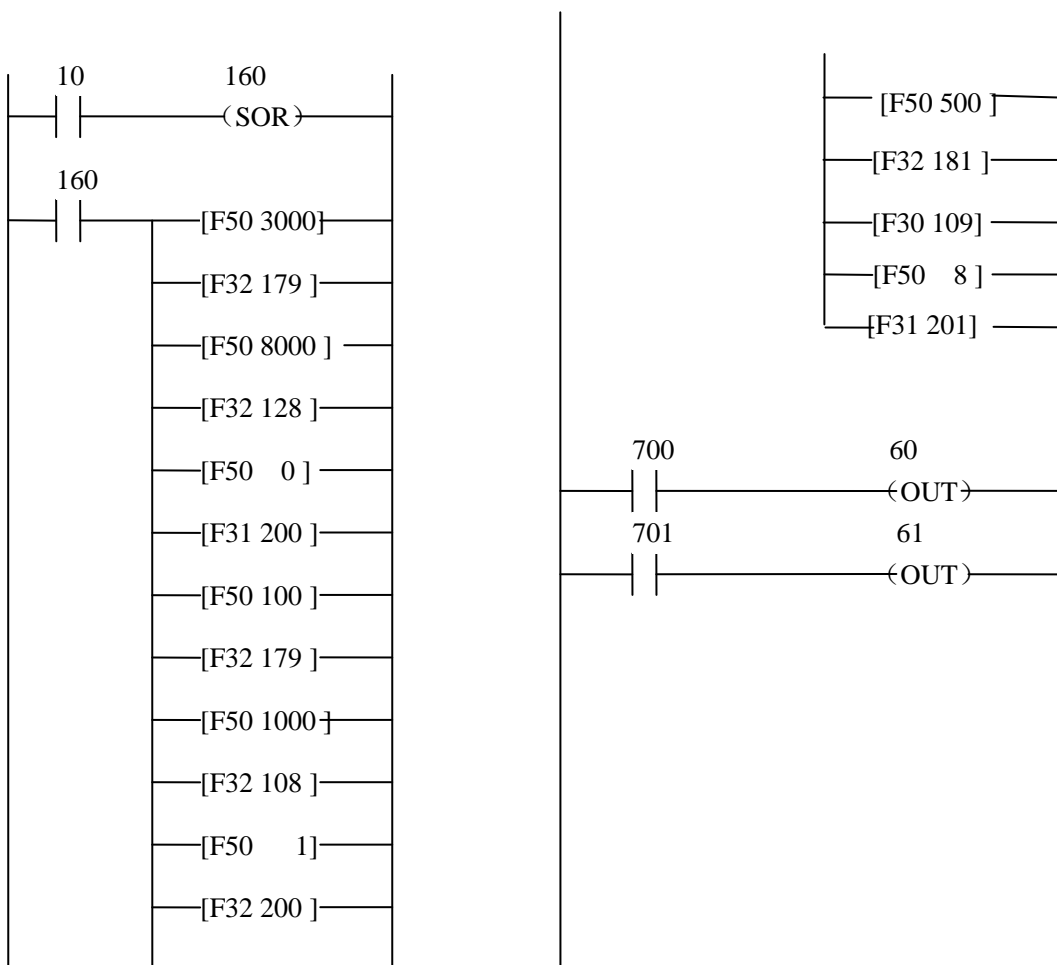
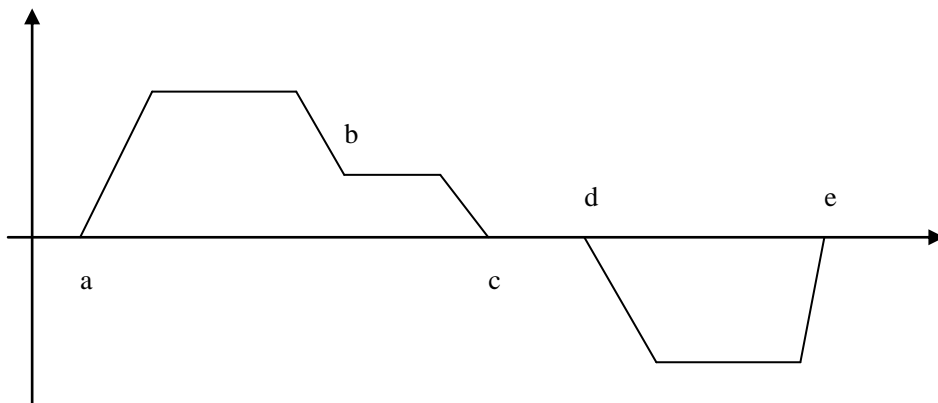


程序梯形图如下：



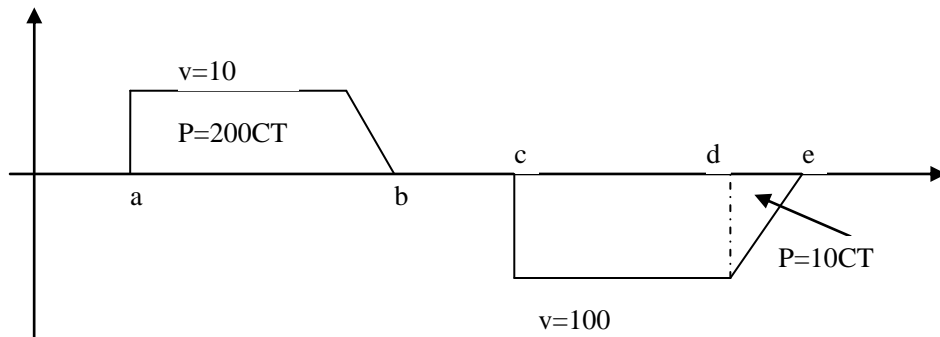
4. 3. 5 例 5：200，201，202 指令的应用

在轴定位模块执行命令过程中，如果需要控制其它的设备，可以用 200，201，202 命令向 SR21/SR22/SE-22 发出信号，以便控制其它设备或执行其它动作。本例假设返回数据的第二部分放在 R70 中，那么如果在执行下面的时间速度图时，希望在 b，c 点使输出点 60，61 分别接通，到了 e 点全部断开，可编制下列程序：

