



DVP-PM 系列

应用技术手册

DVP-PM 应用技术手册



中达电通股份有限公司
地址: 上海市浦东新区民夏路238号
邮编: 201209
电话: (021) 5863-5678
传真: (021) 5863-0003
网址: <http://www.deltagreentech.com.cn>
<http://中达电通>
服务热线: (021) 5863-9595

中达电通公司版权所有
如有改动, 恕不另行通知
PLC11B201105

DVP-PM 应用技术手册

目 录

第 1 章：PM 程序架构说明

1.1	O100 主程序架构	1-1
1.1.1	O100 主程序手动功能运动方式	1-2
1.2	OX 运动子程序架构	1-2
1.3	Pn 子程序架构	1-5
1.4	O100、OX、Pn 整体程序区块规划之架构	1-7

第 2 章：硬件规格及配线方式

2.1	硬件规格	2-1
2.1.1	电源规格	2-1
2.1.2	输入/输出点电气规格	2-1
2.1.3	外观尺寸规格	2-4
2.2	配线方式	2-7
2.2.1	盘内安装配线	2-7
2.2.2	电源输入配线	2-7
2.2.3	安全回路配线	2-7
2.2.4	输入/输出接点配线	2-8
2.2.5	与下位驱动器之配线	2-15
2.3	通讯口说明	2-25
2.3.1	COM1 (RS-232) 规格	2-26
2.3.2	COM2 (RS-485) 规格	2-26
2.3.3	COM3 (RS-232/RS-485)	2-26

第 3 章：PM 各种装置功能(含特 D 特 M 说明)

3.1	PM 各装置编号一览表	3-1
3.2	数值、常数[K、H]、浮点数[F]	3-5
3.3	外部输入/输出接点的编号及功能[X、Y]	3-8
3.4	辅助继电器的编号及功能[M]	3-10
3.5	内部继电器的编号及功能[S]	3-11
3.6	定时器的编号及功能[T]	3-12
3.7	计数器的编号及功能[C]	3-13

3.8 缓存器的编号及功能[D]、[V]、[Z].....	3-20
3.8.1 数据缓存器[D].....	3-20
3.8.2 间接指定缓存器[V]、[Z].....	3-20
3.9 指标[N]、指标[PN]	3-22
3.10 特殊继电器及特殊缓存器	3-23
3.11 特殊继电器及特殊缓存器功能说明	3-39
3.12 手动运动模式特殊缓存器	3-58
3.12.1 手动运动模式特殊缓存器功能说明	3-60
3.12.2 手动运动模式程序介绍	3-82
3.12.3 手动运动模式对应使用位置、速度控制缓存器一览表.....	3-83

第 4 章：基本顺序指令

4.1 基本指令一览表.....	4-1
4.2 基本指令说明	4-3

第 5 章：应用指令分类及基本使用

5.1 应用指令一览表.....	5-1
5.2 应用指令的组成.....	5-5
5.3 应用指令对数值的处理方式.....	5-9
5.4 使用间接指定缓存器 V、Z 来修饰操作数.....	5-12
5.5 指令索引.....	5-13
5.5 应用指令.....	5-18
● (API 00~09) 回路控制	5-18
● (API 10~19) 传送比较	5-19
● (API 20~29) 四则逻辑运算	5-24
● (API 40~49) 数据处理	5-37
● (API 70~79) 外部设定显示	5-40
● (API 100~109) 台达变频器通讯	5-44
● (API 110~138) 浮点运算.....	5-53
● (API 215~223) 比较接点指令.....	5-79
● (API 250~260) 新增指令	5-85

第 6 章：运动指令与 G-Code 分类基本使用说明

6.1 运动指令及 G-Code 指令一览表	6-1
6.2 运动指令及 G-Code 指令的组成	6-3

6.2.1 运动指令	6-3
6.2.2 G-Code 指令	6-4
6.3 运动指令	6-8
6.4 O 指针 / M 码指令	6-48
6.5 G-CODE 指令	6-52

第 7 章：PM 从机使用说明

7.1 主站 (Master) 与 PM 从站 (Slave) 联机控制方式	7-1
7.1.1 架构说明	7-1
7.1.2 主站与从站联机控制范例说明	7-1

第 8 章：应用范例

8.1 使用运动指令与 G-Code 绘制轨迹	8-1
8.1.1 轨迹图	8-1
8.1.2 设计操作步骤	8-3
8.2 PMSOFT 软件「MOTIONSAMPLE」应用范例	8-7
8.2.1 设计架构说明	8-7
8.2.2 设计范例程序	8-8
8.3 自行规划多段变速度	8-11
8.3.1 设计架构说明	8-11
8.3.2 设计范例程序	8-11
8.4 DVP-PM 连接 DVP01PU-H2 第三轴控制 (仅供用于 DVP20PM00D 机种)	8-15

第 9 章电子凸轮

9.1 电子凸轮(E-CAM)简介	9-1
9.2 电子凸轮的实现方式	9-2
9.2.1 初始设定	9-2
9.2.2 主轴讯号来源设定	9-6
9.2.3 电子凸轮启动/停止	9-10
9.3 电子凸轮功能缓存器	9-15
9.4 电子凸轮数据(E-CAM Data) 建立	9-20
9.4.1 使用 PMsoft CAM Chart 建立电子凸轮数据	9-20
9.4.2 使用 DTO / DFROM 指令自动生成电子凸轮数据	9-25
9.5 多凸轮模式	9-47
9.6 电子凸轮实际应用	9-51

9.6.1 高速绕线机机构及动作说明	9-51
9.6.2 飞剪应用	9-56
9.6.3 追剪应用	9-60

第 10 章程序加密设定

10.1 使用设定介绍	10-1
10.1.1 系统信息区	10-1
10.1.2 下载程序设定	10-3
10.1.3 上传程序设定	10-6

第 11 章G-Code 应用

11.1 程序下载(PEP 保护)说明	11-1
11.2 G-Code 下载方法	11-2
11.2.1 PMGDL 软件下载	11-2
11.2.2 使用台达 B Type HMI 下载	11-8
11.2.3 D Register 转 G-Code	11-12
11.3 点胶模式使用设定介绍	11-16
11.3 G/M code 文件数据储存方式	11-20

第 12 章 PMsoft POU

12.1 POU 阶梯图编辑模式环境	12-2
12.1.1 符号表功能	12-2
12.1.2 符号表编辑与符号宣告	12-5
12.1.3 新增 POU 功能块	12-13
12.1.4 新增 POU 管理目录	12-16
12.1.5 汇出 POU 功能块	12-22
12.1.6 汇入 POU 功能块	12-28
12.1.7 系统装置配置表	12-32
12.1.8 阶梯图寻找	12-34
12.1.9 阶梯图取代	12-37
12.2 POU 阶梯图程序操作	12-39
12.2.1 符号取代装置	12-39
12.2.2 功能块 FB 使用	12-40
12.3 POU 在线监控	12-45
12.4 符号或功能块自动显示批注(HINT)	12-49

第 13 章 FPMC 通讯卡

13.1	CANOPEN 通讯功能卡 (DVP-FPMC) 简介	13-1
13.2	功能规格	13-1
13.3	产品外观及安装	13-2
13.4	CR 参数功能介绍	13-3
13.5	控制缓存器 CR 内容说明	13-6
13.6	FPMC 之模式设定	13-23
13.7	FPMC 之 ETHERNET 模式介绍	13-28
13.7.1	ETHERNET 联机介绍 - FPMC 与 HMI 做数据传递范例	13-28
13.7.2	FPMC 与 PMSOFT 联机介绍	13-31
13.8	LED 灯指示说明及故障排除	13-33

第 14 章 高速比较与捕捉功能说明

14.1	高速比较与捕捉功能	14-1
14.2	高速比较	14-4
14.3	捕捉功能	14-7

第 15 章 附录

15.1	附录 A 侦错码原因对照表	15-1
15.2	附录 B 手册修改记录	15-2
15.3	附录 C PM 各机种支持菜单	15-2

台达 DVP-PM 系列主机为高速定位、双轴线性及圆弧补间多功能可编程控制器，其特色是结合了基本指令、应用指令、运动指令及 G-Code 指令等功能，让程序编写设计上更多元化。

本章节主要介绍 DVP-PM 系列主机的程序架构，由于 DVP-PM 主机结合了 PLC 顺序控制及双轴补间定位控制的功能，因此在程序的规划上，主要分为 O100 主程序、OX 运动子程序及 Pn 子程序等三大类，本章节将以此三大类的程序架构进行说明。基本指令、应用指令、运动指令及 G-Code 指令将在第 4~6 章节作详细介绍说明，本章节将不多作说明。下表为各厂牌 PLC 规格比较：

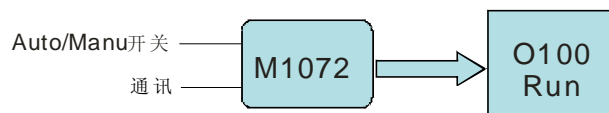
规格	DVP-10PM	AH	永宏 PLC	DVP-20PM
高速输出	1000kHz 4 组	?	920kHz	500kHz 3 组
PWM	精度 0.3% @ 200KHz	?	精度 1%	
硬件高速计数器	6 组(差分 x2, 开集极 x4)	?	4 组	2 组
程序容量	64KSteps	?	20Kstep	64KSteps
执行速度	LD 0.14us MOV 2us DMUL 7.6us DEMUL 6.1us	LD 0.13us MOV 1us	LD 0.33us	

1.1 O100 主程序架构

O100 主程序为 PLC 顺序控制程序，主要为 DVP-PM 系列主机运作的顺序控制主程序。O100 主程序区段中，只支持基本指令及应用指令，使用这两种类型指令，除了提供 I/O 点讯号数据处理及 Pn 子程序的呼叫，还有控制启动 OX0 ~ OX99 共 100 组的 OX 运动子程序，所以 O100 主程序为主控程序的建立，再经由主控程序去设定及启动运动子程序，这是 DVP-PM 运作控制架构阶层关系，下列将对 O100 主程序运作流程及特性说明。

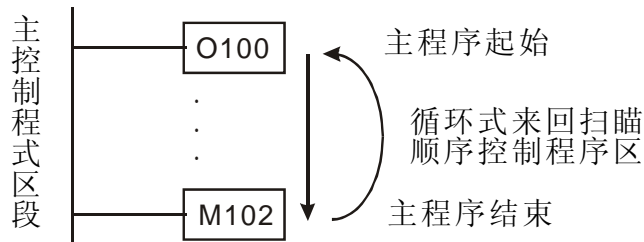
1. 启动 O100 主程序的方法有两种：

- 当上电状态中，DVP-PM 主机上的 AUTO/MANU 开关，由 MANU → AUTO 时，M1072 旗标将自动为 On，而 O100 主程序为 RUN 状态。
- 当上电状态中，可利用通讯的方式，将 M1072 旗标设定为 On，也可使 O100 主程序为 RUN 状态。



- ### 2. 其程序扫描方式为循环式来回扫描，当 O100 主程序启动后，将由 O100 起始旗标进行扫描，扫描到 M102 主程序结束指令时，再重新返回 O100 起始旗标持续的扫描，如下图：

1 PM 程序架构说明



3. O100 主程序停止运作方式有三种，说明如下：

- 当上电状态中，DVP-PM 主机上的 AUTO/MANU 开关，由 AUTO → MANU 时，M1072 旗标将自动为 Off，而 O100 主程序为 STOP 状态，此时 OX 及 Pn 子程序也会停止运作。
- 当上电状态中，可利用通讯的方式，将 M1072 旗标设定为 Off，使 O100 主程序为 STOP 状态，此时 OX 及 Pn 子程序也会停止运作。
- 当主程序于程序设计编译或程序运作出现错误时，O100 主程序也会自动停止运作。请参考第 14 章附录之侦错码原因对照表！

4. O100 主程序有支持基本指令及应用指令两种，因此可依实际需要来设计控制程序，并在主程序中设定运动指令的参数，更可设定运动程序启动编号，进而启动 OX0~OX99 运动子程序。

- O100 主程序不支持运动指令及 G-Code 指令，因此请将运动指令及 G-Code 规划在 OX0~OX99 运动子程序之中，请参考第 1.2 章节说明。
- O100 主程序可呼叫 Pn 子程序，请参考第 1.3 章节之说明。

5. 上述说明整理如下表：

O100 主程序	说明
程序起始	O100 主程序起始旗标 (PMSoft 阶梯图编辑模式下会自动设定，使用者不需撰写)
程序结束	M102 主程序结束指令 (PMSoft 阶梯图编辑模式下会自动设定，使用者不需撰写)
程序执行	1. DVP-PM 主机上的 AUTO/MANU 开关, MANU → AUTO 状态 2. 利用通讯将 M1072 Off → On
运作特性	循环式来回扫描方式及运作
支持指令	支持基本指令、应用指令，共二种
程序数量	于程序中，O100 主程序只能有一组
特色及功能	1. 为 PLC 顺序控制程序 2. 可启动 OX0 ~ OX99 运动子程序及呼叫 Pn 子程序 3. 与 OX0 ~ OX99 运动子程序及 Pn 子程序搭配使用时，3 组程序顺序可任意排列

1.1.1 O100 主程序手动功能运动方式

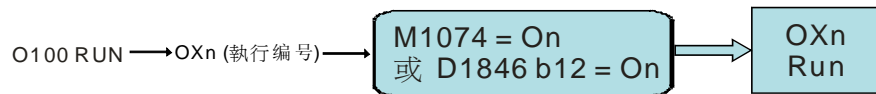
在 O100 主程序内，使用者可利用特殊缓存器，自行规画手动功能运动模式(详细设定方法请参考 3.12 章节)

1.2 OX 运动子程序架构

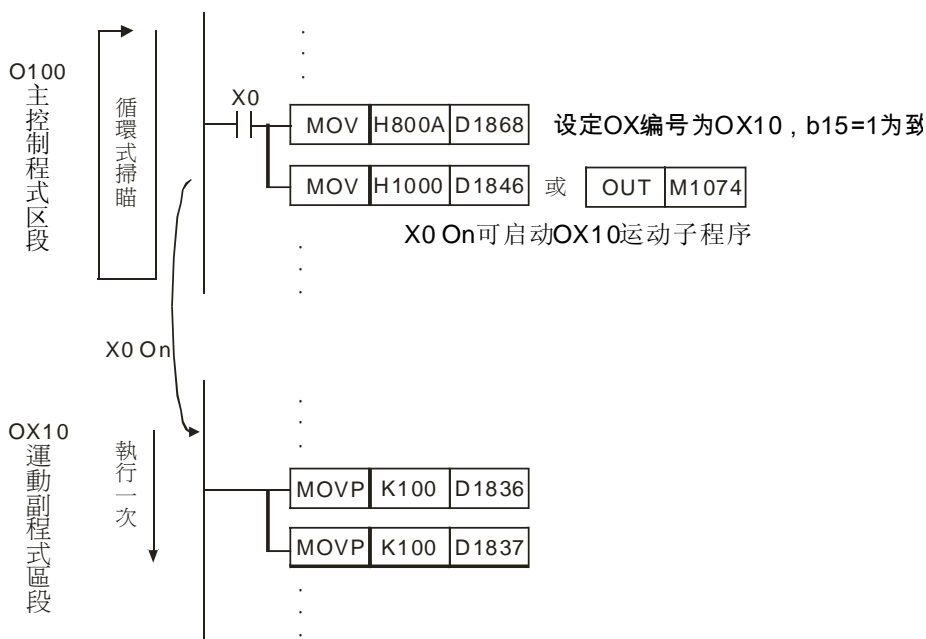
OX0 ~ OX99 运动子程序为运动控制程序，主要为控制 DVP-PM 系列主机进行运动轴运动之子程序。OX0 ~ OX99 运动子程序区段中，有支持基本指令、应用指令、运动指令及 G 码指令，并在程序中可规划呼叫 Pn 子程序。主要提供使用者编写设计控制运动轴移动路径程序，下列将对 OX 运动子程序运作流程及特性说明。

1. 启动 OX0 ~ OX99 运动子程序的方法如下：

- 当 O100 主程序为 RUN 状态下,可在 O100 主程序中设定 OX 执行编号 00~99 (D1868:H8000 ~ H8063) 后,再启动 OX 子程序旗标 M1074 设定为 On 或设定运动轴运转命令 (D1846)的 b12 为 On,即可启动 OX 运动子程序。
- 当启动 OX 运动子程序时,请确定无任何运动子程序正在运转中,才可进行启动。



2. 其程序扫描方式为每启动一次就执行一次，当 O100 主程序启动 OX 运动子程序后，将由 OX 运动子程序的起始旗标进行执行，执行到 M2 运动子程序结束指令时，即结束运动子程序，如下图所示：



上图中当 X0 On 时，将启动 OX10 运动子程序，执行到 M2 运动子程序结束指令时，即停止（只执行一次，如须再执行可重新启动 X0 开关，即可重新启动 OX10 运动子程序）。

3. OX 运动子程序停止运作方式有四种，说明如下：

- 当上电状态中，DVP-PM 主机上的 AUTO/MANU 开关，由 AUTO → MANU 时，M1072 旗标将自动为 Off，而 O100 主程序为 STOP 状态，OX 运动子程序也将停止运作。
- 亦可由外部控制端子 (STOP0) 输入信号，控制 OX 运动子程序停止运作。
- 当上电状态中，可利用通讯的方式，将 D1846 设定为 0 或 M1074 设为 Off，也可使 OX 运动子程序停止运作。

1 PM 程序架构说明

- 当运动子程序于程序设计编译或程序运作出现错误时，OX 运动子程序也会自动停止运作。请参考第 14 章附录之侦错码原因对照表！
4. OX 运动子程序支持基本指令、应用指令、运动指令及 G-Code 指令四种，因此可依需求使用这四种指令来设计运动程序，及设定运动轴参数，以达到运动轴运动控制。
- 请将上述四种指令规划在 OX0~OX99 运动子程序之中！
 - OX 运动子程序有支持 Pn 子程序的呼叫，请参考第 1.3 章节之说明！
5. 上述说明整理如下表：

OX 运动子程序	说明
程序起始	OX 运动子程序 (OX0 ~ OX99, 共 100 组运动子程序) (PMSoft 阶梯图编辑模式下会自动设定, 使用者不需撰写)
程序结束	M2 运动子程序结束 (PMSoft 阶梯图编辑模式下会自动设定, 使用者不需撰写)
程序执行	1. 由 O100 主程序为 RUN 状态中, 设定 D1846_b12 为 1 或 M1074 = On, 即可启动 OX 运动子程序 2. 由 O100 主程序为 RUN 状态中, 利用通讯设定 D1846_b12 为 1 或 M1074 = On, 亦可启动 OX 运动子程序 3. 可由外部控制端子 (STOP0) 输入信号, 控制 OX 运动子程序停止 注意: 当启动 OX 运动子程序时, 请确定无任何运动子程序正在运转中, 才可进行启动。
运作特性	每启动一次只执行一次, 若须再次执行可再重新启动一次即可
支持指令	支持基本指令、应用指令、运动指令、G-Code 指令, 共四种 注意: 使用基本指令、应用指令时, 请避免使用脉波型式指令
程序数量	于程序中, OX 运动子程序能有 100 组, 若须启动其它 OX 编号, 可设定 D1868 后, 再启动 (设定 D1846_b12 为 1 或 SET M1074) 即可
特色及功能	1. 为运动子程序 (只能由 O100 主程序设计程序来启动) 2. 提供第三轴 (Z 轴) 控制, 请参考第 6.4 章节之 G00 及 G01 指令! 3. 提供外部控制端子、程序设计及通讯方式, 来启动 / 停止 OX 运动子程序 4. 可呼叫 Pn 子程序 5. 与 O100 主程序及 Pn 子程序搭配使用时, 3 组程序顺序可任意排列

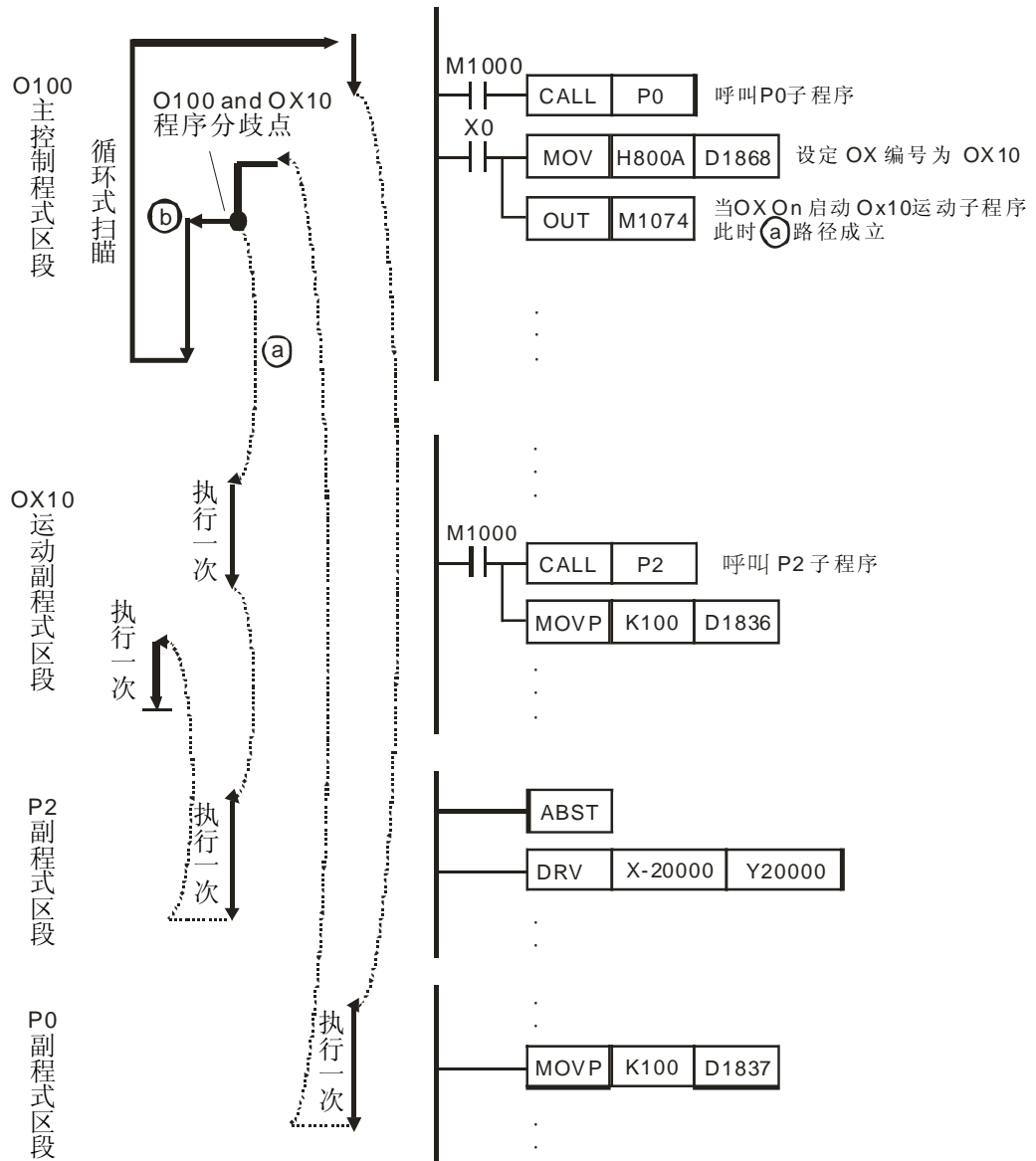
1.3 Pn 子程序架构

Pn 子程序为一般用子程序, 主要为提供 O100 主程序及 OX 运动子程序之呼叫子程序使用。于 O100 主程序呼叫 Pn 子程序时, 则 Pn 子程序区段支持基本指令及应用指令; 若于 OX0 ~ OX99 运动子程序呼叫 Pn 子程序时, 则 Pn 子程序区段支持基本指令、应用指令、运动指令及 G-Code。而不论在 O100 主程序或 OX 运动子程序中呼叫 Pn 子程序, 都会在 Pn 子程序执行时, 跳至 Pn 子程序进行动作, 执行到 SRET 时将返回 CALL Pn 指令的下一行继续执行。

1. 启动 Pn 子程序的方法如下：

- 于 O100 主程序中呼叫 Pn 子程序, 即可启动 Pn 子程序。

- 于 OX 运动子程序中呼叫 Pn 子程序，即可启动 Pn 子程序。
2. 其程序扫描方式为呼叫一次只执行一次，O100 主程序中呼叫 Pn 子程序后，将执行 Pn 子程序，执行到 SRET 子程序结束指令时，即结束 Pn 子程序，并回到 Pn 指针处的下一行程序继续执行；当由 OX 运动子程序呼叫 Pn 子程序后动作亦同，如下图之范例程序说明：



其中 P0 子程序区段中，可自由的编写基本指令及应用指令；于 P2 子程序区段中，可自由的编写基本指令、应用指令、运动指令及 G-Code 指令。

3. Pn 子程序停止运作方式有三种，说明如下：

- 当上电状态中，DVP-PM 主机上的 AUTO/MANU 开关，由 AUTO → MANU 时，M1072 旗标将自动为 Off，而 O100 主程序为 STOP 状态，OX 子程序及 Pn 子程序都将停止运作。
- 当上电状态中，可利用通讯的方式，将 D1846 设定为 0 或 M1074 设为 Off，也可使 OX 运动子程序停止运作，则所呼叫的 Pn 子程序也停止运作。

1 PM 程序架构说明

- 当 Pn 子程序于程序运作出现错误时，Pn 子程序也会自动停止运作。请参考第 14 章附录之侦错码原因对照表！
4. 于 O100 主程序呼叫 Pn 子程序时，则 Pn 子程序只支持基本指令及应用指令；若于 OX0 ~ OX99 运动子程序呼叫 Pn 子程序时，则 Pn 子程序支持基本指令、应用指令、运动指令及 G-Code 指令。
 5. 上述说明整理如下表：

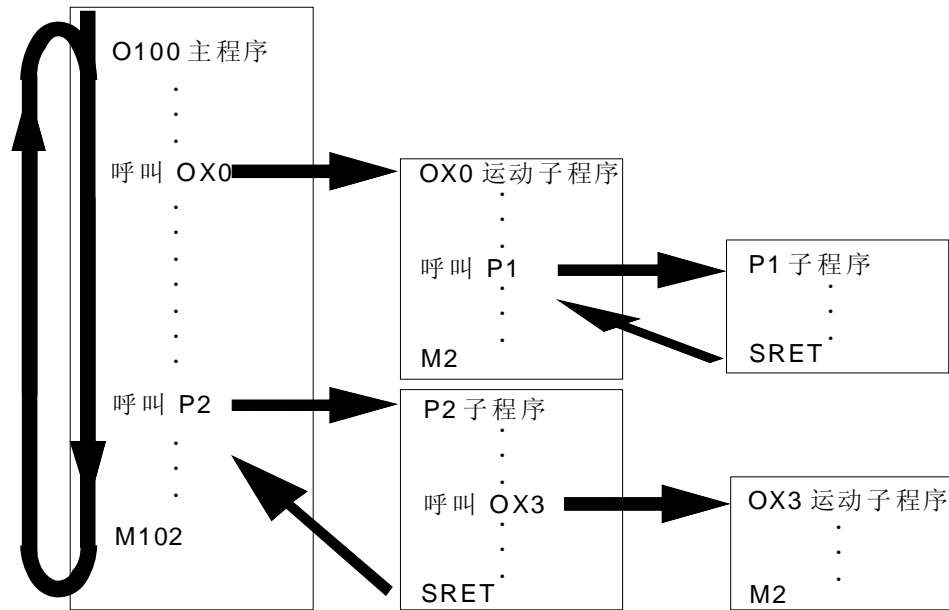
Pn 子程序	说明
程序起始	Pn 子程序起始旗标 (P0 ~ P255) (PMSoft 阶梯图编辑模式下会自动设定，使用者不需撰写)
程序结束	SRET 子程序结束 (PMSoft 阶梯图编辑模式下会自动设定，使用者不需撰写)
程序执行	1. 由 O100 主程序中，呼叫 Pn 子程序 2. 由 OX 运动子程序中，呼叫 Pn 子程序
运作特性	每呼叫一次只执行一次，若须再次执行可再重新呼叫一次即可
支持指令	1. 由 O100 呼叫时：支持基本指令及应用指令，共二种 2. 由 OX 呼叫时：支持基本指令、应用指令、运动指令及 G-Code 指令，共四种 注意：由 OX 呼叫且使用基本指令、应用指令时，请避免使用脉波型式指令。
程序数量	程序中，Pn 子程序能有 256 组
特色及功能	1. 为一般用子程序 2. 提供给 O100 主程序及 OX 运动子程序呼叫用 3. 与 O100 主程序及 OX 运动子程序搭配使用时，3 组程序顺序可任意排列

1.4 O100、OX、Pn 整体程序区块规划之架构

第 1.1~1.3 章节中，介绍到 O100 主程序、OX 运动子程序及 Pn 子程序。在本章节中，将介绍此三者的混用架构，及设计的技巧。

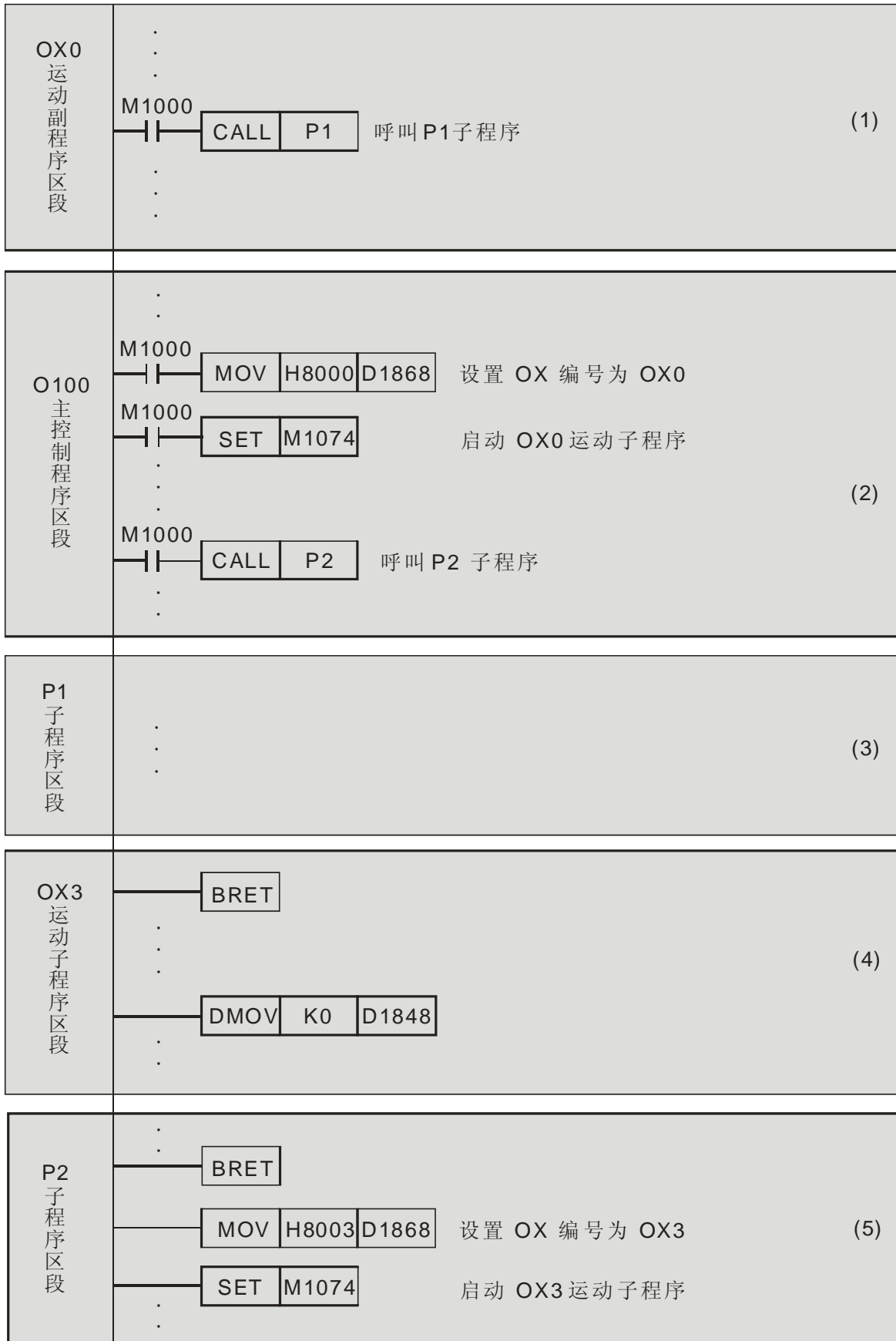
1.4.1 程序架构介绍

假设计 O100 主程序、OX0 运动子程序、OX3 运动子程序、P1 子程序及 P2 子程序，共分成 5 个程序区块，希望设计流程方块如下：



为了方便说明范例程序，故分成 (1) ~ (5) 区段，来进行说明，并将程序规划如下：

1 PM 程序架构说明



范例程序设计说明：

1. 范例程序中的编写顺序是由 (1) 到 (5) 的方式，而在排列顺序上是可以随意的摆放，而没有硬性的规则。
2. O100 主程序 (2) 只能有一组，且不可被其它程序呼叫，但可自由呼叫 OX 运动子程序及 Pn 子程序。
3. OX 运动子程序可被 O100 主程序及 Pn 子程序呼叫，并且可以呼叫 Pn 子程序。
4. Pn 子程序可被 O100 主程序及 OX 运动子程序呼叫，并且可以呼叫 OX 运动子程序。

注意事项：

1. 不可同时执行两个以上的 OX 运动子程序，故当 OX0 运动子程序执行时，则 OX3 运动子程序无法运作；反之，当 OX3 运动子程序执行时，则 OX0 运动子程序无法运作。
2. 由 O100 主程序或 Pn 子程序启动 OX 运动子程序后，就继续执行下一行程序，而不理会 OX 运动子程序。
3. OX 运动子程序启动后，只执行一次，若须再行运作，可重新启动，即可再动作一次。

针对前述之范例，其各程序区段所支持之指令，说明如下表：(O: 有支援, X: 不支援)

区段	O100 主程序	OX 运动子程序 (OX0、OX3)	P1 子程序	P2 子程序
基本指令	O	O	O	O
应用指令	O	O	O	O
运动指令	X	O	O	X
G-Code	X	O	O	X
说明	支持指令为内定	支持指令为内定	由 OX 运动子程序呼叫，故有支持运动指令、G-Code	由 O100 主程序呼叫，故不支持运动指令、G-Code

补充说明：

	主程序	子程序	运动程序
摆放顺序	任意	任意	任意
程序执行	正常 RUN	主程序或运动程序呼叫	主程序或子程序呼叫
运作特性	循环运作	呼叫一次执行一次	呼叫一次执行一次
数量	1	256 个，使用者需求使用	100 个，使用者需求使用

2.1 硬件规格

本章节仅提供电气规格、安装配线说明，其它详细之程序设计及指令说明，请参考 DVP-PM 应用技术手册第 5 ~ 6 章节，选购之周边装置详细说明，请参考产品随机手册。

2.1.1 电源规格

项目	20PM	10PM
电源电压	100~240VAC (-15%~10%), 50/60Hz ± 5%	
电源保险丝容量	2A/250VAC	
消耗功率	60VA	
DC24V 供应电流	500mA	
电源保护	DC24V 输出具短路保护	
突波电压耐受量	1500VAC (Primary-secondary), 1500VAC (Primary-PE), 500VAC (Secondary-PE)	
绝缘阻抗	5 MΩ 以上 (所有输出/入点对地之间 500VDC)	
噪声免疫力	ESD: 8KV Air Discharge, EFT: Power Line: 2KV, Digital I/O: 1KV, Analog & Communication I/O: 250V	
接地	接地配线之线径不得小于电源端 L, N 之线径 (多台 DVP-PM 同时使用时, 请务必单点接地)	
操作/储存环境	操作: 0°C~55°C (温度), 50~95% (湿度), 污染等级 2 储存: -25°C ~70°C (温度), 5~95% (湿度)	
耐振动/冲击	国际标准规范 IEC61131-2, IEC 68-2-6 (TEST Fc) /IEC61131-2 & IEC 68-2-27 (TEST Ea)	
重量 (约 g)	478/688	

2.1.2 输入/输出点电气规格

输入点电气规格

■ DVP20PM00D

端子	说明	响应特性	最大输入	
			电流值	电压值
START0、START1	启动输入	10ms	6mA	24V
STOP0、STOP1	停止输入	10ms	6mA	24V
LSP0 / LSN0、 LSP1 / LSN1	右极限输入/左极限输入(X-Y 轴)	10ms	6mA	24V
A0+、A0-、 A1+、A1-	手摇轮 A 相脉冲输入+,- (差动信号输入) (A1+、A1- 之 Y 轴与 Z 轴共享)	200kHz	15mA	5~24V
B0+、B0-、 B1+、B1-	手摇轮 B 相脉冲输入+,- (差动信号输入) (B1+、B1- 之 Y 轴与 Z 轴共享)	200kHz	15mA	5~24V
PG0+、PG0-、 PG1+、PG1-	零点讯号输入+,- (差动信号输入)	200kHz	15mA	5~24V
DOG0、DOG1	依运转模式不同有下列 2 种变化: 1. 原点复归时为近点信号 2. 一段速或二段速插入启动信号	1ms	6mA	24V
X0~X7	一般输入点	200kHz	15mA	24V

2 硬件规格及配线方式

■ DVP20PM00M

端子	说明	响应特性	最大输入	
			电流值	电压值
START0、START1	启动输入	10ms	6mA	24V
STOP0、STOP1	停止输入	10ms	6mA	24V
LSP0 / LSN0、 LSP1 / LSN1	右极限输入/左极限输入(X-Y 轴)	10ms	6mA	24V
X1 / X2	右极限输入/左极限输入(COM) (Z 轴)	10ms	6mA	24V
A0+、A0-、 A1+、A1-	手摇轮 A 相脉冲冲输入+,- (差动信号输入) (A1+、A1- 之 Y 轴与 Z 轴共享)	200kHz	15mA	5~24V
B0+、B0-、 B1+、B1-	手摇轮 B 相脉冲冲输入+,- (差动信号输入) (B1+、B1- 之 Y 轴与 Z 轴共享)	200kHz	15mA	5~24V
PG0+、PG0-、 PG1+、PG1-	零点讯号输入+,- (差动信号输入)	200kHz	15mA	5~24V
X3	零点讯号输入(COM) (用于 Z 轴)	10ms	6mA	24V
DOG0、DOG1	依运转模式不同有下列 2 种变化: 3. 原点复归时为近点信号 4. 一段速或二段速插入启动信号	1ms	6mA	24V
X0	同 DOG0、DOG1 (COM) (用于 Z 轴)	10ms	6mA	24V
X4~X7	一般输入点	200kHz	15mA	24V

■ DVP10PM00M

端子	说明	响应特性	最大输入	
			电流值	电压值
X0~X7	1. 单相/AB 相输入 2. X-Y-Z-A 四轴 DOG 讯号: X0、X2、X4、X6 3. X-Y-Z-A 四轴 PG 讯号: X1、X3、X5、X7	200kHz	15mA	24V
X10+、X10-、X11+、 X11-	手摇轮脉冲输入+,- (差动信号输入)	200kHz	15mA	5~24V
X12+、X12-、X13+、 X13-	差动信号计数输入	200kHz	15mA	5~24V

输出点电气规格

■ DVP20PM00D

端子	说明	响应特性	最大电流输出
CLR0+、CLR0-、 CLR1+、CLR1-	清除信号 (Servo 驱动器内部偏差计数器清除信号)	10ms	20mA
FP0+、FP0-、 FP1+、FP1-	正反转模式: 正转方向脉冲输出 脉冲/方向: 脉冲输出端 AB 相模式: A 相输出	500kHz	40mA
RP0+、RP0-、 RP1+、RP1-	正反转模式: 反转方向脉冲输出 脉冲/方向: 方向输出端 AB 相模式: B 相输出	500kHz	40mA
Y0~Y7	一般输出端子	200kHz	40mA

■ DVP20PM00M

端子	说明	响应特性	最大电流输出
CLR0+、CLR0-、 CLR1+、CLR1-	清除信号 (Servo 驱动器内部偏差计数器清除信号)	10ms	20mA
Y2	同清除信号 (用于 Z 轴)	10ms	30mA
FP0+、FP0-、 FP1+、FP1-、 FP2+、FP2-	正反转模式：正转方向脉冲输出 脉冲/方向：脉冲输出端 AB 相模式：A 相输出	500kHz	40mA
RP0+、RP0-、 RP1+、RP1-、 RP2+、RP2-	正反转模式：反转方向脉冲输出 脉冲/方向：方向输出端 AB 相模式：B 相输出	500kHz	40mA
Y3~Y7	一般输出端子	200kHz	40mA

■ DVP10PM00M

端子	说明	响应特性	最大电流输出
Y0 ~ Y3	开集极高速脉冲输出(PWM 输出)	200kHz	40mA
Y10+、Y10-、 Y12+、Y12-、 Y14+、Y14-、 Y16+、Y16-	正反转模式：正转方向脉冲输出 脉冲/方向：脉冲输出端 AB 相模式：A 相输出	1000kHz	40mA
Y11+、Y11-、 Y13+、Y13-、 Y15+、Y15-、 Y17+、Y17-	正反转模式：反转方向脉冲输出 脉冲/方向：方向输出端 AB 相模式：B 相输出	1000kHz	40mA

数字输入点

■ 20PM

规格	项目	24VDC 单端共点输入		备注
		低速	高速 200kHz #1	
输入接线型式	由端子 S/S 变换接线为 SINK 或 SOURCE			#1: 输入点 A, B, PG 为高速输入; 其余输入点为低速输入。 #2: 输入点 X0 ~ X7 可作 10 ~ 60 ms 数字滤波调整。
输入动作指示	LED 显示, 灯亮表示为'On', 不亮表示为'Off'			
输入信号电压	-			
动作位准	Off→On	20us		
	On→Off	30us		
反应时间/噪声抑制 #2	10ms / 0.5 us			

2 硬件规格及配线方式

■ 10PM

规格	项目		备注
	双端差动输入	24VDC 单端共点输入	
	高速 200kHz		
输入接线型式	独立接线	由端子 S/S 变换接线为 SINK 或 SOURCE	#1: 输入点可作 10 ~ 60 ms 数字滤波调整; 另外可作 20~85KHz 之高频数字滤波调整。
输入动作指示	LED 显示, 灯亮表示为'On', 不亮表示为'Off'		
输入信号电压	5~24 VDC	24 VDC	
最大电流输入值	15 mA		
动作位准	Off→On	20us	
	On→Off	30us	
反应时间/噪声抑制 #1	10ms / 0.5us		

数字输出点

■ 20PM

规格	项目		单端共点 #1 继电器输出
	低速	高速	
最高交换 (工作) 频率	10kHz	200kHz	负载 On/Off 控制使用
输出动作指示	LED 显示, 灯亮表示为'On', 不亮表示为'Off'		
最小负载	-		2mA/DC 电源
工作电压	5 ~ 30VDC		<250VAC, 30VDC
隔离方式	光耦合隔离		电磁性隔离
电流规格	电阻性	0.5A/1 点(2A/COM)	2A/1 点 (5A/COM)
	电感性	9W (24VDC)	#2
	灯泡	2W(24VDC)	20WDC/100WAC
最大输出延迟时间	Off→On	20us	0.2us
	On→Off	30us	
输出过电流保护	N/A		

■ 10PM

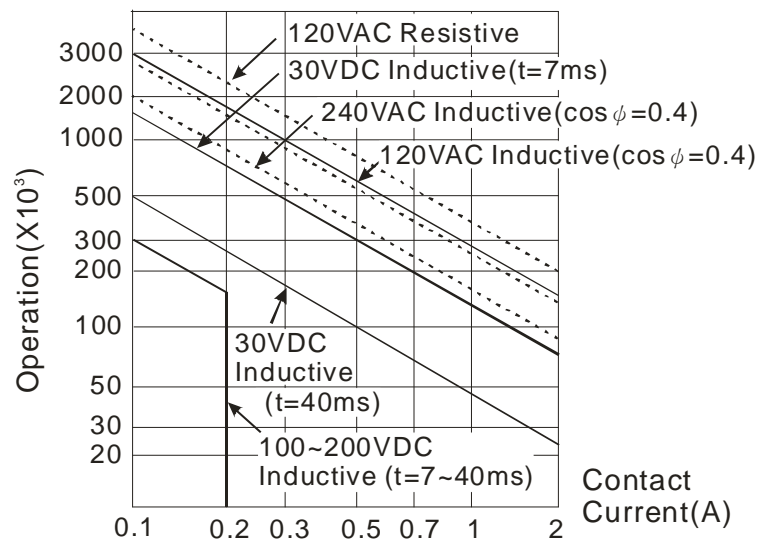
规格		项目	双端差动输出	单端共点晶体管输出
最高交换 (工作) 频率			1 MHz	200 kHz
输出动作指示			LED 显示, 灯亮表示为'On', 不亮表示为'Off'	
输出点配置			Y10 ~ Y17	Y0 ~ Y3
工作电压			5 VDC	5 ~ 30 VDC
最大输出电流值			40 mA	40 mA
隔离方式			驱动级电源隔离	光耦合隔离
电流规格	电阻性		< 25 mA	0.5 A/1 点 (4 A/COM)

2 硬件规格及配线方式

规格		项目	双端差动输出	单端共点晶体管输出
	电感性		--	12 W (24 VDC)
	灯泡		--	2 W (24 VDC)
最大输出 延迟时间	Off→On		0.2 us	
	On→Off			
输出过电流保护			N/A	

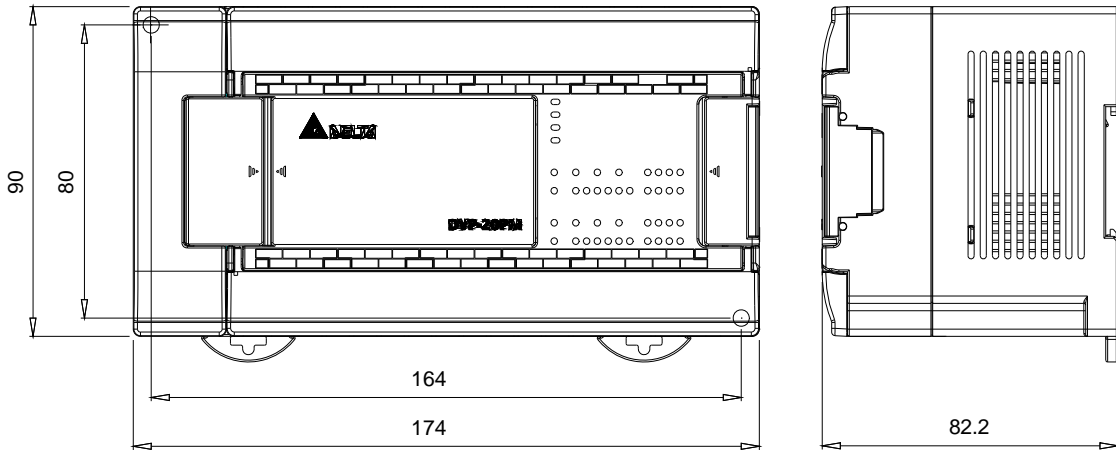
#1: DVP20PM00D 之(Y0 ~ Y7)为继电器输出; DVP20PM00M 之(FP2+, PF2-)为高速晶体管输出, (Y2, Y3)为低速晶体管输出, (Y4 ~ Y7)为继电器输出。Y0 回路共点 C0, Y1 回路共点 C1, Y2 与 Y3 回路共点 C2, Y4~Y7 回路共点 C3。

#2: 生命周期曲线图



2 硬件规格及配线方式

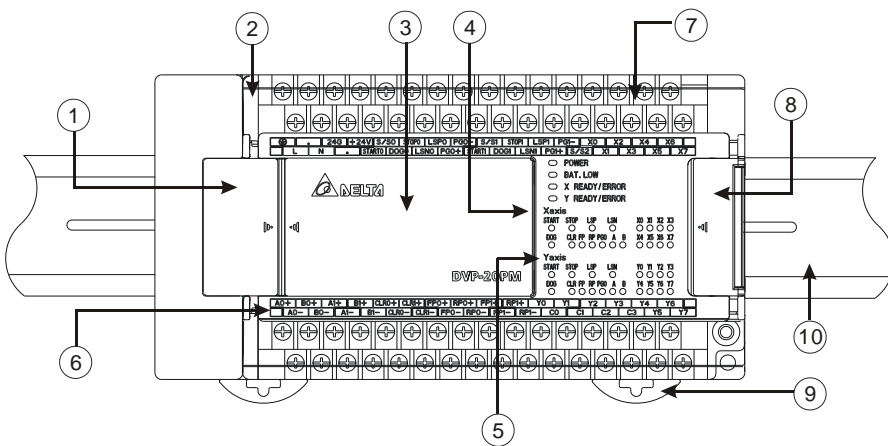
2.1.3 外观尺寸规格



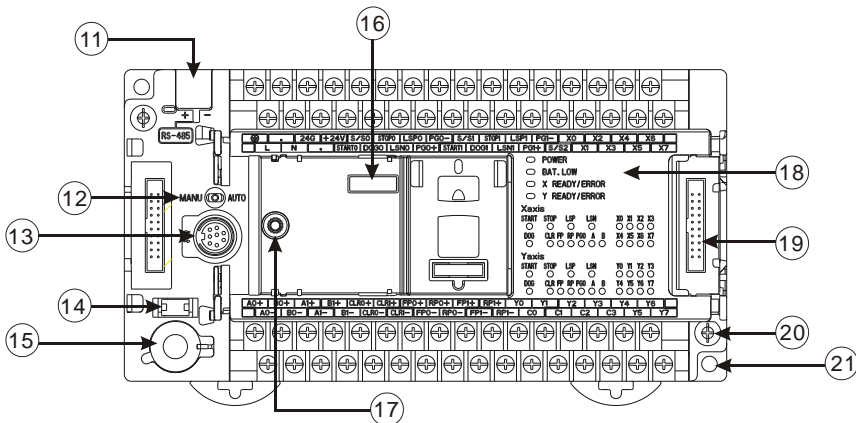
(单位: mm)

各部介绍

■ DVP-20PM

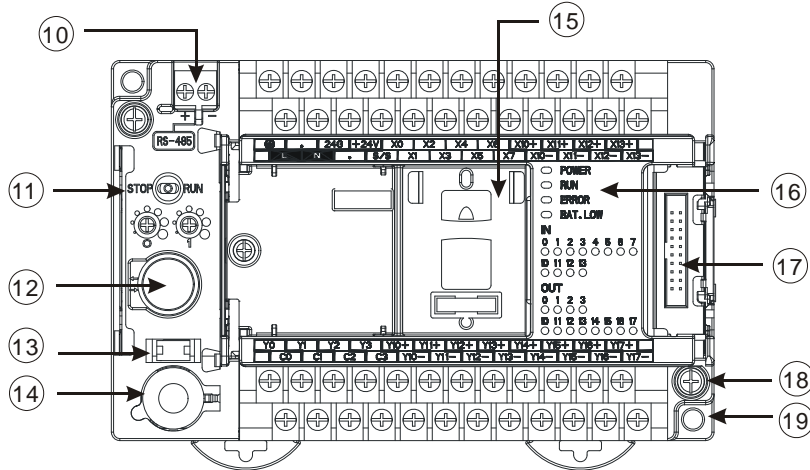
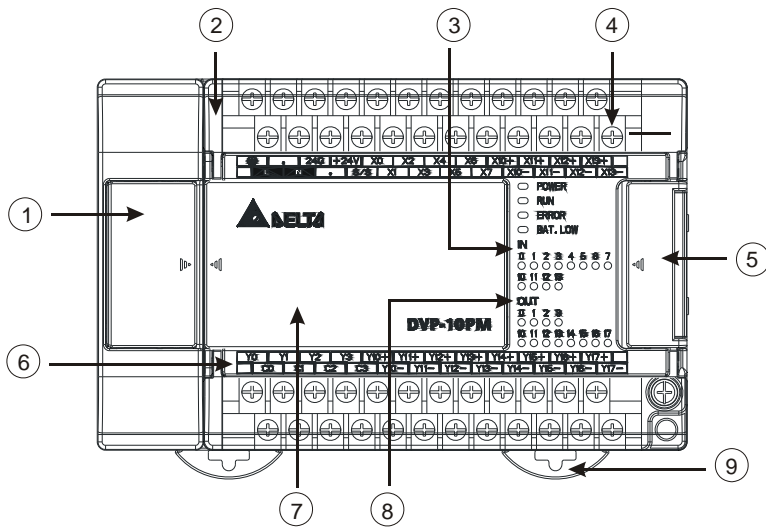


- ① 通讯口上盖
- ② 输出/入端子盖
- ③ 功能卡上盖
- ④ 输入点指示灯
- ⑤ 输出点指示灯
- ⑥ 输入/输出端子编号
- ⑦ 输入/输出端子
- ⑧ I/O 模块接口上盖



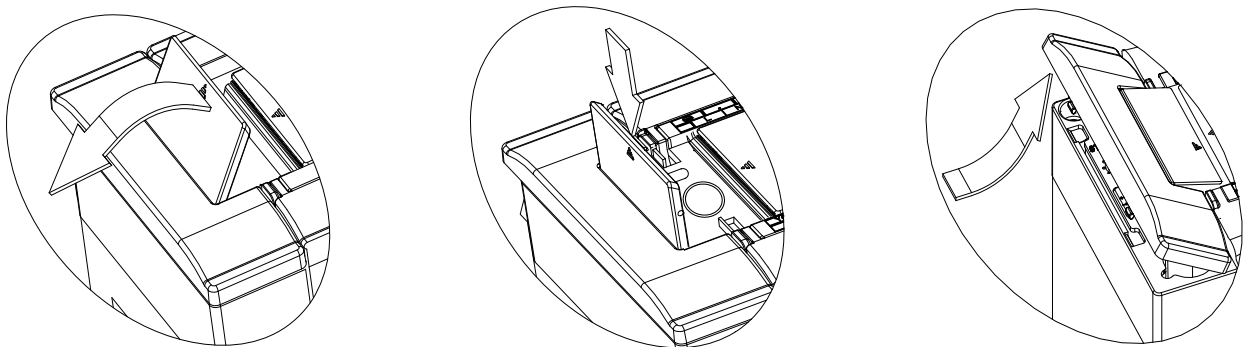
- ⑨ DIN 轨固定扣
- ⑩ DIN 轨 (35mm)
- ⑪ COM2(RS-485)
- ⑫ MANU/AUTO 开关
- ⑬ COM1 (RS-232)
- ⑭ 电池连接插槽
- ⑮ 电池
- ⑯ 功能卡插槽
- ⑰ 功能卡固定孔
- ⑱ 电源、错误及电池状态指示
- ⑲ I/O 模块接口
- ⑳ 机身固定螺丝
- ㉑ 直接固定孔

■ DVP-10PM



- ① 通讯口上盖
- ② 输入 / 输出端子盖
- ③ 输入点指示灯
- ④ 输入 / 输出端子
- ⑤ I/O 模块接口上盖
- ⑥ 输入 / 输出端子编号
- ⑦ 功能卡 / 记忆卡上盖
- ⑧ 输出点指示
- ⑨ DIN 轨固定扣
- ⑩ COM2 通讯端口 (RS-485)
- ⑪ STOP / RUN 开关
- ⑫ COM1 通讯端口 (RS-232)
- ⑬ 电池插槽
- ⑭ 电池
- ⑮ 功能卡插槽
- ⑯ 电源、运行、错误及电池状态
- ⑰ I/O 模块接口
- ⑱ 机身固定螺丝
- ⑲ 直接固定孔

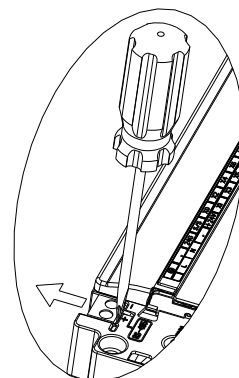
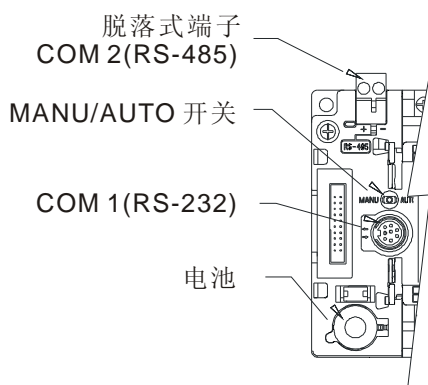
COM1 上盖开启



2 硬件规格及配线方式

更换电池时，请在 1 分钟内完成。

RS-485 端子取出



部 位	说 明
COM2 (RS-485)	Master/Slave 模式两用
MANU/AUTO 开关	DVP-PM RUN/STOP 控制
COM1 (RS-232)	Slave 模式 (与 COM2 可同时使用)

配线端子位置 (详细规格请参考第 2.1.2 章节)

■ DVP-20PM00D

⊕	•	24G	+24V	S/S0	STOP0	LSP0	PG0-	S/S1	STOP1	LSP1	PG1-	X0	X2	X4	X6
L	N	•	START0	DOG0	LSN0	PG0+	START1	DOG1	LSN1	PG1+	S/S2	X1	X3	X5	X7
DVP-20PM00D (AC Power IN, DC Signal IN)															
A0+	B0+	A1+	B1+	CLR0+	CLR1+	FP0+	RP0+	FP1+	RP1+	Y0	Y1	Y2	Y3	Y4	Y6
A0-	B0-	A1-	B1-	CLR0-	CLR1-	FP0-	RP0-	FP1-	RP1-	C0	C1	C2	C3	Y5	Y7

■ DVP-20PM00M


⊕	•	24G	+24V	S/S0	STOP0	LSP0	PG0-	S/S1	STOP1	LSP1	PG1-	X0	X2	X4	X6
L	N	•	START0	DOG0	LSN0	PG0+	START1	DOG1	LSN1	PG1+	S/S2	X1	X3	X5	X7
DVP-20PM00M (AC Power IN, DC Signal IN)															
A0+	B0+	A1+	B1+	CLR0+	CLR1+	FP0+	RP0+	FP1+	RP1+	FP2+	RP2+	Y2	Y3	Y4	Y6
A0-	B0-	A1-	B1-	CLR0-	CLR1-	FP0-	RP0-	FP1-	RP1-	FP2-	RP2-	C2	C3	Y5	Y7

■ DVP-10PM

⊕	•	24G	+24V	X0	X2	X4	X6	X10+	X11+	X12+	X13+
L	N	•	S/S	X1	X3	X5	X7	X10-	X11-	X12-	X13-
DVP-10PM (AC Power IN, DC Signal IN)											
Y0	Y1	Y2	Y3	Y10+	Y11+	Y12+	Y13+	Y14+	Y15+	Y16+	Y17+
C0	C1	C2	C3	Y10-	Y11-	Y12-	Y13-	Y14-	Y15-	Y16-	Y17-

2.2 配线方式

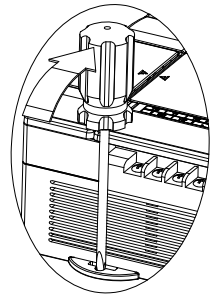
本机为开放型 (OPEN TYPE) 机壳, 因此使用者使用本机时, 必须将之安装于具防尘、防潮及免于电击/冲击意外之外壳配线箱内。另必须具备保护措施 (如: 特殊之工具或钥匙才可打开) 防止非维护人员操作或意外冲击本体, 造成危险及损坏。

交流输入电源不可连接于输入/出信号端, 否则可能造成严重损坏, 请在上电之前再次确认电源配线。本体上之接地端子  务必正确的接地, 可提高产品抗噪声能力。

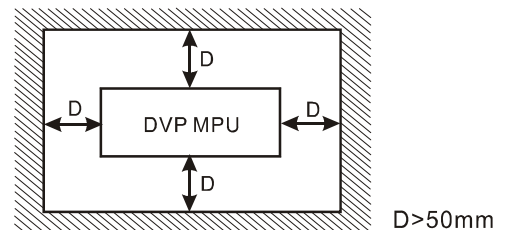
2.2.1 盘内安装配线

DIN 铝轨之安装方法:

适合 35mm 之 DIN 铝轨。在将主机挂上铝轨时, 请先将 PLC 下方之固定塑料片, 以一字形起子插入凹槽并向外撑开拉出, 再将 PLC 挂上铝轨, 之后将固定塑料片压扣回去即可。欲取下主机时, 同样以一字形起子先将固定塑料片撑开, 再将主机以往外向上的方式取出即可。该固定机构塑料片为保持型, 因此撑开后便不会弹回去, 如右图所示:

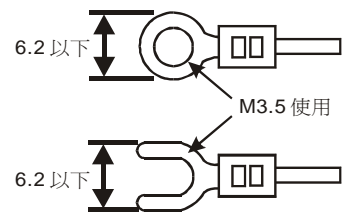


1. 直接锁螺丝方式: 请依产品外型尺寸并使用 M4 螺丝。
2. DVP-PM 在安装时, 请装配于封闭式之控制箱内, 其周围应保持一定之空间, (如右图所示), 以确保 DVP-PM 散热功能正常。



配线注意事项:

1. 出/入配线端请使用 O 型或 Y 型端子, 端子规格如右所示。DVP-PM 端子螺丝扭力为 9.50 kg-cm (8.25 in-lbs)。只能使用 60/75°C 的铜导线。
2. 空端子请勿配线。输入点讯号线与输出点等动力线请勿置于同一线槽内。
3. 锁螺丝及配线时请避免微小的金属导体掉入 PLC 内部, 并在配线完成后, 将位于 PLC 上方散热孔位置的防异物掉入之贴纸撕去, 以保持散热良好。



2.2.2 电源输入配线

DVP-PM 系列主机电源输入为交流输入, 在使用上应注意下列事项:

1. 交流电源输入电压, 范围宽广 (100VAC ~ 240VAC), 电源请接于 L、N 两端, 如果将 AC110V 或 AC220V 接至+24V 端或输入点端, 将使 DVP-PM 损坏, 请使用者特别注意。
2. 主机及 I/O 模块之交流电源输入请同时作 On 或 Off 的动作。
3. 主机之接地端使用 1.6mm 以上之电线接地。
4. 当停电时间低于 10ms 时, DVP-PM 不受影响继续运转, 当停电时间过长或电源电压下降将使 DVP-PM 停止运转, 输出全部 Off, 当电源恢复正常时, DVP-PM 将自动回复运转。(DVP-PM 内部具有停电保

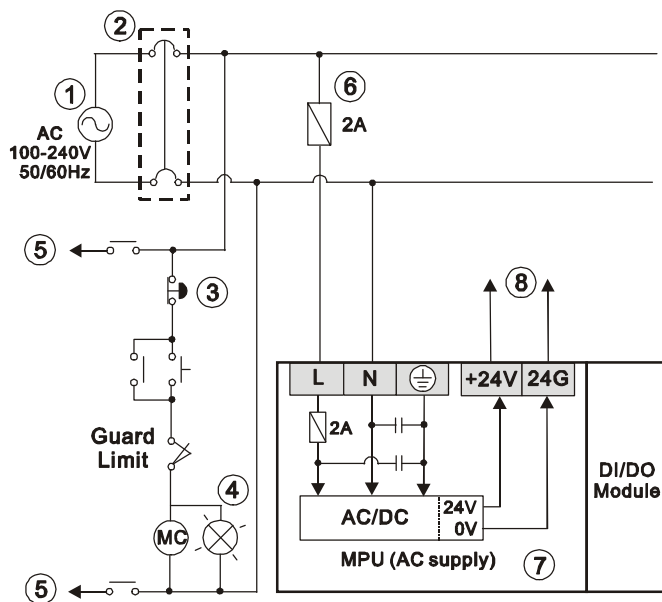
2 硬件规格及配线方式

持的辅助继电器及缓存器，使用者在作程序设计规划时应特别注意使用。)

5. +24V 电源供应输出端，最大为 0.5A，请勿将其它的外部电源连接至此端子。每个输入点驱动电流必须 6 ~ 7mA，若以 16 点输入计算，大约需 100mA，因此+24V 输出给外部负载不可大于 400mA。

2.2.3 安全回路配线

由于 DVP-PM 控制许多装置，任一装置的动作可能都会影响其它装置的动作，因此任一装置的故障都有可能造成整个自动控制系统失控，甚至造成危险。所以在电源端输入回路，建议配置如下的保护回路：



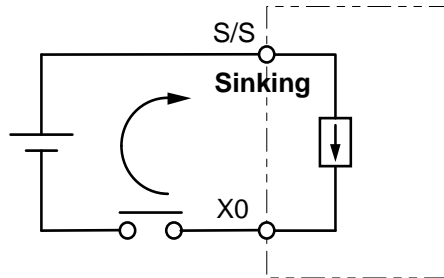
- ① 电源供应：交流 (AC) : 100~240VAC, 50/60Hz。
- ② 断路器。
- ③ 紧急停止：为预防突发状况发生，设置一紧急停止按钮，可在状况发生时，切断系统电源。
- ④ 电源指示灯。
- ⑤ 交流电源负载。
- ⑥ 电源回路保护用保险丝 (3A) 。
- ⑦ DVP-PM 本体。
- ⑧ 直流电源供应输出：24VDC，500mA

2.2.4 输入/输出接点配线

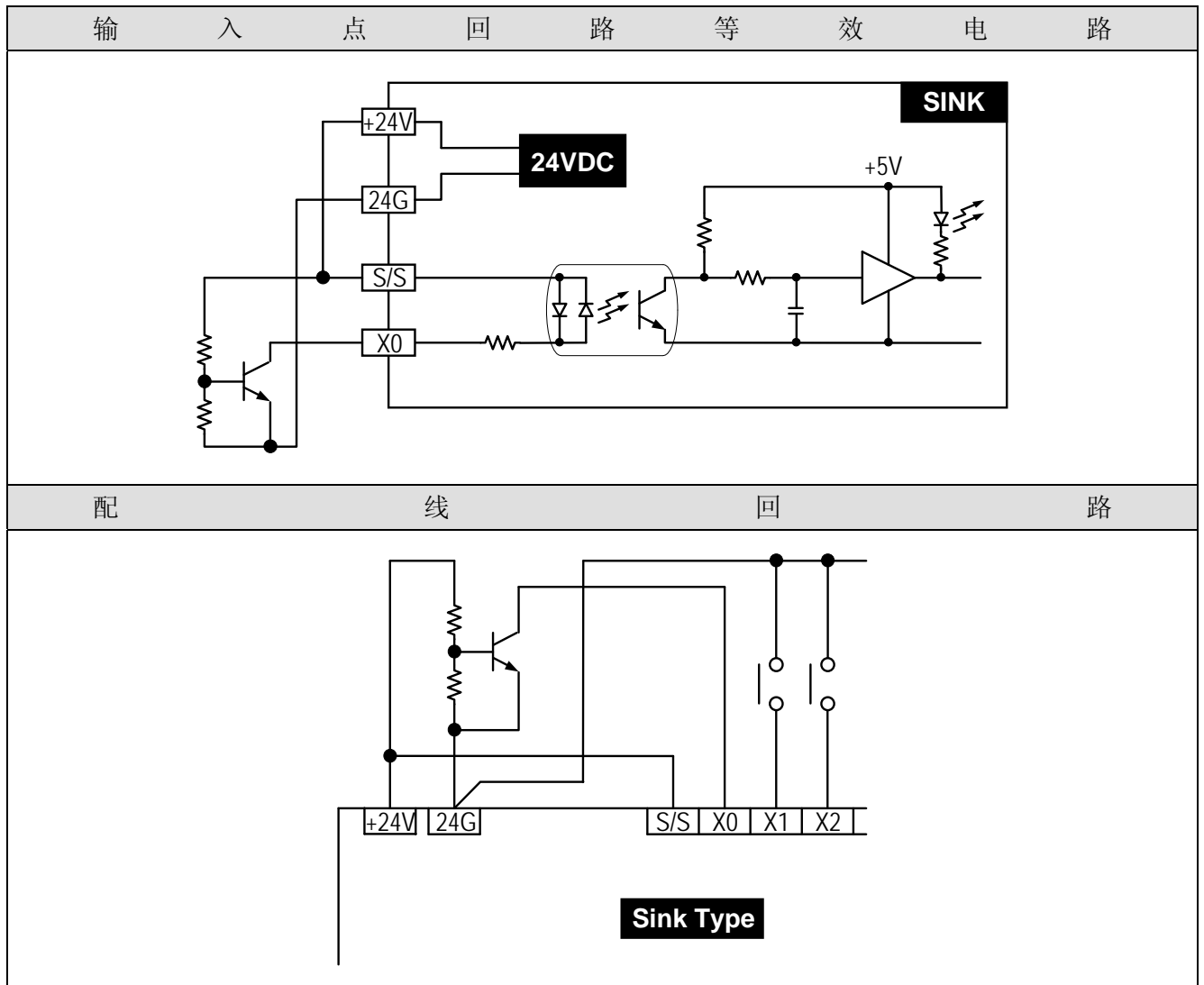
1. 输入点之入力信号为直流电源 DC 输入，DC 型式共有两种接法：

SINK 定义如下：

直流形式 (DC Signal IN)



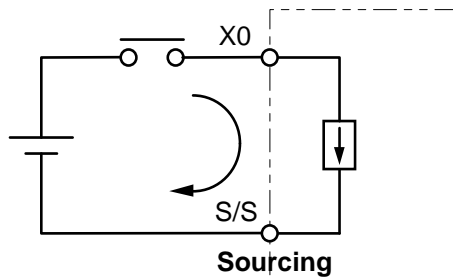
SINK 模式 (电流流入共享端 S/S)



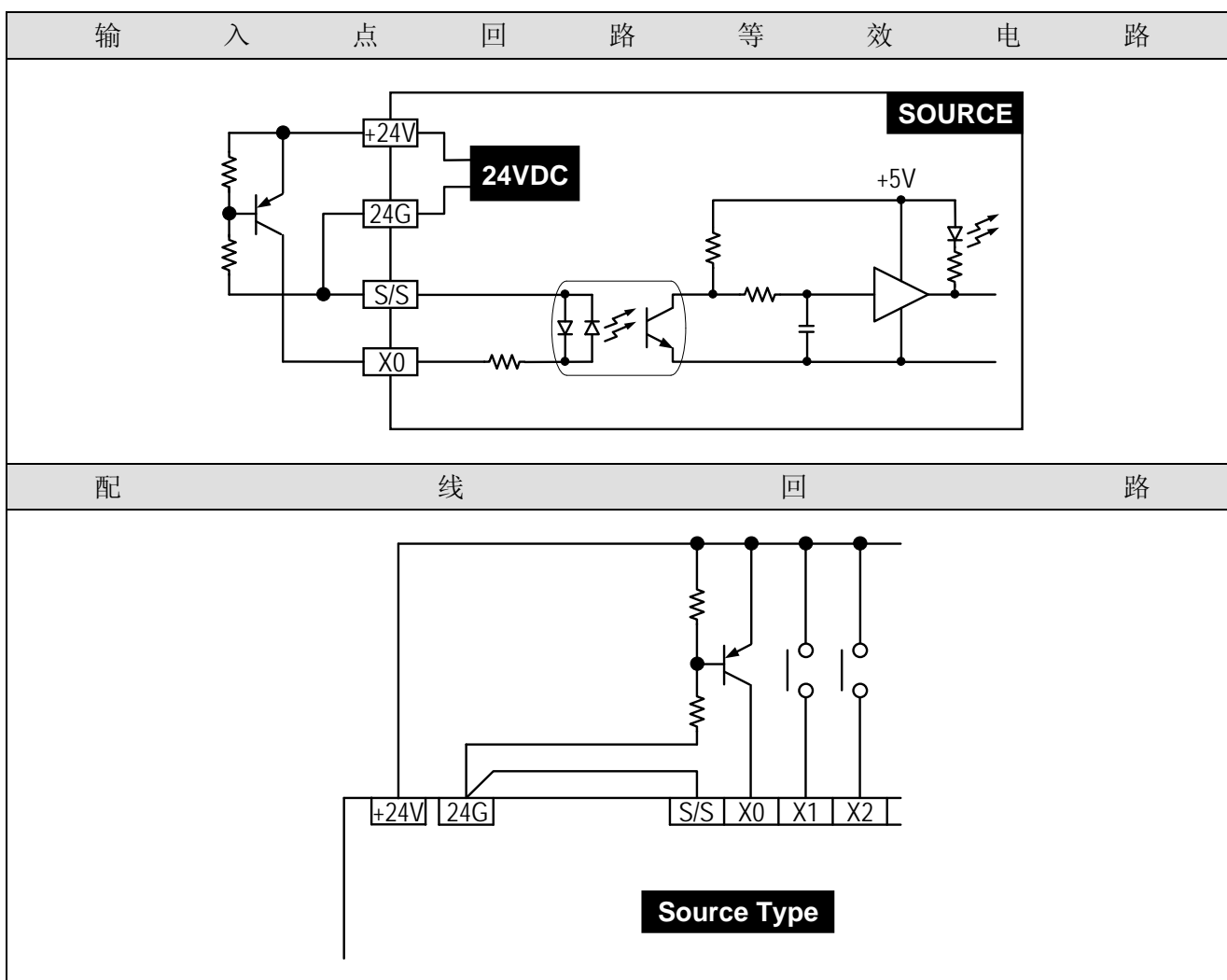
2 硬件规格及配线方式

SOURCE 定义如下:

直流形式 (DC Signal IN)



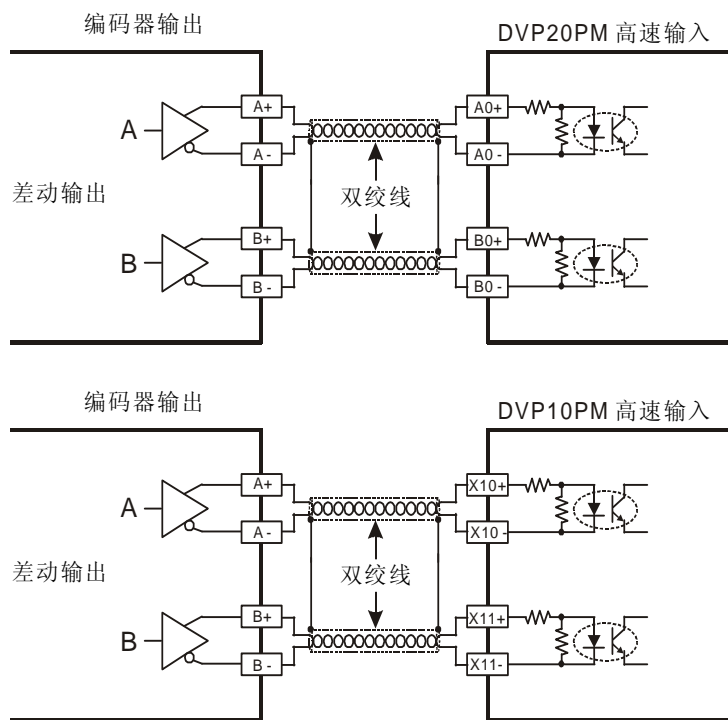
SOURCE 模式 (电流流出共享端 S/S)



2. 差动输入之配线

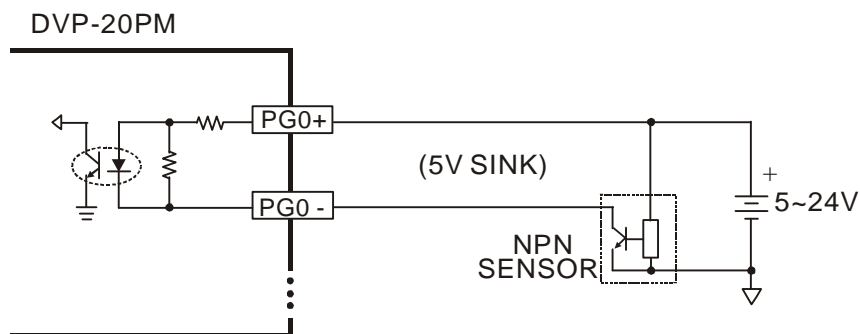
DVP-PM 之 A0 ~ A1 及 B0 ~ B1 均为 DC5 ~ 24V 高速输入电路 (其余则为 DC24V 输入)。此电路最高工作频率可达 200kHz, 主要使用在连接差动 (双线式) Line Driver 输出电路。

差动输入之接线图 (高速、高噪声时使用) :

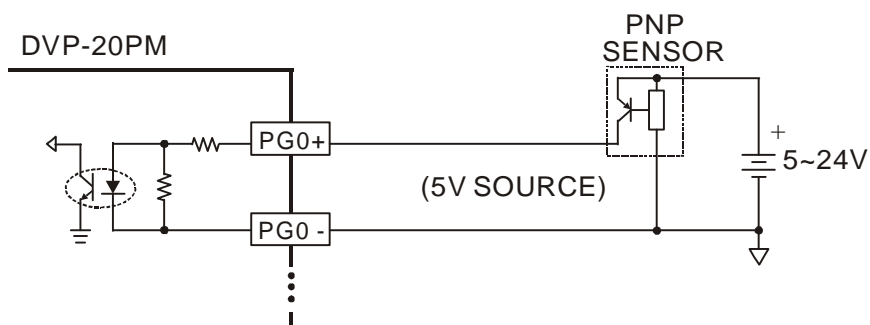


若工作环境频率不高 (小于 50kHz) 且噪声较低, 亦可使用 DC5 ~ 24V SINK / SOURCE 之单端输入方式。DVP-PM DC5V ~ 24V SINK 与 SOURCE 输入之接线图, 如下所示:

DVP-PM DC5V SINK 输入之接线图:

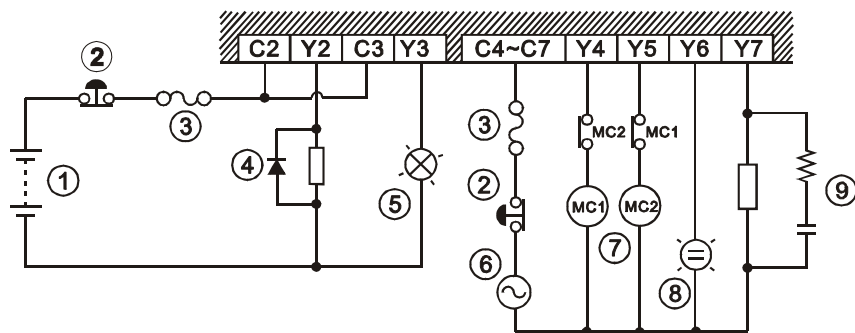


DVP-PM DC5V SOURCE 输入之接线图:



2 硬件规格及配线方式

3. 继电器 (R) 输出回路配线



① 直流电源供给。

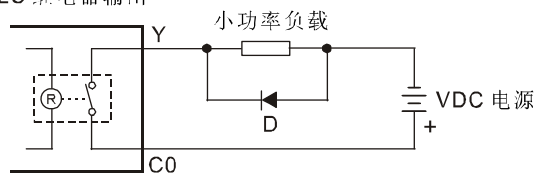
② 紧急停止：使用外部开关。

③ 保险丝：使用 5~10A 的保险丝容量于输出接点的共享点，保护输出点回路。

④ 突波吸收二极管：可增加接点寿命。

1. DC 负载电源之二极管抑制：功率较小时使用。

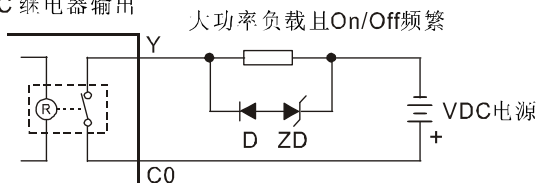
PLC 继电器输出



D: 1N4001 二极管或等效组件

2. DC 负载电源之二极管+Zener 抑制：大功率且 On/Off 频繁时使用。

PLC 继电器输出



D: 1N4001 二极管或等效组件
ZD: 9V Zener 二极管, 5W

⑤ 白炽灯 (电阻性负载)。

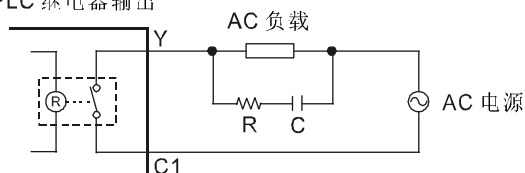
⑥ 交流电源供给。

⑦ 互斥输出：例如，将 Y4 与 Y5 用以控制对应马达的正转及反转，使外部电路形成互锁，配合 PLC 内部程序，确保任何异常突发状况发生时，均有安全的保护措施。

⑧ 指示灯：氖灯。

⑨ 突波吸收器：可减少交流负载上的噪声。

PLC 继电器输出

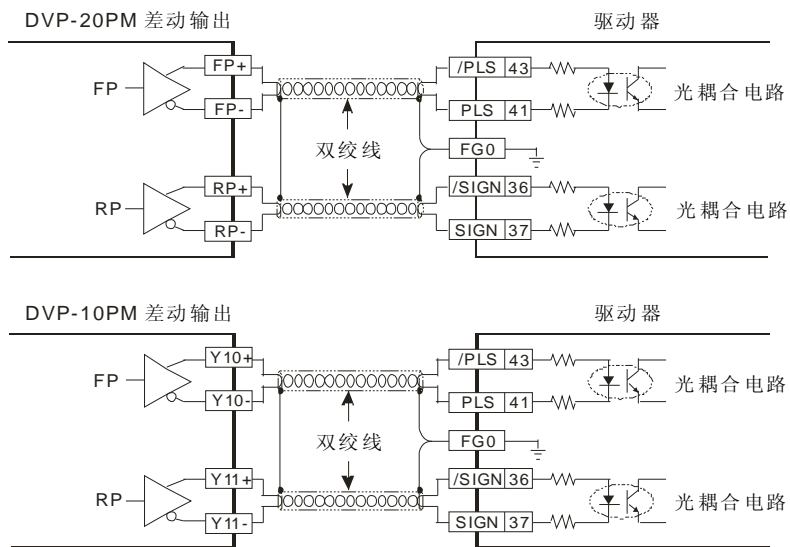


R: 100~120 Ω
C: 0.1~0.24 μF

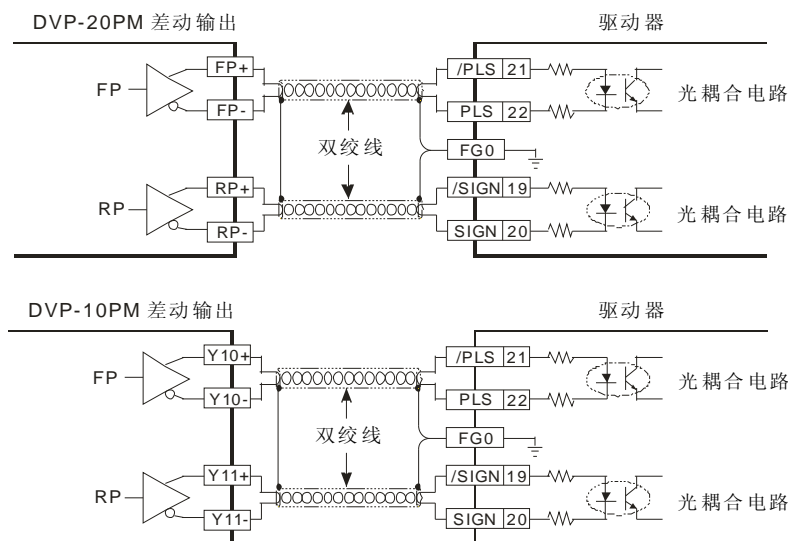
2 硬件规格及配线方式

5. 差动输出之接线图

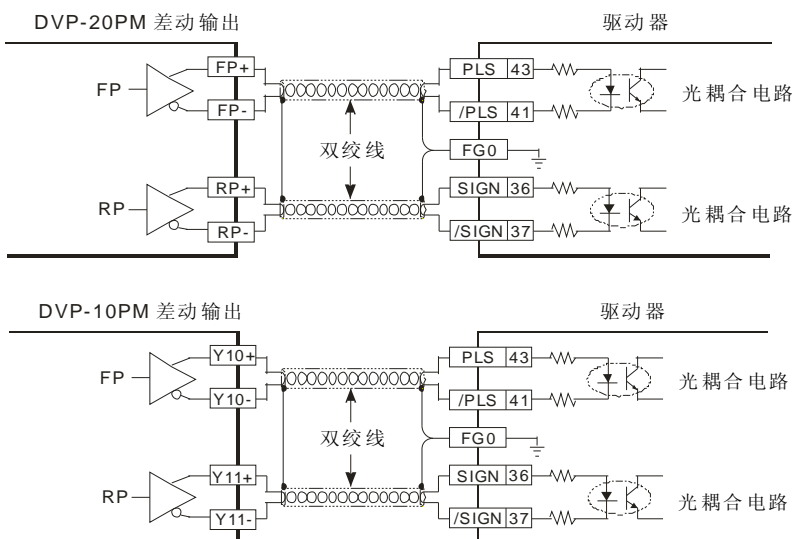
• DVP-PM 差动输出与 ASDA-A 、 ASDA-A+ 及 ASDA-A2 系列驱动器



• DVP-PM 差动输出与 ASDA-B 系列驱动器

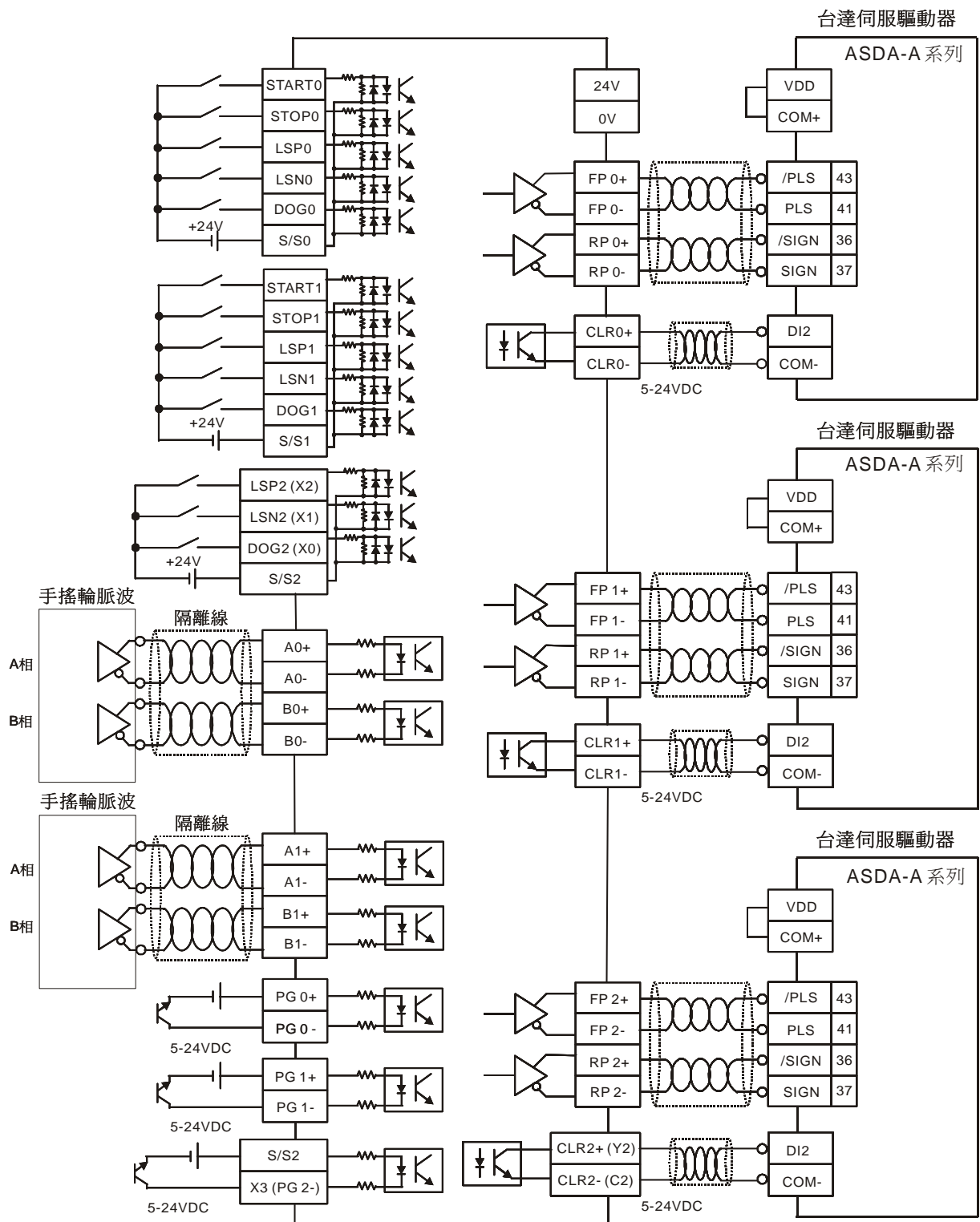


• DVP-PM 差动输出与 ASDA-AB 系列驱动器



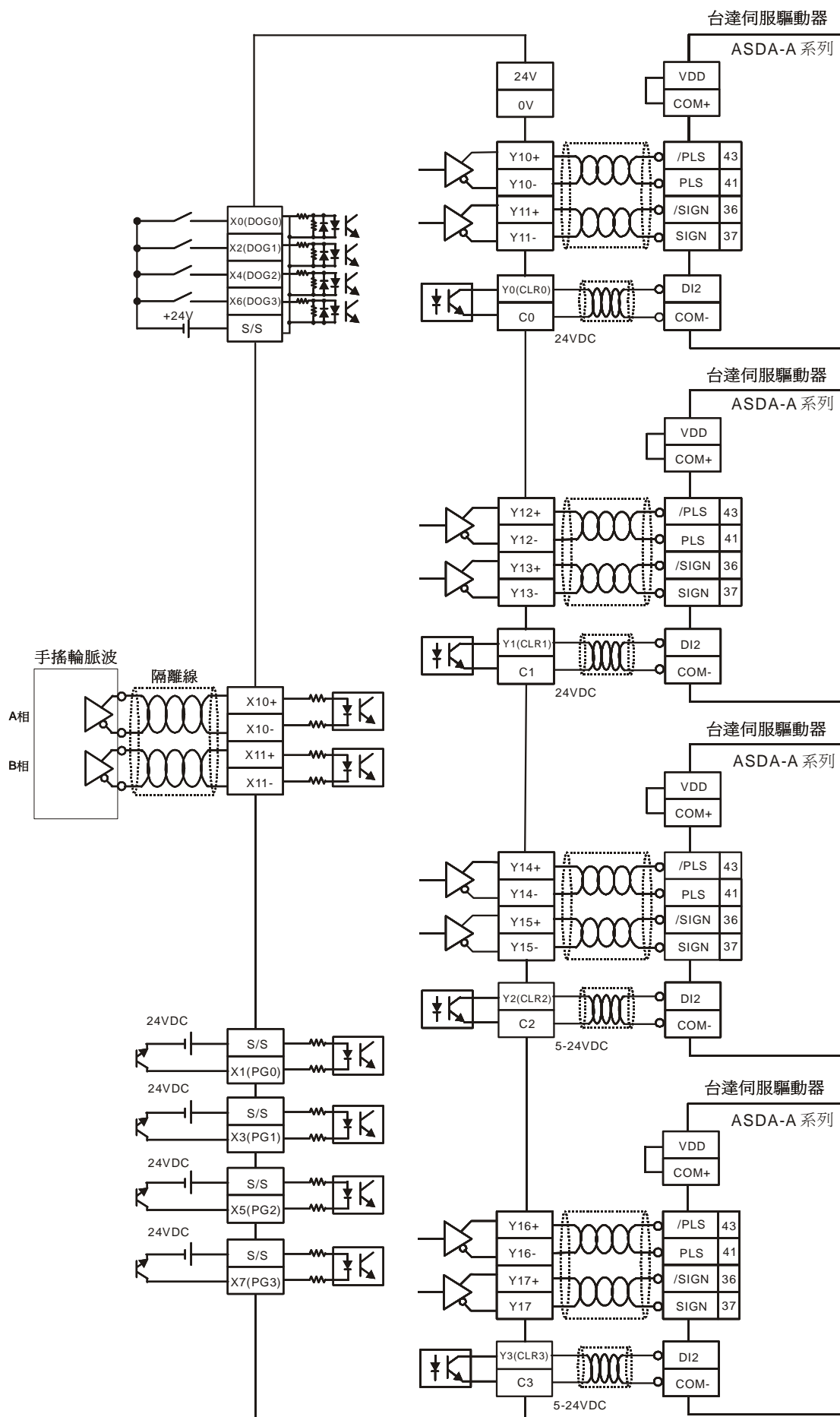
2.2.5 与下位驱动器之配线

DVP-20PM 与台达 ASDA-A 系列伺服驱动器之配线图：



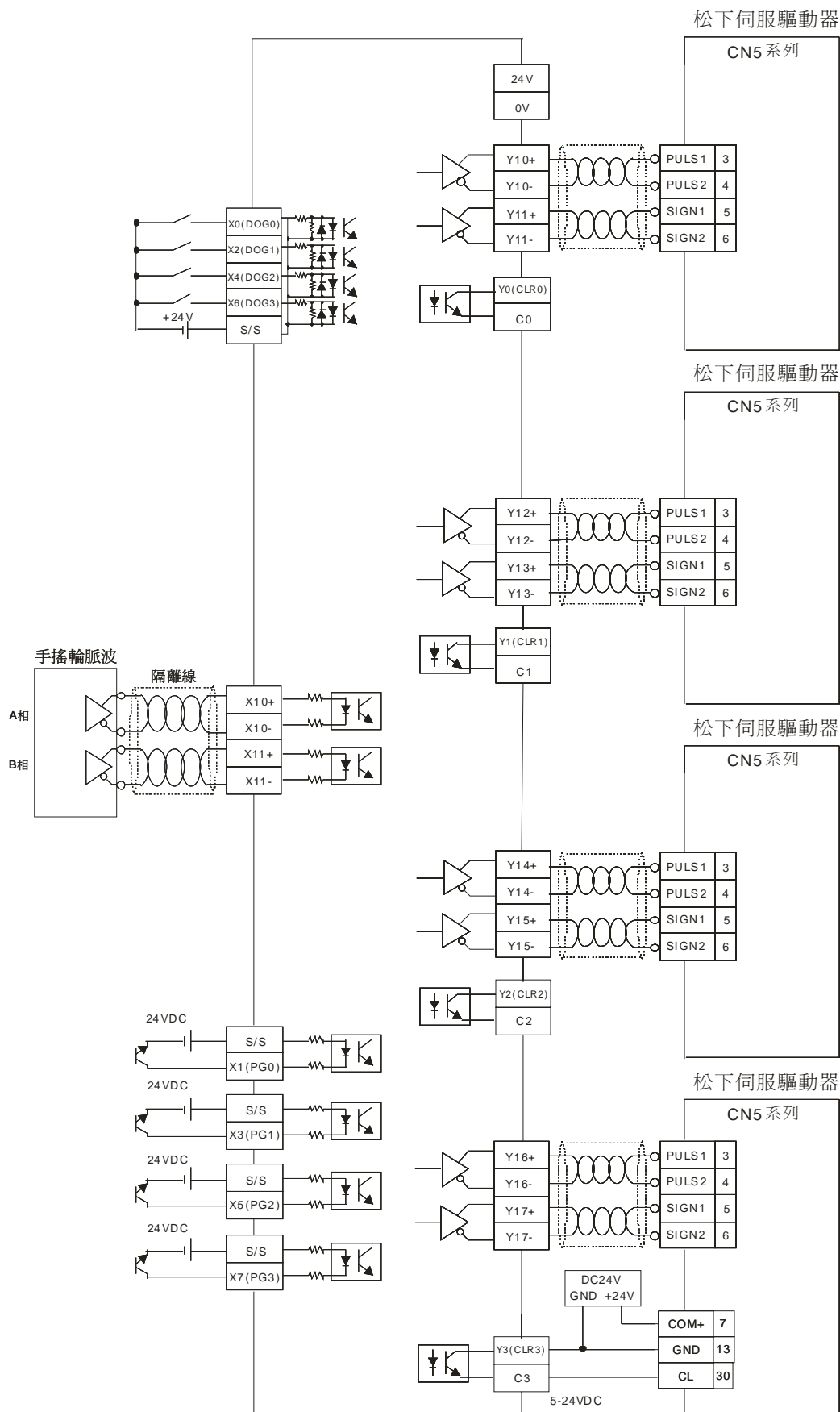
2 硬件规格及配线方式

DVP-10PM 与台达 ASDA-A 系列伺服驱动器之配线图：

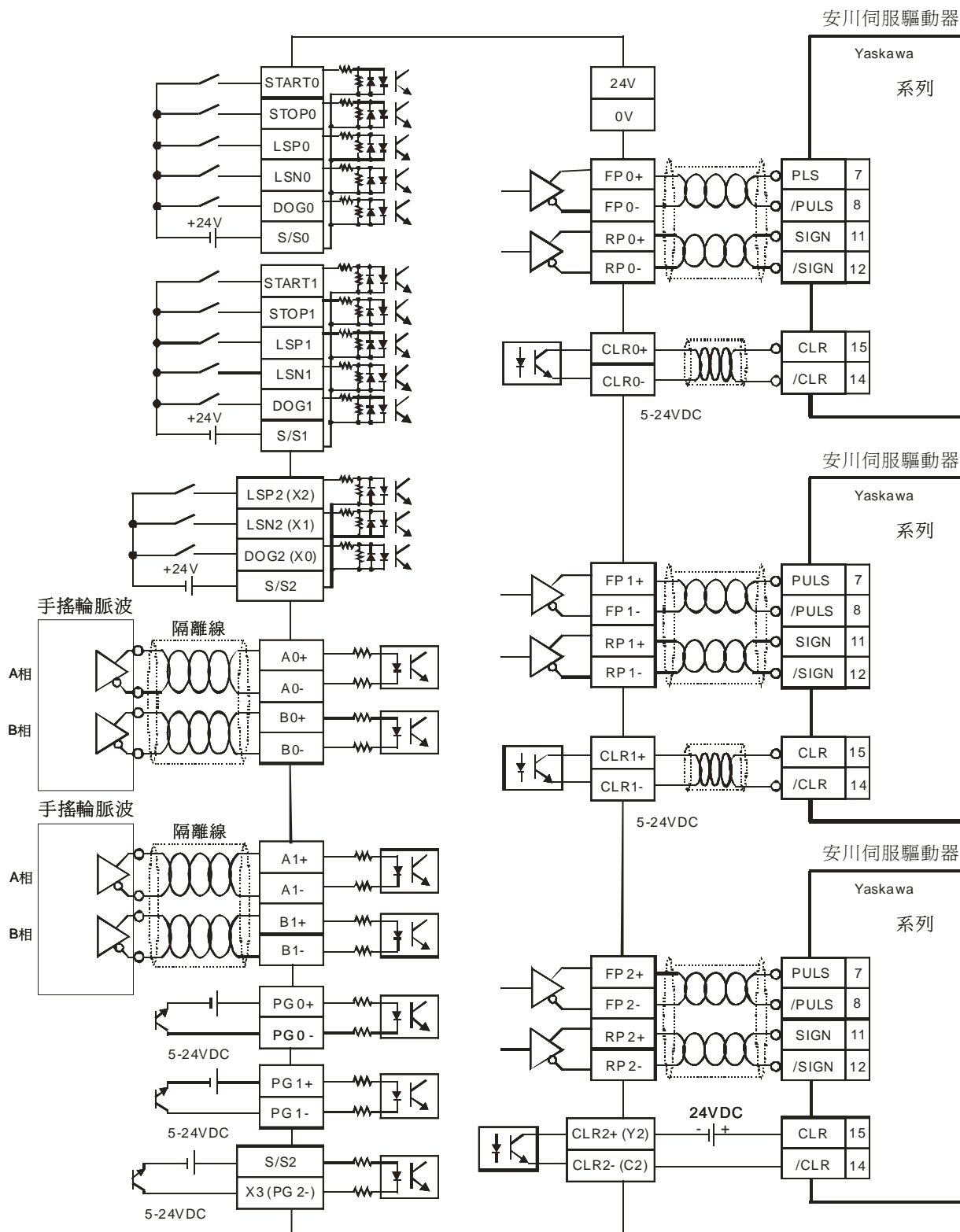


2 硬件规格及配线方式

DVP-10PM 与松下 CN5 系列伺服驱动器之配线图:

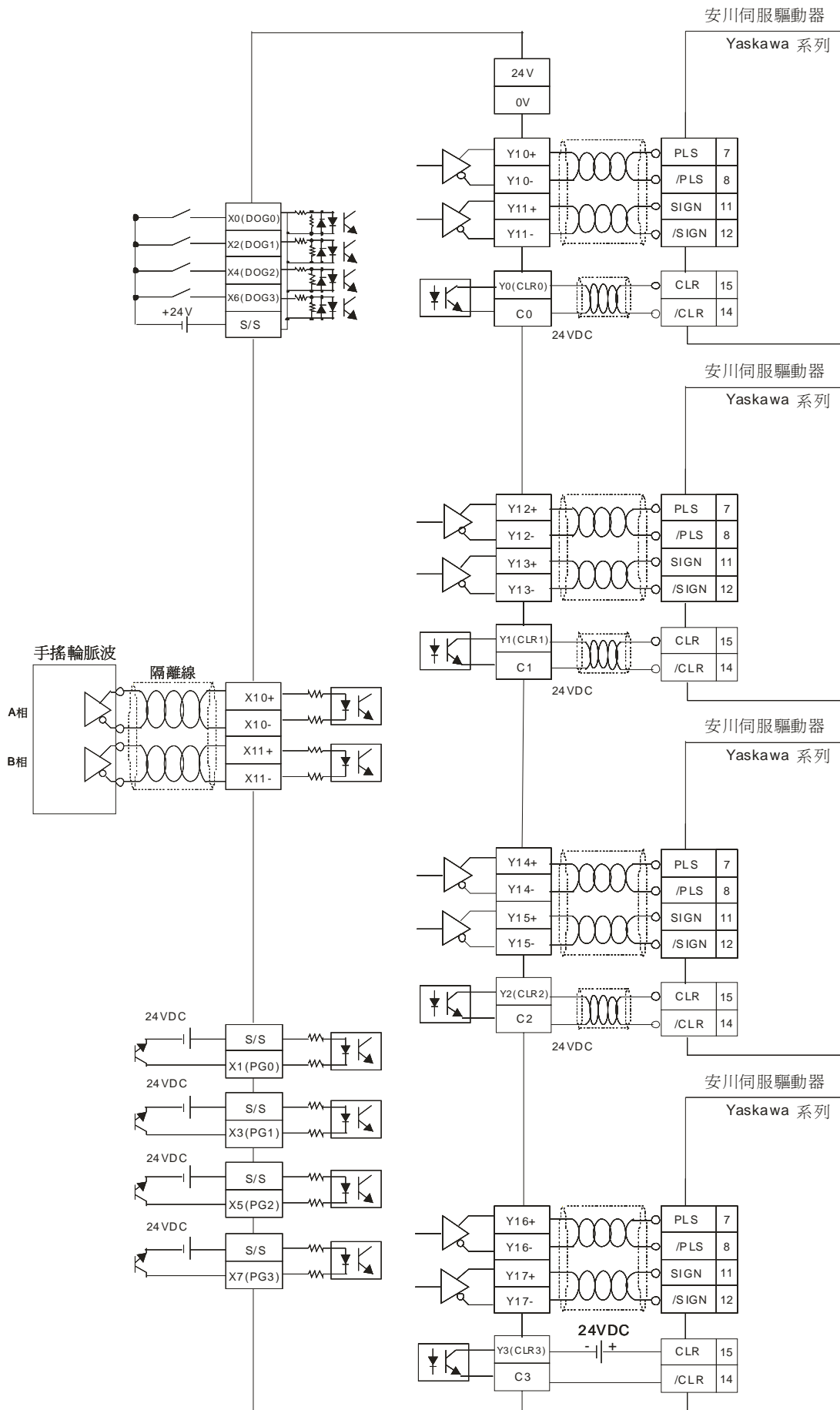


DVP-20PM 与安川伺服驱动器之配线图:

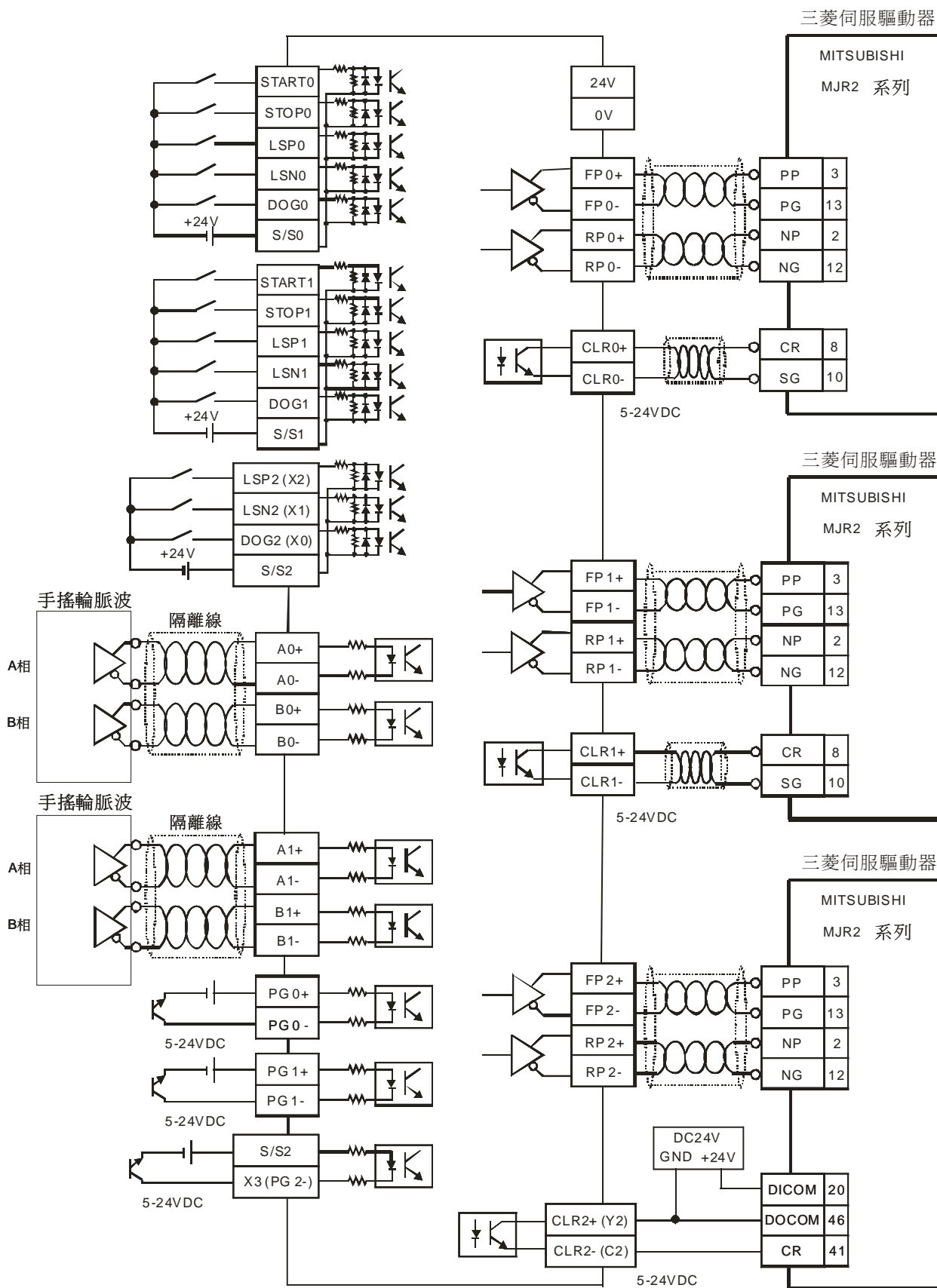


2 硬件规格及配线方式

DVP-10PM 与安川伺服驱动器之配线图：

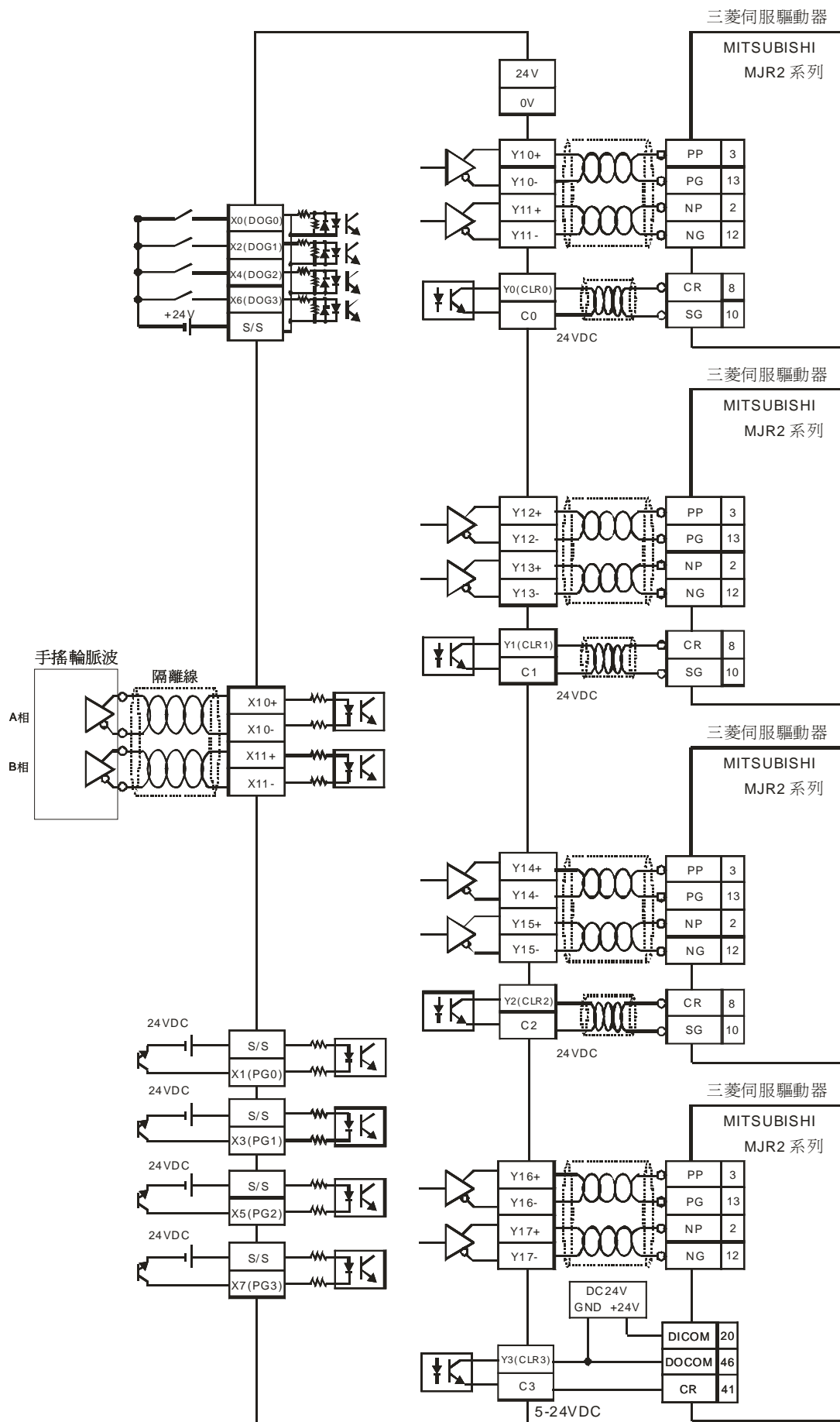


DVP-20PM 与三菱 MJR2 系列伺服驱动器之配线图:

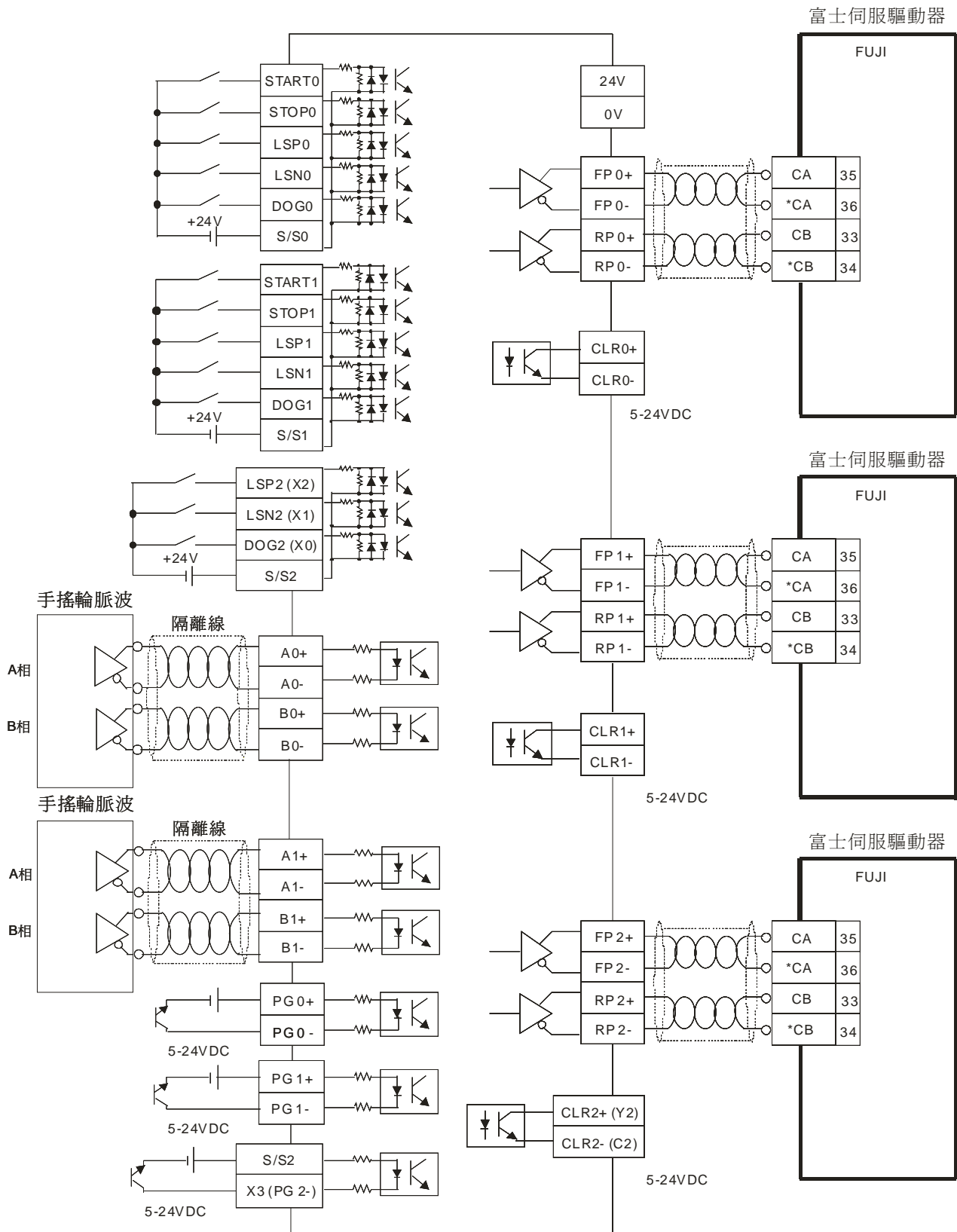


2 硬件规格及配线方式

DVP-10PM 与三菱 MJR2 系列伺服驱动器之配线图:

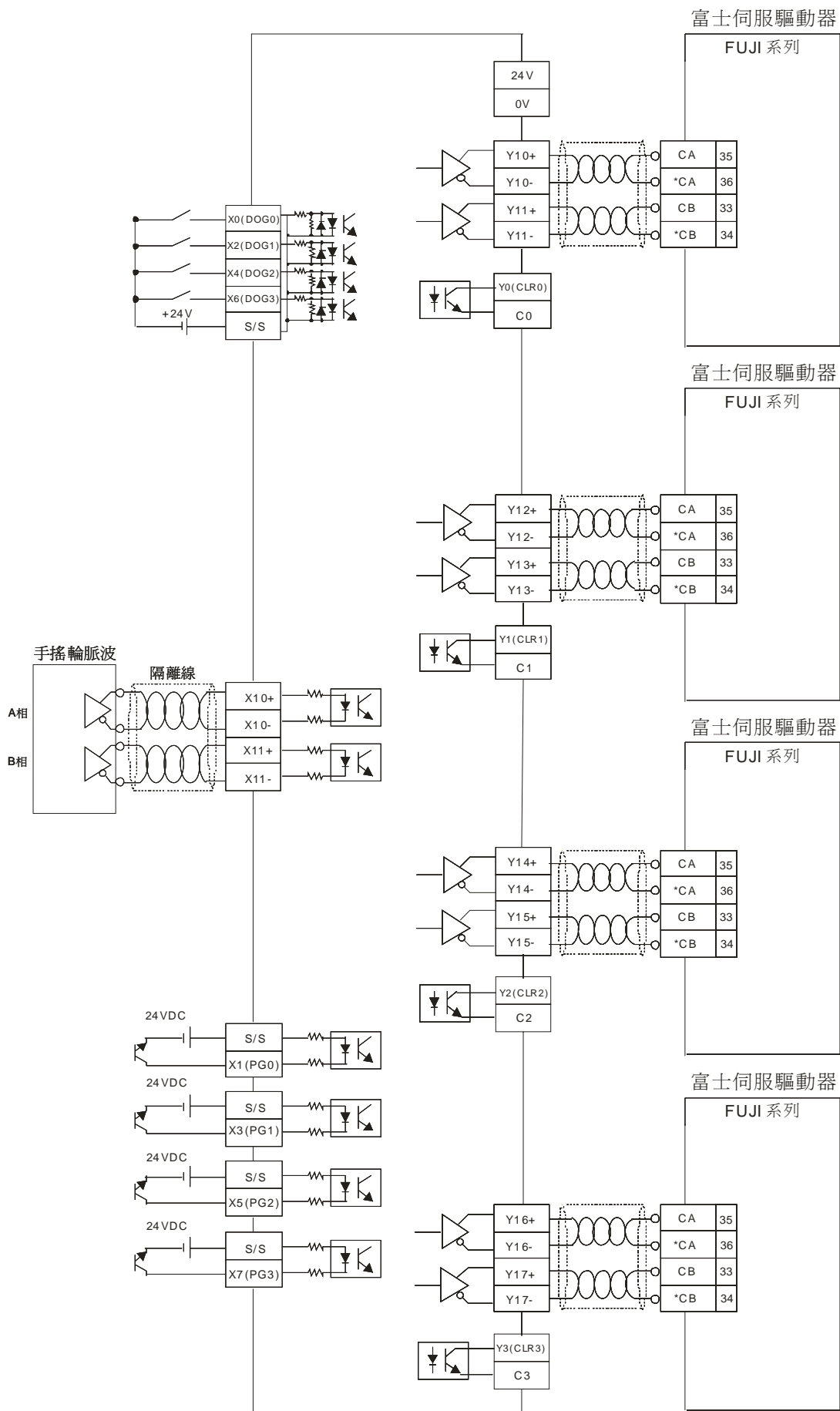


DVP-20PM 与富士伺服驱动器之配线图:



2 硬件规格及配线方式

DVP-10PM 与富士伺服驱动器之配线图：



2.3 通讯口说明

DVP-PM 系列主机的通讯口有内建 COM1(RS-232), COM2(RS-485)及通讯功能卡 COM3(RS-232 或 RS-485), 请参考以下之说明。

COM1: RS-232 通讯端口, 可做从站。为主要的程序编辑通讯端口, 可用于 Modbus ASCII 或 RTU 模式。

COM2: RS-485 通讯端口, 可做主站或从站, 可用于 Modbus ASCII 或 RTU 模式。

COM3: RS-232/RS-485 通讯端口, 可做从站, 可用于 Modbus ASCII 模式。

通讯架构:

通讯端口	RS-232 (COM1)	RS-485 (COM2)	RS-232/RS-485 (COM3)
波特率	110~115,200 bps		110~38,400 bps
数据位长度	7~8 位		
奇偶校验位	偶校验 EVEN / 奇校验 ODD / 无校验 NONE		
停止数据位长度	1~2 数据位		
通讯格式设定缓存器	D1036	D1120	D1109
ASCII 模式	从站有效	主站/从站都有效	从站有效
RTU 模式	从站有效	主站/从站都有效	-
读写数据长度 (ASCII 模式)	100 个缓存器		32 个缓存器
读写数据长度 (RTU 模式)	100 个缓存器		32 个缓存器

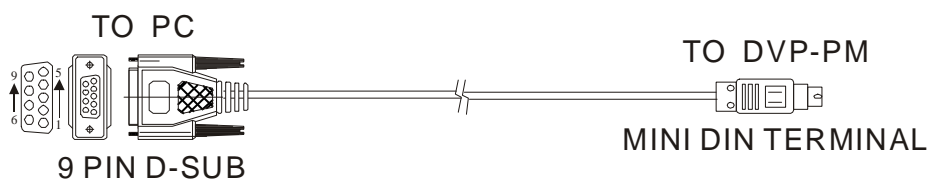
通讯端口的预设通讯格式

- Modbus ASCII 模式
- 7 个数据位
- 1 个停止位
- 偶校验位 (EVEN)
- 9600bps 波特率

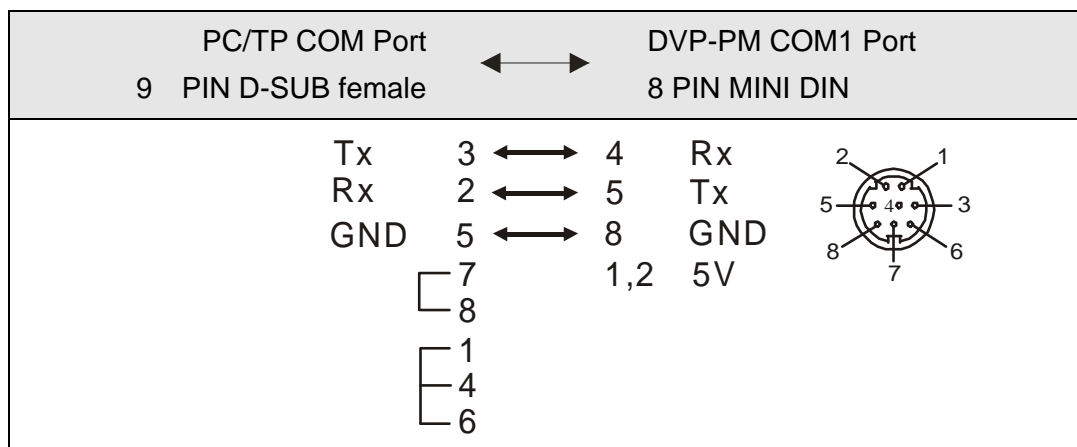
2.3.1 COM1 (RS-232)

1. 通讯口 COM1 为 RS-232 接口之通讯口, 主要为提供 DVP-PM 程序上/下传, 其支持 Modbus 通讯格式 (ASCII/RTU 模式), 且通讯速度为 9,600 ~ 115,200 bps。

通讯传输线路(DVPACAB2A30)说明如下:



2 硬件规格及配线方式



- COM1 之工作模式为 **Slave** 型态，因此可以与人机接口连接，作为监控之用途。

2.3.2 COM2 (RS-485)

- 通讯口 **COM2** 为 **RS-485** 接口之通讯口，主要为提供主/从站的多台通讯联机，其支持 **Modbus** 通讯格式(ASCII/RTU 模式)，且通讯速度为 9,600 ~ 115,200 bps。
- COM2** 之工作模式为 **Master/Slave** 型态：当 **Master** 型态时，可连接台达 **PLC** 或下位驱动器 (如台达伺服、台达变频器、温控器...等)，进行数据的读出/写入及控制；当 **Slave** 型态时，则可以与人机接口连接 (如台达 **TP** 及 **DOP** 系列人机接口) 作为监控之用途。

2.3.3 COM3 (RS-232/RS-485)

- 当使用者对于主机内建 **COM1(RS-232)**，**COM2(RS-485)** 的两个 **COM** 不敷使用时，可采用此卡增加一个 **COM**(编号为 3, 称为 **COM3** 卡, 可选择 **RS-232** 或 **RS-485** 接口两种, 功能卡 **DVP-F232S**，**DVP-F485S** 一般的功能与 **COM1** 相同，但通讯速率仅支持 9600/19200/38400 bps，**ASCII** 模式。
- COM3** 之工作模式为 **Slave** 型态，因此可以与人机接口连接，作为监控之用途。

3.1 DVP-PM 各装置编号一览表

■ 功能规格

项 目	规格			备注
	20PM		10PM	
轴运作方式	双轴/三轴同动线性圆弧补间及独立双轴控制 (*5)		四轴同动及独立四轴控制	
程序储存	内藏 64k Steps 储存器			
单位系	马达单位	复合单位	机械单位	
主机读写 I/O 模块方式	使用 FROM/TO 指令来读取/写入 I/O 模块 CR 的数据内容 如果数据内容为 32Bit 时, 以 2 个 CR 处理			
与主机串接说明	当作 I/O 模块使用时, 内建 CR0~CR199 (对应本身之 D1500~D1699)供主机读写			
脉冲输出方式	三种模式: Pulse/Dir, FP(CW)/RP(CCW), A/B; 均采用差动输出			
最快速度值	单轴最快 500k PPS 补间轴最快 500k PPS			
输入信号	操作开关	AUTO/MANU (自动/手动选择开关)、START (启动)、STOP (停止)	STOP/RUN (自动/手动选择开关)	
	检知器	DOG (近点信号)、LSP (正转极限)、LSN (反转极限)、PG0 (零点)	X0、X2、X4、X6 (近点信号) X1、X3、X5、X7 (零点信号)	
	一般输入点	X0~X7, 可扩充 I/O 模块, 最大扩充点数含主机 256 点		
输出信号	伺服输出信号	FP (正转脉冲)、RP (反转脉冲)、CLR (清除讯号)	Y10+、Y10-、Y12+、Y12-、Y14+、Y14-、Y16+、Y16- (正转脉冲) Y11+、Y11-、Y13+、Y13-、Y15+、Y15-、Y17+、Y17- (反转脉冲)	
	一般输出点	Y0~Y7, 可扩充 I/O 模块, 最大扩充点数含主机 256 点		
	串联通讯口	程序写入/读出通讯口: COM1: RS-232 (可作从站) COM2: RS-485 (可作主站或从站) COM3 通讯卡: RS-232/ RS-485 (可作从站) (选购)		
特殊 I/O 模块	选购	右侧模块与 EH2 系列共享所有模块 AD、DA、PT、TC、XA、PU (最大可扩充 8 台, 不占 I/O 点数) 左侧可连接左侧高速 I/O 模块 (最大可扩充 8 台, 不占 I/O 点数)		
基本指令	27 个		27 个	
应用指令	130 个		130 个	
运动指令	22 个		个	
M-Code	<ul style="list-style-type: none"> ■ OX0 ~ 99 (运动子程序/Positioning Program) : M02 程序停止 (END) ■ M00 ~ M01, M03 ~ M101, M103 ~ M65535 做为程序暂停 (WAIT), 可自由使用 ■ O100 (DVP-PM 主程序/Sub-task Program) : M102 程序停止 (END) 			

3 DVP-PM 各种装置功能

项 目		规格		备注		
		20PM	10PM			
G-Code		G0(快速移动), G1(直线差补), G2(顺时针圆弧差补), G3(逆时针圆弧差补), G4(停顿时间), G17(XY 平面设定), G18(XZ 平面设定), G19(YZ 平面设定), G90(绝对坐标), G91(相对坐标)		无		
自我诊断		参数错误、程序错误、外部错误等各种错误显示				
继电器 (位型态)	X	外部输入继电器	X0 ~ X377, 8 进制编码, 256 点(对应至外部输入点)		合计 512 点	
	Y	外部输出继电器	Y0 ~ Y377, 8 进制编码, 256 点(对应至外部输出点)			
	M	辅助继电器	一般用	M0 ~ M499, 500 点 (*2)		合计 4,096 点, 接点可于程序内做 On/Off。
			停电保持用	M3000 ~ M4095, 1096 点 (*3)		
			特殊用	M500 ~ M999, 500 点 (*3)		
	10ms	M1000 ~ M2999, 2000 点 (部份为停电保持)				
	T	定时器	10ms	T0 ~ T255, 256 点 (*2)		合计 256 点, TMR 指令所指定的定时器, 若计时到达则此同编号 T 的接点将会 On。
	C	计数器	16 位上数	C0 ~ C99, 100 点 (*2)	C0 ~ C99, 100 点 (*2)	合计 250 点, CNT (DCNT) 指令所指定的计数器, 若计时到达则此同编号 C 的接点将会 On。
				C100 ~ C199, 100 点 (*3)	C100 ~ C199, 100 点 (*3)	
			32 位上/下数	C208 ~ C219, 12 点 (*2)	C210 ~ C219, 12 点 (*2)	
C220 ~ C255, 36 点 (*3)				C220 ~ C255, 36 点 (*3)		
32 位高速计数器	C200, C204, 2 点 (*2)		C200, C204, C208, C212, C216, C220, 6 点			
	S	内部继电器	一般用	S0 ~ S499, 500 点 (*2)		合计 1,024 点, 接点可于程序内做 On/Off。
停电保持用			S500 ~ S1023, 524 点 (*3)			
寄存器 (字符组数据)	T	定时器现在值	T0 ~ T255, 16 位定时器, 256 点		计时到达, 定时器接点导通	
	C	计数器现在值	C0 ~ C199, 16 位计数器, 200 点		计数到达, 计数器接点导通	
			C200 ~ C255, 32 位计数器, 56 点			
	D	资料寄存器	一般用	D0 ~ D199, 200 点 (*2)		合计 10,000 点作为数据储存的内存区域, V/Z 可做为间接指定的特殊用途。
			停电保持用	D200 ~ D999, 800 点 (*3)		
D3000 ~ D9999, 7000 点 (*3)						
特殊用			D1000 ~ D2999, 2000 点 (部份为停电保持)			
间接指定用	V0 ~ V7 (16 位), Z0 ~ Z7, 16 点 (32 位) (*1)					

项 目			规格		备注
			20PM	10PM	
指标	P	CJ、CJN、CALL、JMP 指令用	P0 ~ P255, 256 点		CJ、CJN、CALL、JMP 的位置指针
常数	K	10 进制	K-32,768 ~ K32,767 (16 位运算)		
			K-2,147,483,648 ~ K2,147,483,647 (32 位运算)		
	H	16 进制	H0000 ~ HFFFF (16 位运算); H00000000 ~ HFFFFFFFF (32 位运算)		
	F	浮点数	以 32-bit 的长度表示浮点数, 而表示法采用 IEEE754 标准 $\pm 1.1755 \times 10^{-38} \sim \pm 3.4028 \times 10^{+38}$		

- *1: 非停电保持区域, 不可变更。
- *2: 非停电保持区域, 可使用参数设定变更成停电保持区域。
- *3: 停电保持区域, 可使用参数设定变更成非停电保持区域。
- *4: 停电保持固定区域, 不可变更。
- *5: 20PM00M 机种可三轴直线补间及螺旋补间。

■ 具有停电保持及非停电保持的内存装置设定对照一览表:

M 辅助继电器	一般用			特殊辅助继电器
	M0 ~ M499	M500 ~ M999	M3000 ~ M4095	M1000 ~ 2999
	预设为非停电保持	预设为停电保持	预设为非停电保持	(穿插于“一般用”范围之内)
	起始: D1200(K500)*1 结束: D1201(K999) *1			部分为停电保持 不可变更

T 定时器	10 ms
	T0 ~ T255
	预设为非停电保持
	起始: D1202 (K-1) *2, 结束: D1203 (K-1) *2

C 计数器	16 位上数		32 位上/下数	
	C0 ~ C99	C100 ~ C199	C200, C204 C208 ~ C219	C220 ~ C255
	预设非停电保持	预设停电保持	预设非停电保持	预设停电保持
	起始: D1204 (K100) 结束: D1205 (K199)		起始: D1206 (K220) 结束: D1207 (K255)	

S 步进继电器	初始用	一般用	停电保持用
	S0 ~ S9	S10 ~ S499	S500 ~ S1023
	预设非停电保持		预设停电保持
	起始: D1208 (K500), 结束: D1209 (K1023)		

3 DVP-PM 各种装置功能

D 寄存器	一般用	停电保持用	特殊寄存器
	D0 ~ D999	D3000 ~ D9999	D1000 ~ D2999 (穿插于“一般用”及“停电保持用” 范围之内)
	预设非停电保持	预设停电保持	部分停电保持 不可变更
	起始: D1210 (K200) *3 结束: D1211 (K9999) *3		

*1: 若使用者设定 D1200=0、D1201=4095, 则 DVP-PM 会自动略过 M1000 ~ M2999, 将 M0 ~ M999、M3000 ~ M4095 设定为停电保持区。

*2: K-1 表示预设为非停电保持

*3: 若使用者设定 D1210=0、D1211=9999, 则 DVP-PM 会自动略过 D1000 ~ D2999, 将 D0 ~ D999、D3000 ~ D9999 设定为停电保持区。

■ 在电源 On/Off 或主机 MANU/AUTO 模式切换时, 一般装置组件状态动作

内存类型	电源 Off=>On	MANU=>AUTO (STOP=>RUN)	AUTO=>MANU (RUN=>STOP)	M1031 非停电保 持区域全部清除	M1032 停电保持 区域全部清除	出厂 设定值
非停电保持	清除	无变化	M1033 Off 清除	清除	无变化	0
			M1033 On 无变化			
停电保持	无变化		无变化	清除	0	

3.2 数值、常数 [K] / [H]、浮点数 [F]

常 数	K	10 进制	K-32,768 ~ K32,767 (16 位运算) K-2,147,483,648 ~ K2,147,483,647 (32 位运算)
	H	16 进制	H0 ~ HFFFF (16 位运算) H0 ~ HFFFFFFFF (32 位运算)
浮 点 数	F	32-bit 的长度	$\pm 1.1755 \times 10^{-38} \sim \pm 3.4028 \times 10^{+38}$ (以 32-bit 的长度表示浮点数, 而表示法采用 IEEE754 标准)

DVP-PM 内部依据各种不同控制目的, 共使用 5 种数值类型执行运算的工作, 各种数值的任务及功能如下说明。

1. 二进制 (Binary Number, BIN)

DVP-PM 内部之数值运算或储存均采用二进制, 二进制数值及相关术语如下:

位 (Bit) : 位为二进制数值之最基本单位, 其状态非 1 即 0

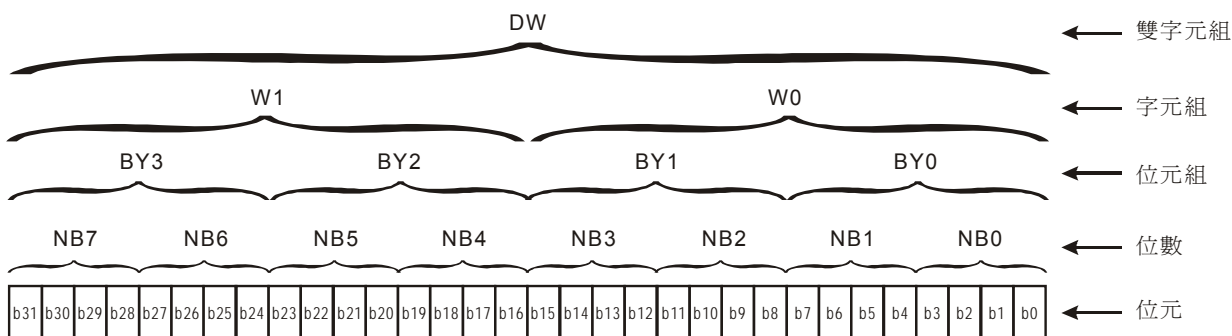
位数 (Nibble) : 由连续的 4 个位所组成 (如 b3~b0) 可用以表示一个位数之 10 进制数字 0~9 或 16 进制之 0~F

字节 (Byte) : 是由连续之两个位数所组成 (亦即 8 位, b7~b0)。可表示 16 进制之 00~FF

字符组 (Word) : 是由连续之两个字节所组成 (亦即 16 位, b15~b0) 可表示 16 进制之 4 个位数值 0000~FFFF

双字符组 (Double Word) : 是由连续之两个字符组所组成 (亦即 32 位, b31~b0), 可表示 16 进制之 8 个位数值 00000000~FFFFFFFF

二进制系统中位、位数、字节、字符组、及双字符组的关系如下图所示:



2. 八进位 (Octal Number, OCT)

DVP-PM 的外部输入及输出端子编号采八进位编码, 例:

- 外部输入: X0~X7, X10~X17... (装置编号)
- 外部输出: Y0~Y7, Y10~Y17... (装置编号)

3. 十进制 (Decimal Number, DEC), 十进制在 DVP-PM 系统应用的时机如下:

- 作为定时器 T、计数器 C 等的设定值, 例: TMR T0 K50。 (K 常数)
- S、M、T、C、D、V、Z、P 等装置的编号, 例: M10、T30。 (装置编号)
- 在应用指令中作为操作数使用, 例: MOV K123 D0。 (K 常数)

4. BCD (Binary Code Decimal, BCD)

以一个位数或 4 个位来表示一个十进制的数据, 故连续的 16 个位可以表示 4 位数的十进制数值数据。主要用于读取指拨开关的输入数值或将数值数据输出至七段显示驱动器显示之用。

5. 十六进制 (Hexadecimal Number, HEX), 十六进制在 DVP-PM 系统应用的时机如下:

- 在应用指令中作为操作数使用, 例: MOV H1A2B D0。 (H 常数)

常数 K:

十进制数值在 DVP-PM 系统中, 通常会在数值前面以一“K”字表示, 例: K100, 表示为十进制, 其数值大小为 100。

3 DVP-PM 各种装置功能

例外:

当使用 K 再搭配位装置 X、Y、M、S 可组合成为位数、字节、字符组或双字符组形式的数据。例: K2Y10、K4M100。在此 K1 代表一个 4 bits 的组合, K2~K4 分别代表 8、12 及 16 bits 的组合。

常数 H:

16 进位数值在 DVP-PM 中, 通常在其数值前面以一“H”字符表示, 例: H100, 其表示为 16 进位, 数值大小为 256。

浮点数 F:

浮点数值在 DVP-PM 中, 通常在其数值前面以一“F”字符表示, 例: F3.123, 其表示为浮点数值, 数值大小为 3.123。

数值对照表:

二进制 (BIN)		八进位 (OCT)	十进制 (DEC)	BCD (Binary Code Decimal)		16 进位 (HEX)
DVP-PM 内部运算用		装置 X、Y 编号	常数 K 装置 M、S、T、C、D、 V、Z、P 编号	指拨开关及 7 段显示器用		常数 H
0 0 0 0	0 0 0 0	0	0	0 0 0 0	0 0 0 0	0
0 0 0 0	0 0 0 1	1	1	0 0 0 0	0 0 0 1	1
0 0 0 0	0 0 1 0	2	2	0 0 0 0	0 0 1 0	2
0 0 0 0	0 0 1 1	3	3	0 0 0 0	0 0 1 1	3
0 0 0 0	0 1 0 0	4	4	0 0 0 0	0 1 0 0	4
0 0 0 0	0 1 0 1	5	5	0 0 0 0	0 1 0 1	5
0 0 0 0	0 1 1 0	6	6	0 0 0 0	0 1 1 0	6
0 0 0 0	0 1 1 1	7	7	0 0 0 0	0 1 1 1	7
0 0 0 0	1 0 0 0	10	8	0 0 0 0	1 0 0 0	8
0 0 0 0	1 0 0 1	11	9	0 0 0 0	1 0 0 1	9
0 0 0 0	1 0 1 0	12	10	0 0 0 1	0 0 0 0	A
0 0 0 0	1 0 1 1	13	11	0 0 0 1	0 0 0 1	B
0 0 0 0	1 1 0 0	14	12	0 0 0 1	0 0 1 0	C
0 0 0 0	1 1 0 1	15	13	0 0 0 1	0 0 1 1	D
0 0 0 0	1 1 1 0	16	14	0 0 0 1	0 1 0 0	E
0 0 0 0	1 1 1 1	17	15	0 0 0 1	0 1 0 1	F
0 0 0 1	0 0 0 0	20	16	0 0 0 1	0 1 1 0	10
0 0 0 1	0 0 0 1	21	17	0 0 0 1	0 1 1 1	11

二进制 (BIN)		八进位 (OCT)	十进制 (DEC)	BCD (Binary Code Decimal)		16 进位 (HEX)
DVP-PM 内部运算用		装置 X、Y 编号	常数 K 装置 M、S、T、C、D、 V、Z、P 编号	指拨开关及 7 段显示器用		常数 H
：	：	：	：	：	：	：
：	：	：	：	：	：	：
：	：	：	：	：	：	：
：	：	：	：	：	：	：
：	：	：	：	：	：	：
0 1 1 0	0 0 1 1	143	99	1 0 0 1	1 0 0 1	63

3.3 外部输入/输出接点的编号及功能 [X] / [Y]

■ 输入继电器：X0~X377

输入继电器 (或称输入端) 其编号采用八进制编码，最大点数可达 256 点，范围如下：X0~X7，X10~X17，……，X370~X377。

■ 输出继电器：Y0~Y377

输出继电器 (或称输出端) 其编号亦采用八进制编码，最大点数可达 256 点，范围如下：Y0~Y7，Y10~Y17，……，Y370~Y377。

■ 输入接点的功能：

输入接点 X 的功能：输入接点 X 与输入装置连接，读取输入讯号进入 DVP-PM。每一个输入接点 X 的 A 或 B 接点于程序中使用次数没有限制。输入接点 X 之 On/Off 只会跟随输入装置的 On/Off 做变化。

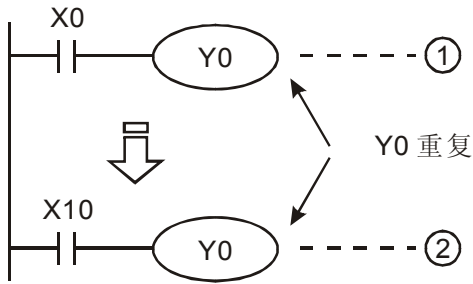
■ M1304 旗标强制 On/Off X 输入点

M1304=Off 时，不可使用周边装置 PMSoft 来强制输入接点 X 之 On/Off。M1304=On 时，允许周边装置 PMSoft 来强制主机输入接点 X 之 On/Off，但此时 DVP-PM 主机外部扫描更新输入点信号功能关闭。

■ 输出接点 Y 的功能：

输出接点 Y 的任务就是送出 On/Off 信号来驱动连接输出接点 Y 的负载。输出接点分成两种，一为继电器 (Relay)，另一为晶体管 (Transistor)，每一个输出接点 Y 的 A 或 B 接点于程序中使用次数没有限制，但输出线圈 Y 的编号，在程序建议仅能使用一次，否则依 DVP-PM 的程序扫描原理，其输出状态的决定权会落在程序中最后的输出 Y 的电路。

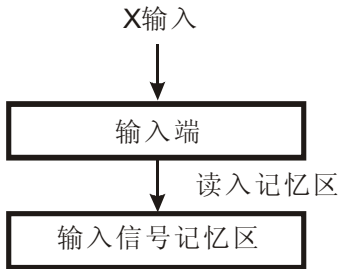
3 DVP-PM 各种装置功能



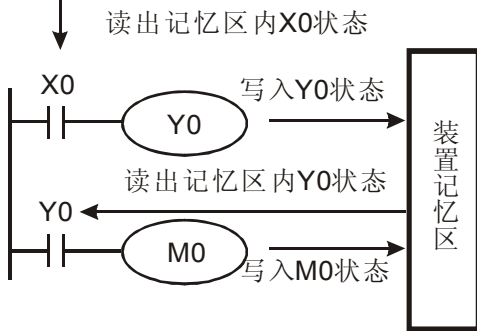
Y0 的输出最后会由电路 ② 决定，亦即由 X10 的 On/Off 决定 Y0 的输出

DVP-PM 对于程序的处理流程(结束再生方式)

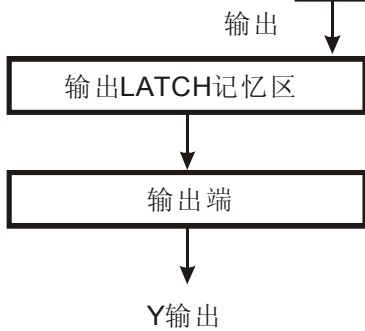
输入信号再生



程序处理



输出再生



● 输入信号再生:

1. DVP-PM 在执行程序之前会将外部输入信号的 On/Off 状态一次读入至输入信号记忆区内。
2. 在程序执行中若输入信号作 On/Off 变化，但是输入信号记忆区内的状态不会改变，一直到下一次扫描开始时再读入输入信号新的 On/Off 状态。
3. 外部信号 On→Off 或 Off→On 变化到程序内接点认定为 On/Off 时期间约有 10ms 的延迟 (但可能会受程序扫描时间的影响)。

● 程序处理:

DVP-PM 读取输入信号记忆区内各输入信号之 On/Off 状态后开始从地址 0 处依序执行程序中的每一指令，其处理结果即各输出线圈之 On/Off 也逐次存入各装置记忆区。

● 输出再生:

1. 当执行到 M102 指令时将装置记忆区内 Y 的 On/Off 状态送到输出 LATCH 记忆区内，而此记忆区就是实际上输出继电器的线圈。
2. 继电器线圈 On→Off 或 Off→On 变化到接点 On/Off 时期间约有 10ms 的延迟。
3. 使用晶体管模块，On→Off 或 Off→On 变化到接点 On/Off 时期间约有 10~20us 的延迟。

3.4 辅助继电器的编号及功能 [M]

辅助继电器的编号：（以 10 进制编号）

辅助继电器 M	一般用	M0~M499, 500 点。可使用参数设定变更成停电保持区域	合计 4,096 点
	停电保持用	M500~M999、M3000~M4095, 1,596 点。可使用参数设定变更成非停电保持区域	
	特殊用	M1000~M2999, 2,000 点。部份为停电保持	

辅助继电器的功能：

辅助继电器 M 与输出继电器 Y 一样有输出线圈及 A、B 接点，而且于程序当中使用次数无限制，使用者可利用辅助继电器 M 来组合控制回路，但无法直接驱动外部负载。依其性质可区分为下列三种：

1. 一般用辅助继电器：一般用辅助继电器于 DVP-PM 运转时若遇到停电，其状态将全部被复归为 Off，再送电时其状态仍为 Off。
2. 停电保持用辅助继电器：停电保持用辅助继电器于 DVP-PM 运转时若遇到停电，其状态将全部被保持，再送电时其状态为停电前状态。
3. 特殊用辅助继电器：每一个特殊用辅助继电器均有其特定之功用，未定义的特殊用辅助继电器请勿使用。各机种之特殊用辅助继电器。请参考 3.10 节特殊继电器及特殊寄存器，3.11 节特殊辅助继电器及特殊寄存器群组功能说明。

3.5 内部继电器的编号及功能 [S]

内部继电器的编号：（以 10 进制编号）

内部继电器 S	一般用	S0~S499, 490 点。可使用参数设定变更成停电保持区域	合计 1,024 点
	停电保持用	S500~S1023, 524 点。可使用参数设定变更成非停电保持区域	

内部继电器的功能：

内部继电器 S 的装置编号为 S0~S1023 共 1024 点，各内部继电器 S 与输出继电器 Y 一样有输出线圈及 A、B 接点，而且于程序当中使用次数无限制，但无法直接驱动外部负载。内部继电器 S 可当作一般的辅助继电器使用。

3.6 定时器的编号及功能 [T]

定时器的编号：（以 10 进制编号）

定时器 T	10ms 一般用	T0~T255, 256 点。可使用参数设定变更成停电保持区域
-------	----------	---------------------------------

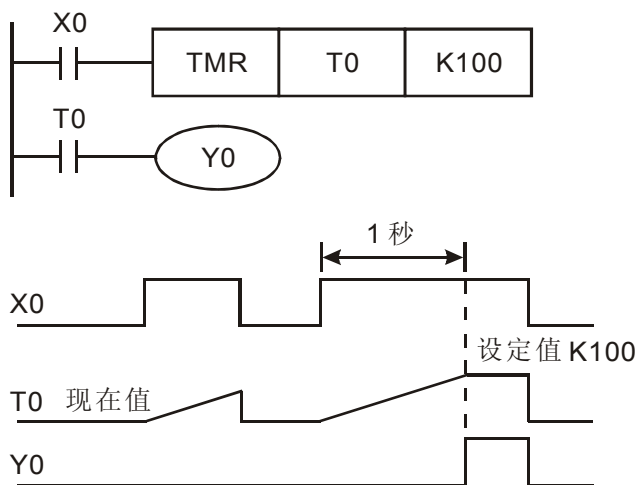
定时器的功能：

定时器是以 10ms 为一个计时单位，计时方式采上数计时，当定时器现在值=设定值时输出线圈导通，设定值为 10 进制 K 值，亦可使用数据寄存器 D 当成设定值。

3 DVP-PM 各种装置功能

定时器之实际设定时间 = 计时单位 * 设定值。

1. 一般用定时器在 TMR 指令执行时计时一次，在 TMR 指令执行时，若计时到达，则输出线圈导通。



- 当 X0=On 时，定时器 T0 之现在值以 10ms 采上数计时，当定时器现在值=设定值 K100 时，输出线圈 T0=On。
- 当 X0=Off 或停电时，定时器 T0 之现在值清除为 0，输出线圈 T0 变为 Off。

设定值的指定方法：定时器之实际设定时间 = 计时单位 * 设定值。

1. 常数指定 K: 设定值直接指定常数 K 值。
2. 间接指定 D: 设定值使用数据寄存器 D 做间接指定。

3.7 计数器的编号及功能 [C]

计数器的编号：(以 10 进制编号)

■ 20PM 机种：

C	计数器	16 位上数	C0 ~ C199, 200 点	合计 248 点	CNT (DCNT) 指令所指定的计数器，若计时到达则此同编号 C 的接点将会 On。
		32 位上/下数	C208 ~C255, 48 点(积算型)		
		32 位高速计数器	C200, C204, 2 点	合计 2 点	C200 输入接点 MPGA0/MPGB0 C204 输入接点 MPGA1/MPGB1

■ 10PM 机种：

C	计数器	16 位上数	C0 ~ C199, 200 点	合计 236 点	CNT (DCNT) 指令所指定的计数器，若计时到达则此同编号 C 的接点将会 On。
		32 位上/下数	C220~C255, 36 点(积算型)		
		32 位高速计数器	C200, C204, C208, C212, C216, C220, 6 点	合计 6 点	C200 输入接点 X0/X1 C204 输入接点 X2/X3 C208 输入接点 X4/X5 C212 输入接点 X6/X7 C216 输入接点 X10+/X10-/X11+/X11- C220 输入接点 X12+/X12-/X13+/X13-

计数器特点：

项目	16 位计数器	32 位计数器	
类型	一般型	一般型	高速型
计数方向	上数	上、下数	
设定值	0 ~ 32,767	-2,147,483,648 ~ +2,147,483,647	
设定值的指定	常数 K 或数据寄存器 D	常数 K 或数据寄存器 D (指定 2 个)	
现在值的变化	计数到达设定值就不再计数	计数到达设定值后，仍继续计数	
输出接点	计数到达设定值，接点导通并保持	上数到达设定值接点导通并保持 On 下数到达设定值接点复归成 Off	
复归动作	RST 指令被执行时现在值归零，接点被复归成 Off		
接点动作	在扫描结束时，统一动作	在扫描结束时，统一动作	计数到达后立即动作，与扫描周期无关

计数器的功能：

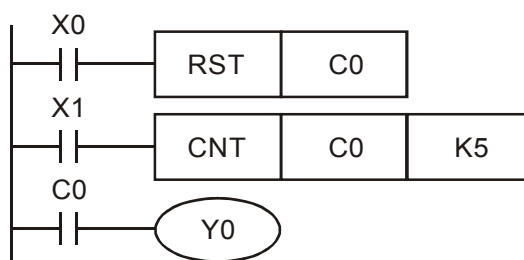
计数器之计数脉冲输入信号由 Off→On 时，计数器现在值等于设定值时输出线圈导通，设定值为 10 进制 K 值，亦可使用数据寄存器 D 当成设定值。

16 位计数器 C0~C199：

- 16 位计数器的设定范围：K0~K32,767。(K0 与 K1 相同，在第一次计数时输出接点马上导通)
- 一般用计数器在 DVP-PM 停电的时候，计数器现在值即被清除，若为停电保持型计数器(积算型)会将停电前的现在值及计数器接点状态记忆着，复电后会继续累计。
- 若使用 MOV 指令、PMSoft 将一个大于设定值的数值传送到 C0 现在值寄存器时，在下次 X1 由 Off→On 时，C0 计数器接点即变成 On，同时现在值内容变成与设定值相同。
- 计数器之设定值可使用常数 K 直接设定或使用寄存器 D (不包含特殊数据寄存器 D1000~D2999)中之数值作间接设定。
- 设定值若使用常数 K 仅可为正数，使用数据寄存器 D 作为设定值可以是正负数。计数器现在值由 32,767 再往上累计时则变为 -32,768。

范例：

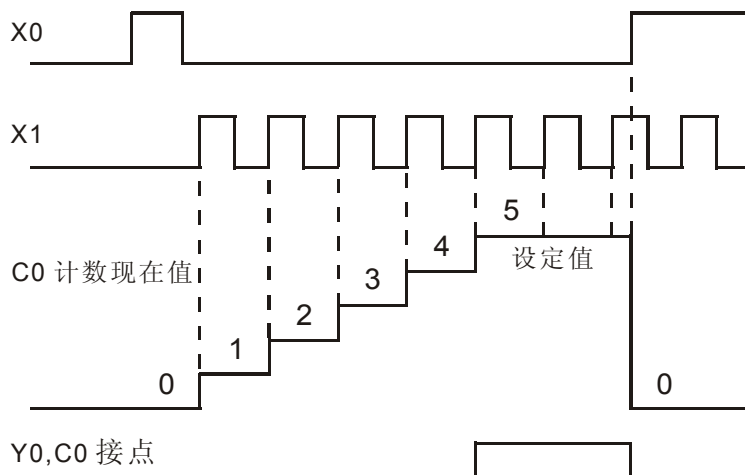
```
LD X0
RST C0
LD X1
CNT C0 K5
LD C0
OUT Y0
```



1. 当 X0=On 时 RST 指令被执行，C0 的现在值归零，输出接点被复归为 Off。

3 DVP-PM 各种装置功能

- 当 X1 由 Off→On 时，计数器之现在值将执行上数 (加一) 的动作。
- 当计数器 C0 计数到达设定值 K5 时，C0 接点导通，C0 现在值=设定值=K5。之后的 X1 触发信号 C0 完全不接受，C0 现在值保持在 K5 处。

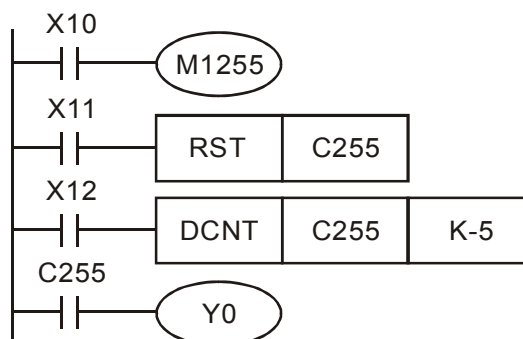


32 位一般用加减计数器 C208~C255:

- 32 位一般用计数器的设定范围：K-2,147,483,648 ~ K2,147,483,647。32 位一般用加减计数器切换上下数用特殊辅助继电器：由 M1208~M1255 来决定。例：M1208=Off 时决定 C208 为加算，M1208=On 时决定 C208 为减算其余类推。
- 设定值可使用常数 K 或使用数据寄存器 D (不包含特殊数据寄存器 D1000~D2999) 作为设定值可以是正负数，若使用数据寄存器 D 则一个设定值占用两个连续的数据寄存器。
- 一般用计数器在 DVP-PM 停电的时候，计数器现在值即被清除，若为停电保持型计数器(积算型)，则会将停电前的现在值及计数器接点状态记忆着，复电后会继续累计。
- 计数器现在值由 2,147,483,647 再往上累计时则变为 -2,147,483,648。同理计数器现在值由 -2,147,483,648 再往下递减时，则变为 2,147,483,647。

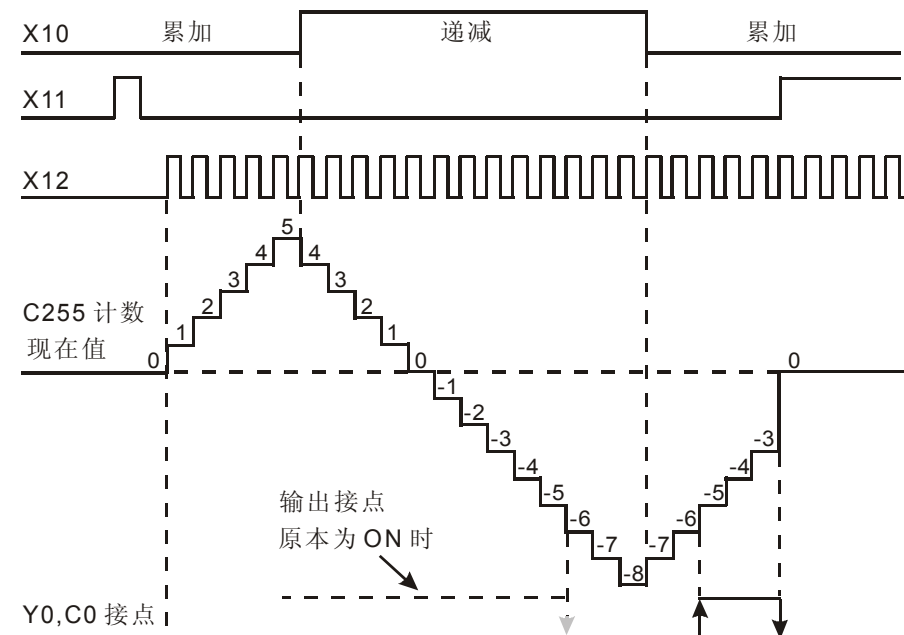
范例:

```
LD    X10
OUT   M1255
LD    X11
RST   C255
LD    X12
DCNT  C255 K-5
LD    C255
OUT   Y0
```



- X10 驱动 M1255 来决定 C255 为加算或减算。
- 当 X11 由 Off→On 时，RST 指令执行，C208 之现在值被清除为 0，且接点变为 Off。
- 当 X12 由 Off→On 时，计数器之现在值将执行上数 (加一) 的动作或下数 (减一) 的动作。
- 当计数器 C255 之现在值从 K-6→K-5 变化时，C255 接点由 Off→On。当计数器 C255 之现在值从 K-5→K-6 变化时，C255 接点由 On→Off。

5. 若使用 MOV 指令、PMSoft 将一个大于设定值的数值传送到 C255 现在值寄存器时，在下次 X1 由 Off→On 时，C255 计数器接点即变成 On，同时现在值内容变成与设定值相同



32 位高速计数器:

■ 20PM 机种(C200、C204):

1. 32 位高速计数器的设定范围: K-2,147,483,648 ~ K2,147,483,647。
2. C200/C204 计数模式设定

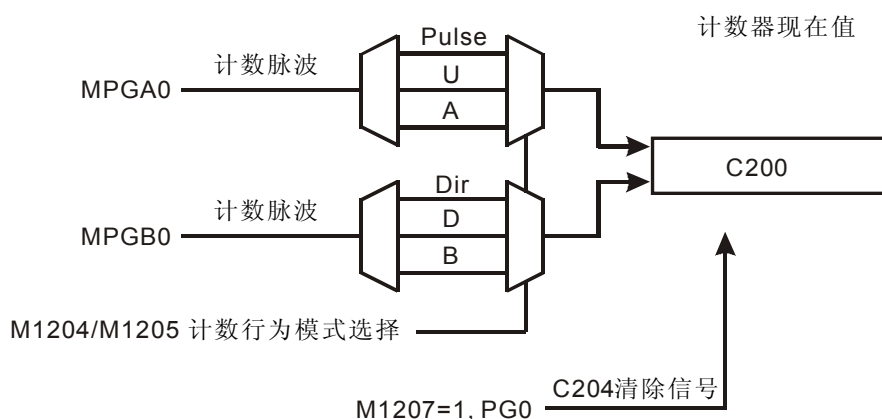
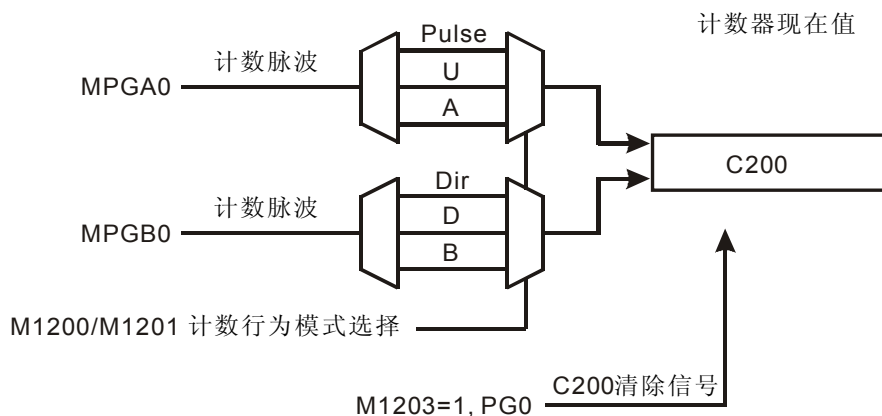
计数器编号	计数模式		清除讯号 设定	清除讯号 外部接脚	外部输入接脚
	装置	设定值			
C200	K1M1200	0: U/D* 1: P/D*	M1203	PG0	A0+、A0-、 B0+、 B0-
C204	K1M1204	2: A/B*(1 倍频) 3: 4A/B(4 倍频)	M1207	PG1	A1+、A1-、 B1+、 B1-

注: U/D: 上数/下数, P/D: 脉冲/方向, A/B: A 相/B 相

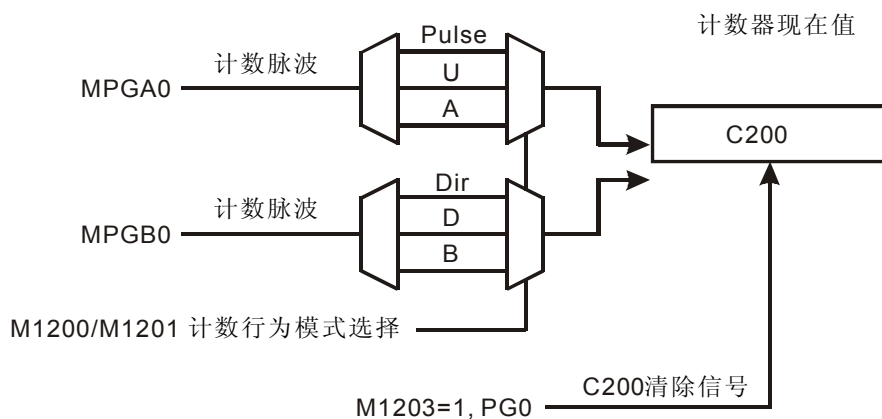
3. 高速计数器 C200/C204 输入信号控制 M1908:

- M1908=Off 时，高速计数器 C200 输入信号由 A0/B0 控制，清除信号由 PG0 控制。高速计数器 C204 输入信号由 A1/B1 控制，清除信号由 PG1 控制。

3 DVP-PM 各种装置功能



- M1908=On 时，XYZ 三轴共享高速计数器 C200，输入信号由 A0/B0 控制。C200 清除信号由 PG0 控制，C204 清除信号由 PG1 控制。



4. 设定值可使用常数 K 或使用数据寄存器 D (不包含特殊数据寄存器 D1000~D2999)。设定值可以是正负数，若使用数据寄存器 D 则一个设定值占用两个连续的数据寄存器。
5. 一般用计数器在 DVP-PM 停电的时候，计数器现在值即被清除，若为停电保持型 (积算型) 计数器，则会将停电前的现在值及计数器接点状态记忆着，复电后会继续累计。
6. 计数器现在值由 2,147,483,647 再往上累计时则变为 -2,147,483,648。同理计数器现在值由 -2,147,483,648 再往下递减时，则变为 2,147,483,647。

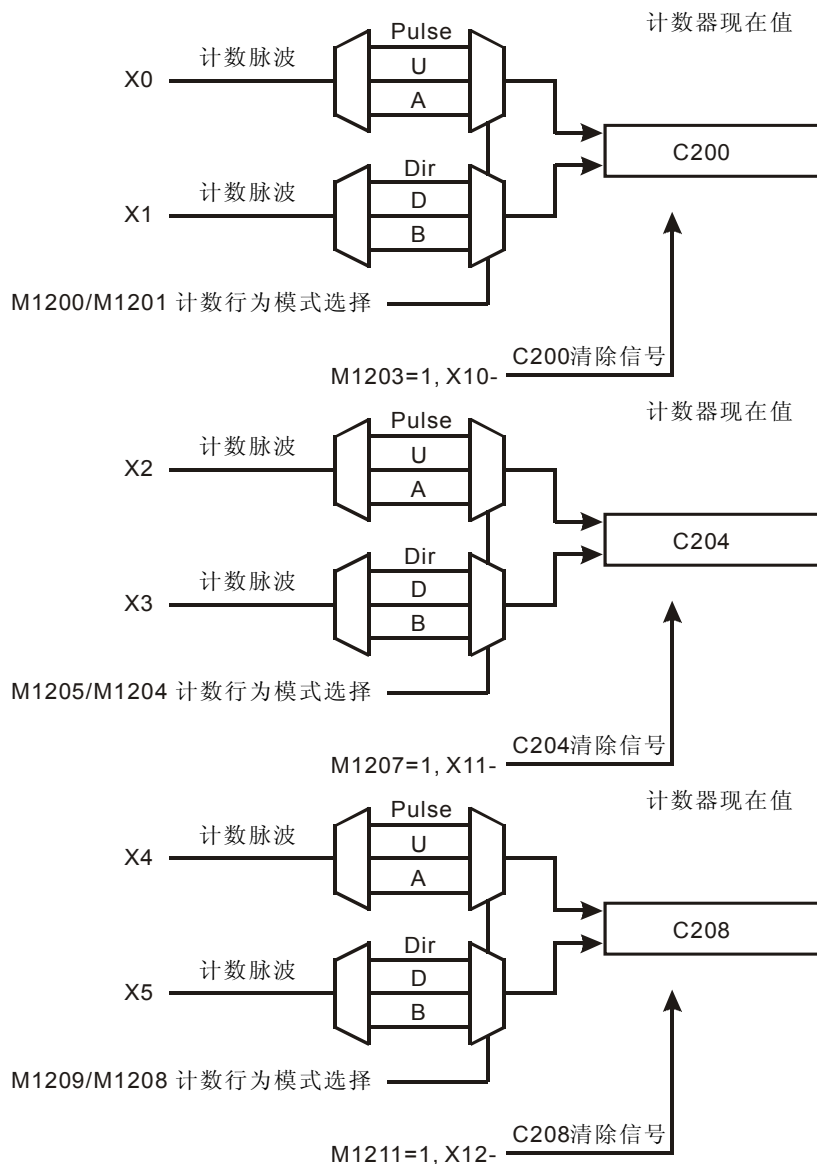
■ 10PM 机种(C200、C204、C208、C212、C216、C220):

1. 32 位高速计数器的设定范围: K-2,147,483,648 ~ K2,147,483,647。

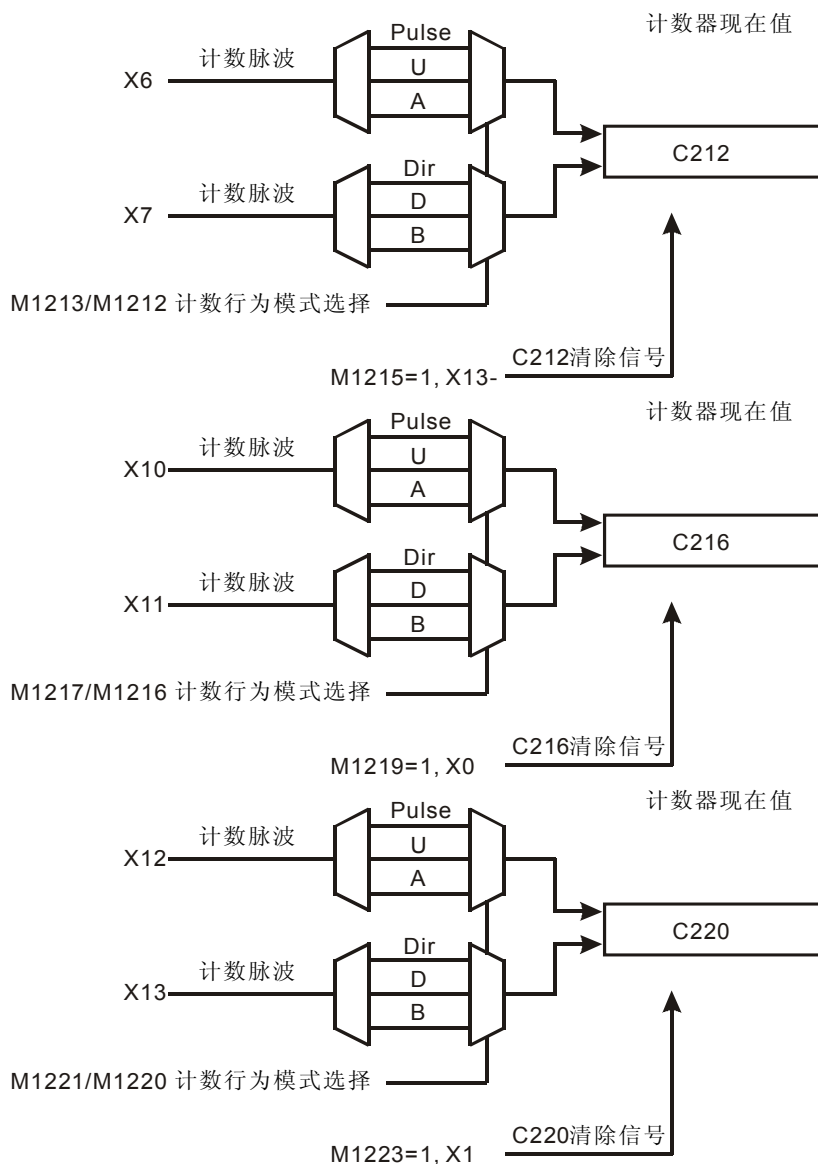
2. 计数模式设定

计数器编号	计数模式		清除讯号设定	清除讯号外部接脚	外部输入接脚
	装置	设定值			
C200	K1M1200	0: U/D* 1: P/D* 2: A/B*(1倍频) 3: 4A/B*(4倍频)	M1203	X10-	X0、X1、S/S
C204	K1M1204		M1207	X11-	X2、X3、S/S
C208	K1M1208		M1211	X12-	X4、X5、S/S
C212	K1M1212		M1215	X13-	X6、X7、S/S
C216	K1M1216		M1219	X0	X10+、X10-、X11+、X11-
C220	K1M1220		M1223	X1	X12+、X12-、X13+、X13-

注: U/D: 上数/下数, P/D: 脉冲/方向, A/B: A相/B相



3 DVP-PM 各种装置功能



3. 设定值可使用常数 K 或使用数据寄存器 D (不包含特殊数据寄存器 D1000~D2999)。设定值可以是正负数，若使用数据寄存器 D 则一个设定值占用两个连续的数据寄存器。
4. 一般用计数器在 DVP-PM 停电的时候，计数器现在值即被清除，若为停电保持型 (积算型) 计数器，则会将停电前的现在值及计数器接点状态记忆着，复电后会继续累计。
5. 计数器现在值由 2,147,483,647 再往上累计时则变为-2,147,483,648。同理计数器现在值由 -2,147,483,648 再往下递减时，则变为 2,147,483,647。

3.8 寄存器的编号及功能 [D]

3.8.1 数据寄存器[D]

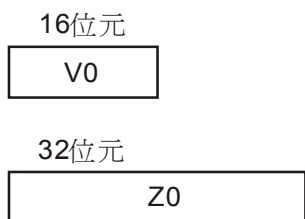
用于储存数值数据，其数据长度为 16 位 (-32,768 ~ +32,767)，最高位为正负号，可储存 -32,768~+32,767 之数值数据，亦可将两个 16 位寄存器合并成一个 32 位寄存器 (D+1,D 编号小的为下 16 位) 使用，而其最高位为正负号，可储存 -2,147,483,648 ~ +2,147,483,647 之数值资料。

数据寄存器 D	一般用	D0~D199, 200 点。可使用参数设定变更成停电保持区域	合计 10,000 点
	停电保持用	D200~D999、D3000~D9999, 7,800 点。 可使用参数设定变更成非停电保持区域	
	特殊用	D1000~D2999, 2,000 点。部份为停电保持	
	间接指定用寄存器 V、Z	V0~V7, Z0~Z7, 16 点	

寄存器依其性质可区分为下列 4 种：

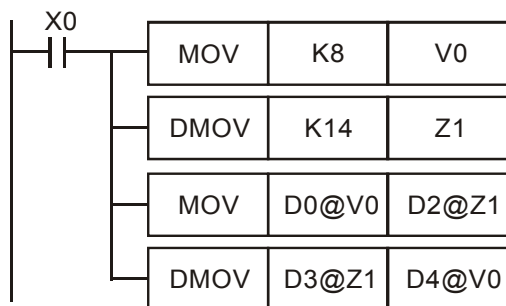
1. 一般用寄存器：当 DVP-PM 由 AUTO→MANU(STOP→RUN)或断电时，寄存器内的数值数据会被清除为 0，如果让 M1033=On 时，则 DVP-PM 由 AUTO→MANU(RUN→STOP)时，数据会保持不被清除，但断电时仍会被清除为 0。
2. 停电保持用寄存器：当 DVP-PM 断电时此区域的寄存器数据不会被清除，仍保持其断电前之数值。清除停电保持用寄存器的内容值，可使用 RST 或 ZRST 指令。
3. 特殊用寄存器：每个特殊用途寄存器都有其特殊定义与用途，主要作为存放系统状态、错误讯息、监视状态之用。请参考 3.10 节特殊继电器及特殊寄存器，3.11 节特殊辅助继电器及特殊寄存器群组功能说明。
4. 间接指定用寄存器 [V]、[Z]：间接指定寄存器 V 为 16 位寄存器，而 Z 为 32 位寄存器，V0 ~ V7、Z0 ~ Z7 共计 16 点。

3.8.2 间接指定用寄存器[V]、[Z]：



V 与一般的数据寄存器一样的都是 16 位的数据寄存器，它可以自由的被写入及读出，若当一般寄存器用，仅能使用在 16 位的指令里。

Z 为 32 位的数据寄存器，若当一般寄存器用，仅能使用在 32 位的指令里。



当 X0=On 时，V0=8、Z1=14。

D8 寄存器内容被搬到 D16; D17 寄存器内容被搬到 D12

如果使用间接指定寄存器 V、Z 来修饰操作数时，16 位指令及 32 位指令，皆可混用。（如左范例）

间接指定寄存器与一般的操作数相同可用来作为搬移或比较，可用于字符装置 (KnX、KnY、KnM、KnS、T、C、D) 及位装置 (X、Y、M、S)。

V0~V7, Z0~Z7 共计 16 点

※ 常数与部分指令并不支持间接指定用法，使用间接指定寄存器 V、Z 来修饰操作数请参考第 5 章 5.4 节。

3 DVP-PM 各种装置功能

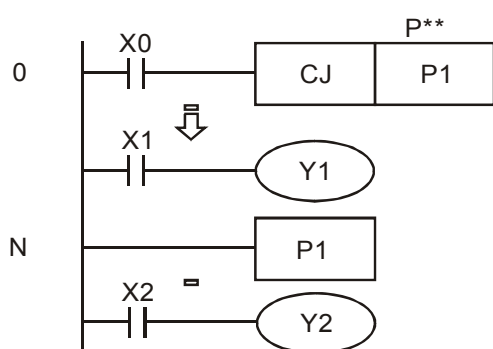
※ 使用间接指定寄存器 V、Z 来修饰操作数时，修饰范围请勿横跨特殊用寄存器 (D1000~D2999) 及特殊辅助继电器 (M1000~M2999) 之区域，以免发生错误。

3.9 指标 [N]、指标 [P_n]

指	N	主控回路用	N0~N7, 8 点	主控回路控制点
标	P	CJ、CJN、JMP 指令用	P0~P255, 256 点	CJ、CJN、JMP 的位置指针

指标 P: 搭配应用指令 API 00 CJ、API 256 CJN、API 257 JMP 使用，详细说明请参考第 5 章节中的 CJ、CJN、JMP 指令使用说明。

● CJ 条件跳跃:



- 当 X0=On 时程序自动从地址 0 跳跃至地址 N (即指定之标签 P1) 继续执行，中间地址跳过不执行。
- 当 X0=Off 时程序如同一般程序由地址 0 继续往下执行，此时 CJ 指令不被执行。

3.10 特殊继电器 M 及特殊寄存器 D

特殊辅助继电器 (特 M) 及特殊数据寄存器 (特 D)，它的种类及功能如下所示。下列各表中，在编号右上角有 “*” 记号可参考下一节的功能说明，像其中属性栏中标示为“R”者，表示仅可作读取的动作，若标示为“R/W”，表示可作读写的动作。另若标示为 “-”，表示无变化。标示为 “#”，则表示系统会依照 DVP-PM 状态作设定，使用者可读取该设定值对照手册之说明，进一步了解系统信息。

特 M	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
M1000*	运转监视常开接点 (a 接点)。AUTO 中常时 On, a 接点。AUTO 的状态下, 此接点 On	○	○	○	Off	On	Off	R	否	Off	3-26
M1001*	运转监视常闭接点 (b 接点)。AUTO 中常时 Off, b 接点。AUTO 的状态下, 此接点 Off	○	○	○	On	Off	On	R	否	On	3-26
M1002*	启始正向 (AUTO 的瞬间'On') 脉冲。初期脉冲, a 接点。AUTO 的瞬间, 产生正向的脉冲, 其宽度 = 扫描周期	○	○	○	Off	On	Off	R	否	Off	3-26
M1003*	启始负向 (AUTO 的瞬间'Off') 脉冲。初期脉冲, a 接点。AUTO 的瞬间, 产生负向的 PULSE, PULSE 的宽度 = 扫描周期	○	○	○	On	Off	On	R	否	On	3-26
M1008	扫描逾时定时器(WDT) On	○	○	○	Off	Off	-	R	否	Off	-
M1009	LV 讯号曾经发生过纪录	○	○	○	Off	-	-	R	否	Off	
M1011	10ms 时钟脉冲, 5ms On/5ms Off	○	○	○	Off	-	-	R	否	Off	-
M1012	100ms 时钟脉冲, 50ms On / 50ms Off	○	○	○	Off	-	-	R	否	Off	-
M1013	1s 时钟脉冲, 0.5s On / 0.5s Off	○	○	○	Off	-	-	R	否	Off	-
M1014	1min 时钟脉冲, 30s On / 30s Off	○	○	○	Off	-	-	R	否	Off	-
M1025	有不正确的通讯服务要求 (当 PC 或 HMI(人机接口)与 DVP-PM 联机时, 在数据的传输当中, 若 DVP-PM 接收到不合法的通讯服务要求时, M1025 会被设定, 且会将错误码存于 D1025)	○	○	○	Off	Off	-	R	否	Off	-
M1026	RAMP 模式选择	○	○	○	Off	-	-	R/W	否	Off	
M1029	脉冲输出 CH0 (Y0, Y1) 执行完毕		○	○	Off	-	-	R	否	Off	
M1031	非停电保持区域全部清除	○	○	○	Off	-	-	R/W	否	Off	-
M1032	停电保持区域全部清除	○	○	○	Off	-	-	R/W	否	Off	-
M1033	非运转中记忆保持	○	○	○	Off	-	-	R/W	否	Off	-
M1034	Y 输出全部禁止	○	○	○	Off	-	-	R/W	否	Off	-
M1035	将 STOP0/START0 当外部 IO 用	○	○	○	Off	Off	Off	R/W	否	Off	-
M1036*	连续路径可跨应用指令	○	○	○	Off	Off	Off	R/W	否	Off	3-32
M1039*	固定扫描时间	○	○	○	Off	-	-	R/W	否	Off	3-29
M1048	警报点状态旗标	○	○	○	Off	-	-	R	否	Off	-
M1049	设定警报点监控旗标	○	○	○	Off	-	-	R/W	否	Off	-
M1072	DVP-PM AUTO 指令执行 (通讯)		○	○	Off	On	Off	R/W	否	Off	-
M1074*	启动 OX 运动子程序		○	○	Off	-	-	R/W	否	Off	3-30

3 DVP-PM 各种装置功能

特 M	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
M1077	电池电压过低、故障或无电池	○	○	○	Off	-	-	R/W	否	Off	-
M1087	LV 讯号动作旗标	○	○	○	Off	-	-	R/W	否	Off	-
M1120*	COM2 (RS-485) 通讯设定保持用, 设定后 D1120 变更无效	○	○	○	Off	Off	-	R/W	否	Off	3-27
M1121	RS-485 通讯数据发送等待	○	○	○	Off	On	-	R	否	Off	-
M1122	送信要求	○	○	○	Off	Off	-	R/W	否	Off	-
M1123	接收完毕	○	○	○	Off	Off	-	R/W	否	Off	-
M1124	接收等待	○	○	○	Off	Off	-	R	否	Off	-
M1125	接收状态解除	○	○	○	Off	Off	-	R/W	否	Off	-
M1127	通讯指令数据传送接收完毕	○	○	○	Off	Off	-	R/W	否	Off	-
M1128	传送中 / 接收中指示	○	○	○	Off	Off	-	R	否	Off	-
M1129	接收逾时	○	○	○	Off	Off	-	R/W	否	Off	-
M1136	COM3 (通讯卡) 通讯设定保持	○	○	○	Off	-	-	R	否	Off	3-27
M1138*	COM1 (RS-232) 通讯设定保持, 设定后 D1036 变更无效	○	○	○	Off	-	-	R/W	否	Off	3-27
M1139*	SLAVE 模式时, COM1 (RS-232) 之 ASCII/RTU 模式选择 (Off: ASCII 模式, On: RTU 模式)。	○	○	○	Off	-	-	R/W	否	Off	3-27
M1140	MODRD/MODWR 数据接收错误	○	○	○	Off	Off	-	R	否	Off	-
M1141	MODRD/MODWR 指令参数错误	○	○	○	Off	Off	-	R	否	Off	-
M1143*	SLAVE 模式时, COM2 (RS-485) 之 ASCII/RTU 模式选择 (Off 时为 ASCII 模式 On 时为 RTU 模式) MASTER 模式时, COM2 (RS-485) 之 ASCII/RTU 模式选择(配合 MODRD/MODWR 指令使用 (Off 时为 ASCII 模式 On 时为 RTU 模式))	○	○	○	Off	-	-	R/W	否	Off	3-27
M1161	8 位处理模式 (On 时 8 位模式 / Off 时 16 位模式)	○	○	○	Off	-	-	R/W	否	Off	-
M1162	SCLP 指令中十进制整数与二进浮点数切换使用旗标, On 时表示二进浮点数, Off 时表示十进制整数	○	○	○	Off	-	-	R/W	否	Off	-
M1168	SMOV 工作模式指定	○	○	○	Off	-	-	R/W	否	Off	-
M1200	C200 计数模式设定	○	○	○	Off	-	-	R/W	否	Off	-
M1201	C200 计数模式设定	○	○	○	Off	-	-	R/W	否	Off	-
M1203	C200 Reset 讯号	○	○	○	Off	-	-	R/W	否	Off	-
M1204	C204 计数模式设定	○	○	○	Off	-	-	R/W	否	Off	-
M1205	C204 计数模式设定	○	○	○	Off	-	-	R/W	否	Off	-
M1207	C204 Reset 讯号	○	○	○	Off	-	-	R/W	否	Off	-
M1208	C208 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1209	C209 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1210	C210 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1211	C211 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-

特 M	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
M1212	C212 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1213	C213 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1214	C214 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1215	C215 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1216	C216 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1217	C217 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1218	C218 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1219	C219 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1220	C220 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1221	C221 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1222	C222 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1223	C223 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1224	C224 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1225	C225 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1226	C226 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1227	C227 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1228	C228 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1229	C229 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1230	C230 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1231	C231 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1232	C232 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1233	C233 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1234	C234 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1235	C235 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1236	C236 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1237	C237 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1238	C238 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1239	C239 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1240	C240 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1241	C241 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1242	C242 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1243	C243 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1244	C244 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1245	C245 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R/W	否	Off	-
M1246	C246 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1247	C247 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1248	C248 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-

3 DVP-PM 各种装置功能

特 M	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
M1249	C249 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1250	C250 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1251	C251 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1252	C252 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1253	C253 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1254	C254 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1255	C255 计数模式设定 (On 时为下数)	○	○	○	Off	-	-	R	否	Off	-
M1303	XCH 指令高低位交换旗标	○	○	○	Off	-	-	R/W	否	Off	
M1304*	X 输入点可强制 On-Off	○	○	○	Off	-	-	R/W	否	Off	3-31
M1744*	OX M code Off 旗标	○	○	○	Off	Off	-	R/W	否	Off	3-31
M1745	OX X 轴原点复归禁能旗标	○	○	○	Off	-	-	R/W	否	Off	-
M1746	非周期式 CAM 启动讯号设定	×	○	○	Off			R/W	否	Off	
M1748	CAM 模式完成停止讯号	×	○	○	Off			R	否	Off	
M1749	凸轮的最高输出频率控制	×	○	○	Off	-	-	R/W	否	Off	-
M1751	X 轴现在位置写入致能	○	○	○	Off	-	-	R/W	否	Off	-
M1752	非零位置起动	×	○	○	Off	-	-	R/W	否	Off	-
	非周期式凸轮遇外部 DOG 立即停止	×									
M1755	非周期式凸轮切换主轴来源	×	○	○	Off	-	-	R/W	否	Off	-
M1756	启动同步输出参考 X0(CLR0=Off 讯号), X1(CLR1=Off 讯号)旗标	×	○	○	Off	-	-	R/W	否	Off	-
M1757	停止凸轮变换解析点(保持目前速度)	×	○	○	Off	-	-	R/W	否	Off	-
M1760	OX 径度/角度使用旗标	○	○	○	Off	-	-	R/W	否	Off	-
M1761	X 轴停止于固定角度	×	○	○	Off	-	-	R/W	否	Off	-
M1792	X 轴 Ready Flag	○	○	○	On	On	On	R	否	On	-
M1793*	X 轴运动错误旗标与清除(X 轴启动时自动清除)	○	○	○	Off	-	-	R/W	否	Off	3-32
M1794*	OX M code On 旗标(OX 启动时自动清除)	○	○	○	Off	-	Off	R	否	Off	3-31
M1795	OX M0 Code On(OX 启动时自动清除)	○	○	○	Off	-	-	R	否	Off	-
M1796	OX M2 Code On(OX 启动时自动清除)	○	○	○	Off	On	-	R	否	Off	-
M1808	OX 零旗标 (OX Zero flag)	○	○	○	Off	-	-	R	否	Off	-
M1809	OX 借位旗标 (OX Borrow flag)	○	○	○	Off	-	-	R	否	Off	-
M1810	OX 进位旗标 (OX Carry flag)	○	○	○	Off	-	-	R	否	Off	-
M1811	OX 浮点数运算错误旗标	○	○	○	Off	Off	-	R	否	Off	-
M1812	电子凸轮初始化旗标	×	○	○	Off			R/W	否	Off	
M1813	电子凸轮周期完成旗标	×	○	○	Off			R/W	否	Off	
M1825	Y 轴原点复归禁能旗标	○	○	○	Off	-	-	R/W	否	Off	-
M1831	Y 轴现在位置写入致能	○	○	○	Off	-	-	R/W	否	Off	-
M1841	Y 轴停止于固定角度	×	○	○	Off	-	-	R/W	否	Off	-
M1872	Y 轴 Ready Flag	○	○	○	On	On	On	R	否	On	

