

客服热线 (021) 5863-9595

## 绵密网络 专业服务

中达电通已建立了 41 个分支机构及服务网点，并塑造训练有素的专业团队，提供客户最满意的服务，公司技术人员能在 2 小时内回应您的问题，并在 48 小时内提供所需服务。

上海 电话:(021)6301-2827 传真:(021)6301-2307	南昌 电话:(0791)6255-010 传真:(0791)6255-102	合肥 电话:(0551)2816-777 传真:(0551)2816-555	南京 电话:(025)8334-6585 传真:(025)8334-6554	杭州 电话:(0571)8882-0610 传真:(0571)8882-0603
武汉 电话:(027)8544-8265 传真:(027)8544-9500	长沙 电话:(0731)8827-7881 传真:(0731)8827-7882	南宁 电话:(0771)5879-599 传真:(0771)2621-502	厦门 电话:(0592)5313-601 传真:(0592)5313-628	广州 电话:(020)3879-2175 传真:(020)3879-2178
济南 电话:(0531)8690-7277 传真:(0531)8690-7099	郑州 电话:(0371)6384-2772 传真:(0371)6384-2656	北京 电话:(010)8225-3225 传真:(010)8225-2308	天津 电话:(022)2301-5082 传真:(022)2335-5006	太原 电话:(0351)4039-475 传真:(0351)4039-047
乌鲁木齐 电话:(0991)6118-160 传真:(0991)6118-289	西安 电话:(029)8836-0640 传真:(029)88360640-8000	成都 电话:(028)8434-2075 传真:(028)8434-2073	重庆 电话:(023)8806-0306 传真:(023)8806-0776	哈尔滨 电话:(0451)5366-0643 传真:(0451)5366-0248
沈阳 电话:(024)2334-1612 传真:(024)2334-1163	长春 电话:(0431)8892-5060 传真:(0431)8892-5065			

# DVP-10PM 应用技术手册 (程序篇)



## DVP-10PM 应用技术手册 (程序篇)



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>

DVP-0179710-01  
2013-03-12

中达电通公司版权所有  
如有改动，恕不另行通知

[www.delta.com.tw/ia](http://www.delta.com.tw/ia)



# DVP-10PM 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 .....	1-2
1.1.1	O100 主程序手动功能运动方式 .....	1-3
1. 2	OX 运动子程序架构 .....	1-3
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
1.4.1	程序架构介绍 .....	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 .....	2-6
2.2.1	盘内安装配线 .....	2-6
2.2.2	电源输入配线 .....	2-6
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 .....	2-14
2. 3	通讯端口说明 .....	2-24
2.3.1	COM1 ( RS-232 ) 规格 .....	2-24
2.3.2	COM2 ( RS-485 ) 规格 .....	2-25
2.3.3	COM3 (RS-232/RS-485) .....	2-25

### 第 3 章：各种装置功能

3. 1	各装置编号一览表 .....	3-1
3. 2	数值、常数[K、H]、浮点数[F] .....	3-4
3. 3	外部输入/输出接点的编号及功能[X、Y] .....	3-7
3. 4	辅助继电器的编号及功能[M] .....	3-9
3. 5	内部继电器的编号及功能[S] .....	3-9

3.6	定时器的编号及功能[T]	3-9
3.7	计数器的编号及功能[C]	3-10
3.8	寄存器的编号及功能[D]、[V]、[Z]	3-16
3.8.1	数据寄存器[D]	3-16
3.8.2	变址寄存器[V]、[Z]	3-16
3.9	指针[N]、指针[PN]	3-17
3.10	特殊继电器 M 及特殊寄存器 D	3-17
3.11	特殊继电器及特殊寄存器功能说明	3-35
3.12	手动运动模式特殊寄存器	3-50
3.12.1	手动运动模式特殊寄存器功能说明	3-52
3.12.2	手动运动模式程序介绍	3-74
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	3-75

## 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	5-8
5.4	使用变址寄存器 V、Z 来修饰操作数	5-11
5.5	指令索引	5-12
5.6	应用指令	5-16
	• (API 00~09) 回路控制	5-16
	• (API 10~19) 传送比较	5-24
	• (API 20~29) 四则逻辑运算	5-37
	• (API 40~49) 数据处理	5-63
	• (API 70~79) 外部设定显示	5-86
	• (API 100~109) 台达变频器通讯	5-91
	• (API 110~138) 浮点运算	5-100
	• (API 215~223) 比较接点指令	5-133
	• (API 250~260) 新增指令	5-147
5.7	运动功能块一览表	5-152
5.8	运动功能块引脚介绍	5-153

5.8.1	输入/输出引脚功能定义	5-153
5.8.2	输入/输出引脚时序图	5-154
5.8.3	PMSoft 软件操作介绍	5-155
5.9	台达定义参数一览表	5-158
5.10	单轴运动功能块说明	5-161
5.10.1	绝对单段速运动	5-161
5.10.2	相对单段速运动	5-164
5.10.3	绝对两段速运动	5-167
5.10.4	相对单段速运动	5-169
5.10.5	插入单段速运动	5-172
5.10.6	插入两段速运动	5-175
5.10.7	寸动运动	5-178
5.10.8	手摇输模式运动	5-180
5.10.9	电子齿轮运动	5-182
5.10.10	原点回归运动	5-184
5.10.11	单轴运动停止	5-186
5.10.12	运动轴参数设定 I	5-187
5.10.13	运动轴参数设定 II	5-189
5.10.14	读取当前位置/速度	5-190
5.10.15	轴状态信息	5-191
5.10.16	设定当前位置	5-193
5.10.17	输入极性设定	5-194
5.11	多轴运动功能块说明	5-195
5.11.1	多轴绝对直线插补	5-195
5.11.2	多轴相对直线插补	5-196
5.11.3	多轴插补停止	5-199
5.12	其它功能块说明	5-201
5.12.1	高速计数器	5-201
5.12.2	高速定时器	5-204
5.12.3	高速比较设定	5-206
5.12.4	高速比较重置	5-208
5.12.5	高速捕捉设定	5-211
5.12.6	高速捕捉遮蔽	5-214
5.12.7	中断设定	5-215

## 第 6 章：多轴插补功能

- 6.1 DVP-10PM 运动模块支持多轴插补 ..... 6-1
- 6.2 TO 指令使用说明 ..... 6-1

## 第 7 章：程序加密设定

- 7.1 使用设定介绍 ..... 7-1
  - 7.1.1 系统信息区 ..... 7-1
  - 7.1.2 下载程序设定 ..... 7-2
  - 7.1.3 上传程序设定 ..... 7-5

## 第 8 章：POU 梯形图编辑模式

- 8.1 POU 梯形图编辑模式环境 ..... 8-1
  - 8.1.1 符号表功能 ..... 8-1
  - 8.1.2 符号表编辑与符号宣告 ..... 8-4
  - 8.1.3 新增 POU 功能块 ..... 8-12
  - 8.1.4 新增 POU 管理目录 ..... 8-15
  - 8.1.5 汇出 POU 功能块 ..... 8-20
  - 8.1.6 汇入 POU 功能块 ..... 8-27
  - 8.1.7 系统装置配置表 ..... 8-31
  - 8.1.8 梯形图寻找 ..... 8-33
  - 8.1.9 梯形图取代 ..... 8-37
- 8.2 POU 梯形图程序操作 ..... 8-37
  - 8.2.1 符号取代装置 ..... 8-37
  - 8.2.2 功能块 FB 使用 ..... 8-39
- 8.3 POU 在线监控 ..... 8-45
- 8.4 符号或功能块自动显示批注(HINT) ..... 8-47

## 第 9 章：CANopen 通讯功能卡

- 9.1 CANOPEN 通讯功能卡 (DVP-FPMC) 简介 ..... 9-1
- 9.2 功能规格 ..... 9-1
- 9.3 产品外观及安装 ..... 9-2
- 9.4 CR 参数功能介绍 ..... 9-3
- 9.5 控制寄存器 CR 内容说明 ..... 9-6
- 9.6 FPMC 之模式设定 ..... 9-21

9.7	FPMC 之 ETHERNET 模式介绍 .....	9-24
9.7.1	ETHERNET 联机介绍 - FPMC 与 HMI 做数据传递范例 .....	9-24
9.7.2	FPMC 与 PMSOFT 联机介绍 .....	9-26
9.8	LED 灯指示说明及故障排除 .....	9-28

## 第 10 章：高速比较与捕捉

10.1	高速比较与捕捉功能 .....	10-1
10.2	高速比较 .....	10-4
10.3	捕捉功能 .....	10-7

## 第 11 章：附录

11.1	附录 A 错误代码原因对照表 .....	11-1
------	----------------------	------

台达 DVP-PM 系列主机为高速定位、双轴线性及圆弧插补多功能可程序控制器，其特色是结合了基本指令、应用指令、运动指令及 G 代码指令等功能，让程序编写设计上更多元化。依照机种，主机支持的程序架构与功能有所不同，下表为各 DVP-PM 机种支持的功能：

功能		DVP-20PM	DVP-10PM
程序区	O100 主程序	○	○
	OX 运动子程序	○	×
	P 子程序	○	○
指令	一般指令/应用指令	○	○
	运动指令	○	×
	G 代码	○	×
	M-Code	○	×
单轴运动	JOG 模式	○	○
	原点回归模式	○	○
	变速模式	○	○
	单段速定位模式	○	○
	插入单段速定位模式	○	○
	兩段速定位模式	○	○
	插入兩段速定位模式	○	○
	单段速外部触发模式	○	○
	手摇輪模式	○	○
	周期/非周期式电子凸輪模式	○	×

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 顺序控制及双轴插补定位控制的功能，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 代码指令将在第 4~6 章作详细介绍说明，本章节将不多作说明。下表为 DVP-PM 机种规格比较：

规格	DVP-10PM	DVP-20PM
高速输出	1000kHz 4 组	500kHz 3 组
PWM	精度 0.3%@200KHz	-
硬件高速计数器	6 组(差分 x2, 开集 极 x4)	2 组
程序容量	64KSteps	64KSteps
执行速度	LD 0.14us MOV 2us DMUL7.6us DEMUL6.1us	-

# 1 PM 程序架构说明

## 1.1 O100 主程序架构

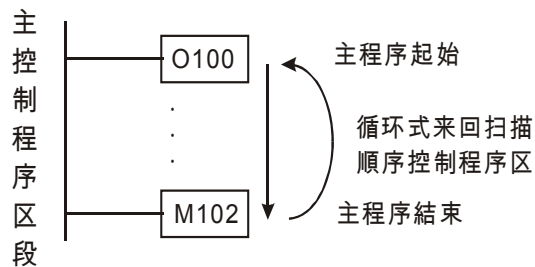
O100 主程序为 PLC 顺序控制程序，主要为 DVP-PM 系列主机运作的顺序控制主程序。O100 主程序区段中，只支持基本指令及应用指令，使用这两种类型指令，除了提供 I/O 点信号数据处理及 Pn 子程序的调用，还有控制启动 OX0 ~ OX99 共 100 组的 OX 运动子程序，所以 O100 主程序为主控程序的建立，再经由主控程序去设定及启动运动子程序，这是 DVP-PM 运作控制架构阶层关系，下列将对 O100 主程序运作流程及特性说明。

### 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 标志将自动为 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 标志设定为 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 起始标志进行扫描，扫描到 M102 主程序结束指令时，再重新返回 O100 起始标志持续的扫描，如下图：



### 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 标志将自动为 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 标志设定为 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 章附录之错误代码原因对照表！

### 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 代码指令，因此请将运动指令及 G 代码规划在 OX0~OX99 运动子程序之中，请参考第 1.2 节说明。
- O100 主程序可调用 Pn 子程序，请参考第 1.3 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 主程序起始标志 (PMSoft 梯形图编辑模式下会自动设定，使用者不需编写)
程序结束	M102 主程序结束指令 (PMSoft 梯形图编辑模式下会自动设定，使用者不需编写)
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 运动子程序及调用 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

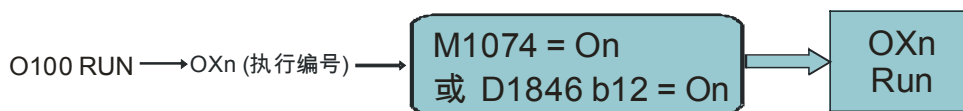
在 O100 主程序内，使用者可利用特殊寄存器，自行规划手动功能运动模式(详细设定方法请参考 3.12 节)

### 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 代码指令，并在程序中可规划调用 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。

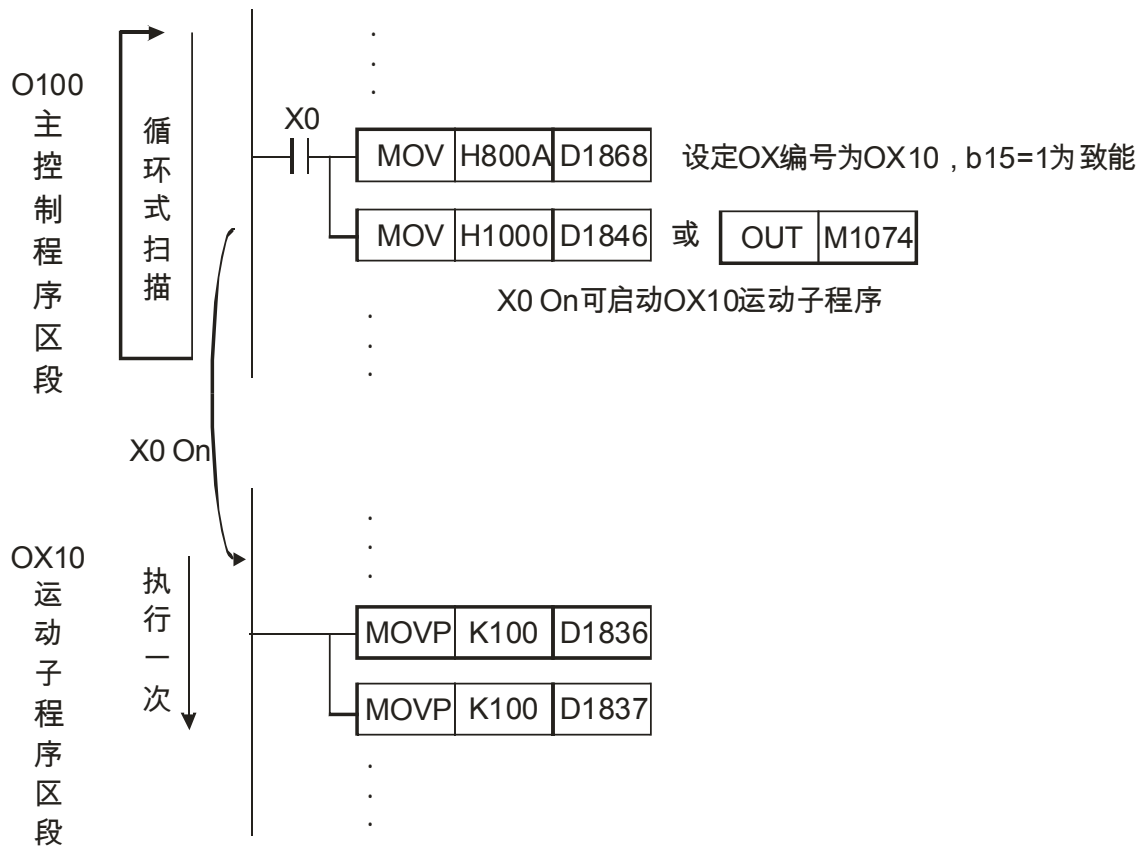
1. 启动 OX0 ~ OX99 运动子程序的方法如下：

- 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 子程序标志 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 运动子程序的起始标志进行执行，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：

# 1 PM 程序架构说明



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 标志将自动为 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。
- 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 章附录之错误代码原因对照表！

### 4. OX 运动子程序支持基本指令、应用指令、运动指令及 G 代码指令四种，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 子程序的调用，请参考第 1.3 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 梯形图编辑模式下会自动设定，使用者不需编写)
程序结束	M2 运动子程序结束 (PMSoft 梯形图编辑模式下会自动设定，使用者不需编写)
程序执行	<ol style="list-style-type: none"> <li>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</li> <li>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</li> <li>3. 可由外部控制端子 (STOP0) 输入信号，控制 OX 运动子程序停止</li> </ol> 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 代码指令，共四种 注意：使用基本指令、应用指令时，请避免使用脉冲型式指令
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	<ol style="list-style-type: none"> <li>1. 为运动子程序 (只能由 O100 主程序设计程序来启动)</li> <li>2. 提供第三轴 (Z 轴) 控制，请参考第 6.4 节之 G00 及 G01 指令！</li> <li>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</li> <li>4. 可调用 Pn 子程序</li> <li>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</li> </ol>

### 1.3 Pn 子程序架构

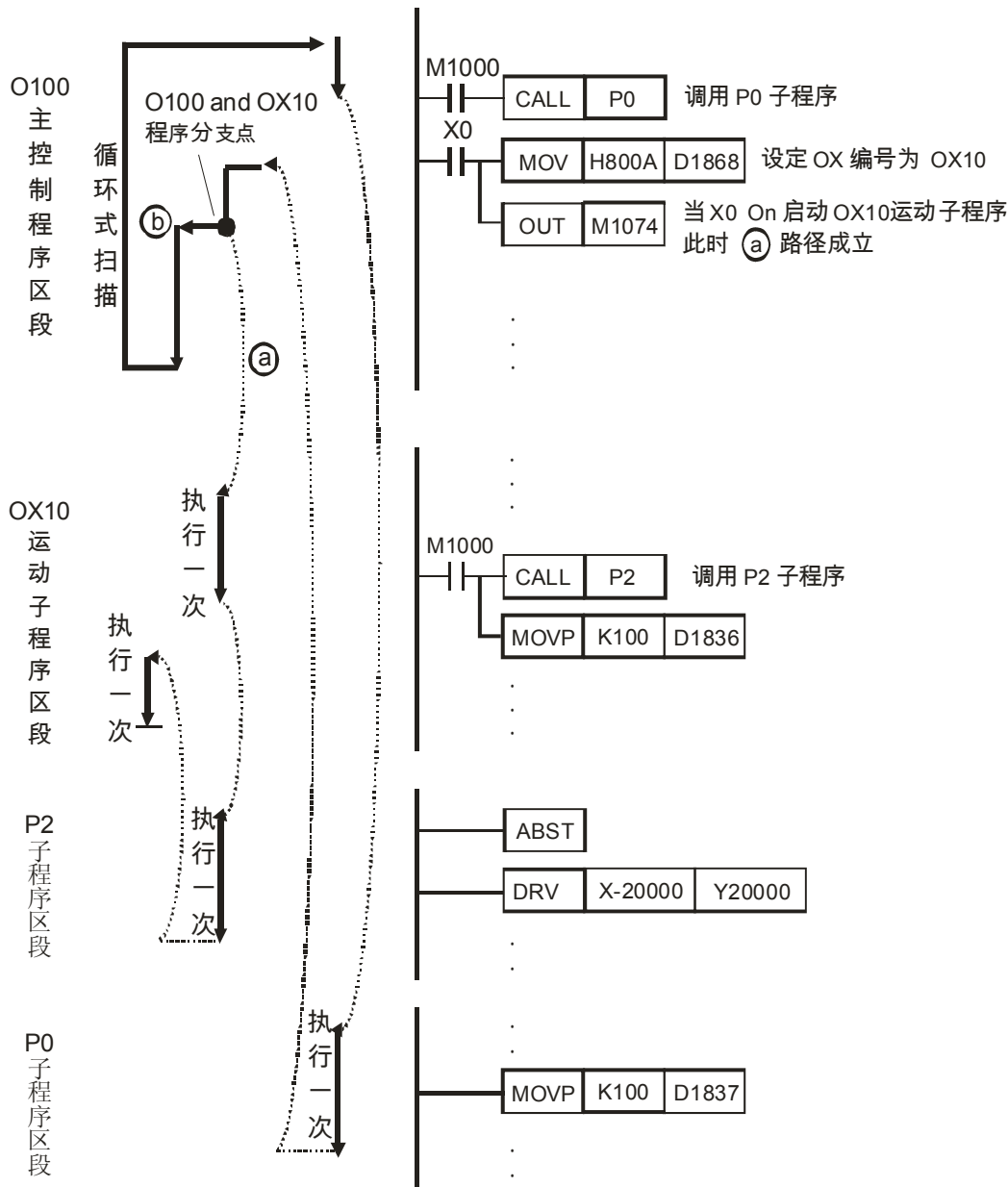
Pn 子程序为一般用子程序，主要为提供 O100 主程序及 OX 运动子程序之调用子程序使用。于 O100 主程序调用 Pn 子程序时，则 Pn 子程序区段支持基本指令及应用指令；若于 OX0 ~ OX99 运动子程序调用 Pn 子程序时，则 Pn 子程序区段支持基本指令、应用指令、运动指令及 G 代码。而不论在 O100 主程序或 OX 运动子程序中调用 Pn 子程序，都会在 Pn 子程序执行时，跳至 Pn 子程序进行动作，执行到 SRET 时将返回 CALL Pn 指令的下一行继续执行。

1. 启动 Pn 子程序的方法如下：

- 于 O100 主程序中调用 Pn 子程序，即可启动 Pn 子程序。
- 于 OX 运动子程序中调用 Pn 子程序，即可启动 Pn 子程序。

2. 其程序扫描方式为调用一次只执行一次，O100 主程序中调用 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 运动子程序调用 Pn 子程序后动作亦同，如下图之范例程序说明：

# 1 PM 程序架构说明



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 代码指令。

### 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 标志将自动为 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，则所调用的 Pn 子程序也停止运作。
- 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 章附录之错误代码原因对照表！

### 4. 于 O100 主程序调用 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 运动子程序调用 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 代码指令。

5. 上述说明整理如下表：

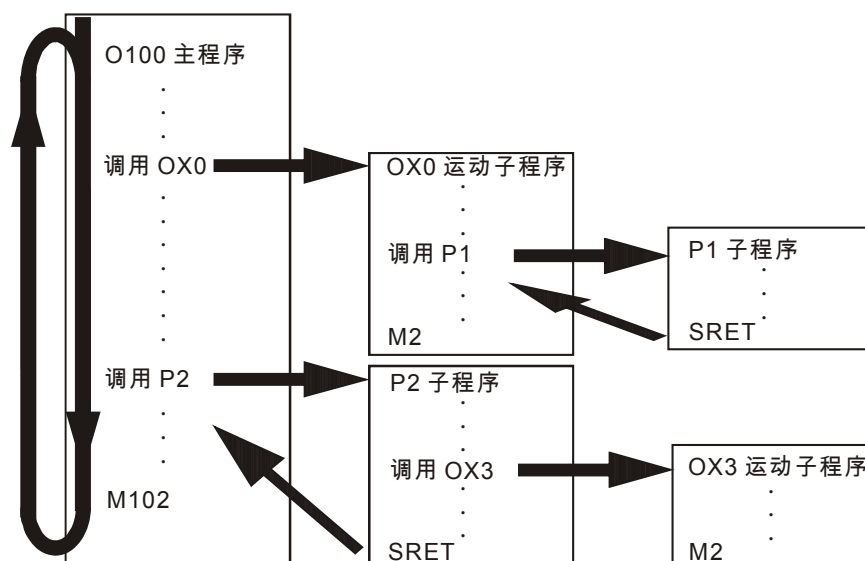
Pn 子程序	说明
程序起始	Pn 子程序起始标志 (P0 ~ P255) (PMSoft 梯形图编辑模式下会自动设定，使用者不需编写)
程序结束	SRET 子程序结束 (PMSoft 梯形图编辑模式下会自动设定，使用者不需编写)
程序执行	1. 由 O100 主程序中，调用 Pn 子程序 2. 由 OX 运动子程序中，调用 Pn 子程序
运作特性	每调用一次只执行一次，若须再次执行可再重新调用一次即可
支持指令	1. 由 O100 调用时：支持基本指令及应用指令，共二种 2. 由 OX 调用时：支持基本指令、应用指令、运动指令及 G 代码指令，共四种 注意：由 OX 调用且使用基本指令、应用指令时，请避免使用脉冲型式指令。
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 运动子程序调用用 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 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：



# 1 PM 程序架构说明

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：



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，且不可被其它程序调用，但可自由调用 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 子程序调用，并且可以调用 Pn 子程序。
4. Pn 子程序可被 O100 主程序及 OX 运动子程序调用，并且可以调用 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：有支持，X：不支持)

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 代码	X	O	O	X
说明	支持指令为内定	支持指令为内定	由 OX 运动子程序调用，故有支持运动指令、G 代码	由 O100 主程序调用，故不支持运动指令、G 代码

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	主程序或运动程序调用	主程序或子程序调用
运作特性	循环运作	调用一次执行一次	调用一次执行一次
数量	1	256 个，使用者需求使用	100 个，使用者需求使用

MEMO



### 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，请参考第 5~6 章，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	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

##### ■ DVP10PM00M 四轴模式

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 信号：X0、X2、X4、X6 3. X-Y-Z-A 四轴 PG 信号：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

##### ■ DVP10PM00M 六轴模式

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 信号：X0、X2、X4、X6 3. X-Y-Z-A 四轴 PG 信号：X1、X3、X5、X7	200kHz	15mA	24V

## 2 硬件规格及配线方式

端子	说明	响应特性	最大输入	
			电流值	电压值
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-	1. 差动信号计数输入 2. B-C 两轴 DOG 信号： (X12+、X12-)、(X13+、X13-) 3. 零点信号需另搭配 I/O 扩展模块	200kHz	15mA	5~24V

### 输出点电气规格

#### ■ DVP10PM00M 四轴模式

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) 零点信号	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

#### ■ DVP10PM00M 六轴模式

端子	说明	响应特性	最大电流输出
Y0、C0 Y2、C2	开集极高速脉冲输出 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	200kHz	40mA
Y1、C1 Y3、C3	开集极高速脉冲输出 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	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

#### ■ 10PM

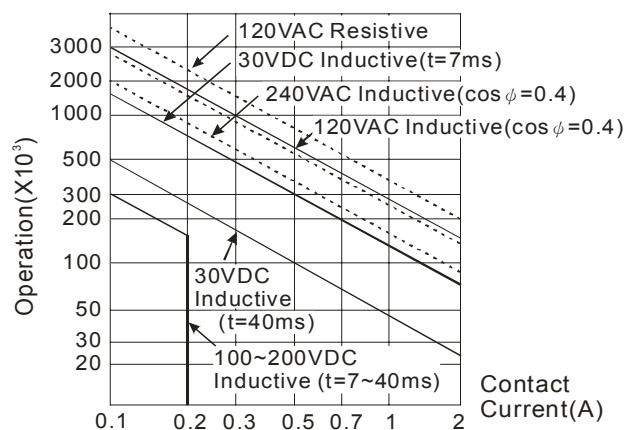
规格 \ 项目		双端差动输入	24VDC 单端共点输入	备注
		高速 200kHz		
输入接线型式		独立接线	由端子 S/S 变换接线为 SINK 或 SOURCE	#1 : 输入点可作 10 ~ 60 ms 数字滤波调整 ; 另外可作 20~85KHz 之高频数字滤波调整。
输入动作指示		LED 显示, 灯亮表示为'On', 不亮表示为'Off'		
输入信号电压		5~24 VDC	24 VDC	
最大电流输入值		15 mA		
动作标准	Off→On	20us		
	On→Off	30us		
反应时间/噪声抑制 #1		10ms / 0.5us		

### 数字输出点

#### ■ 10PM

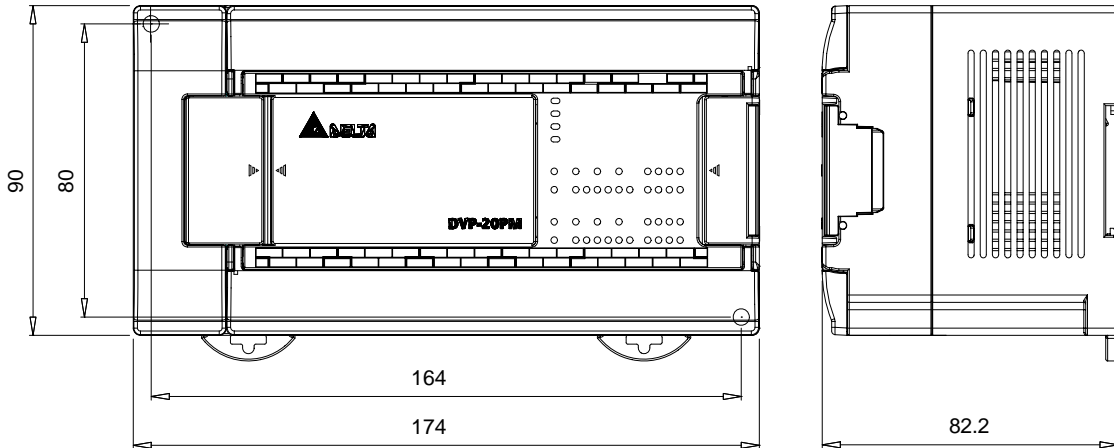
规格 \ 项目		双端差动输出	单端共点晶体管输出
		最高交换 (工作) 频率	
输出动作指示		LED 显示, 灯亮表示为'On', 不亮表示为'Off'	
输出点配置		Y10 ~ Y17	Y0 ~ Y3
工作电压		5 VDC	5 ~ 30 VDC
最大输出电流值		40 mA	40 mA
隔离方式		驱动级电源隔离	光耦合隔离
电流规格	阻抗	< 25 mA	0.5 A/1 点 ( 4 A/COM )
	感抗	--	12 W (24 VDC)
	灯泡	--	2 W (24 VDC)
最大输出延时时间	Off→On	0.2 us	
	On→Off		
输出过电流保护		N/A	

#2 : 生命周期曲线图



## 2 硬件规格及配线方式

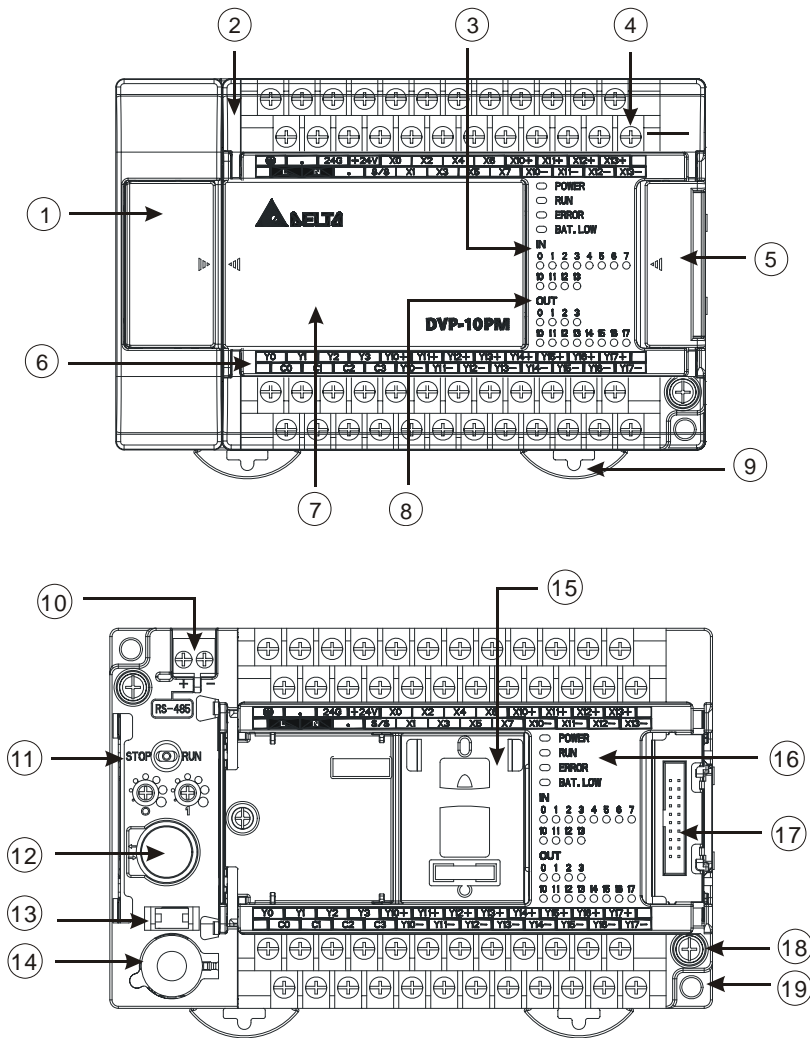
### 2.1.3 外观尺寸规格



(单位：mm)

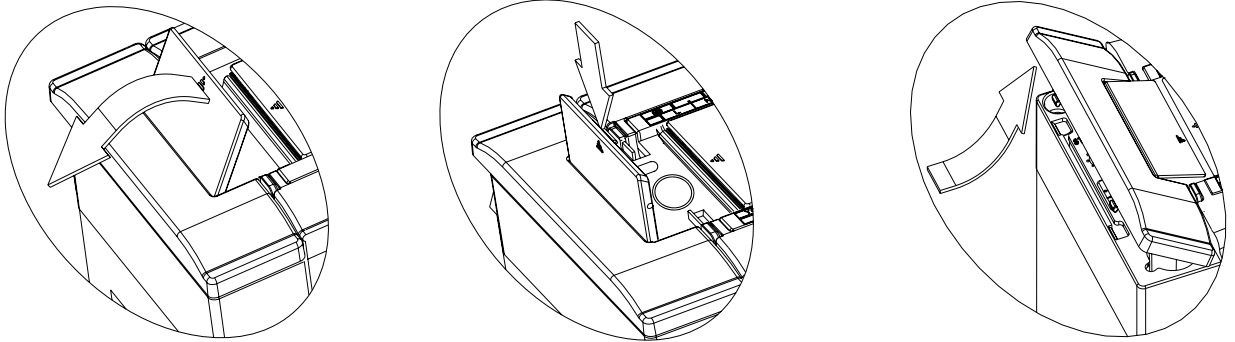
### 各部介绍

#### ■ DVP-10PM



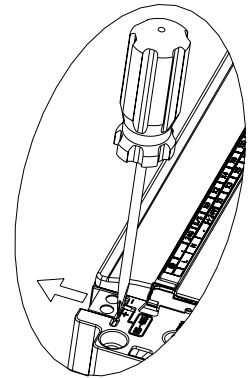
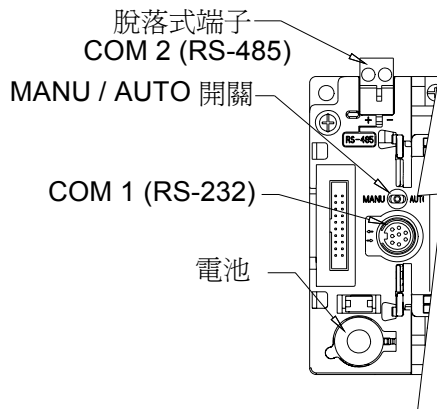
- ① 通讯端口上盖
- ② 输入/输出端子盖
- ③ 输入点指示灯
- ④ 输入/输出端子
- ⑤ I/O 模块接口上盖
- ⑥ 输入/输出端子编号
- ⑦ 功能卡/存储卡上盖
- ⑧ 输出点指示
- ⑨ DIN 轨固定扣
- ⑩ COM2 通讯端口 (RS-485)
- ⑪ STOP / RUN 开关
- ⑫ COM1 通讯端口 (RS-232)
- ⑬ 电池插槽
- ⑭ 电池
- ⑮ 功能卡插槽
- ⑯ 电源、运行、错误及电池状态
- ⑰ I/O 模块接口
- ⑱ 机身固定螺丝
- ⑲ 直接固定孔

COM1 上盖开启



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 节)

### ■ DVP-10PM

⊕	•	24G	+24V	X0	X2	X4	X6	X10+	X11+	X12+	X13+
<b>L N</b>		•	S/S	X1	X3	X5	X7	X10-	X11-	X12-	X13-
<b>DVP-10PM</b> (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 硬件规格及配线方式

### 2.2 配线方式

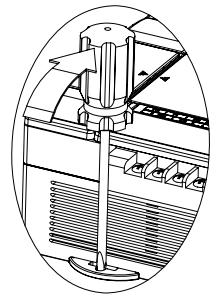
本机为开放型 (OPEN TYPE) 机壳，因此使用者使用本机时，必须将之安装于具防尘、防潮及免于电击/冲击意外之外壳配线箱内。另必须具备保护措施 (如：特殊之工具或钥匙才可打开) 防止非维护人员操作或意外冲击本体，造成危险及损坏。

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。

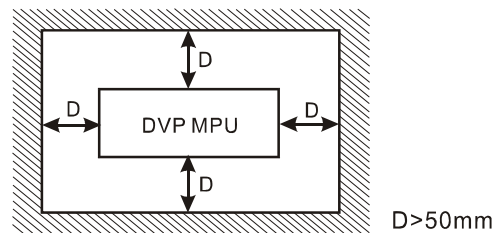
#### 2.2.1 盘内安装配线

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：

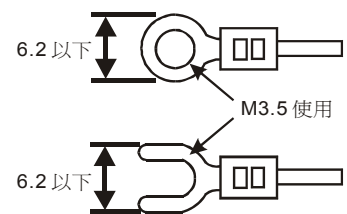


1. 直接锁螺丝方式：请依产品外型尺寸并使用 M4 螺丝。
2. DVP-PM 在安装时，请装配于封闭式之控制箱内，其周围应保持一定之空间，(如右图所示)，以确保 DVP-PM 散热功能正常。



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。输入点信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

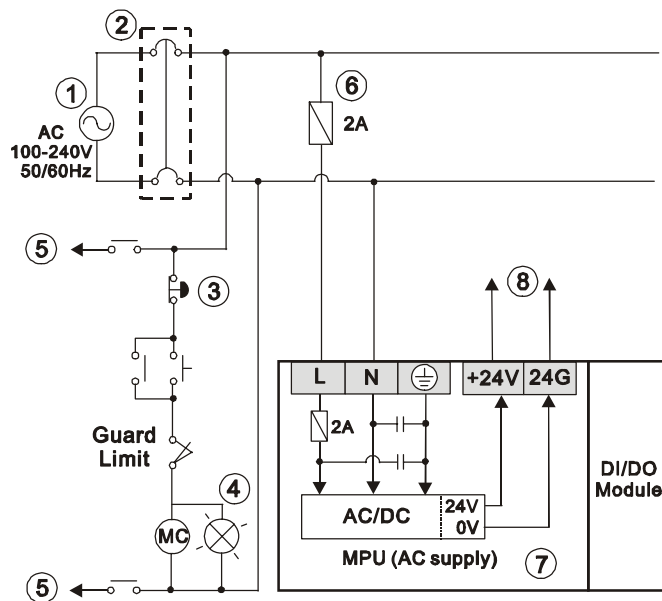
DVP-PM 系列主机电源输入为交流输入，在使用上应注意下列事项：

1. 交流电源输入电压，范围宽 (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 内部具有停电保持的辅助继电器及寄存器，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

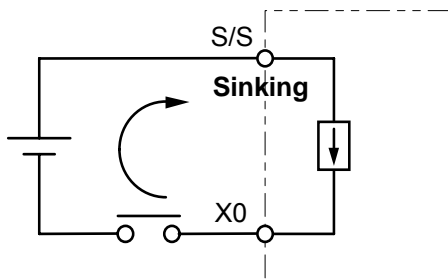
## 2 硬件规格及配线方式

### 2.2.4 输入/输出接点配线

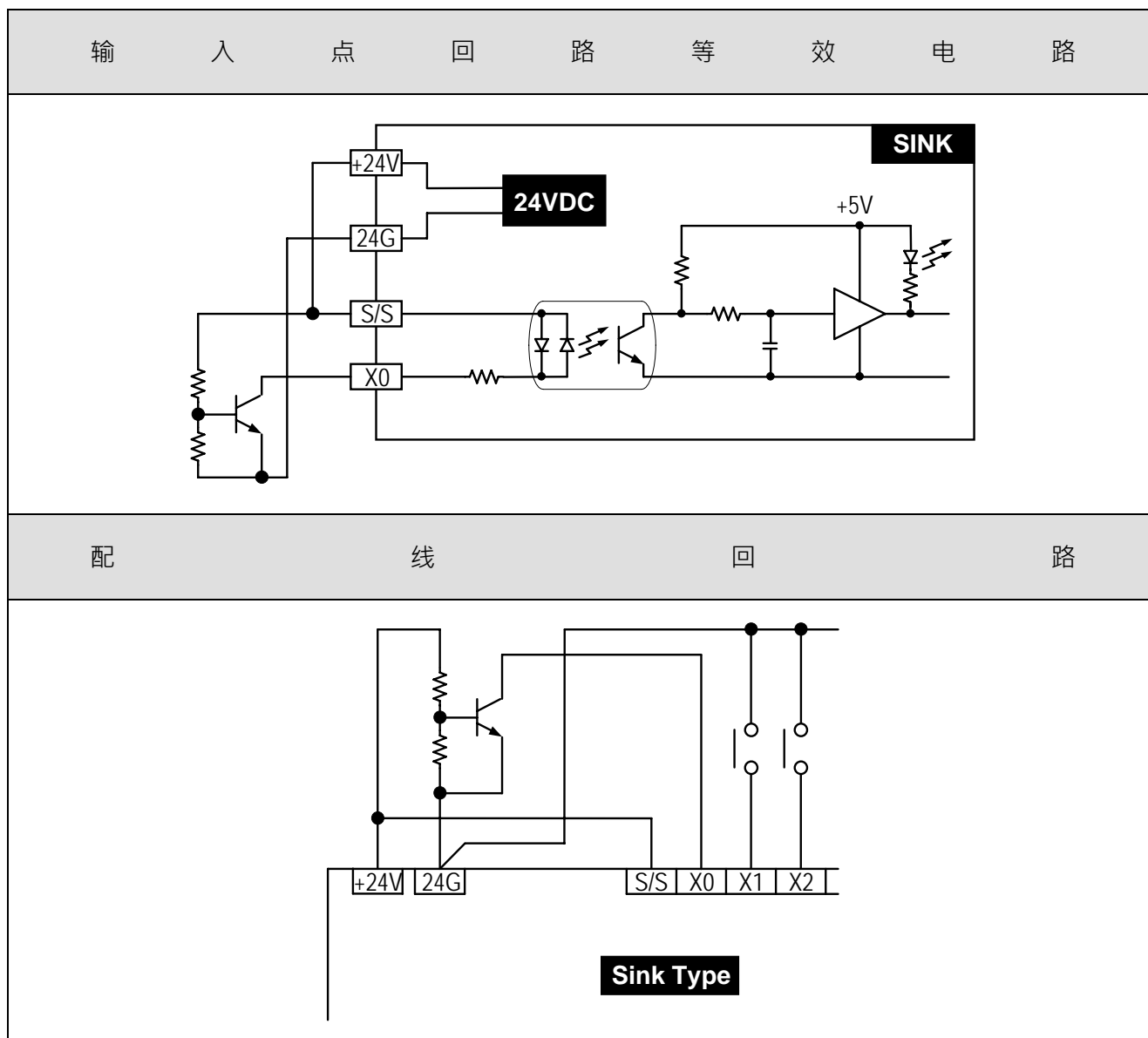
1. 输入点之入力信号为直流电源 DC 输入，DC 型式共有两种接法：

SINK 定义如下：

直流形式 (DC Signal IN)



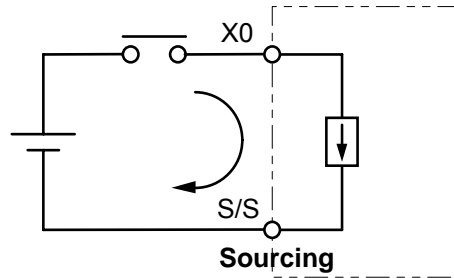
SINK 模式 (电流流入共享端 S/S)



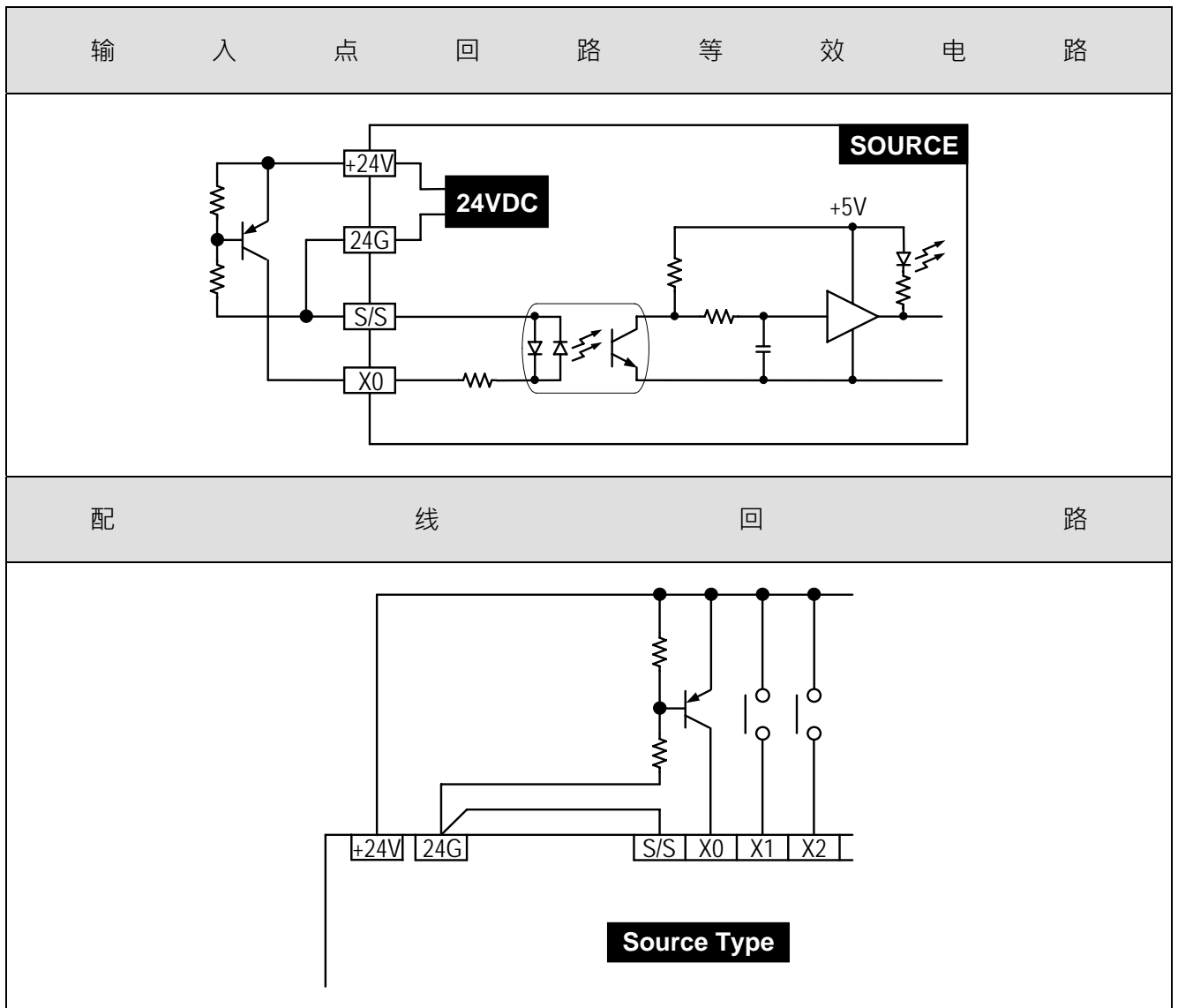


SOURCE 定义如下：

直流形式 (DC Signal IN)



SOURCE 模式 (电流流出共享端 S/S)

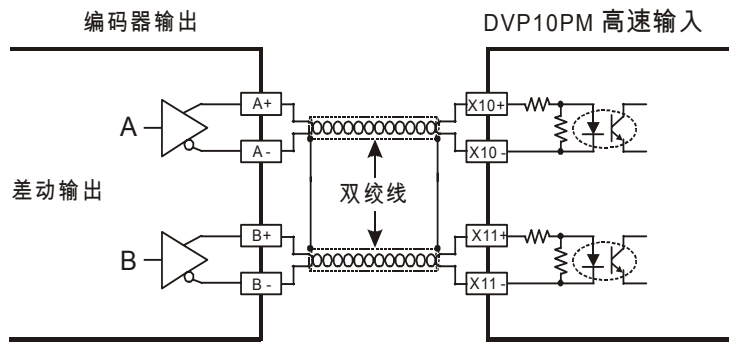


## 2. 差动输入之配线

DVP-10PM 之 X10 ~ X13 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。

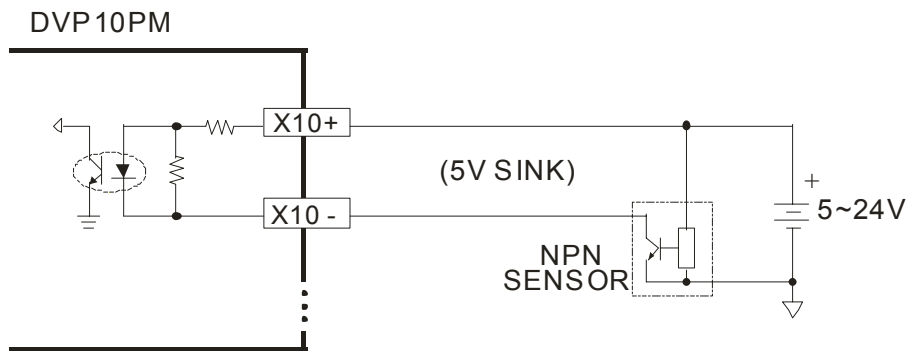
## 2 硬件规格及配线方式

差动输入之接线图 (高速、高噪声时使用) :

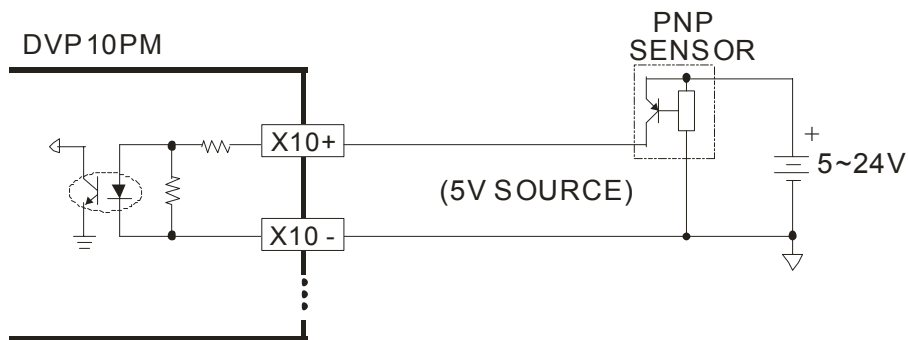


若工作环境频率不高 (小于 50kHz) 且噪声较低, 亦可使用 DC5 ~ 24V SINK / SOURCE 之单端输入方式。DVP-10PM DC5V ~ 24V SINK 与 SOURCE 输入之接线图, 如下所示:

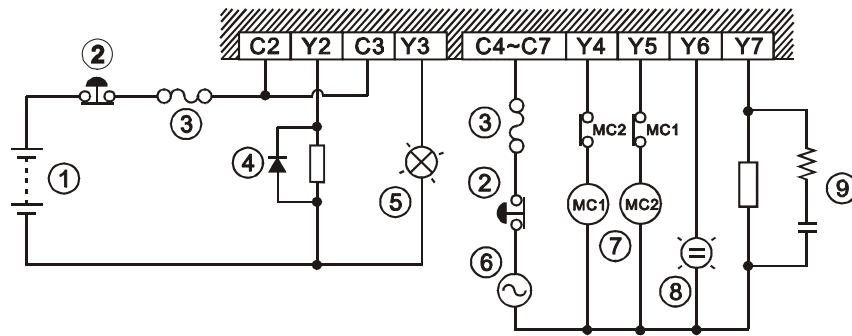
DVP10PM DC5V SINK 输入之接线图:



DVP10PM DC5V SOURCE 输入之接线图:



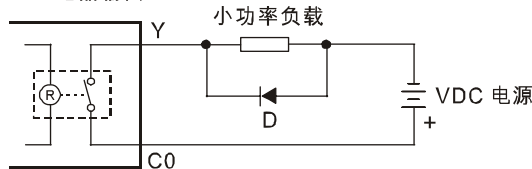
### 3. 继电器 (R) 输出回路配线



- 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。

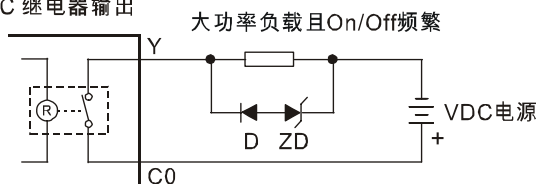
PLC 继电器输出



D: 1N4001 二极管或等效元件

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

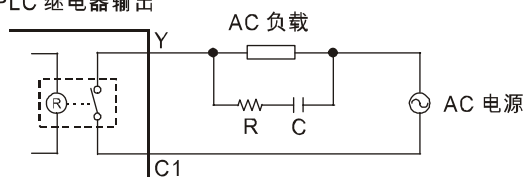
PLC 继电器输出



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

- ⑤ 白炽灯 (阻抗负载)。
- ⑥ 交流电源供给。
- ⑦ 互锁输出：例如，将 Y4 与 Y5 用以控制对应电机的正转及反转，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。

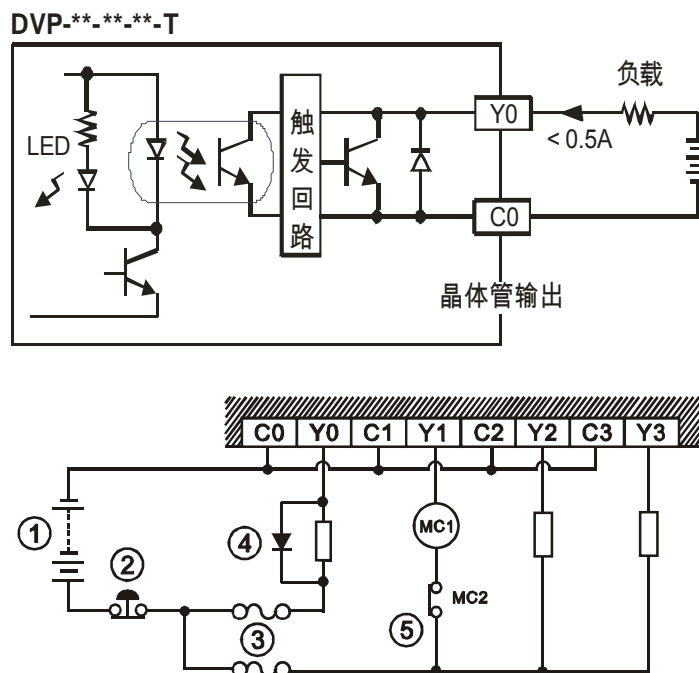
PLC 继电器输出



R: 100~120 Ω  
C: 0.1~0.24 μF

## 2 硬件规格及配线方式

### 4. 晶体管 (T) 输出回路配线



① 直流电源供应。

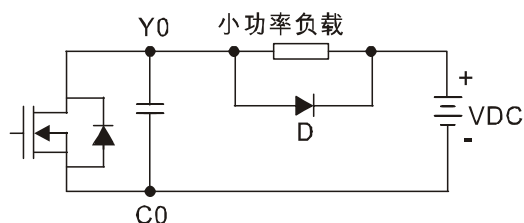
② 紧急停止。

③ 电路回路保护用保险丝。

④ 因晶体管模块输出均为开集极输出 (Open Collector)，若 Y0/Y1 设定为脉冲输出，为确保晶体管模块能够动作正常，必须维持经提升电阻的输出电流大于 0.1A。

1. 二极管抑制：功率较小时使用。

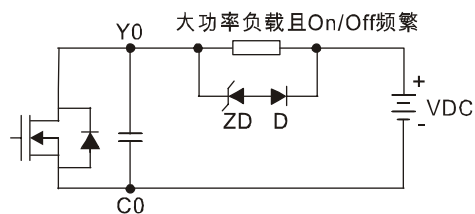
PLC 晶体管输出



D: 1N4001 二极管或等效元件

2. 二极管+Zener 抑制：大功率且 On/Off 频繁时使用。

PLC 晶体管输出

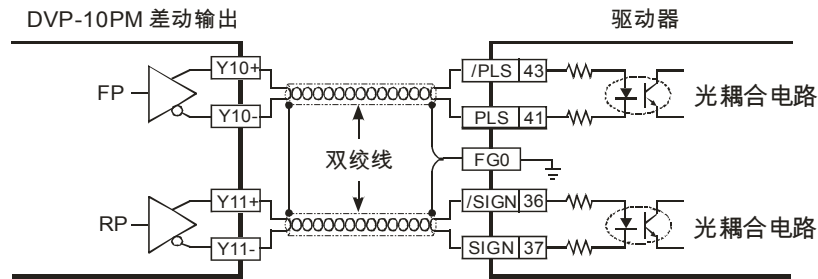


D: 1N4001 二极管或等效元件

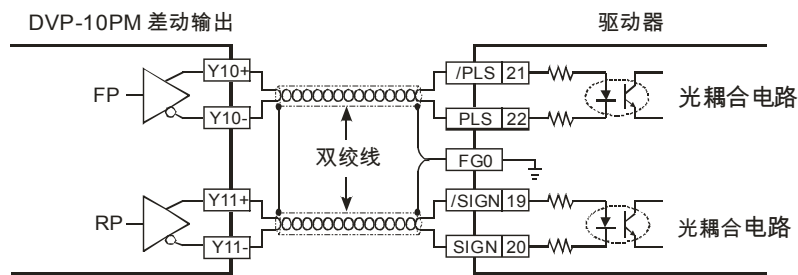
⑤ 互锁输出：例如，将 Y4 与 Y5 用以控制对应电机的正转及反转，使外部电路形成互锁，配合 PLC 内部程序，确保任何异常突发状况发生时，均有安全的保护措施。

### 5. 差动输出之接线图

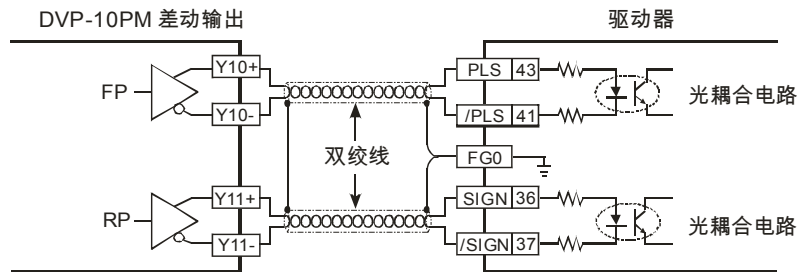
- DVP-10PM 差动输出与 ASDA-A、ASDA-A+ 及 ASDA-A2 系列驱动器



- DVP-10PM 差动输出与 ASDA-B 系列驱动器



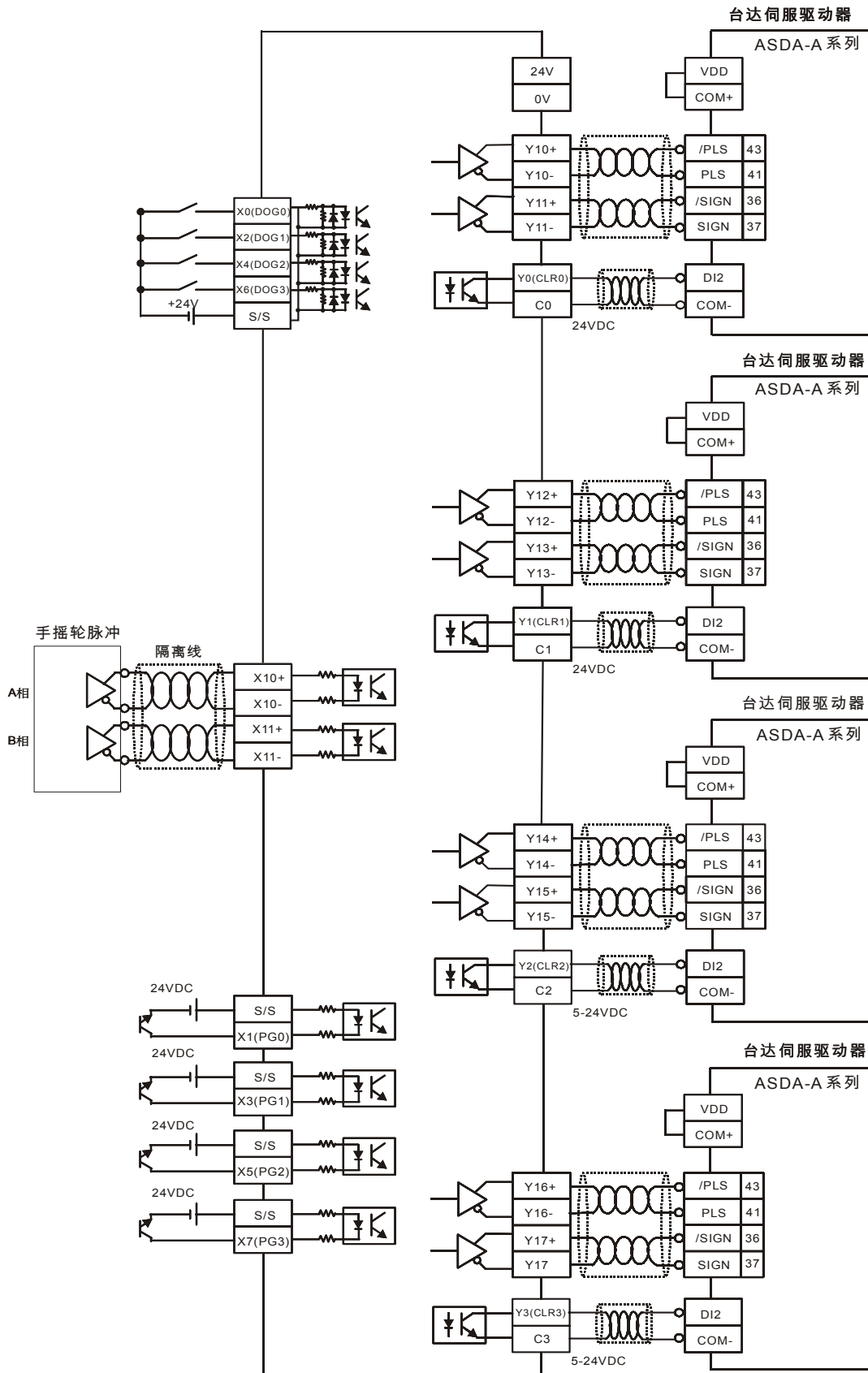
- DVP-10PM 差动输出与 ASDA-AB 系列驱动器



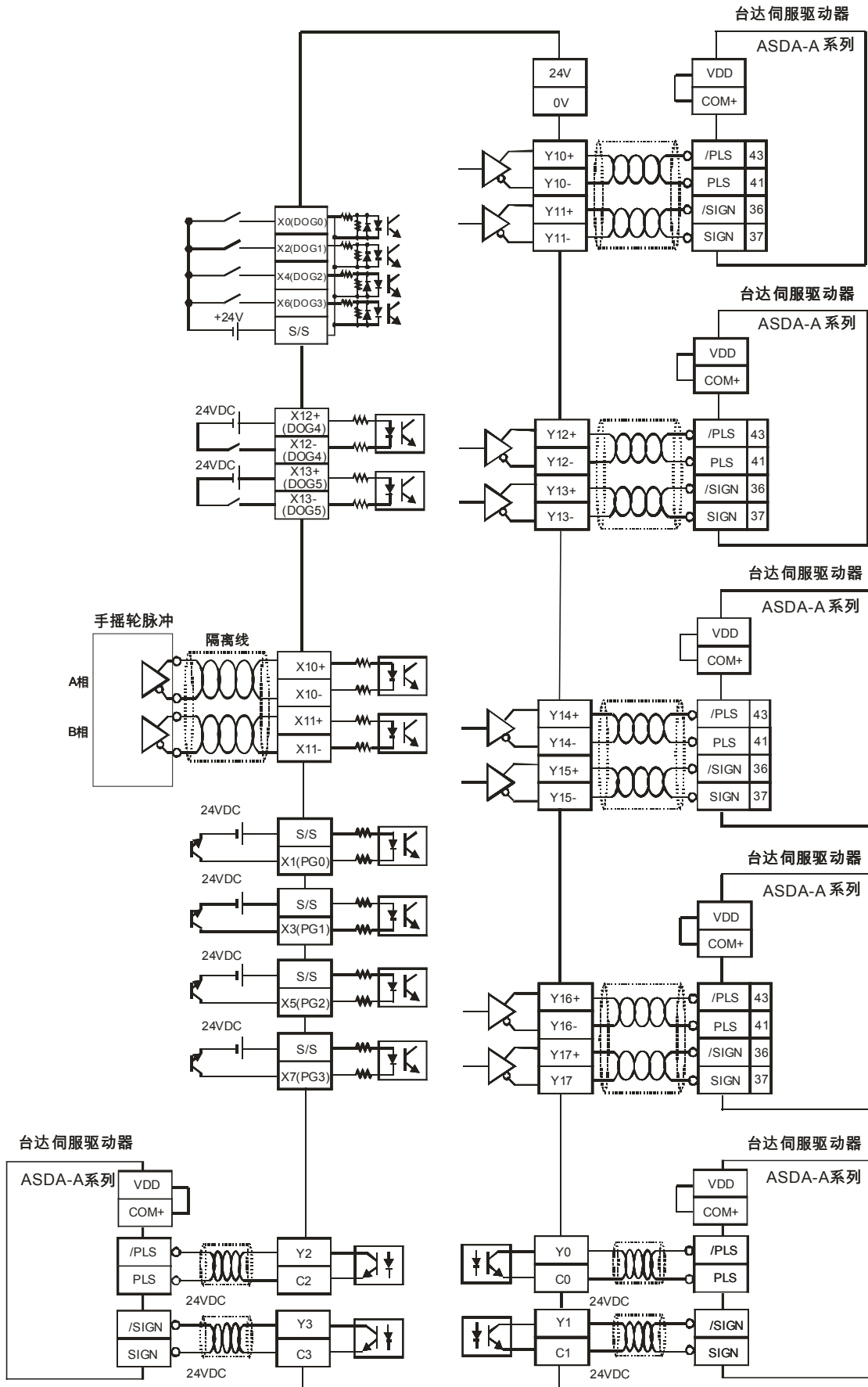
## 2 硬件规格及配线方式

### 2.2.5 与下位驱动器之配线

DVP-10PM 与台达 ASDA-A 系列伺服驱动器之四轴配线图：

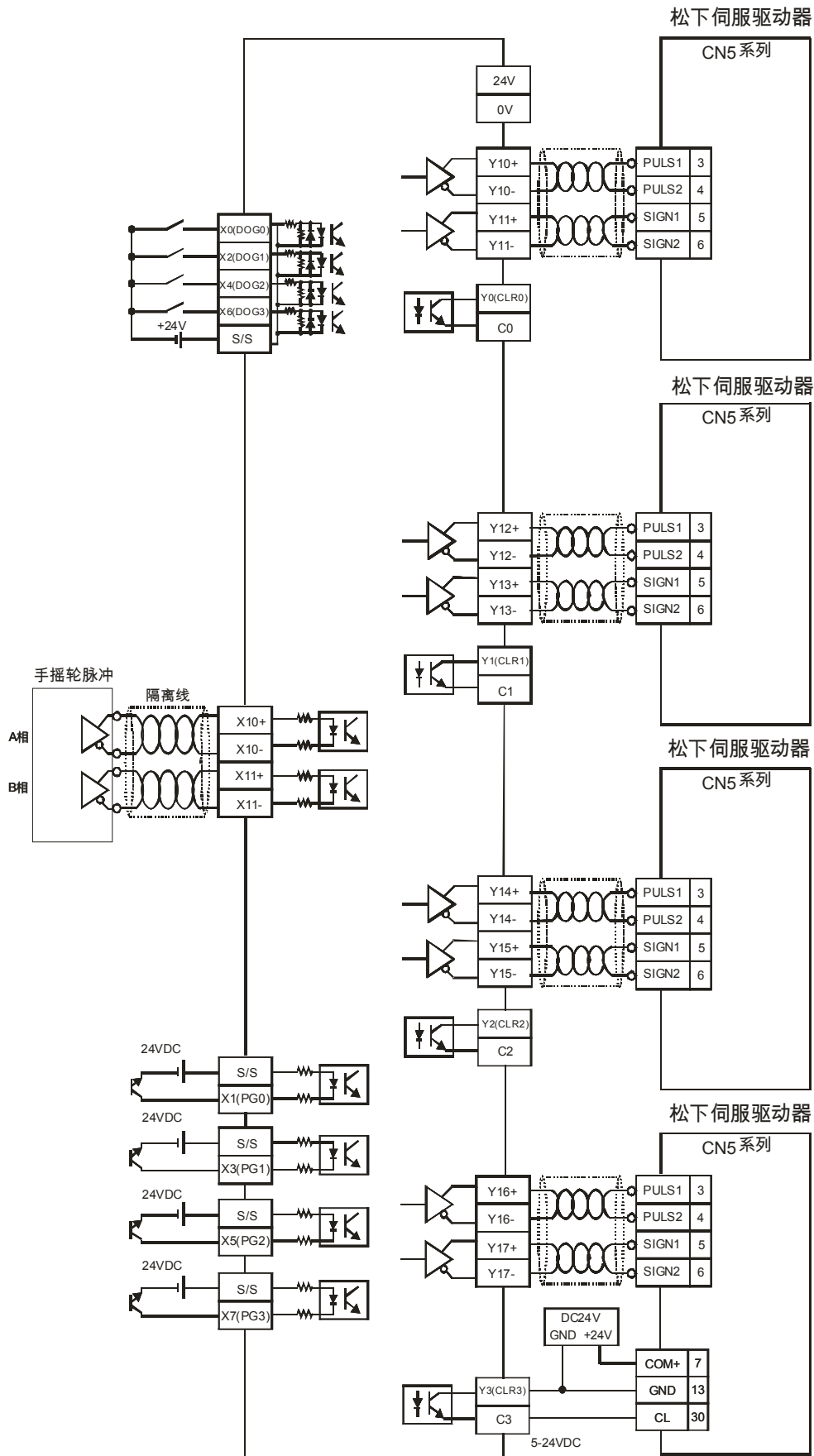


DVP-10PM 与台达 ASDA-A 系列伺服驱动器之六轴配线图：



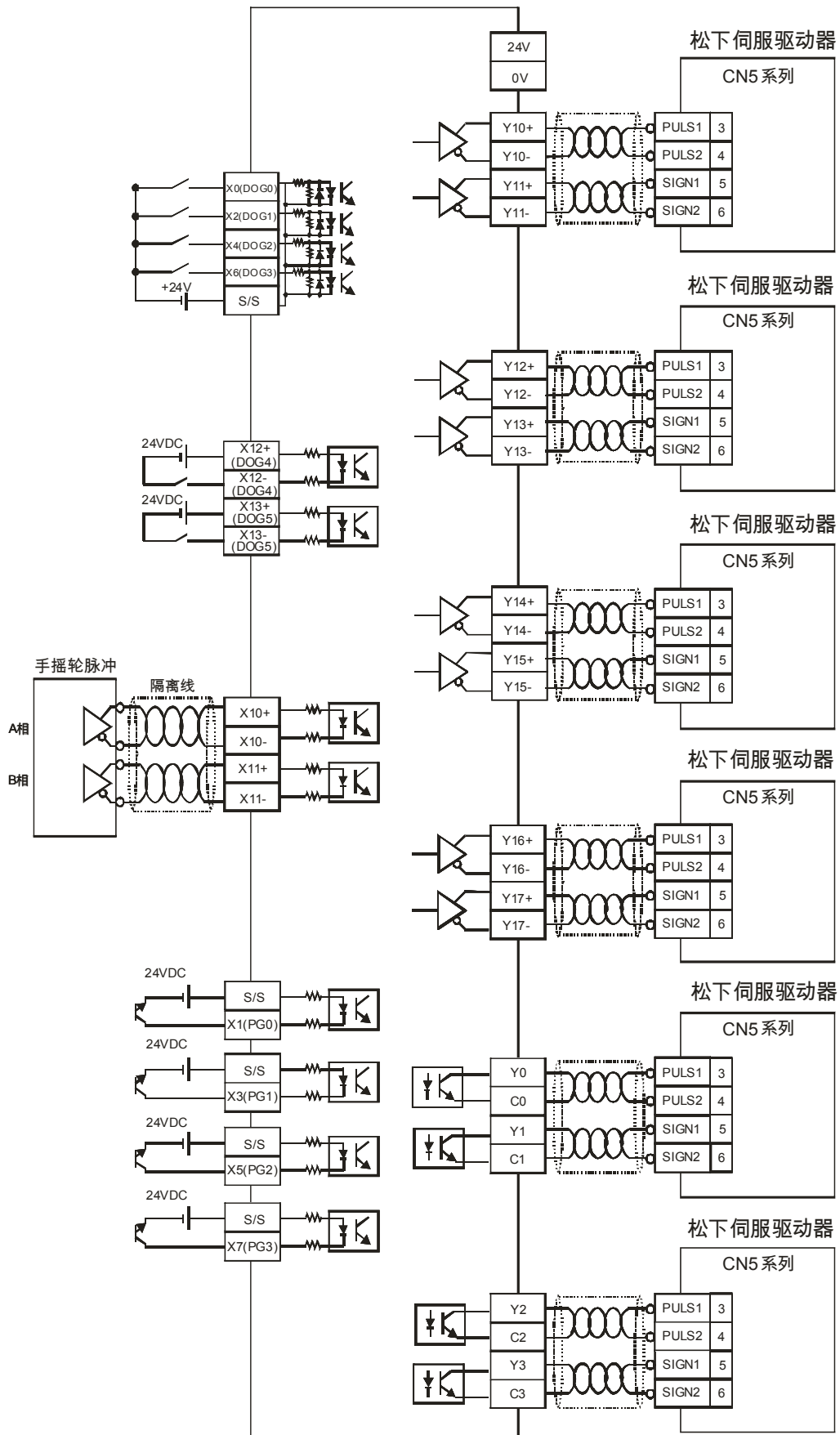
## 2 硬件规格及配线方式

DVP-10PM 与松下 CN5 系列伺服驱动器之四轴配线图：



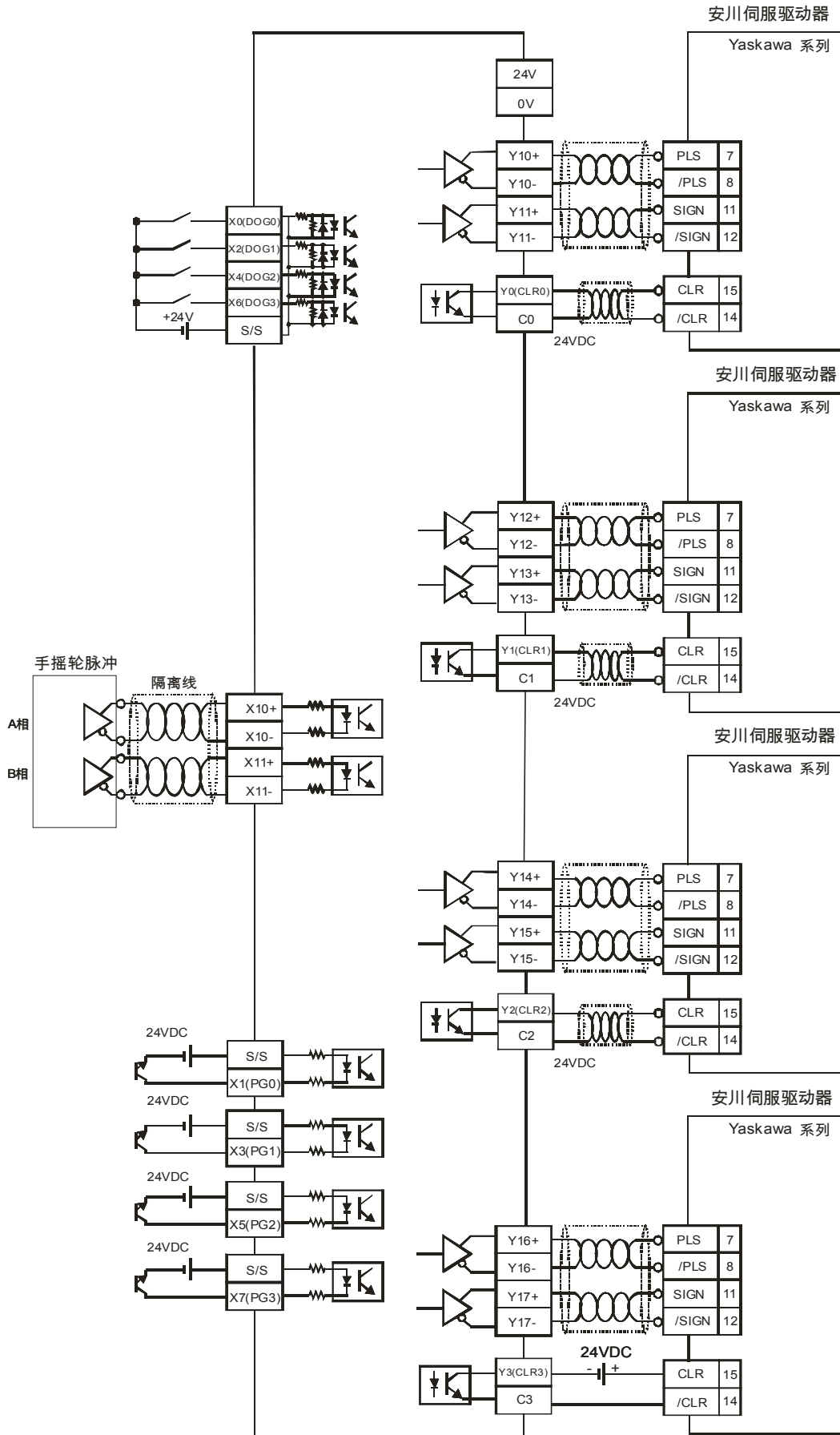


DVP-10PM 与松下 CN5 系列伺服驱动器之六轴配线图：

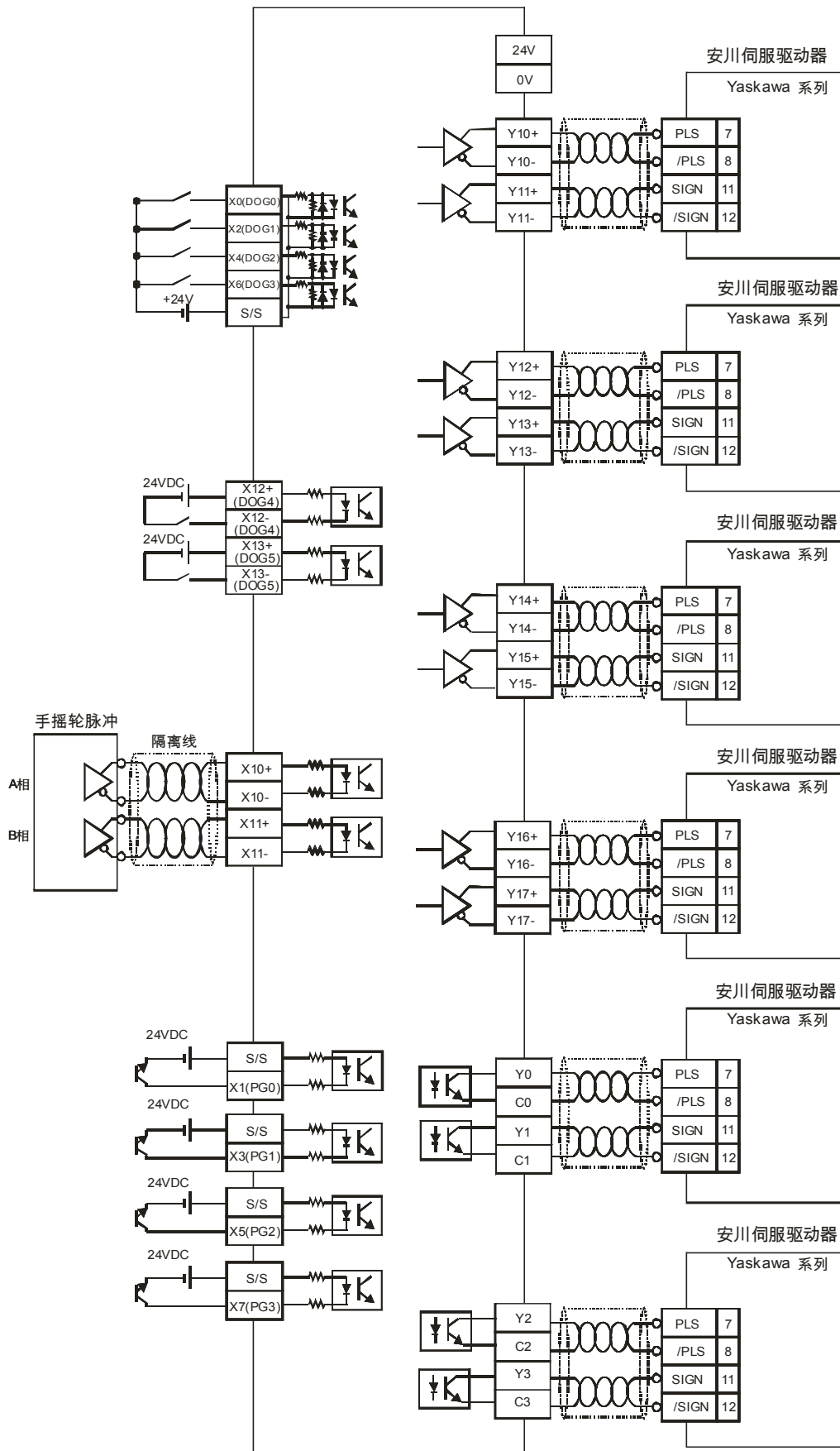


## 2 硬件规格及配线方式

DVP-10PM 与安川伺服驱动器之四轴配线图：

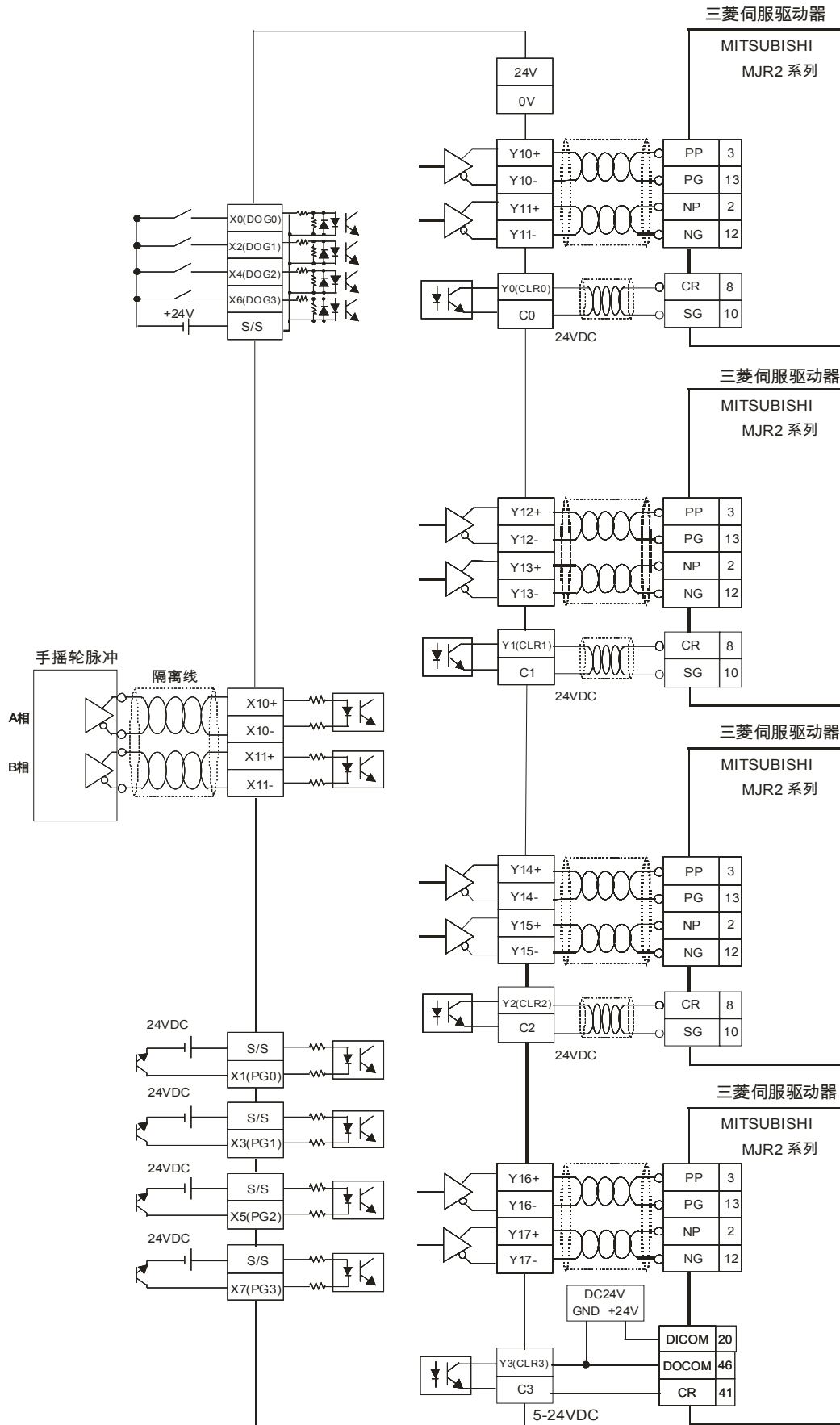


DVP-10PM 与安川伺服驱动器之六轴配线图：

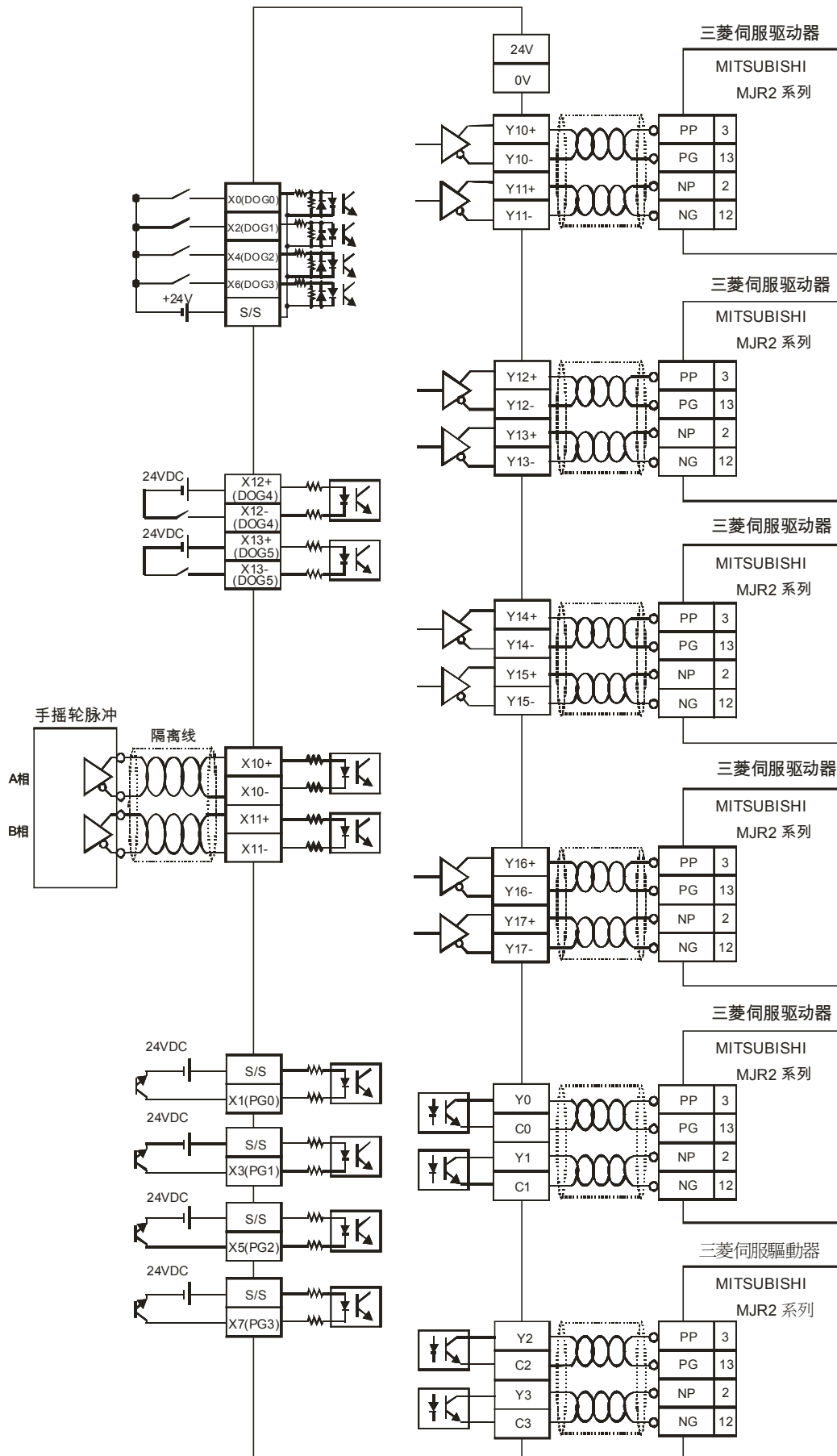


## 2 硬件规格及配线方式

DVP-10PM 与三菱 MJR2 系列伺服驱动器之四轴配线图：

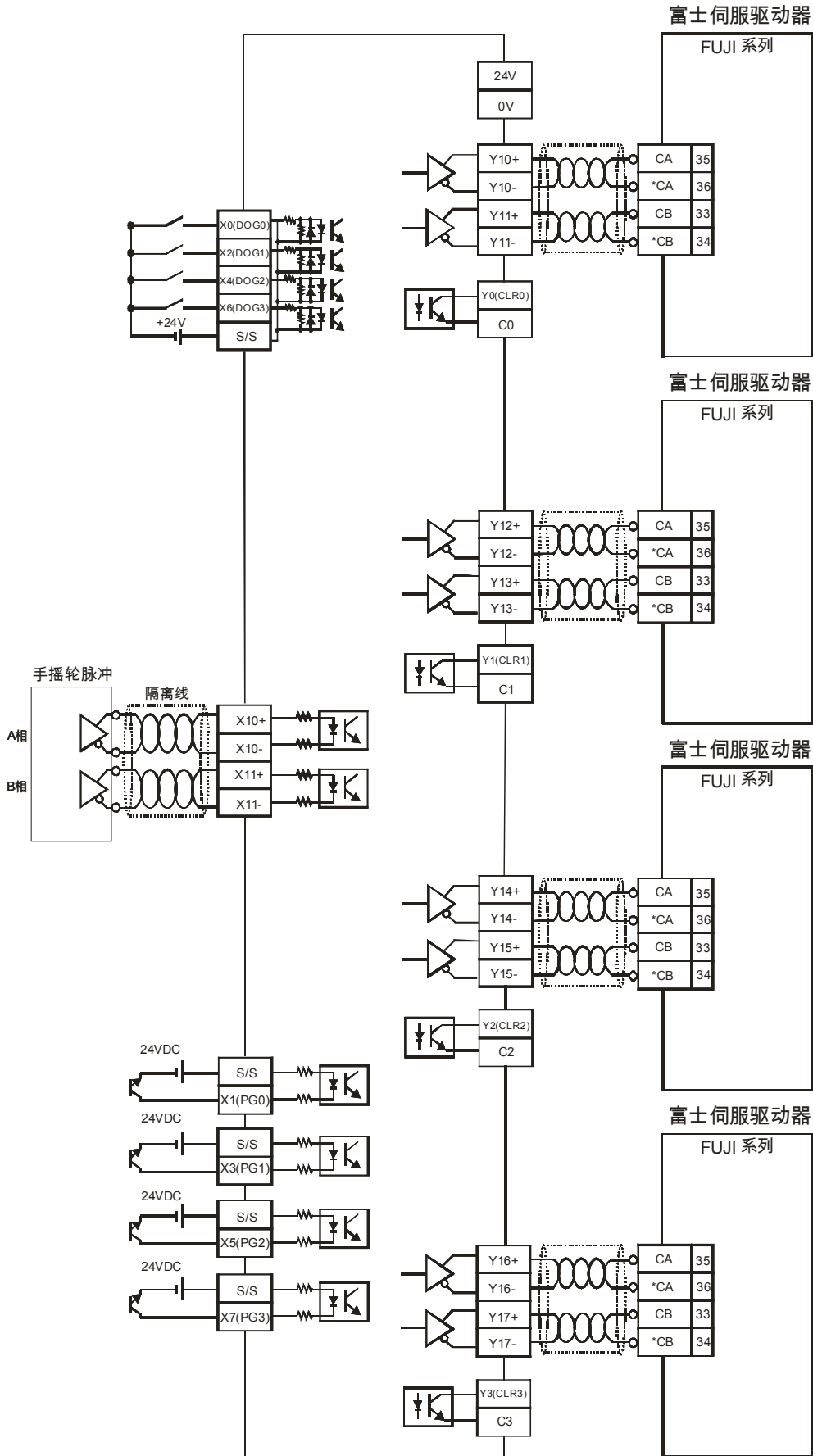


DVP-10PM 与三菱 MJR2 系列伺服驱动器之六轴配线图：

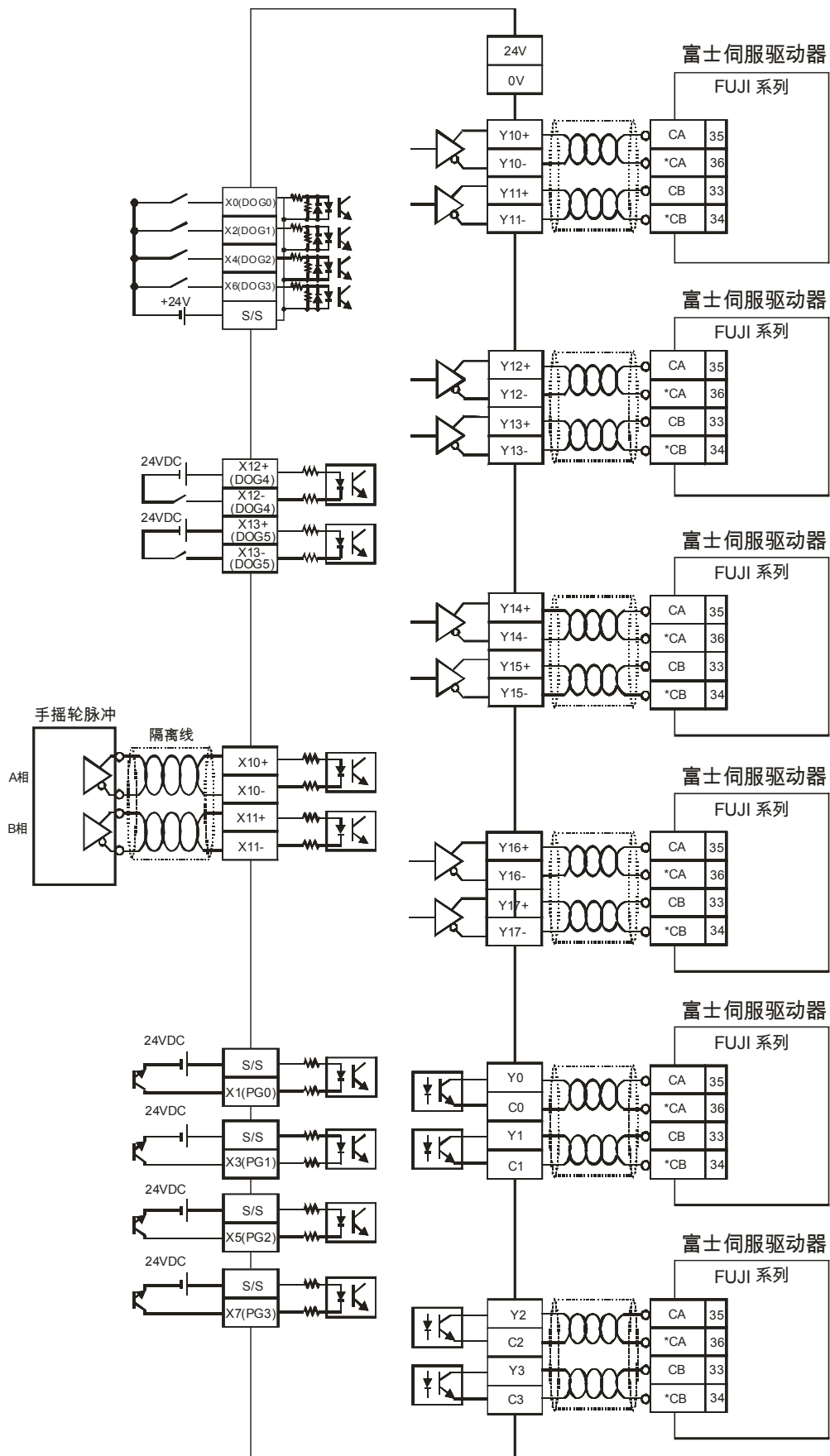


## 2 硬件规格及配线方式

DVP-10PM 与富士伺服驱动器之四轴配线图：



DVP-10PM 与富士伺服驱动器之六轴配线图：



## 2 硬件规格及配线方式

### 2.3 通讯端口说明

DVP-PM 系列主机的通讯端口有内建 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		
通讯格式设定寄存器	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 个寄存器		32 个寄存器
读写数据长度 (RTU 模式)	100 个寄存器		32 个寄存器

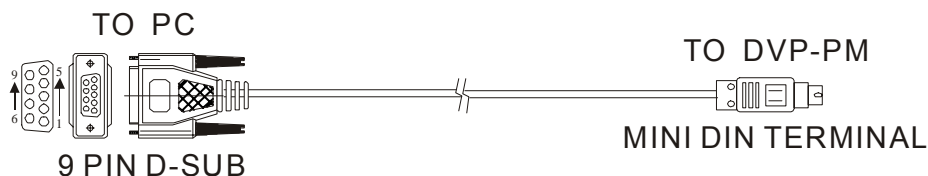
通讯端口的预设通讯格式

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

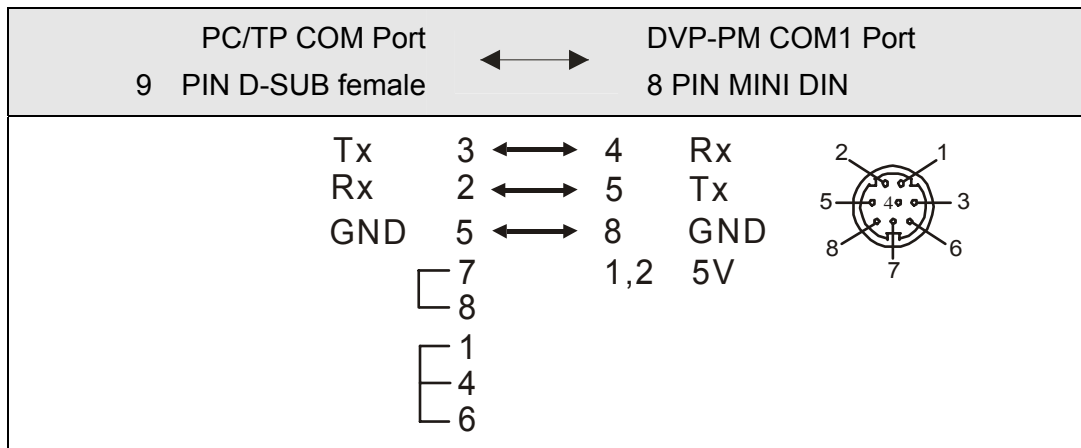
#### 2.3.1 COM1 (RS-232)

1. 通讯端口 COM1 为 RS-232 接口之通讯端口, 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. 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)

1. 通讯端口 COM2 为 RS-485 接口之通讯端口，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。
2. 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)

1. 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。
2. 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。

## 2 硬件规格及配线方式

---

MEMO

## 3.1 各装置编号一览表

### ■ 功能规格

项 目		规格			备注
轴运作方式		六轴同动及独立六轴控制			
程序储存		内存 64k Steps 储存器			
单位系		电机单位	复合单位	机械单位	
主机读写 I/O 模块方式		不支持			
与主机串接说明		不支持			
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 1000k PPS 插补轴最快 1000k PPS			
输入信号	操作开关	STOP/RUN (自动/手动选择开关)			
	差动输入信号	X10+、X10-、X12+、X12- X11+、X11-、X13+、X13-			
	检知器	X0~X7, 可扩展 I/O 模块, 最大扩展点数含主机 256 点			
输出信号	差动输出信号	Y10+、Y10-、Y12+、Y12-、Y14+、Y14-、Y16+、Y16- Y11+、Y11-、Y13+、Y13-、Y15+、Y15-、Y17+、Y17-			
	一般输出点	Y0~Y3, 可扩展 I/O 模块, 最大扩展点数含主机 256 点			
	串联通讯端口	程序写入/读出通讯端口 : 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	支持以下 EH2 系列之特殊右侧模块 : AD、DA、PT、TC、XA、PU (最大可扩展 8 台, 不占 I/O 点数)			
特殊功能卡	选购	支持以下功能卡 : 02AD、02DA、COM3			
基本指令		27 个			
应用指令		130 个			
运动指令		-			
M -Code		<ul style="list-style-type: none"> <li>■ OX0 ~ 99 (运动子程序/Positioning Program) : M02 程序停止 (END)</li> <li>■ M00 ~ M01, M03 ~ M101, M103 ~ M65535 做为程序暂停 (WAIT), 可自由使用</li> <li>■ O100 (DVP-PM 主程序/Sub-task Program) : M102 程序停止 (END)</li> </ul>			
G-Code		不支持			
自我诊断		参数错误、程序错误、外部错误等各种错误显示			

### 3 各种装置功能

项 目			规格	备注	
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)	
		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)	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)	
			32 位上/下数	C210 ~ C219, 12 点 (*2)	
	C220 ~ C255, 36 点 (*3)				
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 (字符串数据)	T	定时器当前值	T0 ~ T255, 16 位定时器, 256 点	计时到达·定时器接点导通	
	C	计数器当前值	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	数据寄存器	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 可做为变址的特殊用途。
			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)		
	变址	V0 ~ V7 (16 位), Z0 ~ Z7, 16 点 (32 位) (*1)			
指针	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。

■ 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)		

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。

### 3 各种装置功能

■ 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)

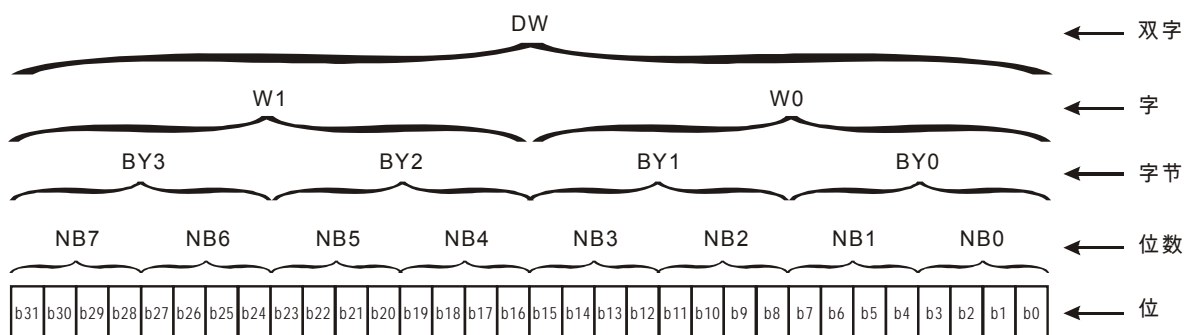
DVP-PM 内部依据各种不同控制目的，共使用 5 种数值类型执行运算的工作，各种数值的任务及功能如下说明。

##### 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(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(Double Word) : 是由连续之两个字所组成 (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、及双字的关系如下图所示：



## 2. 八进制 (Octal Number · OCT)

DVP-PM 的外部输入及输出端子编号采用八进制编码，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值，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。

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、字或双字形式的数据。  
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 进制数值在 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 进制，数值大小为 256。

### 3 各种装置功能

浮点数 F：

浮点数值在 DVP-PM 中，通常在其数值前面以“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)		八进制 (OCT)	十进制 (DEC)	BCD (Binary Code Decimal)		16 进制 (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
:	:	:	:	:	:	:
:	:	:	:	:	:	:
:	:	:	:	:	:	:
:	:	:	:	:	:	:
:	:	:	:	:	:	:
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：

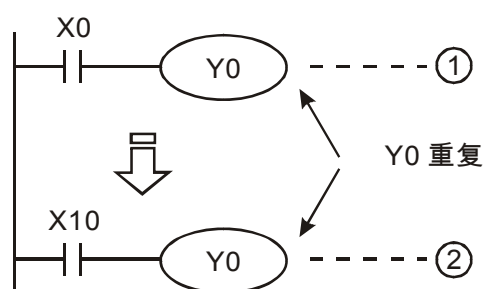
输入接点 X 的功能：输入接点 X 与输入装置连接，读取输入信号进入 DVP-PM。每一个输入接点 X 的 A 或 B 接点于程序中使用次数没有限制。输入接点 X 之 On/Off 只会跟随输入装置的 On/Off 做变化。

## ■ M1304 标志强制 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。

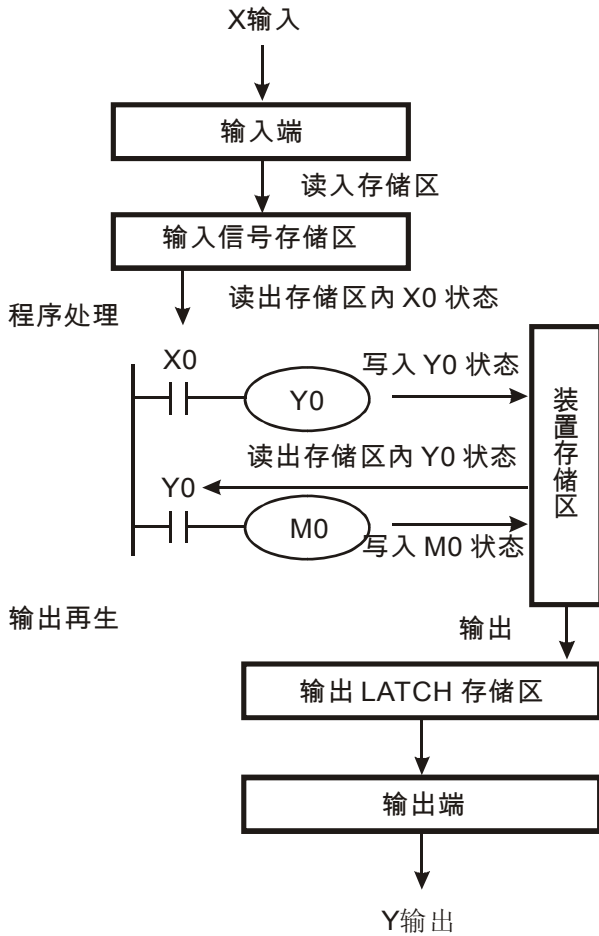


Y0 的输出最后会由电路 ② 决定，亦即由 X10 的 On/Off 决定 Y0 的输出

### 3 各种装置功能

DVP-PM 对于程序的处理流程(结束再生方式)

输入信号再生



● 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存储区内。
2. 在程序执行中若输入信号作 On/Off 变化，但是输入信号存储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 读取输入信号存储区内各输入信号之 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 也逐次存入各装置存储区。

● 输出再生：

1. 当执行到 M102 指令时将装置存储区内 Y 的 On/Off 状态送到输出 LATCH 存储区内，而此存储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，其状态将全部被复位为 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，请参考第 3.10 节特殊继电器及特殊寄存器，第 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
-------	----------	----------------------------------

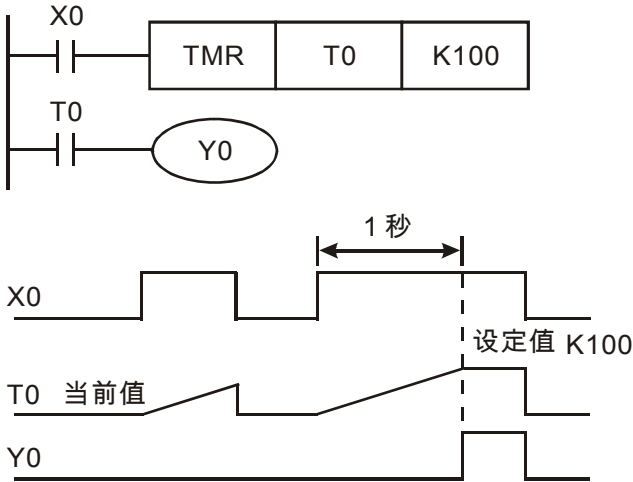
定时器的功能：

定时器是以 10ms 为一个计时单位，计时方式采用上数计时，当定时器当前值=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 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 之当前值以 10ms 采用上数计时，当定时器当前值=设定值 K100 时，输出线圈 T0=On。
- 当 X0=Off 或停电时，定时器 T0 之当前值清除为 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. 变址 D： 设定值使用数据寄存器 D 做变址。

#### 3.7 计数器的编号及功能 [C]

计数器的编号：(以 10 进制编号)

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(积分型)		
		32 位高速计数器	C200, C204, C208, C212, C216, C220, 6 点	合计 6 点	

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)	
当前值的变化	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位成 Off	
复位动作	RST 指令被执行时当前值归零，接点被复位成 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：

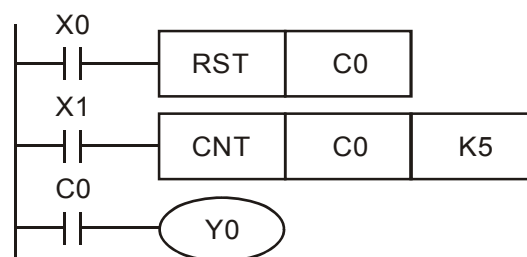
计数器之计数脉冲输入信号由 Off→On 时，计数器当前值等于设定值时输出线圈导通，设定值为 10 进制 K 值，亦可使用数据寄存器 D 当成设定值。

16 位计数器：

1. 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）
2. 一般用计数器在 DVP-PM 停电的时候，计数器当前值即被清除，若为停电保持型计数器(积分型)会将停电前的当前值及计数器接点状态保存着，复电后会继续累计。
3. 若使用 MOV 指令、PMSoft 将一个大于设定值的数值传送到 C0 当前值寄存器时，在下次 X1 由 Off→On 时，C0 计数器接点即变成 On，同时当前值内容变成与设定值相同。
4. 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。
5. 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。计数器当前值由 32,767 再往上累计时则变为 -32,768。

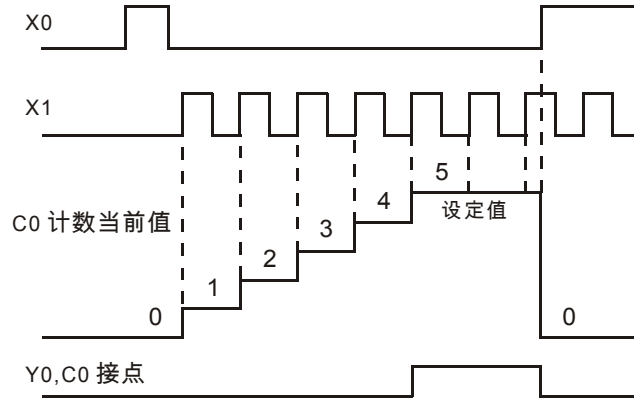
范例：

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



### 3 各种装置功能

1. 当 X0=On 时 RST 指令被执行，C0 的当前值归零，输出接点被复位为 Off。
2. 当 X1 由 Off→On 时，计数器之当前值将执行上数 (加一) 的动作。
3. 当计数器 C0 计数到达设定值 K5 时，C0 接点导通，C0 当前值 = 设定值=K5。之后的 X1 触发信号 C0 完全不接受，C0 当前值保持在 K5 处。