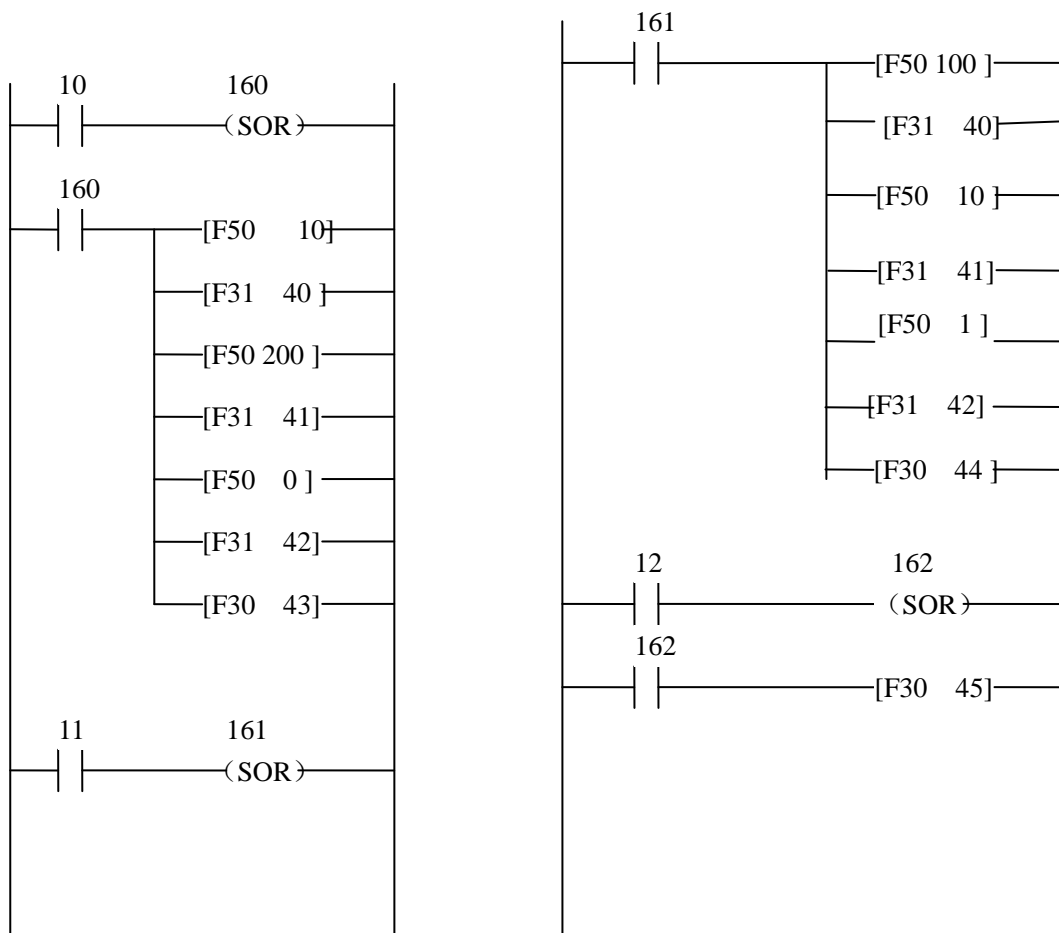


4. 3. 6 例 6: Y 轴运动举例

假定 E-02PP 装在定义号为 0 的槽上，在定义号为 10、11、12 的三个输入点上安了三个按钮，如果要完成下面的功能：按下 10 后，Y 轴以速度 10 向前走 200CT；按下 11 后，Y 轴以速度 100 定速运动，直到按下 12 后，Y 轴再走 10CT 然后停止。动作的时间速度图如下：

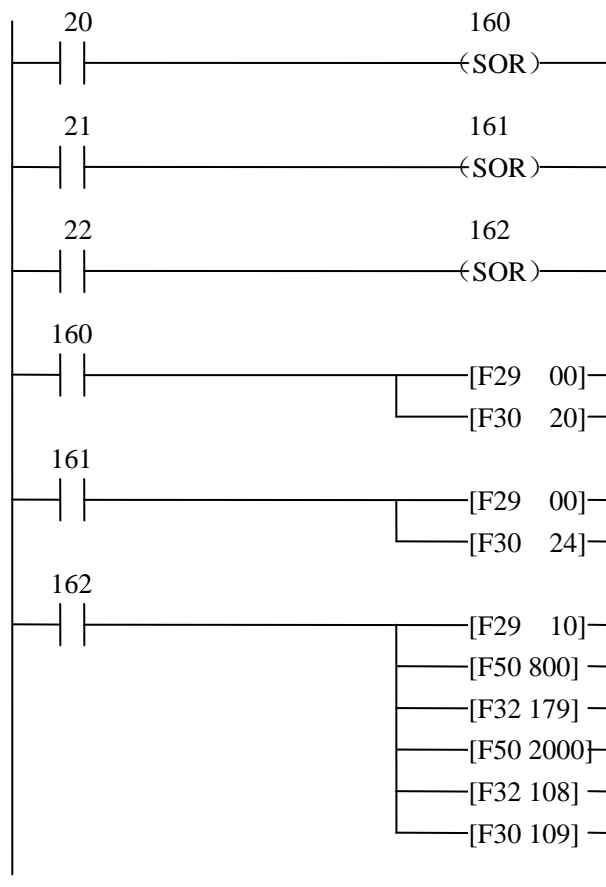


用户程序如下：



4. 3. 7 例 7：一个系统中多个 E-02PP 同时动作的应用

在 SR21/SR22/SE-22 系统中，可以用两个或多个 E-02PP 同时控制不同的步进马达，只要在编程时用 F29 命令指定相应的模块即可。假设有两块 E-02PP，一块放在 0 号槽上，另一块放在 1 号槽上，则其定义号分别为 00、10；有三个按钮，定义号分别为 20、21、22。如果系统需要进行下列运动：按下 20 号按钮，则 1 号槽 E-02PP 进行定位运动。那么程序如下所示：



