客服热线 400 - 820 - 9595

绵密网络 专业服务

中达电通已建立了 48 个分支机构及服务网点，并塑建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)8625-5010
传真 :(0791)8625-5102

合肥
电话 :(0551)6281-6777
传真 :(0551)6281-6555

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)8669-0780
传真 :(029)86690780-8000

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

DELTA 台达
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

AS-0249710-08
2022/05/23

中达电通公司版权所有
如有改动,恕不另行通知

AS 系列程序手册

AS 系列程序手册

www.deltaww.com

Smarter. Greener. Together.

AS 系列程序手册

版本修订一览表

版本	变更内容	发行日期
第一版	第一版发行	2016/07/15
第二版	1.第 2.2.7 节更新 SM22-SM24、SM209、SM211、SM456 新增 SM28、SM36、SM270-SM275、SM580、SM1101、SM1408-SM1439 2.第 2.2.14 节新增 SR28、SR36、SR421-SR451、SR750-SR765、SR820-SR893 及更新 SR902 3.第 3.1.2 节更新 API2800~2807 的说明 4.第 6 章更新 API0104、API0105、API0106、API0107、API0501、API1217、API1224、API1225、API1808、API2208、API2300~API2302、API2701~API2709、API2718、API2720~API2721、API2800、API2801、API2803、 API2804、API2806、API2807、API2808 及新增 API0709、API0710、API1008~API1011、API1608、API2723	2017/01/20
第三版	1.第 2.1.1 节更新位装置 SM 内容 2.第 2.2.7 节更新 SM784-815/SM896-927/SM1440-1447，新增 SM478-479/SM498-499/SM518-519/SM538-539/SM558-559/SM578-579/SM1581-1608/SM1611-1618/SM1621-1628/SM1683-1684/SM1691-1698 3.第 2.2.8 节增加新增 SM 的更新时间 4.第 2.2.14 节新增 SR187-188/SR478-479/SR498-499/SR518-519/SR538-539/SR558-559/SR578-579/SR640-651/SR751-758/SR761-768；更新 SR182-183/SR217-218/SR424/SR825-893/SR1336/SR1376 5.第 2.2.16 节更新第 4 点、第 14 点及第 18 点说明并新增第 15 点说明 6.第 3 章增加新增指令 7.第 6 章增加 API0222/API1012/API1013/API1818/API1906/API2122/API2123/API2808；更新 API0709/API0710/API1601/API1607/API1808/API1812/API1814/API1815/API1817/API2200/API2208/API2300/API2301/API2302/API2303/API2401/API2500/API2700/API2701/API2702/API2703/API2704/API2705/API2706/API2707/API2708/API2709/API2710/API2711/API2712/API2713/API2714/API2715/API2716/API2717/	2017/12/29

版本	变更内容	发行日期
	API2718/API2723/API2800-2808	
第四版	1.第 2 章新增 SM976-990/SM1006-1031/SM1312-1315/ SM1709-1919/SR731-788/SR1312-1315/ SR1435-1511 及增加 AS200 系列适用项目 2.第 3 章增加新增指令 3.第 6 章增加 API0601/API0711/API1226/ API1402-1416/API1819-1820/API2211-2212/ API2809-2811；更新 API0222/API0600/API0702/ API0708/API1004/API1808/API1816/API2200-2203/ API2210/API2701/API2704/API2706/API2707/ API2718/API2721 4.第 7 章更新错误码 7203/8105/8106/8107 的排除方式	2018/03/30
第五版	1.第 1.1.2 节更新机种简易说明中的数字 I/O 模块说明中的输出规格 2.第 2 章新增 SM230-SM261/SM585-SM587/SM1110/ SM1685/SR652-SR655/SR791-SR806/ SR811-SR818/SR823-SR824/SR1015/ SR1520-SR1540/SR1559-SR1592 及更新 SM76-SM91/SM168-SM171/SM455/ SM1006-SM1031/SM1184-SM1199/ SM1216-SM1231/SM1248-SM1263/ SM1271-SM1299/SM1424-SM1439/SM1681/ SM1709-SM1919/SR166-SR179/SR182-SR183/ SR187-SR188/SR215-SR216/SR1000-SR1014/ SR1435-SR1511 3.第 3 章新增指令 API0602/API1417/API2304/ API2724/API2812-API2816 及更新指令 API0711/API1110/API1111/API1402/API1407/ API1408/API1416 4.第 4.4 节增加中断程序内使用间接装置说明 5.第 5 章更新指令 PED/APED/OPED、 NED/ANED/ONED、PLS、PLF 说明 6.第 6 章新增指令 API0602/API1417/API2304/ API2724/API2812-API2816、更新指令 API0018-API0023/API0708-API0711/ API1004-API1007/API1110/API1111/API1226/ API1402-API1410/API1415-API1416/API1816/ API2200/API2201/API2203/API2208/ API2210-API2212/API2300-API2302/API2704/ API2706-API2709/API2712-API2716/API2719/ API2800/API2803/API2805/API2807 及更新高速输出指令一览表 7.第 7.3.1 节及第 7.3.2 节增加报警说明	2019/2/28

版本	变更内容	发行日期
第五版	CH6 API28 ASDA-A2 CAN指令章节 API2803 DRVIC指令码中，将“台达伺服 CAN通讯各轴指定交握参数之特殊寄存器对应表”项目“自定义数值((P0-12))请搭配SM1685启动/关闭”一栏中内容将 SR7791~SR78806 改为 SR791~SR806	2019/6/6
第六版	1.第 1.1.2 节更新 AS300/AS200/AS02/04PU-A/程序编辑电缆/网络通讯电缆机种简介说明、增加 AS100/AS02HC-A/AS04SIL-A/AS-FPFN02/AS-FOPC02 机种内容 2.第 2.1.3 节增加批注说明；第 2.2.7 节 SM 表格增加 AS100 机种内容、将”通讯卡”改为”序列通讯卡”、更新 SM1375-SM1447 说明、新增 SM1448-SM1455/SM1686；第 2.2.11 节增加 16 位计数器搭配 CNT 指令说明；第 2.2.12 节增加 32 位计数器搭配 DCNT 指令说明；第 2.2.13 节增加暂存器性质说明；第 2.2.14 节 SR 表格增加 AS100 机种内容、将”通讯卡”改为”序列通讯卡”、新增 SR42/SR48/SR54/SR60/SR66/SR72/SR78/SR84/SR90/SR96/SR102/SR108/SR114/SR120/SR126/SR132/SR138/SR144/SR150/SR156/SR405/SR406、删除 SR1540；第 2.2.16 节新增 SR168-SR175 说明及 EtherNet/IP 通讯标志/功能暂存器/通讯错误代码说明、更新 SR40-SR161 说明及储存卡/以太网站/数据交换功能相关标志说明 3.第 3 章新增指令 API1014/API1015/API1227/API1418-API1423/API1821/API2817/API2818 相关内容及 API1102-API1108/API1114/API1115/API2806 的 32 位指令相关内容、API1415 指令 LCCAL 修改为 32 位指令 DLCCAL 4.第 4.3 节增加建议使用于一般梯形图语言编辑的指令说明 5.第 6 章所有指令支持数据类型 UINT 更正为不支持、增加指令 API1014/API1015/API1227/API1418-API1423/API1821/API2817/API2818、更新指令 API0202/API0216/API0501/API0600/API0708/API0709/API0710/API0801/API1002/API1003/API1004/API1005/API1006/API1007/API1009/API1102-API1108/API1114/API1115/API1210/API1212/API1213/API1226/API1401/API1402/API1404/API1405/API1407/API1408/API1415/API1416/API1417/API1501-API1504/API1808/API2106/API2116/API2117/API2119/API2200/API2204/API2208/API2209/API2212/API2301/API2704/API2706/API2708/API2709/API2718/API2800/API2803/API2806/API2812-API2815	2020/08/20

版本	变更内容	发行日期
	第 7.1.4 节/第 7.2.8 节增加说明、第 7.2.9 节更新错误码 2001-2027 排除方式说明、第 7.3.1 节更新 AIO 模块及 AS02LC 的 ERROR 灯闪烁为亮 0.2 秒/暗 0.2 秒、第 7.3.4 节 AS00SCM 增加错误代码 16#1500/16#1505	
第七版	1.第 1 章增加机种 AS-PS03C/AS-ATXB 相关信息及更新机种 AS-FEN02/AS-FPFN02/AS-FOPC02 说明 2.第 2 章新增 SM27/SM120-SM151/SM360-SM391/SM603/SM1160-SM1161/SM1700-SM1705/SR27/SR29-SR31/SR1016-SR1019/SR1600-SR1605/SR1700/SR1702/SR1704/SR1706/SR1708/SR1710 相关信息、删除 SM468/SM488/SM508/SM528/SM548/SM568/SR604-SR609 相关信息、第 2.2.16 节更新/新增 SR227/背隙补偿功能/以太网络相关标志/SM230-SM261/脉冲输出自动复位标志/台达专属驱动器 CAN 通讯之自动复位标志/高速比较中断启动 DPLSY 输出说明/CSFO 指令之追随落后补偿使用说明 3.第 3 章增加指令 API1229/API1424/API1425/API1426/API1427 及更新指令 API1228 相关信息 4.第 6 章更新指令 API0501/API0708/API0709/API1004/API1228/API1400/API1401/API1402/API1403/API1404/API1405/API1406/API1407/API1408/API1409/API1410/API1417/API1418/API1419/API1420/API1421/API1422/API1423/API2300/API2301/API2302/API2304/API2703/API2706/API2710/API2712/API2714/API2810/API2817/API2818 说明及新增指令 API1229/PI1424/API1425/API1426/API1427 相关信息 5.第 7.2.9 节新增错误码 16#19B0-16#19FC、第 7.3 节与第 7.4 节增加 AS02/04PU、AS02HC、AS04SIL、AS01DNET 相关信息及更新 AS00SCM 为 AS 远程模块故障排除内容	2021/05/10
第八版	1.第 1 章增加 DIADesigner 软件及机种 AS02ADH-A/AS-FFTP01/AS-FECAT 相关信息 2.第 2 章将步进点继电器更新为一般继电器、新增 SM29/SM38/SM172/SM217/SM222/SM581/SM591-SM596/SM1092/SM1581-SM1684/SM1691-SM1698/SM1921-SM1958、更新 SM465/SM467/SM485/SM487/SM505/SM507/SM525/SM527/SM545/SM547/SM565/SM567/SM585/SM603/SM1581-SM1684/SM1691-SM1698、删除 SR23/SR187/SR188、新增 SR38/SR403/SR404/SR656/SR657/SR658-SR818/	2022/05/23

版本	变更内容	发行日期
第八版	SR1092/SR1093/SR1110-SR1114/SR1191-SR1296/SR1430/SR1431、更新 SR172/SR176-SR179/SR226/SR397/SR623/SR652-SR655/SR658-SR818/SR902/SR1052-SR1083/SR1100-SR1107/SR1120-SR1151/SR1335/SR1375/SR1435/SR1475/SR1520-SR1527/SR1536/SR1537、第 2.2.16 节更新通讯相关功能/PLC 状态变更纪录/以太网络相关标志/数据交换功能标志与寄存器/显示现状模块数量和机种代码/脉冲输出自动复位标志说明 3.第 3 章增加指令 API1016-API1018/API1430/API2214/API2215/API2216/API2305/API2820/API2821/API2822/API2823 及更新指令 API1226/API1228 相关信息 4.第 4.3 节建议使用于一般梯形图语言编辑的指令中指令类别-CANopen 专属通讯，增加指令名称 CANRS 5.第 5 章更新指令 PLF 说明 6.第 6 章增加指令 API1016-API1018/API1430/API2214/API2215/API2216/API2305/API2214/API2215/API2216/API2305/API2820/API2821/API2822/API2823 及更新指令 API0222/API0600/API1100/API1101/API1226/API1228/API1421/API1422/API1424/API1425/API1516/API1819/API1821/API1901/API2704/API2717/API2718/API2719/API2720/API2721/API2724/API2800/API2801/API2802/API2803/API2804/API2805/API2806/API2807/API2809/API2810/API2811/API2812/API2817/API2818 相关信息 7.第 7 章第 7.1.4 节更新下载错误说明，第 7.2 节增加错误码 16#000D/16#0020/16#0021/16#0028/16#0040-16#0047/16#1000-16#10FF/16#140C 相关信息及更新错误码 16#1900-16#191C 说明	2022/05/23

AS 系列程序手册

目录

第 1 章 简介

1.1 一般概述 1-2

1.1.1 相关手册 1-2

1.1.2 机种简易说明 1-2

1.2 软件简述 1-10

1.2.1 编辑软件 1-10

1.2.2 程序组织单元（POU）与工作（TASK） 1-13

第 2 章 装置说明

2.1 装置简介 2-3

2.1.1 装置列表 2-3

2.1.2 I/O 储存区的基本结构 2-4

2.1.3 停电保持储存方式 2-4

2.1.4 停电保持区的装置范围 2-5

2.2 装置功能说明 2-6

2.2.1 数值、常数 2-6

2.2.2 浮点数 2-7

2.2.2.1 单精度浮点数（32 位浮点数） 2-7

2.2.2.2 十进浮点数 2-8

2.2.3 字符串 2-9

2.2.4 输入继电器 X 2-10

2.2.5 输出继电器 Y 2-10

2.2.6 辅助继电器 M 2-11

2.2.7 特殊辅助继电器 SM 2-11

2.2.8 特殊辅助继电器 SM 的更新时间 2-58

2.2.9 一般继电器 S 2-65

2.2.10 定时器 T 2-66

2.2.11 计数器 C 2-68

2.2.1232 位计数器 HC 2-70

2.2.13 数据寄存器 D 2-72
2.2.14 特殊数据寄存器 SR 2-72
2.2.15 特殊数据寄存器 SR 的更新时间 2-116
2.2.16 SM/SR 补充说明.....2-121
2.2.17 变址寄存器 E.....2-153
2.2.18 文件寄存器 FR.....2-153

第 3 章 指令表
3.1 指令类型..... 3-2
3.1.1 基本指令..... 3-2
3.1.2 应用指令..... 3-2
3.2 指令表说明..... 3-3
3.2.1 基本指令..... 3-3
3.2.2 应用指令（依 API 号码排序） 3-4
3.2.3 应用指令（依英文字母排序）..... 3-4
3.2.4 指令装置表说明 3-5
3.3 基本指令一览表 3-7
3.4 应用指令一览表 3-10
3.4.1 应用指令（依 API 号码排序） 3-10
3.4.2 应用指令（依英文字母排序）.....3-47

第 4 章 指令结构
4.1 API 应用指令组成说明 4-2
4.2 操作数使用说明 4-5
4.3 指令使用限制 4-6
4.4 变址（E）说明 4-8
4.5 指针寄存器（PR）说明 4-10
4.6 计时器指针寄存器（TR）说明.....4-12
4.7 16 位计数器指针寄存器（CR）说明.....4-13
4.8 32 位计数器指针寄存器（HCR）说明 4-14
4.9 文件寄存器（FR）说明 4-16

第 5 章 基本指令

5.1 基本指令一览表 5-2
5.2 基本指令说明 5-3

第 6 章 指令表

6.1 比较操作指令 6-4
6.1.1 比较操作指令一览表 6-4
6.1.2 比较操作指令说明 6-7
6.2 四则运算指令说明 6-45
6.2.1 四则运算指令一览表 6-45
6.2.2 四则运算指令说明 6-46
6.3 数据转换指令说明 6-76
6.3.1 数据转换指令一览表 6-76
6.3.2 数据转换指令说明 6-77
6.4 数据转移指令說明 6-113
6.4.1 数据转移指令一览表 6-113
6.4.2 数据转移指令说明 6-114
6.5 程序跳转指令 6-137
6.5.1 程序跳转指令一览表 6-137
6.5.2 程序跳转指令说明 6-138
6.6 程序执行控制指令 6-146
6.6.1 程序执行控制指令一览表 6-146
6.6.2 程序执行控制指令说明 6-147
6.7 I/O 更新指令 6-156
6.7.1 I/O 更新指令一览表 6-156
6.7.2 I/O 更新指令说明 6-157
6.8 便利指令 6-163
6.8.1 便利指令一览表 6-163
6.8.2 便利指令说明 6-164
6.9 逻辑操作指令 6-208
6.9.1 逻辑操作指令一览表 6-208
6.9.2 逻辑操作指令说明 6-209

6.10 旋转指令 6-229
6.10.1 旋转指令一览表 6-229
6.10.2 旋转指令说明 6-230
6.11 计时计数指令 6-240
6.11.1 计时计数指令一览表 6-240
6.11.2 计时计数指令说明 6-241
6.12 移位指令 6-284
6.12.1 移位指令一览表 6-284
6.12.2 移位指令说明 6-285
6.13 数据处理指令 6-318
6.13.1 数据处理指令一览表 6-318
6.13.2 数据处理指令说明 6-319
6.14 结构建立指令 6-376
6.14.1 结构建立指令一览表 6-376
6.14.2 结构建立指令说明 6-377
6.15 模块的数据读写指令 6-384
6.15.1 模块的数据读写指令一览表 6-384
6.15.2 模块的数据读写指令说明 6-385
6.16 浮点数指令 6-500
6.16.1 浮点数指令一览表 6-500
6.16.2 浮点数指令说明 6-501
6.17 万年历指令 6-527
6.17.1 万年历指令一览表 6-527
6.17.2 万年历指令说明 6-528
6.18 外围设备指令 6-555
6.18.1 外围设备指令一览表 6-555
6.18.2 外围设备指令说明 6-556
6.19 通讯指令 6-569
6.19.1 通讯指令一览表 6-569
6.19.2 通讯指令說明 6-570
6.19.3 通讯标志与寄存器说明 6-639
6.20 其它指令 6-641
6.20.1 其它指令一览表 6-641

6.20.2 其它指令说明 6-642
6.21 字串处理指令 6-653
6.21.1 字串处理指令一览表 6-653
6.21.2 字串处理指令说明 6-654
6.22 以太网控制指令 6-703
6.22.1 以太网控制指令一览表 6-703
6.22.2 以太网控制指令说明 6-704
6.23 储存卡/文件寄存器读写指令 6-757
6.23.1 储存卡/文件寄存器读写指令一览表 6-757
6.23.2 储存卡/文件寄存器读写指令说明 6-758
6.24 任务控制指令 6-780
6.24.1 任务控制指令一览表 6-780
6.24.2 任务控制指令说明 6-781
6.25 SFC 控制指令 6-785
6.25.1 SFC 控制指令一览表 6-785
6.25.2 SFC 控制指令说明 6-786
6.26 高速输出指令 6-792
6.26.1 高速输出指令一览表 6-792
6.26.2 高速输出指令说明 6-793
6.27 台达专用 CANopen 与 EtherCAT 通讯指令 6-890
6.27.1 通讯指令一览表 6-890
6.27.2 通讯指令说明 6-891
6.27.3 台达专用 CANopen 通讯之常见问题与排除方法 6-962

第 7 章 故障排除
7.1 故障排除 7-3
7.1.1 基本检查 7-3
7.1.2 清除错误状态 7-3
7.1.3 故障排除程序 7-4
7.1.4 查询错误记录与位置 7-5
7.2 CPU 模块故障排除 7-7
7.2.1 ERROR 灯号常亮 7-7
7.2.2 ERROR 灯号一般闪烁（亮 0.5 秒、暗 0.5 秒） 7-7

7.2.3 ERROR 灯号快速闪烁（亮 0.2 秒、暗 0.2 秒） 7-9
7.2.4 ERROR 灯号慢速闪烁（亮 1 秒、暗 3 秒）.....7-10
7.2.5 BAT. LOW 灯号常亮 7-10
7.2.6 BAT. LOW 灯号一般闪烁（亮 0.5 秒、暗 0.5 秒） 7-11
7.2.7 RUN 与 ERROR 灯号同时闪烁（亮 0.5 秒、暗 0.5 秒）.....7-11
7.2.8 RUN 与 ERROR 灯号交替闪烁（RUN 亮 0.5 秒、换 ERROR 亮 0.5 秒）
.....7-11
7.2.9 其它错误但无灯号显示.....7-11
7.3 其它模块故障排除.....7-18
7.3.1 模拟（AD/DA/XA）及温度（RTD/TC）模块故障排除.....7-18
7.3.1.1 ERROR 灯常亮.....7-18
7.3.1.2 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒） 7-18
7.3.2 AS02/04PU 定位模块故障排除.....7-19
7.3.2.1 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒） 7-19
7.3.3 AS02HC 高速计数模块故障排除.....7-19
7.3.3.1 ERROR 灯常亮.....7-19
7.3.3.2 ERROR 灯闪烁（亮 0.5 秒、暗 0.5 秒） 7-19
7.3.4 AS02LC 称重模块故障排除.....7-20
7.3.4.1 ERROR 灯常亮.....7-20
7.3.4.2 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒） 7-20
7.3.5 AS04SIL IO-Link 通讯模块故障排除 7-21
7.3.5.1 MS 灯闪烁（亮 0.2 秒、暗 0.2 秒） 7-21
7.3.5.2 NS 灯恒亮.....7-21
7.3.5.3 NS 灯闪烁（亮 1 秒、暗 1 秒） 7-21
7.3.6 AS00SCM 为串行通讯模块故障排除.....7-23
7.3.6.1 ERROR 灯常亮.....7-23
7.3.6.2 ERROR 灯一般闪烁（亮 0.5 秒、暗 0.5 秒） 7-23
7.3.7 AS00SCM 为 AS 远程模块故障排除 7-24
7.3.7.1 ERROR 灯常亮.....7-24
7.3.7.2 ERROR 灯慢速闪烁（亮 1 秒、暗 1 秒） 7-24
7.3.7.3 ERROR 灯一般闪烁（亮 0.5 秒、暗 0.5 秒） 7-25
7.3.7.4 ERROR 灯快速闪烁（亮 0.2 秒、暗 0.2 秒） 7-25
7.3.8 AS01DNET 为通讯模块故障排除.....7-25

7.4 模块错误代码对应灯号及状态说明 7-26
7.4.1 CPU 模块错误代码对应灯号及状态 7-26
7.4.2 模拟与温度模块错误代码对应灯号 7-32
7.4.3 AS02/04PU 定位模块错误代码对应灯号 7-33
7.4.4 AS02HC 高速计数模块错误代码对应灯号 7-33
7.4.5 AS02LC 称重模块错误代码对应灯号 7-33
7.4.6 AS04SIL IO-Link 通讯模块错误代码对应灯号 7-34
7.4.7 AS00SCM 为串行通讯模块错误代码对应灯号 7-35
7.4.8 AS00SCM 为 AS 远程模块错误代码对应灯号 7-35
7.4.9 AS01DNET 为主从站模式时的数字显示器显示说明 7-35
7.4.10 AS01DNET 为 RTU 模式时的数字显示器显示说明 7-37

MEMO

1

第1章 简介

目录

1.1 一般概述 1-2
1.1.1 相关手册 1-2
1.1.2 机种简易说明 1-2
1.2 软件简述 1-10
1.2.1 编辑软件 1-10
1.2.2 程序组织单元（POU）与工作（TASK） 1-13

1

1.1 一般概述

本手册描述 AS 系列 PLC 主机程序设计相关的内容介绍，基本指令及应用指令操作方法说明。AS 系列 PLC 主机程序设计可使用 ISPSoft 软件或 DIADesigner 软件，本手册以 ISPSoft 软件为例做说明，若需 DIADesigner 软件相关说明，请参考 DIADesigner 软件使用手册。

1.1.1 相关手册

AS 系列 PLC 相关手册的组成如下表所示。请搭配使用。

- AS 系列快速入门手册：
  主要介绍如何让使用者在详细阅读完所有手册之前，能先快速的建立并使用本系统。
- AS 系列程序手册：
  描述AS系列PLC主机程序设计相关的内容介绍，基本指令及应用指令操作方法说明。
- ISPSoft/DIADesigner 使用手册：
  有关ISPSoft/DIADesigner操作、程序语言介绍（Ladder、SFC、FBD、ST）、POU概念及工作（Task）概念等。
- AS 系列硬件及操作手册：
  电气规格、外观及尺寸、CPU功能介绍、各项装置、I/O配置及故障排除等。
- AS 系列模块手册：
  特殊模块使用介绍，例如网络模块、模拟模块及温度模块等。

1.1.2 机种简易说明

分类	机种名称	说明
电源模块	AS-PS02	输入：100-240VAC，50/60Hz 交流电源输入 输出（供 PLC 内部使用）：24VDC/2A，48W
	AS-PS02A	输入：100-240VAC，50/60Hz 交流电源输入 输出（供 PLC 内部使用）：24VDC/1.5A，36W 输出（供外部使用）：24VDC/0.5A，12W
电源配线辅助模块	AS-ATXB	将主机电源配线位置由左侧底部移至产品下方处上半部，以改善 AS 主机安装于配盘最左侧时不易配线之情况。
AS300 系列 CPU 模块	AS332P-A	CPU 模块，24VDC 电源输入，晶体型(PNP)输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、两张功能卡（选配）以及 32 点 IO（16DI+16DO），支持最大 1024 点 I/O，程序容量 128k steps，采用高密度端子

分类	机种名称	说明
	AS332T-A	CPU 模块，24VDC 电源输入，晶体型(NPN)输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、两张功能卡 (选配) 以及 32 点 IO (16DI+16DO)，支持最大 1024 点 I/O，程序容量 128k steps，采用高密度端子
	AS324MT-A	CPU 模块，24VDC 电源输入，差动+晶体型 (NPN) 输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、两张功能卡(选配)以及 24 点 IO(12DI+12DO)，支持最大 1016 点 I/O，程序容量 128k steps，采用高密度端子
AS300 系列 CPU 模块	AS320P-B	CPU 模块，24VDC 电源输入，晶体型（PNP）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、两张功能卡（选配）以及 20 点 IO（8DI+12DO），支持最大 1024 点 I/O，程序容量 128k steps，采用脱落式端子
	AS320T-B	CPU 模块，24VDC 电源输入，晶体型（NPN）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、两张功能卡（选配）以及 20 点 IO（8DI+12DO），支持最大 1024 点 I/O，程序容量 128k steps，采用脱落式端子
	AS300N-A	CPU 模块，24VDC 电源输入，无输入/输出点，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、两张功能卡（选配），支持最大 1024 点 I/O，程序容量 128k steps
AS200 系列 CPU 模块	AS228P-A	CPU 模块，24VDC 电源输入，晶体型（PNP）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 28 点 IO（16DI+12DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS228T-A	CPU 模块，24VDC 电源输入，晶体型（NPN）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 28 点 IO（16DI+12DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS228R-A	CPU 模块，24VDC 电源输入，继电器（Relay）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 28 点 IO（16DI+12DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS218PX-A	CPU 模块，24VDC 电源输入，晶体型（PNP）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 18 点 IO（8DI+6DO+2AI+2AO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS218TX-A	CPU 模块，24VDC 电源输入，晶体型（NPN）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 18 点 IO（8DI+6DO+2AI+2AO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子

分类	机种名称	说明
	AS218RX-A	CPU 模块 ·24VDC 电源输入, 继电器(Relay)输出, 内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 18 点 IO（8DI+6DO+2AI+2AO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
AS100 系列 CPU 模块	AS132P-A	CPU 模块，100~240VAC 电源输入，晶体型（PNP）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 32 点 IO（16DI+16DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS132T-A	CPU 模块，100~240VAC 电源输入，晶体型（NPN）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 32 点 IO（16DI+16DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS132R-A	CPU 模块，100~240VAC 电源输入，继电器（Relay）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 32 点 IO（16DI+16DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS148P-A	CPU 模块，100~240VAC 电源输入，晶体型（PNP）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 48 点 IO（24DI+24DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS148T-A	CPU 模块，100~240VAC 电源输入，晶体型（NPN）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 48 点 IO（24DI+24DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS148R-A	CPU 模块，100~240VAC 电源输入，继电器（Relay）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 48 点 IO（24DI+24DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS164P-A	CPU 模块，100~240VAC 电源输入，晶体型（PNP）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 64 点 IO（32DI+32DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
	AS164T-A	CPU 模块，100~240VAC 电源输入，晶体型（NPN）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 64 点 IO（32DI+32DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子

分类	机种名称	说明
	AS164R-A	CPU 模块，100~240VAC 电源输入，继电器（Relay）输出，内建 Ethernet、RS-485*2、USB 通讯口、Micro SD 卡接口、CAN 通讯口以及 64 点 IO（32DI+32DO），支持最大 1024 点 I/O，程序容量 64k steps，采用脱落式端子
数位 I/O 模块	AS08AM10N-A	24VDC，5mA，8 点输入，弹片式端子台
	AS08AN01P-A	5 ~ 30VDC，0.5A/点，4A/共点，8 输出点，源型输出，弹片式端子台
	AS08AN01R-A	240VAC/24VDC，2A/点，8A/共点，8 输出点，继电器，弹片式端子台
数位 I/O 模块	AS08AN01T-A	5 ~ 30VDC，0.5A/点，4A/共点，8 输出点，漏型输出，弹片式端子台
	AS16AM10N-A	24VDC，5mA，16 点输入，弹片式端子台
	AS16AN01P-A	5 ~ 30VDC，0.5A/点，4A/共点，16 输出点，源型输出，弹片式端子台
	AS16AN01R-A	240VAC/24VDC，2A/点，8A/共点，16 输出点，继电器，弹片式端子台
	AS16AN01T-A	5 ~ 30VDC，0.5A/点，4A/共点，16 输出点，漏型输出，弹片式端子台
	AS16AP11P-A	24VDC，5mA，8 输入点，5 ~ 30VDC，0.5A/点，4A/共点，8 输出点，源型输出，弹片式端子台
	AS16AP11R-A	24VDC，5mA，8 输入点，240VAC/24VDC，2A/点，8A/共点，8 输出点，继电器，弹片式端子台
	AS16AP11T-A	24VDC，5mA，8 输入点，5 ~ 30VDC，0.5A/点，4A/共点，8 输出点，漏型输出，弹片式端子台
	AS32AM10N-A	24VDC，3.2mA，32 点输入，牛角座连接器（MIL）
	AS32AN02T-A	5 ~ 30VDC，0.1A/点，3.2A/共点，32 点输出，漏型输出，牛角座连接器（MIL）
	AS64AM10N-A	24VDC，3.2mA，64 点输入，牛角座连接器（MIL）
	AS64AN02T-A	5 ~ 30VDC，0.1A/点，3.2A/共点，64 点输出，漏型输出，牛角座连接器（MIL）
模拟 I/O 模块	AS04AD-A	4 通道模拟信号输入 16 位分辨率 0~10V，0/1~5V，-5V~+5V，-10~+10V，0/4~20mA，-20mA~+20mA 转换时间：2ms/通道
	AS08AD-B	8 通道模拟信号输入 16 位分辨率 0~10V，0/1~5V，-5V~+5V，-10~+10V 转换时间：2ms/通道

分类	机种名称	说明
	AS08AD-C	8 通道模拟信号输入 16 位分辨率 0/4~20mA，-20mA~+20mA 转换时间：2ms/通道
	AS04DA-A	4 通道模拟信号输出 12 位分辨率-10v~10V，0~20mA，4~20mA 转换时间：2ms/通道
	AS06XA-A	4 通道模拟信号输入 16 位分辨率 ·0~10V ·0/1~5V ·-5V~+5V ·-10~+10V ·0/4~20mA，-20mA~+20mA 转换时间：2ms/通道 2 通道模拟信号输出 12 位分辨率-10V~10V，0~20mA，4~20mA 转换时间：2ms/通道
	AS02ADH-A	2 通道模拟信号输入 16 位分辨率 ·0~10V ·0/1~5V ·-5V~+5V ·-10~+10V ·0/4~20mA，-20mA~+20mA 高速转换，最快转换时间：20us 全隔离（包含通道间隔离） 提供 2000 笔数值、峰值与触发记录功能 提供 FIR 低通与带通滤波功能
温度模块	AS04RTD-A	4 通道 2 线式或 3 线式 RTD 温度感测 传感器型式：Pt100 / Ni100 / Pt1000 / Ni1000 / JPt100 / LG-Ni1000 / Cu50 / Cu100 / 0~300Ω / 0~3000Ω 分辨率：0.1℃/0.1℉（16 位转换器） 转换时间：200ms/通道
	AS06RTD-A	6 通道 2 线式或 3 线式 RTD 温度感测 传感器型式：Pt100 / Ni100 / Pt1000 / Ni1000 / JPt100 / LG-Ni1000 / Cu50 / Cu100 / 0~300Ω / 0~3000Ω 分辨率：0.1℃/0.1℉（16 位转换器） 转换时间：200ms/通道
	AS04TC-A	4 通道热电耦温度感测 传感器型：J、K、R、S、T、E、N、B 或-100~+100mV 分辨率：0.1℃/0.1℉（24 位转换器） 转换时间：200ms/通道
	AS08TC-A	8 通道热电耦温度感测 传感器型：J、K、R、S、T、E、N、B 或-100~+100mV 分辨率：0.1℃/0.1℉（24 位转换器） 转换时间：200ms/通道

分类	机种名称	说明
秤重模块	AS02LC-A	2 通道 4 线式或 6 线式荷重传感器（Load Cell） 特征值：1、2、4、6、20、40、80 mV/V 精度误差值：万分之一（1/10000）(在 50ms 转换时间下） ADC 分辨率：24 位 转换时间：可选择 2.5 ~ 400ms（共 9 项）
定位模块	AS02PU-A	2 轴运动控制 5~24VDC，1 组(A/B/Z 相)差动输入，硬件输入带宽最高 200KHz 24VDC，5mA，5 点输入，硬件输入带宽最高 1KHz 5VDC，2 轴（4 点）差动输出，最高输出频率 200KHz
	AS04PU-A	4 轴运动控制 24VDC，5mA，6 点输入，硬件输入带宽最高 1KHz 5~30VDC，0.1A，4 轴(8 点)NPN 输出，最高输出频率 100KHz
计数模块	AS02HC-A	2 通道高速计数模块 计数方式可选择脉冲型输入（最大 200kHz）或 SSI 通讯接口输入（最大 1.25MHz） 4 点开集极输出，5~30VDC，0.1A，可搭配高速比较输出
网络模块	AS00SCM-A	串行通讯模块，内建两个通讯接口，可自行搭配通讯卡，支持 MODBUS 协议
	AS01DNET-A	DeviceNet 通讯模块，可以做 DeviceNet 主站或从站。
	AS04SIL-A	IO-Link 模块，内建四个 IO-Link 通讯端口
远程 IO 通讯模块	AS00SCM-A + AS-FCOPM	通讯模块搭配 AS-FCOPM 功能卡
	AS00SCM-A + AS-FEN02	通讯模块搭配 AS-FEN02 功能卡
	AS01DNET-A（RTU）	DeviceNet 远程 IO 从站，其右侧可以连接 AS 系列扩展模块 (包括数字模块、模拟模块、温度模块等)。
功能卡	AS-F232	串行通讯端口，RS232 接口，支持主/从站模式
	AS-F422	串行通讯端口，RS422 接口，支持主/从站模式
	AS-F485	串行通讯端口，RS485 接口，支持主/从站模式
	AS-FCOPM	CANopen 通讯端口，支持 DS301 或 AS 系列远程或台达专属伺服马达控制
	AS-F2AD	2 通道模拟信号输入 0~10V（12 位分辨率），4~20mA（11 位分辨率） 转换时间：3ms/通道
	AS-F2DA	2 通道模拟信号输出 0~10V，4~20mA（12 位分辨率） 转换时间：2ms/通道

分类	机种名称	说明
	AS-FEN02	内建两个以太网接口，支持交换器功能 支持 MODBUS TCP 及 EtherNet/IP Adapter 支持 AS 系列远程控制 仅限使用于 AS300 系列主机
	AS-FPFN02	内建两个以太网接口，支持交换器功能 支持 PROFINET Device（从站） 仅限使用于 AS300 系列主机
	AS-FOPC02	内建两个以太网接口，支持交换器功能 支持 OPC UA Server 与 MODBUS TCP 仅限使用于 AS300 系列主机
	AS-FFTP01	内建一个以太网接口 支持 FTP Server / OPC UA Server / MQTT Client 支持 Web Server / Data Log 仅限使用于 AS300 系列主机
	AS-FECAT	内建两个以太网接口，支持交换器功能（EtherCAT Master 启用前） 支持 MODBUS TCP Server（支持 1 条联机数） 支持 EtherCAT Master 16 轴点到点定位控制（仅适用台达专属驱动器） 仅限使用于 AS300 系列主机
程序编辑电缆	UC-PRG015-01A（1.5M）	PLC 连接 PC 的程序规划电缆（mini USB），适用 AS332T-A、AS332P-A 及 AS324MT-A 机种。
	UC-PRG030-01A（3M）	PLC 连接 PC 的程序规划电缆（mini USB），适用 AS332T-A、AS332P-A 及 AS324MT-A 机种。
	UC-PRG030-20A（3M）	PLC 连接 PC 的程序规划电缆（RJ45），适用 AS332T-A、AS332P-A、AS324MT-A、AS-FEN02、AS-FPFN02 及 AS-FECAT 机种。
I/O 连接线	UC-ET010-24B（1M） UC-ET020-24B（2M） UC-ET030-24B（3M） UC-ET010-24D（1M） UC-ET020-24D（2M） UC-ET030-24D（3M）	I/O 连接线（牛角座）（40Pin ↔ 40Pin）（屏蔽），适用 AS32AM10N-A、AS32AN02T-A、AS64AM10N-A 及 AS64AN02T-A 机种。 I/O 连接线（牛角座）（40Pin ↔ 20Pin x2）（屏蔽）适用 AS332T-A、AS332P-A、AS324MT-A、AS32AM10N-A、AS32AN02T-A、AS64AM10N-A 及 AS64AN02T-A 机种。
配线模块	UB-10-ID16A	I/O 配线模块（16 点输入/输出）（20-Pin 牛角座）适用 AS332T-A、AS332P-A、AS324MT-A、AS32AM10N-A、AS32AN02T-A、AS64AM10N-A 及 AS64AN02T-A 机种。

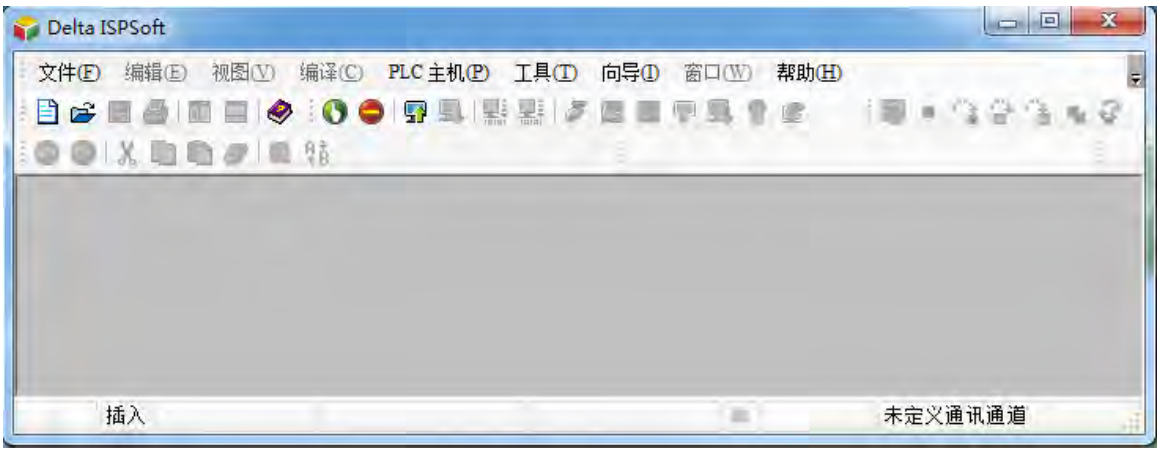
分类	机种名称	说明
	UB-10-ID32A	I/O 配线模块(32 点输入)(40-Pin 牛角座),适用 AS32AM10N-A 及 AS64AM10N-A。
	UB-10-IO32D	牛角座轉換彈片型端子座,40-Pin 牛角座直接轉換為 40-Pin 彈片型端子座，適合 AS332T-A、AS332P-A、AS324MT-A、AS32AM10N-A 及 AS32AN02T-A。
	UB-10-OR16A	I/O 配线模块（16 点继电器输出）(20-Pin 牛角座)，NPN，适用 AS332T-A、AS32AN02T-A 及 AS64AN02T-A。
	UB-10-OR16B	I/O 配线模块（16 点继电器输出）(20-Pin 牛角座)，PNP，适用 AS332P-A。
	UB-10-OT32A	I/O 配线模块（32 点晶体管输出）(40-Pin 牛角座)，NPN，适用 AS32AN02T-A 及 AS64AN02T-A。
网络通讯电缆	UC-CMC003-01A（0.3M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-CMC005-01A（0.5M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-CMC010-01A（1M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-CMC015-01A（1.5M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-CMC020-01A（2M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-CMC030-01A（3M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-CMC050-01A（5M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-CMC100-01A（10M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-CMC200-01A（20M）	CANopen 通讯连接电缆，适用 AS-FCOPM 机种。
	UC-EMC003-02A（0.3M）	Ethernet 通讯连接电缆，适用 AS 系列 CPU 模块、AS-FEN02、AS-FPFN02 以及 AS-FECAT 机种。
	UC-EMC005-02A（0.5M）	Ethernet 通讯连接电缆，适用 AS 系列 CPU 模块、AS-FEN02、AS-FPFN02 以及 AS-FECAT 机种。
	UC-EMC010-02A（1M）	Ethernet 通讯连接电缆，适用 AS 系列 CPU 模块、AS-FEN02、AS-FPFN02 以及 AS-FECAT 机种。
	UC-EMC020-02A（2M）	Ethernet 通讯连接电缆，适用 AS 系列 CPU 模块、AS-FEN02、AS-FPFN02 以及 AS-FECAT 机种。
	UC-EMC050-02A（5M）	Ethernet 通讯连接电缆，适用 AS 系列 CPU 模块、AS-FEN02、AS-FPFN02 以及 AS-FECAT 机种。
	UC-EMC100-02A（10M）	Ethernet 通讯连接电缆，适用 AS 系列 CPU 模块、AS-FEN02、AS-FPFN02 以及 AS-FECAT 机种。
	UC-EMC200-02A（20M）	Ethernet 通讯连接电缆，适用 AS 系列 CPU 模块、AS-FEN02、AS-FPFN02 以及 AS-FECAT 机种。

1

1.2 软件简述

1.2.1 编辑软件

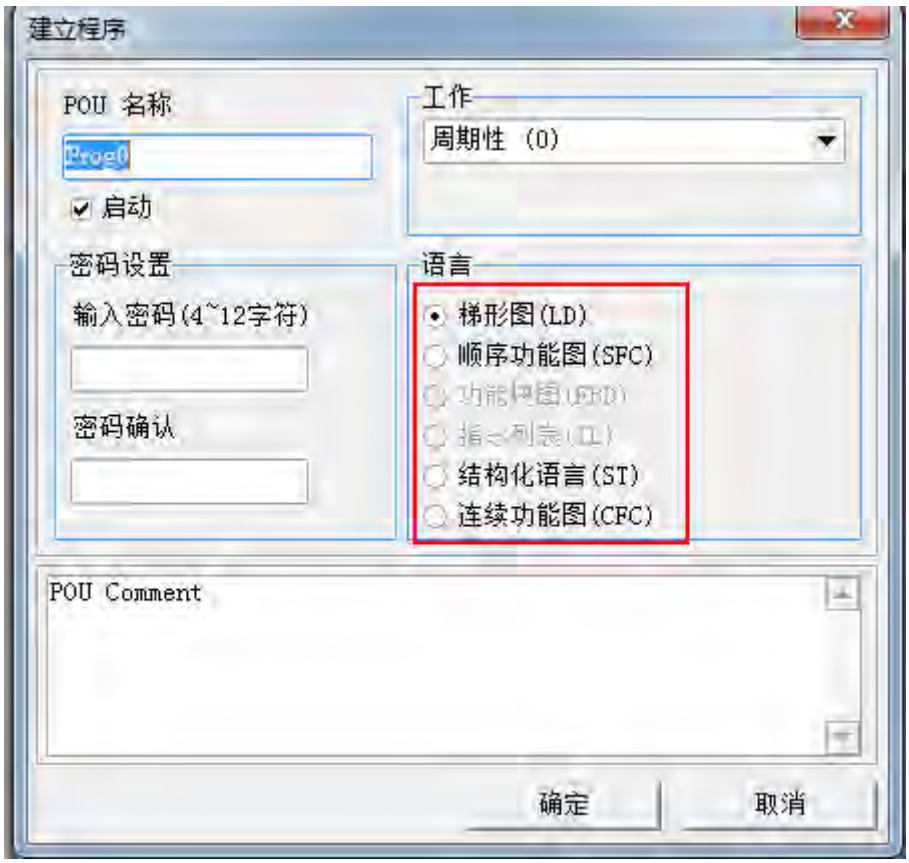
编辑软件 ISPSoft 的概括：



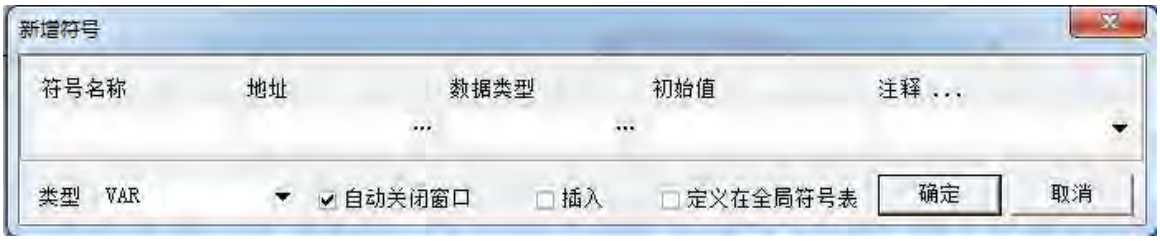
- 支持编程语言：包括梯形图（LD）、顺序功能图（SFC）、结构化语言（ST）、连续功能图（CFC）及 C 语言。

注 1：ISPSoft V3.01 版以上支持连续功能图（CFC）。

注 2：ISPSoft V3.08 版以上支持 C 语言。



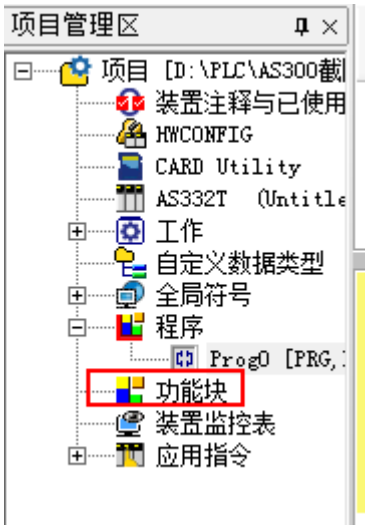
- 使用变量的观念，让用户可自行定义变量符号来取代 PLC 的装置名称，除了大大地提高程序的可读性之外，更省去了用户费心配置装置地址的时间。



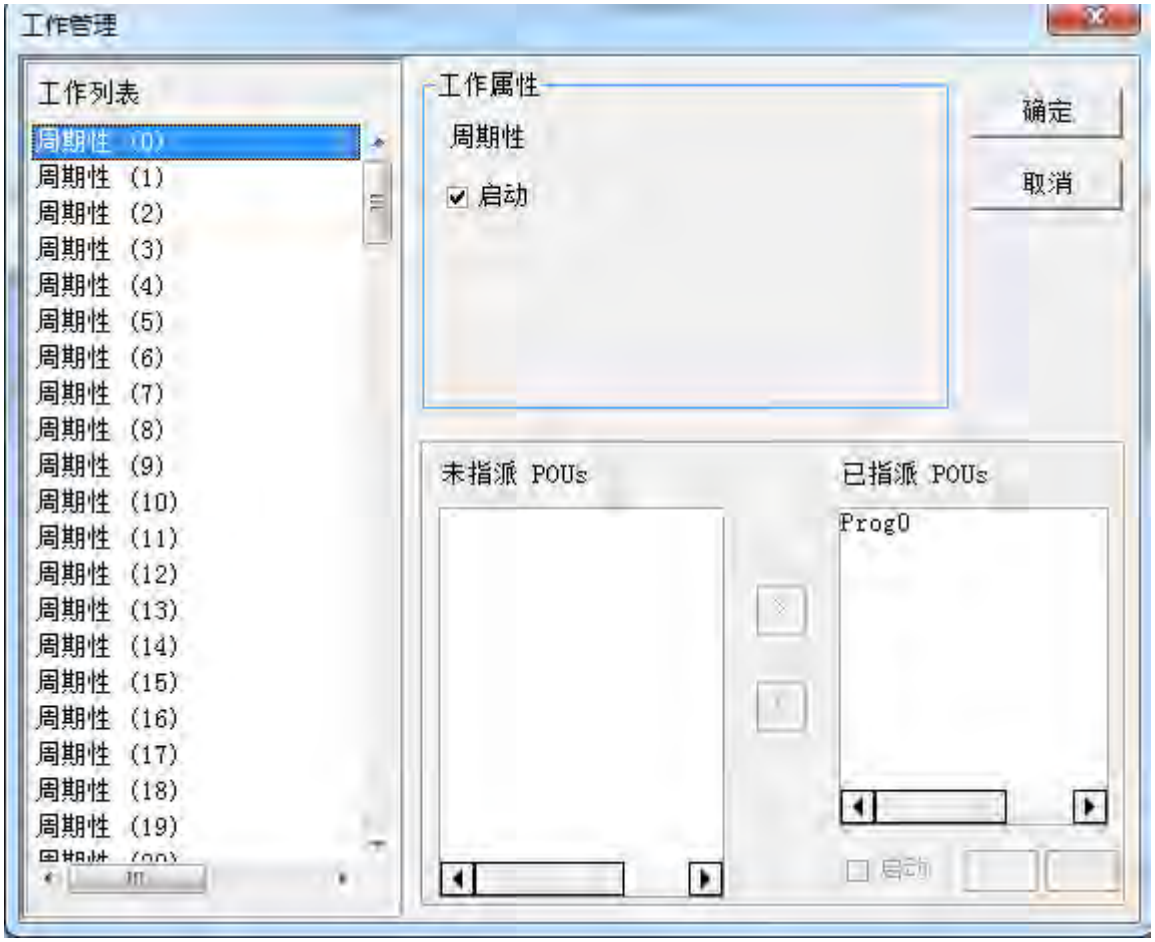
局部符号

类型	符号名称	地址	数据类型	初始值	符号注释
VAR	IN_0	N/A [Auto]	BOOL	FALSE	
VAR	IN_1	N/A [Auto]	BOOL	FALSE	
VAR	IN_2	N/A [Auto]	BOOL	FALSE	
VAR	OUT_0	N/A [Auto]	BOOL	FALSE	
VAR	OUT_1	N/A [Auto]	BOOL	FALSE	

- 导入程序组织单元（POU）的架构，除了可将原本的主程序切割成若干个程序单元之外，并以函式（Function）与功能块（Function Block）来取代传统的子程序，让程序的架构够可以更模块化也更便于管理。



- 使用工作 (Task) 的观念来管理程序的执行顺序，将程序开发的工作提升至项目管理的层次，让大规模的程序开发变的更便于管理。



1.2.2 程序组织单元（POU）与工作（TASK）

程序组织单元（Program Organization Unit，简称 POU）是建构 PLC 程序的基本元素，有别于传统的 PLC 程序，由 IEC61131-3 所导入的程序架构，其特色在于将原本一大段的程序分割成若干个小单元，而这些小单元便称之为 POU。POU 大致可分为以下三种。

1. 程序（PROG）：程序 POU 于 PLC 程序中扮演着主要程序的角色，其执行方式可由设计者自行定义为周期扫描或中断模式，并可在工作（TASK）列表中安排各个程序 POU 的扫描顺序。
2. 功能块（FB）：功能块（Function Block，简称 FB）本身所代表的意义类似于子程序，而在功能块内部所定义的程序，则必须要在该功能块被程序 POU 所调用，并输入相关的参数之后才会执行。
3. 函式（FC）：函式（Function，简称 FC）本身所代表的意义较接近于宏指令，亦即我们可将多笔的运算指令或功能写在函式 POU 当中，之后再于程序 POU 或功能块 POU 当中调用使用即可。

工作（TASK），是指规定使各个程序按照何种顺序或中断条件执行的功能。工作（TASK）的意义在于赋予各个程序 POU 一个明确的执行任务，并指定每个程序 POU 之间的执行顺序或是启动的方式。

基本上一个项目中的程序 POU 并非都会参与实际的执行，而是必须在指派 TASK 之后才可确定该 POU 是否执行以及如何执行；当 POU 未被指派 TASK 时，该 POU 仅会被当做一般的原始码而与项目一起储存，本身并不会被编译为 PLC 的执行码。此外，仅有程序 POU 需要被指派 TASK，至于功能块（FB）或函式（FC）的执行方式，则是依据调用它们的上位 POU 程序而定。TASK 的运行方式大致可分为周期性、定时中断，以及条件中断 3 种

1. 周期性：被指派至周期性 TASK 的程序 POU，其执行方式便是单纯的来回扫描，按照顺序执行。
2. 定时中断工作（TASK）：被指派至定时中断工作的程序 POU，当定时中断工作的时间到达后，被分配至该定时中断工作的所有程序 POU 便会依照排列顺序执行。
3. 条件中断工作（TASK）：条件中断亦可区分为多种类型，如外部中断、I/O 中断…等，规划前必须先确认所选用的项目机种支持哪些中断模式。当程序 POU 被分配至条件中断 TASK 时，其功能便类似中断子程序，当中断条件成立时，如外部触发中断的某个接点被触发，被分配至该 TASK 的所有 POU 便会依照排列顺序执行一遍。

MEMO

2

第2章 装置说明

目录

2.1 装置简介 2-3
2.1.1 装置列表 2-3
2.1.2 I/O 储存区的基本结构 2-4
2.1.3 停电保持储存方式 2-4
2.1.4 停电保持区的装置范围 2-5
2.2 装置功能说明 2-6
2.2.1 数值、常数 2-6
2.2.2 浮点数 2-7
2.2.2.1 单精度浮点数（32 位浮点数） 2-7
2.2.2.2 十进浮点数 2-8
2.2.3 字符串 2-9
2.2.4 输入继电器 X 2-10
2.2.5 输出继电器 Y 2-10
2.2.6 辅助继电器 M 2-11
2.2.7 特殊辅助继电器 SM 2-11
2.2.8 特殊辅助继电器 SM 的更新时间 2-58
2.2.9 一般继电器 S 2-65
2.2.10 定时器 T 2-66
2.2.11 计数器 C 2-68
2.2.12 32 位计数器 HC 2-70
2.2.13 数据寄存器 D 2-72
2.2.14 特殊数据寄存器 SR 2-72
2.2.15 特殊数据寄存器 SR 的更新时间 2-116
2.2.16 SM/SR 补充说明 2-121
2.2.17 变址寄存器 E 2-153

2.2.18 文件寄存器 FR..... 2-153

2.1 装置简介

本章节针对 PLC 所处理的数值、字符串和输入、输出、辅助继电器、定时器、计数器及数据寄存器等等各种装置的配置和功能做说明。

2.1.1 装置列表

形式	装置名称		装置数	范围
位装置	输入继电器	X	1024	X0.0~X63.15
	输出继电器	Y	1024	Y0.0~Y63.15
	数据寄存器	D	48,0000	D0.0~D29999.15
		W	48,0000	W0.0~W29999.15 仅供编辑软件使用*4
	辅助继电器	M	8192	M0~M8191
	特殊辅助标志	SM	4096	SM0~ SM4095
	一般继电器	S	2048	S0~S2047
	定时器	T	512	T0~T511
	计数器	C	512	C0~C511
	32 位计数器	HC	256	HC0~HC255
字符装置	输入继电器	X	64	X0~X63
	输出继电器	Y	64	Y0~Y63
	数据寄存器	D	30000	D0~D29999
		W	30000	W0~W29999 仅供编辑软件使用*4
	特殊数据寄存器	SR	2048	SR0~SR2047
	文件寄存器	FR	65536	FR0~FR65535
	定时器	T	512	T0~T511
	计数器	C	512	C0~C511
	32 位计数器	HC	256（512 words）	HC0~HC255
	变址寄存器	E	10	E0~E9
			5	E10~E14 仅供编辑软件使用*4
常数*1	十进制	K	16 位运算：-32768~32767 32 位运算：-2147483648~2147483647	
常数*2	十六进制	16#	16 位运算：16#0~16#FFFF 32 位运算：16#0~16#FFFFFFFF	
	单精度浮点数	F	32位运算：$\pm1.17549435^{-38}$~$\pm3.40282347^{+38}$	
字符串*3	字符串	“$”	1~31 个字	

*1：十进制的表示方式，在第5、6章的指令装置表中以K来表示，但在ISPSoft中直接输入值，例如K50，请直接输入50。

*2：浮点数的表示方式，在第5、6章的指令装置表中以F来表示，但在ISPSoft中是直接以小数点的方式来表示，例如要输入F500的浮点数，请直接输入500.0。

*3：字符串的表示方式 在第5、6章的指令装置表中以“$”来表示，但在ISPSoft中是以“ ”方式来表示，例如要输入字符串1234，请直接输入“1234”。

*4：仅供编辑软件系统配置使用，因此不做详细说明。

2.1.2 I/O 储存区的基本结构

装置	功能	位存取	字符存取	ISPSoft 修改	强制位状态
X	输入继电器	OK	OK	OK	OK
Y	输出继电器	OK	OK	OK	OK
M	辅助继电器	OK	-	OK	-
SM	特殊辅助标志	OK	-	OK	-
S	一般继电器	OK	-	OK	-
T	定时器	OK	OK	OK	-
C	计数器	OK	OK	OK	-
HC	32 位计数器	OK	OK	OK	-
D	数据寄存器	OK	OK	OK	OK
SR	特殊数据寄存器	-	OK	OK	-
FR	文件寄存器	-	OK*1	-	-
E	变址寄存器	-	OK	OK	-

*1：FR 写入动作需搭配指令执行

2.1.3 停电保持储存方式

PLC 动作 \ 内存类型			非停电保持区		停电保持区	
			Y 装置	其它装置	文件寄存器	其它装置
电源 OFF=>ON			清除	清除	保持	保持
恢复出厂设定值			清除	清除	清除	清除
STOP=>RUN*1	**设定非停电保持区清除**		清除	清除	保持	保持
	设定非停电保持区保持		保持	保持	保持	保持
RUN=>STOP*1	**设定 Y 装置清除**	**SM203=OFF**	清除	保持	保持	保持
	设定 Y 装置保持	**SM203=OFF**	保持	保持	保持	保持
	设定 Y 装置清除	**SM203=ON*3**	清除	清除	保持	保持
	设定 Y 装置保持	**SM203=ON**	保持	清除	保持	保持
SM204=ON 清除非停电保持区*2			清除	清除	保持	保持
SM205=ON 清除停电保持区*2			保持	保持	保持	清除

*1：设定与清除动作请从编辑软件的硬件组态（HWCONFIG）操作；PLC STOP=>RUN默认为**设定非停电保持区清除**；PLC RUN=>STOP默认为**设定Y装置清除**。

*2：该清除时机为SM由OFF=>ON时，当PLC清除完成，系统将自动清除该SM为OFF。

*3：此SM203=ON于RUN=>STOP清除非停电保持区一次，需搭配AS主机韧体版本V1.08.30版(含)以上。

2.1.4 停电保持区的装置范围

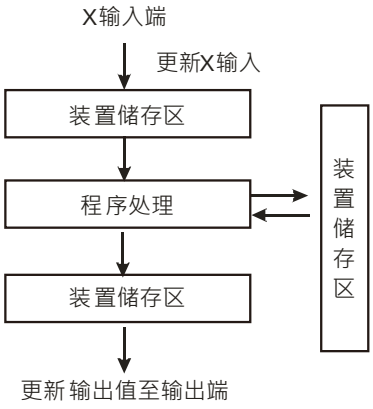
装置	功能	装置范围	停电保持区范围
X	输入继电器	X0~X63	固定非停电保持
Y	输出继电器	Y0~Y63	固定非停电保持
M*1	辅助继电器	M0~M8191	可设定停电保持区范围 默认范围 M6000~M8191
SM	特殊辅助继电器	SM0~SM2047	部分停电保持并且不能被改变 详细内容请参考 SM 菜单
S*1	一般继电器	S0~S1023	可设定停电保持区范围 默认范围 S512~S1023
T	定时器	T0~T511	固定非停电保持
C*1	计数器	C0~C511	可设定停电保持区范围 默认范围 C448~C511
HC*1	32 位计数器	HC0~HC255	可设定停电保持区范围 默认范围 HC128~HC255
D*1	数据寄存器	D0~D29999	可设定停电保持区范围 默认范围 D20000~D23999
		W0~W29999	仅供编辑软件系统配置使用***2**
FR	文件寄存器	FR0~FR65535	固定停电保持
SR	特殊数据寄存器	SR0~SR2047	部分停电保持并且不能被改变 详细内容请参考 SR 菜单
E	变址寄存器	E0~E9	固定非停电保持
		E10~E14	仅供编辑软件系统配置使用***2**

*1：表可利用硬件组态（HWCONFIG）设定停电保持区的范围，范围外的装置就是非停电保持，而设定的范围最大不能超过装置范围。举例说明：设定 M6000~M7000 为停电保持装置，则 M0~M5999 跟 M7001~M8191 都是非停电保持装置。

*2：此装置仅供编辑软件系统配置使用，不做说明。

2.2 装置功能说明

PLC对于程序的处理流程（结束再生方式）

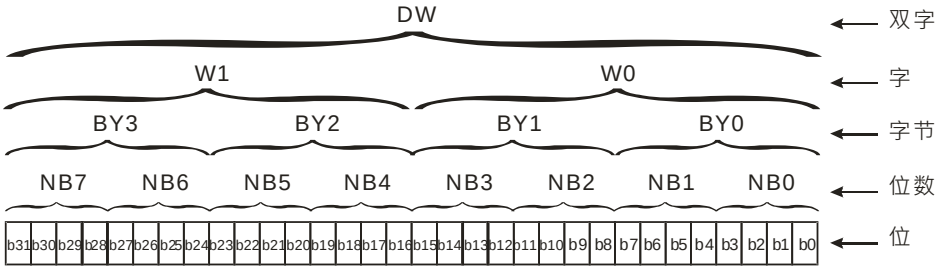


- 更新输入信号：
  1. PLC 在执行程序之前会将外部输入信号状态读入至输入信号储存区内。
  2. 在程序执行中若输入信号作 ON/OFF 变化，但是输入信号储存区内的状态不会改变，一直到下一次扫描开始才会再更新输入信号。
- 程序处理：
  PLC 更新输入信号后，开始从程序的起始地址依序执行程序中的每一指令，其处理结果存入各装置储存区。
- 更新输出状态：
  当执行到 END 指令后将装置储存区内的状态送到用户所分配的输出端。

2.2.1 数值、常数

名称	说明
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十进制数字 0～9 或十六进制之 0～F
字节（Byte）	是由连续之两个位数所组成（亦即 8 位，b7～b0）。可表示十六进制的 00～FF
字符组（Word）	是由连续之两个字节所组成（亦即 16 位，b15～b0）可表示十六进制的 4 个位数值 0000～FFFF
双字符组（Double Word）	是由连续之两个字符组所组成（亦即 32 位，b31～b0），可表示十六进制的 8 个位数值 00000000~FFFFFFFF

二进制系统中位、位数、字节、字符组、及双字符组的关系如下图所示：



PLC 内部依据各种不同控制目的，共使用 4 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）
   PLC 内部的数值运算或储存均采用二进制
2. 十进制（Decimal Number，DEC）
   十进制在 PLC 应用的时机如下：

- 作为定时器 T、计数器 C/HC 等的设定值，例：TMR C0 50。(K 常数)
- S、M、T、C、E...等装置的编号，例：M10、T30。(装置编号)
- X、Y、D...等装置小数点前后的编号，例：X0.0、Y0.11、D10.0。(装置编号)
- Constant K：在应用指令中作为操作数使用，例：MOV 123 D0。(K 常数)

3. BCD (Binary Code Decimal，BCD)

   以一个位数或 4 个位来表示一个十进制的数据，故连续的 16 个位可以表示 4 位数的十进制数值数据。

4. 十六进制 (Hexadecimal Number，HEX)

   十六进制在 PLC 应用的时机如下：

- Constant 16#：在应用指令中作为操作数使用，例：MOV 16#1A2B D0。(十六进制常数)

数值对照表：

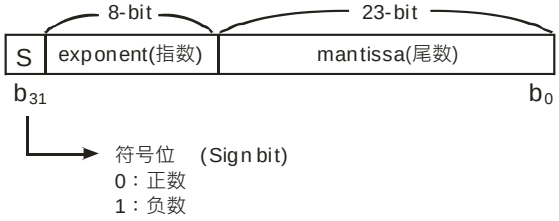
二进制 (BIN)	十进制 (DEC)	BCD (Binary Code Decimal)	十六进制 (HEX)
PLC 内部运算用	常数 K，装置编号	BCD 相关指令	常数 16#，装置编号
0000	0	0000	0
0001	1	0001	1
0010	2	0010	2
0011	3	0011	3
0100	4	0100	4
0101	5	0101	5
0110	6	0110	6
0111	7	0111	7
1000	8	1000	8
1001	9	1001	9
1010	10	-	A
1011	11	-	B
1100	12	-	C
1101	13	-	D
1110	14	-	E
1111	15	-	F
10000	16	0001 0000	10
10001	17	0001 0001	11

2.2.2 浮点数

浮点数的表示方式在 ISPSoft 中是以小数点的方式来表示，例如要输入 500 的浮点数，必须输入 500.0。

2.2.2.1 单精度浮点数 (32 位浮点数)

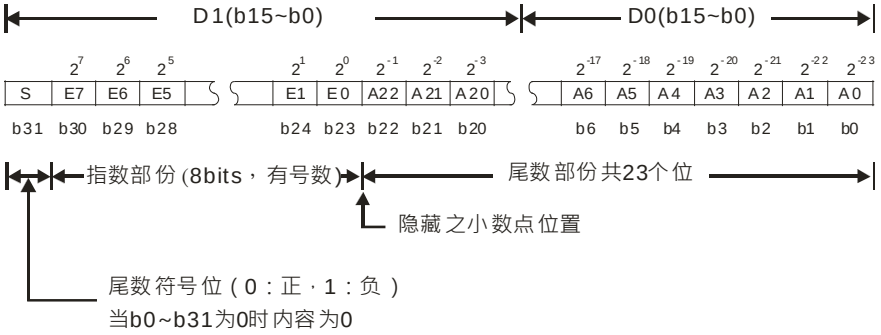
以 32 位的寄存器长度表示浮点数，而表示法是采用 IEEE754 的标准，格式如下：



表达式：$(-1)^S \times 2^{E-B} \times 1.M; B = 127$

因此单精度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$\pm 3.4028\times 10^{+38}$。

AS 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）来说明，如下所示：



范例一：

以单精度浮点数表示 23

步骤一：将 **23** 转换成二进制数字：23.0=10111

步骤二：**将二进制数字正规化**：$10111=1.0111\times 2^4$，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。

$0\ 100\mathbf{0001}1\ 011\mathbf{1000}0000\mathbf{0000}0000\mathbf{0000}_2=41B80000_{16}$

范例二：

以单精度浮点数表示-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。

$1\ 100\mathbf{0001}1\ 011\mathbf{1000}0000\mathbf{0000}0000\mathbf{0000}_2=C1B80000_{16}$

以双精度浮点数表示-23.0

2.2.2.2 十进浮点数

- 单精度浮点数跟双精度浮点数的内容比较无法被人所接受，因此，单精度浮点数跟双精度浮点数可转换成十进浮点数来供人作判断。但是 PLC 对小数点的运算仍旧是使用单精度浮点数跟双精度浮点数。
- 32 位十进浮点数是使用 2 个连续号码的寄存器来表现，较小编号的寄存器号码存放常数部份、较大编号的寄存器号码存放指数部份。

  就以寄存器（D1、D0）来存放一个十进浮点数为例，如下所示。

  $$十进浮点数=[常数D0]*10^{[指数D1]}$$

  底数 D0 = ±1，000~±9，999

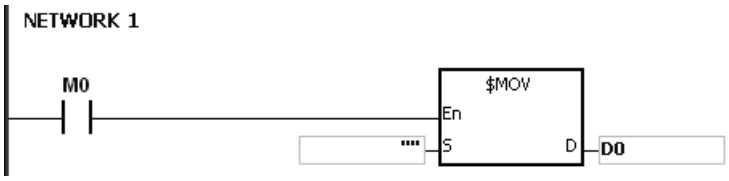
  指数 D1 = - 41~+35

  此外，底数 100 不存在于 D0 的内容，因为，100 是以 1，000×10^{-1} 来表现。32 位十进浮点数的范围为 $\pm 1175\times 10^{-41}$ 到 $\pm 3402\times 10^{+35}$。

2.2.3 字符串

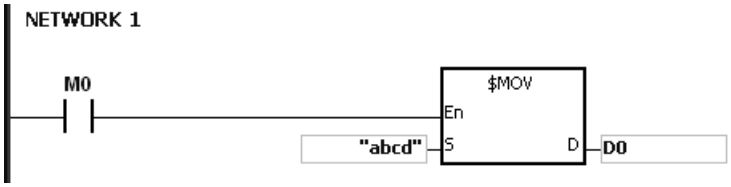
字符串可以处理的是 ASCII 编码的字（*1），一个完整的字符串定义为字符串的起始字符到终止符 NULL 码（16#00）为止。若用户直接输入的是字符串，则最多可以输入 31 个字且 ISPSoft 会自动补上终止符 16#00。若用户输入的是寄存器，则需在结束的后面补上一个终止符 16#00。

1. 字符串搬移 NULL



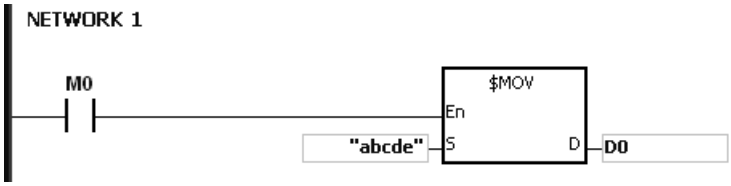
D0=0（NULL）。

2. 字符串为偶数的时候：



D0	16#62（b）	16#61（a）
D1	16#64（d）	16#63（c）
D2	0（NULL）	

3. 字符串为奇数的时候：



D0	16#62（b）	16#61（a）
D1	16#64（d）	16#63（c）
D2	0（NULL）	16#65（e）

*1：ASCII 码转换表

Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
ASCII	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒
Hex	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ASCII	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒
Hex	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
ASCII	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
Hex	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
ASCII	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
Hex	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

ASCII	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Hex	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
ASCII	P	Q	R	S	T	U	V	W	X	Y	Z	☒	☒	☒	☒	☒
Hex	60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F
ASCII	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
Hex	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F
ASCII	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	☒

注：标示为☒，均为不可视字符，请勿设定。

2.2.4 输入继电器 X

- 输入接点 X 的功能：

  输入接点 X 与输入设备（按钮开关，旋钮开关，数字开关等的外部设备）连接，读取输入信号进入 PLC。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 的 ON/OFF 只会跟随输入设备的 ON/OFF 做变化。

- 输入接点的编号（以十进制编号）：

  对 PLC 系列而言，输入端的编号固定从 X0.0 开始算，编号的大小跟随 DIO 模块的输入点数大小而变化，随着与主机的连接顺序来推算出。PLC 机种最大输入点数可达 1024 点，范围如下：X0.0 ~ X63.15。

- 输入的种类：

  输入有刷新输入和直接输入 2 种

  1. 刷新输入：采用程序执行前的外部输入刷新时接收的ON/OFF数据来进行运算的输入方式（如：LD X0.0）
  2. 直接输入：采用指令执行时从外部输入接收的ON/OFF数据进行运算的输入方式（如：LD DX0.0）

2.2.5 输出继电器 Y

- 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（外部信号灯，数字显示器，电磁阀等）。输出接点分成四种，一为继电器（Relay），二为 NPN 型晶体管（Transistor），三为 PNP 型晶体管（Transistor），四为交流硅控器（TRIAC（Thyristors）），每一个输出接点 Y 的 A 或 B 接点在程序中使用次数没有限制，但输出 Y 的编号，在程序建议仅能使用一次，否则依 PLC 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。

- 输出接点的编号（以十进制编号）：

  对 PLC 系列而言，输出端的编号固定从 Y0.0 开始算，编号的多寡跟随 DIO 模块的输出点数大小而变化，随着与主机的连接顺序来推算出。PLC 机种最大输出点数可达 1024 点，范围如下：Y0.0 ~ Y63.15。

  未实际配置使用的 Y 编号可当作一般的装置用。

- 输出的种类：

  输出有刷新输出和直接输出 2 种

  1. 刷新输出：采用程序执行到END指令，依据ON/OFF数据来进行实际输出方式（如：OUT Y0.0）
  2. 直接输出：采用指令执行时，直接依据ON/OFF数据进行实际输出方式（如：OUT DY0.0）

2.2.6 辅助继电器 M

辅助继电器M有A、B接点，而且在程序当中使用次数无限制，用户可利用辅助继电器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二种：

1. 一般用：一般用辅助继电器在PLC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：停电保持用辅助继电器在PLC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。

2.2.7 特殊辅助继电器 SM

每一个特殊辅助继电器均有其特定之功用，未定义的特殊标志请勿使用。

特殊辅助继电器（SM），它的种类及功能如下所示。在编号前有"*"记号可参考SM/SR补充说明，像其中属性栏中标示为"R"者，表示仅可作读取的动作，若标示为"R/W"，表示可作读写的动作。另若标示为"–"，表示无变化。标示为"#"，则表示系统会依照PLC状态作设定，用户可读取该设定值对照手册之说明，可进一步了解系统信息。而其中停电保持一栏标示为"Y"者，表示有停电保持；标示为"N"者，表示不停电保持；标示为"H"者，表示PLC在上电时，以硬件组态（HWCONFIG）设定值执行，PLC程序执行过程中可修改，但不影响原先硬件组态的设定值。