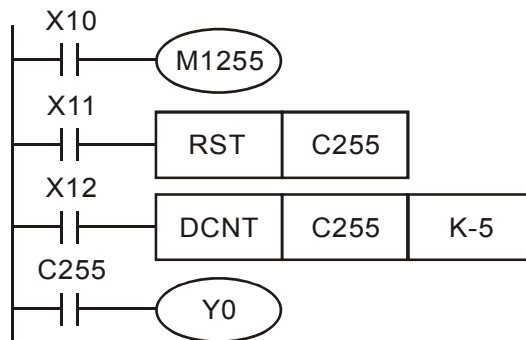


32 位一般用加减法计数器：

1. 32 位一般用计数器的设定范围：K-2,147,483,648 ~ K2,147,483,647。32 位一般用加减法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 为加法，M1208=On 时决定 C208 为减法其余类推。
2. 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。
3. 一般用计数器在 DVP-PM 停电的时候，计数器当前值即被清除，若为停电保持型计数器(积分型)，则会将停电前的当前值及计数器接点状态保存着，复电后会继续累计。
4. 计数器当前值由 2,147,483,647 再往上累计时则变为 -2,147,483,648。同理计数器当前值由 -2,147,483,648 再往下递减时，则变为 2,147,483,647。

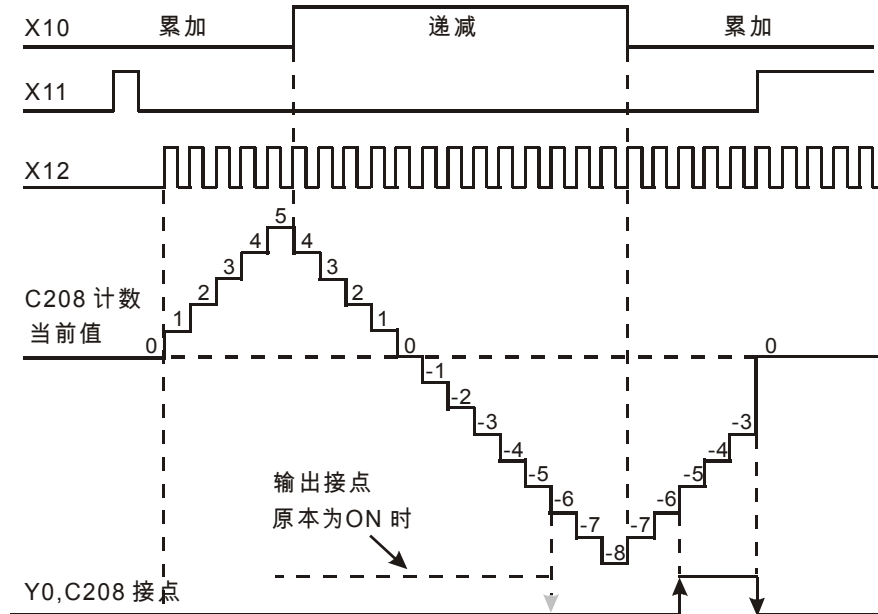
范例：

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



1. X10 驱动 M1255 来决定 C255 为加法或减法。
2. 当 X11 由 Off→On 时，RST 指令执行，C255 之当前值被清除为 0，且接点变为 Off。
3. 当 X12 由 Off→On 时，计数器之当前值将执行上数 (加一) 的动作或下数 (减一) 的动作。

4. 当计数器 C255 之当前值从 K-6→K-5 变化时，C255 接点由 Off→On。当计数器 C255 之当前值从 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 当前值寄存器时，在下次 X1 由 Off→On 时，C255 计数器接点即变成 On，同时当前值内容变成与设定值相同



32 位高速计数器：

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

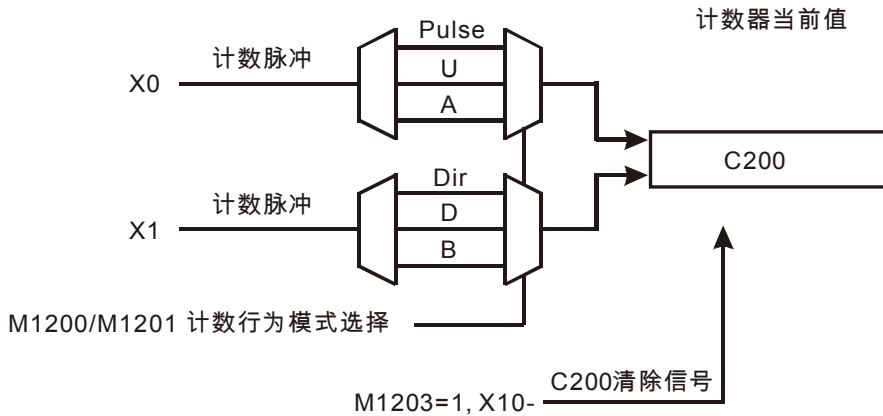
1. 32 位高速计数器的设定范围：K-2,147,483,648 ~ K2,147,483,647。
2. 计数模式设定

计数器编号	计数模式		清除信号设定	清除信号外部引脚	外部输入引脚
	装置	设定值			
C200	K1M1200	■ 计数模式	M1203	X10	X0、X1、S/S
C204	K1M1204	0：U/D*	M1207	X11	X2、X3、S/S
C208	K1M1208	1：P/D*	M1211	X12	X4、X5、S/S
C212	K1M1212	2：A/B*(1 倍频)	M1215	X13	X6、X7、S/S
C216	K1M1216	3：4A/B(4 倍频)	M1219	X0	X10+、X10-、X11+、X11-
C220	K1M1220	■ 计时模式	M1223	X1	X12+、X12-、X13+、X13-
		5：一般模式启动			
		6：周期模式启动			

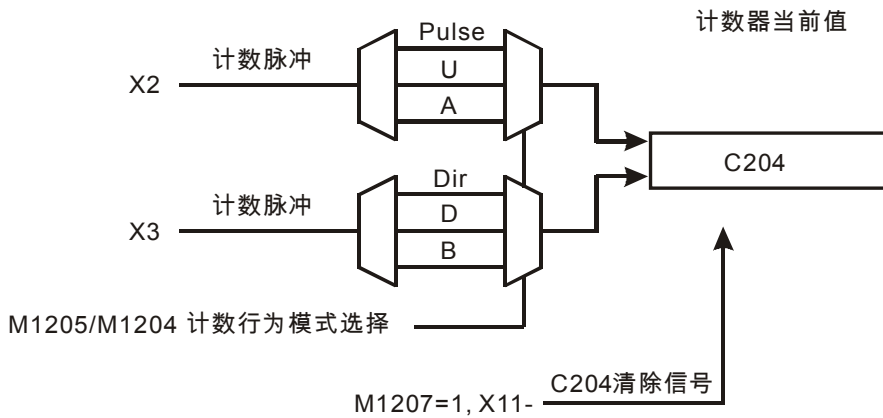
注：U/D：上数/下数，P/D：脉冲/方向，A/B：A 相/B 相

### 3 各种装置功能

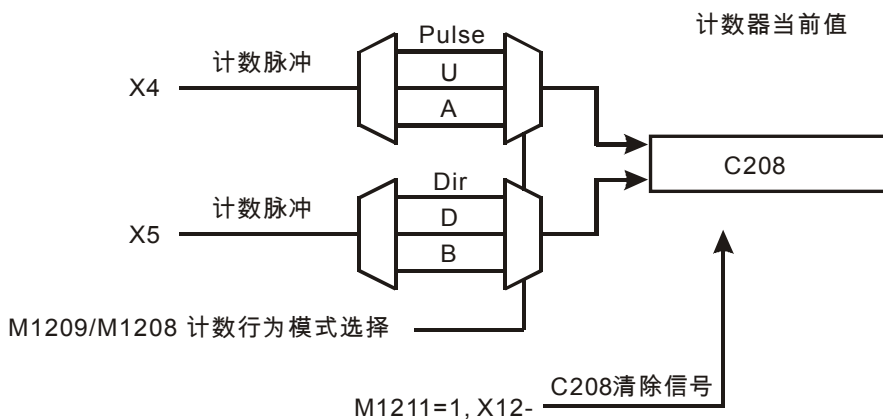
- C200 由 M1200/M1201 选择计数模式、输入信号由 X0、X1 控制；清除信号由 M1203 启动、由 X10 控制。



- C204 由 M1204/M1205 选择计数模式、输入信号由 X2、X3 控制；清除信号由 M1207 设定、由 X11 控制。

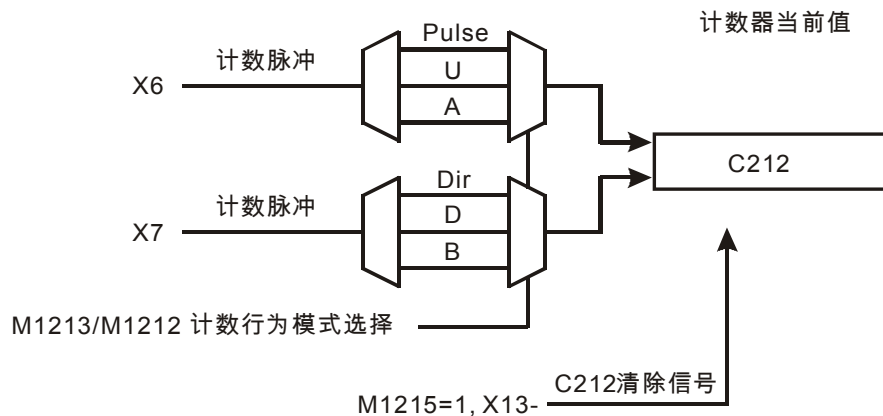


- C208 由 M1208/M1209 选择计数模式、输入信号由 X4、X5 控制；清除信号由 M1211 设定、由 X12 控制。

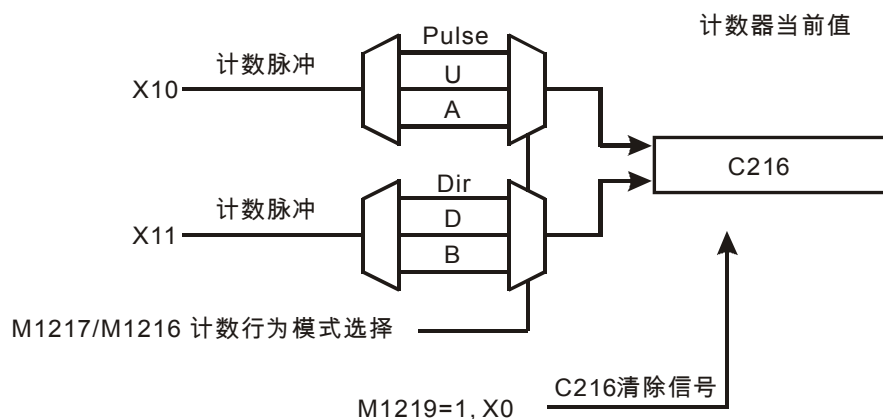




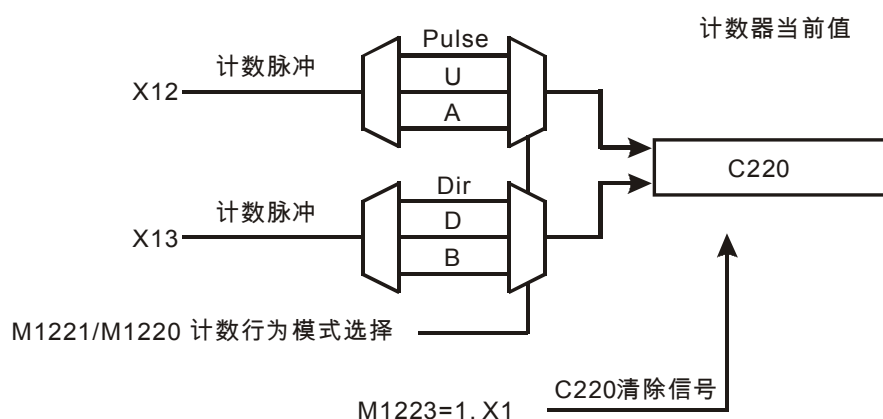
- C212 由 M1212/M1213 选择计数模式、输入信号由 X6、X7 控制；清除信号由 M1215 设定、由 X13 控制。



- C216 由 M1216/M1217 选择计数模式、输入信号由 X10、X11 控制；清除信号由 M1219 设定、由 X0 控制。



- C220 由 M1221/M1220 选择计数模式、输入信号由 X12、X13 控制；清除信号由 M1223 设定、由 X1 控制。



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，计数器当前值即被清除，若为停电保持型 (积分型) 计数器，则会将停电前的当前值及计数器接点状态保存着，复电后会继续累计。
5. 计数器当前值由 2,147,483,647 再往上累计时则变为 -2,147,483,648。同理计数器当前值由 -2,147,483,648 再往下递减时，则变为 2,147,483,647。

# 3 各种装置功能

## 3.8 寄存器的编号及功能 [D]

### 3.8.1 数据寄存器[D]

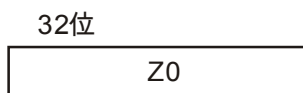
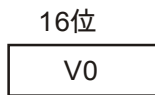
用于储存数值数据，其数据长度为 16 位 (-32,768 ~ +32,767)，最高位为正负号，可储存 -32,768~+32,767 之数值数据，亦可将两个 16 位寄存器合并成一个 32 位寄存器 (D+1,D 编号小的为下 16 位) 使用，而其最高位为正负号，可储存 -2,147,483,648 ~ +2,147,483,647 之数值数据。

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	
	变址寄存器 V、Z	V0~V7 · Z0~Z7 · 16 点	

寄存器依其性质可区分为下列 4 种：

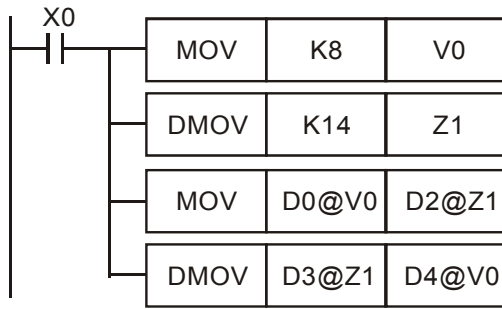
1. 一般用寄存器：当 DVP-PM 由 AUTO→MANU(STOP→RUN)或断电时，寄存器内的数值数据会被清除为 0，如果让 M1033=On 时，则 DVP-PM 由 AUTO→MANU(RUN→STOP)时，数据会保持不被清除，但断电时仍会被清除为 0。
2. 停电保持用寄存器：当 DVP-PM 断电时此区域的寄存器数据不会被清除，仍保持其断电前之数值。清除停电保持用寄存器的内容值，可使用 RST 或 ZRST 指令。
3. 特殊用寄存器：每个特殊用途寄存器均有其特殊定义与用途，主要作为存放系统状态、错误信息、监视状态之用。请参考第 3.10 节特殊继电器及特殊寄存器，第 3.11 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

如果使用变址寄存器 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)

变址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变址法 · 使用变址寄存器 V、Z 来修饰操作数请参考第 5.4 节。

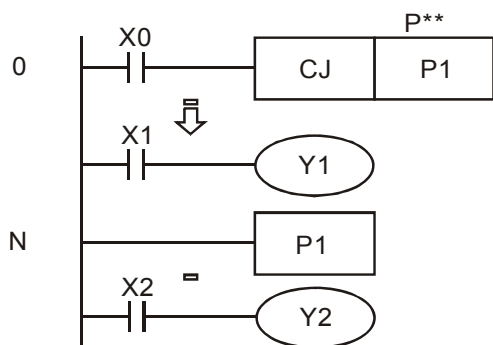
※ 使用变址寄存器 V、Z 来修饰操作数时 · 修饰范围请勿横跨特殊用寄存器 (D1000~D2999) 及特殊辅助继电器 (M1000~M2999) 之区域 · 以免发生错误。

### 3.9 指针 [N]、指针 [Pn]

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

指针 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 条件跳转：



- 当 X0=On 时程序自动从地址 0 跳转至地址 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 “\*” 标号可参考下一节的功能说明 · 如其中属性栏中标示为 “R” · 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。

### 3 各种装置功能

特 M	功能说明	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	3-35
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	3-35
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	3-35
M1003*	启始反向 (AUTO 的瞬间'Off') 脉冲。初期脉冲 · a 接点。AUTO 的瞬间 · 产生反向的 PULSE · PULSE 的宽度 = 扫描周期	On	Off	On	R	否	On	3-35
M1008	扫描逾时定时器(WDT) On	Off	Off	-	R	否	Off	-
M1009	LV 信号曾经发生过纪录	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	非运转中存储保持	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	-
M1039*	固定扫描时间	Off	-	-	R/W	否	Off	3-39
M1048	报警点状态标志	Off	-	-	R	否	Off	-
M1049	设定报警点监控标志	Off	-	-	R/W	否	Off	-
M1072	DVP-PM AUTO 指令执行 (通讯)	Off	On	Off	R/W	否	Off	-
M1077	电池电压过低、故障或无电池	Off	-	-	R/W	否	Off	-
M1087	LV 信号动作标志	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	3-36
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	-

特 M	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
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	3-36
M1138*	COM1 (RS-232) 通讯设定保持 · 设定后 D1036 变更无效	Off	-	-	R/W	否	Off	3-36
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	3-36
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	3-36
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标志 · 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 信号	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 信号	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	-
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	-

### 3 各种装置功能

特 M	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
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	-
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	-

特 M	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
M1303	XCH 指令高低位交换标志	Off	-	-	R/W	否	Off	-
M1304*	X 输入点可强制 On-Off	Off	-	-	R/W	否	Off	3-42
M1744*	OX M 码 Off 标志	Off	Off	-	R/W	否	Off	3-44
M1745	OX X 轴原点回归禁止标志	Off	-	-	R/W	否	Off	-
M1751	X 轴当前位置写入允许	Off	-	-	R/W	否	Off	3-44
M1760	OX 弧度/角度使用标志	Off	-	-	R/W	否	Off	-
M1792	X 轴 Ready Flag	On	On	On	R	否	On	3-44
M1793*	X 轴运动错误标志与清除(X 轴启动时自动清除)	Off	-	-	R/W	否	Off	3-44
M1794*	OX M 码 On 标志(OX 启动时自动清除)	Off	-	Off	R	否	Off	-
M1795	OX M0 码 On(OX 启动时自动清除)	Off	-	-	R	否	Off	-
M1796	OX M2 码 On(OX 启动时自动清除)	Off	On	-	R	否	Off	-
M1808	OX 零标志 (OX Zero flag)	Off	-	-	R	否	Off	-
M1809	OX 借位标志 (OX Borrow flag)	Off	-	-	R	否	Off	-
M1810	OX 进位标志 (OX Carry flag)	Off	-	-	R	否	Off	-
M1811	OX 浮点数运算错误标志	Off	Off	-	R	否	Off	-
M1825	Y 轴原点回归禁止标志	Off	-	-	R/W	否	Off	-
M1831	Y 轴当前位置写入允许	Off	-	-	R/W	否	Off	3-44
M1872	Y 轴 Ready Flag	On	On	On	R	否	On	3-44
M1873*	Y 轴运动错误标志与清除 (Y 轴启动时自动清除)	Off	-	-	R	否	Off	3-44
M1920	O100 弧度/角度使用标志	Off	-	-	R/W	否	Off	-
M1952	O100 Ready flag	On	Off	On	R	否	On	-
M1953*	O100 错误标志/清除	Off	-	-	R/W	否	Off	3-46
M1957	AUTO/MANU(STOP/RUN)开关状态标志	Off	On	-	R	否	Off	-
M1968	O100 零标志 (O100 Zero flag)	Off	-	-	R	否	Off	-
M1969	O100 借位标志 (O100 Borrow flag)	Off	-	-	R	否	Off	-
M1970	O100 进位标志 (O100 Carry flag)	Off	-	-	R	否	Off	-
M1971	O100 浮点数运算错误标志	Off	-	-	R	否	Off	-
M1985	Z 轴原点回归禁止标志	Off	-	-	R/W	否	Off	-
M1991	Z 轴当前位置写入允许	Off	-	-	R/W	否	Off	3-44
M2032	Z 轴 Ready Flag	On	On	On	R	否	On	3-44
M2033*	Z 轴运动错误标志与清除 (Z 轴启动时自动清除)	Off	-	-	R/W	否	Off	3-44
M2065	A 轴原点回归禁止标志	Off	-	-	R/W	否	Off	-
M2071	A 轴当前位置写入允许	Off	-	-	R/W	否	Off	3-44
M2112	A 轴 Ready Flag	On	On	On	R	否	On	3-44
M2113*	A 轴运动错误标志与清除 (A 轴启动时自动清除)	Off	-	-	R/W	否	Off	3-44
M2145	B 轴原点回归禁止标志	Off	-	-	R/W	否	Off	-

### 3 各种装置功能

特 M	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
M2151	B 轴当前位置写入允许	Off	-	-	R/W	否	Off	3-44
M2192	B 轴 Ready Flag	On	On	On	R	否	On	3-44
M2193*	B 轴运动错误标志与清除 (A 轴启动时自动清除)	Off	-	-	R/W	否	Off	3-44
M2225	C 轴原点回归禁止标志	Off	-	-	R/W	否	Off	-
M2231	C 轴当前位置写入允许	Off	-	-	R/W	否	Off	3-44
M2272	C 轴 Ready Flag	On	On	On	R	否	On	3-44
M2273*	C 轴运动错误标志与清除 (A 轴启动时自动清除)	Off	-	-	R/W	否	Off	3-44



特 D	功能说明	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	3-35
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	当前扫描时间 (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 输入滤波设定 · 单位 ms	10	-	-	R/W	否	10	3-36
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	3-36
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	3-39
D1039*	固定扫描时间 (ms)	0	-	-	R/W	否	0	3-39
D1050 ↓ D1055	Modbus 通讯指令数据处理 · DVP-PM 系统会自动将 D1070~D1085 的 ASCII 字数据转换为 HEX · 16 进制数值	0	-	-	R	否	0	-
D1056	功能卡 2AD CH0 当前值	0	#	-	R	否	0	-
D1057	功能卡 2AD CH1 当前值	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 · 当接收端接收后会回传信息 · 该信息会储存于 D1070~D1085 · 使用者可利用该寄存器的内容 · 查看回传数据	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 · 查看指令是否正确	0	-	-	R	否	0	-
D1109	COM3 (通讯卡) 通讯协议	H'86	-	-	R/W	否	H'86	3-36
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	3-36
D1121	DVP-PM 通讯地址	-	-	-	R/W	是	1	-
D1122	发送数据剩余字数	0	0	-	R	否	0	-
D1123	接收数据剩余字数	0	0	-	R	否	0	-
D1124	起始字定义(STX)	H'3A	-	-	R/W	否	H'3A	-
D1125	第一结束字定义(END High)	H'0D	-	-	R/W	否	H'0D	-
D1126	第二结束字定义(END Low)	H'0A	-	-	R/W	否	H'0A	-

### 3 各种装置功能

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
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	3-42
D1142*	数字模块 X 点数	0	-	-	R	否	0	3-42
D1143*	数字模块 Y 点数	0	-	-	R	否	0	3-42
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	3-42
D1201*	M 辅助继电器停电保持结束地址	-	-	-	R/W	是	999	3-42
D1202*	T 定时器停电保持开始地址	-	-	-	R/W	是	-1	3-42
D1203*	T 定时器停电保持结束地址	-	-	-	R/W	是	-1	3-42
D1204*	C16 位计数器停电保持开始地址	-	-	-	R/W	是	100	3-42
D1205*	C16 位计数器停电保持结束地址	-	-	-	R/W	是	199	3-42
D1206*	C32 位计数器停电保持开始地址	-	-	-	R/W	是	220	3-42
D1207*	C32 位计数器停电保持结束地址	-	-	-	R/W	是	255	3-42
D1208*	S 辅助继电器停电保持开始地址	-	-	-	R/W	是	500	3-42
D1209*	S 辅助继电器停电保持结束地址	-	-	-	R/W	是	1023	3-42
D1210*	D 寄存器停电保持开始地址	-	-	-	R/W	是	200	3-42
D1211*	D 寄存器停电保持结束地址	-	-	-	R/W	是	9999	3-42
D1313*	万年历(RTC) 秒 00~59	-	-	-	R/W	是	0	-
D1314*	万年历(RTC) 分 00~59	-	-	-	R/W	是	0	-
D1315*	万年历(RTC) 时 00~23	-	-	-	R/W	是	0	-
D1316*	万年历(RTC) 日 01~31	-	-	-	R/W	是	1	-
D1317*	万年历(RTC) 月 01~12	-	-	-	R/W	是	1	-
D1318*	万年历(RTC) 星期 1~7	-	-	-	R/W	是	2/5	-
D1319*	万年历(RTC) 年 00~99	-	-	-	R/W	是	8/10	-
D1320*	第一台右侧模块代码	0	-	-	R	否	0	3-43
D1321*	第二台右侧模块代码	0	-	-	R	否	0	3-43
D1322*	第三台右侧模块代码	0	-	-	R	否	0	3-43
D1323*	第四台右侧模块代码	0	-	-	R	否	0	3-43
D1324*	第五台右侧模块代码	0	-	-	R	否	0	3-43
D1325*	第六台右侧模块代码	0	-	-	R	否	0	3-43
D1326*	第七台右侧模块代码	0	-	-	R	否	0	3-43
D1327*	第八台右侧模块代码	0	-	-	R	否	0	3-43
D1400	中断允许	0	-	-	R/W	否	0	-
D1401	时间中断周期 (ms)	0	-	-	R/W	否	0	-

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
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	3-44
D1704	OX 等待时间设定值	0	-	-	R	否	0	-
D1705	OX 等待时间当前值	0	-	-	R	否	0	-
D1706	OX RPT 指令设定值	0	-	-	R	否	0	-
D1707	OX RPT 指令当前值	0	-	-	R	否	0	-
D1736	O100 等待时间 (TIM) 设定值	0	-	-	R	否	0	-
D1737	O100 等待时间 (TIM) 当前值	0	-	-	R	否	0	-
D1738	O100 RPT 指令设定值	0	-	-	R	否	0	-
D1739	O100 RPT 指令当前值	0	-	-	R	否	0	-
D1799*	输入端子极性	0	-	-	R/W	否	0	3-45
D1800*	输入端子状态	0	-	-	R	否	0	3-45
D1802*	O100 错误编号	0	-	-	R/W	否	0	3-46
D1803*	O100 错误 STEP 位置	0	0	-	R/W	否	0	3-46
D1806	输入端子滤波系数设定	0	-	-	R/W	否	0	3-45
D1816*	X 轴参数设定	-	-	-	R/W	是	0	3-46
D1818	X 轴电机转一圈所需脉冲数 (Low word)	-	-	-	R/W	是	2000	-
D1819	X 轴电机转一圈所需脉冲数 (High word)	-	-	-	R/W	是	2000	-
D1820	X 轴电机转一圈之移动距离 (Low word)	-	-	-	R/W	是	1000	-
D1821	X 轴电机转一圈之移动距离 (High word)	-	-	-	R/W	是	1000	-
D1822	X 轴最高速度 $V_{MAX}$ (Low word)	-	-	-	R/W	是	500K	-
D1823	X 轴最高速度 $V_{MAX}$ (High word)	-	-	-	R/W	是	500K	-
D1824	X 轴启动速度 $V_{BIAS}$ (Low word)	-	-	-	R/W	是	0	-
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	5000	-
D1828	X 轴原点回归速度 $V_{RT}$ (Low word)	-	-	-	R/W	是	50K	-
D1829	X 轴原点回归速度 $V_{RT}$ (High word)	-	-	-	R/W	是	50K	-
D1830	X 轴原点回归减速速度 $V_{CR}$ (Low word)	-	-	-	R/W	是	1000	-
D1831	X 轴原点回归减速速度 $V_{CR}$ (High word)	-	-	-	R/W	是	1000	-
D1832*	X 轴零点信号数 N	-	-	-	R/W	是	0	-
D1833*	X 轴补偿距离 P	-	-	-	R/W	是	0	-

### 3 各种装置功能

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1834*	X 轴原点位置定义 HP (Low word)	-	-	-	R/W	是	0	-
D1835	X 轴原点位置定义 HP (High word)	-	-	-	R/W	是	0	-
D1836	X 轴加速时间 T <sub>ACC</sub>	-	-	-	R/W	是	500	-
D1837	X 轴减速时间 T <sub>DEC</sub>	-	-	-	R/W	是	500	-
D1838	X 轴目标位置(I) P(I) (Low word)	0	-	-	R/W	否	0	-
	X 轴 PWM 频宽设定(Low word)	0	-	-	R/W	否	0	-
D1839	X 轴目标位置(I) P(I) (High word)	0	-	-	R/W	否	0	-
	X 轴 PWM 频宽设定(High word)	0	-	-	R/W	否	0	-
D1840	X 轴运转速度(I) V(I) (Low word)	1000	-	-	R/W	否	1000	-
D1841	X 轴运转速度(I) V(I) (High word)	1000	-	-	R/W	否	1000	-
D1842	X 轴目标位置(II) P(II) (Low word)	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	-
	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	3-47
D1847*	X 轴工作模式	0	-	-	R/W	否	0	3-48
D1848	X 轴当前位置 CP(PLS) (Low word)	0	-	-	R/W	否	0	-
	从轴位置							
D1849	X 轴当前位置 CP(PLS) (High word)	0	-	-	R/W	否	0	-
	从轴位置							
D1850	X 轴当前速度(PPS) (Low word)	0	0	0	R/W	否	0	-
D1851	X 轴当前速度(PPS) (High word)							
D1852	X 轴当前位置 CP(unit) (Low word)	0	-	-	R/W	否	0	-
D1853	X 轴当前位置 CP(unit) (High word)							
D1854	X 轴当前速度 CS (unit) (Low word)	0	0	0	R/W	否	0	-
D1855	X 轴当前速度 CS (unit) (High word)							
D1856*	X 轴执行状态	0	-	-	R	否	0	3-49
D1857*	OX、X 轴错误编号	0	-	-	R	否	0	3-44
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							

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
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							
D1864*	X 轴手摇轮输入响应速度	-	-	-	R/W	是	5	-
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	-
D1867	X 轴电气零点 (High word)	-	-	-	R/W	是	0	-
D1868*	设定 OX 程序号码	0	0	-	R/W	是	0	3-39
D1869	OX 错误 STEP 位置	0	-	-	R/W	否	0	-
D1872	OX Ready 时·允许 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 码时·允许 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 输入信号重置 M 码	0	-	-	R/W	否	0	-
D1875*	X 轴外部手动启动 (ZRN, MPG, JOG-, JOG+)	-	-	-	R/W	是	0	-
D1896*	Y 轴参数设定	-	-	-	R/W	是	0	3-46
D1898	Y 轴电机转一圈所需脉冲数 (Low word)	-	-	-	R/W	是	2000	-
D1899	Y 轴电机转一圈所需脉冲数 (High word)							
D1900	Y 轴电机转一圈之移动距离 (Low word)	-	-	-	R/W	是	1000	-
D1901	Y 轴电机转一圈之移动距离 (High word)							
D1902	Y 轴最高速度 V <sub>MAX</sub> (Low word)	-	-	-	R/W	是	500K	-
D1903	Y 轴最高速度 V <sub>MAX</sub> (High word)							
D1904	Y 轴启动速度 V <sub>BIAS</sub> (Low word)	-	-	-	R/W	是	0	-
D1905	Y 轴启动速度 V <sub>BIAS</sub> (High word)							
D1906	Y 轴寸动 JOG 速度 V <sub>JOG</sub> (Low word)	-	-	-	R/W	是	5000	-
D1907	Y 轴寸动 JOG 速度 V <sub>JOG</sub> (High word)							
D1908	Y 轴原点回归速度 V <sub>RT</sub> (Low word)	-	-	-	R/W	是	50K	-
D1909	Y 轴原点回归速度 V <sub>RT</sub> (High word)							
D1910	Y 轴原点回归减速速度 V <sub>CR</sub> (Low word)	-	-	-	R/W	是	1000	-
D1911	Y 轴原点回归减速速度 V <sub>CR</sub> (High word)							
D1912	Y 轴零点信号数 N	-	-	-	R/W	是	0	-
D1913	Y 轴补偿距离 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 <sub>ACC</sub>	-	-	-	R/W	是	500	-

### 3 各种装置功能

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1917	Y 轴减速时间 T <sub>DEC</sub>	-	-	-	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	3-47
D1927*	Y 轴工作模式	0	-	-	R/W	否	0	3-48
D1928	Y 轴当前位置 CP(PLS) (Low word)	0	-	-	R/W	否	0	-
D1929	Y 轴当前位置 CP(PLS) (High word)							
D1930	Y 轴当前速度(PPS) (Low word)	0	0	0	R/W	否	0	-
D1931	Y 轴当前速度(PPS) (High word)							
D1932	Y 轴当前位置 CP(unit) (Low word)	0	-	-	R/W	否	0	-
D1933	Y 轴当前位置 CP(unit) (High word)							
D1934	Y 轴当前速度 CS (unit) (Low word)	0	0	0	R/W	否	0	-
D1935	Y 轴当前速度 CS (unit) (High word)							
D1936*	Y 轴执行状态	0	-	-	R	否	0	3-49
D1937*	Y 轴错误编号	0	-	-	R	否	0	3-44
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	-
D1976	Z 轴参数设定	-	-	-	R/W	是	0	3-46

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D1978	Z 轴电机转一圈所需脉冲数 (Low word)	-	-	-	R/W	是	2000	-
D1979	Z 轴电机转一圈所需脉冲数 (High word)							
D1980	Z 轴电机转一圈之移动距离 (Low word)	-	-	-	R/W	是	1000	-
D1981	Z 轴电机转一圈之移动距离 (High word)							
D1982	Z 轴最高速度 V <sub>MAX</sub> (Low word)	-	-	-	R/W	是	500K	-
D1983	Z 轴最高速度 V <sub>MAX</sub> (High word)							
D1984	Z 轴启动速度 V <sub>BIAS</sub> (Low word)	-	-	-	R/W	是	0	-
D1985	Z 轴启动速度 V <sub>BIAS</sub> (High word)							
D1986	Z 轴寸动 JOG 速度 V <sub>JOG</sub> (Low word)	-	-	-	R/W	是	5000	-
D1987	Z 轴寸动 JOG 速度 V <sub>JOG</sub> (High word)							
D1988	Z 轴原点回归速度 V <sub>RT</sub> (Low word)	-	-	-	R/W	是	50K	-
D1989	Z 轴原点回归速度 V <sub>RT</sub> (High word)							
D1990	Z 轴原点回归减速速度 V <sub>CR</sub> (Low word)	-	-	-	R/W	是	1000	-
D1991	Z 轴原点回归减速速度 V <sub>CR</sub> (High word)							
D1992	Z 轴零点信号数 N	-	-	-	R/W	是	0	-
D1993	Z 轴补偿距离 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 <sub>ACC</sub>	-	-	-	R/W	是	500	-
D1997	Z 轴减速时间 T <sub>DEC</sub>	-	-	-	R/W	是	500	-
D1998	Z 轴目标位置(I) P(I) (Low word)	0	-	-	R/W	否	0	-
	Z 轴 PWM 频宽设定(Low word)	0	-	-	R/W	否	0	-
D1999	Z 轴目标位置(I) P(I) (High word)	0	-	-	R/W	否	0	-
	Z 轴 PWM 频宽设定(High word)	0	-	-	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	3-47
D2007	Z 轴工作模式	0	-	-	R/W	否	0	3-48
D2008	Z 轴当前位置 CP(PLS) (Low word)	0	-	-	R/W	否	0	-
D2009	Z 轴当前位置 CP(PLS) (High word)							

### 3 各种装置功能

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D2010	Z 轴当前速度(PPS) (Low word)	0	0	0	R/W	否	0	-
D2011	Z 轴当前速度(PPS) (High word)							
D2012	Z 轴当前位置 CP(unit) (Low word)	0	-	-	R/W	否	0	-
D2013	Z 轴当前位置 CP(unit) (High word)							
D2014	Z 轴当前速度 CS (unit) (Low word)	0	0	0	R/W	否	0	-
D2015	Z 轴当前速度 CS (unit) (High word)							
D2016	Z 轴执行状态	0	-	-	R	否	0	3-49
D2017*	Z 轴错误编号	0	-	-	R	否	0	3-44
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	-
D2056	A 轴参数设定	-	-	-	R/W	是	0	3-46
D2058	A 轴电机转一圈所需脉冲数 (Low word)	-	-	-	R/W	是	2000	-
D2059	A 轴电机转一圈所需脉冲数 (High word)							
D2060	A 轴电机转一圈之移动距离 (Low word)	-	-	-	R/W	是	1000	-
D2061	A 轴电机转一圈之移动距离 (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 轴原点回归速度 $V_{RT}$ (Low word)	-	-	-	R/W	是	50K	-
D2069	A 轴原点回归速度 $V_{RT}$ (High word)							
D2070	A 轴原点回归减速速度 $V_{CR}$ (Low word)	-	-	-	R/W	是	1000	-
D2071	A 轴原点回归减速速度 $V_{CR}$ (High word)							
D2072	A 轴零点信号数 N	-	-	-	R/W	是	0	-
D2073	A 轴补偿距离 P	-	-	-	R/W	是	0	-
D2074	A 轴原点位置定义 HP (Low word)	-	-	-	R/W	是	0	-



特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D2075	A 轴原点位置定义 HP (High word)	-	-	-	R/W	是	0	-
D2076	A 轴加速时间 T <sub>ACC</sub>	-	-	-	R/W	是	500	-
D2077	A 轴减速时间 T <sub>DEC</sub>	-	-	-	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)	0	-	-	R/W	否	0	-
D2079	A 轴目标位置(I) P(I) (High word)	0	-	-	R/W	否	0	-
	A 轴 PWM 频宽设定(High word)	0	-	-	R/W	否	0	-
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)		-	-	R/W	否	1000	-
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)	0	-	-	R/W	否	0	-
D2083	A 轴目标位置(II) P(II) (High word)	0	-	-	R/W	否	0	-
	A 轴 PWM 输出周期设定(High word)	0	-	-	R/W	否	0	-
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)		-	-	R/W	否	2K	-
D2086	A 轴运转命令	0	-	0	R/W	否	0	3-47
D2087	A 轴工作模式	0	-	-	R/W	否	0	3-48
D2088	A 轴当前位置 CP(PLS) (Low word)	0	-	-	R/W	否	0	-
D2089	A 轴当前位置 CP(PLS) (High word)		-	-	R/W	否	0	-
D2090	A 轴当前速度(PPS) (Low word)	0	0	0	R/W	否	0	-
D2091	A 轴当前速度(PPS) (High word)		0	0	R/W	否	0	-
D2092	A 轴当前位置 CP(unit) (Low word)	0	-	-	R/W	否	0	-
D2093	A 轴当前位置 CP(unit) (High word)		-	-	R/W	否	0	-
D2094	A 轴当前速度 CS (unit) (Low word)	0	0	0	R/W	否	0	-
D2095	A 轴当前速度 CS (unit) (High word)		0	0	R/W	否	0	-
D2096	A 轴执行状态	0	-	-	R	否	0	3-49
D2097*	A 轴错误编号	0	-	-	R	否	0	3-44
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	-
D2100	A 轴手摇轮输入频率 (Low word)	0	-	0	R/W	否	0	-
D2101	A 轴手摇轮输入频率 (High word)	0	-	0	R/W	否	0	-
D2102	A 轴累计手摇轮输入脉冲个数 (Low word)	0	-	-	R/W	否	0	-
D2103	A 轴累计手摇轮输入脉冲个数 (High word)	0	-	-	R/W	否	0	-
D2104	A 轴手摇轮输入响应速度	-	-	-	R/W	是	5	-
D2136	B 轴参数设定	-	-	-	R/W	是	0	3-46
D2138	B 轴电机转一圈所需脉冲数 (Low word)	-	-	-	R/W	是	2000	-

### 3 各种装置功能

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D2139	B 轴电机转一圈所需脉冲数 (High word)	-	-	-	R/W	是	2000	-
D2140	B 轴电机转一圈之移动距离 (Low word)	-	-	-	R/W	是	1000	-
D2141	B 轴电机转一圈之移动距离 (High word)				R/W	是	1000	
D2142	B 轴最高速度 V <sub>MAX</sub> (Low word)	-	-	-	R/W	是	500K	-
D2143	B 轴最高速度 V <sub>MAX</sub> (High word)	-	-	-	R/W	是	500K	
D2144	B 轴启动速度 V <sub>BIAS</sub> (Low word)	-	-	-	R/W	是	0	-
D2145	B 轴启动速度 V <sub>BIAS</sub> (High word)				R/W	是	0	
D2146	B 轴寸动 JOG 速度 V <sub>JOG</sub> (Low word)	-	-	-	R/W	是	5000	-
D2147	B 轴寸动 JOG 速度 V <sub>JOG</sub> (High word)				R/W	是	5000	
D2148	B 轴原点回归速度 V <sub>RT</sub> (Low word)	-	-	-	R/W	是	50K	-
D2149	B 轴原点回归速度 V <sub>RT</sub> (High word)	-	-	-	R/W	是	50K	-
D2150	B 轴原点回归减速速度 V <sub>CR</sub> (Low word)	-	-	-	R/W	是	1000	-
D2151	B 轴原点回归减速速度 V <sub>CR</sub> (High word)	-	-	-	R/W	是	1000	-
D2152	B 轴零点信号数 N	-	-	-	R/W	是	0	-
D2153	B 轴补偿距离 P	-	-	-	R/W	是	0	-
D2154	B 轴原点位置定义 HP (Low word)	-	-	-	R/W	是	0	-
D2155	B 轴原点位置定义 HP (High word)	-	-	-	R/W	是	0	-
D2156	B 轴加速时间 T <sub>ACC</sub>	-	-	-	R/W	是	500	-
D2157	B 轴减速时间 T <sub>DEC</sub>	-	-	-	R/W	是	500	-
D2158	B 轴目标位置(I) P(I) (Low word)	0	-	-	R/W	否	0	-
D2159	B 轴目标位置(I) P(I) (High word)	0	-	-	R/W	否	0	-
D2160	B 轴运转速度(I) V(I) (Low word)	1000	-	-	R/W	否	1000	-
D2161	B 轴运转速度(I) V(I) (High word)	1000	-	-	R/W	否	1000	-
D2162	B 轴目标位置(II) P(II) (Low word)	0	-	-	R/W	否	0	-
D2163	B 轴目标位置(II) P(II) (High word)	0	-	-	R/W	否	0	-
D2164	B 轴运转速度(II)V(II) (Low word)	2000	-	-	R/W	否	2K	-
D2165	B 轴运转速度(II)V(II) (High word)	2000	-	-	R/W	否	2K	-
D2166	B 轴运转命令	0	-	0	R/W	否	0	3-47
D2167	B 轴工作模式	0	-	-	R/W	否	0	3-48
D2168	B 轴当前位置 CP(PLS) (Low word)	0	-	-	R/W	否	0	-
D2169	B 轴当前位置 CP(PLS) (High word)	0	-	-	R/W	否	0	-
D2170	B 轴当前速度(PPS) (Low word)	0	0	0	R/W	否	0	-
D2171	B 轴当前速度(PPS) (High word)	0	0	0	R/W	否	0	-
D2172	B 轴当前位置 CP(unit) (Low word)	0	-	-	R/W	否	0	-
D2173	B 轴当前位置 CP(unit) (High word)	0	-	-	R/W	否	0	-
D2174	B 轴当前速度 CS (unit) (Low word)	0	0	0	R/W	否	0	-

特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D2175	B 轴当前速度 CS (unit) (High word)	0	0	0	R/W	否	0	-
D2176	B 轴执行状态	0	-	-	R	否	0	3-49
D2177	B 轴错误编号	0	-	-	R	否	0	3-44
D2178	B 轴电子齿轮 (分子)	-	-	-	R/W	是	1	-
D2179	B 轴电子齿轮 (分母)	-	-	-	R/W	是	1	-
D2180	B 轴手摇轮输入频率 (Low word)	0	-	0	R/W	否	0	-
D2181	B 轴手摇轮输入频率 (High word)	0	-	0	R/W	否	0	-
D2182	B 轴累计手摇轮输入脉冲个数 (Low word)	0	-	-	R/W	否	0	-
D2183	B 轴累计手摇轮输入脉冲个数 (High word)	0	-	-	R/W	否	0	-
D2184	B 轴手摇轮输入响应速度	-	-	-	R/W	是	5	-
D2216	C 轴参数设定	-	-	-	R/W	是	0	3-46
D2218	C 轴电机转一圈所需脉冲数 (Low word)	-	-	-	R/W	是	2000	-
D2219	C 轴电机转一圈所需脉冲数 (High word)	-	-	-	R/W	是	2000	-
D2220	C 轴电机转一圈之移动距离 (Low word)	-	-	-	R/W	是	1000	-
D2221	C 轴电机转一圈之移动距离 (High word)	-	-	-	R/W	是	1000	-
D2222	C 轴最高速度 V <sub>MAX</sub> (Low word)	-	-	-	R/W	是	500K	-
D2223	C 轴最高速度 V <sub>MAX</sub> (High word)	-	-	-	R/W	是	500K	-
D2224	C 轴启动速度 V <sub>BIAS</sub> (Low word)	-	-	-	R/W	是	0	-
D2225	C 轴启动速度 V <sub>BIAS</sub> (High word)	-	-	-	R/W	是	0	-
D2226	C 轴寸动 JOG 速度 V <sub>JOG</sub> (Low word)	-	-	-	R/W	是	5000	-
D2227	C 轴寸动 JOG 速度 V <sub>JOG</sub> (High word)	-	-	-	R/W	是	5000	-
D2228	C 轴原点回归速度 V <sub>RT</sub> (Low word)	-	-	-	R/W	是	50K	-
D2229	C 轴原点回归速度 V <sub>RT</sub> (High word)	-	-	-	R/W	是	50K	-
D2230	C 轴原点回归减速速度 V <sub>CR</sub> (Low word)	-	-	-	R/W	是	1000	-
D2231	C 轴原点回归减速速度 V <sub>CR</sub> (High word)	-	-	-	R/W	是	1000	-
D2232	C 轴零点信号数 N	-	-	-	R/W	是	0	-
D2233	C 轴补偿距离 P	-	-	-	R/W	是	0	-
D2234	C 轴原点位置定义 HP (Low word)	-	-	-	R/W	是	0	-
D2235	C 轴原点位置定义 HP (High word)	-	-	-	R/W	是	0	-
D2236	C 轴加速时间 T <sub>ACC</sub>	-	-	-	R/W	是	500	-
D2237	C 轴减速时间 T <sub>DEC</sub>	-	-	-	R/W	是	500	-
D2238	C 轴目标位置(I) P(I) (Low word)	0	-	-	R/W	否	0	-
D2239	C 轴目标位置(I) P(I) (High word)	0	-	-	R/W	否	0	-
D2240	C 轴运转速度(I) V(I) (Low word)	1000	-	-	R/W	否	1000	-
D2241	C 轴运转速度(I) V(I) (High word)	1000	-	-	R/W	否	1000	-
D2242	C 轴目标位置(II) P(II) (Low word)	0	-	-	R/W	否	0	-

### 3 各种装置功能

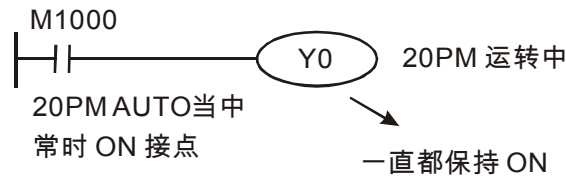
特 D	功能说明	Off ↓ On	MANU ↓ AUTO	AUTO ↓ MANU	属性	停电保持	出厂值	页码
D2243	C 轴目标位置(II) P(II) (High word)	0	-	-	R/W	否	0	-
D2244	C 轴运转速度(II)V(II) (Low word)	2000	-	-	R/W	否	2K	-
D2245	C 轴运转速度(II)V(II) (High word)	2000	-	-	R/W	否	2K	-
D2246	C 轴运转命令	0	-	0	R/W	否	0	-
D2247	C 轴工作模式	0	-	-	R/W	否	0	3-48
D2248	C 轴当前位置 CP(PLS) (Low word)	0	-	-	R/W	否	0	-
D2249	C 轴当前位置 CP(PLS) (High word)	0	-	-	R/W	否	0	-
D2250	C 轴当前速度(PPS) (Low word)	0	0	0	R/W	否	0	-
D2251	C 轴当前速度(PPS) (High word)	0	0	0	R/W	否	0	-
D2252	C 轴当前位置 CP(unit) (Low word)	0	-	-	R/W	否	0	-
D2253	C 轴当前位置 CP(unit) (High word)	0	-	-	R/W	否	0	-
D2254	C 轴当前速度 CS (unit) (Low word)	0	0	0	R/W	否	0	-
D2255	C 轴当前速度 CS (unit) (High word)	0	0	0	R/W	否	0	-
D2256	C 轴执行状态	0	-	-	R	否	0	3-49
D2257	C 轴错误编号	0	-	-	R	否	0	3-44
D2258	C 轴电子齿轮 (分子)	-	-	-	R/W	是	1	-
D2259	C 轴电子齿轮 (分母)	-	-	-	R/W	是	1	-
D2260	C 轴手摇轮输入频率 (Low word)	0	-	0	R/W	否	0	-
D2261	C 轴手摇轮输入频率 (High word)	0	-	0	R/W	否	0	-
D2262	C 轴累计手摇轮输入脉冲个数 (Low word)	0	-	-	R/W	否	0	-
D2263	C 轴累计手摇轮输入脉冲个数 (High word)	0	-	-	R/W	否	0	-
D2264	C 轴手摇轮输入响应速度	-	-	-	R/W	是	5	-

## 3.11 特殊继电器及特殊寄存器群组功能说明

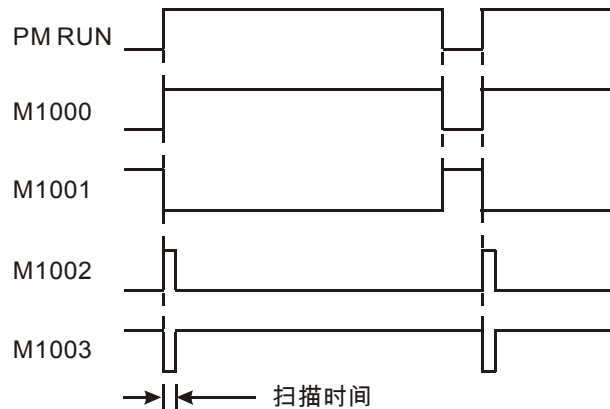
### 运转标志

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。



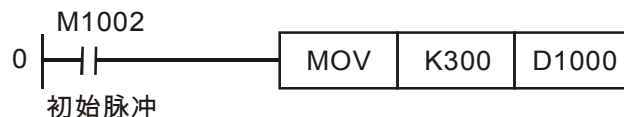
2. M1001 :M1001 为 AUTO 中时·常 Off 接点·即运转监视常闭接点 (B 接点)·DVP-PM 于 AUTO 的状态下 M1001 保持为 Off。
3. M1002 : DVP-PM 开始 AUTO 的第一次扫描 On·之后保持为 Off。该脉冲的宽度为一次扫描时间·当要作各种初始设定工作时使用本接点。
4. M1003 :DVP-PM 开始 AUTO 的第一次扫描 Off·之后一直 On·即起始反向 (AUTO 的瞬间'Off') 脉冲。



### 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。

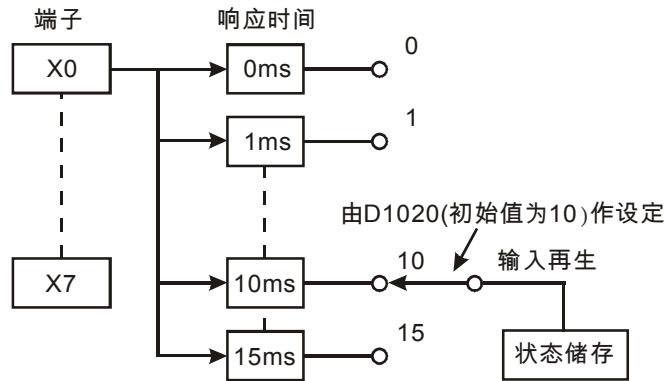
### 3 各种装置功能

#### 输入端 响应时间 的调整

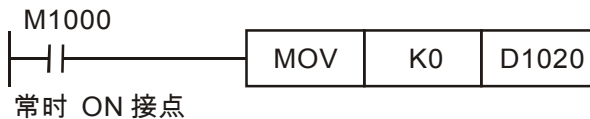
D1020

1. X0~X7 输入端，可由 D1020 的内容来设定输入端接收脉冲的响应时间，设定范围 0~20，单位 ms。

PLC 电源 Off→On 变化时，D1020 的内容自动变成 10。



2. 如果程序中执行下面的程序时，X0~X7 的响应时间被设定为 0ms，由于，输入端均串接 RC 滤波回路的关系，输入端最快的响应时间为 50μs。



3. 当程序中使用高速计数器、中断插入等功能时，不须调整响应时间

#### 通讯口 功能

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

DVP-PM 内建通讯端口 (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：

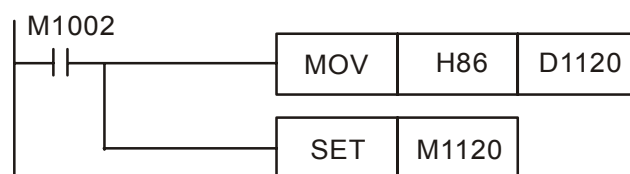
	内 容	0	1
b0	数据长度	b0=0 : 7	b0=1 : 8
b1 b2	同位	b2, b1=00 : 无 (None) b2, b1=01 : 奇同位 (Odd) b2, b1=11 : 偶同位 (Even)	
b3	stop bits	b3=0 : 1 bit	b3=1 : 2 bit
b7~b4	b7~b4=0001 (H1) :	110	bps
	b7~b4=0010 (H2) :	150	bps
	b7~b4=0011 (H3) :	300	bps
	b7~b4=0100 (H4) :	600	bps
	b7~b4=0101 (H5) :	1,200	bps
	b7~b4=0110 (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，会检测 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)



### 3 各种装置功能

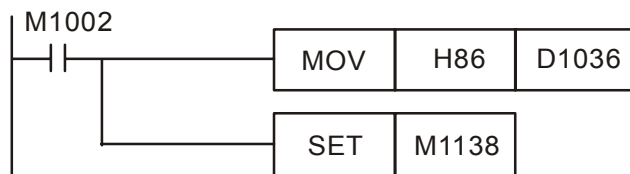
注意事项：

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恢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，会检测 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)



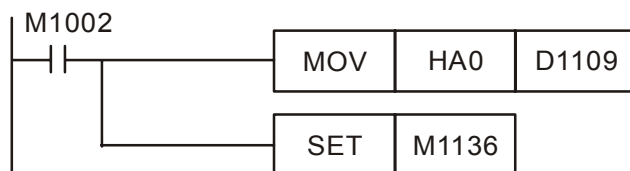
注意事项：

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。
2. 当通讯格式修改完成后，DVP-PM 电源关闭之后再上电就会恢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，会检测 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。

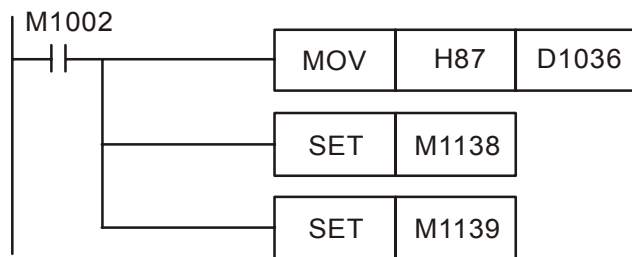
将 COM3 的通讯速率改成 38,400 bps



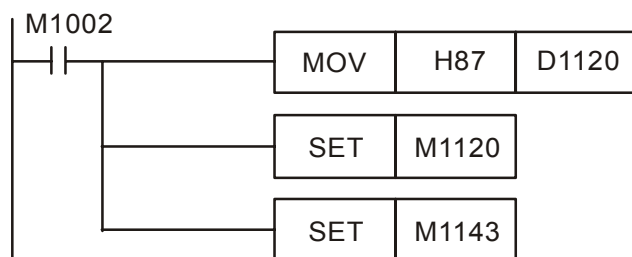
范例四：COM1、COM2 之 RTU 模式设定方式如下：

COM1：(9,600, 8, E, 1, RTU)





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



### 通讯回应 延迟

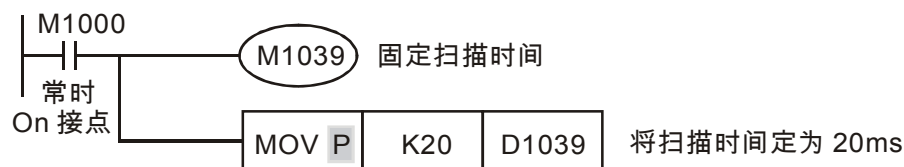
在 RS-485 界面通讯时，DVP-PM 主机当从站 (Slave) 时，数据响应延迟时间可设定，设定范围 0~3,000 (0~30 秒，时间单位为 10ms)，若不在此范围，则 D1038 = 0。设定值必须小于 D1000 的设定值。

D1038

### 固定 扫描时间

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。

M1039、D1039



2. D1010~D1012 所显示的扫描时间亦包括固定的扫描时间。

### 设定OX 程序编号

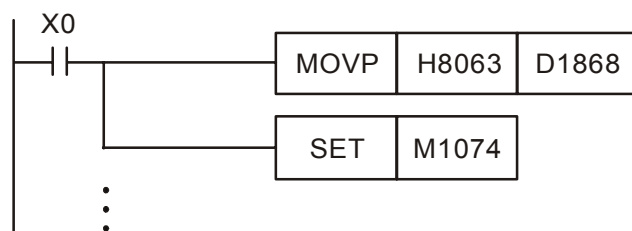
由 D1868 可指定欲执行的 OX 程序编号，其设定步骤如下：

M1074、D1868

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。

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。

### 3 各种装置功能

#### 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		清除计数外部引脚	外部输入引脚
		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-

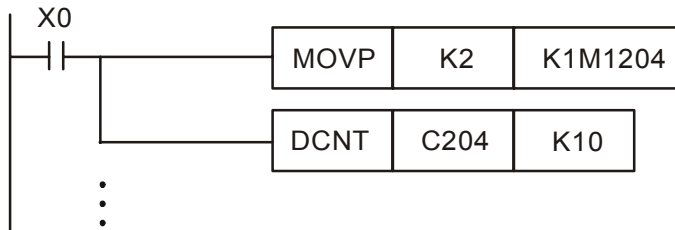
注 1: U/D: 上数/下数, P/D: 脉冲/方向, A/B: A 相/B 相

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。

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：

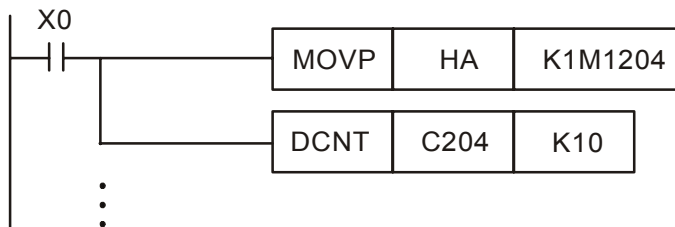


3. 若设定由外部信号清除目前计数值，则模式设定为(K1M1024=H'A)。

M1027	M1026	M1025	M1204
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

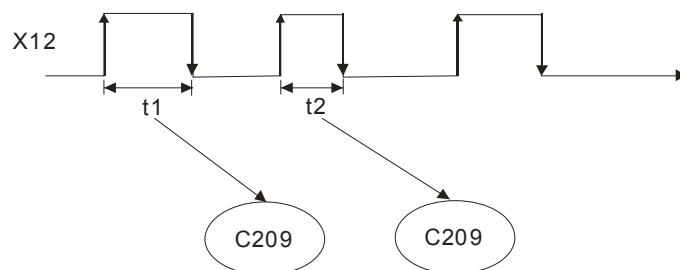
10PM 有六组高速计时功能，其设定数据如下表所示：

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

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

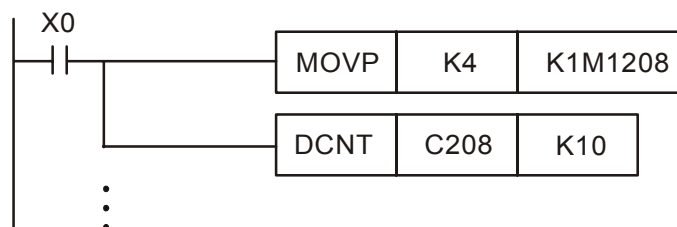
范例一：指定使用第三组定时器为一般模式

- 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。
- 启动 C208 开始计数，计算 X12 外部触发信号上升沿至信号的下降沿之间隔时间，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

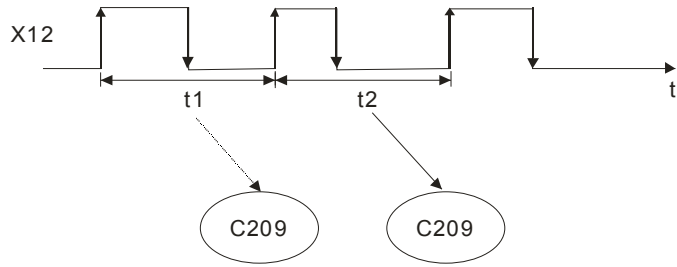
上述步骤之范例程序如下：



范例二：指定使用第三组定时器为周期模式

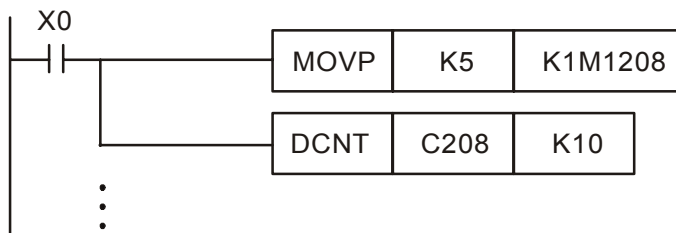
- 若设定计时方式为周期模式(Bit0=1)，则装置 K1M1208=K5。
- 启动 C208 开始计数，计算 X12 外部触发信号上升沿至下个上升沿之间隔时间，将此

### 3 各种装置功能



周期模式，单位 0.01 $\mu$ s

上述步骤之范例程序如下：



3. 周期模式用于测频应用。

#### 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，但硬件灯号不会响应。

M1304

#### 万年历时钟 RTC

1. 万年历时钟相关的特 D。

D1313~D1319

装置	名称	功能说明
D1313	秒	0~59
D1314	分	0~59
D1315	时	0~23
D1316	日	1~31
D1317	月	1~12
D1318	星期	1~7
D1319	年	0~99(公元右两位)

2. 若万年历对应的秒, 分, 时, 日, 月, 星期, 年设定值错误, 则设定错误的装置会变为秒→0, 分→0, 时→0, 日→1, 月→1, 星期→1, 年→0。
3. 万年历为停电保持, 重新上电将从最后断电之时刻继续计时。建议重新上电请重新校正万年历时钟。

#### 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 模块机种代码显示在 D1320 ~ D1327 内。
2. DVP-PM 机种特殊 I/O 模块代码：

D1320~D1327

I/O 模块名称	I/O 模块代码 (HEX)	I/O 模块名称	I/O 模块代码 (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 为中断允许寄存器, 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

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：当外部端子的输入信号于上升沿或下降沿触发时, 不受扫描周期影响, 立即中断目前执行中程序而立即跳到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到时间中断子程序执行。

### 3 各种装置功能

#### 清除执行中 M-Code

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

M1744、M1794、◆ 上述中的 M1794 为 OX M 码的被执行标志，而 D1703 为 OX 的 M 码执行寄存器。  
D1703

#### Ready 标志

1. X、Y、Z、A、B、C 轴各有 Ready 标志，X 轴: M1792, Y 轴: M1872, Z 轴: M2032, A 轴: M2112, B 轴: M2192, C 轴: M2272。可用来判断行程是不是还在运转。

M1792、M1872、2. 以下为 X 轴 Ready 标志动作说明: X 轴在运转前 M1792 为 On, 开始运转后 M1792 变  
M2032、M2112、为 Off，运转完后则变为 On。

M2192、  
M2272

#### 当前位置 写入允许

1. 欲写入 X、Y、Z、A、B、C 轴之当前位置 CP(PLS)时，可执行 M1751=1，则当前位置 D1848 的属性变更为可擦写。

M1751、M1831、2. 若在 M1751=0 的状态下写入 D1848，则 D1848 的数值清除为 0。

M1991、M2071、

M2151、  
M2231

#### 运动错误 清除

1. 当 X 轴、Y 轴、Z 轴、A 轴、B 轴、C 轴发生运动错误时，错误标志分别为 M1793、M1873、M2033、M2113、M2193、M2273，且会将错误信息分别存放至 D1857、D1937、D2017、D2097、D2177、D2257 寄存器之中。

M1793、D1857

M1873、D1937 2. 欲进行除错时，请清除错误信息寄存器与错误标志。

M2033、D2017

M2113、D2097

M2193、D2177

M2273、D2257

## 设定输入端子极性

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

D1799

bit#	输入端子极性	bit#	输入端子极性
0	X0 (DOG0)	8	X10 (MPGA)
1	X1 (PG0)	9	X11 (MPGB)
2	X2 (DOG1)	10	X12 (DOG4)
3	X3 (PG1)	11	X13 (DOG5)
4	X4 (DOG2)	12	⋮
5	X5 (PG2)	13	⋮
6	X6 (DOG3)	14	⋮
7	X7 (PG3)	15	⋮

## 读取输入端子状态

当 bit# 为 ON 时，表示有信号输入。反之，则当 bit# 为 OFF 时，表示无信号输入。

D1800

bit#	输入端子状态	bit#	输入端子状态
0	X0 (DOG0)	8	X10 (MPGA)
1	X1 (PG0)	9	X11 (MPGB)
2	X2 (DOG1)	10	X12 (DOG4)
3	X3 (PG1)	11	X13 (DOG5)
4	X4 (DOG2)	12	⋮
5	X5 (PG2)	13	⋮
6	X6 (DOG3)	14	⋮
7	X7 (PG3)	15	⋮

## 输入端子滤波系数设定

1. 当 X0~X7 输入端子作为计数/计时输入时，可由 D1806 之 Low Byte 内容设定输入端之滤波系数。

D1806

2. 10PM 手摇轮输入滤波设计，可由 D1806 之 Low Byte 内容设定输入端之滤波系数。

3. 滤波系数 =  $\frac{85000}{2^{N+4}}$  (KHz), N=1~19；下表为换算之滤波值。

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

### 3 各种装置功能

8	20.75195	18	0.020266
9	10.37598	19	0.010133
10	5.187988		

4. 当 D1806 设定为 0 时，表示不启动外部端子之滤波功能。
5. 举例说明，D1806=H000A 表示设定输入端子 X0~X7(作计数/计时功能时)及 MPG0/1

之滤波系数为  $= \frac{85000}{2^{10+4}} = 5.187988 \text{ (KHz)}$  ; 输入频率高于 5.187988KHz 之信号将会被

滤除。

#### O100 错误发生检查

1. 当 O100 程序发生错误时，O100 错误标志 M1953 会 SET On，错误信息会被存在 D1802，错误 STEP 位置会存放在 D1803。

M1953、D1802、D1803 2. 错误信息请参阅第 15 章附录 A 的错误信息表。

#### 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，D2136 为 B 轴参数设定，D2216 为 C 轴参数设定。说明如下：

D1816、D1896、  
D1976、D2056、  
D2136、  
D2216

bit#	X-Y-Z-A-B-C 轴参数设定	bit#	X-Y-Z-A-B-C 轴参数设定
0	单位系设定 (注一)	8	原点回归方向 (注四)
1		9	原点回归模式 (注四)
2	位置数据倍率设定 (注二)	10	原点回归 DOG 下降沿检测 (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	电机单位
0	1	机械单位
1	0	复合单位
1	1	

	电机单位	复合单位	机械单位
位置	pulse	um	
	pulse	m deg	
	pulse	10 <sup>-4</sup> inch	
速度	pulse/sec		cm/min
	pulse/sec		10deg/min
	pulse/sec		inch/min



注二：

b3	b2	位置数据倍率设定
0	0	10 <sup>0</sup>
0	1	10 <sup>1</sup>
1	0	10 <sup>2</sup>
1	1	10 <sup>3</sup>

注三：

b5	b4	说明
0	0	正向脉冲+反向脉冲
0	1	脉冲+方向
1	0	A/B 相脉冲 (2 相 2)
1	1	

注四：

bit#	说明
6	b[6]=1：启用 PWM 模式 (1) 运转命令中启动 JOG+运转，则 Y0~Y3 输出 PWM (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 (3) 脉冲宽度设定：D1838、D1918、D1998、D2078 (4) 脉冲频率设定：D1842、D1922、D2002、D2082
8	b[8]=0：执行原点回归方向，往当前位置 (CP) 内容值递减方向执行 b[8]=1：执行原点回归方向，往当前位置 (CP) 内容值递增方向执行
9	b[9]=0：正常模式；b[9]=1：复写模式
10	b[10]=0：原点回归 DOG 下降沿检测 b[10]=1：原点回归 DOG 上升沿检测
11	b[11]=0：当正转运动时，当前位置 (CP) 内容值递增 b[11]=1：当正转运动时，当前位置 (CP) 内容值递减
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	b[13]=0：DOG 信号上升沿触发 b[13]=1：DOG 信号下降沿触发 (插入单段速定位运动模式、插入二段速定位运动模式有效)
14	b[14]=0：为加速采用梯形加速曲线 b[14]=1：为加速采用 S 加速曲线

### 轴运转命令 设定

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，D2166 为 B 轴运转命令设定，D2246 为 C 轴运转命令设定。说明如下：

- D1846 、
- D1926 、
- D2006 、
- D2086 、
- D2166 、
- D2246

bit#	X-Y-Z-A-B-C 轴运转命令	bit#	X-Y-Z-A-B-C 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	-
6	原点回归模式启动	14*2	-
7*1	单段速外部触发	15	-

### 3 各种装置功能

轴工作模式  
设定

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，  
D2087 为 A 轴工作模式设定，D2167 为 B 轴工作模式设定，D2247 为 C 轴工作模式设定。  
说明如下：

D1847、  
D1927、  
D2007、  
D2087、  
D2167、  
D2247

bit#	X-Y-Z-A-B-C 轴工作模式	bit#	X-Y-Z-A-B-C 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	-
4	CLR 极性设定	12	
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 为原点回归完成时输出 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：电机运转中，碰到 STOP 信号输入时，电机减速停，再度下达运动命令时，电机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：电机运转中，碰到 STOP 信号输入时，电机减速停，再度下达运动命令时，电机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：电机运转中，碰到 LSP/LSN 信号输入时，电机减速停止。 b[7]=1：电机运转中，碰到 LSP/LSN 信号输入时，电机立即停止。
8	停止信号设定 (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，允许 $\Phi A_{\pm}$ 或 $\Phi B_{\pm}$ 之触发信号为停止信号。
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 $\Phi A_{\pm}$ 信号上升沿触发则立即停止
	b[10~8]=K2 (010)：输入端 $\Phi A_{\pm}$ 信号下降沿触发则立即停止
10	b[10~8]=K3 (011)：输入端 $\Phi B_{\pm}$ 信号上升沿触发则立即停止
	b[10~8]=K4 (100)：输入端 $\Phi B_{\pm}$ 信号下降沿触发则立即停止

**轴执行状态  
显示**

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，D2176 为 B 轴执行状态，D2156 为 C 轴执行状态。说明如下：

D1856 、  
D1936 、  
D2016 、  
D2096 、  
D2176 、  
D2256

bit#	X-Y-Z-A-B-C 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	-

# 3 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 <sup>*1</sup>	LW <sup>*1</sup>	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	电机转一圈所需脉冲数(A)	1 ~ +2,147,483,647 PLS/REV	K2,000
D1821	D1820	D1901	D1900	D1981	D1980	D2061	D2060	电机转一圈之移动距离(B)	1 ~ +2,147,483,647 <sup>*2</sup>	K1,000
D1823	D1822	D1903	D1902	D1983	D1982	D2063	D2062	最高速度	0 ~ +2,147,483,647 <sup>*3</sup>	K500,000
D1825	D1824	D1905	D1904	D1985	D1984	D2065	D2064	启动速度	0 ~ +2,147,483,647 <sup>*3</sup>	K0
D1827	D1826	D1907	D1906	D1987	D1986	D2067	D2066	寸动 JOG 速度 V <sub>JOG</sub>	0 ~ +2,147,483,647 <sup>*3</sup>	K5,000
D1829	D1828	D1909	D1908	D1989	D1988	D2069	D2068	原点回归速度 V <sub>RT</sub>	0 ~ +2,147,483,647 <sup>*3</sup>	K50,000
D1831	D1830	D1911	D1910	D1991	D1990	D2071	D2070	原点回归减速速度 V <sub>CR</sub>	0 ~ +2,147,483,647 <sup>*3</sup>	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 <sup>*1</sup>	K0
-	D1836	-	D1916	-	D1996	-	D2076	加速时间 T <sub>ACC</sub>	10 ~ +32,767 ms	K100
-	D1837	-	D1917	-	D1997	-	D2077	减速时间 T <sub>DEC</sub>	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 <sup>*1</sup>	K0
D1841	D1840	D1921	D1920	D2001	D2000	D2081	D2080	运转速度(I)V(I)	0 ~ +2,147,483,647 <sup>*1</sup>	K1000
D1843	D1842	D1923	D1922	D2003	D2002	D2083	D2082	目标位置(II)P(II)	-2,147,483,648 ~ +2,147,483,647 <sup>*1</sup>	K0
D1845	D1844	D1925	D1924	D2005	D2004	D2085	D2084	运转速度(II)V(II)	0 ~ +2,147,483,647 <sup>*2</sup>	K2,000
-	D1846	-	D1926	-	D2006	-	D2086	运转命令	b0 ~ b15	H0
-	D1847	-	D1927	-	D2007	-	D2087	工作模式	b0 ~ b15	H0
D1849	D1848	D1929	D1928	D2009	D2008	D2089	D2088	当前位置 CP(PLS)	-2,147,483,648 ~ +2,147,483,647 <sup>*1</sup>	K0
D1851	D1850	D1931	D1930	D2011	D2010	D2091	D2090	当前速度 CS(PPS)	0 ~ +2,147,483,647 PPS	K0
D1853	D1852	D1933	D1932	D2013	D2012	D2093	D2092	当前位置 CP(unit *3)	-2,147,483,648 ~ +2,147,483,647 <sup>*1</sup>	K0
D1855	D1854	D1935	D1934	D2015	D2014	D2095	D2094	当前速度 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 <sup>*1</sup>	LW <sup>*1</sup>	HW	LW	HW	LW	HW	LW			
-	-	-	-	-	-	-	-	停止模式 (OX0-99)	依需求设定对应值	K0
D1867	D1866	D1947	D1946	D2027	D2026	-	-	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 码 输出 high bye	依需求设定对应值	K0
D1874	-	-	-	-	-	-	-	M 码 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

特 D 编号				特殊寄存器名称	设定范围	出厂值
B 轴		C 轴				
HW	LW	HW	LW			
-	D2136	-	D2216	参数设定	b0 ~ b15	H0
D2139	D2138	D2219	D2218	电机转一圈所需脉冲数(A)	1 ~ +2,147,483,647 PLS/REV	K2,000
D2141	D2140	D2221	D2220	电机转一圈之移动距离(B)	1 ~ +2,147,483,647 <sup>*2</sup>	K1,000
D2143	D2142	D2223	D2222	最高速度	0 ~ +2,147,483,647 <sup>*3</sup>	K500,000
D2145	D2144	D2225	D2224	启动速度	0 ~ +2,147,483,647 <sup>*3</sup>	K0
D2147	D2146	D2227	D2226	寸动 JOG 速度 V <sub>JOG</sub>	0 ~ +2,147,483,647 <sup>*3</sup>	K5,000
D2149	D2148	D2229	D2228	原点回归速度 V <sub>RT</sub>	0 ~ +2,147,483,647 <sup>*3</sup>	K50,000
D2151	D2150	D2231	D2230	原点回归减速速度 V <sub>CR</sub>	0 ~ +2,147,483,647 <sup>*3</sup>	K1,000
-	D2152	-	D2232	零点(PG0)信号数 N	0 ~ +32,767 PLS	K0
-	D2153	-	D2233	脉冲信号数 P	-32,768 ~ +32,767 PLS	K0
D2155	D2154	D2235	D2234	原点位置定义 HP	0 ~ ±999,999 <sup>*1</sup>	K0
-	D2156	-	D2236	加速时间 T <sub>ACC</sub>	10 ~ +32,767 ms	K100
-	D2157	-	D2237	减速时间 T <sub>DEC</sub>	10 ~ +32,767 ms	K100
D2159	D2158	D2239	D2238	目标位置(I)P(I)	-2,147,483,648 ~ +2,147,483,647 <sup>*1</sup>	K0
D2161	D2160	D2241	D2240	运转速度(I)V(I)	0 ~ +2,147,483,647 <sup>*1</sup>	K1000
D2163	D2162	D2243	D2242	目标位置(II)P(II)	-2,147,483,648 ~ +2,147,483,647 <sup>*1</sup>	K0
D2165	D2164	D2245	D2244	运转速度(II)V(II)	0 ~ +2,147,483,647 <sup>*2</sup>	K2,000
-	D2166	-	D2246	运转命令	b0 ~ b15	H0
-	D2167	-	D2247	工作模式	b0 ~ b15	H0
D2169	D2168	D2249	D2248	当前位置 CP(PLS)	-2,147,483,648 ~ +2,147,483,647 <sup>*1</sup>	K0
D2171	D2170	D2251	D2250	当前速度 CS(PPS)	0 ~ +2,147,483,647 PPS	K0
D2173	D2172	D2253	D2252	当前位置 CP(unit *3 )	-2,147,483,648 ~ +2,147,483,647 <sup>*1</sup>	K0

### 3 各种装置功能

特 D 编号				特殊寄存器名称	设定范围	出厂值
B 轴		C 轴				
HW	LW	HW	LW			
D2175	D2174	D2255	D2254	当前速度 CS(unit *3)	0 ~ +2,147,483,647 PPS	K0
-	D2176	-	D2256	执行状态	b0 ~ b15	H0
-	D2177	-	D2257	错误编号	请参考错误码表	H0
-	D2178	-	D2258	电子齿轮 (分子)	1 ~ +32,767	K1
-	D2179	-	D2259	电子齿轮 (分母)	1 ~ +32,767	K1
D2181	D2180	D2261	D2260	手摇轮输入频率	由手摇轮输入的脉冲频率	K0
D2183	D2182	D2263	D2262	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
-	D2184	-	D2264	手摇轮输入响应速度	手摇轮输入响应速度	K5
-	-	-	-	停止模式 (OX0-99)	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<sup>-4</sup> 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		参数设定
HW	LW	HW	LW	HW	LW	
	D1816		D1896		D1976	
A 轴		B 轴		C 轴		
HW	LW	HW	LW	HW	LW	
	D2056		D2136		D2216	

[说明]

bit0~bit15 各位所代表的意义如下所示：

1. D1816 (D1896, D1976, D2056, D2136, D2216) 之 b0, b1 : 设定单位系

b1	b0	单位系	说明
0	0	电机单位	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 (电机单位) 为单位
1	1		

	电机单位	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)】、当前位置【CP】。
- 速度数据：最高速度【V<sub>MAX</sub>】、启动速度【V<sub>BIAS</sub>】、寸动 JOG 速度【V<sub>JOG</sub>】、原点回归速度【V<sub>RT</sub>】、原点减速速度【V<sub>CR</sub>】、运转速度(I)【V(I)】、运转速度(II)【V(II)】。

• 范例一：

电机单位 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)

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，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, D2058, D2138, D2218) = 1,000 (Pulse/REV) · D1820 (D1900, D1980, D2060, D2140, D2220) = 100 (um/REV) · 且目标位置 P(I)=10,000 (um) · 运转速度 V(I)=6 (cm/min) · 位置控制器的脉冲指令的脉冲个数及其频率为：离

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

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

运转速度 V(I)：6 (cm/min) = 60,000/60 (um/sec)

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

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

• 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, D2058, D2138, D2218) = 2,000 (Pulse/REV) · D1820 ((D1900, D1980, D2060, D2140, D2220)) = 100 (um/REV) · 若目标位置 P(I)：10,000 (um) · 运转速度 V(I)：10K (PPS) · 位置控制器的脉冲指令的脉冲个数为：由位置控制器计算要跑到 P(I) 所需之脉冲数

$$= \frac{P(I) \text{um}}{B/A} = P(I) \times \frac{A}{B} = 200,000 \text{ PULSE}$$







### 3 各种装置功能

2. D1816 (D1896, D1976, D2056, D2136, D2216) 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)】、当前位置【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 <sup>0</sup>
0	1	位置数据 × 10 <sup>1</sup>
1	0	位置数据 × 10 <sup>2</sup>
1	1	位置数据 × 10 <sup>3</sup>

3. D1816 (D1896, D1976, D2056, D2136, D2216) 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	

4. D1816 (D1896, D1976, D2056, D2136, D2216) 之 b6 : PWM 模式设定

b[6]=1 时·启动 JOG+运转·则 Y0~Y3 输出 PWM

5. D1816 (D1896, D1976, D2056, D2136, D2216) 之 b8 : 原点回归方向

b[8]=0 时·执行原点回归方向·往当前位置 (CP) 内容值反 (递减) 方向执行。

b[8]=1 时·执行原点回归方向·往当前位置 (CP) 内容值正 (递增) 方向执行。

6. D1816 (D1896, D1976, D2056, D2136, D2216) 之 b9 : 原点回归模式

b[9]=0 时·正常模式。即为检测近点 (DOG) 信号产生后·经过 N 个原点回归零点信号后·再经过 P 个原点回归脉冲信号后·电机立刻停止。

b[9]=1 时·复写模式。即为检测近点 (DOG) 信号产生后·当零点信号数 N 脉冲及脉冲信号数 P 脉冲其中一项先到达时·电机立刻停止。

7. D1816 (D1896, D1976, D2056, D2136, D2216) 之 b10 : 原点回归 DOG 下降沿检测

b[10]=0 时·原点回归 DOG 下降沿检测 On·即为 DOG 下降沿检测。

b[10]=1 时·原点回归 DOG 下降沿检测 Off·即为 DOG 上升沿检测。

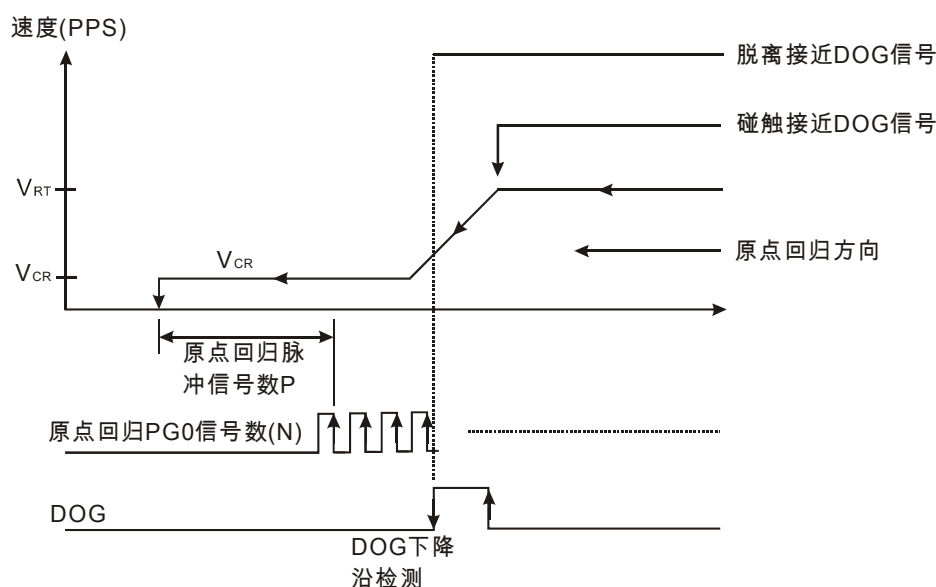
- b[9:10]=00 ⇒ 正常模式·原点回归 DOG 下降沿检测 On

原点回归动作：电机以原点回归速度 VRT 运转·当碰到近点 (DOG) 信号时·电机减速到以原点回归减速速度 VCR 运转·再检测近点 (DOG) 信号下降沿产生后·经过 N 个原点回归之零点信号脉冲后·再经过 P 个原点回归之脉冲信号脉冲后电机停止。



若设定原点回归之 PG0 零点信号数 N 或原点回归之脉冲信号数 P 太小时，碰到近点 (DOG) 信号时，电机开始向原点回归减速速度  $V_{CR}$  做减速动作，再检测近点 (DOG) 信号下降沿产生后，若达到指定之 N 个 PG0 零点信号，再经过 P 个原点回归之脉冲信号立即停止 (不论是否已到达原点回归减速速度  $V_{CR}$ )。

假设原点回归之信号数 N 设定为 0，原点回归之脉冲信号数 P 设定为 0，则碰到近点 (DOG) 信号后再检测近点 (DOG) 信号下降沿产生后立即停止。

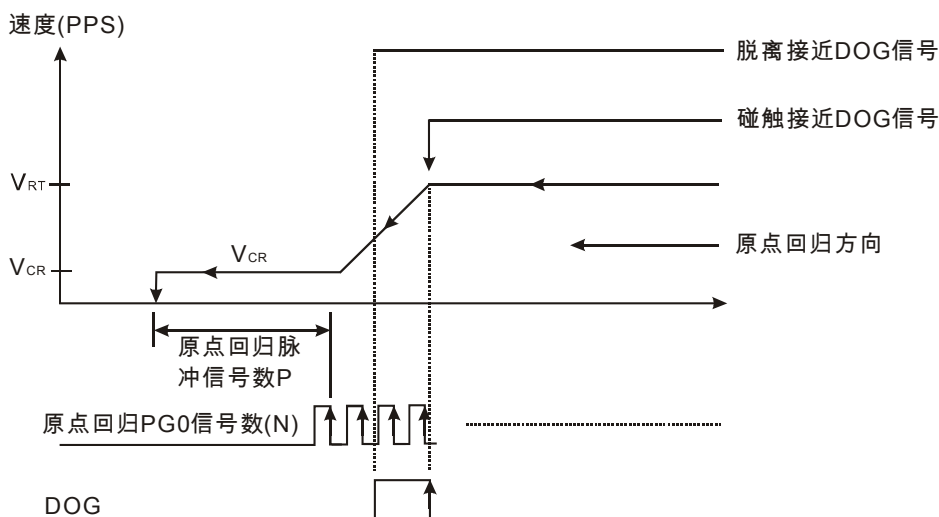


- $b[9:10]=01 \Rightarrow$  正常模式，原点回归 DOG 下降沿检测 Off

原点回归动作：电机以原点回归速度  $V_{RT}$  运转，当碰到近点 (DOG) 信号时，电机开始向原点回归减速速度  $V_{CR}$  做减速动作，经过 N 个原点回归之零点信号脉冲后，再经过 P 个原点回归之脉冲信号脉冲后电机停止。

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

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



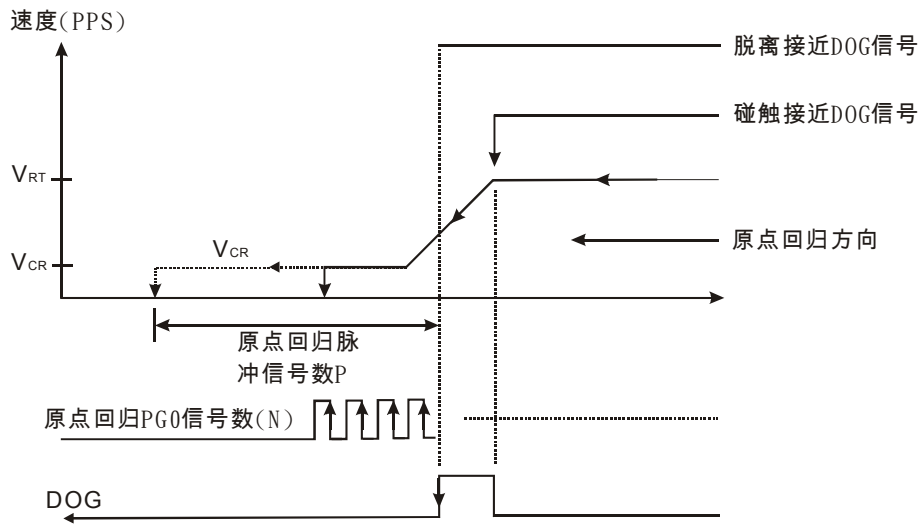
### 3 各种装置功能

- b[9:10]=10⇒复写模式，原点回归 DOG 下降沿检测 On

原点回归动作：电机以原点回归速度  $V_{RT}$  运转，当碰到近点 (DOG) 信号时，电机开始向原点回归减速速度  $V_{CR}$  做减速动作，再检测近点 (DOG) 信号下降沿产生后，此时经过  $N$  个原点回归之零点信号脉冲后，或经过  $P$  个原点回归之脉冲信号脉冲后电机停止。

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

假设原点回归之信号数  $N$  设定为 0，原点回归之脉冲信号数  $P$  设定为 0，则电机检测近点 (DOG) 信号下降沿产生后立即停止。

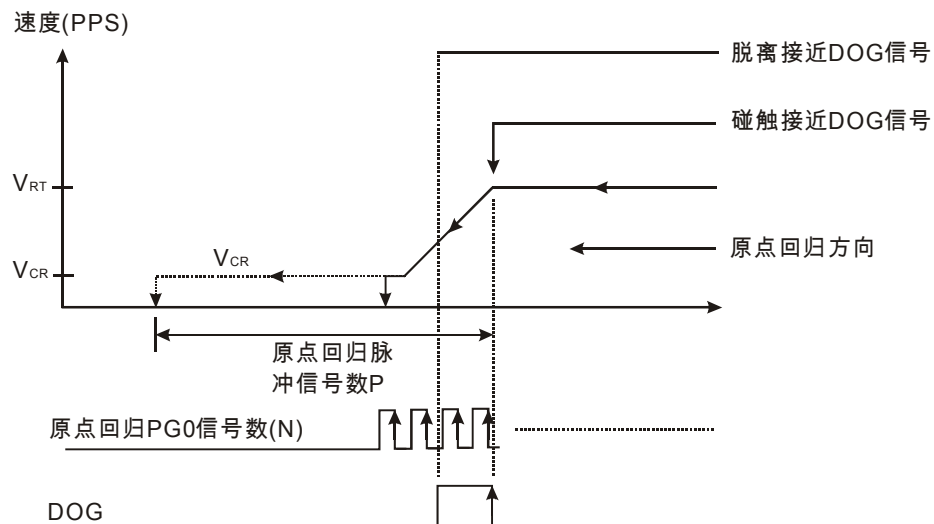


- b[9:10]=11⇒复写模式，原点回归 DOG 下降沿检测 Off

原点回归动作：电机以原点回归速度  $V_{RT}$  运转，当碰到近点 (DOG) 信号时，电机开始向原点回归减速速度  $V_{CR}$  做减速动作，此时经过  $N$  个原点回归之零点信号脉冲后，或经过  $P$  个原点回归之脉冲信号后电机停止。

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

假设原点回归之信号数  $N$  设定为 0，原点回归之脉冲信号数  $P$  设定为 0，或电机碰到近点 (DOG) 信号后立即停止。



8. D1816 (D1896, D1976, D2056, D2136, D2216) 之 b11 : 旋转方向
  - b[11]=0 时，当正转运动时，当前位置 (CP) 内容值递增。
  - b[11]=1 时，当正转运动时，当前位置 (CP) 内容值递减。
9. D1816 (D1896, D1976, D2056, D2136, D2216) 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。
10. D1816 (D1896, D1976, D2056, D2136, D2216) 之 b13 : DOG 触发启动模式
  - b[13]=0 时，DOG 信号上升沿触发。
  - b[13]=1 时，DOG 信号下降沿触发 (插入单段速定位运动模式、插入二段速定位运动模式有效)。
11. D1816 (D1896, D1976, D2056, D2136, D2216) 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		电机转一圈所需脉冲数(A)
HW	LW	HW	LW	HW	LW	
D1819	D1818	D1899	D1898	D1979	D1978	
A 轴		B 轴		C 轴		
HW	LW	HW	LW	HW	LW	
D2059	D2058	D2139	D2138	D2219	D2218	

[说明]

1. 由于伺服驱动器内具有电子齿轮比例设定功能，因此伺服电机转一圈所需之脉冲数并不一定等于伺服电机旋转一圈，编码器所产生的脉冲数：  
电机转一圈所需指令脉冲数  $A \times$  电子齿轮  $(CMX / CDV) =$  编码器旋转一圈产生脉冲数
2. 设定单位依照 D1816 (D1896, D1976, D2056, D2136, D2216) 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；若系统单位设定为电机单位，则参数 A 设定不作用。

### 3 各种装置功能

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1821	D1820	D1901	D1900	D1981	D1980
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2061	D2060	D2141	D2140	D2221	D2220

电机转一圈之移动距离(B)

[说明]

1. 电机转一圈所带动之移动量，单位有三种可供选择，由 D1816 (D1896, D1976, D2056, D2136, D2216) 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, D2136, D2216) 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；若系统单位设定为电机单位，则参数 B 设定不作用。

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1823	D1822	D1903	D1902	D1983	D1982
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2063	D2062	D2143	D2142	D2223	D2222

最高速度  $V_{MAX}$

[说明]

1. 各种操作模式的速度上限。设定范围  $0 \sim +2,147,483,647$  (单位系由 D1816 (D1896, D1976, D2056, D2136, D2216) 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	
HW	LW	HW	LW	HW	LW
D1825	D1824	D1905	D1904	D1985	D1984
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2065	D2064	D2145	D2144	D2225	D2224

启动速度  $V_{BIAS}$

[说明]

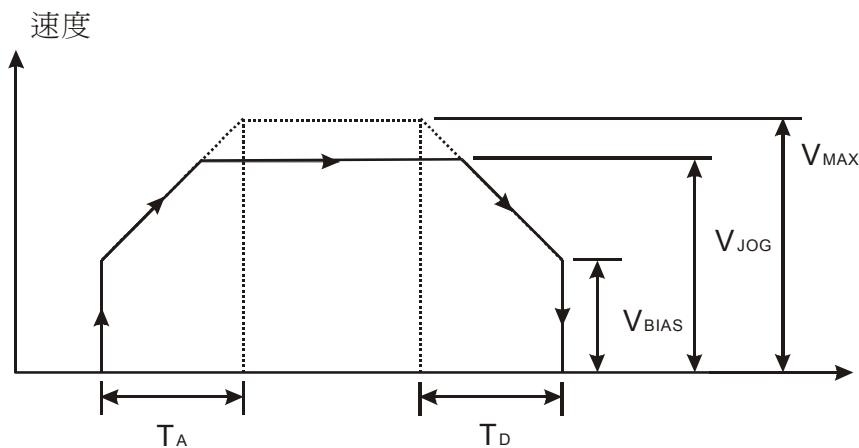
1. 脉冲输出启动速度。设定范围  $0 \sim +2,147,483,647$  (单位由 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b0, b1 设定)。
2. 对应运动指令  $0 \sim 500kPPS$ 。若大于 500k，则以 500k 输出，若小于 0，则以 0 输出。
3. 若为步进驱动系统，请注意步进电机的共振区频率，将启动速度  $V_{BIAS}$  设定在共振区频率之上，作安全启动。

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1827	D1826	D1907	D1906	D1987	D1986
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2067	D2066	D2147	D2146	D2227	D2226

寸动 JOG 速度  $V_{JOG}$

[说明]

1. 寸动指令速度。设定范围  $0 \sim +2,147,483,647$  (单位由 D1816(D1896, D1976, D2056, D2136, D2216) 之 b0, b1 设定)。
2. 对应的脉冲指令  $10 \sim 500\text{kPPS}$ 。若大于  $500\text{k}$ ，则以  $500\text{k}$  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  $= V_{BIAS}$ 。(跳 ERROR)
4. 寸动 JOG 速度  $V_{JOG}$  在执行中不允许改变。



X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1829	D1828	D1909	D1908	D1989	D1988
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2069	D2068	D2149	D2148	D2229	D2228

原点回归速度  $V_{RT}$

[说明]

1. 回到机械原点的速度。设定范围  $0 \sim +2,147,483,647$  (单位由 D1816(D1896, D1976, D2056, D2136, D2216) 之 b0, b1 设定)。
2. 对应的脉冲指令  $10 \sim 500\text{kPPS}$ 。若大于  $500\text{k}$ ，则以  $500\text{k}$  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. 原点回归速度  $V_{RT}$  在执行中不允许改变。

### 3 各种装置功能

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1831	D1830	D1911	D1910	D1991	D1990
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2071	D2070	D2151	D215	D2231	D2230

原点回归减速速度  $V_{CR}$

[说明]

1. 设定范围 0~+2,147,483,647 (单位由 D1816(D1896, D1976, D2056, D2136, D2216) 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. 执行原点回归时，电机以原点回归速度  $V_{RT}$  运转，当碰到 DOG 信号时，电机减速到原点回归减速速度  $V_{CR}$  运转。
4. 为了精准的定位在原点，建议将原点回归减速速度  $V_{CR}$  设定为低速运行。
5. 原点回归减速速度  $V_{CR}$  在执行中不允许改变。

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
	D1832		D1912		D1992
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
	D2072		D2152		D2232

原点回归之零点(PG0)信号数 N

[说明]

1. 设定范围 -32,768~32,767 (PULSE)。正数设定为正方向的零点信号数 N，负数设定为反方向的零点信号数 N。
2. 电机减速停止之参考信号，请参考 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b9, b10 原点回归模式之说明。

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
	D1833		D1913		D1993
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
	D2073		D2153		D2233

原点回归之脉冲信号数 P

[说明]

1. 设定范围  $-32,768 \sim 32,767$  (PULSE)。正数设定为正方向的脉冲信号数 P。负数设定为反方向的脉冲信号数 P。
2. 电机减速停止之参考信号。请参考 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b9, b10 原点回归模式之说明

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1835	D1834	D1915	D1914	D1995	D1994
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2075	D2074	D2155	D2154	D2235	D2234

原点位置定义 HP

[说明]

1. 设定范围  $0 \sim \pm 999,999$  (单位由 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b0, b1 定义)。
2. 原点回归完成时。当前位置 CP 会被更新为原点位置 HP。

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
	D1836		D1916		D1996
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
	D2076		D2156		D2236

加速时间  $T_{ACC}$

[说明]

1. 从 D1824 (D1904, D1984, D2064, D2144, D2224) 启动速度  $V_{BIAS}$  加速到 D1822 (D1902, D1982, D2062, D2142, D2222) 最高速度  $V_{MAX}$  所需的时间。设定值范围  $0 \sim 32767$ 。单位为 ms。
2. 当设定值  $< 10$  时。则视为 10。设定值  $> 32,767$  时。则视为 32,767。
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	
HW	LW	HW	LW	HW	LW
	D1837		D1917		D1997
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
	D2077		D2157		D2237

减速时间  $T_{DEC}$

### 3 各种装置功能

[说明]

1. 从 D1822 (D1902, D1982, D2062, D2142, D2222) 最高速度  $V_{MAX}$  减速到 D1824 (D1904, D1984, D2064, D2144, D2224) 启动速度  $V_{BIAS}$  所需的时间 · 设定值范围 0~32767 · 单位为 ms。
2. 当设定值 < 10 时 · 则视为 10 · 设定值 > 32,767 时 · 则视为 32,767。
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	
HW	LW	HW	LW	HW	LW
D1839	D1838	D1919	D1918	D1999	D1998
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2079	D2078	D2159	D2158	D2239	D2238

目标位置(I) P(I)

[说明]

1. 设定范围 -2,147,483,648~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b0, b1 定义)。
2. 目标位置 P(I)数据属性：
  - 绝对坐标 (D1816 (D1896, D1976, D2056, D2136, D2216) 之 b12=0)  
代表从 0 开始之位置 · 当目标位置(I) P(I)大于当前位置 (D1848 (D1928, D2008, D2088, D2168, D2248)) 时 · 电机正转 · 小于当前位置时 · 电机反转。
  - 相对坐标 (D1816 (D1896, D1976, D2056, D2136, D2216) 之 b12=1)  
代表从当前位置 (D1848 (D1928, D2008, D2088, D2168, D2248)) 开始算电机行走之距离 · 当相对坐标为正数时 · 电机正转 · 相对坐标为负数时 · 电机反转。
3. 目标位置(I) P(I)数据倍率依照 D1816 (D1896, D1976, D2056, D2136, D2216) 参数设定之 b2, b3 倍率设定来变化。

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1841	D1840	D1921	D1920	D2001	D2000
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2081	D2080	D2161	D2160	D2241	D2240

运转速度(I) V(I)

[说明]

1. 设定范围 -2,147,483,647~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056, D2136, D2216) 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, D2166, D2246) 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	
HW	LW	HW	LW	HW	LW
D1843	D1842	D1923	D1922	D2003	D2002
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2083	D2082	D2163	D2162	D2243	D2242

目标位置(II) P(II)

[说明]

1. 设定范围 -2,147,483,648~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b0, b1 定义)。
2. 目标位置 P(II) 数据属性：
  - 绝对坐标 (D1816 (D1896, D1976, D2056, D2136, D2216) 之 b12=0)  
代表从 0 开始之位置 · 当目标位置(II) P(II)大于当前位置 (D1848 (D1928, D2008, D2088, D2168, D2248) ) 时 · 电机正转 · 小于当前位置时 · 电机反转。
  - 相对坐标 (D1816 (D1896, D1976, D2056, D2136, D2216) 之 b12=1)  
代表从当前位置 (D1848 (D1928, D2008, D2088, D2168, D2248) ) 开始算电机行走之距离 · 当相对坐标为正数时 · 电机正转 · 相对坐标为负数时 · 电机反转。
3. 目标位置(II) P(II) 数据倍率依照 D1816 (D1896, D1976, D2056, D2136, D2216) 参数设定之 b2, b3 倍率设定来变化。

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1845	D1844	D1925	D1924	D2005	D2004
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2085	D2084	D2165	D2164	D2245	D2244

运转速度(II)V(II)

[说明]

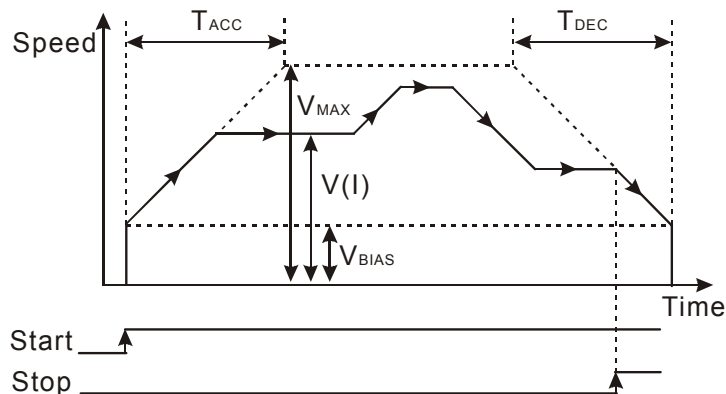
1. 设定范围 0~+2,147,483,647 (单位由 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b0, b1 定义)。
2. 对应的脉冲指令 10~500kPPS。若大于 500k · 则以 500k 输出 · 若小于 10 · 则以 10 输出。
3. 设定范围限制： $V_{MAX} > V(II) > V_{BIAS}$ 。

### 3 各种装置功能

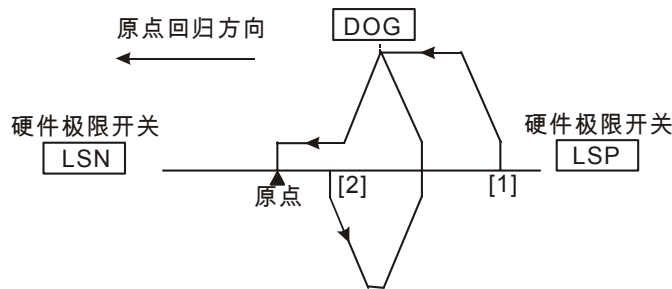
X 轴		Y 轴		Z 轴		运转命令
HW	LW	HW	LW	HW	LW	
	D1846		D1926		D2006	
A 轴		B 轴		C 轴		
HW	LW	HW	LW	HW	LW	
	D2086		D2166		D2246	

[说明]

1. D1846 (D1926, D2006, D2086, D2166, D2246) 之 b0 : 软件 STOP 。
  - 动作时机 : b[0]上升沿触发(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。
2. D1846 (D1926, D2006, D2086, D2166, D2246) 之 b1 : 软件 START 。
  - 当 b[1]上升沿触发(0→1)时, 开始运行启动, 根据 D1846 (D1926, D2006, D2086, D2166, D2246) 运转命令设定运作。
3. D1846 (D1926, D2006, D2086, D2166, D2246) 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-B-C 轴参数设定(D1816、D1896、D1976、D2056、D2136、D2216)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, D2166, D2246) 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, D2166, D2246) 之 b4 : 变速度运转模式 。
  - b[4]=1 启动变速度运动, 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: 运转时, 由启动速度  $V_{BIAS}$  开始加速运转速度 V(I)后稳定运行, 在运转期间, 使用者可以任意的更改运转速度 V(I) DVP-PM 脉冲产生单元会依照运转速度 V(I)的设定值加速 / 减速。
  - 注意, 无法使用外部 STOP 输入接点信号停止变速度运动; 停止方法为设定软件 STOP(b[0]=1), 或是将 b4 设定为 0 即可停止。
  - 动作示意图:



6. D1846 (D1926, D2006, D2086, D2166, D2246) 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, D2098~D2104, D2178~D2184, D2258~D2264)。
7. D1846 (D1926, D2006, D2086, D2166, D2246) 之 b6：原点回归模式启动。
- 当 b[6]=0→1 变化时，开始执行原点回归动作，由于当前位置 (CP) 所处的位置不同，执行原点回归动作也有所不同，可分为二种情况：

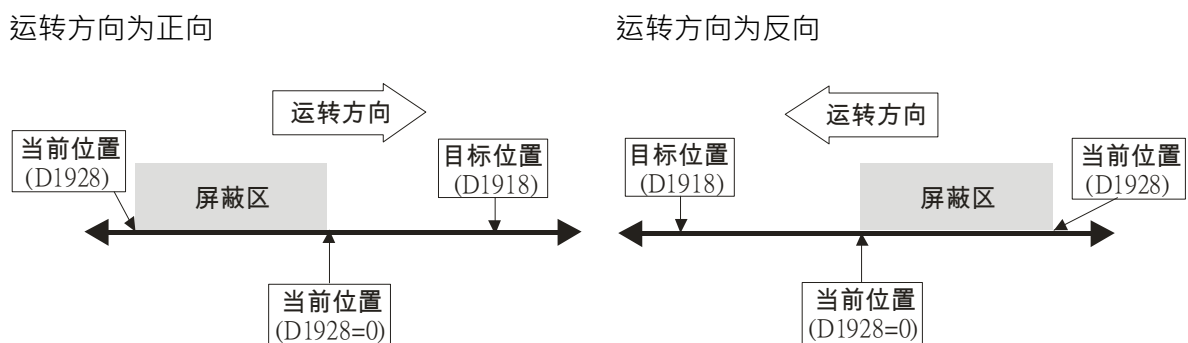


位置(1)：起始位置[1]，在原点及 DOG 信号的右边，DOG 信号=Off。

位置(2)\*：起始位置[2]在原点的右边，DOG 信号=On。

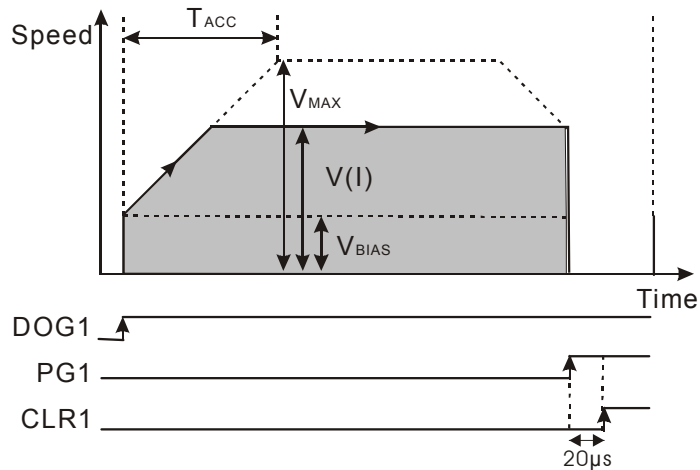
备注\*：位置(2)不支持 B, C 轴

8. D1926 之 b7：单段速外部触发模式启动
- b[7]=1 时启动单段速外部触发启动运动；当外部 DOG1 信号触发后开始执行单段速运动，运行中在非屏蔽区接受到 PG1 之信号触发后，则立即停止运行，且在 PG1 信号触发后经 20μs，输出 CLR1 信号。运动速度根据【V(I)】决定，由脉冲产生单元送出脉冲。
  - 屏蔽区范围如下图所示，由当前位置与目标位置的相对位置决定，若当前位置小于目标位置，运行方向为递增，在(D1928=0)时脱离屏蔽区的范围；若当前位置大于目标位置，则运行方向为递减，在 D1928=0 时脱离屏蔽区的范围。



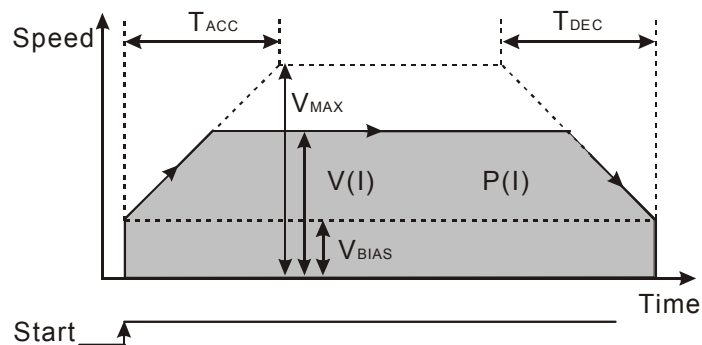
- 在相对坐标定位模式，运转方向由设定位置寄存器 P(I)内容值之符号位决定。
- 在绝对坐标定位模式，目标位置 P(I)大于当前位置时正转，小于时反转。
- 运转时，由启动速度  $V_{BIAS}$  开始加速至运转速度 V(I)后稳定运行，当 PG1 信号 ON 后脉冲立即停止输出。
- 应用的寄存器有  $V_{BIAS}$ ：D1824 (D1904, D1984)、V(I)：D1840(D1920, D2000)、 $V_{MAX}$ ：D1822 (D1902, D1982)、 $T_{ACC}$ ：D1836 (D1916, D1996)等寄存器。

### 3 各种装置功能



9. D1846 (D1926, D2006, D2086, D2166, D2246) 之 b8：单段速定位运动模式启动。

- b[8]=1 启动单段速定位运动，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。
- 在相对坐标定位模式，运转方向由设定位置寄存器 P(I)内容值之符号位决定。
- 在绝对坐标定位模式，目标位置 P(I)大于当前位置时正转，小于时反转。
- 运转中，由启动速度  $V_{BIAS}$  加速至运转速度  $V(I)$ 后稳定运行，接近目标位置 P(I)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, D2134, D2214)、 $V(I)$ ：D1840(D1920, D2000, D2080, D2160, D2240)、 $V_{MAX}$ ：D1822 (D1902, D1982, D2062, D2142, D2222)、 $P(I)$ ：D1838 (D1918, D1998, D2078, D2158, D2238)、 $T_{ACC}$ ：D1836 (D1916, D1996, D2076, D2156, D2236) 及  $T_{DEC}$ ：D1837 (D1917, D1997, D2077, D2157, D2237) 等寄存器。

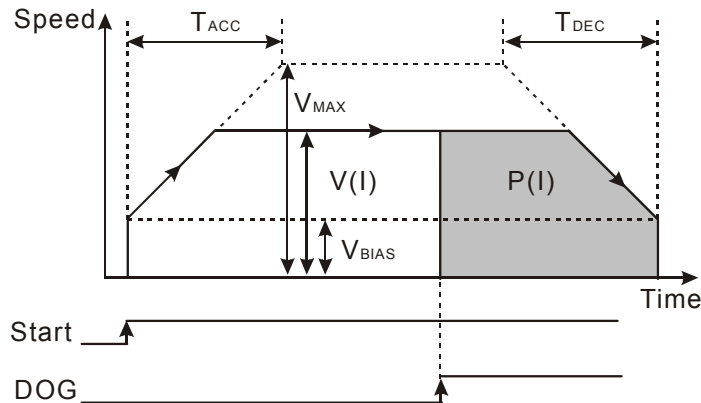


- 配合 X-Y-Z-A-B-C 轴参数设定(D1816、D1896、D1976、D2056、D2136、D2216)之 b[6] PWM 模式，当 b[8]=1 时，启动单段速定位模式，由 Y0~Y3 输出单向脉冲。