4. 2. 8 例 8：采用 E-02PP 模块控制磨床的磨头定位

本程序只列出了磨削加工的两个主要动作，一个是磨削加工，一个是砂轮修整。

磨削的过程如下：

- ①砂轮快速接近工件。
- ②以粗磨进给速度进行磨削。
- ③到预定粗磨尺寸后以精磨进给速度进行磨削。
- ④进给到尺寸后进行一定时间的无进给磨削。
- ⑤工件到尺寸后磨头退回。

砂轮修正的过程如下：

- ① 将磨头移动到砂轮修正位置。
- ② 砂轮快速接近砂轮修正器。
- ③ 以修正速度进给修正砂轮。
- ④ 进给到预定位置后进行一定时间的无进给修正。
- ⑤ 磨头退回。
- ⑥ 将磨头移动到磨削位置。

系统的设置情况：

磨头的传动机构设定为：一个 CT 是 0.25μ ，磨削过程中考虑到砂轮的磨损，每磨削 4 个工件补偿 0.25μ ，磨削 20 个工件后进行砂轮修正，修正量是 25μ 。

磨头退回后坐标为零。

精磨后最终尺寸为磨头到达坐标 20000CT 的位置，磨头的移动距离为：

$$20000 \times 0.25\mu = 5\text{mm}$$

精磨量为 30μ ，即为 120CT，精磨时间为 6 秒。

粗磨速度为精磨速度的 4 倍。

砂轮接近工件时分两种速度，第一种速度是空程移动速度，以允许的最高速度（假定为 10000vu ）运动，第二种速度为准备速度（取粗磨速度的 20 倍），假定马达允许在场 0.1 秒内从静止加速到最高速度。

计算过程：

精磨速度计算：

$$\frac{120\text{CT}}{6 \text{ 秒}} = 1200\text{CT/分} = 12(100\text{CT/分}) = 12\text{vu}$$

粗磨速度计算：

$$12 \times 4 = 48\text{vu}$$

准备速度计算：

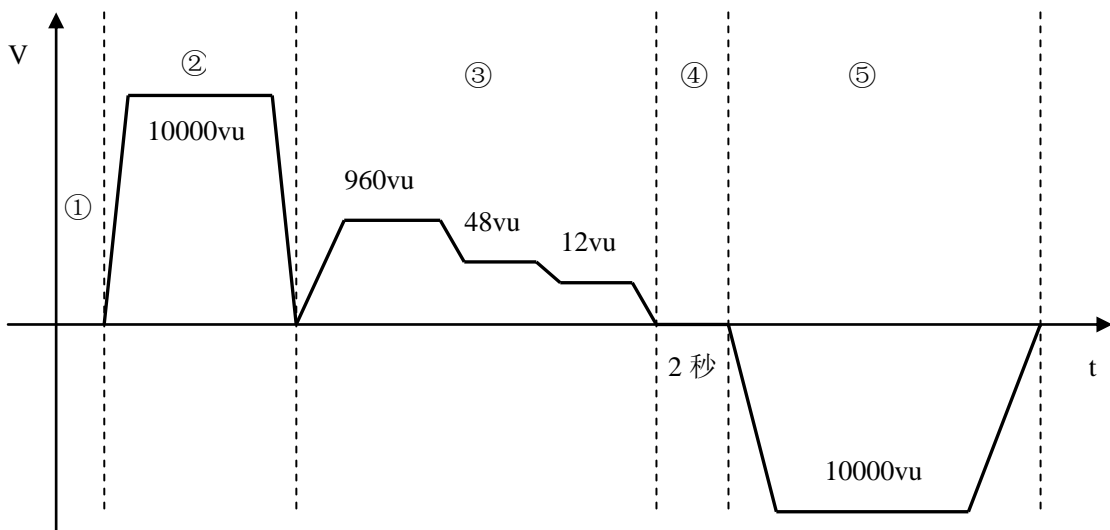
$$48 \times 20 = 960 \text{ vu}$$

空程移动速度： 10000vu

加速度为：

$$10000\text{vu}/0.1 \text{ 秒} = 100000\text{vu} = 10000\text{au}$$

磨削过程的时间速度图：



磨削各阶段的说明：

磨削过程的开始应是工件确实装好后，所以，时间速度图上的①阶段就是等待阶段。一旦装好后，从 SR21/SR22/SE-22 向轴定位模块发送一条 RESTART 命令，就进入阶段②。

阶段②是磨头以快速接近工件的过程。

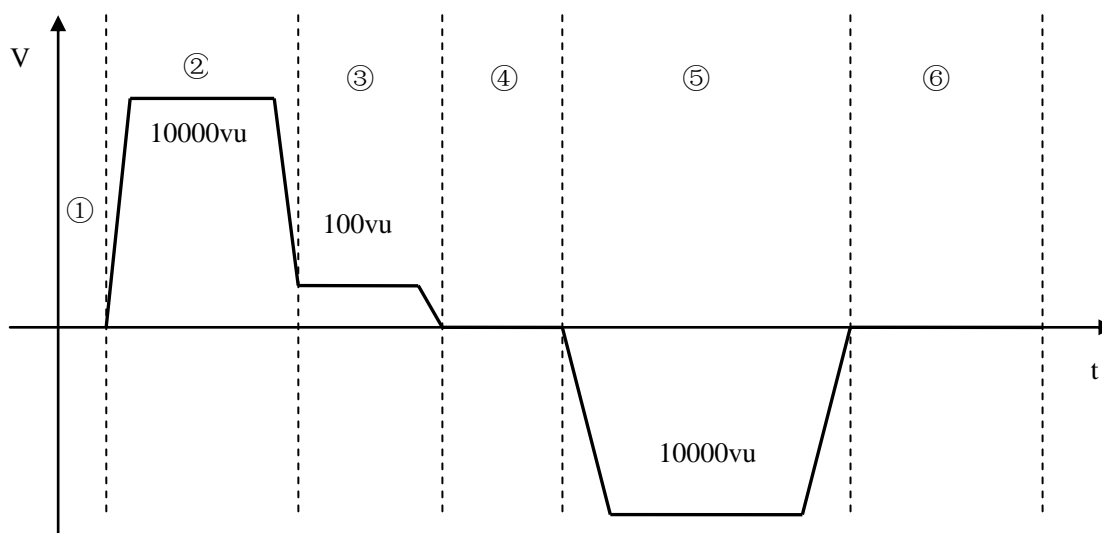
阶段③是一个接触工件→粗磨→精磨的过程。这个过程中从工件到粗磨这一阶段是传感器决定的，所以采用 MVPMOV 命令最后适，其中精磨阶段的距离是保留距离（MVFP 命令设定）。这一阶段中有三个速度，第一个速度为准备速度，即砂轮尚未接触工件时的运动速度，一旦 PLC 从传感器接收到砂轮接触工件的信号，就向轴定位模块发出 VCHG 命令，轴定位模块就改变为第二个速度，即粗磨速度，以粗磨速度运行到最终目标的可保留距离内时，就变为第三速度，即精度速度运行，最后精确定位到最终的目标点。