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM0	运算或元件超出范围错误	○	○	OFF	OFF	–	N	R	OFF
SM1	运算或元件超出范围错误锁定	○	○	OFF	OFF	–	N	R	OFF
SM5	指令检查错误	○	○	OFF	OFF	–	N	R	OFF
SM6	停电保持区数据遗失警告标志	○	○	OFF	–	–	N	R/W	OFF
SM7	24V 电源供应不足	○	○	OFF	–	–	N	R	OFF
*SM8	逾时监视错误	○	○	OFF	–	–	N	R	OFF
SM9	系统错误	○	○	OFF	–	–	N	R	OFF
SM10	扩展模块发生错误	○	○	OFF	–	–	N	R	OFF
*SM22	清除错误记录	○	○	OFF	OFF	OFF	N	R/W	OFF
SM23	清除下载记录	○	○	OFF	OFF	OFF	N	R/W	OFF
SM24	清除 PLC 状态变更记录	○	○	OFF	OFF	OFF	N	R/W	OFF
SM25	当在线编辑模式启动时，在线编辑处理标志开启	○	○	OFF	–	–	N	R	OFF
SM26	当除错模式启动时，除错模式处理标志开启	○	○	OFF	–	–	N	R	OFF
*SM27	AS-F2AD/内建 AD 模拟输入报警标志	○	○	OFF	–	–	N	R	OFF
SM28	高速输出指令发生重复指定正在输出点的错误标志	○	○	OFF	OFF	OFF	N	R/W	OFF
SM29	CSFO 输出过程中，输出频率经比率转换超过上限/无输入却持续输出错误标志	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM30	远程模块发生错误标志	○	○	OFF	–	–	N	R	OFF
SM34	密码输入错误	○	○	OFF	–	–	N	R/W	OFF
*SM36	启动回存系统信息至储存卡内。ON 时，依 SR36 数值执行动作。	○	○	OFF	–	–	N	R/W	OFF
SM38	AS-FEN02/FPFN02/FOPC02/FFTP01 功能卡警告/错误标志	○	-	OFF	–	–	N	R/W	OFF
SM76	序列通讯卡 1 送信标志	○	–	OFF	OFF	–	N	R/W	OFF
SM77	序列通讯卡 2 送信标志	○	–	OFF	OFF	–	N	R/W	OFF
SM78	序列通讯卡 1 接收等待	○	–	OFF	OFF	–	N	R	OFF
SM79	序列通讯卡 2 接收等待	○	–	OFF	OFF	–	N	R	OFF
SM80	序列通讯卡 1 接收完毕	○	–	OFF	OFF	–	N	R/W	OFF
SM81	序列通讯卡 2 接收完毕	○	–	OFF	OFF	–	N	R/W	OFF
SM82	序列通讯卡 1 通讯指令接收数据错误	○	–	OFF	OFF	–	N	R	OFF
SM83	序列通讯卡 2 通讯指令接收数据错误	○	–	OFF	OFF	–	N	R	OFF
SM84	序列通讯卡 1 接收逾时	○	–	OFF	OFF	–	N	R/W	OFF
SM85	序列通讯卡 2 接收逾时	○	–	OFF	OFF	–	N	R/W	OFF
SM86	序列通讯卡 1 8/16 位处理模式选择，ON：8 位，OFF：16 位	○	–	OFF	–	–	N	R/W	OFF
SM87	序列通讯卡 2 8/16 位处理模式选择，ON：8 位，OFF：16 位	○	–	OFF	–	–	N	R/W	OFF
SM90	序列通讯卡 1 通讯协议变更用	○	–	OFF	–	–	N	R/W	OFF
SM91	序列通讯卡 2 通讯协议变更用	○	–	OFF	–	–	N	R/W	OFF
SM94	COM1 灯号控制权变更	○	○	–	–	–	H	R/W	OFF
SM95	COM2 灯号控制权变更	○	○	–	–	–	H	R/W	OFF
*SM96	COM1 送信标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM97	COM2 送信标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM98	COM1 接收等待	○	○	OFF	OFF	–	N	R	OFF
*SM99	COM2 接收等待	○	○	OFF	OFF	–	N	R	OFF
*SM100	COM1 接收完毕	○	○	OFF	OFF	–	N	R/W	OFF
*SM101	COM2 接收完毕	○	○	OFF	OFF	–	N	R/W	OFF
*SM102	COM1 通讯指令接收数据错误	○	○	OFF	OFF	–	N	R/W	OFF
*SM103	COM2 通讯指令接收数据错误	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM104	COM1 接收逾时标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM105	COM2 接收逾时标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM106	COM1 8/16 位处理模式选择，ON：8 位，OFF：16 位	○	○	OFF	–	–	N	R/W	OFF
*SM107	COM2 8/16 位处理模式选择，ON：8 位，OFF：16 位	○	○	OFF	–	–	N	R/W	OFF
*SM120	高速比较中断 I200 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM121	高速比较中断 I201 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM122	高速比较中断 I202 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM123	高速比较中断 I203 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM124	高速比较中断 I210 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM125	高速比较中断 I211 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM126	高速比较中断 I212 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM127	高速比较中断 I213 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM128	高速比较中断 I220 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM129	高速比较中断 I221 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM130	高速比较中断 I222 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM131	高速比较中断 I223 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM132	高速比较中断 I230 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM133	高速比较中断 I231 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM134	高速比较中断 I232 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM135	高速比较中断 I233 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM136	高速比较中断 I240 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM137	高速比较中断 I241 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM138	高速比较中断 I242 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM139	高速比较中断 I243 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM140	高速比较中断 I250 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM141	高速比较中断 I251 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM142	高速比较中断 I252 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM143	高速比较中断 I253 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM144	高速比较中断 I260 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM145	高速比较中断 I261 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM146	高速比较中断 I262 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM147	高速比较中断 I263 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM148	高速比较中断 I264 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM149	高速比较中断 I265 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM150	高速比较中断 I266 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM151	高速比较中断 I267 DPLSY 输出参数暂存标志	○	○	OFF	OFF	–	N	R/W	OFF
SM166	VR0 启动标志（需搭配 SR166）	○	○	OFF	–	–	N	R/W	OFF
SM167	VR1 启动标志（需搭配 SR167）	○	○	OFF	–	–	N	R/W	OFF
SM168	功能卡 1 已有连接标志	○	–	–	–	–	N	R	OFF
SM169	功能卡 1 正在通讯或执行中指示标志	○	–	OFF	–	–	N	R	OFF
SM170	功能卡 2 已有连接标志	○	–	–	–	–	N	R	OFF
SM171	功能卡 2 正在通讯或执行中指示标志	○	–	OFF	–	–	N	R	OFF
SM172	AS-FFTP01 功能卡 READY 标志	○	-	OFF	–	–	N	R	OFF
*SM203	Run=>Stop 时，非停电保持区域全部清除一次	○	○	OFF	–	–	N	R/W	OFF
*SM204	非停电保持区域全部清除	○	○	OFF	OFF	OFF	N	R/W	OFF
*SM205	停电保持区域全部清除	○	○	OFF	OFF	OFF	N	R/W	OFF
SM206	输出全部禁止	○	○	OFF	–	–	N	R/W	OFF
*SM209	COM1 通讯协议变更用	○	○	OFF	OFF	OFF	N	R/W	OFF
*SM210	COM1 ASCII/RTU 模式选择，ON 时为 RTU 模式	○	○	–	–	–	H	R/W	OFF
*SM211	COM2 通讯协议变更用	○	○	OFF	OFF	OFF	N	R/W	OFF
*SM212	COM2 ASCII/RTU 模式选择，ON 时为 RTU 模式	○	○	–	–	–	H	R/W	OFF
SM215	启动 PLC RUN	○	○	OFF	ON	OFF	N	R/W	OFF
SM217	万年历读写无回应标志	○	○	OFF	–	–	N	R	OFF
SM218	万年历无法计时警告标志	○	○	OFF	–	–	N	R	OFF
SM219	万年历电池低电压标志	○	○	OFF	–	–	N	R	OFF
*SM220	万年历±30 秒校正	○	○	OFF	OFF	–	N	R/W	OFF
SM221	日光节约时间（D.S.T.）启动标志	○	○	–	–	–	H	R	OFF
SM222	网络 NTP 校正时间成功标志（ON 表示校时成功，OFF 表示失败或无校正动作）	○	○	OFF	–	–	N	R	OFF
*SM230	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 1 台模块	○	○	–	–	–	Y	R/W	OFF
*SM231	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 2 台模块	○	○	–	–	–	Y	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM232	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 3 台模块	○	○	–	–	–	Y	R/W	OFF
*SM233	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 4 台模块	○	○	–	–	–	Y	R/W	OFF
*SM234	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 5 台模块	○	○	–	–	–	Y	R/W	OFF
*SM235	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 6 台模块	○	○	–	–	–	Y	R/W	OFF
*SM236	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 7 台模块	○	○	–	–	–	Y	R/W	OFF
*SM237	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 8 台模块	○	○	–	–	–	Y	R/W	OFF
*SM238	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 9 台模块	○	○	–	–	–	Y	R/W	OFF
*SM239	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 10 台模块	○	○	–	–	–	Y	R/W	OFF
*SM240	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 11 台模块	○	○	–	–	–	Y	R/W	OFF
*SM241	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 12 台模块	○	○	–	–	–	Y	R/W	OFF
*SM242	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 13 台模块	○	○	–	–	–	Y	R/W	OFF
*SM243	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 14 台模块	○	○	–	–	–	Y	R/W	OFF
*SM244	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 15 台模块	○	○	–	–	–	Y	R/W	OFF
*SM245	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 16 台模块	○	○	–	–	–	Y	R/W	OFF
*SM246	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 17 台模块	○	○	–	–	–	Y	R/W	OFF
*SM247	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 18 台模块	○	○	–	–	–	Y	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM248	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 19 台模块	○	○	–	–	–	Y	R/W	OFF
*SM249	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 20 台模块	○	○	–	–	–	Y	R/W	OFF
*SM250	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 21 台模块	○	○	–	–	–	Y	R/W	OFF
*SM251	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 22 台模块	○	○	–	–	–	Y	R/W	OFF
*SM252	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 23 台模块	○	○	–	–	–	Y	R/W	OFF
*SM253	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 24 台模块	○	○	–	–	–	Y	R/W	OFF
*SM254	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 25 台模块	○	○	–	–	–	Y	R/W	OFF
*SM255	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 26 台模块	○	○	–	–	–	Y	R/W	OFF
*SM256	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 27 台模块	○	○	–	–	–	Y	R/W	OFF
*SM257	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 28 台模块	○	○	–	–	–	Y	R/W	OFF
*SM258	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 29 台模块	○	○	–	–	–	Y	R/W	OFF
*SM259	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 30 台模块	○	○	–	–	–	Y	R/W	OFF
*SM260	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 31 台模块	○	○	–	–	–	Y	R/W	OFF
*SM261	当上电时 I/O 模块配置方式选择”手动+标志”时，搭配此标志启动主机右侧第 32 台模块	○	○	–	–	–	Y	R/W	OFF
SM270	第 1 组手摇轮（X0.0/X0.1）输入反向标志	○	○	OFF	OFF	–	N	R/W	OFF
SM271	第 2 组手摇轮（X0.2/X0.3）输入反向标志	○	○	OFF	OFF	–	N	R/W	OFF
SM272	第 3 组手摇轮（X0.4/X0.5）输入反向标志	○	○	OFF	OFF	–	N	R/W	OFF
SM273	第 4 组手摇轮（X0.6/X0.7）输入反向标志	○	○	OFF	OFF	–	N	R/W	OFF
SM274	第 5 组手摇轮（X0.8/X0.9）输入反向标志	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM275	第 6 组手摇轮（X0.10/X0.11）输入反向标志	○	○	OFF	OFF	–	N	R/W	OFF
SM281	高速计数器群组 1 的方向输入反向功能设定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM282	高速计数器群组 2 的方向输入反向功能设定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM283	高速计数器群组 3 的方向输入反向功能设定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM284	高速计数器群组 4 的方向输入反向功能设定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM285	高速计数器群组 5 的方向输入反向功能设定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM286	高速计数器群组 6 的方向输入反向功能设定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM287	高速计数器群组 7 的方向输入反向功能设定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM288	高速计数器群组 8 的方向输入反向功能设定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM291	高速计数器群组 1 的输入点清除功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
SM292	高速计数器群组 2 的输入点清除功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
SM293	高速计数器群组 3 的输入点清除功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
SM294	高速计数器群组 4 的输入点清除功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
SM295	高速计数器群组 5 的输入点清除功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
SM296	高速计数器群组 6 的输入点清除功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
SM300	HC200 计数器的设定上下数标志（OFF 上数、ON 下数）	○	○	OFF	OFF	–	N	R/W	OFF
SM301	HC201 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM302	HC202 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM303	HC203 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM304	HC204 计数器的设定上下数标志（OFF 上数、ON 下数）	○	○	OFF	OFF	–	N	R/W	OFF
SM305	HC205 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM306	HC206 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM307	HC207 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM308	HC208 计数器的设定上下数标志（OFF 上数、ON 下数）	○	○	OFF	OFF	–	N	R/W	OFF
SM309	HC209 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM310	HC210 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM311	HC211 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM312	HC212 计数器的设定上下数标志（OFF 上数、ON 下数）	○	○	OFF	OFF	–	N	R/W	OFF
SM313	HC213 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM314	HC214 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM315	HC215 计数器显示上下数标志（OFF 上数、ON 下数）	○	○	OFF	–	–	N	R	OFF
SM316	HC216 计数器的设定上下数标志（OFF 上数、ON 下数）	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM317	HC217 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM318	HC218 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM319	HC219 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM320	HC220 计数器的设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM321	HC221 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM322	HC222 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM323	HC223 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM332	HC232 计数器的设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM333	HC233 计数器的显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM334	HC234 计数器的显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM335	HC235 计数器的显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM336	HC236 计数器的设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM337	HC237 计数器的显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM338	HC238 计数器的显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM339	HC239 计数器的显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM340	HC240 计数器的设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM341	HC241 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM342	HC242 计数器设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM343	HC243 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM344	HC244 计数器的设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM345	HC245 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM346	HC246 计数器设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM347	HC247 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM348	HC248 计数器设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM349	HC249 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM350	HC250 计数器设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM351	HC251 计数器显示上下数标志 (OFF 上数、ON 下数)	○	○	OFF	–	–	N	R	OFF
SM352	HC252 计数器设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
SM353	HC253 计数器设定上下数标志 (OFF 上数、ON 下数)	○	○	OFF	OFF	–	N	R/W	OFF
*SM360	高速比较中断 I200 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM361	高速比较中断 I201 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM362	高速比较中断 I202 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM363	高速比较中断 I203 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM364	高速比较中断 I210 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM365	高速比较中断 I211 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM366	高速比较中断 I212 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM367	高速比较中断 I213 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM368	高速比较中断 I220 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM369	高速比较中断 I221 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM370	高速比较中断 I222 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM371	高速比较中断 I223 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM372	高速比较中断 I230 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM373	高速比较中断 I231 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM374	高速比较中断 I232 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM375	高速比较中断 I233 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM376	高速比较中断 I240 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM377	高速比较中断 I241 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM378	高速比较中断 I242 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM379	高速比较中断 I243 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM380	高速比较中断 I250 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM381	高速比较中断 I251 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM382	高速比较中断 I252 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM383	高速比较中断 I253 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM384	高速比较中断 I260 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM385	高速比较中断 I261 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM386	高速比较中断 I262 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM387	高速比较中断 I263 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM388	高速比较中断 I264 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM389	高速比较中断 I265 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM390	高速比较中断 I266 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM391	高速比较中断 I267 DPLSY 输出启动标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM400	常开接点	○	○	OFF	ON	OFF	N	R	OFF
*SM401	常闭接点	○	○	OFF	OFF	ON	N	R	OFF
*SM402	开始正向（RUN 的瞬间“ON”）脉冲	○	○	OFF	ON	OFF	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM403	开始反向（RUN 的瞬间“OFF”）脉冲	○	○	OFF	OFF	ON	N	R	OFF
*SM404	10ms 时钟脉冲，5ms ON/5ms OFF	○	○	OFF	–	–	N	R	OFF
*SM405	100ms 时钟脉冲，50ms ON/50ms OFF	○	○	OFF	–	–	N	R	OFF
*SM406	200ms 时钟脉冲，100ms ON/100ms OFF	○	○	OFF	–	–	N	R	OFF
*SM407	1s 时钟脉冲，0.5s ON/0.5s OFF	○	○	OFF	–	–	N	R	OFF
*SM450	储存卡是否存在标志，ON：存在/OFF：不存在	○	○	–	–	–	N	R	OFF
*SM452	储存卡正被存取（Accessed）中，ON：存取中/OFF：无存取	○	○	OFF	–	–	N	R	OFF
*SM453	储存卡读写错误标志	○	○	OFF	–	–	N	R	OFF
SM454	数据记录器启动标志（ON：启动，OFF：关闭）	○	○	OFF	–	–	N	R/W	OFF
SM455	数据记录器取样状态标志（ON：记录笔数已到达取样笔数）	○	○	OFF	–	–	N	R	OFF
SM456	数据记录器与储存卡动作标志。ON 时，依 SR902 数值执行动作。	○	○	OFF	–	–	N	R/W	OFF
SM457	数据记录器参数状态标志（ON：参数已设定）	○	○	–	–	–	N	R	OFF
SM460	Y0.0 / 轴 1（Y0.0/Y0.1）输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM461	Y0.0 / 轴 1（Y0.0/Y0.1）输出执行完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
SM462	轴 1 输出方向脚（Y0.1）反向功能	○	○	OFF	OFF	–	N	R/W	OFF
*SM463	Y0.0 / 轴 1（Y0.0/Y0.1）暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM464	轴 1（Y0.0/Y0.1）硬件正极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM465	轴 1（Y0.0/Y0.1）正极限报警或回原点碰到正极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM466	轴 1（Y0.0/Y0.1）硬件反极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM467	轴 1（Y0.0/Y0.1）负极限报警或回原点碰到负极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM469	轴 1（Y0.0/Y0.1）加减速斜率固定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM470	Y0.0 / 轴 1（Y0.0/Y0.1）输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
SM471	轴 1（Y0.0/Y0.1）输出完成设定中断 I500	○	○	OFF	OFF	–	N	R/W	OFF
SM472	Y0.1 输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM473	Y0.1 输出完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM474	Y0.1 输出暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM475	Y0.1 输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM476	Y0.0 / 轴 1（Y0.0/Y0.1）输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
*SM477	Y0.1 输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM478	Y0.0 / 轴 1（Y0.0/Y0.1）输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM479	Y0.1 输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM480	Y0.2 / 轴 2（Y0.2/Y0.3）输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM481	Y0.2 / 轴 2（Y0.2/Y0.3）输出执行完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
SM482	轴 2 输出方向脚（Y0.3）反向功能	○	○	OFF	OFF	–	N	R/W	OFF
*SM483	Y0.2 / 轴 2（Y0.2/Y0.3）暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM484	轴 2（Y0.2/Y0.3）硬件正极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM485	轴 2（Y0.2/Y0.3）正极限报警或回原点碰到正极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM486	轴 2（Y0.2/Y0.3）硬件反极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM487	轴 2（Y0.2/Y0.3）负极限报警或回原点碰到负极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM489	轴 2（Y0.2/Y0.3）加减速斜率固定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM490	Y0.2 / 轴 2（Y0.2/Y0.3）输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
SM491	轴 2（Y0.2/Y0.3）输出完成设定中断 I501	○	○	OFF	OFF	–	N	R/W	OFF
SM492	Y0.3 输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM493	Y0.3 单点输出完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM494	Y0.3 单点暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM495	Y0.3 输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM496	Y0.2 / 轴 2（Y0.2/Y0.3）输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
*SM497	Y0.3 输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
SM498	Y0.2 / 轴 2（Y0.2/Y0.3）输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM499	Y0.3 输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM500	Y0.4 / 轴 3（Y0.4/Y0.5）输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM501	Y0.4 / 轴 3（Y0.4/Y0.5）输出执行完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
SM502	轴 3 输出方向脚（Y0.5）反向功能	○	○	OFF	OFF	–	N	R/W	OFF
*SM503	Y0.4 / 轴 3（Y0.4/Y0.5）暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM504	轴 3（Y0.4/Y0.5）硬件正极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM505	轴 3（Y0.4/Y0.5）正极限报警或回原点碰到正极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM506	轴 3（Y0.4/Y0.5）硬件反极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM507	轴 3（Y0.4/Y0.5）负极限报警或回原点碰到负极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM509	轴 3（Y0.4/Y0.5）加减速斜率固定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM510	Y0.4 / 轴 3（Y0.4/Y0.5）输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
SM511	轴 3（Y0.4/Y0.5）输出完成设定中断 I502	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM512	Y0.5 输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM513	Y0.5 单点输出完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM514	Y0.5 单点暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM515	Y0.5 输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM516	Y0.4 / 轴 3（Y0.4/Y0.5）输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
*SM517	Y0.5 输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
SM518	Y0.4 / 轴 3（Y0.4/Y0.5）输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM519	Y0.5 输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM520	Y0.6 / 轴 4（Y0.6/Y0.7）输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM521	Y0.6 / 轴 4（Y0.6/Y0.7）输出执行完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
SM522	轴 4 输出方向脚（Y0.7）反向功能	○	○	OFF	OFF	–	N	R/W	OFF
*SM523	Y0.6 / 轴 4（Y0.6/Y0.7）暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM524	轴 4（Y0.6/Y0.7）硬件正极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM525	轴 4（Y0.6/Y0.7）正极限报警或回原点碰到正极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM526	轴 4（Y0.6/Y0.7）硬件反极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM527	轴 4（Y0.6/Y0.7）负极限报警或回原点碰到负极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM529	轴 4（Y0.6/Y0.7）加减速斜率固定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM530	Y0.6 / 轴 4（Y0.6/Y0.7）输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
SM531	轴 4（Y0.6/Y0.7）输出完成设定中断 I503	○	○	OFF	OFF	–	N	R/W	OFF
SM532	Y0.7 输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM533	Y0.7 单点输出完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM534	Y0.7 单点暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM535	Y0.7 输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM536	Y0.6 / 轴 4（Y0.6/Y0.7）输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
*SM537	Y0.7 输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
SM538	Y0.6 / 轴 4（Y0.6/Y0.7）输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM539	Y0.7 输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM540	Y0.8 / 轴 5（Y0.8/Y0.9）输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM541	Y0.8 / 轴 5（Y0.8/Y0.9）输出执行完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
SM542	轴 5 输出方向脚（Y0.9）反向功能	○	○	OFF	OFF	–	N	R/W	OFF
*SM543	Y0.8 / 轴 5（Y0.8/Y0.9）暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM544	轴 5（Y0.8/Y0.9）硬件正极限启动标志	○	○	–	–	–	Y	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM545	轴 5（Y0.8/Y0.9）正极限报警或回原点碰到正极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM546	轴 5（Y0.8/Y0.9）硬件反极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM547	轴 5（Y0.8/Y0.9）负极限报警或回原点碰到负极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM549	轴 5（Y0.8/Y0.9）加减速斜率固定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM550	Y0.8 / 轴 5（Y0.8/Y0.9）输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
SM551	轴 5（Y0.8/Y0.9）输出完成设定中断 I504	○	○	OFF	OFF	–	N	R/W	OFF
SM552	Y0.9 输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM553	Y0.9 单点输出完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM554	Y0.9 单点暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM555	Y0.9 输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM556	Y0.8 / 轴 5（Y0.8/Y0.9）输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
*SM557	Y0.9 输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
SM558	Y0.8 / 轴 5（Y0.8/Y0.9）输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM559	Y0.9 输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM560	Y0.10 / 轴 6（Y0.10/Y0.11）输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM561	Y0.10 / 轴 6（Y0.10/Y0.11）输出执行完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
SM562	轴 6 输出方向脚（Y0.11）反向功能	○	○	OFF	OFF	–	N	R/W	OFF
*SM563	Y0.10 / 轴 6（Y0.10/Y0.11）暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM564	轴 6（Y0.10/Y0.11）硬件正极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM565	轴 6（Y0.10/Y0.11）正极限报警或回原点碰到正极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM566	轴 6（Y0.10/Y0.11）硬件反极限启动标志	○	○	–	–	–	Y	R/W	OFF
SM567	轴 6（Y0.10/Y0.11）负极限报警或回原点碰到负极限标志	○	○	OFF	OFF	–	N	R/W	OFF
SM569	轴 6（Y0.10/Y0.11）加减速斜率固定标志	○	○	OFF	OFF	–	N	R/W	OFF
SM570	Y0.10 / 轴 6（Y0.10/Y0.11）输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF
SM571	轴 6（Y0.10/Y0.11）输出完成设定中断 I505	○	○	OFF	OFF	–	N	R/W	OFF
SM572	Y0.11 输出执行中标志	○	○	OFF	OFF	–	N	R	OFF
SM573	Y0.11 单点输出完毕标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM574	Y0.11 单点暂停输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM575	Y0.11 输出完成自动复位标志	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM576	Y0.10 / 轴 6（Y0.10/Y0.11）输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
*SM577	Y0.11 输出指令关闭时立即停止输出	○	○	OFF	OFF	–	N	R/W	OFF
SM578	Y0.10 / 轴 6（Y0.10/Y0.11）输出执行中变更目标位置	○	○	OFF	OFF	–	N	R/W	OFF
SM579	Y0.11 输出执行中变更目标位置标志	○	○	OFF	OFF	–	N	R/W	OFF
SM580	输出指令全部紧急停止输出标志	○	○	OFF	OFF	–	N	R/W	OFF
SM581	JOG 指令执行中允许变更加减速时间标志	○	○	OFF	OFF	–	N	R/W	OFF
SM585	执行 DTPWS/DTPWL 指令变更定位规划表参数时，依该轴的加减速时间 SR，同步变更定位规划表的加减速时间标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM586	加减速间隔频率精确计算标志（轴 1~轴 6 共享）	○	○	OFF	OFF	–	N	R/W	OFF
*SM587	定位规划表输出指令（TPO）在输出中关闭，其停止输出动作参照各轴立即暂停输出标志（例如：参照 SM476，SM496，..，SM576）	○	○	OFF	OFF	–	N	R/W	OFF
SM591	定位规划表输出指令（TPO），轴 1 暂停切换至下一段/跳转特定规划表编号标志	○	○	OFF	OFF	–	N	R/W	OFF
SM592	定位规划表输出指令（TPO），轴 2 暂停切换至下一段/跳转特定规划表编号标志	○	○	OFF	OFF	–	N	R/W	OFF
SM593	定位规划表输出指令（TPO），轴 3 暂停切换至下一段/跳转特定规划表编号标志	○	○	OFF	OFF	–	N	R/W	OFF
SM594	定位规划表输出指令（TPO），轴 4 暂停切换至下一段/跳转特定规划表编号标志	○	○	OFF	OFF	–	N	R/W	OFF
SM595	定位规划表输出指令（TPO），轴 5 暂停切换至下一段/跳转特定规划表编号标志	○	○	OFF	OFF	–	N	R/W	OFF
SM596	定位规划表输出指令（TPO），轴 6 暂停切换至下一段/跳转特定规划表编号标志	○	○	OFF	OFF	–	N	R/W	OFF
SM600	零标志（Zero flag）	○	○	OFF	–	–	N	R	OFF
SM601	借位标志（Borrow flag）	○	○	OFF	–	–	N	R	OFF
SM602	进位标志（Carry flag）	○	○	OFF	–	–	N	R	OFF
SM603	XCMP 指令待处理对象数超过工位可处理对象数错误标志	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM604	SORT 排序指令工作模式设定（ON–降序排序，OFF–升幂排序）	○	○	OFF	–	–	N	R/W	OFF
SM605	SMOV 工作模式指定	○	○	OFF	–	–	N	R/W	OFF
SM606	8/16 位工作模式选择	○	○	OFF	–	–	N	R/W	OFF
SM607	矩阵比较标志，比较相同值（SM607=1）或不同值（SM607=0）	○	○	OFF	–	–	N	R/W	OFF
SM608	矩阵搜寻结束标志，当比较到最后一个 bit 时，SM608=1	○	○	OFF	–	–	N	R	OFF
SM609	矩阵搜寻起始标志，当 SM609=1 时由第 0 个 bit 开始比较	○	○	OFF	–	–	N	R	OFF
SM610	矩阵位寻找标志，比较到达时立即停止比较动作，SM610=1	○	○	OFF	–	–	N	R	OFF
SM611	矩阵指针错误标志，指针 Pr 值超出范围则 SM611=1	○	○	OFF	–	–	N	R	OFF
SM612	矩阵指针递增标志，将指针目前值+1	○	○	OFF	–	–	N	R/W	OFF
SM613	矩阵指针清除标志，将指针目前值清除为 0	○	○	OFF	–	–	N	R/W	OFF
SM614	矩阵旋转位移输出进位标志	○	○	OFF	–	–	N	R	OFF
SM615	矩阵位移输入补位标志	○	○	OFF	–	–	N	R/W	OFF
SM616	矩阵旋转位移方向标志 SM616=0：左移，SM616=1：右移	○	○	OFF	–	–	N	R/W	OFF
SM617	矩阵计数字元为 0 或位为 1 标志	○	○	OFF	–	–	N	R/W	OFF
SM618	矩阵计数结果为 0 时 ON	○	○	OFF	–	–	N	R/W	OFF
SM619	当 EI 指令被执行时 ON。	○	○	OFF	OFF	–	N	R	OFF
SM620	表格比较指令 CMPT#全部输出标志，若比较结果为全都输出则 SM620=ON	○	○	OFF	–	–	N	R	OFF
SM621	HC0 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM622	HC1 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM623	HC2 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM624	HC3 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM625	HC4 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM626	HC5 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM627	HC6 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM628	HC7 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM629	HC8 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM630	HC9 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM631	HC10 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM632	HC11 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM633	HC12 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM634	HC13 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM635	HC14 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM636	HC15 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM637	HC16 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM638	HC17 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM639	HC18 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM640	HC19 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM641	HC20 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM642	HC21 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM643	HC22 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM644	HC23 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM645	HC24 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM646	HC25 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM647	HC26 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM648	HC27 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM649	HC28 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM650	HC29 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM651	HC30 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM652	HC31 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM653	HC32 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM654	HC33 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM655	HC34 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM656	HC35 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM657	HC36 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM658	HC37 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM659	HC38 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM660	HC39 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF
SM661	HC40 计数模式设定 (ON 时为下数)	○	○	OFF	–	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM662	HC41 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM663	HC42 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM664	HC43 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM665	HC44 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM666	HC45 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM667	HC46 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM668	HC47 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM669	HC48 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM670	HC49 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM671	HC50 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM672	HC51 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM673	HC52 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM674	HC53 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM675	HC54 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM676	HC55 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM677	HC56 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM678	HC57 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM679	HC58 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM680	HC59 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM681	HC60 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM682	HC61 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM683	HC62 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM684	HC63 计数模式设定（ON 时为下数）	○	○	OFF	–	–	N	R/W	OFF
SM685	DSCLP 使用浮点数运算	○	○	OFF	–	–	N	R/W	OFF
SM686	RAMP 指令连续执行设定	○	○	OFF	–	–	N	R/W	OFF
SM687	RAMP 指令执行完毕	○	○	OFF	–	–	N	R/W	OFF
SM688	INCD 指令执行完毕	○	○	OFF	–	–	N	R/W	OFF
SM690	字符串控制模式	○	○	OFF	–	–	N	R/W	OFF
SM691	HKY 输入为 16 位模式（ON：16 进制输入，OFF：A~F 为功能键）	○	○	OFF	–	–	N	R/W	OFF
SM692	HKY 指令执行完毕标志（ON 一个扫描周期）	○	○	OFF	–	–	N	R/W	OFF
SM693	SEGL 指令执行完毕标志（ON 一个扫描周期）	○	○	OFF	–	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM694	DSW 指令执行完毕标志（ON 一个扫描周期）	○	○	OFF	–	–	N	R/W	OFF
SM695	弧度/角度使用标志，ON 的时候表示角度	○	○	OFF	–	–	N	R/W	OFF
SM749	COM1 数据交换表参数错误标志	○	○	OFF	–	–	N	R/W	OFF
*SM750	COM1 数据交换功能的程序启动标志	○	○	OFF	–	OFF	H	R/W	OFF
*SM752	COM1 数据交换联机 1 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM753	COM1 数据交换联机 2 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM754	COM1 数据交换联机 3 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM755	COM1 数据交换联机 4 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM756	COM1 数据交换联机 5 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM757	COM1 数据交换联机 6 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM758	COM1 数据交换联机 7 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM759	COM1 数据交换联机 8 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM760	COM1 数据交换联机 9 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM761	COM1 数据交换联机 10 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM762	COM1 数据交换联机 11 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM763	COM1 数据交换联机 12 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM764	COM1 数据交换联机 13 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM765	COM1 数据交换联机 14 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM766	COM1 数据交换联机 15 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM767	COM1 数据交换联机 16 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM768	COM1 数据交换联机 17 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM769	COM1 数据交换联机 18 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM770	COM1 数据交换联机 19 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM771	COM1 数据交换联机 20 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM772	COM1 数据交换联机 21 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM773	COM1 数据交换联机 22 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM774	COM1 数据交换联机 23 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM775	COM1 数据交换联机 24 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM776	COM1 数据交换联机 25 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM777	COM1 数据交换联机 26 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM778	COM1 数据交换联机 27 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM779	COM1 数据交换联机 28 启动标志	○	○	OFF	–	–	H	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM780	COM1 数据交换联机 29 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM781	COM1 数据交换联机 30 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM782	COM1 数据交换联机 31 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM783	COM1 数据交换联机 32 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM784	COM1 数据交换联机 1 成功标志	○	○	OFF	–	–	N	R	OFF
*SM785	COM1 数据交换联机 2 成功标志	○	○	OFF	–	–	N	R	OFF
*SM786	COM1 数据交换联机 3 成功标志	○	○	OFF	–	–	N	R	OFF
*SM787	COM1 数据交换联机 4 成功标志	○	○	OFF	–	–	N	R	OFF
*SM788	COM1 数据交换联机 5 成功标志	○	○	OFF	–	–	N	R	OFF
*SM789	COM1 数据交换联机 6 成功标志	○	○	OFF	–	–	N	R	OFF
*SM790	COM1 数据交换联机 7 成功标志	○	○	OFF	–	–	N	R	OFF
*SM791	COM1 数据交换联机 8 成功标志	○	○	OFF	–	–	N	R	OFF
*SM792	COM1 数据交换联机 9 成功标志	○	○	OFF	–	–	N	R	OFF
*SM793	COM1 数据交换联机 10 成功标志	○	○	OFF	–	–	N	R	OFF
*SM794	COM1 数据交换联机 11 成功标志	○	○	OFF	–	–	N	R	OFF
*SM795	COM1 数据交换联机 12 成功标志	○	○	OFF	–	–	N	R	OFF
*SM796	COM1 数据交换联机 13 成功标志	○	○	OFF	–	–	N	R	OFF
*SM797	COM1 数据交换联机 14 成功标志	○	○	OFF	–	–	N	R	OFF
*SM798	COM1 数据交换联机 15 成功标志	○	○	OFF	–	–	N	R	OFF
*SM799	COM1 数据交换联机 16 成功标志	○	○	OFF	–	–	N	R	OFF
*SM800	COM1 数据交换联机 17 成功标志	○	○	OFF	–	–	N	R	OFF
*SM801	COM1 数据交换联机 18 成功标志	○	○	OFF	–	–	N	R	OFF
*SM802	COM1 数据交换联机 19 成功标志	○	○	OFF	–	–	N	R	OFF
*SM803	COM1 数据交换联机 20 成功标志	○	○	OFF	–	–	N	R	OFF
*SM804	COM1 数据交换联机 21 成功标志	○	○	OFF	–	–	N	R	OFF
*SM805	COM1 数据交换联机 22 成功标志	○	○	OFF	–	–	N	R	OFF
*SM806	COM1 数据交换联机 23 成功标志	○	○	OFF	–	–	N	R	OFF
*SM807	COM1 数据交换联机 24 成功标志	○	○	OFF	–	–	N	R	OFF
*SM808	COM1 数据交换联机 25 成功标志	○	○	OFF	–	–	N	R	OFF
*SM809	COM1 数据交换联机 26 成功标志	○	○	OFF	–	–	N	R	OFF
*SM810	COM1 数据交换联机 27 成功标志	○	○	OFF	–	–	N	R	OFF
*SM811	COM1 数据交换联机 28 成功标志	○	○	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM812	COM1 数据交换联机 29 成功标志	○	○	OFF	–	–	N	R	OFF
*SM813	COM1 数据交换联机 30 成功标志	○	○	OFF	–	–	N	R	OFF
*SM814	COM1 数据交换联机 31 成功标志	○	○	OFF	–	–	N	R	OFF
*SM815	COM1 数据交换联机 32 成功标志	○	○	OFF	–	–	N	R	OFF
*SM816	COM1 数据交换联机 1 错误标志	○	○	OFF	–	–	N	R	OFF
*SM817	COM1 数据交换联机 2 错误标志	○	○	OFF	–	–	N	R	OFF
*SM818	COM1 数据交换联机 3 错误标志	○	○	OFF	–	–	N	R	OFF
*SM819	COM1 数据交换联机 4 错误标志	○	○	OFF	–	–	N	R	OFF
*SM820	COM1 数据交换联机 5 错误标志	○	○	OFF	–	–	N	R	OFF
*SM821	COM1 数据交换联机 6 错误标志	○	○	OFF	–	–	N	R	OFF
*SM822	COM1 数据交换联机 7 错误标志	○	○	OFF	–	–	N	R	OFF
*SM823	COM1 数据交换联机 8 错误标志	○	○	OFF	–	–	N	R	OFF
*SM824	COM1 数据交换联机 9 错误标志	○	○	OFF	–	–	N	R	OFF
*SM825	COM1 数据交换联机 10 错误标志	○	○	OFF	–	–	N	R	OFF
*SM826	COM1 数据交换联机 11 错误标志	○	○	OFF	–	–	N	R	OFF
*SM827	COM1 数据交换联机 12 错误标志	○	○	OFF	–	–	N	R	OFF
*SM828	COM1 数据交换联机 13 错误标志	○	○	OFF	–	–	N	R	OFF
*SM829	COM1 数据交换联机 14 错误标志	○	○	OFF	–	–	N	R	OFF
*SM830	COM1 数据交换联机 15 错误标志	○	○	OFF	–	–	N	R	OFF
*SM831	COM1 数据交换联机 16 错误标志	○	○	OFF	–	–	N	R	OFF
*SM832	COM1 数据交换联机 17 错误标志	○	○	OFF	–	–	N	R	OFF
*SM833	COM1 数据交换联机 18 错误标志	○	○	OFF	–	–	N	R	OFF
*SM834	COM1 数据交换联机 19 错误标志	○	○	OFF	–	–	N	R	OFF
*SM835	COM1 数据交换联机 20 错误标志	○	○	OFF	–	–	N	R	OFF
*SM836	COM1 数据交换联机 21 错误标志	○	○	OFF	–	–	N	R	OFF
*SM837	COM1 数据交换联机 22 错误标志	○	○	OFF	–	–	N	R	OFF
*SM838	COM1 数据交换联机 23 错误标志	○	○	OFF	–	–	N	R	OFF
*SM839	COM1 数据交换联机 24 错误标志	○	○	OFF	–	–	N	R	OFF
*SM840	COM1 数据交换联机 25 错误标志	○	○	OFF	–	–	N	R	OFF
*SM841	COM1 数据交换联机 26 错误标志	○	○	OFF	–	–	N	R	OFF
*SM842	COM1 数据交换联机 27 错误标志	○	○	OFF	–	–	N	R	OFF
*SM843	COM1 数据交换联机 28 错误标志	○	○	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM844	COM1 数据交换联机 29 错误标志	○	○	OFF	–	–	N	R	OFF
*SM845	COM1 数据交换联机 30 错误标志	○	○	OFF	–	–	N	R	OFF
*SM846	COM1 数据交换联机 31 错误标志	○	○	OFF	–	–	N	R	OFF
*SM847	COM1 数据交换联机 32 错误标志	○	○	OFF	–	–	N	R	OFF
SM861	COM2 数据交换表初始化错误标志	○	○	OFF	–	–	N	R/W	OFF
*SM862	COM2 数据交换功能由程序启动标志	○	○	OFF	–	OFF	H	R/W	OFF
*SM864	COM2 数据交换联机 1 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM865	COM2 数据交换联机 2 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM866	COM2 数据交换联机 3 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM867	COM2 数据交换联机 4 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM868	COM2 数据交换联机 5 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM869	COM2 数据交换联机 6 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM870	COM2 数据交换联机 7 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM871	COM2 数据交换联机 8 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM872	COM2 数据交换联机 9 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM873	COM2 数据交换联机 10 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM874	COM2 数据交换联机 11 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM875	COM2 数据交换联机 12 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM876	COM2 数据交换联机 13 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM877	COM2 数据交换联机 14 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM878	COM2 数据交换联机 15 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM879	COM2 数据交换联机 16 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM880	COM2 数据交换联机 17 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM881	COM2 数据交换联机 18 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM882	COM2 数据交换联机 19 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM883	COM2 数据交换联机 20 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM884	COM2 数据交换联机 21 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM885	COM2 数据交换联机 22 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM886	COM2 数据交换联机 23 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM887	COM2 数据交换联机 24 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM888	COM2 数据交换联机 25 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM889	COM2 数据交换联机 26 启动标志	○	○	OFF	–	–	H	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM890	COM2 数据交换联机 27 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM891	COM2 数据交换联机 28 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM892	COM2 数据交换联机 29 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM893	COM2 数据交换联机 30 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM894	COM2 数据交换联机 31 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM895	COM2 数据交换联机 32 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM896	COM2 数据交换联机 1 成功标志	○	○	OFF	–	–	N	R	OFF
*SM897	COM2 数据交换联机 2 成功标志	○	○	OFF	–	–	N	R	OFF
*SM898	COM2 数据交换联机 3 成功标志	○	○	OFF	–	–	N	R	OFF
*SM899	COM2 数据交换联机 4 成功标志	○	○	OFF	–	–	N	R	OFF
*SM900	COM2 数据交换联机 5 成功标志	○	○	OFF	–	–	N	R	OFF
*SM901	COM2 数据交换联机 6 成功标志	○	○	OFF	–	–	N	R	OFF
*SM902	COM2 数据交换联机 7 成功标志	○	○	OFF	–	–	N	R	OFF
*SM903	COM2 数据交换联机 8 成功标志	○	○	OFF	–	–	N	R	OFF
*SM904	COM2 数据交换联机 9 成功标志	○	○	OFF	–	–	N	R	OFF
*SM905	COM2 数据交换联机 10 成功标志	○	○	OFF	–	–	N	R	OFF
*SM906	COM2 数据交换联机 11 成功标志	○	○	OFF	–	–	N	R	OFF
*SM907	COM2 数据交换联机 12 成功标志	○	○	OFF	–	–	N	R	OFF
*SM908	COM2 数据交换联机 13 成功标志	○	○	OFF	–	–	N	R	OFF
*SM909	COM2 数据交换联机 14 成功标志	○	○	OFF	–	–	N	R	OFF
*SM910	COM2 数据交换联机 15 成功标志	○	○	OFF	–	–	N	R	OFF
*SM911	COM2 数据交换联机 16 成功标志	○	○	OFF	–	–	N	R	OFF
*SM912	COM2 数据交换联机 17 成功标志	○	○	OFF	–	–	N	R	OFF
*SM913	COM2 数据交换联机 18 成功标志	○	○	OFF	–	–	N	R	OFF
*SM914	COM2 数据交换联机 19 成功标志	○	○	OFF	–	–	N	R	OFF
*SM915	COM2 数据交换联机 20 成功标志	○	○	OFF	–	–	N	R	OFF
*SM916	COM2 数据交换联机 21 成功标志	○	○	OFF	–	–	N	R	OFF
*SM917	COM2 数据交换联机 22 成功标志	○	○	OFF	–	–	N	R	OFF
*SM918	COM2 数据交换联机 23 成功标志	○	○	OFF	–	–	N	R	OFF
*SM919	COM2 数据交换联机 24 成功标志	○	○	OFF	–	–	N	R	OFF
*SM920	COM2 数据交换联机 25 成功标志	○	○	OFF	–	–	N	R	OFF
*SM921	COM2 数据交换联机 26 成功标志	○	○	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM922	COM2 数据交换联机 27 成功标志	○	○	OFF	–	–	N	R	OFF
*SM923	COM2 数据交换联机 28 成功标志	○	○	OFF	–	–	N	R	OFF
*SM924	COM2 数据交换联机 29 成功标志	○	○	OFF	–	–	N	R	OFF
*SM925	COM2 数据交换联机 30 成功标志	○	○	OFF	–	–	N	R	OFF
*SM926	COM2 数据交换联机 31 成功标志	○	○	OFF	–	–	N	R	OFF
*SM927	COM2 数据交换联机 32 成功标志	○	○	OFF	–	–	N	R	OFF
*SM928	COM2 数据交换联机 1 错误标志	○	○	OFF	–	–	N	R	OFF
*SM929	COM2 数据交换联机 2 错误标志	○	○	OFF	–	–	N	R	OFF
*SM930	COM2 数据交换联机 3 错误标志	○	○	OFF	–	–	N	R	OFF
*SM931	COM2 数据交换联机 4 错误标志	○	○	OFF	–	–	N	R	OFF
*SM932	COM2 数据交换联机 5 错误标志	○	○	OFF	–	–	N	R	OFF
*SM933	COM2 数据交换联机 6 错误标志	○	○	OFF	–	–	N	R	OFF
*SM934	COM2 数据交换联机 7 错误标志	○	○	OFF	–	–	N	R	OFF
*SM935	COM2 数据交换联机 8 错误标志	○	○	OFF	–	–	N	R	OFF
*SM936	COM2 数据交换联机 9 错误标志	○	○	OFF	–	–	N	R	OFF
*SM937	COM2 数据交换联机 10 错误标志	○	○	OFF	–	–	N	R	OFF
*SM938	COM2 数据交换联机 11 错误标志	○	○	OFF	–	–	N	R	OFF
*SM939	COM2 数据交换联机 12 错误标志	○	○	OFF	–	–	N	R	OFF
*SM940	COM2 数据交换联机 13 错误标志	○	○	OFF	–	–	N	R	OFF
*SM941	COM2 数据交换联机 14 错误标志	○	○	OFF	–	–	N	R	OFF
*SM942	COM2 数据交换联机 15 错误标志	○	○	OFF	–	–	N	R	OFF
*SM943	COM2 数据交换联机 16 错误标志	○	○	OFF	–	–	N	R	OFF
*SM944	COM2 数据交换联机 17 错误标志	○	○	OFF	–	–	N	R	OFF
*SM945	COM2 数据交换联机 18 错误标志	○	○	OFF	–	–	N	R	OFF
*SM946	COM2 数据交换联机 19 错误标志	○	○	OFF	–	–	N	R	OFF
*SM947	COM2 数据交换联机 20 错误标志	○	○	OFF	–	–	N	R	OFF
*SM948	COM2 数据交换联机 21 错误标志	○	○	OFF	–	–	N	R	OFF
*SM949	COM2 数据交换联机 22 错误标志	○	○	OFF	–	–	N	R	OFF
*SM950	COM2 数据交换联机 23 错误标志	○	○	OFF	–	–	N	R	OFF
*SM951	COM2 数据交换联机 24 错误标志	○	○	OFF	–	–	N	R	OFF
*SM952	COM2 数据交换联机 25 错误标志	○	○	OFF	–	–	N	R	OFF
*SM953	COM2 数据交换联机 26 错误标志	○	○	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM954	COM2 数据交换联机 27 错误标志	○	○	OFF	–	–	N	R	OFF
*SM955	COM2 数据交换联机 28 错误标志	○	○	OFF	–	–	N	R	OFF
*SM956	COM2 数据交换联机 29 错误标志	○	○	OFF	–	–	N	R	OFF
*SM957	COM2 数据交换联机 30 错误标志	○	○	OFF	–	–	N	R	OFF
*SM958	COM2 数据交换联机 31 错误标志	○	○	OFF	–	–	N	R	OFF
*SM959	COM2 数据交换联机 32 错误标志	○	○	OFF	–	–	N	R	OFF
SM976	AS 远程模块#1 联机状态标志 （ON：已联机，OFF：断线或未联机）	○	○	OFF	–	–	N	R	OFF
SM977	AS 远程模块#2 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM978	AS 远程模块#3 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM979	AS 远程模块#4 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM980	AS 远程模块#5 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM981	AS 远程模块#6 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM982	AS 远程模块#7 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM983	AS 远程模块#8 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM984	AS 远程模块#9 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM985	AS 远程模块#10 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM986	AS 远程模块#11 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM987	AS 远程模块#12 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM988	AS 远程模块#13 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM989	AS 远程模块#14 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM990	AS 远程模块#15 联机状态标志	○	○	OFF	–	–	N	R	OFF
SM1000	以太网络设定标志。ON 时，将 SR1000~SR1006 的数据写入闪存	○	○	OFF	–	–	N	R/W	OFF
SM1001	以太网络联机状态	○	○	OFF	–	–	N	R	OFF
*SM1006	AS-FEN02 数据交换功能由 PLC 程控标志	○	–	○	○	OFF	–	–	N
*SM1008	AS-FEN02 数据交换联机 1 启动标志	○	–	○	○	OFF	–	–	N
*SM1009	AS-FEN02 数据交换联机 2 启动标志	○	–	○	○	OFF	–	–	N
*SM1010	AS-FEN02 数据交换联机 3 启动标志	○	–	○	○	OFF	–	–	N
*SM1011	AS-FEN02 数据交换联机 4 启动标志	○	–	○	○	OFF	–	–	N
*SM1012	AS-FEN02 数据交换联机 5 启动标志	○	–	○	○	OFF	–	–	N
*SM1013	AS-FEN02 数据交换联机 6 启动标志	○	–	○	○	OFF	–	–	N

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1014	AS-FEN02 数据交换联机 7 启动标志	○	–	○	○	OFF	–	–	N
*SM1015	AS-FEN02 数据交换联机 8 启动标志	○	–	○	○	OFF	–	–	N
*SM1016	AS-FEN02 数据交换联机 1 通讯成功标志	○	–	○	○	OFF	–	–	N
*SM1017	AS-FEN02 数据交换联机 2 通讯成功标志	○	–	○	○	OFF	–	–	N
*SM1018	AS-FEN02 数据交换联机 3 通讯成功标志	○	–	○	○	OFF	–	–	N
*SM1019	AS-FEN02 数据交换联机 4 通讯成功标志	○	–	○	○	OFF	–	–	N
*SM1020	AS-FEN02 数据交换联机 5 通讯成功标志	○	–	○	○	OFF	–	–	N
*SM1021	AS-FEN02 数据交换联机 6 通讯成功标志	○	–	○	○	OFF	–	–	N
*SM1022	AS-FEN02 数据交换联机 7 通讯成功标志	○	–	○	○	OFF	–	–	N
*SM1023	AS-FEN02 数据交换联机 8 通讯成功标志	○	–	○	○	OFF	–	–	N
*SM1024	AS-FEN02 数据交换联机 1 通讯错误标志	○	–	○	○	OFF	–	–	N
*SM1025	AS-FEN02 数据交换联机 2 通讯错误标志	○	–	○	○	OFF	–	–	N
*SM1026	AS-FEN02 数据交换联机 3 通讯错误标志	○	–	○	○	OFF	–	–	N
*SM1027	AS-FEN02 数据交换联机 4 通讯错误标志	○	–	○	○	OFF	–	–	N
*SM1028	AS-FEN02 数据交换联机 5 通讯错误标志	○	–	○	○	OFF	–	–	N
*SM1029	AS-FEN02 数据交换联机 6 通讯错误标志	○	–	○	○	OFF	–	–	N
*SM1030	AS-FEN02 数据交换联机 7 通讯错误标志	○	–	○	○	OFF	–	–	N
*SM1031	AS-FEN02 数据交换联机 8 通讯错误标志	○	–	○	○	OFF	–	–	N
*SM1090	TCP 联机忙碌（Connection Busy）	○	○	OFF	–	–	N	R	OFF
*SM1091	UDP 联机忙碌（Connection Busy）	○	○	OFF	–	–	N	R	OFF
SM1092	MODBUS TCP 从站通讯端口号错误标志，V1.10.20（含）以上支持	○	○	OFF	–	–	N	R/W	OFF
SM1100	以太网络错误标志	○	○	OFF	–	–	N	R	OFF
*SM1106	基本管理–以太网络联机错误（Ethernet Connect Error）	○	○	OFF	–	–	N	R	OFF
*SM1107	以太网络基本管理–基本设定错误（Basic Setting Error）	○	○	OFF	–	–	N	R	OFF
*SM1109	TCP/UDP Socket 基本管理–本地端口已被使用（Local Port Already Used）	○	○	OFF	–	–	N	R	OFF
SM1111	EtherNet/IP 数据 Swap 标志	○	○	OFF	–	–	N	R/W	OFF
*SM1113	Email 发送服务失败	○	○	OFF	–	–	N	R	OFF
*SM1116	Email 发送条件 1 功能启动开关	○	○	OFF	–	–	N	R	OFF
*SM1117	Email 发送条件 1 发送中标志	○	○	OFF	–	–	N	R	OFF
*SM1119	Email 发送条件 1 邮件发送成功	○	○	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1120	Email 发送条件 1 邮件内容有错误，无法发送成功	○	○	OFF	–	–	N	R	OFF
*SM1122	Email 发送条件 1 发送后，SMTP 回复逾时	○	○	OFF	–	–	N	R	OFF
*SM1123	Email 发送条件 1 发送后，SMTP 回复错误	○	○	OFF	–	–	N	R	OFF
*SM1124	Email 发送条件 1 发送时，检查附件大小错误	○	○	OFF	–	–	N	R	OFF
*SM1125	Email 发送条件 1 发送时，找不到附件	○	○	OFF	–	–	N	R	OFF
*SM1126	Email 发送条件 2 功能启动开关	○	○	OFF	–	–	N	R	OFF
*SM1127	Email 发送条件 2 发送中标志	○	○	OFF	–	–	N	R	OFF
*SM1129	Email 发送条件 2 邮件发送成功	○	○	OFF	–	–	N	R	OFF
*SM1130	Email 发送条件 2 邮件内容有错误，无法发送成功	○	○	OFF	–	–	N	R	OFF
*SM1132	Email 发送条件 2 发送后，SMTP 回复逾时	○	○	OFF	–	–	N	R	OFF
*SM1133	Email 发送条件 2 发送后，SMTP 回复错误	○	○	OFF	–	–	N	R	OFF
*SM1134	Email 发送条件 2 发送时，检查附件大小错误	○	○	OFF	–	–	N	R	OFF
*SM1135	Email 发送条件 2 发送时，找不到附件	○	○	OFF	–	–	N	R	OFF
*SM1136	Email 发送条件 3 功能启动开关	○	○	OFF	–	–	N	R	OFF
*SM1137	Email 发送条件 3 发送中标志	○	○	OFF	–	–	N	R	OFF
*SM1139	Email 发送条件 3 邮件发送成功	○	○	OFF	–	–	N	R	OFF
*SM1140	Email 发送条件 3 邮件内容有错误，无法发送成功	○	○	OFF	–	–	N	R	OFF
*SM1142	Email 发送条件 3 发送后，SMTP 回复逾时	○	○	OFF	–	–	N	R	OFF
*SM1143	Email 发送条件 3 发送后，SMTP 回复错误	○	○	OFF	–	–	N	R	OFF
*SM1144	Email 发送条件 3 发送时，检查附件大小错误	○	○	OFF	–	–	N	R	OFF
*SM1145	Email 发送条件 3 发送时，找不到附件	○	○	OFF	–	–	N	R	OFF
*SM1146	Email 发送条件 4 功能启动开关	○	○	OFF	–	–	N	R	OFF
*SM1147	Email 发送条件 4 发送中标志	○	○	OFF	–	–	N	R	OFF
*SM1149	Email 发送条件 4 邮件发送成功	○	○	OFF	–	–	N	R	OFF
*SM1150	Email 发送条件 4 邮件内容有错误，无法发送成功	○	○	OFF	–	–	N	R	OFF
*SM1152	Email 发送条件 4 发送后，SMTP 回复逾时	○	○	OFF	–	–	N	R	OFF
*SM1153	Email 发送条件 4 发送后，SMTP 回复错误	○	○	OFF	–	–	N	R	OFF
*SM1154	Email 发送条件 4 发送时，检查附件大小错误	○	○	OFF	–	–	N	R	OFF
*SM1155	Email 发送条件 4 发送时，找不到附件	○	○	OFF	–	–	N	R	OFF
SM1160	参数还原过程已完成标志 （OFF：当 SR1600 数值由 1 变 0 时，会清除一次为 OFF， ON：当还原完成且全部正常时，会被设为 ON）	○	○	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1161	参数还原过程错误标志 （OFF：当 SR1600 数值由 1 变 0 时，会清除一次为 OFF， ON：当还原过程有错误时，会被设为 ON）	○	○	OFF	–	–	N	R	OFF
*SM1166	Ethernet 数据交换参数错误标志	○	○	–	–	–	N	R	OFF
*SM1167	Ethernet 通讯口数据交换功能启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1168	Ethernet 通讯口数据交换联机 1 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1169	Ethernet 通讯口数据交换联机 2 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1170	Ethernet 通讯口数据交换联机 3 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1171	Ethernet 通讯口数据交换联机 4 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1172	Ethernet 通讯口数据交换联机 5 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1173	Ethernet 通讯口数据交换联机 6 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1174	Ethernet 通讯口数据交换联机 7 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1175	Ethernet 通讯口数据交换联机 8 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1176	Ethernet 通讯口数据交换联机 9 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1177	Ethernet 通讯口数据交换联机 10 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1178	Ethernet 通讯口数据交换联机 11 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1179	Ethernet 通讯口数据交换联机 12 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1180	Ethernet 通讯口数据交换联机 13 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1181	Ethernet 通讯口数据交换联机 14 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1182	Ethernet 通讯口数据交换联机 15 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1183	Ethernet 通讯口数据交换联机 16 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1184	Ethernet 通讯口数据交换联机 17 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1185	Ethernet 通讯口数据交换联机 18 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1186	Ethernet 通讯口数据交换联机 19 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1187	Ethernet 通讯口数据交换联机 20 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1188	Ethernet 通讯口数据交换联机 21 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1189	Ethernet 通讯口数据交换联机 22 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1190	Ethernet 通讯口数据交换联机 23 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1191	Ethernet 通讯口数据交换联机 24 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1192	Ethernet 通讯口数据交换联机 25 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1193	Ethernet 通讯口数据交换联机 26 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1194	Ethernet 通讯口数据交换联机 27 启动标志	○	–	OFF	–	–	H	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1195	Ethernet 通讯口数据交换联机 28 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1196	Ethernet 通讯口数据交换联机 29 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1197	Ethernet 通讯口数据交换联机 30 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1198	Ethernet 通讯口数据交换联机 31 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1199	Ethernet 通讯口数据交换联机 32 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1200	Ethernet 通讯口数据交换联机 1 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1201	Ethernet 通讯口数据交换联机 2 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1202	Ethernet 通讯口数据交换联机 3 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1203	Ethernet 通讯口数据交换联机 4 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1204	Ethernet 通讯口数据交换联机 5 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1205	Ethernet 通讯口数据交换联机 6 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1206	Ethernet 通讯口数据交换联机 7 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1207	Ethernet 通讯口数据交换联机 8 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1208	Ethernet 通讯口数据交换联机 9 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1209	Ethernet 通讯口数据交换联机 10 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1210	Ethernet 通讯口数据交换联机 11 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1211	Ethernet 通讯口数据交换联机 12 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1212	Ethernet 通讯口数据交换联机 13 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1213	Ethernet 通讯口数据交换联机 14 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1214	Ethernet 通讯口数据交换联机 15 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1215	Ethernet 通讯口数据交换联机 16 成功标志	○	○	OFF	–	–	N	R	OFF
*SM1216	Ethernet 通讯口数据交换联机 17 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1217	Ethernet 通讯口数据交换联机 18 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1218	Ethernet 通讯口数据交换联机 19 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1219	Ethernet 通讯口数据交换联机 20 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1220	Ethernet 通讯口数据交换联机 21 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1221	Ethernet 通讯口数据交换联机 22 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1222	Ethernet 通讯口数据交换联机 23 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1223	Ethernet 通讯口数据交换联机 24 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1224	Ethernet 通讯口数据交换联机 25 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1225	Ethernet 通讯口数据交换联机 26 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1226	Ethernet 通讯口数据交换联机 27 成功标志	○	–	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1227	Ethernet 通讯口数据交换联机 28 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1228	Ethernet 通讯口数据交换联机 29 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1229	Ethernet 通讯口数据交换联机 30 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1230	Ethernet 通讯口数据交换联机 31 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1231	Ethernet 通讯口数据交换联机 32 成功标志	○	–	OFF	–	–	N	R	OFF
*SM1232	Ethernet 通讯口数据交换联机 1 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1233	Ethernet 通讯口数据交换联机 2 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1234	Ethernet 通讯口数据交换联机 3 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1235	Ethernet 通讯口数据交换联机 4 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1236	Ethernet 通讯口数据交换联机 5 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1237	Ethernet 通讯口数据交换联机 6 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1238	Ethernet 通讯口数据交换联机 7 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1239	Ethernet 通讯口数据交换联机 8 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1240	Ethernet 通讯口数据交换联机 9 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1241	Ethernet 通讯口数据交换联机 10 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1242	Ethernet 通讯口数据交换联机 11 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1243	Ethernet 通讯口数据交换联机 12 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1244	Ethernet 通讯口数据交换联机 13 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1245	Ethernet 通讯口数据交换联机 14 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1246	Ethernet 通讯口数据交换联机 15 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1247	Ethernet 通讯口数据交换联机 16 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1248	Ethernet 通讯口数据交换联机 17 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1249	Ethernet 通讯口数据交换联机 18 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1250	Ethernet 通讯口数据交换联机 19 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1251	Ethernet 通讯口数据交换联机 20 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1252	Ethernet 通讯口数据交换联机 21 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1253	Ethernet 通讯口数据交换联机 22 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1254	Ethernet 通讯口数据交换联机 23 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1255	Ethernet 通讯口数据交换联机 24 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1256	Ethernet 通讯口数据交换联机 25 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1257	Ethernet 通讯口数据交换联机 26 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1258	Ethernet 通讯口数据交换联机 27 错误标志	○	–	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1259	Ethernet 通讯口数据交换联机 28 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1260	Ethernet 通讯口数据交换联机 29 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1261	Ethernet 通讯口数据交换联机 30 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1262	Ethernet 通讯口数据交换联机 31 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1263	Ethernet 通讯口数据交换联机 32 错误标志	○	–	OFF	–	–	N	R	OFF
SM1269	Socket 配置错误	○	○	OFF	–	–	N	R/W	OFF
SM1270	TCP Socket 1—联机标志	○	○	OFF	–	–	N	R	OFF
SM1271	TCP Socket 1—数据接收完成标志	○	○	OFF	–	–	N	R	OFF
SM1272	TCP Socket 1—数据传送完成标志	○	○	OFF	–	–	N	R	OFF
SM1273	TCP Socket 1—联机建立中标志	○	○	OFF	–	–	N	R	OFF
SM1274	TCP Socket 1—联机关闭中标志	○	○	ON	–	–	Y	R	ON
SM1275	TCP Socket 1—数据传送中标志	○	○	OFF	–	–	N	R	OFF
SM1277	TCP Socket 1—错误标志	○	○	OFF	–	–	N	R	OFF
SM1278	TCP Socket 2—联机标志	○	○	OFF	–	–	N	R	OFF
SM1279	TCP Socket 2—数据接收完成标志	○	○	OFF	–	–	N	R	OFF
SM1280	TCP Socket 2—数据传送完成标志	○	○	OFF	–	–	N	R	OFF
SM1281	TCP Socket 2—联机建立中标志	○	○	OFF	–	–	N	R	OFF
SM1282	TCP Socket 2—联机关闭中标志	○	○	ON	–	–	Y	R	ON
SM1283	TCP Socket 2—数据传送中标志	○	○	OFF	–	–	N	R	OFF
SM1285	TCP Socket 2—错误标志	○	○	OFF	–	–	N	R	OFF
SM1286	TCP Socket 3—联机标志	○	○	OFF	–	–	N	R	OFF
SM1287	TCP Socket 3—数据接收完成标志	○	○	OFF	–	–	N	R	OFF
SM1288	TCP Socket 3—数据传送完成标志	○	○	OFF	–	–	N	R	OFF
SM1289	TCP Socket 3—联机建立中标志	○	○	OFF	–	–	N	R	OFF
SM1290	TCP Socket 3—联机关闭中标志	○	○	ON	–	–	Y	R	ON
SM1291	TCP Socket 3—数据传送中标志	○	○	OFF	–	–	N	R	OFF
SM1293	TCP Socket 3—错误标志	○	○	OFF	–	–	N	R	OFF
SM1294	TCP Socket 4—联机标志	○	○	OFF	–	–	N	R	OFF
SM1295	TCP Socket 4—数据接收完成标志	○	○	OFF	–	–	N	R	OFF
SM1296	TCP Socket 4—数据传送完成标志	○	○	OFF	–	–	N	R	OFF
SM1297	TCP Socket 4—联机建立中标志	○	○	OFF	–	–	N	R	OFF
SM1298	TCP Socket 4—联机关闭中标志	○	○	ON	–	–	Y	R	ON

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1299	TCP Socket 4——数据传送中标志	○	○	OFF	–	–	N	R	OFF
SM1301	TCP Socket 4—错误标志	○	○	OFF	–	–	N	R	OFF
SM1312	RTU-EN01 联机 1 通讯状态标志(On:联机中，Off:未联机或已断线)	○	○	OFF	–	–	N	R	OFF
SM1313	RTU-EN01 联机 2 通讯状态标志	○	○	OFF	–	–	N	R	OFF
SM1314	RTU-EN01 联机 3 通讯状态标志	○	○	OFF	–	–	N	R	OFF
SM1315	RTU-EN01 联机 4 通讯状态标志	○	○	OFF	–	–	N	R	OFF
SM1334	UDP Socket 1—开启标志	○	○	OFF	–	–	N	R	OFF
SM1335	UDP Socket 1—数据接收中标志	○	○	OFF	–	–	N	R	OFF
SM1336	UDP Socket 1—数据传送中标志	○	○	OFF	–	–	N	R	OFF
SM1337	UDP Socket 1—关闭标志	○	○	ON	–	–	Y	R	ON
SM1338	UDP Socket 1—错误标志	○	○	OFF	–	–	N	R	OFF
SM1339	UDP Socket 2—开启标志	○	○	OFF	–	–	N	R	OFF
SM1340	UDP Socket 2—数据接收中标志	○	○	OFF	–	–	N	R	OFF
SM1341	UDP Socket 2—数据传送中标志	○	○	OFF	–	–	N	R	OFF
SM1342	UDP Socket 2—关闭标志	○	○	ON	–	–	Y	R	ON
SM1343	UDP Socket 2—错误标志	○	○	OFF	–	–	N	R	OFF
SM1344	UDP Socket 3—开启标志	○	○	OFF	–	–	N	R	OFF
SM1345	UDP Socket 3—数据接收中标志	○	○	OFF	–	–	N	R	OFF
SM1346	UDP Socket 3—数据传送中标志	○	○	OFF	–	–	N	R	OFF
SM1347	UDP Socket 3—关闭标志	○	○	ON	–	–	Y	R	ON
SM1348	UDP Socket 3—错误标志	○	○	OFF	–	–	N	R	OFF
SM1349	UDP Socket 4—开启标志	○	○	OFF	–	–	N	R	OFF
SM1350	UDP Socket 4—数据接收中标志	○	○	OFF	–	–	N	R	OFF
SM1351	UDP Socket 4—数据传送中标志	○	○	OFF	–	–	N	R	OFF
SM1352	UDP Socket 4—关闭标志	○	○	ON	–	–	Y	R	ON
SM1353	UDP Socket 4—错误标志	○	○	OFF	–	–	N	R	OFF
*SM1375	EtherNet/IP 主站（Scanner）数据交换功能的启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1376	EtherNet/IP 主站（Scanner）数据交换联机 1 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1377	EtherNet/IP 主站（Scanner）数据交换联机 2 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1378	EtherNet/IP 主站（Scanner）数据交换联机 3 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1379	EtherNet/IP 主站（Scanner）数据交换联机 4 启动标志	○	○	OFF	–	–	H	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1380	EtherNet/IP 主站（Scanner）数据交换联机 5 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1381	EtherNet/IP 主站（Scanner）数据交换联机 6 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1382	EtherNet/IP 主站（Scanner）数据交换联机 7 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1383	EtherNet/IP 主站（Scanner）数据交换联机 8 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1384	EtherNet/IP 主站（Scanner）数据交换联机 9 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1385	EtherNet/IP 主站（Scanner）数据交换联机 10 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1386	EtherNet/IP 主站（Scanner）数据交换联机 11 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1387	EtherNet/IP 主站（Scanner）数据交换联机 12 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1388	EtherNet/IP 主站（Scanner）数据交换联机 13 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1389	EtherNet/IP 主站（Scanner）数据交换联机 14 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1390	EtherNet/IP 主站（Scanner）数据交换联机 15 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1391	EtherNet/IP 主站（Scanner）数据交换联机 16 启动标志	○	○	OFF	–	–	H	R/W	OFF
*SM1392	EtherNet/IP 主站（Scanner）数据交换联机 17 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1393	EtherNet/IP 主站（Scanner）数据交换联机 18 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1394	EtherNet/IP 主站（Scanner）数据交换联机 19 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1395	EtherNet/IP 主站（Scanner）数据交换联机 20 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1396	EtherNet/IP 主站（Scanner）数据交换联机 21 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1397	EtherNet/IP 主站（Scanner）数据交换联机 22 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1398	EtherNet/IP 主站（Scanner）数据交换联机 23 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1399	EtherNet/IP 主站（Scanner）数据交换联机 24 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1400	EtherNet/IP 主站（Scanner）数据交换联机 25 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1401	EtherNet/IP 主站（Scanner）数据交换联机 26 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1402	EtherNet/IP 主站（Scanner）数据交换联机 27 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1403	EtherNet/IP 主站（Scanner）数据交换联机 28 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1404	EtherNet/IP 主站（Scanner）数据交换联机 29 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1405	EtherNet/IP 主站（Scanner）数据交换联机 30 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1406	EtherNet/IP 主站（Scanner）数据交换联机 31 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1407	EtherNet/IP 主站（Scanner）数据交换联机 32 启动标志	○	–	OFF	–	–	H	R/W	OFF
*SM1408	EtherNet/IP 主站（Scanner）数据交换联机 1 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1409	EtherNet/IP 主站（Scanner）数据交换联机 2 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1410	EtherNet/IP 主站（Scanner）数据交换联机 3 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1411	EtherNet/IP 主站（Scanner）数据交换联机 4 错误标志	○	○	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1412	EtherNet/IP 主站（Scanner）数据交换联机 5 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1413	EtherNet/IP 主站（Scanner）数据交换联机 6 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1414	EtherNet/IP 主站（Scanner）数据交换联机 7 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1415	EtherNet/IP 主站（Scanner）数据交换联机 8 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1416	EtherNet/IP 主站（Scanner）数据交换联机 9 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1417	EtherNet/IP 主站（Scanner）数据交换联机 10 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1418	EtherNet/IP 主站（Scanner）数据交换联机 11 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1419	EtherNet/IP 主站（Scanner）数据交换联机 12 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1420	EtherNet/IP 主站（Scanner）数据交换联机 13 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1421	EtherNet/IP 主站（Scanner）数据交换联机 14 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1422	EtherNet/IP 主站（Scanner）数据交换联机 15 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1423	EtherNet/IP 主站（Scanner）数据交换联机 16 错误标志	○	○	OFF	–	–	N	R	OFF
*SM1424	EtherNet/IP 主站（Scanner）数据交换联机 17 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1425	EtherNet/IP 主站（Scanner）数据交换联机 18 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1426	EtherNet/IP 主站（Scanner）数据交换联机 19 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1427	EtherNet/IP 主站（Scanner）数据交换联机 20 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1428	EtherNet/IP 主站（Scanner）数据交换联机 21 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1429	EtherNet/IP 主站（Scanner）数据交换联机 22 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1430	EtherNet/IP 主站（Scanner）数据交换联机 23 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1431	EtherNet/IP 主站（Scanner）数据交换联机 24 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1432	EtherNet/IP 主站（Scanner）数据交换联机 25 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1433	EtherNet/IP 主站（Scanner）数据交换联机 26 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1434	EtherNet/IP 主站（Scanner）数据交换联机 27 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1435	EtherNet/IP 主站（Scanner）数据交换联机 28 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1436	EtherNet/IP 主站（Scanner）数据交换联机 29 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1437	EtherNet/IP 主站（Scanner）数据交换联机 30 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1438	EtherNet/IP 主站（Scanner）数据交换联机 31 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1439	EtherNet/IP 主站（Scanner）数据交换联机 32 错误标志	○	–	OFF	–	–	N	R	OFF
*SM1440	EtherNet/IP 从站（Adapter）IO Connection 1 错误标志	○	○	OFF	-	-	N	R	OFF
*SM1441	EtherNet/IP 从站（Adapter）IO Connection 2 错误标志	○	○	OFF	-	-	N	R	OFF
*SM1442	EtherNet/IP 从站（Adapter）IO Connection 3 错误标志	○	○	OFF	-	-	N	R	OFF
*SM1443	EtherNet/IP 从站（Adapter）IO Connection 4 错误标志	○	○	OFF	-	-	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1444	EtherNet/IP 从站（Adapter）IO Connection 5 错误标志	○	○	OFF	-	-	N	R	OFF
*SM1445	EtherNet/IP 从站（Adapter）IO Connection 6 错误标志	○	○	OFF	-	-	N	R	OFF
*SM1446	EtherNet/IP 从站（Adapter）IO Connection 7 错误标志	○	○	OFF	-	-	N	R	OFF
*SM1447	EtherNet/IP 从站（Adapter）IO Connection 8 错误标志	○	○	OFF	-	-	N	R	OFF
*SM1448	EtherNet/IP 从站（Adapter）IO Connection 1 已被联机	○	○	OFF	-	-	N	R	OFF
*SM1449	EtherNet/IP 从站（Adapter）IO Connection 2 已被联机	○	○	OFF	-	-	N	R	OFF
*SM1450	EtherNet/IP 从站（Adapter）IO Connection 3 已被联机	○	○	OFF	-	-	N	R	OFF
*SM1451	EtherNet/IP 从站（Adapter）IO Connection 4 已被联机	○	○	OFF	-	-	N	R	OFF
*SM1452	EtherNet/IP 从站（Adapter）IO Connection 5 已被联机	○	○	OFF	-	-	N	R	OFF
*SM1453	EtherNet/IP 从站（Adapter）IO Connection 6 已被联机	○	○	OFF	-	-	N	R	OFF
*SM1454	EtherNet/IP 从站（Adapter）IO Connection 7 已被联机	○	○	OFF	-	-	N	R	OFF
*SM1455	EtherNet/IP 从站（Adapter）IO Connection 8 已被联机	○	○	OFF	-	-	N	R	OFF
SM1581	台达伺服 CAN 通讯站号 1 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 1 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1582	台达伺服 CAN 通讯站号 2 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 2 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1583	台达伺服 CAN 通讯站号 3 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 3 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1584	台达伺服 CAN 通讯站号 4 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 4 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1585	台达伺服 CAN 通讯站号 5 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 5 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1586	台达伺服 CAN 通讯站号 6 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 6 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1587	台达伺服 CAN 通讯站号 7 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 7 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1588	台达伺服 CAN 通讯站号 8 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 8 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1601	台达变频 CAN 通讯站号 21 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 9 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1602	台达变频 CAN 通讯站号 22 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 10 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1603	台达变频 CAN 通讯站号 23 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 11 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1604	台达变频 CAN 通讯站号 24 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 12 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1605	台达变频 CAN 通讯站号 25 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 13 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1606	台达变频 CAN 通讯站号 26 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 14 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1607	台达变频 CAN 通讯站号 27 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 15 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1608	台达变频 CAN 通讯站号 28 的通讯控制权自动回归标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 16 的通讯控制权自动回归标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1611	台达变频器 CAN 通讯站号 21 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 9 的断线错误标志	○	○	OFF	OFF	–	N	R	OFF
SM1612	台达变频器 CAN 通讯站号 22 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 10 的断线错误标志	○	○	OFF	OFF	–	N	R	OFF
SM1613	台达变频器 CAN 通讯站号 23 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 11 的断线错误标志	○	○	OFF	OFF	–	N	R	OFF
SM1614	台达变频器 CAN 通讯站号 24 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 12 的断线错误标志	○	○	OFF	OFF	–	N	R	OFF
SM1615	台达变频器 CAN 通讯站号 25 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 13 的断线错误标志	○	○	OFF	OFF	–	N	R	OFF
SM1616	台达变频器 CAN 通讯站号 26 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 14 的断线错误标志	○	○	OFF	OFF	–	N	R	OFF
SM1617	台达变频器 CAN 通讯站号 27 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 15 的断线错误标志	○	○	OFF	OFF	–	N	R	OFF
SM1618	台达变频器 CAN 通讯站号 28 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 16 的断线错误标志	○	○	OFF	OFF	–	N	R	OFF
SM1621	台达变频器 CAN 通讯站号 21 启动中标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 9 的伺服 Servo On / 变频启动中标志	○	○	OFF	OFF	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1622	台达变频器 CAN 通讯站号 22 启动中标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 10 的伺服 Servo On / 变频启动中标志	○	○	OFF	OFF	–	N	R	OFF
SM1623	台达变频器 CAN 通讯站号 23 启动中标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 11 的伺服 Servo On / 变频启动中标志	○	○	OFF	OFF	–	N	R	OFF
SM1624	台达变频器 CAN 通讯站号 24 启动中标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 12 的伺服 Servo On / 变频启动中标志	○	○	OFF	OFF	–	N	R	OFF
SM1625	台达变频器 CAN 通讯站号 25 启动中标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 13 的伺服 Servo On / 变频启动中标志	○	○	OFF	OFF	–	N	R	OFF
SM1626	台达变频器 CAN 通讯站号 26 启动中标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 14 的伺服 Servo On / 变频启动中标志	○	○	OFF	OFF	–	N	R	OFF
SM1627	台达变频器 CAN 通讯站号 27 启动中标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 15 的伺服 Servo On / 变频启动中标志	○	○	OFF	OFF	–	N	R	OFF
SM1628	台达变频器 CAN 通讯站号 28 启动中标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 16 的伺服 Servo On / 变频启动中标志	○	○	OFF	OFF	–	N	R	OFF
SM1631	台达伺服 CAN 通讯站号 1 定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 1 的伺服定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1632	台达伺服 CAN 通讯站号 2 定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 2 的伺服定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1633	台达伺服 CAN 通讯站号 3 定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 3 的伺服定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1634	台达伺服 CAN 通讯站号 4 定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 4 的伺服定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1635	台达伺服 CAN 通讯站号 5 定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 5 的伺服定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1636	台达伺服 CAN 通讯站号 6 定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 6 的伺服定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1637	台达伺服 CAN 通讯站号 7 定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 7 的伺服定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1638	台达伺服 CAN 通讯站号 8 定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 8 的伺服定位完成标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1641	台达伺服 CAN 通讯站号 1 暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 1 的伺服暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1642	台达伺服 CAN 通讯站号 2 暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 2 的伺服暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1643	台达伺服 CAN 通讯站号 3 暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 3 的伺服暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1644	台达伺服 CAN 通讯站号 4 暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 4 的伺服暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1645	台达伺服 CAN 通讯站号 5 暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 5 的伺服暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1646	台达伺服 CAN 通讯站号 6 暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 6 的伺服暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1647	台达伺服 CAN 通讯站号 7 暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 7 的伺服暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1648	台达伺服 CAN 通讯站号 8 暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 8 的伺服暂停标志	○	○	OFF	OFF	–	N	R/W	OFF
SM1651	台达伺服 CAN 通讯站号 1 Servo ON 标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 1 的伺服 Servo ON / 变频启动中标志	○	-	OFF	OFF	–	N	R	OFF
SM1652	台达伺服 CAN 通讯站号 2 Servo ON 标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 2 的伺服 Servo ON / 变频启动中标志	○	-	OFF	OFF	–	N	R	OFF
SM1653	台达伺服 CAN 通讯站号 3 Servo ON 标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 3 的伺服 Servo ON / 变频启动中标志	○	-	OFF	OFF	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1654	台达伺服 CAN 通讯站号 4 Servo ON 标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 4 的伺服 Servo ON / 变频启动中标志	○	-	OFF	OFF	–	N	R	OFF
SM1655	台达伺服 CAN 通讯站号 5 Servo ON 标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 5 的伺服 Servo ON / 变频启动中标志	○	-	OFF	OFF	–	N	R	OFF
SM1656	台达伺服 CAN 通讯站号 6 Servo ON 标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 6 的伺服 Servo ON / 变频启动中标志	○	-	OFF	OFF	–	N	R	OFF
SM1657	台达伺服 CAN 通讯站号 7 Servo ON 标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 7 的伺服 Servo ON / 变频启动中标志	○	-	OFF	OFF	–	N	R	OFF
SM1658	台达伺服 CAN 通讯站号 8 Servo ON 标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 8 的伺服 Servo ON / 变频启动中标志	○	-	OFF	OFF	–	N	R	OFF
SM1661	台达伺服 CAN 通讯站号 1 往返功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 1 的伺服往返功能启动标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1662	台达伺服 CAN 通讯站号 2 往返功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 2 的伺服往返功能启动标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1663	台达伺服 CAN 通讯站号 3 往返功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 3 的伺服往返功能启动标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1664	台达伺服 CAN 通讯站号 4 往返功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 4 的伺服往返功能启动标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1665	台达伺服 CAN 通讯站号 5 往返功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 5 的伺服往返功能启动标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1666	台达伺服 CAN 通讯站号 6 往返功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 6 的伺服往返功能启动标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1667	台达伺服 CAN 通讯站号 7 往返功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 7 的伺服往返功能启动标志	○	-	OFF	OFF	–	N	R/W	OFF
SM1668	台达伺服 CAN 通讯站号 8 往返功能启动标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 8 的伺服往返功能启动标志	○	-	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1671	台达伺服 CAN 通讯站号 1 往返方向标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯站号 1 的伺服往返方向标志	○	○	OFF	OFF	–	N	R	OFF
SM1672	台达伺服 CAN 通讯站号 2 往返方向标志	○	○	OFF	OFF	–	N	R	OFF
	动器 ECAT 通讯站号 2 的伺服往返方向标志	○	○	OFF	OFF	–	N	R	OFF
SM1673	台达伺服 CAN 通讯站号 3 往返方向标志	○	○	OFF	OFF	–	N	R	OFF
	动器 ECAT 通讯站号 3 的伺服往返方向标志	○	○	OFF	OFF	–	N	R	OFF
SM1674	台达伺服 CAN 通讯站号 4 往返方向标志	○	○	OFF	OFF	–	N	R	OFF
	动器 ECAT 通讯站号 4 的伺服往返方向标志	○	○	OFF	OFF	–	N	R	OFF
SM1675	台达伺服 CAN 通讯站号 5 往返方向标志	○	○	OFF	OFF	–	N	R	OFF
	动器 ECAT 通讯站号 5 的伺服往返方向标志	○	○	OFF	OFF	–	N	R	OFF
SM1676	台达伺服 CAN 通讯站号 6 往返方向标志	○	○	OFF	OFF	–	N	R	OFF
	动器 ECAT 通讯站号 6 的伺服往返方向标志	○	○	OFF	OFF	–	N	R	OFF
SM1677	台达伺服 CAN 通讯站号 7 往返方向标志	○	○	OFF	OFF	–	N	R	OFF
	动器 ECAT 通讯站号 7 的伺服往返方向标志	○	○	OFF	OFF	–	N	R	OFF
SM1678	台达伺服 CAN 通讯站号 8 往返方向标志	○	○	OFF	OFF	–	N	R	OFF
	动器 ECAT 通讯站号 8 的伺服往返方向标志	○	○	OFF	OFF	–	N	R	OFF
SM1681	台达伺服 CAN 通讯 INITC 与 CASD 完成标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯 CASD 完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1682	台达伺服 CAN 通讯错误标志	○	○	OFF	OFF	–	N	R	OFF
	台达驱动器 ECAT 通讯错误标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1683	台达变频 CAN 通讯 INITC 与 CASD 完成标志	○	○	OFF	OFF	–	N	R	OFF
SM1684	台达驱动器 CAN 通讯控制之连动（OFF）或独立（ON）运作设定	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯控制之连动（OFF）或独立（ON）运作设定	○	–	OFF	OFF	–	N	R/W	OFF
SM1685	启动台达伺服参数 P0-12 自定义数据读取功能	○	○	OFF	OFF	–	N	R/W	OFF
SM1686	OFF：连续站号初始化模式 ON：指定单台初始化模式	○	○	OFF	OFF	–	N	R/W	OFF
SM1691	台达伺服 CAN 通讯站号 1 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 1 断线错误标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1692	台达伺服 CAN 通讯站号 2 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 2 断线错误标志	○	–	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1693	台达伺服 CAN 通讯站号 3 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 3 断线错误标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1694	台达伺服 CAN 通讯站号 4 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 4 断线错误标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1695	台达伺服 CAN 通讯站号 5 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 5 断线错误标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1696	台达伺服 CAN 通讯站号 6 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 6 断线错误标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1697	台达伺服 CAN 通讯站号 7 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 7 断线错误标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1698	台达伺服 CAN 通讯站号 8 的 Heartbeat 错误标志	○	○	OFF	OFF	–	N	R/W	OFF
	台达驱动器 ECAT 通讯站号 8 断线错误标志	○	–	OFF	OFF	–	N	R/W	OFF
*SM1700	CSFO 指令 Y0.0 / Y0.1 轴追随落后补偿标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM1701	CSFO 指令 Y0.2 / Y0.3 轴追随落后补偿标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM1702	CSFO 指令 Y0.4 / Y0.5 轴追随落后补偿标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM1703	CSFO 指令 Y0.6 / Y0.7 轴追随落后补偿标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM1704	CSFO 指令 Y0.8 / Y0.9 轴追随落后补偿标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM1705	CSFO 指令 Y0.10 / Y0.11 轴追随落后补偿标志	○	○	OFF	OFF	–	N	R/W	OFF
*SM1709	序列通讯卡 1 数据交换表初始化错误标志	○	–	OFF	–	–	N	R/W	OFF
*SM1710	序列通讯卡 1 数据交换表由 PLC 程控标志	○	–	OFF	–	OFF	N	R/W	OFF
*SM1712	序列通讯卡 1 数据交换联机 1 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1713	序列通讯卡 1 数据交换联机 2 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1714	序列通讯卡 1 数据交换联机 3 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1715	序列通讯卡 1 数据交换联机 4 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1716	序列通讯卡 1 数据交换联机 5 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1717	序列通讯卡 1 数据交换联机 6 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1718	序列通讯卡 1 数据交换联机 7 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1719	序列通讯卡 1 数据交换联机 8 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1720	序列通讯卡 1 数据交换联机 9 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1721	序列通讯卡 1 数据交换联机 10 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1722	序列通讯卡 1 数据交换联机 11 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1723	序列通讯卡 1 数据交换联机 12 启动标志	○	–	OFF	–	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1724	序列通讯卡 1 数据交换联机 13 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1725	序列通讯卡 1 数据交换联机 14 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1726	序列通讯卡 1 数据交换联机 15 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1727	序列通讯卡 1 数据交换联机 16 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1728	序列通讯卡 1 数据交换联机 17 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1729	序列通讯卡 1 数据交换联机 18 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1730	序列通讯卡 1 数据交换联机 19 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1731	序列通讯卡 1 数据交换联机 20 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1732	序列通讯卡 1 数据交换联机 21 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1733	序列通讯卡 1 数据交换联机 22 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1734	序列通讯卡 1 数据交换联机 23 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1735	序列通讯卡 1 数据交换联机 24 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1736	序列通讯卡 1 数据交换联机 25 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1737	序列通讯卡 1 数据交换联机 26 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1738	序列通讯卡 1 数据交换联机 27 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1739	序列通讯卡 1 数据交换联机 28 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1740	序列通讯卡 1 数据交换联机 29 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1741	序列通讯卡 1 数据交换联机 30 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1742	序列通讯卡 1 数据交换联机 31 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1743	序列通讯卡 1 数据交换联机 32 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1744	序列通讯卡 1 数据交换联机 1 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1745	序列通讯卡 1 数据交换联机 2 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1746	序列通讯卡 1 数据交换联机 3 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1747	序列通讯卡 1 数据交换联机 4 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1748	序列通讯卡 1 数据交换联机 5 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1749	序列通讯卡 1 数据交换联机 6 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1750	序列通讯卡 1 数据交换联机 7 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1751	序列通讯卡 1 数据交换联机 8 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1752	序列通讯卡 1 数据交换联机 9 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1753	序列通讯卡 1 数据交换联机 10 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1754	序列通讯卡 1 数据交换联机 11 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1755	序列通讯卡 1 数据交换联机 12 通讯成功标志	○	–	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1756	序列通讯卡 1 数据交换联机 13 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1757	序列通讯卡 1 数据交换联机 14 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1758	序列通讯卡 1 数据交换联机 15 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1759	序列通讯卡 1 数据交换联机 16 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1760	序列通讯卡 1 数据交换联机 17 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1761	序列通讯卡 1 数据交换联机 18 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1762	序列通讯卡 1 数据交换联机 19 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1763	序列通讯卡 1 数据交换联机 20 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1764	序列通讯卡 1 数据交换联机 21 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1765	序列通讯卡 1 数据交换联机 22 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1766	序列通讯卡 1 数据交换联机 23 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1767	序列通讯卡 1 数据交换联机 24 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1768	序列通讯卡 1 数据交换联机 25 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1769	序列通讯卡 1 数据交换联机 26 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1770	序列通讯卡 1 数据交换联机 27 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1771	序列通讯卡 1 数据交换联机 28 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1772	序列通讯卡 1 数据交换联机 29 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1773	序列通讯卡 1 数据交换联机 30 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1774	序列通讯卡 1 数据交换联机 31 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1775	序列通讯卡 1 数据交换联机 32 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1776	序列通讯卡 1 数据交换联机 1 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1777	序列通讯卡 1 数据交换联机 2 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1778	序列通讯卡 1 数据交换联机 3 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1779	序列通讯卡 1 数据交换联机 4 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1780	序列通讯卡 1 数据交换联机 5 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1781	序列通讯卡 1 数据交换联机 6 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1782	序列通讯卡 1 数据交换联机 7 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1783	序列通讯卡 1 数据交换联机 8 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1784	序列通讯卡 1 数据交换联机 9 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1785	序列通讯卡 1 数据交换联机 10 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1786	序列通讯卡 1 数据交换联机 11 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1787	序列通讯卡 1 数据交换联机 12 通讯错误标志	○	–	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1788	序列通讯卡 1 数据交换联机 13 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1789	序列通讯卡 1 数据交换联机 14 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1790	序列通讯卡 1 数据交换联机 15 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1791	序列通讯卡 1 数据交换联机 16 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1792	序列通讯卡 1 数据交换联机 17 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1793	序列通讯卡 1 数据交换联机 18 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1794	序列通讯卡 1 数据交换联机 19 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1795	序列通讯卡 1 数据交换联机 20 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1796	序列通讯卡 1 数据交换联机 21 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1797	序列通讯卡 1 数据交换联机 22 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1798	序列通讯卡 1 数据交换联机 23 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1799	序列通讯卡 1 数据交换联机 24 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1800	序列通讯卡 1 数据交换联机 25 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1801	序列通讯卡 1 数据交换联机 26 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1802	序列通讯卡 1 数据交换联机 27 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1803	序列通讯卡 1 数据交换联机 28 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1804	序列通讯卡 1 数据交换联机 29 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1805	序列通讯卡 1 数据交换联机 30 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1806	序列通讯卡 1 数据交换联机 31 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1807	序列通讯卡 1 数据交换联机 32 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1821	序列通讯卡 2 数据交换表初始化错误标志	○	–	OFF	–	–	N	R/W	OFF
*SM1822	序列通讯卡 2 数据交换表由 PLC 程控标志	○	–	OFF	–	OFF	N	R/W	OFF
*SM1824	序列通讯卡 2 数据交换联机 1 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1825	序列通讯卡 2 数据交换联机 2 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1826	序列通讯卡 2 数据交换联机 3 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1827	序列通讯卡 2 数据交换联机 4 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1828	序列通讯卡 2 数据交换联机 5 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1829	序列通讯卡 2 数据交换联机 6 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1830	序列通讯卡 2 数据交换联机 7 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1831	序列通讯卡 2 数据交换联机 8 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1832	序列通讯卡 2 数据交换联机 9 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1833	序列通讯卡 2 数据交换联机 10 启动标志	○	–	OFF	–	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1834	序列通讯卡 2 数据交换联机 11 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1835	序列通讯卡 2 数据交换联机 12 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1836	序列通讯卡 2 数据交换联机 13 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1837	序列通讯卡 2 数据交换联机 14 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1838	序列通讯卡 2 数据交换联机 15 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1839	序列通讯卡 2 数据交换联机 16 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1840	序列通讯卡 2 数据交换联机 17 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1841	序列通讯卡 2 数据交换联机 18 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1842	序列通讯卡 2 数据交换联机 19 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1843	序列通讯卡 2 数据交换联机 20 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1844	序列通讯卡 2 数据交换联机 21 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1845	序列通讯卡 2 数据交换联机 22 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1846	序列通讯卡 2 数据交换联机 23 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1847	序列通讯卡 2 数据交换联机 24 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1848	序列通讯卡 2 数据交换联机 25 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1849	序列通讯卡 2 数据交换联机 26 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1850	序列通讯卡 2 数据交换联机 27 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1851	序列通讯卡 2 数据交换联机 28 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1852	序列通讯卡 2 数据交换联机 29 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1853	序列通讯卡 2 数据交换联机 30 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1854	序列通讯卡 2 数据交换联机 31 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1855	序列通讯卡 2 数据交换联机 32 启动标志	○	–	OFF	–	–	N	R/W	OFF
*SM1856	序列通讯卡 2 数据交换联机 1 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1857	序列通讯卡 2 数据交换联机 2 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1858	序列通讯卡 2 数据交换联机 3 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1859	序列通讯卡 2 数据交换联机 4 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1860	序列通讯卡 2 数据交换联机 5 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1861	序列通讯卡 2 数据交换联机 6 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1862	序列通讯卡 2 数据交换联机 7 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1863	序列通讯卡 2 数据交换联机 8 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1864	序列通讯卡 2 数据交换联机 9 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1865	序列通讯卡 2 数据交换联机 10 通讯成功标志	○	–	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1866	序列通讯卡 2 数据交换联机 11 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1867	序列通讯卡 2 数据交换联机 12 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1868	序列通讯卡 2 数据交换联机 13 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1869	序列通讯卡 2 数据交换联机 14 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1870	序列通讯卡 2 数据交换联机 15 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1871	序列通讯卡 2 数据交换联机 16 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1872	序列通讯卡 2 数据交换联机 17 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1873	序列通讯卡 2 数据交换联机 18 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1874	序列通讯卡 2 数据交换联机 19 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1875	序列通讯卡 2 数据交换联机 20 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1876	序列通讯卡 2 数据交换联机 21 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1877	序列通讯卡 2 数据交换联机 22 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1878	序列通讯卡 2 数据交换联机 23 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1879	序列通讯卡 2 数据交换联机 24 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1880	序列通讯卡 2 数据交换联机 25 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1881	序列通讯卡 2 数据交换联机 26 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1882	序列通讯卡 2 数据交换联机 27 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1883	序列通讯卡 2 数据交换联机 28 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1884	序列通讯卡 2 数据交换联机 29 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1885	序列通讯卡 2 数据交换联机 30 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1886	序列通讯卡 2 数据交换联机 31 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1887	序列通讯卡 2 数据交换联机 32 通讯成功标志	○	–	OFF	–	–	N	R	OFF
*SM1888	序列通讯卡 2 数据交换联机 1 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1889	序列通讯卡 2 数据交换联机 2 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1890	序列通讯卡 2 数据交换联机 3 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1891	序列通讯卡 2 数据交换联机 4 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1892	序列通讯卡 2 数据交换联机 5 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1893	序列通讯卡 2 数据交换联机 6 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1894	序列通讯卡 2 数据交换联机 7 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1895	序列通讯卡 2 数据交换联机 8 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1896	序列通讯卡 2 数据交换联机 9 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1897	序列通讯卡 2 数据交换联机 10 通讯错误标志	○	–	OFF	–	–	N	R	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SM1898	序列通讯卡 2 数据交换联机 11 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1899	序列通讯卡 2 数据交换联机 12 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1900	序列通讯卡 2 数据交换联机 13 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1901	序列通讯卡 2 数据交换联机 14 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1902	序列通讯卡 2 数据交换联机 15 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1903	序列通讯卡 2 数据交换联机 16 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1904	序列通讯卡 2 数据交换联机 17 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1905	序列通讯卡 2 数据交换联机 18 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1906	序列通讯卡 2 数据交换联机 19 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1907	序列通讯卡 2 数据交换联机 20 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1908	序列通讯卡 2 数据交换联机 21 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1909	序列通讯卡 2 数据交换联机 22 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1910	序列通讯卡 2 数据交换联机 23 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1911	序列通讯卡 2 数据交换联机 24 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1912	序列通讯卡 2 数据交换联机 25 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1913	序列通讯卡 2 数据交换联机 26 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1914	序列通讯卡 2 数据交换联机 27 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1915	序列通讯卡 2 数据交换联机 28 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1916	序列通讯卡 2 数据交换联机 29 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1917	序列通讯卡 2 数据交换联机 30 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1918	序列通讯卡 2 数据交换联机 31 通讯错误标志	○	–	OFF	–	–	N	R	OFF
*SM1919	序列通讯卡 2 数据交换联机 32 通讯错误标志	○	–	OFF	–	–	N	R	OFF
SM1921	台达驱动器 ECAT 通讯站号 9 的伺服定位完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1922	台达驱动器 ECAT 通讯站号 10 的伺服定位完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1923	台达驱动器 ECAT 通讯站号 11 的伺服定位完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1924	台达驱动器 ECAT 通讯站号 12 的伺服定位完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1925	台达驱动器 ECAT 通讯站号 13 的伺服定位完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1926	台达驱动器 ECAT 通讯站号 14 的伺服定位完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1927	台达驱动器 ECAT 通讯站号 15 的伺服定位完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1928	台达驱动器 ECAT 通讯站号 16 的伺服定位完成标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1931	台达驱动器 ECAT 通讯站号 9 的伺服暂停标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1932	台达驱动器 ECAT 通讯站号 10 的伺服暂停标志	○	–	OFF	OFF	–	N	R/W	OFF

SM	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SM1933	台达驱动器 ECAT 通讯站号 11 的伺服暂停标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1934	台达驱动器 ECAT 通讯站号 12 的伺服暂停标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1935	台达驱动器 ECAT 通讯站号 13 的伺服暂停标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1936	台达驱动器 ECAT 通讯站号 14 的伺服暂停标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1937	台达驱动器 ECAT 通讯站号 15 的伺服暂停标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1938	台达驱动器 ECAT 通讯站号 16 的伺服暂停标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1941	台达驱动器 ECAT 通讯站号 9 的伺服往返功能启动标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1942	台达驱动器 ECAT 通讯站号 10 的伺服往返功能启动标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1943	台达驱动器 ECAT 通讯站号 11 的伺服往返功能启动标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1944	台达驱动器 ECAT 通讯站号 12 的伺服往返功能启动标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1945	台达驱动器 ECAT 通讯站号 13 的伺服往返功能启动标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1946	台达驱动器 ECAT 通讯站号 14 的伺服往返功能启动标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1947	台达驱动器 ECAT 通讯站号 15 的伺服往返功能启动标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1948	台达驱动器 ECAT 通讯站号 16 的伺服往返功能启动标志	○	–	OFF	OFF	–	N	R/W	OFF
SM1951	台达驱动器 ECAT 通讯站号 9 的伺服往返方向标志	○	–	OFF	OFF	–	N	R	OFF
SM1952	台达驱动器 ECAT 通讯站号 10 的伺服往返方向标志	○	–	OFF	OFF	–	N	R	OFF
SM1953	台达驱动器 ECAT 通讯站号 11 的伺服往返方向标志	○	–	OFF	OFF	–	N	R	OFF
SM1954	台达驱动器 ECAT 通讯站号 12 的伺服往返方向标志	○	–	OFF	OFF	–	N	R	OFF
SM1955	台达驱动器 ECAT 通讯站号 13 的伺服往返方向标志	○	–	OFF	OFF	–	N	R	OFF
SM1956	台达驱动器 ECAT 通讯站号 14 的伺服往返方向标志	○	–	OFF	OFF	–	N	R	OFF
SM1957	台达驱动器 ECAT 通讯站号 15 的伺服往返方向标志	○	–	OFF	OFF	–	N	R	OFF
SM1958	台达驱动器 ECAT 通讯站号 16 的伺服往返方向标志	○	–	OFF	OFF	–	N	R	OFF

注 1：有*的 SM 请参考 SM/SR 补充说明。

注 2：当 SM 的电源 OFF→ON 状态为"–"，且停电保持区为"N"时，即表示由硬件组态（HWCONFIG）设定的参数决定。

注 3：SM 中所述的「序列通讯卡」系指 AS-F232、AS-F422 及 AS-F485。

2.2.8 特殊辅助继电器 SM 的更新时间

特殊辅助继电器	状态与更新时机说明
SM0~SM1	系统自动设定与清除–程序执行有误自动设 ON
SM5	系统自动设定与清除–程序写入 PLC 有误自动设 ON
SM6	上电时检查停电保持区数据是否遗失，若有遗失自动设 ON，需用户自行清除
SM7	24V 电源供应不足时自动设 ON，需用户自行清除
SM8	系统设定，用户清除 PLC 程序扫描逾时（WDT）发生设为 ON，重新下载 PLC 程序或重新上电后 OFF
SM9	系统自行检查有误自动设为 ON，系统自动清除
SM10	扩展模块发生错误自动设为 ON，系统自动清除
SM22~SM24	用户设定 ON 通知系统执行清除，清除完成后自动设为 OFF
SM25~SM26	用户操作编辑软件进入自动设 ON；退出线上编辑自动设 OFF
SM27	更新 AS-F2AD/内建 AD 模拟数值，由系统设定与清除
SM28	系统自行检查有误自动设为 ON，由用户清除
SM29	系统设定/用户清除
SM30	远程模块发生错误自动设为 ON，系统自动清除
SM34	输入密码错误自动设为 ON，系统自动清除
SM36	用户设定 ON 通知系统执行回存系统信息至储存卡功能，执行完毕自动 OFF
SM38	系统设定/用户清除
SM76~SM77	用户设定 ON 通知系统执行通讯命令，传送完毕后系统自动清除
SM78~SM79	通讯等待中会自动 ON，通讯接收完毕自动 OFF
SM80~SM81	通讯接收完毕自动设 ON，由用户清除
SM82~SM83	通讯响应错误自动设 ON，由用户清除
SM84~SM85	通讯接收逾时时自动设 ON，由用户清除
SM86~SM87	用户设定 ON 表 8 位模式，OFF 表 16 位模式，由用户清除
SM90~SM91	用户设定 ON 后会变更通讯，完成后系统自动清除
SM94~SM95	上电时 HWCONFIG 决定，之后用户可变更
SM96~SM97	用户设定 ON 通知系统执行通讯命令，传送完毕后系统自动清除
SM98~SM99	通讯等待中会自动 ON，通讯接收完毕自动 OFF
SM100~SM101	通讯接收完毕自动设 ON，由用户清除
SM102~SM103	通讯响应错误自动设 ON，由用户清除
SM104~SM105	通讯接收逾时时自动设 ON，由用户清除
SM106~SM107	用户设定 ON 表 8 位模式，OFF 表 16 位模式，由用户清除
SM120~SM151	用户设定，执行 DPLSY 指令时由系统清除
SM166~SM167	用户设定与清除
SM168~SM171	系统设定与清除
SM172	系统设定与清除

特殊辅助继电器	状态与更新时机说明
SM203	用户设定 ON 后，PLC RUN=>STOP 时清除完毕，系统自动 OFF
SM204~SM205	用户设定 ON 后会系统执行清除动作，清除完毕系统自动 OFF
SM206	用户设定 ON 表示输出全部禁止，由用户清除
SM209	用户设定 ON 表执行 COM1 通讯协议变更，完成后系统自动清除
SM210	用户设定 ON 表 COM1 为 RTU 模式，OFF 表 ASCII 模式
SM211	用户设定 ON 表执行 COM2 通讯协议变更，完成后系统自动清除
SM212	用户设定 ON 表 COM2 为 RTU 模式，OFF 表 ASCII 模式
SM215	用户设定 ON 表示 PLC RUN，OFF 表示 PLC STOP
SM217	系统设定
SM218	上电检查万年历是否正常计时，若无法计时自动设 ON，由用户清除
SM219	随时监看万年历电池电压，电压不足时自动设 ON，系统自动清除
SM220	用户设定 ON 后，万年历会执行±30 秒校正，由用户清除
SM221	根据 HWCONFIG 做更新，或 API1607 DST 指令被执行时会自动更新，ON 表示 DST 功能启动中
SM222	系统设定与清除
SM230~SM261	HWCONFIG 下载时更新，用户可自行设定。
SM270 ~ SM275	执行 CSFO 指令时用户设定 ON 表示启动输入反向，OFF 表示关闭输入反向
SM281~SM288	用户设定 ON 表示下数，OFF 表示上数
SM291~SM296	用户设定 ON 启动输入点清除功能，OFF 表示关闭
SM300	用户设定 ON 表示下数，OFF 表示上数
SM301~SM303	系统自动显示 ON 表示下数，OFF 表示上数
SM304	用户设定 ON 表示下数，OFF 表示上数
SM305~SM307	系统自动显示 ON 表示下数，OFF 表示上数
SM308	用户设定 ON 表示下数，OFF 表示上数
SM309~SM311	系统自动显示 ON 表示下数，OFF 表示上数
SM312	用户设定 ON 表示下数，OFF 表示上数
SM313~SM315	系统自动显示 ON 表示下数，OFF 表示上数
SM316	用户设定 ON 表示下数，OFF 表示上数
SM317~SM319	系统自动显示 ON 表示下数，OFF 表示上数
SM320	用户设定 ON 表示下数，OFF 表示上数
SM321~SM323	系统自动显示 ON 表示下数，OFF 表示上数
SM332	用户设定 ON 表示下数，OFF 表示上数
SM333~SM335	系统自动显示 ON 表示下数，OFF 表示上数
SM336	用户设定 ON 表示下数，OFF 表示上数
SM337~SM339	系统自动显示 ON 表示下数，OFF 表示上数
SM340	用户设定 ON 表示下数，OFF 表示上数
SM341	系统自动显示 ON 表示下数，OFF 表示上数

特殊辅助继电器	状态与更新时机说明
SM342	用户设定 ON 表示下数，OFF 表示上数
SM343	系统自动显示 ON 表示下数，OFF 表示上数
SM344	用户设定 ON 表示下数，OFF 表示上数
SM345	系统自动显示 ON 表示下数，OFF 表示上数
SM346	用户设定 ON 表示下数，OFF 表示上数
SM347	系统自动显示 ON 表示下数，OFF 表示上数
SM348	用户设定 ON 表示下数，OFF 表示上数
SM349	系统自动显示 ON 表示下数，OFF 表示上数
SM350	用户设定 ON 表示下数，OFF 表示上数
SM351	系统自动显示 ON 表示下数，OFF 表示上数
SM352~SM353	用户设定 ON 表示下数，OFF 表示上数
SM360~SM391	用户设定与清除
SM400~SM403	PLC RUN 时，系统自动每一扫描周期更新一次
SM404	系统自动每 5ms 更新一次设定或清除
SM405	系统自动每 50ms 更新一次设定或清除
SM406	系统自动每 100ms 更新一次设定或清除
SM407	系统自动每 0.5s 更新一次设定或清除
SM450	储存卡被插入主机时系统自动设定 ON，移除时自动 OFF
SM452~SM453	系统自动设定与清除
SM454	用户设定与清除
SM455	系统自动显示
SM456	用户设定 ON 执行储存，完成后系统自动清除
SM457	系统自动显示
SM460	系统自动显示
SM461	系统显示/用户清除
SM462~SM464	用户清除与设定
SM465	系统显示/用户清除
SM466	用户清除与设定
SM467	系统显示/用户清除
SM469~SM471	用户清除与设定
SM472	系统自动显示
SM473	系统显示/用户清除
SM474~SM477	用户清除与设定
SM478~SM479	用户设定，系统执行完成后清除
SM480	系统自动显示
SM481	系统显示/用户清除
SM482~SM484	用户清除与设定