特 M	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
M1873*	Y 轴运动错误旗标与清除 (Y 轴启动时自动清除)	○	○	○	Off	-	-	R	否	Off	3-32
M1908	高速计数器 C200/C204 输入信号控制	×	○	○	Off	-	-	R/W	否	Off	
M1909	虚轴模式设定 1	×	○	○	Off	-	-	R/W	否	Off	
M1910	虚轴模式设定 2	×	○	○	Off	-	-	R/W	否	Off	
M1920	O100 径度/角度使用旗标	○	○	○	Off	-	-	R/W	否	Off	-
M1952	O100 Ready flag	○	○	○	On	Off	On	R	否	On	-
M1953*	O100 错误旗标/清除	○	○	○	Off	-	-	R/W	否	Off	3-35
M1957	AUTO/MANU(STOP/RUN)开关状态旗标	○	○	○	Off	On	-	R	否	Off	-
M1968	O100 零旗标 (O100 Zero flag)	○	○	○	Off	-	-	R	否	Off	-
M1969	O100 借位旗标 (O100 Borrow flag)	○	○	○	Off	-	-	R	否	Off	-
M1970	O100 进位旗标 (O100 Carry flag)	○	○	○	Off	-	-	R	否	Off	-
M1971	O100 浮点数运算错误旗标	○	○	○	Off	-	-	R	否	Off	-
M1985	Z 轴原点复归禁能旗标	×	×	○	Off	-	-	R/W	否	Off	-
M1991	Z 轴现在位置写入致能	○	○	○	Off	-	-	R/W	否	Off	-
M2001	Z 轴停止于固定角度	○	○	○	Off	-	-	R/W	否	Off	-
M2032	Z 轴 Ready Flag	×	×	○	On	On	On	R	否	On	-
M2033*	Z 轴运动错误旗标与清除 (Z 轴启动时自动清除)	×	×	○	Off	-	-	R/W	否	Off	3-32

补充说明：20D=DVP20PM00D, 20M=DVP20PM00M

3 DVP-PM 各种装置功能

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1000*	程序扫描逾时定时器 (WDT) (单位: ms)	○	○	○	200	-	-	R/W	否	200	3-26
D1005	DVP-PM 机种系统程序版本 (以出厂时的版本显示)	○	○	○	#	-	-	R	否	#	-
D1002	程序容量	○	○	○	65535	-	-	R	否	65535	-
D1003	程序内存内容总和	○	○	○	-	-	-	R	是	0	-
D1008	WDT 定时器 On 之 STEP 地址	○	○	○	0	-	-	R	否	0	-
D1010	现在扫描时间 (单位: 1ms)	○	○	○	0	-	-	R	否	0	-
D1011	最小扫描时间 (单位: 1ms)	○	○	○	0	-	-	R	否	0	-
D1012	最大扫描时间 (单位: 1ms)	○	○	○	0	-	-	R	否	0	-
D1020	X0~X7 输入滤波器, 单位 ms	○	○	○	10	-	-	R/W	否	10	-
D1025	通讯要求发生错误时的代码	○	○	○	0	0	-	R	否	0	-
D1036*	COM1 通讯协议	○	○	○	H'86	-	-	R/W	否	H'86	3-27
D1038*	DVP-PM 主机 RS-485 通讯当从站时, 数据响应延迟时间设定, 设定范围 0~3000, 时间定义 (10 ms)	○	○	○	-	-	-	R/W	是	0	3-29
D1039*	固定扫描时间 (ms)	○	○	○	0	-	-	R/W	否	0	3-29
D1050 ↓ D1055	Modbus 通讯指令数据处理, DVP-PM 系统会自动将 D1070~D1085 的 ASCII 字符数据转换为 HEX, 16 进位数值	○	○	○	0	-	-	R	否	0	-
D1056	功能卡 2AD CH0 现在值	○	○	○	0	#	-	R	否	0	-
D1057	功能卡 2AD CH1 现在值	○	○	○	0	#	-	R	否	0	-
D1070 ↓ D1085	Modbus 通讯指令数据处理, DVP-PM 内建 RS-485 通讯便利指令, 该指令执行时所送出指令, 当受信端接收后会回传讯息, 该讯息会储存于 D1070~D1085, 使用者可利用该寄存器的内容, 检视回传数据	○	○	○	0	-	-	R	否	0	-
D1089 ↓ D1099	Modbus 通讯指令数据处理, DVP-PM 内建 RS-485 通讯便利指令, 该指令执行时所送出的指令字符储存于 D1089~D1099, 使用者可根据该寄存器的内容, 检视指令是否正确	○	○	○	0	-	-	R	否	0	-
D1109	COM3 (通讯卡) 通讯协议	○	○	○	H'86	-	-	R/W	否	H'86	3-27
D1110	功能卡 2AD CH0 平均值	○	○	○	0	#	-	R	否	0	-
D1111	功能卡 2AD CH1 平均值	○	○	○	0	#	-	R	否	0	-
D1116	功能卡 2DA CH0	○	○	○	0	-	-	R/W	否	0	-
D1117	功能卡 2DA CH1	○	○	○	0	-	-	R/W	否	0	-
D1120*	COM2 (RS-485) 通讯协议	○	○	○	H'86	-	-	R/W	否	H'86	3-27
D1121	DVP-PM 通讯地址	○	○	○	-	-	-	R/W	是	1	-
D1122	发送资料剩余字数	○	○	○	0	0	-	R	否	0	-
D1123	接收数据剩余字数	○	○	○	0	0	-	R	否	0	-
D1124	起始字符定义(STX)	○	○	○	H'3A	-	-	R/W	否	H'3A	

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1125	第一结束字符定义(END High)	○	○	○	H'0D	-	-	R/W	否	H'0D	
D1126	第二结束字符定义(END Low)	○	○	○	H'0A	-	-	R/W	否	H'0A	
D1129	通讯逾时异常, 时间定义 (ms)	○	○	○	0	-	-	R/W	否	0	-
D1130	RS-485 当主机时从机 MODBUS 回传错误码记录	○	○	○	0	0	-	R	否	0	-
D1140*	右侧模块台数, 最多 8 台	○	○	○	0	-	-	R	否	0	3-31
D1142*	数字模块 X 点数	○	○	○	0	-	-	R	否	0	3-31
D1143*	数字模块 Y 点数	○	○	○	0	-	-	R	否	0	3-31
D1149	功能卡 ID (0: 没有插卡, 3: COM3 卡, 8: 2AD 卡, 9: 2DA 卡)	○	○	○	0	-	-	R	否	0	-
D1200*	M 辅助继电器停电保持开始地址	○	○	○	-	-	-	R/W	是	500	3-31
D1201*	M 辅助继电器停电保持结束地址	○	○	○	-	-	-	R/W	是	999	3-31
D1202*	T 定时器停电保持开始地址	○	○	○	-	-	-	R/W	是	-1	3-31
D1203*	T 定时器停电保持结束地址	○	○	○	-	-	-	R/W	是	-1	3-31
D1204*	C16 位计数器停电保持开始地址	○	○	○	-	-	-	R/W	是	100	3-31
D1205*	C16 位计数器停电保持结束地址	○	○	○	-	-	-	R/W	是	199	3-31
D1206*	C32 位计数器停电保持开始地址	○	○	○	-	-	-	R/W	是	220	3-31
D1207*	C32 位计数器停电保持结束地址	○	○	○	-	-	-	R/W	是	255	3-31
D1208*	S 辅助继电器停电保持开始地址	○	○	○	-	-	-	R/W	是	500	3-31
D1209*	S 辅助继电器停电保持结束地址	○	○	○	-	-	-	R/W	是	1023	3-31
D1210*	D 寄存器停电保持开始地址	○	○	○	-	-	-	R/W	是	200	3-31
D1211*	D 寄存器停电保持结束地址	○	○	○	-	-	-	R/W	是	9999	3-31
D1320*	第一台右侧模块代号	○	○	○	0	-	-	R	否	0	3-31
D1321*	第二台右侧模块代号	○	○	○	0	-	-	R	否	0	3-31
D1322*	第三台右侧模块代号	○	○	○	0	-	-	R	否	0	3-31
D1323*	第四台右侧模块代号	○	○	○	0	-	-	R	否	0	3-31
D1324*	第五台右侧模块代号	○	○	○	0	-	-	R	否	0	3-31
D1325*	第六台右侧模块代号	○	○	○	0	-	-	R	否	0	3-31
D1326*	第七台右侧模块代号	○	○	○	0	-	-	R	否	0	3-31
D1327*	第八台右侧模块代号	○	○	○	0	-	-	R	否	0	3-31
D1328*	G-Code G00、G01 第三轴控制, 目标位置(Low word)	×	×	○	0	-	-	R/W	否	否	6-47
D1329*	G-Code G00、G01 第三轴控制, 目标位置(High word)										
D1330*	G-Code G00 第三轴控制, 做为条件判断(Low word)	×	×	○	0	-	-	R/W	否	否	6-47
	G-Code G01 第三轴控制, 补间运转速度(Low word)										6-49

3 DVP-PM 各种装置功能

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1331*	G-Code G00 第三轴控制, 做为条件判断(High word)	×	×	○	0	-	-	R/W	否	否	6-47
	G-Code G01 第三轴控制, 补间运转速度(High word)										6-49
D1400	中断致能	○	○	○	0	-	-	R/W	否	0	
D1401	时间中断周期 (ms)	○	○	○	0	-	-	R/W	否	0	
D1500	FROM/TO 数据区块, 对应 CR#0	○	○	○	H6260	-	-	R	否	H6260	-
D1501 ↓ D1699	FROM/TO 数据区块, 对应 CR#1~CR#199	○	○	○	0	-	-	R/W	否	0	-
D1700	OX 程序执行编号	○	○	○	0	-	-	R	否	0	-
D1702	OX 程序执行 STEP 号码	○	○	○	0	-	-	R	否	0	-
D1703*	OX 执行 M 码	○	○	○	0	-	-	R	否	0	3-31
D1704	OX 等待时间设定值	○	○	○	0	-	-	R	否	0	-
D1705	OX 等待时间目前值	○	○	○	0	-	-	R	否	0	-
D1706	OX RPT 指令设定值	○	○	○	0	-	-	R	否	0	-
D1707	OX RPT 指令目前值	○	○	○	0	-	-	R	否	0	-
D1708	X 轴行走距离补偿值 (Low word)	○	○	○	0	-	-	R	否	0	-
D1709	X 轴行走距离补偿值 (High word)										
D1710	X 轴中心点补偿值 (Low word)	×	○	○	0	-	-	R	否	0	-
D1711	X 轴中心点补偿值 (High word)										
D1712	圆弧半径补偿值 (Low word)	×	○	○	0	-	-	R	否	0	-
D1713	圆弧半径补偿值 (High word)										
D1724	Y 轴行走距离补偿值(Low word)	○	○	○	0	-	-	R	否	0	-
D1725	Y 轴行走距离补偿值(High word)										
D1726	Y 轴中心点补偿值(Low word)	×	○	○	0	-	-	R	否	0	-
D1727	Y 轴中心点补偿值(High word)										
D1736	O100 等待时间 (TIM) 设定值	○	○	○	0	-	-	R	否	0	-
D1737	O100 等待时间 (TIM) 目前值	○	○	○	0	-	-	R	否	0	-
D1738	O100 RPT 指令设定值	○	○	○	0	-	-	R	否	0	-
D1739	O100 RPT 指令目前值	○	○	○	0	-	-	R	否	0	-
D1796*	连续补间减速速度	○	○	○	0	-	-	R	否	0	3-32
D1798*	G-Code 移动速度百分比设定	○	○	○	0	-	-	R	否	0	3-33
D1799*	输入端子极性	○	○	○	0	-	-	R/W	否	0	3-33
D1800*	输入端子状态	○	○	○	0	-	-	R	否	0	3-34
D1802*	O100 错误编号	○	○	○	0	-	-	R/W	否	0	3-35
D1803*	O100 错误 STEP 位置	○	○	○	0	0	-	R/W	否	0	3-35

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1804*	输入端子极性 2	×	×	○	0	-	-	R/W	否	0	3-34
D1805*	输入端子状态 2	×	×	○	0	-	-	R	否	0	3-35
D1806	High Byte 为 MPG0/1 滤波系数, D1806 Low Byte 为其它输入端子(除了 X 点)滤波系数, 滤波频率=85000/2N+4(kHz)	○	○	○	0	-	-	R/W	否	0	-
D1816*	X 轴参数设定	○	○	○	-	-	-	R/W	是	0	3-35
D1818	X 轴马达转一圈所需脉冲数 (Low word)	○	○	○	-	-	-	R/W	是	2000	-
D1819	X 轴马达转一圈所需脉冲数 (High word)	○	○	○	-	-	-	R/W	是	1000	-
D1820	X 轴马达转一圈之移动距离 (Low word)	○	○	○	-	-	-	R/W	是	500K	-
D1821	X 轴马达转一圈之移动距离 (High word)	○	○	○	-	-	-	R/W	是	0	-
D1822	X 轴最高速度 V _{MAX} (Low word)	○	○	○	-	-	-	R/W	是	5000	-
D1823	X 轴最高速度 V _{MAX} (High word)	○	○	○	-	-	-	R/W	是	50K	-
D1824	X 轴启动速度 V _{BIAS} (Low word)	○	○	○	-	-	-	R/W	是	1000	-
D1825	X 轴启动速度 V _{BIAS} (High word)	○	○	○	-	-	-	R/W	是	0	-
D1826	X 轴寸动 JOG 速度 V _{JOG} (Low word)	○	○	○	-	-	-	R/W	是	5000	-
D1827	X 轴寸动 JOG 速度 V _{JOG} (High word)	○	○	○	-	-	-	R/W	是	50K	-
D1828	X 轴原点复归速度 V _{RT} (Low word)	○	○	○	-	-	-	R/W	是	1000	-
D1829	X 轴原点复归速度 V _{RT} (High word)	○	○	○	-	-	-	R/W	是	0	-
D1830	X 轴原点复归减速速度 V _{CR} (Low word)	○	○	○	-	-	-	R/W	是	0	-
D1831	X 轴原点复归减速速度 V _{CR} (High word)	×	○	○	-	-	-	R/W	是	0	3-36
D1832*	X 轴零点信号数 N	×	○	○	-	-	-	R/W	是	0	3-36
	非周期式电子凸轮重复次数设定	○	○	○	-	-	-	R/W	是	0	-
D1833*	X 轴补充距离 P	×	○	○	-	-	-	R/W	是	0	3-36
	电子凸轮脉冲余数分配设定	○	○	○	-	-	-	R/W	是	0	-
D1834*	X 轴原点位置定义 HP (Low word)	×	○	○	-	-	-	R/W	是	0	3-36
	电子凸轮启动延迟脉冲设定	○	○	○	-	-	-	R/W	是	0	-
D1835	X 轴原点位置定义 HP (High word)	×	○	○	-	-	-	R/W	是	0	-
	电子凸轮启动延迟脉冲(CAM offset) 设定	○	○	○	-	-	-	R/W	是	500	-
D1836	X 轴加速时间 T _{ACC}	○	○	○	-	-	-	R/W	是	500	-
D1837	X 轴减速时间 T _{DEC}	○	○	○	-	-	-	R/W	是	0	-
	X 轴目标位置(I) P(I) (Low word)	○	×	×	○	-	-	R/W	否	0	-
D1838	CAM 同步位置下限	○	○	○	○	-	-	R/W	否	0	-
	X 轴 PWM 频宽设定(Low word)	○	○	○	○	-	-	R/W	否	0	-
	X 轴目标位置(I) P(I) (High word)	○	○	○	○	-	-	R/W	否	0	-
D1839	CAM 同步位置下限	×	○	○	○	-	-	R/W	否	0	-
	X 轴 PWM 频宽设定(High word)	○	×	×	○	-	-	R/W	否	0	-

3 DVP-PM 各种装置功能

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1840	X 轴运转速度(I) V(I) (Low word)	○	○	○	1000	-	-	R/W	否	1000	-
	电子凸轮最高输出频率设定	-0	○	○	1000	-	-	R/W	否	1000	-
D1841	X 轴运转速度(I) V(I) (High word)	○	○	○	1000	-	-	R/W	否	1000	-
	电子凸轮最高输出频率设定	×	○	○	1000	-	-	R/W	否	1000	-
D1842	X 轴目标位置(II) P(II) (Low word)	○	○	○	0	-	-	R/W	否	0	-
	CAM 同步位置上限	×	○	○	0	-	-	R/W	否	0	-
	X 轴 PWM 输出周期设定(Low word)	○	×	×	0	-	-	R/W	否	0	-
D1843	X 轴目标位置(II) P(II) (High word)	○	○	○	0	-	-	R/W	否	0	-
	CAM 同步位置上限	×	○	○	0	-	-	R/W	否	0	-
	X 轴 PWM 输出周期设定(Low word)	○	×	×	0	-	-	R/W	否	0	-
D1844	X 轴运转速度(II)V(II) (Low word)	○	○	○	2000	-	-	R/W	否	2000	-
D1845	X 轴运转速度(II)V(II) (High word)										
D1846*	X 轴运转命令	○	○	○	0	-	0	R/W	否	0	3-38
	电子凸轮启动设定	○	○	○	0	-	0	R/W	否	0	3-38
D1847*	X 轴工作模式	×	○	○	0	-	-	R/W	否	0	3-37
D1848	X 轴现在位置 CP(PLS) (Low word)	○	○	○	0	-	-	R/W	否	0	-
	从轴位置										
D1849	X 轴现在位置 CP(PLS) (High word)	○	○	○	0	-	-	R/W	否	0	-
	从轴位置										
D1850	X 轴现在速度(PPS) (Low word)	○	○	○	0	0	0	R/W	否	0	-
D1851	X 轴现在速度(PPS) (High word)										
D1852	X 轴现在位置 CP(unit) (Low word)	○	○	○	0	-	-	R/W	否	0	-
D1853	X 轴现在位置 CP(unit) (High word)										
D1854	X 轴现在速度 CS (unit) (Low word)	○	○	○	0	0	0	R/W	否	0	-
D1855	X 轴现在速度 CS (unit) (High word)										
D1856*	X 轴执行状态	○	○	○	0	-	-	R	否	0	3-39
D1857*	OX、X 轴错误编号	○	○	○	0	-	-	R	否	0	3-32
D1858	X 轴电子齿轮 (分子)	○	○	○	-	-	-	R/W	是	1	-
D1859	X 轴电子齿轮 (分母)	○	○	○	-	-	-	R/W	是	1	-
D1860	X 轴手摇轮输入频率 (Low word)	○	○	○	0	0	-	R/W	否	0	-
	主轴频率										
D1861	X 轴手摇轮输入频率 (High word)	○	○	○	0	0	-	R/W	否	0	-
	主轴频率										
D1862	X 轴累计手摇轮输入脉冲个数 (Low word)	○	○	○	0	-	-	R/W	否	0	-
	主轴位置										
D1863	X 轴累计手摇轮输入脉冲个数 (High word)	○	○	○	0	-	-	R/W	否	0	-
	主轴位置										

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1864*	X 轴手摇轮输入响应速度	○	○	○	-	-	-	R/W	是	5	-
	电子凸轮主轴接收脉冲型式设定	×	○	○	-	-	-	R/W	是	5	3-62
D1865	OX 0~99 停止模式 (K1: 下次启动走完未完行程, K2: 下次启动执行下一指令, 其它: 重新开始)	○	○	○	-	-	-	R/W	是	0	-
D1866	X 轴电气零点 (Low word)	○	○	○	-	-	-	R/W	是	0	-
	非周期式凸轮主轴切换范围	×	○	○	-	-	-	R/W	是	0	-
D1867	X 轴电气零点 (High word)	○	○	○	-	-	-	R/W	是	0	-
	非周期式凸轮主轴切换范围	×	○	○	-	-	-	R/W	是	0	-
D1868*	设定 OX 程序号码	○	○	○	0	0	-	R/W	是	0	3-30
	电子凸轮图表指定	×	○	○	0	0	-	R/W	是	0	3-30
D1869	OX 错误 STEP 位置	○	○	○	0	-	-	R/W	否	0	-
D1872	OX Ready 时, 致能 Y 输出。 high byte 设定 K1; low byte : 指定 Y 输出起始编号	○	○	○	0	-	-	R/W	否	0	-
D1873	OX 执行 M code 时, 致能 Y 输出。 high byte 设定 K1; low byte : 指定 Y 输出起始编号	○	○	○	-	-	-	R/W	是	0	-
D1874	设定 X 输入信号编号, 使用此 X 输入讯号重置 M code	○	○	○	0	-	-	R/W	否	0	-
D1875*	X 轴外部手动启动 (ZRN, MPG, JOG-, JOG+)	○	○	○	-	-	-	R/W	是	0	3-39
D1877	X 轴 PID 闭回路控制	×	○	○	-	-	-	R/W	否	0	
D1878	X 轴闭回路命令现在值(Low word)	×	○	○	-	-	-	R	否	0	
D1879	X 轴闭回路命令现在值(High word)	×	○	○	-	-	-	R	否	0	
D1880	X 轴闭回路 P 值(Low word)	×	○	○	-	-	-	R/W	否	0	
D1881	X 轴闭回路 P 值(High word)	×	○	○	-	-	-	R/W	否	0	
D1882	X 轴闭回路 I 值(Low word)	×	○	○	-	-	-	R/W	否	0	
D1883	X 轴闭回路 I 值(High word)	×	○	○	-	-	-	R/W	否	0	
D1884	X 轴闭回路差异值(Low word)	×	○	○	-	-	-	R	否	0	
D1885	X 轴闭回路差异值(High word)	×	○	○	-	-	-	R	否	0	
D1886	X 轴闭回路累积差异值(Low word)	×	○	○	-	-	-	R	否	0	
D1887	X 轴闭回路累积差异值(High word)	×	○	○	-	-	-	R	否	0	
D1888	X 轴闭回路累积最大差异值(Low word)	×	○	○	-	-	-	R	否	0	
D1889	X 轴闭回路累积最大差异值(High word)	×	○	○	-	-	-	R	否	0	
D1896*	Y 轴参数设定	○	○	○	-	-	-	R/W	是	0	3-35
D1898	Y 轴马达转一圈所需脉冲数 (Low word)	○	○	○	-	-	-	R/W	是	2000	-
D1899	Y 轴马达转一圈所需脉冲数 (High word)										
D1900	Y 轴马达转一圈之移动距离 (Low word)	○	○	○	-	-	-	R/W	是	1000	-
D1901	Y 轴马达转一圈之移动距离 (High word)										

3 DVP-PM 各种装置功能

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1902	Y 轴最高速度 V _{MAX} (Low word)	○	○	○	-	-	-	R/W	是	500K	-
D1903	Y 轴最高速度 V _{MAX} (High word)										
D1904	Y 轴启动速度 V _{BIAS} (Low word)	○	○	○	-	-	-	R/W	是	0	-
D1905	Y 轴启动速度 V _{BIAS} (High word)										
D1906	Y 轴寸动 JOG 速度 V _{JOG} (Low word)	○	○	○	-	-	-	R/W	是	5000	-
D1907	Y 轴寸动 JOG 速度 V _{JOG} (High word)										
D1908	Y 轴原点复归速度 V _{RT} (Low word)	○	○	○	-	-	-	R/W	是	50K	-
D1909	Y 轴原点复归速度 V _{RT} (High word)										
D1910	Y 轴原点复归减速速度 V _{CR} (Low word)	○	○	○	-	-	-	R/W	是	1000	-
D1911	Y 轴原点复归减速速度 V _{CR} (High word)										
D1912	Y 轴零点信号数 N	○	○	○	-	-	-	R/W	是	0	-
D1913	Y 轴补充距离 P	○	○	○	-	-	-	R/W	是	0	-
D1914	Y 轴原点位置定义 HP (Low word)	○	○	○	-	-	-	R/W	是	0	-
D1915	Y 轴原点位置定义 HP (High word)										
D1916	Y 轴加速时间 T _{ACC}	○	○	○	-	-	-	R/W	是	500	-
D1917	Y 轴减速时间 T _{DEC}	○	○	○	-	-	-	R/W	是	500	-
D1918	Y 轴目标位置(I) P(I) (Low word)	○	○	○	0	-	-	R/W	否	0	-
	Y 轴 PWM 频宽设定(Low word)	○	×	×	0	-	-	R/W	否	0	-
D1919	Y 轴目标位置(I) P(I) (High word)	○	○	○	0	-	-	R/W	否	0	-
	Y 轴 PWM 频宽设定(High word)	○	×	×	0	-	-	R/W	否	0	-
D1920	Y 轴运转速度(I) V(I) (Low word)	○	○	○	1000	-	-	R/W	否	1000	-
D1921	Y 轴运转速度(I) V(I) (High word)										
D1922	Y 轴目标位置(II) P(II) (Low word)	○	○	○	0	-	-	R/W	否	0	-
	Y 轴 PWM 输出周期设定(Low word)	○	×	×	0	-	-	R/W	否	0	-
D1923	Y 轴目标位置(II) P(II) (High word)	○	○	○	0	-	-	R/W	否	0	-
	Y 轴 PWM 输出周期设定(High word)	○	×	×	0	-	-	R/W	否	0	-
D1924	Y 轴运转速度(II)V(II) (Low word)	○	○	○	2000	-	-	R/W	否	2000	-
D1925	Y 轴运转速度(II)V(II) (High word)										
D1926*	Y 轴运转命令	○	○	○	0	-	0	R/W	否	0	3-38
D1927*	Y 轴工作模式	×	○	○	0	-	-	R/W	否	0	3-38
D1928	Y 轴现在位置 CP(PLS) (Low word)	○	○	○	0	-	-	R/W	否	0	-
D1929	Y 轴现在位置 CP(PLS) (High word)										
D1930	Y 轴现在速度(PPS) (Low word)	○	○	○	0	0	0	R/W	否	0	-
D1931	Y 轴现在速度(PPS) (High word)										
D1932	Y 轴现在位置 CP(unit) (Low word)	○	○	○	0	-	-	R/W	否	0	-
D1933	Y 轴现在位置 CP(unit) (High word)										

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1934	Y 轴现在速度 CS (unit) (Low word)	○	○	○	0	0	0	R/W	否	0	-
D1935	Y 轴现在速度 CS (unit) (High word)										
D1936*	Y 轴执行状态	○	○	○	0	-	-	R	否	0	3-39
D1937*	Y 轴错误编号	○	○	○	0	-	-	R	否	0	3-32
D1938	Y 轴电子齿轮 (分子)	○	○	○	-	-	-	R/W	是	1	-
D1939	Y 轴电子齿轮 (分母)	○	○	○	-	-	-	R/W	是	1	-
D1940	Y 轴手摇轮输入频率 (Low word)	×	○	○	0	-	0	R/W	否	0	-
D1941	Y 轴手摇轮输入频率 (High word)										
D1942	Y 轴累计手摇轮输入脉冲个数 (Low word)	○	○	○	0	-	-	R/W	否	0	-
D1943	Y 轴累计手摇轮输入脉冲个数 (High word)										
D1944	Y 轴手摇轮输入响应速度	×	○	○	-	-	-	R/W	是	5	-
D1946	Y 轴电气零点 (Low word)	○	○	○	-	-	-	R/W	是	0	-
D1947	Y 轴电气零点 (High word)										
D1955*	Y 轴外部手动启动 (ZRN, MPG, JOG-, JOG+)	○	○	○	-	-	-	R/W	是	4	3-39
D1957	Y 轴 PID 闭回路控制	×	○	○	-	-	-	R/W	否	0	
D1958	Y 轴闭回路命令现在值(Low word)	×	○	○	-	-	-	R	否	0	
D1959	Y 轴闭回路命令现在值(High word)	×	○	○	-	-	-	R	否	0	
D1960	Y 轴闭回路 P 值(Low word)	×	○	○	-	-	-	R/W	否	0	
D1961	Y 轴闭回路 P 值(High word)	×	○	○	-	-	-	R/W	否	0	
D1962	Y 轴闭回路 I 值(Low word)	×	○	○	-	-	-	R/W	否	0	
D1963	Y 轴闭回路 I 值(High word)	×	○	○	-	-	-	R/W	否	0	
D1964	Y 轴闭回路差异值(Low word)	×	○	○	-	-	-	R	否	0	
D1965	Y 轴闭回路差异值(High word)	×	○	○	-	-	-	R	否	0	
D1966	Y 轴闭回路累积差异值(Low word)	×	○	○	-	-	-	R	否	0	
D1967	Y 轴闭回路累积差异值(High word)	×	○	○	-	-	-	R	否	0	
D1968	Y 轴闭回路累积最大差异值(Low word)	×	○	○	-	-	-	R	否	0	
D1969	Y 轴闭回路累积最大差异值(High word)	×	○	○	-	-	-	R	否	0	
D1976	Z 轴参数设定	○	×	○	-	-	-	R/W	是	0	-
D1978	Z 轴马达转一圈所需脉冲数 (Low word)	○	×	○	-	-	-	R/W	是	2000	-
D1979	Z 轴马达转一圈所需脉冲数 (High word)										
D1980	Z 轴马达转一圈之移动距离 (Low word)	○	×	○	-	-	-		是	1000	-
D1981	Z 轴马达转一圈之移动距离 (High word)										
D1982	Z 轴最高速度 V _{MAX} (Low word)	○	×	○	-	-	-	R/W	是	500K	-
D1983	Z 轴最高速度 V _{MAX} (High word)										
D1984	Z 轴启动速度 V _{BIAS} (Low word)	○	×	○	-	-	-	R/W	是	0	-
D1985	Z 轴启动速度 V _{BIAS} (High word)										

3 DVP-PM 各种装置功能

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1986	Z 轴寸动 JOG 速度 V _{JOG} (Low word)	○	×	○	-	-	-	R/W	是	5000	-
D1987	Z 轴寸动 JOG 速度 V _{JOG} (High word)										
D1988	Z 轴原点复归速度 V _{RT} (Low word)	○	×	○	-	-	-	R/W	是	50K	-
D1989	Z 轴原点复归速度 V _{RT} (High word)										
D1990	Z 轴原点复归减速速度 V _{CR} (Low word)	○	×	○	-	-	-	R/W	是	1000	-
D1991	Z 轴原点复归减速速度 V _{CR} (High word)										
D1992	Z 轴零点信号数 N	○	×	○	-	-	-	R/W	是	0	-
D1993	Z 轴补充距离 P	○	×	○	-	-	-	R/W	是	0	-
D1994	Z 轴原点位置定义 HP (Low word)	○	×	○	-	-	-	R/W	是	0	-
D1995	Z 轴原点位置定义 HP (High word)										
D1996	Z 轴加速时间 T _{ACC}	○	×	○	-	-	-	R/W	是	500	-
D1997	Z 轴减速时间 T _{DEC}	○	×	○	-	-	-	R/W	是	500	-
D1998	Z 轴目标位置(I) P(I) (Low word)	○	○	○	○	-	-	R/W	否	0	-
	Z 轴 PWM 频宽设定(Low word)	○	×	×	○	-	-	R/W	否	0	-
D1999	Z 轴目标位置(I) P(I) (High word)	○	○	○	○	-	-	R/W	否	0	-
	Z 轴 PWM 频宽设定(High word)	○	×	×	○	-	-	R/W	否	0	-
D2000	Z 轴运转速度(I) V(I) (Low word)	○	×	○	1000	-	-	R/W	否	1000	-
D2001	Z 轴运转速度(I) V(I) (High word)										
D2002	Z 轴目标位置(II) P(II) (Low word)	○	○	○	0	-	-	R/W	否	0	-
	Z 轴 PWM 输出周期设定(Low word)	○	×	×	0	-	-	R/W	否	0	-
D2003	Z 轴目标位置(II) P(II) (High word)	○	○	○	0	-	-	R/W	否	0	-
	Z 轴 PWM 输出周期设定(High word)	○	×	×	0	-	-	R/W	否	0	-
D2004	Z 轴运转速度(II)V(II) (Low word)	○	×	○	2000	-	-	R/W	否	2K	-
D2005	Z 轴运转速度(II)V(II) (High word)										
D2006	Z 轴运转命令	○	×	○	0	-	0	R/W	否	0	-
D2007	Z 轴工作模式	×	×	○	0	-	-	R/W	否	0	-
D2008	Z 轴现在位置 CP(PLS) (Low word)	○	×	○	0	-	-	R/W	否	0	-
D2009	Z 轴现在位置 CP(PLS) (High word)										
D2010	Z 轴现在速度(PPS) (Low word)	○	×	○	0	0	0	R/W	否	0	-
D2011	Z 轴现在速度(PPS) (High word)										
D2012	Z 轴现在位置 CP(unit) (Low word)	○	×	○	0	-	-	R/W	否	0	-
D2013	Z 轴现在位置 CP(unit) (High word)										
D2014	Z 轴现在速度 CS (unit) (Low word)	○	×	○	0	0	0	R/W	否	0	-
D2015	Z 轴现在速度 CS (unit) (High word)										
D2016	Z 轴执行状态	○	×	○	0	-	-	R	否	0	-
D2017*	Z 轴错误编号	○	×	○	0	-	-	R	否	0	3-32

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D2018	Z 轴电子齿轮 (分子)	○	×	○	-	-	-	R/W	是	1	-
D2019	Z 轴电子齿轮 (分母)	○	×	○	-	-	-	R/W	是	1	-
D2020	Z 轴手摇轮输入频率 (Low word)	○	×	○	0	-	0	R/W	否	0	-
D2021	Z 轴手摇轮输入频率 (High word)										
D2022	Z 轴累计手摇轮输入脉冲个数 (Low word)	○	×	○	0	-	-	R/W	否	0	-
D2023	Z 轴累计手摇轮输入脉冲个数 (High word)										
D2024	Z 轴手摇轮输入响应速度	○	×	○	-	-	-	R/W	是	5	-
D2026	Z 轴电气零点 (Low word)	○	×	○	-	-	-	R/W	是	0	-
D2027	Z 轴电气零点 (High word)										
D2029	OZ 错误 STEP 位置 (目前没使用, 暂定为保留)	○	×	○	-	-	-	R/W	是	0	-
D2037	Z 轴 PID 闭回路控制	×	×	○	-	-	-	R/W	否	0	
D2038	Z 轴闭回路命令现在值(Low word)	×	×	○	-	-	-	R	否	0	
D2039	Z 轴闭回路命令现在值(High word)	×	×	○	-	-	-	R	否	0	
D2040	Z 轴闭回路 P 值(Low word)	×	×	○	-	-	-	R/W	否	0	
D2041	Z 轴闭回路 P 值(High word)	×	×	○	-	-	-	R/W	否	0	
D2042	Z 轴闭回路 I 值(Low word)	×	×	○	-	-	-	R/W	否	0	
D2043	Z 轴闭回路 I 值(High word)	×	×	○	-	-	-	R/W	否	0	
D2044	Z 轴闭回路差异值(Low word)	×	×	○	-	-	-	R	否	0	
D2045	Z 轴闭回路差异值(High word)	×	×	○	-	-	-	R	否	0	
D2046	Z 轴闭回路累积差异值(Low word)	×	×	○	-	-	-	R	否	0	
D2047	Z 轴闭回路累积差异值(High word)	×	×	○	-	-	-	R	否	0	
D2048	Z 轴闭回路累积最大差异值(Low word)	×	×	○	-	-	-	R	否	0	
D2049	Z 轴闭回路累积最大差异值(High word)	×	×	○	-	-	-	R	否	0	
D2056	A 轴参数设定	○	×	×	-	-	-	R/W	是	0	-
D2058	A 轴马达转一圈所需脉冲数 (Low word)	○	×	×	-	-	-	R/W	是	2000	-
D2059	A 轴马达转一圈所需脉冲数 (High word)										
D2060	A 轴马达转一圈之移动距离 (Low word)	○	×	×	-	-	-		是	1000	-
D2061	A 轴马达转一圈之移动距离 (High word)										
D2062	A 轴最高速度 V_{MAX} (Low word)	○	×	×	-	-	-	R/W	是	500K	-
D2063	A 轴最高速度 V_{MAX} (High word)										
D2064	A 轴启动速度 V_{BIAS} (Low word)	○	×	×	-	-	-	R/W	是	0	-
D2065	A 轴启动速度 V_{BIAS} (High word)										
D2066	A 轴寸动 JOG 速度 V_{JOG} (Low word)	○	×	×	-	-	-	R/W	是	5000	-
D2067	A 轴寸动 JOG 速度 V_{JOG} (High word)										
D2068	A 轴原点复归速度 V_{RT} (Low word)	○	×	×	-	-	-	R/W	是	50K	-
D2069	A 轴原点复归速度 V_{RT} (High word)										

3 DVP-PM 各种装置功能

特 D	功能说明	10M	20D	20M	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D2070	A 轴原点复归减速速度 V _{CR} (Low word)	○	×	×	-	-	-	R/W	是	1000	-
D2071	A 轴原点复归减速速度 V _{CR} (High word)										
D2072	A 轴零点信号数 N	○	×	×	-	-	-	R/W	是	0	-
D2073	A 轴补充距离 P	○	×		-	-	-	R/W	是	0	-
D2074	A 轴原点位置定义 HP (Low word)	○	×	×	-	-	-	R/W	是	0	-
D2075	A 轴原点位置定义 HP (High word)										
D2076	A 轴加速时间 T _{ACC}	○	×	×	-	-	-	R/W	是	500	-
D2077	A 轴减速时间 T _{DEC}	○	×	×	-	-	-	R/W	是	500	-
D2078	A 轴目标位置(I) P(I) (Low word)	○	×	×	0	-	-	R/W	否	0	-
	A 轴 PWM 频宽设定(Low word)										
D2079	A 轴目标位置(I) P(I) (High word)	○	×	×							
	A 轴 PWM 频宽设定(High word)										
D2080	A 轴运转速度(I) V(I) (Low word)	○	×	×	1000	-	-	R/W	否	1000	-
D2081	A 轴运转速度(I) V(I) (High word)										
D2082	A 轴目标位置(II) P(II) (Low word)	○	×	×	0	-	-	R/W	否	0	-
	A 轴 PWM 输出周期设定(Low word)										
D2083	A 轴目标位置(II) P(II) (High word)	○	×	×							
	A 轴 PWM 输出周期设定(High word)										
D2084	A 轴运转速度(II)V(II) (Low word)	○	×	×	2000	-	-	R/W	否	2K	-
D2085	A 轴运转速度(II)V(II) (High word)										
D2086	A 轴运转命令	○	×	×	0	-	0	R/W	否	0	-
D2088	A 轴现在位置 CP(PLS) (Low word)	○	×	×	0	-	-	R/W	否	0	-
D2089	A 轴现在位置 CP(PLS) (High word)										
D2090	A 轴现在速度(PPS) (Low word)	○	×	×	0	0	0	R/W	否	0	-
D2091	A 轴现在速度(PPS) (High word)										
D2092	A 轴现在位置 CP(unit) (Low word)	○	×	×	0	-	-	R/W	否	0	-
D2093	A 轴现在位置 CP(unit) (High word)										
D2094	A 轴现在速度 CS (unit) (Low word)	○	×	×	0	0	0	R/W	否	0	-
D2095	A 轴现在速度 CS (unit) (High word)										
D2096	A 轴执行状态	○	×	×	0	-	-	R	否	0	-
D2097*	A 轴错误编号	○	×	×	0	-	-	R	否	0	3-32
D2098	A 轴电子齿轮 (分子)	○	×	×	-	-	-	R/W	是	1	-
D2099	A 轴电子齿轮 (分母)	○	×	×	-	-	-	R/W	是	1	-

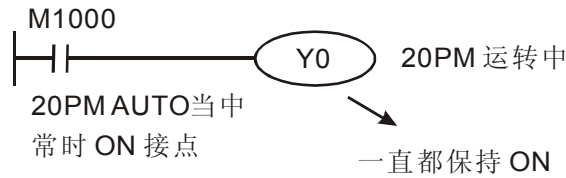
补充说明：20D=DVP20PM00D, 20M=DVP20PM00M

3.11 特殊继电器及特殊寄存器群组功能说明

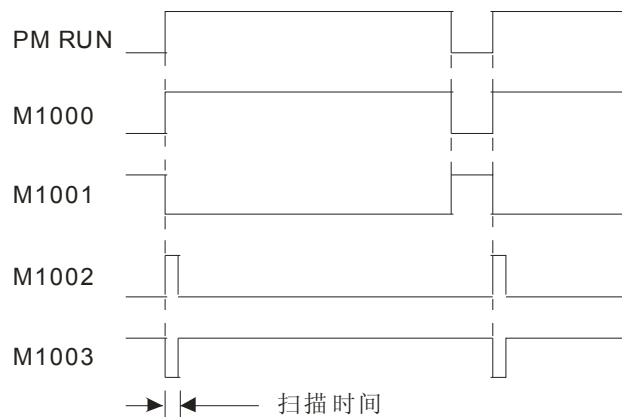
20PM的 运转标志

M1000~M1003

1. M1000: M1000 为 AUTO 中时, 常 On 接点, 即运转监视常开接点 (A 接点), DVP-PM 于 AUTO 的状态下, M1000 保持为 On。



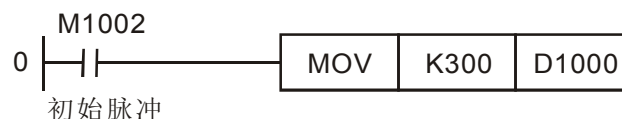
2. M1001: M1001 为 AUTO 中时, 常 Off 接点, 即运转监视常闭接点 (B 接点), DVP-PM 于 AUTO 的状态下 M1001 保持为 Off。
3. M1002: DVP-PM 开始 AUTO 的第一次扫描 On, 之后保持为 Off。该脉冲的宽度为一次扫描时间, 当要作各种初始设定工作时使用本接点。
4. M1003: DVP-PM 开始 AUTO 的第一次扫描 Off, 之后一直 On。即起始负向 (AUTO 的瞬间'Off') 脉冲。



监控 定时器

D1000

1. 监控定时器专门用来监视 DVP-PM 的扫描时间, 当扫描时间超过监控定时器的设定时间时, ERROR 红色指示灯长亮, 输出全部变成 Off。
2. 监控定时器时间的初始值为 200ms, 当程序长或是运算过于复杂时, 可于程序中使用 MOV 指令来变更监控定时器的设定值, 如下所示, 将监控定时器的设定值变更为 300ms。



3. 监控定时器最大可设定至 32,767ms, 但必须注意, 监控定时器设定过大时, 运算异常发生的检出时机将会跟着被拖慢。因此, 若非复杂的运算使得扫描时间超过 200ms, 一般的情况下请维持在 200ms 以下较佳。
4. 指令运算过于复杂或者是 DVP-PM 主机连接众多的特殊模块时都会造成扫描时间过长, 扫描时间是否超过 D1000 的设定值, 请监视 D1010~D1012。此种情况下, 可变更 D1000 的设定值。

3 DVP-PM 各种装置功能

通讯口 功能

M1120、M1136、
M1138、M1139、
M1143、D1036、
D1109、D1120

DVP-PM 内建通讯口 (COM1: RS-232、COM2: RS-485) 支持 MODBUS ASCII/RTU 通讯格式, 速率最高可达 115,200 bps。(COM3 通讯卡: RS-232/RS-485) 支持 MODBUS ASCII 通讯格式, 速率最高可达 38,400 bps。COM1, COM2, COM3 可同时使用。

COM1 仅作为从站 (Slave) 使用, 支持 ASCII/RTU 通讯格式, 支持可调整通讯速率, 速率最高可达 115,200 bps, 及修改数据位长度 (Data bits, Parity bits, Stop bits)。

COM2 可作主站或从站, 均支持 ASCII/RTU 通讯格式, 支持可调整通讯速率, 速率最高可达 115,200 bps, 及修改数据位长度 (Data bits, Parity bits, Stop bits)。

COM3 仅作为从站(Slave)使用, 支持 ASCII 通讯格式 7,E,1(Data bits, Parity bits, Stop bits), 支持可调整通讯速率, 速率最高可达 38400 bps, COM2 或 COM3 不可同时当从站(Slave)使用。

◎ 通讯格式设定:

- COM1
1. 通讯格式由 D1036 设定, (b8~b15) 并未支持。
 2. M1138 通讯设定保持。
 3. ASCII/RTU 模式设定 M1139
- COM2
1. 通讯格式由 D1120 设定。
 2. M1120 通讯设定保持。
 3. ASCII/RTU 模式设定 M1143。
- COM3
1. 通讯格式由 D1109 设定, (b0~b3、b8~b15)并未支持。
 2. M1136 通讯设定保持。

其设定方法请参考下表:

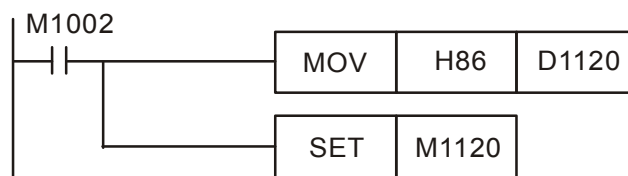
	内 容	0	1
b0	数据长度	b0=0: 7	b0=1: 8
b1 b2	同位	b2, b1=00 : 无 (None) b2, b1=01 : 奇同位 (Odd) b2, b1=11 : 偶同位 (Even)	
b3	stop bits	b3=0: 1 bit	b3=1: 2 bit
b7~b4	b7~b4=0001 (H1) :	110	bps
	b7~b4=0010 (H2) :	150	bps
	b7~b4=0011 (H3) :	300	bps
	b7~b4=0100 (H4) :	600	bps
	b7~b4=0101 (H5) :	1,200	bps
	b7~b4=0110 (H6) :	2,400	bps
	b7~b4=0111 (H7) :	4,800	bps
	b7~b4=1000 (H8) :	9,600	bps
	b7~b4=1001 (H9) :	19,200	bps

	内 容	0	1
b7~b4	b7~b4=1010 (HA) :	38,400	bps
	b7~b4=1011 (HB) :	57,600	bps
	b7~b4=1100 (HC) :	115,200	bps
b8	起始字符选择	b8=0: 无	b8=1: D1124
b9	第一结束字符选择	b9=0: 无	b9=1: D1125
b10	第二结束字符选择	b10=0: 无	b10=1: D1126
b15~b11	无定义		

范例一：COM2 通讯格式修改方式

若要修改 COM2 的通讯格式，请在程序最上端加入下面程序代码，当 DVP-PM 由 MANU 到 AUTO 时，在 DVP-PM 的第一次扫描时间时，会侦测 M1120 是否有 On，若有则会依据 D1120 的设定值去更改 COM2 的相关设定。

将 COM2 的通讯格式改成 ASCII 模式、9,600bps、7 Data bits、Even parity、1 Stop bits (9600, 7, E 1)



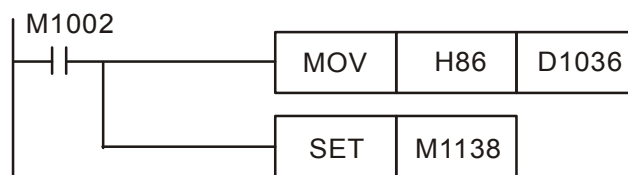
注意事项：

1. 当 COM2 要当成 SLAVE 端口使用时，请勿在程序中有任何通讯指令存在。
2. 当通讯格式修改完成后，将 DVP-PM 由 AUTO→MANU，通讯格式不会变化。
3. 当通讯格式修改完成后，DVP-PM 电源关闭之后再上电就会回复到出厂设定的通讯格式。

范例二：COM1 通讯格式的修改方式

若要修改 COM1 的通讯格式，请在程序最上端加入下面程序代码，当 DVP-PM 由 MANU 到 AUTO 时，在 DVP-PM 的第一次扫描时间时，会侦测 M1138 是否有 On，若有则会依据 D1036 的设定值去更改 COM1 的相关设定。

将 COM1 的通讯格式改成 ASCII 模式、9,600bps、7 Data bits、Even parity、1 Stop bits (9600, 7, E, 1)



注意事项：

1. 当通讯格式修改完成后，将 DVP-PM 由 AUTO→MANU，通讯格式不会变化。

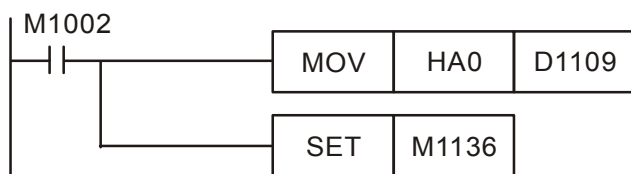
3 DVP-PM 各种装置功能

2. 当通讯格式修改完成后, DVP-PM 电源关闭之后再上电就会回复到出厂设定的通讯格式。

范例三: COM3 通讯格式的修改方式:

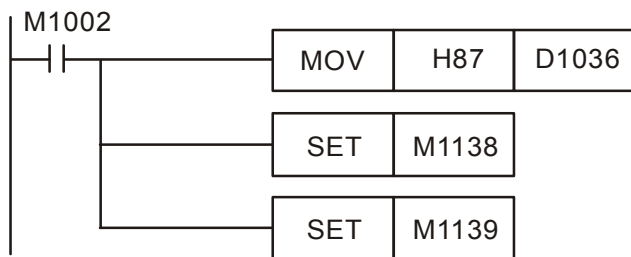
COM3 的通讯格式固定为 7 Data bits、Even parity、1 Stop bits, 若要修改 COM3 通讯速率为 38,400 bps 请在程序最上端加入下面程序代码, 当 DVP-PM 由 MANU 到 AUTO 时, 在 DVP-PM 的第一次扫描时间时, 会侦测 M1136 是否有 On, 若有则会依据 D1109 的设定值去更改 COM3 的相关设定。

将 COM3 的通讯速率改成 38,400 bps

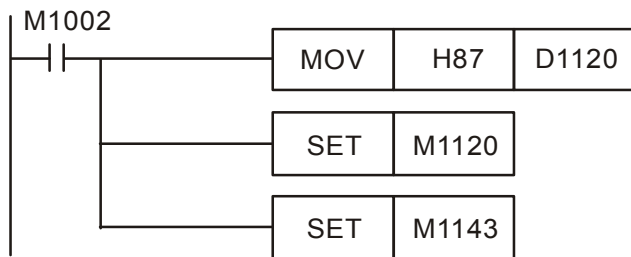


范例四: COM1、COM2 之 RTU 模式设定方式如下:

COM1: (9,600, 8, E, 1, RTU)



COM2: (9,600, 8, E, 1, RTU)



通讯回应
延迟

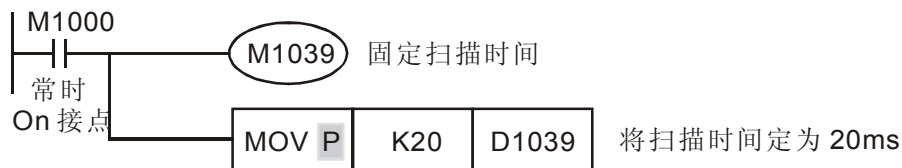
D1038

在 RS-485 界面通讯时, DVP-PM 主机当从站 (Slave) 时, 数据响应延迟时间可设定, 设定范围 0~3,000 (0~30 秒, 时间单位为 10ms), 若不在此范围, 则 D1038 = 0。设定值必须小于 D1000 的设定值。

固定
扫描时间

M1039、D1039

1. 将 M1039 为 On, 则程序的扫描时间固定以 D1039 的内容来决定, 当程序执行完毕, 必须等待固定的扫描时间到达时才执行下一次的扫描, 如果 D1039 的内容小于实际上程序的扫描时间时, 则以实际上程序的扫描时间为主。



2. D1010~D1012 所显示的扫描时间亦包括固定的扫描时间。

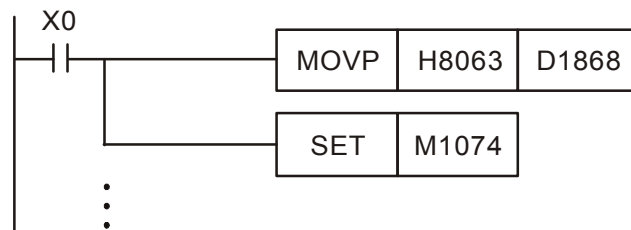
设定OX
程式编号

由 D1868 可指定欲执行的 OX 程序编号，其设定步骤如下：

1. 首先设定 D1868 的 b[14]=1 或 b[15]=1 或 b[14]=b[15]=1。(前述三种条件只要成立其中一项即可)，同时将 D1868 的 b[0~13] 写入 K99 (=H63)，即为设定 OX 编号数值为 OX99。结合前述两条件，因此将 H8063 写至 D1868 寄存器之中。
2. 上述步骤 1 设定完成后，SET M1074，即可启动 D1868 所指定的 OX 编号程序。

M1074、D1868

上述步骤 1~2 之范例程序如下：



在 O100 主程序中，设计由 X0 启动子程序 OX99，并且执行 OX99 内之程序。

高速计数
功能

10PM 有六组计数功能，其设定数据如下表所示：

M1200、C200
M1204、C204
M1208、C208
M1212、C212
M1216、C216
M1220、C220

组别	计数器 编号	计数模式		清除计数 外部接脚	外部输入接脚
		装置	设定值		
第一组	C200	K1M1200	0: U/D* 1: P/D* 2: A/B*(1 倍频) 3: 4A/B(4 倍频)	X10- / M1203	X0、X1、S/S
第二组	C204	K1M1204		X11- / M1207	X2、X3、S/S
第三组	C208	K1M1208		X12- / M1211	X4、X5、S/S
第四组	C212	K1M1212		X13- / M1215	X6、X7、S/S
第五组	C216	K1M1216		X0 / M1219	X10+、X10-、X11+、X11-
第六组	C220	K1M1220		X1 / M1223	X12+、X12-、X13+、X13-

注 1: U/D: 上数/下数, P/D: 脉冲/方向, A/B: A 相/B 相

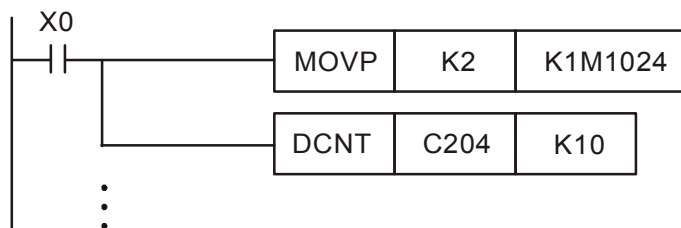
注 2: 1~4 组计数器输入为开集极线路, 5、6 组输入为差动线路。

3 DVP-PM 各种装置功能

指定使用第二组高速计数器，其设定步骤如下：

1. 首先设定第二组计数器模式为 A/B 相(一倍频)计数模式(K1M1204=K2)。
2. 启动 C204 开始计数。

上述步骤 1~2 之范例程序如下：

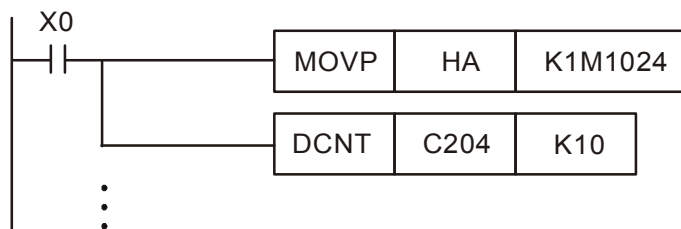


3. 若设定由外部讯号清除目前计数值，则模式设定为(K1M1024=H'A)。

M1027	M1026	M1025	M1024
1	0	1	0

4. 启动 C204 开始计数，当接点 X11-为 ON，C204 立即被清除为零。

上述步骤 3~4 之范例程序如下：



高速计时功能

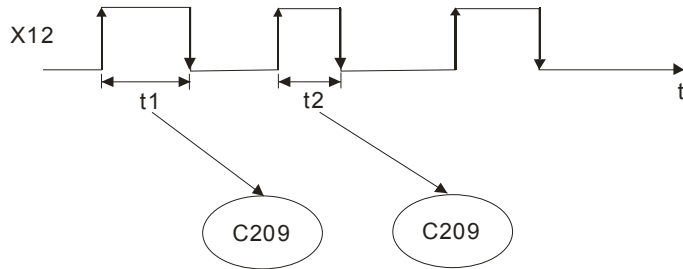
10PM 有六组计时功能，其设定数据如下表所示：

M1200、C201
M1204、C205
M1208、C209
M1212、C213
M1216、C217
M1220、C221

组别	计时装置	计时模式				启动计时外部讯号	计时储存装置	
		装置	设定值					
第一组	C200	K1M1200					X10	C201
第二组	C204	K1M1204	Bit3	Bit2	Bit1	Bit0	X11	C205
第三组	C208	K1M1208	x	启动计时	x	模式设定	X12	C209
第四组	C212	K1M1212	Bit2: 启动计时功能 Bit0: 等于 0 为一般模式，计时区间为上缘至下缘 等于 1 为周期模式，计时区间为上缘至下一个上缘				X13	C213
第五组	C216	K1M1216					X0	C217
第六组	C220	K1M1220					X1	C221

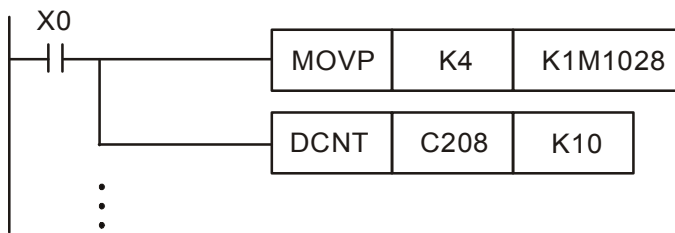
指定使用第三组定时器，其设定步骤如下：

1. 首先设定第三组定时器计时模式为一般模式(Bit0=0)、启动计时功能(Bit2=1)，则装置 K1M1208=K4。
2. 启动 C208 开始计数，计算 X12 外部触发讯号上缘至讯号的下缘之间隔时间，将此值写入 C209，写入数值单位为 0.01us，如下图所示。

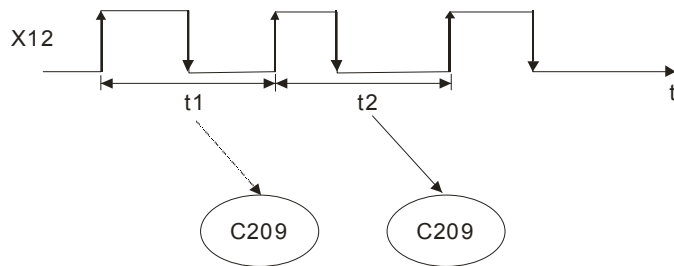


一般模式，单位 0.01μs

上述步骤 1~2 之范例程序如下：

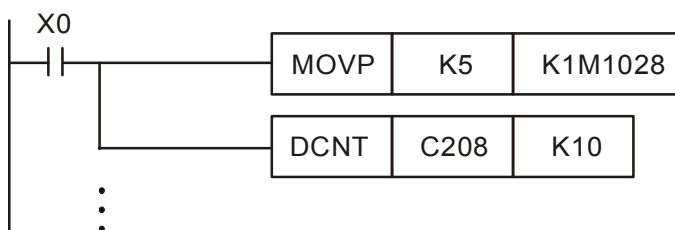


3. 若设定计时方式为周期模式(Bit0=1)，则装置 K1M1208=K5。
4. 启动 C208 开始计数，计算 X12 外部触发讯号上缘至下个上缘之间隔时间，将此值写入 C209，写入数值单位为 0.01us，如下图所示。



周期模式，单位 0.01μs

上述步骤 3~4 之范例程序如下：



3 DVP- PM 各种装置功能

扩展连接检测

1. D1140: 右侧模块 (AD、DA、XA、PT、TC、RT、HC、PU) 台数, 最多 8 台。
2. D1142: 数字模块输入 X 点数。
3. D1143: 数字模块输出 Y 点数。

D1140、D1142、
D1143

停电保持范围设定

1. DVP-PM 主机停电保持范围设定, 依照开始地址编号到结束地址编号之范围为停电保持区域。
2. 请参考第 3.1 节之对照表。

D1200~D1211

X输入点可强制On/Off

当 M1304=On 时, 主机上之 X 输入点 (X0~X17) 可利用周边装置如 PMSoft 做强制 On-Off 之动作, 但硬件灯号不会反应。

M1304

右侧特殊扩展模块代码

1. 若有连接特殊 I/O 模块将会依排列顺序将 I/O 模块机种代号显示在 D1320 ~ D1327 内。
2. DVP-PM 机种特殊 I/O 模块代号:

D1320~D1327

I/O 模块名称	I/O 模块代号 (HEX)	I/O 模块名称	I/O 模块代号 (HEX)
DVP04AD-H2	H'6400	DVP01PU-H2	H'6110
DVP04DA-H2	H'6401	DVP04PT-H2	H'6402
DVP04TC-H2	H'6403	DVP06XA-H2	H'6604
DVP-PM	H'6260	DVP01HC-H2	H'6120

中断寄存器

1. D1400 为中断致能寄存器, 欲启动以下 9 个中断请将对应之 BIT#ON, 说明如下:

D1400、D1401

bit#	中断	中断编号
0	时间中断	I0
1	外部端子 Start0 / X0*	I1
2	外部端子 Stop0 / X1*	I2
3	外部端子 Start1 / X2*	I3
4	外部端子 Stop1 / X3*	I4
5	外部端子 X4*	I5
6	外部端子 X5*	I6
7	外部端子 X6*	I7
8	外部端子 X7*	I8

备注*:仅适用于 DVP-10PM 机种

2. 当启动之中断种类为时间中断，则可在 D1401 输入时间中断之周期。
3. 中断的种类分别为时间中断与外部中断，说明如下：
 - 外部中断：当外部端子的输入讯号于上缘或下缘触发时，不受扫描周期影响，立即中断目前执行中程序而立即跳至指定的中断子程序处执行，当中断子程序执行完毕后即返回中断前执行的程序。
 - 时间中断：PM 每隔时间中断之周期(D1401)便自动中断目前执行之程序而跳至时间中断子程序执行。

清除执行中
M-Code

- ◆ 欲清除 OX M Code 命令时，可执行 M1744=1，当 M1744 被执行时，会同时清除 D1703 与 M1794。

M1744、M1794、
D1703

- ◆ 上述中的 M1794 为 OX M Code 的被执行旗标，而 D1703 为 OX 的 M Code 执行寄存器。

Ready标志

1. X,Y,Z 三轴各有 Ready 旗标，X 轴: M1792, Y 轴: M1872, Z 轴: M2032。可用来判断行程是不是还在运转。
2. 以下为 X 轴 Ready 旗标动作说明: X 轴在运转前 M1792 为 On, 开始运转后 M1792 变为 Off, 运转完后则变为 On。

M1792、M1872、
M2032

现在位置
写入允许

1. 欲写入 X、Y、Z 轴之现在位置 CP(PLS)时，可执行 M1751=1，则现在位置 D1848 的属性变更为可擦写。

M1751、M1831、
M1991

2. 若在 M1751=0 的状态下写入 D1848，则 D1848 的数值清除为 0。

运动错误
清除

1. 当 X 轴、Y 轴或 Z 轴发生运动错误时，错误旗标分别为 M1793、M1873、M2033，且会将错误讯息分别存放至 D1857、D1937、D2017 寄存器之中。

M1793、D1857
M1873、D1937
M2033、D2017

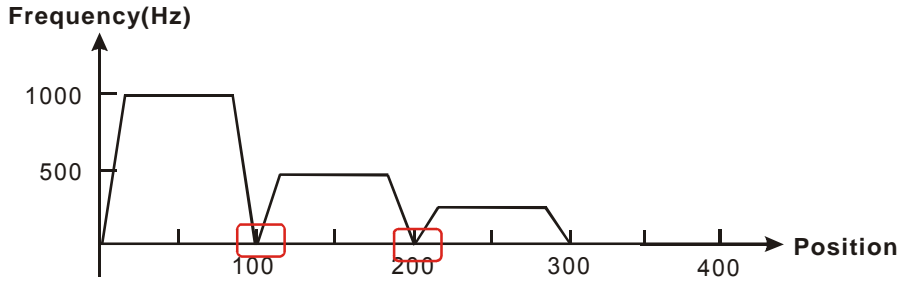
2. 欲进行除错时，请清除错误讯息寄存器与错误旗标。

3 DVP- PM 各种装置功能

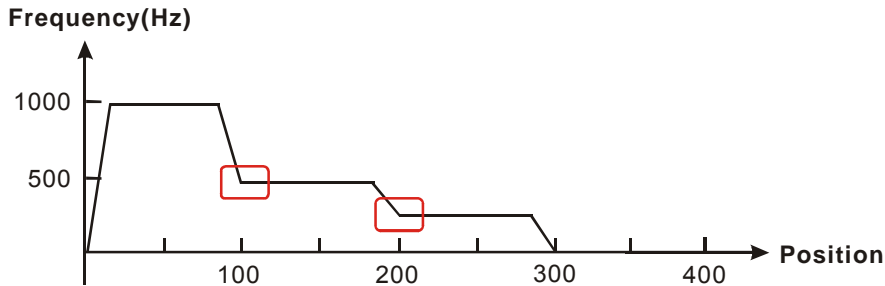
连续插补

D1796、M1036

1. 当使用 D1796 设定最大减速速度(kHz)时, 路径减速下降时会以 D1796 设定值与实际减速速度值, 二者取小的频率值做为转折点。
2. 当 D1796 > 0 时即产生连续路径效果, 但只有对 G1/ G2/ G3/ LIN/ CW/ CCW 指令 有效; 可设定 M1036 为 On (M1036 = k1), 使其它的应用指令也具有连续路径效果。
3. 设定连续插补 D1796 = k0 (关闭连续插补功能) : 设定为 k0, 则不管减速速度多少, 都会降至 0kHz 再加速, 如下图所示。



4. 设定连续插补 D1796 = k500 (开启连续插补功能) : 减速速度会与 D1796 比较, 以较小值为低点再加速, 如下图所示。

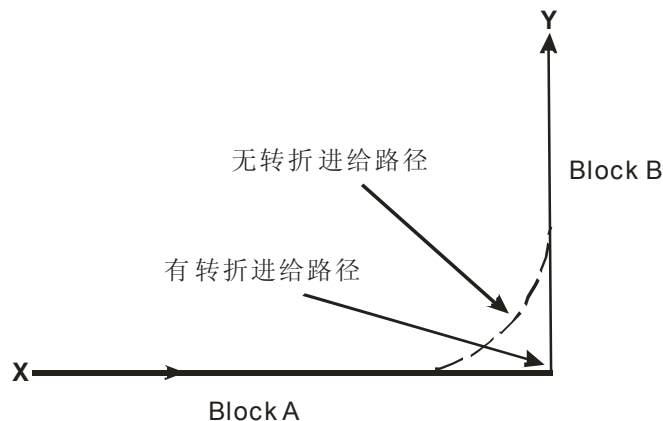


5. 连续路径示意图

在未设定 D1796 连续插补时, X → Y 路径为直角转折后向上; 在设定连续插补 D1796 后, X → Y 路径为较平滑的曲线向上。

G01 X100 F1000; (Block A)

Y100; (Block B)



Block A: 执行 X 轴移动路径; Block B: 执行 Y 轴移动路径

设定G-Code
移动速度
百分比

D1798

1. 当 D1798 设为 100 时, G-Code 移动速度以原设定速度移动; 当 D1798 设为 1000 时, G-Code 移动速度以原设定速度 10 倍移动; 当 D1798 设为 50 时, G-Code 移动速度以原设定速度一半移动; 以此类推。
2. D1798 设定百分比使 G-Code 移动速度超过 500k(HZ) 时, 则会以 500k(HZ) 速度移动。

设定输入
端子极性

D1799

欲设定下表内之输入端子的极性为 a 接点时, 请将 bit# On。若为 b 接点时, 则将 bit# Off 即可。

20PM:

bit#	X 轴输入端子极性	bit#	Y 轴输入端子极性
0	PG0	8	PG1
1	MPGB0	9	MPGB1
2	MPGA0	10	MPGA1
3	LSN0	11	LSN1
4	LSP0	12	LSP1
5	DOG0	13	DOG1
6	STOP0	14	STOP1
7	START0	15	START1

10PM:

bit#	输入端子极性
0	DOG0
1	PG0
2	DOG1
3	PG1
4	DOG2
5	PG2
6	DOG3
7	PG3

3 DVP-PM 各种装置功能

读取输入端子状态

当 bit# 为 On 时，表示有讯号输入。反之，则当 bit# 为 Off 时，表示无讯号输入。

20PM:

D1800

bit#	X 轴输入端子状态	bit#	Y 轴输入端子状态
0	PG0	8	PG0
1	MPGB0	9	MPGB1
2	MPGA0	10	MPGA1
3	LSN	11	LSN
4	LSP	12	LSP
5	DOG	13	DOG
6	STOP	14	STOP
7	START	15	START

10PM:

bit#	输入端子状态
0	DOG0
1	PG0
2	DOG1
3	PG1
4	DOG2
5	PG2
6	DOG3
7	PG3

设定输入端子极性 2

欲设定下表内之输入端子的极性为 a 接点时，请将 bit# On。若为 b 接点时，则将 bit# Off 即可。

D1804

20PM:

bit#	Z 轴输入端子极性
0	PG2
1	MPGB1
2	MPGA1
3	LSN2
4	LSP2
5	DOG2

10PM:

bit#	输入端子极性
0	X3(DOG3)
1	MPGB1
2	MPGA1
3	X2(DOG2)
4	X1(DOG1)
5	X0(DOG0)

读取输入端子状态 2

当 bit# 为 On 时，表示有讯号输入。反之，则当 bit# 为 Off 时，表示无讯号输入。

D1805

20PM:

bit#	Z 轴输入端子状态
0	PG2
1	MPGB1
2	MPGA1
3	LSN2
4	LSP2
5	DOG2

10PM:

bit#	输入端子状态
0	X3
1	MPGB1
2	MPGA1
3	X2
4	X1
5	X0

O100 错误发生检查

1. 当 O100 程序发生错误时，O100 错误旗标 M1953 会 SET On，错误讯息会被存在 D1802，错误 STEP 位置会存放在 D1803。

M1953、D1802、2. 错误讯息请参阅第 14 章附录 A 的错误讯息表。

D1803

X-Y-Z-A 轴参数设定

D1816 为 X 轴参数设定，D1896 为 Y 轴参数设定，D1976 为 Z 轴参数设定，D2056 为 A 轴参数设定。说明如下：

D1816、D1896、
D1976、D2056

bit#	X-Y-Z 轴参数设定	bit#	X-Y-Z 轴参数设定
0	单位系设定 (注一)	8	原点复归方向 (注四)
1		9	原点复归模式 (注四)
2	位置数据倍率设定 (注二)	10	原点复归 DOG 下降缘侦测 (注四)
3		11	脉冲旋转方向 (注四)
4	脉冲型式 (注三)	12	相对绝对坐标 (注四)
5		13	DOG 触发启动模式 (注四)
6*	PWM 模式 (注四)	14	曲线选择 (注四)
7		15	

备注*:仅 DVP-10PM 机种支持此功能

注一:

b1	b0	单位系
0	0	马达单位
0	1	机械单位
1	0	复合单位
1	1	

注二:

	马达单位	复合单位	机械单位
位置	pulse	um	
	pulse	m deg	
	pulse	10 ⁻⁴ inch	
速度	pulse/sec		cm/min
	pulse/sec		10deg/min
	pulse/sec		inch/min

注三:

3 DVP-PM 各种装置功能

b3	b2	位置数据倍率设定	b5	b4	说明
0	0	10^0	0	0	正向脉冲+反向脉冲
0	1	10^1	0	1	脉冲+方向
1	0	10^2	1	0	A/B 相脉冲 (2 相 2)
1	1	10^3	1	1	

注四:

bit#	说明
6	<p>b[6]=1: 启用 PWM 模式</p> <p>(1) 运转命令中启动 JOG+运转, 则 Y0~Y3 输出 PWM</p> <p>(2) 运转命令中启动单段速定位运动, 则 Y0~Y3 输出单向脉冲定位</p> <p>(3) 脉冲宽度设定: D1838、D1918、D1998、D2078</p> <p>(4) 脉冲频率设定: D1842、D1922、D2002、D2082</p>
8	<p>b[8]=0: 执行原点复归方向, 往现在位置 (CP) 内容值递减方向执行</p> <p>b[8]=1: 执行原点复归方向, 往现在位置 (CP) 内容值递增方向执行</p>
9	b[9]=0: 正常模式; b[9]=1: 覆写模式
10	<p>b[10]=0: 原点复归 DOG 下降缘侦测</p> <p>b[10]=1: 原点复归 DOG 上升缘侦测</p>
11	<p>b[11]=0: 当正转运动时, 现在位置 (CP) 内容值递增</p> <p>b[11]=1: 当正转运动时, 现在位置 (CP) 内容值递减</p>
12	b[12]=0: 设定为绝对坐标定位; b[12]=1: 设定为相对坐标定位
13	<p>b[13]=0: DOG 信号上升缘触发</p> <p>b[13]=1: DOG 信号下降缘触发</p> <p>(插入单段速定位运动模式、插入二段速定位运动模式有效)</p>
14	<p>b[14]=0: 为加速采梯形加速曲线</p> <p>b[14]=1: 为加速采 S 加速曲线</p>

电子凸轮
CAM 设定

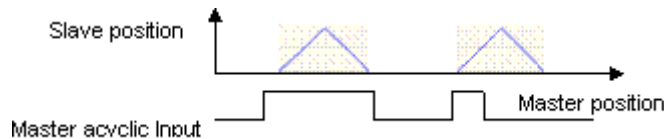
此功能仅 DVP-20PM 机种支持。

1. D1832 重复次数(Times of E-CAM repetitions) 寄存器:

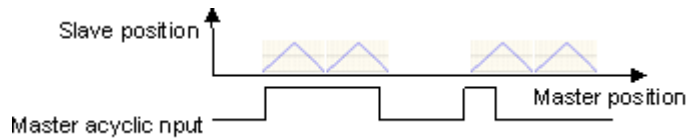
D1832、D1833、
D1834

此功能由 D1832 寄存器控制重复执行 CAM Data 周期次数，当 D1832 的值超过 H8000(Bit15=1) 时，将变为周期式电子凸轮。如下图范例说明，当 D1832=0 时，不重复执行 CAM Data，若 D1832=1 时，将重复执行一次 CAM Data，以此类推。

D1832 = 0 (Slave position: 被动轴位置, Mater position: 主动轴位置, Master acyclic Input: 凸轮启动讯号)



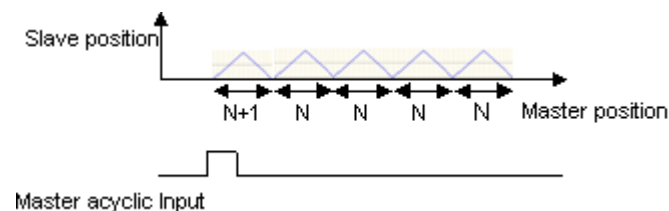
D1832 = 1



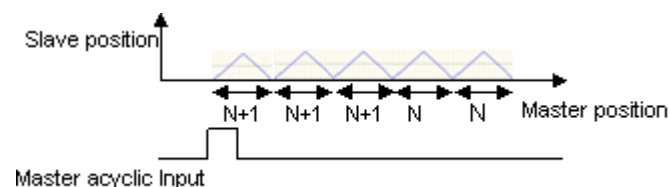
2. D1833 脉冲余数(Remaining E-CAM pulses)分配寄存器:

当主轴运转固定为 202 个脉冲数，而执行一个周期 CAM Data 需要 50 个脉冲数时，执行结果为 4 个 CAM Data 并剩下两个脉冲，因此 D1832 设定 3 (重复执行三次)，D1833 设定 2，执行第一个和第二个 CAM Data 周期凸轮，要执行下一个 CAM Data 周期凸轮之前，会多走一个脉冲，以此类推把脉冲无法整除余数平均。如下图范例(A)的说明，当 D1832=4 及 D1833=1 时，会执行 5 次 CAM Data，只有执行完第一组 CAM Data，主轴(Master)会再走一个脉冲(Pulse)，才会再执行下一个 CAM Data，以此类推，可知下图范例(B)中，D1833=3 时，在第一、第二和第三的 CAM Data 执行完，执行下一组 CAM Data 之前，主轴(Master)会再走一个脉冲(Pulse)。

D1832 = 4、D1833=1



D1832 = 4、D1833=3



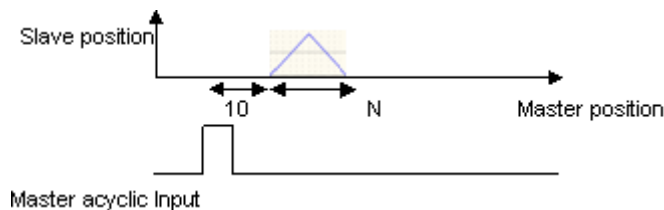
3 DVP- PM 各种装置功能

※ D1833 的数据不可大于 D1832。

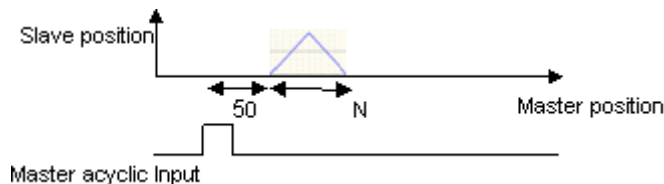
3. D1834 启动延迟(Pulses for delay start)寄存器:

执行非周期电子凸轮，在接受一个凸轮启动讯号(Master Acyclic Input) 时，若要延迟主轴(Master) 转动数个脉冲后再运作 CAM Data，可使用 D1834 寄存器设定延迟脉冲数。如下图标，当系统接受一个凸轮启动讯号(Master acyclic)，主轴(Master) 在转动 D1834 设定的脉冲数后，再接着立即执行 CAM Data。

D1834 = 10



D1834 = 50



X-Y-Z-A 轴
运转命令

D1846 为 X 轴运转命令设定与电子凸轮启动设定，D1926 为 Y 轴运转命令设定，D2006 为 Z 轴运转命令设定，D2086 为 A 轴运转命令设定。说明如下：

D1846 、
D1926 、
D2006 、 D2086

bit#	X-Y-Z-A 轴运转命令	bit#	X-Y-Z-A 轴运转命令
0	软件 STOP	8	单段速定位运动模式启动
1	软件 START	9	单段速插入定位运动模式启动
2	JOG+ 运转	10	两段速定位运动模式启动
3	JOG- 运转	11	两段速插入定位运动模式启动
4	变速度运转模式启动	12	OX 设定 0: Stop; 1: Start
5	手摇轮输入操作	13*2	周期性电子凸轮启动
6	原点复归模式启动	14*2	非周期性电子凸轮启动
7*1	单段速外部触发	15	—

00D 手摇轮(MPG)输入: X 轴(A0+/A0-, B0+/B0-), Y 轴 (A1+/A1-, B1+/B1-)

00M 手摇轮(MPG)输入: X 轴(A0+/A0-, B0+/B0-), Y 轴与 Z 轴(A1+/A1-, B1+/B1-)

*注 1: 仅 DVP-20PM 机种之 Y 轴(D1926)支持此功能

*注 2: 仅 DVP-20PM 机种之 X 轴(D1846)支持此功能

X-Y-Z 轴
工作模式

D1847 为 X 轴工作模式设定，D1927 为 Y 轴工作模式设定，D2007 为 Z 轴工作模式设定。
说明如下：

D1847 、
D1927 、
D2007 、

bit#	X-Y-Z 轴工作模式	bit#	X-Y-Z 轴工作模式
0	—	8	MASK 选择设定
1	—	9	
2	CLR 信号输出模式	10	
3	CLR 输出 On/Off 控制	11	01: 多轴中断插入电子凸轮数据设定
4	CLR 极性设定	12	10: 电子凸轮建表式脉冲输出设定 11: 多轴凸轮正反转双向运动模式
5	STOP 模式设定	13	—
6	手摇轮 MPG 范围限制	14	—
7	LSP/LSN 停止模式	15	回归出厂设定

bit#	说明
2	当 b[2]=0 时，输出信号 CLR 为原点复归完成时输出 130ms 给 Servo 的信号，作为伺服内部偏差计数器的清除信号 当 b[2]=1 时，CLR 输出点作为一般输出点，状态由 b[3]控制 On/Off
3	当 b[3]=0 时，输出点 CLR 为 Off，当 b[3]=1 时，输出点 CLR 为 On
4	当 b[4]=0 时，CLR 极性为 a 接点。 当 b[4]=1 时，CLR 极性为 b 接点。
5	b[5]=0: 马达运转中，碰到 STOP 信号输入时，马达减速停，再度下达运动命令时，马达会忽略之前未完成的距离，立即继续下一步骤的距离。 b[5]=1: 马达运转中，碰到 STOP 信号输入时，马达减速停，再度下达运动命令时，马达会先移动之前未完成的距离，再执行下一定位行程。
6	b[6]=0: 手摇轮 (MPG) 输入之脉冲输出的允许范围无限制。 b[6]=1: 手摇轮 (MPG) 输入之脉冲输出的允许范围，被限制于 P(I) 与 P(II) 的范围内操作，当超出范围频率波减速停止输出。
7	b[7]=0: 马达运转中，碰到 LSP/LSN 信号输入时，马达减速停止。 b[7]=1: 马达运转中，碰到 LSP/LSN 信号输入时，马达立即停止。
8	停止讯号设定 (1 段速运转、2 段速运转、1 段速插入运转、2 段速插入运转) 使用手摇轮时，致能 ΦA_{\pm} 或 ΦB_{\pm} 之触发信号为停止讯号。
9	b[10~8]=K0 (000) 或其它数值: 无立即停止功能
10	b[10~8]=K1 (001) : 输入端 ΦA_{\pm} 讯号上升缘触发则立即停止
	b[10~8]=K2 (010) : 输入端 ΦA_{\pm} 讯号下降缘触发则立即停止
	b[10~8]=K3 (011) : 输入端 ΦB_{\pm} 讯号上升缘触发则立即停止 b[10~8]=K4 (100) : 输入端 ΦB_{\pm} 讯号下降缘触发则立即停止
11	b[12~11]=K1 (01) : 多轴中断插入电子凸轮数据
12	b[12~11]=K3 (11) : 多轴凸轮正反转双向运动模式

3 DVP- PM 各种装置功能

**X-Y-Z-A轴
执行状态**

D1856 为 X 轴执行状态，D1936 为 Y 轴执行状态，D2016 为 Z 轴执行状态，D2096 为 A 轴执行状态。说明如下：

D1856、D1936、
D2016、D2096

bit#	X-Y-Z-A 轴执行状态
0	正向脉冲输出中
1	反向脉冲输出中
2	行程动作中
3	错误产生
4	行程暂停中
5	正向 MPG 输入
6	反向 MPG 输入
7	—

**X-Y轴
外部启动模式**

D1875、D1955

1. 当 D1875、D1955 之 High Byte=H01 表示致能外部输入模式，H00 禁能外部输入模式。
2. 当 D1875、D1955 之 Low Byte 表示指定外部输入 X 起始编号，连续四个，为 X0 ~ X3，分别启动 JOG+、JOG-、MPG、ZRN。
3. 举例：D1875、D1955=H0110，表示 X10~X13 分别可启动 JOG+、JOG-、MPG、ZRN。

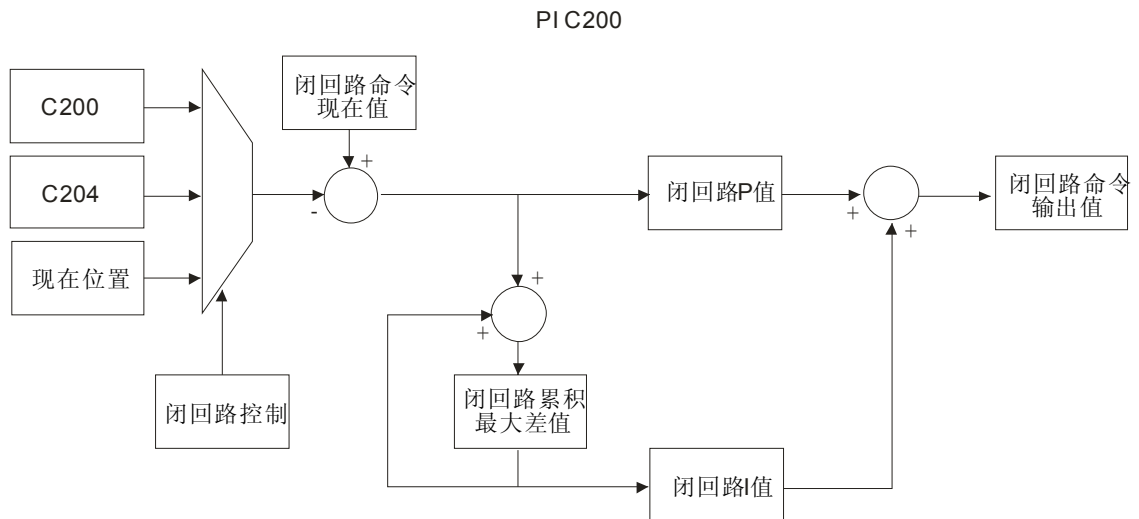
**闭回路
PID控制**

D1877、
D1878、D1880、
D1882、
D1888

1. D1877 为闭回路 PID 控制寄存器，用来启动闭回路控制功能。其闭回路的回授来源有三种可以选择。

寄存器值	回授来源
1	C200
2	C204
3	现在位置(D1848、D1928、D2008)
其它	闭回路控制功能禁能

2. D1878、D1879 为闭回路命令现在值，回授来源透过闭回路控制，使输出值与此设定值相同。
3. D1880、D1881 为闭回路 P 值，用来设定闭回路之比例增益。
4. D1882、D1883 为闭回路 I 值，用来设定闭回路之积分增益。
5. D1888、D1889 为闭回路累积最大差值寄存器，用来限制输入值和输出值两值之间差。
6. 下图为闭回路控制之方块图：



使用闭回路控制选择回授来源后，和闭回路命令现在值做比较，经由 PID 闭回路控制使得回授来源值等于闭回路命令现在值。回授来源值和闭回路命令现在值的差值储存在闭回路差值寄存器。设定闭回路累积最大差值用来限制回授来源与闭回路命令的差值范围，若超出最大值则停止输出。

3 DVP-PM 各种装置功能

3.12 手动运动模式特殊寄存器

运动模式特殊寄存器 (特 D)，它的种类及功能如下所示。详细说明可参考本节的功能说明，使用者可参考各特殊寄存器之使用说明，进一步设定参数值以及了解系统信息。

特 D 编号								特殊寄存器名称	设定范围	出厂值
X 轴		Y 轴		Z 轴		A 轴				
HW*1	LW*1	HW	LW	HW	LW	HW	LW			
	D1816		D1896		D1976		D2056	参数设定	b0 ~ b15	H0
D1819	D1818	D1899	D1898	D1979	D1978	D2059	D2058	马达转一圈所需脉冲数 (A)	1 ~ +2,147,483,647 PLS/REV	K2,000
D1821	D1820	D1901	D1900	D1981	D1980	D2061	D2060	马达转一圈之移动距离 (B)	1 ~ +2,147,483,647 *2	K1,000
D1823	D1822	D1903	D1902	D1983	D1982	D2063	D2062	最高速度	0 ~ +2,147,483,647 *3	K500,000
D1825	D1824	D1905	D1904	D1985	D1984	D2065	D2064	启动速度	0 ~ +2,147,483,647 *3	K0
D1827	D1826	D1907	D1906	D1987	D1986	D2067	D2066	寸动 JOG 速度 V _{JOG}	0 ~ +2,147,483,647 *3	K5,000
D1829	D1828	D1909	D1908	D1989	D1988	D2069	D2068	原点复归速度 V _{RT}	0 ~ +2,147,483,647 *3	K50,000
D1831	D1830	D1911	D1910	D1991	D1990	D2071	D2070	原点复归减速速度 V _{CR}	0 ~ +2,147,483,647 *3	K1,000
	D1832		D1912		D1992		D2072	零点 (PG0) 信号数 N	0 ~ +32,767 PLS	K0
	D1833		D1913		D1993		D2073	脉冲信号数 P	-32,768 ~ +32,767 PLS	K0
D1835	D1834	D1915	D1914	D1995	D1994	D2075	D2074	原点位置定义 HP	0 ~ ±999,999 *1	K0
	D1836		D1916		D1996		D2076	加速时间 T _{ACC}	10 ~ +32,767 ms	K100
	D1837		D1917		D1997		D2077	减速时间 T _{DEC}	10 ~ +32,767 ms	K100
D1839	D1838	D1919	D1918	D1999	D1998	D2079	D2078	目标位置(I) P(I)	-2,147,483,648 ~ +2,147,483,647 *1	K0
D1841	D1840	D1921	D1920	D2001	D2000	D2081	D2080	运转速度(I) V(I)	-2,147,483,648 ~ +2,147,483,647 *1	K1000
D1843	D1842	D1923	D1922	D2003	D2002	D2083	D2082	目标位置(II) P(II)	-2,147,483,648 ~ +2,147,483,647 *1	K0
D1845	D1844	D1925	D1924	D2005	D2004	D2085	D2084	运转速度(II)V(II)	0 ~ +2,147,483,647 *2	K2,000
	D1846		D1926		D2006		D2086	运转命令	b0 ~ b15	H0
	D1847		D1927		D2007			工作模式	b0 ~ b15	H0
D1849	D1848	D1929	D1928	D2009	D2008	D2089	D2088	现在位置 CP(PLS)	-2,147,483,648 ~ +2,147,483,647 *1	K0
D1851	D1850	D1931	D1930	D2011	D2010	D2091	D2090	现在速度 CS(PPS)	0 ~ +2,147,483,647 PPS	K0
D1853	D1852	D1933	D1932	D2013	D2012	D2093	D2092	现在位置 CP(unit *3)	-2,147,483,648 ~ +2,147,483,647 *1	K0
D1855	D1854	D1935	D1934	D2015	D2014	D2095	D2094	现在速度 CS(unit *3)	0 ~ +2,147,483,647 PPS	K0
	D1856		D1936		D2016		D2096	执行状态	b0 ~ b15	H0
	D1857		D1937		D2017		D2097	错误编号	请参考错误码表	H0
	D1858		D1938		D2018		D2098	电子齿轮 (分子)	1 ~ +32,767	K1
	D1859		D1939		D2019		D2099	电子齿轮 (分母)	1 ~ +32,767	K1
D1861	D1860	D1941	D1940	D2021	D2020	D2101	D2100	手摇轮输入频率	由手摇轮输入的脉冲频率	K0
D1863	D1862	D1943	D1942	D2023	D2022	D2103	D2102	累计手摇轮输入脉冲个数	计数由手摇轮输入的脉冲个数	K0
	D1864		D1944		D2024		D2104	手摇轮输入响应速度	手摇轮输入响应速度	K5

特 D 编号								特殊寄存器名称	设定范围	出厂值
X 轴		Y 轴		Z 轴		A 轴				
HW*1	LW*1	HW	LW	HW	LW	HW	LW			
D1865								停止模式 (OX0-99)	依需求设定对应值	K0
D1867	D1866	D1947	D1946	D2027	D2026	D2107	D2106	电子原点	依需求设定对应值	K0
D1868								程序号码指定方式	依需求设定对应值	K0
D1869								Ox 错误 STEP 位置	依需求设定对应值	K0
D1872								Ready 输出 high bye	依需求设定对应值	K0
D1873								M code 输出 high bye	依需求设定对应值	K0
D1874								M code off 输入 X 起点编号	依需求设定对应值	K0
D1875		D1955						外部手动启动 (ZRN, MPG, JOG-, JOG+)	依需求设定对应值	K0

*1: HW: High Word; LW: Low Word。

*2: 单位可选择 um/rev、m deg/rev 及 10⁻⁴ inch/rev。

*3: 设定单位依照 D1816、D1896、D1976 "参数设定" 之 b0, b1 单位系设定而变化。

3.12.1 手动运动模式特殊寄存器功能说明

X 轴		Y 轴		Z 轴		A 轴		参数设定
HW	LW	HW	LW	HW	LW	HW	LW	
	D1816		D1896		D1976		D2056	

[说明]

bit0~bit15 各位所代表的意义如下所示:

◆ D1816 (D1896, D1976, D2056) 之 b0, b1: 设定单位系

b1	b0	单位系	说明
0	0	马达单位	以脉冲为单位
0	1	机械单位	以长度, 角度为单位
1	0	复合单位	位置以长度, 角度 (机械单位) 为单位, 速度则以脉冲 (马达单位) 为单位
1	1		

	马达单位	复合单位	机械单位
位置	pulse	um	
	pulse	m deg	
	pulse	10-4inch	
速度	pulse/sec		cm/min
	pulse/sec		10deg/min
	pulse/sec		inch/min

- 位置数据: 原点位置【HP】、目标位置(I)【P(I)】、目标位置(II)【P(II)】、现在位置【CP】。

3 DVP- PM 各种装置功能

- 速度数据：最高速度【V_{MAX}】、启动速度【V_{BIAS}】、寸动 JOG 速度【V_{JOG}】、原点复归速度【V_{RT}】、原点减速速度【V_{CR}】、运转速度(I)【V(I)】、运转速度(II)【V(II)】。

- 范例一：

马达单位 b[1:0] = 00 ⇒ 单位系：位置数据：Pulse，速度数据：Pulse/sec (PPS)。设定：目标位置 P(I)：10,000 (Pulse)，运转速度 V(I)：10K (PPS)

[说明]

位置控制器只要送出 10,000 个 Pulse (频率为 10kPPS)，可移动至目标位置，每个 Pulse 所移动之距离，乃是由使用者依据设备参数计算每一个脉冲可移动多少距离。

- 范例二：

机械单位 b[1:0]=01 ⇒ 单位系：位置数据：um，速度数据：cm/min。若设定 D1818 (D1898, D1978) = 1,000 (Pulse/REV)，D1820 (D1900, D1980) = 100 (um/REV)，且目标位置 P(I)=10,000 (um)，运转速度 V(I)=6 (cm/min)，求位置控制器的脉冲指令的脉冲个数及其频率？

[说明]

$$\text{距离} = \frac{\text{距离}}{\underbrace{\text{圈}}_B} \times \frac{\text{圈}}{\underbrace{\text{脉冲数}}_{1/A}} \times \text{脉冲数}$$

$$\text{由位置控制器计算要跑到 } P(I) \text{ 所需之脉冲数} = \frac{P(I) \text{um}}{B/A} = P(I) \times \frac{A}{B} = 100,000 \text{ Pulse}$$

$$\text{运转速度 } V(I): 6 \text{ (cm/min)} = 60,000/60 \text{ (um/sec)}$$

$$\text{速度} = \frac{\text{距离}}{\text{时间}} = \frac{\text{距离}}{\underbrace{\text{圈}}_B} \times \frac{\text{圈}}{\underbrace{\text{脉冲数}}_{1/A}} \times \frac{\text{脉冲数}}{\text{时间}} \quad \text{PPS, pulse/sec}$$

$$\text{由位置控制器计算脉冲频率(PPS)} = V(I) \times \frac{10^4}{60} \times \frac{A}{B} = \frac{60,000}{60} \times \frac{1,000}{100} = 10,000 \text{ PPS}$$

- 范例三：

复合单位 b[1:0] = 10, 11 ⇒ 单位系：位置数据：um，速度数据：pulse/sec (PPS)

D1818 (D1898, D1978) = 2,000 (Pulse/REV)，D1820 (D1900, D1980) = 100 (um/REV)，若目标位置 P(I)：10,000 (um)，运转速度 V(I)：10K (PPS)，求位置控制器的脉冲指令的脉冲个数？

[说明]







$$\text{由位置控制器计算要跑到 P(I) 所需之脉冲数} = \frac{P(I) \mu\text{m}}{\frac{B}{A}} = P(I) \times \frac{A}{B} = 200,000 \text{ PULSE}$$

◆ D1816 (D1896, D1976, D2056) 之 b2, b3: 位置数据倍率设定

位置相关数据，原点位置【HP】、目标位置(I)【P(I)】、目标位置(II)【P(II)】、现在位置【CP】的设定值必须乘以下倍率值。

b3	b2	位置数据倍率
0	0	位置数据 × 10 ⁰
0	1	位置数据 × 10 ¹
1	0	位置数据 × 10 ²
1	1	位置数据 × 10 ³

◆ D1816 (D1896, D1976, D2056) 之 b4, b5: 脉冲输出方式

b5	b4	脉冲输出型式 (正逻辑)	说明
0	0	FP 正转脉冲  RP 反转脉冲 	双脉冲
0	1	FP 脉冲  RP 方向(DIR) 	单脉冲
1	0	FP A相脉冲 	A/B 相脉冲
1	1	RP B相脉冲  正转 反转	

◆ D1816 (D1896, D1976, D2056) 之 b6: PWM 模式设定

b[6]=1 时，启动 JOG+运转，则 Y0~Y3 输出 PWM

◆ D1816 (D1896, D1976, D2056) 之 b8: 原点复归方向

b[8]=0 时，执行原点复归方向，往现在位置 (CP) 内容值反 (递减) 方向执行。

b[8]=1 时，执行原点复归方向，往现在位置 (CP) 内容值正 (递增) 方向执行。

◆ D1816 (D1896, D1976, D2056) 之 b9: 原点复归模式

b[9]=0 时，正常模式。即为侦测近点 (DOG) 信号产生后，经过 N 个原点复归零点讯号后，再经过 P 个原点复归脉冲讯号后，马达立刻停止。

b[9]=1 时，覆写模式。即为侦测近点 (DOG) 信号产生后，当零点信号数 N 脉冲及脉冲信号数 P 脉冲

3 DVP-PM 各种装置功能

其中一项先到达时，马达立刻停止。

◆ D1816 (D1896, D1976, D2056) 之 b10: 原点复归 DOG 下降缘侦测

b[10]=0 时，原点复归 DOG 下降缘侦测 On，即为 DOG 下降缘侦测。

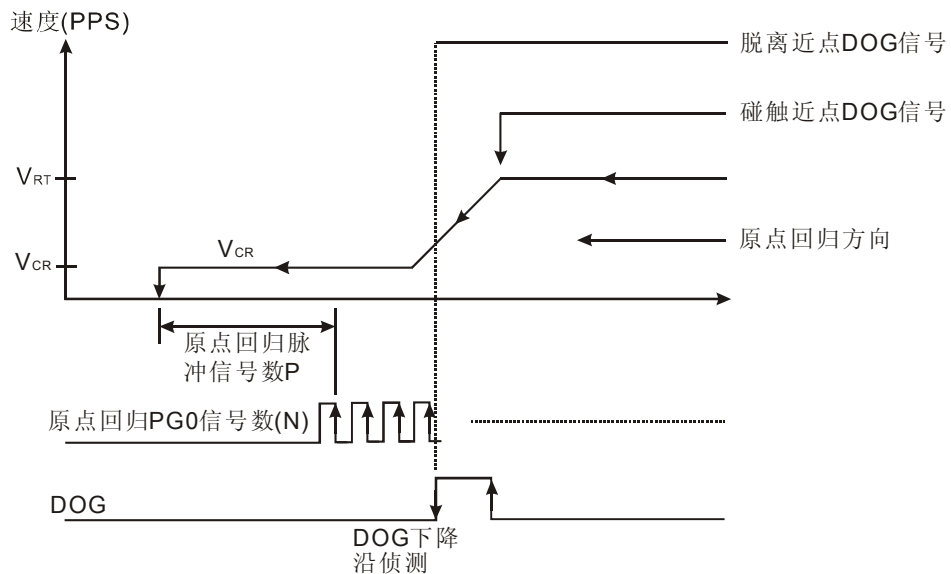
b[10]=1 时，原点复归 DOG 下降缘侦测 Off，即为 DOG 上升缘侦测。

1. b[9:10]=00⇒正常模式，原点复归 DOG 下降缘侦测 On

原点复归动作：马达以原点复归速度 VRT 运转，当碰到近点 (DOG) 信号时，马达减速到以原点复归减速速度 VCR 运转，再侦测近点 (DOG) 信号下降缘产生后，经过 N 个原点复归之零点信号脉冲后，再经过 P 个原点复归之脉冲信号脉冲后马达停止。

若设定原点复归之 PG0 零点信号数 N 或原点复归之脉冲信号数 P 太小时，碰到近点 (DOG) 信号时，马达开始向原点复归减速速度 VCR 做减速动作，再侦测近点 (DOG) 信号下降缘产生后，若达到指定之 N 个 PG0 零点信号，再经过 P 个原点复归之脉冲信号立即停止 (不论是否已到达原点复归减速速度 VCR)。

假设原点复归之信号数 N 设定为 0，原点复归之脉冲信号数 P 设定为 0，则碰到近点 (DOG) 信号后再侦测近点 (DOG) 信号下降缘产生后立即停止。

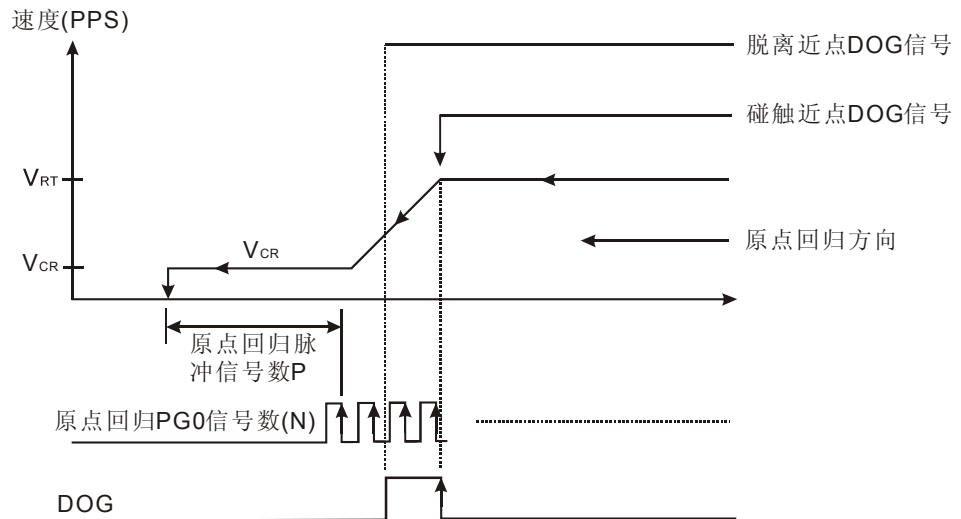


2. b[9:10]=01⇒正常模式，原点复归 DOG 下降缘侦测 Off

原点复归动作：马达以原点复归速度 VRT 运转，当碰到近点 (DOG) 信号时，马达开始向原点复归减速速度 VCR 做减速动作，经过 N 个原点复归之零点信号脉冲后，再经过 P 个原点复归之脉冲信号脉冲后马达停止。

若设定原点复归之 PG0 零点信号数 N 或原点复归之脉冲信号数 P 太小时，碰到近点 (DOG) 信号时，马达开始向原点复归减速速度 VCR 做减速动作，若达到 N 个指定之 PG0 零点信号数，再经过 P 个原点复归之脉冲信号立即停止 (不论是否已到达原点复归减速速度 VCR)。

假设原点复归之信号数 N 设定为 0，原点复归之脉冲信号数 P 设定为 0，则碰到近点 (DOG) 信号后立即停止。

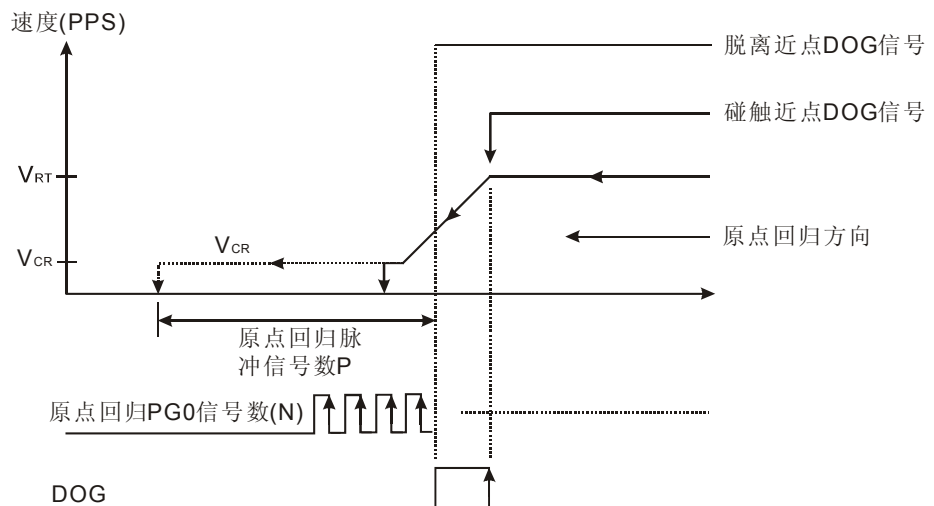


3. b[9:10]=10⇒覆写模式，原点复归 DOG 下降缘侦测 On

原点复归动作：马达以原点复归速度 V_{RT} 运转，当碰到近点 (DOG) 信号时，马达开始向原点复归减速速度 V_{CR} 做减速动作，再侦测近点 (DOG) 信号下降缘产生后，此时经过 N 个原点复归之零点信号脉冲后，或经过 P 个原点复归之脉冲信号脉冲后马达停止。

若设定原点复归之 PG0 零点信号数 N 或原点复归之脉冲信号数 P 太小时，碰到近点 (DOG) 信号时，马达减速到原点复归减速速度 V_{CR} 运转，若达到 N 个指定之 PG0 零点信号，或达到 P 个原点复归之脉冲信号立即停止 (不论是否已到达原点复归减速速度 V_{CR})。

假设原点复归之信号数 N 设定为 0，原点复归之脉冲信号数 P 设定为 0，则马达侦测近点 (DOG) 信号下降缘产生后立即停止。



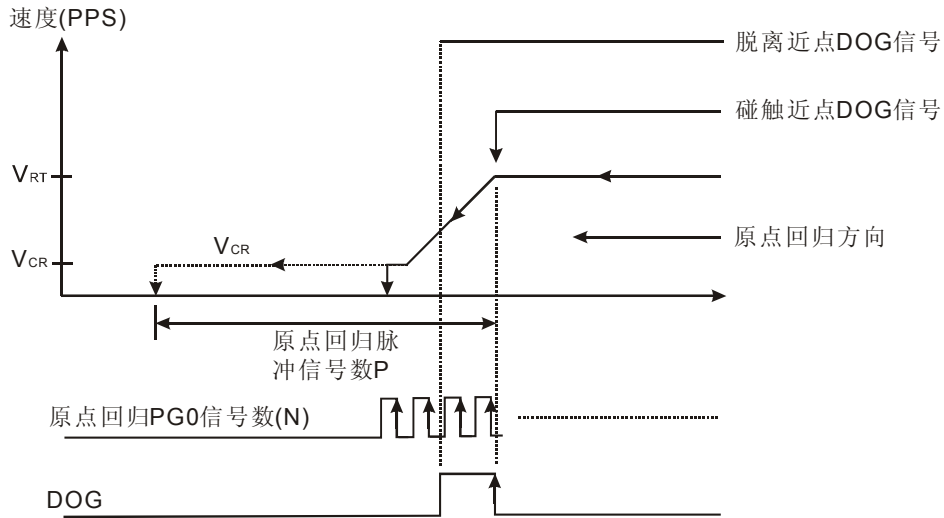
4. b[9:10]=11⇒覆写模式，原点复归 DOG 下降缘侦测 Off

原点复归动作：马达以原点复归速度 V_{RT} 运转，当碰到近点 (DOG) 信号时，马达开始向原点复归减速速度 V_{CR} 做减速动作，此时经过 N 个原点复归之零点信号脉冲后，或经过 P 个原点复归之脉冲信号后马达停止。

若设定原点复归之 PG0 零点信号数 N 或原点复归之脉冲信号数 P 太小时，碰到近点 (DOG) 信号时，马达开始向原点复归减速速度 V_{CR} 做减速动作，若达到 N 个指定之 PG0 零点信号，或达到 P 个原点复归之脉冲信号数立即停止 (不论是否已到达原点复归减速速度 V_{CR})。

3 DVP-PM 各种装置功能

假设原点复归之信号数 N 设定为 0 ，原点复归之脉冲信号数 P 设定为 0 ，或马达碰到近点 (DOG) 信号后立即停止。



- D1816 (D1896, D1976, D2056) 之 b11: 旋转方向
- b[11]=0 时，当正转运动时，现在位置 (CP) 内容值递增。
- b[11]=1 时，当正转运动时，现在位置 (CP) 内容值递减。
- D1816 (D1896, D1976, D2056) 之 b12: 绝对/相对坐标设定
- 当 b[12]=0 时，设定为绝对坐标定位。
- 当 b[12]=1 时，设定为相对坐标定位。
- D1816 (D1896, D1976, D2056) 之 b13: DOG 触发启动模式
- b[13]=0 时，DOG 信号上升缘触发。
- b[13]=1 时，DOG 信号下降缘触发 (插入单段速定位运动模式、插入二段速定位运动模式有效)。
- D1816 (D1896, D1976, D2056) 之 b14: 加减速曲线选择
- b[14]=0 时，为梯形加减速曲线。
- b[14]=1 时，为 S 加减速曲线。

X 轴		Y 轴		Z 轴		A 轴		马达转一圈所需脉冲数 (A)
HW	LW	HW	LW	HW	LW	HW	LW	
D1819	D1818	D1899	D1898	D1979	D1978	D2059	D2058	

[说明]

1. 由于伺服驱动器内具有电子齿轮比例设定功能，因此伺服马达转一圈所需之脉冲数并不一定等于伺服

马达旋转一圈，编码器所产生的脉冲数：

馬達轉一圈所需指令脈波數 $A \times \text{電子齒輪} (CMX / CDV) = \text{編碼器旋轉一圈產生脈波數}$

设定单位依照 D1816 (D1896, D1976, D2056) 参数设定之 b0, b1 单位系设定来变化，当系统单位系设定为机械单位或复合单位时，参数 A 设定有效；若系统单位设定为马达单位，则参数 A 设定不作用。

X 轴		Y 轴		Z 轴		A 轴		马达转一圈之移动距离 (B)
HW	LW	HW	LW	HW	LW	HW	LW	
D1821	D1820	D1901	D1900	D1981	D1980	D2061	D2060	

[说明]

1. 马达转一圈所带动之移动量，单位有三种可供选择，由 D1816 (D1896, D1976, D2056) 之 b0, b1 作设定： $B = 1 \sim +2,147,483,647$ (um/Rev, mdeg/Rev, 10^{-4} inch/Rev)
2. 设定单位依照 D1816 (D1896, D1976, D2056) 之 b0, b1 参数设定单位系设定来变化。当系统单位系设定为机械单位或复合单位时，参数 B 设定有效；若系统单位设定为马达单位，则参数 B 设定不作用。

X 轴		Y 轴		Z 轴		A 轴		最高速度 V_{MAX}
HW	LW	HW	LW	HW	LW	HW	LW	
D1823	D1822	D1903	D1902	D1983	D1982	D2063	D2062	

[说明]

1. 各种操作模式的速度上限。设定范围 $0 \sim +2,147,483,647$ (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 设定)。
2. 对应运动指令 $10 \sim 500kPPS$ ，若大于 500k，则以 500k 输出，若小于 10，则以 10 输出。

X 轴		Y 轴		Z 轴		A 轴		启动速度 V_{BIAS}
HW	LW	HW	LW	HW	LW	HW	LW	
D1825	D1824	D1905	D1904	D1985	D1984	D2065	D2064	

[说明]

1. 脉冲输出启动速度。设定范围 $0 \sim +2,147,483,647$ (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1

3 DVP-PM 各种装置功能

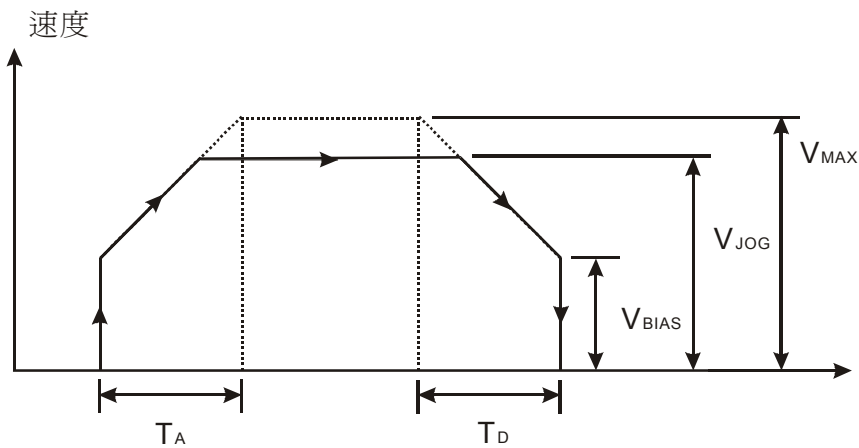
设定)。

2. 对应运动指令 0~500kPPS。若大于 500k，则以 500k 输出，若小于 0，则以 0 输出。
3. 若为步进驱动系统，请注意步进马达的共振区频率，将启动速度 V_{BIAS} 设定在共振区频率之上，作安全启动。

X 轴		Y 轴		Z 轴		A 轴		寸动 JOG 速度 V_{JOG}
HW	LW	HW	LW	HW	LW	HW	LW	
D1827	D1826	D1907	D1906	D1987	D1986	D2067	D2066	

[说明]

1. 寸动指令速度。设定范围 0~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 设定)。
2. 对应的脉冲指令 10~500kPPS。若大于 500k，则以 500k 输出，若小于 10，则以 10 输出。
3. 设定范围限制 $V_{MAX} > V_{JOG} > V_{BIAS}$ 。若 V_{JOG} 设定 $> V_{MAX}$ 设定，则 V_{JOG} 输出 = V_{MAX} ，若 V_{JOG} 设定 $< V_{BIAS}$ 设定，则 V_{JOG} 输出 = V_{BIAS} 。
4. 寸动 JOG 速度 V_{JOG} 在执行中不允许改变。



X 轴		Y 轴		Z 轴		A 轴		原点复归速度 V_{RT}
HW	LW	HW	LW	HW	LW	HW	LW	
D1829	D1828	D1909	D1908	D1989	D1988	D2069	D2068	

[说明]

1. 回到机械原点的速度。设定范围 0~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 设定)。

2. 对应的脉冲指令 10~500kPPS。若大于 500k，则以 500k 输出，若小于 10，则以 10 输出。
3. 设定范围限制 $V_{MAX} > V_{RT} > V_{BIAS}$ 。
4. 原点复归速度 V_{RT} 在执行中不允许改变。

X 轴		Y 轴		Z 轴		A 轴		原点复归减速速度 V_{CR}
HW	LW	HW	LW	HW	LW	HW	LW	
D1831	D1830	D1911	D1910	D1991	D1990	D2071	D2070	

[说明]

1. 设定范围 0~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 设定)。
2. 对应的脉冲指令 10~500kPPS。若大于 500k，则以 500k 输出，若小于 10，则以 10 输出。
3. 执行原点复归时，马达以原点复归速度 V_{RT} 运转，当碰到近点 (DOG) 信号时，马达减速到原点复归减速速度 V_{CR} 运转。
4. 为了精准的定位在原点，建议将原点复归减速速度 V_{CR} 设定为低速运行。
5. 原点复归减速速度 V_{CR} 在执行中不允许改变。

X 轴		Y 轴		Z 轴		A 轴		原点复归之零点(PG0) 信号数 N
HW	LW	HW	LW	HW	LW	HW	LW	
	D1832		D1912		D1992		D2072	

[说明]

1. 设定范围 -32,768~32,767 (PULSE)。正值设定为正方向的零点信号数 N，负值设定为反方向的零点信号数 N。
2. 马达减速停止之参考信号，请参考 D1816 (D1896, D1976, D2056) 之 b9, b10 原点复归模式之说明。

X 轴		Y 轴		Z 轴		A 轴		原点复归之脉冲信号数 P
HW	LW	HW	LW	HW	LW	HW	LW	
	D1833		D1913		D1993		D2073	

[说明]

1. 设定范围 -32,768~32,767 (PULSE)。正值设定为正方向的脉冲信号数 P，负值设定为反方向的脉冲信号数 P。

3 DVP- PM 各种装置功能

2. 马达减速停止之参考信号，请参考 D1816 (D1896, D1976, D2056) 之 b9, b10 原点复归模式之说明

X 轴		Y 轴		Z 轴		A 轴		原点位置定义 HP
HW	LW	HW	LW	HW	LW	HW	LW	
D1835	D1834	D1915	D1914	D1995	D1994	D2075	D2074	

[说明]

1. 设定范围 0~±999,999 (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 定义)。
2. 原点复归完成时，现在位置 CP 会被更新为原点位置 HP。

X 轴		Y 轴		Z 轴		A 轴		加速时间 T _{ACC}
HW	LW	HW	LW	HW	LW	HW	LW	
	D1836		D1916		D1996		D2076	

[说明]

1. 从 D1824 (D1904, D1984, D2064) 启动速度 V_{BIAS} 加速到 D1822 (D1902, D1982, D2062) 最高速度 V_{MAX} 所需的时间。
2. 当设定值<10ms 时，则视为 10ms，设定值>32,767ms 时，则视为 32,767 ms。
3. S 加速曲线控制时，若要完整之 S 加速曲线控制，请将最高速度 V_{MAX} 设定成与运转速度相同

X 轴		Y 轴		Z 轴		A 轴		减速时间 T _{DEC}
HW	LW	HW	LW	HW	LW	HW	LW	
	D1837		D1917		D1997		D2077	

[说明]

1. 从 D1822 (D1902, D1982, D2062) 最高速度 V_{MAX} 减速到 D1824 (D1904, D1984, D2064) 启动速度 V_{BIAS} 所需的时间。
2. 当设定值<10ms 时，则视为 10ms，设定值>32,767 ms 时，则视为 32,767 ms。
3. S 加速曲线控制时，若要完整之 S 加速曲线控制，请将最高速度 V_{MAX} 设定成与运转速度相同。

X 轴		Y 轴		Z 轴		A 轴		目标位置(I) P(I)
HW	LW	HW	LW	HW	LW	HW	LW	
D1839	D1838	D1919	D1918	D1999	D1998	D2079	D2078	

[说明]

1. 设定范围 -2,147,483,648~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 定义)。
2. 目标位置 P(I)数据属性:
 - 绝对坐标 (D1816 (D1896, D1976, D2056) 之 b12=0)
代表从 0 开始之位置, 当目标位置(I) P(I)大于现在位置 (D1848 (D1928, D2008, D2088)) 时, 马达正转, 小于现在位置时, 马达反转。
 - 相对坐标 (D1816 (D1896, D1976, D2056) 之 b12=1)
代表从现在位置 (D1848 (D1928, D2008, D2088)) 开始算马达行走之距离, 当相对坐标为正值时, 马达正转, 相对坐标为负值时, 马达反转。
3. 目标位置(I) P(I)数据倍率依照 D1816 (D1896, D1976, D2056) 参数设定之 b2, b3 倍率设定来变化。

X 轴		Y 轴		Z 轴		A 轴		运转速度(I) V(I)
HW	LW	HW	LW	HW	LW	HW	LW	
D1841	D1840	D1921	D1920	D2001	D2000	D2081	D2080	

[说明]

1. 设定范围 0~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 定义)。
2. 对应的脉冲指令 10~500kPPS。若大于 500k, 则以 500k 输出, 若小于 10, 则以 10 输出。
3. 设定范围限制: $V_{MAX} > V(I) > V_{BIAS}$ 。
4. 当变速度运转时 (D1846 (D1926, D2006, D2086) 之 b4=1), 运转速度(I) V(I)在运转中可改变。V(I) 的符号为 '+' 时正转, 符号为 '-' 时反转。

X 轴		Y 轴		Z 轴		A 轴		目标位置(II) P(II)
HW	LW	HW	LW	HW	LW	HW	LW	
D1843	D1842	D1923	D1922	D2003	D2002	D2083	D2082	

[说明]

1. 设定范围 0~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 定义)。

3 DVP- PM 各种装置功能

2. 目标位置 P(II) 数据属性:

- 绝对坐标 (D1816 (D1896, D1976, D2056) 之 b12=0)

代表从 0 开始之位置, 当目标位置(II) P(II)大于现在位置 (D1848 (D1928, D2008, D2088)) 时, 马达正转, 小于现在位置时, 马达反转。

相对坐标 (D1816 (D1896, D1976, D2056) 之 b12=1)

代表从现在位置 (D1848 (D1928, D2008, D2088)) 开始算马达行走之距离, 当相对坐标为正值时, 马达正转, 相对坐标为负值时, 马达反转。

- #### 3. 目标位置(II) P(II) 数据倍率依照 D1816 (D1896, D1976, D2056) 参数设定之 b2, b3 倍率设定来变化。

X 轴		Y 轴		Z 轴		A 轴		运转速度(II)V(II)
HW	LW	HW	LW	HW	LW	HW	LW	
D1845	D1844	D1925	D1924	D2005	D2004	D2085	D2084	

[说明]

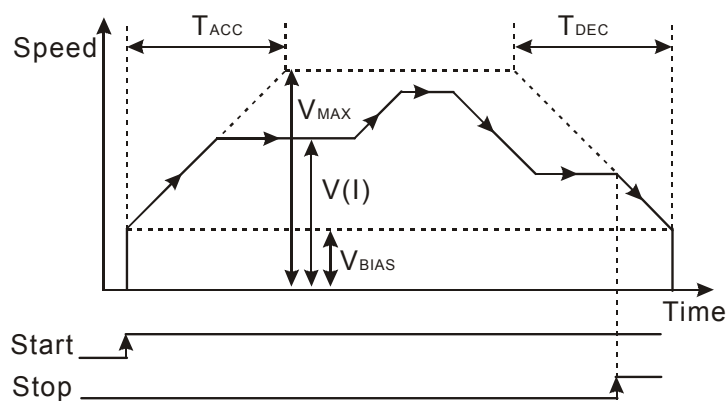
- 设定范围 $-2,147,483,648 \sim +2,147,483,647$ (单位由 D1816 (D1896, D1976, D2056) 之 b0, b1 定义)。
- 对应的脉冲指令 10~500kPPS。若大于 500k, 则以 500k 输出, 若小于 10, 则以 10 输出。
- 设定范围限制: $V_{MAX} > V(II) > V_{BIAS}$ 。

X 轴		Y 轴		Z 轴		A 轴		运转命令
HW	LW	HW	LW	HW	LW	HW	LW	
	D1846		D1926		D2006		D2086	

[说明]

- D1846 (D1926, D2006, D2086) 之 b0: 软件 STOP。
 - 动作时机: b[0]上缘触发(0→1)。
 - 此信号与外部输入强制停止信号 (STOP) 功能完全相同。定位控制器均会做减速停止定位。
- D1846 (D1926, D2006, D2086) 之 b1: 软件 START。
 - 当 b[1]上缘触发(0→1)时, 开始运行启动, 根据 D1846 (D1926, D2006, D2086) 运转命令设定运作。

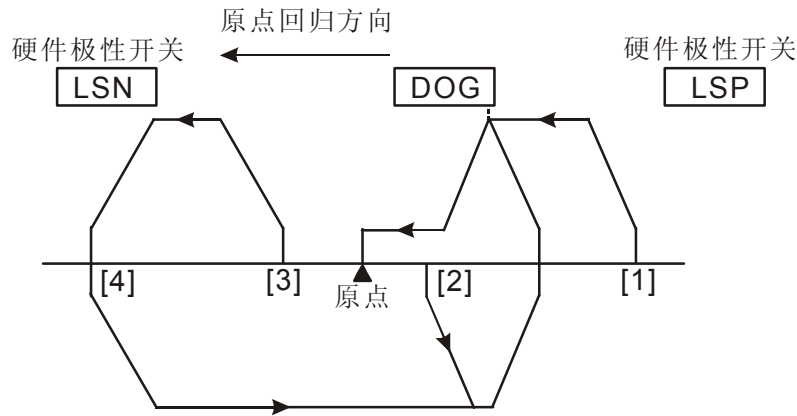
3. D1846 (D1926, D2006, D2086) 之 b2: JOG+运转。
 - 当 b[2]=1 时, JOG+运转发送正方向脉冲 (CW)。
 - 配合 X-Y-Z-A 轴参数设定(D1816、D1896、D1976、D2056)之 b[6] PWM 模式, 当 b[2]=1 时, 启动脉冲频宽调变(PWM)输出。
4. D1846 (D1926, D2006, D2086) 之 b3: JOG-运转。
 - 当 b[3]=1 时, JOG-运转发送反方向脉冲 (CCW)。
5. D1846 (D1926, D2006, D2086) 之 b4: 变速度运转模式。
 - 当 b[4]被触发即接收到变速度运动命令后, 且启动运转 (START On) 后, 这时定位控制器开始执行变速度 V(I) 运转, 开始由 DVP-PM 脉冲产生单元送出脉冲。
 - 动作说明: 运转速度会由启动速度 V_{BIAS} 开始加速至期望之设定速度 V(I) 稳态运行, 此时于脉冲产生期间, 使用者可以任意的更改速度 V(I), DVP-PM 脉冲产生单元会自动的依循所设定的速度执行加速 / 减速。注意, 外部 STOP 输入接点讯号无法停止变速度模式, 须在运转命令(D1846、D1926、D2006)中设定软件 STOP(b[0]=1), 停止变速度运转模式; 或是将 b4 设定为 0 即可停止变速度运转模式。
 - 动作示意图:



6. D1846 (D1926, D2006, D2086) 之 b5: 手摇轮输入操作。
 - 当 b[5]=1 时, 启动手摇轮脉冲输入功能。细部设定请参照 D1858~D1864 (D1938~D1944, D2018~D2024)。
7. D1846 (D1926, D2006, D2086) 之 b6: 原点复归模式启动。
 - 当 b[6]=0→1 变化时, 开始执行原点复归动作, 由于目前位置 (CP) 所处的位置不同, 执行原点复归动作也有所不同, 可分为四种情况:

3 DVP- PM 各种装置功能

原点复归运动行程:



位置(1): 起始位置[1], 在原点及 DOG 信号的右边, DOG 信号=Off。

位置(2): 起始位置[2]在原点的右边, DOG 信号=On。

位置(3)*: 起始位置[3]在原点及 DOG 信号的左边, DOG 信号=Off, LSN 信号=Off。

位置(4)*: 起始位置如上图位置[4]在原点及 DOG 信号的左边, DOG 信号=Off, LSN 信号 =On。

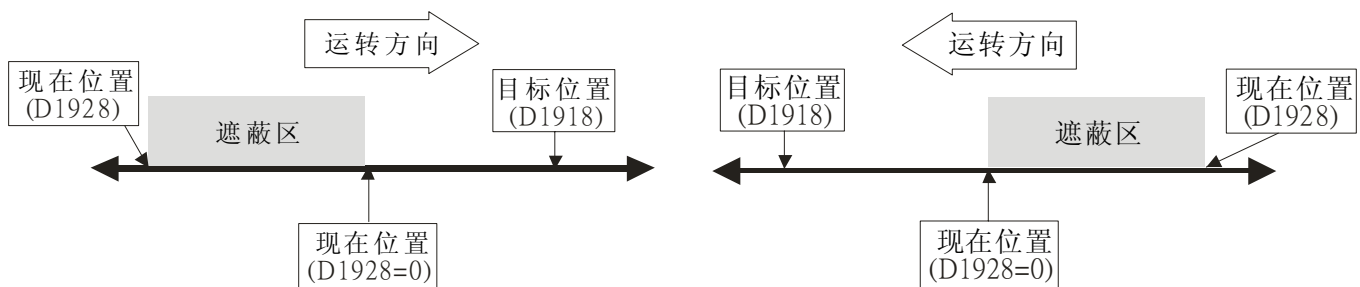
备注*: 10PM 无 LSN、LSP 讯号, 故不支持位置(3)与位置(4)之动作。

8. D1926 之 b7: 单段速外部触发模式启动

- 当 b[7]被触发即接收到单段速外部触发命令时, 且接受到 DOG1 触发讯号后, 开始执行单段速运动, 当在非屏蔽区接受到 PG1 之讯号触发后, 则立即停止运行, 且在 PG1 讯号触发后经 20μs, CLR1 会接着触发。运动速度根据【V(l)】决定, 由脉冲产生单元送出脉冲。
- 屏蔽区范围如下图所示, 由现在位置与目标位置的相对位置决定, 若现在位置小于目标位置, 运行方向为递增, 在(D1928=0)时脱离屏蔽区的范围; 若现在位置大于目标位置, 则运行方向为递减, 在 D1928=0 时脱离屏蔽区的范围。

运转方向为正向

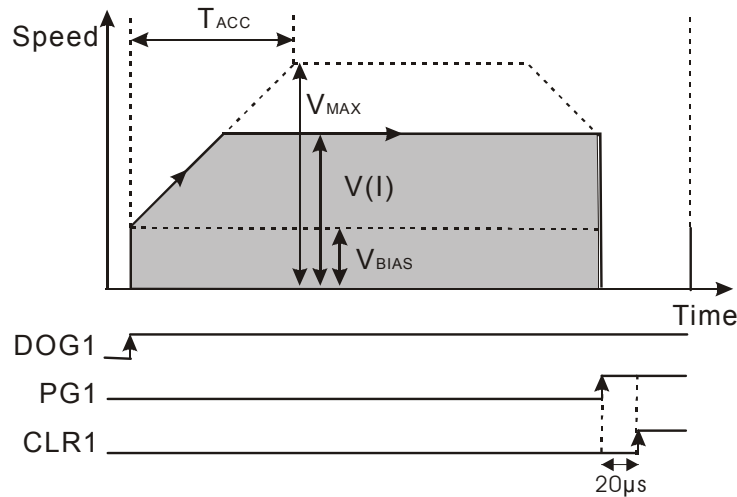
运转方向为反向



运动方向:

- 相对坐标定位由设定位置寄存器 P(l)内容值之符号位决定。
- 绝对坐标定位由目标位置(l) P(l) (D1838 (D1918, D1998)) 大于现在位置时正转, 小于时反转。

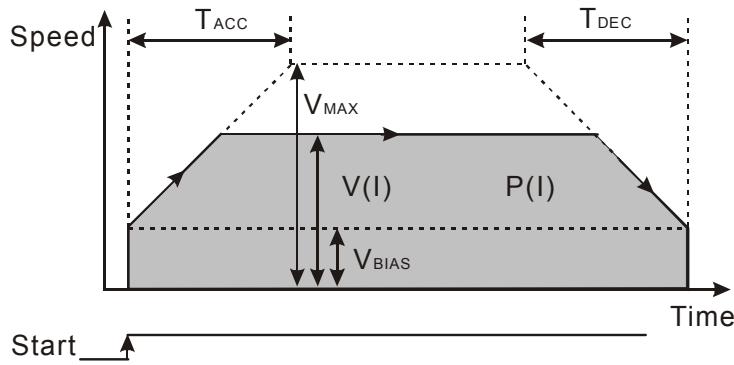
- 而运转速度会由启动速度 V_{BIAS} 开始加速至期望之设定速度 $V(I)$ 稳态运行, 当 PG1 讯号 ON 后脉冲立即停止输出。
- 应用的寄存器有 V_{BIAS} : D1824 (D1904, D1984) 、 $V(I)$: D1840(D1920, D2000)、 V_{MAX} : D1822 (D1902, D1982) 、 T_{ACC} : D1836 (D1916, D1996)等寄存器。



9. D1846 (D1926, D2006, D2086) 之 b8: 单段速定位运动模式启动。

- 当 b[8]被触发即接收到单段速定位运动命令时, 且启动运转 (START On) 后, 开始执行第 1 段定位程序, 定位的步进数及运动速度根据【P(I)】和【V(I)】决定, 开始由脉冲产生单元送出脉冲。
- 相对坐标定位由设定位置寄存器 P(I)内容值之符号位决定。
- 绝对坐标定位由目标位置(I) P(I) (D1838 (D1918, D1998, D2078)) 大于现在位置时正转, 小于时反转。
- 而运转速度会由启动速度 V_{BIAS} 开始加速至期望之设定速度 $V(I)$ 稳态运行, 直到快到达位置寄存器 P(I)之内容值时, 脉冲产生单元会开始减速至启动速度而后停, 其间脉冲产生单元总共发送产生出 P(I)个脉冲数。
- 应用的寄存器有 V_{BIAS} : D1824 (D1904, D1984, D2054) 、 $V(I)$: D1840(D1920, D2000, D2080)、 V_{MAX} : D1822 (D1902, D1982, D2062) 、 P(I): D1838 (D1918, D1998, D2078) 、 T_{ACC} : D1836 (D1916, D1996, D2076) 及 T_{DEC} : D1837 (D1917, D1997, D2077) 等寄存器。

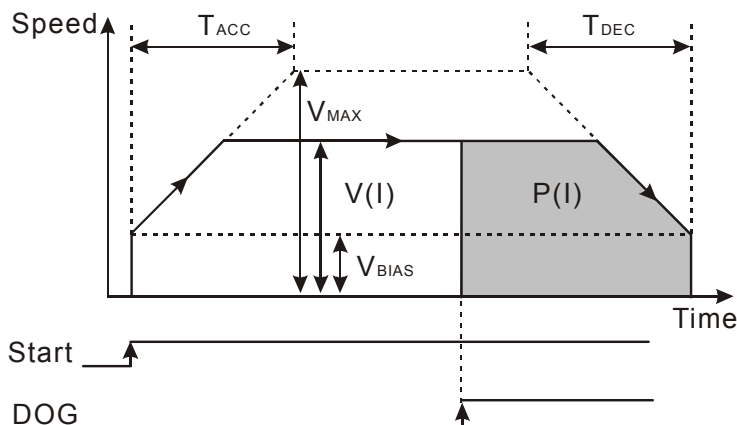
3 DVP-PM 各种装置功能



- 配合 X-Y-Z-A 轴参数设定(D1816、D1896、D1976、D2056)之 b[6] PWM 模式，当 b[8]=1 时，启动单段速定位模式，由 Y0~Y3 输出单向脉冲。

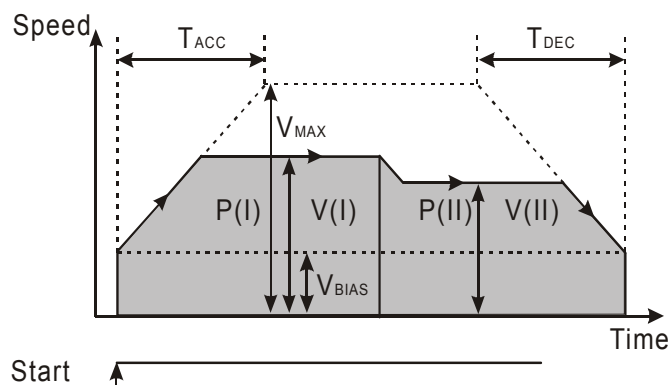
10. D1846 (D1926, D2006, D2086) 之 b9: 插入单段速定位运动模式启动

- 当 b[9]被触发即接收到插入单段速定位运动命令后，且启动运转 (START On) 后，开始由脉冲产生单元送出脉冲，当外部近点 DOG 信号动作时，则重新加载目标位置(I)【P(I)】内容值。
- 运转方向：相对坐标定位由设定位置寄存器 P(I)内容值之符号位决定；绝对坐标定位由目标位置 (I) P(I): D1838 (D1918, D1998, D2078) 大于现在位置时正转，小于时反转。
- 运转速度会由启动速度 V_{BIAS} 开始加速至期望之设定速度 $V(I)$ 稳态运行，于脉冲送出期间遇到 DOG 近点信号触发时，脉冲产生单元会再送出位置寄存器 P(I) 内容值之步进数，同样地此时和单段速定位运动模式一样，产生的脉冲个数直到快到达位置寄存器 P(I) 之内容值时，减速至启动速度而后停。
- 应用的寄存器有 V_{BIAS} : D1824 (D1904, D1984, D2054) 、 $V(I)$: D1840 (D1920, D2000, D2080) 、 V_{MAX} : D1822 (D1902, D1982, D2062) 、P(I): D1838 (D1918, D1998, D2078) 、 T_{ACC} : D1836 (D1916, D1996, D2076) 及 T_{DEC} : D1837 (D1917, D1997, D2077) 等寄存器。



11. D1846 (D1926, D2006, D2086) 之 b10: 连续两段速定位运动模式启动

- 当 b[10]被触发，连续两段速定位运动命令启动，启动运转 (START= On)，第 1 段位置定位完成目标位置(I)P(I) 后，会立即开始第 2 段位置定位动作。
- 运转方向：相对坐标定位由设定位置寄存器 P(I) 内容值之符号位决定；绝对坐标定位由目标位置(I) P(I)： D1838 (D1918, D1998, D2078) 大于现在位置时正转，小于时反转。
- 运转速度会由启动速度 V_{BIAS} 开始加速至期望之设定速度 $V(I)$ 稳态运行，直到脉冲产生送出与位置寄存器 P(I) 相同的脉冲个数后，此时会由设定速度 $V(I)$ ，再次地加速/减速至设定速度 $V(II)$ 后，同样稳态地以设定速度 $V(II)$ 运行，一直快到达位置寄存器 P(II) 之内容值时，减速至启动速度而后停，其间总共发送产生出 P(I)+P(II) 个脉冲数。
- 应用的寄存器有 V_{BIAS} : D1824 (D1904, D1984, D2054) 、 $V(I)$: D1840 (D1920, D2000, D2080) 、 $V(II)$: D1844 (D1924, D2004, D2084) 、 V_{MAX} : D1822 (D1902, D1982, D2062) 、P(I): D1838 (D1918, D1998, D2078) 、P(II): D1842 (D1922, D2002, D2082); T_{ACC} : D1836 (D1916, D1996, D2076) 及 T_{DEC} : D1837 (D1917, D1997, D2077) 等寄存器。



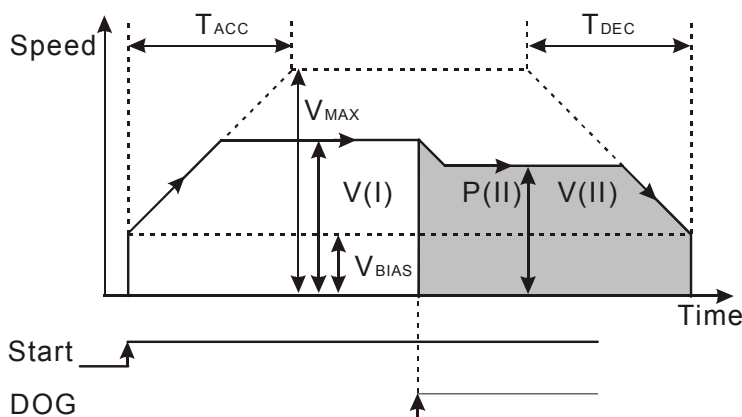
12. D1846 (D1926, D2006, D2086) 之 b11: 插入二段速定位运动模式启动

- 当 b[11]被触发，外部两段速定位运动命令后，且启动运转 (START On) 后，第 1 段位置定位期间，若外部近点 DOG 信号动作后，会立即开始第 2 段位置定位动作。开始由脉冲产生单元送出脉冲。
- 运转方向：相对坐标定位由设定位置寄存器 P(I) 内容值之符号位决定。绝对坐标定位由目标位置(I) P(I)： D1838 (D1918, D1998, D2078) 大于现在位置时正转，小于时反转。
- 而运转速度会由启动速度 V_{BIAS} 开始加速至期望之设定速度 $V(I)$ 稳态运行，于脉冲送出期间遇到 DOG 近点信号触发时，脉冲产生单元会再次地由目前行进速度加速/减速至设定速度 $V(II)$ 后，同样稳态地以设定速度 $V(II)$ 运行，于执行第二段速度时，其间外部 STOP 接点输入时，则会制

3 DVP- PM 各种装置功能

使脉冲产生单元立即中止脉冲输出。

- 应用的寄存器有 V_{BIAS} : D1824 (D1904, D1984, D2054) 、 $V(I)$: D1840 (D1920, D2000, D2080) 、 $V(II)$: D1844 (D1924, D2004, D2084) 、 V_{MAX} : D1822 (D1902, D1982, D2062) 、 $P(I)$: D1838 (D1918, D1998, D2078) 、 $P(II)$: D1842 (D1922, D2002, D2082); T_{ACC} : D1836 (D1916, D1996, D2076) 及 T_{DEC} : D1837 (D1917, D1997, D2077) 等寄存器。



13. D1846 (D1926, D2006) 之 b12: OX 启动设定

- $b[12]=1$, 启动 OX 程序; $b[12]=0$, 关闭 OX 程序

X 轴		Y 轴		Z 轴		工作模式
HW	LW	HW	LW	HW	LW	
	D1847		D1927		D2007	

[说明]

1. D1847 (D1927, D2007) 之 b2: CLR 信号输出模式。

- 当 $b[2]=0$ 时, 输出信号 CLR 为原点复归完成时, 输出给伺服驱动器信号 (时间长度约为 130ms), 作为伺服内部偏差计数器的清除信号。
- 当 $b[2]=1$ 时, CLR 输出点作为一般输出点, 状态由 $b[3]$ 控制 On/Off。

2. D1847 (D1927, D2007) 之 b3: CLR 输出 On/Off 控制。

- 当 $b[3]=0$ 时, 输出点 CLR 为 Off。
- 当 $b[3]=1$ 时, 输出点 CLR 为 On。

3. D1847 (D1927, D2007) 之 b4: CLR 极性设定。

- 当 $b[4]=0$ 时, CLR 极性为 a 接点。
- 当 $b[4]=1$ 时, CLR 极性为 b 接点。

4. D1847 (D1927, D2007) 之 b5: STOP 模式。
 - b[5]=0: 马达运转中, 碰到 STOP 信号输入时, 马达减速停, 再度下达运动命令时, 马达会忽略之前未完成的距离, 立即继续下一步骤的距离。
 - b[5]=1: 马达运转中, 碰到 STOP 信号输入时, 马达减速停, 再度下达运动命令时, 马达会先移动之前未完成的距离, 再执行下一定位行程。

5. D1847 (D1927, D2007) 之 b6: 手摇轮 MPG 范围限制。
 - b[6]=0: 手摇轮 (MPG) 输入之脉冲输出的允许范围无限制。
 - b[6]=1: 手摇轮 (MPG) 输入之脉冲输出的允许范围, 被限制于 P(I) 与 P(II) 的范围内操作, 当超出范围频率波减速停止输出。

6. D1847 (D1927, D2007) 之 b7: LSP/LSN 停止模式。
 - b[7]=0: 马达运转中, 碰到 LSP/LSN 信号输入时, 马达减速停止。
 - b[7]=1: 马达运转中, 碰到 LSP/LSN 信号输入时, 马达立即停止。

7. D1847 (D1927, D2007) 之 b8~b10: MASK 选择设定。
 - MASK 选择设定 (1 段速运转、2 段速运转、1 段速插入运转、2 段速插入运转) K0 (000) 或其它数值: 无 MASK 功能。
 - b[10~8]=K1 (001) : MASK 以输入端 ΦA_{\pm} 之上升缘触发。
 - b[10~8]=K2 (010) : MASK 以输入端 ΦA_{\pm} 之下降缘触发。
 - b[10~8]=K3 (011) : MASK 以输入端 ΦB_{\pm} 之上升缘触发。
 - b[10~8]=K4 (100) : MASK 以输入端 ΦB_{\pm} 之下降缘触发。

8. D1847 (D1927, D2007) 之 b15: 回归出厂设定
 - b[15]=1: 所有参数回归出厂值设定。

X 轴		Y 轴		Z 轴		A 轴		现在位置 CP(PLS)
HW	LW	HW	LW	HW	LW	HW	LW	
D1849	D1848	D1929	D1928	D2009	D2008	D2089	D2088	

[说明]

1. 显示范围: -2,147,483,648~+2,147,483,647。
2. 现在位置, 以脉冲值 (PLS) 表示, 由 D1816 (D1896, D1976, D2056) 之 b0, b1 作设定。原点复归

3 DVP- PM 各种装置功能

完成时，D1834 (D1914, D1994, D2074) 原点位置定义 HP，会填入现在位置 CP (PLS) 数据中。

X 轴		Y 轴		Z 轴		A 轴		现在速度 CS(PPS)
HW	LW	HW	LW	HW	LW	HW	LW	
D1851	D1850	D1931	D1930	D2011	D2010	D2091	D2090	

[说明]

1. 显示范围：0~+2,147,483,647。
2. 现在速度，以 PPS 表示。

X 轴		Y 轴		Z 轴		A 轴		现在位置 CP(unit)
HW	LW	HW	LW	HW	LW	HW	LW	
D1853	D1852	D1933	D1932	D2013	D2012	D2093	D2092	

[说明]

1. 显示范围：-2,147,483,648~+2,147,483,647。
2. 设定单位依照 D1816 (D1896, D1976, D2056) 之 b0, b1 单位系设定变化。原点复归完成时，D1834 (D1914, D1994, D2074) 原点位置定义 HP 填入 D1852 (D1932, D2012, D2092) 现在位置数据中。

X 轴		Y 轴		Z 轴		A 轴		现在速度 CS(unit)
HW	LW	HW	LW	HW	LW	HW	LW	
D1855	D1854	D1935	D1934	D2015	D2014	D2095	D2094	

[说明]

1. 显示范围：0~+2,147,483,647。
2. 设定单位依照 D1816 (D1896, D1976, D2056) 之 b0, b1 单位系设定来变化。

X 轴		Y 轴		Z 轴		A 轴		执行状态
HW	LW	HW	LW	HW	LW	HW	LW	
	D1856		D1936		D2016		D2096	

[说明]

bit#	D1856 (D1936, D2016)	bit#	D1856 (D1936, D2016)
0	正向脉冲输出中	8	未定义
1	反向脉冲输出中	9	未定义
2	行程动作中	10	未定义
3	错误产生	11	未定义
4	行程暂停中	12	未定义
5	正向 MPG 输入	13	未定义
6	反向 MPG 输入	14	未定义
7	未定义	15	未定义

X 轴		Y 轴		Z 轴		A 轴		错误编号
HW	LW	HW	LW	HW	LW	HW	LW	
	D1857		D1937		D2017		D2097	

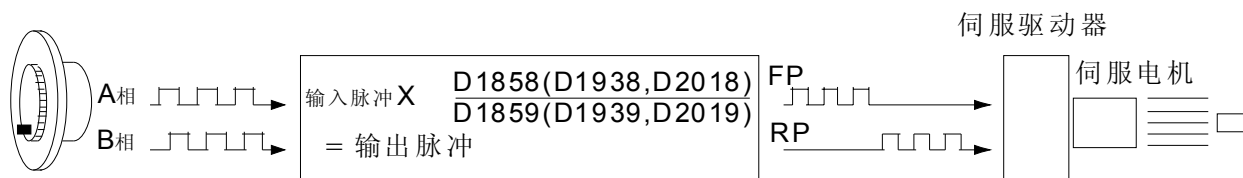
[说明]

详细内容请参考第 14 章侦错码原因对照表。

X 轴		Y 轴		Z 轴		A 轴		电子齿轮比
HW	LW	HW	LW	HW	LW	HW	LW	
	D1858		D1938		D2018		D2098	电子齿轮 (分子)
	D1859		D1939		D2019		D2099	电子齿轮 (分母)

[说明]

1. 当运转命令 D1846 (D1926, D2006, D2086) 之 b5 设为 On, 表示为手摇轮脉冲输入工作模式启动。
2. 以手摇轮产生 A/B 相脉冲输入至 ΦA 及 ΦB , FP/ RP 输出与输入脉冲关系如下图所示:



3. 在运作期间, 若 LSP 或 LSN 被启动, 则输出立刻停止, 若是 LSP 启动, 则正向脉冲被禁止, 反相脉冲允许, 若为 LSN 启动, 则反向脉冲被禁止, 正向脉冲允许。
4. 输出运行速度为手摇轮产生的脉冲输入频率与电子齿轮 (D1858 (D1938, D2018, D2098), D1859 (D1939, D2019, D2099)) 成比例的关系。

3 DVP-PM 各种装置功能

X 轴		Y 轴		Z 轴		A 轴		手摇轮输入频率
HW	LW	HW	LW	HW	LW	HW	LW	
D1861	D1860	D1941	D1940	D2021	D2020	D2101	D2100	

[说明]

1. 由手摇轮输入的脉冲频率不受电子齿轮比影响。

X 轴		Y 轴		Z 轴		A 轴		累计手摇轮输入脉冲个数
HW	LW	HW	LW	HW	LW	HW	LW	
D1863	D1862	D1943	D1942	D2023	D2022	D2103	D2102	

[说明]

1. 计数由手摇轮输入的脉冲个数，正转脉冲输入，该计数值为“加”动作，若为反转脉冲输入，则该计数值为“减”动作。
2. 计数值不受电子齿轮比 (D1858 (D1938, D2018, D2098), D1859 (D1939, D2019, D2099)) 设定影响。






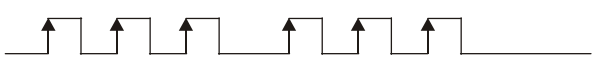
X 轴		Y 轴		Z 轴		A 轴		手摇轮输入响应速度
HW	LW	HW	LW	HW	LW	HW	LW	
	D1864		D1944		D2024		D2104	

[说明]

1. 响应速度设定愈快，表示命令脉冲输出与手摇轮脉冲输入时序越同步。
2. 响应速度设定愈慢，表示命令脉冲输出反应落后于手摇轮脉冲输入时序。

设定值	响 应 速 度
≥5	4ms (初始值)
4	32 ms
3	108 ms
2	256 ms
1 或 0	500 ms

3. D1864 (D1944, D2024) 之 b8, b9: 手摇轮输入脉冲型式设定

b8	b9	输入脉冲型式 (正逻辑)	说明
0	0	FP 正轉脈波  RP 反轉脈波 	双脉冲
0	1	FP 脈波  RP 方向(DIR) 	单脉冲
1	0	FP A相脈波 	A/B 相脉冲
1	1	RP B相脈波 	

3.12.2 手动运动模式介绍

1. DVP-PM 定位模块共有 10 种运动模式：

- | | |
|-----------------|--------------------|
| 1. 机械原点复归操作 | 6. 连续两段速定位运动模式 |
| 2. 寸动运动模式操作 | 7. 插入二段速定位运动模式 |
| 3. 单段速定位运动模式 | 8. 变速度运动模式 |
| 4. 插入单段速定位运动模式 | 9. 手摇轮 (MPG) 输入操作 |
| 5. 单段速外部触发运动模式* | 10. 周期/非周期式电子凸轮模式* |

备注*:仅 20PM 支援

2. 当多个工作模式同时被启动时，其处理优先级如下：

- | | |
|---------------------|-------------------|
| 1. STOP 停止 | 7. 单段速定位运动模式 |
| 2. 机械原点复归操作 | 8. 插入单段速定位运动模式 |
| 3. 寸动 (JOG+) 运动模式操作 | 9. 单段速外部触发运动模式 |
| 4. 寸动 (JOG-) 运动模式操作 | 10. 连续两段速定位运动模式 |
| 5. 手摇轮 (MPG) 输入操作 | 11. 插入二段速定位运动模式 |
| 6. 变速度运动模式 | 12. 周期/非周期式电子凸轮模式 |

当其中一个工作模式执行中，另一工作模式又被启动时，DVP-PM 仍维持原来工作模式操作。

3. 两种脉冲加速曲线：

- | | |
|---------|---------|
| 1. 梯形曲线 | 2. S 曲线 |
|---------|---------|

3 DVP-PM 各种装置功能

3.12.3 手动运动模式对应使用位置、速度控制寄存器一览表：

运动模式对应使用寄存器								参数名称 代码	操作模式									
X轴		Y轴		Z轴		A轴			寸动 JOG	原点 复归	单段 速度 定位	插入 单段 速度 定位	单段 速外 部触 发	连续 两段 速度 定位	插入 二段 速度 定位	变速 度	手摇 轮(MPG) 输入	
HW	LW	HW	LW	HW	LW	HW	LW											
D1819	D1818	D1899	D1898	D1979	D1978	D2059	D2058	马达转一圈所 须脉冲数 A	若单位系 (D1816, D1896, D1976_b0, b1) 选择为马达单位, 则不须设定 若为机械单位或复合单位系, 则必须作设定									
D1821	D1820	D1901	D1900	D1981	D1980	D2061	D2060	马达转一圈移 动距离 B										
	D1816		D1896		D1976		D2056	参数设定	○	○	○	○	○	○	○	○	○	
D1823	D1822	D1903	D1902	D1983	D1982	D2063	D2062	最高速度 V _{MAX}	○	○	○	○	○	○	○	○	○	
D1825	D1824	D1905	D1904	D1985	D1984	D2065	D2064	启动速度 V _{BIAS}	○	○	○	○	○	○	○	○	○	
D1827	D1826	D1907	D1906	D1987	D1986	D2067	D2066	寸动 JOG 速度 V _{JOG}	○	-	-	-	-	-	-	-	-	
D1829	D1828	D1909	D1908	D1989	D1988	D2069	D2068	原点复归速度 V _{RT}										
D1831	D1830	D1911	D1910	D1991	D1990	D2071	D2070	原点复归减速 速度 V _{CR}										
	D1832		D1912		D1992		D2072	原点复归之零 点 (PG0) 信号 数 N	-	○	-	-	-	-	-	-	-	
	D1833		D1913		D1993		D2073	原点复归之脉 冲信号数 P										
D1835	D1834	D1915	D1914	D1995	D1994	D2075	D2074	原点位置定义 HP										
	D1836		D1916		D1996		D2076	加速时间 T _{ACC}	○	○	○	○	○	○	○	○	○	-
	D1837		D1917		D1997		D2077	减速时间 T _{DEC}	○	○	○	○	○	○	○	○	○	-
D1839	D1838	D1919	D1918	D1999	D1998	D2079	D2078	目标位置(I) P(I)	-	-	○	○	○	○	○	○	-	○
D1841	D1840	D1921	D1920	D2001	D2000	D2081	D2080	运转速度(I) V(I)	-	-	○	○	○	○	○	○	○	-
D1843	D1842	D1923	D1922	D2003	D2002	D2083	D2082	目标位置(II) P(II)	-	-	-	-	-	○	○	-	○	
D1845	D1844	D1925	D1924	D2005	D2004	D2085	D2084	运转速度(II) V(II)	-	-	-	-	-	○	○	-	-	
	D1846		D1926		D2006		D2086	运转命令	○	○	○	○	○	○	○	○	○	
	D1847		D1927		D2007		D2087	工作模式	○	○	○	○	○	○	○	○	○	
D1849	D1848	D1929	D1928	D2009	D2008	D2089	D2088	现在位置 CP (PLS)	○	○	○	○	○	○	○	○	○	
D1851	D1850	D1931	D1930	D2011	D2010	D2091	D2090	现在速度 CS (PPS)	○	○	○	○	○	○	○	○	○	
D1853	D1852	D1933	D1932	D2013	D2012	D2093	D2092	现在位置 CP (unit)	○	○	○	○	○	○	○	○	○	
D1855	D1854	D1935	D1934	D2015	D2014	D2095	D2094	现在速度 CS (unit)	○	○	○	○	○	○	○	○	○	
	D1858		D1938		D2018		D2098	电子齿轮分子	-	-	-	-	-	-	-	-	○	

运动模式对应使用寄存器								参数名称 代码	操作模式								
X 轴		Y 轴		Z 轴		A 轴			寸动 JOG	原点 复归	单段 速度 定位	插入 单段 速度 定位	单段 速外 部触 发	连续 两段 速度 定位	插入 二 段 速 定 位	变 速 度	手 摇 轮 (MPG) 输 入
HW	LW	HW	LW	HW	LW	HW	LW										
	D1859		D1939		D2019		D2099	电子齿轮分母	-	-	-	-	-	-	-	-	◎
D1861	D1860	D1941	D1940	D2021	D2020	D2101	D2100	手摇轮输入频率	-	-	-	-	-	-	-	-	◎
D1863	D1862	D1943	D1942	D2023	D2022	D2103	D2102	累计手摇轮输入脉冲数	-	-	-	-	-	-	-	-	◎
	D1864		D1944		D2024		D2104	MPG 响应速度	-	-	-	-	-	-	-	-	◎
D1865								停止模式 (OX0-99)	-	-	-	-	-	-	-	-	-
D1867	D1866	D1947	D1946	D2027	D2026		D2106	电子原点	-	-	-	-	-	-	-	-	-
D1868								程序号码指定方式	-	-	-	-	-	-	-	-	-
D1869								OX 错误 STEP 位置	-	-	-	-	-	-	-	-	-
D1872								Ready 输出 High byte	◎	◎	◎	◎	◎	◎	◎	◎	◎
D1873								M Code 输出 High byte	-	-	-	-	-	-	-	-	-
D1874								M Code off 输入 X 起点编号	-	-	-	-	-	-	-	-	-
D1875		D1955						外部手动启动 (ZRN, MPG, JOG-, JOG+)	◎	-	-	-	-	-	-	-	◎

◎：表示动作于该操作模式时，相关之控制寄存器。

4.1 基本指令一览表

📖 一般指令

指令码	功能	操作数	执行速度(us)		STEP	页码
			20PM	10PM		
LD	载入 A 接点	X、Y、M、S、T、C	3.3	0.14	3	4-3
LDI	载入 B 接点	X、Y、M、S、T、C	3.3	0.14	3	4-3
AND	串联 A 接点	X、Y、M、S、T、C	3.3	0.14	3	4-3
ANI	串联 B 接点	X、Y、M、S、T、C	3.3	0.14	3	4-4
OR	并联 A 接点	X、Y、M、S、T、C	3.3	0.14	3	4-4
ORI	并联 B 接点	X、Y、M、S、T、C	3.3	0.14	3	4-5
ANB	串联回路方块	无	2.3	-	3	4-5
ORB	并联回路方块	无	2.3	-	3	4-5

📖 输出指令

指令码	功能	操作数	执行速度(us)		STEP	页码
			20PM	10PM		
OUT	驱动线圈	Y、M、S	7.3	-	3	4-6
SET	动作保持(On)	Y、M、S	5.6	-	3	4-6
RST	接点或寄存器清除	Y、M、S、T、C、D、V、	6.9	-	3	4-7

📖 定时器、计数器

API	指令码	功能	操作数	执行速度(us)		STEP	页码
				20PM	10PM		
96	TMR	16 位定时器	T-K 或 T-D	19	6	5	4-7
97	CNT	16 位计数器	C-K 或 C-D (16 位)	16	2.8	5	4-8
97	DCNT	32 位计数器	C-K 或 C-D (32 位)	16.5	2.8	6	4-8

📖 接点上升缘/下降缘检出指令

API	指令码	功能	操作数	执行速度(us)		STEP	页码
				20PM	10PM		
90	LDP	正缘检出动作开始	X、Y、M、S、T、C	12.3	0.4	3	4-9
91	LDF	负缘检出动作开始	X、Y、M、S、T、C	12.3	0.5	3	4-9
92	ANDP	正缘检出串联连接	X、Y、M、S、T、C	12.3	0.4	3	4-9
93	ANDF	负缘检出串联连接	X、Y、M、S、T、C	12.3	0.4	3	4-10
94	ORP	正缘检出并联连接	X、Y、M、S、T、C	12.6	0.5	3	4-10
95	ORF	负缘检出并联连接	X、Y、M、S、T、C	12.6	0.4	3	4-11

4 基本顺序指令

上下微分输出指令

API	指令码	功能	操作数	执行速度(us)		STEP	页码
				20PM	10PM		
89	PLS	上微分输出	Y、M	20.7	0.2	3	4-11
99	PLF	下微分输出	Y、M	20.9	0.3	3	4-12

其它指令

指令码	功能	操作数	执行速度(us)		STEP	页码
			20PM	10PM		
P	指标	P0~P255	-	-	1	4-12

4.2 基本指令说明

指令	功 能							适用机种	
LD	载入 A 接点							20PM	10PM
								✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	✓	—
	✓	✓	✓	✓	✓	✓	—		

指令说明

- ◆ LD 指令用于左母线开始的 A 接点或一个接点回路块开始的 A 接点，它的作用是把当前内容保存，同时把取来的接点状态存入累积寄存器内。

程式范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
AND	X1	串联 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

指令	功 能							适用机种	
LDI	载入 B 接点							20PM	10PM
								✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	✓	—
	✓	✓	✓	✓	✓	✓	—		

指令说明

- ◆ LDI 指令用于左母线开始的 B 接点或一个接点回路块开始的 B 接点，它的作用是把当前内容保存，同时把取来的接点状态存入累积寄存器内。

程式范例

梯形图：



指令码：

说明：

LDI	X0	载入 X0 之 B 接点
AND	X1	串联 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

指令	功 能							适用机种	
AND	串联 A 接点							20PM	10PM
								✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	✓	—
	✓	✓	✓	✓	✓	✓	—		

指令说明

- ◆ AND 指令用于 A 接点的串联连接，先读取目前所指定串联接点的状态再与接点之前逻辑运算结果作“及”(AND)的运算，并将结果存入累积寄存器内。

4 基本顺序指令

程式范例

梯形图:



指令码:

说明:

LDI	X1	载入 X1 之 B 接点
AND	X0	串联 X0 之 A 接点
OUT	Y1	驱动 Y1 线圈

指令	功能						适用机种	
ANI	串联 B 接点						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	—	

指令说明

- ◆ ANI 指令用于 B 接点的串联连接，它的作用是先读取目前所指定串联接点的状态再与接点之前逻辑运算结果作“及”(AND) 的运算，并将结果存入累积寄存器内。

程式范例

梯形图:



指令码:

说明:

LD	X1	载入 X1 之 A 接点
ANI	X0	串联 X0 之 B 接点
OUT	Y1	驱动 Y1 线圈

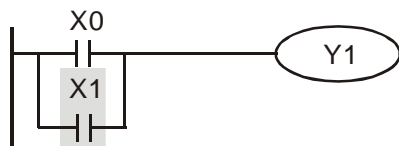
指令	功能						适用机种	
OR	并联 A 接点						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	—	

指令说明

- ◆ OR 指令用于 A 接点的并联连接，它的作用是先读取目前所指定串联接点的状态再与接点之前逻辑运算结果作“或”(OR) 的运算，并将结果存入累积寄存器内。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
OR	X1	并联 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

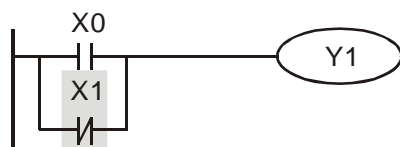
指令	功 能							适用机种	
	ORI	并联 B 接点							20PM
✓									✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999		
	✓	✓	✓	✓	✓	✓	—		

指令说明

- ◆ ORI 指令用于 B 接点的并联连接，它的作用是先读取目前所指定串联接点的状态再与接点之前逻辑运算结果作“或”(OR) 的运算，并将结果存入累积寄存器内。

程式范例

梯形图:



指令码:

说明:

```
LD    X0    载入 X0 之 A 接点
ORI  X1    并联 X1 之 B 接点
OUT   Y1    驱动 Y1 线圈
```

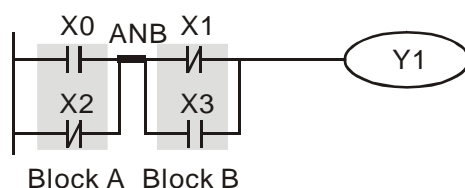
指令	功 能							适用机种	
	ANB	串联回路方块							20PM
✓									✓
操作数	无								

指令说明

- ◆ ANB 是将前一保存的逻辑结果与目前累积寄存器的内容作“及”(AND) 的运算。

程式范例

梯形图:



指令码:

说明:

```
LD    X0    载入 X0 之 A 接点
ORI   X2    并联 X2 之 B 接点
LDI   X1    载入 X1 之 B 接点
OR    X3    并联 X3 之 A 接点
ANB                串联回路方块
OUT   Y1    驱动 Y1 线圈
```

指令	功 能							适用机种	
	ORB	并联回路方块							20PM
✓									✓
操作数	无								

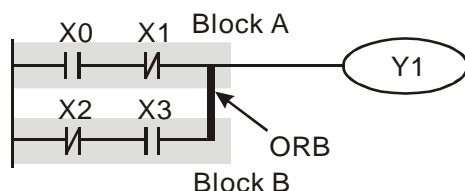
4 基本顺序指令

指令说明

- ◆ ORB 是将前一保存的逻辑结果与目前累积寄存器的内容作“或” (OR) 的运算。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
ANI	X1	串联 X1 之 B 接点
LDI	X2	载入 X2 之 B 接点
AND	X3	串联 X3 之 A 接点
ORB		并联回路方块
OUT	Y1	驱动 Y1 线圈

指令	功能						适用机种	
OUT	驱动线圈						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	—	✓	✓	✓	—	—	—	

指令说明

- ◆ 将 OUT 指令之前的逻辑运算结果输出至指定的组件。
- ◆ 线圈接点动作:

运算结果	OUT 指令		
	线圈	接点	
		A 接点 (常开)	B 接点 (常闭)
FALSE	Off	不导通	导通
TRUE	On	导通	不导通

程式范例

梯形图:



指令码:

说明:

LDI	X0	载入 X0 之 B 接点
AND	X1	串联 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

指令	功能						适用机种	
SET	动作保持 (On)						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	—	✓	✓	✓	—	—	—	

指令说明

- ◆ 当 SET 指令被驱动, 其指定的组件被设定为 On, 且被设定的组件会维持 On, 不管 SET 指令是否仍被驱动。可利用 RST 指令将该组件设为 Off。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
ANI	Y0	串联 Y0 之 B 接点
SET	Y1	Y1 动作保持 (On)

指令	功能						适用机种	
RST	接点或寄存器清除						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	—	✓	✓	✓	✓	✓	✓	

指令说明

- ◆ 当 RST 指令被驱动, 其指定的组件的动作如下:

元件	状态
S, Y, M	线圈及接点都会被设定为 Off。
T, C	目前计时或计数值会被设为 0, 且线圈及接点都会被设定为 Off。
D, V, Z	内容值会被设为 0。

- ◆ 若 RST 指令没有被执行, 其指定组件的状态保持不变。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
RST	Y5	Y5 接点清除

指令	功能		适用机种	
TMR	16 位定时器		20PM	10PM
			✓	✓
操作数	T-K	T0~T255, K0~K32,767		
	T-D	T0~T255, D0~D9,999		

指令说明

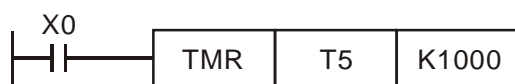
- ◆ 当 TMR 指令执行时, 其所指定的定时器线圈受电, 定时器开始计时, 当到达所指定的定时值 (计时值 \geq 设定值), 其接点动作如下:

NO(Normally Open) 接点	闭合
NC(Normally Close) 接点	开路

4 基本顺序指令

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
TMR	T5 K1000	T5 定时器 设定值为 K1000

补充说明

- ◆ 定时器操作数 T 使用范围请参考各系列机种功能规格表。

指令	功 能		适用机种	
	CNT	16 位计数器		20PM
			✓	✓
操作数	C-K	C0~C199, K0~K32,767		
	C-D	C0~C199, D0~D9,999		

指令说明

- ◆ 当 CNT 指令由 Off→On 执行, 表示所指定的计数器线圈由失电→受电, 则该计数器计数值加 1, 当计数到达所指定的定数值 (计数值 = 设定值), 其接点动作如下:

NO(Normally Open) 接点	闭合
NC(Normally Close) 接点	开路

- ◆ 当计数到达之后, 若再有计数脉波输入, 其接点及计数值均保持不变, 若要重新计数或作清除的动作, 请利用 RST 指令。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
CNT	C20 K100	C20 计数器设定 值为 K100

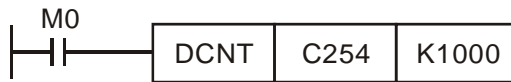
指令	功 能		适用机种	
	DCNT	32 位计数器		20PM
			✓	✓
操作数	C-K	C200、C204、C208、C212、C216、C220、C221~C255, K-2,147,483,648~K2,147,483,647		
	C-D	C200、C204、C208、C212、C216、C220、C221~C255, D0~D9,999		

指令说明

- ◆ DCNT 为 32 位计数器 C200 至 C255 之启动指令。
- ◆ 一般用加减计数器 C200~C255，当 DCNT 指令由 Off→On 时，计数器之现在值将执行上数 (加一) 的动作或下数 (减一) 的动作，依特 M1200~1234 的设定模式。

程式范例

梯形图：



指令码：

说明：

LD	M0	载入 M0 之 A 接点
DCNT	C254 K1000	C254 计数器设定值为 K1000

指令	功 能						适用机种	
LDP	正缘检出动作开始						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	—	

指令说明

- ◆ LDP 指令用法上与 LD 相同，但动作不同，它的作用是指当前内容保存，同时把取来的接点上升缘检出状态存入累积寄存器内。

程式范例

梯形图：



指令码：

说明：

LDP	X0	X0 正缘检出动作开始
AND	X1	串联 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

补充说明

- ◆ 各操作数使用范围请参考各系列机种功能规格表。
- ◆ 若 DVP-PM 电源开启前，指定上升缘接点的状态为 On，则电源开启后该上升缘接点为 TRUE。

指令	功 能						适用机种	
LDF	负缘检出动作开始						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	—	

指令说明

- ◆ LDF 指令用法上与 LD 相同，但动作不同，它的作用是指当前内容保存，同时把取来的接点下降缘检出状态存入累积寄存器内。

4 基本顺序指令

程式范例

梯形图:



指令码:

说明:

LDF	X0	X0 负缘检出动作开始
AND	X1	串联 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

指令	功能						适用機種	
ANDP	正缘检出串联连接						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	—	

指令说明

◆ ANDP 指令用于接点上升缘检出的串联连接。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
ANDP	X1	X1 正缘检出串联连接
OUT	Y1	驱动 Y1 线圈

指令	功能						适用機種	
ANDF	负缘检出串联连接						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	—	

指令说明

◆ ANDF 指令用于接点下降缘检出的串联连接。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
ANDF	X1	X1 负缘检出串联连接
OUT	Y1	驱动 Y1 线圈

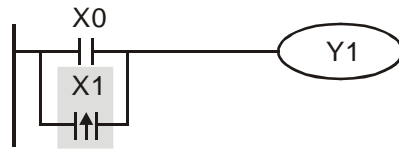
指令	功能						适用機種	
ORP	正缘检出并联连接						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	—	

指令说明

◆ ORP 指令用于接点上升缘检出的并联连接。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
ORP	X1	X1 正缘检出并联连接
OUT	Y1	驱动 Y1 线圈

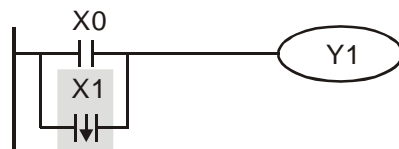
指令	功 能						适用机种	
ORF	负缘检出并联连接						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	—	

指令说明

◆ ORF 指令用于接点下降缘检出的并联连接。

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
ORF	X1	X1 负缘检出并联连接
OUT	Y1	驱动 Y1 线圈

指令	功 能						适用机种	
PLS	上微分输出						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	—	✓	✓	—	—	—	—	

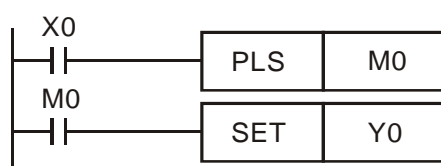
指令说明

◆ 上微分输出指令。当 X0=Off→On (正缘触发) 时 PLS 指令被执行, M0 送出一脉冲, 脉冲长度为一次扫描时间。

4 基本顺序指令

程式范例

梯形图:



指令码:

说明:

LD	X0	载入 X0 之 A 接点
PLS	M0	M0 上微分输出
LD	M0	载入 M0 之 A 接点
SET	Y0	Y0 动作保持 (On)

时序图:



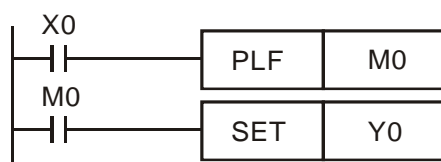
指令	功能						适用机种	
PLF	下微分输出						20PM	10PM
							✓	✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	—	✓	✓	—	—	—	—	

指令说明

- ◆ 下微分输出指令。当 X0= On→Off (负缘触发) 时 PLF 指令被执行, M0 送出一脉冲, 脉冲长度为一次扫描时间。

程式范例

梯形图:

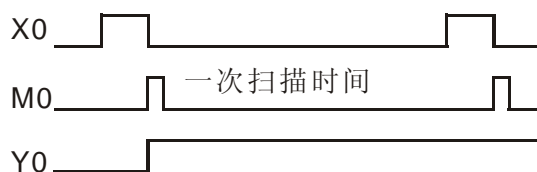


指令码:

说明:

LD	X0	载入 X0 之 A 接点
PLF	M0	M0 下微分输出
LD	M0	载入 M0 之 A 接点
SET	Y0	Y0 动作保持 (On)

时序图:



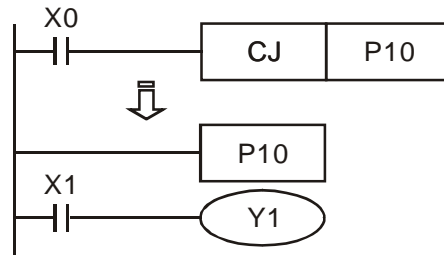
指令	功能		适用机种	
P	指标		20PM	10PM
			✓	✓
操作数	P0~P255			

指令说明

指标 P 用于 API 00 CJ、API 01 CALL、API 256 CJN、API 257 JMP 使用不需从编号 0 开始，但是编号不能重复使用，否则会发生不可预期的错误。

程式范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
CJ	P10	跳跃指令 CJ 到 P10
:		
P10		指标 P10
LD	X1	载入 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

5.1 应用指令一览表

分类	API	指令码		P 指令	功能	支持机种		STEPS		页码
		16 位	32 位			20PM	10PM	16 位	32 位	
回路控制	00	CJ	-	✓	条件跳跃	✓	✓	3	-	5-16
	01	CALL	-	✓	呼叫子程序	✓	✓	3	-	5-19
	02	SRET	-	-	子程序结束	✓	✓	1	-	5-19
	07	WDT	-	✓	逾时监视定时器	✓	✓	1	-	5-21
	08	RPT	-	-	循环电路起始 (只能一层)	✓	✓	3	-	5-22
	09	RPE	-	-	循环电路结束	✓	✓	1	-	5-22
传送比较	10	CMP	DCMP	✓	比较设定输出	✓	✓	7	9	5-24
	11	ZCP	DZCP	✓	区域比较	✓	✓	9	12	5-25
	12	MOV	DMOV	✓	数据移动	✓	✓	5	6	5-26
	13	SMOV	-	✓	位数移动	✓	✓	11	-	5-27
	14	CML	DCML	✓	反转传送	✓	✓	5	9	5-30
	15	BMOV	-	✓	全部传送	✓	✓	7	-	5-31
	16	FMOV	DFMOV	✓	多点移动	✓	✓	7	13	5-33
	17	XCH	DXCH	✓	数据的交换	✓	✓	5	9	5-34
	18	BCD	DBCD	✓	BIN→BCD 变换	✓	✓	5	5	5-35
19	BIN	DBIN	✓	BCD→BIN 变换	✓	✓	5	5	5-36	
四则逻辑运算	20	ADD	DADD	✓	BIN 加法	✓	✓	7	9	5-37
	21	SUB	DSUB	✓	BIN 减法	✓	✓	7	9	5-39
	22	MUL	DMUL	✓	BIN 乘法	✓	✓	7	9	5-41
	23	DIV	DDIV	✓	BIN 除法	✓	✓	7	9	5-42
	24	INC	DINC	✓	BIN 加一	✓	✓	3	3	5-43
	25	DEC	DDEC	✓	BIN 减一	✓	✓	3	3	5-44
	26	WAND	DWAND	✓	逻辑及 (AND) 运算	✓	✓	7	9	5-45
	27	WOR	DWOR	✓	逻辑或 (OR) 运算	✓	✓	7	9	5-46
	28	WXOR	DWXOR	✓	逻辑互斥或 (XOR) 运算	✓	✓	7	9	5-47
29	NEG	DNEG	✓	取负数 (取 2 的补码)	✓	✓	3	3	5-48	
旋转位移	30	ROR	DROR	✓	右旋转	✓	✓	5	9	5-50
	31	ROL	DROR	✓	左旋转	✓	✓	5	9	5-51
	32	RCR	DRCR	✓	附进位旗标右旋转	✓	✓	5	9	5-52
	33	RCL	DRCL	✓	附进位旗标左旋转	✓	✓	5	9	5-53
	34	SFTR	-	✓	位右移	✓	✓	9	-	5-54
	35	SFTL	-	✓	位左移	✓	✓	9	-	5-55
	36	WSFR	-	✓	寄存器右移	✓	✓	9	-	5-56
	37	WSFL	-	✓	寄存器左移	✓	✓	9	-	5-58
	38	SFWR	-	✓	位移写入	✓	✓	7	-	5-59
	39	SFRD	-	✓	位移读出	✓	✓	7	-	5-60

5 应用指令分类及基本使用

分类	API	指令码		P 指令	功能	支持机种		STEPS		页码
		16 位	32 位			20PM	10PM	16 位	32 位	
资料	40	ZRST	-	✓	区域清除	✓	✓	5	-	5-61
	41	DECO	-	✓	译码器	✓	✓	7	-	5-62
	42	ENCO	-	✓	编码器	✓	✓	7	-	5-64
	43	SUM	DSUM	✓	On 位数量	✓	✓	5	9	5-66
	44	BON	DBON	✓	On 位判定	✓	✓	7	13	5-67
	45	MEAN	DMEAN	✓	平均值	✓	✓	7	13	5-68
	46	ANS	-	-	警报点输出	✓	✓	7	-	5-69
	47	ANR	-	✓	警报点复归	✓	✓	1	-	5-70
	48	SQR	DSQR	✓	BIN 开平方根	✓	✓	5	9	5-72
	49	-	DFLT	✓	BIN 整数→二进制浮点数变换	✓	✓	-	6	5-73
高速处理	50	REF	-	✓	I/O 更新处理	✓	✓	5	-	5-75
便利指令	61	SER	DSER	✓	多点比较	✓	✓	9	17	5-77
	66	ALT	-	✓	On/Off 交替	✓	✓	3	-	5-79
	67	RAMP	DRAMP	-	倾斜信号	✓	✓	9	17	5-80
	69	SORT	DSORT	-	数据排序	✓	✓	11	21	5-82
I/O	78	FROM	DFROM	✓	I/O 模块 CR 数据读出	✓	✓	9	12	5-84
	79	TO	DTO	✓	I/O 模块 CR 数据写入	✓	✓	9	13	5-85
	87	ABS	DABS	✓	绝对值	✓	✓	3	5	5-88
基本指令	89	PLS	-	-	上微分输出	✓	✓	3	-	4-11
	90	LDP	-	-	正缘检出动作开始	✓	✓	3	-	4-9
	91	LDF	-	-	负缘检出动作开始	✓	✓	3	-	4-9
	92	ANDP	-	-	正缘检出串联连接	✓	✓	3	-	4-9
	93	ANDF	-	-	负缘检出串联连接	✓	✓	3	-	4-10
	94	ORP	-	-	正缘检出并联连接	✓	✓	3	-	4-10
	95	ORF	-	-	负缘检出并联连接	✓	✓	3	-	4-11
	96	TMR	-	-	定时器	✓	✓	5	-	4-7
	97	CNT	DCNT	-	计数器	✓	✓	5	6	4-8
	99	PLF	-	-	下微分输出	✓	✓	3	-	4-12
通讯	100	MODRD	-	-	MODBUS 数据读取	✓	✓	7	-	5-89
	101	MODW	-	-	MODBUS 资料写入	✓	✓	7	-	5-93
浮点运算	110	-	DECMP	✓	二进制浮点数比较	✓	✓	7	9	5-98
	111	-	DEZCP	✓	二进制浮点数区域比较	✓	✓	9	12	5-99
	112	-	DMOV	✓	浮点数值数据移动	✓	✓	-	9	5-100
	116	-	DRAD	✓	角度→径度	✓	✓	-	6	5-101
	117	-	DDEG	✓	径度→角度	✓	✓	-	6	5-102
	120	-	DEADD	✓	二进制浮点数加法	✓	✓	7	9	5-103

分类	API	指令码		P 指令	功能	支持机种		STEPS		页码
		16 位	32 位			20PM	10PM	16 位	32 位	
	121	-	DESUB	✓	二进制浮点数减法	✓	✓	7	9	5-104
	122	-	DEMUL	✓	二进制浮点数乘法	✓	✓	7	9	5-105
	123	-	DEDIV	✓	二进制浮点数除法	✓	✓	7	9	5-106
	124	-	DEXP	✓	二进制浮点数取指数	✓	✓	-	6	5-107
	125	-	DLN	✓	二进制浮点数取自然对数	✓	✓	-	6	5-108
	126	-	DLOG	✓	二进制浮点数取对数	✓	✓	-	9	5-109
	127	-	DESQR	✓	二进制浮点数开平方根	✓	✓	5	6	5-110
	128	-	DPOW	✓	浮点数权值指令	✓	✓	-	9	5-111
	129	-	DINT	✓	二进制浮点数→BIN 整数变换	✓	✓	-	6	5-113
	130	-	DSIN	✓	二进制浮点数→SIN 运算	✓	✓	5	6	5-114
	131	-	DCOS	✓	二进制浮点数→COS 运算	✓	✓	5	6	5-116
	132	-	DTAN	✓	二进制浮点数→TAN 运算	✓	✓	5	6	5-118
	133	-	DASIN	✓	二进制浮点数→ASIN 运算	✓	✓	-	6	5-120
	134	-	DACOS	✓	二进制浮点数→ACOS 运算	✓	✓	-	6	5-121
	135	-	DATAN	✓	二进制浮点数→ATAN 运算	✓	✓	-	6	5-122
	136	-	DSINH	✓	二进制浮点数→SINH 运算	✓	✓	-	6	5-123
	137	-	DCOSH	✓	二进制浮点数→COSH 运算	✓	✓	-	6	5-124
	138	-	DTANH	✓	二进制浮点数→TANH 运算	✓	✓	-	6	5-125
	172	-	DADDR	✓	浮点数值加法	✓	✓	-	13	5-126
	173	-	DSUBR	✓	浮点数值减法	✓	✓	-	13	5-127
	174	-	DMULR	✓	浮点数值乘法	✓	✓	-	13	5-128
	175	-	DDIVR	✓	浮点数值除法	✓	✓	-	13	5-129
接点型态逻辑运算	215	LD&	DLD&	-	S1 & S2	✓	✓	5	7	5-130
	216	LD	DLD	-	S1 S2	✓	✓	5	7	5-130
	217	LD^	DLD^	-	S1 ^ S2	✓	✓	5	7	5-130
	218	AND&	DAND&	-	S1 & S2	✓	✓	5	7	5-131
	219	AND	DAND	-	S1 S2	✓	✓	5	7	5-131
	220	AND^	DAND^	-	S1 ^ S2	✓	✓	5	7	5-131
	221	OR&	DOR&	-	S1 & S2	✓	✓	5	7	5-132
222	OR	DOR	-	S1 S2	✓	✓	5	7	5-132	
223	OR^	DOR^	-	S1 ^ S2	✓	✓	5	7	5-132	
接点型态比较指令	224	LD=	DLD=	-	S1 = S2	✓	✓	5	7	5-133
	225	LD>	DLD>	-	S1 > S2	✓	✓	5	7	5-133
	226	LD<	DLD<	-	S1 < S2	✓	✓	5	7	5-133
	228	LD<>	DLD<>	-	S1 ≠ S2	✓	✓	5	7	5-133
	229	LD<=	DLD<=	-	S1 ≤ S2	✓	✓	5	7	5-133
	230	LD>=	DLD>=	-	S1 ≥ S2	✓	✓	5	7	5-133
232	AND=	DAND=	-	S1 = S2	✓	✓	5	7	5-134	

5 应用指令分类及基本使用

分类	API	指令码		P 指令	功能	支持机种		STEPS		页码
		16 位	32 位			20PM	10PM	16 位	32 位	
	233	AND>	DAND>	-	S1 > S2	✓	✓	5	7	5-134
	234	AND<	DAND<	-	S1 < S2	✓	✓	5	7	5-134
	236	AND<>	DAND<	-	S1 ≠ S2	✓	✓	5	7	5-134
	237	AND<=	DAND<	-	S1 ≦ S2	✓	✓	5	7	5-134
	238	AND>=	DAND>	-	S1 ≧ S2	✓	✓	5	7	5-134
	240	OR=	DOR=	-	S1 = S2	✓	✓	5	7	5-135
	241	OR>	DOR>	-	S1 > S2	✓	✓	5	7	5-135
	242	OR<	DOR<	-	S1 < S2	✓	✓	5	7	5-135
	244	OR<>	DOR<>	-	S1 ≠ S2	✓	✓	5	7	5-135
	245	OR<=	DOR<=	-	S1 ≦ S2	✓	✓	5	7	5-135
	246	OR>=	DOR>=	-	S1 ≧ S2	✓	✓	5	7	5-135
其它指令	147	SWAP	DSWAP	✓	上/下字节交换	✓	✓	3	5	5-136
	154	RAND	DRAND	✓	随机数值产生	✓	✓	7	13	5-137
	202	SCAL	-	✓	比例值运算	✓	✓	9	-	5-138
	203	SCLP	DSCLP	✓	参数型比例值运算	✓	✓	7	13	5-140
	256	CJN	-	✓	反条件跳跃	✓	✓	3	-	5-144
	257	JMP	-	-	无条件跳跃	✓	✓	3	-	5-145
	258	BRET	-	-	回母线	✓	✓	1	-	5-146
	259	MMOV	-	✓	16→32 位数值转换	✓	✓	6	-	5-147
260	RMOV	-	✓	32→16 位数值转换	✓	✓	6	-	5-148	

5.2 应用指令的组成

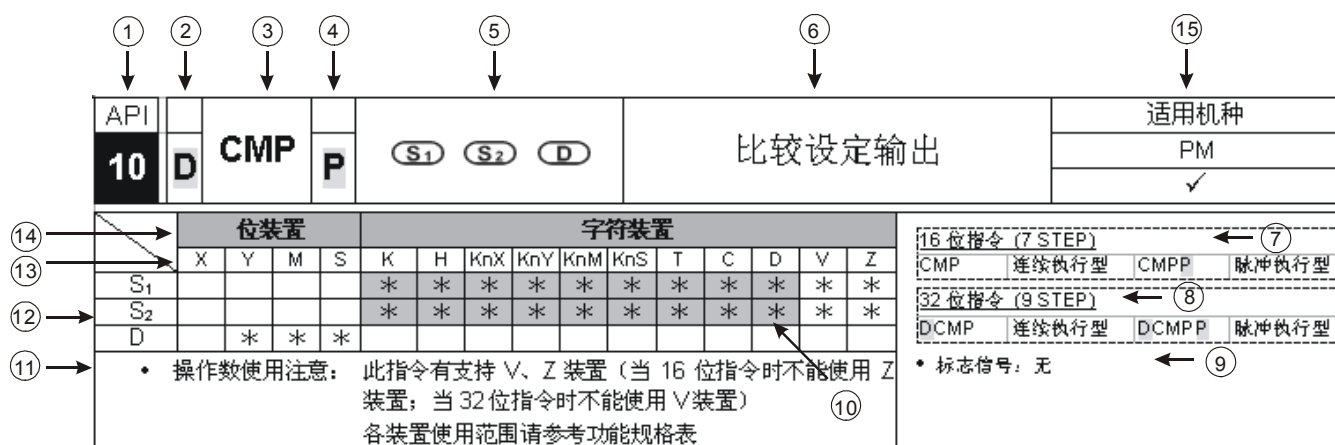
- ◆ 应用指令的结构可分为两部份：指令名及操作数

指令名： 表示指令执行功能

操作数： 表示该指令运算处理的装置

应用指令的指令部份通常占 1 个地址 (Step)，而 1 个操作数会根据 16 位指令或 32 位指令的不同占 2 或 3 个地址。

- ◆ 应用指令的格式说明



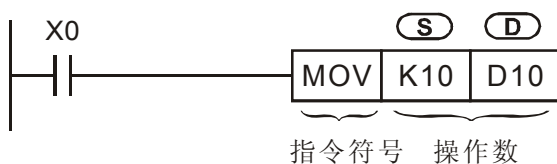
- ① 应用指令 API 编号号码
- ② 上方框表示具有 16 位指令。若为虚线表示此应用指令无 16 位指令
下方框表示具有 32 位指令，若为虚线表示此应用指令无 32 位指令。若有 32 位指令方框内以 **D** 表示 (例：API 10 **DCMP**)
- ③ 应用指令名
- ④ 上方框表示有些指令在应用上通常是使用脉冲指令，方框内以 **☺** 表示
下方框表示具有脉冲执行型指令，方框内以 **P** 表示 (例：API 12 **MOVP**)
- ⑤ 应用指令的操作数格式
- ⑥ 应用指令功能描述
- ⑦ 16 位指令所占的地址数，连续执行型指令名称与脉冲执行型指令名称
- ⑧ 32 位指令所占的地址数，连续执行型指令名称与脉冲执行型指令名称
- ⑨ 与该应用指令有相关之旗标信号
- ⑩ 符号 ‘*’ 标示者又含灰底色者，表示该装置可使用间接指定寄存器 V、Z 修饰
- ⑪ 操作数使用注意事项
- ⑫ 有符号 ‘*’ 标示者，表示该操作数可使用的装置
- ⑬ 装置名称
- ⑭ 装置型式
- ⑮ 适用该应用指令的系列機種

5 应用指令分类及基本使用

◆ 应用指令的输入

应用指令中有些指令仅有指令部份(指令名)构成,例如: BRET...或 SRET 等等,但是大部份都是指令部份再加上好几个操作数所组合而成。

DVP-PM 系列机种的应用指令是以指令号码 API 00~API 260 来指定的,同时每个指令均有其专用的名称符号,例如: API 12 的指令名称符号为 MOV (数据搬移)。若利用梯形图编辑软件 (PMSoft) 作该指令的输入,只需要直接打入该指令的名称 "MOV" 即可,若以程序书写器 (HPP03) 输入程序,则必须输入其 API 指令号码。而应用指令都会有不同的操作数指定,以 MOV 指令而言:



此指令是将 S 指定的操作数之值搬移至 D 所指定的目的操作数。其中:

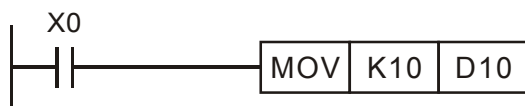
S	来源操作数;若来源操作数有一个以上,那么则以 S₁ , S₂ ...分别表示。
D	目的操作数;若目的操作数有一个以上,那么则以 D₁ , D₂ ...分别表示。
若操作数只可指定常数 K/H 或寄存器时,那么则以 m , m₁ , m₂ , n , n₁ , n₂ 表示。	

◆ 操作数长度(16 位指令或 32 位指令)

操作数的数值内容,其长度可分为 16 位及 32 位,因此部份指令处理不同长度的资料则分为 16 及 32 位的指令,用以区分 32 位的指令只需要在 16 位指令前加上 "D"来表示即可。

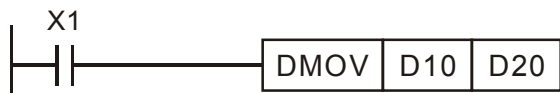
16 位 MOV 指令

当 X0=On 时, K10 被传送至 D10



32 位 DMOV 指令

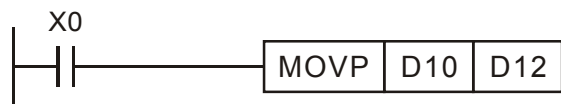
当 X1=On 时, (D11,D10) 的内容被传送至 (D21,D20)



◆ 连续执行型 / 脉冲执行型

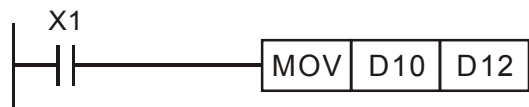
以指令的执行方式来说亦可分成「连续执行型」及「脉冲执行型」2 种。由于指令不被执行时,所需的执行时间比较短,因此程序中尽可能的使用脉冲执行型指令,可减少扫描周期。在指令后面加上 "P" 记号的指令,即为脉冲执行型指令。有些指令大部份的应用上都是使用脉冲执行型方式,如 INC、DEC 及位移相关等指令,因此于各指令的记号右上方均加上「Ⓢ」标志代表该指令通常是使用脉冲执行型。

脉冲执行型



当 X0 由 Off→On 变化时，MOVP 指令被执行一次，该次扫描指令不再被执行，因此称之为脉冲执行型指令

连续执行型



于 X1=On 的每次扫描周期，MOV 指令均被执行一次，因此称之为连续执行型指令

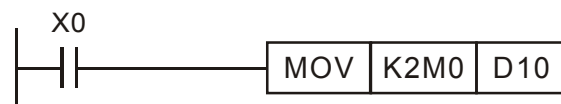
上图的两个条件接点 X0、X1=Off 时，指令不被执行，目的地操作数 D 的内容没有变化

◆ 操作数的指定对象

1. X、Y、M、S 等位装置也可以组合成字符装置使用，在应用指令里以 KnX、KnY、KnM、KnS 的型态来存放数值数据作运算。
2. 数据寄存器 D、定时器 T、计数器 C、间接指定寄存器 V、Z、都是一般操作数所指定的对象。
3. 数据寄存器一般为 16 位长度，也就是 1 个 D 寄存器，若指定 32 位长度的数据寄存器时，是指定连续号码的 2 个 D 寄存器。
4. 若 32 位指令的操作数指定 D0，则 (D1、D0) 所组成的 32 位数据寄存器被占用，D1 为上位 16 位，而 D0 为下位 16 位。定时器 T、16 位计数器 C0~C199 被使用的规则亦相同。
5. 32 位计数器 C200~C255 若是当数据寄存器来使用时，可指定为 32 位指令的操作数。

◆ 操作数数据格式

1. 装置 X、Y、M 及 S 只能作为单点的 On/Off，我们将之定义为位装置 (Bit device)。
2. 16 位 (或 32 位) 装置 T、C、D 及 V、Z 等寄存器，我们将之定义为字符装置 (Word device)。
3. 利用 Kn (其中 n = 1 表示 4 个位，所以 16 位可由 K1~K4，32 位可由 K1~K8) 加在位装置 X、Y、M 及 S 前，可将其定义为字符装置，因此可作字符装置的运算，例如 K2M0 即表示 8 位，M0~M7。



当 X0=On 时，将 M0~M7 的内容搬移 D10 的位 0~7，而位 8~15 则设为 0。

◆ 位装置组合成字符装置的数值数据处理

16 位指令		32 位指令	
16 位所指定的数值为：K-32,768~K32,767		32 位所指定的数值为：K-2,147,483,648~K2,147,483,647	
指定位数(K1~K4)的数值为：		指定位数(K1~K8)的数值为：	
K1 (4 个位)	0~15	K1 (4 个位)	0~15
K2 (8 个位)	0~255	K2 (8 个位)	0~255
K3 (12 个位)	0~4,095	K3 (12 个位)	0~4,095

5 应用指令分类及基本使用

16 位指令		32 位指令	
16 位所指定的数值为: K-32,768~K32,767		32 位所指定的数值为: K-2,147,483,648~K2,147,483,647	
指定位数(K1~K4)的数值为:		指定位数(K1~K8)的数值为:	
K4 (16 个位)	-32,768~+32,767	K4 (16 个位)	0~65,535
		K5 (20 个位)	0~1,048,575
		K6 (24 个位)	0~167,772,165
		K7 (28 个位)	0~268,435,455
		K8 (32 个位)	-2,147,483,648~+2,147,483,647

◆ 一般的旗标信号

(例) M1968: 零旗号 M1969: 借位旗号 M1970: 进位旗号

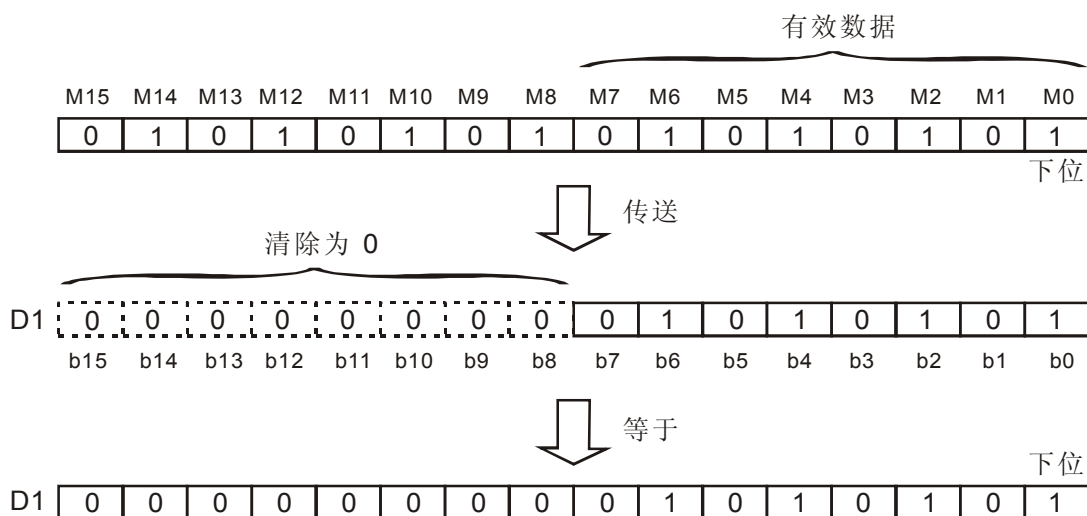
对应着应用指令运算结果, DVP-PM 系列有下列的旗标信号 (Flag)。

无论那一个旗标信号都会在指令被执行时,随着指令的运算结果作 On 或 Off 的变化,例如:于 O100~M102 的主控制程序区段中,如果使用 ADD/SUB/MUL/DIV 等数值运算指令,执行结果会影响 M1968~M1970 等旗标信号的状态。但是当指令不被执行时,旗标信号的 On/Off 状态被保持住。请注意上述旗标信号的动作,会与许多指令有关,请参阅个别指令说明。

5.3 应用指令对数值的处理方式

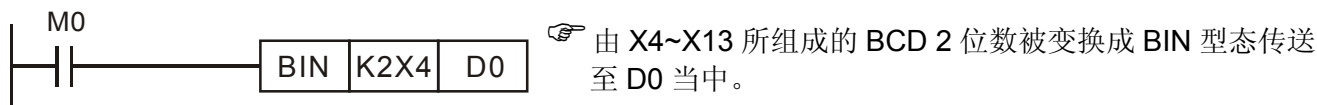
◆ X、Y、M、S 等只有 On/Off 变化的装置称之为位装置 (Bit Device), 而 T、C、D、V、Z 等专门用来存放数值的装置称之为字符装置 (Word Device)。虽然说位装置只能作 On/Off 变化,但是加上特定的宣告位装置也可以数值的型态被使用于应用指令的操作数当中,所谓的宣告是在位装置的前面加上位数,它是以 Kn 来表现。

◆ 16 位的数值可使用 K1~K4 而 32 位的数值则可使用 K1~K8。例如: K2M0 是由 M0~M7 所组成的 8 位数值。



◆ 将 K1M0、K2M0、K3M0 传送至 16 位的寄存器当中,不足的上位数据补 0。将 K1M0、K2M0、K3M0、K4M0、K5M0、K6M0、K7M0 传送至 32 位的寄存器也一样,不足的上位数据补 0。

- ◆ 16 位 (或 32 位) 的运算动作中, 操作数的内容若是指定 K1~K3 (或 K4~K7) 的位装置时, 不足的上位数据被视为 0, 因此一般都是被认定为正数的运算。



- ◆ 位装置的编号可自由指定, 但是 X 及 Y 的个位数号码请尽可能的指定 0。 (X0、X10、X20...Y0、Y10) M 及 S 的个位数号码尽可能的指定为 8 的倍数, 但仍以 0 为最恰当, 如 M0、M10、M20...等。
- ◆ 连续号码的指定

以数据寄存器 D 为例, D 的连续号码为 D0、D1、D2、D3、D4...