在粗磨阶段，如果在进入保留距离之前又接收到来自 SR21/SR22/SE-22CPU 的命令，也可以提前进入精磨阶段。

阶段④是无进给磨削阶段，时间为 2 秒。

阶段⑤是磨头快速退回到原始位置，磨削的一个周期就到此结束。

磨头修正的时间速度图：



磨头修正各阶段的说明如下：

①是一个等待阶段，此时，SR21/SR22/SE-22 正在控制磨头从磨削位置移动到修正位置，一旦修正的准备工作完成以后，SR21/SR22/SE-22 就向轴定位模块发出一个 RESTART 命令，开始进入②的阶段。

②是磨头快速接近修正的阶段，以快速将磨头运动到修正器附近。②阶段结束时不为零，而是正好等于下一阶段的速度。

③是磨头修正阶段，以修正速度向前运动。注意，修正中采用了与磨削不同的变速方法，这里仅仅是为了说明各种运行方式，实际应用中也可以采用与磨削过程一样的方法，根据外界条件来决定何时变速。

④在最终修正位置延时一定的时间，以消除弹性变形。

⑤修正结束，将磨头退回到原始位置。

⑥等待阶段，此时 SR21/SR22/SE-22 应控制磨头从修正位置返回到磨削位置，返回后就可进入以后的磨削阶段。

磨削和修正砂轮是两个固定的过程，把两个过程各定义为一个子程序在时需要调用，编程过程比较简单。

磨削子程序：

- | | | |
|------------|-------|------------------|
| 1. STEP | 0 | 将步号清零 |
| 2. WAIT | | 等待 |
| 3. MOVVEL | 10000 | 设定空程运行速度 |
| 4. PMOV9 | 18000 | 设定空程运行距离 |
| 5. MOVPMOV | 0 | 可变速定位，进行准备，粗磨，精磨 |
| 6. DELAY | 200 | 无进给磨削 2 秒 |
| 7. PMOV9 | 0 | 返回 |
| 8. RET | | 子程序结束 |

上述子程序中各行说明如下：

1. STEP 0 是将步号清零，以后每次执行磨削过程时，各个阶段的步号就相对固定了，例如下一个 WAIT 命令执行时，步号总是 1。

2. WAIT 这条命令的目的是为了使磨削的动作与工件的安装工序取得同步，一旦工件安装完毕以后，SR21/SR22/SE-22 向轴定位模块发一个 RESTART 命令，就可开始执行下面的动作。

3. MOVVEL 10000 设定空行程速度为 10000vu。

4. PMOV9 18000 将磨头移到坐标位 18000 的位置，可以不进行最终的精定位过程，以提高效率，而且在此命令执行完后，实际速度尚未为零，便于与下一个运动的速度相衔接。

5. MOVPMOV0 该命令允许磨头在到达目标以前改变速度，各阶段的速度由速度表中的值来确定。这个例子中的速度表中的值和保留距离值只需确定一次，所以不放在子程序内执行，而在初始化时做一次。修改运动目标是由零号位置寄存器决定的。因为在磨削过程中，砂轮因为磨损和修正要经常改变尺寸，所以这里在位置寄存器中存放这个最终位置值，便于在需要时进行修改。

6. DEKAY 200 是延时 2 秒的命令 执行无进给磨削动作。

7. PMOV9 0 是退回磨头到坐标为零的位置，为下次动作做准备。

8. RET 为子程序的结束。

相应地根据修正过程的时间速度图编出修正过程的子程序，本子程序只考虑工作过程中的砂轮修正，未考虑砂轮修正的要求。

- | | | |
|-------------|-------|---|
| 1. STEP | 10 | 为修正过程设定初始步号。 |
| 2. WAIT | | 等待磨头移到修正位置。 |
| 3. PTABSUBO | 105 | 将 0 号位置寄存器值减少一定量比磨削最终值减小 105CT，确保磨头快速前时不接触到修正器。 |
| 4. MOVVEL | 10000 | 为快速前进设定速度。 |
| 5. FMOV0 | | 设定快速前进的最终位置为零号寄存器中规定的值。 |
| 6. PTABDD0 | 200 | 将 0 号位置寄存器的值增加一定量（即比磨削后的值增加 95）。 |
| 7. MOVVEL | 100 | 设定修正进给速度。 |
| 8. PMOV0 | | 精确定位到修正位置。 |
| 9. DELAY | 100 | 延时 1 秒。 |
| 10. MOVVEL | 10000 | 为快速退回设定速度。 |
| 11. PMOV9 | 0 | 磨头退回坐标为 0 的位置，即磨削过程开始的位置。 |
| 12. WAIT | | 等待磨头移动到磨削位置。 |

13. RET 修正过程子程序定义完毕。

有了这两个基本过程子程序以后，再设计出整个工作周期的程序，考虑如下：

磨削补偿子程序：调用磨削子程序 4 次以后，对 0 号位置寄存器中的值加 1。

修正和磨削过程子程序：每修正一次，调用磨削补偿子程序 5 次。

换砂轮以后执行的子程序（假定新砂轮的修正已由其它程序控制并完）：设 0 号位置寄存器的初值；调用修正磨削子程序 5 次。

全部子程序定义如下：

SUBR0 换砂轮后执行的子程序。
 DPTAB0 19905 0 号位置寄存器初值设定。
 CALL1 5 调用修正，磨削子程序 5 次。
 RET

SUBR1 修正、磨削子程序。
 CALL2 修正 1 次。
 CALL3 5 磨削补偿过程执行 5 次。
 RET

SUBR2 修正子程序。

STEP 10
 WAIT
 PTABSUB0 105
 MOVVEL 10000
 PMOV0
 PTABADD0 200
 MOVVEL 100
 PMOV0
 DELAY 100
 PMOV0
 DELAY 100
 MOVVEL 10000
 PMOV9 0
 WAIT
 RET