特殊辅助继电器	状态与更新时机说明
SM485	系统显示/用户清除
SM486	用户清除与设定
SM487	系统显示/用户清除
SM489~SM491	用户清除与设定
SM492	系统自动显示
SM493	系统显示/用户清除
SM494~SM497	用户清除与设定
SM498~SM499	用户设定，系统执行完成后清除
SM500	系统自动显示
SM501	系统显示/用户清除
SM502~SM504	用户清除与设定
SM505	系统显示/用户清除
SM506	用户清除与设定
SM507	系统显示/用户清除
SM509~SM511	用户清除与设定
SM512	系统自动显示
SM513	系统显示/用户清除
SM514~SM517	用户清除与设定
SM518~SM519	用户设定，系统执行完成后清除
SM520	系统自动显示
SM521	系统显示/用户清除
SM522~SM524	用户清除与设定
SM525	系统显示/用户清除
SM526	用户清除与设定
SM527	系统显示/用户清除
SM529~SM531	用户清除与设定
SM532	系统自动显示
SM533	系统显示/用户清除
SM534~SM537	用户清除与设定
SM538~SM539	用户设定，系统执行完成后清除
SM540	系统自动显示
SM541	系统显示/用户清除
SM542~SM544	用户清除与设定
SM545	系统显示/用户清除
SM546	用户清除与设定
SM547	系统显示/用户清除
SM549~SM551	用户清除与设定

特殊辅助继电器	状态与更新时机说明
SM552	系统自动显示
SM553	系统显示/用户清除
SM554~SM557	用户清除与设定
SM558~SM559	用户设定，系统执行完成后清除
SM560	系统自动显示
SM561	系统显示/用户清除
SM562~SM564	用户清除与设定
SM565	系统显示/用户清除
SM566	用户清除与设定
SM567	系统显示/用户清除
SM569~SM569	用户清除与设定
SM572	系统自动显示
SM573	系统显示/用户清除
SM574	用户清除与设定
SM580	用户设定 ON 关闭高速输出，完成后系统自动清除
SM581	用户设定与清除
SM585	用户设定，系统执行完成后清除
SM586~SM587	用户清除与设定
SM591~SM596	用户设定，系统或用户清除
SM600~SM602	系统自动根据指令结果作设定/清除
SM603	执行 XCMP 指令
SM604	用户设定 ON 做降序排序；设定 OFF 做升幂排序
SM605	用户设定与清除
SM606	用户设定 ON 表 8 位模式，OFF 表 16 位
SM607	用户自行设定
SM608	指令执行时更新
SM609	用户自行设定
SM610~SM611	指令执行时更新
SM612~SM613	用户自行设定
SM614	指令执行时更新
SM615~SM617	用户自行设定
SM618	指令执行时更新
SM619	EI 或 DI 指令执行时更新
SM620	CMPT 指令执行时更新
SM621~SM686	用户自行设定
SM687	RAMP 指令执行时更新
SM688	INCD 指令执行时更新

特殊辅助继电器	状态与更新时机说明
SM690~SM691	用户自行设定
SM692	HKY 指令执行时更新
SM693	SEGL 指令执行时更新
SM694	DSW 指令执行时更新
SM695	用户自行设定
SM749	1.每次下载 Data Exchange 参数后 2.上电时系统更新
SM750~SM783	1.下载 Data Exchange 参数后 2.用户设定或清除
SM784~SM847	系统自动更新
SM861	1.每次下载 Data Exchange 参数后 2.上电时系统更新
SM862~SM895	1.下载 Data Exchange 参数后 2.用户设定或清除
SM896~SM959	系统自动更新
SM976~SM990	系统自动更新
SM1000	用户设定/储存后系统清除
SM1001	以太网络线联机时为 ON，未连接为 OFF
SM1006	1. 下载 Data Exchange 参数后 2. 用户设定/清除
SM1008~SM1015	1. 下载 Data Exchange 参数后 2. 用户设定/清除
SM1016~SM1031	系统自动更新
SM1090	TCP 连接满时为 ON
SM1091	UDP 连接满时为 ON
SM1092	系统更新
SM1100	执行以太网控制指令（API2200-API2210）时或网络线重新连接时
SM1106	PHY 初始化失败时为 ON
SM1107	IP、子网屏蔽、网关设定有误时为 ON
SM1109	Socket 功能启动且使用同一个通讯端口时为 ON
SM1111	用户设定或清除
SM1113	服务器错误时为 ON
SM1116	PLC 参数的触发条件启用时为 ON
SM1117	PLC 参数的触发条件被触发时为 ON
SM1119	触发条件启用时且最后一封邮件已成功送出时为 ON
SM1120	启用本触发条件且最近一封邮件发送错误时为 ON
SM1122~SM1123	触发条件被触发时且 SMTP 服务器响应逾时时为 ON

特殊辅助继电器	状态与更新时机说明
SM1124	触发条件被触发时且附件大小超过限制时为 ON
SM1125	触发条件被触发时且附件不存在时为 ON
SM1126~SM1127	PLC 参数的触发条件启用时为 ON
SM1129	触发条件启用时且最后一封邮件已成功送出时为 ON
SM1130	启用本触发条件且最近一封邮件发送错误时为 ON
SM1132	触发条件被触发时且 SMTP 服务器响应逾时时为 ON
SM1133	触发条件被触发时且 SMTP 服务器响应错误时为 ON
SM1134	触发条件被触发时且附件大小超过限制时为 ON
SM1135	触发条件被触发时且附件不存在时为 ON
SM1136	PLC 参数的触发条件启用时为 ON
SM1137	PLC 参数的触发条件被触发时为 ON
SM1139	触发条件启用时且最后一封邮件已成功送出时为 ON
SM1140	启用本触发条件且最近一封邮件发送错误时为 ON
SM1142	触发条件被触发时且 SMTP 服务器响应逾时时为 ON
SM1143	触发条件被触发时且 SMTP 服务器响应错误时为 ON
SM1144	触发条件被触发时且附件大小超过限制时为 ON
SM1145	触发条件被触发时且附件不存在时为 ON
SM1146	PLC 参数的触发条件启用时为 ON
SM1147	PLC 参数的触发条件被触发时为 ON
SM1149	触发条件启用时且最后一封邮件已成功送出时为 ON
SM1150	启用本触发条件且最近一封邮件发送错误时为 ON
SM1152	触发条件被触发时且 SMTP 服务器响应逾时时为 ON
SM1153	触发条件被触发时且 SMTP 服务器响应错误时为 ON
SM1154	触发条件被触发时且附件大小超过限制时为 ON
SM1155	触发条件被触发时且附件不存在时为 ON
SM1160~SM1161	执行台达设备参数还原
SM1166	1.每次下载 Data Exchange 参数后 2.上电时系统更新
SM1167~SM1199	1.下载 Data Exchange 参数后 2.用户设定或清除
SM1200~SM1263	系统更新
SM1269	Socket 配置有误时为 ON
SM1270~SM1353	执行 socket 功能时更新
SM1312~SM1315	系统自动更新
SM1375~SM1407	1.下载 EtherNet/IP 数据交换参数后 2.用户设定或清除
SM1408~SM1439	EtherNet/IP 数据交换通讯发生错误时为 ON

特殊辅助继电器	状态与更新时机说明
SM1440~SM1447	EtherNet/IP 从站 IO Connection 联机建立且发生通讯逾时为 ON
SM1448~SM1455	1. EtherNet/IP 从站 IO Connection 联机建立为 ON 2. 当未建立联机或发生通讯断线时为 OFF
SM1581~SM1588	用户设定与清除
SM1601~SM1608	用户设定与清除
SM1611~SM1618	系统设定与清除
SM1621~SM1628	系统设定与清除
SM1631~SM1638	系统设定/用户清除
SM1641~SM1648	用户设定与清除
SM1651~SM1658	系统设定与清除
SM1661~SM1668	用户设定与清除
SM1671~SM1682	系统设定与清除
SM1683	系统设定与清除
SM1684~SM1685	用户设定与清除
SM1691~SM1698	系统设定与清除
SM1700~SM1705	用户设定与清除
SM1709~SM1710	用户设定与清除
SM1712~SM1743	用户设定与清除
SM1744~SM1807	系统设定与清除
SM1821~SM1822	用户设定与清除
SM1824~SM1855	用户设定与清除
SM1856~SM1919	系统设定与清除
SM1921~SM1928	系统设定与清除
SM1931~SM1938	用户设定与清除
SM1941~SM1948	用户设定与清除
SM1951~SM1958	系统设定与清除

2.2.9 一般继电器 S

一般继电器 S 为用户控制流程的最基本装置，可自行使用在阶梯图（LD）中进行排程控制。

一般继电器 S 的装置编号为 S0～S2047 共 2048 点，各点 S 装置与一般辅助继电器 M 一样，有输出线圈及 A、B 接点，而且于程序当中无使用次数限制。

依其性质可区分为下列二种：

一般用：一般继电器S于PLC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。

停电保持用：停电保持用一般继电器 S 于 PLC 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。

2.2.10 定时器 T

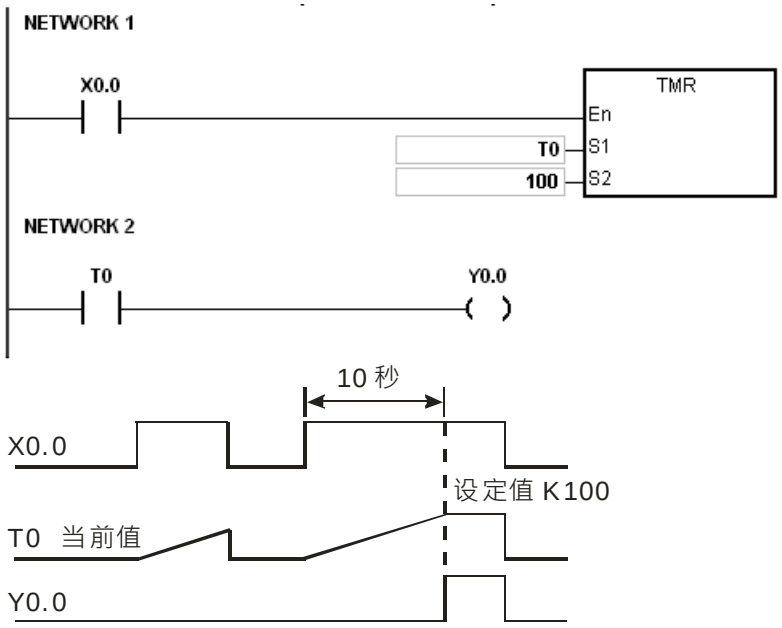
1. 100ms 定时器：TMR 指令所指定的 T 定时器以 100ms 为单位计时
2. 1ms 定时器：TMRH 指令所指定的 T 定时器以 1ms 为单位计时。
3. 运算型 T 定时器为 ST0~ST511，但若要使用装置监控监看时，请监看 T0~T511。
4. 在程序中同一个 T 定时器如果重复使用 (包含使用在不同指令 TMR、TMRH 中)，则设定值以最快到达的为主。
5. 在程序中同一个 T 定时器如果重复使用，其中一个条件接点 OFF 时则 T 会 OFF。
6. 在程序中同一个 T 定时器如果重复使用为 T 与 ST，其中一个条件接点 OFF 时则 T 会 OFF。
7. 当 T 定时器 ON->OFF 且条件式为 ON 时，T 计时值归零并重新计时。
8. 当 TMR 指令执行时，其系统内部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>=设定值)，其接点动作将驱动为 ON。

1. 一般用定时器 T

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 X0.0=ON 时，定时器 T0 的当前值以 100ms 采上数计时，当定时器当前值=设定值 100 时，输出线圈 T0=ON。

- 当 X0.0=OFF 或停电时，定时器 T0 之当前值清为 0，输出线圈 T0 变为 OFF。

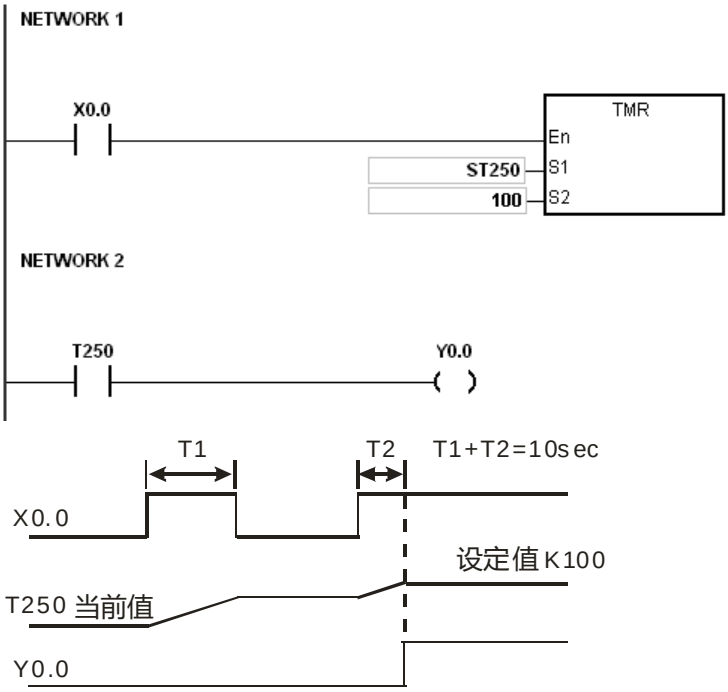


2. 运算型定时器 ST

运算型定时器在 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。只要在装置 T 之前加上一个 S，就会变成运算型定时器 ST 装置，表示目前的 T 变成运算型定时器，则条件接点 OFF 时运算型 T 的值不会被清除，条件接点=ON 的时候，T 由目前的值开始累积计时。

- 当 X0.0=ON 时，定时器 T250 之当前值以 100ms 采上数计时，当定时器当前值=设定值 100 时，输出线圈 T250=ON。
- 当计时中若 X0.0=OFF 或停电时，定时器 T250 暂停计时，当前值不变，待 X0.0 再 ON 时，继续计时，

其当前值往上累加直到定时器当前值=设定值 100 时，输出线圈 T250=ON。



3. 功能块（Function Block）用定时器（T/ST）

功能块或中断插入服务程序中若使用到定时器时，请使用定时器 T412~T511。

功能块用定时器于 END 指令执行时计时一次，在 END 指令执行时，若定时器当前值等于设定值，则输出线圈导通。

一般用的定时器，若是使用在功能块或中断插入中而该功能块不被执行时，定时器就无法正确的被计时。

2.2.11 计数器 C

16 位计数器特点：

项目	16 位计数器
类型	一般型
编号	C0~C511
计数方向	上数
设定值	0~32,767
设定值的指定	常数或数据寄存器 D
当前值的变化	计数到达设定值就不再计数
输出接点	计数到达设定值，接点导通并保持
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

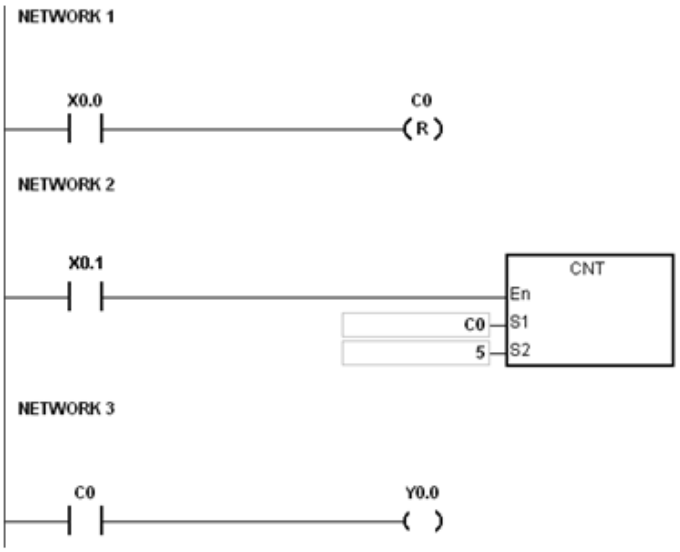
计数器的功能：

计数器的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，设定值为十进制常数值，亦可使用数据寄存器 D 当成设定值。

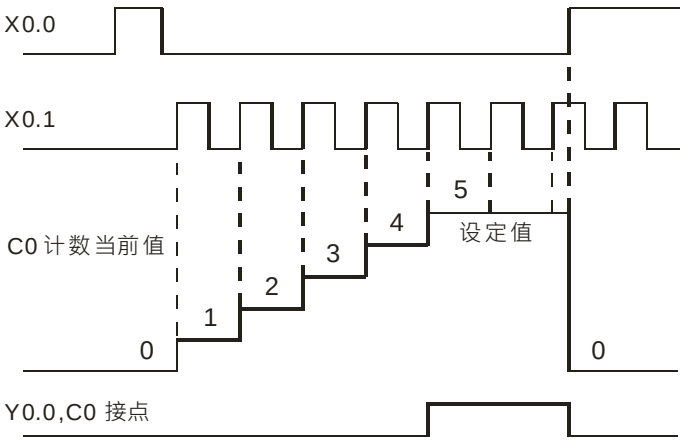
16 位计数器：

1. 16 位计数器的设定范围：0~32,767。(0 与 1 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 PLC 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会将停电前的当前值及计数器接点状态储存着，复电后会继续累计。
3. 若使用 MOV 指令 ISPSoft 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 X0.1 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直接设定或使用寄存器 D 中之数值作间接设定。
5. 设定值可使用常数或使用数据寄存器 D 作为设定值可以是正反数。计数器当前值由 32,767 再往上累计时则变为 0。
6. 此 C 计数器请搭配 CNT 指令，并且请参考 API 1003 CNT 指令说明，下述范例仅为一般常见使用方式。

范例：



1. 当 X0.0=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 X0.1 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 5 时，C0 接点导通，C0 当前值 = 设定值=5。之后的 X0.1 触发信号 C0 完全不接受，C0 当前值保持在 5 处。



2.2.12 32 位计数器 HC

32 位计数器特点：

项目	32 位计数器		
类型	一般上下数型	一般上数型	高速型计数器
编号	HC0 ~ HC63	HC64 ~ HC199	HC200 ~ HC255
计数方向	上、下数	上数	上、下数
设定值	-2,147,483,648~+2,147,483,647		
设定值的指定	常数或数据寄存器 D（32 位 D 寄存器）		
当前值的变化	计数到达设定值后，仍继续计数		
输出接点	上数到达设定值接点导通并保持 ON 下数到小于设定值接点复位成 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 DCNT 指令扫描结束时动作		

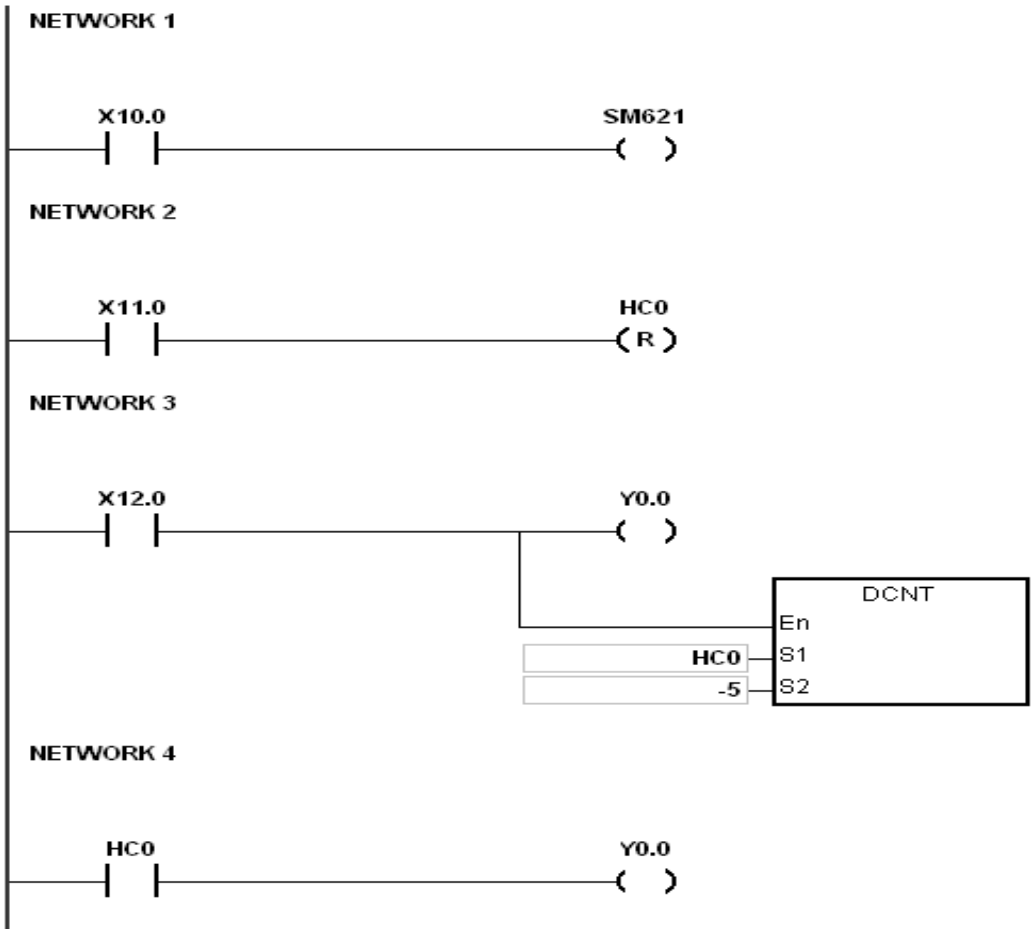
32 位一般用计数器：

1. 32 位一般用计数器的设定范围：-2,147,483,648~2,147,483,647。
2. 32 位一般用上下数计数器切换上/下数需搭配特殊辅助继电器：由 SM621~SM684 来决定。例：SM621=OFF 时决定 HC0 为上数（加 1），SM621=ON 时决定 HC0 为下数（减 1）其余类推。
3. 设定值可使用常数或使用数据寄存器 D 作为设定值可以是正反数，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 D。
4. 一般用计数器在 PLC 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，则会将停电前的当前值及计数器接点状态储存着，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。
6. 此 HC 计数器请搭配 DCNT 指令，并且请参考 API 1004 DCNT 指令说明，下页范例仅为一般常见使用方式。

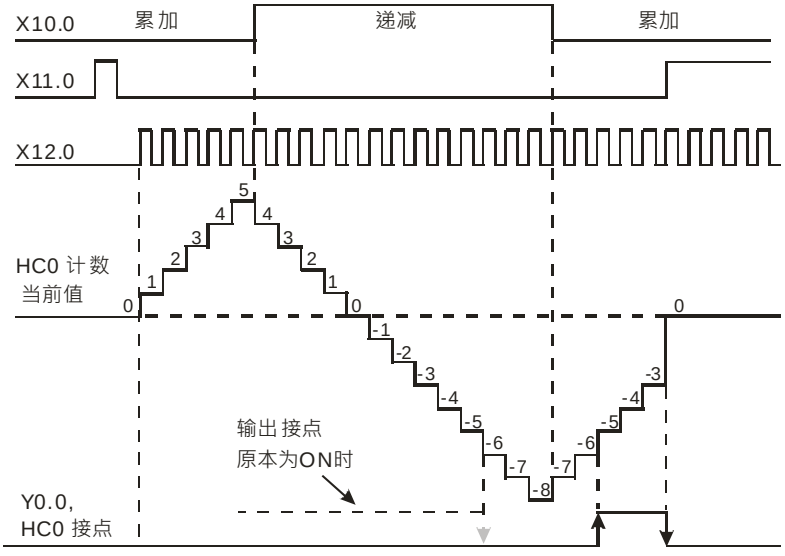
32 位高速型计数器：

详细使用说明请参考 API1004 DCNT 指令说明。

范例：



1. X10.0 驱动 SM621 来决定 HC0 为上数（加 1）或下数（减 1）。
2. 当 X11.0 由 OFF→ON 时，RST 指令执行，HC0 之当前值被清为 0，且接点变为 OFF。
3. 当 X12.0 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 HC0 之当前值从-6→-5 变化时，HC0 接点由 OFF→ON。当计数器 HC0 之当前值从-5→-6 变化时，HC0 接点由 ON→OFF。



2.2.13 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），最高位为正反号，可储存-32,768~+32,767的数值数据，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，而其最高位为正反号，可储存-2,147,483,648~+2,147,483,647的数值数据。亦可将四个16位寄存器合并成一个64位寄存器（D+3，D+2，D+1，D编号小的为下16位）使用，而其最高位为正反号，可储存-9,223,372,036,854,776~+9,223,372,036,854,775,807。也可用于与DIO之外的模块更新CR值之用，与模块更新CR值的D装置配置设定请参考ISPSoft手册的硬件组态说明。

寄存器依其性质可区分为下列三种：

1. 一般用寄存器：当 PLC 由 STOP→RUN 或断电时，寄存器内的数值数据会被清除为 0，如果想要 PLC 由 STOP→RUN 时，数据会保持不被清除，请参考 ISPSoft 手册的硬件组态（HWCONFIG）说明，但断电时仍会被清除为 0。
2. 停电保持用寄存器：当 PLC 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. 常态数据交换区：当 PLC 系统内有配置模块时，此时主机将会自动于每一次扫描周期时，与模块进行一次的常态数据交换，并且会被固定配置在 D26000~D29999 之间。至于常态数据交换区的范围说明请参考操作手册第 8 章说明，以及各模块实际对应之 D 装置编号，请于操作硬件组态配置 HWCONFIG 时，仔细观看软件自动配置之编号区域。

2.2.14 特殊数据寄存器 SR

每个特殊数据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。特殊数据寄存器（SR），它的种类及功能如下所示。在编号前有"*"记号可参考SM/SR补充说明，像其中属性栏中标示为"R"者，表示仅可作读取的动作，若标示为"R/W"，表示可作读写的动作。另若标示为"–"，表示无变化。标示为"#"，则表示系统会依照PLC状态作设定，用户可读取该设定值对照手册的说明，可进一步了解系统信息。而其中停电保持一栏标示为"Y"者，表示有停电保持；标示为"N"者，表示不停电保持；标示为"H"者，表示PLC在上电时，以硬件组态（HWCONFIG）设定值执行，PLC程序执行过程中可修改，但不影响原先硬件组态的设定值。

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR0	运算或元件超出范围错误之侦错号码	○	○	0	0	-	N	R	0
SR1 SR2	运算错误地址（32 位）	○	○	0	0	-	N	R	0
SR4	文法检查错误的侦错号码	○	○	0	0	-	N	R	0
SR5 SR6	文法检查错误地址（32 位）	○	○	0	0	-	N	R	0
*SR8 SR9	WDT 定时器 ON 的 Step 地址（32 位）	○	○	0	-	-	N	R	0
*SR27	AS-F2AD/内建 AD 模拟输入报警代码	○	○	0	-	-	N	R/W	0
SR28	高速输出指令最近一次发生重复输出点的编号	○	○	-1	-1	-1	N	R	-1

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR29	CSFO 指令最近一次发生错误时的输出点编号	○	○	-1	-1	-1	N	R	-1
SR30	记录最近一次高速输出点重复输出时的指令步序编号	○	○	0	0	-	N	R	0
SR31									
SR32	记录最新一次发生超出元件范围的指令地址	○	○	-1	-1	-	N	R	-1
SR33									
*SR36	系统信息储存至储存卡内的功能代码，需搭配 SM36 启动。	○	○	0	-	-	N	R/W	0
SR38	AS-FEN02/FPFN02/FOPC02/FFTP01 功能卡警告/错误代码	○	-	0	-	-	N	R/W	0
*SR40	错误记录（Error Log）的有效组数	○	○	-	-	-	Y	R	0
*SR41	错误记录的指针	○	○	-	-	-	Y	R	0
*SR42	第一组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR43	第一组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR44	第一组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR45	第一组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR46	第一组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR47	第一组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR48	第二组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR49	第二组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR50	第二组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR51	第二组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR52	第二组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR53	第二组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR54	第三组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR55	第三组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR56	第三组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR57	第三组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR58	第三组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR59	第三组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR60	第四组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR61	第四组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR62	第四组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR63	第四组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR64	第四组错误记录的时间日及时	○	○	-	-	-	Y	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR65	第四组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR66	第五组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR67	第五组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR68	第五组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR69	第五组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR70	第五组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR71	第五组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR72	第六组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR73	第六组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR74	第六组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR75	第六组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR76	第六组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR77	第六组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR78	第七组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR79	第七组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR80	第七组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR81	第七组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR82	第七组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR83	第七组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR84	第八组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR85	第八组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR86	第八组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR87	第八组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR88	第八组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR89	第八组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR90	第九组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR91	第九组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR92	第九组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR93	第九组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR94	第九组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR95	第九组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR96	第十组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR97	第十组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR98	第十组错误记录的错误代码	○	○	-	-	-	Y	R	0

特殊 寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR99	第十组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR100	第十组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR101	第十组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR102	第十一组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR103	第十一组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR104	第十一组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR105	第十一组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR106	第十一组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR107	第十一组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR108	第十二组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR109	第十二组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR110	第十二组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR111	第十二组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR112	第十二组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR113	第十二组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR114	第十三组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR115	第十三组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR116	第十三组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR117	第十三组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR118	第十三组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR119	第十三组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR120	第十四组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR121	第十四组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR122	第十四组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR123	第十四组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR124	第十四组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR125	第十四组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR126	第十五组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR127	第十五组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR128	第十五组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR129	第十五组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR130	第十五组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR131	第十五组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR132	第十六组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR133	第十六组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR134	第十六组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR135	第十六组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR136	第十六组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR137	第十六组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR138	第十七组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR139	第十七组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR140	第十七组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR141	第十七组错误记录的时间年及月	○	○	-	-	-	Y	R	0
SR142	第十七组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR143	第十七组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR144	第十八组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR145	第十八组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR146	第十八组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR147	第十八组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR148	第十八组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR149	第十八组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR150	第十九组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR151	第十九组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR152	第十九组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR153	第十九组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR154	第十九组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR155	第十九组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
*SR156	第二十组错误记录之主机或远程编号	○	○	-	-	-	Y	R	0
*SR157	第二十组错误记录的模块编号	○	○	-	-	-	Y	R	0
*SR158	第二十组错误记录的错误代码	○	○	-	-	-	Y	R	0
*SR159	第二十组错误记录的时间年及月	○	○	-	-	-	Y	R	0
*SR160	第二十组错误记录的时间日及时	○	○	-	-	-	Y	R	0
*SR161	第二十组错误记录的时间分及秒	○	○	-	-	-	Y	R	0
SR162 SR163	PLC 已上电时间记录（单位：分钟）（32 位）	○	○	-	-	-	Y	R	-
SR166	VR0 数值显示（需搭配 SM166）	○	-	0	-	-	N	R	0
SR167	VR1 数值显示（需搭配 SM167）	○	-	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR168	AS300 系列：功能卡 1 F2AD 通道 1 输入数值 AS218 系列：内建 AD 通道 1 输入数值	○	○	0	-	-	N	R	0
SR169	AS300 系列：功能卡 1 F2AD 通道 2 输入数值 AS218 系列：内建 AD 通道 2 输入数值	○	○	0	-	-	N	R	0
SR170	功能卡 2 F2AD 通道 1 输入数值	○	-	0	-	-	N	R	0
SR171	功能卡 2 F2AD 通道 2 输入数值	○	-	0	-	-	N	R	0
SR172	AS300 系列：功能卡 1 F2DA 通道 1 输出数值 AS218 系列：内建 DA 通道 1 输出数值	○	○	0	-	0	N	R/W	0
SR173	AS300 系列：功能卡 1 F2DA 通道 2 输出数值 AS218 系列：内建 DA 通道 2 输出数值	○	○	0	-	0	N	R/W	0
SR174	功能卡 2 F2DA 通道 1 输出数值	○	-	0	-	0	N	R/W	0
SR175	功能卡 2 F2DA 通道 2 输出数值	○	-	0	-	0	N	R/W	0
SR176	序列通讯卡 1（COM11）通讯站号	○	-	-	-	-	N	R	1
SR177	序列通讯卡 1（COM11）通讯协议代码	○	-	-	-	-	N	R	0x24
SR178	序列通讯卡 2（COM12）通讯站号	○	-	-	-	-	N	R	1
SR179	序列通讯卡 2（COM12）通讯协议代码	○	-	-	-	-	N	R	0x24
SR180	最近一次警告性错误码	○	○	0	-	-	N	R	0
SR182	序列通讯卡 1（COM11）通讯逾时时间（单位：ms） 设定为 0 则为不逾时	○	-	-	-	-	N	R/W	200
SR183	序列通讯卡 2（COM12）通讯逾时时间（单位：ms） 设定为 0 则为不逾时	○	-	-	-	-	N	R/W	200
SR185	全部远程模块的通讯周期时间（单位：ms）	○	○	0	-	-	N	R	0
SR190	高速计数器群组 1 的倍频设定（默认 1 倍频）	○	○	1	-	-	N	R/W	1
SR191	高速计数器群组 2 的倍频设定（默认 1 倍频）	○	○	1	-	-	N	R/W	1
SR192	高速计数器群组 3 的倍频设定（默认 1 倍频）	○	○	1	-	-	N	R/W	1
SR193	高速计数器群组 4 的倍频设定（默认 1 倍频）	○	○	1	-	-	N	R/W	1
SR194	高速计数器群组 5 的倍频设定（默认 1 倍频）	○	○	1	-	-	N	R/W	1
SR195	高速计数器群组 6 的倍频设定（默认 1 倍频）	○	○	1	-	-	N	R/W	1
SR196	高速计数器群组 7 的倍频设定（默认 1 倍频）	○	○	1	-	-	N	R/W	1
SR197	高速计数器群组 8 的倍频设定（默认 1 倍频）	○	○	1	-	-	N	R/W	1
SR198	圆周率（π）浮点数数值（32 位）	○	○	16#0FDB	16#0FDB	16#0FDB	N	R	16#0FDB
SR199				16#4049	16#4049	16#4049	N		16#4049
*SR201	COM1 通讯站号	○	○	-	-	-	H	R/W	1

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR202	COM2 通讯站号	○	○	-	-	-	H	R/W	1
*SR209	COM1 通讯协议	○	○	-	-	-	H	R/W	16#0024
*SR210	COM1 通讯逾时时间（单位：ms），设定为 0 表不逾时	○	○	-	-	-	H	R/W	0
*SR212	COM2 通讯协议	○	○	-	-	-	H	R/W	16#0024
*SR213	COM2 通讯逾时时间（单位：ms），设定为 0 表不逾时	○	○	-	-	-	H	R/W	0
*SR215	功能卡 1 代号	○	-	-	-	-	N	R	0
*SR216	功能卡 2 代号	○	-	-	-	-	N	R	0
SR217	COM1 自定义波特率数值（单位：100 bps）	○	○	96	-	-	H	R/W	96
SR218	COM2 自定义波特率数值（单位：100 bps）	○	○	96	-	-	H	R/W	96
*SR220	万年历（RTC）年（公元）00~99（BCD）	○	○	-	-	-	Y	R	0
*SR221	万年历（RTC）月 01~12（BCD）	○	○	-	-	-	Y	R	1
*SR222	万年历（RTC）日 01~31（BCD）	○	○	-	-	-	Y	R	1
*SR223	万年历（RTC）时 00~23（BCD）	○	○	-	-	-	Y	R	0
*SR224	万年历（RTC）分 00~59（BCD）	○	○	-	-	-	Y	R	0
*SR225	万年历（RTC）秒 00~59（BCD）	○	○	-	-	-	Y	R	0
*SR226	万年历（RTC）星期 1~7（BCD）	○	○	-	-	-	Y	R	1
*SR227	储存下载动作的有效组数（最多记录 20 次）	○	○	-	-	-	Y	R	0
*SR228	最新一次下载动作的记录指针	○	○	-	-	-	Y	R	0
*SR229	下载记录第 1 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR230	下载记录第 1 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR231	下载记录第 1 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR232	下载记录第 1 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR233	下载记录第 2 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR234	下载记录第 2 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR235	下载记录第 2 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR236	下载记录第 2 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR237	下载记录第 3 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR238	下载记录第 3 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR239	下载记录第 3 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR240	下载记录第 3 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR241	下载记录第 4 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR242	下载记录第 4 组，记录时间年月	○	○	-	-	-	Y	R	0

特殊 寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR243	下载记录第 4 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR244	下载记录第 4 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR245	下载记录第 5 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR246	下载记录第 5 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR247	下载记录第 5 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR248	下载记录第 5 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR249	下载记录第 6 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR250	下载记录第 6 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR251	下载记录第 6 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR252	下载记录第 6 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR253	下载记录第 7 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR254	下载记录第 7 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR255	下载记录第 7 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR256	下载记录第 7 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR257	下载记录第 8 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR258	下载记录第 8 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR259	下载记录第 8 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR260	下载记录第 8 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR261	下载记录第 9 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR262	下载记录第 9 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR263	下载记录第 9 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR264	下载记录第 9 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR265	下载记录第 10 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR266	下载记录第 10 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR267	下载记录第 10 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR268	下载记录第 10 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR269	下载记录第 11 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR270	下载记录第 11 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR271	下载记录第 11 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR272	下载记录第 11 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR273	下载记录第 12 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR274	下载记录第 12 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR275	下载记录第 12 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR276	下载记录第 12 组，记录时间分秒	○	○	-	-	-	Y	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR277	下载记录第 13 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR278	下载记录第 13 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR279	下载记录第 13 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR280	下载记录第 13 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR281	下载记录第 14 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR282	下载记录第 14 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR283	下载记录第 14 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR284	下载记录第 14 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR285	下载记录第 15 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR286	下载记录第 15 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR287	下载记录第 15 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR288	下载记录第 15 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR289	下载记录第 16 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR290	下载记录第 16 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR291	下载记录第 16 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR292	下载记录第 16 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR293	下载记录第 17 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR294	下载记录第 17 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR295	下载记录第 17 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR296	下载记录第 17 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR297	下载记录第 18 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR298	下载记录第 18 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR299	下载记录第 18 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR300	下载记录第 18 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR301	下载记录第 19 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR302	下载记录第 19 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR303	下载记录第 19 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR304	下载记录第 19 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR305	下载记录第 20 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR306	下载记录第 20 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR307	下载记录第 20 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR308	下载记录第 20 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR309	记录 PLC 状态变更的总数（最大 20 组）	○	○	-	-	-	Y	R	0
*SR310	记录 PLC 状态变更最新一次的指针	○	○	-	-	-	Y	R	0

特殊 寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR311	PLC 状态变更记录第 1 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR312	PLC 状态变更记录第 1 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR313	PLC 状态变更记录第 1 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR314	PLC 状态变更记录第 1 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR315	PLC 状态变更记录第 2 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR316	PLC 状态变更记录第 2 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR317	PLC 状态变更记录第 2 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR318	PLC 状态变更记录第 2 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR319	PLC 状态变更记录第 3 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR320	PLC 状态变更记录第 3 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR321	PLC 状态变更记录第 3 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR322	PLC 状态变更记录第 3 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR323	PLC 状态变更记录第 4 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR324	PLC 状态变更记录第 4 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR325	PLC 状态变更记录第 4 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR326	PLC 状态变更记录第 4 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR327	PLC 状态变更记录第 5 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR328	PLC 状态变更记录第 5 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR329	PLC 状态变更记录第 5 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR330	PLC 状态变更记录第 5 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR331	PLC 状态变更记录第 6 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR332	PLC 状态变更记录第 6 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR333	PLC 状态变更记录第 6 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR334	PLC 状态变更记录第 6 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR335	PLC 状态变更记录第 7 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR336	PLC 状态变更记录第 7 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR337	PLC 状态变更记录第 7 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR338	PLC 状态变更记录第 7 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR339	PLC 状态变更记录第 8 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR340	PLC 状态变更记录第 8 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR341	PLC 状态变更记录第 8 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR342	PLC 状态变更记录第 8 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR343	PLC 状态变更记录第 9 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR344	PLC 状态变更记录第 9 组，记录时间年月	○	○	-	-	-	Y	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR345	PLC 状态变更记录第 9 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR346	PLC 状态变更记录第 9 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR347	PLC 状态变更记录第 10 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR348	PLC 状态变更记录第 10 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR349	PLC 状态变更记录第 10 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR350	PLC 状态变更记录第 10 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR351	PLC 状态变更记录第 11 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR352	PLC 状态变更记录第 11 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR353	PLC 状态变更记录第 11 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR354	PLC 状态变更记录第 11 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR355	PLC 状态变更记录第 12 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR356	PLC 状态变更记录第 12 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR357	PLC 状态变更记录第 12 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR358	PLC 状态变更记录第 12 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR359	PLC 状态变更记录第 13 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR360	PLC 状态变更记录第 13 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR361	PLC 状态变更记录第 13 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR362	PLC 状态变更记录第 13 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR363	PLC 状态变更记录第 14 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR364	PLC 状态变更记录第 14 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR365	PLC 状态变更记录第 14 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR366	PLC 状态变更记录第 14 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR367	PLC 状态变更记录第 15 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR368	PLC 状态变更记录第 15 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR369	PLC 状态变更记录第 15 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR370	PLC 状态变更记录第 15 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR371	PLC 状态变更记录第 16 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR372	PLC 状态变更记录第 16 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR373	PLC 状态变更记录第 16 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR374	PLC 状态变更记录第 16 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR375	PLC 状态变更记录第 17 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR376	PLC 状态变更记录第 17 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR377	PLC 状态变更记录第 17 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR378	PLC 状态变更记录第 17 组，记录时间分秒	○	○	-	-	-	Y	R	0

特殊 寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR379	PLC 状态变更记录第 18 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR380	PLC 状态变更记录第 18 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR381	PLC 状态变更记录第 18 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR382	PLC 状态变更记录第 18 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR383	PLC 状态变更记录第 19 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR384	PLC 状态变更记录第 19 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR385	PLC 状态变更记录第 19 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR386	PLC 状态变更记录第 19 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR387	PLC 状态变更记录第 20 组，记录动作编号	○	○	-	-	-	Y	R	0
*SR388	PLC 状态变更记录第 20 组，记录时间年月	○	○	-	-	-	Y	R	0
*SR389	PLC 状态变更记录第 20 组，记录时间日时	○	○	-	-	-	Y	R	0
*SR390	PLC 状态变更记录第 20 组，记录时间分秒	○	○	-	-	-	Y	R	0
*SR391	万年历（RTC）年（公元）00~99（十进制）	○	○	-	-	-	Y	R	0
*SR392	万年历（RTC）月 01~12（十进制）	○	○	-	-	-	Y	R	1
*SR393	万年历（RTC）日 01~31（十进制）	○	○	-	-	-	Y	R	1
*SR394	万年历（RTC）时 00~23（十进制）	○	○	-	-	-	Y	R	0
*SR395	万年历（RTC）分 00~59（十进制）	○	○	-	-	-	Y	R	0
*SR396	万年历（RTC）秒 00~59（十进制）	○	○	-	-	-	Y	R	0
*SR397	万年历（RTC）星期 1~7（十进制）	○	○	-	-	-	Y	R	1
SR403	主机功能卡 1 韧体版本代码（举例：1.06.70 版，其 10 进制数值为 10670，数值为 0 时，表示无韧体）	○	-	-	-	-	N	R	0
SR404	主机功能卡 2 韧体版本代码（举例：1.06.70 版，其 10 进制数值为 10670，数值为 0 时，表示无韧体）	○	-	-	-	-	N	R	0
SR405	主机内部主核心韧体版本代码（举例：十进制数值 10670 = v1.06.70 版）	○	○	-	-	-	Y	R	-
SR406	主机内部副核心韧体版本代码（规则同 SR405，数值为 0 表示无副核心）	○	○	-	-	-	Y	R	-
SR407	RUN 后，每秒加 1，重复计数由 0~32767 再由 -32768~0	○	○	0	0	-	N	R/W	0
SR408	RUN 后 每完成一次扫描后加 1 ,重复计数由 0~32767 再由-32768~0	○	○	0	0	-	N	R/W	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR411	目前扫描时间储存至 SR411 及 SR412，测量单位为 100 微秒，SR411：储存毫秒部分（储存范围：0~65535）SR412：储存微秒部份。（储存范围：0~900）例：目前扫描时间为 12.3 毫秒，储存以下值，SR411=12；SR412=300	○	○	0	-	-	N	R	0
SR412	目前扫描时间储存至 SR411 及 SR412，测量单位为 100 微秒，SR411：储存毫秒部分（储存范围：0~65535）SR412：储存微秒部份。（储存范围：0~900）例：目前扫描时间为 12.3 毫秒，储存以下值，SR411=12；SR412=300	○	○	0	-	-	N	R	0
SR413	储存扫描时间最大值到 SR413 及 SR414，测量单位为 100 微秒 SR413：储存毫秒	○	○	0	-	-	N	R	0
SR414									
SR415	储存扫描时间最小值到 SR415 及 SR416，测量单位为 100 微秒。SR415：储存毫秒	○	○	0	-	-	N	R	0
SR416									
SR421	时间中断 I601 的自定义中断时间（单位时间 1ms），默认 0 表示以 HWCONFIG 设定为准	○	○	0	0	-	N	R/W	0
SR422	时间中断 I602 的自定义中断时间（单位时间 1ms），默认 0 表示以 HWCONFIG 设定为准	○	○	0	0	-	N	R/W	0
SR423	时间中断 I603 的自定义中断时间（单位时间 1ms），默认 0 表示以 HWCONFIG 设定为准	○	○	0	0	-	N	R/W	0
SR424	时间中断 I604 的自定义中断时间（单位时间 0.1ms），默认 0 表示以 HWCONFIG 设定为准	○	○	0	0	-	N	R/W	0
SR440	网络 MAC 地址（注：只会在上电时更新一次此信息）（Ex：12:34:56:78:9A:BC => SR440=16#1234、SR441=16#5678、SR442=16#9ABC）	○	○	-	-	-	Y	R	-
SR441		○	○	-	-	-	Y	R	-
SR442		○	○	-	-	-	Y	R	-
SR443	主机序号（注：只会在上电时更新一次此信息）EX：AS324MTAW15500012 AS➔ SR443 = 16#5341 32 ➔ SR444 = 16#3233 4M ➔ SR445 = 16#4D34 TA ➔ SR446 = 16#4154 W1 ➔ SR447 = 16#3157 55 ➔ SR448 = 16#3535 00 ➔ SR449 = 16#3030 01 ➔ SR450 = 16#3130 2 ➔ SR451 = 16#0032	○	○	-	-	-	Y	R	-
SR444		○	○	-	-	-	Y	R	-
SR445		○	○	-	-	-	Y	R	-
SR446		○	○	-	-	-	Y	R	-
SR447		○	○	-	-	-	Y	R	-
SR448		○	○	-	-	-	Y	R	-
SR449		○	○	-	-	-	Y	R	-
SR450		○	○	-	-	-	Y	R	-
SR451		○	○	-	-	-	Y	R	-

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR453	当储存卡卡有错误发生，错误代码将被记录	○	○	-	-	-	Y	R	0
SR460 SR461	Y0.0 / 轴 1（Y0.0/Y0.1）之当前输出位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR462	轴 1（Y0.0/Y0.1）的输出模式选择	○	○	-	-	-	Y	R/W	0
SR463	轴 1（Y0.0/Y0.1）的启动与结束频率	○	○	-	-	-	Y	R/W	200
SR464	轴 1（Y0.0/Y0.1）的加速时间	○	○	-	-	-	Y	R/W	200
SR465	轴 1（Y0.0/Y0.1）的减速时间	○	○	-	-	-	Y	R/W	200
SR466	轴 1（Y0.0/Y0.1）的寸动（JOG）频率	○	○	-	-	-	Y	R/W	200
SR467	轴 1（Y0.0/Y0.1）规划表输出执行中的编号代码	○	○	0	0	-	N	R	0
SR468	轴 1 的机械单位转换分子数值	○	○	-	-	-	H	R/W	0
SR469	轴 1 的机械单位转换分母数值	○	○	-	-	-	H	R/W	0
SR470 SR471	轴 1 的机械单位位置（单精度浮点数）	○	○	-	-	-	Y	R	0
SR472 SR473	轴 1 的固定式斜率之目标频率	○	○	-	-	-	Y	R	0
SR474 SR475	Y0.1 单点输出的当前位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR476	Y0.1 单点输出的启动/结束频率	○	○	-	-	-	Y	R/W	200
SR477	Y0.1 单点输出的加速/减速时间	○	○	-	-	-	Y	R/W	200
SR478	轴 1（Y0.0）背隙补偿之脉冲个数	○	○	-	-	-	Y	R/W	0
SR479	Y0.1 背隙补偿的脉冲个数	○	○	-	-	-	Y	R/W	0
SR480 SR481	Y0.2 / 轴 2（Y0.2/Y0.3）的当前输出位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR482	轴 2（Y0.2/Y0.3）的输出模式选择	○	○	-	-	-	Y	R/W	0
SR483	Y0.2 / 轴 2（Y0.2/Y0.3）的启动与结束频率	○	○	-	-	-	Y	R/W	200
SR484	Y0.2 / 轴 2（Y0.2/Y0.3）的加速时间	○	○	-	-	-	Y	R/W	200
SR485	Y0.2 / 轴 2（Y0.2/Y0.3）的减速时间	○	○	-	-	-	Y	R/W	200
SR486	轴 2（Y0.2/Y0.3）的寸动（JOG）频率	○	○	-	-	-	Y	R/W	200
SR487	Y0.2 / 轴 2（Y0.2/Y0.3）规划表输出执行中的编号代码	○	○	0	0	-	N	R	0
SR488	轴 2 的机械单位转换分子数值	○	○	-	-	-	H	R/W	0
SR489	轴 2 的机械单位转换分母数值	○	○	-	-	-	H	R/W	0
SR490 SR491	轴 2 之机械单位位置（单精度浮点数）	○	○	-	-	-	Y	R	0
SR492 SR493	轴 2 的固定式斜率的目标频率	○	○	-	-	-	Y	R	0
SR494	Y0.3 单点输出的当前位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR495									
SR496	Y0.3 单点输出的启动/结束频率	○	○	-	-	-	Y	R/W	200
SR497	Y0.3 单点输出的加速/减速时间	○	○	-	-	-	Y	R/W	200
SR498	轴 2（Y0.2）背隙补偿的脉冲个数	○	○	-	-	-	Y	R/W	0
SR499	Y0.3 背隙补偿的脉冲个数	○	○	-	-	-	Y	R/W	0
SR500 SR501	Y0.4 / 轴 3（Y0.4/Y0.5）当前输出位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR502	轴 3（Y0.4/Y0.5）的输出模式选择	○	○	-	-	-	Y	R/W	0
SR503	Y0.4 / 轴 3（Y0.4/Y0.5）的启动与结束频率	○	○	-	-	-	Y	R/W	200
SR504	Y0.4 / 轴 3（Y0.4/Y0.5）的加速时间	○	○	-	-	-	Y	R/W	200
SR505	Y0.4 / 轴 3（Y0.4/Y0.5）的减速时间	○	○	-	-	-	Y	R/W	200
SR506	轴 3（Y0.4/Y0.5）的寸动（JOG）频率	○	○	-	-	-	Y	R/W	200
SR507	轴 3（Y0.4/Y0.5）规划表输出执行中的编号代码	○	○	0	0	-	N	R	0
SR508	轴 3 的机械单位转换分子数值	○	○	-	-	-	H	R/W	0
SR509	轴 3 的机械单位转换分母数值	○	○	-	-	-	H	R/W	0
SR510 SR511	轴 3 的机械单位位置（单精度浮点数）	○	○	-	-	-	Y	R	0
SR512 SR513	轴 3 的固定式斜率的目标频率	○	○	-	-	-	Y	R	0
SR514 SR515	Y0.5 单点输出的当前位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR516	Y0.5 单点输出的启动/结束频率	○	○	-	-	-	Y	R/W	200
SR517	Y0.5 单点输出的加速/减速时间	○	○	-	-	-	Y	R/W	200
SR518	轴 3（Y0.4）背隙补偿的脉冲个数	○	○	-	-	-	Y	R/W	0
SR519	Y0.5 背隙补偿的脉冲个数	○	○	-	-	-	Y	R/W	0
SR520 SR521	Y0.6 / 轴 4（Y0.6/Y0.7）的当前输出位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR522	轴 4（Y0.6/Y0.7）的输出模式选择	○	○	-	-	-	Y	R/W	0
SR523	Y0.6 / 轴 4（Y0.6/Y0.7）的启动与结束频率	○	○	-	-	-	Y	R/W	200
SR524	Y0.6 / 轴 4（Y0.6/Y0.7）的加速时间	○	○	-	-	-	Y	R/W	200
SR525	Y0.6 / 轴 4（Y0.6/Y0.7）的减速时间	○	○	-	-	-	Y	R/W	200
SR526	轴 4（Y0.6/Y0.7）的寸动（JOG）频率	○	○	-	-	-	Y	R/W	200
SR527	轴 4（Y0.6/Y0.7）规划表输出执行中的编号代码	○	○	0	0	-	N	R	0
SR528	轴 4 的机械单位转换分子数值	○	○	-	-	-	H	R/W	0
SR529	轴 4 的机械单位转换分母数值	○	○	-	-	-	H	R/W	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR530 SR531	轴 4 之机械单位位置（单精度浮点数）	○	○	-	-	-	Y	R	0
SR532 SR533	轴 4 的固定式斜率之目标频率	○	○	-	-	-	Y	R	0
SR534 SR535	Y0.7 单点输出之当前位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR536	Y0.7 单点输出之启动/结束频率	○	○	-	-	-	Y	R/W	200
SR537	Y0.7 单点输出之加速/减速时间	○	○	-	-	-	Y	R/W	200
SR538	轴 4（Y0.6）背隙补偿之脉冲个数	○	○	-	-	-	Y	R/W	0
SR539	Y0.7 背隙补偿之脉冲个数	○	○	-	-	-	Y	R/W	0
SR540 SR541	Y0.8 / 轴 5（Y0.8/Y0.9）之当前输出位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR542	轴 5（Y0.8/Y0.9）之输出模式选择	○	○	-	-	-	Y	R/W	0
SR543	Y0.8 / 轴 5（Y0.8/Y0.9）的启动与结束频率	○	○	-	-	-	Y	R/W	200
SR544	Y0.8 / 轴 5（Y0.8/Y0.9）的加速时间	○	○	-	-	-	Y	R/W	200
SR545	Y0.8 / 轴 5（Y0.8/Y0.9）的减速时间	○	○	-	-	-	Y	R/W	200
SR546	轴 5（Y0.8/Y0.9）的寸动（JOG）频率	○	○	-	-	-	Y	R/W	200
SR547	轴 5（Y0.8/Y0.9）规划表输出执行中的编号代码	○	○	0	0	-	N	R	0
SR548	轴 5 的机械单位转换分子数值	○	○	-	-	-	H	R/W	0
SR549	轴 5 的机械单位转换分母数值	○	○	-	-	-	H	R/W	0
SR550 SR551	轴 5 的机械单位位置（单精度浮点数）	○	○	-	-	-	Y	R	0
SR552 SR553	轴 5 的固定式斜率的目标频率	○	○	-	-	-	Y	R	0
SR554 SR555	Y0.9 单点输出的当前位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR556	Y0.9 单点输出的启动/结束频率	○	○	-	-	-	Y	R/W	200
SR557	Y0.9 单点输出的加速/减速时间	○	○	-	-	-	Y	R/W	200
SR558	轴 5（Y0.8）背隙补偿的脉冲个数	○	○	-	-	-	Y	R/W	0
SR559	Y0.9 背隙补偿的脉冲个数	○	○	-	-	-	Y	R/W	0
SR560 SR561	Y0.10 / 轴 6（Y0.10/Y0.11）的当前输出位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR562	轴 6（Y0.10/Y0.11）的输出模式选择	○	○	-	-	-	Y	R/W	0
SR563	Y0.10 / 轴 6（Y0.10/Y0.11）的启动与结束频率	○	○	-	-	-	Y	R/W	200
SR564	Y0.10 / 轴 6（Y0.10/Y0.11）的加速时间	○	○	-	-	-	Y	R/W	200
SR565	Y0.10 / 轴 6（Y0.10/Y0.11）的减速时间	○	○	-	-	-	Y	R/W	200

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR566	轴 6（Y0.10/Y0.11）的寸动（JOG）频率	○	○	-	-	-	Y	R/W	200
SR567	轴 6（Y0.10/Y0.11）规划表输出执行中的编号代码	○	○	0	0	-	N	R	0
SR568	轴 6 的机械单位转换分子数值	○	○	-	-	-	H	R/W	0
SR569	轴 6 的机械单位转换分母数值	○	○	-	-	-	H	R/W	0
SR570 SR571	轴 6 之机械单位位置（单精度浮点数）	○	○	-	-	-	Y	R	0
SR572 SR573	轴 6 的固定式斜率的目标频率	○	○	-	-	-	Y	R	0
SR574 SR575	Y0.11 单点输出的当前位置（单位：脉冲个数）	○	○	-	-	-	Y	R/W	0
SR576	Y0.11 单点输出的启动/结束频率	○	○	-	-	-	Y	R/W	200
SR577	Y0.11 单点输出的加速/减速时间	○	○	-	-	-	Y	R/W	200
SR578	轴 6（Y0.10）背隙补偿之脉冲个数	○	○	-	-	-	Y	R/W	0
SR579	Y0.11 背隙补偿的脉冲个数	○	○	-	-	-	Y	R/W	0
*SR580 *SR581	轴 1（Y0.0/Y0.1）正向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR582 *SR583	轴 1（Y0.0/Y0.1）反向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR584 *SR585	轴 2（Y0.2/Y0.3）正向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR586 *SR587	轴 2（Y0.2/Y0.3）反向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR588 *SR589	轴 3（Y0.4/Y0.5）正向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR590 *SR591	轴 3（Y0.4/Y0.5）反向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR592 *SR593	轴 4（Y0.6/Y0.7）正向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR594 *SR595	轴 4（Y0.6/Y0.7）反向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR596 *SR597	轴 5（Y0.8/Y0.9）正向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR598 *SR599	轴 5（Y0.8/Y0.9）反向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
*SR600 *SR601	轴 6（Y0.10/Y0.11）正向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR602 *SR603	轴 6（Y0.10/Y0.11）反向软件极限位置（单位：脉冲个数）	○	○	-	-	-	H	R/W	0
SR610 SR611	轴 1（Y0.0/Y0.1）当前输出速度（单位：Hz）	○	○	0	0	0	N	R	0
SR612 SR613	轴 2（Y0.2/Y0.3）当前输出速度（单位：Hz）	○	○	0	0	0	N	R	0
SR614 SR615	轴 3（Y0.4/Y0.5）当前输出速度（单位：Hz）	○	○	0	0	0	N	R	0
SR616 SR617	轴 4（Y0.6/Y0.7）当前输出速度（单位：Hz）	○	○	0	0	0	N	R	0
SR618 SR619	轴 5（Y0.8/Y0.9）当前输出速度（单位：Hz）	○	○	0	0	0	N	R	0
SR620 SR621	轴 6（Y0.10/Y0.11）当前输出速度（单位：Hz）	○	○	0	0	0	N	R	0
SR623	外部输入中断 X0.0~X0.15 下降沿触发条件显示状态	○	○	FFFF	FFFF	-	N	R	FFFF
SR624	外部输入中断 X0.0~X0.15 上升沿触发条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR625	高速比较中断 I200~I233 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR626	高速比较中断 I240~I253 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR627	高速比较中断 I260~I267 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR628	通讯中断 I300~I307 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR629	输出中断 I500~I505 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR630	输出中断 I510~I519 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR632	时间中断 I601~I604 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR633	扩展模块中断 I400~I415 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR634	扩展模块中断 I416~I431 条件状态显示	○	○	FFFF	FFFF	-	N	R	FFFF
SR640	自定义搭配 Y0.0 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR641	自定义搭配 Y0.1 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR642	自定义搭配 Y0.2 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR643	自定义搭配 Y0.3 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR644	自定义搭配 Y0.4 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR645	自定义搭配 Y0.5 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR646	自定义搭配 Y0.6 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR647	自定义搭配 Y0.7 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR648	自定义搭配 Y0.8 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR649	自定义搭配 Y0.9 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR650	自定义搭配 Y0.10 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
SR651	自定义搭配 Y0.11 输出脉冲方向脚的提前输出时间设定 0~20ms（单位 1ms）	○	○	0	-	-	N	R/W	0
*SR652 *SR653	显示最新定位输出加速区段脉冲个数（32 位）	○	○	0	0	-	N	R	0
*SR654 *SR655	显示最新定位输出减速区段脉冲个数（32 位）	○	○	0	0	-	N	R	0
SR656 SR657	台达驱动器 ECAT 通讯 SDO abort code（32 位）	○	-	0	0	-	N	R/W	0
SR658	台达驱动器 CAN 通讯错误站号	○	○	0	-	-	N	R/W	0
	台达驱动器 ECAT 通讯错误站号	○	-	0	-	-	N	R/W	0
SR659	台达驱动器 CAN 通讯错误代码	○	○	0	-	-	N	R/W	0
	台达驱动器 ECAT 通讯错误代码	○	-	0	-	-	N	R/W	0
SR661	台达伺服专属 CAN 通讯站号 1 之 PR 命令	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 1 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR662	台达伺服专属 CAN 通讯站号 2 之 PR 命令	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 2 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR663	台达伺服专属 CAN 通讯站号 3 之 PR 命令	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 3 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR664	台达伺服专属 CAN 通讯站号 4 之 PR 命令	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
	台达驱动器 ECAT 通讯站号 4 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR665	台达伺服专属 CAN 通讯站号 5 之 PR 命令	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 5 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR666	台达伺服专属 CAN 通讯站号 6 之 PR 命令	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 6 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR667	台达伺服专属 CAN 通讯站号 7 之 PR 命令	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 7 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR668	台达伺服专属 CAN 通讯站号 8 之 PR 命令	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 8 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR671	台达伺服专属 CAN 通讯站号 1 之异警代码	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 1 之伺服异警代码	○	-	0	-	-	N	R	0
SR672	台达伺服专属 CAN 通讯站号 2 之异警代码	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 2 之伺服异警代码	○	-	0	-	-	N	R	0
SR673	台达伺服专属 CAN 通讯站号 3 之异警代码	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 3 之伺服异警代码	○	-	0	-	-	N	R	0
SR674	台达伺服专属 CAN 通讯站号 4 之异警代码	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 4 之伺服异警代码	○	-	0	-	-	N	R	0
SR675	台达伺服专属 CAN 通讯站号 5 之异警代码	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 5 之伺服异警代码	○	-	0	-	-	N	R	0
SR676	台达伺服专属 CAN 通讯站号 6 之异警代码	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 6 之伺服异警代码	○	-	0	-	-	N	R	0
SR677	台达伺服专属 CAN 通讯站号 7 之异警代码	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 7 之伺服异警代码	○	-	0	-	-	N	R	0
SR678	台达伺服专属 CANo 通讯站号 8 之异警代码	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 8 之伺服异警代码	○	-	0	-	-	N	R	0
SR681	台达伺服专属 CAN 通讯站号 1 之 DO 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 1 之伺服 DO 状态	○	-	0	-	-	N	R	0
SR682	台达伺服专属 CAN 通讯站号 2 之 DO 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 2 之伺服 DO 状态	○	-	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR683	台达伺服专属 CAN 通讯站号 3 之 DO 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 3 之伺服 DO 状态	○	-	0	-	-	N	R	0
SR684	台达伺服专属 CAN 通讯站号 4 之 DO 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 4 之伺服 DO 状态	○	-	0	-	-	N	R	0
SR685	台达伺服专属 CAN 通讯站号 5 之 DO 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 5 之伺服 DO 状态	○	-	0	-	-	N	R	0
SR686	台达伺服专属 CAN 通讯站号 6 之 DO 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 6 之伺服 DO 状态	○	-	0	-	-	N	R	0
SR687	台达伺服专属 CAN 通讯站号 7 之 DO 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 7 之伺服 DO 状态	○	-	0	-	-	N	R	0
SR688	台达伺服专属 CAN 通讯站号 8 之 DO 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 8 之伺服 DO 状态	○	-	0	-	-	N	R	0
SR691 SR692	台达伺服专属 CAN 通讯站号 1 现在位置（32 位）	○	○	0	-	-	N	R	0
SR691 SR692	台达驱动器 ECAT 通讯站号 1 之伺服现在位置(32 位)	○	-	0	-	-	N	R	0
SR693 SR694	台达伺服专属 CAN 通讯站号 2 现在位置（32 位）	○	○	0	-	-	N	R	0
SR693 SR694	台达驱动器 ECAT 通讯站号 2 之伺服现在位置(32 位)	○	-	0	-	-	N	R	0
SR695 SR696	台达伺服专属 CAN 通讯站号 3 现在位置（32 位）	○	○	0	-	-	N	R	0
SR695 SR696	台达驱动器 ECAT 通讯站号 3 之伺服现在位置(32 位)	○	-	0	-	-	N	R	0
SR697 SR698	台达伺服专属 CAN 通讯站号 4 现在位置（32 位）	○	○	0	-	-	N	R	0
SR697 SR698	台达驱动器 ECAT 通讯站号 4 之伺服现在位置(32 位)	○	-	0	-	-	N	R	0
SR699 SR700	台达伺服专属 CAN 通讯站号 5 现在位置（32 位）	○	○	0	-	-	N	R	0
SR699 SR700	台达驱动器 ECAT 通讯站号 5 之伺服现在位置(32 位)	○	-	0	-	-	N	R	0
SR701 SR702	台达伺服专属 CAN 通讯站号 6 现在位置（32 位）	○	○	0	-	-	N	R	0
SR701 SR702	台达驱动器 ECAT 通讯站号 6 之伺服现在位置(32 位)	○	-	0	-	-	N	R	0
SR703 SR704	台达伺服专属 CAN 通讯站号 7 现在位置（32 位）	○	○	0	-	-	N	R	0
SR703 SR704	台达驱动器 ECAT 通讯站号 7 之伺服现在位置(32 位)	○	-	0	-	-	N	R	0
SR705 SR706	台达伺服专属 CAN 通讯站号 8 现在位置（32 位）	○	○	0	-	-	N	R	0
SR705	台达驱动器 ECAT 通讯站号 8 之伺服现在位置(32 位)	○	-	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR706									
SR711 SR712	台达伺服专属 CAN 通讯站号 1 目标位置（32 位）	○	○	0	-	-	N	R	0
SR711 SR712	台达驱动器 ECAT 通讯站号 1 之伺服目标位置(32 位)	○	-	0	-	-	N	R	0
SR713 SR714	台达伺服专属 CAN 通讯站号 2 目标位置（32 位）	○	○	0	-	-	N	R	0
SR713 SR714	台达驱动器 ECAT 通讯站号 2 之伺服目标位置(32 位)	○	-	0	-	-	N	R	0
SR715 SR716	台达伺服专属 CAN 通讯站号 3 目标位置（32 位）	○	○	0	-	-	N	R	0
SR715 SR716	台达驱动器 ECAT 通讯站号 3 之伺服目标位置(32 位)	○	-	0	-	-	N	R	0
SR717 SR718	台达伺服专属 CAN 通讯站号 4 目标位置（32 位）	○	○	0	-	-	N	R	0
SR717 SR718	台达驱动器 ECAT 通讯站号 4 之伺服目标位置(32 位)	○	-	0	-	-	N	R	0
SR719 SR720	台达伺服专属 CAN 通讯站号 5 目标位置（32 位）	○	○	0	-	-	N	R	0
SR719 SR720	台达驱动器 ECAT 通讯站号 5 之伺服目标位置(32 位)	○	-	0	-	-	N	R	0
SR721 SR722	台达伺服专属 CAN 通讯站号 6 目标位置（32 位）	○	○	0	-	-	N	R	0
SR721 SR722	台达驱动器 ECAT 通讯站号 6 之伺服目标位置(32 位)	○	-	0	-	-	N	R	0
SR723 SR724	台达伺服专属 CAN 通讯站号 7 目标位置（32 位）	○	○	0	-	-	N	R	0
SR723 SR724	台达驱动器 ECAT 通讯站号 7 之伺服目标位置(32 位)	○	-	0	-	-	N	R	0
SR725 SR726	台达伺服专属 CAN 通讯站号 8 目标位置（32 位）	○	○	0	-	-	N	R	0
SR725 SR726	台达驱动器 ECAT 通讯站号 8 之伺服目标位置(32 位)	○	-	0	-	-	N	R	0
SR731	台达伺服专属 CAN 通讯站号 1 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 1 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR732	台达伺服专属 CAN 通讯站号 2 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 2 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR733	台达伺服专属 CAN 通讯站号 3 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 3 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR734	台达伺服专属 CAN 通讯站号 4 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 4 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR735	台达伺服专属 CAN 通讯站号 5 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 5 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR736	台达伺服专属 CAN 通讯站号 6 之 DI 状态	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
	台达驱动器 ECAT 通讯站号 6 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR737	台达伺服专属 CAN 通讯站号 7 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 7 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR738	台达伺服专属 CAN 通讯站号 8 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 8 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR741	台达伺服专属 CAN 通讯站号 1 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 1 之伺服当前扭矩（单位 0.1%）	○	-	0	-	-	N	R	0
SR742	台达伺服专属 CAN 通讯站号 2 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 2 之伺服当前扭矩（单位 0.1%）	○	-	0	-	-	N	R	0
SR743	台达伺服专属 CAN 通讯站号 3 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 3 之伺服当前扭矩（单位 0.1%）	○	-	0	-	-	N	R	0
SR744	台达伺服专属 CAN 通讯站号 4 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 4 之伺服当前扭矩（单位 0.1%）	○	-	0	-	-	N	R	0
SR745	台达伺服专属 CAN 通讯站号 5 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 5 之伺服当前扭矩（单位 0.1%）	○	-	0	-	-	N	R	0
SR746	台达伺服专属 CAN 通讯站号 6 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 6 之伺服当前扭矩（单位 0.1%）	○	-	0	-	-	N	R	0
SR747	台达伺服专属 CAN 通讯站号 7 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 7 之伺服当前扭矩（单位 0.1%）	○	-	0	-	-	N	R	0
SR748	台达伺服专属 CAN 通讯站号 8 之当前扭矩（单位	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
	0.1%)								
	台达驱动器 ECAT 通讯站号 8 之伺服当前扭矩 (单位 0.1%)	○	-	0	-	-	N	R	0
SR751	台达变频专属 CAN 通讯站号 21 之当前状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 9 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR752	台达变频专属 CAN 通讯站号 22 之当前状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 10 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR753	台达变频专属 CAN 通讯站号 23 之当前状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 11 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR754	台达变频专属 CAN 通讯站号 24 之当前状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 12 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR755	台达变频专属 CAN 通讯站号 25 之当前状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 13 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR756	台达变频专属 CAN 通讯站号 26 之当前状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 14 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR757	台达变频专属 CAN 通讯站号 27 之当前状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 15 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR758	台达变频专属 CAN 通讯站号 28 之当前状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 16 之伺服 PR 命令 / 变频状态	○	-	0	-	-	N	R	0
SR761	台达变频专属 CAN 通讯站号 21 之当前转速	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 9 之伺服异警代码	○	-	0	-	-	N	R	0
SR762	台达变频专属 CAN 通讯站号 22 之当前转速	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 10 之伺服异警代码	○	-	0	-	-	N	R	0
SR763	台达变频专属 CAN 通讯站号 23 之当前转速	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 11 之伺服异警代码	○	-	0	-	-	N	R	0
SR764	台达变频专属 CAN 通讯站号 24 之当前转速	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
	台达驱动器 ECAT 通讯站号 12 之伺服异警代码	○	-	0	-	-	N	R	0
SR765	台达变频专属 CAN 通讯站号 25 之当前转速	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 13 之伺服异警代码	○	-	0	-	-	N	R	0
SR766	台达变频专属 CAN 通讯站号 26 之当前转速	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 14 之伺服异警代码	○	-	0	-	-	N	R	0
SR767	台达变频专属 CAN 通讯站号 27 之当前转速	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 15 之伺服异警代码	○	-	0	-	-	N	R	0
SR768	台达变频专属 CAN 通讯站号 28 之当前转速	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 16 之伺服异警代码	○	-	0	-	-	N	R	0
SR771	台达变频专属 CAN 通讯站号 21 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 9 之伺服当前扭矩（单位 0.1%）	○	-	0	-	-	N	R	0
SR772	台达变频专属 CAN 通讯站号 22 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 10 之伺服当前扭矩(单位 0.1%）	○	-	0	-	-	N	R	0
SR773	台达变频专属 CAN 通讯站号 23 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 11 之伺服当前扭矩(单位 0.1%）	○	-	0	-	-	N	R	0
SR774	台达变频专属 CAN 通讯站号 24 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 12 之伺服当前扭矩(单位 0.1%）	○	-	0	-	-	N	R	0
SR775	台达变频专属 CAN 通讯站号 25 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 13 之伺服当前扭矩(单位 0.1%）	○	-	0	-	-	N	R	0
SR776	台达变频专属 CAN 通讯站号 26 之当前扭矩（单位 0.1%）	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 14 之伺服当前扭矩(单位 0.1%）	○	-	0	-	-	N	R	0
SR777	台达变频专属 CAN 通讯站号 27 之当前扭矩（单位	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
	0.1%)								
	台达驱动器 ECAT 通讯站号 15 之伺服当前扭矩(单位 0.1%)	○	-	0	-	-	N	R	0
SR778	台达变频专属 CAN 通讯站号 28 之当前扭矩 (单位 0.1%)	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 16 之伺服当前扭矩(单位 0.1%)	○	-	0	-	-	N	R	0
SR781	台达变频专属 CAN 通讯站号 21 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 9 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR782	台达变频专属 CAN 通讯站号 22 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 10 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR783	台达变频专属 CAN 通讯站号 23 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 11 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR784	台达变频专属 CAN 通讯站号 24 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 12 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR785	台达变频专属 CAN 通讯站号 25 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 13 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR786	台达变频专属 CAN 通讯站号 26 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 14 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR787	台达变频专属 CAN 通讯站号 27 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 15 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR788	台达变频专属 CAN 通讯站号 28 之 DI 状态	○	○	0	-	-	N	R	0
	台达驱动器 ECAT 通讯站号 16 之伺服 DI 状态	○	-	0	-	-	N	R	0
SR791	台达伺服专属 CAN 通讯站号 1 读取自定义数值 (32 位)	○	○	0	-	-	N	R	0
SR792									
SR791	台达驱动器 ECAT 通讯站号 1 之伺服读取自定义数值 (32 位)	○	-	0	-	-	N	R	0
SR792									
SR793	台达伺服专属 CAN 通讯站号 2 读取自定义数值 (32 位)	○	○	0	-	-	N	R	0
SR794									
SR793	台达驱动器 ECAT 通讯站号 2 之伺服读取自定义数值 (32 位)	○	-	0	-	-	N	R	0
SR794									
SR795	台达伺服专属 CAN 通讯站号 3 读取自定义数值 (32 位)	○	○	0	-	-	N	R	0
SR796									
SR795	台达驱动器 ECAT 通讯站号 3 之伺服读取自定义数值	○	-	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR796	(32 位)								
SR797 SR798	台达伺服专属 CAN 通讯站号 4 读取自定义数值 (32 位)	○	○	0	-	-	N	R	0
SR797 SR798	台达驱动器 ECAT 通讯站号 4 之伺服读取自定义数值 (32 位)	○	-	0	-	-	N	R	0
SR799 SR800	台达伺服专属 CAN 通讯站号 5 读取自定义数值 (32 位)	○	○	0	-	-	N	R	0
SR799 SR800	台达驱动器 ECAT 通讯站号 5 之伺服读取自定义数值 (32 位)	○	-	0	-	-	N	R	0
SR801 SR802	台达伺服专属 CAN 通讯站号 6 读取自定义数值 (32 位)	○	○	0	-	-	N	R	0
SR801 SR802	台达驱动器 ECAT 通讯站号 6 之伺服读取自定义数值 (32 位)	○	-	0	-	-	N	R	0
SR803 SR804	台达伺服专属 CAN 通讯站号 7 读取自定义数值 (32 位)	○	○	0	-	-	N	R	0
SR803 SR804	台达驱动器 ECAT 通讯站号 7 之伺服读取自定义数值 (32 位)	○	-	0	-	-	N	R	0
SR805 SR806	台达伺服专属 CAN 通讯站号 8 读取自定义数值 (32 位)	○	○	0	-	-	N	R	0
SR805 SR806	台达驱动器 ECAT 通讯站号 8 之伺服读取自定义数值 (32 位)	○	-	0	-	-	N	R	0
SR811	台达伺服专属 CAN 通讯站号 1 定位完成允许范围 (0 表示现在位置与目标位置相同，才能设定完成标志。假设数值为 5，表示现在位置与目标位置差距+5 ~ -5 以内，PLC 将会设定完成标志)	○	○	0	-	-	N	RW	0
	台达驱动器 ECAT 通讯站号 1 之伺服定位完成允许范围	○	-	0	-	-	N	RW	0
SR812	台达伺服专属 CAN 通讯站号 2 定位完成允许范围	○	○	0	-	-	N	RW	0
	台达驱动器 ECAT 通讯站号 2 之伺服定位完成允许范围	○	-	0	-	-	N	RW	0
SR813	台达伺服专属 CAN 通讯站号 3 定位完成允许范围	○	○	0	-	-	N	RW	0
	台达驱动器 ECAT 通讯站号 3 之伺服定位完成允许范围	○	-	0	-	-	N	RW	0
SR814	台达伺服专属 CAN 通讯站号 4 定位完成允许范围	○	○	0	-	-	N	RW	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
	台达驱动器 ECAT 通讯站号 4 之伺服定位完成允许范围	○	-	0	-	-	N	RW	0
SR815	台达伺服专属 CAN 通讯站号 5 定位完成允许范围	○	○	0	-	-	N	RW	0
	台达驱动器 ECAT 通讯站号 5 之伺服定位完成允许范围	○	-	0	-	-	N	RW	0
SR816	台达伺服专属 CAN 通讯站号 6 定位完成允许范围	○	○	0	-	-	N	RW	0
	台达驱动器 ECAT 通讯站号 6 之伺服定位完成允许范围	○	-	0	-	-	N	RW	0
SR817	台达伺服专属 CAN 通讯站号 7 定位完成允许范围	○	○	0	-	-	N	RW	0
	台达驱动器 ECAT 通讯站号 7 之伺服定位完成允许范围	○	-	0	-	-	N	RW	0
SR818	台达伺服专属 CAN 通讯站号 8 定位完成允许范围	○	○	0	-	-	N	RW	0
	台达驱动器 ECAT 通讯站号 8 之伺服定位完成允许范围	○	-	0	-	-	N	RW	0
SR820	CANopen DS301 主从站状态代码	○	○	-	-	-	Y	R	0
SR821	CANopen DS301 版本代码	○	○	-	-	-	Y	R	-
SR822	CANopen 通讯速率（基本单位为 1kbps）	○	○	-	-	-	H	R	125
*SR823	CAN 通讯的硬件接收错误计数值（Max. 255）	○	○	0	-	-	N	R	0
*SR824	CAN 通讯的硬件传送错误计数值（Max. 255）	○	○	0	-	-	N	R	0
SR825	CANopen DS301 主站运行状态代码	○	○	-1	-	-	N	R	-1
SR826	CANopen DS301 从站编号 1~16 运行状态标志	○	○	-1	-	-	N	R	-1
SR827	CANopen DS301 从站编号 17~32 运行状态标志	○	○	-1	-	-	N	R	-1
SR828	CANopen DS301 从站编号 33~48 运行状态标志	○	○	-1	-	-	N	R	-1
SR829	CANopen DS301 从站编号 49~64 运行状态标志	○	○	-1	-	-	N	R	-1
SR830	CANopen DS301 从站编号 1 运行状态代码	○	○	-1	-	-	N	R	-1
SR831	CANopen DS301 从站编号 2 运行状态代码	○	○	-1	-	-	N	R	-1
SR832	CANopen DS301 从站编号 3 运行状态代码	○	○	-1	-	-	N	R	-1
SR833	CANopen DS301 从站编号 4 运行状态代码	○	○	-1	-	-	N	R	-1
SR834	CANopen DS301 从站编号 5 运行状态代码	○	○	-1	-	-	N	R	-1
SR835	CANopen DS301 从站编号 6 运行状态代码	○	○	-1	-	-	N	R	-1
SR836	CANopen DS301 从站编号 7 运行状态代码	○	○	-1	-	-	N	R	-1
SR837	CANopen DS301 从站编号 8 运行状态代码	○	○	-1	-	-	N	R	-1
SR838	CANopen DS301 从站编号 9 运行状态代码	○	○	-1	-	-	N	R	-1
SR839	CANopen DS301 从站编号 10 运行状态代码	○	○	-1	-	-	N	R	-1

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR840	CANopen DS301 从站编号 11 运行状态代码	○	○	-1	-	-	N	R	-1
SR841	CANopen DS301 从站编号 12 运行状态代码	○	○	-1	-	-	N	R	-1
SR842	CANopen DS301 从站编号 13 运行状态代码	○	○	-1	-	-	N	R	-1
SR843	CANopen DS301 从站编号 14 运行状态代码	○	○	-1	-	-	N	R	-1
SR844	CANopen DS301 从站编号 15 运行状态代码	○	○	-1	-	-	N	R	-1
SR845	CANopen DS301 从站编号 16 运行状态代码	○	○	-1	-	-	N	R	-1
SR846	CANopen DS301 从站编号 17 运行状态代码	○	○	-1	-	-	N	R	-1
SR847	CANopen DS301 从站编号 18 运行状态代码	○	○	-1	-	-	N	R	-1
SR848	CANopen DS301 从站编号 19 运行状态代码	○	○	-1	-	-	N	R	-1
SR849	CANopen DS301 从站编号 20 运行状态代码	○	○	-1	-	-	N	R	-1
SR850	CANopen DS301 从站编号 21 运行状态代码	○	○	-1	-	-	N	R	-1
SR851	CANopen DS301 从站编号 22 运行状态代码	○	○	-1	-	-	N	R	-1
SR852	CANopen DS301 从站编号 23 运行状态代码	○	○	-1	-	-	N	R	-1
SR853	CANopen DS301 从站编号 24 运行状态代码	○	○	-1	-	-	N	R	-1
SR854	CANopen DS301 从站编号 25 运行状态代码	○	○	-1	-	-	N	R	-1
SR855	CANopen DS301 从站编号 26 运行状态代码	○	○	-1	-	-	N	R	-1
SR856	CANopen DS301 从站编号 27 运行状态代码	○	○	-1	-	-	N	R	-1
SR857	CANopen DS301 从站编号 28 运行状态代码	○	○	-1	-	-	N	R	-1
SR858	CANopen DS301 从站编号 29 运行状态代码	○	○	-1	-	-	N	R	-1
SR859	CANopen DS301 从站编号 30 运行状态代码	○	○	-1	-	-	N	R	-1
SR860	CANopen DS301 从站编号 31 运行状态代码	○	○	-1	-	-	N	R	-1
SR861	CANopen DS301 从站编号 32 运行状态代码	○	○	-1	-	-	N	R	-1
SR862	CANopen DS301 从站编号 33 运行状态代码	○	○	-1	-	-	N	R	-1
SR863	CANopen DS301 从站编号 34 运行状态代码	○	○	-1	-	-	N	R	-1
SR864	CANopen DS301 从站编号 35 运行状态代码	○	○	-1	-	-	N	R	-1
SR865	CANopen DS301 从站编号 36 运行状态代码	○	○	-1	-	-	N	R	-1
SR866	CANopen DS301 从站编号 37 运行状态代码	○	○	-1	-	-	N	R	-1
SR867	CANopen DS301 从站编号 38 运行状态代码	○	○	-1	-	-	N	R	-1
SR868	CANopen DS301 从站编号 39 运行状态代码	○	○	-1	-	-	N	R	-1
SR869	CANopen DS301 从站编号 40 运行状态代码	○	○	-1	-	-	N	R	-1
SR870	CANopen DS301 从站编号 41 运行状态代码	○	○	-1	-	-	N	R	-1
SR871	CANopen DS301 从站编号 42 运行状态代码	○	○	-1	-	-	N	R	-1
SR872	CANopen DS301 从站编号 43 运行状态代码	○	○	-1	-	-	N	R	-1
SR873	CANopen DS301 从站编号 44 运行状态代码	○	○	-1	-	-	N	R	-1