10. D1846 (D1926, D2006, D2086, D2166, D2246) 之 b9：插入单段速定位运动模式启动

- b[9]=1 执行插入单段速运动；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, D2158, D2238) 大于当前位置时正转，小于时反转。
- 运转速度会由启动速度  $V_{BIAS}$  开始加速至运转速度  $V(I)$  稳定运行，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。

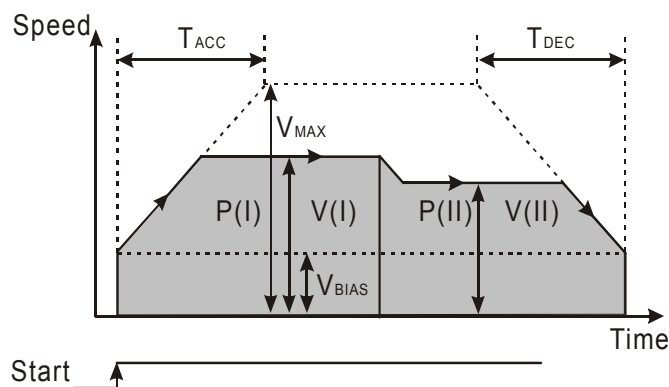
- 应用的寄存器有  $V_{BIAS}$  : D1824 (D1904, D1984, D2054, D2134, D2214) 、 $V(I)$  : D1840 (D1920, D2000, D2080, D2160, D2240) 、 $V_{MAX}$  : D1822 (D1902, D1982, D2062, D2142, D2222) 、 $P(I)$  : D1838 (D1918, D1998, D2078, D2158, D2238) 、 $T_{ACC}$  : D1836 (D1916, D1996, D2076, D2156, D2236) 及  $T_{DEC}$  : D1837 (D1917, D1997, D2077, D2157, D2237) 等寄存器。



#### 11. D1846 (D1926, D2006, D2086, D2166, D2246) 之 b10 : 连续两段速定位运动模式启动

- $b[10]=1$  启动连续两段速定位运动；以运转速度  $V(I)$  运行，到达第一段目标位置  $P(I)$  后，立即以运转速度  $V(II)$  运行至第二段目标位置  $P(II)$ 。
- 相对坐标定位模式下，运转方向由目标位置  $P(I)$  内容值之符号位决定，设定正数则正转，设定负数则反转；绝对坐标定位模式下，目标位置  $P(I)$  大于当前位置时正转，小于时反转。
- 运转时，由启动速度  $V_{BIAS}$  加速至运转速度  $V(I)$  稳定运行，直到输出目标位置  $P(I)$  的脉冲数后，此时会由运转速度  $V(I)$  加速/减速至运转速度  $V(II)$  稳定运行，当当前位置接近目标位置  $P(II)$  时减速至启动速度  $V_{BIAS}$ ，其间总共发送产生出  $P(I)+P(II)$  个脉冲数。

12. 应用的寄存器有  $V_{BIAS}$  : D1824 (D1904, D1984, D2054, D2134, D2214) 、 $V(I)$  : D1840 (D1920, D2000, D2080, D2160, D2240) 、 $V(II)$  : D1844 (D1924, D2004, D2084, D2164, D2244) 、 $V_{MAX}$  : D1822 (D1902, D1982, D2062, D2142, D2222) 、 $P(I)$  : D1838 (D1918, D1998, D2078, D2158, D2238) 、 $P(II)$  : D1842 (D1922, D2002, D2082, D2162, D2242) ;  $T_{ACC}$  : D1836 (D1916, D1996, D2076, D2156, D2236) 及  $T_{DEC}$  : D1837 (D1917, D1997, D2077, D2157, D2237) 等寄存器。

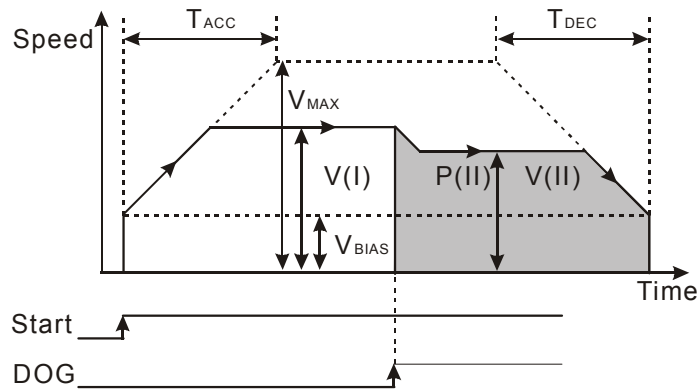


#### 13. D1846 (D1926, D2006, D2086, D2166, D2246) 之 b11 : 插入二段速定位运动模式启动

- $b[11]=1$  启动外部两段速定位运动；第 1 段位置定位期间以运转速度  $V(I)$  运行，当外部 DOG 信号触发，会立即以运转速度  $V(II)$  运行至第 2 段目标位置  $P(II)$ 。
- 相对坐标定位模式中，运转方向由目标位置  $P(I)$  内容值之符号位决定。

### 3 各种装置功能

- 绝对坐标定位模式中，目标位置 P(I) 大于当前位置时正转，小于时反转。
- 运转中，由启动速度  $V_{BIAS}$  开始加速至运转速度 V(I) 后稳定运行，当外部 DOG 信号触发时，立即加速/减速至运转速度 V(II) 后稳定运行；执行第二段速度中，当外部 STOP 信号触发频率波输出立即停止。
- 应用的寄存器有  $V_{BIAS}$  : D1824 (D1904, D1984, D2054, D2134, D2214) 、V(I) : D1840 (D1920, D2000, D2080, D2160, D2240) 、V(II) : D1844 (D1924, D2004, D2084, D2164, D2244) 、 $V_{MAX}$  : D1822 (D1902, D1982, D2062, D2142, D2222) 、P(I) : D1838 (D1918, D1998, D2078, D2158, D2238) 、P(II) : D1842 (D1922, D2002, D2082, D2162, D2242) ;  $T_{ACC}$  : D1836 (D1916, D1996, D2076, D2156, D2236) 及  $T_{DEC}$  : D1837 (D1917, D1997, D2077, D2157, D2237) 等寄存器。



#### 14. D1846 (D1926, D2006, D2086, D2166, D2246) 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	
A 轴		B 轴		C 轴		
HW	LW	HW	LW	HW	LW	
	D2087		D2167		D2247	

#### [说明]

- D1847 (D1927, D2007, D2087, D2167, D2247) 之 b2 : CLR 信号输出模式。
  - 当  $b[2]=0$  时 输出信号 CLR 为原点回归完成时 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。
- D1847 (D1927, D2007, D2087, D2167, D2247) 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。
- D1847 (D1927, D2007, D2087, D2167, D2247) 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。
- D1847 (D1927, D2007, D2087, D2167, D2247) 之 b5 : STOP 模式。

- b[5]=0：电机运转中，碰到 STOP 信号输入时，电机减速停，再度下达运动命令时，电机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：电机运转中，碰到 STOP 信号输入时，电机减速停，再度下达运动命令时，电机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, D2087, D2167, D2247) 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, D2087, D2167, D2247) 之 b7：LSP/LSN 停止模式。
    - b[7]=0：电机运转中，碰到 LSP/LSN 信号输入时，电机减速停止。
    - b[7]=1：电机运转中，碰到 LSP/LSN 信号输入时，电机立即停止。
  7. D1847 (D1927, D2007, D2087, D2167, D2247) 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  $\Phi A_{\pm}$  之上升沿触发。
    - b[10~8]=K2 (010)：MASK 以输入端  $\Phi A_{\pm}$  之下降沿触发。
    - b[10~8]=K3 (011)：MASK 以输入端  $\Phi B_{\pm}$  之上升沿触发。
    - b[10~8]=K4 (100)：MASK 以输入端  $\Phi B_{\pm}$  之下降沿触发。
  8. D1847 (D1927, D2007, D2087, D2167, D2247) 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		当前位置 CP(PLS)
HW	LW	HW	LW	HW	LW	
D1849	D1848	D1929	D1928	D2009	D2008	
A 轴		B 轴		C 轴		
HW	LW	HW	LW	HW	LW	
D2089	D2088	D2169	D2168	D2249	D2248	

[说明]

1. 显示范围：-2,147,483,648~+2,147,483,647。
2. 当前位置，以脉冲值 (PLS) 表示，由 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b0, b1 作设定。原点回归完成时，D1834 (D1914, D1994, D2074, D2154, D2234) 原点位置定义 HP，会填入当前位置 CP (PLS) 数据中。

### 3 各种装置功能

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1851	D1850	D1931	D1930	D2011	D2010
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2091	D2090	D2171	D2170	D2251	D2250

当前速度 CS(PPS)

[说明]

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

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1853	D1852	D1933	D1932	D2013	D2012
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2093	D2092	D2173	D2172	D2253	D2252

当前位置 CP(unit)

[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, D2136, D2216) 之 b0, b1 单位系设定变化。原点回归完成时，D1834 (D1914, D1994, D2074, D2154, D2234) 原点位置定义 HP 填入 D1852 (D1932, D2012, D2092, D2172, D2252) 当前位置数据中。

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
D1855	D1854	D1935	D1934	D2015	D2014
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
D2095	D2094	D2175	D2174	D2255	D2254

当前速度 CS(unit)

[说明]

1. 显示范围：0~+2,147,483,647。
2. 设定单位依照 D1816 (D1896, D1976, D2056, D2136, D2216) 之 b0, b1 单位系设定来变化。



X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
	D1856		D1936		D2016
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
	D2096		D2176		D2256

执行状态

[说明]

bit#	D1856 (D1936, D2016...)
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	未定义

X 轴		Y 轴		Z 轴	
HW	LW	HW	LW	HW	LW
	D1857		D1937		D2017
A 轴		B 轴		C 轴	
HW	LW	HW	LW	HW	LW
	D2097		D2177		D2257

错误编号

[说明]

详细内容请参考第 14 章错误代码原因对照表。

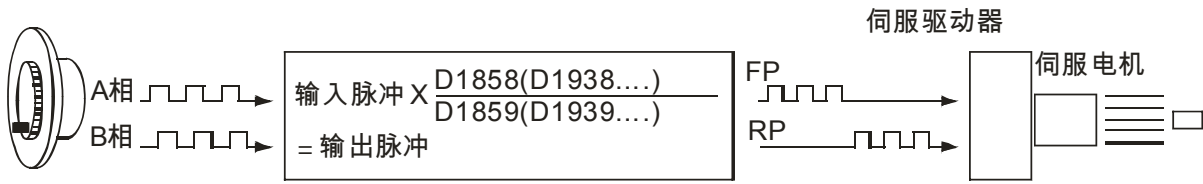
X 轴		Y 轴		Z 轴		电子齿轮比
HW	LW	HW	LW	HW	LW	
	D1858		D1938		D2018	电子齿轮 (分子)
	D1859		D1939		D2019	电子齿轮 (分母)
A 轴		B 轴		C 轴		电子齿轮比
HW	LW	HW	LW	HW	LW	
	D2098		D2178		D2258	电子齿轮 (分子)
	D2099		D2179		D2259	电子齿轮 (分母)

[说明]

1. 当运转命令 D1846 (D1926, D2006, D2086, D2166, D2246) 之 b5 设为 On，表示为手摇轮脉冲输入工作模式启动。

### 3 各种装置功能

2. 以手摇轮产生 A/B 相脉冲输入至  $\Phi A$  及  $\Phi B$ ，FP/ RP 输出与输入脉冲关系如下图所示：



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。

3. 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, D2178, D2258) · D1859 (D1939, D2019, D2099, D2179, D2259) ) 成比例的关系。

X 轴		Y 轴		Z 轴		手摇轮输入频率
HW	LW	HW	LW	HW	LW	
D1861	D1860	D1941	D1940	D2021	D2020	
A 轴		B 轴		C 轴		
HW	LW	HW	LW	HW	LW	
D2101	D2100	D2181	D2180	D2261	D2260	

[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		累计手摇轮输入脉冲个数
HW	LW	HW	LW	HW	LW	
D1863	D1862	D1943	D1942	D2023	D2022	
A 轴		B 轴		C 轴		
HW	LW	HW	LW	HW	LW	
D2103	D2102	D2183	D2182	D2263	D2262	

[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, D2178, D2258) · D1859 (D1939, D2019, D2099, D2179, D2259) ) 设定影响。







X 轴		Y 轴		Z 轴		手摇轮输入响应速度
HW	LW	HW	LW	HW	LW	
	D1864		D1944		D2024	
A 轴		B 轴		C 轴		
HW	LW	HW	LW	HW	LW	
	D2104		D2184		D2264	

[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，表示命令脉冲输出响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, D2104, D2184, D2264) 之 b8, b9：手摇轮输入脉冲型式设定

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
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	4*A/B 相脉冲

# 3 各种装置功能

## 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：

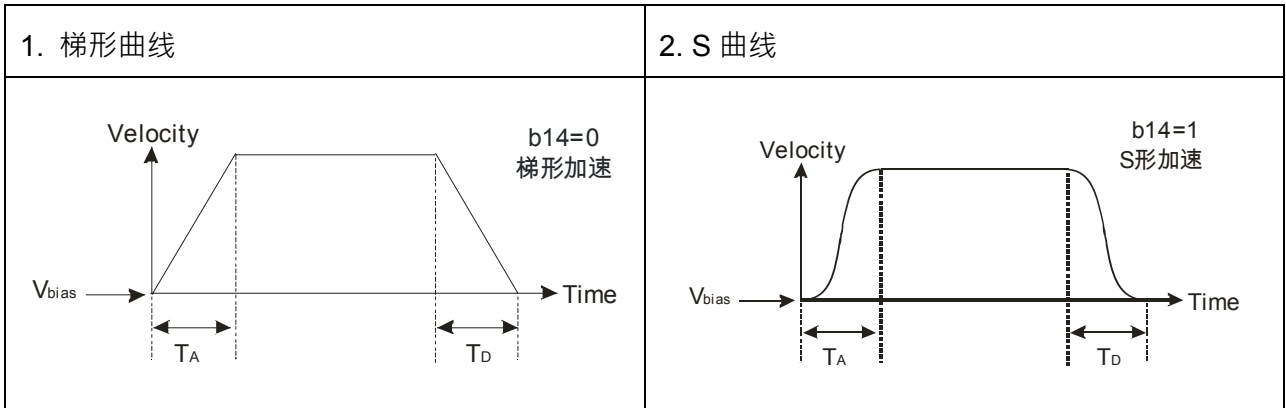
- |                 |                    |
|-----------------|--------------------|
| 1. 机械原点回归操作     | 6. 连续两段速定位运动模式     |
| 2. 寸动运动模式操作     | 7. 插入二段速定位运动模式     |
| 3. 单段速定位运动模式    | 8. 变速度运动模式         |
| 4. 插入单段速定位运动模式  | 9. 手摇轮 (MPG) 输入操作  |
| 5. 单段速外部触发运动模式* | 10. 周期/非周期式电子凸轮模式* |

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. 机械原点回归操作         | 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          |                 |

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：



### 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									
									寸动 JOG	原点回归	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	插入一段速度定位	变速度	手摇轮(MPG)输入	
X 轴	Y 轴		Z 轴		A 轴													
HW	LW	HW	LW	HW	LW	HW	LW											
D1819	D1818	D1899	D1898	D1979	D1978	D2059	D2058	电机转一圈所需脉冲数 A	若单位系选择为电机单位·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	电机转一圈移动距离 B										
-	D1816	-	D1896	-	D1976	-	D2056	参数设定	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	
D1823	D1822	D1903	D1902	D1983	D1982	D2063	D2062	最高速度 V <sub>MAX</sub>	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	
D1825	D1824	D1905	D1904	D1985	D1984	D2065	D2064	启动速度 V <sub>BIAS</sub>	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	
D1827	D1826	D1907	D1906	D1987	D1986	D2067	D2066	寸动 JOG 速度 V <sub>JOG</sub>	⊙	-	-	-	-	-	-	-	-	
D1829	D1828	D1909	D1908	D1989	D1988	D2069	D2068	原点回归速度 V <sub>RT</sub>	-	⊙	-	-	-	-	-	-	-	
D1831	D1830	D1911	D1910	D1991	D1990	D2071	D2070	原点回归减速速度 V <sub>CR</sub>										
-	D1832	-	D1912	-	D1992	-	D2072	原点回归之零点 (PG0) 信号数 N										
-	D1833	-	D1913	-	D1993	-	D2073	原点回归之脉冲信号数 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 <sub>ACC</sub>	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	-
-	D1837	-	D1917	-	D1997	-	D2077	减速时间 T <sub>DEC</sub>	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	-
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	当前位置 CP (PLS)	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	
D1851	D1850	D1931	D1930	D2011	D2010	D2091	D2090	当前速度 CS (PPS)	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	
D1853	D1852	D1933	D1932	D2013	D2012	D2093	D2092	当前位置 CP (unit)	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	
D1855	D1854	D1935	D1934	D2015	D2014	D2095	D2094	当前速度 CS (unit)	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	
-	D1858	-	D1938	-	D2018	-	D2098	电子齿轮分子	-	-	-	-	-	-	-	-	⊙	
-	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	-	-	-	-	-	-	-	-	⊙	

### 3 各种装置功能

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	原点回归	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	插入一段速度定位	变速度	手摇轮(MPG)输入
HW	LW	HW	LW	HW	LW	HW	LW										
D1865	-	-	-	-	-	-	-	停止模式 (OX0-99)	-	-	-	-	-	-	-	-	
D1867	D1866	D1947	D1946	D2027	D2026	-	-	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 码 输出 High byte	-	-	-	-	-	-	-	-	
D1874	-	-	-	-	-	-	-	M 码 off 输入 X 起点编号	-	-	-	-	-	-	-	-	
D1875	-	D1955	-	-	-	-	-	外部手动启动 (ZRN, MPG, JOG-, JOG+)	⊙	-	-	-	-	-	-	⊙	

运动模式对应使用寄存器				参数名称代码	操作模式								
B 轴		C 轴			寸动 JOG	原点回归	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	插入一段速度定位	变速度	手摇轮(MPG)输入
HW	LW	HW	LW										
D2139	D2138	D2219	D2218	电机转一圈所需脉冲数 A	若单位系选择为电机单位, 则不须设定 若为机械单位或复合单位系, 则必须作设定								
D2141	D2140	D2221	D2220	电机转一圈移动距离 B									
-	D2136	-	D2216	参数设定	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙
D2143	D2142	D2223	D2222	最高速度 $V_{MAX}$	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙
D2145	D2144	D2225	D2224	启动速度 $V_{BIAS}$	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙
D2147	D2146	D2227	D2226	寸动 JOG 速度 $V_{JOG}$	⊙	-	-	-	-	-	-	-	-
D2149	D2148	D2229	D2228	原点回归速度 $V_{RT}$	-	⊙	-	-	-	-	-	-	-
D2151	D2150	D2231	D2230	原点回归减速速度 $V_{CR}$									
-	D2152	-	D2232	原点回归之零点 (PG0) 信号数 N									
-	D2153	-	D2233	原点回归之脉冲信号数 P									
D2155	D2154	D2235	D2234	原点位置定义 HP									
-	D2156	-	D2236	加速时间 $T_{ACC}$	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	-

运动模式对应使用寄存器				参数名称代码	操作模式								
B 轴		C 轴			手动 JOG	原点回归	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	插入一段速度定位	变速度	手摇轮(MPG)输入
HW	LW	HW	LW										
-	D2157	-	D2237	减速时间 T <sub>DEC</sub>	◎	◎	◎	◎	◎	◎	◎	◎	-
D2159	D2158	D2239	D2238	目标位置(I) P(I)	-	-	◎	◎	◎	◎	◎	-	◎
D2161	D2160	D2242	D2240	运转速度(I) V(I)	-	-	◎	◎	◎	◎	◎	◎	-
D2163	D2162	D2243	D2242	目标位置(II) P(II)	-	-	-	-	-	◎	◎	-	◎
D2165	D2164	D2245	D2244	运转速度(II) V(II)	-	-	-	-	-	◎	◎	-	-
-	D2166	-	D2246	运转命令	◎	◎	◎	◎	◎	◎	◎	◎	◎
-	D2167	-	D2247	工作模式	◎	◎	◎	◎	◎	◎	◎	◎	◎
D2169	D2168	D2249	D2248	当前位置 CP (PLS)	◎	◎	◎	◎	◎	◎	◎	◎	◎
D2171	D2170	D2251	D2250	当前速度 CS (PPS)	◎	◎	◎	◎	◎	◎	◎	◎	◎
D2173	D2172	D2253	D2252	当前位置 CP (unit)	◎	◎	◎	◎	◎	◎	◎	◎	◎
D2175	D2174	D2255	D2254	当前速度 CS (unit)	◎	◎	◎	◎	◎	◎	◎	◎	◎
-	D2178	-	D2258	电子齿轮分子	-	-	-	-	-	-	-	-	◎
-	D2179	-	D2259	电子齿轮分母	-	-	-	-	-	-	-	-	◎
D2181	D2180	D2261	D2260	手摇轮输入频率	-	-	-	-	-	-	-	-	◎
D2183	D2182	D2263	D2262	累计手摇轮输入脉冲数	-	-	-	-	-	-	-	-	◎
-	D2184	-	D2264	MPG 响应速度	-	-	-	-	-	-	-	-	◎

◎：表示动作于该操作模式时，相关之控制寄存器。

### 3 各种装置功能

---

MEMO



## 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
LD	载入 A 接点	X、Y、M、S、T、C	0.14	3	4-3
LDI	载入 B 接点	X、Y、M、S、T、C	0.14	3	4-3
AND	串联 A 接点	X、Y、M、S、T、C	0.14	3	4-3
ANI	串联 B 接点	X、Y、M、S、T、C	0.14	3	4-4
OR	并联 A 接点	X、Y、M、S、T、C	0.14	3	4-4
ORI	并联 B 接点	X、Y、M、S、T、C	0.14	3	4-5
ANB	串联回路方块	无	-	3	4-5
ORB	并联回路方块	无	-	3	4-6

## 📖 输出指令

指令码	功 能	操 作 数	执行速度(us)	STEP	页 码
OUT	驱动线圈	Y、M、S	-	3	4-6
SET	动作保持(On)	Y、M、S	-	3	4-7
RST	接点或寄存器清除	Y、M、S、T、C、D、V、	-	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
96	TMR	16 位定时器	T-K 或 T-D	6	5	4-8
97	CNT	16 位计数器	C-K 或 C-D (16 位)	2.8	5	4-8
97	DCNT	32 位计数器	C-K 或 C-D (32 位)	2.8	6	4-9

## 📖 接点上升沿/下降沿检出指令

API	指令码	功 能	操作数	执行速度(us)	STEP	页 码
90	LDP	上升沿检出动作开始	X、Y、M、S、T、C	0.4	3	4-9
91	LDF	下降沿检出动作开始	X、Y、M、S、T、C	0.5	3	4-10
92	ANDP	上升沿检出串联连接	X、Y、M、S、T、C	0.4	3	4-10
93	ANDF	下降沿检出串联连接	X、Y、M、S、T、C	0.4	3	4-10

## 4 基本顺序指令

---

API	指令码	功 能	操作数	执行速度(us)	STEP	页 码
94	ORP	上升沿检出并联连接	X、Y、M、S、T、C	0.5	3	4-11
95	ORF	下降沿检出并联连接	X、Y、M、S、T、C	0.4	3	4-11

### 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
89	PLS	上微分输出	Y、M	0.2	3	4-12
99	PLF	下微分输出	Y、M	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
P	指针	P0~P255	-	1	4-13

## 4.2 基本指令说明

指令	功 能							适用机种
LD	载入 A 接点							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 指令用于左主线开始的 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。

### 程序范例

梯形图：



指令码：

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

说明：

指令	功 能							适用机种
LDI	载入 B 接点							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 指令用于左主线开始的 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。

### 程序范例

梯形图：



指令码：

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

说明：

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

# 4 基本顺序指令

## 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。

## 程序范例

梯形图：



指令码：

说明：

LDI	X1	载入 X1 之 B 接点
<b>AND</b>	<b>X0</b>	串联 X0 之 A 接点
OUT	Y1	驱动 Y1 线圈

指令	功能	适用機種
<b>ANI</b>	串联 B 接点	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。

## 程序范例

梯形图：



指令码：

说明：

LD	X1	载入 X1 之 A 接点
<b>ANI</b>	<b>X0</b>	串联 X0 之 B 接点
OUT	Y1	驱动 Y1 线圈

指令	功能	适用機種
<b>OR</b>	并联 A 接点	10PM
		✓

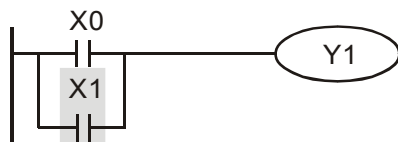
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999
	✓	✓	✓	✓	✓	✓	-

## 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。

## 程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
<b>OR</b>	<b>X1</b>	并联 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

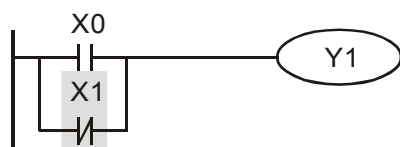
指令	功 能							适用机种
<b>ORI</b>	并联 B 接点							10PM
								✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	-	

### 指令说明

- ◆ **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。

### 程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
<b>ORI</b>	<b>X1</b>	并联 X1 之 B 接点
OUT	Y1	驱动 Y1 线圈

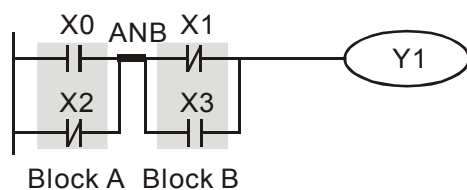
指令	功 能							适用机种
<b>ANB</b>	串联回路方块							10PM
								✓
操作数	无							

### 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。

### 程序范例

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
<b>ANB</b>		串联回路方块
OUT	Y1	驱动 Y1 线圈

## 4 基本顺序指令

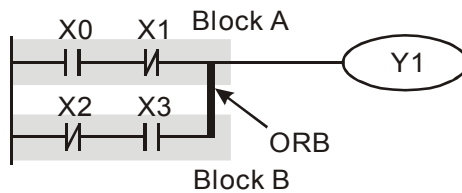
指令	功能	适用機種
ORB	并联回路方块	10PM
		✓
操作数	无	

### 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。

### 程序范例

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
<b>ORB</b>		<b>并联回路方块</b>
OUT	Y1	驱动 Y1 线圈

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

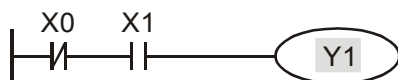
### 指令说明

- ◆ 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

### 程序范例

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
<b>OUT</b>	<b>Y1</b>	<b>驱动 Y1 线圈</b>

指令	功 能							适用机种
<b>SET</b>	动作保持 (On)							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。

### 程序范例

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
<b>SET</b>	<b>Y1</b>	<b>Y1 动作保持 (On)</b>

指令	功 能							适用机种
<b>RST</b>	接点或寄存器清除							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

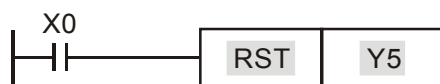
- ◆ 当 RST 指令被驱动，其指定的组件的动作如下：

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。

### 程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
<b>RST</b>	<b>Y5</b>	<b>Y5 接点清除</b>

## 4 基本顺序指令

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

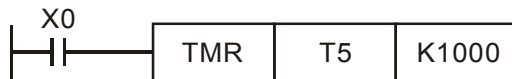
### 指令说明

- ◆ 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 Closed) 接点	开路

### 程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
<b>TMR</b>	<b>T5 K1000</b>	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		10PM
			✓
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 Closed) 接点	开路

- ◆ 当计数到达之后，若再有计数脉冲输入，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。

### 程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
<b>CNT</b>	<b>C20 K100</b>	C20 计数器设定 值为 K100



指令	功能		适用机种
DCNT	32 位计数器		10PM
			✓
操作数	C-K	C200、C204、C208~C255、K-2,147,483,648~K2,147,483,647	
	C-D	C200、C204、C208~C255、D0~D9,999	

### 指令说明

- ◆ DCNT 为 32 位计数器 C200 至 C255 之启动指令。
- ◆ 一般用加减法计数器 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。

### 程序范例

梯形图：



指令码：

说明：

LD	M0	载入 M0 之 A 接点
<b>DCNT</b>	<b>C254 K1000</b>	C254 计数器 设定值为 K1000

指令	功能							适用机种
LDP	上升沿检出动作开始							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沿检出状态存入累积寄存器内。

### 程序范例

梯形图：



指令码：

说明：

<b>LDP</b>	<b>X0</b>	X0 上升沿检出动作开始
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，指定上升沿接点的状态为 On，则电源开启后该上升沿接点为 TRUE。

## 4 基本顺序指令

指令	功 能							适用機種
LDF	下降沿检出动作开始							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沿检出状态存入累积寄存器内。

### 程序范例

梯形图：



指令码：

说明：

LDF	X0	X0 下降沿检出动作开始
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	上升沿检出串联连接							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 指令用于接点上升沿检出的串联连接。

### 程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
ANDP	X1	X1 上升沿检出串联连接
OUT	Y1	驱动 Y1 线圈

指令	功 能							适用機種
ANDF	下降沿检出串联连接							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 指令用于接点下降沿检出的串联连接。

程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
<b>ANDF</b>	<b>X1</b>	X1 下降沿检出串联连接
OUT	Y1	驱动 Y1 线圈

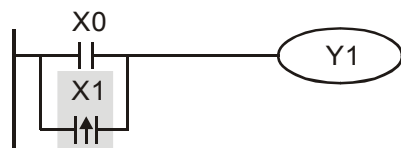
指令	功能							适用机种
<b>ORP</b>	上升沿检出并联连接							10PM ✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	-	

指令说明

◆ ORP 指令用于接点上升沿检出的并联连接。

程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
<b>ORF</b>	<b>X1</b>	X1 上升沿检出并联连接
OUT	Y1	驱动 Y1 线圈

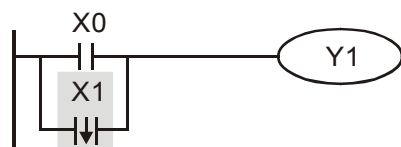
指令	功能							适用机种
<b>ORF</b>	下降沿检出并联连接							10PM ✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	✓	✓	✓	✓	✓	✓	-	

指令说明

◆ ORF 指令用于接点下降沿检出的并联连接。

程序范例

梯形图：



指令码：

说明：

LD	X0	载入 X0 之 A 接点
<b>ORF</b>	<b>X1</b>	X1 下降沿检出并联连接
OUT	Y1	驱动 Y1 线圈

## 4 基本顺序指令

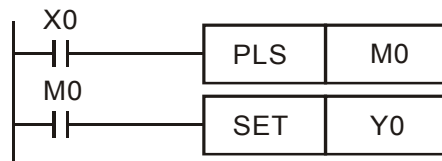
指令	功能							适用機種
PLS	上微分输出							10PM
								✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	-	✓	✓	-	-	-	-	

### 指令说明

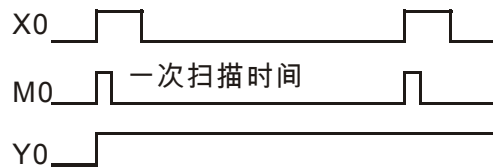
- ◆ 上微分输出指令。当 X0=Off→On (上升沿触发) 时 PLS 指令被执行，M0 送出一脉冲，脉冲长度为一次扫描时间。

### 程序范例

梯形图：



时序图：



指令码：

说明：

LD X0 载入 X0 之 A 接点

<b>PLS</b>	<b>M0</b>	M0 上微分输出
------------	-----------	----------

LD M0 载入 M0 之 A 接点

SET Y0 Y0 动作保持 (On)

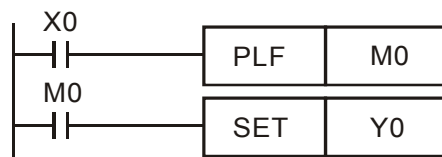
指令	功能							适用機種
PLF	下微分输出							10PM
								✓
操作数	X0~X377	Y0~Y377	M0~M4,095	S0~S1,023	T0~T255	C0~C255	D0~D9,999	
	-	✓	✓	-	-	-	-	

### 指令说明

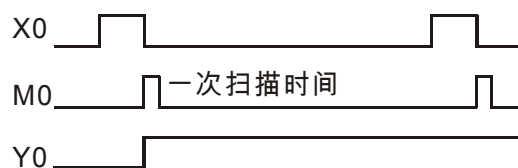
- ◆ 下微分输出指令。当 X0= On→Off (下降沿触发) 时 PLF 指令被执行，M0 送出一脉冲，脉冲长度为一次扫描时间。

### 程序范例

梯形图：



时序图：



指令码：

说明：

LD X0 载入 X0 之 A 接点

<b>PLF</b>	<b>M0</b>	M0 下微分输出
------------	-----------	----------

LD M0 载入 M0 之 A 接点

SET Y0 Y0 动作保持 (On)

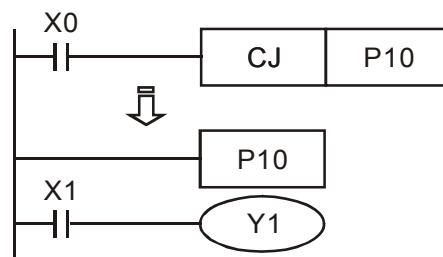
指令	功能	适用机种
P	指针	10PM
		✓
操作数	P0~P255	

### 指令说明

指针 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。

### 程序范例

梯形图：



指令码：

说明：

LD X0 载入 X0 之 A 接点  
CJ P10 跳转指令 CJ 到 P10

：

<b>P10</b>		指针 P10
LD	X1	载入 X1 之 A 接点
OUT	Y1	驱动 Y1 线圈

MEMO

## 5.1 应用指令一览表

分类	API	指令码		P 指令	功能	STEPS		页码
		16 位	32 位			16 位	32 位	
回路控制	00	CJ	-	✓	条件转移	3	-	5-16
	01	CALL	-	✓	调用子程序	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	-	-	循环回路起始 (只能一层)	3	-	5-22
	09	RPE	-	-	循环回路结束	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	✓	数据传送	5	6	5-26
	13	SMOV	-	✓	移位传送	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	✓	多点传送	7	13	5-33
	17	XCH	DXCH	✓	数据交换	5	9	5-34
	18	BCD	DBCD	✓	BIN→BCD 转换	5	5	5-35
19	BIN	DBIN	✓	BCD→BIN 转换	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	✓	逻辑与 (AND) 运算	7	9	5-45
	27	WOR	DWOR	✓	逻辑或 (OR) 运算	7	9	5-46
	28	WXOR	DWXOR	✓	逻辑异或 (XOR) 运算	7	9	5-47
29	NEG	DNEG	✓	求补码 (取 2 的补码)	3	3	5-48	
旋转移位	30	ROR	DROR	✓	右循环	5	9	5-50
	31	ROL	DROL	✓	左循环	5	9	5-51
	32	RCR	DRCR	✓	附进位标志右循环	5	9	5-52
	33	RCL	DRCL	✓	附进位标志左循环	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	-	✓	移位写入	7	-	5-60
39	SFRD	-	✓	移位读出	7	-	5-62	

# 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	
数据	40	ZRST	-	✓	区域清除	5	-	5-63
	41	DECO	-	✓	译码器	7	-	5-64
	42	ENCO	-	✓	编码器	7	-	5-66
	43	SUM	DSUM	✓	On 位数量	5	9	5-68
	44	BON	DBON	✓	On 位判定	7	13	5-69
	45	MEAN	DMEAN	✓	平均值	7	13	5-70
	46	ANS	-	-	警报点输出	7	-	5-71
	47	ANR	-	✓	警报点复位	1	-	5-72
	48	SQR	DSQR	✓	BIN 开平方根	5	9	5-74
	49	-	DFLT	✓	BIN 整数→二进制浮点数转换	-	6	5-75
高速处理	50	REF	-	✓	I/O 更新处理	5	-	5-77
便利指令	61	SER	DSER	✓	多点比较	9	17	5-79
	66	ALT	-	✓	On/Off 交替	3	-	5-81
	67	RAMP	DRAMP	-	斜坡信号	9	17	5-82
	69	SORT	DSORT	-	数据排序	11	21	5-84
I/O	78	FROM	DFROM	✓	I/O 模块 CR 数据读出	9	12	5-86
	79	TO	DTO	✓	I/O 模块 CR 数据写入	9	13	5-87
	87	ABS	DABS	✓	绝对值	3	5	5-90
基本指令	89	PLS	-	-	上微分输出	3	-	4-12
	90	LDP	-	-	上升沿检出动作开始	3	-	4-9
	91	LDF	-	-	下降沿检出动作开始	3	-	4-10
	92	ANDP	-	-	上升沿检出串联连接	3	-	4-10
	93	ANDF	-	-	下降沿检出串联连接	3	-	4-10
	94	ORP	-	-	上升沿检出并联连接	3	-	4-11
	95	ORF	-	-	下降沿检出并联连接	3	-	4-11
	96	TMR	-	-	定时器	5	-	4-8
	97	CNT	DCNT	-	计数器	5	6	4-8
通讯	100	MODRD	-	-	MODBUS 数据读取	7	-	5-91
	101	MODWR	-	-	MODBUS 数据写入	7	-	5-95
浮点运算	110	-	DECMP	✓	二进制浮点数比较	7	9	5-100
	111	-	DEZCP	✓	二进制浮点数区域比较	9	12	5-101
	112	-	DMOVP	✓	浮点数值数据传送	-	9	5-102
	116	-	DRAD	✓	角度→弧度	-	6	5-103
	117	-	DDEG	✓	弧度→角度	-	6	5-104
	120	-	DEADD	✓	二进制浮点数加法	7	9	5-105



# 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	
	121	-	DESUB	✓	二进制浮点数减法	7	9	5-106
	122	-	DEMUL	✓	二进制浮点数乘法	7	9	5-107
	123	-	DEDIV	✓	二进制浮点数除法	7	9	5-108
	124	-	DEXP	✓	二进制浮点数取指数	-	6	5-109
	125	-	DLN	✓	二进制浮点数取自然对数	-	6	5-110
	126	-	DLOG	✓	二进制浮点数取对数	-	9	5-112
	127	-	DESQR	✓	二进制浮点数开平方根	5	6	5-113
	128	-	DPOW	✓	浮点数权值指令	-	9	5-114
	129	-	DINT	✓	二进制浮点数→BIN 整数转换	-	6	5-116
	130	-	DSIN	✓	二进制浮点数→SIN 运算	5	6	5-117
	131	-	DCOS	✓	二进制浮点数→COS 运算	5	6	5-119
	132	-	DTAN	✓	二进制浮点数→TAN 运算	5	6	5-121
	133	-	DASIN	✓	二进制浮点数→ASIN 运算	-	6	5-123
	134	-	DACOS	✓	二进制浮点数→ACOS 运算	-	6	5-124
	135	-	DATAN	✓	二进制浮点数→ATAN 运算	-	6	5-125
	136	-	DSINH	✓	二进制浮点数→SINH 运算	-	6	5-126
	137	-	DCOSH	✓	二进制浮点数→COSH 运算	-	6	5-127
	138	-	DTANH	✓	二进制浮点数→TANH 运算	-	6	5-128
	172	-	DADDR	✓	浮点数值加法	-	13	5-129
	173	-	DSUBR	✓	浮点数值减法	-	13	5-130
	174	-	DMULR	✓	浮点数值乘法	-	13	5-131
	175	-	DDIVR	✓	浮点数值除法	-	13	5-132
接点类型逻辑运算	215	LD&	DLD&	-	S1 & S2	5	7	5-133
	216	LD	DLD	-	S1   S2	5	7	5-133
	217	LD^	DLD^	-	S1 ^ S2	5	7	5-133
	218	AND&	DAND&	-	S1 & S2	5	7	5-134
	219	AND	DAND	-	S1   S2	5	7	5-134
	220	AND^	DAND^	-	S1 ^ S2	5	7	5-134
	221	OR&	DOR&	-	S1 & S2	5	7	5-135
	222	OR	DOR	-	S1   S2	5	7	5-135
	223	OR^	DOR^	-	S1 ^ S2	5	7	5-135
接点类型比较指	224	LD=	DLD=	-	S1 = S2	5	7	5-136
	225	LD>	DLD>	-	S1 > S2	5	7	5-136
	226	LD<	DLD<	-	S1 < S2	5	7	5-136
	228	LD<>	DLD<>	-	S1 ≠ S2	5	7	5-136
	229	LD<=	DLD<=	-	S1 ≤ S2	5	7	5-136
	230	LD>=	DLD>=	-	S1 ≥ S2	5	7	5-136
	232	AND=	DAND=	-	S1 = S2	5	7	5-137

## 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	
	233	AND>	DAND>	-	$S1 > S2$	5	7	5-137
	234	AND<	DAND<	-	$S1 < S2$	5	7	5-137
	236	AND<>	DAND<>	-	$S1 \neq S2$	5	7	5-137
	237	AND<=	DAND<=	-	$S1 \leq S2$	5	7	5-137
	238	AND>=	DAND>=	-	$S1 \geq S2$	5	7	5-137
	240	OR=	DOR=	-	$S1 = S2$	5	7	5-138
	241	OR>	DOR>	-	$S1 > S2$	5	7	5-138
	242	OR<	DOR<	-	$S1 < S2$	5	7	5-138
	244	OR<>	DOR<>	-	$S1 \neq S2$	5	7	5-138
	245	OR<=	DOR<=	-	$S1 \leq S2$	5	7	5-138
	246	OR>=	DOR>=	-	$S1 \geq S2$	5	7	5-138
其它 指令	147	SWAP	DSWAP	✓	上/下字节交换	3	5	5-139
	154	RAND	DRAND	✓	随机数值产生	7	13	5-140
	202	SCAL	-	✓	比例值运算	9	-	5-141
	203	SCLP	DSCLP	✓	参数型比例值运算	7	13	5-143
	256	CJN	-	✓	反条件转移	3	-	5-147
	257	JMP	-	-	无条件转移	3	-	5-148
	258	BRET	-	-	回总线	1	-	5-149
	259	MMOV	-	✓	16→32 位数值转换	6	-	5-150
260	RMOV	-	✓	32→16 位数值转换	6	-	5-151	

## 5.2 应用指令的组成

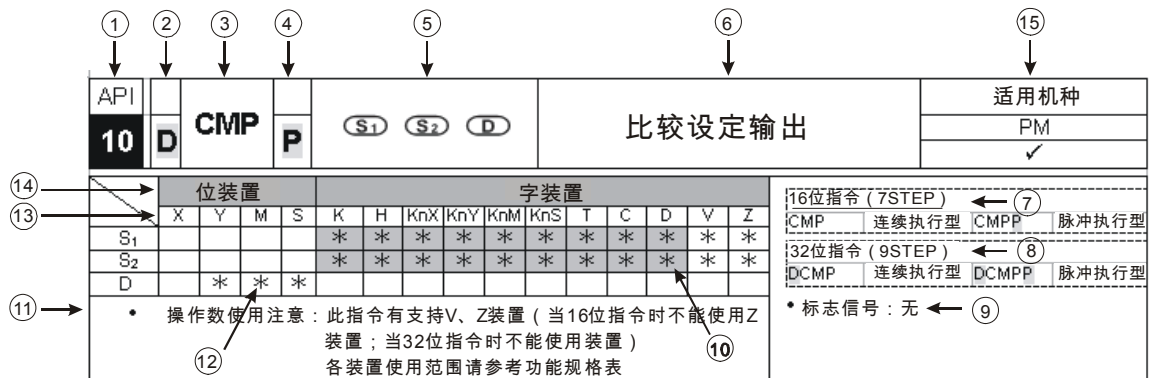
◆ 应用指令的结构可分为两部份：指令名及操作数

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



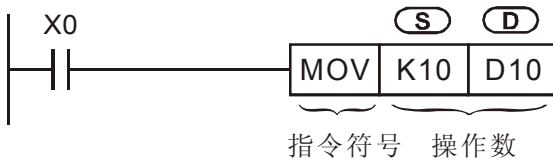
- ① 应用指令 API 编号号码
- ② 上方框表示具有 16 位指令。若为虚线表示此应用指令无 16 位指令  
下方框表示具有 32 位指令，若为虚线表示此应用指令无 32 位指令。若有 32 位指令方框内以 **D** 表示 (例：API 10 **D**CMP)
- ③ 应用指令名
- ④ 上方框表示有些指令在应用上通常是使用脉冲指令，方框内以 ☺ 表示  
下方框表示具有脉冲执行型指令，方框内以 **P** 表示 (例：API 12 **P**MOV<sub>P</sub>)
- ⑤ 应用指令的操作数格式
- ⑥ 应用指令功能描述
- ⑦ 16 位指令所占的地址数，连续执行型指令名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标志信号
- ⑩ 符号 ‘\*’ 标示者又含灰底色者，表示该装置可使用变址寄存器 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 装置类型
- ⑮ 适用该应用指令的系列机种

# 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，若以手持程序书写器 (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：

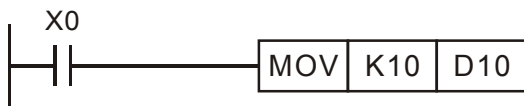
<b>S</b>	来源操作数；若来源操作数有一个以上，那么则以 $S_1$ 、 $S_2$ ... 分别表示。
<b>D</b>	目的操作数；若目的操作数有一个以上，那么则以 $D_1$ 、 $D_2$ ... 分别表示。
若操作数只可指定常数 K/H 或寄存器时，那么则以 $m$ 、 $m_1$ 、 $m_2$ 、 $n$ 、 $n_1$ 、 $n_2$ 表示。	

## ◆ 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数据则分为 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)

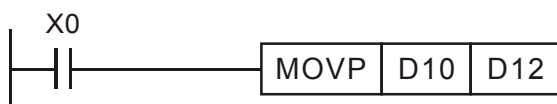


## ◆ 连续执行型脉冲执行型

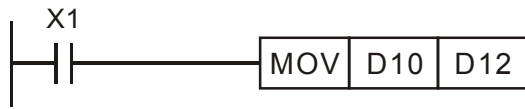
以指令的执行方式来说亦可分成「连续执行型」及「脉冲执行型」2 种。由于指令不被执行时，所需的执行时间比较短，因此程序中尽可能的使用脉冲执行型指令，可减少扫描周期。在指令后面加上 "P" 标号的指令，即为脉冲执行型指令。有些指令大部份的应用上都是使用脉冲执行型方式，如 INC、DEC 及移位相关等指令，因此于各指令的标号右上方均加上「P」标志代表该指令通常是使用脉冲执行型。

### 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

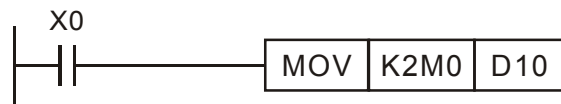
上图的两个条件接点 X0、X1=Off 时，指令不被执行，目的地操作数 D 的内容没有变化

◆ 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 的类型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、变址寄存器 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
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

# 5 应用指令分类及基本使用

◆ 一般的标志信号

(例) M1968：零标志            M1969：借位标志            M1970：进位标志

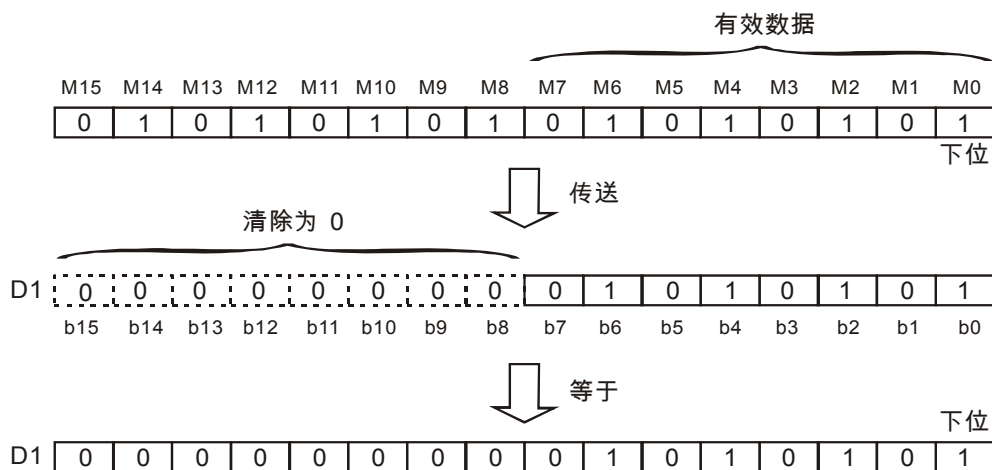
对应着应用指令运算结果，DVP-PM 系列有下列的标志信号 (Flag)。

无论那一个标志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 等标志信号的状态。但是当指令不被执行时，标志信号的 On/Off 状态被保持住。请注意上述标志信号的动作，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

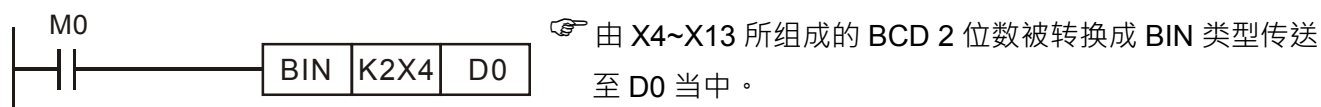
◆ X、Y、M、S 等只有 On/Off 变化的装置称之为位装置 (Bit Device)，而 T、C、D、V、Z 等专门用来存放数值的装置称之为字符装置 (Word Device)。虽然说位装置只能作 On/Off 变化，但是加上特定的宣告位装置也可以数值的类型被使用于应用指令的操作数当中，所谓的宣告是在位装置的前面加上位数，它是以 Kn 来表示。

◆ 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。

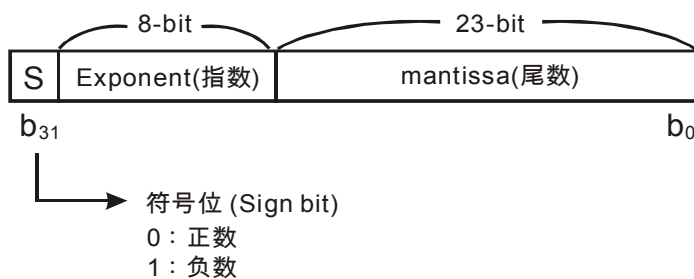
DVP-PM 系列机种的内部数值运算一般是以 BIN 整数值为准。整数执行除算时，例： $40 \div 3 = 13$ ，余数为 1。整数执行开平方动作时，小数点会被舍弃掉。但若使用小数点运算指令则可求出小数点。

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：



# 5 应用指令分类及基本使用

可表达的大小为：

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

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：23.0=10111

步骤二：将二进制正规化：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

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

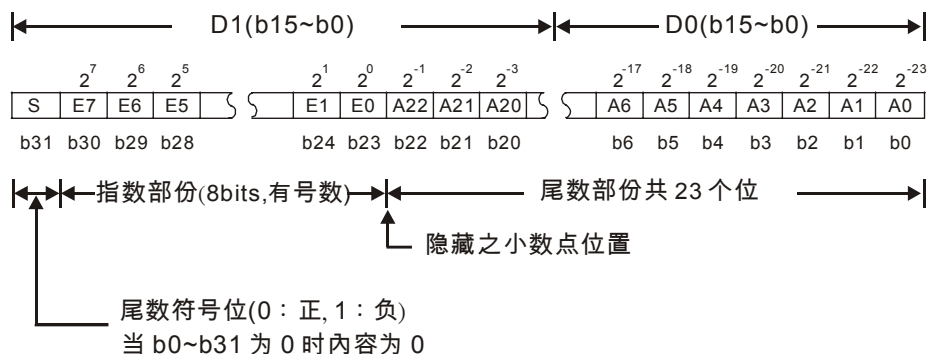
步骤四：组合符号位、指数、尾数成为浮点数。

$$0 \ 10000011 \ 0111 \ 10000000000000000000_2 = 41B80000_{16}$$

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示，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。

$$\text{十进制浮点数} = [\text{常数 } D0] \times 10^{[\text{指数 } D1]}$$

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

$$\text{指数 } D1 = -41 \sim +35$$

此外，底数 100 不存在于 D0 的内容，因为，100 是以  $1,000 \times 10^{-1}$  来表示。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 的转换指令：(DEBCD)

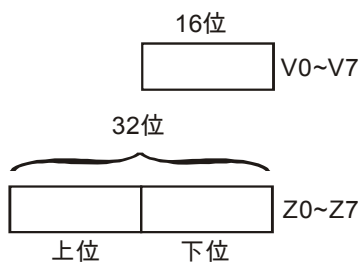
十进浮点数 → 二进制浮点数 的转换指令：(DEBIN)

◆ 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 等标志信号的状态。使用浮点运算指令，执行结果也会影响零标志信号 (M1968)、借位标志信号 (M1970) 及进位标志信号 (M1969)。说明如下：

- 零标志信号：运算结果为 0 时，M1968=On。
- 借位标志信号：运算结果超出最小处理单位时，M1970=On。
- 进位标志信号：运算结果绝对值超出使用范围时，M1969=On。

## 5.4 使用变址寄存器 V、Z 来修饰操作数

变址寄存器 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	5-37
	46	ANS	-	-	警报点输出	7	-	5-71
	47	ANR	-	✓	警报点复位	1	-	5-72
	66	ALT	-	✓	On/Off 交替	3	-	5-81
	87	ABS	DABS	✓	绝对值	3	5	5-90
	92	ANDP	-	-	上升沿检出串联连接	3	-	4-10
	93	ANDF	-	-	下降沿检出串联连接	3	-	4-10
	133	-	DASIN	✓	二进制浮点数 ASIN 运算	-	6	5-123
	134	-	DACOS	✓	二进制浮点数 ACOS 运算	-	6	5-124
	135	-	DATAN	✓	二进制浮点数 ATAN 运算	-	6	5-125
	218	AND&	DAND&	-	S1 & S2	5	7	5-134
	219	AND	DAND	-	S1   S2	5	7	5-134
	220	AND^	DAND^	-	S1 ^ S2	5	7	5-134
	232	AND=	DAND=	-	S1 = S2	5	7	5-137
	233	AND>	DAND>	-	S1 > S2	5	7	5-137
	234	AND<	DAND<	-	S1 < S2	5	7	5-137
	236	AND<>	DAND<>	-	S1 ≠ S2	5	7	5-137
	237	AND<=	DAND<=	-	S1 ≤ S2	5	7	5-137
238	AND>=	DAND>=	-	S1 ≥ S2	5	7	5-137	
B	15	BMOV	-	✓	全部传送	7	-	5-31
	18	BCD	DBCD	✓	BIN→BCD 转换	5	5	5-35
	19	BIN	DBIN	✓	BCD→BIN 转换	5	5	5-36
	44	BON	DBON	✓	On 位判定	7	13	5-69
	258	BRET	-	-	回总线	1	-	5-149
C	00	CJ	-	✓	条件转移	3	-	5-16
	01	CALL	-	✓	调用子程序	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	4-9
	131	-	DCOS	✓	二进制浮点数 COS 运算	5	6	5-119
	137	-	DCOSH	✓	二进制浮点数 COSH 运算	-	6	5-127
	256	CJN	-	✓	反条件转移	3	-	5-147
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	-	5-64
	117	-	DDEG	✓	弧度→角度	-	6	5-104

# 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	
E	42	ENCO	-	✓	编码器	7	-	5-66
	110	-	DECMP	✓	二进制浮点数比较	7	9	5-100
	111	-	DEZCP	✓	二进制浮点数区域比较	9	12	5-101
	112	-	DMOV	✓	浮点数值数据传送	-	9	5-102
	120	-	DEADD	✓	二进制浮点数加法	7	9	5-105
	121	-	DESUB	✓	二进制浮点数减法	7	9	5-106
	122	-	DEMUL	✓	二进制浮点数乘法	7	9	5-107
	123	-	DEDIV	✓	二进制浮点数除法	7	9	5-108
	124	-	DEXP	✓	二进制浮点数取指数	-	6	5-109
	127	-	DESQR	✓	二进制浮点数开平方根	5	6	5-113
	172	-	DADDR	✓	浮点数值加法	-	13	5-129
	173	-	DSUBR	✓	浮点数值减法	-	13	5-130
	174	-	DMULR	✓	浮点数值乘法	-	13	5-131
	175	-	DDIVR	✓	浮点数值除法	-	13	5-132
	F	16	FMOV	DFMOV	✓	多点传送	7	13
49		-	DFLT	✓	BIN 整数→二进制浮点数转换	-	6	5-75
78		FROM	DFROM	✓	扩展模块 CR 数据读出	9	12	5-86
I	24	INC	DINC	✓	BIN 加一	3	3	5-43
	129	-	DINT	✓	二进制浮点数→BIN 整数转换	-	6	5-116
J	257	JMP	-	-	无条件转移	3	-	5-148
L	90	LDP	-	-	上升沿检出动作开始	3	-	4-9
	91	LDF	-	-	下降沿检出动作开始	3	-	4-10
	125	-	DLN	✓	二进制浮点数取自然对数	-	6	5-110
	126	-	DLOG	✓	二进制浮点数取对数	-	9	5-112
	215	LD&	DLD&	-	S1 & S2	5	7	5-133
	216	LD	DLD	-	S1   S2	5	7	5-133
	217	LD^	DLD^	-	S1 ^ S2	5	7	5-133
	224	LD=	DLD=	-	S1 = S2	5	7	5-136
	225	LD>	DLD>	-	S1 > S2	5	7	5-136
	226	LD<	DLD<	-	S1 < S2	5	7	5-136
	228	LD<>	DLD<>	-	S1 ≠ S2	5	7	5-136
	229	LD≤	DLD≤	-	S1 ≤ S2	5	7	5-136
	230	LD≥	DLD≥	-	S1 ≥ S2	5	7	5-136
M	12	MOV	DMOV	✓	数据传送	5	6	5-26
	22	MUL	DMUL	✓	BIN 乘法	7	9	5-41
	45	MEAN	DMEAN	✓	平均值	7	13	5-70
	100	MODRD	-	-	MODBUS 数据读取	7	-	5-91
	101	MODWR	-	-	MODBUS 数据写入	7	-	5-95
	259	MMOV	-	✓	16→32 位数值转换	6	-	5-150

# 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	
N	29	NEG	DNEG	✓	求补码 (取 2 的补码)	3	3	5-48
O	94	ORP	-	-	上升沿检出并联连接	3	-	4-11
	95	ORF	-	-	下降沿检出并联连接	3	-	4-11
	221	OR&	DOR&	-	S1 & S2	5	7	5-135
	222	OR	DOR	-	S1   S2	5	7	5-135
	223	OR^	DOR^	-	S1 ^ S2	5	7	5-135
	240	OR=	DOR=	-	S1 = S2	5	7	5-138
	241	OR>	DOR>	-	S1 > S2	5	7	5-138
	242	OR<	DOR<	-	S1 < S2	5	7	5-138
	244	OR<>	DOR<>	-	S1 ≠ S2	5	7	5-138
	245	OR<=	DOR<=	-	S1 ≤ S2	5	7	5-138
	246	OR>=	DOR>=	-	S1 ≥ S2	5	7	5-138
P	89	PLS	-	-	上微分输出	3	-	4-12
	99	PLF	-	-	下微分输出	3	-	4-12
	128	-	DPOW	✓	浮点数权值指令	-	9	5-114
R	08	RPT	-	-	循环回路起始 (只能一层)	3	-	5-22
	09	RPE	-	-	循环回路结束	1	-	5-22
	30	ROR	DROR	✓	右循环	5	9	5-50
	31	ROL	DROL	✓	左循环	5	9	5-51
	32	RCR	DRCR	✓	附进位标志右循环	5	9	5-52
	33	RCL	DRCL	✓	附进位标志左循环	5	9	5-53
	50	REF	-	✓	I/O 更新处理	5	-	5-77
	67	RAMP	DRAMP	-	斜坡信号	9	17	5-82
	116	-	DRAD	✓	角度→弧度	-	6	5-103
	154	RAND	DRAND	✓	随机数值产生	7	13	5-140
260	RMOV	-	✓	32→16 位数值转换	6	-	5-151	
S	02	SRET	-	-	子程序结束	1	-	5-19
	13	SMOV	-	✓	位数传送	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	-	✓	移位写入	7	-	5-60
	39	SFRD	-	✓	移位读出	7	-	5-62
	48	SQR	DSQR	✓	BIN 开平方根	5	9	5-74
	43	SUM	DSUM	✓	On 位数量	5	9	5-68
	61	SER	DSER	✓	多点比较	9	17	5-79
	69	SORT	DSORT	-	数据排序	11	21	5-84
	147	SWAP	DSWAP	✓	上/下字节交换	3	5	5-139
	130	-	DSIN	✓	二进制浮点数 SIN 运算	5	6	5-117
136	-	DSINH	✓	二进制浮点数 SINH 运算	-	6	5-126	

## 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	
S	202	SCAL	-	✓	比例值运算	9	-	5-141
	203	SCLP	DSCLP	✓	参数型比例值运算	7	13	5-143
T	79	TO	DTO	✓	扩展模块 CR 数据写入	9	13	5-87
	96	TMR	-	-	定时器	5	-	4-8
	132	-	DTAN	✓	二进制浮点数 TAN 运算	5	6	5-121
	138	-	DTANH	✓	二进制浮点数 TANH 运算	-	6	5-128
W	07	WDT	-	✓	逾时监视定时器	1	-	5-21
	26	WAND	DWAND	✓	逻辑与 (AND) 运算	7	9	5-45
	27	WOR	DWOR	✓	逻辑或 (OR) 运算	7	9	5-46
	28	WXOR	DWXOR	✓	逻辑异或 (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	-	5-63

# 5 应用指令分类及基本使用

## 5.6 应用指令

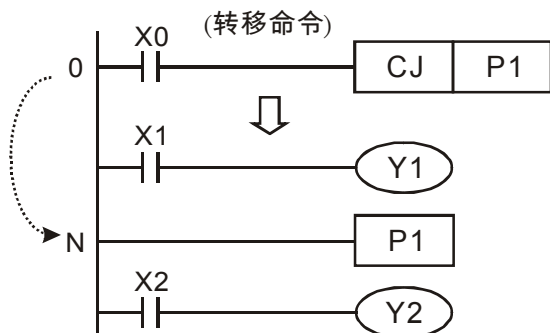
API																			适用机种	
<b>00</b>		<b>CJ</b>		<b>P</b>		<b>S</b>													10PM	
																			✓	
		位装置				字符装置											16 位指令 (3 STEP)			
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	CJ 连续执行型 CJP 脉冲执行型			
		<ul style="list-style-type: none"> <li>操作数使用注意：S 操作数可指定 P</li> <li>S 操作数可指定 P0~P255</li>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

### 指令说明

- ◆ **S**：条件转移之目的指标。
- ◆ 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。
- ◆ 转移执行中各种装置动作情形说明：
  1. Y、M、S 保持转移发生前之状态。
  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转移前被驱动，则在转移执行中该装置仍处于清除状态。

### 程序范例 (一)

- ◆ 当 X0=On 时，程序自动从地址 0 转移至地址 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。



程序范例  
(二)

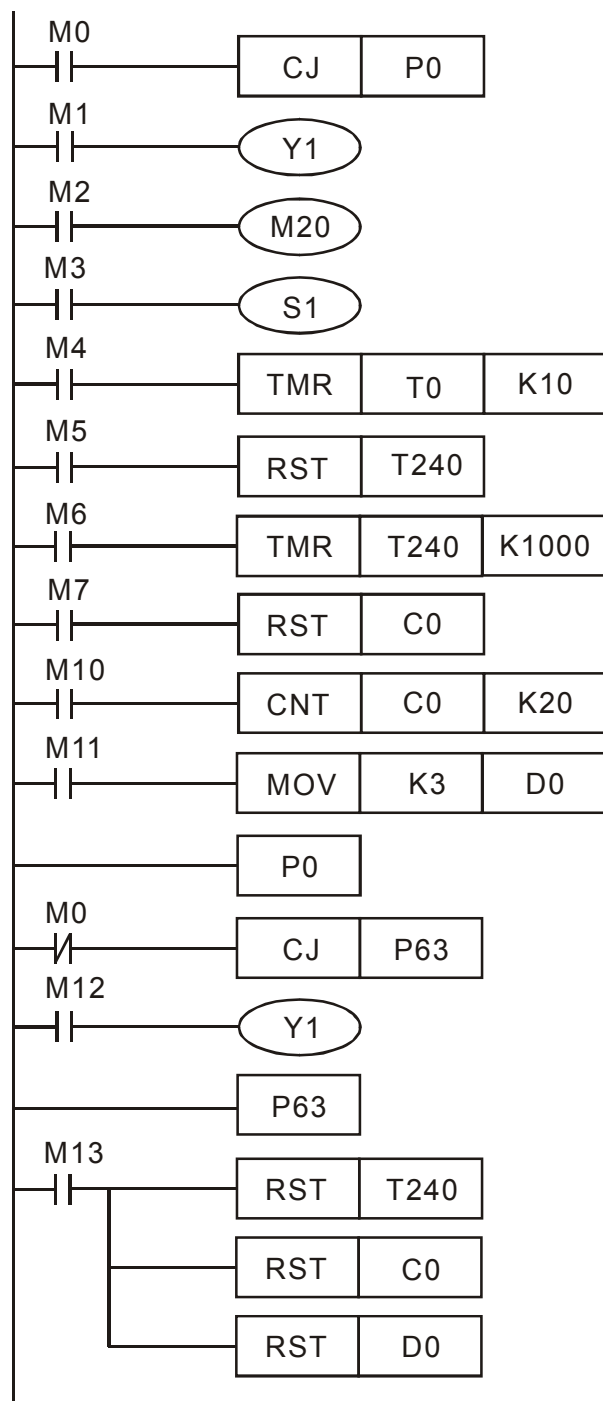
◆ 底下表格为下列程序中，各个装置状态变化。

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 <sup>注一</sup> 、M20、S1 Off
	M1、M2、M3 On	M1、M2、M3 由 On→Off	Y1 <sup>注一</sup> 、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
<b>01</b>		<b>CALL</b>		<b>P</b>														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	CALL 連續執行型    CALLP 脈衝執行型	
		<ul style="list-style-type: none"> <li>操作數使用注意：S 操作數可指定 P</li> <li>S 操作數可指定 P0~P255</li> <li>P 裝置不支持 V、Z 修飾</li> </ul>															32 位指令	
																	<ul style="list-style-type: none"> <li>標志信號：無</li> </ul>	

### 指令說明

- ◆ **S**：調用子程序之指針。
- ◆ 指針所指定的子程序請於 M102、M2 和 SRET 指令後編寫。
- ◆ 指針 **P** 之號碼在被 **CALL** 使用時，不可與 **CJ**、**CJN**、**JMP** 指令指定相同之號碼。
- ◆ 若僅使用 **CALL** 指令則可不限次數調用同一指針號碼之子程序。

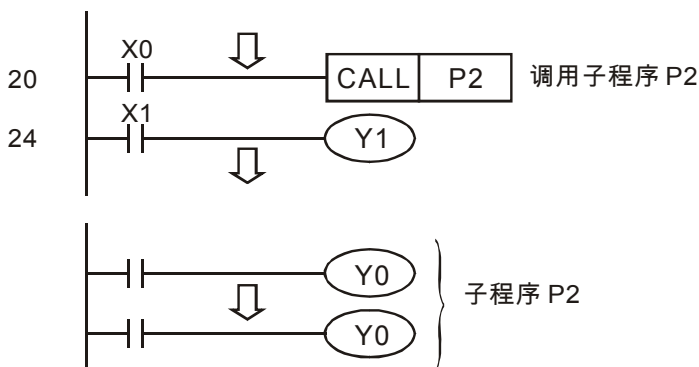
API																		適用機種
<b>02</b>		<b>SRET</b>																10PM
																		✓
		位裝置				字裝置											16 位指令 (1 STEP)	
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SRET 連續執行型    -    -	
		<ul style="list-style-type: none"> <li>操作數使用注意：無操作數</li> <li>不須接點驅動的指令</li> </ul>															32 位指令	
																	<ul style="list-style-type: none"> <li>標志信號：無</li> </ul>	

### 指令說明

- ◆ 表示子程序結束。子程序執行結束由 **SRET** 返回 O100 主程序，執行原調用該子程序 **CALL** 指令的下一個指令。

### 程序範例 (一)

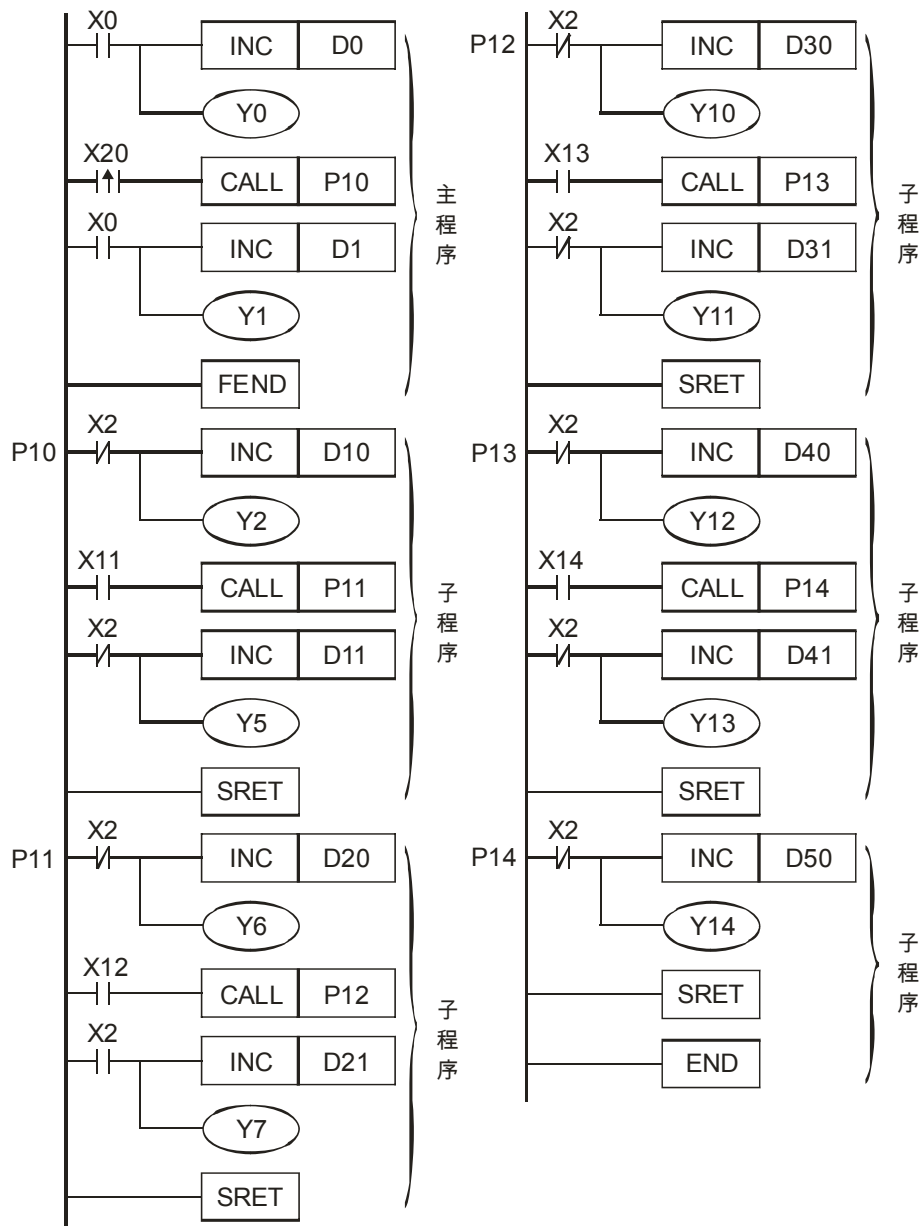
- ◆ 當 X0 為 On 時，則執行 **CALL** 指令，轉移到 P2 執行所指定的子程序，當執行 **SRET** 指令時，則回到地址 24，繼續往下執行。



# 5 应用指令分类及基本使用

程序范例  
(二)

- ◆ 当 X20 为由 Off 到 On 之上升沿触发执行 CALL P10 指令，转移到 P10 执行所指定的子程序。
- ◆ 当 X11 为 On 时，则执行 CALL P11，转移到 P11 执行所指定的子程序。
- ◆ 当 X12 为 On 时，则执行 CALL P12，转移到 P12 执行所指定的子程序。
- ◆ 当 X13 为 On 时，则执行 CALL P13，转移到 P13 执行所指定的子程序。
- ◆ 当 X14 为 On 时，则执行 CALL P14，转移到 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。

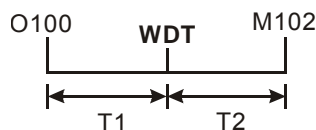


API																		适用機種		
<b>07</b>		<b>WDT</b>																10PM		
																		✓		
		位裝置				字裝置											16 位指令 (1 STEP)			
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WDT	連續執行型	WDTP	脈衝執行型
<ul style="list-style-type: none"> <li>操作數使用注意：无操作數 不須接點驅動的指令</li> </ul>																	32 位指令			
				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

### 指令說明

- ◆ WDT 指令可用來清除 PM 中的監控定時器的計時時間。當 PM 的掃描 (由地址 0 至 END 或 FEND 指令執行時間) 超過 200ms 時, PM ERROR 的指示燈會亮, 使用者必須將 PM 電源 Off 再 On, PM 會依據 MANU/AUTO 開關來判斷 RUN/STOP 狀態, 若無 MANU/AUTO 開關, 則 PM 會自動回到 STOP 狀態。
- ◆ 令逾時監視定時器動作的時機：
  1. PM 系統發生異常。
  2. 程序執行時間太長, 造成掃描週期大於 D1000 的內容值。可以下列 2 種方法來改善：

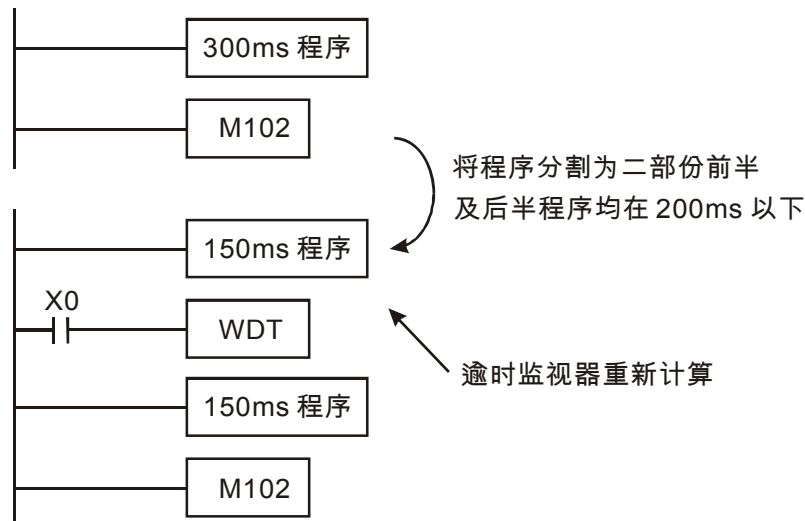
使用 WDT 指令



可由 D1000(出廠設定值為 200ms)的設定值改變逾時監視時間。

### 程序範例

- ◆ 若程序掃描時間為 300ms，此時，可將程序分割為 2 部份，並在中間放入 WDT 指令，使得前半及後半程序都在 200ms 以下。



### 補充說明

- ◆ WDT 指令用於輸入條件成立時的程序掃描。可編寫程序強制 WDT 指令只在一個週期內執行。PM 用戶可以使用 WDT 指令的脈衝型指令 WDTP。
- ◆ PM 逾時監視定時器的預設定值是 200ms。此時間限制可由用戶自行定制, 透過編輯數據寄存器 D1000 即可。

# 5 应用指令分类及基本使用

API																		适用機種
<b>08</b>		<b>RPT</b>																10PM
																		✓

	位装置				字符装置													
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z			
S					*	*	*	*	*	*	*	*	*	*	*			
<ul style="list-style-type: none"> <li>操作数使用注意：不须接点驱动指令 此指令有支持 V 装置 各装置使用范围请参考功能规格表</li> </ul>																		
<div style="border: 1px dashed black; padding: 5px;"> <b>16 位指令 (3 STEP)</b>            RPT      连续执行型      -      -         </div> <div style="border: 1px dashed black; padding: 5px; margin-top: 5px;"> <b>32 位指令</b>            -      -      -      -         </div> <p style="text-align: center;">标志信号：无</p>																		

### 指令说明

- ◆ **S**：回路重复执行的次数。
- ◆ 循环式 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																		适用機種
<b>09</b>		<b>RPE</b>																10PM
																		✓

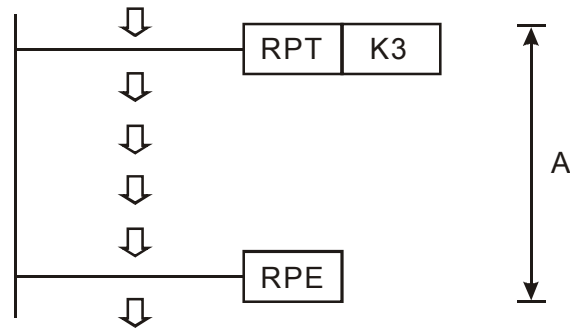
	位装置				字符装置													
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z			
<ul style="list-style-type: none"> <li>操作数使用注意：无操作数 不须接点驱动指令</li> </ul>																		
<div style="border: 1px dashed black; padding: 5px;"> <b>16 位指令 (1 STEP)</b>            RPE      连续执行型      -      -         </div> <div style="border: 1px dashed black; padding: 5px; margin-top: 5px;"> <b>32 位指令</b>            -      -      -      -         </div> <p style="text-align: center;">标志信号：无</p>																		

### 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：
  1. RPE 指令在 RPT 指令之前。
  2. 有 RPT 指令没有 RPE 指令。
  3. RPT~RPE 指令个数不同时。
- ◆ 循环式 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。

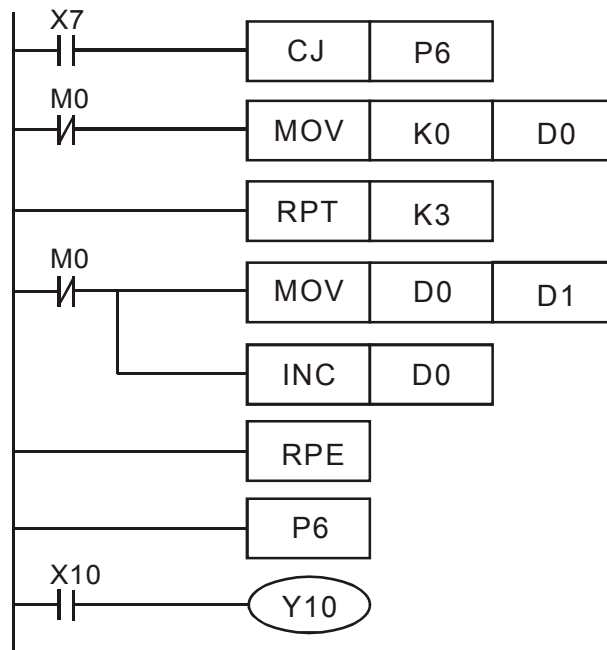
程序范例  
(一)

- ◆ 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：



程序范例  
(二)

- ◆ 当 X7 = Off 时，PM 会执行 RPT~RPE 之间的程序，当 X7 = On 时，CJ 指令执行转移至 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

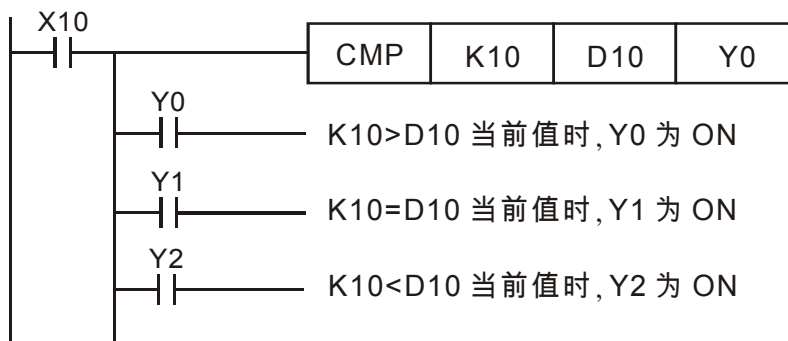
API																	适用机种
<b>10</b>	<b>D</b>	<b>CMP</b>	<b>P</b>		(S <sub>1</sub> )	(S <sub>2</sub> )	(D)	比较设定输出									10PM
																	✓
	位装置				字符装置												16 位指令 (7 STEP) CMP 连续执行型 CMPP 脉冲执行型 32 位指令 (9 STEP) DCMPP 连续执行型 DCMPP 脉冲执行型 • 标志信号：无
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*		
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*		
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																	

## 指令说明

- ◆ S<sub>1</sub>：比较值 1。S<sub>2</sub>：比较值 2。D：比较结果。
- ◆ 比较值 S<sub>1</sub> 与 S<sub>2</sub> 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。

## 程序范例

- ◆ 指定装置为 Y0，则自动占有 Y0、Y1 及 Y2。
- ◆ 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 ≥、≤、≠ 之结果时，可将 Y0~Y2 串并联即可取得。



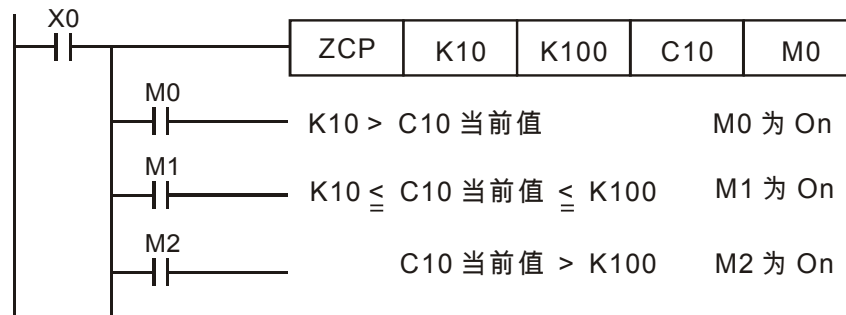
API																		适用机种		
<b>11</b>	<b>D</b>	<b>ZCP</b>	<b>P</b>		(S <sub>1</sub> )	(S <sub>2</sub> )	(S)	(D)	区域比较									10PM		
✓																				
	位装置				字符装置												16 位指令 (9 STEP) ZCP 连续执行型 ZCPP 脉冲执行型 32 位指令 (12 STEP) DZCP 连续执行型 DZCPP 脉冲执行型 • 标志信号：无			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z					
S <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*					
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*					
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																				

### 指令说明

- ◆ S<sub>1</sub>：区域比较之下限值。S<sub>2</sub>：区域比较之上限值。S：比较值。D：比较结果。
- ◆ 比较值 S 与下限 S<sub>1</sub> 及上限 S<sub>2</sub> 作比较，其比较结果在 D 作表示。
- ◆ S<sub>2</sub> 必须比 S<sub>1</sub> 大。
- ◆ D 操作数会占用连续 3 个点。

### 程序范例

- ◆ 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																		适用机种
<b>12</b>	<b>D</b>	<b>MOV</b>	<b>P</b>	(S)	(D)	数据传送												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							*	*	*	*	*	*	*	*	*			
<ul style="list-style-type: none"> <li>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</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>		

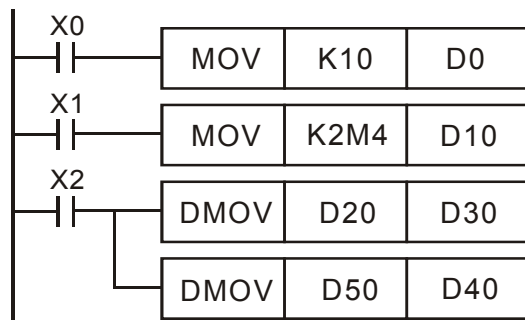
### 指令说明

- ◆ **S**：数据之来源。**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。
- ◆ 若运算结果为 32 位输出时，(如应用指令 MUL 等) 数据搬动则必须要用 **DMOV** 指令。

### 程序范例

- ◆ 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 当前值传送至 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) 当前值传送至 (D31、D30) 数据寄存器内。同时，将 (D51、D50) 当前值传送至 (D41、D40) 数据寄存器内。





API				(S) (m1) (m2) (D)													适用機種	
13		SMOV	P															10PM
																		✓

	位装置				字符装置													
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z			
S							*	*	*	*	*	*	*	*	*			16 位指令 (11 STEP) SMOV 连续执行型 SMOV <sub>P</sub> 脉冲执行型 <hr/> 32 位指令 - - - - • 标志信号：M1168 SMOV 工作模式指定
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)。

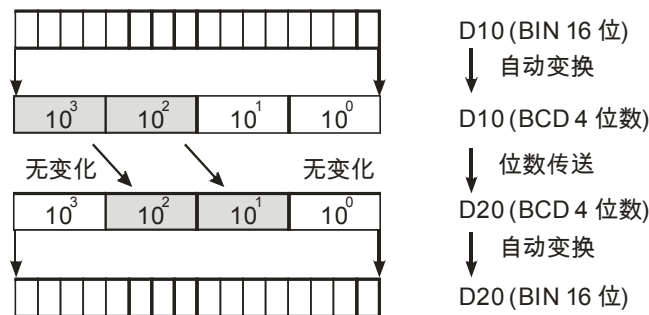
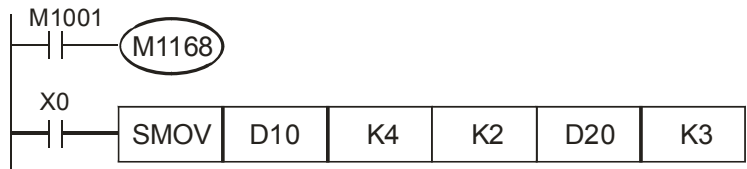
指令说明

- ◆ **S**：数据源。**m<sub>1</sub>**：数据来源传送起始位数。**m<sub>2</sub>**：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 允许操作 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。
- ◆ **m<sub>1</sub>** 的范围: 1 - 4
- ◆ **m<sub>2</sub>** 的范围: 1 - **m<sub>1</sub>** (不能大于 **m<sub>1</sub>**)
- ◆ **n** 的范围: **m<sub>2</sub>** - 4 (不能小于 **m<sub>2</sub>**)

# 5 应用指令分类及基本使用

程序范例  
(一)

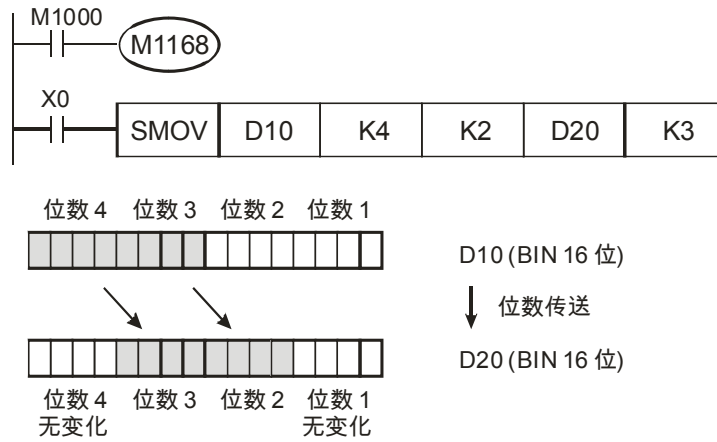
- ◆ 当 M1168=Off 时(BCD 模式), X0=On, 指定 D10 的 10 进制数值的第 4 位数(也即千位数)开始往低位计算的 2 位数内容传送至 D20 的 10 进制数值的第 3 位数(也即百位数)开始往低位计算的 2 位数中。而 D20 的  $10^3$  及  $10^0$  于本指令被执行后内容没有变化。
- ◆ 当 BCD 值超过 0~9,999 的范围时, PM 判定为运算错误, 指令不执行, M1067, M1068=On, D1067 记录错误代码 0E18 (Hex)。



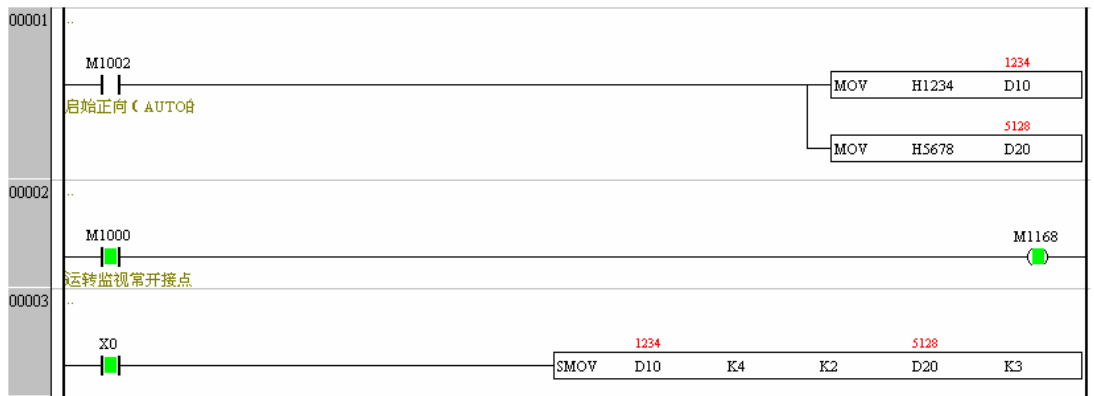
- ◆ 如果在执行前 D10=H1234, D20=H5678, 执行后 D10 的值保持不变, 而 D20=H5128。(实际执行结果为 09A2)

## 程序范例 (二)

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

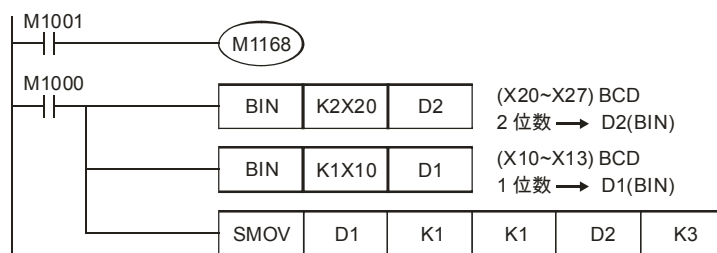
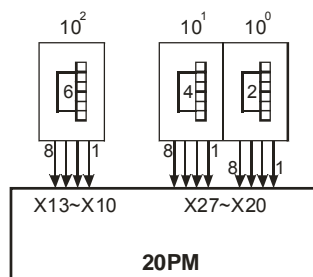


执行结果↓



## 程序范例 (三)

- ◆ 将右 2 位指拨开关传送至 D2 的右 2 位数, 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
<b>14</b>	<b>D</b>	<b>CML</b>	<b>P</b>	(S)	(D)												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							*	*	*	*	*	*	*	*	*					

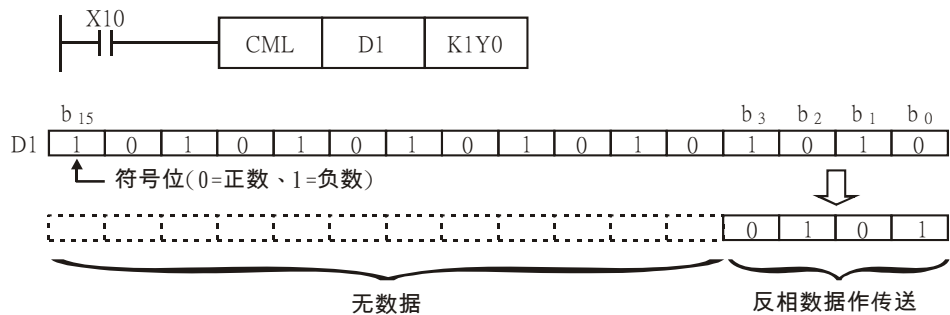
• 操作数使用注意：此指令有支持 V Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置)  
 各装置使用范围请参考功能规格表  
 指定 KnX(Y/M/S)等装置的位装置时，限于 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。

• 标志信号：无

### 指令说明

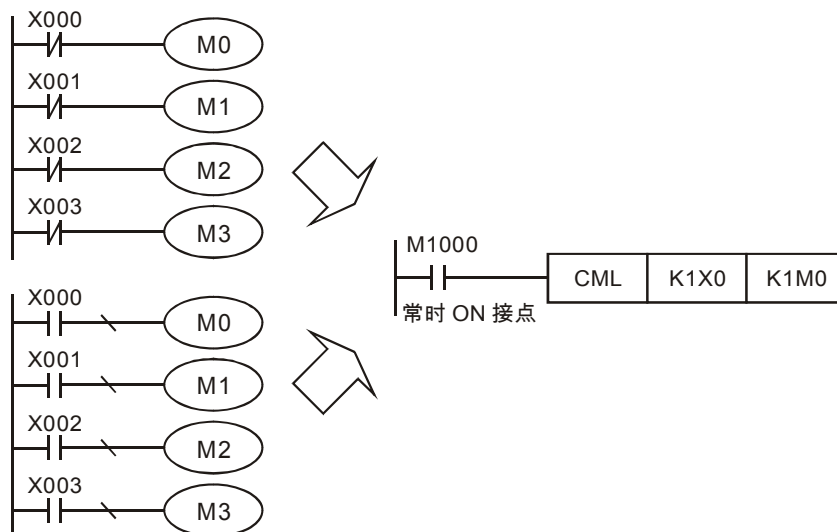
- ◆ S：传送数据来源。D：传送的目的地装置。
- ◆ 将 S 的内容全部取反(0→1, 1→0)传送到 D 当中。
- ◆ 当 X10=On 时，将 D1 的 b0~b3 内容反相后传送到 Y0~Y3。

### 程序范例 (一)



### 程序范例 (二)

- ◆ 下图左边的回路当也可以使用 CML 指令来表示，如下图右所示。



API																	适用機種
<b>15</b>		<b>BMOV</b>	<b>P</b>	(S)	(D)	(n)	全部传送										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					*	*					*	*	*					

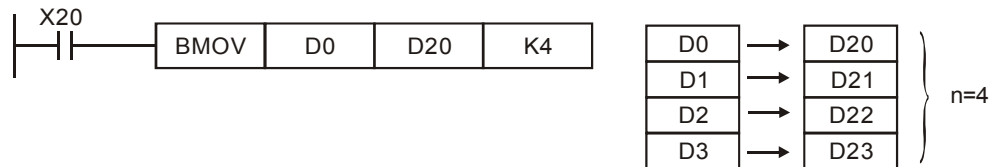
• 标志信号：无

• 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)  
 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)。

### 指令说明

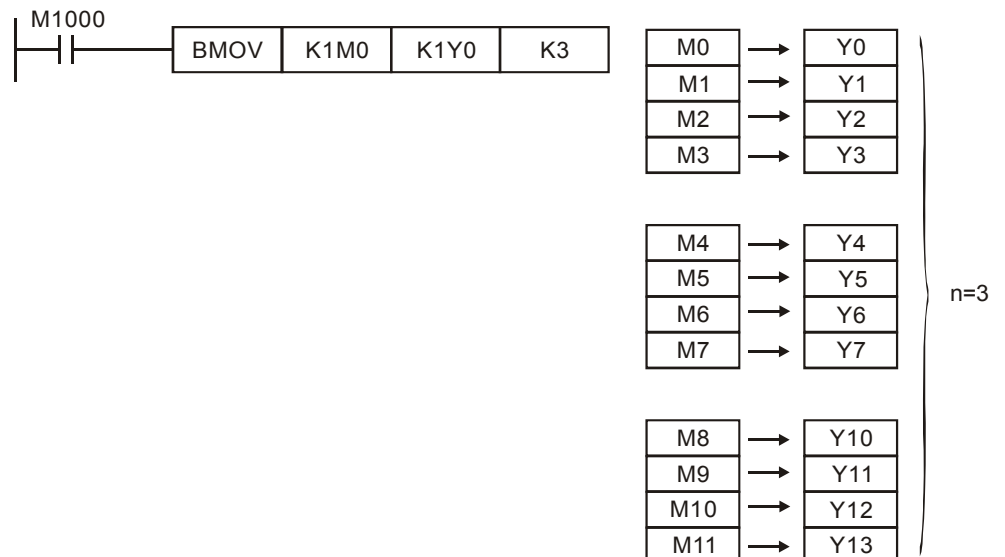
- ◆ **S**：来源装置起始。◆ **D**：目的地装置起始。◆ **n**：传送区块长度。
- ◆ 此指令用于传送多笔数据到新的寄存器。◆ **S** 所指定的装置起始号码开始算 **n** 个寄存器的内容被传送至 **D** 所指定的装置起始号码开始算 **n** 个寄存器当中，如果 **n** 所指定点数超过该装置的使用范围时，只有有效范围被传送。
- ◆ **n** 的范围 1~512。
- ◆ 当 X20=On, D0~D3 的内容被传送到 D20~D23 中。

### 程序范例 (一)



### 程序范例 (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。



# 5 应用指令分类及基本使用

## 程序范例 (三)

- ◆ 为了防止两个操作数所指定传送的号码重复时，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
<b>16</b>		<b>FMOV</b>		<b>P</b>	(S)	(D)	(n)	多点传送									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	FMOV P	
S							*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*					
n					*	*												

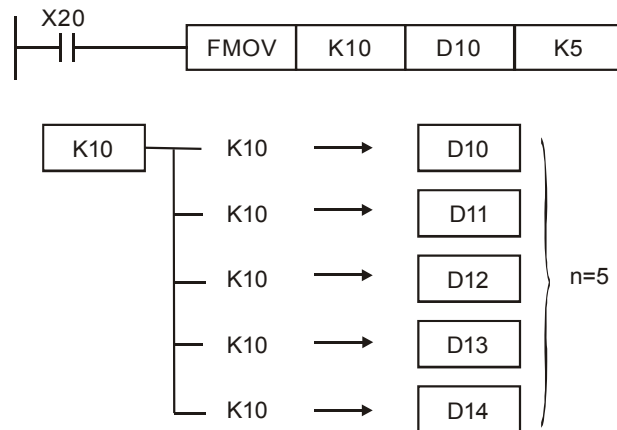
• 标志信号：无

• 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)。

指令说明

- ◆ S：数据的来源。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。

程序范例



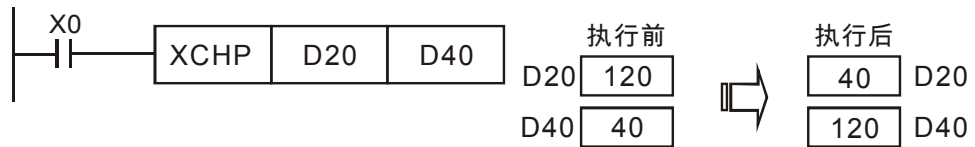
# 5 应用指令分类及基本使用

API																		适用機種
<b>17</b>	<b>D</b>	<b>XCH</b>	<b>P</b>			(D1)	(D2)	数据的交换										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		
D <sub>1</sub>							*	*	*	*	*	*	*	*	*			
D <sub>2</sub>							*	*	*	*	*	*	*	*	*			
<ul style="list-style-type: none"> <li>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)</li> <li>各装置使用范围请参考功能规格表</li> <li>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)。</li> </ul>																32 位指令 (9 STEP)		
																DXCH 连续执行型 DFXCHP 脉冲执行型		
																标志信号：无		

### 指令说明

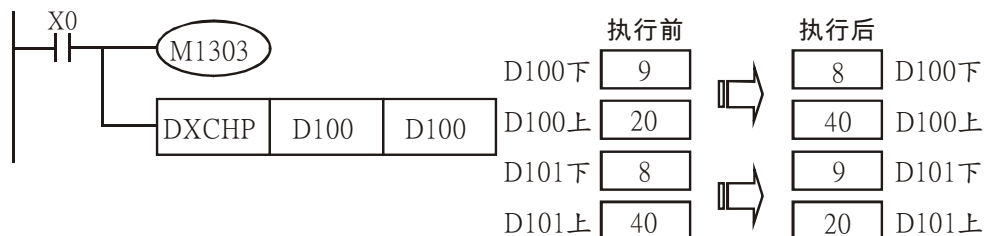
- ◆ D<sub>1</sub>：互相交换的数据 1。D<sub>2</sub>：互相交换的数据 2。
- ◆ 将 D<sub>1</sub> 与 D<sub>2</sub> 的内容互换。
- ◆ 此指令最好以脉冲的方式执行(XCHP)。
- ◆ 当 X0=Off→On, D20 与 D40 的内容互换。

### 程序范例



### 补充说明

- ◆ 16 位指令当 D<sub>1</sub> 及 D<sub>2</sub> 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 D<sub>1</sub> 及 D<sub>2</sub> 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。





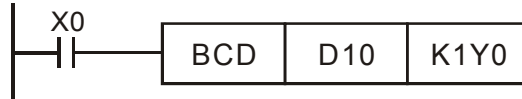
API																		适用機種
<b>18</b>	<b>D</b>	<b>BCD</b>	<b>P</b>	(S)	(D)	BCD 转换												10PM
																		✓
	位装置				字符装置												16 位指令 (5 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	BCD 连续执行型 BCDP 脉冲执行型		
S							*	*	*	*	*	*	*	*	*			
D								*	*	*	*	*	*	*	*			
• 操作数使用注意：此指令有支持 V、Z 装置（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																• 标志信号： Ox O100 M1793 M1953 运算错误标志 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）。

### 程序范例

- ◆ 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		(S)	(D)	BIN 转换										适用機種		
19	D		P															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			
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																• 标志信号： Ox      O100 M1793   M1953    运算错误标志 Operation error flag		

## 指令说明

- ◆ S：数据之来源。D：转换之结果数据。
- ◆ 二进制 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。

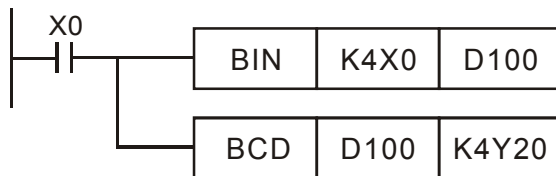
## 程序范例

- ◆ 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：
  1. 当 DVP-PM 要从外界读取一个 BCD 类型指拨开关时，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。
  2. 当 DVP-PM 要将内部储存的数据经由外界一个 BCD 类型的 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。
  3. 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																		适用机种
<b>20</b>	<b>D</b>	<b>ADD</b>	<b>P</b>	(S <sub>1</sub> )	(S <sub>2</sub> )	(D)	BIN 加法										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 <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*			
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*			
D							*	*	*	*	*	*	*	*	*			
<ul style="list-style-type: none"> <li>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</li> </ul>																	<ul style="list-style-type: none"> <li>标志信号</li> <li>Ox      O100</li> <li>M1808   M1968   零标志 Zero flag</li> <li>M1809   M1969   借位标志 Borrow flag</li> <li>M1810   M1970   进位标志 Carry flag</li> <li>请参考下列补充说明</li> </ul>	

### 指令说明

- ◆ S<sub>1</sub>：被加数。S<sub>2</sub>：加数。D：和。
- ◆ 将两个数据源：S<sub>1</sub> 及 S<sub>2</sub> 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)。
- ◆ 加法相关标志变化。

#### 16 位 BIN 加法：

1. 运算结果为 0 时，零标志 (Zero flag) 为 On。
2. 运算结果小于 -32,768 时，借位标志 (Borrow flag) 为 On。
3. 运算结果大于 32,767 时，进位标志 (Carry flag) 为 On。

#### 32 位 BIN 加法：

1. 运算结果为 0 时，零标志 (Zero flag) 为 On。
2. 运算结果小于 -2,147,483,648 时，借位标志 (Borrow flag) 为 On。
3. 运算结果大于 2,147,483,647 时，进位标志 (Carry flag) 为 On。

### 程序范例 (一)

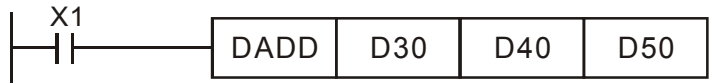
- ◆ 16 位 BIN 加法：当 X0=On 时，被加数 D0 内容加上加数 D10 之内容将结果存在 D20 之内容当中。



# 5 应用指令分类及基本使用

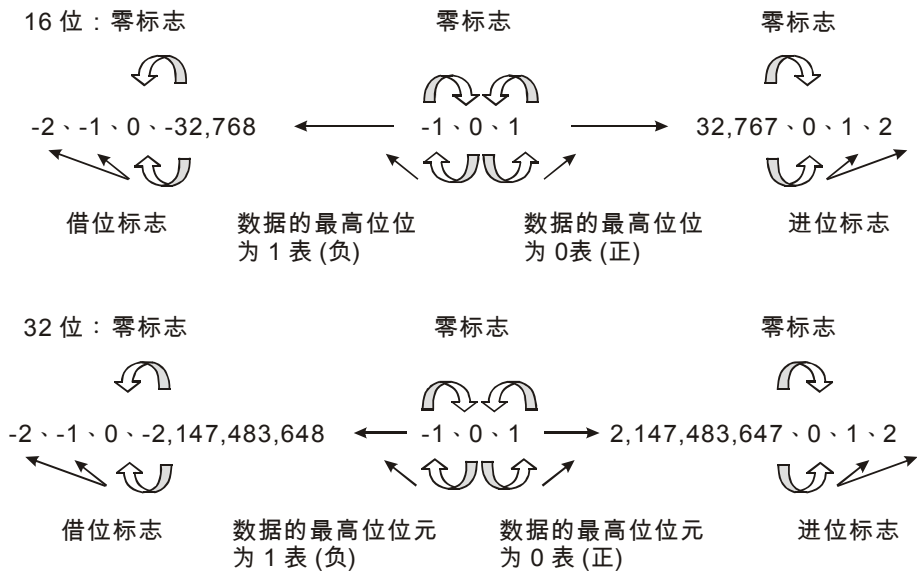
程序范例  
(二)

- ◆ 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

- ◆ 标志动作与数值的正负关系：



API																	适用機種		
<b>21</b>	<b>D</b>	<b>SUB</b>	<b>P</b>	(S <sub>1</sub> )	(S <sub>2</sub> )	(D)	BIN 減法										10PM		
																	✓		
	位裝置				字符裝置												16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SUB	連續執行型	SUBP	脈衝執行型
S <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*				
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*				
D							*	*	*	*	*	*	*	*	*				
<p>• 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</p>																	<p>• 標志信號</p> <p>Ox      O100</p> <p>M1808   M1968   零標志 Zero flag</p> <p>M1809   M1969   借位標志 Borrow flag</p> <p>M1810   M1970   進位標志 Carry flag</p> <p>• 請參考下列補充說明</p>		

**指令說明**

- ◆ S<sub>1</sub>：被減數。S<sub>2</sub>：減數。D：差。
- ◆ 將兩個數據源：S<sub>1</sub> 及 S<sub>2</sub> 以 BIN 方式相減的結果存於 D。
- ◆ 各數據的最高位為符號位 0 表 (正) 1 表 (負)，因此可做代數減法運算。
- ◆ 減法相關標志變化。

16 位 BIN 減法：

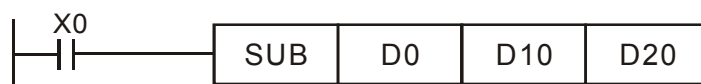
1. 運算結果為 0 時，零標志 (Zero flag) 為 On。
2. 運算結果小於 -32,768 時，借位標志 (Borrow flag) 為 On。
3. 運算結果大於 32,767 時，進位標志 (Carry flag) 為 On。

32 位 BIN 減法：

1. 運算結果為 0 時，零標志 (Zero flag) 為 On。
2. 運算結果小於 -2,147,483,648 時，借位標志 (Borrow flag) 為 On。
3. 運算結果大於 2,147,483,647 時，進位標志 (Carry flag) 為 On。

- ◆ 標志動作與數值的正負關係參考標志動作與數值的正負關係請參考上頁指令 ADD 之補充說明。
- ◆ 16 位 BIN 減法：當 X0=On 時，將 D0 內容減掉 D10 內容將差存在 D20 之內容中。

**程序範例 (一)**

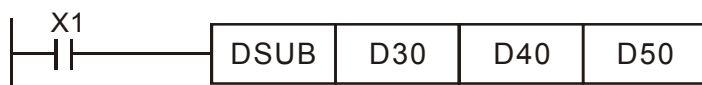


## 5 应用指令分类及基本使用

---

程序范例  
(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)

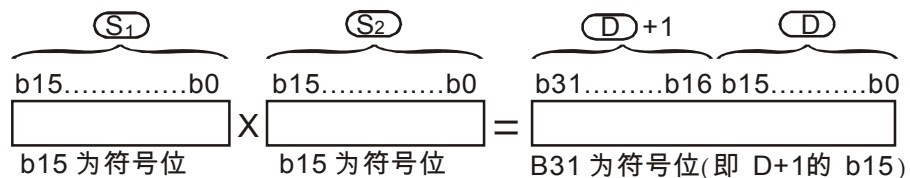


API																		适用机种	
<b>22</b>	<b>D</b>	<b>MUL</b>	<b>P</b>	(S <sub>1</sub> )	(S <sub>2</sub> )	(D)	BIN 乘法										10PM		
✓																			
	位装置				字符装置												16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	MUL	连续执行型	MULP	脉冲执行型
S <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*				
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*				
D											*	*	*						
<ul style="list-style-type: none"> <li>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</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>			

### 指令说明

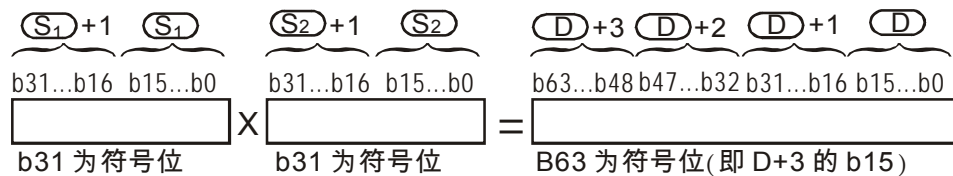
- ◆ S<sub>1</sub>：被乘数。S<sub>2</sub>：乘数。D：积。
- ◆ 将两个数据源：S<sub>1</sub> 及 S<sub>2</sub> 以有号数二进制方式相乘后的积存于 D。必须注意 16 位及 32 位运算时，S<sub>1</sub>、S<sub>2</sub> 及 D 的正负号位。

- ◆ 16 位 BIN 乘法运算：



符号位=0 为正数，符号位=1 为负数。

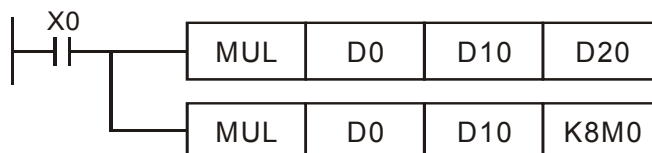
- ◆ 32 位 BIN 乘法运算：



符号位=0 为正数，符号位=1 为负数。

### 程序范例

- ◆ 16 位 D0 乘上 16 位 D10 其结果是 32 位之积，上 16 位存于 D21，下 16 位存于 D20 内，结果之正负由最左边位之 Off/On 来代表正或负值。

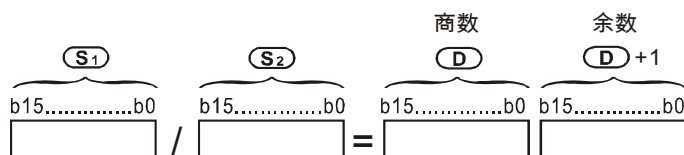


# 5 应用指令分类及基本使用

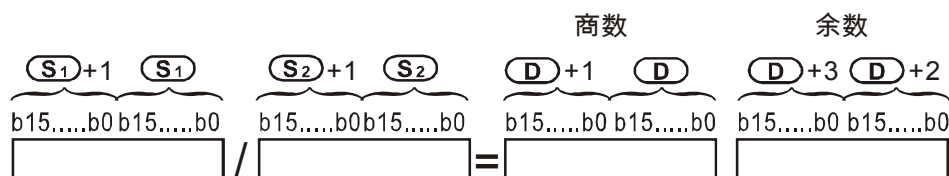
API																		适用機種			
<b>23</b>	<b>D</b>	<b>DIV</b>	<b>P</b>	(S <sub>1</sub> )	(S <sub>2</sub> )	(D)	BIN 除法										10PM				
																		✓			
	位裝置				字裝置												16 位指令 (7 STEP)				
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	DIV	連續執行型	DIVP	脈衝執行型		
S <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*						
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*						
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																		• 標志信號：無			