SUBR3 磨削补偿子程序
 CALL4 4 磨削 4 次。
 PTABADD0 1 补偿值加到 0 号位置寄存器中。
 RET

SUBR4 磨削子程序。

STEP 0
 WAIT
 MOVVEL 10000
 PMOV9 18000

```
MVPMOV0
DELAY 200
PMOV9 0
RET
```

在磨削子程序中，用到了 MVPMOV 命令，这个命令是执行磨削时运行命令，各阶段的运行速度由速度表规定的，而且还有相应的保留距离等与该命令相关，这些内容在执行该命令之前预先进行设定，由于这些参数值在运行过程中并不改变，所以考虑安排一个初始化程序，在执行工作周期之前，先执行这个初始化程序：

```
初始化程序如下：
DVTAB0 10000
DVTAB1 48
DVTAB2 12
DVTAB3 0
MUFP 120
MOVACC 9999
RDATA 11
```

上述程序定义了速度表中 V0=1000vu，V1=48 vu，V2=12vu，保留距离为 120CT，即精磨量为 120/4=30μ，运行中改变速度时采用的加速度为 9999au，根据计算，该值本应为 10000au，但因为 10000 和 9999 相差很小，而在编程时用 9999 可以由较少的指令完成。对于其它不很重要的参数，如速度，也可以在这种情况下适当调整，使程序较为简单。RDATA 命令设定返回参数格式为步号和实际位置。

以上程序还不能直接装入到 SR21/SR22/SE-22 的 CPU 中去，而需要变为 SR21/SR22/SE-22 的指令以后，才可以装入。第三章命令说明中，对每一条命令都列出了 SR21/SR22/SE-22 的参考指令格式，为了编程方便，这里向用户推荐一种变换用的表格，并说明用法。（本章的最后有一页空表，可供复制使用）

SR21/SR22/SE-22 轴定位模块编程用表

参数长度	SR21/SR22/SE-22 指令					轴定位命令		
	F50	F60	F50	F3X	命令码	命令	参数	注 释
5	1	○	1400	3	140	DPTAB0	19905	0 号位置寄存器初值

例如对一条指令，DPTAB0 19905，首先将命令的助记符 DPTAB0 填入“命令”一栏，参数值也填入相应的栏中，就以这种格式将设计好的程序填入。做完以后，在参数长度栏中填入相应的数字，如参数 19905 为 5 位数，则填 5。如是命令需要的参数不是常数而是放在 SR21/SR22/SE-22 数据寄存器中的值，则应考虑参数最大可能的位数，如果参数为负数，则无论大小，一虑应取 8 位。如果参数位数为 0，则参数位数就认为是 0，然后根据参数位置在 F3X 栏中填入相应的数字，规则如下：

参数位数	F3X
0	0
1~2	1
3~4	2
5~8	3

下一步如果 F3X 栏中为 3，那么在相应的 F60 栏中画一个“○”，然后在左右两个 F50 栏中填入数字，F60 栏中无“○”时左边的 F50 栏中也不必填入内容。需要时，左边的 F50 栏填高位数字，右边的 F50 填低位数字，负数在左边的 F50 栏填写 4 位数字，最高位为 8。如果命令的参数不是常数，而是 SR21/SR22/SE-22 的数据寄存器号时，则填和相应的寄存器号，最后根据“命令一览表”查出命令码，在“命令码”一栏中填入。这样 SR21/SR22/SE-22 的指令内容就已基本变换好了，按从左到右的次序填入到 SR21/SR22/SE-22 中去即可。