特殊 寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR874	CANopen DS301 从站编号 45 运行状态代码	○	○	-1	-	-	N	R	-1
SR875	CANopen DS301 从站编号 46 运行状态代码	○	○	-1	-	-	N	R	-1
SR876	CANopen DS301 从站编号 47 运行状态代码	○	○	-1	-	-	N	R	-1
SR877	CANopen DS301 从站编号 48 运行状态代码	○	○	-1	-	-	N	R	-1
SR878	CANopen DS301 从站编号 49 运行状态代码	○	○	-1	-	-	N	R	-1
SR879	CANopen DS301 从站编号 50 运行状态代码	○	○	-1	-	-	N	R	-1
SR880	CANopen DS301 从站编号 51 运行状态代码	○	○	-1	-	-	N	R	-1
SR881	CANopen DS301 从站编号 52 运行状态代码	○	○	-1	-	-	N	R	-1
SR882	CANopen DS301 从站编号 53 运行状态代码	○	○	-1	-	-	N	R	-1
SR883	CANopen DS301 从站编号 54 运行状态代码	○	○	-1	-	-	N	R	-1
SR884	CANopen DS301 从站编号 55 运行状态代码	○	○	-1	-	-	N	R	-1
SR885	CANopen DS301 从站编号 56 运行状态代码	○	○	-1	-	-	N	R	-1
SR886	CANopen DS301 从站编号 57 运行状态代码	○	○	-1	-	-	N	R	-1
SR887	CANopen DS301 从站编号 58 运行状态代码	○	○	-1	-	-	N	R	-1
SR888	CANopen DS301 从站编号 59 运行状态代码	○	○	-1	-	-	N	R	-1
SR889	CANopen DS301 从站编号 60 运行状态代码	○	○	-1	-	-	N	R	-1
SR890	CANopen DS301 从站编号 61 运行状态代码	○	○	-1	-	-	N	R	-1
SR891	CANopen DS301 从站编号 62 运行状态代码	○	○	-1	-	-	N	R	-1
SR892	CANopen DS301 从站编号 63 运行状态代码	○	○	-1	-	-	N	R	-1
SR893	CANopen DS301 从站编号 64 运行状态代码	○	○	-1	-	-	N	R	-1
SR900 SR901	数据记录器已取样笔数（32 位）	○	○	0	-	-	N	R	0
SR902	数据记录器与储存卡动作代码。需搭配 SM456 启动。	○	○	0	-	-	N	R/W	0
*SR1000 *SR1001	以太网络 IP 地址（32 位）	○	○	-	-	-	H	R/W	0
*SR1002 *SR1003	以太网络网络屏蔽地址（32 位）	○	○	-	-	-	H	R/W	0
*SR1004 *SR1005	以太网络网关地址（32 位）	○	○	-	-	-	H	R/W	0
*SR1006	TCP 保持连接时间（sec.）	○	○	-	-	-	H	R/W	30
*SR1007	以太网络传输速度	○	○	0	-	-	N	R	0
*SR1009	TCP 目前联机数	○	○	0	-	-	N	R	0
*SR1010	TCP 重传时间（ms）	○	○	-	-	-	N	R/W	20
*SR1011	MODBUS/TCP Server 联机数	○	○	0	-	-	N	R	0
*SR1012	MODBUS/TCP Client 联机数	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR1013	EtherNet/IP Adapter 联机数	○	○	0	-	-	N	R	0
*SR1014	EtherNet/IP Scanner 联机数	○	○	0	-	-	N	R	0
*SR1015	MODBUS/TCP Server 报文接收等待逾时时间（ms）	○	○	10	-	-	N	R/W	10
*SR1016	MODBUS/TCP Client 成功建立联机后 ,延迟开始发送的等待时间（ms）（0~1000）	○	○	0	-	-	N	R/W	0
*SR1017	每个扫描周期最多可处理 Ethernet 的报文数量	○	○	-	-	-	Y	R/W	80
*SR1018	最近一次扫描周期处理的 Ethernet 报文数	○	○	0	-	-	N	R	0
*SR1019	Ethernet 缓冲区满载而丢掉的 Ethernet 报文累积数量	○	○	0	-	-	N	R/W	0
SR1020	EtherNet/IP 联机 1 状态使用	○	○	0	-	-	N	R	0
SR1021	EtherNet/IP 联机 2 状态使用	○	○	0	-	-	N	R	0
SR1022	EtherNet/IP 联机 3 状态使用	○	○	0	-	-	N	R	0
SR1023	EtherNet/IP 联机 4 状态使用	○	○	0	-	-	N	R	0
SR1024	EtherNet/IP 联机 5 状态使用	○	○	0	-	-	N	R	0
SR1025	EtherNet/IP 联机 6 状态使用	○	○	0	-	-	N	R	0
SR1026	EtherNet/IP 联机 7 状态使用	○	○	0	-	-	N	R	0
SR1027	EtherNet/IP 联机 8 状态使用	○	○	0	-	-	N	R	0
SR1028	EtherNet/IP 联机 9 状态使用	○	○	0	-	-	N	R	0
SR1029	EtherNet/IP 联机 10 状态使用	○	○	0	-	-	N	R	0
SR1030	EtherNet/IP 联机 11 状态使用	○	○	0	-	-	N	R	0
SR1031	EtherNet/IP 联机 12 状态使用	○	○	0	-	-	N	R	0
SR1032	EtherNet/IP 联机 13 状态使用	○	○	0	-	-	N	R	0
SR1033	EtherNet/IP 联机 14 状态使用	○	○	0	-	-	N	R	0
SR1034	EtherNet/IP 联机 15 状态使用	○	○	0	-	-	N	R	0
SR1035	EtherNet/IP 联机 16 状态使用	○	○	0	-	-	N	R	0
SR1036	EtherNet/IP 联机 17 状态使用	○	○	0	-	-	N	R	0
SR1037	EtherNet/IP 联机 18 状态使用	○	○	0	-	-	N	R	0
SR1038	EtherNet/IP 联机 19 状态使用	○	○	0	-	-	N	R	0
SR1039	EtherNet/IP 联机 20 状态使用	○	○	0	-	-	N	R	0
SR1040	EtherNet/IP 联机 21 状态使用	○	○	0	-	-	N	R	0
SR1041	EtherNet/IP 联机 22 状态使用	○	○	0	-	-	N	R	0
SR1042	EtherNet/IP 联机 23 状态使用	○	○	0	-	-	N	R	0
SR1043	EtherNet/IP 联机 24 状态使用	○	○	0	-	-	N	R	0
SR1044	EtherNet/IP 联机 25 状态使用	○	○	0	-	-	N	R	0
SR1045	EtherNet/IP 联机 26 状态使用	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR1046	EtherNet/IP 联机 27 状态使用	○	○	0	-	-	N	R	0
SR1047	EtherNet/IP 联机 28 状态使用	○	○	0	-	-	N	R	0
SR1048	EtherNet/IP 联机 29 状态使用	○	○	0	-	-	N	R	0
SR1049	EtherNet/IP 联机 30 状态使用	○	○	0	-	-	N	R	0
SR1050	EtherNet/IP 联机 31 状态使用	○	○	0	-	-	N	R	0
SR1051	EtherNet/IP 联机 32 状态使用	○	○	0	-	-	N	R	0
SR1052	EtherNet/IP 联机 1 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1053	EtherNet/IP 联机 2 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1054	EtherNet/IP 联机 3 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1055	EtherNet/IP 联机 4 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1056	EtherNet/IP 联机 5 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1057	EtherNet/IP 联机 6 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1058	EtherNet/IP 联机 7 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1059	EtherNet/IP 联机 8 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1060	EtherNet/IP 联机 9 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1061	EtherNet/IP 联机 10 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1062	EtherNet/IP 联机 11 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1063	EtherNet/IP 联机 12 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1064	EtherNet/IP 联机 13 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1065	EtherNet/IP 联机 14 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1066	EtherNet/IP 联机 15 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1067	EtherNet/IP 联机 16 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1068	EtherNet/IP 联机 17 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1069	EtherNet/IP 联机 18 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1070	EtherNet/IP 联机 19 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1071	EtherNet/IP 联机 20 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1072	EtherNet/IP 联机 21 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1073	EtherNet/IP 联机 22 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1074	EtherNet/IP 联机 23 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1075	EtherNet/IP 联机 24 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1076	EtherNet/IP 联机 25 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1077	EtherNet/IP 联机 26 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1078	EtherNet/IP 联机 27 更新时间 (单位：ms)	○	○	0	-	-	N	R	0
SR1079	EtherNet/IP 联机 28 更新时间 (单位：ms)	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR1080	EtherNet/IP 联机 29 更新时间（单位：ms）	○	○	0	-	-	N	R	0
SR1081	EtherNet/IP 联机 30 更新时间（单位：ms）	○	○	0	-	-	N	R	0
SR1082	EtherNet/IP 联机 31 更新时间（单位：ms）	○	○	0	-	-	N	R	0
SR1083	EtherNet/IP 联机 32 更新时间（单位：ms）	○	○	0	-	-	N	R	0
SR1092	MODBUS TCP 从站通讯端口号（主机为从站），V1.10.20（含）以上支持	○	○	502	-	-	N	R/W	502
SR1093	MODBUS TCP 数据交换表通讯端口号(主机为主站)，V1.10.20（含）以上支持	○	○	502	-	-	N	R/W	502
SR1100	输入封包计数器（高 16 位）	○	○	0	-	-	N	R	0
SR1101	输入封包计数器（低 16 位）								
SR1102	输入字节计数器（高 16 位）	○	○	0	-	-	N	R	0
SR1103	输入字节计数器（低 16 位）								
SR1104	输出封包计数器（高 16 位）	○	○	0	-	-	N	R	0
SR1105	输出封包计数器（低 16 位）								
SR1106	输出字节计数器（高 16 位）	○	○	0	-	-	N	R	0
SR1107	输出字节计数器（低 16 位）								
SR1110	变更 NTP 参数之变更代码（需在 NTP 启动状况下，并且输入 16#1234 变更命令代码，系统变更完成后将自动清除代码）	○	○	0	0	-	N	R/W	0
SR1111	NTP 服务器 IP 位（高 16 位）	○	○	0	0	-	N	R/W	0
SR1112	NTP 服务器 IP 位（低 16 位）	○	○	0	0	-	N	R/W	0
SR1113	NTP 更新周期时间（单位：min.）	○	○	0	0	-	N	R/W	0
SR1114	NTP 时区（单位: Hour）	○	○	0	0	-	N	R/W	0
SR1116	Email 计数器	○	○	0	-	-	N	R	0
SR1117	Email 错误计数器	○	○	0	-	-	N	R	0
*SR1120	Ethernet 数据交换联机 1 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1121	Ethernet 数据交换联机 2 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1122	Ethernet 数据交换联机 3 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1123	Ethernet 数据交换联机 4 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1124	Ethernet 数据交换联机 5 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1125	Ethernet 数据交换联机 6 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1126	Ethernet 数据交换联机 7 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1127	Ethernet 数据交换联机 8 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1128	Ethernet 数据交换联机 9 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR1129	Ethernet 数据交换联机 10 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1130	Ethernet 数据交换联机 11 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1131	Ethernet 数据交换联机 12 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1132	Ethernet 数据交换联机 13 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1133	Ethernet 数据交换联机 14 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1134	Ethernet 数据交换联机 15 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1135	Ethernet 数据交换联机 16 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1136	Ethernet 数据交换联机 17 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1137	Ethernet 数据交换联机 18 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1138	Ethernet 数据交换联机 19 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1139	Ethernet 数据交换联机 20 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1140	Ethernet 数据交换联机 21 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1141	Ethernet 数据交换联机 22 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1142	Ethernet 数据交换联机 23 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1143	Ethernet 数据交换联机 24 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1144	Ethernet 数据交换联机 25 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1145	Ethernet 数据交换联机 26 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1146	Ethernet 数据交换联机 27 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1147	Ethernet 数据交换联机 28 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1148	Ethernet 数据交换联机 29 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1149	Ethernet 数据交换联机 30 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1150	Ethernet 数据交换联机 31 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1151	Ethernet 数据交换联机 32 的实际连接时间(单位:ms)	○	○	0	-	-	N	R	0
*SR1152	Ethernet 数据交换联机 1 的错误代码	○	○	0	-	-	N	R	0
*SR1153	Ethernet 数据交换联机 2 的错误代码	○	○	0	-	-	N	R	0
*SR1154	Ethernet 数据交换联机 3 的错误代码	○	○	0	-	-	N	R	0
*SR1155	Ethernet 数据交换联机 4 的错误代码	○	○	0	-	-	N	R	0
*SR1156	Ethernet 数据交换联机 5 的错误代码	○	○	0	-	-	N	R	0
*SR1157	Ethernet 数据交换联机 6 的错误代码	○	○	0	-	-	N	R	0
*SR1158	Ethernet 数据交换联机 7 的错误代码	○	○	0	-	-	N	R	0
*SR1159	Ethernet 数据交换联机 8 的错误代码	○	○	0	-	-	N	R	0
*SR1160	Ethernet 数据交换联机 9 的错误代码	○	○	0	-	-	N	R	0
*SR1161	Ethernet 数据交换联机 10 的错误代码	○	○	0	-	-	N	R	0
*SR1162	Ethernet 数据交换联机 11 的错误代码	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR1163	Ethernet 数据交换联机 12 的错误代码	○	○	0	-	-	N	R	0
*SR1164	Ethernet 数据交换联机 13 的错误代码	○	○	0	-	-	N	R	0
*SR1165	Ethernet 数据交换联机 14 的错误代码	○	○	0	-	-	N	R	0
*SR1166	Ethernet 数据交换联机 15 的错误代码	○	○	0	-	-	N	R	0
*SR1167	Ethernet 数据交换联机 16 的错误代码	○	○	0	-	-	N	R	0
*SR1168	Ethernet 数据交换联机 17 的错误代码	○	○	0	-	-	N	R	0
*SR1169	Ethernet 数据交换联机 18 的错误代码	○	○	0	-	-	N	R	0
*SR1170	Ethernet 数据交换联机 19 的错误代码	○	○	0	-	-	N	R	0
*SR1171	Ethernet 数据交换联机 20 的错误代码	○	○	0	-	-	N	R	0
*SR1172	Ethernet 数据交换联机 21 的错误代码	○	○	0	-	-	N	R	0
*SR1173	Ethernet 数据交换联机 22 的错误代码	○	○	0	-	-	N	R	0
*SR1174	Ethernet 数据交换联机 23 的错误代码	○	○	0	-	-	N	R	0
*SR1175	Ethernet 数据交换联机 24 的错误代码	○	○	0	-	-	N	R	0
*SR1176	Ethernet 数据交换联机 25 的错误代码	○	○	0	-	-	N	R	0
*SR1177	Ethernet 数据交换联机 26 的错误代码	○	○	0	-	-	N	R	0
*SR1178	Ethernet 数据交换联机 27 之错误代码	○	○	0	-	-	N	R	0
*SR1179	Ethernet 数据交换联机 28 之错误代码	○	○	0	-	-	N	R	0
*SR1180	Ethernet 数据交换联机 29 之错误代码	○	○	0	-	-	N	R	0
*SR1181	Ethernet 数据交换联机 30 之错误代码	○	○	0	-	-	N	R	0
*SR1182	Ethernet 数据交换联机 31 之错误代码	○	○	0	-	-	N	R	0
*SR1183	Ethernet 数据交换联机 32 之错误代码	○	○	0	-	-	N	R	0
SR1191	台达驱动器 ECAT 通讯站号 9 之伺服 DO 状态	○	-	0	0	-	N	R	0
SR1192	台达驱动器 ECAT 通讯站号 10 之伺服 DO 状态	○	-	0	0	-	N	R	0
SR1193	台达驱动器 ECAT 通讯站号 11 之伺服 DO 状态	○	-	0	0	-	N	R	0
SR1194	台达驱动器 ECAT 通讯站号 12 之伺服 DO 状态	○	-	0	0	-	N	R	0
SR1195	台达驱动器 ECAT 通讯站号 13 之伺服 DO 状态	○	-	0	0	-	N	R	0
SR1196	台达驱动器 ECAT 通讯站号 14 之伺服 DO 状态	○	-	0	0	-	N	R	0
SR1197	台达驱动器 ECAT 通讯站号 15 之伺服 DO 状态	○	-	0	0	-	N	R	0
SR1198	台达驱动器 ECAT 通讯站号 16 之伺服 DO 状态	○	-	0	0	-	N	R	0
SR1201 SR1202	台达驱动器 ECAT 通讯站号 9 之伺服现在位置（32 位）	○	-	0	0	-	N	R	0
SR1203 SR1204	台达驱动器 ECAT 通讯站号 10 之伺服现在位置（32 位）	○	-	0	0	-	N	R	0
SR1205	台达驱动器 ECAT 通讯站号 11 之伺服现在位置（32	○	-	0	0	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR1206	位）								
SR1207 SR1208	台达驱动器 ECAT 通讯站号 12 之伺服现在位置（32 位）	○	-	0	0	-	N	R	0
SR1209 SR1210	台达驱动器 ECAT 通讯站号 13 之伺服现在位置（32 位）	○	-	0	0	-	N	R	0
SR1211 SR1212	台达驱动器 ECAT 通讯站号 14 之伺服现在位置（32 位）	○	-	0	0	-	N	R	0
SR1213 SR1214	台达驱动器 ECAT 通讯站号 15 之伺服现在位置（32 位）	○	-	0	0	-	N	R	0
SR1215 SR1216	台达驱动器 ECAT 通讯站号 16 之伺服现在位置（32 位）	○	-	0	0	-	N	R	0
SR1221 SR1222	台达驱动器 ECAT 通讯站号 9 之伺服目标位置（32 位）	○	-	0	0	-	N	R	0
SR1223 SR1224	台达驱动器 ECAT 通讯站号 10 之伺服目标位置（32 位）	○	-	0	0	-	N	R	0
SR1225 SR1226	台达驱动器 ECAT 通讯站号 11 之伺服目标位置（32 位）	○	-	0	0	-	N	R	0
SR1227 SR1228	台达驱动器 ECAT 通讯站号 12 之伺服目标位置（32 位）	○	-	0	0	-	N	R	0
SR1229 SR1230	台达驱动器 ECAT 通讯站号 13 之伺服目标位置（32 位）	○	-	0	0	-	N	R	0
SR1231 SR1232	台达驱动器 ECAT 通讯站号 14 之伺服目标位置（32 位）	○	-	0	0	-	N	R	0
SR1233 SR1234	台达驱动器 ECAT 通讯站号 15 之伺服目标位置（32 位）	○	-	0	0	-	N	R	0
SR1235 SR1236	台达驱动器 ECAT 通讯站号 16 之伺服目标位置（32 位）	○	-	0	0	-	N	R	0
SR1241 SR1242	台达驱动器 ECAT 通讯站号 9 之伺服读取自定义数值（32 位）	○	-	0	0	-	N	R	0
SR1243 SR1244	台达驱动器 ECAT 通讯站号 10 之伺服读取自定义数值（32 位）	○	-	0	0	-	N	R	0
SR1245 SR1246	台达驱动器 ECAT 通讯站号 11 之伺服读取自定义数值（32 位）	○	-	0	0	-	N	R	0
SR1247 SR1248	台达驱动器 ECAT 通讯站号 12 之伺服读取自定义数	○	-	0	0	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
	值（32 位）								
SR1249 SR1250	台达驱动器 ECAT 通讯站号 13 之伺服读取自定义数值（32 位）	○	-	0	0	-	N	R	0
SR1251 SR1252	台达驱动器 ECAT 通讯站号 14 之伺服读取自定义数值（32 位）	○	-	0	0	-	N	R	0
SR1253 SR1254	台达驱动器 ECAT 通讯站号 15 之伺服读取自定义数值（32 位）	○	-	0	0	-	N	R	0
SR1255 SR1256	台达驱动器 ECAT 通讯站号 16 之伺服读取自定义数值（32 位）	○	-	0	0	-	N	R	0
SR1261	台达驱动器 ECAT 通讯站号 9 之伺服定位完成允许范围	○	-	0	0	-	N	R/W	0
SR1262	台达驱动器 ECAT 通讯站号 10 之伺服定位完成允许范围	○	-	0	0	-	N	R/W	0
SR1263	台达驱动器 ECAT 通讯站号 11 之伺服定位完成允许范围	○	-	0	0	-	N	R/W	0
SR1264	台达驱动器 ECAT 通讯站号 12 之伺服定位完成允许范围	○	-	0	0	-	N	R/W	0
SR1265	台达驱动器 ECAT 通讯站号 13 之伺服定位完成允许范围	○	-	0	0	-	N	R/W	0
SR1266	台达驱动器 ECAT 通讯站号 14 之伺服定位完成允许范围	○	-	0	0	-	N	R/W	0
SR1267	台达驱动器 ECAT 通讯站号 15 之伺服定位完成允许范围	○	-	0	0	-	N	R/W	0
SR1268	台达驱动器 ECAT 通讯站号 16 之伺服定位完成允许范围	○	-	0	0	-	N	R/W	0
SR1291	定位规划表输出指令（TPO），轴 1 跳转特定规划表编号，搭配 SM591 使用。	○	○	0	0	-	N	R/W	0
SR1292	定位规划表输出指令（TPO），轴 2 跳转特定规划表编号，搭配 SM592 使用。	○	○	0	0	-	N	R/W	0
SR1293	定位规划表输出指令（TPO），轴 3 跳转特定规划表编号，搭配 SM593 使用。	○	○	0	0	-	N	R/W	0
SR1294	定位规划表输出指令（TPO），轴 4 跳转特定规划表编号，搭配 SM594 使用。	○	○	0	0	-	N	R/W	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR1295	定位规划表输出指令（TPO），轴 5 跳转特定规划表编号，搭配 SM595 使用。	○	○	0	0	-	N	R/W	0
SR1296	定位规划表输出指令（TPO），轴 6 跳转特定规划表编号，搭配 SM596 使用。	○	○	0	0	-	N	R/W	0
*SR1312	RTU-EN01 联机 1 通讯代码	○	○	0	-	-	N	R	0
*SR1313	RTU-EN01 联机 2 通讯代码	○	○	0	-	-	N	R	0
*SR1314	RTU-EN01 联机 3 通讯代码	○	○	0	-	-	N	R	0
*SR1315	RTU-EN01 联机 4 通讯代码	○	○	0	-	-	N	R	0
SR1318	Socket 输入计数器	○	○	0	-	-	N	R	0
SR1319	Socket 输出计数器	○	○	0	-	-	N	R	0
SR1320	Socket 错误计数器	○	○	0	-	-	N	R	0
*SR1335	COM1 数据交换实际执行周期时间（单位:ms）	○	○	0	-	-	N	R	0
*SR1336	COM1 数据交换已启动联机编号	○	○	0	-	-	N	R	0
*SR1340	COM1 数据交换联机 1 错误代码	○	○	0	-	-	N	R	0
*SR1341	COM1 数据交换联机 2 错误代码	○	○	0	-	-	N	R	0
*SR1342	COM1 数据交换联机 3 错误代码	○	○	0	-	-	N	R	0
*SR1343	COM1 数据交换联机 4 错误代码	○	○	0	-	-	N	R	0
*SR1344	COM1 数据交换联机 5 错误代码	○	○	0	-	-	N	R	0
*SR1345	COM1 数据交换联机 6 错误代码	○	○	0	-	-	N	R	0
*SR1346	COM1 数据交换联机 7 错误代码	○	○	0	-	-	N	R	0
*SR1347	COM1 数据交换联机 8 错误代码	○	○	0	-	-	N	R	0
*SR1348	COM1 数据交换联机 9 错误代码	○	○	0	-	-	N	R	0
*SR1349	COM1 数据交换联机 10 错误代码	○	○	0	-	-	N	R	0
*SR1350	COM1 数据交换联机 11 错误代码	○	○	0	-	-	N	R	0
*SR1351	COM1 数据交换联机 12 错误代码	○	○	0	-	-	N	R	0
*SR1352	COM1 数据交换联机 13 错误代码	○	○	0	-	-	N	R	0
*SR1353	COM1 数据交换联机 14 错误代码	○	○	0	-	-	N	R	0
*SR1354	COM1 数据交换联机 15 错误代码	○	○	0	-	-	N	R	0
*SR1355	COM1 数据交换联机 16 错误代码	○	○	0	-	-	N	R	0
*SR1356	COM1 数据交换联机 17 错误代码	○	○	0	-	-	N	R	0
*SR1357	COM1 数据交换联机 18 错误代码	○	○	0	-	-	N	R	0
*SR1358	COM1 数据交换联机 19 错误代码	○	○	0	-	-	N	R	0
*SR1359	COM1 数据交换联机 20 错误代码	○	○	0	-	-	N	R	0
*SR1360	COM1 数据交换联机 21 错误代码	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR1361	COM1 数据交换联机 22 错误代码	○	○	0	-	-	N	R	0
*SR1362	COM1 数据交换联机 23 错误代码	○	○	0	-	-	N	R	0
*SR1363	COM1 数据交换联机 24 错误代码	○	○	0	-	-	N	R	0
*SR1364	COM1 数据交换联机 25 错误代码	○	○	0	-	-	N	R	0
*SR1365	COM1 数据交换联机 26 错误代码	○	○	0	-	-	N	R	0
*SR1366	COM1 数据交换联机 27 错误代码	○	○	0	-	-	N	R	0
*SR1367	COM1 数据交换联机 28 错误代码	○	○	0	-	-	N	R	0
*SR1368	COM1 数据交换联机 29 错误代码	○	○	0	-	-	N	R	0
*SR1369	COM1 数据交换联机 30 错误代码	○	○	0	-	-	N	R	0
*SR1370	COM1 数据交换联机 31 错误代码	○	○	0	-	-	N	R	0
*SR1371	COM1 数据交换联机 32 错误代码	○	○	0	-	-	N	R	0
*SR1375	COM2 数据交换实际执行周期时间（单位:ms）	○	○	0	-	-	N	R	0
*SR1376	COM2 数据交换正在通讯之联机编号	○	○	0	-	-	N	R	0
*SR1380	COM2 数据交换联机 1 错误代码	○	○	0	-	-	N	R	0
SR1381	COM2 数据交换联机 2 错误代码	○	○	0	-	-	N	R	0
SR1382	COM2 数据交换联机 3 错误代码	○	○	0	-	-	N	R	0
SR1383	COM2 数据交换联机 4 错误代码	○	○	0	-	-	N	R	0
SR1384	COM2 数据交换联机 5 错误代码	○	○	0	-	-	N	R	0
SR1385	COM2 数据交换联机 6 错误代码	○	○	0	-	-	N	R	0
SR1386	COM2 数据交换联机 7 错误代码	○	○	0	-	-	N	R	0
SR1387	COM2 数据交换联机 8 错误代码	○	○	0	-	-	N	R	0
SR1388	COM2 数据交换联机 9 错误代码	○	○	0	-	-	N	R	0
SR1389	COM2 数据交换联机 10 错误代码	○	○	0	-	-	N	R	0
SR1390	COM2 数据交换联机 11 错误代码	○	○	0	-	-	N	R	0
SR1391	COM2 数据交换联机 12 错误代码	○	○	0	-	-	N	R	0
SR1392	COM2 数据交换联机 13 错误代码	○	○	0	-	-	N	R	0
SR1393	COM2 数据交换联机 14 错误代码	○	○	0	-	-	N	R	0
SR1394	COM2 数据交换联机 15 错误代码	○	○	0	-	-	N	R	0
SR1395	COM2 数据交换联机 16 错误代码	○	○	0	-	-	N	R	0
SR1396	COM2 数据交换联机 17 错误代码	○	○	0	-	-	N	R	0
SR1397	COM2 数据交换联机 18 错误代码	○	○	0	-	-	N	R	0
SR1398	COM2 数据交换联机 19 错误代码	○	○	0	-	-	N	R	0
SR1399	COM2 数据交换联机 20 错误代码	○	○	0	-	-	N	R	0
SR1400	COM2 数据交换联机 21 错误代码	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
SR1401	COM2 数据交换联机 22 错误代码	○	○	0	-	-	N	R	0
SR1402	COM2 数据交换联机 23 错误代码	○	○	0	-	-	N	R	0
SR1403	COM2 数据交换联机 24 错误代码	○	○	0	-	-	N	R	0
SR1404	COM2 数据交换联机 25 错误代码	○	○	0	-	-	N	R	0
SR1405	COM2 数据交换联机 26 错误代码	○	○	0	-	-	N	R	0
SR1406	COM2 数据交换联机 27 错误代码	○	○	0	-	-	N	R	0
SR1407	COM2 数据交换联机 28 错误代码	○	○	0	-	-	N	R	0
SR1408	COM2 数据交换联机 29 错误代码	○	○	0	-	-	N	R	0
SR1409	COM2 数据交换联机 30 错误代码	○	○	0	-	-	N	R	0
SR1410	COM2 数据交换联机 31 错误代码	○	○	0	-	-	N	R	0
SR1411	COM2 数据交换联机 32 错误代码	○	○	0	-	-	N	R	0
SR1430	OPC UA Server 联机数	○	-	0	-	-	N	R	0
SR1431	OPC UA Client 联机数	○	-	0	-	-	N	R	0
*SR1435	序列通讯卡 1 数据交换实际执行周期时间（单位:ms）	○	-	0	-	-	N	R	0
*SR1436	序列通讯卡 1 目前执行数据交换联机编号	○	-	0	-	-	N	R	0
*SR1440	序列通讯卡 1 数据交换联机 1 错误代码	○	-	0	-	-	N	R	0
*SR1441	序列通讯卡 1 数据交换联机 2 错误代码	○	-	0	-	-	N	R	0
*SR1442	序列通讯卡 1 数据交换联机 3 错误代码	○	-	0	-	-	N	R	0
*SR1443	序列通讯卡 1 数据交换联机 4 错误代码	○	-	0	-	-	N	R	0
*SR1444	序列通讯卡 1 数据交换联机 5 错误代码	○	-	0	-	-	N	R	0
*SR1445	序列通讯卡 1 数据交换联机 6 错误代码	○	-	0	-	-	N	R	0
*SR1446	序列通讯卡 1 数据交换联机 7 错误代码	○	-	0	-	-	N	R	0
*SR1447	序列通讯卡 1 数据交换联机 8 错误代码	○	-	0	-	-	N	R	0
*SR1448	序列通讯卡 1 数据交换联机 9 错误代码	○	-	0	-	-	N	R	0
*SR1449	序列通讯卡 1 数据交换联机 10 错误代码	○	-	0	-	-	N	R	0
*SR1450	序列通讯卡 1 数据交换联机 11 错误代码	○	-	0	-	-	N	R	0
*SR1451	序列通讯卡 1 数据交换联机 12 错误代码	○	-	0	-	-	N	R	0
*SR1452	序列通讯卡 1 数据交换联机 13 错误代码	○	-	0	-	-	N	R	0
*SR1453	序列通讯卡 1 数据交换联机 14 错误代码	○	-	0	-	-	N	R	0
*SR1454	序列通讯卡 1 数据交换联机 15 错误代码	○	-	0	-	-	N	R	0
*SR1455	序列通讯卡 1 数据交换联机 16 错误代码	○	-	0	-	-	N	R	0
*SR1456	序列通讯卡 1 数据交换联机 17 错误代码	○	-	0	-	-	N	R	0
*SR1457	序列通讯卡 1 数据交换联机 18 错误代码	○	-	0	-	-	N	R	0
*SR1458	序列通讯卡 1 数据交换联机 19 错误代码	○	-	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR1459	序列通讯卡 1 数据交换联机 20 错误代码	○	-	0	-	-	N	R	0
*SR1460	序列通讯卡 1 数据交换联机 21 错误代码	○	-	0	-	-	N	R	0
*SR1461	序列通讯卡 1 数据交换联机 22 错误代码	○	-	0	-	-	N	R	0
*SR1462	序列通讯卡 1 数据交换联机 23 错误代码	○	-	0	-	-	N	R	0
*SR1463	序列通讯卡 1 数据交换联机 24 错误代码	○	-	0	-	-	N	R	0
*SR1464	序列通讯卡 1 数据交换联机 25 错误代码	○	-	0	-	-	N	R	0
*SR1465	序列通讯卡 1 数据交换联机 26 错误代码	○	-	0	-	-	N	R	0
*SR1466	序列通讯卡 1 数据交换联机 27 错误代码	○	-	0	-	-	N	R	0
*SR1467	序列通讯卡 1 数据交换联机 28 错误代码	○	-	0	-	-	N	R	0
*SR1468	序列通讯卡 1 数据交换联机 29 错误代码	○	-	0	-	-	N	R	0
*SR1469	序列通讯卡 1 数据交换联机 30 错误代码	○	-	0	-	-	N	R	0
*SR1470	序列通讯卡 1 数据交换联机 31 错误代码	○	-	0	-	-	N	R	0
*SR1471	序列通讯卡 1 数据交换联机 32 错误代码	○	-	0	-	-	N	R	0
*SR1475	序列通讯卡 2 数据交换实际执行周期时间（单位:ms）	○	-	0	-	-	N	R	0
*SR1476	序列通讯卡 2 目前执行数据交换联机编号	○	-	0	-	-	N	R	0
*SR1480	序列通讯卡 2 数据交换联机 1 错误代码	○	-	0	-	-	N	R	0
*SR1481	序列通讯卡 2 数据交换联机 2 错误代码	○	-	0	-	-	N	R	0
*SR1482	序列通讯卡 2 数据交换联机 3 错误代码	○	-	0	-	-	N	R	0
*SR1483	序列通讯卡 2 数据交换联机 4 错误代码	○	-	0	-	-	N	R	0
*SR1484	序列通讯卡 2 数据交换联机 5 错误代码	○	-	0	-	-	N	R	0
*SR1485	序列通讯卡 2 数据交换联机 6 错误代码	○	-	0	-	-	N	R	0
*SR1486	序列通讯卡 2 数据交换联机 7 错误代码	○	-	0	-	-	N	R	0
*SR1487	序列通讯卡 2 数据交换联机 8 错误代码	○	-	0	-	-	N	R	0
*SR1488	序列通讯卡 2 数据交换联机 9 错误代码	○	-	0	-	-	N	R	0
*SR1489	序列通讯卡 2 数据交换联机 10 错误代码	○	-	0	-	-	N	R	0
*SR1490	序列通讯卡 2 数据交换联机 11 错误代码	○	-	0	-	-	N	R	0
*SR1491	序列通讯卡 2 数据交换联机 12 错误代码	○	-	0	-	-	N	R	0
*SR1492	序列通讯卡 2 数据交换联机 13 错误代码	○	-	0	-	-	N	R	0
*SR1493	序列通讯卡 2 数据交换联机 14 错误代码	○	-	0	-	-	N	R	0
*SR1494	序列通讯卡 2 数据交换联机 15 错误代码	○	-	0	-	-	N	R	0
*SR1495	序列通讯卡 2 数据交换联机 16 错误代码	○	-	0	-	-	N	R	0
*SR1496	序列通讯卡 2 数据交换联机 17 错误代码	○	-	0	-	-	N	R	0
*SR1497	序列通讯卡 2 数据交换联机 18 错误代码	○	-	0	-	-	N	R	0
*SR1498	序列通讯卡 2 数据交换联机 19 错误代码	○	-	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR1499	序列通讯卡 2 数据交换联机 20 错误代码	○	-	0	-	-	N	R	0
*SR1500	序列通讯卡 2 数据交换联机 21 错误代码	○	-	0	-	-	N	R	0
*SR1501	序列通讯卡 2 数据交换联机 22 错误代码	○	-	0	-	-	N	R	0
*SR1502	序列通讯卡 2 数据交换联机 23 错误代码	○	-	0	-	-	N	R	0
*SR1503	序列通讯卡 2 数据交换联机 24 错误代码	○	-	0	-	-	N	R	0
*SR1504	序列通讯卡 2 数据交换联机 25 错误代码	○	-	0	-	-	N	R	0
*SR1505	序列通讯卡 2 数据交换联机 26 错误代码	○	-	0	-	-	N	R	0
*SR1506	序列通讯卡 2 数据交换联机 27 错误代码	○	-	0	-	-	N	R	0
*SR1507	序列通讯卡 2 数据交换联机 28 错误代码	○	-	0	-	-	N	R	0
*SR1508	序列通讯卡 2 数据交换联机 29 错误代码	○	-	0	-	-	N	R	0
*SR1509	序列通讯卡 2 数据交换联机 30 错误代码	○	-	0	-	-	N	R	0
*SR1510	序列通讯卡 2 数据交换联机 31 错误代码	○	-	0	-	-	N	R	0
*SR1511	序列通讯卡 2 数据交换联机 32 错误代码	○	-	0	-	-	N	R	0
*SR1520	AS-FEN02 数据交换联机 1 之实际连接时间(单位:ms)	○	-	0	-	-	N	R	0
*SR1521	AS-FEN02 数据交换联机 2 之实际连接时间(单位:ms)	○	-	0	-	-	N	R	0
*SR1522	AS-FEN02 数据交换联机 3 之实际连接时间(单位:ms)	○	-	0	-	-	N	R	0
*SR1523	AS-FEN02 数据交换联机 4 之实际连接时间(单位:ms)	○	-	0	-	-	N	R	0
*SR1524	AS-FEN02 数据交换联机 5 之实际连接时间(单位:ms)	○	-	0	-	-	N	R	0
*SR1525	AS-FEN02 数据交换联机 6 之实际连接时间(单位:ms)	○	-	0	-	-	N	R	0
*SR1526	AS-FEN02 数据交换联机 7 之实际连接时间(单位:ms)	○	-	0	-	-	N	R	0
*SR1527	AS-FEN02 数据交换联机 8 之实际连接时间(单位:ms)	○	-	0	-	-	N	R	0
*SR1528	AS-FEN02 数据交换联机 1 之错误代码	○	-	0	-	-	N	R	0
*SR1529	AS-FEN02 数据交换联机 2 之错误代码	○	-	0	-	-	N	R	0
*SR1530	AS-FEN02 数据交换联机 3 之错误代码	○	-	0	-	-	N	R	0
*SR1531	AS-FEN02 数据交换联机 4 之错误代码	○	-	0	-	-	N	R	0
*SR1532	AS-FEN02 数据交换联机 5 之错误代码	○	-	0	-	-	N	R	0
*SR1533	AS-FEN02 数据交换联机 6 之错误代码	○	-	0	-	-	N	R	0
*SR1534	AS-FEN02 数据交换联机 7 之错误代码	○	-	0	-	-	N	R	0
*SR1535	AS-FEN02 数据交换联机 8 之错误代码	○	-	0	-	-	N	R	0
*SR1536	AS-FEN02/AS-FECAT TCP 目前联机数	○	-	0	-	-	N	R	0
*SR1537	AS-FEN02/AS-FECAT MODBUS/TCP Server 联机数	○	-	0	-	-	N	R	0
*SR1538	AS-FEN02 MODBUS/TCP Client 联机数	○	-	0	-	-	N	R	0
*SR1539	AS-FEN02 EtherNet/IP Adapter 联机数	○	-	0	-	-	N	R	0
*SR1559	主机已联机的远程模块数量 (不含远程右侧模块数量)	○	○	0	-	-	N	R	0

特殊寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
*SR1560	主机右侧连接的模块数量	○	○	0	-	-	N	R	0
*SR1561	显示主机或远程模块右侧第 1 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1562	显示主机或远程模块右侧第 2 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1563	显示主机或远程模块右侧第 3 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1564	显示主机或远程模块右侧第 4 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1565	显示主机或远程模块右侧第 5 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1566	显示主机或远程模块右侧第 6 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1567	显示主机或远程模块右侧第 7 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1568	显示主机或远程模块右侧第 8 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1569	显示主机或远程模块右侧第 9 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1570	显示主机或远程模块右侧第 10 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1571	显示主机或远程模块右侧第 11 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1572	显示主机或远程模块右侧第 12 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1573	显示主机或远程模块右侧第 13 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1574	显示主机或远程模块右侧第 14 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1575	显示主机或远程模块右侧第 15 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1576	显示主机或远程模块右侧第 16 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1577	显示主机或远程模块右侧第 17 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1578	显示主机或远程模块右侧第 18 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1579	显示主机或远程模块右侧第 19 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1580	显示主机或远程模块右侧第 20 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1581	显示主机或远程模块右侧第 21 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1582	显示主机或远程模块右侧第 22 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1583	显示主机或远程模块右侧第 23 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1584	显示主机或远程模块右侧第 24 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1585	显示主机或远程模块右侧第 25 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1586	显示主机或远程模块右侧第 26 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1587	显示主机或远程模块右侧第 27 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1588	显示主机或远程模块右侧第 28 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1589	显示主机或远程模块右侧第 29 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1590	显示主机或远程模块右侧第 30 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1591	显示主机或远程模块右侧第 31 台模块的机种代码	○	○	0	-	-	N	R	0
*SR1592	显示主机或远程模块右侧第 32 台模块的机种代码	○	○	0	-	-	N	R	0
SR1600	参数还原执行代码	○	○	-	-	-	Y	R/W	0

特殊 寄存器SR	功能说明	AS300 系列	AS100/AS200 系列	OFF ⇩ ON	STOP ⇩ RUN	RUN ⇩ STOP	停电保持	属性	出厂设定
	（0：不执行（默认值） 1 :当 PLC 上电或 Stop➔Run 时，执行参数还原动作，直到执行完成时才清除为 0）								
SR1601	参数还原的目标驱动器编号设定 （SR1601.0~ SR1601.15 对应到目标驱动器编号 1~16） （OFF：不执行（默认值） ON：被指定执行参数还原之动作）	○	○	-	-	-	Y	R/W	0
SR1602	参数还原的目标驱动器编号设定 （SR1602.0~ SR1602.15 对应到目标驱动器编号 17~32） （编号 17~32 为保留区）	○	○	-	-	-	Y	R/W	0
SR1603	参数还原的错误总台数 （0：表示无错误发生 1~16：表示还原过程中的错误总台数）	○	○	0	-	-	N	R	0
SR1604	参数还原的最后一台错误编号 （0 ~ 255：记录最后发生错误驱动器 IP 最后后缀 （当 SM1161 错误标志为 ON 才有效））	○	○	0	-	-	N	R	0
SR1605	参数还原的最后一台错误代码	○	○	0	-	-	N	R	0
*SR1700	CSFO 指令 Y0.0 / Y0.1 轴追随落后补偿速度（Hz）	○	○	0	0	-	N	R/W	0
*SR1702	CSFO 指令 Y0.2 / Y0.3 轴追随落后补偿速度（Hz）	○	○	0	0	-	N	R/W	0
*SR1704	CSFO 指令 Y0.4 / Y0.5 轴追随落后补偿速度（Hz）	○	○	0	0	-	N	R/W	0
*SR1706	CSFO 指令 Y0.6 / Y0.7 轴追随落后补偿速度（Hz）	○	○	0	0	-	N	R/W	0
*SR1708	CSFO 指令 Y0.8 / Y0.9 轴追随落后补偿速度（Hz）	○	○	0	0	-	N	R/W	0
*SR1710	CSFO 指令 Y0.10 / Y0.11 轴追随落后补偿速度（Hz）	○	○	0	0	-	N	R/W	0

注 1：*之 SR 请参考 SM/SR 补充说明

注 2：SR 中所述之「序列通讯卡」系指 AS-F232、AS-F422 及 AS-F485。

2.2.15 特殊数据寄存器 SR 的更新时间

特殊数据寄存器	更新时间或时机
SR0~SR2	程序执行错误时
SR4~SR6	语法检查错误时
SR8~SR9	PLC程序扫描逾时
SR27	更新AS-F2AD/内建AD模拟数值，由系统设定与清除
SR28~SR31	高速输出指令发生重复输出点时
SR32	有错误发生时更新一次，-1表示无错误发生
SR36	用户设定，启动SM36执行完回存系统信息至储存卡功能后，系统自动清除
SR38	系统更新/用户清除
SR40~SR161	有错误发生时更新一次
SR162~SR163	产品制作完成时开始计时，每上电达1分钟即计数累加一次
SR166~SR171	系统更新
SR172~SR175	用户设定
SR176~SR179	依据HWCONFIG设定显示
SR180	重新上下电清除
SR182~SR183	上电时以HWCONFIG设定值为主，之后用户可修改
SR185	当有启动远程模块时，每次扫描一个循环后，系统自动更新周期时间
SR190~SR197	用户自行设定
SR198~SR199	上电时设定
SR201~SR213	上电时以HWCONFIG设定值为主，之后用户可修改
SR215~SR216	系统设定
SR217~SR218	上电时以HWCONFIG设定值为主，之后用户可修改
SR220~SR226	每一扫描周期更新一次
SR227~SR308	程序下载至PLC时更新
SR309~SR390	PLC状态改变时更新
SR391~SR397	每扫描周期更新
SR403~SR404	系统更新
SR405~SR406	上电时设定
SR407	每1s更新一次
SR408~SR416	每执行至End时更新
SR421~SR424	用户设定
SR440~SR443	上电时更新
SR444~SR451	上电时更新
SR453	错误发生时更新
SR460	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR462~SR466	用户设定

特殊数据寄存器	更新时间或时机
SR467	系统更新
SR468~SR469	上电时，由定位规划表设定为主，之后用户可修改
SR470	系统更新
SR472	用户设定
SR474	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR476~SR479	用户设定
SR480	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR482~SR486	用户设定
SR487	系统更新
SR488~SR489	上电时，由定位规划表设定为主，之后用户可修改
SR490	系统更新
SR492	用户设定
SR494	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR496~SR499	用户设定
SR500	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR502~SR506	用户设定
SR507	系统更新
SR508~SR509	上电时，由定位规划表设定为主，之后用户可修改
SR510	系统更新
SR512	用户设定
SR514	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR516~SR519	用户设定
SR520	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR522~SR526	用户设定
SR527	系统更新
SR528~SR529	上电时，由定位规划表设定为主，之后用户可修改
SR530	系统更新
SR532	用户设定
SR534	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR536~SR539	用户设定
SR540	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR542~SR546	用户设定
SR547	系统更新
SR548~SR549	上电时，由定位规划表设定为主，之后用户可修改
SR550	系统更新
SR552	用户设定

特殊数据寄存器	更新时间或时机
SR554	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR556~SR559	用户设定
SR560	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR562~SR566	用户设定
SR567	系统更新
SR568~SR569	上电时，由定位规划表设定为主，之后用户可修改
SR570	系统更新
SR572	用户设定
SR574	高速输出指令启动且被程序扫描执行时，由系统更新，若未启动输出时，才允许用户修改
SR576~SR579	用户设定
SR580~SR603	上电时，由HWCONFIG设定为主，之后用户可修改
SR610~SR621	输出指令执行时系统自动更新
SR623~SR634	执行EIX或DIX指令时，由系统更新 ON表示中断允许（Enable），OFF表示禁止（Disable）
SR640~SR651	用户设定
SR652~SR655	系統更新
SR656~SR657	系统设定/用户清除
SR658~SR726	系统更新
SR731~SR738	系统更新
SR741~SR748	系统更新
SR751-SR768	用户设定
SR771~SR778	系统更新
SR781~SR788	系统更新
SR791~SR806	系統更新
SR811~SR818	用户設定
SR820	由CANopen Builder设定
SR821	韧体更新
SR822	由HWCONFIG设定
SR823，SR824	系統更新
SR825~SR893	系统更新
SR900~SR901	系统更新
SR902	用户设定
SR1000~SR1006	1. 用户设定 2. 上电时或HWCONFIG下载之后，各更新一次
SR1007	系统更新
SR1009	系统更新
SR1010	用户设定及清除

特殊数据寄存器	更新时间或时机
SR1011~SR1014	系统更新
SR1015	用户设定
SR1016	用户设定
SR1017	用户设定
SR1018	每个扫描周期，由系统更新
SR1019	1. 每个扫描周期，由系统更新; 2. 用户清除
SR1020~SR1107	1. 每一次联机时更新一次 2. 每一扫描周期更新一次
SR1092-SR1093	用户设定
SR1110	用户设定/系统清除
SR1111~SR1114	用户设定
SR1116~SR1117	参数下载至PLC时更新
SR1120~SR1183	通讯完成时更新
SR1191~SR1198	系统更新
SR1201~SR1216	系统更新
SR1221~SR1236	系统更新
SR1241~SR1256	系统更新
SR1261~SR1268	用户设定
SR1291~SR1296	用户设定/系统清除
SR1312~SR1315	通讯时更新
SR1318~SR1320	参数下载至PLC时或电源开启时更新
SR1335~SR1336	数据交换启动后，每一次扫描周期更新
SR1340~SR1371	有错误发生时更新
SR1375~SR1376	数据交换启动后，每一次扫描周期更新
SR1380~SR1411	有错误发生时更新
SR1430~SR1431	系统更新
SR1435~SR1436	通讯时更新
SR1440~SR1471	通讯错误时更新
SR1475~SR1476	通讯时更新
SR1480~SR1511	通讯错误时更新
SR1520~SR1535	AS-FEN02通讯完成时更新
SR1536~SR1539	系统自动更新
SR1559~SR1592	上电时和远程模块断线/重连时由系统更新
SR1600~SR1602	用户设定
SR1603~SR1605	执行台达设备参数还原时，由系统更新

特殊数据寄存器	更新时间或时机
SR1700、SR1702 SR1704、SR1706 SR1708、SR1710	用户设定

2.2.16 SM / SR 补充说明

1. 扫描逾时定时器

- SM8/SR8

  当程序执行时发生扫描逾时 PLC ERROR 错误指示灯常亮，此时 SM8=ON。

  SR8 之内容值为 WDT 定时器 ON 之 STEP 地址。

2. 清除警告灯号

- SM22

  当 SM22 为 ON 的时候，会将错误记录清除，以及清除警告的灯号。

3. AD/DA 功能卡与主机内建说明

- 模拟输入

适用 AS300 主机搭配 AS-F2AD 功能卡		
项 目	电 压 输 入	电 流 输 入
额定输入范围	0V ~ +10V	4mA ~ 20mA
分辨率	12 位	11 位
额定数位转换范围	0 ~ 4000	0 ~ 2000
SR 编号（通道编号）	SR168（AS-F2AD 功能卡 1 信道 1） SR169（AS-F2AD 功能卡 1 信道 2） SR170（AS-F2AD 功能卡 2 信道 1） SR171（AS-F2AD 功能卡 2 信道 2）	

适用 AS218 主机内建 AD 功能			
项 目	电 压 输 入	电 流 输 入	电 流 输 入
额定输入范围	-10V ~ +10V	-20mA ~ 20mA	4mA ~ 20mA#1
分辨率	12 位	11 位	11 位
额定数位转换范围	-2000 ~ 2000	-1000 ~ 1000	0 ~ 1000
SR 编号（通道编号）	SR168（通道 1） SR169（通道 2）		

注#1：4~20mA 需搭配 AS218 主机韧体版本 V1.06.20 版（含）、AD/DA 韧体 V1.06.20 版（含），以及 ISPSoft V3.08 版（含）以上

- SM27/SR27 模拟输入报警标志和报警代码说明

  在 HWCONFIG 启动模拟输入报警功能，当模拟输入发生转换超过范围或是断线的情况，SM27 模拟输入报警标志会 ON，SR27 模拟输入报警代码会记录报警原因。

  AS300/AS200 V1.10.00（含）以上支持此功能。

  AS300：功能卡 1/2 AS-F2AD 设定

AS200：内建 AD 设定

代码	描述	排除方式
16#0040	功能卡 1 AS-F2AD/内建 AD 输入通道 1 数值转换超过范围	确认实际输入信号
16#0041	功能卡 1 AS-F2AD/内建 AD 输入通道 2 数值转换超过范围	确认实际输入信号
16#0042	功能卡 2 AS-F2AD 输入通道 1 数值转换超过范围	确认实际输入信号
16#0043	功能卡 2 AS-F2AD 输入通道 2 数值转换超过范围	确认实际输入信号
16#0044	功能卡 1 AS-F2AD/内建 AD 输入通道 1 电流输入 4mA~20mA 模式断线	确认接线
16#0045	功能卡 1 AS-F2AD/内建 AD 输入通道 2 电流输入 4mA~20mA 模式断线	确认接线
16#0046	功能卡 2 AS-F2AD 输入通道 1 电流输入 4mA~20mA 模式断线	确认接线
16#0047	功能卡 2 AS-F2AD 输入通道 2 电流输入 4mA~20mA 模式断线	确认接线

AS-F2AD 功能卡：

模拟输入模式	数值对应范围	备注
电压 0V ~ 10V	0 ~ 4000	1. 数值 > 4047 视为超过范围 2. 不支持断线检测

模拟输入模式	数值对应范围	备注
电流 4mA ~ 20mA	0 ~ 2000	1. 数值 < -24 或是 > 2023 视为超过范围 2. 数值 < -36 视为断线

AS218 内建模拟输入：

模拟输入模式	数值对应范围	备注
电压 -10V ~ 10V	-2000 ~ 2000	1. 数值 < -2024 或是 > 2023 视为超过范围 2. 不支持断线检测
电流 -20mA ~ 20mA	-1000 ~ 1000	1. 数值 < -1012 或是 > 1011 视为超过范围 2. 不支持断线检测
电流 4mA ~ 20mA	0 ~ 1000	1. 数值 < -12 或是 > 1011 视为超过范围 2. 数值 < -36 视为断线

- 模拟输出

适用 AS300 主机搭配 AS-F2DA 功能卡		
项 目	**电 压 输 出**	**电 流 输 出**
额定输出范围	0 ~ +10V	4 ~ 20mA
分辨率	12 位	12 位
额定数位值输入范围	0 ~ 4000	0 ~ 4000
SR 编号（通道编号）	SR172（AS-F2DA 功能卡 1 信道 1） SR173（AS-F2DA 功能卡 1 信道 2） SR174（AS-F2DA 功能卡 2 信道 1） SR175（AS-F2DA 功能卡 2 信道 2）	

适用 AS218 主机内建 DA 功能		
项 目	**电 压 输 出**	**电 流 输 出**
额定输出范围	-10V ~ +10V	0 ~ 20mA
分辨率	12 位	12 位
额定数位值输入范围	-2000 ~ 2000	0 ~ 4000
SR 编号（通道编号）	SR172（通道 1） SR173（通道 2）	

4. 万年历

- SM220，SR220~SR226，SR391~SR397

SM220：万年历时钟的±30 秒校正。当 SM220 OFF→ON 触发时作校正。

若万年历时钟秒为 0~29 秒时，分不动，秒归 0。

若万年历时钟秒为 30~59 秒时，分加 1，秒归 0。

万年历时钟 SR220~SR226，SR391~SR397 对应的功能以及内容值如下：

装置		功能名称	内容值范围
BCD 码	十进制		
SR220	SR391	年	0~99（公元右两位）
SR221	SR392	月	1~12
SR222	SR393	日	1~31
SR223	SR394	时	0~23
SR224	SR395	分	0~59
SR225	SR396	秒	0~59
SR226	SR397	星期	1~7

SR391 ~SR397 分别对应到 SR220~SR226，不同之处为 SR220~SR226 为 SR391~SR397 的 BCD 值，例如：12 月在 SR392 中为 12，在 SR221 中为 BCD 码 12。

万年历时钟的读取及校正请参考第 6 章万年历指令。

5. 通讯相关功能

- SM96~SM107，SM209~SM212，SR201~SR202，SR209~SR216

SR215，SR216 用来记录 PLC COM 通讯接口的代号， 代号代表的接口功能如下表：

代号	0	1	2	3	4	5	6
接口功能	未插卡	RS232	RS422	RS485	F2AD	F2DA	FCOPM
代号	7	8	9	10	11		
接口功能	FEN02	FPFN02	FOPC02	FFTP01	FECAT		

FCOPM/FEN02/FPFN02/FOPC02/FFTP01/FECAT 功能卡仅支持 PLC 功能卡 2

若 PLC COM 接口为 RS485、RS232 或 RS422 时，SR209 记录 PLC COM1 的通讯格式，SR212 记录 PLC COM2 的通讯格式。

SR177（序列通讯卡 1（COM11））、SR179（序列通讯卡 2（COM12））、SR209（COM1）、SR212（COM2）其通讯协议设定值如下表说明，其它有关 SM、SR 设定通讯的详细说明请参考第 6 章通讯指令。

b0	数据长度			7（内容值=0）	8（内容值=1）
b1 **b2**	同位			00 : 无（None）	
				01 : 奇同位（Odd）	
				10 : 偶同位（Even）	
b3	stop bits			1 bit（内容值=0）	2 bits（内容值=1）
b4 **b5** **b6** **b7**	0001	（H 1）	:	4800	
	0010	（H 2）	:	9600	
	0011	（H 3）	:	19200	
	0100	（H 4）	:	38400	
	0101	（H 5）	:	57600	
	0110	（H 6）	:	115200	

	0111	（H 7）	:	230400	
	1000	（H 8）	:	500000	
	1001	（H 9）	:	921000	
	1010	（16#A）	:	未定义	
	1011	（16#B）	:	未定义	
	1100	（16#C）	:	未定义	
	1101	（16#D）	:	未定义	
	1110	（16#E）	:	未定义	
	1111	（16#F）	:	用户自定义*1	
b8~b15	无定义（保留）				

*1：COM1 与 COM2 用户自定义之通讯速率，请由 ISPSoft 之硬件组态 HWCONFIG 或者 SR217/SR218 自行指定通讯速率值（基本单位 100bps）设定；序列通讯卡 1（COM11）或序列通讯卡 2（COM12）请自行搭配 SR187/SR188 设定，HWCONFIG 不提供设定。举例：设定 96 表示 9600bps。

*2：有关通讯相关标志与装置，请参考第 6.19.3 节通讯标志与寄存器使用

6. 清除装置内容标志

- SM203/SM204/SM205

装置编号	被清除的装置
SM203/SM204 清除非停电保持区域	非停电保持区装置如下： 非停电保持的 X，Y，S，M 接点、 非停电保持的 T，C，HC 内容值及接点、 非停电保持的 D、E 内容值 清除装置 WDT 不会动作。
SM205 清除停电保持区域	停电保持区装置如下： 停电保持的 T，C，HC 内容值及接点、 停电保持的 M 接点 停电保持的 D 内容值 清除装置 WDT 不会动作。

关于各装置的停电保持区请参考停电保持区的装置范围。

7. PLC 错误记录（Error Log）

- SR40~SR161

SR40：储存错误记录（Error Log）的有效组数，每一组占用 6 个寄存器，最多记录 20 组，若超出 20 之后，将一直为 20。

SR41：错误记录的指针，会指向最新一组的错误记录。当错误发生时，错误记录指针增加 1，当超出 19 之后，会再变为 0。指标的范围值为 0~19，例如：SR41=3 表示第 4 组为最新一笔错误记录，若是发现 SR40 为 20，则表示已经累积记录超过 20 组记录，并且已经循环记录。

SR42~SR161：记录各组错误发生的时间及代码，其中”插槽”字段记录主机（0）与远程（1~15）模块编号，各 SR 所对应的功能对照如下表：

组别	插槽	模块 ID	错误代码	错误发生时间					
				年	月	日	时	分	秒
1	SR42 下	SR43	SR44	SR45 上	SR45 下	SR46 上	SR46 下	SR47 上	SR47 下
2	SR48 下	SR49	SR50	SR51 上	SR51 下	SR52 上	SR52 下	SR53 上	SR53 下
3	SR54 下	SR55	SR56	SR57 上	SR57 下	SR58 上	SR58 下	SR59 上	SR59 下
4	SR60 下	SR61	SR62	SR63 上	SR63 下	SR64 上	SR64 下	SR65 上	SR65 下
5	SR66 下	SR67	SR68	SR69 上	SR69 下	SR70 上	SR70 下	SR71 上	SR71 下
6	SR72 下	SR73	SR74	SR75 上	SR75 下	SR76 上	SR76 下	SR77 上	SR77 下
7	SR78 下	SR79	SR80	SR81 上	SR81 下	SR82 上	SR82 下	SR83 上	SR83 下
8	SR84 下	SR85	SR86	SR87 上	SR87 下	SR88 上	SR88 下	SR89 上	SR89 下
9	SR90 下	SR91	SR92	SR93 上	SR93 下	SR94 上	SR94 下	SR95 上	SR95 下
10	SR96 下	SR97	SR98	SR99 上	SR99 下	SR100 上	SR100 下	SR101 上	SR101 下
11	SR102 下	SR103	SR104	SR105 上	SR105 下	SR106 上	SR106 下	SR107 上	SR107 下
12	SR108 下	SR109	SR110	SR111 上	SR111 下	SR112 上	SR112 下	SR113 上	SR113 下
13	SR114 下	SR115	SR116	SR117 上	SR117 下	SR118 上	SR118 下	SR119 上	SR119 下
14	SR120 下	SR121	SR122	SR123 上	SR123 下	SR124 上	SR124 下	SR125 上	SR125 下
15	SR126 下	SR127	SR128	SR129 上	SR129 下	SR130 上	SR130 下	SR131 上	SR131 下
16	SR132 下	SR133	SR134	SR135 上	SR135 下	SR136 上	SR136 下	SR137 上	SR137 下
17	SR138 下	SR139	SR140	SR141 上	SR141 下	SR142 上	SR142 下	SR143 上	SR143 下
18	SR144 下	SR145	SR146	SR147 上	SR147 下	SR148 上	SR148 下	SR149 上	SR149 下
19	SR150 下	SR151	SR152	SR153 上	SR153 下	SR154 上	SR154 下	SR155 上	SR155 下
20	SR156 下	SR157	SR158	SR159 上	SR159 下	SR160 上	SR160 下	SR161 上	SR161 下

8. PLC 下载记录（Download Log）

- SR227~SR308

  SR227：储存下载动作的有效组数，最多 20 组，且每一组占用 4 个寄存器。目前会记录的下载动作编号为 1~3，如下表说明。韧体版本 V1.10 版（含）以上及 ISPSoft 编辑软件 V3.13 版（含）以上支持动作编号 000A~0015

动作编号（HEX）	下载动作
0x0001	下载程序
0x0002	下载 PLC 设定
0x0003	下载 I/O 表
0x000A	PLC 执行码（ISP）
0x000B	PLC 参数（HWCONFIG）
0x000C	模块配置（HWCONFIG）
0x000D	PLC 执行码和程序（ISP）
0x000E	PLC 执行码（在线编辑）

动作编号（HEX）	下载动作
0x000F	PLC 执行码和程序（在线编辑）
0x0010	PLC 执行码和程序（SD 卡还原）
0x0011	模块配置和 PLC 参数（HWCONFIG）
0x0012	模块配置和数据交换表（HWCONFIG）
0x0013	模块配置、PLC 参数和数据交换表（HWCONFIG）
0x0014	模块配置（EIP Builder）
0x0015	模块配置、PLC 参数和数据交换表（SD 卡还原）

SR228：下载动作的记录指针（Download Log pointer），会指向最新一组的下载动作记录。每执行一次下载动作时，下载动作记录指摽增加 1.指针的范围值为 0~19，例如 SR228=3 表示第 4 组。

SR229~SR308：记录各组程序下载的时间及动作编号，SR 对应的功能说明如下。

组数	动作编号	*下载动作时间					
		年	月	日	时	分	秒
1	SR229	SR230 上	SR230 下	SR231 上	SR231 下	SR232 上	SR232 下
2	SR233	SR234 上	SR234 下	SR235 上	SR235 下	SR236 上	SR236 下
3	SR237	SR238 上	SR238 下	SR239 上	SR239 下	SR240 上	SR240 下
4	SR241	SR242 上	SR242 下	SR243 上	SR243 下	SR244 上	SR244 下
5	SR245	SR246 上	SR246 下	SR247 上	SR247 下	SR248 上	SR248 下
6	SR249	SR250 上	SR250 下	SR251 上	SR251 下	SR252 上	SR252 下
7	SR253	SR254 上	SR254 下	SR255 上	SR255 下	SR256 上	SR256 下
8	SR257	SR258 上	SR258 下	SR259 上	SR259 下	SR260 上	SR260 下
9	SR261	SR262 上	SR262 下	SR263 上	SR263 下	SR264 上	SR264 下
10	SR265	SR266 上	SR266 下	SR267 上	SR267 下	SR268 上	SR268 下
11	SR269	SR270 上	SR270 下	SR271 上	SR271 下	SR272 上	SR272 下
12	SR273	SR274 上	SR274 下	SR275 上	SR275 下	SR276 上	SR276 下
13	SR277	SR278 上	SR278 下	SR279 上	SR279 下	SR280 上	SR280 下
14	SR281	SR282 上	SR282 下	SR283 上	SR283 下	SR284 上	SR284 下
15	SR285	SR286 上	SR286 下	SR287 上	SR287 下	SR288 上	SR288 下
16	SR289	SR290 上	SR290 下	SR291 上	SR291 下	SR292 上	SR292 下
17	SR293	SR294 上	SR294 下	SR295 上	SR295 下	SR296 上	SR296 下
18	SR297	SR298 上	SR298 下	SR299 上	SR299 下	SR300 上	SR300 下
19	SR301	SR302 上	SR302 下	SR303 上	SR303 下	SR304 上	SR304 下
20	SR305	SR306 上	SR306 下	SR307 上	SR307 下	SR308 上	SR308 下

*下载时间：时间数据的存为 BCD 格式。时间的内容值范围如下：

功能名称	内容值范围
年	00~99（公元右两位）

功能名称	内容值范围
月	01~12
日	01~31
时	00~23
分	00~59
秒	00~59

9. PLC 状态变更记录（PLC Status Log）

- SR309~SR390

SR309：记录 PLC 状态变更（PLC Status Log）的有效组数，最多 20 组，且每一组占用 4 个寄存器。目前会记录的 PLC 状态变更编号为 1~5，如下表说明。

PLC 变更动作	动作编号
上电（POWER ON）	1
断电（POWER OFF）	2
PLC 开始运转（PLC RUN）	3
PLC 停止运转（PLC STOP）	4
PLC 恢复出厂设定	5
SM580 紧急输出停止（EMERGENCY STOP）	7
写入万年历（WRITE RTC）	8

SR310：PLC 状态变更的指针（PLC Status Log Pointer），会指向最新一组的 PLC 状态变更的记录。当 PLC 的状态被变更一次，记录指针增加 1。指针的范围值为 0~19，例如 SR310=3 表示第 4 组。

SR311~SR390：记录各组 PLC 状态变更发生的时间，SR 对应的功能说明如下：

组数	动作编号	*PLC 变更时间					
		年	月	日	时	分	秒
1	SR311	SR312 上	SR312 下	SR313 上	SR313 下	SR314 上	SR314 下
2	SR315	SR316 上	SR316 下	SR317 上	SR317 下	SR318 上	SR318 下
3	SR319	SR320 上	SR320 下	SR321 上	SR321 下	SR322 上	SR322 下
4	SR323	SR324 上	SR324 下	SR325 上	SR325 下	SR326 上	SR326 下
5	SR327	SR328 上	SR328 下	SR329 上	SR329 下	SR330 上	SR330 下
6	SR331	SR332 上	SR332 下	SR333 上	SR333 下	SR334 上	SR334 下
7	SR335	SR336 上	SR336 下	SR337 上	SR337 下	SR338 上	SR338 下
8	SR339	SR340 上	SR340 下	SR341 上	SR341 下	SR342 上	SR342 下
9	SR343	SR344 上	SR344 下	SR345 上	SR345 下	SR346 上	SR346 下
10	SR347	SR348 上	SR348 下	SR349 上	SR349 下	SR350 上	SR350 下
11	SR351	SR352 上	SR352 下	SR353 上	SR353 下	SR354 上	SR354 下
12	SR355	SR356 上	SR356 下	SR357 上	SR357 下	SR358 上	SR358 下
13	SR359	SR360 上	SR360 下	SR361 上	SR361 下	SR362 上	SR362 下

组数	动作编号	*PLC 变更时间					
		年	月	日	时	分	秒
14	SR363	SR364 上	SR364 下	SR365 上	SR365 下	SR366 上	SR366 下
15	SR367	SR368 上	SR368 下	SR369 上	SR369 下	SR370 上	SR370 下
16	SR371	SR372 上	SR372 下	SR373 上	SR373 下	SR374 上	SR374 下
17	SR375	SR376 上	SR376 下	SR377 上	SR377 下	SR378 上	SR378 下
18	SR379	SR380 上	SR380 下	SR381 上	SR381 下	SR382 上	SR382 下
19	SR383	SR384 上	SR384 下	SR385 上	SR385 下	SR386 上	SR386 下
20	SR387	SR388 上	SR388 下	SR389 上	SR389 下	SR390 上	SR390 下

*PLC 变更时间：时间数据的存为 BCD 格式。时间的内容值范围如下：

功能名称	内容值范围
年	00~99（公元右两位）
月	01~12
日	01~31
时	00~23
分	00~59
秒	00~59

10. PLC 运行标志

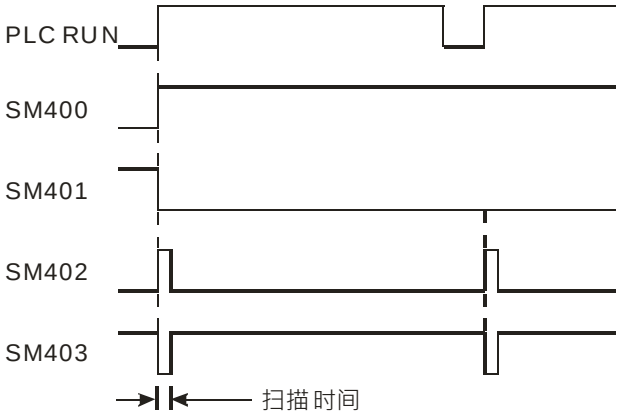
- SM400~SM403

SM400：常开接点。

SM401：常闭接点。

SM402：PLC 开始 RUN 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。

SM403：PLC 开始 RUN 的第一次扫描 OFF，之后一直 ON。即开始反向（RUN 的瞬间“OFF”）脉冲。



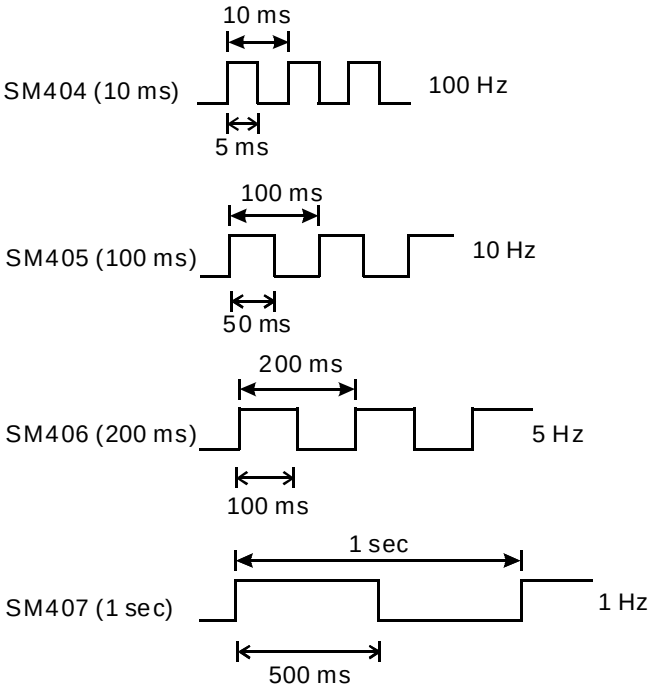
11. 内部时间脉冲

- SM404，SM405，SM406，SM407

  PLC 内部提供时钟脉冲。只要 PLC 通上电源，这时钟脉冲会自动动作。备注：此频率不受 PLC 扫描时间影响。

装置	功能说明
SM404	10ms 时钟脉冲，5ms ON/5ms OFF
SM405	100ms 时钟脉冲，50ms ON/50ms OFF
SM406	200ms 时钟脉冲，100ms ON/100ms OFF
SM407	1s 时钟脉冲，0.5s ON/0.5s OFF

时钟脉冲示意图如下：



12. 储存卡相关标志

- SM36、SM450~SM453、SM456、SR36、SR453、SR902

  储存卡主要提供用户备份 PLC 内容。其对应的 SM、SR 功能如下说明，详细使用方式请参考第 6 章储存

卡读写指令。

装置	功能说明
SM36	启动回存系统信息至储存卡内。ON 时，依 SR36 数值执行动作。
SM450	储存卡是否存在标志，ON：存在/OFF：不存在。
SM451	储存卡在写保护的模式里
SM452	储存卡正被存取（Accessed）中，ON：存取中/OFF：无存取。
SM453	储存卡运行中有错误发生，ON：表错误发生。
SM456	数据记录器与储存卡动作标志。ON 时，依 SR902 数值执行动作。
SR36	系统信息储存至储存卡内之功能代码，需搭配 SM36 启动。
SR453	当储存卡有错误发生，错误代码将被记录。 A. 16#005D ➔ 未插卡 B. 16#005E ➔ 无法载入 SD 卡文件系统 C. 16#005F ➔ 路径错误或文件不存在 D. 16#0060 ➔ 无法产生默认文件夹 E. 16#0061 ➔ 储存卡内存空间不足 F. 16#0062 ➔ 储存卡为写保护 G. 16#0063 ➔ 文件写入失败 H. 16#0064 ➔ 文件读取失败 I. 16#0065 ➔ 文件属性为只读状态 J. 16#0066 ➔ 还原时，PLC 在 RUN 状态 K. 16#0067 ➔ PLC ID 比对不一致 L. 16#0068 ➔ 密码比对不一致 M. 16#0069 ➔ 路径或文件名比对主机类型不一致
SR902	数据记录器与储存卡动作代码。需搭配 SM456 启动。（H5AA5：将数据记录器取样数据写入储存卡）。
SR36	系统信息储存至储存卡内之功能代码，需搭配 SM36 启动。 A. 数值 K1122（备份）#1，代表**备份** PLC 程序（含 PLC ID，密码）至 SD 卡内。可在 Run 或 Stop 执行。 路径文件名：SDCard\PLC CARD\AS300\SysDup\user_program.dup B. 数值 K2211（还原）#2，代表从 SD 卡**还原** PLC 程序（含 PLC ID，密码）。须在 Stop 状态下执行。 路径文件名：SDCard\PLC CARD\AS300\SysDup\user_program.dup C. 数值 K3344（备份）#1，代表备份停电保持装置（D，S，M，C，HC）和停电保持装置起始/结束范围参数至 SD 卡内。可在 Run 或 Stop 执行。 路径文件名：SDCard\PLC CARD\AS300\SysDup\retained_data.dup D. 数值 K4433（还原）#2，代表从 SD 卡还原停电保持装置（D，S，M，C，HC）和停电保持装置起始/结束范围参数。须在 Stop 状态下执行。

装置	功能说明
	路径文件名：SDCard\PLC CARD\AS300\SysDup\retained_data.dup E. 数值 K5566（备份）#1，代表备份全部 PLC 内部程序与参数设定数据至 SD 卡内，但不储存停电保持装置的状态与数值。须在 Stop 状态下执行。 F. 数值 K5577（备份）#1，代表备份全部 PLC 内部程序与参数设定数据，以及储存停电保持装置的状态与数值至 SD 卡内。须在 Stop 状态下执行。 E & F 功能就如同使用 ISPSoft Card Utility 备份完整的项目内容。其储存路径文件名：SDCard\PLC CARD\AS300\SysDup\AS300_save_backup.dup G. 数值 K6755（还原）#2，代表从 SD 卡还原上述 E 或 F 项已备份所有数据。须在 Stop 状态下执行。 路径文件名：SDCard\PLC CARD\AS300\SysDup\AS300_save_backup.dup （注：若是要采用上电自动还原功能，请在完成备份后，再自行修改文件名，并且存放于底下路径→SDCard\PLC CARD\AS300\SysDup\AS300_BACKUP.dup） H. 数值 K5588（备份）#1，代表备份全部 PLC 内部 FR 文件寄存器至 SD 卡内。须在 Stop 状态下执行。 其储存路径文件名：SDCard\PLC CARD\AS300\SysDup\AS300_FR_backup.dup I. 数值 K8855（还原）#2，代表从 SD 卡还原全部 FR 数据至 PLC 内。须在 Stop 状态下执行。 路径与文件名需与 H 项一样。

注#1：备份时若已有相同文件名存在，则会直接覆盖原先旧文件，不会自动将旧文件再备份一份。

注#2：路径与文件名须符合文件内指定名称，若要还原时文件名不符合，将被视为不执行任何动作。

注：以上路径与文件名皆以 AS300 为例说明，若非使用 AS300 主机时，请自行修改路径与文件名，若未修改符合主机系列名称，则 PLC 将会无法执行任何动作，并会设定错误代码。

注：SR36 说明内的 A，B，C & D 功能，AS 韧体需搭配 V1.04.30 版（含）以上；SR36 说明内的 E，F & G 功能，AS 韧体需搭配 V1.04.90 版(含)以上；SR36 说明内的 H，I 功能，AS 韧体需搭配 V1.05.30 版(含)以上。

注：PLC 程序包含：PLC 程序执行码、程序代码、项目密码与 PLC 标识符。
参数设定包含：于 HWCONFIG 配置与硬件参数设定值、所有通讯数据交换表，以及定位规划表。
停电保持包含：停电保持区范围设定参数，以及所有停电保持装置的数值。

- **进行 PLC 程序还原时，会比对备份文件与主机的 PLC ID 与密码**

  ◆ PLC ID 比对：(备份时有设定 PLC ID，可防止 SD 卡直接复制至别台空机)

	主机有 PLC ID	主机无 PLC ID
备份文件有 PLC ID	比对一致：进一步比对密码 比对不一致：不还原	不还原
备份文件无 PLC ID	不还原	进一步比对密码

  ◆ 密码比对：(可防止 PLC 程序被打开，以及可直接复制到空机内)

	主机有密码	主机无密码
备份文件有密码	比对一致：还原 比对不一致：不还原	还原，同时设定密码
备份文件无密码	不还原	还原

13. 高速输出指令执行中，指令关闭或暂停输出时，采用立即停止输出标志

- SM587：定位规划表输出中执行暂停功能时，依照下述各轴设定之暂停动作执行（V1.06.00 版（含）以后支持）
- SM476、SM477、SM496、SM497、SM516、SM517、SM536、SM537、SM556、SM557、SM576、SM577：当高速输出指令正在执行输出中，此时指令被关闭或者启动立即暂停标志时，此标志 OFF（默认值）表示执行减速停止之动作；若标志为 ON，则表示执行立即停止输出之动作。
- SM463、SM474、SM483、SM494、SM503、SM514、SM523、SM534、SM543、SM554、SM563、SM574：当标志为启动输出暂停功能，当此标志由 OFF→ON 时，表示输出进入暂停状态（其暂停动作需参考上述标志动作）；若标志由 ON→OFF 时，则表示执行剩余未输出完毕之输出个数。
- **定位输出轴 1~轴 6（即偶数高速输出点）之加减速区段选择标志（V1.06.00 版（含）以后支持）。**
- SM586
  OFF（默认值）：加减速输出区段采用 250 段高密度切换速度，其加减速时间误差约 10%。
  ON：加减速输出区段采用 25 段切换速度，其加减速时间误差约 1%。

14. 定位控制之软件极限功能

- SR580~SR603
  正向输出极限定义：输出当前位置大于正向极限位置时，即立即停止输出。
  反向输出极限定义：输出当前位置小于反向极限位置时，即立即停止输出。
  当正向与反向两个极限值皆为 0 时，表示软件极限功能不启动。由于此软件极限功能为输出指令被扫描时，才会进行检查输出位置之动作，故停止动作将会受 PLC 程序扫描影响，若需更快且实时停止之动作，请改用外部输入点为极限方式。

15. 背隙补偿功能

- SR478、SR479、SR498、SR499、SR518、SR519、SR538、SR539、SR558、SR559、SR578、SR579.AS 系列最多可设置 12 点高速输出点（Y0.0 ~ Y0.11），其各个输出点分别搭配一个 SR，可供用户设定背隙补偿的输出个数。此参数可设定数值范围为 0~32767，若设定数值<= 0，则表示不启动背隙补偿功能。
- 韧体版本从 V1.02.30（含）以上支持。采用偶数轴输出并且内定方向输出轴，其支持的输出指令为 JOG, DZRN、DPLSV、DDRVI、DDRVA、DPPMR、DPPMA、DCICR、DCICA、DCICCR、DCICCA、DCCMR、DCCMA、DPPGB 及 TPO。V1.10（含）以上支持 CSFO；采用单数输出轴或者方向为自定输出点的输出指令仅有支持 JOG、DPLSV、DDRVI 及 DDRVA。
- 当启动背隙补偿之后，若刚好使用以上支持的指令，并且在刚切换方向输出，然后又立即关闭指令时，PLC 必须自动跑完背隙补偿的个数，才会真正的停止输出，以防止下回启动输出时，存在背隙还未补偿的情况。

16. 每次定位轴输出时，显示该次定位输出之加减速区段分别输出多少个数（V1.06.00 版（含）以后支持）

- SR652~SR653（32 位数值）

加速区间输出个数：计算启动速度加速至目标速度（不含）之间所需输出的累积个数。

- SR654~SR655（32 位数值）

  减速区间输出个数：计算目标速度（不含）减速至停止输出之间所需输出的累积个数。

17. 显示 CAN 通讯质量状况（V1.06.00 版（含）以后支持）

- SR823

  CAN 通讯接收时硬件错误计数器：当计数器累积计数大于 127，表示 CAN 通讯硬件经常被干扰，请检查通讯线路、调降传送速度或手动调整 CAN 取样点。

- SR824

  CAN 通讯发送时硬件错误计数器：当计数器累积计数大于 127，表示 CAN 通讯硬件经常被干扰，请检查通讯线路、调降传送速度或手动调整 CAN 取样点。

18. 以太网络相关标志

- SM1000、SR1000~SR1015、SR1016、SR1536~SR1539（V1.06.00 版（含）以后支持 SR1016、SR1536~SR1539）

SM/SR 编号	功能名称	动作说明
SM1000	以太网络设定标志	ON 时，PLC 将 SR1000~SR1006 设定值永久保存至 PLC 内部 HWCONFIG 储存区内，并且立即生效。当储存完毕时，PLC 自动清除 SM1000 为 OFF。 备注 1：请勿连续对此标志设定为 ON，避免造成永久保存内存损坏。 备注 2：设定时机限制在 PLC 为 STOP 状态下，才允许进行永久储存之动作。
SR1000	Ethernet IP 地址（32 位）	举例：192.168.1.5，则 SR1000 为 16#C0A8，SR1001 为 0105
SR1001		
SR1002	以太网络网络屏蔽地址（32 位）	举例：255.255.255.0，则 SR1002 为 16#FFFF，SR1003 为 FF00
SR1003		
SR1004	以太网络网关地址（32 位）	举例：192.168.1.1，则 SR1004 为 16#C0A8，SR1005 为 0101
SR1005		
SR1006	TCP 保持联机时间	单位：秒，默认 30 秒，可设定范围时间：1~32000，设定超出范围时自动以最小或最大值执行
SR1007	以太网当前速度	单位：Mbps，举例：100 表示 100Mbps
SR1009	TCP 目前联机数	显示当下所有 TCP 联机数
SR1010	TCP 重传时间	单位：ms，默认 20ms，可设定范围时间：0~32000 0 表示系统自动依当前 TCP 通讯时间的平均值时间重传，其它设定超出范围时，自动以最小或最大值执行
SR1011	MODBUS/TCP Server 联机数	由系统自动更新
SR1012	MODBUS/TCP Client 联机数	由系统自动更新
SR1013	EtherNet/IP Adapter 联机数	由系统自动更新
SR1014	EtherNet/IP Scanner 联机数	由系统自动更新

SM/SR 编号	功能名称	动作说明
SR1015	MODBUS/TCP Server 接收等待逾时时间	单位：ms，默认 10ms，可设定范围时间：0~32000，设定超出范围时自动以最小或最大值执行
SR1016	MODBUS/TCP Client 成功建立联机后，延迟开始发送的等待时间	单位：ms，默认 10ms，可设定范围时间：0~1000，设定超出范围时自动以最小或最大值执行
SR1536	AS-FEN02/AS-FECAT TCP 目前联机数	显示当下所有 TCP 联机数
SR1537	AS-FEN02/AS-FECAT MODBUS/TCP Server 联机数	由系统自动更新
SR1538	AS-FEN02 MODBUS/TCP Client 联机数	由系统自动更新
SR1539	AS-FEN02 EtherNet/IP Adapter 联机数	由系统自动更新

- Ethernet 通讯质量相关装置，V1.10.00 版（含）以后支持

SR 编号	功能名称	动作说明
SR1017	每个扫描周期最多可处理 Ethernet 的报文数量	默认 80，可设定范围：10~1000，设定超出范围时自动以最小或最大值执行
SR1018	最近一次扫描周期处理的 Ethernet 报文数	由系统自动更新
SR1019	Ethernet 缓冲区满载而丢掉的 Ethernet 报文累积数量	由系统自动更新，用户可依需求手动清除为 0

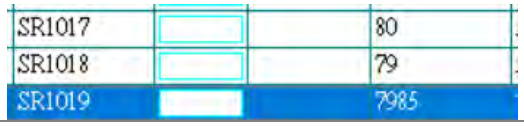
范例说明：

1. 网络质量判断

步骤一、确认 SR1019 是否为 0

SR1019 = 0	网络质量良好
SR1019 > 0	网络曾发生异常，请跳至步骤二

步骤二、确认当下 SR1017 与 SR1018 数值

SR1018/SR1017<0.8	表示网络负荷低于 80%，网络目前质量良好，请手动将 SR1019 设置为 0，观察网络异常发生频率。若偶尔发生，请联系网络管理员，确认网络中是否存在可能会造成网络异常的设备。
SR1018/SR1017≥0.8	表示网络负荷高于等于 80%，处于高负载状态。若同时 SR1019 数值不为 0，表示网络正在发生异常，请降低数据交换负载，或参考下方说明调整网络能力。 

2. 调整网络能力

- 用户可手动将 SR1017 数值提高，若发现 SR1019 已不会再增加，但 SR1018/SR1017 比例依旧高于 0.8，则表示仍处于通讯满载的临界状态，建议继续增加 SR1017 数值，将比例压低至 0.8 以下。每次变更 SR1017 后，可手动清除 SR1019 进行验证。

SR1017			150
SR1018			110
SR1019			0

- 当 SR1017 数值提高后，需注意对扫描时间的影响，若没有发生网络流量过大情况，SR1017 的数值不会影响扫描时间。若发生网络异常，主机以较多资源处理网络报文，将导致扫描时间提高，用户须自行评估系统可承受的扫描时间上升程度。网络持续满载的情况下，SR1017 每增加 10，扫描时间大约增加 2.2ms（参考测试环境：无右侧模块，无 RTU 与数据交换表）。
- 若无法在许可的扫描时间范围内处理网络报文，请参考下列手法降低网络负载：
  - 降低主机的网络联机数，例如减少 Modbus TCP 或 EtherNet/IP 数据交换台数。
  - 降低数据交换更新频率，例如提高 Modbus TCP 数据交换最短更新周期，或提高 EtherNet/IP 数据交换 RPI 参数。

- SM1090、SM1091、SM1106~SM1109

SM 编号	功能名称	动作说明
SM1090	TCP 联机忙碌	ON 时表示 TCP 联机逾时
SM1091	UDP 联机忙碌	ON 时表示 UDP 联机逾时
SM1106	以太网络联机错误	OFF：以太网络自动协商成功 ON：以太网络自动协商失败
SM1107	以太网络基本设定错误	OFF：基本设定正确 ON：基本设定错误
SM1109	TCP/ UDP 本地埠已被使用	ON 时表示使用到相同的端口

关于标志对应的 LED 灯状态、错误代码及其它详细说明请参考 AS 操作手册第 12 章。

19. Email 发送的设定

- SM1113，SM1116~SM1155

  若 Email 发送后失败，则 Email 服务失败标志 SM1113 会被设定为 ON。

  邮件发送条件 1（Trigger1）~邮件发送条件 4（Trigger4）的参数设定与触发后的结果标志（SM1116~SM1155）说明如下表：

项目 / 功能说明	条件 1	条件 2	条件 3	条件 4
Email 功能启动	SM1116	SM1126	SM1136	SM1146
	ON 为功能启动，OFF 为功能关闭			
Email 发送中标志	SM1117	SM1127	SM1137	SM1147
	ON 为正在发送中， OFF 为已结束发送			
Email 发送成功状态	SM1119	SM1129	SM1139	SM1149
	ON 为发送完成，且已接收回应成功			

项目 / 功能说明	条件 1	条件 2	条件 3	条件 4
Email 发送错误状态 1	SM1120	SM1130	SM1140	SM1150
	发送时，检查邮件内容有错误，无法发送成功			
SMTP 回复逾时标志	SM1122	SM1132	SM1142	SM1152
	发送后，SMTP 服务器回应逾时			
SMTP 回复错误标志	SM1123	SM1133	SM1143	SM1153
	发送后，SMTP 服务器回应错误			
Email 发送错误状态 2	SM1124	SM1134	SM1144	SM1154
	发送时，检查附件大小超过限制			
Email 发送错误状态 3	SM1125	SM1135	SM1145	SM1155
	发送时，检查附件不存在			

20. 数据交换功能标志与寄存器

- COM1 与 COM2 通讯口数据交换功能标志如下：

SM 编号	属性	COM1 数据交换参数内容说明
SM750	R/W	数据交换启动标志
SM752 ~ SM783	R/W	数据交换联机编号 1~32 之启动标志
SM784 ~ SM815	R	数据交换联机编号 1~32 之成功标志
SM816 ~ SM847	R	数据交换联机编号 1~32 之错误标志
SM 编号	**属性**	**COM2 数据交换参数内容说明**
SM862	R/W	数据交换启动标志
SM864 ~ SM895	R/W	数据交换联机编号 1~32 之启动标志
SM896 ~ SM927	R	数据交换联机编号 1~32 之成功标志
SM928 ~ SM959	R	数据交换联机编号 1~32 之错误标志

- COM1 与 COM2 通讯口数据交换功能寄存器如下：

SR 编号	说明
SR1335	COM1 数据交换实际执行联机 1~32 的周期时间
SR1336	COM1 数据交换目前执行周期联机编号
SR1340 ~ SR1371	COM1 数据交换联机编号 1~32 的错误代码
SR1375	COM2 数据交换实际执行联机 1~32 的周期时间
SR1376	COM2 数据交换目前执行周期联机编号
SR1380 ~ SR1411	COM2 数据交换联机编号 1~32 的错误代码

以上错误代码 1~7 为标准 MODBUS 协议之响应错误码，当错误代码为 9 时，表示通讯逾时。

- 序列通讯卡 1 与序列通讯卡 2 通讯口数据交换功能标志如下：V1.06.00 版（含）以后支持

SM 编号	属性	序列通讯卡 1 数据交换参数内容说明
SM1710	R/W	数据交换启动标志
SM1712 ~ SM1743	R/W	数据交换联机编号 1~32 之启动标志
SM1744 ~ SM1775	R	数据交换联机编号 1~32 之成功标志
SM1776 ~ SM1807	R	数据交换联机编号 1~32 之错误标志

SM 编号	属性	序列通讯卡 2 数据交换参数内容说明
SM1822	R/W	数据交换启动标志
SM1824 ~ SM1855	R/W	数据交换联机编号 1~32 之启动标志
SM1856 ~ SM1887	R	数据交换联机编号 1~32 之成功标志
SM1888 ~ SM1919	R	数据交换联机编号 1~32 之错误标志

- 序列通讯卡 1 与序列通讯卡 2 通讯口数据交换功能寄存器如下：V1.06.00 版（含）以后支持

SR 编号	说明
SR1435	序列通讯卡 1 数据交换实际执行联机 1~32 的周期时间
SR1436	序列通讯卡 1 数据交换目前执行周期联机编号
SR1440 ~ SR1471	序列通讯卡 1 数据交换联机编号 1~32 的错误代码
SR1475	序列通讯卡 2 数据交换实际执行联机 1~32 的周期时间
SR1476	序列通讯卡 2 数据交换目前执行周期联机编号
SR1480 ~ SR1511	序列通讯卡 2 数据交换联机编号 1~32 的错误代码

以上错误代码 1~7 为标准 MODBUS 协议之响应错误码，当错误代码为 9 时，表示通讯逾时。

- Ethernet 通讯口数据交换功能标志如下：

SM 编号	属性	Ethernet 数据交换参数内容说明
SM1167	R/W	数据交换启动标志
SM1168 ~ SM1199	R/W	数据交换联机编号 1~32 之启动标志
SM1200 ~ SM1231	R	数据交换联机编号 1~32 之成功标志
SM1232 ~ SM1263	R	数据交换联机编号 1~32 之错误标志

- Ethernet 通讯口数据交换功能寄存器如下：

SR 编号	说明
SR1120 ~ SR1151	分别为联机编号 1~32 之实际执行通讯时间
SR1152 ~ SR1183	分别为联机编号 1~32 之通讯错误代码

- Ethernet 通讯口数据交换功能通讯端口设定寄存器如下：V1.10.20 版（含）以后支持

SM 编号	属性	说明

SM1092	R/W	MODBUS TCP 从站通讯端口号错误标志

SR 编号	说明
SR1092	MODBUS TCP 从站通讯端口号（主机为从站）
SR1093	MODBUS TCP 数据交换表通讯端口号（主机为主站）

操作说明：

1. 主机作为 MODBUS TCP 通讯从站时
   - 默认开启通讯端口号 502 被远程主站联机，此通讯端口号为 MODBUS TCP 规范所使用。
   - 当远程主站的目标通讯端口无法使用 502 时，可修改 SR1092 为所需之通讯端口号。
   - 此设定值不具有停电保持，每次重新上电时将自动恢复为 502，使用时请利用 PLC 程序做设定。
   - 当 SR1092 设定的通讯端口号已被其他功能占用（例如 Socket、Web）时，设定不会生效，且 SM1092 将自动设为 ON。
2. 主机建立数据交换表作为 MODBUS TCP 通讯主站时
   - 默认远程从站的通讯端口号为 502，此通讯端口号为 MODBUS TCP 规范所使用。
   - 当远程从站无法使用 502 作为通讯端口号时，可修改 SR1093 为所需通讯端口号。
   - 此设定值不具有停电保持，每次重新上电时将自动恢复为 502，使用时请利用 PLC 程序做设定。
   - 此设定值仅修改预期远程从站通讯端口号，不会影响主机的自身通讯端口。

- AS-FEN02 Ethernet 通讯口数据交换功能标志如下：V1.06.00 版（含）以上支持

SM 编号	属性	Ethernet 数据交换参数内容说明
SM1006	R/W	数据交换启动标志
SM1008 ~ SM1015	R/W	数据交换联机编号 1~8 之启动标志
SM1016 ~ SM1023	R	数据交换联机编号 1~8 之成功标志
SM1024 ~ SM1031	R	数据交换联机编号 1~8 之错误标志

- AS-FEN02 Ethernet 通讯口数据交换功能寄存器如下：V1.06.00 版（含）以后支持

SR 编号	说明
SR1520 ~ SR1527	分别为联机编号 1~8 之实际执行通讯时间
SR1528 ~ SR1535	分别为联机编号 1~8 之通讯错误代码

- Ethernet 通讯错误代码如下表所示：

错误代码	说明
16#00XX	远程设备响应错误通讯代码，代码 1~7 为标准 MODBUS TCP 协议之响应错误码
16#F000	Ethernet 网络未连接
16#F001	远程设备响应逾时
16#F003	TCP 联机通讯逾时
16#F007	回应命令错误
16#F009	远程设备中断联机

- Ethernet 通讯口与 RTU-EN01 模块通讯状态如下表（备注 :在未联机成功前 ,AS 对应区数据(RX/RY/RCR Read/RCR Write）不建议开始应用）

SM / SR 编号		说明	
SM1312 ~ SM1315		RTU-EN01 联机编号 1~4 之通讯状态标志	
SR1312 ~ SR1315		RTU-EN01 联机编号 1~4 之通讯状态代码	
SR 状态代码	SM 状态	说明	
0	Off	联机关闭	
1	On	联机成功	
2	Off	通讯响应逾时	
3	Off	已强制断线	
4	Off	RTU-EN01 响应内容检查错误	
5	Off	RTU-EN01 回应错误	
6	Off	网络未连接或建立联机失败	
SM 状态标志变化		RX 与 RCR Read 数据（输入映像区）	RY 与 RCR Write 数据（输出映像区）
Off → On		清除为 0	清除为 0
On → Off		不变	不变

- EtherNet/IP 通讯标志如下：V1.06.00 版（含）以后支持

SM 编号	属性	EtherNet/IP 数据交换参数内容说明
SM1375	R/W	EtherNet/IP 主站（Scanner）数据交换功能之启动标志
SM1376 ~ SM1407	R/W	EtherNet/IP 主站（Scanner）数据交换联机编号 1~32 之启动标志
SM1408 ~ SM1439	R	EtherNet/IP 主站（Scanner）数据交换联机编号 1~32 之错误标志
SM1440 ~ SM1447	R	EtherNet/IP 从站（Adapter）联机编号 1~8 之错误标志
SM1448 ~ SM1455#1	R	EtherNet/IP 从站（Adapter）联机编号 1~8 已被联机标志

注#1：V1.06.50 版（含）以后支持

- EtherNet/IP 通讯功能寄存器如下：V1.06.00 版（含）以后支持

SR 编号	属性	EtherNet/IP 数据交换参数内容说明
SR1020 ~ SR1051	R	EtherNet/IP 主站（Scanner）数据交换联机编号 1~32 之状态代码

- EtherNet/IP 通讯错误代码如下表所示：详细错误码与处理方法请参阅 AS 操作手册第九章

错误代码	说明
16#0100	I/O Connections 重复建立
16#0106	多 Scanner 建立 I/O Connections 冲突
16#0110	Adapter configuration 参数设定错误
16#0111	Adapter RPI 参数设定错误
16#0113	I/O Connections 联机数不足
16#0119	Non-Listen only 联机建立失败

错误代码	说明
16#0127	Adapter input size 参数错误
16#0128	Adapter output size 设定错误
16#012D	Consumed TAG 参数错误
16#012E	Produced TAG 参数错误
16#0204	建立 I/O Connections 时通讯逾时
16#0302	网络配置超过产品 PPS 规格
16#0315	Adapter input/output instance 参数设定错误
16#0203	I/O Connections 通讯逾时

21. SM230~SM261 为上电时 I/O 模块" 手动+标志"配置方式之启动标志"手动+标志（仅主机右侧模块）" 时，仅主机右侧第 1~32 台模块启动标志（韧体 V1.06.00 版（含）以后支持）

"手动+标志（主机与远程右侧模块）"时，主机与远程的右侧第 1~32 台模块启动标志（韧体 V1.08.50 版（含）以后支持）(ISPSoft 需 V3.12 版以后才有此选项）

提供用户减少模块配置，并且不希望变更原有的硬件配置时，可不需修改 PLC 程序，也可直接应用。但在使用上有 2 点限制：

A. 需以最大的模块配置需求先行规划，并储存至 AS 主机内部

B. 实际配置时只可减少模块，並且原先模块排列顺序不可交换

用户可用 AS 系列的"手动+标志"模式，搭配 SM230 ~ SM261，规画机台最完整的组合，日后如需减少模块，只须关闭不使用的模块启动标志，即可设定成符合实际机台的模块配置。因此用户只须写一份 PLC 程序及下载同一份 IO 配置表即可因应不同的应用。

范例：

步骤 1：HWCONFIG 选择手动+标志模式

步骤 2：设计主机右侧完整模块组合并下载至 HWCONFIG

步骤 3：当机台 16AP，04DA，08AM，3 台模块（红色框）可关闭 SM231、SM233 及 SM235 标志，并且断电在上电之后，机台运行的实际使用配置如下图所示，但 PLC 程序内 IO 编号皆不需变更。