对于指定位数的位装置而言, 连续号码以下所示。

K1X0	K1X4	K1X10	K1X14.....
K2Y0	K2Y10	K2Y20	Y2X30.....
K3M0	K3M12	K3M24	K3M36.....
K4S0	K4S16	K4S32	K4S48.....

因此位装置号码如上, 请勿跳号以免造成混乱。此外, 如果将 K4Y0 使用于 32 位的运算当中, 上位 16 位被视为 0。32 位的数据请使用 K8Y0。

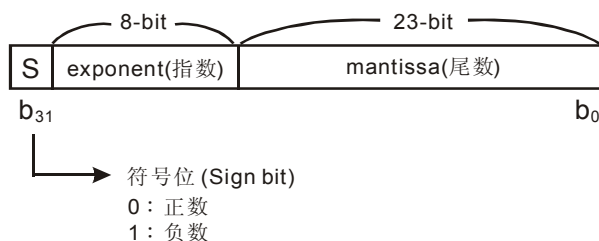
DVP-PM 系列机种的内部数值运算一般是以 BIN 整数值为准。整数执行除算时, 例: $40 \div 3 = 13$, 余数为 1。整数执行开平方动作时, 小数点会被舍弃掉。但若使用小数点运算指令则可求出小数点。

与小数点有关的应用指令如下所示。

API 110 (D ECMP)	API 111 (D EZCP)	API 116 (D RAD)	API 117 (D DEG)
API 120 (D EADD)	API 121 (D ESUB)	API 122 (D EMUL)	API 123 (D EDIV)
API 124 (D EXP)	API 125 (D LN)	API 126 (D LOG)	API 127 (D ESQR)
API 128 (D POW)	API 129 (D INT)	API 130 (D SIN)	API 131 (D COS)
API 132 (D TAN)	API 133 (D ASIN)	API 134 (D ACOS)	API 135 (D ATAN)
API 136 (D SINH)	API 137 (D COSH)	API 138 (D TANH)	

二进制浮点数表示法

DVP-PM 以 32 位的长度表示浮点数, 而表示法系采用 IEEE754 的标准, 格式如下:



可表达的大小为:

$$(-1)^S \times 2^{E-B} \times 1.M \quad \text{其中 } B = 127$$

5 应用指令分类及基本使用

因此 32 位的浮点数的数目范围为 $\pm 2^{-126}$ 到 $\pm 2^{+128}$ 相当于 $\pm 1.1755 \times 10^{-38}$ 到 $\pm 3.4028 \times 10^{+38}$ 。

范例一：以 32 位的浮点数表示 23

步骤一：将 23 转换成二进制数：23.0=10111

步骤二：将二进制数正规化：10111=1.0111 $\times 2^4$ ，其中 0111 为尾数，4 为指数。

步骤三：求出指数部份的储存值

$$\because E-B=4 \rightarrow E-127=4 \therefore E=131=10000011_2$$

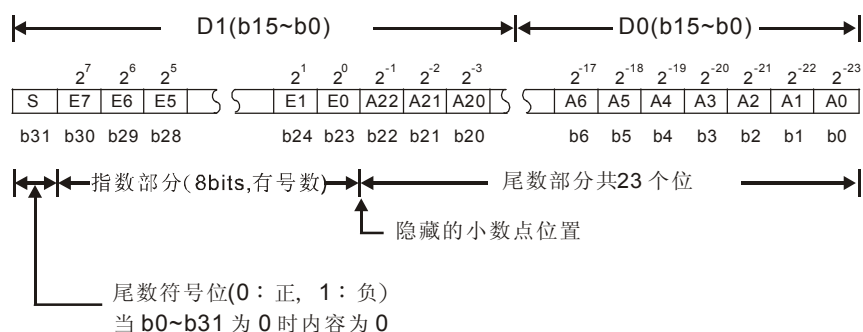
步骤四：组合符号位、指数、尾数成为浮点数。

$$0\ 10000011\ 011100000000000000000000_2=41B80000_{16}$$

范例二：以 32 位的浮点数表示 -23.0

-23.0 浮点格式与 23.0 的转换步骤完全相同，只需将符号位改为 1 即可。

DVP-PM 系列机种使用 2 个连续号码的寄存器组成 32 位的浮点数，我们以寄存器 (D1、D0) 来存放一个二进制浮点数为例，如下所示：



十进浮点数

◆ 二进制浮点数的内容比较无法被人所接受，因此二进制浮点数可转换成十进浮点数来供人作判断。但是 DVP-PM 系列主机对小数点的运算仍旧是使用二进制浮点数。

◆ 十进浮点数是使用 2 个连续号码的寄存器来表现，较小编号的寄存器号码存放常数部份、较大编号的寄存器号码存放指数部份。

就以寄存器 (D1、D0) 来存放一个十进浮点数为例，如下所示。

$$\text{十進浮點數} = [\text{指數 D1}] * 10$$

$$\text{底數 D0} = \pm 1,000 \sim \pm 9,999$$

$$\text{指數 D1} = -41 \sim +35$$

此外，底数 100 不存在于 D0 的内容，因为，100 是以 $1,000 \times 10^{-1}$ 来表现。十进浮点数的范围为 $\pm 1,175 \times 10^{-41}$ 到 $\pm 3,402 \times 10^{+35}$ 。

◆ 十进浮点数可使用于下列的指令中。

二进制浮点数 → 十进浮点数 的变换指令: (DEBCD)

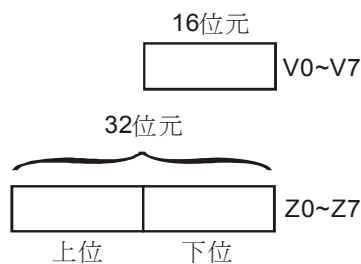
十进浮点数 → 二进制浮点数 的变换指令: (DEBIN)

◆ 当 O100~M102 的主控制程序区段中, 如果使用 ADD/SUB/MUL/DIV 等数值运算指令, 执行结果会影响 M1968~M1970 等旗标信号的状态。使用浮点运算指令, 执行结果也会影响零旗标信号 (M1968)、借位旗标信号 (M1970) 及进位旗标信号 (M1969)。说明如下:

- 零旗标信号: 运算结果为 0 时, M1968=On。
- 借位旗标信号: 运算结果超出最小处理单位时, M1970=On。
- 进位旗标信号: 运算结果绝对值超出使用范围时, M1969=On。

5.4 使用间接指定寄存器 V、Z 来修饰操作数

间接指定寄存器 V、Z 为 16 位及 32 位寄存器, DVP-PM 系列机种 V0~V7, Z0~Z7 共计 16 点。



V 是 16 位的数据寄存器, 它们可以自由的被写入及读出。但要使用 32 位长度时, 必须指定 Z。

DVP-PM 机种可修饰之装置为: P、I、X、Y、M、S、KnX、KnY、KnM、KnS、T、C、D。

可使用 V、Z 作修饰的各部装置如上所示。但是 V、Z 不可修饰本身、不可修饰常数 (K4@Z0 无效), 也不可以修饰 Kn。(K4M0@Z0 有效、K0@Z0M0 无效)。于个别应用指令说明中, 凡是于操作数表格中加入灰阶之操作数都可使用 V、Z 作修饰。

5 应用指令分类及基本使用

5.5 指令索引

● 指令以字母排列

分类	API	指令码		P 指令	功能	STEPS		页码
		16 位	32 位			16 位	32 位	
A	20	ADD	DADD	✓	BIN 加法	7	9	
	46	ANS	-	-	警报点输出	7	-	5-69
	47	ANR	-	✓	警报点复归	1	-	5-70
	66	ALT	-	✓	On/Off 交替	3	-	
	87	ABS	DABS	✓	绝对值	3	5	
	92	ANDP	-	-	正缘检出串联连接	3	-	
	93	ANDF	-	-	负缘检出串联连接	3	-	
	133	-	DASIN	✓	二进制浮点数 ASIN 运算	-	6	
	134	-	DACOS	✓	二进制浮点数 ACOS 运算	-	6	
	135	-	DATAN	✓	二进制浮点数 ATAN 运算	-	6	
	218	AND&	DAND&	-	S1 & S2	5	7	
	219	AND	DAND	-	S1 S2	5	7	
	220	AND^	DAND^	-	S1 ^ S2	5	7	
	232	AND=	DAND=	-	S1 = S2	5	7	
	233	AND>	DAND>	-	S1 > S2	5	7	
234	AND<	DAND<	-	S1 < S2	5	7		
236	AND<>	DAND<>	-	S1 ≠ S2	5	7		
237	AND<=	DAND<=	-	S1 ≤ S2	5	7		
238	AND>=	DAND>=	-	S1 ≥ S2	5	7		
B	15	BMOV	-	✓	全部传送	7	-	
	18	BCD	DBCD	✓	BIN→BCD 变换	5	5	
	19	BIN	DBIN	✓	BCD→BIN 变换	5	5	
	44	BON	DBON	✓	On 位判定	7	13	5-67
	258	BRET	-	-	回母线	1	-	
C	00	CJ	-	✓	条件跳跃	3	-	5-16
	01	CALL	-	✓	呼叫子程序	3	-	5-19
	10	CMP	DCMP	✓	比较设定输出	7	9	5-24
	14	CML	DCML	✓	反转传送	5	9	5-30
	97	CNT	DCNT	-	计数器	5	6	4-8
	131	-	DCOS	✓	二进制浮点数 COS 运算	5	6	5-116
	137	-	DCOSH	✓	二进制浮点数 COSH 运算	-	6	5-124
	256	CJN	-	✓	反条件跳跃	3	-	5-144
D	23	DIV	DDIV	✓	BIN 除法	7	9	5-42
	25	DEC	DDEC	✓	BIN 减一	3	3	5-44
	41	DECO	-	✓	译码器	7	-	5-62
	117	-	DDEG	✓	径度→角度	-	6	5-102
E	42	ENCO	-	✓	编码器	7	-	5-64
	110	-	DECMP	✓	二进制浮点数比较	7	9	5-98

分类	API	指令码		P 指令	功能	STEPS		页码
		16 位	32 位			16 位	32 位	
	111	-	DEZCP	✓	二进制浮点数区域比较	9	12	5-99
	112		DMOV	✓	浮点数值数据移动		9	5-100
	120	-	DEADD	✓	二进制浮点数加法	7	9	5-103
	121	-	DESUB	✓	二进制浮点数减法	7	9	5-104
	122	-	DEMUL	✓	二进制浮点数乘法	7	9	5-105
	123	-	DEDIV	✓	二进制浮点数除法	7	9	5-106
	124	-	DEXP	✓	二进制浮点数取指数	-	6	5-107
	127	-	DESQR	✓	二进制浮点数开平方根	5	6	5-110
	172	-	DADDR	✓	浮点数值加法	-	13	5-126
	173	-	DSUBR	✓	浮点数值减法	-	13	5-127
	174	-	DMULR	✓	浮点数值乘法	-	13	5-128
	175	-	DDIVR	✓	浮点数值除法	-	13	5-129
	F	16	FMOV	DFMOV	✓	多点移动	7	13
49		-	DFLT	✓	BIN 整数→二进制浮点数变换	-	6	5-73
78		FROM	DFROM	✓	扩充模块 CR 数据读出	9	12	5-84
I	24	INC	DINC	✓	BIN 加一	3	3	5-43
	129	-	DINT	✓	二进制浮点数→BIN 整数变换	-	6	5-113
J	257	JMP	-	-	无条件跳跃	3	-	5-145
L	90	LDP	-	-	正缘检出动作开始	3	-	4-9
	91	LDF	-	-	负缘检出动作开始	3	-	4-9
	125	-	DLN	✓	二进制浮点数取自然对数	-	6	5-108
	126	-	DLOG	✓	二进制浮点数取对数	-	9	5-109
	215	LD&	DLD&	-	S1 & S2	5	7	5-130
	216	LD	DLD	-	S1 S2	5	7	5-130
	217	LD^	DLD^	-	S1 ^ S2	5	7	5-130
	224	LD=	DLD=	-	S1 = S2	5	7	5-133
	225	LD>	DLD>	-	S1 > S2	5	7	5-133
	226	LD<	DLD<	-	S1 < S2	5	7	5-133
	228	LD<>	DLD<>	-	S1 ≠ S2	5	7	5-133
	229	LD<=	DLD<=	-	S1 ≦ S2	5	7	5-133
230	LD>=	DLD>=	-	S1 ≧ S2	5	7	5-133	
M	12	MOV	DMOV	✓	数据移动	5	6	5-26
	22	MUL	DMUL	✓	BIN 乘法	7	9	5-41
	45	MEAN	DMEAN	✓	平均值	7	13	5-68
	100	MODRD	-	-	MODBUS 数据读取	7	-	5-89
	101	MODWR	-	-	MODBUS 资料写入	7	-	5-93
	259	MMOV	-	✓	16→32 位数值转换	6	-	5-147
N	29	NEG	DNEG	✓	取负数 (取 2 的补码)	3	3	5-48
O	94	ORP	-	-	正缘检出并联连接	3	-	4-10
	95	ORF	-	-	负缘检出并联连接	3	-	4-11

5 应用指令分类及基本使用

分类	API	指令码		P 指令	功能	STEPS		页码
		16 位	32 位			16 位	32 位	
	221	OR&	DOR&	-	S1 & S2	5	7	5-132
	222	OR	DOR	-	S1 S2	5	7	5-132
	223	OR^	DOR^	-	S1 ^ S2	5	7	5-132
	240	OR=	DOR=	-	S1 = S2	5	7	5-135
	241	OR>	DOR>	-	S1 > S2	5	7	5-135
	242	OR<	DOR<	-	S1 < S2	5	7	5-135
	244	OR<>	DOR<>	-	S1 ≠ S2	5	7	5-135
	245	OR<=	DOR<=	-	S1 ≧ S2	5	7	5-135
	246	OR>=	DOR>=	-	S1 ≦ S2	5	7	5-135
P	89	PLS	-	-	上微分输出	3	-	4-11
	99	PLF	-	-	下微分输出	3	-	4-12
	128	-	DPOW	✓	浮点数权值指令	-	9	5-111
R	08	RPT	-	-	循环电路起始 (只能一层)	3	-	5-22
	09	RPE	-	-	循环电路结束	1	-	5-22
	30	ROR	DROR	✓	右旋转	5	9	5-50
	31	ROL	DROR	✓	左旋转	5	9	5-51
	32	RCR	DRCR	✓	附进位旗标右旋转	5	9	5-52
	33	RCL	DRCL	✓	附进位旗标左旋转	5	9	5-53
	50	REF	-	✓	I/O 更新处理	5	-	5-75
	67	RAMP	DRAMP	-	倾斜信号	9	17	5-80
	116	-	DRAD	✓	角度→弧度	-	6	5-101
	154	RAND	DRAND	✓	随机数值产生	7	13	5-137
260	RMOV	-	✓	32→16 位数值转换	6	-	5-148	
S	02	SRET	-	-	子程序结束	1	-	5-19
	13	SMOV	-	✓	位数移动	11	-	5-27
	21	SUB	DSUB	✓	BIN 减法	7	9	5-39
	34	SFTR	-	✓	位右移	9	-	5-54
	35	SFTL	-	✓	位左移	9	-	5-55
	38	SFWR	-	✓	位移写入	7	-	5-59
	39	SFRD	-	✓	位移读出	7	-	5-60
	48	SQR	DSQR	✓	BIN 开平方根	5	9	5-72
	43	SUM	DSUM	✓	On 位数量	5	9	5-66
	61	SER	DSER	✓	多点比较	9	17	5-77
	69	SORT	DSORT	-	数据排序	11	21	5-82
	147	SWAP	DSWAP	✓	上/下字节交换	3	5	5-136
	130	-	DSIN	✓	二进制浮点数 SIN 运算	5	6	5-114
136	-	DSINH	✓	二进制浮点数 SINH 运算	-	6	5-123	
202	SCAL	-	✓	比例值运算	9	-	5-140	
203	SCLP	DSCLP	✓	参数型比例值运算	7	13	5-144	
T	79	TO	DTO	✓	扩充模块 CR 数据写入	9	13	5-85

分类	API	指令码		P 指令	功能	STEPS		页码
		16 位	32 位			16 位	32 位	
	96	TMR	—	—	定时器	5	—	4-7
	132	—	DTAN	✓	二进制浮点数 TAN 运算	5	6	5-118
	138	—	DTANH	✓	二进制浮点数 TANH 运算	—	6	5-125
W	07	WDT	—	✓	逾时监视定时器	1	—	5-21
	26	WAND	DWAND	✓	逻辑及 (AND) 运算	7	9	5-45
	27	WOR	DWOR	✓	逻辑或 (OR) 运算	7	9	5-46
	28	WXOR	DWXOR	✓	逻辑互斥或 (XOR) 运算	7	9	5-47
	36	WSFR	—	✓	寄存器右移	9	—	5-56
	37	WSFL	—	✓	寄存器左移	9	—	5-58
X	17	XCH	DXCH	✓	数据的交换	5	9	5-34
Z	11	ZCP	DZCP	✓	区域比较	9	12	5-25
	40	ZRST	—	✓	区域清除	5	—	5-61

5 应用指令分类及基本使用

5.6 应用指令

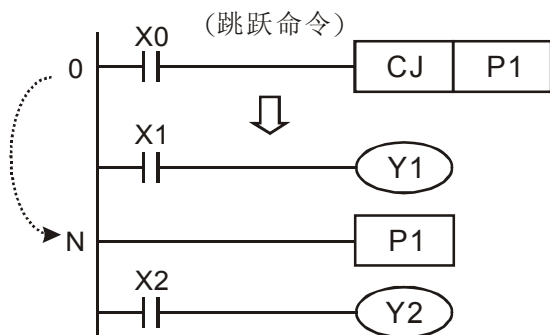
API																		适用機種		
00			CJ	P	S														20PM	10PM
																			✓	✓
		位装置			字符装置												:16 位指令 (3 STEP)			
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	:CJ 连续执行型 CJP 脉冲执行型			
		<ul style="list-style-type: none"> 操作数使用注意: S 操作数可指定 P S 操作数可指定 P0~P255 P 装置不支持 V、Z 修饰 															:32 位指令			
																	• 旗标信号: 无			

指令说明

- ◆ **S**: 条件跳跃之目的指标。
- ◆ 当使用者希望 O100 主程序中的某一部份不需要执行时，以缩短扫描时间，以及使用于双重输出时，可使用 **CJ** 或 **CJP** 指令。
- ◆ 指针 **P** 所指之程序若在 **CJ** 指令之前，需注意会发生 **WDT** 逾时之错误，O100 主程序停止运转，请注意使用。
- ◆ **CJ** 指令可重复指定同一指标 **P**，但 **CJ** 与 **CALL** 不可指定同一指标 **P**，否则会产生错误。
- ◆ 跳跃执行中各种装置动作情形说明：
 1. **Y**、**M**、**S** 保持跳跃发生前之状态。
 2. 执行计时中之 **10ms** 定时器会暂停计时。
 3. 一般计数器会停止计数，以及一般应用指令不会被执行。
 4. 定时器之清除指令若在跳跃前被驱动，则在跳跃执行中该装置仍处于清除状态。

程式范例 (一)

- ◆ 当 **X0=On** 时，程序自动从地址 **0** 跳跃至地址 **N** (即指定之标签 **P1**) 继续执行，中间地址跳过不执行。
- ◆ 当 **X0=Off** 时，程序如同一般程序由地址 **0** 继续往下执行，此时 **CJ** 指令不被执行。



程式范例
(二)

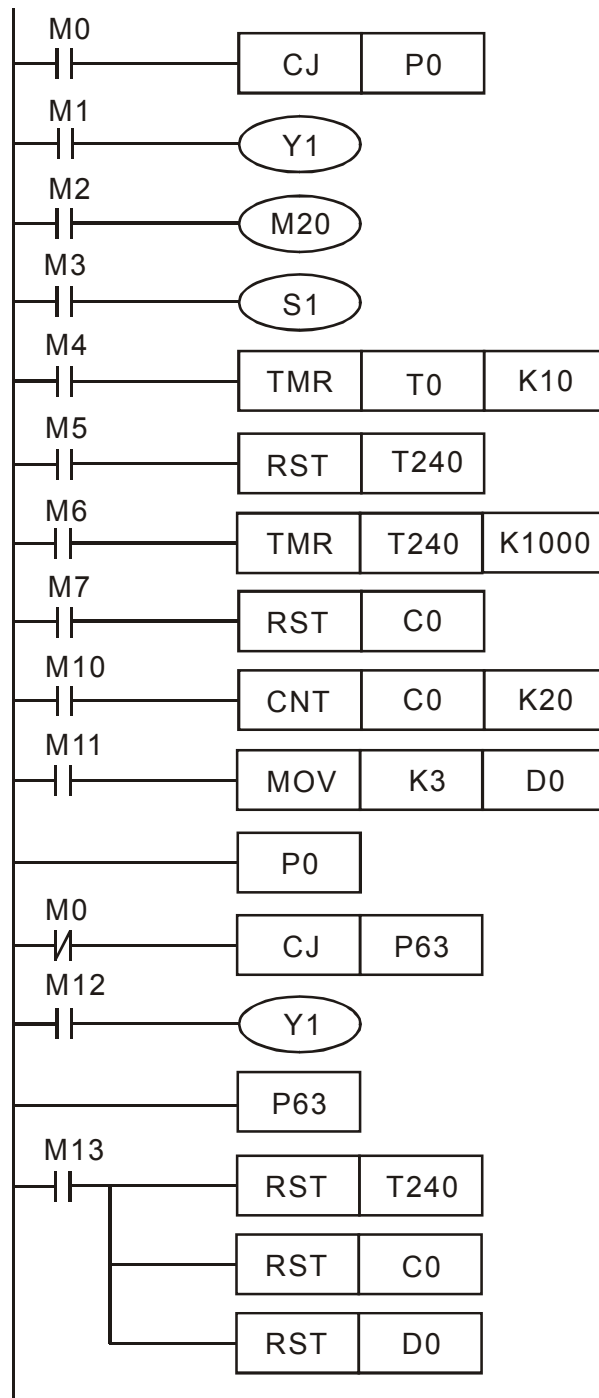
◆ 底下表格为下列程序中，各个装置状态变化。

装置	CJ 执行前 接点状态	CJ 执行中 接点状态	CJ 执行中 输出线圈状态
Y、M、S	M1、M2、M3 Off	M1、M2、M3 由 Off→On	Y1 ^{注一} 、M20、S1 Off
	M1、M2、M3 On	M1、M2、M3 由 On→Off	Y1 ^{注一} 、M20、S1 On
10ms 定时器	M4 Off	M4 由 Off→On	定时器 T0 不作计时动作
	M4 On	M4 由 On→Off	定时器 T0 立即停止计时并 保持，M0 由 On→Off，T0 被清为 0
	M6 Off	M6 由 Off→On	定时器 T240 不作计时动作
	M6 On	M6 由 On→Off	定时器 T240 立即停止计时 并保持，M0 由 On→Off， T240 被清为 0
C0~C234	M7、M10 Off	M10 On/Off 触发	计数器 C0 不计数
	M7 Off、M10 On/Off 触发	M10 On/Off 触发	计数器 C0 停止计数并保持， M0 Off 后，C0 继续计数
应用指令	M11 Off	M11 由 Off→On	应用指令不执行
	M11 On	M11 由 On→Off	被跳过之应用指令不执行， 但是 API 53~59、API 157~159 继续动作

注一：Y1 为双重输出，M0 为 Off 时，由 M1 控制，M0 为 On 时，由 M12 控制。

5 应用指令分类及基本使用

- ◆ Y1 为双重输出，M0=Off 时，由 M1 来控制，M0=On 时，由 M12 来控制。



API																	适用機種	
01	CALL	P	S	呼叫子程序													20PM	10PM
				✓	✓													
位裝置		字符裝置															16 位指令 (3 STEP)	
X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	CALL	连续执行型	CALLP	脉冲执行型
• 操作数使用注意: S 操作数可指定 P S 操作数可指定 P0~P255 P 装置不支持 V、Z 修饰																	32 位指令	
																	- - - -	
																	• 旗标信号: 无	

指令说明

- ◆ S: 呼叫子程序之指针。
- ◆ 指针所指定的子程序请于 M102、M2 和 SRET 指令后编写。
- ◆ 指针 P 之号码在被 CALL 使用时, 不可与 CJ、CJN、JMP 指令指定相同之号码。
- ◆ 若仅使用 CALL 指令则可不限次数呼叫同一指针号码之子程序。

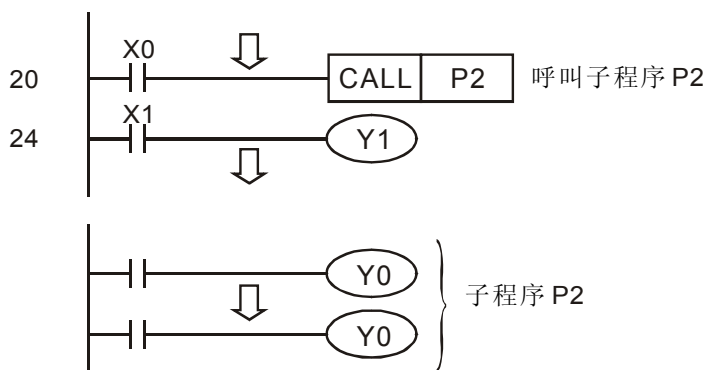
API																		适用機種	
02	SRET	子程序结束															20PM	10PM	
		✓	✓																
位裝置		字符裝置															16 位指令 (1 STEP)		
X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SRET	连续执行型	-	-	
• 操作数使用注意: 无操作数 不须接点驱动的指令																	32 位指令		
																	- - - -		
																	• 旗标信号: 无		

指令说明

- ◆ 表示子程序结束。子程序执行结束由 SRET 返回 O100 主程序, 执行原呼叫该子程序 CALL 指令的下一个指令。

程式范例 (一)

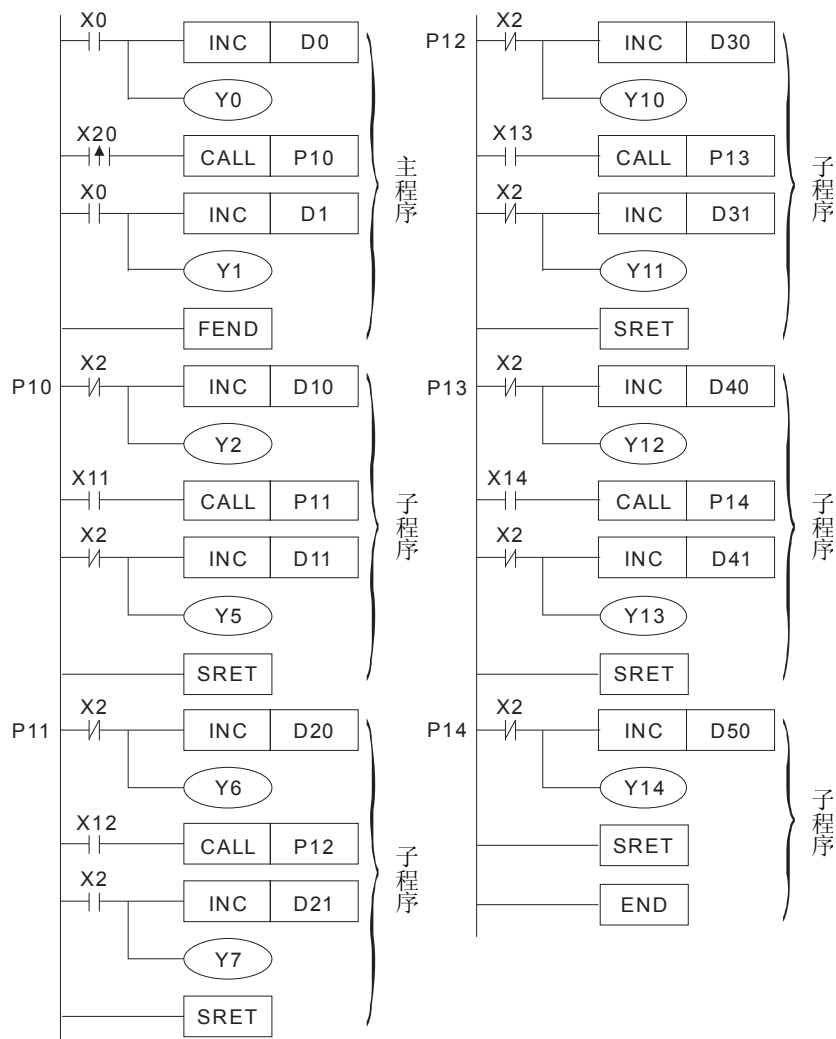
- ◆ 当 X0 为 On 时, 则执行 CALL 指令, 跳跃到 P2 执行所指定的子程序, 当执行 SRET 指令时, 则回到地址 24, 继续往下执行。



5 应用指令分类及基本使用

程式范例 (二)

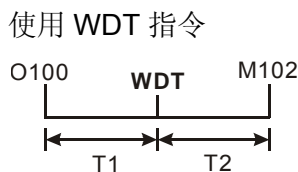
- ◆ 当 X20 为由 Off 到 On 之正缘触发执行 CALL P10 指令，跳跃到 P10 执行所指定的子程序。
- ◆ 当 X11 为 On 时，则执行 CALL P11，跳跃到 P11 执行所指定的子程序。
- ◆ 当 X12 为 On 时，则执行 CALL P12，跳跃到 P12 执行所指定的子程序。
- ◆ 当 X13 为 On 时，则执行 CALL P13，跳跃到 P13 执行所指定的子程序。
- ◆ 当 X14 为 On 时，则执行 CALL P14，跳跃到 P14 执行所指定的子程序，当执行到 SRET 指令时，则回到前一个 P※ 子程序继续往下执行。
- ◆ 在 P10 子程序中执行到 SRET 指令后回到主程序。



API																适用機種					
07	WDT															20PM		10PM			
																✓		✓			
		位裝置				字裝置												16 位指令 (1 STEP)			
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WDT	連續執行型	WDT _P	脈衝執行型	
		<ul style="list-style-type: none"> 操作數使用注意： 无操作數 不須接點驅動的指令 														32 位指令					
																<ul style="list-style-type: none"> 旗標信號：无 					

指令说明

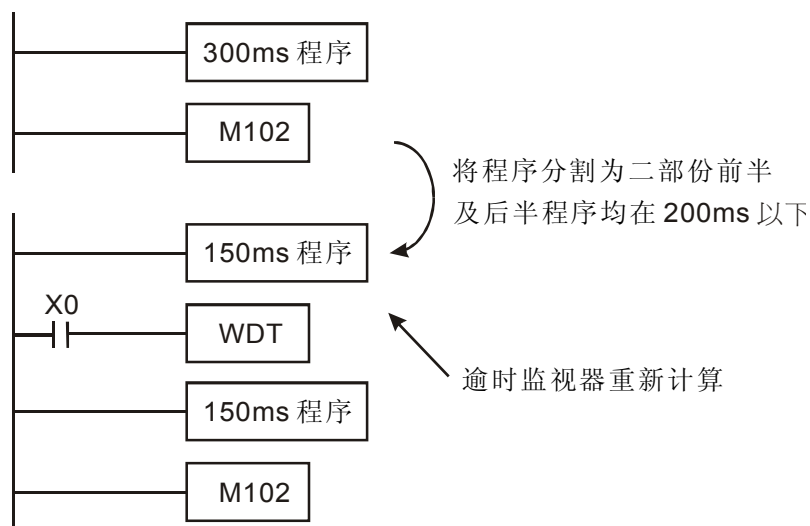
- ◆ WDT 指令可用来清除 PM 中的监控定时器的计时时间。当 PM 的扫描 (由地址 0 至 END 或 FEND 指令执行时间) 超过 200ms 时, PM ERROR 的指示灯会亮, 使用者必须将 PM 电源 Off 再 On, PM 会依据 MANU/AUTO 开关来判断 RUN/STOP 状态, 若无 MANU/AUTO 开关, 则 PM 会自动回到 STOP 状态。
- ◆ 令逾时监视定时器动作的时机：
 - ◆ PM 系统发生异常。
 - ◆ 程序执行时间太长, 造成扫描周期大于 D1000 的内容值。可以下列 2 种方法来改善：



可由 D1000(出厂设定值为 200ms)的设定值改变逾时监视时间。

程式范例

- ◆ 若程序扫描时间为 300ms, 此时, 可将程序分割为 2 部份, 并在中间放入 WDT 指令, 使得前半及后半程序都在 200ms 以下。



补充说明

- ◆ WDT 指令用于输入条件成立时的程序扫描。可编写程序强制 WDT 指令只在一个周期内执行。PM 用户可以使用 WDT 指令的脉冲型指令 WDT_P。
- ◆ PM 逾时监视定时器的预设设定值是 200ms。此时间限制可由用户自行定制, 透过编辑数据寄存器 D1000 即可。

5 应用指令分类及基本使用

API 08	RPT	Ⓢ	巢串回路起始											适用机种				
														20PM	10PM			
														✓	✓			
S	位装置				字符装置										16 位指令 (3 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	RPT	连续执行型	-
• 操作数使用注意：不须接点驱动的指令 此指令有支持 V 装置 各装置使用范围请参考功能规格表																32 位指令		
																— — — —		
																旗标信号：无		

指令说明

- ◆ S：回路重复执行的次数。
- ◆ 巢串式 RPT~RPE 回路只能使用一层，当回路层数超过一层，就会显示错误产生。

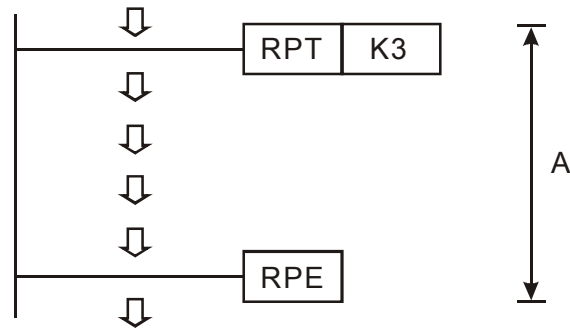
API 09	RPE	巢串回路结束											适用机种					
													20PM	10PM				
													✓	✓				
S	位装置				字符装置										16 位指令 (1 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	RPE	连续执行型	-
• 操作数使用注意：无操作数 不须接点驱动的指令																32 位指令		
																— — — —		
																• 旗标信号：无		

指令说明

- ◆ 由 RPT 指令指定 RPT~RPE 循环来回执行 N 次后跳出 RPT~RPE 循环往下继续执行。
- ◆ 指定次数范围 N=K1~K32,767，当指定次数范围 N≤K1 时，都视为是 K1。
- ◆ 当不执行 RPT~RPE 回路时，可使用 CJ 指令来跳出回路。
- ◆ 下列情形会产生错误：
 - ◆ RPE 指令在 RPT 指令之前。
 - ◆ 有 RPT 指令没有 RPE 指令。
 - ◆ RPT~RPE 指令个数不同时。
- ◆ 巢串式 RPT~RPE 回路只能使用一层，当回路次数超过一次，就会显示错误产生。

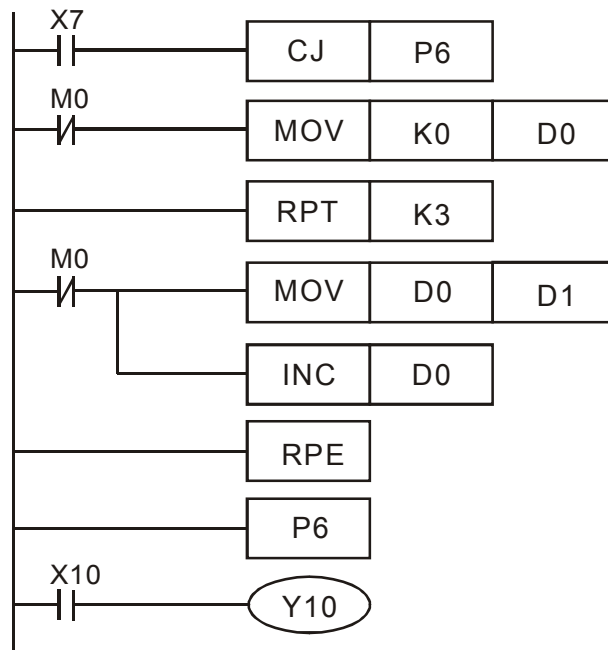
程式范例
(一)

- ◆ 如果想使 A 区段程序执行 3 次，可利用 RPT~RPE 之写法如下所示：



程式范例
(二)

- ◆ 当 X7 = Off 时，PM 会执行 RPT~RPE 之间的程序，当 X7 = On 时，CJ 指令执行跳跃至 P6 处，RPT~RPE 之间的程序跳过不执行。



5 应用指令分类及基本使用

API																	适用机种		
10	D	CMP	P		(S ₁)	(S ₂)	(D)	比较设定输出									20PM	10PM	
																	✓	✓	
	位装置				字符装置												:16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	CMP	连续执行型	CMPP	脉冲执行型
S ₁					*	*	*	*	*	*	*	*	*	*	*				
S ₂					*	*	*	*	*	*	*	*	*	*	*				
D		*	*	*															
<ul style="list-style-type: none"> 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 																:32 位指令 (9 STEP)			
																DCMP	连续执行型	DCMPP	脉冲执行型
																• 旗标信号：无			

指令说明

◆ **S₁**：比较值 1。**S₂**：比较值 2。**D**：比较结果。

◆ 比较值 **S₁** 与 **S₂** 内容作比较，其比较结果在 **D** 作表示。

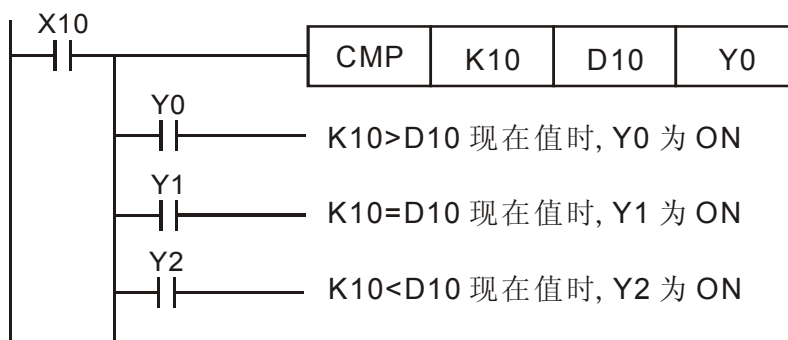
◆ D 操作数会占用连续 3 个点。

◆ 指定装置为 Y0，则自动占有 Y0，Y1 及 Y2。

程式范例

◆ 当 X10=On 时，CMP 指令执行，Y0，Y1 及 Y2 其中之一会 On，当 X10=Off 时，CMP 指令不执行，Y0，Y1 及 Y2 状态保持在 X10=Off 之前的状态。

◆ 若需要得到 \geq 、 \leq 、 \neq 之结果时，可将 Y0~Y2 串并联即可取得。



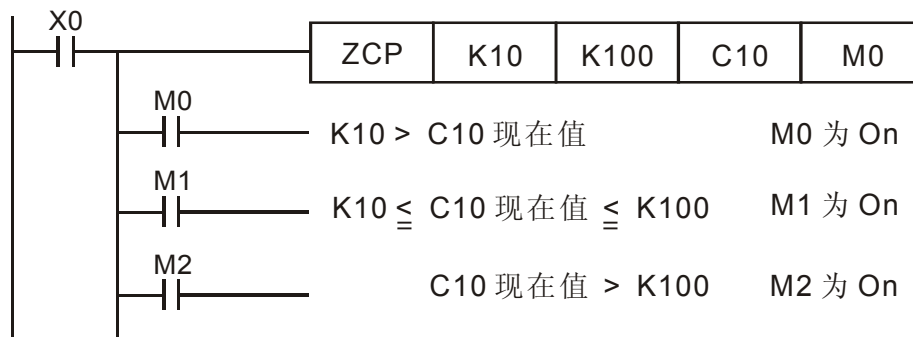
API																		适用機種	
11	D	ZCP	P		(S ₁)	(S ₂)	(S)	(D)	区域比较								20PM	10PM	
																		✓	✓
	位装置				字符装置												16 位指令 (9 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ZCP	连续执行型	ZCPP	脉冲执行型
S ₁					*	*	*	*	*	*	*	*	*	*	*				
S ₂					*	*	*	*	*	*	*	*	*	*	*				
S					*	*	*	*	*	*	*	*	*	*	*				
D		*	*	*															
• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表																	• 旗标信号：无		

指令说明

- ◆ **S₁**：区域比较之下限值。**S₂**：区域比较之上限值。**S**：比较值。**D**：比较结果。
- ◆ 比较值 **S** 与下限 **S₁** 及上限 **S₂** 作比较，其比较结果在 **D** 作表示。
- ◆ **S₂** 必须比 **S₁** 大。
- ◆ D 操作数会占用连续 3 个点。

程式范例

- ◆ 指定装置为 M0，则自动占有 M0，M1 及 M2。
- ◆ 当 X0=On 时，ZCP 指令执行，M0，M1 及 M2 其中之一会 On，当 X0=Off 时，ZCP 指令不执行，M0，M1 及 M2 状态保持在 X0=Off 之前的状态。



5 应用指令分类及基本使用

API																	适用機種		
12	D	MOV	P		(S)	(D)												20PM	10PM
																		✓	✓
	位装置				字符装置												16 位指令 (5 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	MOV 连续执行型 MOVP 脉冲执行型			
S					*	*	*	*	*	*	*	*	*	*	*				
D							*	*	*	*	*	*	*	*	*				
• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表																32 位指令 (6 STEP) DMOV 连续执行型 DMOVP 脉冲执行型 • 旗标信号：无			

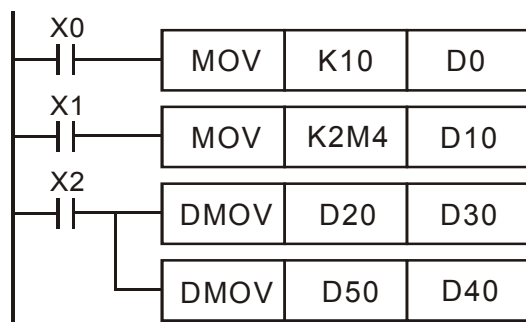
指令说明

- ◆ **S**：资料之来源。**D**：数据之搬移目的地。
- ◆ 当该指令执行时，将 **S** 的内容直接搬移至 **D** 内。当指令不执行时，**D** 内容不会变化。
- ◆ 若演算结果为 32 位输出时，(如应用指令 MUL 等) 数据搬动则必须要用 **DMOV** 指令。

程式范例

- ◆ 16 位数据搬移，须使用 MOV 指令。
 1. 当 X0=Off 时，D0 内容没有变化，若 X0=On 时，将数值 K10 传送至 D0 数据寄存器内。
 2. 当 X1=Off 时，D10 内容没有变化，若 X1=On 时，将 K2M4 现在值传送至 D10 数据寄存器内。
- ◆ 32 位数据搬移，须使用 DMOV 指令。

当 X2=Off 时，(D31、D30)、(D41、D40) 内容没有变化，若 X2=On 时，将 (D21、D20) 现在值传送至 (D31、D30) 数据寄存器内。同时，将 (D51、D50) 现在值传送至 (D41、D40) 数据寄存器内。



API				(S) (m1) (m2) (D)	位数移动	适用机种	
13	SMOV	P		(n)		20PM	10PM
						✓	✓

	位装置				字符装置											16 位指令 (11 STEP) SMOV 连续执行型 SMOV P 脉冲执行型	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		32 位指令
S							*	*	*	*	*	*	*	*	*	*	
m1					*	*											
m2					*	*											
D								*	*	*	*	*	*	*	*	*	
n					*	*											

• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置)
 各装置使用范围请参考功能规格表
 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。

• 旗标信号：M1168 SMOV 工作模式指定

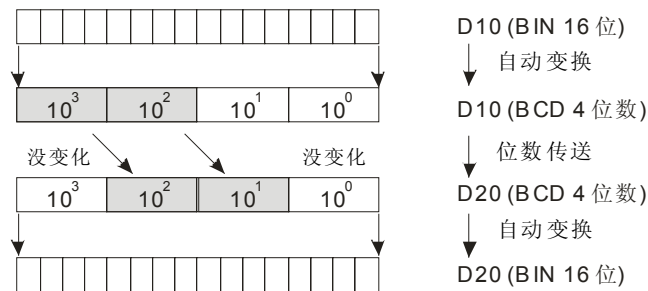
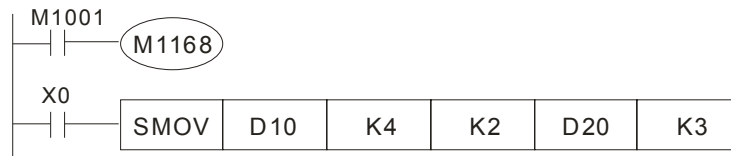
指令说明

- ◆ **S**：资料来源。**m₁**：数据来源传送起始位数。**m₂**：数据来源传送位数的个数。**D**：传送的目的地装置。**n**：传送的目的地起始位数。
- ◆ **BCD 模式(M1168=Off)**：
此模式下 SMOV 致能操作 BCD 数，与 SMOV 操作十进制数字类似。也就是说，此指令复制操作数 S(S 是 4 位 BCD 数)的指定位数并传送至操作数 D(D 同样也是 4 位 BCD 数)。
- ◆ **BIN 模式(M1168=On)**：
此指令复制操作数 S(S 是 4 位十进制数字)的指定位数并传送至操作数 D(D 同样也是 4 位十进制数字)。目标寄存器的现有数据被覆盖。
- ◆ **m₁** 的范围：1 - 4
- ◆ **m₂** 的范围：1 - m₁ (不能大于 m₁)
- ◆ **n** 的范围：m₂ - 4 (不能小于 m₂)

5 应用指令分类及基本使用

程式范例 (一)

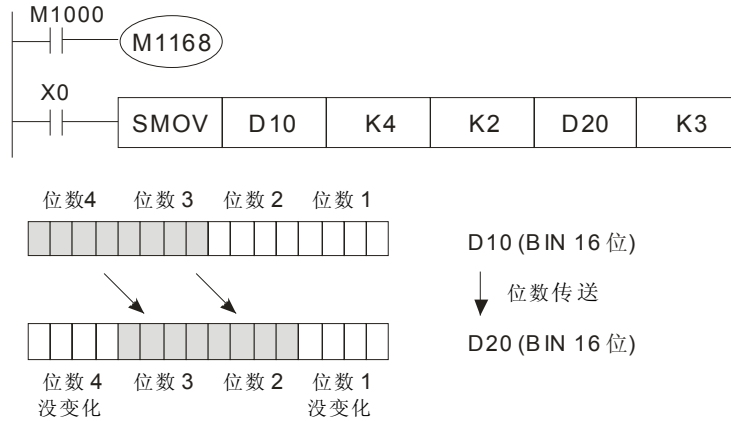
- ◆ 当 M1168=Off 时(BCD 模式), X0=On, 指定 D10 的 10 进制数值的第 4 位数(也即千位数)开始往低位计算的 2 位数内容传送至 D20 的 10 进制数值的第 3 位数(也即百位数)开始往低位计算的 2 位数中。而 D20 的 10^3 及 10^0 于本指令被执行后内容没有变化。
- ◆ 当 BCD 值超过 0~9,999 的范围时, PM 判定为运算错误, 指令不执行, M1067, M1068=On, D1067 记录错误代码 0E18 (Hex)。



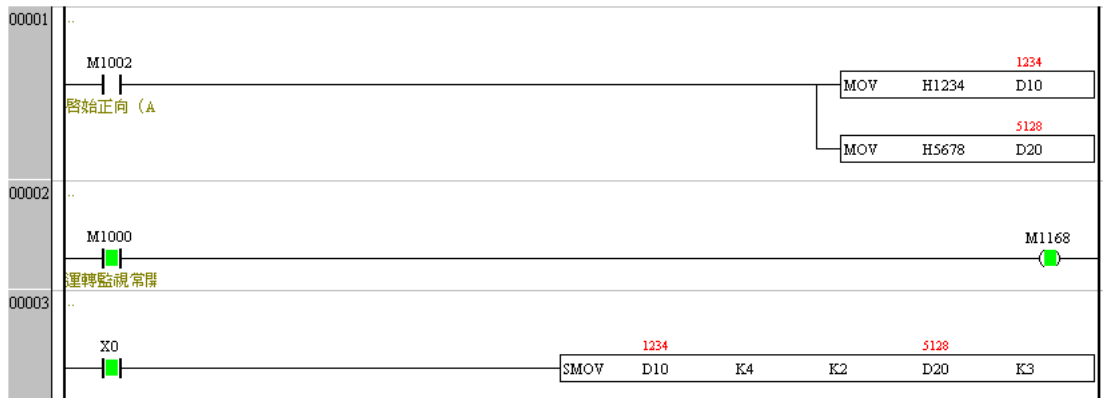
- ◆ 如果在执行前 D10=H1234, D20=H5678, 执行后 D10 的值保持不变, 而 D20=H5128。(实际执行结果为 09A2)

程式范例
(二)

- ◆ 当 M1168=On(BIN 模式)时, 使用 SMOV 指令的话, D10, D20 并不会作 BCD 变换, 而是以 BIN 型态 4 个位为一个单位作传送。

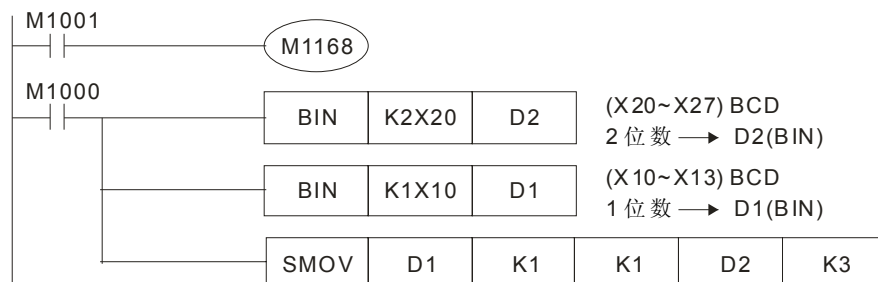
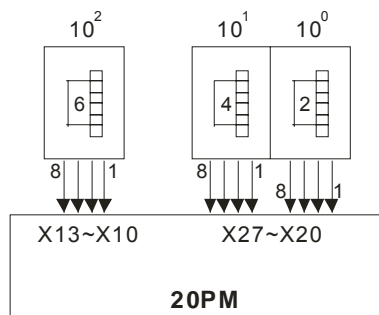


执行结果 ↓



程式范例
(三)

- ◆ 将右 2 位指拨开关传送至 D2 的右 2 位, 左 1 位指拨开关传送至 D1 的右 1 位数当中。使用 SMOV 指令将 D1 的第 1 位传送至 D2 的第 3 位数将两组指拨开关合成 1 组。



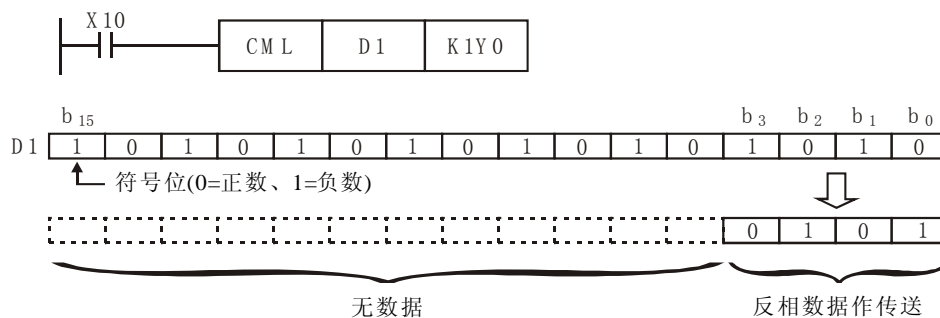
5 应用指令分类及基本使用

API																	适用機種		
14	D	CML	P	S	D	反转传送											20PM	10PM	
																	✓	✓	
	位装置				字符装置											16位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	CML	连续执行型	CMLP	脉冲执行型
S					*	*	*	*	*	*	*	*	*	*	*				
D							*	*	*	*	*	*	*	*	*				
操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。																32位指令 (6 STEP) DCML 连续执行型 DCMLP 脉冲执行型			
																旗标信号：无			

指令说明

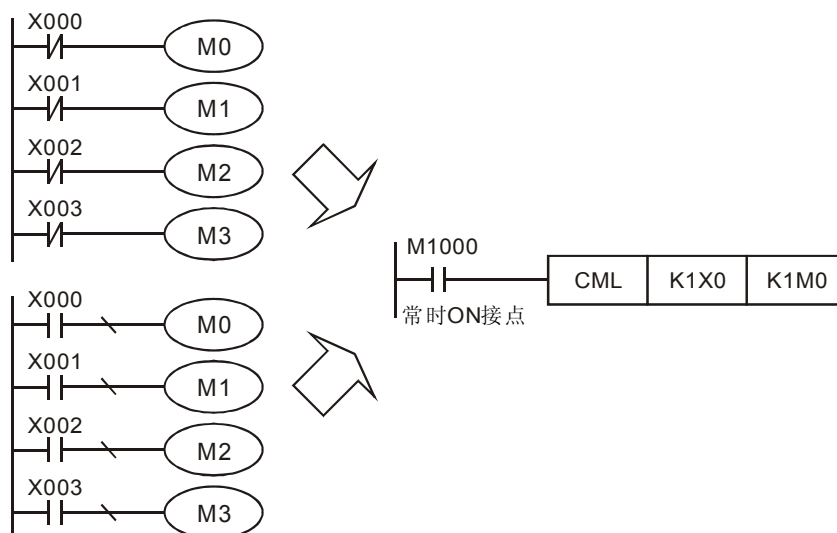
- ◆ **S**：传送数据来源。**D**：传送的目的地装置。
- ◆ 将 **S** 的内容全部取反(0→1, 1→0)传送到 **D** 当中。
- ◆ 当 X10=On 时，将 D1 的 b0~b3 内容反相后传送到 Y0~Y3。

程式范例 (一)



程式范例 (二)

- ◆ 下图左边的回路当也可以使用 CML 指令来表现，如下图右所示。

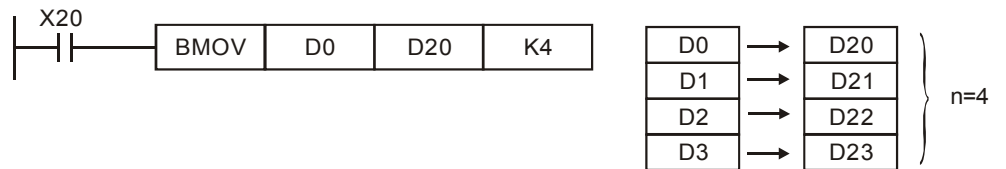


API																	适用機種	
15	BMOV	P	S	D	n	全部传送											20PM	10PM
						✓	✓											
	位装置				字符装置											16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	BMOV 连续执行型	BMOV P 脉冲执行型	
S							*	*	*	*	*	*	*					
D								*	*	*	*	*	*					
n					*	*					*	*	*					
<p>操作数使用注意：各装置使用范围请参考功能规格表</p> <p>指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)</p> <p>指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。</p>																		
<p>• 旗标信号：无</p>																		

指令说明

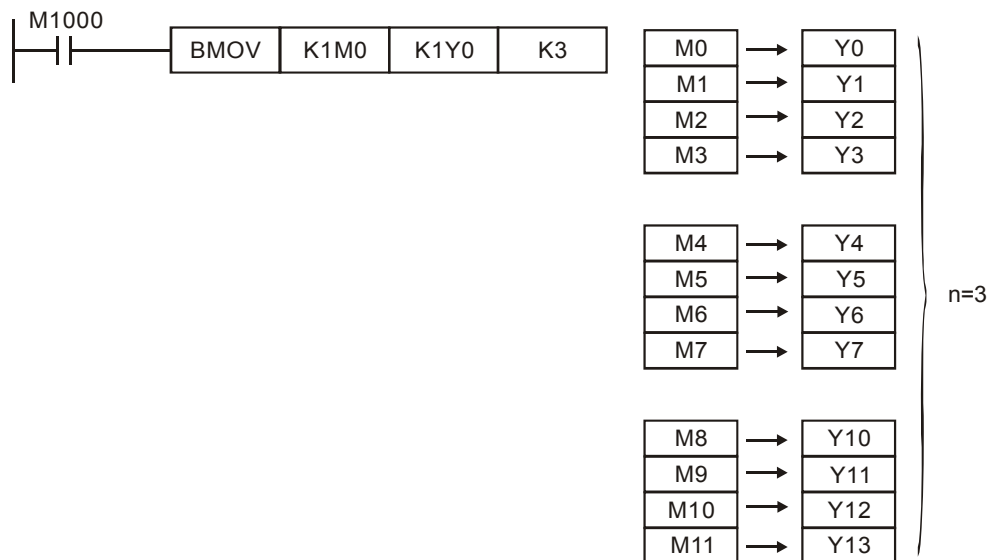
- ◆ **S**：来源装置起始。**D**：目的地装置起始。**n**：传送区块长度。
- ◆ 此指令用于传送多笔数据到新的寄存器。**S** 所指定的装置起始号码开始算 **n** 个寄存器的内容被传送至 **D** 所指定的装置起始号码开始算 **n** 个寄存器当中，如果 **n** 所指定点数超过该装置的使用范围时，只有有效范围被传送。
- ◆ **n** 的范围 1~512。
- ◆ 当 X20=On, D0~D3 的内容被传送到 D20~D23 中。

程式范例 (一)



程式范例 (二)

- ◆ 如果指定位装置 KnX, KnY, KnM, KnS 作传送时，**S** 及 **D** 的位数必须相同。



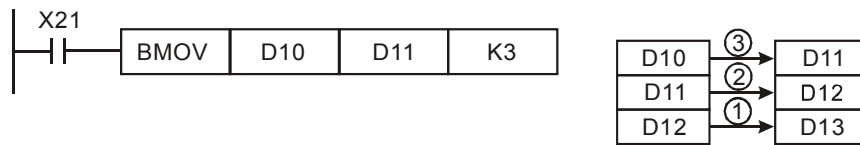
程式范例
(三)

- ◆ 为了防止两个操作数所指定传送的号码重迭时，所造成的混乱，请注意两个操作数所指定号码大小的安排。

1. 当 $S > D$, BMOV 指令以 ①→②→③ 的顺序传送



2. 当 $S < D$, the BMOV 指令以 ③→②→① 的顺序传送, D11~D13 内容数值都是 D10 的内容。



API																适用機種	
16	FMOV	P	S	D	n	多点传送										20PM	10PM
						✓	✓										

	位装置				字符装置												16位指令 (7 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	FMOV 连续执行型	FMOV P 脉冲执行型	
S							*	*	*	*	*	*	*	*	*			
D								*	*	*	*	*	*	*				
n					*	*												

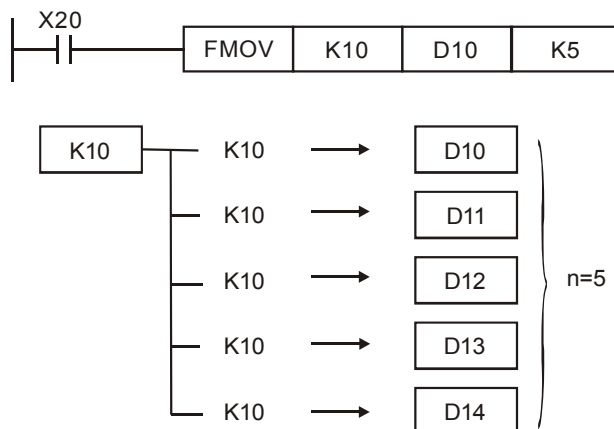
• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置)
 各装置使用范围请参考功能规格表
 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。

• 旗标信号：无

指令说明

- ◆ **S**：资料的来源。**D**：目的地装置起始。**n**：传送区块长度。
- ◆ **S** 的内容被传送至 **D** 所指定的装置起始号码开始算 **n** 个寄存器当中，如果 **n** 所指定点数超过该装置的使用范围时，只有有效范围被传送
- ◆ **n** 的范围 1~512。
- ◆ 当 X20=On 时，K10 被传送到由 D10 开始的连续 5 个寄存器(D10~D14)中。

程式范例



5 应用指令分类及基本使用

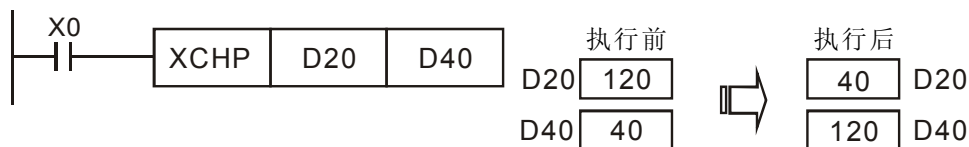
API																	适用機種				
17	D	XCH	P	D1		D2		数据的交换									20PM	10PM			
																	✓	✓			
		位装置				字符装置											16 位指令 (5 STEP)				
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	XCH	连续执行型	XCHP	脉冲执行型	
D1								*	*	*	*	*	*	*	*	*					
D2								*	*	*	*	*	*	*	*	*					
操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时, 限以 8 进制或 10 进制中 16 的倍数为起始编号。如: K1X0, K4Y20 (八进制); K1M0, K4S16 (十进制)。																	32 位指令 (9 STEP)				
																	DXCH		连续执行型	DXCHP	脉冲执行型
																	旗标信号: 无				

指令说明

- ◆ D₁: 互相交换的数据 1。D₂: 互相交换的数据 2。
- ◆ 将 D₁ 与 D₂ 的内容互换。
- ◆ 此指令最好以脉冲的方式执行(XCHP)。

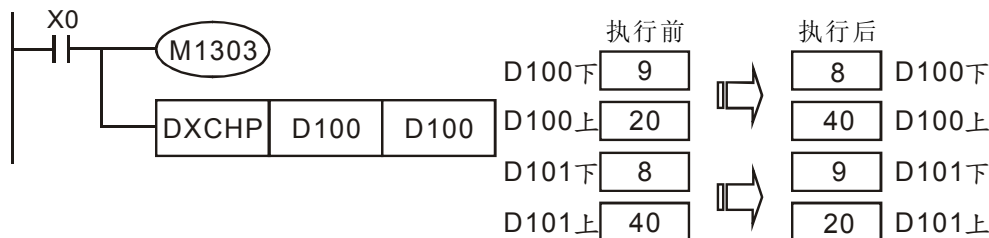
程式范例

- ◆ 当 X0=Off→On, D20 与 D40 的内容互换。



补充说明

- ◆ 16 位指令当 D₁ 及 D₂ 所指定的装置相同时, 且 M1303=On, 则该装置的上下 8 位内容互相交换。
- ◆ 32 位指令当 D₁ 及 D₂ 所指定的装置相同时, 且 M1303=On, 则该 32 位装置个别上下 16 位内容互相交换。
- ◆ 当 X0=On 时, 且 M1303=On, D100 的 16 位内容与 D101 的 16 位内容与 D101 的 16 位内容互相交换。



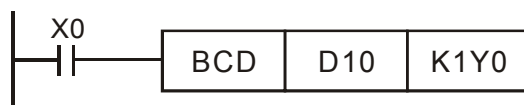
API																	适用機種		
18	D	BCD	P	(S)	(D)													20PM	10PM
																		✓	✓
	位装置				字符装置										16 位指令 (5 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	BCD	连续执行型	BCDP	脉冲执行型
S							*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*	*	*				
<ul style="list-style-type: none"> 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 																<ul style="list-style-type: none"> 旗标信号： 			
																Ox	O100		
																M1793	M1953	演算错误旗标	Operation error flag

指令说明

- ◆ 2 进制 S 的内容 (BIN 值) 被转换成 BCD 码数据，然后传送到 D 中。
- ◆ 如果 BCD 数据转换范围在 0 到 9,999 之外，则 BCD 指令不会执行。如果 BCD 转换数据范围是在 0 到 99,999,999 之外，则 **DBC**D 指令不会执行
- ◆ BCD 指令把定位单元中的二进制数据转成 BCD 码数据，再输出到外部装置中 (如 7 段显示器等等)。

程式范例

- ◆ 当 X0=On 时，D10 之 BIN 值被转换成 BCD 值后，将结果的个位数存于 K1Y0 (Y0~Y3) 四个 bit 组件。



若 D10=001E (Hex) =0030 (十进制)，则执行结果 Y0~Y3=0000 (BIN)。

5 应用指令分类及基本使用

API	W	BIN	P	(S)	(D)	BIN 转换										适用機種			
19	D																	20PM	10PM
																		✓	✓
	位装置				字符装置												16 位指令 (5 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	BIN 连续执行型 BINP 脉冲执行型			
S							*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*	*	*				
• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表																• 旗标信号： Ox O100 M1793 M1953 演算错误旗标 Operation error flag			

指令说明

- ◆ **S**：资料之来源。**D**：变换之结果资料。
- ◆ 2 进制 **S** 的内容 (BCD 值) 被转换成 BIN 码数据，然后传送到 **D** 中。
- ◆ 数据来源 **S** 的内容有效数值范围：BCD (0~9,999)，**DBCD** (0~99,999,999)。
- ◆ 常数 K、H 会自动转换成 BIN，故不需运用此指令。

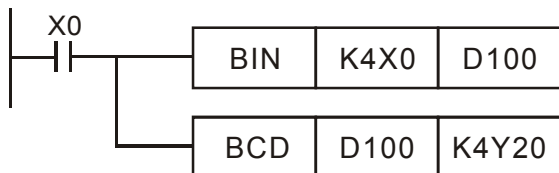
程式范例 (一)

- ◆ 当 X0=On 时，K1M0 之 BCD 值被转换成 BIN 值后，将结果存于 D10 中。



补充说明

- ◆ BCD 与 BIN 指令应用说明：
 3. 当 DVP-PM 要从外界读取一个 BCD 型态指拨开关时，就必须使用 BIN 指令先将读取到的数据转换成 BIN 值再储存在 DVP-PM 内。
 4. 当 DVP-PM 要将内部储存的资料经由外界一个 BCD 型态的 7 段显示器显示出来时，就必须使用 BCD 指令先将要显示的内部数据转换成 BCD 值再送到 7 段显示器。
 5. 当 X0=On 时，将 K4X0 BCD 值转换成 BIN 值传送到 D100，再将 D100 之 BIN 值转换成 BCD 值传送到 K4Y20。



API																适用機種			
20	D	ADD	P	(S1)	(S2)	(D)	BIN 加法										20PM	10PM	
																	✓	✓	
	位装置				字符装置											16 位指令 (7 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ADD	连续执行型	ADDP	脉冲执行型
S ₁					*	*	*	*	*	*	*	*	*	*	*				
S ₂					*	*	*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*	*	*				
<ul style="list-style-type: none"> 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 																<ul style="list-style-type: none"> 旗标信号 <ul style="list-style-type: none"> Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明 			

指令说明

- ◆ **S₁**: 被加数。**S₂**: 加数。**D**: 和。
- ◆ 将两个资料来源: **S₁** 及 **S₂** 以 BIN 方式相加的结果存于 **D**。
- ◆ 各数据的最高位为符号位 0 表 (正) 1 表 (负), 因此可做代数加法运算。(例如: 3+ (-9) =-6)。
- ◆ 加法相关旗号变化。

16 位 BIN 加法:

1. 演算结果为 0 时, 零旗号 (Zero flag) 为 On。
2. 演算结果小于 -32,768 时, 借位旗号 (Borrow flag) 为 On。
3. 演算结果大于 32,767 时, 进位旗号 (Carry flag) 为 On。

32 位 BIN 加法:

1. 演算结果为 0 时, 零旗号 (Zero flag) 为 On。
2. 演算结果小于 -2,147,483,648 时, 借位旗号 (Borrow flag) 为 On。
3. 演算结果大于 2,147,483,647 时, 进位旗号 (Carry flag) 为 On。

程式范例 (一)

- ◆ 16 位 BIN 加法: 当 X0=On 时, 被加数 D0 内容加上加数 D10 之内容将结果存在 D20 之内容当中。



5 应用指令分类及基本使用

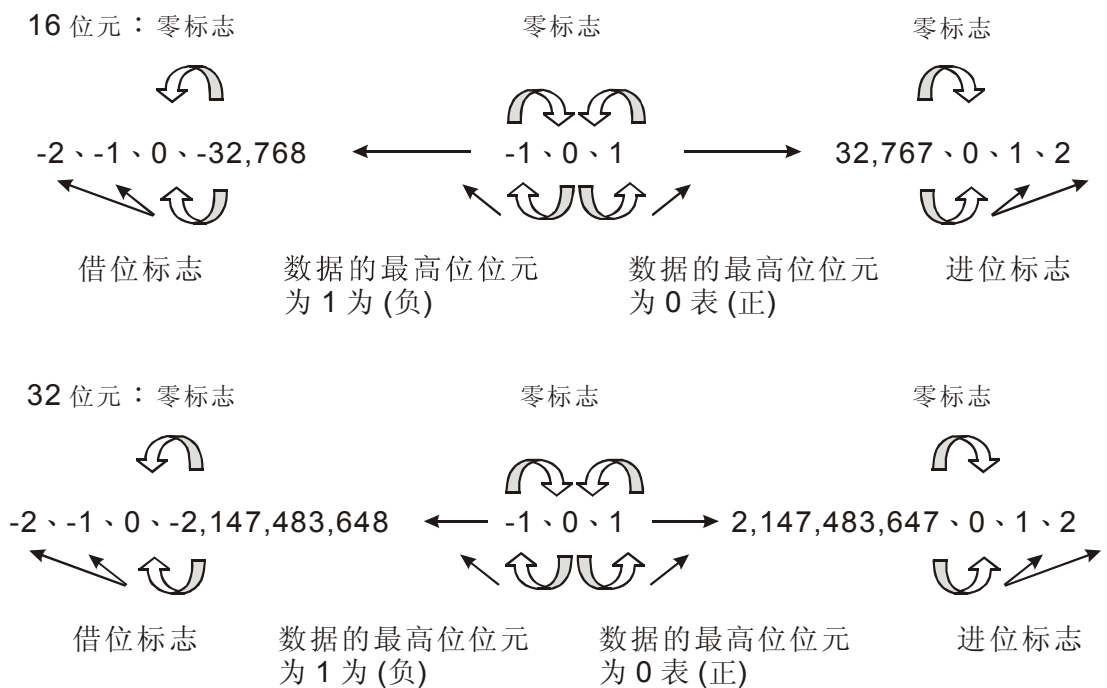
程式范例 (二)

- ◆ 32 位 BIN 加法：当 X1=On 时，被加数 (D31、D30) 内容加上加数 (D41、D40) 之内容将结果存在 (D51、D50) 之中。(其中 D30、D40、D50 为低 16 位数据，D31、D41、D51 为高 16 位数据)



补充说明

- ◆ 旗号动作与数值的正负关系：



API																适用机种		
21	D	SUB	P	(S1)	(S2)	(D)	BIN 减法										20PM	10PM
																	✓	✓
	位装置				字符装置												16 位指令 (7 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SUB 连续执行型 SUBP 脉冲执行型		
S ₁					*	*	*	*	*	*	*	*	*	*	*			
S ₂					*	*	*	*	*	*	*	*	*	*	*			
D								*	*	*	*	*	*	*	*			
<ul style="list-style-type: none"> 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 																32 位指令 (9 STEP) DSUB 连续执行型 DSUBP 脉冲执行型		
<ul style="list-style-type: none"> 旗标信号 Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 																<ul style="list-style-type: none"> 请参考下列补充说明 		

指令说明

- ◆ S₁: 被减数。S₂: 减数。D: 差。
- ◆ 将两个资料来源: S₁ 及 S₂ 以 BIN 方式相减的结果存于 D。
- ◆ 各数据的最高位位为符号位 0 表 (正) 1 表 (负), 因此可做代数减法运算。
- ◆ 减法相关旗号变化。

16 位 BIN 减法:

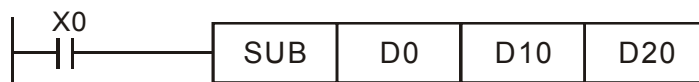
1. 演算结果为 0 时, 零旗号 (Zero flag) 为 On。
2. 演算结果小于 -32,768 时, 借位旗号 (Borrow flag) 为 On。
3. 演算结果大于 32,767 时, 进位旗号 (Carry flag) 为 On。

32 位 BIN 减法:

1. 演算结果为 0 时, 零旗号 (Zero flag) 为 On。
2. 演算结果小于 -2,147,483,648 时, 借位旗号 (Borrow flag) 为 On。
3. 演算结果大于 2,147,483,647 时, 进位旗号 (Carry flag) 为 On。

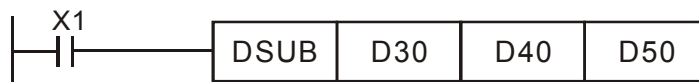
- ◆ 旗号动作与数值的正负关系参考旗号动作与数值的正负关系请参考上页指令 ADD 之补充说明。

程式范例 (一)



程式范例 (二)

- ◆ 32 位 BIN 减法: 当 X1=On 时, (D31、D30) 内容减掉 (D41、D40) 之内容后, 将差存在 (D51、D50) 之中。(其中 D30、D40、D50 为低 16 位数据, D31、D41、D51 为高 16 位数据)

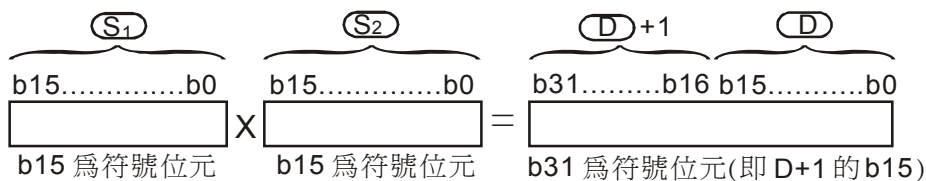


5 应用指令分类及基本使用

API																		适用機種		
22	D	MUL	P	(S1)	(S2)	(D)	BIN 乘法										20PM	10PM		
																		✓	✓	
	位裝置				字裝置												16 位指令 (7 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	MUL 連續執行型		MULP 脈衝執行型		
S1					*	*	*	*	*	*	*	*	*	*	*					
S2					*	*	*	*	*	*	*	*	*	*	*					
D											*	*	*							
• 操作數使用注意: 此指令有支持 V、Z 裝置 (當 16 位指令時不能使用 Z 裝置; 當 32 位指令時不能使用 V 裝置) 各裝置使用範圍請參考功能規格表																• 旗標信號: 無				

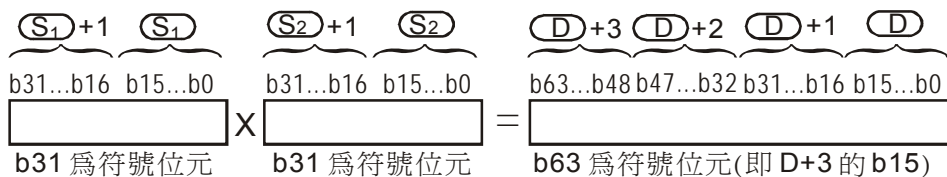
指令說明

- ◆ **S1**: 被乘數。 **S2**: 乘數。 **D**: 積。
- ◆ 將兩個資料源: **S1** 及 **S2** 以有號數二進制方式相乘後的積存於 **D**。必須注意 16 位及 32 位運算時, **S1**、**S2** 及 **D** 的正負號位。
- ◆ 16 位 BIN 乘法運算:



符號位=0 為正數, 符號位=1 為負數。

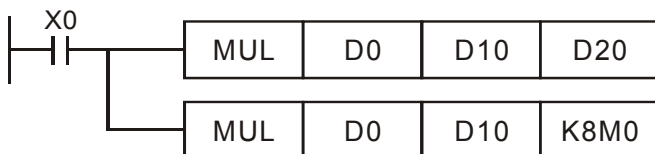
- ◆ 32 位 BIN 乘法運算:



符號位=0 為正數, 符號位=1 為負數。

程式範例

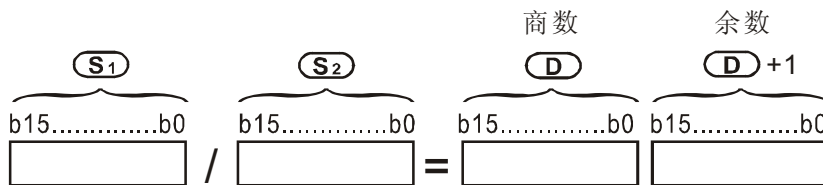
- ◆ 16 位 D0 乘上 16 位 D10 其結果是 32 位之積, 上 16 位存於 D21, 下 16 位存於 D20 內, 結果之正負由最左邊位之 Off/On 來代表正或負值。



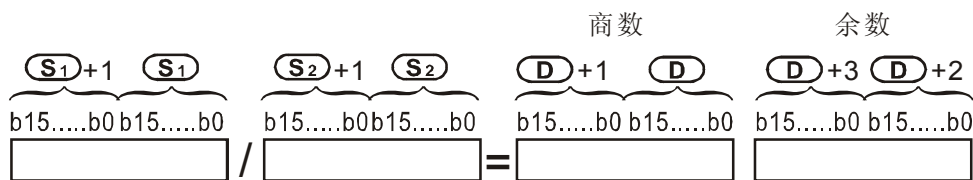
API																适用機種		
23	D	DIV	P	(S ₁)	(S ₂)	(D)	BIN 除法										20PM	10PM
																	✓	✓
	位装置				字符装置										16 位指令 (7 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	DIV	DIVP	
S ₁					*	*	*	*	*	*	*	*	*	*	*			
S ₂					*	*	*	*	*	*	*	*	*	*	*			
D											*	*	*					
• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表																	• 旗标信号：无	

指令说明

- ◆ S₁: 被除数。 S₂: 除数。 D: 商及余数。
- ◆ 将两个资料来源: S₁ 及 S₂ 以有号数二进制方式相除后的商及余数存于 D。必须注意 16 位及 32 位运算时, S₁、S₂ 及 D 的正负号位。
- ◆ 除数为 0 时, 指令不执行。
- ◆ 16 位 BIN 除法运算:

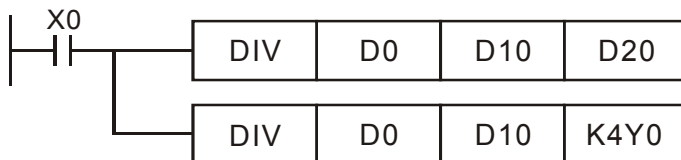


- ◆ 32 位 BIN 除法运算:



程式范例

- ◆ 当 X0=On 时, 被除数 D0 除以除数 D10 而结果商被指定放于 D20, 余数指定放于 D21 内。所得结果之正负由最高位位之 Off/On 来代表正或负值。



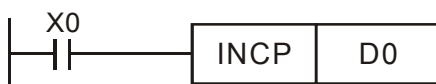
5 应用指令分类及基本使用

API																		适用機種		
24	D	INC	P	(D)	BIN 加一													20PM	10PM	
																		✓	✓	
D	位装置				字符装置												16 位指令 (3 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	INC	连续执行型	INCP	脉冲执行型	
							*	*	*	*	*	*	*	*	*	32 位指令 (3 STEP)				
																DINC	连续执行型	DINCP	脉冲执行型	
• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表																• 旗标信号：无				

指令说明

- ◆ **D**：目的地装置。
- ◆ 若指令不是脉冲执行型，则当指令执行时，程序每次扫描周期被指定的装置 **D** 内容都会加 1。
- ◆ 本指令一般都是使用脉冲执行型指令 (**INCP**、**DINCP**)。
- ◆ 16 位运算时，32,767 再加 1 则变为-32,768。32 位运算时，2,147,483,647 再加 1 则变为-2,147,483,648。

程式范例



API																	适用機種		
25	D	DEC	P															20PM	10PM
																		✓	✓
	位裝置				字裝置												16 位指令 (3 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	DEC 連續執行型 DECP 脈衝執行型			
D							*	*	*	*	*	*	*	*	*	32 位指令 (3 STEP)			
	<ul style="list-style-type: none"> 操作數使用注意：此指令有支持 V、Z 裝置 (當 16 位指令時不能使用 Z 裝置；當 32 位指令時不能使用 V 裝置) 各裝置使用範圍請參考功能規格表 															DDEC 連續執行型 DDECP 脈衝執行型			
																<ul style="list-style-type: none"> 旗標信號：無 			

指令說明

- ◆ **D**：目的地裝置。
- ◆ 若指令不是脈衝執行型，當指令執行時，程序每次掃描週期被指定的裝置 **D** 內容都會減 1。
- ◆ 本指令一般都是使用脈衝執行型指令 (**DECP**、**DDECP**)。
- ◆ 16 位運算時，-32,768 再減 1 則變為 32,767。32 位運算時，-2,147,483,648 再減 1 則變為 2,147,483,647。

程式範例



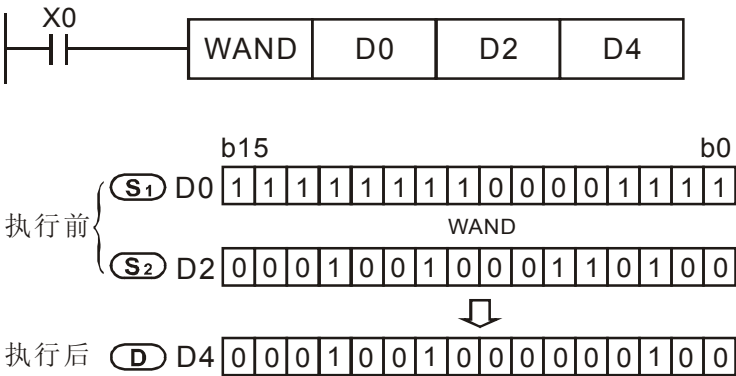
5 应用指令分类及基本使用

API																适用機種			
26	D	WAND	P	(S1)	(S2)	(D)	逻辑及(AND)运算										20PM	10PM	
																	✓	✓	
	位装置				字符装置										16 位指令 (7 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WAND	连续执行型	WANDP	脉冲执行型
S1					*	*	*	*	*	*	*	*	*	*	*				
S2					*	*	*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*	*	*				
• 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表																• 旗标信号: 无			

指令说明

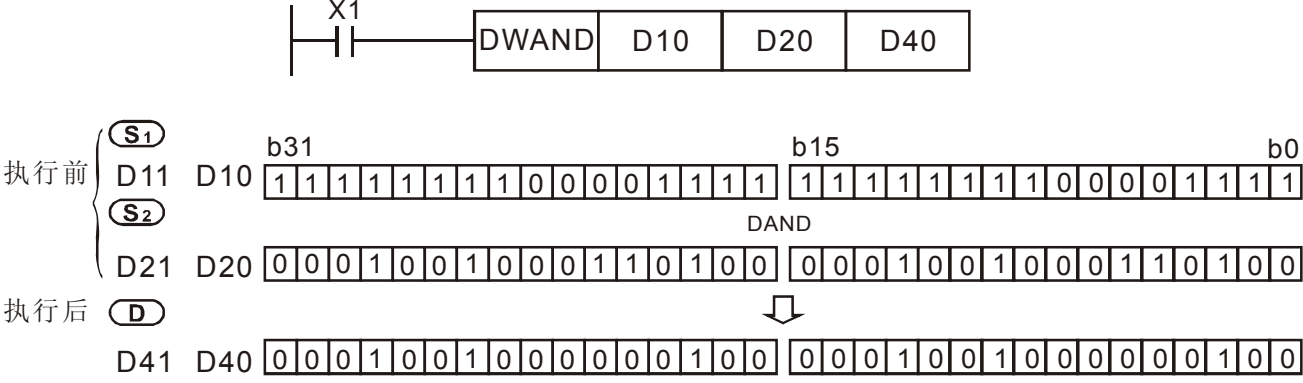
- ◆ **S1**: 数据来源装置 1。 **S2**: 数据来源装置 2。 **D**: 运算结果。
- ◆ 两个资料来源: **S1** 及 **S2** 作逻辑的'及' (AND) 运算并将结果存于 **D**。
- ◆ 逻辑的'及' (AND) 运算之规则为任一为 0 结果为 0。
- ◆ 当 X0=On 时, 16 位 D0 与 D2 作 WAND, 逻辑及 (AND) 运算, 将结果存于 D4 中。

程式范例 (一)



程式范例 (二)

- ◆ 当 X1=On 时, 32 位 (D11、D10) 与 (D21、D20) 作 DWAND, 逻辑及 (AND) 运算, 将结果存于 (D41、D40) 中。

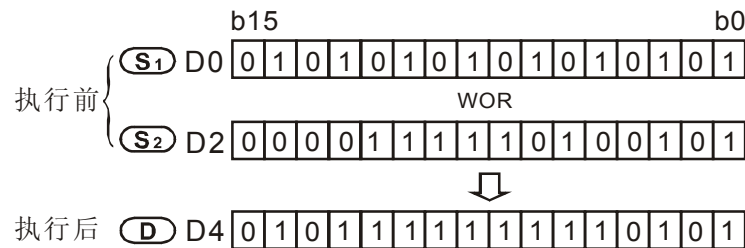
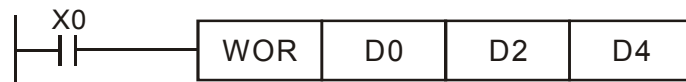


API																适用机种			
27	D	WOR	P	S1	S2	D	逻辑或(OR)运算										20PM	10PM	
							✓	✓											
	位装置				字符装置										16 位指令 (7 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WOR	连续执行型	WORP	脉冲执行型
S1					*	*	*	*	*	*	*	*	*	*	*				
S2					*	*	*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*	*	*				
• 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表																• 旗标信号: 无			

指令说明

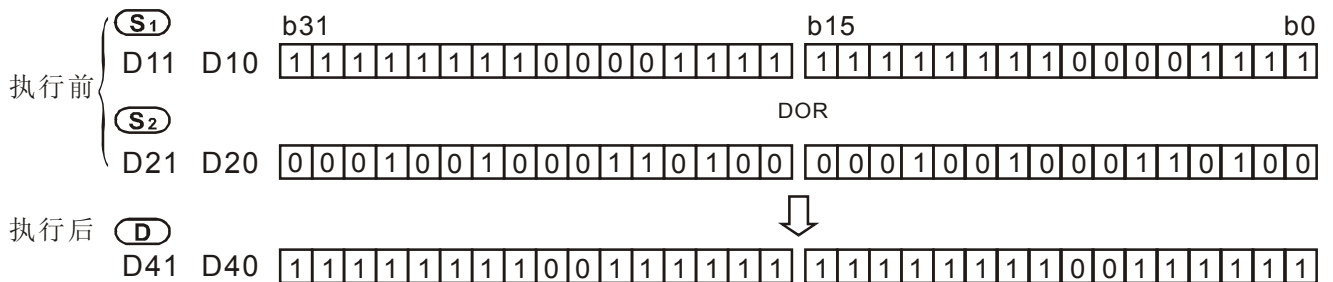
- ◆ **S1**: 数据来源装置 1。 **S2**: 数据来源装置 2。 **D**: 运算结果。
- ◆ 两个资料来源: **S1** 及 **S2** 作逻辑的'或' (OR) 运算结果存于 **D**。
- ◆ 逻辑的'或' (OR) 运算之规则为任一为 1 结果为 1。
- ◆ 当 X0=On 时, 16 位 D0 与 D2 作 WOR, 逻辑或 (OR) 运算, 将结果存于 D4 中。

程式范例 (一)



程式范例 (二)

- ◆ 当 X1=On 时, 32 位 (D11、D10) 与 (D21、D20) 作 DWOR, 逻辑或 (OR) 运算, 将结果存于 (D41、D40) 中。



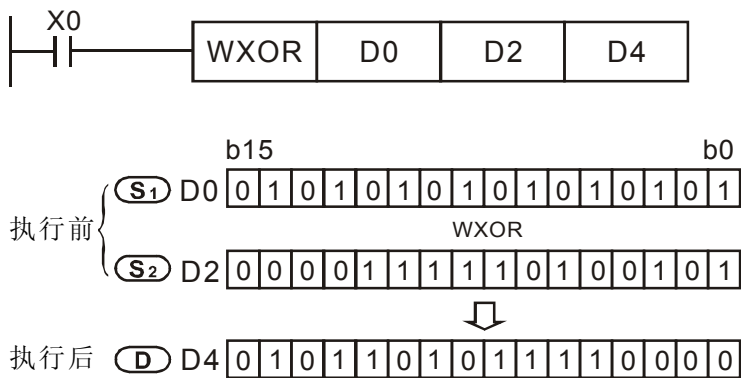
5 应用指令分类及基本使用

API 28	D	WXOR	P	(S1) (S2) (D)	逻辑互斥或(XOR)运算										适用机种	
															20PM	10PM
															✓	✓
位装置		字符装置										:16 位指令 (7 STEP)				
X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WXOR 连续执行型	WXORP 脉冲执行型
S1				*	*	*	*	*	*	*	*	*	*	*	:32 位指令 (9 STEP)	
S2				*	*	*	*	*	*	*	*	*	*	*	DWXOR 连续执行型	DWXORP 脉冲执行型
D							*	*	*	*	*	*	*	*	• 旗标信号: 无	
• 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表																

指令说明

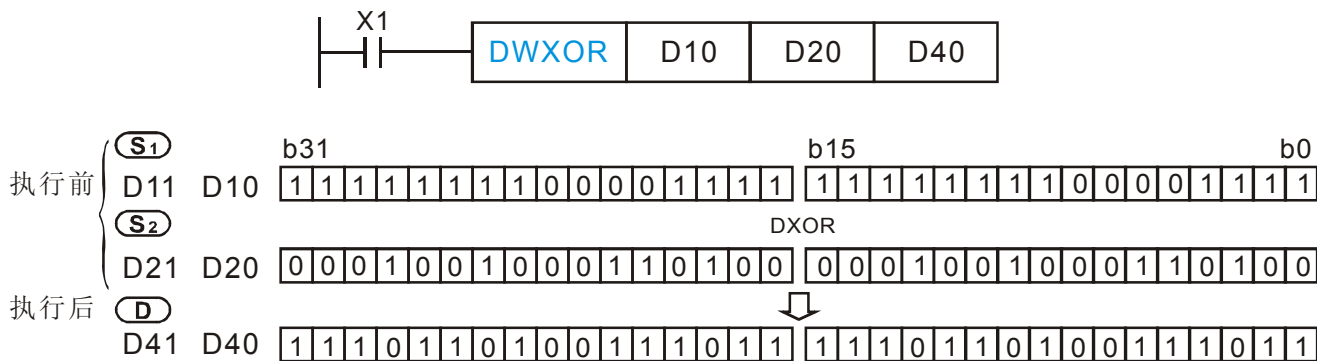
- ◆ **S1**: 数据来源装置 1。 **S2**: 数据来源装置 2。 **D**: 运算结果。
- ◆ 两个资料来源: **S1** 及 **S2** 作逻辑的'互斥或' (XOR) 运算结果存于 **D**。
- ◆ 逻辑的'互斥或' (XOR) 运算之规则为两者相同结果为 0, 两者不同结果为 1。
- ◆ 当 X0=On 时, 16 位 D0 与 D2 作 WXOR, 逻辑互斥或 (XOR) 运算, 将结果存于 D4 中。

程式范例 (一)



程式范例 (二)

- ◆ 当 X1=On 时, 32 位 (D11、D10) 与 (D21、D20) 作 DXOR, 逻辑互斥或 (XOR) 运算, 将结果存于 (D41、D40) 中。



API	W	NEG		(D)	2 的补码										适用机种			
29	D		P												20PM	10PM		
													✓	✓				
位装置				字符装置											16 位指令 (7 STEP)			
X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	NEG	连续执行型	NEGP	脉冲执行型
D							*	*	*	*	*	*	*	*	32 位指令 (9 STEP)			
• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表														DNEG		连续执行型	DNEGP	脉冲执行型
														• 旗标信号：无				

指令说明

◆ **D**：欲取 2 的补码之装置。本指令可将负数的 BIN 值转换成绝对值。

◆ 本指令一般都是使用脉冲执行型指令 (NEGP、DNEGP)。

程式范例 (一)

◆ 当 X0=Off→On 时，D10 内容的各位全部反相 (0→1、1→0) 后再加 1 存放于原寄存器 D10 当中。

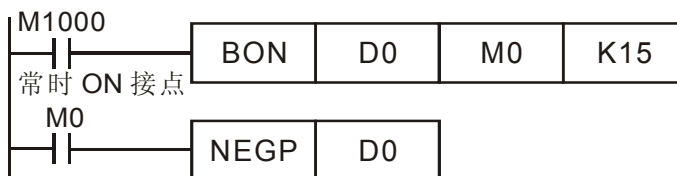


程式范例 (二)

◆ 求负数的绝对值：

1. 当 D0 的 b15 为“1”时，M0=On。(D0 表示为负数)

2. M0=On 时，用 NEG 指令将 D0 取 2 的补码可得到其绝对值。



程式范例 (三)

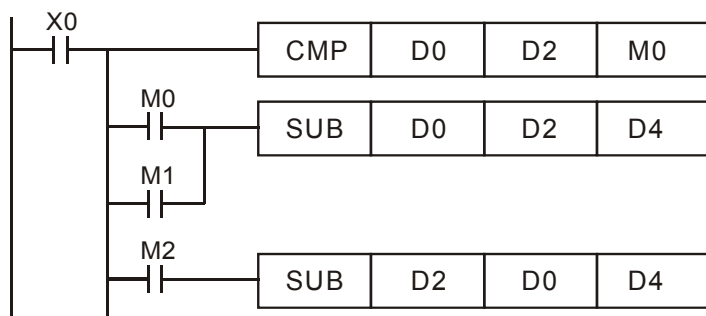
◆ 减法运算之差取绝对值，当 X0=On 时：

1. 若 D0>D2 时，M0=On。

2. 若 D0=D2 时，M1=On。

3. 若 D0<D2 时，M2=On。

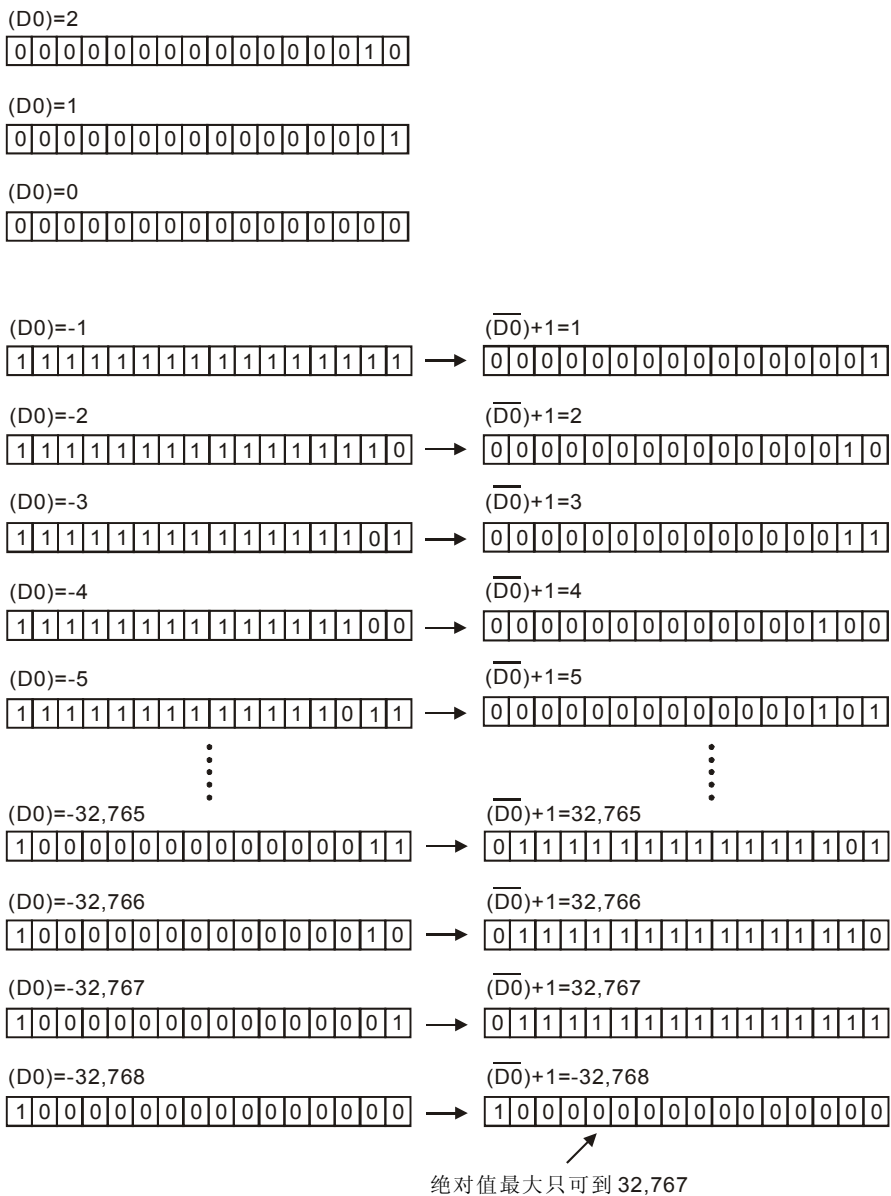
4. 此可得 D4 保持为正值。



补充说明

◆ 负数的表现及绝对值

1. 正负数是以寄存器最上位（最左边）的位内容来表现，为“0”时，为正数、为“1”时，为负数。
2. 遇到负数时，可使用 **NEG** 指令将它转成绝对值。

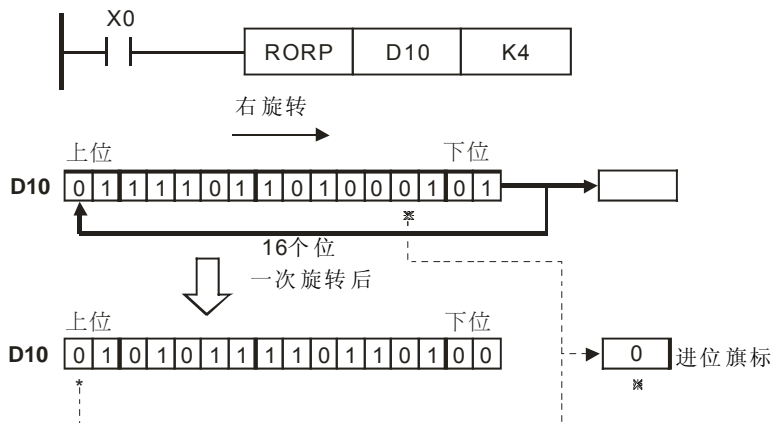


API																	适用機種			
30	D	ROR	P	(D)	(n)	右旋转											20PM	10PM		
																	✓	✓		
		位装置				字符装置											16位指令 (5 STEP)			
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ROR	连续执行型	RORP	脉冲执行型
D								*	*	*	*	*	*	*	*	*				
N						*	*													
<ul style="list-style-type: none"> 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时, 限于 8 进制或 10 进制中 16 的倍数为起始编号。如: K1X0, K4Y20 (八进制); K1M0, K4S16 (十进制). 																	<ul style="list-style-type: none"> 旗标信号 Ox O100 M1810 M1970 进位旗标 Carry flag 请参考下列补充说明 			

指令说明

- ◆ **D**: 欲旋转之装置。**n**: 一次旋转之位数。
- ◆ 将 **D** 所指定的装置内容一次向右旋转 **n** 个位。
- ◆ 最后旋转的一位的状态被传送到进位旗标。
- ◆ 本指令一般都是用脉冲执行型指令 (RORP, DRORP)。
- ◆ 如果操作数 **D** 指定为 KnY, KnM, KnS 时, 只有 K4 (16 位)及 K8 (32 位)有效。
- ◆ **n** 操作数有效范围: $1 \leq n \leq 16$ (16 位), $1 \leq n \leq 32$ (32 位)。
- ◆ 当 X0 从 Off→On 变化时, D10 的 16 个位以 4 个位为一组往右旋转, 如下图所示标明*的位内容被传送至进位旗标信号内。

程式范例



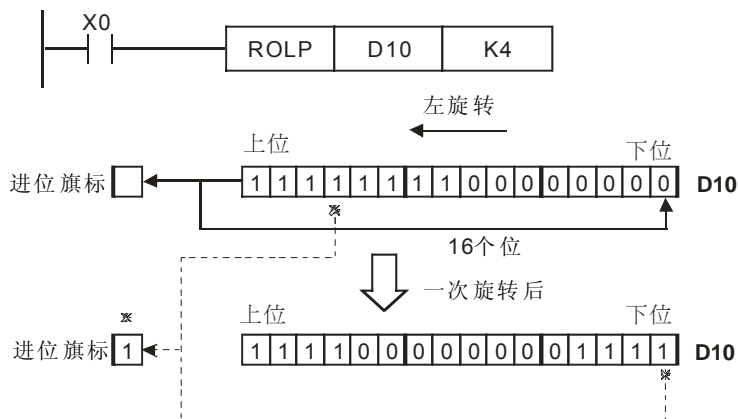
5 应用指令分类及基本使用

API																	适用機種			
31	D	ROL	P	D n		左旋转										20PM	10PM			
				✓	✓															
		位装置				字符装置										16 位指令 (5 STEP)				
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ROL	连续执行型	ROLP	脉冲执行型
D								*	*	*	*	*	*	*	*	*	32 位指令 (9 STEP)			
n					*	*											DROL	连续执行型	DROLP	脉冲执行型
<ul style="list-style-type: none"> 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时, 限以 8 进制或 10 进制中 16 的倍数为起始编号。如: K1X0, K4Y20 (八进制); K1M0, K4S16 (十进制)。 																	<ul style="list-style-type: none"> 旗标信号 Ox O100 M1810 M1970 进位旗标 Carry flag 请参考下列补充说明 			

指令说明

- ◆ **D**: 欲旋转之装置。**n**: 一次旋转之位数。
- ◆ 将 **D** 所指定的装置内容一次向左旋转 **n** 个位。
- ◆ 最后旋转的一位的状态被传送到进位旗标 (进位旗标)。
- ◆ 本指令一般都是用脉冲执行型指令 (ROLP, DROLP)。
- ◆ 如果操作数 **D** 指定为 KnY, KnM, KnS 时, 只有 K4 (16 位)及 K8 (32 位)有效。
- ◆ **n** 操作数有效范围: $1 \leq n \leq 16$ (16 位), $1 \leq n \leq 32$ (32 位)。
- ◆ 当 X0 从 Off→On 变化时, D10 的 16 个位以 4 个位一组往左旋转, 如下图所示标明※的位内容被传送至进位旗标信号内。

程式范例

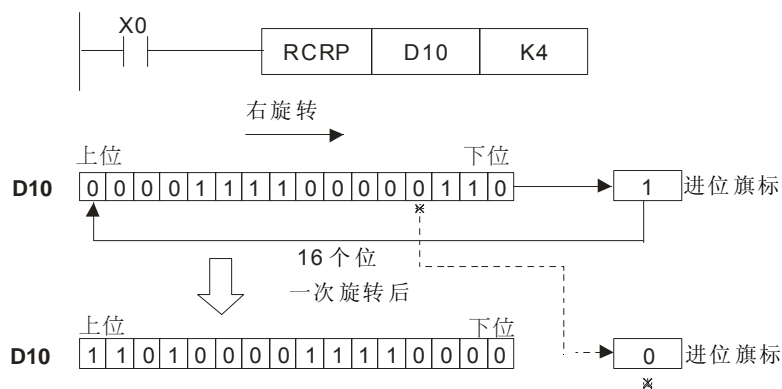


API																	适用机种		
32	D	RCR	P	(D)	(n)	附进位旗标右旋转											20PM	10PM	
																		✓	✓
	位装置				字符装置											16 位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	RCR 连续执行型 RCRP 脉冲执行型			
D							*	*	*	*	*	*	*	*	*				
n					*	*										32 位指令 (9 STEP)			
																DRCR 连续执行型 DRCP 脉冲执行型			
<ul style="list-style-type: none"> 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时, 限以 8 进制或 10 进制中 16 的倍数为起始编号。如: K1X0, K4Y20 (八进制); K1M0, K4S16 (十进制) 																<ul style="list-style-type: none"> 旗标信号 Ox O100 M1810 M1970 进位旗标 Carry flag 请参考下列补充说明 			

指令说明

- ◆ **D**: 欲旋转之装置。**n**: 一次旋转之位数。
- ◆ 将 **D** 所指定的装置内容连同进位旗标, 一次向右旋转 **n** 个位。
- ◆ 最后旋转的一位的状态被传送到进位旗标 (进位旗标)。在下面的指令操作时, 进位旗标第一个被传送到目的的设备。
- ◆ 本指令一般都是用脉冲执行型指令 (RCRP, DRCP)。
- ◆ 如果操作数 **D** 指定为 KnY, KnM, KnS 时, 只有 K4 (16 位)及 K8 (32 位)有效。
- ◆ **n** 操作数有效范围: $1 \leq n \leq 16$ (16 位), $1 \leq n \leq 32$ (32 位)。
- ◆ 当 X0 从 Off→On 变化时, D10 的 16 个位连同进位旗标共 17 个位以 4 个位为一组往右旋转, 如下图所示标明*的位内容被传送至进位旗标信号内。

程式范例



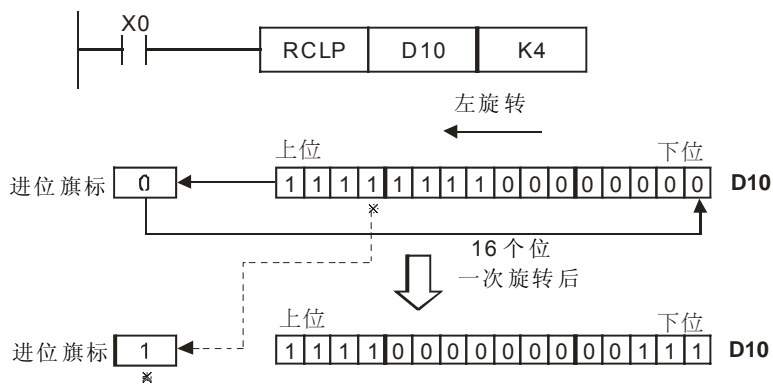
5 应用指令分类及基本使用

API																		适用機種	
33	D	RCL	P	(D)	(n)	附进位旗标左旋转												20PM	10PM
																		✓	✓
	位装置				字符装置												16 位指令 (5 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	RCL 连续执行型 RCLP 脉冲执行型			
D								*	*	*	*	*	*	*	*				
n					*	*										32 位指令 (9 STEP)			
																DRCL 连续执行型 DRCLP 脉冲执行型			
<p>• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)</p> <p>• 旗标信号 Ox O100 M1810 M1970 进位旗标 Carry flag</p> <p>• 请参考下列补充说明</p>																			

指令说明

- ◆ **D**：欲旋转之装置。**n**：一次旋转之位数。
- ◆ 将 **D** 所指定的装置内容连同进位旗标，一次向左旋转 **n** 个位。
- ◆ 最后旋转的一位的状态被传送到进位旗。在下面的指令操作时，M1022 第一个被传送到目的的设备。
- ◆ 本指令一般都是用脉冲执行型指令 (RCLP, DRCLP)。
- ◆ 如果操作数 **D** 指定为 KnY, KnM, KnS 时，只有 K4 (16 位)及 K8 (32 位)有效。
- ◆ **n** 操作数有效范围: $1 \leq n \leq 16$ (16 位), $1 \leq n \leq 32$ (32 位)。
- ◆ 当 X0 从 Off→On 变化时，D10 的 16 个位连同进位旗标共 17 个位以 4 个位一组往左旋转，如下图所示标明*的位内容被传送至进位旗标信号内。

程式范例



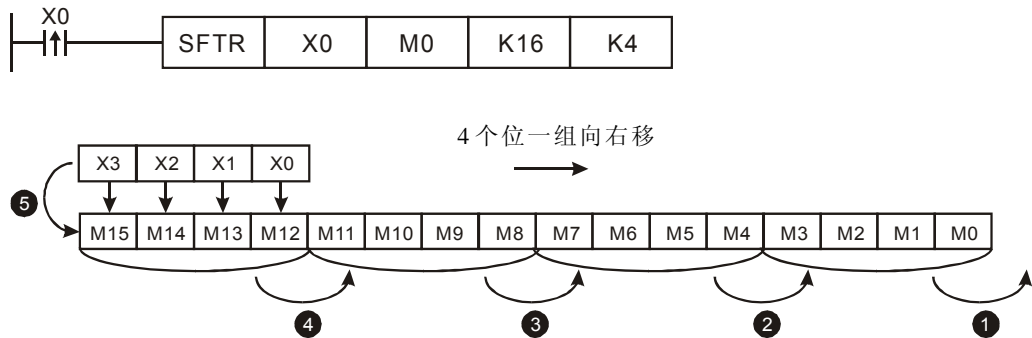
API																	适用机种			
34	SFTR	P	S	D	n1	n2	位右移										20PM	10PM		
																	✓	✓		
		位装置				字符装置										16 位指令 (9 STEP)				
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SFTR	连续执行型	SFTRP	脉冲执行型
S		*	*	*	*												32 位指令			
D			*	*	*												— — — —			
n1						*	*										• 旗标信号: 无			
n2						*	*													
<ul style="list-style-type: none"> 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 																				

指令说明

- ◆ **S**: 移位装置之起始编号。 **D**: 欲移位装置之起始编号。 **n₁**: 欲移位之数据长度。 **n₂**: 一次移位的位数。
- ◆ 将 **D** 开始之起始编号, 具有 **n₁** 个数字元 (位移寄存器长度) 的位装置, 以 **n₂** 位个数来右移。而 **S** 开始起始编号以 **n₂** 位个数移入 **D** 中来填补位空位。
- ◆ 本指令一般都是使用脉冲执行型指令(SFTRP)。
- ◆ **n₁**, **n₂** 操作数有效范围: $1 \leq n_2 \leq n_1 \leq 1024$ 。

程式范例

- ◆ 当 X0 Off → On 时, 由 M0~M15 组成 16 位, 以 4 位作右移。
- ◆ 扫描一次的位左移依照下列编号 ①~⑤ 动作。
 - ① M3~M0 → 进位
 - ② M7~M4 → M3~M0
 - ③ M11~M8 → M7~M4
 - ④ M15~M12 → M11~M8
 - ⑤ X3~X0 → M15~M12 完成



5 应用指令分类及基本使用

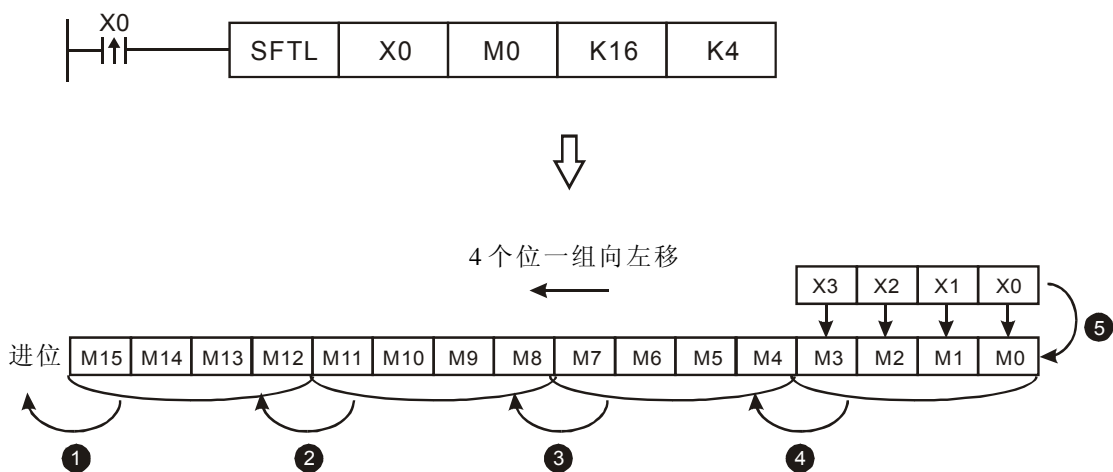
API																	适用机种	
35	SFTL	P	S	D	n1	n2	位左移										20PM	10PM
																	✓	✓
	位装置				字符装置												16 位指令 (9 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SFTL 连续执行型 SFTLP 脉冲执行型		
S	*	*	*	*												32 位指令		
D		*	*	*												— — — —		
n1					*	*										• 旗标信号: 无		
n2					*	*												
<ul style="list-style-type: none"> 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置); 各装置使用范围请参考功能规格表 																		

指令说明

- ◆ **S**: 移位装置之起始编号。**D**: 欲移位装置之起始编号。**n1**: 欲移位之数据长度。**n2**: 一次移位的位数。
- ◆ 将 **D** 开始之起始编号, 具有 **n1** 个数字元 (位移寄存器长度) 的位装置, 以 **n2** 位个数来左移。而 **S** 开始起始编号以 **n2** 位个数移入 **D** 中来填补位空位。
- ◆ 本指令一般都是使用脉冲执行型指令(SFTRP)。
- ◆ **n1, n2** 操作数有效范围: $1 \leq n_2 \leq n_1 \leq 1024$ 。

程式范例

- ◆ 当 X0 Off → On 时, 由 M0~M15 组成 16 位, 以 4 位作左移。
- ◆ 扫描一次的位左移依照下列编号 ①~⑤ 动作。
 - ① M15~M12 → 进位
 - ② M11~M8 → M15~M12
 - ③ M7~M4 → M11~M8
 - ④ M3~M0 → M7~M4
 - ⑤ X3~X0 → M3~M0 完成



API		WSFR			(S) (D) (n1) (n2)	寄存器右移	适用机种	
36			P				20PM	10PM
							✓	✓

	位装置				字符装置										16位指令 (9 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WSFR	WSFRP
S							*	*	*	*	*	*	*				
D								*	*	*	*	*	*				
n1					*	*											
n2					*	*											

• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置)
 各装置使用范围请参考功能规格表
 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)

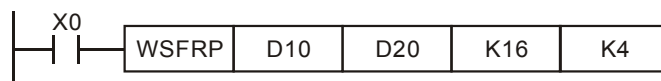
16 位指令 (9 STEP)
 WSFR 连续执行型 WSFRP 脉冲执行型
 32 位指令
 — — — —
 • 旗标信号：无

指令说明

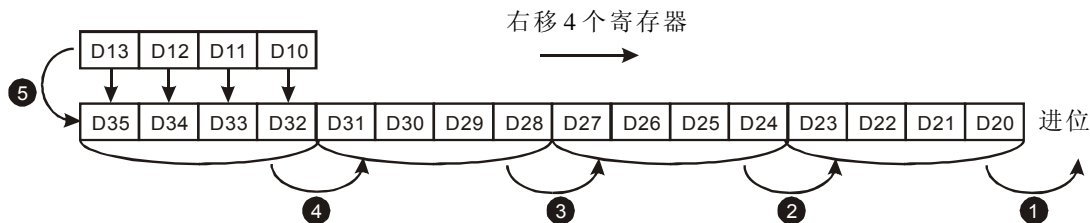
- ◆ **S**：移位装置之起始编号。**D**：欲移位装置之起始编号。**n1**：欲移位之数据长度。**n2**：一次移位之字符数。
- ◆ 将 **D** 开始之起始编号，具有 **n1** 个字符长度的数据串行，以 **n2** 个字符来右移。而 **S** 开始起始编号以 **n2** 字符个数移入 **D** 中来填补字符空位。
- ◆ 本指令一般都是使用脉冲执行型指令(WSFRP)。
- ◆ 当操作数 **S**, **D** 使用位数据类型时，数据类型必须是匹配的。例如，一个类型是 KnX, KnY, KnM, KnS 另一个类型是 T, C, D。
- ◆ 当使用操作数 **S** 和 **D** 位数据类型时，Kn 的值必须是匹配的。
- ◆ **n1, n2** 操作数有效范围: $1 \leq n2 \leq n1 \leq 512$

程式范例 (一)

- ◆ 当 X0 从 Off → On 时，由 D20~D35 所组成的 16 个寄存器数据串行为位移区域，以 4 个寄存器来右移。
- ◆ 扫描一次的字右移行作依照下列编号 ①~⑤ 动作。
 - ① D23~D20 → 进位
 - ② D27~D24 → D23~D20
 - ③ D31~D28 → D27~D24
 - ④ D35~D32 → D31~D28
 - ⑤ D13 ~D10 → D35~D32 完成

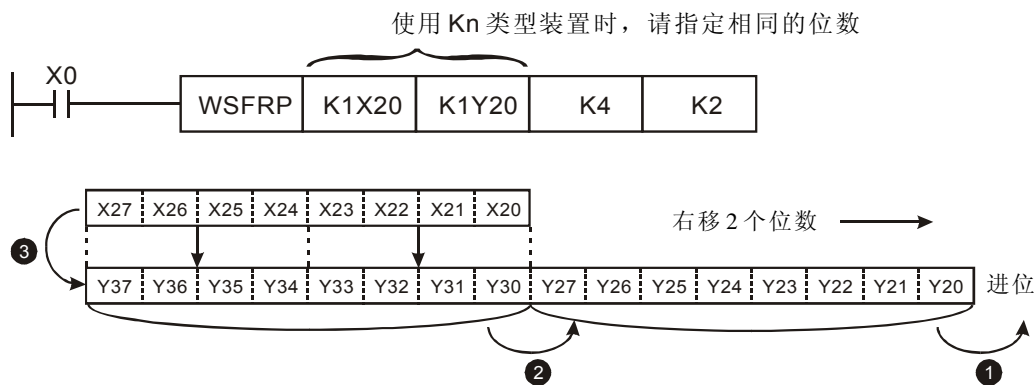


5 应用指令分类及基本使用



程式范例
(二)

- ◆ 当 X0 从 Off → On 时，由 Y20~Y37 所组成的位寄存器数据串行为位移区域，以 2 个位数来右移。
- ◆ 扫描一次的字右移行作依照下列编号 ①~③ 动作。
 - ① Y27~Y20 → 进位
 - ② Y37~Y30 → Y27~Y20
 - ③ X27~X20 → Y37~Y30 完成



API		WSFL		S	D	n1	寄存器左移							适用机种	
37			P			n2								20PM	10PM
													✓	✓	

	位装置				字符装置											16 位指令 (9 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WSFL	连续执行型	WSFLP	脉冲执行型
S							*	*	*	*	*	*	*						
D								*	*	*	*	*	*						
n1					*	*													
n2					*	*													

• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置)
 各装置使用范围请参考功能规格表
 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。

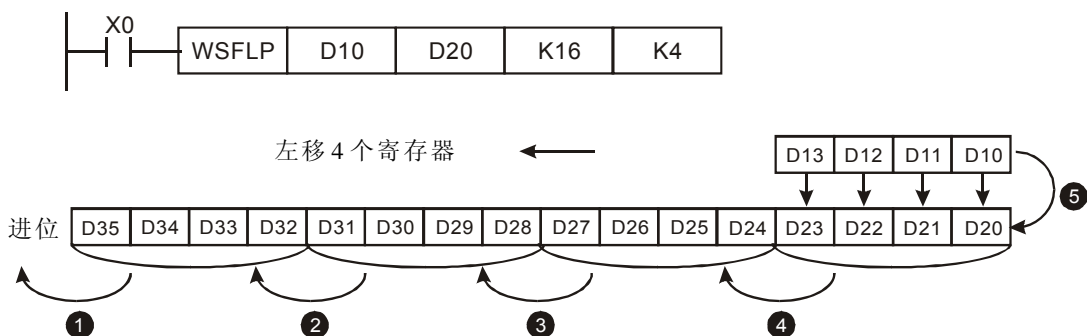
• 旗标信号：无

指令说明

- ◆ **S**：移位装置之起始编号。**D**：欲移位装置之起始编号。**n1**：欲移位之数据长度。**n2**：一次移位之字符数。
- ◆ 将 **D** 开始之起始编号，具有 **n1** 个字符长度的数据串行，以 **n2** 个字符来左移。而 **S** 开始起始编号以 **n2** 字符个数移入 **D** 中来填补字符空位。
- ◆ 本指令一般都是使用脉冲执行型指令(WSFLP)。
- ◆ 当操作数 **S**、**D** 使用位数据类型时，数据类型必须是匹配的。例如，一个类型是 KnX, KnY, KnM, KnS 另一个类型是 **T, C, D**。
- ◆ 当使用操作数 **S** 和 **D** 位数据类型时，Kn 的值必须是匹配的。
- ◆ **n1, n2** 操作数有效范围：1 ≤ n2 ≤ n1 ≤ 512

程式范例

- ◆ 当 X0 从 Off → On 时，由 D20~D35 所组成的 16 个寄存器数据串行为位移区域，以 4 个寄存器来左移。
- ◆ 扫描一次的字左移行作依照下列编号 ①~⑤ 动作。
 - ① D35~D32 → 进位
 - ② D31~D28 → D35~D32
 - ③ D27~D24 → D31~D28
 - ④ D23~D20 → D27~D24
 - ⑤ D13~D10 → D23~D20 完成



5 应用指令分类及基本使用

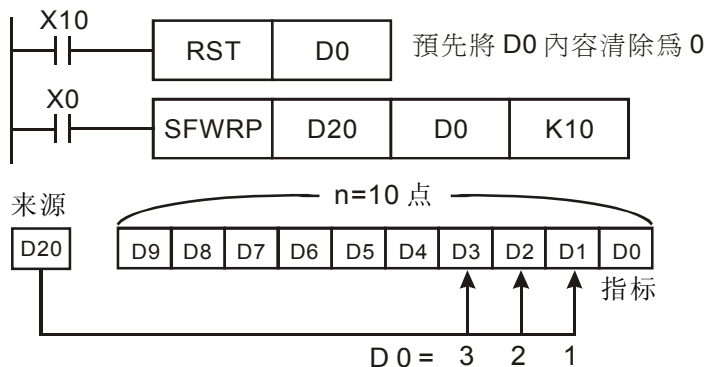
API																	适用机种		
38		SFWR	P	(S)	(D)	(n)	位移写入										20PM	10PM	
																		✓	✓
	位装置				字符装置										16 位指令 (9 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SFWR 连续执行型 SFWRP 脉冲执行型			
S					*	*	*	*	*	*	*	*	*	*	*	32 位指令			
D								*	*	*	*	*	*			— — — —			
n					*	*										旗标信号			
<ul style="list-style-type: none"> 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。 																Ox O100 M1808 M1968 零旗标信号 Zero flag			

指令说明

- ◆ **S**：位移写入数据串行之装置。**D**：数据串行之起始编号。**n**：数据串行之长度。
- ◆ 将 **D** 起始编号开始 **n** 个字符装置的数据串行定义为先入先出数据串行，以第一个编号装置作为指针，当指令执行时，指针内容值先加 1，之后 **S** 所指定的装置其内容值会写入先入先出数据串行中由指针所指定的位置，当指针的内容超过 **n-1** 时，本指令不再处理写入的新值，进位旗标信号 **On**。
- ◆ 当指标的内容超过 **n-1** 时，本指令不再处理写入的新值，进位旗标信号 **M1022=On**。
- ◆ 本指令一般都是使用脉冲执行型指令 (SFWRP)。
- ◆ **n** 操作数有效范围: $2 \leq n \leq 512$

程式范例

- ◆ 开始先将指标 **D0** 清除为 0，当 **X0=Off → On** 变化时，**D20** 的内容被传送至 **D1** 当中，指标 **D0** 内容变成 1。变更 **D20** 的内容后，将 **X0** 再 **Off → On** 一次，则 **D20** 的内容被传送至 **D2** 当中，**D0** 内容变成 2。
- ◆ 指令执行一次位移写入动作依照下列编号 ①~②动作。
 - ① **D20** 的内容被传送至 **D1** 当中。
 - ② 指标 **D0** 内容变成 1。



补充说明

- ◆ 本指令 API38 SFWR 与 API39 SFRD 可搭配使用，执行先入先出数据串行的写入读出控制。

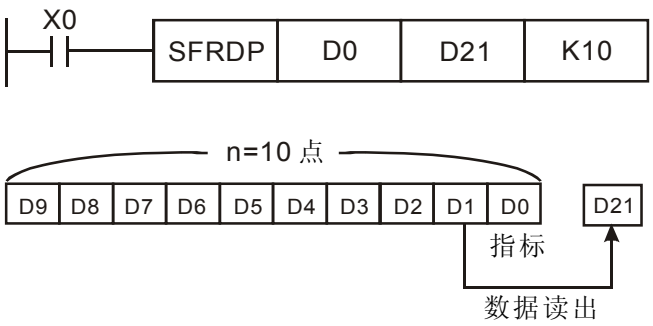
API																	适用机种			
39	SFRD	P	S	D	n	位移读出										20PM	10PM			
						✓	✓													
	位装置					字符装置										16 位指令 (9 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SFRD 连续执行型 SFRD P 脉冲执行型				
	S				*	*		*	*	*	*	*	*	*	*		32 位指令			
	D							*	*	*	*	*	*	*	*	*	— — — —			
n					*	*										旗标信号				
<ul style="list-style-type: none"> 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。 																	Ox O100 M1808 M1968 零旗标信号 Zero flag			

指令说明

- ◆ **S**：数据串行之起始编号。**D**：数据串行位移读出之装置。**n**：数据串行之长度。
- ◆ 将 **S** 起始编号开始 **n** 个字符装置的数据串行定义为先入先出数据串行，以第一个编号装置作为指针，当指令执行时，指针内容值先减 1，之后 **S** 所指定的装置其内容值会写入先入先出数据串行中由指针所指定的位置，当指针的内容等于 0 时，本指令不再处理数据读出的动作，零旗标信号 On。
- ◆ 当指标的内容等于 0 时，本指令不再处理数据读出的动作，零旗标信号 M1020=On。
- ◆ 本指令一般都是使用脉冲执行型指令 (SFRDP)。
- ◆ **n** 操作数有效范围: $2 \leq n \leq 512$

程式范例

- ◆ 当 X0 从 Off→On 变化时，D1 的内容被传送至 D21 内，D9~D2 全部往右位移一个寄存器(D9 内容保持不变)，指标 D0 内容减 1。
- ◆ 执行指令一次位移读出动作依照下列编号 ①~③动作。
 - ① D1 的内容被读出传送至 D21 当中。
 - ② D9~D2 全部往右位移一个寄存器。
 - ③ 指标 D0 内容减 1。



5 应用指令分类及基本使用

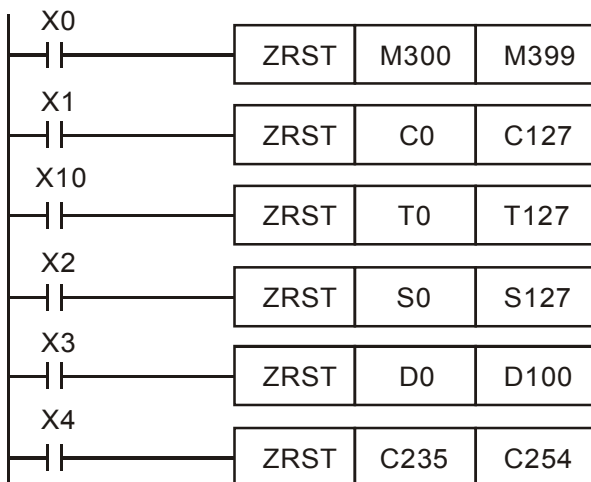
API			☺												适用機種				
40	ZRST	P	Ⓧ ₁	Ⓧ ₂	区域清除											20PM	10PM		
																✓	✓		
	位装置				字符装置											16 位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ZRST	连续执行型	ZRSTP	脉冲执行型
D ₁		*	*	*							*	*	*			32 位指令			
D ₂		*	*	*							*	*	*			— — — — —			
<ul style="list-style-type: none"> 操作数使用注意: D₁操作数编号 ≤ D₂操作数编号 D₁、D₂操作数必须指定相同类型装置 所有装置不支持 V、Z 修饰 各装置使用范围请参考功能规格表 																旗标信号: 无			

指令说明

- ◆ **D₁**: 区域清除起始装置。**D₂**: 区域清除结束装置。
- ◆ 16 位计数器与 32 位计数器可混在一起使用 ZRST 指令。
- ◆ 当 **D₁** 操作数编号 > **D₂** 操作数编号时, 只有 **D₂** 指定之操作数被清除。

程式范例

- ◆ 当 X0 为 On 时, 辅助继电器 M300 ~ M399 被清除成 Off。
- ◆ 当 X1 为 On 时, 16 位计数器 C0 ~ C127 全部清除。(写入 0, 并将接点及线圈清除成 Off)。
- ◆ 当 X10 为 On 时, 定时器 T0 ~ T127 全部清除。(写入 0, 并将接点及线圈清除成 Off)。
- ◆ 当 X2 为 On 时, 步进点 S0 ~ S127 被清除成 Off。
- ◆ 当 X3 为 On 时, 数据寄存器 D0 ~ D100 数据被清除为 0。
- ◆ 当 X4 为 On 时, 32 位计数器 C235 ~ C254 全部清除。(写入 0, 并将接点及线圈清除成 Off)。



补充说明

- ◆ 装置可以单独使用清除指令 (RST), 如位装置 Y、M、S 和字符装置 T、C、D。

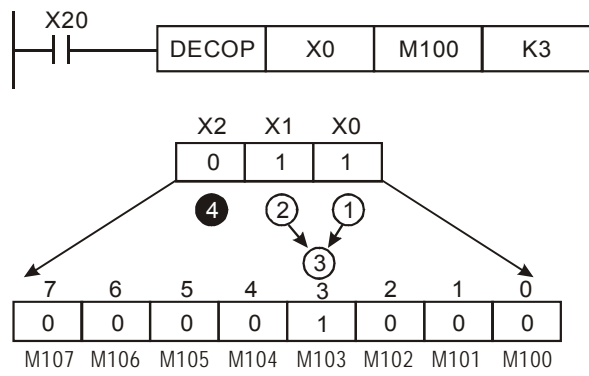
API																	适用机种																																																																																															
41	DECO	P	S	D	n	译码器											20PM	10PM																																																																																														
						✓	✓																																																																																																									
						<table border="1"> <tr> <th colspan="2">位装置</th> <th colspan="11">字符装置</th> <th colspan="4">16 位指令 (7 STEP)</th> </tr> <tr> <td>X</td><td>Y</td><td>M</td><td>S</td><td>K</td><td>H</td><td>KnX</td><td>KnY</td><td>KnM</td><td>KnS</td><td>T</td><td>C</td><td>D</td><td>V</td><td>Z</td> <td>DECO</td><td>连续执行型</td><td>DECO P</td><td>脉冲执行型</td> </tr> <tr> <td>S</td><td>*</td><td>*</td><td>*</td><td>*</td><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td>*</td><td>*</td><td>*</td><td>*</td><td>*</td> <td colspan="4">32 位指令</td> </tr> <tr> <td>D</td><td></td><td>*</td><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td><td>*</td><td>*</td><td>*</td><td>*</td> <td colspan="4">— — — —</td> </tr> <tr> <td>n</td><td></td><td></td><td></td><td></td><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> <td colspan="4">旗标信号: 无</td> </tr> </table>																	位装置		字符装置											16 位指令 (7 STEP)				X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	DECO	连续执行型	DECO P	脉冲执行型	S	*	*	*	*	*	*					*	*	*	*	*	32 位指令				D		*	*	*							*	*	*	*	*	— — — —				n					*	*							
位装置		字符装置											16 位指令 (7 STEP)																																																																																																			
X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	DECO	连续执行型	DECO P	脉冲执行型																																																																																														
S	*	*	*	*	*	*					*	*	*	*	*	32 位指令																																																																																																
D		*	*	*							*	*	*	*	*	— — — —																																																																																																
n					*	*										旗标信号: 无																																																																																																
<p>• 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表</p>																																																																																																																

指令说明

- ◆ **S**: 译码来源装置。 **D**: 存放译码结果之装置。 **n**: 译码位长度。
- ◆ 来源装置 **S** 的下位 “**n**” 位作译码, 并将其“**2ⁿ**” 位长度存于 **D**。
- ◆ 本指令一般都是使用脉冲执行型指令 (DECOP)。
- ◆ **D** 的范围为 1~8。

程式范例 (一)

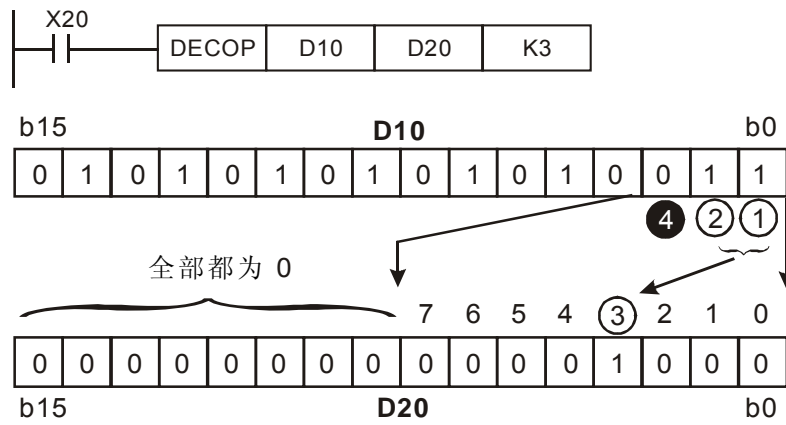
- ◆ 当 **D** 是位装置时, **n** 有效范围: $0 < n \leq 8$ 。 但是如果 $n=0$ or $n>8$, 会发生错误。
- ◆ 当 $n=8$ 时, 可做最大解碼 $2^8 = 256$ 点。
- ◆ 当 **X20** 从 Off → On 时, 指令将 **X0~X2** 内容值译码到 **M100~M107**。
- ◆ 如果 **S = 3**, **M103** (从 **M100** 开始算第 3 个位) = On。
- ◆ 当指令执行后, **X20** 变为 Off。 已经做解碼输出者照常动作。



程式范例 (二)

- ◆ 当 **D** 为字符装置时, **n** 有效范围: $0 < n \leq 8$, 如果 $n=0$ or $n>8$, 会发生错误。
- ◆ 当 $n=8$ 时, 可做最大解碼 $2^8 = 256$ 点。
- ◆ 当 **X20** 从 Off → On, 时, 指令将 **D10** 中 (**b2~ b0**) 的内容值解碼到 **D20** 的 (**b7~b0**)。 **D20** 中未被使用的位 (**b15~ b8**) 全部变为 0。
- ◆ **D10** 的下位 3 位作译码存放于 **D20** 的下位 8 位, 上 8 位皆为 0。
- ◆ 当指令执行后, **X20** 变为 Off 后。 已经做解碼输出者照常动作。

5 应用指令分类及基本使用



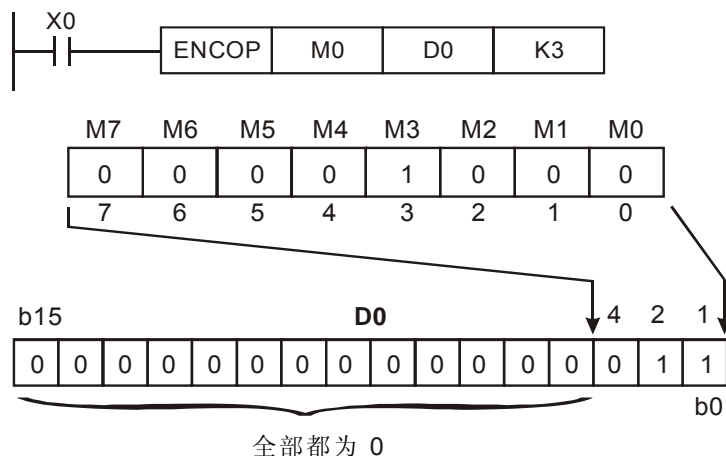
API																	适用机种		
42		ENCO	P	S	D	n	译码器										20PM	10PM	
																	✓	✓	
	位装置				字符装置										16 位指令 (7 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ENCO 连续执行型 ENCO P 脉冲执行型			
S	*	*	*	*							*	*	*	*	*	32 位指令			
D											*	*	*	*	*	— — — —			
n					*	*										旗标信号: 无			
<p>• 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表</p>																			

指令说明

- ◆ **S**: 编码来源装置。**D**: 存放编码结果的装置。**n**: 编码位长度。
- ◆ 来源装置 **S** 的下位“ 2^n ”位长度的数据作编码, 并将结果存于 **D**。
- ◆ 如果数据来源装置 **S** 有数位为 1 时, 则处理由高位往低位的第 1 个为 1 的位。
- ◆ 本指令一般都是使用脉冲执行型指令(ENCOP)。
- ◆ **S** 为位装置时, $n=1\sim 8$, 当 **S** 为字符装置时, $n=1\sim 4$ 。

程式范例 (一)

- ◆ 当 **S** 为位装置时, **n** 有效范围: $0 < n \leq 8$ 。如果 $n=0$ 或者 $n>8$, 会发生错误。
- ◆ 当 $n=8$ 时, 可做最大编码 $2^8=256$ 点。
- ◆ 当 X0 从 Off \rightarrow On 时, 指令将 2^3 位数据 (M0 to M7) 编码存放于 D0 的下位 3 位 (b2~b0) 内。D0 中未被使用的位 (b15~b3) 全部变为 0。
- ◆ 指令执行后, X0 变为 Off 后, **D** 内资料不变。

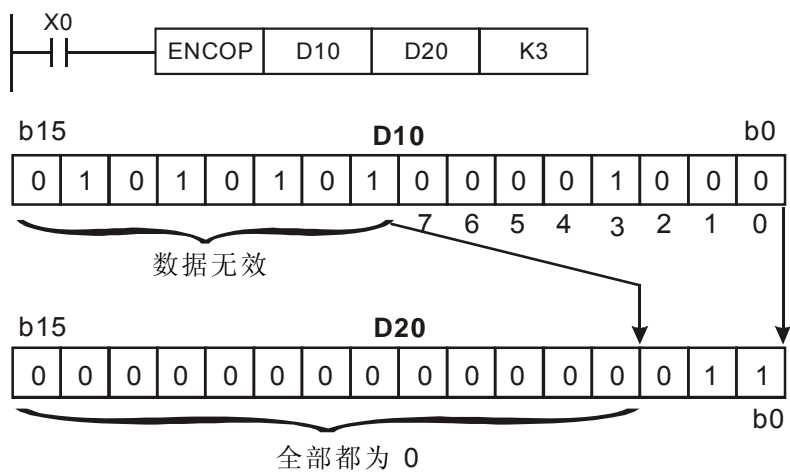


程式范例 (二)

- ◆ 当 **S** 为字符装置时, **n** 有效范围: $0 < n \leq 4$ 。如果 $n=0$ 或者 $n>4$ 时, 会发生错误。
- ◆ 当 $n=4$ 时, 可做 $2^4=16$ 点编码。
- ◆ 当 X0 从 Off \rightarrow On 时, D10 内 2^3 位数据 (b0~b7) 编码存放于 D20 之下位 3 位 (b2~b0) 内, D20 中未被使用之位 (b15~b3) 全部变为 0。(D10 内 b8~b15 为无效数据)。

5 应用指令分类及基本使用

- ◆ 当指令执行后, X0 变为 Off, D 内资料不变。



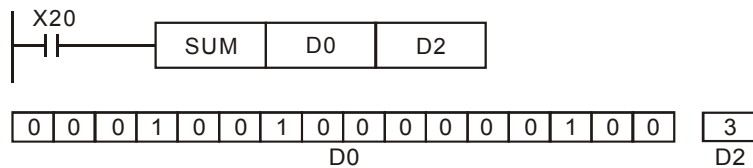
API																适用機種		
43	D	SUM	P	(S)	(D)	On 位数量										20PM	10PM	
																	✓	✓
	位装置				字符装置										16 位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SUM	SUMP	
S							*	*	*	*	*	*	*	*	*			
D											*	*	*	*	*			
<p>• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 KnX(Y/M/S)等装置的位装置起始编号限定为 16 的倍数 (包含 0),例如:K1X0, K4Y20, K4M16,</p>																<p>• 旗标信号</p> <p>Ox O100</p> <p>M1808 M1968 零旗标信号 Zero flag</p>		

指令说明

- ◆ **S**：来源装置。**D**：存放计数值的目的地装置。
- ◆ 在 **S** 中，所有位内容为“1”的总数将被储存于 **D**。
- ◆ 如果来源装置 **S** 的 16 个位全部为“0”时，零旗标信号 On。
- ◆ 当使用 32 位指令的时候，**D** 会占用两个寄存器。

程式范例

- ◆ 当 X20 =On 时, D0 的 16 个位中，内容为“1”的位总数被存于 D2 当中。



5 应用指令分类及基本使用

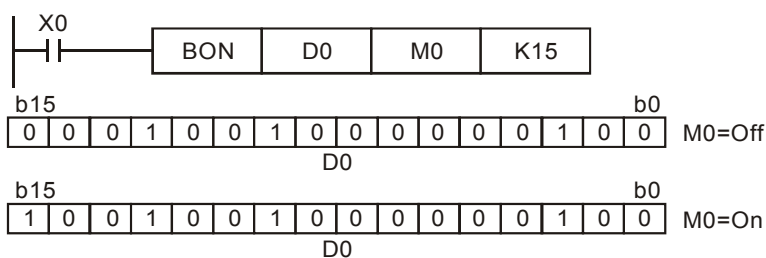
API					ON 位判定										适用機種				
44	D	BON	P												20PM	10PM			
															✓	✓			
	位装置				字符装置										16 位指令 (7 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	BON	连续执行型	BON P	脉冲执行型
S					*	*	*	*	*	*	*	*	*	*	*				
D		*	*	*															
n					*	*					*	*	*	*	*				
<ul style="list-style-type: none"> 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KNX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制) 															16 位指令 (7 STEP) 32 位指令 (13 STEP)				
															DBON 连续执行型 DBON P 脉冲执行型 • 旗标信号：无				

指令说明

- ◆ **S**：来源装置。**D**：存放判定结果的装置。**n**：指定判定的位(自 0 开始编号)。
- ◆ 来源装置特定位的状态被表示在目标位置。
- ◆ 操作数 **n** 的有效范围：**n=0~15** (16 位)，**n=0~31** (32 位)。

程式范例

- ◆ 当 X0 = On 时，若是 D0 的第 15 个位为“1”时，M0=On，为“0”时，M0=Off。
- ◆ X0 变成 Off，M0 仍保持之前的状态。



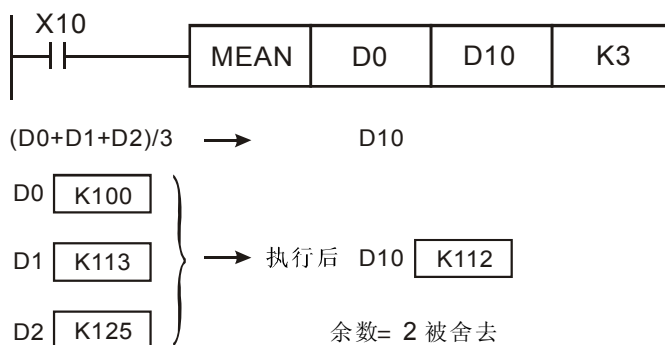
API																适用機種			
45	D	MEAN	P	S		D		n		平均值						20PM	10PM		
				✓	✓														
	位装置				字符装置												16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	MEAN	连续执行型	MEAN P	脉冲执行型
S					*	*	*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*	*	*				
n					*	*	*	*	*	*	*	*	*	*	*				
<ul style="list-style-type: none"> 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KNX(Y/M/S)等装置的位装置时,限以 8 进制或 10 进制中 16 的倍数为起始编号。如: K1X0, K4Y20 (八进制); K1M0, K4S16 (十进制) 																<ul style="list-style-type: none"> 旗标信号：无 			

指令说明

- ◆ **S**: 欲取平均值之起始装置。**D**: 存放平均值的装置。**n**: 取平均值的装置个数。
- ◆ 将 **S** 起始的 **n** 个装置内容相加后取平均值存入 **D** 中。
- ◆ 如果计算中出现余数时,余数会被舍去。
- ◆ 如果 **S** 没有在有效范围内,只有正常范围内的装置编号被处理。
- ◆ **n** 如果是 1~64 以外的数值时,PM 认定为“指令运算错误”。
- ◆ 操作数 **n** 的有效范围 : **n=1~64**。

程式范例

- ◆ 当 X10 = On 时, D0 开始算的 3 个(n=3)寄存器的内容全部相加,相加之后再除以 3 以求得平均值并存于指定的 D10 当中,余数被舍去。



5 应用指令分类及基本使用

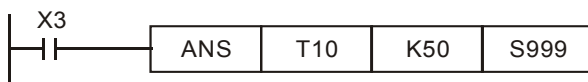
API																		适用機種		
46	ANS	P	S m			D			警报点输出										20PM	10PM
																✓	✓			
S	位装置				字符装置												16 位指令 (7 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ANS	连续执行型	ANS P	脉冲执行型	
m					*						*					32 位指令				
D				*												— — — —				
<ul style="list-style-type: none"> 操作数使用注意：所有装置不支持 V、Z 修饰 各装置使用范围请参考功能规格表 																<ul style="list-style-type: none"> 旗标信号：无 				

指令说明

- ◆ **S**：侦测警报定时器。**m**：计时时间设定。**n**：警报点装置。
- ◆ ANS 指令是用来驱动警报点输出的专用指令。
- ◆ 操作数 **S** 有效范围: T0~T183
操作数 **m** 有效范围: K1~K32,767 单位 100 ms。
操作数 **D** 有效范围: S912~S1023
见 ANR 更多的信息

程式范例

- ◆ X3=On 超过 5 秒钟时，警报点 S999=On，之后就算是 X3 变成 Off，S999 会继续保持 On。(但是 T10 会复归成 Off、现在值=0)。



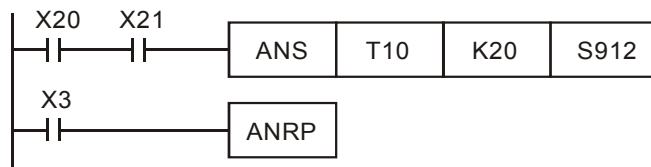
API																	适用機種		
47	ANR			P	警报点复归												20PM	10PM	
																	✓	✓	
位装置				字符装置												16 位指令 (1 STEP)			
X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ANR 连续执行型 ANR P 脉冲执行型				
<ul style="list-style-type: none"> 操作数使用注意: 无操作数 不须接点驱动的指令 															32 位指令				
															<ul style="list-style-type: none"> 旗标信号: 无 				

指令说明

- ◆ ANR 指令是用来复归警报点的专用指令。
- ◆ 复数个警报点同时 On 的时候, 较小号码的警报点被复归。
- ◆ 本指令一般都是使用脉冲执行型指令 (ANRP)

程式范例

- ◆ X20 与 X21 同时 On 超过 2 秒钟时, 警报点 S912 = On。如 X20 或 X21 变成 Off, 警报点 S912 会继续保持 On。T10 会被复归成 Off, 现在值为 0。
- ◆ X20 与 X21 同时 On 未满足 2 秒钟时, T10 的现在值被复归成 0。
- ◆ X3 从 Off → On 时, 动作中的警报点 S912~S1023 被复归。
- ◆ X3 再次从 Off → On 时, 次小号码警报点被复归。



5 应用指令分类及基本使用

补充说明

◆ 警报点的应用:

X0=前进开关

X1=后退开关

X2=前端定位开关

X3=后端定位开关

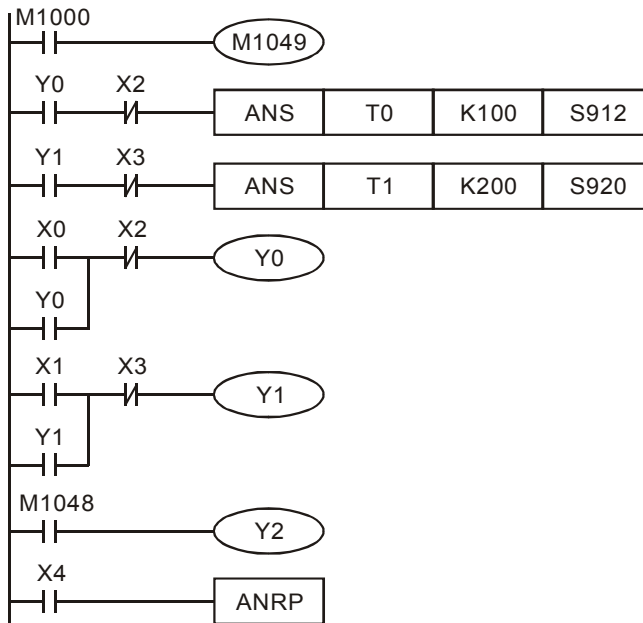
X4=警报点复归按钮

Y0=前进

Y1=前进

Y2=警报指示器

S912=前进警报点 S920=后退警报点



1. M1049=On 时, M1048、D1049 才有效。。
2. Y0=On 超过 10 秒物件未到达前端定位处 X2 时, S912=On。
3. Y1=On 超过 20 秒物件未到达后端定位处 X3 时, S920=On。
4. 当后退开关 X1=On, 后退装置 Y1=On, 直到物件到达后端定位开关 X3 时, Y1 才变为 Off。
5. 有警报点被驱动时, 警报指示器 Y2=On。
6. 当警报点的复归点按钮 X4 每 On 一次, 动作中的警报点号码就被复归一个, 复归的顺序从较小的号码开始。

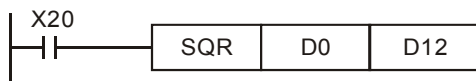
API																适用機種			
48	D	SQR	P	S	D	开平方根										20PM	10PM		
																	✓	✓	
	位装置				字符装置												16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SQR	连续执行型	SQR P	脉冲执行型
	S										*					32 位指令			
	m				*											DSQR	连续执行型	DSQRP	脉冲执行型
D				*											• 旗标信号: Ox O100 M1808 M1968 零旗标信号 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列指令说明				
• 操作数使用注意: 所有装置不支持 V、Z 修饰 各装置使用范围请参考功能规格表																			

指令说明

- ◆ **S**: 欲开平方根之来源装置。**D**: 存放结果的装置。
- ◆ 将 **S** 所指定的装置内容值开平方根后, 存放于 **D** 所指定装置。
- ◆ **S** 只可以指定正数, 若指定负数时, **PM** 视为错误, 指令不执行。
- ◆ 运算结果 **D** 只求整数, 小数点被舍弃。有小数点被舍弃时, 借位旗标信号 **On**。
- ◆ 运算结果 **D** 为 0 时, 零旗标信号 **On**。

程式范例

- ◆ 当 X20=On, 将 D0 内容值开平方根后, 存放于 D12。



$$\sqrt{D0} \rightarrow D12$$

5 应用指令分类及基本使用

API																适用機種		
49	D	FLT	P	S	D	BIN 整数→2 进小数点变换										20PM	10PM	
						✓	✓											
	位装置				字符装置												16 位指令	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S					*	*							*			32 位指令 (6 STEP)		
D													*			:DFLT 连续执行型 DFLTP 脉冲执行型		
操作数使用注意：各装置使用范围请参考功能规格表 此指令只有 32 位指令 DFLT、DFLTP 有效													旗标信号： Ox O100 M1808 M1968 零旗标信号 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列指令说明					

指令说明

- ◆ **S**: 变换来源装置。 **D**: 存放变换结果之装置。
- ◆ 将 BIN 整数变换成 2 进小数点值。
 1. 若转换结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
 2. 若转换结果的绝对值小于可表示之 最小浮点值，则借位旗号=On。
 3. 若转换结果为 0，则零旗号=On。

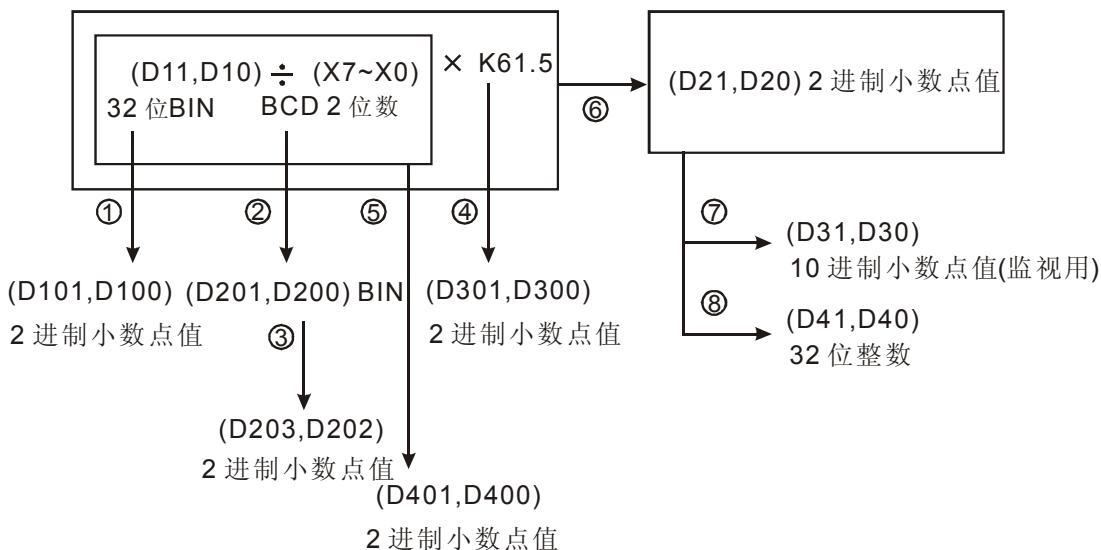
程式范例 (一)

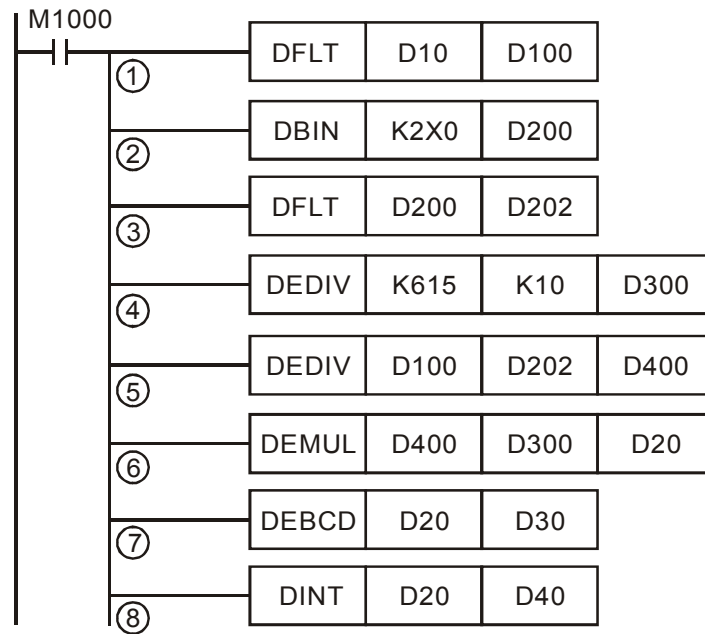
- ◆ 当 X11=On 时，将 D1、D0 (内为 BIN 整数) 变换成 D21、D20 (2 进小数点值)。
- ◆ 若 32-bit 寄存器 D0 (D1) =K100,000，则 X11=On，转换后浮点数之 32-bit 数值为 H4735000，存于 32-bit 寄存器 D20 (D21) 内。



程式范例 (二)

- ◆ 使用应用指令来完成下列的算式。





1. 将 D11、D10 (内为 BIN 整数) 变换成 D101、D100 (2 进小数点值)。
2. 将 X7~X0 (BCD 值) 变换成 D201、D200 (BIN 值)。
3. 将 D201、D200 (内为 BIN 整数) 变换成 D203、D202 (2 进小数点值)。
4. 将 $K615 \div K10$ 结果存于 D301、D300 (2 进小数点值)。
5. 2 进小数点除法 (D101、D100) \div (D203、D202) 结果存于 D401、D400 (2 进小数点值)。
6. 2 进小数点乘法 (D401、D400) \times (D301、D300) 结果存于 D21、D20 (2 进小数点值)。
7. 2 进小数点值 D21、D20 变换成 10 进小数点值 D31、D30。
8. 2 进小数点值 D21、D20 变换成 BIN 整数 D41、D40。

5 应用指令分类及基本使用

API																	适用機種				
50		REF		P	(D)	(n)	I/O 更新处理										20PM	10PM			
																		✓	✓		
	位装置				字符装置												16 位指令 (7 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z			REF	连续执行型	REF P	脉冲执行型
D	*	*																			
n					*	*															
<ul style="list-style-type: none"> 操作数使用注意：所有装置不支持 V、Z 修饰 各装置使用范围请参考功能规格表 																<ul style="list-style-type: none"> 旗标信号：无 					

指令说明

- ◆ **D**：欲 I/O 更新处理起始装置。**n**：I/O 更新处理数目。
- ◆ **PM** 的输入/出端子的状态全部为程序扫描至 **END** 后，才作状态的更新，其中输入点的状态是在程序开始扫描时，自外部输入点的状态读入存在输入点内存中，而输出端子在 **END** 指令后，才将输出点内存内容送至输出装置。因此在运算过程中需要最新的输入/出数据，则可利用本指令。
- ◆ **D** 操作数必须指定 **X0**、**X10**、**Y0**、**Y10**... 等个位数为 **0** 之编号。数字扩充机的 I/O 点无法使用此指令做立即更新处理。
- ◆ **D** 操作数指定的输入点及输出点仅限于主机的 I/O 点：
 1. **D** 指定 **X0** 且 $n \leq 8$ ，则只有 **X0~X7** 会被更新，如果 $n > 8$ ，则主机上所有输入及输出点都会被更新。
 2. **D** 指定 **Y0** 且 $n = 8$ ，则只有 **Y0~X7** 会被更新，如果 $n > 8$ ，则主机上所有输入及输出点都会被更新。
 3. **D** 指定 **X10** 或 **Y10** 且 **n** 不管多少，则主机上除了 **X0~X7** 或 **Y0~Y3** 之外，其余输入及输出点皆会被更新。
- ◆ **n** 操作数范围 $n = 4 \sim$ 主机的 I/O 点，且为 **4** 之倍数。
- ◆ 当 **X0 = On** 时，**PM** 可以读到 **X0~X7** 输入点的状态，输入信号更新，并没有输入延迟。

程式范例 (一)



程式范例 (二)

- ◆ 当 **X0 = On** 时，**Y0~Y3** 的 4 点输出信号将实时被送至输出端，不必到 **END** 指令才输出。



程式范例 (三)

- ◆ 当 **X0 = On** 时，输入 **X10** 之后点数，输出 **Y4** 之后点数皆会被更新。



或



API															适用機種		
61	D	SER	P	S1	S2	D	n	多点比较								20PM	10PM
																✓	✓

	位装置					字符装置										16 位指令 (9 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SER	连续执行型	SER P	脉冲执行型
S1							*	*	*	*	*	*	*						
S2					*	*	*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*						
N					*	*							*						

• 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置)
 各装置使用范围请参考功能规格表
 指定 KnX(Y/M/S)等装置的位装置时, 限以 8 进制或 10 进制中 16 的倍数为起始编号。如: K1X0, K4Y20 (八进制); K1M0, K4S16 (十进制)。

• 旗标信号: 无

指令说明

- ◆ **S1**: 多点比较的数据区块的起始装置。**S2**: 欲比较的数值数据。**D**: 存放比较结果的起始装置 (占用 5 个连续的装置)。**n**: 被比较的数据区块长度。
- ◆ **S1** 指定被比较寄存器区域的号码, **n** 指定被比较的笔数, 该多笔被比较寄存器的内容与 **S2** 所指定的数据作比较, 比较结果被存放于 **D** 所指定的数个寄存器当中。
- ◆ 使用 32 位指令时若指定寄存器, **S1**, **S2**, **D**, **n** 会指定 32 位寄存器。
- ◆ **n** 操作数范围: $n=1\sim 256$ (16 位指令); $n=1\sim 128$ (32 位指令)。
- ◆ 当 $X0=On$ 时, 由 D10~D19 组成之数据区块与 D0 作比较, 结果存放在 D50~D52 中, 当相等值不存在时, D50~D52 的内容全部为 0。
- ◆ 大小比较以代数型态进行。 ($-10 < 2$)。
- ◆ 所有比较数据之最小值编号记录在 D53, 最大值编号记录在 D54。当最小值最大值不只一个时, 会记录编号大者。

程式范例



5 应用指令分类及基本使用

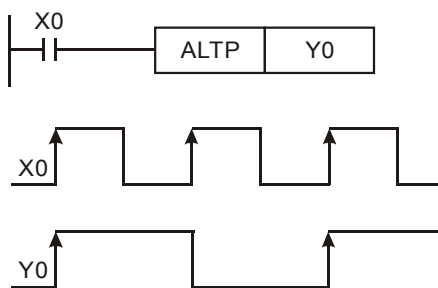
API 66	ALT	P	(D)	On/Off 交替											适用机种				
															20PM	10PM			
															✓	✓			
D	位装置				字符装置											16 位指令 (9 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ALT	连续执行型	ALT P	脉冲执行型
		*	*	*															
• 操作数使用注意：所有装置不支持 V、Z 修饰 各装置使用范围请参考功能规格表																32 位指令 (17 STEP)			
																• 旗标信号：无			

指令说明

- ◆ **D**：目的地装置。
- ◆ ALT 指令执行时，**D** On/Off 交换。
- ◆ 本指令一般都是使用脉冲执行型指令(ALTP)。

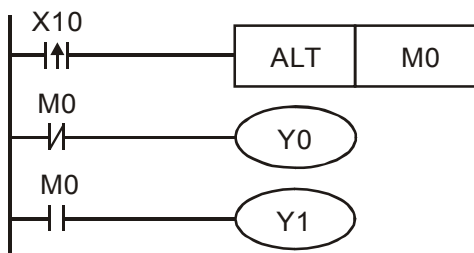
程式范例 (一)

- ◆ 当第一次 X0 从 Off→On 时, Y0=On。第二次 X0 从 Off→On 时, Y0=Off。



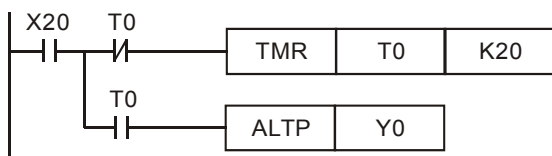
程式范例 (二)

- ◆ 使用单一开关控制启动与停止。一开始时, M0=Off 故 Y0=On、Y1=Off, 当 X10 作第一次 On/Off 时, M0=On 故 Y1=On、Y0=Off, 第二次 On/Off 时, M0=Off 故 Y0=On 而 Y1=Off。



程式范例 (三)

- ◆ 如下例所示, Y0 会产生闪烁的动作。当 X20=On 时, T0 每隔 2 秒产生一个脉冲, Y0 输出会依 T0 脉冲做 On/Off 交替变化。



API					倾斜信号										适用机种				
67	D	RAMP			S1	S2	D	n										20PM	10PM
																		✓	✓

	位装置				字符装置											
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	
S1													*			
S2													*			
D													*			
N													*			

• 操作数使用注意： 所有装置不支持 V、Z 修饰
各装置使用范围请参考功能规格表

16 位指令 (9 STEP)
RAMP 连续执行型 -

32 位指令 (17 STEP)
D RAMP 连续执行型 -

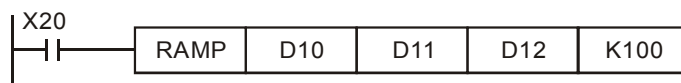
• 旗标信号： M1026 启动模态，请参考补充说明
M1029 指令执行完毕

指令说明

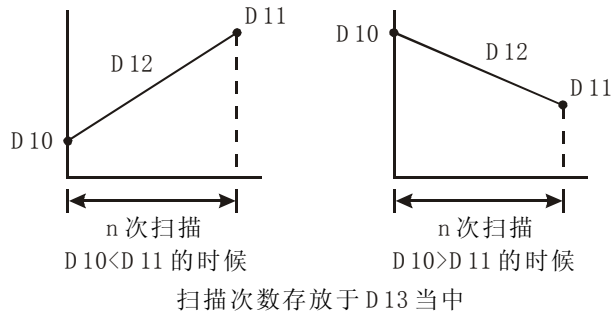
- ◆ **S₁**：倾斜信号之起点设定。**S₂**：倾斜信号之终点设定。**D**：倾斜信号之经过时间值。(占用两个连续的装置)。**n**：扫描次数。(n = 1~32,767)。
- ◆ 本指令是一个求斜率的指令，斜率是否线性与扫描时间有绝对的关系，因此使用本指令时，通常预先将扫描时间加以固定。
- ◆ 当 RAMP 指令驱动节点由 Off 变为 On 时，倾斜信号的现在值会从倾斜信号的起点设置值向倾斜信号的终点设置值变化，**D+1** 装置的值从 0 开始计数当前的扫描次数。倾斜信号的现在值存储于 **D** 装置内，当前的扫描次数存储于 **D+1** 装置内。
- ◆ 若 **n** 操作数使用 **D** 装置时，须注意 **D** 的内容须等待指令停止执行时才能变化，于指令执行中不能变化。
- ◆ 于程序中首先将 M1039 驱动为 On 即可固定扫描时间，再使用 MOV 指令将固定扫描时间设定值写入至特殊数据寄存器 D1039 当中即可。假设该值为 30ms，以上述程序为例，n = K100，则 D10 至 D11 的时间为 3 秒 (30ms×100)。
- ◆ 指令执行中，启动信号 X20 变成 Off 时，指令停止执行，当 X20 再度 On 的时候，D12 的内容被复归为 0 而重新计算。
- ◆ 当 M1026=Off 且 RAMP 执行完成旗标 M1029=On 时，D12 的内容值会被复归成 D10 的设定值。
- ◆ 本指令若是与模拟信号输出搭配使用时，可执行缓冲启动/停止的动作。

程式范例

- ◆ 预先将倾斜信号的起点设定值写入 D10 及倾斜信号的终点设定值写入 D11 内，当 X20=On 时，D12 的值会从 D10 设定值向 D11 的设定值增加，D12 的内容值从 D10 的设定值变化至 D12 现在值经过的扫描次数存于 D13 当中。

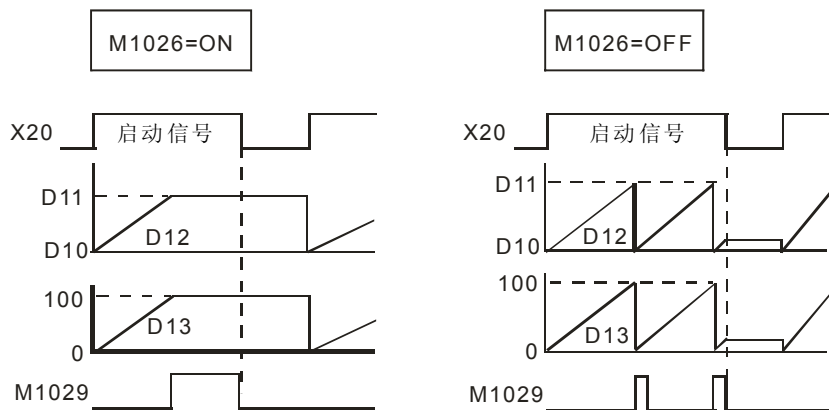


5 应用指令分类及基本使用



补充说明

- ◆ M1026 (RAMP 模式选择) 的 On/Off, D12 的内容变化如下:

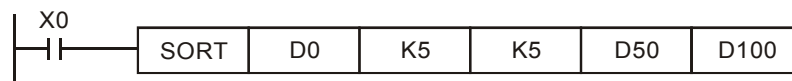


API																适用機種		
69	D	SORT		S	m1	m2	D	n	数据排序							20PM	10PM	
																✓	✓	
	位装置				字符装置												16 位指令 (9 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	:SORT 连续执行型 -		
S													*			:32 位指令 (17 STEP)		
m1					*	*										:DSORT 连续执行型 -		
m2					*	*												
D													*					
N					*	*							*					
<ul style="list-style-type: none"> 操作数使用注意： 所有装置不支持 V、Z 修饰 各装置使用范围请参考功能规格表 																		

指令说明

- ◆ **S**：原始数据区块之起始装置。**m1**：被排序之数据组数。**(m1=1~32)**。**m2**：每笔数据之字段数。**(m2=1~6)**。**D**：存放排序结果数据区块之起始装置。**n**：数据排序的参考值。**(n=1~ m2)**，指定列的分类排序以代数的形式从小到大排列。
- ◆ 排序结果显示于 **D** 所指定的起始号码开始算的 **m1×m2** 个寄存器当中，因此，**S** 与 **D** 指定同一个寄存器的话，排序结果将与原来被排序的数据 **S** 内容相同。
- ◆ **S** 寄存器的起始号码的最右边编号指定 **0** 比较理想。
- ◆ 本指令必须经过 **m1** 次的扫描时间之后才被排序完成，排序完成时执行完毕旗标信号 **M1029=On**。
- ◆ 本指令于程序中使用次数并无限制，但是同时间仅有 **1** 个指令被执行。
- ◆ 当 **X0=On** 时，指定执行数据排序作业，排序完成时，**M1029=On**。指令执行中请勿变更排序数据内容，若是要数据重新排序时，请 **X0** 再 **Off→On** 一次即可。

程式范例



1. 排序数据构成例

		← 资料数: m2 个 →				
		数据域位				
	行列	1	2	3	4	5
		学生编号	国文	英文	数学	理化
↑ 资料 个数 : m1 个 ↓	1	(D0) 1	(D5) 90	(D10) 75	(D15) 66	(D20) 79
	2	(D1) 2	(D6) 55	(D11) 65	(D16) 54	(D21) 63
	3	(D2) 3	(D7) 80	(D12) 98	(D17) 89	(D22) 90
	4	(D3) 4	(D8) 70	(D13) 60	(D18) 99	(D23) 50
	5	(D4) 5	(D9) 95	(D14) 79	(D19) 75	(D24) 69

5 应用指令分类及基本使用

2. 当 D100=K3 时的排序后数据

		← 资料数: m2 个 →				
		数据域位				
		1	2	3	4	5
行列		学生编号	国文	英文	数学	理化
↑ 资料 个数 : m1 个 ↓	1	(D50) 4	(D55) 70	(D60) 60	(D65) 99	(D70) 50
	2	(D51) 2	(D56) 55	(D61) 65	(D66) 54	(D71) 63
	3	(D52) 1	(D57) 90	(D62) 75	(D67) 66	(D72) 79
	4	(D53) 5	(D58) 95	(D63) 79	(D68) 75	(D73) 69
	5	(D54) 3	(D59) 80	(D64) 98	(D69) 89	(D74) 90

3. 当 D100=K5 时的排序后数据

		← 资料数: m2 个 →				
		数据域位				
		1	2	3	4	5
行列		学生编号	国文	英文	数学	理化
↑ 资料 个数 : m1 个 ↓	1	(D50) 4	(D55) 70	(D60) 60	(D65) 99	(D70) 50
	2	(D51) 2	(D56) 55	(D61) 65	(D66) 54	(D71) 63
	3	(D52) 1	(D57) 90	(D62) 75	(D67) 66	(D72) 79
	4	(D53) 5	(D58) 95	(D63) 79	(D68) 75	(D73) 69
	5	(D54) 3	(D59) 80	(D64) 98	(D69) 89	(D74) 90

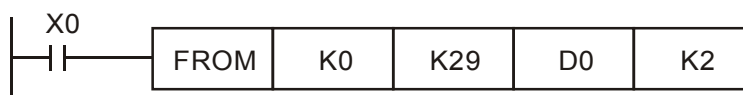
API																适用機種		
78	D	FROM	P	m₁	m₂	D	n	特殊模块 CR 数据读出								20PM	10PM	
																✓	✓	
	位装置				字符装置												16 位指令 (9 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	FROM 连续执行型 FROMP 脉冲执行型		
m ₁					*	*					*	*	*	*	*	32 位指令 (17 STEP)		
m ₂					*	*					*	*	*	*	*	DFROM 连续执行型 DFROMP 脉冲执行型		
D											*	*	*	*	*	• 请参考下列补充说明		
n					*	*					*	*	*	*	*			
<p>• 操作数使用注意: m₁操作数使用范围(16 及 32 位指令): 0~255 m₂操作数使用范围(16 及 32 位指令): 0~499 n 操作数使用范围: (16 位指令): 1~(500- m₂) (32 位指令): 1~(500- m₂)/2 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置)</p>																		

指令说明

- ◆ **m₁**: 特殊模块所在之编号。 **m₂**: 欲读取特殊模块之 CR (Controlled Register) 编号。 **D**: 存放读取数据的位置。 **n**: 一次读取之数据笔数。
- ◆ DVP-PM 系列機種利用此指令读取特殊模块之 CR 数据。
- ◆ 特殊模块所在之编号算法请参考 TO 指令补充说明。

程式范例

- ◆ 将编号为 0 特殊模块的 CR#29 的内容读出至 DVP-PM 的 D0 当中, CR#30 的内容读出至 DVP-PM 的 D1 当中, 一次读取二笔 (n=2)。
- ◆ X0=On 的时候指令被执行, X0 变成 Off 时, 指令不执行, 之前读出的数据其内容没有变化。



5 应用指令分类及基本使用

API																	适用机种	
79	D	TO	P	m ₁	m ₂	S	n	特殊模块 CR 数据写入								20PM	10PM	
								✓	✓									

	位装置					字符装置										16 位指令 (9 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	TO	连续执行型	TOP	脉冲执行型
m ₁					*	*					*	*	*	*	*				
m ₂					*	*					*	*	*	*	*				
S					*	*					*	*	*	*	*				
n					*	*					*	*	*	*	*				

• 操作数使用注意: m₁操作数使用范围(16 及 32 位指令): 0~255
 m₂操作数使用范围(16 及 32 位指令): 0~499
 n 操作数使用范围:
 (16 位指令): 1~(500- m₂)
 (32 位指令): 1~(500- m₂)/2
 此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置; 当 32 位指令时不能使用 V 装置)

• 请参考下列补充说明

指令说明

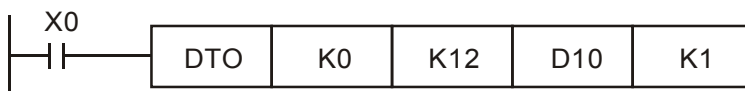
◆ m₁: 特殊模块所在之编号。 m₂: 欲写入特殊模块之 CR (Controlled Register) 编号。 S: 写入 CR 的资料。 n: 一次写入之资料笔数。

◆ DVP-PM 系列机种利用此指令将数据写入特殊模块之 CR 内。

程式范例

◆ 使用 32 位指令 DTO, 程序的动作是将 D11、D10 的内容写入编号为 0 之特殊模块的 CR#13、#12 当中, 一次只写入一笔 (n=1)。

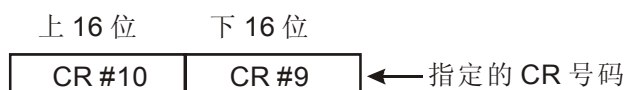
◆ X0=On 时, 指令被执行, X0 变成 Off 时, 指令不被执行, 写入的数据没有变化。