例如表中的例了，轴定位命令 DPTAB0 19905 变换为 SR21/SR22/SE-22 指令就是：

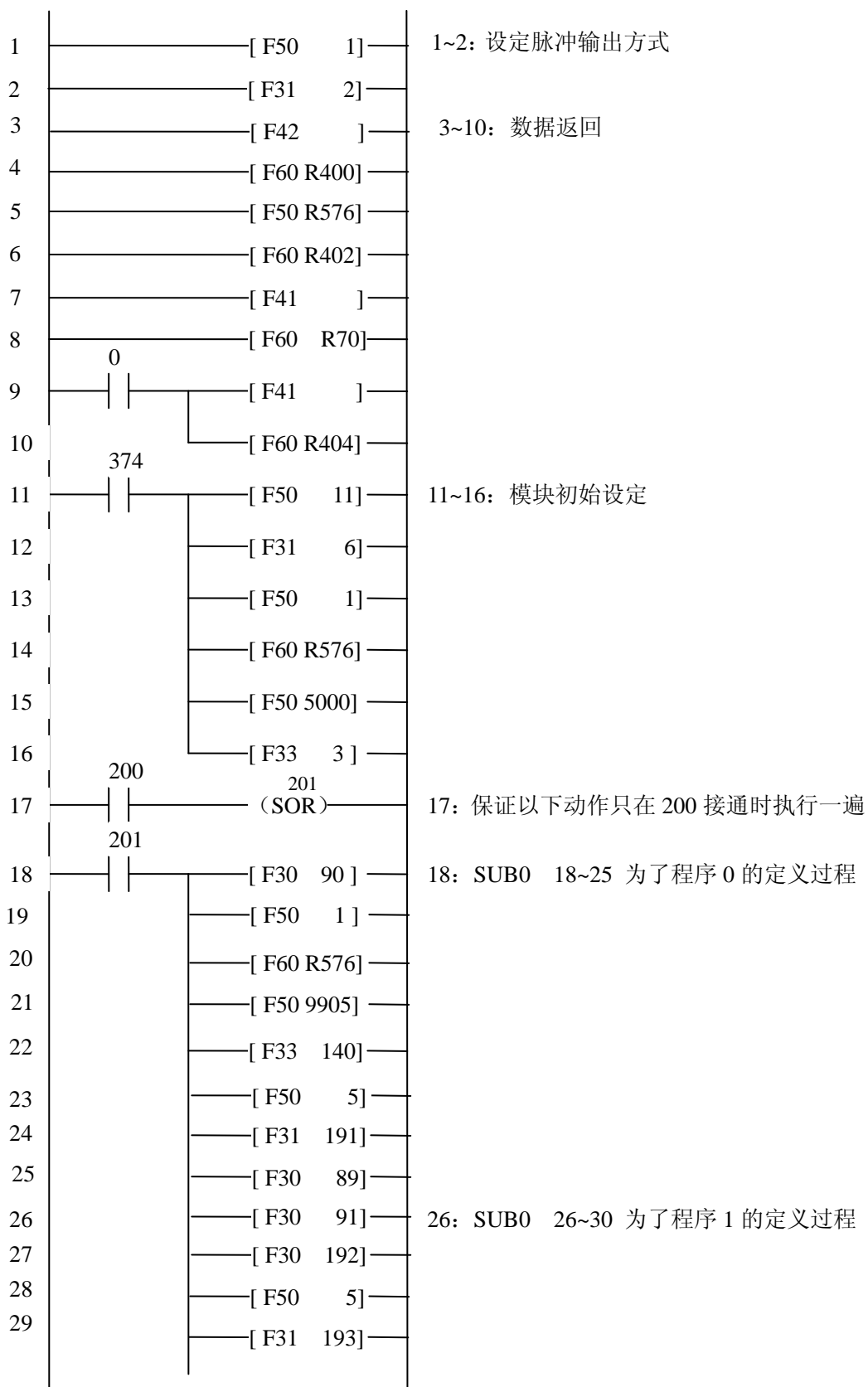
```
F50 1
F60 R576
F50 9905
F33 140
```

下面是本例子中子程序定义部分和参数初始化部分的程序按上述方法处理的表格：

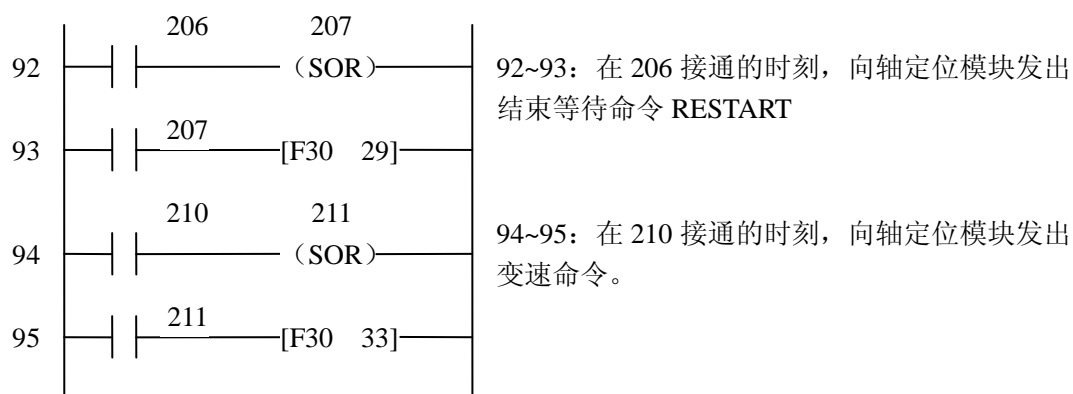
参数长度	SR21/SR22/SE-22 指令					轴定位命令		
	F50	F60	F50	F3X	命令码	命令	参数	注释
				0	90	SUBR0		子程序 0
5	1	○	9905	3	140	DPTB0	19905	
1			5	0	191	CALL1	5	
				0	89	RET		
				0	91	SUBR1		子程序 1
				0	192	CALL2		
1			5	1	193	CALL3	5	
				0	89	RET		
				0	92	SUBR2		子程序 2
2			10	1	182	STEP	10	
				0	183	WAIT		
3			105	2	160	PTABSUB0	105	
5	1	○	0	3	179	MOVVEL	10000	
				0	120	FMOV0		
3			200	2	150	PTABADD0	200	
3			100	2	179	MOVVEL	100	
				0	100	PMOV0		
3			100	2	181	DELAY	100	
5	1	○	0	3	179	MOVVEL	10000	
				0	109	PMOV9	0	
				0	183	WAIT		
				0	89	RET		

				0	93	SUB3		子程序 3
1			4	1	194	CALL4	4	
1			1	1	150	PTABADD0	1	
				0	89	RET		
				0	94	SUBR4		子程序 4
				0	182	STEP	0	
				0	183	WAIT		
5	1	○	0	3	179	MOVVEL	10000	
5	1	○	8000	3	109	PMOV9	18000	
				0	130	MVPMOV0		
3			200	2	181	DELAY	200	
				0	109	PMOV9	0	
				0	89	RET		
3			1000	2	170	DVTAB0	1000	初始化部分
2			48	1	171	DVTAB1	48	
1			12	1	172	DVTAB2	12	
				0	173	DVTAB3	0	
3			120	2	148	MVFP	120	
4			9999	2	180	MOVACC	9999	
2			11	1	6	RDATA	11	

作好了上面的表后，就可以在 SR21/SR22/SE-22 上编出下列程序：



30	[F30 89]	
31	[F30 92]	31: SUB2 31~55 为子程序 2 的定义过程
32	[F50 10]	
33	[F31 182]	
34	[F30 183]	
35	[F50 105]	
36	[F32 160]	
37	[F50 1]	
38	[F60 R576]	
39	[F50 0]	
40	[F33 179]	
41	[F30 120]	
42	[F50 200]	
43	[F32 150]	
44	[F50 100]	
45	[F32 179]	
46	[F30 100]	
47	[F50 100]	
48	[F32 181]	
49	[F50 1]	
50	[F60 R576]	
51	[F50 0]	
52	[F33 179]	
53	[F30 109]	
54	[F30 183]	
55	[F30 89]	
56	[F30 93]	56: SUB3 56~61 为了程序 3 的定义过程
57	[F50 4]	
58	[F31 194]	
59	[F50 1]	
60	[F31 150]	
61	[F30 89]	
62	[F30 94]	62: SUB4 62~77 为了程序 4 的定义过程



除了以上的程序以外, 实际的磨床工作程还应有下列内容:

1. 参数设定。这些参数如不进行设定的话, 模块就采用默认值, 一般采用默认值也能进行正常工作, 但可能不是最好的状态。

2. 调整过程。调整过程除了在磨床调试需要以外, 每次工作之前, 找原点动作是应该进行的, 只有找原点以后, 运行中的位置坐标才是反映砂轮在磨床上的实际位置的。此外, 还有急停之类的动作也是必要的。

步进驱动器的控制。一般步进驱动器除了可以接收速度控制信号脉冲以外, 还有一些开关量的 I/O 信号, 用于工作状态控制, PLC 应将这些 I/O 点连到开关量的 I/O 模块, 根据步进驱动器的要求进行相应的控制。

其它控制过程。磨削加工除了砂轮的进给运动以外, 还有其它的动作, 如上下料、磨头在磨削、停止位置之间的运动, 装新砂轮及新砂轮的修正等。

上述程序未在这里列出, 但现在假设上述动作已按要求设计好程序, 并为执行磨削、修正工作做好了必要的准备。

以上程序在执行过程如下。

①使 200 号线圈接通, 定义所有的子程序。

②使 201 号线圈接通, 设定初始参数值。

③使 202 号线圈接通, 显示器显示返回数据值。关于显示器驱动的内容请参阅显示器驱动器模块的说明。

④通断 204 号线圈一次, 使修正、磨削周期开始执行。

⑤在执行过程中, 有三处要执行等待命令, 这三个等待命令分别为:

步号为 01 时, 等待工件安装完毕。

步号为 11 时, 等待砂轮从磨削位置移动到修正位置。

步号为 16 时, 等待砂轮从修正位置移动到磨削位置。

在程序的其它部分应该有判别这三个等待正在进行, 同时监视所需要的动作是否完成的程序, 一旦动作完成就使 206 线圈通断一次, 发出 RESTART 命令, 继续下面的工序。

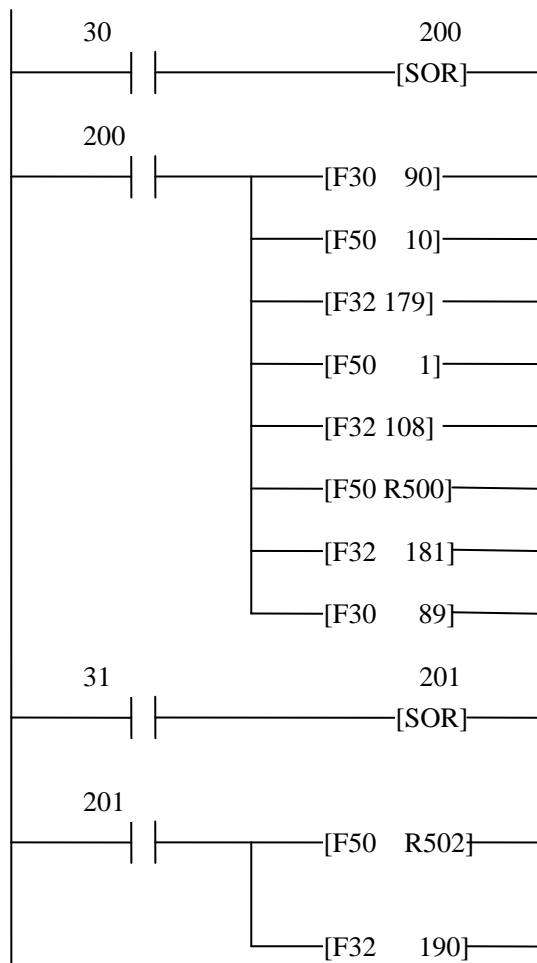
⑥在磨削中步号为 3 的工序中, 开始是以准备速度运行的, 这时 PLC 应随时监视砂轮是否已接触工件, 一旦发现接触工件, 就应使 210 号线圈通断一次, 使砂轮以进给速度进给。

⑦当磨削到 100 个工件后, 上述工作周期结束, 这时可以开始下一个工作周期, 即上述④开始循环工作。

4. 2. 9 例 9：速度低于 1vu 时的处理

E-02PP 的速度是从 1vu~60000vu，而实际工作中往往需要低于 1vu 的速度，这时要作特殊处理。本例就是说明 1vu 以下的处理。

本例中，R500 存放的是速度的倒数乘以 100，R502 存放的是要走的 CT 数。例如：以速度为 0.1Hz 走 3500CT 时，R500 应设为 1000，R502 应设为 3500。执行时，先接通 30，定义子程序；然后接通 31，走 R502 中设定的 CT 数。



附录：智能模块控制指令组说明

智能模块控制指令组是在标准的 SR21/SR22/SE-22 上增设的控制专用的智能模块（例如轴定位模块）而用的指令，这些指令的含义和用法介绍如下。

F29 是指定操作对象智能模块的指令，它是一条双字节的指令。F29 后面跟一个数据用来指定 I/O 定义号。

I/O 号表示 F3X、F4X 指令要进行操作的智能模块所在的 I/O 槽对应的 I/O 定义号，I/O 定义号可以是 00, 10, 20……150, 700, 710, ……760。使用 F3X, F4X 命令时，作用的对象是由 F29 指令所指定的 I/O 槽上所插的智能模块，所以编程时使用 F3X, F4X 命令时，必须注意确定是对所选的模块进行操作。如果系统中只使用了一个智能模块，那么即使不使用 F29 指令，PLC 也能自动确定智能模块的 I/O 定义号。

如果用 F29 指令时，所选的 I/O 定义号上的模块不是智能模块，那么 SR21/SR22/SE-22 在语法检查时，将会发生 I/O 选择错的信息 E02。

F30, F31, F32, F33 是向智能模块传送控制命令和参数的命令，一般传送给智能模块的控制命令有下列格式：

控制命令码 [控制参数]

其中控制参数并非每个控制命令都需要，哪个命令需要控制参数，控制参数需要哪些格式，要根据智能模块的要求来定，这一点可以参考专用的智能模块说明书。

F30~F33 指令的命令格式相同，它是一条双字节的指令。第一个字是 F30, F31, F32, F33 命令码，后面跟一个字是 0~255 之间的常数，命令的功能分别说明如下：