实际使用配置

完整模块组合配置时的数据对应区与 IO 编号

规划顺序	0	1	2	3	4	5	6	7
型号	AS332	16AP	16AP	04AD	04DA	06XA	08AM	08AN
对应装置起始编号		X1.0/Y1.0	X2.0/Y2.0	D28040	D28060	D28080	X3.0	Y3.0

实际配置后，对应区与 IO 编号都不会改变

实际顺序	0	1	3	5	7
型号	AS332	16AP	04AD	06XA	08AN
对应装置起始编号		X1.0/Y1.0	D28040	D28080	Y3.0

使用 FROM/TO 指令或是其它模块指令时，所指定的模块位置，是依完整模块配置所规划的模块顺序。

22. 显示现状模块数量和机种代码

SR 编号	说明
SR1559	主机已联机的远程模块数量（不含远程右侧模块数量）
SR1560	主机右侧连接的模块数量
SR1561~SR1592	显示主机或远程模块右侧第 1~32 台模块的机种代码

DIO 类（数字）模块					
型号	机种代码	型号	机种代码	型号	机种代码
AS08AM10N-A	16#0201	AS08AN01T-A	16#0206	AS08AN01P-A	16#0207
AS08AN01R-A	16#0208	AS16AM10N-A	16#0210	AS16AP11T-A	16#0212
AS16AP11P-A	16#0213	AS16AP11R-A	16#0214	AS16AN01T-A	16#0216
AS16AN01P-A	16#0217	AS16AN01R-A	16#0218	AS32AM10N-A	16#0220
AS32AN02T-A	16#0226	AS64AM10N-A	16#0230	AS64AN02T-A	16#0236

AIO 类（模拟、温度与称重）模块					
型号	机种代码	型号	机种代码	型号	机种代码
AS04AD-A	16#0301	AS08AD-B	16#0302	AS08AD-C	16#0303
AS02ADH-A	16#0304	AS04DA-A	16#0305	AS06XA-A	16#0309
AS04RTD-A	16#0310	AS06RTD-A	16#0311	AS04TC-A	16#0315
AS08TC-A	16#0316	AS02LC-A	16#0320		

NIO 类（通讯）模块					
型号	机种代码	型号	机种代码	型号	机种代码
AS00SCM-A	16#0401	AS01DNET-A	16#0402	AS04SIL-A	16#0403

PIO 类（定位与计数）模块					
型号	机种代码	型号	机种代码	型号	机种代码
AS02PU-A	16#0501	AS04PU-A	16#0502	AS02HC-A	16#0505

范例说明：

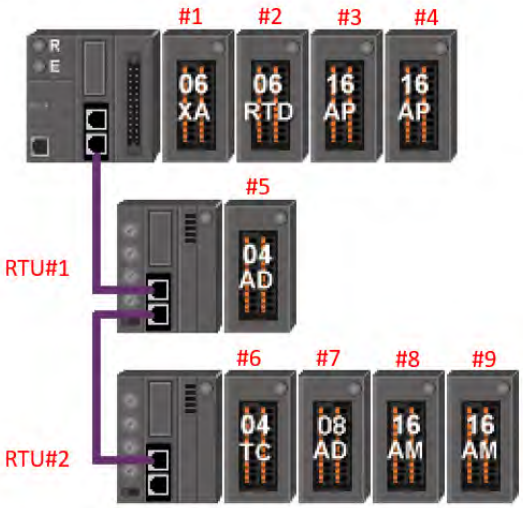
以下图配置为例，上电时检测模块后，各 SR 将呈现如下：

SR1559 = 2

SR1560 = 4

各模块顺序编号由主机右侧开始，再依 RTU 顺序。

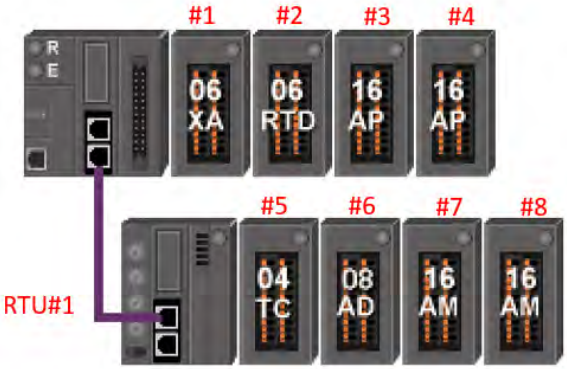
SR 编号（顺序）	SR1651（#1）	SR1652（#2）	SR1653（#3）	SR1654（#4）	SR1655(#5)
机种代码	16#0309	16#0311	16#0212	16#0212	16#0301
SR 编号（顺序）	SR1656（#6）	SR1657（#7）	SR1658（#8）	SR1659（#9）	
机种代码	16#0315	16#0302	16#0210	16#0210	



当 RTU 断线，重连时，会再更新 SR。

如果 RTU#1 断线：

SR1559 = 1
SR1560 = 4



23. 脉冲输出自动复位标志

- 执行脉冲输出指令时，为避免不同类或不同位置触发之输出指令，重复控制相同的脉冲输出装置，因此 PLC 将会由先启动的脉冲指令取得脉冲输出装置的控制权，直到该脉冲指令关闭，才会释放控制权。当使用 ST 语言编辑、中断子程序或是在执行一次性的功能块中使用脉冲输出指令时，由于这三种编辑方式的 PLC 程序扫描特性，将无法在脉冲输出指令被关闭时被 PLC 扫描，于是将会造成脉冲输出控制权一直被占用，以及后续启动其它脉冲输出指令时无法正常输出脉冲。
- 为了解决上述这类只扫描脉冲输出指令一次的情况，建议可使用脉冲输出自动复位标志，由 PLC 自行监看指定输出装置是否已经输出完毕，并释放脉冲输出控制权。

  注意：此类自动复位标志由用户自行设定，当控制权被释放之后，该标志将会自动被清除。

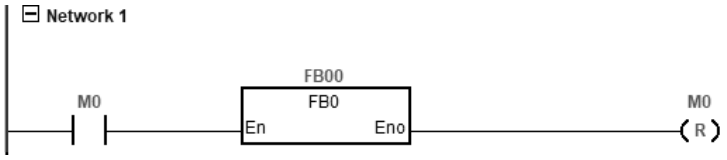
偶数输出轴	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
自动复位标志	SM470	SM490	SM510	SM530	SM550	SM570
奇数输出轴	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
自动复位标志	SM475	SM495	SM515	SM535	SM555	SM575

- 与自动复位相关之脉冲输出指令列表：

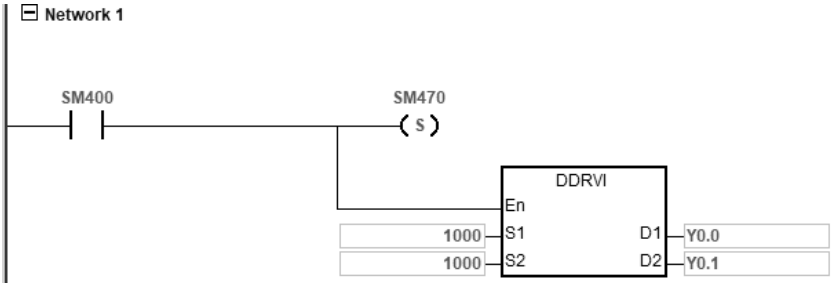
脉冲指令名称	偶数轴	奇数轴	支持与否	自动复位时机点	补充说明
PWM/DPWM	V	V	Yes	复位标志设 ON	• 属于没有设定目标输出个数的指令。 • 建议使用于关闭指令之后，指令将不再被扫描的应用情境（类似采用 ST 语言编辑）
JOG/DJOG	V	V			
DPLSV	V	V			
DPPGB	V				
CSFO	V				
PLSR	V	V	Yes	复位标志 ON 而且脉冲输出完毕	• 属于需设定目标脉冲输出个数的指令。 • 有带方向的脉冲指令，若碰正反极限，也会自动复位。 • 双轴以上输出的指令，须设定相关轴的复位标志，才能分别释放多轴的控制权。
PLSY	V	V			
DDRVI	V	V			
DDRVA	V	V			
DDRVM	V				
DPPMR	V				
DPPMA	V				
DCICR	V				
DCICA	V				
DCICCR	V				
DCICCA	V				
DCCMR	V				
DCCMA	V				
DZRN	V		Yes（V1.12.00）	复位标志 ON 而且脉冲输出完成标志 ON	• 每次都须扫描与监控执行情形。 • 不建议在一次性调用功能块使用。 • 在 ST 语言，关闭指令时，须搭配复位标志使用。
DZRN2	V	V			

程序范例 1（适用于使用功能块或中断程序）

POU



功能块（FB）或中断程序：



动作说明：

A. 当 M0=ON 时，触发执行一次性功能块，Y0.0 输出 1000 个 1kHz 频率脉冲，并且设定 Y0.0 的复位标志（SM470）。

B. 当 Y0.0 脉冲输出完毕，PLC 释放 Y0.0 输出控制权。

C. 当 M0 再次由 OFF->ON 时，Y0.0 脉冲输出即会重新启动。

程序范例 2（使用 ST 语言）

```
0001 IF M0 THEN
0002     DDRVI(10000, 1000, Y0.0, Y0.1);
0003     M1 := TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
0008
```

动作说明：

A. 当 M0=ON 时，Y0.0 输出 1000 个 1kHz 频率脉冲，并且设定 M1=ON，用来记录 Y0.0 控制权已经被使用。

B. 当 M0=OFF 时，脉冲输出完毕 SM461=ON 并且 M1=ON 时，触发 Y0.0 的复位标志 SM470=ON，通知 PLC 释放 Y0.0 输出控制权。

C. 当 M0 再次由 OFF->ON 时，脉冲输出即会重新启动。

24. 台达专属驱动器 CAN 通讯之自动复位标志

- 执行 CAN 通讯指令时，为避免不同类或相同 CAN 通讯指令控制相同站号 ID 的伺服/变频装置，PLC 将会在启动的 CAN 通讯指令时，也同时取得伺服/变频装置的通讯控制权，直到该指令关闭，才会释放控制权。因此使用 ST 语言编辑或是在中断子程序、执行一次性的功能块中使用 CAN 通讯指令时，由于这三种编辑方式的 PLC 程序扫描特性，将无法在 CAN 通讯指令被关闭时被 PLC 扫描，于是造成执行其它 CAN 通讯指令时，无法正常控制伺服/变频。
- 为解决上述情况，可设定 CAN 通讯自动复位标志为 ON，由 PLC 自行监看通讯与伺服/变频装置是否完成动作，若已完成即可释放其控制权

  注意：此自动复位标志由用户自行设定，当控制权释放之后，PLC 会自动清除此标志。

伺服站号	1	2	3	4	5	6	7	8
复位标志	SM1581	SM1582	SM1583	SM1584	SM1585	SM1586	SM1587	SM1588

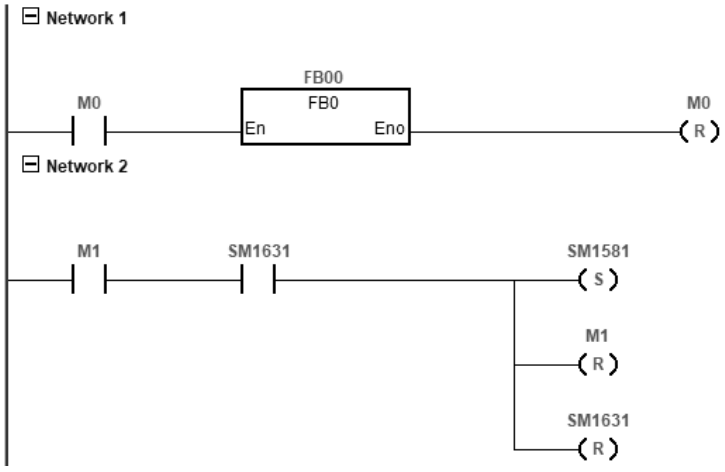
变频站号	21	22	23	24	25	26	27	28
复位标志	SM1601	SM1602	SM1603	SM1604	SM1605	SM1606	SM1607	SM1608

- 与自动复位相关之 CAN 通讯指令列表

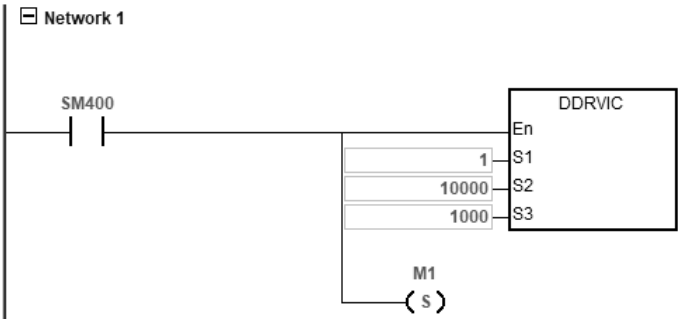
指令名称	伺服	变频	支持与否	复位时机点	补充说明
INITC	V	V	Yes	复位标志 ON	
ASDON	V	V			
CASD	V	V			
DDRVIC	V				
DDRVAC	V				
DPLSVC	V	V			
ZRNC/DZRNC	V				
RSTD	V	V			
DCSFOC	V				
DTQC	V	V			
DTQLC	V	V			
ZRNM	V		No		每次都须扫描与监控执行情形，更新完成/错误标志。
EMER	V	V			
COPRW	V	V			
COPWL/DCOPWL	V	V			
ECAMD	V				
DECAMS	V				
ECAMA	V				
ECAMC	V				

程序范例 1（使用功能块）

POU：



FB：



动作说明：

A. 当 M0=ON 时，触发执行一次性功能块，伺服#1 以 100.0 r/min 速度移动到相对位置 10000 PUU，并且设定 M1=ON，用来记录伺服轴#1 通讯控制权已经被使用。

B. 当伺服到达定位 SM1631=ON 并且 M1=ON 时，触发伺服#1 的复位标志 SM1581=ON，通知 PLC 释放伺服#1 通讯控制权。

C. 当 M0 再次由 OFF->ON 时，伺服#1 即会重新定位。

程序范例 2（使用 ST 语言）

```
0001 IF M0 THEN
0002     DDRVIC(1, 10000, 1000);
0003     M1 := TRUE;
0004 ELSIF M1 AND SM1631 THEN
0005     SM1581 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
0008
```

动作说明：

A. 当 M0=ON 时，伺服#1 以 100.0 r/min 速度移动到相对位置 10000 PUU，并且设定 M1=ON，用来记录伺服轴#1 通讯控制权已经被使用。

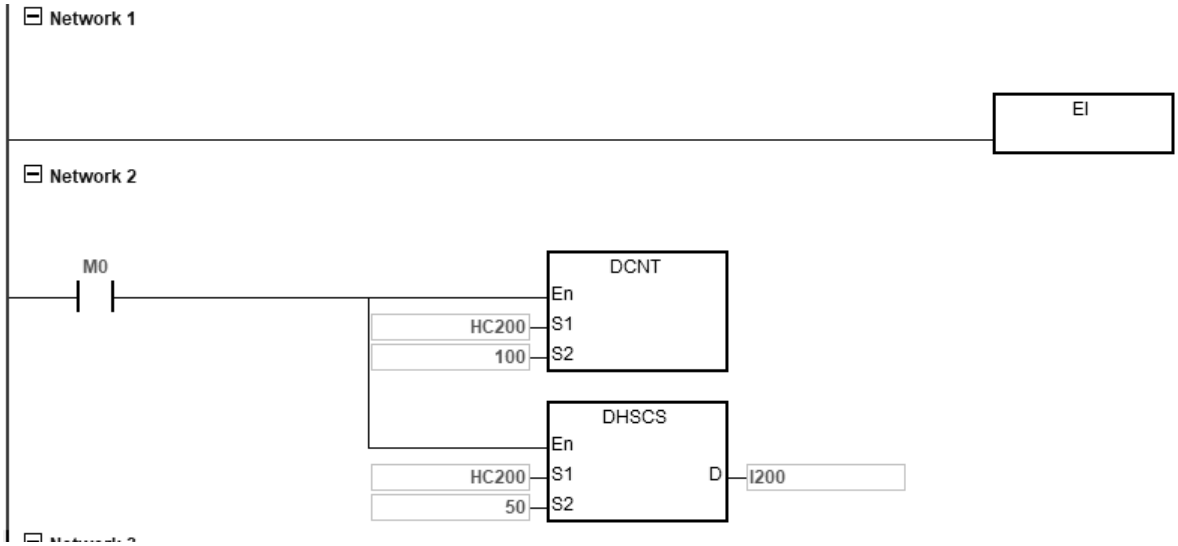
B. 当 M0=OFF 时，伺服到达定位 SM1631=ON 并且 M1=ON 时，触发伺服#1 的复位标志 SM1581=ON，通知 PLC 释放伺服#1 通讯控制权。

C. 当 M0 再次由 OFF->ON 时，伺服#1 即会重新定位。

25. 高速比较中断（I2xx）启动 DPLSY 输出说明（支持版本：V1.08.70 版（含）以上）

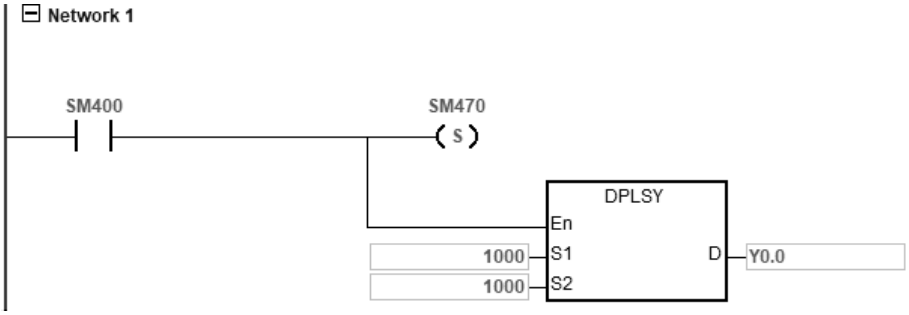
一般使用方式，会是在高速比较中断程序内调用 DPLSY 输出，范例 PLC 程序写法如下：

- 先在周期性程序内启动高速计数与高速比较指令



程序补充说明：M0 为 ON 时，启动 HC200 输入计数，当计数值到达 50，触发高速比较中断 I200。

于高速比较中断 I200 程序内写入 DPLSY 指令：



程序补充说明：Y0.0 输出 1000 个 1KHz 脉冲。设定复位标志，当输出完毕后，自动释放脉冲输出控制权。

- 使用情境说明：当输入计数值由 49 -> 50，到实际 Y0.0 输出，会经过以下 A~D 四个顺序流程：

  A. 输入信号计数，触发高速比较中断

  B. 插断 PLC 程序，执行高速比较中断子程序

  C. 执行 DPLSY 指令

  D. Y0.0 实际输出

  由于 A->B 流程，将依输入计数触发的时间点，而受到当下执行的 PLC 指令、更新 IO 点或通讯处理等系统执行流程影响，间接造成 A->D 的响应时间每次误差范围大。

- 为改善上述之情况，AS 主机新增下表 SM 标志，加快 A->D 的处理流程。

高速比较 中断编号	DPLSY 输出参数 预先暂存标志#1	比较中断触发 DPLSY 输出标志#2
I200	SM120	SM360
I201	SM121	SM361
I202	SM122	SM362
I203	SM123	SM363

高速比较 中断编号	DPLSY 输出参数 预先暂存标志#1	比较中断触发 DPLSY 输出标志#2
I210	SM124	SM364
I211	SM125	SM365
I212	SM126	SM366
I213	SM127	SM367
I220	SM128	SM368
I221	SM129	SM369
I222	SM130	SM370
I223	SM131	SM371
I230	SM132	SM372
I231	SM133	SM373
I232	SM134	SM374
I233	SM135	SM375
I240	SM136	SM376
I241	SM137	SM377
I242	SM138	SM378
I243	SM139	SM379
I250	SM140	SM380
I251	SM141	SM381
I252	SM142	SM382
I253	SM143	SM383
I260	SM144	SM384
I261	SM145	SM385
I262	SM146	SM386
I263	SM147	SM387
I264	SM148	SM388
I265	SM149	SM389
I266	SM150	SM390
I267	SM151	SM391

注#1：DPLSY 输出参数预先暂存标志 SM120~SM151。

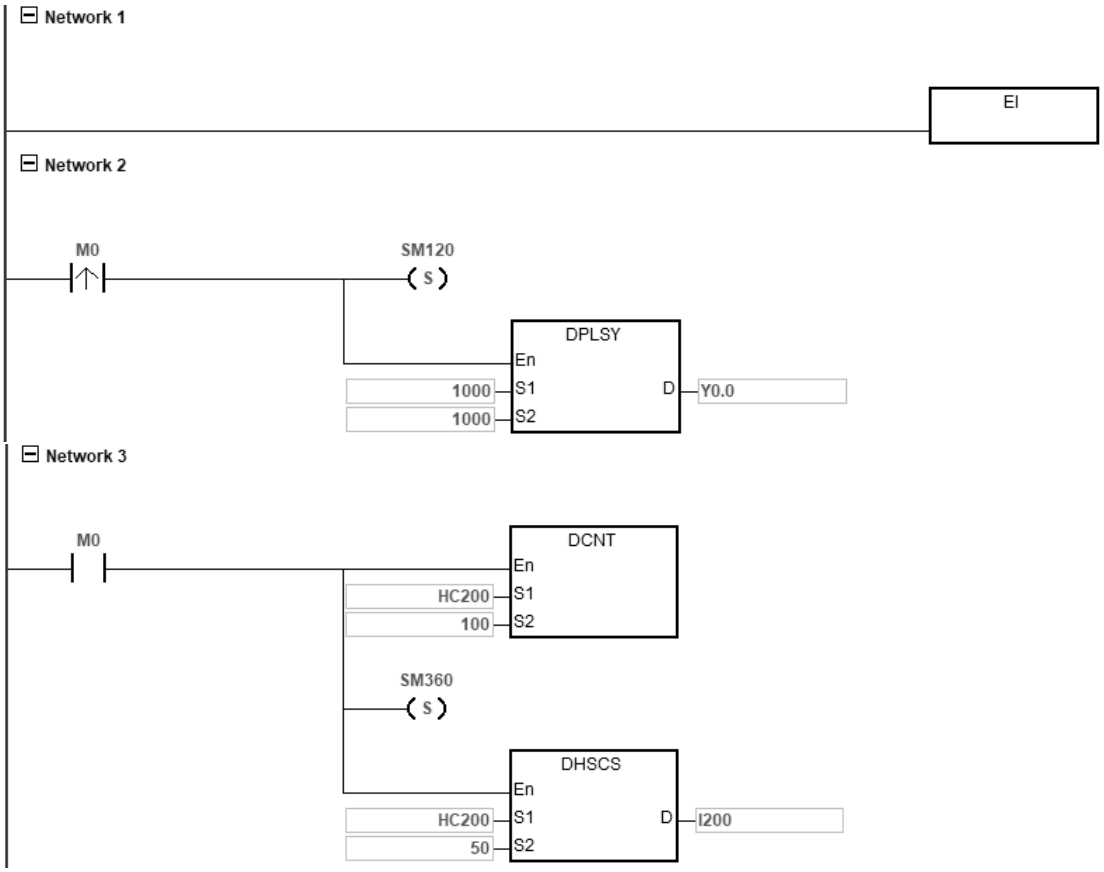
用户可通过此 SM 设为 ON，预先暂存 DPLSY 指令输出参数（包含输出点编号、目标频率与个数）至内存内存，等待高速比较中断被触发且有设定启动输出标志 SM 时，将会立即输出。此预先暂存标志被储存之后，将自动清除该标志。每次变更输出暂存参数时，需重新设定预先暂存标志为 ON。

注#2：高速比较中断启动 DPLSY 输出标志 SM360~SM391。

指定高速比较中断（I2XX）发生时，若对应之启动输出标志为 ON，将会立即依之前暂存的 DPLSY 输出参数执行脉冲输出，并等待脉冲输出完毕后，会自动释放脉冲输出控制权。

范例 PLC 程序写法如下：

只需在周期性程序内设定好参数即可



程序补充说明 :M0 为 ON 触发设定 SM120 为 ON ，将 DPLSY 输出参数(Y0.0 输出轴 ，输出 1000 个 1KHz 脉冲) 暂存；接着启动 HC200 输入计数，设定高速比较中断启动 DPLSY 输出标志；设定计数值到达 50，触发高速比较中断 I200。之后当高速比较中断被触发时，那 Y0.0 将输出 1000 个 1KHz 脉冲。

高速比较中断 I200 子程序：

程序补充说明：已经都在一般 POU 程序区完成，故不需针对高速输出编辑任何程序。可另外编辑其它指令，如：计数中断发生次数。

26. CSFO 指令之追随落后补偿使用说明（支持版本：V1.08.30 版（含）以上）

执行 CSFO 指令时，PLC 会计算输入的速度与个数，然后经由比率转换公式，换算出对应之输出速度与个数，由于从抓取到追随过程，会有一定的响应时间，进而造成追随落后的现象，尤其当输入速度越来越快时，就会造成追随输出的落后距离就越长。

因此 ，当有需要长时间追随 ，并且需求追随输出位置尽量同步来源轴 ，则可使用下表所列之追随落后补偿 SM/SR

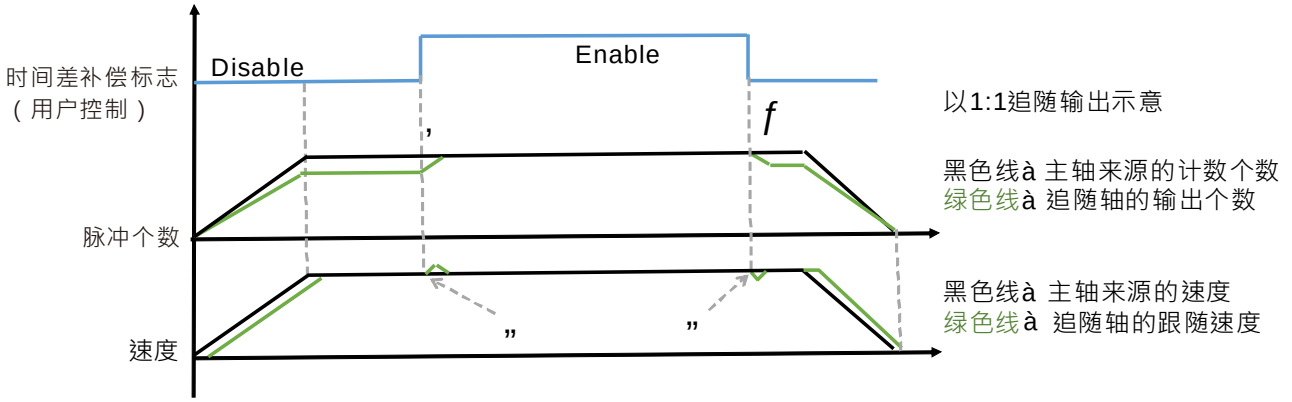
改善：

CSFO 指令输出轴	启动补偿标志	补偿速度（Hz）
Y0.0 / Y0.1	SM1700	SR1700
Y0.2 / Y0.3	SM1701	SR1702
Y0.4 / Y0.5	SM1702	SR1704
Y0.6 / Y0.7	SM1703	SR1706
Y0.8 / Y0.9	SM1704	SR1708
Y0.10 / Y0.11	SM1705	SR1710

- 补偿速度 SR：由用户设定，数值范围为 50Hz~10KHz（基本单位为 50Hz）。在补偿过程中，此速度值为每次计算追随输出时，所需增减输出速度的依据，当补偿至同步来源轴的位置时，PLC 将自动停止补偿动作。为避免瞬间补偿过程，可能发生加减速过快而影响机械运作，建议用户从最低速度 50Hz 开始尝试与验证。注意：由于基本单位是 50Hz，若输入 80Hz 将视为 50Hz；输入 120Hz 时视为 100Hz。

- 启动补偿标志 SM：标志为 ON 时，依落后距离开始补偿速度去追随，增加实际输出速度和个数，直到输出位置同步；当标志为 OFF 时，会依时间补偿速度，将减少输出速度和个数，直到恢复原先落后之状况。注意：此启动补偿标志，建议于停止追随前先关闭补偿标志，避免在脉冲输入停止时，还有补偿输出个数存在，造成多次启动而累积很多补偿输出个数的误差。

下图为落后补偿之输出曲线示意图，其输入/输出个数比例为 1：1。



上图标示的时机点说明：

①：还未启动落后补偿功能，输出个数会因为输入速度越快，而落后越长距离

②：启动补偿标志，依据输入速度和补偿速度，增加输出个数，直到输出位置同步。

③：关闭补偿标志，依据输入速度和补偿速度，减少输出个数，直到恢复至正常落后输出之原状态。

④：当"启动/关闭"补偿功能时，PLC 会将换算的输出速度"增加/减少"补偿速度输出，以利补偿功能平滑的执行。

举例说明启动时状况：假设当次追随速度为 1KHz，以及补偿速度为 100Hz，则当次实际输出速度有可能为 1.1KHz 输出。

2.2.17 变址寄存器 E

变址寄存器E是16位的数据寄存器，跟一般的寄存器一样可以被读、写，但主要功能是做变址寄存器使用，使用范围为E0~E9。变址的使用方式，请参考第4.4节变址说明。

2.2.18 文件寄存器 FR

AS系列针对大量参数储存的需求，提供文件寄存器（FR）供用户储存大量的参数数据。

用户可通过ISPSoft上的文件寄存器编辑功能，进行参数的编辑、上载与下载功能。

PLC程序执行过程中，可直接读取FR的内容值，若有需要写入FR时，请参考API2303 MEMW指令说明。

MEMO

3

第3章 指令表

目录

3.1 指令类型 3-2
3.1.1 基本指令 3-2
3.1.2 应用指令 3-2
3.2 指令表说明 3-3
3.2.1 基本指令 3-3
3.2.2 应用指令（依 API 号码排序） 3-4
3.2.3 应用指令（依英文字母排序） 3-4
3.2.4 指令装置表说明 3-5
3.3 基本指令一览表 3-7
3.4 应用指令一览表 3-10
3.4.1 应用指令（依 API 号码排序） 3-10
3.4.2 应用指令（依英文字母排序） 3-47

3.1 指令类型

AS 指令的主要类型包括顺序基本指令、应用指令。

3.1.1 基本指令

分类	说明
接点指令	载入接点，串联接点，并联接点...等
输出指令	位设备输出，脉冲输出
主控制指令	控制共通串联接点
高低电平检出接点指令	触发加载接点，串联接点，并联接点的相关指令
上下微分输出指令	位设备微分输出
其它指令	其它指令

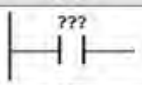
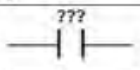
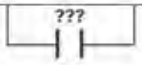
3.1.2 应用指令

API	分类	说明
0000~0083	比较操作指令	比较操作如=、<>、>、>=、<、<=...等
0100~0118	四则运算指令	BIN 或 BCD 的加法、减法、乘法或除法
0200~0222	数据转换指令	将 BCD 转换成 BIN 和将 BIN 转换成 BCD
0300~0310	数据转移指令	传送指定的数据
0400~0402	程序跳转指令	程序跳转
0500~0504	程序执行控制指令	允许或禁止中断程序
0600~0602	I/O 更新指令	运行局部更新
0700~0711	便利指令	用于以下目的的指令：计数器增加/减小，教学定时器，特殊功能定时器，循环台最短距离控制...等
0800~0817	逻辑操作指令	逻辑操作，如逻辑加法，逻辑乘法等
0900~0904	循环指令	指定数据的循环移位
1000~1018	计时计数指令	定时器，计数器 指令
1100~1115	移位指令	指定数据的移位
1200~1229	数据处理指令	16 位数据查询，数据处理如译码和编码
1300~1302	结构建立指令	循环回路，调用功能块...等
1400~1430	模块的数据读/写指令	特殊功能模块的数据读/写
1500~1517	浮点数指令	浮点数运算指令
1600~1608	万年历指令	万年历时间（年，月，日，小时，分，秒和星期）的读取、更新与比较
1700~1704	外围设备指令	连接到外围设备的 I/O
1806~1821	通讯指令	通讯控制接口设备
1900~1906	其它指令	其它不符合以上范畴的指令，如通讯逾时重置指令和定时时钟指令

API	分类	说明
2100~2123	字符串处理指令	CBIN/BCD 和 ASCII 之间的转换；BIN 和字符串之间的转换；浮点十进制数和字符串之间的转换、字符串处理等等
2200~2216	以太网控制指令	控制以太网数据交换相关指令
2300~2305	储存卡/文件寄存器读写指令	储存卡/文件寄存器读写相关指令
2400~2401	任务控制指令	程序中任务控制相关指令
2500~2502	SFC 控制指令	控制 SFC 相关指令
2700~2724	高速输出指令	高速输出与定位控制相关指令
2800~2818	台达专用 CANopen 与 EtherCAT 通讯指令	与台达产品 CANopen 与 EtherCAT 通讯相关指令

3.2 指令表说明

3.2.1 基本指令

指令码	符号	功能	操作数
LD		载入/串联/并联 A 接点	DX、X、Y、M、SM、S、T、C、HC、D
AND			
OR			

① ② ③ ④

说明：

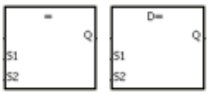
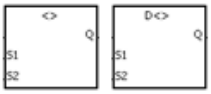
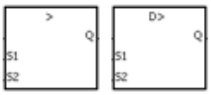
①：指令的名称

②：在 ISPSoft 梯形图所显示的图形

③：指令的功能用途说明

④：指令所支持的操作数

3.2.2 应用指令（依 API 号码排序）

API	指令码		P 指令	符号	功能
	16 位	32 位			
0000	LD=	DLD=	–		接点类型比较 S1＝S2 导通 S1≠S2 不导通
0001	LD<>	DLD<>	–		接点类型比较 S1≠S2 导通 S1＝S2 不导通
0002	LD>	DLD>	–		接点类型比较 S1＞S2 导通 S1≦S2 不导通

① ② ③ ④ ⑤ ⑥

说明：

①：指令 API 编号。

②：指令的名称。

③：16 位的指令，如有支持 32 位，则在指令前加 D，即为 32 位指令。

④：是否有 PULSE 指令，有：✓。无：–。

有 PULSE 时，要使用 PULSE 指令，只需在指令后加 P。

⑤：在 ISPSoft 梯形图所显示的图形。

⑥：指令的功能用途说明。

3.2.3 应用指令（依英文字母排序）

分类	API	指令码（位）		P 指令	功能
		16	32		
F	1508	–	FTANH	✓	浮点数 TANH 运算
	0057	–	FZCP	✓	浮点数区域比较
	0202	FLT	DFLT	✓	BIN 整数→单精度浮点数变换
	1300	FOR	–	–	循环回路起始
	1400	FROM	DFROM	✓	扩展模块 CR 数据读出
G	0209	GBIN	DGBIN	✓	GRY 码→BIN 变换
	0402	GOEND	–	–	跳转到 END
	1902	GPWM	–	–	一般用脉冲宽度调变
	0208	GRY	DGRY	✓	BIN→GRY 码变换

① ② ③ ④ ⑤ ⑥

说明：

①：指令名称开头英文字母的分类。

②：指令 API 编号。

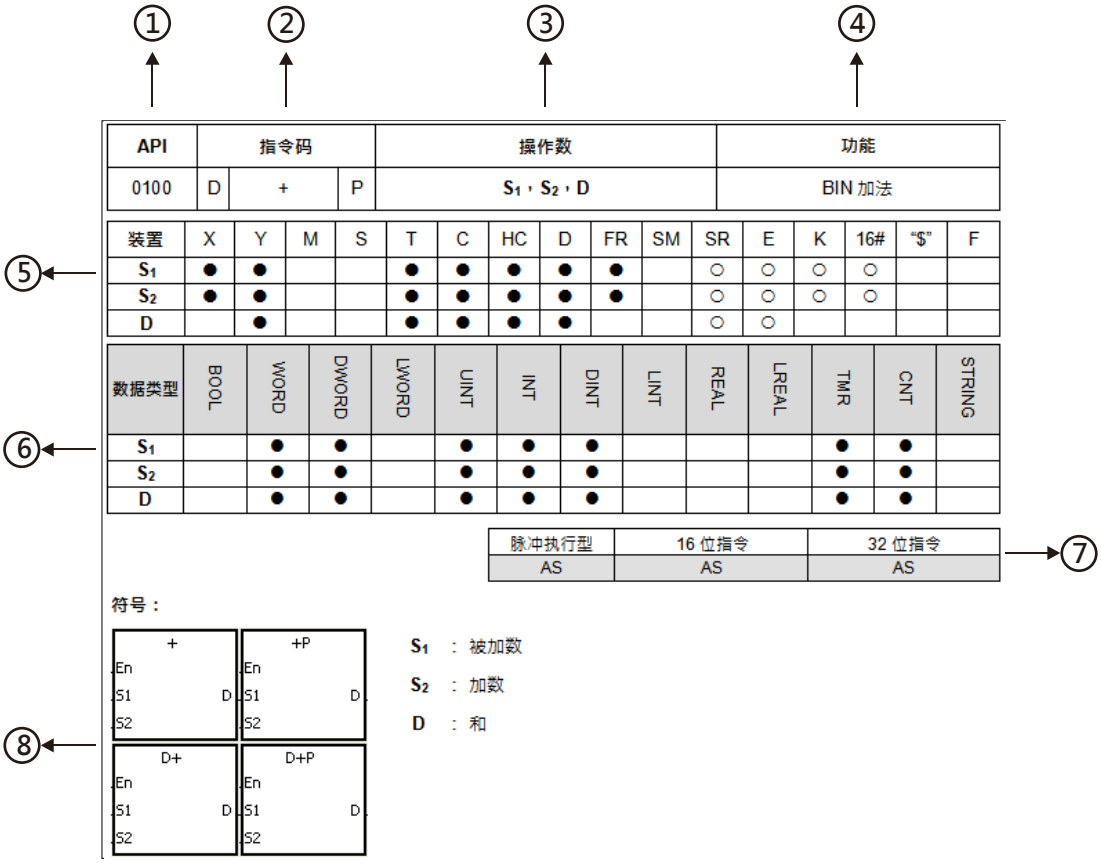
③～④：指令的名称。16 位的指令，如有支持 32 位，则在指令前加 D，即为 32 位指令。

⑤：是否有脉冲型指令（P 指令），有：✓。无：—。

要使用脉冲型指令，需在指令后加 P。

⑥：指令的功能用途说明。

3.2.4 指令装置表说明



说明：

①：指令 API 编号。

②：指令的名称。16 位的指令，如有支持 32 位，则在指令前加 D，即为 32 位指令。有脉冲执行型的指令则会在指令名称后有 P。

③：指令的操作数。

④：指令的功能用途说明。

⑤：表示指令每个操作数所支持的装置

1. 其中 K 表示为十进制，在 ISPSoft 中直接输入数字，如：要输入 30，在 ISPSoft 中需输入 30。
2. 16#表示为十六进制，如：要输入 30，在 ISPSoft 中需输入 16#1E。
3. F/DF：浮点数的表示方式，但在 ISPSoft 中是直接以小数点的方式来表示，例如要输入 F500 的浮点数，请直接输入 500.0。
4. "$" :字符串的表示方式 ,在 ISPSoft 中是以" "方式来表示 ,例如要输入字符串 1234 ,请直接输入"1234"。
5. ○：空心圆，表示此装置不支持 E 修饰。

6. ●：实心圆，表示此装置支持 E 修饰。

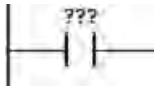
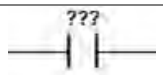
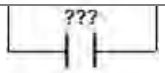
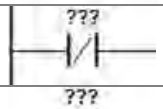
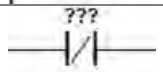
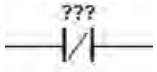
⑥：表示指令每个操作数所支持的数据类型

⑦：此指令之格式，是否有脉冲执行型、16 位及 32 位。

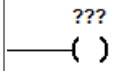
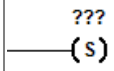
⑧：引脚示意图。

3.3 基本指令一览表

- 接点指令

指令码	符号	功能	操作数
LD		载入/串联/并联 A 接点	DX、X、Y、M、SM、S、T、C、HC、D
AND			
OR			
LDI		载入/串联/并联 B 接点	DX、X、Y、M、SM、S、T、C、HC、D
ANI			
ORI			

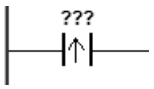
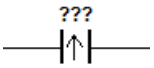
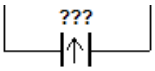
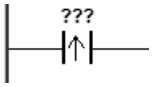
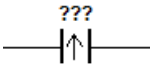
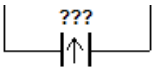
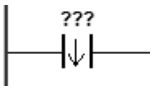
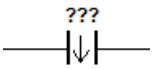
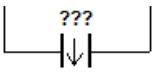
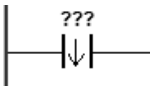
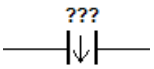
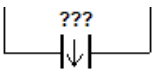
- 输出指令

指令码	符号	功能	执行条件	操作数
OUT		驱动线圈		DY、Y、M、SM、S、T、C、HC、D
SET		动作保持(ON)		DY、Y、M、SM、S、T、C、HC、D

- 主控制指令

指令码	符号	功能	操作数
MC		共通串联接点的连结	N
MCR		共通串联接点的解除	N

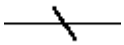
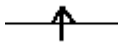
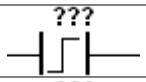
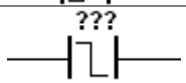
- 高低电平检出接点指令

指令码	符号	功能	执行条件	操作数
LDP		上升沿检出动作开始/串联/并联连接		DX、X、Y、M、SM、S、T、C、HC、D
ANDP				
ORP				
PED		上升沿检出动作开始/串联/并联连接		X、Y、M、SM、S、T、C、HC、D
APED				
OPED				
LDF		下降沿检出动作开始/串联/并联连接		DX、X、Y、M、SM、S、T、C、HC、D
ANDF				
ORF				
NED		下降沿检出动作开始/串联/并联连接		X、Y、M、SM、S、T、C、HC、D
ANED				
ONED				

- 上下微分输出指令

指令码	符号	功能	执行条件	操作数
PLS		上微分输出		Y、M、SM、S
PLF		下微分输出		Y、M、SM、S

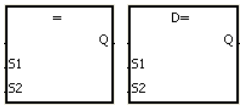
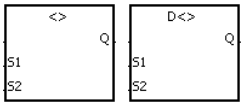
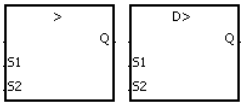
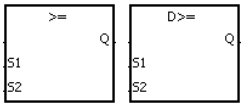
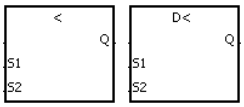
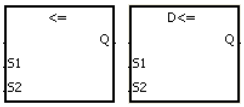
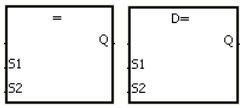
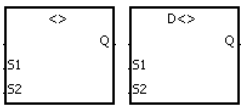
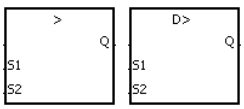
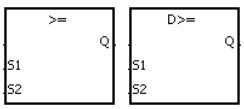
- 其它指令

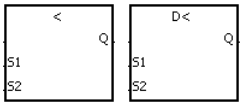
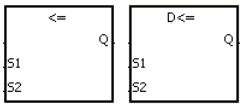
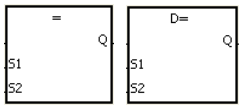
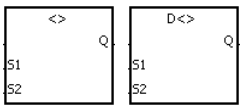
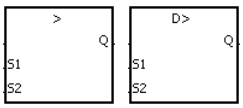
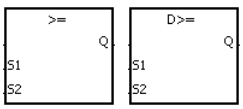
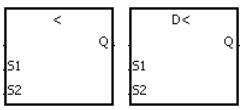
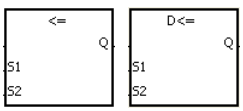
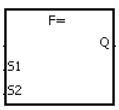
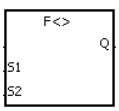
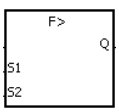
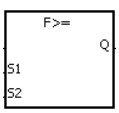
指令码	符号	功能	操作数
INV		运算结果反相	—
NP		上升沿触发指令	—
PN		下降沿触发指令	—
FB_NP		上升沿触发指令	Y、M、S、D
FB_PN		下降沿触发指令	Y、M、S、D

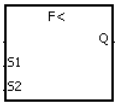
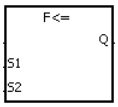
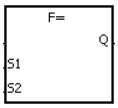
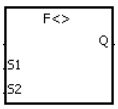
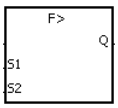
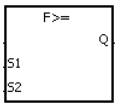
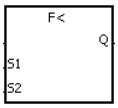
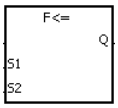
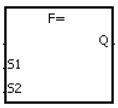
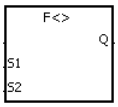
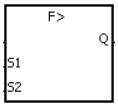
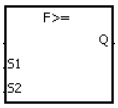
3.4 应用指令一览表

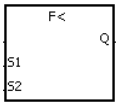
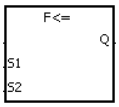
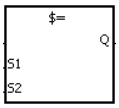
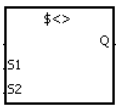
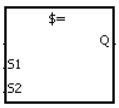
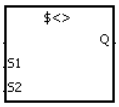
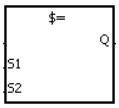
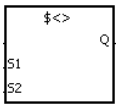
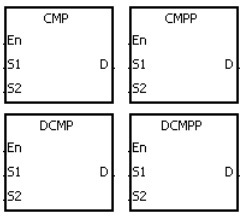
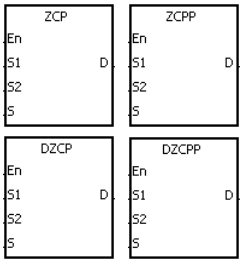
3.4.1 应用指令（依 API 号码排序）

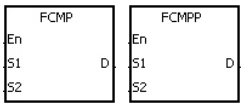
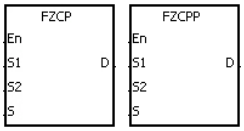
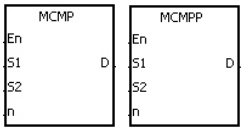
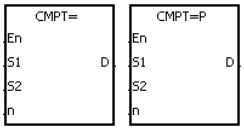
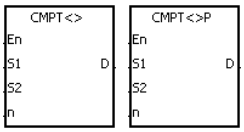
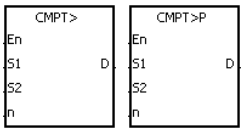
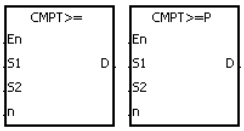
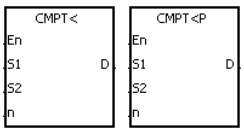
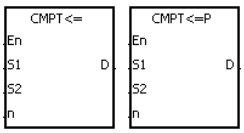
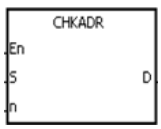
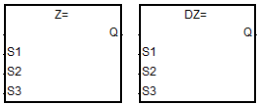
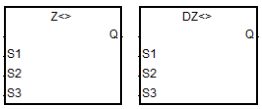
- 比较操作指令

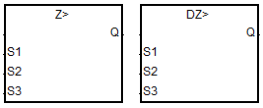
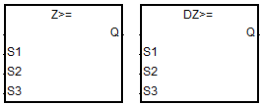
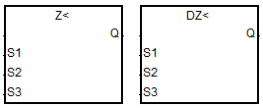
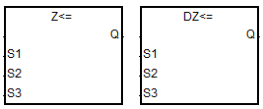
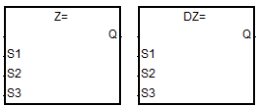
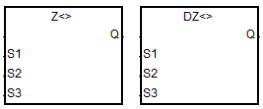
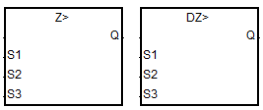
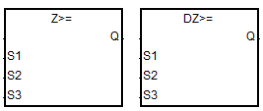
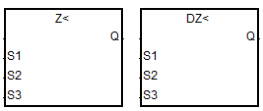
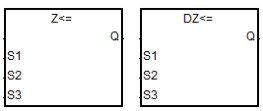
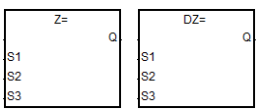
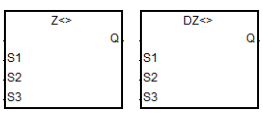
API	指令码		P 指令	符号	功能
	16 位	32 位			
0000	LD=	DLD=	—		接点类型比较 S1 = S2 导通 S1≠S2 不导通
0001	LD<>	DLD<>	—		接点类型比较 S1≠S2 导通 S1 = S2 不导通
0002	LD>	DLD>	—		接点类型比较 S1 > S2 导通 S1≦S2 不导通
0003	LD>=	DLD>=	—		接点类型比较 S1≧S2 导通 S1 < S2 不导通
0004	LD<	DLD<	—		接点类型比较 S1 < S2 导通 S1≧S2 不导通
0005	LD<=	DLD<=	—		接点类型比较 S1≦S2 导通 S1 > S2 不导通
0006	AND=	DAND=	—		串连接点类型比较 S1 = S2 导通 S1≠S2 不导通
0007	AND<>	DAND<>	—		串连接点类型比较 S1≠S2 导通 S1 = S2 不导通
0008	AND>	DAND>	—		串连接点类型比较 S1 > S2 导通 S1≦S2 不导通
0009	AND>=	DAND>=	—		串连接点类型比较 S1≧S2 导通 S1 < S2 不导通

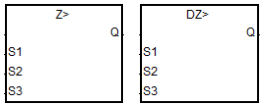
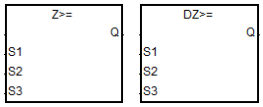
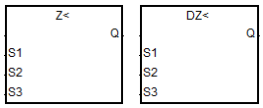
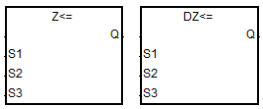
API	指令码		P 指令	符号	功能
	16 位	32 位			
0010	AND<	DAND<	—		串连接点类型比较 S1 < S2 导通 S1≧S2 不导通
0011	AND<=	DAND<=	—		串连接点类型比较 S1≦S2 导通 S1 > S2 不导通
0012	OR=	DOR=	—		并联接点类型比较 S1 = S2 导通 S1≠S2 不导通
0013	OR<>	DOR<>	—		并联接点类型比较 S1≠S2 导通 S1 = S2 不导通
0014	OR>	DOR>	—		并联接点类型比较 S1 > S2 导通 S1≦S2 不导通
0015	OR>=	DOR>=	—		并联接点类型比较 S1≧S2 导通 S1 < S2 不导通
0016	OR<	DOR<	—		并联接点类型比较 S1 < S2 导通 S1≧S2 不导通
0017	OR<=	DOR<=	—		并联接点类型比较 S1≦S2 导通 S1 > S2 不导通
0018	—	FLD=	—		浮点数接点类型比较 S1 = S2 导通 S1≠S2 不导通
0019	—	FLD<>	—		浮点数接点类型比较 S1≠S2 导通 S1 = S2 不导通
0020	—	FLD>	—		浮点数接点类型比较 S1 > S2 导通 S1≦S2 不导通
0021	—	FLD>=	—		浮点数接点类型比较 S1≧S2 导通 S1 < S2 不导通

API	指令码		P 指令	符号	功能
	16 位	32 位			
0022	—	FLD<	—		浮点数接点类型比较 S1 < S2 导通 S1≧S2 不导通
0023	—	FLD<=	—		浮点数接点类型比较 S1≦S2 导通 S1 > S2 不导通
0024	—	FAND=	—		浮点数串连接点类型比较 S1 = S2 导通 S1≠S2 不导通
0025	—	FAND<>	—		浮点数串连接点类型比较 S1≠S2 导通 S1 = S2 不导通
0026	—	FAND>	—		浮点数串连接点类型比较 S1 > S2 导通 S1≦S2 不导通
0027	—	FAND>=	—		浮点数串连接点类型比较 S1≧S2 导通 S1 < S2 不导通
0028	—	FAND<	—		浮点数串连接点类型比较 S1 < S2 导通 S1≧S2 不导通
0029	—	FAND<=	—		浮点数串连接点类型比较 S1≦S2 导通 S1 > S2 不导通
0030	—	FOR=	—		浮点数并连接点类型比较 S1 = S2 导通 S1≠S2 不导通
0031	—	FOR<>	—		浮点数并连接点类型比较 S1≠S2 导通 S1 = S2 不导通
0032	—	FOR>	—		浮点数并连接点类型比较 S1 > S2 导通 S1≦S2 不导通
0033	—	FOR>=	—		浮点数并连接点类型比较 S1≧S2 导通 S1 < S2 不导通

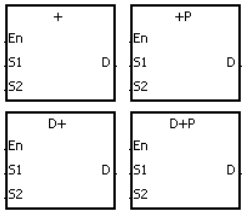
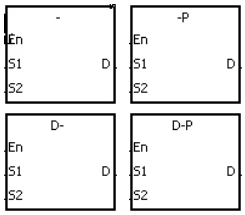
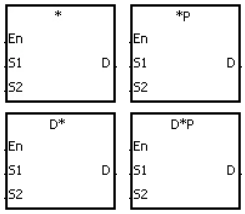
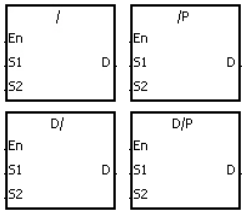
API	指令码		P 指令	符号	功能
	16 位	32 位			
0034	—	FOR<	—		浮点数并连接点类型比较 S1 < S2 导通 S1≧S2 不导通
0035	—	FOR<=	—		浮点数并连接点类型比较 S1≦S2 导通 S1 > S2 不导通
0036	LD$=	—	—		字符串接点类型比较 S1 = S2 导通 S1≠S2 不导通
0037	LD$<>	—	—		字符串接点类型比较 S1≠S2 导通 S1 = S2 不导通
0042	AND$=	—	—		字符串串连接点类型比较 S1 = S2 导通 S1≠S2 不导通
0043	AND$<>	—	—		字符串串连接点类型比较 S1≠S2 导通 S1 = S2 不导通
0048	OR$=	—	—		字符串并连接点类型比较 S1 = S2 导通 S1≠S2 不导通
0049	OR$<>	—	—		字符串并连接点类型比较 S1≠S2 导通 S1 = S2 不导通
0054	CMP	DCMP	✓		比较设定输出
0055	ZCP	DZCP	✓		区域比较

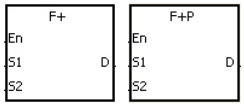
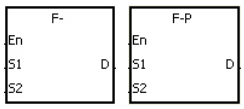
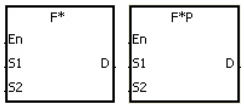
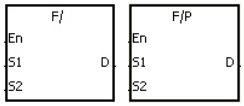
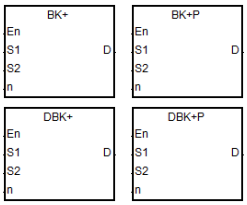
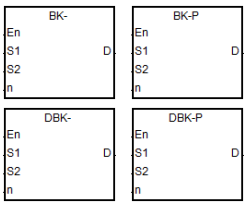
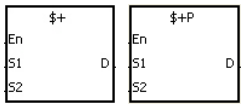
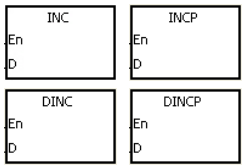
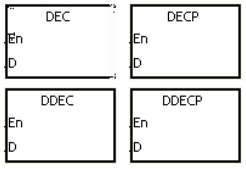
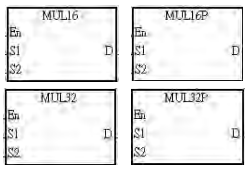
API	指令码		P 指令	符号	功能
	16 位	32 位			
0056	—	FCMP	✓		浮点数比较
0057	—	FZCP	✓		浮点数区域比较
0058	MCMP	—	✓		矩阵比较
0059	CMPT=	—	✓		表格比较，=
0060	CMPT<>	—	✓		表格比较，≠
0061	CMPT>	—	✓		表格比较，>
0062	CMPT>=	—	✓		表格比较，≧
0063	CMPT<	—	✓		表格比较，<
0064	CMPT<=	—	✓		表格比较，≦
0065	CHKADR	—	—		接点类型指针寄存器地址检查
0066	LDZ=	DLDZ=	—		接点类型绝对值比较 $\lvert S_1 - S_2 \rvert = \lvert S_3 \rvert$ 导通 $\lvert S_1 - S_2 \rvert \neq \lvert S_3 \rvert$ 不导通
0067	LDZ<>	DLDZ<>	—		接点类型绝对值比较 $\lvert S_1 - S_2 \rvert \neq \lvert S_3 \rvert$ 导通 $\lvert S_1 - S_2 \rvert = \lvert S_3 \rvert$ 不导通

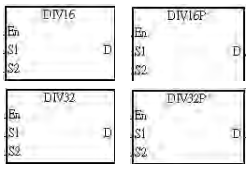
API	指令码		P 指令	符号	功能
	16 位	32 位			
0068	LDZ>	DLDZ>	—		接点类型绝对值比较 | S_1- S_2| > | S_3| 导通 | S_1- S_2| ≦ | S_3| 不导通
0069	LDZ>=	DLDZ>=	—		接点类型绝对值比较 | S_1- S_2|≧| S_3| 导通 | S_1- S_2| < | S_3| 不导通
0070	LDZ<	DLDZ<	—		接点类型绝对值比较 | S_1- S_2| < | S_3| 导通 | S_1- S_2|≧ | S_3| 不导通
0071	LDZ<=	DLDZ<=	—		接点类型绝对值比较 | S_1- S_2|≦| S_3| 导通 | S_1- S_2| > | S_3| 不导通
0072	ANDZ=	DANDZ=	—		接点类型绝对值比较 | S_1- S_2| = | S_3| 导通 | S_1- S_2| ≠ | S_3| 不导通
0073	ANDZ<>	DANDZ<>	—		接点类型绝对值比较 | S_1- S_2|≠| S_3| 导通 | S_1- S_2| = | S_3| 不导通
0074	ANDZ>	DANDZ>	—		接点类型绝对值比较 | S_1- S_2| > | S_3| 导通 | S_1- S_2| ≦ | S_3| 不导通
0075	ANDZ>=	DANDZ>=	—		接点类型绝对值比较 | S_1- S_2|≧| S_3| 导通 | S_1- S_2| < | S_3| 不导通
0076	ANDZ<	DANDZ<	—		接点类型绝对值比较 | S_1- S_2| < | S_3| 导通 | S_1- S_2|≧ | S_3| 不导通
0077	ANDZ<=	DANDZ<=	—		接点类型绝对值比较 | S_1- S_2|≦| S_3| 导通 | S_1- S_2| > | S_3| 不导通
0078	ORZ=	DORZ=	—		接点类型绝对值比较 | S_1- S_2| = | S_3| 导通 | S_1- S_2| ≠ | S_3| 不导通
0079	ORZ<>	DORZ<>	—		接点类型绝对值比较 | S_1- S_2| ≠ | S_3| 导通 | S_1- S_2| = | S_3| 不导通

API	指令码		P 指令	符号	功能
	16 位	32 位			
0080	ORZ>	DORZ>	—		接点类型绝对值比较 | S_1- S_2| > | S_3| 导通 | S_1- S_2| ≦ | S_3| 不导通
0081	ORZ>=	DORZ>=	—		接点类型绝对值比较 | S_1- S_2| ≧ | S_3| 导通 | S_1- S_2| < | S_3| 不导通
0082	ORZ<	DORZ<	—		接点类型绝对值比较 | S_1- S_2| < | S_3| 导通 | S_1- S_2| ≧ | S_3| 不导通
0083	ORZ<=	DORZ<=	—		接点类型绝对值比较 | S_1- S_2| ≦ | S_3| 导通 | S_1- S_2| > | S_3| 不导通

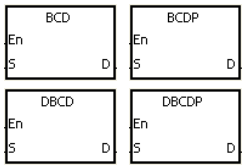
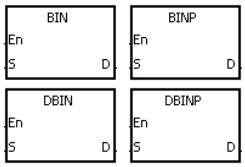
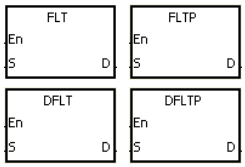
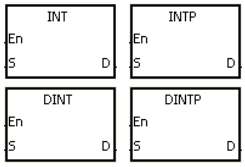
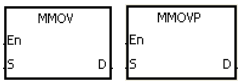
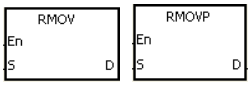
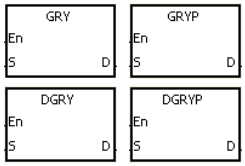
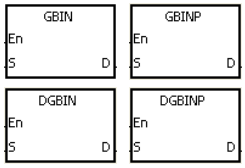
- 四则运算指令

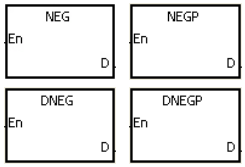
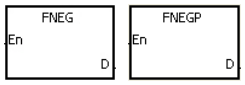
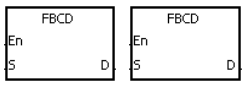
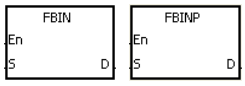
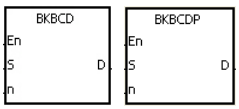
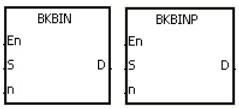
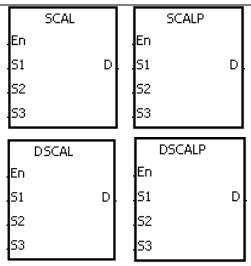
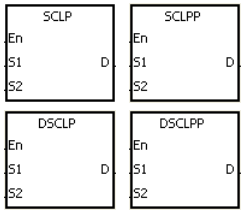
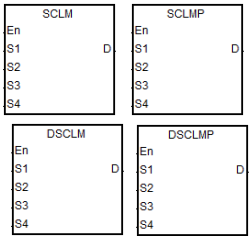
API	指令码		P 指令	符号	功能
	16 位	32 位			
0100	+	D+	✓		BIN 加法 S1+S2=D
0101	-	D-	✓		BIN 减法 S1-S2=D
0102	*	D*	✓		BIN 乘法 S1*S2=D
0103	/	D/	✓		BIN 除法 S1/S2=D

API	指令码		P 指令	符号	功能
	16 位	32 位			
0104	—	F+	✓		浮点数加法 S1+S2=D
0105	—	F-	✓		浮点数减法 S1-S2=D
0106	—	F*	✓		浮点数乘法 S1*S2=D
0107	—	F/	✓		浮点数除法 S1/S2=D
0112	BK+	DBK+	✓		连续区块 BIN 加法
0113	BK-	DBK-	✓		连续区块 BIN 减法
0114	$+	—	✓		字符串链结
0115	INC	DINC	✓		BIN 加一
0116	DEC	DDEC	✓		BIN 减一
0117	MUL16	MUL32	✓		16 位专用 BIN 乘法/ 32 位专用 BIN 乘法

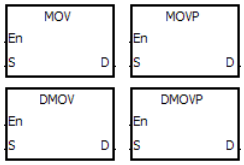
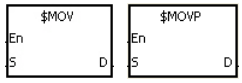
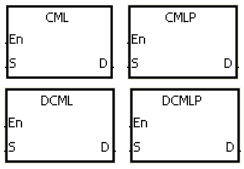
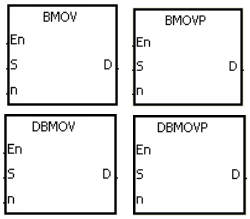
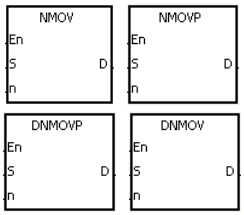
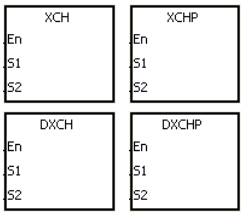
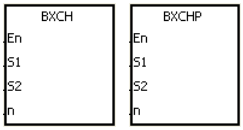
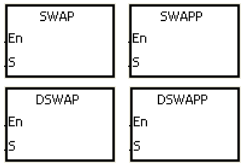
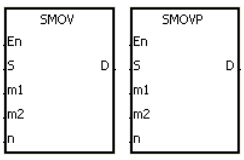
API	指令码		P 指令	符号	功能
	16 位	32 位			
0118	DIV16	DIV32	✓		16 位专用 BIN 除法/ 32 位专用 BIN 除法

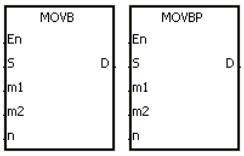
- 数据转换指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
0200	BCD	DBCD	✓		BIN→BCD 变换
0201	BIN	DBIN	✓		BCD→BIN 变换
0202	FLT	DFLT	✓		BIN 整数→二进浮点数变换
0204	INT	DINT	✓		二进浮点数→BIN 整数变换
0206	MMOV	—	✓		16→32 位数值转换
0207	RMOV	—	✓		32→16 位数值转换
0208	GRY	DGRY	✓		BIN→GRY 码变换
0209	GBIN	DGBIN	✓		GRY 码→BIN 变换

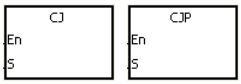
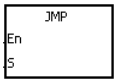
API	指令码		P 指令	符号	功能
	16 位	32 位			
0210	NEG	DNEG	✓		取负数（取 2 的补码）
0211	—	FNEG	✓		32 位浮点正负符号反相
0212	—	FBCD	✓		二进浮点数→十进浮点数
0213	—	FBIN	✓		十进浮点数→二进浮点数
0214	BKBCD	—	✓		连续区块 BIN→BCD 变换
0215	BKBIN	—	✓		连续区块 BCD→BIN 变换
0216	SCAL	DSCAL	✓		比例运算
0217	SCLP	DSCLP	✓		参数型比例运算
0222	SCLM	DSCLM	✓		多点式区域比例运算

- 数据转移指令

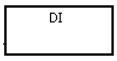
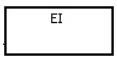
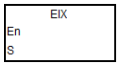
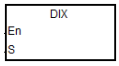
API	指令码		P 指令	符号	功能
	16 位	32 位			
0300	MOV	DMOV	✓		数据移动 S：数据源 D：数据目的地
0302	$MOV	—	✓		字符串移动
0303	CML	DCML	✓		反转传送
0304	BMOV	DBMOV	✓		全部传送
0305	NMOV	DNMOV	✓		多点移动
0306	XCH	DXCH	✓		数据的交换
0307	BXCH	—	✓		全部交换
0308	SWAP	DSWAP	✓		上/下 BYTE 变换
0309	SMOV	—	✓		位数移动

API	指令码		P 指令	符号	功能
	16 位	32 位			
0310	MOVB	—	✓		多位移动

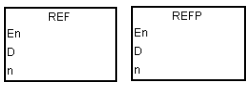
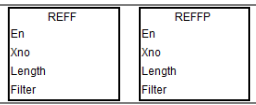
- 程序跳转指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
0400	CJ	—	✓		条件跳转
0401	JMP	—	—		无条件跳转
0402	GOEND	—	—		跳转到 END

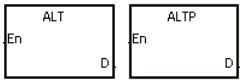
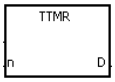
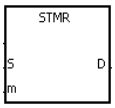
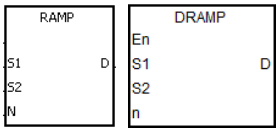
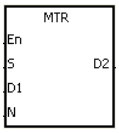
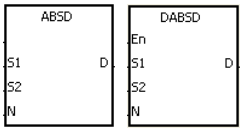
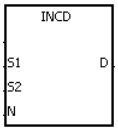
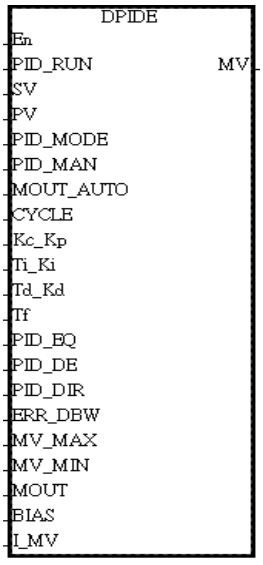
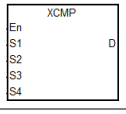
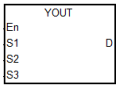
- 程序执行控制指令

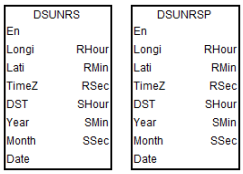
API	指令码		P 指令	符号	功能
	16 位	32 位			
0500	DI	—	—		全部中断插入关闭
0501	EI	—	—		全部中断插入启动
0503	EIX	—	—		指定中断启动
0504	DIX	—	—		指定中断关闭

- I/O 更新指令

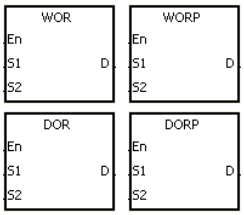
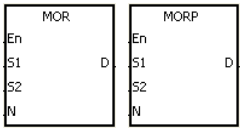
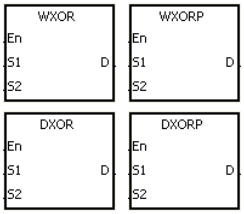
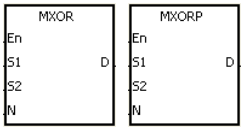
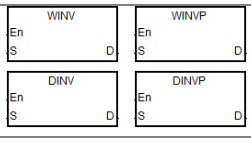
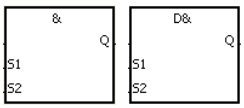
API	指令码		P 指令	符号	功能
	16 位	32 位			
0600	REF	—	✓		I/O 更新处理
0601	—	DHSRF	✓		高速比较值立即刷新
0602	REFF	—	✓		更新输入点滤波时间

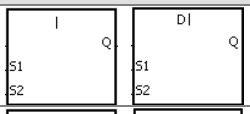
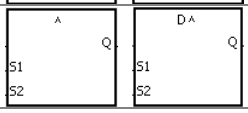
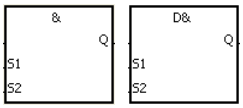
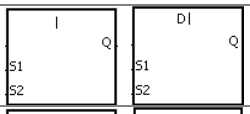
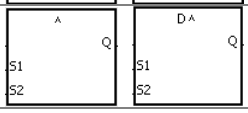
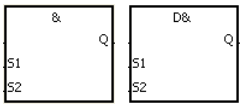
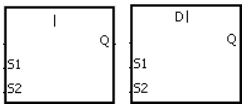
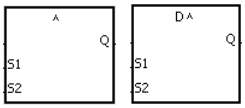
- 便利指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
0700	ALT	—	✓		ON/OFF 交替
0701	TTMR	—	—		教导式定时器
0702	STMR	—	—		特殊定时器
0703	RAMP	DRAMP	—		倾斜信号
0704	MTR	—	—		矩阵输入
0705	ABSD	DABSD	—		绝对方式凸轮控制
0706	INCD	—	—		相对方式凸轮控制
0708	—	DPIDE	—		PID 运算
0709	XCMP	—	—		多任务位输入比较设定
0710	YOUT	—	—		多任务位比较与输出

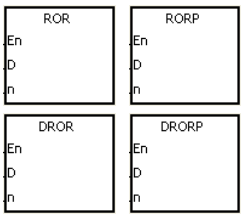
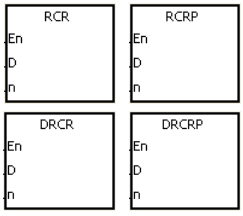
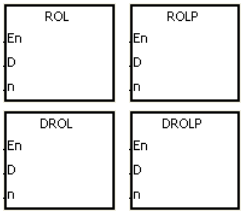
API	指令码		P 指令	符号	功能
	16 位	32 位			
0711	—	DSUNRS	✓		日出与日落时间

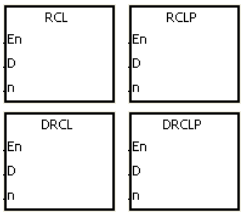
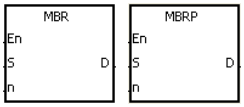
- 逻辑操作指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
0800	WAND	DAND	✓		逻辑与（AND）运算
0801	MAND	—	✓		矩阵与（AND）运算
0802	WOR	DOR	✓		逻辑或（OR）运算
0803	MOR	—	✓		矩阵或（OR）运算
0804	WXOR	DXOR	✓		逻辑异或（XOR）运算
0805	MXOR	—	✓		矩阵异或（XOR）运算
0808	WINV	DINV	✓		反向或（INV）运算
0809	LD&	DLD&	—		S1&S2

API	指令码		P 指令	符号	功能
	16 位	32 位			
0810	LD|	DLD|	—		S1|S2
0811	LD^	DLD^	—		S1^S2
0812	AND&	DAND&	—		S1&S2
0813	AND|	DAND|	—		S1|S2
0814	AND^	DAND^	—		S1^S2
0815	OR&	DOR&	—		S1&S2
0816	OR|	DOR|	—		S1|S2
0817	OR^	DOR^	—		S1^S2

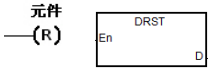
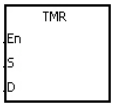
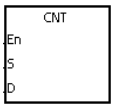
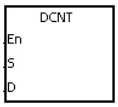
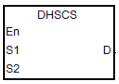
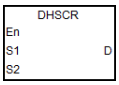
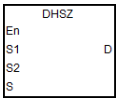
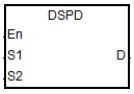
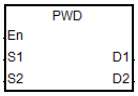
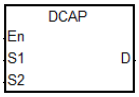
- 循环指令

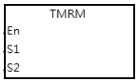
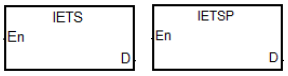
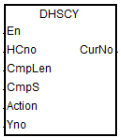
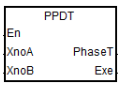
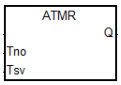
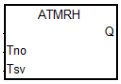
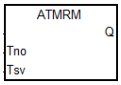
API	指令码		P 指令	符号	功能
	16 位	32 位			
0900	ROR	DROR	✓		右循环
0901	RCR	DRCR	✓		附进位标志右循环
0902	ROL	DROL	✓		欲循环的装置

API	指令码		P 指令	符号	功能
	16 位	32 位			
0903	RCL	DRCL	✓		附进位标志左循环
0904	MBR	—	✓		矩阵位循环

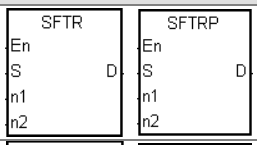
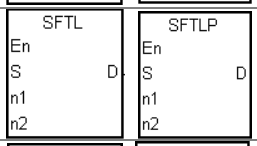
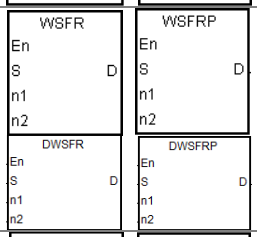
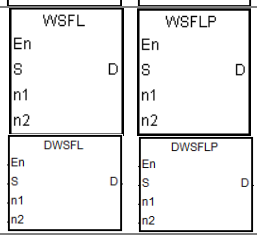
3

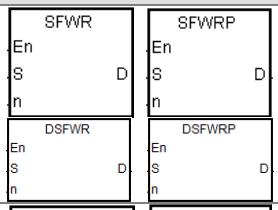
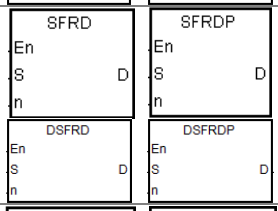
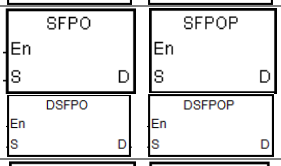
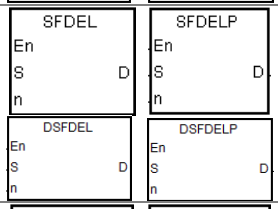
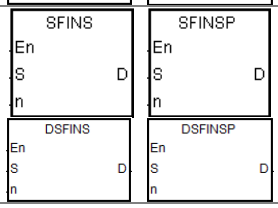
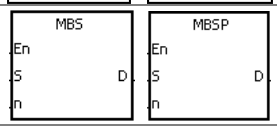
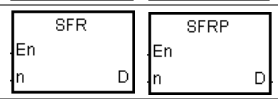
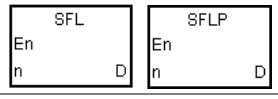
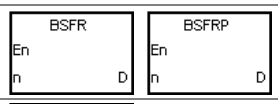
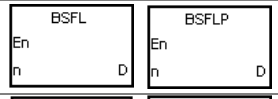
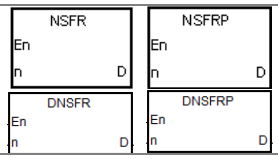
- 计时计数指令

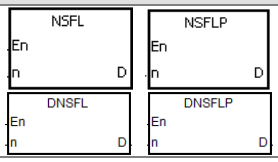
API	指令码		P 指令	符号	功能
	16 位	32 位			
1000	RST	DRST	—		接点或寄存器清除
1001	TMR	—	—		16 位定时器（单位：100ms）
1002	TMRH	—	—		16 位定时器（单位：1ms）
1003	CNT	—	—		16 位计数器
1004	—	DCNT	—		32 位计数器 （包含高速计数器使用说明）
1005	—	DHSCS	—		高速比较设定
1006	—	DHSCR	—		高速比较清除
1007	—	DHSZ	—		高速区域比较
1008	—	DSPD	—		速度检测指令
1009	PWD	—	—		脉宽检测指令
1010	—	DCAP	—		外部输入中断获取高速计数值

API	指令码		P 指令	符号	功能
	16 位	32 位			
1011	TMRM	—	—		16 位定时器（单位：10ms）
1012	IETS	—	✓		指令运行时间量测开始
1013	IETE	—	✓		指令运行时间量测结束
1014	—	DHSCY	—		循环式计数搭配比较表格输出
1015	PPDT	—	—		两相输入之相位差时间检测
1016	ATMR	—	—		接点型 16 位定时器（100ms）
1017	ATMRH	—	—		接点型 16 位定时器（1ms）
1018	ATMRM	—	—		接点型 16 位定时器（10ms）

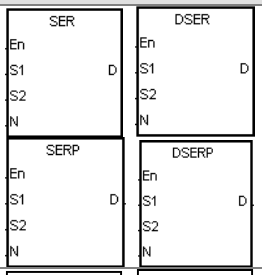
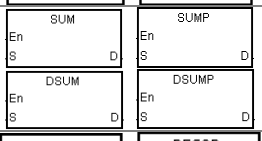
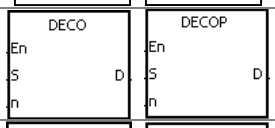
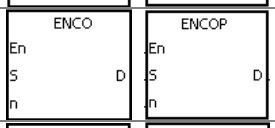
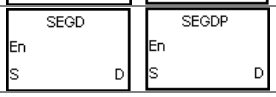
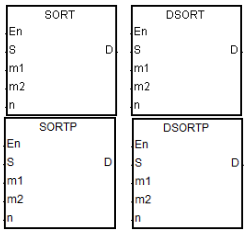
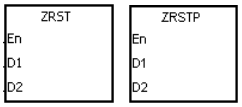
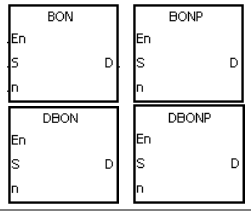
- 移位指令

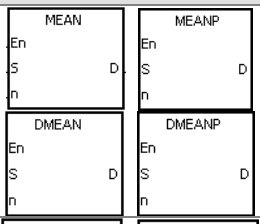
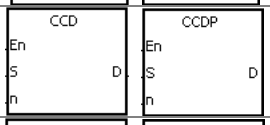
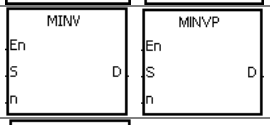
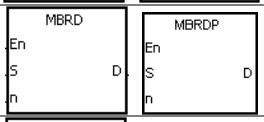
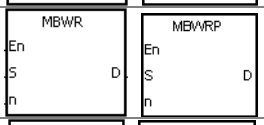
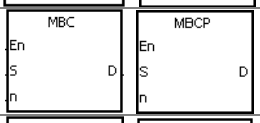
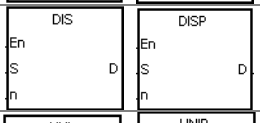
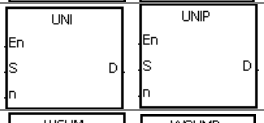
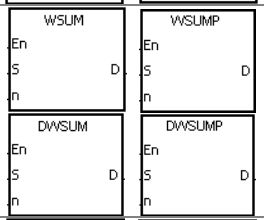
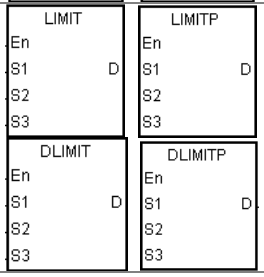
API	指令码		P 指令	符号	功能
	16 位	32 位			
1100	SFTR	—	✓		位右移
1101	SFTL	—	✓		位左移
1102	WSFR	DWSFR	✓		寄存器右移
1103	WSFL	DWSFL	✓		寄存器左移

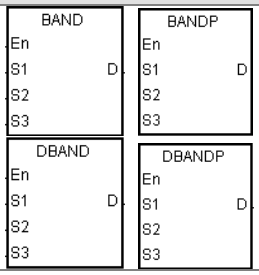
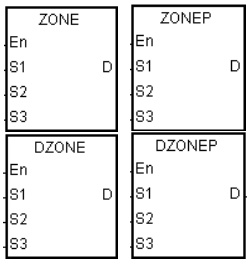
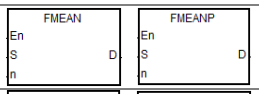
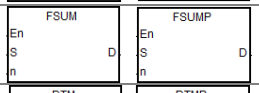
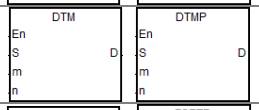
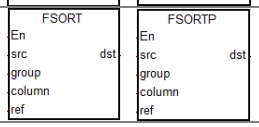
API	指令码		P 指令	符号	功能
	16 位	32 位			
1104	SFWR	DSFWR	✓		位移写入
1105	SFRD	DSFRD	✓		位移读出
1106	SFPO	DSFPO	✓		读出数据串行最新数据
1107	SFDEL	DSFDEL	✓		删除数据串行中的数据
1108	SFINS	DSFINS	✓		插入数据到数据串行中
1109	MBS	—	✓		矩阵位位移
1110	SFR	DSFR	✓		16 位寄存器位右移
1111	SFL	DSFL	✓		16 位寄存器位左移
1112	BSFR	—	✓		n 个位右移 1 个位
1113	BSFL	—	✓		n 个位左移 1 个位
1114	NSFR	DNSFR	✓		n 个寄存器右移

API	指令码		P 指令	符号	功能
	16 位	32 位			
1115	NSFL	DNSFL	✓		n 个寄存器左移

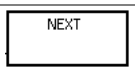
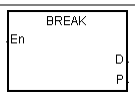
- 数据处理指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
1200	SER	DSER	✓		多点比较
1201	SUM	DSUM	✓		ON 位数量
1202	DECO	—	✓		译码器
1203	ENCO	—	✓		编码器
1204	SEGD	—	✓		7 段显示器解码
1205	SORT	DSORT	✓		数据排序
1206	ZRST	—	✓		区域清除
1207	BON	DBON	✓		ON 位判定

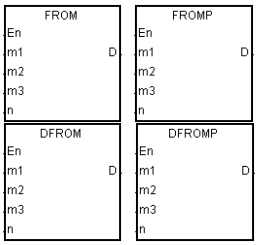
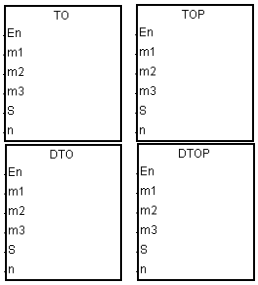
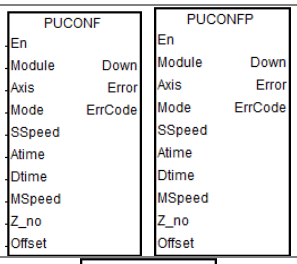
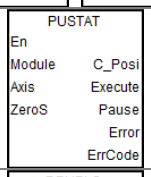
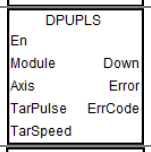
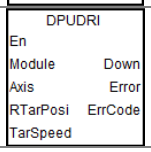
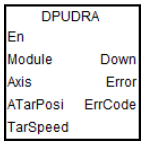
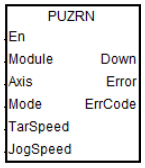
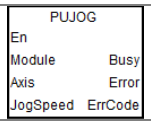
API	指令码		P 指令	符号	功能
	16 位	32 位			
1208	MEAN	DMEAN	✓		平均值
1209	CCD	—	✓		总和检查
1210	ABS	DABS	✓		绝对值
1211	MINV	—	✓		矩阵反相
1212	MBRD	—	✓		矩阵位读出
1213	MBWR	—	✓		矩阵位写入
1214	MBC	—	✓		矩阵位状态计数
1215	DIS	—	✓		16 位数据的 4 位分组
1216	UNI	—	✓		16 位数据的 4 位链结
1217	WSUM	DWSUM	✓		16 位数据的总和计算
1221	LIMIT	DLIMIT	✓		BIN 16 位和 BIN 32 位数据的高低限控制

API	指令码		P 指令	符号	功能
	16 位	32 位			
1222	BAND	DBAND	✓		BIN 16 位和 32 位死区控制
1223	ZONE	DZONE	✓		BIN 16 位和 32 位区域控制
1224	—	FMEAN	✓		浮点数平均值
1225	—	FSUM	✓		浮点数加总
1226	DTM	—	✓		数据转换与搬移
1228	ZSET	—	✓		区域设定
1229	—	FSORT	✓		浮点数数据排序

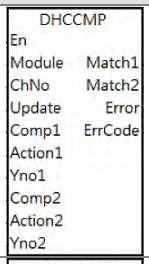
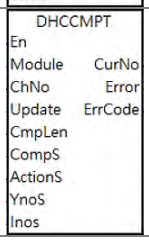
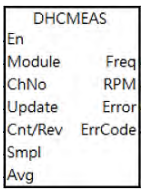
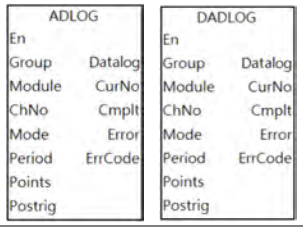
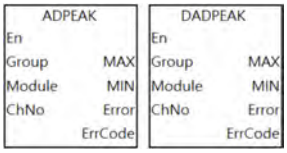
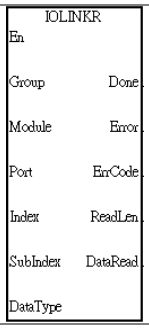
- 结构建立指令

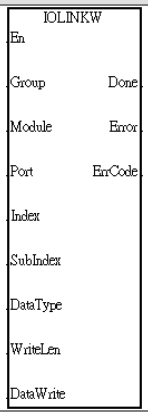
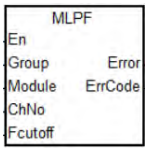
API	指令码		P 指令	符号	功能
	16 位	32 位			
1300	FOR	—	—		循环回路起始
1301	NEXT	—	—		循环回路结束
1302	BREAK	—	—		强制结束 FOR-NEXT 循环

- 模块的数据读/写指令

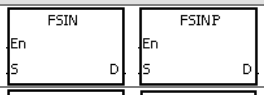
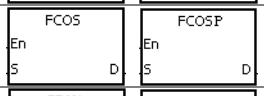
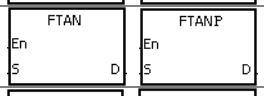
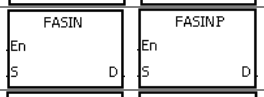
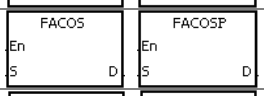
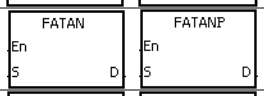
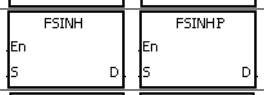
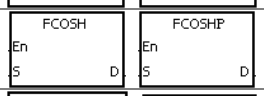
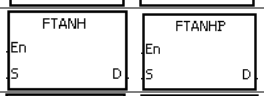
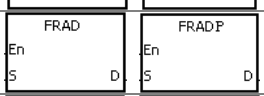
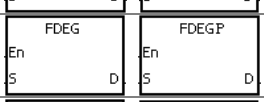
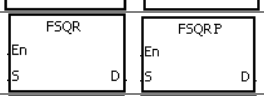
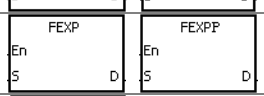
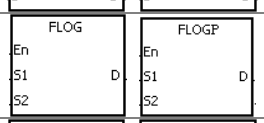
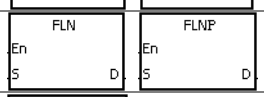
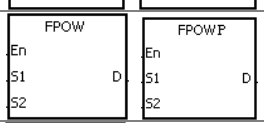
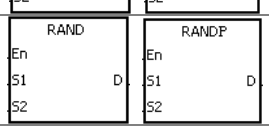
API	指令码		P 指令	符号	功能
	16 位	32 位			
1400	FROM	DFROM	✓		特殊模块 CR 数据读出
1401	TO	DTO	✓		特殊模块 CR 数据写入
1402	—	DPUCONF	✓		PU 模块输出控制参数设定
1403	PUSTAT	—	—		PU 模块输出状态读回
1404	—	DPUPLS	—		PU 模块脉冲输出（无加减速）
1405	—	DPUDRI	—		PU 模块相对定位输出（附加减速）
1406	—	DPUDRA	—		PU 模块绝对寻址输出（附加减速）
1407	—	DPUZRN	—		PU 模块原点回归
1408	—	DPUJOG	—		PU 模块寸动输出

API	指令码		P 指令	符号	功能
	16 位	32 位			
1409	—	DPUMPG	—	DPUMPG En Module OPulse Axis OSpeed InMode Error InPulse ErrCode InSpeed Rate	PU 模块手摇轮控制输出
1410	—	DPUCNT	—	DPUCNT En Module InPulse InMode InSpeed Period Error ZeroS ErrCode	PU 模块高速计数器功能
1415	—	DLCCAL	—	DLCCAL En Group CPoint Module Done ChNo Adone Trigger Error TPoint ErrCode TWeight	LC 模块通道校正
1416	—	DLCWEI	—	LCWEI En Group Weight Module Status ChNo Error Stable ErrCode ZeroS TareS TareW	LC 模块重量值读取
1417	—	DMPID	—	DMPID	温度模块 PID 控制
1418	—	DHCCNT	—	DHCCNT En Module CurCnt ChNo ST Update MT Action AStat Value RefCnt Dir CntStat Error ErrCode	HC 模块计数器控制
1419	—	DHCCAP	—	DHCCAP En Module Capt1 ChNo CmpIt1 TrgSel Capt2 CmpIt2 Error ErrCode	HC 模块计数值获取
1420	HCDO	—	—	HCDO En Module Dostate Update Error Dodata ErrCode	HC 模块输出点控制