## 指令說明

- ◆ S<sub>1</sub>：被除數。S<sub>2</sub>：除數。D：商及余數。
- ◆ 將兩個數據源：S<sub>1</sub> 及 S<sub>2</sub> 以有號數二進制方式相除後的商及余數存於 D。必須注意 16 位及 32 位運算時，S<sub>1</sub>、S<sub>2</sub> 及 D 的正負號位。
- ◆ 除數為 0 時，指令不執行。
- ◆ 16 位 BIN 除法運算：

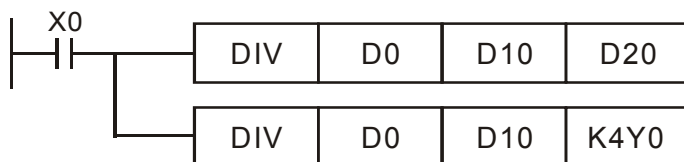


- ◆ 32 位 BIN 除法運算：



## 程序範例

- ◆ 當 X0=On 時，被除數 D0 除以除數 D10 而結果商被指定放於 D20，余數指定放於 D21 內。所得結果之正負由最高位位之 Off/On 來代表正或負值。



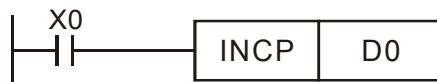


API																		适用機種		
<b>24</b>	<b>D</b>	<b>INC</b>	<b>P</b>	<b>(D)</b>	<b>BIN 加一</b>													10PM		
																		✓		
D	位装置				字符装置												16 位指令 (3 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	INC 连续执行型		INCP 脉冲执行型		
							*	*	*	*	*	*	*	*	*	DINC 连续执行型		DINCP 脉冲执行型		
<ul style="list-style-type: none"> <li>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)</li> <li>各装置使用范围请参考功能规格表</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>				

### 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。
- ◆ 当 X0=Off→On 时，D0 内容自动加 1。

### 程序范例



# 5 应用指令分类及基本使用

API																		适用機種
<b>25</b>	<b>D</b>	<b>DEC</b>	<b>P</b>															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	DEC 连续执行型    DECP 脉冲执行型		
D							*	*	*	*	*	*	*	*	*	DDEC 连续执行型    DDECP 脉冲执行型		
<ul style="list-style-type: none"> <li>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)</li> <li>各装置使用范围请参考功能规格表</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>		

## 指令说明

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

## 程序范例



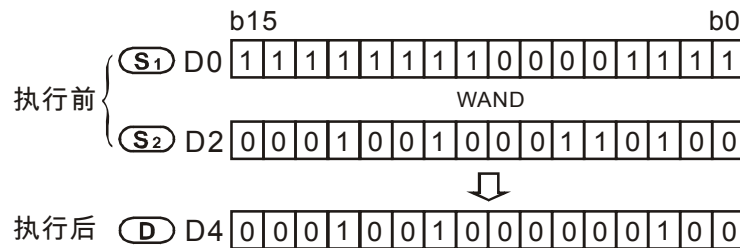
API																		适用機種	
<b>26</b>	<b>D</b>	<b>WAND</b>	<b>P</b>	(S1)	(S2)	(D)	逻辑与(AND)运算										10PM		
✓																			
	位装置				字符装置												16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WAND	连续执行型	WANDP	脉冲执行型
S1					*	*	*	*	*	*	*	*	*	*	*				
S2					*	*	*	*	*	*	*	*	*	*	*				
D								*	*	*	*	*	*	*	*				
<ul style="list-style-type: none"> <li>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</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>			

### 指令说明

- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。D：运算结果。
- ◆ 两个数据源：S<sub>1</sub> 及 S<sub>2</sub> 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。

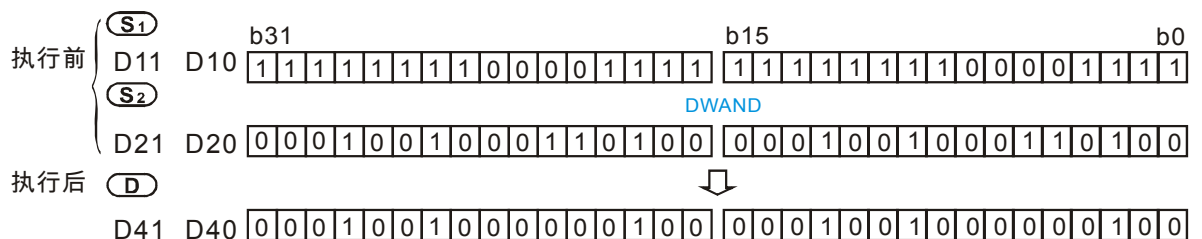
### 程序范例 (一)

- ◆ 当 X0=On 时，16 位 D0 与 D2 作 WAND，逻辑与 (AND) 运算，将结果存于 D4 中。



### 程序范例 (二)

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



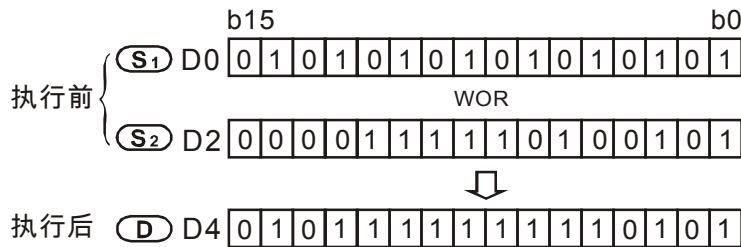
# 5 应用指令分类及基本使用

API																	适用机种		
27	D	WOR	P	(S1)	(S2)	(D)	逻辑或(OR)运算										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								*	*	*	*	*	*	*	*				
<ul style="list-style-type: none"> <li>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</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>			

### 指令说明

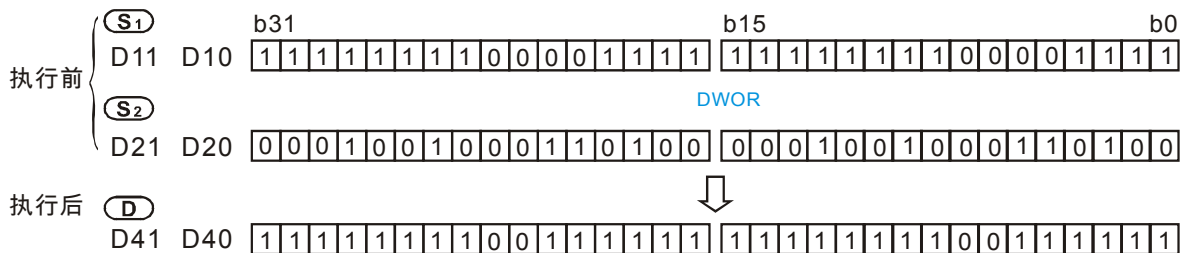
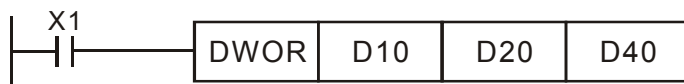
- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。D：运算结果。
- ◆ 两个数据源：S<sub>1</sub> 及 S<sub>2</sub> 作逻辑的'或' (OR) 运算结果存于 D。
- ◆ 逻辑的'或' (OR) 运算之规则为任一为 1 结果为 1。
- ◆ 当 X0=On 时，16 位 D0 与 D2 作 WOR，逻辑或 (OR) 运算，将结果存于 D4 中。

### 程序范例 (一)



### 程序范例 (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。

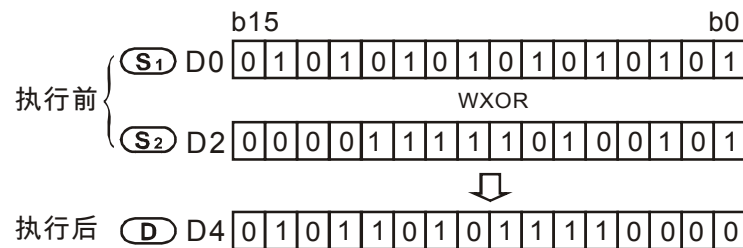
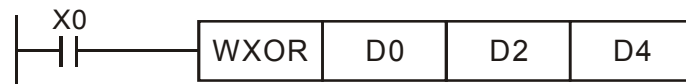


API																	适用機種	
<b>28</b>	<b>D</b>	<b>WXOR</b>	<b>P</b>	(S1)	(S2)	(D)	逻辑异或(XOR)运算										10PM	
✓																		
		位装置				字符装置										16 位指令 (7 STEP)		
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	WXOR 连续执行型	WXORP 脉冲执行型
S1					*	*	*	*	*	*	*	*	*	*	*	*		
S2					*	*	*	*	*	*	*	*	*	*	*	*		
D								*	*	*	*	*	*	*	*	*		
• 标志信号：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																		
• 32 位指令 (9 STEP) DWXOR 连续执行型 DWXORP 脉冲执行型																		

### 指令说明

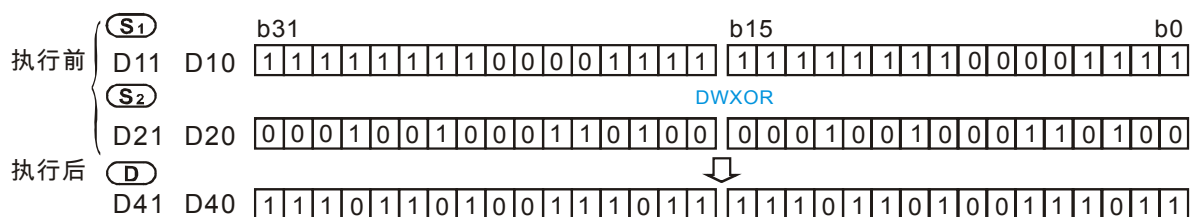
- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。D：运算结果。
- ◆ 两个数据源：S<sub>1</sub> 及 S<sub>2</sub> 作逻辑的'异或' (XOR) 运算结果存于 D。
- ◆ 逻辑的'异或' (XOR) 运算之规则为两者相同结果为 0，两者不同结果为 1。
- ◆ 当 X0=On 时，16 位 D0 与 D2 作 WXOR，逻辑异或 (XOR) 运算，将结果存于 D4 中。

### 程序范例 (一)



### 程序范例 (二)

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



# 5 应用指令分类及基本使用

API	W	NEG		<b>D</b>	2 的补码										适用機種					
29	D		P												10PM					
															✓					
D	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	
<ul style="list-style-type: none"> <li>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)</li> <li>各装置使用范围请参考功能规格表</li> </ul>															32 位指令 (9 STEP)					
															DNEG	连续执行型	DNEGP	脉冲执行型	<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>	

### 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**)。

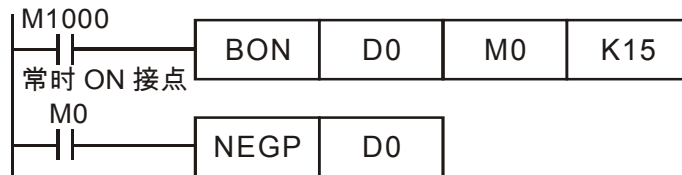
### 程序范例 (一)

- ◆ 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。



### 程序范例 (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。



### 程序范例 (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。



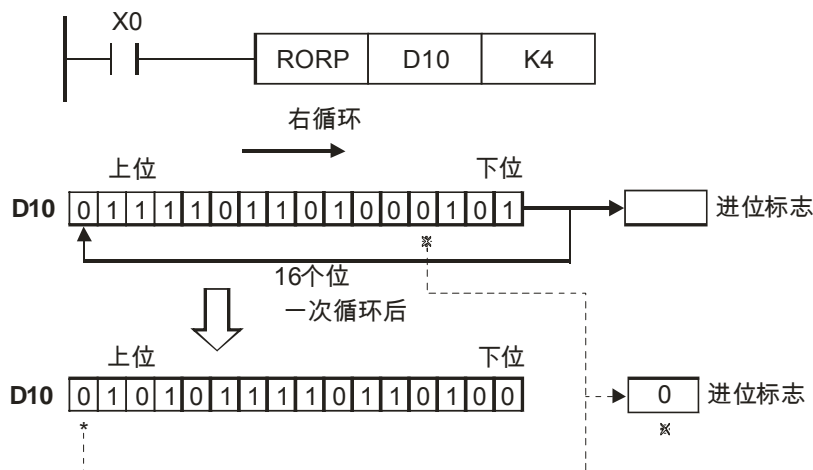
# 5 应用指令分类及基本使用

API																	适用機種	
<b>30</b>	<b>D</b>	<b>ROR</b>	<b>P</b>														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"> <li>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)。</li> </ul>																32 位指令 (9 STEP) DROR 连续执行型 DRORP 脉冲执行型 <ul style="list-style-type: none"> <li>标志信号：无</li> <li>Ox O100</li> <li>M1810 M1970 进位标志 Carry flag</li> </ul> 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循环 **n** 个位。
- ◆ 最后旋转的一位的状态被传送到进位标志。
- ◆ 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 ≤ n ≤ 16 (16 位), 1 ≤ n ≤ 32 (32 位)。
- ◆ 当 X0 从 Off→On 变化时，D10 的 16 个位以 4 个位为一组往右循环，如下图所示标明※的位内容被传送到进位标志信号内。

## 程序范例





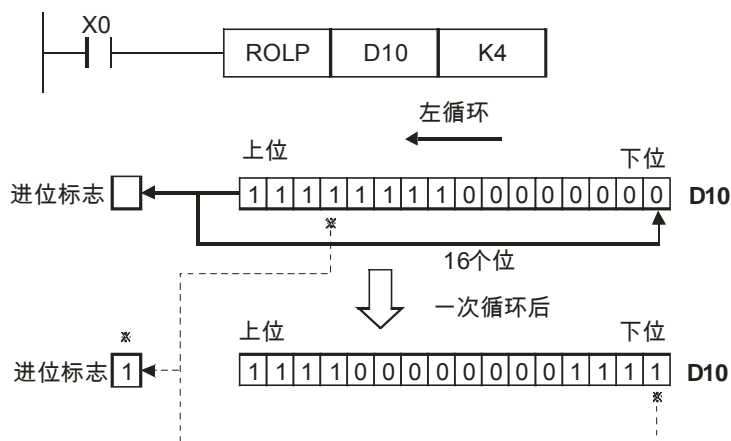
API																	适用机种	
<b>31</b>	<b>D</b>	<b>ROL</b>	<b>P</b>	(D)	(n)												10PM	
																	✓	
	位装置				字符装置												16 位指令 (5 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		ROL 连续执行型 ROLP 脉冲执行型	
D							*	*	*	*	*	*	*	*	*			
n					*	*												
<p>• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 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>• 标志信号</p> <p>Ox O100 M1810 M1970 进位标志 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** 所指定的装置内容一次向左循环 **n** 个位。
- ◆ 最后旋转的一位的状态被传送到进位标志 (进位标志)。
- ◆ 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)。

### 程序范例

- ◆ 当 X0 从 Off→On 变化时，D10 的 16 个位以 4 个位一组往左循环，如下图所示标明\*的位内容被传送至进位标志信号内。



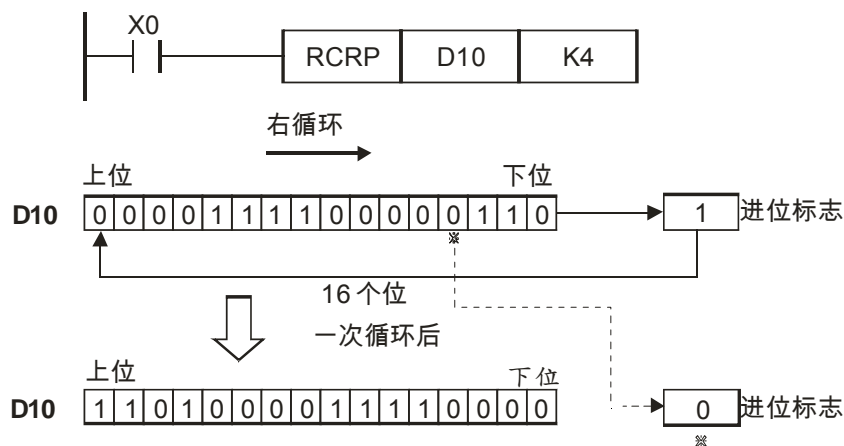
# 5 应用指令分类及基本使用

API																	适用機種	
<b>32</b>	<b>D</b>	<b>RCR</b>	<b>P</b>	<b>(D)</b>	<b>(n)</b>	附进位标志右循环										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					*	*												
<ul style="list-style-type: none"> <li>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)</li> </ul>																32 位指令 (9 STEP) DRCP 连续执行型 DRCRP 脉冲执行型		
<ul style="list-style-type: none"> <li>标志信号 Ox O100 M1810 M1970 进位标志 Carry flag</li> <li>请参考下列补充说明</li> </ul>																		

## 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标志，一次向右循环 **n** 个位。
- ◆ 最后旋转的一位的状态被传送到进位标志 (进位标志)。在下面的指令操作时，进位标志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)。

## 程序范例

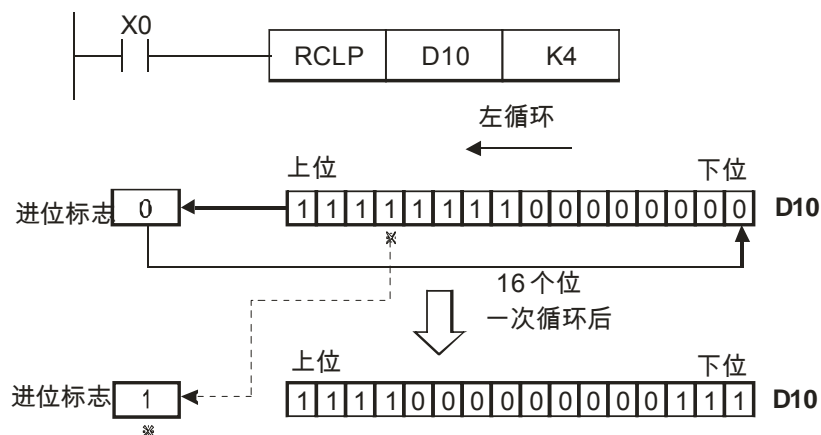


API																		适用机种
<b>33</b>	<b>D</b>	<b>RCL</b>	<b>P</b>	<b>(D)</b>	<b>(n)</b>	附进位标志左循环												10PM
																		✓
	位装置				字符装置												16 位指令 (5 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z			RCL 连续执行型 RCLP 脉冲执行型
D							*	*	*	*	*	*	*	*	*			
n					*	*												
<ul style="list-style-type: none"> <li>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)</li> </ul>																<ul style="list-style-type: none"> <li>标志信号 Ox O100 M1810 M1970 进位标志 Carry flag</li> <li>请参考下列补充说明</li> </ul>		

### 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标志，一次向左循环 **n** 个位。
- ◆ 最后旋转的一位的状态被传送到进位标志。在下面的指令操作时，M1022 第一个被传送到目的设备。
- ◆ 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 个位连同进位标志共 17 个位以 4 个位一组往左循环，如下图所示标明※的位内容被传送至进位标志信号内。

### 程序范例



# 5 应用指令分类及基本使用

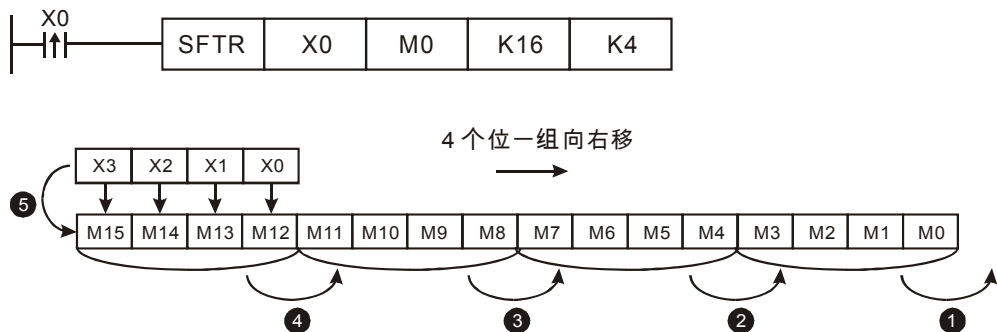
API																		适用機種	
<b>34</b>		<b>SFTR</b>	<b>P</b>	(S)	(D)	(n1)	(n2)	位右移										10PM	
																		✓	
		位装置				字符装置												16 位指令 (9 STEP)	
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		SFTR 连续执行型 SFTRP 脉冲执行型	
S		*	*	*	*														
D			*	*	*														
n1						*	*												
n2						*	*												
• 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																			
• 标志信号：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。**n<sub>1</sub>**：欲移位之数据长度。**n<sub>2</sub>**：一次移位的位数。
- ◆ 将 **D** 开始之起始编号，具有 **n<sub>1</sub>** 个数字元 (移位寄存器长度) 的位装置，以 **n<sub>2</sub>** 位个数来右移。而 **S** 开始起始编号以 **n<sub>2</sub>** 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)。
- ◆ **n<sub>1</sub>**, **n<sub>2</sub>** 操作数有效范围:  $1 \leq n_2 \leq n_1 \leq 1024$ 。

## 程序范例

- ◆ 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 → M3~M0
  - ② M7~M4 → M3~M0
  - ③ M11~M8 → M7~M4
  - ④ M15~M12 → M11~M8
  - ⑤ X3~X0 → M15~M12 完成



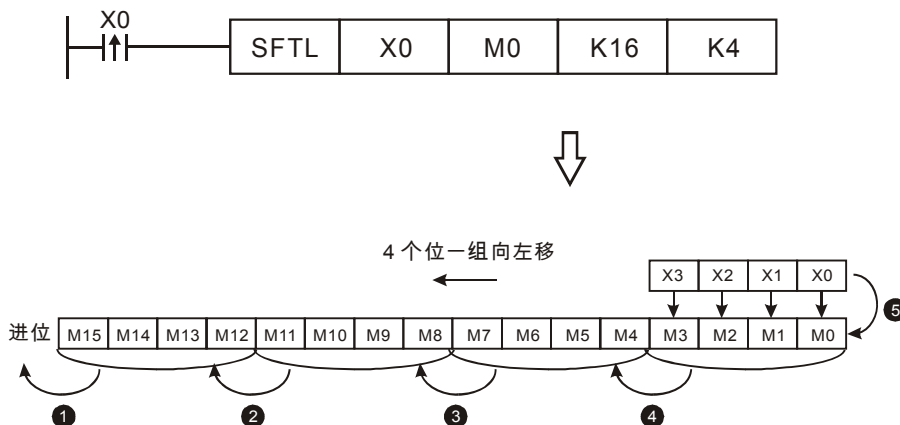
API																		适用機種
<b>35</b>		<b>SFTL</b>	<b>P</b>	(S)	(D)	(n1)	(n2)	位左移										10PM
																		✓
	位装置				字符装置												16 位指令 (9 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SFTL 连续执行型 SFTLP 脉冲执行型		
S	*	*	*	*													32 位指令	
D		*	*	*													-	
n1					*	*											• 标志信号：无	
n2					*	*												
<ul style="list-style-type: none"> <li>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</li> </ul>																		

### 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。**n<sub>1</sub>**：欲移位之数据长度。**n<sub>2</sub>**：一次移位的位数。
- ◆ 将 **D** 开始之起始编号，具有 **n<sub>1</sub>** 个数字元 (移位寄存器长度) 的位装置，以 **n<sub>2</sub>** 位个数来左移。而 **S** 开始起始编号以 **n<sub>2</sub>** 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)。
- ◆ **n<sub>1</sub>**, **n<sub>2</sub>** 操作数有效范围:  $1 \leq n_2 \leq n_1 \leq 1024$ 。

### 程序范例

- ◆ 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



# 5 应用指令分类及基本使用

API																	适用机种
<b>36</b>		<b>WSFR</b>	<b>P</b>	(S)	(D)	(n1)	(n2)	寄存器右移								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					*	*												

• 标志信号：无

16 位指令 (9 STEP)  
 WSFR 连续执行型 WSFRP 脉冲执行型

32 位指令  
 - - - - -

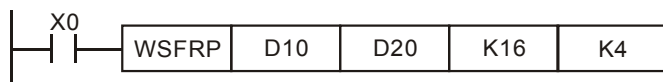
• 操作数使用注意：此指令有支持 V·Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置)  
 各装置使用范围请参考功能规格表  
 指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)

### 指令说明

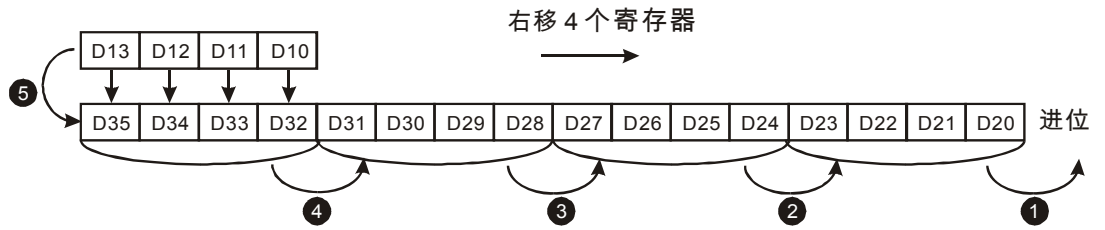
- ◆ S：移位装置之起始编号。D：欲移位装置之起始编号。n<sub>1</sub>：欲移位之数据长度。n<sub>2</sub>：一次移位之字符数。
- ◆ 将 D 开始之起始编号，具有 n<sub>1</sub> 个字符长度的数据串行，以 n<sub>2</sub> 个字符来右移。而 S 开始起始编号以 n<sub>2</sub> 字符个数移入 D 中来填补字符空位。
- ◆ 本指令一般都是使用脉冲执行型指令(WSFRP)。
- ◆ 当操作数 S, D 使用位数据类型时，数据类型必须是匹配的。例如，一个类型是 KnX, KnY, KnM, KnS 另一个类型是 T, C, D。
- ◆ 当使用操作数 S 和 D 数据类型时，Kn 的值必须是匹配的。
- ◆ n<sub>1</sub>, n<sub>2</sub> 操作数有效范围: 1 ≤ n<sub>2</sub> ≤ n<sub>1</sub> ≤ 512

### 程序范例 (一)

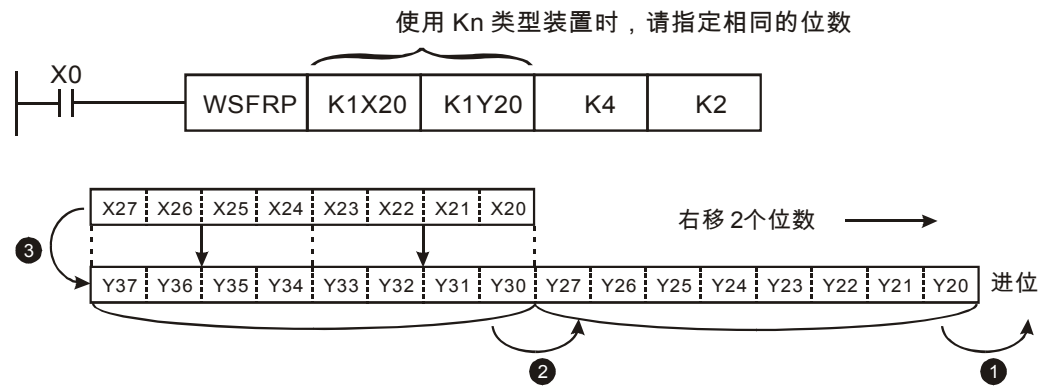
- ◆ 当 X0 从 Off → On 时，由 D20~D35 所组成的 16 个寄存器数据串行为移位区域，以 4 个寄存器来右移。
- ◆ 扫描一次的字右移行作依照下列编号 ①~⑤ 动作。
  - ① D23~D20 → 进位
  - ② D27~D24 → D23~D20
  - ③ D31~D28 → D27~D24
  - ④ D35~D32 → D31~D28
  - ⑤ D13 ~D10 → D35~D32 完成



程序范例  
(二)

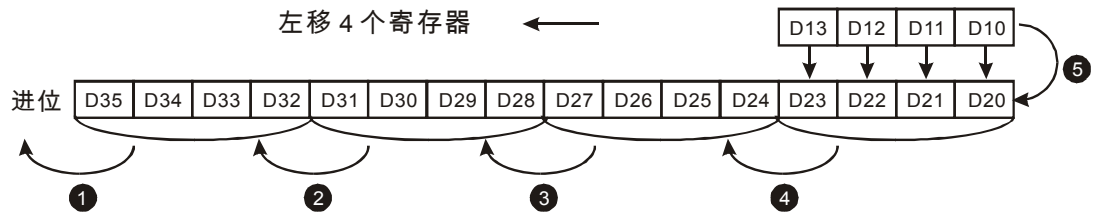


- ◆ 当 X0 从 Off → On 时, 由 Y20~Y37 所组成的位寄存器数据串行为移位区域, 以 2 个位数来右移。
- ◆ 扫描一次的字右移行作依照下列编号 ①~③ 动作。
  - ① Y27~Y20 → 进位
  - ② Y37~Y30 → Y27~Y20
  - ③ X27~X20 → Y37~Y30 完成









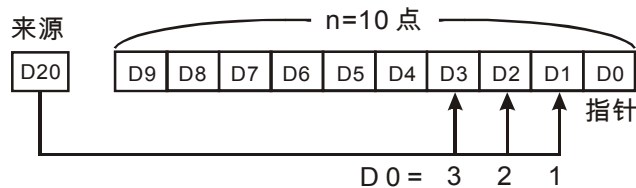
# 5 应用指令分类及基本使用

API																	适用機種	
<b>38</b>		<b>SFWR</b>	<b>P</b>	(S)	(D)	(n)	移位写入										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					*	*										-		
<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)</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)。</p>																标志信号		
																Ox	O100	
																M1808	M1968	零标志信号 Zero flag

## 指令说明

- ◆ **S**：移位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，进位标志信号 **On**。
- ◆ 当指针的内容超过 **n-1** 时，本指令不再处理写入的新值，进位标志信号 **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:  $2 \leq n \leq 512$
- ◆ 开始先将指针 **D0** 清除为 0，当 **X0=Off → On** 变化时，**D20** 的内容被传送至 **D1** 当中，指针 **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。
- ◆ 指令执行一次移位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 指针 **D0** 内容变成 1。

## 程序范例



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。

# 5 应用指令分类及基本使用

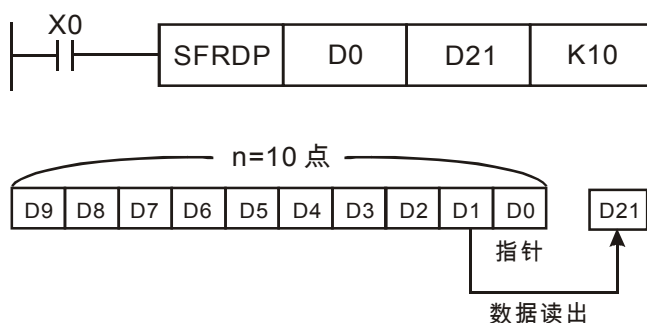
API																适用机种		
<b>39</b>		<b>SFRD</b>		<b>P</b>	(S)	(D)	(n)	移位读出								10PM		
																✓		
	位装置				字符装置												16 位指令 (9 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SFRD 连续执行型 SFRD P 脉冲执行型		
S					*	*		*	*	*	*	*	*			32 位指令		
D								*	*	*	*	*	*	*	*	-		
n					*	*										-		
<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)</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>																标志信号		
																Ox	O100	
																M1808	M1968	零标志信号 Zero flag

### 指令说明

- ◆ **S**：数据串行之起始编号。**D**：数据串行移位读出之装置。**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，零标志信号 On。
- ◆ 当指针的内容等于 0 时，本指令不再处理数据读出的动作，零标志信号 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:  $2 \leq n \leq 512$

### 程序范例

- ◆ 当 X0 从 Off→On 变化时，D1 的内容被传送至 D21 内，D9~D2 全部往右移位一个寄存器(D9 内容保持不变)，指针 D0 内容减 1。
- ◆ 执行指令一次移位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 全部往右移位一个寄存器。
  - ③ 指针 D0 内容减 1。



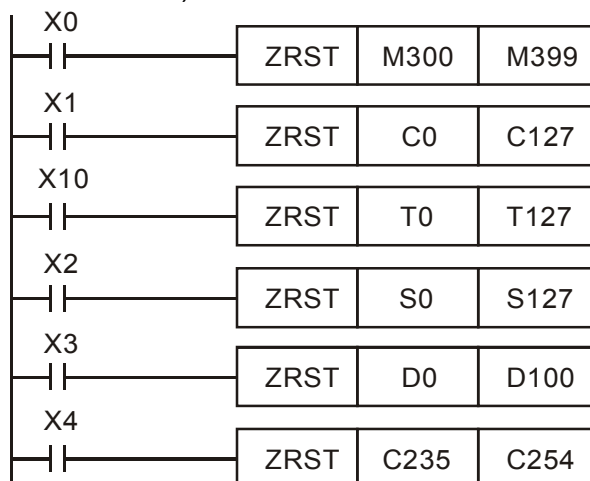
API			☺														适用機種	
<b>40</b>		<b>ZRST</b>	<b>P</b>		(D1)	(D2)											10PM	
																	✓	
	位装置				字符装置												16 位指令 (5 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		ZRST 连续执行型 ZRSTP 脉冲执行型	
D <sub>1</sub>		*	*	*							*	*	*				32 位指令 - - - - -	
D <sub>2</sub>		*	*	*							*	*	*	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

### 指令说明

- ◆ D<sub>1</sub>：区域清除起始装置。D<sub>2</sub>：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 D<sub>1</sub> 操作数编号 > D<sub>2</sub> 操作数编号时，只有 D<sub>2</sub> 指定之操作数被清除。

### 程序范例

- ◆ 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。

# 5 应用指令分类及基本使用

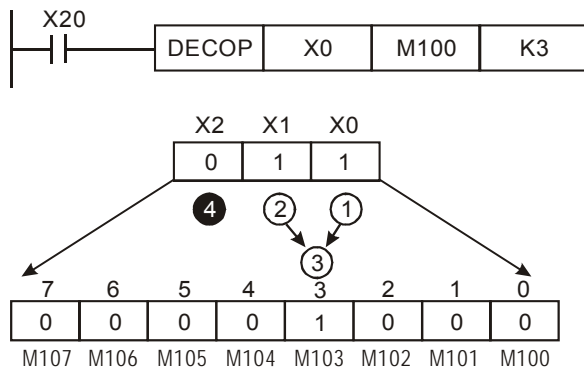
API																	适用機種	
<b>41</b>		<b>DECO</b>	<b>P</b>	(S)	(D)	(n)	译码器										10PM	
																	✓	
	位装置				字符装置										16 位指令 (7 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	DECO 连续执行型 DECO P 脉冲执行型		
S	*	*	*	*	*	*					*	*	*	*	*	32 位指令		
D		*	*	*							*	*	*	*	*	-		
n					*	*										-		
<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)</p> <p>各装置使用范围请参考功能规格表</p>																	标志信号：无	

### 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“2<sup>n</sup>” 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。

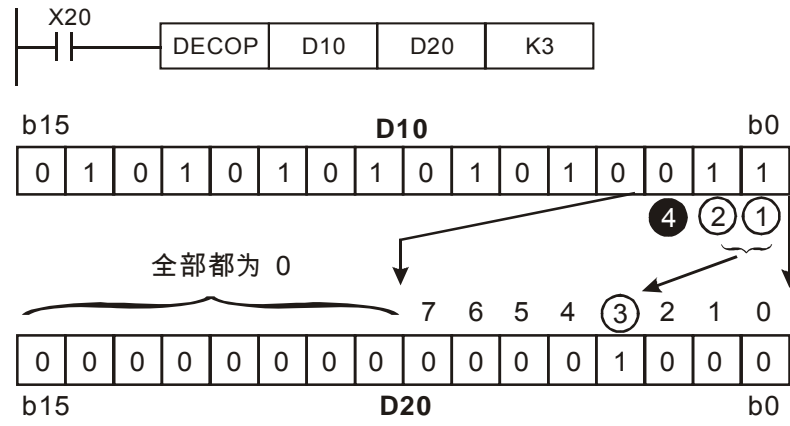
### 程序范例 (一)

- ◆ 当 D 是位装置时, n 有效范围: 0 < n ≤ 8。但是如果 n=0 or n>8, 会发生错误。
- ◆ 当 n=8 时, 可做最大解码 2<sup>8</sup>= 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。已经做解码输出者照常动作。



### 程序范例 (二)

- ◆ 当 D 为字符装置时, n 有效范围: 0 < n ≤ 8, 如果 n=0 or n>8, 会发生错误。
- ◆ 当 n=8 时, 可做最大解码 2<sup>8</sup>= 256 点。
- ◆ 当 X20 从 Off → On 时, 指令将 D10 中 (b2~ b0) 的内容值解码到 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。已经做解码输出者照常动作。



# 5 应用指令分类及基本使用

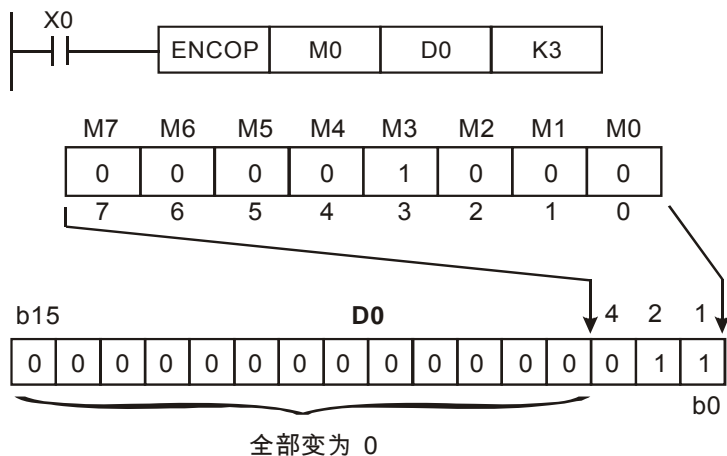
API																	适用機種	
<b>42</b>		<b>ENCO</b>	<b>P</b>	(S)	(D)	(n)	译码器										10PM	
																	✓	
	位装置				字符装置										16 位指令 (7 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ENCO 连续执行型 ENCO P 脉冲执行型		
S	*	*	*	*							*	*	*	*	*	32 位指令		
D											*	*	*	*	*	-		
n					*	*										-		
<p>• 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)</p> <p>各装置使用范围请参考功能规格表</p>																	标志信号：无	

### 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<sup>n</sup>”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 有多数位为 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=1~8,当 S 为字符装置时，n=1~4。

### 程序范例 (一)

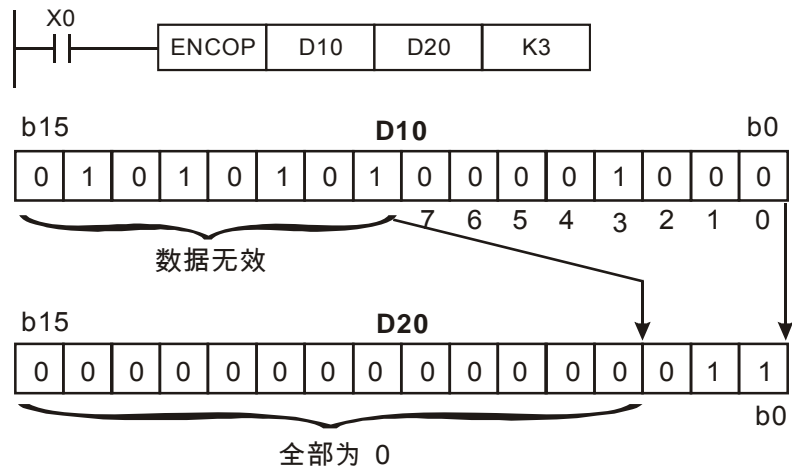
- ◆ 当 S 为位装置时，n 有效范围: 0 < n ≤ 8。如果 n=0 或者 n>8，会发生错误。
- ◆ 当 n=8 时，可做最大编码 2<sup>8</sup>= 256 点。
- ◆ 当 X0 从 Off → On 时，指令将 2<sup>3</sup> 位数据 (M0 to M7)编码存放于 D0 的下位 3 位 (b2~b0)内。D0 中未被使用的位 (b15~b3) 全部变为 0。
- ◆ 指令执行后，X0 变为 Off 后，D 内数据不变。





程序范例  
(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 → 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)。
- ◆ 当指令执行后, **X0** 变为 Off, **D** 内数据不变。



# 5 应用指令分类及基本使用

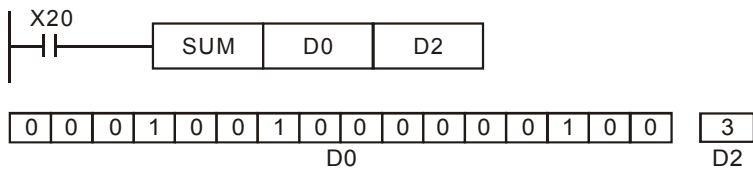
API																		适用機種		
<b>43</b>	<b>D</b>	<b>SUM</b>	<b>P</b>	(S)	(D)	On 位數量												10PM		
																		✓		
	位裝置				字裝置												16 位指令 (5 STEP)			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SUM 連續執行型 SUMP 脈衝執行型				
S							*	*	*	*	*	*	*	*	*					
D											*	*	*	*	*	32 位指令 (9 STEP)				
																DSUM 連續執行型 DSUMP 脈衝執行型				
<ul style="list-style-type: none"> <li>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 KnX(Y/M/S)等裝置的位裝置起始編號限定為 16 的倍數 (包含 0),例如:K1X0, K4Y20, K4M16,</li> </ul>																<ul style="list-style-type: none"> <li>標誌信號 Ox O100 M1808 M1968 零標誌信號 Zero flag</li> </ul>				

### 指令說明

- ◆ **S**：來源裝置。**D**：存放計數值的目的地裝置。
- ◆ 在 **S** 中，所有位內容為“1”的總數將被儲存於 **D**。
- ◆ 如果來源裝置 **S** 的 16 個位全部為“0”時，零標誌信號 On。
- ◆ 當使用 32 位指令的時候，**D** 會占用兩個寄存器。

### 程序範例

- ◆ 當 X20 =On 時，D0 的 16 個位中，內容為“1”的位總數被存於 D2 當中。



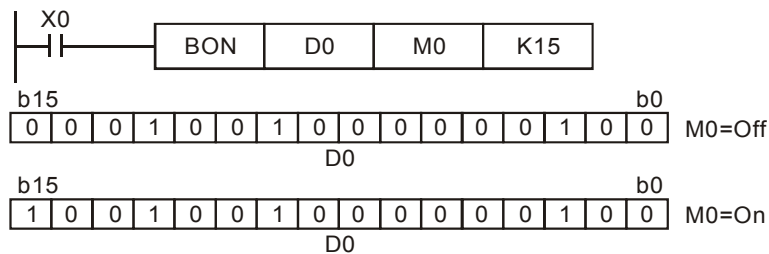
API																	适用机种		
<b>44</b>	<b>D</b>	<b>BON</b>	<b>P</b>			(S) (D)											10PM		
						(n)											✓		
	位装置				字符装置												16 位指令 (7 STEP)		
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	BON	连续执行型	BON P	脉冲执行型
S					*	*	*	*	*	*	*	*	*	*	*	32 位指令 (13 STEP)			
D	*	*	*													DBON	连续执行型	DBON P	脉冲执行型
n					*	*						*	*	*	*	• 标志信号：无			
<ul style="list-style-type: none"> <li>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)</li> </ul>																			

### 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)。

### 程序范例

- ◆ 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。



# 5 应用指令分类及基本使用

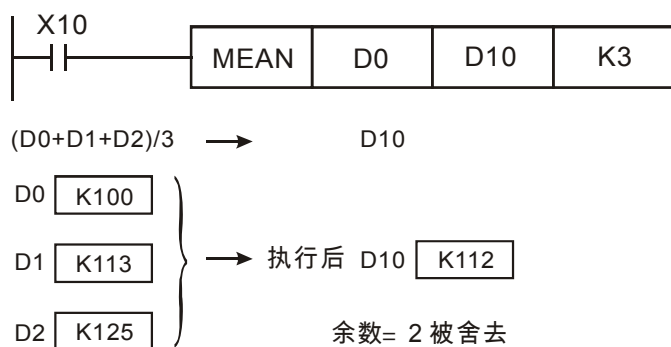
API																	适用機種	
45	D	MEAN	P	S	D	n	平均值										10PM	
																	✓	
	位装置				字符装置												16 位指令 (7 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	MEAN 连续执行型 MEAN P 脉冲执行型		
S					*	*	*	*	*	*	*	*	*			32 位指令 (13 STEP)		
D							*	*	*	*	*	*	*	*	*	DMEAN 连续执行型 DMEAN P 脉冲执行型		
n					*	*	*	*	*	*	*	*	*	*	*	• 标志信号：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 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>																		

## 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。

## 程序范例

- ◆ 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 以求得平均值并存储于指定的 D10 当中，余数被舍去。



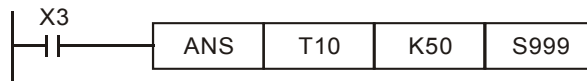
API																		适用機種	
<b>46</b>		<b>ANS</b>		<b>P</b>		<b>S</b>	<b>m</b>											10PM	
						<b>D</b>												✓	
		位装置				字符装置												16 位指令 (7 STEP)	
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		ANS 连续执行型    ANS P 脉冲执行型	
S												*							
m						*													
D					*														
<ul style="list-style-type: none"> <li>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</li> </ul>																		32 位指令 - - - - - <ul style="list-style-type: none"> <li>标志信号：无</li> </ul>	

**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

**程序范例**

- ◆ 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 会复位成 Off、当前值=0)。



# 5 应用指令分类及基本使用

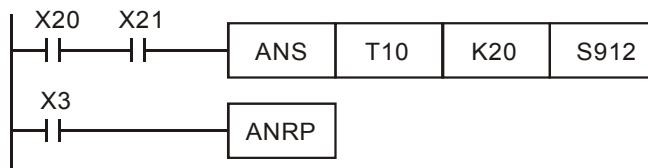
API																			适用機種	
47		ANR		P															10PM	
																			✓	
		位装置				字符装置												16 位指令 (1 STEP)		
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ANR	连续执行型	ANR P	脉冲执行型
		<ul style="list-style-type: none"> <li>操作数使用注意：无操作数 不须接点驱动的指令</li> </ul>																32 位指令		
															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

指令说明

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

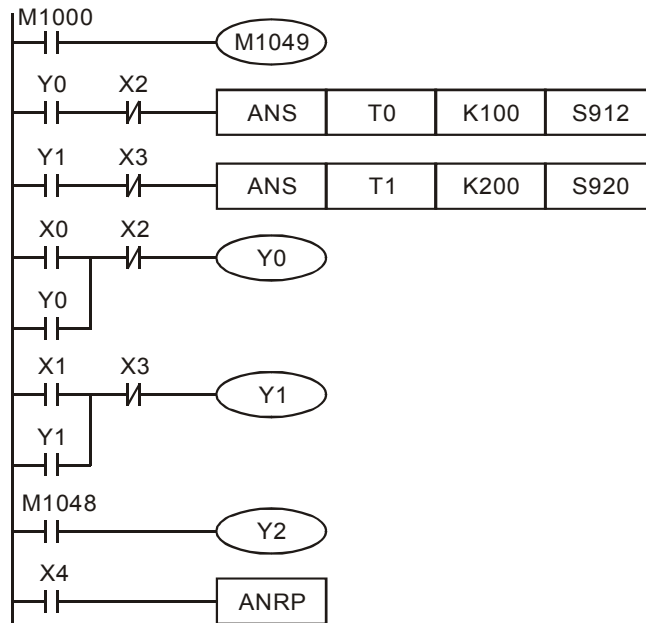
程序范例

- ◆ 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 会被复位成 Off，当前值为 0。
- ◆ X20 与 X21 同时 On 未满足 2 秒钟时，T10 的当前值被复位成 0。
- ◆ X3 从 Off → On 时，动作中的警报点 S912~S1023 被复位。
- ◆ X3 再次从 Off → On 时，次小号码警报点被复位。



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=警报点复位按钮
  - Y0=前进                            Y1=前进
  - Y2=警报显示器
  - 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, 警报显示器 Y2=On。
6. 当警报点的复位按钮 X4 每 On 一次, 动作中的警报点号码就被复位一个, 复位的顺序从较小的号码开始。

# 5 应用指令分类及基本使用

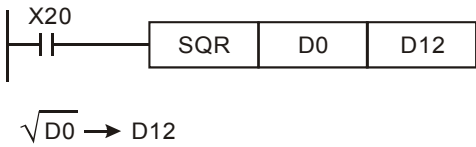
API																		适用機種
48	D	SQR	P	(S)	(D)	开平方根											10PM	
																		✓
	位装置				字符装置												16 位指令 (7 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	SQR 连续执行型 SQR P 脉冲执行型		
S											*					32 位指令		
m					*											DSQR 连续执行型 DSQR P 脉冲执行型		
D				*												• 标志信号： Ox O100 M1808 M1968 零标志信号 Zero flag M1809 M1969 借位标志 Borrow flag M1810 M1970 进位标志 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，借位标志信号 **On**。
- ◆ 运算结果 **D** 为 0 时，零标志信号 **On**。

### 程序范例

- ◆ 当 **X20=On**，将 **D0** 内容值开平方根后，存放于 **D12**。





API																适用機種	
<b>49</b>	<b>D</b>	<b>FLT</b>	<b>P</b>		<b>(S)</b>	<b>(D)</b>	<b>BIN 整数→2 进小数点转换</b>									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					*	*							*			-	
D													*			-	
<p>• 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 <b>DFLT</b>、<b>DFLTP</b> 有效</p>																<p>32 位指令 (6 STEP)</p> <p><b>DFLT</b> 连续执行型    <b>DFLTP</b> 脉冲执行型</p> <p>• 标志信号：</p> <p>Ox    O100</p> <p>M1808    M1968    零标志信号 Zero flag</p> <p>M1809    M1969    借位标志 Borrow flag</p> <p>M1810    M1970    进位标志 Carry flag</p> <p>请参考下列指令说明</p>	

### 指令说明

- ◆ **S**：转换来源装置。 **D**：存放转换结果之装置。
- ◆ 将 BIN 整数转换成 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，则进位标志=On。
  2. 若转换结果的绝对值小于可表示之 最小浮点值，则借位标志=On。
  3. 若转换结果为 0，则零标志=On。

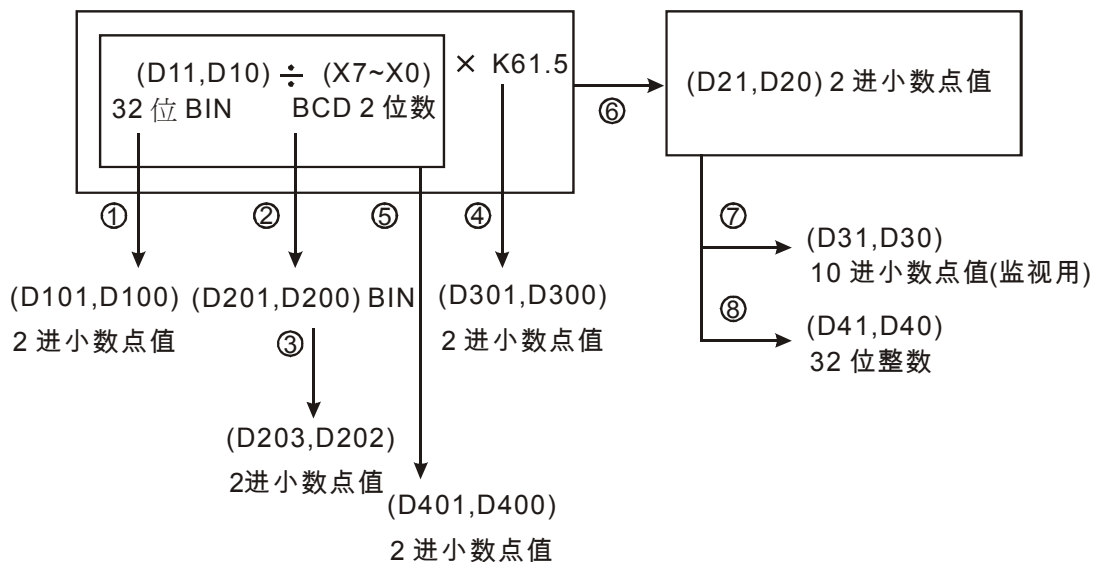
### 程序范例 (一)

- ◆ 当 X11=On 时，将 D1、D0 (内为 BIN 整数) 转换成 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。

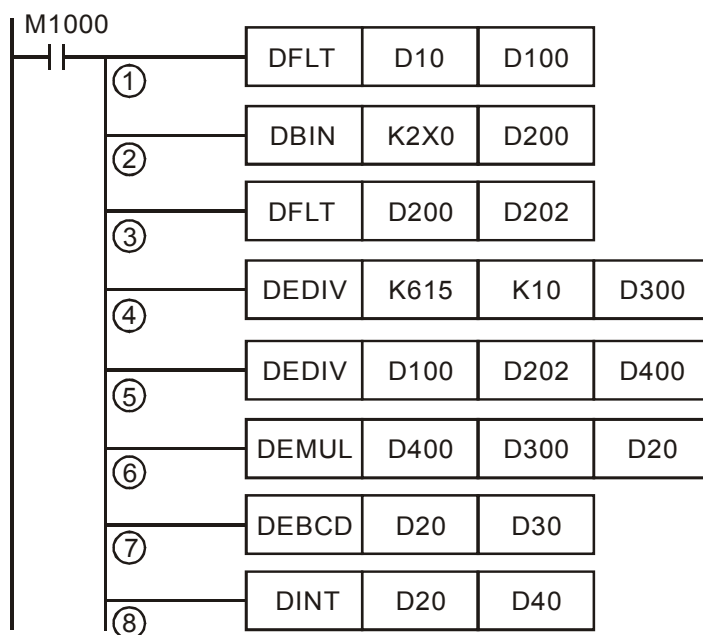


### 程序范例 (二)

- ◆ 使用应用指令来完成下列的算式。



## 5 应用指令分类及基本使用



1. 将 D11、D10 (内为 BIN 整数) 转换成 D101、D100 (2 进小数点值)。
2. 将 X7~X0 (BCD 值) 转换成 D201、D200 (BIN 值)。
3. 将 D201、D200 (内为 BIN 整数) 转换成 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 转换成 10 进小数点值 D31、D30。
8. 2 进小数点值 D21、D20 转换成 BIN 整数 D41、D40。

API																		适用機種
<b>50</b>		<b>REF</b>		<b>P</b>	<b>(D)</b>	<b>(n)</b>	<b>I/O 更新处理</b>										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	*	*														32 位指令		
n					*	*										-		
<ul style="list-style-type: none"> <li>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</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>		

### 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。数字扩展机的 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：
  - D** 指定 **X0** 且 **n** ≤ 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。
  - 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。
  - 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~ 主机的 I/O 点，且为 4 之倍数。

### 程序范例 (一)

- ◆ 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。



### 程序范例 (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。



## 5 应用指令分类及基本使用

---

程序范例  
(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
<b>61</b>	<b>D</b>	<b>SER</b>	<b>P</b>	<b>S1</b>	<b>S2</b>	<b>D</b>	<b>n</b>	多点比较								10PM
																✓

	位装置					字符装置											
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S1							*	*	*	*	*	*	*			16 位指令 (9 STEP) SER 连续执行型 SER P 脉冲执行型	
S2					*	*	*	*	*	*	*	*	*	*	*	32 位指令 (17 STEP) DSER 连续执行型 DSERP 脉冲执行型	
D								*	*	*	*	*	*			• 标志信号：无	
N					*	*							*				

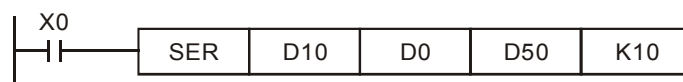
• 操作数使用注意：此指令有支持 V、Z 装置 (当 16 位指令时不能使用 Z 装置；当 32 位指令时不能使用 V 装置)  
各装置使用范围请参考功能规格表  
指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号。如：K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)。

### 指令说明

- ◆ **S<sub>1</sub>**：多点比较的数据区块的起始装置。**S<sub>2</sub>**：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。
- ◆ **S<sub>1</sub>** 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 **S<sub>2</sub>** 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, **S<sub>1</sub>**, **S<sub>2</sub>**, **D**, **n** 会指定 32 位寄存器。
- ◆ **n** 操作数范围: n=1~256 (16 位指令); n=1~128 (32 位指令)。

### 程序范例

- ◆ 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。
- ◆ 大小比较以代数类型进行。(-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。



# 5 应用指令分类及基本使用

<b>S<sub>1</sub></b>	内容值	比较数据	数据编号	比较结果	<b>D</b>	内容值	说明
D10	88	<b>S<sub>2</sub></b>  D0=K100	0		D50	4	相等值的数据个数
D11	100		1	相等	D51	1	第一个相等值的编号
D12	110		2		D52	8	最后一个相等值的编号
D13	150		3		D53	7	最小值的编号
D14	100		4	相等	D54	9	最大值的编号
D15	300		5				
D16	100		6	相等			
D17	5		7	最小			
D18	100		8	相等			
D19	500		9	最大			

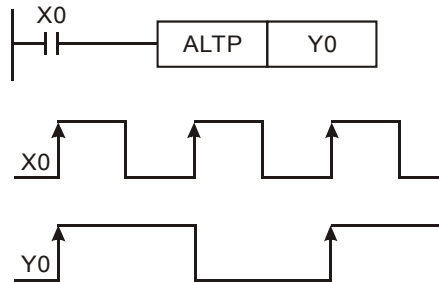
API																	适用機種	
<b>66</b>		<b>ALT</b>		<b>P</b>		<b>D</b>											10PM	
																	✓	
	位裝置				字符裝置												16 位指令 (9 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	ALT 连续执行型 ALT P 脉冲执行型		
D		*	*	*													32 位指令 (17 STEP)	
<ul style="list-style-type: none"> <li>操作数使用注意：所有装置不支持 V、Z 修饰</li> <li>各装置使用范围请参考功能规格表</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>		

### 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)。

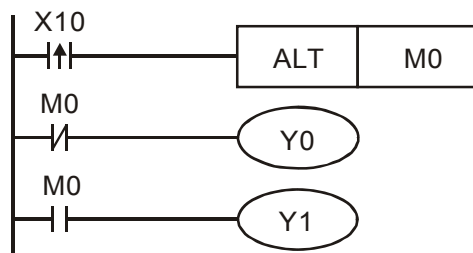
### 程序范例 (一)

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



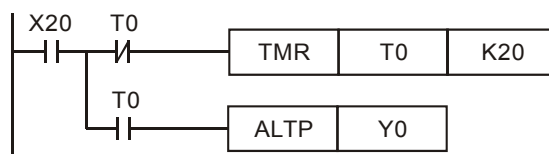
### 程序范例 (二)

- ◆ 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。



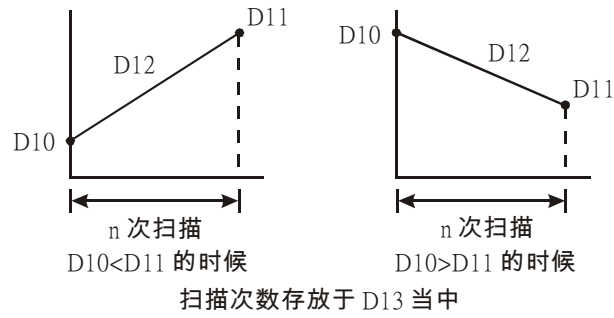
### 程序范例 (三)

- ◆ 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。



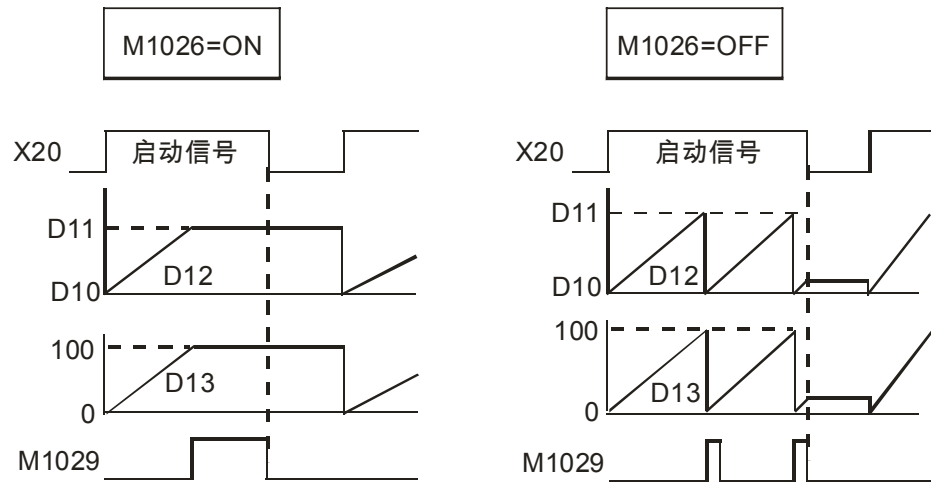






补充说明

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



# 5 应用指令分类及基本使用

API																		适用機種		
69	D	SORT		(S)	(m1)	(m2)	(D)	(n)	数据排序									10PM		
✓																				
	位装置				字符装置												16 位指令 (9 STEP) SORT 连续执行型 - <hr/> 32 位指令 (17 STEP) DSORT 连续执行型 - • 标志信号：无			
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z					
S													*							
m1					*	*														
m2					*	*														
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																				

## 指令说明

- ◆ **S**：原始数据区块之起始装置。**m<sub>1</sub>**：被排序之数据组数。**(m<sub>1</sub> = 1~32)**。**m<sub>2</sub>**：每笔数据之字段数。**(m<sub>2</sub> = 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~ m<sub>2</sub>)**，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 **m<sub>1</sub> × m<sub>2</sub>** 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 **m<sub>1</sub>** 次的扫描时间之后才被排序完成，排序完成时执行完毕标志信号 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。

## 程序范例



### 1. 排序数据构成例

		← 数据数: m2 个 →				
		数据域位				
		1	2	3	4	5
		学生编号	国文	英文	数学	理化
↑ 数据 个 数 :	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

## 2. 当 D100=K3 时的排序后数据

		数据数: 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
↑ 数据 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

		数据数: 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
↑ 数据 个数 : :	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 应用指令分类及基本使用

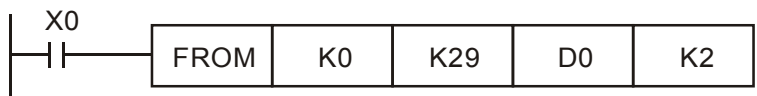
API																		适用机种
<b>78</b>	<b>D</b>	<b>FROM</b>	<b>P</b>	<b>(m<sub>1</sub>)</b>	<b>(m<sub>2</sub>)</b>	<b>(D)</b>	<b>(n)</b>	特殊模块 CR 数据读出										10PM
																		✓
	位装置				字符装置												16 位指令 (9 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	FROM 连续执行型 FROMP 脉冲执行型		
m <sub>1</sub>					*	*					*	*	*	*	*			
m <sub>2</sub>					*	*					*	*	*	*	*			
D											*	*	*	*	*			
n					*	*					*	*	*	*	*			
<ul style="list-style-type: none"> <li>操作数使用注意：m<sub>1</sub> 操作数使用范围(16 及 32 位指令)：0~255 m<sub>2</sub> 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- m<sub>2</sub>) (32 位指令)：1~(500- m<sub>2</sub>)/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)</li> </ul>																		
<ul style="list-style-type: none"> <li>请参考下列补充说明</li> </ul>																		

### 指令说明

- ◆ m<sub>1</sub>：特殊模块所在之编号。m<sub>2</sub>：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 系列机种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。

### 程序范例

- ◆ 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，指令不被执行，之前读出的数据其内容没有变化。



API																适用机种
<b>79</b>	<b>D</b>	<b>TO</b>	<b>P</b>	<b>m<sub>1</sub></b>	<b>m<sub>2</sub></b>	<b>S</b>	<b>n</b>	特殊模块 CR 数据写入								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 <sub>1</sub>					*	*						*	*	*	*	*				
m <sub>2</sub>					*	*						*	*	*	*	*				
S					*	*						*	*	*	*	*				
n					*	*						*	*	*	*	*				

• 操作数使用注意：m<sub>1</sub> 操作数使用范围(16 及 32 位指令)：0~255  
 m<sub>2</sub> 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- m<sub>2</sub>)  
 (32 位指令)：1~(500- m<sub>2</sub>)/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)

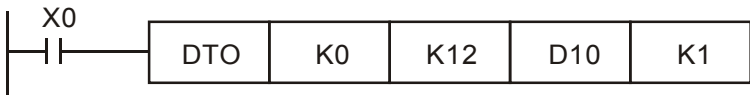
• 请参考下列补充说明

### 指令说明

- ◆ m<sub>1</sub>：特殊模块所在之编号。m<sub>2</sub>：欲写入特殊模块之 CR (Controlled Register) 编号。S：写入 CR 的数据。n：一次写入之数据笔数。
- ◆ DVP-PM 系列机种利用此指令将数据写入特殊模块之 CR 内。

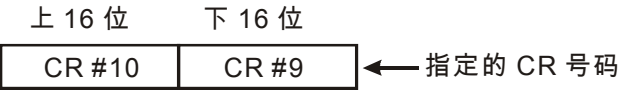
### 程序范例

- ◆ 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。



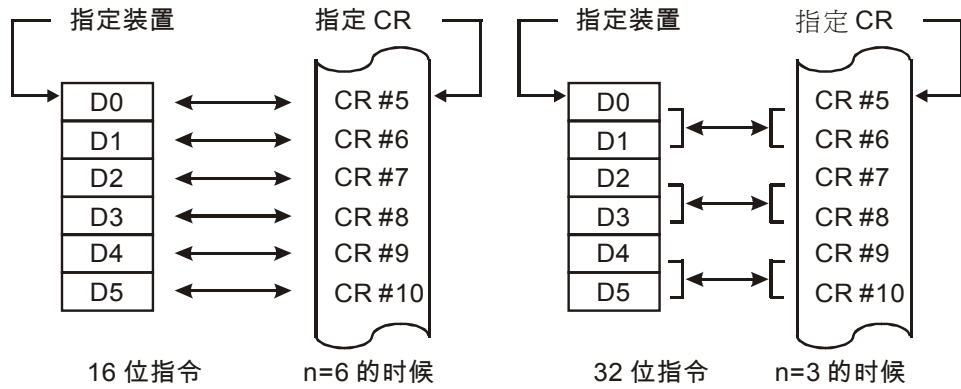
### 补充说明

- ◆ 指令操作数的规则
  1. m<sub>1</sub>：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. m<sub>2</sub>：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。

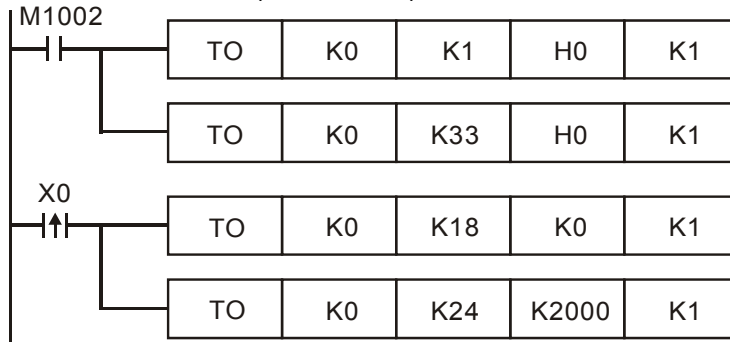
# 5 应用指令分类及基本使用



◆ FROM / TO 指令应用范例说明：

程序范例  
(一)

◆ 调整 DVP04AD-H2 之 A/D 转换特性曲线，将 CH1 之 Offset 值设为 0V (=K0<sub>LSB</sub>)，GAIN 值设为 2.5V (=K2,000<sub>LSB</sub>)。



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<sub>LSB</sub> 写入 CR#18 内。将 GAIN 值 K2,000<sub>LSB</sub> 写入 CR#24 内。

程序范例  
(二)

◆ 调整 DVP04AD-H2 之 A/D 转换特性曲线，将 CH2 之 Offset 值设为 2mA (=K400<sub>LSB</sub>)，GAIN 值设为 18 mA (=K3,600<sub>LSB</sub>)。

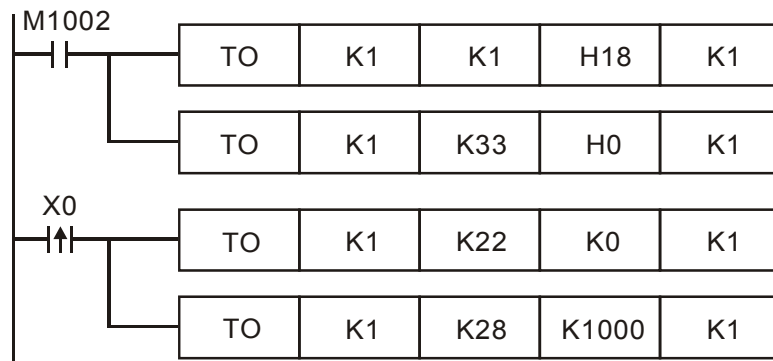


1. 对模拟信号输入模块编号 0 之 CR#1 写入 H18，CH2 设为模式 3 (电流输入 -20mA~+20mA)。
2. CR#33 写入 H0，允许 CH1 ~ CH4 都可特性微调。

- 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。

程序范例  
(三)

- ◆ 调整 DVP02DA-H2 之 D/A 转换特性曲线，将 CH2 之 Offset 值设为 0mA (=K0<sub>LSB</sub>)，GAIN 值设为 10mA (=K1,000<sub>LSB</sub>)。



- 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)。
- CR#33 写入 H0，允许 CH1、CH2 特性微调。
- 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。

程序范例  
(四)

- ◆ 调整 DVP02DA-H2 之 D/A 转换特性曲线，将 CH2 之 Offset 值设为 2mA (=K400<sub>LSB</sub>)，GAIN 值设为 18mA (=K3,600<sub>LSB</sub>)。



- 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)。
- CR#33 写入 H0，允许 CH1、CH2 特性微调。
- 当 X0=Off→On 时，将 Offset 值  $K400_{LSB}$  写入 CR#
- 23 内，将 GAIN 值  $K3,600_{LSB}$  写入 CR#29 内。

# 5 应用指令分类及基本使用

API																		适用機種			
87	D	ABS	P	Ⓚ	绝对值												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
																		• 标志信号：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)。

程序范例

- ◆ 当 X0 从 Off→On, D0 内容取绝对值。



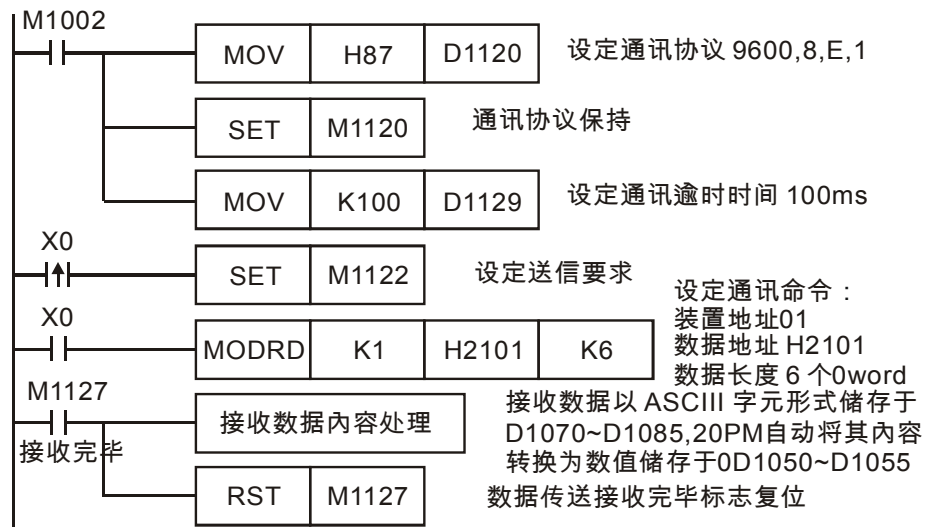


API																	适用機種
100		MODRD			(S <sub>1</sub> )	(S <sub>2</sub> )	(n)										10PM
																	✓
		位装置				字符装置											16 位指令 (7 STEP) MODRD 连续执行型 - - <hr/> 32 位指令 - - - - • 标志信号： M1120~M1129、M1140~M1143 请参考下列补充说明
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	
		S <sub>1</sub>				*	*							*			
		S <sub>2</sub>				*	*							*			
		n				*	*							*			
• 操作数使用注意：S <sub>1</sub> 操作数范围 K0~K254 n 操作数范围 K1 ≤ n ≤ K6 各装置使用范围请参考功能规格表																	

### 指令说明

- ◆ S<sub>1</sub>：联机装置地址。S<sub>2</sub>：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<sub>2</sub> 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信息，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数据转为数值转存于 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标志 M1140、M1141 会被清除。
- ◆ DVP-PM 与 VFD-B 系列变频器联机 (ASCII Mode、M1143=Off)

### 程序范例 (一)



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”

## 5 应用指令分类及基本使用

DVP-PM 传送数据寄存器 (传送信息)

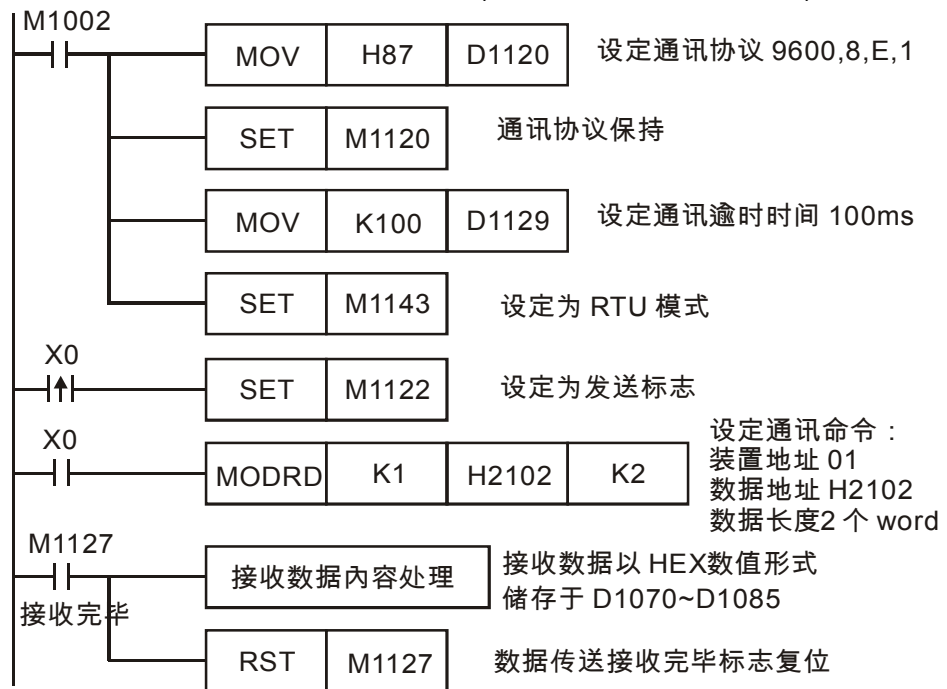
寄存器	DATA		说明	
D1089 下	'0'	30 H	ADR 1	ADR (1,0) 为变频器地址
D1089 上	'1'	31 H	ADR 0	
D1090 下	'0'	30 H	CMD 1	CMD (1,0) 为命令码
D1090 上	'3'	33 H	CMD 0	
D1091 下	'2'	32 H	起始数据地址 Starting Data Address	
D1091 上	'1'	31 H		
D1092 下	'0'	30 H		
D1092 上	'1'	31 H		
D1093 下	'0'	30 H	数据 (word) 个数 Number of Data (count by word)	
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 (响应信息)

寄存器	DATA		说明	
D1070 下	'0'	30 H	ADR 1	
D1070 上	'1'	31 H		
D1071 下	'0'	30 H	CMD 1	
D1071 上	'3'	33 H		
D1072 下	'0'	30 H	数据 (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	
D1085 上	'B'	42 H		

程序范例  
(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 (传送信息)

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	数据 (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

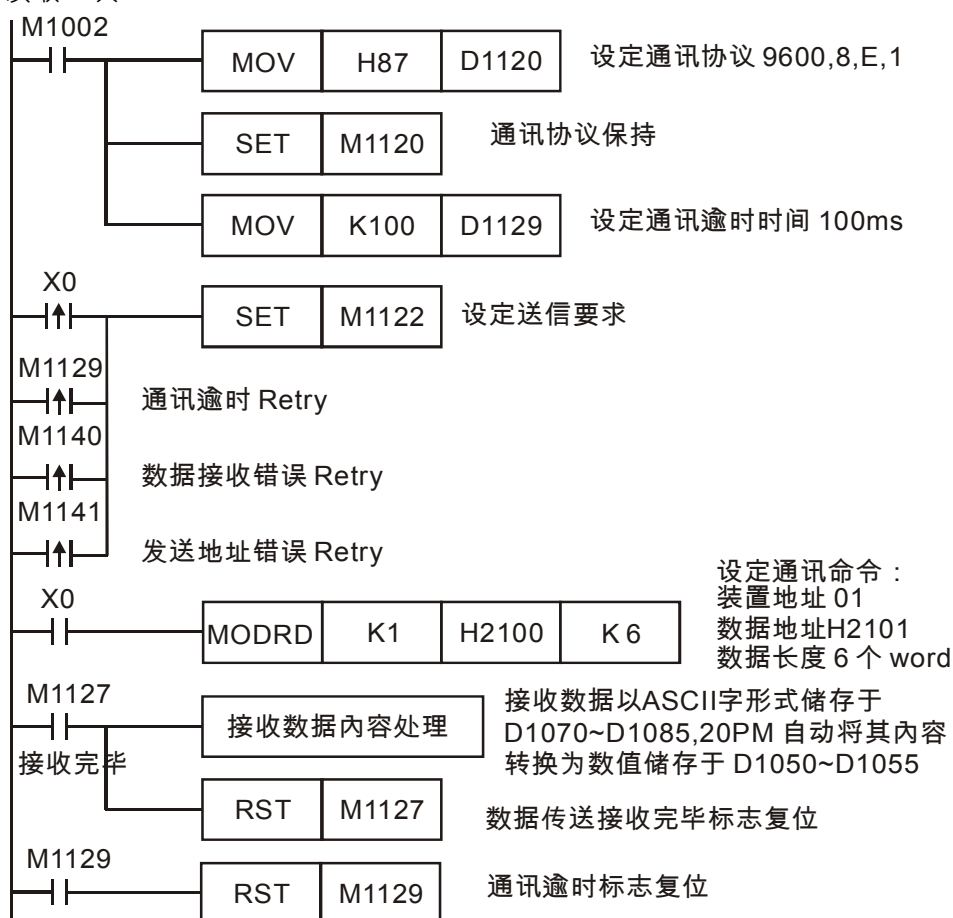
DVP-PM 接收数据寄存器 (响应信息)

寄存器	DATA	说明
D1070 下	01 H	Address
D1071 下	03 H	Function
D1072 下	04 H	数据 (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

# 5 应用指令分类及基本使用

## 程序范例 (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。
- ◆ 若通讯超时则 M1129 标志为 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 标志为 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 标志为 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沿 (LDP · ANDP · ORP) 及接点下降沿 (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。

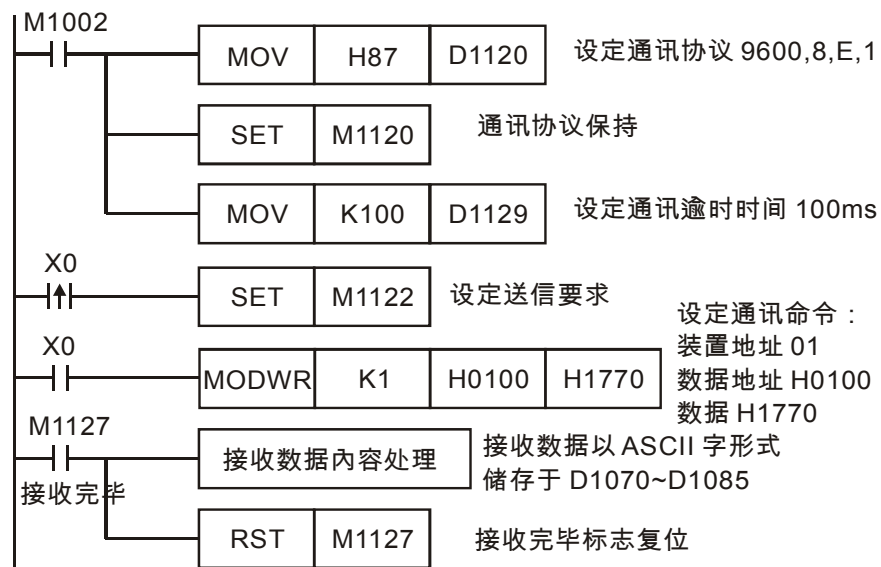
API																	适用机种
<b>101</b>		<b>MODWR</b>				<b>(S<sub>1</sub>) (S<sub>2</sub>) (n)</b>											10PM
																	✓
	位装置				字符装置												16 位指令 (7 STEP) MODWR 连续执行型 - - <hr/> 32 位指令 - - - • 标志信号： M1120~M1129、M1140~M1143 请参考下列补充说明
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S <sub>1</sub>					*	*							*				
S <sub>2</sub>					*	*							*				
n					*	*							*				
• 操作数使用注意：S <sub>1</sub> 操作数范围 K0~K254 各装置使用范围请参考功能规格表																	

### 指令说明

- ◆ **S<sub>1</sub>**：联机装置地址。**S<sub>2</sub>**：欲写入数据的地址。**n**：欲写入的数据。
- ◆ 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)。
- ◆ **S<sub>2</sub>** 欲写入数据的地址。若地址对于被指定的装置不合法，则会响应错误信息，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标志 M1140、M1141 会被清除。

### 程序范例 (一)

- ◆ DVP-PM 与 VFD-B 系列变频器联机 (ASCII Mode、M1143=Off)



## 5 应用指令分类及基本使用

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 (传送信息)

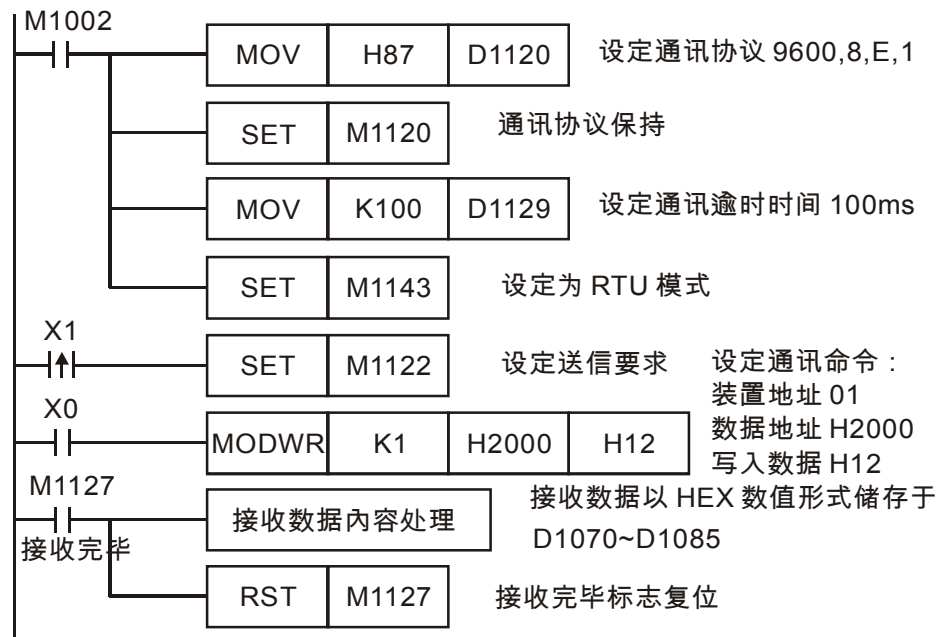
寄存器	DATA		说明	
D1089 下	'0'	30 H	ADR 1	ADR (1,0) 为变频器地址
D1089 上	'1'	31 H	ADR 0	
D1090 下	'0'	30 H	CMD 1	CMD (1,0) 为命令码
D1090 上	'6'	36 H	CMD 0	
D1091 下	'0'	30 H	数据地址 Data Address	
D1091 上	'1'	31 H		
D1092 下	'0'	30 H		
D1092 上	'0'	30 H		
D1093 下	'1'	31 H	数据内容 Data contents	
D1093 上	'7'	37 H		
D1094 下	'7'	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 (响应信息)

寄存器	DATA		说明	
D1070 下	'0'	30 H	ADR 1	ADR (1,0) 为变频器地址
D1070 上	'1'	31 H	ADR 0	
D1071 下	'0'	30 H	CMD 1	CMD (1,0) 为命令码
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 (0,1) 为错误检查码
D1076 上	'1'	31 H	LRC CHK 0	

程序范例  
(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 (传送信息)

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

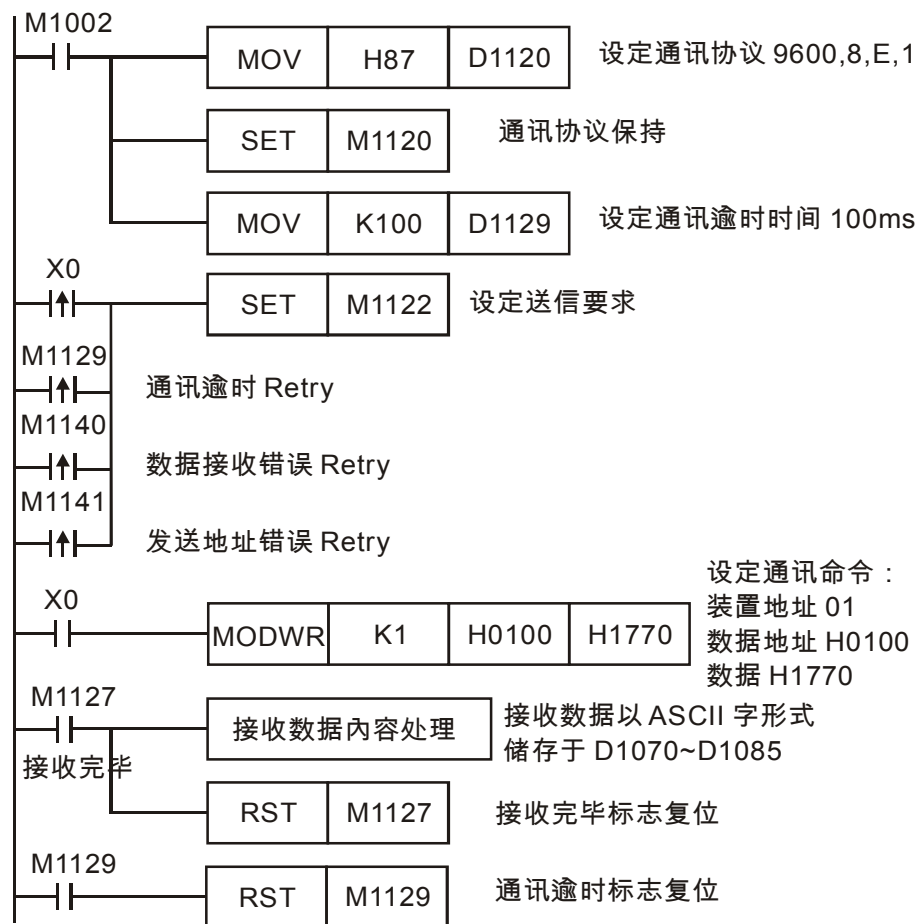
DVP-PM 接收数据寄存器 (响应信息)

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

## 5 应用指令分类及基本使用

### 程序范例 (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 标志为 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 标志为 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 标志为 On · 程序中由 M1141 触发送信要求 M1122 再写入一次。

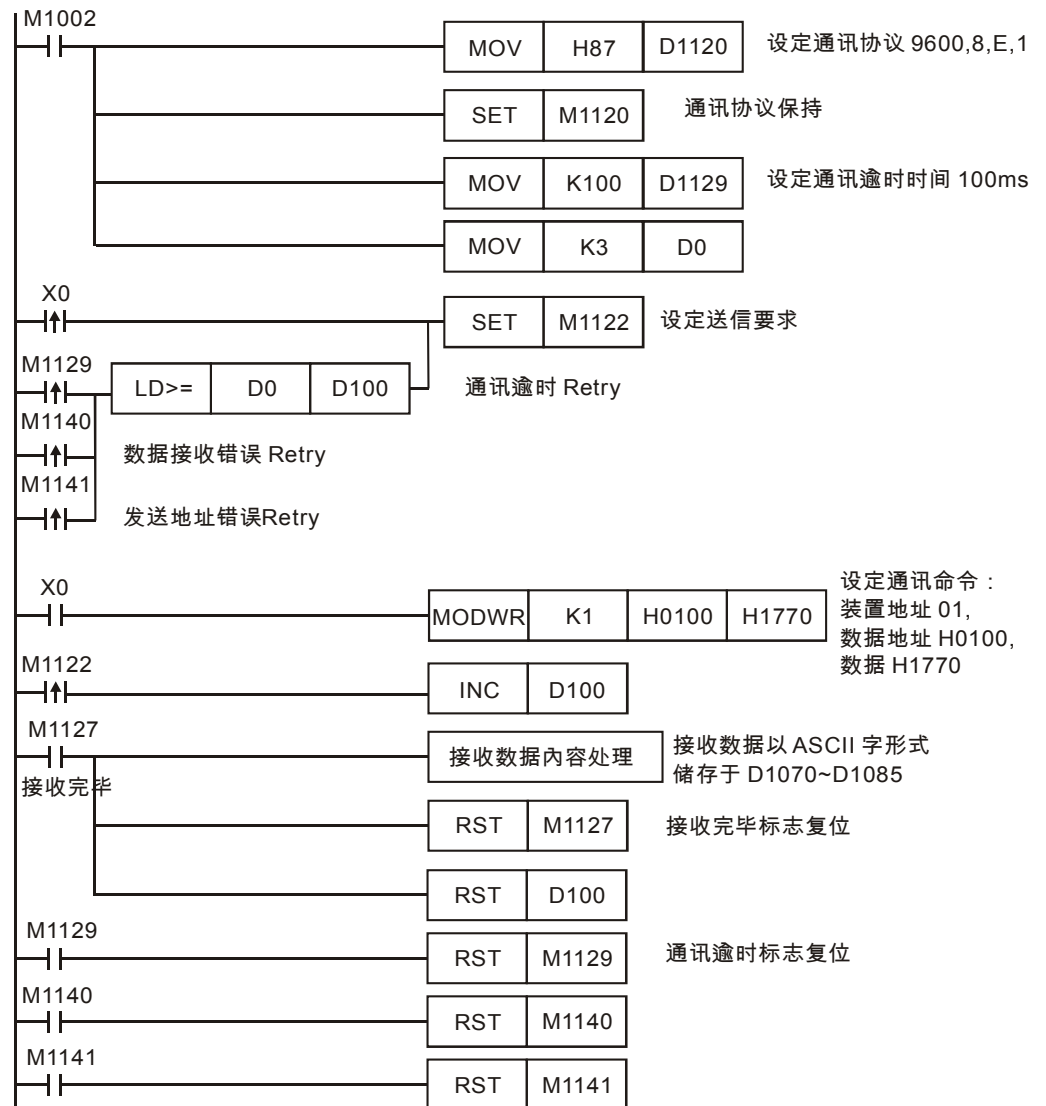


### 程序范例 (四)

- ◆ 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 将数据 H1770 (K6,000) 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 标志为 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 标志为 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 标志为 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沿 (LDP、ANDP、ORP) 及接点下降沿 (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。

# 5 应用指令分类及基本使用

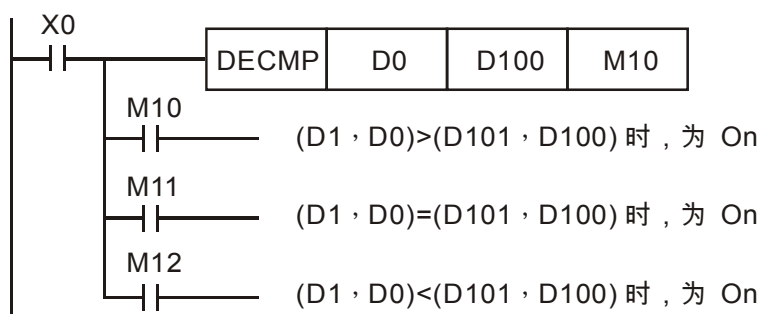
API <b>110</b>	<b>D</b>	<b>ECMP</b>	<b>P</b>	<b>(S1)</b> <b>(S2)</b> <b>(D)</b>	二进制浮点数比较										适用機種 10PM ✓	
位装置				字符装置												16 位指令
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-
S1					*								*			-
S2					*								*			-
D		*	*	*												-
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>本指令只有 32 位指令 <b>DECMP</b>、<b>DECMP</b> 有效</li> <li>D 操作数会占用连续 3 点</li> <li>F 表示浮点数输入，使用时要有小数点</li> </ul>																32 位指令 (9 STEP) <b>DECMP</b> 连续执行型 <b>DECMP</b> 脉冲执行型 • 标志信号：无

## 指令说明

- ◆ **S<sub>1</sub>**：二进制浮点数比较值 1。**S<sub>2</sub>**：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。
- ◆ **S<sub>1</sub>** 或 **S<sub>2</sub>** 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。

## 程序范例

- ◆ 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。

API																适用机种
<b>111</b>	<b>D</b>	<b>EZCP</b>	<b>P</b>	(S <sub>1</sub> )	(S <sub>2</sub> )	(S)	(D)	二进制浮点数区域比较								10PM
																✓

	位装置				字符装置											
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	
S <sub>1</sub>					*									*		
S <sub>2</sub>					*									*		
S					*									*		
D		*	*	*												

• 操作数使用注意：D 操作数会占用连续 3 点  
 S<sub>1</sub> 操作数内容值请小于 S<sub>2</sub> 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

16 位指令  
 32 位指令 (12 STEP)  
**DEZCP** 连续执行型    **DEZCPP** 脉冲执行型  
 • 标志信号：无

### 指令说明

- ◆ S<sub>1</sub>：区域比较之二进制浮点数下限值。S<sub>2</sub>：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 S<sub>1</sub> 及二进制浮点数上限值 S<sub>2</sub> 作比较，其比较结果在 D 作表示。
- ◆ S<sub>1</sub> 或 S<sub>2</sub> 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 S<sub>1</sub> 大于二进制浮点数上限值 S<sub>2</sub> 时，则指令以二进制浮点数下限值 S<sub>1</sub> 作为上下限值进行比较。

### 程序范例

- ◆ 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。

# 5 应用指令分类及基本使用

API																		适用機種
<b>112</b>	D	<b>MOVR</b>					(S)	(D)	浮点数值数据传送									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	-		
S																		
D							*	*	*	*	*	*				32 位指令		
<ul style="list-style-type: none"> <li>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)</li> </ul>																DMOVR 连续执行型 DMOVR P 脉冲执行型 <ul style="list-style-type: none"> <li>标志信号：无</li> </ul>		

### 指令说明

- ◆ **S**：浮点数值数据来源。 **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。

### 程序范例

- ◆ 当 X0= Off 时, D10, D11 内容没有变化。当 X0=On 时, 将 F1.2 浮点数当前值传送至 D10, D11 数据寄存器内。



API																	适用机种	
<b>116</b>	<b>D</b>	<b>RAD</b>	<b>P</b>				(S)	(D)									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 DRAD、DRADP 有效</p>																<p>32 位指令 (6 STEP)</p> <p>DRAD 连续执行型 DRADP 脉冲执行型</p>		
<p>标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1809 M1969 借位标志 Borrow flag</p> <p>M1810 M1970 进位标志 Carry flag</p> <p>请参考下列补充说明</p>																		

### 指令说明

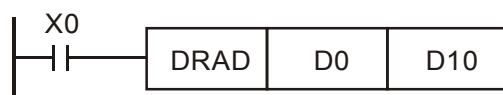
- ◆ S：数据来源（角度）。D：转换之结果（弧度）。
- ◆ 使用下列公式将角度转换成弧度。

$$\text{弧度} = \text{角度} \times (\pi/180)$$

- ◆ 若转换结果的绝对值大于可表示之最大浮点值，则进位标志=On。
- ◆ 若转换结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若转换结果为 0，则零标志=On。

### 程序范例

- ◆ 当 X0=On 时，指定二进制浮点数 (D1·D0) 之角度值，将角度转换成弧度值后存于 (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。

# 5 应用指令分类及基本使用

API																		适用機種
<b>117</b>	<b>D</b>	<b>DEG</b>	<b>P</b>	<b>S</b>	<b>D</b>	弧度→角度											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													*			32 位指令 (6 STEP)		
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表。</li> <li>F 表示浮点数输入，使用时要有小数点</li> <li>本指令只有 32 位指令 <b>DDEG</b>、<b>DDEGP</b> 有效</li> </ul>																		
<ul style="list-style-type: none"> <li>标志信号：</li> <li>Ox O100</li> <li>M1808 M1968 零标志 Zero flag</li> <li>M1809 M1969 借位标志 Borrow flag</li> <li>M1810 M1970 进位标志 Carry flag</li> </ul> <p>请参考下列补充说明</p>																		

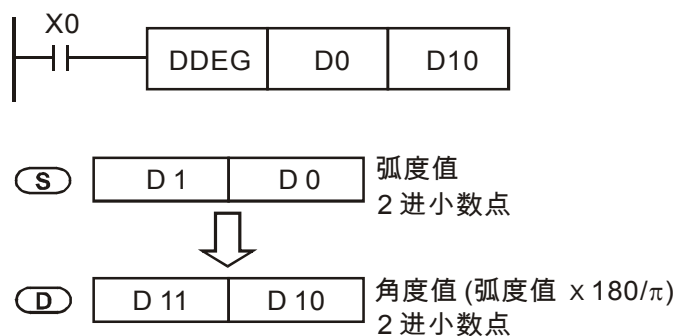
### 指令说明

- ◆ **S**：数据来源（弧度）。**D**：转换之结果（角度）。
  - ◆ 使用下列公式将弧度转换成角度。
- $$\text{角度} = \text{弧度} \times (180/\pi)$$

- ◆ 若转换结果的绝对值大于可表示之最大浮点值，则进位标志=On。
- ◆ 若转换结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若转换结果为 0，则零标志=On。

### 程序范例

- ◆ 当 X0=On 时，指定二进制浮点数 (D1 · D0) 之角度值，将弧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。

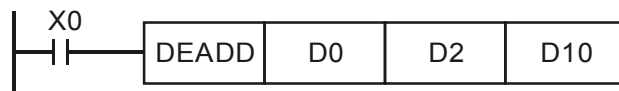
API					二进制浮点数加算										适用機種		
120	D	EADD	P	(S1)	(S2)	(D)											10PM
															✓		
	位装置				字符装置											16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S <sub>1</sub>					*								*			-	
S <sub>2</sub>					*								*			-	
D													*			-	
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>F 表示浮点数输入，使用时要有小数点</li> <li>本指令只有 32 位指令 DEADD、DEADDP 有效</li> </ul>															32 位指令 (9 STEP) DEADD 连续执行型 DEADDP 脉冲执行型 标志信号： Ox O100 M1808 M1968 零标志 Zero flag M1809 M1969 借位标志 Borrow flag M1810 M1970 进位标志 Carry flag 请参考下列补充说明		

### 指令说明

- ◆ S<sub>1</sub>：被加数。S<sub>2</sub>：加数。D：和。
- ◆ S<sub>1</sub> 所指定的寄存器内容加上 S<sub>2</sub> 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类型进行。
- ◆ S<sub>1</sub> 或 S<sub>2</sub> 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。
- ◆ S<sub>1</sub> 及 S<sub>2</sub> 可指定相同的寄存器编号，此种情况下若是使用“连续执行”类型的指令时，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，则进位标志=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若运算结果为 0，则零标志=On。

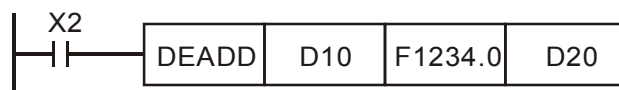
### 程序范例 (一)

- ◆ 当 X0=On 时，将二进制浮点数 (D11·D0) + 二进制浮点数 (D3·D2)，结果存放在 (D11·D10) 中。



### 程序范例 (二)

- ◆ 当 X2=On 时，将二进制浮点数 (D11·D10) + F1234.0 (自动转换为二进制浮点数)，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。

# 5 应用指令分类及基本使用

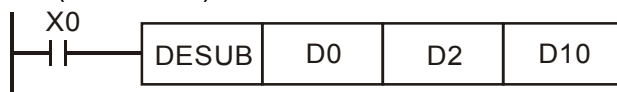
API														适用機種			
121	D	ESUB	P	$(S_1)$ $(S_2)$ $(D)$										二进制浮点数减算		10PM	
																	✓
	位装置				字符装置										16 位指令		
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S <sub>1</sub>					*								*			-	
S <sub>2</sub>					*								*			-	
D													*			-	
操作数使用注意：各装置使用范围请参考功能规格表 F 表示浮点数输入，使用时要有小数点 本指令只有 32 位指令 DEADD、DEADDP 有效													32 位指令 (9 STEP) DESUB 连续执行型 DESUBP 脉冲执行型				
标志信号： Ox O100 M1808 M1968 零标志 Zero flag M1809 M1969 借位标志 Borrow flag M1810 M1970 进位标志 Carry flag 请参考下列补充说明																	

## 指令说明

- ◆ S<sub>1</sub>：被减数。S<sub>2</sub>：减数。D：差。
- ◆ S<sub>1</sub> 所指定的寄存器内容减掉 S<sub>2</sub> 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类型进行。
- ◆ S<sub>1</sub> 或 S<sub>2</sub> 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。
- ◆ S<sub>1</sub> 及 S<sub>2</sub> 可指定相同的寄存器编号，此种情况下若是使用“连续执行”类型的指令时，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，则进位标志=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若运算结果为 0，则零标志=On。

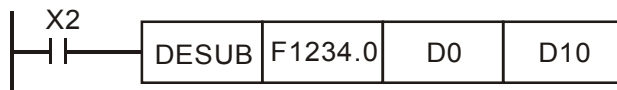
## 程序范例 (一)

- ◆ 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。



## 程序范例 (二)

- ◆ 当 X2=On 时，将 F1234.0 (自动转换为二进制浮点数) - 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	
122	D	EMUL	P	(S <sub>1</sub> )	(S <sub>2</sub> )	(D)	二进制浮点数乘算										10PM
✓																	

	位装置				字符装置											
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	
S <sub>1</sub>					*								*			16 位指令 - - - - 32 位指令 (9 STEP) DEMUL 连续执行型 DEMULP 脉冲执行型 标志信号： Ox O100 M1810 M1970 进位标志 Carry flag 请参考下列补充说明
S <sub>2</sub>					*								*			
D													*			

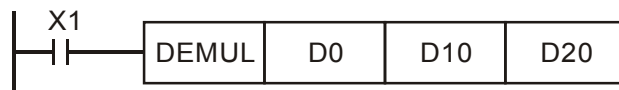
• 操作数使用注意：各装置使用范围请参考功能规格表  
 F 表示浮点数输入，使用时要有小数点  
 本指令只有 32 位指令 DEMUL、DEMULP 有效

### 指令说明

- ◆ S<sub>1</sub>：被乘数。S<sub>2</sub>：乘数。D：积。
- ◆ S<sub>1</sub> 所指定的寄存器内容乘上 S<sub>2</sub> 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类型进行。
- ◆ S<sub>1</sub> 或 S<sub>2</sub> 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。
- ◆ S<sub>1</sub> 及 S<sub>2</sub> 可指定相同的寄存器编号，此种情况下若是使用“连续执行”类型的指令时，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，则进位标志=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若运算结果为 0，则零标志=On。

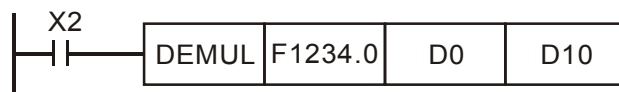
### 程序范例 (一)

- ◆ 当 X1=On 时，将二进制浮点数 (D11·D10) 乘上二进制浮点数 (D01·D00) 将积存放至 (D21·D20) 所指定的寄存器当中。



### 程序范例 (二)

- ◆ 当 X2=On 时，将 F1234.0 (自动转换为二进制浮点数) × 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。

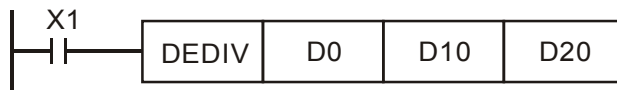
# 5 应用指令分类及基本使用

API <b>123</b>	<b>D</b>	<b>EDIV</b>	<b>P</b>	(S1) (S2) (D)	二进制浮点数除算										适用机种 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 <sub>1</sub>					*								*			-
S <sub>2</sub>					*								*			32 位指令 (9 STEP)
D													*			DEDIV 连续执行型 DEDIVP 脉冲执行型
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>F 表示浮点数输入，使用时要有小数点</li> <li>本指令只有 32 位指令 <b>DEDIV</b>、<b>DEDIVP</b> 有效</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：</li> <li>Ox O100</li> <li>M1808 M1968 零标志 Zero flag</li> <li>M1809 M1969 借位标志 Borrow flag</li> <li>M1810 M1970 进位标志 Carry flag</li> <li>M1793 M1953 运算错误标志 Operation error flag</li> </ul> <p>请参考下列补充说明</p>

## 指令说明

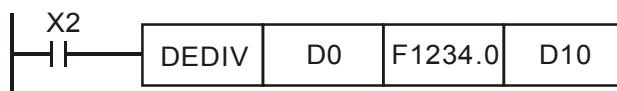
- ◆ S<sub>1</sub>：被除数。S<sub>2</sub>：除数。D：商及余数。
- ◆ S<sub>1</sub> 所指定的寄存器内容除以 S<sub>2</sub> 所指定的寄存器内容，商被存放至 D 所指定的寄存器当中，除算的动作全部以二进制浮点数类型进行。
- ◆ S<sub>1</sub> 或 S<sub>2</sub> 来源操作数若是指定浮点数 F 的话，指令会以该二进制浮点数值来作除算。
- ◆ 除数 S<sub>2</sub> 的内容若为 0 即被认定为“运算错误”，指令不执行，运算错误标志 ON，并记录错误码 H'0E19。
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位标志=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若运算结果为 0，则零标志=On。
- ◆ 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。

## 程序范例 (一)



## 程序范例 (二)

- ◆ 当 X2=On 时，将二进制浮点数 (D1·D0) ÷ F1234.0 (自动转换为二进制浮点数)，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。

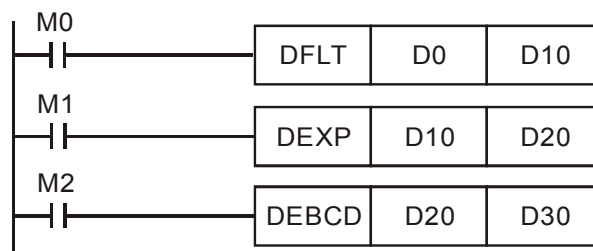
API																	适用机种	
<b>124</b>	<b>D</b>	<b>EXP</b>	<b>P</b>														10PM	
																	✓	
		位装置				字符装置										16 位指令		
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-	
S						*								*			-	
D														*			-	
<p>操作数使用注意：各装置使用范围请参考功能规格表</p> <p>F 表示浮点数输入，使用时要有小数点</p> <p>本指令只有 32 位指令 <b>DEXP</b>、<b>DEXPP</b> 有效</p>																		
<p>标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1809 M1969 借位标志 Borrow flag</p> <p>M1810 M1970 进位标志 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。
- ◆  $EXP^{[D+1 \cdot D]} = [S+1 \cdot S]$ ，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，则进位标志=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若运算结果为 0，则零标志=On。

### 程序范例

- ◆ 当 M0 为 On 时，将 (D11·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。

# 5 应用指令分类及基本使用

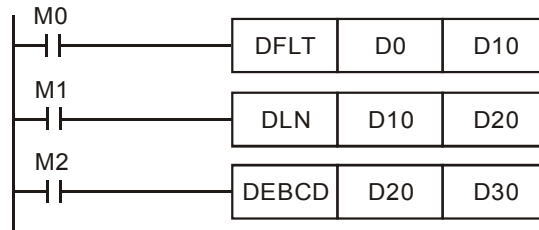
API																		适用機種		
125	D	LN	P		(S)	(D)	二进制浮点数取自然对数										10PM			
																		✓		
	位装置				字符装置												16 位指令			
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-				
S					*								*			-				
D													*			-				
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>F 表示浮点数输入，使用时要有小数点</li> <li>本指令只有 32 位指令 DLN、DLNP 有效</li> </ul>																32 位指令 (6 STEP) DLN 连续执行型 DLNP 脉冲执行型				
<ul style="list-style-type: none"> <li>标志信号：</li> <li>Ox O100</li> <li>M1808 M1968 零标志 Zero flag</li> <li>M1809 M1969 借位标志 Borrow flag</li> <li>M1810 M1970 进位标志 Carry flag</li> <li>M1793 M1953 运算错误标志 Operation error flag</li> </ul>																				
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。
- $$LN[S+1 \cdot S] = [D+1 \cdot D]$$
- ◆ 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，运算错误标志 ON，并记录错误码 H'0E19。
  - ◆  $e^D = S \rightarrow D$  操作数内容值 = lnS；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，则进位标志=On。
  - ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位标志=On。
  - ◆ 若运算结果为 0，则零标志=On。

## 程序范例

- ◆ 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。

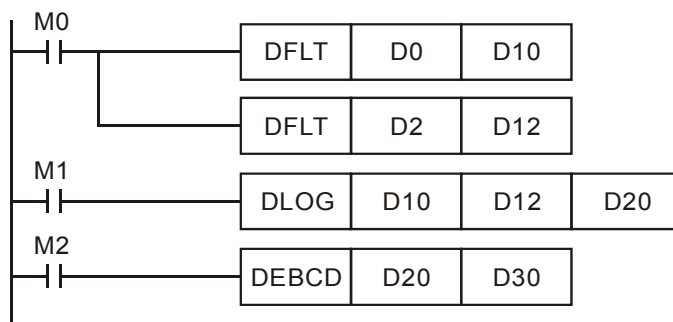
# 5 应用指令分类及基本使用

API																		适用机种
126	D	LOG	P		(S <sub>1</sub> )	(S <sub>2</sub> )	(D)	二进制浮点数取对数										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 <sub>1</sub>					*								*			-		
S <sub>2</sub>					*								*			-		
D													*			-		
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>F 表示浮点数输入，使用时要有小数点</li> <li>本指令只有 32 位指令 <b>DLOG</b>、<b>DLOGP</b> 有效</li> </ul>																32 位指令 (9 STEP) <b>DLOG</b> 连续执行型 <b>DLOGP</b> 脉冲执行型		
<ul style="list-style-type: none"> <li>标志信号：           <ul style="list-style-type: none"> <li>Ox O100</li> <li>M1808 M1968 零标志 Zero flag</li> <li>M1809 M1969 借位标志 Borrow flag</li> <li>M1810 M1970 进位标志 Carry flag</li> </ul> </li> </ul> 请参考下列补充说明																		

## 指令说明

- ◆ **S<sub>1</sub>**：运算底数装置。**S<sub>2</sub>**：运算来源装置。**D**：运算结果装置。
- ◆ 将 **S<sub>1</sub>** 内容及 **S<sub>2</sub>** 内容为操作数做 log 运算，结果存放于 **D**。
- ◆ **S<sub>1</sub>**、**S<sub>2</sub>** 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 **S<sub>1</sub>**、**S<sub>2</sub>** 需为浮点数值。
- ◆ **S<sub>1</sub><sup>D</sup> = S<sub>2</sub>**，求 D 值 →  $\text{Log}_{S_1} S_2 = D$
- ◆ 若运算结果的绝对值大于可表示之最大浮点值，则进位标志=On。
- ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若运算结果为 0，则零标志=On。
- ◆ 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)