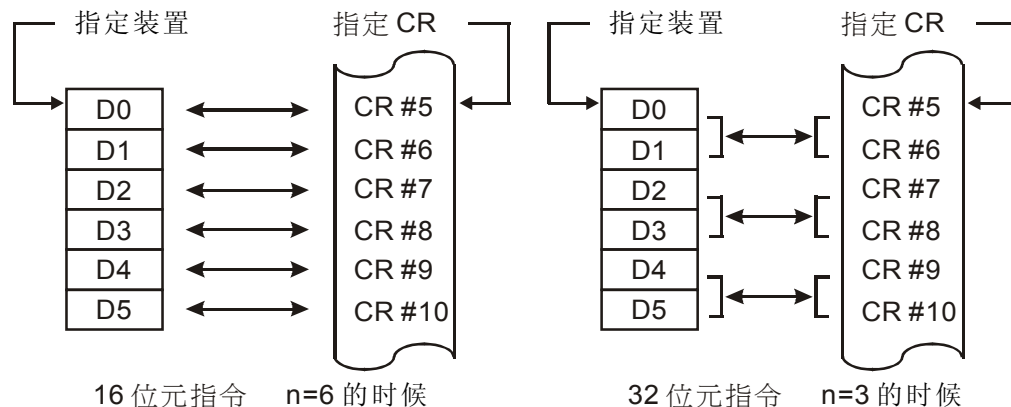
补充说明

◆ 指令操作数的规则

1. m₁: 特殊模块的排列号码, DVP-PM 主机所连接特殊模块的编号。
特殊模块所在之编号算法是以最靠近主机的模块编号为 0, 依序排列, 最多可挂 8 台特殊模块, 且不占用 I/O 点数。
2. m₂: CR 的号码, 特殊模块的内部内建多组 16 位长度的内存, 称之为 CR (Controlled Register)。 CR 的编号以 10 进制编码, 特殊模块的各种运转情况及设定值均被包含在里面。
3. 如果使用 FROM/TO 指令时, 一次以一个编号的 CR 为读出/写入单位, 若是使用 DFROM/DTO 指令时, 一次以 2 个编号的 CR 为读出/写入单位。



4. 传送组数 n, 16 位指令的 n=2 与 32 位指令的 n=1 意义相同。



◆ FROM / TO 指令应用范例说明:

程式范例
(一)

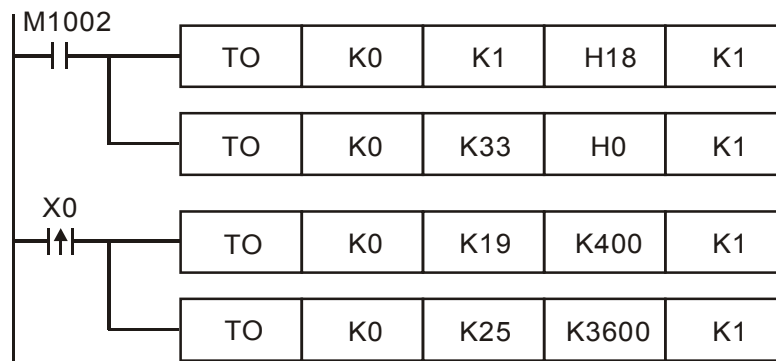
◆ 调整 DVP04AD-H2 之 A/D 转换特性曲线, 将 CH1 之 Offset 值设为 0V (=K0_{LSB}), GAIN 值设为 2.5V (=K2,000_{LSB})。



1. 对模拟信号输入模块编号 0 之 CR#1 写入 H0, CH1 设为模式 0 (电压输入 -10V~+10V)。
2. CR#33 写入 H0, 允许 CH1~CH4 都可特性微调。
3. 当 X0=Off→On 时, 将 Offset 值 K0_{LSB} 写入 CR#18 内。将 GAIN 值 K2,000_{LSB} 写入 CR#24 内。

程式范例
(二)

◆ 调整 DVP04AD-H2 之 A/D 转换特性曲线, 将 CH2 之 Offset 值设为 2mA (=K400_{LSB}), GAIN 值设为 18 mA (=K3,600_{LSB})。



1. 对模拟信号输入模块编号 0 之 CR#1 写入 H18, CH2 设为模式 3 (电流输入 -20

5 应用指令分类及基本使用

程式范例 (三)

mA~+20mA)。

2. CR#33 写入 H0, 允许 CH1 ~ CH4 都可特性微调。
3. 当 X0=Off→On 时, 将 Offset 值 K400_{LSB} 写入 CR#19 内。将 GAIN 值 K3,600_{LSB} 写入 CR#25 内。

- ◆ 调整 DVP02DA-H2 之 D/A 转换特性曲线, 将 CH2 之 Offset 值设为 0mA (=K0_{LSB}), GAIN 值设为 10mA (=K1,000_{LSB})。



1. 对模拟信号输出模块编号 1 之 CR#1 写入 H18, CH2 设为模式 3 (电流输出 0mA~+20mA)。
2. CR#33 写入 H0, 允许 CH1、CH2 特性微调。
3. 当 X0=Off→On 时, 将 Offset 值 K0_{LSB} 写入 CR#22 内。将 GAIN 值 K1,000_{LSB} 写入 CR#28 内。

程式范例 (四)

- ◆ 调整 DVP02DA-H2 之 D/A 转换特性曲线, 将 CH2 之 Offset 值设为 2mA (=K400_{LSB}), GAIN 值设为 18mA (=K3,600_{LSB})。



1. 对模拟信号输出模块编号 1 之 CR#1 写入 H10, CH2 设为模式 2 (电流输出 +4mA~+20mA)。
2. CR#33 写入 H0, 允许 CH1、CH2 特性微调。
3. 当 X0=Off→On 时, 将 Offset 值 K400_{LSB} 写入 CR#
4. 23 内。将 GAIN 值 K3,600_{LSB} 写入 CR#29 内。

API																适用机种				
87	D	ABS	P	D	绝对值											20PM	10PM			
																✓	✓			
D	位装置				字符装置											16 位指令 (9 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ABS	连续执行型	ABS P	脉冲执行型	
							*	*	*	*	*	*	*	*	*					
<p>• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KNX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)</p>																	32 位指令 (17 STEP)			
																	DABS	连续执行型	DABSP	脉冲执行型
																	• 旗标信号：无			

指令说明

- ◆ **D**：欲取绝对值之装置。
- ◆ 当 **ABS** 指令执行，被指定的组件 **D** 取绝对值。
- ◆ 本指令一般都是使用脉冲执行型指令(**ABSP**, **DABSP**)。

程式范例

- ◆ 当 X0 从 Off→On, D0 内容取绝对值。



5 应用指令分类及基本使用

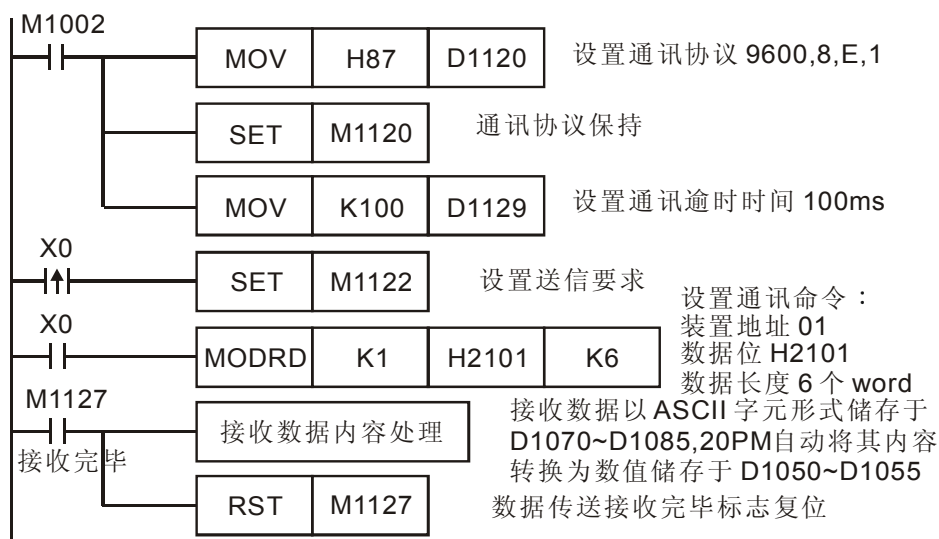
API					MODBUS 数据读取										适用機種				
100					(S_1) (S_2) (n)										20PM	10PM			
															✓	✓			
	位装置				字符装置										16 位指令 (7 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	MODRD 连续执行型	-	-	
S_1					*	*							*			32 位指令		-	-
S_2					*	*							*					-	-
n					*	*							*					-	-
• 操作数使用注意: S_1 操作数范围 K0~K254 n 操作数范围 $K1 \leq n \leq K6$ 各装置使用范围请参考功能规格表															• 旗标信号: M1120~M1129、M1140~M1143 请参考下列补充说明				

指令说明

- ◆ S_1 : 联机装置地址。 S_2 : 欲读取数据的地址。 n : 读取数据长度。
- ◆ MODRD 指令系针对 MODBUS ASCII 模式/RTU 模式的通讯外围设备专用的驱动指令。台达 VFD 变频器内建的 RS-485 通讯接口皆符合 MODBUS 的通讯格式 (除了 VFD-A 系列), 可利用 MODRD 指令对台达变频器进行通讯控制 (数据读取)。
- ◆ S_2 欲读取数据的地址。若地址对于被指定的联机装置不合法, 则联机装置会响应错误讯息, DVP-PM 将错误码储存于 D1130, 同时, M1141 会 On。
- ◆ 联机外围装置回传的数据储存于 D1070~D1085。接收完毕后, DVP-PM 会自动检查所接收的数据是否有误, 若发生错误则 M1140 会 On。
- ◆ 若使用 ASCII 模式, 由于回传的数据均为 ASCII 字符, DVP-PM 会另外将回传主要的资料转为数值转存于 D1050~D1055。若使用 RTU 模式则 D1050~D1055 无效。
- ◆ 当 M1140=On 或 M1141=On 之后, 再传送一笔正确数据给外围装置, 若回传的数据正确则旗标 M1140, M1141 会被清除。

程式范例 (一)

- ◆ DVP-PM 与 VFD-B 系列变频器联机 (ASCII Mode, M1143=Off)



DVP-PM ⇒ VFD-S, DVP-PM 传送: "01 03 2101 0006 D4"
 VFD-S ⇒ DVP-PM, DVP-PM 接收: "01 03 0C 0100 1766 0000 0000 0136 0000 3B"

DVP-PM 传送数据寄存器 (传送讯息)

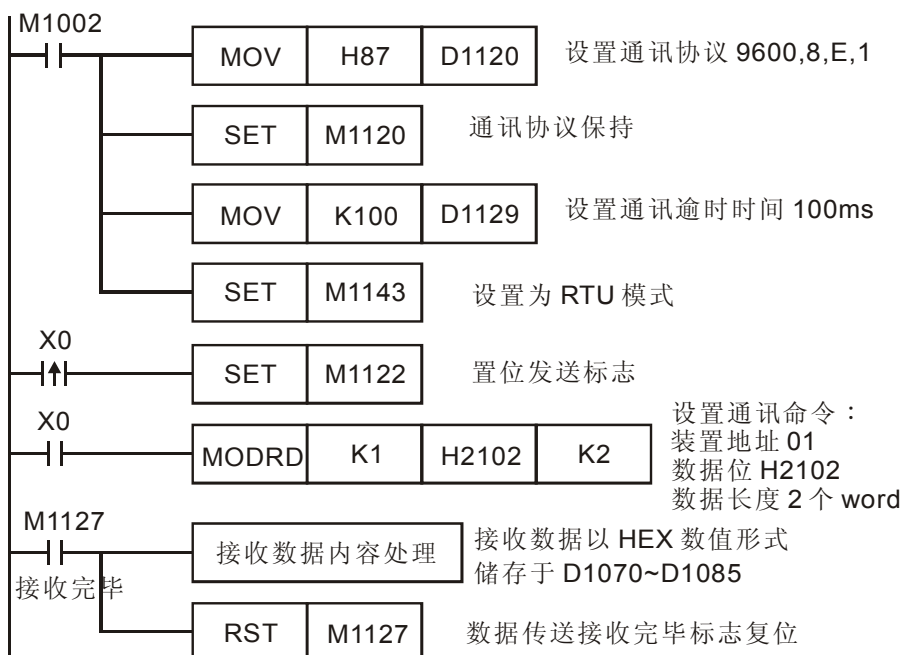
寄存器	DATA		说 明	
D1089 下	'0'	30 H	ADR 1	ADR (1,0) 为变频器地址
D1089 上	'1'	31 H	ADR 0	
D1090 下	'0'	30 H	CMD 1	CMD (1,0) 为命令码
D1090 上	'3'	33 H	CMD 0	
D1091 下	'2'	32 H	起始数据地址 Starting Data Address	
D1091 上	'1'	31 H		
D1092 下	'0'	30 H		
D1092 上	'1'	31 H		
D1093 下	'0'	30 H	资料 (word) 个数 Number of Data (count by word)	
D1093 上	'0'	30 H		
D1094 下	'0'	30 H		
D1094 上	'6'	36 H		
D1095 下	'D'	44 H	LRC CHK 1	LRC CHK (0,1) 为错误检查码
D1095 上	'4'	34 H	LRC CHK 0	

DVP-PM 接收数据寄存器 (响应讯息)

寄存器	DATA		说 明	
D1070 下	'0'	30 H	ADR 1	ADR 0
D1070 上	'1'	31 H	ADR 0	
D1071 下	'0'	30 H	CMD 1	CMD 0
D1071 上	'3'	33 H	CMD 0	
D1072 下	'0'	30 H	资料 (BYTE) 个数 Number of Data (count by Byte)	
D1072 上	'C'	43 H		
D1073 下	'0'	30 H	地址 2101 H 的内容	DVP-PM 自动将 ASCII 字符转换为数值储存于 D1050=0100 H
D1073 上	'1'	31 H		
D1074 下	'0'	30 H		
D1074 上	'0'	30 H		
D1075 下	'1'	31 H	地址 2102 H 的内容	DVP-PM 自动将 ASCII 字符转换为数值储存于 D1051=1766 H
D1075 上	'7'	37 H		
D1076 下	'6'	36 H		
D1076 上	'6'	36 H		
D1077 下	'0'	30 H	地址 2103 H 的内容	DVP-PM 自动将 ASCII 字符转换为数值储存于 D1052=0000 H
D1077 上	'0'	30 H		
D1078 下	'0'	30 H		
D1078 上	'0'	30 H		
D1079 下	'0'	30 H	地址 2104 H 的内容	DVP-PM 自动将 ASCII 字符转换为数值储存于 D1053=0000 H
D1079 上	'0'	30 H		
D1080 下	'0'	30 H		
D1080 上	'0'	30 H		
D1081 下	'0'	30 H	地址 2105 H 的内容	DVP-PM 自动将 ASCII 字符转换为数值储存于 D1054=0136 H
D1081 上	'1'	31 H		
D1082 下	'3'	33 H		
D1082 上	'6'	36 H		
D1083 下	'0'	30 H	地址 2106 H 的内容	DVP-PM 自动将 ASCII 字符转换为数值储存于 D1055=0000 H
D1083 上	'0'	30 H		
D1084 下	'0'	30 H		
D1084 上	'0'	30 H		
D1085 下	'3'	33 H	LRC CHK 1	LRC CHK 0
D1085 上	'B'	42 H	LRC CHK 0	

程式范例
(二)

◆ DVP-PM 与 VFD-B 系列变频器联机 (RTU Mode, M1143=On)



DVP-PM ⇒ VFD-B, DVP-PM 传送: **01 03 2102 0002 6F F7**

VFD-B ⇒ DVP-PM, DVP-PM 接收: **01 03 04 1770 0000 FE 5C**

DVP-PM 传送数据寄存器 (传送讯息)

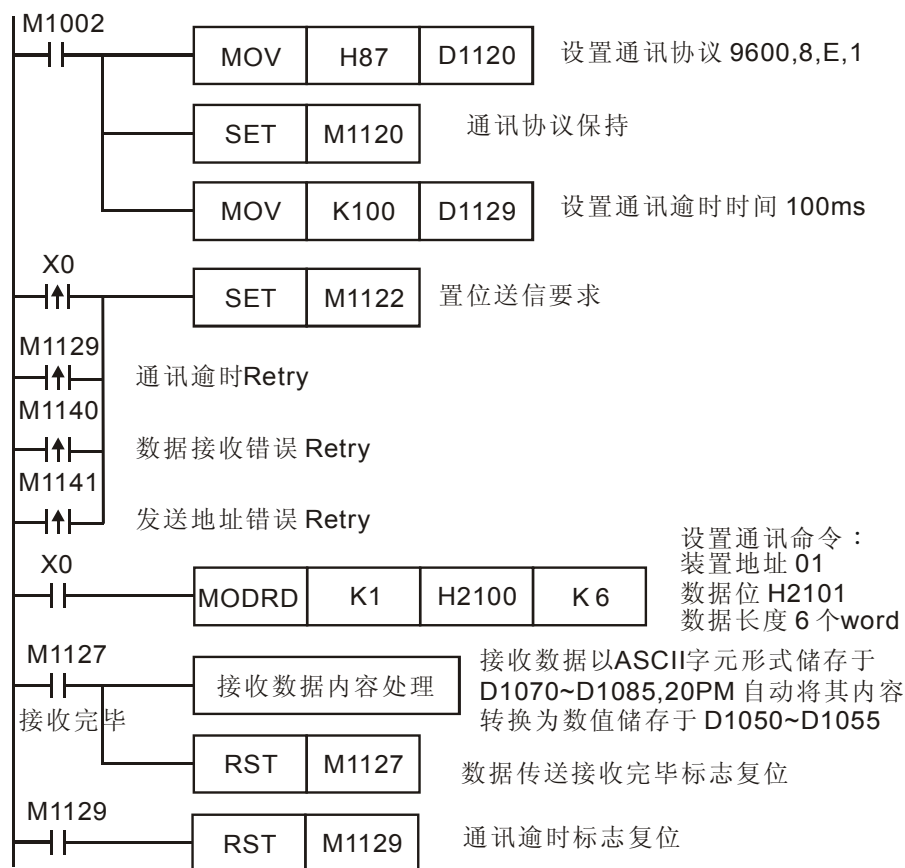
寄存器	DATA	说明
D1089 下	01 H	Address
D1090 下	03 H	Function
D1091 下	21 H	起始数据地址
D1092 下	02 H	Starting Data Address
D1093 下	00 H	资料 (word) 个数
D1094 下	02 H	Number of Data (count by word)
D1095 下	6F H	CRC CHK Low
D1096 下	F7 H	CRC CHK High

DVP-PM 接收数据寄存器 (响应讯息)

寄存器	DATA	说明
D1070 下	01 H	Address
D1071 下	03 H	Function
D1072 下	04 H	资料 (Byte) 个数, Number of Data (Byte)
D1073 下	17 H	地址 2102 H 的内容
D1074 下	70 H	
D1075 下	00 H	地址 2103 H 的内容
D1076 下	00 H	
D1077 下	FE H	CRC CHK Low
D1078 下	5C H	CRC CHK High

程式范例 (三)

- ◆ DVP-PM 与 VFD-B 系列变频器联机 (ASCII Mode, M1143=Off), 当通讯逾时、接收数据错误及发送地址错误之 Retry。
- ◆ 当 X0=On 时, DVP-PM 将装置地址 01 之 VFD-B 变频器数据地址 H2100 内数据读出, 数据以 ASCII 字符形式储存于 D1070~D1085。DVP-PM 自动将其内容转换为数值储存于 D1050~D1055。
- ◆ 若通讯逾时则 M1129 旗标为 On, 程序中由 M1129 触发送信要求 M1122 再读取一次。
- ◆ 若数据接收错误则 M1140 旗标为 On, 程序中由 M1140 触发送信要求 M1122 再读取一次。
- ◆ 若发送地址错误则 M1141 旗标为 On, 程序中由 M1141 触发送信要求 M1122 再读取一次。



补充说明

- ◆ API 100 MODRD (Function Code H03) 指令前面启动条件不可使用接点上升缘 (LDP, ANDP, ORP) 及接点下降缘 (LDF, ANDF, ORF)。否则存放在接收寄存器的数据会不正确。
- ◆ 本指令于程序中使用次数并无限制, 但是同时间仅有一个指令被执行。

5 应用指令分类及基本使用

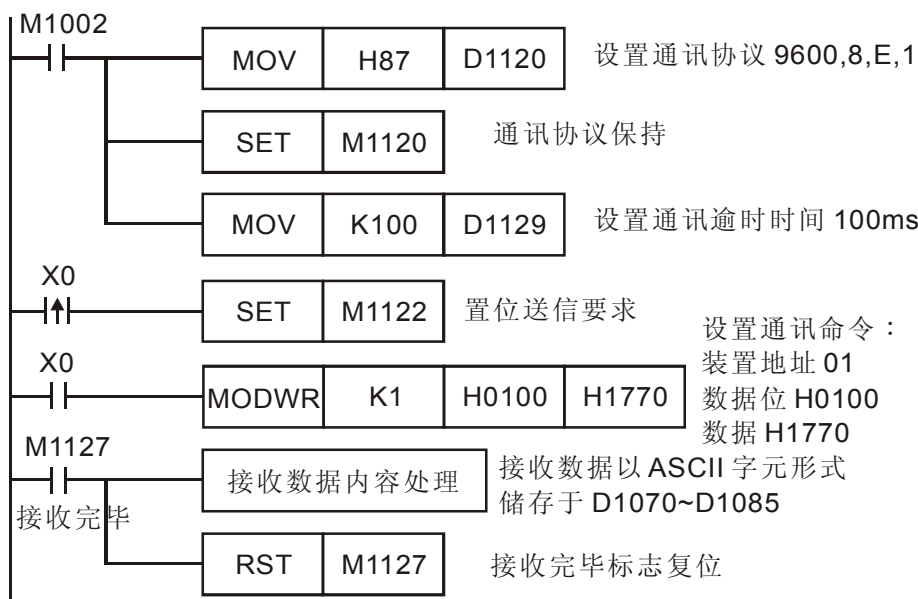
API																	适用機種	
101	MODWR				S ₁ S ₂ n			MODBUS 资料写入									20PM	10PM
																	✓	✓
	位装置				字符装置												16 位指令 (7 STEP) MODWR 连续执行型 — — 32 位指令 — — — —	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z			
S ₁					*	*							*					
S ₂					*	*							*					
n					*	*							*					
• 操作数使用注意: S ₁ 操作数范围 K0~K254 各装置使用范围请参考功能规格表																	• 旗标信号: M1120~M1129、M1140~M1143 请参考下列补充说明	

指令说明

- ◆ **S₁**: 联机装置地址。 **S₂**: 欲写入数据的地址。 **n**: 欲写入的资料。
- ◆ MODWR 指令系针对 MODBUS ASCII 模式 / RTU 模式的通讯外围设备专用的驱动指令。台达 VFD 变频器内建 RS-485 通讯接口皆符合 MODBUS 的通讯格式 (除了 VFD-A 系列), 因此可利用 MODWR 指令对台达变频器进行通讯控制 (数据写入)。
- ◆ **S₂** 欲写入数据的地址。若地址对于被指定的装置不合法, 则会响应错误讯息, 错误码储存于 D1130, 同时, M1141 会 On。例如 8000H 对 VFD-B 不合法, 则 M1141 On, D1130=2, 错误码请参考 VFD-B 使用手册)。
- ◆ 外围装置回传的数据储存于 D1070~1076。接收完毕后, DVP-PM 会自动检查所接收的数据是否有误, 若发生错误则 M1140 会 On。
- ◆ 当 M1140=On 或 M1141=On 之后, 再传送一笔正确数据给外围装置, 若回传的数据正确则旗标 M1140, M1141 会被清除。

程式范例 (一)

- ◆ DVP-PM 与 VFD-B 系列变频器联机 (ASCII Mode, M1143=Off)



DVP-PM ⇨ VFD-B, DVP-PM 传送: “ 01 06 0100 1770 71 ”

VFD-B ⇨ DVP-PM, DVP-PM 接收: “ 01 06 0100 1770 71 ”

DVP-PM 传送数据寄存器 (传送讯息)

寄存器	DATA		说明	
D1089 下	'0'	30 H	ADR 1	ADR (1,0) 为变频器地址
D1089 上	'1'	31 H	ADR 0	
D1090 下	'0'	30 H	CMD 1	CMD (1,0) 为命令码
D1090 上	'6'	36 H	CMD 0	
D1091 下	'0'	30 H	数据地址 Data Address	
D1091 上	'1'	31 H		
D1092 下	'0'	30 H		
D1092 上	'0'	30 H		
D1093 下	'1'	31 H	数据内容 Data contents	
D1093 上	'7'	37 H		
D1094 下	'7'	37 H		
D1094 上	'0'	30 H		
D1095 下	'7'	37 H	LRC CHK 1	LRC CHK (0,1) 为错误检查码
D1095 上	'1'	31 H	LRC CHK 0	

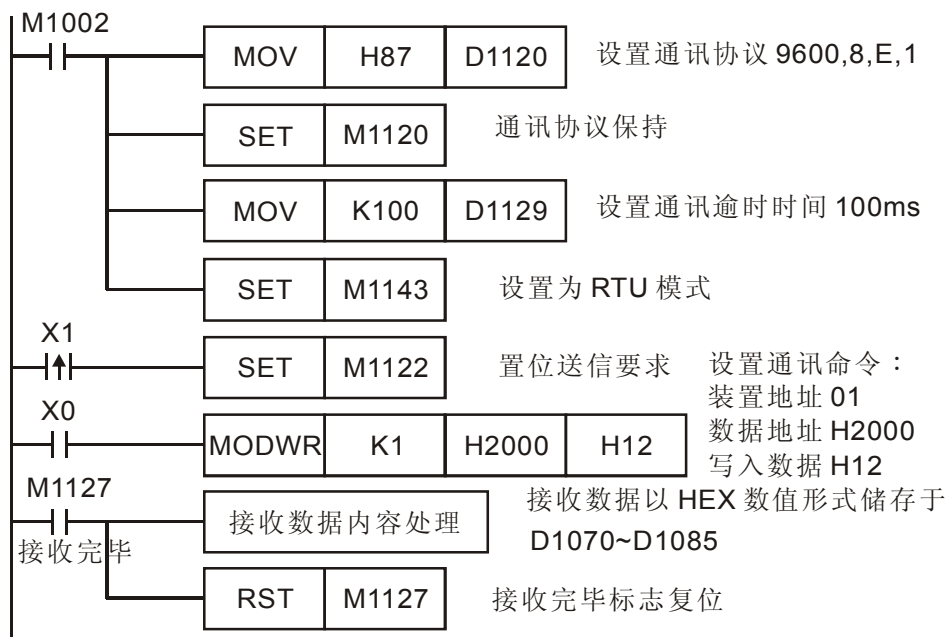
DVP-PM 接收数据寄存器 (响应讯息)

寄存器	DATA		说明	
D1070 下	'0'	30 H	ADR 1	ADR 0
D1070 上	'1'	31 H	ADR 0	
D1071 下	'0'	30 H	CMD 1	CMD 0
D1071 上	'6'	36 H	CMD 0	
D1072 下	'0'	30 H	数据地址 Data Address	
D1072 上	'1'	31 H		
D1073 下	'0'	30 H		
D1073 上	'0'	30 H		
D1074 下	'1'	31 H	数据内容 Data content	
D1074 上	'7'	37 H		
D1075 下	'7'	37 H		
D1075 上	'0'	30 H		
D1076 下	'7'	37 H	LRC CHK 1	LRC CHK 0
D1076 上	'1'	31 H	LRC CHK 0	

5 应用指令分类及基本使用

程式范例 (二)

◆ DVP-PM 与 VFD-B 系列变频器联机 (RTU Mode, M1143=On)



DVP-PM ⇒ VFD-B, DVP-PM 传送: **01 06 2000 0012 02 07**

VFD-B ⇒ DVP-PM, DVP-PM 接收: **01 06 2000 0012 02 07**

DVP-PM 传送数据寄存器 (传送讯息)

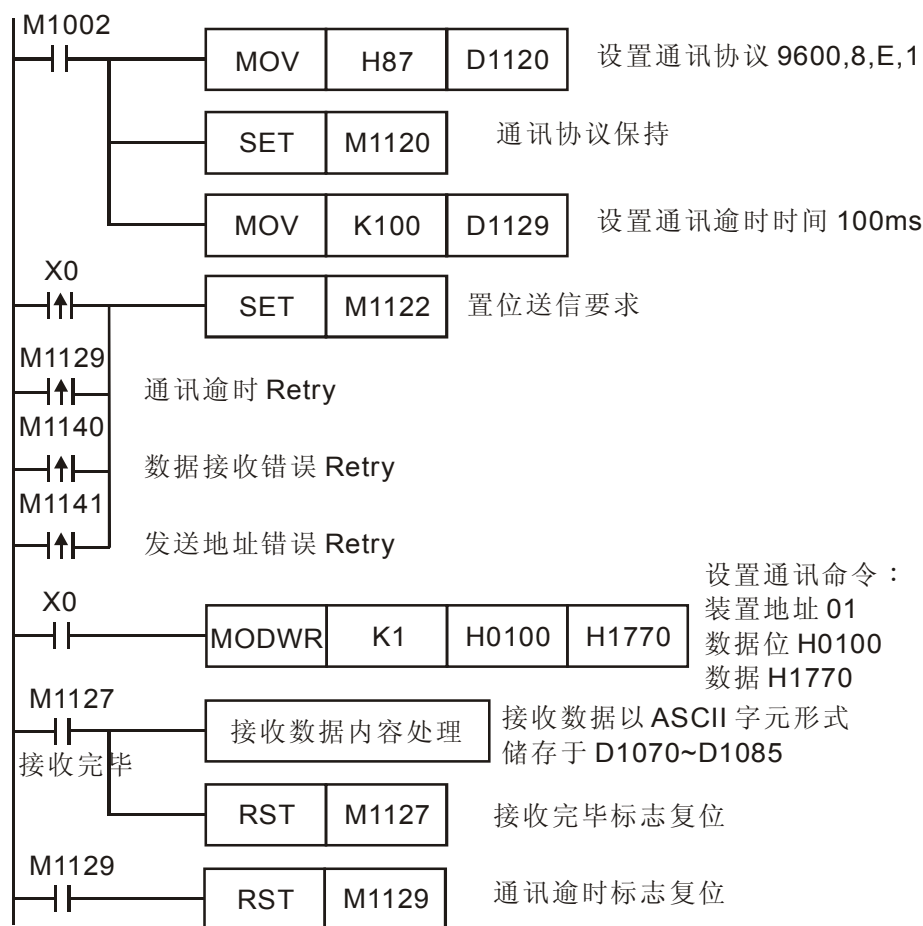
寄存器	DATA	说明
D1089 下	01 H	Address
D1090 下	06 H	Function
D1091 下	20 H	数据地址
D1092 下	00 H	Data Address
D1093 下	00 H	数据内容
D1094 下	12 H	Data content
D1095 下	02 H	CRC CHK Low
D1096 下	07 H	CRC CHK High

DVP-PM 接收数据寄存器 (响应讯息)

寄存器	DATA	说明
D1070 下	01 H	Address
D1071 下	06 H	Function
D1072 下	20 H	数据地址
D1073 下	00 H	Data Address
D1074 下	00 H	数据内容
D1075 下	12 H	Data content
D1076 下	02 H	CRC CHK Low
D1077 下	07 H	CRC CHK High

程式范例
(三)

- ◆ DVP-PM 与 VFD-B 系列变频器联机 (ASCII Mode, M1143=Off), 当通讯超时、接收数据错误及发送地址错误之 Retry。
- ◆ 当 X0=On 时, DVP-PM 将数据 H1770 (K6000) 写入装置地址 01 之 VFD-B 变频器数据地址 H0100 内。
- ◆ 若通讯超时则 M1129 旗标为 On, 程序中由 M1129 触发送信要求 M1122 再写入一次。
- ◆ 若数据接收错误则 M1140 旗标为 On, 程序中由 M1140 触发送信要求 M1122 再写入一次。
- ◆ 若发送地址错误则 M1141 旗标为 On, 程序中由 M1141 触发送信要求 M1122 再写入一次。

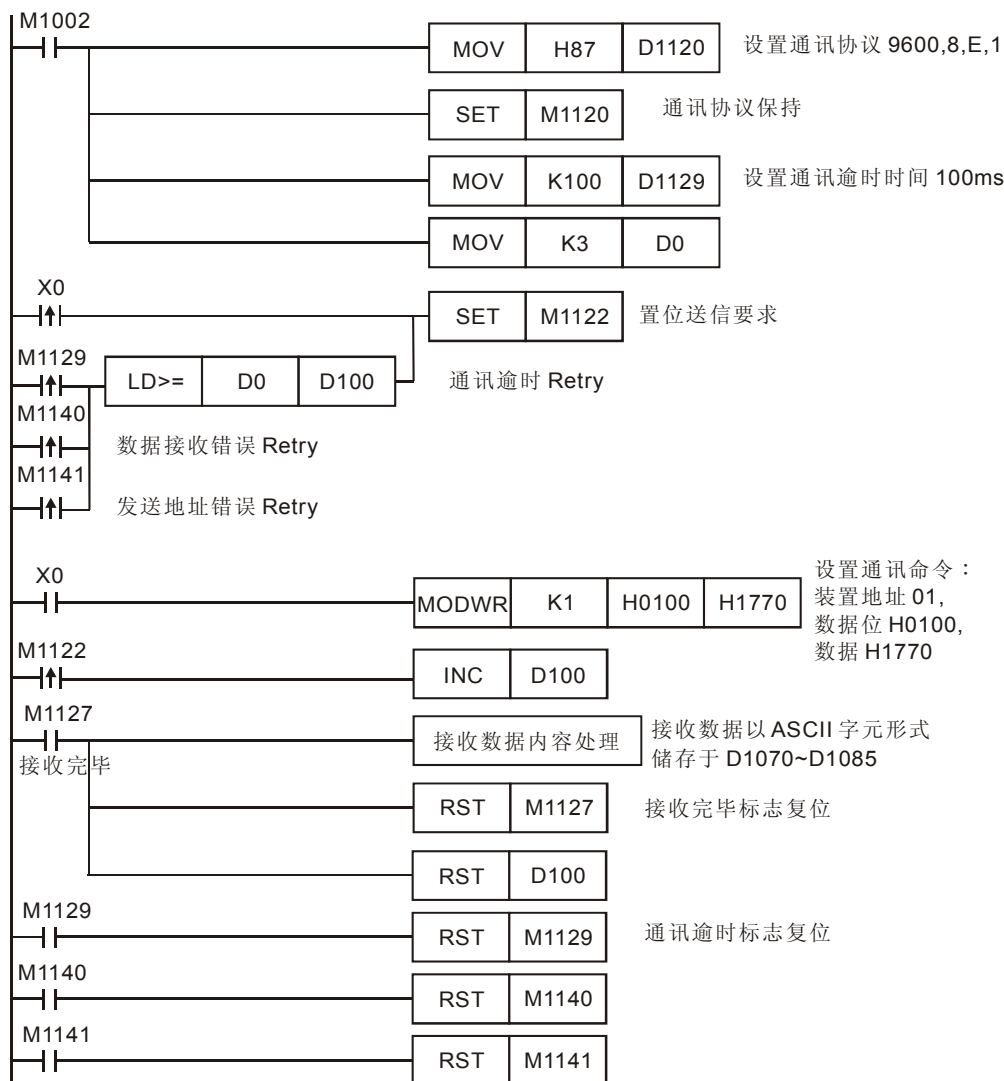


程式范例
(四)

- ◆ DVP-PM 与 VFD-B 系列变频器联机 (ASCII Mode, M1143=Off), 当通讯超时、接收数据错误及发送地址错误之 Retry, Retry 次数 D0, 预设 3 次。当通讯 Retry 成功则恢复由使用者触发条件来控制。
- ◆ 当 X0=On 时, DVP-PM 将资料 H1770 (K6,000) 写入装置地址 01 之 VFD-B 变频器数据地址 H0100 内。

5 应用指令分类及基本使用

- ◆ 若通讯超时则 M1129 旗标为 On，程序中由 M1129 触发送信要求 M1122 再写入一次，Retry 次数 D0，预设 3 次。
- ◆ 若数据接收错误则 M1140 旗标为 On，程序中由 M1140 触发送信要求 M1122 再写入一次，Retry 次数 D0，预设 3 次。
- ◆ 若发送地址错误则 M1141 旗标为 On，程序中由 M1141 触发送信要求 M1122 再写入一次，Retry 次数 D0，预设 3 次。



补充说明

- ◆ API 101 MODWR (Function Code H06、H10) 指令前面启动条件使用接点上升缘 (LDP, ANDP, ORP) 及接点下降缘 (LDF, ANDF, ORF)，须先启动送信要求 M1122，才可正确动作。
- ◆ 本指令于程序中使用次数并无限制，但是同时间仅有一个指令被执行。

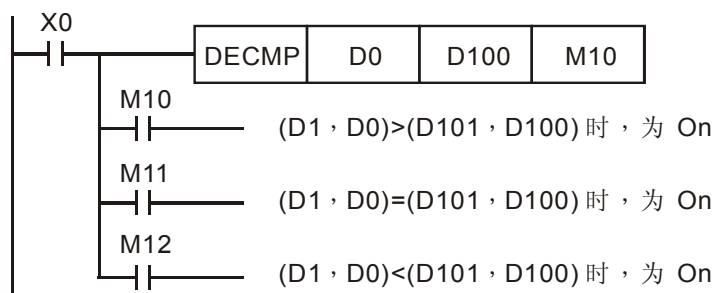
API																适用機種		
110	D	ECMP	P				(S₁)	(S₂)	(D)	二进制浮点数比较						20PM	10PM	
																✓	✓	
		位装置				字符装置										16 位指令		
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S1						*								*			-	
S2						*								*			-	
D		*	*	*													-	
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DECMP、DECMPP 有效 D 操作数会占用连续 3 点 F 表示浮点数输入，使用时要有小数点 																32 位指令 (9 STEP) DECMP 连续执行型 DECMPP 脉冲执行型 • 旗标信号：无		

指令说明

- ◆ **S₁**：二进制浮点数比较值 1。 **S₂**：二进制浮点数比较值 2。 **D**：比较结果，占用连续 3 点。
- ◆ 二进制浮点数值 1 与二进制浮点数比较值 2 作比较，比较的结果 (>、=、<) 在 **D** 作表示。
- ◆ **S₁** 或 **S₂** 来源操作数若是指定浮点数 F 的话，指令会以该二进制浮点数值来作比较。

程式范例

- ◆ 指定装置为 M10，则自动占有 M10~M12。
- ◆ 当 X0=On 时，**DECMP** 指令执行，M10~M12 其中之一会 On，当 X0=Off 时，**DECMP** 指令不执行，M10~M12 状态保持在 X0=Off 之前的状态。
- ◆ 若需要得到 ≥、≤、≠ 之结果时，可将 M10~M12 串并联即可取得。
- ◆ 若要清除其结果请使用 **RST** 或 **ZRST** 指令。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

5 应用指令分类及基本使用

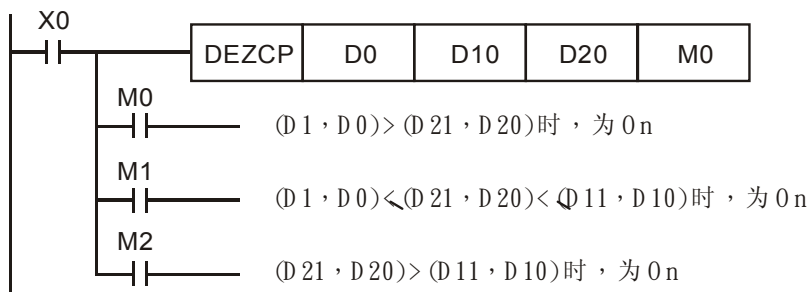
API																	适用機種		
111	D	EZCP	P	(S ₁) (S ₂) (S) (D)				二进制浮点数区域比较										20PM	10PM
														✓	✓				
		位装置			字符装置													16 位指令	
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S ₁					*									*			-		
S ₂					*									*			-		
S					*									*			-		
D		*	*	*													-		
<p>• 操作数使用注意: D 操作数会占用连续 3 点 S₁ 操作数内容值请小于 S₂ 操作数内容值 各装置使用范围请参考功能规格表 F 表示浮点数输入, 使用时要有小数点 本指令只有 32 位指令 DEZCP、DEZCPP 有效</p>																		<p>32 位指令 (12 STEP) DEZCP 连续执行型 DEZCPP 脉冲执行型</p> <p>• 旗标信号: 无</p>	

指令说明

- ◆ **S₁**: 区域比较之二进制浮点数下限值。**S₂**: 区域比较之二进制浮点数上限值。**S**: 二进制浮点数比较值。**D**: 比较结果, 占用连续 3 点。
- ◆ 二进制浮点数比较值 **S** 与二进制浮点数下限值 **S₁** 及二进制浮点数上限值 **S₂** 作比较, 其比较结果在 **D** 作表示。
- ◆ **S₁** 或 **S₂** 来源操作数若是指定浮点数 **F** 的话, 指令会以该二进制浮点数值来作比较。
- ◆ 当二进制浮点数下限值 **S₁** 大于二进制浮点数上限值 **S₂** 时, 则指令以二进制浮点数下限值 **S₁** 作为上下限值进行比较。

程式范例

- ◆ 指定装置为 M0, 则自动占有 M0~M2。
- ◆ 当 X0=On 时, **DEZCP** 指令执行, M0~M2 其中之一会 On, 当 X0=Off 时, **EZCP** 指令不执行, M0~M2 状态保持在 X0=Off 之前的状态。
- ◆ 若要清除其结果请使用 **RST** 或 **ZRST** 指令。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API																适用機種		
112	D	MOVR	P	S	D	浮点数值数据移动										20PM	10PM	
						✓	✓											
S	位装置				字符装置												16 位指令 (7 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	-	
D							*	*	*	*	*	*				32 位指令		
																DMOVR 连续执行型 DMOVR P 脉冲执行型		
<ul style="list-style-type: none"> 操作数使用注意： 所有装置不支持 V、Z 修饰 各装置使用范围请参考功能规格表 指定 KNX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制) 																<ul style="list-style-type: none"> 旗标信号：无 		

指令说明

- ◆ **S**：浮点数值资料来源。**D**：数据之搬移目的地。
- ◆ 该指令可直接在 **S** 操作数输入浮点数值。
- ◆ 当该指令执行时，将 **S** 的内容直接搬移至 **D**，当指令不执行时，**D** 内容不会变化。

程式范例

- ◆ 当 X0= Off 时，D10，D11 内容没有变化。当 X0=On 时，将 F1.2 浮点数现在值传送至 D10，D11 数据寄存器内。



5 应用指令分类及基本使用

API																	适用機種	
116	D	RAD	P		S	D	角度→徑度										20PM	10PM
																	✓	✓
S	位裝置				字裝置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	---		
D					*								*			32 位指令 (6 STEP)		
													*			DRAD 连续執行型 DRADP 脈衝執行型		
<ul style="list-style-type: none"> 操作数使用注意：各裝置使用範圍請參考功能規格表。 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DRAD、DRADP 有效 														<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明 				

指令说明

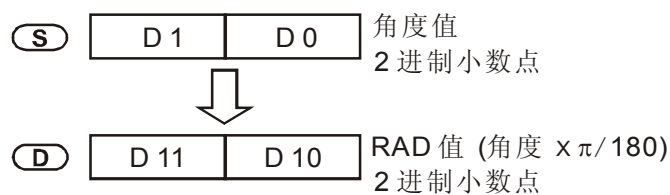
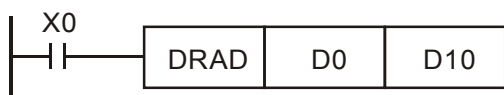
- ◆ S: 资料来源 (角度)。D: 变换之结果 (径度)。
- ◆ 使用下列公式将角度转换成径度。

$$\text{徑度} = \text{角度} \times (\pi / 180)$$

- ◆ 若转换结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若转换结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若转换结果为 0，则零旗号=On。

程式范例

- ◆ 当 X0=On 时，指定二进制浮点数 (D1, D0) 之角度值，将角度转换成径度值后存于 (D11, D10) 当中，内容为二进制浮点数。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API																	适用机种		
117	D	DEG	P	S	D													20PM	10PM
																		✓	✓
	位装置				字符装置												16 位指令		
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-			
S					*								*			-			
D													*			-			
<p>操作数使用注意：各装置使用范围请参考功能规格表。</p> <p>F 表示浮点数输入，使用时要有小数点</p> <p>本指令只有 32 位指令 DDEG、DDEGP 有效</p>																<p>32 位指令 (6 STEP)</p> <p>DDEG 连续执行型 DDEGP 脉冲执行型</p>			
<p>◆ 旗标信号：</p> <p>Ox O100</p> <p>M1808 M1968 零旗标 Zero flag</p> <p>M1809 M1969 借位旗标 Borrow flag</p> <p>M1810 M1970 进位旗标 Carry flag</p> <p>请参考下列补充说明</p>																			

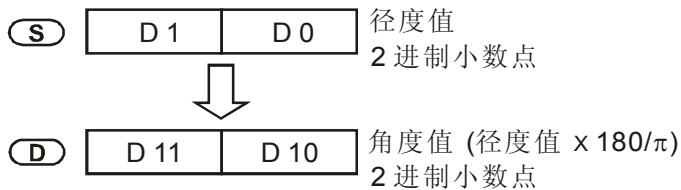
指令说明

- ◆ **S**：资料来源 (径度)。**D**：变换之结果 (角度)。
- ◆ 使用下列公式将径度转换成角度。

$$\text{角度} = \text{径度} \times (180/\pi)$$
- ◆ 若转换结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若转换结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若转换结果为 0，则零旗号=On。

程式范例

- ◆ 当 X0=On 时，指定二进制浮点数 (D1, D0) 之角度值，将径度值转换成角度后存于 (D11, D10) 当中，内容为二进制浮点数。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

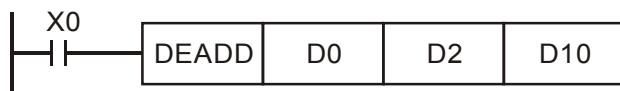
5 应用指令分类及基本使用

API					二进制浮点数加算										适用機種		
120	D	EADD	P	(S ₁) (S ₂) (D)											20PM	10PM	
															✓	✓	
	位装置				字符装置										16 位指令		
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S ₁					*								*			-	
S ₂					*								*			-	
D													*			-	
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DEADD、DEADDP 有效 																16 位指令 - - - 32 位指令 (9 STEP) DEADD 连续执行型 DEADDP 脉冲执行型 • 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明	

指令说明

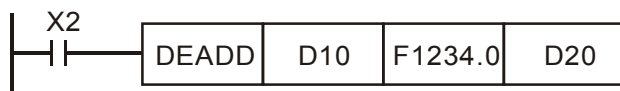
- ◆ **S₁**：被加数。**S₂**：加数。**D**：和。
- ◆ **S₁** 所指定的寄存器内容加上 **S₂** 所指定的寄存器内容，和被存放至 **D** 所指定的寄存器当中，加算的动作全部以二进制浮点数型态进行。
- ◆ **S₁** 或 **S₂** 来源操作数若是指定浮点数 F 的话，指令会以该二进制浮点数值来作加算。
- ◆ **S₁** 及 **S₂** 可指定相同的寄存器编号，此种情况下若是使用“连续执行”型态的指令时，在条件接点 On 的期间，该寄存器于每一次扫描时，均会被加算一次，一般的情况下都是使用脉冲执行型指令 (**DEADDP**)。
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若运算结果为 0，则零旗号=On。
- ◆ 当 X0=On 时，将二进制浮点数 (D11, D10) + 二进制浮点数 (D3, D2)，结果存放在 (D11, D10) 中。

程式范例 (一)



程式范例 (二)

- ◆ 当 X2=On 时，将二进制浮点数 (D11, D10) + F1234.0 (自动变换为二进制浮点数)，结果存放在 (D21, D20) 中。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

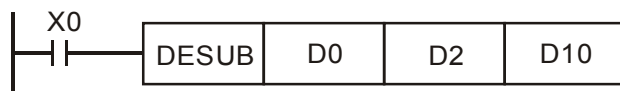
API																适用机种		
121	D	ESUB	P		(S ₁)	(S ₂)	(D)	二进制浮点数减算								20PM	10PM	
																✓	✓	
		位装置				字符装置										16 位指令		
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	32 位指令 (9 STEP)	
S ₁						*								*			DESUB 连续执行型 DESUBP 脉冲执行型	
S ₂						*								*				
D														*				
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DEADD、DEADDP 有效 															<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 			
请参考下列补充说明																		

指令说明

- ◆ S₁：被减数。S₂：减数。D：差。
- ◆ S₁ 所指定的寄存器内容减掉 S₂ 所指定的寄存器内容，差被存放至 D 所指定的寄存器当中，减算的动作全部以二进制浮点数型态进行。
- ◆ S₁ 或 S₂ 来源操作数若是指定浮点数 F 的话，指令会以该二进制浮点数值来作减算。
- ◆ S₁ 及 S₂ 可指定相同的寄存器编号，此种情况下若是使用“连续执行”型态的指令时，在条件接点 On 的期间，该寄存器于每一次扫描时，均会被减算一次，一般的情况下都是使用脉冲执行型指令 (DESUBP)。
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若运算结果为 0，则零旗号=On。

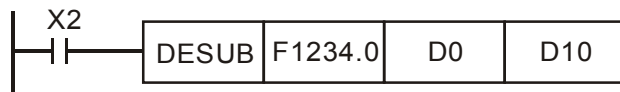
程式范例 (一)

- ◆ 当 X0=On 时，将二进制浮点数 (D1, D0) - 二进制浮点数 (D3, D2)，结果存放在 (D11, D10) 中。



程式范例 (二)

- ◆ 当 X2=On 时，将 F1234.0 (自动变换为二进制浮点数) - 二进制浮点数 (D1, D0)，结果存放在 (D11, D10) 中。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

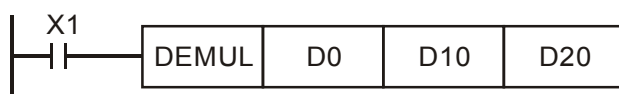
5 应用指令分类及基本使用

API																适用機種			
122	D	EMUL	P		(S ₁)	(S ₂)	(D)	二进制浮点数乘算								20PM	10PM		
																	✓	✓	
	位装置				字符装置											16 位指令			
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-			
S ₁					*								*			-			
S ₂					*								*			-			
D													*			-			
• 操作数使用注意：各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DEMUL 、 DEMULP 有效													• 旗标信号： Ox O100 M1810 M1970 进位旗标 Carry flag 请参考下列补充说明						

指令说明

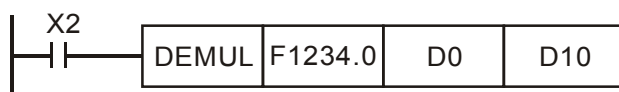
- ◆ **S₁**：被乘数。**S₂**：乘数。**D**：积。
- ◆ **S₁** 所指定的寄存器内容乘上 **S₂** 所指定的寄存器内容，积被存放至 **D** 所指定的寄存器当中，乘算的动作全部以二进制浮点数型态进行。
- ◆ **S₁** 或 **S₂** 来源操作数若是指定浮点数 **F** 的话，指令会以该二进制浮点数值来作乘算。
- ◆ **S₁** 及 **S₂** 可指定相同的寄存器编号，此种情况下若是使用“连续执行”型态的指令时，在条件接点 **On** 的期间，该寄存器于每一次扫描时，均会被乘算一次，一般的情况下都是使用使用脉冲执行型指令 (**DEMULP**)。
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若运算结果为 0，则零旗号=On。
- ◆ 当 **X1=On** 时，将二进制浮点数 (**D11, D10**) 乘上二进制浮点数 (**D21, D20**) 将积存放至 (**D21, D20**) 所指定的寄存器当中。

程式范例 (一)



程式范例 (二)

- ◆ 当 **X2=On** 时，将 **F1234.0** (自动变换为二进制浮点数) × 二进制浮点数 (**D11, D10**)，结果存放在 (**D11, D10**) 中。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

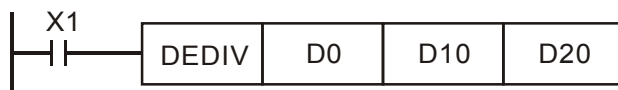
API																适用机种	
123	D	EDIV	P		(S ₁)	(S ₂)	(D)	二进制浮点数除算								20PM	10PM
																✓	✓
	位装置				字符装置										16 位指令		
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S ₁					*								*			-	
S ₂					*								*			-	
D													*			-	
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DEDIV、DEDIVP 有效 													<ul style="list-style-type: none"> 32 位指令 (9 STEP) DEDIV 连续执行型 DEDIVP 脉冲执行型 				
<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag M1793 M1953 演算错误旗标 Operation error flag 													<ul style="list-style-type: none"> 请参考下列补充说明 				

指令说明

- ◆ **S₁**：被除数。**S₂**：除数。**D**：商及余数。
- ◆ **S₁** 所指定的寄存器内容除以 **S₂** 所指定的寄存器内容，商被存放至 **D** 所指定的寄存器当中，除算的动作全部以二进制浮点数型态进行。
- ◆ **S₁** 或 **S₂** 来源操作数若是指定浮点数 **F** 的话，指令会以该二进制浮点数值来作除算。
- ◆ 除数 **S₂** 的内容若为 0 即被认定为“运算错误”，指令不执行，演算错误旗标 ON，并记录错误码 H'0E19。
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若运算结果为 0，则零旗号=On。

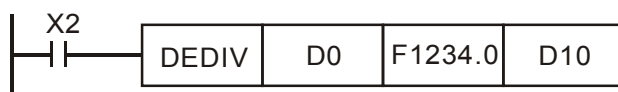
程式范例 (一)

- ◆ 当 X1=On 时，将二进制浮点数 (D1, D0) 除以二进制浮点数 (D11, D10) 将商存放至 (D21, D20) 所指定的寄存器当中。



程式范例 (二)

- ◆ 当 X2=On 时，将二进制浮点数 (D1, D0) ÷ F1234.0 (自动变换为二进制浮点数)，结果存放在 (D11, D10) 中。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

5 应用指令分类及基本使用

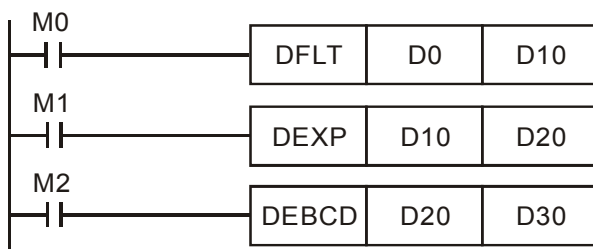
API																	适用機種	
124	D	EXP	P			(S)	(D)	二进制浮点数取指数									20PM	10PM
																	✓	✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S					*								*			-		
D													*			-		
<p>• 操作数使用注意：各装置使用范围请参考功能规格表</p> <p>F 表示浮点数输入，使用时要有小数点</p> <p>本指令只有 32 位指令 DEXP、DEXPP 有效</p>														<p>• 旗标信号：</p> <p>Ox O100</p> <p>M1808 M1968 零旗标 Zero flag</p> <p>M1809 M1969 借位旗标 Borrow flag</p> <p>M1810 M1970 进位旗标 Carry flag</p> <p>请参考下列补充说明</p>				

指令说明

- ◆ **S**：运算来源装置。**D**：运算结果装置。
- ◆ $EXP^{[D+1, D]} = [S+1, S]$ ，以 $e = 2.71828$ 为底数，**S** 为指数做 EXP 运算。
- ◆ **S** 内容正负数都有效，指定 **D** 寄存器时必须使用 32 位数据格式，运算时均以浮点数方式执行，故 **S** 需为浮点数值。
- ◆ **D** 操作数内容值 $= e^S$ ； $e = 2.71828$ ，**S** 为指定的来源数据
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若运算结果为 0，则零旗号=On。

程式范例

- ◆ 当 M0 为 On 时，将 (D1, D0) 值转成二进制浮点数存于 (D11, D10) 寄存器中。
- ◆ 当 M1 为 On 时，(D11, D10) 为指数做 EXP 运算，其值为二进制浮点数值并存放于 (D21, D20) 寄存器中。
- ◆ 当 M2 为 On 时，将 (D21, D20) 二进制浮点数值转成十进浮点数值并存于 (D31, D30) 寄存器中。(此时 D31 为表示 D30 的 10 次幂方)



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API																适用机种		
125	D	LN	P			(S)	(D)	二进制浮点数取自然对数								20PM	10PM	
																✓	✓	
		位装置				字符装置										16 位指令		
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S						*								*			-	
D														*			-	
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DLN、DLNP 有效 																32 位指令 (6 STEP) DLN 连续执行型 DLNP 脉冲执行型		
<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag M1793 M1953 演算错误旗标 Operation error flag 																请参考下列补充说明		

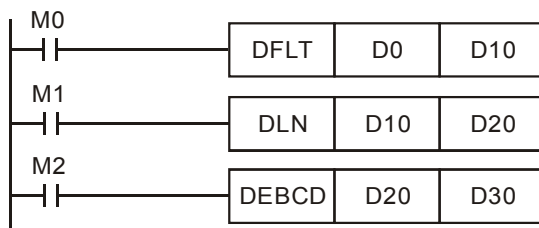
指令说明

- ◆ **S**：运算来源装置。**D**：运算结果装置。
- ◆ 以 **S** 为操作数做自然对数 \ln 运算。

$$\text{LN}[\text{S}+1, \text{S}] = [\text{D}+1, \text{D}]$$
- ◆ **S** 内容只有正数有效，指定 **D** 寄存器时必须使用 32 位数据格式，运算时均以浮点数方式执行，故 **S** 需为浮点数值。
- ◆ **S** 内容若不为正数即被认定为“运算错误”，指令不执行，演算错误旗标 ON，并记录错误码 H'0E19。
- ◆ $e^{\text{D}} = \text{S} \rightarrow \text{D}$ 操作数内容值 = $\ln \text{S}$ ；**S** 为指定的来源数据
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位旗号 = On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位旗号 = On。
- ◆ 若运算结果为 0，则零旗号 = On。

程式范例

- ◆ 当 M0 为 On 时，将 (D1, D0) 值转成二进制浮点数存于 (D11, D10) 寄存器中。
- ◆ 当 M1 为 On 时，将 (D11, D10) 寄存器为真数做 \ln 运算，其值为二进制浮点数并存放于 (D21, D20) 寄存器中。
- ◆ 当 M2 为 On 时，将二进制浮点数值转成十进浮点数值并存于 (D31, D30) 寄存器中。(此时 D31 为表示 D30 的 10 次幂方)



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

5 应用指令分类及基本使用

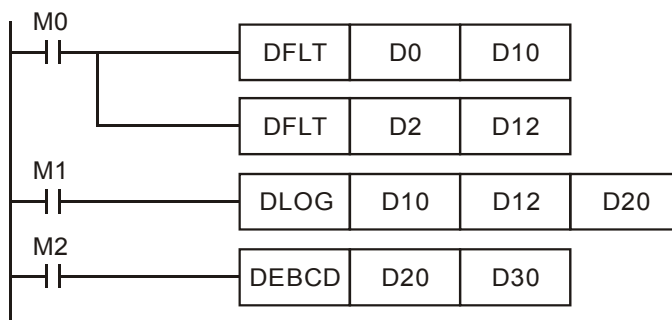
API																适用機種	
126	D	LOG	P	(S ₁)	(S ₂)	(D)	二进制浮点数取对数									20PM	10PM
																✓	✓
	位装置				字符装置											16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S ₁					*								*			-	
S ₂					*								*			-	
D													*			-	
操作数使用注意：各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DLOG 、 DLOGP 有效													32 位指令 (9 STEP) DLOG 连续执行型 DLOGP 脉冲执行型				
旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag													请参考下列补充说明				

指令说明

- ◆ **S₁**：运算底数装置。**S₂**：运算来源装置。**D**：运算结果装置。
- ◆ 将 **S₁** 内容及 **S₂** 内容为操作数做 log 运算，结果存放于 **D**。
- ◆ **S₁**、**S₂** 内容值只有正数有效，指定 **D** 寄存器时必须使用 32 位数据格式，运算时均以浮点数方式执行，故 **S₁**、**S₂** 需为浮点数值。
- ◆ $S_1^D = S_2$ ，求 D 值 $\rightarrow \text{Log}_{S_1} S_2 = D$
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若运算结果为 0，则零旗号=On。

程式范例

- ◆ 当 M0 为 On 时，将 (D1, D0) 内容及 (D3, D2) 内容转成二进制浮点数分别存于 (D11, D10) 及 (D13, D12) 32 位寄存器中。
- ◆ 当 M1 为 On 时，将 (D11, D10) 及 (D13, D12) 32 位寄存器二进制浮点数值做 log 运算并将结果存于 (D21, D20) 32 位寄存器中。
- ◆ 当 M2 为 On 时，将 (D21, D20) 32 位寄存器二进制浮点数值转成十进浮点数值并存于 (D30, D31) 中。(此时 D31 为表示 D30 的 10 次幂方)



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API																	适用机种			
127	D	ESQR	P			(S)	(D)	二进制浮点数开平方根										20PM	10PM	
																		✓	✓	
	位装置				字符装置												16 位指令			
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-				
S					*								*			-				
D													*			-				
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 S 操作数范围须大于等于 0 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DESQR、DESQRP 有效 																<ul style="list-style-type: none"> 32 位指令 (6 STEP) DESQLR 连续执行型 DESQRP 脉冲执行型 				
<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1793 M1953 演算错误旗标 Operation error flag 																				
请参考下列补充说明																				

指令说明

- ◆ **S**：欲开平方根来源装置。 **D**：开平方根之结果。
- ◆ **S** 所指定的寄存器内容被开平方，所得的结果暂存于 **D** 所指定的寄存器内容，开平方的动作全部以二进制浮点数值态进行。
- ◆ **S** 来源操作数若是指定浮点数 **F** 的话，指令会将该浮点数转为二进制浮点数值来作运算。
- ◆ 若开平方根之结果为 0 时，零旗标=On。
- ◆ 来源运算只有正数有效，负数时，视为“运算错误”，指令不执行，演算错误旗标 ON，并记录错误码 H'0E19。

程式范例 (一)

- ◆ 当 X0=On 时，将二进制浮点数 (D1, D0) 取开平方根，将结果存放至 (D11, D10) 所指定的寄存器当中。



$$\sqrt{(D1 \cdot D0)} \rightarrow (D11 \cdot D10)$$

2 进制小数点 2 进制小数点

程式范例 (二)

- ◆ 当 X2=On 时，将 F1234.0 (自动变换为二进制浮点数) 取开平方根，结果存放在 (D11, D10) 中。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

5 应用指令分类及基本使用

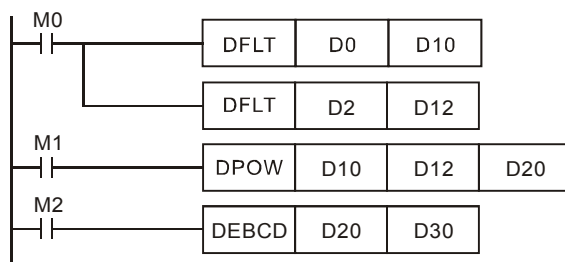
API																适用机种		
128	D	POW	P	(S ₁)	(S ₂)	(D)	浮点数权值指令										20PM	10PM
																	✓	✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S ₁					*								*			-		
S ₂					*								*			-		
D													*			-		
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DPOW、DPOWP 有效 F 表示浮点数输入，使用时要有小数点 																<ul style="list-style-type: none"> 32 位指令 (9 STEP) DPOW 连续执行型 DPOWP 脉冲执行型 		
<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag M1793 M1953 演算错误旗标 Operation error flag 																		
请参考下列补充说明																		

指令说明

- ◆ **S₁**：底数装置。**S₂**：次幂数装置。**D**：运算结果装置。
- ◆ 将二进制浮点数据 **S₁** 及 **S₂** 以次幂数相乘后存放于 **D**。
$$D = POW [S_1+1, S_1] [S_2+1, S_2]$$
- ◆ **S₁** 内容值只有正数有效，**S₂** 内容值正负值都有效。指定 **D** 寄存器时必须使用 32 位数据格式，运算时均以浮点数方式执行，故 **S₁**、**S₂** 需为浮点数值。
- ◆ 若 **S₁** 及 **S₂** 的内容值无效，视为“运算错误”，指令不执行，演算错误旗标 ON，并记录错误码 H'0E19。
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位旗号=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位旗号=On。
- ◆ 若运算结果为 0，则零旗号=On。

程式范例

- ◆ 当 M0 为 On 时，将 (D1, D0) 内容及 (D3, D2) 内容转成二进制浮点数分别存于 (D11, D10) 及 (D13, D12) 32 位寄存器中。
- ◆ 当 M1 为 On 时，将 (D11, D10) 及 (D13, D12) 32 位寄存器二进制浮点数做 pow 运算并将结果存于 (D21, D20) 32 位寄存器中。
- ◆ 当 M2 为 On 时，将 (D21, D20) 32 位寄存器二进制浮点数值转成十进浮点数值并存于 (D31, D30) 寄存器中。(此时 D31 为表示 D30 的 10 次幂方)



补充说明

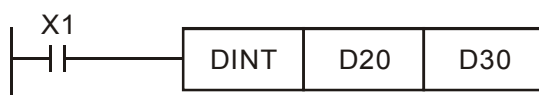
- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API																适用機種		
129	D	INT	P	S	D	二进制浮点数→BIN 整数变换										20PM	10PM	
																	✓	✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S													*			32 位指令 (5 STEP)		
D													*			DINT 连续执行型 DINTP 脉冲执行型		
操作数使用注意：各装置使用范围请参考功能规格表 此指令只有 32 位指令 DINT、DINTP 有效													旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列指令说明					

指令说明

- ◆ **S**：欲变换之来源装置。**D**：变换之结果。
- ◆ **S** 所指定的寄存器内容以二进制浮点数型态被变换成 BIN 整数暂存于 **D** 所指定的寄存器当中，BIN 整数浮点数被舍弃。
- ◆ 本指令的动作与 API 49 DFLT 指令刚好相反。
- ◆ 变换结果若为 0 时，零旗标=On。
变换结果有浮点数被舍弃时，借位旗标=On。
变换结果若超出下列范围时 (进位)，进位旗标=On。
32 位指令：-2,147,483,648~2,147,483,647

程式范例



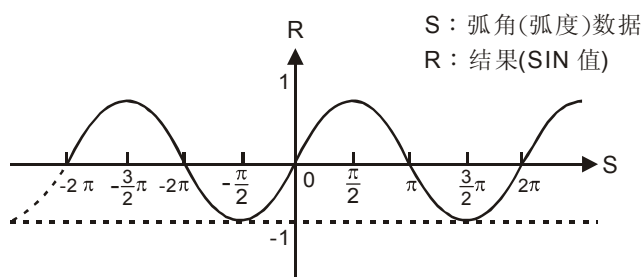
5 应用指令分类及基本使用

API																适用機種		
130	D	SIN	P		(S)	(D)	二进制浮点数 SIN 运算									20PM	10PM	
																✓	✓	
		位装置			字符装置											16 位指令		
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S						*								*			32 位指令 (6 STEP)	
D														*			DSIN 连续执行型 DSINP 脉冲执行型	
<ul style="list-style-type: none"> 操作数使用注意：角度范围：$0^\circ \leq \text{角度} < 360^\circ$ 各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DSIN、DSINP 有效 														<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1760 M1920 径度/角度使用旗标 请参考下列补充说明 				

指令说明

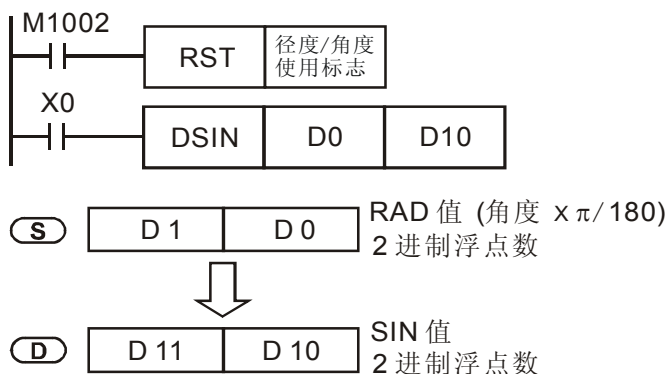
- ◆ **S**：指定的来源值。 **D**：取 SIN 值结果。
- ◆ **S** 所指定的来源可指定为径度或角度，由径度/角度使用旗标决定。
- ◆ 当径度/角度使用旗标=Off 时，指定为径度模式，径度 (RAD) 值等于 (角度 $\times \pi / 180$)。
- ◆ 当径度/角度使用旗标=On 时，指定为角度模式，角度范围： $0^\circ \leq \text{角度值} < 360^\circ$ 。
- ◆ 当计算结果若为 0 时，零旗标=ON。
- ◆ 将 **S** 所指定的来源值，求取 SIN 值后存于 **D** 所指定的寄存器当中。

下图显示弧角与结果的关系：



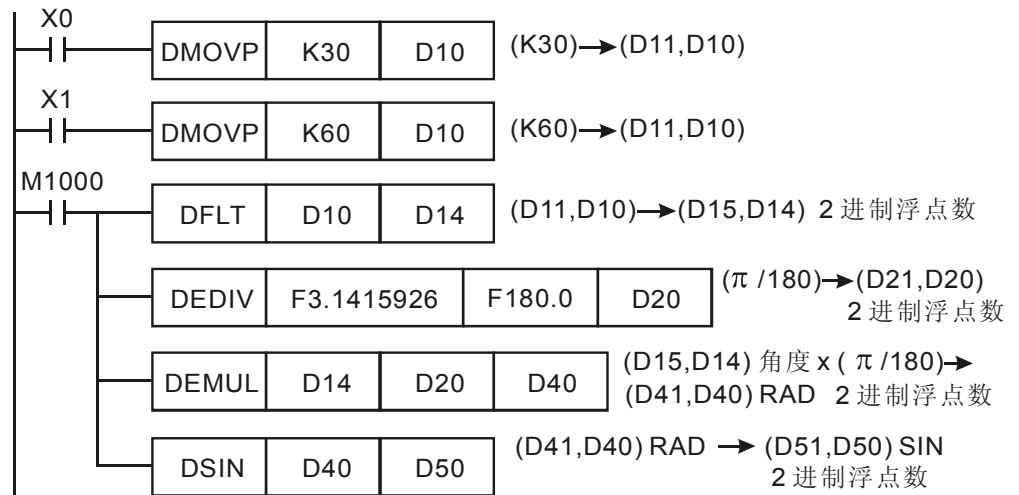
程式范例 (一)

- ◆ 径度/角度使用旗标=Off, 指定为径度模式, 当 X0=On 时, 指定二进制浮点数 (D1, D0) 之径度 (RAD) 值求取 SIN 值后存于 (D11, D10) 当中, 内容为二进制浮点数。



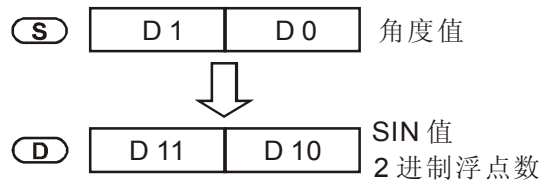
程式范例
(二)

- ◆ 径度/角度使用旗标=Off, 指定为径度模式, 由输入端 X0 及 X1 来选择角度, 转成径度 (RAD) 值后求取 SIN 值。



程式范例
(三)

- ◆ 径度/角度使用旗标=On, 指定为角度模式, 当 X0=On 时, 指定 (D1, D0) 之角度值, 角度范围: $0^\circ \leq \text{角度值} < 360^\circ$ 。求取 SIN 值后存于 (D11, D10) 当中, 内容为二进制浮点数。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

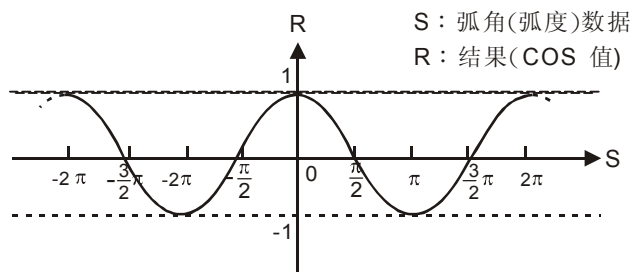
5 应用指令分类及基本使用

API																适用機種		
131	D	COS	P	(S)	(D)	二进制浮点数 COS 运算										20PM	10PM	
																✓	✓	
		位装置				字符装置										16 位指令		
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S						*								*			-	
D														*			-	
<ul style="list-style-type: none"> 操作数使用注意：角度范围：$0^\circ \leq \text{角度} < 360^\circ$ 各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DCOS、DCOSP 有效 														<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1760 M1920 径度/角度使用旗标 请参考下列补充说明 				

指令说明

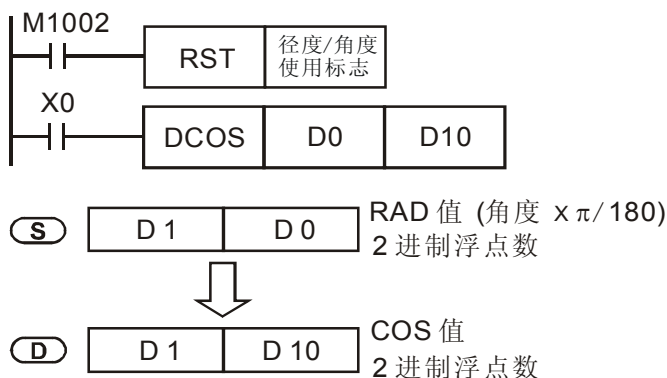
- ◆ **S**：指定的来源值。**D**：取 COS 值结果。
- ◆ **S** 所指定的来源可指定为径度或角度，由旗标径度/角度使用旗标决定。
- ◆ 当径度/角度使用旗标=Off 时指定为径度模式，径度值等于 $(\text{角度} \times \pi / 180)$ 。
- ◆ 当径度/角度使用旗标=On 时指定为角度模式，角度范围： $0^\circ \leq \text{角度值} < 360^\circ$ 。
- ◆ 当计算结果若为 0 时,零旗标=ON。
- ◆ 将 **S** 所指定的来源值，求取 COS 值后存于 **D** 所指定的寄存器当中。

下图显示弧角与结果的关系：



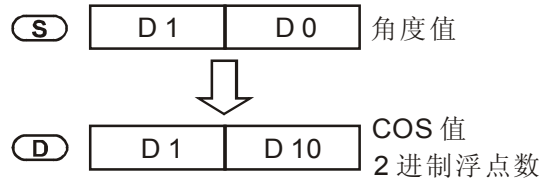
- ◆ 旗标径度/角度使用旗标径度/角度切换：当径度/角度使用旗标=Off 时，**S** 所指定的为径度 (RAD) 值。当径度/角度使用旗标=On 时，**S** 所指定的为角度值 ($0^\circ \sim 360^\circ$)。
- ◆ 径度/角度使用旗标=Off, 指定为径度模式, 当 X0=On 时, 指定二进制浮点数 (D1, D0) 之径度值求取 COS 值后存于 (D11, D10) 当中, 内容为二进制浮点数。

程式范例 (一)



程式范例
(二)

- ◆ 径度/角度使用旗标=On, 指定为角度模式, 当 X0=On 时, 指定 (D1, D0) 之角度值, 角度范围: $0^\circ \leq \text{角度值} < 360^\circ$ 。求取 COS 值后存于 (D11, D10) 当中, 内容为二进制浮点数。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

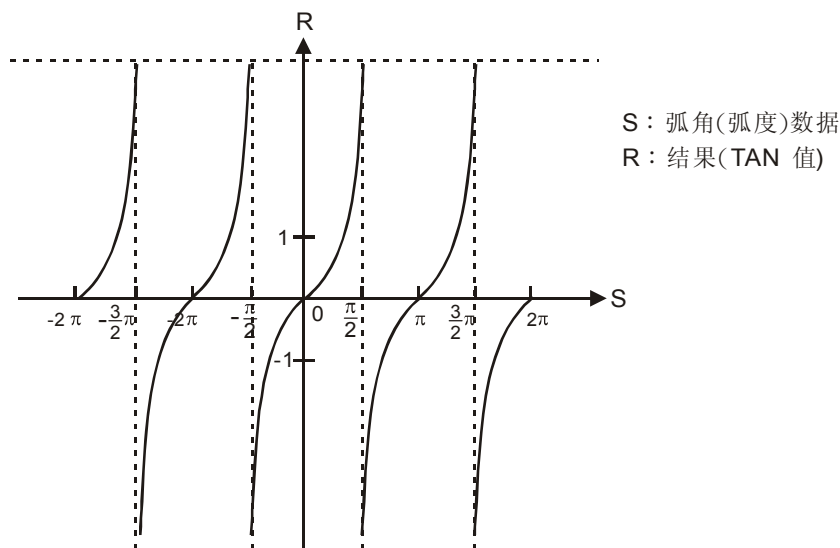
5 应用指令分类及基本使用

API																适用機種		
132	D	TAN	P		(S)	(D)	二进制浮点数 TAN 运算									20PM	10PM	
																✓	✓	
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S					*								*			-		
D													*			-		
<ul style="list-style-type: none"> 操作数使用注意：角度范围：$0^{\circ} \leq \text{角度} < 360^{\circ}$ 各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DTAN、DTANP 有效 														<ul style="list-style-type: none"> 32 位指令 (6 STEP) DTAN 连续执行型 DTANP 脉冲执行型 				
<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1760 M1920 弧度/角度使用旗标 请参考下列补充说明 																		

指令说明

- ◆ S: 指定的来源值。D: 取 TAN 值结果。
- ◆ S 所指定的来源可指定为弧度或角度，由旗标弧度/角度使用旗标决定。
- ◆ 当弧度/角度使用旗标=Off 时，指定为弧度模式，弧度 (RAD) 值等于 (角度 $\times \pi / 180$)。
- ◆ 当弧度/角度使用旗标=On 时，指定为角度模式，角度范围： $0^{\circ} \leq \text{角度值} < 360^{\circ}$ 。
- ◆ 当计算结果若为 0 时，零旗标=ON。
- ◆ 将 S 所指定的来源值，求取 TAN 值后存于 D 所指定的寄存器当中。

下图显示弧角与结果的关系：



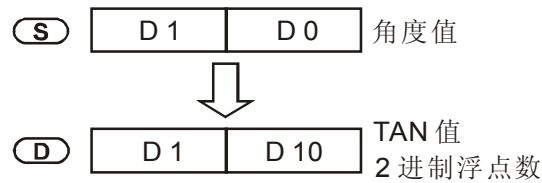
程式范例 (一)

- ◆ 弧度/角度使用旗标=Off, 指定为弧度模式, 当 X0=On 时, 指定二进制浮点数 (D11, D0) 之弧度 (RAD) 值求取 TAN 值后存于 (D11, D10) 当中, 内容为二进制浮点数。



程式范例
(二)

- ◆ 径度/角度使用旗标=On, 指定为角度模式, 当 X0=On 时, 指定 (D1, D0) 之角度值, 角度范围: $0^\circ \leq \text{角度值} < 360^\circ$ 。求取 TAN 值后存于 (D11, D10) 当中, 内容为二进制浮点数。



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

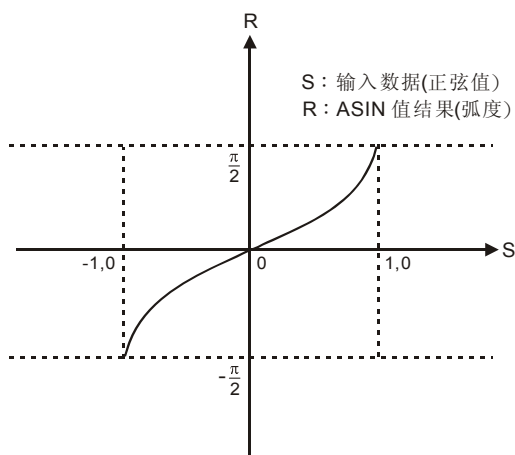
5 应用指令分类及基本使用

API											适用机种																																																																			
133	D	ASIN	P	S		D		二进制浮点数 ASIN 运算										20PM	10PM																																																											
				✓	✓																																																																									
<table border="1"> <tr> <th colspan="4">位装置</th> <th colspan="12">字符装置</th> </tr> <tr> <td>X</td><td>Y</td><td>M</td><td>S</td> <td>F</td><td>H</td><td>KnX</td><td>KnY</td><td>KnM</td><td>KnS</td><td>T</td><td>C</td><td>D</td><td>V</td><td>Z</td> </tr> <tr> <td>S</td><td></td><td></td><td></td> <td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td><td></td><td></td> </tr> <tr> <td>D</td><td></td><td></td><td></td> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td><td></td><td></td> </tr> </table>														位装置				字符装置												X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	S				*								*			D												*			16 位指令 — — — —			
位装置				字符装置																																																																										
X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z																																																																
S				*								*																																																																		
D												*																																																																		
32 位指令 (6 STEP) DASIN 连续执行型 DASINP 脉冲执行型														旗标信号: Ox O100 M1808 M1968 零旗标 Zero flag M1793 M1953 演算错误旗标 Operation error flag 请参考下列补充说明																																																																
操作数使用注意: 各装置使用范围请参考功能规格表 本指令只有 32 位指令 DASIN 、 DASINP 有效 F 表示浮点数输入, 使用时要有小数点																																																																														

指令说明

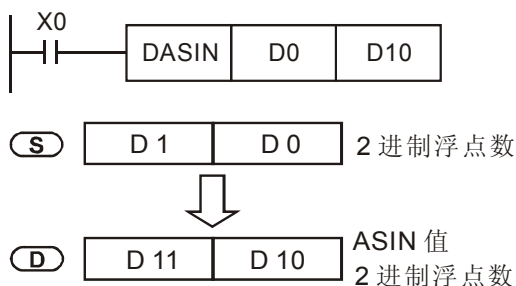
- ◆ **S**: 指定的来源 (二进制浮点数)。**D**: 取 ASIN 值结果。
- ◆ $ASIN \text{ 值} = \sin^{-1}$

下图显示输入数据与结果的关系:



- ◆ **S** 操作数指定的正弦值数值之十进浮点值只能介于 $-1.0 \sim +1.0$ 之间, 若不在此范围内则指令不执行, 演算错误旗标 ON, 并记录错误码 H'0E19。
- ◆ 若转换结果为 0, 则零旗号=On。
- ◆ 当 X0=On 时, 指定二进制浮点数 (D1, D0) 求取 ASIN 值后存于 (D11, D10) 当中, 内容为二进制浮点数。

程式范例



补充说明

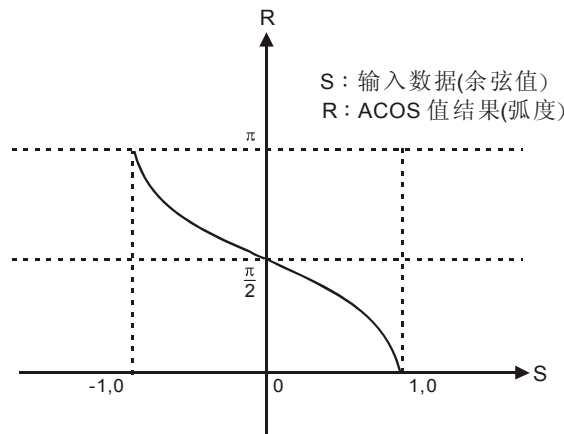
- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API					二进制浮点数 ACOS 运算										适用機種										
134	D	ACOS	P	S	D											20PM	10PM								
																	✓	✓							
				位装置										字符装置										16 位指令	
				X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-						
S								*								*			32 位指令 (6 STEP)						
D																*			DACOS 连续执行型 DACOSP 脉冲执行型						
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DACOS、DACOSP 有效 F 表示浮点数输入，使用时要有小数点 																<ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1793 M1953 演算错误旗标 Operation error flag 									
																请参考下列补充说明									

指令说明

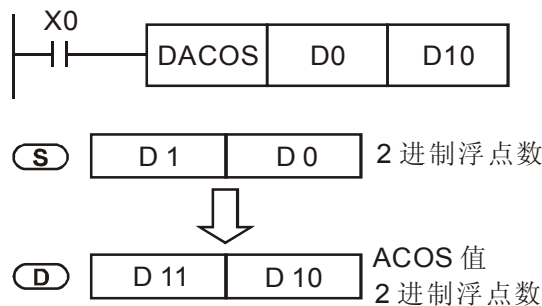
- ◆ **S**: 指定的来源 (二进制浮点数)。**D**: 取 ACOS 值结果。
- ◆ ACOS 值 = \cos^{-1}

下图显示输入数据与结果的关系：



- ◆ **S** 操作数指定的余弦值数值之十进浮点值只能介于 $-1.0 \sim 1.0$ 之间，若不在此范围内则指令不执行，演算错误旗标 ON，并记录错误码 H'0E19。
- ◆ 若转换结果为 0，则零旗号=On。
- ◆ 当 X0=On 时，指定二进制浮点数 (D1, D0) 求取 ACOS 值后存于 (D11, D10) 当中，内容为二进制浮点数。

程式范例



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

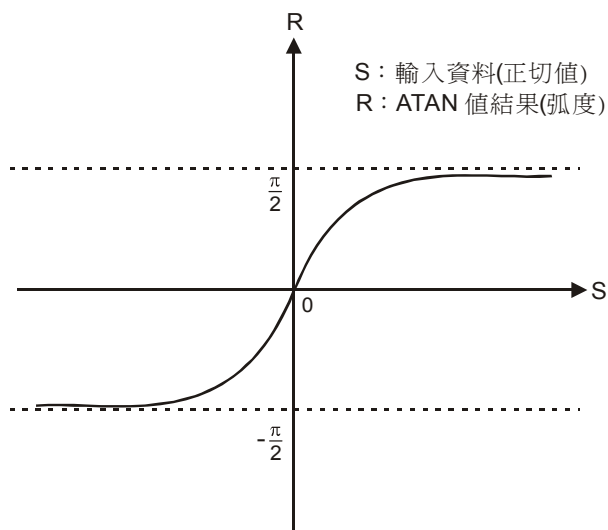
5 应用指令分类及基本使用

API 135	D	ATAN	P	(S) (D)	二进制浮点数 ATAN 运算										适用机种																																																															
																	20PM	10PM																																																												
																	✓	✓																																																												
				<table border="1"> <tr> <th colspan="4">位装置</th> <th colspan="12">字符装置</th> </tr> <tr> <td>X</td><td>Y</td><td>M</td><td>S</td> <td>F</td><td>H</td><td>KnX</td><td>KnY</td><td>KnM</td><td>KnS</td><td>T</td><td>C</td><td>D</td><td>V</td><td>Z</td> </tr> <tr> <td>S</td><td></td><td></td><td></td> <td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td><td></td><td></td> </tr> <tr> <td>D</td><td></td><td></td><td></td> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td><td></td><td></td> </tr> </table>												位装置				字符装置												X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	S				*								*			D												*			16 位指令 — — — — —	
位装置				字符装置																																																																										
X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z																																																																
S				*								*																																																																		
D												*																																																																		
				32 位指令 (6 STEP) DATAN 连续执行型 DATANP 脉冲执行型												旗标信号: Ox O100 M1808 M1968 零旗标 Zero flag 请参考下列补充说明																																																														
操作数使用注意: 各装置使用范围请参考功能规格表 本指令只有 32 位指令 DATAN、DATANP 有效 F 表示浮点数输入, 使用时要有小数点																																																																														

指令说明

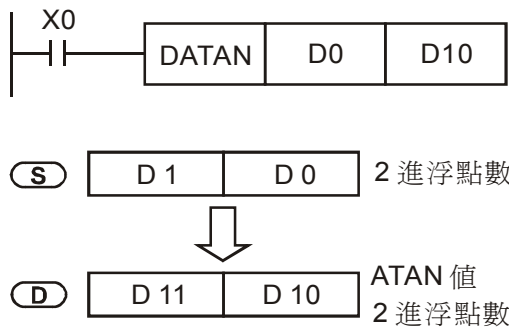
- ◆ **S:** 指定的来源 (二进制浮点数)。 **D:** 取 ATAN 值结果。
- ◆ ATAN 值 = \tan^{-1}

下图显示输入数据与结果的关系:



- ◆ 若转换结果为 0, 则零旗号=On。
- ◆ 当 X0=On 时, 指定二进制浮点数 (D1, D0) 求取 TAN 值后存于 (D11, D10) 当中, 内容为二进制浮点数。

程式范例



补充说明

- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API											适用機種							
136	D	SINH	P	S		D		二进制浮点数 SINH 运算						20PM	10PM			
				✓	✓													
		位装置				字符装置												
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	16 位指令	
S					*									*			— — —	
D														*			32 位指令 (6 STEP)	
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DSINH、DSINHP 有效 F 表示浮点数输入，使用时要有小数点 																DSINH 连续执行型 DSINHP 脉冲执行型		
<ul style="list-style-type: none"> 旗标信号： <ul style="list-style-type: none"> Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明																		

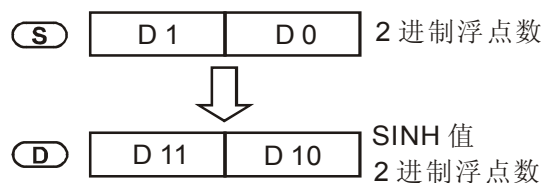
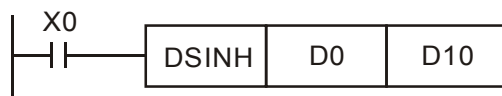
指令说明

◆ **S**: 指定的来源 (二进制浮点数)。 **D**: 取 SINH 值结果。

◆ \sinh 值 = $(e^s - e^{-s}) / 2$

程式范例

◆ 当 X0=On 时, 指定二进制浮点数 (D1, D0) 求取 SINH 值后存于 (D11, D10) 当中, 内容为二进制浮点数。



◆ 若转换结果的绝对值大于可表示之最大浮点值, 则进位旗号=On。

◆ 若转换结果的绝对值小于可表示之最小浮点值, 则借位旗号=On。

◆ 若转换结果为 0, 则零旗号=On。

补充说明

◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

5 应用指令分类及基本使用

API																	适用機種		
137	D	COSH	P		S	D	二进制浮点数 COSH 运算										20PM	10PM	
							✓	✓											
		位装置				字符装置										16 位指令			
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S						*								*			-		
D														*			-		
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DCOSH、DCOSH P 有效 F 表示浮点数输入，使用时要有小数点 										<ul style="list-style-type: none"> 32 位指令 (6 STEP) DCOSH 连续执行型 DCOSH P 脉冲执行型 旗标信号： <ul style="list-style-type: none"> Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明 									

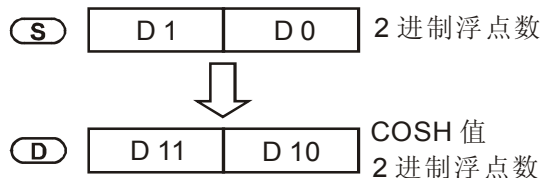
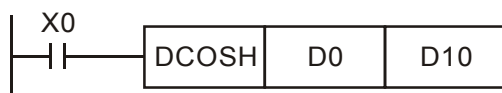
指令说明

◆ **S**：指定的来源（二进制浮点数）。 **D**：取 COSH 值结果。

◆ $\cosh \text{ 值} = (e^s + e^{-s}) / 2$

程式范例

◆ 当 X0=On 时，指定二进制浮点数 (D1, D0) 求取 COSH 值后存于 (D11, D10) 当中，内容为二进制浮点数。



◆ 若转换结果的绝对值大于可表示之最大浮点值，则进位旗号=On。

◆ 若转换结果的绝对值小于可表示之最小浮点值，则借位旗号=On。

◆ 若转换结果为 0，则零旗号=On。

补充说明

◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API					二进制浮点数 TANH 运算										适用機種			
138	D	TANH	P	(S)	(D)											20PM	10PM	
																	✓	✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S					*								*			-		
D													*			-		
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DTANH、DTANHP 有效 F 表示浮点数输入，使用时要有小数点 														<ul style="list-style-type: none"> 32 位指令 (6 STEP) DTANH 连续执行型 DTANHP 脉冲执行型 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明 				

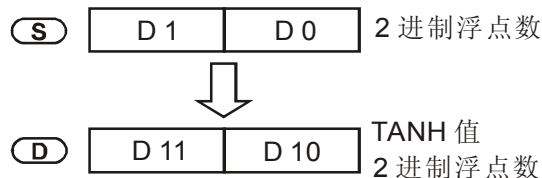
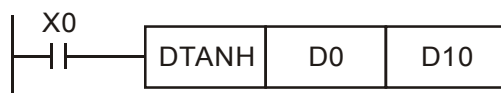
指令说明

◆ S: 指定的来源 (二进制浮点数)。 D: 取 TANH 值结果。

◆ $\tanh \text{ 值} = (e^s - e^{-s}) / (e^s + e^{-s})$

程式范例

◆ 当 X0=On 时, 指定二进制浮点数 (D1, D0) 求取 TANH 值后存于 (D11, D10) 当中, 内容为二进制浮点数。



◆ 若转换结果的绝对值大于可表示之最大浮点值, 则进位旗号=On。

◆ 若转换结果的绝对值小于可表示之最小浮点值, 则借位旗号=On。

◆ 若转换结果为 0, 则零旗号=On。

补充说明

◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

5 应用指令分类及基本使用

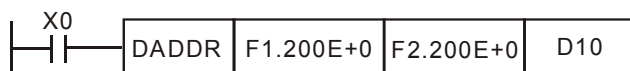
API																	适用机种	
172	D	ADDR	P	(S1)	(S2)	(D)	浮点数值加算										20PM	10PM
																	✓	✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S1													*			-		
S2													*			-		
D													*			-		
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DADDR、DADDRP 有效 																32 位指令 (13 STEP) DADDR 连续执行型 DADDRP 脉冲执行型 • 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明		

指令说明

- ◆ **S1**：浮点数值被加数。**S2**：浮点数值被加数。**D**：和。
- ◆ **S1**, **S2** 操作数可输入浮点数值。
- ◆ **DADDR** 指令可直接在 **S1**, **S2** 操作数输入浮点数值(例如: F1.2), 或以寄存器 D 存放浮点数值。
- ◆ 当 **S1**, **S2** 操作数, 以寄存器 D 存放浮点数值, 其功能与 API 120 EADD 相同。
- ◆ 当 **DADDR** 指令执行时, **D** 操作数将会存放浮点数值运算后的结果。
- ◆ **S1**, **S2** 可指定相同的寄存器编号, 此种情况下若是使用连续执行型态的指令时, 在条件接点 On 的期间, 该寄存器于每一次扫描时, 均会被加算一次, 一般的情况下都是使用脉冲执行型指令 (**DADDRP**)。
- ◆ 若运算结果的绝对值大于可表示的最大浮点数值, 则进位旗标=On。若运算结果的绝对值小于可表示的最大浮点数值, 则借位旗标=On。若运算结果为 0, 则零旗标 =On。

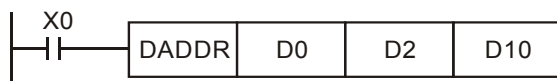
程式范例 (一)

- ◆ 当 X0=On 时, 将 F1.200E+0 浮点数值(输入浮点数 F1.2 在梯形图上显示科学记号 F1.200E+0, 浮点位数可由 WPLSoft 上检视功能来设定), 加上 F2.200E+0 浮点数值, 其运算结果为 F3.400E+0 存放至 (D10, D11) 数据寄存器内。



程式范例 (二)

- ◆ 当 X0=On 时, 将浮点数值(D1, D0)+浮点数值(D3, D2), 结果存放在 (D11, D10)中。



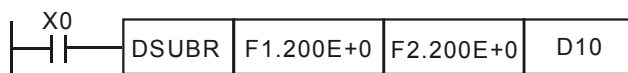
API																	适用機種		
173	D	SUBR	P	(S1)	(S2)	(D)	浮点数值加算										20PM	10PM	
																		✓	✓
	位装置				字符装置												16 位指令		
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-			
S1													*			-			
S2													*			-			
D													*			-			
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DSUBR、DSUBRP 有效 																32 位指令 (13 STEP) DSUBR 连续执行型 DSUBRP 脉冲执行型			
<ul style="list-style-type: none"> 旗标信号： <ul style="list-style-type: none"> Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明																			

指令说明

- ◆ **S1**：浮点数值被减数。**S2**：浮点数值减数。**D**：差。
- ◆ **S1**, **S2** 操作数可输入浮点数值。
- ◆ **DSUBR** 指令可直接在 **S1**, **S2** 操作数输入浮点数值(例如: F1.2), 或以寄存器 **D** 存放浮点数值。
- ◆ 当 **S1**, **S2** 操作数, 以寄存器 **D** 存放浮点数值, 其功能与 API 121 **ESUB** 相同。
- ◆ 当 **DSUBR** 指令执行时, **D** 操作数将会存放浮点数值运算后的结果。
- ◆ **S1**, **S2** 可指定相同的寄存器编号, 此种情况下若是使用连续执行型态的指令时, 在条件接点 **On** 的期间, 该寄存器于每一次扫描时, 均会被减算一次, 一般的情况下都是使用脉冲执行型指令 (**DSUBRP**)。
- ◆ 若运算结果的绝对值大于可表示的最大浮点数值, 则进位旗标=On。若运算结果的绝对值小于可表示的最大浮点数值, 则借位旗标=On。若运算结果为 0, 则零旗标 =On。

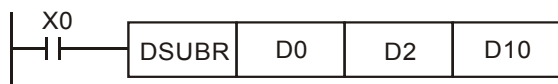
程式范例 (一)

- ◆ 当 **X0=On** 时, 将 F1.200E+0 浮点数值(输入浮点数 F1.2 在梯形图上显示科学记号 F1.200E+0, 浮点位数可由 WPLSoft 上检视功能来设定), 减去 F2.200E+0 浮点数值, 其运算结果为 F-1.000E+0 存放至 **D10**, **D11** 数据寄存器内。



程式范例 (二)

- ◆ 当 **X0=On** 时, 将浮点数值(**D1**, **D0**) - 浮点数值(**D3**, **D2**), 结果存放在 (**D11**, **D10**)中。



5 应用指令分类及基本使用

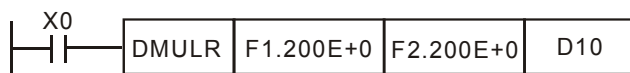
API																	适用机种		
174	D	MULR	P	(S1)	(S2)	(D)	浮点数值乘算										20PM	10PM	
																		✓	✓
	位装置				字符装置												16 位指令		
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-			
S1													*			-			
S2													*			-			
D													*			-			
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 本指令只有 32 位指令 DSUBR、DSUBRP 有效 																32 位指令 (13 STEP) DMULR 连续执行型 DMULRP 脉冲执行型 <ul style="list-style-type: none"> 旗标信号： Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 请参考下列补充说明 			

指令说明

- ◆ **S1**：浮点数值被乘数。**S2**：浮点数值乘数。**D**：积。
- ◆ **S1**, **S2** 操作数可输入浮点数值。
- ◆ **DMULR** 指令可直接在 **S1**, **S2** 操作数输入浮点数值(例如: F1.2), 或以寄存器 D 存放浮点数值。
- ◆ 当 **S1**, **S2** 操作数, 以寄存器 D 存放浮点数值, 其功能与 API 122 EMUL 相同。
- ◆ 当 **DMULR** 指令执行时, **D** 操作数将会存放浮点数值运算后的结果。
- ◆ **S1**, **S2** 可指定相同的寄存器编号, 此种情况下若是使用连续执行型态的指令时, 在条件接点 On 的期间, 该寄存器于每一次扫描时, 均会被乘算一次, 一般的情况下都是使用脉冲执行型指令 (**DMULRP**)。
- ◆ 若运算结果的绝对值大于可表示的最大浮点数值, 则进位旗标=On。若运算结果的绝对值小于可表示的最大浮点数值, 则借位旗标=On。若运算结果为 0, 则零旗标 =On。

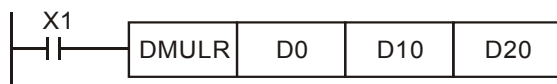
程式范例 (一)

- ◆ 当 X0=On 时, 将 F1.200E+0 浮点数值(输入浮点数 F1.2 在梯形图上显示科学记号 F1.200E+0, 浮点位数可由 WPLSoft 上检视功能来设定), 乘上 F2.200E+0 浮点数值, 其运算结果为 F2.640E+0 存放至 (D10, D11) 数据寄存器内。



程式范例 (二)

- ◆ 当 X1=On 时, 将浮点数值 (D1, D0)乘以浮点数值(D11, D10)将积存放至 (D21, D20) 数据寄存器内。



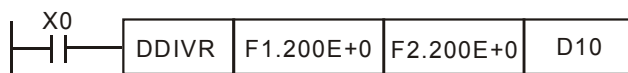
API																	适用機種		
175	D	DIVR	P	(S1)	(S2)	(D)	浮点数值除算										20PM	10PM	
																		✓	✓
	位装置				字符装置												16 位指令		
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	— — — —			
S1													*			32 位指令 (13 STEP)			
S2													*			DDIVR 连续执行型 DDIVP 脉冲执行型			
D													*			• 旗标信号:			
<ul style="list-style-type: none"> 操作数使用注意: 各装置使用范围请参考功能规格表 本指令只有 32 位指令 DSUBR、DSUBRP 有效 																<ul style="list-style-type: none"> Ox O100 M1808 M1968 零旗标 Zero flag M1809 M1969 借位旗标 Borrow flag M1810 M1970 进位旗标 Carry flag 			
请参考下列补充说明																			

指令说明

- ◆ **S1**: 浮点数值被除数。**S2**: 浮点数值除乘数。**D**: 商。
- ◆ **S1**, **S2** 操作数可输入浮点数值。
- ◆ **DDIVR** 指令可直接在 **S1**, **S2** 操作数输入浮点数值(例如: F1.2), 或以寄存器 **D** 存放浮点数值。
- ◆ 当 **S1**, **S2** 操作数, 以寄存器 **D** 存放浮点数值, 其功能与 API 123 **EDIVL** 相同。
- ◆ 当 **DDIVR** 指令执行时, **D** 操作数将会存放浮点数值运算后的结果。
- ◆ **S1**, **S2** 可指定相同的寄存器编号, 此种情况下若是使用连续执行型态的指令时, 在条件接点 **On** 的期间, 该寄存器于每一次扫描时, 均会被除算一次, 一般的情况下都是使用脉冲执行型指令 (**DDIVRP**)。
- ◆ 若运算结果的绝对值大于可表示的最大浮点数值, 则进位旗标=On。若运算结果的绝对值小于可表示的最大浮点数值, 则借位旗标=On。若运算结果为 0, 则零旗标 =On。

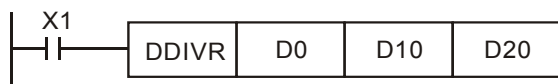
程式范例 (一)

- ◆ 当 **X0=On** 时, 将 **F1.200E+0** 浮点数值(输入浮点数 **F1.2** 在梯形图上显示科学记号 **F1.200E+0**, 浮点位数可由 **WPLSoft** 上检视功能来设定), 除以 **F2.200E+0** 浮点数值, 其运算结果为 **F0.545E+0** 存放至 (**D10**, **D11**) 数据寄存器内。



程式范例 (二)

- ◆ 当 **X1=On** 时, 将浮点数值 (**D1**, **D0**) 除以浮点数值 (**D11**, **D10**) 将商存放至 (**D21**, **D20**)数据寄存器内。



5 应用指令分类及基本使用

API																适用機種					
215~217	D	LD#					(S ₁)	(S ₂)	接点型态逻辑运算 LD#								20PM	10PM			
																✓	✓				
	位装置				字符装置												16 位指令 (5 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	LD#	连续执行型	-	-		
S ₁					*	*	*	*	*	*	*	*	*	*	*						
S ₂					*	*	*	*	*	*	*	*	*	*	*						
• 操作数使用注意: #: &、 、^ 各装置使用范围请参考功能规格表												32 位指令 (7 STEP)		DLD#		连续执行型		-		-	
												• 旗标信号: 无									

指令说明

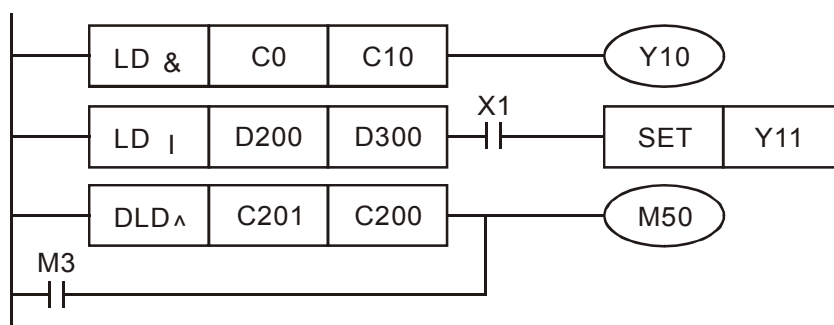
- ◆ **S₁**: 数据来源装置 1。 **S₂**: 数据来源装置 2。
- ◆ **S₁** 与 **S₂** 之内容作比较的指令, 比较结果不为 0 时, 该指令导通, 比较结果为 0 时, 该指令不导通。
- ◆ LD# 的指令可直接与母线连接使用

API No.	16-bit 指令	32-bit 指令	导通条件		非导通条件	
215	LD&	DLD&	S ₁	& S ₂ ≠ 0	S ₁	& S ₂ = 0
216	LD	DLD	S ₁	S ₂ ≠ 0	S ₁	S ₂ = 0
217	LD^	DLD^	S ₁	^ S ₂ ≠ 0	S ₁	^ S ₂ = 0

- ◆ &: 逻辑的'及' (AND) 运算。
- ◆ |: 逻辑的'或' (OR) 运算。
- ◆ ^: 逻辑的'互斥或' (XOR) 运算。
- ◆ 32 位计数器 (C200~C255) 以本指令作运算时, 一定要使用 32 位指令 (DLD#), 若是使用 16 位指令 (LD#) 时, DVP-PM 判定为"程序错误", 主机面板上 ERROR 指示灯闪烁。

程式范例

- ◆ C0 与 C10 的内容做逻辑的'及' (AND) 运算不等于 0 时, Y10=On。
- ◆ D200 与 D300 的内容做逻辑的'或' (OR) 运算不等于 0 时, 而且 X1=On 的时候, Y11=On 并保持住。
- ◆ C201 与 C200 的内容做逻辑的'互斥或' (XOR) 运算不等于 0 时或是 M3=On 的时候, M50=On。



API																适用機種			
218~220	D	AND #			(S ₁)	(S ₂)	接点型态逻辑运算 AND #									20PM	10PM		
																✓	✓		
		位装置				字符装置										16 位指令 (5 STEP)			
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	AND # 连续执行型	-	-
S ₁						*	*	*	*	*	*	*	*	*	*	*			
S ₂						*	*	*	*	*	*	*	*	*	*	*			
• 操作数使用注意: #: &、 、^ 各装置使用范围请参考功能规格表																	• 旗标信号: 无		

指令说明

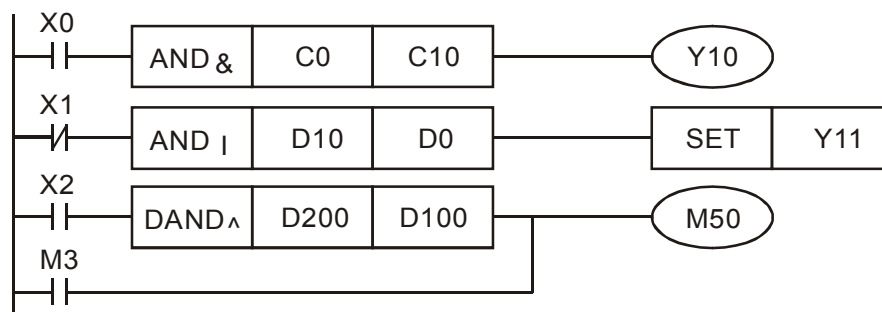
- ◆ S₁: 数据来源装置 1。 S₂: 数据来源装置 2。
- ◆ S₁ 与 S₂ 之内容作比较的指令, 比较结果不为 0 时, 该指令导通, 比较结果为 0 时, 该指令不导通。
- ◆ AND # 的指令是与接点串接的运算指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
218	AND&	DAND&	S ₁ & S ₂ ≠ 0	S ₁ & S ₂ = 0
219	AND	DAND	S ₁ S ₂ ≠ 0	S ₁ S ₂ = 0
220	AND^	DAND^	S ₁ ^ S ₂ ≠ 0	S ₁ ^ S ₂ = 0

- ◆ &: 逻辑的'及' (AND) 运算。
- ◆ |: 逻辑的'或' (OR) 运算。
- ◆ ^: 逻辑的'互斥或' (XOR) 运算。
- ◆ 32 位计数器 (C200~C255) 以本指令作运算时, 一定要使用 32 位指令 (DAND #), 若是使用 16 位指令 (AND #) 时, DVP-PM 判定为"程序错误", 主机面板上 ERROR 指示灯闪烁。

程式范例

- ◆ 当 X0=On 时且 C0 与 C10 的内容做逻辑的'及' (AND) 运算不等于 0 时, Y10=On。
- ◆ 当 X1=Off 时且 D10 与 D0 的内容做逻辑的'或' (OR) 运算不等于 0 时, Y11=On 并保持住。
- ◆ 当 X2=On 时且 32 位寄存器 D200 (D201) 与 32 位寄存器 D100 (D101) 的内容做逻辑的'互斥或' (XOR) 运算不等于 0 时或是 M3=On 的时候, M50=On。



5 应用指令分类及基本使用

API																	适用機種			
221~223	D	OR #			(S1)	(S2)	接点型态逻辑运算 OR #										20PM	10PM		
																		✓	✓	
	位装置				字符装置												16 位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	OR# 连续执行型 - -				
S1					*	*	*	*	*	*	*	*	*	*	*	32 位指令 (7 STEP)				
S2					*	*	*	*	*	*	*	*	*	*	*	DOR# 连续执行型 - -				
• 操作数使用注意: #: &、 、^ 各装置使用范围请参考功能规格表																• 旗标信号: 无				

指令说明

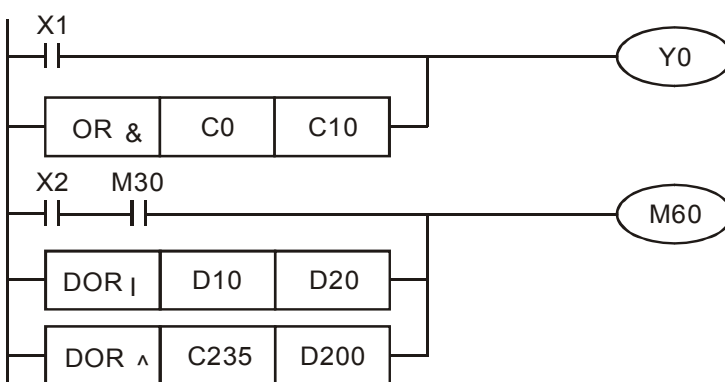
- ◆ **S1**: 数据来源装置 1。 **S2**: 数据来源装置 2。
- ◆ **S1** 与 **S2** 之内容作比较的指令, 比较结果不为 0 时, 该指令导通, 比较结果为 0 时, 该指令不导通。
- ◆ OR# 的指令是与接点并接的运算指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
221	OR&	DOR&	S1 & S2 ≠ 0	S1 & S2 = 0
222	OR	DOR	S1 S2 ≠ 0	S1 S2 = 0
223	OR^	DOR^	S1 ^ S2 ≠ 0	S1 ^ S2 = 0

- ◆ &: 逻辑的'及'(AND) 运算。
- ◆ |: 逻辑的'或'(OR) 运算。
- ◆ ^: 逻辑的'互斥或'(XOR) 运算。
- ◆ 32 位计数器 (C200~C255) 以本指令作运算时, 一定要使用 32 位指令 (DOR#), 若是使用 16 位指令 (OR#) 时, DVP-PM 判定为"程序错误", 主机面板上 ERROR 指示灯闪烁。

程式范例

- ◆ 当 X1=On 时或 C0 与 C10 的内容做逻辑的'及'(AND) 运算不等于 0 时, Y0=On。
- ◆ 当 X2 及 M30 都等于 On 的时候, 或者是 32-bit 寄存器 D10 (D11) 与 32 位寄存器 D20 (D21) 的内容做逻辑的'或'(OR) 运算不等于 0 时, 或者是 32 位计数器 C235 与 32 位寄存器 D200 (D201) 的内容做逻辑的'互斥或'(XOR) 运算不等于 0 时, M60=On。



API																	适用機種			
224~230	D	LD※					(S1)	(S2)	接点型态比较 LD※									20PM	10PM	
																		✓	✓	
	位装置				字符装置												16 位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	LD※	连续执行型	-	-	
S ₁					*	*	*	*	*	*	*	*	*	*	*		32 位指令 (7 STEP)			
S ₂					*	*	*	*	*	*	*	*	*	*	*		DLD※	连续执行型	-	-
• 操作数使用注意: ※: =、>、<、<>、≤、≥ 各装置使用范围请参考功能规格表																• 旗标信号: 无				

指令说明

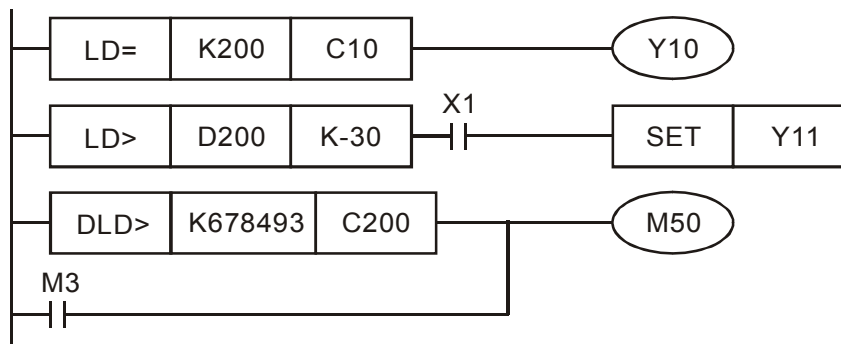
- ◆ S₁: 数据来源装置 1。 S₂: 数据来源装置 2。
- ◆ S₁ 与 S₂ 之内容作比较的指令, 以 API 224 LD= 为例, 比较结果为”等于”时, 该指令导通, ”不等于”时, 该指令不导通。
- ◆ LD※的指令可直接与母线连接使用

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
224	LD=	DLD=	S ₁ = S ₂	S ₁ ≠ S ₂
225	LD>	DLD>	S ₁ > S ₂	S ₁ ≤ S ₂
226	LD<	DLD<	S ₁ < S ₂	S ₁ ≥ S ₂
228	LD<>	DLD<>	S ₁ ≠ S ₂	S ₁ = S ₂
229	LD≤	DLD≤	S ₁ ≤ S ₂	S ₁ > S ₂
230	LD≥	DLD≥	S ₁ ≥ S ₂	S ₁ < S ₂

- ◆ 32 位计数器 (C200~C255) 以本指令作比较时, 一定要使用 32 位指令 (DLD※), 若是使用 16 位指令 (LD※) 时, DVP-PM 判定为”程序错误”, 主机面板上 ERROR 指示灯闪烁。

程式范例

- ◆ C10 的内容等于 K200 时, Y10=On。
- ◆ 当 D200 的内容大于 K-30, 而且 X1=On 的时候, Y11=On 并保持住。
- ◆ C200 的内容小于 K678,493 或者是 M3=On 的时候, M50=On。



5 应用指令分类及基本使用

API																		适用機種		
232~238	D	AND※					(S1)	(S2)	接点型态比较 AND※										20PM	10PM
																		✓	✓	
	位装置				字符装置												16 位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	AND※	连续执行型	-	-	
S1					*	*	*	*	*	*	*	*	*	*	*					
S2					*	*	*	*	*	*	*	*	*	*	*					
• 操作数使用注意: ※: =、>、<、<>、≤、≥ 各装置使用范围请参考功能规格表																32 位指令 (7 STEP)				
																DAND※	连续执行型	-	-	
																• 旗标信号: 无				

指令说明

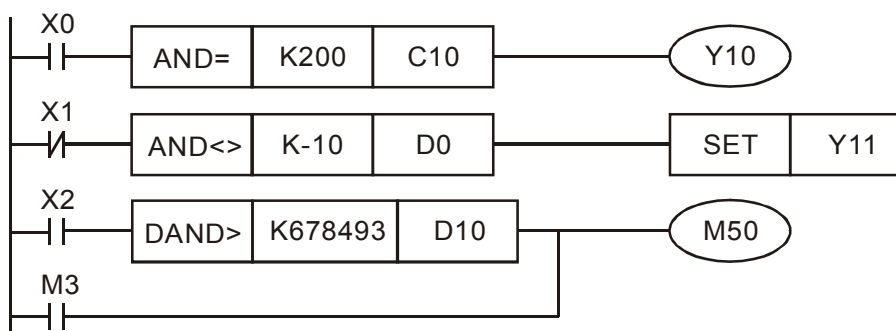
- ◆ **S₁**: 数据来源装置 1。 **S₂**: 数据来源装置 2。
- ◆ **S₁** 与 **S₂** 之内容作比较的指令, 以 API 232 AND= 为例, 比较结果为等于时, 该指令导通, 不等于时, 该指令不导通。
- ◆ AND※的指令是与接点串接的比较指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
232	AND=	DAND=	S₁ = S₂	S₁ ≠ S₂
233	AND>	DAND>	S₁ > S₂	S₁ ≤ S₂
234	AND<	DAND<	S₁ < S₂	S₁ ≥ S₂
236	AND<>	DAND<>	S₁ ≠ S₂	S₁ = S₂
237	AND≤	DAND≤	S₁ ≤ S₂	S₁ > S₂
238	AND≥	DAND≥	S₁ ≥ S₂	S₁ < S₂

- ◆ 32 位计数器 (C200~C255) 以本指令作比较时, 一定要使用 32 位指令 (DAND※), 若是使用 16 位指令 (AND※) 时, DVP-PM 判定为“程序错误”, 主机面板上 ERROR 指示灯闪烁。

程式范例

- ◆ 当 X0=On 时且 C10 的现在值又等于 K200 时, Y10=On。
- ◆ 当 X1=Off 而寄存器 D0 的内容又不等于 K-10 的时候, Y11=On 并保持住。
- ◆ 当 X2=On 而且 32 位寄存器 D0 (D11) 的内容又小于 678,493 的时候或 M3=On 时, M50=On。



API																适用機種			
240~246	D	OR※				(S1)	(S2)	接点型态比较 OR※								20PM	10PM		
																✓	✓		
	位装置				字符装置										16 位指令 (5 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	OR※	连续执行型	-	-
S ₁					*	*	*	*	*	*	*	*	*	*	*				
S ₂					*	*	*	*	*	*	*	*	*	*	*				
• 操作数使用注意: ※: =、>、<、<>、≤、≥ 各装置使用范围请参考功能规格表																• 旗标信号: 无			

指令说明

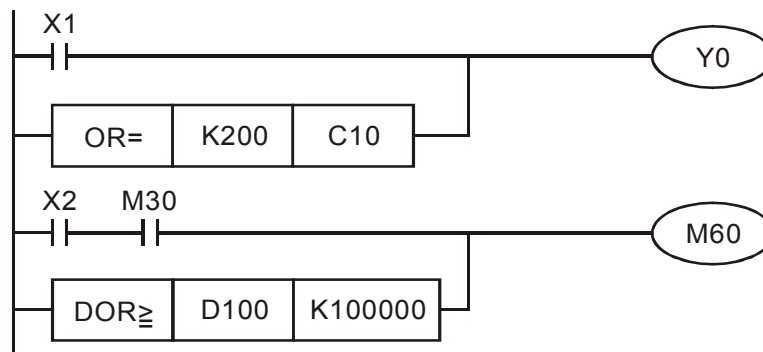
- ◆ S₁: 数据来源装置 1。 S₂: 数据来源装置 2。
- ◆ S₁ 与 S₂ 之内容作比较的指令, 以 API 240 OR= 为例, 比较结果为等于时, 该指令导通, 不等于时, 该指令不导通。
- ◆ OR※的指令是与接点并接的比较指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
240	OR=	DOR=	S ₁ = S ₂	S ₁ ≠ S ₂
241	OR>	DOR>	S ₁ > S ₂	S ₁ ≤ S ₂
242	OR<	DOR<	S ₁ < S ₂	S ₁ ≥ S ₂
244	OR<>	DOR<>	S ₁ ≠ S ₂	S ₁ = S ₂
245	OR≤	DOR≤	S ₁ ≤ S ₂	S ₁ > S ₂
246	OR≥	DOR≥	S ₁ ≥ S ₂	S ₁ < S ₂

- ◆ 32 位计数器 (C200~C255) 以本指令作比较时, 一定要使用 32 位指令 (DOR※), 若是使用 16 位指令 (OR※) 时, DVP-PM 判定为“程序错误”, 主机面板上 ERROR 指示灯闪烁。

程式范例

- ◆ 当 X1=On 时, 或者是 C10 的现在值等于 K200 时, Y0=On。
- ◆ 当 X2 及 M30 都等于 On 的时候, 或者是 32 位寄存器 D100 (D101) 的内容大于或等于 K100,000 时, M60=On。



5 应用指令分类及基本使用

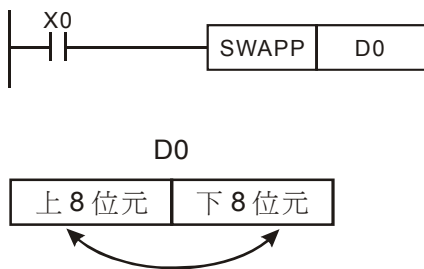
API																	适用機種		
147	D	SWAP	P		(S)	上下 8 位互换											20PM	10PM	
																	✓	✓	
S	位装置				字符装置											16 位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SWAP	连续执行型	SWAPP	脉冲执行型
							*	*	*	*	*	*	*	*	*	DSWAP	连续执行型	DSWAPP	脉冲执行型
<p>• 操作数使用注意：此指令有支持 V、Z 装置（当 16 位指令时不能使用装置；当 32 位指令时不能使用 V 装置）</p> <p>各装置使用范围请参考功能规格表</p> <p>指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)</p>																32 位指令 (7 STEP)			
																• 旗标信号：无			

指令说明

- ◆ **S**：欲执行上下位 8 位互相交换之装置。
- ◆ 16 位指令时，上位 8 位与下位 8 位的内容互相交换。
- ◆ 32 位指令时，两个寄存器的上位 8 位与下位 8 位的内容互相交换。
- ◆ 本指令一般使用脉冲执行型指令 (SWAPP, DSWAPP)。

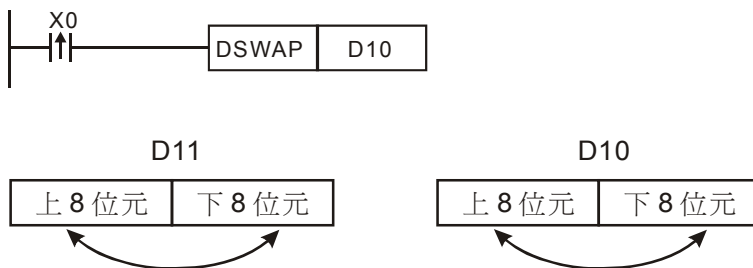
程式范例 (一)

- ◆ 当 X0=On 时，将 D0 的上 8 位与下位 8 位的内容互相交换。



程式范例 (二)

- ◆ 当 X0=On 时，将 D11 的上位 8 位与下位 8 位的内容互相交换，D10 的上位 8 位与下位 8 位的内容互相交换。



API																		适用機種			
154	D	RAND	P				(S1)	(S2)	(D)									随机数值	20PM	10PM	
																			✓	✓	
	位装置				字符装置												16 位指令 (5 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z			RAND	连续执行型	RANDP	脉冲执行型
S ₁					*	*	*	*	*	*	*	*	*	*	*	*					
S ₂					*	*	*	*	*	*	*	*	*	*	*	*					
D								*	*	*	*	*	*	*	*	*					
<ul style="list-style-type: none"> 操作数使用注意: 此指令有支持 V、Z 装置 (当 16 位指令时不能使用装置; 当 32 位指令时不能使用 V 装置) 各装置使用范围请参考功能规格表 指定 KnX(Y/M/S)等装置的位装置时, 限以 8 进制或 10 进制中 16 的倍数为起始编号。如: K1X0, K4Y20 (八进制); K1M0, K4S16 (十进制) 																	32 位指令 (7 STEP)				
																	DRAND	连续执行型	DRANDP	脉冲执行型	
																	• 旗标信号: 无				

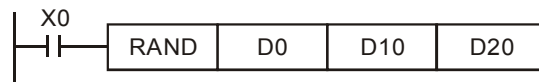
指令说明

- ◆ **S₁**: 随机数产生的范围下限。 **S₂**: 随机数产生的范围上限。 **D**: 随机数产生的结果。
- ◆ 16 位操作数 **S₁**, **S₂** 使用范围: $K0 \leq S_1, S_2 \leq K32,767$, 32 位操作数 **S₁**, **S₂** 使用范围: $K0 \leq S_1, S_2 \leq K2,147,483,647$ 。

- ◆ 操作数 **S₁** \leq 操作数 **S₂**, 若使用者输入 **S₁** > **S₂**, PM 判断为运算错误。

程式范例

- ◆ 当 X0=On, RAND 指令产生介于范围下限 D0 与范围上限 D10 的随机数, 将结果存放于 D20 内。

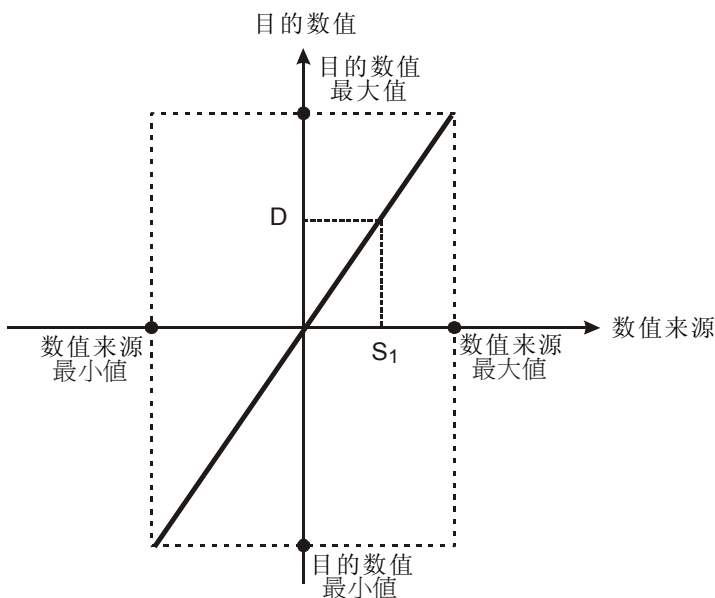


5 应用指令分类及基本使用

API					比例运算										适用機種				
202		SCAL	P	(S ₁)	(S ₂)	(S ₃)	(D)											20PM	10PM
					*	*									*			✓	✓
					*	*									*				
					*	*									*				
															*				
16 位指令 (9 STEP) SCAL 连续执行型 SCALP 脉冲执行型 32 位指令 (7 STEP) — — — — — • 旗标信号: 无																			
• 操作数使用注意: 各装置使用范围请参考功能规格表																			

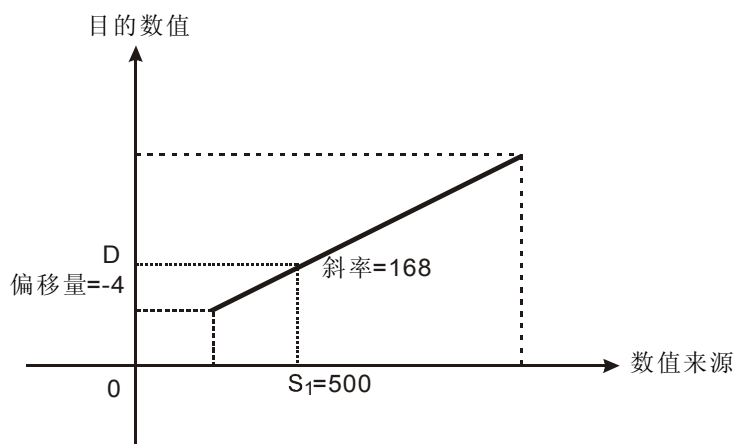
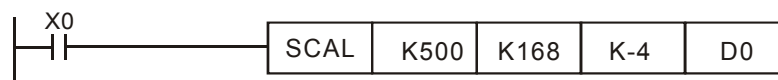
指令说明

- ◆ S₁: 来源数值资料。S₂: 斜率, 单位为 0.001。S₃: 偏移量。D: 目的地装置。
- ◆ 操作数范围 S₁, S₂, S₃ 是 -32767~32767
- ◆ 指令内部运算公式为: $D = (S_1 \times S_2) \div 1000 + S_3$
- ◆ S₂ 和 S₃ 的数值须由使用者依下列斜率与偏移量公式先行运行, 然后将小数点 4 舍 5 入后, 再取 16 位的整数输入。
- ◆ 斜率公式为: $S_2 = [(目的数值最大值 - 目的) \div (来源数值最大值 - 来源数值最小值)] \times 1000$
- ◆ 偏移量公式为: $S_3 = 目的数值最小值 - 来源数值最小值 \times S_2 \div 1000$
- ◆ 输出曲线如下图所示:



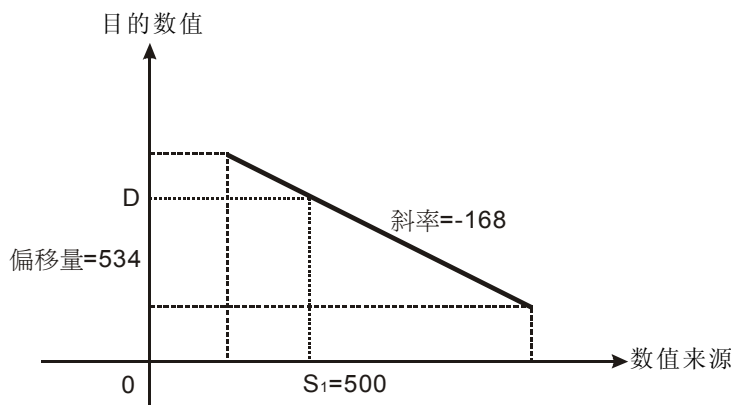
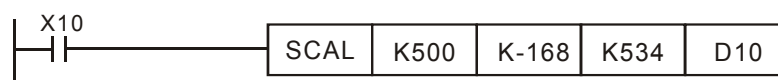
程式范例 (一)

- ◆ 已知 S₁ 数值资料来源为 500, S₂ 斜率为 168, S₃ 偏移量为 -4., 当 X0=On 时, SCAL 指令执行, 可在 D0 得到所要求的比例值。
- ◆ 运算方式: $D_0 = (500 \times 168) \div 1000 + (-4) = 80$



程式范例
(二)

- ◆ 已知 S_1 数值资料来源为 500, S_2 斜率为-168, S_3 偏移量 534。当 X10=On 时, SCAL 指令执行, 可在 D10 得到所要求的比例值。
- ◆ 运算方式: $D10 = (500 \times -168) \div 1000 + 534 = 450$



补充说明

- ◆ 此 SCAL 指令示用于已知斜率与偏移量, 若不知斜率与偏移量建议使用 SCLP 指令来做运算。
- ◆ 输入参数 S_2 时, 其输入数值必须为-32,768 ~ 32,767 之间的数值 (实际数值为-32,768 ~ 32,767)。若是 S_2 实际数值超过范围时, 请改用 SCLP 指令运算。
- ◆ 使用者运用斜率换算公式时, 须注意来源数值最大值, 必须大于来源数值最小值, 而目的数值最大值, 并不限制大于目的数值最小值。
- ◆ 若 $D > 32,767$, 则 $D = 32,767$ 。若 $D < -32,768$, 则 $D = -32,768$ 。

5 应用指令分类及基本使用

API																		适用机种	
203	D	SCLP	P	(S1)	(S2)	(D)	参数型比例运算										20PM	10PM	
																		✓	✓
	位装置				字符装置												16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SCLP 连续执行型 SCLPP 脉冲执行型			
S ₁					*	*							*			32 位指令 (13 STEP)			
S ₂					*	*							*			DSCLP 连续执行型 DSCLPP 脉冲执行型			
S ₃					*	*							*			• 旗标信号: 旗标信号: M1162 十进制整数与二进制浮点数使用旗标, On 时表示二进制浮点数			
D													*						
• 操作数使用注意: 各装置使用范围请参考功能规格表																			

指令说明

- ◆ S₁: 数值资料来源。S₂: 参数, 单位为 0.001。D: 目的地装置。
- ◆ 16 位指令 S₂ 参数设定内容如下:

装置编号	参数名称与说明	设定范围
S ₂	来源数值最大值	-32768~32767
S ₂ +1	来源数值最小值	-32768~32767
S ₂ +2	目的数值最大值	-32768~32767
S ₂ +3	目的数值最小值	-32768~32767

- ◆ 16 位指令 S₂ 操作数将连续占用 4 个装置。
- ◆ 32 位指令 S₂ 参数设定内容如下:

装置编号	参数名称与说明	设定范围	
		整数	浮点数
S ₂ 、S ₂ +1	来源数值最大值	-2,147,483,648~2,147,483,647	32 位浮点数范围
S ₂ +2、3	来源数值最小值		
S ₂ +4、5	目的数值最大值		
S ₂ +6、7	目的数值最小值		

- ◆ 32 位指令 S₂ 操作数将连续占用 8 个装置
- ◆ 指令内部运算公式为 $D = [(S_1 - \text{来源数值最小值}) \times (\text{目的数值最大值} - \text{目的数值最小值})] \div (\text{来源数值最大值} - \text{来源数值最小值}) + \text{目的数值最小值}$
- ◆ 来源数值和目的数值运算关系:

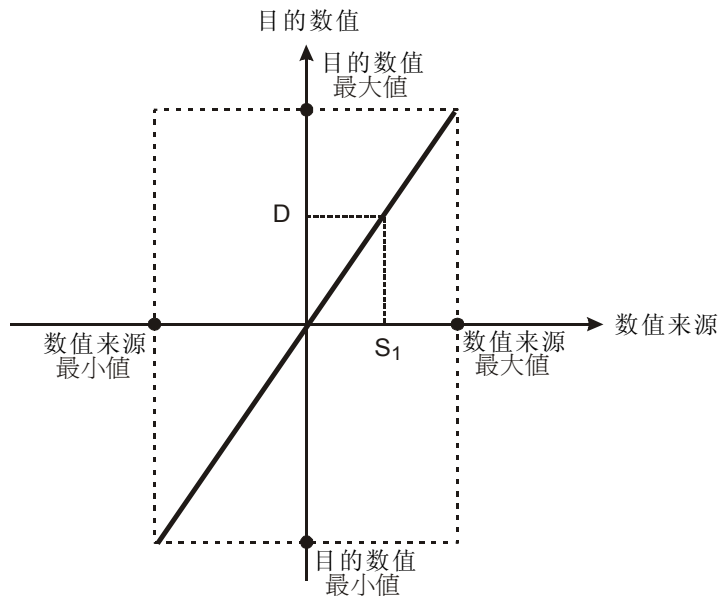
$$y=kx+b, y=\text{目的数值 (D)}, k=\text{斜率}=(\text{目的数值最大值} - \text{目的数值最小值})\div(\text{来源数值最大值} - \text{来源数值最小值}), x=\text{来源数值(S}_1), b=\text{偏移量}=\text{目的数值最小值} - \text{来源数值最小值} \times \text{斜率}$$

- ◆ 将上面的各参数带入公式 $y=kx+b$, 即可推导得出指令内部运算公式:

$$y=kx+b = D = k S_1 + b = \text{斜率} \times S_1 + \text{偏移量} = \text{斜率} \times S_1 + \text{目的数值最小值} - \text{来源数值最小值} \times \text{斜率} = \text{斜率} \times (S_1 - \text{来源数值最小值}) + \text{目的数值最小值} = (S_1 - \text{来源数值最小值}) \times (\text{目的数值最大值} - \text{目的数值最小值}) \div (\text{来源数值最大值} - \text{来源数值最小值}) + \text{目的数值最小值}$$

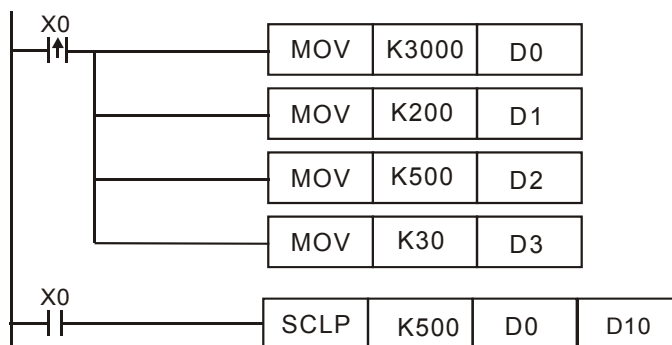
最小值) + 目的数值最小值

- ◆ 假如 $S_1 >$ 来源数值最大值, $S_1 =$ 来源数值最大值。假如 $S_1 <$ 来源数值最小值, $S_1 =$ 来源数值最小值。当输入数值与参数设定完成后, 则其输出曲线将如下图所示:

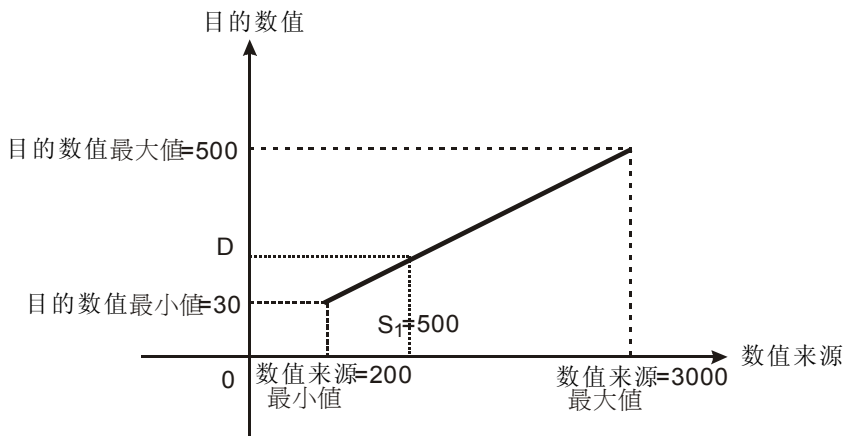


程式范例
(一)

- ◆ 已知 S_1 数值资料来源为 500, 来源数值最大值 $D0=3000$, 来源数值最小值 $D1=200$, 目的数值最大值 $D2=500$, 目的数值最小值 $D3=30$ 。当 $X0=On$ 时, SCLP 指令执行, 可在 D10 得到所要求的比例值。
- ◆ 运算方式: $D10 = [(500 - 200) \times (500 - 30)] \div (3000 - 200) + 30 = 80.35$ 。取整数, $D10 = 80$

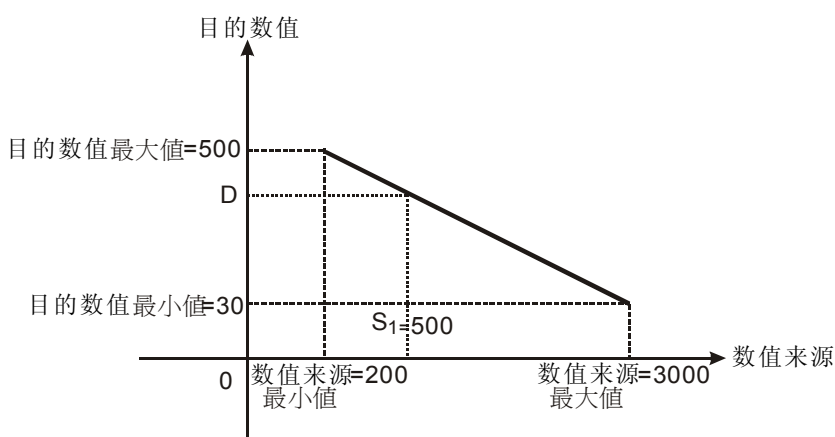
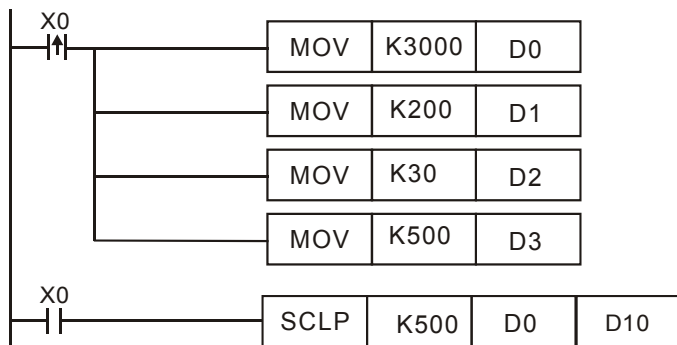


5 应用指令分类及基本使用

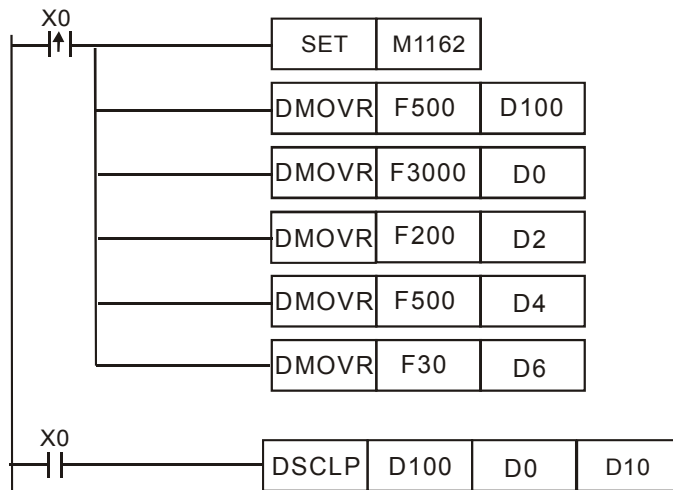


程式范例 (二)

- ◆ 已知 S_1 数值资料来源为 500，来源数值最大值 $D0=3000$ ，来源数值最小值 $D1=200$ ，目的数值最大值 $D2=30$ ，目的数值最小值 $D3=500$ 。当 $X0=On$ ，SCLP 指令执行，可在 D10 得到所要求的比例值。
- ◆ 运算方式: $D10 = [(500-200) \times (30-500)] \div (3000-200) + 500 = 449.64$ 。四舍五入取整数, $D10 = 450$



- ◆ 已知 S_1 数值资料来源 $D100=F500$ ，来源数值最大值 $D0=F3000$ ，来源数值最小值 $D2=F200$ ，目的数值最大值 $D4=F500$ ，目的数值最小值 $D6=F30$ 。当 $X0=On$ 时，SET M1162，使用浮点数运算且 DSCLP 指令执行。可在 D10 得到所要求比例值。
- ◆ 运算方式: $D10 = [(F500 - F200) \times (F500 - F30)] \div (F3000 - F200) + F30 = F80.35$ 。取整数, $D10 = F80$ 。



补充说明

- ◆ 16 位 S_1 操作数数值设定范围: 来源数值最大值 $\geq S_1 \geq$ 来源数值最小值, -32768~32767。如果超出边界值以边界值运算。
- ◆ 32 位 S_1 整数操作数数值设定范围: 来源数值最大值 $\geq S_1 \geq$ 来源数值最小值, -2,147,483,648~2,147,483,647。如果超出边界值以边界值运算。
- ◆ 32 位 S_1 浮点数操作数数值设定范围: 来源数值最大值 $\geq S_1 \geq$ 来源数值最小值, 依 32 位浮点数范围。如果超出边界值以边界值运算。
- ◆ 使用者运用时, 须注意来源数值最大值, 必须大于来源数值最小值, 而目的数值最大值, 并不限制大于目的数值最小值。

5 应用指令分类及基本使用

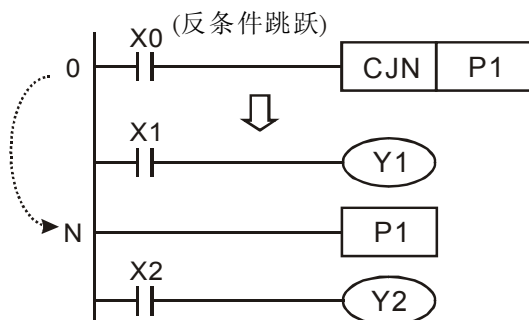
API																	适用機種			
256		CJN		P		(S)												20PM	10PM	
																		✓	✓	
		位裝置				字裝置											16 位指令 (3 STEP)			
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	CJN 連續執行型		CJNP 脈衝執行型	
		<ul style="list-style-type: none"> 操作數使用注意：S 操作數可指定 P S 操作數可指定 P0~P255 P 裝置不支持 V、Z 修飾 															32 位指令			
																	旗標信號：無			

指令說明

- ◆ **S**：反條件跳躍之目的指標。
- ◆ 當 **CJN** 指令前的條件接點為導通狀態時，則至下一行程序動作；反之，當條件接點為不導通狀態時，則跳躍至所指定的 **P** 指標處動作。
- ◆ 當使用者希望 **O100** 程序中的某一部份不需要執行時，以縮短掃描時間，以及使用於雙重輸出時，可使用 **CJN** 或 **CJNP** 指令。
- ◆ 指針 **P** 所指之程序若在 **CJN** 指令之前，需注意會發生 **WDT** 逾時之錯誤，**O100** 停止運轉，請注意使用。
- ◆ **CJN** 指令可重複指定同一指標 **P**，但 **CJN** 與 **CALL** 不可指定同一指標 **P**，否則會產生錯誤。
- ◆ 跳躍執行中各種裝置動作情形說明：
 1. **Y**、**M**、**S** 保持跳躍發生前之狀態。
 2. 執行計時中之 **10ms** 定時器會暫停計時。
 3. 一般計數器會停止計數；而一般應用指令不會被執行。
 4. 定時器之清除指令若在跳躍前被驅動，則在跳躍執行中該裝置仍處於清除狀態。

程式範例

- ◆ 當 **X0=Off** 時，程序自動從地址 **0** 跳躍至地址 **N** (即指定之標籤 **P1**) 繼續執行，中間地址跳過不執行。
- ◆ 當 **X0=On** 時，程序如同一般程序由地址 **0** 繼續往下執行，此時 **CJN** 指令不被執行。



API																	适用機種					
257	JMP																	20PM	10PM			
																		✓	✓			
		位裝置				字符裝置											16 位指令 (3 STEP)					
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	JMP 连续执行型		-	-		
		<ul style="list-style-type: none"> 操作数使用注意: S 操作数可指定 P S 操作数可指定 P0~P255 不须接点驱动的指令 P 装置不支持 V、Z 修饰 																	32 位指令		-	-
																			旗标信号: 无			

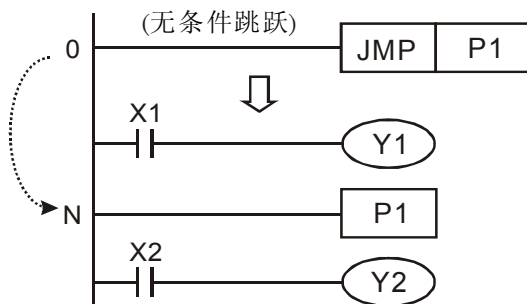
指令说明

◆ JMP 指令功能与 CJ 指令雷同, 不同之处只差在 CJ 指令前需要接点驱动, 而 JMP 指令前是不须接点驱动的指令。

◆ JMP 指令不支持 JMPP 脉冲执行型。

程式范例

◆ 当程序扫描至第 0 行程序时, 无论 JMP 指令前是否有无条件接点 (也不考虑条件接点的状态的情况), 程序皆从地址 0 跳跃至地址 N (即指定之标签 P1) 继续执行, 中间地址跳过不执行。



5 应用指令分类及基本使用

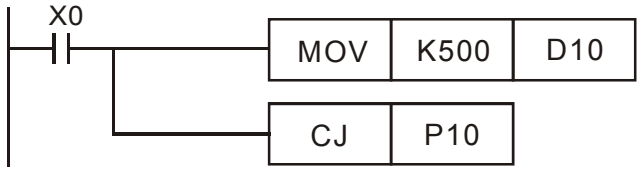
API																				适用機種		
258			BRET																	回母线	20PM	10PM
																					✓	✓
		位装置				字符装置												:16 位指令 (1 STEP)				
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	:BRET 连续执行型 - -					
		<ul style="list-style-type: none"> 操作数使用注意: 无操作数 不须接点驱动的指令 																:32 位指令				
																		- - - - -				
																		旗标信号: 无				

指令说明

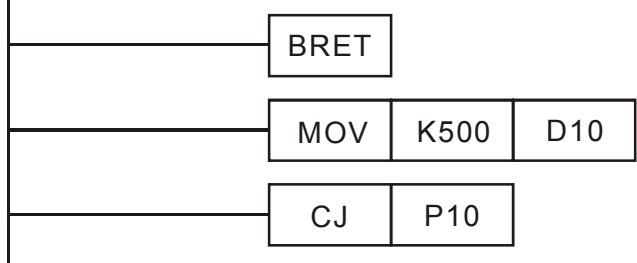
- ◆ BRET 指令无须接点驱动。
- ◆ 执行 BRET 指令之后，原本须条件接点驱动的指令相当于已连接上母线，可以直接执行这些指令。

程式范例

- ◆ 一般程序，需要 X0 为 On 时，才会执行条件接点后的指令。如下所示：



- ◆ 当加入 BRET 指令之后，原本须条件接点驱动的指令，就像已经连接上母线，而可直接执行动作。如下所示：



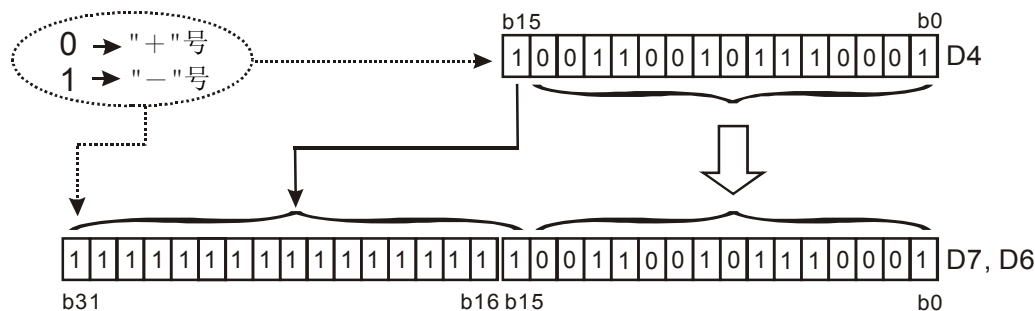
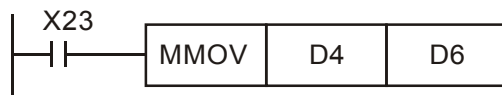
API																适用机种		
259	MMOV	P	S		D		16→32 位数值转换										20PM	10PM
			✓	✓														
		位装置				字符装置										16 位指令 (5 STEP)		
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	MMOV 连续执行型	MMOVP 脉冲执行型
S					*	*	*	*	*	*	*	*	*	*	*	*	32 位指令	
D								*	*	*	*	*	*	*	*	*	— — — — —	
<ul style="list-style-type: none"> 操作数使用注意：各装置使用范围请参考功能规格表 																	旗标信号：无	

指令说明

- ◆ **S**：数据之来源 (16 位)。**D**：数据之搬移目的地 (32 位)。
- ◆ 将 16 位装置 **S** 中的数据传送到 32 位的装置 **D** 中，其中指定的符号位被重复的复制存放在目的地。

程式范例

- ◆ 当 X23 为 0N 时，D4 的数据传送到 D6 和 D7。



在以上的例子中，D4 的 b15 位数据传送到 (D7/D6) 的 b15 到 b31 位，变成负数 (和 D4 的一样)。

5 应用指令分类及基本使用

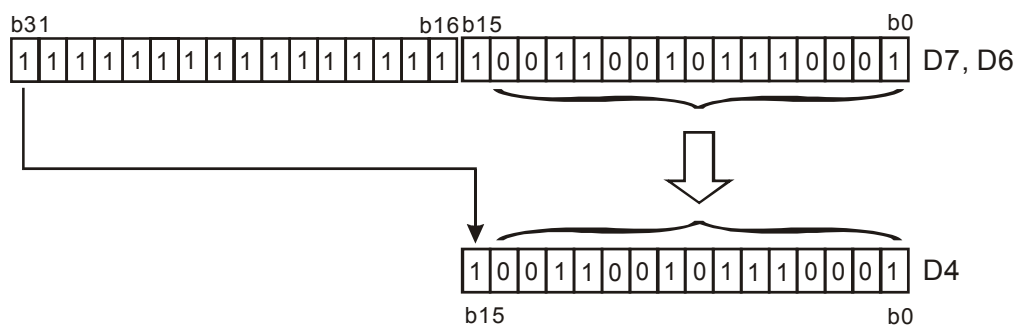
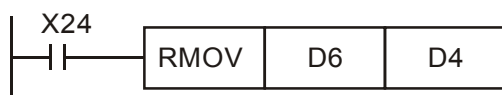
API																适用機種			
260		RMOV					(S)	(D)									20PM	10PM	
			P														✓	✓	
	位裝置				字裝置										16 位指令 (5 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	RMOV 連續執行型		RMOVP 脈衝執行型	
S					*	*	*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*	*	*				
• 操作數使用注意：各裝置使用範圍請參考功能規格表																旗標信號：無			

指令說明

- ◆ **S**：數據之來源 (32 位)。**D**：數據之搬移目的地 (16 位)。
- ◆ 將 32 位裝置 **S** 中的數據傳送到 16 位的裝置 **D** 中，其中符號位被保留。


程式範例

- ◆ 當 X24 為 0N 時，D6 和 D7 的數據傳送到 D4。



當 X24 為 0N 時，(S) 中最高位 (D7: b31) 被傳送到 (D) 中最高位 (D4: b15) 中，其它位(b0~b14)則對應傳送，而 b15~b30 被忽略未被傳送。

6.1 运动指令及 G-Code 指令一览表

 运动指令

分类	MON	指令码	功能	适用機種			响应时间	页码
				20D	20M	10M		
运动指令	00	DRV	两轴高速定位	✓	✓		20~25ms	6-7
			三轴高速定位		✓		20~25ms	6-7
	01	LIN	双轴同动直线补间移动（可考虑剩余距离）	✓	✓		20~22ms	6-10
			三轴同动直线补间移动（可考虑剩余距离）		✓		20~25ms	6-10
	02	CW	顺时针圆弧移动（设定圆心位置，可考虑剩余距离）	✓	✓		20~24ms	6-14
			顺时针螺旋移动（设定圆心位置，可考虑剩余距离）		✓		20~25ms	6-14
	03	CCW	逆时针圆弧移动（设定圆心位置，可考虑剩余距离）	✓	✓		20~24ms	6-14
			逆时针螺旋移动（设定圆心位置，可考虑剩余距离）		✓		20~25ms	6-14
	04	CW	顺时针圆弧移动（设定半径长度，可考虑剩余距离）	✓	✓		20~24ms	6-20
			顺时针螺旋移动（设定半径长度，可考虑剩余距离）		✓		20~25ms	6-20
	05	CCW	逆时针圆弧移动（设定半径长度，可考虑剩余距离）	✓	✓		20~24ms	6-20
			逆时针螺旋移动（设定半径长度，可考虑剩余距离）		✓		20~25ms	6-20
	06	TIM	停顿时间	✓	✓		-	6-25
	07	DRVZ	返回机械原点（原点复位）	✓	✓		20~25ms	6-26
	08	SETR	电气零点设定	✓	✓		-	6-29
	09	DRVR	返回电气零点	✓	✓		20~25ms	6-30
	10	INTR	双轴同动单段速补间移动（忽略剩余距离）	✓	✓		20~25ms	6-32
	11	SINTR	单段速插入	✓	✓		20~25ms	6-33
	12	DINTR	两段速插入指令	✓	✓		20~25ms	6-35
13	MOVC	设定直线位移补偿	✓	✓		-	6-37	
14	CNTC	圆弧圆心补偿	✓	✓		-	6-38	
15	RADC	圆弧半径补偿	✓	✓		-	6-39	
16	CANC	取消运动补偿	✓	✓		-	6-40	
17	ABST	绝对坐标系统设定	✓	✓		-	6-41	
18	INCT	相对坐标系统设定	✓	✓		-	6-41	
19	SETT	设定现在位置	✓	✓		-	6-42	

6 运动指令与 G-Code 分类基本使用说明

📖 O 指针 / M 码指令

指令码	功能	说明	适用機種			页 码
			20D	20M	10M	
O	程序指针	主程序 O100, 运动子程序 OX0~OX99	✓	✓		6-45
M	M 码指令	M0~M65535 M102: O100 主程序结束指令 M2: OX0~OX99 运动子程序结束指令	✓	✓		6-46

📖 G 码指令

分类	G 码	运动指令码	功能	适用機種			页码
				20D	20M	10M	
G 码指令	00	DRV	两轴高速定位	✓	✓		6-44
	00	DRV	三轴高速定位		✓		6-44
	01	LIN	双轴同动直线补间移动 (可考虑剩余距离)	✓	✓		6-47
	01	LIN	三轴同动直线补间移动 (可考虑剩余距离)		✓		6-47
	02	CW	顺时针圆弧移动 (设定圆心位置, 可考虑剩余距离)	✓	✓		6-50
	02	CW	顺时针螺旋移动 (设定圆心位置, 可考虑剩余距离)		✓		6-50
	03	CCW	逆时针圆弧移动 (设定圆心位置, 可考虑剩余距离)	✓	✓		6-50
	03	CCW	逆时针螺旋移动 (设定圆心位置, 可考虑剩余距离)		✓		6-50
	02	CW	顺时针圆弧移动 (设定半径长度, 可考虑剩余距离)	✓	✓		6-51
	02	CW	顺时针螺旋移动 (设定半径长度, 可考虑剩余距离)		✓		6-51
	03	CCW	逆时针圆弧移动 (设定半径长度, 可考虑剩余距离)	✓	✓		6-51
	03	CCW	逆时针螺旋移动 (设定半径长度, 可考虑剩余距离)		✓		6-51
	04	TIM	停顿时间	✓	✓		6-51
	17	NULL	XY 平面设定		✓		6-51
	18	NULL	XZ 平面设定		✓		6-51
	19	NULL	YZ 平面设定		✓		6-51
	90	ABST	绝对坐标系统设定	✓	✓		6-52
91	INCT	相对坐标系统设定	✓	✓		6-52	

6.2 运动指令及 G-Code 指令的组成

6.2.1 运动指令

◆ 运动指令的结构可分为两部份：指令名及操作数

指令名：		表示指令执行功能
操作数	菜单示：	参数标记(X, Y, Z, F, FX, FY, FZ, R, I, J, K)
	参数设定：	设定参数数值大小

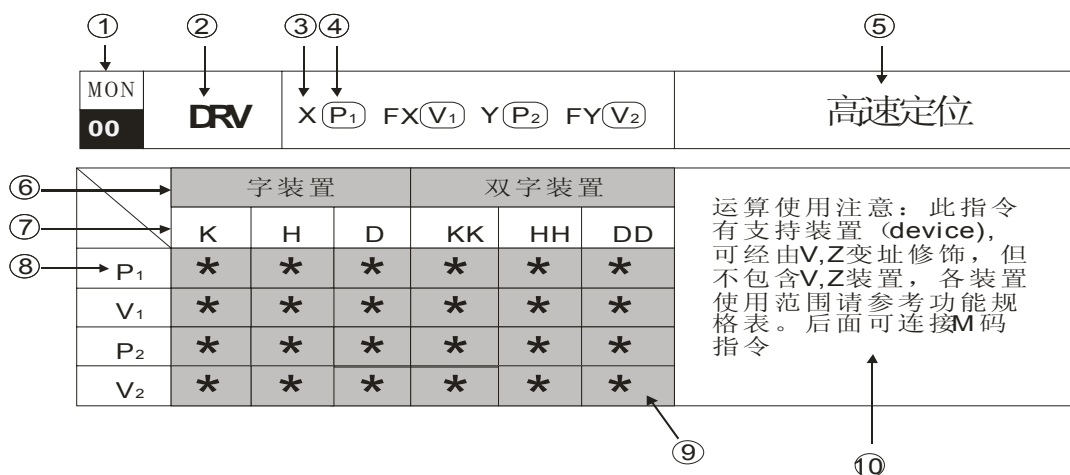
- 操作数菜单示之参数标记必须填写。
- 操作数参数设定可输入以下几种型式：
 1. 仅输入阿拉伯数字与 32 位寄存器 (DD)，操作数参数都以 32 位计算。
例如：DRV X1000 FX1000 Y1000 FYDD1000。
 2. 输入 K、H、D 与阿拉伯数字，操作数参数以 16 位计算。
例如：DRV XK1000 FXH1000 YK1000 FYD1000。
 3. 运动指令内之操作数参数输入阿拉伯数字，可同时支持 16 位及 32 位交叉使用。
例如：DRV X1000 FXHH1000 YK1000 FY1000。

◆ 计算运动指令所占程序空间

- 运动指令名称在 20D 机种都占程序空间 1 个地址 (Step)，而 20M 机种，只有 DRV、LIN、CW、CCW 和 SETT 等指令名占程序空间 2 个地址，其余也都是 1 个地址。
- 操作数参数，仅输入阿拉伯数字之操作数，1 个操作数占 3 个地址 (Step)。
- 操作数参数，输入 K、H、D、DD 与阿拉伯数字之操作数，1 个操作数占 2 个地址 (Step)。
- 操作数参数，输入 KK、HH 与阿拉伯数字之操作数，1 个操作数占 3 个地址 (Step)。

◆ 运动指令的格式说明

6 运动指令与 G-Code 分类基本使用说明



- ① 运动指令 MON (Motion Number) 编号号码
- ② 运动指令名称
- ③ 运动指令操作数参数标记
- ④ 运动指令操作数参数数值大小
- ⑤ 运动指令功能描述
- ⑥ 装置型式
- ⑦ 装置名称
- ⑧ 有符号 '*' 标示者, 表示该操作数可使用的装置
- ⑨ 符号 '*' 标示且含灰底色者, 表示该装置可使用间接指定寄存器 V、Z 修饰
- ⑩ 操作数使用注意事项
- Ⓜ 适用机种, 有符号'✓'表示该机种支持此运动指令

◆ 运动指令的输入

运动指令中有些指令仅有指令部份 (指令名) 构成, 例如: DRVZ、SETR...或 ABS 等等, 但是大部份都是指令部份再加上好几个操作数所组合而成。运动指令前不须条件接点。

6.2.2 G-Code 指令

◆ G-Code 指令的结构可分为两部份: 指令名及操作数

指令名:		表示指令执行功能
操作数	菜单示:	参数标记(X, Y, Z, I, J, K, R, F)
	参数设定:	设定参数数值大小

- 操作数菜单之参数标记必须填写。
- 操作数参数设定可输入型式: 输入阿拉伯数字整数或小数, 操作数参数都以 32 位计算。

例 1: G00 X100 Y100 例 2: G00 X100.0 Y100.0。

- 20M 机种操作数参数除上述形式设定, 还可输入 16 位寄存器 (D) 与 32 位寄存器 (DD)。例如下面所示:

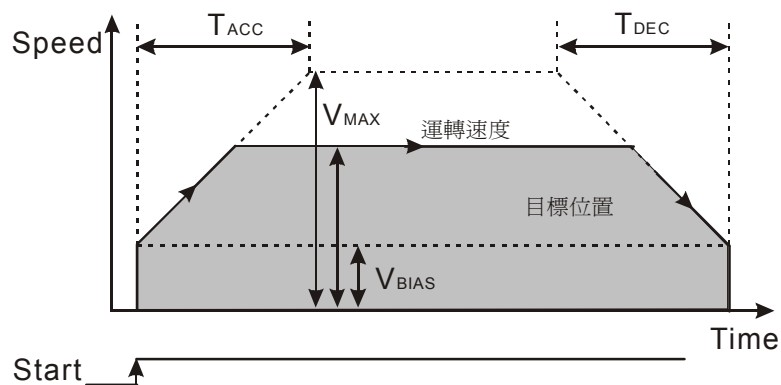
G0 XD11 YDD20 FDD25;

6.3 运动指令

MON 00	DRV	X (P ₁) FX (V ₁) Y (P ₂) FY (V ₂) Z (P ₃) FZ (V ₃)					高速定位	适用机种	
								20PM	10PM
								✓	
		字符装置			双字符装置			操作数使用注意：此指令有支持装置 (device)， 可经由 V、Z 变址修饰，各装置使用范围请参考 功能规格表。后面可接 M 码指令	
		K	H	D	KK	HH	DD		
P ₁	*	*	*	*	*	*	*		
V ₁	*	*	*	*	*	*	*		
P ₂	*	*	*	*	*	*	*		
V ₂	*	*	*	*	*	*	*		
P ₃	*	*	*	*	*	*	*		
V ₃	*	*	*	*	*	*	*		

指令说明

- ◆ **P₁**: X 轴目标位置。 **V₁**: X 轴移动速度。 **P₂**: Y 轴目标位置。 **V₂**: Y 轴移动速度。
P₃: Z 轴目标位置。 **V₃**: Z 轴移动速度。
- ◆ **V₁**、**V₂**、**V₃** 的移动速度最高设定值为 V_{MAX} 。
- ◆ 参数范围: K (-32,768~32,767)、H (0~FFFF)、D (0~9999) 占 16 位, KK (-2,147,483,648 ~ 2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 加减速时间与启动速度可由特 D 来设定。
- ◆ 加减速时间会依最高速度设定等比例增加或减少。
- ◆ 动作图:



- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。
- ◆ 编写指令方式: 有设定轴移动速度, 一定要有该轴 (X 轴、Y 轴或 Z 轴) 目标位置, 有目标位置不一定要设移动速度, 以此规则可有很多种组合。
- ◆ 20D 机种只支持两轴高速定位 (X 轴或 Y 轴), 由上述下指令方式得知, 此指令有 8 种组合。

6 运动指令与 G-Code 分类基本使用说明

NO.	运动指令	操作数参数组合
1	DRV	X (P ₁)
2		X (P ₁) FX (V ₁)
3		Y (P ₂)
4		Y (P ₂) FY (V ₂)
5		X (P ₁) Y (P ₂)
6		X (P ₁) Y (P ₂) FY (V ₂)
7		X (P ₁) FX (V ₁) Y (P ₂)
8		X (P ₁) FX (V ₁) Y (P ₂) FY (V ₂)

- ◆ 20M 机种支持三轴高速定位（X 轴、Y 轴或 Z 轴），由上述编写指令方式，此指令有 26 种组合。

NO.	运动指令	操作数参数组合
1	DRV	X (P ₁)
2		X (P ₁) FX (V ₁)
3		Y (P ₂)
4		Y (P ₂) FY (V ₂)
5		X (P ₁) Y (P ₂)
6		X (P ₁) Y (P ₂) FY (V ₂)
7		X (P ₁) FX (V ₁) Y (P ₂)
8		X (P ₁) FX (V ₁) Y (P ₂) FY (V ₂)
9		Z (P ₃)
10		X (P ₁) Z (P ₃)
11		X (P ₁) FX (V ₁) Z (P ₃)
12		Y (P ₂) Z (P ₃)
13		Y (P ₂) FY (V ₂) Z (P ₃)
14		X (P ₁) Y (P ₂) Z (P ₃)
15		X (P ₁) Y (P ₂) FY (V ₂) Z (P ₃)
16		X (P ₁) FX (V ₁) Y (P ₂) Z (P ₃)
17		X (P ₁) FX (V ₁) Y (P ₂) FY (V ₂) Z (P ₃)
18		Z (P ₃) FZ (V ₃)
19		X (P ₁) Z (P ₃) FZ (V ₃)
20		X (P ₁) FX (V ₁) Z (P ₃) FZ (V ₃)
21		Y (P ₂) Z (P ₃) FZ (V ₃)
22		Y (P ₂) FY (V ₂) Z (P ₃) FZ (V ₃)
23		X (P ₁) Y (P ₂) Z (P ₃) FZ (V ₃)
24		X (P ₁) Y (P ₂) FY (V ₂) Z (P ₃) FZ (V ₃)
25		X (P ₁) FX (V ₁) Y (P ₂) Z (P ₃) FZ (V ₃)
26		X (P ₁) FX (V ₁) Y (P ₂) FY (V ₂) Z (P ₃) FZ (V ₃)

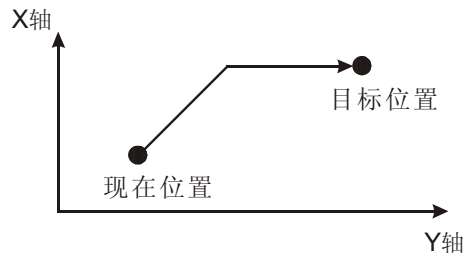
程序范例

- ◆ 有设定该轴目标位置，但没设定该轴移动速度，则以最高速度 V_{MAX} 运转。

- ◆ **DRV XK12345 YH7567 FYKK40000;**

此指令是快速两轴直线移动到目标位置 (K12,345, H7567)，其目标位置有可能是绝对坐标或是相对坐标，这由之前最靠近这指令的定义坐标系指令(G90、G91、ABST、INCT)来决定。而 X 轴移动速度没设定，将以最高速度 V_{MAX} 运转速度输出，Y 轴则以每秒 40K 脉冲速度。

- ◆ 移动路径:



- ◆ 操作数装置组合:

DRV XKK-345289 FXD100 YDD10Z5 FYDD102

DRV XDD20 FXHH2345 YKK456 FYDD0 Z567 Fz234

以上的指令都是合法，其中由 D 装置为间接设定值。

补充说明

- ◆ 特殊寄存器说明:

D1822、D1823 X 轴最高输出速度设定

D1824、D1825 X 轴启动频率设定

D1836 X 轴加速时间设定

D1837 X 轴减速时间设定

D1902、D1903 Y 轴最高输出速度设定

D1904、D1905 Y 轴启动频率设定

D1916 Y 轴加速时间设定

D1917 Y 轴减速时间设定

D1982、D1983 Z 轴最高输出速度设定

D1984、D1985 Z 轴启动频率设定

D1996 Z 轴加速时间设定

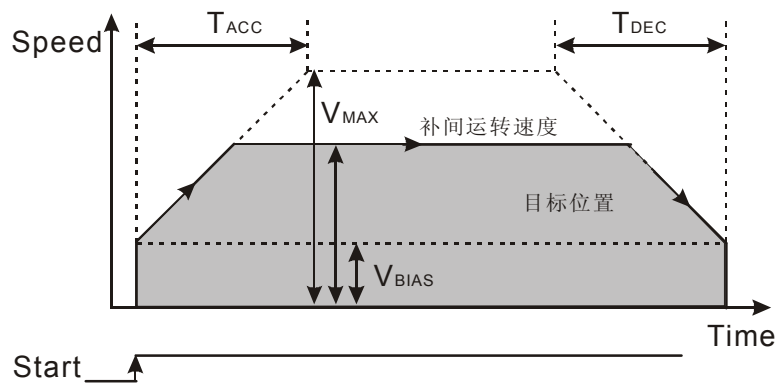
D1997 Z 轴减速时间设定

6 运动指令与 G-Code 分类基本使用说明

MON 01	LIN	(P ₁) Y (P ₂) Z (P ₃) F (V)	直线补间移动 (可考虑剩余距离)			适用机种	
						20PM	10PM
						✓	
	字符装置			双字符装置			操作数使用注意：此指令有支持装置 (device)， 可经由 V、Z 变址修饰，各装置使用范围请参考 功能规格表。后面可接 M 码指令
	K	H	D	KK	HH	DD	
	P ₁	*	*	*	*	*	
	P ₂	*	*	*	*	*	
	P ₃	*	*	*	*	*	
F	*	*	*	*	*		

指令说明

- ◆ P₁: X 轴目标位置。P₂: Y 轴目标位置。P₃: Z 轴目标位置。V: 直线补间运转速度。
- ◆ V: 运转速度设定最高 V_{MAX}。
- ◆ 参数范围：K (-32,768~32,767)、H (0~FFFF)、D (0~9999) 占 16 位，KK (-2,147,483,648 ~2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 加减速时间与启动速度可由特 D 来设定。
- ◆ 加减速时间会依最高速度设定等比例增加或减少。
- ◆ 动作图：



- ◆ 补间运转速度可由特殊寄存器 X 轴：D1850~D1851，Y 轴：D1930~D1931，Z 轴：D2010~D2011 监控。
- ◆ 在特殊寄存器 D1865 可由设定停止模式，考虑剩余距离（设定请参考补充说明）。
- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。
- ◆ 编写指令方式：一定要有目标位置，但不一定要设移动速度，以此规则可有很多种组合。
- ◆ 20D 机种只支持两轴同动补间移动（X 轴或 Y 轴），由上述下指令方式得知，此指令有 6 种组合。

NO.	运动指令	操作数参数组合
1	LIN	X (P ₁)
2		X (P ₁) F (V)
3		Y (P ₂)
4		Y (P ₂) F (V)
5		X (P ₁) Y (P ₂)
6		X (P ₁) Y (P ₂) F (V)

- ◆ 20M 机种支持三轴同动补间移动（X 轴、Y 轴或 Z 轴），由上述下指令方式得知，此指令有 14 种组合。

NO.	运动指令	操作数参数组合
1	LIN	X (P ₁)
2		X (P ₁) F (V)
3		Y (P ₂)
4		Y (P ₂) F (V)
5		X (P ₁) Y (P ₂)
6		X (P ₁) Y (P ₂) F (V)
7		Z (P ₃)
8		Z (P ₃) F (V)
9		X (P ₁) Z (P ₃)
10		X (P ₁) Z (P ₃) F (V)
11		Y (P ₂) Z (P ₃)
12		Y (P ₂) Z (P ₃) F (V)
13		X (P ₁) Y (P ₂) Z (P ₃)
14		X (P ₁) Y (P ₂) Z (P ₃) F (V)

- ◆ 有设定目标位置，但没设定移动速度，则以最高速度 V_{MAX} 运转。

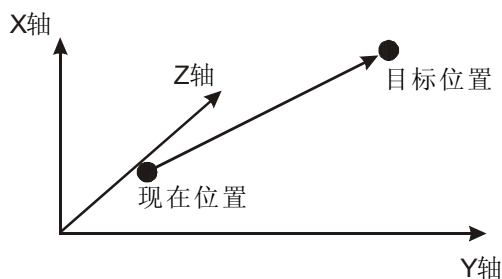
6 运动指令与 G-Code 分类基本使用说明

程序范例

◆ **LIN XK12345 YH7567 ZKK3280 FKK40000;**

此指令是直线移动到目标位置 (K12,345, H7567, ZKK3280)，其目标位置有可能是绝对坐标或是相对坐标，由最靠近这指令的定义坐标系指令来决定。而以 40kHz 直线移动到目的。

◆ 移动路径:



补充说明

◆ 相关寄存器说明: (各项特殊寄存器内容请参考第三章各功能装置说明)

D1822、D1823 X 轴最高输出速度设定

D1824、D1825 X 轴启动频率设定

D1836 X 轴加速时间设定

D1837 X 轴减速时间设定

D1865 OX 0~99 停止模式

K1 → 下次启动走完未完行程 (考虑剩余距离)

K2 → 下次启动执行下一指令, 其它 → 重新开始

D1902、D1903 Y 轴最高输出速度设定

D1904、D1905 Y 轴启动频率设定

D1916 Y 轴加速时间设定

D1917 Y 轴减速时间设定

D1982、D1983 Z 轴最高输出速度设定

D1984、D1985 Z 轴启动频率设定

D1996 Z 轴加速时间设定

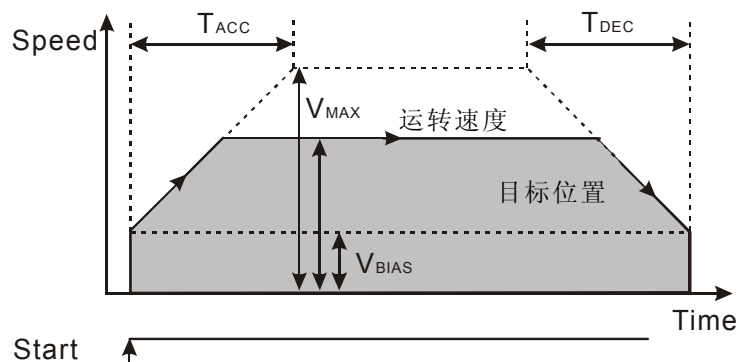
D1997 Z 轴减速时间设定

6 运动指令与 G-Code 分类基本使用说明

MON	CW / CCW	X (P ₁) Y (P ₂) Z (P ₃) I (P ₄) J (P ₅) K (P ₆) F (V)	顺时针圆弧/螺旋移动 逆时针圆弧移动/螺旋移动 (设定圆心位置)			适用机种	
02						20PM	10PM
03						✓	
	字符装置			双字符装置			操作数使用注意：此指令有支持装置 (device)，可经由 V、Z 变址修饰，后面可接 M 码指令
	K	H	D	KK	HH	DD	
P ₁	*	*	*	*	*	*	
P ₂	*	*	*	*	*	*	
P ₃	*	*	*	*	*	*	
P ₄	*	*	*	*	*	*	
P ₅	*	*	*	*	*	*	
P ₆	*	*	*	*	*	*	
V	*	*	*	*	*	*	

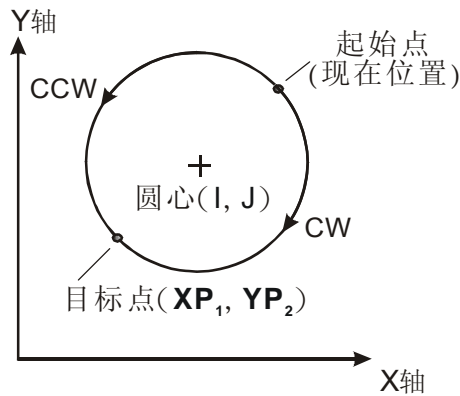
指令说明

- ◆ **P₁**: X轴目标位置。 **P₂**: Y轴目标位置。 **P₃**: Z轴目标位置。 **P₄**: 圆弧 X轴圆心向量坐标。 **P₅**: 圆弧 Y轴圆心向量坐标。 **P₆**: 圆弧 Z轴圆心向量坐标。 **V**: 圆弧/螺旋补间运转速度。
- ◆ **P₄**、**P₅**和**P₆**: 以圆弧或螺旋圆心为基准点，相对于目前位置向量值，也就是圆心坐标减掉目前位置(圆弧起点)。
- ◆ **V**: 运转速度设定最高 500kHz。
- ◆ 参数范围:K(-32,768~32,767)、H(0~FFFF)、D(0~9999)占 16 位, KK(-2,147,483,648 ~ 2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 加减速时间与启动速度可由特 D 来设定。
- ◆ 加减速时间会依最高速度设定等比例增加或减少。
- ◆ X-Y-Z 轴个别输出运转动作图:

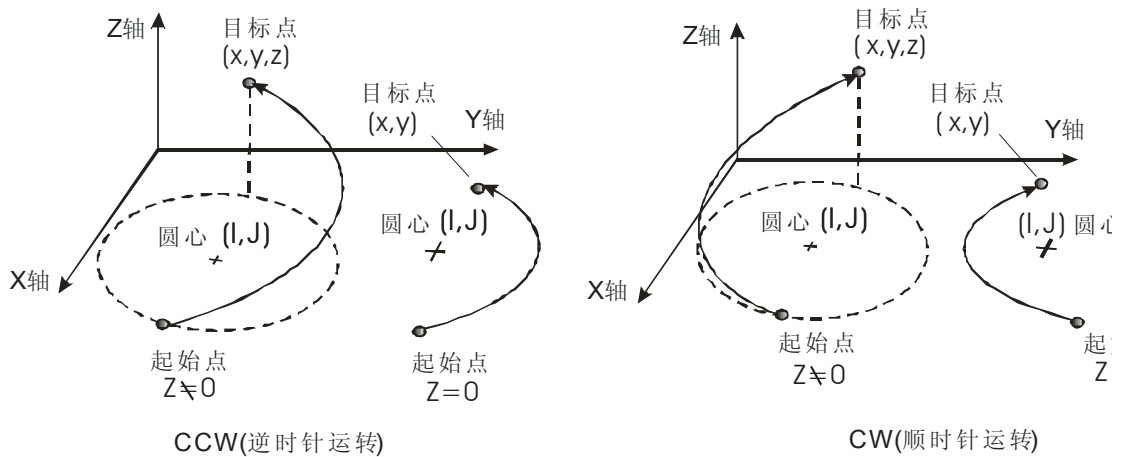


- ◆ 双轴圆弧补间运转：20D 机种只能用 XY 平面 XY 两轴圆弧补间。20M 机种可使用 G-CODE 指令中的 G17(XY 平面设定)、G18(XZ 平面设定)或 G19(YZ 平面设定)，控制任两轴圆弧补间。下图所示为 XY 两轴圆弧补间。

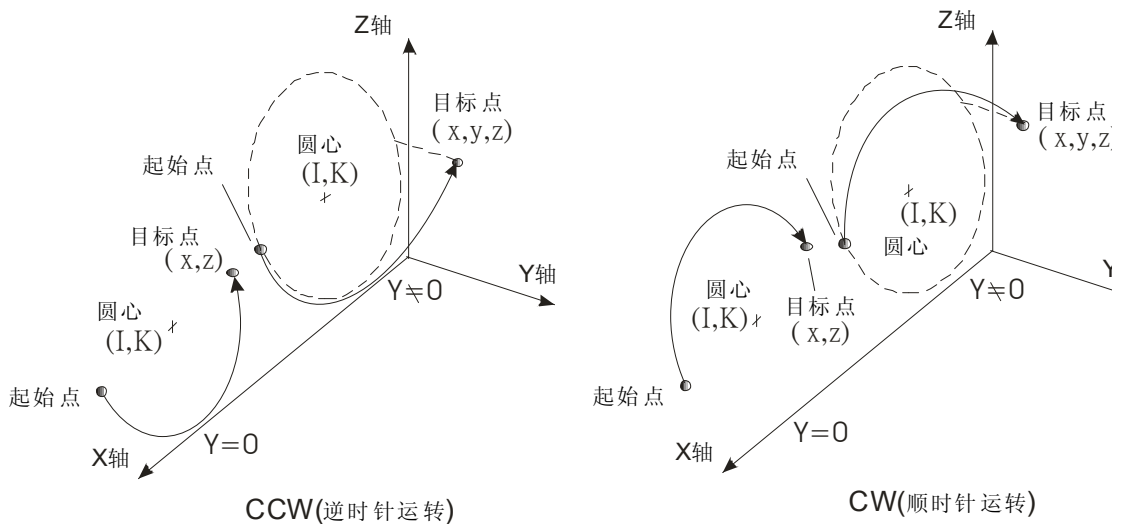
6 运动指令与 G-Code 分类基本使用说明



- ◆ 螺旋补间运转：适用于 20M 机种，此功能为两轴圆弧补间，再增加一轴垂直两轴圆弧平面的高度移动，而这三轴是同动运转，其为圆弧补间功能的延伸。当执行螺旋补间指令而高度值变化是零，就是圆弧功能。
- ◆ 下图为螺旋补间在 G17 模式(XY 平面)下，假如 Z 轴无变化量($Z=0$)，螺旋补间运作同 XY 轴两轴圆弧补间。

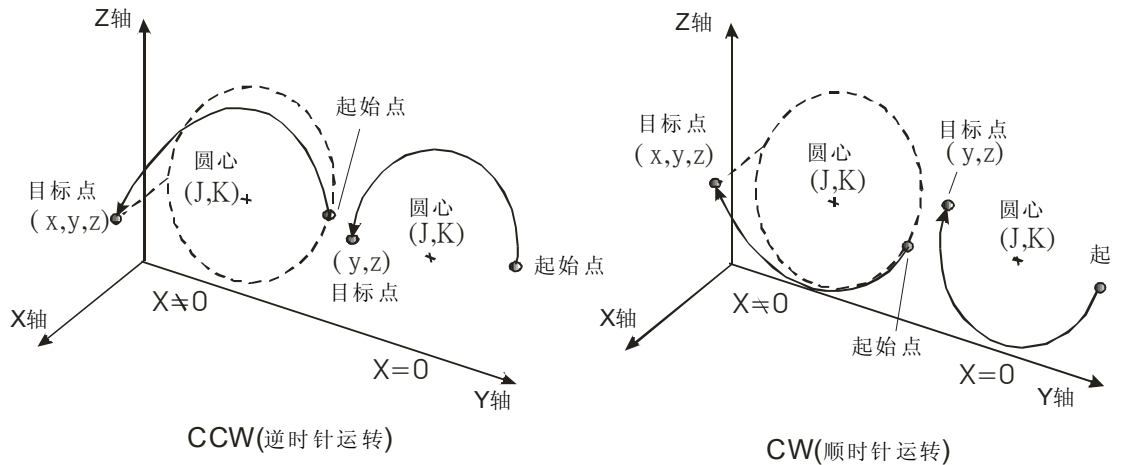


- ◆ 下图为螺旋补间在 G18 模式(XZ 平面)下，假如 Y 轴无变化量($Y=0$)，螺旋补间运作同 XZ 轴两轴圆弧补间。



6 运动指令与 G-Code 分类基本使用说明

- ◆ 下图为螺旋补间在 G19 模式(YZ 平面)下，假如 X 轴无变化量($X=0$)，螺旋补间运作同 YZ 轴两轴圆弧补间。



- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。
- ◆ 编写指令原则：(1)一定要有目标位置，及圆心位置，但不一定要设移动速度。(2)目标位置任一轴与目前位置对应轴无变化量，可省略。(3)圆心位置两轴任一轴与目前位置对应轴无变化量，可省略。按上述三种原则，编写圆弧/螺旋指令可有多种组合。
- ◆ 20D 机种只支持两轴 XY 平面圆弧补间移动功能，按照编写指令原则有 18 种组合，如下表所示。

NO.	运动指令	操作数参数组合
1	CW/CCW	$X(P_1) I(P_3)$
2		$X(P_1) I(P_3) F(V)$
3		$X(P_1) J(P_4)$
4		$X(P_1) J(P_4) F(V)$
5		$X(P_1) I(P_3) J(P_4)$
6		$X(P_1) I(P_3) J(P_4) F(V)$
7		$Y(P_2) I(P_3)$
8		$Y(P_2) I(P_3) F(V)$
9		$Y(P_2) J(P_4)$
10		$Y(P_2) J(P_4) F(V)$
11		$Y(P_2) I(P_3) J(P_4)$
12		$Y(P_2) I(P_3) J(P_4) F(V)$
13		$X(P_1) Y(P_2) I(P_3)$
14		$X(P_1) Y(P_2) I(P_3) F(V)$
15		$X(P_1) Y(P_2) J(P_4)$
16		$X(P_1) Y(P_2) J(P_4) F(V)$
17		$X(P_1) Y(P_2) I(P_3) J(P_4)$
18		$X(P_1) Y(P_2) I(P_3) J(P_4) F(V)$