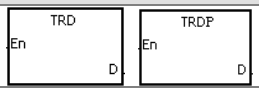
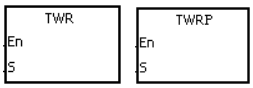
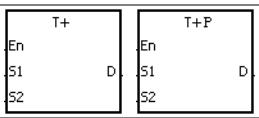
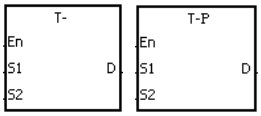
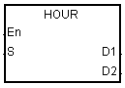
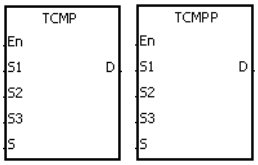
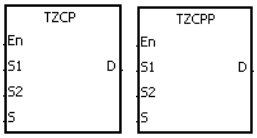
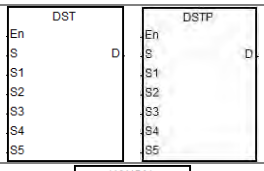
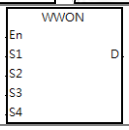
API	指令码		P 指令	符号	功能
	16 位	32 位			
1421	—	DHCCMP	—		HC 模块比较输出
1422	—	DHCCMPT	—		HC 模块表格比较输出
1423	—	DHCMEAS	—		HC 模块频率及转速量测
1424	ADLOG	DADLOG	–		模拟输入模块数据记录
1425	ADPEAK	DADPEAK	–		模拟输入模块峰值记录
1426	IOLINKR	–	–		IO-Link 装置参数读出

API	指令码		P 指令	符号	功能
	16 位	32 位			
1427	IOLINKW	–	–		IO-Link 装置参数写入
1430	MLPF	–	–		模拟输入低通滤波器

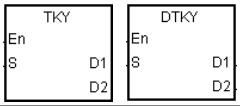
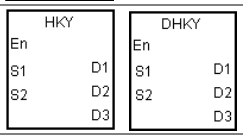
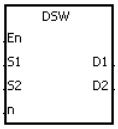
- 浮点数指令

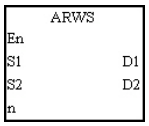
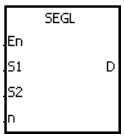
API	指令码		P 指令	符号	功能
	16 位	32 位			
1500	—	FSIN	✓		浮点数 SIN 运算
1501	—	FCOS	✓		浮点数 COS 运算
1502	—	FTAN	✓		浮点数 TAN 运算
1503	—	FASIN	✓		浮点数 ASIN 运算
1504	—	FACOS	✓		浮点数 ACOS 运算
1505	—	FATAN	✓		浮点数 ATAN 运算
1506	—	FSINH	✓		浮点数 SINH 运算
1507	—	FCOSH	✓		浮点数 COSH 运算
1508	—	FTANH	✓		浮点数 TANH 运算
1509	—	FRAD	✓		角度→弧度
1510	—	FDEG	✓		弧度→角度
1511	SQR	DSQR	✓		BIN 开平方根
1512	—	FSQR	✓		浮点数开平方根
1513	—	FEXP	✓		浮点数取指数
1514	—	FLOG	✓		浮点数取对数
1515	—	FLN	✓		浮点数取自然对数
1516	—	FPOW	✓		浮点数权值指令
1517	RAND	—	✓		随机数值

- 万年历指令

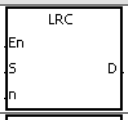
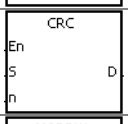
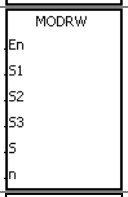
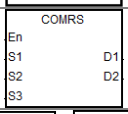
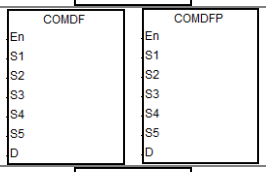
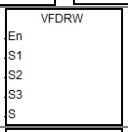
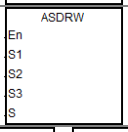
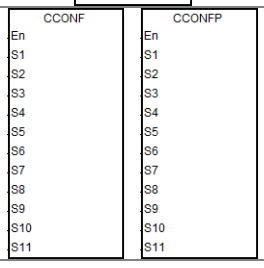
API	指令码		P 指令	符号	功能
	16 位	32 位			
1600	TRD	—	✓		万年历数据读出
1601	TWR	—	✓		万年历数据写入
1602	T+	—	✓		万年历数据加法运算
1603	T-	—	✓		万年历数据减法运算
1604	HOUR	—	—		运转定时器
1605	TCMP	—	✓		万年历数据比较
1606	TZCP	—	✓		万年历数据区域比较
1607	DST	—	✓		日光节约时间设定
1608	WWON	—	—		每周工作时间控制开关

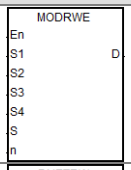
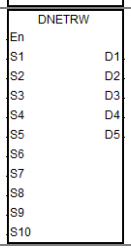
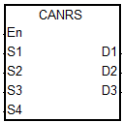
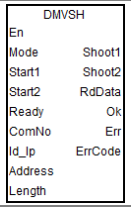
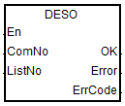
- 外围设备指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
1700	TKY	DTKY	—		10 键键盘输入
1701	HKY	DHKY	—		16 键键盘输入
1702	DSW	—	—		指拨开关输入

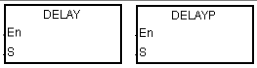
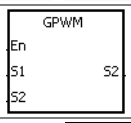
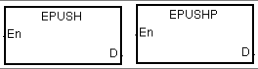
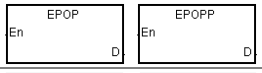
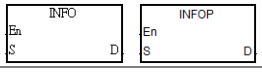
API	指令码		P 指令	符号	功能
	16 位	32 位			
1703	ARWS	—	—		箭头键盘输入
1704	SEGL	—	—		七段显示器扫描输出

- 通讯指令

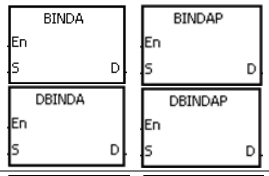
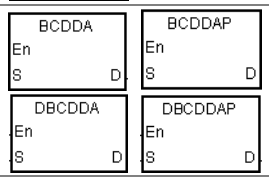
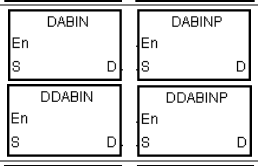
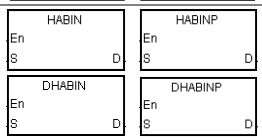
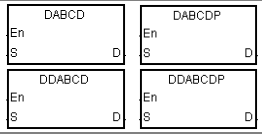
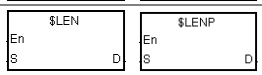
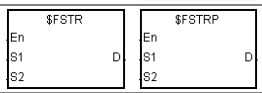
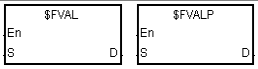
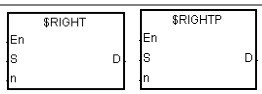
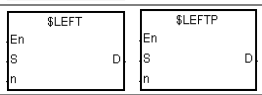
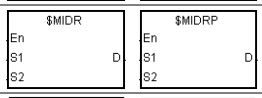
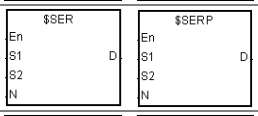
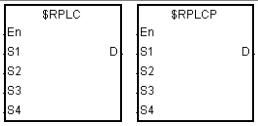
API	指令码		P 指令	符号	功能
	16 位	32 位			
1806	LRC	—	—		和检查 LRC 模式
1807	CRC	—	—		和检查 CRC 模式
1808	MODRW	—	—		MODBUS 数据读写
1812	COMRS	—	—		自定义串行通讯传送与接收指令
1813	COMDF	—	✓		串行通讯端口通讯格式设定指令
1814	VFDRW	—	—		台达变频器专属串行通讯指令
1815	ASDRW	—	—		台达伺服专属串行通讯指令
1816	CCONF	—	✓		通讯端口数据交换表参数设定

API	指令码		P 指令	符号	功能
	16 位	32 位			
1817	MODRWE	—	—		MODBUS 直接指定通讯端口通讯指令
1818	DNETRW	—	—		DeviceNet 通讯读写指令
1819	CANRS	—	—		CAN 自定义通讯传送与接收指令
1820	DMVSH	—	—		台达 DMV 启动检验与通讯指令
1821	DESO	—	—		指定数据交换表通讯编号发送一次

- 其它指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
1900	WDT	—	✓		逾时监视定时器
1901	DELAY	—	✓		延时指令
1902	GPWM	—	—		一般用脉冲宽度调变
1904	EPUSH	—	✓		变址寄存器存入
1905	EPOP	—	✓		变址寄存器读出
1906	INFO	—	✓		系统信息读出

- 字符串处理指令

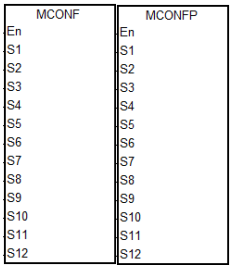
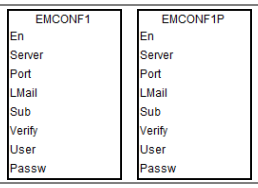
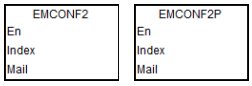
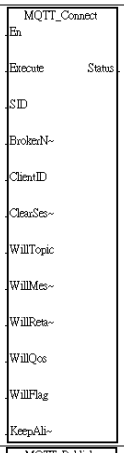
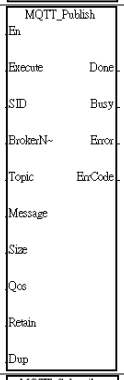
API	指令码		P 指令	符号	功能
	16 位	32 位			
2100	BINDA	DBINDA	✓		有号数十进制→ASCII 变换
2101	BINHA	DBINHA	✓		BIN 十六进制→十六进制 ASCII 变换
2102	BCDDA	DBCDDA	✓		BCD→ASCII 变换
2103	DABIN	DDABIN	✓		有号数十进制 ASCII→有号数十进制 BIN 变换
2104	HABIN	DHABIN	✓		十六进制 ASCII→十六进制 BIN 变换
2105	DABCD	DDABCD	✓		BCD ASCII→BCD 变换
2106	$LEN	—	✓		计算字符串长度
2109	$FSTR	—	✓		Float→String
2110	$FVAL	—	✓		String→Float
2111	$RIGHT	—	✓		从右边获取字符串
2112	$LEFT	—	✓		从左边获取字符串
2113	$MIDR	—	✓		区段获取字符串
2115	$SER	—	✓		字符串搜寻
2116	$RPLC	—	✓		字符串取代

API	指令码		P 指令	符号	功能
	16 位	32 位			
2117	$DEL	—	✓	$DEL: En, S1, S2, S3, D; $DELP: En, S1, S2, S3, D	指定字符串删除
2118	$CLR	—	✓	$CLR: En, S; $CLRP: En, S	字符串清除
2119	$INS	—	✓	$INS: En, S1, S2, S3, D; $INSP: En, S1, S2, S3, D	字符串插入
2122	SPLIT	—	✓	SPLIT: En, S1, S2, S3, S4, D1, D2; SPLITP: En, S1, S2, S3, S4, D1, D2	字符串分离
2123	MERGE	—	✓	MERGE: En, S1, S2, S3, D1, D2; MERGEP: En, S1, S2, S3, D1, D2	字符串合并

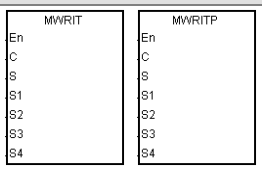
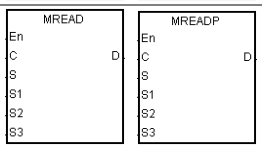
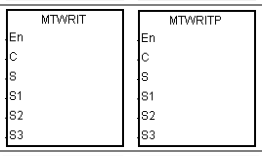
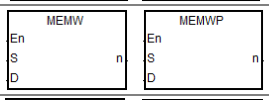
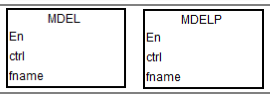
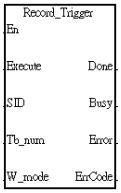
3

- 以太网控制指令

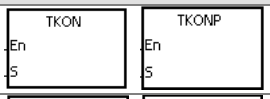
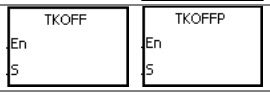
API	指令码		P 指令	符号	功能
	16 位	32 位			
2200	SOPEN	—	✓	SOPEN: En, S1, S2, S3; SOPENP: En, S1, S2, S3	开启 Socket
2201	SSEND	—	✓	SSEND: En, S1, S2; SSENDP: En, S1, S2	通过已开启的 Socket 传送数据
2203	SCLOSE	—	✓	SCLOSE: En, S1, S2; SCLOSEP: En, S1, S2	关闭 Socket
2204	MSEND	—	✓	MSEND: En, S1, S2, S3, D; MSENDP: En, S1, S2, S3, D	寄送电子邮件
2206	INTOA	—	✓	INTOA: En, S, D; INTOAP: En, S, D	IP→字符串变换
2207	IATON	—	✓	IATON: En, S, D; IATONP: En, S, D	字符串→IP 变换
2208	EIPRW	—	—	EIPRW: En, S1, S2, S3, S4, S5, S6, S7, n, S, D1, D2	EtherNet/IP 数据读写
2209	SCONF	—	✓	SCONF: En, S1, S2, S3, S4, S5, S6, S7, S8, S9, S10; SCONFP: En, S1, S2, S3, S4, S5, S6, S7, S8, S9, S10	Socket 参数设定

API	指令码		P 指令	符号	功能
	16 位	32 位			
2210	MCONF	–	✓		MODBUS TCP 数据交换设定
2211	EMCONF1	–	□		E-mail server 参数设定
2212	EMCONF2	–	□		E-mail 地址设定
2214	–	MQTT_ Connect	–		建立 MQTT 联机
2215	MQTT_ Publish	–	–		发布 MQTT
2216	MQTT_ Subscribe	–	–		订阅 MQTT

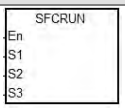
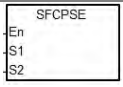
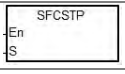
- 储存卡/文件寄存器读写指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
2300	MWRIT	–	✓		写入储存卡数据
2301	MREAD	–	✓		读取储存卡数据
2302	MTWRIT	–	✓		写入字符串至储存卡
2303	MEMW	–	✓		文件寄存器 FR 写入
2304	MDEL	–	✓		储存卡文件删除
2305	Record_ trigger	–	–		记录触发

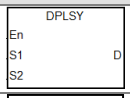
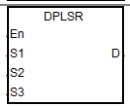
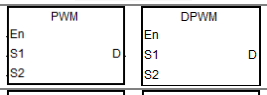
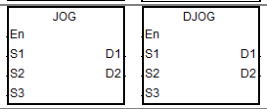
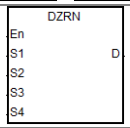
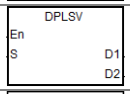
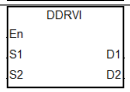
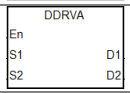
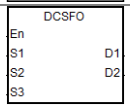
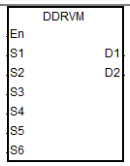
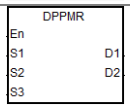
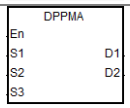
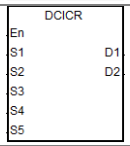
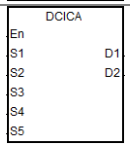
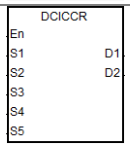
- 任务控制指令

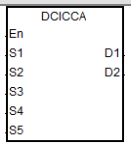
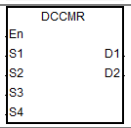
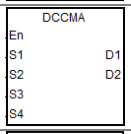
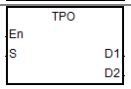
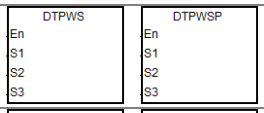
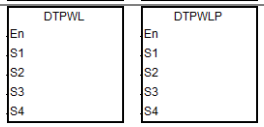
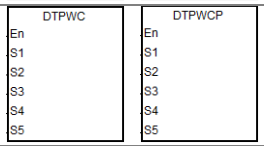
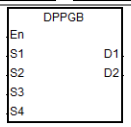
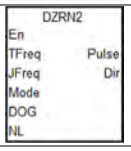
API	指令码		P 指令	符号	功能
	16 位	32 位			
2400	TKON	–	✓		周期性任务（Cyclic Task）启动
2401	TKOFF	–	✓		周期性任务（Cyclic Task）关闭

- SFC 控制指令

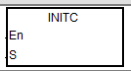
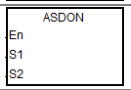
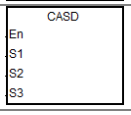
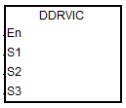
API	指令码		P 指令	符号	功能
	16 位	32 位			
2500	SFCRUN	–	–		SFC 启动
2501	SFCPSE	–	–		SFC 暂停
2502	SFCSTP	–	–		SFC 停止

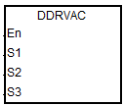
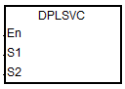
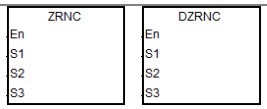
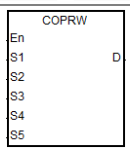
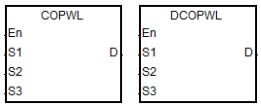
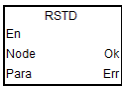
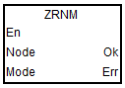
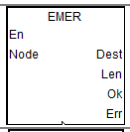
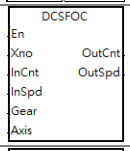
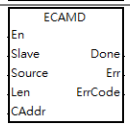
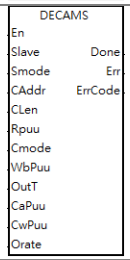
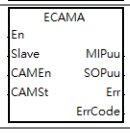
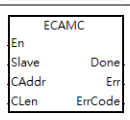
- 高速输出指令

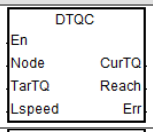
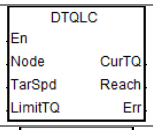
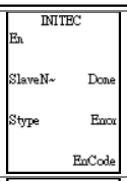
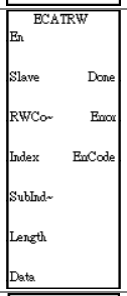
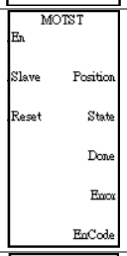
API	指令码		P 指令	符号	功能
	16 位	32 位			
2700	–	DPLSY	–		高速脉冲输出（无加减速）
2701	–	DPLSR	–		高速脉冲输出（附加减速）
2702	PWM	DPWM	–		脉宽调变输出
2703	JOG	DJOG	–		寸动输出
2704	–	DZRN	–		原点回归
2705	–	DPLSV	–		可调变脉冲输出
2706	–	DDRVI	–		相对地址点对点定位输出
2707	–	DDRVA	–		绝对地址点对点定位输出
2708	CSFO	–	–		速度追随输出
2709	–	DDRVM	–		对标型定位输出
2710	–	DPPMR	–		相对地址点对点双轴同动输出
2711	–	DPPMA	–		绝对地址点对点双轴同动输出
2712	–	DCICR	–		相对位置顺时针圆弧插补
2713	–	DCICA	–		绝对位置顺时针圆弧插补
2714	–	DCICCR	–		相对位置逆时针圆弧插补

API	指令码		P 指令	符号	功能
	16 位	32 位			
2715	–	DCICCA	–		绝对位置逆时针圆弧插补
2716	–	DCCMR	–		相对位置画圆
2717	–	DCCMA	–		绝对位置画圆
2718	TPO	–	–		定位规划表控制输出
2719	–	DTPWS	✓		定位规划表的单轴输出参数写入
2720	–	DTPWL	✓		定位规划表的直线插补参数写入
2721	–	DPTWC	✓		定位规划表的圆弧插补参数写入
2723	–	DPPGB	–		点对点来回定位控制
2724	–	DZRN2	–		原点回归 2 (可指定方向输出点)

- 台达专用 CANopen 与 EtherCAT 通讯指令

API	指令码		P 指令	符号	功能
	16 位	32 位			
2800	INITC	–	–		CANopen 专用通讯初始化
2801	ASDON	–	–		伺服启动与停止
2802	CASD	–	–		伺服加减速设定
2803	–	DDRVIC	–		伺服相对位置定位控制

API	指令码		P 指令	符号	功能
	16 位	32 位			
2804	–	DDRVAC	–		伺服绝对位置定位控制
2805	–	DPLSVC	–		伺服速度控制
2806	ZRNC	DZRNC	–		伺服原点回归
2807	COPRW	–	–		CANopen 通讯读写数据
2808	COPWL	DCOPWL	–		CANopen 多笔参数写入
2809	RSTD	—	—		发送重置或 NMT 命令
2810	ZRNM	—	—		伺服原点模式设定
2811	EMER	—	—		读取 Emergency 信息
2812	–	DCSFOC	–		通讯型控制伺服追随功能
2813	ECAMD	–	–		E-CAM 表下载至从站
2814	–	DECAMS	–		E-CAM 参数设定与规划
2815	ECAMA	–	–		E-CAM 启动与状态显示
2816	ECAMC	–	–		变更 E-CAM 表起始位置与长度

API	指令码		P 指令	符号	功能
	16 位	32 位			
2817	–	DTQC	–		扭矩控制
2818	–	DTQLC	–		扭矩限制之速度控制
2820	INITEC	—	—		专用 EtherCAT 通讯初始化
2821	ECATRW	—	—		EtherCAT SDO 数据读写
2822	MOTST	—	—		EtherCAT 从站 Motion 状态读取
2823	ECATST	—	—		EtherCAT 从站通讯状态读取

3.4.2 应用指令（依英文字母排序）

分类	API	指令码（位）		P 指令	功能
		16	32		
符号	0114	$+	–	✓	字符串链结
	2118	$CLR	–	✓	字符串清除
	2117	$DEL	–	✓	指定字符串删除
	2109	$FSTR	–	✓	Float→String
	2110	$FVAL	–	✓	String→Float
	2119	$INS	–	✓	字符串插入
	2112	$LEFT	–	✓	从左边截取字符串
	2106	$LEN	–	✓	计算字符串长度
	2113	$MIDR	–	✓	区段截取字符串
	0302	$MOV	–	✓	字符串移动
	2111	$RIGHT	–	✓	从右边截取字符串
	2116	$RPLC	–	✓	字符串取代
	2115	$SER	–	✓	字符串搜寻
	0102	*	D*	✓	BIN 乘法
	0103	/	D/	✓	BIN 除法
	0100	+	D+	✓	BIN 加法
A	1210	ABS	DABS	✓	绝对值
	0705	ABSD	DABSD	–	绝对方式凸轮控制
	1424	ADLOG	DADLOG	–	模拟输入模块数据记录
	1425	ADPEAK	DADPEAK	–	模拟输入模块峰值记录
	0700	ALT	–	✓	ON/OFF 交替
	0043	AND$<>	–	–	S1≠S2
	0042	AND$=	–	–	S1 = S2
	0812	AND&	DAND&	–	S_1&S_2
	0814	AND^	DAND^	–	S_1^S_2
	0813	AND|	DAND|	–	S_1|S_2
	0010	AND<	DAND<	–	S1 < S2
	0011	AND<=	DAND<=	–	S1≦S2
	0007	AND<>	DAND<>	–	S1≠S2
	0006	AND=	DAND=	–	S1 = S2
	0008	AND>	DAND>	–	S1 > S2
	0009	AND>=	DAND>=	–	S1≧S2
	0076	ANDZ<	DANDZ<	–	|S1-S2| < |S3|

分类	API	指令码（位）		P 指令	功能
		16	32		
	0077	ANDZ<=	DANDZ<=	–	|S1-S2|≦|S3|
	0073	ANDZ<>	DANDZ<>	–	|S1-S2|≠|S3|
	0072	ANDZ=	DANDZ=	–	|S1-S2|=|S3|
	0074	ANDZ>	DANDZ>	–	|S1-S2| > |S3|
	0075	ANDZ>=	DANDZ>=	–	|S1-S2|≧|S3|
	1703	ARWS	–	–	箭头键盘输入
A	2801	ASDON	–	–	伺服启动与停止
	1815	ASDRW	–	–	台达伺服通讯指令
	1016	ATMR	–	–	接点型 16 位定时器（100ms）
	1017	ATMRH	–	–	接点型 16 位定时器（1ms）
	1018	ATMRM	–	–	接点型 16 位定时器（10ms）
	1222	BAND	DBAND	✓	死区控制
	0200	BCD	DBCD	✓	BIN→BCD 变换
	2102	BCDDA	DBCDDA	✓	BCD→ASCII 变换
	0201	BIN	DBIN	✓	BCD→BIN 变换
	2100	BINDA	DBINDA	✓	有号数十进制→ASCII 变换
	2101	BINHA	DBINHA	✓	BIN 十六进制→十六进制 ASCII 变换
	0113	BK-	DBK-	✓	连续区块 BIN 减法
	0112	BK+	DBK+	✓	连续区块 BIN 加法
B	0214	BKBCD	–	✓	连续区块 BIN→BCD 变换
	0215	BKBIN	–	✓	连续区块 BCD→BIN 变换
	0304	BMOV	DBMOV	✓	全部传送
	1207	BON	DBON	✓	ON 位判定
	1302	BREAK	–	–	强制结束 FOR-NEXT 循环
	1113	BSFL	–	✓	n 个位左移 1 个位
	1112	BSFR	–	✓	n 个位右移 1 个位
	0307	BXCH	–	✓	全部交换
	1819	CANRS	–	–	CAN 自定义通讯传送与接收指令
	2802	CASD	–	–	伺服加减速设定
	1209	CCD	–	✓	总和检查
C	1816	CCONF	–	✓	通讯端口数据交换表参数设定
	0065	CHKADR	–	–	接点类型指针寄存器地址检查
	0400	CJ	–	✓	条件跳转
	0303	CML	DCML	✓	反转传送
	0054	CMP	DCMP	✓	比较设定输出

分类	API	指令码（位）		P 指令	功能
		16	32		
	0063	CMPT<	–	✓	表格比较，<
	0064	CMPT<=	–	✓	表格比较，<=
	0060	CMPT<>	–	✓	表格比较，<>
	0059	CMPT=	–	✓	表格比较，=
	0061	CMPT>	–	✓	表格比较，>
C	0062	CMPT>=	–	✓	表格比较，>=
	1003	CNT	–	–	16 位计数器
	1813	COMDF	–	✓	串行通讯端口格式设定
	1812	COMRS	–	–	自定义通讯传送与接收指令
	2807	COPRW	–	–	CANopen 通讯读写数据
	2808	COPWL	DCOPWL	–	CANopen 多笔参数写入
	1807	CRC	–	–	和检查 CRC 模式
	2708	CSFO	–	–	速度追随输出
D	0101	–	D-	✓	BIN 减法
	1010	–	DCAP	–	外部输入中断获取高速计数值
	2717	–	DCCMA	–	绝对位置画圆
	2716	–	DCCMR	–	相对位置画圆
	2713	–	DCICA	–	绝对位置顺时针圆弧插补
	2715	–	DCICCA	–	绝对位置逆时针圆弧插补
	2714	–	DCICCR	–	相对位置逆时针圆弧插补
	2712	–	DCICR	–	相对位置顺时针圆弧插补
	1004	–	DCNT	–	32 位计数器（包含高速计数器使用说明）
	2812	–	DCSFOC	–	通讯型控制伺服追随功能
	2707	–	DDRVA	–	绝对地址点对点定位输出
	2804	–	DDRVAC	–	伺服绝对位置定位控制
	2706	–	DDRVI	–	相对地址点对点定位输出
	2803	–	DDRVIC	–	伺服相对位置定位控制
	2709	–	DDRVM	–	对标型定位输出
	2814	–	DECAMS	–	E-CAM 参数设定与规划
	1419	–	DHCCAP	–	HC 模块计数值获取
	1421	–	DHCCMP	–	HC 模块比较输出
	1422	–	DHCCMPT	–	HC 模块表格比较输出
	1418	–	DHCCNT	–	HC 模块计数器控制
	1423	–	DHCMEAS	–	HC 模块频率及转速量测

分类	API	指令码（位）		P 指令	功能
		16	32		
	1006	–	DHSCR	–	高速比较清除
	1005	–	DHSCS	–	高速比较设定
	1014	–	DHSCY	–	循环式计数搭配比较表格输出
	0601	–	DHSRF	✓	高速比较值立即刷新
	1007	–	DHSZ	–	高速区域比较
	1415	–	DLCCAL	–	LC 模块通道校正
	1417	–	DMPID	–	温度模块 PID 控制
	0708	–	DPIDE	–	PID 运算
	2701	–	DPLSR	–	高速脉冲输出（附加减速）
	2705	–	DPLSV	–	可调变脉冲输出
	2805	–	DPLSVC	–	伺服速度控制
	2700	–	DPLSY	–	高速脉冲输出（无加减速）
	2723	–	DPPGB	–	点对点来回定位控制
	2711	–	DPPMA	–	绝对地址点对点双轴同动输出
	2710	–	DPPMR	–	相对地址点对点双轴同动输出
	2721	–	DPTWC	✓	定位规划表之圆弧插补参数写入
	1410	–	DPUCNT	–	PU 模块高速计数器功能
	1402	–	DPUCONF	✓	PU 模块输出控制参数设定
	1406	–	DPUDRA	–	PU 模块绝对寻址输出（附加减速）
	1405	–	DPUDRI	–	PU 模块相对定位输出（附加减速）
	1408	–	DPUJOG	–	PU 模块寸动输出
	1409	–	DPUMPG	–	PU 模块手摇轮控制输出
	1404	–	DPUPLS	–	PU 模块脉冲输出（无加减速）
	1407	–	DPUZRN	–	PU 模块原点回归
	1008	–	DSPD	–	速度检测指令
	0711	–	DSUNRS	✓	日出与日落时间
	2720	–	DTPWL	✓	定位规划表之直线插补参数写入
	2719	–	DTPWS	✓	定位规划表之单轴输出参数写入
	2817	–	DTQC	–	扭矩控制
	2818	–	DTQLC	–	扭矩限制之速度控制
	2704	–	DZRN	–	原点回归
	2724	–	DZRN2	–	原点回归 2（可指定方向输出点）
	2105	DABCD	DDABCD	✓	BCD ASCII→BCD 变换

分类	API	指令码（位）		P 指令	功能
		16	32		
	2103	DABIN	DDABIN	✓	有号数十进制 ASCII→有号数十进制 BIN 变换
	0116	DEC	DDEC	✓	BIN 减一
	1202	DECO	–	✓	译码器
	1901	DELAY	–	✓	延时指令
	1821	DESO	–	–	指定数据交换表通讯编号发送一次
	0500	DI	–	–	全部中断插入关闭
	1215	DIS	–	✓	16 位数据的 4 位分组
	0118	DIV16	DIV32	✓	DIV16：16 位专用 BIN 除法 DIV32：32 位专用 BIN 除法
	0504	DIX	–	–	指定中断关闭
	1820	DMVSH	–	–	台达 DMV 启动检验与通讯指令
	1818	DNETRW	–	–	DeviceNet 通讯读写指令
	1607	DST	–	✓	日光节约时间设定
	1702	DSW	–	–	指拨开关输入
	1226	DTM	–	✓	数据转换与搬移
E	2815	ECAMA	–	–	E-CAM 启动与状态显示
	2816	ECAMC	–	–	变更 E-CAM 表起始位置与长度
	2813	ECAMD	–	–	E-CAM 表下载至从站
	2821	ECATRW	–	–	EtherCAT SDO 数据读写
	2823	ECATST	–	–	EtherCAT 从站通讯状态读取
	0501	EI	–	–	全部中断插入启动
	2208	EIPRW	–	–	EtherNet/IP 数据读写
	0503	EIX	–	–	指定中断启动
	2211	EMCONF1	–	✓	E-mail server 参数设定
	2212	EMCONF2	–	✓	E-mail 地址设定
	1203	ENCO	–	✓	编码器
	1905	EPOP	–	✓	变址寄存器读出
	1904	EPUSH	–	✓	变址寄存器存入
F	0105	–	F-	✓	浮点数减法
	0106	–	F*	✓	浮点数乘法
	0107	–	F/	✓	浮点数除法
	0104	–	F+	✓	浮点数加法
	1504	–	FACOS	✓	浮点数 ACOS 运算

分类	API	指令码（位）		P 指令	功能
		16	32		
	0028	–	FAND<	–	S1 < S2
	0029	–	FAND<=	–	S1≦S2
	0025	–	FAND<>	–	S1≠S2
	0024	–	FAND=	–	S1 = S2
	0026	–	FAND>	–	S1 > S2
	0027	–	FAND>=	–	S1≧S2
	1503	–	FASIN	✓	浮点数 ASIN 运算
	1505	–	FATAN	✓	浮点数 ATAN 运算
	0212	–	FBCD	✓	单精度浮点数→十进浮点数
	0213	–	FBIN	✓	十进浮点数→单精度浮点数
	0056	–	FCMP	✓	浮点数比较
	1501	–	FCOS	✓	浮点数 COS 运算
	1507	–	FCOSH	✓	浮点数 COSH 运算
	1510	–	FDEG	✓	弧度→角度
	1513	–	FEXP	✓	浮点数取指数
	0022	–	FLD<	–	S1 < S2
	0023	–	FLD<=	–	S1≦S2
	0019	–	FLD<>	–	S1≠S2
	0018	–	FLD=	–	S1 = S2
	0020	–	FLD>	–	S1 > S2
	0021	–	FLD>=	–	S1≧S2
	1515	–	FLN	✓	浮点数取自然对数
	1514	–	FLOG	✓	浮点数取对数
	1224	–	FMEAN	✓	浮点数平均值
	0211	–	FNEG	✓	单精度浮点数正负符号反相
	0034	–	FOR<	–	S1 < S2
	0035	–	FOR<=	–	S1≦S2
	0031	–	FOR<>	–	S1≠S2
	0030	–	FOR=	–	S1 = S2
	0032	–	FOR>	–	S1 > S2
	0033	–	FOR>=	–	S1≧S2
	1516	–	FPOW	✓	浮点数权值指令
	1509	–	FRAD	✓	角度→弧度
	1500	–	FSIN	✓	浮点数 SIN 运算
	1506	–	FSINH	✓	浮点数 SINH 运算

分类	API	指令码（位）		P 指令	功能
		16	32		
	1229	–	FSORT	✓	浮点数数据排序
	1512	–	FSQR	✓	浮点数开平方根
	1225	–	FSUM	✓	浮点数加总
	1502	–	FTAN	✓	浮点数 TAN 运算
	1508	–	FTANH	✓	浮点数 TANH 运算
	0057	–	FZCP	✓	浮点数区域比较
	0202	FLT	DFLT	✓	BIN 整数→单精度浮点数变换
	1300	FOR	–	–	循环回路起始
	1400	FROM	DFROM	✓	扩展模块 CR 数据读出
G	0209	GBIN	DGBIN	✓	GRY 码→BIN 变换
	0402	GOEND	–	–	跳转到 END
	1902	GPWM	–	–	一般用脉冲波宽调变
	0208	GRY	DGRY	✓	BIN→GRY 码变换
H	2104	HABIN	DHABIN	✓	十六进制 ASCII→十六进制 BIN 变换
	1420	HCDO	–	–	HC 模块输出点控制
	1701	HKY	DHKY	–	16 键键盘输入
	1604	HOUR	–	–	运转定时器
I	2207	IATON	–	✓	字符串→IP 变换
	1012	IETS	–	✓	指令运行时间量测开始
	1013	IETE	–	✓	指令运行时间量测结束
	0115	INC	DINC	✓	BIN 加一
	0706	INCD	–	–	相对方式凸轮控制
	1906	INFO	–	✓	系统信息读出
	2800	INITC	–	–	CAN 专用通讯初始化
	2820	INITEC	–	–	EtherCAT 专用通讯初始化
	0204	INT	DINT	✓	单精度浮点数→BIN 整数变换
	2206	INTOA	–	✓	IP→字符串变换
	1426	IOLINKR	–	–	IO-Link 装置参数读出
	1427	IOLINKW	–	–	IO-Link 装置参数写入
J	0401	JMP	–	–	无条件跳转
	2703	JOG	DJOG	–	寸动输出
L	1416	LCWEI	–	–	LC 模块重量值读取
	0037	LD$<>	–	–	S1≠S2
	0036	LD$=	–	–	S1 = S2
	0809	LD&	DLD&	–	S_1&S_2

分类	API	指令码（位）		P 指令	功能
		16	32		
	0811	LD^	DLD^	–	S_1^S_2
	0810	LD|	DLD|	–	S_1|S_2
	0004	LD<	DLD<	–	S1 < S2
	0005	LD<=	DLD<=	–	S1≦S2
	0001	LD<>	DLD<>	–	S1≠S2
	0000	LD=	DLD=	–	S1 = S2
	0002	LD>	DLD>	–	S1 > S2
	0003	LD>=	DLD>=	–	S1≧S2
	0070	LDZ<	DLDZ<	–	|S1-S2| < |S3|
	0071	LDZ<=	DLDZ<=	–	|S1-S2|≦|S3|
	0067	LDZ<>	DLDZ<>	–	|S1-S2|≠|S3|
	0066	LDZ=	DLDZ=	–	|S1-S2|=|S3|
	0068	LDZ>	DLDZ>	–	|S1-S2| > |S3|
	0069	LDZ>=	DLDZ>=	–	|S1-S2|≧|S3|
	1221	LIMIT	DLIMIT	✓	高低限控制
	1806	LRC	–	–	和检查 LRC 模式
M	0801	MAND	–	✓	矩阵及（AND）运算
	2304	MDEL	–	✓	储存卡文件删除
	2123	MERGE	–	✓	字符串合并
	1214	MBC	–	✓	矩阵位状态计数
	0904	MBR	–	✓	矩阵位旋转
	1212	MBRD	–	✓	矩阵位读出
	1109	MBS	–	✓	矩阵位位移
	1213	MBWR	–	✓	矩阵位写入
	0058	MCMP	–	✓	矩阵比较
	2210	MCONF	–	✓	MODBUS TCP 数据交换设定
	1208	MEAN	DMEAN	✓	平均值
	2303	MEMW	–	✓	文件寄存器 FR 写入
	1211	MINV	–	✓	矩阵反相
	1430	MLPF	–	–	模拟输入低通滤波器
	0206	MMOV	–	✓	16→32 位数值转换
	1808	MODRW	–	–	MODBUS 通讯指令
	1817	MODRWE	–	–	MODBUS 直接指定通讯端口通讯指令
	0803	MOR	–	✓	矩阵或（OR）运算
	2822	MOTST	–	–	EtherCAT 从站 Motion 状态读取

分类	API	指令码（位）		P 指令	功能
		16	32		
	0300	MOV	DMOV	✓	数据移动
	0310	MOVB	–	✓	多位移动
	2214	–	MQTT_Connect	–	建立 MQTT 联机
	2215	MQTT_Publish	–	–	发布 MQTT
	2216	MQTT_Subscribe	–	–	订阅 MQTT
	2301	MREAD	–	✓	将数据从储存卡里读取到 PLC
	2204	MSEND	–	✓	寄送电子邮件
	0704	MTR	–	–	矩阵输入
	2302	MTWRIT	–	✓	将文字写入到储存卡里
	0117	MUL16	MUL32	✓	MUL16：16 位专用 BIN 乘法 MUL32：32 位专用 BIN 乘法
	2300	MWRIT	–	✓	将 PLC 数据数据写入到储存卡里
	0805	MXOR	–	✓	矩阵异或（XOR）运算
N	0210	NEG	DNEG	✓	取负数（取 2 的补码）
	1301	NEXT	–	–	循环回路结束
	0305	NMOV	DNMOV	✓	多点移动
	1115	NSFL	DNSFL	✓	n 个寄存器左移
	1114	NSFR	DNSFR	✓	n 个寄存器右移
O	0049	OR$<>	–	–	S1≠S2
	0048	OR$=	–	–	S1 = S2
	0815	OR&	DOR&	–	S_1&S_2
	0817	OR^	DOR^	–	S_1^S_2
	0816	OR|	DOR|	–	S_1|S_2
	0016	OR<	DOR<	–	S1 < S2
	0017	OR<=	DOR<=	–	S1≦S2
	0013	OR<>	DOR<>	–	S1 ≠ S2
	0012	OR=	DOR=	–	S1 = S2
	0014	OR>	DOR>	–	S1 > S2
	0015	OR>=	DOR>=	–	S1≧S2
	0082	ORZ<	DORZ<	–	|S1-S2| < |S3|
	0083	ORZ<=	DORZ<=	–	|S1-S2|≦|S3|
	0079	ORZ<>	DORZ<>	–	|S1-S2|≠|S3|
	0078	ORZ=	DORZ=	–	|S1-S2|=|S3|
	0080	ORZ>	DORZ>	–	|S1-S2| > |S3|
	0081	ORZ>=	DORZ>=	–	|S1-S2|≧|S3|

分类	API	指令码（位）		P 指令	功能
		16	32		
P	1015	PPDT	–	–	两相输入之相位差时间检测
	1403	PUSTAT	–	–	PU 模块输出状态读回
	1009	PWD	–	–	脉宽检测指令
	2702	PWM	DPWM	–	脉宽调变输出
R	0703	RAMP	DRAMP	–	周期型之倾斜信号
	1517	RAND	–	✓	随机数值
R	0903	RCL	DRCL	✓	附进位标志左旋转
	0901	RCR	DRCR	✓	附进位标志右旋转
	2305	Record_trigger	–	–	记录触发
	0600	REF	–	✓	I/O 更新处理
	0602	REFF	–	✓	更新输入点滤波时间
	0207	RMOV	–	✓	32→16 位数值转换
	0902	ROL	DROL	✓	左旋转
	0900	ROR	DROR	✓	右旋转
	1000	RST	DRST	–	接点或寄存器清除
S	0216	SCAL	DSCAL	✓	比例运算
	0222	SCLM	DSCLM	✓	多点式区域比例运算
	2203	SCLOSE	–	✓	关闭 Socket
	0217	SCLP	DSCLP	✓	参数型比例运算
	2209	SCONF	–	✓	Socket 参数设定
	1204	SEGD	–	✓	7 段显示器解码
	1704	SEGL	–	–	七段显示器扫描输出
	1200	SER	DSER	✓	多点比较
	2501	SFCPSE	–	–	SFC 暂停
	2500	SFCRUN	–	–	SFC 启动
	2502	SFCSTP	–	–	SFC 停止
	1107	SFDEL	DSFDEL	✓	删除数据串行中的数据
	1108	SFINS	DSFINS	✓	插入数据到数据串行中
	1111	SFL	DSFL	✓	16 位寄存器位左移
	1106	SFPO	DSFPO	✓	读出数据串行最新数据
	1110	SFR	DSFR	✓	16 位寄存器位右移
	1105	SFRD	DSFRD	✓	位移读出
	1101	SFTL	–	✓	位左移
	1100	SFTR	–	✓	位右移

分类	API	指令码（位）		P 指令	功能
		16	32		
	1104	SFWR	DSFWR	✓	位移写入
	0309	SMOV	–	✓	位数移动
	2122	SPLIT	–	✓	字符串分离
	2200	SOPEN	–	✓	开启 Socket
	1205	SORT	DSORT	✓	数据排序
	1511	SQR	DSQR	✓	BIN 开平方根
	2201	SSEND	–	✓	通过已开启的 Socket 传送数据
	0702	STMR	–	–	特殊定时器
	1201	SUM	DSUM	✓	ON 位数量
	0308	SWAP	DSWAP	✓	上/下 BYTE 变换
T	1603	T-	–	✓	万年历数据减法运算
	1602	T+	–	✓	万年历数据加法运算
	1605	TCMP	–	✓	时间比较
	2401	TKOFF	–	✓	周期性任务（Cyclic Task）关闭
	2400	TKON	–	✓	周期性任务（Cyclic Task）启动
	1700	TKY	DTKY	–	10 键键盘输入
	1001	TMR	–	–	16 位定时器（单位：100ms）
	1002	TMRH	–	–	16 位定时器（单位：1ms）
	1011	TMRM	–	–	16 位定时器（单位：10ms）
	1401	TO	DTO	✓	扩展模块 CR 数据写入
	2718	TPO	–	–	定位规划表控制输出
	1600	TRD	–	✓	万年历数据读出
	0701	TTMR	–	–	教导式定时器
	1601	TWR	–	✓	万年历数据写入
	1606	TZCP	–	✓	万年历数据区域比较
U	1216	UNI	–	✓	16 位数据的 4 位链结
V	1814	VFDRW	–	–	台达变频器通讯指令
W	0800	WAND	DAND	✓	逻辑与（AND）运算
	1608	WWON	–	–	每周工作时间控制开关
	1900	WDT	–	✓	逾时监视定时器
	0808	WINV	DINV	✓	逻辑反向或（INV）运算
	0802	WOR	DOR	✓	逻辑或（OR）运算
	1103	WSFL	DWSFL	✓	寄存器左移
	1102	WSFR	DWSFR	✓	寄存器右移
	1217	WSUM	DWSUM	✓	总和运算

分类	API	指令码（位）		P 指令	功能
		16	32		
	0804	WXOR	DXOR	✓	逻辑异或（XOR）运算
X	0306	XCH	DXCH	✓	数据的交换
	0709	XCMP	–	–	多任务位输入比较设定
Y	0710	YOUT	–	–	多任务位比较与输出
Z	0055	ZCP	DZCP	✓	区域比较
	1223	ZONE	DZONE	✓	区域控制
	2806	ZRNC	DZRNC	–	伺服原点回归
	1206	ZRST	–	✓	区域清除
	1228	ZSET	–	✓	区域设定

4

第4章 指令结构

目录

4.1 API 应用指令组成说明..... 4-2
4.2 操作数使用说明..... 4-5
4.3 指令使用限制..... 4-6
4.4 变址（E）说明..... 4-8
4.5 指针寄存器（PR）说明.....4-10
4.6 定时器指针寄存器（TR）说明 4-12
4.7 16 位计数器指针寄存器（CR）说明 4-13
4.8 32 位计数器指针寄存器（HCR）说明.....4-14
4.9 文件寄存器（FR）说明.....4-16

4.1 API 应用指令组成说明

PLC 指令提供一个特定的指令码及 API 编号，以便储存。下面的表格中指令的 API 编号为 0300，指令码为 MOV，MOV 的功能描述是“数据移动”。

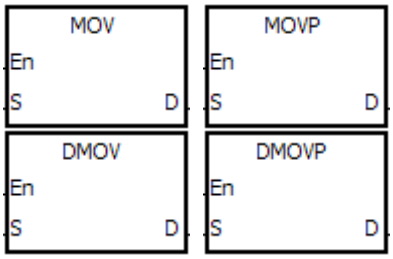
API	指令码			操作数	功能
0300	D	MOV	P	**S**，**D**	数据移动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		○
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●		●		●	●	
D		●	●			●	●		●		●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 数据源

D ： 数据目的地

1. 操作数区域会列出各种指令所用到的装置。S，D，n，m 会根据它的功能使用于操作数，如果不止一个操作数并且功能相同时，会附加编号使用，如 S_1，S_2…。
2. 脉冲执行型指令要求在指令码后面加上“P”，而 32 位指令要求在指令码前面加上“D”，如“D***P”，“***”是指令码。
3. 操作数区域中，32 位单精度浮点数表示为 F。
4. 上表中含●的装置支持 E 修饰，含 ○ 的装置不支持 E 修饰。例如，操作数 **S** 支持 D 装置的 E 修饰。
5. 适用机种注明在表格的右下角。详细的指令变化情况，可以对照着表格下方确认指令是否支持脉冲执行型，16 位指令，32 位指令。
6. 如果要在功能块中使用指令时，操作数区域中有支持装置 X、Y、D，则必须使用 PR 指针寄存器来运算；有支持装置 T、C、HC，则必须使用 TR、CR、HCR 指针寄存器来运算；详细说明请参考第 4.4~4.7 节指针寄存器说明或 ISPSoft 使用手册第 7.2.4 节指针类型的变量符号说明。
7. MOV 指令在 ISPSoft 中梯形图显示之方块符号说明如下：

   MOV，**MOVP**，**DMOV**，**DMOVP**：表示指令码

   En：表示致能

S：数据源（适用操作数格式为 Word/Double Word）

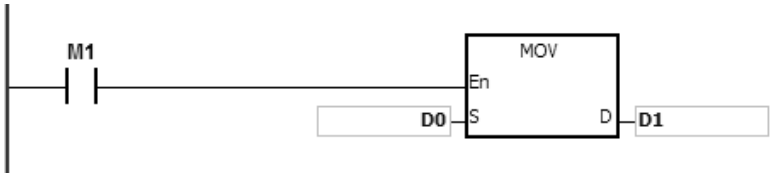
D：数据目的地（适用操作数格式为 Word/Double Word）

指令组成：

应用指令中有些指令仅有指令部份（指令码）构成，例如：EI，DI…或 WDT 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。

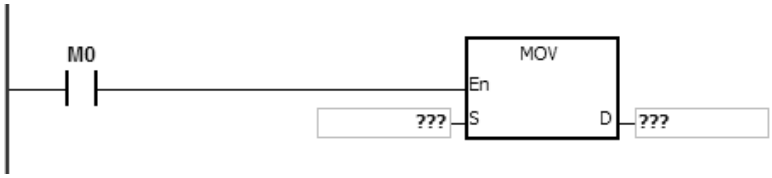
每个应用指令均有其指令编号（API）及专用的名称符号，例如：API0300 的指令码符号为 MOV（数据搬移）。

直接输入法：利用梯形图编辑软件 ISPSoft 作该指令的输入，以 MOV 指令而言只需要直接输入指令的名称与操作数指定“MOV D0 D1”即可：



拖曳输入法：选择 ISPSoft 左侧项目管理区内的函式库，将数据处理分类中“MOV”指令拖曳到梯形图编辑区即可，操作数再另行指定。

工具栏输入法：选择 ISPSoft 工具栏的操作数/应用指令/功能块选取按钮，选取函示库类别，再选择数据处理分类中“MOV”指令插入即可，操作数再另行指定。



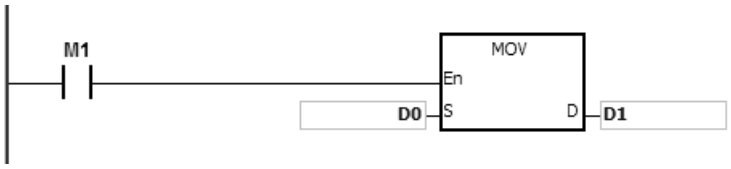
S	来源操作数；若来源操作数有一个以上，那么则以 $\mathbf{S_1}$，$\mathbf{S_2}$…分别表示。
D	目的操作数；若目的操作数有一个以上，那么则以 $\mathbf{D_1}$，$\mathbf{D_2}$…分别表示。
若操作数指定数目、地址时，那么则以 **m**，$\mathbf{m_1}$，$\mathbf{m_2}$，**n**，$\mathbf{n_1}$，$\mathbf{n_2}$ 表示。	

操作数长度（16 位指令或 32 位指令或浮点数指令）：

16 位指令或 32 位指令

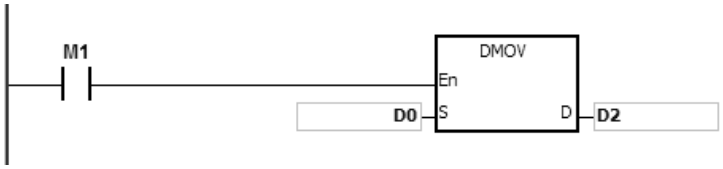
操作数的数值内容，其长度可分为 16 位及 32 位，因此部份指令处理不同长度的数据则分为 16 及 32 位的指令，用以区分 32 位的指令只需要在 16 位指令前加上“D”来表示即可。

16 位 MOV 指令



当 M1=ON，D0 被传送至 D1

32 位 DMOV 指令

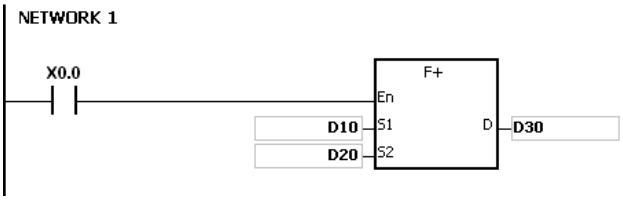


当 M1=ON 时，(D1，D0) 的内容被传送至（D3，D2）

浮点数指令

浮点数指令仅支持 32 位单精度浮点数指令，浮点数指令操作数的数值内容表示方式开头都会以“F”来表示。详细的浮点数说明可以参考第二章的浮点数内容。

32 位单精度浮点数指令 F+

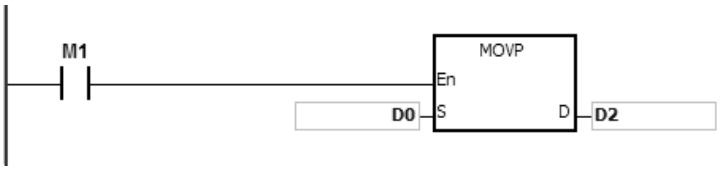


当 X0.0=ON 时，单精度浮点数（D11，D10）+单精度浮点数（D21，D20）的内容被传送至（D31，D30）

连续执行型与脉冲执行型：

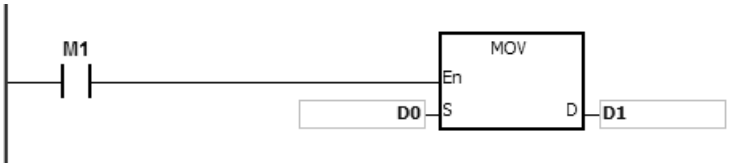
1. 以指令的执行方式来说亦可分成「连续执行型」及「脉冲执行型」2 种。由于指令不被执行时，所需的运行时间比较短，因此程序中尽可能的使用脉冲执行型指令可减少扫描周期。
2. “脉冲”功能可以让相关的指令启动上升沿触发的控制输入。指令在一个扫描周期内被执行 ON。
3. 之后，若控制输入保持为 ON，且关联的指令没有执行，为了重新执行指令，控制输入必须再次从 OFF 到 ON。
4. 脉冲执行型指令。

脉冲执行型



当 M1 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 M1=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

上图的条件接点 M1=OFF 时，指令不被执行，目的地操作数 D 的内容没有变化。

4.2 操作数使用说明

AS 系列之操作数分为「用户可直接指定」跟「编辑软件系统内部使用」两大类，其支持之种类与说明，分别说明如下。

用户可直接**指定之操作数如下：**

1. 输入继电器：X0.0 ~ X63.15 或 X0 ~ X63
2. 输出继电器：Y0.0 ~ Y63.15 或 Y0 ~ Y63
3. 内部继电器：M0 ~ M8191
4. 步进点继电器：S0 ~ S2047
5. 定时器：T0 ~ T511
6. 16 位计数器：C0 ~ C511
7. 32 位计数器：HC0 ~ HC255
8. 数据寄存器：D0 ~ D29999 或 D0.0 ~ D29999.15
9. 文件寄存器：FR0 ~ FR65535
10. 特殊标志：SM0 ~ SM2047
11. 特殊寄存器：SR0 ~ SR2047
12. 变址寄存器：E0 ~ E9
13. 常数：十进制常数 K，十六进制常数 16#
14. 字符串：“$”
15. 浮点数：单精度浮点数 F
16. 寄存器一般为 16 位长度，也就是 1 个寄存器，若指定 32 位长度的数据寄存器时，是指定连续号码的 2 个寄存器。
17. 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，及 16 位计数器 C 被使用的规则亦相同。
18. 32 位计数器 HC 若是当数据寄存器来使用时，只有 32 位指令的操作数可指定。
19. 变址寄存器 E 限制在 16 位指令中使用。

PS.有关装置的介绍请参考第二章装置说明。

编辑软件系统内部使用之操作数如下：（此类别是系统自动配置，用户不需设定）

1. 自动配置变量宣告（含 Bool、WORD、INT …等等）：U0 ~ U16387 及 W0 ~ W29999
2. 工作（TASK）任务启动与关闭：TK0 ~ TK31
3. 指针类型的变量符号：各种指针类型的变量符号、配合的装置名称及使用限制请参考下表。

指针类型	使用限制	
一般指针 POINTER	装置范围	PR0 ~ PR15，PR0.0~PR15.15
	数量限制	每个功能块定义中，最多可宣告 16 个

指针类型	使用限制	
	可传指针	WORD/DWORD/LWORD/INT/DINT/LINT 等类型的变量符号或 D 装置或 X、Y 等字组装置（如 X0、Y0）。
定时器专用指针 T_POINTER	装置范围	TR0 ~ TR7
	数量限制	每个功能块定义中，最多可宣告 8 个
	可传指针	T 装置或 TIMER 类型的变量符号
计数器专用指针 C_POINTER	装置范围	CR0 ~ CR7
	数量限制	每个功能块定义中，最多可宣告 8 个
	可传指针	C 装置或 COUNTER 类型的变量符号
高速计数器专用指针 HC_POINTER	装置范围	HCR0 ~ HCR7
	数量限制	每个功能块定义中，最多可宣告 8 个
	可传指针	HC 装置或指定配置 HC 装置的 COUNTER 类型符号

4.3 指令使用限制

- 只能在功能块（Function Block）中使用的指令

  API0065 CHKADR，FB_NP，FB_PN，NED，ANED，ONED，PED，APED，OPED。
- 中断工作（TASK）中不能使用的指令

  GOEND。
- 功能块（Function Block）中不支持的指令

  LDP，ANDP，ORP，LDF，ANDF，ORF，PLS，PLF，NP，PN，MC/MCR，GOEND，应用命令中所有脉冲执行型指令。上述部分指令若还是要使用在功能块中，可以替代的指令如下：

功能块中不可使用指令	功能块中替代指令
LDP/ANDP/ORP	PED/APED/OPED
LDF/ANDF/ORF	NED/ANED/ONED
PLS	-
PLF	-
NP	FB_NP
PN	FB_PN
MC	-
MCR	-
应用命令中所有脉冲执行型指令	注一

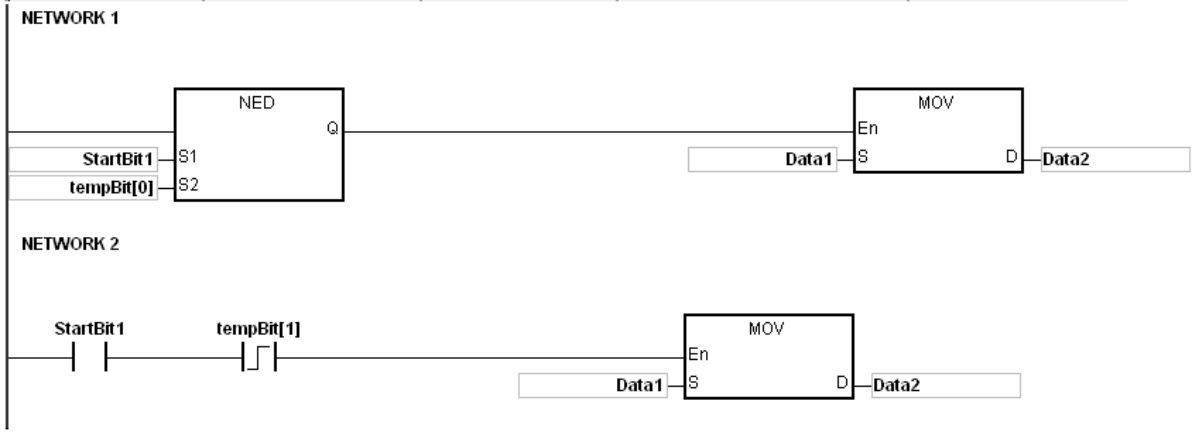
注一：脉冲执行型指令，不可使用在功能块中，如果要在 FB 功能块中达到脉冲执行型指令功能，可参考下列范例。

范例：

1. 先宣告系统用的 10 个 Bit 变量 tempBit[10]。
2. 当 StartBit1 由 OFF=>ON 时，方法一（NEXTWORK1）与方法二（NEXTWORK2）皆仅会执行一次 MOV 指令，皆可达到 MOVP 指令的功能，用户可以择一使用。
3. 系统用的 tempBit 变量，不可重复使用。

局部符号

类型	符号名称	地址	数据类型	初始值
VAR	tempBit	N/A [Auto]	ARRAY [10] OF BOOL	N/A
VAR	StartBit1	N/A [Auto]	WORD	N/A
VAR	Data1	N/A [Auto]	WORD	N/A
VAR	Data2	N/A [Auto]	WORD	N/A



- 建议使用于一般梯形图（Ladder）语言编辑的指令

  由于下表所列之指令皆须在启动执行时，以及关闭指令时得配合执行关闭动作，因此建议仅使用于一般梯形图语言编辑，避免采用只有启动执行一次的编程方式（如：功能块只调用一次），以及关闭指令就不再被扫描的执行方式（如：ST 与 SFC 语言编程）。若是该功能块会被程序单元多处调用执行时，也不建议使用。

指令类别	指令名称
计时计数	CNT、DCNT、DHSCS、DHSCR、DHSZ、DHSCY
接口设备	TKY、DTKY、HKY、DHKY、DSW、ARWS
便利	RAMP、DRAMP、MTR、INCD
模块读写	DPUCONF、PUSTAT
CANopen 专属通讯	CANRS、COPRW、COPWL、DCOPWL、ZRNM、ECAMD、ECAMS、ECAMA、ECAMC
文件寄存器写入	MEMW
通讯	DNETRW
其他	INFO

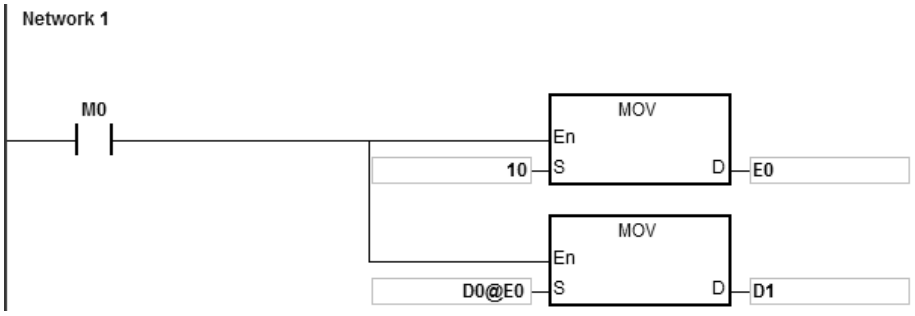
4.4 变址（E）说明

变址寄存器 E 是 16 位的数据寄存器，跟一般的寄存器一样可以被读、写，但主要功能是做变址寄存器使用，使用范围为 E0~E9。通常变址被使用在暂时性的局部变量应用，不建议当作全局变量。

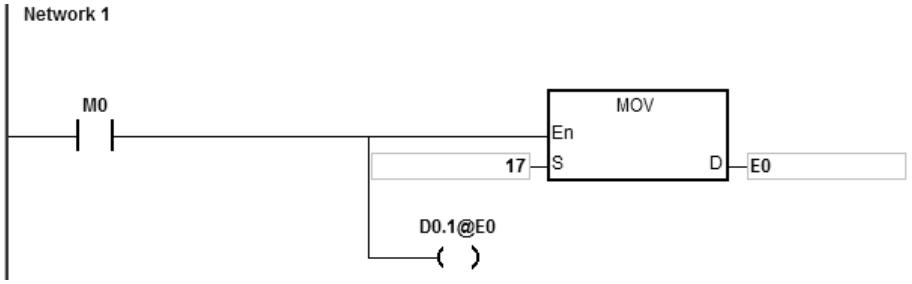
变址的使用方式范列如下：

1. 直接使用寄存器名称来修饰。

   当 M0=ON 时，E0=10，则 D0@E0=D（0+10）=D10，D1=D10。



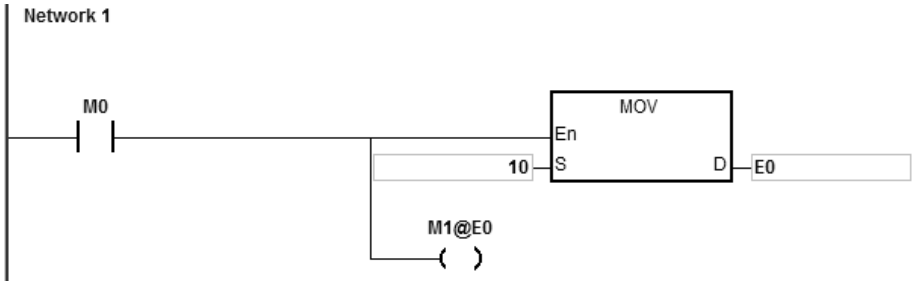
   当 M0=ON 时，E0=10，则计变址的计算方式为 D0.1@E0=D0.（1+10）=D0.11，因此 D0.11 会 ON。



   备注 1：AS 系列支持 D0.1@E0 变址修饰，但不支持 D0@E1.1@E0 这类双层变址修饰。

   备注 2：若 E0=17，则计变址的计算方式为 D0.1@E0=D0.（1+17）=D1.2，因此 D1.2 会 ON。Bit 的部份 1@E0=（1+17）=18，但 Bit 的部份最大数为 15，因此 18 除 16=1 余 2，表示进位 1 并余 2，所以最后修饰的结果为 D（0+1）.2=D1.2，则结果 D1.2 会 ON。

   当 M0=ON 时，E0=10，M1@E0=M（1+10）=M11，则 M11=ON。

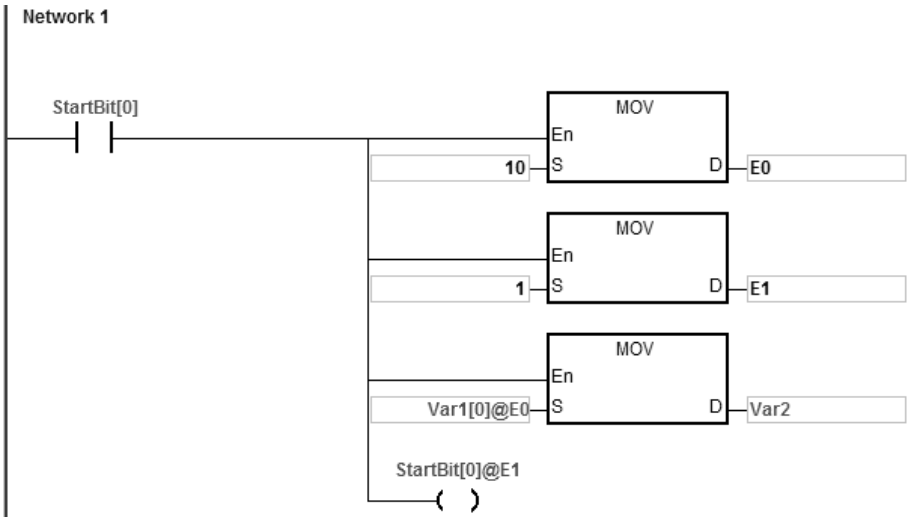


2. 先宣告变量再修饰。

   - 先在 ISPSoft 的变量表宣告三个变量 StartBit，Var1，Var2
     StartBit 的类型为 BOOL 数组，大小为 2 个 Bit，范围为 StartBit[0]~StartBit[1]
     Var1 的类型为 WORD 数组，大小为 11 个 WORD，范围为 Var1[0]~Var1[10]
     Var2 的类型为 WORD，大小为 1 个 WORD

局部符号				
类型	符号名称	地址	数据类型	初始值
VAR	StartBit	N/A [Auto]	BOOL	FALSE
VAR	Var1	N/A [Auto]	ARRAY [11] OF WORD	N/A
VAR	Var2	N/A [Auto]	WORD	0

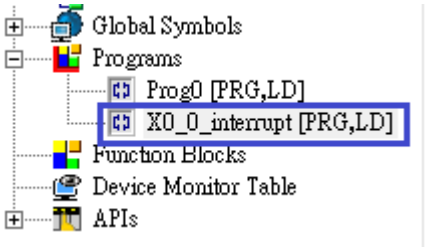
- 当 StartBit[0]=ON 时，E0=10，E1=1，Var1[0]@E0=Var1[10]，所以 Var2=Var1[10]，StrartBit[0]@E1=StartBit[1]，所以 StartBit=ON。

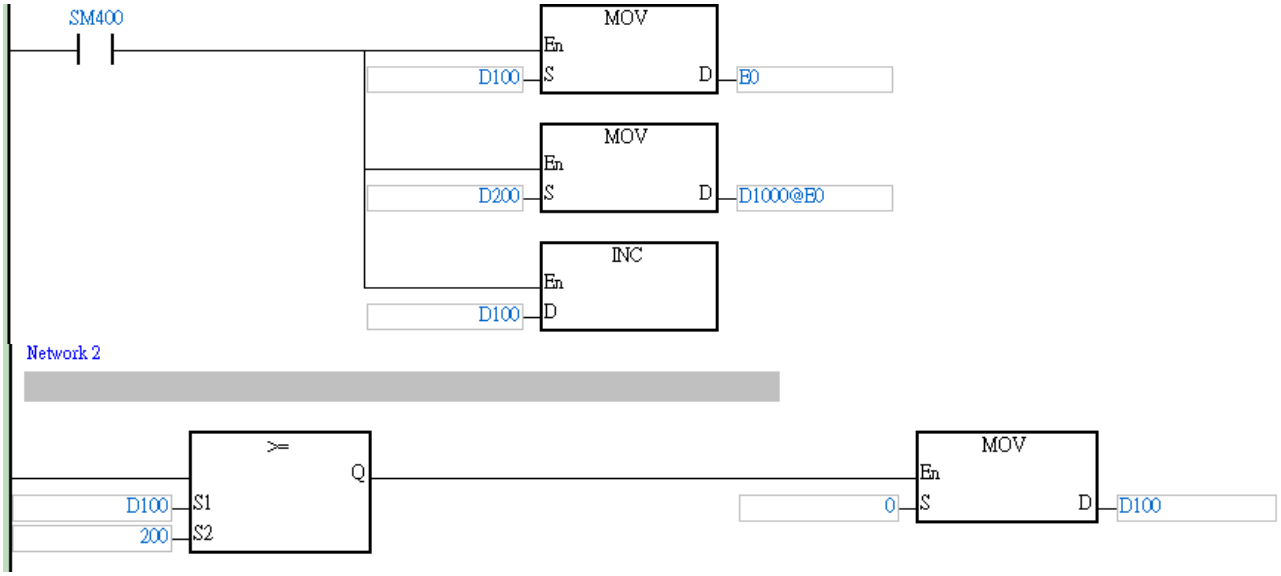


补充说明：当使用 ISPSoft 宣告变量时，又要将变量变址偏移，必需注意变量偏移到哪个地址，避免程序造成误动作。

3. 中断程序内使用变址装置说明：

- E0~E9 的装置元件，可分为两个独立内存区块，分别为一般程序区（POU, SFC）与功能块（FB）使用，以及中断程序区使用，此两个区块内的 E0~E9 分别各自独立，不会相互影响。
- ISPSoft 软件监看之 E 装置数值为一般程序区的 E 装置。
- 举例说明：假设用户同时应用变址装置 E0 于一般程序区与中断程序内，当中断发生时，PLC 在进入中断程序前，会自动将一般程序区之 E0 数值暂时储存，待中断程序执行完毕之后，再另行回复原有 E0 数值，以避免一般程序区的 E0 内存数值被中断程序影响，而改变一般程序区当下的执行结果。
- 若用户有需要监看中断内的 E 装置数值，或者一般程序区也要共享此 E 装置的数值时，请参考下面介绍之使用范例。（注：不建议 E 装置当做全局变量使用，若当做全局变量应用，则请另行宣告或定义其他装置元件）
- 范例：定义 D100 为 E0 变址装置的全局变量，当每次 X0.0 外部中断产生时，将 D200 数值依照累计的次数分别储存至 D1000~D1199 之间。中断使用 E 装置的范例程序请参考下列图标。





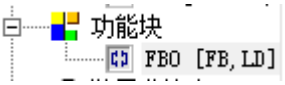
4

4.5 指针寄存器（PR）说明

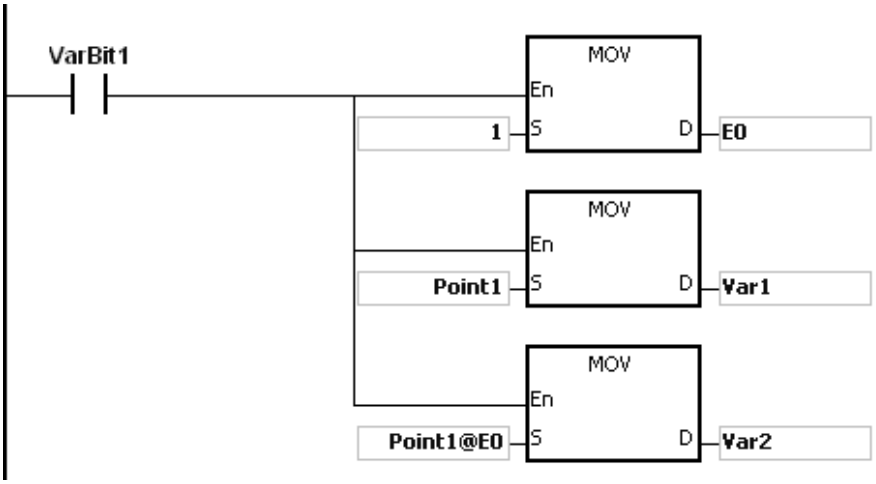
- ISPSoft 有支持功能块的功能，当功能块的变量类别宣告为 VAR_IN_OUT，数据类型为 POINTER（指针）时，此变量将配置 PR 装置，PR 可以传入的装置为（X，Y，D）以及 Address 为 ISPSoft 自动配置的变量。
- 每个功能块中可以宣告的 PR 个数为 16 个：PR0 ~ PR15，或是使用位 PR0.0 ~ PR15.15。

范例：

1. 先使用 ISPSoft 建立一个程序 POU（程序组织单元）。
2. 建立一个功能块，名称为 FB0。



3. FB0 功能块中的程序。



4. 在 FB0 功能块中的变量宣告。

   类别选择 VAR_IN_OUT，符号名称为 Point1，数据类型选择 POINTER，此变量将配置为指针寄存器。

局部符号				
类型	符号名称	地址	数据类型	初始值
VAR	VarBit1	N/A [Auto]	BOOL	FALSE
VAR	Var1	N/A [Auto]	WORD	0
VAR	Var2	N/A [Auto]	WORD	0
VAR_IN_OUT	Point1	N/A [Auto]	POINTER	N/A

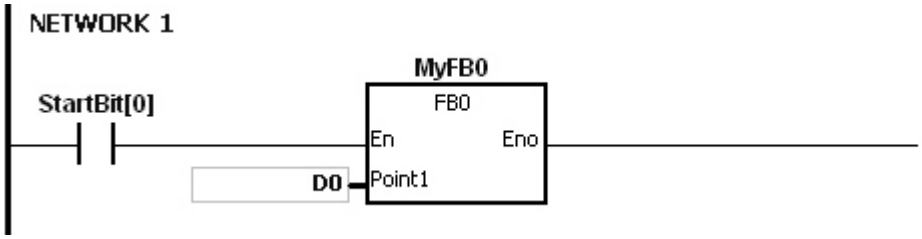
5. POU 中的变数宣告。

局部符号				
类型	符号名称	地址	数据类型	初始值
VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	[2(FALSE)]
VAR	CVar1	N/A [Auto]	ARRAY [2] OF WORD	[2(0)]
VAR	MyFB0	N/A [Auto]	FB0	N/A

6. 然后在 POU 中调用 FB0。

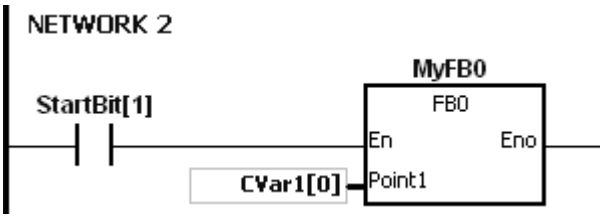
7. POU 中的程式。

   方法一：当 StartBit[0]=ON 时，D0 的地址将传入给 FB0 中的 Point1。



   当 FB0 功能块中的 VarBit1=ON 时，E0=1，Var1=D0，Point1@E0=D(0+1)=D1，所以 Var2=D1。

   方法二：当 StartBit[1]=ON 时，CVar1[0]的地址将传入给 FB0 中的 Point1。



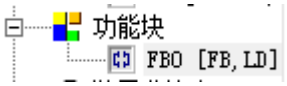
   当 FB0 功能块中的 VarBit1=ON 时，E0=1，Var1=CVar1[0]，Point1@E0=CVar1(0+1)=Cvar1[1]，所以 Var2=CVar1[1]。

4.6 定时器指针寄存器（TR）说明

- ISPSoft 有支持功能块的功能，如果要在功能块中使用定时器时，必需在功能块中宣告一个 TR 装置来并在调用功能块时传入 T 装置的指针。
- 当功能块的变量类别宣告为 VAR_IN_OUT，数据类型为 T_POINTER 时，此变量将配置 TR 装置，TR 可以传入的装置为（T）以及 ISPSoft 配置为定时器的变量。
- 每个功能块中可以宣告 TR 的个数为 8 个：TR0~TR7。
- 如果要在功能块中使用指令时，操作数区域中有支持装置 T，则必须使用 TR 指针寄存器来运算。

范例：

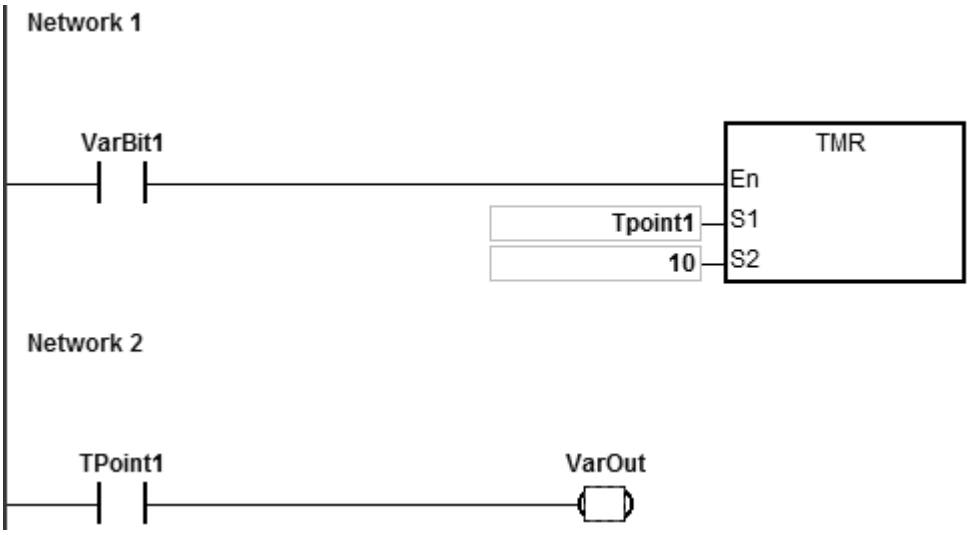
1. 先使用 ISPSoft 建立一个程序 POU（程序组织单元）。
2. 建立一个功能块，名称为 FB0。



3. 在 FB0 功能块中的变数宣告。
   类别选择 VAR_IN_OUT，符号名称名称为 TPoint1，数据类型选择 T_POINTER，此变量将配置为定时器指针寄存器。

局部符号				
类型	符号名称	地址	数据类型	初始值
VAR	VarBit1	N/A [Auto]	BOOL	FALSE
VAR_IN_OUT	TPoint1	N/A [Auto]	T_POINTER	N/A
VAR	VarOut	N/A [Auto]	BOOL	FALSE

4. FB0 功能块中的程序。



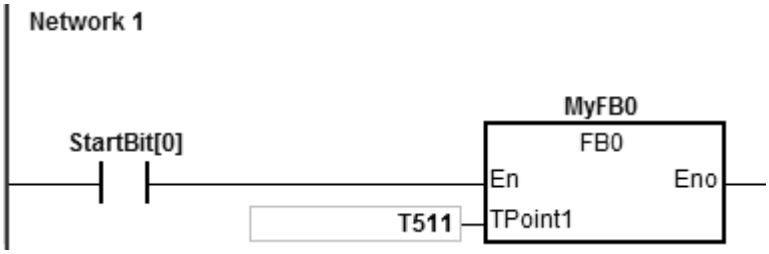
5. POU 中的变数宣告。
   其中 CVar1 必需宣告为 TIMER 的数据类型

局部符号				
类型	符号名称	地址	数据类型	初始值
VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	[2(FALSE)]
VAR	CVar1	T0	TIMER	0
VAR	MyFB0	N/A [Auto]	FB0	N/A

6. 然后在 POU 中调用 FB0。

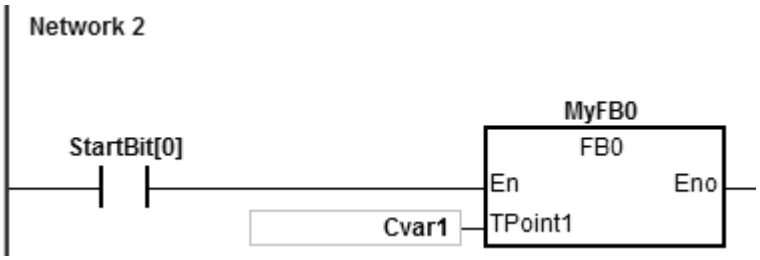
7. POU 中的程序。

方法一：当 StartBit[0]=ON 时，T511 的地址将传入给 FB0 中的 TPoint1。



当 FB0 功能块中的 VarBit1=ON 时，TMR 指令执行，TPoint1（T511）开始计数，当 TPoint1 计数到达时 VarOut=ON。

方法二：当 StartBit[1]=ON 时，CVar1[0]的地址将传入给 FB0 中的 TPoint1。



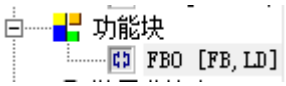
当 FB0 功能块中的 VarBit1=ON 时，TMR 指令执行，TPoint1（CVar1）开始计数，当 TPoint1 计数到达时 VarOut=ON。

4.7 16 位计数器指针寄存器（CR）说明

- ISPSoft 有支持功能块的功能，如果要在功能块中使用 16 位计数器时，必需在功能块中宣告一个 CR 装置，并在调用功能块时传入 C 装置的指针。
- 当功能块的变量类别宣告为 VAR_IN_OUT，数据类型为 C_POINTER 时，此变量将配置 CR 装置，CR 可以传入的装置为（C）以及 ISPSoft 配置为计数器的变量。
- 每个功能块中可以宣告 CR 的个数为 8 个：CR0~CR7。
- 如果要在功能块中使用指令时，操作数区域中有支持装置 C，则必须使用 CR 指针寄存器来运算。

范例：

1. 先使用 ISPSoft 建立一个程序 POU（程序组织单元）。
2. 建立一个功能块，名称为 FB0。



3. 在 FB0 功能块中的变数宣告。

类别选择 VAR_IN_OUT，符号名称为 CPoint1，数据类型选择 C_POINTER，此变量将配置为计数器指针寄存器。

局部符号

类型	符号名称	地址	数据类型	初始值
VAR	VarBit1	N/A [Auto]	BOOL	FALSE
VAR_IN_OUT	CPoint	N/A [Auto]	C_POINTER	N/A

4. FB0 功能块中和程序。

5. POU 中的变数宣告。

   其中 CVar1 必需宣告为 COUNTER 的数据类型。

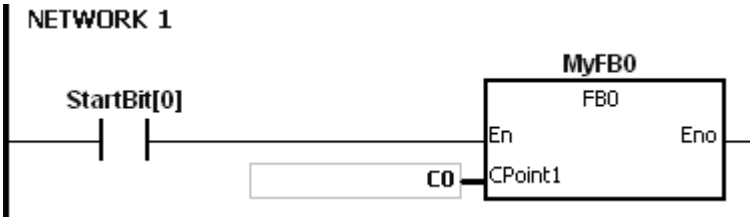
局部符号

类型	符号名称	地址	数据类型	初始值
VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	[2(FALSE)]
VAR	CVar1	C1	COUNTER	N/A
VAR	MyFB0	N/A [Auto]	FB0	N/A

6. 然后在 POU 中调用 FB0。

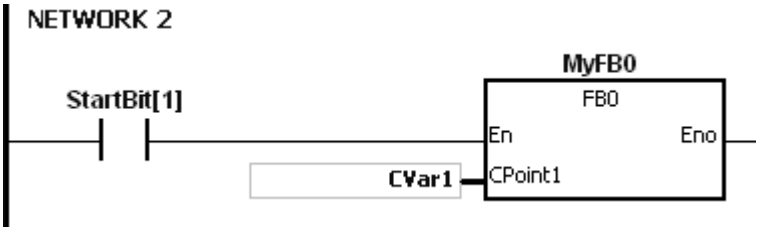
7. POU 中的程序。

   方法一：当 StartBit[0]=ON 时，C0 的地址将传入给 FB0 中的 CPoint1。



   当 FB0 功能块中的 VarBit1=ON 时，CPoint1（C0）=ON。

   方法二：当 StartBit[1]=ON 时，CVar1 的地址将传入给 FB0 中的 CPoint1。



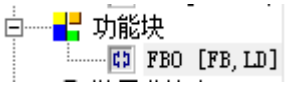
   当 FB0 功能块中的 VarBit1=ON 时，CPoint1（CVar1）=ON。

4.8 32 位计数器指针寄存器（HCR）说明

- ISPSoft 有支持功能块的功能，如果要在功能块中使用 32 位计数器时，必需在功能块中宣告一个 HCR 装置，并在调用功能块时传入 HC 装置的指针。
- 当功能块的变量类别宣告为 VAR_IN_OUT，数据类型为 HC_POINTER 时，此变量将配置 HCR 装置，HCR 可以传入的装置为（HC）以及 ISPSoft 配置为计数器的变量。
- 每个功能块中可以宣告 HCR 的个数为 8 个：HCR0~HCR7。
- 如果要在功能块中使用指令时，操作数区域中有支持装置 HC，则必须使用 HCR 指针寄存器来运算。

范例：

1. 先使用 ISPSoft 建立一个程序 POU（程序组织单元）。
2. 建立一个功能块，名称为 FB0。



3. 在 FB0 功能块中的变数宣告。

   类别选择 VAR_IN_OUT，符号名称为 HCPoint1，数据类型选择 HC_POINTER，此变量将配置为计数器指针寄存器。

局部符号				
类型	符号名称	地址	数据类型	初始值
VAR	VarBit1	N/A [Auto]	BOOL	FALSE
VAR_IN_OUT	HCPoint1	N/A [Auto]	HC_POINTER	N/A

4. FB0 功能块中的程序。

5. POU 中的变数宣告。

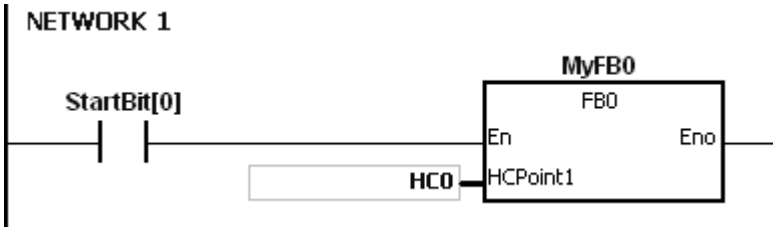
   其中 CVar1 必需宣告为 COUNTER 的数据类型，并自行在地址字段中填入 HC 装置的实际地址。

局部符号				
类型	符号名称	地址	数据类型	初始值
VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	[2(FALSE)]
VAR	CVar1	HC1	COUNTER	N/A
VAR	MyFB0	N/A [Auto]	FB0	N/A

6. 然后在 POU 中调用 FB0。

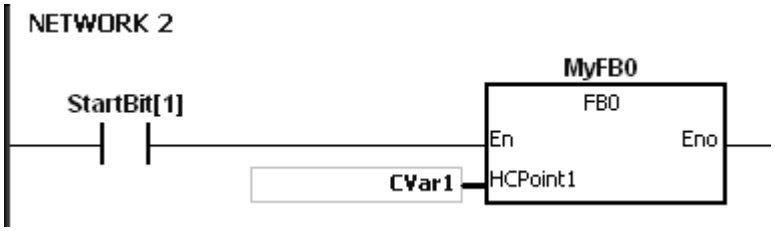
7. 在 POU 中调用 FB0。

   方法一：当 StartBit[0]=ON 时，HC0 的地址将传入给 FB0 中的 HCPoint1。



   当 FB0 功能块中的 VarBit1=ON 时，HCPoint1（HC0）=ON。

   方法二：当 StartBit[1]=ON 时，CVar1 的地址将传入给 FB0 中的 HCPoint1。



   当 FB0 功能块中的 VarBit1=ON 时，HCPoint1（CVar1）=ON。

4.9 文件寄存器（FR）说明

- AS 系列针对大量参数储存的需求，提供文件寄存器（FR）供用户储存大量的参数数据。
- 用户可透过 ISPSoft 上的文件寄存器编辑功能，进行参数的编辑、上载与下载功能。
- PLC 程序执行过程中，可直接读取 FR 的内容值，若有需要写入 FR 时，请参考 API2303 MEMW 指令说明。

5

第5章 基本指令

目录

5.1 基本指令一览表 5-2
5.2 基本指令说明 5-3

5.1 基本指令一览表

指令码	功能	操作数	运行时间（us）
LD/AND/OR	载入/串联/并联 A 接点	DX、X、Y、M、SM、S、T、C、HC、D	0.025
LDI/ANI/ORI	载入/串联/并联 B 接点	DX、X、Y、M、SM、S、T、C、HC、D	0.03
OUT	驱动线圈	DY、Y、M、SM、S、T、C、HC、D	0.04
SET	动作保持（ON）	DY、Y、M、SM、S、T、C、HC、D	0.04
MC/MCR	共通串联接点之连结/解除	N	0.24
LDP/ANDP/ORP	上升沿检出动作开始/串联/并联连接	DX、X、Y、M、SM、S、T、C、HC、D	0.22
LDF/ANDF/ORF	下降沿检出动作开始/串联/并联连接	DX、X、Y、M、SM、S、T、C、HC、D	0.22
PED/APED/OPED	上升沿检出动作开始/串联/并联连接	X、Y、M、SM、S、T、C、HC、D	0.22
NED/ANED/ONED	下降沿检出动作开始/串联/并联连接	X、Y、M、SM、S、T、C、HC、D	0.22
PLS	上微分输出	Y、M、SM、S、D	0.22
PLF	下微分输出	Y、M、SM、S、D	0.22
INV	运算结果反相	–	0.22
NP	上升沿触发指令	–	0.24
PN	下降沿触发指令	–	0.24
FB_NP	上升沿触发指令	Y、M、S、D	0.24
FB_PN	下降沿触发指令	Y、M、S、D	0.24

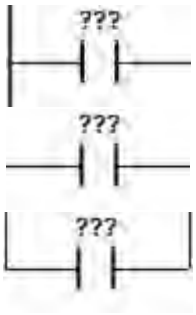
5.2 基本指令说明

指令码	操作数	功能
LD/AND/OR	**S**	载入/串联/并联 A 接点

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
S	●		●	●	●	○	●	●	●	●	●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

符号：



LD

AND

OR

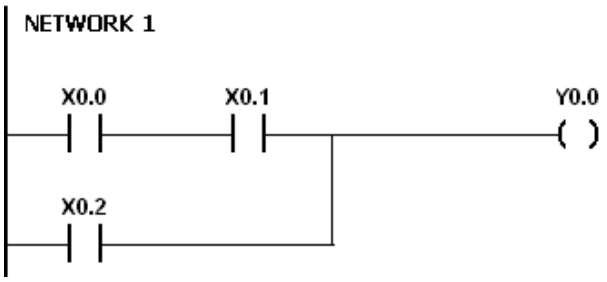
S ： 指定装置

指令说明：

1. LD 指令用于左母线开始的 A 接点或一个接点回路块开始的 A 接点，它的作用是把当前内容保存，同时把取来的接点状态存入累积寄存器内。
2. 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。
3. 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：