## 程序范例



## 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。

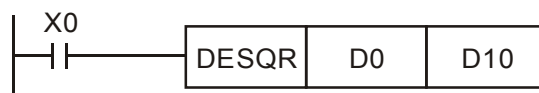
API																		适用机种
127	D	ESQR	P				(S)	(D)	二进制浮点数开平方根									10PM
																		✓
		位装置				字符装置											16 位指令	
		X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S						*								*			-	
D														*			-	
<p>操作数使用注意：各装置使用范围请参考功能规格表</p> <p>S 操作数范围须大于等于 0</p> <p>F 表示浮点数输入，使用时要有小数点</p> <p>本指令只有 32 位指令 DESQR、DESQRP 有效</p>																		
<p>32 位指令 (6 STEP)</p> <p>DESQLR 连续执行型 DESQRP 脉冲执行型</p> <p>标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1793 M1953 运算错误标志 Operation error flag</p> <p>请参考下列补充说明</p>																		

### 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类型进行。
- ◆ 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，零标志=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，运算错误标志 ON，并记录错误码 H'0E19。

### 程序范例 (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。



$$\sqrt{(D1, D0)} \rightarrow (D11, D10)$$

2 进小数点          2 进小数点

### 程序范例 (二)

- ◆ 当 X2=On 时，将 F1234.0 (自动转换为二进制浮点数) 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。

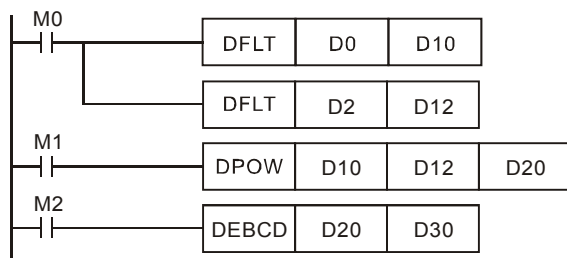
# 5 应用指令分类及基本使用

API																		适用機種		
128	D	POW	P		(S <sub>1</sub> )	(S <sub>2</sub> )	(D)											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 <sub>1</sub>					*								*			-				
S <sub>2</sub>					*								*			-				
D													*			-				
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>本指令只有 32 位指令 <b>DPOW</b>、<b>DPOWP</b> 有效</li> <li>F 表示浮点数输入，使用时要有小数点</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：</li> <li>Ox O100</li> <li>M1808 M1968 零标志 Zero flag</li> <li>M1809 M1969 借位标志 Borrow flag</li> <li>M1810 M1970 进位标志 Carry flag</li> <li>M1793 M1953 运算错误标志 Operation error flag</li> </ul>				
请参考下列补充说明																				

## 指令说明

- ◆ **S<sub>1</sub>**：底数装置。**S<sub>2</sub>**：次幂数装置。**D**：运算结果装置。
  - ◆ 将二进制浮点数据 **S<sub>1</sub>** 及 **S<sub>2</sub>** 以次幂数相乘后存放于 **D**。
- $$D = POW [S_{1+1} \cdot S_1]^{[S_{2+1} \cdot S_2]}$$
- ◆ **S<sub>1</sub>** 内容值只有正数有效，**S<sub>2</sub>** 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 **S<sub>1</sub>**、**S<sub>2</sub>** 需为浮点数值。
  - ◆ 若 **S<sub>1</sub>** 及 **S<sub>2</sub>** 的内容值无效，视为“运算错误”，指令不执行，运算错误标志 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，则进位标志=On。
  - ◆ 若运算结果的绝对值小于可表示之最小浮点值，则借位标志=On。
  - ◆ 若运算结果为 0，则零标志=On。
  - ◆ 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)

## 程序范例





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

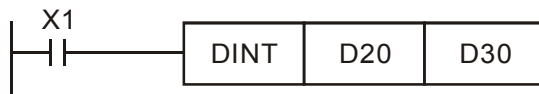
API																	适用機種	
<b>129</b>	<b>D</b>	<b>INT</b>	<b>P</b>	<b>S</b>	<b>D</b>	二进制浮点数→BIN 整数转换											10PM	
																	✓	
	位装置				字符装置												16 位指令	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S													*			-		
D													*			-		
操作数使用注意：各装置使用范围请参考功能规格表 此指令只有 32 位指令 <b>DINT</b> 、 <b>DINTP</b> 有效													32 位指令 (5 STEP) <b>DINT</b> 连续执行型 <b>DINTP</b> 脉冲执行型					
标志信号： Ox    O100 M1808    M1968    零标志 Zero flag M1809    M1969    借位标志 Borrow flag M1810    M1970    进位标志 Carry flag													请参考下列指令说明					

### 指令说明

- ◆ **S**：欲转换之来源装置。**D**：转换之结果。
- ◆ **S** 所指定的寄存器内容以二进制浮点数类型被转换成 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。
- ◆ 转换结果若为 0 时，零标志=On。  
转换结果有浮点数被舍弃时，借位标志=On。  
转换结果若超出下列范围时 (进位)，进位标志=On。  
32 位指令：-2,147,483,648~2,147,483,647

### 程序范例

- ◆ 当 X1=On 时，将二进制浮点数 (D21 · D20) 转换成 BIN 整数将结果存放至 (D31 · D30) 当中，BIN 整数浮点数被舍弃。

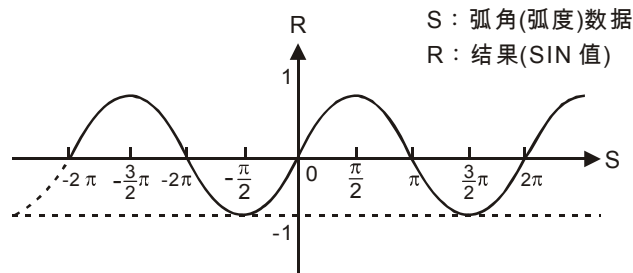


API																	适用機種	
130	D	SIN	P		(S)	(D)	二进制浮点数 SIN 运算										10PM	
																	✓	
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S					*								*			-		
D													*			-		
<p>• 操作数使用注意：角度范围：<math>0^\circ \leq \text{角度} &lt; 360^\circ</math></p> <p>各装置使用范围请参考功能规格表</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 DSIN、DSINP 有效</p>																		
<p>32 位指令 (6 STEP)</p> <p>DSIN 连续执行型 DSINP 脉冲执行型</p> <p>• 标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1760 M1920 弧度/角度使用标志</p> <p>请参考下列补充说明</p>																		

### 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弧度或角度，由弧度/角度使用标志决定。
- ◆ 当弧度/角度使用标志=Off 时，指定为弧度模式，弧度 (RAD) 值等于 (角度  $\times \pi / 180$ )。
- ◆ 当弧度/角度使用标志=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，零标志=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：

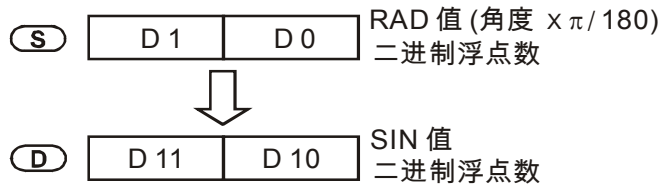


### 程序范例 (一)

- ◆ 弧度/角度使用标志=Off，指定为弧度模式，当 X0=On 时，指定二进制浮点数 (D11·D10) 之弧度 (RAD) 值求取 SIN 值后存于 (D11·D10) 当中，内容为二进制浮点数。

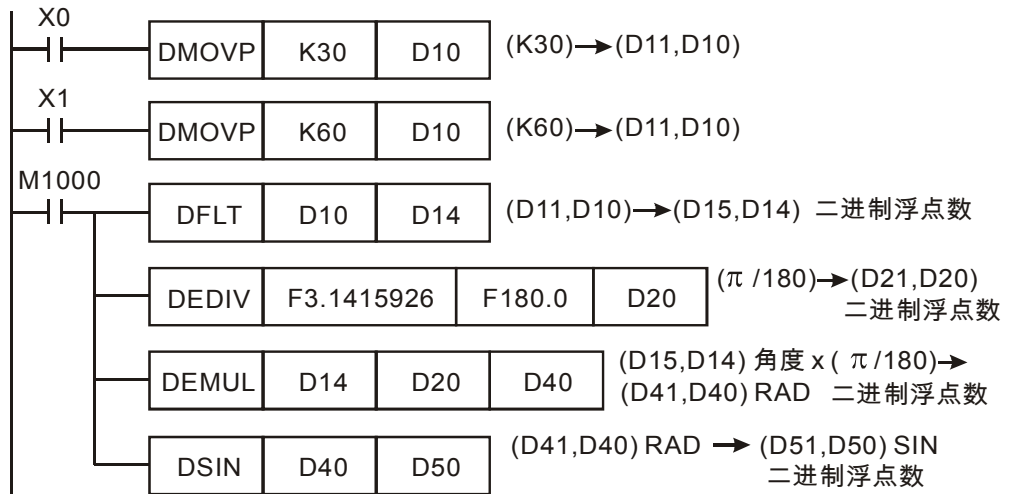


# 5 应用指令分类及基本使用



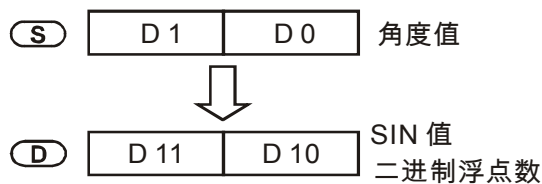
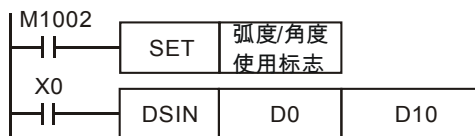
## 程序范例 (二)

- ◆ 弧度/角度使用标志=Off，指定为弧度模式，由输入端 X0 及 X1 来选择角度，转换成弧度 (RAD) 值后求取 SIN 值。



## 程序范例 (三)

- ◆ 弧度/角度使用标志=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

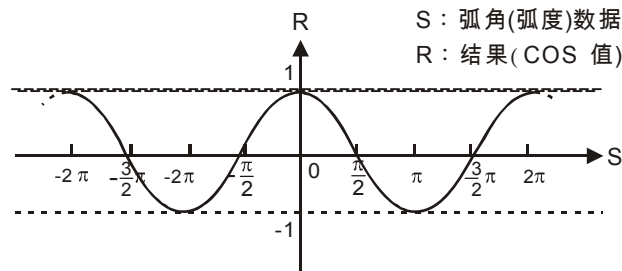
- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API																	适用機種
131	D	COS	P	(S)	(D)	二进制浮点数 COS 运算										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：角度范围：<math>0^{\circ} \leq \text{角度} &lt; 360^{\circ}</math></p> <p>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 DCOS、DCOSP 有效</p>																	
<p>32 位指令 (6 STEP)</p> <p>DCOS 连续执行型 DCOSP 脉冲执行型</p> <p>• 标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1760 M1920 弧度/角度使用标志</p> <p>请参考下列补充说明</p>																	

### 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弧度或角度，由标志弧度/角度使用标志决定。
- ◆ 当弧度/角度使用标志=Off 时，指定为弧度模式，弧度 (RAD) 值等于 (角度  $\times \pi / 180$ )。
- ◆ 当弧度/角度使用标志=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，零标志=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：

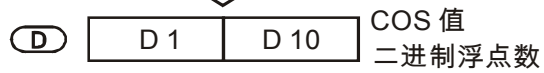
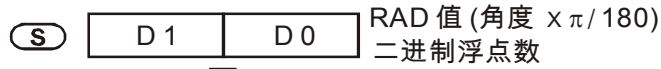


- ◆ 标志弧度/角度使用标志弧度/角度切换：当弧度/角度使用标志=Off 时，**S** 所指定的为弧度 (RAD) 值。当弧度/角度使用标志=On 时，**S** 所指定的为角度值 ( $0^{\circ} \sim 360^{\circ}$ )。
- ◆ 弧度/角度使用标志=Off 指定为弧度模式，当 X0=On 时，指定二进制浮点数 (D11，D10) 之弧度值求取 COS 值后存于 (D11，D10) 当中，内容为二进制浮点数。

### 程序范例 (一)

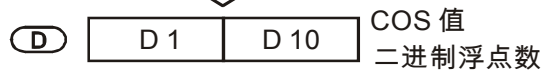


# 5 应用指令分类及基本使用



程序范例  
(二)

- ◆ 弧度/角度使用标志=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

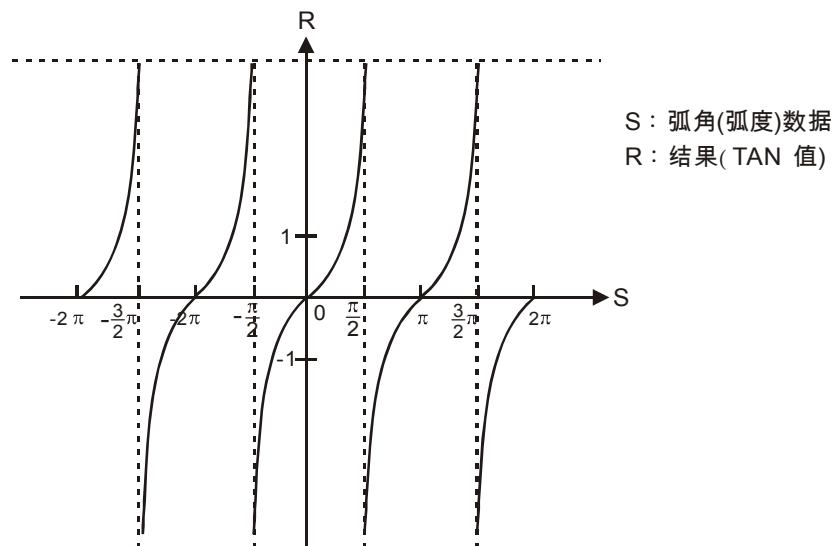
- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API																	适用機種		
132	D	TAN	P		(S)	(D)	二进制浮点数 TAN 运算										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													*			DTAN 连续执行型 DTANP 脉冲执行型			
<p>• 操作数使用注意：角度范围：<math>0^\circ \leq \text{角度} &lt; 360^\circ</math></p> <p>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 DTAN、DTANP 有效</p>																			
<p>• 标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1760 M1920 弧度/角度使用标志</p> <p>请参考下列补充说明</p>																			

### 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，由标志弧度/角度使用标志决定。
- ◆ 当弧度/角度使用标志=Off 时，指定为弧度模式，弧度 (RAD) 值等于 (角度  $\times \pi/180$ )。
- ◆ 当弧度/角度使用标志=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，零标志=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。

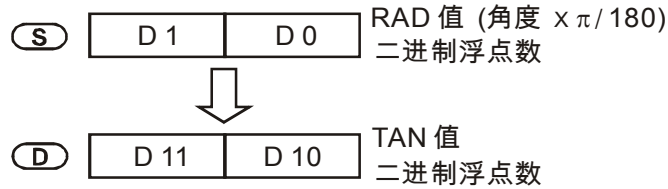
下图显示弧角与结果的关系：



### 程序范例 (一)

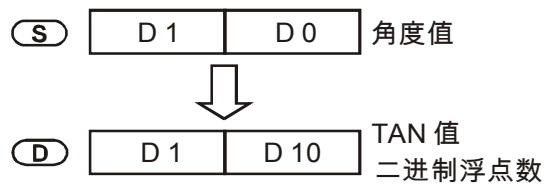
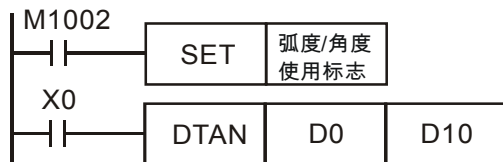
- ◆ 弧度/角度使用标志=Off 指定为弧度模式，当 X0=On 时，指定二进制浮点数 (D11·D0) 之弧度 (RAD) 值求取 TAN 值后存于 (D11·D10) 当中，内容为二进制浮点数。

# 5 应用指令分类及基本使用



程序范例  
(二)

- ◆ 弧度/角度使用标志=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。



API		ASIN		(S) (D)	二进制浮点数 ASIN 运算	适用機種
133	D		P			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) DASIN 连续执行型 DASINP 脉冲执行型  • 标志信号： Ox   O100 M1808 M1968 零标志 Zero flag M1793 M1953 运算错误标志 Operation error flag  请参考下列补充说明
D														*			

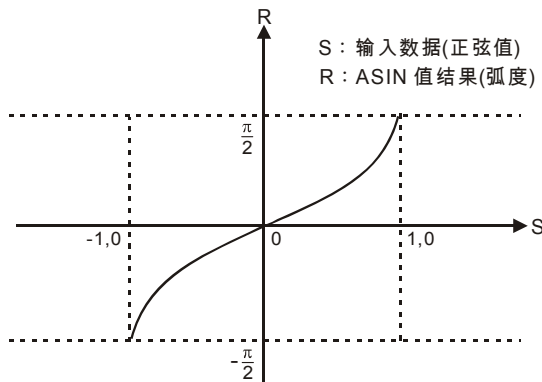
• 操作数使用注意：各装置使用范围请参考功能规格表

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

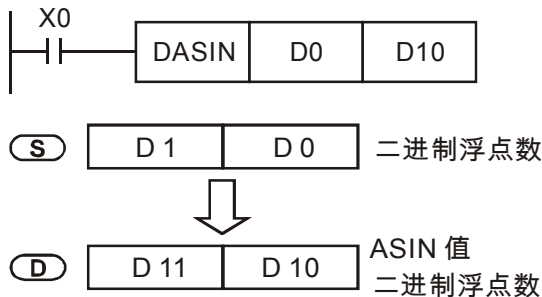
### 指令说明

- ◆ **S**：指定的来源（二进制浮点数）。**D**：取 ASIN 值结果。
  - ◆ **ASIN 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，运算错误标志 ON，并记录错误码 H'0E19。
- ◆ 若转换结果为 0，则零标志=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。

### 程序范例



### 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。

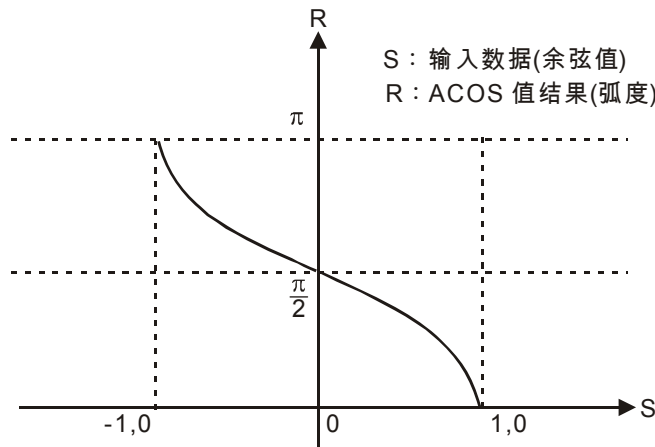
# 5 应用指令分类及基本使用

API																		适用機種																																																														
134	D	ACOS	P				(S)	(D)	二进制浮点数 ACOS 运算										10PM																																																													
																		✓																																																														
<table border="1"> <thead> <tr> <th rowspan="2"></th> <th colspan="4">位装置</th> <th colspan="12">字符装置</th> </tr> <tr> <th>X</th> <th>Y</th> <th>M</th> <th>S</th> <th>F</th> <th>H</th> <th>KnX</th> <th>KnY</th> <th>KnM</th> <th>KnS</th> <th>T</th> <th>C</th> <th>D</th> <th>V</th> <th>Z</th> </tr> </thead> <tbody> <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> </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> </tr> </tbody> </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													*		
	位装置				字符装置																																																																											
	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>本指令只有 32 位指令 <b>DACOS</b>、<b>DACOSP</b> 有效</p> <p>F 表示浮点数输入，使用时要有小数点</p>																																																																																
<p>16 位指令</p> <p>32 位指令 (6 STEP)</p> <p><b>DACOS</b> 连续执行型 <b>DACOSP</b> 脉冲执行型</p> <p>• 标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1793 M1953 运算错误标志 Operation error flag</p> <p>请参考下列补充说明</p>																																																																																

## 指令说明

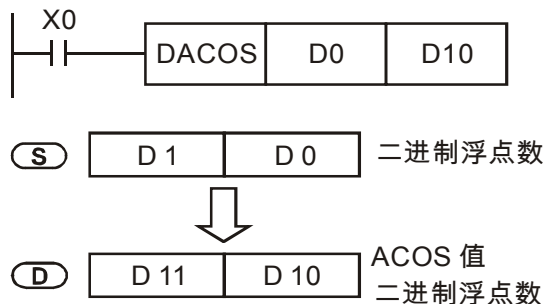
- ◆ **S**：指定的来源（二进制浮点数）。**D**：取 ACOS 值结果。
- ◆ ACOS 值 =  $\cos^{-1}$

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，运算错误标志 ON，并记录错误码 H'0E19。
- ◆ 若转换结果为 0，则零标志=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。

## 程序范例



## 补充说明

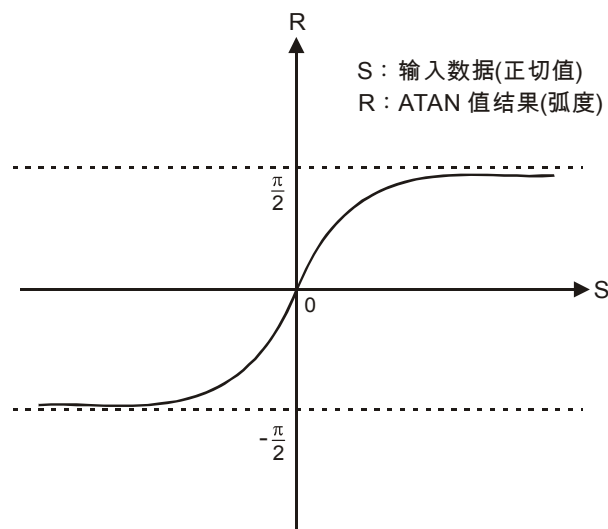
- ◆ 浮点数的运算功能请参考第 5.3 节应用指令对数值的处理方式之说明。

API																	适用機種	
135	D	ATAN	P		(S)	(D)	二进制浮点数 ATAN 运算										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>本指令只有 32 位指令 <b>DATAN</b>、<b>DATANP</b> 有效</p> <p>F 表示浮点数输入，使用时要有小数点</p>																		
<p>• 标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>请参考下列补充说明</p>																		

### 指令说明

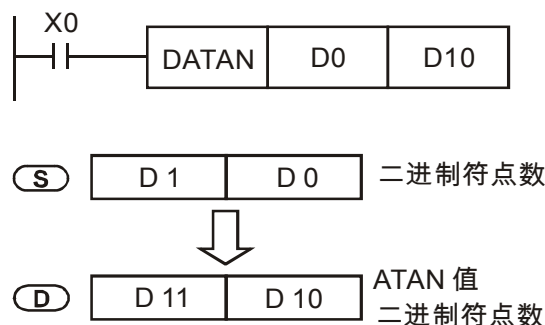
- ◆ **S**：指定的来源（二进制浮点数）。**D**：取 ATAN 值结果。
- ◆  $ATAN \text{ 值} = \tan^{-1}$

下图显示输入数据与结果的关系：



- ◆ 若转换结果为 0，则零标志=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。

### 程序范例



### 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

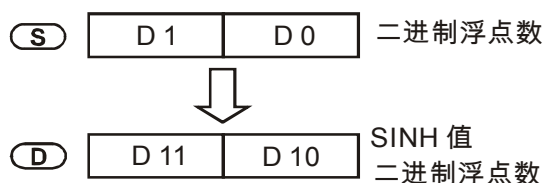
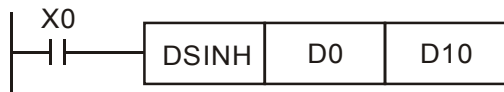
API														适用机种								
136	D	SINH	P	S		D		二进制浮点数 SINH 运算													10PM	
																			✓			
位装置				字符装置													16 位指令					
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-						
S					*								*			-						
D													*			-						
<p>• 操作数使用注意：各装置使用范围请参考功能规格表</p> <p>本指令只有 32 位指令 DSINH、DSINH P 有效</p> <p>F 表示浮点数输入，使用时要有小数点</p>																	<p>32 位指令 (6 STEP)</p> <p>DSINH 连续执行型 DSINH P 脉冲执行型</p> <p>• 标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1809 M1969 借位标志 Borrow flag</p> <p>M1810 M1970 进位标志 Carry flag</p> <p>请参考下列补充说明</p>					

指令说明

- ◆ S：指定的来源（二进制浮点数）。D：取 SINH 值结果。
- ◆  $\sinh \text{ 值} = (e^s - e^{-s}) / 2$

程序范例

- ◆ 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，则进位标志=On。
- ◆ 若转换结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若转换结果为 0，则零标志=On。
- ◆ 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。

补充说明

API																		适用機種			
137	D	COSH	P		S	D												10PM			
二进制浮点数 COSH 运算																					
	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>本指令只有 32 位指令 <b>DCOSH</b>、<b>DCOSH P</b> 有效</p> <p>F 表示浮点数输入，使用时要有小数点</p>																					
<p>• 标志信号：</p> <p>Ox O100</p> <p>M1808 M1968 零标志 Zero flag</p> <p>M1809 M1969 借位标志 Borrow flag</p> <p>M1810 M1970 进位标志 Carry flag</p> <p>请参考下列补充说明</p>																					

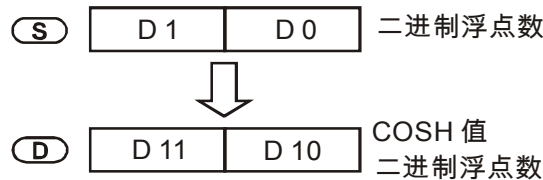
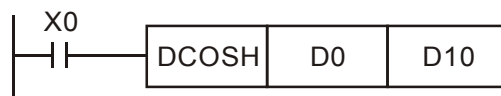
### 指令说明

◆ **S**：指定的来源（二进制浮点数）。**D**：取 COSH 值结果。

◆  $\cosh \text{ 值} = (e^s + e^{-s}) / 2$

### 程序范例

◆ 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，则进位标志=On。

◆ 若转换结果的绝对值小于可表示之最小浮点值，则借位标志=On。

◆ 若转换结果为 0，则零标志=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

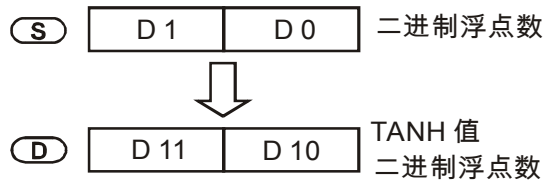
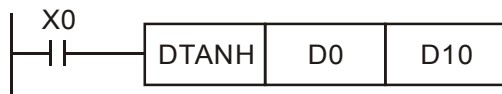
API																		适用機種
138	D	TANH	P		(S)	(D)	二进制浮点数 TANH 运算										10PM	
																		✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S					*								*			-		
D													*			-		
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>本指令只有 32 位指令 DTANH、DTANHP 有效</li> <li>F 表示浮点数输入，使用时要有小数点</li> </ul>														<ul style="list-style-type: none"> <li>标志信号：</li> <li>Ox O100</li> <li>M1808 M1968 零标志 Zero flag</li> <li>M1809 M1969 借位标志 Borrow flag</li> <li>M1810 M1970 进位标志 Carry flag</li> <li>请参考下列补充说明</li> </ul>				
<p>32 位指令 (6 STEP)</p> <p>DTANH 连续执行型 DTANHP 脉冲执行型</p>																		

## 指令说明

- ◆ S：指定的来源（二进制浮点数）。D：取 TANH 值结果。
- ◆  $\tanh$  值 =  $(e^s - e^{-s}) / (e^s + e^{-s})$

## 程序范例

- ◆ 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，则进位标志=On。
- ◆ 若转换结果的绝对值小于可表示之最小浮点值，则借位标志=On。
- ◆ 若转换结果为 0，则零标志=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。

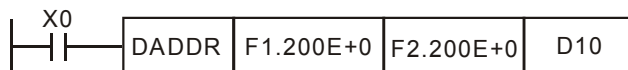
API																		适用機種
<b>172</b>	<b>D</b>	<b>ADDR</b>	<b>P</b>	(S1)	(S2)	(D)	浮点数值加算										10PM	
																		✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S1													*			-		
S2													*			-		
D													*			-		
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>本指令只有 32 位指令 <b>DADDR</b>、<b>DADDRP</b> 有效</li> </ul>														<ul style="list-style-type: none"> <li>标志信号：</li> <li>Ox O100</li> <li>M1808 M1968 零标志 Zero flag</li> <li>M1809 M1969 借位标志 Borrow flag</li> <li>M1810 M1970 进位标志 Carry flag</li> <li>请参考下列补充说明</li> </ul>				

### 指令说明

- ◆ **S<sub>1</sub>**：浮点数值被加数。**S<sub>2</sub>**：浮点数值被加数。**D**：和。
- ◆ **S<sub>1</sub>, S<sub>2</sub>** 操作数可输入浮点数值。
- ◆ **DADDR** 指令可直接在 **S<sub>1</sub>, S<sub>2</sub>** 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 **S<sub>1</sub>, S<sub>2</sub>** 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。
- ◆ **S<sub>1</sub>, S<sub>2</sub>** 可指定相同的寄存器编号, 此种情况下若是使用连续执行类型的指令时, 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, 则进位标志=On。若运算结果的绝对值小于可表示的最大浮点数值, 则借位标志=On。若运算结果为 0, 则零标志 =On。

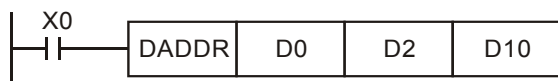
### 程序范例 (一)

- ◆ 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标号 **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。



### 程序范例 (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

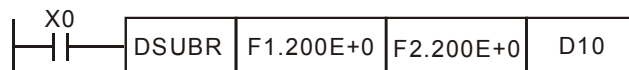
API																		适用机种
173	D	SUBR	P	(S1)	(S2)	(D)	浮点数值加算										10PM	
																		✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S1													*			-		
S2													*			-		
D													*			-		
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>本指令只有 32 位指令 DSUBR、DSUBRP 有效</li> </ul>																32 位指令 (13 STEP) DSUBR 连续执行型 DSUBRP 脉冲执行型  • 标志信号： Ox O100 M1808 M1968 零标志 Zero flag M1809 M1969 借位标志 Borrow flag M1810 M1970 进位标志 Carry flag  请参考下列补充说明		

### 指令说明

- ◆ S<sub>1</sub>：浮点数值被减数。S<sub>2</sub>：浮点数值减数。D：差。
- ◆ S<sub>1</sub>, S<sub>2</sub> 操作数可输入浮点数值。
- ◆ DSUBR 指令可直接在 S<sub>1</sub>, S<sub>2</sub> 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 S<sub>1</sub>, S<sub>2</sub> 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。
- ◆ S<sub>1</sub>, S<sub>2</sub> 可指定相同的寄存器编号, 此种情况下若是使用连续执行类型的指令时, 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, 则进位标志=On。若运算结果的绝对值小于可表示的最大浮点数值, 则借位标志=On。若运算结果为 0, 则零标志 =On。

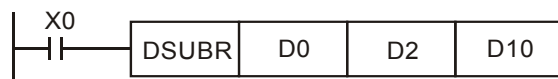
### 程序范例 (一)

- ◆ 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标号 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。



### 程序范例 (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。





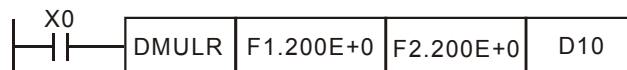
API					浮点数值乘算										适用機種	
174	D	MULR	P	(S1) (S2) (D)											10PM	
														✓		
	位装置				字符装置										16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-
S1													*			-
S2													*			-
D													*			-
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>本指令只有 32 位指令 DSUBR、DSUBRP 有效</li> </ul>																32 位指令 (13 STEP) DMULR 连续执行型 DMULRP 脉冲执行型 • 标志信号： Ox O100 M1808 M1968 零标志 Zero flag M1809 M1969 借位标志 Borrow flag M1810 M1970 进位标志 Carry flag 请参考下列补充说明

### 指令说明

- ◆ S<sub>1</sub>：浮点数值被乘数。S<sub>2</sub>：浮点数值乘数。D：积。
- ◆ S<sub>1</sub>, S<sub>2</sub> 操作数可输入浮点数值。
- ◆ DMULR 指令可直接在 S<sub>1</sub>, S<sub>2</sub> 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 S<sub>1</sub>, S<sub>2</sub> 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。
- ◆ S<sub>1</sub>, S<sub>2</sub> 可指定相同的寄存器编号, 此种情况下若是使用连续执行类型的指令时, 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, 则进位标志=On。若运算结果的绝对值小于可表示的最大浮点数值, 则借位标志=On。若运算结果为 0, 则零标志 =On。

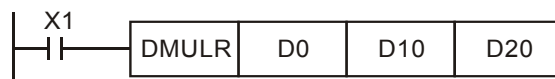
### 程序范例 (一)

- ◆ 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标号 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。



### 程序范例 (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。



# 5 应用指令分类及基本使用

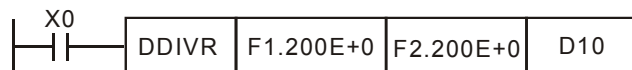
API																		适用机种
175	D	DIVR	P	(S1)	(S2)	(D)	浮点数值除算										10PM	
																		✓
	位装置				字符装置												16 位指令	
	X	Y	M	S	F	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	-		
S1													*			-		
S2													*			-		
D													*			-		
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>本指令只有 32 位指令 <b>DSUBR</b>、<b>DSUBRP</b> 有效</li> </ul>																32 位指令 (13 STEP) <b>DDIVR</b> 连续执行型 <b>DDIVP</b> 脉冲执行型  • 标志信号： Ox O100 M1808 M1968 零标志 Zero flag M1809 M1969 借位标志 Borrow flag M1810 M1970 进位标志 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类型的指令时, 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, 则进位标志=On。若运算结果的绝对值小于可表示的最大浮点数值, 则借位标志=On。若运算结果为 0, 则零标志 =On。

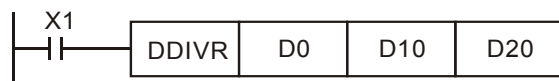
### 程序范例 (一)

- ◆ 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标号 **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。



### 程序范例 (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。



API																		适用機種
215~217	D	LD #					(S1)	(S2)										10PM
																		✓
		位装置				字符装置												
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S <sub>1</sub>						*	*	*	*	*	*	*	*	*	*	*		16 位指令 (5 STEP) LD # 连续执行型 - -
S <sub>2</sub>						*	*	*	*	*	*	*	*	*	*	*		32 位指令 (7 STEP) 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 • 标志信号：无																		

### 指令说明

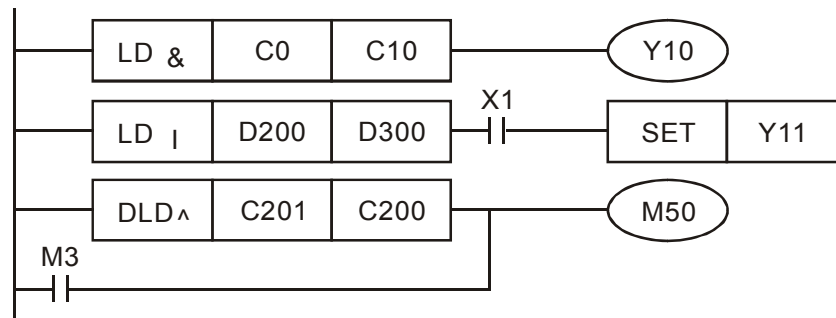
- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。
- ◆ S<sub>1</sub> 与 S<sub>2</sub> 之内容作比较的指令，比较结果不为 0 时，该指令导通，比较结果为 0 时，该指令不导通。
- ◆ LD # 的指令可直接与总线连接使用

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
215	LD&	DLD&	S <sub>1</sub> & S <sub>2</sub> ≠ 0	S <sub>1</sub> & S <sub>2</sub> = 0
216	LD	DLD	S <sub>1</sub>   S <sub>2</sub> ≠ 0	S <sub>1</sub>   S <sub>2</sub> = 0
217	LD^	DLD^	S <sub>1</sub> ^ S <sub>2</sub> ≠ 0	S <sub>1</sub> ^ S <sub>2</sub> = 0

- ◆ &：逻辑的'及' (AND) 运算。
- ◆ |：逻辑的'或' (OR) 运算。
- ◆ ^：逻辑的'异或' (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。

### 程序范例

- ◆ 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'异或' (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
218~220	D	AND #			(S1)	(S2)												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 <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*		32 位指令 (7 STEP)	
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*		DAND # 连续执行型 - -	
<ul style="list-style-type: none"> <li>操作数使用注意：#、&amp;、 、^</li> <li>各装置使用范围请参考功能规格表</li> </ul>																<ul style="list-style-type: none"> <li>标志信号：无</li> </ul>		

## 指令说明

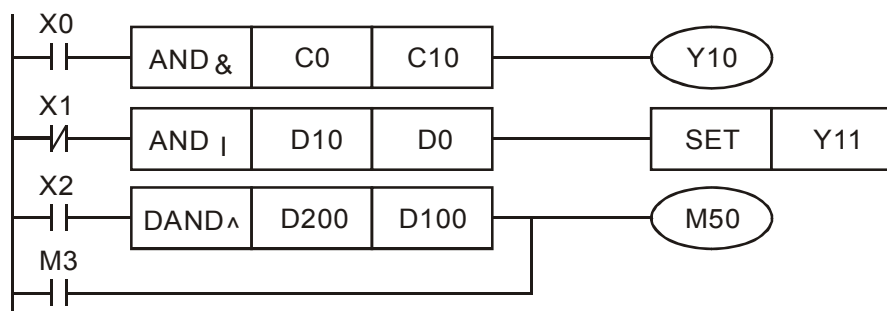
- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。
- ◆ S<sub>1</sub> 与 S<sub>2</sub> 之内容作比较的指令，比较结果不为 0 时，该指令导通，比较结果为 0 时，该指令不导通。
- ◆ AND # 的指令是与接点串接的运算指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
218	AND&	DAND&	S <sub>1</sub> & S <sub>2</sub> ≠ 0	S <sub>1</sub> & S <sub>2</sub> = 0
219	AND	DAND	S <sub>1</sub>   S <sub>2</sub> ≠ 0	S <sub>1</sub>   S <sub>2</sub> = 0
220	AND^	DAND^	S <sub>1</sub> ^ S <sub>2</sub> ≠ 0	S <sub>1</sub> ^ S <sub>2</sub> = 0

- ◆ &：逻辑的‘及’ (AND) 运算。
- ◆ |：逻辑的‘或’ (OR) 运算。
- ◆ ^：逻辑的‘异或’ (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。

## 程序范例

- ◆ 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‘异或’ (XOR) 运算不等于 0 时或是 M3=On 的时候，M50=On。



API																	适用機種
221~223	D	OR #			(S <sub>1</sub> )	(S <sub>2</sub> )											10PM
																	✓
	位装置				字符装置												16 位指令 (5 STEP) OR # 连续执行型 - - <hr/> 32 位指令 (7 STEP) DOR # 连续执行型 - - • 标志信号：无
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*	*	
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*	*	
• 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

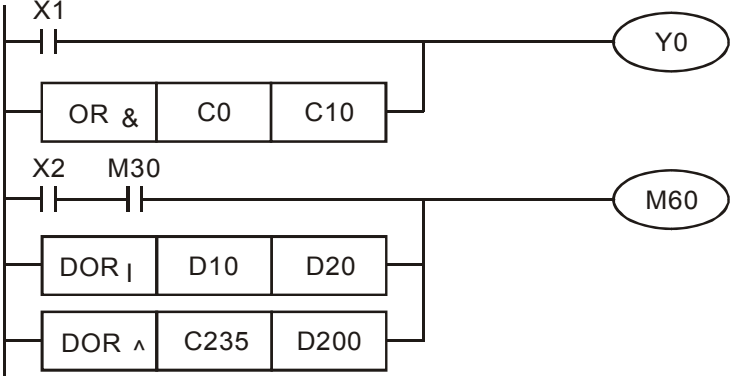
- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。
- ◆ S<sub>1</sub> 与 S<sub>2</sub> 之内容作比较的指令，比较结果不为 0 时，该指令导通，比较结果为 0 时，该指令不导通。
- ◆ OR # 的指令是与接点并接的运算指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
221	OR&	DOR&	S <sub>1</sub> & S <sub>2</sub> ≠ 0	S <sub>1</sub> & S <sub>2</sub> = 0
222	OR	DOR	S <sub>1</sub>   S <sub>2</sub> ≠ 0	S <sub>1</sub>   S <sub>2</sub> = 0
223	OR^	DOR^	S <sub>1</sub> ^ S <sub>2</sub> ≠ 0	S <sub>1</sub> ^ S <sub>2</sub> = 0

- ◆ &：逻辑的'及' (AND) 运算。
- ◆ |：逻辑的'或' (OR) 运算。
- ◆ ^：逻辑的'异或' (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。

### 程序范例

- ◆ 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'异或' (XOR) 运算不等于 0 时，M60=On。



# 5 应用指令分类及基本使用

API																		适用机种
224~230	D	LD※					(S1)	(S2)										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 - -		
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) DLD※ 连续执行型 - - • 标志信号：无		

## 指令说明

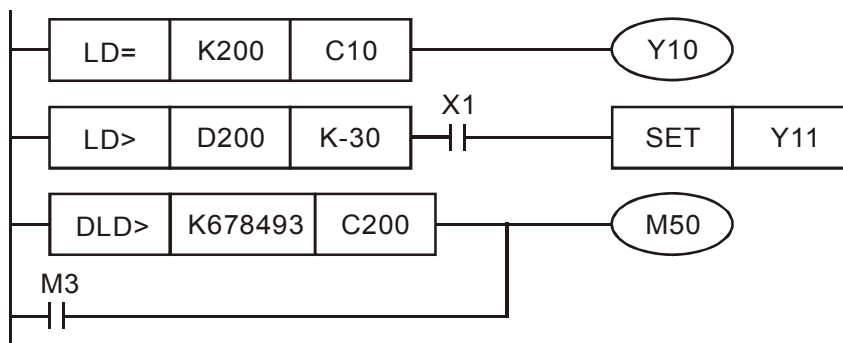
- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。
- ◆ S<sub>1</sub> 与 S<sub>2</sub> 之内容作比较的指令，以 API 224 LD= 为例，比较结果为“等于”时，该指令导通，“不等于”时，该指令不导通。
- ◆ LD※的指令可直接与总线连接使用

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
224	LD =	DLD =	S <sub>1</sub> = S <sub>2</sub>	S <sub>1</sub> ≠ S <sub>2</sub>
225	LD >	DLD >	S <sub>1</sub> > S <sub>2</sub>	S <sub>1</sub> ≤ S <sub>2</sub>
226	LD <	DLD <	S <sub>1</sub> < S <sub>2</sub>	S <sub>1</sub> ≥ S <sub>2</sub>
228	LD < >	DLD < >	S <sub>1</sub> ≠ S <sub>2</sub>	S <sub>1</sub> = S <sub>2</sub>
229	LD < =	DLD < =	S <sub>1</sub> ≤ S <sub>2</sub>	S <sub>1</sub> > S <sub>2</sub>
230	LD > =	DLD > =	S <sub>1</sub> ≥ S <sub>2</sub>	S <sub>1</sub> < S <sub>2</sub>

- ◆ 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。

## 程序范例

- ◆ 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。



API																		适用機種	
232~238	D	AND※					(S1)	(S2)	接点类型比较 AND※										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 <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*				
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*				
• 操作数使用注意：※：=、>、<、<>、≤、≥ 各装置使用范围请参考功能规格表																	• 标志信号：无		

### 指令说明

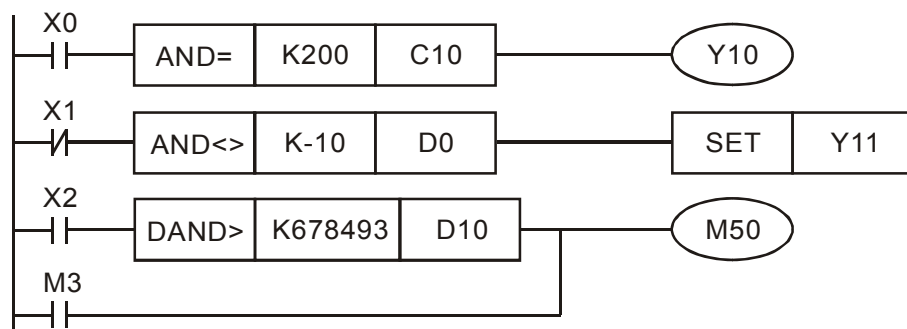
- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。
- ◆ S<sub>1</sub> 与 S<sub>2</sub> 之内容作比较的指令，以 API 232 AND= 为例，比较结果为等于时，该指令导通，不等于时，该指令不导通。
- ◆ AND※的指令是与接点串接的比较指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
232	AND =	DAND =	S <sub>1</sub> = S <sub>2</sub>	S <sub>1</sub> ≠ S <sub>2</sub>
233	AND >	DAND >	S <sub>1</sub> > S <sub>2</sub>	S <sub>1</sub> ≤ S <sub>2</sub>
234	AND <	DAND <	S <sub>1</sub> < S <sub>2</sub>	S <sub>1</sub> ≥ S <sub>2</sub>
236	AND < >	DAND < >	S <sub>1</sub> ≠ S <sub>2</sub>	S <sub>1</sub> = S <sub>2</sub>
237	AND < =	DAND < =	S <sub>1</sub> ≤ S <sub>2</sub>	S <sub>1</sub> > S <sub>2</sub>
238	AND > =	DAND > =	S <sub>1</sub> ≥ S <sub>2</sub>	S <sub>1</sub> < S <sub>2</sub>

- ◆ 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。

### 程序范例

- ◆ 当 X0=On 时且 C10 的当前值又等于 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。



# 5 应用指令分类及基本使用

API																			适用機種
240~246	D	OR※					(S <sub>1</sub> )	(S <sub>2</sub> )	接点类型比较 OR※										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 <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*				
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*				
• 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) DOR※ 连续执行型 - - • 标志信号：无			

## 指令说明

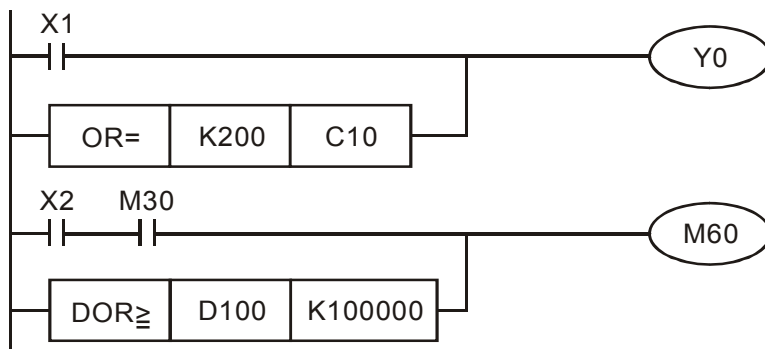
- ◆ S<sub>1</sub>：数据来源装置 1。S<sub>2</sub>：数据来源装置 2。
- ◆ S<sub>1</sub> 与 S<sub>2</sub> 之内容作比较的指令，以 API 240 OR= 为例，比较结果为等于时，该指令导通，不等于时，该指令不导通。
- ◆ OR※的指令是与接点并接的比较指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
240	OR =	DOR =	S <sub>1</sub> = S <sub>2</sub>	S <sub>1</sub> ≠ S <sub>2</sub>
241	OR >	DOR >	S <sub>1</sub> > S <sub>2</sub>	S <sub>1</sub> ≤ S <sub>2</sub>
242	OR <	DOR <	S <sub>1</sub> < S <sub>2</sub>	S <sub>1</sub> ≥ S <sub>2</sub>
244	OR < >	DOR < >	S <sub>1</sub> ≠ S <sub>2</sub>	S <sub>1</sub> = S <sub>2</sub>
245	OR ≤	DOR ≤	S <sub>1</sub> ≤ S <sub>2</sub>	S <sub>1</sub> > S <sub>2</sub>
246	OR ≥	DOR ≥	S <sub>1</sub> ≥ S <sub>2</sub>	S <sub>1</sub> < S <sub>2</sub>

- ◆ 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。

## 程序范例

- ◆ 当 X1=On 时，或者是 C10 的当前值等于 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。





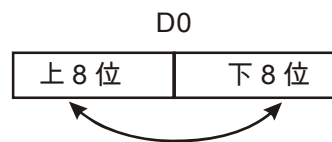
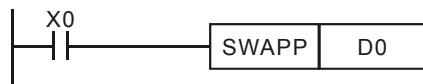
API																	适用機種
<b>147</b>	<b>D</b>	<b>SWAP</b>	<b>P</b>														10PM
																	✓
	位装置				字符装置												
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S							*	*	*	*	*	*	*	*	*		
<p>• 操作数使用注意：此指令有支持 V·Z 装置（当 16 位指令时不能使用装置；当 32 位指令时不能使用 V 装置）</p> <p>各装置使用范围请参考功能规格表</p> <p>指定 KnX(Y/M/S)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>																16 位指令 (5 STEP)	
																SWAP 连续执行型 SWAPP 脉冲执行型	
																32 位指令 (7 STEP)	
																DSWAP 连续执行型 DSWAPP 脉冲执行型	
																• 标志信号：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)。

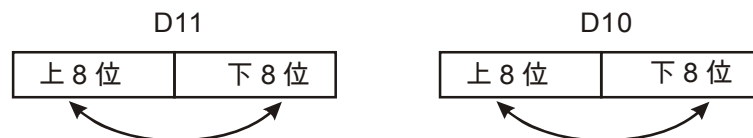
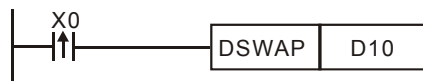
### 程序范例 (一)

- ◆ 当 X0=On 时，将 D0 的上 8 位与下 8 位的内容互相交换。



### 程序范例 (二)

- ◆ 当 X0=On 时，将 D11 的上位 8 位与下 8 位的内容互相交换, D10 的上位 8 位与下 8 位的内容互相交换。



# 5 应用指令分类及基本使用

API																		适用機種
<b>154</b>	<b>D</b>	<b>RAND</b>	<b>P</b>				(S1)	(S2)	(D)									10PM
																		✓

	位装置				字符装置												
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S <sub>1</sub>					*	*	*	*	*	*	*	*	*	*	*		16 位指令 (5 STEP) RAND 连续执行型 RANDP 脉冲执行型
S <sub>2</sub>					*	*	*	*	*	*	*	*	*	*	*		32 位指令 (7 STEP) DRAND 连续执行型 DRANDP 脉冲执行型
D								*	*	*	*	*	*	*	*		• 标志信号：无

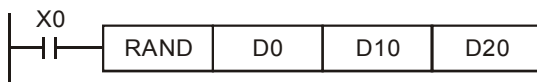
• 操作数使用注意：此指令有支持 V、Z 装置（当 16 位指令时不能使用装置；当 32 位指令时不能使用 V 装置）  
各装置使用范围请参考功能规格表  
指定 KnX(Y/M/S)等装置的位装置时，限以 8 进制或 10 进制中 16 的倍数为起始编号 如 K1X0, K4Y20 (八进制)；K1M0, K4S16 (十进制)

### 指令说明

- ◆ S<sub>1</sub>：随机数产生的范围下限。S<sub>2</sub>：随机数产生的范围上限。D：随机数产生的结果。
- ◆ 16 位操作数 S<sub>1</sub>, S<sub>2</sub> 使用范围:  $K0 \leq S_1, S_2 \leq K32,767$ , 32 位操作数 S<sub>1</sub>, S<sub>2</sub> 使用范围:  $K0 \leq S_1, S_2 \leq K2,147,483,647$ 。
- ◆ 操作数 S<sub>1</sub> ≤ 操作数 S<sub>2</sub>，若使用者输入 S<sub>1</sub> > S<sub>2</sub>, PM 判断为运算错误。

### 程序范例

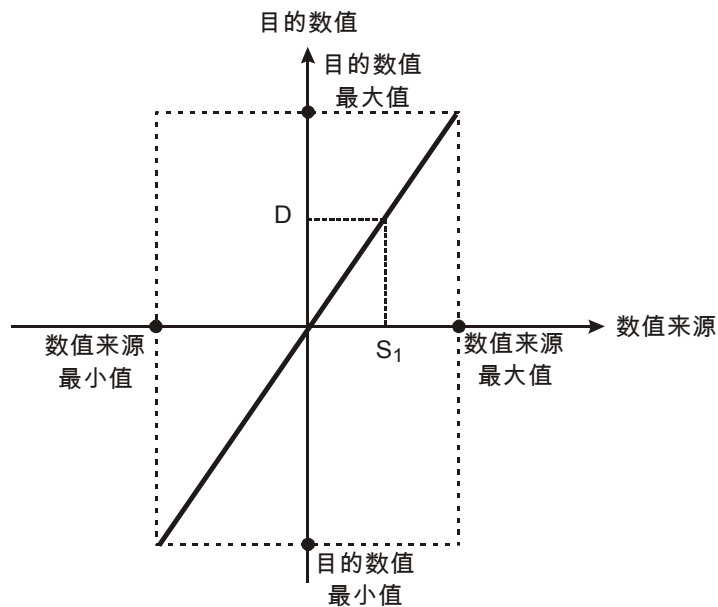
- ◆ 当 X0=On, RAND 指令产生介于范围下限 D0 与范围上限 D10 的随机数，将结果存放至 D20 内。



API																	适用机种	
202		SCAL	P	(S1)	(S2)	(S3)	(D)	比例运算								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	SCAL 连续执行型 SCALP 脉冲执行型		
S1					*	*							*			32 位指令 (7 STEP)		
S2					*	*							*			-		
S3					*	*							*			-		
D													*			-		
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> <li>标志信号：无</li> </ul>																		

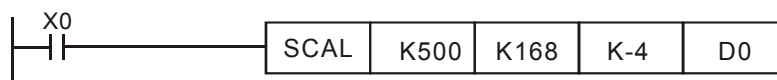
### 指令说明

- ◆ S<sub>1</sub>：来源数值数据。S<sub>2</sub>：斜率，单位为 0.001。S<sub>3</sub>：偏移量。D：目的地装置。
- ◆ 操作数范围 S<sub>1</sub>, S<sub>2</sub>, S<sub>3</sub> 是 -32767~32767
- ◆ 指令内部运算公式为:  $D = (S_1 \times S_2) \div 1000 + S_3$
- ◆ S<sub>2</sub> 和 S<sub>3</sub> 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 位的整数值输入。
- ◆ 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：

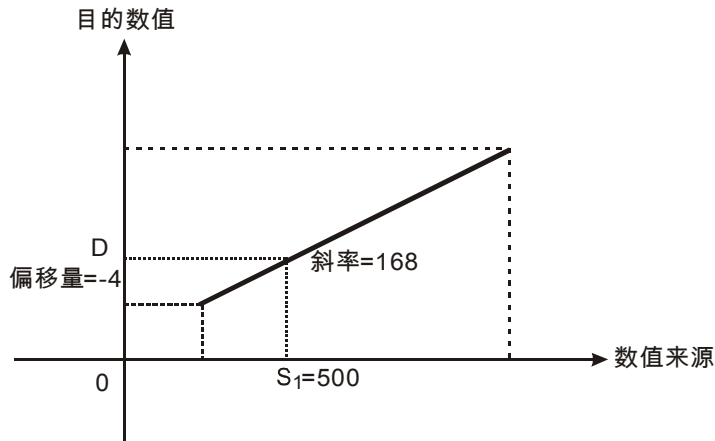


### 程序范例 (一)

- ◆ 已知 S<sub>1</sub> 数值数据来源为 500, S<sub>2</sub> 斜率为 168, S<sub>3</sub> 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:  $D_0 = (500 \times 168) \div 1000 + (-4) = 80$

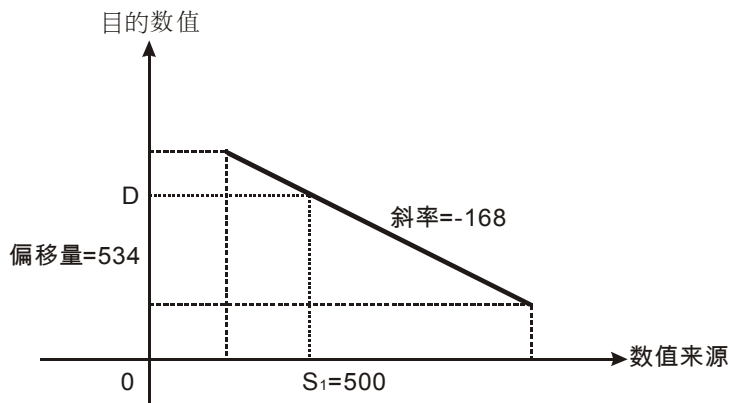
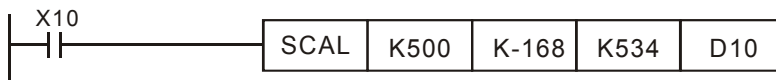


## 5 应用指令分类及基本使用



### 程序范例 (二)

- ◆ 已知  $S_1$  数值数据来源为 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  $D = -32,768$ 。

API																	适用机种	
203	D	SCLP	P		(S1)	(S2)	(D)	参数型比例运算									10PM	
																	✓	
	位装置				字符装置												16 位指令 (7 STEP) SCLP 连续执行型 SCLPP 脉冲执行型 32 位指令 (13 STEP) DSCLP 连续执行型 DSCLPP 脉冲执行型 • 标志信号：标志信号：M1162 十进制整数与二进制浮点数使用标志，On 时表示二进制浮点数	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z			
S <sub>1</sub>					*	*							*					
S <sub>2</sub>					*	*							*					
S <sub>3</sub>					*	*							*					
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**

- ◆ S<sub>1</sub>：数值数据来源。S<sub>2</sub>：参数，单位为 0.001。D：目的地装置。
- ◆ 16 位指令 S<sub>2</sub> 参数设定内容如下：

装置编号	参数名称与说明	设定范围
S <sub>2</sub>	来源数值最大值	-32768~32767
S <sub>2</sub> +1	来源数值最小值	-32768~32767
S <sub>2</sub> +2	目的数值最大值	-32768~32767
S <sub>2</sub> +3	目的数值最小值	-32768~32767

- ◆ 16 位指令 S<sub>2</sub> 操作数将连续占用 4 个装置。
- ◆ 32 位指令 S<sub>2</sub> 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 <sub>2</sub> 、S <sub>2</sub> +1	来源数值最大值	-2,147,483,648~2,147,483,647	32 位浮点数范围
S <sub>2</sub> +2、3	来源数值最小值		
S <sub>2</sub> +4、5	目的数值最大值		
S <sub>2</sub> +6、7	目的数值最小值		

- ◆ 32 位指令 S<sub>2</sub> 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：

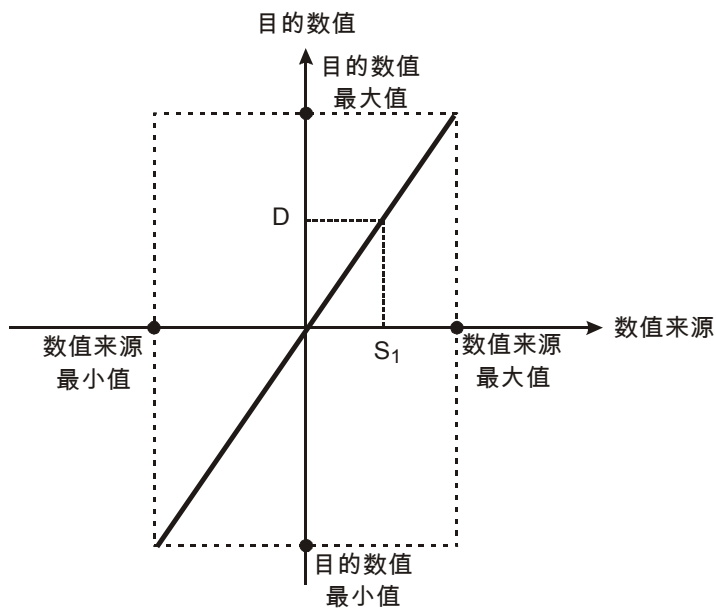
$$y = kx + b, y = \text{目的数值 (D)}, k = \text{斜率} = (\text{目的数值最大值} - \text{目的数值最小值}) \div (\text{来源数值最大值} - \text{来源数值最小值}), x = \text{来源数值 (S}_1), b = \text{偏移量} = \text{目的数值最小值} - \text{来源数值最小值} \times \text{斜率}$$

- ◆ 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：

# 5 应用指令分类及基本使用

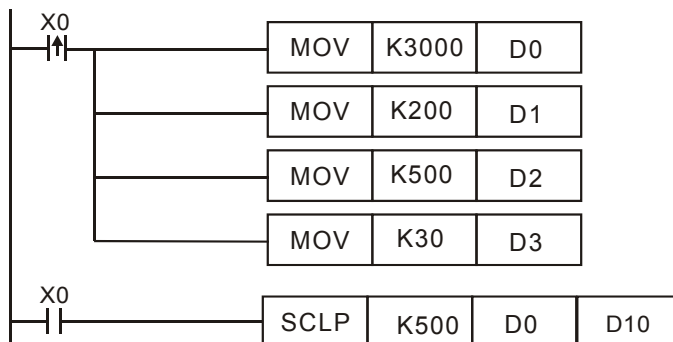
$y=kx+b = D = k S_1 + b = \text{斜率} \times S_1 + \text{偏移量} = \text{斜率} \times S_1 + \text{目的数值最小值} - \text{来源数值最小值} \times \text{斜率} = \text{斜率} \times (S_1 - \text{来源数值最小值}) + \text{目的数值最小值} = (S_1 - \text{来源数值最小值}) \times (\text{目的数值最大值} - \text{目的数值最小值}) \div (\text{来源数值最大值} - \text{来源数值最小值}) + \text{目的数值最小值}$

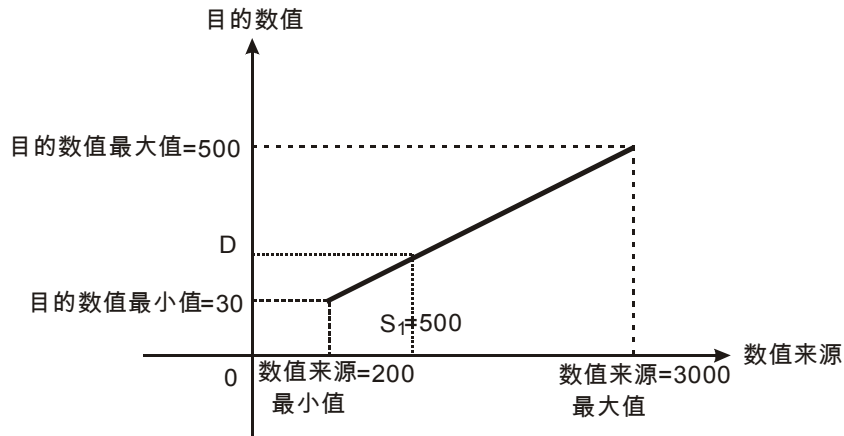
- ◆ 假如  $S_1 >$  来源数值最大值,  $S_1 =$  来源数值最大值。假如  $S_1 <$  来源数值最小值,  $S_1 =$  来源数值最小值。当输入数值与参数设定完成后, 则其输出曲线将如下图所示:



**程序范例 (一)**

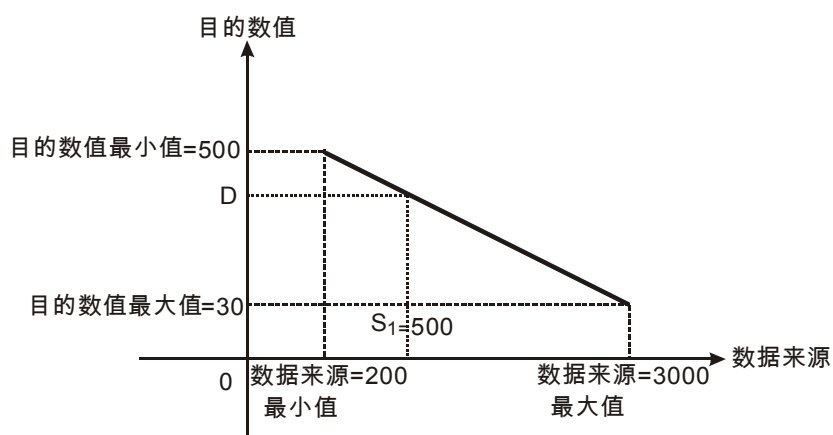
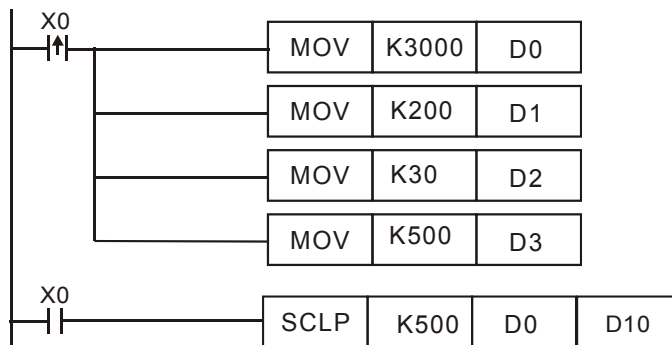
- ◆ 已知  $S_1$  数值数据来源为 500, 来源数值最大值  $D0=3000$ , 来源数值最小值  $D1=200$ , 目的数值最大值  $D2=500$ , 目的数值最小值  $D3=30$ 。当  $X0=On$  时, SCLP 指令执行, 可在 D10 得到所要求的比例值。
- ◆ 运算方式:  $D10 = [(500 - 200) \times (500 - 30)] \div (3000 - 200) + 30 = 80.35$ 。取整数,  $D10 = 80$





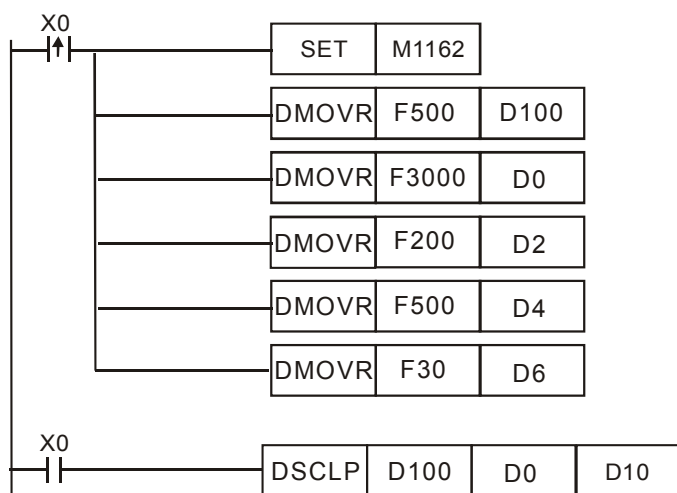
程序范例  
(二)

- ◆ 已知  $S_1$  数值数据来源为 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  $S_1$  数值数据来源  $D100=F500$ ，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， $D10 = F80$ 。

## 5 应用指令分类及基本使用



### 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,  $-32768 \sim 32767$ 。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,  $-2,147,483,648 \sim 2,147,483,647$ 。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。



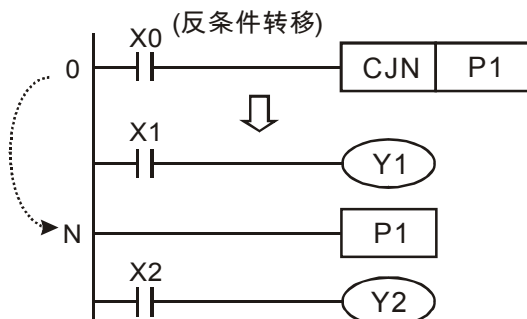
API																		适用機種		
256		CJN		P		(S)												10PM		
																		✓		
		位装置				字符装置										16 位指令 (3 STEP)				
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	CJN	连续执行型	CJNP	脉冲执行型
<ul style="list-style-type: none"> <li>操作数使用注意：S 操作数可指定 P</li> <li>S 操作数可指定 P0~P255</li> <li>P 装置不支持 V、Z 修饰</li>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

## 指令说明

- ◆ **S**：反条件转移之目的指标。
- ◆ 当 **CJN** 指令前的条件接点为导通状态时，则至下一行程序动作；反之，当条件接点为不导通状态时，则转移至所指定的 **P** 指标处动作。
- ◆ 当使用者希望 **O100** 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 **CJN** 或 **CJNP** 指令。
- ◆ 指针 **P** 所指之程序若在 **CJN** 指令之前，需注意会发生 **WDT** 逾时之错误，**O100** 停止运转，请注意使用。
- ◆ **CJN** 指令可重复指定同一指标 **P**，但 **CJN** 与 **CALL** 不可指定同一指标 **P**，否则会产生错误。
- ◆ 转移执行中各种装置动作情形说明：
  1. **Y**、**M**、**S** 保持转移发生前之状态。
  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；而一般应用指令不会被执行。
  4. 定时器之清除指令若在转移前被驱动，则在转移执行中该装置仍处于清除状态。

## 程序范例

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



# 5 应用指令分类及基本使用

API																				适用機種
257			<b>JMP</b>																	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	JMP 连续执行型 - -				
<ul style="list-style-type: none"> <li>操作数使用注意：S 操作数可指定 P</li> <li>S 操作数可指定 P0~P255</li> <li>不须接点驱动的指令</li> <li>P 装置不支持 V、Z 修饰</li> </ul>																32 位指令				
																- - - -				
																标志信号：无				

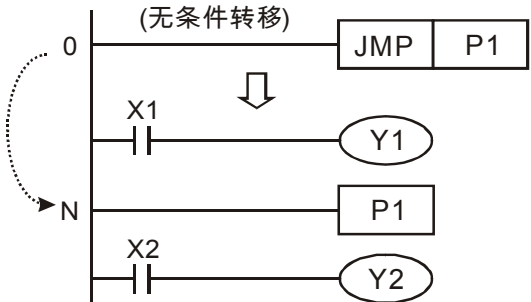
## 指令说明

◆ 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。

## 程序范例

◆ 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 转移至地址 N（即指定之标签 P1）继续执行，中间地址跳过不执行。



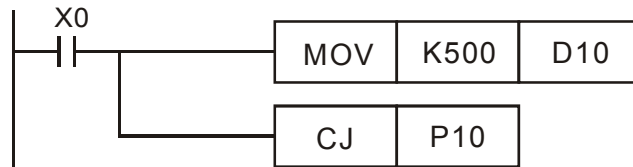
API														回总线			适用機種			
258	BRET																10PM			
																✓				
位装置				字符装置										16 位指令 (1 STEP)						
X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	BRET 连续执行型 - -					
• 操作数使用注意：无操作数 不须接点驱动的指令																	32 位指令			
																	标志信号：无			

### 指令说明

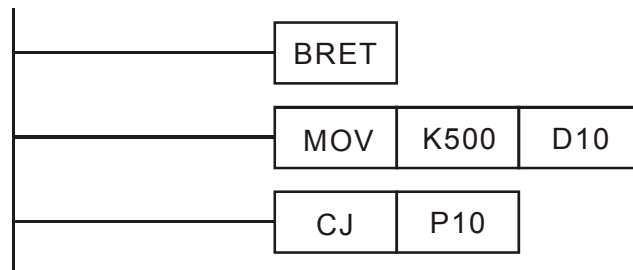
- ◆ BRET 指令无须接点驱动。
- ◆ 执行 BRET 指令之后，原本须条件接点驱动的指令相当于已连接上总线，可以直接执行这些指令。

### 程序范例

- ◆ 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，就像已经连接上总线，而可直接执行动作。如下所示：



# 5 应用指令分类及基本使用

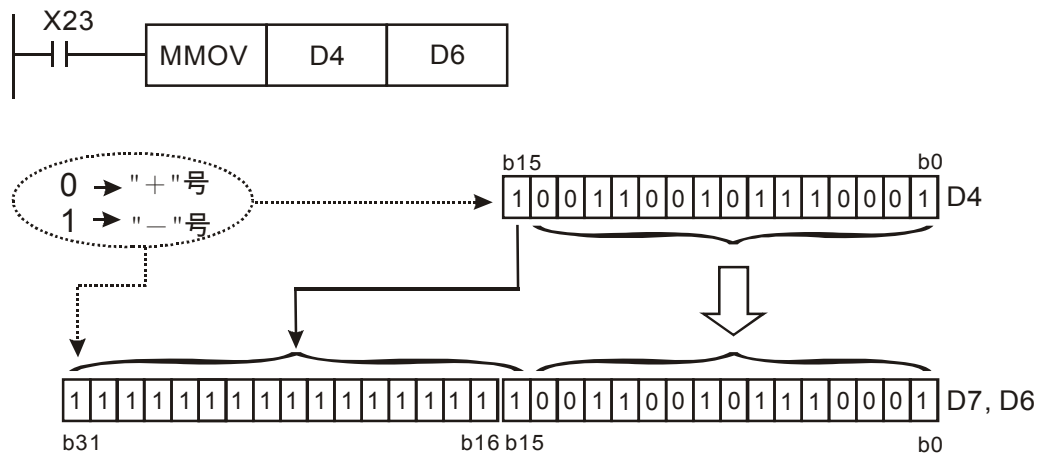
API																	适用機種	
259		MMOV		P		(S)	(D)										10PM	
																	✓	
	位装置				字符装置												16 位指令 (5 STEP)	
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z	MMOV 连续执行型    MMOVP 脉冲执行型		
S					*	*	*	*	*	*	*	*	*	*	*			
D							*	*	*	*	*	*	*	*	*			
<ul style="list-style-type: none"> <li>操作数使用注意：各装置使用范围请参考功能规格表</li> </ul>																32 位指令		
																标志信号：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。

## 程序范例

- ◆ 当 X23 为 0N 时，D4 的数据传送到 D6 和 D7。



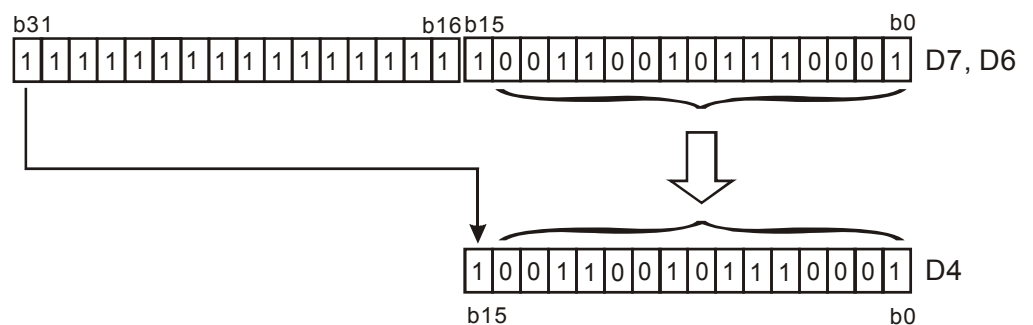
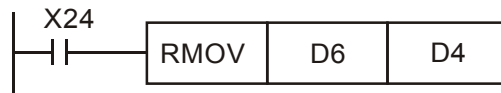
在以上的例子中，D4 的 b15 位数据传送到 (D7/D6) 的 b15 到 b31 位，变成负数 (和 D4 的一样)。

API																	适用机种
<b>260</b>		<b>RMOV</b>		<b>P</b>		(S)	(D)										10PM
																	✓
	位装置				字符装置												16 位指令 (5 STEP) RMOV 连续执行型 RMOV <sub>P</sub> 脉冲执行型 32 位指令 - - - - 标志信号：无
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	V	Z		
S					*	*	*	*	*	*	*	*	*	*	*	*	
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

- ◆ **S**：数据之来源 (32 位)。**D**：数据之搬移目的地 (16 位)。
- ◆ 将 32 位装置 **S** 中的数据传送到 16 位的装置 **D** 中，其中符号位被保留。
- ◆ 当 X24 为 0N 时，D6 和 D7 的数据传送到 D4。

### 程序范例



当 X24 为 0N 时，(S) 中最高位 (D7 : b31) 被传送到 (D) 中最高位 (D4 : b15) 中，其它位 (b0~b14) 则对应传送，而 b15~b30 被忽略未被传送。

# 5 应用指令分类及基本使用

## 5.7 运动功能块一览表

分类	功能块名称	功能叙述	页码
单轴运动功能	绝对单段速运动	轴运行至指定的绝对位置	5-161
	相对单段速运动	轴运行至与当前的相对位置	5-164
	绝对两段速运动	轴以两段速度运行至指定的绝对位置	5-167
	相对两段速运动	轴以两段速度运行至与当前的绝对位置	5-169
	插入单速段运动	轴以外部触发点当单段速运动起始点	5-172
	插入两段速运动	轴以外部触发点当两段速运动起始点	5-175
	寸动运动	轴以指定速度连续运动	5-178
	手摇轮模式运动	轴跟随手轮运动	5-180
	电子齿轮运动	轴跟随电子齿轮功能运动	5-182
	原点回归运动	轴运行原点回归动作	5-184
	单轴运动停止	轴运动停止	5-186
	参数设定 1	轴参数设定 1	5-187
	参数设定 2	轴参数设定 2	5-189
	读取当前位置/速度	显示轴当前位置与速度	5-190
	轴状态信息	显示轴目前的运动状态以及错误状态	5-191
	设定当前位置	设定轴当前位置	5-193
	输入极性设定	设定接点极性	5-194
多轴功能	多轴绝对直线插补	多轴绝对直线插补	5-195
	多轴相对直线插补	多轴相对直线插补	5-196
	多轴插补停止	多轴插补停止	5-199
其它	高速计数器	高速计数器设定及启动	5-201
	高速定时器	高速定时器设定及启动	5-204
	高速比较设定	高速比较器功能设定	5-206
	高速比较重置	重置高速比较器设定	5-208
	高速捕捉设定	高速捕捉功能设定	5-211
	高速捕捉遮蔽	遮蔽功能设定	5-214
	中断设定	设定中断程序功能	5-215

## 5.8 运动功能块引脚介绍

### 5.8.1 输入/输出引脚功能定义

下表为运动功能块的常见输入与输出引脚列表，单一运动功能块并非会有下表的所有引脚，例如以输入脚来说，一个功能块只会有 **Execute** 或 **Enable** 其中之一。

输入引脚			
名称	说明	格式	设定值
<b>Execute</b>	启动功能块	BOOL	TRUE / FALSE
<b>Enable</b>	启动功能块	BOOL	TRUE / FALSE

输出引脚			
名称	说明	格式	设定值
<b>Done</b>	功能块动作完成	BOOL	功能块完成时
<b>Valid</b>	输出值有效	BOOL	<b>Enable</b> 上升沿时
<b>Busy</b>	功能块动作执行中	BOOL	<b>Execute</b> 上升沿时
<b>Aborted</b>	功能块被其它命令中断	BOOL	执行中被命令停止
<b>Error</b>	功能块产生错误	BOOL	执行中发生错误

在一个运动功能块中必定包含 **Execute** 引脚或是 **Enable** 引脚，目的在启动此运动功能块。而为了显示运动功能块的执行状态一般会包含 **Busy** 和 **Done** 两种引脚，如果此功能块会被其它运动功能块中断时，则会增加 **Aborted**，另外 **Error** 引脚主要目的在显示此运动功能块在启动过程中发生错误。

单一运动功能块的输入引脚除上列的 **Execute** 和 **Enable** 外，会有其它提供给使用者作运动数据输入的引脚，这接数据/状态引脚会有以下特性：

- 输入数据取用时机：
  - 功能块启动为 **Execute** 时：数据在 **Execute** 上升沿触发被取用，要让变更数据再次生效需在修改后输入参数后再次让 **Execute** 上升沿触发。
  - 功能块启动为 **Enable** 时：参数会在 **Enable** 上升沿触发被取用，但与搭配 **Execute** 使用相比会比较常被设计为执行中持续更新的形式。
- 输入数据超过范围：
 

运动功能块在伴随着超出允许范围的数据输入时被启动，会造成输入的数据被限制或运动功能块产生错误。若此时产生轴的错误结果，此乃为运动功能块的应用错误所导致，需使用者在应用程序中避免错误的输入产生。
- 输出状态的互斥性：
  - 功能块启动为 **Execute** 时：**'Busy'**，**'Done'**，**'Aborted'**，**'Error'**彼此互斥，在一个运动功能块中同时只能有一个为真，当 **Execute** 为真时，里面其中一个必须为真，且**'Busy'**，**'Done'**，**'Aborted'**，**'Error'**同时间只有一个会被设置。

## 5 应用指令分类及基本使用

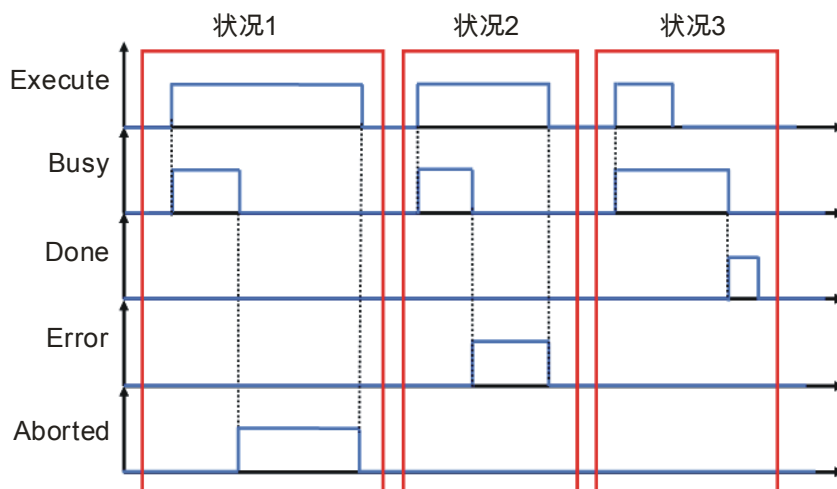
- 功能块启动为 **Enable** 时：输出‘Valid’ and ‘Error’彼此互斥，同时间只有一个能被设置。
- 输出数据/状态有效时机：
  - 功能块启动为 **Execute** 时：‘Done’、‘Error’、‘Aborted’以及数据输出会在‘Execute’下降沿时被重置，然而‘Execute’下降沿不会停止甚至也不影响 FB 实际的执行。即使在运动功能块完成前 **Execute** 就被重置，相对应的输出状态仍会产生并保持一个周期，如果运动功能块在完成之前又接收到新的 **execute**，运动功能块将不会对之前动作的‘Done’、‘Aborted’有任何反馈，且可能会产生运动功能块错误。
  - 功能块启动为 **Enable** 时：‘Valid’、‘Busy’、‘Error’输出会尽快跟着‘Enable’下降沿被重置。
- **Done** 输出特性：

‘Done’ 输出会在被命令的运动成功完成时被设置
- **Busy** 输出特性：
  - 功能块启动为 **Execute** 时：每个运动功能块会有一个 **Busy** 输出用来反应运动功能块尚未完成，并且新的输出状态(值)可预期会产生。**Busy** 在‘Execute’上升沿被设置，在 **Done**、‘Aborted’、‘Error’ 被设置时会被重置。
  - 功能块启动为 **Enable** 时：每个运动功能块会有一个 **Busy** 输出用来反应运动功能块尚未完成，并且新的输出状态(值)可预期会产生。**Busy** 在‘Enable’上升沿被设置，且只要运动功能块还在执行动作就会保持住。同时，对应的输出仍会变化。
- **Aborted** 输出特性：

‘Aborted’在运动尚未完成就被命令停止时会被设置。
- 输入 **Enable** 与输出 **Valid** 关系：

搭配 **Enable** 时若运动功能块有输出状态或数值的引脚就会有 **Valid** 输出引脚来表示这些输出是否有效。**Valid** 输出只在 **Enable** 为真及输出有效时为真，若运动功能块有错误会让输出不有效，**Valid** 会为假，直到错误状况消失，输出值重新有效‘Valid’才会再次被设置。

### 5.8.2 输入/输出引脚时序图

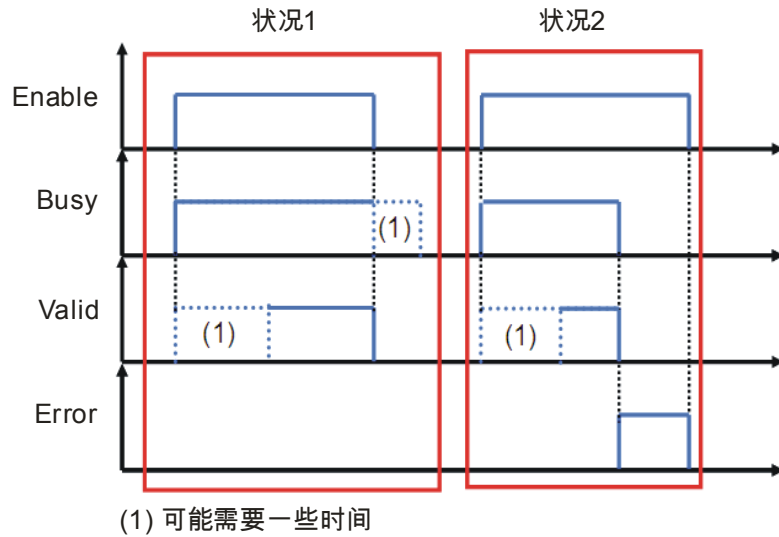


状况 1：运动功能块发生被强制中断时，输入 **Execute** 与输出 **Busy**、**Done**、**Error**、**Aborted** 时序图

状况 2：运动功能块发生被错误时，输入 **Execute** 与输出 **Busy**、**Done**、**Error**、**Aborted** 时序图

状况 3：运动功能块完成正常动作时，输入 **Execute** 与输出 **Busy**、**Done**、**Error**、**Aborted** 时序图





状况 1：运动功能块正常动作时，输入 Enable 与输出 Busy、Valid、Error 时序图

状况 2：运动功能块发生被错误时，输入 Enable 与输出 Busy、Valid、Error 时序图

### 5.8.3 PMSoft 软件操作介绍

以介绍台达运动功能块在 PMSoft 软件上面的操作方式。

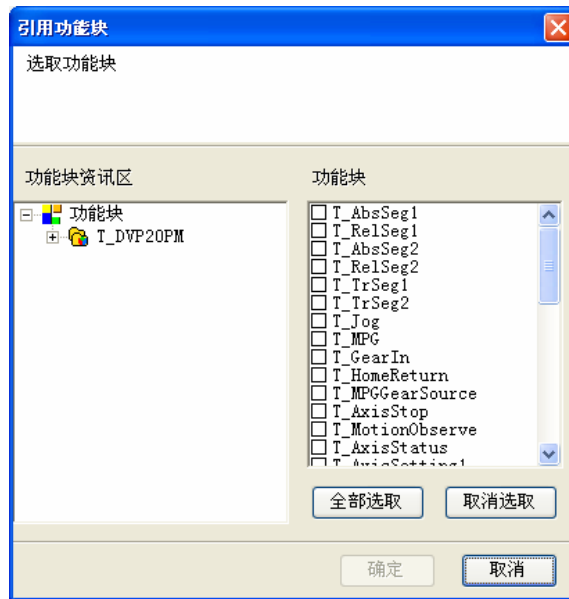
(1) 使用 PMSoft 开启新档案后，在功能块图标上利用鼠标右键叫出操作选单。



在选单中选择台达运动控制函数库，选择引用运动控制功能块。

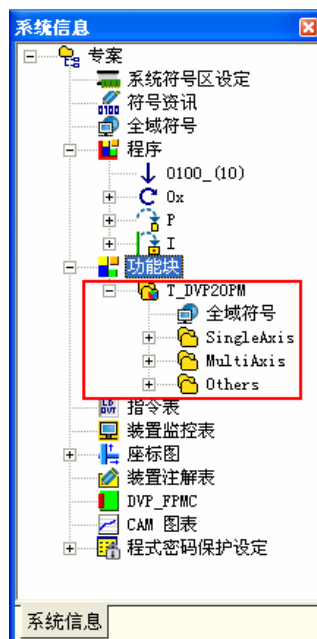
# 5 应用指令分类及基本使用

(2) 點選引用功能块后，会跳出以下窗口



在窗口中可以个别选择要引用的功能块，或选择全部选取，会引用全部的功能块，在选择完成后點選确定，即可开始引用台达运动控制功能块。

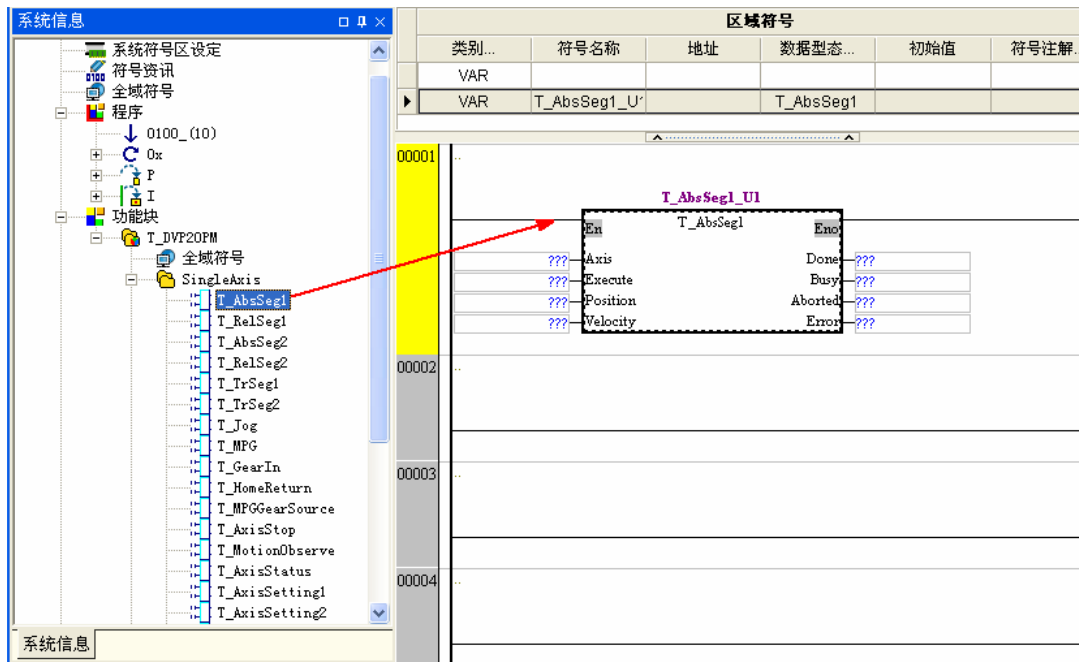
(3) 在引用功能块页面點選确定，系统信息中功能块会自动增加台达运动控制功能块。



● 数据夹的架构如下：

- + SingleAxis
- + MultiAxis
- + Others

- 各数据夹定义如下：
    - SingleAxis 为单轴运动，包含单轴点对点运动，电子齿轮跟随，电子凸轮跟随。
    - MultiAxis 为多轴运动，包含 G 码启动操作，多轴直线插补。
    - Net 为通讯功能，包含 DMCNET 以及 Ethernet。
    - Others 为其它功能，包含扩展存储卡操作，计数计时，高速比较捕捉，中断设定。
- (4) 使用台达运动控制功能块，仅需从功能块数据夹内拖拉欲使用的功能块，即可开始操作。



# 5 应用指令分类及基本使用

## 5.9 台达定义参数一览表

台达定义参数用于台达运动控制功能块的输入引脚，便于使用者在操作功能块时不需而外寻找功能块输入脚的说明，直接透过参数的操作即可设定想要的操作，参数列表如下：

参数名称	参数类型	数值	使用功能块	功能叙述
TRUE	BOOL	TRUE	所有功能块	输入引脚，写入 TRUE
FALSE	BOOL	FALSE		输入引脚，写入 FALSE
mcRising	BOOL	TRUE	插入两段速定位、	DOG 信号上升沿
mcFalling	BOOL	FALSE	插入单段速定位、原点回归	DOG 信号下降沿
mcPositive	BOOL	TRUE	原点回归	原点回归正转启动
mcNegative	BOOL	FALSE		原点回归反转启动
mcSCurve	BOOL	TRUE	参数设定 2	速度曲线为 S 曲线
mcTrapezoid	BOOL	FALSE		速度曲线为梯形曲线
mcNC	BOOL	TRUE	输入极性设定	常闭接点
mcNO	BOOL	FALSE		常开接点
mcUp_Up	BOOL	TRUE	高速计时	触发方式从上升沿到下降沿
mcUp_Down	BOOL	FALSE		触发方式从上升沿到下降沿
mcCmpSet	BOOL	TRUE	高速比较	比较成立时启动输出
mcCmpRst	BOOL	FALSE		比较成立时重置输出
mcMotor	WORD	0	参数设定 2	单位为电机单位
mcMachine	WORD	1		单位为机械单位
mcComp	WORD	2		单位为复合单位
mcUD	WORD	0	参数设定 2、 高速计数	脉冲类型为正反转脉冲
mcPD	WORD	1		脉冲类型为脉冲+方向
mcAB	WORD	2		脉冲类型为 AB 相脉冲
mc4AB	WORD	3		脉冲类型为 AB 相脉冲 4 倍频

参数名称	参数类型	数值	使用功能块	功能叙述
IntTimer	WORD	0	中断设定	中断信号来源为时间触发
IntX00	WORD	1		中断信号来源为外部 X0
IntX01	WORD	2		中断信号来源为外部 X1
IntX02	WORD	3		中断信号来源为外部 X2
IntX03	WORD	4		中断信号来源为外部 X3
IntX04	WORD	5		中断信号来源为外部 X4
IntX05	WORD	6		中断信号来源为外部 X5
IntX06	WORD	7		中断信号来源为外部 X6
IntX07	WORD	8		中断信号来源为外部 X7
IntStart0	WORD	1		中断信号来源为外部 Start0
IntStop0	WORD	2		中断信号来源为外部 Stop0
IntStart1	WORD	3		中断信号来源为外部 Start1
IntStop1	WORD	4		中断信号来源为外部 Stop1
mcCmpAxis1	WORD	0		高速比较
mcCmpAxis2	WORD	1	比较来源为第 2 轴目前位置	
mcCmpAxis3	WORD	2	比较来源为第 3 轴目前位置	
mcCmpAxis4	WORD	3	比较来源为第 4 轴目前位置	
mcCmpC200	WORD	4	比较来源为 C200 计数值	
mcCmpC204	WORD	5	比较来源为 C204 计数值	
mcCmpC208	WORD	6	比较来源为 C208 计数值	
mcCmpC212	WORD	7	比较来源为 C212 计数值	
mcCmpCLR0	WORD	0	高速比较	比较时，操作装置为 CLR0
mcCmpCLR1	WORD	1		比较时，操作装置为 CLR1
mcCmpY0	WORD	0		比较时，操作装置为 Y0
mcCmpY1	WORD	1		比较时，操作装置为 Y1
mcCmpY2	WORD	2		比较时，操作装置为 Y2
mcCmpY3	WORD	3		比较时，操作装置为 Y3
mcCmpRstC200	WORD	4		比较时，操作装置为 C200
mcCmpRstC204	WORD	5		比较时，操作装置为 C204
mcCmpRstC208	WORD	6		比较时，操作装置为 C208
mcCmpRstC212	WORD	7		比较时，操作装置为 C212

## 5 应用指令分类及基本使用

参数名称	参数类型	数值	使用功能块	功能叙述
mcCapAxis1	WORD	1	高速捕捉	捕捉来源为第 1 轴当前位置
mcCapAxis2	WORD	2		捕捉来源为第 2 轴当前位置
mcCapAxis3	WORD	3		捕捉来源为第 3 轴当前位置
mcCapAxis4	WORD	4		捕捉来源为第 4 轴当前位置
mcCapC200	WORD	7		捕捉来源为 C200 计数值
mcCapC204	WORD	8		捕捉来源为 C204 计数值
mcCapC208	WORD	9		捕捉来源为 C208 计数值
mcCapC212	WORD	10		捕捉来源为 C212 计数值
mcCapPG0	WORD	0	高速捕捉	捕捉触发信号来源 PG0
mcCapMPGB0	WORD	1		捕捉触发信号来源 MPGB0
mcCapMPGA0	WORD	2		捕捉触发信号来源 MPGA0
McCapLSN0	WORD	3		捕捉触发信号来源 LSN0
McCapLSP0	WORD	4		捕捉触发信号来源 LSP0
McCapDOG0	WORD	5		捕捉触发信号来源 DOG0
mcCapStop0	WORD	6		捕捉触发信号来源 Stop0
mcCapStart0	WORD	7		捕捉触发信号来源 Start0
mcCapPG1	WORD	8		捕捉触发信号来源 PG1
mcCapMPGB1	WORD	9		捕捉触发信号来源 MPGB1
mcCapMPGA1	WORD	10		捕捉触发信号来源 MPGA1
mcCapLSN1	WORD	11		捕捉触发信号来源 LSN1
mcCapLSP1	WORD	12		捕捉触发信号来源 LSP1
McCapDOG1	WORD	13		捕捉触发信号来源 DOG1
mcCapStop1	WORD	14		捕捉触发信号来源 Stop1
mcCapStart1	WORD	15		捕捉触发信号来源 Start1
mcX0	WORD	0		捕捉触发信号来源 X0
mcX1	WORD	1		捕捉触发信号来源 X1
mcX2	WORD	2		捕捉触发信号来源 X2
mcX3	WORD	3		捕捉触发信号来源 X3
mcX4	WORD	4	捕捉触发信号来源 X4	
mcX5	WORD	5	捕捉触发信号来源 X5	
mcX6	WORD	6	捕捉触发信号来源 X6	
mcX7	WORD	7	捕捉触发信号来源 X7	
mcX10	WORD	8	捕捉触发信号来源 X10	
mcX11	WORD	9	捕捉触发信号来源 X11	
mcX12	WORD	10	捕捉触发信号来源 X12	
mcX13	WORD	11	捕捉触发信号来源 X13	

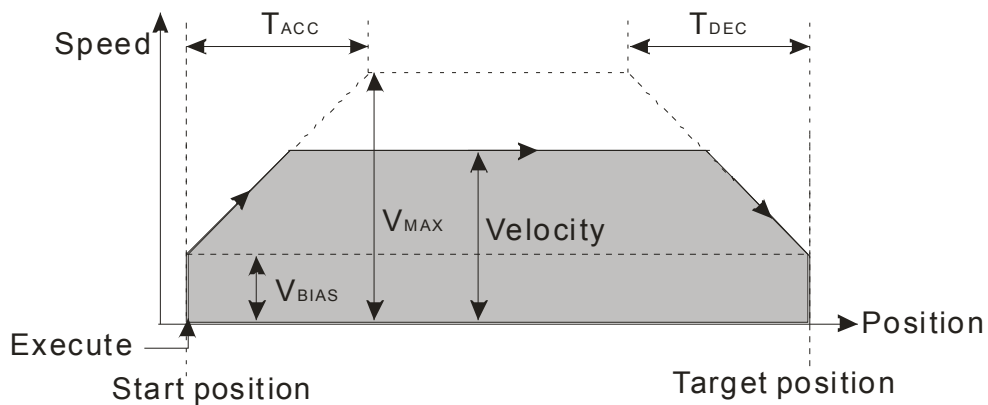
## 5.10 单轴运动功能块说明

### 5.10.1 绝对单段速运动

En	T_AbsSeg1	Eno
Axis		Done
Execute		Busy
Position		Aborted
Velocity		Error

#### 1. 功能块说明

启动各轴的绝对单段速运动。运转中，由启动速度  $V_{BIAS}$  加速至运转速度  $Velocity$  后稳态运行，接近目标位置  $Position$  时，脉冲产生单元会开始减速至启动速度而后停。 $V_{bias}$  ·  $V_{MAX}$  ·  $T_{ACC}$  ·  $T_{DEC}$  设定可参考  $AxisSetting1$  功能块。输入引脚  $Position$  和  $Velocity$  预设单位为脉冲数以及每秒脉冲数，若需修改单位系需利用  $AxisSetting2$  的功能块修改。



#### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	10PM00D :K1~K6	Execute 上升沿时
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
Position	绝对位置	DWORD	K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿时
Velocity	目标速度	DWORD	K1 ~ K2,147,483,647	执行中持续更新

输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>定位完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若 Execute 为 FALSE，下个周期 Done 会变成 FALSE</li> </ul>

# 5 应用指令分类及基本使用

输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>被中断时·若 Execute 为 FALSE·下个周期 Aborted 会变成 FALSE</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> <li>启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>Execute下降沿时</li> </ul>

其中输入引脚 Distance 和 Velocity 单位预设 为脉冲数以及每秒脉冲数，若需修改单位系需利用参数设定的运动功能块修改。

### 3. 错误产生及排除

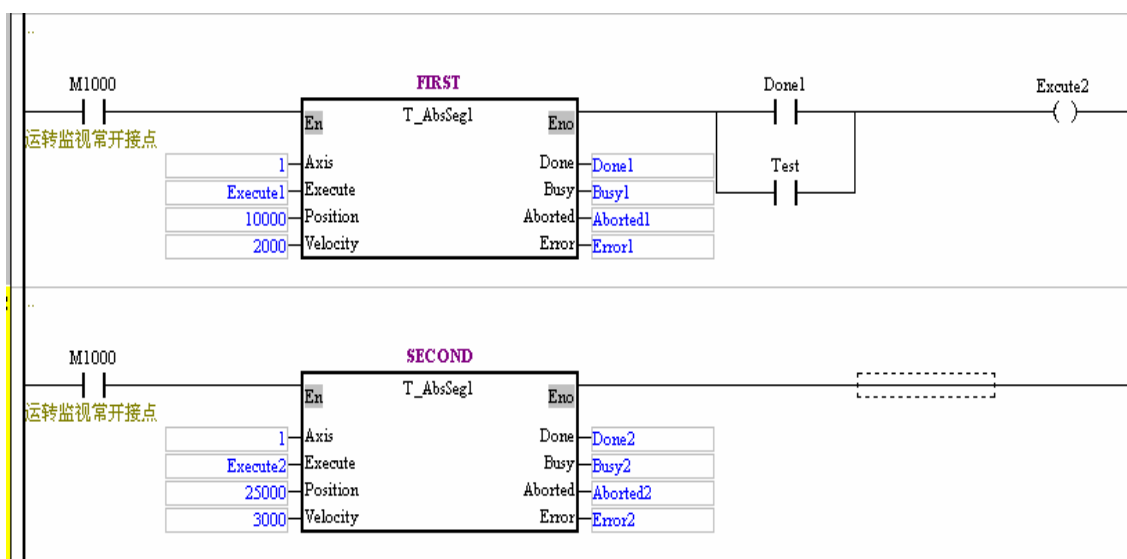
错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 范例程序

此范例的目的

- 第一个单段速运动完成后，连续执行第二个单段速运动。
- 第一个单段速运动尚未完成前，先执行第二个单段速运动。

首先设定运动功能块 FIRST 为设定第 1 轴以每秒 2000 脉冲的速度，运行至 10000 脉冲的位置，运动功能块 SECOND 为设定第 1 轴，以每秒 3000 脉冲的速度，运行至 25000 脉冲的位置。



- 第一个单段速运动完成后，连续执行第二个单段速运动。

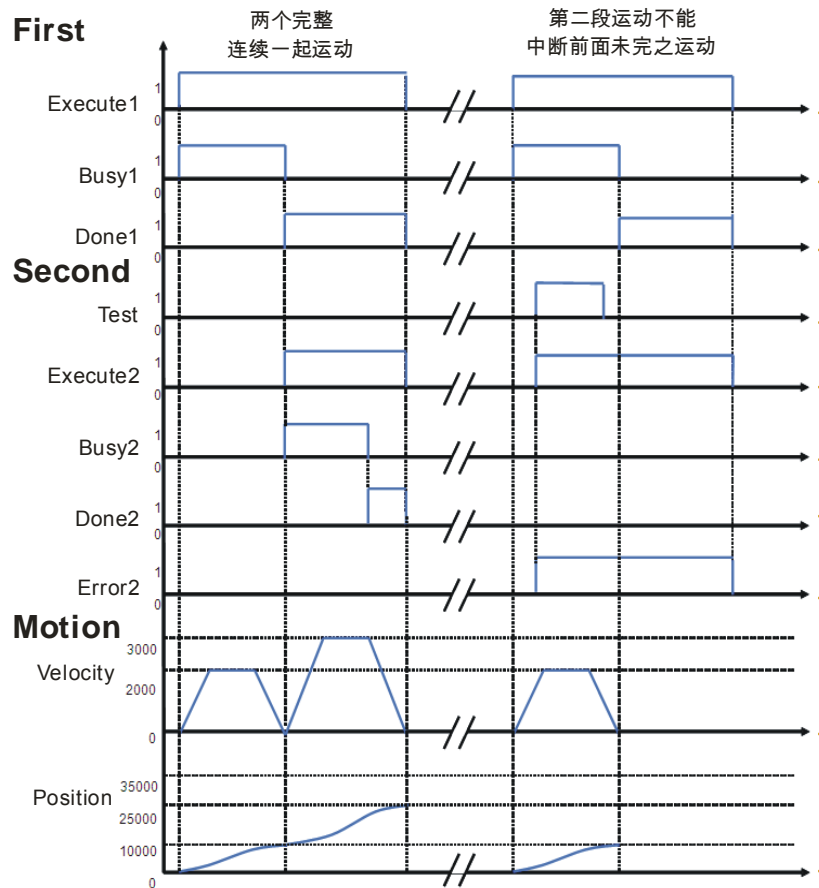
操作步骤：

- 启动 Execute1
- 等待 Done2 或 Error2 上升沿产生



- 第一个单段速运动尚未完成前，预先执行第二个单段速运动。  
操作步骤：  
(a) 启动 Execute1  
(b) 在 Busy1 为 True 时，启动 Test  
(c) 等待 Done2 或 Error2 上升沿产生

两种操作结果其时序图与第 1 轴运动结果如下：



- 第一个单段速运动完成后，连续执行第二个单段速运动。  
可以看到运动功能块 FIRST 动作完成后，轴执行运动功能块 SECOND，而最后轴的运动位置为 250000。
- 第一个单段速运动尚未完成前，先执行第二个单段速运动。  
可看到在 Error2 出现后，轴的最后位置为 10000，运动功能块 SECOND 的命令无效。

## 5. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

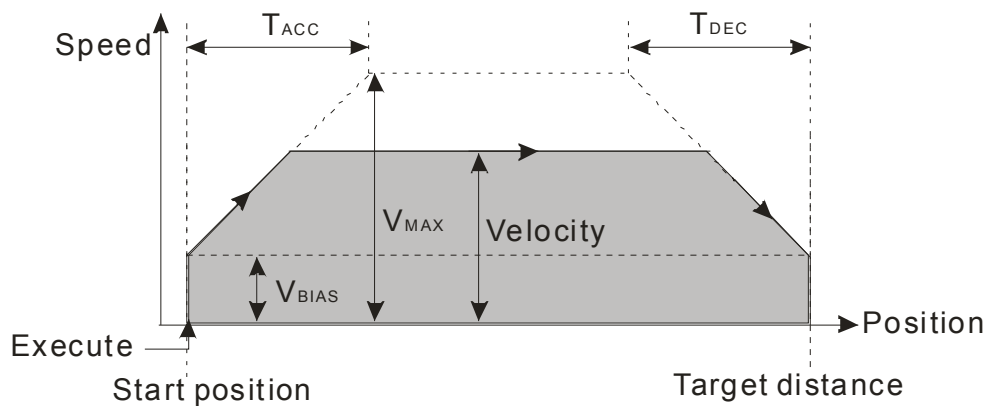
# 5 应用指令分类及基本使用

## 5.10.2 相对单段速运动

En	T_RelSeg1	Eno
Axis		Done
Execute		Busy
Distance		Aborted
Velocity		Error

### 1. 功能块说明

启动各轴的相对单段速运动。启动各轴的绝对单段速运动。运转中，由启动速度  $V_{BIAS}$  加速至运转速度  $Velocity$  后稳态运行，接近目标位置  $Distance$  时，脉冲产生单元会开始减速至启动速度而后停。 $V_{bias}$  ·  $V_{MAX}$  ·  $T_{ACC}$  ·  $T_{DEC}$  设定可参考 `AxisSetting1` 功能块。输入引脚 `Distance` 和 `Velocity` 预设单位为脉冲数以及每秒脉冲数，若需修改单位系需利用 `AxisSetting2` 的功能块修改。



### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
Distance	相对移位	DWORD	K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿时
Velocity	目标速度	DWORD	K1 ~ K2,147,483,647	执行中持续更新

输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	定位完成时	Execute 下降沿时 定位完成时若 Execute 为 FALSE，下个周期 Done 会变成 FALSE
Busy	执行中	BOOL	Execute 上升沿时	Done 上升沿时 Error 上升沿时 Aborted 上升沿时

输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Aborted	功能块被其它命令中断	BOOL	执行中被命令停止	Execute 下降沿时 被中断时·若 Execute 为 FALSE·下个周期 Aborted 会变成 FALSE
Error	功能块产生错误	BOOL	输入参数错误 启动时设定轴已在运动状态	Execute 下降沿时

其中输入引脚 **Distance** 和 **Velocity** 单位预设分别为脉冲数以及每秒脉冲数·若需修改单位系需利用参数设定的运动功能块修改。

### 3. 错误产生及排除

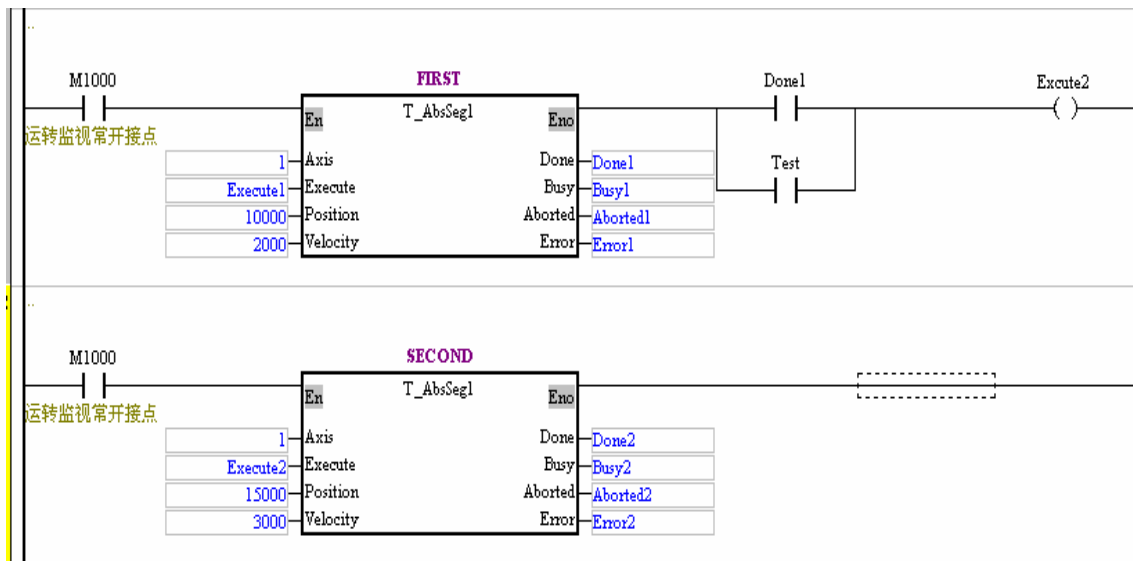
错误状况	排除方式
输入数值错误	检查输入引脚输入数值·是否超过可允许范围
功能块动作冲突	确认功能块在启动时·其它单轴运动功能块为未启动或已完成

### 4. 范例程序

此范例的目的

- 第一个单段速运动完成后·连续执行第二个单段速运动。
- 第一个单段速运动尚未完成前·先执行第二个单段速运动。

首先设定运动功能块 **FIRST** 为第 1 轴·以每秒 2000 脉冲的速度·运行 10000 个脉冲的距离·运动功能块 **SECOND** 为设定第 1 轴·以每秒 3000 脉冲的速度·运行至 15000 个脉冲的距离。