- ◆ 20M 机种可做圆弧/螺旋补间运作，配合 G-CODE 指令中的 G17、G18 和 G19 等平面

6 运动指令与 G-Code 分类基本使用说明

转换，按照编写指令原则有 78 种组合，如下表所示。

NO.	运动指令	操作数参数组合	G17	G18	G19
1		X(P ₁) I(P ₃)	✓	✓	
2		X(P ₁) I(P ₃) F(V)	✓	✓	
3		X(P ₁) J(P ₄)	✓		
4		X(P ₁) J(P ₄) F(V)	✓		
5		X(P ₁) I(P ₃) J(P ₄)	✓		
6		X(P ₁) I(P ₃) J(P ₄) F(V)	✓		
7		Y(P ₂) I(P ₃)	✓		
8		Y(P ₂) I(P ₃) F(V)	✓		
9		Y(P ₂) J(P ₄)	✓		✓
10		Y(P ₂) J(P ₄) F(V)	✓		✓
11		Y(P ₂) I(P ₃) J(P ₄)	✓		
12		Y(P ₂) I(P ₃) J(P ₄) F(V)	✓		
13		X(P ₁) Y(P ₂) I(P ₃)	✓	✓	
14		X(P ₁) Y(P ₂) I(P ₃) F(V)	✓	✓	
15		X(P ₁) Y(P ₂) J(P ₄)	✓		✓
16		X(P ₁) Y(P ₂) J(P ₄) F(V)	✓		✓
17		X(P ₁) Y(P ₂) I(P ₃) J(P ₄)	✓		
18		X(P ₁) Y(P ₂) I(P ₃) J(P ₄) F(V)	✓		
19		X(P ₁) Z(P ₃) I(P ₃)	✓	✓	
20		X(P ₁) Z(P ₃) I(P ₃) F(V)	✓	✓	
21	CW/CCW	X(P ₁) Z(P ₃) J(P ₄)	✓		✓
22		X(P ₁) Z(P ₃) J(P ₄) F(V)	✓		✓
23		X(P ₁) Z(P ₃) I(P ₃) J(P ₄)	✓		
24		X(P ₁) Z(P ₃) I(P ₃) J(P ₄) F(V)	✓		
25		Y(P ₂) Z(P ₃) I(P ₃)	✓	✓	
26		Y(P ₂) Z(P ₃) I(P ₃) F(V)	✓	✓	
27		Y(P ₂) Z(P ₃) J(P ₄)	✓		✓
28		Y(P ₂) Z(P ₃) J(P ₄) F(V)	✓		✓
29		Y(P ₂) Z(P ₃) I(P ₃) J(P ₄)	✓		
30		Y(P ₂) Z(P ₃) I(P ₃) J(P ₄) F(V)	✓		
31		X(P ₁) Y(P ₂) Z(P ₃) I(P ₃)	✓	✓	
32		X(P ₁) Y(P ₂) Z(P ₃) I(P ₃) F(V)	✓	✓	
33		X(P ₁) Y(P ₂) Z(P ₃) J(P ₄)	✓		✓
34		X(P ₁) Y(P ₂) Z(P ₃) J(P ₄) F(V)	✓		✓
35		X(P ₁) Y(P ₂) Z(P ₃) I(P ₃) J(P ₄)	✓		
36		X(P ₁) Y(P ₂) Z(P ₃) I(P ₃) J(P ₄) F(V)	✓		
37		X(P ₁) K(P ₆)		✓	
38		X(P ₁) K(P ₆) F(V)		✓	
39		X(P ₁) I(P ₃) K(P ₆)		✓	
40		X(P ₁) I(P ₃) K(P ₆) F(V)		✓	

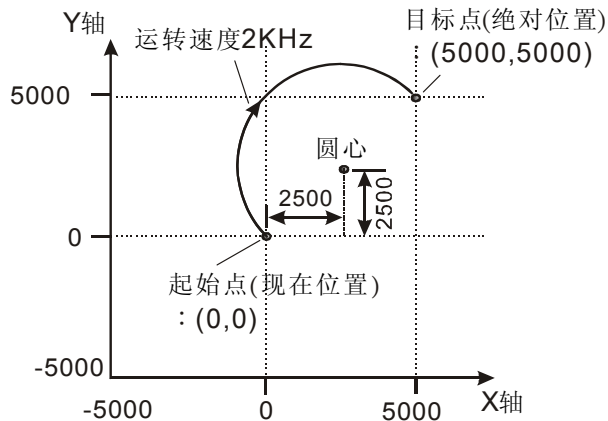
6 运动指令与 G-Code 分类基本使用说明

NO.	运动指令	操作数参数组合			
41	CW/CCW	Z (P ₃) I (P ₃)		✓	
42		Z (P ₃) I (P ₃) F (V)		✓	
43		Z (P ₃) K (P ₆)		✓	✓
44		Z (P ₃) K (P ₆) F (V)		✓	✓
45		Z (P ₃) I (P ₃) K (P ₆)		✓	
46		Z (P ₃) I (P ₃) K (P ₆) F (V)		✓	
47		X (P ₁) Z (P ₃) K (P ₆)		✓	✓
48		X (P ₁) Z (P ₃) K (P ₆) F (V)		✓	✓
49		X (P ₁) Z (P ₃) I (P ₃) K (P ₆)		✓	
50		X (P ₁) Z (P ₃) I (P ₃) K (P ₆) F (V)		✓	
51		X (P ₁) Y (P ₂) K (P ₆)		✓	✓
52		X (P ₁) Y (P ₂) K (P ₆) F (V)		✓	✓
53		X (P ₁) Y (P ₂) I (P ₃) K (P ₆)		✓	
54		X (P ₁) Y (P ₂) I (P ₃) K (P ₆) F (V)		✓	
55		Y (P ₂) Z (P ₃) K (P ₆)		✓	✓
56		Y (P ₂) Z (P ₃) K (P ₆) F (V)		✓	✓
57		Y (P ₂) Z (P ₃) I (P ₃) K (P ₆)		✓	
58		Y (P ₂) Z (P ₃) I (P ₃) K (P ₆) F (V)		✓	
59		X (P ₁) Y (P ₂) Z (P ₃) K (P ₆)		✓	✓
60		X (P ₁) Y (P ₂) Z (P ₃) K (P ₆) F (V)		✓	✓
61		X (P ₁) Y (P ₂) Z (P ₃) I (P ₃) K (P ₆)		✓	
62		X (P ₁) Y (P ₂) Z (P ₃) I (P ₃) K (P ₆) F (V)		✓	
63		Y (P ₂) K (P ₆)			✓
64		Y (P ₂) K (P ₆) F (V)			✓
65		Y (P ₂) J (P ₄) K (P ₆)			✓
66		Y (P ₂) J (P ₄) K (P ₆) F (V)			✓
67		Z (P ₃) J (P ₄)			✓
68		Z (P ₃) J (P ₄) F (V)			✓
69		Z (P ₃) J (P ₄) K (P ₆)			✓
70		Z (P ₃) J (P ₄) K (P ₆) F (V)			✓
71		Y (P ₂) Z (P ₃) J (P ₄) K (P ₆)			✓
72		Y (P ₂) Z (P ₃) J (P ₄) K (P ₆) F (V)			✓
73		X (P ₁) Y (P ₂) J (P ₄) K (P ₆)			✓
74		X (P ₁) Y (P ₂) J (P ₄) K (P ₆) F (V)			✓
75		X (P ₁) Z (P ₃) J (P ₄) K (P ₆)			✓
76		X (P ₁) Z (P ₃) J (P ₄) K (P ₆) F (V)			✓
77		X (P ₁) Y (P ₂) Z (P ₃) J (P ₄) K (P ₆)			✓
78		X (P ₁) Y (P ₂) Z (P ₃) J (P ₄) K (P ₆) F (V)			✓

6 运动指令与 G-Code 分类基本使用说明

- ◆ 有设定目标位置，但没设定移动速度，则以最高速度 V_{MAX} 运转。
- ◆ 圆弧移动路径或螺旋路径垂直投影轨迹可以全圆 360 度。
- ◆ 范例一 说明：程序一开始为设定绝对坐标，**CW** 为顺时针圆弧指令，圆弧起点位置为 (0,0)，圆弧终点位置为 (5000,5000)，圆心在相对于圆弧起点位置 (2500,2500)，而 2000Hz 输出速度移动。

程序范例

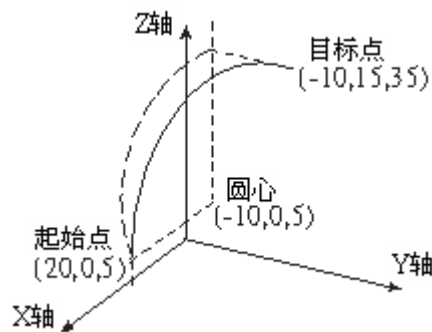


程序写法如下：

ABST;

CW XK10000 YK10000 IK2500 JK2500 FK2000;

- ◆ 范例二 说明：程序一开始为设定绝对坐标，在 **XZ** 平面模式下，**CW** 为顺时针螺旋指令，螺旋终点位置为 (-10,15,35)，螺旋圆心为 (-10,0,5)，而 2000Hz 输出速度移动。



程序写法如下：

ABS;

G18;

CW XK-10 YK15 ZK35 IK-30 JK0(可省略) KK0(可省略) FK2000;

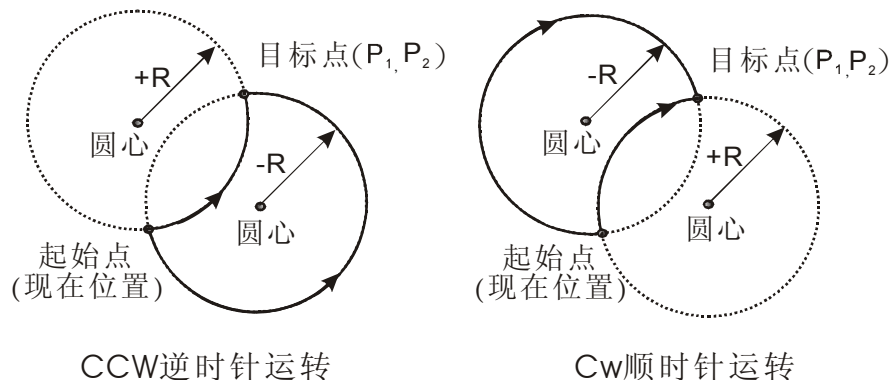
6 运动指令与 G-Code 分类基本使用说明

MON 04 05	CW / CCW	X(P ₁) Y(P ₂) Z(P ₃) R(L) F(V)	顺时针圆弧/螺旋移动 逆时针圆弧/螺旋移动 (设定半径长度)	适用机种	
				20PM	10PM
				✓	

	字符装置			双字符装置			操作数使用注意：此指令有支持装置(device)， 可经由 V、Z 变址修饰，后面可接 M 码指令
	K	H	D	KK	HH	DD	
P ₁	*	*	*	*	*	*	
P ₂	*	*	*	*	*	*	
P ₃	*	*	*	*	*	*	
L	*	*	*	*	*	*	
V	*	*	*	*	*	*	

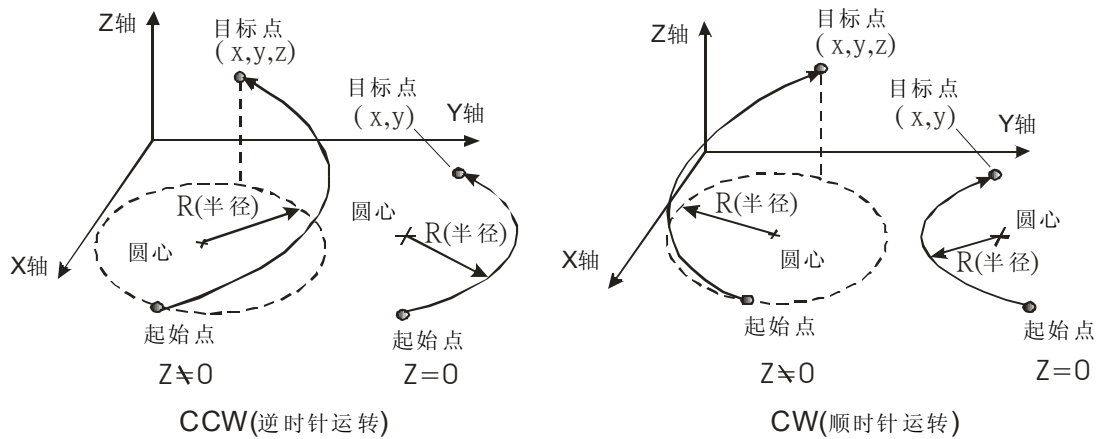
指令说明

- ◆ **P₁**: X 轴目标位置。 **P₂**: Y 轴目标位置。 **P₃**: Z 轴目标位置。 **L**: 圆弧半径 (弧角小于 180 度, R 为"+", 弧角等于大于 180 度, R 为"-")。 **V**: 圆弧/螺旋补间运转速度。
- ◆ **V**: 运转速度设定最高 V_{MAX}。
- ◆ 参数范围: K (-32,768~32,767)、H (0~FFFF)、D (0~9999) 占 16 位, KK (-2,147,483,648 ~ 2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 加减速时间与启动速度可由特 D 来设定。
- ◆ 加减速时间会依最高速度设定等比例增加或减少。
- ◆ 双轴圆弧补间运转: 20D 机种只能用 XY 平面 XY 两轴圆弧补间。20M 机种可使用 G-CODE 指令中的 G17(XY 平面设定)、G18(XZ 平面设定)或 G19(YZ 平面设定), 控制任两轴圆弧补间。下图所示为两轴圆弧补间。

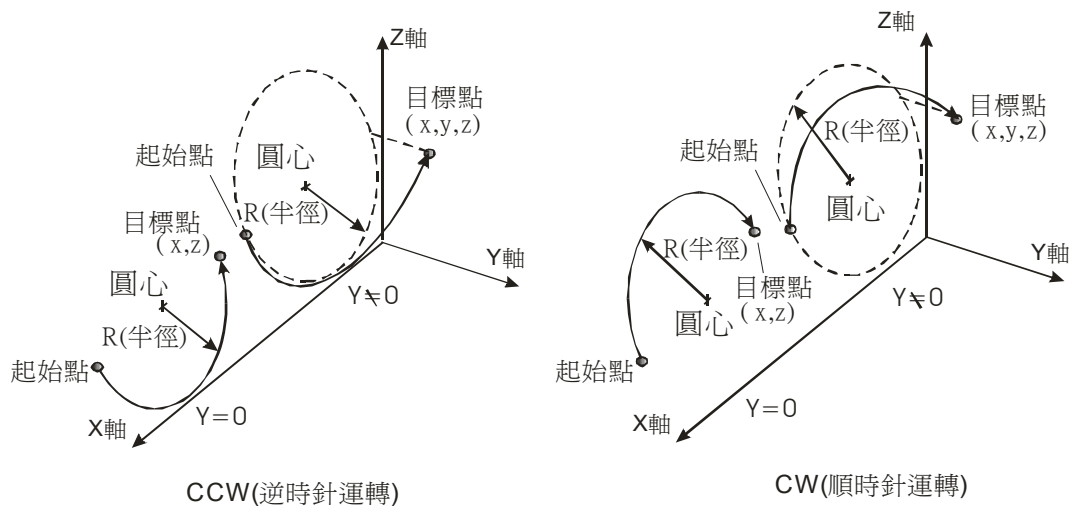


- ◆ 螺旋补间运转: 适用于 20M 机种, 此功能为两轴圆弧补间, 再增加一轴垂直两轴圆弧平面的高度移动, 而这三轴是同动运转, 其为圆弧补间功能的延伸。当执行螺旋补间指令而高度值是零, 就是圆弧功能。
- ◆ 下图为螺旋补间在 G17 模式(XY 平面)下, 假如 Z 轴无变化量(Z=0), 螺旋补间运作同 XY 轴两轴圆弧补间。

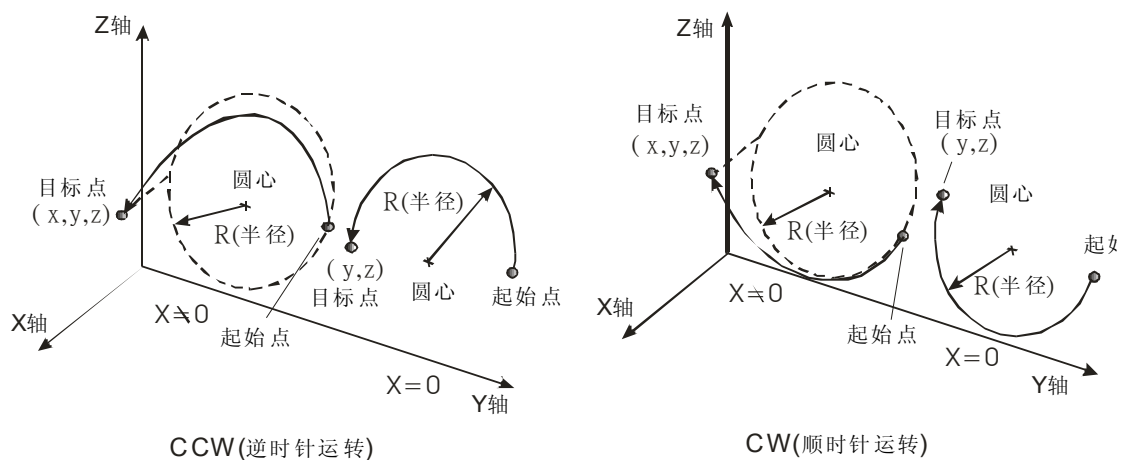
6 运动指令与 G-Code 分类基本使用说明



- ◆ 下图为螺旋补间在 G18 模式(XZ 平面)下, 假如 Y 轴无变化量(Y=0), 螺旋补间运作同 XZ 轴两轴圆弧补间。



- ◆ 下图为螺旋补间在 G19 模式(YZ 平面)下, 假如 X 轴无变化量(X=0), 螺旋补间运作同 YZ 轴两轴圆弧补间。



- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。
- ◆ 编写指令原则: (1)一定要有目标位置, 及圆弧半径, 但不一定要设移动速度。(2)目

6 运动指令与 G-Code 分类基本使用说明

标位置任一轴与目前位置对应轴无变化量，可省略。按上述两个原则，编写圆弧/螺旋指令可有多种组合。

- ◆ 20D 机种只支持两轴 XY 平面圆弧补间移动功能，按照编写指令原则有 6 种组合，如下表所示。

NO.	运动指令	操作数参数组合
1	CW/CCW	X(P ₁) R(L)
2		X(P ₁) R(L) F(V)
3		Y(P ₂) R(L)
4		Y(P ₂) R(L) F(V)
5		X(P ₁) Y(P ₂) R(L)
6		X(P ₁) Y(P ₂) R(L) F(V)

- ◆ 20M 机种可做圆弧/螺旋补间运作，配合 G-CODE 指令中的 G17、G18 和 G19 等平面转换，即按照编写指令原则有 14 种组合，如下表所示。

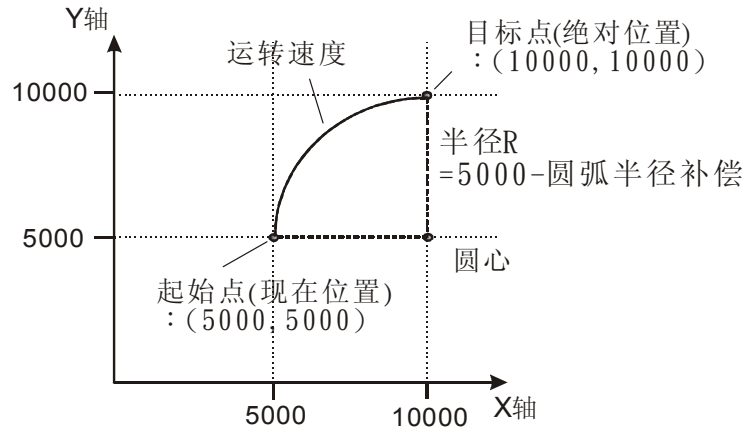
NO.	运动指令	操作数参数组合	G17	G18	G19
1	CW/CCW	X(P ₁) R(L)	✓	✓	
2		X(P ₁) R(L) F(V)	✓	✓	
3		Y(P ₂) R(L)	✓		✓
4		Y(P ₂) R(L) F(V)	✓		✓
5		X(P ₁) Y(P ₂) R(L)	✓	✓	✓
6		X(P ₁) Y(P ₂) R(L) F(V)	✓	✓	✓
7		X(P ₁) Z(P ₃) R(L)	✓	✓	✓
8		X(P ₁) Z(P ₃) R(L) F(V)	✓	✓	✓
9		Y(P ₂) Z(P ₃) R(L)	✓	✓	✓
10		Y(P ₂) Z(P ₃) R(L) F(V)	✓	✓	✓
11		X(P ₁) Z(P ₃) Y(P ₂) R(L)	✓	✓	✓
12		X(P ₁) Z(P ₃) Y(P ₂) R(L) F(V)	✓	✓	✓
13		Z(P ₃) R(L)		✓	✓
14		Z(P ₃) R(L) F(V)		✓	✓

- ◆ 有设定目标位置，但没设定移动速度，则以最高速度 V_{MAX} 运转。

6 运动指令与 G-Code 分类基本使用说明

程序范例

- ◆ 范例一 说明：程序一开始设定绝对坐标，CW 为顺时针圆弧指令，圆弧起点位置为 (5000,5000)，圆弧终点位置为 (10000,10000)，半径为 5000，而弧角小于 180 度，所以是正。而以每秒 1,000 个脉冲输出速度移动。

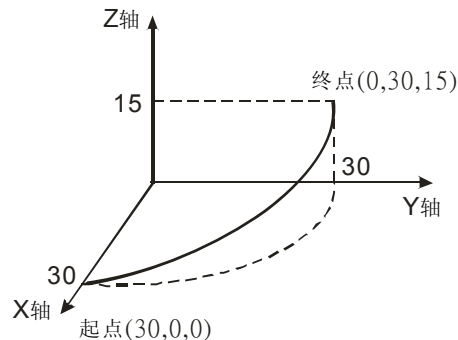


程序写法如下：

ABST;

CW XK10000 YK10000 RK5000 FK1000;

- ◆ 范例二 说明：程序一开始设定绝对坐标，CCW 为逆时针螺旋指令，螺旋起点位置为 (30,0,0)，螺旋终点位置为 (0,30,15)，半径为 30.0，而弧角小于 180 度，所以是正。而以每秒 100 个脉冲输出速度移动。



程序写法如下：

ABST;

G17;

CCW XK0 YK30 ZK15 RK30 FK100;

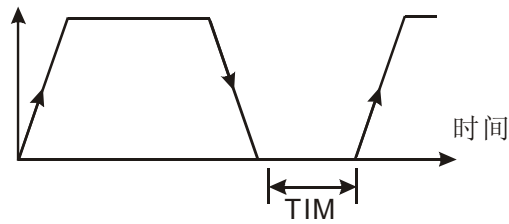
- ◆ 圆弧半径补偿设定方法请参考 MON 15 RADC 指令说明。
- ◆ 此圆弧移动无法移动全圆 360 度。

6 运动指令与 G-Code 分类基本使用说明

MON	TIM	(T)					停顿时间		适用机种	
06									20PM	10PM
							✓			
	字符装置			双字符装置			操作数使用注意：此指令有支持装置 (device)，可由 V、Z 变址修饰，各装置使用范围请参考功能规格表，后面可接 M 码指令			
	K	H	D	KK	HH	DD				
T	*	*	*	*	*	*				

指令说明

- ◆ **T**：设定停顿时间，单位为 10 毫秒。“K100”表示停顿 1 秒。
- ◆ 停顿时间是用来设定指令与指令间的等待时间



- ◆ 参数范围：K (0~32,767)、H (0~FFFF)、D (0~9999) 占 16 位，KK (-2,147,483,648~2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。

程序范例

- ◆ 程序写法如下所示：



6 运动指令与 G-Code 分类基本使用说明

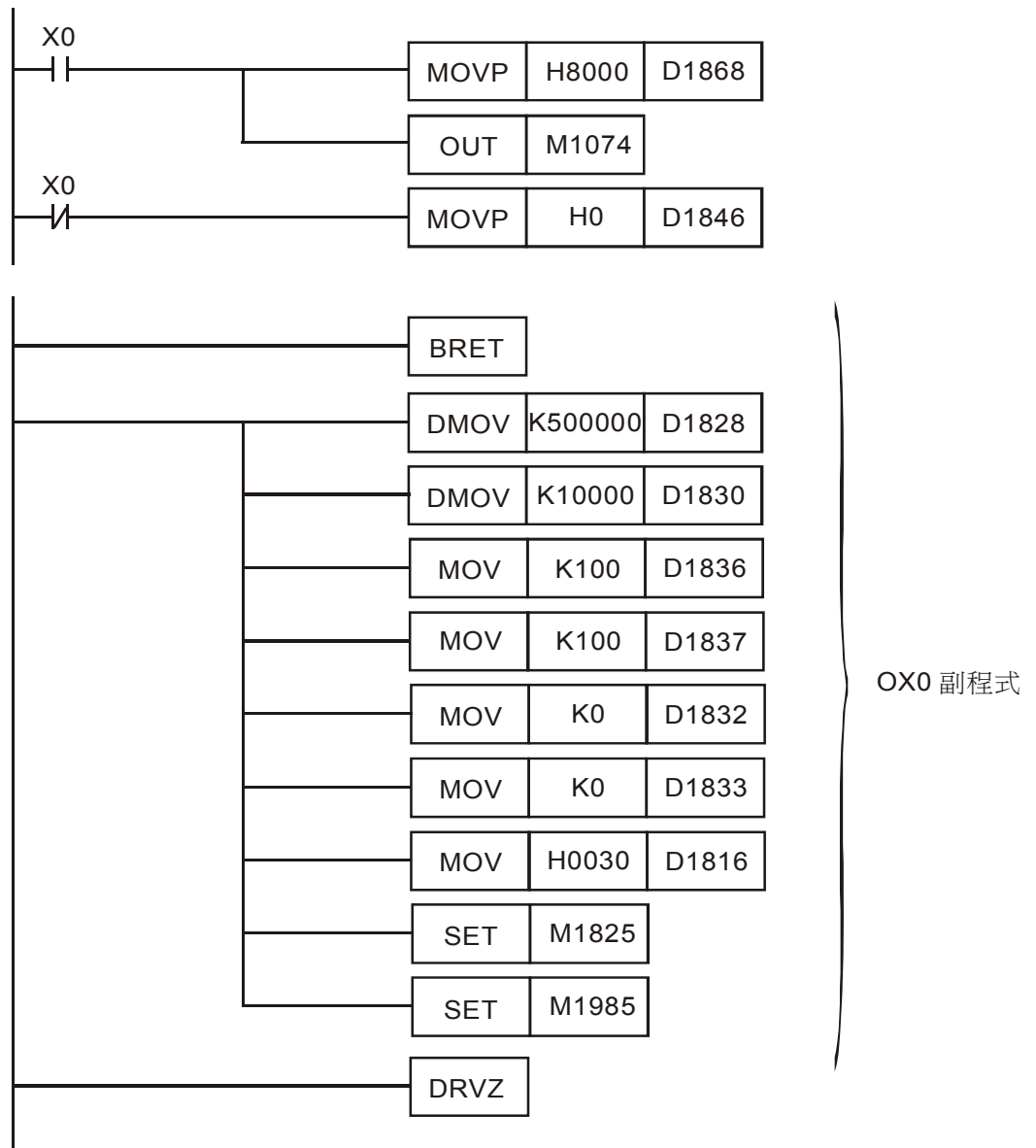
MON 07	DRVZ	无操作数	返回机械原点 (原点复位)	适用机种	
				20PM	10PM
				✓	
无操作数：后面可接 M 码指令					

指令说明

- ◆ 当启动指令 DRVZ, X、Y、Z 轴将同时执行原点复位动作。
- ◆ 在启动 DRVZ 指令前须先设定 DRVZ 相关参数设定值, 相关参数如下: (相关特 D 寄存器请参考本指令补充说明)
 1. 原点复位速度 V_{RT} : 回到机械原点的速度。原点复位速度 V_{RT} 在执行中不允许改变。设定范围: 0~500kHz, 设定范围限制 $V_{MAX} > V_{RT} > V_{BIAS}$
 2. 原点复位减速速度 V_{CR} : 当运转中触发原点信号, 产生的运转速度, 为了精准的定位在原点, 建议将原点复位减速速度 V_{CR} 设定为低速运行。设定范围: 0~500kHz, 设定值须小于原点复位速度, 原点复位减速速度 V_{CR} 在执行中不允许改变。
 3. 加速时间: 到达原点复位速度 V_{RT} 之加速时间
 4. 减速时间: 由速度 V_{RT} 减速到速度 V_{CR} 与 V_{CR} 减速至速度零之减速时间
 5. 原点复位零点 (PG0) 信号数 N: 马达减速停止之参考信号。当侦测到近点 (DOG) 信号产生后, 开始计数零点信号数 N 脉冲, 作为停止参考信号, 设定范围 0~+32,767 PLS
 6. 原点复位之脉冲信号数 P: 马达减速停止之参考信号。正值设定为正方向的脉冲信号数 P, 负值设定为反方向的脉冲信号数 P。设定范围 -32,768~32,767 (PULSE)
 7. X 与 Y 轴原点复位禁止选择。
- ◆ 以下参数由特殊寄存器, X 轴 D1816、Y 轴 D1896、Z 轴 D1976 参数设定
 1. 原点复位方向: $b[8]=0$ 时, 表执行原点复位方向, 往现在位置 (CP) 内容值递减方向执行, 当 $b[8]=1$ 时, 则往现在位置 (CP) 内容值递增方向执行
 2. 原点复位模式: $b[9]=0$: 正常模式。 $b[9]=1$: 覆写模式
 3. 原点复位 DOG 下降缘侦测: $b[10]=0$: 原点复位 DOG 下降缘侦测 On。。 $b[10]=1$: 原点复位 DOG 下降缘侦测 Off
 4. 由上列第 2、3 项可知原点复位有四种模式组合, 此四种组合运作方式请参考第 3.12 章节之动作说明。

程序范例

- ◆ 当起动 X0 后，将执行 OX0 内程序 DRVZ 指令，设定 Y、Z 轴之原点复位禁止，X 轴以 100ms 加速时间到达原点复位速度 (V_{RT}) 500kHz 并寻找机械原点，当原点讯号 DOG 触发则以 100ms 减速时间减速到 10kHz 运转，当 DOG 讯号离开(下降缘触发)，原点复位模式为正常模式，开始计算原点复位零点 (PG0) 信号数 N 与原点复位之脉冲信号数 P，因范例中 N 与 P 信号皆设定为 0，故 DOG 讯号下降缘触发后即完成定位。



补充说明

- ◆ 相关标志信号说明(各项寄存器说明容请参考第三章各种装置功能)
 - M1745: OX X 轴原点复位禁止标志
 - M1825: OX Y 轴原点复位禁止标志
 - M1985: OX Z 轴原点复位禁止标志
 - M1074: OX 运动子程序致能标志

6 运动指令与 G-Code 分类基本使用说明

◆ 相关特殊寄存器说明

D1816	X 轴参数设定
D1846	X 轴运转命令
D1868	设定 OX 程序号码
D1828、D1829	X 轴原点复位速度 V_{RT}
D1830、D1831	X 轴原点复位减速速度 V_{CR}
D1832	X 轴零点信号数 N (一般为设定 Z 相讯号的取样数量)
D1833	X 轴补充距离 P (一般为设定偏移位置量, 以脉冲数为单位距离)
D1836	X 轴加速时间 T_{ACC}
D1837	X 轴减速时间 T_{DEC}
D1896	Y 轴参数设定
D1908、D1909	Y 轴原点复位速度 V_{RT}
D1910、D1911	Y 轴原点复位减速速度 V_{CR}
D1912	Y 轴零点信号数 N (一般为设定 Z 相讯号的取样数量)
D1913	Y 轴补充距离 P (一般为设定偏移位置量, 以脉冲数为单位距离)
D1916	Y 轴加速时间 T_{ACC}
D1917	Y 轴减速时间 T_{DEC}
D1976	Z 轴参数设定
D1988、D1989	Z 轴原点复位速度 V_{RT}
D1990、D1991	Z 轴原点复位减速速度
D1992	Z 轴零点信号数 N (一般为设定 Z 相讯号的取样数量)
D1993	Z 轴补充距离 P (一般为设定偏移位置量, 以脉冲数为单位距离)
D1996	Z 轴加速时间 T_{ACC}
D1997	Z 轴减速时间 T_{DEC}

6 运动指令与 G-Code 分类基本使用说明

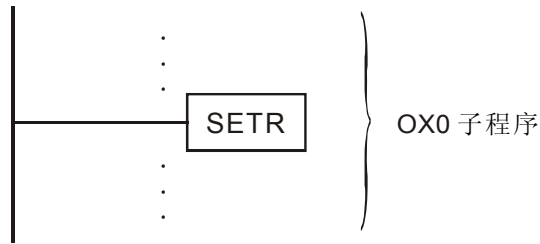
MON 08	SETR	无操作数	电气零点设定	适用机种	
				20PM	10PM
				✓	
无操作数：后面可接 M 码指令					

指令说明

- ◆ 当执行指令 SETR，可将 X 轴、Y 轴与 Z 轴现在位置设定为电气零点，即将现在位置寄存器内容值移动至电气零点寄存器。20D 机种只支持两轴 XY 电气零点设定功能，20M 机种支持三轴 XYZ 电气零点设定功能。

程序范例

- ◆ 程序写法如下所示：



补充说明

- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)

D1848、D1849 X 轴现在位置 CP

D1866、D1867 X 轴电气零点

D1928、D1929 Y 轴现在位置 CP

D1946、D1947 Y 轴电气零点

D2008、D2009 Z 轴现在位置 CP

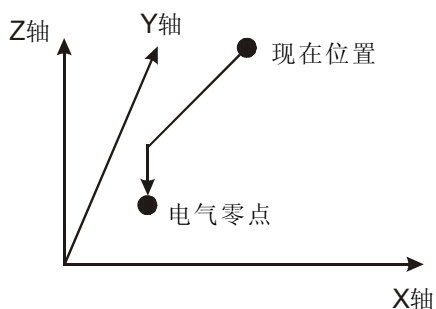
D2026、D2027 Z 轴电气零点

6 运动指令与 G-Code 分类基本使用说明

MON 09	DRVR	无操作数	返回电气零点	适用机种	
				20PM	10PM
				✓	
无操作数：后面可接 M 码指令					

指令说明

- ◆ 当执行指令 DRVR，X、Y、Z 轴将以最高速度 V_{MAX} (0~500kHz) 返回电气零点。
- ◆ 移动路径：



程序范例

- ◆ 程序写法如下所示：



补充说明

- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)

D1822、D1823 X 轴最高速度 V_{MAX}

D1848、D1849 X 轴现在位置 CP

D1866、D1867 X 轴电气零点

D1902、D1903 Y 轴最高速度 V_{MAX}

D1928、D1929 Y 轴现在位置 CP

D1946、D1947 Y 轴电气零点

D1982、D1983 Z 轴最高速度 V_{MAX}

D2008、D2009 Z 轴现在位置 CP

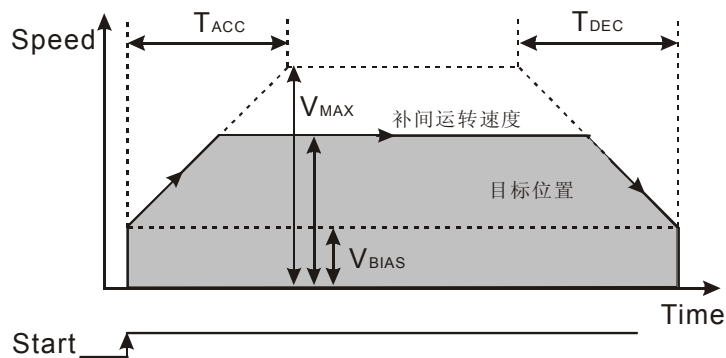
D2026、D2027 Z 轴电气零点

6 运动指令与 G-Code 分类基本使用说明

MON 10	INTR	X (P ₁) Y (P ₂) F (V)	双轴同动单段速补间移动(忽略剩余距离)			适用机种	
						20PM	10PM
						✓	
	字符装置			双字符装置			操作数使用注意：此指令有支持装置 (device)，可经由 V、Z 变址修饰，各装置使用范围请参考功能规格表，后面可接 M 码指令
	K	H	D	KK	HH	DD	
P ₁	*	*	*	*	*	*	
P ₂	*	*	*	*	*	*	
V	*	*	*	*	*	*	

指令说明

- ◆ **P₁**: X 轴目标位置。 **P₂**: Y 轴目标位置。 **V**: 双轴直线补间运转速度。
- ◆ **V**: 运转速度设定最高为 V_{MAX} 。
- ◆ 参数范围: K (-32,768~32,767)、H (0~FFFF)、D (0~9999) 占 16 位, KK (-2,147,483,648 ~ 2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 加减速时间与启动速度可由特 D 来设定。
- ◆ 加减速时间会依最高速度设定等比例增加或减少。
- ◆ 动作图:



- ◆ 补间运转速度可由特殊寄存器 X 轴: D1850~D1851, Y 轴: D1930~D1931 监控。
- ◆ 此指令使用方式, 与 LIN 指令相同 (LIN 可设定 STOP 模式: 考虑剩余距离)
- ◆ 一定要有目标位置, 但不一定要设移动速度, 以此规则, 此指令可有 6 种组合。

NO.	运动指令	操作数参数组合
1	INTR	X (P ₁)
2		X (P ₁) F (V)
3		Y (P ₂)
4		Y (P ₂) F (V)
5		X (P ₁) Y (P ₂)
6		X (P ₁) Y (P ₂) F (V)

补充说明

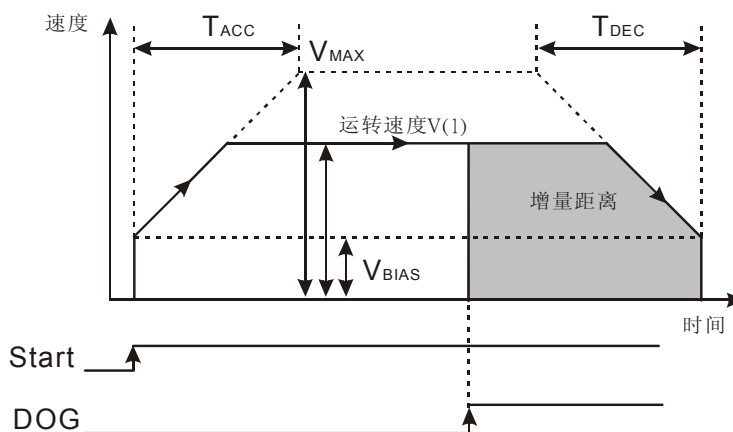
- ◆ 相关特殊寄存器说明: 请参考 LIN 指令。

6 运动指令与 G-Code 分类基本使用说明

MON 11	SINTR	X (P ₁)/Y (P ₂), F (V)	单段速插入	适用机种				
				20PM	10PM			
				✓				
	字符装置			双字符装置			操作数使用注意：此指令有支持装置 (device)，可经由 V、Z 变址修饰，各装置使用范围请参考功能规格表，后面可接 M 码指令	
	K	H	D	KK	HH	DD		
	P ₁	*	*	*	*	*		*
	P ₂	*	*	*	*	*		*
V	*	*	*	*	*	*		

指令说明

- ◆ P₁: X 轴增量距离。 P₂: Y 轴增量距离。 V: 运转速度。
- ◆ V: 运转速度设定最高为 V_{MAX}。
- ◆ SINTR 指令第一个操作数可选择 X 轴或 Y 轴做单段速插入定位。
- ◆ 参数范围：K (-32,768~32,767)、H (0~FFFF)、D (0~9999) 占 16 位，KK (-2,147,483,648 ~ 2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 加减速时间与启动速度可由特 D 来设定。
- ◆ 加减速时间会依最高速度设定等比例增加或减少。
- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。
- ◆ 动作图：

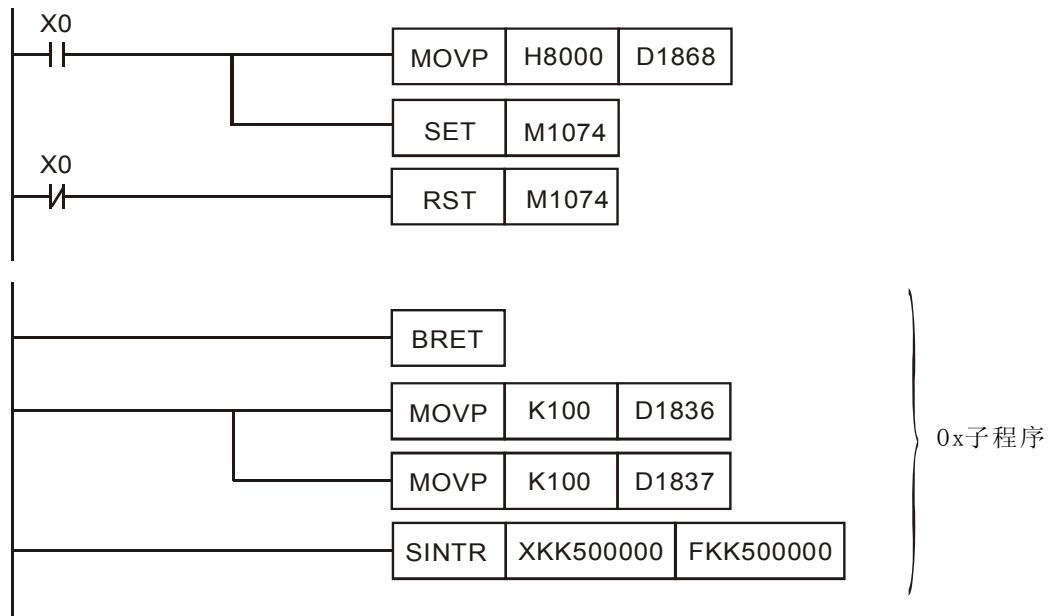


- ◆ 当起动 SINTR 指令，运转速度会由启动速度 V_{BIAS} 开始加速至期望之设定速度 V(I) 稳态运行，当脉冲送出期间遇到 DOG 近点信号触发时，将会以指令设定之增量距离进行运转。
- ◆ 一定要有增量距离及运转速度，以此规则，此指令可有 2 种组合。

NO.	运动指令	操作数参数组合
1	SINTR	X (P ₁) F (V)
2		Y (P ₂) F (V)

程序范例

- ◆ 当 X0=ON, 将执行 OX0 内程序 SINTR 指令, X 轴以 100 ms 加速时间到达单段插入运转速度 500KHZ, 当近点讯号 DOG 触发则以指令设定之增量距离 500,000 个脉冲输出完毕完成定位。
- ◆ 当 X0=OFF, 将关闭 OX0 程序与 SINTR 指令。



补充说明

- ◆ 即使在程序中使用的是绝对坐标方式, 在执行了 SINTR 后, 位移也将被看作是增量地址。
- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)

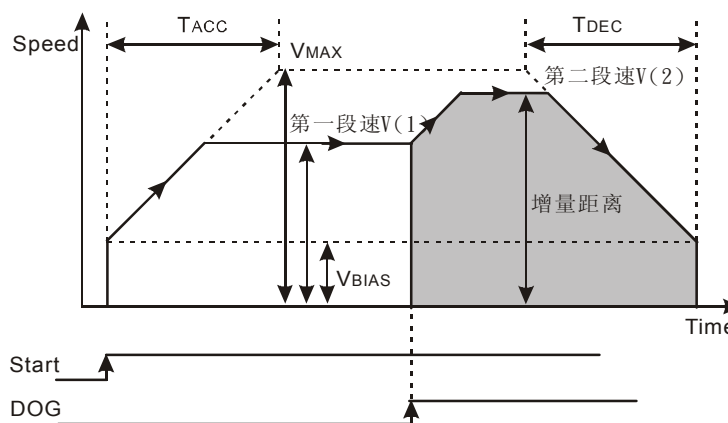
- D1836 X 轴加速时间 T_{ACC}
- D1837 X 轴减速时间 T_{DEC}
- D1848、D1849 X 轴现在位置 CP
- D1916 Y 轴加速时间 T_{ACC}
- D1917 Y 轴减速时间 T_{DEC}
- D1928、D1929 Y 轴现在位置 CP

6 运动指令与 G-Code 分类基本使用说明

MON 12	DINTR	X(P ₁)/Y(P ₂), F(V ₁) F(V ₂)	两段速插入指令			适用机种	
						20PM	10PM
						✓	
	字符装置			双字符装置			操作数使用注意：此指令有支持装置（device），可经由 V、Z 变址修饰，各装置使用范围请参考功能规格表，后面可接 M 码指令
	K	H	D	KK	HH	DD	
P ₁	*	*	*	*	*	*	
P ₂	*	*	*	*	*	*	
V ₁	*	*	*	*	*	*	
V ₂	*	*	*	*	*	*	

指令说明

- ◆ **P₁**: X 轴增量距离。 **P₂**: Y 轴增量距离。 **V₁**: 第一段运转速度设定。 **V₂**: 第二段运转速度设定。
- ◆ **V₁**、**V₂**: 运转速度设定最高为 V_{MAX} 。
- ◆ DINTR 指令第一个操作数，可选择 X 轴或 Y 轴做两段速插入定位。
- ◆ 参数范围：K (-32,768~32,767)、H (0~FFFF)、D (0~9999) 占 16 位，KK (-2,147,483,648~2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 加减速时间与启动速度可由特 D 来设定。
- ◆ 加减速时间会依最高速度设定等比例增加或减少。
- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。
- ◆ 动作图：

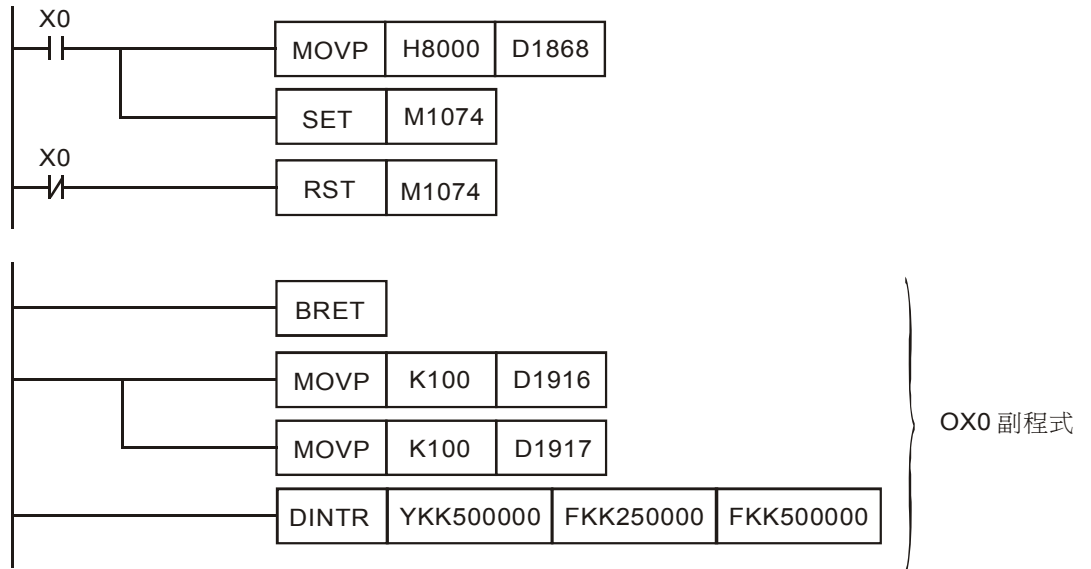


- ◆ 当启动 DINTR 指令, 运转速度会由启动速度 V_{BIAS} 开始加速到达第一段速设定 $V(1)$ 稳态运行, 当 DOG 近点信号触发时, 将加速至第二段速度 $V(2)$ 并以设定之增量距离运转。
- ◆ 一定要有增量距离及运转速度, 以此规则, 此指令可有 2 种组合。

NO.	运动指令	操作数参数组合
1	DINTR	X(P ₁) F(V ₁) F(V ₂)
2		Y(P ₂) F(V ₁) F(V ₂)

程序范例

- ◆ 当 X0=On, 将执行 OX0 内程序 DINTR 指令, Y 轴以 100ms 加速时间到达第一段速运转速度 250kHz 稳定运转, 当近点讯号 DOG 触发, 则加速至第二段速运转速度 500kHz 并以增量距离 500,000 个脉冲输出完毕, 完成定位。
- ◆ X0=Off, 将关闭 OX0 程序与 DINTR 指令。



补充说明

- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)

D1836 X 轴加速时间 T_{ACC}
 D1837 X 轴减速时间 T_{DEC}
 D1848、D1849 X 轴现在位置 CP
 D1916 Y 轴加速时间 T_{ACC}
 D1917 Y 轴减速时间 T_{DEC}
 D1928、D1929 Y 轴现在位置 CP

6 运动指令与 G-Code 分类基本使用说明

MON 13	MOVC	X(L ₁) Y(L ₂)	设定直线位移补偿	适用机种			
				20PM	10PM		
				✓			
	字符装置			双字符装置			操作数使用注意：此指令有支持装置 (device)，可经由 V、Z 变址修饰，各装置使用范围请参考功能规格表，后面可接 M 码指令
	K	H	D	KK	HH	DD	
	L ₁	*	*	*	*	*	
L ₂	*	*	*	*	*	*	

指令说明

- ◆ L₁: X 轴补偿值。 L₂: Y 轴补偿值。
- ◆ 可选择仅设定 X 轴之补偿值，例如：MOVC XDD0。
- ◆ 执行 MOVC 指令将会把指令设定之补偿值自动写入特殊寄存器
X 轴：D1708~D1709，Y 轴：D1724~D1725。
- ◆ 直线位移补偿可用于 DRV、LIN、INTR 指令。
- ◆ 当补偿值写入补偿寄存器，再执行直线位移指令，会依设定之补偿值进行补偿动作。
- ◆ 参数范围：K (-32,768~32,767)、H (0~FFFF)、D (0~9999) 占 16 位，KK (-2,147,483,648~2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。

NO.	运动指令	操作数参数组合
1	MOVC	X(L ₁)
2		X(L ₁) Y(L ₂)

补充说明

- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)
D1708、D1709 X 轴行走距离补偿值
D1724、D1725 Y 轴行走距离补偿值

6 运动指令与 G-Code 分类基本使用说明

MON 14	CNTC	I(L ₁) J(L ₂)		圆弧圆心补偿			适用机种	
							20PM	10PM
							✓	
	字符装置			双字符装置			操作数使用注意：此指令有支持装置（device），可经由 V、Z 变址修饰，各装置使用范围请参考功能规格表，后面可接 M 码指令	
	K	H	D	KK	HH	DD		
	L ₁	*	*	*	*	*		
L ₂	*	*	*	*	*	*		

指令说明

- ◆ L₁： X 轴圆心向量补偿值。 L₂： Y 轴圆心向量补偿值。
- ◆ 执行 CNTC 指令，会把指令设定之补偿值自动写入特殊寄存器 X 轴： D1710~D1711， Y 轴： D1726~D1727。
- ◆ 圆弧圆心补偿可用于 CW、CCW 指令。
- ◆ 当补偿值写入补偿寄存器，再执行圆弧指令，会依设定之补偿值进行补偿动作。
- ◆ 参数范围： K（-32,768~32,767）、H（0~FFFF）、D（0~9999）占 16 位，KK（-2,147,483,648~2,147,483,647）、HH（0~FFFFFFFF）、DD（0~9998）占 32 位。
- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。

补充说明

- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)
D1710、D1711 X 轴中心点补偿值
D1726、D1727 Y 轴中心点补偿值

6 运动指令与 G-Code 分类基本使用说明

MON 15	RADC	R(L)			圆弧半径补偿			适用机种	
								20PM	10PM
								✓	
L	字符装置			双字符装置			操作数使用注意：此指令有支持装置（device），可经由 V、Z 变址修饰，各装置使用范围请参考功能规格表，后面可接 M 码指令		
	K	H	D	KK	HH	DD			
	*	*	*	*	*	*			

指令说明

- ◆ L: 圆弧半径向量补偿值。
- ◆ 执行 RADC 指令，会把指令设定之补偿值自动写入特殊寄存器 D1712~D1713。
- ◆ 圆弧半径补偿可用于 CW、CCW 指令。
- ◆ 当补偿值写入补偿寄存器，再执行圆弧指令，会依设定之补偿值进行补偿动作。
- ◆ 参数范围：K（-32,768~32,767）、H（0~FFFF）、D（0~9999）占 16 位，KK（-2,147,483,648~2,147,483,647）、HH（0~FFFFFFFF）、DD（0~9998）占 32 位。
- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。

补充说明

- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)
D1712、D1713 圆弧半径补偿值

6 运动指令与 G-Code 分类基本使用说明

MON	CANC	无操作数	取消运动补偿	适用机种	
16				20PM	10PM
				✓	
无操作数：后面可接 M 码指令					

指令说明

- ◆ 执行 CANC 指令，将取消所有运动补偿值，即自动清除特殊寄存器 D1708~D1709、D1724~D1725、D1710~D1711、D1726~D1727、D1712~D1713。

补充说明

- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)

D1708、D1709 X 轴行走距离补偿值

D1724、D1725 Y 轴行走距离补偿值

D1710、D1711 X 轴中心点补偿值

D1726、D1727 Y 轴中心点补偿值

D1712、D1713 圆弧半径补偿值

6 运动指令与 G-Code 分类基本使用说明

MON 17	ABST	无操作数	绝对坐标系统设定	适用机种	
				20PM	10PM
				✓	
MON 18	INCT	无操作数	相对坐标系统设定	适用机种	
				20PM	10PM
				✓	

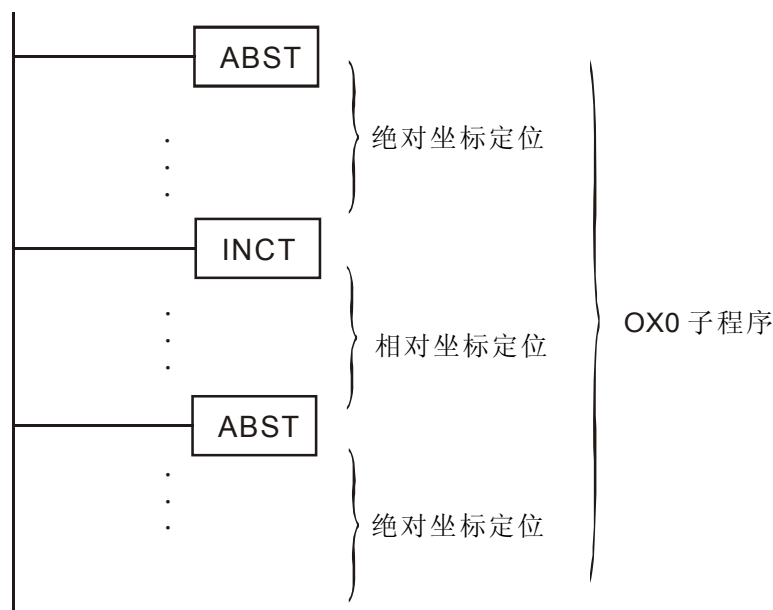
无操作数：后面可接 M 码指令

指令说明

- ◆ 执行指令 **ABST** 绝对坐标：代表从 0 开始之位置，当目标位置大于现在位置时，马达正转，小于现在位置时，马达反转。
- ◆ 执行指令 **INCT** 相对坐标：代表从现在位置开始算马达行走之距离，当相对坐标为正值时，马达正转，相对坐标为负值时，马达反转。
- ◆ 圆弧圆心坐标 (I, J)、半径 (R) 以及由 **SINTR** 和 **DINTR** 指令设定增量距离，都是被看成增量值。

程序范例

- ◆ 当 PM 由 **MANU** 切换到 **AUTO**，若程序中无指定 **ABST** 或 **INCT** 时，则初始设定为 **ABST** 绝对坐标系统，当 **INCT** 执行后，其下一行之后的运动指令（如：**DRV**、**LIN**、**CW**、**CCW**），将以相对坐标系统运作，程序写法如下所示：



6 运动指令与 G-Code 分类基本使用说明

MON 19	SETT	X (P ₁) Y (P ₂) Z (P ₃)			设定现在位置			适用机种		
								20PM	10PM	
								✓		
	字符装置			双字符装置			操作数使用注意：此指令有支持装置（device），可由 V、Z 变址修饰，各装置使用范围请参考功能规格表，后面可接 M 码指令			
	K	H	D	KK	HH	DD				
	P ₁	*	*	*	*	*				*
	P ₂	*	*	*	*	*				*
P ₃	*	*	*	*	*	*				

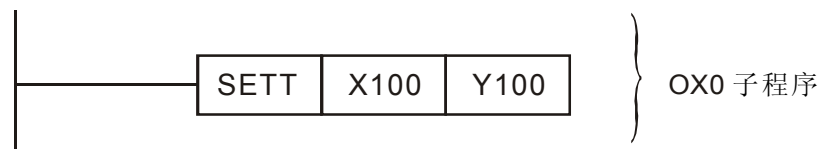
指令说明

- ◆ **P₁**: X 轴现在位置设定。 **P₂**: Y 轴现在位置设定。 **P₃**: Z 轴现在位置设定。
- ◆ 执行 SETT 指令, 会把指令设定之现在位置自动写入特殊寄存器 X 轴: D1848~D1849、Y 轴: D1928~D1929、Z 轴: D2008~D2009。
- ◆ 参数范围: K (-32,768~32,767)、H (0~FFFF)、D (0~9999) 占 16 位, KK (-2,147,483,648~2,147,483,647)、HH (0~FFFFFFFF)、DD (0~9998) 占 32 位。
- ◆ 指令中的参数装置 16 位与 32 位可以混合使用。
- ◆ 此指令执行时, 目前值寄存器中的值改成由该指令指定的值, 因此机械零点和电气零点都会被改变。
- ◆ 编写指令方式: 一定要有一轴目标位置, 以此规则, 此指令可有 7 种组合。

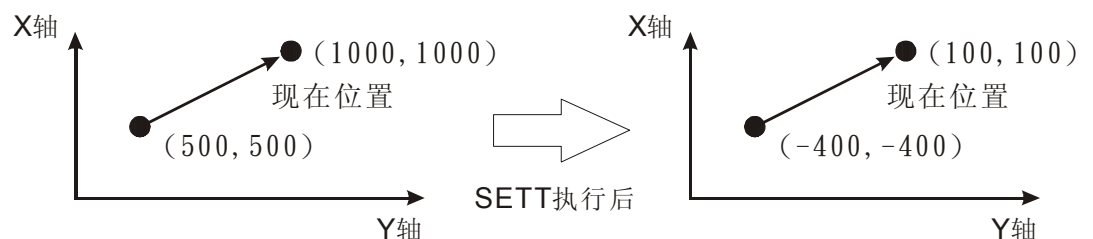
NO.	运动指令	操作数参数组合	20D	20M
1	SETT	X (P ₁)	✓	✓
2		Y (P ₂)	✓	✓
3		X (P ₁) Y (P ₂)	✓	✓
4		Z (P ₃)	-	✓
5		X (P ₁) Z (P ₃)	-	✓
6		Y (P ₂) Z (P ₃)	-	✓
7		X (P ₁) Y (P ₂) Z (P ₃)	-	✓

程序范例

- ◆ 程序写法如下所示:



- ◆ SETT 执行后, 现在位置改变, 其余相关之相对位置也随之改变, 如下所示。



6 运动指令与 G-Code 分类基本使用说明

补充说明

- ◆ 相关特殊寄存器说明(各项寄存器说明容请参考第三章各种装置功能)

D1848、D1849 X 轴现在位置 CP

D1928、D1929 Y 轴现在位置 CP

D2008、D2009 Z 轴现在位置 CP

6.4 O 指针 / M 码指令

指令	功能	适用机种	
O	程序指针	20PM	10PM
		✓	✓
说明	主程序指针: O100 运动子程序指针: OX0 ~ OX99		

指令说明

- ◆ O100 为一般主程序起始指针，须由主程序来启动 OX0~OX99 运动子程序运转，而执行 M102 指令表示 O100 主程序结束。
- ◆ OX0~OX99 为 100 个运动子程序指针，可由使用者任选编号对不同运动路径进行设计，由 D1868 (CR72) 寄存器储存子程序编号，其中 D1868 的 bit14 或 bit15 位要为 1，最后再 D1846 bit12=On 或 M1074=On 控制启动，下面范例说明。

范例：要启动 OX99 运动子程序两个步骤：

- (1) 设定启动编号：D1868=H8063 (或 H4063、HC063)。
- (2) 启动 OX99：D1846=H1000 或 M1074=On。

- ◆ 指针 OX0~OX99 的运动子程序结束指令都是 M2；

程序范例

- ◆ 下列为两组程序范例，N0000~N0100 为 O100 主程序，N0102~N0304 为编号 OX50 运动子程序：

行号	程序
N0000	O100
N0001	LD M1000
N0002	MOV H8063 D1868
N0007	OUT M1074
:	:
:	:
N0099	OUT Y30
N0100	M102
N0101	NOP
N0102	OX50
N0103	DRVZ
N0104	ABS
:	:
:	:
N0304	M2

6 运动指令与 G-Code 分类基本使用说明

指令	功能	适用机种	
M	M 码指令	20PM	10PM
		✓	
操作数	M0 ~ M65535 M102: O100 主程序结束指令, M2: OX0~OX99 运动子程序结束指令		

指令说明

- ◆ M 码指令用于运动指令中, 执行 M 码时, 首先将 M 码编号存进 D1703, 当 M 码动作时 M1794 状态自动为 0N; 若 M1744 设为 On, 则 M1794 将变为 Off 状态, 表示 M 码执行完毕。
- ◆ 执行 M 码可控制 Y 输出, 将 D1873 high byte 设为 1, 表示开启输出致能, Low Byte 设定 Y 输出起始编号, 当 M1794 状态变为 On (即 M 码执行), D1873 所对应设定的 Y 编号输出为 On, 当 M1794 状态变为 Off, Y 就变 Off, 请参考程序范例 (一) 的说明。
- ◆ M 码指令一般用在 OX00~OX99 子程序区段中。
- ◆ M 码指令可分为两种模式: 一种是 after 模式, 一种是 with 模式。这两种模式的差别为启动 M 码指令的时序不同。请参考程序范例 (二) 的说明。
- ◆ M 码执行结束使 M1794 状态由 On 变为 Off, 其中有二种方式:
 - (1) 直接把 M1794 设为零, 即可执行重置动作。
 - (2) 直接把 M1744 设为 On 状态。

补充说明

- ◆ M 码两个特殊指定用法, 就是 O100 主程序指定结束指令 M102, 及 OX0~OX99 的运动程序指定结束指令 M2。因此应用 M 码时, 请避免使用到 M2 及 M102。

程序范例 (一)

- ◆ 当执行 M6 时, 欲由 Y 装置来显示 M 码目前执行编号, 设计流程如下:
 - (1) 首先设定 D1873 的参数值如下:

```
MOV H0□□□ D1873
      |
      |→ N:Y输出的起始编号
      |
      |→ 0:禁止
      |   1:允许
```

- (2) 执行 M 码 (M6), 此时 DVP-PM 自动将 K6(二进制 0110)写入 D1703 中, 并将 D1703 的值写入 K2Y_N, 其中 N 为 Y 输出起始编号。动作如下:

```
MOV H6 D1703

MOV D1703 K2YN
```

上述两行程序为 M 码启动时, 将自动运作, 故不需使用者编写。

(3) 当 D1873 的 N 设定如下表时，K2Y_N 的输出状态，如下表所示：

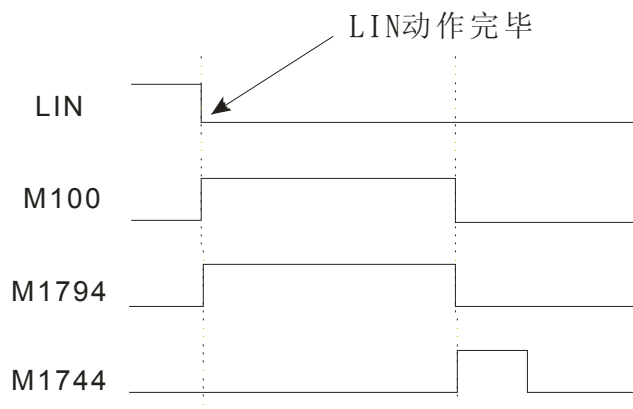
D1873	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
H00□□	Y 输出无动作							
H0100	0	0	0	0	0	1	1	0
H0101	0	0	0	0	1	1	0	-
H0102	0	0	0	1	1	0	-	-
H0103	0	0	1	1	0	-	-	-
.	.							
.	.							
.	.							

程序范例
(二)

◆ after 模式：单独一行程序中，只有 M 码指令。如下所示：



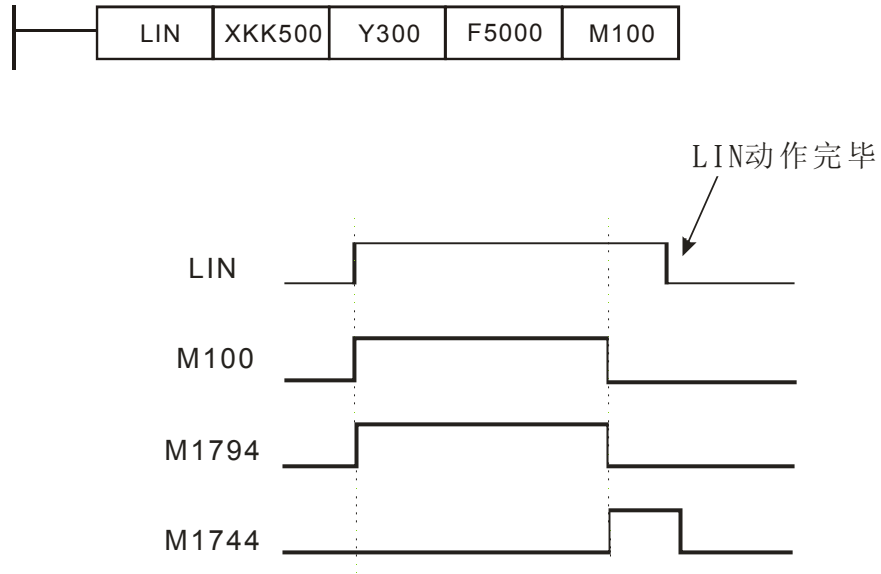
动作时序如下图：



当 LIN 指令动作完毕后，M 码 (编号 M100) 将自动启动，此时 M1794 也自动为 On；欲停止 M100 时，将 M1744 On 即可结束 M100 之动作。如果要再次启动 M 码，只须重新设定好，由程序中规划再次启动 M 码即可。

6 运动指令与 G-Code 分类基本使用说明

- ◆ with 模式：M 码指令摆放于运动指令尾端。如下所示：



当 LIN 指令触发时，同时 M 码 (编号 M100) 会自动启动，此时 M1794 自动为 On；欲停止 M100 时，将 M1744 On 即可结束 M100 之动作。如果要再启动 M 码，则须等待 LIN 指令动作完成后，重新设定参数，即可再由程序设计触发 M 码动作。

程序范例
(三)

- ◆ 下列程序范例，其中行号 N0100 和 N0304 为 M 码特殊指令，N0105 为 M 码 with 模式使用方式，最后行号 N0250 为 M 码 after 模式。

行号	程序
N0000	O100
N0001	LD M1000
:	:
:	:
N0099	OUT Y30
N0100	M102
N0101	NOP
N0102	OX50
N0103	DRVZ
N0104	ABS
N0105	DRV XD10 FXD12 M20
:	:
N0250	M08
:	:
N0304	M2

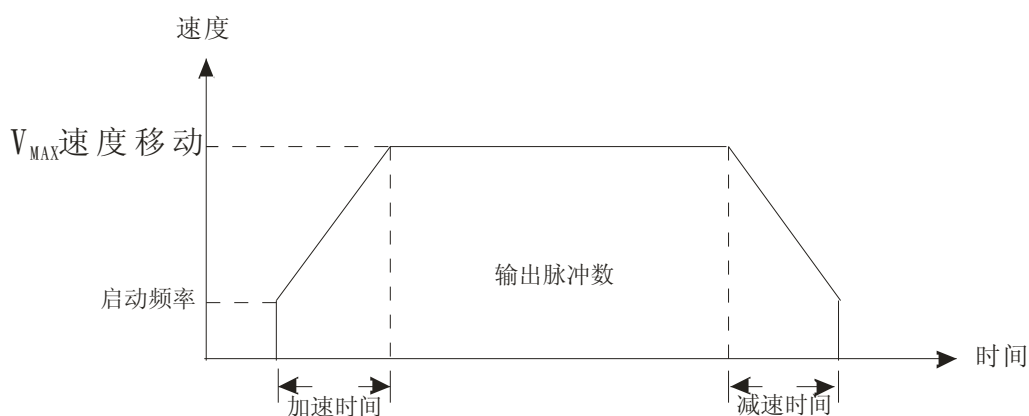
6 运动指令与 G-Code 分类基本使用说明

6.5 G-Code 指令

G 码	G00	X (P₁) Y (P₂) Z (P₃)	三轴高速定位	适用机种	
				20PM	10PM
				✓	

指令说明

- ◆ **P₁**: X 轴目标位置。 **P₂**: Y 轴目标位置。 **P₃**: Z 轴目标位置。
- ◆ 使用 20D 机种时, Z 轴目标位置(**P₃**)是由内建第三轴控制, 请参考 6-51 页说明。
- ◆ 无输入小数点, 参数设定范围: (-2,147,483,648~2,147,483,647); 有输入小数点, 参数设定范围: (-2,147,483.648~2,147,483.647)。
- ◆ 20M 机种操作数参数除上述形式设定, 还可输入 16 位寄存器 (D) 与 32 位寄存器 (DD)。
- ◆ 特殊寄存器说明请参考运动指令 MON 00 DRV 指令补充说明。
- ◆ 执行 G00 指令, 移动速度固定以最快速度 V_{MAX} 移动。
- ◆ 位置设定有延续功能, 参考补充说明。
- ◆ 加减速时间与启动速度可由特 D 来设定。
- ◆ 加减速时间与启动速度会受特殊寄存器“最高速度之设定”的影响, 而呈等比例增加或减少。
- ◆ 运转动作图:



- ◆ 20D 机种无三轴高速定位功能, 需外挂高速输出模块 DVP-01PU, 将 X-Y 轴设计为 2 轴高速定位, 而 Z 轴为独立高速移动。为机械运作安全, 当 G00 指令执行时, 会先执行 Z 轴高速移动, 再执行 X-Y 轴高速定位。因此 PM 执行 G00 指令中, 若含有 X-Z、Y-Z、X-Y-Z 等三种组合时, 会自动分解成如下:

G00 ZP₃ (A)

G00 XP₁ YP₂ (B)

上述 (A) (B) 两行指令程序说明, 请参考后续之详细说明!

补充说明

- ◆ 位置设定有延续功能, 例如:

```
G00 X500.0 Y500.0 Z500.0;
```

```
X1000.0 Y1000.0 Z1000.0;
```

当执行完第一行程序 G00 指令, 然后再执行第二行程序, 第二行会自动以第一行指令 G00 去到达目标位置。

- ◆ 使用 20D 机种, 当 G 码程序中 G00 使用到 Z 轴目标位置 (内建第三轴控制) 时, 范例程序如下:

```
G00 X1000 Y1000 Z100;
```

PM 于程序编译后, 将同化如下:

```
G00 Z100; (A)
```

```
G00 X1000 Y1000; (B)
```

首先执行步骤 (A), 此时 Z 轴快速移动到 K100 的位置, 再执行步骤 (B), 以最快速度移动到 (1000,1000) 的目标位置。

6 运动指令与 G-Code 分类基本使用说明

G 码	G00	Z (P ₃)	三轴高速定位 (第三轴控制)	适用机种	
				20PM	10PM
				✓	

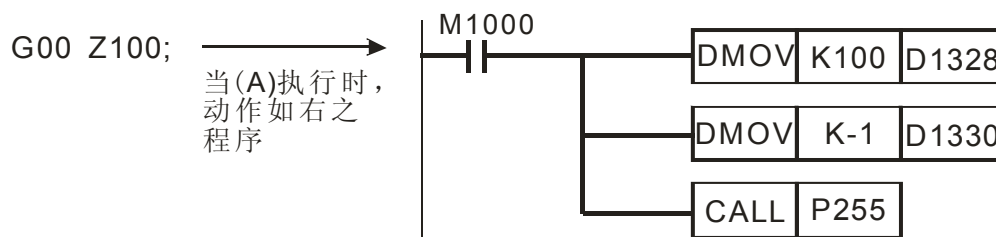
补充说明

- ◆ 20D 机种无三轴高速定位功能，需外挂高速输出模块 DVP-01PU 实现第三轴控制。
- ◆ 使用 20D 机种时，当 G 码程序中 G00 使用到 Z 轴目标位置（内建第三轴控制）时，范例程序如下：

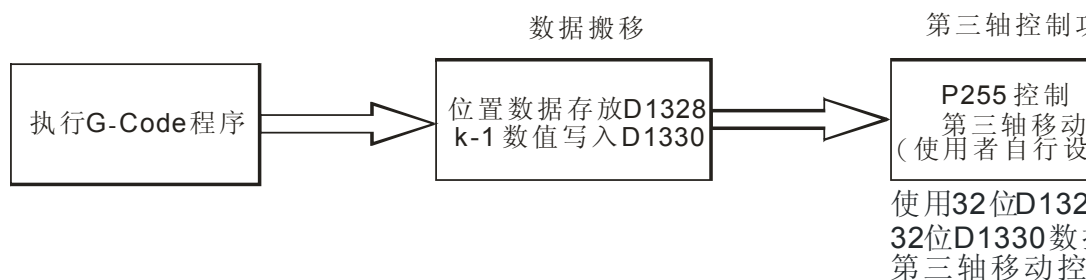
G00 Z100; (A)

首先执行上列程序 (A)，此时 PM 会将 Z 轴的目标位置值 K100 写入 32 位 D1328；而 G00 移动速度是内定参数控制，因此将 K-1 数值写入 32 位 D1330 (K-1 作为判断条件用，K-1 表示无数据设定，系统会以最高速度来运作)，之后调用 P255 之子程序，执行 P255 子程序。

步骤 (A) 的动作说明如下：



其中 P255 子程序，是使用者使用 32 位 D1328 与 32 位 D1330 控制数据，进一步编写程控第三轴（如：提笔、夹放...等功能）。



因此使用 Z 轴移动控制时，请避免重复使用到 D1328~D1331、P255 等装置。

G 码	G00	X (P ₁) Y (P ₂)	二轴高速定位	适用机种	
				20PM	10PM
				✓	

指令说明

- ◆ 主要用于 20D 机种，其定位动作内容请参考运动指令 MON 00 DRV 指令说明。

G 码	G01	X (P₁) Y (P₂) Z (P₃) F (V)	三轴直线补间移动 (可考虑剩余距离)		适用机种	
					20PM	10PM
					✓	

指令说明

- ◆ **P₁**: X 轴目标位置。 **P₂**: Y 轴目标位置。 **P₃**: Z 轴目标位置。 **V**: 参轴直线补间运转速度。
- ◆ 使用 20D 机种时, Z 轴目标位置(**P₃**)是由内建第三轴控制, 请见 6-53 页说明。
- ◆ 无输入小数点, **P₁**、**P₂**、**P₃** 参数设定范围: (-2,147,483,648~2,147,483,647), **V** 参数设定范围: (0~500,000); 有输入小数点, **P₁**、**P₂**、**P₃** 参数设定范围: (-2,147,483.648~2,147,483.647), **V** 参数设定范围: (0~500.0)。
- ◆ 20M 机种操作数参数除上述形式, 还可输入 16 位 (D) 与 32 位寄存器 (DD)。
- ◆ 运转速度有延续功能, 参考补充说明。
- ◆ 定位动作内容请参考运动指令 MON 01 LIN 指令中 20M 机种。
- ◆ 20D 机种无三轴同动移动功能, 需配合外挂高速输出模块 DVP-01PU, 将 X-Y 轴设计为双轴补间移动, 而 Z 轴为独立轴移动。为机械运作安全, 当 G01 指令执行时, 先执行 Z 轴移动, 再执行 X-Y 轴补间运动功能。因此 PM 执行 G01 指令时, 若含有 X-Z、Y-Z、X-Y-Z 等三种组合时, 会自动分解成如下:

G01 ZP₃ FV (A)

G01 XP₁ YP₂ FV (B)

上述 (A) (B) 两行指令程序说明, 请参考 6-53 页之详细说明!

补充说明

- ◆ 运转速度有延续功能, 例如:
G01 X100 Y100 Z400 F200;
X200 Y200;
 当执行完第一行程序 G1 指令, 然后再执行第二行程序, 第二行会自动以第一行指令 G01 指令 F200 之速度到达目标位置。
- ◆ 使用 20D 机种时, 当 G 码程序中 G01 使用到 Z 轴目标位置 (内建第三轴控制) 时, 范例程序如下:

G01 X1000 Y1000 Z100 F200;

PM 于程序编译后, 将同化如下:

G01 Z100 F200;... (A)

G01 X1000 Y1000 F200; ... (B)

首先执行步骤 (A), 此时 Z 轴以 K200 速度移动到目标位置 K100, 再执行步骤 (B), 以运转速度 K200 移动到 (1000,1000) 的目标位置。

6 运动指令与 G-Code 分类基本使用说明

G 码	G01	Z (P₃) F (V)	三轴直线补间移动 (第三轴控制)	适用机种	
				20PM	10PM
				✓	

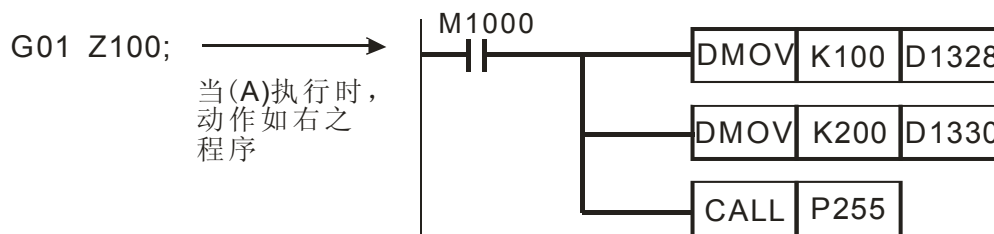
补充说明

- ◆ 20D 机种无三轴同动移动功能，需外挂高速输出模块 DVP-01PU 实现第三轴控制。
- ◆ 使用 20D 机种时，当 G 码程序中 G01 使用到 Z 轴目标位置（内建第三轴控制）时，范例程序如下：

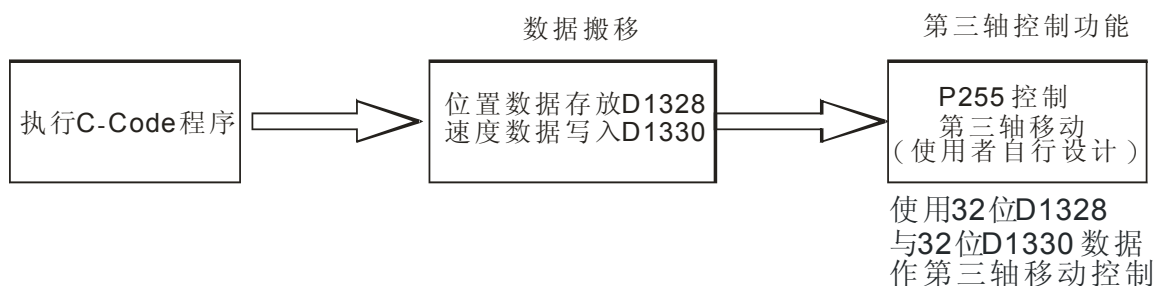
G01 Z100 F200;... (A)

首先执行上列程序 (A)，此时 PM 会将 Z 轴的目标位置值 K100 写入 32 位 D1328；将双轴直线补间运转速度 K200 数值写入 32 位 D1330，并且调用 P255 之子程序且执行后，开始执行。

步骤 (A) 的动作说明如下：



其中 P255 子程序，是使用者使用 32 位 D1328 与 32 位 D1330 控制数据，编写程控第三轴动作（如：提笔、夹放...等功能）。



因此使用 Z 轴移动控制时，请避免重复使用到 D1328~D1331、P255 等装置。

G 码	G01	X (P₁) Y (P₂) F (V)	双轴直线补间移动 (可考虑剩余距离)	适用机种	
				20PM	10PM
				✓	✓

补充说明

- ◆ 主要用于 20D 机种，其定位动作内容请参考运动指令 MON 01 LIN 指令。

G 码	G02	X (P ₁) Y (P ₂) Z (P ₃)	顺时针圆弧移动/顺时针螺旋 移动/逆时针圆弧移动/逆时针 螺旋移动(设定圆心位置)	适用机种	
	G03	I (P ₄) J (P ₅) K (P ₆) F (V)		20PM	10PM
				✓	

补充说明

- ◆ **P₁**: X轴目标位置。 **P₂**: Y轴目标位置。 **P₃**: Z轴目标位置。 **P₄**: 圆弧X轴圆心位置。 **P₅**: 圆弧Y轴圆心位置。 **P₆**: 圆弧Z轴圆心位置。 **V**: 圆弧/螺旋补间运转速度。
- ◆ **P₄**、**P₅**和**P₆**: 以圆弧或螺旋圆心为基准点, 相对于目前位置向量值。
- ◆ 无输入小数点, **P₁**、**P₂**、**P₃**、**P₄**、**P₅**、**P₆** 参数设定范围: (-2,147,483,648~2,147,483,647), **V** 参数设定范围: (0~500,000); 有输入小数点, **P₁**、**P₂**、**P₃**、**P₄**、**P₅**、**P₆** 参数设定范围: (-2,147,483.648~2,147,483.647), **V** 参数设定范围: (0~500.0)。
- ◆ 20M 机种操作数参数除上述形式设定, 还可输入 16 位寄存器 (D) 与 32 位寄存器 (DD)。
- ◆ 运转速度有延续功能, 参考补充说明。
- ◆ 定位动作内容请参考运动指令 MON 02 CW、MON 03 CCW 指令。

补充说明

- ◆ 运转速度有延续功能, 例如:

G02 X0.0 Y100.0 I0.0 J50.0 F100.0;

G02 X0.0 Y0.0 I0.0 J50.0;

- ◆ 当执行完第一行程序 G02 指令, 然后再执行第二行程序, 第二行会自动以第一行指令 G02 指令 F100.0 之速度到达目标位置。

6 运动指令与 G-Code 分类基本使用说明

G 码	G02 G03	$X(P_1) \quad Y(P_2) \quad Z(P_3)$ $R(L) \quad F(V)$	顺时针圆弧移动/顺时针螺旋 移动/逆时针圆弧移动/逆时针 螺旋移动(设定半径长度)	适用机种	
				20PM	10PM
				✓	

指令说明

- ◆ **P₁**: 圆弧 X 轴目标位置。 **P₂**: 圆弧 Y 轴目标位置。 **P₃**: Z 轴目标位置。 **L**: 圆弧半径(弧角小于 180 度, R 为"+", 弧角等于大于 180 度, R 为"-")。 **V**: 圆弧移动到目标位置速度。
- ◆ 无输入小数点, **P₁**、**P₂**、**P₃**、**R** 参数设定范围: (-2,147,483,648~2,147,483,647), **V** 参数设定范围: (0~500,000); 有输入小数点, **P₁**、**P₂**、**P₃**、**R** 参数设定范围: (-2,147,483.648~2,147,483.647), **V** 参数设定范围: (0~500.0)。
- ◆ 20M 机种操作数参数除上述形式设定, 还可输入 16 位寄存器 (D) 与 32 位寄存器 (DD)。
- ◆ 定位动作内容请参考运动指令 MON 04 CW、MON 05 CCW 指令。

G 码	G04	$X(T) / P(T)$	停顿时间	适用机种	
				20PM	10PM
				✓	

指令说明

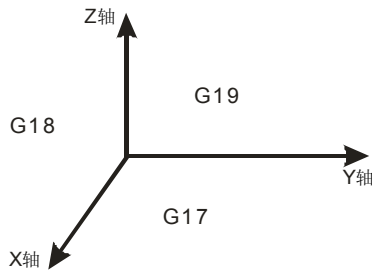
- ◆ **XT**: 设定停顿时间, 单位为 1 秒。G4 X1 表示停顿 1 秒, G4 X2.5 表示停顿 2.5 秒。
- ◆ **PT**: 设定停顿时间, 单位为 1 毫秒。G4 P100 表示停顿 0.1 秒, G4 P4500 表示停顿 4.5 秒, 设定值以 10ms 为基数, 若设定值小于 10ms 时, 将以 0ms 为设定值, 以此类推当设定值若为 23ms 时, 则以 20ms 为设定值。
- ◆ 20M 机种操作数参数除上述形式设定, 还可输入 16 位寄存器 (D) 与 32 位寄存器 (DD)。
- ◆ 动作内容, 请参考运动指令 MON 06 TIM 指令。

G 码	G17	无操作数	选择 XY 平面	适用机种	
	G18		选择 XZ 平面	20PM	10PM
	G19		选择 YZ 平面	✓	

指令说明

- ◆ 此三个指令动作内容, 用于决定圆弧补间和螺旋补间作平面选择, 对于直线补间无影响。
- ◆ 程序执行时, 此三个工作平面可相互切换。程序中若无设定任何平面选择, 系统初始状态为 XY 平面(G17)状态。
- ◆ 下图所示三种平面

6 运动指令与 G-Code 分类基本使用说明



G 码	G90	无操作数	绝对坐标系统设定	适用机种	
				20PM	10PM
				✓	

指令说明

- ◆ 指令动作内容，请参考运动指令 MON 17 ABS 指令。

G 码	G91	无操作数	相对坐标系统设定	适用机种	
				20PM	10PM
				✓	

指令说明

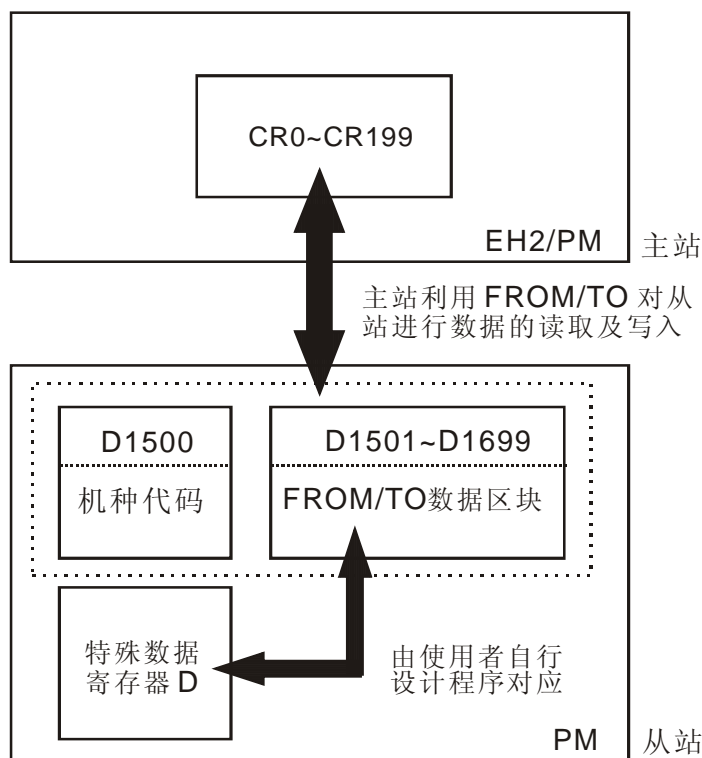
- ◆ 指令动作内容，请参考运动指令 MON 18 INC 指令。

7.1 DVP-EH2、DVP-PM 主站（Master）与 DVP-PM 从站（Slave）联机控制方式

DVP-PM 机种，在内部有规划一个特殊寄存器区块，可相对应于主站（Master）的控制寄存器（Controlled Register），使用者可利用此架构，自行规划从站（Slave）欲受主站（Master）控制的数据传递，以达到主站（Master）与从站（Slave）的数据交换及运动模式控制的功能。

7.1.1 架构说明

- ◆ DVP-EH2、DVP-PM 主站利用 FROM/TO 指令下达命令数据于 DVP-PM 从站，进一步驱动 DVP-PM 从站，执行各项动作模式。
- ◆ DVP-EH2、DVP-PM 主站使用 FROM/TO 指令，可擦写 DVP-PM 从站的 CR（Controlled Register）范围为 CR#0~CR#199，对应从站内部规划的特殊寄存器区（范围：D1500~D1699）。



7.1.2 主站与从站联机控制范例说明

- ◆ 设定方式：分别规划主站与从站程序
 - DVP-PM 从站：规划欲受主站控制的数据，利用 MOV 指令，将数据移动至 DVP-PM 内部规划的特殊寄存器区块。
 - 主站：规划欲控制的从站控制寄存器（Controlled Register）。

◆ 范例一：

【控制要求】

7 DVP-PM 从机使用说明

- DVP-EH2 主站下 FROM/TO 命令，对应从站 DVP-PM 内部特殊寄存器(D1500~D1699)，控制 X、Y 轴，执行多种手动运动模式(详细设定参考 3.12.3)。

【主站控制寄存器（Controlled Register）与从站特殊寄存器（Special Register）对应表】

主站	从站		寄存器说明
	内部自动规划	使用者自行规划	
CR#0	D1500	系统内定	DVP-PM 从机机种代号
CR#1	D1501	D1846	X 轴运转命令
CR#2~3	D1502~D1503	D1848~D1849	X 轴现在位置 CP (PLS)
CR#4~5	D1504~D1505	D1850~D1851	X 轴现在速度 (PPS)
CR#6~7	D1506~D1507	D1860~D1861	X 轴手摇轮输入频率
CR#8~9	D1508~D1509	D1862~D1863	X 轴累计手摇轮输入脉波个数

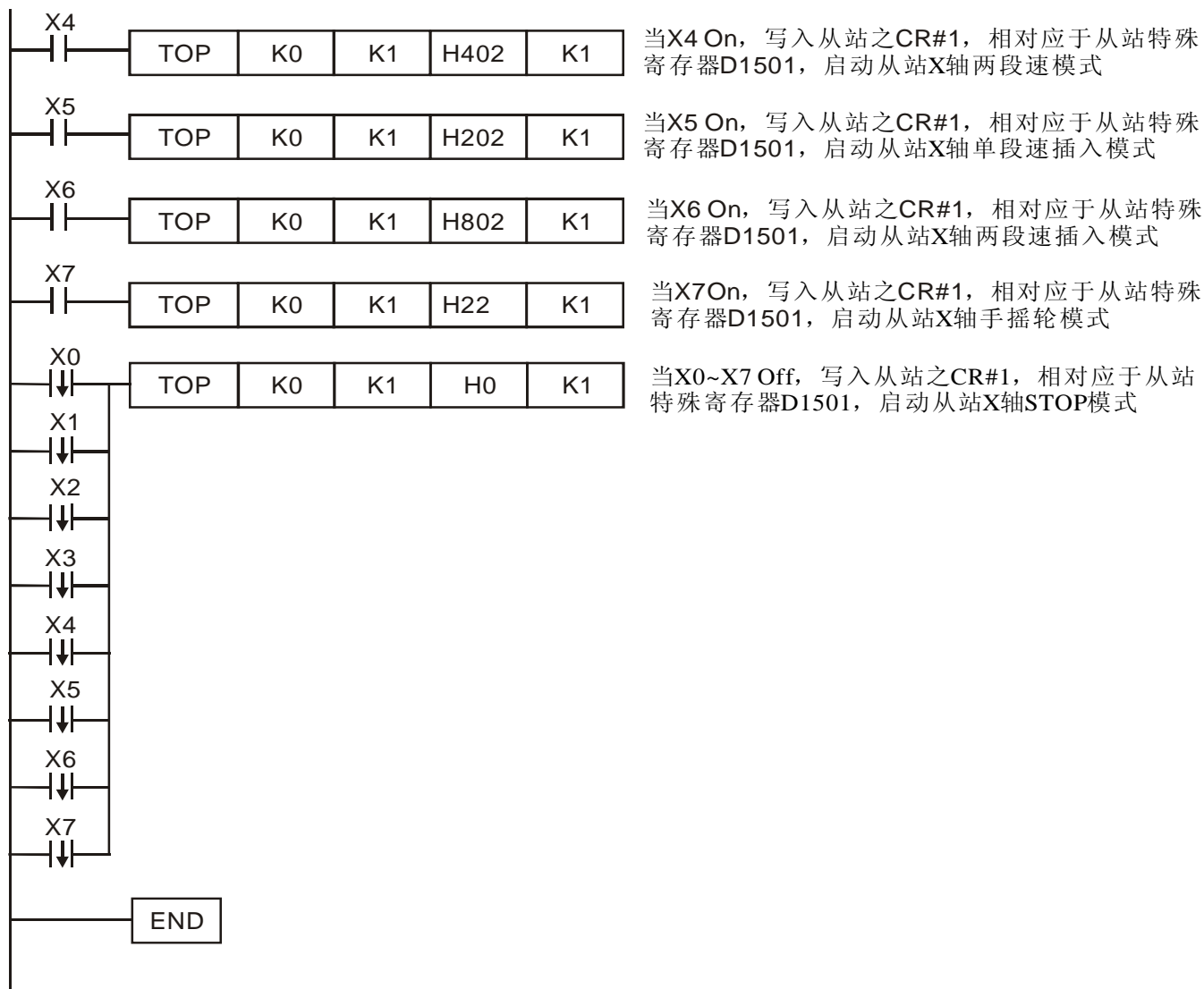
1. 若须使用到 DVP-PM 从站的其它模式功能时，可参考第 3 章，将所须功能的寄存器对应到上表中的「使用者自行规划」栏，并加入范例程序中，即可执行 DVP-PM 所提供的各种控制模式。
2. 从站内部自动规划的特殊寄存器区范围为 D1500~D1699，其中 D1500 为系统内定（只读）寄存器，主要是存放 DVP-PM 机种代码（H6260），故可由自由使用的范围为 D1501~1699。

【主站（DVP-EH2）程序说明】

梯形图：

动作说明：





从站 (DVP-PM) 程序说明:

梯形图

动作说明





◆ 范例二

【控制要求】

- DVP-EH2 主站下 FROM/TO 命令，对应从站 DVP-PM 内部特殊寄存器(D1500~D1699)，控制 OX 运动程序，执行多种运动模式(运动指令用法请参考第六章)。

【主站控制寄存器（Control Register）与从站特殊寄存器（Special Register）对应表】

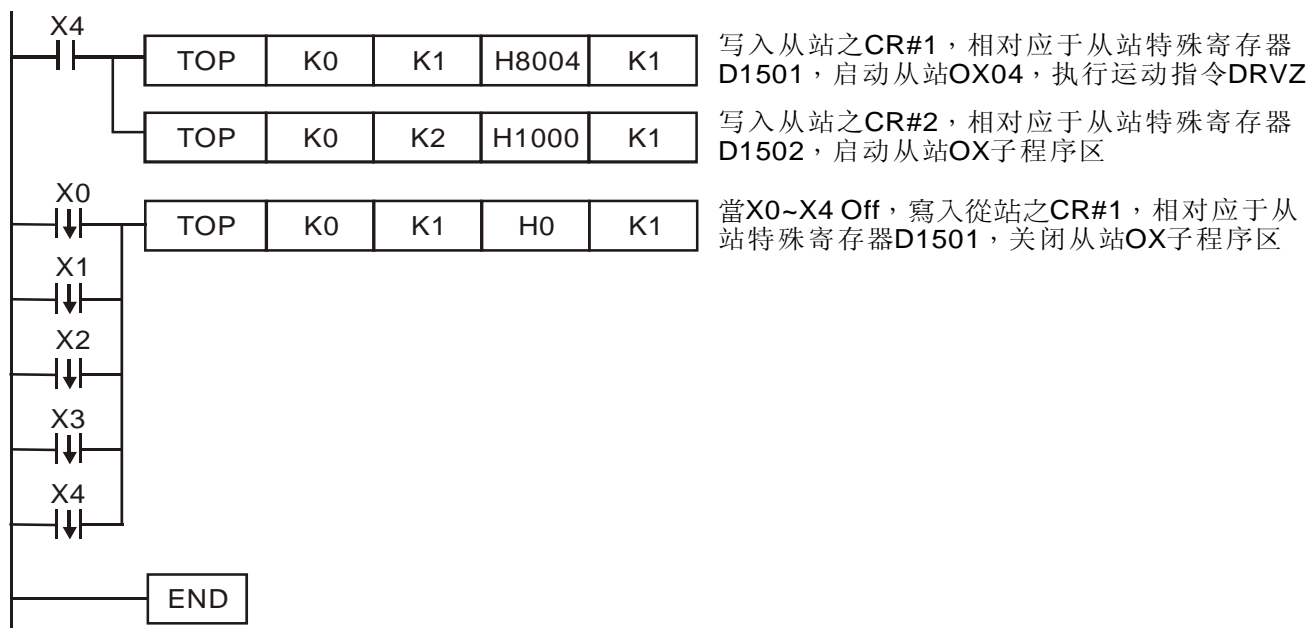
主站	从站		寄存器说明
	内部自动规划	使用者自行规划	
CR#0	D1500	-	DVP-PM 从机机种代号
CR#1	D1501	D1868	设定 OX 程序编号
CR#2	D1502	D1846	X 轴（OX）运转命令

【主站（DVP-EH2）程序说明】

梯形图：

动作说明：





【从站（DVP-PM）程序说明】

指令模式：

```

O100
LD M1002
DMOV K0 D1848
DMOV K0 D1928
LD M1000
MOV D1501 D1868
MOV D1502 D1846
M102

OX00
DRV X200000 FX100000 Y200000 FY100000
M2

OX01
LIN X100000 Y100000 F200000
M2

OX02
CW X0 Y100000 IO J50000 F200000
M2

OX03
CCW X0 Y100000 IO J50000 F200000
M2

OX04
BRET
DMOV K200000 D1828
DMOV K100000 D1830
DMOV K200000 D1908
DMOV K100000 D1910
DRVZ
M2
    
```

动作说明：

将初始化值规划于 O100 主程序区块，在从站启动 O100，将 X、Y 轴现在位置清除为 0，呼叫 OXn 运动子程序

在子程序区块 OX00 规划为运动指令 DRV

在子程序区块 OX01 规划为运动指令 LIN

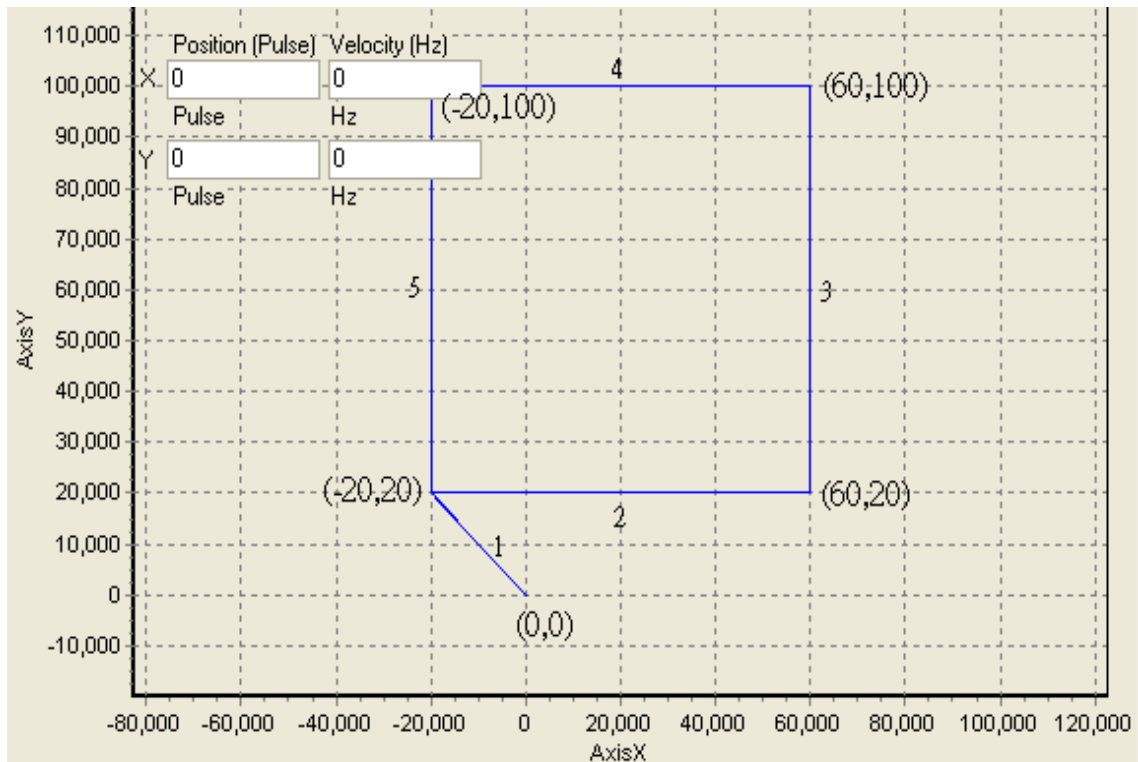
在子程序区块 OX02 规划为运动指令 CW

在子程序区块 OX03 规划为运动指令 CCW

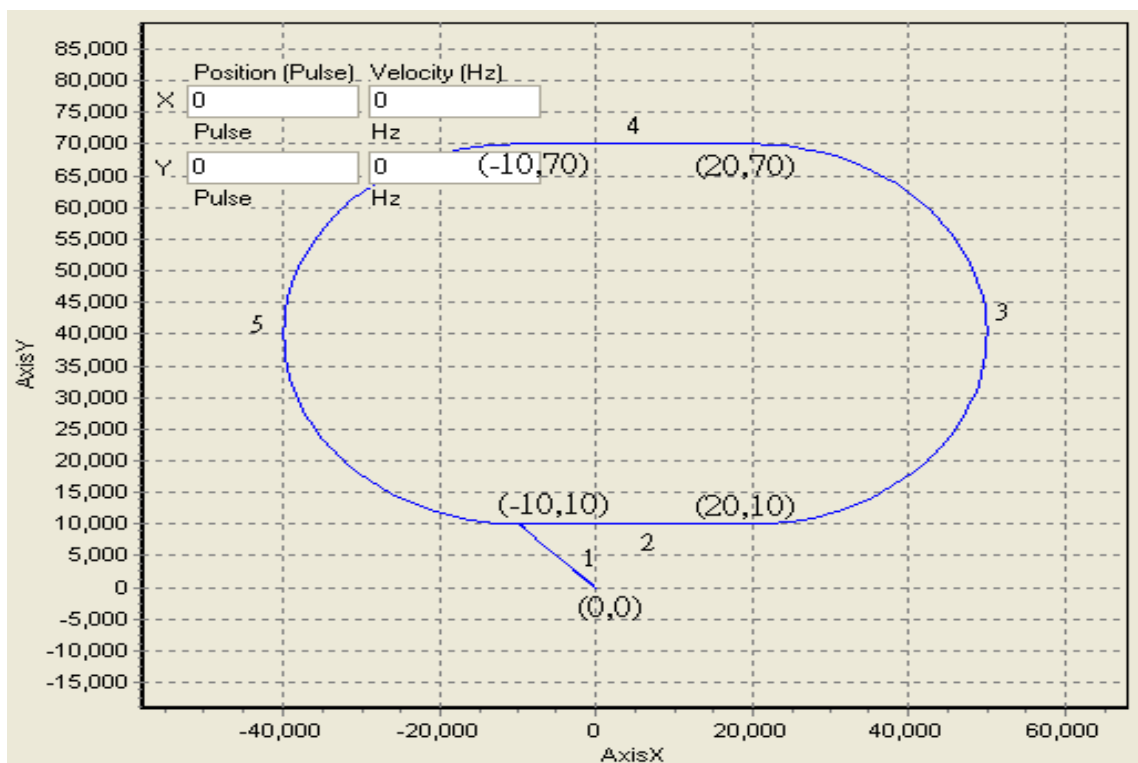
在子程序区块 OX04 规划为运动指令 DRVZ，并设定 DRVZ 指令相关参数

8.1 使用运动指令与 G-Code 来编写绘制轨迹

8.1.1 轨迹图

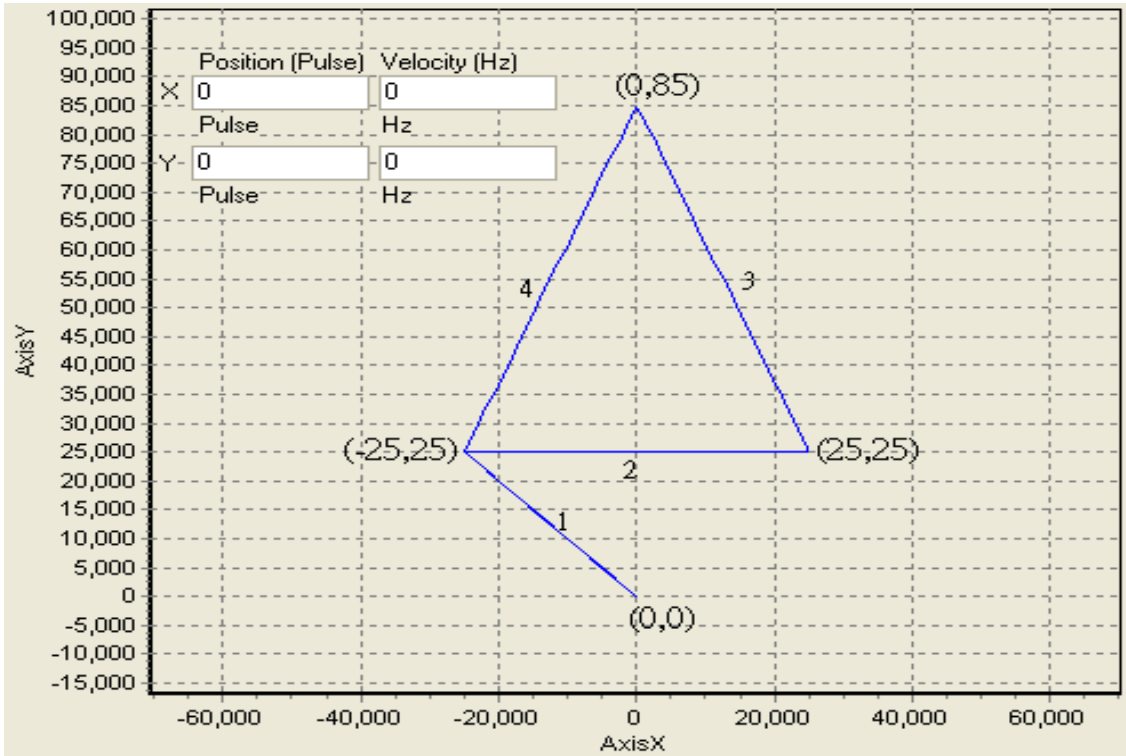


轨迹(一)

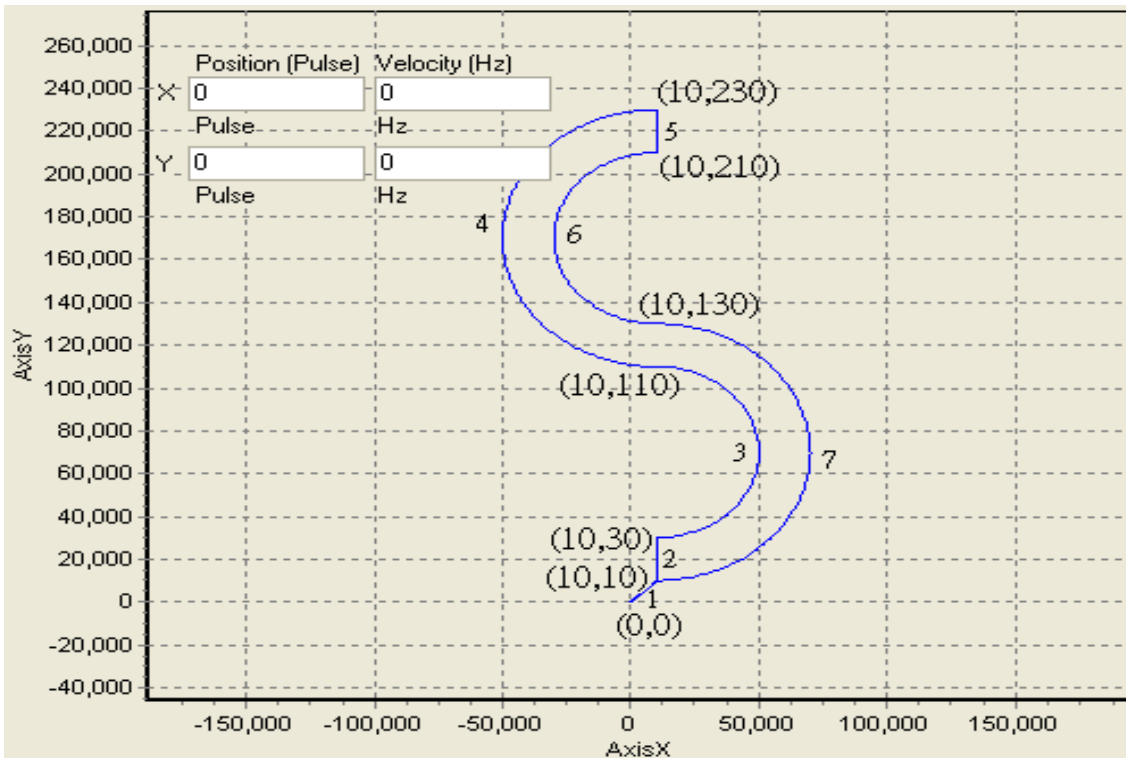


轨迹(二)

8 应用范例



轨迹(三)



轨迹(四)

8.1.2 设计操作步骤

1. 轨迹(一)规划四点的绝对坐标如下(-20, 20)、(60, 20)、(60, 100)、(-20, 100), 并由坐标(0, 0)出发执行。
2. 轨迹(二)规划四点的绝对坐标如下(-10, 10)、(20, 10)、(20, 70)、(-10, 70), 并由坐标(0, 0)出发执行。
3. 轨迹(三)规划三点的绝对坐标如下(-25, 25)、(25, 25)、(0, 85), 并由坐标(0, 0)出发执行。
4. 轨迹(四)规划七点的绝对坐标如下(10, 10)、(10, 30)、(10, 110)、(10, 230)、(10, 210)、(10, 130)、(10, 10), 并由坐标(0, 0)出发执行。
5. 运动指令程序代码撰写如下:

指令模式: 将初始化值规划于 O100 主程序区块, 将 X、Y 轴现在位置清除为 0, 并启动 OX0 子程序区块

O100			O100 主程序
LD	M1002		
DMOV	K0	D1848	X 轴现在位置归零
DMOV	K0	D1928	Y 轴现在位置归零
RST	M1074		关闭 OX 运动子程序
MOV	H8000	D1868	写入欲启动 OX 编号, 编号为 0
SET	M1074		启动 OX 运动子程序

M102

OX0 子程序: 在子程序区块呼叫指针 P0

OX0			OX 运动子程序
BRET			回母线
CALL	P0		呼叫 P0 子程序

M2

以下程序代码为轨迹(一)运动指令写法

P0				P0 子程序
ABST				取绝对坐标
DRV	X-20000	Y20000		快速移动至指定位置
LIN	X60000	Y20000	F20000	直线补间移动至指定位置, 另一写法 LIN X60000 F20000
LIN	X60000	Y100000	F20000	直线补间移动至指定位置, 另一写法 LIN Y100000
LIN	X-20000	Y100000	F20000	直线补间移动至指定位置, 另一写法 LIN X-20000
LIN	X-20000	Y20000	F20000	直线补间移动至指定位置, 另一写法 LIN Y20000
SRET				

以下程序代码为轨迹(一)G-Code 写法: 在指针区块规划所要运行的运动程序

P0			P0 子程序
G90			取绝对坐标
G00	X-20.0	Y20.0	快速移动至指定位置

8 应用范例

G01	X60.0	Y20.0	F20.0	直线补间移动至指定位置，另一写法 G01 X60.0 F20.0
G01	X60.0	Y100.0	F20.0	直线补间移动至指定位置，另一写法 G01 Y100.0
G01	X-20.0	Y100.0	F20.0	直线补间移动至指定位置，另一写法 G01 X-20.0
G01	X-20.0	Y20.0	F20.0	直线补间移动至指定位置，另一写法 G01 Y20.0

SRET

以下程序代码为轨迹(二)运动指令写法

P0				P0 子程序	
ABST				取绝对坐标	
DRV	X-10000	Y10000		快速移动至指定位置	
LIN	X20000	Y10000	F40000	直线补间移动至指定位置，另一写法 LIN X20000 F40000	
CCW	X20000	Y70000	J30000	F20000	圆弧补间移动至指定位置，另一写法 CCW Y70000 J30000 F20000
LIN	X-10000	Y70000	F20000	直线补间移动至指定位置，另一写法 LIN X-10000	
CCW	X-10000	Y10000	J-30000	F20000	圆弧补间移动至指定位置，另一写法 CCW Y10000 J-30000

SRET

以下程序代码为轨迹(二)G-Code 写法

P0				P0 子程序	
G90				取绝对坐标	
G00	X-10.0	Y10.0		快速移动至指定位置	
G01	X20.0	Y10.0	F40.0	直线补间移动至指定位置，另一写法 G01 X20.0 F40.0	
G03	X20.0	Y70.0	J30.0	F20.0	圆弧补间移动至指定位置，另一写法 G03 Y70.0 J30.0 F20.0
G01	X-10.0	Y70.0	F20.0	直线补间移动至指定位置，另一写法 G01 X-10.0	
G03	X-10.0	Y10.0	J-30.0	F20.0	圆弧补间移动至指定位置，另一写法 G03 Y10.0 J-30.0

SRET

以下程序代码为轨迹(三)运动指令写法

P0				P0 子程序
INCT				取相对坐标
DRV	X-25000	Y25000		快速移动至指定位置
LIN	X50000	Y0	F20000	直线补间移动至指定位置
LIN	X-25000	Y60000	F20000	直线补间移动至指定位置，另一写法 LIN X-25000 Y60000
LIN	X-25000	Y-60000	F20000	直线补间移动至指定位置，另一写法 LIN X-25000 Y-60000
DRV	X25000	Y-25000		快速移动至指定位置

SRET

以下程序代码为轨迹(三)G-Code 写法

P0				P0 子程序
G91				取相对坐标
G00	X-25.0	Y25.0		快速移动至指定位置
G01	X50.0	Y0	F20.0	直线补间移动至指定位置
G01	X-25.0	Y60.0	F20.0	直线补间移动至指定位置, 另一写法 G01 X-25.0 Y60.0
G01	X-25.0	Y-60.0	F20.0	直线补间移动至指定位置, 另一写法 G01 X-25.0 Y-60.0
G00	X25.0	Y-25.0		快速移动至指定位置
SRET				

以下程序代码为轨迹(四)运动指令写法

P0				P0 子程序	
ABST				取绝对坐标	
DRV	X10000	Y10000		快速移动至指定位置	
LIN	X10000	Y30000	F20000	直线补间移动至指定位置, 另一写法 LIN Y30000 F20000	
CCW	X10000	Y110000	J40000	F20000	圆弧补间移动至指定位置, 另一写法 CCW Y110000 J40000
CW	X10000	Y230000	R60000	F15000	圆弧补间移动至指定位置, 另一写法 CW Y230000 R60000 F15000
LIN	X10000	Y210000	F15000		直线补间移动至指定位置, 另一写法 LIN Y210000
CCW	X10000	Y130000	J-40000	F15000	圆弧补间移动至指定位置, 另一写法 CCW Y130000 J-40000
CW	X10000	Y10000	R60000	F20000	圆弧补间移动至指定位置, 另一写法 CW Y10000 R60000 F20000
SRET					

以下程序代码为轨迹(四)G-Code 写法

P0				P0 子程序	
G90				取绝对坐标	
G00	X10.0	Y10.0		快速移动至指定位置	
G01	X10.0	Y30.0	F20.0	直线补间移动至指定位置, 另一写法 G01 Y30.0 F20.0	
G03	X10.0	Y110.0	J40.0	F20.0	圆弧补间移动至指定位置, 另一写法 G03 Y110.0 J40.0
G02	X10.0	Y230.0	R60.0	F15.0	圆弧补间移动至指定位置, 另一写法 G02 Y230.0 R60.0 F15.0
G01	X10.0	Y210.0	F15.0		直线补间移动至指定位置, 另一写法 G01 Y210.0
G03	X10.0	Y130.0	J-40.0	F15.0	圆弧补间移动至指定位置, 另一写法 G03 Y130.0 J-40.0
G02	X10.0	Y10.0	R60.0	F20.0	圆弧补间移动至指定位置, 另一写法 G02 Y10.0 R60.0 F20.0
SRET					

6. DVP-PM M1072=On 即开始执行运动模式

8 应用范例

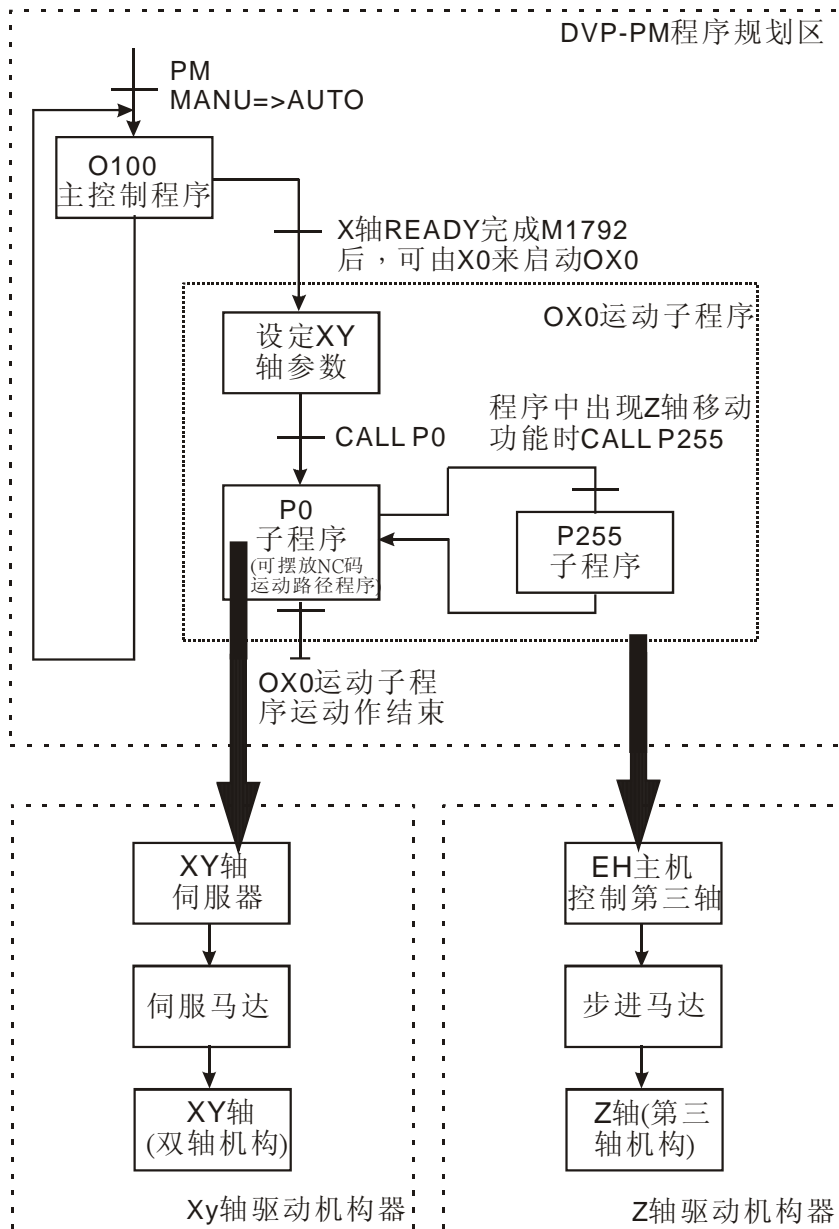
8.2 PMSoft 软件「motionSample」应用范例

利用此范例程序来实现以 DVP-PM 来画出英文字母或任何图形以及文字。若希望能运用在任何双轴的控制设备上，也可将范例程序进行修改，将可实现更多样化的控制程序。参考路径：首先打开 PMSoft 软件，点选「File」后，选择「Open Examples...」会出现对话框，再选择文件名为「motionSample_26Letter」的文档，即可开启此应用范例程序。

8.2.1 设计架构说明

当决定以 DVP-PM 来实现画出英文字母及图形时，必须先将英文字母及图形转成 G 码(即为 NC 码)，再设计 DVP-PM 的主控制程序，由于 DVP-PM 的 20D 机种只提供双轴 (XY 轴) 补间运动，故必须多增加 Z 轴来控制第三轴的「提笔」功能；为了增加第三轴的控制，在此范例中将利用 DVP-EH 系列的主机(亦可由其它控制器来取代)，以完成第三轴的控制动作。

整体设计架构图如下所示：



8.2.2 设计范例程序

DVP-PM 范例程序可分成四大区段：OX0 ~ M2, O100 ~ M102, P255 ~ SRET, P0 ~ SRET, 分别说明如下：

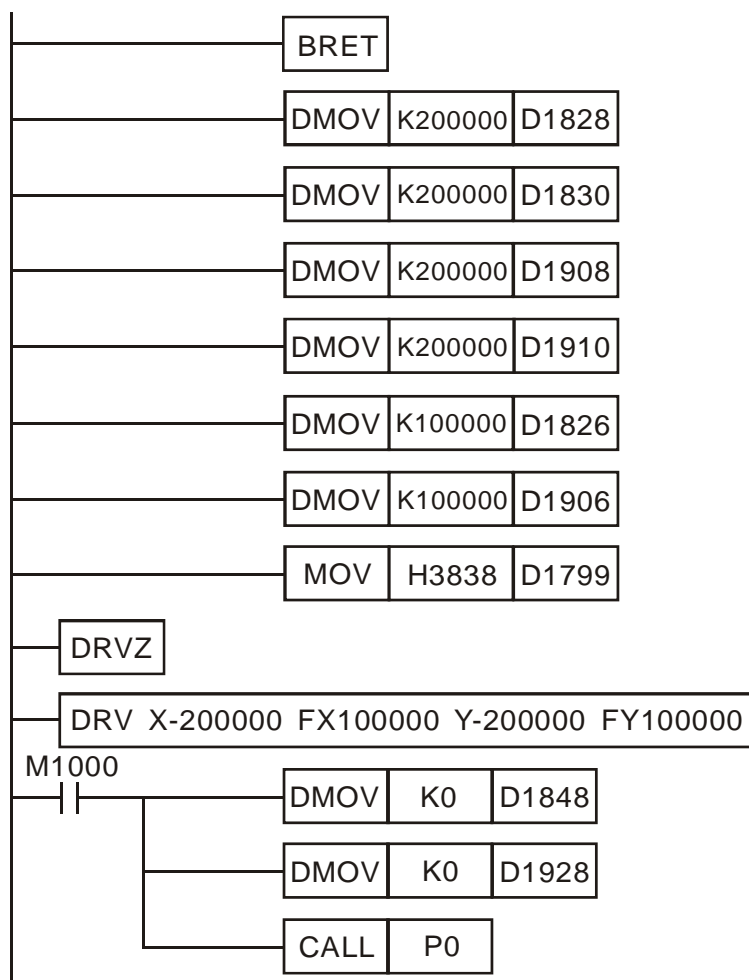
1. OX0 ~ M2: XY 轴功能参数设定区。

此区段主要为设定 X 及 Y 轴的参数，当 DVP-PM 为 AUTO 状态时，且 OX Ready 完毕 (M1792=ON) 时，则 X0 为 ON 将可启动 OX0 子程序。当 OX0 子程序启动后，设定原点复归所需之参数、JOG 寸动速度，以及设定 XY 轴的输入端子极性。

接着启动原点复归后，将 XY 轴以 100kHz 速度移动到 (-200000,-200000) 的工作位置，再将现在位置清除为 0，然后呼叫 P0 子程序后，并在 P0 子程序执行完毕后，即结束 OX0 子程序。

若须使用到其它控制模式时，请参考第 3 章之特 D 说明。

梯形图：



动作说明：

X轴原点复归速度 V_{RT} 为200kHz

X轴原点复归减速速度 V_{CR} 为200kHz

Y轴原点复归速度 V_{RT} 为200kHz

Y轴原点复归减速速度 V_{CR} 为200kHz

设定X轴JOG速度为100kHz

设定Y轴JOG速度为100kHz

设定XY轴输入端子极性

启动XY轴执行原点复归

XY轴以100kHz速度移动到 (-200000,-200000) 的位置

X轴现在位置清除为0

Y轴现在位置清除为0

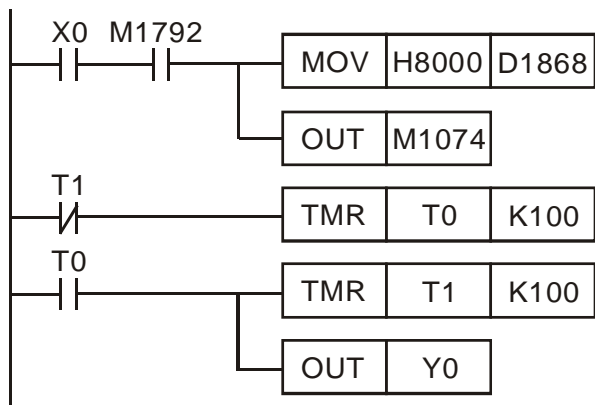
呼叫P0子程序

2. O100 ~ M102: 主程控区

此 O100 主程序主要为控制 OX0 子程序是否启动，当程序中 X0 (为启动 OX0 的条件开关) 和 M1792 (为判断 OX 是否 Ready 完成的旗标) 皆为 ON 状态时，将可启动 OX0 子程序。使用者更可规划执行其它运算动作于主程序之中。

8 应用范例

梯形图：



动作说明：

当OX Ready完成(M1792=ON)
，准备启动OX0运动子程序

启动OX0运动子程序

可规划执行其它运算动作

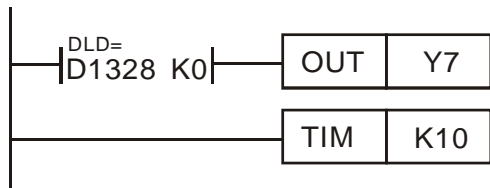
可规划执行其它运算动作

可规划执行其它运算动作

3. P255 ~ SRET：第三轴（Z 轴）控制信号产生区

当 P0 子程序中的 G-Code（即 NC 码）指令 G00 及 G01 出现 Z 操作数时，P255 子程序将会自动启动。利用 D1328 值来决定 Y7 是否 ON/OFF，以提供信号给 DVP-EH 主机控制提笔或下笔动作（即为 Z 轴之上下动作）。详细说明请参考第 6 章节中的 G00、G01 指令说明。

梯形图：



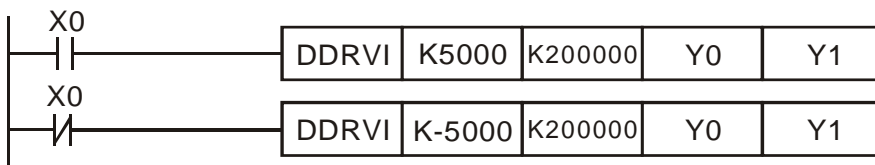
动作说明：

利用Z轴目标位置(D1328)来决定
，Y7为提笔控制信号

停顿0.1秒

由于 DVP-PM 的 P255 子程序中，有规划驱动 Z 轴的控制信号（Y7），并将 Y7 的 ON/OFF 信号，提供给 DVP-EH 系列主机的外部输入点 X0 装置。因此，当 X0 为 ON 状况时，则输出脉波控制步进马达将 Z 轴往上移动（提笔）；X0 为 OFF 状态时，则输出脉波控制步进马达将 Z 轴往下移动（下笔）。

DVP-EH 主机程序规划如下：

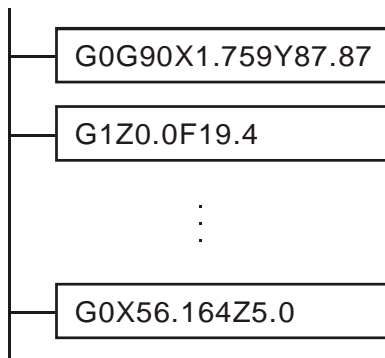


将 DVP-EH 主机的输入装置 Y0 及 Y1 配接至步进马达的脉波输入端。

4. P0 ~ SRET：双轴（X-Y 轴）补间控制运动区

使用者将文字或图形转成 G-Code（NC 码）后，为了使程序更简化，因此不将 G-Code 放入 OX0 子程序中，而将 G-Code 式放入 P0 子程序之中，配合上述 1~3 区段之程序，将可进行文字或图形的绘制。如下范例程序：

梯形图：



动作说明：

G-Code(NC码)放置区

当完成 1 ~ 4 区段程序后，即可实现以 DVP-PM 来画出英文字母或任何图形，以及文字。

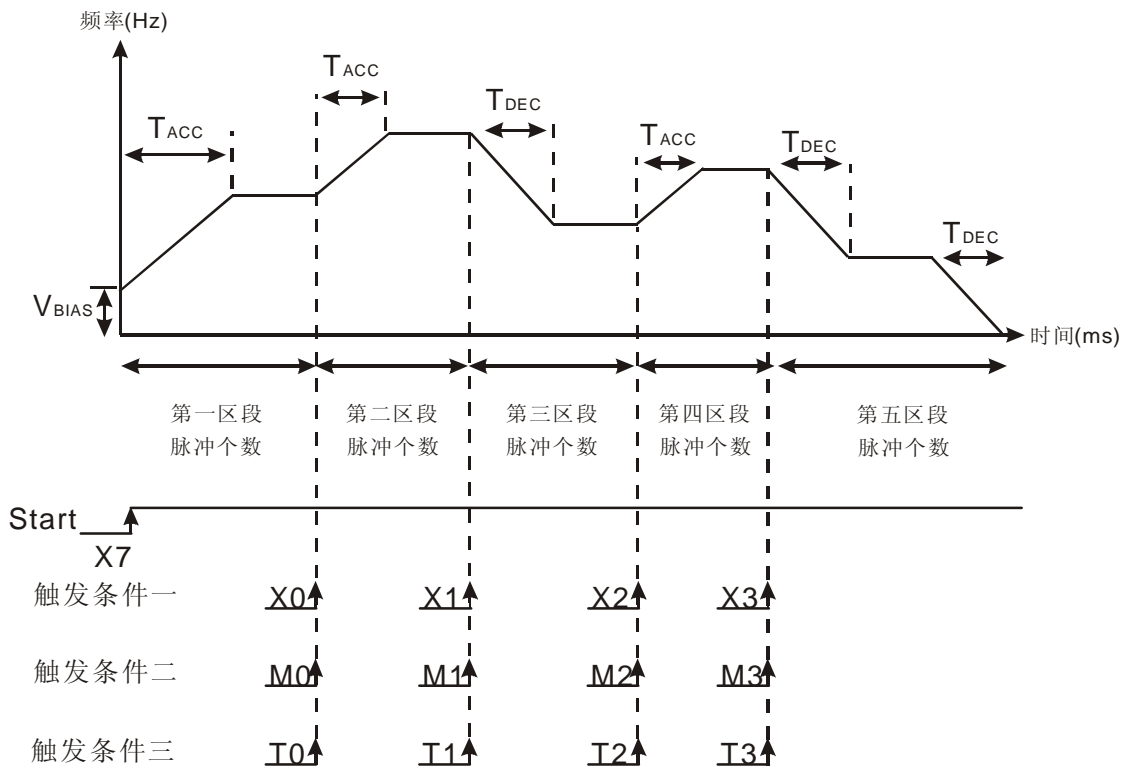
8 应用范例

8.3 自行规划多段变速度

利用单段速输出模式，在固定之行程内，自行规划多段变速度之触发条件。

8.3.1 设计架构说明

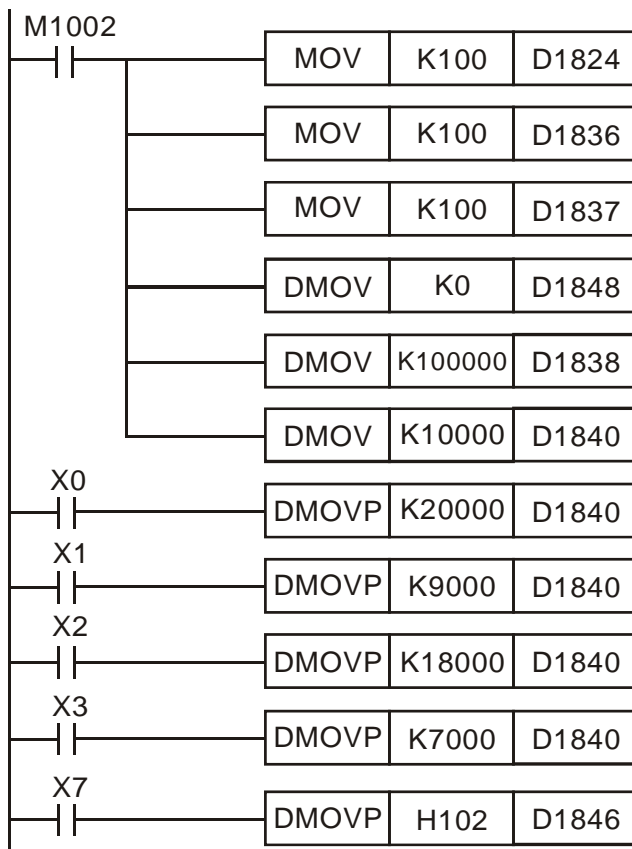
1. 触发条件一：外部输入讯号控制，X0~X3 分别切换第二段速~第五段速。
2. 触发条件二：比较现在位置控制，M0~M3 分别切换第二段速~第五段速。
3. 触发条件三：由时间控制，T0~T3 分别切换第二段速~第五段速。



32 位之 D1838(设定总输出脉波数)=第一区段+第二区段+.....+第五区段脉波数。

8.3.2 设计范例程序

触发条件一梯形图:



动作说明:

设定X轴之启动速度 V_{BIAS} 设定X轴之加速时间 T_{ACC} 设定X轴之减速时间 T_{DEC}

X轴现在位置清除为0

设定X轴所有区段行走之位移量

设定X轴第一区段运转速度

X0=On, 变换运转速度为 20,000Hz

X1=On, 变换运转速度为 9,000Hz

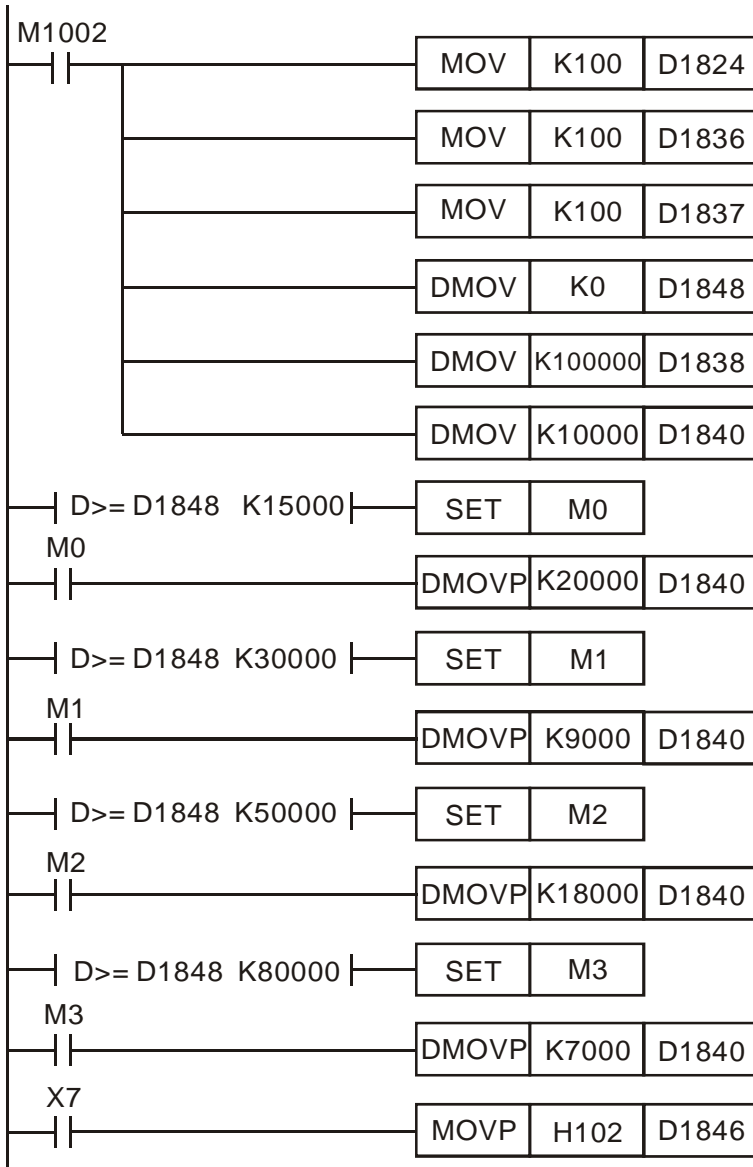
X2=On, 变换运转速度为 18,000Hz

X3=On, 变换运转速度为 7,000Hz, 输出至总行程结束

X7=On, 软件启动X轴运转命令(单段速)

8 应用范例

触发条件二梯形图：



动作说明：

设定X轴之启动速度 V_{BIAS}

设定X轴之加速时间 T_{ACC}

设定X轴之减速时间 T_{DEC}

X轴现在位置清除为0

设定X轴所有区段行走之位移量

设定X轴第一区段运转速度

比对现在位置

M0=On，变换运转速度为20,000Hz

比对现在位置

M1=On，变换运转速度为9,000Hz

比对现在位置

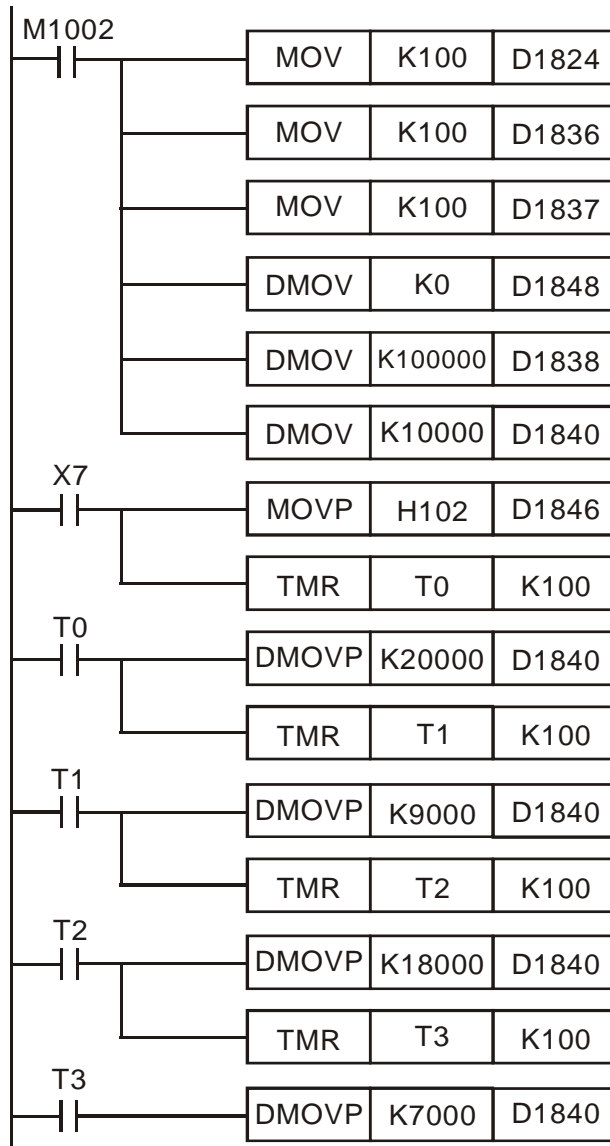
M2=On，变换运转速度为18,000Hz

比对现在位置

M3=On，变换运转速度为7,000Hz，
输出至总行程结束

X7=On，软件启动X轴运转命令
(单段速)

触发条件三梯形图：



动作说明：

设定X轴之启动速度 V_{BIAS}

设定X轴之加速时间 T_{ACC}

设定X轴之减速时间 T_{DEC}

X轴现在位置清除为0

设定X轴所有区段行走之位移量

设定X轴第一区段运转速度

X7=On, 软件启动X轴运转命令(单段速), 并开始计数准备切换第二区段

T0=On, 变换运转速度为20,000Hz, 并开始计数准备切换第三区段

T1=On, 变换运转速度为9,000Hz, 并开始计数准备切换第四区段

T2=On, 变换运转速度为18,000Hz, 并开始计数准备切换第五区段

T3=On, 变换运转速度为7,000Hz, 输出至总行程结束

8 应用范例

8.4 DVP-PM (Master) 连接 DVP01PU-H2 (Slave) 第三轴控制 (仅供用于 DVP20PM00D 机种)

动作说明:

1. 启动 O100, 执行 OX0。
2. 在子程序 OX0, 遇到 G01 Z-25000 F10000, 会呼叫 P255。
3. 在 P255, 当 D1328<0, 执行 DVP01PU-H2 目标位置为 K1000, 运转速度为 K10000。
4. 执行完 P255 返回 OX0, 并开始等待 10 秒。
5. 在子程序 OX0, 遇到 G01 Z10000 F20000, 会呼叫 P255。
6. 在 P255, 当 D1328>0, 执行 DVP01PU-H2 目标位置为 K2000, 运转速度为 K20000。
7. 执行完 P255 返回 OX0。

主程序与子程序码撰写如下:

指令模式: 将初始化值规划于 O100 主程序区块, 将 X、Y 轴现在位置清除为 0, 并启动 OX0 子程序区块
O100 主程序

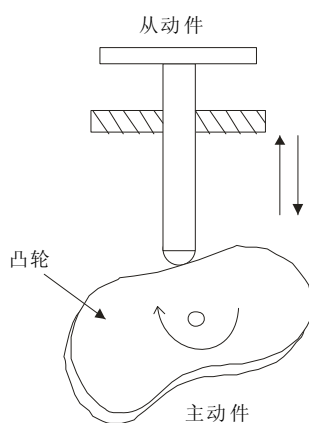
(O100)					O100 主程序
LD	M1002				
MOV	H8000	D1868			写入欲启动 OX 编号, 编号为 0
SET	M1074				启动 OX 运动子程序
M102					
	OX0 子程序				
(OX0)					OX 运动子程序
G01	Z-25000	F10000			G01 第三轴控制
TIM	K1000				停顿 10 秒
G01	Z10000	F20000			G01 第三轴控制
M2					
	P255 子程序				
(P255)					P255 子程序
BRET					
TO	K0	K31	H2	K1	DVP01PU 软件关闭
DLD>=	K0	D1328			D1328 比较判断
DTO	K0	K23	K1000	K1	设定 DVP01PU 目标位置
DTO	K0	K25	D1330	K1	设定 DVP01PU 运转速度
TO	K0	K32	H1	K1	设定 DVP01PU 单段速
TO	K0	K31	H100	K1	DVP01PU 软件启动
DLD<	K0	D1328			D1328 比较判断
DTO	K0	K23	K2000	K1	设定 DVP01PU 目标位置
DTO	K0	K25	D1330	K1	设定 DVP01PU 运转速度

TO	K0	K32	H1	K1	设定 DVP01PU 单段速
TO	K0	K31	H100	K1	DVP01PU 软件启动
SRET					

9.1 电子凸轮(E-CAM)简介

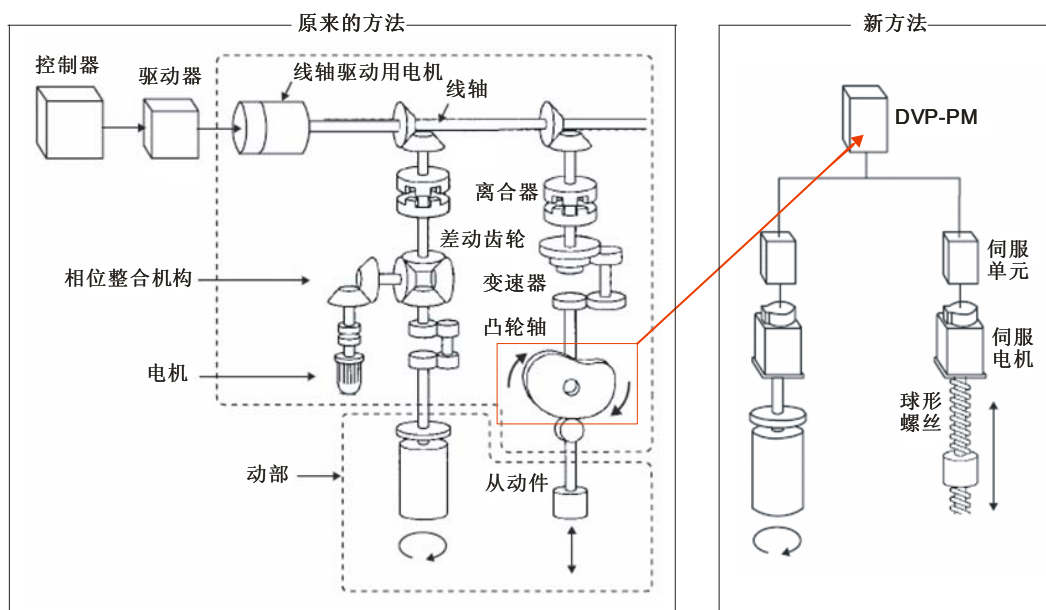
传统的机械凸轮机构，是由凸轮、从动件以及机架等三部份组成。

- ◆ 机械凸轮是一种不规则形状的机件，一般为等转速的输入件，可经由直接接触传递运动到从动件，使从动件依设定的规律运动。
- ◆ 从动件为机械凸轮所驱动的被动件，一般为产生不等速、不连续、不规则运动的输出件。
- ◆ 机架则是用来支持机械凸轮与从动件的机件。



相对传统的方式，使用电子凸轮有着以下的好处：

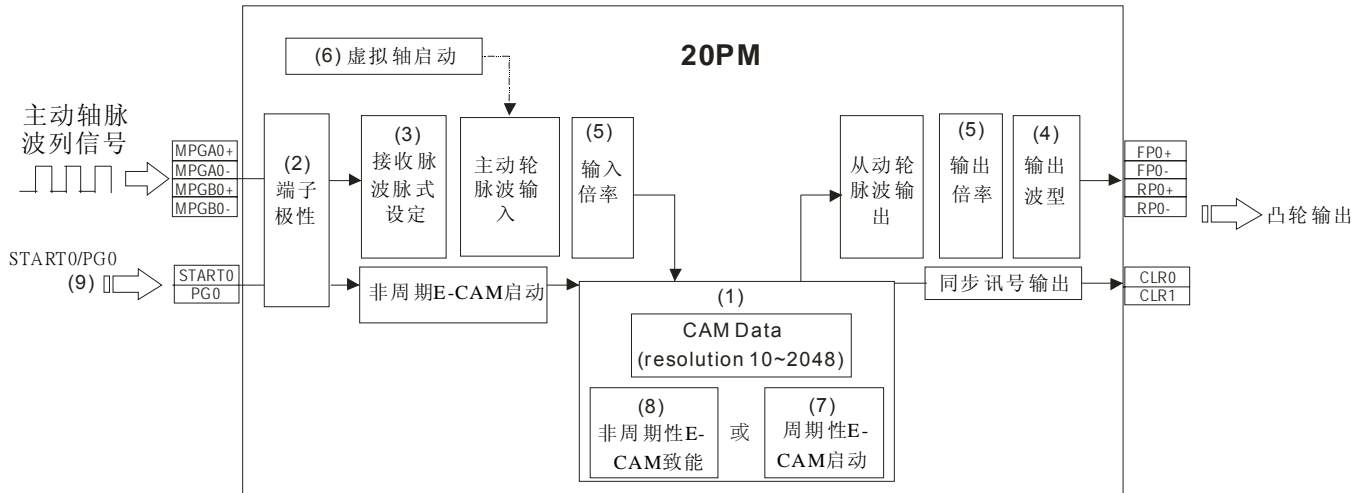
- ◆ 较友善的使用者接口
- ◆ 不同的产品需要不同的凸轮曲线，可利用软件修改电子凸轮表内电子凸轮数据，不用修改机构
- ◆ 可以有较高的加速
- ◆ 运行可以较为平顺



9 电子凸轮

9.2 电子凸轮的实现方式

■ 架构图



■ 操作步骤

STEP1	STEP2	STEP3
初始设定	选择主轴信号来源	启动/停止电子凸轮
(1)建立电子凸轮数据 (2)端子设定 (3)输入/出脉冲形式设定 (4)输入/输出倍率 启动角度设定	主轴伺服编码器 主轴输出脉冲信号 (6)虚拟轴启动	(7)周期性凸轮启动/停止 (8)非周期性凸轮启动/停止

9.2.1 初始设定

9.2.1.1 建立电子凸轮数据(E-CAM Data)

20PM 提供两种建立电子凸轮数据：

方法 1：使用 DTO 指令自动生成电子凸轮数据

方法 2：使用 PMsoft CAM Chart 绘制电子凸轮曲线

建立电子凸轮数据详细设定方法请参考章节 9.4 节。

9.2.1.2 端子设定

■ 输入端子

1. MPGA0/MPGB0: 主轴脉冲信号输入端子，最高可以接收 200kHz 脉冲信号。
2. START0/PG0: 非周期式电子凸轮模式下凸轮启动信号输入端子。

■ 输出端子

1. FP/RP: 电子凸轮脉冲信号输出端子, 最高可以输出 500kHz 脉冲信号。
2. CLR0/CLR1: 电子凸轮同步信号输出端子, 当目标位置(I)(D1839,D1838) \leq 主轴位置 \leq 目标位置(II)(D1843,1842)时同步信号 ON; 详细使用说明请参考章节 9.3。

■ 输入信号极性设定

输入端子的极性为 A 接点时, 请将 Bit# = ON。若为 B 接点时, 则将 Bit# = OFF 即可。若需主轴 (X 轴手摇轮接收端) 输入脉冲极性 MPGA0/MPGB0 设为 A 接点, 则将 D1799 b1/b2 设 ON, 即 D1799 = 6。

D1799 输入端子极性

Bit#	X 轴输入端子极性
0	PG0
1	MPGB0
2	MPGA0
7	START0

■ 输入端子数字滤波功能

D1806 High Byte 为 MPG0/1 滤波系数, D1806 Low Byte 为其它输入端子(除了 X 点)滤波系数, 滤波频率 = $85000/2^{N+4}$ (kHz)。下表为 N 与对应的滤波系数参考表:

N	kHz	N	kHz
1	2656.25	11	2.593994
2	1328.125	12	1.296997
3	664.0625	13	0.648499
4	332.0313	14	0.324249
5	166.0156	15	0.162125
6	83.00781	16	0.081062
7	41.50391	17	0.040531
8	20.75195	18	0.020266
9	10.37598	19	0.010133
10	5.187988		

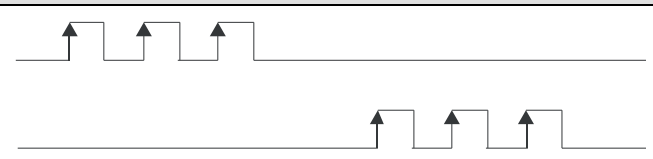
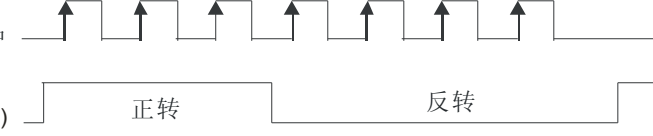
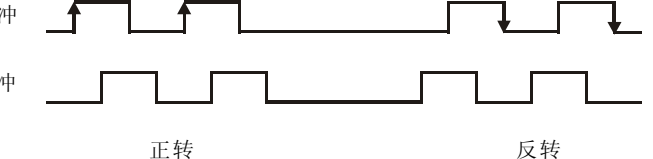
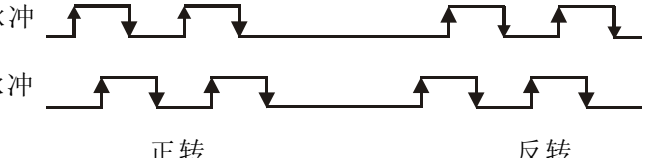
9 电子凸轮

9.2.1.3 输入/出脉冲形式设定

若主轴输入来源为 PM 之 Y 轴实际信号输出或是虚轴模式时，则主轴接收脉冲型式设定与 Y 轴输出脉冲型式要搭配设定；假设主轴接收脉冲型态设定为单脉冲(P/D)输入时，Y 轴输出脉冲设定也要设为单脉冲(P/D)，最后硬件线路只要接 MPGA0(A0+、A0-)，对应 Y 轴输出(FP1+、FP1-)。

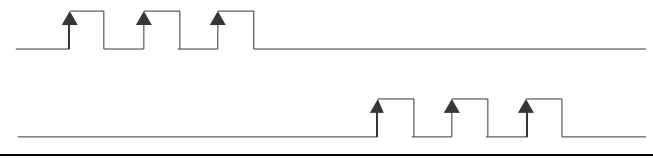
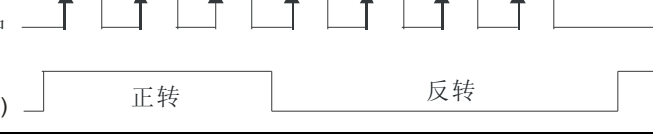
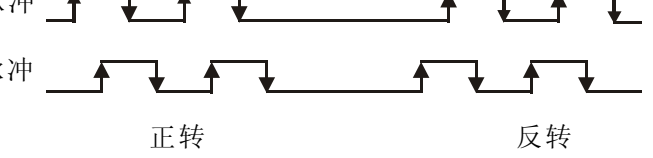
■ 输入脉冲形式设定

手摇轮输入设定 D1864

b9	b8	输入脉冲型式 (正逻辑)	说明
0	0	FP 正转脉冲  RP 反转脉冲	双脉冲(FP/RP)
0	1	FP 脉冲  RP 方向 (DIR)	单脉冲(P/D)
1	0	FP A相脉冲  RP B相脉冲	A/B 相脉冲 (1 倍频)
1	1	FP A相脉冲  RP B相脉冲	A/B 相脉冲 (4 倍频)

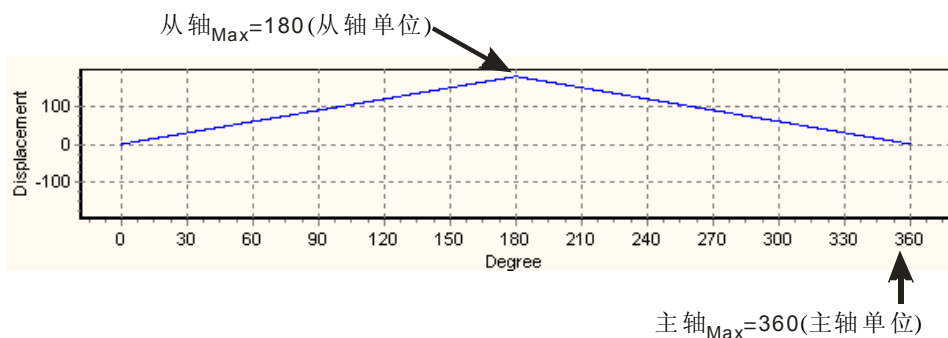
■ 输出脉冲形式设定

脉冲输出方式设定 D1816、D1896、D1976

b5	b4	脉冲输出型式 (正逻辑)	说明
0	0	FP 正转脉冲  RP 反转脉冲	双脉冲(FP/RP)
0	1	FP 脉冲  RP 方向 (DIR)	单脉冲(P/D)
1	0	FP A相脉冲  RP B相脉冲	A/B 相脉冲 (AB)
1	1		

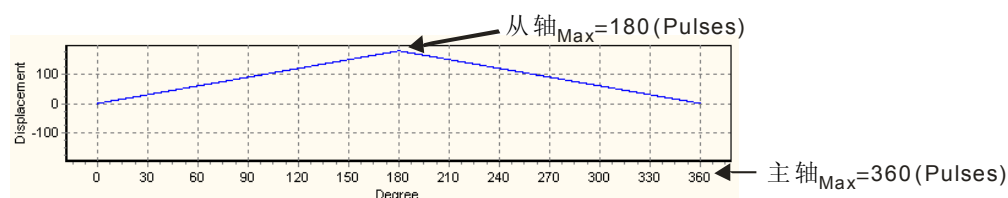
9.2.1.4 比例设定(输入/输出倍率)

DVP-PM 的电子凸轮功能有提供 E-CAM Data(E-CAM Data)倍率放大的功能, 当使用者输入 E-CAM Data 如下图时, 透过 D1816 b0/b1 单位系设定与电子齿轮比(D1858,D1859)设定, 可以将 E-CAM Data 做尺寸的放大 / 缩小, 控制输入/输出倍率。



D1816 参数设定

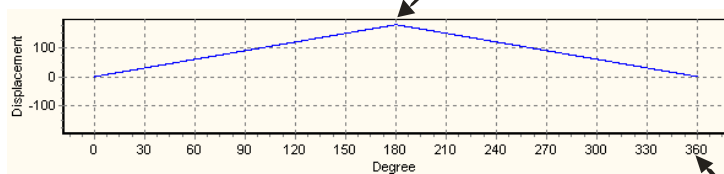
b1	b0	单位系
0	0	马达单位



D1816 参数设定

b1	b0	单位系
0	1	机械单位
1	0	复合单位
1	1	

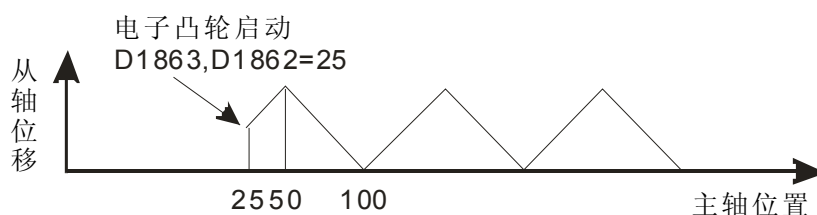
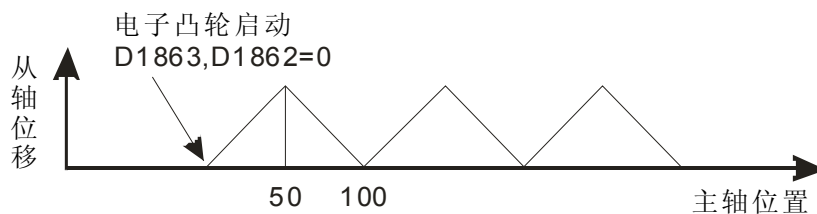
$$\frac{\text{从轴}_{\text{Max}}=180 \times \text{D1819} \cdot \text{D1818} (\text{马达转一圈所需脉冲数})}{\text{D1821} \cdot \text{D1820} (\text{马达转一圈之移动距离})} (\text{Pulses}) = \text{输出倍率}$$



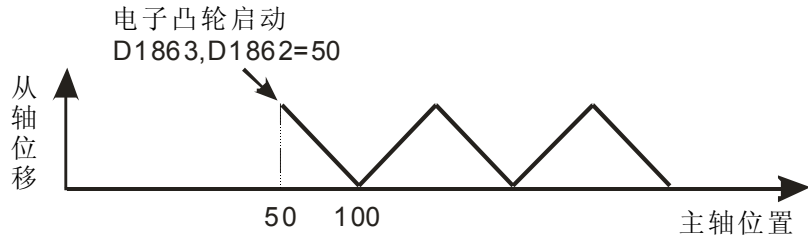
$$\frac{\text{主轴}_{\text{Max}}=360 \times \text{D1858} \text{电子齿轮 (分子)}}{\text{D1859} \text{电子齿轮 (分母)}} (\text{Pulses}) = \text{输入倍率}$$

9.2.1.5 电子凸轮起始角度选择

使用者可藉由设定 M1752 来启动起始角度选择功能, 同时将要启动的位置写在 D1863, D1862 内。



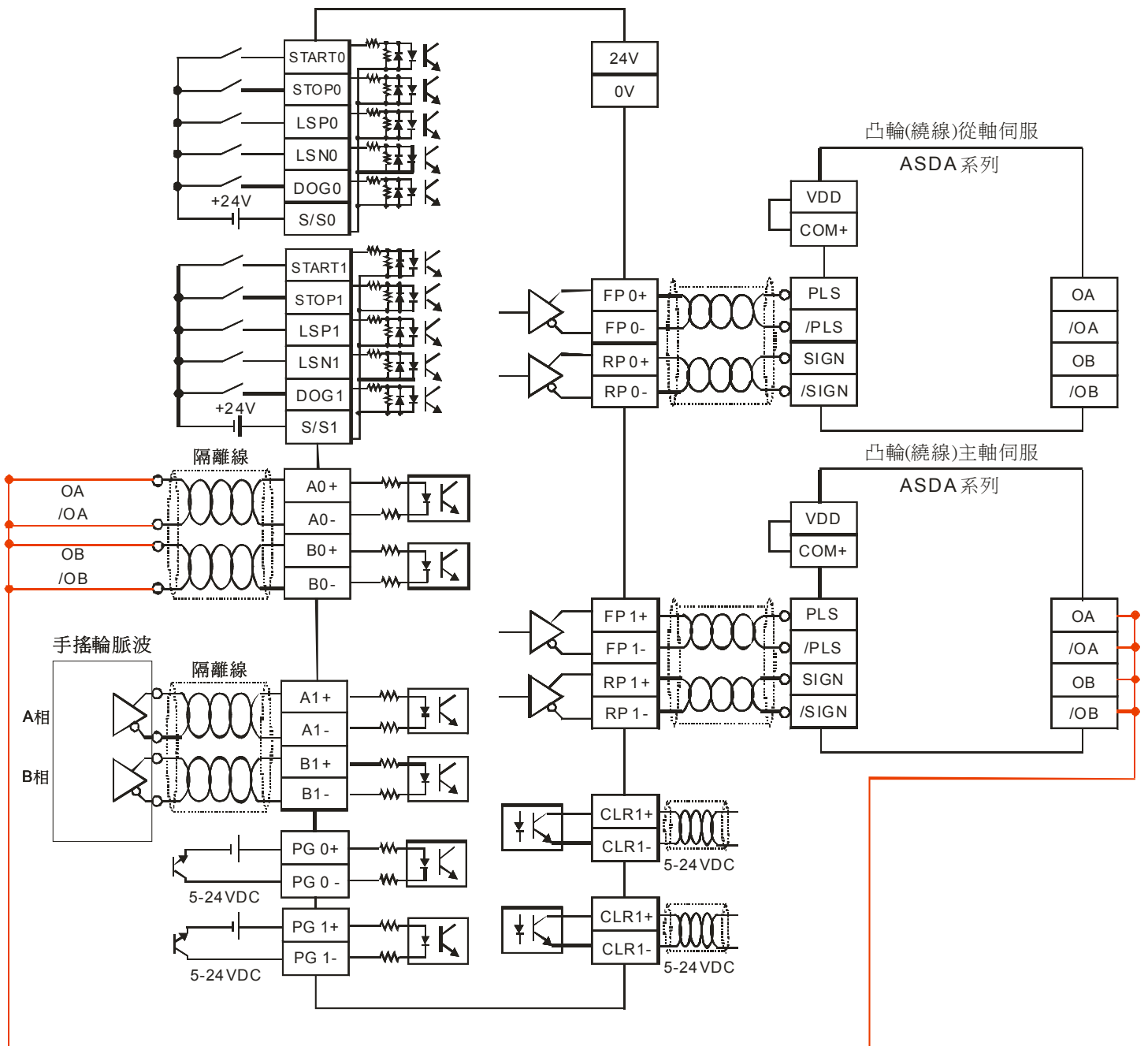
9 电子凸轮



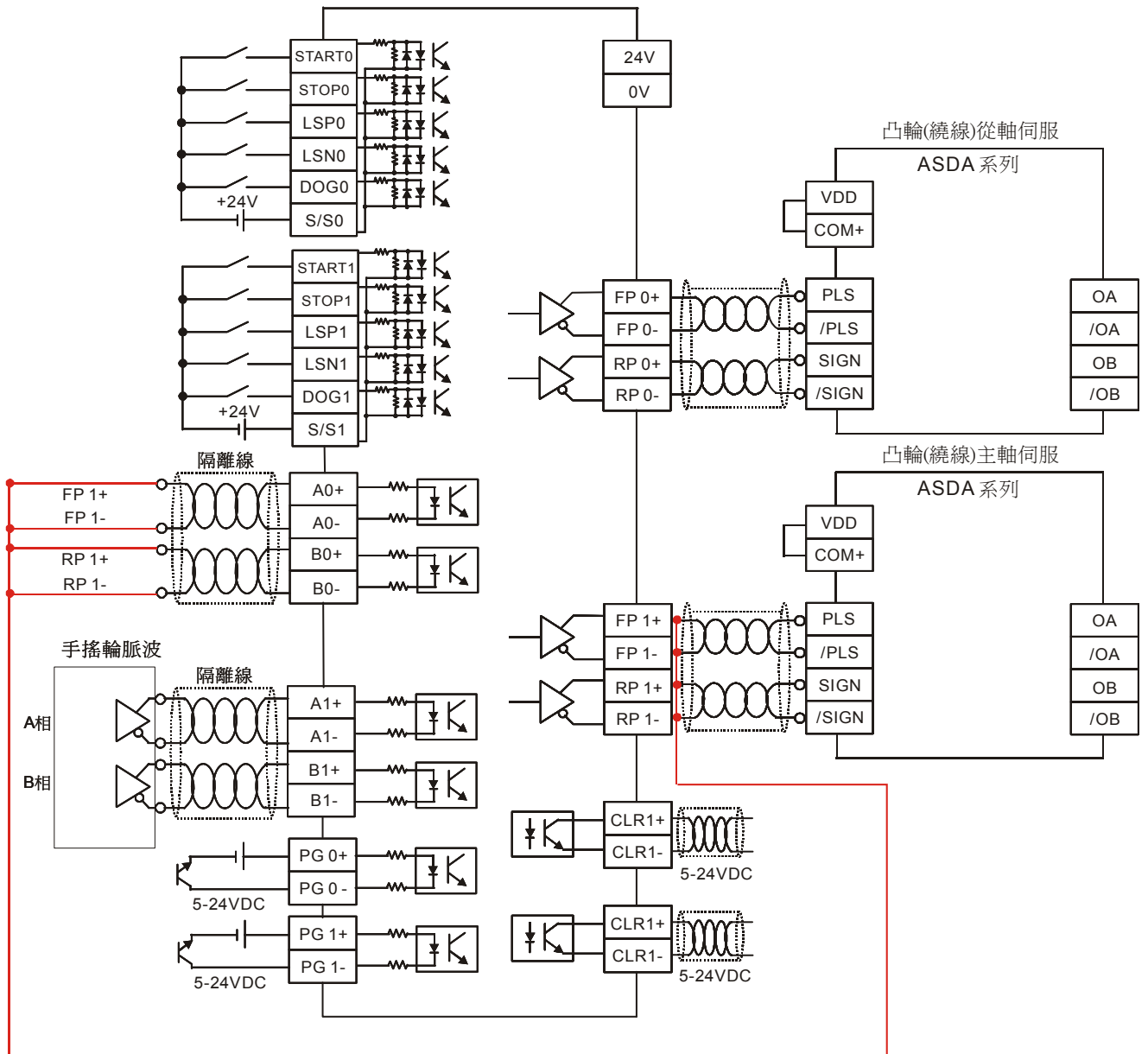
9.2.2 主轴信号来源设定

获取主轴位置有下列三种方法:

方法一：从主轴伺服编码器获取主轴位置，在获得编码器信号之后，将其换算成主轴位置。



方法二：由主轴输出脉冲获取主轴位置，将主轴输出脉冲信号进行处理之后，将其换算成主轴位置。

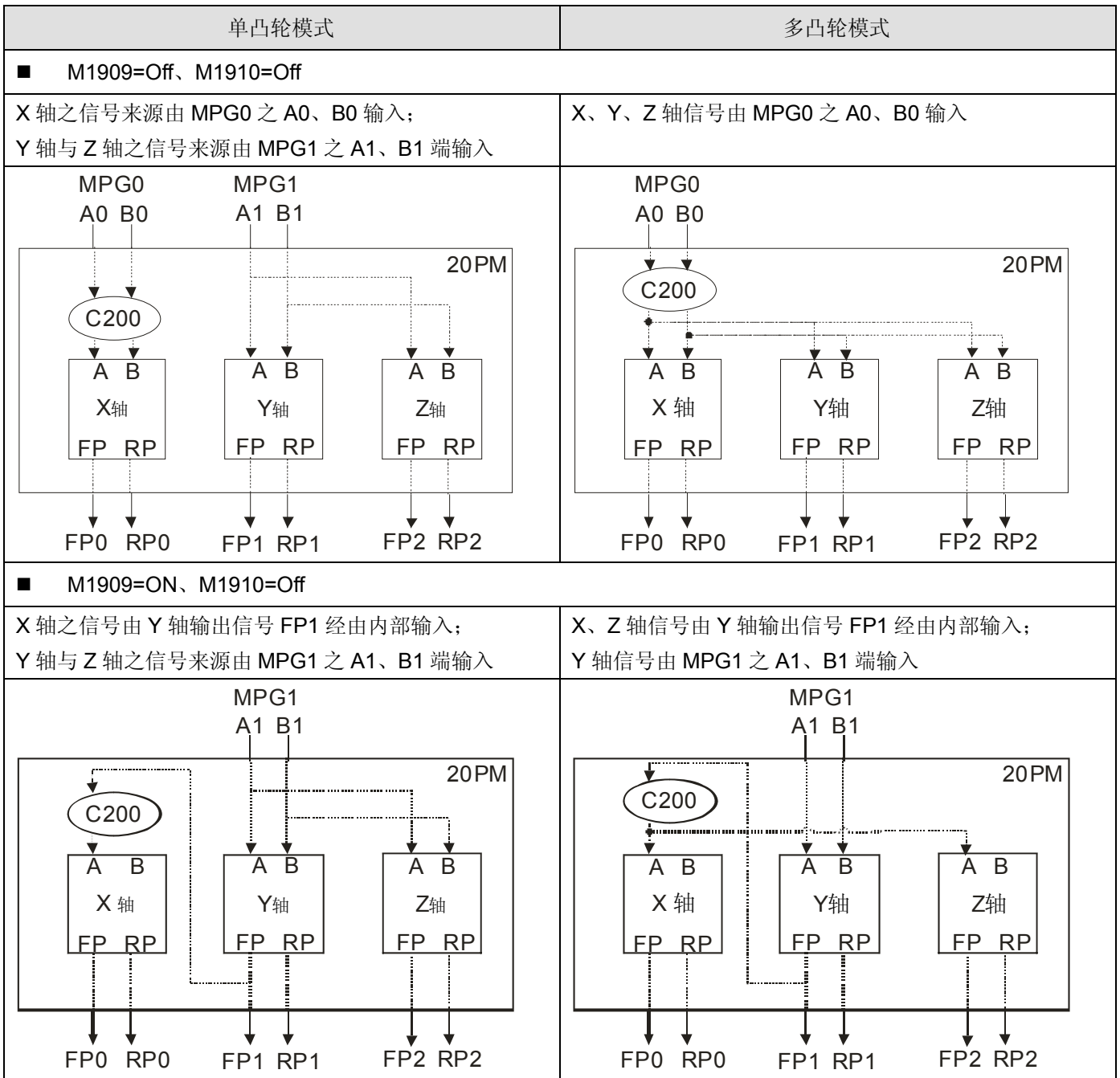


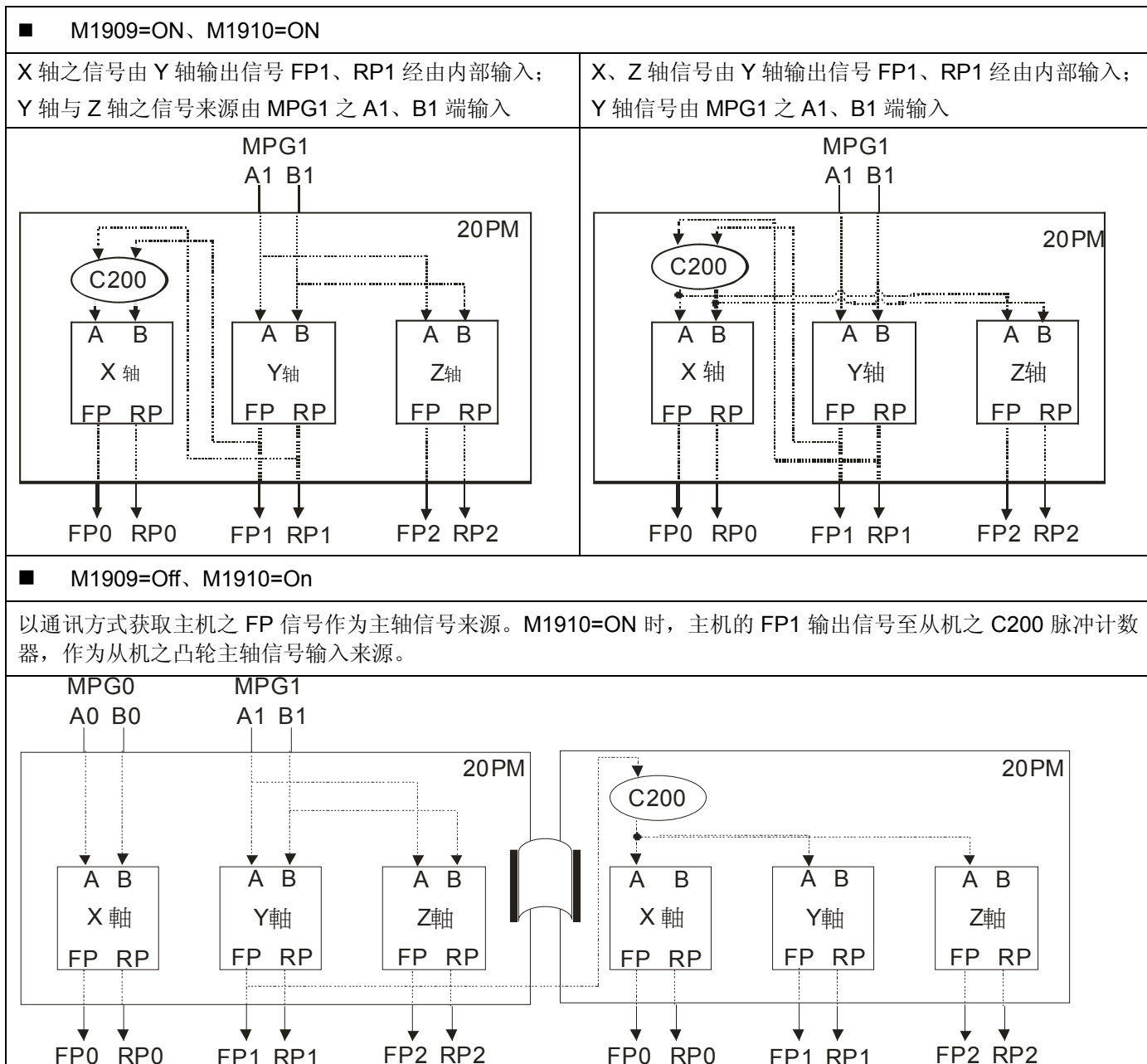
9 电子凸轮

方法三：采用虚轴模式：

在虚轴模式下，X轴及Z轴凸轮可以直接撷取20PM内部的Y轴输出信号作为主轴信号来源，毋须从外部安装线路。使用方法为设定M1909由内部连结Y轴的FP到A0，设定M1910由内部连结Y轴的RP到B0，再利用内部计数器C200做为主轴脉冲计数器。仅设定M1910时，则利用通讯的方式，由主机之Y轴FP1作为主轴信号来源。

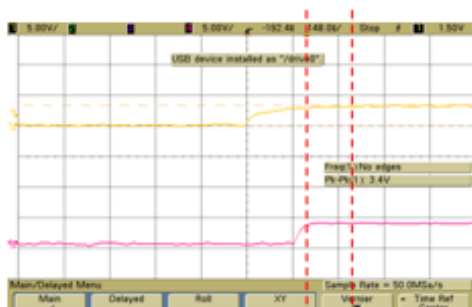
设定虚轴模式时，需考虑凸轮轴的手摇轮输入形式与Y轴脉冲形式之配合，设定方法请参照章节9.2.1.3说明。以下介绍单凸轮与多凸轮采用虚轴模式时的信号示意图：



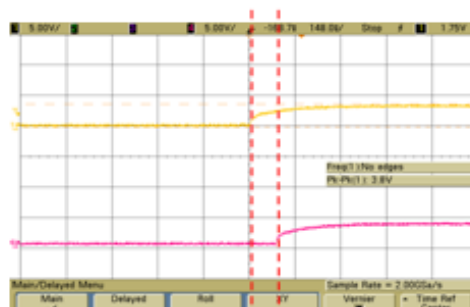


【备注】以下面的数据来算约需 29 台才会落后一个 500KHz 脉冲(2us)，可以说是毫无延迟，同时经过每一台 PM 的信号，无信号衰减问题。

2台PM的延迟
0.148us



1台PM的延迟
0.07us

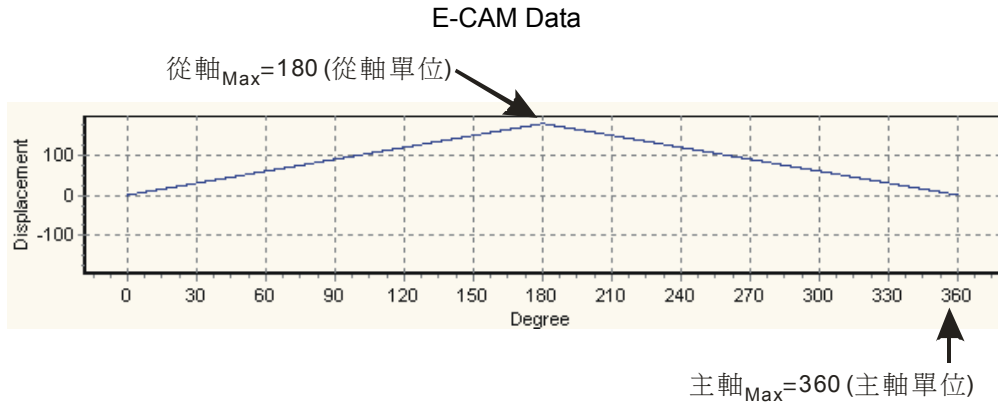


9 电子凸轮

9.2.3 电子凸轮启动/停止

9.2.3.1 周期式电子凸轮启动/停止

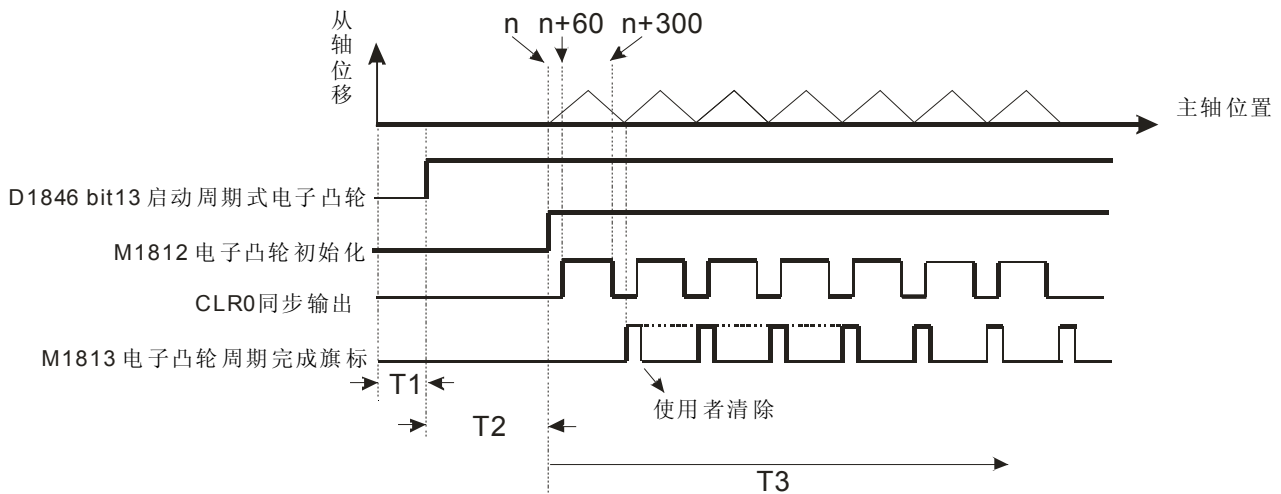
周期式电子凸轮指的是在主轴不断前进的同时，凸轮轴会依 E-CAM Data(E-CAM Data)实现相对应的位置，但 E-CAM Data 只定义了一个周期的数据，所以在此模式下主/从轴的位置关系就是 E-CAM Data 的不断重复延伸。



■ 周期式电子凸轮启动

以下周期式电子凸轮启动顺序。

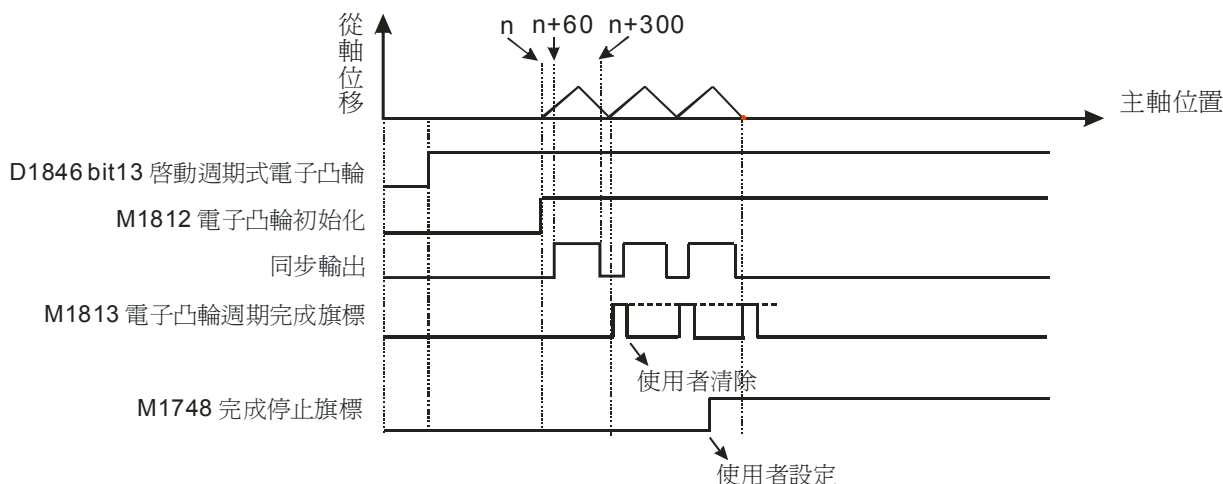
1. 在时间 T1 时，D1846 Bit13 = ON，启动周期式电子凸轮。
2. 经过时间 T2 后，PM 主动设定 M1812(电子凸轮初始化完成标志)。
3. 在时间 T3 期间，电子凸轮初始化完成开始周期性动作。从轴开始依照 E-CAM Data 的位置关系跟随主轴的动作，同时 CLR0 根据同步点范围输出
4. 当一个周期运行完毕，电子凸轮周期完成标志 M1813 = ON，使用者自行清除完成标志后便继续下一个周期运行开始。



■ 周期式电子凸轮停止模式

以下周期式电子凸轮停止顺序：

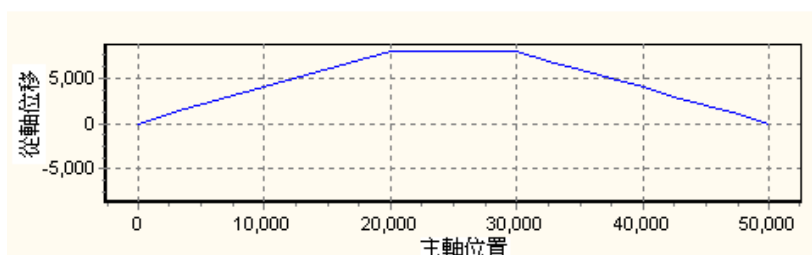
1. 当周期式电子凸轮模式允许信号 D1846 Bit13 = OFF 时，电子凸轮 X 轴就立即停止运作。
2. 当周期式电子凸轮运作时，系统接受到完成停止标志 M1748= ON，周期式电子凸轮会继续运作完 E-CAM Data，从动轴才停止运作，如下图所示。



■ 范例说明

【功能说明】

下图所示为电子凸轮数据，其中主轴长度为 50000，输出单位为马达单位；同步点范围为 20000 ~ 30000，当运转至同步区中 CLRO 信号会触发，可用来做为控制信号。建立电子凸轮数据步骤请参考章节 9.4。硬件线路 Y 轴输出脉冲(FP1+、FP-、RP+、RP-)，接到 X 轴手摇轮输入端(A0+、A0-、B0+、B0-)，表示主轴输入端位置是接收 Y 轴输出脉冲。



此范例是用软件 PMSoft 对 E-CAM Chart-0 设定好 E-CAM Data，详细建立电子凸轮曲线步骤请参考 9.4 章节。硬件线路 Y 轴输出脉冲(FP1+、FP-、RP+、RP-)，接到 X 轴手摇轮输入端(A0+、A0-、B0+、B0-)，表示主轴输入端位置是接收 Y 轴脉冲输出为输入。

【操作步骤】

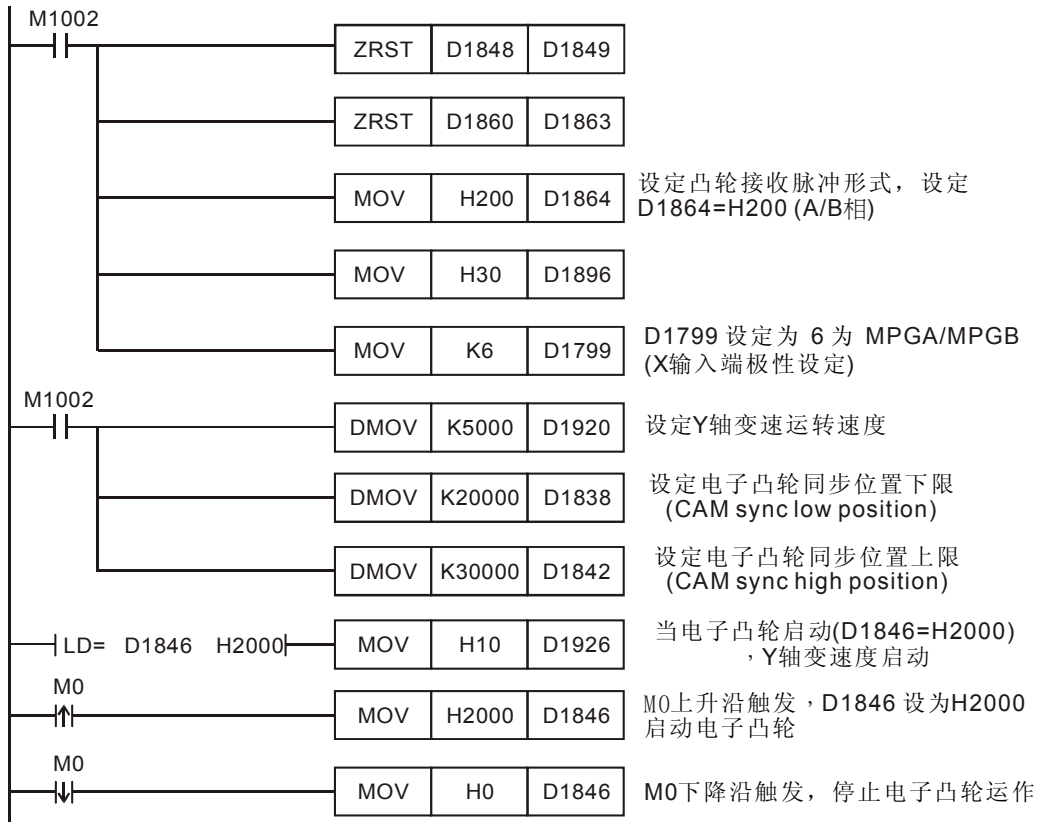
1. 程序执行时，首先执行设定特殊寄存器，设定的参数如下：
 - (1) D1848、D1849 与 D1862、D1863、D1868 内容清除为 0
 - (2) 主轴接收脉冲形式为 A/B 相(D1864=H200)
 - (3) Y 轴输出脉冲形式为 A/B 相(H30)