F30: 将常数作为控制命令码传送给智能模块。

F31: 将常数作为控制命令码传送给智能模块，同时将累加器中低 2 位 BCD 码作为参数传送给智能模块。

F32: 将常数作为控制命令码传送给智能模块，同时将累加器中 4 位 BCD 码作为参数传送给智能模块。

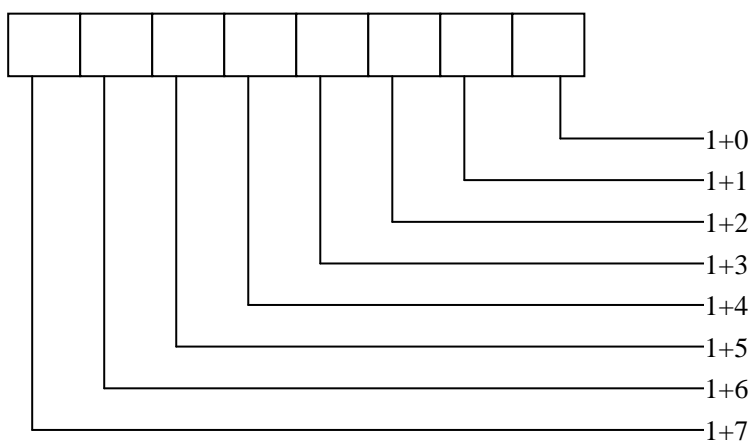
F33: 将常数作为控制命令码传送给智能模块，同时将累加器中的 4 位 BCD 码和辅助累加器中的 4 位 BCD 码作为参数传送给智能模块，其中，辅助累加器中的 4 位 BCD 码是高 4 位；累加器中的 4 位 BCD 码是低 4 位。

对于 SR21/SR22/SE-22 的 CPU 来说，有下列规定：

1. 控制命令码为 0~255 之间的常数，CPU 把大于 255 的数看作 255，请用户注意。
2. 控制参数可以是 0 个，2 个，4 个，8 个 BCD 码来表示，SR21/SR22/SE-22 不检查传送的内容是否为 BCD 码，如果智能模块约定容许参数是二进制的的话，也可以传送二进制数。
3. 如果传送的参数是正数，那么可以根据需要传送的参数位数来决定参数长度，例如对同一个命令来说，有时参数超过 10000，而有时参数小于 100，超过 10000 时，要采用 8 个 BCD 码的传送方式；而小于 100 时，只要用 2 个 BCD 码的传送方式；如果参数为 0 时，可以用无参数的传送方式。
4. 如果智能模块能接受负数，关于负数的格式和传送方法由智能模块的使用说明规定，采用相应的表达方式即可。

命令	传送内容	参数 BCD 码位数
F30	<div style="border: 1px solid black; padding: 2px; display: inline-block;">命令码</div>	0
F31	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">累加器</div> <div style="border: 1px solid black; padding: 2px; display: flex; align-items: center;"> <div style="border: 1px solid black; width: 40px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; width: 40px; height: 20px; margin-right: 5px; text-align: center;">//////</div> </div> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 2px; display: inline-block;">命令码</div> </div> </div>	2
F32	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">累加器</div> <div style="border: 1px solid black; padding: 2px; display: flex; align-items: center;"> <div style="border: 1px solid black; width: 40px; height: 20px; margin-right: 5px; text-align: center;">//////</div> <div style="border: 1px solid black; width: 40px; height: 20px; margin-right: 5px; text-align: center;">//////</div> </div> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 2px; display: inline-block;">命令码</div> </div> </div>	4
F33	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">累加器</div> <div style="border: 1px solid black; padding: 2px; display: flex; align-items: center;"> <div style="border: 1px solid black; width: 40px; height: 20px; margin-right: 5px; text-align: center;">//////</div> <div style="border: 1px solid black; width: 40px; height: 20px; margin-right: 5px; text-align: center;">//////</div> </div> <div style="text-align: center;"> <div style="border: 1px solid black; padding: 2px; display: inline-block;">命令码</div> </div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 10px;"> <div style="text-align: center;">辅助累加器</div> <div style="border: 1px solid black; padding: 2px; display: flex; align-items: center;"> <div style="border: 1px solid black; width: 40px; height: 20px; margin-right: 5px; text-align: center;">//////</div> <div style="border: 1px solid black; width: 40px; height: 20px; margin-right: 5px; text-align: center;">//////</div> </div> </div>	8

F40, F41, F42 命令是从智能模块读取返回的状态、数据的命令，从智能模块返回的状态和数据为下列格式：



返回数据也可以选择没有，也可以选择为 16 位二进制（4 位 BCD 码）或 32 位二进制数（8 位 BCD 码）。

高位部分	低位部分
------	------

辅助累加器 累加器

当选择 16 位二进制时，返回数据存放到累加器中，当选择 32 位二进制时，高 16 位放在辅助累加器中，低 16 位放在累加器中，返回数位的格式由智能模块规定。

F40, F41, F42 命令的功能说明如下：

F40: 将返回状态的 8 个开关量分别送到定义号由 F29 规定的 8 个触点中，用户可以象输入点一样来使用这些触点。

F41: 除了 F40 的功能以外，将返回数据的 16 位二进制数送到累加器中。

F42: 除了 F40 的功能以外，将返回数据的 32 位二进制数送到累加器和辅助累加器中。

智能模块与 SR21/SR22/SE-22 之间，相互通过这几组指令沟通信息，各种命令和返回数据的含义及格式由各种智能模块规定。

如果 PLC 未安装智能模块而用户用 F3X, F4X 指令时，将会出现 E09 错误信息。

指令输入方法。



I/O 定义号



常数（智能模块命令码）



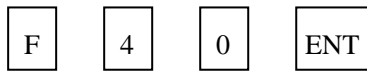
常数（智能模块命令码）



常数（智能模块命令码）



常数（智能模块命令码）



光洋电子(无锡)有限公司

Koyo ELECTRONICS (WUXI) CO., LTD.

地址：江苏省无锡市滨湖区建筑西路 599 号 1 栋 21 层

邮编：214072

电话：0510-85167888

传真：0510-85161393

<http://www.koyoele.com.cn>

KEW-M1232A

2015 年 8 月