- 第一个单段速运动完成后·连续执行第二个单段速运动。

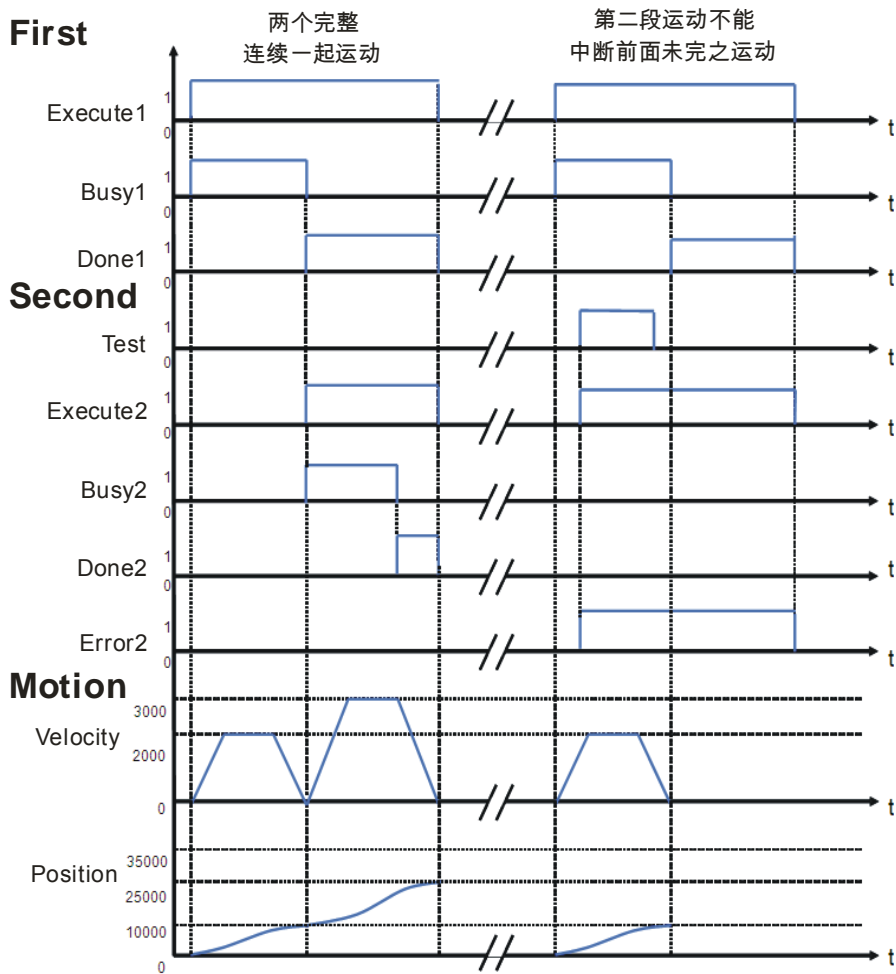
操作步骤：

- 启动 Execute1
- 等待 Done2 或 Error2 上升沿产生

# 5 应用指令分类及基本使用

- 第一个单段速运动尚未完成前，预先执行第二个单段速运动。  
操作步骤：
  - (a) 启动 Execute1
  - (b) 在 Busy1 为 True 时，启动 Test。
  - (c) 等待 Done2 或 Error2 上升沿产生

两种操作结果其时序图与第 1 轴运动结果如下：



- 第一个单段速运动完成后，连续执行第二个单段速运动。  
可以看到运动功能块 FIRST 动作完成后，轴移动到 10000，而后轴执行运动功能块 SECOND，轴以 10000 无开始移动 15000 的距离，其最后位置为 250000。
- 第一个单段速运动尚未完成前，先执行第二个单段速运动。  
可看到在 Error2 出现后，轴的最后位置为 10000，运动功能块 SECOND 的命令无效。

## 5. 支持模块

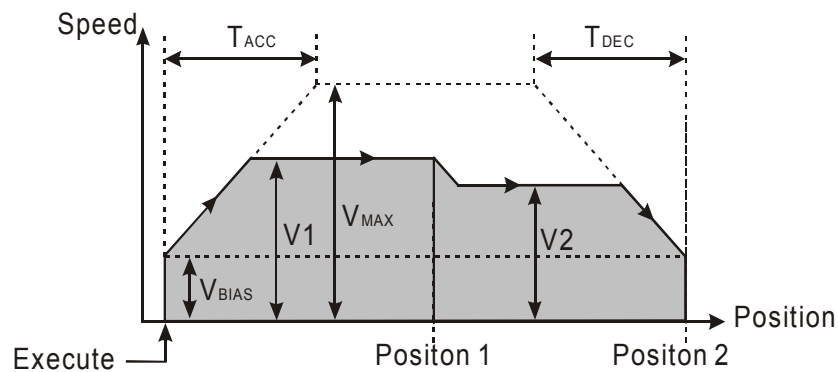
此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

## 5.10.3 绝对两段速运动

En	T_AbsSeg2	Eno
Axis		Done
Execute		Busy
Position1		Aborted
Velocity1		Error
Position2		
Velocity2		

### 1. 功能块说明

启动各轴的绝对两段速运动。由启动速度  $V_{BIAS}$  加速至运转速度  $V_1$  稳态运行，直到到达目标位置  $Position1$  的脉冲数后，此时会由运转速度  $V_1$  加速/减速至运转速度  $V_2$  稳态运行，当当前位置接近目标位置  $Position2$  时减速至启动速度  $V_{BIAS}$  而后停止。 $Position2$  设定需大于  $Position1$ 。 $V_{BIAS}$ 、 $V_{MAX}$ 、 $T_{ACC}$ 、 $T_{DEC}$  设定可参考  $AxisSetting1$  功能块。输入引脚  $Position1$ 、 $Position2$ 、 $Velocity1$  和  $Velocity2$  预设单位为脉冲数以及每秒脉冲数，若需修改单位系需利用  $AxisSetting2$  的功能块修改。



### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
Position1	第一段绝对位置	DWORD	K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿时
Velocity2	第一段目标速度	DWORD	K1 ~ K2,147,483,647	Execute 上升沿时
Position1	第二段绝对位置	DWORD	K-2,147,483,648 ~ K2,147,483,647 且需同时大于或同时小于当前位置和 Position1	Execute 上升沿时
Velocity2	第二段目标速度	DWORD	K1 ~ K2,147,483,647	Execute 上升沿时

# 5 应用指令分类及基本使用

输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>定位完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若 Execute 为 FALSE，下个周期 Done 会变成 FALSE</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error 上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>被中断时，若 Execute 为 FALSE，下个周期 Aborted 会变成 FALSE</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> <li>启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> </ul>

其中输入引脚 Position1 和 Position2 单位预设 为脉冲数，而 Velocity1 和 Velocity2 为每秒脉冲数，若需修改单位系需利用参数设定的运动功能块修改。

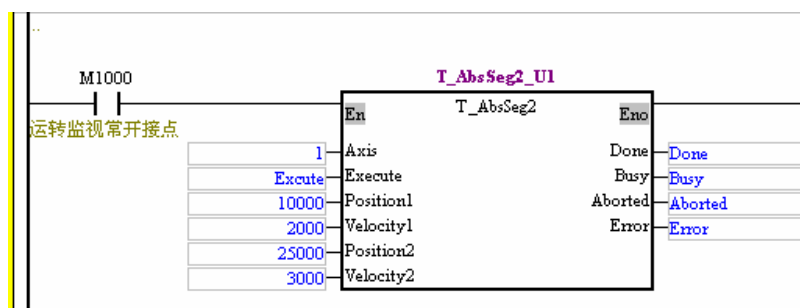
### 3. 错误产生及排除

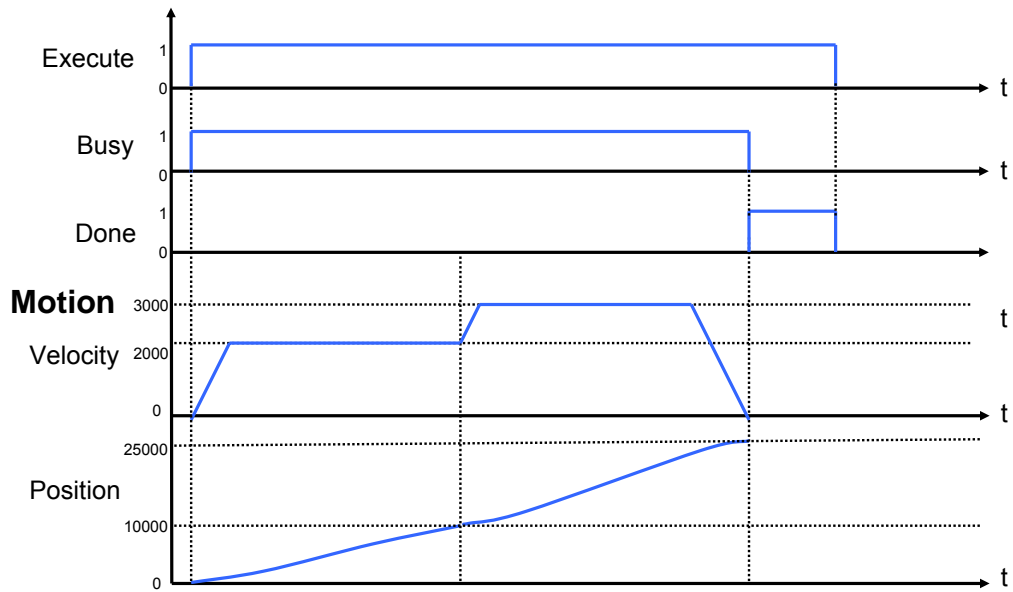
错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 范例程序

此范例的目的

- 利用绝对两段速功能块，操作指定的运动轴作绝对位置的两段速运动。  
首先设定第一段运行的速度为每秒 2000 脉冲，运行的目标位置距为 10000 个脉冲，而第二段绝对位置的运行参数为每秒 3000 脉冲的速度，运行至 25000 个脉冲的目标位置。





在运动功能块启动后，第一轴先以目标速度每秒 2000 脉冲运行到目标位置 10000 处，而后再以第二段目标速度每秒 3000 脉冲运行到目标位置 25000 处。

## 5. 支持模块

此运动功能块支持 DVP10PM00M 一机种 DVP 之运动控制模块。

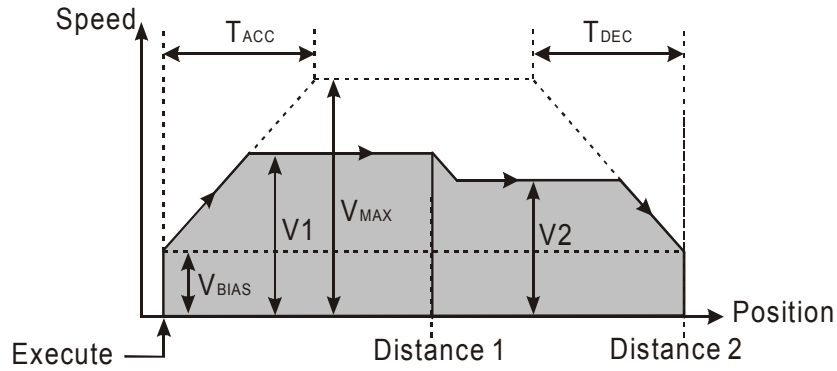
### 5.10.4 相对两段速运动

En	T_RelSeg2	Eno
Axis		Done
Execute		Busy
Distance1		Aborted
Velocity1		Error
Distance2		
Velocity2		

#### 1. 功能块说明

启动各轴的相对两段速运动。由启动速度 VBIAS 加速至运转速度 Velocity1 稳态运行，直到输出目标位置 Distance1 的脉冲数后，此时会由运转速度 Velocity1 加速/减速至运转速度 Velocity2 稳态运行，当当前位置接近目标位置 Distance2 时减速至启动速度 VBIAS 而后停止。Distance2 设定需与 Distance1 正负符号相同。Vbias, VMAX, TACC, TDEC 设定可参考 AxisSetting1 功能块。输入引脚 Distance1, Distance2, Velocity1 和 Velocity2 预设单位为脉冲数以及每秒脉冲数，若需修改单位系需利用 AxisSetting2 的功能块修改。

# 5 应用指令分类及基本使用



## 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
Distance1	第一段相对移位	DWORD	K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿时
Velocity1	第一段目标速度	DWORD	K1 ~ K2,147,483,647	Execute 上升沿时
Distance2	第二段相对移位	DWORD	K-2,147,483,648 ~ K2,147,483,647 且需与 Distance1 同号	Execute 上升沿时
Velocity1	第二段目标速度	DWORD	K1 ~ K2,147,483,647	Execute 上升沿时

输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>定位完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若 Execute 为 FALSE · 下个周期 Done 会变成 FALSE</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>被中断时 · 若 Execute 为 FALSE · 下个周期 Aborted 会变成 FALSE</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> <li>启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>Execute下降沿时</li> </ul>



其中输入引脚 Distance1 和 Distance 2 单位预设为脉冲数，而 Velocity1 和 Velocity2 为每秒脉冲数，若需修改单位系需利用参数设定的运动功能块修改。

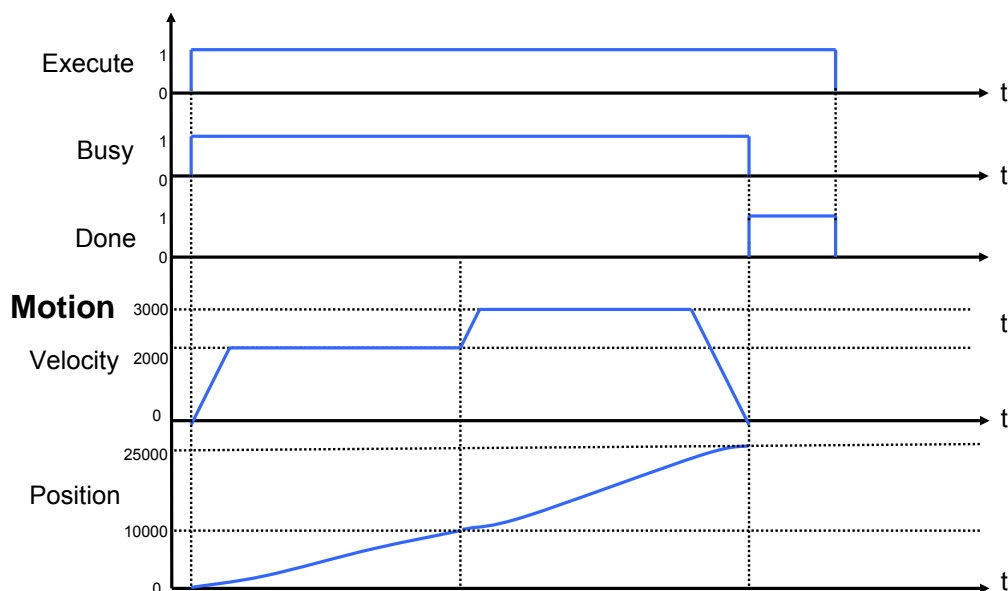
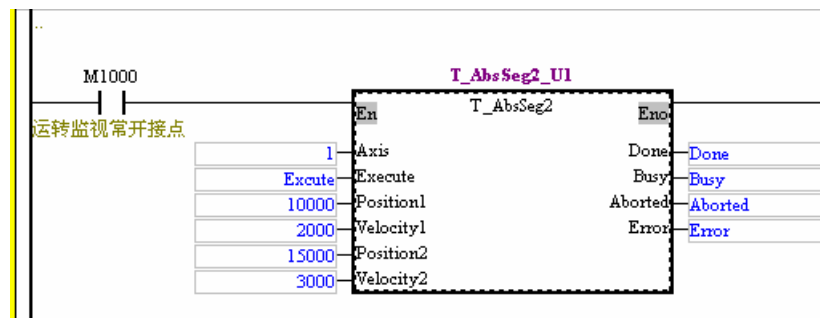
### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 范例程序

此范例的目的

- 利用相对两段速功能块，操作指定的运动轴作相对移位的两段速运动。  
首先设定第一段运行的速度为每秒 2000 脉冲，运行的距离为 10000 个脉冲，而第二段绝对位置的运行参数为每秒 3000 脉冲的速度，运行的距离为 15000 个脉冲。



在运动功能块启动后，第一轴先以目标速度每秒 2000 脉冲运行 10000 脉冲的距离，而后再以第二段目标速度每秒 3000 脉冲运行 25000 个脉冲的距离。

### 5. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

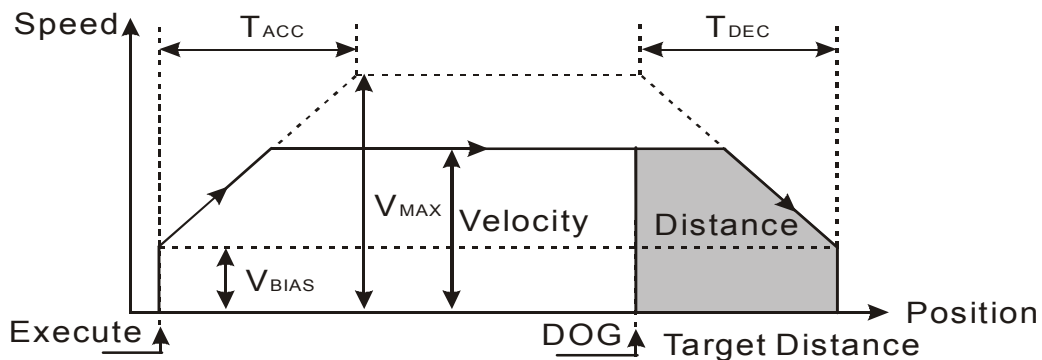
# 5 应用指令分类及基本使用

## 5.10.5 插入单段速运动

En	T_TxSeg1	Eno
Axis		Done
Execute		Busy
DogEdge		Aborted
Distance		Error
Velocity		

### 1. 功能块说明

运转速度会由启动速度  $V_{BIAS}$  开始加速至运转速度  $Velocity$  稳态运行，于运行期间遇到 DOG 近点信号触发时，会再送出位置寄存器  $Distance$  内容值之脉冲数，当脉冲个数直到快到达位置寄存器  $Distance$  之内容值时，减速至启动速度而后停。 $V_{bias}$ 、 $V_{MAX}$ 、 $T_{ACC}$ 、 $T_{DEC}$  设定可参考  $AxisSetting1$  功能块。其中输入引脚  $Distance$  单位预设脉冲数，而  $Velocity$  为每秒脉冲数，若需修改单位系需利用  $AxisSetting2$  功能块修改。而  $DogEdge$  可利用变量  $mcRising$  和  $mcFalling$  来设定启动信号为 Dog 的上升沿或下降沿。



### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
DogEdge	近点触发方向	BOOL	mcRising ( TRUE ) / mcFalling ( FALSE )	Execute 上升沿时
Distance	Dog 触发后插入的移位	DWORD	K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿时
Velocity	目标速度	DWORD	K1 ~ K2,147,483,647	Execute 上升沿时

输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>定位完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若 Execute 为 FALSE，下个周期 Done 会变成 FALSE</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>被中断时，若 Execute 为 FALSE，下个周期 Aborted 会变成 FALSE</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> <li>启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>Execute下降沿时</li> </ul>

其中输入引脚 **Distance** 单位预设 为脉冲数，而 **Velocity** 为每秒脉冲数，若需修改单位系需利用参数设定的运动功能块修改。而 **DogEdge** 可利用变量 **mcRising** 和 **mcFalling** 来设定启动信号为 **Dog** 的上升沿或下降沿。

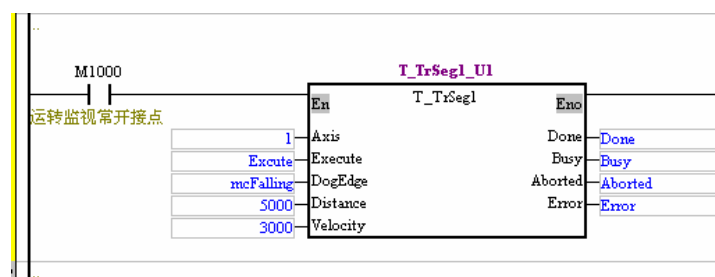
### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

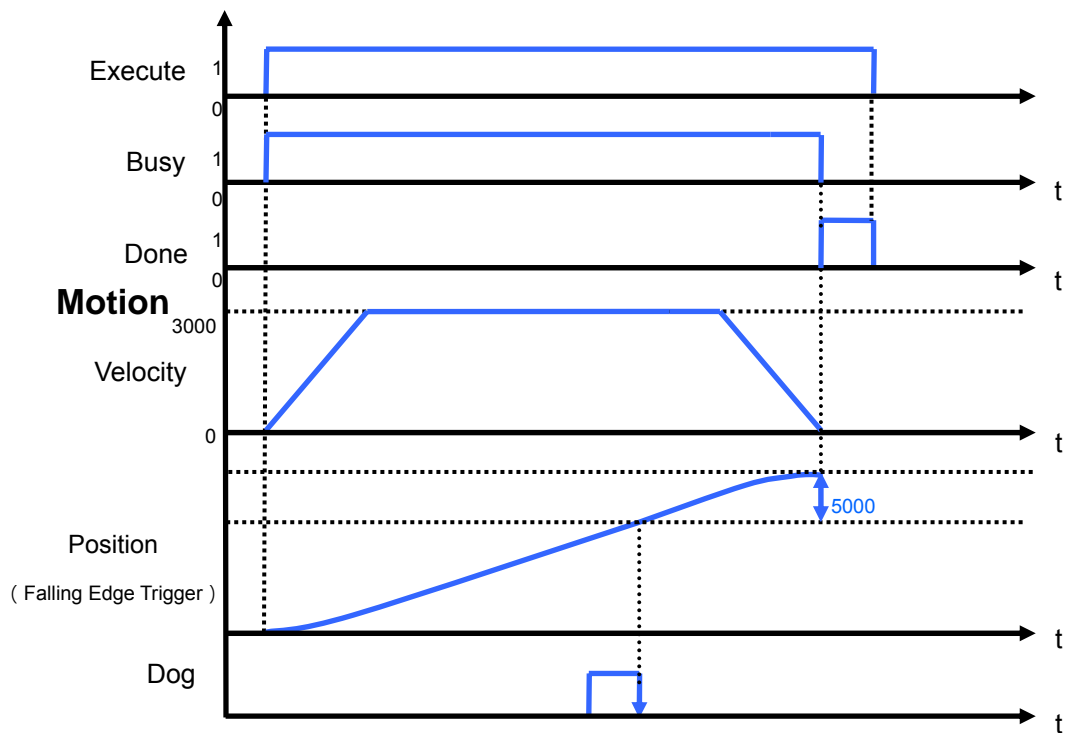
### 4. 范例程序

范例一：

- 利用插入单段速功能块，操作指定的运动轴利用 **Dog** 下降沿作单段速运动的起始信号。首先设运行的速度为每秒 3000 脉冲，而遇到 **Dog** 的下降沿开始运行的距离为 5000 个脉冲，可看到运动轨迹再遇到 **Dog** 下降沿后，运行 5000 个脉冲，等到 5000 个脉冲的距离完成后，**Done** 信号被设定。

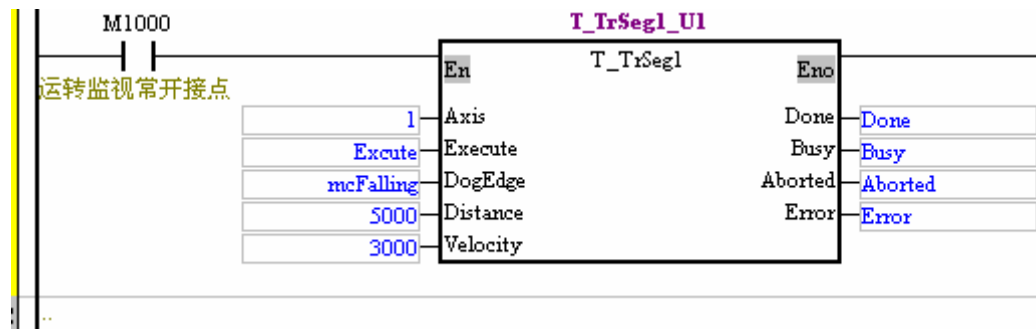


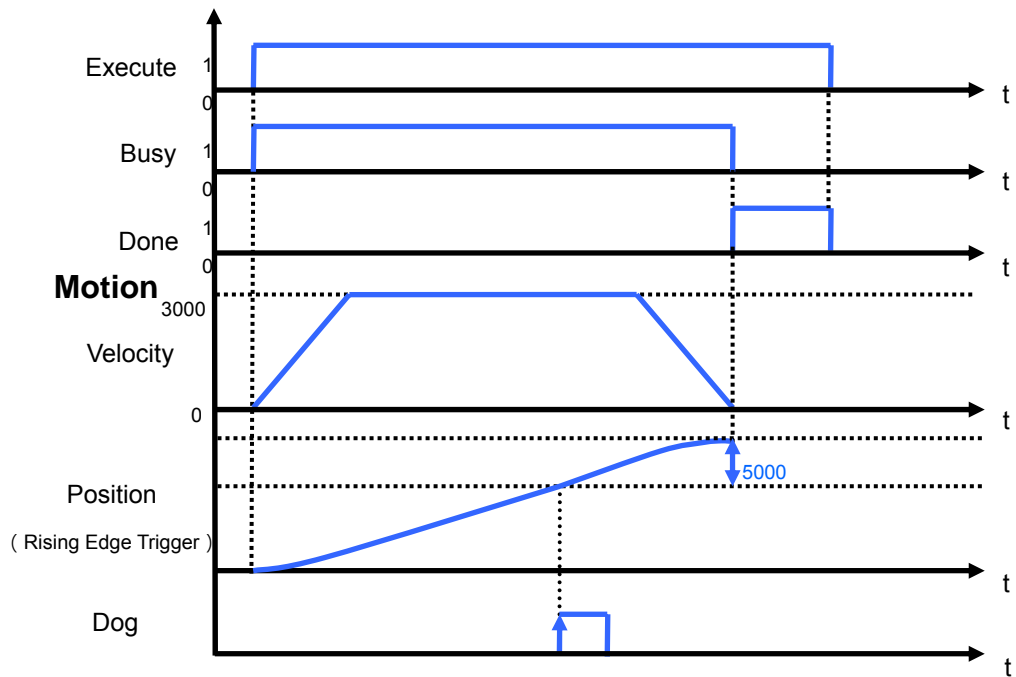
# 5 应用指令分类及基本使用



范例二：

- 利用插入单段速功能块，操作指定的运动轴利用 Dog 上升沿作单段速运动的切换信号。首先设运行的速度为每秒 3000 脉冲，而遇到 Dog 的上升沿开始运行的距离为 5000 个脉冲，可看到运动轨迹再遇到 Dog 上升沿后，运行 5000 个脉冲，等到 5000 个脉冲的距离完成后，Done 信号被设定。





## 5. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

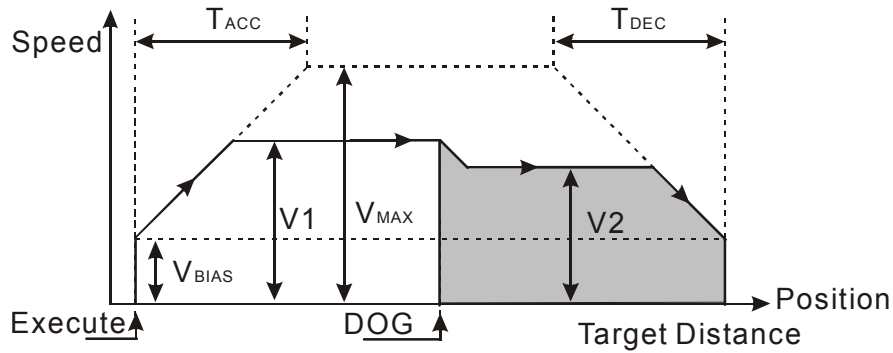
### 5.10.6 插入两段速运动

En	T_TrSeg2	Eno
Axis		Done
Execute		Busy
DogEdge		Aborted
Velocity1		Error
Distance		
Velocity2		

#### 1. 功能块说明

启动各轴的插入两段速运动。运转中，由启动速度  $V_{BIAS}$  开始加速至运转速度  $V_1$  后稳态运行，当外部  $DOG$  信号触发时，立即加速/减速至运转速度  $V_2$  后稳态运行，直到输出距离接近所设定的  $Distance$  后，减速启动速度而后停。 $V_{bias}$ 、 $V_{MAX}$ 、 $TACC$ 、 $TDEC$  设定可参考  $AxisSetting1$  功能块。输入引脚  $Distance$  单位预设 为脉冲数，而  $Velocity1$  和  $Velocity2$  为每秒脉冲数，若需修改单位系需利用  $AxisSetting2$  功能块修改。 $DogEdge$  可利用变量  $mcRising$  和  $mcFalling$  来设定启动信号为  $Dog$  的上升沿或下降沿。

# 5 应用指令分类及基本使用



## 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
DogEdge	近点触发方向	BOOL	mcRising ( TRUE ) / mcFalling ( FALSE )	Execute 上升沿时
Velocity1	Dog 触发前之目标速度	DWORD	K1 ~ K2,147,483,647	Execute 上升沿时
Distance	Dog 触发后插入的移位	DWORD	K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿时
Velocity2	Dog 触发后之目标速度	DWORD	K1 ~ K2,147,483,647	Execute 上升沿时

输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>定位完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若 Execute 为 FALSE，下个周期 Done 会变成 FALSE</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error 上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>被中断时，若 Execute 为 FALSE，下个周期 Aborted 会变成 FALSE</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> <li>启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> </ul>

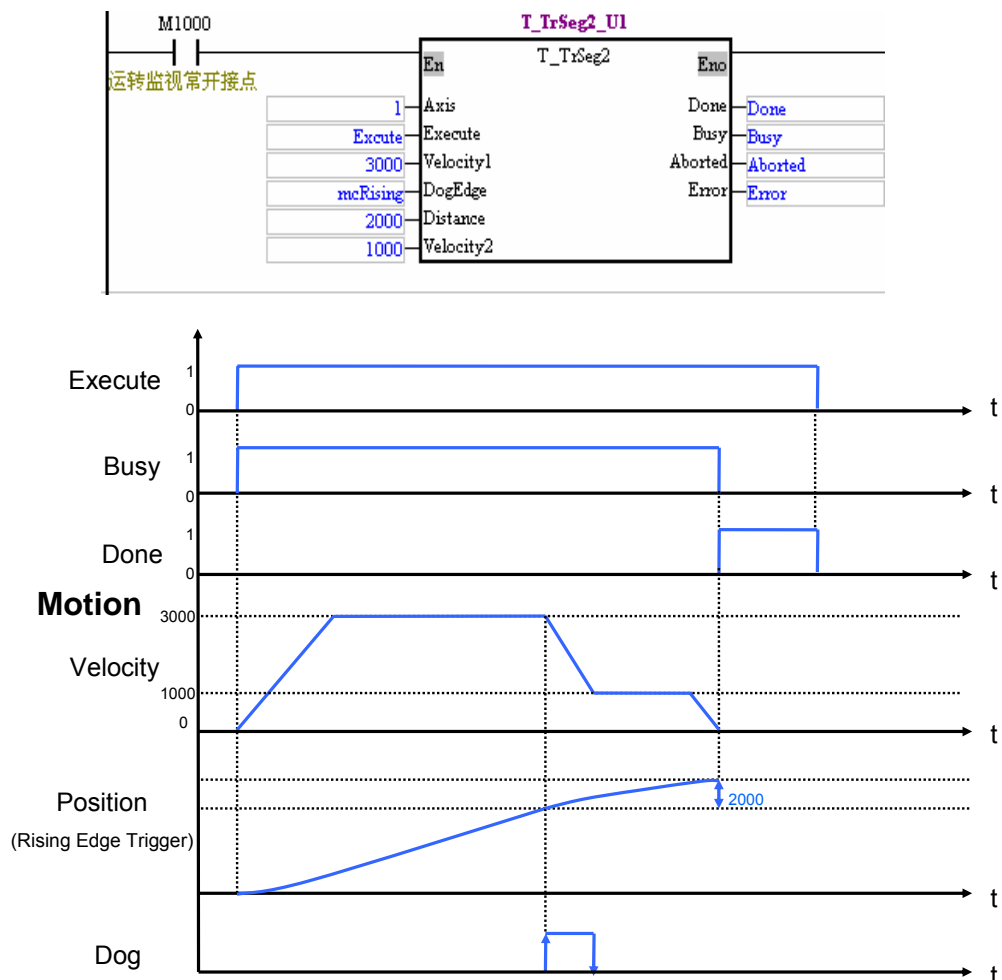
其中输入引脚 **Distance** 单位预设为脉冲数，而 **Velocity1** 和 **Velocity2** 为每秒脉冲数，若需修改单位系需利用参数设定的运动功能块修改。而 **DogEdge** 可利用变量 **mcRising** 和 **mcFalling** 来设定启动信号为 **Dog** 的上升沿或下降沿。

### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 范例程序

利用插入两段速功能块，操作指定的运动轴利用 **Dog** 上升沿作两段速运动的切换信号。首先设运行的速度为每秒 3000 脉冲，而遇到 **Dog** 的上升沿开始以每秒 1000 脉冲的速度，运行 5000 个脉冲的距离。



由时序图可得知运动轨迹再遇到 **Dog** 上升沿前，先以每秒 3000 脉冲的速度运转，而后遇到 **Dog** 上升沿，以每秒 1000 个脉冲，运行 2000 个脉冲的距离，等到距离到达后，**Done** 信号被设定。

### 5. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

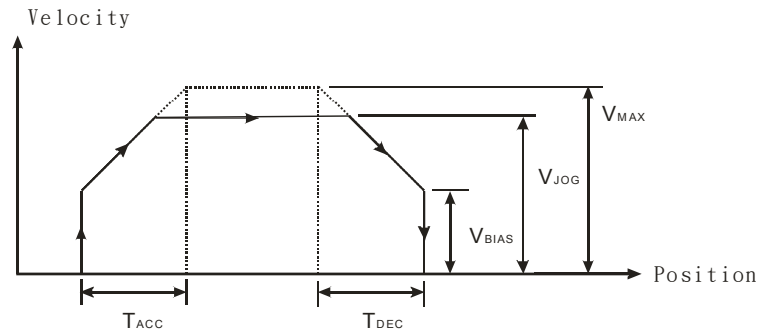
# 5 应用指令分类及基本使用

## 5.10.7 寸动运动

En	T_Jog	Eno
Axis		Busy
PositiveEnable		Aborted
NegativeEnable		Error
Velocity		

### 1. 功能块说明

启动各轴的寸动运动，利用 **Axis** 的输入数据指定某一轴，利用 **Velocity** 的输入值做为寸动的运行速度，而后依照 **PositiveEnable** 和 **NegativeEnable** 的输入值上升沿做为运动启动设定，其中 **PositiveEnable** 为正向的寸动运动，**NegativeEnable** 为反向的寸动运动。输入引脚 **Velocity** 为每秒脉冲数，若需修改单位系需利用 **AxisSetting2** 的运动功能块修改。



### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
PositiveEnable	正向寸动运动允许	BOOL	TRUE / FALSE	<ul style="list-style-type: none"> <li>● 正反向同时上升沿触发时 正向优先于反向，同时反向允许会被 Reset 到 FALSE。</li> <li>● 仅 PositiveEnable 上升沿触发时，若 Enable-为 TRUE 则 NegativeEnable 会被 Reset 到 FALSE-同时由反向运动减速到停止后再转成正向运动。</li> </ul>
NegativeEnable	反向寸动运动允许	BOOL	TRUE / FALSE	<ul style="list-style-type: none"> <li>● 正反向同时上升沿触发时 正向优先于反向，同时反向允许会被 Reset 到 FALSE。</li> <li>● 仅 Enable-上升沿触发时，若 PositiveEnable 为 TRUE 则 NegativeEnable 会被 Reset 到 FALSE-，同时由正向运动减速到停止后再转成反向运动。</li> </ul>
Velocity	目标速度	DWORD	K1 ~ K2,147,483,647	执行中持续更新



输出引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>PositiveEnable上升沿或</li> <li>NegativeEnable上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>运动停止时</li> <li>Error上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>PositiveEnable 下降沿或 NegativeEnable 下降沿时</li> <li>被中断时·若 PositiveEnable 和 NegativeEnable 为 FALSE·下个周期 Aborted 会变成 FALSE</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> <li>启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>PositiveEnable下降沿或 NegativeEnable下降沿时</li> </ul>

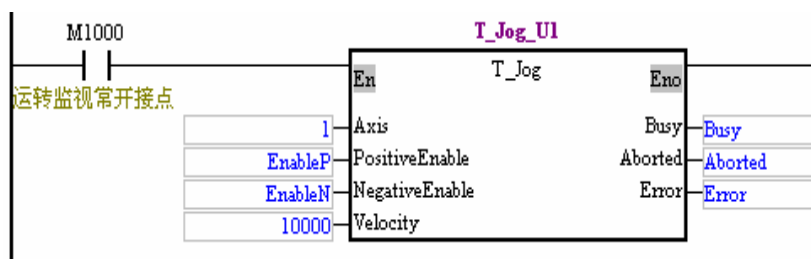
其中输入引脚 Velocity 为每秒脉冲数，若需修改单位系需利用参数设定的运动功能块修改。

### 3. 错误产生及排除

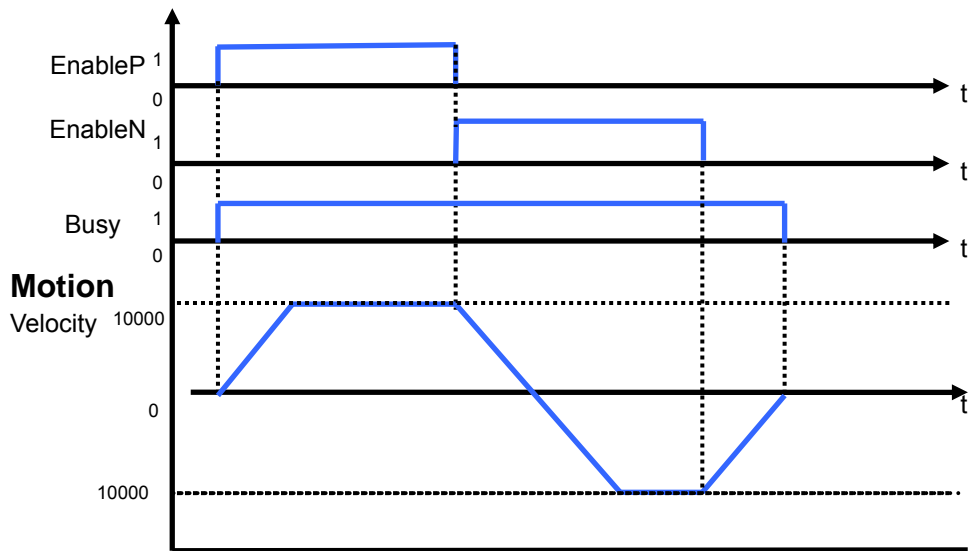
错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 范例程序

利用寸动功能块，操作指定的运动轴，利用 Enable+操作正向寸动，利用 Enable-操作反向寸动。首先设运行的速度为每秒 10000 脉冲，而利用操作 Enable+和 Enable-为 1 来控制第一轴作正向寸动与反向寸动。



# 5 应用指令分类及基本使用



当 Enable+ 为 1 时，第一轴以每秒 10000 脉冲的速度运转，作正向寸动，而当 Enable- 为 1 时，第一轴以每秒 10000 个脉冲作反向寸动，当两者均非 1 时第一轴停止运动。

## 5. 支持模块

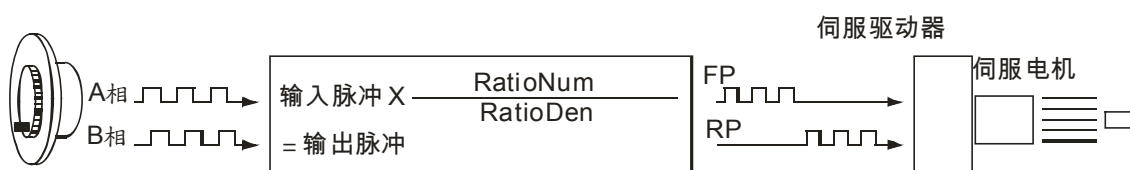
此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

### 5.10.8 手摇轮模式运动

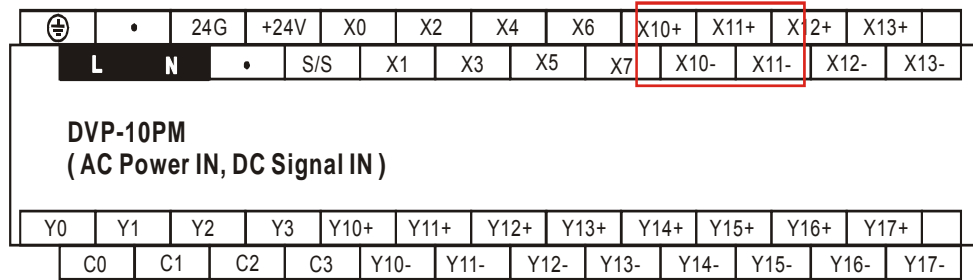
En	T_MPG	Eno
Axis		Valid
Enable		Busy
Reset		Aborted
RatioNum		Error
RatioDen		InputPulses
		InputFreq

#### 1. 功能块说明

此运动功能块，目的在启动手摇轮模式运动，利用 Axis 的输入数据指定某一轴，此轴的运动为跟随外部手摇轮动作，而轴位置与外部手轮输入的比例关系，由 RatioNum 和 RatioDen 决定，手摇轮响应速度会依照 TACC, TDEC 来执行，TACC, TDEC 可利用 AxisSetting1 功能块作设定。



下图说明手摇轮输入组别：



方框部份为第 1~6 轴手摇轮输入

## 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Enable 上升沿时
Enable	手摇轮模式 运动允许	BOOL	TRUE / FALSE	-
Reset	允许前重置 手摇轮计数 量	BOOL	TRUE / FALSE	Enable 上升沿时
RatioNum	电子齿轮比 之分子	DWORD	K0 ~ K32,767	执行中持续更新
RatioDen	电子齿轮比 之分母	DWORD	K1 ~ K32,767	执行中持续更新

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	运动完成	BOOL	<ul style="list-style-type: none"> <li>• Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>• 运动停止时</li> <li>• Error上升沿时</li> <li>• Aborted 上升沿时</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>• 执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>• Enable 下降沿时</li> <li>• 被中断时 · 若 Enable 为 FALSE · 下个周期 Aborted 会变成 FALSE</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>• 执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>• Enable 下降沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>• 输入参数错误</li> <li>• 启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>• Enable下降沿时</li> </ul>

## 5 应用指令分类及基本使用

输出数值引脚				
引脚名称	功能说明	数据格式	输出范围	更新类型
InputPulses	累计手摇轮输入脉冲个数	DWORD	K-2,147,483,648 ~ K2,147,483,647	执行中持续更新
InputFreq	手摇轮输入频率	DWORD	K 0 ~ K 2,147,483,647	执行中持续更新

### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

#### 5.10.9 电子齿轮运动

En	T_GearIn	Eno
Axis		Valid
Enable		Busy
Reset		Aborted
RatioNum		Error
RatioDen		InputPulses
		InputFreq

#### 1. 功能块说明

此运动功能块，旨在启动电子齿轮运动。在此动作过程中，RatioNum 和 RatioDen 决定从轴跟随主轴齿轮比的分子与分母，而 Reset 信号做为主轴累计输入脉冲数的清除信号。齿轮的响应速度不会依照 TACC, TDEC 作执行，直接根据输入来源来作运行。电子齿轮输入组别与手摇轮相同。

#### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	从轴编号	WORD	K1~K6	Enable 上升沿时
Enable	齿轮啮合运动允许	BOOL	TRUE / FALSE	-
Reset	允许时重置 InputPulses	BOOL	TRUE / FALSE	Enable 上升沿时
RatioNum	电子齿轮比之分子	DWORD	K-32,767 ~ K32,767	执行中持续更新
RatioDen	电子齿轮比之分母	DWORD	K1 ~ K32,767	执行中持续更新

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	输出之数值有效	BOOL	<ul style="list-style-type: none"> <li>• Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>• 运动停止时</li> <li>• Error上升沿时</li> <li>• Aborted 上升沿时</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>• 执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>• Enable 下降沿时</li> <li>• 被中断时，若 Enable 为 FALSE，下个周期 Aborted 会变成 FALSE</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>• 执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>• Enable下降沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>• 输入参数错误</li> <li>• 启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>• Enable下降沿时</li> </ul>

输出数值引脚				
引脚名称	功能说明	数据格式	输出范围	更新类型
InputPulses	累计主轴输入脉冲个数	DWORD	K-2,147,483,648 ~ K2,147,483,647	执行中持续更新
InputFreq	主轴输入脉冲频率	DWORD	K 0 ~ K 2,147,483,647	执行中持续更新

### 3. 错误产生及排除：

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

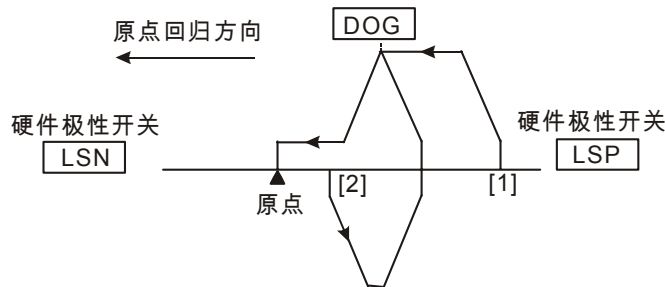
# 5 应用指令分类及基本使用

## 5.10.10 原点回归运动

En	T_HomeReturn	Eno
Axis		Done
Execute		Busy
Direction		Aborted
DogEdge		Error
HomePosition		
VRT		
VCR		
Signal_N		
Distance_P		

### 1. 功能块说明

此运动功能块，旨在启动原点回归运动，利用 Axis 的输入数据指定某一轴，依照 Direction 决定原点回归方向，VRT 决定原点回归的速度，而 DogEdge 决定遇到原点以上升沿或下降沿为触发信号，再遇到原点以后以 VCR 做为原点到零点速度，零点信号的个数以 Signal\_N 决定，若有补充的脉冲数需增加则利用 Distance\_P 大小决定，原点回归动作完成后以 Home Position 的数值做为该轴的当前位置。下图为回原点示意图：



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。

备注\*：位置(2)不支持第 5 轴与第 6 轴。

### 2. 输入/输出说明：

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
Direction	原点回归方向切换	BOOL	mcNegative ( FALSE ) / mcPositive ( TRUE )	Execute 上升沿时
DogEdge	Dog 近点触发方向	BOOL	mcFalling ( FALSE ) / mcRising ( TRUE )	Execute 上升沿时
Home Position	原点位置定义	DWORD	K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿时

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
VRT	原点回归速度	DWORD	K1 ~ K1000000	Execute 上升沿时
VCR	原点回归减速速度	DWORD	K1 ~ VRT	Execute 上升沿时
Signal_N	零点信号数	WORD	K0 ~ K32,767	Execute 上升沿时
Distance_P	补充脉冲数	WORD	K0 ~ K32,767	Execute 上升沿时

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>原点回归完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若Execute为FALSE，下个周期Done会变成FALSE</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Aborted	功能块被其它命令中断	BOOL	<ul style="list-style-type: none"> <li>执行中被命令停止</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>被中断时，若Execute为FALSE，下个周期Aborted会变成FALSE</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> <li>启动时设定轴已在运动状态</li> </ul>	<ul style="list-style-type: none"> <li>Execute下降沿时</li> </ul>

### 3. 错误产生及排除

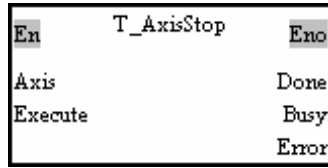
错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 支持模块

此运动功能块支持 DVP10PM00M 此一种 DVP 机种之运动控制模块。

# 5 应用指令分类及基本使用

## 5.10.11 单轴运动停止



### 1. 功能块说明

此运动功能块，目的在停止运动，利用 Axis 的输入数据指定某一轴，将其运动动作强制停止。

### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>原点回归完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若 Execute 为 FALSE，下个周期 Done 会变成 FALSE。</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error 上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> <li>该轴目前在非单轴、齿轮及凸轮之类别执行运动</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> </ul>

### 3. 错误产生及排除

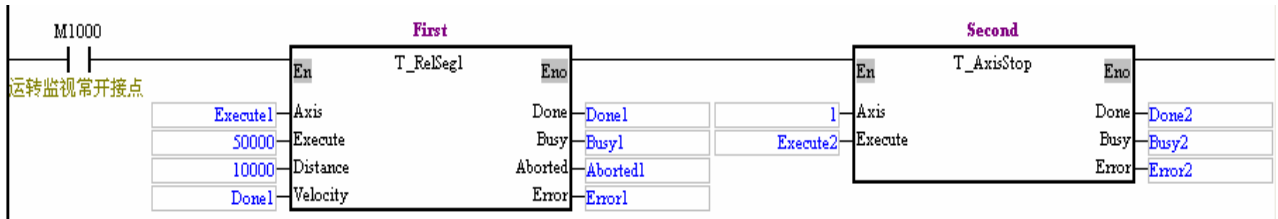
错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

### 4. 范例程序

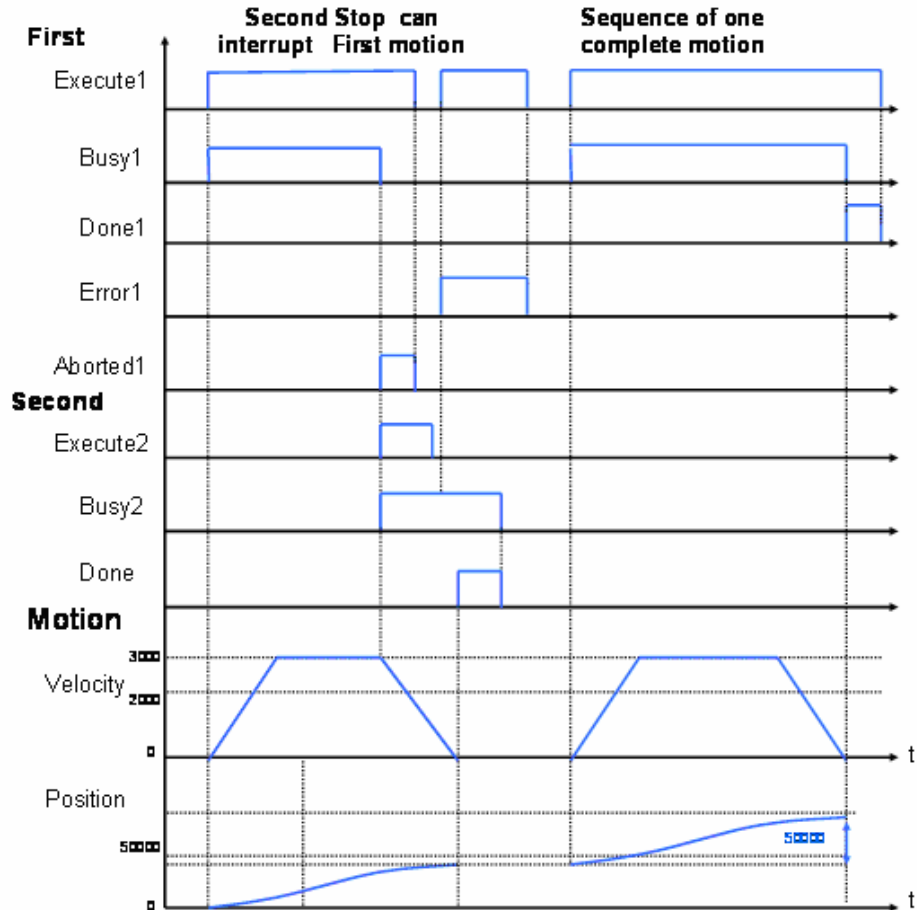
指定特定轴作相对单段速运动，再利用运动停止功能块将动作强制停止。

FIRST 为一个相对单段速运动之功能块，其中设定第 1 轴以每秒 10,000 脉冲的速度运行 50,000 脉冲的距离，SECOND 为运动停止功能块，所指定的运动轴也为第 1 轴。





启动 FIRST 运动功能块，在 Done1 上未为 1 时，利用 Execute2 启动 SECOND。



启动 FIRST 后，第 1 轴以目标速度每秒 10,000 脉冲运行，在 SECOND 启动后 FIRST 的 Aborted1 信号会被设置为 1，此时 Busy1 为 0，第 1 轴同时开始停止动作。再停止动作的过程中，不接受任何运动的启动，若启动其它运动动作均会发生错误信息。

## 5. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

### 5.10.12 运动轴参数设定 I

En	T_AxisSetting1	Eno
Axis		Done
Execute		Busy
Vmax		Error
Vbias		
Tacc		
Tdec		

#### 1. 功能块说明

# 5 应用指令分类及基本使用

此运动功能块，旨在设定运动相关参数，利用 Axis 的输入数据指定某一轴，设定此轴在运动过程中可以达到的最高速度，启动速度，加速时间及减速时间。AxisSetting2 单位系设定将影响 Vmax, Vbias 的单位。

## 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时 启动运动	BOOL	TRUE / FALSE	-
Vmax	最高速度	DWORD	K1 ~ K2,147,483,647	Execute 上升沿时
Vbias	启动速度	DWORD	K0 ~ K2,147,483,647	Execute 上升沿时
Tacc	加速时间 (单位:ms)	WORD	K0 ~ K32,767	Execute 上升沿时
Tdec	减速时间 (单位:ms)	WORD	K0 ~ K32,767	Execute 上升沿时

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>原点回归完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若 Execute 为 FALSE，下个周期 Done 会变成 FALSE。</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error 上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Error	功能块产生 错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> </ul>

## 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

## 4. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

## 5.10.13 运动轴参数设定 II

En	T_AxisSetting2	Eno
Axis		Done
Execute		Busy
Vcurve		Error
OutputType		
Unit		
PulseRev		
DistanceRev		

### 1. 功能块说明

此运动功能块，旨在设定运动相关参数，利用 Axis 的输入数据指定某一轴，设定此轴在运动过程中的速度曲线，控制脉冲的输出类型，以及目前运动过程所在的位置与速度的单位系设定，并且单位系的设定需同时有电机转一圈所需的脉冲数及电机转一圈的传送距离。

单位系可分为三种：电机单位、机械单位与复合单位，单位系统转换将依照所输入的电机转一圈所需脉冲数(PulseRev)与电机转一圈传送距离(DistanceRev)作换算，下表为各单位系于位置与速度相互对应之单位。

	电机单位	机械单位	复合单位
位置	pulse	um	um
	pulse	m deg	m deg
	pulse	10 <sup>-4</sup> inch	10 <sup>-4</sup> inch
速度	pulse/sec	cm/min	pulse/sec
	pulse/sec	10deg/min	pulse/sec
	pulse/sec	inch/min	pulse/sec

### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时 启动运动	BOOL	TRUE / FALSE	-
Vcure	速度曲线	BOOL	mcTrapezoid:FALSE mcSCurve:TRUE	Execute 上升沿时
OutputType	输出脉冲类型	WORD	mcUD:0 mcPD:1 mcAB:2 mc4AB:3	Execute 上升沿时
Unit	单位系设定	WORD	mcMotor:0 mcMachine:1 mcComp:2	Execute 上升沿时

# 5 应用指令分类及基本使用

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
PulseRev	电机转一圈所需脉冲数	WORD	K1 ~ K2,147,483,647	Execute 上升沿时
DistanceRev	电机转一圈之传送距离	WORD	K1 ~ K2,147,483,647	Execute 上升沿时

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	<ul style="list-style-type: none"> <li>原点回归完成时</li> </ul>	<ul style="list-style-type: none"> <li>Execute 下降沿时</li> <li>定位完成时若Execute为FALSE，下个周期Done会变成FALSE。</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Execute 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Done 上升沿时</li> <li>Error上升沿时</li> <li>Aborted 上升沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> </ul>	<ul style="list-style-type: none"> <li>Execute下降沿时</li> </ul>

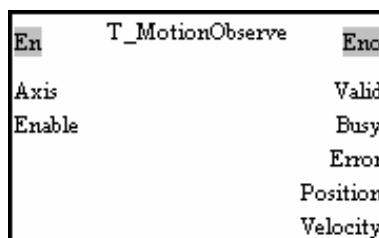
### 3. 错误产生及排除：

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

### 4. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

#### 5.10.14 读取当前位置/速度



#### 1. 功能块说明

此运动功能块，目的在读取轴当前位置与速度，利用 Axis 的输入数据指定某一轴，而在启动此功能块后，可透过 Position 得知此轴的当前位置，Velocity 得知此轴的当前速度。

#### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Enable 上升沿时
Enable	手摇轮模式 运动允许	BOOL	TRUE / FALSE	-

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	运动完成	BOOL	<ul style="list-style-type: none"> <li>Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Enable 下降沿时</li> <li>Error 上升沿时</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>Enable 下降沿时</li> <li>Error 上升沿时</li> </ul>
Error	功能块产生 错误	BOOL	<ul style="list-style-type: none"> <li>输入参数错误</li> </ul>	<ul style="list-style-type: none"> <li>Enable 下降沿时</li> </ul>

输出数值引脚				
引脚名称	功能说明	数据格式	输出范围	更新类型
Position	当前位置 ( Pulse Unit )	DWORD	K -2,147,483,648 ~ K 2,147,483,647	执行中持续更新
Velocity	当前速度 ( Pulse Unit )	DWORD	K 0 ~ K 2,147,483,647	执行中持续更新

### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

### 4. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

#### 5.10.15 轴状态信息

En	T_AxisStatus	Eno
Axis		Valid
Enable		Busy
ClearError		Error
		Mode
		AxisReady
		AxisError
		AxisErrorID

#### 1. 功能块说明

此运动功能块，目的为读取以及清除轴的错误状态，利用 Axis 的输入数据指定某一轴，透过 ClearError 清除轴的错误状态，轴目前的错误状态则透过 AxisErrorID 显示。

#### 2. 输入/输出说明

# 5 应用指令分类及基本使用

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Enable 上升沿时
Enable	读取允许	BOOL	TRUE / FALSE	-
ClearError	上升沿触发清除错误状态	BOOL	TRUE / FALSE	执行中

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	输出之数值有效	BOOL	<ul style="list-style-type: none"> <li>• Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>• Enable 下降沿时</li> <li>• Error 上升沿时</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>• Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>• Enable 下降沿时</li> <li>• Error 上升沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>• 输入参数错误</li> </ul>	<ul style="list-style-type: none"> <li>• Enable 下降沿时</li> </ul>

输出数值引脚				
引脚名称	功能说明	数据格式	输出范围	更新类型
Mode	运动模式	WORD	H0~H302 (*1)	执行中持续更新
AxisReady	运动轴待命标志	BOOL	TRUE / FALSE	执行中持续更新
AxisError	运动轴错误标志	BOOL	TRUE / FALSE	执行中持续更新
AxisErrorID	错误代码	WORD	H0002~HC4FF	执行中持续更新

\*1 : Mode 输入值定义如下

输入值	定义
H0	闲置
H100	单轴运动停止中
H101	绝对单段速定位运动
H102	相对单段速定位运动
H103	绝对单段速定位运动
H104	相对连续两段速定位运动
H105	插入单速段定位运动
H106	插入两段速定位运动
H107	寸动运动
H108	手摇轮模式运动
H109	原点回归运动
H10A	电子齿轮运动
H300	多轴插补停止
H31x	多轴绝对直线插补运动
H32x	多轴相对直线插补运动

DVP 模块错误码定义请参照第 15 章附录 A 错误代码原因对照表

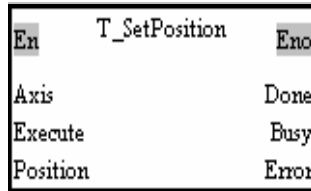
### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

## 4. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

### 5.10.16 设定当前位置



#### 1. 功能块说明

此运动功能块，目的在设定运动轴当前位置，利用 Axis 的输入数据指定某一轴，透过 Position 设定此轴的当前位置。注意:请避免用于设置同步运动(凸轮、齿轮)主轴之当前位置，避免运动出现异常。

#### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Axis	运动轴编号	WORD	K1~K6	Execute 上升沿时
Execute	上升沿时 启动运动	BOOL	TRUE / FALSE	-
Position	当前位置	DWORD	K -2,147,483,648 ~ K 2,147,483,647	Execute 上升沿时

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	运动完成	BOOL	• 位置写入完成	• Execute 下降沿时
Busy	执行中	BOOL	• Execute 上升沿时	• Done 上升沿时 • Error上升沿时
Error	功能块产生错误	BOOL	• 输入参数错误	• Execute下降沿时

#### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

#### 4. 支持模块

此运动功能块支持 DVP10PM00M 此种 DVP 机种之运动控制模块。

# 5 应用指令分类及基本使用

## 5.10.17 输入极性设定

En	T_InputPolarit~	Eno
Enable		Valid
X0_Dog0		Dog0_X0
X1_Pg0		Pg0_X1
X2_Dog1		Dog1_X2
X3_Pg1		Pg1_X3
X4_Dog2		Dog2_X4
X5_Pg2		Pg2_X5
X6_Dog3		Dog3_X6
X7_Pg3		Pg3_X7
X10_mpgA		mpgA_X10
X11_mpgB		mpgB_X11
X12_Dog4		Dog4_X12
X13_Dog5		Dog5_X13
		Busy

### 1. 功能块说明

此功能块，旨在设定输入点极性，分别可设定为常开接点或是常闭接点，默认值各接点皆为常开接点(N.O.)

### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Enable	手摇轮模式 运动允许	BOOL	TRUE / FALSE	-
X0_Dog0	端子极性	BOOL	mcNO:FALSE mcNC:TRUE	执行中持续更新
X1_Pg0	端子极性	BOOL		
X2_Dog1	端子极性	BOOL		
X3_Pg1	端子极性	BOOL		
X4_Dog2	端子极性	BOOL		
X5_Pg2	端子极性	BOOL		
X6_Dog3	端子极性	BOOL		
X7_Pg3	端子极性	BOOL		
X10_mpgA	端子极性	BOOL		
X11_mpgB	端子极性	BOOL		
X12_Dog4	端子极性	BOOL		
X13_Dog5	端子极性	BOOL		



输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	运动完成	BOOL	• Enable 上升沿时	• Enable 下降沿时
Busy	执行中	BOOL	• Enable 上升沿时	• Enable 下降沿时
Dog0_X0	端子极性	BOOL	• 执行中输入信号上升沿	• 执行中输入信号下降沿 • Enable 下降沿时
Pg0_X1	端子极性	BOOL		
Dog1_X2	端子极性	BOOL		
Pg1_X3	端子极性	BOOL		
Dog2_X4	端子极性	BOOL		
Pg2_X5	端子极性	BOOL		
Dog3_X6	端子极性	BOOL		
Pg3_X7	端子极性	BOOL		
mpgA_X10	端子极性	BOOL		
mpgB_X11	端子极性	BOOL		
Dog4_X12	端子极性	BOOL		
Dog5_X13	端子极性	BOOL		

### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

### 4. 支持模块：

此运动功能块支持 DVP10PM00M 此种 DVP 机种之运动控制模块。

## 5.11 多轴运动功能块说明

### 5.11.1 多轴绝对直线插补

En	T_AbsMoveLinear	Eno
AxisGroup		Done
Execute		Busy
Position		Error
Velocity		Aborted

#### 1. 功能块说明

此运动功能块，目的在启动指定多轴的绝对位置插补群组，利用输入 **AxisGroup** 设定执行的编号，**Position** 设定群组中每一轴目标的绝对位置，**Velocity** 设定此群组的插补速度。

# 5 应用指令分类及基本使用

## 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
AxesGroup	插补轴群组	WORD[6]	[, , , , , , ] 0:不设定 n:加入第 n(1~6)轴 (第一栏须设定)	Execute 上升沿
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
Position	目标位置群组	DWORD[6]	[, , , , , , ] K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿
Velocity	插补速度设定	DWORD	K1~ K2,147,483,647	Execute 上升沿

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	写入完成	BOOL	● 写入完成时	● Execute 下降沿时
Busy	执行中	BOOL	● Execute 上升沿时	● Error 上升沿时 ● Done 上升沿时
Error	功能块产生错误	BOOL	● 输入参数错误	● Execute 下降沿时
Aborted	功能块被其它命令中断	BOOL	● 执行中被命令停止	● Execute 下降沿时

## 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

## 4. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

### 5.11.2 多轴相对直线插补

En	T_RelMoveLinear	Eno
·	AxesGroup	Done
·	Execute	Busy
·	Distance	Error
·	Velocity	Aborted

## 1. 功能块说明

此运动功能块，目的在启动指定多轴的相对距离插补群组，利用输入 **AxesGroup** 设定执行的编号，**Distance** 设定群组中每一轴相对当前的相对距离，**Velocity** 设定此群组的插补速度。

## 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
AxesGroup	插补轴群组	WORD[6]	[, , , , , , ] 0:不设定 n:加入第 n(1~6)轴 (第一栏须设定)	Execute 上升沿
Execute	上升沿时启动运动	BOOL	TRUE / FALSE	-
Distance	目标移位群组	DWORD[6]	[, , , , , , ] K-2,147,483,648 ~ K2,147,483,647	Execute 上升沿
Velocity	插补速度设定	DWORD	K1~ K2,147,483,647	Execute 上升沿

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	写入完成	BOOL	● 写入完成时	● Execute 下降沿时
Busy	执行中	BOOL	● Execute 上升沿时	● Error 上升沿时 ● Done 上升沿时
Error	功能块产生错误	BOOL	● 输入参数错误	● Execute 下降沿时
Aborted	功能块被其它命令中断	BOOL	● 执行中被命令停止	● Execute 下降沿时

## 3. 错误产生及排除

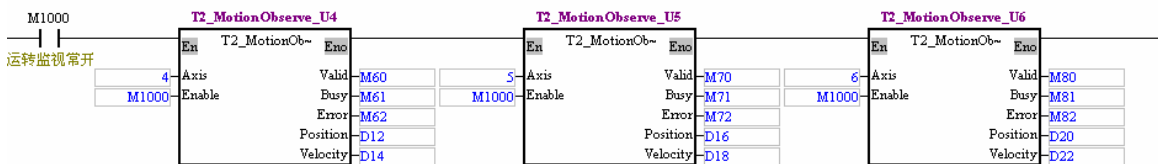
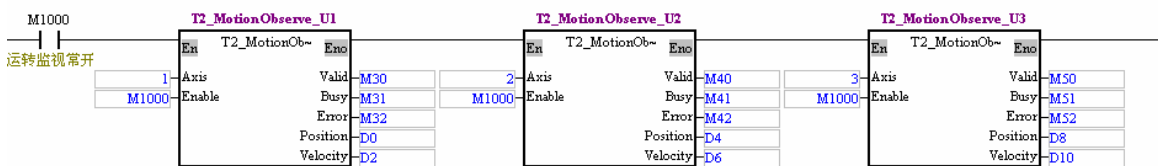
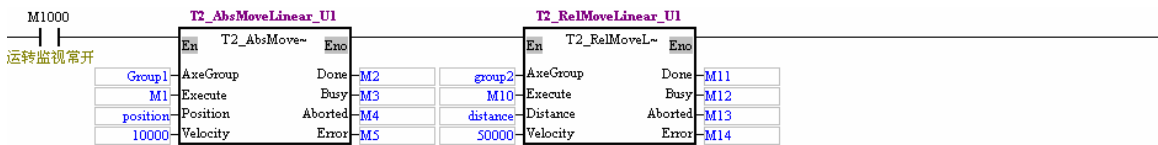
错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
功能块动作冲突	确认功能块在启动时，其它单轴运动功能块为未启动或已完成

## 4. 范例程序：

此范例的目的在使用 1 个绝对直线插补功能块与 1 个相对直线插补功能块，来执行所设定的群组作直线插补。

# 5 应用指令分类及基本使用

Local Symbols					
Class...	Identifiers	Address	Type...	Initial	Comment...
VAR	Group1		WORD[6]	[0(6)]	
VAR	Group2		WORD[6]	[0(6)]	
VAR	Position		DWORD[6]	[0(6)]	
VAR	Distance		DWORD[6]	[0(6)]	



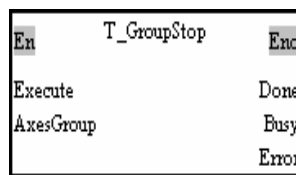
- 在 O100 区域变量内建立 Group1, Group2 此两个为 WORD 类型的数组，数组大小为 6
- 在 O100 区域变量内建立 Position, Distance 此两个为 DWORD 类型的数组，数组大小为 6
- 程序下载执行时，会设定 Group1 内的数组为[1, 2, 3, 0, 0, 0]表示使用第 1·2·3 轴作直线插补
- 程序下载执行时，会设定 Group2 内的数组为[4, 5, 6, 0, 0, 0]表示使用第 4·5·6 轴作直线插补

- 程序下载执行时，会设 Position 内的数组为[15000, 30000, -15000, 0, 0, 0]表示使用第 1·2·3 轴作绝对直线插补传送位置
- 程序下载执行时，会设 Distance 内的数组为[1000, 10000, -10000, 0, 0, 0]表示使用第 4·5·6 轴作相对直线插补传送距离
- 设定 M1 = ON，启动多轴绝对直线插补运动。
- 执行 M10 = ON，启动多轴相对直线插补运动。
- 可观察各轴的 MotionObserve 直线插补的位置是否正确。

## 5. 支持模块

此运动功能块支持 DVP10PM00M 一种 DVP 机种之运动控制模块。

### 5.1.1.3 多轴插补停止



#### 1. 功能块说明

此运动功能块，目的在停止指定多轴的插补群组，利用输入 AxesGroup 设定执行的编号。

#### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Execute	上升沿时命令直线插补停止	BOOL	TRUE / FALSE	-
AxesGroup	设定停止之插补轴群组(与启动群组相同)	WORD[6]	[_,_,_,_,_,_] 0:不设定 n:加入第 n(1~6)轴 (第一栏须设定)	Execute 上升沿

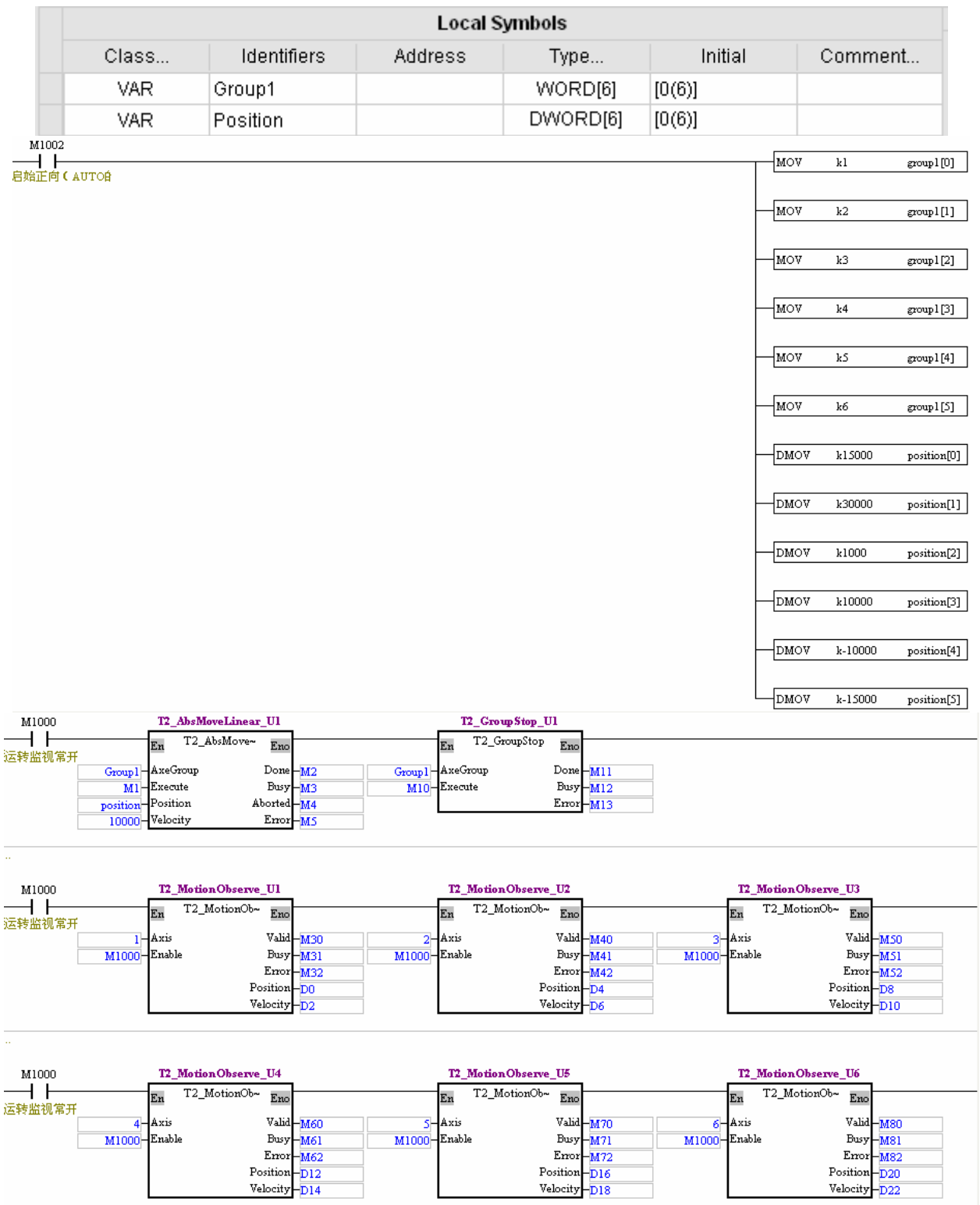
输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Done	写入完成	BOOL	● 写入完成时	● Execute 下降沿时
Busy	执行中	BOOL	● Execute 上升沿时	● Error 上升沿时 ● Done 上升沿时
Error	功能块产生错误	BOOL	● 输入参数错误	● Execute 下降沿时

#### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

此范例的目的在使用 1 个绝对直线插补功能块与 1 个相对直线插补功能块，来执行所设定的群组作直线插补。

# 5 应用指令分类及基本使用



- 在 O100 区域变量内建立 Group1 为 WORD 类型的数组，数组大小为 6
- 在 O100 区域变量内建立 Position 为 DWORD 类型的数组，数组大小为 6
- 程序下载执行时，会设定 Group1 内的数组为[1, 2, 3, 4, 5, 6]表示使用第 1 · 2 · 3 · 4 · 5 · 6 轴作直线插补
- 程序下载执行时，会设 Position 内的数组为[15000, 30000, 1000, 10000, -10000, -15000]表示使用第 1 · 2 · 3 · 4 · 5 · 6 轴作绝对直线插补传送位置
- 设定 M1 = ON，启动多轴绝对直线插补运动。

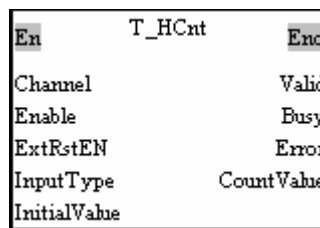
- 当 M1 = ON 时，立即执行 M10 = ON，启动多轴停止，此时多轴绝对直线插补功能块输出引脚 Aborted 会为 ON 且多轴插补停止输出引脚 DONE = ON 表示完成此组插补停止动作。

## 4. 支持模块

此运动功能块支持 DVP10PM00M 此种 DVP 机种之运动控制模块。

## 5.12 其它功能块说明

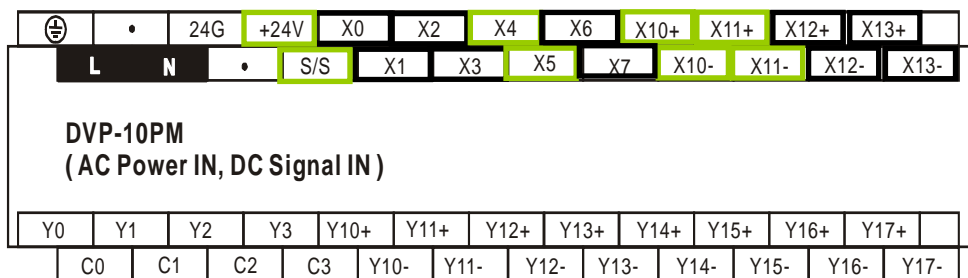
### 5.12.1 高速计数器



#### 1. 功能块说明

此运动功能块，目的为启动高速计数器，利用输入 Channel 决定计数器类别，InputType、ExternalRst、InitialValue 为计数器的参数设定，输出 CounterValue 则为计数中所得的计数值。

下图说明高速计数输入组别：



X0, X1 为第 0 组高速输入；X2, X3 为第 1 组高速输入；X4, X5 为第 2 组高速输入；X6, X7 为第 3 组高速输入；X10+, X10-, X11+, X11- 为第 4 组高速输入；X12+, X12-, X13+, X13- 为第 5 组高速输入

#### 2. 输入/输出说明：

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Channel	设定计数器组别	WORD	0~5 (*1)	Enable 上升沿
Enable	上升沿触发执行此功能块	BOOL	TRUE / FALSE	-
InputType	输入脉冲类型	WORD	mcUD:0 mcPD:1 mcAB:2 mc4AB:3	执行中持续更新
ExternalRst	外部重置开关	BOOL	TRUE / FALSE (*2)	Enable 上升沿
InitialValue	高速计数设定[Pulse]	DWORD	K0~2,147,483,647	Enable 上升沿

# 5 应用指令分类及基本使用

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	中断允许状态	BOOL	<ul style="list-style-type: none"> <li>● 设定完成并开启中断功能</li> </ul>	<ul style="list-style-type: none"> <li>● Enable 下降沿时</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>● Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>● Error 上升沿时</li> <li>● Enable 下降沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>● 输入参数错误</li> <li>● 指定来源已被占用</li> </ul>	<ul style="list-style-type: none"> <li>● Enable 下降沿时</li> </ul>

输出数值引脚				
引脚名称	功能说明	数据格式	输出数值范围	数据有效时机
CounterValue	计数值	DWORD	K0~2,147,483,647	当 Valid=ON 时持续更新数值

(\*1) Channel 输入值定义如下

(\*2) 启动外部重置计数器对应引脚如下

输入值	定义
0	C200
1	C204
2	C208
3	C212
4	C216
5	C220

组别	重置引脚
0	X10
1	X11
2	X12
3	X13
4	X0
5	X1

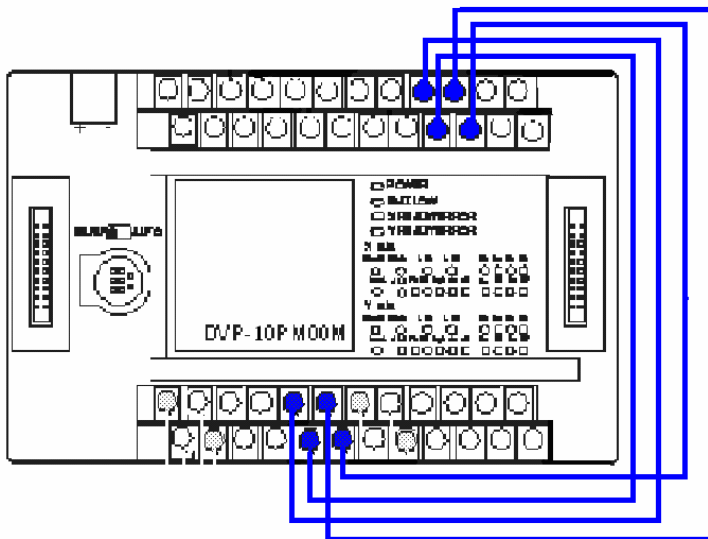
### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
此组别已在使用中	使用另外一组计数器，或将使用中的组别关闭

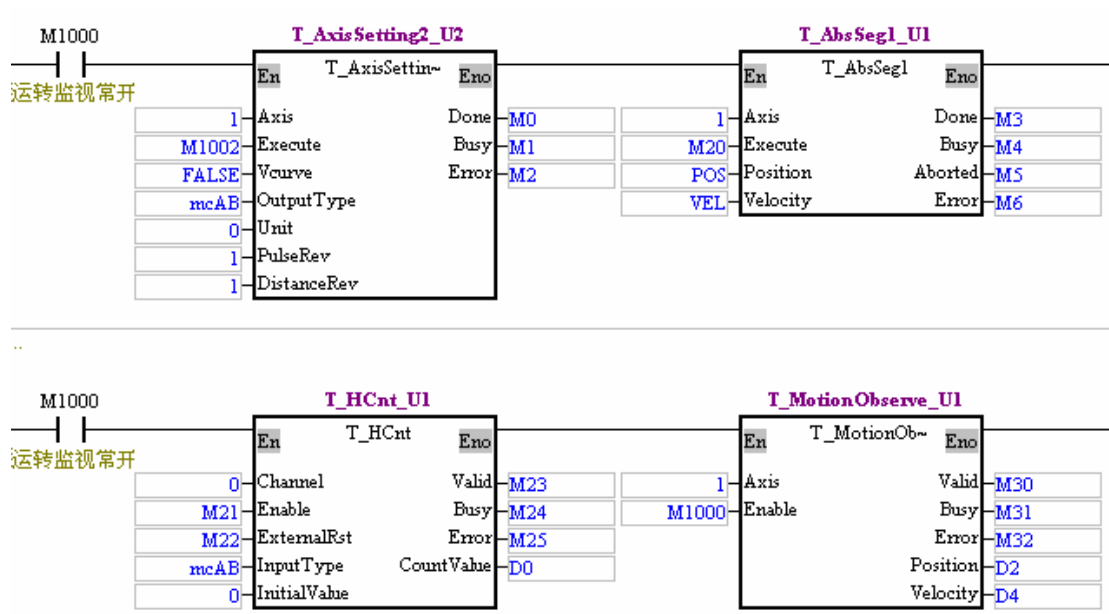
### 4. 范例程序：

此范例的目的在使用第 1 轴脉冲输出，给第 0 组计数器计数，查看输出脉冲数是否与输入脉冲数相同。外部配线下如下图所示：





Y10+接至 X10+  
 Y10-接至 X10-  
 Y11+接至 X11+  
 Y11-接至 X11-



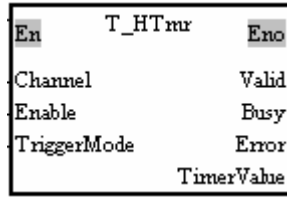
- 执行后程序会自动设定轴 1 脉冲输出类型为 AB 相
- 设定 M21 = ON · 启动第 0 组计数功能
- 设定 POS = 30,000 · VEL = 10,000
- 执行 M20 = ON · 启动绝对单段速传送
- 当 M3 = ON 时 · 比较 D0(计数值)与 D2 目前位置 · 是否相同。

## 5. 支持模块

此运动功能块支持 DVP10PM00M 此一种 DVP 机种之运动控制模块。

# 5 应用指令分类及基本使用

## 5.12.2 高速定时器



### 1. 功能块说明

此运动功能块，目的为启动高速定时器，利用输入 **Channel** 决定计数器类别，**TriggerMode** 为定时器计时模式设定 **Timevalue** 则为定时器中所得的计时数值，单位为  $0.01\ us$ 。高速计时组别与计数器相同，请参考计数器组别。

### 2. 输入/输出说明：

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Channel	设定计数器组别	WORD	0~5 (*1)	Enable 上升沿
Enable	上升沿触发执行此功能块	BOOL	TRUE / FALSE	-
TriggerMode	计时触发模式设定	BOOL	mcUp_Down(0): 上升沿计数至下绿 mcUp_Up(1): 上升沿计数至上升沿	Enable 上升沿

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	中断允许状态	BOOL	● 设定完成并开启中断功能	● Enable 下降沿时
Busy	执行中	BOOL	● Enable 上升沿时	● Error 上升沿时 ● Enable 下降沿时
Error	功能块产生错误	BOOL	● 输入参数错误 ● 指定来源已被占用	● Enable 下降沿时

输出数值引脚				
引脚名称	功能说明	数据格式	输出数值范围	数据有效时机
TimeValue	计数值	DWORD	K0~2,147,483,647	当 Valid=ON 时持续更新数值

(\*1) Channel 输入值定义与启动计时引脚如下

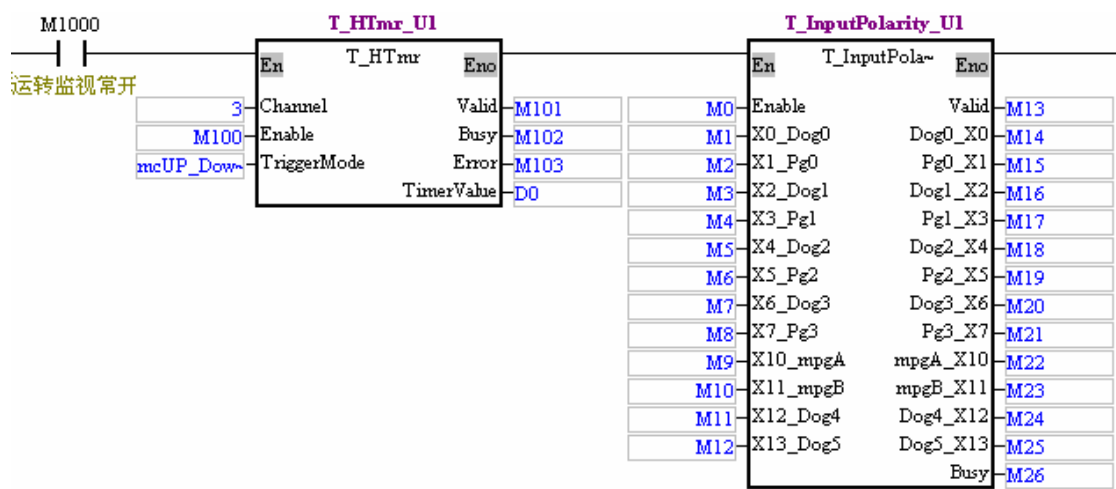
输入值	定义	启动计时引脚
0	C200	X10
1	C204	X11
2	C208	X12
3	C212	X13
4	C216	X0
5	C220	X1

### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
此组别已在使用中	使用另外一组计数器，或将使用中的组别关闭

### 4. 范例程序：

此范例的目的在使用 1 个高速定时器，利用输入极性设定仿真定时器输入信号变化。



- 设定 M0 = ON，启动输入极性设定功能块。
- 设定 M100 = ON，启动第 0 组计数功能。
- 设定 M12 = ON
- 设定 M12 = OFF
- 此时 **TimerValue** 会显示数值，此数值表示 M12 由 ON→OFF 之间的时间，将 **TimerValue** \* 0.01 即为 *us*

### 5. 支持模块

此运动功能块支持 DVP10PM00M 此一种 DVP 机种之运动控制模块。

# 5 应用指令分类及基本使用

## 5.12.3 高速比较设定

En	T_Compare	Eno
Channel		Valid
Enable		Busy
Source		Error
CmpMode		
OutputDevice		
OutputMode		
CmpValue		

### 1. 功能块说明

此运动功能块·目的为启动高速比较功能·利用输入 Channel 决定比较器组别·Source 为比较器来源·CmpMode、OutputDevie 为比较器的参数设定·比较成立时输出 Status

### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Channel	设定比较器组别	WORD	0~7	Enable 上升沿时
Enable	上升沿触发执行此功能块	BOOL	TRUE / FALSE	-
Source	设定比较来源	WORD	mcCmpAxis1:0 mcCmpAxis2:1 mcCmpAxis3:2 mcCmpAxis4:3 mcCmpC200:4 mcCmpC204:5 mcCmpC208:6 mcCmpC212:7	Enable 上升沿时
CmpMode	设定比较方式	WORD	mcCmp=:1 mcCmp>=:2 mcCmp<=:3	Enable 上升沿时
OutputDevice	当比较条件成立时输出装置	WORD	mcCmpY0:0 mcCmpY1:1 mcCmpY2:2 mcCmpY3:3 mcCmpRstC200:4 mcCmpRstC204:5	Enable 上升沿时
OutputMode	设定输出装置输出方式	BOOL	mcCmpSet:TRUE mcCmpRst:FALSE	Enable 上升沿
CmpValue	设定比较值	DWORD	K-2,147,483,647~K2,147,483,647	Enable 上升沿

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	中断允许状态	BOOL	● 设定完成并开启中断功能	● Enable 下降沿时

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>● Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>● Error 上升沿时</li> <li>● Enable 下降沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>● 输入参数错误</li> <li>● 指定来源已被占用</li> </ul>	<ul style="list-style-type: none"> <li>● Enable 下降沿时</li> </ul>

注意：高速比较与高速捕捉相加最大组别为 8 组

◆ 比较来源说明：

- mcCmpAxis1：比较来源为第 1 轴当前位置
- mcCmpAxis2：比较来源为第 2 轴当前位置
- mcCmpAxis3：比较来源为第 3 轴当前位置
- mcCmpAxis4：比较来源为第 4 轴当前位置
- mcCmpC200：比较来源为 C200
- mcCmpC204：比较来源为 C204
- mcCmpC208：比较来源为 C208
- mcCmpC212：比较来源为 C212

◆ 比较方式：

- mcCmp=：比较来源「等于」比较值输出条件成立装置
- mcCmp>=：比较来源「大于等于」比较值输出条件成立装置
- mcCmp<=：比较来源「小于等于」比较值输出条件成立装置

◆ 条件成立输出装置

- mcCmpY0：输出 Y0 装置
- mcCmpY1：输出 Y1 装置
- mcCmpY2：输出 Y2 装置
- mcCmpY3：输出 Y3 装置
- mcCmpRstC200：输出 C200 装置
- mcCmpRstC204：输出 C204 装置
- mcCmpRstC208：输出 C208 装置
- mcCmpRstC212：输出 C212 装置

◆ 输出方式

- 当装置为 Y0, Y1, Y2, Y3 时
  - McCmpSet：允许
  - McCmpRst：禁能
- 当装置为 C200, C204, C208, C212 时
  - McCmpSet：清除装置
  - McCmpRst：装置计数

### 3. 错误产生及排除

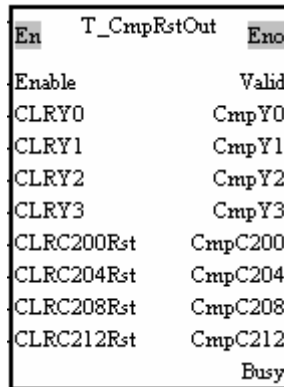
# 5 应用指令分类及基本使用

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
此组别已被使用	使用别组组别

## 4. 支持模块

此运动功能块支持 DVP10PM00M 此一种 DVP 机种之运动控制模块。

### 5.12.4 高速比较重置



#### 1. 功能块说明

此运动功能块，目的为重置高速比较功能，利用 CLRclr0、CLRclr1、CLRY2、CLRY3、CLRC200Rst、CLR C204Rst 决定清除的输出，其输出端则表示目前比较输出装置状态为允许/禁能

#### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Enable	上升沿触发执行此功能块	BOOL	TRUE / FALSE	-
CLRY0	清除比较输出	BOOL	TRUE / FALSE	执行中持续更新
CLRY1				
CLRY2				
CLRY3				
CLRC200Rst				
CLRC204Rst				
CLRC208Rst				
CLRC212Rst				

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	中断允许状态	BOOL	<ul style="list-style-type: none"> <li>● 设定完成并开启中断功能</li> </ul>	<ul style="list-style-type: none"> <li>● Enable 下降沿时</li> </ul>
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>● Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>● Error 上升沿时</li> <li>● Enable 下降沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>● 输入参数错误</li> <li>● 指定来源已被占用</li> </ul>	<ul style="list-style-type: none"> <li>● Enable 下降沿时</li> </ul>

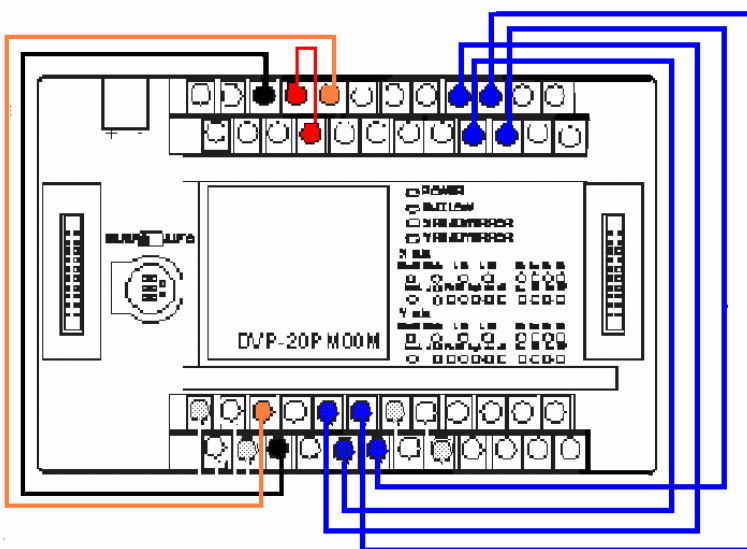
输出数值引脚				
引脚名称	功能说明	数据格式	输出数值范围	数据有效时机
CmpY0	比较输出状态	BOOL	TRUE / FALSE	当 Valid=ON 时持续更新
CmpY1				
CmpY2				
CmpY3				
CmpC200				
CmpC204				
CmpC208				
CmpC212				

### 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

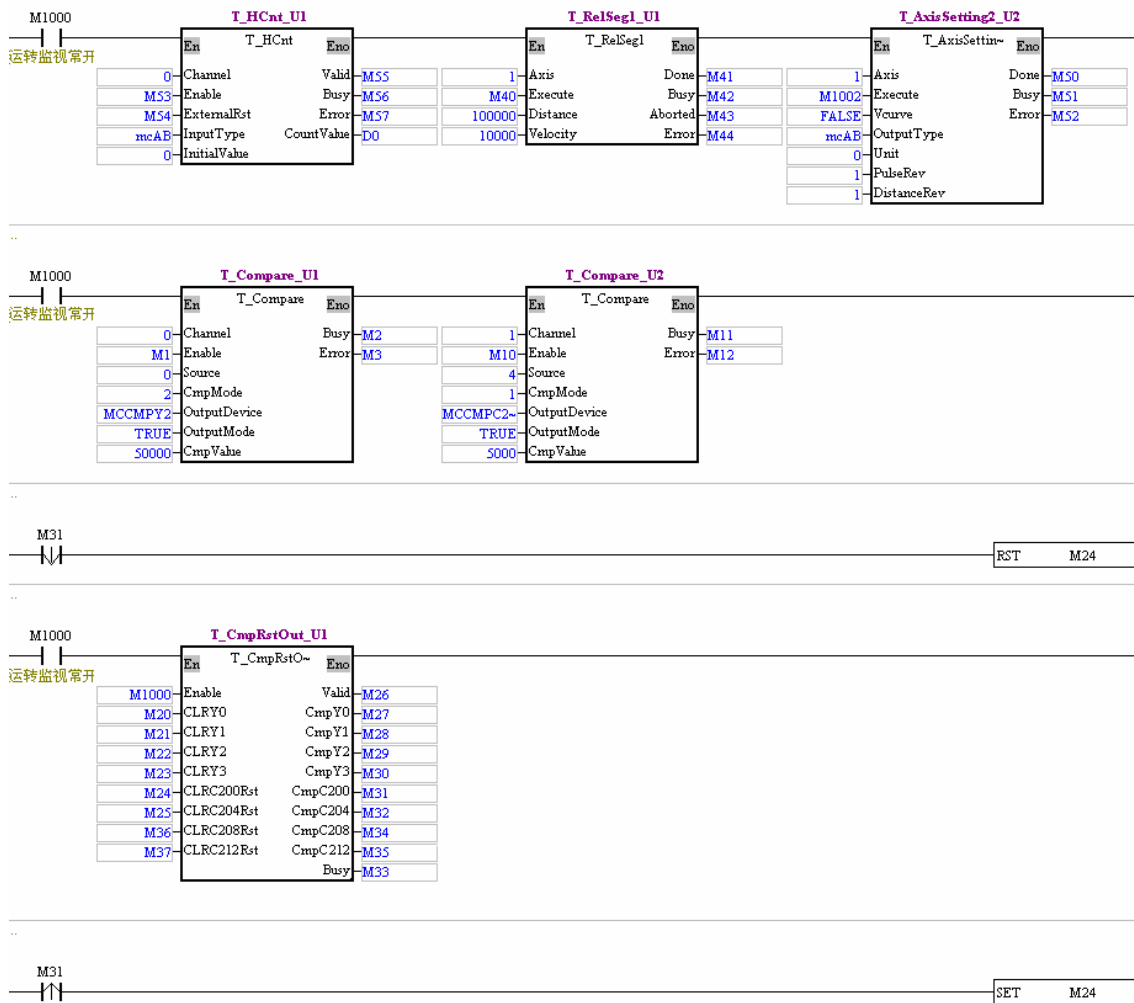
### 4. 范例程序：

此范例目的设定 2 组高速比较器，当条件到达后，查看输出装置是否有依照设定执行。外部配线如下图所示：



Y10±接至 X10±  
 Y11±接至 X11±  
 C2 接至 24G  
 Y2 接至 X0  
 S/S 接至+24V

# 5 应用指令分类及基本使用



- 执行后程序会自动设定轴 1 脉冲输出类型为 AB 相，并启动高速比较重置功能块，进行输出状态读取。
- 设定 M53 = ON，启动高速计数
- 设定 M1 = ON，设定第 0 组为高速比较器。  
高速比较器设定：当第 1 轴当前位置大于等于 50,000 时，Set Y2。
- 设定 M10 = ON，设定第 1 组为高速比较器  
高速比较器设定：当 C200 当前位置等于 5,000 时，将 C200 数值清除为 0
- 设定 M40 = ON，启动第 1 轴相对位置移位，移位 100,000 脉冲后结束
- 当 C200 = 5,000，输出状态 M31 = ON，表示第 1 组高速比较器条件成立，便将 C200 数值清除为 0。  
当 M31 上升沿启动后，会清除 C200 条件成立的状态，后续当 C200 输入值再次为 5,000 时，高速比较器条件才会再次成立将 C200 数值清除为 0，若没有将条件成立的状态清除(即输出引脚 CmpC200 = ON)，下次条件成立，高速比较器不会作动。
- 当第 1 轴当前位置为 100,000 时，第 0 组高速比较器条件成立会输出 Y2 信号，可查看 X0 灯号是否亮起。



- 当 X0 灯号亮起，表示条件成立，可利用 M22 将条件成立状态清除，让 X0 = OFF。

## 5. 支持模块

此运动功能块支持 DVP10PM00M 此种 DVP 机种之运动控制模块。

### 5.12.5 高速捕捉设定

En	T_Capture	Eno
Channel		Valid
Enable		Busy
Source		Error
TriggerDevice		CapValue
InitialValue		

#### 1. 功能块说明

此运动功能块，目的为启动高速捕捉功能，利用输入 Channel 决定比较器组别，Source 为捕捉信号来源，TriggerDevice、InitialValue 为捕捉的参数设定，捕捉触发时输出 CapValue。

#### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Channel	设定捕捉器组别	WORD	0~7	Enable 上升沿
Enable	上升沿触发执行此功能块	BOOL	TRUE / FALSE	-
Source	设定捕捉来源	WORD	mcCapAxis1:0 mcCapAxis2:1 mcCapAxis3:2 mcCapAxis4:3 mcCapC200:4 mcCapC204:5 mcCapC208:6 mcCapC212:7	Enable 上升沿
TriggerDevice	捕捉触发装置	WORD	mcX0:0 mcX1:1 mcX2:2 mcX3:3 mcX4:4 mcX5:5 mcX6:6 mcX7:7 mcX10:8 mcX11:9 mcX12:10 mcX13:11	Enable 上升沿
InitialValue	捕捉	DWORD	K-2,147,483,648 ~ K2,147,483,647	Enable 上升沿

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	中断允许状态	BOOL	● 设定完成并开启	● Enable 下降沿时

## 5 应用指令分类及基本使用

			中断功能	
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>● Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>● Error 上升沿时</li> <li>● Enable 下降沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>● 输入参数错误</li> <li>● 指定来源已被占用</li> </ul>	<ul style="list-style-type: none"> <li>● Enable 下降沿时</li> </ul>

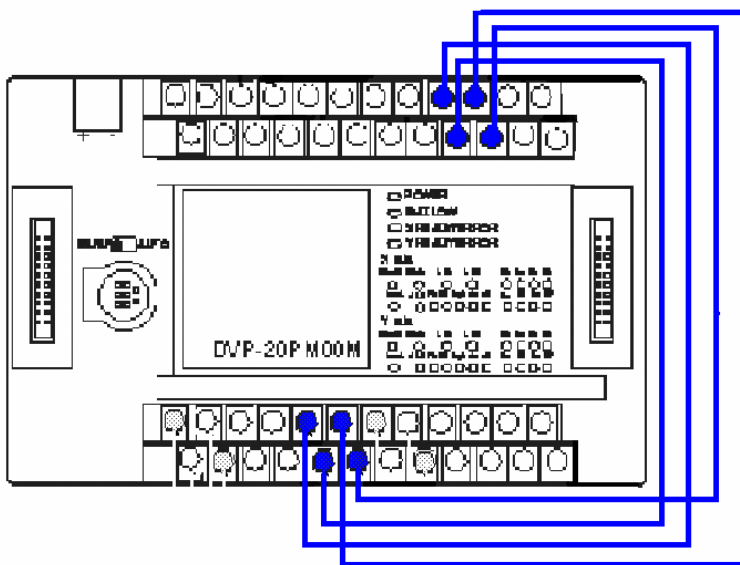
输出数值引脚				
引脚名称	功能说明	数据格式	输出数值范围	数据有效时机
CapValue	捕捉值	DWORD	K-2,147,483,648 ~ K2,147,483,647	执行中持续更新数值·若无触发事件·捕捉值会维持上一次捕捉到的值

### 3. 错误产生及排除：

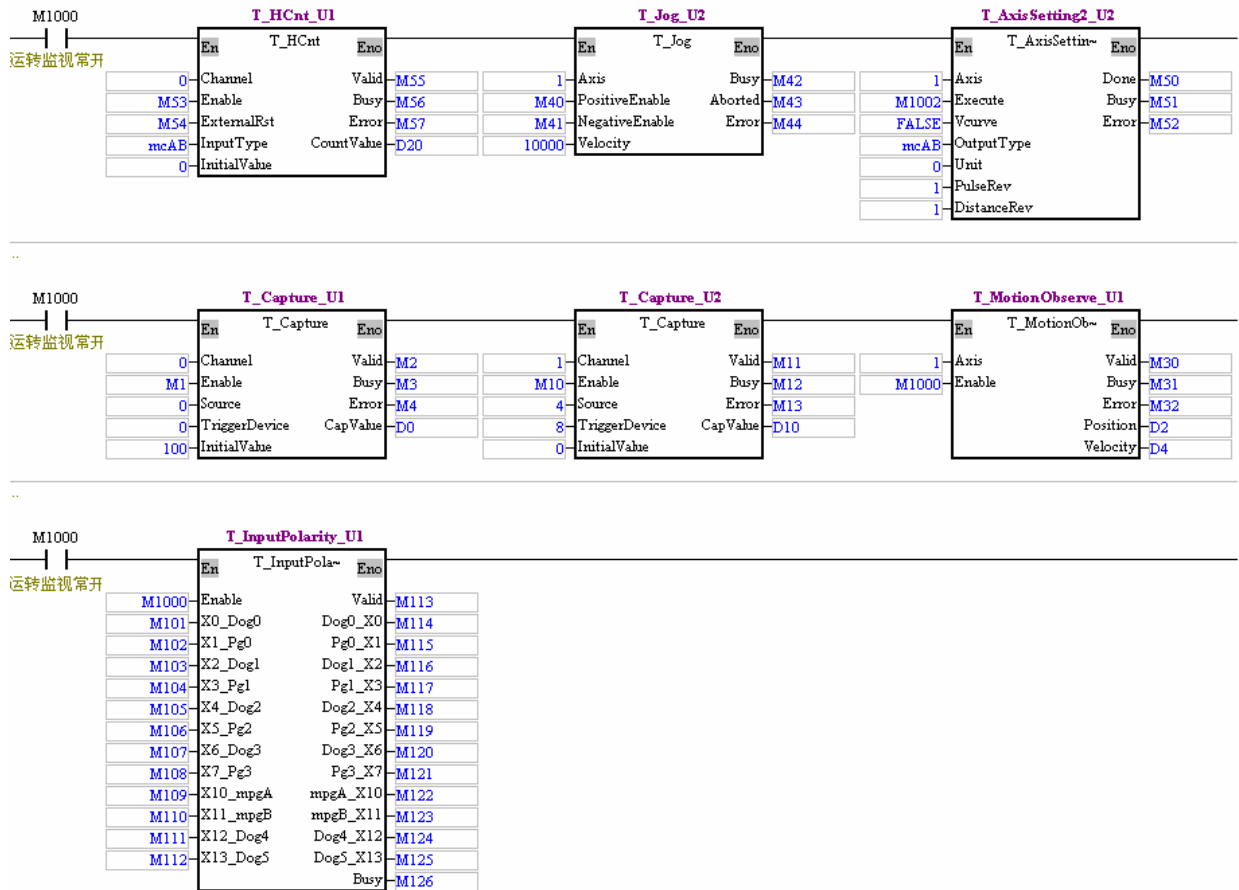
错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围
此组别已被使用	使用其它组别

### 4. 范例程序：

此范例使用 2 组高速捕捉器。当外部信号触发后，第 1 组捕捉第 1 轴当前位置值，第 2 组捕捉 C200 当前计数值，利用 InputPolarity 功能块修改极性作为触发信号，外部接线图如下所示。



Y10±接至 X10±  
Y11±接至 X11±



- 执行后程序会自动设定轴 1 脉冲输出类型为 AB 相，并启动 MotionObserve 读取第 1 轴当前位置值与当前速度
- 设定 M53 = ON，启动高速计数
- 设定 M1 = ON, 设定第 0 组为高速捕捉器  
高速捕捉器设定：当 X0 信号触发，捕捉第 1 轴当前位置值。
- 设定 M10 = ON，启动第 1 组高速捕捉器  
高速捕捉器设定：当 X10 信号触发，捕捉 C200 当前计数值。
- 设定 M40 = ON，启动第 1 轴寸动运行。
- 当 M101 = ON，改变 X0 极性为常闭，产生上升沿信号，此时 D0 数值会改变。
- 当 M109 = ON，改变 X10 极性为常闭，产生上升沿信号，此时 D10 数值会改变。

## 5. 支持模块

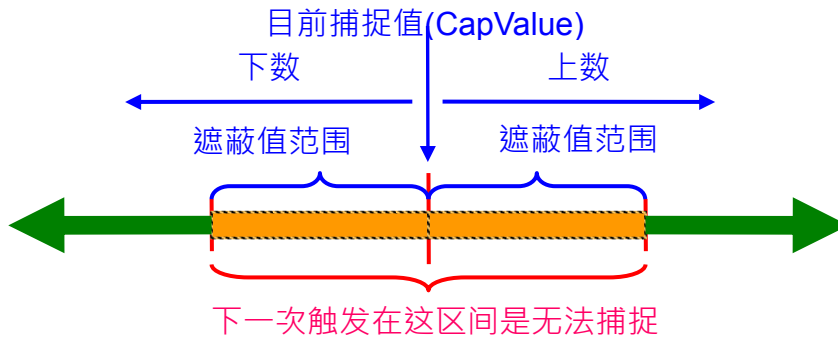
此运动功能块支持 DVP10PM00M 此种 DVP 机种之运动控制模块。

# 5 应用指令分类及基本使用

## 5.12.6 高速捕捉遮蔽



### 1. 功能块说明



此运动功能块，目的为启动高速捕捉遮蔽，利用输入 **MaskValue** 决定捕捉遮蔽值。由下图知，若启动高速捕捉遮蔽时，将以目前捕捉值为基准，上数与下数的遮蔽值范围内，此时捕捉信号触发将无效。

### 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
Enable	上升沿触发执行此功能块	BOOL	TRUE / FALSE	-
MaskValue	设定捕捉遮蔽值[Hz]	DWORD	K0~2,147,483,647	设定捕捉遮蔽值[Hz]

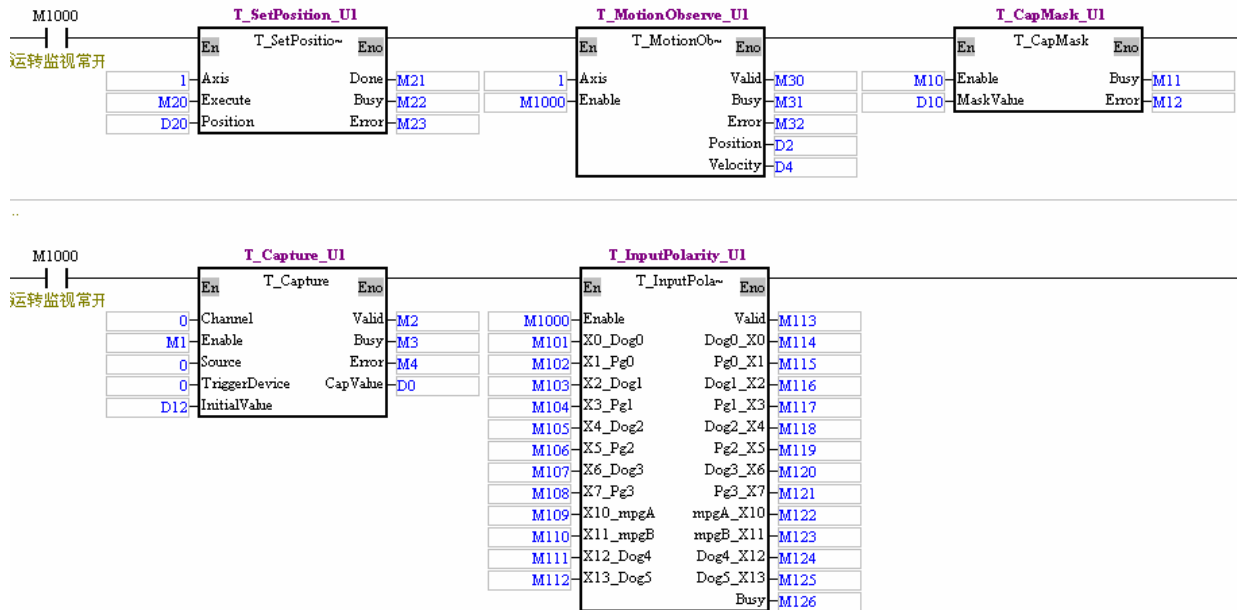
输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Busy	执行中	BOOL	<ul style="list-style-type: none"> <li>● Enable 上升沿时</li> </ul>	<ul style="list-style-type: none"> <li>● Error 上升沿时</li> <li>● Enable 下降沿时</li> </ul>
Error	功能块产生错误	BOOL	<ul style="list-style-type: none"> <li>● 输入参数错误</li> <li>● 指定来源已被占用</li> </ul>	<ul style="list-style-type: none"> <li>● Enable 下降沿时</li> </ul>

### 3. 错误产生及排除：

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

## 4. 范例程序：

此范例使用 1 组高速捕捉器与遮蔽功能块。启动遮蔽功能块，在 CapValue±MarkValue 范围内的第 1 轴位置数值，外部信号触发后，将不会捕捉。

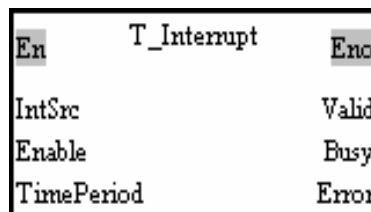


- 执行后程序会读取第 1 轴当前位置值与当前速度
- 设定 D20 = 0 并执行 M20 = ON 将第 1 轴位置写入为 0
- 设定 D12 = 100 与 M1 = ON，启动高速捕捉  
高速捕捉设定：当 X0 信号触发，捕捉第 1 轴当前位置值
- 设定 D10 = 500 与 M10 = ON，启动遮蔽功能
- 设定 M101 = ON，此时捕捉信号触发，但 CapValue 数值仍然为 100
- 执行写入第 1 轴当前位置为 500
- 设定 M101 = ON，此时捕捉信号触发，但 CapValue 数值仍然为 100
- 执行写入第 1 轴当前位置为 600
- 设定 M100 = ON，此时捕捉信号触发，CapValue 数值为 600

## 5. 支持模块

此运动功能块支持 DVP10PM00M 此种 DVP 机种之运动控制模块。

### 5.12.7 中断设定



#### 1. 功能块说明

此运动功能块，目的为设定设定与启动中断功能。

# 5 应用指令分类及基本使用

## 2. 输入/输出说明

输入引脚				
引脚名称	功能说明	数据格式	允许范围设定值	数据有效时机
IntSCR	设定中断组别	WORD	IntTimer:0 IntX00:1 IntX01:2 IntX02:3 IntX03:4 IntX04:5 IntX05:6 IntX06:7 IntX07:8	Enable 上升沿时
Enable	上升沿触发执行此功能块	BOOL	TRUE / FALSE	-
TimePeriod	设定时间中断周期[ms] (端子中断不使用)	WORD	K1~65,535	执行中持续更新