9 电子凸轮

- (4) 主轴输入端极性设定为 MPGA/MPGB(D1799 = 6)
- (5) 设定 Y 轴移动速度(此范例是以 Y 轴变速度输入脉冲给主轴输入端)
- (6) 设定电子凸轮同步位置上下限 D1839, D1838 = 20000 与 D1843, D1842 = 30000; 一个电子凸轮周期为 50000 脉冲, 主轴位置在 20000 ~ 30000 脉冲时(可由 D1862 监控), CLR0 状态为 ON, 但 DVP-PM 主机外壳上 X 轴的 CLR LED 不会显示, 监控方式可由 CLR0- 接 0V, CLR0+ 接 Xn 输入点, S/S2 接 24V, 观察 Xn 为 ON/OFF 即为 CLR0 状态)

2. Set M0 启动电子凸轮, 当电子凸轮启动时, Y 轴开始执行变速度运动; 主轴接收 Y 轴变速度输入脉冲, 从轴依照电子凸轮曲线输出脉冲, 且当主轴位置在每个周期 20000 ~ 30000 时, CLR0 为持续 ON 状态。
3. RST M0 电子凸轮停止运作。

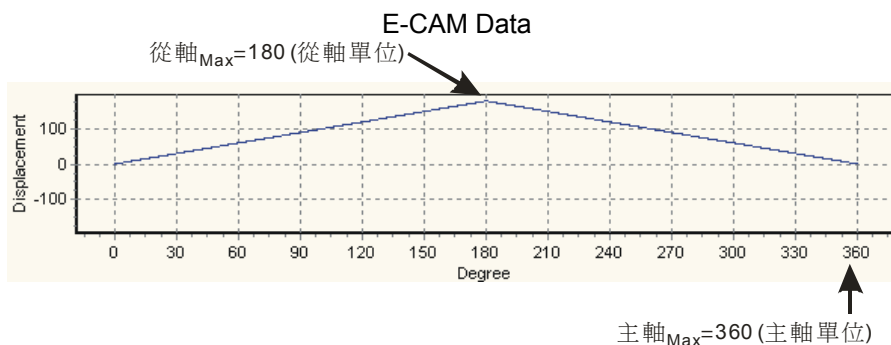
【PM 程序】



9.2.3.2 非周期式电子凸轮启动/停止

非周期式电子凸轮指的是在主轴不断前进的同时，凸轮轴开始依 E-CAM Data(CAM Data)实现相对应的位置的动作时机，是在凸轮启动信号输入之后。不同于周期式电子凸轮，在此模式下主 / 从轴的位置关系实际上只动作一个周期，即 E-CAM Data 只动作一次。

使用非周期式电子凸轮外部启动信号 START0，要设定 M1035 为 ON，表示将 STOP0/START0 当外部输入点使用，且 M1746 要设为 OFF，表示启动非周期式电子凸轮是接收 START0 信号，而不是 PG0。

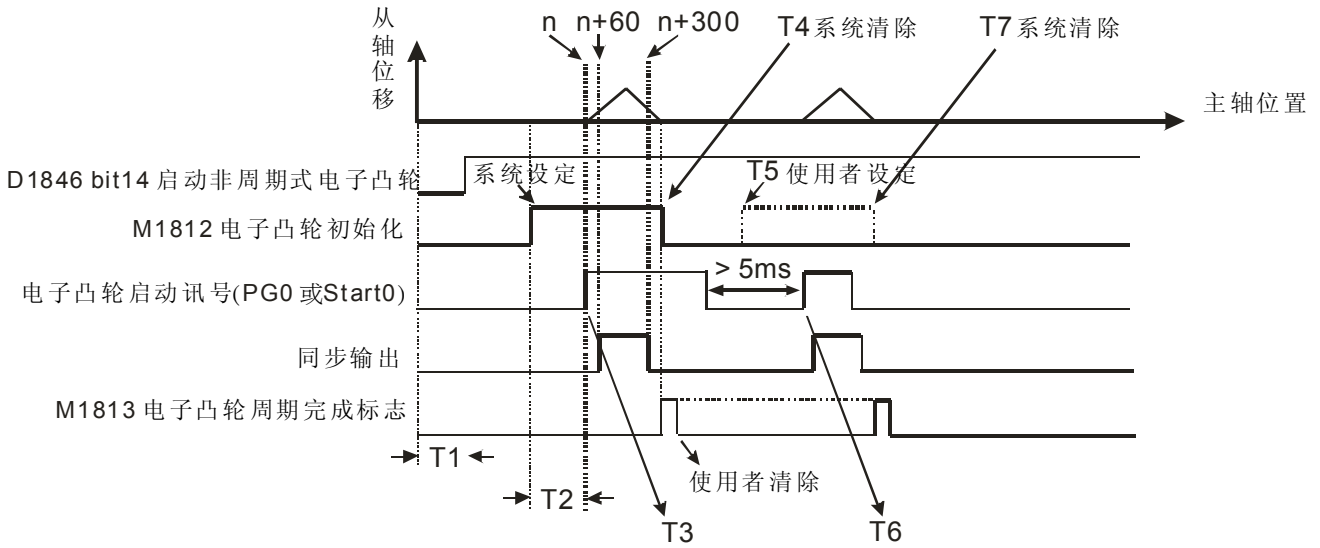


■ 非周期式电子凸轮启动

以下非周期式电子凸轮停止顺序：

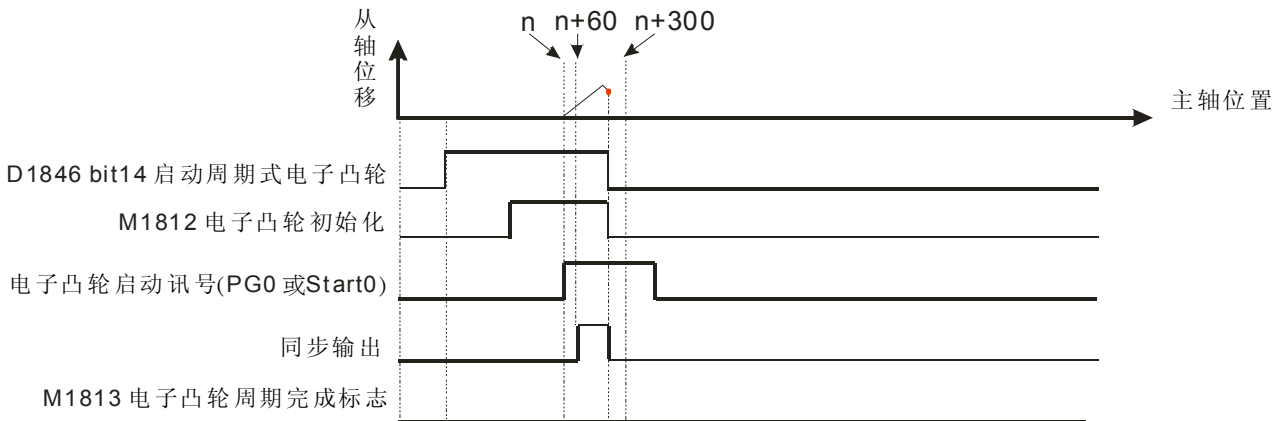
1. 在时间 T1 时，D1846 Bit14 = ON，启动了非周期式电子凸轮。
2. 并经过时间 T2 的计算时间后，PM 主动设定 M1812 = ON，（非周期式电子凸轮初始化完成），此时从轴还不会跟随主轴的动作。
3. 在时间 T3 电子凸轮启动信号 ON (M1746 = ON 时，PG0 为电子凸轮启动信号。M1746 = OFF 且 M1035 = ON 时，START0 为电子凸轮启动信号)，从轴开始依照 E-CAM Data 的位置关系跟随主轴的动作一个周期。(M1035 = ON，将 STOP0/START0 当外部 IO 用)
4. 周期完成后在时间 T4，PM 主动清除 M1812 = ON 的状态，同时使用者也可以利用 M1813 来得知是否完成。
5. 时间 T5 使用者程序可以选择是否再设定 M1812 = ON，以供下次完成判断之用。
6. 时间 T6/T7 将 T3~T4 的动作重复一次，要注意的是凸轮启动信号上升缘的间隔时间需要在 5ms 以上。
7. CLR0(Sync out)在凸轮动作的期间也会开始依据同步点范围输出。

9 电子凸轮

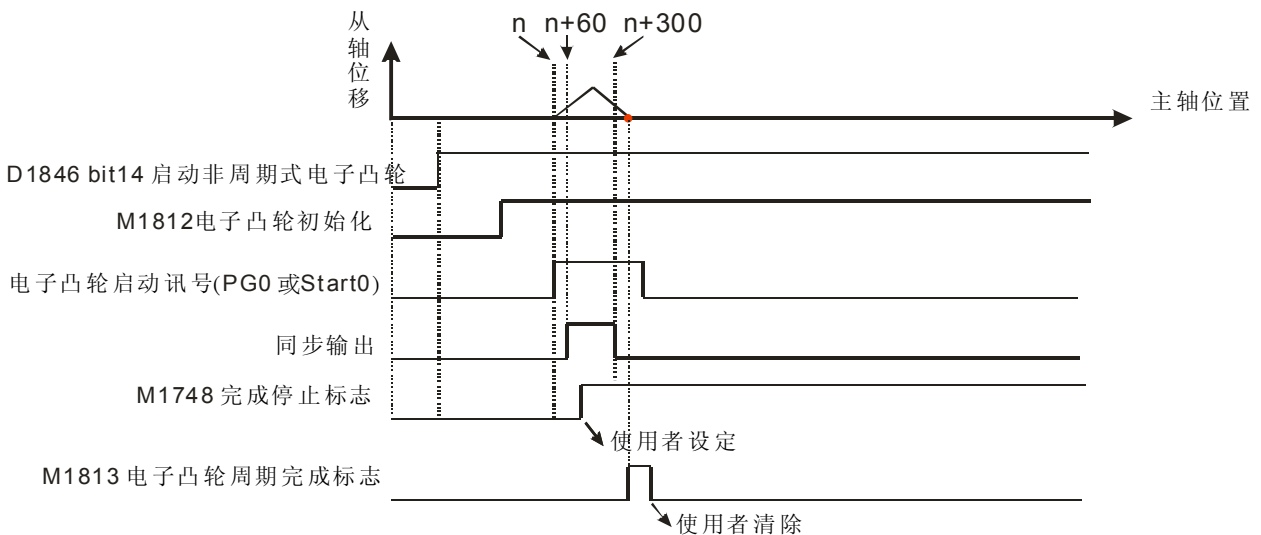


非周期式电子凸轮停止模式

1. 当非周期式电子凸轮模式允许信号 D1846 Bit14= OFF 时，电子凸轮从动轴就立即停止运作，如下图所示。



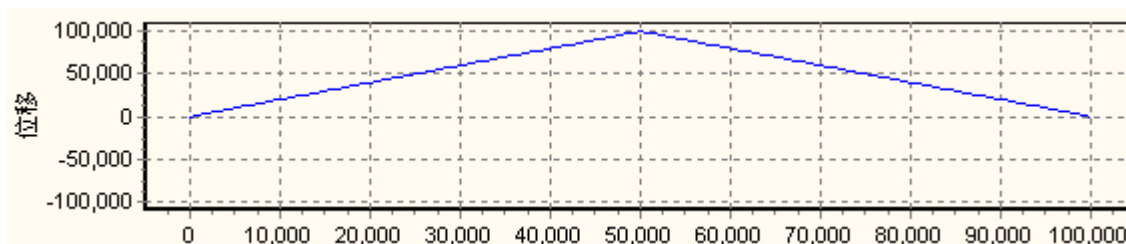
2. 当非周期式电子凸轮运作时，系统接受到完成停止标志 M1748= ON，非周期式电子凸轮会继续运作完 E-CAM Data，从动轴才停止运作，如下图所示。



■ 范例说明

【功能说明】

下图所示为电子凸轮运作 E-CAM Data (主轴长 0 ~ 100000 为一个周期), 其输出是马达单位, 当外部信号 START0 上缘触发, 执行连续三个 E-CAM Data (D1832=2), 等 START0 上缘再触发, 再执行连续三个 E-CAM Data, 以此类推。

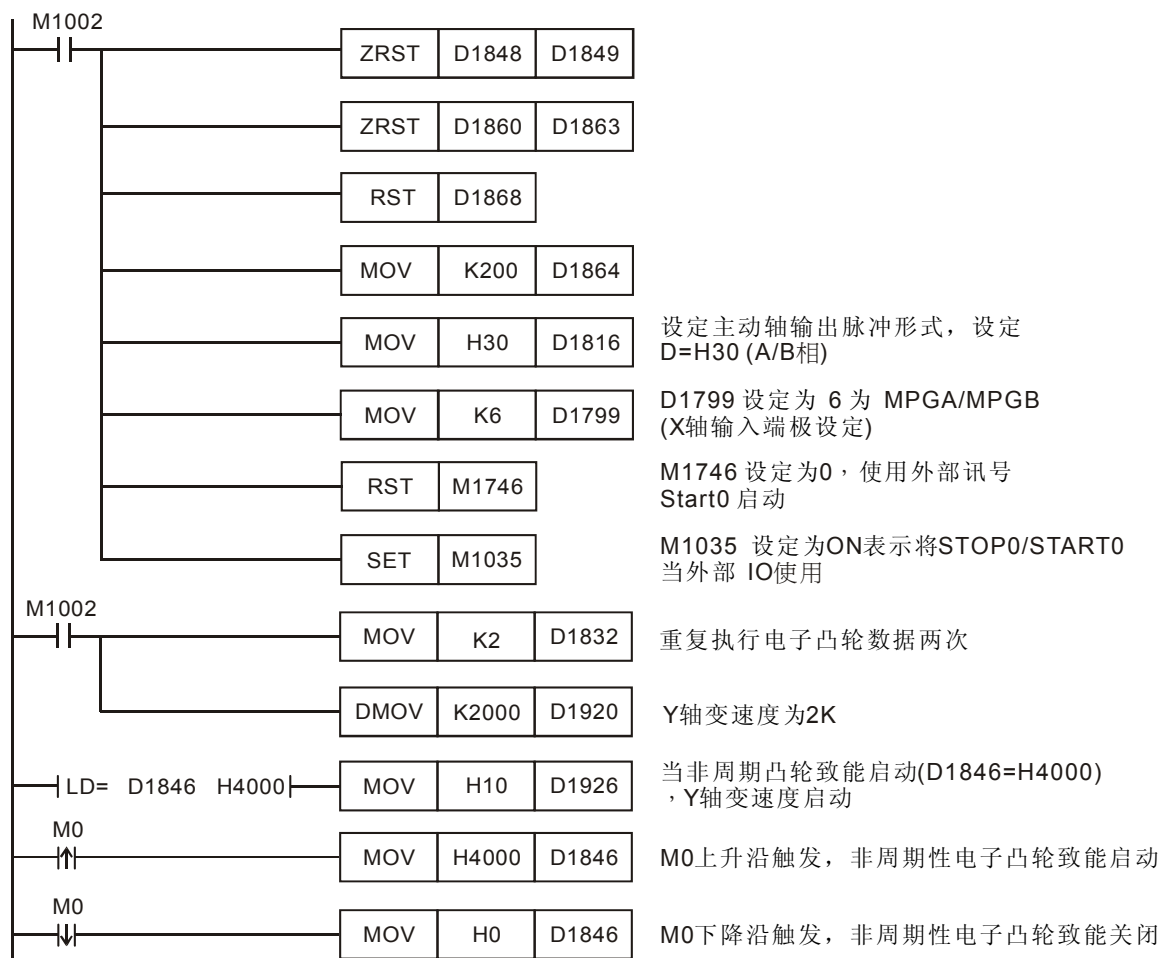


此范例是用软件 PMSof 对 E-CAM Chart-0, 详细建立电子凸轮曲线步骤请参考 9.4 章节。硬件线路 Y 轴输出脉冲(FP1+、FP-、RP+、RP-), 接到 X 轴手摇轮输入端(A0+、A0-、B0+、B0-), 表示主轴输入端位置是接收 Y 轴脉冲输出为输入。

【操作步骤】

- 程序执行时, 首先执行设定特殊寄存器, 设定的参数如下:
 - (1) D1848、D1849 与 D1862、D1863、D1868 内容清除为 0
 - (2) 主轴接收脉冲形式为 A/B 相(D1864=H200)
 - (3) Y 轴输出脉冲形式为 A/B 相(H30)
 - (4) 主轴输入端设为 MPGA0/MPGB0 (D1799 = 6)
 - (5) 设定 M1035 = ON, 表示将 STOP0/START0 当外部 IO 使用
 - (6) 设定 M1746 = OFF, 使用外部信号 START0 启动非周期性凸轮
 - (7) 设定 D1832=2(重复执行 E-CAM Data 两次)
 - (8) Y 轴移动速度(此范例是以 Y 轴变速度输入脉冲给主轴输入端)
- Set M0 启动非周期性电子凸轮; 当 D1846=H4000, Y 轴启动变速度运动, 此时 X 轴尚未输出。
- 外部信号 START0 触发, X 轴随着电子凸轮曲线输出; 执行 3 个周期停止输出后。
- Rst M0, 关闭电子凸轮模式; 若在电子凸轮运行时 Rst M0, 则 X 轴立即停止输出。

【PM 程序】

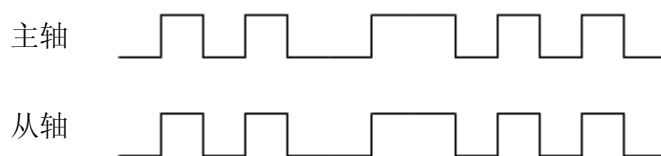


9.3 电子凸轮功能寄存器

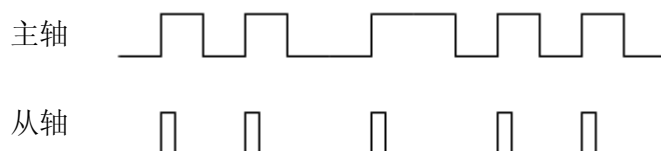
■ 电子凸轮最高输出频率

1. D1841, D1840 (VI) 电子凸轮最高输出频率设定；未设定此寄存器时，将无法正常输出。
2. M1749: 电子凸轮最高输出频率控制。

M1749 = OFF 时，从轴电子凸轮输出频率会跟随主轴的频率，但最高输出频率为 300kHz。



M1749 = ON 时，从轴电子凸轮最高输出频率由 VI 指定。



■ 周期式电子凸轮模式允许信号/完成

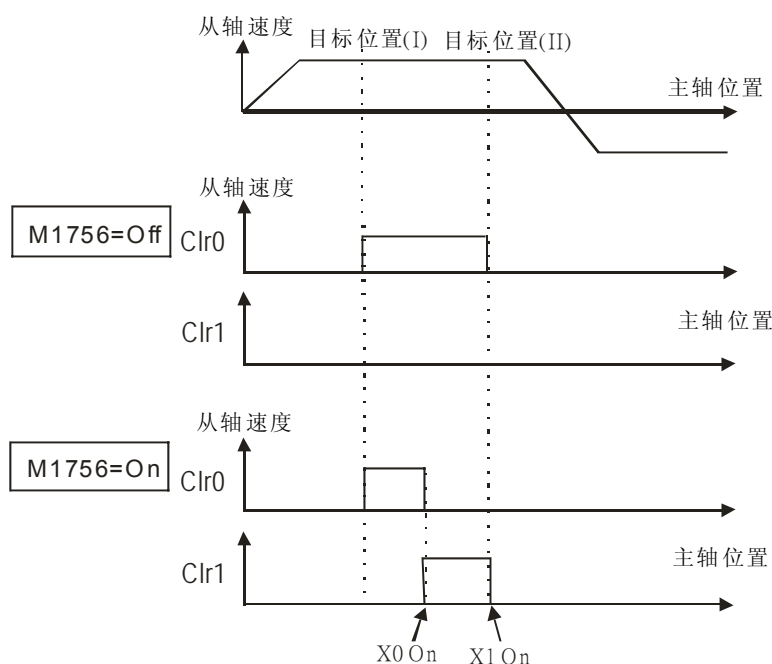
3. **D1846**: 周期式电子凸轮模式允许。当 **D1846** 的 **Bit13** 设为 **ON** 时, 启动周期式电子凸轮, 设定为 **OFF** 时停止周期式电子凸轮。
4. **M1813**: 周期式电子凸轮完成标志。当周期式电子凸轮执行完毕后, 此标志会自动设为 **ON**; 欲重新启动周期式电子凸轮, 则使用者需先清除此标志状态。

■ 非周期式电子凸轮模式允许

当 **D1846** 的 **Bit14** 设为 **ON** 时, 启动非周期式电子凸轮, 设定为 **OFF** 时停止非周期式电子凸轮。

■ 电子凸轮同步位置上/下限

1. **M1756** 设为 **ON**, 启动同步输出参考点功能。其动作顺序为当现在位置处于目标位置 **P(I)** 与目标位置 **P(II)** 之间时 **CLR0** 信号为 **ON**, 当外部信号 **X0** 触发, **CLR0** 信号变为 **OFF**, **CLR1** 信号为 **ON**, 直到外部信号 **X1** 触发, **CLR** 信号为 **OFF**, 顺序如下图所示。
2. 若 **M1756** 为 **OFF**, 则照电子凸轮的同步位置上/下限设定, 当目标位置 **(PI)** \leq 主轴位置 \leq 目标位置 **(PII)** 时, 同步信号 **CLR0 = ON**。



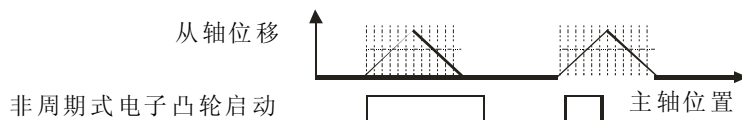
■ 电子凸轮初始化完成标志

当以电子凸轮允许信号启动电子凸轮后, **DVP-PM** 会先计算电子凸轮相关数据, 初始化完毕后自动设定 **M1812** 为 **ON**, 使用者需自行清除此标志状态。

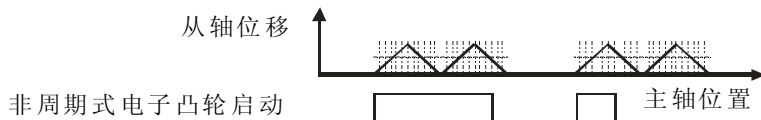
■ 非周期式电子凸轮重复执行寄存器

此功能由 **D1832** 寄存器控制重复执行 **E-CAM Data** 周期次数, 不过当 **D1832** 的值超过 **H8000** (**Bit15 = 1**) 时, 将变为周期式电子凸轮。当 **D1832** 数据等于零时, 不重复执行 **E-CAM Data**, 若 **D1832 = 1** 时, 将重复执行一次, 以此类推。

D1832 = 0

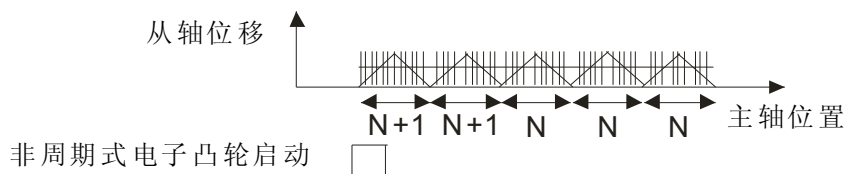


D1832 = 1

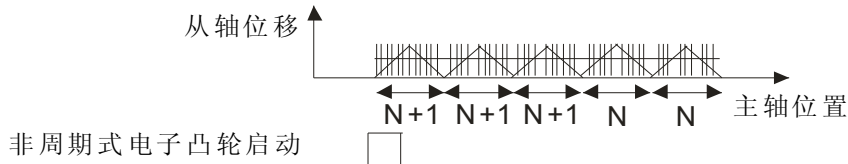


■ 非周期式电子凸轮余数分配寄存器

主轴脉冲总数和凸轮执行的总脉冲数无法整除时，藉由设定此寄存器来平均无法整除的脉冲数，设定 D1833 指定需多执行一个 pulse 的周期次数。配合 D1832 的凸轮执行周期重复次数，以下图为例，当 D1832 = 4 及 D1833 = 2 时，非周期性凸轮的执行次数总共为 5 次，且第一次和第二次周期执行时主轴会再走一个脉冲。(注：D1833 设定值不能大于 D1832 设定值)。



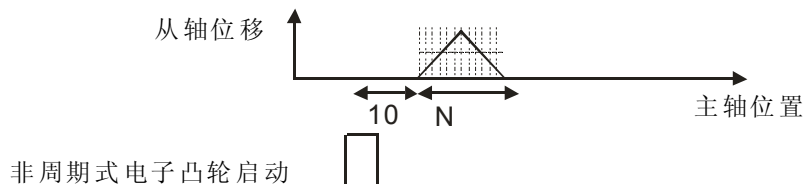
以此类推，当 D1832 = 4 及 D1833 = 3 时，在第一组、第二组和第三组的 E-CAM Data 执行完，再执行下一组 E-CAM Data 之前，主轴皆会再走一个脉冲。



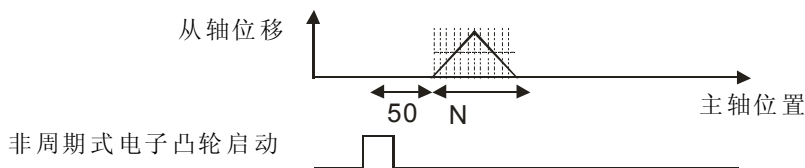
■ 非周期式电子凸轮启动延迟脉冲设定

执行非周期电子凸轮时，接受一个凸轮启动信号，若没有要立即执行 E-CAM Data，而要延迟主轴转动几个派波，才运作 E-CAM Data，此时可在 D1834 寄存器设定延迟脉冲数，下图可清楚看到，当系统接受一个凸轮启动信号，主轴转动了 D1834 设定的脉冲数，才会立即执行 E-CAM Data。

D1834 = 10



D1834 = 50



■ 非周期式电子凸轮启动标志

1. 当 M1746 设 OFF 时，非周期式电子凸轮启动允许信号为外部信号 START0，若 M1746 设 ON 时，非周期式电子凸轮启动允许信号为外部信号 PG0。
2. 使用非周期式电子凸轮外部启动信号 START0，要设定 M1035 为 ON，表示将 STOP0/START0 当外部 IO 使用，且 M1746 要设为 OFF，表示启动非周期式电子凸轮是接收 START0 信号，而不是 PG0。

■ 电子凸轮模式完成停止信号

当 M1748 设 ON 时，电子凸轮完成 E-CAM Data 则立即停止电子凸轮运作。

■ 主轴/从轴位置与频率

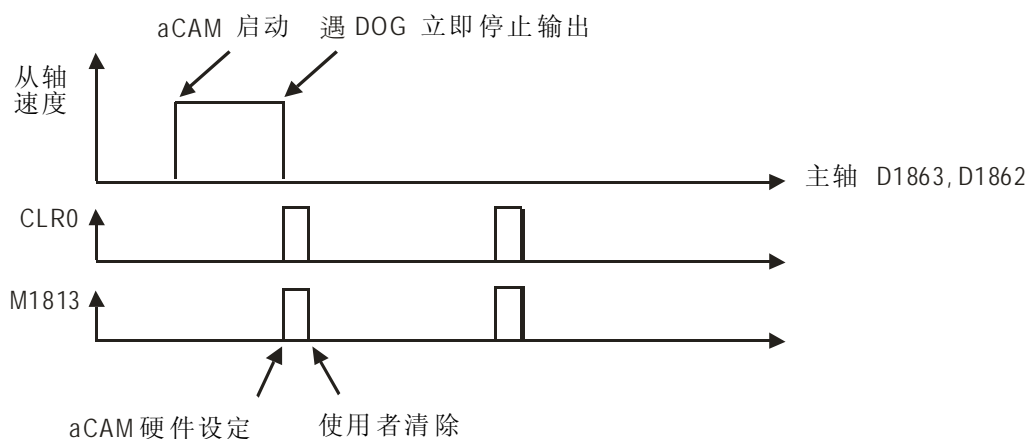
手摇轮脉冲个数/主轴位置 D1863, 1862, 手摇轮频率/主轴频率 D1861, 1860, 现在位置 CP/从轴位置 D1849, 1848。

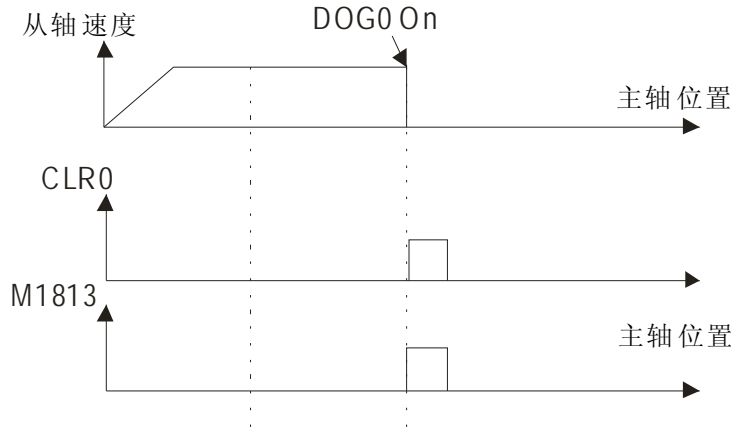
■ 电子凸轮表(E-CAM Chart)选择

1. 每一个电子凸轮图表内包含一组 E-CAM Data，PM 提供 3 组电子凸轮表(E-CAM Chart-0~E-CAM Chart-2)供使用者来设定 E-CAM Data(E-CAM Data0~E-CAM Data2)。
2. 在 E-CAM 启动前由 D1868 (电子凸轮表指定) 来选择要用的 E-CAM Data0~2:
3. E-CAM Chart-0: D1868 = 0, E-CAM Chart-1: D1868 = H'8001, E-CAM Chart-2: D1868 = H'8002。

■ 非周期式凸轮遇外部 Dog 立即停止

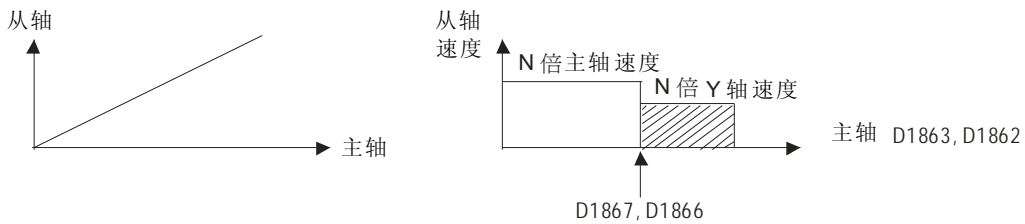
于非周期电子凸轮模式下，设定 M1752，外部 DOG 信号触发时立即停止凸轮，搭配 M1755 做裁切同时每次停止于 DOG 开关上，当 M1752 允许时除了遇 DOG 会停止外，PM 的 CLR0 会同时输出用以清除伺服误差，使用者需在 M1813 周期结束标志为 ON 时清除 M1813 才能清除 CLR0 做下次的裁切。





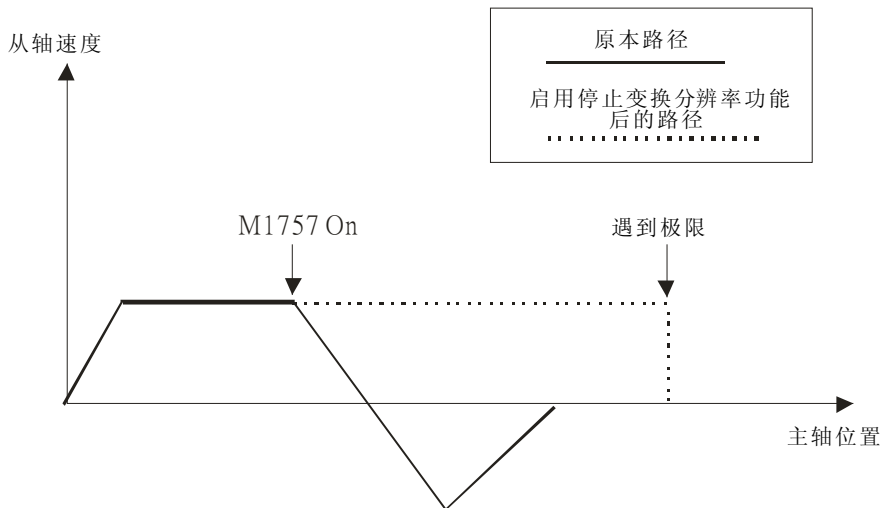
■ 非周期式凸轮切换主轴来源

设定 M1755 非周期式凸轮可于运行的过程中切换主轴的来源为内部 Y 轴，当主轴计数值 (D1863,D1862) 大于非周期式凸轮主轴切换范围 (D1866,D1867) 时将自动切换，而非周期式凸轮周期结束后恢复来源为 MPG0 (M1909, M1910 虚轴控制不可使用)。主要用途在于控制当 DOG 信号触发时从轴停止输出时的速度相同，不会因外部 Sensor 的特性及主轴速度而停在不同位置。



■ 停止凸轮变换解析点

当 M1757 设为 ON 时，从轴会保持现在速度，直到遇到左右极限后停止输出。



9.4 电子凸轮数据(E-CAM Data) 建立

9.4.1 使用 PMsoft CAM Chart 建立电子凸轮数据

定义主轴/从轴（凸轮轴）之间的关系，称之为电子凸轮数据(E-CAM Data)。

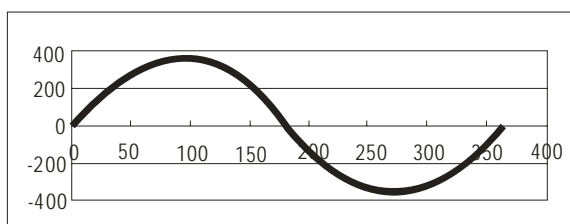
在数据的输入上 E-CAM Data 有两种方法表述：

- 方法 1：采用两者的函数关系。
- 方法 2：采用 X、Y 的点对点关系。

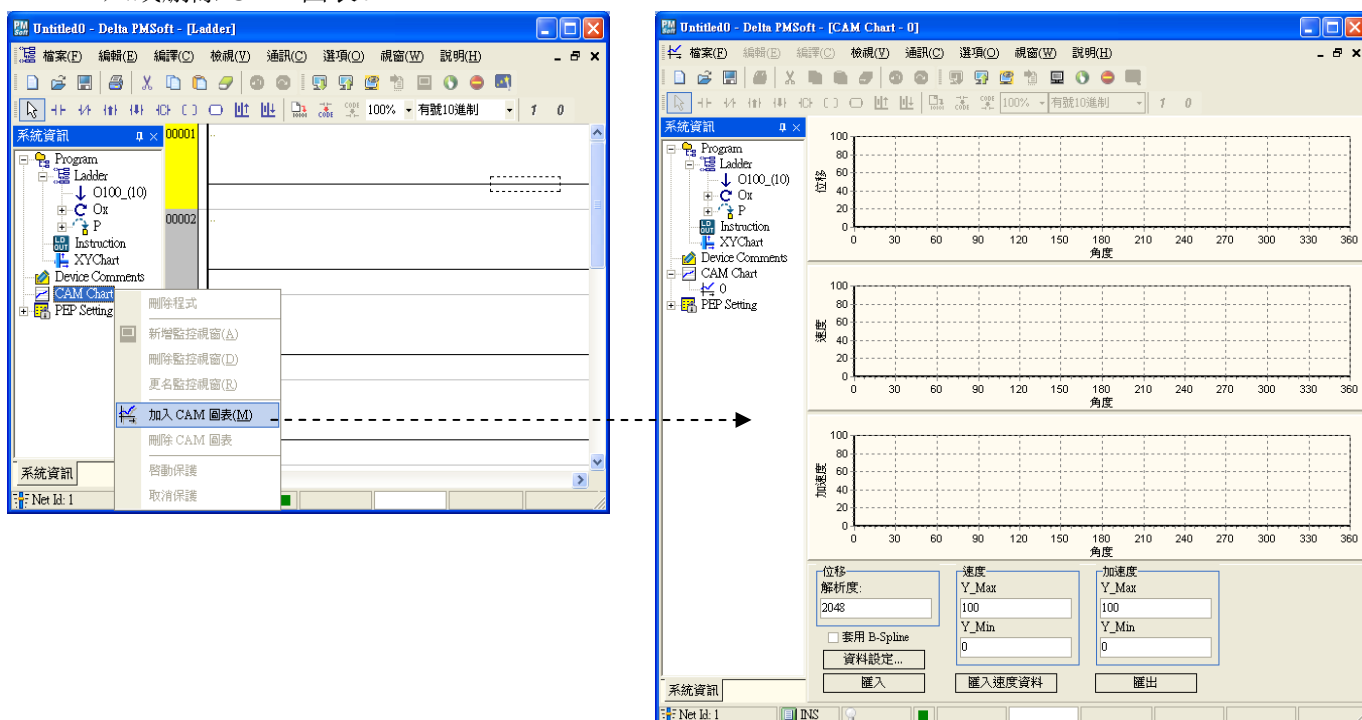
E-CAM Data 的获取也有两种途径：

- 途径 1：根据主从轴的标准函数关系。
- 途径 2：根据实际工作中测量到的点与点之间的对应关系。

E-CAM Data 可以定义多个 CAM 曲线。关系确定后，根据主轴的位置，就能得到从轴的位置。例如正弦信号的 E-CAM Data 如下图所示：



E-CAM Data 在 PMSoft 中称为 CAM Chart，在系统信息列中选择 CAM Chart 按鼠标右键可以加入或删除 CAM 图表。



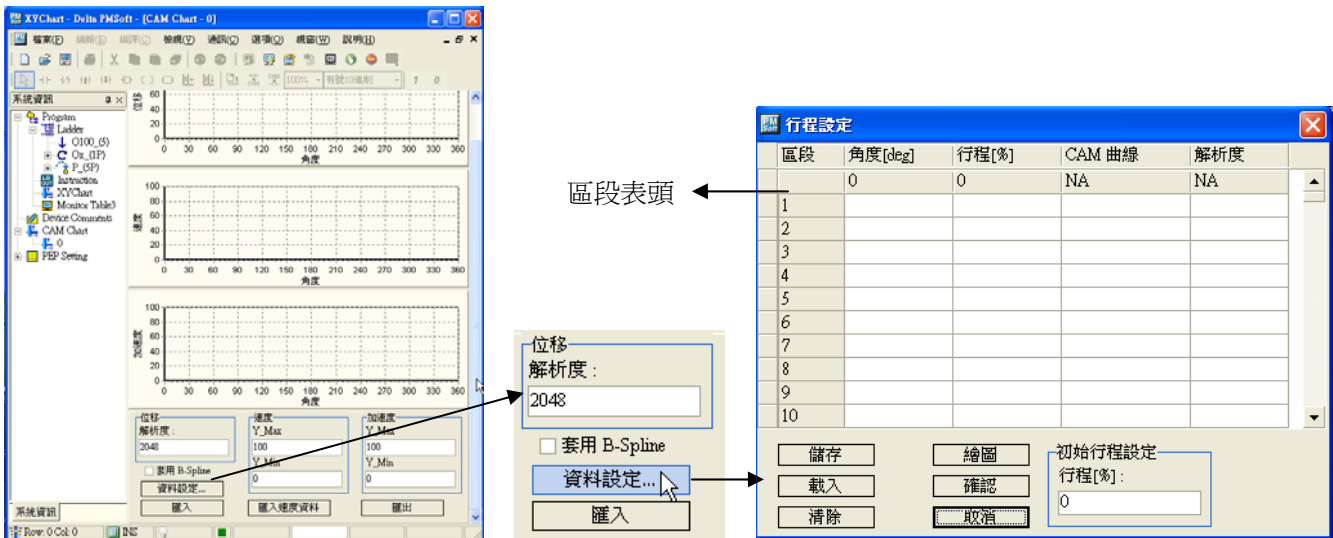
CAM 图表上主要分为 4 个部份，分别为主 / 从轴相对位置、主 / 从轴相对速度、主 / 从轴相对加速度、及最下方的数据设定。前 3 部份用来显示使用者所设定的 E-CAM Data，其中横轴的部份皆为主轴的位置，纵轴分别为从轴的位置，从轴跟主轴的速度比，从轴跟主轴的加速度比。数据设定区介绍如下：

1. 位移分辨率：提供使用者设定这个 E-CAM Data 总共所占用的数据点数，可设定范围从 10~2048 点。其 1 点 E-CAM Data 占用 10 Steps 的使用者程序区(共 64k Steps)。
2. 速度：自动计算并显示的速度最高 / 最低值
3. 加速度：自动计算并显示的加速度最高 / 最低值
4. 数据设定：通过函数方式来描述主 / 从轴的位移变化

5. 汇入：通过点对点的方式来描述主 / 从轴的位移变化
6. 汇出：将主 / 从轴的变化关系以点对点的方式输出至档案
7. 汇入速度数据：通过点对点的方式来描述主 / 从轴的速度变化

9.4.1.1 以函数方式来描述主从轴的位移变化

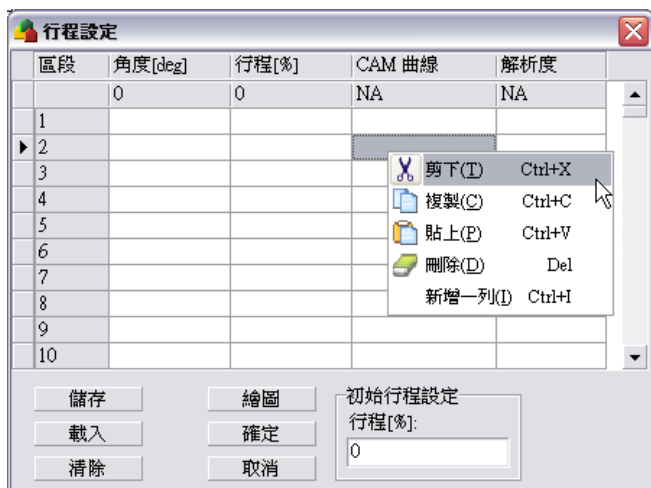
在数据设定区里选择“数据设定”会出现「行程设定」的窗口，这个窗口可以让使用者将整个凸轮的曲线以函数的方式描述出来，而不需要以点对点的方式描述，如下图所示。目前 DVP-PM 提供了 5 种凸轮曲线供使用者选择，分别是：CONSt Speed、CONSt Acc.、Single Hypot.、Cycloid、与 BSpline。



在行程设定的窗口是由区段所组成，每个区段提供使用者设定一段凸轮曲线，再由数个区段组成整个凸轮曲线。每个区段的组成有角度、行程、凸轮曲线、及分辨率，兹说明如下：

- 角度：主轴角度 / 位移，主轴为位移必为大于 0 的数值且为递增。
- 行程：从轴角度 / 位移，从轴为位移可为正或为负。
- 凸轮曲线：目前区段所使用的函数
- 分辨率：目前区段所使用的数据点数，整个 E-CAM Data 可设定范围从 10~2048 点。其 1 点 E-CAM Data 占用 10 Steps 的使用者程序区(共 64k Steps)。如果没有设定则 PMSOft 会由剩余的总点数均分。分辨率设定依设备需求来设定，分辨率越高设备运作较平滑，但 E-CAM Data 占的容量越大。

整个行程设定窗口操作类似于 Excel，使用者可以对每个 Cell 或是整个区段进行复制、贴上、剪下等编辑动作，鼠标的右键可以显示快捷菜单。要对区段进行编辑动作时需选取区段标题。

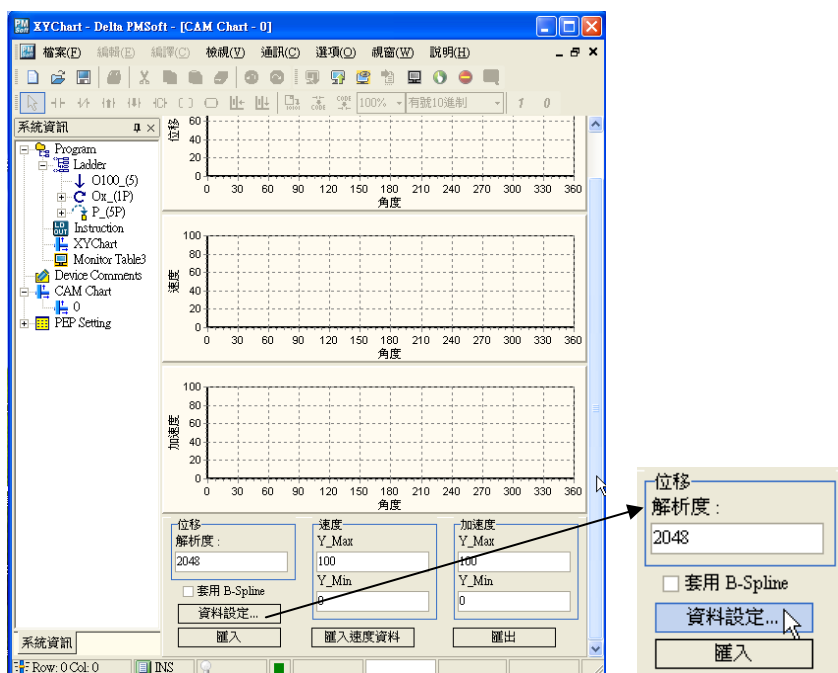


按钮功能说明：

- 储存：将目前区段的设定暂存
- 载入：加载暂存的区段数据
- 清除：清除所有的区段数据
- 绘图：将区段的数据编译产生 E-CAM Data，且绘制在 E-CAM Chart 上。
- 确认：将区段的数据编译产生 E-CAM Data，且绘制在 E-CAM Chart 上，并关闭行程设定窗口。
- 取消：关闭行程设定窗口

9.4.1.2 以点对点方式来描述主从轴的位移变化

- 另一种描述主 / 从轴位移变化的方式，是直接以点对点的数据加到 E-CAM Data 里，如下图所示，一个 E-CAM Chart 最多可以输入 2048 点的 E-CAM Data。



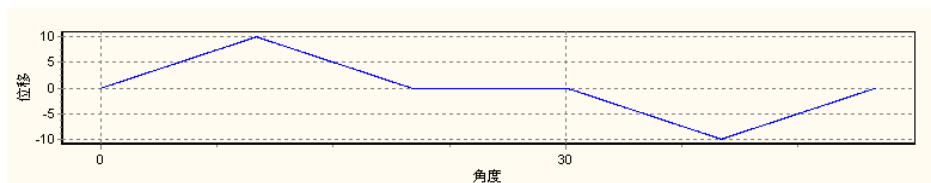
按下“汇出”后，PMSOft 会将目前在 E-CAM Chart 的位移关系图、速度关系图及加速度关系图中的 E-CAM Data 储存到 PMSOft 安装目录下 CAMData 的子目录，里面一共会有 3 个档案分别为：

- <PMSOft 安装目录>\CAMData\Data_S.txt：点对点的位移数据
- <PMSOft 安装目录>\CAMData\Data_V.txt：点对点的速度数据
- <PMSOft 安装目录>\CAMData\Data_A.txt：点对点的加速度数据

Data_S.txt、Data_V.txt、Data_A.txt，如下图所示。

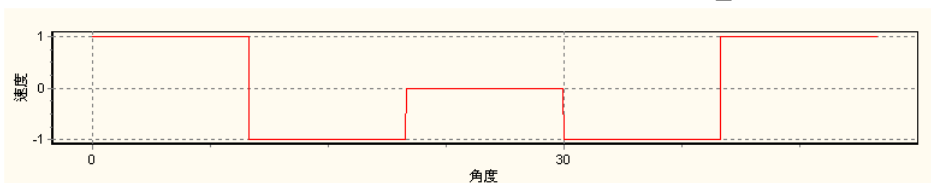
位移关系图

储存位置：<PMSOft 安装目录>\CAMData\Data_S.txt



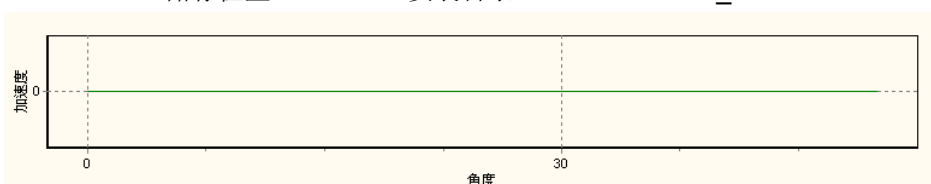
速度关系图

储存位置: <PMSoft 安装目录>\CAMData\Data_V.txt



加速度关系图

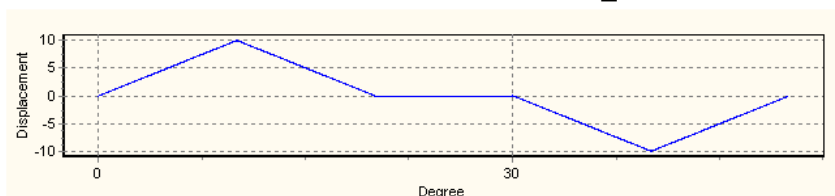
储存位置: <PMSoft 安装目录>\CAMData\Data_A.txt



在 PMSoft 安装目录下 CAMData 的子目录中的 Data_S.txt, 输入 X、Y 点对点的为移数据, 而后在 PMSoft 的 E-CAM Chart 中按下“汇入”后 PMSoft 会读取以存在 Data_S.txt 的内容, 并绘制在 E-CAM Chart 的位移关系图上, 同时速度关系图及加速度关系图会以 Data_S.txt 的数据计算出来一并绘制, 如下图所示。

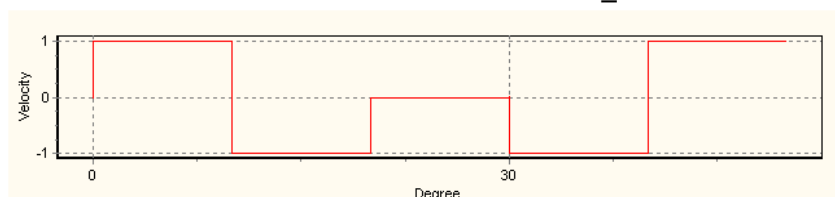
位移关系图

<PMSoft 安装目录>\CAMData\Data_S.txt



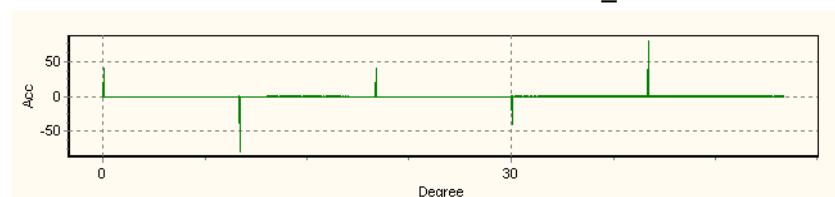
速度关系图

<PMSoft 安装目录>\CAMData\Data_V.txt



加速度关系图

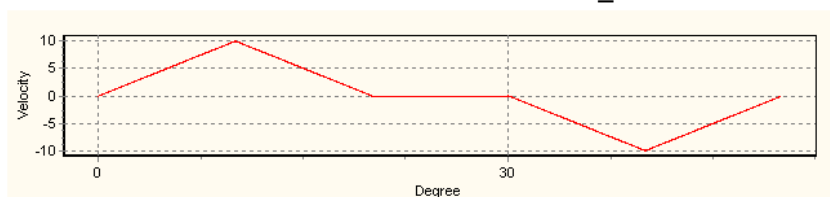
<PMSoft 安装目录>\CAMData\Data_A.txt



按下“汇入速度数据”后, PMSoft 会读取以存在 Data_S.txt 的内容, 并绘制在 E-CAM Chart 的速度关系图上, 同时位移关系图及加速度关系图会以 Data_S.txt 的数据计算出来一并绘制, 如下图所示。

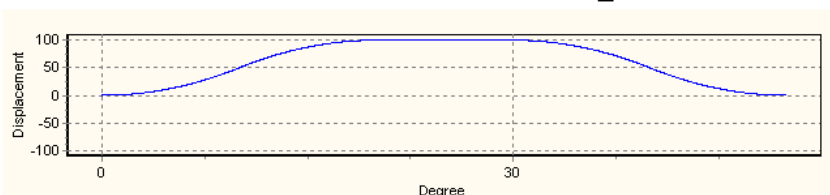
位移关系图

<PMSoft 安装目录>\CAMData\Data_S.txt



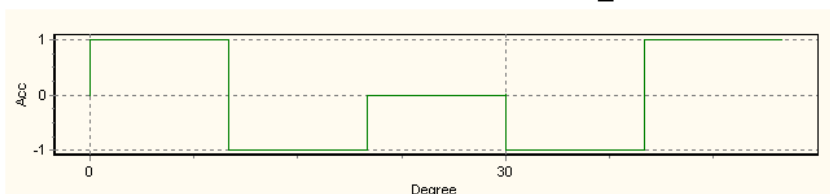
速度关系图

<PMSoft 安装目录>\CAMData\Data_V.txt



加速度关系图

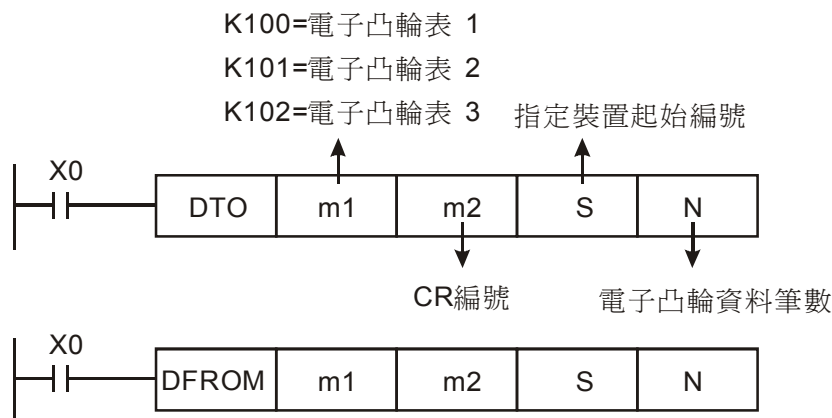
<PMSoft 安装目录>\CAMData\Data_A.txt



9.4.2 使用 DTO / DFROM 指令自动生成电子凸轮数据

20PM 规划 3 个特殊模块供电子凸轮表 0~2 用，其模块所在编号为 K100、K101、K102，透过 DTO / DFROM 二个指令，使用者可以在程序建立或修改电子凸轮数据。电子凸轮数据为浮点数值型态，所以使用动态写入的数据要先使用 DFLT 指令转成二进制浮点数。而使用 PMSoft ECAM Chart 产生的电子凸轮数据已转成浮点数值型态，在 PMSof 监控表监控电子凸轮数据时要使用浮点数值型态。

自动生成电子凸轮数据的指令格式如下图所示，在这里 CAM0~CAM2 被当成第 100~102 个特殊模块，即要对 CAM 读出或写入时 m1 要设定为 100，m2 代表写入 CR 之起始编号，S 为指定装置之起始编号，n 代表写入之数据笔数，每笔数据有两个 Word，设定值需为 2 的倍数。另外要注意 16 位的 TO/FROM 指令目前不支持。



■ 电子凸轮表控制寄存器 CR 一览表：

9 电子凸轮

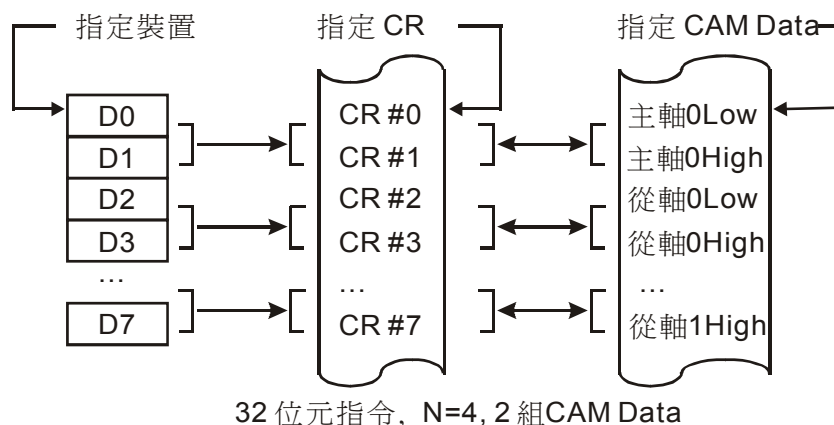
CR# 编号	寄存器功能	数据类型	寄存器深度
0~9999	建立单笔电子凸轮数据	DWord	1
10000	建立飞剪电子凸轮数据	DWord	10
10001	读回电子凸轮执行状态	DWord	7
10002	CAM Chart 动态变更	DWord	1
10003	平滑曲线	DWord	3
10005	读取捕捉模式下，读回电子凸轮执行状态	DWord	5

■ 各控制寄存器说明:

CR#0~9999: 单笔电子凸轮数据动态建立/变更

[说明]

20PM 内建 3 组电子凸轮表 E-CAM Chart 0~E-CAM Chart 2，在每个电子凸轮表可建立 10000 点数据，使用控制寄存器 CR#0~9999 设定。每两点数据为一组 E-CAM Data，由主轴位置和从轴位移组成。每笔数据的单位为 Dword，写入或读出 n 笔数据时，CR 数量等于 n 乘 2。变更单笔电子凸轮数据说明请参考章节 9.4.2.1。



CR#10000: 飞剪曲线自动生成

[说明]

20PM 提供自动建立飞剪曲线功能寄存器，让使用者设定实际飞剪机构相关参数与运转需求，即可生成与应用符合飞剪的电子凸轮曲线。建立飞剪曲线所需的参数如下表所示：

位置	数据	数据格式	说明
深度 1	主轴长度	整数	送料轴移动裁切长度，单位 mm
深度 2	从轴长度	整数	裁切轴长度(含刀具长度)，单位 mm
	同步区起始位置		启动同步区产生功能时，此寄存器用于设定同步区起始位置
深度 3	从轴同步长度	整数	主从轴之同步长度，单位 mm

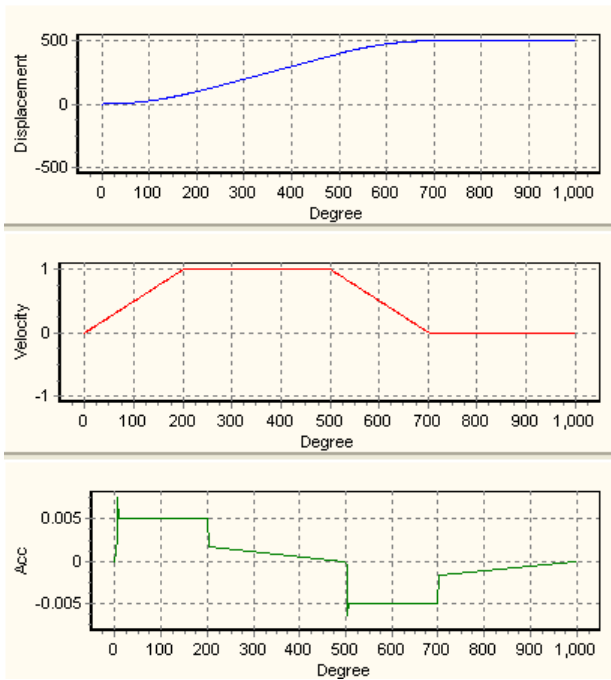
	同步区结束位置		启动同步区产生功能时，此寄存器用于设定同步区结束位置
深度 4	从轴同步倍率	浮点数	在同步区时主轴与从轴速度相等，同步倍率计算方法： $V_1 = V_2 \Rightarrow \frac{F_1 \times 3.14 \times D_1}{R_1} = \frac{F_2 \times 3.14 \times D_2}{R_2} \Rightarrow \frac{F_2}{F_1} = \frac{R_2 / D_2}{R_1 / D_1}$ 其中 $V_1(V_2)$ =主(从)轴速度($^{mm}/_{sec}$)、 $F_1(F_2)$ =主(从)轴速度(Hz)、 $D_1(D_2)$ =主(从)轴直径、 $R_1(R_2)$ =主(从)轴一圈脉冲数
深度 5	从轴最高倍率限制	浮点数	
深度 6	加速曲线 (Low word)	整数	E-CAM 加速曲线设定： 0:const speed 1:const Acc 2:SingleHypot 3:Cycloid
	CAM 曲线 (High word)	整数	起始、停止、以及不同同步区位置的各段飞剪曲线选择： 0:leftCAM 同步区位于前端曲线 1:midCAMall 2:midCAMbegin 起始曲线 3:midCAMend 结束曲线(选择此曲线,凸轮执行完毕后会 自动停止) 5: rightCAM 同步区位于后端曲线 备注：选择起始与结束曲线时，系统自动计算主轴长度等于倍率乘上 $1/2$ 从轴长度 生成设定 b[13]=1:产生同步区；启动此功能时，在深度 2 与深度 3 写入同步区的起始结束位置。 b[14]=1:仅允许单笔数据动态变更 b[15]=1:接续前次数据
深度 7	飞剪曲线生成结果	整数	显示飞剪凸轮曲线生成结果(读回时显示)： 0: OK 1: 条件无法满足 2: CAM 长度不足

深度 6 分别设定加速曲线与 CAM 曲线，主要是设定飞剪凸轮加速曲线及 CAM 曲线，以下为各个加速曲线与 CAM 曲线图形，以便使用者了解曲线设定。

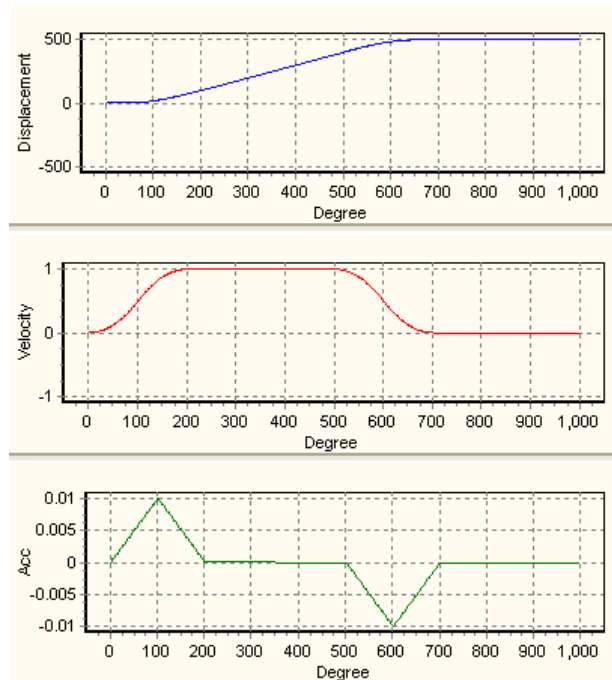
9 电子凸轮

加速曲线

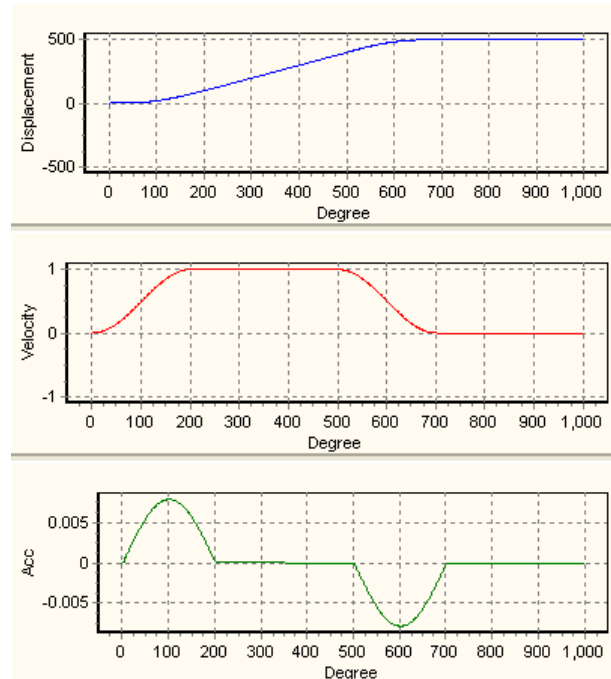
0: const speed



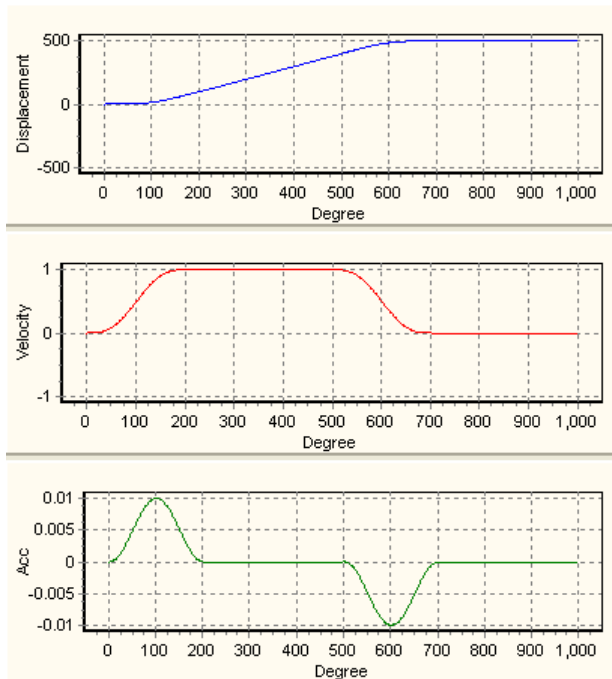
1: const Acc



2: SingleHypot

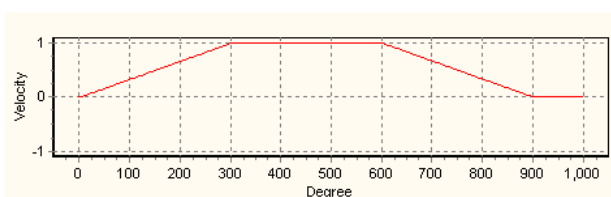


3: Cycloid

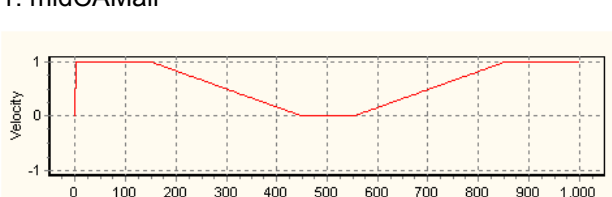


CAM 曲线

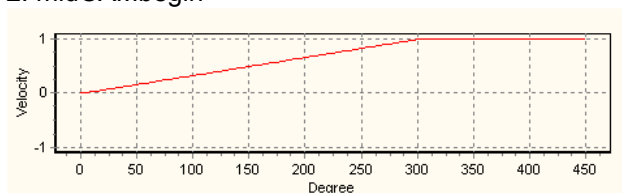
0: leftCAM



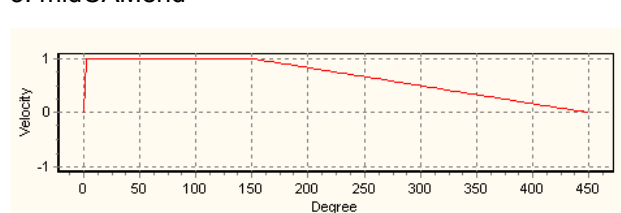
1: midCAMall



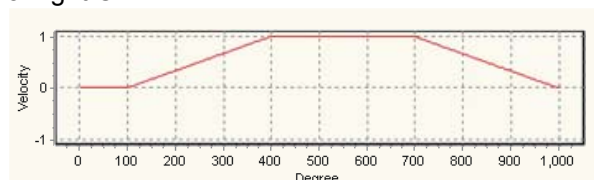
2: midCAMbegin



3: midCAMend



5: right CAM



详细建立飞剪曲线说明请参考章节 9.4.2.3。

针对飞剪应用中另外一种厚料裁切，其裁切材料厚度较大，在电子凸轮曲线设定上需另外考虑到裁切时的角度以及裁切中切刀的速度；20PM 另外提供厚料飞剪曲线自动生成寄存器，其设定格式如下所示：

位置	数据	数据格式	说明
深度 1	厚料飞剪格式设定	整数	用于设定厚料飞剪的格式 0: 均分异步区(当刀数能均分脉冲时选择此格式) 1: 指定异步区(当刀数无法均分脉冲时选择此格式) b[2]=1:指定同步区点数(于深度 12 位置设定) b[3]=1:接续前次凸轮数据 b[4]=1: 强制不平滑加减速以固定分辨率
深度 2	主轴一圈脉冲数	整数	主轴伺服输出一圈脉冲数，单位 Pulse
深度 3	主轴一周长度	整数	主轴周长，单位 mm
深度 4	从轴一周脉冲数	浮点数	从轴伺服输出一圈脉冲数，单位 Pulse
深度 5	从轴一周长度	浮点数	从轴周长，单位 mm
深度 6	从轴同步起始角度	整数	设定范围 180~360 度
深度 7	从轴同步结束角度	整数	设定范围 180~360 度
深度 8	主轴切长	整数	欲裁切的长度，单位 mm
深度 9	从轴刀数	整数	飞剪应用中所使用的裁刀数量。
深度 10	主轴同步区前长度	整数	起始角度模式设定为 1 时，此寄存器用于设定同步区前之长度
深度 11	从轴同步区前夹角	整数	起始角度模式设定为 1 时，此寄存器用于设定同步区前之角度
深度 12	同步区点数	整数	当格式设定中 b[2]=1 时，用此寄存器设定同步区点数

详细建立厚料飞剪曲线说明请参考章节 9.4.2.4

9 电子凸轮

CR#10001: 读取电子凸轮执行状态数据

[说明]

用于电子凸轮执行中，读取目前电子凸轮执行状态数据，仅能使用 DFROM 指令。读回的信息如下表所示：

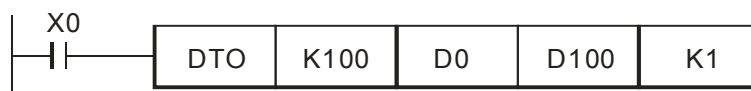
位置	数据	数据格式	说明
深度 1	目前主轴长度计数	整数	Dog 信号时 Rst
深度 2	主轴长度计数	整数	Dog 信号至 Dog 信号
深度 3	目前 CAM 曲线	整数	
深度 4	从轴计数	整数	每次 Dog 信号更新
深度 5	从轴计数差值	整数	Dog 信号 至 CAM 周期结束点
深度 6	主轴计数	整数	每次 CAMac(start0, pg0)更新
深度 7	主轴计数差值	整数	凸轮启动(start0, pg0)至 CAM 周期结束点

读取电子凸轮执行状态数据可应用在实际应用中刀轴误差判断，详细判断刀轴误差判断方法请参考章节 9.4.2.5。

CR#10002: CAM Chart 动态变更

[说明]

20PM 内建 3 组电子凸轮表 E-CAM Chart 0~E-CAM Chart 2，在程序执行中欲变更电子凸轮表时，可设定此寄存器指定下一个周期要执行的 E-CAM Chart。在使用上需注意 PM 中最少要有 2 组以上的 E-CAM Chart 可供变换。以下为使用范例，当 X0 = ON 时，指定 D0 = K10002 表示电子凸轮图表动态变更。由 D100 内容值(0, 1, 2)来选择 E-CAM Chart-0~E-CAM Chart-2。



CR#10003: 建立平滑曲线

[说明]

当原始数据点数少使得绘制的凸轮曲线不够平滑时，可使用建立平滑曲线功能。关于平滑曲线的使用与说明于章节 9.4.2.2 详细介绍。其平滑处理所需设定的参数如下表所示：

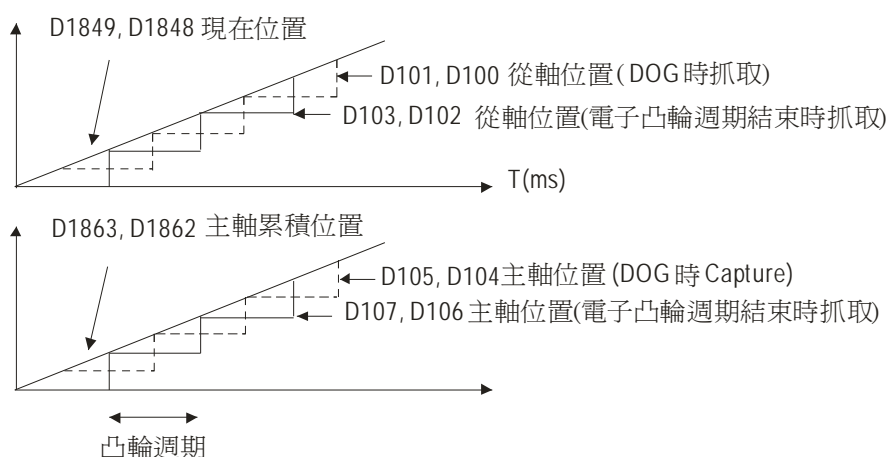
位置	数据	数据格式	说明
深度 1	主轴最大值	整数	
深度 2	原数据点数	整数	
	插值总点数	整数	
深度 3	平滑系数	整数	3: 3 次曲线 4: 4 次曲线 5: 5 次曲线

			6: 6 次曲线 Other: 3 次曲线
	摆线样式	整数	0: B 样条主轴均分 1: B 样条主轴不均分 2: C 样条(原数据点数最多 20 点) 3: CC 样条 4: 摆线 Cycloid

CR#10005: 捕捉模式，读取电子凸轮状态数据

[说明]

位置	数据	数据格式	说明
深度 1	从轴位置	整数	DOG 时捕捉
深度 2	从轴位置	整数	电子凸轮周期结束时捕捉
深度 3	主轴位置	整数	DOG 时捕捉
深度 4	主轴位置	浮点数	电子凸轮周期结束时捕捉
深度 5	主轴位置	浮点数	START0[m1746= OFF] or PG0[M1746= ON] 时捕捉



读取电子凸轮执行状态的捕捉数据配合特殊寄存器功能 M1757 停止凸轮变换解析点，可应用在实际追剪应用中检查切刀是否到位，详细方法请参考章节 9.4.2.6。

9.4.2.1 单笔电子凸轮数据(E-CAM Data) 动态建立/变更

使用 PMsoft E-CAM Chart 建立电子凸轮数据后，E-CAM Data 就跟着使用者程序一起下载到 DVP-PM 里，如要修改电子凸轮图表内的 E-CAM Data，则要在 PMSoft 上修改完后再次下载。因此 DVP-PM 提供了动态修改 E-CAM Data 的方式给使用者，可于程序执行中变更单组电子凸轮数据。

在程序执行中动态更改电子凸轮数据，同时可变更 E-CAM Data 长度；DVP-PM 利用判断一组电子

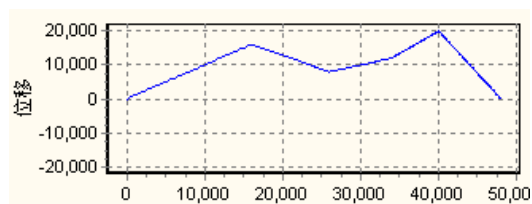
9 电子凸轮

凸轮数据皆为 0 时，表示凸轮数据执行完毕；若 E-CAM Chart 原本有 10 点数据，要变成 5 点数据，可以使用动态变更，将第 6 点数据皆设定为 0，即可将数据点数由 10 点变为 5 点。

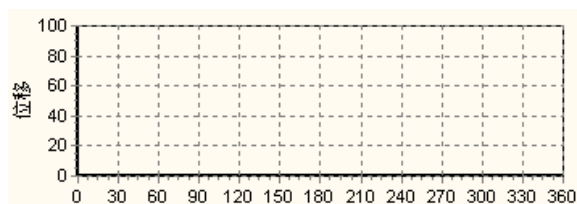
■ 范例说明

【功能说明】

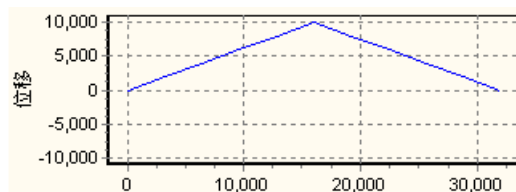
使用单笔数据动态修改方式，变更原有 E-CAM Chart0 内容数据，如下图所示，图(a)表示原有的 E-CAM Chart0 内容，欲修改的 E-CAM Chart 当中可能已包含数据或是尚无数据；经单笔数据动态修改三点主从坐标点为(0,0)、(16000,10000)和 (32000,0)，如下图(b)所示，最后结束时加上(0,0)结束凸轮执行，写入的数据点数为 4 点。



或



(a)原有的 E-CAM Chart 内容

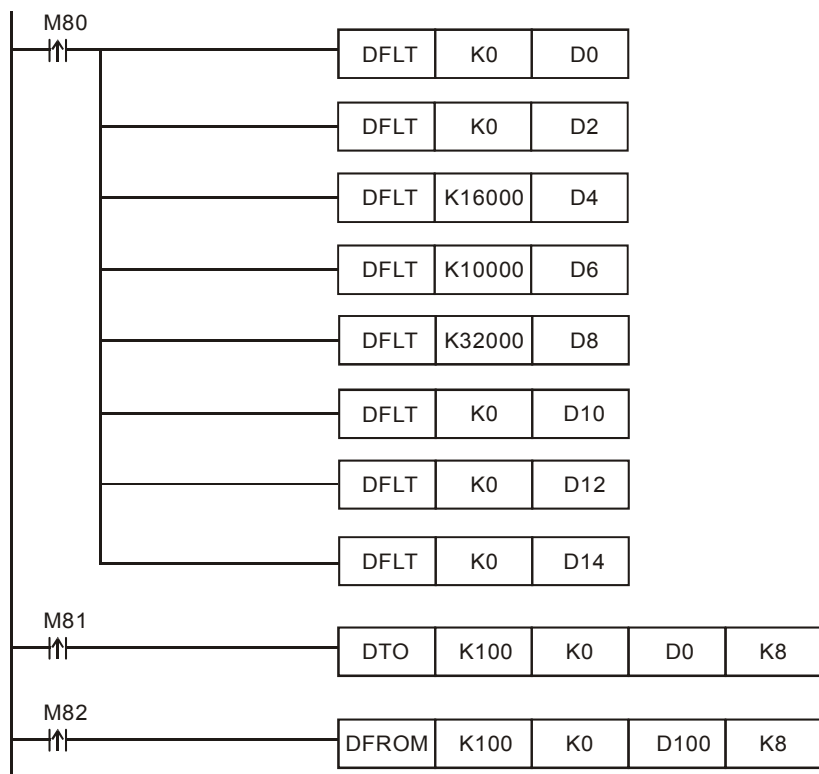


(b)三点坐标组成的 E-CAM Chart

【操作步骤】

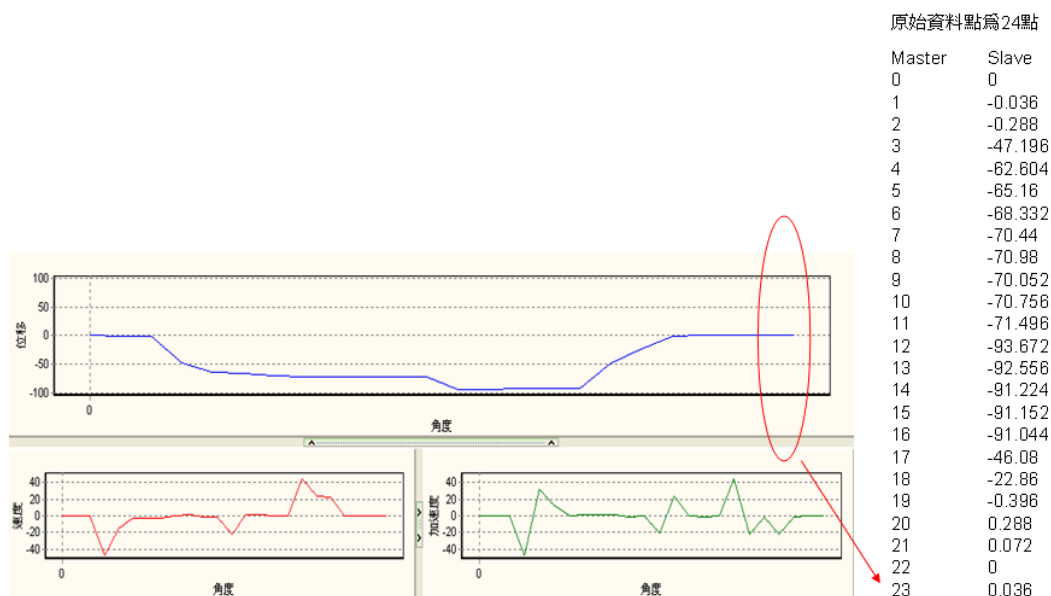
1. Set M80，将 8 笔数据将转成二进制浮点数，分别存放在 D0 至 D15
2. Set M81，将 D0 到 D15 的 8 笔数据写入特殊模块 K100，设定 E-CAM Chart0 的第 0~7 笔数据。
3. Set M82，将 E-CAM Chart0 的 0~7 笔数据读回，存放在 D100 至 D115
4. 使用监控表监看 D100~D115 的内容(监控数值型态为浮点数)，确认写入与读回数据相同。

【程序】



9.4.2.2 建立平滑曲线

平滑处理用意为针对主轴的数据点数做插值与倍率调整动作，让最后汇出的电子凸轮曲线会因解晰度更高而较为平滑，能够提高机件在运行中更加平顺稳定。平滑处理过程以下为例，假设原始数据点数为 24 点，平滑处理设定主轴最大值为 10000 点，则数据间隔所调整的倍率为 $10000/23=434$ 倍，接着依据插值总点数和平滑系数设定生成每个区间新增点数的数据，最后从轴依据原本与主轴的比例关系绘出 10000 点电子凸轮曲线。下图为原始 24 点数据：

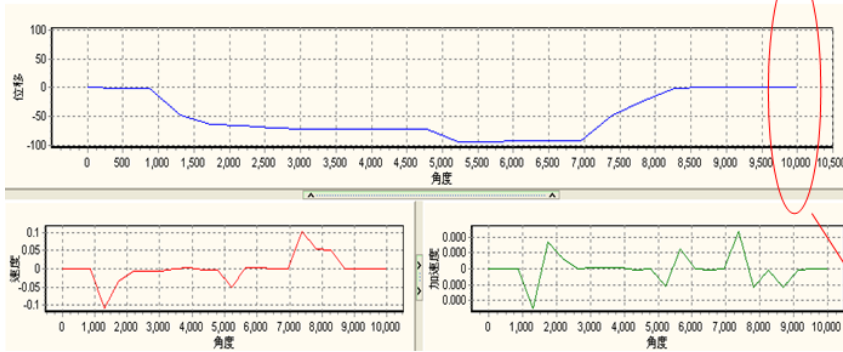


主轴最大值设定为 10000 点，经由倍率调整后，原本位置 0~23 乘上调整倍率后变为位置 0~10000：

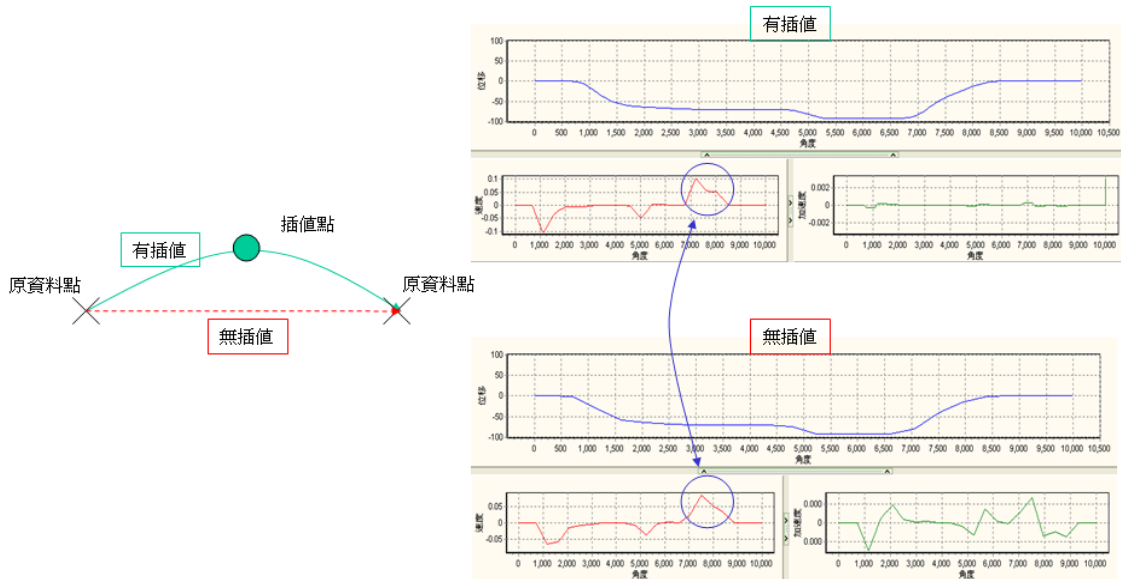
9 电子凸轮

原始資料點為24點
經倍率調整至主軸最
大值為10000

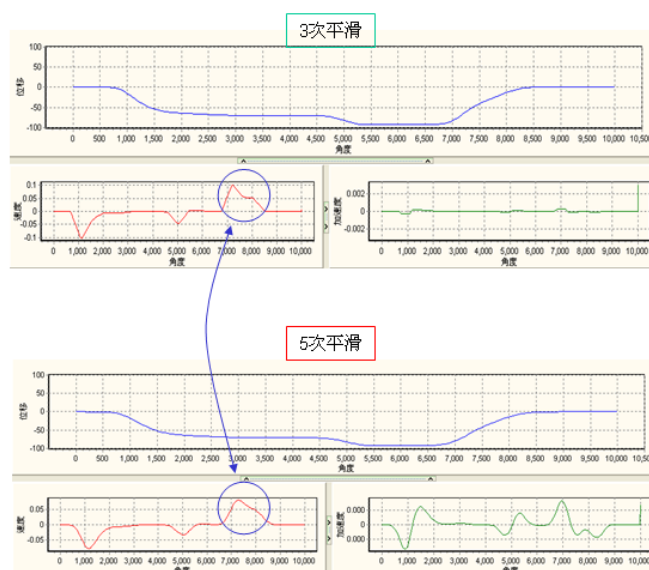
Master	Slave
0	0
434	-0.036
869	-0.288
1304	-47.196
1739	-62.604
2173	-65.16
2608	-68.332
3043	-70.44
3478	-70.98
3913	-70.052
4347	-70.756
4782	-71.496
5217	-93.672
5652	-92.556
6086	-81.224
6521	-91.152
6956	-91.044
7391	-46.08
7826	-22.86
8260	-0.396
8695	0.288
9130	0.072
9565	0
10000	0.036



在原始数据点与点之间放大的区间插入数值，使生成曲线的变化更加平滑；插入比较主轴数据点与点之间执行插值与未执行插值，曲线生成后的差异：



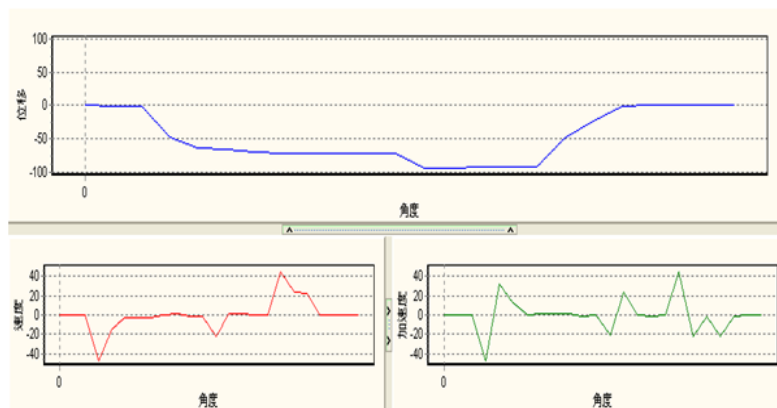
而平滑系数设定越高，所生成的曲线也会更加平滑，以下为同样电子凸轮曲线经过 3 次平滑与 5 次平滑的差异：



■ 范例说明

【功能说明】

将 E-CAM Chart0 的原始曲线做平滑处理，原始数据点数为 24 点；而平滑处理参数设定中，主轴最大值为 10000 点，插值点数为 200 点，平滑系数为 5 次平滑，摆线方式为 B 样条主轴均分。E-CAM Chart0 的原始曲线如下图所示：



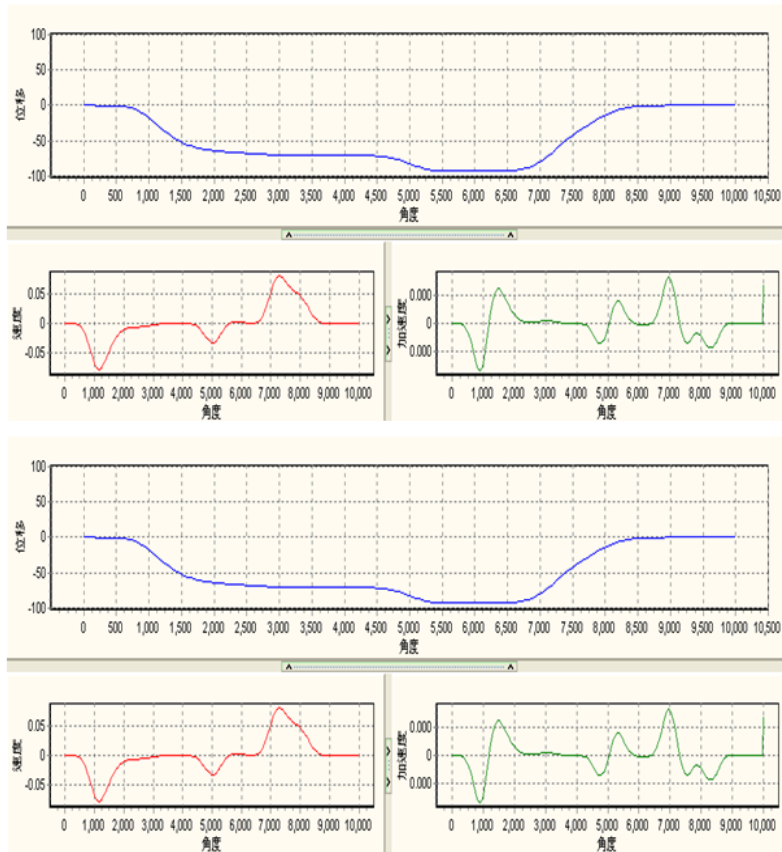
【操作步骤】

1. 在 E-CAM Chart0 建立一个原始点数 23 点的曲线，利用 E-CAM Chart 汇入功能建立电子凸轮关系，其 x、y 关系如下(建立方法请参考章节 9.4)

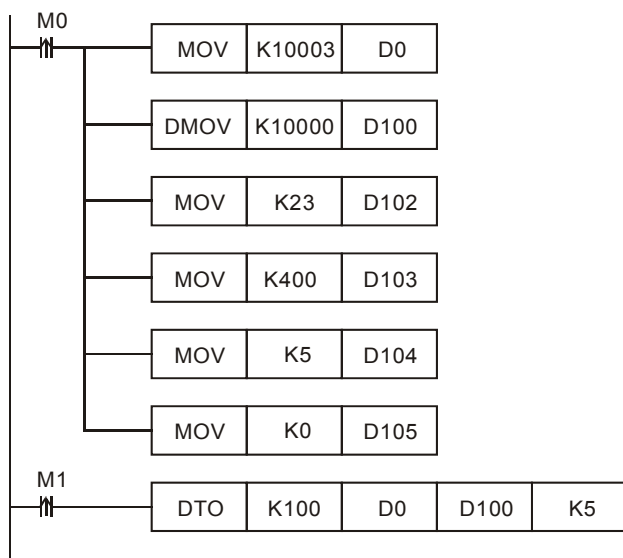
X	Y	x	y	x	y	x	y
0	0	6	-68.332	12	-93.672	18	-22.86
1	-0.036	7	-70.44	13	-92.556	19	-0.396
2	-0.288	8	-70.98	14	-91.224	20	0.288
3	-47.196	9	-70.052	15	-91.152	21	0.072
4	-62.604	10	-70.756	16	-91.044	22	0
5	-65.16	11	-71.496	17	-46.08	23	0.036

9 电子凸轮

2. 修改 E-CAM Chart0 的分辨率，分辨率的值必须不小于插值点数+原始点数
3. Set M0 将欲设定平滑曲线的参数写入 D100~D105 当中
4. Set M1 将 D0 到 D15 的 8 笔数据写入特殊寄存器 K10003，设定平滑曲线
5. 上载 PM 程序，检视 CAM Chart0 所生成的曲线，平滑且插值后的结果如下图所示



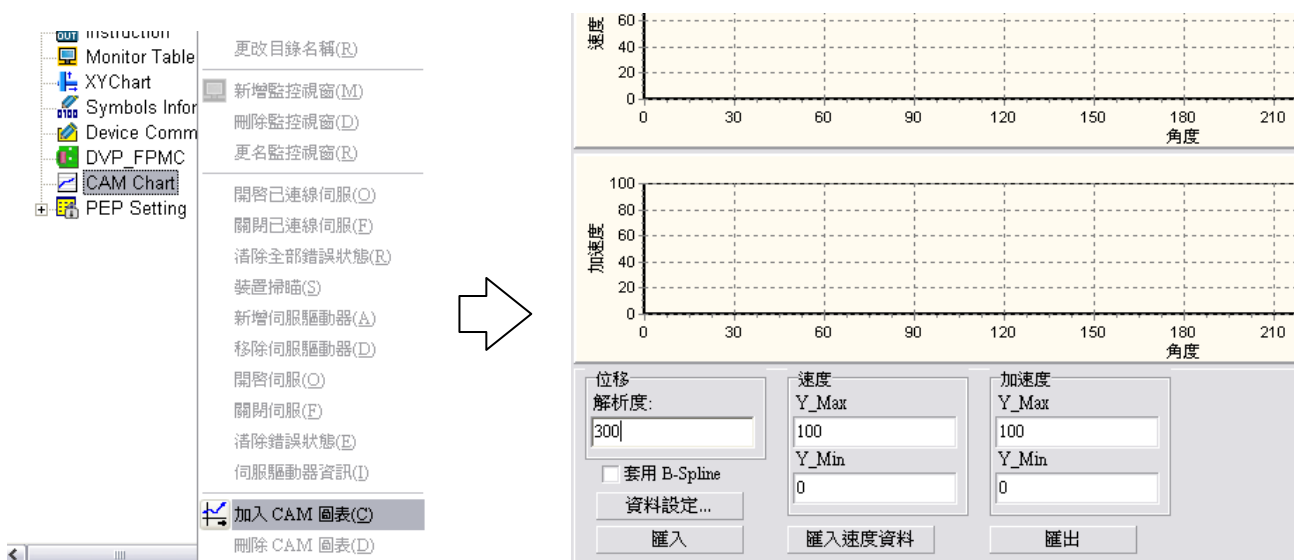
【PM 程序】



9.4.2.3 飞剪电子凸轮数据建立

建立飞剪曲线需要两个步骤:

1. 开启两组 CAM Chart: 在 PMSoft 开启两组 CAM Chart, 分辨率设定则依照生成的飞剪曲线数量而定, 一个飞剪曲线需要 300 个分辨率; 再将这两组空白的 E-CAM Chart 下载至 20PM 当中, 供执行中储存电子凸轮数据。

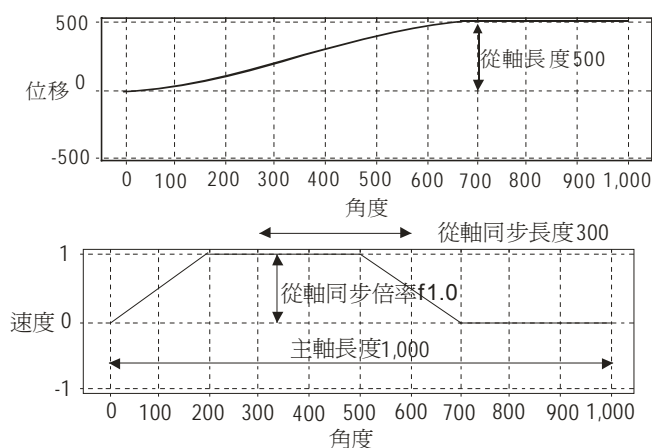


2. 设定 CR#10000 飞剪曲线自动生成寄存器: 将实际应用飞剪机具参数、欲裁切的长度、同步长度以及输入至特殊模块 K100 的 CR#10000, 执行后生成飞剪曲线。

■ 范例说明(一)

【功能说明】

本范例说明使用飞剪曲线自动生成寄存器建立飞剪曲线的步骤, 其中飞剪设定参数为主轴长度=1000、从轴长度=500、同步长度=300、同步倍率=100; 另外将 CAM 表设定为仅能单笔动态修改, 避免程序执行当中修改掉飞剪曲线。下图为此范例生成之飞剪曲线与飞剪参数设定值说明:

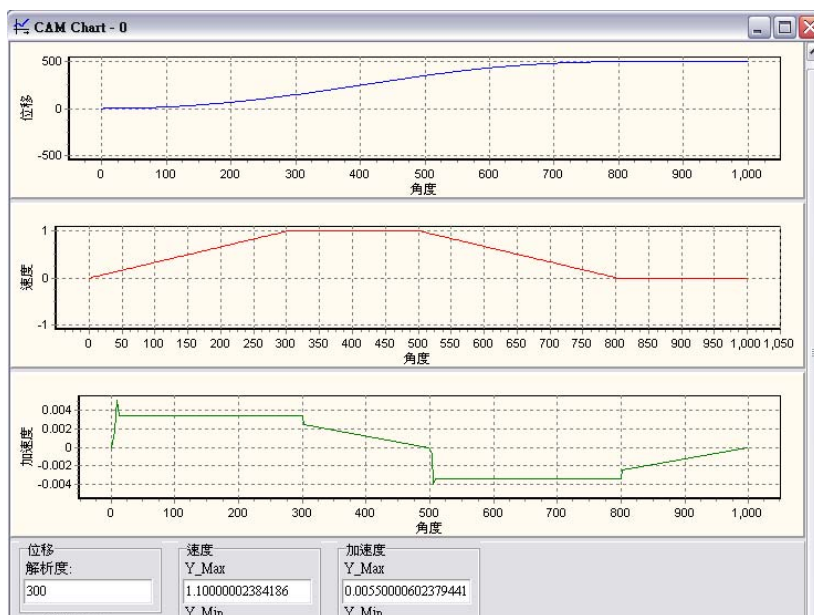


参数	设定值
主轴长度	1000
从轴长度	500
同步长度	200
同步倍率	1.0
同步倍率限制	10.0
加速曲线	0
CAM 曲线	0
生成设定	仅允许单笔动态变更
生成结果	0

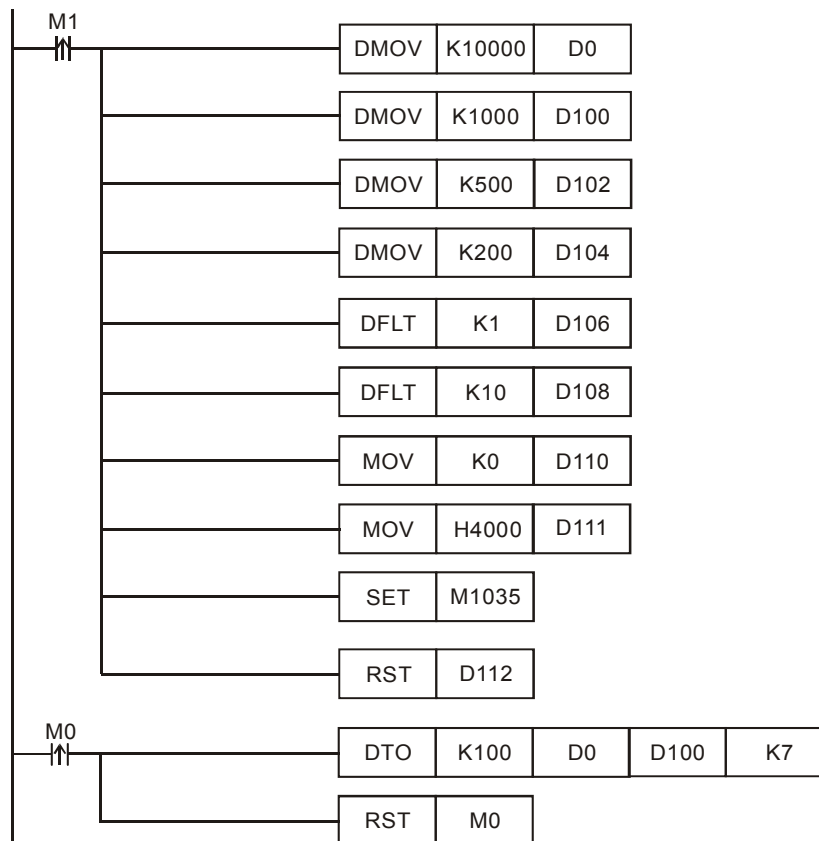
9 电子凸轮

【操作步骤】

1. 在 PMsoft 中开启两组 CAM Chart，分辨率设定为 300。
2. 将程序下载至 20PM 后执行
3. Set M1 将飞剪参数写入 D100~D112，将 CR 编号 K10000 写入 D0。
4. Set M0 将 D0 到 D15 的 8 笔数据写入特殊寄存器 K10000，执行飞剪曲线自动生成。
5. 停止执行 PM，将程序上载。
6. 检视上载后的 CAM Chart0 曲线，为自动生成的飞剪曲线。



【PM 程序】



■ 范例说明(二)

【功能说明】

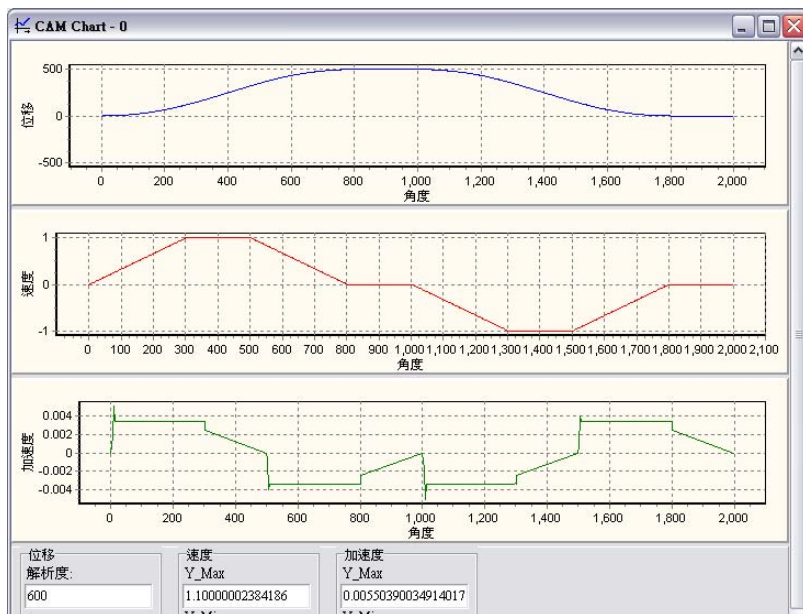
此范例利用两个飞剪曲线建立一个追剪曲线；这里使用到接续前一数据的设定，完成第二个飞剪曲线，在第二个飞剪曲线设定上仅需将从轴长度由正改为负，剩余参数延续前一个飞剪曲线，完成追剪动作中返回的曲线。

操作上首先建立第一个飞剪的动态曲线，第二个 CAM 表的动态数据设定时，建立一个从轴长度为负，且接续第一个 CAM 表设定，即可建立一个追剪曲线 CAM 表。由于是两个飞剪曲线合成一个追剪曲线，故在 PMsoft CAM Chart 中开启的 2 个分辨率为 600 的空白 CAMChart，下表为此范例生成之追剪曲线参数设定：

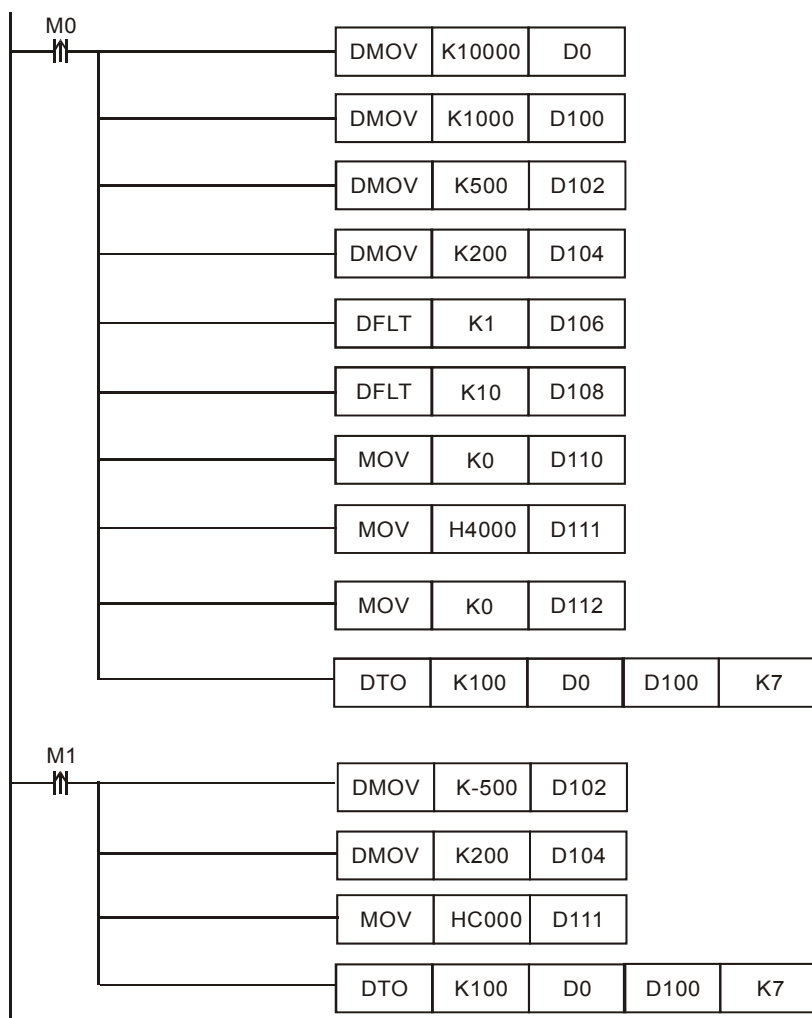
参数	第一个飞剪曲线参数	第二个飞剪曲线参数
主轴长度	1000	1000
从轴长度	500	-500
同步长度	200	200
同步倍率	1.0	1.0
同步倍率限制	10.0	10.0
加速曲线	0	0
CAM 曲线	0	0
生成设定	仅允许单笔动态变更	仅允许单笔动态变更 延续上次数据
生成结果	0	0

【操作步骤】

1. 在 PMsoft 中开启两组 CAM Chart，分辨率设定为 600
2. 将程序下载至 20PM 后执行
3. Set M0 将飞剪参数写入 D100~D112，将 CR 编号 K10000 写入 D0；将参数写入特殊模块 100 的 CR10000，执行飞剪曲线自动生成，生成第一个飞剪曲线
4. Set M1 改变飞剪参数中的 D102、D104 与 D111；将参数写入特殊模块 100 的 CR10000，执行飞剪曲线自动生成，生成第二段飞剪曲线
5. 停止执行 PM，将程序上载
6. 检视上载后的 CAM Chart0 曲线，为自动生成的追剪曲线



【PM 程序】



■ 范例程序(三)

【功能说明】

此范例说明利用飞剪曲线自动生成寄存器建立一个内容为多个飞剪曲线的电子凸轮曲线,可以在裁切轴上有多把刀的应用。本范例示范如何建立三个飞剪曲线,操作方法为建立好第一个飞剪曲线后,接着后续两个飞剪曲线使用接续前次数据功能,即可串连三个飞剪曲线生成一个电子凸轮曲线。

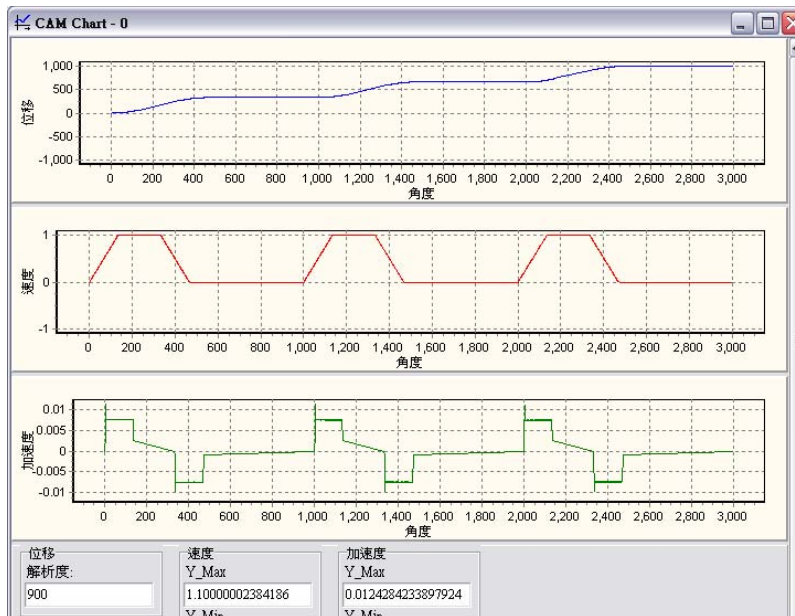
另外考虑到裁切轴刀数无法平均旋转一圈脉冲数的情况,若未处理剩余脉冲数会造成累积误差。处理方式为将总脉冲数除与刀数取得的脉冲数作为每个飞剪曲线的从轴长度,剩下的脉冲数加在最后一次飞剪生成当中,让一个电子凸轮周期执行的从轴长度总和等于总脉冲数。

下表为此范例生成之追剪曲线参数设定:

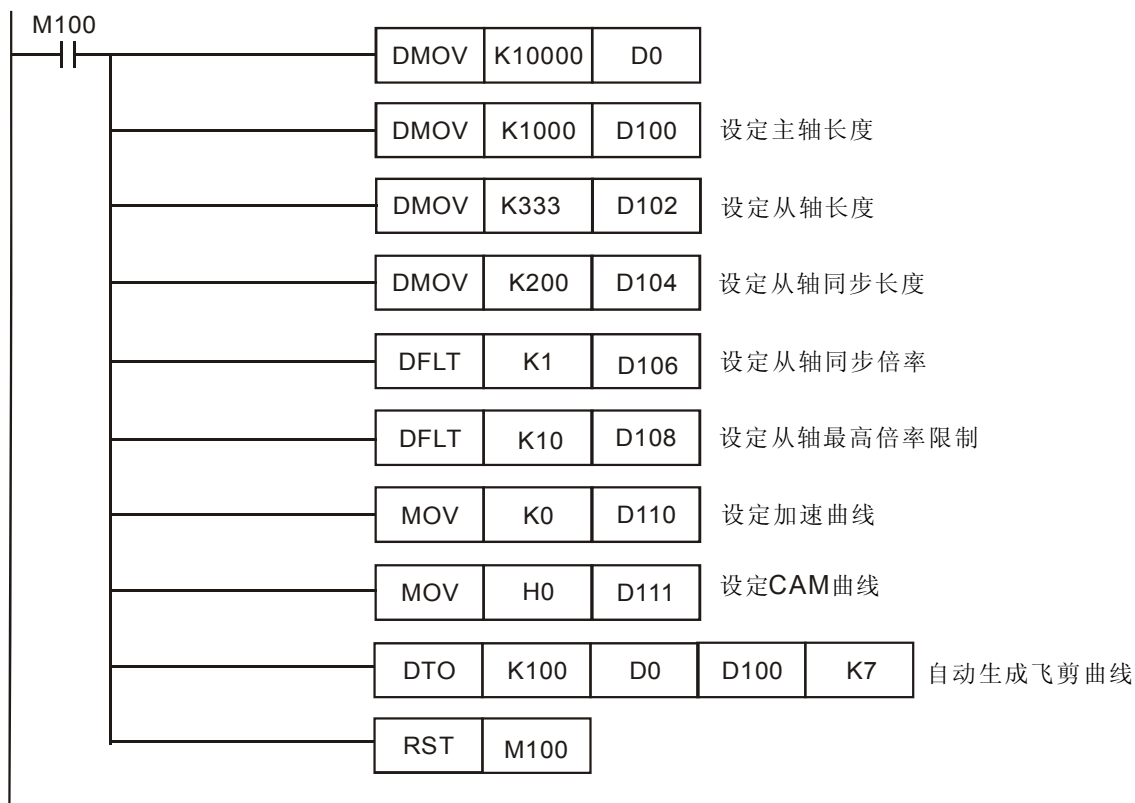
参数	第一个飞剪曲线参数	第二个飞剪曲线参数	第三个飞剪曲线参数
主轴长度	1000	1000	1000
从轴长度	333	333	334
同步长度	200	200	200
同步倍率	1.0	1.0	1.0
同步倍率限制	10.0	10.0	10.0
加速曲线	0	0	0
CAM 曲线	0	0	0
生成设定	仅允许单笔动态变更	延前上次数据	延续前次数据
生成结果	0	0	0

【操作步骤】

1. 在 PMsoft 中开启两组 CAM Chart, 分辨率设定为 900
2. 将程序下载至 20PM 后执行
3. Set M100 将飞剪参数写入 D100~D112, 将 CR 编号 K10000 写入 D0; 将参数写入特殊模块 100 的 CR10000, 执行飞剪曲线自动生成, 生成第一个飞剪曲线
4. Set M101 改变飞剪参数中的 D111; 将参数写入特殊模块 100 的 CR10000, 执行飞剪曲线自动生成, 生成第二段飞剪曲线
5. Set M102 改变飞剪参数中的 D102 与 D111; 将参数写入特殊模块 100 的 CR10000, 执行飞剪曲线自动生成, 生成第三段飞剪曲线
6. 停止执行 PM, 将程序上载
7. 检视上载后的 CAM Chart0 曲线, 为自动生成的电子凸轮曲线



【PM 程序】



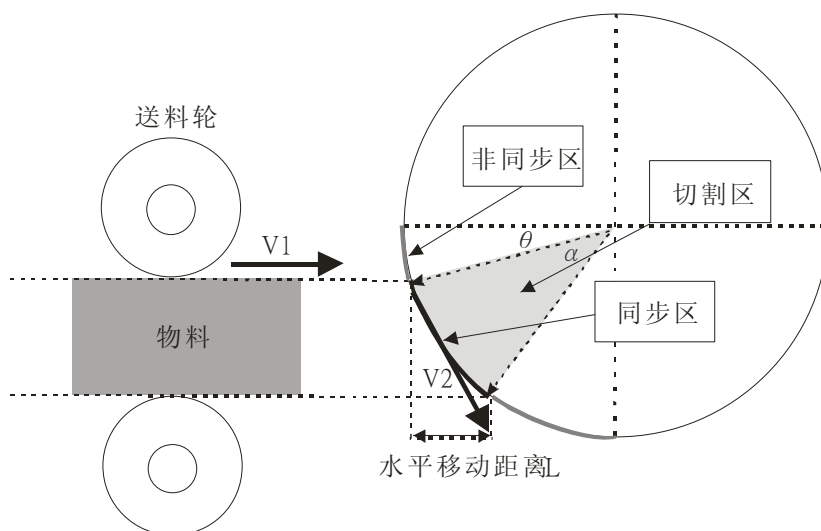


9.4.2.4 厚料飞剪电子凸轮数据建立

一般的飞剪动作刀具与物料的接触为一点，在规画同步点时最少只需一点即可，不需考虑物料的厚度。对于厚料的裁切例如肥皂、钢板等，需考虑到下刀速度与同步速度的关系，当进入同步裁切区后，物料水平移动送料与切刀水平速度相同，确保切面为垂直不产生弧形。下图为物料与刀具的角度的关系图，其中 V_1 为送物料速度， V_2 为下刀速度，灰色区域为同步区(即为切割区)，当物料的高度为固定时，刀具与物料初接触角度 θ 为固定值，而切入角度 α 会随着切刀移动而变化，主从轴的同步速度倍率的关系为

$$\frac{V_2}{V_1} = \frac{1}{\sin(\alpha + \theta)}$$
。如下图所示，在同步区切刀是旋转的，其水平移动距离必须与材料的移动距离一致，

而在同步区内主从轴所移动的距离为 $L = R \cos \theta - R \sin \alpha$ ，在异步区则是根据所裁长度的大小来加减速。



9 电子凸轮

此外，考虑到切刀的起始点位置，使用从轴上的刀数来判断每个切刀的起始位置角度是否相同，若刀数能够平均总脉冲数时，则可算出每个切刀所需的脉冲数，将其换成角度来看，即得知每个切刀的起点与终点，在厚料飞剪的程序设定中仅需设定同步区起始与结束角度即可。若刀数无法平均总脉冲数时，则需另外设定同步区前的长度以及同步区前夹角来设定出切刀的起点与终点。以下将分别说明在刀数能平均总脉冲数情况与不能平均总脉冲数情况的程序设定。

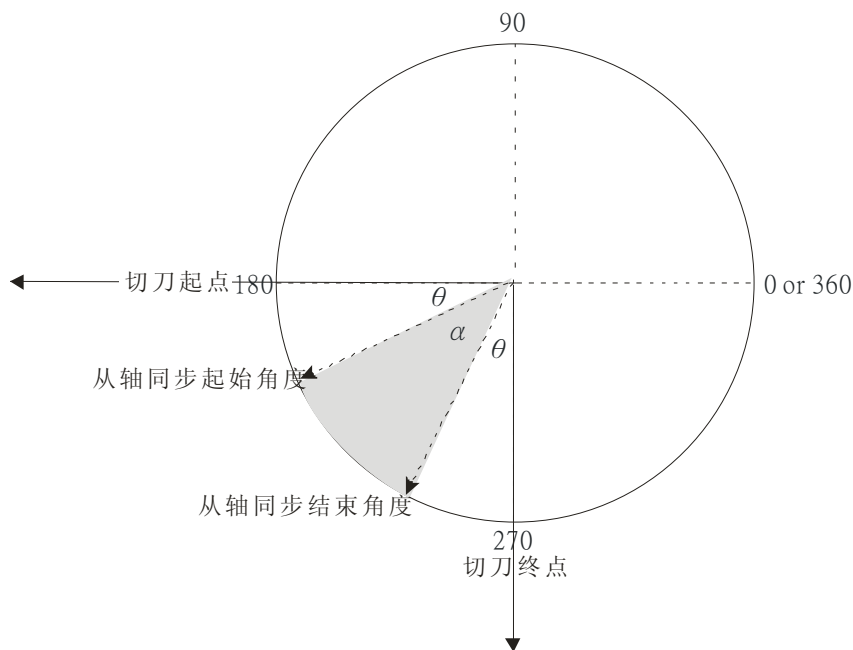
切入角度设置必须小于或等于实际切入角度，如果设定切入角度大于实际切入角度会在切入点产生弧度切面，同样设置同步结束角度必须大于或等于同步结束角度。

■ 总脉冲可均分状态：

从轴刀数若能够平分总脉冲数的情况，设定均分异步区的功能，在 **PM** 中会自动计算设定每次裁切动作的所需的脉冲数。如下图所示，实际调整中仅需考虑从轴同步起始角度与从轴同步结束角度即可算出同步区的角度 α 与同步区前后的异步区角度 θ ：

$\alpha = \text{從軸同步區結束角度} - \text{從軸同步區起始角度}$

$$\theta = \left[\frac{360}{\text{从轴一周脉冲数}} \right] \times \left[\frac{\text{从轴一周脉冲数}}{\text{从轴刀数}} - \alpha - \frac{\text{从轴一周脉冲数}}{360} \right] \div 2$$

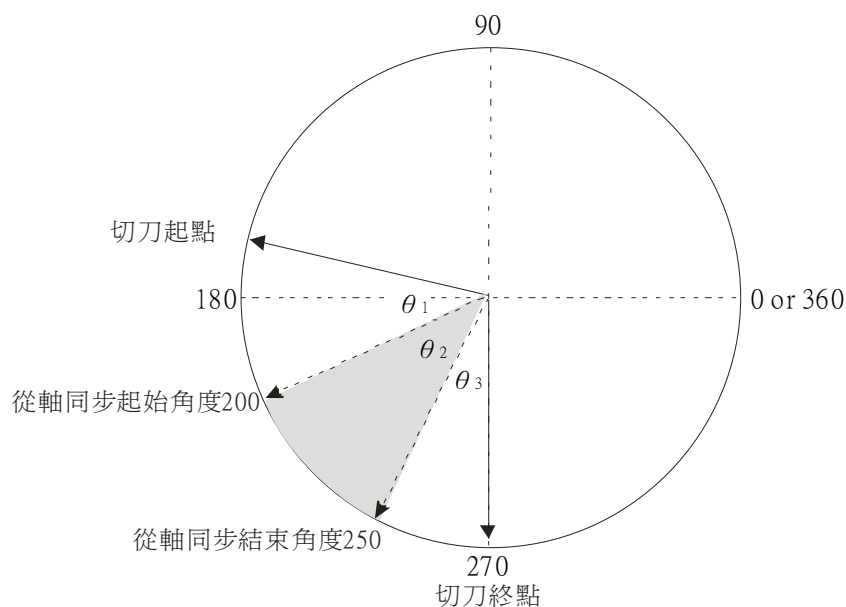


实际调整运作的步骤为先选定好同步起始与结束角度后，再将切刀的位置调整至切刀起点，接着启动凸轮咬合，检视物料的切刀面是否有弯曲的情况，来微调同步起始角度与结束角度，直到切刀面为平整，表示调整完成。

■ 总脉冲无法均分的状态

当从轴刀数无法平均一周总脉冲数时，则需考虑到每个切刀起始位置与结束位置，且同步区前后的异步角度不同，在 **CAM** 曲线设定中需另外输入同步区前异步角度和长度，以计算切刀起点位置。如下图所示，在实际调整中需各别选定同步区前异步角度 θ_1 、同步区前长度、同步起始角度以及同步结束角度，另外各切刀所需的脉冲数需各别在 **CAM** 设定，从以上参数调整每个切刀 **CAM** 曲线后，

接着执行微调的动作，直到切面为平滑。



■ 范例说明

【功能说明】

此范例说明使用飞剪曲线自动生成指令建立厚料飞剪曲线，范例中设计送料轴直径为 170(mm)、裁切轴直径为 240(mm)、使用刀数为 4 把、主从轴一圈脉冲总数为 200000、切入点角度 $\theta=210$ 度，同步结束角度 250 度；范例中，主轴直径等于 170(mm)，切刀数量总共有 4 把，推算切长为 $170 \times 314 \div 4 = 13345$ 。其设定参数如下表所示。

参数	设定值
厚料飞剪格式设定	1
主轴一圈脉冲数	200000
主轴一周长度	53380
从轴一周脉冲数	200000
从轴一周长度	75360
从轴同步起始角度	210
从轴同步结束角度	250
主轴切长	13345
从轴刀数	4
主轴同步区前长度	-
从轴同步区前夹角	-
同步区点数	-

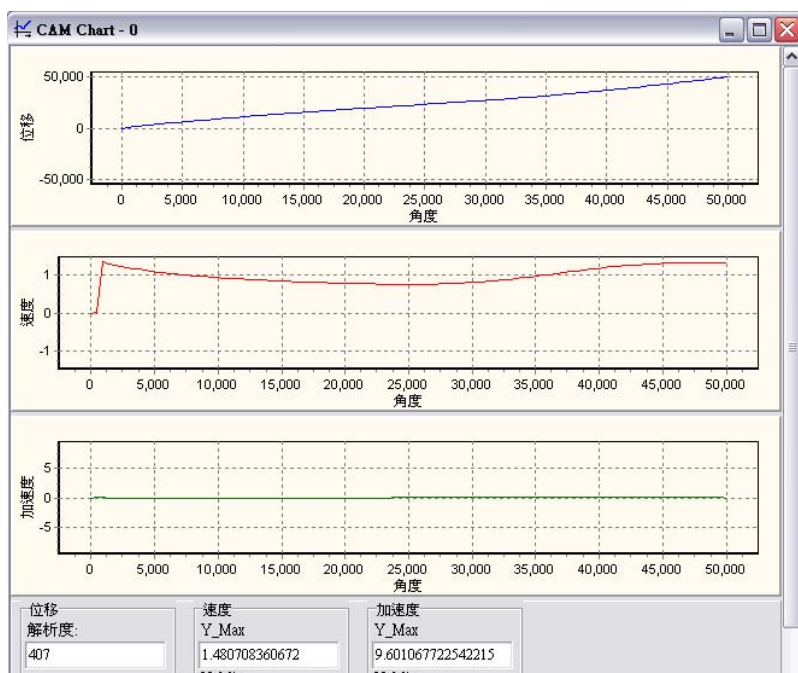
【操作步骤】

1. 在 PMsoft 中开启两组 CAM Chart 0，分辨率设定为 407
2. 将程序下载至 20PM 后执行
3. Set M0 将厚料飞剪参数写入 D100~D116，将 CR 编号 K10000 写入 D0；将参数写入特殊模块 100

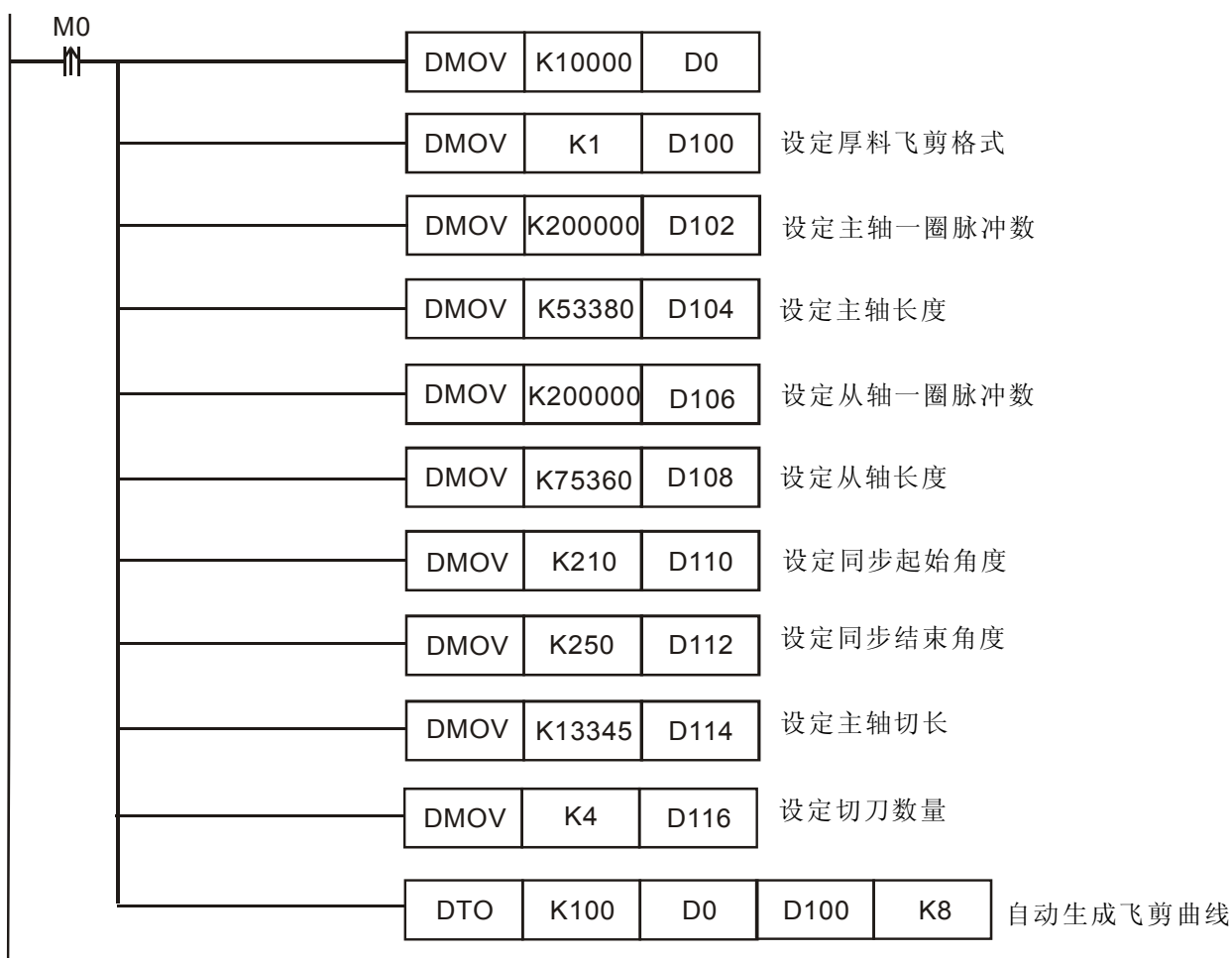
9 电子凸轮

的 CR10000，执行飞剪曲线自动生成，生成厚料飞剪曲线

4. 停止执行 PM，将程序上载
5. 检视上载后的 CAM Chart0 曲线，为自动生成的厚料飞剪曲线

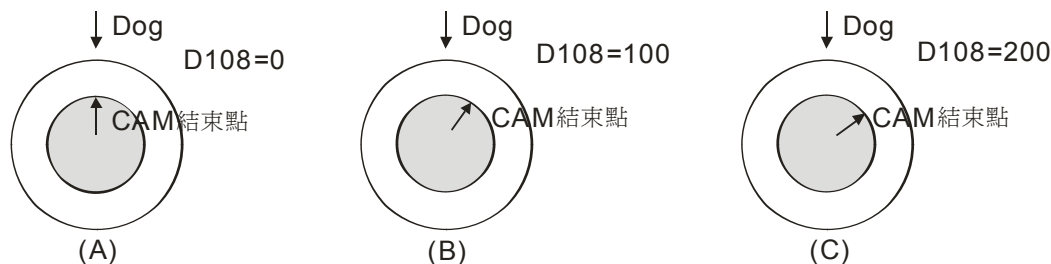


【PM 程序】



9.4.2.5 刀轴误差补偿

在实际应用中，由于实际在机具上输出的位置和 PM 依照电子凸轮曲线输出会有误差，且在凸轮周期执行次数越多，误差越大；故利用外部信号 DOG 至 CAM 结束点的差值(D108)来进行补偿，首先要备份一次 D108 初始值(图 A)，之后再以备份值与出现误差之后的从轴计数差值 D108(图 B, C)来计算两者间差值，作为补偿值参考。



实际操作方法为：

1. 读回电子凸轮状态的从轴计数差值，这个差值为 DOG 信号触发捕捉的值与 CAM 结束时捕捉的值相减后得到。
2. 下次周期执行开始产生刀轴误差后，将这时读回的从轴计数差值与从轴计数差值初始值相减，得到从轴计数差值误差。
3. 将电子凸轮参数中的从轴长度与从轴计数差值误差相减，得到实际机具输出与电子凸轮输出的偏差值；当偏差值超出或小于 0.5 倍从轴长度时，便进行补偿
4. 补偿方式如下：

$$\begin{cases} \text{偏差值} > \frac{1}{2} \text{从轴长度} \Rightarrow \text{补偿值} = (\text{偏差值} - \text{从轴长度}) \times \text{补偿比例} \\ \text{偏差值} < \frac{1}{2} \text{从轴长度} \Rightarrow \text{补偿值} = (\text{偏差值} + \text{从轴长度}) \times \text{补偿比例} \end{cases}$$

5. 下次凸轮周期则将补偿值加入从轴长度中，完成刀轴补偿。

■ 范例说明

【功能说明】

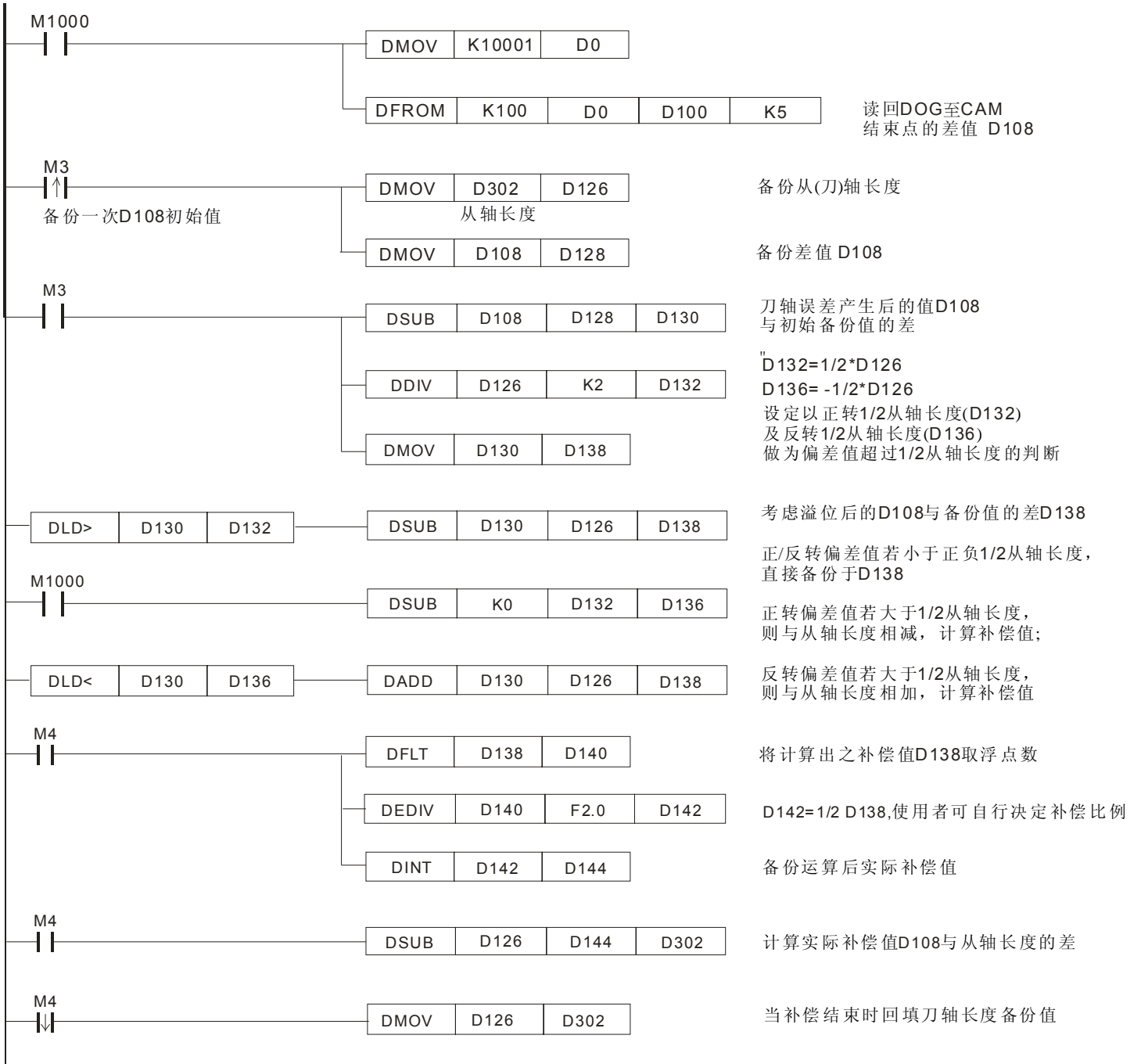
此范例应用在电子凸轮执行中，可用于判断目前从轴的凸轮输出与实际输出之间是否误差，并且在发生误差时，做从轴输出的补偿。其中寄存器 D302 表示从轴长度，实际操作时将从轴长度的值写入 D302 当中。

【操作步骤】

1. 程序执行，读取特殊模块 K100 的 CR#10001 电子凸轮执行状态至 D100~D108
2. Set M3 将目前从轴长度写入寄存器 D126、将从轴计数误差写入寄存器 D128 当中作为初始值；计算从轴计数误差初始值与误差开始后的从轴计数误差两者间的差值；另外将从轴长度除以 2 作为比较值。
3. 若从轴计数差值误差大于 1/2 从轴长度，则将从轴计数差值误差与从轴长度相减，得到补偿值
4. 若从轴计数差值误差小于 1/2 从轴长度，则将从轴计数差值误差与从轴长度相加，得到补偿值
5. Set M4 将计算出的补偿值乘以补偿比例后，得到实际补偿值

6. Rst M4 结束补偿时，将补偿值填回从轴长度。

【PM 程序】



※Set M4进行补偿及补偿值的大小需依实际情形调整倍率(目前设定为差值的0.5倍)

※需以DTO进行动态切换(建立飞剪CAM Data)变换补偿后的刀轴长D302

9.4.2.6 追剪应用-判断切刀到位状态

追剪应用时有时需在返回时检查切刀是否到位，如不到位保持目前速度往前至遇极限，不然在返回时切刀如不到位可能会损伤机械结构。此功能可与目前凸轮周期中的主轴位置，在程序中比较是否启动 M1757。目前凸轮周期中的主轴位置可由 D1863..D1862 目前主轴累积位置及利用 FROM 读出凸轮数据状态 2 读取中的主轴位置(CAM 周期结束时 Capture)差值取得。

■ 范例说明

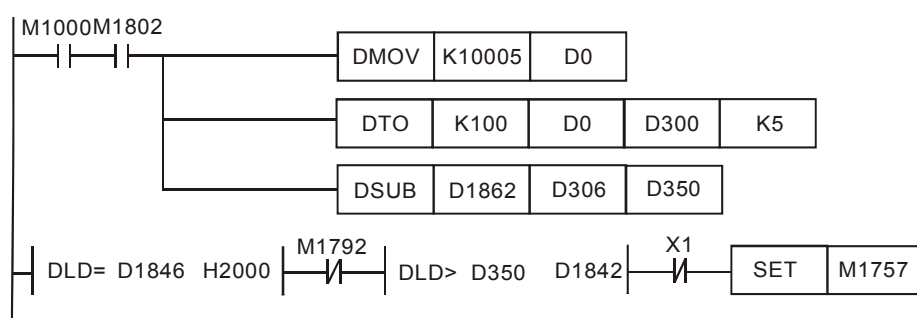
【功能说明】

此范例应用在追剪应用执行中，可用于判断目前主轴设定位置与主轴输出之间是否误差，若实际输出并未到位时，则启动 M1757 让主轴位置持续输出至遇到左右极限后停止。

【操作步骤】

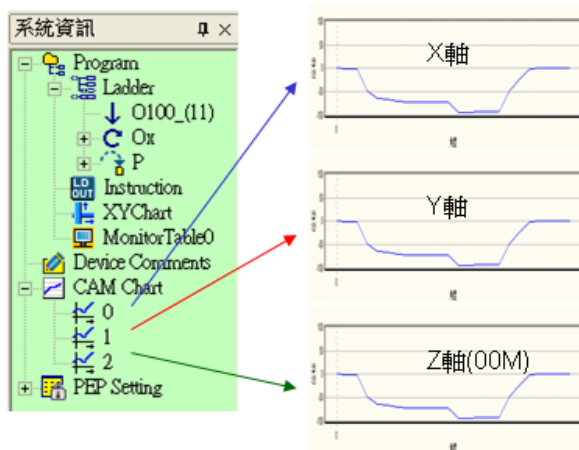
1. 程序执行，当凸轮周期执行完毕时读取特殊模块 K100 的 CR#10005 电子凸轮执行状态至 D300~D308
2. 将累积手摇轮脉冲数(主轴累积位置 D1862)减去读回的主轴位置捕捉值(D306)，得到实际主轴位置(D350)
3. 比较实际主轴位置(D350)与目标位置(D1842)后，若实际主轴位置大于目标位置，且上极限未启动时，启动 M1757，停止 CAM 执行。

【PM 程序】



9.5 多凸轮模式

多轴凸轮模式没有像单凸轮模式有区分周期和非周期模式，仅以连续周期的方式运作，在使用多凸轮功能时，凸轮轴只能固定使用一组 CAM 表，即 PMsoft 之 CAM Chart 0、1、2 分别对应 x、y、z 三轴，如下图：



多凸轮启动与单凸轮相同，由 X 轴运动命令之 D1846 寄存器来启动，不同的是要设定各凸轮轴的运动模式并开启 C200 计数器以计算主轴脉冲数，以下先针对这三项做说明：

1. D1846

D1846=H'2000 启动多凸轮运动，与启动单凸轮周期模式相同。

2. 工作模式设定(X-D1847 Y-D1927 Z-D2007)

在多凸轮时，须先将 X 轴之 D1847 之 Bit12..11 设定为 01，而 D1927 与 D2007 则视 Y、Z 凸轮轴是否启动来设定寄存器之 Bit11。

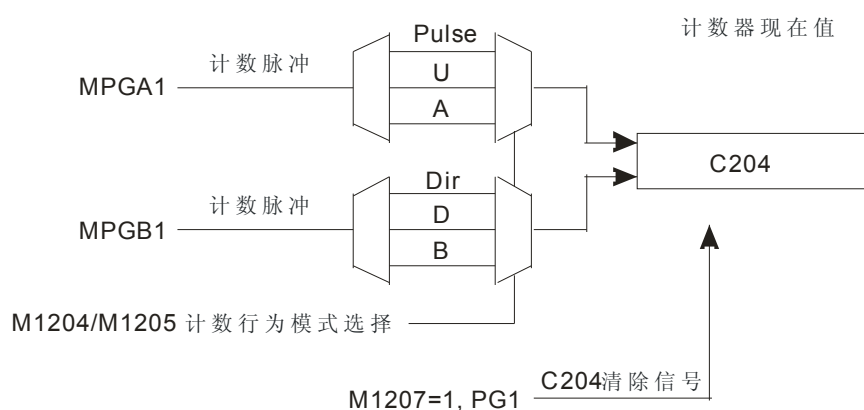
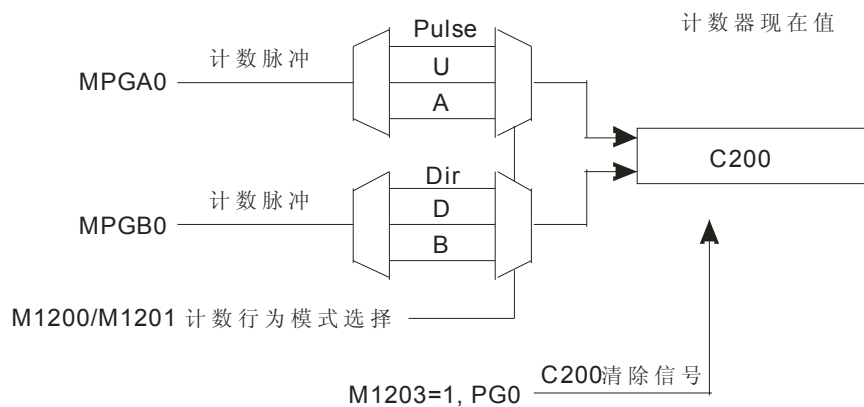
3. C200/C204 计数器

设定并启动 C200/C204，多轴中断插入电子凸轮会使用 C200/C204 做为主轴的输入信号，计数模式设定需配合主轴信号之脉冲形式。

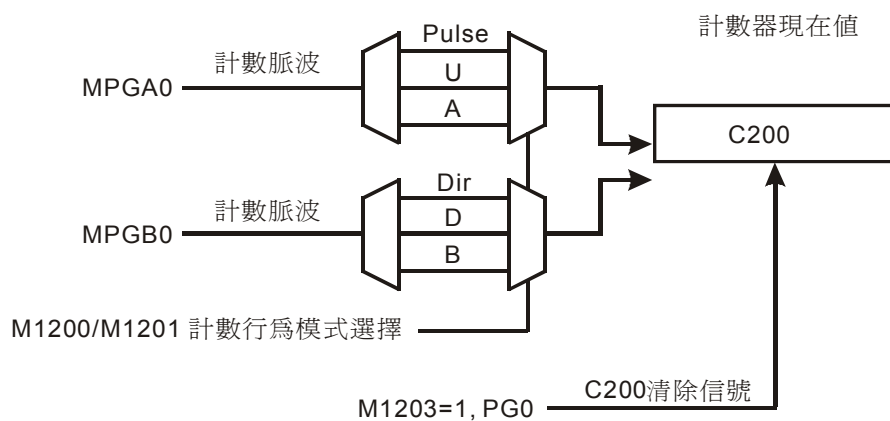
C200/C204 计数模式设定

装置	C200 计数模式	装置	C204 计数模式
M1200=0, M1201=0	U/D*	M1204=0, M1205=0	U/D
M1200=1, M1201=0	P/D*	M1204=1, M1205=0	P/D
M1200=0, M1201=1	A/B*(1 倍频)	M1204=0, M1205=1	A/B(1 倍频)
M1200=1, M1201=1	4A/B(4 倍频)	M1204=1, M1205=1	4A/B(4 倍频)

M1908=Off 时，高速计数器 C200 输入信号由 A0/B0 控制，清除信号由 PG0 控制。高速计数器 C204 输入信号由 A1/B1 控制，清除信号由 PG1 控制。



M1908=On 时，XYZ 三轴共享高速计数器 C200，输入信号由 A0/B0 控制。C200 清除信号由 PG0 控制，C204 清除信号由 PG1 控制。

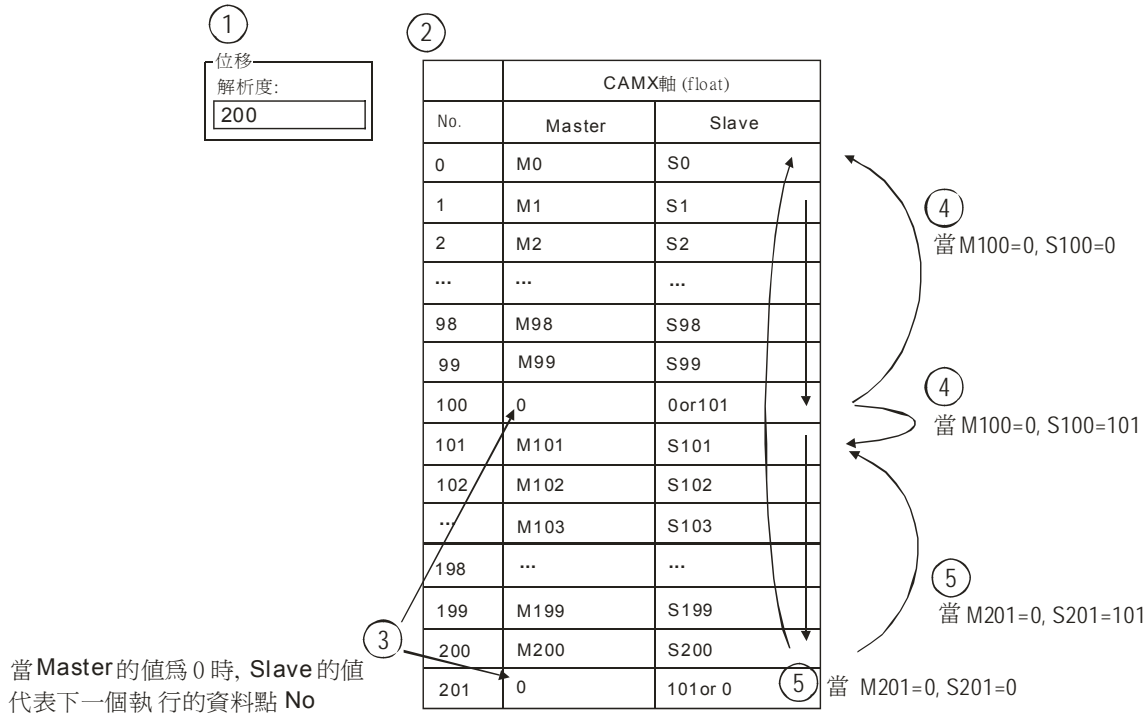


9 电子凸轮

多凸轮运动有下列两种运动模式：

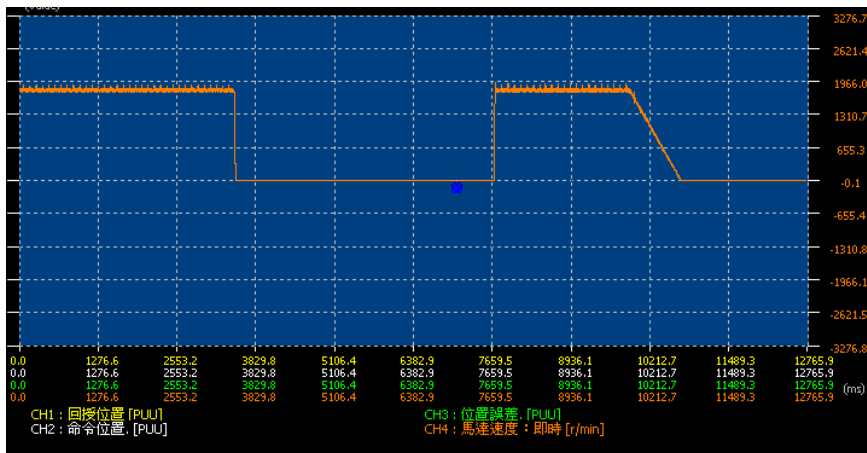
1. 跳点切换运动模式

在多轴中断插入电子凸轮中为了让使用者可以动态切换电子凸轮表的数据，我们可以将一个 M 点的电子凸轮表分区块，一次只使用一个区块，而新的数据可以加在下一个区块中，在区块间插入一笔 (0,n1)数据，当凸轮表开始后碰到主轴值为 0 的点时，其从轴值 n1 会被判读为下一笔被执行的数据编号，而切换到第 n1 笔凸轮点，下面我们以一个 N 点的电子凸轮表，将其前 202 点分为 2 个区块来使用。



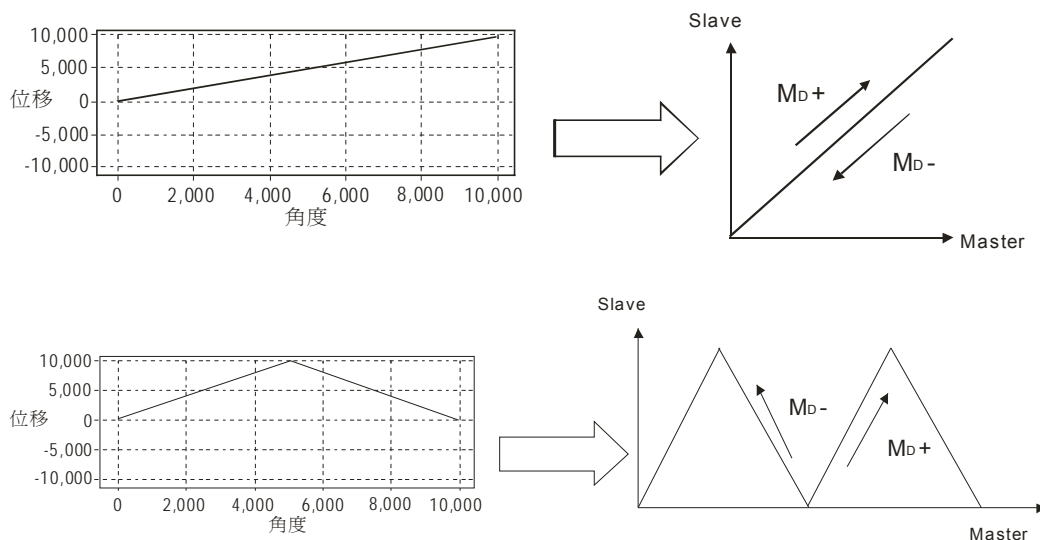
利用了 No100 及 No201 Slave 点来指定下一执行的数据点，所以前一个区块在执行时使用者可以预排下一区块的数据，当第二区块之起点数据主轴(M102)值等于零时，第二区块之主轴及从轴皆会以前一区块之终点为基准执行凸轮运动(M99+M102,S99+S101)；起点主轴值不为零；从轴会改以 0 为基准(M99+M102,0+S101)。

若运转速度设定 V(I)设定为 0Hz，则可启动凸轮的煞车减速功能，减速速度为 T_{dec}(D1837)。下图显示含煞车减速功能之差异，左图无减速，右图使用减速功能。



2. 正反转双向运动模式

多凸轮运动，凸轮动作会随主轴的方向作连动，轴工作模式需设 **Bit12,11 = ON(K3)**。以下面 2 个凸轮表为例，其随主轴动作方式如下图，反转时到第 0 点时会自动跳到凸轮表最后一点，重复执行该凸轮数据。此外，使用反转时需避免与跳点模式混用，以避免反转至主轴值为 0 的切换点而发生问题。



另外正反转双向模式下建立凸轮表，须注意单一段凸轮表数据末端，不能有其它主轴为零的多余点存在，如下图。

Data_S.txt - 记事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

0	0
100	100
200	400
300	600
400	800
500	1000
600	1200
700	1400
800	1600
900	1800
1000	2000
1100	2200
1200	2400
1300	2600
1400	2800
1500	3000
1600	3200
1700	3400
1800	3600
1900	3800
2000	4000
0	0
0	0
0	0
0	0
0	0

The last five rows of the table are enclosed in a red box with a large 'X' over them, indicating that these zero-axis points are invalid and should be removed.

9.6 电子凸轮实际应用

9.6.1 高速绕线机机构及动作说明

本章节介绍的是全自动无骨架系列高速绕线机，可以绕制不同规格的空心线圈，如：传动线圈，扬声器线圈，天线线圈以及各种无骨架通用线圈。设备具有性能可靠，高速高效率，自动化程度高，适合于线圈的大批量生产。绕制各种线圈如下图：



一般普通绕线机采用内置脉冲功能的小型 PLC, 通过绕线轴编码器速度输出到 PLC 内置高速输入点, 将绕线轴与排线轴的速度比进行简单速度同步, 这种方法受 PLC 运算影响, 同步精度差, 假设计算量大, CPU 处理时间会较长, 因此会出现绕线不均匀, 堆积, 塌陷等问题, 严重影响绕线成品的质量, 举例来说, PLC 对绕线轴编码器作高速计数, 当到达计数值时利用中断方式控制排线轴电机反向绕制, 但受 CPU 运算处理时间的影响会出现滞后而产生误差, 在低速的情况下尚可达到基本绕制要求, 但是对于高速绕制多层线圈时就会出现线圈端面不齐整, 使成品质量下降。

台达 DVP-PM 是一款专用运动控制型 PLC, 采用高速双 CPU 结构形式, 利用独立 CPU 处理运动控制算法, 可以很好地实现各种运动轨迹控制、逻辑动作控制, 直线 / 圆弧插补控制等, 在高速绕线机中正是利用了 DVP-PM 运动控制器的电子凸轮功能, 有效的解决上述绕线在转换方向时出现的绕制不均匀、堆积、不平整等问题。

9.6.1.1 高速绕线机机构及动作说明

高速绕线机设备结构共包含九部分机构, 分述如下:



(1) 机架

机架由角钢框架及不锈钢台面组成, 并设置脚轮便于移动, 当设备到位后可将支脚调低作为稳定支撑。

(2) 张力机构

安装于进线部分, 作为绕线张力调节, 保证线圈绕制时维持张力恒定, 张力调节器具有调节旋钮可针对不同需求进行张力调节设定, 调整完毕后, 张力调节器自动控制绕线张力。

(3) 绕线机构

主要由台达 B 系列 200W 伺服电机、同步齿形带、绕线飞叉组成，是电子凸轮运动中的绕制主轴，铜线经过飞叉旋转绕制于绕线模头上，是绕线机主要运动部件之一。

(4) 排线机构

包括台达 B 系列 100W 伺服电机、精密直线螺杆、精密导轨、气动滑叉等，是电子凸轮运动中的排线从轴，在绕线运动中跟随绕线主轴正反向往复运动实现排线动作，是绕线机主要运动部件之一。

(5) 工作转台

由分度步进电机、旋转台、线叉、绕线模头组成，该设备为多任务位绕线机，在绕线同时执行模头预热、剪线、加热、脱模等工艺动作，这需要工作转台按不同工位动作完成。

(6) 剪线机构

为气动执行机构，主要是将绕制完成的线圈两端引线剪断。

(7) 脱模机构

由分度步进电机、气动脱模块成，将绕制完成的成品从绕线模头取下。

(8) 热风系统

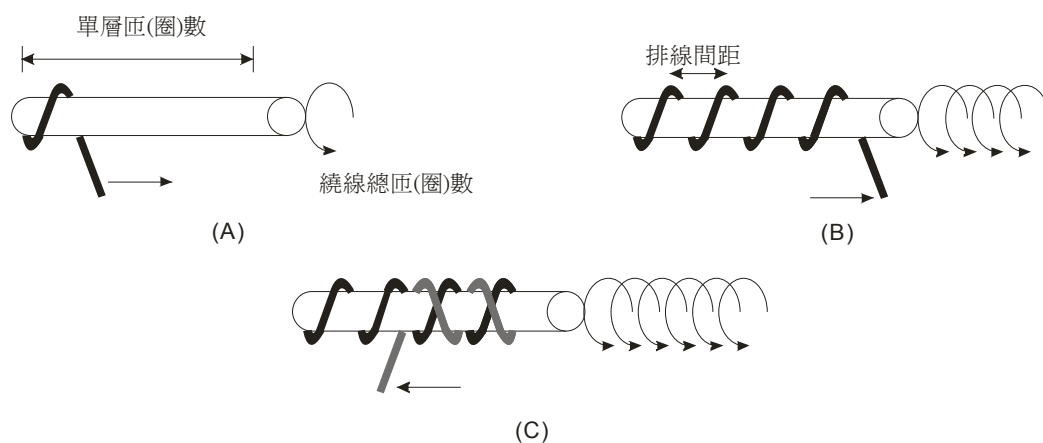
设备配置两个可调温度 220V 热风枪，在绕线前将模头预热，绕线后对线圈进行热风处理便于脱模。

(9) 电气控制

包含电气控制箱、触摸屏操作盒。采用 DVP-20PM00D 运动控制器作为控制核心，触摸屏作为人机交换，伺服电机作为执行机构，实现转轴与排线的精确控制，从而保证绕线的精度。

其中第(3)利用 DVP-PM 的一段速模式配合第(4)项利用 DVP-PM 的电子凸轮模式来达到高速绕线的动作，其余的项目可利用 DVP-PM 内容或是一般 PLC 的顺序逻辑动作即可达成。

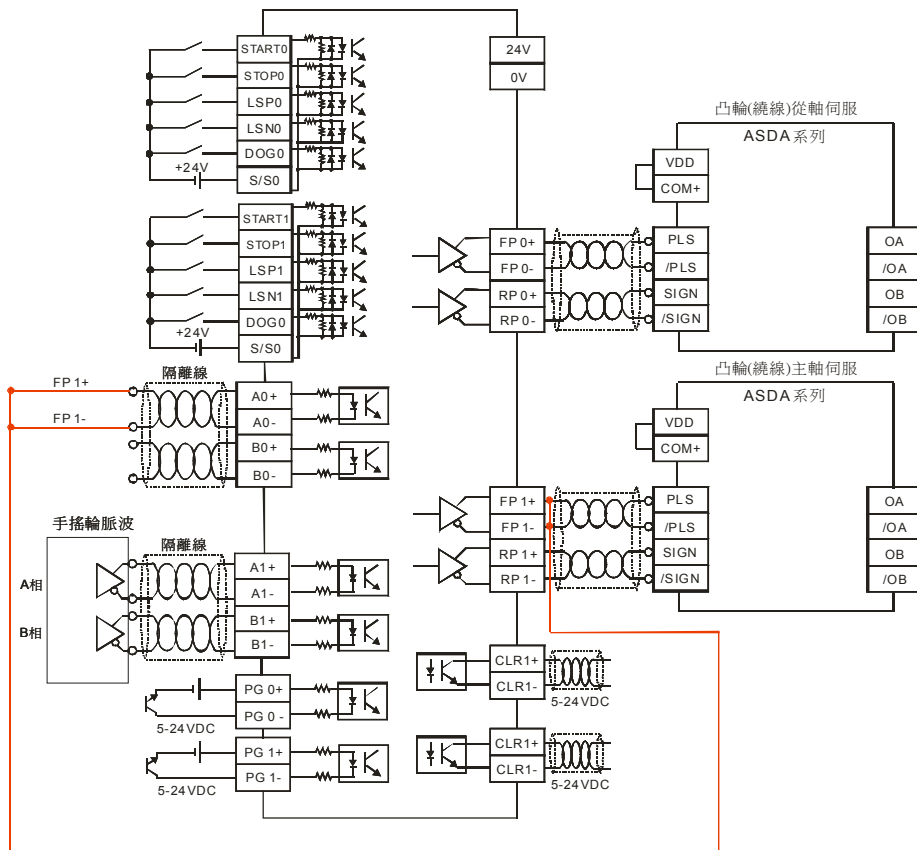
绕线机动作如下图，排线(从)轴追随着绕线(主)轴的转动在单层匝(圈)数的范围内来回摆动，彼此的关系成一定的比例。一开始排线轴位于单层匝数的最左侧位置，当绕线轴转动一匝时，排线轴移动一个排线间距，如图(A)所示，当排线轴移动到单层匝数范围最右侧时，如图(B)所示，排线追随的方向变更为向左，如图(C)所示，当排线轴再移动到单层匝数范围最左侧时，排线追随的方向再变更为向右。



9 电子凸轮

由以上的动作来看可以知道绕线机有 3 个主要的输入参数分别是 1.单层匝数、2.绕线总匝、3.排线间距，其中排线间距等于线径加上线与线间的间距。另外，还需要知道绕线机机械参数(mm/pulses)才能确定绕轴 / 排轴间的比例关系，机械参数包括机构及伺服参数，伺服参数等于电子齿轮比乘上马达转一圈所需脉冲数(pulses/revolutiON)，而机构参数则等于一匝移动多少距离(mm/revolutiON)；机构参数除以伺服参数才是整体的机械参数(mm/pulses)。在知道输入条件后，考虑绕线机其动作在排在线一个周期接着一个周期不断重复，所以采用 DVP-PM X 轴电子凸轮的 Cyclic 模式，而绕线方面的动作是同一方向不断前进，所以采用 DVP-PM Y 轴的一段速。

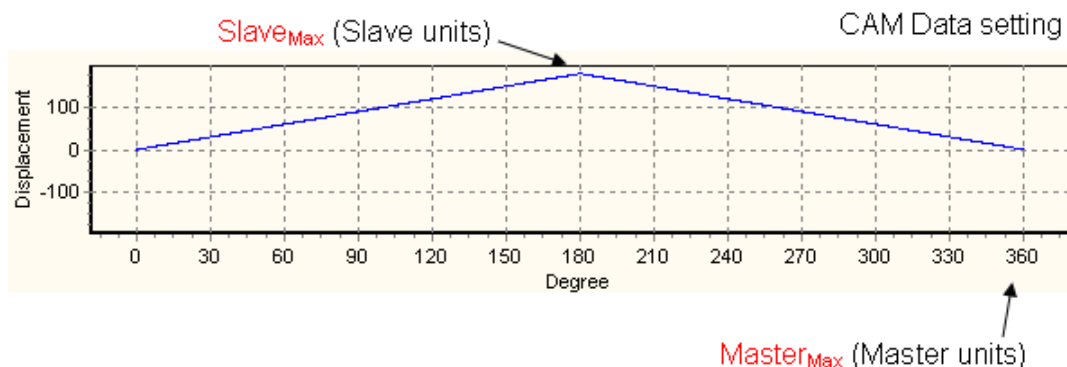
9.6.1.2 硬件配线



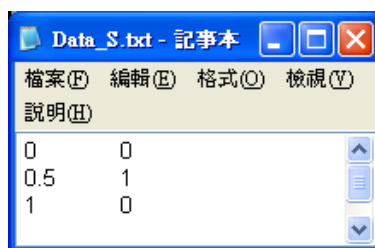
在上面的接线图里，将 Y 轴输出 FP1+、FP1- 接到 X 轴的手摇轮 / 电子凸轮输入 A0+、A0-，由于这样的接线方式，绕轴的旋转方向可以是顺时针或逆时针，对排轴不会造成任何影响，因此这里只接单向即可。

9.6.1.3 建立电子凸轮曲线

在 PMSoft 中建立 E-CAM Chart 如下图：



在 E-CAM Data 的建立上采用汇入的方式，档案 Data_S.txt 内容为：



Data_S.txt 内容设定 SlaveMax=1、MasterMax=1，这样的电子凸轮来做为绕线机的最小单位，之后可以利用 D1816 之 Bit1/Bit0 设定电子凸轮运行于机械单位，再依绕线输入的条件决定电子凸轮放大多少倍率；使用者也可直接设定电子凸轮运行于马达单位，接着再用 DTO 动态修改 SlaveMax/MasterMax。在知道输入条件后，需要换算出的设定有 1. 绕线(主/Y)轴一段速的距离、2. SlaveMax/MasterMax，接下来说明在输入条件下如何求得这些设定。

输入条件	单层匝数		N1	
	绕线总匝数		N2	
	排线间距 (mm)		D	
	绕线(主/Y)轴	机构参数 (mm/revolutiON)		因为主轴是直接驱动，所以没有实际的行走距离，但绕线的基本要求是主轴走一匝，从轴移动多少距离，因此主轴的机构参数可以看成跟从轴相同。 $A_{Master} = A_{Slave}$
		伺服参数 (pulses/revolutiON)		B_{Master}
		机械参数 (mm/pulses)		$C_{Master} = A_{Master}/B_{Master}$
排线(从/X)轴	机构参数 (mm/revolutiON)		A_{Slave}	
	伺服参数 (pulses/revolutiON)		B_{Slave}	
	机械参数 (mm/pulses)		$C_{Slave} = A_{Slave}/B_{Slave}$	
设定	绕线(主/Y)轴	一段速距离 (pulses)	$=N2 \times B_{Master}$	
	排线(从/X)轴	Master _{Max} (pulses)	$=2 \times N1 \times B_{Master}$ 上面的 2 是由于建立的电子凸轮代表的是双层的排线动作。	
		Slave _{max} (pulses)	$=N1 \times D/C_{Slave}$	

下面分别以二个参数来说明：

■ 参数 1

9 电子凸轮

输入条件	单层匝数		$N1 = 10$
	绕线总匝数		$N2 = 80$
	排线间距 (mm)		$D = 0.2$
	绕线(主/Y)轴	机构参数 (mm/revolutiON)	因为主轴是直驱，所以没有实际的行走距离，但绕线的基本要求是主轴走一匝，从轴移动多少距离，因此主轴的机构参数可以看成跟从轴相同。 $A_{Master} = A_{Slave}$
		伺服参数 (pulses/revolutiON)	$B_{Master} = 3600$
		机械参数 (mm/pulses)	$C_{Master} = A_{Master}/B_{Master}$
	排线(从/X)轴	机构参数 (mm/revolutiON)	$A_{Slave} = 10$
		伺服参数 (pulses/revolutiON)	$B_{Slave} = 10000$
		机械参数 (mm/pulses)	$C_{Slave} = A_{Slave}/B_{Slave}$
设定	绕线(主/Y)轴	一段速距离 (pulses)	$=N2 \times B_{Master} = 80 \times 3600 = 288000$
	排线(从/X)轴	$Master_{Max}$ (pulses)	$=2 \times N1 \times B_{Master} = 2 \times 10 \times 3600 = 72000$ 上面的 2 是由于建立的电子凸轮代表的是双层的排线动作。
		$Slave_{max}$ (pulses)	$=N1 \times D/C_{Slave} = 10 \times 0.2 / (0.1/100)$ $=2000$

■ 参数 2

输入条件	单层匝数		$N1 = 20$
	绕线总匝数		$N2 = 100$
	排线间距 (mm)		$D = 0.3$
	绕线(主/Y)轴	机构参数 (mm/revolutiON)	因为主轴是直驱，所以没有实际的行走距离，但绕线的基本要求是主轴走一匝，从轴移动多少距离，因此主轴的机构参数可以看成跟从轴相同。 $A_{Master} = A_{Slave}$
		伺服参数 (pulses/revolutiON)	$B_{Master} = 3600$
		机械参数 (mm/pulses)	$C_{Master} = A_{Master}/B_{Master}$
	排线(从/X)轴	机构参数 (mm/revolutiON)	$A_{Slave} = 10$
		伺服参数 (pulses/revolutiON)	$B_{Slave} = 10000$
		机械参数 (mm/pulses)	$C_{Slave} = A_{Slave}/B_{Slave}$
设定	绕线(主/Y)轴	一段速距离 (pulses)	$=N2 \times B_{Master} = 100 \times 3600 = 360000$
	排线(从/X)轴	$Master_{Max}$ (pulses)	$=2 \times N1 \times B_{Master} = 2 \times 20 \times 3600 = 144000$ 上面的 2 是由于建立的电子凸轮代表的是双层的排线动作。
		$Slave_{max}$ (pulses)	$=N1 \times D/C_{Slave} = 20 \times 0.3 / (0.1/100)$ $=6000$

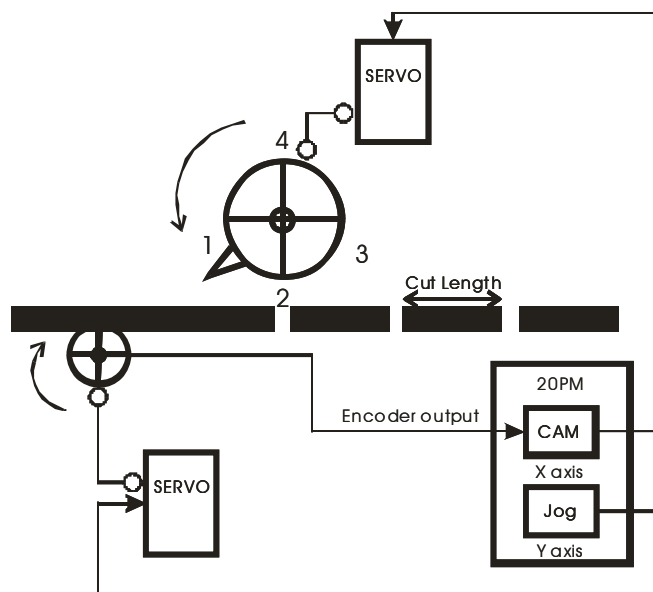
9.6.2 飞剪应用

在送料裁切应用上，传统的做法是使用走停式，送料轴先走到固定的长度，随后剪料轴再动作，之后不断重复“送料停”及“剪料停”的过程。这种方法的缺点在送料轴走停的过程中，所需要的加减速使生产效益无法提高，因此新的做法是采用送料不停的方式来达成，一般有二种送料裁切的方法分别为飞剪跟追剪，两者的差异为追剪是往返运动，而飞剪是为同向运动，所设定的 CAM 表曲线也不同。而另一种厚料裁切属于同向剪切的一种，和一般飞剪不同之处主要在于切刀从下刀至切断材料这个过程中速度同步区的曲线是不断变化，并非保持常数比例。以下将分别对这三种应用详细说明。

9.6.2.1 飞剪动作说明

当飞刀在进行裁剪时，送料的输送轴并不会停止，所以当飞刀轴的刀接触到被切料时，其运转的速度必须与送料轴同速运行。系统在进行裁剪时，若裁刀速度比送料的输送系统运行速度慢，则会造成物料的堆挤；而若裁刀速度比送料的输送系统运行速度快，则会拉伸物料，破坏物料。

首先说明飞剪的设定方式，假设配线如下图所示，其中 1, 2, 3, 4 分别为同步起点、同步中点、同步离开点、及等待点(起始点)，其动作为随着主轴(Y 轴)的移动一开始凸轮轴(X 轴)停于位置 4，之后前进加速至位置 1 达到速度同步，再持续到位置 3，接着减速后回到位置 4，之后不断重复这个动作。

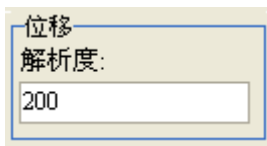


9.6.2.2 建立电子凸轮曲线

在 CAM 的规划上 PMSoft 是先产生位移相对关系再生成速度及加速度关系，但在这里以相对速度的关系来规划，因此要先产生速度关系再产生位移及加速度，但由于以位移关系图来推算速度关系对使用者而言不甚方便，因此使用者可以先在 PMSoft 将位移数据(含速度、加速度数据)汇出，再以此功能 [匯入速度資料](#)，把汇出的速度数据汇入，将所需的速度关系建立在位移关系图上，步骤如下：

9 电子凸轮

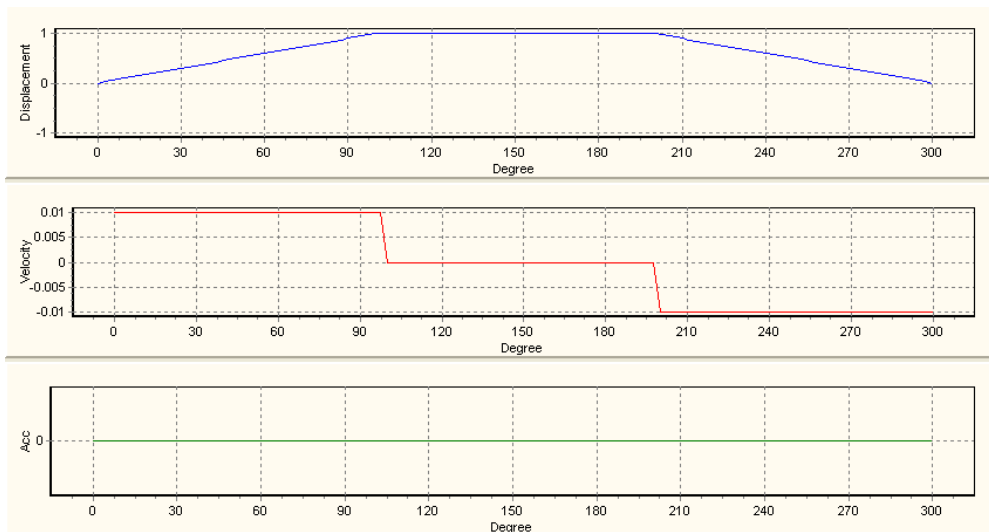
1. 设定所需的分辨率为 200 点。



2. 将速度关系建立在位移关系图上，假设位置 1 在主轴位置 100，位置 2 在主轴位置 200，位置 3 在主轴位置 300。

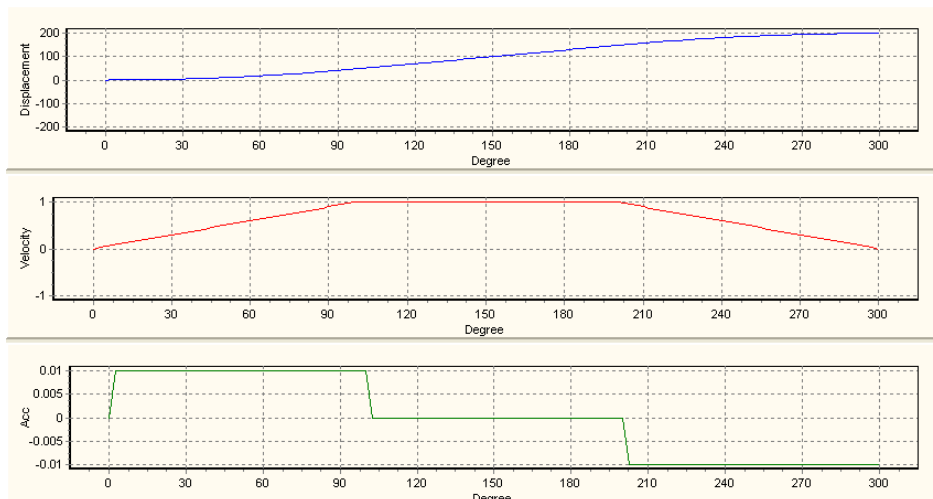


可得位移关系图如下：



3. 将位移关系图汇出，再将其当成速度数据汇入，即可得到所要的速度关系图。



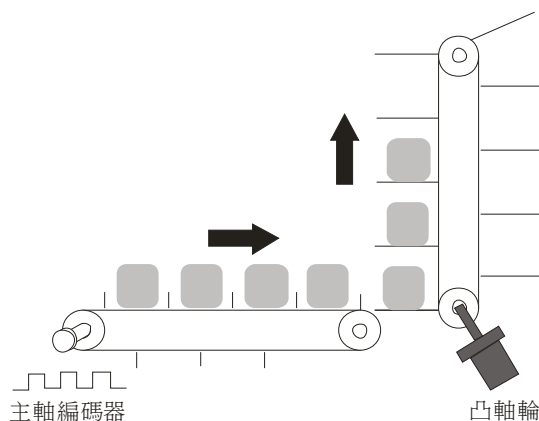


在位移关系图中，从轴转一圈的脉冲量 $Slavemax=200$ 可以由计算速度关系图的面积求得

$$(1*100/2+1*(200-100)+1*(300-200)/2)=200$$

而主轴的位移量(脉冲量)跟裁切长度是相等的，使用者可以利用 9.3.1.6 比例设定的说明利用倍率来放大至需要的大小。

其它应用如下图所示，使用在同步输送带上类似于飞剪，同样有标记但垂直输送带的凸轮曲线最好设成等速，如果使用与飞剪相同的曲线，则会造成垂直输送带加减速频繁而走走停停。

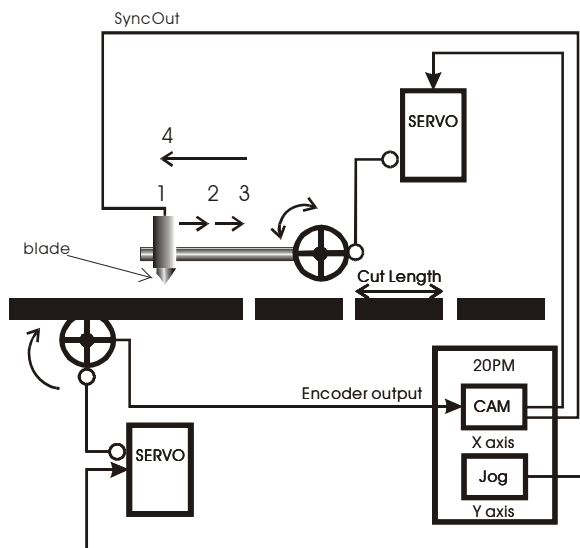


9.6.3 追剪应用

所谓追剪系统，是当系统在进行裁剪的同时，送料轴并不会停止，所以凸轮轴在裁切时必须与送料轴保持同速，而且同速时间必于足够让裁刀完成裁剪并脱离到安全的位置。追剪凸轮轴会带动裁刀及其整组裁切机构的移动，使其在裁剪时能与主轴保持等速。

9.6.3.1 追剪动作说明

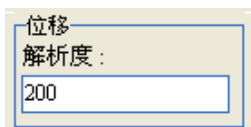
假设配线如下图所示，其中 1, 2, 3, 4 分别为等待点(起始点)、同步点、同步离开点、及等待点(起始点)，其动作为随着主轴(Y 轴)的移动，一开始凸轮轴(X 轴)停于位置 1，之后前进加速至位置 2 达到速度同步，持续到位置 3，接着减速后往相反方向回到位置 4(假设位置 1 跟位置 4 是相同的)，之后不断重复这个动作。



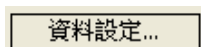
9.6.3.2 建立电子凸轮曲线

在 CAM 的规划上 PMSoft 是先产生位移相对关系再生成速度及加速度关系，但在这里则以相对速度的关系来规划，因此要先产生速度关系再产生位移及加速度，但由于以位移关系图来推算速度关系对使用者而言不甚方便，因此使用者可以先在 PMSoft 将位移数据(含速度、加速度数据)汇出，再以此功能 [匯入速度資料](#)，把汇出的速度数据汇入，将所需的速度关系建立在位移关系图上，步骤如下：

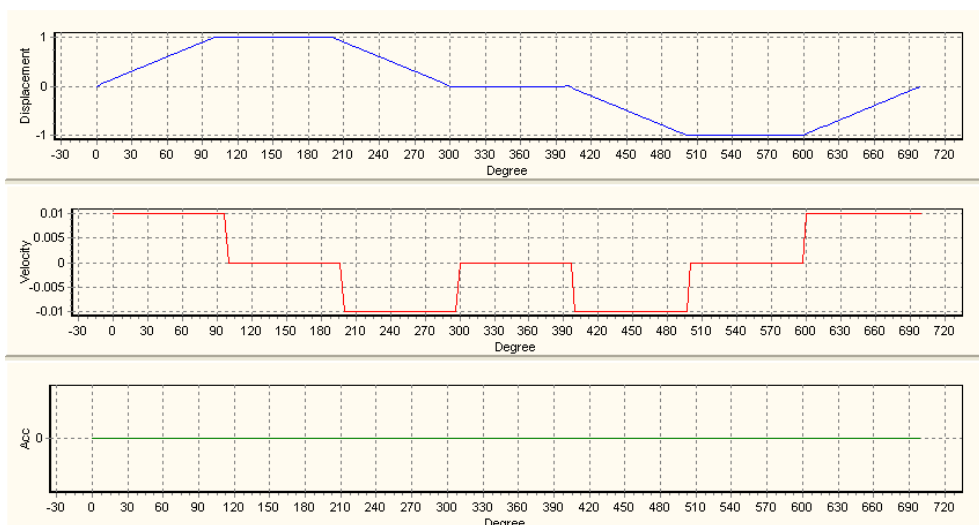
1. 设定所需的分辨率为 200 点。



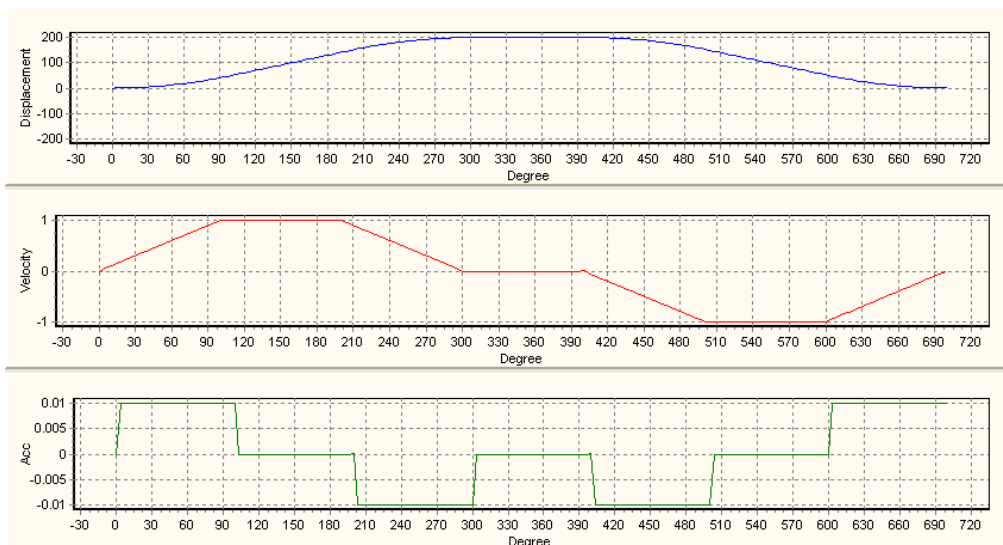
2. 将速度关系建立在位移关系图上，假设位置 2 在主轴位置 100，位置 3 在主轴位置 200，位置 4 在主轴位置 700。



可得位移关系图如下：



3. 将位移关系图汇出，再将其当成速度数据汇入，即可得到所要的速度关系图。



在位移关系图中从轴转一圈的脉冲量 $Slavemax=200$ 可以由计算速度关系图的面积求得

正面积：

$$(1 \cdot 100/2 + 1 \cdot (200 - 100) + 1 \cdot (300 - 200)/2) = 200$$

负面积：

$$((-1) \cdot 100/2 + (-1) \cdot (200 - 100) + (-1) \cdot (300 - 200)/2) = -200$$

之后使用者可以设定比例利用倍率来放大至需要的大小。跟飞剪不同的是在这里还需要设定 Syncout 输出信号启动的范围，可使用电子凸轮同步位置上 / 下限功能达成。另外，在反转回位置 4 的速度上使用者也可以设定成以较高速度运行来节省整个周期的时间，举例如下：

區段	角度[deg]	行程[%]	CAM 曲線	解析度
	0	0	NA	NA
1	100	1	Const Speed	33
2	200	1	Const Speed	33
3	300	0	Const Speed	33
4	400	0	Const Speed	33
5	500	-2	Const Speed	33
6	600	0	Const Speed	35
7				
8				
9				
10				

初始行程設定
 行程[%]:
 0

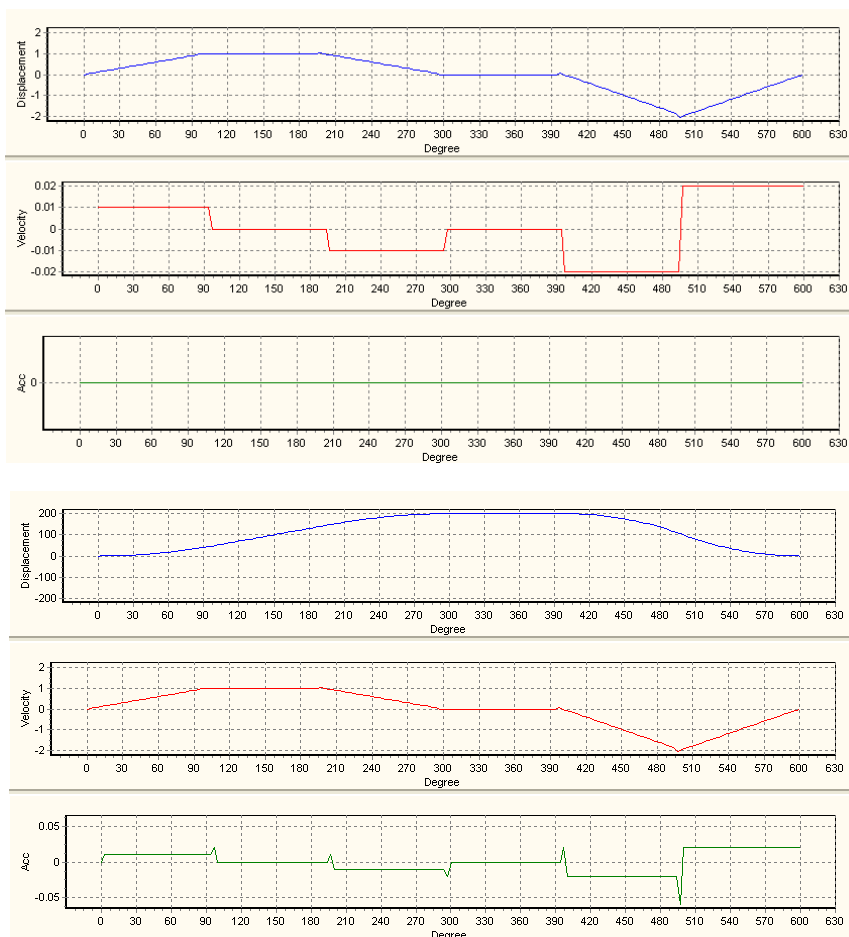
在设定里位置 1~3 的设定跟之前相同，但在位置 4 改成了 600，同时在位置 500 时反转的速度设定为 2 倍，如主轴以 100kHz 运行时可节省 $700 - 600 = 100\text{pulses}$ ，即约 1ms 的时间。要注意的是如果位置 1 跟位置 4 要相同(即回到同一点)，在速度规划上正的面积要跟负的面积相同。

正面积:

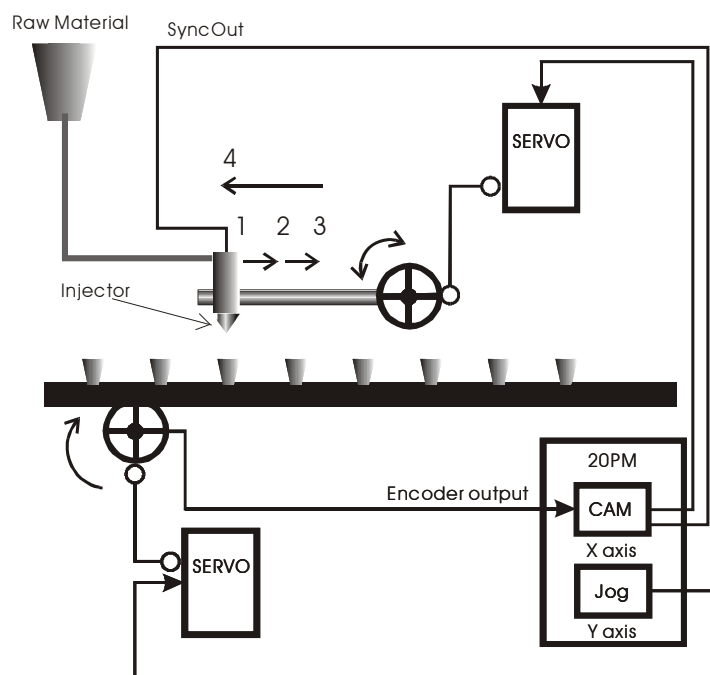
$$(1 \times 100 / 2 + 1 \times (200 - 100) + 1 \times (300 - 200) / 2) = 200$$

负面积:

$$(-2 \times (600 - 400) / 2) = -200$$

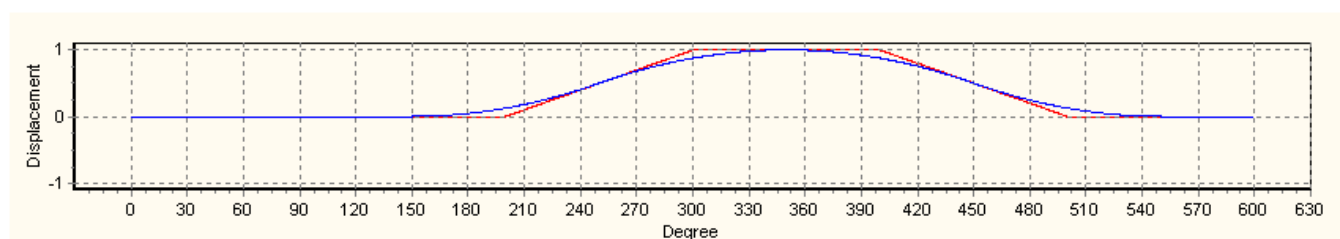


另一种追剪可以套用的例子是饮料充填，其动作跟前述的追剪例子大致相同，只是机构由刀子换成了注射口，配线图如下图示。



饮料充填机

在 PMSoft 中另外提供了 B-Spline 来将曲线平滑化，让整个动作更为平顺，对机构来说可以减少消耗增长其使用寿命。目前提供的 B-Spline 可以使得规划出来的曲线在 2 次微分时仍具连续的特性，也就是说规划 B-Spline 的位移曲线其速度曲线及加速度曲线都会是连续的。下面举例其主从轴的位置相同，不同的在于 CAM Curve 一个选择 CONST Speed(红色线)，一个选择 B-Spline(蓝色线)。

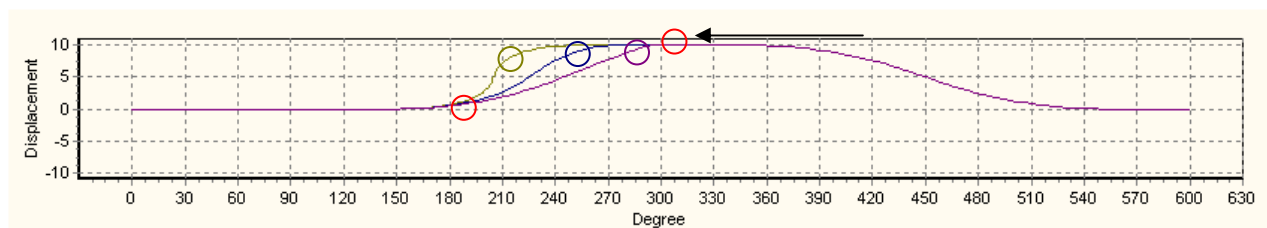


Sect. No.	Angle[deg]	Stroke[%]	CAM Curve	Resolution
	0	0	NA	NA
1	100	0	Const Speed	341
2	200	0	Const Speed	341
3	300	1	Const Speed	341
4	400	1	Const Speed	341
5	500	0	Const Speed	341
6	600	0	Const Speed	343
7				
8				
9				
10				

Sect. No.	Angle[deg]	Stroke[%]	CAM Curve	Resolution
	0	0	NA	NA
1	100	0	BSpline	341
2	200	0	BSpline	341
3	300	1	BSpline	341
4	400	1	BSpline	341
5	500	0	BSpline	341
6	600	0	BSpline	343
7				
8				
9				
10				

9 电子凸轮

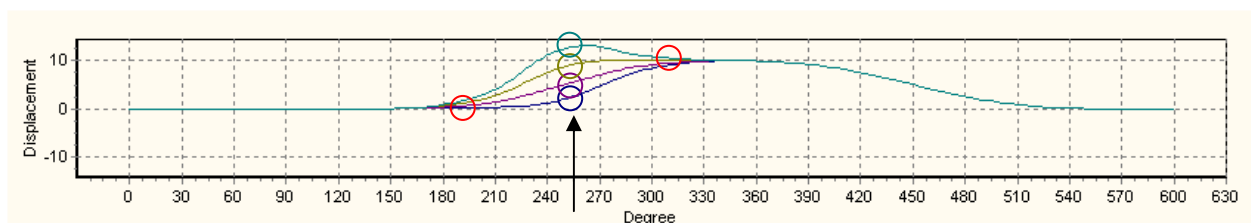
B-Spline 曲线有一个特性，即主从轴的位置只会影响到局部的曲线，下面的例子在主轴位置 200 跟 300 中间分别加入一个点为 210、250、290，可以观察到它的曲线变化。



Sect. No.	Angle[deg]	Stroke[%]	CAM Curve	Resolution
	0	0	NA	NA
▶ 1	100	0	BSpline	28
2	200	0	BSpline	28
3	210	10	BSpline	28
4	300	10	BSpline	28
5	400	10	BSpline	28
6	500	0	BSpline	28
7	600	0	BSpline	32
8				
9				
10				

Save Draw
Load OK
Clear Cancel

上面的例子示范了中间点横向(主轴位置)的影响，下面再举一个例子说明中间点纵向(从轴位置)的影响其从轴位置分别为 1、5、10、15。



Sect. No.	Angle[deg]	Stroke[%]	CAM Curve	Resolution
	0	0	NA	NA
▶ 1	100	0	BSpline	28
2	200	0	BSpline	28
3	250	15	BSpline	28
4	300	10	BSpline	28
5	400	10	BSpline	28
6	500	0	BSpline	28
7	600	0	BSpline	32
8				
9				
10				

Save Draw
Load OK
Clear Cancel

10.1 使用设置介绍

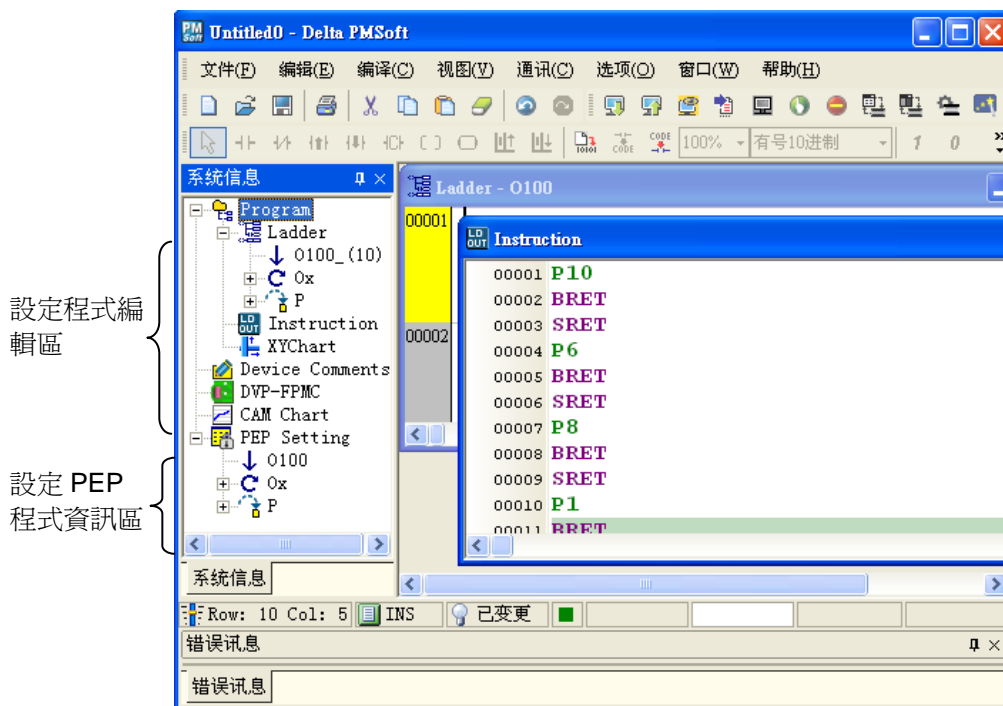
DVP-20PM 主机可透过 PMSoft 软件提供使用者设置密码以保护编写的程序。下载程序至 20PM 主机之前设置密码，来达到保护程序被不正当获取或修改的功能，因此在上传有加密的程序时，必须输入正确密码，才能上传程序到 PMSoft 提供使用者修改。PMSoft 可分别对主程序 O100、Oxn (n: 0~99, 100 个) 运动程序及 Pm (m: 0~255, 256 个) 子程序做各别加密设置。

使用设置主要分为三个部份，分别为系统信息区设置、下载设置及上传设置，下列将分几个单元介绍密码锁定设置(PEP Setting)。

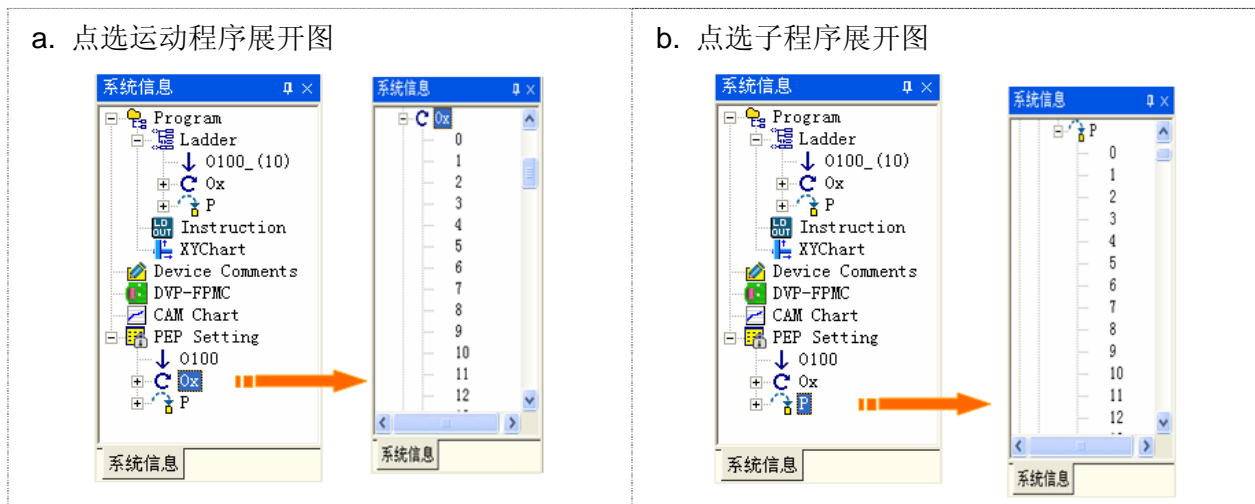
10.1.1 系统信息区

系统信息区主要分成两部份：程序编辑区(Program)，用于切换程序编辑窗口及了解目前系统程序使用状态；特定程序加密设置区(PEP Setting)，用于保护程序的加密，编辑窗口中总共有 357 组程序可被设置加密保护，O100 主程序有 1 组，Oxn 运动程序有 100 组，及 Pm 子程序有 256 组。下列将系统信息分四种菜单详细说明：

- 显示系统信息：系统信息列可显示系统信息及指定的程序加密保护设置。

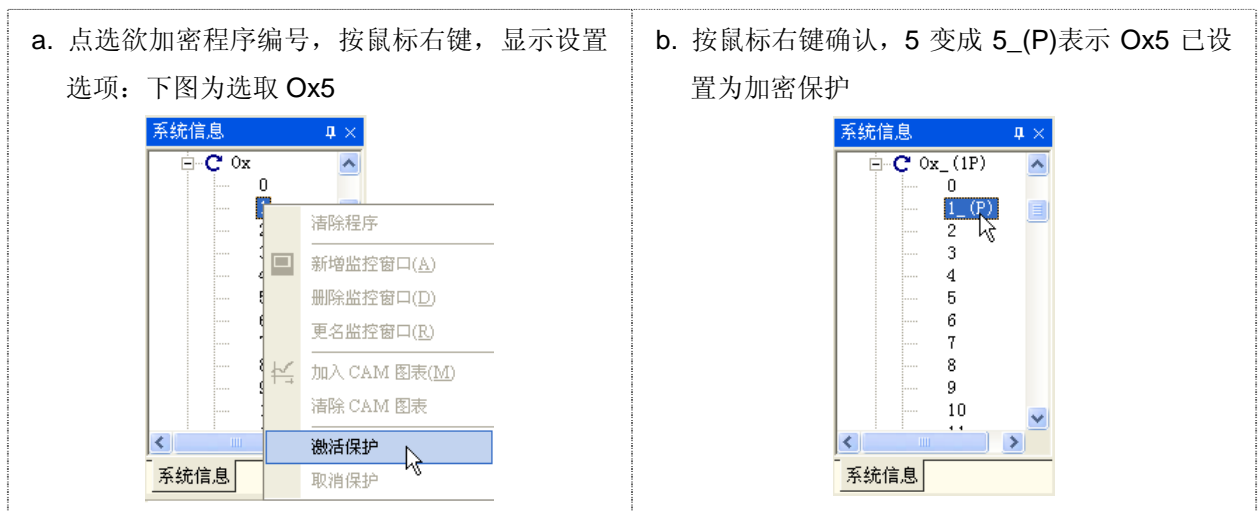


- 运动程序和子程序展开：由下图(a)(b)所示，点选运动子程序或是子程序将分别展开所属的程序编号，分别有 100 个和 256 个。



■ 特定程序加密的套用设置:

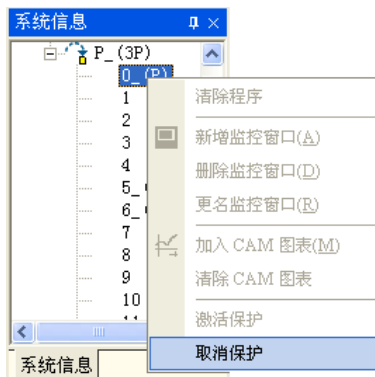
1. 在欲进行加密的运动程序或子程序编号上按鼠标右键，会显示启动保护(Enable Protection)的功能选项(如下图 a 示)。
2. 此时被设置的程序编号会多加一个 _(P) 符号，表示此程序将被加密保护，如下图 b 所示，Ox_(1P) 表示一个运动程序将被加密，以此类推 Ox_(4P)表示有四个运动程序被设置为需要加密，而子程序也是同样原理，但主程序只有一个，请直接在 O100 节点上设置启动保护，使其为 O100_(P)。



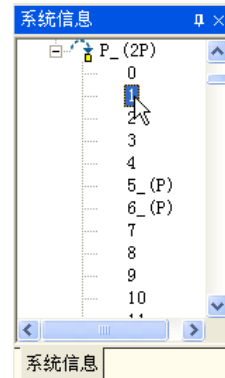
■ 解除保护的套用设置:

1. 在欲进行解密运动程序或子程序编号上按鼠标右键，会显示取消保护(Disable Protection)的功能选项，如下图(a)示。
2. 此时原来被设置加密的程序编号，由 _(P) 符号变成只有程序编号，如下图 b 所示，若原本是 P_(3P)表示三个子程序被加密，经过解除一个保护设置之后，即会变为 P_(2P)，以此类推。

- a. 點選欲解密程序编号，按鼠标右键，显示设置选项：下图为选取 6_(P)




- b. 按鼠标右键确认，6_(P)变成 6 表示 P6 已设为非加密保护




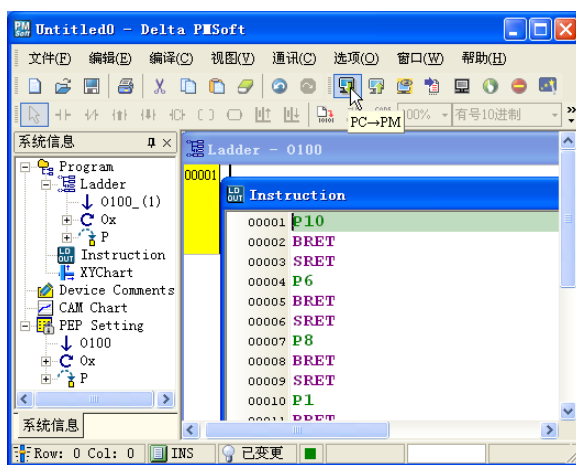
10.1.2 下载程序设置

下载程序时，会弹跳出功能窗口，可勾选是否套用 PEP 保护程序设置，若无勾选套用 PEP 保护程序，PMSoft 会把所有的程序传送到主机一般程序储存区，若勾选套用 PEP 保护程序，PMSoft 会要求设置被加密保护的密码，之后下载程序时，会把未加密设置的程序存放到一般程序储存区，而有加密设置的程序或 CAM 数据则会被存放程序保护储存区。下列将分两种状况详细说明其操作步骤：

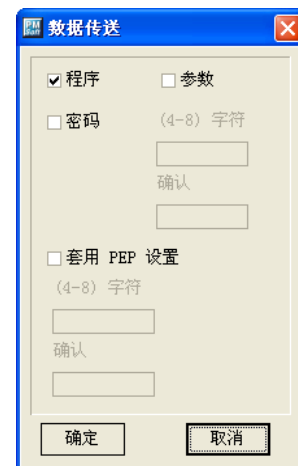
■ 勾选套用(PEP)保护程序设置：

1. 选择『通讯 (C)』功能并點選 PC→PM(D)，或在工具列上點選图示  (如下图 a)，会显示功能窗口(如下图 b)。

- a. 點選工具列上的图示 



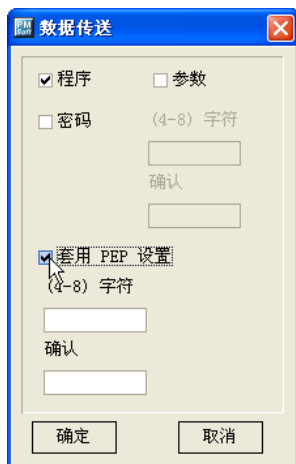
- b. 设置是否套用保护程序设置



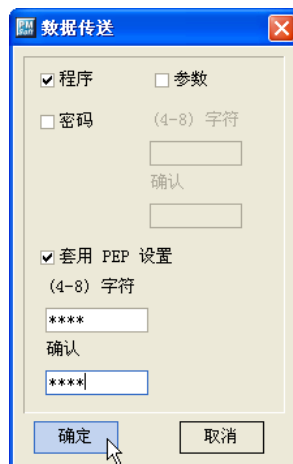
2. 勾选套用 PEP 保护设置(如下图 a)，此时需设置 4~8 字符的加密保护密码，请分别输入密码与密码确认字段后按确定(如下图 b)。若为第一次设置加密保护密码，在进行传输时，PMSoft 会直接执行下载程序或 CAM 数据到 DVP-PM 主机(如下图 c)。

10 程序加密设定

a. 勾选套用 PEP 设置



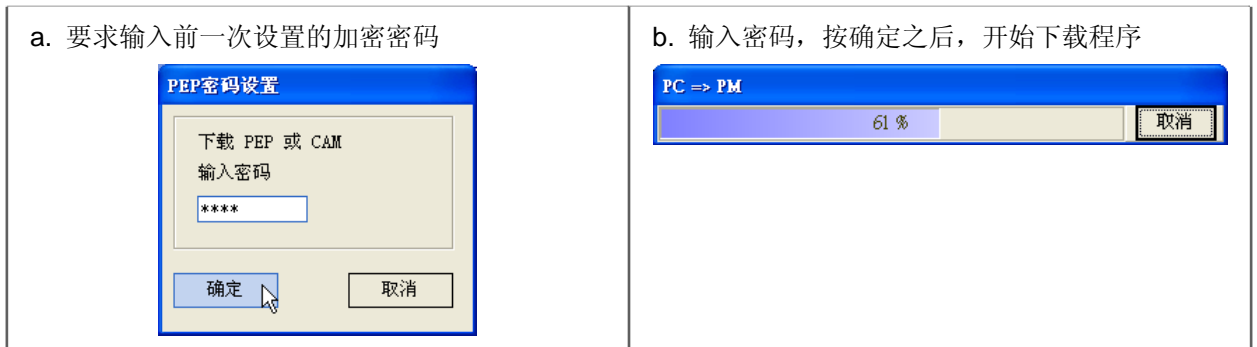
b. 分别输入密码与密码确定字段




c. 若是第一次设置密码，即开始下载程序

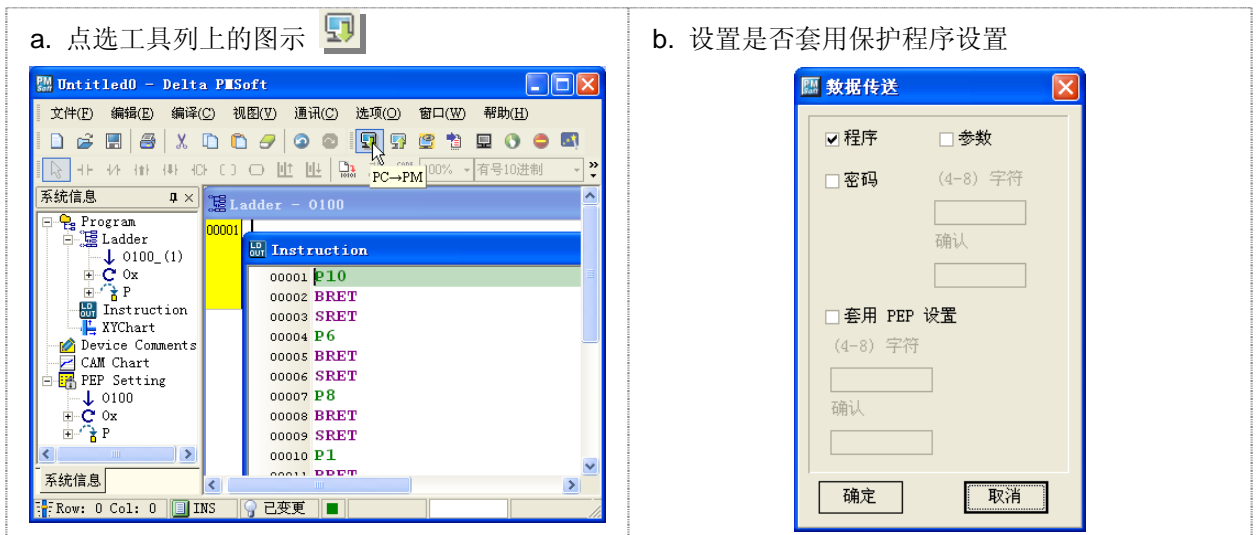


3. 在第 2 步骤执行传输时，若之前已有设置过加密保护程序，PMSoft 会要求输入前一次设置加密保护的密码(如下图 a)，按键确定后，才开始下载程序或 CAM 数据到 DVP-PM 主机(如下图 b)。

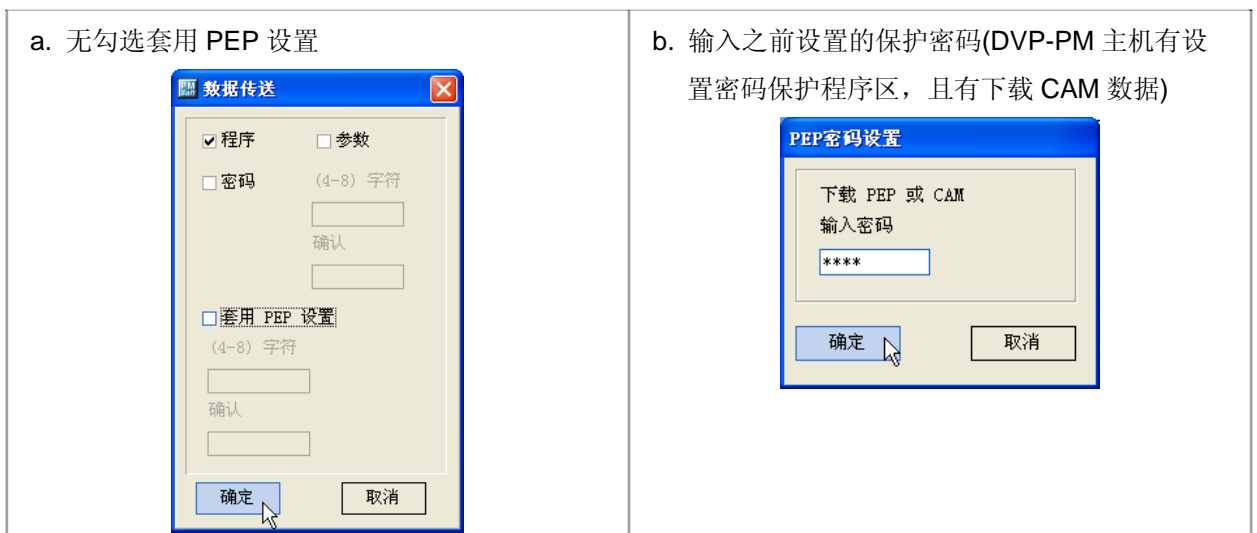


■ 无勾选套用(PEP)保护程序:

1. 选择『通讯 (C)』功能并点选 PC→PM(D)，或点选工具列上的图示  (如下图 a)，会显示功能窗口(如下图 b)。



2. 无勾选套用 PEP 设置(如下图 a)，若之前 DVP-PM 主机有设置 PEP 保护密码而未解除，且下载程序有 CAM 数据，PMSoft 会要求输入之前设置的密码 (解除密码以供 CAM 数据存放)，在输入密码按确定之后(如下图 b)，即开始执行下载程序到 DVP-PM 主机(如下图 c)。



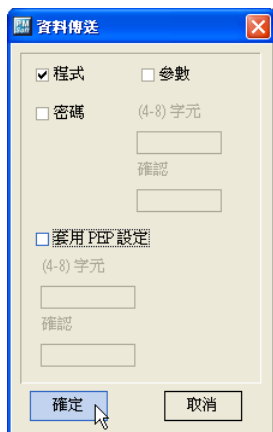
10 程序加密设定

c. 开始下载程序



3. 在第 2 步骤选取确定数据传送时，若之前 DVP-PM 主机有设置 PEP 保护密码而未解除，而下载程序无 CAM 数据，系统将直接下载程序到 DVP-PM 主机(如下图 b)。

a. 无勾选套用 PEP 设置

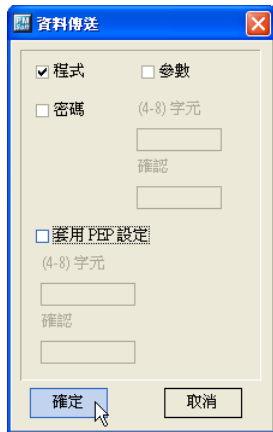


b. 开始下载程序(DVP-PM 主机有设置密码保护程序区，无 CAM 数据下载)



4. 在第 2 步骤选取确定数据传送时，若之前 DVP-PM 主机无设置 PEP 保护密码，而下载程序不论有无 CAM 数据，系统将直接下载程序到 DVP-PM 主机(如下图 b)。

a. 无勾选套用 PEP 设置




b. 开始下载程序(DVP-PM 主机无设置密码保护程序区，无论程序有无内含 CAM 数据)

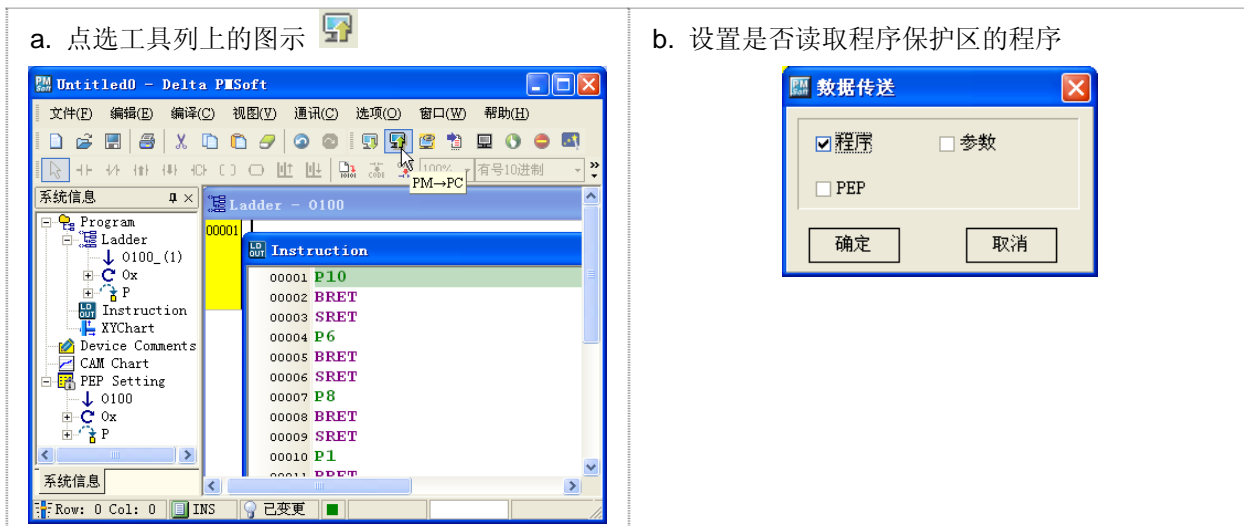


10.1.3 上传程序设置

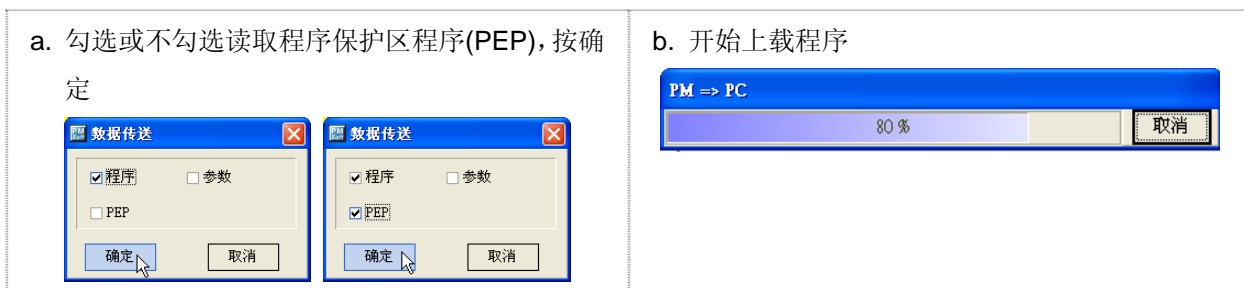
上载程序时，会弹跳出功能窗口提供使用者选取是否读取保护程序区的程序，若无勾选读取 PEP 保护程序，只会上传主机一般程序储存区的程序，若勾选读取 PEP 保护程序，则会要求输入设置的加密密码，之后才能上传主机一般程序储存区和保护程序区的程序。下列将分三种状况详细说明上传程序设置操作步骤：

■ DVP-PM 主机无设置保护程序：


1. 选择『通讯 (C)』功能并点选 PM→PC(U)，或在工具列上点选图示  (如下图 a)，会显示功能窗口(如下图 b)。

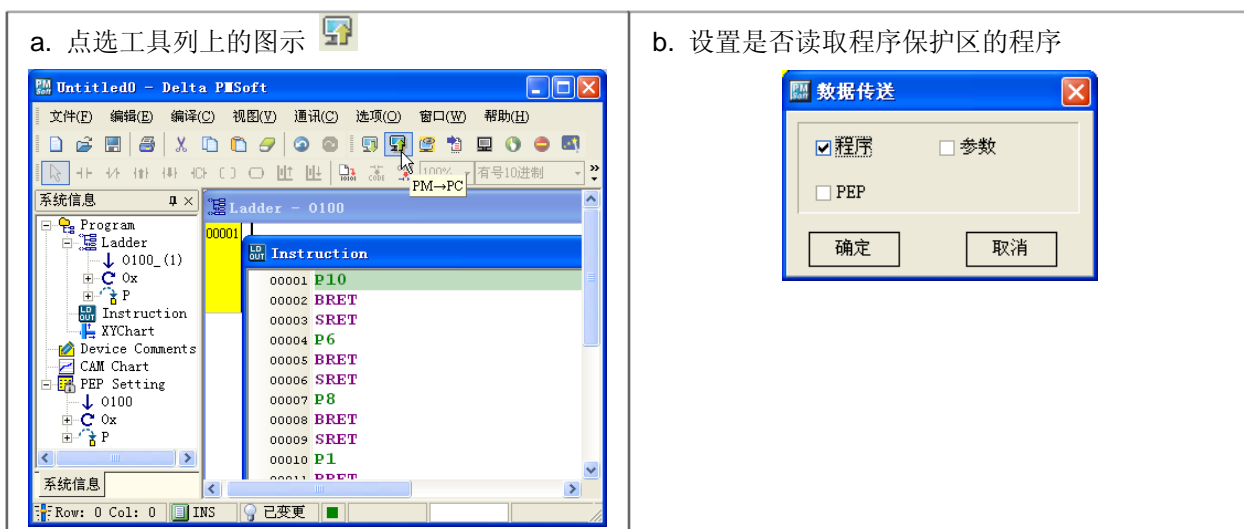


2. 无论是否有勾选读取程序保护区程序(PEP)，按确定后(如下图 a)，将直接读取一般程序或 CAM 数据至 PC (如下图 b)。



■ DVP-PM 主机有设置特定保护程序但无 CAM 数据:

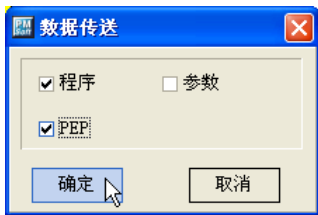
1. 选择『通讯 (C)』功能并点选 PM→PC(U)，或在工具列上点选图示  (如下图 a)，会显示功能窗口(如下图 b)。



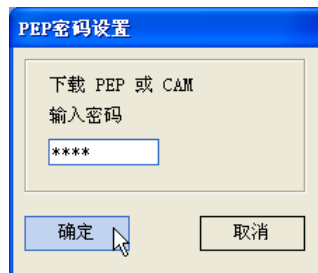
2. 若勾选读取特定保护程序(PEP)，按确定后(如下图 a)，将要求输入保护密码，在确认密码正确后(如下图 b)，即开始上传读取被设置保护程序至 PC (如下图 c)。

10 程序加密设定

a. 勾选读取程序保护区程序(PEP)，按确定



b. 在密码窗口，输入加密密码后按确定

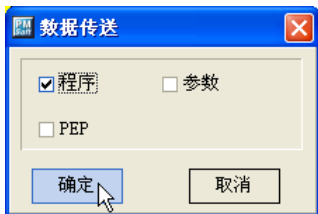


c. 开始上传程序

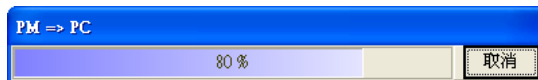


3. 若在第 2 步骤无勾选读取特定保护程序(PEP)，按确定键后(如下图 a)，系统将会直接上传程序至 PC (如下图 b)。


a. 无勾选套用 PEP 设置




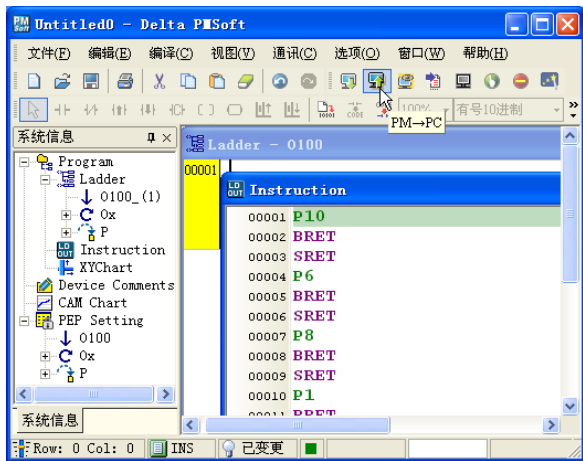
b. 系统上传程序(无设置读取密码保护程序区)



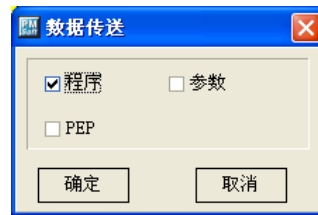
■ DVP-PM 主机有设置特定保护程序且含有 CAM 数据：

1. 选择『通讯 (C)』功能并点选 PM→PC(U)，或在工具列上点选图示  (如下图 a)，会显示功能窗口(如下图 b)。

a. 用鼠标点选图标工具列上的 

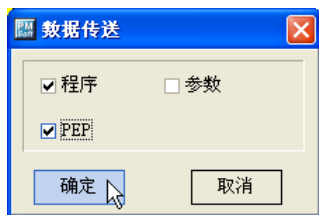


b. 设置是否读取程序保护区的程序

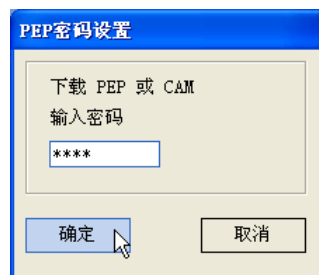


2. 若勾选读取程序保护区程序(PEP)，按确定后(如下图 a)，将要求输入特定保护密码，在确认密码正确后(如下图 b)，即开始上传读取一般程序、特定保护程序和 CAM 数据至 PC(如下图 c)。

a. 勾选读取程序保护区程序(PEP)，按确定



b. 在密码窗口，输入加密密码后按确定

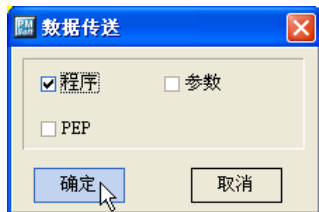


c. 开始上传程序



3. 若在第 2 步骤无勾选读取特定保护程序(PEP)，按确定键后，系统将会要求输入特定保护密码(如下图 b)，在确认密码正确后才开始上传读取一般程序和 CAM 数据(如下图 c)。

a. 无勾选读取程序保护区程序(PEP)，按确定



b. 在密码窗口，输入加密密码后按确定



c. 开始上传程序



此章节主要说明提升 G code 使用便利性及进阶强化 G code 应用,以下载 G code 檔的方式可提升 G code 檔使用便利性,而藉由平稳等速执行 G code 檔即为强化 G code 档应用,主要用于点胶机设备。

11.1 G code 档下载

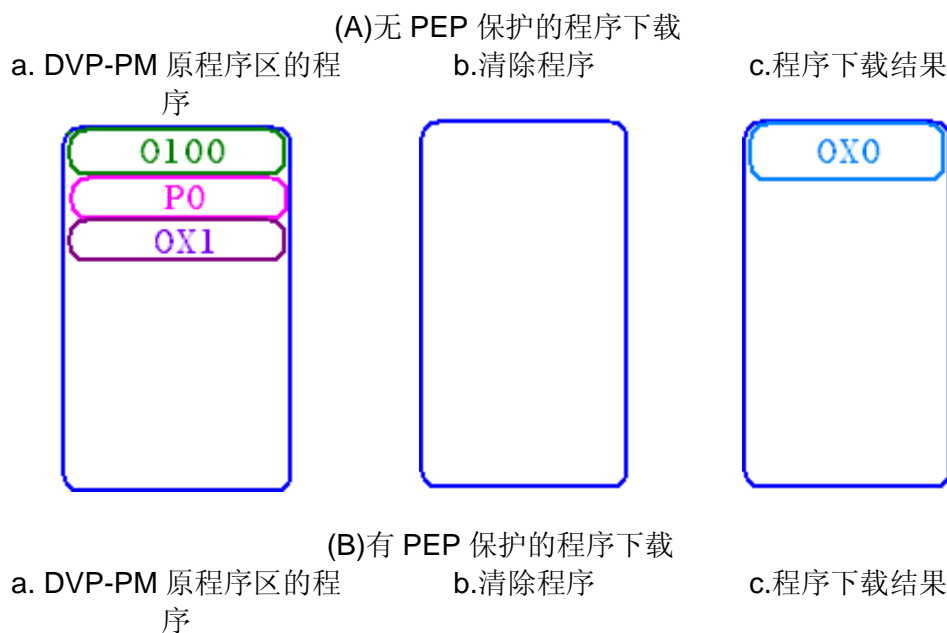
在产业界使用 G code 文件程序运作的 CNC (数值控制机),其操作方式及功能都固定,因此 PM 主机程序都固定不会变更。使用者可藉由以下三种方式预先设置 G code 档案来源,完成后在运作加工时,由人机选择 G code 档来源(选择人机链接库、记忆卡里面的程序或是与 PC 通讯而边传边运作方式),即可达成控制需求。G code 档案来源如下所示:

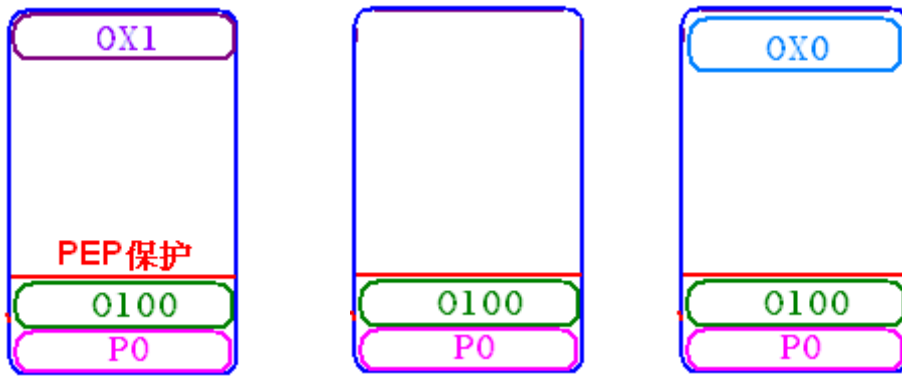
1. 在人机上面编辑并储存 G code 程序
2. 将市面上 CAM(计算机辅助制造)等软件转成 G code 文件,储存在记忆卡,插入 CNC 的记忆卡槽
3. 使用 PC(含 NB)下载 G code 檔至 CNC

为达到此需求,DVP-20PM00M 提供两种方式提供使用者下载 G code 文件至主机运行,分别为使用 PMGDL 软件以及使用特定的 HMI 下载。在了解上述两种方式之前,先说明 DVP-20PM00M 程序下载方式,再详细说明操作方法。

11.1.1 程序下载(PEP 保护)

PMSoft 下载程序至 DVP-PM 会把 DVP-PM 程序区没有 PEP 保护的程序清除掉,然后再下载程序至一般程序区,如下图(A)(B)OX0 所示。DVP-PM 的程序架构由 O100 主程序呼叫运动程序,或由主程序呼叫子程序进一步呼叫运动程序,故为避免不更动的主程序及子程序下载时被清除,而影响到呼叫运动程序的动作,习惯上将主程序及子程序设定 PEP 保护。





如上图，将不更动的程序如 O100 及 P0 使用 PEP 保护功能，然后将随时需要更换的 G code 文件程序如 OX0 下载至一般程序区(无设定 PEP 保护区块)。详细 PEP 操作说明请参考本手册第 10 章。

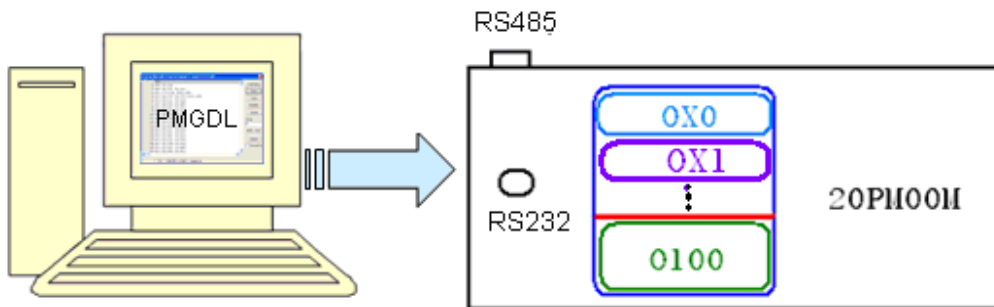
11.1.2 程序下载操作方法

■ PMGDL 软件下载

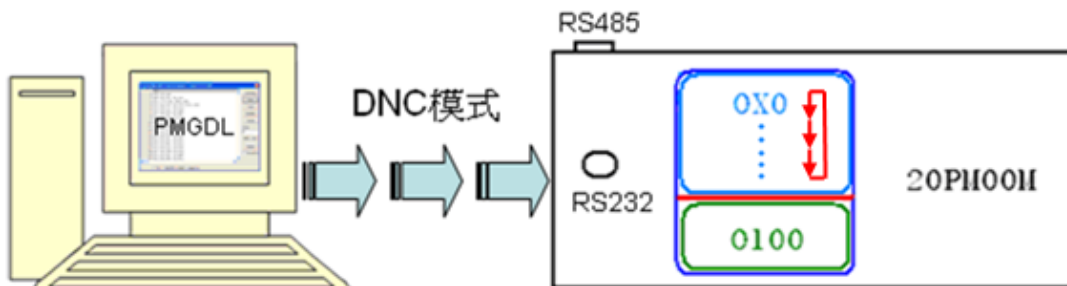
PMGDL 软件下载有两种方式：

1. 一般模式：下载 G code 档至 PM 后，再执行 OXn 运动程序。
2. DNC 模式：边传送边执行 OXn 运动程序。

下图(A)为 PMGDL 软件选择将 G code 文件下载至 PM 程序区 OX0、OX1...等，而下图(B)为 DNC 模式边传送边运作程序，适用于庞大 G code 文件程序，下图中 PM00M 中的 O100 程序需预先设定以 PEP 保护的方式下载至 PM，然后使 O100 运作设定 PM 的 COM1(RS232 模式)或 COM2(RS485 模式)与 PMGDL 通讯，接收 PMDGL 传送之 G code 文件程序。



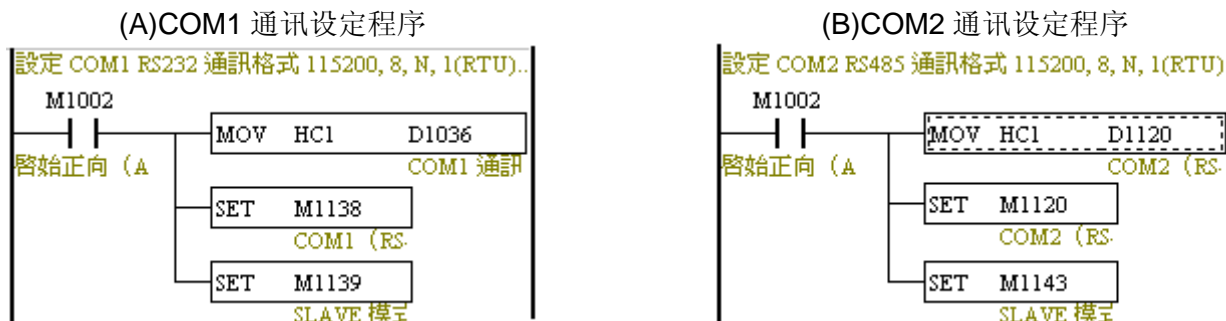
(A) 一般 G code 档下载



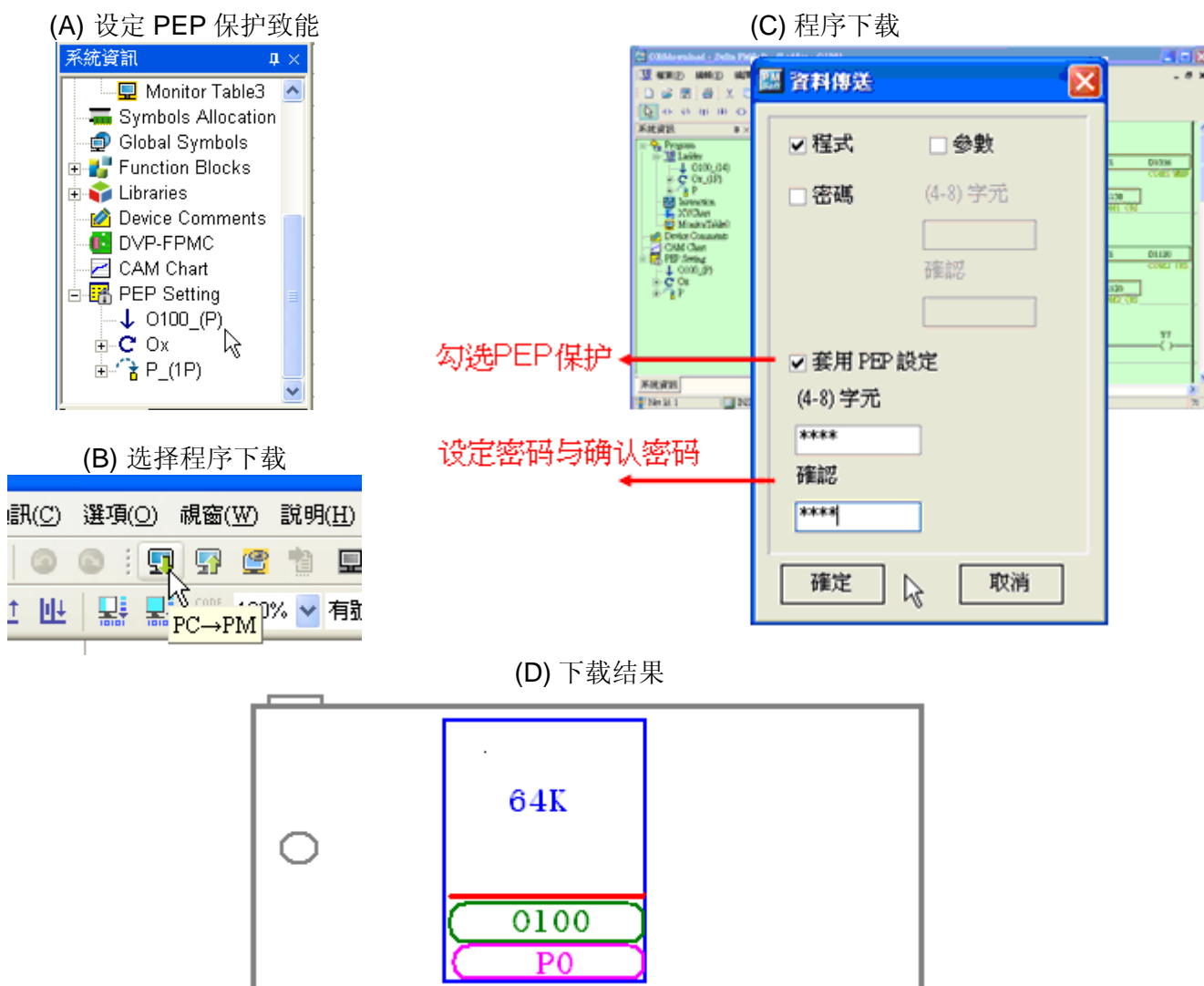
(B) DNC 模式下载

1. 一般模式:

Step 1: 编写 O100 主程序。程序必须含 M code 处理及设定一个通讯端口(COM1/COM2), 格式为 115200, 8, N, 1(RTU)与 PMGDL 软件通讯。下图(A)为设定通讯端口 COM1 程序, 下图(B)为设定通讯端口 COM2 程序。



Step 2: 使用 PMSoft 将 O100 程序或将不变更的子程序设定 PEP 保护致能, 下载至 PM, 如下图所示。



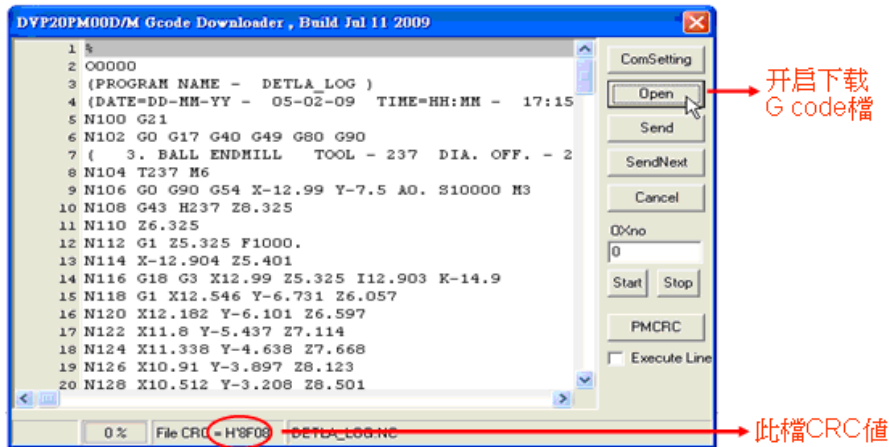
11 G 码应用

Step 3: 开启 PMGDL 软件，设定通讯端口(计算机与 PM 通讯端口编号)，如下图所示，而通讯格式 115200, 8, N, 1 不可修改。

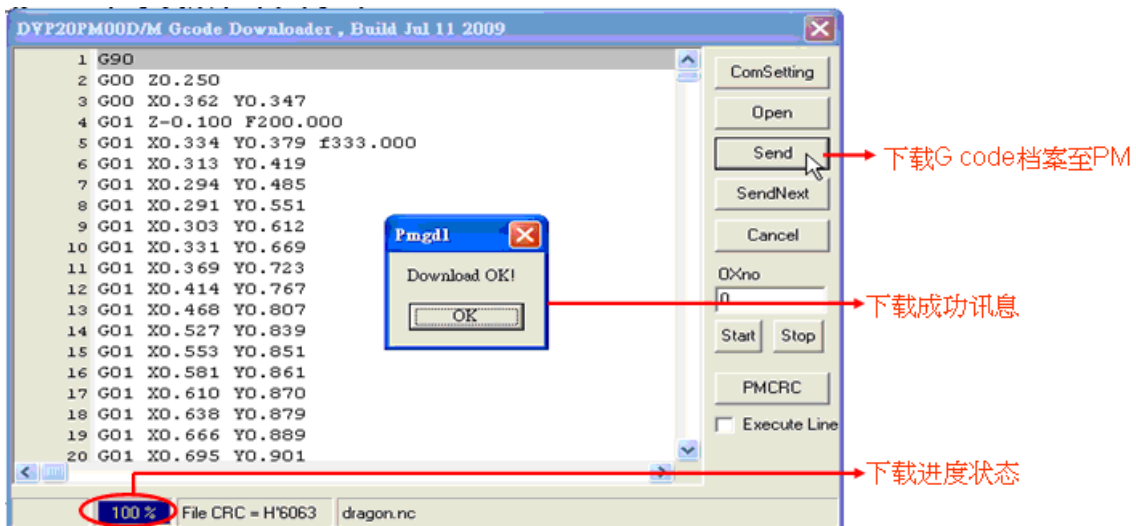


Step 4: 点选「开启」检视下载的 G code 档案，同时左下角会出现此档案的 CRC，接着再按「传送」下载 G code 档至 PM。

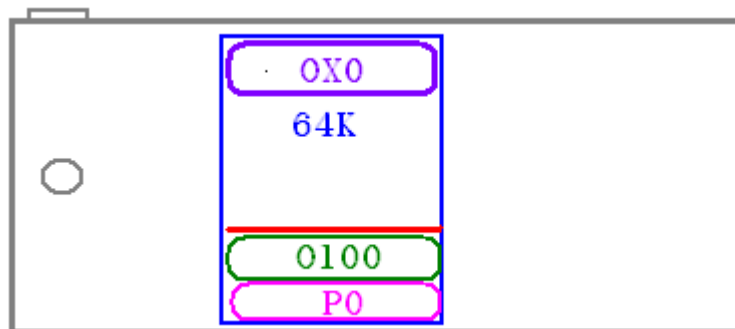
(1) 点选「开启」检视下载的 G code 档案



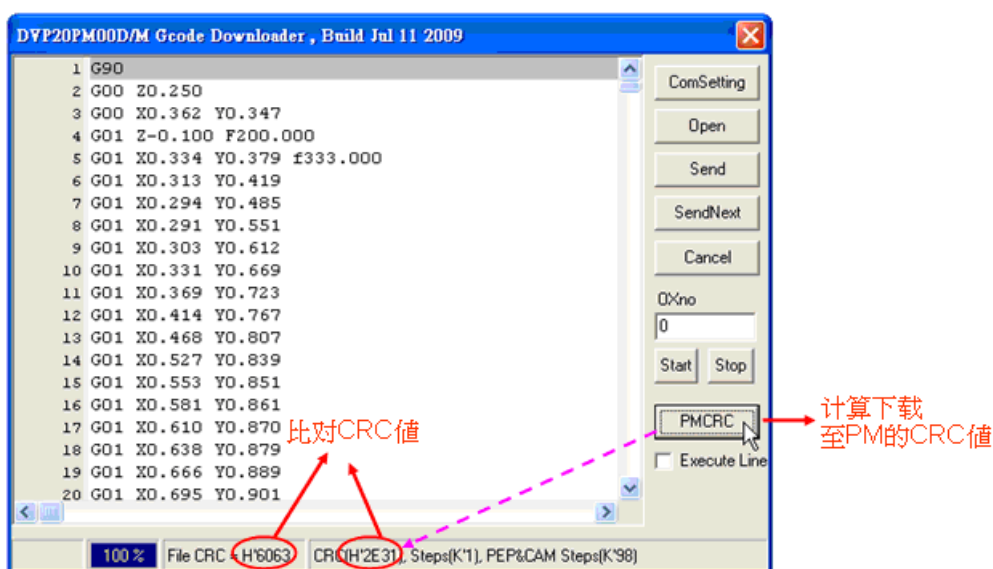
(2) 按「传送」下载 G code 档至 PM。



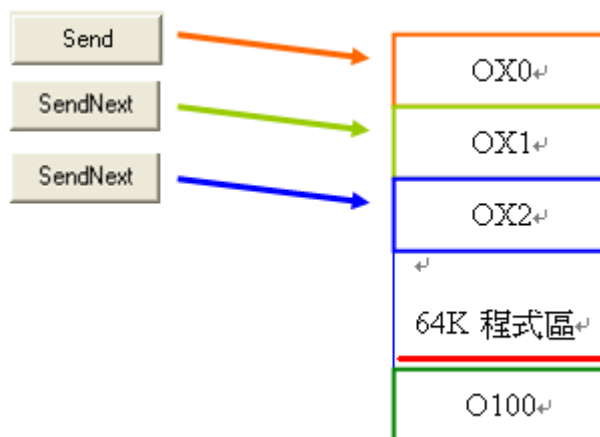
(3) 下载成功



Step 5: 可按 PM 之 CRC 计算下载至 PM 的 CRC 值, 然后比对 G code 檔 CRC 值是否相同。在 Step4 中若显示下载成功表示两者 CRC 值一样, 则 Step5 可以省略。



Step 6: PMGDL 软件可依序下载多个 G code 檔至 PM, 按「传送」下载到 OX0, 按「传送下一个」下载到 OXn, (n 会自动加 1), 以此类推, 如下图所示。假如只下载 OX0 可省略此步骤。



PS: 下次要更换 G code 文件程序时, 只需执行 Step 3 之后的步骤。

11 G 码应用

2. DNC 模式下载方式:

此种方式适用于下载 G code 文件程序超过 5000 行的程序，当 PMGDL 勾选 DNC 模式传送 G code 文件程序至 PM，PM 会自动执行 OX0 程序。

Step 1: 下载至 PM 之程序必须包含 OX0 运动程序，OX0 运动程序中不需编写 G code 程序，点选 PMSOft 系统信息区 OX，再双击 0，自动产生 OX0 编辑窗口，这样就产生 OX0 运动程序，其它步骤同一般下载模式之 Step 1。

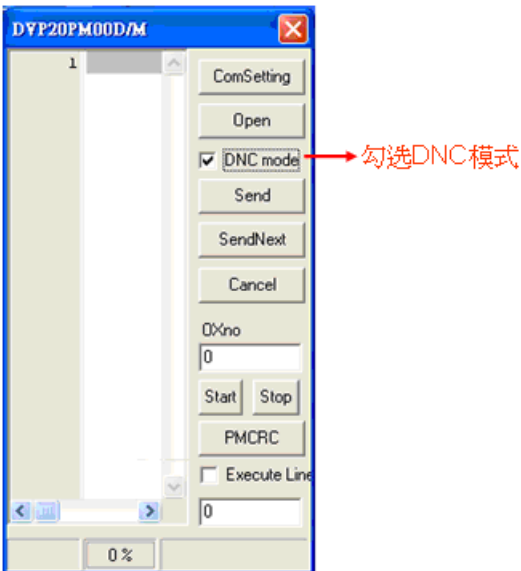
Step 2: 同一般下载模式之 Step 2。

Step 3: 同一般下载模式之 Step 3。

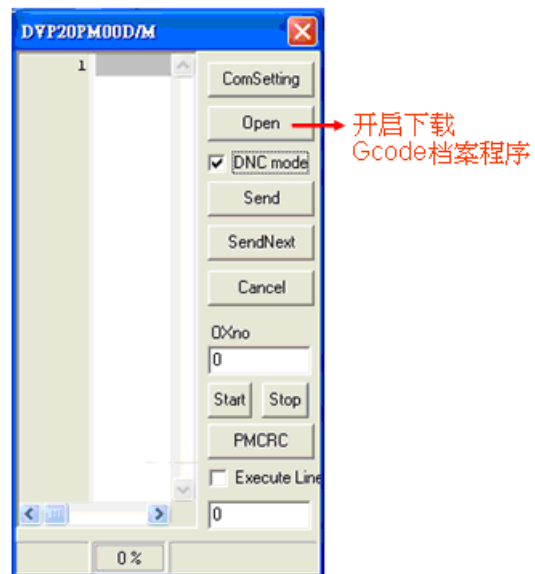
Step 4: 勾选 DNC 模式(已勾选就省略此步骤)，点选「开启」检视下载的 G code 档案，

按「传送」下载 G code 檔至 PM。，下载顺利自动启动 OX0 程序。

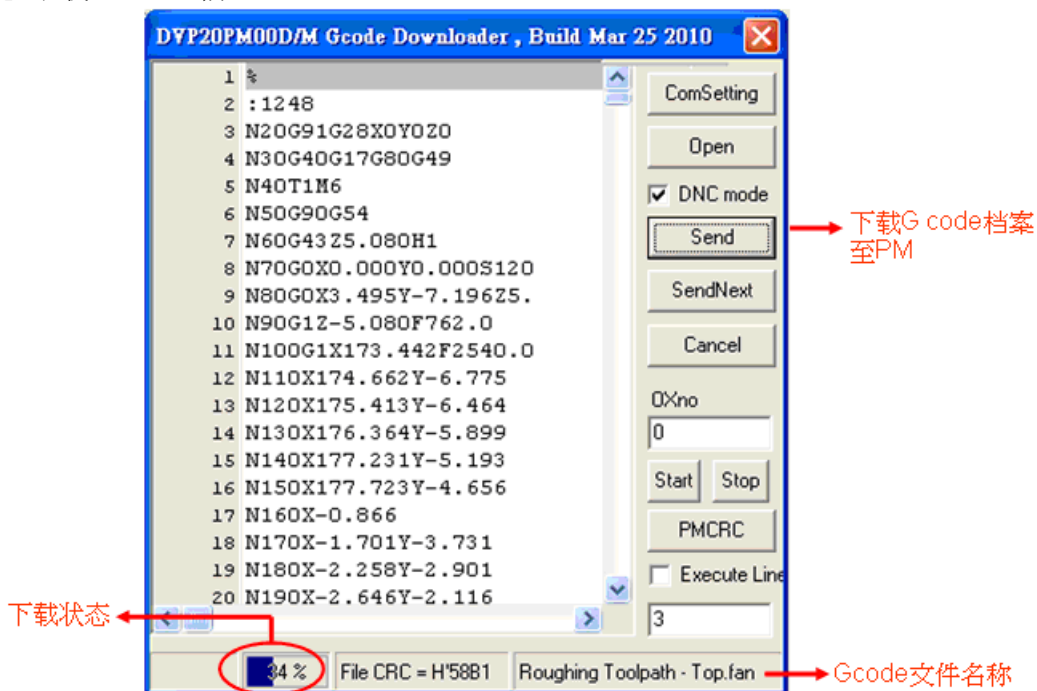
(1) 勾选 DNC 模式



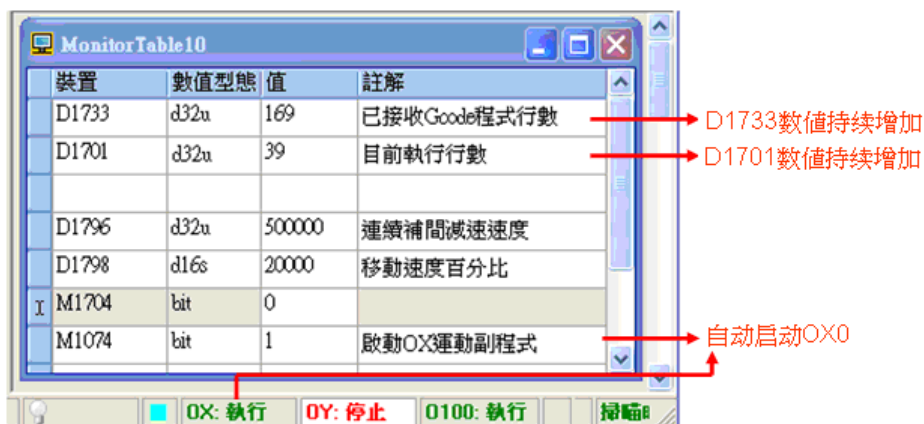
(2) 点选「开启」检视下载的 G code 档案



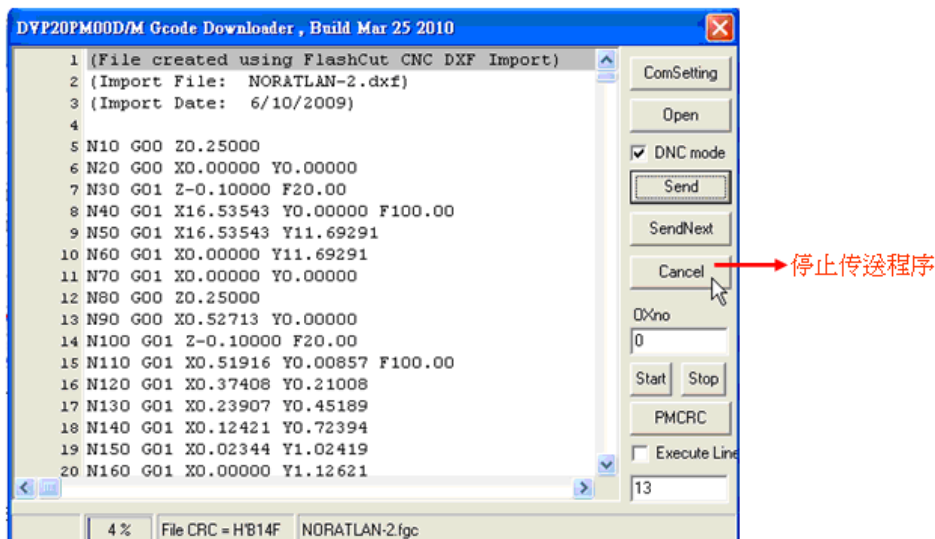
(3) 按「传送」下载 G code 檔至 PM。



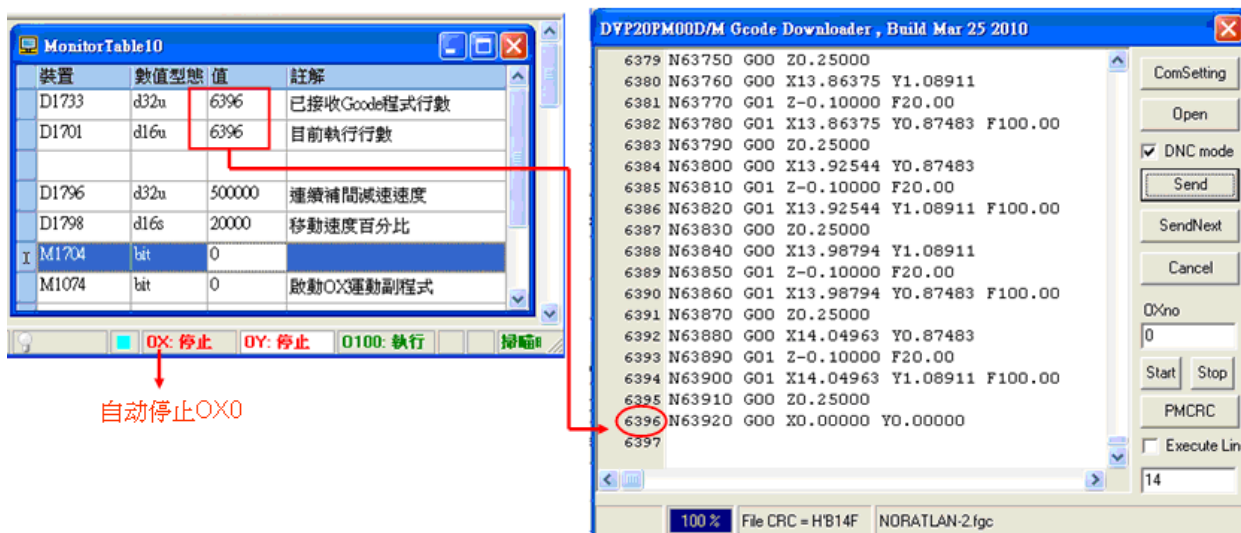
Step 5: 可使用 PMSoft 监控观察 D1734, D1733(目前 PM 已接收 G code 档行数)及 D1702..D1701(目前已执行行数)是否持续跳动增加, 且自动启动 OX0 程序, 如下图所示。



Step 6: 程序运作中若没有要停止, 则忽略此步骤。若程序要停止不要继续运作, 只要点选 PMGDL 软件中的「取消」按钮(如下图所示), 而程序执行的行数等于下载行数时, 就停止运作 G code 程序。若要重新下载运作程序, 需先停止 O100 程序后, 再启动 O100 程序。



Step 7: 程序运作(加工)结束, 可在监控表看到执行的行数数值与下载行数数值相同, 且下载的行数与 PMGDL 软件中显示的行数也相同, 程序执行结束, OX0 程序自动停止, 如下图所示。

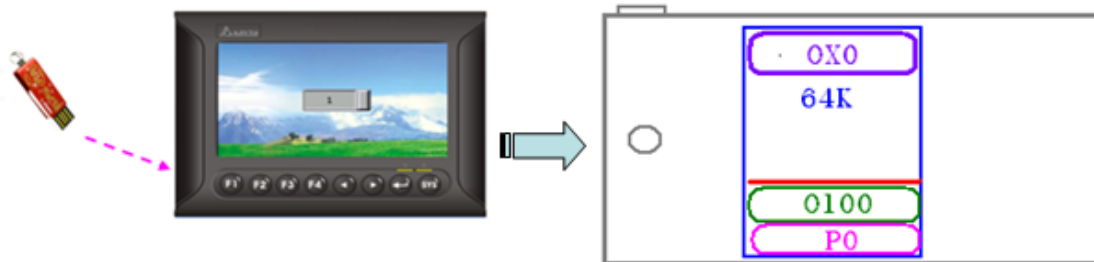


11 G 码应用

Step 8: 若要执行其它程序，可再从 Step 4 操作。

■ B Type HMI 下载

下图所示为使用 HMI 下载 G code 文件至 PM 程序区 OX0，其中 PM00M 中的 O100 程序(含其它子程序)需预先设定以 PEP 保护的方式下载至 PM，然后使 O100 运作设定 PM 的 COM1(RS232 模式)或 COM2(RS485 模式)与 HMI 通讯，再接收 HMI 传送的 G code 文件程序。



Step 1: 设定安装一台可下载 G code 文件(NC file)功能的 HMI:

(1)HMI 机种: **DOP-BS0722P00**

(2)安装 HMI 程序编辑软件 Screen Editor 2.00.12

(3)执行 Patch_DOPB_072909 软件更新 Screen Editor 2.00.12(已更新过就不用再执行)

(4)编辑 HMI 程序:

(a)开启 Screen Editor 2.00.12, 开新档案

(b)「控制命令」设定: 进入"选项">>"设定模块参数">>"控制命令", 设定参数如下图:



(c)「通讯」设定: 进入"选项">>"设定模块参数">>"通讯"设定如下图:

控制器: Delta PM RTU
通讯端口: COM1/COM2/COM3
通讯接口: RS232/RS485
数据位: 8(RTU 模式)
停止位: 1
速率: 9600/19200/38400/57600/115200
同位: None

(5)进入 G code 档下载画面设定：在 HMI 内部存储器地址\$0 写入值 65535

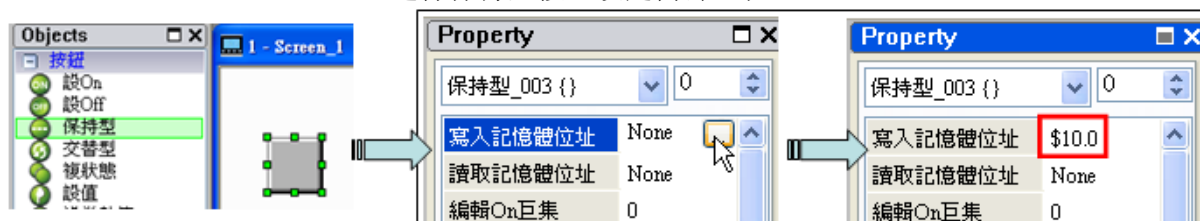
自动进入下载画面，如下所示(a)(b)两种方式：

(a)跳出下载 G code 画面的控制画面时，使用一个数值输入组件，设定写入地址\$0，可输入数值范围为 65535：



(b)或是使用一个按钮组件，设定宏指令方式，当按此按钮会对地址\$0 写入 65535。下图说明使用按钮控制宏写入 65535 数值至地址\$0。

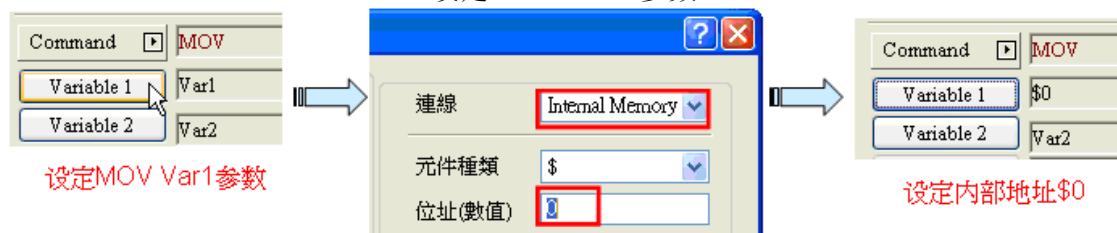
选择保持型按钮设定内部记忆地址



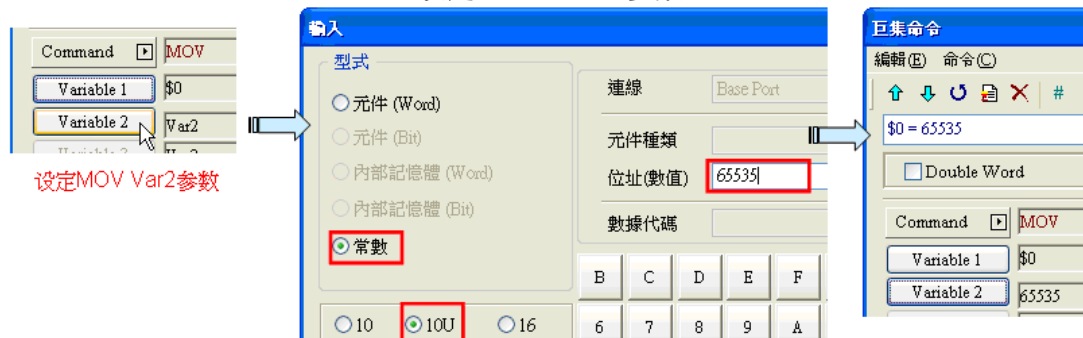
编辑宏使用 MOV 指令

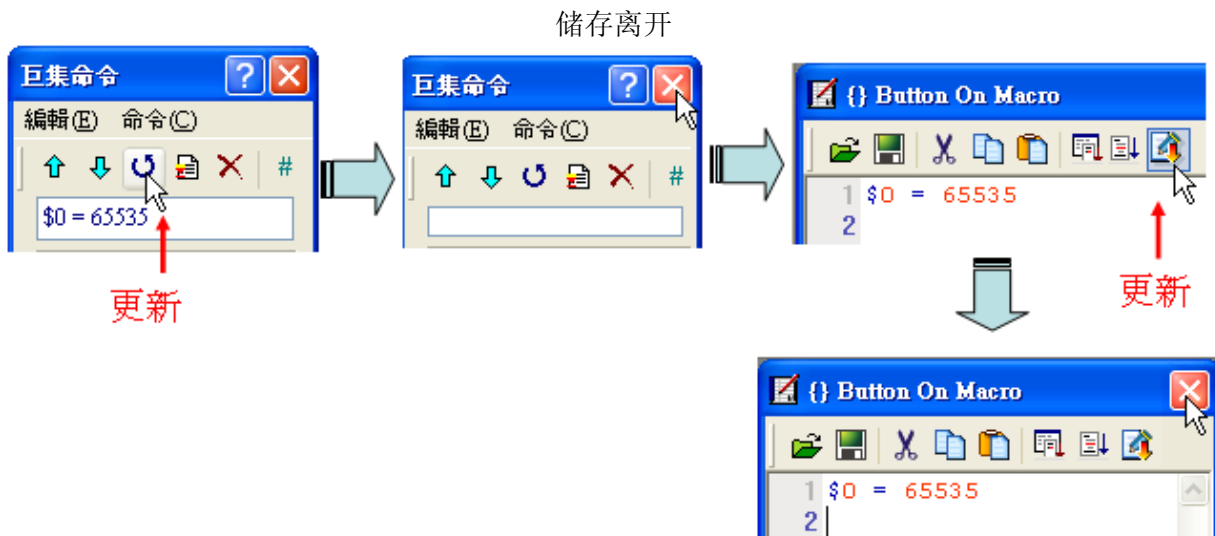


设定 MOV Var1 参数



设定 MOV Var2 参数

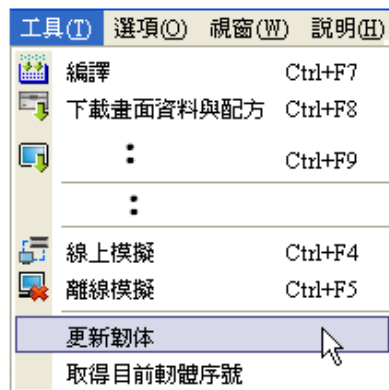




(6)HMI 程序编辑完成下载至人机

(7)更新人机韧体(已更新过即省略此步骤):

(a)选择工具 >>”更新韧体”，进行韧体更新，设定如下图:



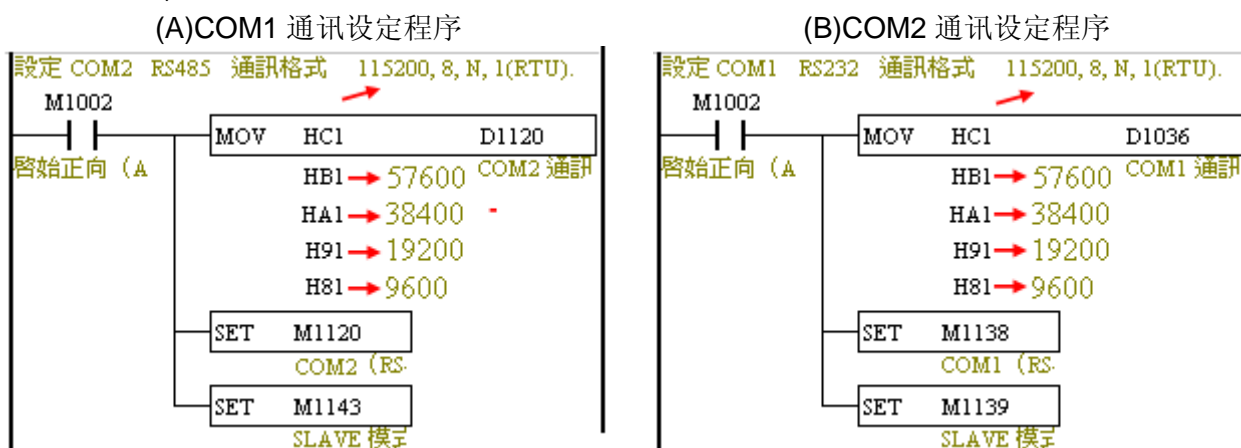
(b)选择”工具”>>”取得目前韧体序号”，确认人机韧体版本更改为 **2.0122**



Step 2: 编辑并复制所需要 G code 檔至 USB 随身碟(随身碟格式需为 FAT32)根目录下, 插入 HMI 后方左下角 HOST USB 埠。



Step 3: 编写 O100 主程序, 程序必须含 M code 处理及设定一个通讯端口(COM1/COM2)格式为 RTU,115200(57600/38400/19200/9600), 8, N, 1, 其通讯格式与人机要搭配, 下图所示为通讯端口 (COM1/COM2)程序设定。

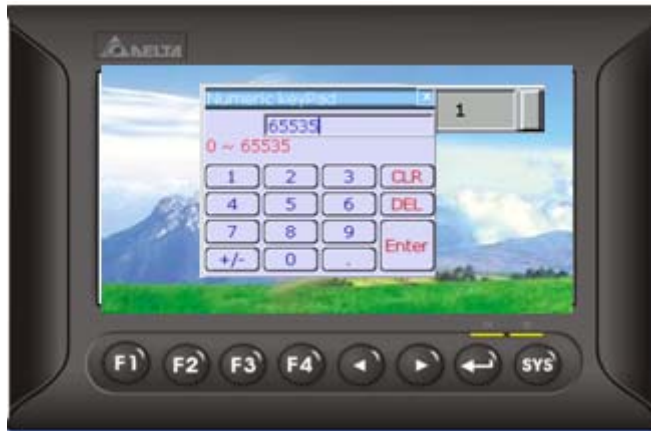


Step 4: 使用 PMSOft 将 O100 程序或将不变更的子程序设定 PEP 保护致能后, 下载至 PM, 操作方式同 PMGDL 下载之 Step 2。

11 G 码应用

Step 5: 以通讯线连接 20PM 与 B type HMI。

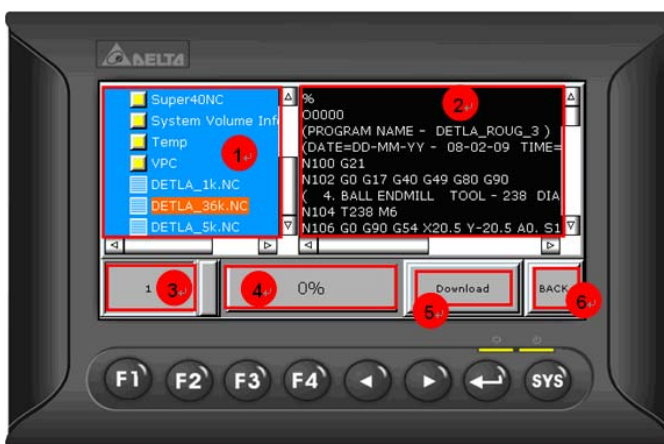
(a) 人机画面上数值输入组件输入 65535



(b) 手指点选 ENTER 进入下载 G code 档画面。



Step 6: 进入 G-Code 下载页面，各部份功能如下图所示。在(1)区选择下载 G code 档，按 Download 进行 G code 档下载，下载完成，按 BACK 键回上一页，若下载程序过大会显示 CRC 值错误。



1	档案选取窗口： 可在此窗口内选择欲下载的 G code 档案
2	G code 预览窗口： 可预览选取的 G code 档案内容
3	站号设定： 设定 20PM 站号
4	下载进度： 显示 G code 下载进度
5	Download： 按下此按钮执行下载动作
6	Back： 按下此按钮回到上一页

Step 7: 要换 G code 文件程序，可再操作 Step 5 与 Step 6 的步骤。

■ 其它型号 HMI 下载

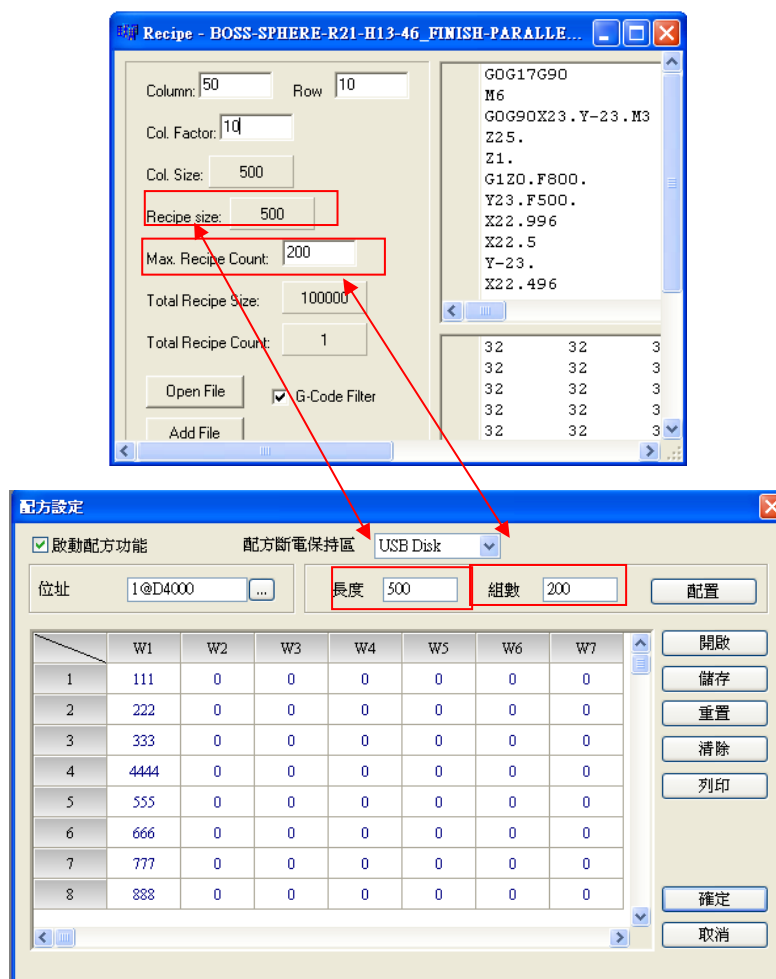
当使用其它型号的 HMI 与 PM 联机时，仅需设定好 HMI 的传送配方与 PM 的接收 Gcode 程序后，便可直接透过 HMI 将 GCODE 传送至 PM。

PM 程序设定下载 GCODE 的程序流程为先初始化下载动作与参考人机端的配方来设定配方。接着开始下载，依照配方设定将人机中的 GCODE 接收到指定的缓存器当中，需注意的是，配方为设定传送一次 GCODE 的数量，但在实际应用中，一个 Gcode 档案则需执行多次传送，故将档案分成组别，一次传送一组，每次传送完毕后便进行转码，如果转码结果正确后会在指定的缓存器显示过滤后的 ASCII 码，若转码结果错误，则会在指定的缓存器会显示 0xffff 与错误的列数。

接着 PM 在控制缓存器写入值来变更组别，切换至下一组传送，反复执行下载动作直到当 D3002 与 D3003 的值为 20H，表示 GCODE 档案全部传送完毕，最后执行转码完毕后检查转码结果。在进行传送之前，注意需先建立一个空白的运动子程序以储存接收的 GCODE，以及在主程序 O100 需使用 PEP 保护避免在上载程序时覆盖掉原本的 O100 程序

1. HMI 端配方设定：

Step1: 首先在 HMI 端设定传送时的配方，所谓配方即为在传送 GCODE 档案时的格式，包括一次传送的总行数与每行包含的字数等设定。下图为 Recipe 软件与 HMI 端的配方设定，在 Recipe 软件中，设定每行 50 个字转为一行 Gcode，总共转为 10 行，故在 HMI 端设定与 Recipe 软件设定相同，长度总共为 500 个字，组数为 200 组。



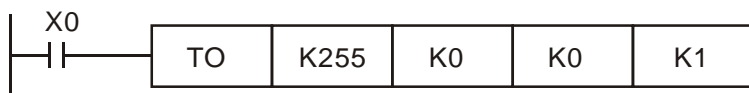
Step2: 在人机上指定配方控制缓存器与配方组别设定缓存器，这两个缓存器设定会对应到 PM 的缓存器，在 PM 中藉由写入 D65 对人机发送控制命令，例如执行配方读取、写入等命令，写入 D66 即可变更组别。



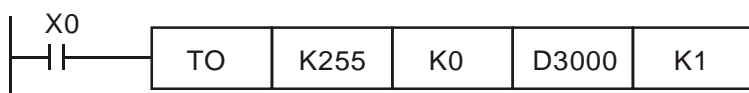
2. PM 程序设定:

PM 使用 DTO/FROM 指令执行下载 Gcode，其使用的指令如下所示：

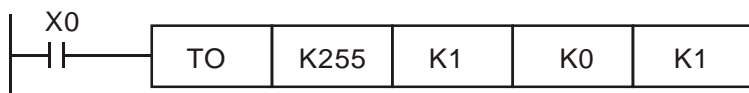
■ 初始化下载动作



■ 进行转码

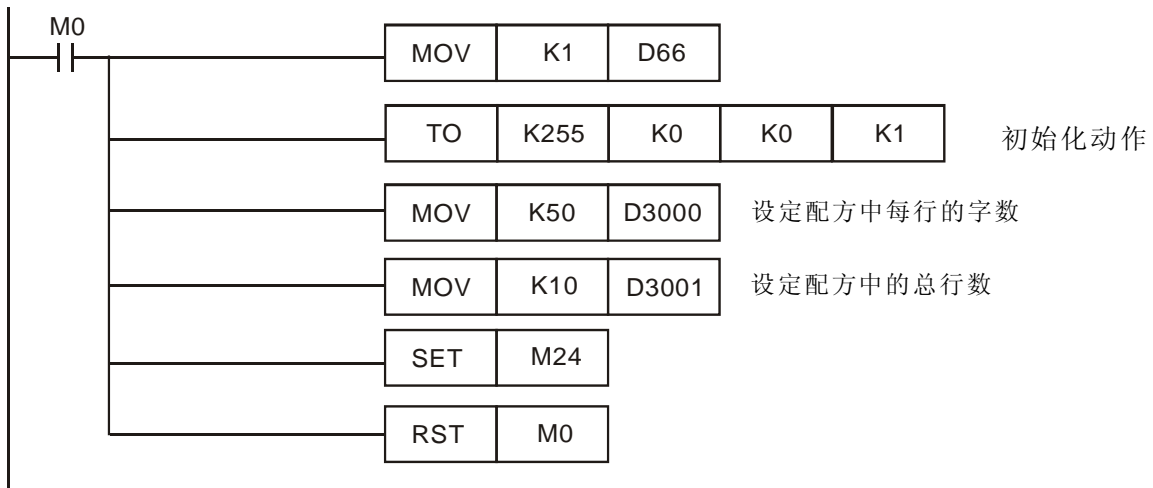


■ 转码完毕后进行检查

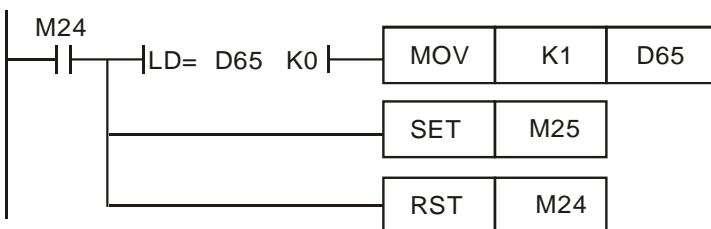


以下程序为 PM 接收 GCODE 程序，在配方设定上一行 50 个字，总共传送 10 行，接着进行转码，转码完毕后，若转码结果正确，则结果储存至 D3002~D3501 当中，若转码结果错误，则在 D3000 会显示 0xffff，D3001 显示错误的行数。最后，在上传 PM 程序后在 OX0 程序当中看到先前接收的 GCODE。

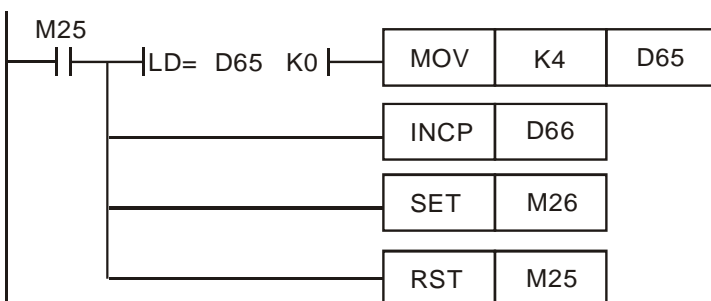
Step1: 在配方组别设定缓存器 D66 设定配方组别为 1，进行转码的初始动作以及设定配方中的行数与字数，需和人机中设定的配方相同。



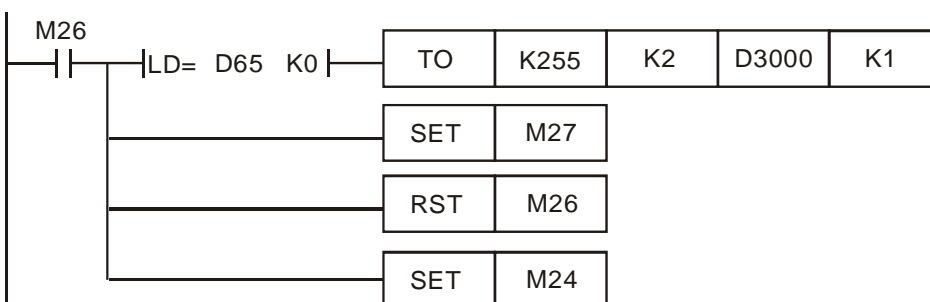
Step2: 人机读取 D65 缓存器的值作为控制配方的命令，D65=1 表示人机执行配方组别变更。



Step3: 当组别变更完成后，D65 的值会变为 0，接着下达配方写入命令，并且将组别 D66 加一，以下载下一组资料。

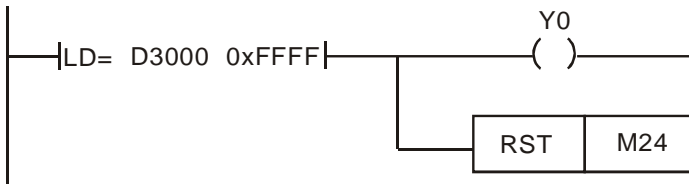


Step4: 当每次动作完成后，D65 的值会变成 0，当每组数据下载完毕后，在 PM 里执行转码命令，将接收的数据转为 ASCII 储存在 D3002~D3502 当中。以上动作执行完毕后，便重复继续下载下一组数据。

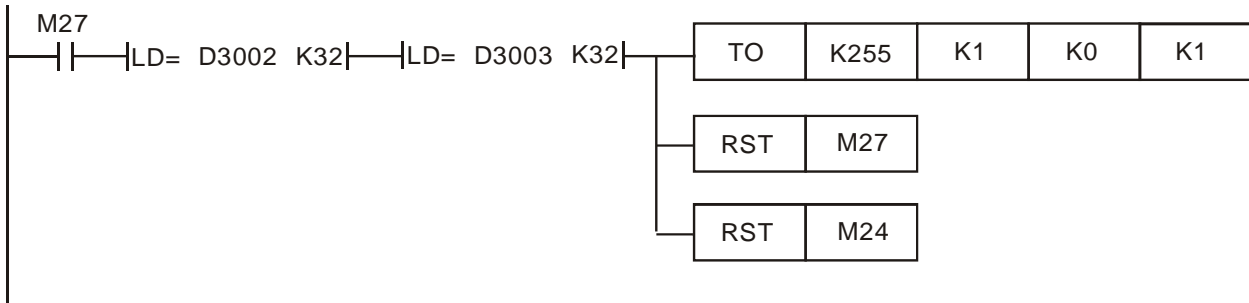


11 G 码应用

Step5: 若在转码发生错误时，D3000 会显示错误码，藉由判断来停止下一个下载动作。



Step6: 当 D3002、D3003 的值等于 K32 时表示人机中所有数据都下载完毕，发送 TO 指令 PM 进行检查。



11.2 点胶模式使用设定介绍

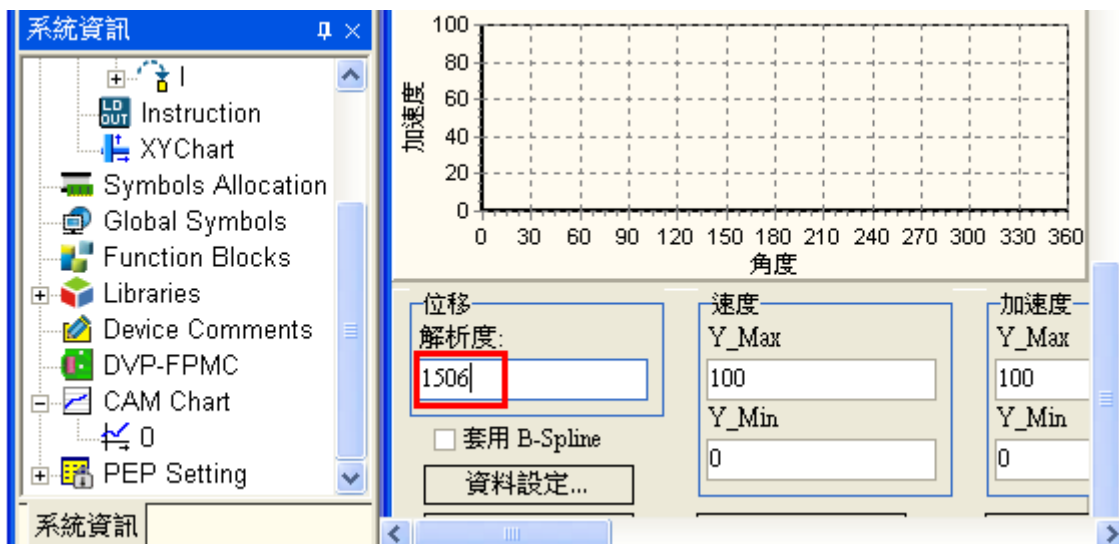
因应点胶的应用需求，DVP-PM00M 提供特殊功能，能达到点胶时等速平稳移动，而不至于停顿导致点胶不均匀。

此功能是使用 CAM Table 储存 G/M code 程序数据，然后使用电子凸轮模式，执行 CAM Table 数据。20PM 有三个 CAM Table，分别储存 OX0、OX1 及 OX2 三组 G/M code 程序，每个 CAM Table 可设定最大容量 2048 组数据，而 2048 组要平均摆放 XYZ 三轴位置及速度数据，而 G code 程序只支持 G00、G01、G04 和 M code(只支持 With mode)，因此一个 CAM Table 可储存 G00、G01、G04 和 M code 的指令数总合 680 个(扣掉开头及结束两组资料)。下面将详细说明操作步骤与 G/M code 数据储存方式。

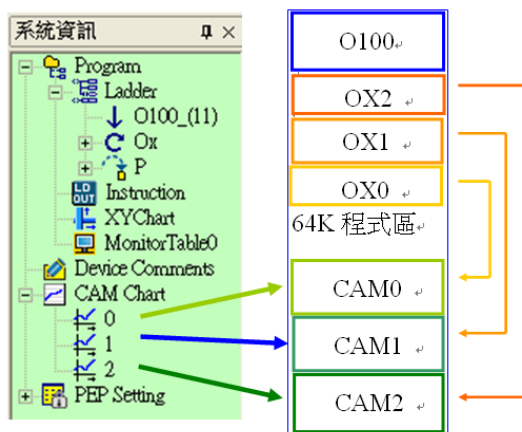
11.2.1 操作步骤

操作步骤分为四大步骤，依序为 1. 伺服位置平滑常数设定，2. 开启 CAM Table 空间，3. CAM Table 资料装填，4. 点胶模式运作，分别说明如下：

1. 伺服位置平滑常数设定：使用台达的服务器时，请设定 p1-08 参数值为 5~8，依机构调整适当值。
2. 开启 CAM Table 空间：使用者依需求设定 CAM Table 分辨率，如点胶轨迹 OX0 程序中需要 500 点(含 G04 及 M code 指令)，则 CAM Chart-0 分辨率需要设定为 1506，如下图所示，OX1 或 OX2 以此类推。



3. CAM Table 资料装填: 将 OXn(n: 0~2)程序中 G/M code 数据放入对应 CAM Table, 如下图所示:



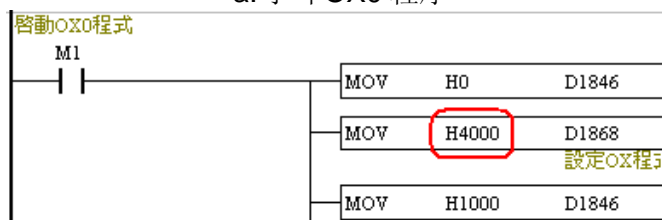
CAM Table 数据装填需要先设定三轴工作模式, 然后呼叫 OXn:

(1) 设定三轴工作模式:

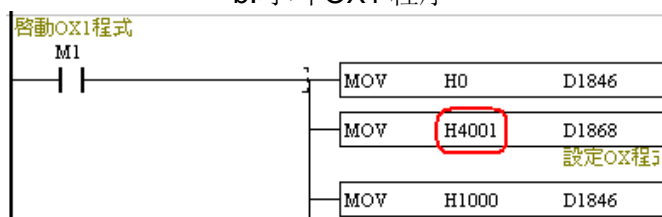
X 軸	Y 軸	Z 軸	
D1847=H1000	D1927=H1000	D2007=H1000	工作模式(bit12)

(2) 呼叫 OXn: 呼叫 OXn, 将 G/M code 指令数据存放于对应之 CAM Table, 而不运作 OXn 程序, 读取完数据结束, M1792 上缘触发。如下所示为一般呼叫 OX0, OX1 和 OX2 运动程序。

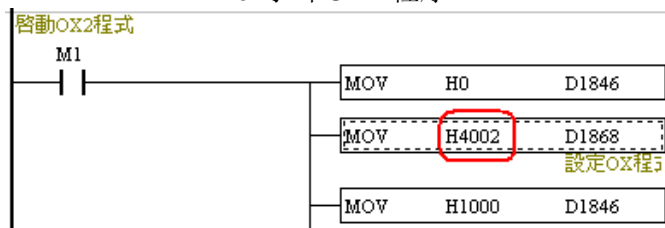
a. 呼叫 OX0 程序



b. 呼叫 OX1 程序



c. 呼叫 OX2 程序



Ps: 在这步骤之前不要设定连续路径(M1036), 否则读取 G/M code 文件的数据是为连续路径的数据

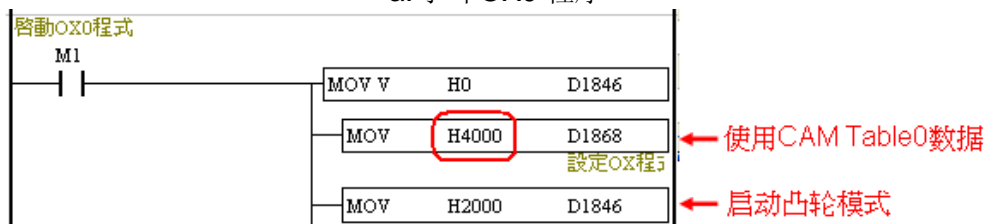
4. 点胶模式运作: 执行 CAM Table 中 G/M code 数据程序, 需设定三轴工作模式, 然后设定使用哪一个 CAM Table 数据运作, 最后设定 D1846=H2000(凸轮模式)起动, CAM Table 数据运作完毕时 M1792 上缘触发。

(1) 设定三轴工作模式: 如下三轴装置已设定为 H1000, 则可省略

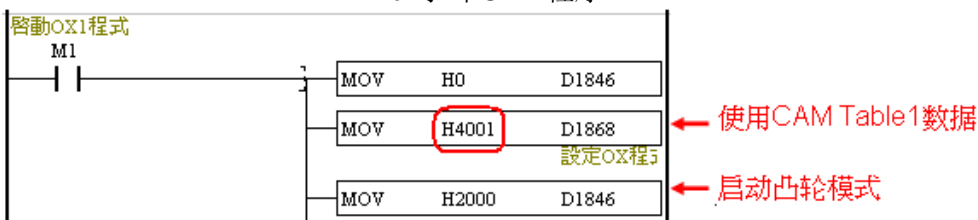
X 轴	Y 轴	Z 轴	
D1847=H1000	D1927=H1000	D2007=H1000	工作模式(bit12)

(2) 启动点胶模式运作 OXn 运动程序

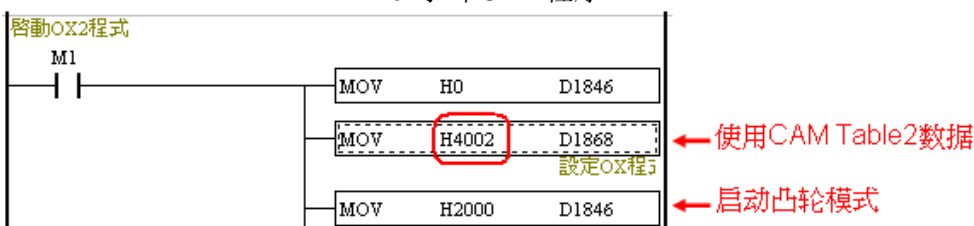
a. 呼叫 OX0 程序



b. 呼叫 OX1 程序

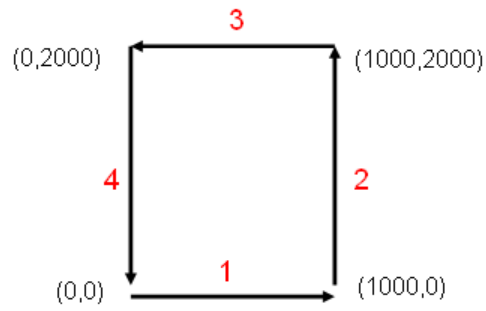


c. 呼叫 OX2 程序

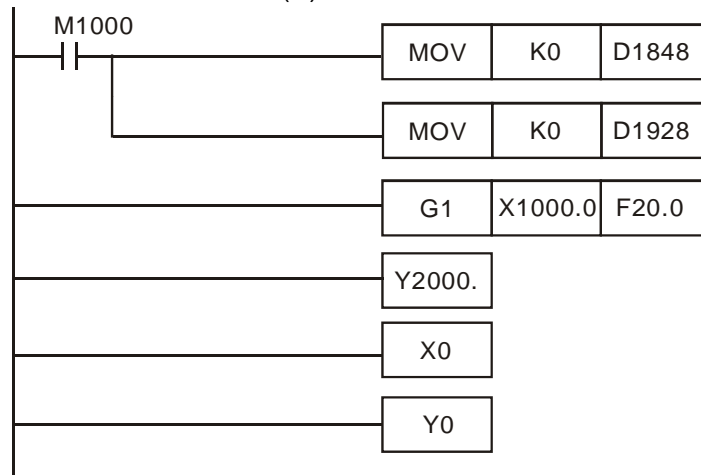


当没有执行此功能时, 工作模式要设为 0(D1847=0、D1927=0 和 D2007=0), 否则执行一段速(1st)时将发生错误。

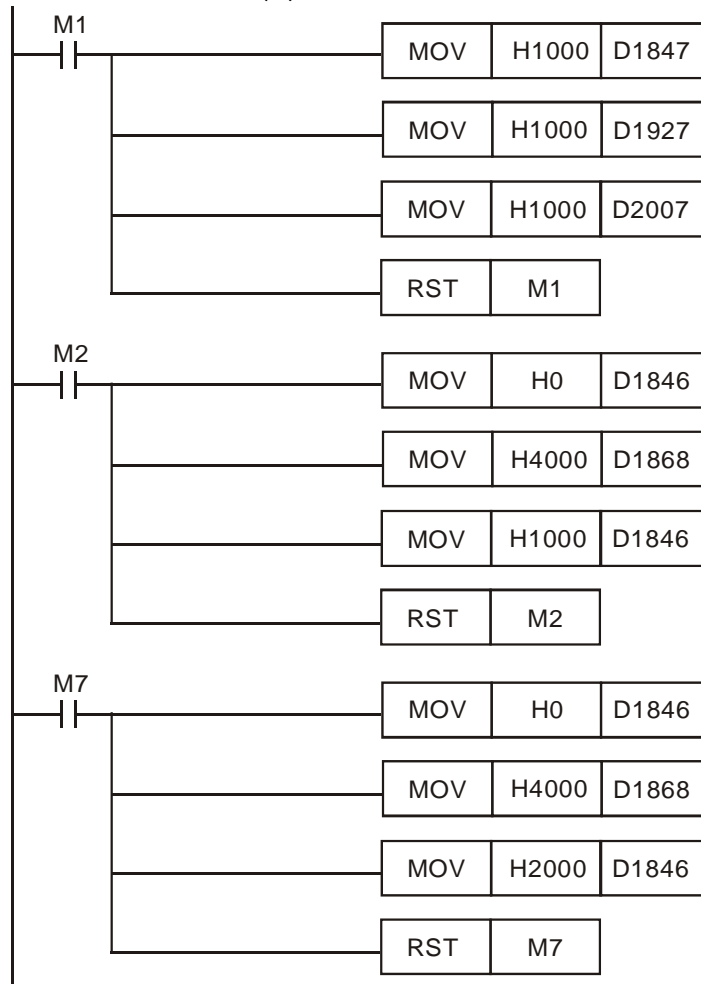
范例：执行 OX0 运动程序



(A)OX0 程序



(B)O100 主程序



11 G 码应用

过程控制如上图(B)所示:

步骤一: M1=ON 设定三轴工作模式:

- (1) D1847=H1000
- (2) D1927=H1000
- (3) D2007=H1000

步骤二: M2=ON 读取 OX0 程序数据放入 CAM Table0:

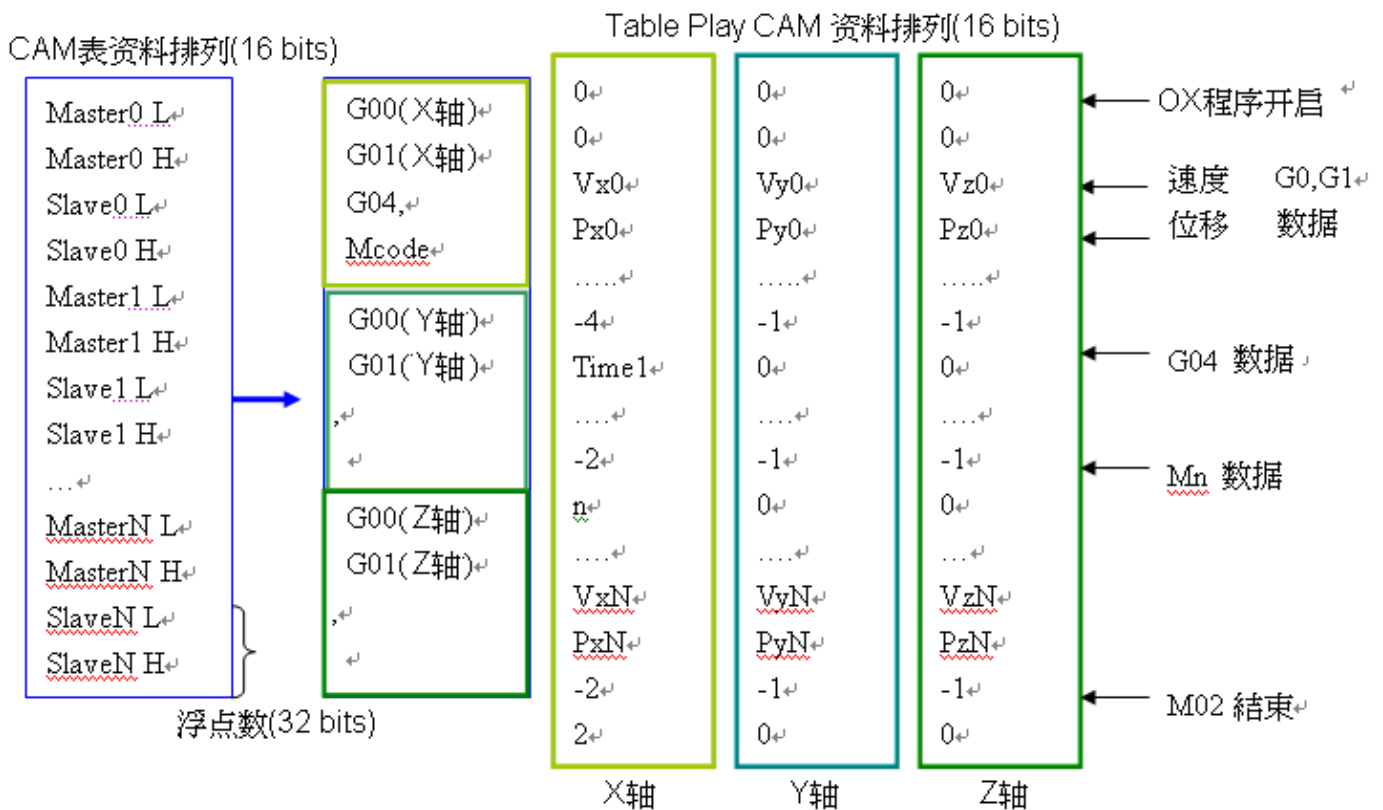
- (1) 当 D1868=H4000 时, 选择 OX0 致能设定。
- (2) 读取 OX0 数据写入 CAM Table0。

步骤三: M7=ON 运作 OX0 程序(运作数据来源在 CAM Table0):

- (1) 当 D1868=H4000 时, 选择 CAM Table0。
- (2) 电子凸轮模式。

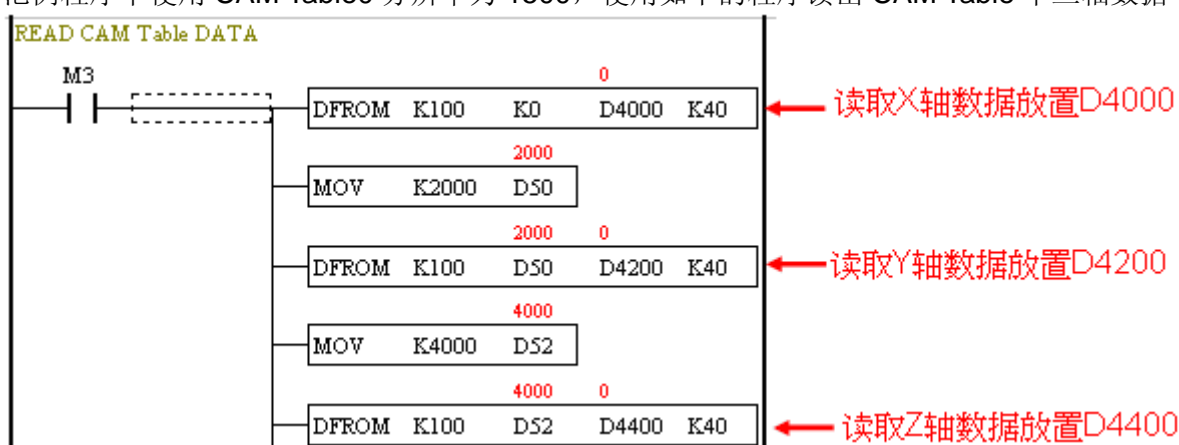
11.2.2 G/M code 文件数据储存方式:

将 OXn 程序数据放入对应 CAM Table 中, 只能摆放 G00, G01, G04 和 M code(With mode)指令, 下图所示为 CAM Table 摆放指令数据的方式。





范例程序中使用 CAM Table0 分辨率为 1500，使用如下的程序读出 CAM Table 中三轴数据



Ps: 指令 DFROM (DTO) K100 K0 D4000 K40 中，设定值 K100 表示 CAM Table0 的资料区，若设定 K101 表示 CAM Table1 的资料区，若设定 K102 表示 CAM Table2 的资料区。

装置	數...	值
D4000	float	0
D4002	float	0
D4004	float	22283.70898
D4006	float	20000
D4008	float	500000
D4010	float	40000
D4012	float	-4
D4014	float	300
D4016	float	-2
D4018	float	45
D4020	float	31653.05273
D4022	float	69000
D4024	float	-2
D4026	float	2

装置	數...	值
D4200	float	0
D4202	float	0
D4204	float	33425.5625
D4206	float	30000
D4208	float	500000
D4210	float	50000
D4212	float	-1
D4214	float	0
D4216	float	-2
D4218	float	45
D4220	float	-1
D4222	float	0
D4224	float	-2
D4226	float	2

装置	數...	值	註解
D4400	float	0	开始数据
D4402	float	0	
D4404	float	44567.41797	
D4406	float	40000	
D4408	float	500000	
D4410	float	9000	
D4412	float	-1	
D4414	float	0	
D4416	float	-2	
D4418	float	45	
D4420	float	50972.33203	
D4422	float	55700	
D4424	float	-2	
D4426	float	2	結束M2

X轴资料区 Y轴资料区 Z轴资料区

Program - OX0

```

00001 COMMENT
G1 X20. Y30. Z40. F60.
G0 X40. Y50. Z9.
G04 P3000
M45
G01 X69.0 Z55.7
    
```

可由上表资料可得知如下结论：

- (1) CAM Table 开头第一笔资料都为 0
- (2) G01 指令：速度，下一组数据为位置数据，没有数据以-1 表式

11 G 码应用

- (3) G00 指令：速度都为 5000000(500K)，下一组数据为位置数据，没有数据以-1 表示
- (4) 停止移动：在对应 G01 或是 G00 指令数据中在速度地址写入-1，执行该轴指令是停止的
- (5) G04 指令：以-4 表示 G04，下一组数据为停留时间，单位 10ms
- (6) Mcode 指令：以-2 表示 Mcode，下一组资料为 Mcode 编号
- (7) 程序结束为-2 和 2 表示 M02
- (8) G04 指令数据只摆放在 X 轴数据区
- (9) Mcode 指令三轴数据区都有摆放

PMSOft1.03V 之后版本(不含 1.03 版)才含有「程序单元」(POU)功能的编辑环境,可开启旧版的程序。POU 程序编辑只适用阶梯图编辑环境,其主要两大功能:

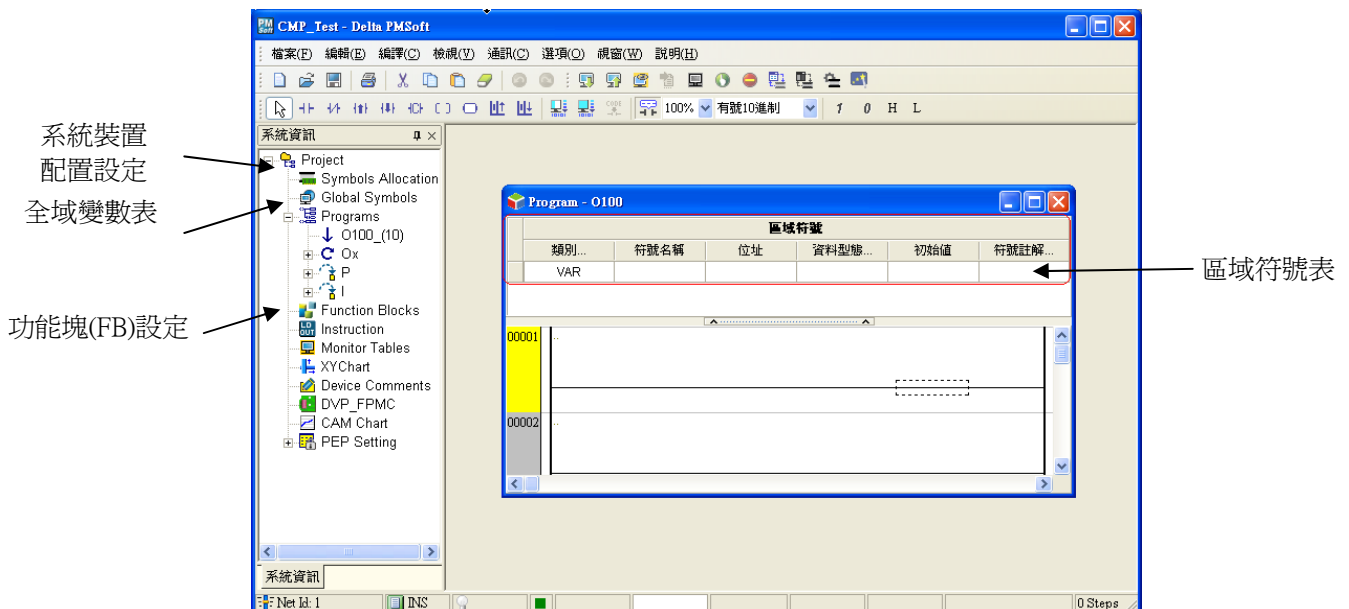
功能一: 使用者可以编辑特殊功能块,简称 FB(Function Block),FB 用于 O100 主程序、OXn 运动程序、Pn 子程序和 FB 程序中。PMSOft 可将 FB 程序汇入和汇出。

功能二: 阶梯图程序中的装置可使用符号取代。阶梯图中的符号及 FB 中的符号可由使用者配置装置或是由系统配置装置。

因此 POU 的功能可使程序编辑更模块化,简化复杂程序,减短程序开发时间。接下来分几部分对新增 POU 使用环境功能、操作方法、POU 监控方法及符号和功能块自动显示批注(HINT)做详细说明。

12.1 POU 阶梯图编辑模式环境

PMSOft 开新项目或开启旧项目,进入阶梯图模式的编辑环境,在 POU 功能方面,有符号表(区域符号表及全域符号表)、系统装置配置和功能块(FB)等组件设定,如下图所示。



12.1.1 符号表功能

使用者可透过符号表来宣告及定义符号。符号可分区域符号和全域符号,因此符号表有区域符号表和全域符号表,区域符号只用于当下的编辑窗口中,而全域符号用于整各项目 POU 程序中。

■ 区域符号表:

POU 有程序及功能块二种型式,因此区域符号表存在于程序编辑窗口(O100、Pn 子程序及 OXn 运动程序)与功能块程序编辑窗口,以下说明区域符号在这两种窗口使用及差异:

12 POU 阶梯图编辑模式

1. 程序编辑窗口:

(1)类别只支持 VAR 型态，且于现行程序内部使用。

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	TEST_1		SET_FB		
VAR	Result		BOOL	FALSE	
VAR	BIT1		BOOL	FALSE	

(2)「地址」栏皆需填入值。

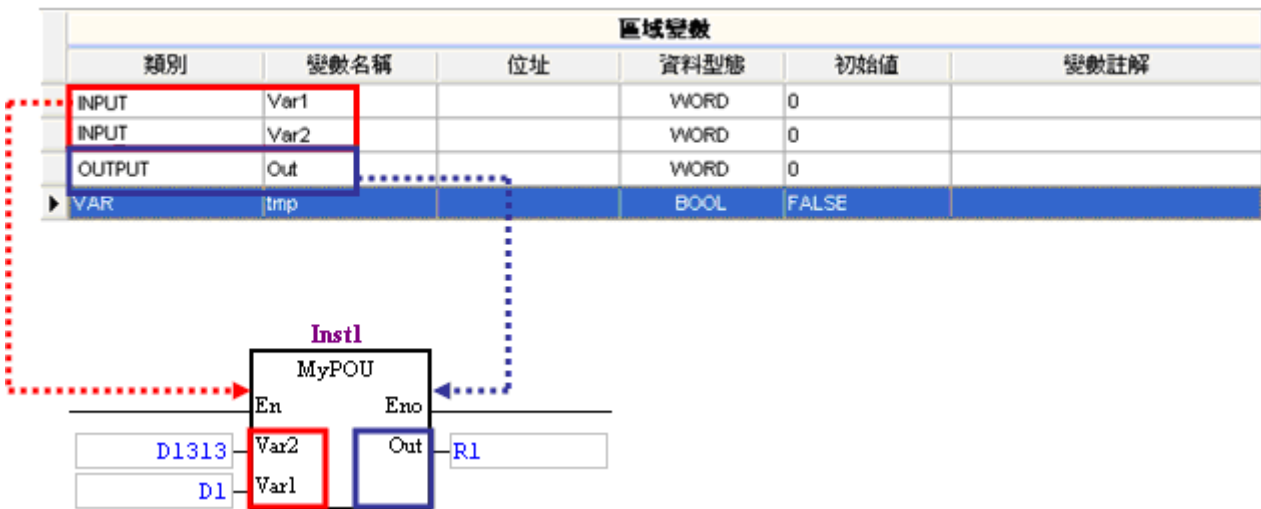
2. 功能块程序编辑窗口:

(1)支持三种类别: VAR、INPUT 和 OUTPUT。


區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	DATA		WORD	0	
INPUT	FINISH		BOOL	FALSE	
OUTPUT	Tn		TIMER	0	

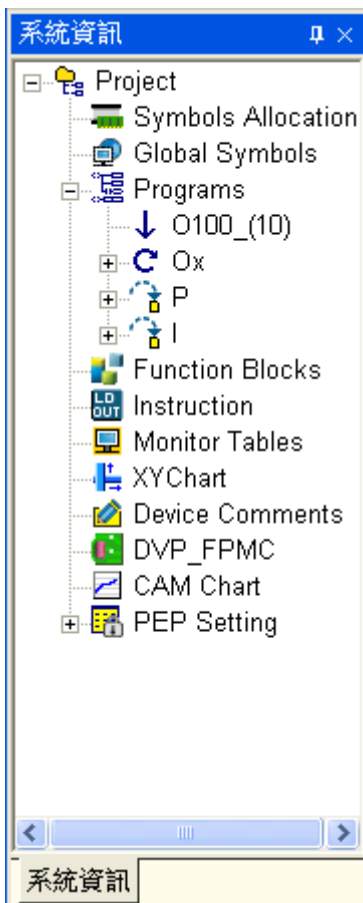
(2)「地址」字段可保留空白。

(3)当类别型态设定为 INPUT 或 OUTPUT 时，符号将成为功能块的输出或输入接口。图示说明如下:



■ 全域符号表：

全域符号表只有一个，双击系统信息区  图示(如下图(A)所示)，将弹出下图(B)全域符号表。



(A) 系统信息区

全域符号表				
符號名稱	位址	資料型態...	初始值	符號註解...

(B) 全域符号表

■ 符号命名规则：下图红框为符号表编写符号的字段。

區域符号					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	TEST_1		SET_FB		
VAR	Result		BOOL	FALSE	
VAR	BIT1		BOOL	FALSE	

符号命名规则要符合下面叙述所有条件

1. 不允许有符号如(~!@#\$\$%^&*())...等)
2. 符号名称不区分大小写
3. 符号名称可由底线符号、英文字母、数字所组成
4. 不可有连续两个以上的底线符号
5. 底线符号不可以出现在最后一个字符
6. 最大长度为 20 个字符
7. 名称中不可以含有空格符
8. 符号名称不可以为装置名称
9. 符号名称不可以为常数
10. 符号前两个字符为"DD"后面接十进制的数值，则为不合法的符号名称

12 POU 阶梯图编辑模式

■ 符号表使用规则:

1. 同一个列上的符号名称(ID), 数据型态(Type), 初始值(Initial)等必须要相互搭配
2. 同一个列上的符号名称(ID), 数据型态(Type), 初始值(Initial)不可单独存在
3. 在同一符号窗体中, 符号不可以重复宣告
4. 若符号有指定装置(Address), 不可以超过该装置的使用范围
5. G code 指令不支持符号
6. 索引符号在区域符号表要设定地址为 V0 ~ V7 或 Z0 ~ Z7 等装置地址(下图所示)
7. 索引符号指定装置是 Vx(V0 ~ V7), 符号型态(Type)要设为 WORD(下图所示)
8. 索引符号指定装置是 Zx(V0 ~ V7), 符号型态(Type)要设为 DWORD(下图所示)

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	SHIFT_V	V0	WORD	0	
VAR	SHIFT_Z	Z0	DWORD	0	

12.1.2 符号表编辑与符号宣告

符号表有区域符号表与全域符号表, 两者操作方式都一样, 差别是区域符号表有类别(class)字段(如下图所示)可选择, 接下来依序说明符号表编辑方式:



全域符号表



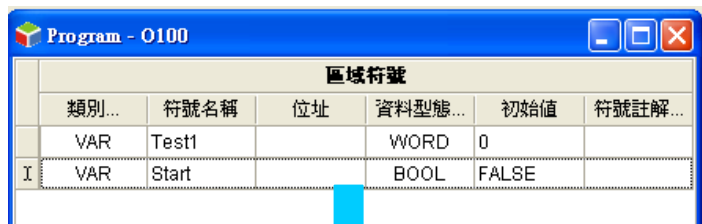
区域符号表

■ 符号表中宣告符号及新增符号表字段

1. 光标点选符号表空白列(如下图(A)所示)或是光标在符号表最后面一列, 按一下“Enter”, 产生一列空白列(如下图(B)所示), 此时可宣告设定新符号。





(A)已有空白列



(B)产生空白列


属性说明:

- a、类别: 符号型态类别。区域符号表才有类别这个字段, 功能块中的区域符号有 VAR、INPUT 和 OUTPUT 三种类别, 而其它区域符号表只有 VAR 类别。使用者可直接以键盘输入或点选定义数据型态字段(Class)而产生 , 点选  弹出可供选择的类别窗口, 点选类

别种类设定类别(如下图所示):

Step 1: 点选类别字段

區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
VAR						

Step 2: 点选 



區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
VAR						

Step 3: 点选类别

區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
VAR						

Step 4: 结果

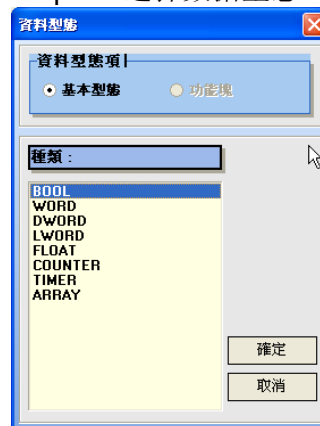
區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
INPUT						


- b、符号名称：光标点选字段，输入符号名称。
- c、地址：地址可由系统自动配置装置或使用者输入指定装置。
- d、数据类型态：符号的数据型态，其分为简易(Simple)及功能块(Function Block)。使用者可直接以键盘输入或点选数据类型态字段(Type)产生 ，点选 ，弹出可供选择的数据型态窗口(如下图所示)：


Step 1: 点选义数据类型态字段

區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
INPUT	Start					

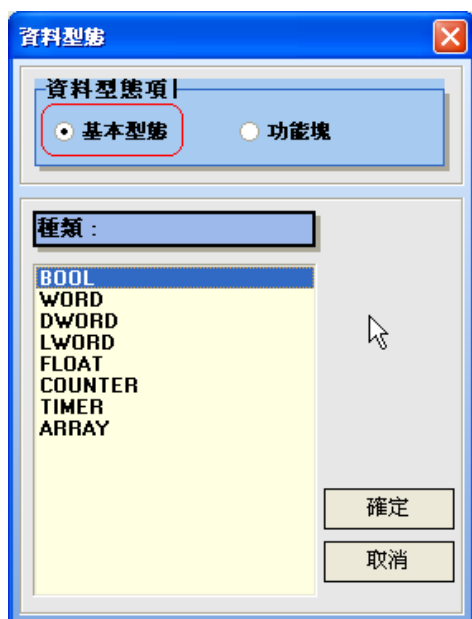
Step 3: 选择数据类型态



Step 2: 点选 

區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
INPUT	Start					

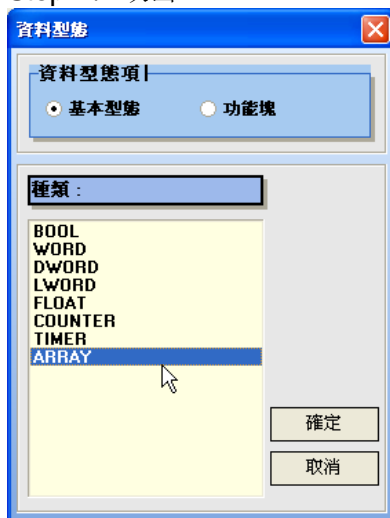
(a) 简易型态(Simple Type): 一般装置符号宣告



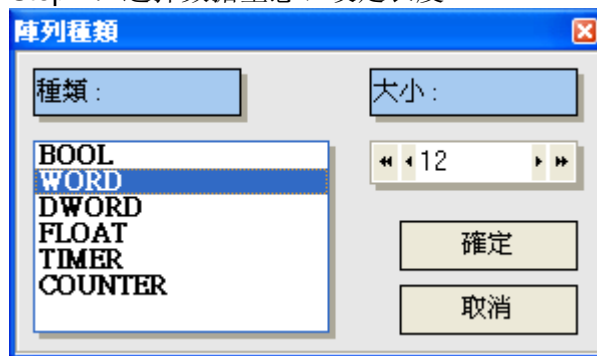
12 POU 阶梯图编辑模式

- **BOOL**: 宣告符号型态是一位(BIT), 系统配置为 M 装置。
- **WORD**: 宣告符号型态是 16 位, 系统配置 D 装置。
- **DWORD**: 宣告符号型态是 32 位, 系统配置连续 2 个 D 装置。
- **LWORD**: 宣告符号型态是 64 位, 系统配置连续 4 个 D 装置。
- **FLOAT**: 宣告符号型态是 32 位浮点数, 系统配置为 2 个 D 装置, 用于浮点数指令。
- **COUNTER**: 宣告符号型态是 16 位计数装置, 系统配置 C 计数装置。
- **TIMER**: 宣告符号型态是 16 位计时装置, 系统配置 T 装置。
- **ARRAY**: 宣告数组型态符号, 点选进入, 可选择各种型态数组及设定长度, 如下图所示, 宣告符号 **data** 为 WORD 型态长度 2 的数组符号。

Step 1: 双击 ARRAY



Step 2: 选择数据型态、设定长度



Step 3: 结果

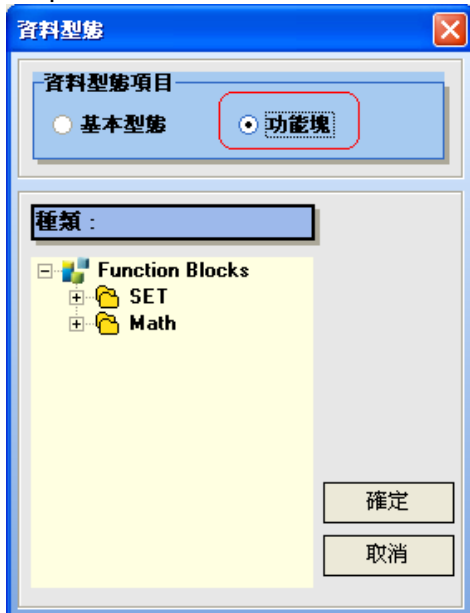
區域符號						
	類別...	符號名稱	位址	資料型態...	初始值	符號註解...
	VAR	Test1		WORD	0	
	VAR	Start		BOOL	FALSE	
▶	VAR			WORD[12]		

- (b) 功能块型态(Function Block): 宣告符号型态为功能块, 下图所示种类里头的型态是功能块名称。

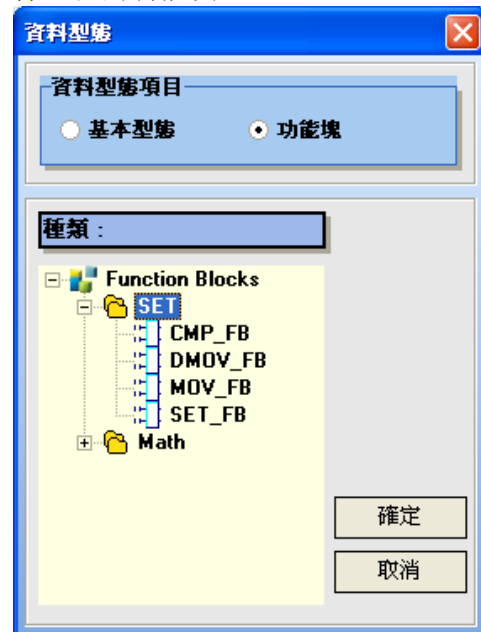


使用功能块一定要将功能块作符号宣告，下列三个步骤为功能块符号宣告：

Step 1: 选择符号为功能块型态



Step 2: 若功能块含有目录，则点选目录展开，选择宣告的功能块



Step 3: 结果

區域符號						
	類別...	符號名稱	位址	資料型態...	初始值	符號註解...
	VAR	Test1		WORD	0	
	VAR	Start		BOOL	FALSE	
	VAR	Data		WORD[12]		
I	VAR	ADD		ADD_FB		

e、 初始值: 符号的默认值。初始值在下载时可写入 PLC。

f、 批注: 符号的相关信息或说明

■ 符号表初始值设定功能

在输入完数据型态后，点选在符号窗体中初始值的字段，即可进行初始值的输入，若此符号不需进行初始化，则需将初始值字段清空。

12 POU 阶梯图编辑模式

1. 当数据类型为 **BOOL**，则初始值需设定为 **TRUE** 或 **FALSE**。

區域符號						
	類別...	符號名稱	位址	資料型態...	初始值	符號註解...
I	VAR	Bit		BOOL	FALSE	

2. 当数据类型为 **WORD**, **DWORD** 时，其初始值可以加入 'K' 或 'H' 的前缀修饰，分别表示数值为 10 进制或 16 进制 (例如: K100, H200)。

區域符號						
	類別...	符號名稱	位址	資料型態...	初始值	符號註解...
	VAR	Word1	D7000	WORD	100	
	VAR	Word2	D7001	WORD	K100	
	VAR	Word3	D7002	WORD	H200	

3. 当数据类型为数组型态时，点选初始值字段，会出现数组初始值设定窗口，对数组中的位置进行数值的输入，数值输入完成后按下「确定」，就会自动产生数组的初始值。

陣列初始值設定 ✖

陣列資料型態: WORD[4] 確定

取消

位置	初始值
[0]	1
[1]	1
[2]	0
[3]	0

执行后

區域符號					
	類別...	符號名稱	位址	資料型態...	初始值
	VAR	Test1		WORD	0
	VAR	Start		BOOL	FALSE
▶	VAR	Data		WORD[4]	[1(2),0(2)] ...
	VAR	ADD		ADD_FB	

4. 数组初始值的格式如下：

- 数值需放置在左右中括号之间
- 数组中各个位置的数值需由逗号作为区隔。(例如: [1,2,3,4])
- 如有连续重复的数值可以用「数值+(重复次数)」的方式表示。(例如: 1(2)表示有连续两个数值为 1)

區域符號					
	類別...	符號名稱	位址	資料型態...	初始值
▶	VAR	WordArray	D7000	WORD[4]	[1(2),0(2)]

■ 在符号表中修改符号

修改符号表符号跟设定新符号方式一样，直接在符号列表字段设定修改符号数据或数据型态，如下图所示修改符号类别。

區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
I	VAR					
	VAR					
	INPUT					
	OUTPUT					

注 1: 功能块型态的 POU，不允许宣告数据型态为功能块的符号

注 2: 功能块型态的 POU，若符号为 INPUT 或 OUTPUT，不能指定地址，由系统自行配置。

■ 移动符号位置

1. 选择一笔要移动的符号

區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
I	VAR		BOOL	FALSE		
	VAR		WORD[4]	[1(2),0(2)]		
	VAR		ADD_FB			

2. 按组合键 (Alt + ↓) 或 (Alt + ↑)，将符号向上移动或向下移动

區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
	VAR		WORD[4]	[1(2),0(2)]		
	VAR		BOOL	FALSE		
	VAR		ADD_FB			


■ 符号表删除

◆ 方法一: 点选「编辑(E)」菜单中的「删除(D)」。

◆ 方法二: 点选功能图标列上的 。

◆ 方法三: 点选鼠标右键菜单中「删除(D)」命令。

◆ 方法四: 按下键盘上的 (Delete) 键。网络区段:

单列删除: 光标点选要删除之行列，该列最左边显示  图形，使用上面所示四种方法之一，删除指定的符号列(如下图所示删除 GA2 符号)。

12 POU 阶梯图编辑模式

Step 1: 點選預刪除符號列


全域符號				
符號名稱	位址	資料型態...	初始值	符號註解...
GA1		WORD[12]		
GA2		WORD		
GA3		CMP_FB		
GA4		COUNTER[4]		

Step 2: 刪除(選擇四種方式之 1)

全域符號				
符號名稱	位址	資料型態...	初始值	符號註解...
GA1		WORD[12]		
GA2		WORD		
GA3		CMP_FB		
GA4		COUNTER[4]		

Step 3: 結果

全域符號				
符號名稱	位址	資料型態...	初始值	符號註解...
GA1		WORD[12]		
GA2		WORD		
GA4		COUNTER[4]		

多列刪除: 光標點選要刪除起始列, 該列最左邊顯示  圖形, 按組合鍵 (Shift+ 光標點選刪除終止列), 使用上面所示四種方法之一, 刪除指定連續多列符號(如下圖所示刪除 GA2 至 GA5 符號)。

Step 1: 點選預刪除起始列

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	GA1		WORD[2]		
VAR	GA2		BOOL		
VAR	GA3		WORD		
VAR	GA4		BOOL		
VAR	GA5		TIMER		
VAR	GA6		COUNTER		
VAR	GA7		WORD		
VAR	GA8		FLOAT[5]		

Step 2: Shift+ 游標點選刪除終止列

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	GA1		WORD[2]		
VAR	GA2		BOOL		
VAR	GA3		WORD		
VAR	GA4		BOOL		
VAR	GA5		TIMER		
VAR	GA6		COUNTER		
VAR	GA7		WORD		
VAR	GA8		FLOAT[5]		

Step 3: 刪除(選擇四種方式之 1)

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	GA1		WORD[2]		
VAR	GA2		BOOL		
VAR	GA3		WORD		
VAR	GA4		BOOL		
VAR	GA5		TIMER		
VAR	GA6		COUNTER		
VAR	GA7		WORD		
VAR	GA8		FLOAT[5]		

Step 4: 結果

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	GA1		WORD[2]		
VAR	GA2		BOOL		
VAR	GA7		WORD		
VAR	GA8		FLOAT[5]		

■ 符號匯出

符號匯出是把符號表中所有符號都匯出儲存成一個格式為*.csv 的檔案。符號匯出的方法說明如下。

1. 鼠標光標移到符號表中, 按鼠標右鍵, 點選匯出符號表(Export Symbols)

Step 1: 鼠标光标移至符号表中

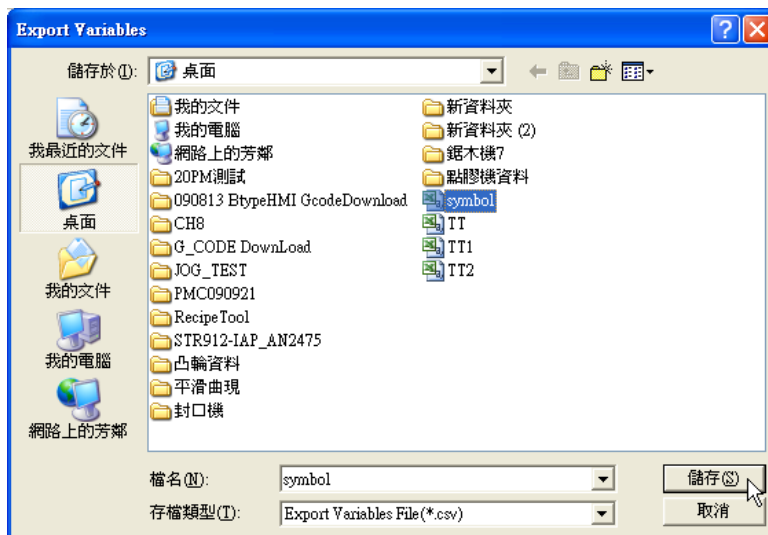
區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註...
VAR	tt	D7014	WORD	0	
VAR	add_16		add_fb		
VAR	ee	D7017	WORD	0	

Step 2: 按鼠标右键，点选汇出符号表 (Export Symbols)

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註...
VAR	tt	D7014	WORD	0	
VAR	add_16		add_fb		
VAR	ee	D7017	WORD	0	

00001
LD= TT EI
00002

2. 设定符号文件名称及路径



■ 符号汇入

符号汇入是选择符号文件，将符号摆放至符号表中。符号汇入的方法说明如下。

1. 鼠标光标移到符号表中，按鼠标右键，点选汇入符号表 (Import Symbols)

Step 1: 鼠标光标移至符号表中

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	ADD_P7		add_fb		

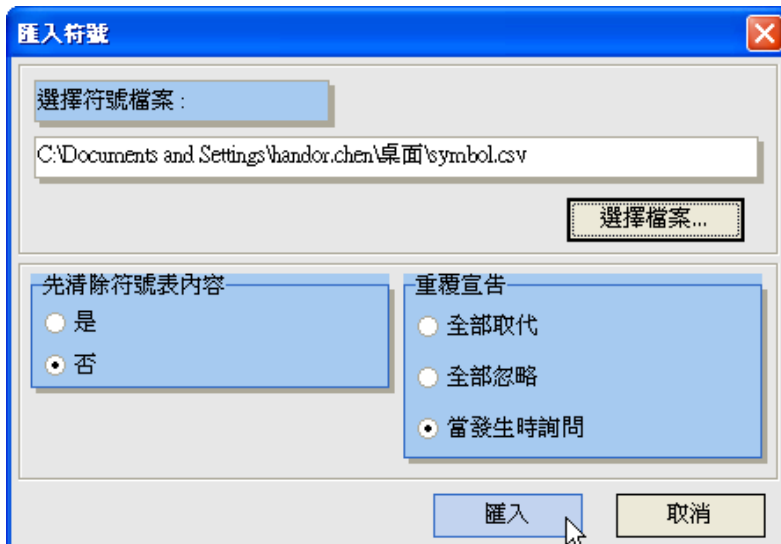
Step 2: 按鼠标右键，点选汇入符号表 (Import Symbols)

區域符號					
類別...	符號名稱	位址	資料型態...	初始值	符號註解...
VAR	ADD_P7		add_fb		

00001
M2000
K7-check
00002
00003

2. 选择汇入符号文件及汇入至符号表的方式

12 POU 阶梯图编辑模式



- 选择符号档案：选择要汇入符号文件
- 先清除符号表内容：点选「是」表示要汇入符号之前会清除符号表里面的符号，点选「否」是不清除符号表里面的符号。
- 重复宣告：点选「全部取代」时当遇到汇入符号与符号表中的符号一样将取代。点选「全部忽略」时当遇到汇入符号与符号表中的符号一样将忽略不取代。点选「当发生时询问」时当遇到汇入符号与符号表中的符号一样时会有对话框询问是否取代。

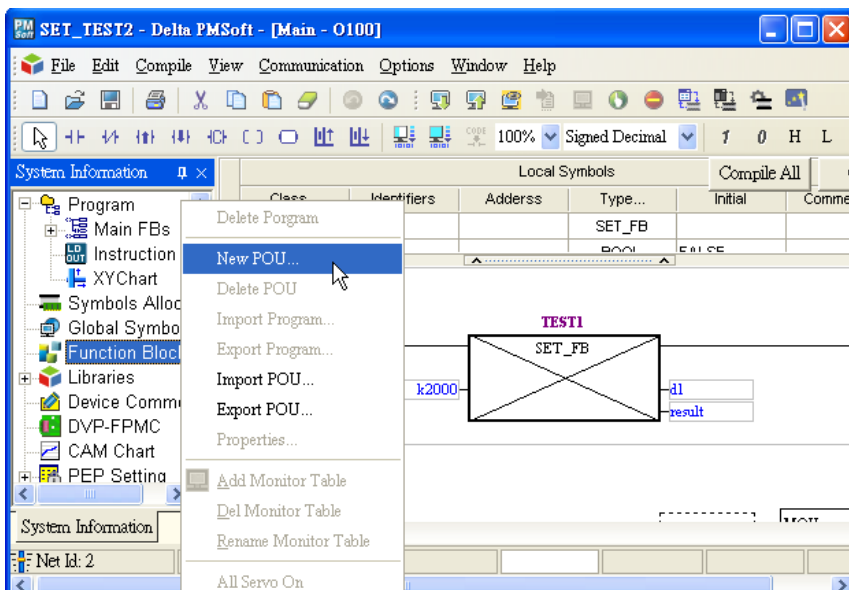
12.1.3 新增 POU 功能块

功能块能简化程序且可以重复使用。透过全域符号及功能块区域符号表中类别之 **INPUT** 定义输入符号，作为功能块运作所需数据来源输入接脚接口，而类别 **OUTPUT** 定义符号为输出，为功能块运作结果输出之接脚接口，使用者可针对自己的需求设计功能块的功能。

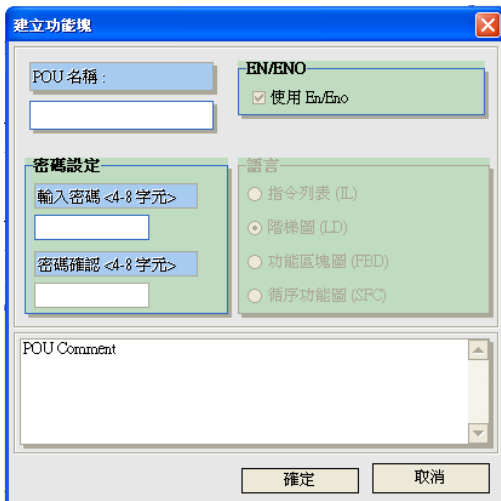
■ 新增 POU 功能块，操作步骤：

当新增项目时，首先要开启一个 POU 做为程序编辑区。建立 POU 的方法说明如下。

(1) 在「系统信息区」的「功能块」功能节点上按鼠标右键，选取「新增 POU」。



(2) PMSoft 会显示「建立功能块」设定窗口。



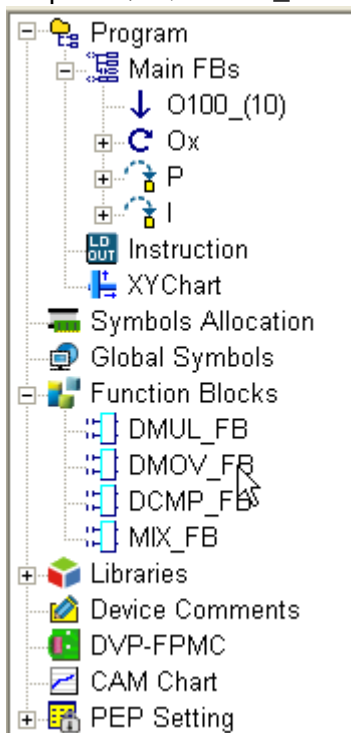
POU 名称： 一个项目中不可使用相同名称的 POU。

POU 命名原则

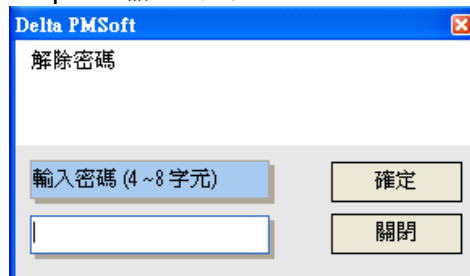
- 不允许有符号如(~!@#\$%^&*())...等
- 不允许同一个项目中有相同 POU 名称
- 不区分大小写。
- 最多允许 20 个字符。

密码设定： 当 POU 设定密码，要解密才能看到功能块的程序，如下图所示开启 DMOV_FB，而 DMOV_FB 功能块已被设定密码，要经过解密才能看到 POU 内部程序，若密码输入错误，会显示密码不合法讯息。

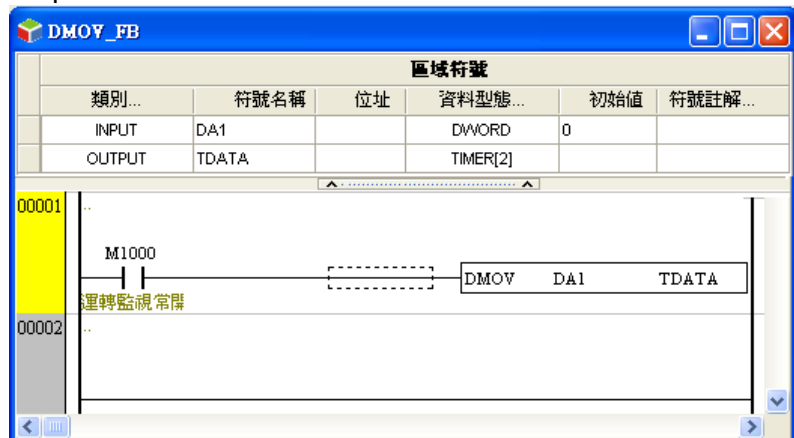
Step 1: 双击 DMOV_FB



Step 2: 输入密码

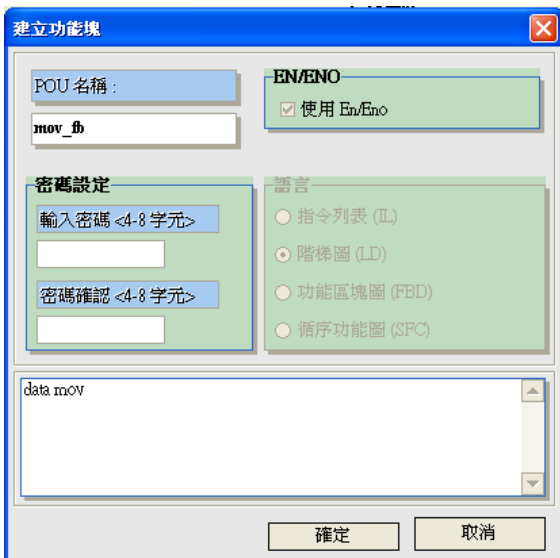


Step 3: 结果

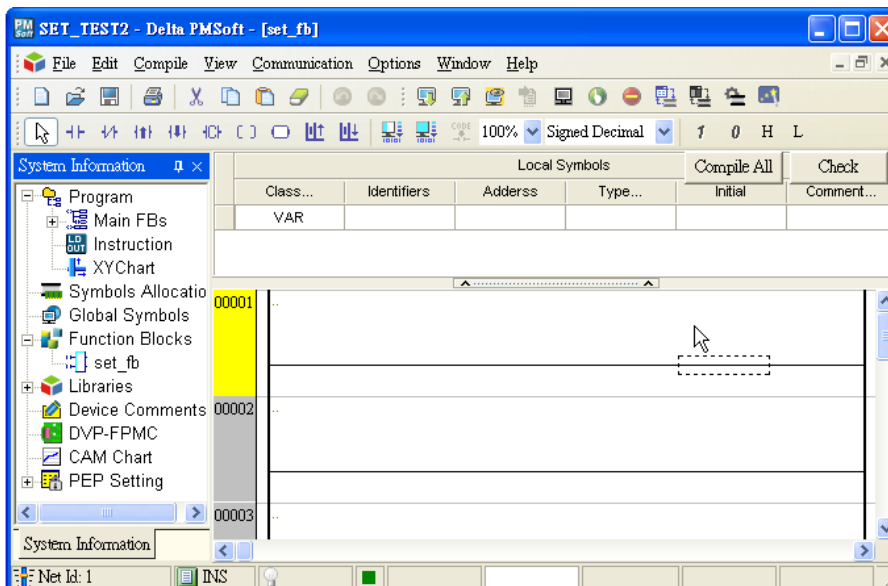


POU 批注： 设定窗口下方之空白区域，提供使用者批注 POU 使用功能。如下图所示：

12 POU 阶梯图编辑模式



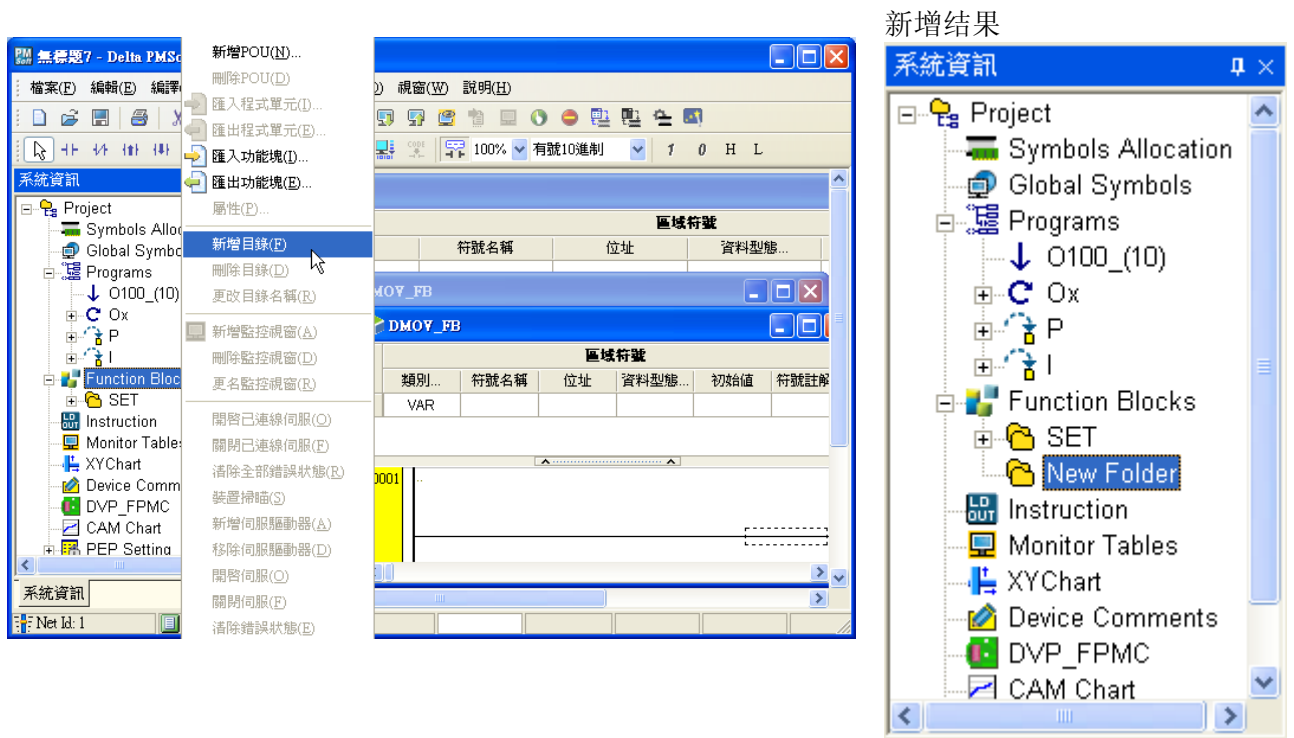
(3) 按「确定」后显示阶梯图编辑画面。



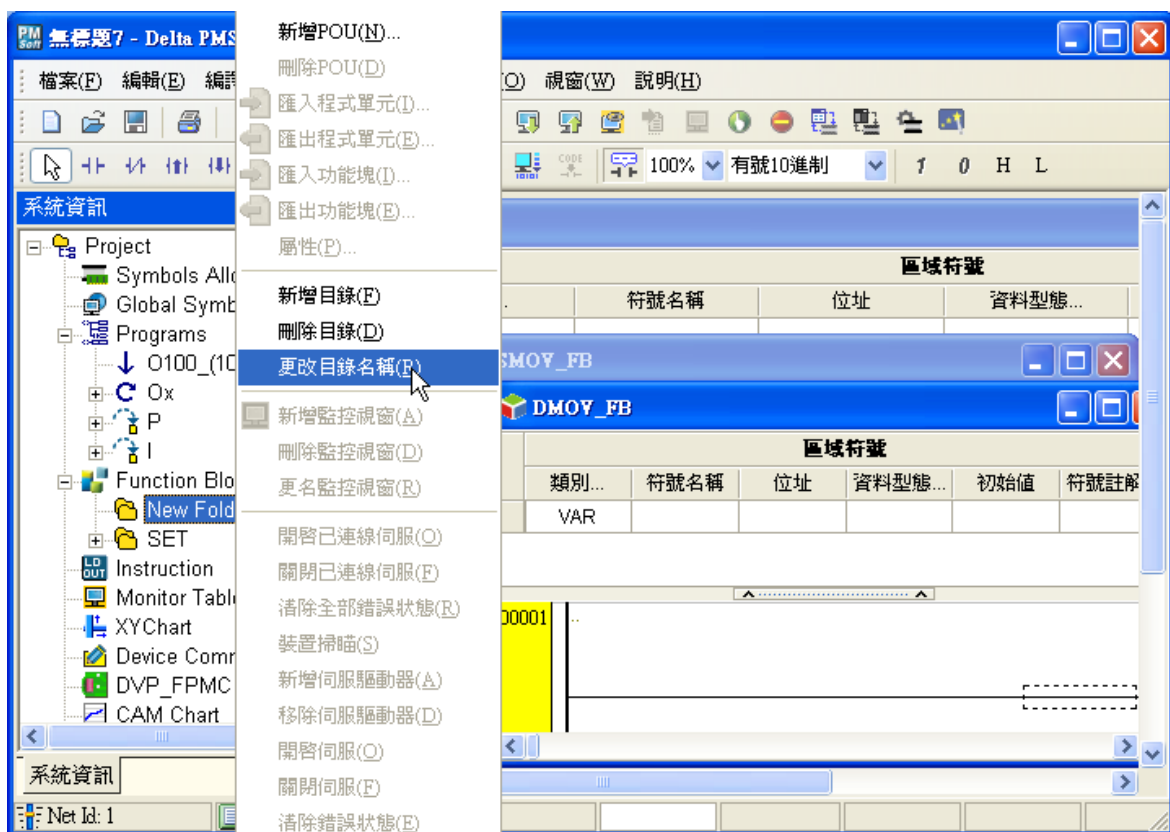
12.1.4 新增 POU 管理目录

让使用者可以在 **Function Block** 下建立目录，以方便分类管理。下列说明如何建立与使用 **Function Block** 的管理目录：

(1) 新增目录：在「系统信息区」上，光标移在「功能块」节点上按鼠标右键，选取「新增目录」，在 **Function Block** 下一阶层自动新增一个 **New Folder** 的目录



(2) 更改目录名称：光标移在「New Folder」节点上按鼠标右键，选取「更改目录名称」

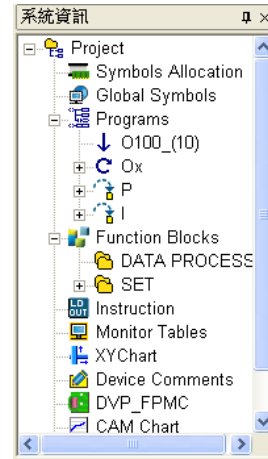


此时会显示更改目录名称窗口，在窗口内在编辑完新名称后，按下「确定」后即完成名称的更改。

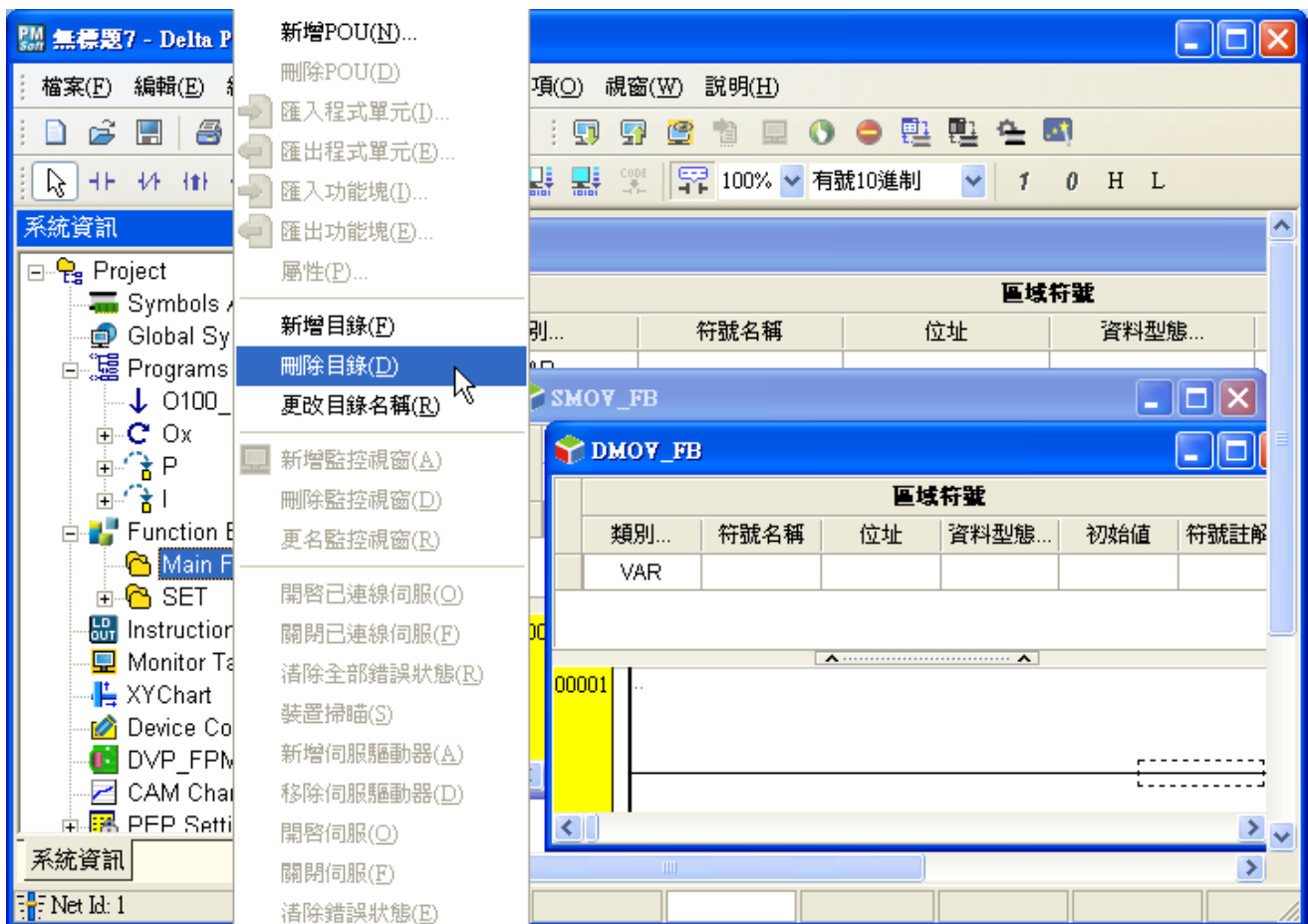
12 POU 阶梯图编辑模式



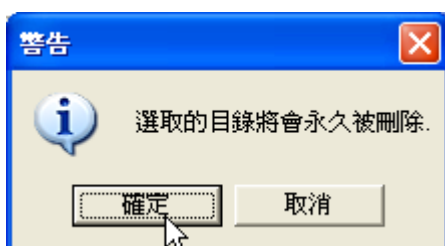
更改目录名称后的结果



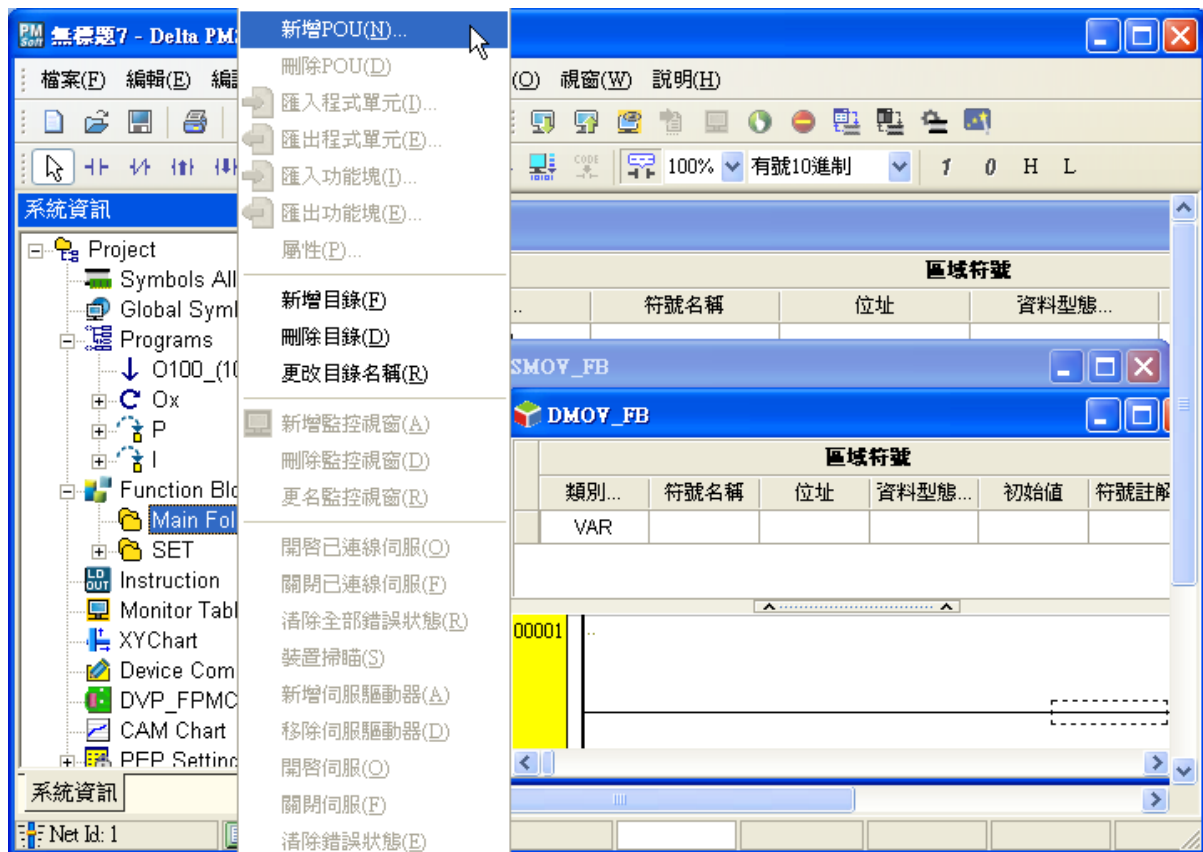
(3) 删除目录: 此功能可将已建立的目录删除, 将光标移在欲删除的目录节点上按鼠标右键, 选取「删除目录」



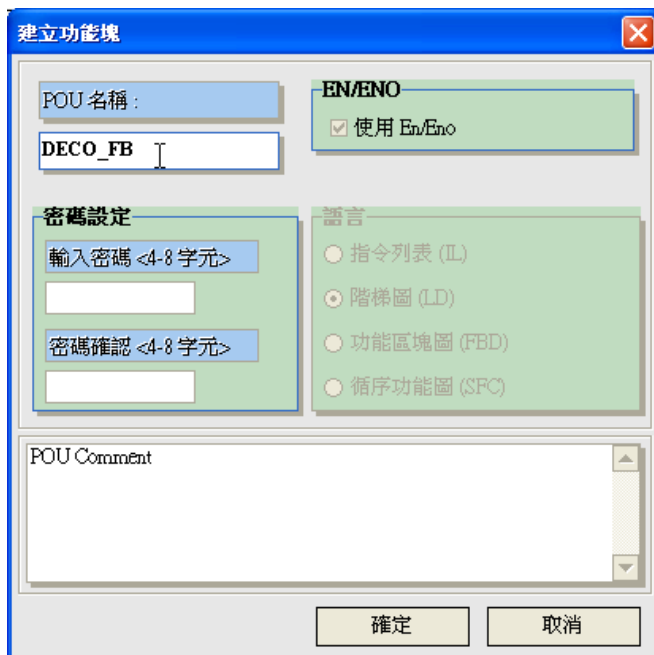
此时会出现询问窗口, 若选择「确定」, 则所选取的目录将会被删除



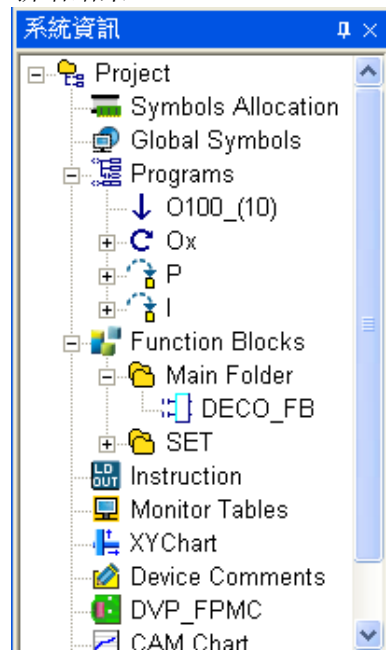
(4)在目录中新增 POU：在目录节点上点鼠标右键选择「新增 POU」



即显示「建立功能块」窗口，命名规则参考先前介绍的 POU 命名规则，将名称键入后点选确定即可在该目录下看见新增 POU 的结果。



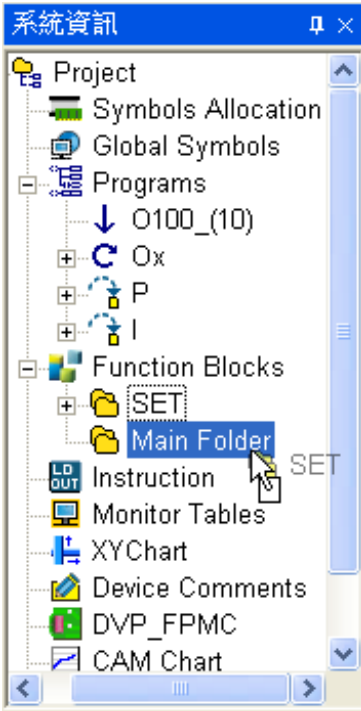
新增结果



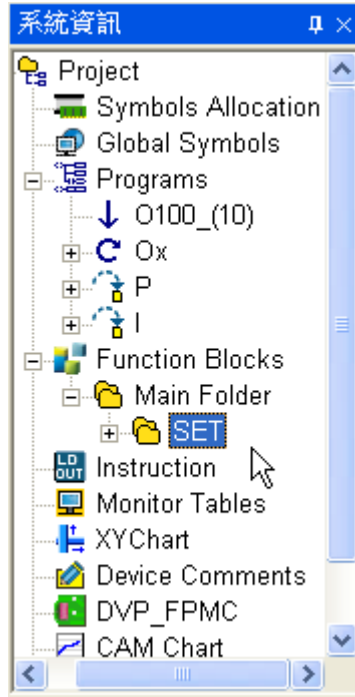
(5) 鼠标拖曳移动节点：使用鼠标按住欲移动的目录或是 POU 拖拉至欲放置的目录，即可将 Function Blocks 节点下的目录或 POU 节点搬移到其它的目录节点下。