1. 载入 X0.0 的 A 接点，串联 X0.1 的 A 接点，并联 X0.2 的 A 接点，驱动 Y0.0 线圈。
2. 当 X0.0 和 X0.1=ON 或 X0.2=ON 时，Y0.0=ON。

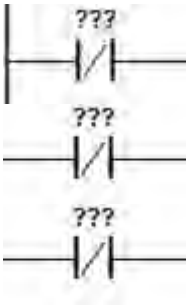


指令码	操作数	功能
LDI/ANI/ORI	**S**	载入/串联/并联 B 接点

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
S	●		●	●	●	○	●	●	●	●	●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

符号：



LDI

ANI

ORI

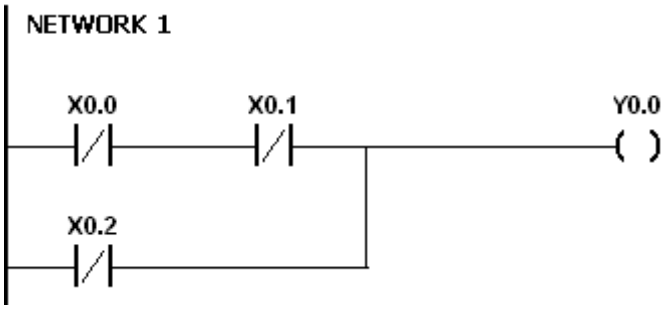
S ： 指定装置

指令说明：

1. LDI 指令用于左母线开始的 B 接点或一个接点回路块开始的 B 接点，它的作用是把当前内容保存，同时把取来的接点状态存入累积寄存器内。
2. ANI 指令用于 B 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。
3. 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：

1. 载入 X0.0 的 B 接点，串联 X0.1 的 B 接点，并联 X0.2 的 B 接点，驱动 Y0.0 线圈。
2. 当 X0.0 和 X0.1=OFF 或 X0.2=OFF 时，Y0.0=ON。

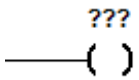


指令码	操作数	功能
OUT	**D**	驱动线圈

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
D		●		●	●	○	●				●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

符号：



D ： 指定装置

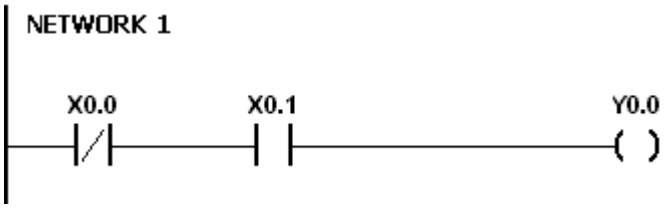
指令说明：

1. 将 OUT 指令之前的逻辑运算结果输出至指定的组件。
2. 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：

1. 加载 X0.0 的 B 接点，串联 X0.1 的 A 接点，驱动 Y0.0 线圈。
2. 当 X0.0=OFF 且 X0.1=ON 时，Y0.0=ON。

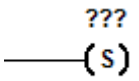


指令码	操作数	功能
SET	**D**	动作保持（ON）

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
D		●		●	●	○	●				●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

符号：



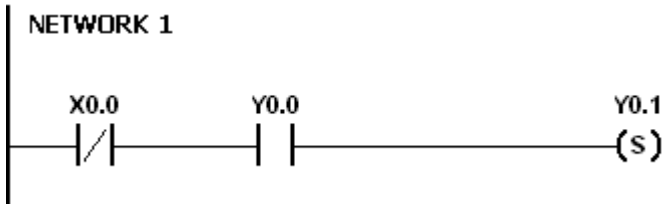
D ：指定装置

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。

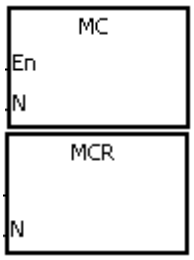
程序范例：

1. 载入 X0.0 之 B 接点，串联 Y0.0 之 A 接点，Y0.1 动作保持（ON）。
2. 当 X0.0=OFF 且 Y0.0=ON 时，Y0.1=ON 且即使运算结果改变，Y0.1 亦保持 ON 的状态。



指令码	操作数	功能
MC/MCR	**N**	共通串联接点之连结/解除

符号：



MC

MCR

N ：循环程序结构层数　　N0~N31

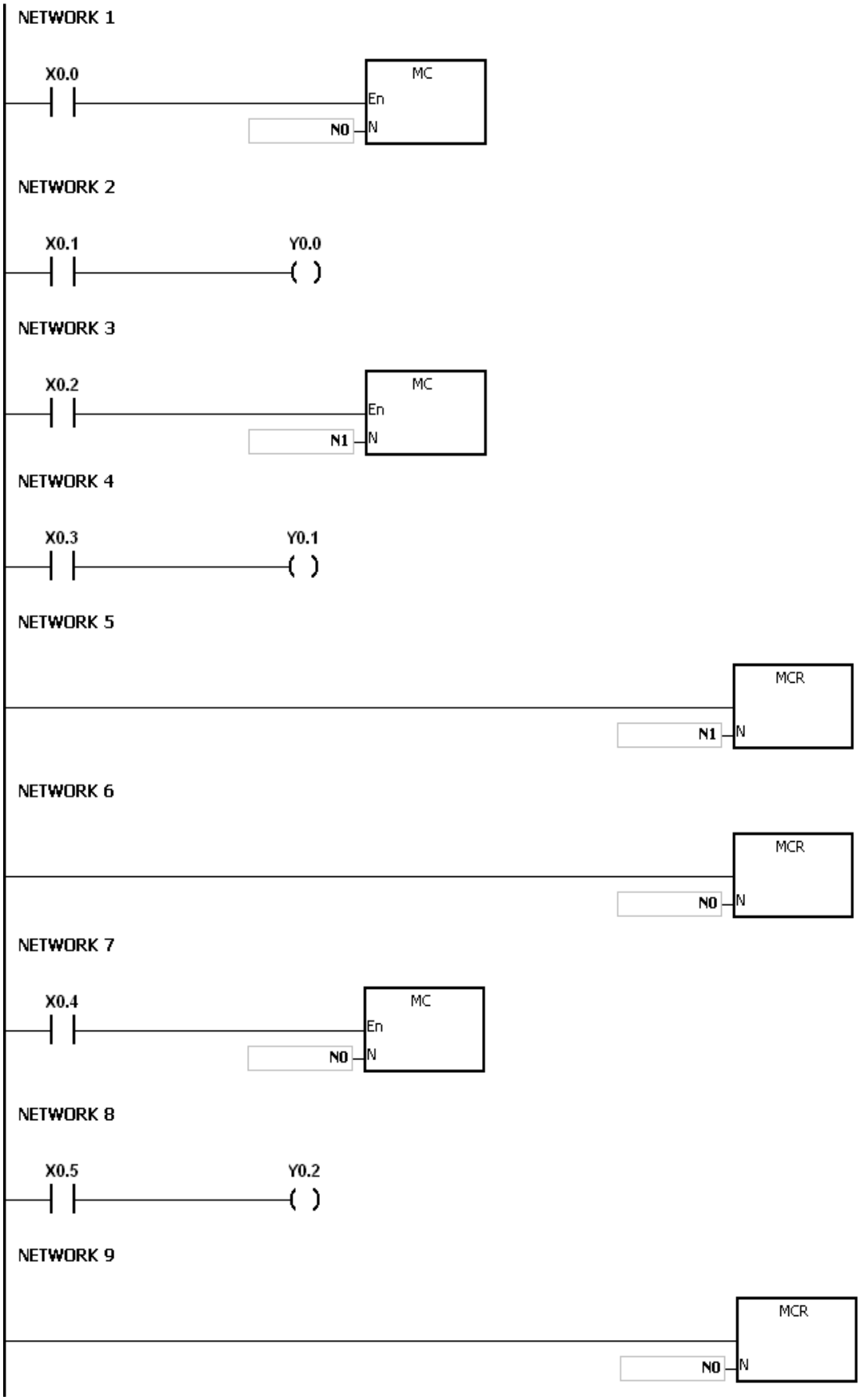
指令说明：

1. MC 为主控起始指令，当 MC 指令执行时，位于 MC 与 MCR 指令之间的指令照常执行。当 MC 指令 OFF 时，位于 MC 与 MCR 指令之间的指令动作如下所示：

指令区分	说明
一般定时器	计时值归零，线圈失电，接点不动作
功能块用定时器	计时值归零，线圈失电，接点不动作
运算型定时器	线圈失电，计时值及接点保持目前状态
计数器	线圈失电，计数值及接点保持目前状态
OUT 指令驱动的线圈	全部不受电
SET，RST 指令驱动的组件	保持目前状态
应用指令	全部不动作，但 FOR-NEXT 循环回路仍会来回执行 N 次，但 FOR-NEXT 间的任何指令依 MC-MCR 之间其它指令相同动作

2. MCR 为主控结束指令，置于主控程序最后，在 MCR 指令之前不可有接点指令。
3. MC-MCR 主控程序指令支持循环程序结构，最多可 32 层，使用时依 N0~N31 的顺序，请参考程序范例所示。

程序范例：

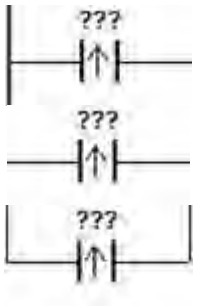

NETWORK 1
X0.0
MC
En
N0
N
NETWORK 2
X0.1
Y0.0
NETWORK 3
X0.2
MC
En
N1
N
NETWORK 4
X0.3
Y0.1
NETWORK 5
MCR
N1
N
NETWORK 6
MCR
N0
N
NETWORK 7
X0.4
MC
En
N0
N
NETWORK 8
X0.5
Y0.2
NETWORK 9
MCR
N0
N

指令码	操作数	功能
LDP/ANDP/ORP	**S**	上升沿检出动作开始/串联/并联连接

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
S	●		●	●	●	○	●	●	●	●	●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

符号：



LDP

ANDP

ORP

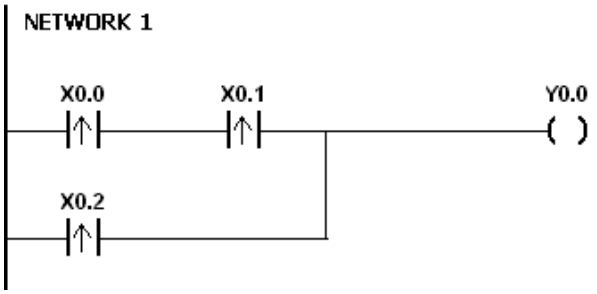
S ： 指定装置

指令说明：

1. 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。
2. ANDP 指令用于接点上升沿检出的串联连接。
3. ORP 指令用于接点上升沿检出的并联连接。
4. 上升沿检出动作，必须在指令扫描到的时候才会得知装置目前的状态，下一次扫描到指令才会判断装置状态是否有变化。
5. 子程序中请使用 PED、APED、OPED 指令。

程序范例：

1. X0.0 上升沿检出动作开始，串联 X0.1 的上升沿检出，并联 X0.2 的上升沿检出，驱动 Y0.0 线圈。
2. 当 X0.0 和 X0.1 同时由 OFF 到 ON 或 X0.2 由 OFF 到 ON 时，Y0.0 会 ON 一个扫描周期。

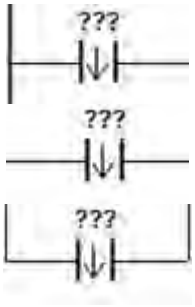


指令码	操作数	功能
LDF/ANDF/ORF	**S**	下降沿检出动作开始/串联/并联连接

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
S	●		●	●	●	○	●	●	●	●	●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

符号：



LDF

ANDF

ORF

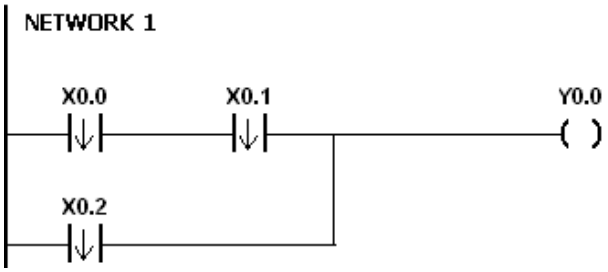
S ：指定装置

指令说明：

1. 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。
2. ANDF 指令用于接点下降沿检出的串联连接。
3. ORF 指令用于接点下降沿检出的并联连接。
4. 下降沿检出动作，必须在指令扫描到的时候才会得知装置目前的状态，下一次扫描到指令才会判断装置状态是否有变化
5. 子程序中请使用 NED、ANED、ONED 指令。

程序范例：

1. X0.0 下降沿检出动作开始，串联 X0.1 的下降沿检出，并联 X0.2 的下降沿检出，驱动 Y0.0 线圈。
2. 当 X0.0 和 X0.1 同时由 ON 到 OFF 或 X0.2 由 ON 到 OFF 时，Y0.0 会 ON 一个扫描周期。

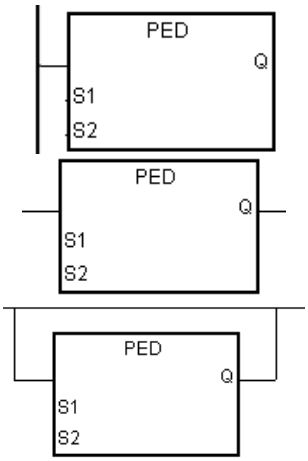


指令码	操作数	功能
PED/APED/OPED	**S_1**，**S_2**	上升沿检出动作开始/串联/并联连接

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
S_1			●	●	●	○	●	●	●	●	●
S_2				●	●		●				●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2	●												

符号：



PED　　**S_1**　：　指定装置

APED　　**S_2**　：　PLC 内部使用

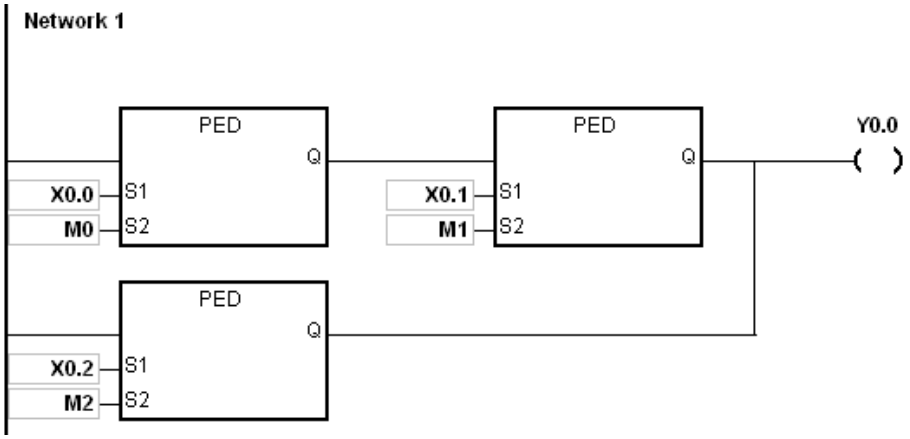
OPED　　**Q**　：　执行结果状态输出

指令说明：

1. PED 指令用法上与 LDP 相同、APED 指令用法上与 ANDP 相同、OPED 指令用法上与 ORP 相同，不同之处在于需多指定一个 **S_2** 的 Bit 装置，让 PLC 内部记住该接点的上一次状态，此 **S_2** 装置在程序中的其它地方请勿重复使用，否则可能会造成执行结果错误。
2. APED 指令用于接点上升沿检出的串联连接。
3. OPED 指令用于接点上升沿检出的并联连接。
4. 上升沿检出动作，必须在指令扫描到的时候才会得知装置目前的状态，下一次扫描到指令才会判断装置状态是否有变化。
5. PED/APED/OPED 指令只能在功能块中使用。
6. Q 执行结果状态输出由指令执行完成后自行输出，用户不需填入任何装置。

程序范例：

1. X0.0 上升沿检出动作开始，串联 X0.1 的上升沿检出，并联 X0.2 的上升沿检出，驱动 Y0.0 线圈。
2. 当 X0.0 和 X0.1 同时由 OFF 到 ON 或 X0.2 由 OFF 到 ON 时，Y0.0 会 ON 一个扫描周期。

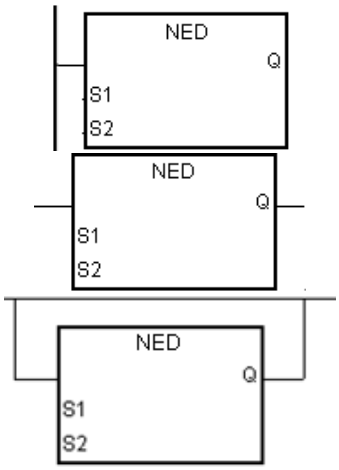


指令码	操作数	功能
NED/ANED/ONED	S_1，S_2	下降沿检出动作开始/串联/并联连接

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
S_1			●	●	●	○	●	●	●	●	●
S_2				●	●		●				●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2	●												

符号：



NED

ANED

ONED

S_1 ： 指定装置

S_2 ： PLC 内部使用

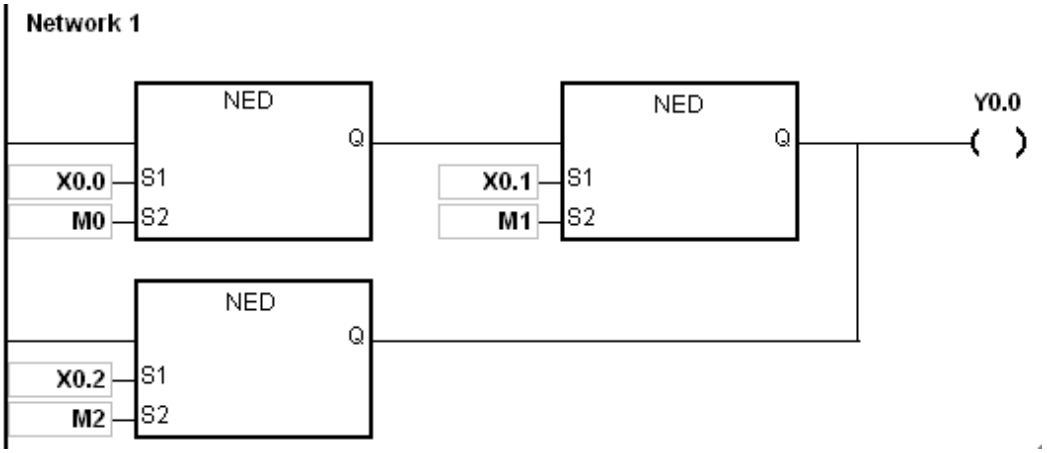
Q ： 执行结果状态输出

指令说明：

1. NED 指令用法上与 LDF 相同、ANED 指令用法上与 ANDF 相同、ONED 指令用法上与 ORF 相同，不同之处在于需多指定一个 **S_2** 的 Bit 装置，让 PLC 内部记住该接点的上一次状态，此 **S_2** 装置在程序中的其它地方请勿重复使用，否则可能会造成执行结果错误。
2. ANED 指令用于接点下降沿检出的串联连接。
3. ONED 指令用于接点下降沿检出的并联连接。
4. 下降沿检出动作，必须在指令扫描到的时候才会得知装置目前的状态，下一次扫描到指令才会判断装置状态是否有变化。
5. NED/ANED/ONED 指令只能在功能块中使用。
6. Q 执行结果状态输出由指令执行完成后自行输出，用户不需填入任何装置。

程序范例：

1. X0.0 下降沿检出动作开始，串联 X0.1 的下降沿检出，并联 X0.2 的下降沿检出，驱动 Y0.0 线圈。
2. 当 X0.0 和 X0.1 同时由 ON 到 OFF 或 X0.2 由 ON 到 OFF 时，Y0.0 会 ON 一个扫描周期。

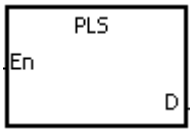


指令码	操作数	功能
PLS	**D**	上微分输出

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
D				●	●	○	●				●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

符号：



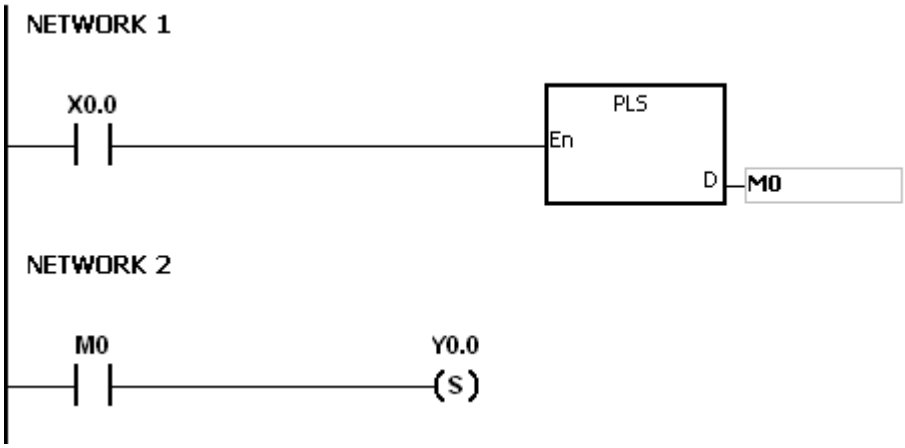
PLS | **D** ： 指定装置

指令说明：

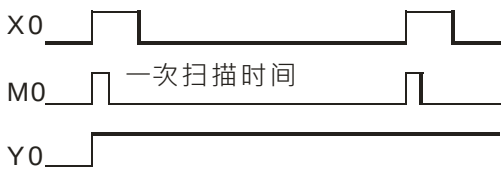
1. 上微分输出指令。当条件接点由 OFF 到 ON (上升沿触发) 时，PLS 指令被执行，D 送出一次脉冲，脉冲长度为一次扫描时间。
2. 功能块中请勿使用。
3. ISPSoft V3.06.01 版以上才支持 D 元件，以及 BOOL 类型宣告之变量。

程序范例：

当 X0.0=ON 时，M0 ON 一个 Pulse 的时间；M0=ON 时，Y0.0 Set 为 ON。



时序图：



指令码	操作数	功能
PLF	**D**	下微分输出

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
D				●	●	○	●				●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

符号：

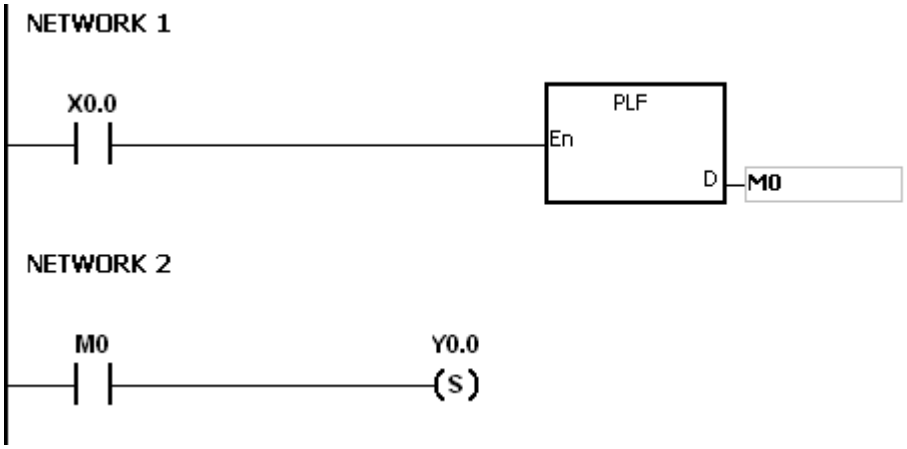
PLF | **D** :指定装置

指令说明：

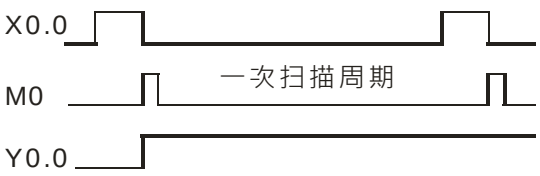
1. 下微分输出指令。当条件接点由 ON 到 OFF (下降沿触发) 时，PLF 指令被执行，D 送出一次脉冲，脉冲长度为一次扫描时间。
2. 功能块中请勿使用。
3. ISPSoft V3.17 版以上才支持 DF 元件，以及 BOOL 类型宣告之变量。

程序范例：

当 X0.0=ON 到 OFF 时，M0 ON 一个 Pulse 的时间；M0=ON 时，Y0.0 Set 为 ON。

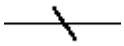


时序图：



指令码	操作数	功能
INV	-	运算结果反相

符号：

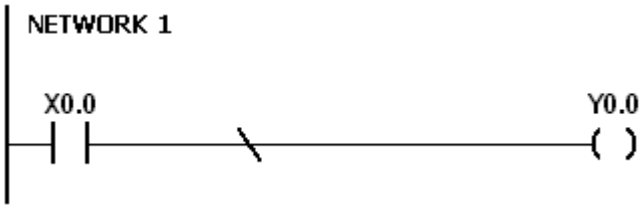


指令说明：

将 INV 指令之前的逻辑运算结果反相存入累积寄存器内。

程序范例：

当 X0.0=ON 时，Y0.0=OFF；X0.0=OFF 时，Y0.0=ON。



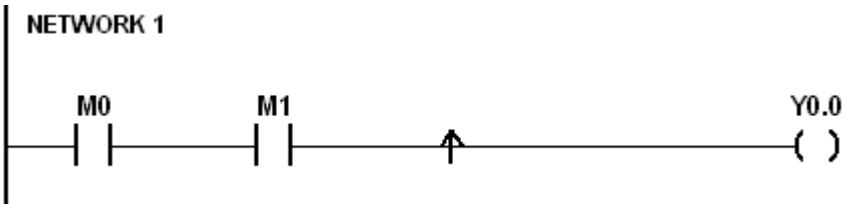
指令码	操作数	功能
NP	-	上升沿触发指令

符号：

指令说明：

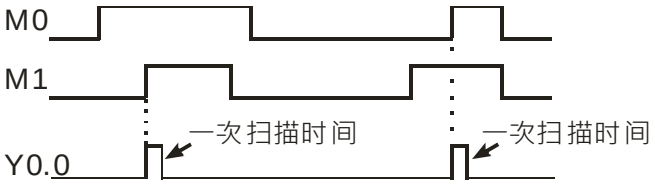
1. 当累积寄存器由 0 变为 1 时，此指令将使累积寄存器维持一次扫描周期的 1，然后第二次扫描周期之后，自动将累积寄存器改为 0。
2. 在 Function Block 中请使用 FB_NP 指令。

程序范例：



指令：		操作说明：
LD	M0	载入 M0 的 A 接点
AND	M1	串入 M1 之 A 接点
NP		上升沿触发指令
OUT	Y0.0	驱动线圈 Y0.0

时序图：



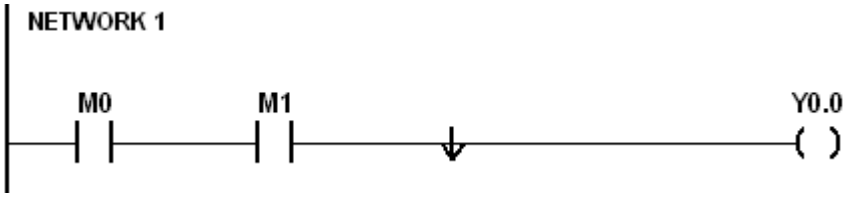
指令码	操作数	功能
PN	-	下降沿触发指令

符号：

指令说明：

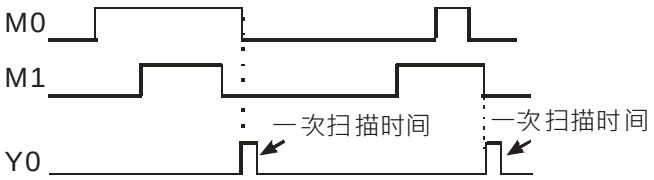
1. 当累积寄存器由 1 变为 0 时，此指令将使累积寄存器维持一次扫描周期的 1，然后第二次扫描周期之后，自动将累积寄存器改为 0。
2. 在 Function Block 中请使用 FB_P N 指令。

程序范例：



指令：		操作说明：
LD	M0	载入 M0 的 A 接点
AND	M1	串入 M1 之 A 接点
PN		下降沿触发指令
OUT	Y0.0	驱动线圈 Y0.0

时序图：

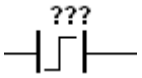


指令码	操作数	功能
FB_NP	**S**	上升沿触发指令

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
S				●	●		●				●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

符号：

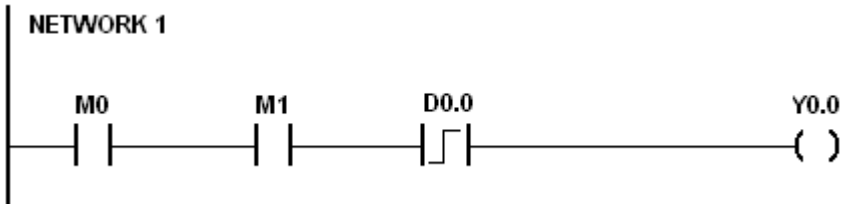


S : PLC 内部使用

指令说明：

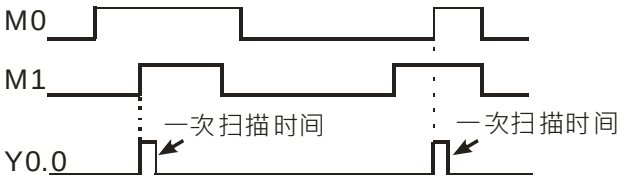
1. 当累积寄存器由 0 变为 1 时，此指令将使累积寄存器维持一次扫描周期的 1，然后第二次扫描周期之后，自动将累积寄存器改为 0。
2. **S** 装置，让 PLC 内部记住该接点的上一次状态，此 **S** 装置在程序中的其它地方请勿重复使用，否则可能会造成执行结果错误。
3. **本指令只能在 Function Block 中使用。**

程序范例：



指令：		操作说明：
LD	M0	载入 M0 的 A 接点
AND	M1	串入 M1 之 A 接点
FB_NP	D0.0	上升沿触发指令
OUT	Y0.0	驱动线圈 Y0.0

时序图：

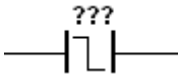


指令码	操作数	功能
FB_PN	**S**	下降沿触发指令

装置	DX	DY	X	Y	M	SM	S	T	C	HC	D
S				●	●		●				●

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

符号：

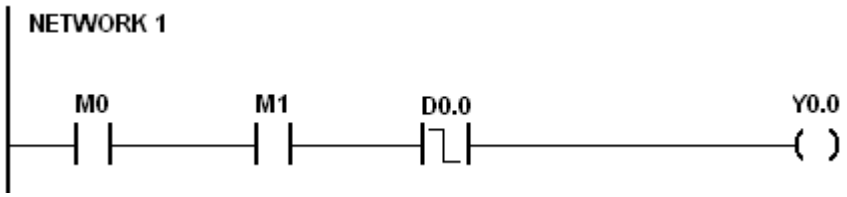


S ：PLC 内部使用

指令说明：

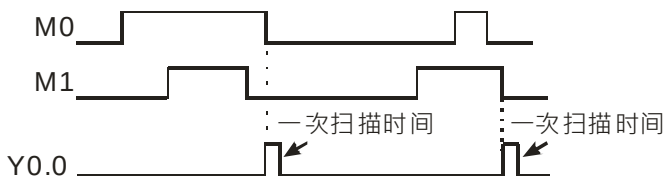
1. 当累积寄存器由 1 变为 0 时，此指令将使累积寄存器维持一次扫描周期的 1，然后第二次扫描周期之后，自动将累积寄存器改为 0。
2. **S** 装置，让 PLC 内部记住该接点的上一次状态，此 **S** 装置在程序中的其它地方请勿重复使用，否则可能会造成执行结果错误。
3. **本指令只能在 Function Block 中使用。**

程序范例：



指令：		操作说明：
LD	M0	载入 M0 的 A 接点
AND	M1	串入 M1 之 A 接点
FB_PN	D0.0	下降沿触发指令
OUT	Y0.0	驱动线圈 Y0.0

时序图：



MEMO

6

第6章 指令表

目录

6.1 比较操作指令 6-4
6.1.1 比较操作指令一览表 6-4
6.1.2 比较操作指令说明 6-7
6.2 四则运算指令说明 6-45
6.2.1 四则运算指令一览表 6-45
6.2.2 四则运算指令说明 6-46
6.3 数据转换指令说明 6-76
6.3.1 数据转换指令一览表 6-76
6.3.2 数据转换指令说明 6-77
6.4 数据转移指令說明 6-113
6.4.1 数据转移指令一览表 6-113
6.4.2 数据转移指令说明 6-114
6.5 程序跳转指令 6-137
6.5.1 程序跳转指令一览表 6-137
6.5.2 程序跳转指令说明 6-138
6.6 程序执行控制指令 6-146
6.6.1 程序执行控制指令一览表 6-146
6.6.2 程序执行控制指令说明 6-147
6.7 I/O 更新指令 6-156
6.7.1 I/O 更新指令一览表 6-156
6.7.2 I/O 更新指令说明 6-157
6.8 便利指令 6-163
6.8.1 便利指令一览表 6-163

6.8.2 便利指令说明 6-164
6.9 逻辑操作指令 6-208
6.9.1 逻辑操作指令一览表 6-208
6.9.2 逻辑操作指令说明 6-209
6.10 旋转指令 6-229
6.10.1 旋转指令一览表 6-229
6.10.2 旋转指令说明 6-230
6.11 计时计数指令 6-240
6.11.1 计时计数指令一览表 6-240
6.11.2 计时计数指令说明 6-241
6.12 移位指令 6-284
6.12.1 移位指令一览表 6-284
6.12.2 移位指令说明 6-285
6.13 数据处理指令 6-318
6.13.1 数据处理指令一览表 6-318
6.13.2 数据处理指令说明 6-319
6.14 结构建立指令 6-376
6.14.1 结构建立指令一览表 6-376
6.14.2 结构建立指令说明 6-377
6.15 模块的数据读写指令 6-384
6.15.1 模块的数据读写指令一览表 6-384
6.15.2 模块的数据读写指令说明 6-385
6.16 浮点数指令 6-500
6.16.1 浮点数指令一览表 6-500
6.16.2 浮点数指令说明 6-501
6.17 万年历指令 6-527
6.17.1 万年历指令一览表 6-527
6.17.2 万年历指令说明 6-528
6.18 外围设备指令 6-555
6.18.1 外围设备指令一览表 6-555

6.18.2 外围设备指令说明 6-556
6.19 通讯指令 6-569
6.19.1 通讯指令一览表 6-569
6.19.2 通讯指令說明 6-570
6.19.3 通讯标志与寄存器说明 6-639
6.20 其它指令 6-641
6.20.1 其它指令一览表 6-641
6.20.2 其它指令说明 6-642
6.21 字串处理指令 6-653
6.21.1 字串处理指令一览表 6-653
6.21.2 字串处理指令说明 6-654
6.22 以太网控制指令 6-703
6.22.1 以太网控制指令一览表 6-703
6.22.2 以太网控制指令说明 6-704
6.23 储存卡/文件寄存器读写指令 6-757
6.23.1 储存卡/文件寄存器读写指令一览表 6-757
6.23.2 储存卡/文件寄存器读写指令说明 6-758
6.24 任务控制指令 6-780
6.24.1 任务控制指令一览表 6-780
6.24.2 任务控制指令说明 6-781
6.25 SFC 控制指令 6-785
6.25.1 SFC 控制指令一览表 6-785
6.25.2 SFC 控制指令说明 6-786
6.26 高速输出指令 6-792
6.26.1 高速输出指令一览表 6-792
6.26.2 高速输出指令说明 6-793
6.27 台达专用 CANopen 和 EtherCAT 通讯指令 6-890
6.27.1 通讯指令一览表 6-890
6.27.2 通讯指令说明 6-891
6.27.3 台达专用 CANopen 通讯之常见问题与排除方法 6-962

6.1 比较操作指令

6.1.1 比较操作指令一览表

API	指令码（位）		P 指令	功能
	16	32		
0000	LD=	DLD=	–	S1 = S2
0001	LD<>	DLD<>	–	S1≠S2
0002	LD>	DLD>	–	S1 > S2
0003	LD>=	DLD>=	–	S1≧S2
0004	LD<	DLD<	–	S1 < S2
0005	LD<=	DLD<=	–	S1≦S2
0006	AND=	DAND=	–	S1 = S2
0007	AND<>	DAND<>	–	S1≠S2
0008	AND>	DAND>	–	S1 > S2
0009	AND>=	DAND>=	–	S1≧S2
0010	AND<	DAND<	–	S1 < S2
0011	AND<=	DAND<=	–	S1≦S2
0012	OR=	DOR=	–	S1 = S2
0013	OR<>	DOR<>	–	S1 ≠ S2
0014	OR>	DOR>	–	S1 > S2
0015	OR>=	DOR>=	–	S1≧S2
0016	OR<	DOR<	–	S1 < S2
0017	OR<=	DOR<=	–	S1≦S2
0018	–	FLD=	–	S1 = S2
0019	–	FLD<>	–	S1≠S2
0020	–	FLD>	–	S1 > S2
0021	–	FLD>=	–	S1≧S2
0022	–	FLD<	–	S1 < S2
0023	–	FLD<=	–	S1≦S2
0024	–	FAND=	–	S1 = S2
0025	–	FAND<>	–	S1≠S2
0026	–	FAND>	–	S1 > S2
0027	–	FAND>=	–	S1≧S2
0028	–	FAND<	–	S1 < S2
0029	–	FAND<=	–	S1≦S2
0030	–	FOR=	–	S1 = S2
0031	–	FOR<>	–	S1≠S2
0032	–	FOR>	–	S1 > S2

API	指令码（位）		P 指令	功能
	16	32		
0033	–	FOR>=	–	S1≧S2
0034	–	FOR<	–	S1 < S2
0035	–	FOR<=	–	S1≦S2
0036	LD$=	–	–	S1 = S2
0037	LD$<>	–	–	S1≠S2
0042	AND$=	–	–	S1 = S2
0043	AND$<>	–	–	S1≠S2
0048	OR$=	–	–	S1 = S2
0049	OR$<>	–	–	S1≠S2
0054	CMP	DCMP	✓	比较设定输出
0055	ZCP	DZCP	✓	区域比较
0056	–	FCMP	✓	浮点数比较
0057	–	FZCP	✓	浮点数区域比较
0058	MCMP	–	✓	矩阵比较
0059	CMPT=	–	✓	表格比较，=
0060	CMPT<>	–	✓	表格比较，<>
0061	CMPT>	–	✓	表格比较，>
0062	CMPT>=	–	✓	表格比较，>=
0063	CMPT<	–	✓	表格比较，<
0064	CMPT<=	–	✓	表格比较，<=
0065	CHKADR	–	–	接点类型指针寄存器地址检查
0066	LDZ=	DLDZ=	–	|S1-S2|=|S3|
0067	LDZ<>	DLDZ<>	–	|S1-S2|≠|S3|
0068	LDZ>	DLDZ>	–	|S1-S2| > |S3|
0069	LDZ>=	DLDZ>=	–	|S1-S2|≧|S3|
0070	LDZ<	DLDZ<	–	|S1-S2| < |S3|
0071	LDZ<=	DLDZ<=	–	|S1-S2|≦|S3|
0072	ANDZ=	DANDZ=	–	|S1-S2|=|S3|
0073	ANDZ<>	DANDZ<>	–	|S1-S2|≠|S3|
0074	ANDZ>	DANDZ>	–	|S1-S2| > |S3|
0075	ANDZ>=	DANDZ>=	–	|S1-S2|≧|S3|
0076	ANDZ<	DANDZ<	–	|S1-S2| < |S3|
0077	ANDZ<=	DANDZ<=	–	|S1-S2|≦|S3|
0078	ORZ=	DORZ=	–	|S1-S2|=|S3|
0079	ORZ<>	DORZ<>	–	|S1-S2|≠|S3|
0080	ORZ>	DORZ>	–	|S1-S2| > |S3|

API	指令码（位）		P 指令	功能
	16	32		
0081	ORZ>=	DORZ>=	–	|S1-S2|≧|S3|
0082	ORZ<	DORZ<	–	|S1-S2| < |S3|
0083	ORZ<=	DORZ<=	–	|S1-S2|≦|S3|

6.1.2 比较操作指令说明

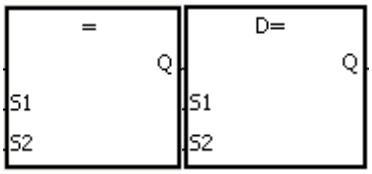
API	指令码			操作数	功能
0000~0005	D	LD※		S_1 · S_2	接点类型比较 LD※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 LD=跟 DLD=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

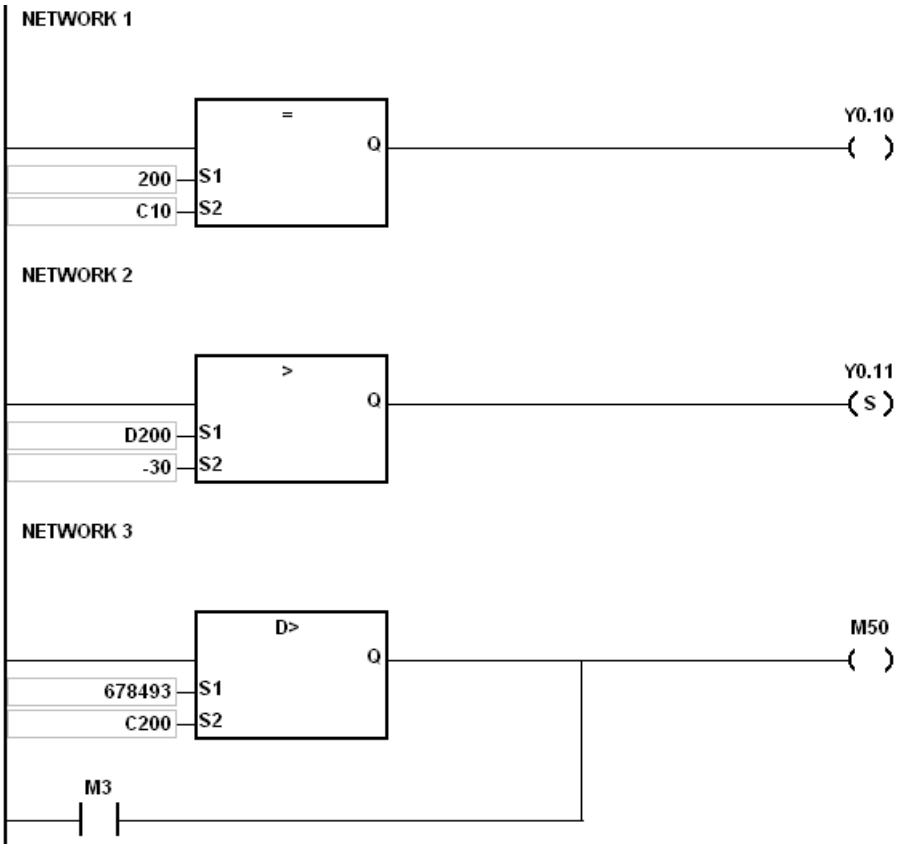
指令说明：

1. S_1 与 S_2 的内容做比较的指令。以"LD="做为例子，比较结果为"等于"时，该指令导通；"不等于"时，该指令不导通。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

API No.	16 位指令	32 位指令	导通条件	非导通条件
0000	LD =	**D**LD =	$S_1 = S_2$	$S_1 \neq S_2$
0001	LD < >	**D**LD < >	$S_1 \neq S_2$	$S_1 = S_2$
0002	LD >	**D**LD >	$S_1 > S_2$	$S_1 \leqq S_2$
0003	LD > =	**D**LD > =	$S_1 \geqq S_2$	$S_1 < S_2$
0004	LD <	**D**LD <	$S_1 < S_2$	$S_1 \geqq S_2$
0005	LD < =	**D**LD < =	$S_1 \leqq S_2$	$S_1 > S_2$

程序范例：

1. C10 的内容等于 200 时，Y0.10=ON。
2. 当 D200 的内容大于-30，Y0.11=ON 并保持住。
3. （C201，C200）的内容小于 678,493 或者是 M3=ON 时，M50=ON。



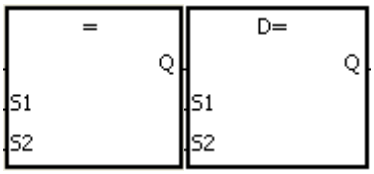
API	指令码			操作数	功能
0006~0011	D	AND※		S_1 · S_2	接点类型比较 AND※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 AND=跟 DAND=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

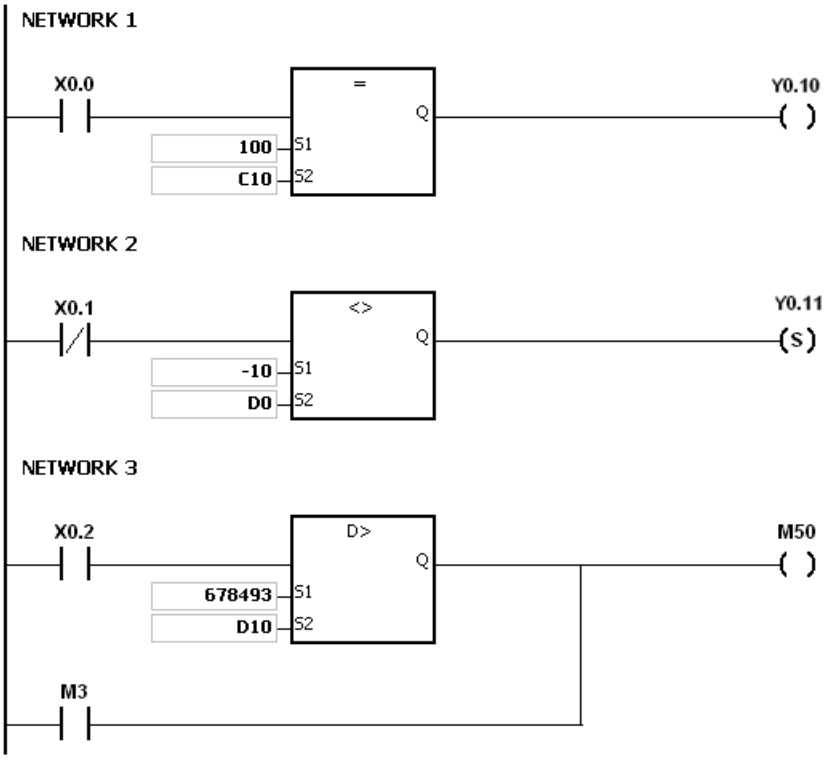
指令说明：

1. S_1 与 S_2 的内容做比较的指令。以"AND="做为例子，比较结果为"等于"时，该指令导通；"不等于"时，该指令不导通。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

API No.	16 位指令	32 位指令	导通条件	非导通条件
0006	AND =	**D**AND =	$S_1 = S_2$	$S_1 \neq S_2$
0007	AND < >	**D**AND < >	$S_1 \neq S_2$	$S_1 = S_2$
0008	AND >	**D**AND >	$S_1 > S_2$	$S_1 \leqq S_2$
0009	AND > =	**D**AND > =	$S_1 \geqq S_2$	$S_1 < S_2$
0010	AND <	**D**AND <	$S_1 < S_2$	$S_1 \geqq S_2$
0011	AND < =	**D**AND < =	$S_1 \leqq S_2$	$S_1 > S_2$

程序范例：

1. 当 X0.0=ON 时且 C10 的当前值又等于 100 时，Y0.10=ON。
2. 当 X0.1=OFF 而寄存器 D0 的内容又不等于-10 时，Y0.11=ON 并保持住。
3. 当 X0.2=ON 而且 32 位寄存器（D11，D0）的内容又小于 678,493 时或 M3=ON 时，M50=ON。



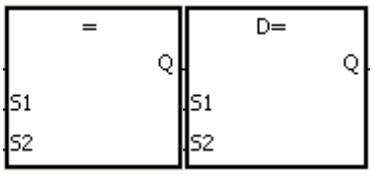
API	指令码			操作数	功能
0012~0017	D	OR※		S_1，S_2	接点类型比较 OR※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 OR=跟 DOR=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

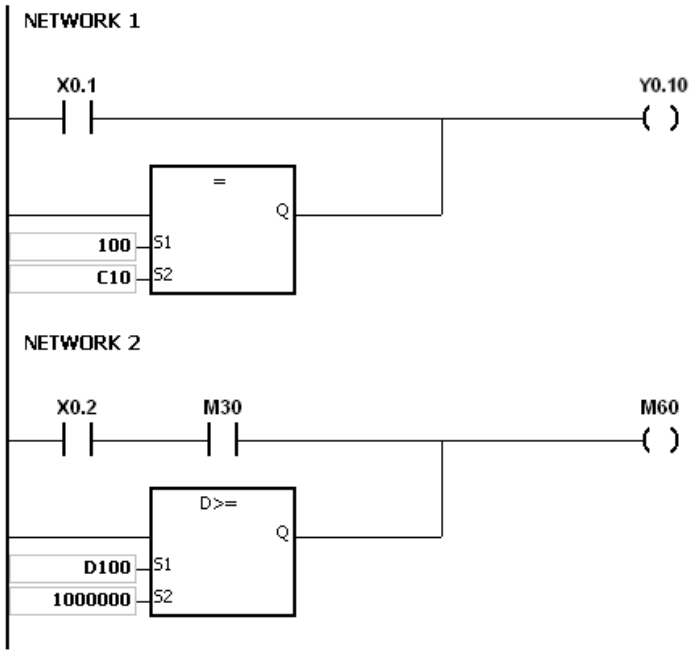
指令说明：

1. S_1 与 S_2 的内容做比较的指令。以“OR=”做为例子，比较结果为“等于”时，该指令导通；“不等于”时，该指令不导通。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

API No.	16 位指令	32 位指令	导通条件	非导通条件
0012	OR =	**D**OR =	$S_1 = S_2$	$S_1 \neq S_2$
0013	OR < >	**D**OR < >	$S_1 \neq S_2$	$S_1 = S_2$
0014	OR >	**D**OR >	$S_1 > S_2$	$S_1 \leqq S_2$
0015	OR > =	**D**OR > =	$S_1 \geqq S_2$	$S_1 < S_2$
0016	OR <	**D**OR <	$S_1 < S_2$	$S_1 \geqq S_2$
0017	OR < =	**D**OR < =	$S_1 \leqq S_2$	$S_1 > S_2$

程序范例：

1. 当 X0.1=ON 时，或者是 C10 的当前值等于 100 时，Y0.10=ON。
2. 当 X0.2 及 M30 都等于 ON 时，或者是 32 位寄存器（D101，D100）的内容大于或等于 1,000,000 时，M60=ON。



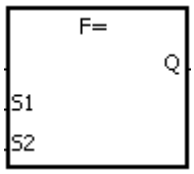
API	指令码	操作数	功能
0018~0023	FLD※	S_1 · S_2	浮点数接点类型比较 LD※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○					○
S_2	●	●			●	●	●	●	●		○					○

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



以 FLD=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

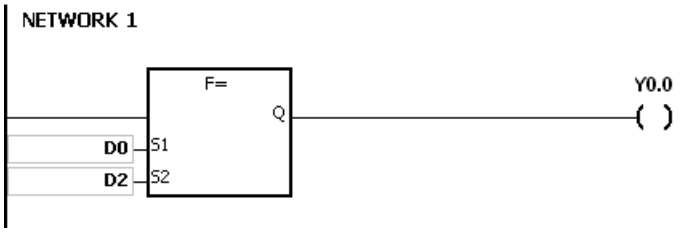
指令说明：

1. 此指令为 32 位单精度浮点数接点类型比较指令。
2. S_1 与 S_2 的内容以浮点数类型做比较的指令。以"FLD="做为例子，比较结果为"等于"时，该指令导通；"不等于"时，该指令不导通。使用 "FLD=" 指令时，请参考补充说明第 2 点说明。

API No.	32 位指令	导通条件	非导通条件
0018	FLD =	$S_1 = S_2$	$S_1 \neq S_2$
0019	FLD < >	$S_1 \neq S_2$	$S_1 = S_2$
0020	FLD >	$S_1 > S_2$	$S_1 \leqq S_2$
0021	FLD > =	$S_1 \geqq S_2$	$S_1 < S_2$
0022	FLD <	$S_1 < S_2$	$S_1 \geqq S_2$
0023	FLD < =	$S_1 \leqq S_2$	$S_1 > S_2$

程序范例：

以单精度浮点数指令为例，D0 的值等于 D2 的值时，Y0.0 ON。



补充说明：

1. 当 **S_1** 或 **S_2** 的内容值超出浮点数可以表示的范围时，接点不导通，SM0=ON，错误码 SR0=16#2013。
2. 当 S1 或 S2 的其中一个浮点数是由外部通讯输入，或者由编辑软件转换后输入时，则非常有机率发生极微小的数值误差，进而造成 "FLD=" 指令，其相等比较的结果无法成立，建议请改用 "FLD<" 与 "FAND>" 指令，或 "FLD>" 与 "FAND<" 指令组合搭配判断。

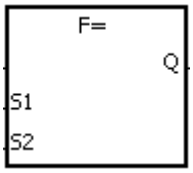
API	指令码	操作数	功能
0024~0029	FAND※	S_1，S_2	浮点数接点类型比较 AND※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○					○
S_2	●	●			●	●	●	●	●		○					○

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



以 FAND=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

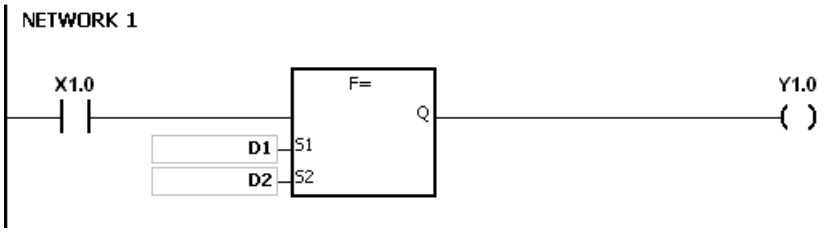
指令说明：

1. 此指令为 32 位单精度浮点数接点类型比较指令。
2. S_1 与 S_2 的内容以浮点数类型做比较的指令。以"FAND="做为例子，比较结果为"等于"时，该指令导通；"不等于"时，该指令不导通。

API No.	32 位指令	导通条件	非导通条件
0024	FAND =	$S_1 = S_2$	$S_1 \neq S_2$
0025	FAND < >	$S_1 \neq S_2$	$S_1 = S_2$
0026	FAND >	$S_1 > S_2$	$S_1 \leqq S_2$
0027	FAND > =	$S_1 \geqq S_2$	$S_1 < S_2$
0028	FAND <	$S_1 < S_2$	$S_1 \geqq S_2$
0029	FAND < =	$S_1 \leqq S_2$	$S_1 > S_2$

程序范例：

以单精度浮点数指令为例，当 X1.0=ON 且 D1 的值等于 D2 的值时，Y1.0 ON。



补充说明：

1. 当 S_1 或 S_2 的内容值超出浮点数可以表示的范围时，接点不导通，SM0=ON，错误码 SR0=16#2013。

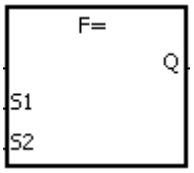
API	指令码	操作数	功能
0030~ 0035	FOR※	**S_1，S_2**	浮点数接点类型比较 OR※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○					○
S_2	●	●			●	●	●	●	●		○					○

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



以 FOR=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

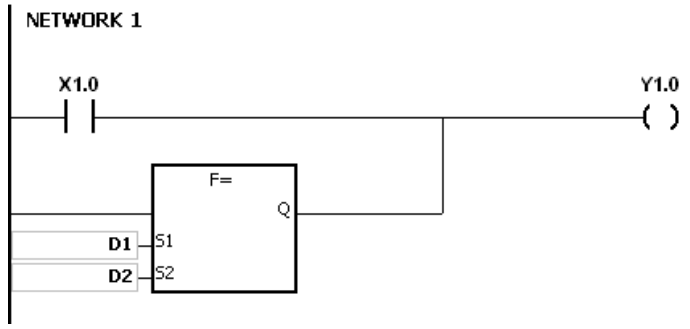
指令说明：

1. 此指令为 32 位单精度浮点数接点类型比较指令。
2. **S_1** 与 **S_2** 的内容以浮点数类型做比较的指令。以"FOR="做为例子，比较结果为"等于"时，该指令导通；"不等于"时，该指令不导通。

API No.	32 位指令	导通条件	非导通条件
0030	FOR =	**$S_1 = S_2$**	**$S_1 \neq S_2$**
0031	FOR < >	**$S_1 \neq S_2$**	**$S_1 = S_2$**
0032	FOR >	**$S_1 > S_2$**	**$S_1 \leqq S_2$**
0033	FOR > =	**$S_1 \geqq S_2$**	**$S_1 < S_2$**
0034	FOR <	**$S_1 < S_2$**	**$S_1 \geqq S_2$**
0035	FOR < =	**$S_1 \leqq S_2$**	**$S_1 > S_2$**

程序范例：

X1.0=ON 或者 D1 的值等于 D2 时，Y1.0=ON。



补充说明：

1. 当 **S_1** 或 **S_2** 的内容值超出浮点数可以表示的范围时，接点不导通，SM0=ON，错误码 SR0=16#2013。

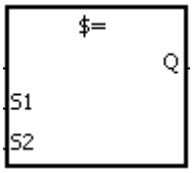
API	指令码	操作数	功能
0036~0037	LD$※	S_1，S_2	字符串类型比较 LD$※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●						○	

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													●
S_2													●

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



以 LD$=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

指令说明：

1. S_1 与 S_2 的内容以字符串类型做比较的指令。
2. S_1 与 S_2 的字符串内容限制最多 256 个字（包含结束符号 16#00）。
3. 以“LD$=”做为例子，比较结果为“等于”时，该接点导通；“不等于”时，该接点不导通。

API No.	指令	导通条件	非导通条件
0036	LD$=	$S_1 = S_2$	$S_1 \neq S_2$
0037	LD$< >	$S_1 \neq S_2$	$S_1 = S_2$

4. **S**~**S**+n（这里的 n 表示第 n 个装置，最大比对限制为 256 个字）中有包含 16#00 时，才能判断为一个完整字符串，例如：

	b15 ~ b8	b7 ~ b0
S	16#32(2)	16#31(1)
S+1	16#34(4)	16#33(3)
S+2	16#00	16#35(5)

"12345"

5. 字符串完全相同时，如下范例 S_1 与 S_2 的字符串都是“ABCDE”，各指令执行结果如下表所列：

	b15 ~ b8	b7 ~ b0
S1	16#42(B)	16#41(A)
S1+1	16#44(D)	16#43(C)
S1+2	16#00	16#45(E)

"ABCDE"

比较符号 [ ]

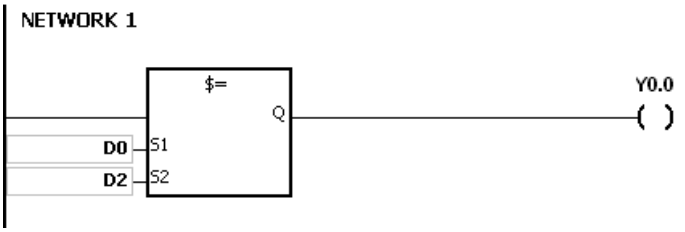
	b15 ~ b8	b7 ~ b0
S2	16#42(B)	16#41(A)
S2+1	16#44(D)	16#43(C)
S2+2	16#00	16#45(E)

"ABCDE"

比较符号	比较结果
$=	导通
$< >	不导通

程序范例：

D0~16#00 的字符串等于 D2~16#00 的字符串时，Y0.0 ON。



补充说明：

1. 若字符串长度超过 256 个字还没有出现结束符号（16#00），则指令不执行，SM0=ON，错误码 SR0=16#200E。
2. 字符串比对过程中，若其中一个字符串已经到达结束符号（16#00），即表示比对长度已到达。

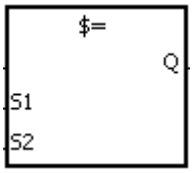
API	指令码	操作数	功能
0042~ 0043	AND$※	**S_1 · S_2**	字符串类型比较 AND$※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●						○	

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													●
S_2													●

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



以 AND$=为例

S_1 ： 数据源 1

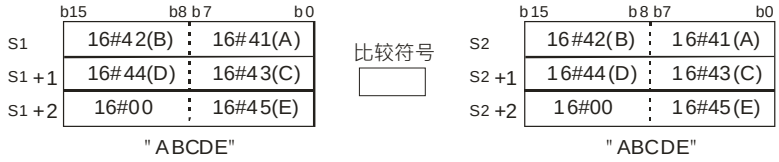
S_2 ： 数据源 2

指令说明：

1. **S_1** 与 **S_2** 的内容以字符串类型做比较的指令。
2. **S_1** 与 **S_2** 的字符串内容限制最多 256 个字（包含结束符号 16#00）。
3. 以“AND$=”做为例子，比较结果为“等于”时，该接点导通，“不等于”时，该接点不导通。

API No.	指令	导通条件	非导通条件
0042	AND$ =	**$S_1 = S_2$**	**$S_1 \neq S_2$**
0043	AND$ < >	**$S_1 \neq S_2$**	**$S_1 = S_2$**

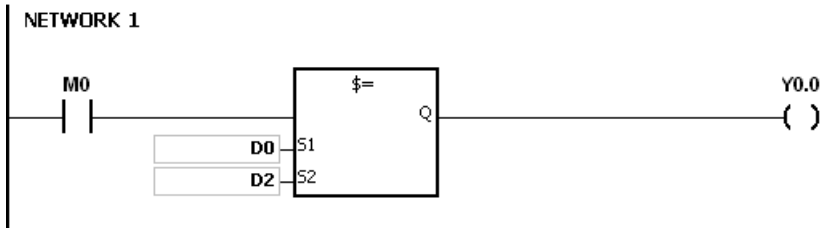
4. **S**~**S**+n（这里的 n 表示第 n 个装置，最大比对限制为 256 个字）中有包含 16#00 时，才能判断为一个完整字符串。
5. 字符串完全相同时，如下范例 **S_1** 与 **S_2** 的字符串都是“ABCDE”，各指令执行结果如下表所列：



比较符号	比较结果
$ =	导通
$ < >	不导通

程序范例：

当 M0=ON 且 D0~16#00 的字符串等于 D2~16#00 的字符串，则 Y0.0 ON。



补充说明：

1. 若字符串长度超过 256 个字还没有出现结束符号（16#00），则指令不执行，SM0=ON，错误码 SR0=16#200E。
2. 字符串比对过程中，若其中一个字符串已经到达结束符号（16#00），即表示比对长度已到达。

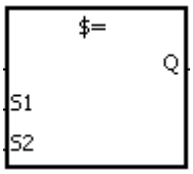
API	指令码	操作数	功能
0048~ 0049	OR$※	**S_1 · S_2**	字符串类型比较 OR$※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●						○	

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													●
S_2													●

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



以 OR$=为例

S_1 ： 数据源 1

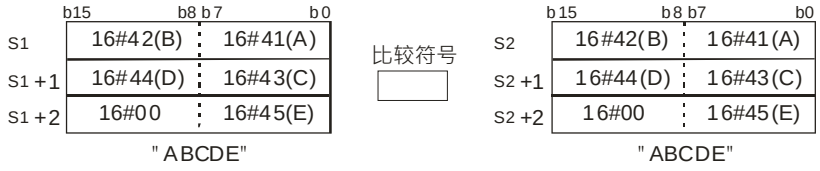
S_2 ： 数据源 2

指令说明：

1. **S_1** 与 **S_2** 的内容以字符串类型做比较的指令。
2. **S_1** 与 **S_2** 的字符串内容限制最多 256 个字（包含结束符号 16#00）。
3. 以“OR$=”做为例子，比较结果为“等于”时，该接点导通；“不等于”时，该接点不导通。

API No.	指令	导通条件	非导通条件
0048	OR$ =	**$S_1 = S_2$**	**$S_1 \neq S_2$**
0049	OR$ < >	**$S_1 \neq S_2$**	**$S_1 = S_2$**

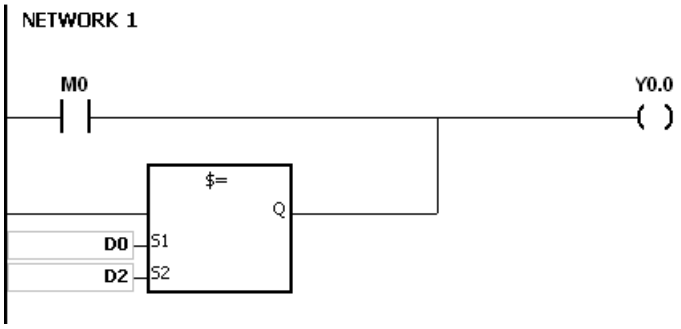
4. **S**~**S**+n（这里的 n 表示第 n 个装置，最大比对限制为 256 个字）中有包含 16#00 时，才能判断为一个完整字符串。
5. 字符串完全相同时，如下范例 **S_1** 与 **S_2** 的字符串都是“ABCDE”，各指令执行结果如下表所列：



比较符号	比较结果
$ =	导通
$ < >	不导通

程序范例：

当 M0=ON 或者是 D0~16#00 的字符串等于 D2~16#00 的字符串时，Y0.0 ON。



补充说明：

1. 若字符串长度超过 256 个字还没有出现结束符号（16#00），则指令不执行，SM0=ON，错误码 SR0=16#200E。
2. 字符串比对过程中，若其中一个字符串已经到达结束符号（16#00），即表示比对长度已到达。

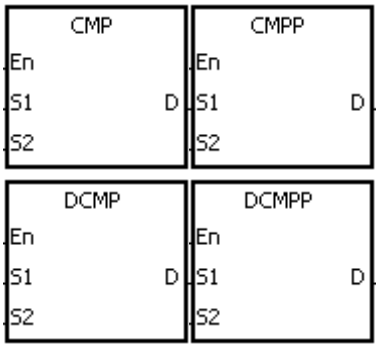
API	指令码			操作数	功能
0054	D	CMP	P	**S_1**，**S_2**，**D**	比较设定输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S1 ： 比较值 1

S2 ： 比较值 2

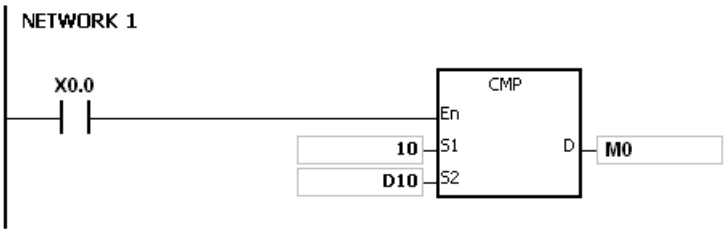
D ： 比较结果

指令说明：

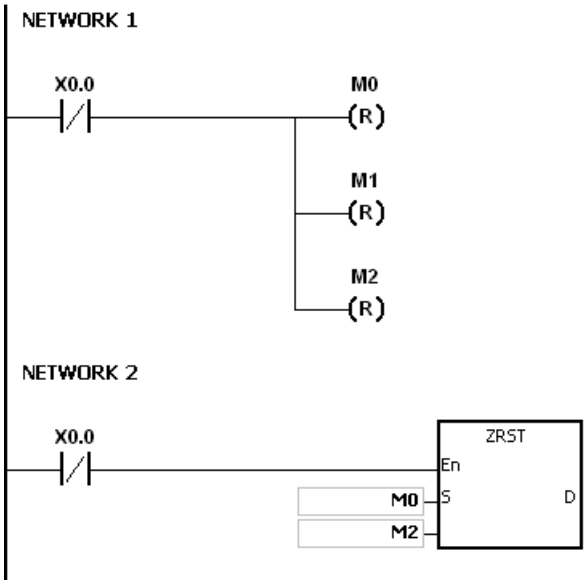
1. 将操作数 **S_1** 和 **S_2** 的内容以有号数十进制作大小比较，其比较结果在 **D** 作表示。
2. 操作数 **D** 占用 3 个连续装置，**D**，**D**+1，**D**+2 用于储存比较结果，如果 **S_1**>**S_2**，**D**=ON；如果 **S_1**=**S_2**，**D**+1=ON；如果 **S_1**<**S_2**，**D**+2=ON。
3. **D**CMP、**D**CMPP 指令才可使用 HC 装置，但不可使用 E 装置。

程序范例：

1. 如果操作数 D 设为 M0，那么比较结果会显示在 M0，M1，M2 中，如下所示。
2. 当 X0.0=ON，CMP 指令执行，M0，M1，M2 其中的某个装置会 ON。当 X0.0=OFF，CMP 指令停止执行并且 M0，M1，M2 保持它们的当前值。



3. 若要清除其比较结果请使用 RST 或 ZRST 指令。



补充说明：

1. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of BOOL。
2. **D**+2 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

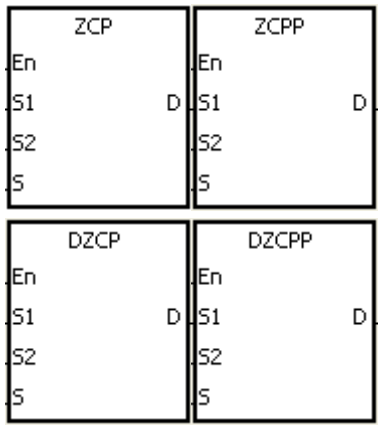
API	指令码			操作数	功能
0055	D	ZCP	P	**S_1、S_2、S、D**	区域比较

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
S	●	●			●	●	●	●	●		○	○	○	○		
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
S		●	●			●	●				●	●	
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S1 ： 区间比较的下限值

S2 ： 区间比较的上限值

S ： 比较值

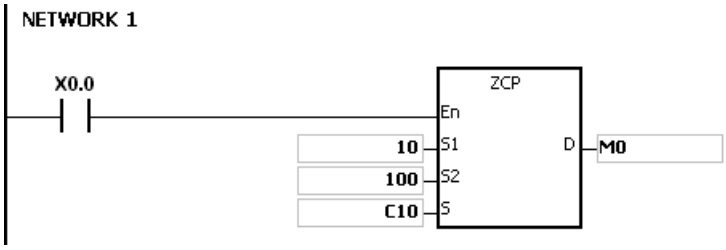
D ： 比较结果

指令说明：

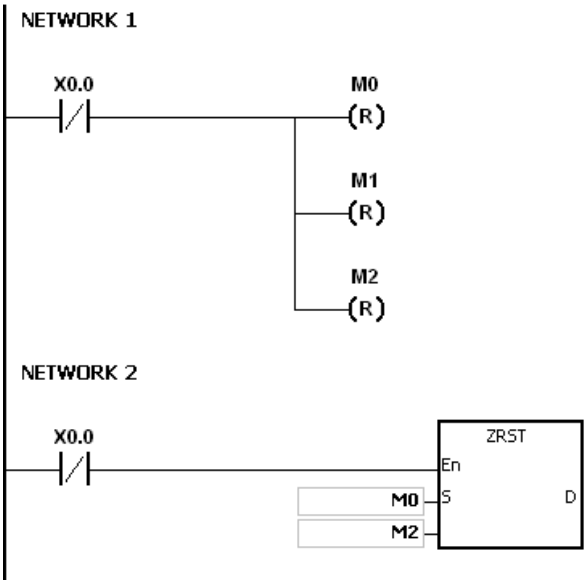
1. 比较值 **S** 与下限 **S_1** 及上限 **S_2** 以有号数十进制数值做比较，其比较结果在 **D** 作表示。
2. 操作数 **S_1** 必须比 **S_2** 小。当 **S_1>S_2**，指令执行时把 **S_1** 作为上/下限值进行比较。
3. 操作数 **D** 占用 3 个连续的装置，**D**，**D**+1，**D**+2 储存比较结果。如果 **S_1>S**，**D**=ON；如果 **S_1**≦**S**≦**S_2**，**D**+1=ON；如果 **S_2<S**，**D**+2=ON。
4. DZCP、**D**ZCPP 指令才可使用 HC 装置，但不可使用 E 装置。

程序范例：

1. 比较结果指定装置为 M0，则自动占有 M0，M1 及 M2。
2. 当 X0.0=ON 时，ZCP 指令执行，M0，M1 及 M2 其中之一会 ON，当 X0.0=OFF 时，ZCP 指令不执行，M0，M1 及 M2 状态保持在 X0.0=OFF 之前的状态。



3. 若要清除其结果请使用 RST 或 ZRST 指令。



补充说明：

1. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of BOOL。
2. **D**+2 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

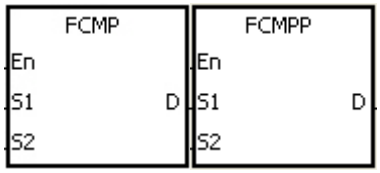
API	指令码			操作数	功能
0056		FCMP	P	**S_1，S_2，D**	浮点数比较

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○					○
S_2	●	●			●	●	●	●	●		○					○
D		●	●	●				●		○						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D	●												

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S_1 ： 浮点数比较值 1

S_2 ： 浮点数比较值 2

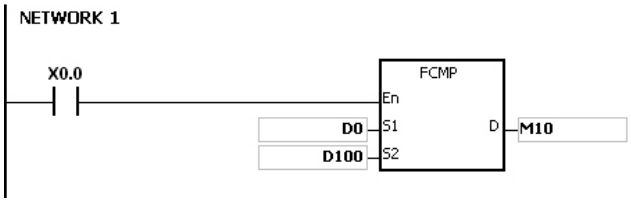
D ： 比较结果

指令说明：

1. 浮点数比较值 **S_1** 与浮点数比较值 **S_2** 进行比较，比较的结果（ > ，= ，< ）在 **D** 作表示。
2. 操作数 **D** 占用 3 个连续装置，**D**，**D**+1，**D**+2 用于储存比较结果，如果 **S_1**>**S_2**，**D**=ON；如果 **S_1**=**S_2**，**D**+1=ON；如果 **S_1**<**S_2**，**D**+2=ON。

程序范例：

1. 若比较结果指定装置为 M10 则自动占有 M10~M12。
2. 当 X0.0=ON 时，FCMP 指令执行，M10~M12 其中之一会 ON，当 X0.0=OFF，FCMP 指令不执行，M10~M12 状态保持在 X0.0=OFF 之前的状态。
3. 若需得到≧，≦，≠的结果时，可将 M10~M12 串并联即可取得。
4. 若要清除其结果请使用 RST 或 ZRST 指令。



补充说明：

1. 当 **S_1** 或 **S_2** 的内容值超出浮点数可以表示的范围时，接点不导通，SM0=ON，错误码 SR0=16#2013。
2. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of BOOL。
3. **D**+2 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

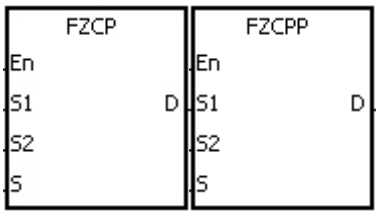
API	指令码			操作数	功能
0057		FZCP	P	**S_1 · S_2 · S · D**	浮点数区域比较

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○					○
S_2	●	●			●	●	●	●	●		○					○
S	●	●			●	●	●	●	●		○					○
D		●	●	●				●		○						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
S									●				
D	●												

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S_1 ：浮点数区域比较

S_2 ：区域比较之浮点数上限值

S ：浮点数比较值

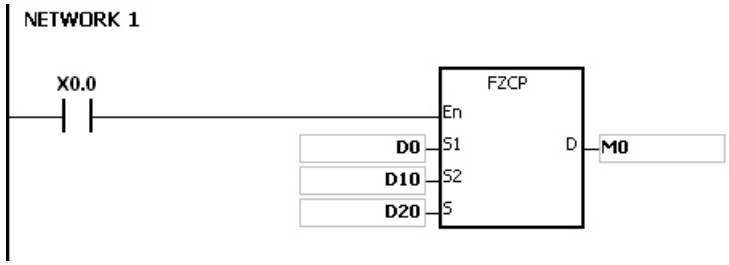
D ：比较结果

指令说明：

1. 浮点数比较值 **S** 与浮点数下限值 **S_1** 及浮点数上限值 **S_2** 做比较，其比较结果在 **D** 做表示。
2. 操作数 **S_1** 必须比 **S_2** 小。当 **S_1>S_2**，指令执行时把 **S_1** 作为上/下限值进行比较。
3. 操作数 **D** 占用 3 个连续的装置，**D**，**D**+1，**D**+2 储存比较结果。如果 **S_1>S**，**D**=ON；如果 **S_1**≦**S**≦**S_2**，**D**+1=ON；如果 **S_2<S**，**D**+2=ON。

程序范例：

1. 若比较结果指定装置为 M0，则自动占有 M0~M2。
2. 当 X0.0=ON 时，FZCP 指令执行，M0~M2 其中之一会 ON。当 X0.0=OFF，FZCP 指令不执行，M0~M2 状态保持在 X0.0=OFF 之前的状态。
3. 若要清除其结果请使用 RST 或 ZRST 指令。



补充说明：

1. 当 **S_1** 或 **S_2** 或 **S** 的内容值超出浮点数可以表示的范围时，接点不导通，SM0=ON，错误码 SR0=16#2013。
2. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of BOOL。
3. **D**+2 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

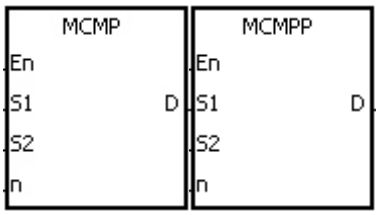
API	指令码		操作数	功能
0058	MCMP	P	**S_1 · S_2 · n · D**	矩阵比较

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●		○					
S_2	●	●			●	●		●	●		○					
n	●	●			●	●		●	●		○		○	○		
D		●			●	●		●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 矩阵来源装置 1

S_2 ： 矩阵来源装置 2

n ： 数组长度

D ： 指针 Pr 用以存放目标的地址值

指令说明：

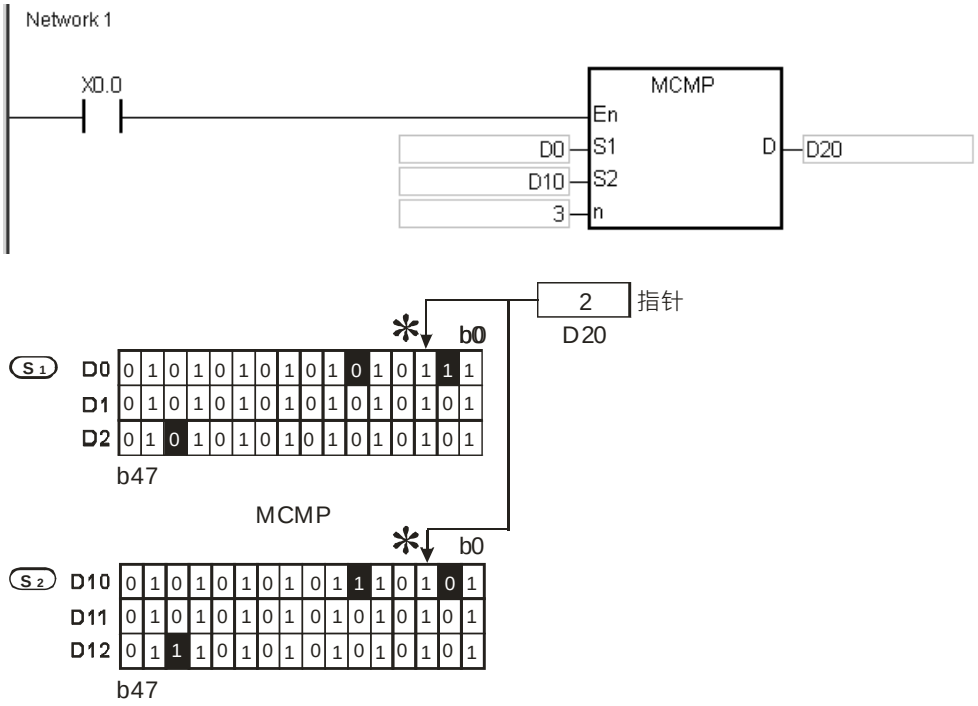
1. 每次比较依两个矩阵来源指针 Pr 地址。将 **S_1** 或 **S_2** 两个矩阵中的每一个 bit 从地址 **D**+1 开始做比较，找出值不同的地址，再将此地址值存到 **D** 中，完成此次比较。
2. n 操作数的范围为 1~256。
3. 由矩阵比较标志 SM607 决定比较相同值（SM607=1）或不同值（SM607=0），当比较到达时立即停止比较动作，矩阵位寻找标志 SM610=1。当比较到最后一个 bit 时，矩阵搜寻结束标志 SM608=ON 比较到达的编号存于 **D** 中，下一次扫描周期时，再由第 0 个 bit 开始比较，同时矩阵搜寻起始标志 SM609=1。当 **D** 的值超过范围时指针错误标志 SM611=1。
4. 在矩阵指令运作中，通常需要有一个 16 位寄存器来指定矩阵中 16n 个单点的某个单点当作运算对象。此寄存器称为矩阵的指针 Pr（Pointer），由用户于指令中指定，其有效范围为 0~16n-1，分别对应至矩阵中的位 b0~b16n-1。在运作中应避免更动到 Pr 值，以免影响其正确的比较找寻，若 Pr 值超出此范围则矩阵指针错误标志 SM611 设为 1，且本指令不执行。
5. 若矩阵搜寻结束标志 SM608 与矩阵位寻找标志 SM610 同时发生则会同时=1。

程序范例：

1. 当 X0.0 由 OFF→ON，矩阵搜寻起始标志 SM609=0，故由指针当时值加 1 的 bit 地址（标注 * 处）开始往下比较找寻位状态不同（SM607=0 为找不同）者。

2. 设指针当时值 D20=2，当 X0.0 由 OFF→ON 时动作 4 次，可得到如（❶，❷，❸，❹）四个执行结果。

- D20=5，矩阵位寻找标志 SM610=1，矩阵搜寻结束标志 SM608=0。
- D20=45，矩阵位寻找标志 SM610=1，矩阵搜寻结束标志 SM608=0。
- D20=47，矩阵位寻找标志 SM610=0，矩阵搜寻结束标志 SM608=1。
- D20=1，矩阵位寻找标志 SM610=1，矩阵搜寻结束标志 SM608=0。



补充说明：

1. 运算错误码说明：

   当 **S_1+n**-1 或 **S_2+n**-1 的装置超过范围时，该指令不执行，SM0=ON，错误码 SR0=16#2003。

   若 **n** 操作数的值不在 1~256 之间，该指令不执行，且 SM0=ON，错误码 SR0=16#200B。

2. 标志信号说明：

   SM607： 矩阵比较标志，比较相同值 SM607=1，比较不同值 SM607=0

   SM608： 矩阵搜寻结束标志，当比较到最后一个 bit 时，SM608=1

   SM609： 矩阵搜寻起始标志，由第一个 bit 开始比较，SM609=1

   SM610： 矩阵位寻找标志，比较到达时立即停止比较动作 SM610=1

   SM611： 矩阵指针错误标志，指针 Pr 值超出此范围则 SM611=1

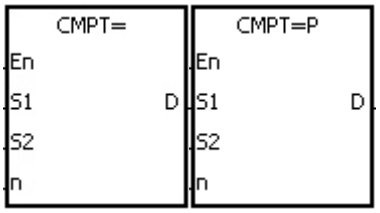
API	指令码			操作数	功能
0059~0064		CMPT※	P	**S_1 · S_2 · n · D**	表格比较指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●		○		○	○		
S_2	●	●			●	●		●	●		○					
n	●	●			●	●		●	●		○	○	○	○		
D		●	●	●				●		●						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
n		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 来源起始装置 1

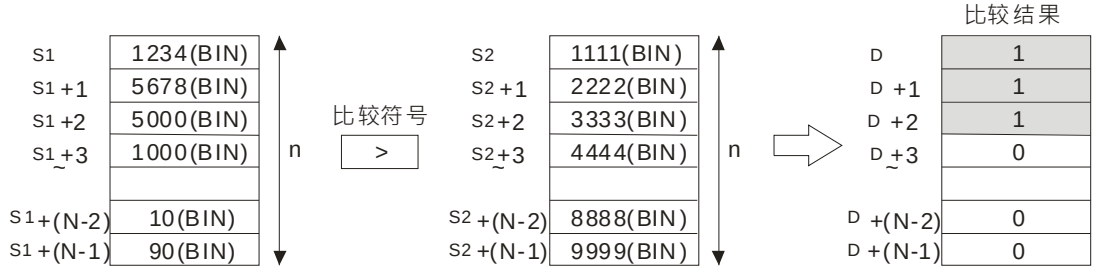
S_2 ： 来源起始装置 2

n ： 比较数据长度

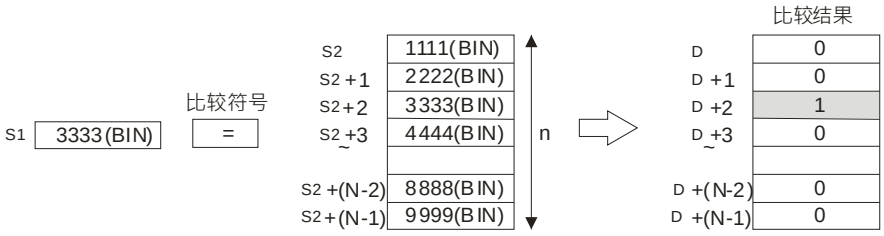
D ： 目标装置

指令说明：

1. **S_1** 与 **S_2** 的内容以有号数十进制做 **n** 笔比较，将结果存放于 **D**。
2. **n** 操作数的范围为 1~256。
3. **D** 操作数写入值都以 1 位写入，未被对应之 bit 值不变。
4. 表格比较指令 CMPT#全部输出标志，若比较结果为全都输出（全都 ON），则 SM620=ON，反之 SM620=OFF。
5. 如果 **S_1** 的来源指定为装置，则比较方式如下：



6. 如果 **S_1** 的来源直接指定为-32768~32767，则比较方式如下：

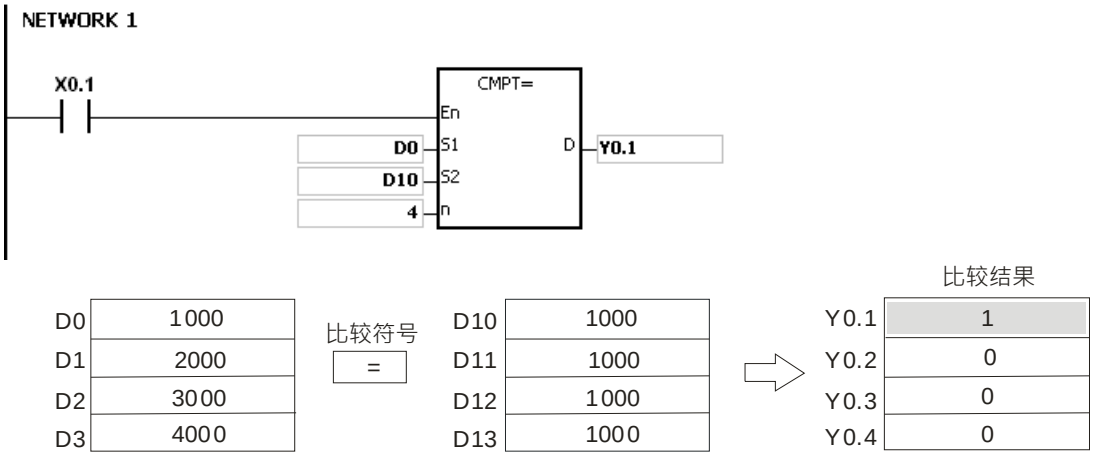


7. 各指令比较结果如下：

API No.	16 位指令	比较结果	
		1	0
0059	CMPT =	$S_1 = S_2$	$S_1 \neq S_2$
0060	CMPT < >	$S_1 \neq S_2$	$S_1 = S_2$
0061	CMPT >	$S_1 > S_2$	$S_1 \leqq S_2$
0062	CMPT > =	$S_1 \geqq S_2$	$S_1 < S_2$
0063	CMPT <	$S_1 < S_2$	$S_1 \geqq S_2$
0064	CMPT < =	$S_1 \leqq S_2$	$S_1 > S_2$

程序范例：

D0~D3 与 D10~D13 数据比对，若比较结果相同，对应的 Y0.1~Y0.4=ON。



补充说明：

1. 若 n 不在 1~256 之间，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. **S_1~S_1+n** 与 **S_2~S_2+n** 与 **D** 操作数的装置如果不足，则指令不执行，SM0=ON，错误码 SR0=16#2003。

API	指令码	操作数	功能
0065	CHKADR	**S**，**n**，**D**	接点类型指针寄存器地址检查

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S																
n	●	●			●	●		●	●		○	○	○	○		
D		●	●	●				●		○						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													
n		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：

S ： 指针寄存器

n ： 装置个数

D ： 检查结果

6

指令说明：

1. 检查 **S** 指针寄存器的内容值~（**S** 的内容值+**n**-1）是否超过 **S** 指针寄存器所指定的装置类别范围，检查结果没有超出装置范围时，**D** 装置为 ON，否则 **D** 装置为 OFF。
2. **S** 支持 D、T、C、HC 的指针寄存器（POINTER/T_POINTER/C_POINTER/HC_POINTER）。
3. **n** 的范围 1~1024。
4. 本指令只可以在功能块中使用。通常使用于初期开发程序，并且无法掌握装置范围是否会超出范围时，当程序已完成时，即可删除此指令。

程序范例：

1. 使用 ISPSoft 建立一个 Program（Prog0），及一个 Function Block（FB0）。

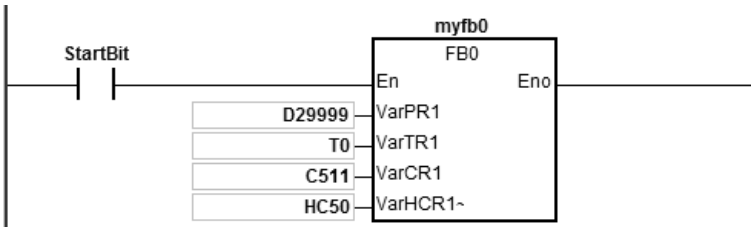
并且在 Program 中宣告两个变量如下：

Local Symbols				
Declaration Type	Identifiers	Address	Type...	Initial Value
VAR	myfb0	N/A [Auto]	FB0	N/A
VAR	StartBit	N/A [Auto]	BOOL	FALSE

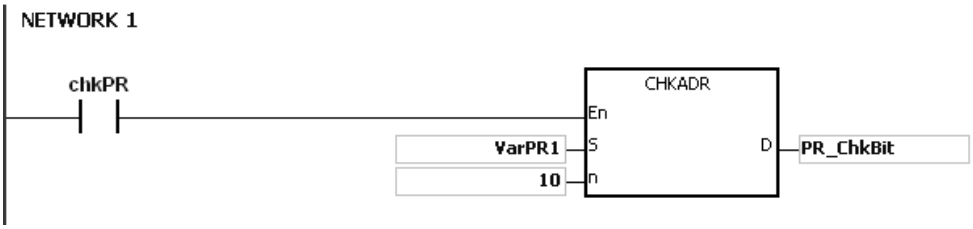
2. 在已建立的 Function Block 中宣告 VarPR1、VarTR1、VarCR1 以及 VarHCR1，Type 分别指定为 POINTER、T_POINTER、C_POINTER、HC_POINTER。

Local Symbols				
Declaration Type	Identifiers	Address	Type...	Initial Value
VAR_IN_OUT	VarPR1	N/A [Auto]	POINTER	N/A
VAR_IN_OUT	VarTR1	N/A [Auto]	T_POINTER	N/A
VAR_IN_OUT	VarCR1	N/A [Auto]	C_POINTER	N/A
VAR_IN_OUT	VarHCR1	N/A [Auto]	HC_POINTER	N/A
VAR	PR_ChkBit	N/A [Auto]	BOOL	FALSE
VAR	TR_ChkBit	N/A [Auto]	BOOL	FALSE
VAR	CR_ChkBit	N/A [Auto]	BOOL	FALSE
VAR	HCR_ChkBit	N/A [Auto]	BOOL	FALSE
VAR	chkPR	N/A [Auto]	BOOL	N/A
VAR	chkTR	N/A [Auto]	BOOL	N/A
VAR	chkCR	N/A [Auto]	BOOL	N/A
VAR	chkHCR	N/A [Auto]	BOOL	N/A