输出状态引脚				
引脚名称	功能说明	数据格式	产生上升沿时机	产生下降沿时机
Valid	中断允许状态	BOOL	● 设定完成并开启中断功能	● Enable 下降沿时
Busy	执行中	BOOL	● Enable 上升沿时	● Error 上升沿时 ● Enable 下降沿时
Error	功能块产生错误	BOOL	● 输入参数错误 ● 指定来源已被占用	● Enable 下降沿时

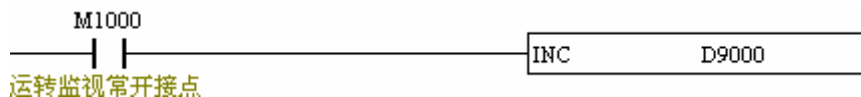
## 3. 错误产生及排除

错误状况	排除方式
输入数值错误	检查输入引脚输入数值，是否超过可允许范围

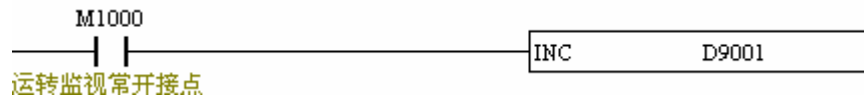
## 4. 范例程序

此范例使用 2 个中断分别设定时间中断与外部中断，利用输入极性设定仿真外部接点信号输入。

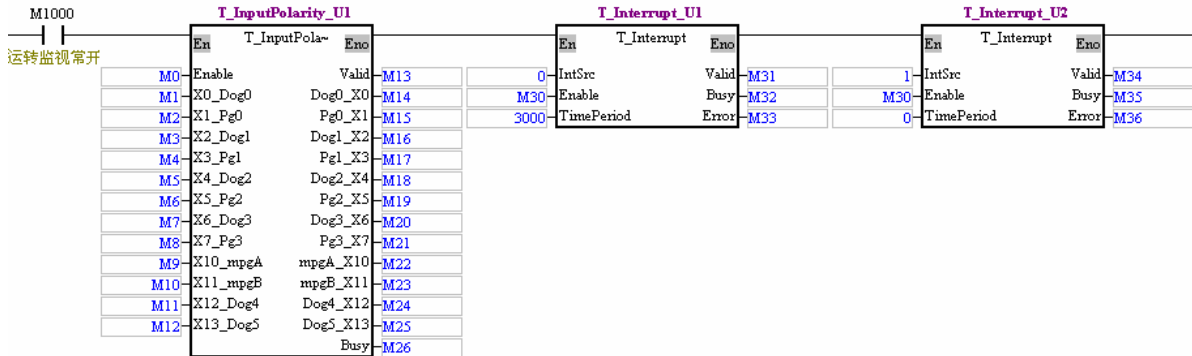
I0 程序：



I1 程序：



O100 程序：



- 设定 M0=ON，启动输入极性设定功能块。
- 设定 M30 = ON，启动时间中启功能与 I1 外部中断功能
- 时间中断启动后，每 3 秒中便会执行一次，可观察 D9000 每 3 秒增加一次数值
- I1 外部中断启动后，可设定 M1 仿真外部信号输入。当 M1 由 OFF→ON 时，D9001 便会增加一次。

## 5. 支持模块

此运动功能块支持 DVP10PM00M 此一种 DVP 机种之运动控制模块。

MEMO



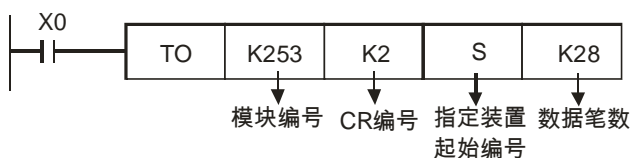
## 6.1 DVP-10PM 运动模块支持多轴插补

利用 TO 指令在程序中下达实现自定义直线插补命令

## 6.2 TO 指令使用说明

DVP10PM 可透过 TO 指令设定/读取各轴的直线插补功能开启以及关闭，以下说明如何使用 TO 指令来设定直线插补功能的状态。

### ■ CR#2 设定插补开启



位置	设定控制
S · S+1	插补轴选择
S+2 · S+3	插补速度
S+4 · S+5	轴 X 位置
S+6 · S+7	轴 Y 位置
S+8 · S+9	轴 Z 位置
S+10 · S+11	轴 A 位置
S+12 · S+13	轴 B 位置
S+14 · S+15	轴 C 位置

其中插补轴选择 (S · S+1) 定义如下：

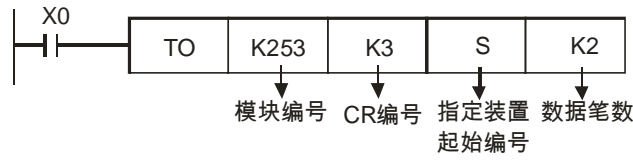
Bit	31 : 12	11 : 10	9 : 8	7 : 6	5 : 4	3 : 2	1 : 0
轴号	不使用	轴 C	轴 B	轴 A	轴 Z	轴 Y	轴 X

每一轴使用两个 bit 作为句柄，而句柄定义如下：

内容值	定义
0	不参与插补
1	直线插补开启
2	目前不使用
3	目前不使用

# 6 多轴插补功能

## ■ CR#3 设定插补停止



位置	设定控制
S · S+1	插补轴选择

其中插补轴选择 ( S · S+1 ) 定义如下：

Bit	31 : 12	11 : 10	9 : 8	7 : 6	5 : 4	3 : 2	1 : 0
轴号	不使用	轴 C	轴 B	轴 A	轴 Z	轴 Y	轴 X

每一轴使用两个 bit 作为句柄，而句柄定义如下：

内容值	定义
0	不参与插补
1	直线插补停止
2	目前不使用
3	目前不使用

■ 而各轴在运行直线插补时的相关设定参数可利用 D1816 ( D1896 · D1976... )，来做设定上的修改。参数设定定义如下：

bit#	轴参数设定	bit#	轴参数设定
0	单位系设定 ( 注一 )	8	原点回归方向 ( 注三 )
1		9	原点回归模式 ( 注三 )
2	位置数据倍率设定 ( 注二 )	10	原点回归 DOG 下降沿检测 ( 注三 )
3		11	脉冲旋转方向 ( 注三 )
4	脉冲型式 ( 注二 )	12	相对绝对坐标 ( 注三 )
5		13	DOG 触发启动模式 ( 注三 )
6*	PWM 模式 ( 注三 )	14	
7	cap 模块输出点 ON 立即停止	15	

注一：

b1	b0	单位系
0	0	电机单位
0	1	机械单位
1	0	复合单位
1	1	

	电机单位	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

注二：

b3	b2	位置数据倍率设定	b5	b4	说明
0	0	100	0	0	正向脉冲+反向脉冲
0	1	101	0	1	脉冲+方向
1	0	102	1	0	A/B 相脉冲 ( 2 相 2 )
1	1	103	1	1	

注三：

bit#	说明
6	b[6]=1：启用 PWM 模式 (1) 运转命令中启动 JOG+运转，则 Y0~Y3 输出 PWM (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 (3) 脉冲宽度设定：D1838、D1918、D1998、D2078... (4) 脉冲频率设定：D1842、D1922、D2002、D2082...
8	b[8]=0：执行原点回归方向，往当前位置 ( CP ) 内容值递减方向执行 b[8]=1：执行原点回归方向，往当前位置 ( CP ) 内容值递增方向执行
9	b[9]=0：正常模式；b[9]=1：覆写模式
10	b[10]=0：原点回归 DOG 下降沿检测 b[10]=1：原点回归 DOG 上升沿检测
11	b[11]=0：当正转运动时，当前位置 ( CP ) 内容值递增 b[11]=1：当正转运动时，当前位置 ( CP ) 内容值递减
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	b[13]=0：DOG 信号上升沿触发 b[13]=1：DOG 信号下降沿触发 ( 插入单段速定位运动模式、插入二段速定位运动模式有效 )
14	b[14]=0：为加速采梯形加速曲线 b[14]=1：为加速采 S 加速曲线

而各轴的直线插补动作是否完成可利用运动标志 M1792 ( M1872、M1952.... ) 来判断。

MEMO

## 7.1 使用设定介绍

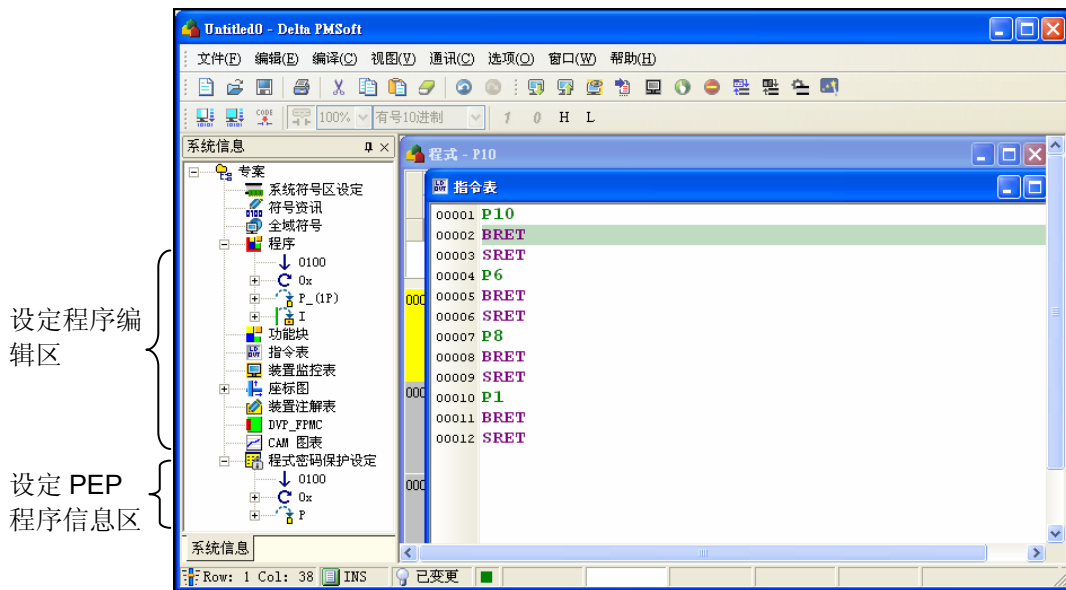
DVP-10PM 主机可透过 PMSoft 软件提供使用者设定密码以保护编写的程序。下载程序至 10PM 主机之前设定密码，来达到保护程序被不正当获取或修改的功能，因此在上传有加密的程序时，必须输入正确密码，才能上传程序到 PMSoft 提供使用者修改。PMSoft 可分别对主程序 O100、Oxn (n : 0~99, 100 个) 运动程序及 Pm (m : 0~255, 256 个) 子程序做各别加密设定。

使用设定主要分为三个部份，分别为系统信息区设定、下载设定及上传设定，下列将分几个单元介绍密码锁定设定(PEP Setting)。

### 7.1.1 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；特定程序加密设定区(PEP Setting)，用于保护程序的加密，编辑窗口中总共有 357 组程序可被设定加密保护，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定。



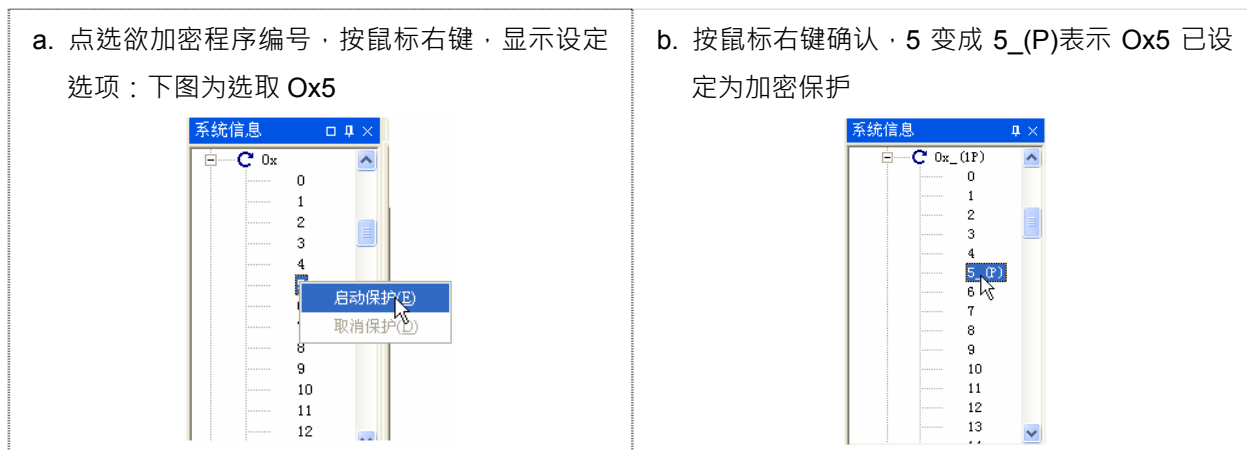
- 运动程序和子程序展开：由下图(a)(b)所示，点选运动子程序或是子程序将分别展开所属的程序编号，分别有 100 个和 256 个。



# 7 程序加密设定

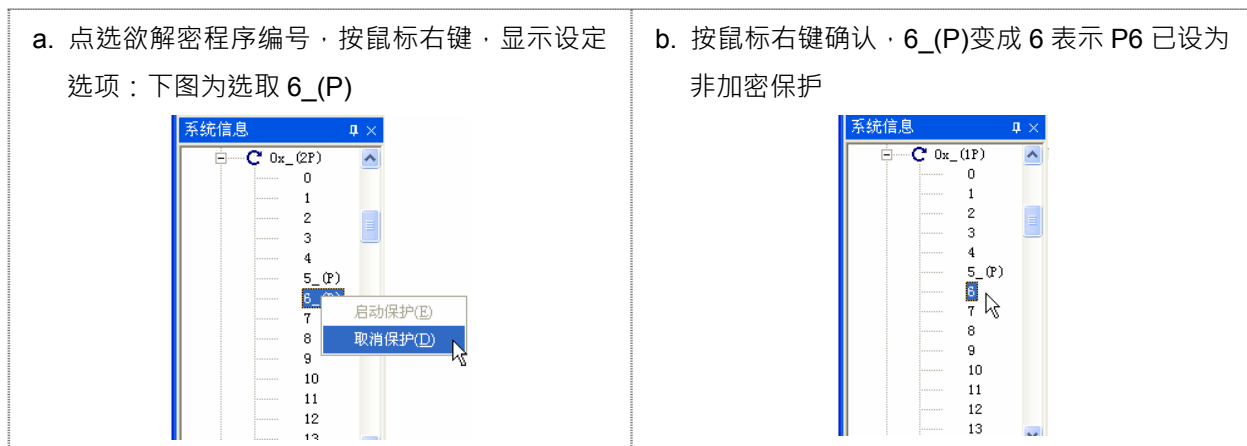
## ■ 特定程序加密的套用设定：

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. 此时被设定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定为需要加密，而子程序也是同样原理，但主程序只有一个，请直接在 O100 节点上设定启动保护，使其为 O100\_(P)。



## ■ 解除保护的套用设定：


1. 在欲进行解密的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. 此时原来被设定加密的程序编号，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定之后，即会变为 P\_(2P)，以此类推。

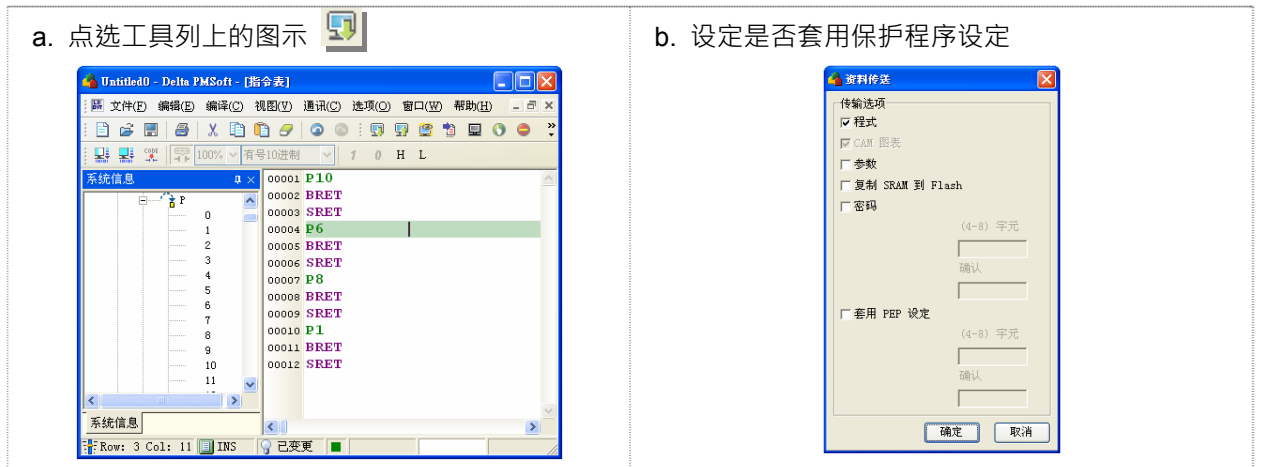


### 7.1.2 下载程序设定

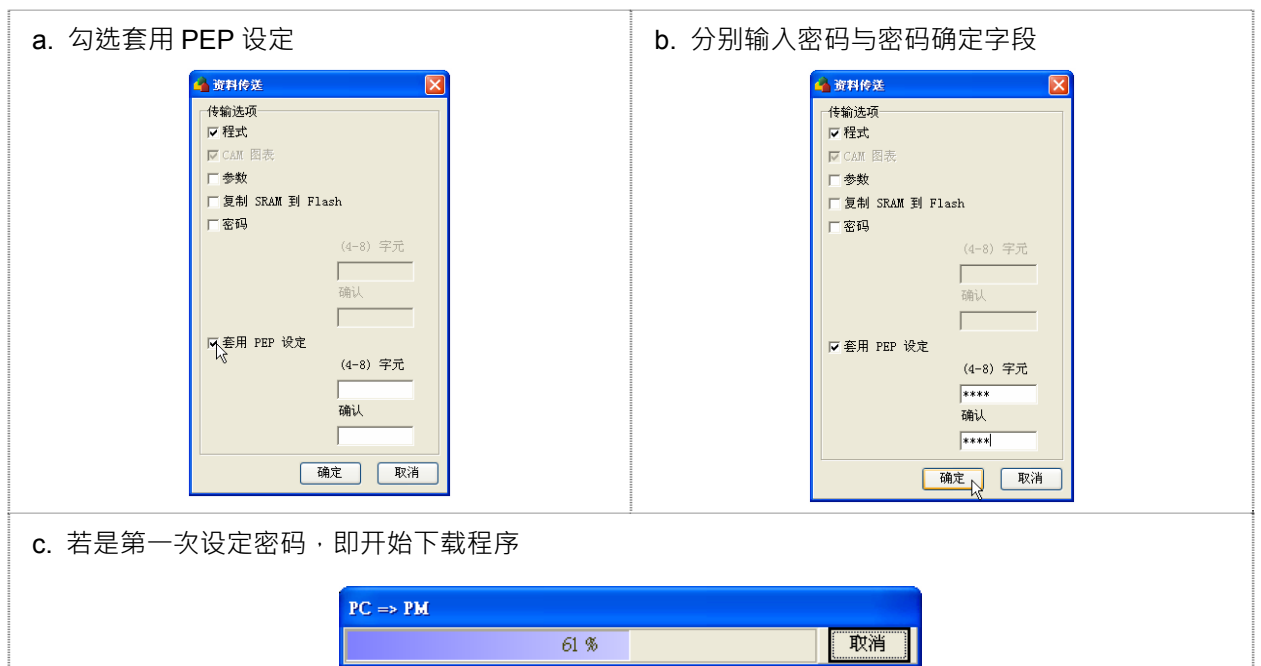
下载程序时，会弹跳出功能窗口，可勾选是否套用 PEP 保护程序设定，若无勾选套用 PEP 保护程序，PMSoft 会把所有的程序传送到主机一般程序储存区，若勾选套用 PEP 保护程序，PMSoft 会要求设定被加密保护的密码，之后下载程序时，会把未加密设定的程序存放一般程序储存区，而有加密设定的程序或 CAM 数据则会被存放程序保护储存区。下列将分两种状况详细说明其操作步骤：

■ 勾选套用(PEP)保护程序设定：

1. 选择『通讯 ( C )』功能并点选 PC→PM(D) ，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)。

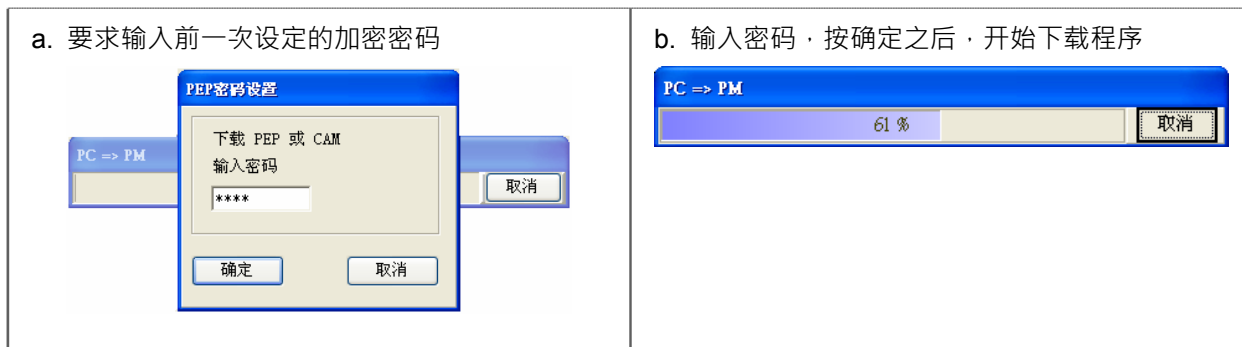


2. 勾选套用 PEP 保护设定(如下图 a) ，此时需设定 4~8 字符的加密保护密码 ，请分别输入密码与密码确认字段后按确定(如下图 b) 。若为第一次设定加密保护密码 ，在进行传输时 ，PMSOft 会直接执行下载程序或 CAM 数据到 DVP-PM 主机(如下图 c) 。




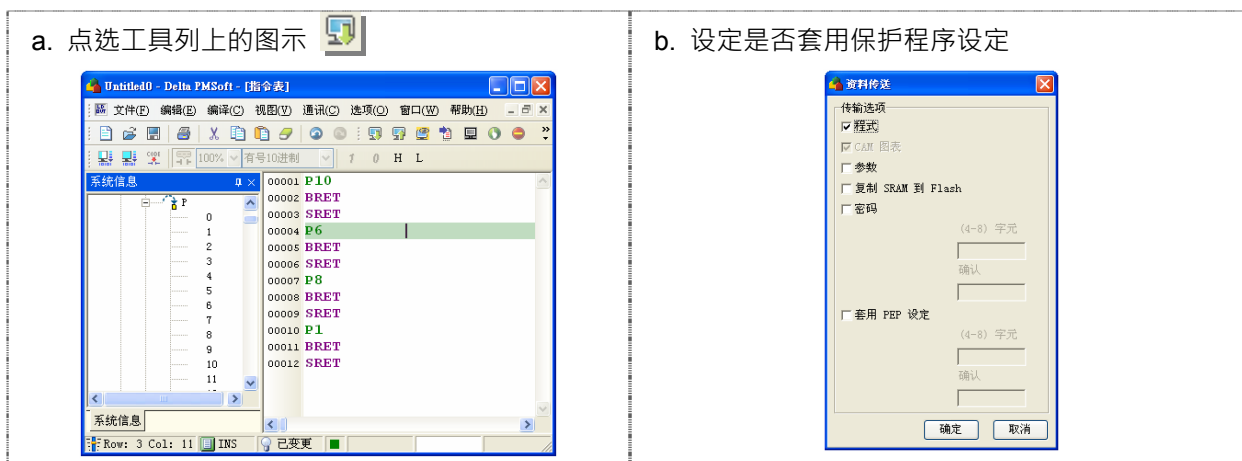
# 7 程序加密设定

3. 在第 2 步骤执行传输时，若之前已有设定过加密保护程序，PMSoft 会要求输入前一次设定加密保护的密码(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)。

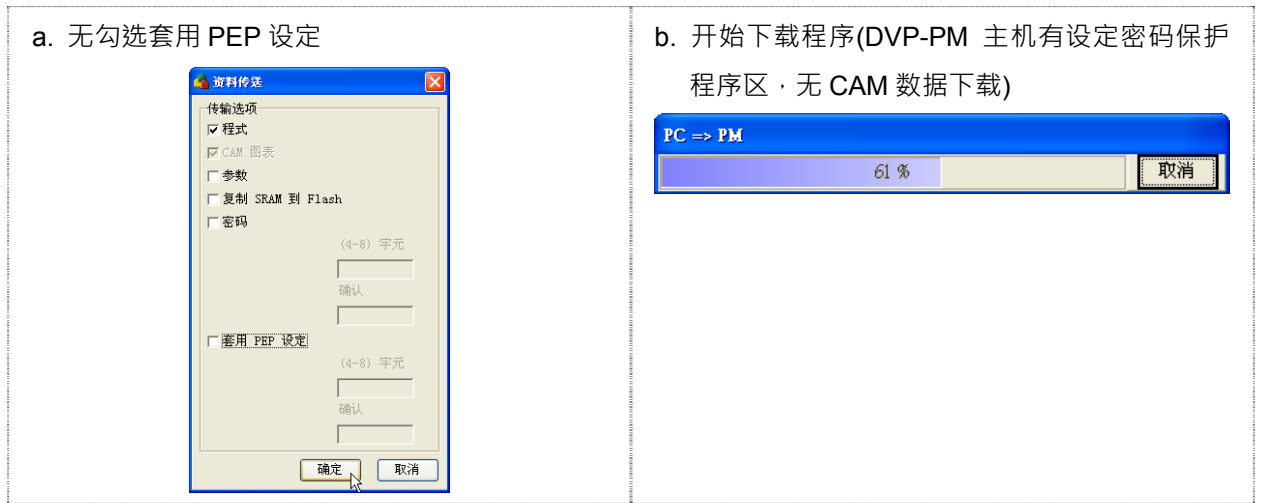


2. 无勾选套用 PEP 设定(如下图 a)，若之前 DVP-PM 主机有设定 PEP 保护密码而未解除，且下载程序有 CAM 数据，PMSoft 会要求输入之前设定的密码 (解除密码以供 CAM 数据存放)，在输入密码按确定之后(如下图 b)，即开始执行下载程序到 DVP-PM 主机(如下图 c)。

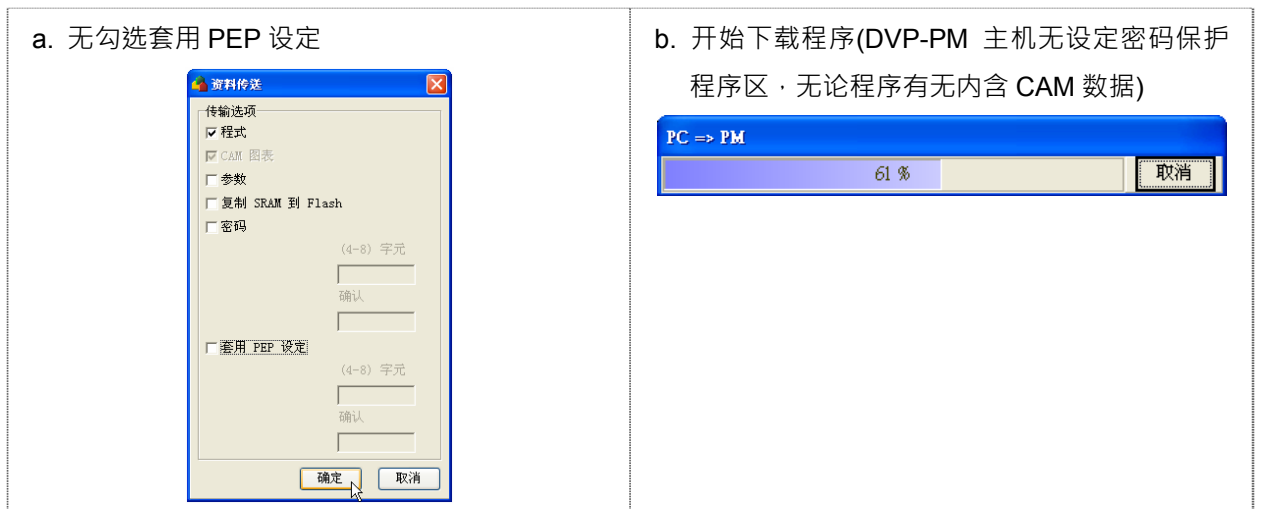




3. 在第 2 步骤选取确定数据传送时，若之前 DVP-PM 主机有设定 PEP 保护密码而未解除，而下载程序无 CAM 数据，系统将直接下载程序到 DVP-PM 主机(如下图 b)。



4. 在第 2 步骤选取确定数据传送时，若之前 DVP-PM 主机无设定 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)。




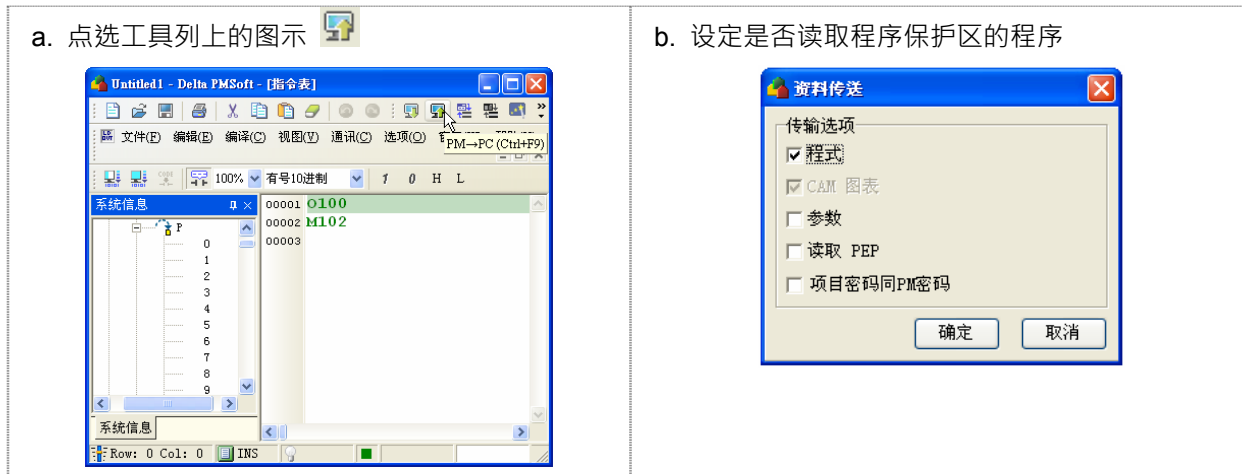
### 7.1.3 上传程序设定

上传程序时，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，则会要求输入设定的加密密码，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定操作步骤：

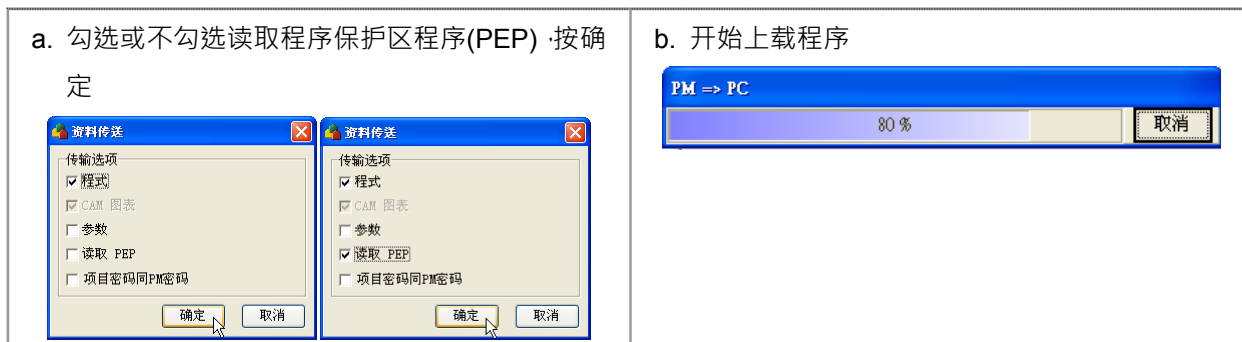
# 7 程序加密设定

## ■ DVP-PM 主机无设定保护程序：


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)。

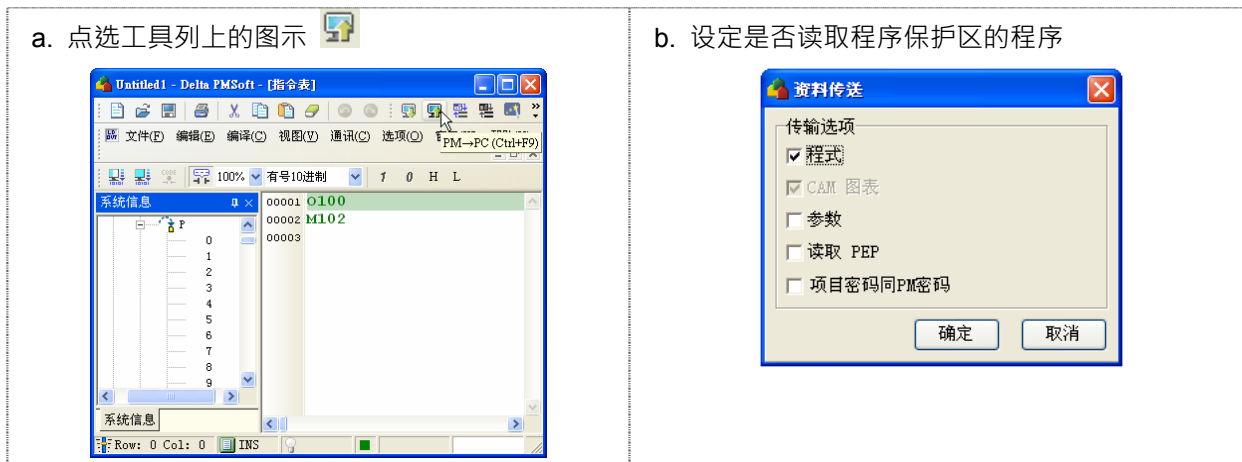


2. 无论是否有勾选读取程序保护区程序(PEP) · 按确定后(如下图 a) · 将直接读取一般程序或 CAM 数据至 PC (如下图 b)。



## ■ DVP-PM 主机有设定特定保护程序但无 CAM 数据：

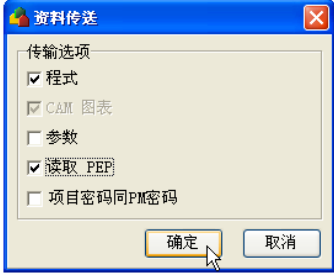
1. 选择『通讯 ( C )』功能并点选 PM→PC(U) · 或在工具列上点选图示  (如下图 a) · 会显示功能窗口(如下图 b)。




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定保护程序至 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

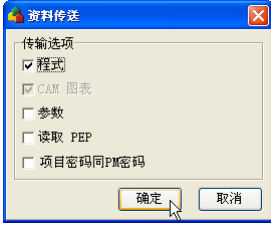


c. 开始上传程序




3. 若在第 2 步骤无勾选读取特定保护程序(PEP) · 按确定键后(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 设定




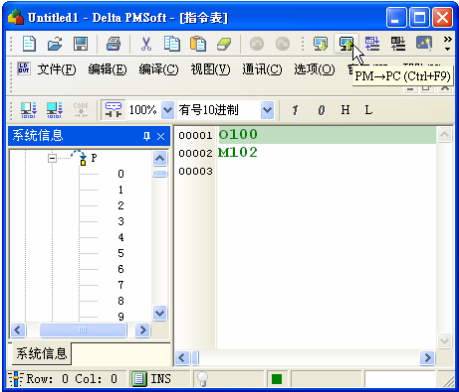
b. 系统上传程序(无设定读取密码保护程序区)



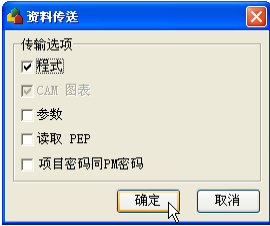
■ DVP-PM 主机有设定特定保护程序且含有 CAM 数据：

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 



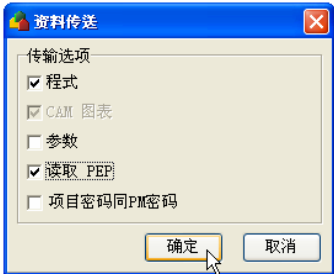
b. 设定是否读取程序保护区的程序



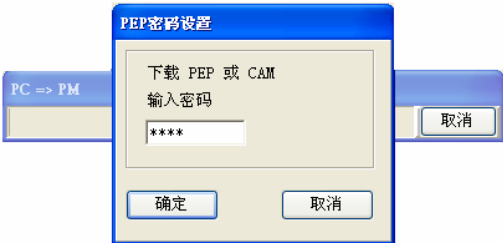
## 7 程序加密设定

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



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 程序编辑只适用梯形图编辑环境，其主要两大功能：

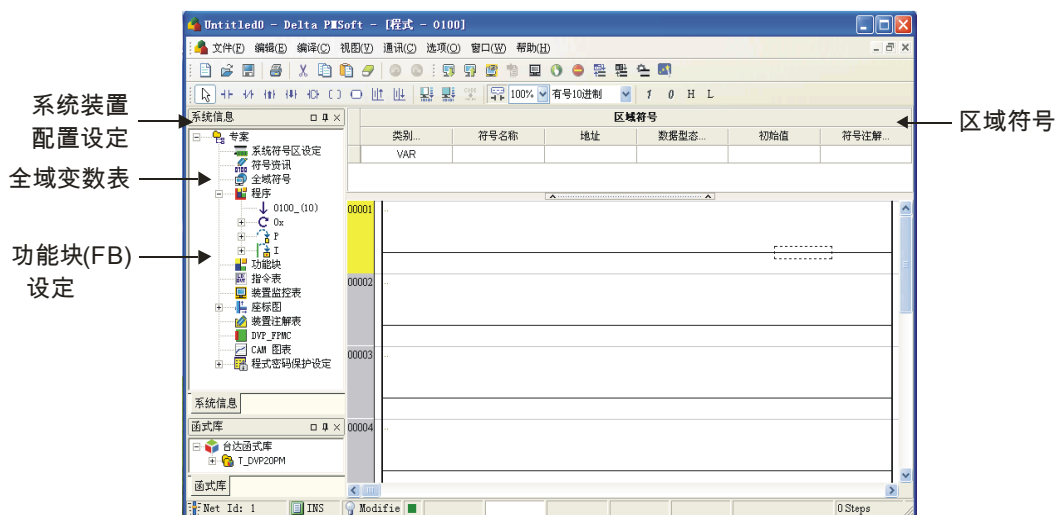
功能一：使用者可以编辑特殊功能块，简称 FB(Function Block)，FB 用于 O100 主程序、OXn 运动程序、Pn 子程序和 FB 程序中。PMSOft 可将 FB 程序汇入和汇出。

功能二：梯形图程序中的装置可使用符号取代。梯形图中的符号及 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。

## 8.1 POU 梯形图编辑模式环境

PMSOft 开新项目或开启旧项目，进入梯形图模式的编辑环境，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。



### 8.1.1 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：

# 8 POU 梯形图编辑模式

## 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。

区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
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)支持四种类别: VAR、INPUT、OUTPUT 和 INOUT。

VAR：区域符号只限定在功能块内部使用

INPUT：储存由外部输入到功能块的数据

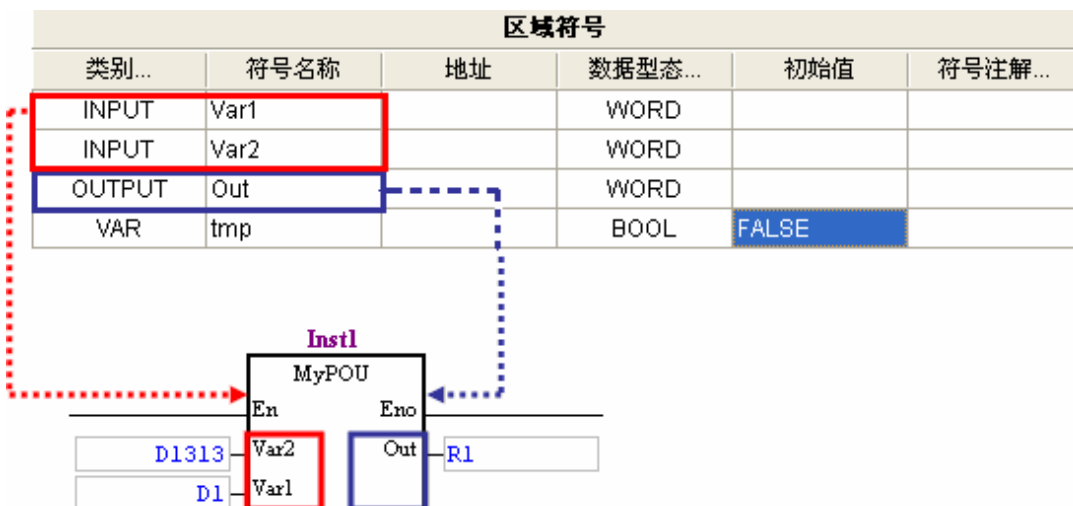
OUTPUT：将功能块内运算的数据输出到外部

INOUT：将输入到功能块的数据经运算过后输出到外部


区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	DATA		WORD	0	
INPUT	FINISH		BOOL	FALSE	
OUTPUT	Tn		TIMER	0	

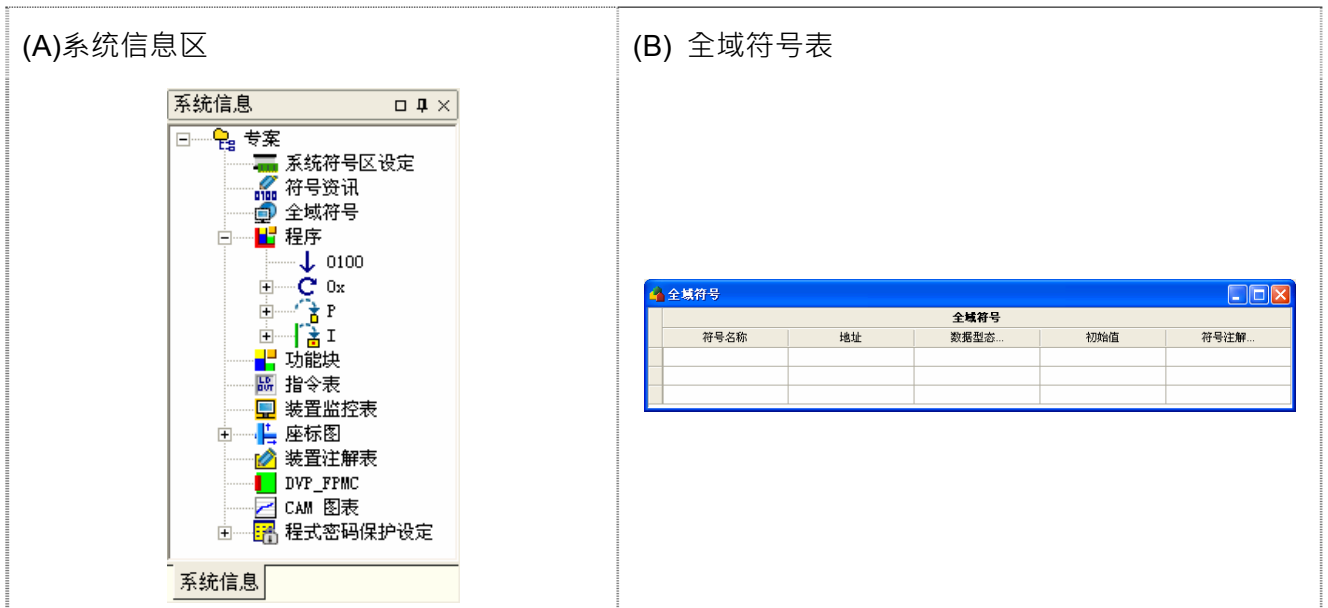
(2)「地址」字段可保留空白。

(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。



■ 符号命名规则：下图方框为符号表编写符号的字段。

区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
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

# 8 POU 梯形图编辑模式

## ■ 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 码指令不支持符号
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)

区域符号					
类别...	符号名称	地址	数据型态...	初始值	符号注解...
VAR	SHIFT_V	V0	WORD	0	
I	VAR	SHIFT_Z	DWORD	0	

### 8.1.2 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：





## ■ 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。





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**

区域符号					
类别...	符号名称	地址	数据型态...	初始值	符号注解...
VAR	Test1		WORD	0	
VAR	Start		BOOL	FALSE	
VAR					

**Step 2: 点选 **



区域符号					
类别...	符号名称	地址	数据型态...	初始值	符号注解...
VAR	Test1		WORD	0	
VAR	Start		BOOL	FALSE	
VAR					

**Step 3: 点选类别**

区域符号					
类别...	符号名称	地址	数据型态...	初始值	符号注解...
VAR	Test1		WORD	0	
VAR	Start		BOOL	FALSE	
I	INPUT				
000	INOUT				
	INOUT				
	OUTPUT				

**Step 4: 结果**

区域符号					
类别...	符号名称	地址	数据型态...	初始值	符号注解...
VAR	Test1		WORD	0	
VAR	Start		BOOL	FALSE	
I	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、数据型态:符号的数据型态,其分为基本及功能块(Function Block)。使用者可直接以键盘输入或点选数据型态字段(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: 点选数据型态字段**

区域符号					
类别...	符号名称	地址	数据型态...	初始值	符号注解...
VAR	Test1		WORD	0	
I	INPUT	Start			

**Step 3: 选择数据型态**

**数据型态**


数据型态项目

基本型态     功能块

---

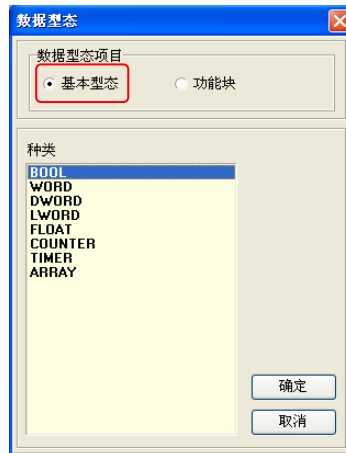
种类

BOOL  
 WORD  
 DWORD  
 LWORD  
 FLOAT  
 COUNTER  
 TIMER  
 ARRAY

**Step 2: 点选 **

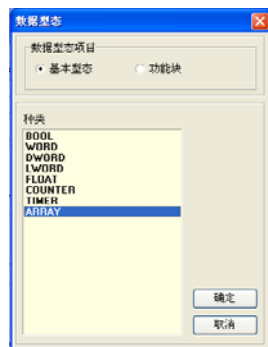
区域符号					
类别...	符号名称	地址	数据型态...	初始值	符号注解...
VAR	Test1		WORD	0	
I	INPUT	Start			

## d.1 基本型态：一般装置符号宣告

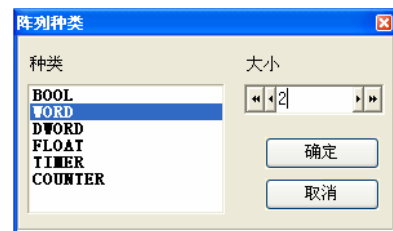


- **BOOL**：宣告符号型态是一位(BIT)，系统配置为 M 装置。
- **WORD**：宣告符号型态是 16 位，系统配置 D 装置。
- **DWORD**：宣告符号型态是 32 位，系统配置连续 2 个 D 装置。
- **LWORD**：宣告符号型态是 64 位，系统配置连续 4 个 D 装置。
- **FLOAT**：宣告符号型态是 32 位浮点数，系统配置为 2 个 D 装置，用于浮点数指令。
- **COUNTER**：宣告符号型态是 16 位计数装置，系统配置 C 计数装置。
- **TIMER**：宣告符号型态是 16 位计时装置，系统配置 T 装置。
- **ARRAY**：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



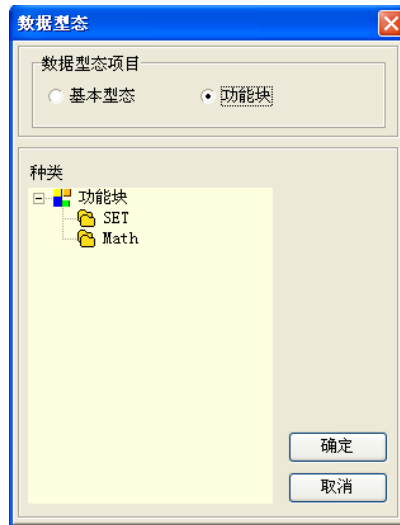
### Step 2：选择数据型态、设定长度



### Step 3：结果

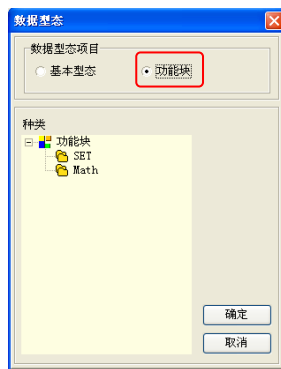
区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	Test1		WORD	0	
VAR	Start		BOOL	FALSE	
• VAR	data		WORD[2]		

d.2 功能块型态：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。

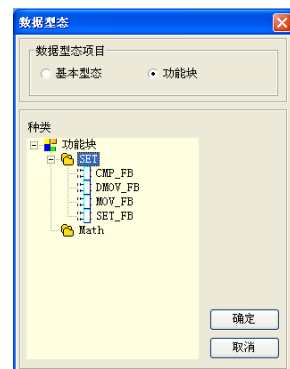


使用功能块一定要将功能块作符号宣告，下列三个步骤为功能块符号宣告：

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

区域符号						
	类别...	符号名称	地址	数据型态...	初始值	符号注解...
I	VAR	Test1		WORD	0	
	VAR	Start		BOOL	FALSE	
	VAR	data		WORD[2]		
	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、 符号注解: 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。

# 8 POU 梯形图编辑模式

1. 当数据型态为 BOOL，则初始值需设定为 TRUE 或 FALSE。

区域符号						
	类别...	符号名称	地址	数据型态...	初始值	符号注解...
I	VAR	Bit		BOOL	FALSE	

2. 当数据型态为 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)。

区域符号						
	类别...	符号名称	地址	数据型态...	初始值	符号注解...
	VAR	Word1	D7000	WORD	100	
	VAR	Word2	D7001	WORD	K100	
I	VAR	Word3	D7002	WORD	H200	

3. 当数据型态为数组型态时，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。

**阵列初始值设定**

阵列数据型态: WORD [4]

位置	初始值
[0]	1
[1]	1
[2]	0
[3]	0

执行后

区域符号						
	类别...	符号名称	地址	数据型态...	初始值	符号注解...
	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)

区域符号						
	类别...	符号名称	地址	数据型态...	初始值	符号注解...
I	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类型，如下图修改符号类别。

区域符号						
类别...	符号名称	地址	数据类型...	初始值	符号注解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
VAR						

注 1: 功能块型态的 POU，不允许宣告数据类型为功能块的符号

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

区域符号						
类别...	符号名称	地址	数据类型...	初始值	符号注解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
VAR	Finish		BOOL	FALSE		
VAR	Data		WORD[4]	[1(2),0(2)]		
VAR	ADD		ADD_FB			


#### 2. 按组合键 [Alt + ↓] 或 [Alt + ↑]，将符号向上移动或向下移动

区域符号						
类别...	符号名称	地址	数据类型...	初始值	符号注解...	
VAR	Test1		WORD	0		
VAR	Start		BOOL	FALSE		
VAR	Data		WORD[4]	[1(2),0(2)]		
VAR	Finish		BOOL	FALSE		
VAR	ADD		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：


# 8 POU 梯形图编辑模式

**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)。

**Step 1：点选预删除符号列**


全域符号				
符号名称	地址	数据类型...	初始值	符号注解...
GA1		WORD[12]		
GA2		WORD		
GA3		CMP_FB		
 GA4		COUNTER		

**Step 2：删除(选择四种方式之一)**

全域符号				
符号名称	地址	数据类型...	初始值	符号注解...
GA1		WORD[12]		
GA2		WORD		
 GA3		CMP_FB		
GA4		COUNTER		

**Step 3：结果**

全域符号				
符号名称	地址	数据类型...	初始值	符号注解...
GA1		WORD[12]		
GA2		WORD		
 GA4		COUNTER		

**多列删除**：光标点选要删除起始列，该列最左边显示  图形，按组合键 [ Shift+ 光标点选删除终止列 ]，使用上面所示四种方法之一，删除指定连续多列符号(如下图所示删除GA2 至GA5 符号)。

**Step 1：点选预删除起始列**

区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	GA1		WORD[12]		
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+ 游标点选删除终止列**

区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	GA1		WORD[12]		
VAR	GA2		BOOL		
 VAR	GA3		WORD		
VAR	GA4		BOOL		
VAR	GA5		TIMER		
VAR	GA6		COUNTER		
VAR	GA7		WORD		
VAR	GA8		FLOAT[5]		

**Step 3：删除(选择四种方式之一)**

区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	GA1		WORD[12]		
VAR	GA2		BOOL		
 VAR	GA3		WORD		
 VAR	GA4		BOOL		
 VAR	GA5		TIMER		
 VAR	GA6		COUNTER		
VAR	GA7		WORD		
VAR	GA8		FLOAT[5]		

**Step 4：结果**

区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	GA1		WORD[12]		
VAR	GA2		BOOL		
VAR	GA7		WORD		
 VAR	GA8		FLOAT[5]		

## ■ 符号汇出


符号汇出是把符号表中所有符号都汇出储存成一个格式为\*.csv 的档案。符号汇出的方法说明如下。

1. 鼠标光标移到符号表中，按鼠标右键，点选汇出符号表。

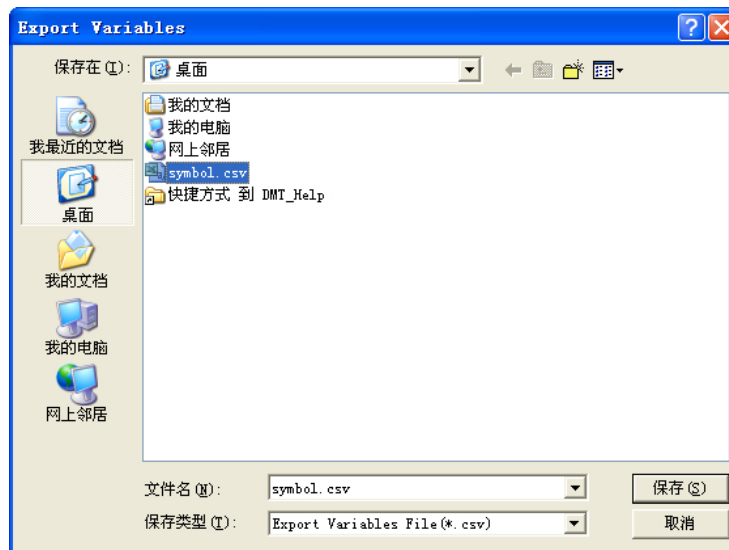
**Step 1 : 鼠标光标移至符号表中**

区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注...
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**



2. 设定符号文件名称及路径

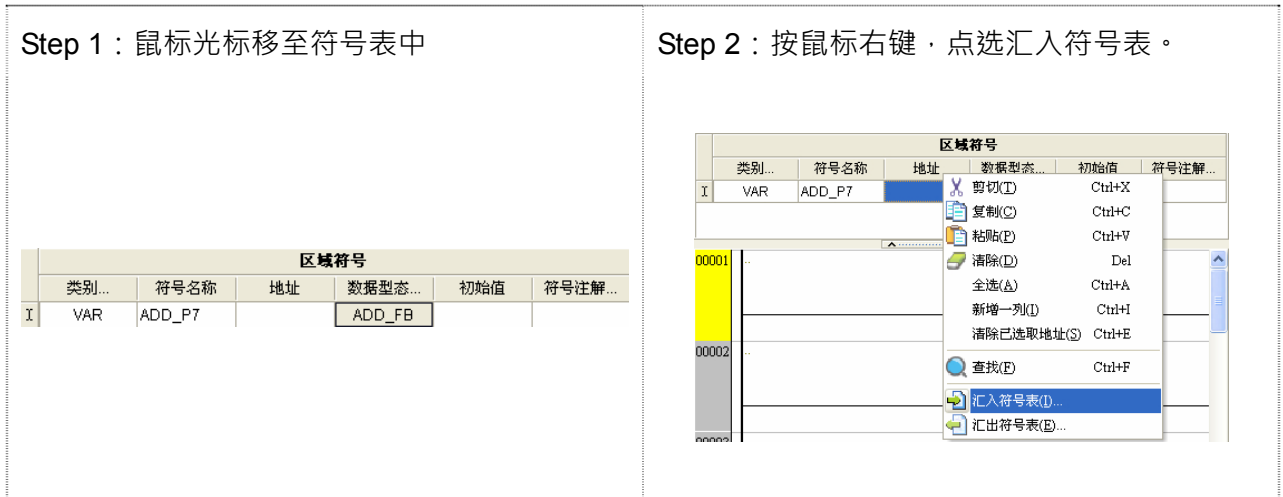


# 8 POU 梯形图编辑模式

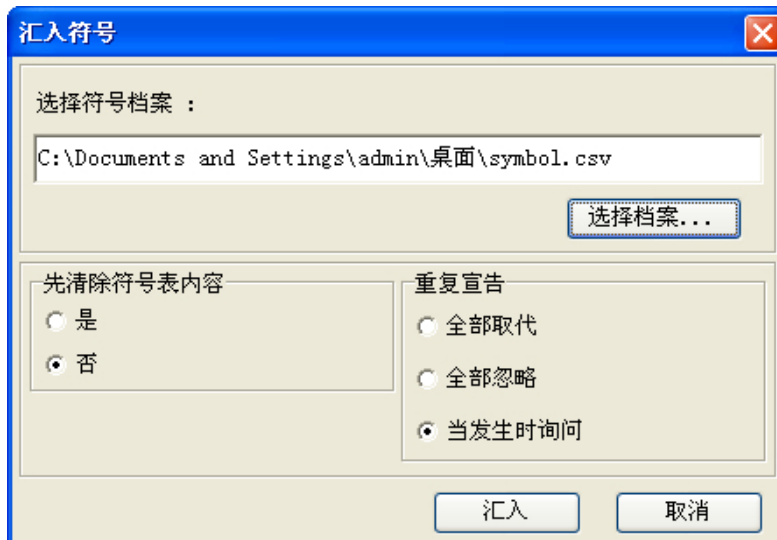
## ■ 符号汇入

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。



2. 选择汇入符号文件及汇入至符号表的方式



- 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。

### 8.1.3 新增 POU 功能块

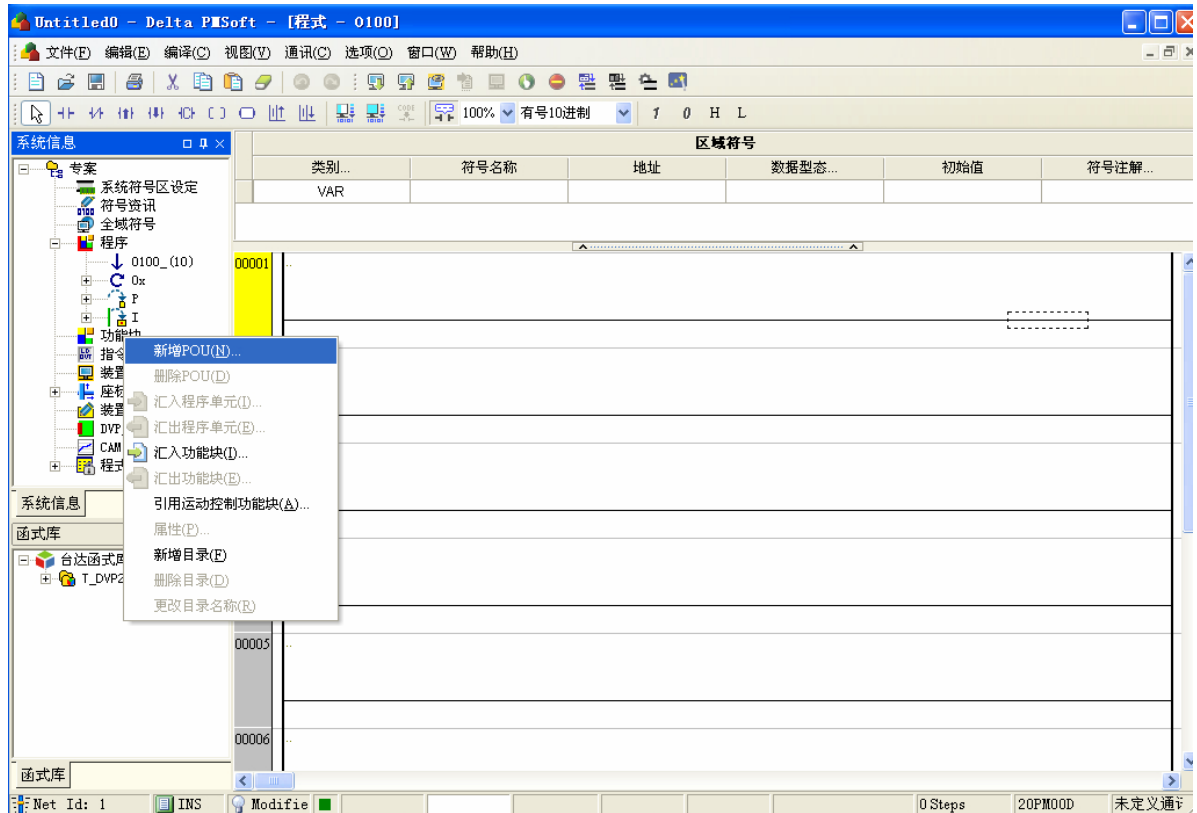
功能块能简化程序且可以重复使用。透过全域符号及功能块区域符号表中类别之 INPUT 定义输入符号，作为功能块运作所需数据来源输入引脚接口，而类别 OUTPUT 定义符号为输出，为功能块运作结果输出之引脚接口，使用者可针对自己的需求设计功能块的功能。



■ 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**

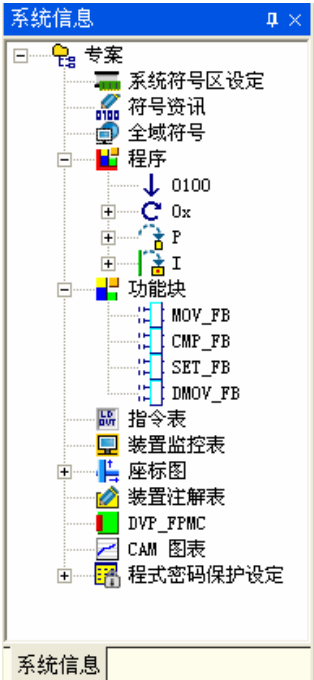
- 不允许有符号如(~!@#\$\$%^&\*())...等
- 不允许同一个项目中有相同 POU 名称
- 不区分大小写。

## 8 POU 梯形图编辑模式

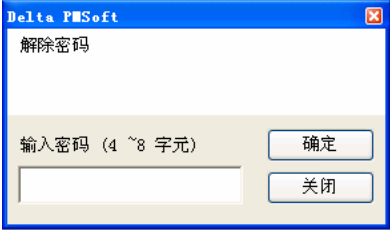
- 最多允许 20 个字符。

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，会显示密码不合法信息。

Step 1：双击 DMOV\_FB



Step 2：输入密码




Step 3：结果



区域符号						
	类别...	符号名称	地址	数据类型...	初始值	符号注解...
I	INPUT	DA1		DWORD	0	
	OUTPUT	TDATA		TIMER[2]		

**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：



属性

POU 名称  
MOV\_FB

En/Eno  
 使用 En/Eno

版本  
1.00

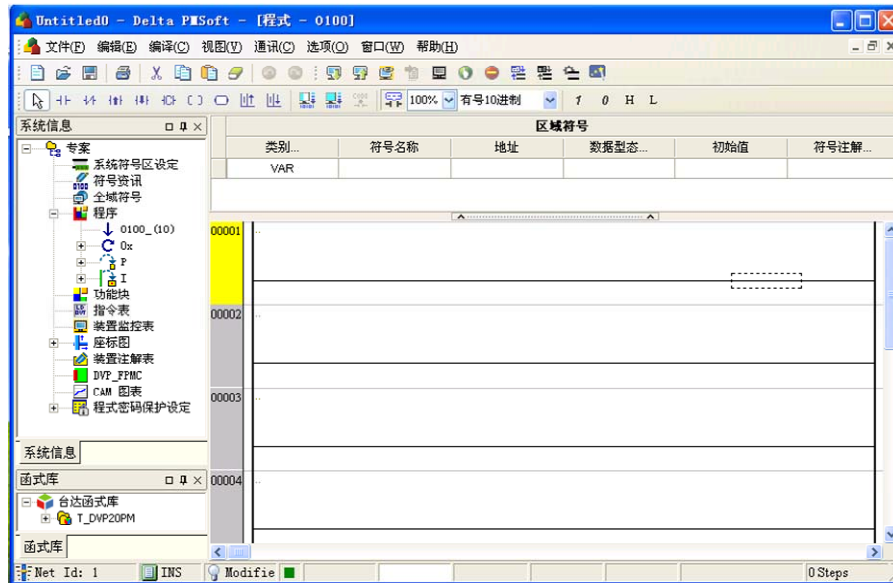
密码设定  
输入密码 <4-8 字节>  
密码确认 <4-8 字节>

语言  
 阶梯图 (LD)  
 连续功能图 (CFC)

Data movl

确定 取消

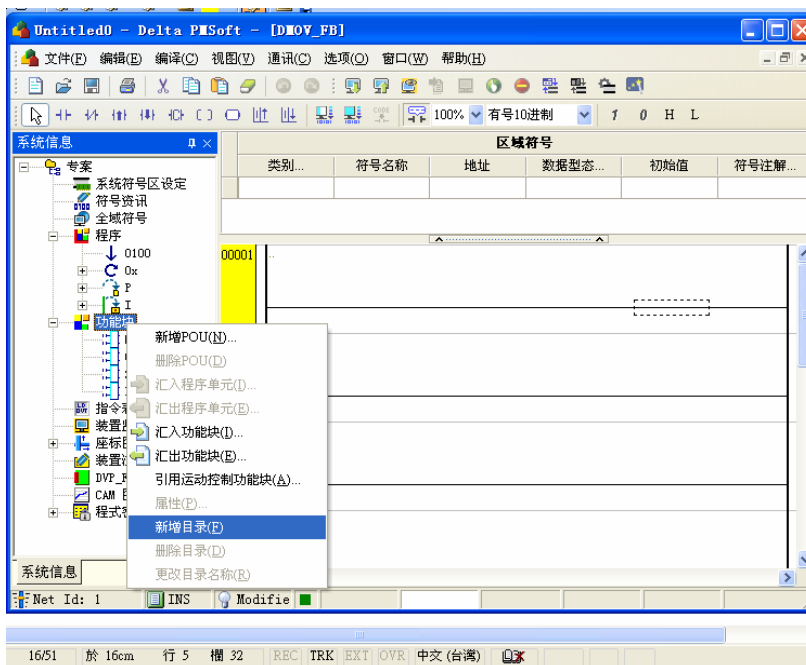
(3) 按「确定」后显示梯形图编辑画面。



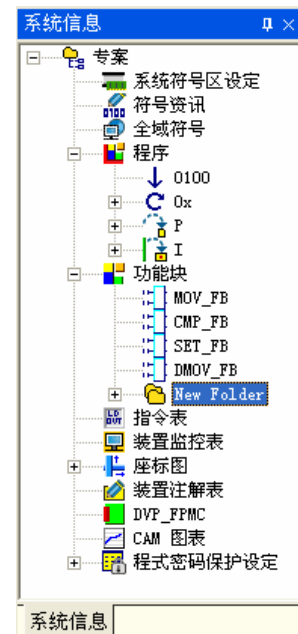
## 8.1.4 新增 POU 管理目录

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。

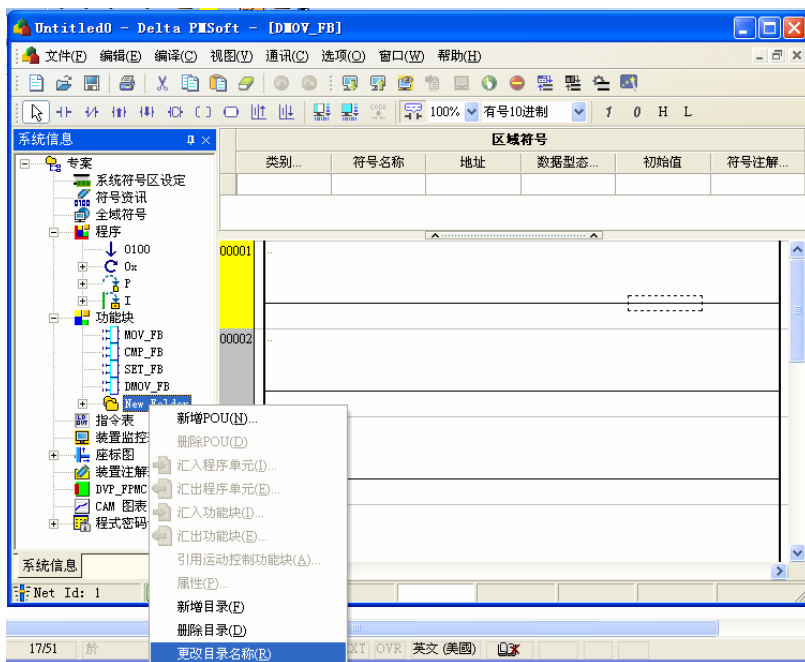


新增结果

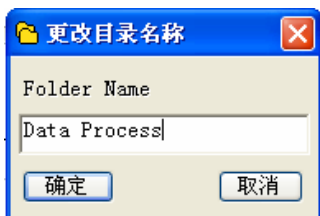


## 8 POU 梯形图编辑模式

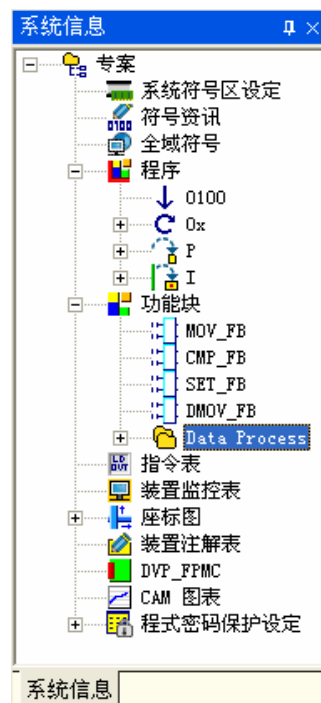
(2) 更改目录名称：光标移在「New Folder」节点上按鼠标右键，选取「更改目录名称」。



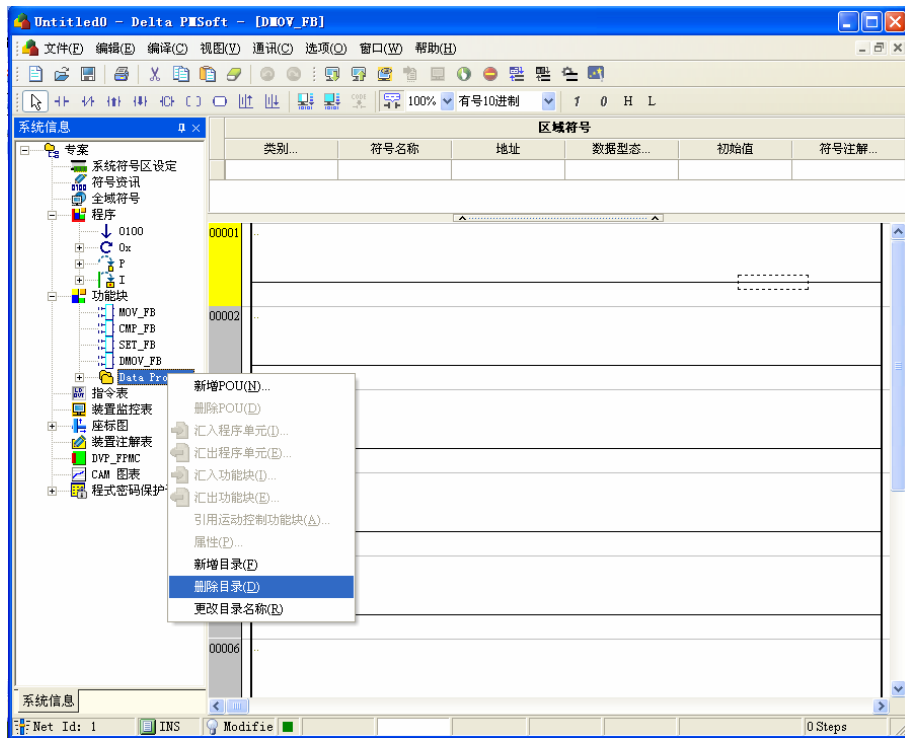
此时会显示更改目录名称窗口，在窗口内在编辑完新名称后，按下「确定」后即完成名称的更改。



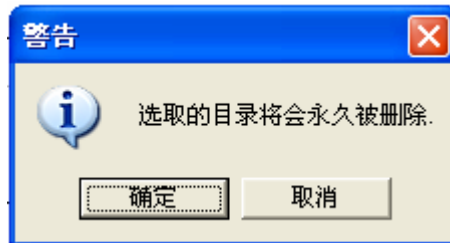
更改目录名称后的结果



(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」。

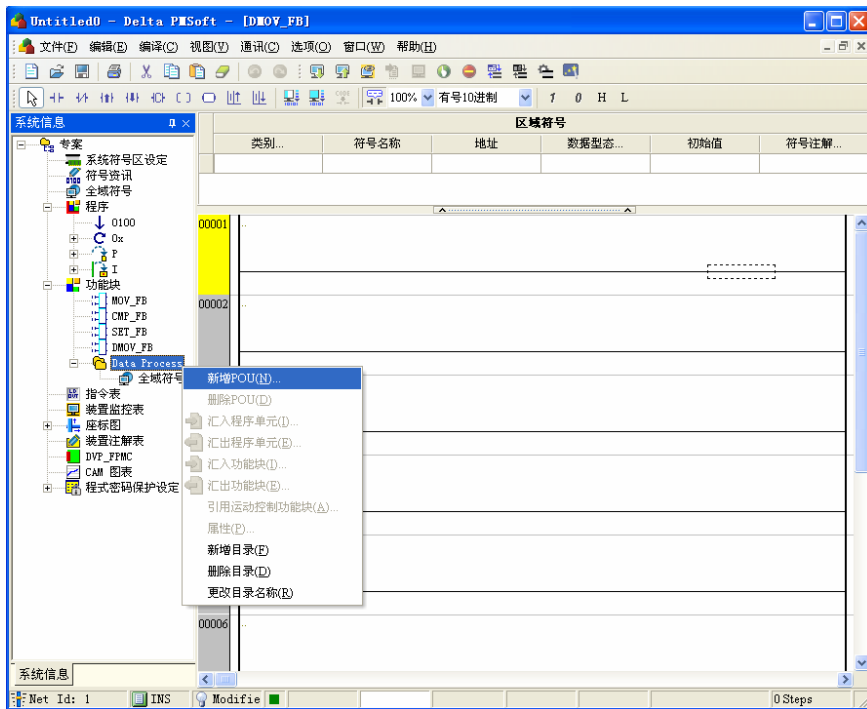


此时会出现询问窗口，若选择「确定」，则所选取的目录将会被删除。



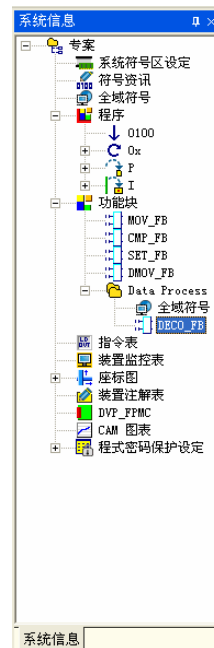
# 8 POU 梯形图编辑模式

(4) 在目录中新增 POU：在目录节点上点鼠标右键选择「新增 POU」



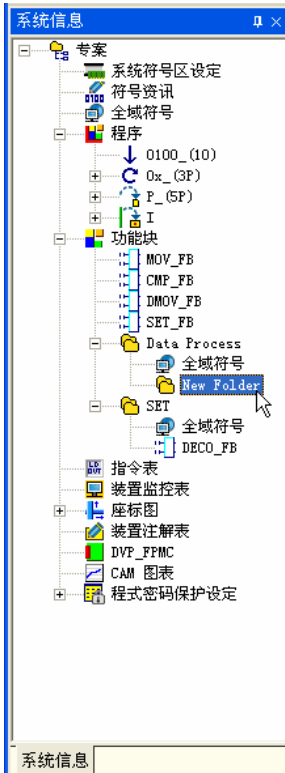
即显示「建立功能块」窗口，命名规则参考先前介绍的 POU 命名规则，将名称键入后点选确定即可在该目录下看见新增 POU 的结果。

新增结果

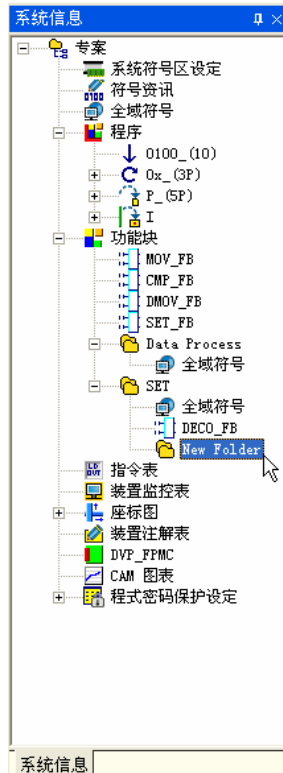


(5) 鼠标拖曳移动节点：使用鼠标按住欲移动的目录或是 POU 拖拉至欲放置的目录，即可将 Function Blocks 节点下的目录或 POU 节点搬移到其它的目录节点下。

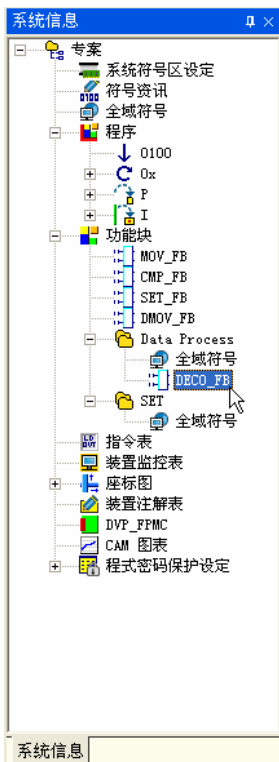
移动目录至另外一个目录底下



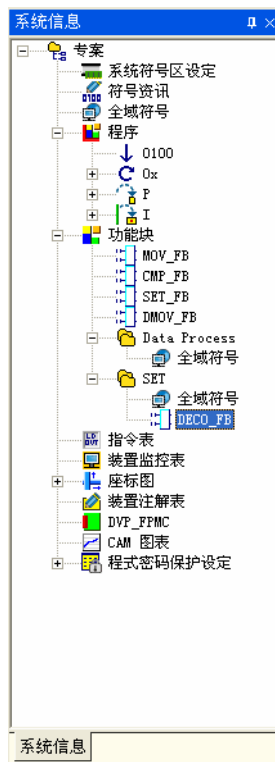
移动结果



移动 POU 至目录底下



移动结果



## 8 POU 梯形图编辑模式

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，会有警告信息产生告知已有相同的目录名称。

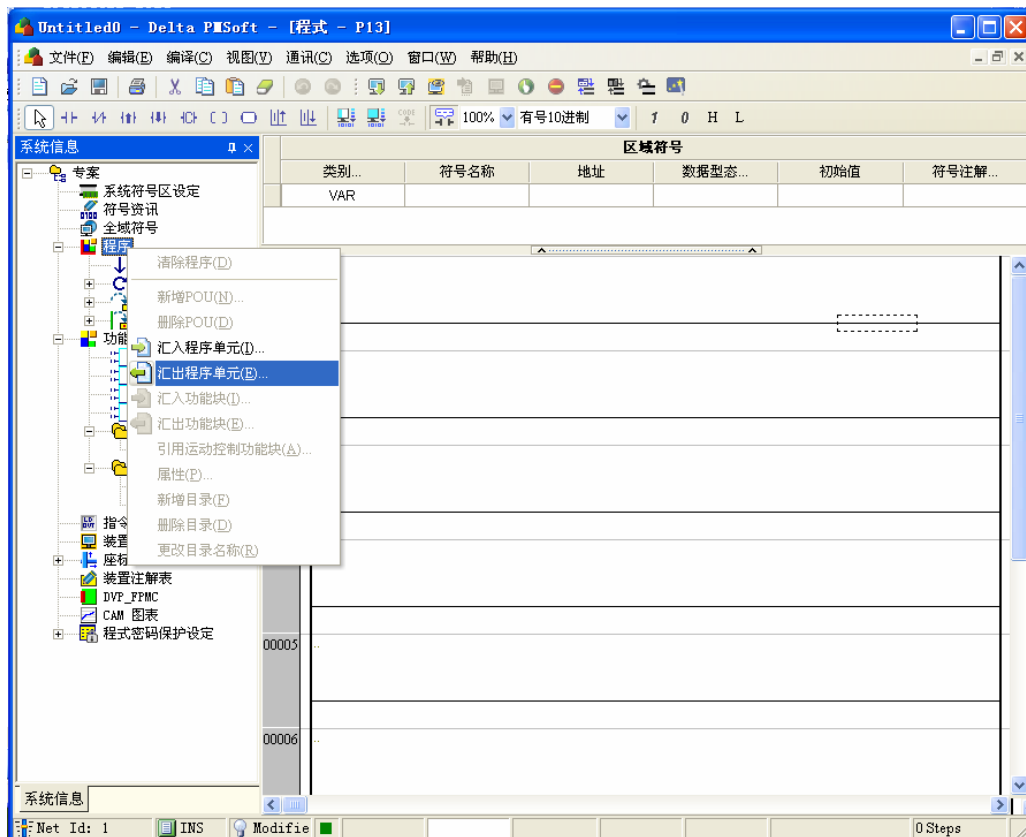


### 8.1.5 汇出 POU 功能块

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：

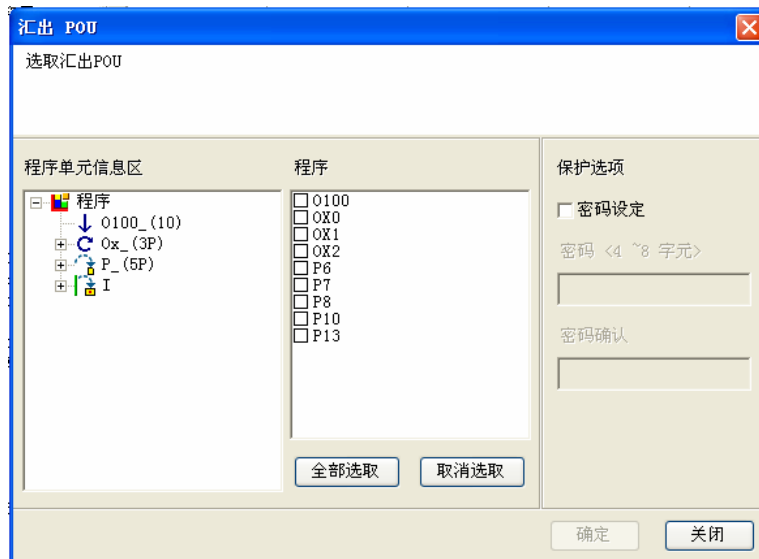
- 程序型态 POU 汇出：

1. 在「系统信息区」上，光标移在「程序」节点上按鼠标右键，选取「汇出程序单元」。

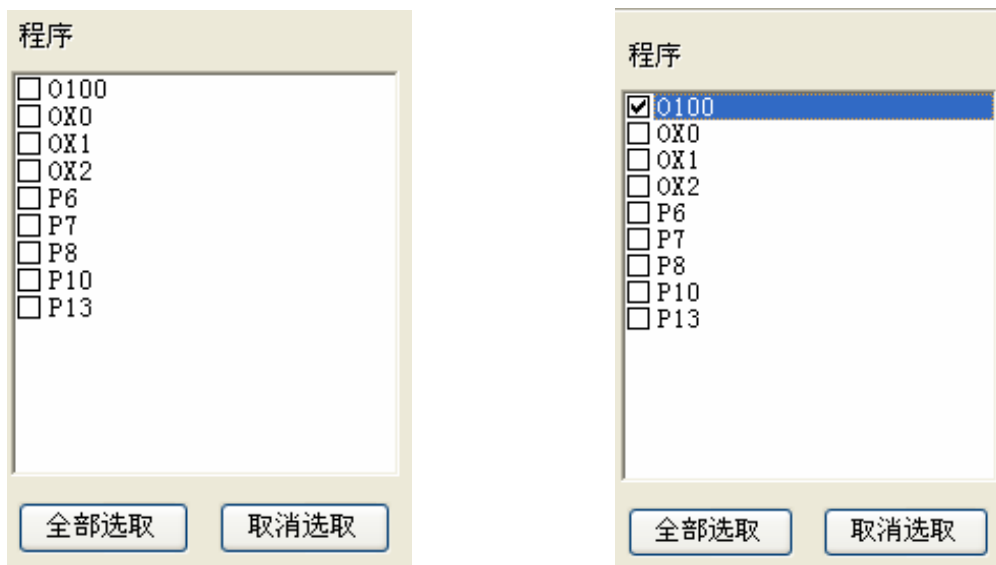




2. PMSoft 会显示程序单元信息区与程序区下所有的功能块，可勾选需要汇出之 POU。若在程序单元信息区指定区域(主程序区 O100、运动子程序区 Ox、子程序区 P)，则在程序画面仅显示该目录中的 POU。



方法一：使用鼠标点选，如下图所示选择 O100 功能块。



## 8 POU 梯形图编辑模式

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。



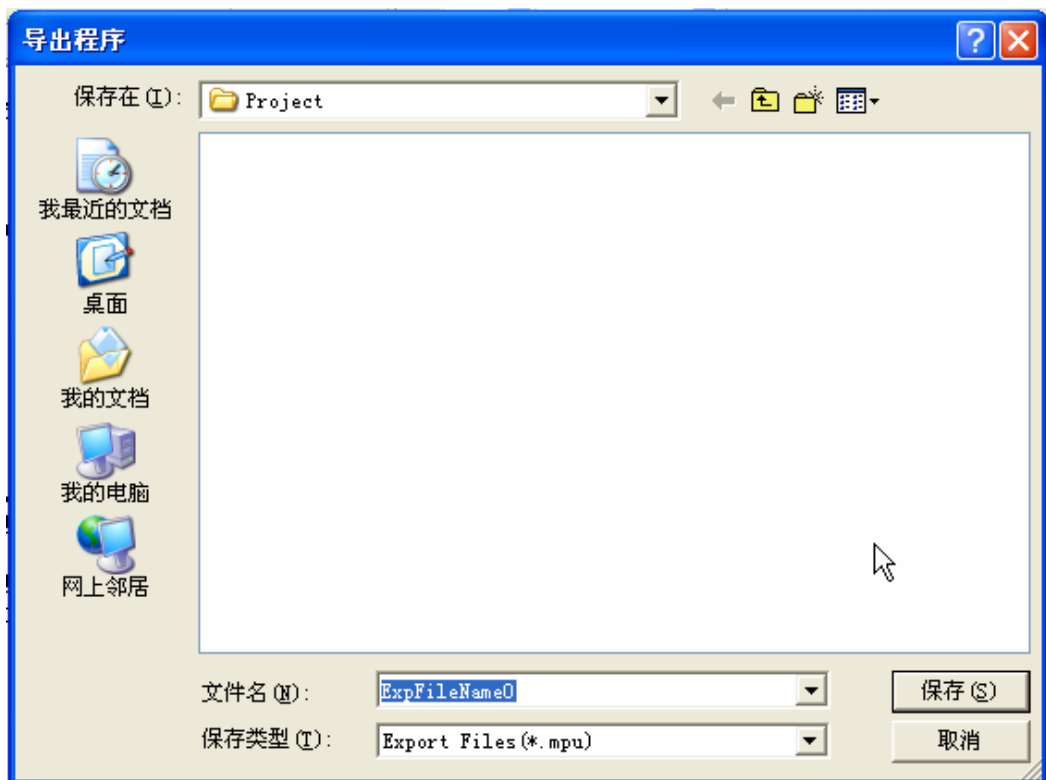
方法四：全部取消，如下图所示点选「取消选取」取消全部功能块。



3. 保护选项：功能块链接库加密保护。勾选密码设定以设定密码及确认。



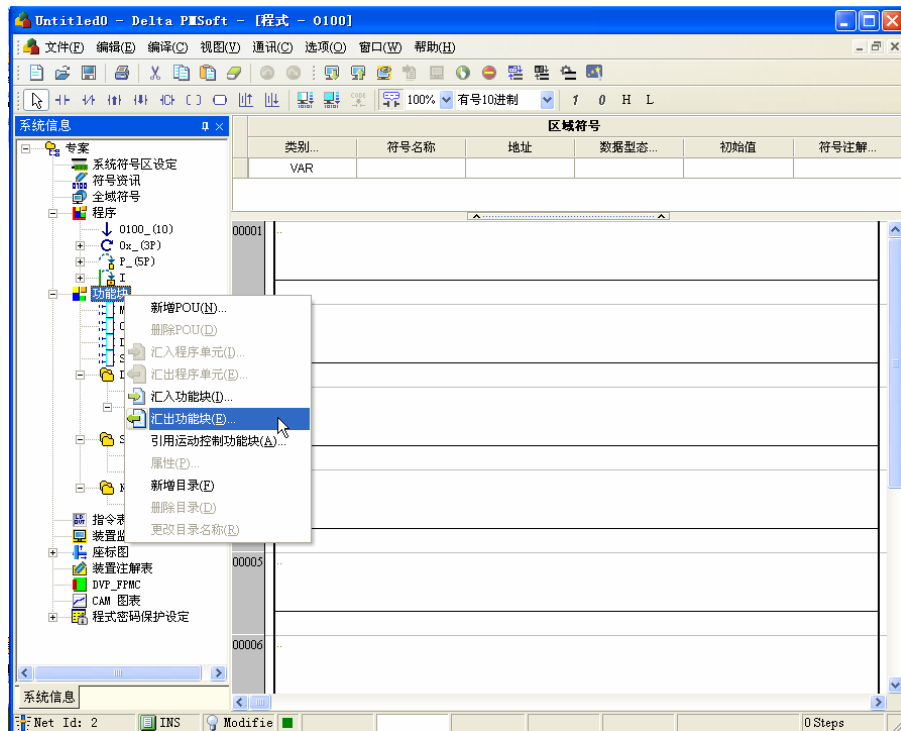
4. 点选确定后跳出存盘窗口，在此设定定文件名称及路径，扩展名为\*.mpu。



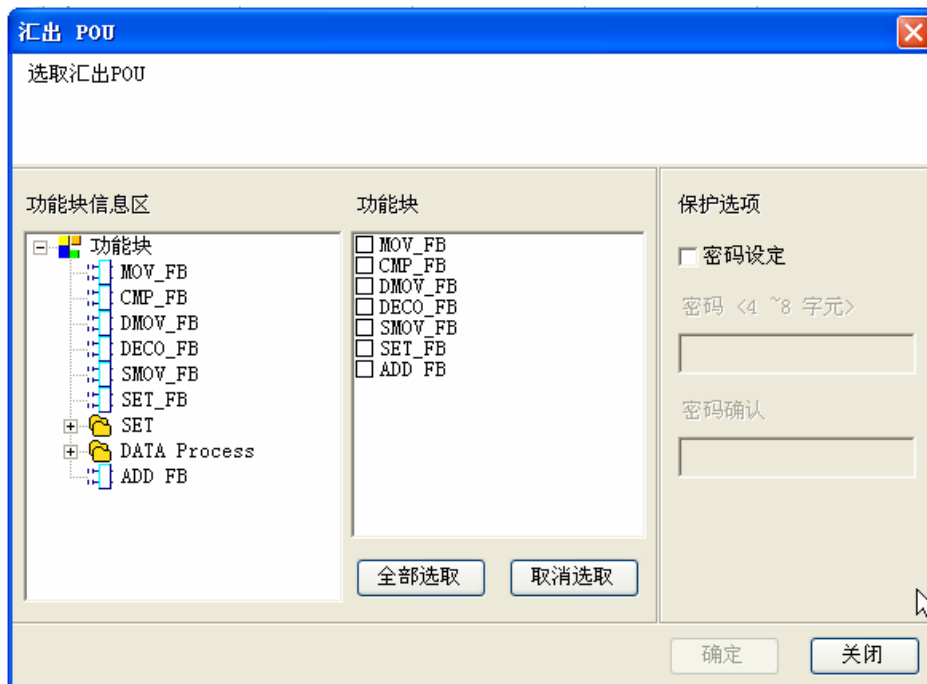
## 8 POU 梯形图编辑模式

### ■ 功能块型态 POU 汇出

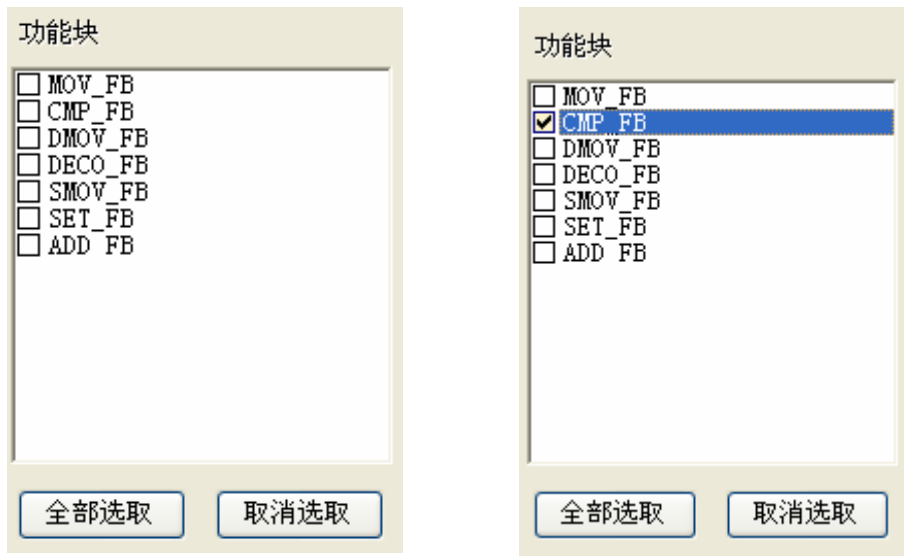
1. 在「系统信息区」上，光标移到「功能块」节点上按鼠标右键，选取「汇出功能块」。



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。



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。

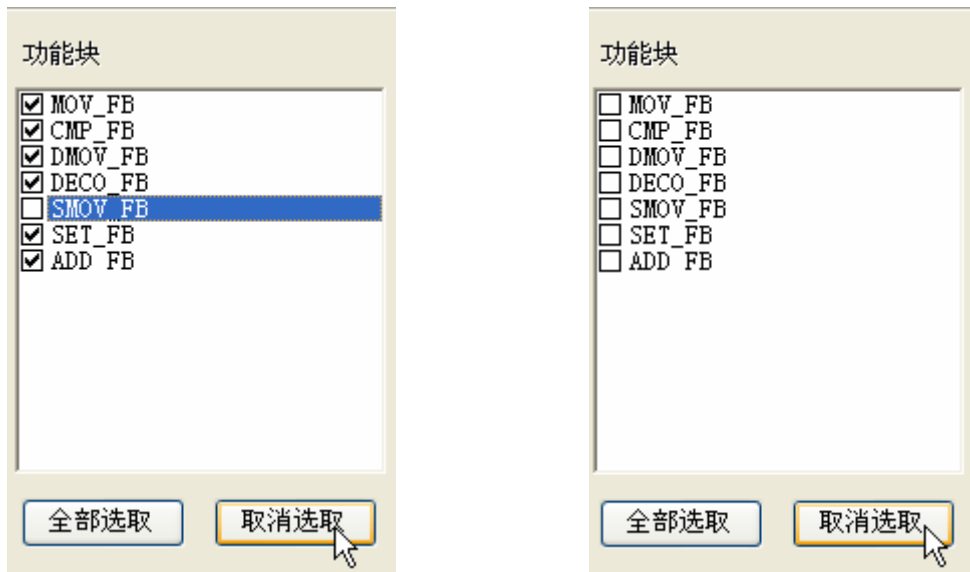


方法三：取消选择，如下图所示用鼠标点选已勾选的功能块。



## 8 POU 梯形图编辑模式

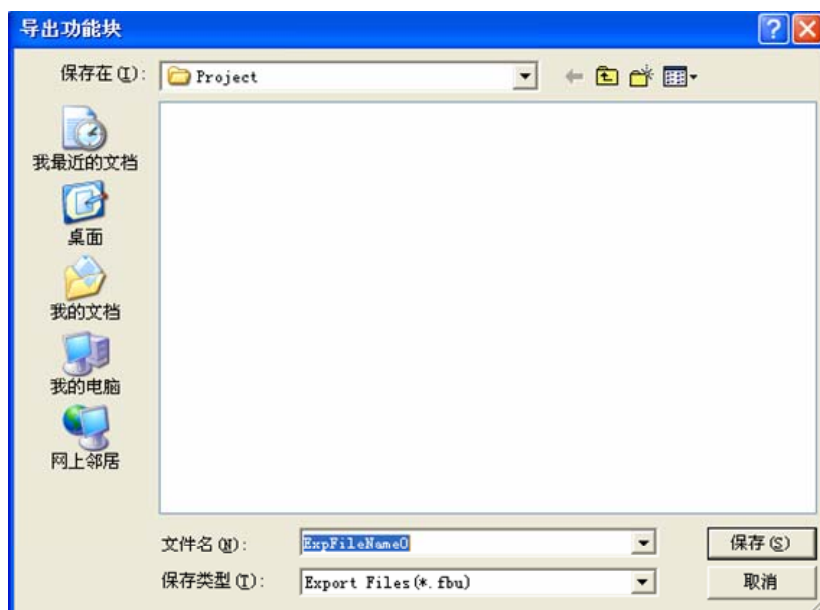
方法四：全部取消，如下图所示点选「取消选取」取消全部功能块。



3. 保护选项：功能块链接库加密保护。勾选密码设定以设定密码及确认密码。



4. 点选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。

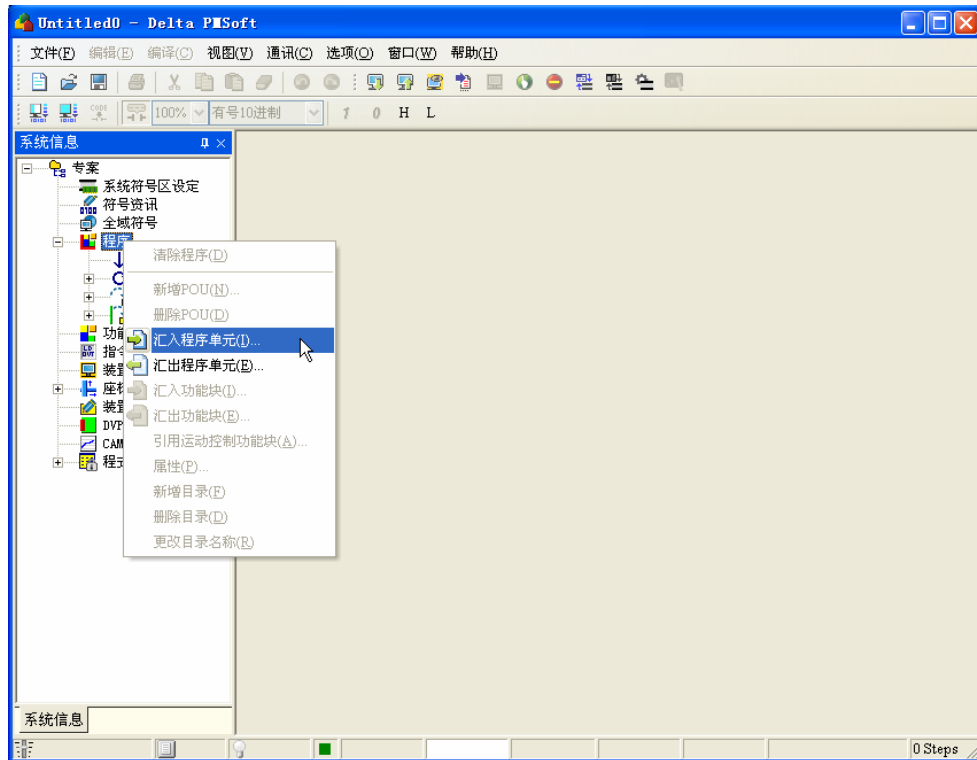


### 8.1.6 汇入 POU 功能块

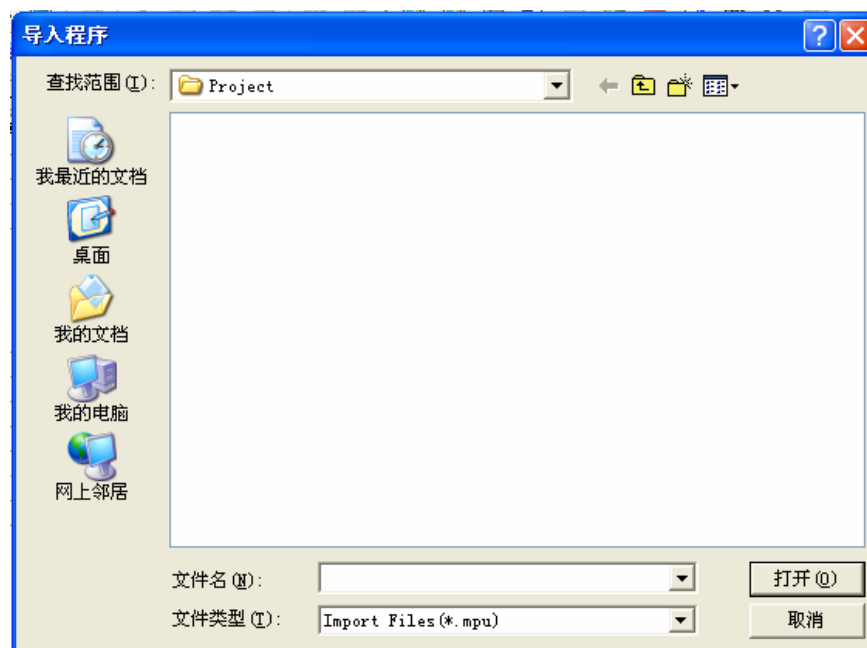
将功能块链接库汇入项目程序，免去使用者花费时间编写同样功能块，POU 的型态分为程序与功能块两种，下列说明汇入功能块方法：

- 程序型态 POU 汇入：

1. 在「系统信息区」上，光标移在「程序」节点上按鼠标右键，选取「汇入程序单元」。

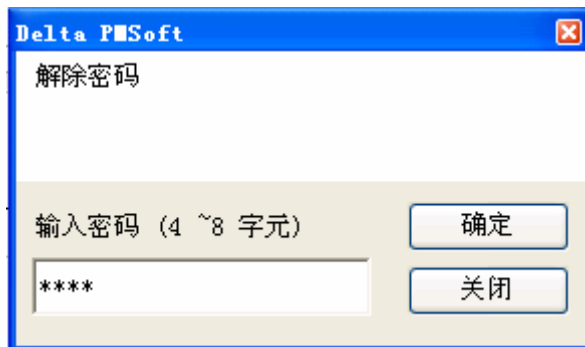


2. PMSOFT 显示安装文件里头目录中的程序单元档案(\*.mpu) 或是其它路径中的程序单元档案，选择档案，按开启。

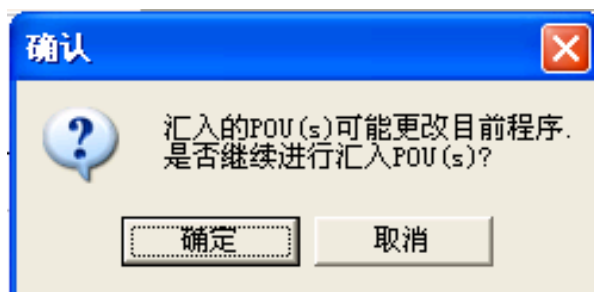


## 8 POU 梯形图编辑模式

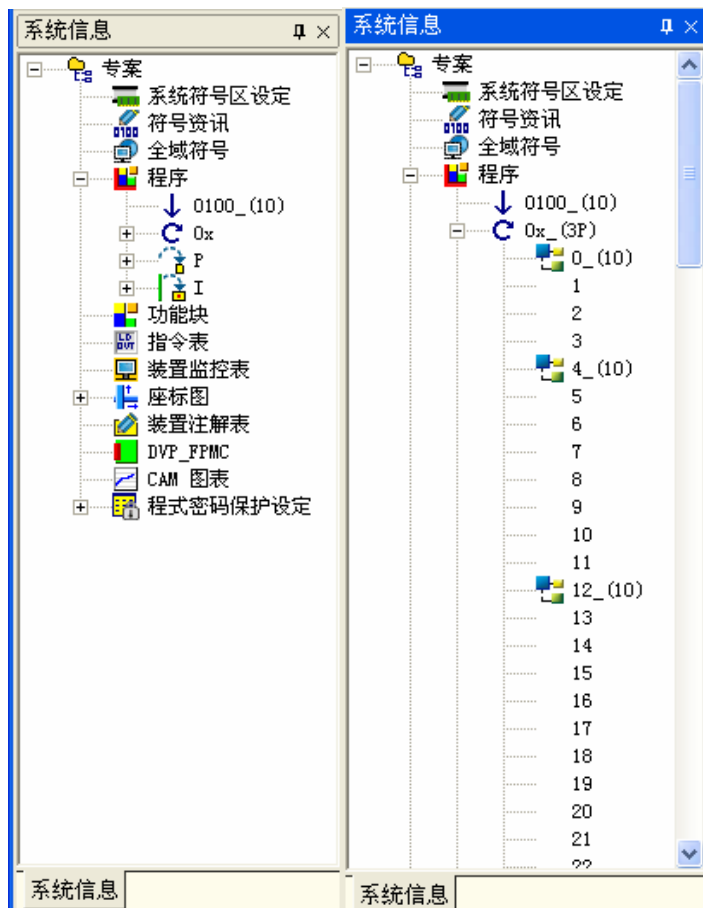
- 解除密码：若程序单元的连接库档案有加密，则如下图所示输入正确密码，才能汇入功能块。若功能块连接库档案无加密，则系统略过此步骤。



- 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。



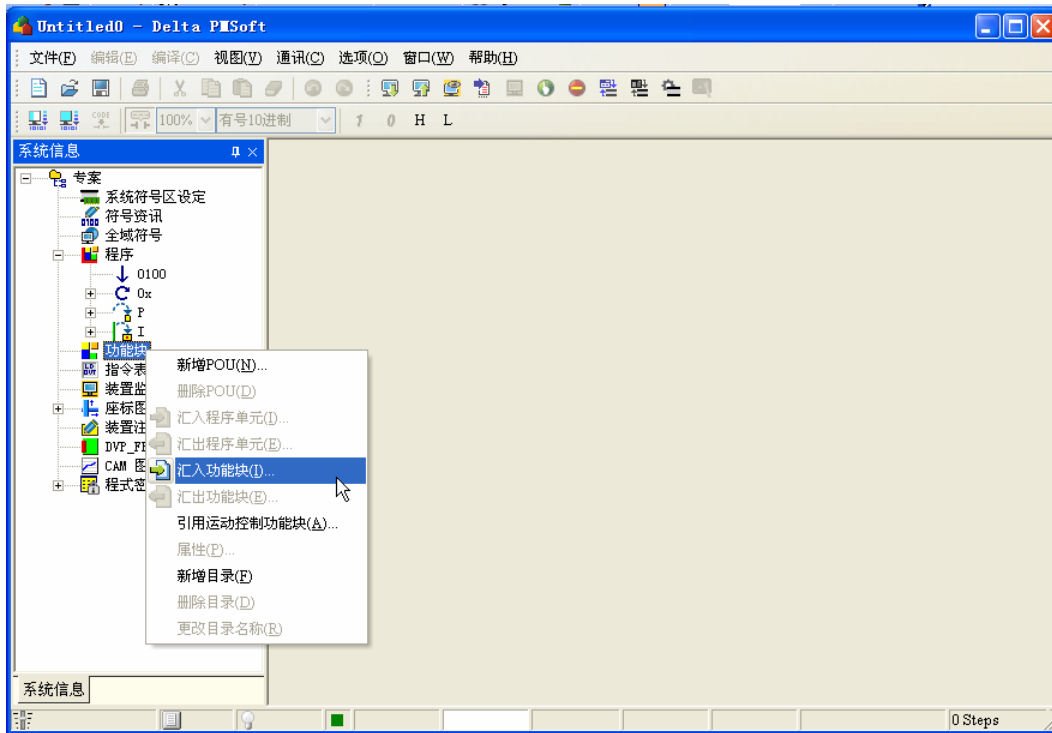
- 结果：如下图所示，汇入的程序可点选 Programs 的各项程序区中看见。



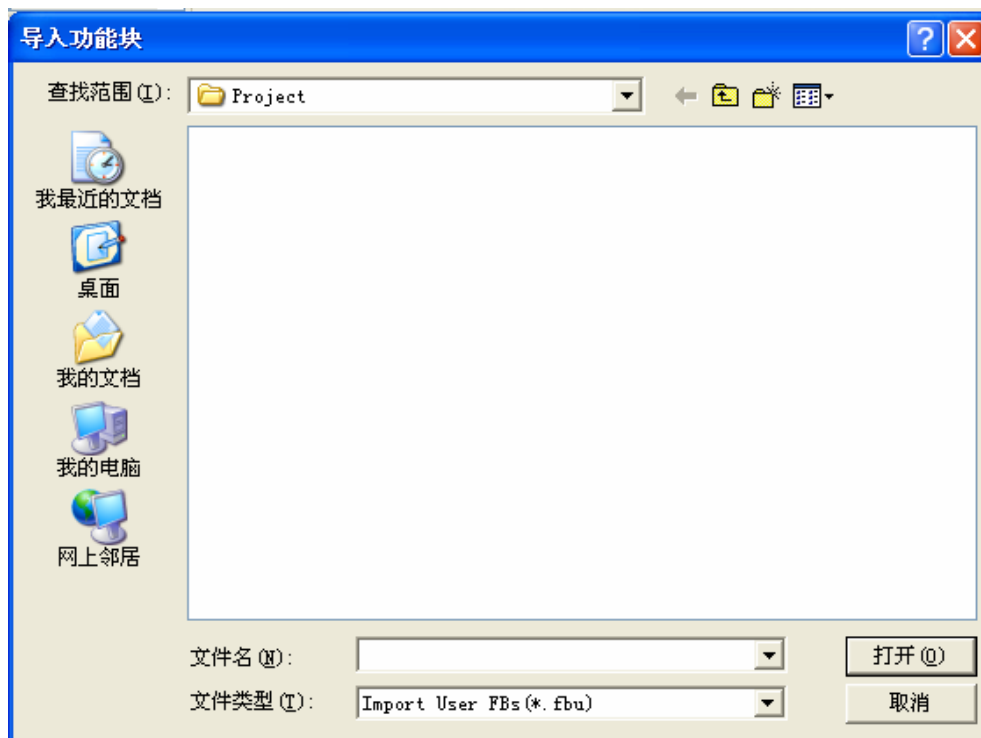


## ■ 功能块型态 POU 汇入

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」。

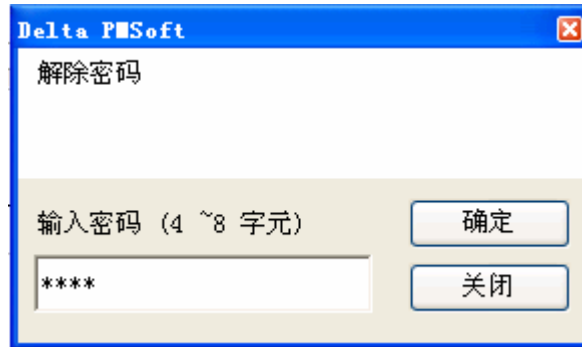


2. PMSoft 显示安装文件里头目录中的功能块文件(\*.fbu)·或是其它路径中的功能块文件·选择文件·按开启。

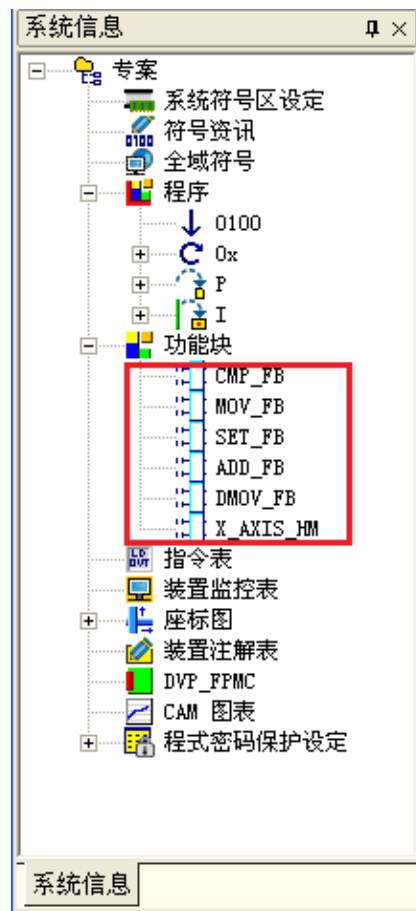


## 8 POU 梯形图编辑模式

- 解除密码：若功能块链接库文件有加密，则如下图所示输入正确密码，才能汇入功能块。若功能块链接库文件无加密，则系统略过此步骤。



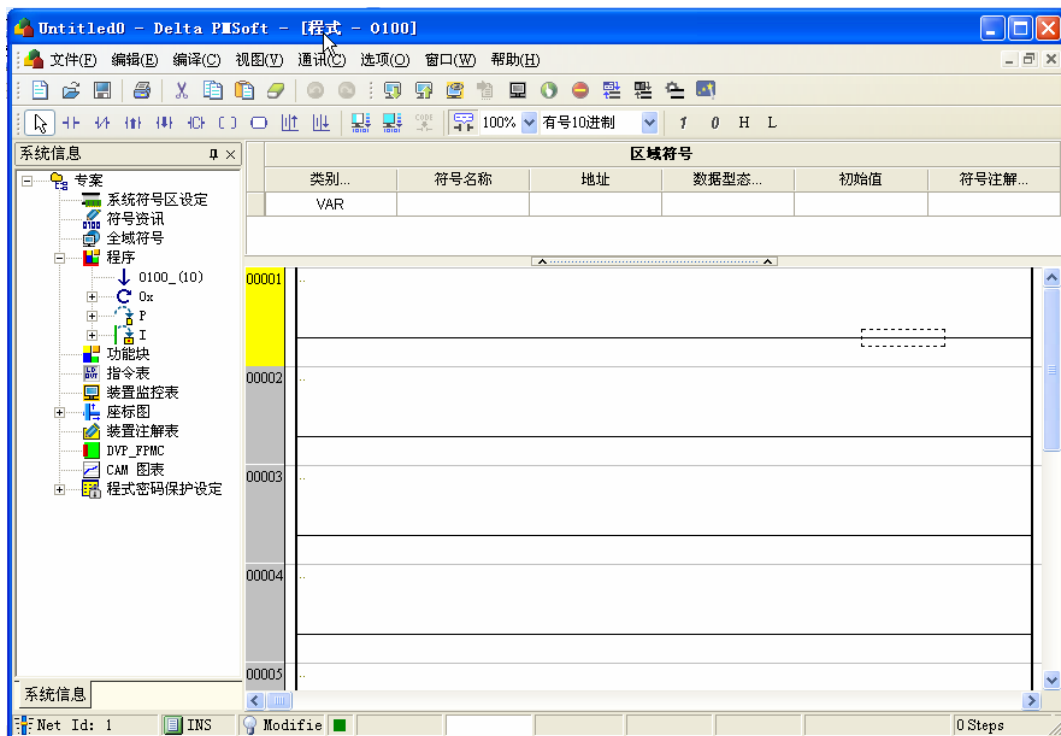
- 结果：如下图所示，方框为汇入之功能块。



### 8.1.7 系统装置配置表

PMSoft 系统装置配置表供使用者设定数据寄存器、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。



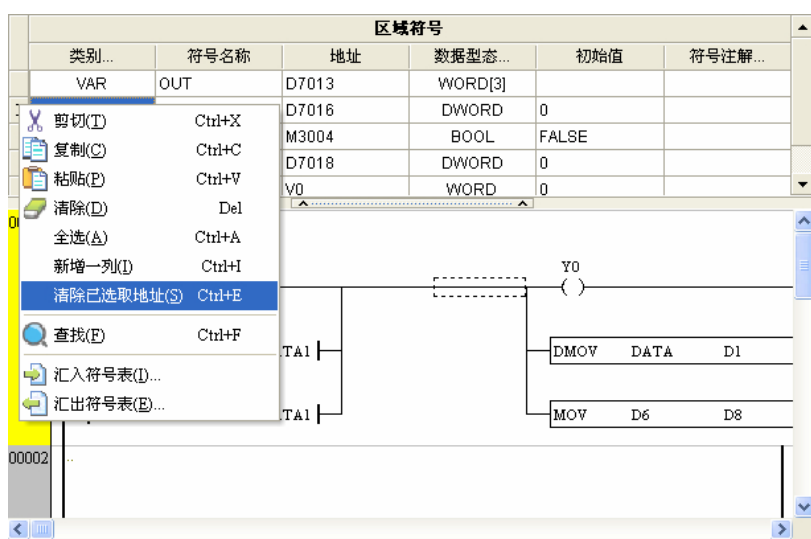
# 8 POU 梯形图编辑模式

## 装置范围设定

- D 数据寄存器：初始设定范围 D7000~D9999，可设定范围为 D0 ~ D9999，设定要避免特 D 数据寄存器。
  - T 计时寄存器：初始设定范围 T100~T192，可设定范围为 T0 ~ T255。
  - C 计数寄存器：初始设定范围 C100~C199，可设定范围为 C0 ~ C255。
  - M 辅助寄存器：初始设定范围 M3000~M4095，可设定范围为 M0 ~ M4095，设定要避免特 M 辅助寄存器。
  - P 指针寄存器：初始设定范围 P100~P254。
3. 点选『清除系统已配置符号地址』此时会将全域符号表和每个 POU 的区域符号窗体中地址字段的数值清除。