12 POU 阶梯图编辑模式

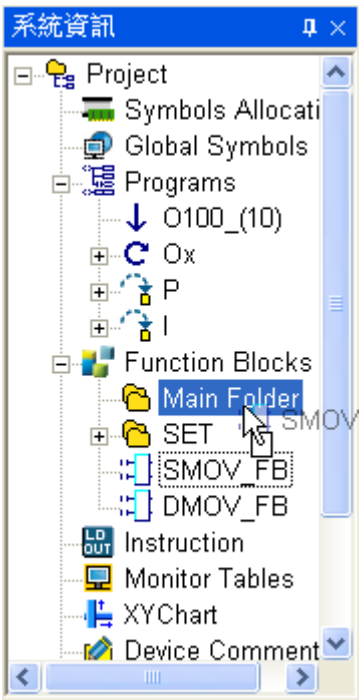
移动目录至另外一个目录底下



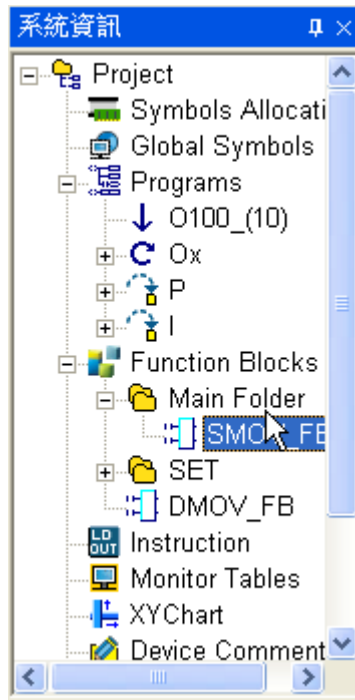
移动结果



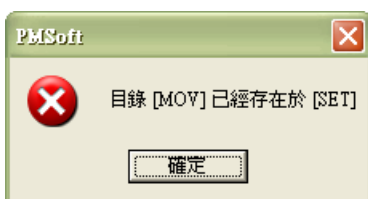
移动 POU 至目录底下



移动结果



要注意的是若被拖曳的节点为目录时，该目录节点下的所有节点也会跟着移动；若拖曳到的目的节点下的同一阶层有相同的目录名称时，会有警告讯息产生告知已有相同的目录名称。

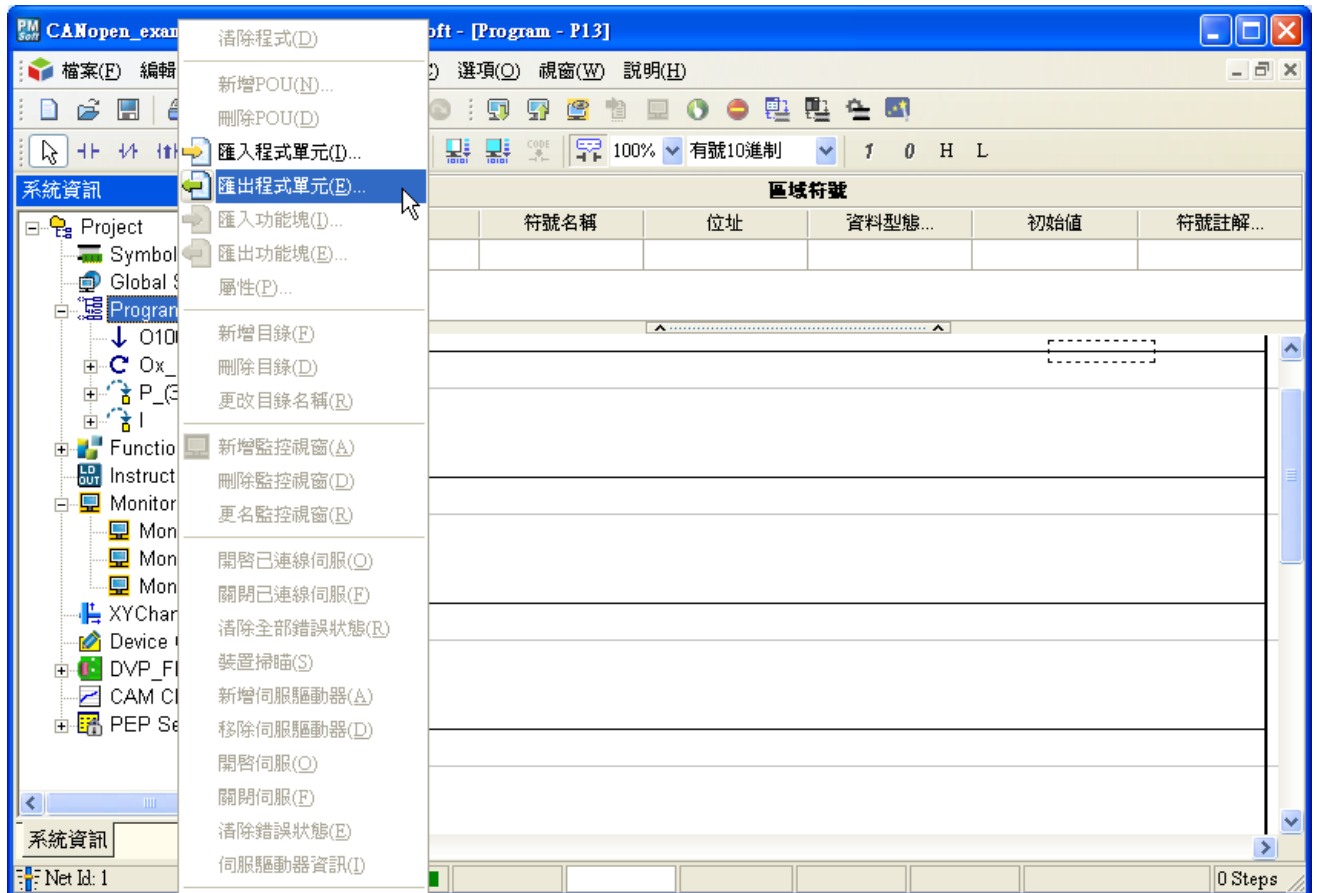


12.1.5 汇出 POU 功能块

将项目中编写的功能块汇出，以便编写其它项目程序时可汇入使用。POU 的型态分为程序与功能块两种，下列说明汇出功能块方法：

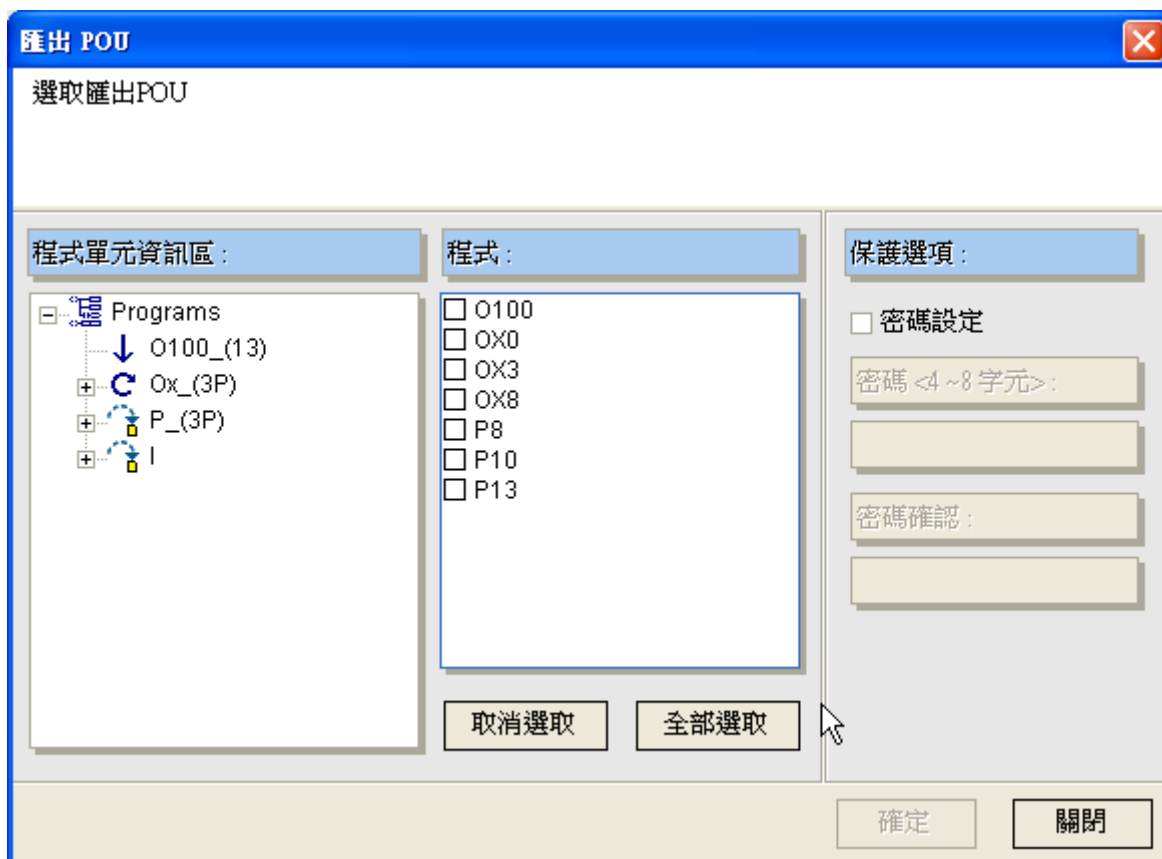
- 程序型态 POU 汇出：

1. 在「系统信息区」上，光标移在「Programs」节点上按鼠标右键，选取「汇出程序单元」。

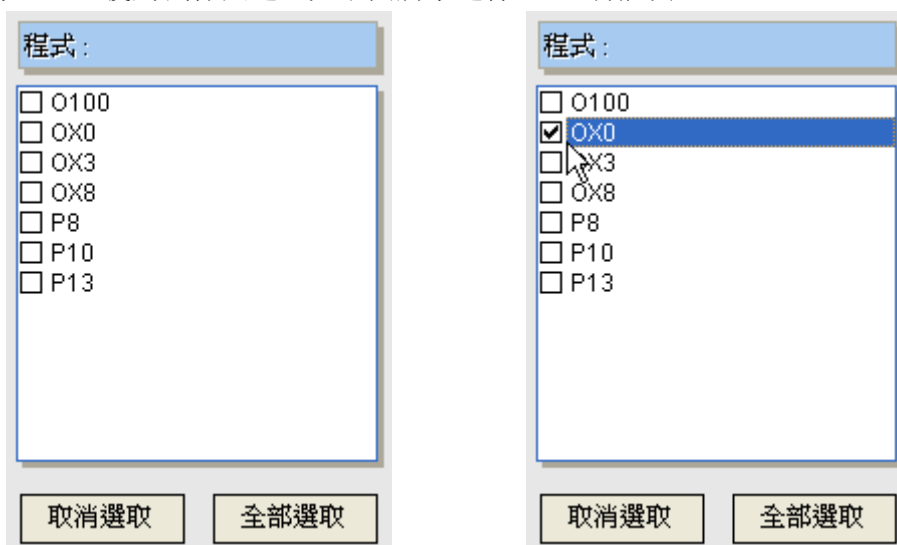


2. PMSoft 会显示程序单元信息区与程序区下所有的功能块，可勾选需要汇出之 POU。若在程序单元信息区指定区域(主程序区 O100、运动子程序区 Ox、子程序区 P)，则在程序画面仅显示该目录中的 POU。

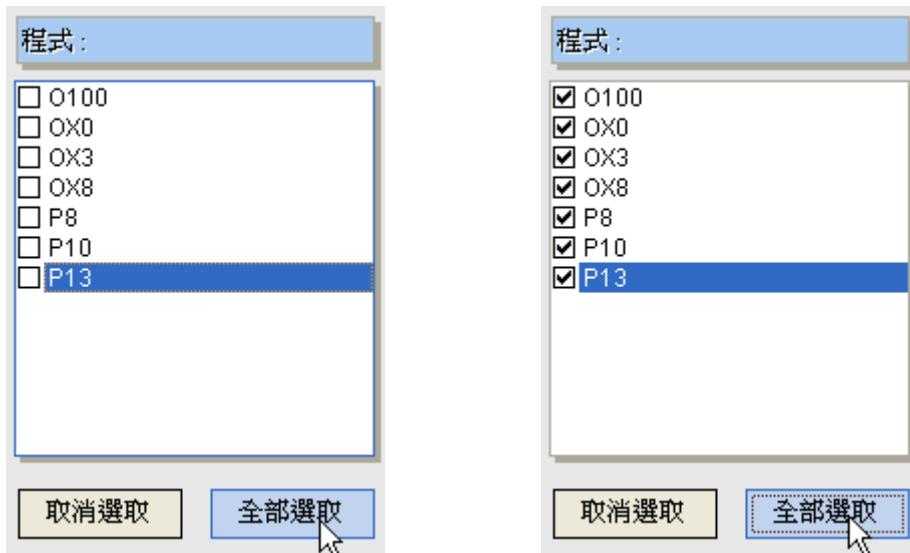
12 POU 阶梯图编辑模式



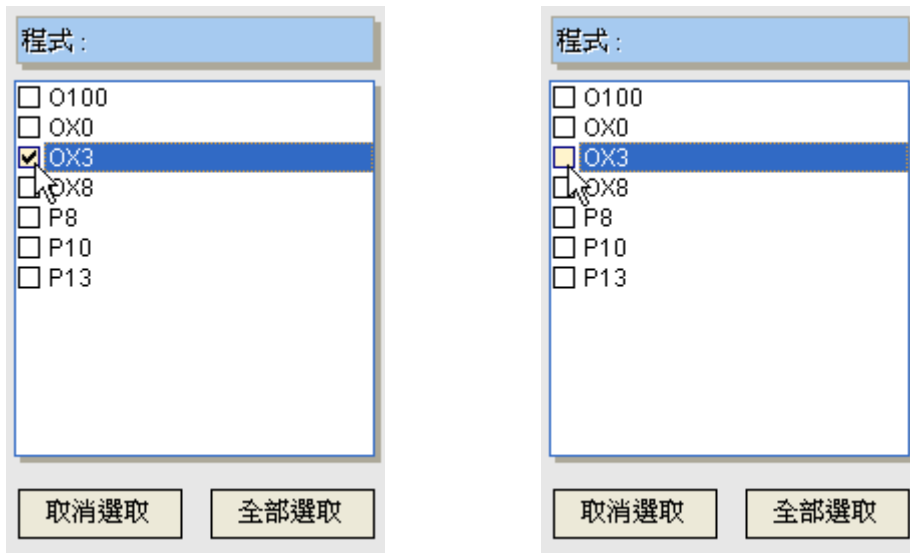
方法一：使用鼠标点选，如下图所示选择 OX0 功能块。



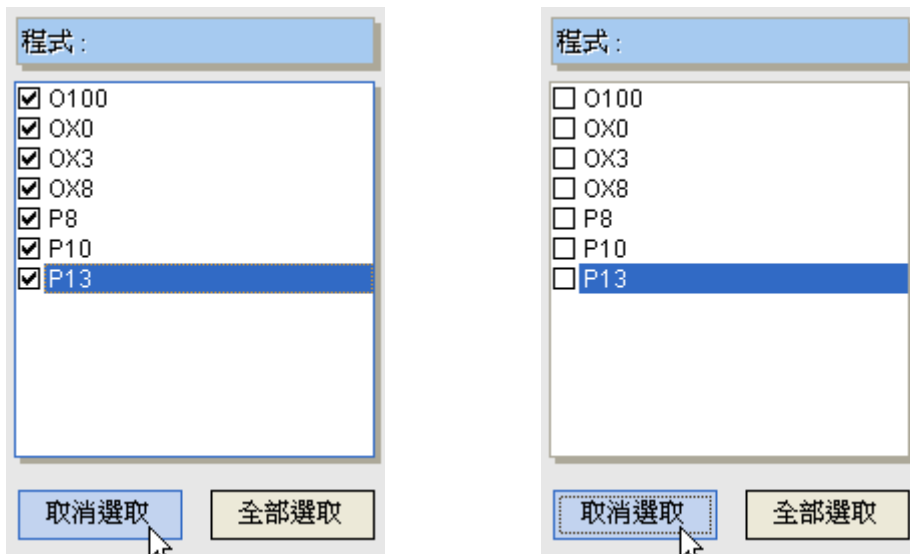
方法二：选择全部，如下图所示点选「全部选取」选择全部功能块。



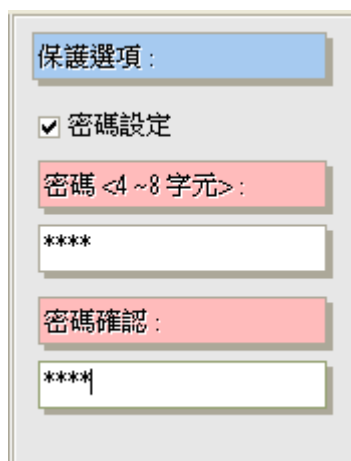
方法三：取消选择，如下图所示用鼠标点选已勾选的功能块。



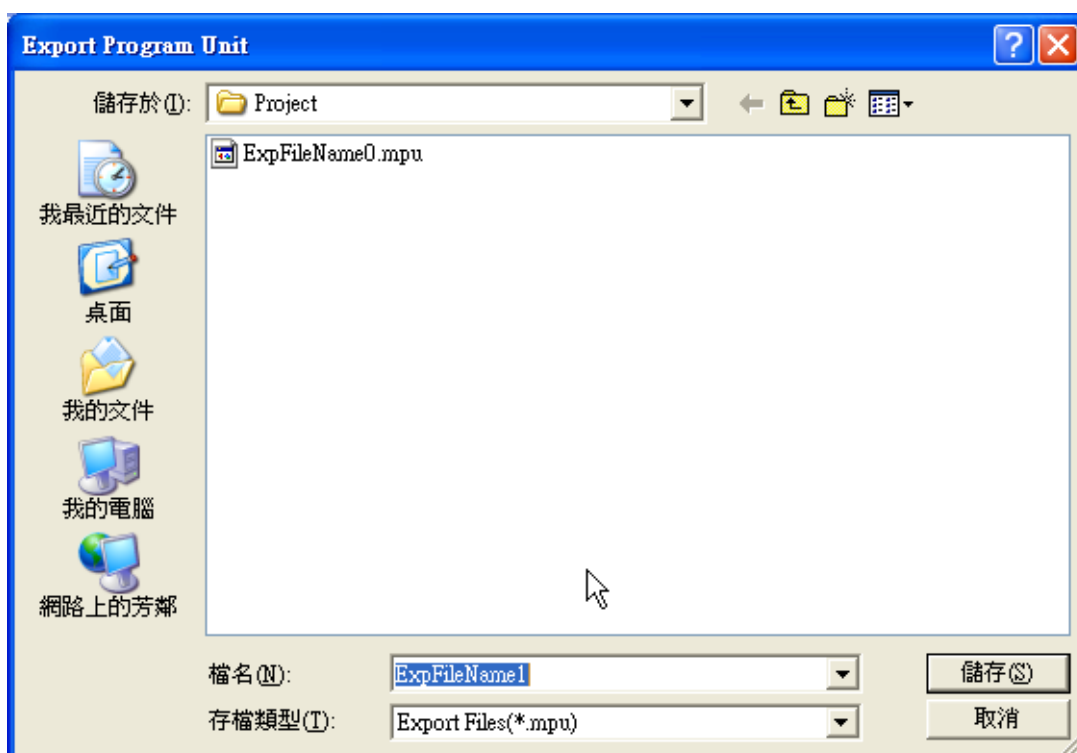
方法四：全部取消，如下图所示点选「取消选取」取消全部功能块。



3. Protection Option: 功能块链接库加密保护。勾选 Password setting, 设定密码及确认。

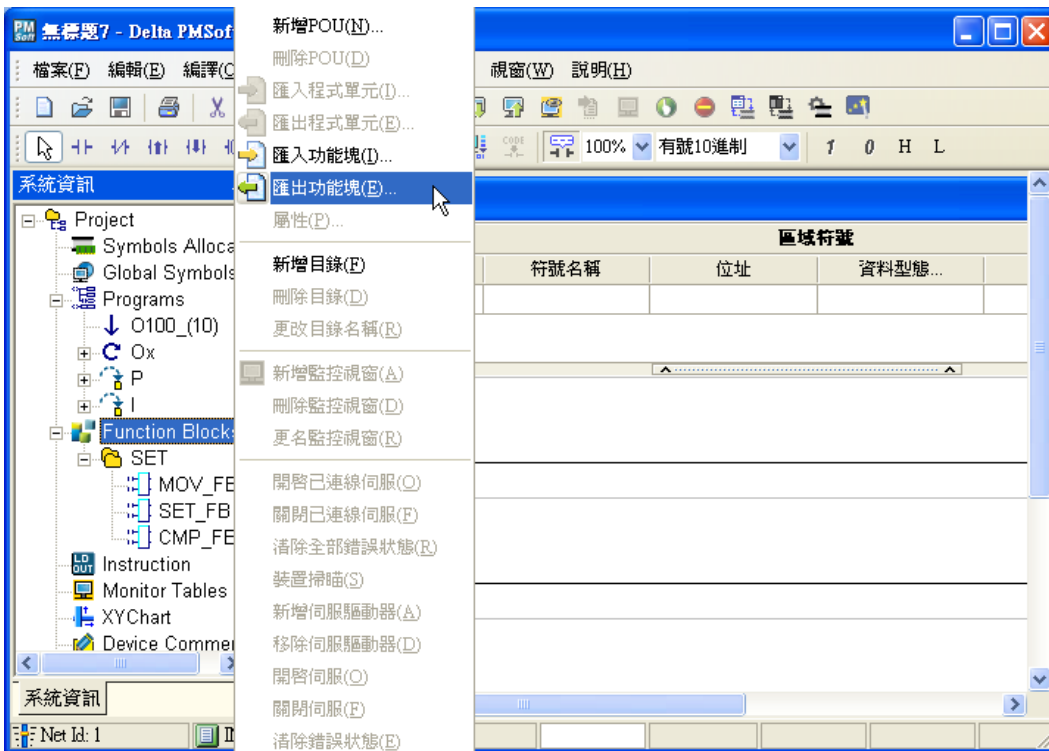


4. 點選確定後跳出存盤窗口，在此設定定文件名稱及路徑，擴展名為*.mpu。

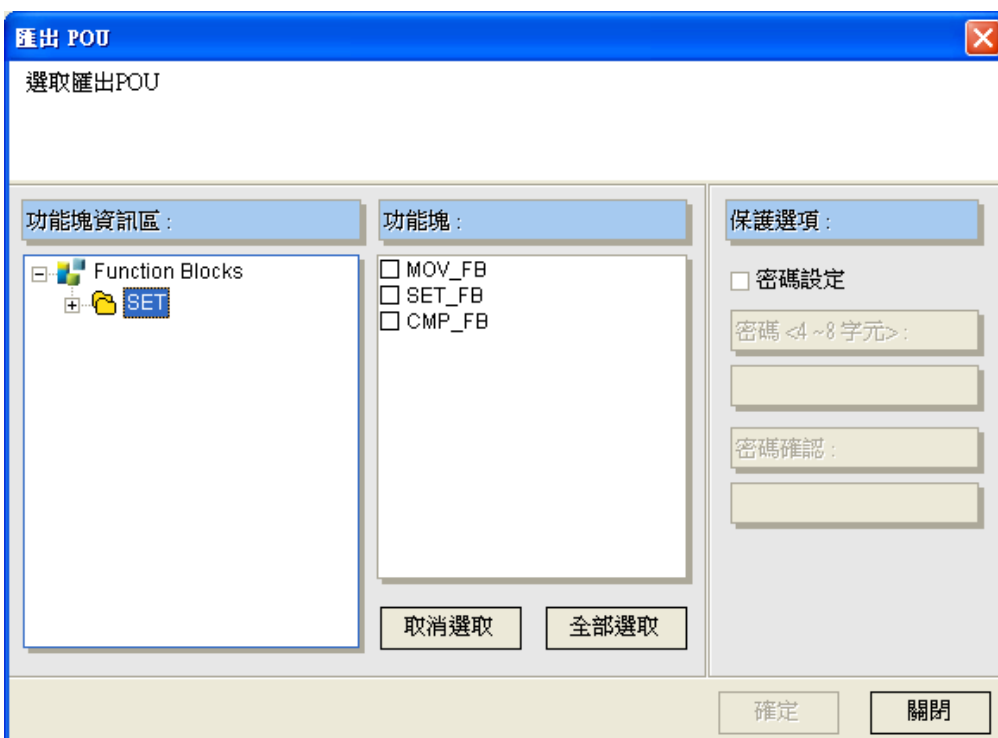


■ 功能块型态 POU 汇出

1. 在「系统信息区」上，光标移在「功能块」节点上按鼠标右键，选取「汇出 POU」。

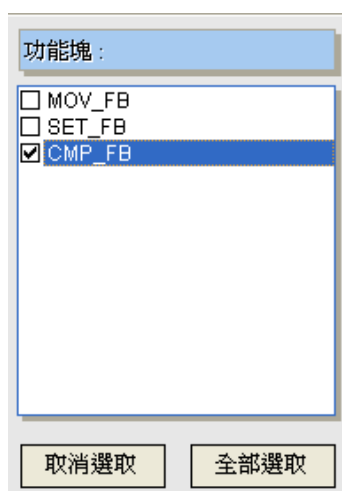
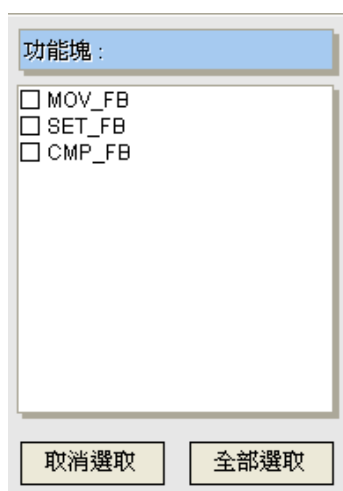


2. PMSoft 会显示功能块信息区与 Function Blocks 下所有的功能块，可勾选需要汇出之功能块 POU。若在功能块信息区指定目录，则在功能块画面仅显示该目录中的功能块。

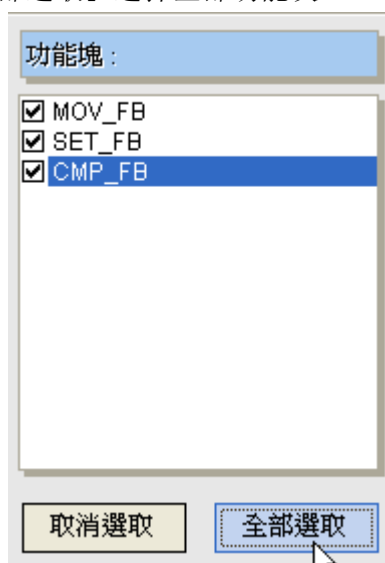
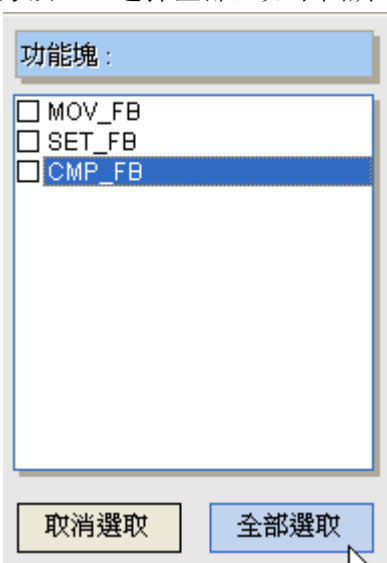


方法一：使用鼠标点选，如下图所示选择 CMP_FB 功能块。

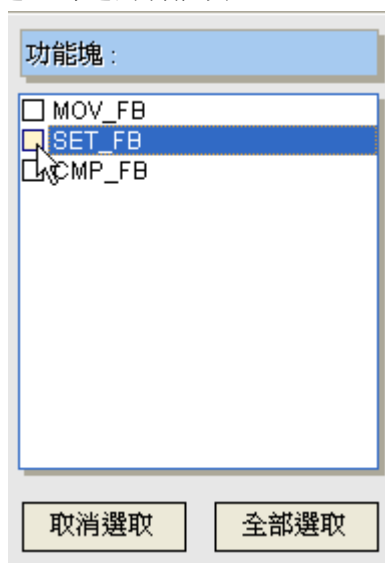
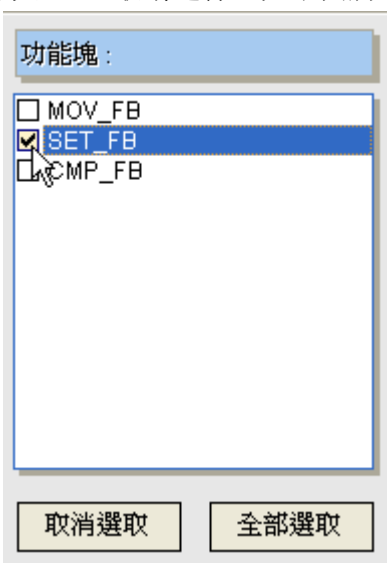
12 POU 阶梯图编辑模式



方法二：选择全部，如下图所示点选「全部选取」选择全部功能块。



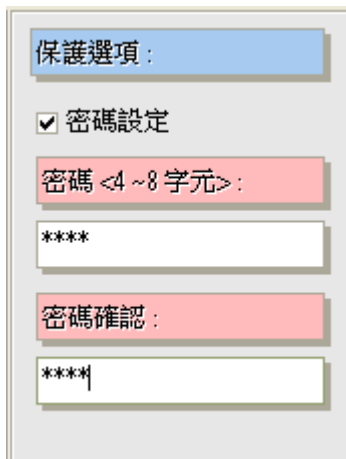
方法三：取消选择，如下图所示用鼠标点选已勾选的功能块。



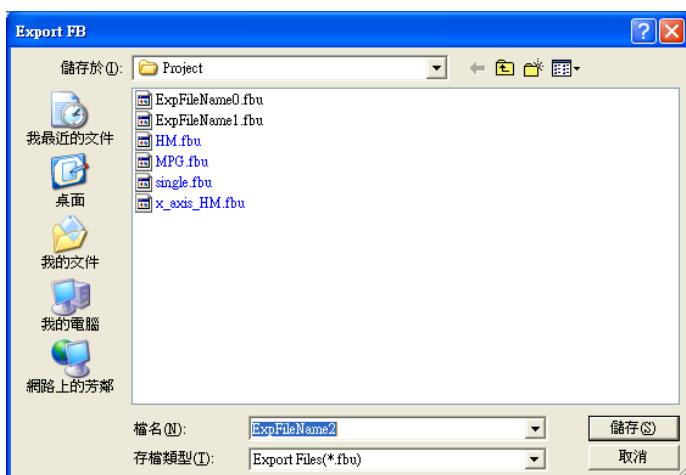
方法四：全部取消，如下图所示点选「取消选取」取消全部功能块。



3. Protection Option: 功能块链接库加密保护。勾选 Password setting, 设定密码及确认密码。



4. 點選确定后跳出存盘窗口，在此设定定文件名称及路径，扩展名为*.fbu。



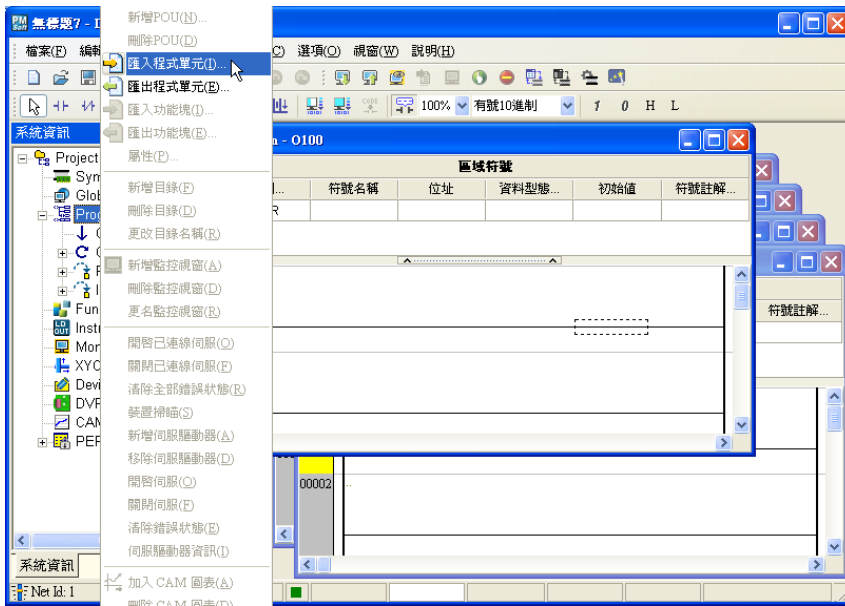
12.1.6 汇入 POU 功能块

将功能块链接库汇入项目程序，免去使用者花费时间编写同样功能块，POU 的型态分为程序与功能块两种,下列说明汇入功能块方法：

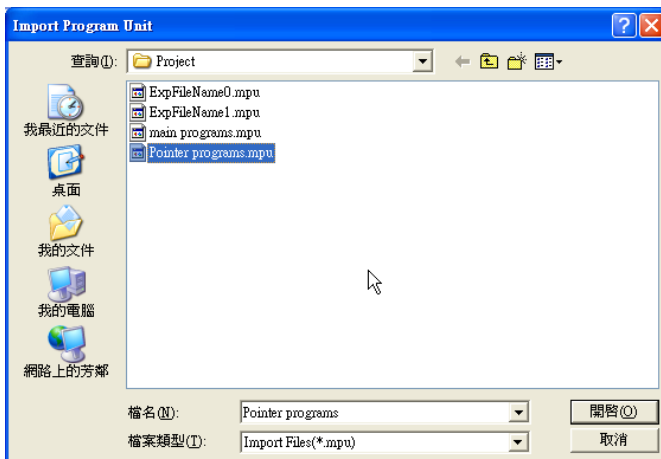
- 程序型态 POU 汇入：

12 POU 阶梯图编辑模式

1. 在「系统信息区」上，光标移在「Programs」节点上按鼠标右键，选取「汇入程序单元」。



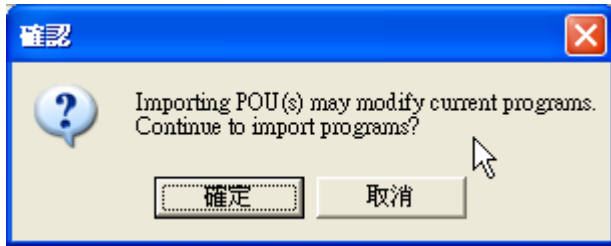
2. PMSoft 显示安装文件里头目录中的程序单元档案(*.mpu),或是其它路径中的程序单元档案,选择档案,按开启。



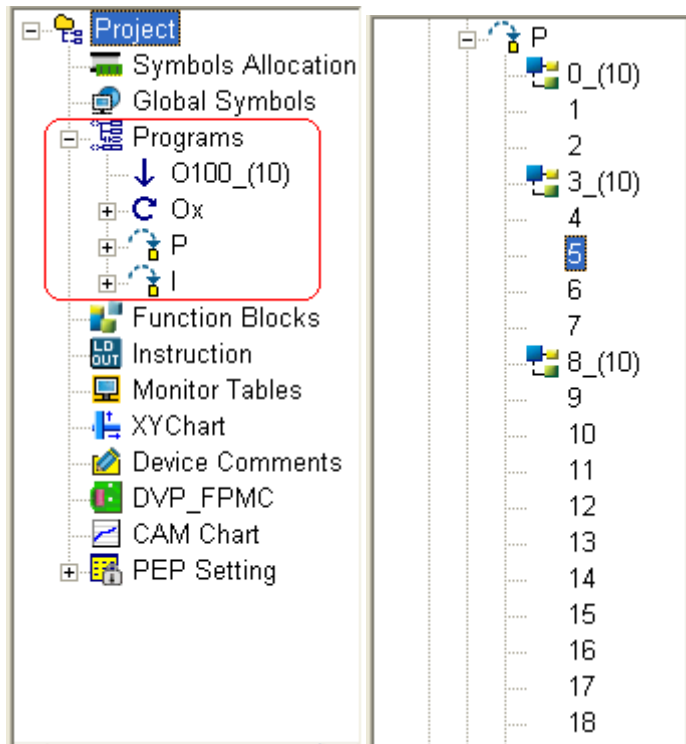
3. 解密码: 若程序单元的连接库档案有加密,则如下图所示输入正确密码,才能汇入功能块。若功能块连接库档案无加密,则系统略过此步骤。



4. 出现确认画面: 当汇入程序区单元的 POU 时,会告诉使用者将会变更现在主程序,确认是否继续,若要汇入主程序区档案点选继续。

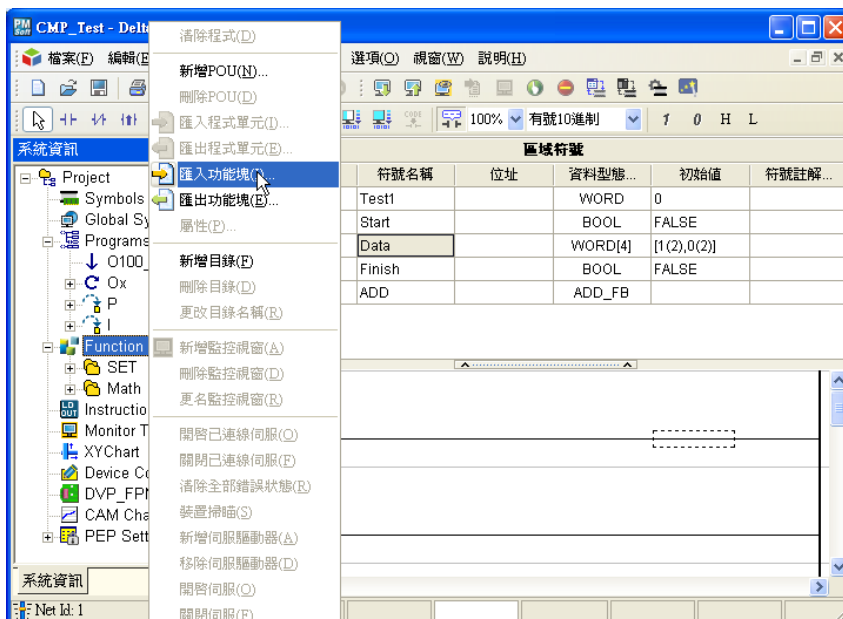


5. 结果：如下图所示，汇入的程序可点选 Programs 的各项程序区中看见。



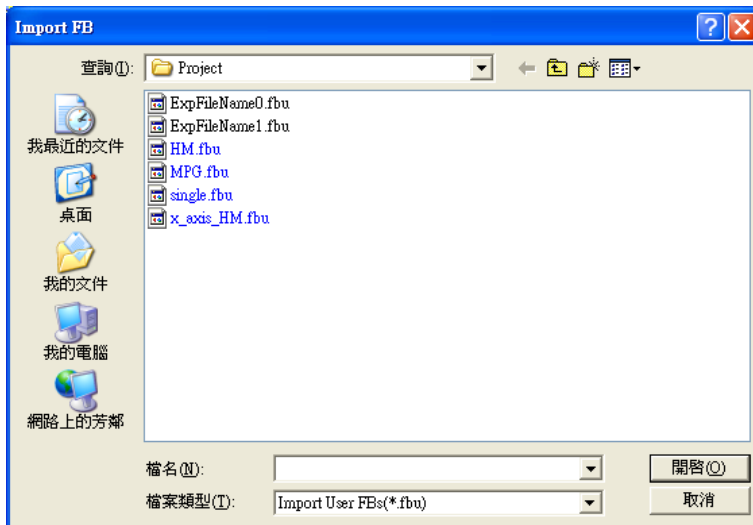
■ 功能块型态 POU 汇入

1. 在「系统信息区」上，光标移在「功能块」节点上按鼠标右键，选取「汇入功能块」。



12 POU 阶梯图编辑模式

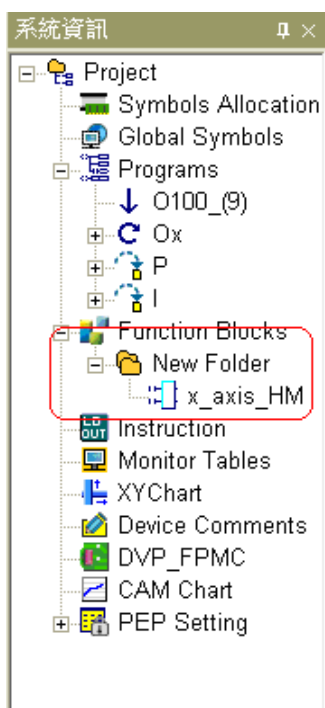
2. PMSoft 显示安装文件里头目录中的功能块档案(*.fbu)，或是其它路径中的功能块档案，选择档案，按开启。



3. 解密码：若功能块链接库档案有加密，则如下图所示输入正确密码，才能汇入功能块。若功能块链接库档案无加密，则系统略过此步骤。



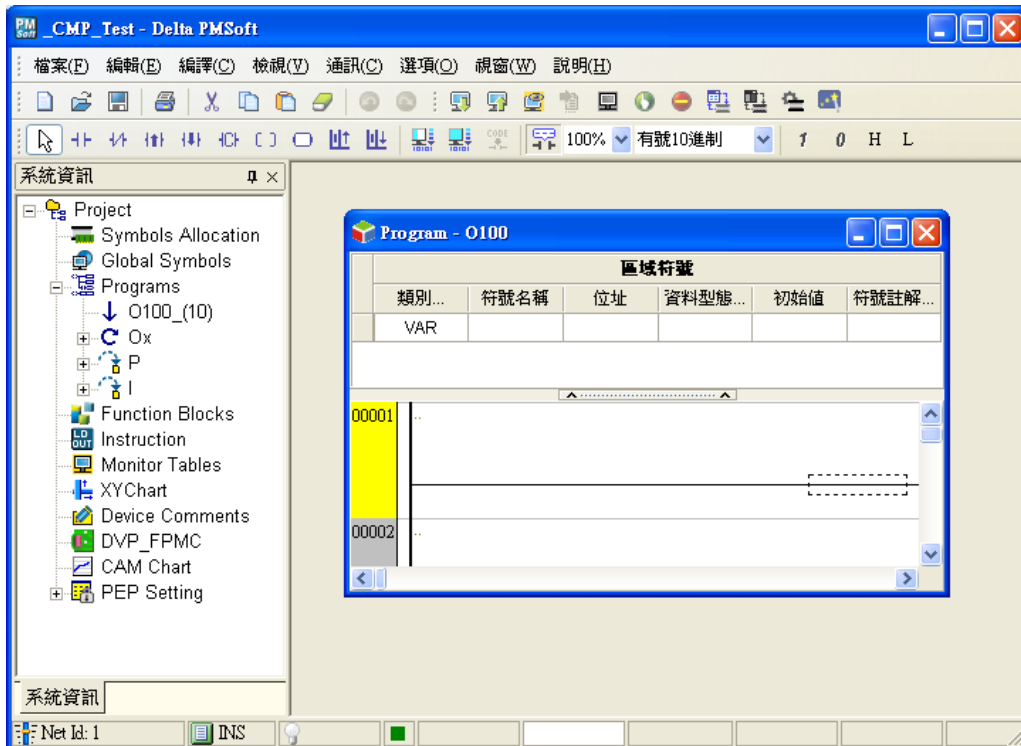
4. 结果：如下图所示，红色框框为汇入之功能块



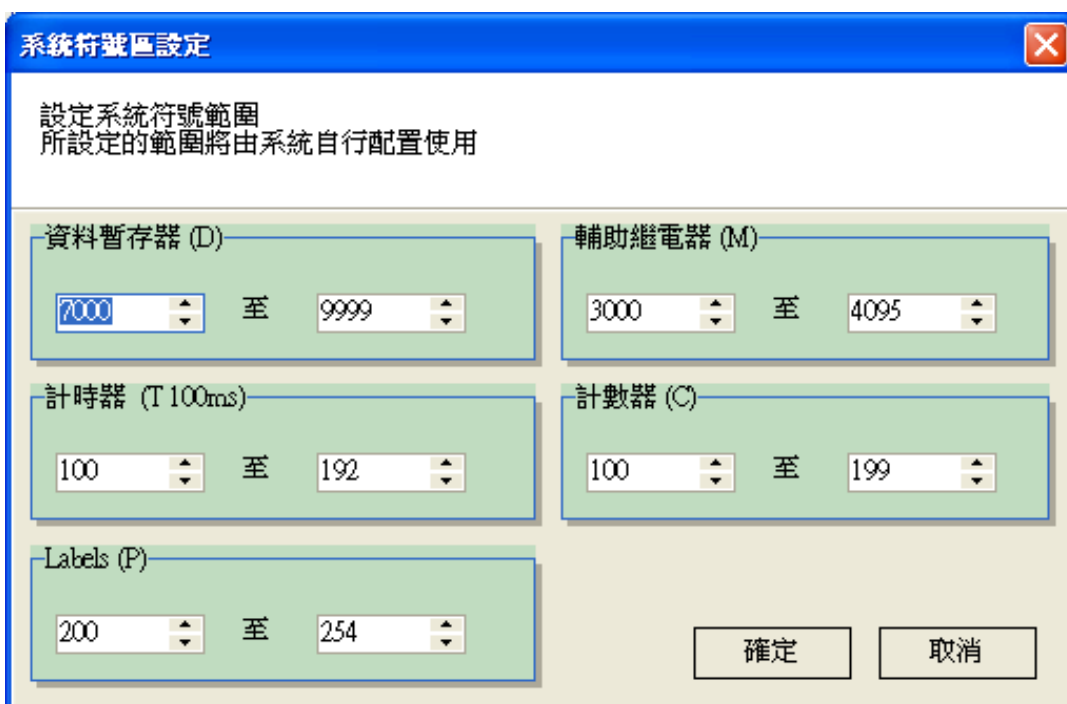
12.1.7 系统装置配置表

PMSoft 系统装置配置表供使用者设定数据缓存器、辅助继电器、定时器、计数器及指针装置范围。使用者可编辑设定装置范围，以在编译时由系统配置符号的定义地址装置。开启系统装置配置表的方法如下说明：

1. 在「系统信息区」双击「系统装置配置设定」如下图所示。



2. PMSoft 会显示系统装置设定区。



12 POU 阶梯图编辑模式

装置范围设定

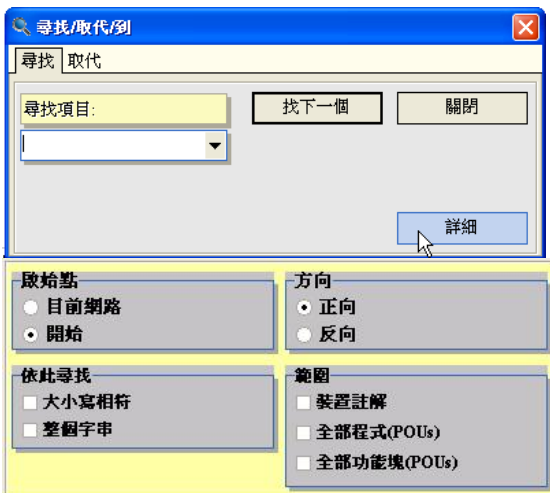
- D 数据缓存器：初始设定范围 D7000~D9999，可设定范围为 D0 ~ D9999，设定要避免特 D 数据缓存器。
- T 计时缓存器：初始设定范围 T100~T192，可设定范围为 T0 ~ T255。
- C 计数缓存器：初始设定范围 C100~C199，可设定范围为 C0 ~ C255。
- M 辅助缓存器：初始设定范围 M3000~M4095，可设定范围为 M0 ~ M4095，设定要避免特 M 辅助缓存器。
- P 指针缓存器：初始设定范围 P200~P254，20PM00M 可设定范围为 P0 ~ P255，20PM00D 可设定范围为 P0 ~ P254。

12.1.8 阶梯图寻找

PMSoft 的寻找功能分为阶梯图寻找以及 IL 寻找，以下内容说明阶梯图寻找。阶梯图寻找可应用在 POU 程序寻找与符号表寻找，分别说明如下。

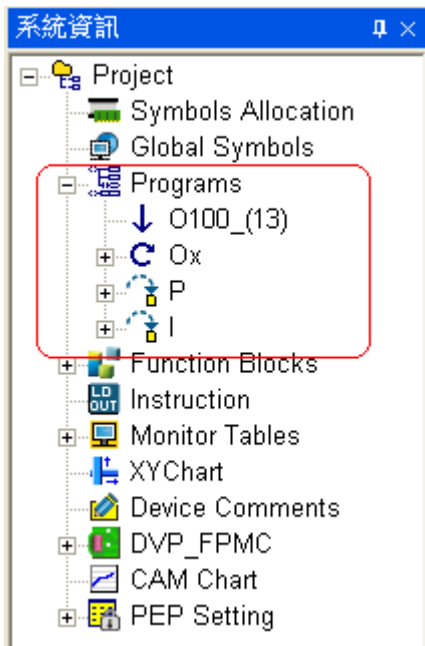
■ POU 程序寻找：

在 POU 阶梯图中按下鼠标右键，并从功能选单中选取寻找(F)命令。使用此命令寻找程序中的指令、装置和符号，如果输入的指令、装置和符号存在，则会跳到寻找的指令或是装置，而寻找的方式可以按寻找对话框下角的「详细」按钮，会有新增寻找设定，如下图所示。

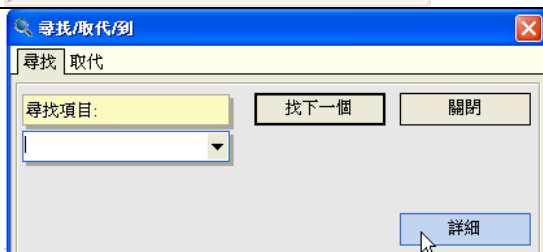
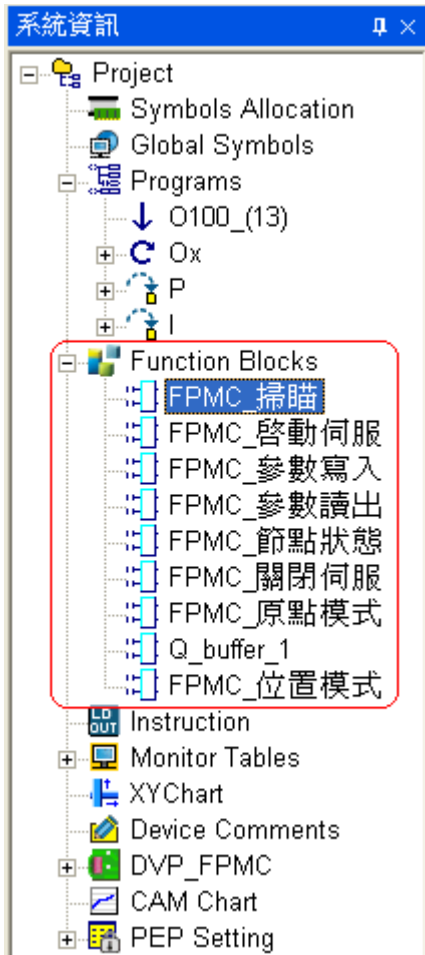


其中寻找方式有下列几项：

1. 目前网络：寻找是要从目前编辑的网络区段开始寻找。
2. 开始：寻找是要从目前编辑窗口开始位置开始寻找。
3. 大小写相符：寻找的数据大小写要都一样。
4. 整个字符串：寻找的数据是要整个字符串去比对
5. 正向：寻找的方式是由上往下寻找。
6. 反向：寻找的方式是由下往上寻找。
7. 装置批注：寻找的位置只能在装置批注区寻找。
8. 全部程序(POUs)：寻找范围为(O100, Ox0-Ox99, P0-P255)



9. 全部功能块(POUs): 寻找范围为使用者自订的功能块



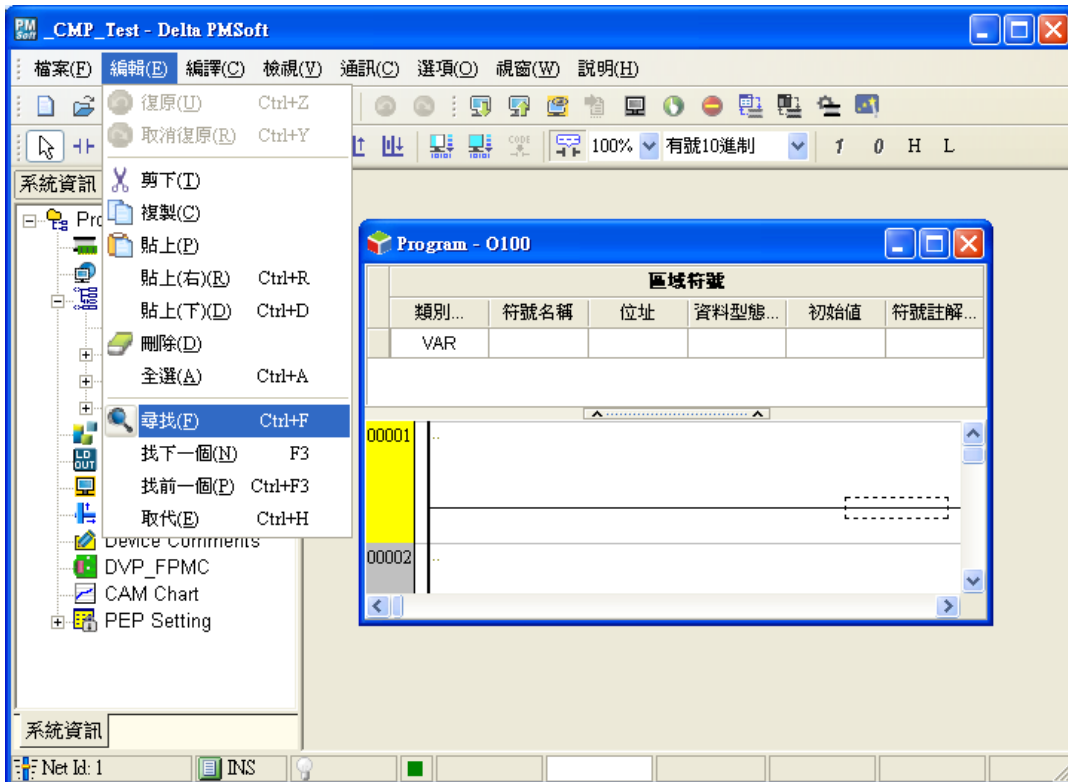
12 POU 阶梯图编辑模式

输入好寻找文字，设定好寻找方式，按『找下一个』按钮，就开始寻找，被找到的文字会被反白，再按一次「找下一个」按钮，会跳到下一个被找到的文字，以此类推，假如再按一次「找下一个」按钮，已没有数据可寻找，会显示「不存在」讯息窗口。



■ 开启寻找功能窗口的方法：

◇ 方法一：从功能列上选取编辑（E）中「寻找(F)」命令。



◇ 方法二：鼠标右键菜单中「寻找(F)」命令。

◇ 方法三：键入复合键（Ctrl）+（F）。

找下一个(N)

找下一个(N)的功能是以上一次寻找的文字为寻找数据，在阶梯程序中往下寻找，其寻找方式是根据寻找窗口里头的设定(下图所示)，找下一个(N)的功能就是寻找方向的正向。



◇ 方法一：从功能列上选取编辑（E）中「找下一个(N)」命令。

◇ 方法二：利用快速键，按键盘 F3 键。

◆ 方法三：鼠标右键菜单中「找下一个(N)」命令。

找前一个(P)

找前一个(P)的功能是以上一次寻找的文字为寻找数据，在阶梯程序中往上寻找，其寻找方式是根据寻找窗口里头的设定，找前一个(P)的功能就是寻找方向的反向。

◆ 方法一：从功能列上选取编辑 (E) 中「找前一个(P)」命令。

◆ 方法二：键入复合键 (Ctrl) + (F3)。

◆ 方法三：鼠标右键菜单中「找前一个(P)」命令。

■ 符号表寻找

符号表有全域符号表及区域符号表，寻找只能对当时点选的符号表寻找，而不能寻找其它 POU 的符号表中的装置和符号。开启寻找功能窗口方式如下：

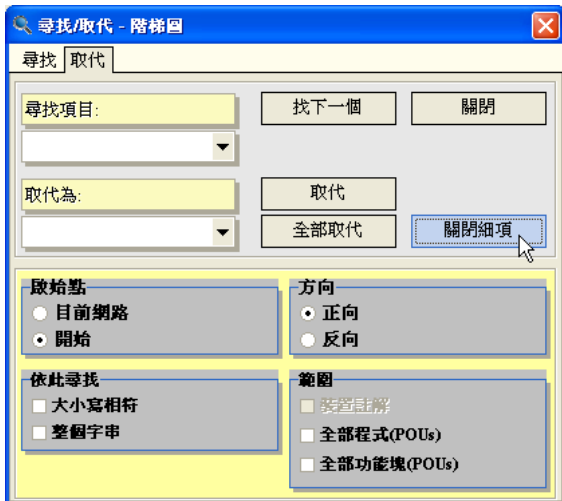
在符号窗体中按下鼠标右键，并从功能选单中选取寻找(F)命令



12.1.9 阶梯图取代

取代功能只能应用在程序上的指令、装置及符号的取代，而符号表没有取代功能。使用取代要搭配寻找功能才能完成取代动作，使用方式一开始要开启取代窗口，而取代功能窗口中，含有寻找功能，有寻找到对应的文字或数字才能做取代的动作，而执行取代过程中，是先做寻找动作再做取代的动作，寻找动作方式的设定同前面寻找功能设定。使用取代要注意是否取代完导致产生阶梯图中错误的指令、不合法的装置和错误的指令组合。下图为取代功能窗口(有开启寻找详细设定窗口)，其功能包含寻找功能。

12 POU 阶梯图编辑模式



下列是开取代功能窗口得方式

- ◆ 方法一：从功能列上选取编辑 (E) 中「取代(E)」命令。
- ◆ 方法二：鼠标右键菜单中「取代(E)」命令。
- ◆ 方法三：键入复合键 (Ctrl) + (F)。

12.2 POU 阶梯图程序操作

POU 阶梯图型式上与一般阶梯图有一些差异，但转成 IL 程序语法是一样的。主要差异是符号取代装置和 POU 功能块使用。

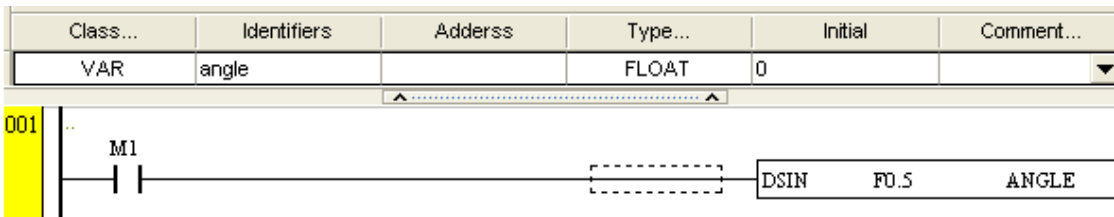
12.2.1 符号取代装置

在主程序 O100 主程序、OXn(n: 0 ~ 99)运动程序及 Pn(n: 0~255)子程序装置，都可由编辑窗口中区域符号表宣告的符号，或全域符号表之符号取代，取代规则如下：

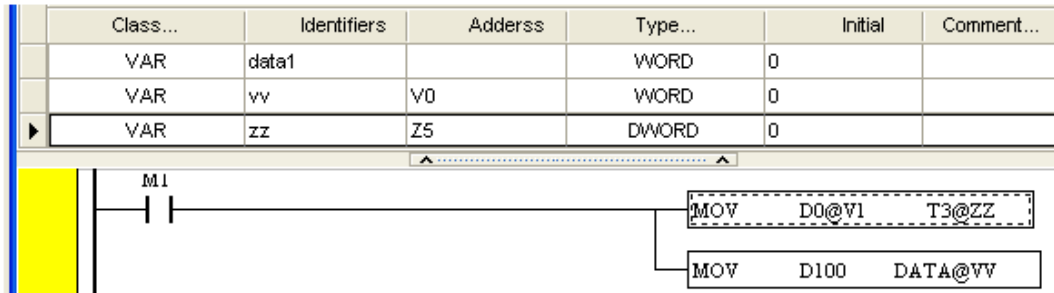
- 型态长度要符合指令操作数装置。如 DATA2 符号在 DMUL DATA2 SCALE2 RESULT2 中要符合 DMUL 指令操作数的规则，所以 DATA2 宣告型态只能 DWORD、WORD[2]、TIMER[2]、COUNTER[2]等 32 位数据，其它指令以此类推，确保 PMSoft 编译时配置正确的装置及足够装置。

Class...	Identifiers	Address	Type...	Initial	Comment...
VAR	data1		WORD	0	
VAR	result1		BOOL[3]		
VAR	data2		DWORD	0	
VAR	scale2		COUNTER[2]		
VAR	result2		LWORD	0	
VAR	data3		WORD[2]		
VAR	scale3		TIMER[2]		
VAR	result3		WORD[4]		

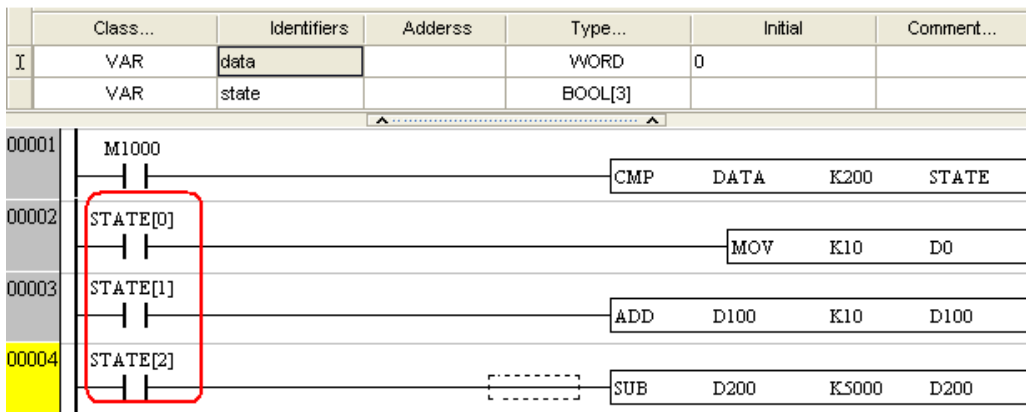
- 浮点数指令中 D 操作数之缓存器只能选用浮点数型态符号 FLOAT，如下图所示 DSIN 浮点数指令中 ANGLE 符号要宣告为 FLOAT。



- 使用索引时，必须加上@字符，如下图所示。



- 使用指针读取数组符号数据，如 `buffer` 为型态 `word[4]` 符号，可用 `buffer[2]` 读取第二个 WORD (指针 2) 的数据。如下图所示 `CMP` 指令最后一个操作数占用 3 位长度，所以 `STATE` 需宣告符号型态为 `BOOL[3]`，分别以 `STATE[0]`，`STATE[1]`，`STATE[2]` 表示指标 0，指标 1，指标 2 三种状态，以处理不同的运算。



- 使用指针及索引读取数组符号数据，如 `buffer` 数据型态为 `word[10]`，可用 `buffer[m]@Vn` 或 `buffer[m]@Zn` ($m: 0\sim 9, n: 0\sim 7$) 读取第 $m+Vn$ (Zn) 个位置的 `buffer` 数据 ($0\leq m+Vn$ (Zn) $\leq 9^*$)， Vn 与 Zn 也可使用符号宣告。如下图所示 `MOV` 指令中 `buffer[2]@v2`、`buffer[4]@z4`、`buffer[1]@Vd` 和 `buffer[6]@Zd` 用法，其中 Vd 是 V 装置符号 (符号表中宣告)，而 Zd 是 Z 装置符号 (符号表中宣告)。

12 POU 阶梯图编辑模式

區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解...	
VAR	buffer	D7000	WORD[10]			
VAR	Vd	V2	WORD	0		
VAR	Zd	Z4	DWORD	0		

*Ps:程序运作时，数组符号中数组指针与索引装置加总之数据长度不可超过数组符号宣告的长度。

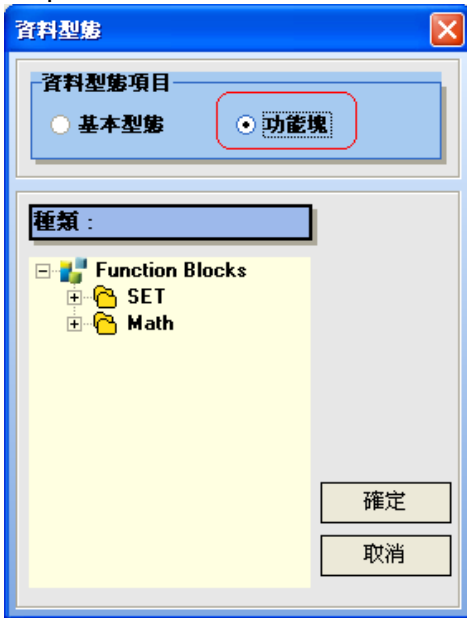
12.2.2 功能块 FB 使用

支持功能块的 POU 阶梯图与一般阶梯图在使用上有些差异，以下说明功能块 FB 的使用及使用功能块之 POU 阶梯图的特点。

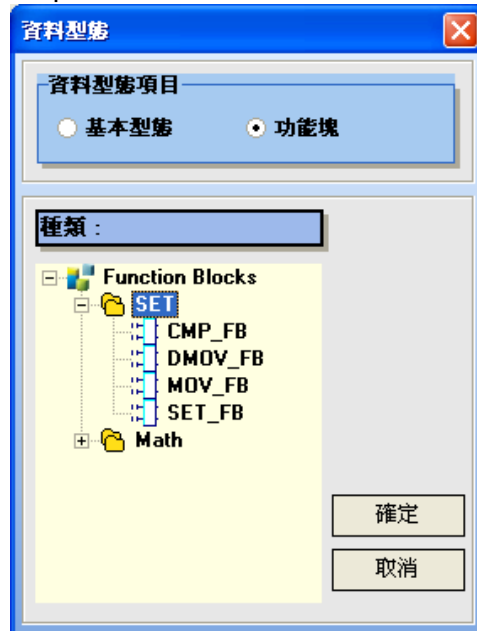
- 功能块的使用：使用功能有三个步骤：功能块宣告，功能块引用及编辑功能块的 Pin 脚，功能块宣告及功能块引用这两个步骤使用顺序对调不会影响使用结果。

步骤一：功能块宣告

Step 1: 选择宣告符号为功能块型態



Step 2: 选择目录后，點選欲宣告的功能块



Step 3: 结果

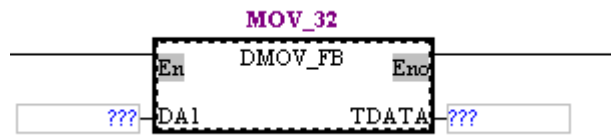
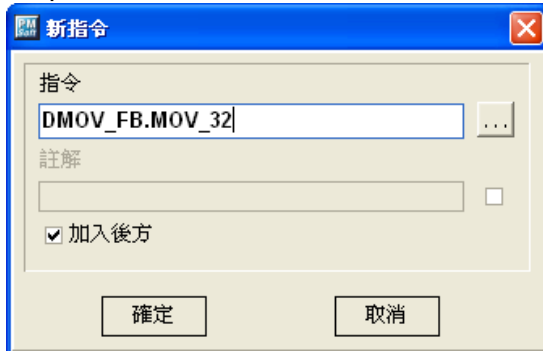
區域符號						
類別...	符號名稱	位址	資料型態...	初始值	符號註解	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
VAR	Data		WORD[12]			
I	VAR	ADD	ADD_FB			

步骤二：功能块引用有三种方法：

方法一：输入指令的方式输入功能块名称(DMOV_FB) + "." + 符号名称(MOV_32)。

Step 1: 输入 DMOV_FB.MOV_32

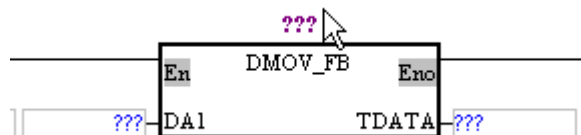
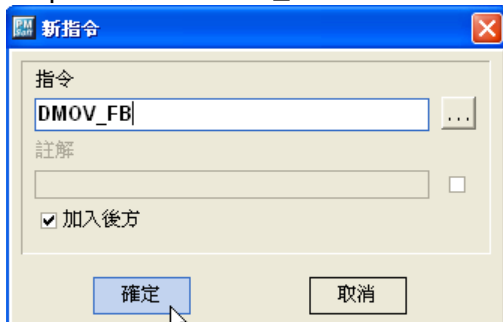
Step 2: 结果



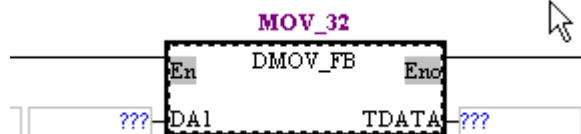
方法二：输入功能块名称(DMOV_FB)再输入符号名称(MOV_32)。

Step 1: 输入 DMOV_FB

Step 2: 点击???再输入符号名称 MOV_32



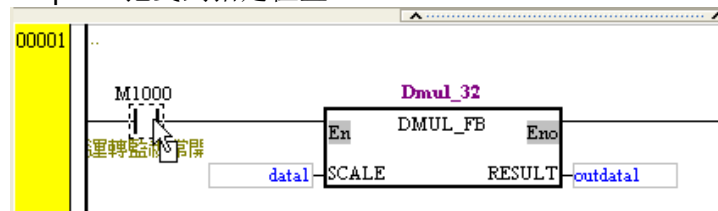
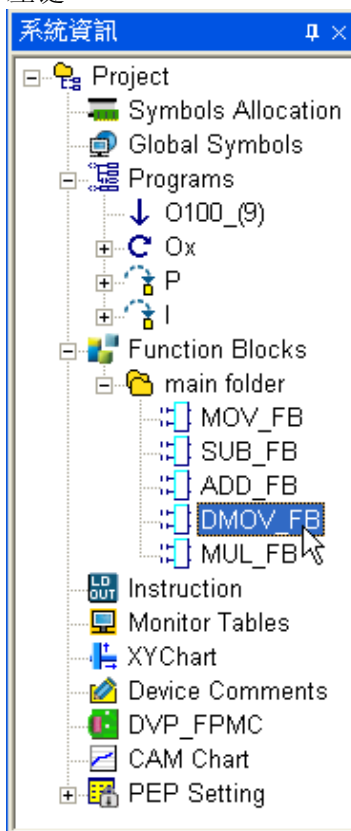
Step 3: 结果



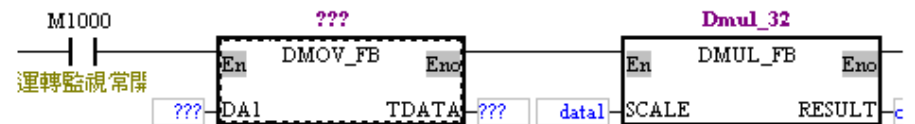
方法三：拖曳功能块，再来输入功能块名称。

Step 1: 选择功能块按鼠标左键

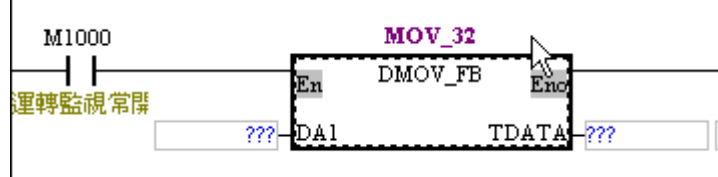
Step 2: 拖曳到指定位置



Step 3: 点击???再输入符号名称 MOV_32



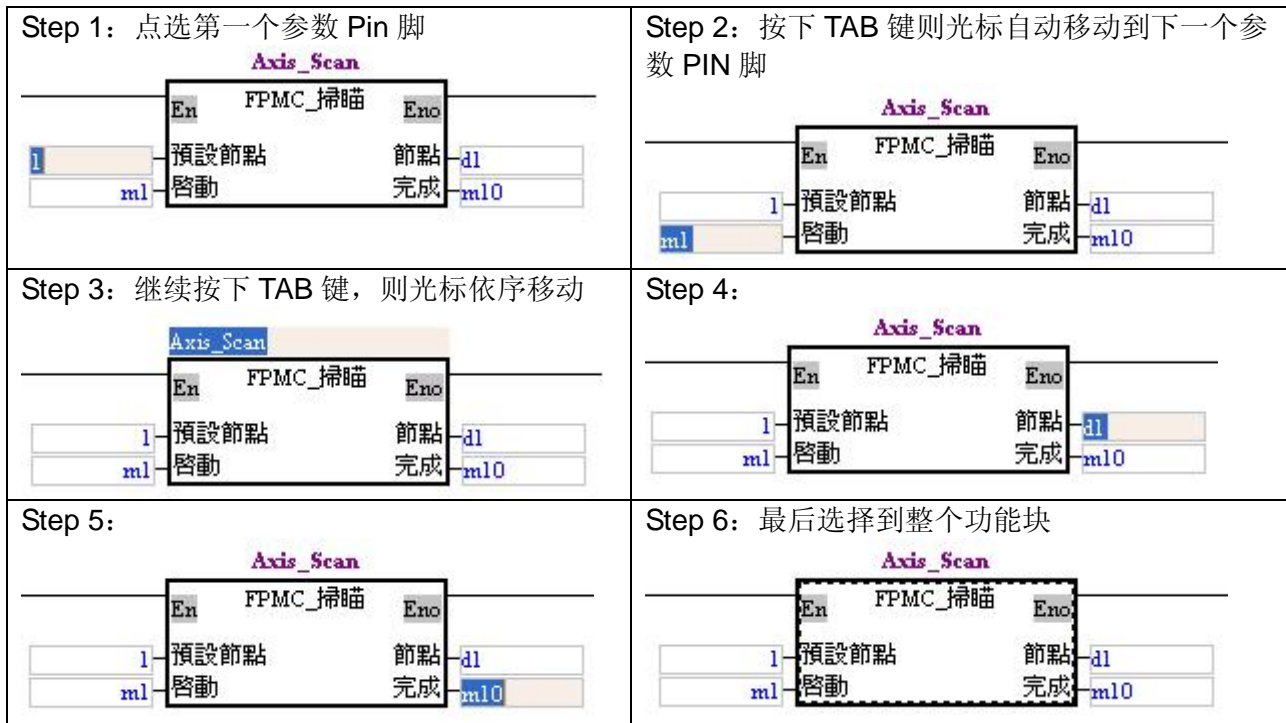
Step 4: 结果



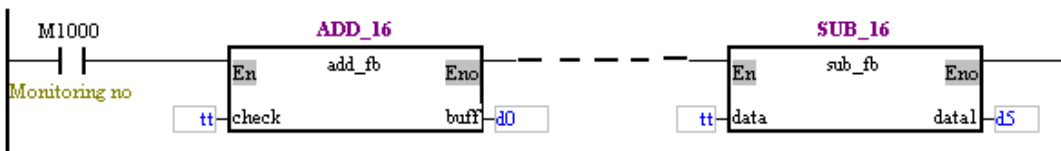
12 POU 阶梯图编辑模式

步骤三：编辑功能块的 Pin 脚

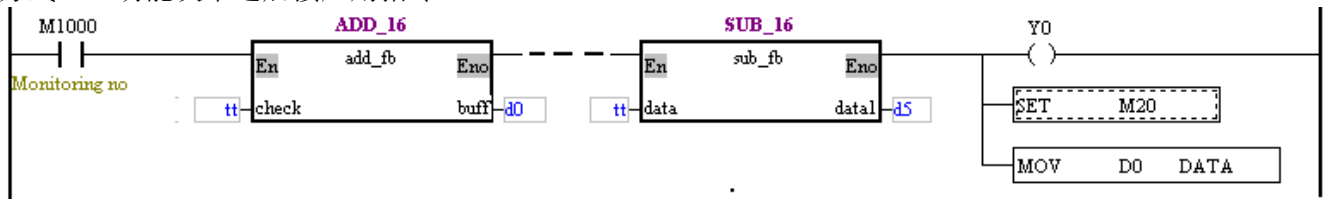
用鼠标直接点选功能块的编辑区，或是将游标点选到阶梯图中的功能块，按下 **TAB** 键，此时光标便会进入功能块可编辑的区域，输入完成后可以按下 **ENTER** 键，此时光标会依照功能块输入 **PIN** 脚，实体名称和输出 **Pin** 脚的顺序开始移动，方便继续对功能块做输入的动作。



- 阶梯图差异：功能块在阶梯图上是属于组件，但其在阶梯图只有以下两种方式
方式一：功能块串连



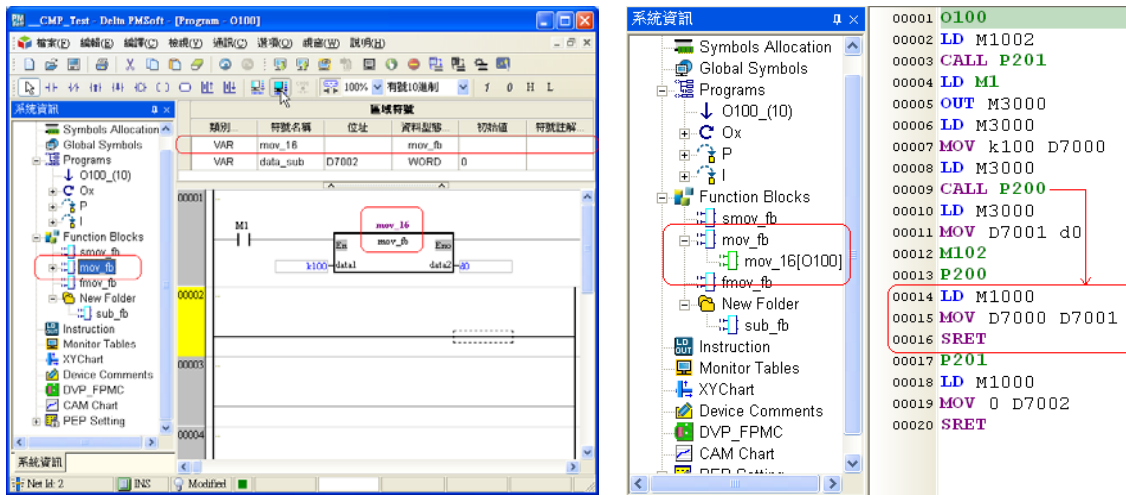
- 方式二：功能块串连后接应用指令



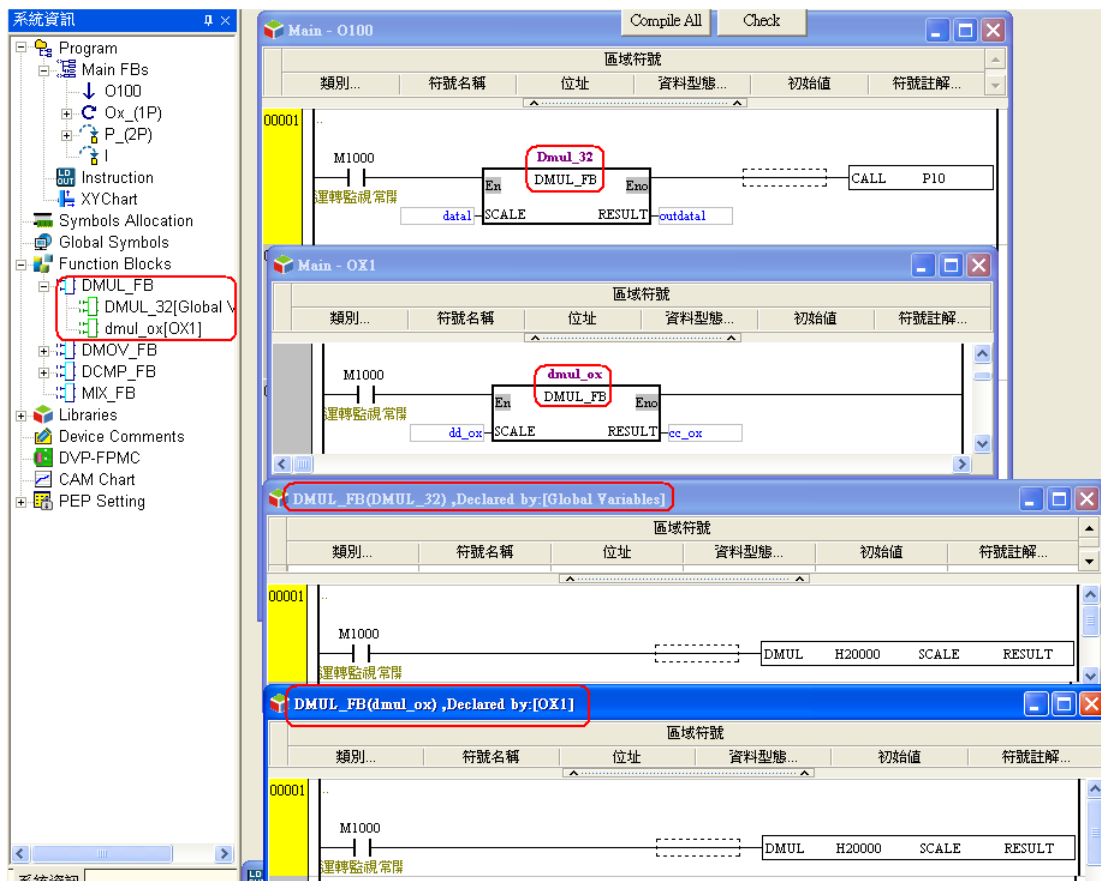
- 功能块是一种 Pn 子程序，经过编译之后转成 IL 程序。如下图所示将功能块转成 P200 子程序。



Step 1: 按编译

Step 2: 结果

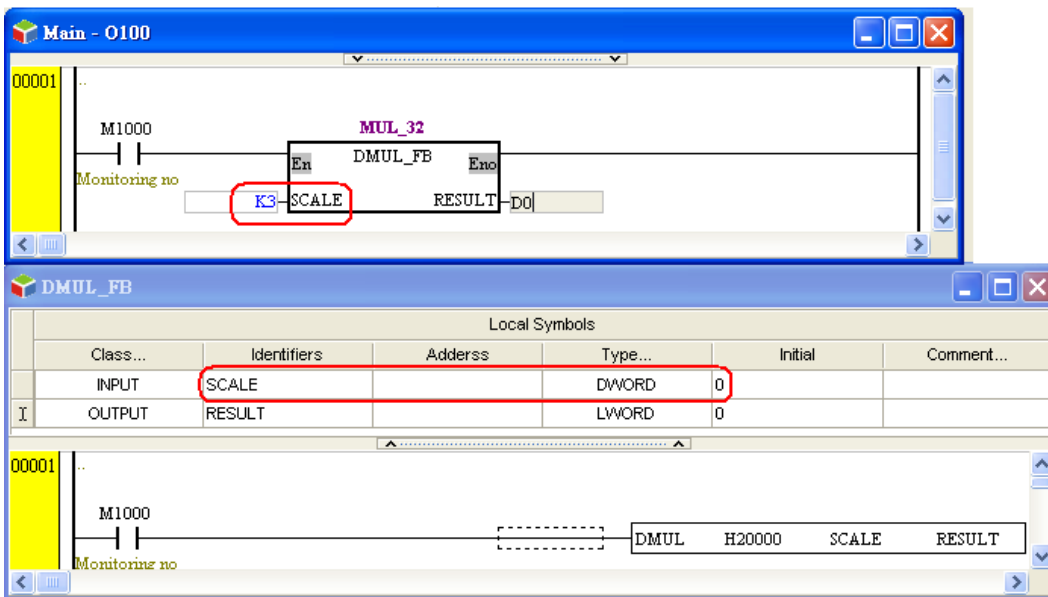


- 功能块可重复使用，使用同一个功能块型态，但使用不同符号，会转成不同 P 指针子程序，而占用 P 指针，并且在编译后转出不同之实体阶梯图(instance)。宣告使用功能块及编写使用 P 子程序转成 IL 不可超过 256 个子程序。如下图所示，系统信息区中可看到使用 2 个子程序，而在 O100 主程序中及 OX1 运动程序中皆有使用功能块 DMUL_FB，但符号名称分别在全域符号表及 OX1 区域符号表被宣告，因此占用 2 个 P 指标，并且在编译后转出不同之实体阶梯图。



- 使用功能块的输入输出型态要互相配合，否则按  检查或  编译，会发生错误；但功能块的输入接脚使用 WORD 或 DWORD 型态之符号时，功能块之输入数据可指定 K 或 H 常数，如下图所示：

12 POU 阶梯图编辑模式

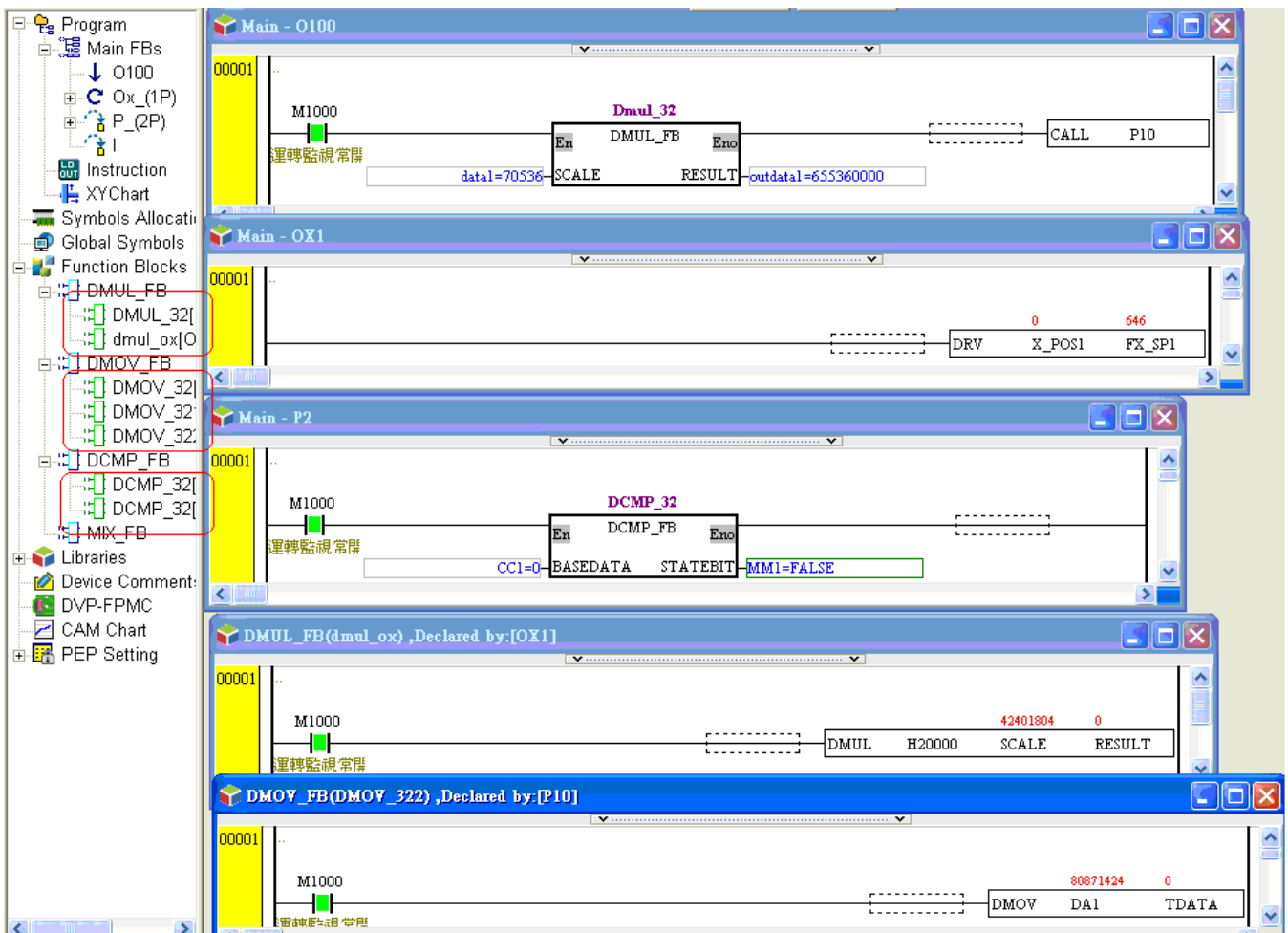


12.3 POU 在线监控

POU 程序运作可于阶梯图监控状态中直接于程序或监控表中修改数据，以下将详细说明。

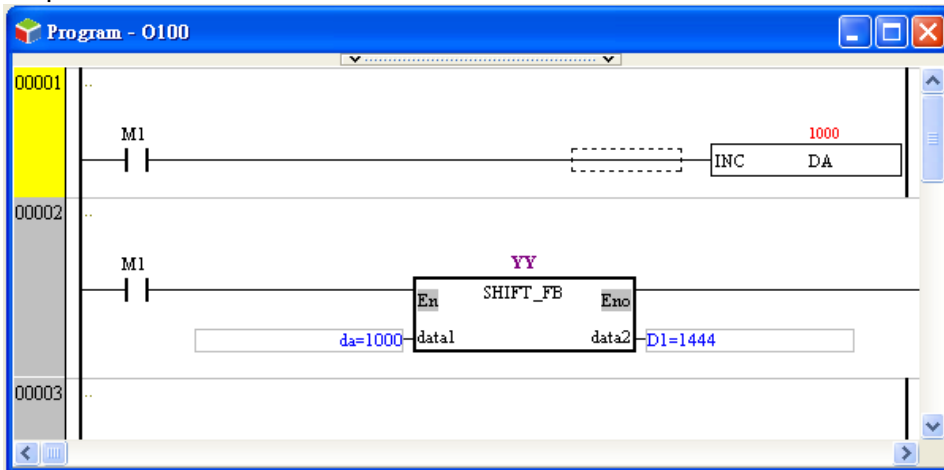
- 阶梯图上监控：点选 即可在线监控 O100、P 子程序、OX 运动程序及功能块编译后转出之实体阶梯图

如下图所示，红色圈选部分为监控 O100 运动程序及功能块实体程序。

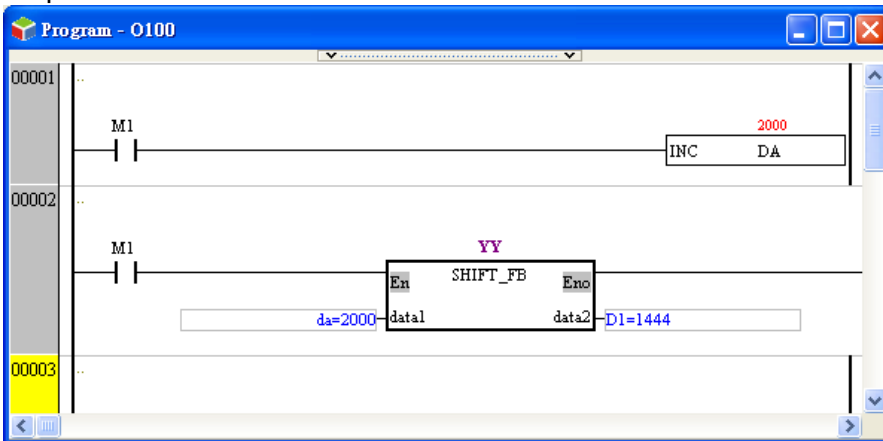


- 监控修改数据：如下图所示，O100 主程序中将变量 DA 值为 1000 改为 2000

Step 1: 点选功能块中输入变量 DA，然后输入 2000



Step 2: 变量 DA 显示 2000



- 监控表监控符号：在装置编号字段输入符号表名称(O100, OXn, Pn, Global) + "." + 符号名称。如下图所示在监控表 10 中要监控 O100 主程序中 TIMES 符号、OX2 运动程序中 X_POSIT 符号、P5 子程序中 STOR 符号及全域符号表中 BUFFER。

装置	数值型態	值	註解
O100.TIMES	d16s	1	
OX2.X_POSIT	d32s	42336268	
P5.STOR	d16s	6000	
GLOBAL.BUFFER	d16s	5000	

- 监控表的符号快速输入窗体：在监控窗体上点鼠标右键选择「符号输入...」接着会显示「符号选取」窗口。在「符号选取」窗口中会列出项目中宣告的所有符号，点选左边字段的来源项目，则右边字段将列出所有与该项目相关的符号。选择适当的变量，并按「确定」将所有符号加至数值监控表中。

12 POU 阶梯图编辑模式

Step 1: 在监控窗体上点鼠标右键选择「符号输入...」



Step 2: 点选欲监控的 O100、P 子程序、OX 运动程序或功能块的实体程序，在符号栏中显示此功能块实体程序的所有变量。



Step 3: 选择欲监控的变量，可使用勾选或是点选全部选取来选择变量，选择好后点选确定。



Step 4: 监控窗体中会自动产生监控符号

裝置	數值型態	值	註解
O100.AXIS_SCAN	bit		0 => 關閉POU, 1 => 起動POU
O100.AXIS_SCAN	bit		工作中
O100.AXIS_SCAN	d16s		
O100.AXIS_SCAN	d16s		

12.4 符号或功能块自动显示批注(HINT)

批注功能可方便使用者了解功能块的功能、功能块的接脚定义及符号型态。只要在符号表批注区(图(A))和功能块批注区(图(B))编写说明，程序中使用这些符号和功能块时，鼠标光标移到感应区域，将会自动显示符号、接脚或是功能块信息。

(A)符号表批注

類別...	符號名稱	位址	資料型態	初始值	符號註解...
INPUT	預設節點		WORD	0	節點編號，可接受數值範圍 = 1
INPUT	啟動		BOOL	FALSE	0 => 關閉POU, 1 => 起動POU
OUTPUT	節點		WORD	0	工作的節點編號
OUTPUT	完成		BOOL	FALSE	工作完畢

(B)功能块批注

建立功能塊



POU 名稱:

使用 En/Eno

密碼設定: 輸入密碼 <4-8 字元>
密碼確認 <4-8 字元>

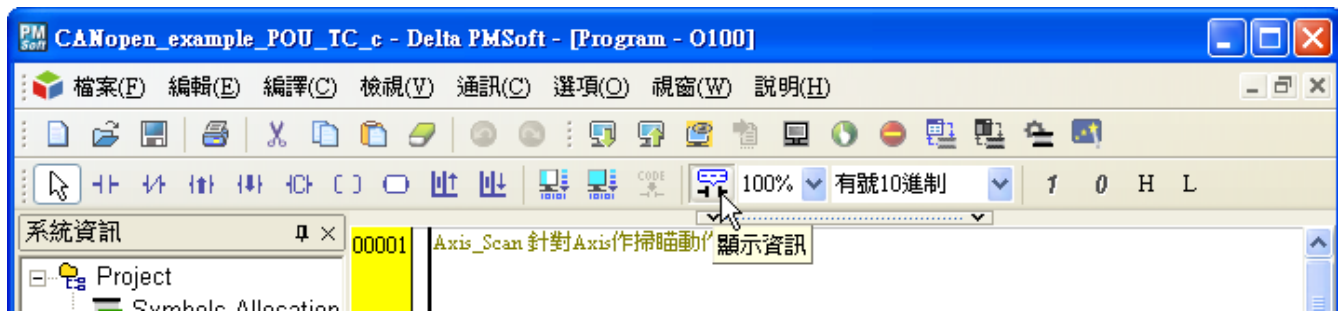
語言: 指令列表 (IL) 階梯圖 (LD) 功能區塊圖 (FBD) 循序功能圖 (SFC)

POU輸入
 預設節點: 節點編號，可接受數值範圍 = 1 ~ 4
 啟動: 0 => 關閉POU, 1 => 起動POU
 POU輸出
 節點: 工作的節點編號
 完成: 工作完畢

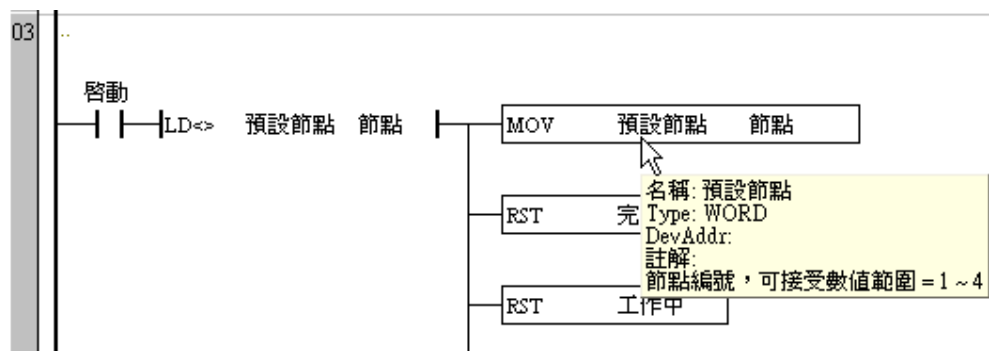
- 批注显示: 开启批注显示功能可利用下图(A)所示按钮,  表示批注功能已开启,  表示批注功能已关闭, 下图 (B)(C)(D)所示, 分别为鼠标光标移至符号、功能块接脚和功能块名称时显示之批注。

12 POU 阶梯图编辑模式

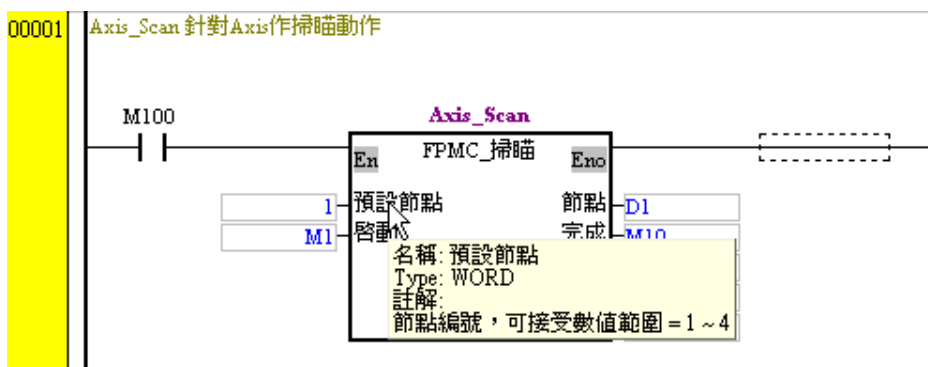
(A) 批注显示致能按钮



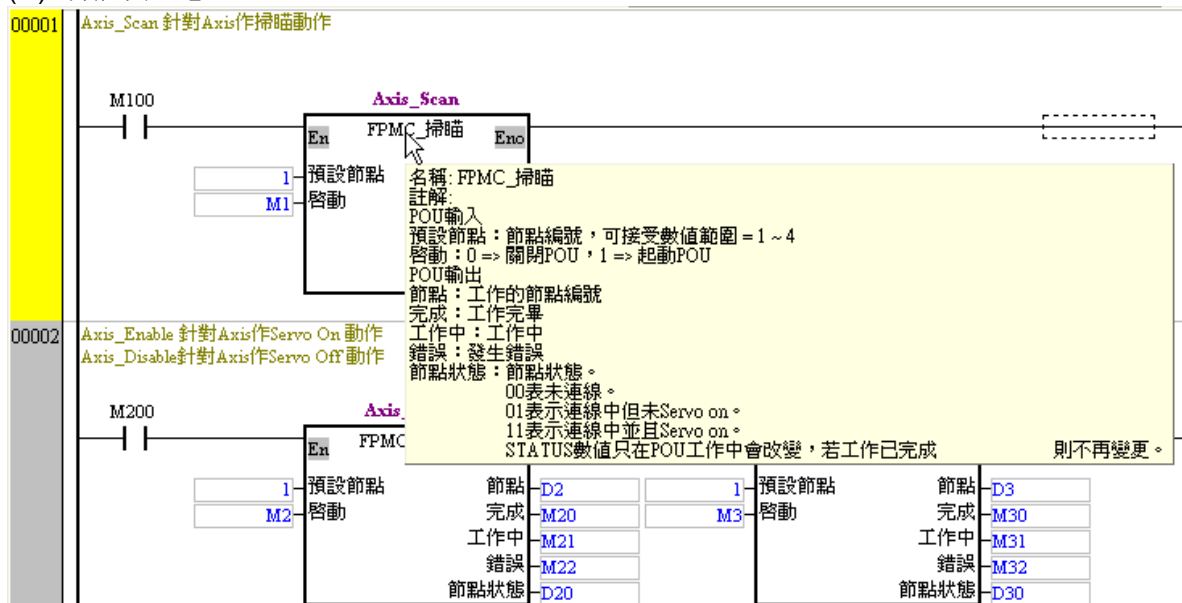
(B) 符号批注显示



(C) 功能块接脚讯息



(D) 功能块讯息



13.1 CANopen 通讯功能卡 (DVP-FPMC) 简介

DVP-FPMC 是运行于 DVP-PM 系列主机的 CANopen 通讯功能卡, 提供主机做数据交换。DVP-FPMC CANopen 通讯功能卡, 具备功能如下:

- ◆ 符合 CANopen 标准协议 DS301v4.02
- ◆ 支持 NMT 服务
- ◆ 支持 SDO 服务
- ◆ 支持 CANopen 标准协议 DS402v2.0: 最多支持四轴运动轴
- ◆ 运动轴支持 Profile Position mode

13.2 功能规格

● CANopen 连接器

项目	规格
传输方式	CAN
电气隔离	500VDC
接头	可插拔式连接器 (5.08mm)
传输电缆	两条通讯线、一条屏蔽线和一条接地线

● Ethernet 连接器

项目	规格
传输方式	Ethernet
电气隔离	500VDC
接头	可插拔式连接器 (5.08mm)
传输电缆	两条通讯线、一条屏蔽线和一条接地线

● 通讯

项目	规格
信息类型	PDO、SDO、SYNC (同步对象)、Emergency (紧急对象)、NMT
串行传输速度	支持 500k、1M bps (位/秒)
产品代码	254
设备类型	0 (Non-Profile)
厂商 ID	477 (台达电子)

13 CANopen 通讯功能卡

● 电气规格

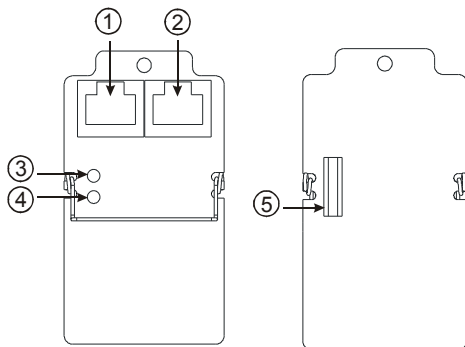
项目	规格
电源电压	由主机经由内部总线供应 24VDC (-15% ~ 20%)
消耗电力	1.7 W
绝缘电压	500 V
重量 (约,g)	66 (g)

● 环境规格

项目	规格
噪声免疫力	ESD (IEC 61131-2, IEC 61000-4-2): 8KV Air Discharge, 4kV Contact Discharge EFT (IEC 61131-2, IEC 61000-4-4): Power Line: 2KV, Digital I/O: 1kV Analog & Communication I/O: 1kV Damped-Oscillatory Wave: Power Line: 1KV, Digital I/O: 1kV RS (IEC 61131-2, IEC 61000-4-3): 80MHz ~ 1000MHz , 1.4GHz ~ 2.0GHz , 10V/m
操作/储存环境	操作: 0°C ~ 55°C (温度)、50 ~ 95% (湿度)、污染等级 2 储存: -25°C ~ 70°C (温度)、5 ~ 95% (湿度)
耐震动/冲击	国际标准规范 IEC 61131-2、IEC 68-2-6 (TEST Fc)/IEC 61131-2 & IEC 68-2-27 (TEST Ea)
标准	IEC 61131-2

13.3 产品外观及安装

产品外观:



① CANopen 通讯连接器

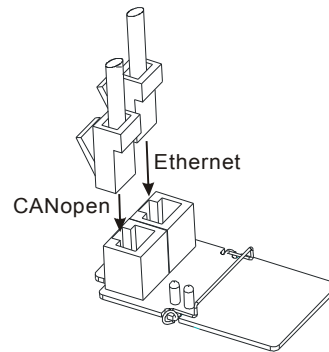
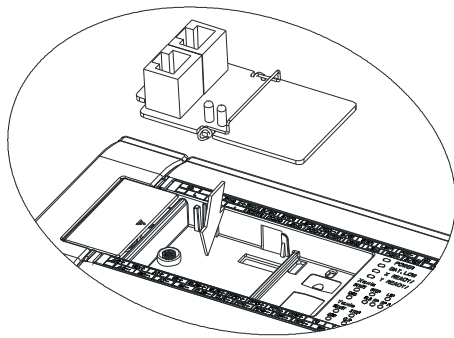
② Ethernet 通讯连接器

③ CANopen 指示灯

④ Ethernet 指示灯

⑤ 通讯连接器

安装 DVPFPMC 卡在 PM 主机上与通讯线路连接:



13.4 CR 参数功能介绍

- 共同参数区

CR# 编号	寄存器名称	保持型	资料型态	寄存器深度
#001	FPMC 韧体版本	R	Word	1
#052	FPMC 同步功能设定	R/W	Word	1
#053	CANopen 网络节点 ID 设定	R/W	Word	1
#054	Canopen 传输速率设定	R/W	Word	1
#055	SDO / NMT Timeout 等待时间	R/W	Word	1
#056	FPMC 错误状态	R/W	Word	1
#059	FPMC IP 与 Port 设定	R/W	Word	3
#062	Ethernet 联机命令	R/W	Word	1
#063	Ethernet 主站 IP 和 Port 设定	R/W	Word	3
#064	Ethernet 向主站传送数据长度	R/W	Word	1
#065	Ethernet 向主站传送数据内容	R/W	Word	512
#066	Ethernet 向主站接收数据长度	R	Word	1
#067	Ethernet 向主站接收数据内容	R	Word	512
#070	SDO 伺服节点编号	R/W	Word	1
#071	SDO 写入/读取之控制/显示	R/W	Word	1
#072	SDO 对象辞典索引	R/W	Word	1
#073	SDO 对象辞典传送/接收之数据寄存器 1	R/W	Word	512
#074	SDO 对象辞典传送/接收之数据寄存器 2	R/W	Word	512
#075	SDO 对象辞典传送/接收之数据寄存器 3	R/W	Word	512
#076	SDO 对象辞典传送/接收之数据寄存器 4	R/W	Word	512
#080	NMT 动作命令	R/W	Word	1

13 CANopen 通讯功能卡

● A2 模式之四轴参数区

CR# 编号	寄存器名称	保持型	资料型态	寄存器深度
#010	Canopen 扫描	R/W	Word	1
#020	Canopen 通讯状态	R	Word	1
#040	伺服错误状态	R	Word	1
#050	针对所有已联机伺服之控制命令	R/W	Word	1
#090	FPMC Buffer 资料	R/W	Word	32
#091	FPMC Buffer 数据位置	R/W	Word	32
#092	FPMC Buffer 资料	R/W	Word	32
#093	FPMC Buffer 数据位置	R/W	Word	32
#n00	节点编号	R	Word	1
#n01	厂商代码	R	Word	1
#n02				
#n03	产品代码	R	Word	1
#n04				
#n05	韧体版本	R	Word	1
#n06				
#n07	厂商产品类别	R	Word	1
#n08				
#n09	CANopen 网络通讯状态	R	Word	1
#n10	紧急错误码	R	Word	1
#n11	厂商错误代码	R	Word	1
#n12				
#n20	伺服状态	R	Word	1
#n21	伺服运动状态	R	Word	1
#n22	伺服之现在位置	R	Word	1
#n23				
#n40	节点所联机伺服之控制命令	R/W	Word	1
#n50	节点之 SDO 控制	R/W	Word	1
#n51	节点之 SDO 对象辞典索引	R/W	Word	1
#n52	节点之 SDO 发送/接收数据寄存器 1	R/W	Word	512
#n53	节点之 SDO 发送/接收数据寄存器 2	R/W	Word	512
#n54	节点之 SDO 发送/接收数据寄存器 3	R/W	Word	512
#n55	节点之 SDO 发送/接收数据寄存器 4	R/W	Word	512
#n60	节点之伺服控制	R/W	Word	1
#n61	节点之模式控制	R/W	Word	1
#n70	位置模式目标位置	R/W	Word	1
#n71				
#n72	位置模式运转速度	R/W	Word	1
#n73				

CR# 编号	寄存器名称	保持型	资料 型态	寄存器 深度
#n74	位置模式运转加速时间(ms)	R/W	Word	1
#n75				
#n76	位置模式运转减速时间(ms)	R/W	Word	1
#n77				
#n78	位置模式控制	R/W	Word	1
#n80	回归原点方式	R/W	Word	1
#n81	离原点偏移量	R/W	Word	1
#n82				
#n83	回原点速度	R/W	Word	1
#n84				
#n85	回原点碰 DOG 之后的速度	R/W	Word	1
#n86				
#n87	回原点加速时间	R/W	Word	1
#n88				
#n89	回原点控制	R/W	Word	1
#n90	补间模式目标位置	R/W	Word	1
#n91				
#n92	补间模式启动	R/W	Word	1

● Canopen 一般模式

CR# 编号	寄存器名称	保持型	资料 型态	寄存器 深度
#500	CANopen 模式切换	R/W	Word	1
#504	从站 Heartbeat 命令	R/W	Word	1
#505	从站 Heartbeat 命令执行状态	R	Word	1
#506	从站 Heartbeat 状态	R	Word	1

● 对象辞典相关参数区

寄存器代码	Data	R/W/N	Type	深度
#1006h	同步周期设定	R/W	DWord	1
#1017h	FPMC heartbeat 周期设定	R/W	Word	1
#1400h~#143Fh	接收模式 PDO 参数设定	R/W	Word	3
#1600h~#163Fh	接收模式 PDO 数据映像参数设定	R/W	DWord	4
#1800h~#183Fh	传送模式 PDO 参数设定	R/W	Word	3
#1A00h~#1A3Fh	传送模式 PDO 数据映像参数设定	R/W	Dword	4
#2000h~#207Fh	PDO 数据寄存器	R/W	Word	4

13 CANopen 通讯功能卡

13.5 控制寄存器 CR 内容说明

- 共同参数区

CR#001: FPMC 韧体版本

[说明]

本机之韧体版本，以 16 进制显示，例如：H'8161，表示 8 月 16 日下午韧体发行日期。

CR#052: FPMC 同步功能设定

[说明]

此寄存器设定两种功能：

- Low Byte 设定 FPMC 启动 Canopen 网络之同步功能，当此值设定为 1 时，便启动 FPMC 发送同步封包至 Canopen 网络；设定为 0 时，便关闭同步功能。
- High Byte 设定 FPMC 与 PM 之同步周期，设定值*5=PM 之同步周期设定寄存器(D1040)。当该值大于 0 时，便启动 FPMC 与 PM 之间的同步。

Bit	Bit[15:8]	Bit[7:0]
内容值	FPMC 与 PM 之同步周期数设定	设定 FPMC 启动 Canopen 之同步功能

CR#053: CANopen 节点 ID 设定

[说明]

设定 CANopen 之节点 ID，设定范围 5~127，初始值为 127。

CR#054: CANopen 传输速率设定

[说明]

设定 CANopen 网络之传输速率；当设定完毕后，Bit[15]显示网络传输速率之设定状态，设定中显示为 1，设定完毕后显示为 0。举例说明：当设定 CANopen 网络传输速率为 1000kb/s 时，仅需将 CR#054 填入值 K1000 即可。

Bit	Bit[15]	Bit[14:0]
设定值	设定状态显示 0: 完成 1: 设定中	1000: CANopen 速率=1000kb/s 500: CANopen 速率=500kb/s

CR#055: SDO / NMT Timeout 等待时间

[说明]

设定 CANopen 网络 SDO / NMT Timeout 之等待时间，单位为 ms，初始值为 1000。

CR#056: FPMC 错误状态

[说明]

显示 FPMC 之错误状态，错误状态值请参考以下说明：

错误状态	内容值	解决方式
CANopen 连接错误	C1	确认目前 CANopen 存在从站节点
Ethernet 连接错误	E1	确认通讯模块与 Ethernet 连接正常

CR#059: FPMC IP 与 Port 设定

[说明]

设定 FPMC 之 IP 与 Port，寄存器深度为 3 个 Word，初始值 IP 地址为 192.168.0.100，Port 为 1024。

设定之格式请参考以下范例：IP:192.168.0.100 Port:1024

Word 0		Word 1		Word 2
H-byte	L-byte	H-byte	L-byte	1024
192	168	0	100	

CR#062: Ethernet 联机命令/状态显示

[说明]

内容值为设定 FPMC 之 Ethernet 网络联机命令与 Ethernet 网络联机状态显示。

- H'0: 显示 Ethernet 之网络状态为未联机
- H'30: 显示 Ethernet 之网络状态为已联机
- H'10: 发送 Ethernet 网络联机命令
- H'20: 发送 Ethernet 网络断线命令

CR#063: Ethernet 主站 IP 和 Port 设定

[说明]

设定欲联机对象之 Ethernet 网络 IP 与 Port，深度为 3 个 Words。设定格式请参考下表：

Word 0		Word 1		Word 2
H-byte	L-byte	H-byte	L-byte	1024
192	168	0	100	

CR#064~CR#67: Ethernet 传送/接收数据长度与内容

[说明]

FPMC 向联机对象发送 Ethernet 数据与长度设定之寄存器，最大容量为 1024Byte。

- 传送数据时，使用者将数据写入 CR#065、将数据长度写入 CR#064 之后，PMC 会自动将 CR#064 与 CR#065 内容清除为 0。

13 CANopen 通讯功能卡

- 接收数据时，使用者先读取 CR#066 内容，判断 PMC 收到的数据长度后，再读回 CR#067 相同长度的数据内容。

CR#070: SDO 伺服节点编号

[说明]

设定 CanOpen 网络上欲传送对象之 SDO 节点编号。设定范围为 1~127。

CR#071: SDO 写入/读取之控制/显示

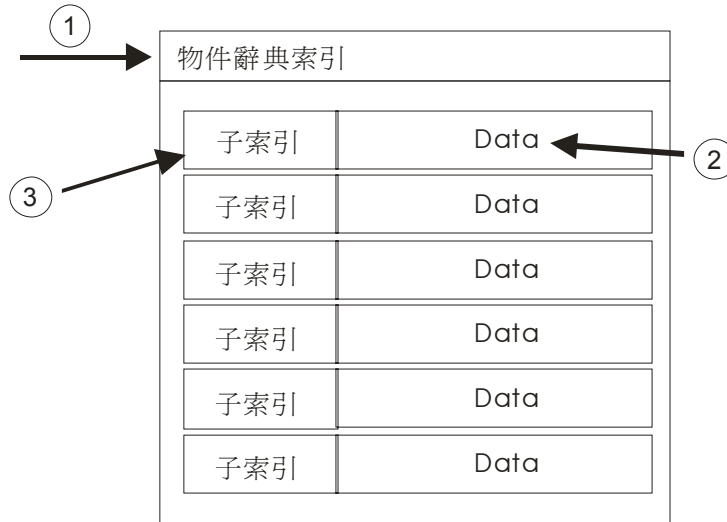
[说明]

控制 SDO 的上/下载之动作，并显示结果，设定格式请参考下表：

Bit	Bit[15:8]	Bit[7:4]	Bit[3]	Bit[2:0]
设定值	欲传送对象的对象辞典索引下的子索引	数据长度(单位 byte) 数值范围为 1~8 写入时需填入写入的数据长度	错误 旗标	命令 0:完成 1:写入(含检查) 2:读取(含检查) 3:写入(不含检查) 4:读取(不含检查)

举例说明：SDO 传送数据方式示意图如下所示：

- 指定欲传送对象(CR#70)之对象辞典索引 CR#072。
- 将欲发送之数据写入 CR#073~CR#076 数据寄存器。
- 参考上表，设定 CR#071 之 Bit[15:8]指定子索引以及设定写入/读取。



CR#072: SDO 对象辞典索引

[说明]

欲传送对象的对象辞典索引，设定范围为 H'0000~H'FFFF。

CR#073~CR#76: SDO 对象辞典传送/接收之数据寄存器 1~4

[说明]

存放 SDO 传输中待传送或已接收的数据寄存器，最大容量为 1024Bytes。当 SDO 传输发生错误时，错误代码会存放在 CR#073 与 CR#074。当一次使用 CR#073~CR#076 时，CR#073 为低位(LSB)，CR#076 为高位(MSB)。

CR#080: NMT 动作命令

[说明]

当 FPMC 为主站时，对从站作 Canopen 网络通讯状态变更命令。其命令与设定格式请参考下表：

Bit	Bit[15:8]	Bit[7:0]
设定值	网络管理命令 1: 启动节点通讯 2: 关闭节点通讯 128: 切换操作模式 129: 重置节点通讯	从站节点编号

● A2 模式之参数区

A2 模式为针对连接台达伺服 ASDA-A2 机种，规划 Canopen 节点 ID1~4 做为伺服用之节点，以及规划 CR#100~CR#499 对应伺服参数设定之控制寄存器，其中 CR#100~CR#199 为伺服节点=1 之功能寄存器，CR#200~CR#299 为伺服节点=2 之功能寄存器，CR#300~CR#399 为伺服节点=3 之功能寄存器，CR#400~CR#499 为伺服节点=4 之功能寄存器。在寄存器名称说明当中，n 为特殊寄存器之百位数，n=1~4。A2 模式之功能寄存器仅能在 A2 模式下使用。

CR#010: Canopen 扫描

[说明]

执行扫描 Canopen 网络之节点 1~4；当该 Bit 设定为 1 时，会执行扫描对应之节点，扫描成功后此寄存器之内容会自动清除为 0。CR 与相对应的节点编号请参考下表：

Bit	Bit[15:4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
Node 编号	保留	Node 4	Node 3	Node 2	Node 1

CR#020: CAN bus 通讯状态

[说明]

内容显示该节点之通讯状态，以两个 Bit 表示：00=未连接，01=已连接，11=可运作。CR 之 Bit 与相对应的节点编号请参考下表：

13 CANopen 通讯功能卡

Bit	Bit[15:8]	Bit[7:6]	Bit[5:4]	Bit[3:2]	Bit[1:0]
Node 编号	保留	Node 4	Node 3	Node 2	Node 1

CR#040: 伺服错误状态

[说明]

内容值显示伺服发生错误，该节点之伺服发生错误时，对应的 **Bit=1**；当下达错误清除命令成功时，此寄存器之内容值会清除为 0。CR 与相对应的节点编号请参考下表：

Bit	Bit[15:4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
Node 编号	保留	Node 4	Node 3	Node 2	Node 1

CR#050: 伺服联机控制命令

[说明]

对 Canopen 网络上所有联机成功的节点所连接之伺服发送控制命令，设定为 1 表示开启已联机伺服，设定为 128 表示关闭已联机伺服，设定为 129 表示清除所有错误状态，当设定完成后该内容值会自动清除为 0。CR 之设定格式请参考下表：

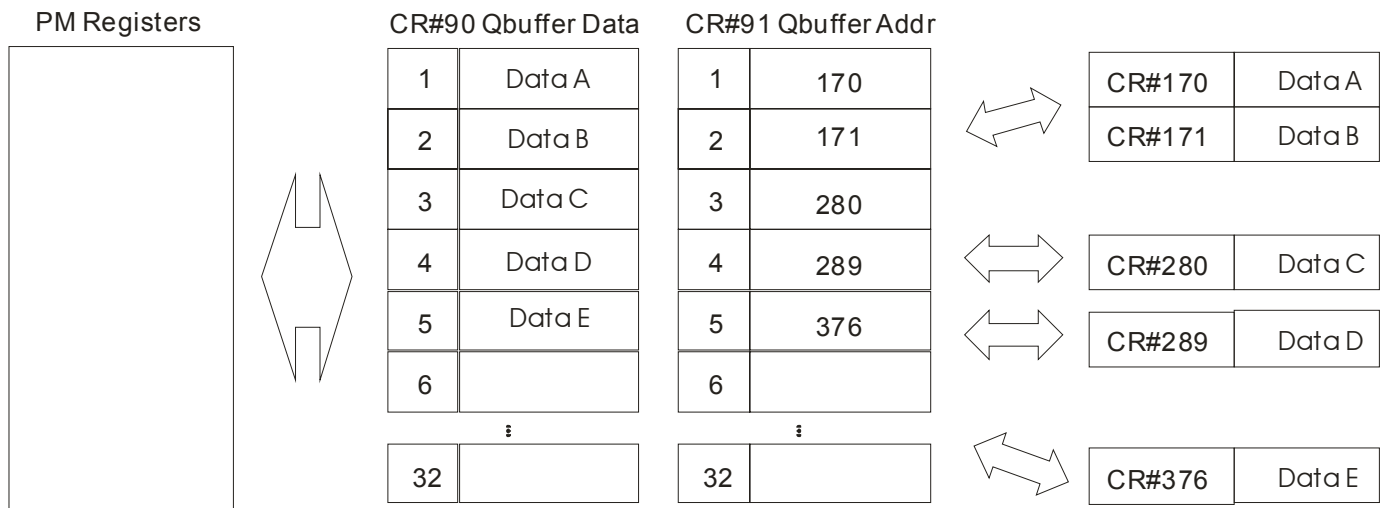
Bit	Bit[15:8]	Bit[7:0]
内容值	保留	ALL Servo On: 1 ALL Servo off: 128 Error Reset: 129

CR#090~CR#93: FPMC QBuffer 数据/数据位置

[说明]

Qbuffer 针对 CR100~CR499 之寄存器，可将多笔不连续的功能寄存器 CR，使用一次 TO/FROM 指令进行一次传输/读取动作。其优点为减少执行多次 TO/FROM 动作，缩短扫描时间。Qbuffer 的动作示意图如下所示：

举例说明，各别将 CR#170、171 以及 280、289 与 376 不连续之特殊寄存器编号写入 CR#91 FPMC QBuffer 数据位置寄存器中，FPMC 会自动将各编号寄存器之内容依序对应到 CR#90 FPMC QBuffer 数据寄存器中，使用者仅需读写 CR#90，即可一次修改 QBuffer 中所有对应的特殊寄存器内容。



CR#n00: 节点编号

[说明]

显示伺服中设定 Canopen 网络之节点编号。

- 当伺服节点 ID=1: CR#100=1
- 当伺服节点 ID=2: CR#200=2
- 当伺服节点 ID=3: CR#300=3
- 当伺服节点 ID=4: CR#400=4

CR#n01~CR#n02: 厂商代码

[说明]

显示 ASDA-A2 伺服之生产厂商代码，数据型态为 Dword。

CR#n03~CR#n04: 产品代码

[说明]

显示 ASDA-A2 伺服之产品代码，数据型态为 Dword。

CR#n05~CR#n06: 韧体版本

[说明]

显示 ASDA-A2 伺服之韧体版本，数据型态为 Dword。

CR#n07~CR#n08: 厂商产品类别

[说明]

显示 ASDA-A2 伺服生产厂商之产品类别代码，数据型态为 Dword。

13 CANopen 通讯功能卡

CR#n09: CANopen 网络通讯状态

[说明]

显示该节点目前 Canopen 网络之通讯状态，内容值与相关意义请参考下表。

状态	显示值
断线	H'1
已联机	H'2
操作模式	H'5
发生错误	H'6
节点重置中	H'7

CR#n10: 紧急错误码

[说明]

显示 Canopen 协议规范中，当网络某节点运作发生错误时所产生的错误代码。

CR#n11~CR#n12: 厂商错误代码

[说明]

显示厂商定义当 ASDA-A2 伺服发生错误时，产生的错误代码。其错误码请参考台达 ASDA-A2 伺服之应用手册。

CR#n20: 伺服状态

[说明]

内容值为 ASDA-A2 伺服目前状态，该状态值以及代表的意义请参考下表：

Status word															
X	OM	OM	OM	X	TR	RM	X	WR	X	QS	X	FT	SO	X	RS
Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- RS: 伺服待机，当伺服初始化完成后，该旗标会设定为 1，表示可以启动。
- SO: 伺服已启动，当伺服启动后，该旗标会设定为 1。
- FT: 错误旗标，当伺服发生错误时，该旗标会设定为 1。
- QS: 该旗标为 1 时，可执行急停命令。
- WR: 警告旗标，当伺服出现警告时，该旗标会设定为 1。
- RM: 该旗标为 1 时，可执行远程监控。
- TR: 运动命令完成时，该旗标会设定为 1。
- OM[14:12]: 用于显示各项运动模式之状态，如下表所示。

	位置模式	回原点模式	补间模式
OM[12]	已成功设定目标位置	回原点模式执行中	补间模式执行中
OM[13]	马达发生追随错误	回原点模式发生错误	X
OM[14]	X	X	同步启动

CR#n21: 伺服运动状态

[说明]

显示伺服目前执行之运动模式。

内容值	运动模式
0x01	位置模式
0x06	回原点模式
0x07	补间模式

CR#n22~CR#n23: 伺服之现在位置

[说明]

显示伺服现在位置，数据类型为 Dword。

CR#n40: 节点所联机伺服之控制命令

[说明]

针对该节点的伺服发送控制命令，设定为 1 表示开启已联机伺服，设定为 128 表示关闭已联机伺服，设定为 129 表示清除错误状态。CR 之设定格式请参考下表：

Bit	Bit[15:8]	Bit[7:0]
内容值	保留	Servo On: 1 Servo off: 128 Error Reset: 129

CR#n50: 节点之 SDO 控制

[说明]

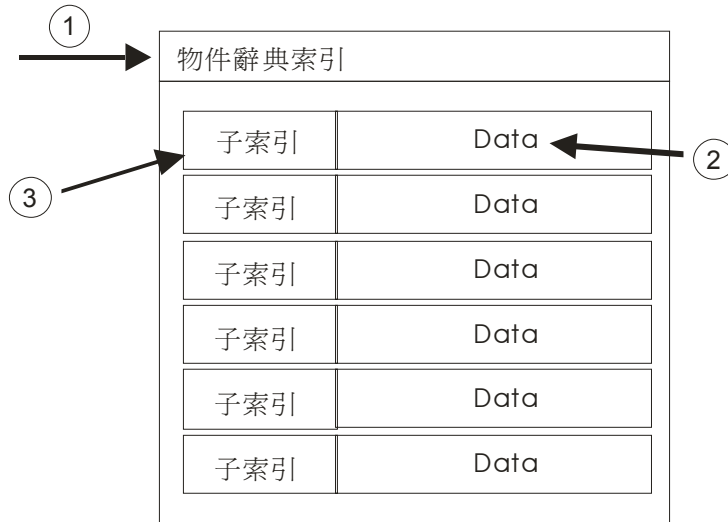
控制该节点之 SDO 的上/下载之动作，并显示结果，设定格式请参考下表：

Bit	Bit[15:8]	Bit[7:4]	Bit[3]	Bit[2:0]
设定值	欲传送对象的对象辞典索引下的子索引	数据长度(单位 byte) 数值范围为 1~8 写入时需填入写入的数据长度	错误 旗标	命令 0:完成 1:写入(含检查) 2:读取(含检查) 3:写入(不含检查) 4:读取(不含检查)

13 CANopen 通讯功能卡

举例说明：SDO 传送数据方式示意图如下所示：

1. 指定该节点之对象辞典索引 CR#n51。
2. 将欲发送之数据写入 CR#n52~CR#n55 数据寄存器。
3. 参考上表，设定 CR#n50 之 Bit[15:8]指定子索引以及设定写入/读取。



CR#n51: 节点之 SDO 对象辞典索引

[说明]

设定该节点的对象辞典索引，设定范围为 H'0000~H'FFFF。

CR#n52~CR#n55: 节点之 SDO 发送/接收数据寄存器 1~4

[说明]

存放 SDO 传输中待传送或已接收的数据寄存器，最大容量为 1024Bytes。当 SDO 传输发生错误时，错误代码会存放在 CR#n52 与 CR#n53。当一次使用 CR#n52~CR#n55 时，CR#n52 为低位(LSB)，CR#n55 为高位(MSB)。

CR#n60: 节点之伺服控制

[说明]

此寄存器用于控制该节点所连接之伺服，各位代表的控制意义请参考下表：

Control word															
X	X	X	X	X	X	X	X	FR	OM	OM	OM	EO	X	X	X
Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- EO: 伺服启动命令，此 bit 设定为 1 时该伺服将会启动
- FR: 伺服错误重置命令，此 bit 设定为 1 时该伺服将进行错误重置动作
- OM: 用于控制各项运动模式之功能，如下表所示。

	位置模式	回原点模式	补间模式
OM[4]	设定新的目标位置	启动回原点动作	X
OM[5]	允许运动中更改目标位置	X	X
OM[6]	绝对/相对定位	X	X

备注：X 表示保留。

CR#n61: 模式控制

[说明]

设定该节点伺服之运动模式，其设定值以及代表的意义请参考下表：

设定值	运动模式
0x01	位置模式
0x06	回原点模式
0x07	补间模式

CR#n70~CR#n71: 位置模式目标位置

[说明]

设定位置模式之目标位置，数据型态为 Dword。

CR#n72~CR#n73: 位置模式运转速度

[说明]

设定位置模式之运转速度，数据型态为 Dword。

CR#n74~CR#n75: 位置模式运转加速时间

[说明]

设定位置模式之运转加速时间，数据型态为 Dword。

CR#n76~CR#n77: 位置模式运转减速时间

[说明]

设定位置模式之运转减速时间，数据型态为 Dword。

CR#n78: 位置模式设定

[说明]

针对该节点之 ASDA-A2 伺服设定位置模式为绝对/相对定位，设定值与意义如下所示。

- 0: 表示位置模式执行完毕。

13 CANopen 通讯功能卡

- 1: 设定位置模式为绝对寻址, 当运动完成后此 CR 内容会清除为 0。
- 2: 设定位置模式为相对定位, 当运动完成后此 CR 内容会清除为 0。
- 3: 设定位置模式为绝对寻址, 当运动完成后此 CR 内容会保留设定。

CR#n80: 回原点方式

[说明]

设定回原点方式, 范围为 1~35。各种回原点方式详细说明请参考 CIA DSP 402 V2.0 之文件名称 Drives and Motio Control, Chapter 13 Homing Modes。

CR#n81~CR#n82: 离原点偏移量

[说明]

设定回原点后的偏移量, 设定范围-2,147,483,648~2,147,483,647, 数据型态为 Dword。

CR#n83~CR#n84: 回原点速度

[说明]

设定回到机械原点的速度, 设定范围 0~2,147,483,647, 数据型态为 Dword。

CR#n85~CR#n86: 回原点碰 DOG 之后的速度

[说明]

设定回原点模式中, 碰到 DOG 之后的运行速度, 设定范围 0~2,147,483,647, 数据型态为 Dword。

CR#n87~CR#n88: 回原点加速时间

[说明]

设定回原点模式中之加速时间, 设定范围 0~2,147,483,647, 数据型态为 Dword。

CR#n89: 回原点模式启动

[说明]

内容值设定为 1 时启动回原点模式, 当运动完成后, 内容自动清除为 0。

CR#n90~CR#n91: 补间模式目标位置

[说明]

设定补间模式之目标位置, 设定范围-2,147,483,648~2,147,483,647, 数据型态为 Dword。

CR#n92: 补间模式启动

[说明]

内容值设定为 1 时启动回补间模式；内容值设定为 0 时关闭补间模式。

- Canopen 一般模式

CR#500: CANopen 模式切换

[说明]

FPMC 卡之工作模式切换；设定为 1 为 A2 模式，设定为 2 为一般模式。初始值为 1。当模式设定为一般模式时，则 A2 模式之功能寄存器将无法使用，反之工作模式设定为 A2 模式时，一般模式的功能将无法使用。

CR#504: 从站 Heartbeat 命令

[说明]

在一般模式下启动从站 Heartbeat 命令时，主站会一次启动节点编号 1~16 从站的 Heartbeat 机制。寄存器设定值为 0 表示关闭，设定值为 1 表示开启 Heartbeat 功能。

CR#505: 从站 Heartbeat 命令执行状态

[说明]

此寄存器内容值显示启动节点编号 1~16 从站 Heartbeat 命令的执行状态，内容值为 0 时表示执行完毕，内容值为 1 时表示执行中。

CR#506: 从站 Heartbeat 状态

[说明]

此寄存器内容值显示节点编号 1~16 从站 Heartbeat 状态，使用 Bit 个别表示节点 1~16，如下表所示。若该从站成功启动 Heartbeat 机制，该 Bit 显示为 1，若未开启 Heartbeat 机制，则 Bit 显示为 0。另外，当主站与该从站联机断线时，此寄存器对应的 Bit 也会显示为 0。

		從站Heartbeat顯示															
從站節點編號		16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

13 CANopen 通讯功能卡

- 对象辞典相关参数区

CR#H'1006: 同步周期设定

[说明]

设定一般模式下, Canopen 同步封包发送之时间间隔(单位: μs), 数据型态为 Dword, 初始值为 5000。目前同步周期的精度为 ms, 小于 1ms 的数值将忽略。同步周期的建议设定为, 当 CANopen 网络上存在 4 个以内同步发送的 PDO 时, 同步周期设定最小为 3ms, 当存在 4~8 个同步发送的 PDO 时, 同步周期设定最小为 4ms, 也就是每增加 4 个 PDO 时, 最小同步周期将增加 1ms。

CR#H'1017: Heartbeat 周期设定

[说明]

设定 FPMC 的 Heartbeat 发送周期时间, 单位为 ms, 初始值为 0。

CR#H'1400~CR#H'143F: 接收模式 PDO 参数设定

[说明]

设定一般模式下, 接收模式之 PDO 传输相关参数, 此寄存器大小为 3 个 Words, 设定格式请参考下表。

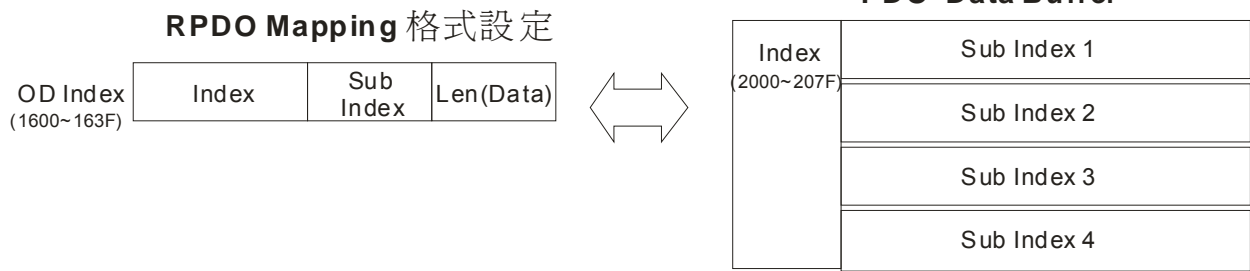
传输模式	PDO ID	
Word2	Word1(High)	Word0(Low)

- PDO ID: Canopen 网络中封包代码, 使用 2 个 Words 表示。初始值如下所示:
CR#H'1400=H180+FPMC 本身 Node ID(CR#053)
CR# H'1401=H280+FPMC 本身 Node ID(CR#053)
CR#H'1403=H380+FPMC 本身 Node ID(CR#053)
CR#H'1404=H480+FPMC 本身 Node ID(CR#053)
- 传输模式: 依设定值决定传输模式, 当设定值为 1~240, 表示 PDO 会跟随 Canopen 网络同步封包发送, 并且每隔设定值之同步周期均会发送一次。若设定值为 241~255 则不动作。初始值=241。

CR#H'1600~CR#H'163F: 接收模式 PDO 数据映像参数设定

[说明]

一般模式中, 设定接收模式 PDO 数据暂存区映像之地址型态与参数。PDO 数据映像参数数据型态为 Dword, 其中第一个 word 为对象辞典之索引, 第二个 word 中, high byte 为该对象辞典索引之子索引, low byte 为数据型态, 设定单位为 bit。设定格式与相对应之 PDO Data Buffer 请参考下图所示。



CR#H'1800~CR#H'183F: 传送模式 PDO 参数设定

[说明]

设定一般模式下，传送模式之 PDO 传输相关参数，此寄存器大小为 3 个 Words，设定格式请参考下表。

传输模式	PDO ID	
Word2	Word1(High)	Word0(Low)

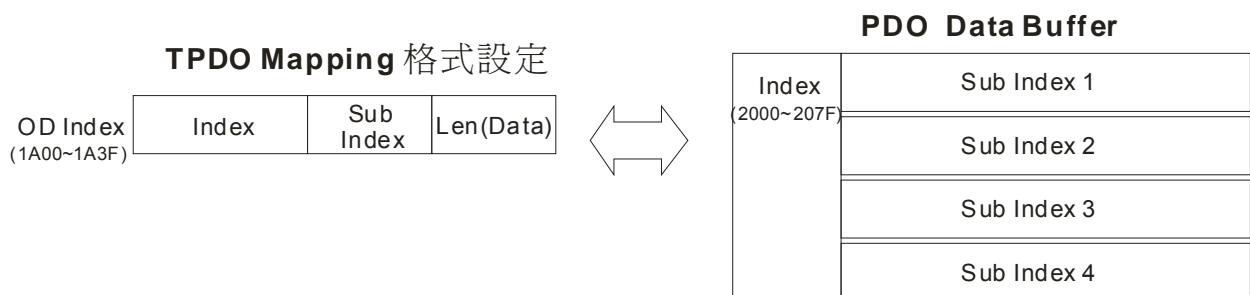
- PDO ID: Canopen 网络中封包代码，使用 2 个 Words 表示。初始值如下所示：
 CR#H'1800=H200+FPMC 本身 Node ID(CR#053)
 CR# H'1801=H300+FPMC 本身 Node ID(CR#053)
 CR#H'1803=H400+FPMC 本身 Node ID(CR#053)
 CR#H'1804=H500+FPMC 本身 Node ID(CR#053)
- 传输模式：依设定值决定传输模式，当设定值为 1~240，表示 PDO 会跟随 Canopen 网络同步封包发送，并且每隔设定值之同步周期均会发送一次。若设定值为 241~255 则不动作。初始值=241。

传输模式	PDO ID	
Word2	Word1(High)	Word0(Low)

CR#H'1A00~CR#H'1A3F: 传送模式 PDO 数据映像参数设定

[说明]

一般模式中，设定传送模式 PDO 数据暂存区映像之地址型态与参数。PDO 数据映像参数数据型态为 Dword，其中第一个 word 为对象辞典之索引，第二个 word 中，high byte 为该对象辞典索引之子索引，low byte 为数据型态，设定单位为 bit。设定格式与相对应之 PDO Data Buffer 请参考下图所示。



13 CANopen 通讯功能卡

CR#H'2000~CR#H'207F: PDO 数据寄存器

[说明]

PDO 传送接收的数据，目前 FPMC 卡开放对象辞典中 2000H~207FH 为资料暂存区，每一个索引包含 4 个可存放数据的子索引，其中子索引的大小为 1 个 Word。若 PDO 传输的数据大于 1 个 word，则可使用多个子索引区域进行数据的传输。

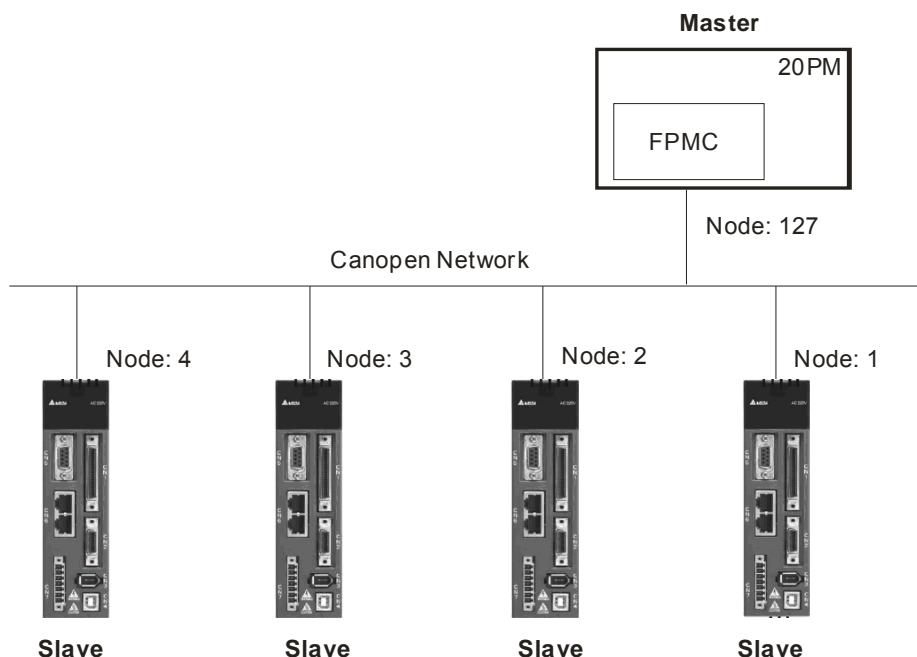
● FPMC 对象字典表

CR 位置 (16 进制)	数据类型	子指标个数	名称	数据类型	属性	允许作 mapping 对象
1000	VAR	1	产品类别	UNSIGNED32	RO	N
1006	VAR	1	同步周期	UNSIGNED32	RW	N
1018	ARRAY	5	产品信息	UNSIGNED32	RO	N
1200	ARRAY	3	主站 SDO 参数	UNSIGNED32	RO	N
1280	ARRAY	4	从站 SDO 参数	UNSIGNED32	RO	N
:	:	:	:	:	:	:
128F	ARRAY	4	从站 SDO 参数	UNSIGNED32	RO	N
1400	ARRAY	6	RPDO 参数	UNSIGNED32	RW	N
:	:	:	:	:	:	:
143F	ARRAY	6	RP DO 参数	UNSIGNED32	RW	N
1600	ARRAY	9	RPDO 映射参数	UNSIGNED32	RW	N
:	:	:	:	:	:	:
163F	ARRAY	9	RPDO 映射参数	UNSIGNED32	RW	N
1800	ARRAY	6	TPDO 参数	UNSIGNED32	RW	N
:	:	:	:	:	:	:
183F	ARRAY	6	TPDO 参数	UNSIGNED32	RW	N
1A00	ARRAY	9	TPDO 映射参数	UNSIGNED32	RW	N
:	:	:	:	:	:	:
1A3F	ARRAY	9	TPDO 映射参数	UNSIGNED32	RW	N
2000	ARRAY	5	PDO 数据寄存器	UNSIGNED32	RW	Y
:	:	:	:	:	:	:
207F	ARRAY	5	PDO 数据寄存器	UNSIGNED32	RW	Y
6000	ARRAY	5	模式控制	UNSIGNED8	R	Y
6100	ARRAY	17	驱动器控制	UNSIGNED16	R	Y
6120	ARRAY	17	位置模式参数	UNSIGNED32	R	Y
6200	ARRAY	5	驱动器运动模式状态	UNSIGNED8	RW	Y
6300	ARRAY	5	驱动器状态	UNSIGNED16	RW	Y
6320	ARRAY	5	驱动器位置	UNSIGNED32	RW	Y

13.6 FPMC 之模式设定

■ A2 模式

在 A2 模式中，DVP-FPMC 使用 CANopen 网络与 4 组台达伺服 ASDA-A2 通讯。在通讯过程中，主站为 DVP-FPMC，伺服为从站，架构图如下图所示，FPMC 之预设节点编号为 127，而规划联机对象为节点编号 1~4，使用上将对应伺服之 Canopen 节点编号设定为 1~4，即可与 FPMC 进行数据的交换。



在 A2 模式中规划六组 PDO 作为伺服参数设定用，使用者透过读写功能寄存器即可使用 Canopen 网络设定监看伺服信息，不需另行设定 PDO 相关参数。FPMC 与各节点伺服之 PDO 设定配置中，分别规划四组给 FPMC 与两组 PDO 给伺服发送数据用，发送的数据定义如下表所示。

PDO	Master 发送	Slave 发送
1	位置模式目标位置(CR#n70~CR#n71) 位置模式运转速度(CR#n72~CR#n73)	
2	位置模式运转加速时间(CR#n74~CR#n75) 位置模式运转减速时间(CR#n76~CR#n77)	
3	节点之伺服控制(CR#n60)	
4	补间模式目标位置(CR#n90~CR#n91)	
5		伺服状态(CR#n20) 伺服运动状态(CR#n21)
6		伺服之现在位置(CR#n22~ CR#n23)

A2 模式联机设定:

- ASDA-A2 伺服设定

在建立 Canopen 联机之前, 首先设定伺服的模式为 Canopen 模式, 其操作步骤如下所示:

1. 设定 A2 Keypad P1-01 为 0x0B (此设定为将 mode 设定为 CANopen mode)
2. 设定 A2 Keypad P3-00, 其数值代表 Node 编号, 数值范围为 0x01~0x04
3. 设定 Keypad P3-01, 设定 Baudrate 0x0403 代表 1Mbps (2: 500 kbps; 4: 1Mbps), 目前 FPMC 支持 1M (初始设定) 和 500k

- FPMC 设定

当所有装置之 Canopen 网络参数设定完毕后, 即可使用 FPMC 建立 Canopen 网络, 其步骤如下所示:

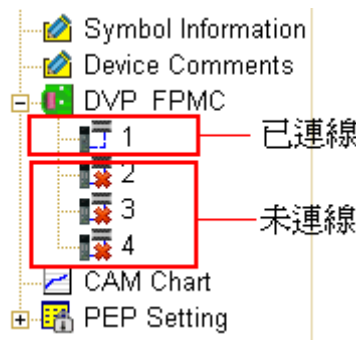
1. CR#500 内容值填入 1, 将 FPMC 模式设定为 A2 模式
2. CR#053 内容填入欲设定之 Canopen 网络节点 ID, FPMC 预设的节点 ID 为 127。
3. CR#010 内容填入 H'FFFF, 扫描所有联机伺服
4. From CR#010 的内容, 检视内容值是否清除为 0
5. CR#050 内容值填入 1, 开启所有已联机节点之伺服, 命令伺服 Servo On。

其中步骤 2~步骤 4 可使用 PMsoft 软件实现, 在 PMsoft 接口中欲开启使用 DVP-FPMC 功能, 在系统信息栏中展开 DVP-FPMC icon, 用鼠标右键选择欲执行的功能。扫描开启联机伺服, 操作步骤如下所示:

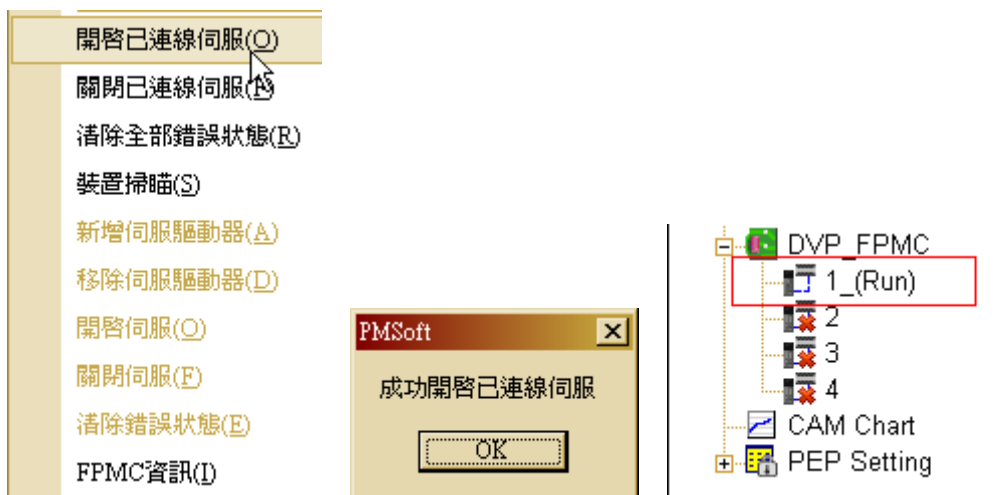
1. 选择 DVP-FPMC 之图标后，使用鼠标右键选择「装置扫描」



2. 当扫描完毕后，显示各节点联机状态



3. 选择 DVP-FPMC 之图标后，使用鼠标右键选择「开启已联机伺服」，当伺服成功开启时，便出现成功开启已联机伺服窗口，以及另外在各节点之图标显示伺服运行中的状态。



■ 一般模式

在一般模式下使用者需设定 FPMC 和 Slave 的 PDO 各项参数，透过 TO/FROM 指令对 FPMC 之功能寄存器设定，而在伺服中则必须利用 SDO 来对 PDO 的参数作写入的动作。FPMC 设定 PDO 的流程如下：

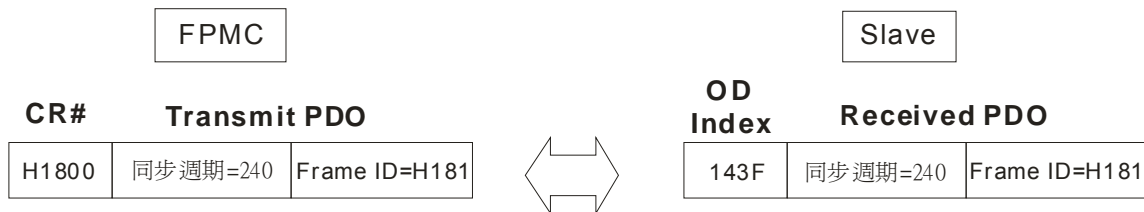
13 CANopen 通讯功能卡

1. 设定 PDO 的传输参数

PDO 参数中包含 Frame ID 与同步周期设定，Frame ID 设定范围为 181h ~ 578h 之间，需注意 Master 与 Slave 通讯对应的 PDO，其 Frame ID 与同步周期设定需相同。PDO 分为传输模式之 PDO(TPDO)与接收模式之 PDO(RPDO)，以下将分别介绍。

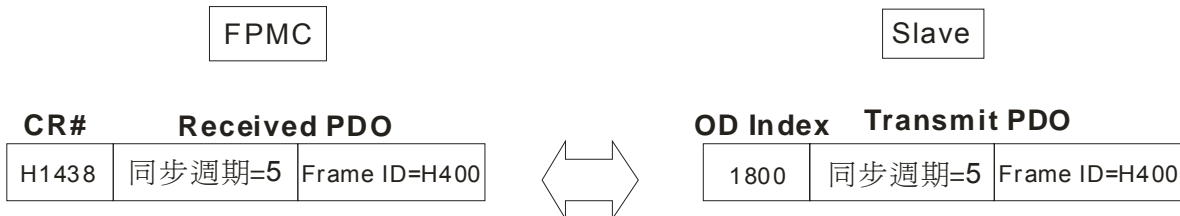
● TPDO 设定

FPMC 的 1800H~183FH 为 TPDO，对应的对象为 Slave 的 RPDO，举例说明，Master 使用对象辞典索引为 1800 之 TPDO 与 Slave 之对象辞典索引为 143F 之 RPDO 通讯，传输参数设定中，同步周期为 240，Frame ID 为 H'181，如下图所示：

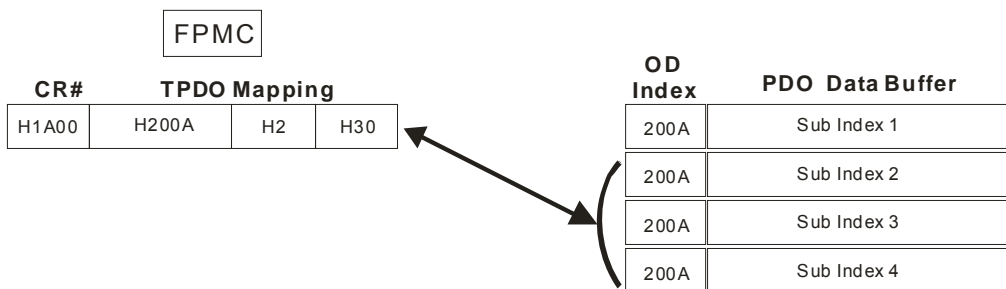


● RPDO 设定

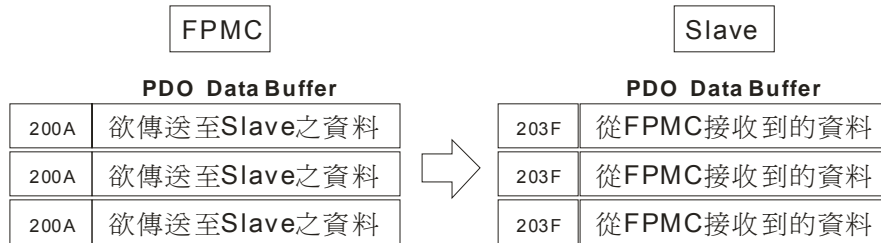
FPMC 的 1400H~143FH 为 RPDO，对应的对象为 Slave 的 TPDO，举例说明，Master 使用对象辞典索引为 1438 之 TPDO 与 Slave 之对象辞典索引为 1800 之 RPDO 通讯，传输参数设定中，同步周期为 5，Frame ID 为 H'400，如下图所示。



2. 设定 PDO Mapping 参数：设定映像到 PDO Data Buffer 中的位置，以及设定数据长度大小，数据大小最多可设定为 64 个 Bits，也就是可同时占用 4 个 PDO Data Buffer Sub Index。举例说明：设定 FPMC 的 TPDO 映像参数为 H200A 的第 2 个子索引当中，数据长度为 48 个 Bits，故占用的位置为 PDO Data Buffet 当中的 Sub Index 2~4。



3. 设定 PDO Data: 在上述设定的对象辞典中的位置, 写入 PDO 传输的数据。举例说明, FPMC 之 TPDO Mapping 设定由 PDO Data Buffer 索引为 200A 之 Sub index 2~3 为 TPDO 的数据位置; 而 Slave 端之 RPDO Mapping 设定由索引为 203F 之 Sub index 1~3 为 RPDO 的数据位置, 网络启动后, 会依照同步周期时间, 每隔同步周期数发送/接收一次数据。



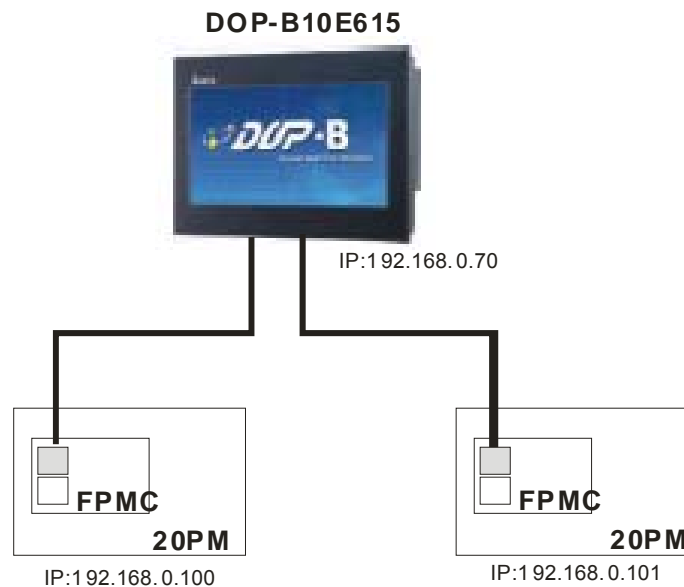
13.7 FPMC 之 Ethernet 模式介绍

FPMC 通讯卡另外支持 Ethernet 联机, 可用于连接含 Ethernet 功能之设备或是利用此模式与 PC 联机, 配合 PMsoft 执行程序上下载与实时监控等动作。使用方法仅需将通讯线连接 FPMC 的网络接口以及欲连接之设备之网络接口即可, 硬件配置请参考章节 13.3 产品外观与安装。当 FPMC 与 PC 可联机时 Ethernet 指示灯会亮起, 当指示灯未亮起, 需检查硬件或 PC 设定是否有错误。

13.7.1 Ethernet 联机介绍 — FPMC 与 HMI 做数据传递范例

■ 架构说明

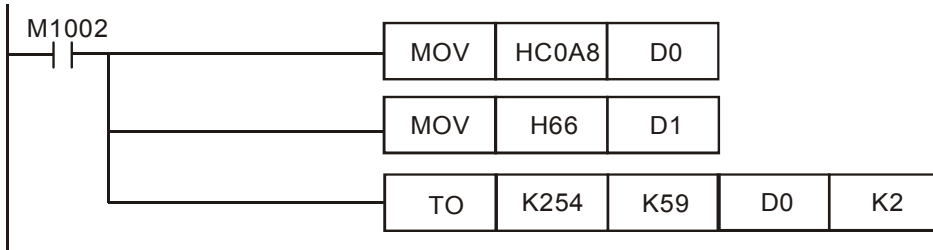
此联机范例使用 2 台 20PM 搭配 FPMC 通讯卡与触控式人机 DOP-B10E615 进行 Ethernet 联机作数据传递, 设备联机架构如下图所示。在人机程序中设定控制 2 台 PM 之 Y0~Y7。



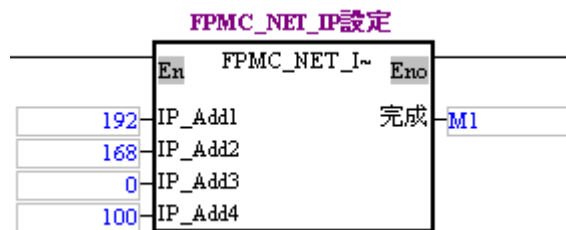
13 CANopen 通讯功能卡

■ FPMC 设定

在此应用范例中 FPMC 定义为从站，仅需设定本身的 IP 地址，等待主站连接，而不需设定欲连接对象之 IP 地址以及 Ethernet 之联机设定。设定 FPMC IP 方法为在 CR#59 填入欲设定的 IP，格式请参考章节 13.5 之 CR#59 内容说明。以设定 192.168.0.100 为例，PM 程序如下所示：



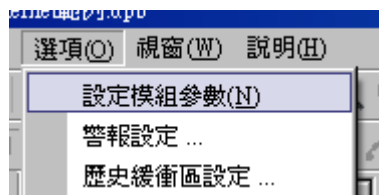
FPMC IP 地址设定之功能块如下图所示，输入接脚 IP_Addr1~IP_Addr4 分别表示 IP 地址的格式，以范例中 IP 地址 192.168.0.100 所示，设定完毕后输出接脚完成显示为 ON。



■ 人机端程序设定

人机作为主站连接两部从站，除了设定本身的 IP 地址外，在通讯设定上另外需建立两部从站之 Ethernet 联机。人机建立与 FPMC 之 Ethernet 联机步骤如下所示：

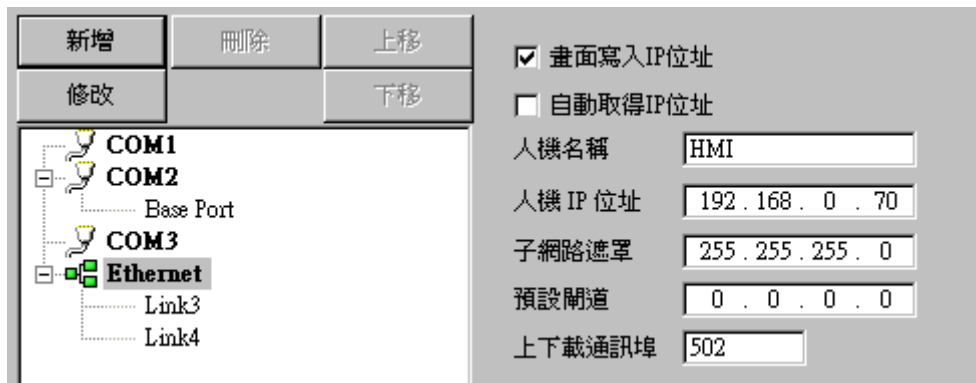
1. 在开启档案后在工具列中『选项』选择『设定模块参数』



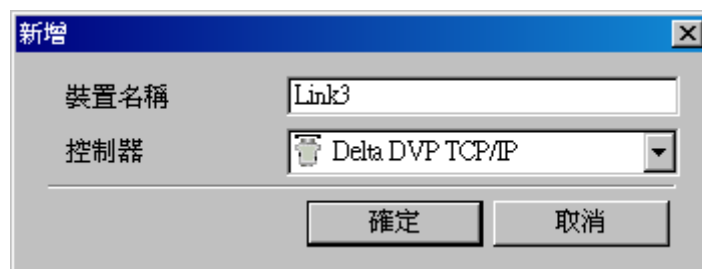
2. 在『一般』页面中，设定人机接口种类为 DOP-B10E615



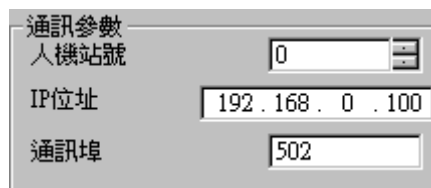
3. 在『通讯』页面选单中点选『Ethernet』，在旁边画面选项勾选『画面写入 IP 位置』，输入人机 IP 位置



4. 新增通讯对象之联机，点选『新增』后跳出设定窗口，设定装置名称，控制器选择 Delta DVT TCP/IP，设定完毕后选择『确定』。



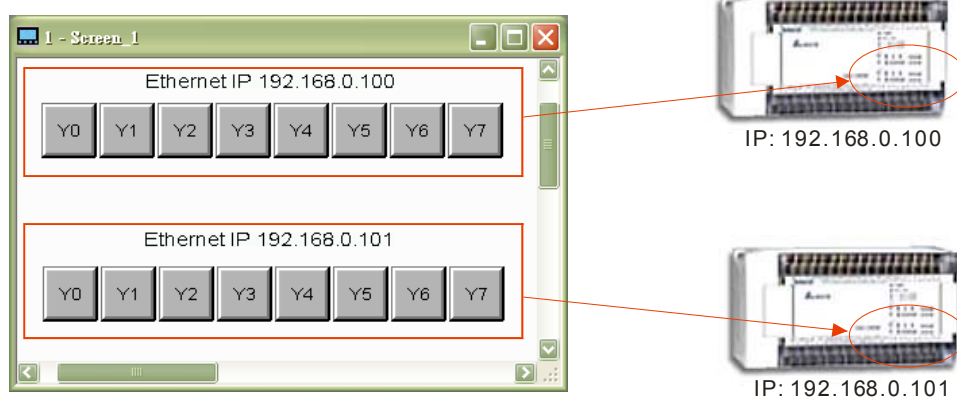
5. 设定通讯对象 IP，点选新增的通讯对象，在通讯参数中 IP 位置，设定 FPMC 卡的预定的 IP 位置。



6. 在使用的组件的输入页面中的联机选择先前定义的装置名称，即可利用 Ethernet 将此组件定义的内存位置做操作。



另外，在人机端设定控制两部从站 PM 的 Y0~Y7 输出，接口设定如下图所示。按钮 Y0~Y7 对应写入内存地址为 Link3 和 Link4 的 Y0~Y7，即为两部从站 PM 的 Y0~Y7。以上设定完毕后即可进行人机与两部 PM 之 Ethernet 联机。



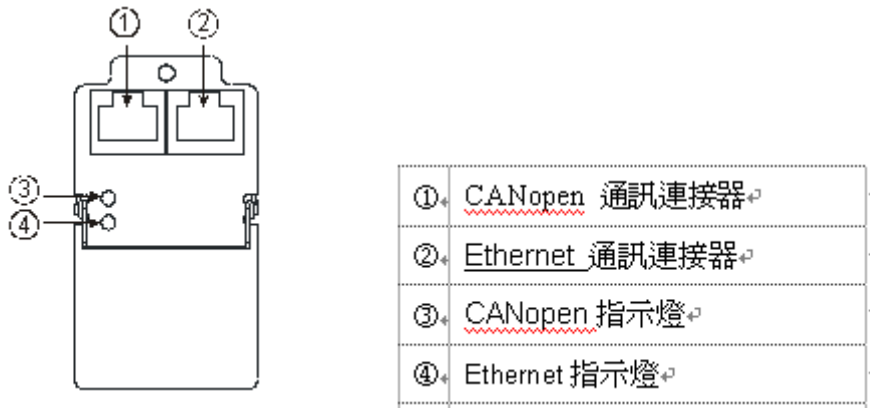
13 CANopen 通讯功能卡

13.7.2 FPMC 与 PMsoft 联机介绍

PMsoft 在通讯设定中，传输方式新增一 FPMC 功能，搭配 FPMC 的 Ethernet 联机功能，为 PM 新增一项与 PC 之通讯接口。配合 PM 编辑软件 PMsoft，利用 Ethernet 方式做 PM 程序上下载以及实时监控。

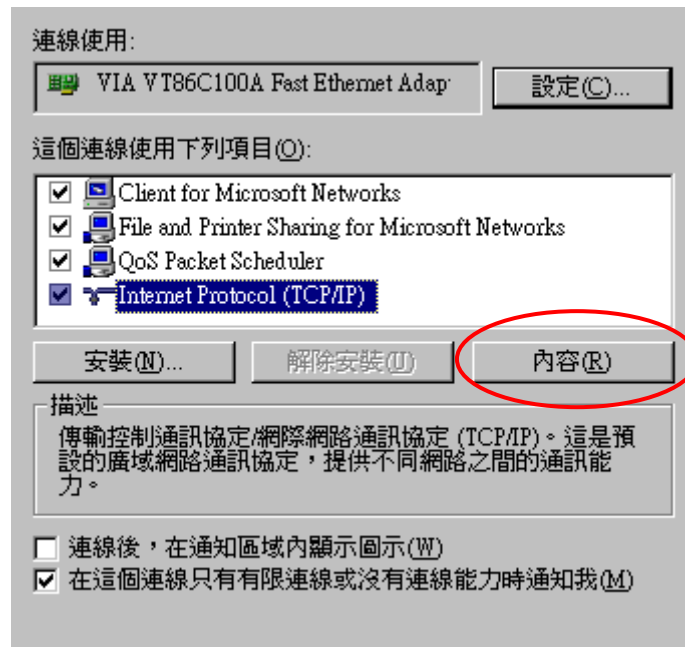
■ 硬件配线

在硬件联机上利用一般往路线连接 FPMC 的网络接口以及 PC 上的网络接口即可，当 FPMC 与 PC 可联机时 Ethernet 指示灯会亮起，当指示灯未亮起，需检查硬件或 PC 设定是否有错误。

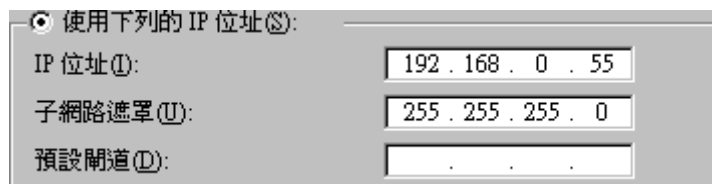


■ PC 端设定

1. 选取 PC 的局域网络内容中 TCP/IP，并点选内容开启 TCP/IP 内容页面

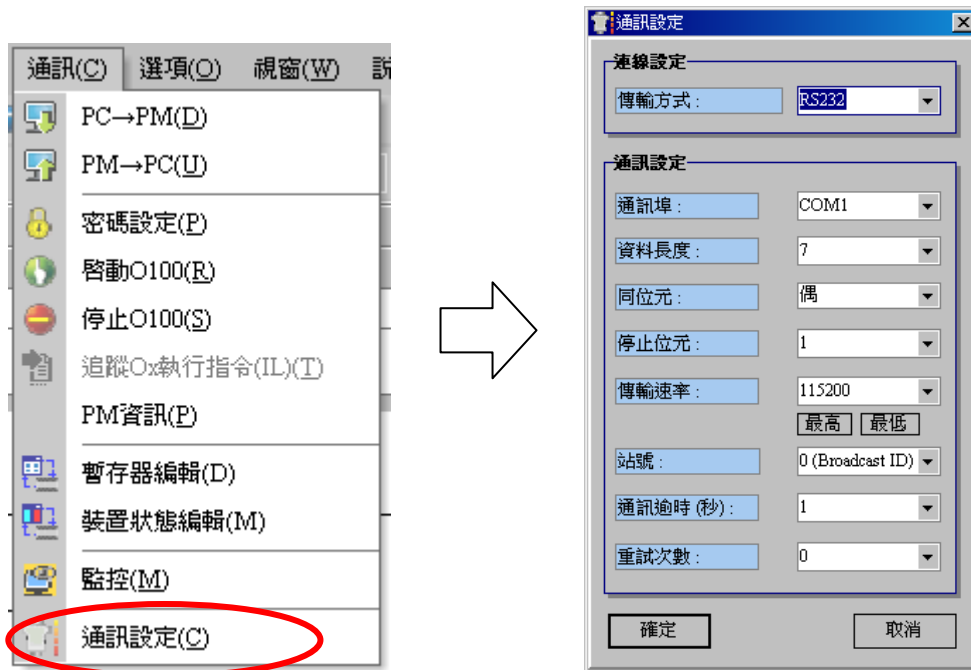


2. 在内容页面中选取，使用下列 IP 位置。IP 位置填入 192.168.0.55，最后一位可填除 100 外其它 1~255 的数值，子网掩码填入 255.255.255.0，完成后按确定



■ PMsoft 设定流程




1. 开启 PMsoft 选取通讯中的通讯设定，开启设定页面



2. 在设定页面中联机设定的传输方式选取 FPMC 而后按下确定



成功后，即可利用 Ethernet 做程序上下下载以及监控动作。

3. 程序下载动作，若要程序下载时，可按下快捷工具栏中的 PC->PM  或是利用下拉选单中通讯里面 PC->PM 做下载动作，下载流程与一般 PMsoft 程序下载流程相同。
4. 程序上传动作，若要程序上传时，可按下快捷工具栏中的 PM->PC  或是利用下拉选单中通讯里面 PM->PC 做上传动作，上传流程与一般 PMsoft 程序上传流程相同。
5. 实时监控动作，若要实时监控时，可按下快捷工具栏中的监控  或是利用下拉选单中通讯里面监控做实时监控，实时监控流程与一般 PMsoft 实时监控流程相同。

13 CANopen 通讯功能卡

13.8 LED 灯指示说明及故障排除

- CANopen 灯显示说明

LED 灯状态	显示说明	处理方法
绿灯灭	未连接上 Canopen 线	检查线路是否连接确实正确
绿灯亮	Canopen 线连接正常	无须任何动作

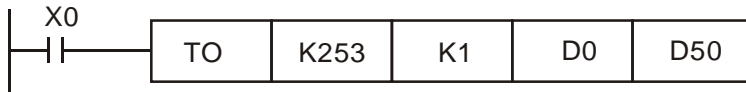
- Ethernet 灯显示说明

LED 灯状态	显示说明	处理方法
绿灯灭	未连接上网络	检查网络线是否连接确实
绿灯亮	网络联机正常	无须任何动作
绿灯闪烁	网络运作中	

14.14 高速比较与捕捉功能

DVP-PM 配置 8 组高速比较与捕捉功能，透过 FROM/TO 指令设定/读取 CR#253 控制高速比较或是捕捉功能状态与数据，下表说明如何使用 FROM/TO 指令来设定/读取高速比较与捕捉功能的状态：

■ 设定控制/读取



- D0 设定起始组别 n(n=0~7)
- D1 0
- D3,D2 组别 n 控制寄存器 CRn
- D5,D4 组别 n 数据寄存器 DRn
- D7,D6 组别 n+1 控制寄存器 CRn
- D9,D8 组别 n+1 数据寄存器 DRn
- :
- D31,D30 组别 n+7 控制寄存器 CRn
- D33,D32 组别 n+7 数据寄存器 DRn
- D50 设定数据长度，2+m*4(每组数据要 4word),m 为连续使用数据组数，最多 8 组

(1) 高速比较之控制寄存器 CRn 设定格式如下：

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
設定項目					比較結果輸出				輸出動作		比較條件		比較資料來源			

設定項目	Bit	設定值	20PM	10PM
比较数据 来源	[3-0]	0	X 轴现在位置	X 轴现在位置
		1	Y 轴现在位置	Y 轴现在位置
		2	Z 轴现在位置	Z 轴现在位置
		3	C200 计数值	A 轴现在位置
		4	C204 计数值	C200 计数值
		5	—	C204 计数值
		6	—	C208 计数值
		7	—	C212 计数值
比较 条件	[5-4]	1	等于(=)	等于(=)
		2	大于等于(≥)	大于等于(≥)
		3	小于等于(≤)	小于等于(≤)
输出 动作	[7-6]	0	触发(Set)	触发(Set)
		1	重置(Rst)	重置(Rst)
		2、3	—	输出不动作
比较 结果	[11-8]	0	CLR0	Y0
		1	CLR1	Y1
		2	Y2	Y2

14 高速比较与捕捉

	3	Y3	Y3
	4	清除 C200 计数值	清除 C200 计数值
	5	清除 C204 计数值	清除 C204 计数值
	6	CAM 前置量结束	清除 C208 计数值
	7	CAM 咬合	清除 C212 计数值

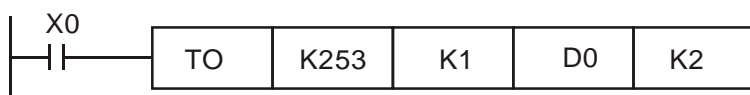
(2) 捕捉之控制寄存器 CRn 设定格式如下:

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
設定項目	觸發來源設定				輸出結果				輸出動作		捕捉設定		捕捉資料來源			

設定項目	Bit	設定值	20PM	10PM
捕捉資料來源	[3-0]	0	X 轴现在位置	X 轴现在位置
		1	Y 轴现在位置	Y 轴现在位置
		2	Z 轴现在位置	Z 轴现在位置
		3	C200 计数值	A 轴现在位置
		4	C204 计数值	C200 计数值
		5	—	C204 计数值
		6	—	C208 计数值
		7	—	C212 计数值
捕捉設定	[5-4]	0	补捉设定为 0	
輸出動作	[7-6]	0	—	輸出不动作
		1	—	
		2	—	触发(Set)
		3	—	重置(Rst)
輸出結果	[11-8]	0	—	Y0
		1	—	Y1
		2	—	Y2
		3	—	Y3
		4	—	清除 C200 计数值
		5	—	清除 C204 计数值
		6	—	清除 C208 计数值
		7	—	清除 C212 计数值
外部觸發來源設定	[15-12]	0	X 轴 PG0	X0
		1	X 轴 MPGB	X1
		2	X 轴 MPGA	X2
		3	X 轴 LSN	X3
		4	X 轴 LSP	X4
		5	X 轴 DOG	X5

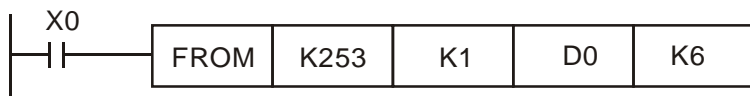
	6	X 轴 STOP	X6
	7	X 轴 START	X7
	8	Y 轴 PG0	X8
	9	Y 轴 MPGB	X9
	10	Y 轴 MPGA	X10
	11	Y 轴 LSN	X11
	12	Y 轴 LSP	—
	13	Y 轴 DOG	—
	14	Y 轴 STOP	—
	15	Y 轴 START	—

■ 设定初始状态



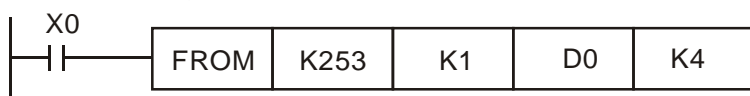
D0 0
D1 1, 清除所有输出及捕捉致能

■ 读取初始状态



D0 0
D1 1
D3,D2 输出状态
D5,D4 捕捉致能 CapE (8bits)

■ 设定捕捉遮没值

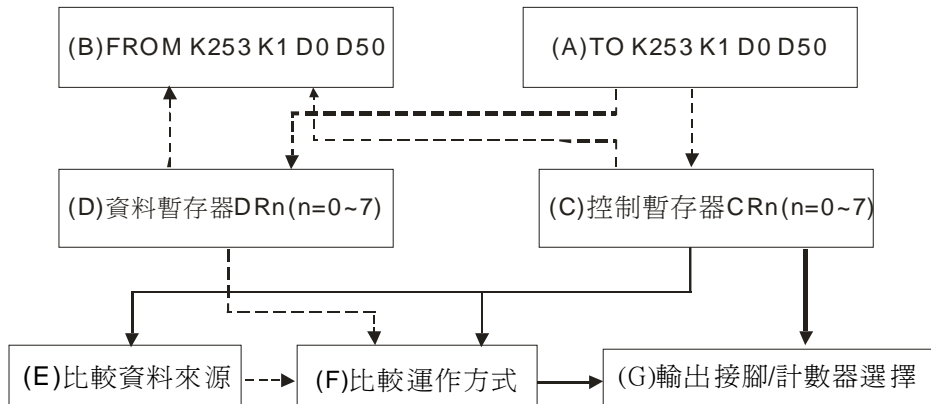


D0 0
D1 2, 设定捕捉遮没值
D3,D2 遮没值

14 高速比较与捕捉

14.2 高速比较

高速比较的功能如下图所示，使用 **FROM/TO** 指令读取/写入设定值去控制运作方式及比较数据，下图说明高速比较功能的运作方式。



※ 虚线为数据流，实线为控制流程

方块图(A)：使用 **TO** 指令写入数据至控制寄存器(方块图 C)及数据寄存器(方块图 D)。

方块图(B)：使用 **FROM** 指令读取数据至控制寄存器(方块图 C)及数据寄存器(方块图 D)。

方块图(C)：控制数据寄存器依照其接收 **TO** 指令数据去设定比较数据来源(方块图 E)、比较运作方式(方块图 F)及输出接脚设定(方块图 G)。

方块图(D)：储存 **TO** 指令写入的比较数据用与方块图(E)所设定的另外一组数据比较。

方块图(E)：比较数据来源有三轴现在位置、**C200**(CAM 主轴计数值，X 轴手轮)和 **C204** 计数值(Y 轴手轮)等五种选择，其中 **C200/C204** 可参考前面高速计数器章节。

方块图(F)：比较方式有等于、大于等于和小于等于三种。当方块图(D)数据与方块图(E)所设定数据来源比较方式成立，而触发方式为设定(**Bit7-6=0**)，即方块图(G)之接脚(**Clr0**、**Clr1**、**Y2** 和 **Y3** 或是 **C200/C204** 计数器 (若 **Bit7-6** 清除则计数器数值清除为零，不能再计数)，若触发方式设定为清除(**Bit7-6=1**) 就对设定的接脚清除或是清除计数器状态，方可计数)。附注：**Clr0**、**Clr1**、**Y2** 和 **Y3** 触发 **PM** 外壳 **LED** 灯不会显示，可将 **Clr0-**、**Clr1-**、**C2** 接 **24G**，**Clr0+**、**Clr1+**、**Y2** 和 **Y3** 接 **Xn**，而 **S/S2** 接 **+24V**，当 **Clr0**、**Clr1**、**Y2** 和 **Y3** 其中触发，其对应的 **X** 点 **LED** 灯将显示。

方块图(G)：当比较成立，触发接脚选择设定或是计数器设定，分别有 **Clr0**、**Clr1**、**Y2(00M)**、**Y3(00M)**、**C200** 和 **C204** 六种选择设定。

其高速比较功能的运作流程为：方块图(A)**TO** 指令写入数据设定相关控制寄存器→方块图(E)设定的数据来源数据与方块图(D)数据寄存器数据比较符合方块图(F)设定→立即根据方块图(G)接脚状态输出，或是 **C200/C204** 清除为零停止计数或是可再计数。

■ 范例：

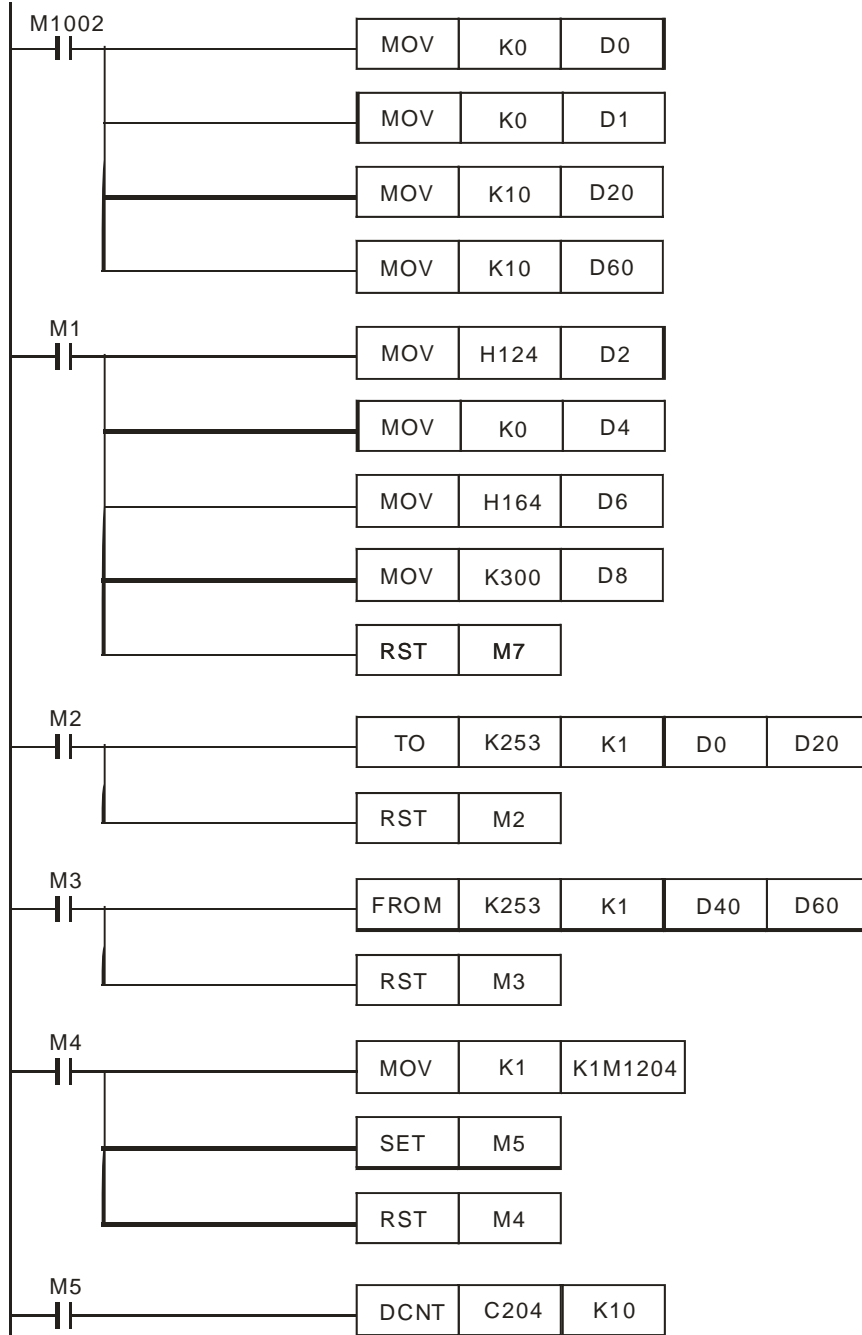
以下范例使用 C204 高速计数，当计数值超过 100 时触发 Clr1，超过 300 时重置 Clr1；程序中使用两组比较器，一组用来触发 Clr1，另外一组用于重置 Clr1。在 Clr1 触发时，PM 主机外壳上的 Y 轴 CLR 讯号灯并不会显示，但可透过外部线路得知 Clr1 是否为 ON，故在外部接线中将 CLR1-端子接 24G，CLR1+接 X7，S/S2 接+24V，以及外接一手摇轮。程序如下图所示：

比较器初始设定：使用=0,1 两组比较器，FROM/TO 数据长度 10 笔

比较器设定：

n=0，比较来源 C204(b[3:0]=4)，设定大于等于(b[5:4]=2)，触发 SET(b[7:6]=0)，触发讯号 CLR1(b[11:8]=1)，比较值=100

n=1，比较来源 C204(b[3:0]=4)，设定大于等于(b[5:4]=2)，触发 RST(b[7:6]=1)，触发讯号 CLR1(b[11:8]=1)，比较值=300



14 高速比较与捕捉

程序顺序控制如上图所示：

步骤一：启动 O100 时先执行高速比较初始设定：

- (1) D0=0 → 设定第一组比较起始组别 n=0
- (2) D1=0
- (3) D20=10→ TO 指令写入 10 笔数据，有两组高速比较数据
- (4) D60=10→ FROM 指令读取 10 笔数据，有两组高速比较数据

步骤二：设定 M1，设定两组高速比较数据

- (1) 第一组 控制数据 D3,D2=H124 → 比较来源 C204(Bit3-0=4)，大于等于设定(Bit5-4=2)，触发设定 (Bit7-6=0)，触发接脚设定 Clr0 (Bit11-8=1)
- (2) 第一组 暂存数据 D5,D4=K100 设定比较值大于等于 100 将触发
- (3) 第二组 控制数据 D7,D6=H164 → 比较来源 C204(Bit3-0=4)，大于等于设定(Bit5-4=2)，触发清除 (Bit7-6=1)，触发接脚设定 Clr0 (Bit11-8=1)
- (4) 第二组 暂存数据 D9,D8=K300 设定比较值大于等于 300 将触发

步骤三：设定 M2 启动两组高速比较功能

步骤四：设定 M3 读取两组高速比较功能设定状态，如下图所示。



装置	数值...	值	註解
C204	d32u	0	Y軸手輪計數值
D44	d32u	100	讀取到 n=0 資料暫存器資料 D5..D4
D48	d32u	300	讀取到 n=1 資料暫存器資料 D9..D8
D40	d16u	0	讀取 D0 資料
D41	d16u	0	讀取 D1 資料
D42	h32	00000124	讀取到 n=0 控制暫存器資料 D3..D2
D44	d32u	100	讀取到 n=0 資料暫存器資料 D5..D4
D46	h32	00000164	讀取到 n=1 控制暫存器資料 D7..D6
D48	d32s	300	讀取到 n=1 資料暫存器資料 D9..D8

步骤五：设定 M4 设定 C204 为 PD(脉波+方向)，然后设定 M5，启动计数。

步骤六：转动手摇轮，看 C204 是否有计数，如下图所示。



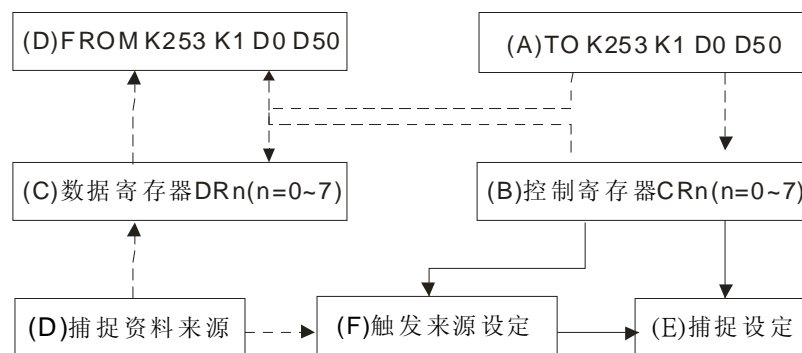
装置	数值...	值	註解
C204	d32u	95	Y軸手輪計數值
D44	d32u	100	讀取到 n=0 資料暫存器資料 D5..D4
D48	d32u	300	讀取到 n=1 資料暫存器資料 D9..D8

步骤七：转动手摇轮，当 C204 值大于 100，观察 PM 外壳的 X7 指示灯是否有亮，表示已触发 Clr1。

步骤八：手轮持续转动，当 C204 值大于 300，观察 PM 外壳的 X7 指示灯是否有熄灭，表示已重置 Clr1。

14.3 捕捉功能

一般程序扫描执行读取三轴位置数据或是 C200/C204 计数数据会有误差值，使用捕捉功能透过外部输入讯号立即执行读取数据的动作，可以避免误差发生的情形。下图说明捕捉功能的运作方式



※ 虚线为数据流，实线为控制流程

方块图(A)：使用 TO 指令写入数据至控制寄存器(方块图 B)及数据寄存器(方块图 C)。

方块图(B)：控制寄存器依照其接收 TO 指令数据去设定捕捉数据来源(方块图 D)、捕捉设定(方块图 E)及触发来源接脚设定(方块图 F)。

方块图(C)：储存 TO 指令写入的数据，或是储存外部硬件讯号触发而将方块图(D)所设定的数据来源装置数据。

方块图(D)：捕捉的来源为：1.三轴现在位置、2.C200(CAM 主轴计数值，为 X 轴手摇轮)、3.C204 计数值(Y 轴手摇轮)，其中 C200/C204 可参考前面高速计数器章节。

方块图(E)：设定捕捉功能。

方块图(F)：触发捕捉讯号来源。

方块图(G)：使用 FROM 指令读取数据至数据寄存器(方块图 C)及控制寄存器(方块图 B)，其中数据寄存器存放数据为捕捉到的数据。

捕捉功能的运作流程：TO 指令写入数据设定相关控制寄存器(方块图 A)→当外部触发讯号产生(方块图 F)→立即捕捉方块图(D)所设定的数据来源，将此数据放入方块图(C)的数据寄存器→使用 FROM 指令读取捕捉到的数据。

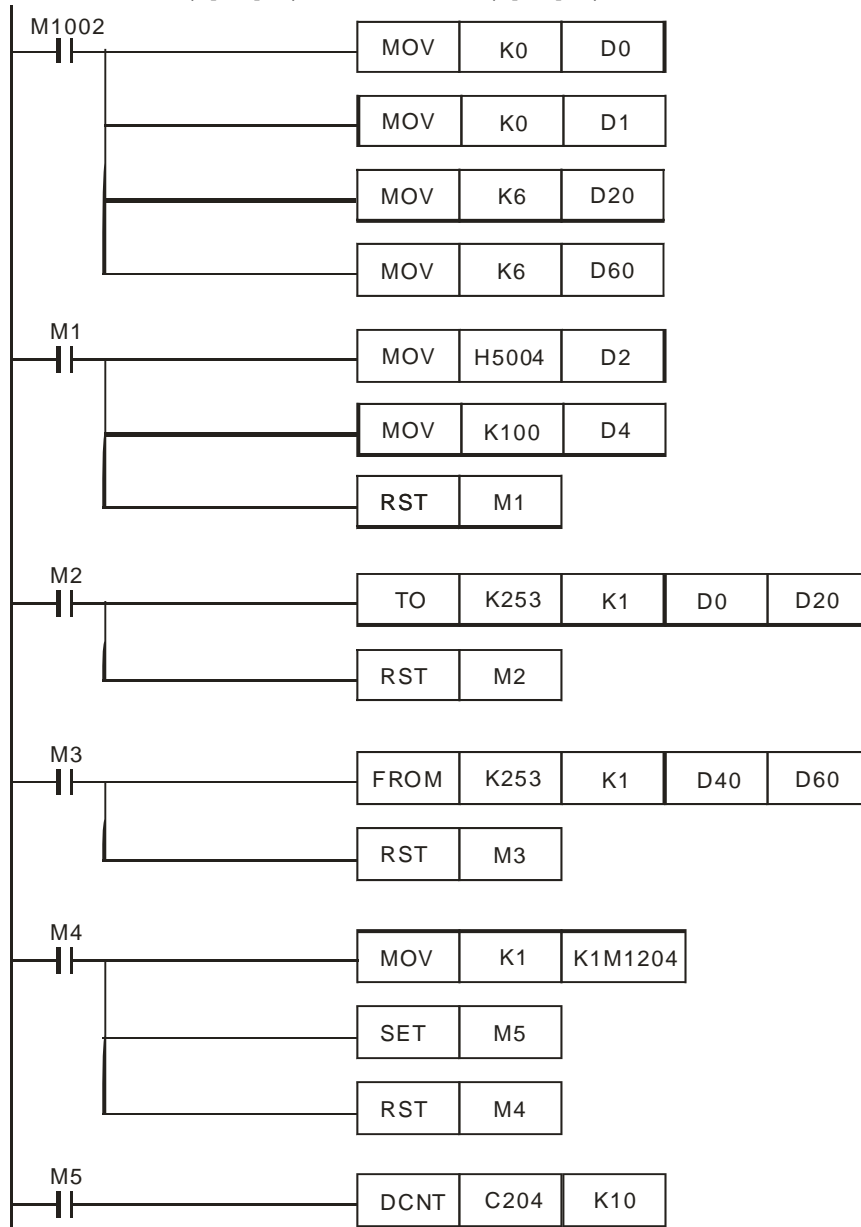
14 高速比较与捕捉

■ 范例：

使用 Y 轴手摇轮启动 C204(Y 轴手轮)高速计数功能，使用外部讯号 DOG0 触发补捉 C204 计数值(手轮高速转动时)，程序如下图所示：

捕捉初始设定：使用第一组 n=0 捕捉器，数据长度=6

捕捉来源 C204(b[3:0]=4)，设定捕捉模式(b[5:4]=0)，捕捉讯号为 X_DOG(b[15:12]=5)



程序顺序控制如上图所示：

步骤一：O100 运作 M1002 瞬间为 ON 捕捉初始设定：

- (1) D0=0 → 设定第一组捕捉起始组别 n=0
- (2) D1=0
- (3) D20=6 → TO 指令写入 6 笔数据，只有一组捕捉资料

(4) D60=6→ FROM 指令读取 6 笔数据，只有一组捕捉数据

步骤二：M1=1 设定捕捉

(1) 控制数据 D3..D2=H5004 → 捕捉来源 C204(Bit3-0=4)，捕捉模式设定(Bit5-4=0)，触发补捉讯号 X 轴_DOG0(Bit15-12=5)

(2) 暂存资料 D5..D4=K100 可任意值忽略

步骤三：M2=1 使用 TO 指令写入，捕捉功能启动

步骤四：M3=1 使用 FROM 指令读取第一组捕捉设定状态，如下为读取状态值



裝置	數...	值	註解
C204	d32s	0	
D44	d32u	100	對應 D5..D4 資料暫存器
D40	d16s	0	對應 D0
D41	d16s	0	對應 D1
D42	h32	00005004	對應 D3..D2 控制暫存器
D44	d32u	100	對應 D5..D4 資料暫存器

步骤五：设定 M4，设定 C204 为 PD(脉波+方向)。设定 M5。启动计数

步骤六：转动手轮，监看 C204 是否有计数，如下图所示：



裝置	數...	值	註解
C204	d32s	244	
D44	d32u	100	對應 D5..D4 資料暫存器
D40	d16s	0	對應 D0
D41	d16s	0	對應 D1
D42	h32	00005004	對應 D3..D2 控制暫存器
D44	d32u	100	對應 D5..D4 資料暫存器

步骤七：持续转动手摇轮，并使用外部讯号触发 DOG0。

步骤八：设定 M3 读取第一组捕捉设定状态，D44 为触发 DOG0 时捕捉到 C204 的计数值。

14 高速比较与捕捉



装置	數...	值	註解
C204	d32s	726	
D44	d32u	677	對應 D5..D4 資料暫存器
D40	d16s	0	對應 D0
D41	d16s	0	對應 D1
D42	h32	00005004	對應 D3..D2 控制暫存器
D44	d32u	677	對應 D5..D4 資料暫存器

DOG0信号触发
捕捉C204计数值

15.1 附录 A 侦错码原因对照表

将程序写入 DVP-PM 内部后，在不同的程序区块 O100、OX 若发生 ERROR，错误指示灯闪烁，错误旗标 On，原因可能是指令操作数（装置）使用不合法，程序语法回路错误，或运动参数设定不当，可根据错误缓存器的错误码（16 进制编码）并对照下表，以得知错误原因。

◆ O100/OX 对应之错误讯息表

程序区块	O100				OX			
错误类别	程序错误	运动错误			程序错误	运动错误		
		X 轴	Y 轴	Z 轴		X 轴	Y 轴	Z 轴
错误旗标	M1953	M1793	M1873	M2033	M1793	M1793	M1873	M2033
错误缓存器	D1802	D1857	D1937	D2017	D1857	D1857	D1937	D2017
STEP 数	D1803	D1869			D1869	D1869		

◆ 程序区及运动轴之错误码(以 16 进制表示)

错误码	原因	错误码	原因
0002	使用子程序无内容	0031	正方向脉波禁止
0003	CJ、CJN、JMP 缺少对应的 Pn	0032	反方向脉波禁止
0004	主程序中有子程序旗标	0033	到达左右极限
0005	缺少子程序	0040	装置组件使用范围错误
0006	同程序中指针重复	0041	MODRD、MODWR 通讯逾时
0007	子程序指针重复	0044	V/Z 修饰错误
0008	不同子程序中的跳跃指令指针重复	0045	浮点数转换错误
0009	跳跃指令与呼叫子程序指令使用相同旗标	0E18	BCD 转换错误
000A	指针与子程序的指针一样	0E19	除法演算错误（除数=0）
0011	目标位置(I)设定错误	C401	一般回路错误
0012	目标位置(II)设定错误	C402	LD/LDI 指令连续使用 9 次以上
0021	运转速度(I)设定错误	C404	RPT~RPE 超过 1 层以上
0022	运转速度(II)设定错误	C405	SRET 使用在 RPT~RPE 之间
0023	原点复归速度 (V _{RT}) 设定错误	C4EE	程序中没有结束指令 (M102、M2)
0024	原点复归减速速度 (V _{CR}) 设定错误	C4FF	无此指令/操作数格式或范围错误
0025	寸动 JOG 速度设定错误		

15.2 附录 B 手册修改记录

项次	说明	章节
1	硬件配线图中新增 Z 轴	Chapter 2
2	电气规格新增最大输入电流值及最大输出电流值	Chapter 2
3	新增修改特 D 缓存器指令与说明 D1798、D1864、D1846、D1832~D1834、D1945、D1948~D1949、D1952~D1954、 D2025、D2028~D2029、D2032~D2035 新增修改特 M 缓存器指令与说明 M1035、M1036、M1811、M1905、M1956、M2000、M2036、M2048~M2051	Chapter 3
4	新增 DRV 三轴高速定位、LIN 三轴同动直线补间移动、CW/CCW 螺旋移动、 XY/XZ/YZ 平面设定使用说明	Chapter 6
5	指令说明区分 20D & 20M 机种	Chapter 6
6	新增 G 码可用缓存器 D 指定变量	Chapter 6
7	新增第 8.4 章节 DVP-PM(Master) 连接 DVP01PU-H2(Slave) 第三轴控制	Chapter 8
8	新增第 9 章电子凸轮	Chapter 9
9	修改 PM 搭配各系列伺服驱动器之配线图	Chapter 2
10	修改差动输入之配线的说明	Chapter 2
11	修改 M1796 的 Menu→Auto 为 On	Chapter 3
12	删除阶梯图范例中 O100、M102、OXn / M2、Pn / SRET 的指令行	Chapter 1,3,5,6,7,8
13	删除其它指令中的 NOP 项与说明	Chapter 4
14	删除第 6 章指令表头说明含有“但不包含 V、Z 装置”字符串	Chapter 6
15	新增第 10 章 PEP 功能说明	Chapter10
16	新增第 11 章 G 码应用	Chapter11
17	新增第 12 章 PMSoft POU 功能	Chapter12
18	新增第 13 章 DVP FPMC 使用说明	Chapter13
19	指令说明表格中适用机种选项加入 10PM	Chapter4,5,6
20	加入 10PM 之硬件规格以及各家服务器与 10PM 之硬件配线图	Chapter 2
19	新增第 14 章高速比较与捕捉功能说明	Chapter14

15.3 附录 C PM 各机种功能支援表

功能		DVP-20PM00D	DVP-20PM00M	DVP-10PM	章节
程序区	O100 主程序	支援	支援	支援	1.1
	OX 运动子程序	支援	支援	X	1.2
	P 子程序	支援	支援	支援	1.3
指令	一般指令/应用指令	130 个	130 个	130 个	第四章
	运动指令	22 个	22 个	X	6.2
	G-Code	10 个	10 个	X	6.4
	M-Code	65536 个	65536 个	X	6.3
单轴运动	JOG 模式	二轴	三轴	四轴	3.11
	原点复归模式	二轴	三轴	四轴	3.11
	变速度模式	二轴	三轴	四轴	3.11
	单段速定位模式	二轴	三轴	四轴	3.11
	插入单段速定位模式	二轴	三轴	四轴	3.11
	两段速定位模式	二轴	三轴	四轴	3.11
	插入两段速定位模式	二轴	三轴	四轴	3.11
	单段速外部触发模式	X	三轴	X	3.11
	手摇轮模式	二轴	三轴	四轴	3.11
	周期/非周期式电子凸轮模式	二轴	三轴	X	3.11
PWM	X	X	四轴	3.11	
高速计数器	高速捕捉	8 组	8 组	8 组	14.2
	高速比较	8 组	8 组	8 组	14.3
	高速计时	X	X	6 组	3.11
扩充卡	FPMC	支援	支援	支援	第十三章
	FPMSD	支援	支援	支援	-