4. 若欲清除单一符号之地址，则在符号字段上按下鼠标右键，选取『清除已选取地址』选项，此时系统会将已选取范围中地址字段的数值清除。



清除选取地址后，VarIn 和 B1 两个符号的地址字段已被清除：

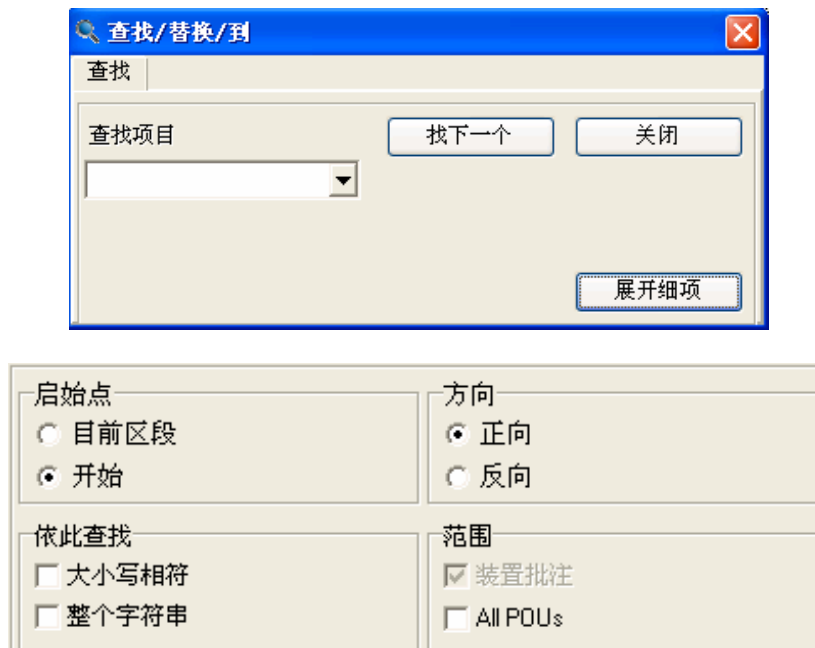
区域符号						
类别...	符号名称	地址	数据类型...	初始值	符号注解...	
VAR	OUT	D7013	WORD[3]			
VAR	VarIn		DWORD	0		
VAR	B1		BOOL	FALSE		
VAR	data	D7018	DWORD	0		
VAR	data1	V0	WORD	0		

### 8.1.8 梯形图查找

PMSoft 的查找功能分为梯形图查找以及 IL 查找，以下内容说明梯形图查找。梯形图查找可应用在 POU 程序查找与符号表查找，分别说明如下。

#### ■ POU 程序查找：

在 POU 梯形图中按下鼠标右键，并从功能选单中选取查找(F)命令。使用此命令查找程序中的指令、装置和符号，如果输入的指令、装置和符号存在，则会跳到查找的指令或是装置，而查找的方式可以按查找对话框下角的「更多」按钮，会有新增查找设定，如下图所示。

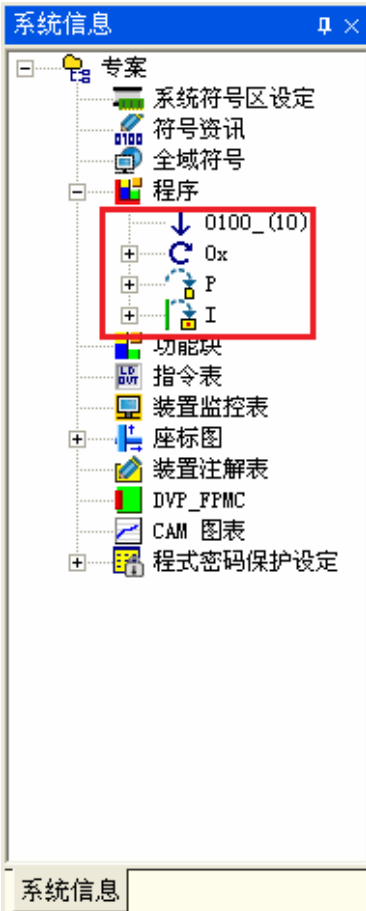


其中查找方式有下列几项：

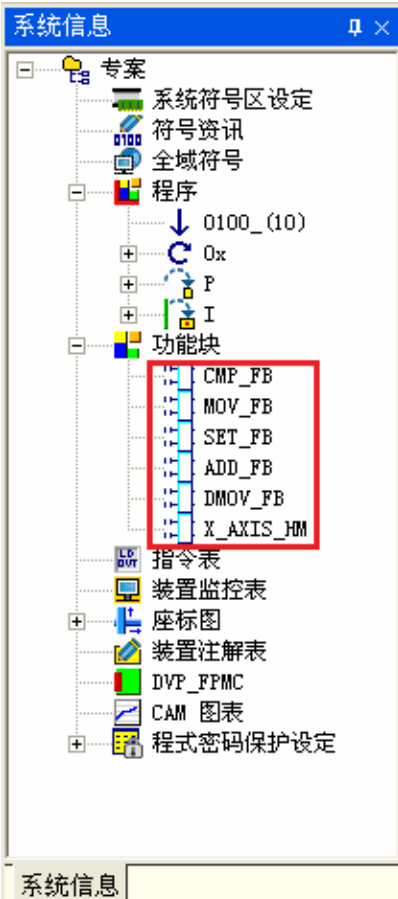
1. 目前区段：查找是要从目前编辑的网络区段开始查找。
2. 开始：查找是要从目前编辑窗口开始位置开始查找。
3. 大小写相符：查找的数据大小写要都一样。
4. 整个字符串：查找的数据是要整个字符串去比对
5. 正向：查找的方式是由上往下查找。
6. 反向：查找的方式是由下往上查找。
7. 装置批注：查找的位置只能在装置批注区查找。

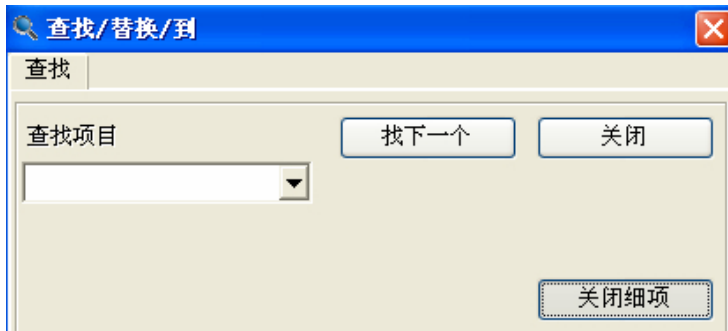
# 8 POU 梯形图编辑模式

8. 全部程序(POUs) : 查找范围为(O100, Ox0-Ox99, P0-P255)



9. 全部功能块(POUs) : 查找范围为使用者自订的功能块



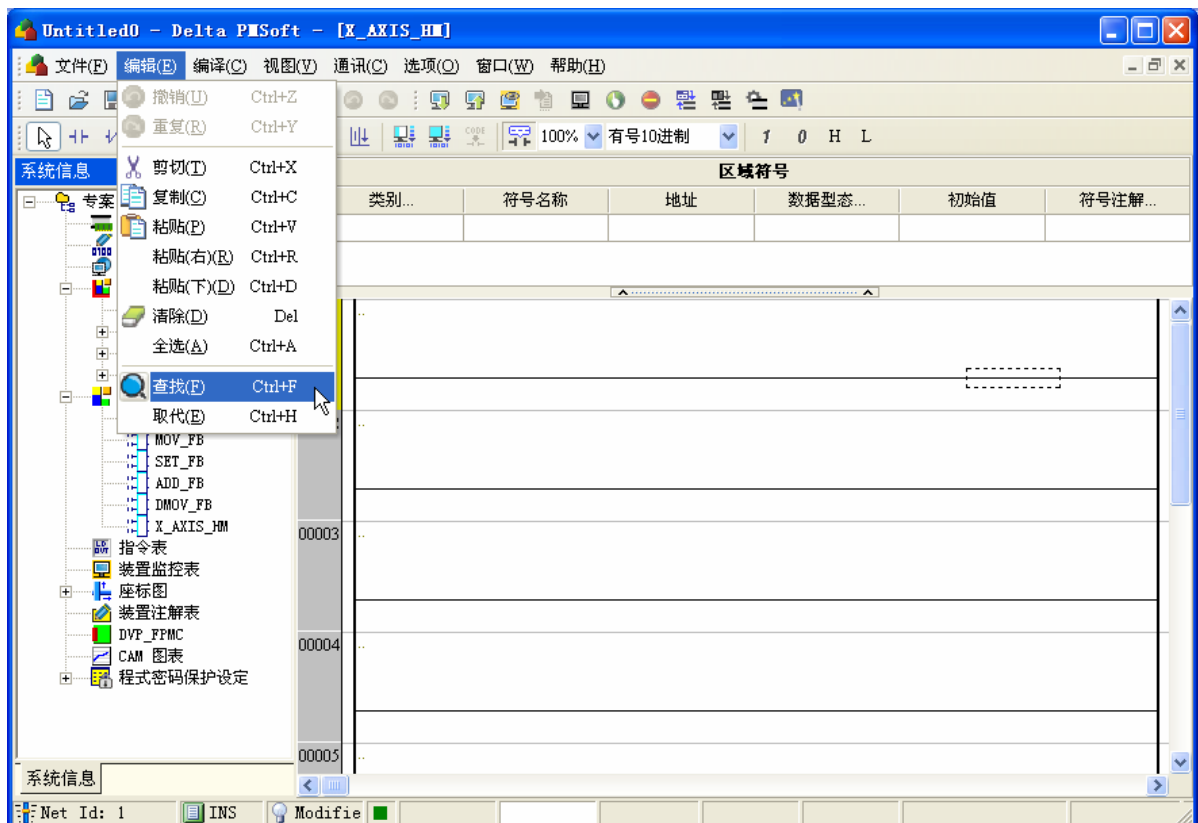


输入好查找文字，设定好查找方式，按『找下一个』按钮，就开始查找，被找到的文字会被反白，再按一次「找下一个」按钮，会跳到下一个被找到的文字，以此类推，假如再按一次「找下一个」按钮，已没有数据可查找，会显示「不存在」信息窗口。



■ 开启查找功能窗口的方法：

◆ 方法一：从功能列上选取编辑 (E) 中「查找(F)」命令。



◆ 方法二：鼠标右键菜单中「查找(F)」命令。

◆ 方法三：键入复合键 [Ctrl] + [F]。

## 8 POU 梯形图编辑模式

### 找下一个(N)

找下一个(N)的功能是以上一次查找的文字为查找数据，在梯形程序中往下查找，其查找方式是根据查找窗口里头的设定(下图所示)，找下一个(N)的功能就是查找方向的正向。

<b>起始点</b> <input type="radio"/> 目前区段 <input checked="" type="radio"/> 开始	<b>方向</b> <input checked="" type="radio"/> 正向 <input type="radio"/> 反向
<b>依此查找</b> <input type="checkbox"/> 大小写相符 <input type="checkbox"/> 整个字符串	<b>范围</b> <input checked="" type="checkbox"/> 装置批注 <input type="checkbox"/> All POU's

- ◆ 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)的功能是以上一次查找的文字为查找数据，在梯形程序中往上查找，其查找方式是根据查找窗口里头的设定，找前一个(P)的功能就是查找方向的反向。

- ◆ 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。

### ■ 符号表查找

符号表有全域符号表及区域符号表，查找只能对当时点选的符号表查找，而不能查找其它 POU 的符号表中的装置和符号。开启查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查找(F)命令





### 8.1.9 梯形图取代

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。使用取代要搭配查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，含有查找功能，有查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，是先做查找动作再做取代的动作，查找动作方式的设定同前面查找功能设定。使用取代要注意是否取代完导致产生梯形图中错误的指令、不合法的装置和错误的指令组合。下图为取代功能窗口(有开启查找详细设定窗口)，其功能包含查找功能。



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 ]。

## 8.2 POU 梯形图程序操作

POU 梯形图型式上与一般梯形图有一些差异，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。

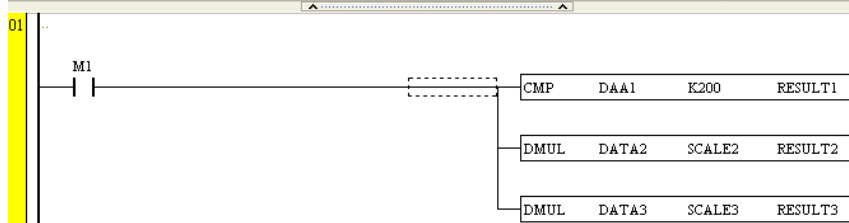
### 8.2.1 符号取代装置

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。

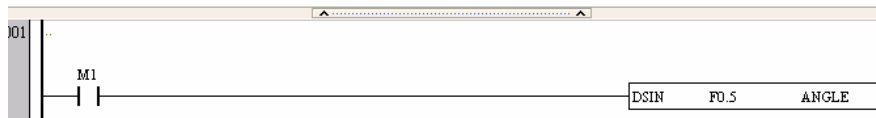
# 8 POU 梯形图编辑模式

类别...	符号名称	地址	数据类型...	初始值	符号注解...
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[2]		



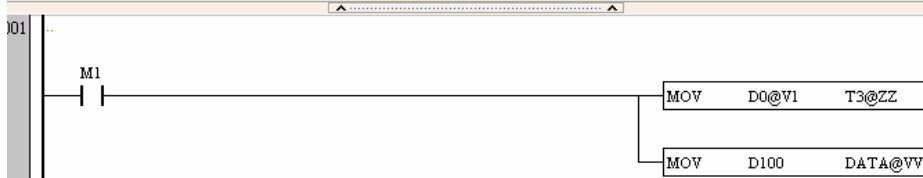
- 浮点数指令中 D 操作数之寄存器只能选用浮点数型态符号 **FLOAT**，如下图所示 DSIN 浮点数指令中 **ANGLE** 符号要宣告为 **FLOAT**。

类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	angle		FLOAT	0	



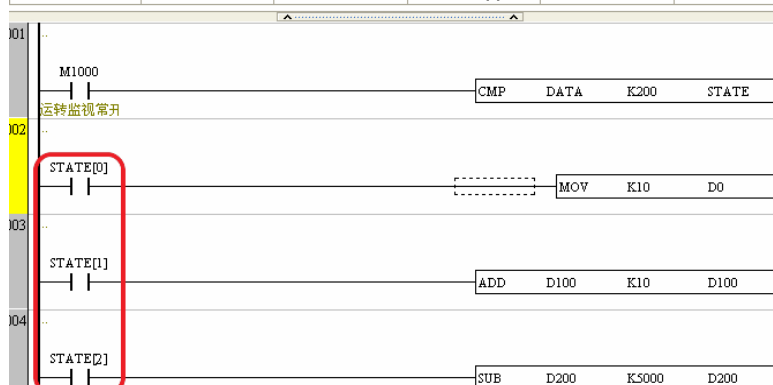
- 使用索引时，必须加上@字符，如下图所示。

类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	data1		WORD	0	
VAR	w	V0	WORD	0	
VAR	zz	Z5	DWORD	0	

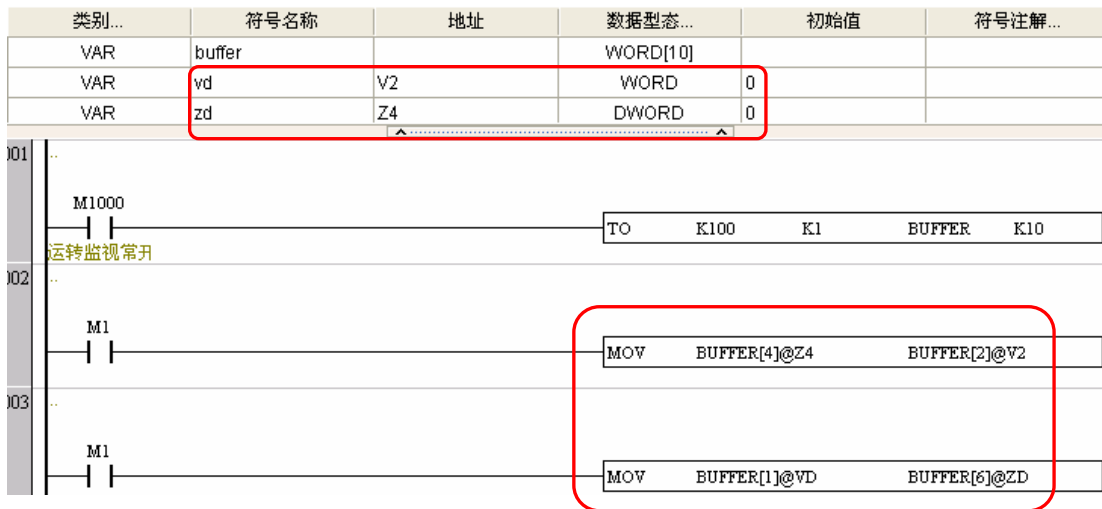


- 使用指针读取数组符号数据，如 **buffer** 为型态 **word[4]** 符号，可用 **buffer[2]** 读取第二个 **WORD** (指针 2) 的数据。如下图所示 **CMP** 指令最后一个操作数占用 3 位长度，所以 **STATE** 需宣告符号型态为 **BOOL[3]**，分别以 **STATE[0]**、**STATE[1]**、**STATE[2]** 表示指标 0、指标 1、指标 2 三种状态，以处理不同的运算。

类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	data		WORD	0	
VAR	state		BOOL[3]		



- 使用指针及索引读取数组符号数据，如 `buffer` 数据类型为 `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)。



\*注：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。

## 8.2.2 功能块 FB 使用

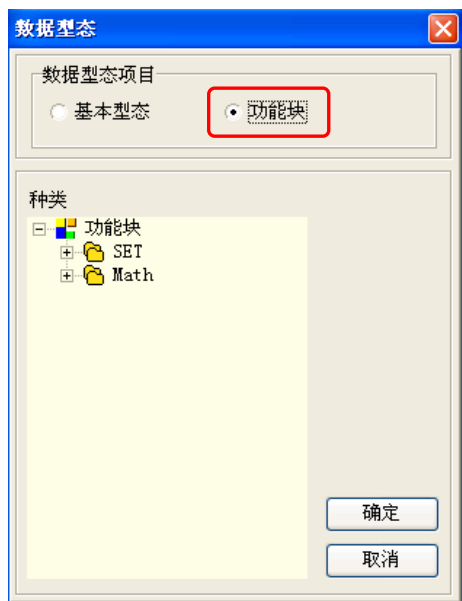
支持功能块的 POU 梯形图与一般梯形图在使用上有些差异，以下说明功能块 FB 的使用及使用功能块之 POU 梯形图的特点。

- 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。

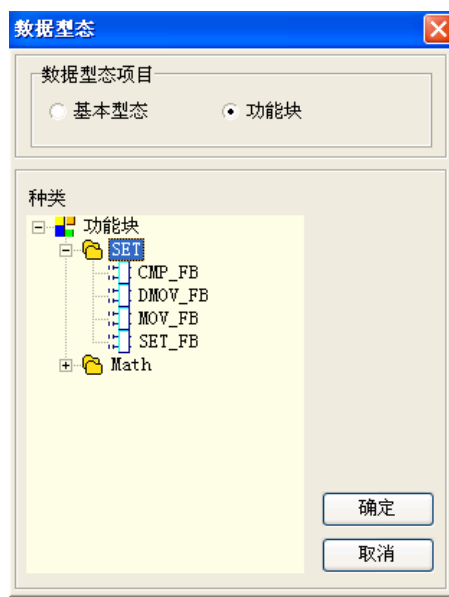
# 8 POU 梯形图编辑模式

## 步骤一：功能块宣告

Step 1：选择宣告符号为功能块型态



Step 2：选择目录后，点选欲宣告的功能块



Step 3：结果

区域符号					
类别...	符号名称	地址	数据类型...	初始值	符号注解...
VAR	Test1		WORD	0	
VAR	Start		BOOL	FALSE	
VAR	Data		WORD[4]		
VAR	ADD		ADD_FB		

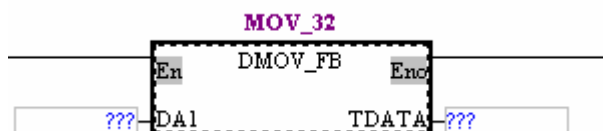
## 步骤二：功能块引用有三种方法：

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

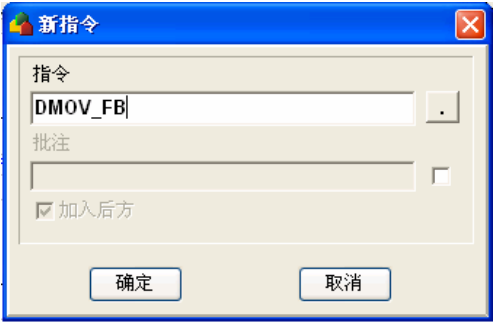


Step 2：结果

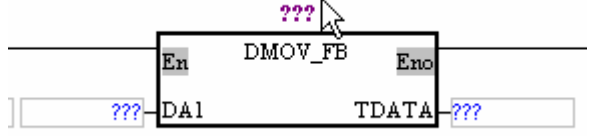


方法二：输入功能块名称(DMOV\_FB)再输入符号名称(MOV\_32)

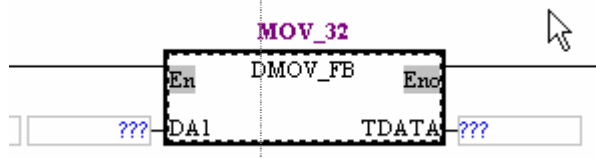
**Step 1：输入 DMOV\_FB**



**Step 2：点击???再输入符号名称 MOV\_32**

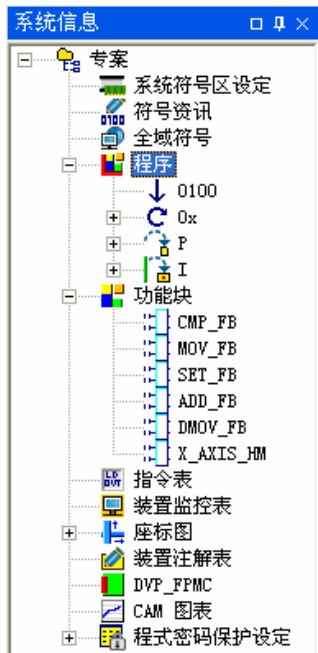


**Step 3：结果**

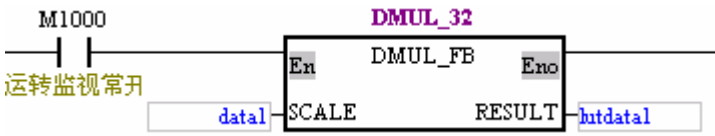


方法三：拖曳功能块，再来输入功能块名称。


**Step 1：选择功能块按鼠标左键**



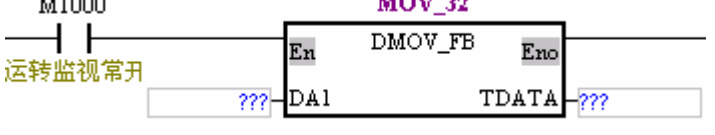
**Step 2：拖曳到指定位置**



**Step 3：点击???再输入符号名称 MOV\_32**



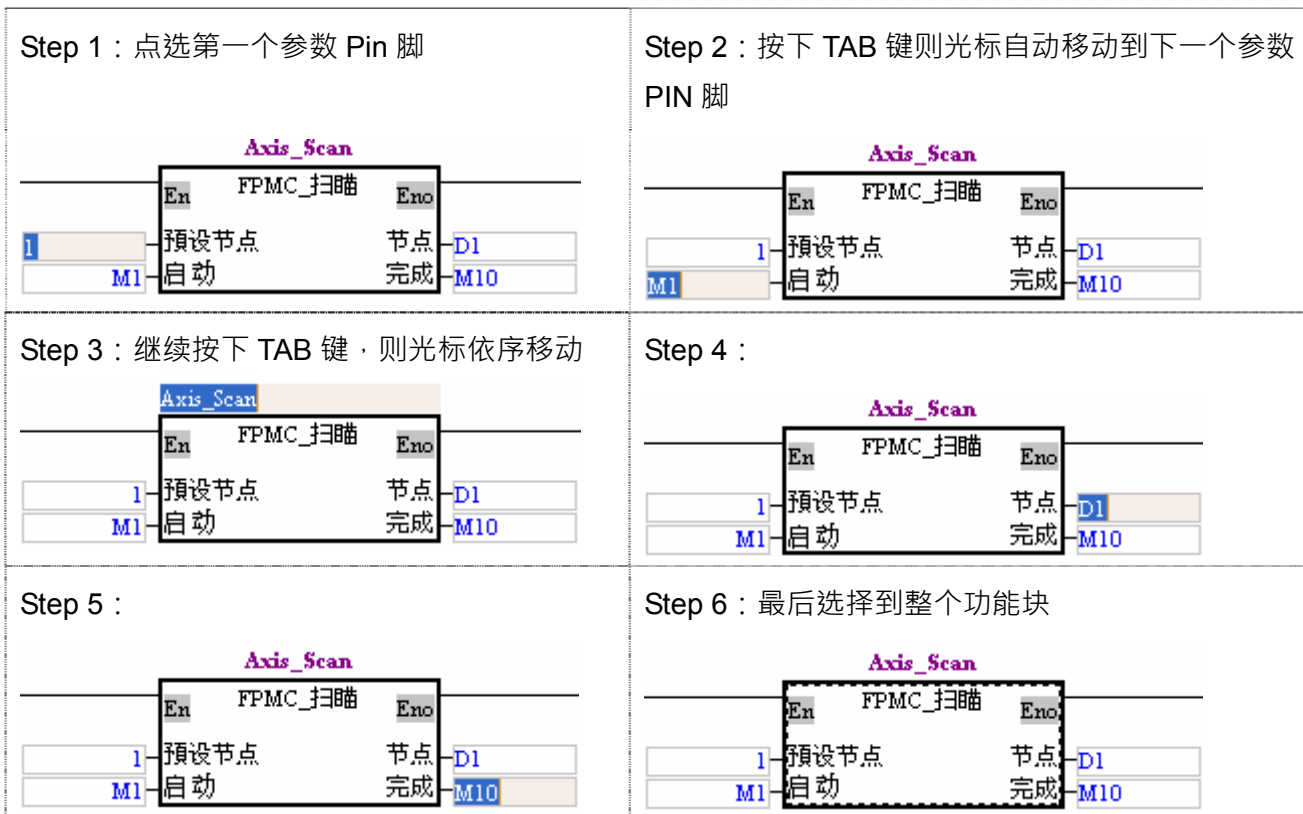
**Step 4：结果**



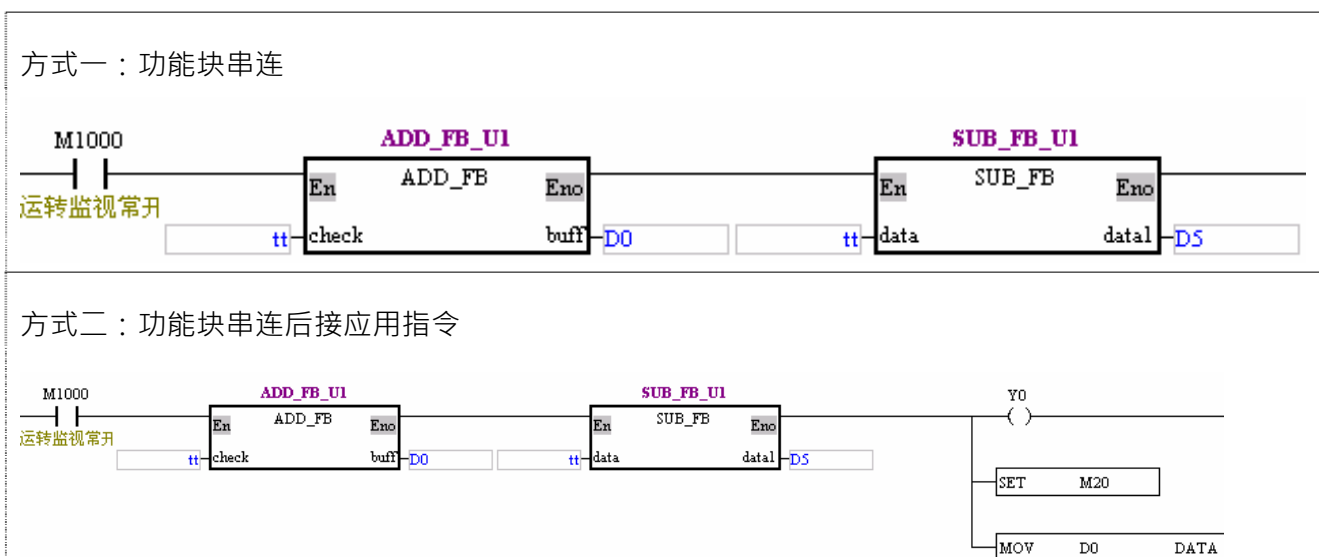
### 步骤三：编辑功能块的 Pin 脚

用鼠标直接点选功能块的编辑区，或是将游标点选到梯形图中的功能块，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。

# 8 POU 梯形图编辑模式



■ 梯形图差异：功能块在梯形图上是属于组件，但其在梯形图只有以下两种方式



- 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。

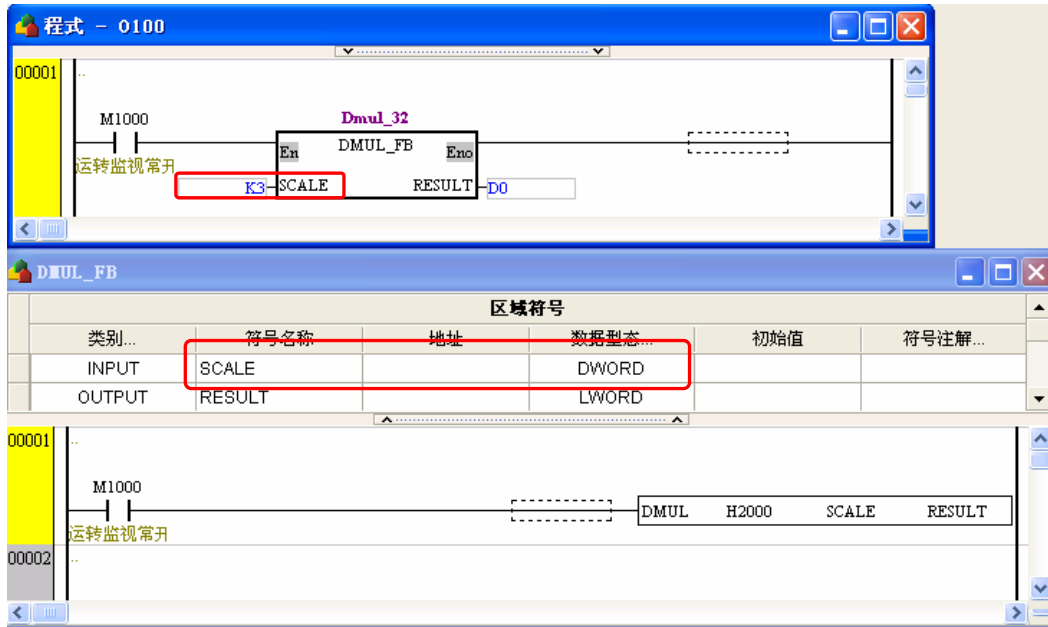
**Step 1 : 按编译**

**Step 2 : 结果**

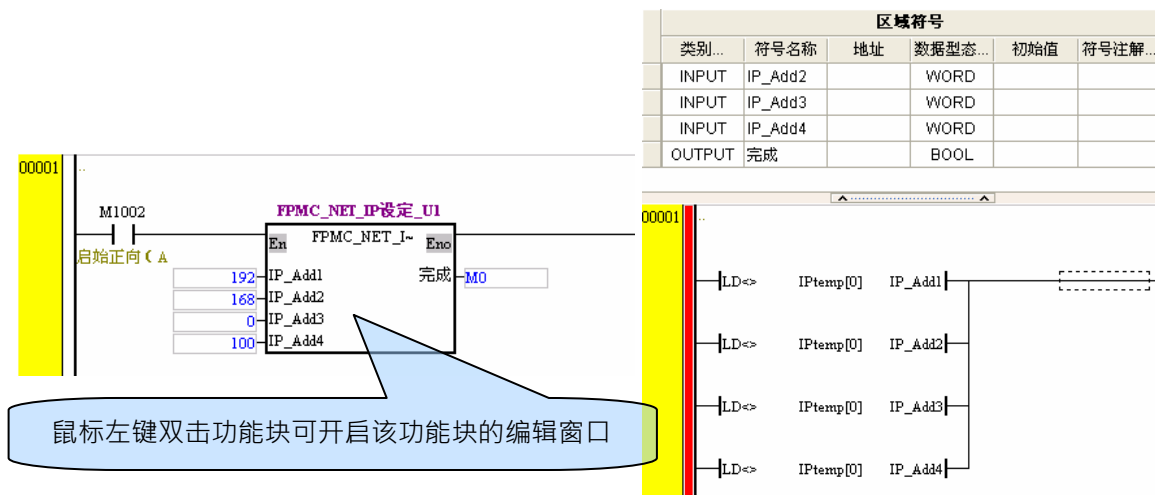
- 功能块可重复使用，使用同一个功能块型态，但使用不同符号，会转成不同 P 指针子程序，而占用 P 指针，并且在编译后转出不同之实体梯形图(instance)。宣告使用功能块及编写使用 P 子程序转成 IL 不可超过 256 个子程序。如下图所示，系统信息区中可看到使用 2 个子程序，而在 O100 主程序中及 OX1 运动程序中皆有使用功能块 DMUL\_FB，但符号名称分别在全域符号表及 OX1 区域符号表被宣告，因此占用 2 个 P 指标，并且在编译后转出不同之实体梯形图。

# 8 POU 梯形图编辑模式

- 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；但功能块的输入引脚使用 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：




- 在 Ladder 中用鼠标左键双击功能块节点，会直接开启该功能块编辑窗口

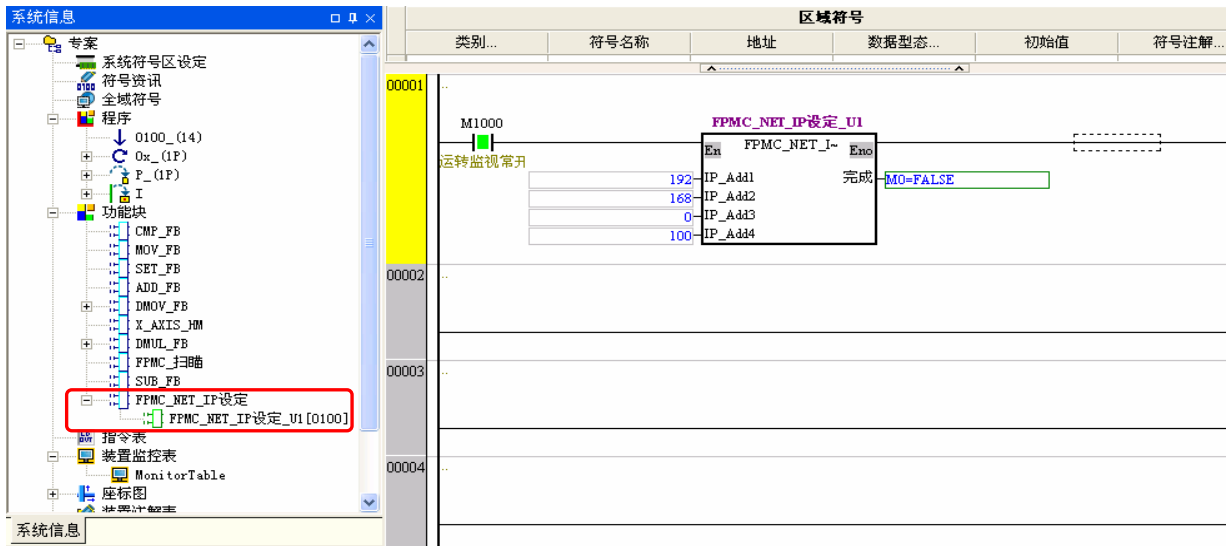




## 8.3 POU 在线监控

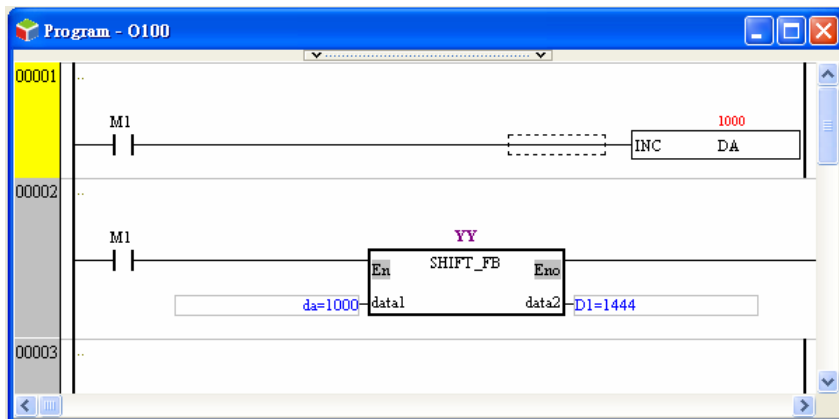
POU 程序运作可于梯形图监控状态中直接于程序或监控表中修改数据，以下将详细说明。

- 梯形图上监控：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梯形图如下图所示，圈选部分为监控 O100 运动程序及功能块实体程序。

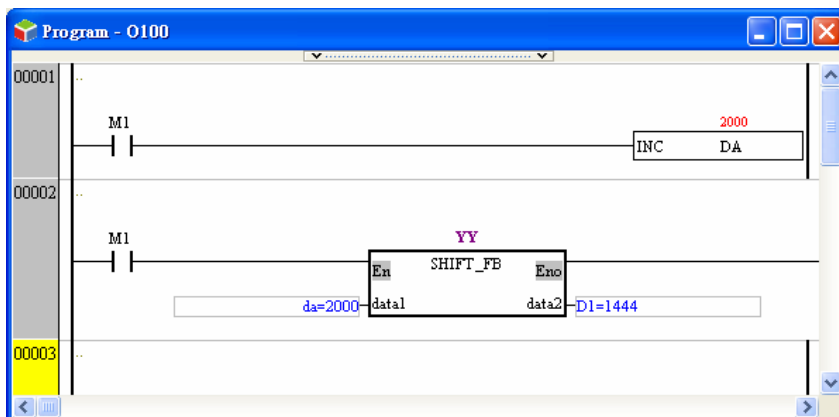


- 监控修改数据：如下图所示，O100 主程序中将变量 DA 值为 1000 改为 2000

Step 1：点选功能块中输入变量 DA，然后输入 2000

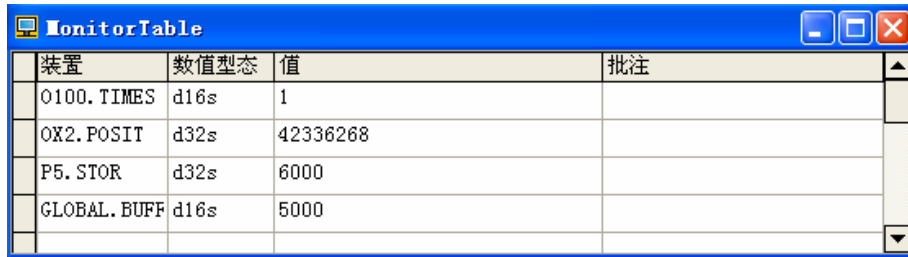


Step 2：变量 DA 显示 2000



## 8 POU 梯形图编辑模式

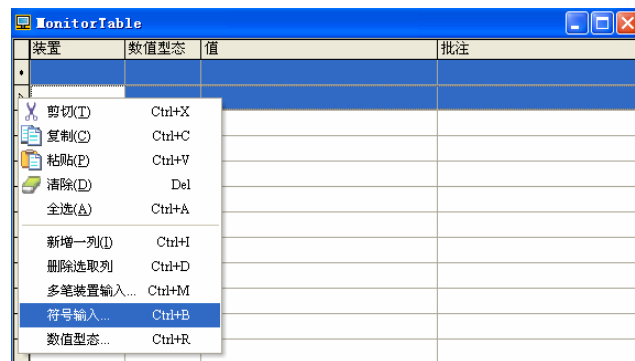
■ 监控表监控符号：在装置编号字段输入符号表名称(O100, OXn, Pn, Global) + "." + 符号名称。如下图所示在监控表 10 中要监控 O100 主程序中 TIMES 符号、OX2 运动程序中 X\_POSIT 符号、P5 子程序中 STOR 符号及全域符号表中 BUFFER。



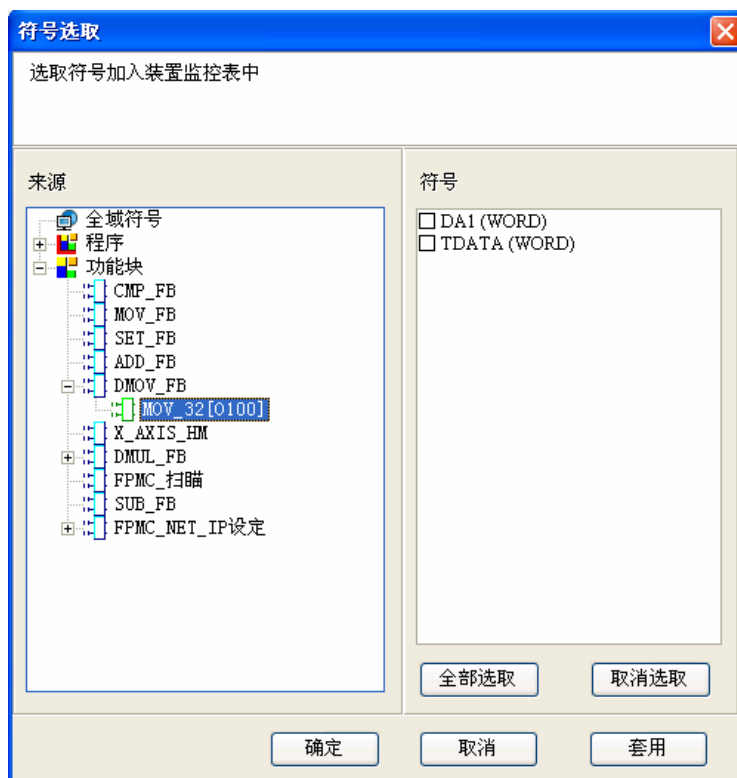
装置	数值型态	值	批注
O100.TIMES	d16s	1	
OX2.POSIT	d32s	42336268	
P5.STOR	d32s	6000	
GLOBAL.BUFF	d16s	5000	

■ 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。

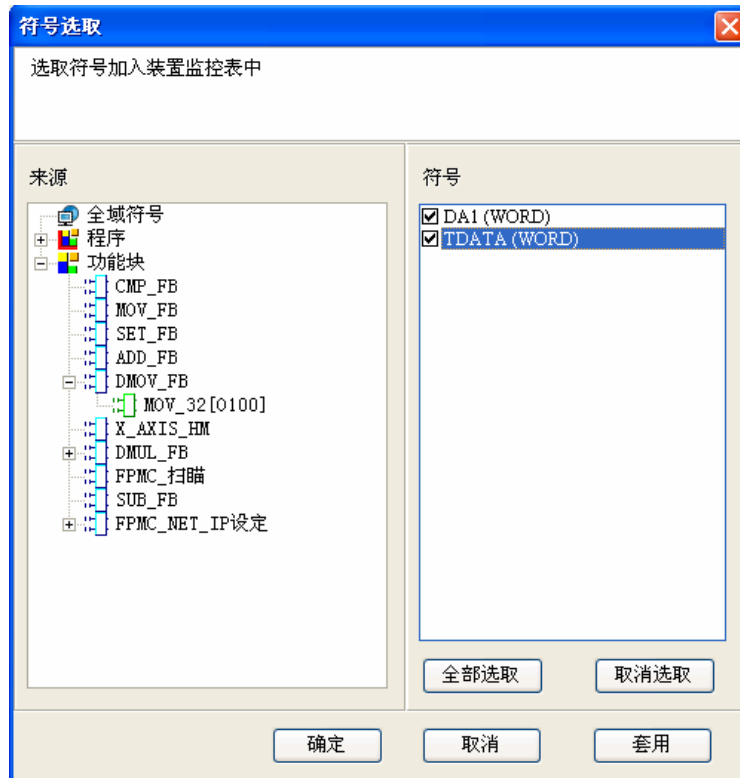
■ Step 1：在监控窗体上点鼠标右键选择「符号输入...」



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

装置	数值型态	值	批注
0100.MOV_32.	d16s		
0100.MOV_32.	d16s		

## 8.4 符号或功能块自动显示批注

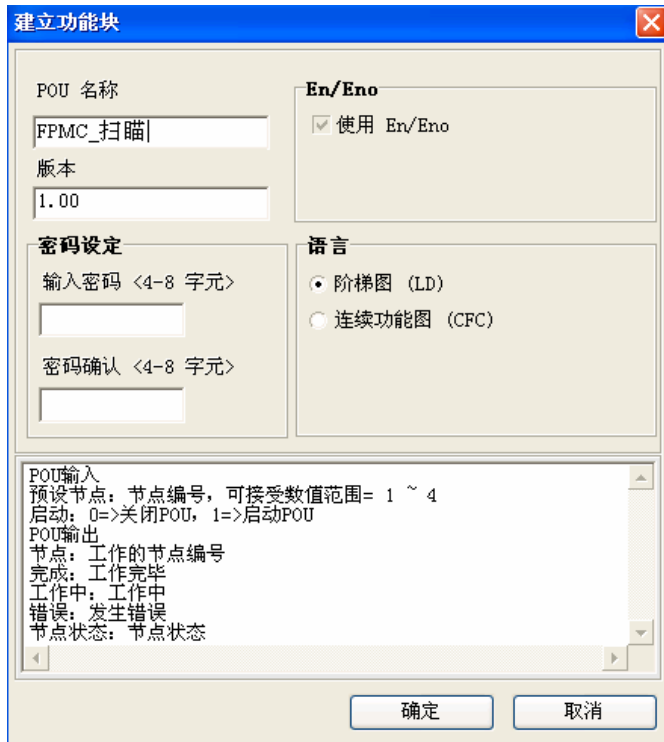
批注功能可方便使用者了解功能块的功能、功能块的引脚定义及符号型态。只要在符号表批注区(图(A))和功能块批注区(图(B))编写说明，程序中使用这些符号和功能块时，鼠标光标移到感应区域，将会自动显示符号、引脚或是功能块信息。



(A)符号表批注

区域符号					
类别...	符号名称	地址	数据型态...	初始值	符号注解...
INPUT	预设节点		WORD	0	节点编号，可接
INPUT	启动		BOOL	FALSE	0=>关闭PSU，
OUTPUT	节点		WORD	0	工作的节点编号
I	完成		BOOL	FALSE	工作完毕

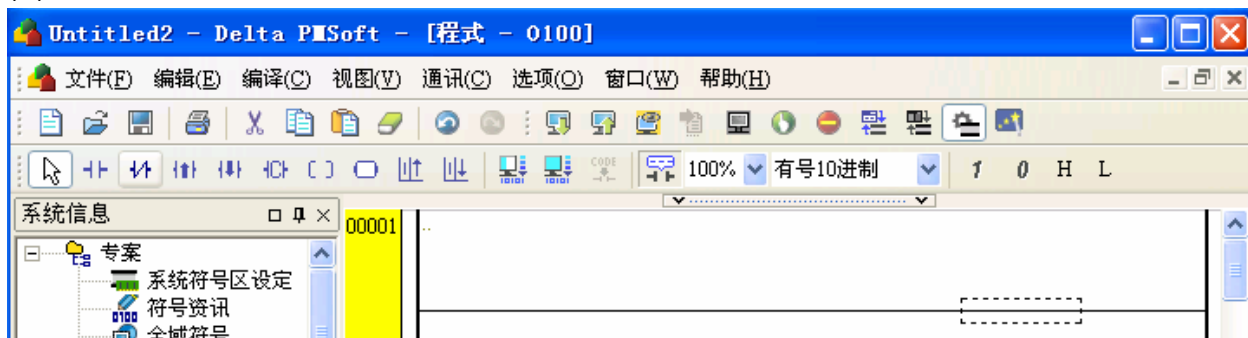
# 8 POU 梯形图编辑模式

## (B)功能块批注

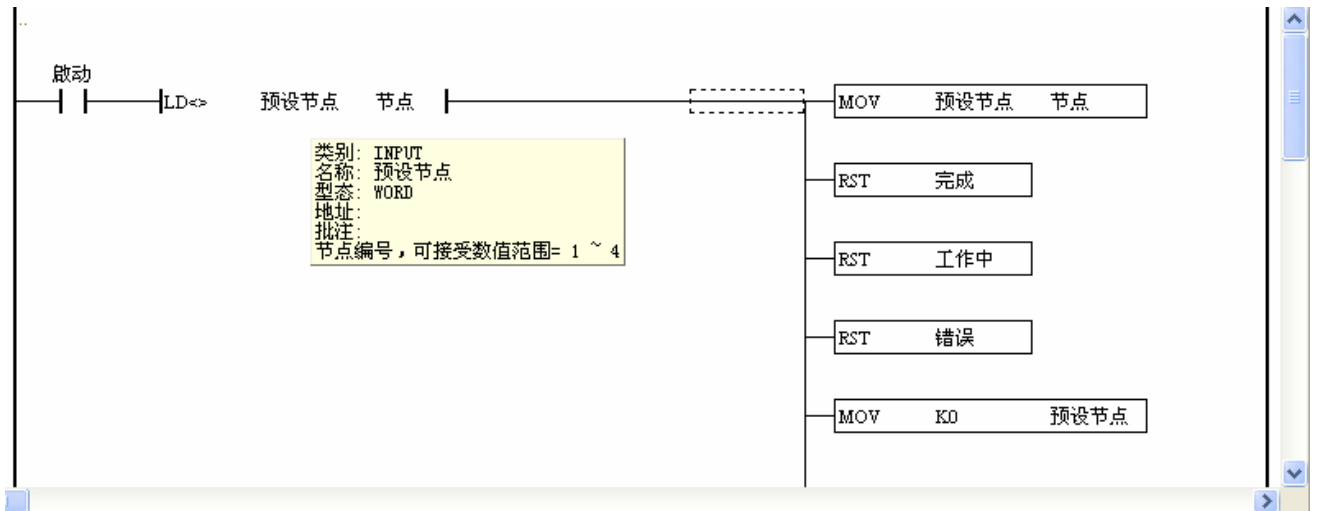


- 批注显示：开启批注显示功能可利用下图(A)所示按钮， 表示批注功能已开启， 表示批注功能已关闭，下图 (B)(C)(D)所示，分别为鼠标光标移至符号、功能块引脚和功能块名称时显示之批注。
- 符号批注显示项目含『类别』、『名称』、『型态』、『地址』、『批注』等信息。
- 功能块批注显示项目含『名称』、『批注』等信息。

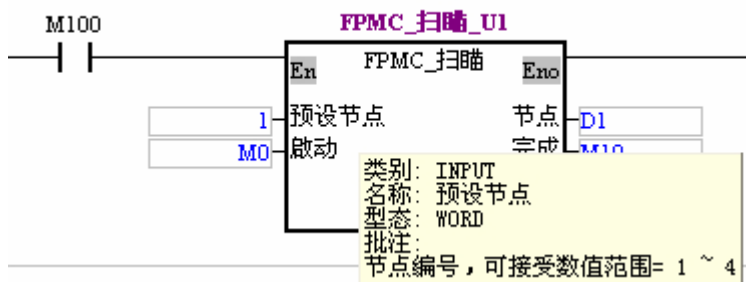
### (A) 批注显示致能按钮



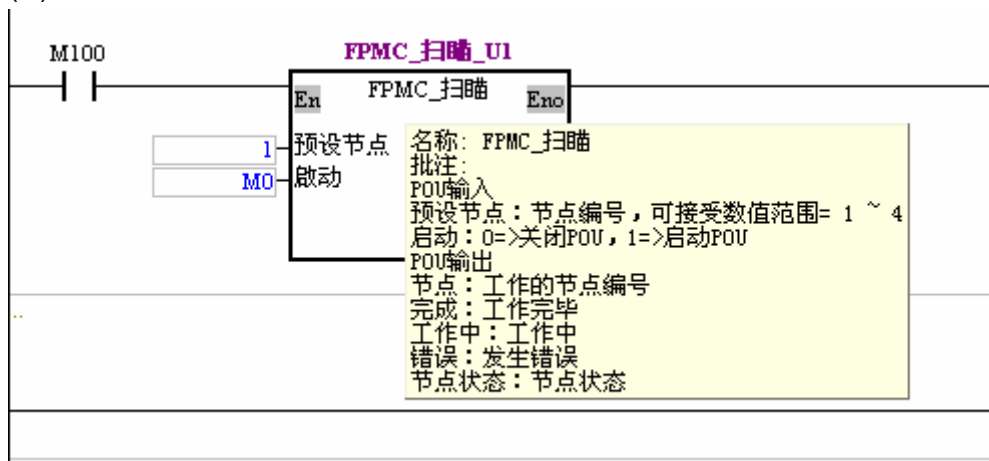
## (B) 符号批注显示



## (C) 功能块引脚信息



## (D) 功能块信息



MEMO

## 9.1 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

## 9.2 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、Heartbeat
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 )

# 9 CANopen 通讯功能卡

- 电气规格

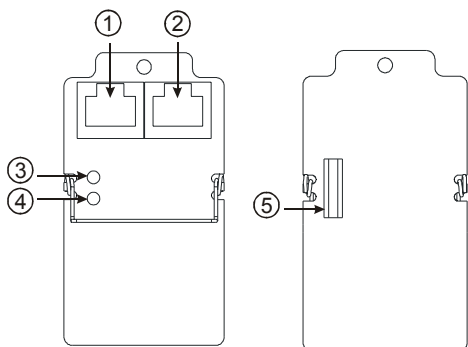
项目	规格
电源电压	由主机经由内部总线供应 24VDC (-15% ~ 20%)
消耗电力	1.7 W
绝缘电压	500 V
重量 (约,g)	66 (g)

- 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

## 9.3 产品外观及安装

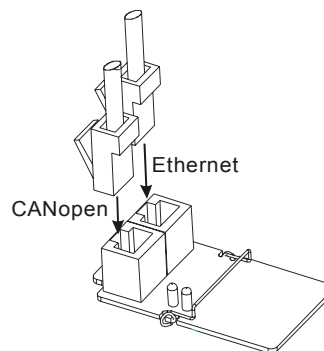
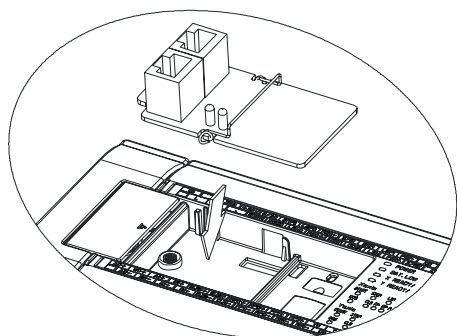
产品外观 :



- 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：



## 9.4 CR 参数功能介绍

- 共同参数区

CR# 编号	寄存器名称	保持型	数据类型	寄存器深度
#001	FPMC 本体版本	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 对象字典索引	R/W	Word	1
#073	SDO 对象字典传送/接收之数据寄存器 1	R/W	Word	512
#074	SDO 对象字典传送/接收之数据寄存器 2	R/W	Word	512
#075	SDO 对象字典传送/接收之数据寄存器 3	R/W	Word	512
#076	SDO 对象字典传送/接收之数据寄存器 4	R/W	Word	512
#080	NMT 动作命令	R/W	Word	1

# 9 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	数据类型	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 数据	R/W	Word	32
#091	FPMC Buffer 数据位置	R/W	Word	32
#092	FPMC Buffer 数据	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	软体版本	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	伺服之当前位置	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 对象字典索引	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	数据类型	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	插补模式目标位置	R/W	Word	1
#n91				
#n92	插补模式启动	R/W	Word	1

● CANopen 一般模式

CR# 编号	寄存器名称	保持型	数据类型	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

● 对象字典相关参数区

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 数据映射参数设定	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 数据映射参数设定	R/W	Dword	4
#2000h~#207Fh	PDO 数据寄存器	R/W	Word	4

# 9 CANopen 通讯功能卡

## 9.5 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 发送同步报文至 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	

# 9 CANopen 通讯功能卡

## 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。
- 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

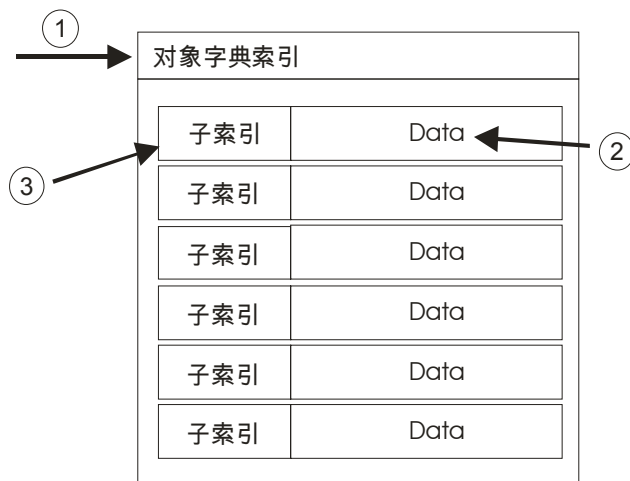
[说明]

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	欲传送对象的对象字典索引下的子索引	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	错误标志	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：

1. 指定欲传送对象(CR#70)之对象字典索引 CR#072。
2. 将欲发送之数据写入 CR#073~CR#076 数据寄存器。
3. 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 对象字典索引

[说明]

欲传送对象的对象字典索引，设定范围为 H'0000~H'FFFF。

## CR#073~CR#076 : SDO 对象字典传送/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；当扫描成功后会自动启动 HeartBeat 机制。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

## 9 CANopen 通讯功能卡

### 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：

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

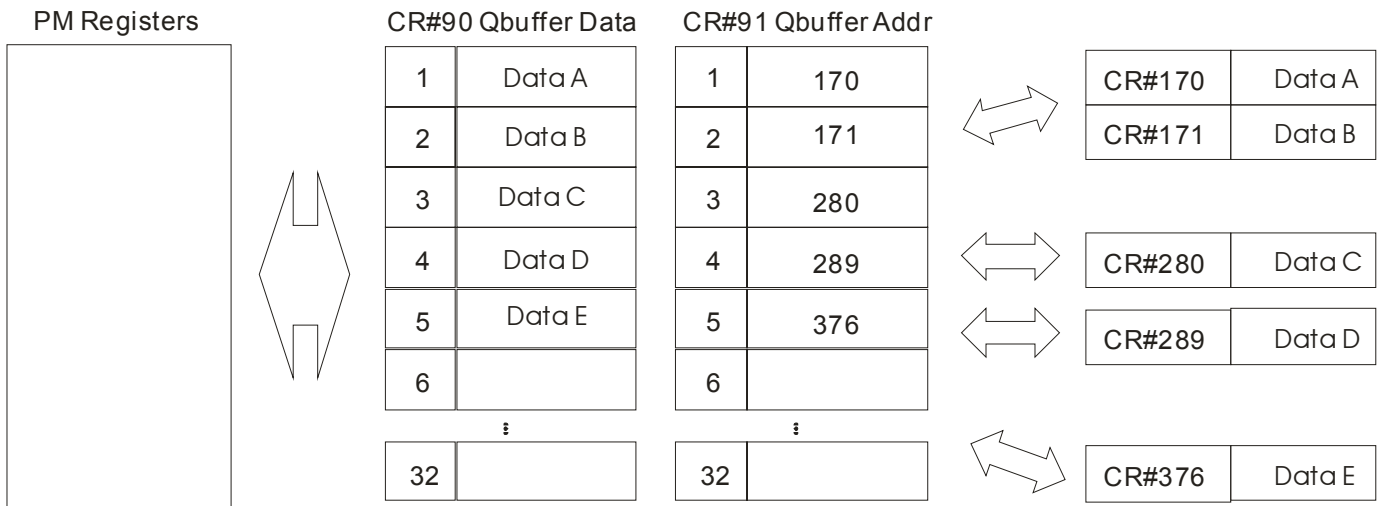
### CR#090~CR#93 : FPMC QBuffer 数据/数据位置

#### [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 的动作示意如下图所示：

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，数据类型为 Dword。

**CR#n03~CR#n04 : 产品代码**

[说明]

显示 ASDA-A2 伺服之产品代码，数据类型为 Dword。

**CR#n05~CR#n06 : 韧体版本**

[说明]

显示 ASDA-A2 伺服之韧体版本，数据类型为 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，数据类型为 Dword。

# 9 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，该标志会设定为 1，表示可以启动。
- SO：伺服已启动，当伺服启动后，该标志会设定为 1。
- FT：错误标志，当伺服发生错误时，该标志会设定为 1。
- QS：该标志为 1 时，可执行急停命令。
- WR：警告标志，当伺服出现警告时，该标志会设定为 1。
- RM：该标志为 1 时，可执行远程监控。
- TR：运动命令完成时，该标志会设定为 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	插补模式
OM[12]	已成功设定目标位置	回原点模式执行中	插补模式执行中
OM[13]	电机发生跟随错误	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	插补模式

**CR#n22~CR#n23 : 伺服之当前位置**

[说明]

显示伺服当前位置，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]

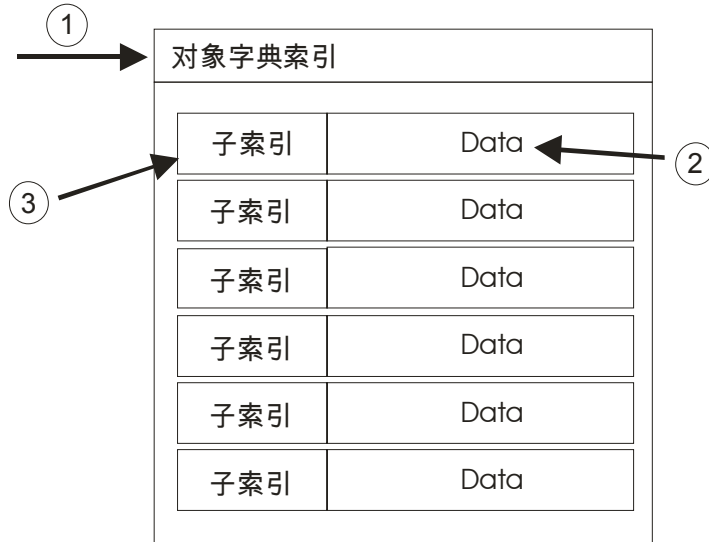
控制该节点之 SDO 的上/下载之动作，并显示结果，设定格式请参考下表：

Bit	Bit[15:8]	Bit[7:4]	Bit[3]	Bit[2:0]
设定值	欲传送对象的对象字典索引下的子索引	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	错误标志	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)

# 9 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. 指定该节点之对象字典索引 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 对象字典索引

[说明]

设定该节点的对象字典索引，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	插补模式
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	插补模式

## 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，数据类型为 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，数据类型为 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，数据类型为 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，数据类型为 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。
- 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。

## 9 CANopen 通讯功能卡

- 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，数据类型为 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，数据类型为 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，数据类型为 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，数据类型为 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：插补模式目标位置

#### [说明]

设定插补模式之目标位置，设定范围-2,147,483,648~2,147,483,647，数据类型为 Dword。

### CR#n92：插补模式启动

#### [说明]

内容值设定为 1 时启动回插补模式；内容值设定为 0 时关闭插补模式。

- 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 从站的 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。

		从站Heartbeat状态															
从站节点编号		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

## 9 CANopen 通讯功能卡

- 对象字典相关参数区

### CR#H'1006：同步周期设定

#### [说明]

设定一般模式下，CANopen 同步报文发送之时间间隔(单位：μs)，数据类型为 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。当设定值为 0 时表示不启动 FPMC 之 HeartBeat 机制。

### 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)

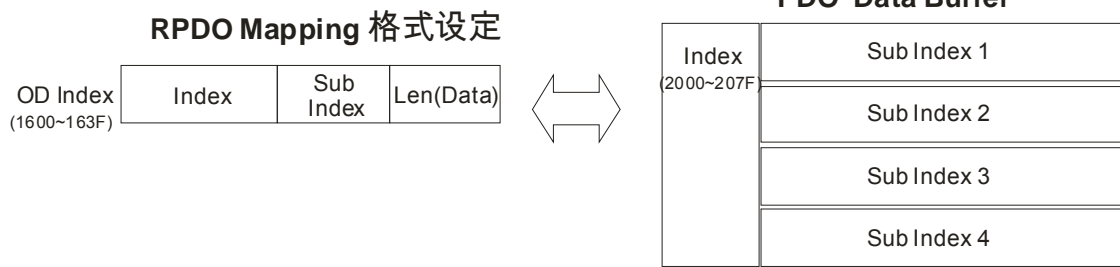
- PDO ID：CANopen 网络中报文代码，使用 2 个 Words 表示。初始值如下所示：  
CR#H'1400=H180+FPMC 本身 Node ID(CR#053)  
CR# H'1401=H280+FPMC 本身 Node ID(CR#053)  
CR#H'1403=H380+FPMC 本身 Node ID(CR#053)  
CR#H'1404=H480+FPMC 本身 Node ID(CR#053)
- 传输模式：依设定值决定传输模式，当设定值为 1~240，表示 PDO 会跟随 CANopen 网络同步报文发送，并且每隔设定值之同步周期均会发送一次。若设定值为 241~255 则不动作。初始值=241。

### CR#H'1600~CR#H'163F：接收模式 PDO 数据映射参数设定

#### [说明]

一般模式中，设定接收模式 PDO 数据存储器区映射之地址类型与参数。PDO 数据映射参数数据类型为 Dword，其中第一个 word 为对象字典之索引，第二个 word 中，high byte 为该对象字典索引之子索引，low byte 为数据类型，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)

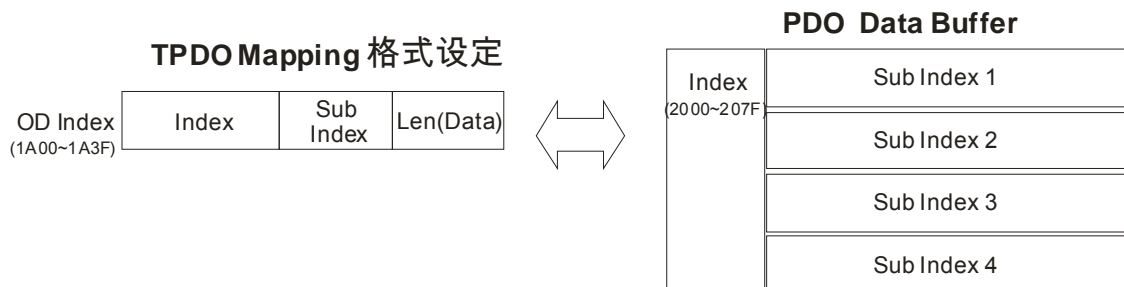
- PDO ID : CANopen 网络中报文代码，使用 2 个 Words 表示。初始值如下所示：  
 CR#H'1800=H200+FPMC 本身 Node ID(CR#053)  
 CR# H'1801=H300+FPMC 本身 Node ID(CR#053)  
 CR#H'1803=H400+FPMC 本身 Node ID(CR#053)  
 CR#H'1804=H500+FPMC 本身 Node ID(CR#053)
- 传输模式：依设定值决定传输模式，当设定值为 1~240，表示 PDO 会跟随 CANopen 网络同步报文发送，并且每隔设定值之同步周期均会发送一次。若设定值为 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 数据映射参数设定**

[说明]

一般模式中，设定传送模式 PDO 数据存储区映射之地址类型与参数。PDO 数据映射参数数据类型为 Dword，其中第一个 word 为对象字典之索引，第二个 word 中，high byte 为该对象字典索引之子索引，low byte 为数据类型，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。



# 9 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 卡开放对象字典中 2000H~207FH 为数据存储区，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。

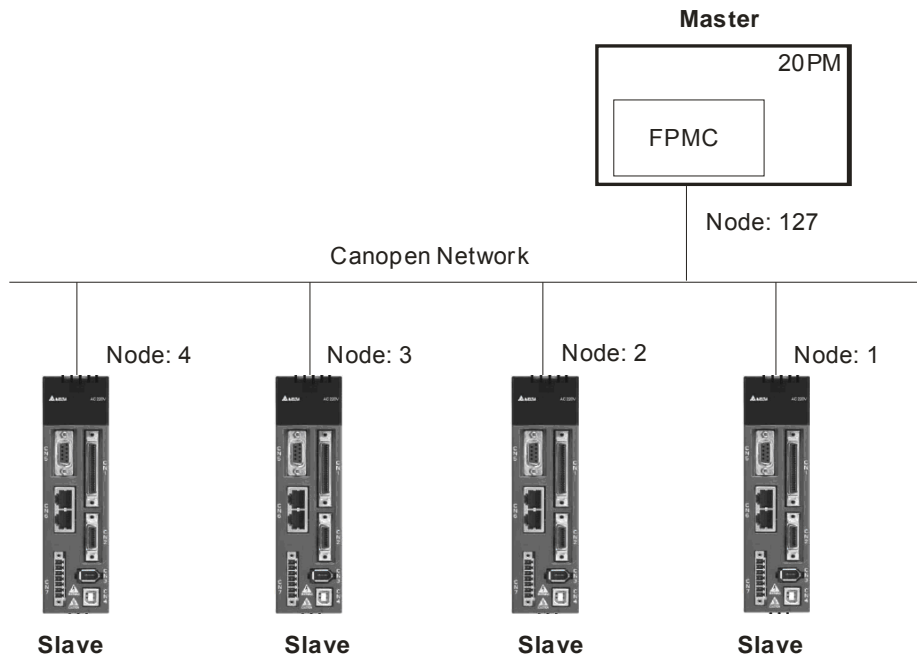
- FPMC 对象字典表

CR 位置 (16 进制)	数据类型	子索引个数	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

## 9.6 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	插补模式目标位置(CR#n90~CR#n91)	
5		伺服状态(CR#n20) 伺服运动状态(CR#n21)
6		伺服之当前位置(CR#n22~ CR#n23)

# 9 CANopen 通讯功能卡

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，查看内容值是否清除为 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。

- 一般模式

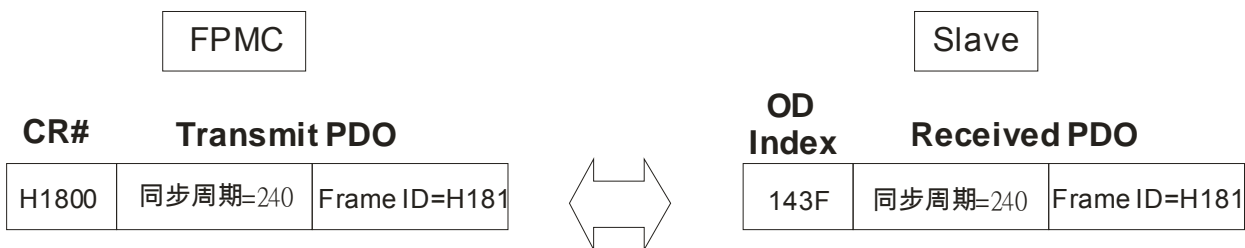
在一般模式下使用者需设定 FPMC 和 Slave 的 PDO 各项参数，透过 TO/FROM 指令对 FPMC 之功能寄存器设定，而在伺服中则必须利用 SDO 来对 PDO 的参数作写入的动作。FPMC 设定 PDO 的流程如下：

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。

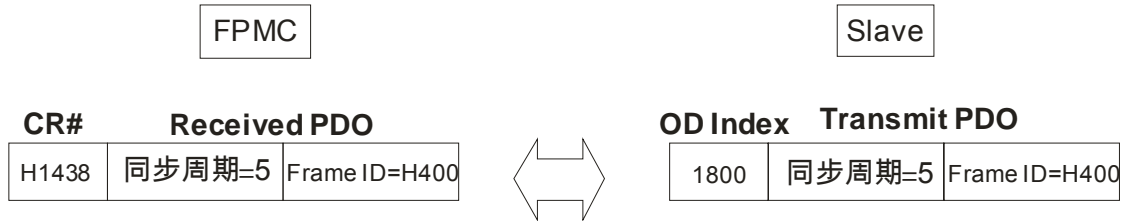
- TPDO 设定

FPMC 的 1800H~183FH 为 TPDO，对应的对象为 Slave 的 RPDO，举例说明，Master 使用对象字典索引为 1800 之 TPDO 与 Slave 之对象字典索引为 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：



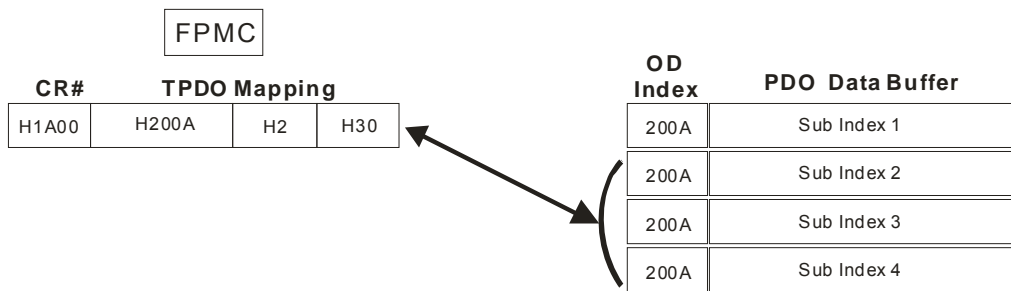
- RPDO 设定

FPMC 的 1400H~143FH 为 RPDO，对应的对象为 Slave 的 TPDO，举例说明，Master 使用对象字典索引为 1438 之 TPDO 与 Slave 之对象字典索引为 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。



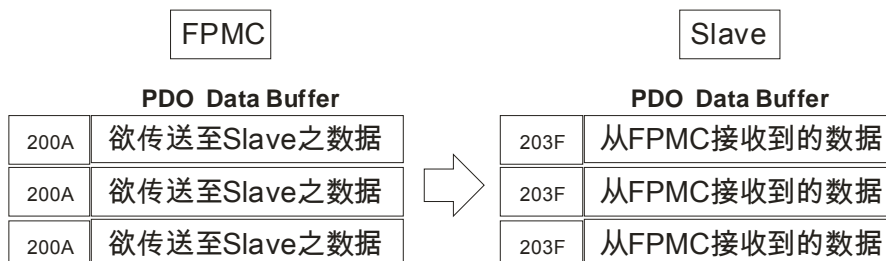
2. 设定 PDO Mapping 参数

设定映射到 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 映射参数为 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 Buffer 当中的 Sub Index 2~4。



3. 设定 PDO Data

在上述设定的对象字典中的位置，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。



# 9 CANopen 通讯功能卡

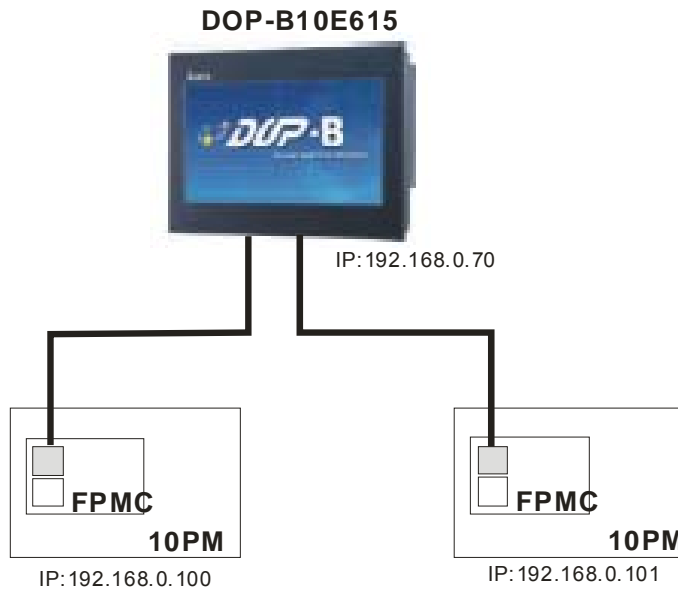
## 9.7 FPMC 之 Ethernet 模式介绍

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。

### 9.7.1 Ethernet 联机介绍 - FPMC 与 HMI 做数据传递范例

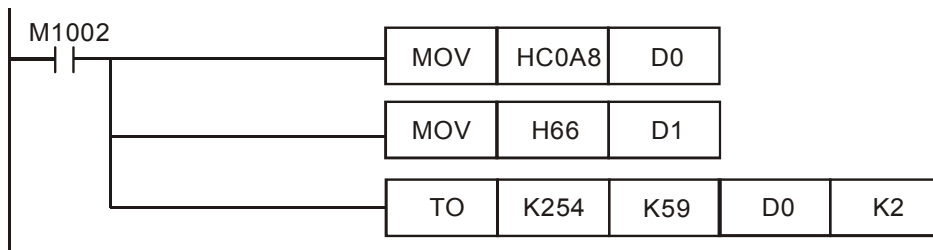
#### ■ 架构说明

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。

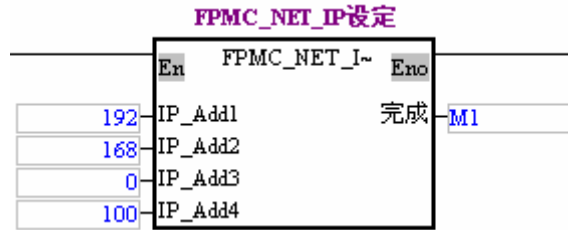


#### ■ FPMC 设定

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，格式请参考第 13.5 节之 CR#59 内容说明。以设定 192.168.0.100 为例，PM 程序如下所示：



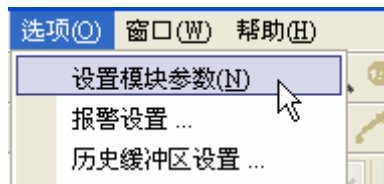
FPMC IP 地址设定之功能块如下图所示，输入引脚 IP\_Addr1~IP\_Addr4 分别表示 IP 地址的格式，以范例中 IP 地址 192.168.0.100 所示，设定完毕后输出引脚完成显示为 ON。



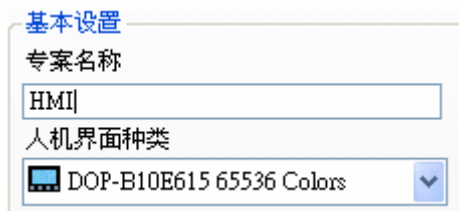
## ■ 人机端程序设定

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：

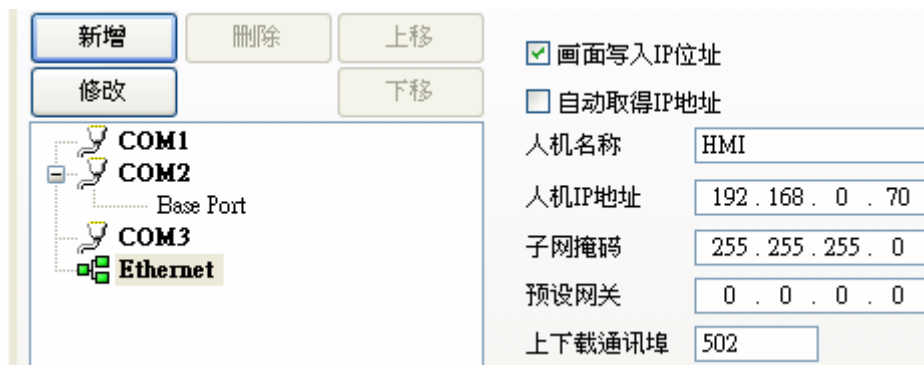
1. 在开启档案后在工具列中『选项』选择『设定模块参数』



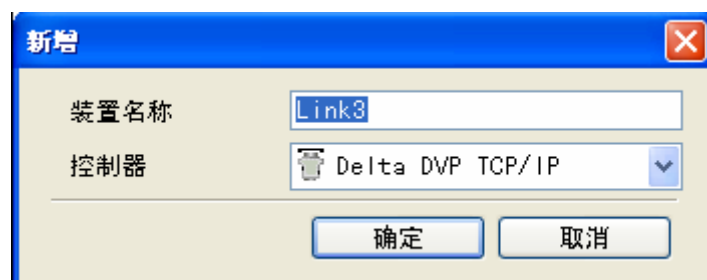
2. 在『一般』页面中，设定人机接口种类为 DOP-B10E615



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

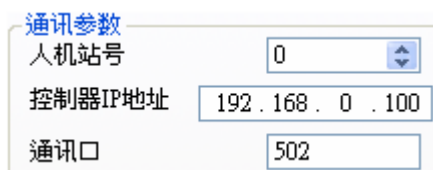


4. 新增通讯对象之联机，点选『新增』后跳出设定窗口，设定装置名称，控制器选择 Delta DVT TCP/IP，设定完毕后选择『确定』。

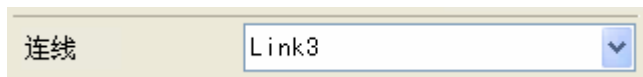


# 9 CANopen 通讯功能卡

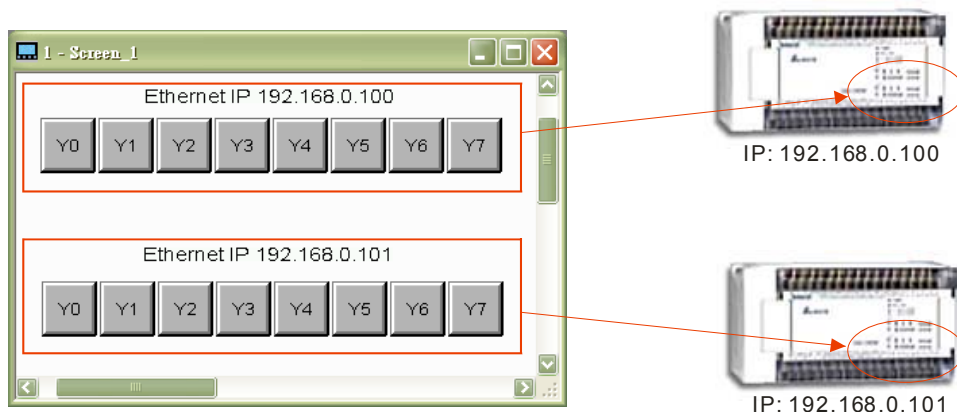
5. 设定通讯对象 IP ·点选新增的通讯对象 ·在通讯参数中 IP 位置 ·设定 FPMC 卡的预定的 IP 位置。



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。

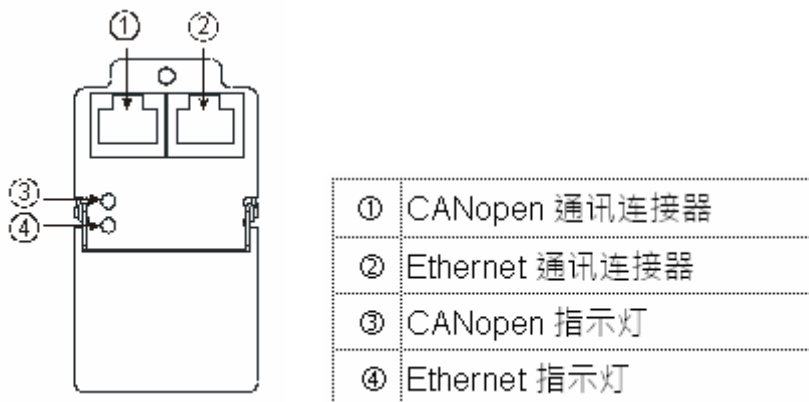


## 9.7.2 FPMC 与 PMSOft 联机介绍

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。



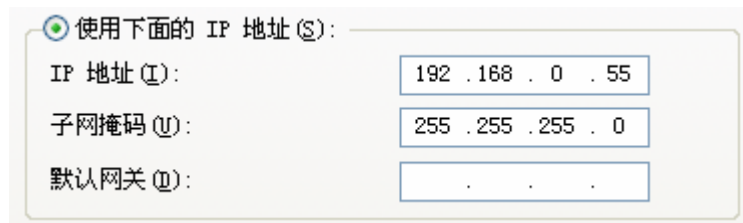


## ■ PC 端设定

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

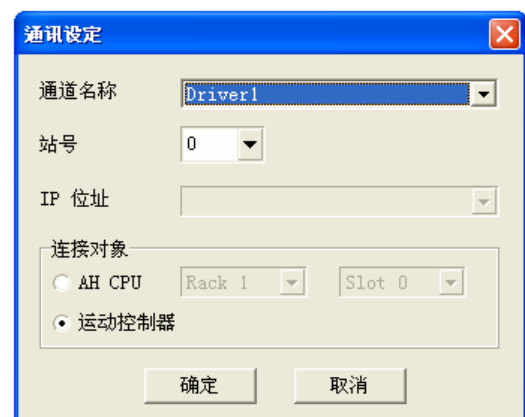
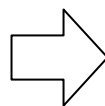


2. 在内容页面中选取，使用下列 IP 位置。IP 位置填入 192.168.0.55，最后一位可填除 100 外其它 1~255 的数值，子网掩码填入 255.255.255.0，完成后按确定



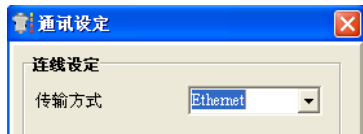
## ■ PMSOft 设定流程

1. 开启 PMSOft 选取通讯中的通讯设定，开启设定页面



# 9 CANopen 通讯功能卡

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






成功会跳出设定完成窗口



若失败时会跳出联机错误窗口



成功后，即可利用 Ethernet 做程序上下载以及监控动作。

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。

## 9.8 LED 指示灯说明及故障排除

### ■ 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

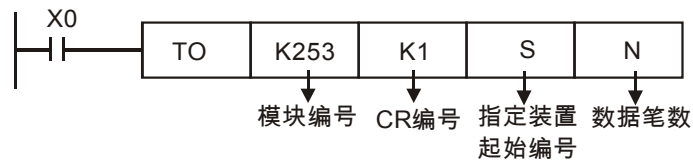
### ■ Ethernet 灯显示说明

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	

## 10.1 高速比较与捕捉功能

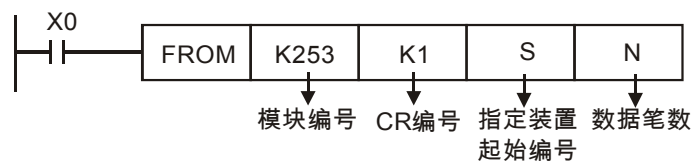
DVP-PM 配置特殊模块 K253 的寄存器 CR#1 设定 8 组高速比较与捕捉功能 透过 FROM/TO 指令设定/读取控制高速比较或是捕捉功能状态与数据，以下说明如何使用 FROM/TO 指令来设定/读取高速比较与捕捉功能的状态。

### ■ 设定控制



位置	设定控制	清除输出 及捕捉允许	设定捕捉 遮蔽值
S	起始组别 n(n=0~7)	0	0
S <sub>+1</sub>	0	1	2
S <sub>+2</sub> , S <sub>+3</sub>	组别 n 控制寄存器 CRn		
S <sub>+4</sub> , S <sub>+5</sub>	组别 n 数据寄存器 DRn		
S <sub>+6</sub> , S <sub>+7</sub>	组别 n+1 控制寄存器 CRn		
S <sub>+8</sub> , S <sub>+9</sub>	组别 n+1 数据寄存器 DRn		
:	:		
S <sub>+30</sub> , S <sub>+31</sub>	组别 n+7 控制寄存器 CRn		
S <sub>+32</sub> , S <sub>+33</sub>	组别 n+7 数据寄存器 DRn		
S <sub>+N*2</sub>	数据长=2+m*4, m=使用组数 · 最多 8 组		

### ■ 读取状态



位置	读取各计数器状态	读取捕捉输出状态
S	起始组别 n(n=0~7)	0
S <sub>+1</sub>	0	1
S <sub>+2</sub> , S <sub>+3</sub>	组别 n 控制寄存器 CRn	输出状态
S <sub>+4</sub> , S <sub>+5</sub>	组别 n 数据寄存器 DRn	捕捉允许 CapE (8bits)
S <sub>+6</sub> , S <sub>+7</sub>	组别 n+1 控制寄存器 CRn	
S <sub>+8</sub> , S <sub>+9</sub>	组别 n+1 数据寄存器 DRn	

# 10 高速比较与捕捉

位置	读取各计数器状态	读取捕捉输出状态
:	:	
S <sub>+30</sub> ,S <sub>+31</sub>	组别 n+7 控制寄存器 CRn	
S <sub>+32</sub> ,S <sub>+33</sub>	组别 n+7 数据寄存器 DRn	
S <sub>+N*2</sub>	数据长=2+m*4,m=使用组数 · 最多 8 组	

## ■ 设定控制/读取

(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
设定项目					比较结果输出				输出动作		比较条件		比较数据来源			

设定项目	Bit	设定值	10PM
比较数据来源	[3-0]	0	X 轴当前位置
		1	Y 轴当前位置
		2	Z 轴当前位置
		3	A 轴当前位置
		4	C200 计数值
		5	C204 计数值
		6	C208 计数值
比较条件	[5-4]	1	等于(=)
		2	大于等于(≥)
		3	小于等于(≤)
输出动作	[7-6]	0	触发(Set)
		1	重置(Rst)
		2、3	输出不动作
比较结果	[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		

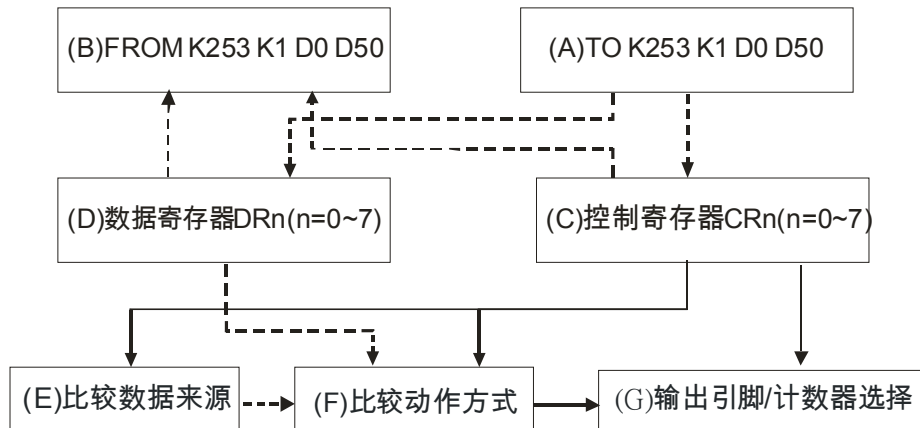
(2) 捕捉之控制寄存器 CRn 设定格式如下：

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
设定项目	触发来源设定				—						捕捉设定		捕捉数据来源			

设定项目	Bit	设定值	10PM
捕捉数据来源	[3-0]	0	X 轴当前位置
		1	Y 轴当前位置
		2	Z 轴当前位置
		3	A 轴当前位置
		4	C200 计数值
		5	C204 计数值
		6	C208 计数值
		7	C212 计数值
捕捉设定	[5-4]	0	捕捉设定为 0
外部触发来源设定	[15-12]	0	X0
		1	X1
		2	X2
		3	X3
		4	X4
		5	X5
		6	X6
		7	X7
		8	X10
		9	X11
		10	X12
		11	X13
		12	-
		13	-
14	-		
15	-		

## 10.2 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)及输出引脚设定(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)：设定比较数据来源。

方块图(F)：设定比较方式。

方块图(G)：设定比较成立时的触发模式。

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)引脚状态输出，或是 C200/C204 清除为零停止计数或是可再计数。

## ■ 范例说明

### 【功能说明】

以下范例使用 C204 高速计数，当计数值超过 100 时触发 Y1，超过 300 时重置 Y1；程序中使用两组比较器，一组用来触发 Y1，另外一组用于重置 Y1。在 Y1 触发时，PM 主机外壳上的 Y1 信号灯并不会显示，但可透过外部线路得知 Y1 是否为 ON，故在外部接线中将 C1 端子接 24G，Y1 接 X7，S/S2 接+24V，以及外接一手摇轮。

### 【操作步骤】

- 启动 O100 时先执行高速比较初始设定
  - D0=0 → 设定第一组比较起始组别 n=0
  - D1=0
  - D20=10 → TO 指令写入 10 笔数据，有两组高速比较数据
  - D60=10 → FROM 指令读取 10 笔数据，有两组高速比较数据
- 设定 M1，设定两组高速比较数据
  - 第一组 控制数据 D3,D2=H125 → 比较来源 C204(Bit3-0=5)，大于等于设定(Bit5-4=2)，触发设定 (Bit7-6=0)，触发引脚设定 Y1 (Bit11-8=1)
  - 第一组 暂存数据 D5,D4=K100 设定比较值大于等于 100 将触发
  - 第二组 控制数据 D7,D6=H165 → 比较来源 C204(Bit3-0=5)，大于等于设定(Bit5-4=2)，触发清除 (Bit7-6=1)，触发引脚设定 Y1 (Bit11-8=1)
  - 第二组 暂存数据 D9,D8=K300 设定比较值大于等于 300 将触发
- 设定 M2 启动两组高速比较功能
- 设定 M3 读取两组高速比较功能设定状态，如下图所示。



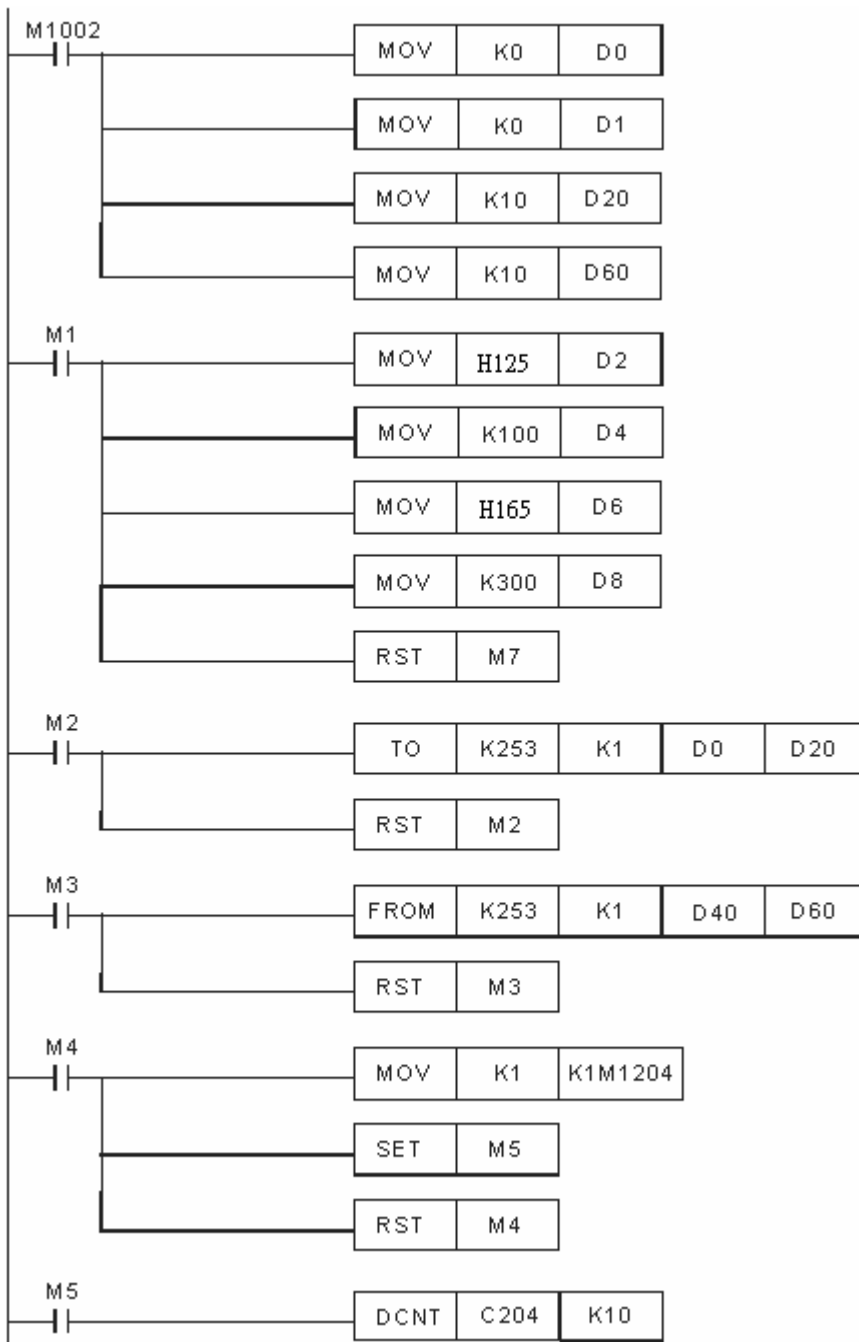
装置	数值型态	值	批注
C204	d32u	0	Y轴手轮计数值
D44	d32u	100	读取到n=0数据寄存器D5..D4
D48	d32u	300	读取到n=1数据寄存器D9..D8
D40	d16u	0	读取D0数据
D41	d16u	0	读取D1数据
D42	h32	00000125	读取到n=0数据寄存器D3..D2
D44	d32u	100	读取到n=0数据寄存器D5..D4
D46	h32	00000165	读取到n=1数据寄存器D7..D6
D48	d32s	300	读取到n=1数据寄存器D9..D8

- 设定 M4 设定 C204 为 PD(脉冲+方向)，然后设定 M5，启动计数。
- 转动手摇轮，看 C204 是否有计数，如下图所示。

装置	数值型态	值	批注
C204	d32u	95	Y轴手轮计数值
D44	d32u	100	读取到n=0数据寄存器D5..D4
D48	d32u	300	读取到n=1数据寄存器D9..D8

7. 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 Y1。
8. 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 Y1。

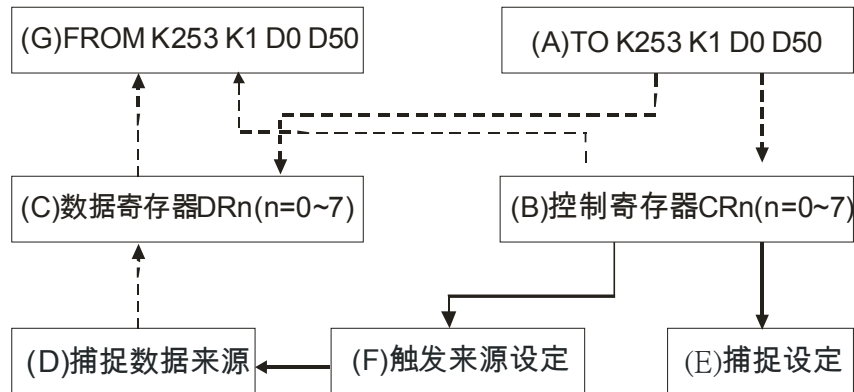
### 【PM 程序】





### 10.3 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信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



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)及触发来源引脚设定(方块图 F)。

方块图(C)：储存 TO 指令写入的数据，或是储存外部硬件信号触发而将方块图(D)所设定的数据来源装置数据。

方块图(D)：设定捕捉的来源。

方块图(E)：设定捕捉功能。

方块图(F)：触发捕捉信号来源。

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)→当外部触发信号产生(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。

## ■ 范例说明

### 【功能说明】

使用 Y 轴手摇轮启动 C204(Y 轴手轮)高速计数功能，当外部信号 X5 触发时捕捉 C204 计数值(手轮高速转动时)。

### 【操作步骤】

1. 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，只有一组捕捉数据
  - (4) D60=6→ FROM 指令读取 6 笔数据，只有一组捕捉数据
2. Set M1 设定捕捉
  - (1) 控制数据 D3..D2=H5005 → 捕捉来源 C204(Bit3-0=5)，捕捉模式设定(Bit5-4=0)，触发捕捉信号 X5(Bit15-12=5)
  - (2) 暂存数据 D5..D4=K100 可任意值忽略
3. Set M2 使用 TO 指令写入，捕捉功能启动
4. Set M3 使用 FROM 指令读取第一组捕捉设定状态，如下为读取状态值。

装置	数值型态	值	批注
C204	d32u	0	Y轴手轮计数值
D44	d32u	100	对应D5..D4数据寄存器
D40	d16u	0	读取D0数据
D41	d16u	0	读取D1数据
D42	h32	00005005	对应D3..D2控制寄存器
D44	d32u	100	对应D5..D4数据寄存器

5. Set M4 设定 C204 为 PD(脉冲+方向)。设定 M5。启动计数
6. 转动手轮，监看 C204 是否有计数，如下图所示

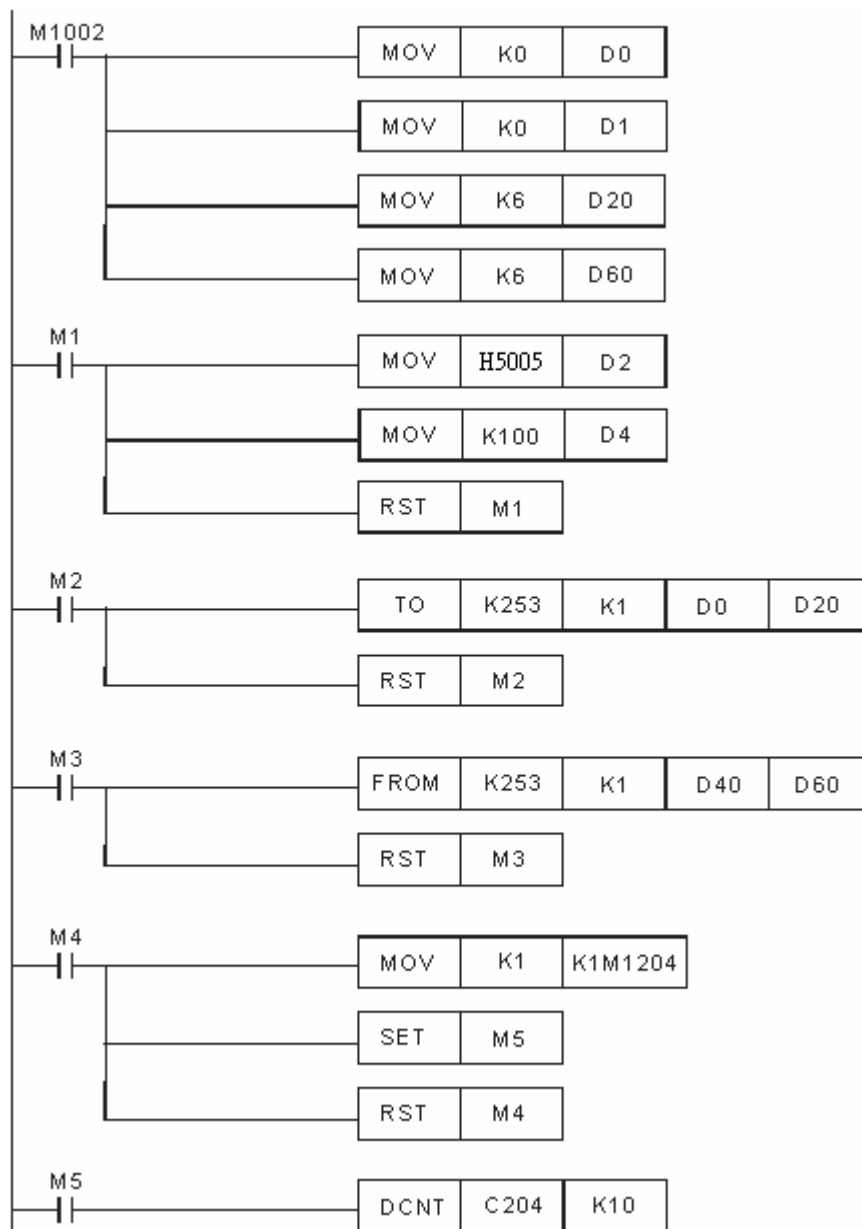
装置	数值型态	值	批注
C204	d32u	244	Y轴手轮计数值
D44	d32u	100	对应D5..D4数据寄存器
D40	d16u	0	读取D0数据
D41	d16u	0	读取D1数据
D42	h32	00005005	对应D3..D2控制寄存器
D44	d32u	100	对应D5..D4数据寄存器

7. 持续转动手摇轮，并使用外部信号触发 X5。
8. Set M3 读取第一组捕捉设定状态，D44 为触发 X5 时捕捉到 C204 的计数值。

MonitorTable0			
装置	数值型态	值	批注
C204	d32u	726	Y轴手轮计数值
D44	d32u	677	对应D5..D4数据寄存器
D40	d16u	0	读取D0数据
D41	d16u	0	读取D1数据
D42	h32	00005005	对应D3..D2控制寄存器
D44	d32u	677	对应D5..D4数据寄存器

X5 触发捕捉到  
C204 计数值

### 【PM 程序】



MEMO

### 11.1 附录 A 错误代码原因对照表

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，错误标志 On，原因可能是指令操作数（装置）使用不合法，程序编写回路错误，或运动参数设定不当，可根据错误寄存器的错误代码（16 进制编码）并对照下表，以得知错误原因。

#### ◆ O100/OX 对应之错误信息表

程序区块	O100						
错误类别	程序错误	运动错误					
		X 轴	Y 轴	Z 轴	A 轴	B 轴	C 轴
错误标志	M1953	M1793	M1873	M2033	M2113	M2193	M2273
错误寄存器	D1802	D1857	D1937	D2017	D2097	D2177	D2257
STEP 数	D1803	D1869					
程序区块	OX						
错误类别	程序错误	运动错误					
		X 轴	Y 轴	Z 轴	A 轴	B 轴	C 轴
错误标志	M1793	M1793	M1873	M2033	M2113	M2193	M2273
错误寄存器	D1857	D1857	D1937	D2017	D2097	D2177	D2257
STEP 数	D1869	D1869					

#### ◆ 程序区及运动轴之错误代码(以 16 进制表示)

错误代码	原因	错误代码	原因
0002	使用子程序无内容	0031	正方向脉冲禁止
0003	CJ、CJN、JMP 缺少对应的 Pn	0032	反方向脉冲禁止
0004	主程序中有子程序标志	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	不同子程序中的转移指令指针重复	0045	浮点数转换错误
0009	转移指令与调用子程序指令使用相同标志	0E18	BCD 转换错误
000A	指针与子程序的指针一样	0E19	除法运算错误（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	原点回归速度（V <sub>RT</sub> ）设定错误	C4EE	程序中没有结束指令（M102、M2）
0024	原点回归减速速度（V <sub>CR</sub> ）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		

## MEMO