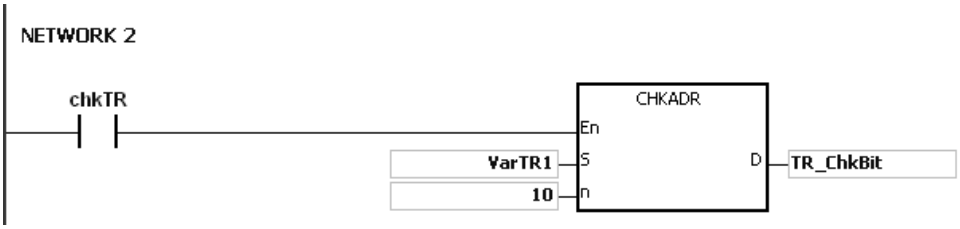
3. 在 Program 中调用已建立好的 Function Block FB0，并且将 D29999、T0、C511、HC50 指定给 FB0 中的 VarPR1、VarTR1、VarCR1、VarHCR1。



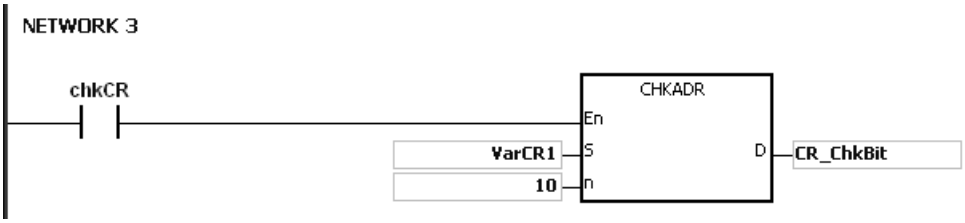
4. 在 FB0 中使用 CHKADR 来检查 VarPR1、VarTR1、VarCR1、VarHCR1 是否超出范围。
5. 当 chkPR=ON，VarPR1 所代表的实际装置为 D29999，而 D 装置的合法范围为 D0~D29999，所以 D29999+10-1=D30008 已超出范围，则 PR_ChkBit=OFF。



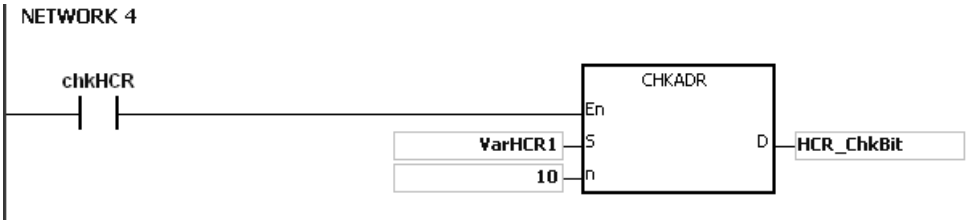
6. 当 chkTR=ON，VarTR1 所代表的实际装置为 T0，而 T 装置的合法范围为 T0~T511，所以 T0+10-1=T9 没有超出范围，则 TR_ChkBit=ON。



7. 当 chkCR=ON，VarCR1 所代表的实际装置为 C511，而 C 装置的合法范围为 C0~C511，所以 C511+10-1=C520 已超出范围，则 CR_ChkBit=OFF。



8. 当 chkHCR=ON，VarHCR1 所代表的实际装置为 HC50，而 HC 装置的合法范围为 HC0~HC255，所以 HC50+10-1=HC59 没有超出范围，则 HCR_ChkBit=ON。



补充说明：

1. **S** 的内容值（实际装置地址）如果一开始就不合法，超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 若 **n** 不在 1~1024 之间，指令不执行，SM0=ON，错误码 SR0=16#200B。

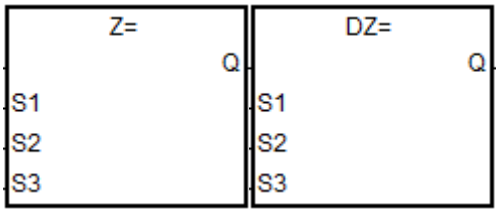
API	指令码			操作数	功能
0066~0071	D	LDZ※		S_1 · S_2 · S_3	接点类型绝对值比较 LDZ※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
S_3	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
S_3		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 LDZ= 跟 DLDZ= 为例

S_1 ： 数据源 1

S_2 ： 数据源 2

S_3 ： 数据源之比对值

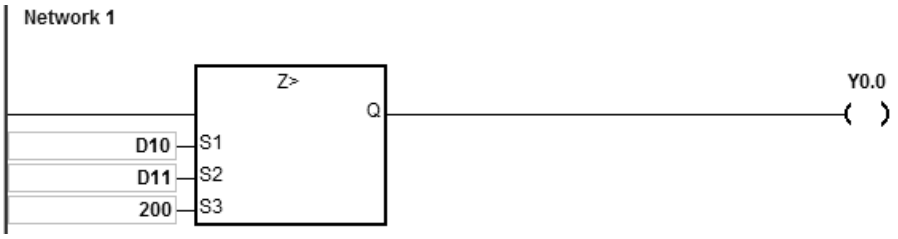
指令说明：

1. S_1 与 S_2 的内容相减之后取绝对值，并与 S_3 绝对值比较的指令。以"LDZ="做为例子，比较结果为"等于"时，该指令导通；"不等于"时，该指令不导通。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

API No.	16 位指令	32 位指令	导通条件	非导通条件
0066	LDZ =	**D**LDZ =	$\|S_1-S_2\| = \|S_3\|$	$\|S_1-S_2\| \neq \|S_3\|$
0067	LDZ < >	**D**LDZ < >	$\|S_1-S_2\| \neq \|S_3\|$	$\|S_1-S_2\| = \|S_3\|$
0068	LDZ >	**D**LDZ >	$\|S_1-S_2\| > \|S_3\|$	$\|S_1-S_2\| \leqq \|S_3\|$
0069	LDZ > =	**D**LDZ > =	$\|S_1-S_2\| \geqq \|S_3\|$	$\|S_1-S_2\| < \|S_3\|$
0070	LDZ <	**D**LDZ <	$\|S_1-S_2\| < \|S_3\|$	$\|S_1-S_2\| \geqq \|S_3\|$
0071	LDZ < =	**D**LDZ < =	$\|S_1-S_2\| \leqq \|S_3\|$	$\|S_1-S_2\| > \|S_3\|$

程序范例：

D10 减去 D11 的绝对值内容大于 200 时，Y0.0=ON。若数值小于等于 200 时，Y0.0 = OFF。



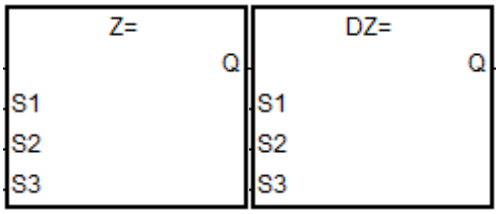
API	指令码			操作数	功能
0072~0077	D	ANDZ※		S_1，S_2，S_3	接点类型绝对值比较 ANDZ※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
S_3	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
S_3		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 ANDZ=跟 DANDZ=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

S_3 ： 数据源之比对值

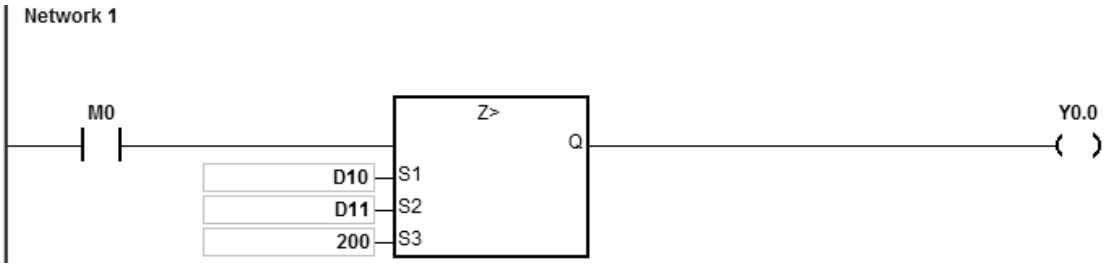
指令说明：

1. S_1 与 S_2 的内容相减之后取绝对值，并与 S_3 绝对值比较的指令。以“ANDZ=”做为例子，比较结果为“等于”时，该指令导通；“不等于”时，该指令不导通。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

API No.	16 位指令	32 位指令	导通条件	非导通条件
0072	ANDZ =	**D**ANDZ =	$\|S_1-S_2\|=\|S_3\|$	$\|S_1-S_2\|\neq\|S_3\|$
0073	ANDZ < >	**D**ANDZ < >	$\|S_1-S_2\|\neq\|S_3\|$	$\|S_1-S_2\|=\|S_3\|$
0074	ANDZ >	**D**ANDZ >	$\|S_1-S_2\|>\|S_3\|$	$\|S_1-S_2\|\leqq\|S_3\|$
0075	ANDZ > =	**D**ANDZ > =	$\|S_1-S_2\|\geqq\|S_3\|$	$\|S_1-S_2\|<\|S_3\|$
0076	ANDZ <	**D**ANDZ <	$\|S_1-S_2\|<\|S_3\|$	$\|S_1-S_2\|\geqq\|S_3\|$
0077	ANDZ < =	**D**ANDZ < =	$\|S_1-S_2\|\leqq\|S_3\|$	$\|S_1-S_2\|>\|S_3\|$

程序范例：

M0 为 ON 时，并且 D10 减去 D11 的绝对值内容大于 200 时，Y0.0=ON；若数值小于等于 200 时，Y0.0 = OFF。



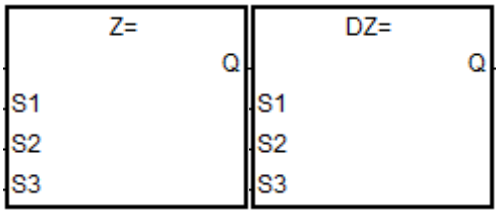
API	指令码			操作数	功能
0078~0083	D	ORZ※		S_1，S_2，S_3	接点类型绝对值比较 ORZ※

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
S_3	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
S_3		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 ORZ=跟 DORZ=为例

S_1 ： 数据源 1

S_2 ： 数据源 2

S_3 ： 数据源之比对值

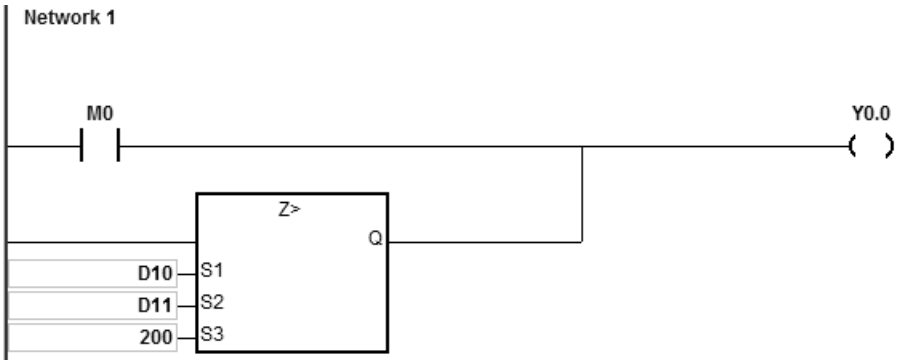
指令说明：

1. S_1 与 S_2 的内容相减之后取绝对值，并与 S_3 绝对值比较的指令。以"ORZ="做为例子，比较结果为"等于"时，该指令导通；"不等于"时，该指令不导通。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

API No.	16 位指令	32 位指令	导通条件	非导通条件
0078	ORZ =	**D**ORZ =	$\|S_1-S_2\| = \|S_3\|$	$\|S_1-S_2\| \neq \|S_3\|$
0079	ORZ < >	**D**ORZ < >	$\|S_1-S_2\| \neq \|S_3\|$	$\|S_1-S_2\| = \|S_3\|$
0080	ORZ >	**D**ORZ >	$\|S_1-S_2\| > \|S_3\|$	$\|S_1-S_2\| \leqq \|S_3\|$
0081	ORZ > =	**D**ORZ > =	$\|S_1-S_2\| \geqq \|S_3\|$	$\|S_1-S_2\| < \|S_3\|$
0082	ORZ <	**D**ORZ <	$\|S_1-S_2\| < \|S_3\|$	$\|S_1-S_2\| \geqq \|S_3\|$
0083	ORZ < =	**D**ORZ < =	$\|S_1-S_2\| \leqq \|S_3\|$	$\|S_1-S_2\| > \|S_3\|$

程序范例：

M0 为 ON 时，或者 D10 减去 D11 的绝对值内容大于 200 时，Y0.0=ON；若数值小于等于 200 时，Y0.0 = OFF。



6.2 四则运算指令

6.2.1 四则运算指令一览表

API	指令码（位）		P 指令	功能
	16	32		
0100	+	D+	✓	BIN 加法
0101	-	D-	✓	BIN 减法
0102	*	D*	✓	BIN 乘法
0103	/	D/	✓	BIN 除法
0104	–	F+	✓	浮点数加法
0105	–	F-	✓	浮点数减法
0106	–	F*	✓	浮点数乘法
0107	–	F/	✓	浮点数除法
0112	BK+	DBK+	✓	连续区块 BIN 加法
0113	BK-	DBK-	✓	连续区块 BIN 减法
0114	$+	–	✓	字符串链结
0115	INC	DINC	✓	BIN 加一
0116	DEC	DDEC	✓	BIN 减一
0117	MUL16	MUL32	✓	MUL16：16 位专用 BIN 乘法 MUL32：32 位专用 BIN 乘法
0118	DIV16	DIV32	✓	DIV16：16 位专用 BIN 除法 DIV32：32 位专用 BIN 除法

6.2.2 四则运算指令说明

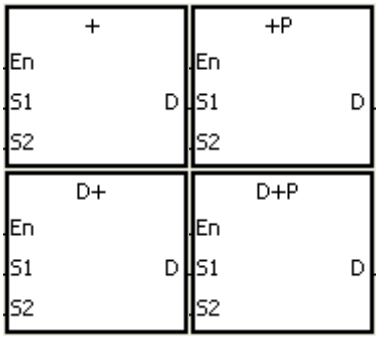
API	指令码			操作数	功能
0100	D	+	P	**S_1，S_2，D**	BIN 加法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 被加数

S_2 ： 加数

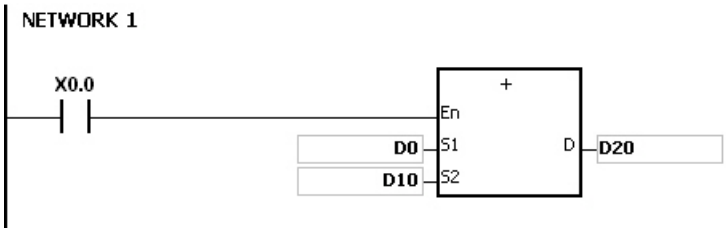
D ： 和

指令说明：

1. **S_1** 及 **S_2** 以 BIN 方式相加的结果存于 D。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 组件。
3. 标志：SM600 零标号（Zero flag），SM601 借位标号（Borrow flag），SM602 进位标号（Carry flag）。
4. 运算结果为 0 时，零标号（Zero flag）SM600 为 ON，否则为 OFF。
5. 16 位 BIN 加法：
   运算结果大于 16 位 BIN 可表示范围时，进位标号（Carry flag）SM602 为 ON，否则为 OFF。
6. 32 位 BIN 加法：
   运算结果大于 32 位 BIN 可表示范围时，进位标号（Carry flag）SM602 为 ON，否则为 OFF。

程序范例一：

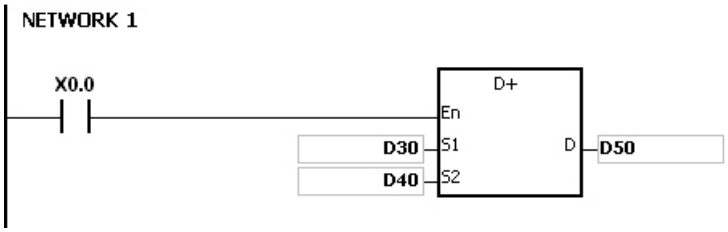
16 位 BIN 加法：当 X0.0=ON 时，被加数 D0 内容加上加数 D10 之内容将结果存在 D20 之内容当中。



- 当 D0=100，D10=10，D0+D10=110，D20=110。
- 当 D0=16#7FFF，D10=16#1，D0+D10=16#8000，D20=16#8000。
- 当 D0=16#FFFF，D10=16#1，D0+D10=16#10000，此时运算结果超出 16 位 BIN 可表示范围，则进位标号 SM602=ON，D20=16#0，因为运算结果为 16#0，所以零标号 SM600=ON。

程序范例二：

32 位 BIN 加法：当 X0.0=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）。



- 当（D31、D30）=11111111，（D41、D40）=44444444，（D31、D30）+ （D41、D40）=55555555，此时运算结果，（D51、D50）=55555555。
- 当（D31、D30）=16#80000000（D41、D40）=16#FFFFFFFF（D31、D30）+（D41、D40）=16#17FFFFFF，此时运算结果超出 32 位 BIN 可表示范围，则进位标号 SM602=ON ，（D51、D50）=16#7FFFFFFF。

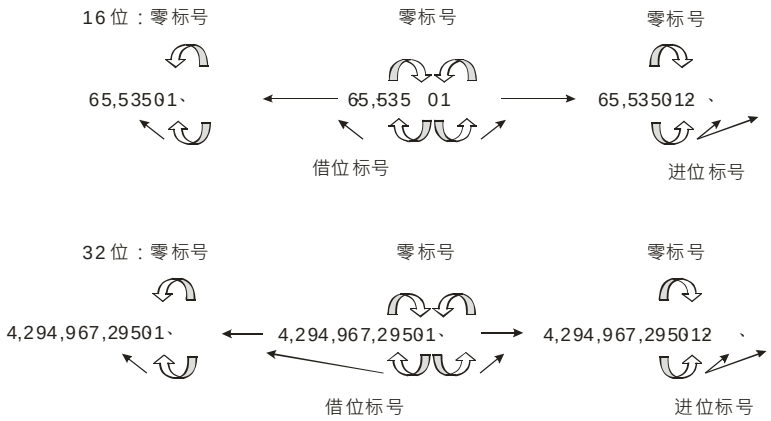
标志操作：

16 位指令：

1. 如果操作结果是“0”，零标志 SM600 被设定成 ON。
2. 如果操作结果超出 65,535，进位标志 SM602 被设定成 ON。

32 位指令：

1. 如果操作结果是“0”，零标志 SM600 被设定成 ON。
2. 如果操作结果超出 4,294,967,295，进位标志 SM602 被设定成 ON。



API	指令码			操作数	功能
0101	D	-	P	**S_1 · S_2 · D**	BIN 减法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：

S_1 ： 被减数

S_2 ： 减数

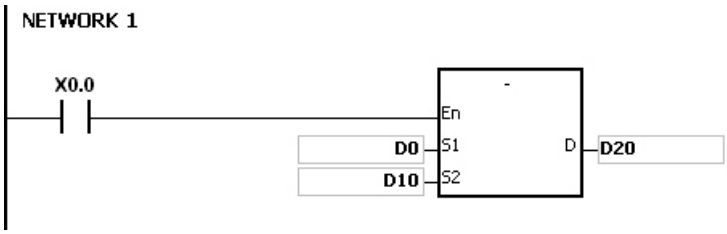
D ： 差

指令说明：

1. **S_1** 及 **S_2** 以 BIN 方式相减的结果存于 **D**。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 组件。
3. 标志：SM600 零标号（Zero flag），SM601 借位标号（Borrow flag），SM602 进位标号（Carry flag）。
4. 运算结果为 0 时，零标号（Zero flag）SM600 为 ON，否则为 OFF。
5. 运算发生借位时，借位标号（Borrow flag）SM601 为 ON，否则为 OFF。

程序范例一：

16 位 BIN 减法：当 X0.0=ON 时，将 D0 内容减掉 D10 内容将差存在 D20 之内容中，如下说明：

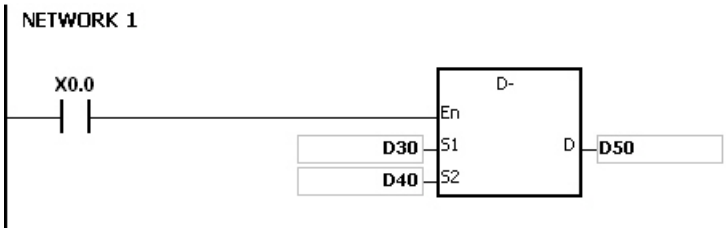


- 当 D0=100，D10=10，D0-D10=90，D20=90。
- 当 D0=16#8000，D10=16#1，D0-D10=16#7FFF，D20=16#7FFF。

- 当 D0=16#1，D10=16#2，D0-D10=16#FFFF，运算发生借位，借位标号 SM601=ON，D20=16#FFFF。
- 当 D0=16#0，D10=16#FFFF，D0-D10=16#F0001，运算发生借位，借位标号 SM601=ON，D20=16#1。

程序范例二：

32 位 BIN 减法：当 X0.0=ON 时，(D31、D30)内容减掉(D41、D40)之内容将差存在(D51、D50)之中。(其中 D30、D40、D50 为低 16 位数据，D31、D41、D51 为高 16 位数据)，如下说明：



- 当(D31、D30)=55555555，(D41、D40)=11111111，(D31、D30)-(D41、D40)=44444444，(D51、D50)=44444444。
- 当(D31、D30)=16#80000000，(D41、D40)=16#FFFFFFFF，(D31、D30)-(D41、D40)=16#F80000001，运算发生借位，借位标号 SM601=ON，(D51、D50)=16#80000001。

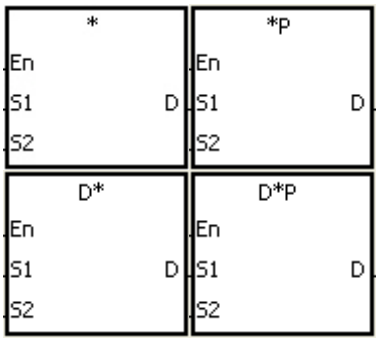
API	指令码			操作数	功能
0102	D	*	P	**S_1 · S_2 · D**	BIN 乘法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



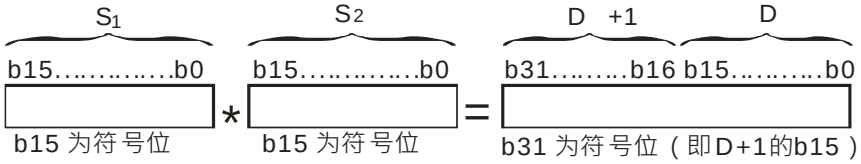
S_1 ： 被乘数

S_2 ： 乘数

D ： 积

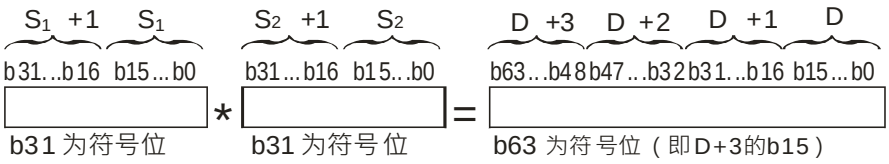
指令说明：

1. **S_1** 及 **S_2** 以有号数二进制方式相乘后的积存于 **D**。
2. D*指令才可以使用 HC 装置。
3. 16 位 BIN 乘法运算：



积为 32 位数据，储存在（D+1，D）组成的 32 位寄存器中，且符号位 b31=0 为正数，符号位 b31=1 为负数。

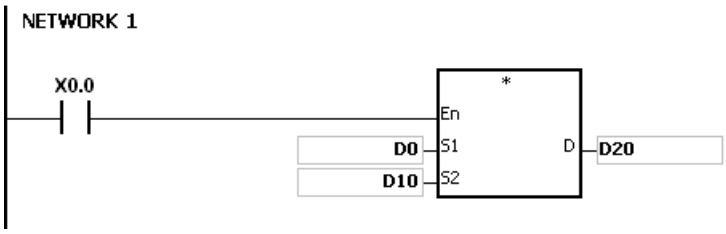
4. 32 位 BIN 乘法运算：



积为 64 位数据，储存在（D+3，D+2，D+1，D）组成的 64 位寄存器中，且符号位 b63=0 为正数，符号位 b63=1 为负数。

程序范例：

16 位 D0 乘上 16 位 D10 得到一个 32 位之积，结果存在（D21，D20）。高 16 位存于 D21，低 16 位存于 D20，结果之正负由最高位（b31）之 OFF/ON 来指示。OFF 表示正的（0），同时 ON 表示负的（1）。



D0×D10=（D21，D20）

16 位×16 位=32 位

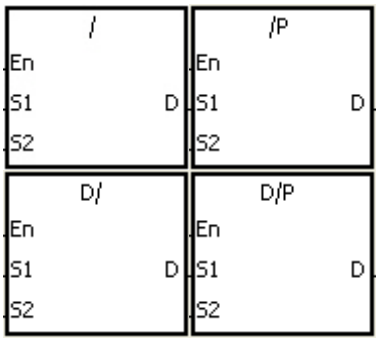
API	指令码			操作数	功能
0103	D	/	P	**S_1 · S_2 · D**	BIN 除法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



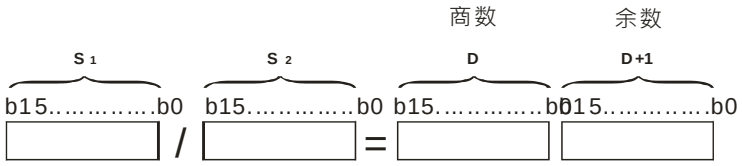
S_1 ： 被除数

S_2 ： 除数

D ： 商及余数

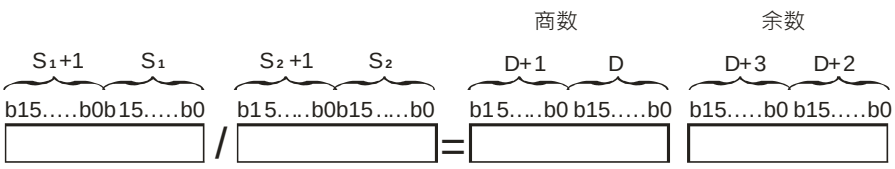
指令说明：

1. **S_1** 及 **S_2** 以有号数二进制方式相除后的商及余数存于 **D**。
2. 32 位指令才可以使用 HC 装置。
3. 符号位=0 为正数，符号位=1 为负数。
4. 16 位 BIN 除法运算：



D 操作数连续占用两个，**D** 储存商，**D**+1 储存余数。

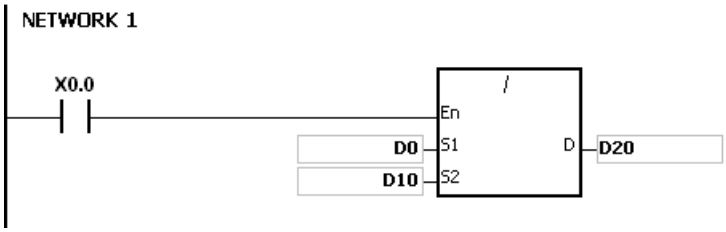
5. 32 位 BIN 除法运算：



D 操作数连续占用两个，(**D**+1，**D**) 储存商，(**D**+3，**D**+2) 储存余数。

程序范例：

当 X0.0=ON 时，被除数 D0 除以除数 D10 而结果商被指定放于 D20，余数指定放于 D21 内。所得结果之正负由最高位位之 OFF/ON 来代表正或负值。



补充说明：

1. 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 若除数为零，指令不执行，SM0=ON，错误码 SR0=16#2012。
3. 16 位指令的 D 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。
4. 32 位指令的 D 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of DWORD/DINT。

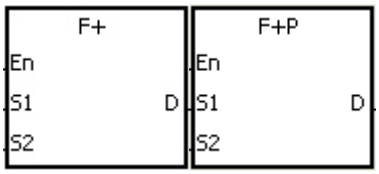
API	指令码			操作数	功能
0104		F+	P	**S_1 · S_2 · D**	浮点数加法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●							○
S_2	●	●			●	●	●	●	●							○
D		●			●	●	●	●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S_1 ： 被加数

S_2 ： 加数

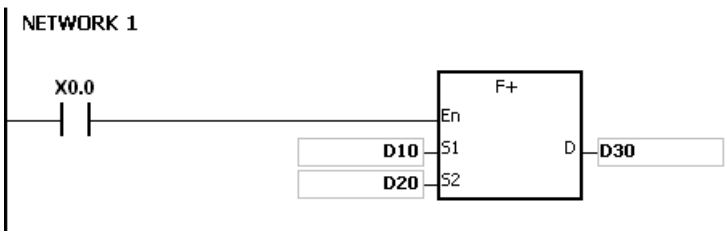
D ： 和

指令说明：

1. 此指令为 32 位单精度浮点数加法的运算。
2. **S_1** 及 **S_2** 以浮点数方式相加的结果存于 **D**。
3. SM600（零标志）、SM601（借位标志）、SM602（进位标志）说明：
   - 运算结果为 0 时，零标号（Zero flag）SM600 为 ON。
   - 若运算结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000 且借位标号 SM601=ON。
   - 若运算结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000 且进位标号 SM602=ON。

程序范例：

32 位单精度浮点数加法：当 X0.0=ON 时，被加数（D11，D10）=16#3FB9999A（表浮点值 1.450）加上加数（D21，D20）=16#4046B852（表浮点值 3.105）将结果存在（D31，D30）=16#4091C28F（表浮点值 4.555）之内容当中。



补充说明：

当 S_1 或 S_2 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

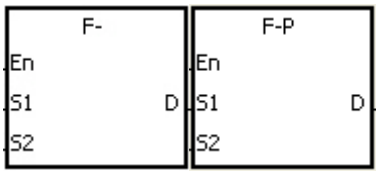
API	指令码			操作数	功能
0105		F-	P	S_1，S_2，D	浮点数减法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●							○
S_2	●	●			●	●	●	●	●							○
D		●			●	●	●	●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



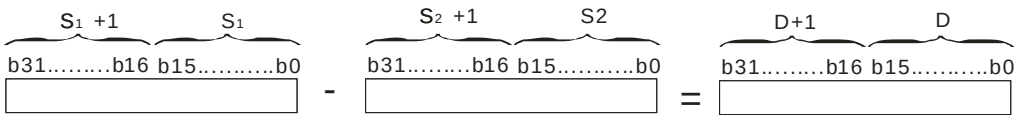
S_1 ： 被减数

S_2 ： 减数

D ： 差

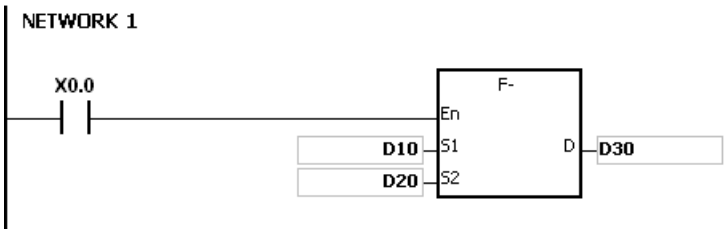
指令说明：

1. 此指令为 32 位单精度浮点数减法的运算。
2. **S_1** 及 **S_2** 以浮点数类型相减的结果存于 **D**。
3. SM600（零标志）、SM601（借位标志）、SM602（进位标志）说明：
   - 若运算结果为 0，则零标号 SM600=ON。
   - 若运算结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000 且 SM601=ON。
   - 若运算结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000 且 SM602=ON。



程序范例：

32 位单精度浮点数减法：当 X0.0=ON 时，被减数（D11，D10）内容减去减数（D21，D20）之内容将结果存在（D31，D30）之内容当中。



补充说明：

当 **S_1** 或 **S_2** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	指令码			操作数	功能
0106		F*	P	**S_1 · S_2 · D**	浮点数乘法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●							○
S_2	●	●			●	●	●	●	●							○
D		●			●	●	●	●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

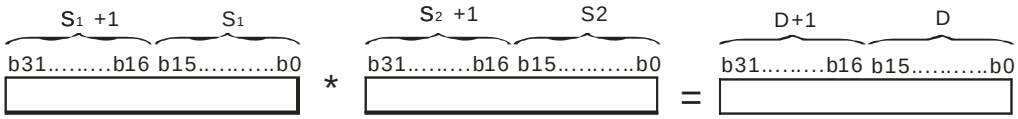
S_1 ： 被乘数 Double Word/Long Word

S_2 ： 乘数 Double Word/Long Word

D ： 积 Double Word/Long Word

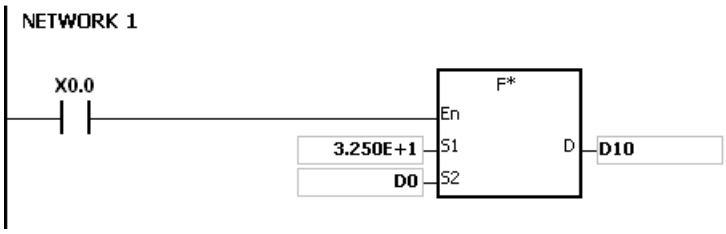
指令说明：

1. 此指令为 32 位单精度浮点数乘法的运算。
2. **S_1** 及 **S_2** 以浮点数类型相乘的结果存于 **D**。
3. SM600（零标志）、SM601（借位标志）、SM602（进位标志）说明：
   - 若运算结果为 0，则零标号 SM600=ON。
   - 若运算结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000 且 SM601=ON。
   - 若运算结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000 且 SM602=ON。



程序范例：

32 位单精度浮点数乘法：当 X0.0=ON 时，被乘数常数 32.5 乘上乘数（D1，D0）之内容，并将结果存在（D11，D10）之内容当中。



补充说明：

当 **S_1** 或 **S_2** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

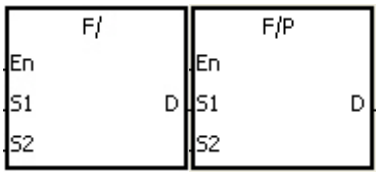
API	指令码			操作数	功能
0107		F/	P	**S_1 · S_2 · D**	浮点数除法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●							○
S_2	●	●			●	●	●	●	●							○
D		●			●	●	●	●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



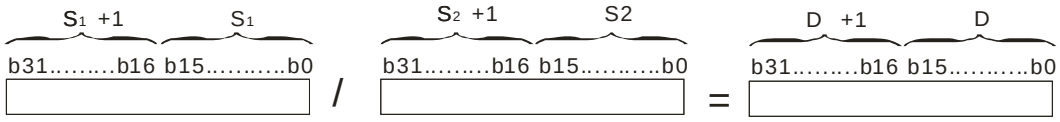
S_1 ： 被除数

S_2 ： 除数

D ： 商

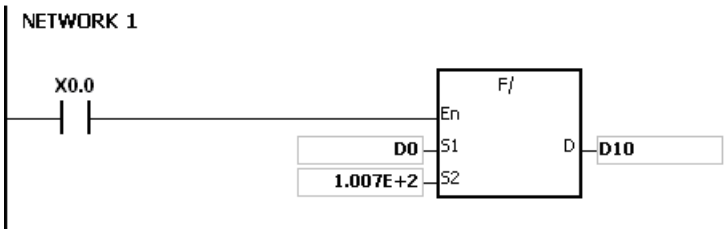
指令说明：

1. 此指令为 32 位单精度浮点数除法的运算。
2. **S_1** 及 **S_2** 以单精度浮点数类型相除的结果存于 **D**。
3. SM600（零标志）、SM601（借位标志）、SM602（进位标志）说明：
   - 若运算结果为 0，则零标号 SM600=ON。
   - 若运算结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000 且 SM601=ON。
   - 若运算结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000 且 SM602=ON。



程序范例：

32 位单精度浮点数除法：当 X0.0=ON 时，被除数（D1，D0）之内容除以除数 100.7，并将结果存在（D11，D10）之内容当中。



补充说明：

1. 除数为 0 时，指令不执行，SM0=ON，错误码 SR0=16#2012。
2. 当 S_1 或 S_2 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

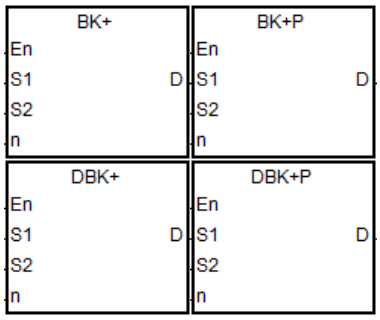
API	指令码			操作数	功能
0112	D	BK+	P	**S_1 · S_2 · n · D**	连续区块 BIN 加法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●							
S_2	●	●			●	●	●	●	●				○	○		
n	●	●			●	●	●	●	●				○	○		
D		●			●	●	●	●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●					●	●	
S_2		●				●					●	●	
n		●				●					●	●	
D		●				●					●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 被加数

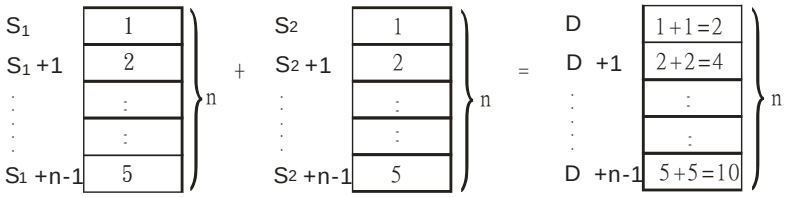
S_2 ： 加数

n ： 要相加的笔数

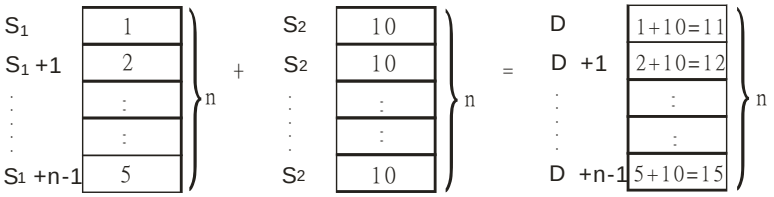
D ： 和

指令说明：

1. 连续 **n** 个 **S_1** 及 **S_2** 以 BIN 方式相加的结果存于 **D**。
2. **n** 操作数的范围为 1~256。
3. 32 位指令才可以使用 HC 装置。
4. 运算结果为 0 时，**零标号**（Zero flag）SM600 为 ON。
5. 16 位指令，运算结果小于–32,768 时，借位标号（Borrow flag）SM601 为 ON。
6. 16 位指令，运算结果大于 32,767 时，**进位**标号（Carry flag）SM602 为 ON。
7. 32 位指令，运算结果小于–21,474,836,488 时，借位标号（Borrow flag）SM601 为 ON。
8. 32 位指令，运算结果大于 2,147,483,647 时，进位标号（Carry flag）SM602 为 ON。
9. 当 **S_2** 是装置（非 K/16#）的时候:

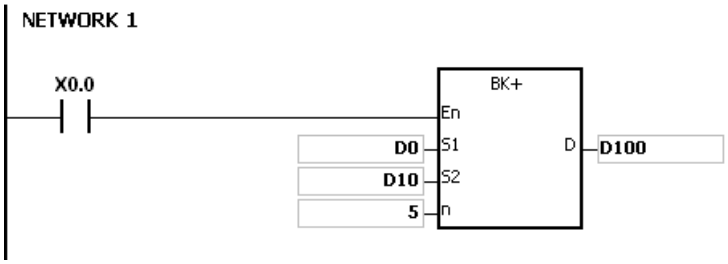


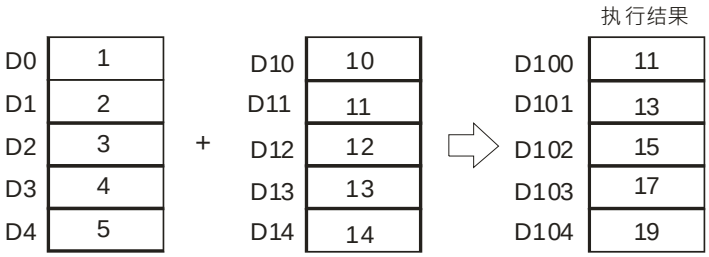
10. 当 **S_2** 是**常数** K/16#的时候：



程序范例一：

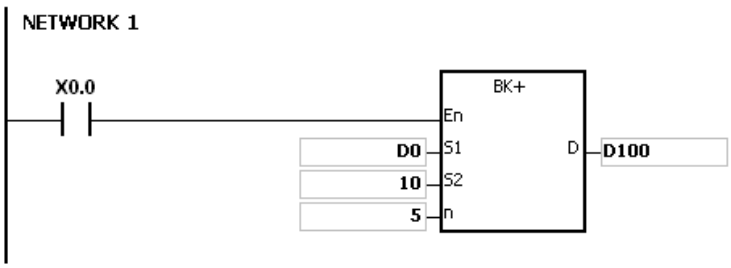
X0.0=ON 时，将被加数 D0~D4 内容与加数 D10~D14 内容作 BIN 相加后储存至 D100~D104 当中。

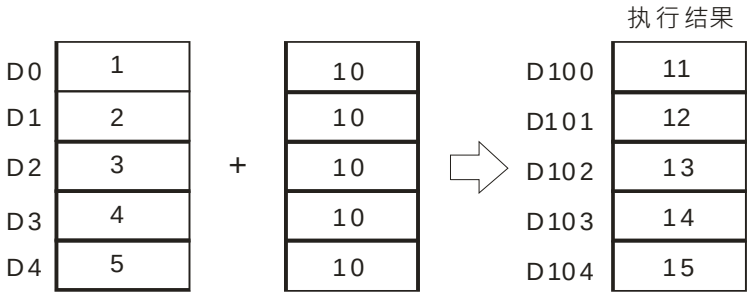




程序范例二：

X0.0=ON 时，将被加数 D0~D4 的内容与加数 10 的内容作 BIN 相加后储存至 D100~D104 当中。





补充说明：

1. 16 位指令中，**S_1**、**S_2** 或 **D** 装置的起始地址~**n**-1 的地址，超出装置范围时，指令不执行 SM0=ON，错误码 SR0=16#2003。
2. 32 位指令中，**S_1**、**S_2** 或 **D** 装置的起始地址~2***n**-1 的地址，超出装置范围时，指令不执行 SM0=ON，错误码 SR0=16#2003。
3. 若 **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 16 位指令中，若 **S_1**~**S_1**+**n**-1 的装置与 **D**~**D**+**n**-1 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。
5. 32 位指令中，若 **S_1**~**S_1**+2***n**-1 的装置与 **D**~**D**+2***n**-1 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。
6. 16 位指令中，若 **S_2**~**S_2**+**n**-1 的装置与 **D**~**D**+**n**-1 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。
7. 32 位指令中，若 **S_2**~**S_2**+2***n**-1 的装置与 **D**~**D**+2***n**-1 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。

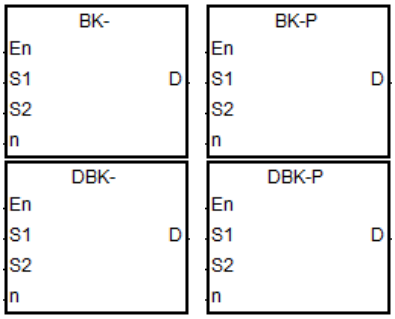
API	指令码			操作数	功能
0113	D	BK-	P	**S_1，S_2，n，D**	连续区块 BIN 减法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●							
S_2	●	●			●	●	●	●	●				○	○		
n	●	●			●	●	●	●	●				○	○		
D		●			●	●	●	●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●					●	●	
S_2		●				●					●	●	
n		●				●					●	●	
D		●				●					●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 被减数

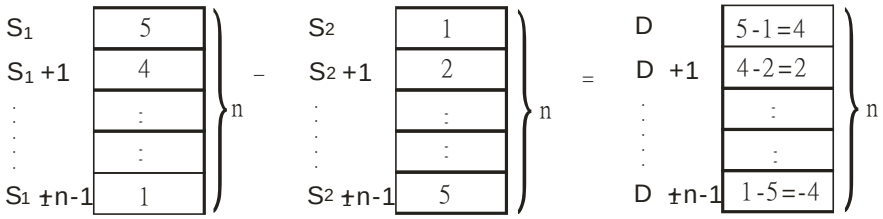
S_2 ： 减数

n ： 要相减的笔数

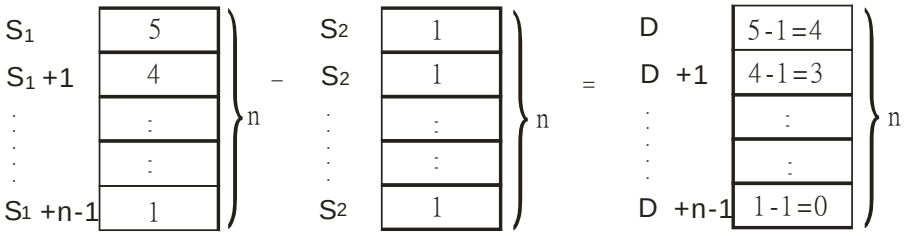
D ： 差

指令说明：

1. 连续 **n** 个 **S_1** 及 **S_2** 以 BIN 方式相减的结果存于 **D**。
2. **n** 操作数的范围为 1~256。
3. 32 位指令才可以使用 HC 装置。
4. 运算结果为 0 时，零标号（Zero flag）SM600 为 ON。
5. 16 位指令，运算结果小于–32,768 时，借位标号（Borrow flag）SM601 为 ON。
6. 16 位指令，运算结果大于 32,767 时，进位标号（Carry flag）SM602 为 ON。
7. 32 位指令，运算结果小于–2,147,483,648 时，借位标号（Borrow flag）SM601 为 ON。
8. 32 位指令，运算结果大于 2,147,483,647 时，进位标号（Carry flag）SM602 为 ON。
9. 以 16 位指令为例，当 **S_2** 是装置（非 K/16#）的时候:

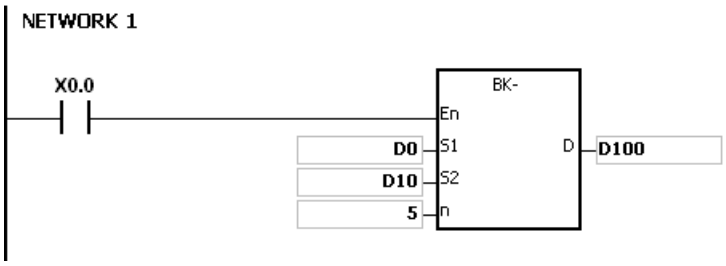


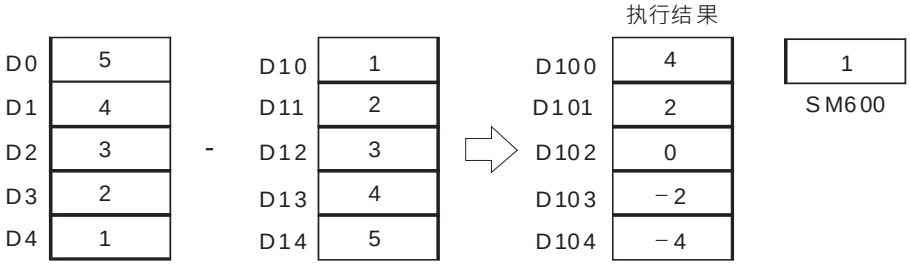
10. 以 16 位指令为例，当 **S_2** 是常数 K/16#的时候：



程序范例一：

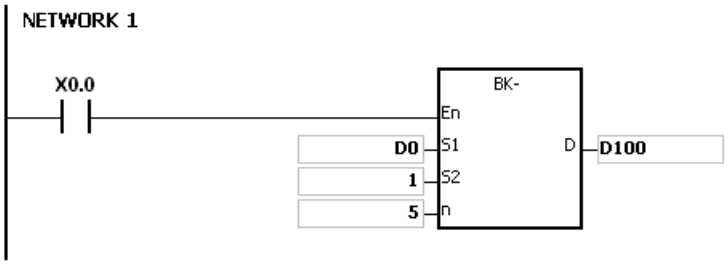
X0.0=ON 时，将被减数 D0~D4 内容与减数 D10~D14 内容作 BIN 相减后储存至 D100~D104 当中。

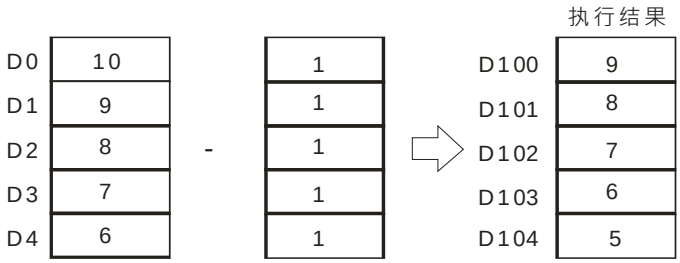




程序范例二：

X0.0=ON 时，将被加数 D0~D4 的内容与减数 1 的内容作 BIN 相减后储存至 D100~D104 当中。





补充说明：

1. 16 位指令中，**S_1**、**S_2**或 **D** 装置的起始地址~**n**-1 的地址，超出装置范围时，指令不执行 SM0=ON，错误码 SR0=16#2003。
2. 32 位指令中，**S_1**、**S_2**或 **D** 装置的起始地址~2***n**-1 的地址，超出装置范围时，指令不执行 SM0=ON，错误码 SR0=16#2003。
3. 若 **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 16 位指令中，若 **S_1**~**S_1**+**n**-1 的装置与 **D**~**D**+**n**-1 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。
5. 32 位指令中，若 **S_1**~**S_1**+2***n**-1 的装置与 **D**~**D**+2***n**-1 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。
6. 16 位指令中，若 **S_2**~**S_2**+**n**-1 的装置与 **D**~**D**+**n**-1 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。
7. 32 位指令中，若 **S_2**~**S_2**+2***n**-1 的装置与 **D**~**D**+2***n**-1 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。

API	指令码			操作数	功能
0114		$+	P	**S_1 · S_2 · D**	字符串链结

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●						○	
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													●
S_2													●
D													●

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

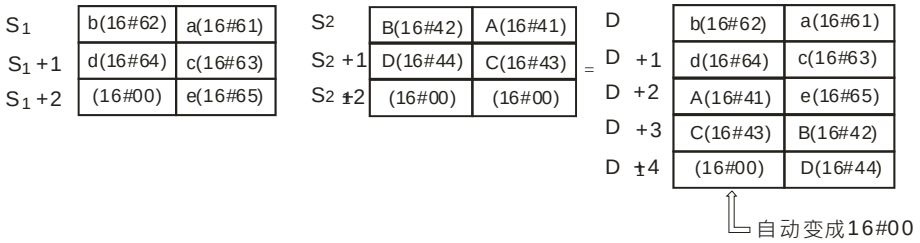
S_1 ： 字符串 1

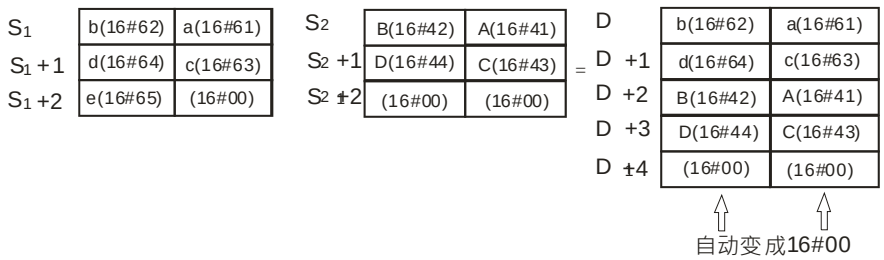
S_2 ： 字符串 2

D ： 储存装置起始地址

指令说明：

1. 该指令执行时，将 **S_1** 操作数 **S_1~**16#00（不搬 16#00）与 **S_2** 操作数 **S_2~** 16#00（不搬 16#00）之间的字符串链结后搬移至 **D** 操作数内且结尾会补上结后缀（16#00）。当指令不执行时，**D** 内容不会变化。
2. **S_1**、**S_2** 及 **D** 操作数的来源不是字符串（$）时，字符串内容限制最多 256 个字（包含结束符号 16#00）。
3. **S_1** 操作数及 **S_2** 操作数两个字符串链结的结果存于 **D** 操作数的说明如下：





4. 若 **S_1** 或 **S_2** 操作数的来源不是字符串时，就要在数据结尾后面有一个结后缀（16#00）。

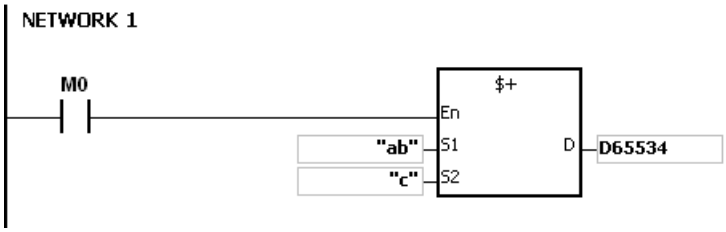
5. 若 **S_1** 或 **S_2** 操作数不是字符串且该指令执行时，第一个字符就是结后缀（16#00），则还是会搬移结后缀（16#00）做链结。

6. 字符串"abcde"在 **S_1** 操作数表示为：

S_1	b(16#62)	a(16#61)
S_1+1	d(16#64)	c(16#63)
S_1+2	(16#00)	e(16#65)

程序范例：

给定 **S_1**="ab"，**S_2**="c"，启动条件接点 M0 后，D65534=16#6261，D65535=16#0063



补充说明：

1. 若 **S_1** 或 **S_2** 操作数为字符串时，**S_1** 或 **S_2** 操作数各可以搬移 1~31 个字符。
2. 当 **D** 装置的空间不足以存放 **S_1+S_2** 字符串时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. 当 **S_1+S_2** 字符串>256（含结束符号 16#00）时，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. **S_1** 或 **S_2** 装置与 **D** 装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。
5. **S_1** 或 **S_2** 没有 16#00 当结尾时，指令不执行，SM0=ON，错误码 SR0=16#200E。

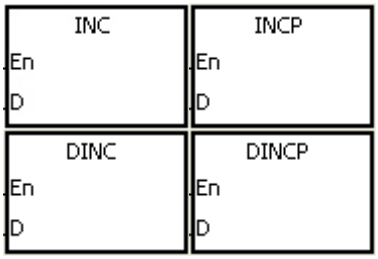
API	指令码			操作数	功能
0115	D	INC	P	**D**	BIN 加 1

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ： 目的地装置

指令说明：

1. **D** 内容加 1。
2. DINC 指令才可以使用 HC 装置，但不可使用 E 装置。
3. 16 位运算时，32,767 再加 1 则变为-32,768。32 位运算时，2,147,483,647 再加 1 则变为-2,147,483,648。

程序范例：

当 X0.0=OFF→ON 时，D0 内容自动加 1。

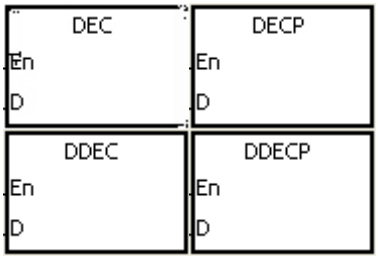
API	指令码			操作数	功能
0116	D	DEC	P	**D**	BIN 减 1

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ：目的地装置

指令说明：

1. **D** 内容减 1。
2. DDEC 指令才可以使用 HC 装置，但不可使用 E 装置。
3. 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：

当 X0.0=OFF→ON 时，D0 内容自动减 1。

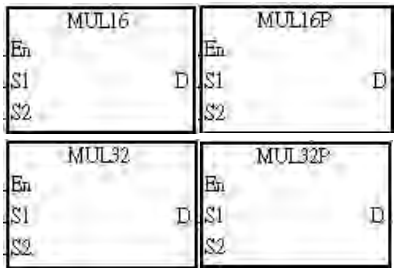
API	指令码			操作数	功能
0117		MUL16 MUL32	P	**S_1 · S_2 · D**	16 位专用 BIN 乘法 32 位专用 BIN 乘法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



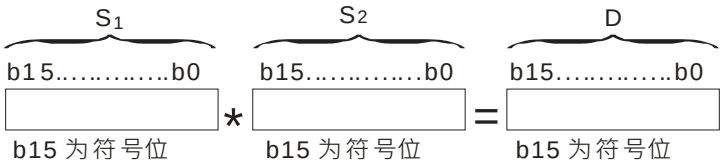
S_1 ： 被乘数

S_2 ： 乘数

D ： 积

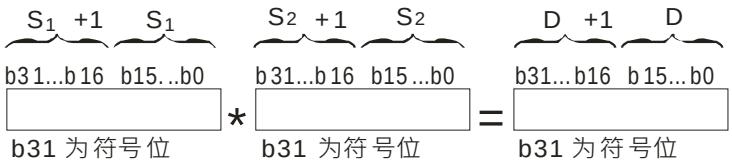
指令说明：

1. **S_1** 及 **S_2** 以有号数二进制方式相乘后的积存于 **D**。
2. MUL32 指令才可以使用 HC 装置，但不可使用 E 装置。
3. 16 位 BIN 乘法运算：



积为 16 位数据，储存在 D 的 16 位寄存器中，且符号位 b15=0 为正数，符号位 b15=1 为负数。

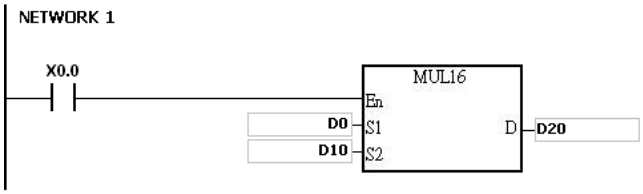
4. 32 位 BIN 乘法运算：



积为 32 位数据，储存在（D+1，D）组成的 32 位寄存器中，且符号位 b31=0 为正数，符号位 b31=1 为负数。

程序范例：

16 位 D0 乘上 16 位 D10 得到一个 16 位之积，结果存在 D20。结果之正负由最高位（b15）之 OFF/ON 来指示。OFF 表示正的（0），同时 ON 表示负的（1）。



D0×D10=D20

16 位×16 位=16 位

补充说明：

1. 当 16 位乘法之积超出 16 位有号数可表示范围时，则数值比 16 位最大正数（K32767）还大或者数值比最小负数（K-32768）还小时，设定 SM602 进位标志为 ON；并只写入低 16 位的数值
2. 若 16 位指令相乘结果需要得到完整的数值（记录为 32 位），请改用 API0102 * / *P 指令，详细说明请参考该指令。
3. 当 32 位乘法之积超出 32 位有号数可表示范围时，则数值比 32 位最大正数（K2147483647）还大或者数值比最小负数（K-2147483648）还小时，设定 SM602 进位标志为 ON；并只写入低 32 位的数值。
4. 若 32 位指令相乘结果需要得到完整的数值（记录为 64 位），请改用 API0102 D* / D*P 指令，详细说明请参考该指令。

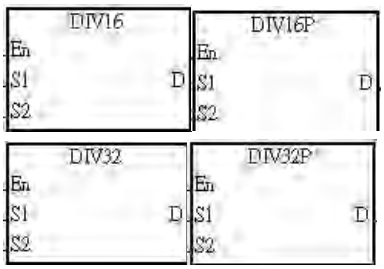
API	指令码			操作数	功能
0118		DIV16 DIV32	P	**S_1 · S_2 · D**	16 位专用 BIN 除法 32 位专用 BIN 除法

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	●	●		
S_2	●	●			●	●	●	●	●		○	○	●	●		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



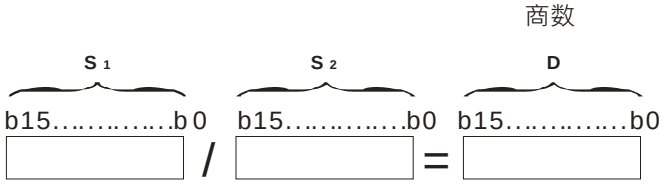
S_1 ： 被除数

S_2 ： 除数

D ： 商

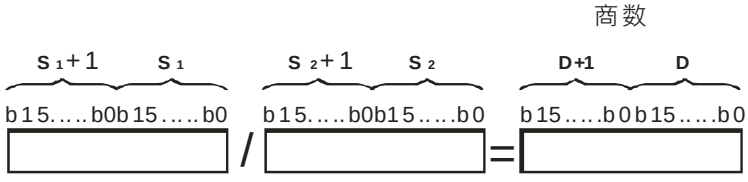
指令说明：

1. **S_1** 及 **S_2** 以有号数二进制方式相除后的商存于 **D**。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。
3. 符号位=0 为正数，符号位=1 为负数。
4. 16 位 BIN 除法运算：



D 操作数储存商。

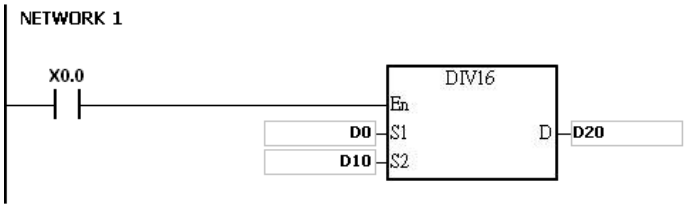
5. 32 位 BIN 除法运算：



D 操作数续占用两个，(**D**+1，**D**) 储存商。

程序范例：

当 X0.0=ON 时，被除数 D0 除以除数 D10 而结果商被指定放于 D20。所得结果之正负由最高位位之 OFF/ON 来代表正或负值。



补充说明：

1. 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 若除数为零，指令不执行，SM0=ON，错误码 SR0=16#2012。
3. 若需要记录余数，请改用 API0103，详细说明请参考该指令。

6.3 数据转换指令

6.3.1 数据转换指令一览表

API	功能码（位）		P 指令	功能
	16	32		
0200	BCD	DBCD	✓	BIN→BCD 变换
0201	BIN	DBIN	✓	BCD→BIN 变换
0202	FLT	DFLT	✓	BIN 整数→单精度浮点数变换
0204	INT	DINT	✓	单精度浮点数→BIN 整数变换
0206	MMOV	–	✓	16→32 位数值转换
0207	RMOV	–	✓	32→16 位数值转换
0208	GRY	DGRY	✓	BIN→GRY 码变换
0209	GBIN	DGBIN	✓	GRY 码→BIN 变换
0210	NEG	DNEG	✓	取负数（取 2 的补码）
0211	–	FNEG	✓	单精度浮点数正负符号反相
0212	–	FBCD	✓	单精度浮点数→十进浮点数
0213	–	FBIN	✓	十进浮点数→单精度浮点数
0214	BKBCD	–	✓	连续区块 BIN→BCD 变换
0215	BKBIN	–	✓	连续区块 BCD→BIN 变换
0216	SCAL	DSCAL	✓	比例运算
0217	SCLP	DSCLP	✓	参数型比例运算
0222	SCLM	DSCLM	✓	多点式区域比例运算

6.3.2 数据转换指令说明

API	功能码			操作数	功能
0200	D	BCD	P	**S**，**D**	BIN→BCD 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○	○				
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：

S ：变换来源装置

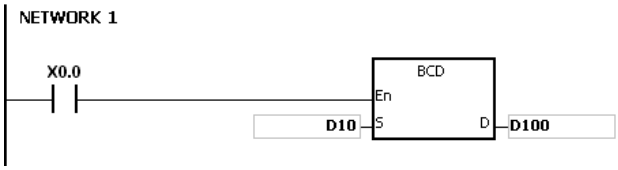
D ：变换的结果

指令说明：

1. 数据源 **S** 的内容（BIN 值）作 BCD 的转换，存于 **D**。
2. DBCD 才可以使用 HC 装置，但不可使用 E 装置。
3. PLC 内的四则运算及 INC、DEC 指令都是以 BIN 方式来执行。所以在应用方面，当要看到 10 进制数值的显示器时，用 BCD 转换即可将 BIN 值变为 BCD 值输出。

程序范例：

1. 当 X0.0=ON 时，D10 的 BIN 值被转换成 BCD 值后，将结果存于 D100 当中。



2. 若 D10=16#04D2=1234，则执行结果 D100=16#1234

补充说明：

1. 在 BCD 转换结果若超过 0~9,999，指令不执行，SM0=ON，错误码 SR0=16#200D。(BCD 值以 HEX 表示有任一位数不在 0~9 的范围内)。
2. 在 DBCD 转换结果若超过 0~99,999,999，指令不执行，SM0=ON，错误码 SR0=16#200D。(BCD 值以 HEX 表示有任一位数不在 0~9 的范围内)。

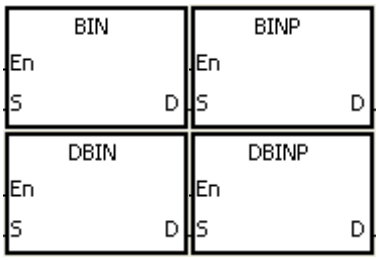
API	功能码			操作数	功能
0201	D	BIN	P	**S**、**D**	BCD→BIN 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○				
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 变换来源装置

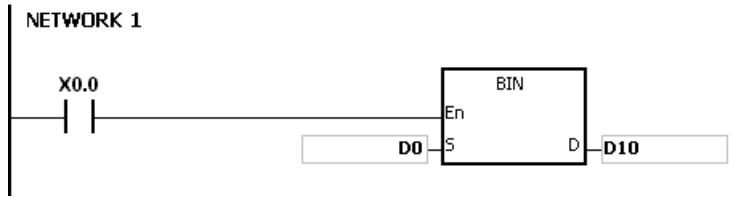
D ： 变换的结果

指令说明：

1. 数据源 **S** 的内容（BCD：0~9,999）、DBCD（0~99,999,999）作 BIN 的转换，存于 D。
2. 数据源 **S** 的内容有效数值范围：BCD（0~9,999）、DBCD（0~99,999,999）。
3. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。
4. 常数 K、16#会自动转换成 BIN，故不需运用此指令。

程序范例：

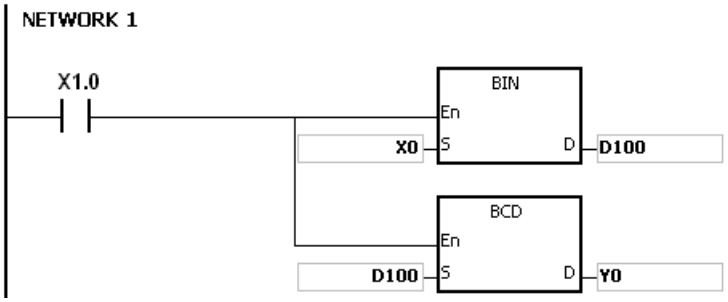
当 X0.0=ON 时，D0 的 BCD 值被转换成 BIN 值后，将结果存于 D10 中。

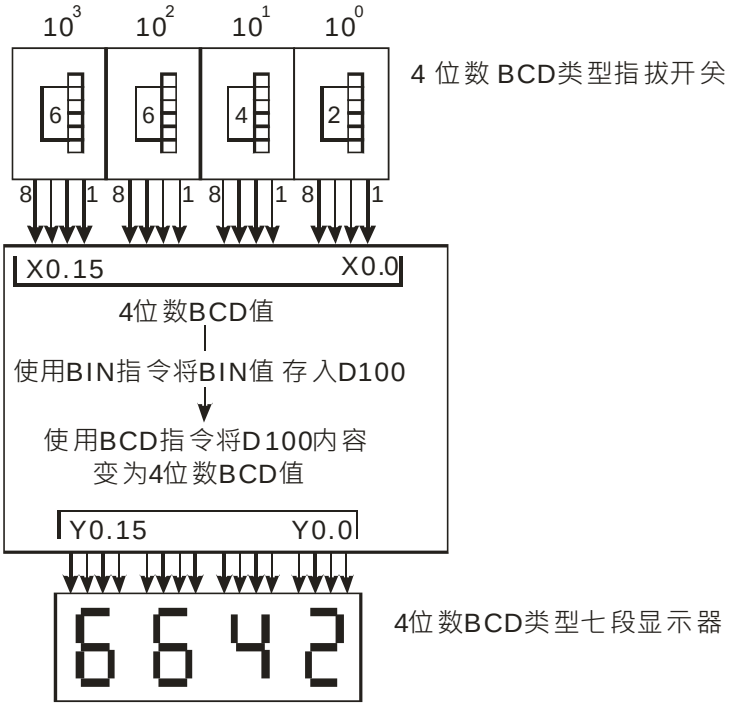


补充说明：

1. 当 **S** 的数据内容并非为 BCD 值（以 HEX 表示有任一位数不在 0~9 的范围内），则将会产生运算错误，SM0=ON，错误码 SR0=16#200D。
2. BCD 与 BIN 指令应用说明：
   - 当 PLC 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 PLC 内。

- 当 PLC 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。
- 当 X1.0=ON 时，将 X0.0~X0.15 BCD 值转换成 BIN 值传送到 D100，再将 D100 的 BIN 值转换成 BCD 值传送到 Y0.0~Y0.15。





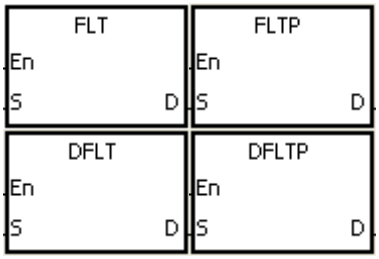
API	功能码			操作数	功能
0202	D	FLT	P	**S**、**D**	BIN 整数→单精度浮点数变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○				
D		●					●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D									●				

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 变换来源装置

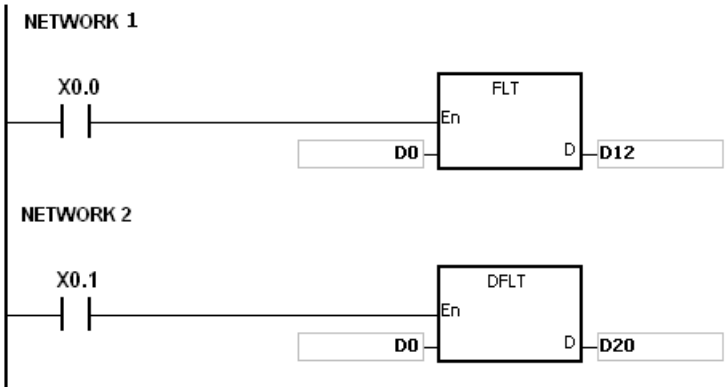
D ： 存放变换结果之装置

指令说明：

1. 将 BIN 整数变换成单精度浮点数。
2. FLT 中 **S** 不可使用 HC 装置，**D** 不可使用 E 装置。
3. FLT 中 **S** 变换来源装置占用 1 个寄存器，**D** 存放变换结果的装置占用 2 个寄存器。
4. DFLT 中 **S** 变换来源装置占用 2 个寄存器，**D** 存放变换结果的装置占用 2 个寄存器。
   - 若转换结果的绝对值大于可表示之最大浮点值，则进位标号 SM602=ON，**D** 存放最大浮点值。
   - 若转换结果的绝对值小于可表示之最小浮点值，则借位标号 SM601=ON，**D** 存放最小浮点值。
   - 若转换结果为 0，则零标号 SM600=ON。

程序范例一：

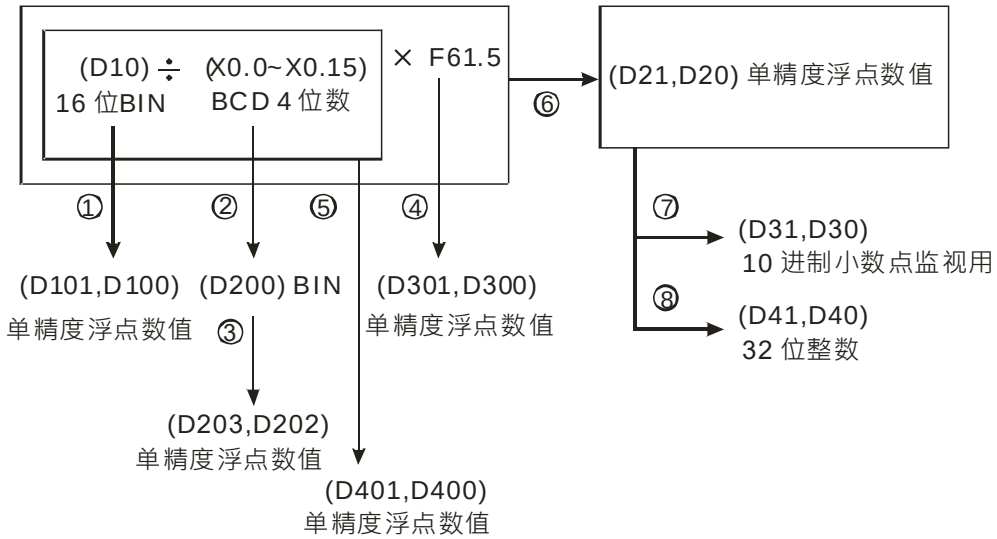
1. 当 X0.0=ON 时，将 D0（内为 BIN 整数）变换成单精度浮点数存放于（D13、D12）。
2. 当 X0.1=ON 时，将 D1、D0（内为 BIN 整数）变换成单精度浮点数存放于（D21、D20）。
3. 若 D0=10，则 X0.0=ON，转换后单精度浮点数为 16#41200000，存于 32 位寄存器（D13、D12）内。
4. 若 32 位寄存器（D1、D0）=100,000，则 X0.1=ON，转换后单精度浮点数为 16#47C35000，存于 32 位寄存器（D21、D20）内。

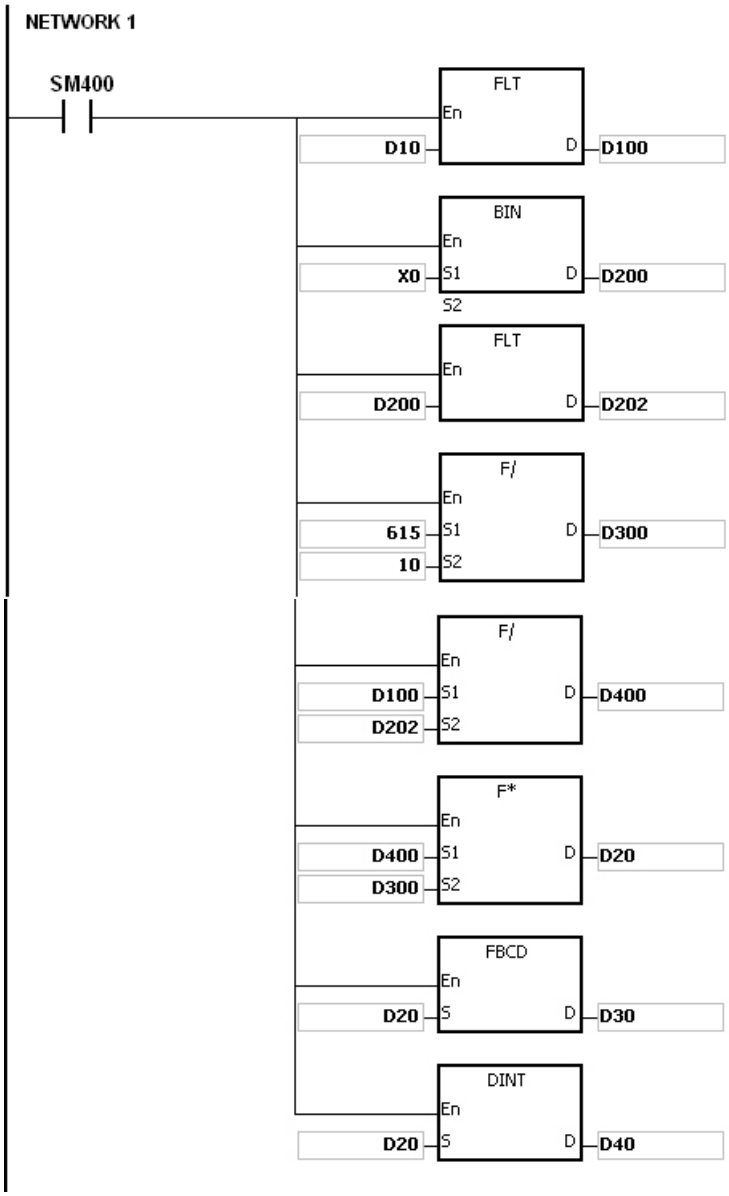


程序范例二：

使用应用指令来完成下列的算式。

- 将 D10（内为 BIN 整数）变换成单精度浮点数存放于（D101、D100）。
- 将 X0.0~X0.15（BCD 值）变换成 D200（BIN 值）。
- 将 D200（内为 BIN 整数）变换成单精度浮点数存放于（D203、D202）。
- 将常数 615÷10 结果存于 D301、D300（单精度浮点数）。
- 单精度浮点数除法（D101、D100）÷（D203、D202）结果存于 D401、D400（单精度浮点数）。
- 单精度浮点数乘法（D401、D400）×（D301、D300）结果存于 D21、D20（单精度浮点数）。
- 单精度浮点数 D21、D20 变换成 10 进小数点值 D31、D30。
- 单精度浮点数 D21、D20 变换成 BIN 整数 D41、D40。




NETWORK 1
SM400
FLT
En
D10
D
D100
BIN
En
X0
S1
D
D200
S2
FLT
En
D200
D
D202
F/
En
615
S1
D
D300
10
S2
F/
En
D100
S1
D
D400
D202
S2
F*
En
D400
S1
D
D20
D300
S2
FBCD
En
D20
S
D
D30
DINT
En
D20
S
D
D40

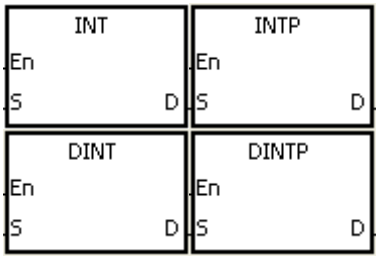
API	功能码			操作数	功能
0204	D	INT	P	**S**，**D**	单精度浮点数→BIN 整数变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○					
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ：变换来源装置

D ：变换的结果

指令说明：

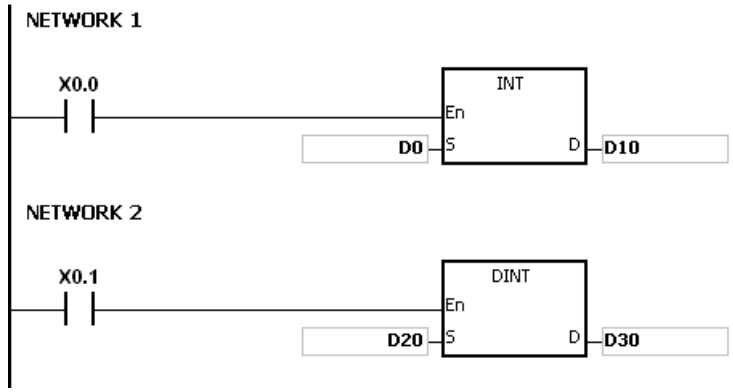
1. **S** 所指定的寄存器内容以单精度浮点数类型被变换成 BIN 整数暂存于 D 所指定的寄存器当中，BIN 整数浮点数被舍弃。
2. INT 中 **S** 变换来源装置占用 2 个寄存器，D 存放变换结果的装置占用 1 个寄存器。
3. DINT 中 **S** 变换来源装置占用 2 个寄存器，D 存放变换结果的装置占用 2 个寄存器。
4. INT 中 **D** 不可使用 HC 装置。DINT 不可使用 E 装置。
5. 本指令的动作与 API0202（FLT）指令刚好相反。
6. 变换结果若为 0 时，零标志 SM600=ON。
7. 变换结果有浮点数被舍弃时，借位标志 SM601=ON。
8. 变换结果若超出下列范围时（溢位），进位标志 SM602=ON。

   INT/IINTP：-32,768~32,767

   DINT/DINTP：-2,147,483,648~2,147,483,647

程序范例：

1. 当 X0.0=ON 时，将单精度浮点数（D1，D0）变换成 BIN 整数将结果存放至 D10 当中，BIN 整数浮点数被舍弃。
2. 当 X0.1=ON 时，将单精度浮点数（D21，D20）变换成 BIN 整数将结果存放至（D31，D30）当中，BIN 整数浮点数被舍弃。



补充说明：

当 **S** 内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	功能码			操作数	功能
0206		MMOV	P	**S**・**D**	16→32 位数值转换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●		○	○	○	○		
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●	●				●	●	
D			●				●						

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 变换来源装置

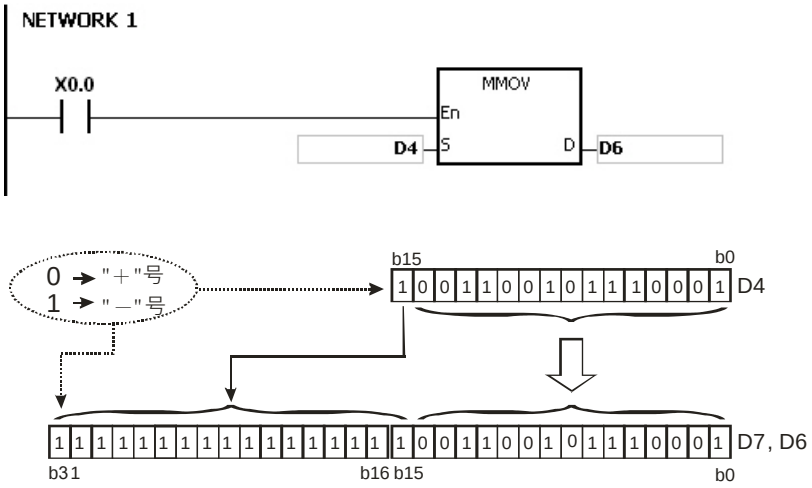
D ： 变换的结果

指令说明：

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 X0.0 为 0N 时，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	功能码			操作数	功能
0207		RMOV	P	**S**・**D**	32→16 位数值转换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○		○	○		
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S			●				●						
D		●				●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 变换来源装置

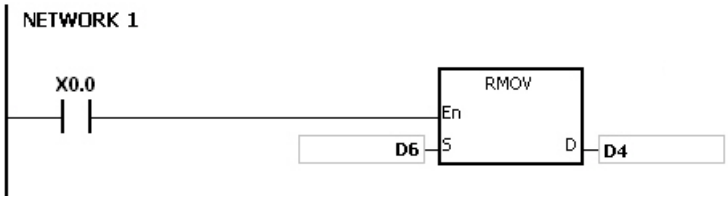
D ： 变换的结果

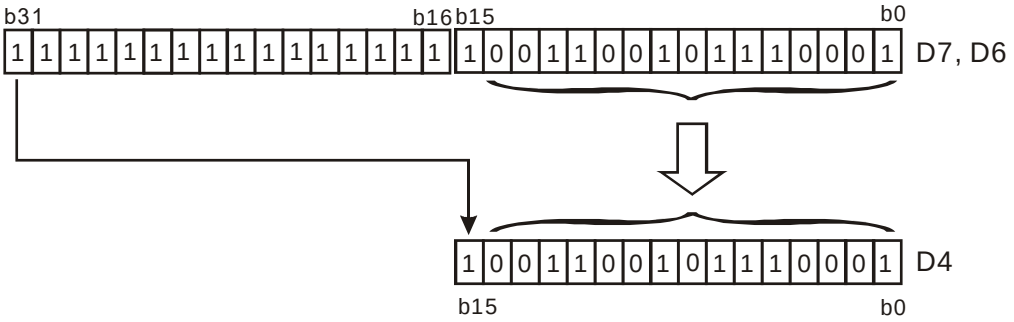
指令说明：

将 32 位装置 **S** 中的数据传送到 16 位的装置 **D** 中，其中指定的符号位被保留。

程序范例：

当 X0.0 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。





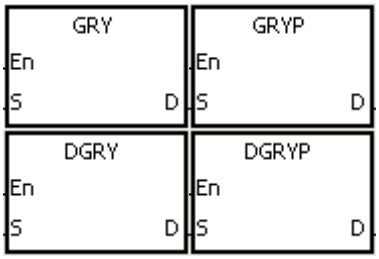
API	功能码			操作数	功能
0208	D	GRY	P	**S**・**D**	BIN→GRAY 码变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 变换来源装置

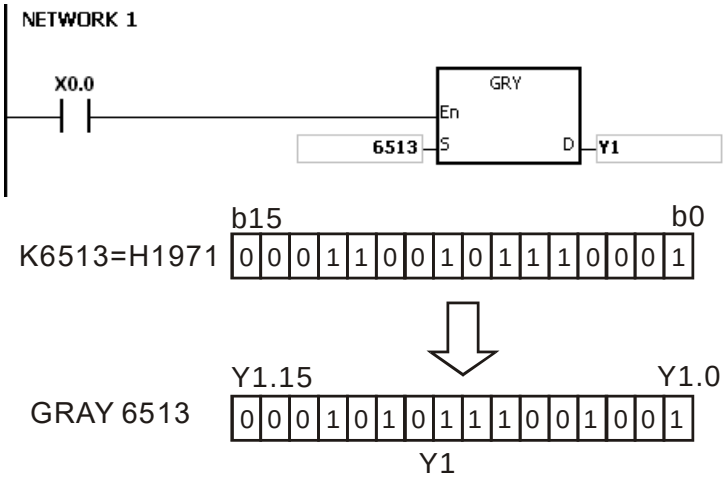
D ： 变换之结果

指令说明：

1. 将 S 所指定装置的内容值（BIN 值）变换格雷码（GRAY CODE）后存放到 D 所指定的装置中。
2. DGRY 才可使用 HC 装置，但不可使用 E 装置。
3. **S** 的有效范围如下所示：
   16 位指令：0~32,767
   32 位指令：0~2,147,483,647

程序范例：

当 X0.0=ON 时，将常数 6513 变换格雷码（GRAY CODE）后存放到 Y1.0~Y1.15。



补充说明：

S 的内容值小于 0，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。

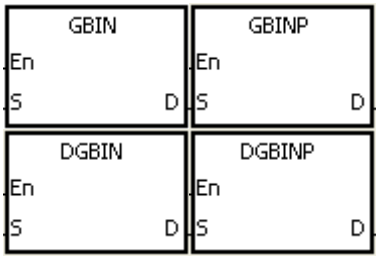
API	功能码			操作数	功能
0209	D	GBIN	P	**S**・**D**	GRAY 码→BIN 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 变换来源装置

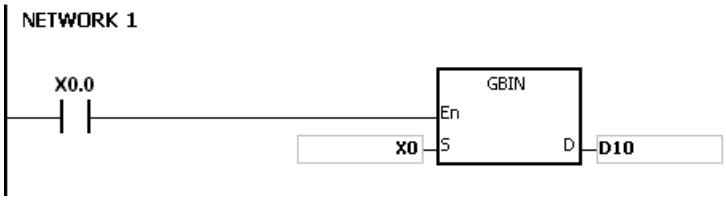
D ： 变换的结果

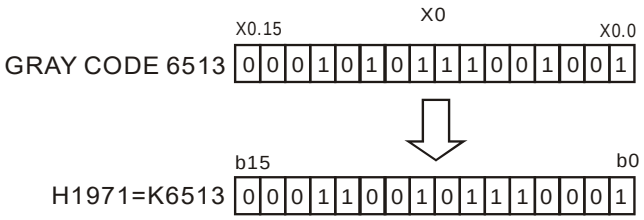
指令说明：

1. 将 **S** 所指定装置的内容值（格雷码（GRAY CODE））变换成 BIN 值后存放到 D 所指定的装置中。
2. 本指令将连接于 PLC 输入端的绝对位置型编码器（此编码器的输出值通常是格雷码）的内容变换成 BIN 值存放到指定的寄存器当中。
3. DGBIN 才可使用 HC 装置，但不可使用 E 装置。
4. **D** 的有效范围如下所示：
   16 位指令：0~32,767
   32 位指令：0~2,147,483,647

程序范例：

当 X0.0 时，将 X0.0~X0.15 输入点所连接的绝对位置型编码器其格雷码（GRAY CODE）变换成 BIN 值后存放到 D10 中。





补充说明：

S 的内容值小于 0，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。

API	功能码			操作数	功能
0210	D	NEG	P	**D**	2 的补码

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：

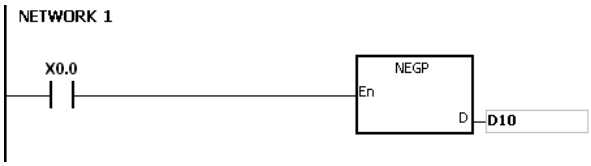
D ： 欲取 2 的补码的装置

指令说明：

1. 本指令可将负数的 BIN 值转换成绝对值。
2. DNEG 才可使用 HC 装置，但不可使用 E 装置。
3. 本指令一般都是使用脉冲执行型指令（NEGP、DNEGP）。

程序范例一：

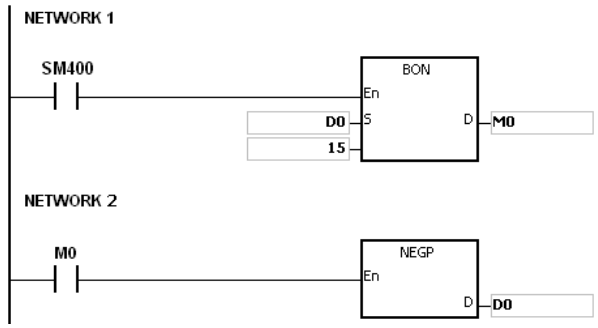
当 X0.0=OFF→ON 时，D10 内容的各位全部反相（0→1、1→0）后再加 1 存放于原寄存器 D10 当中。



程序范例二：

求负数的绝对值：

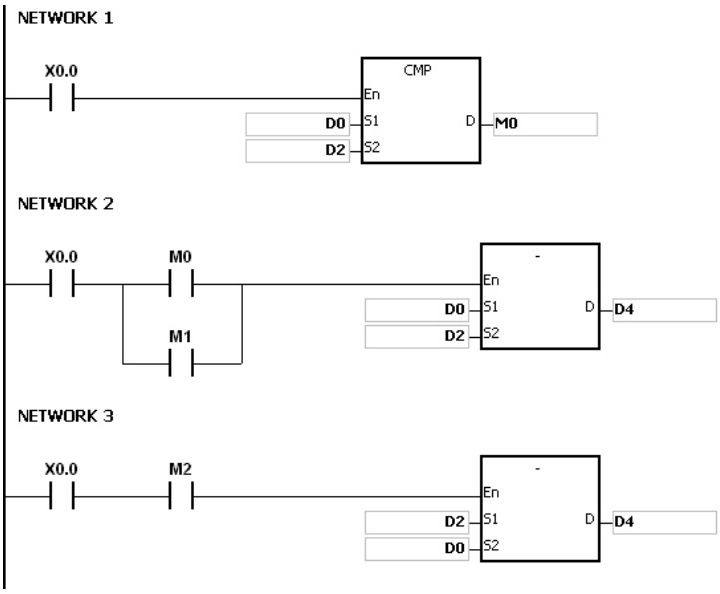
1. 当 D0 的第 15 个位为“1”时，M0=ON。(D0 表示为负数)
2. M0=ON 时，用 NEG 指令将 D0 取 2 的补码可得到其绝对值。



程序范例三：

减法运算之差取绝对值，当 X0.0=ON 时：

1. 若 D0>D2 时，M0=ON。
2. 若 D0=D2 时，M1=ON。
3. 若 D0<D2 时，M2=ON。
4. 此可得 D4 保持为正值。



补充说明：

数的表现及绝对值：

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

(D0)=2

0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

(D0)=1

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

(D0)=0

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

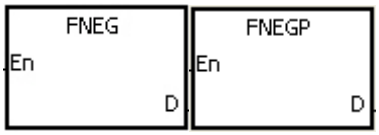
API	功能码			操作数	功能
0211		FNEG	P	**D**	单精度浮点数正负符号反相

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



D ： 欲取符号反向的装置

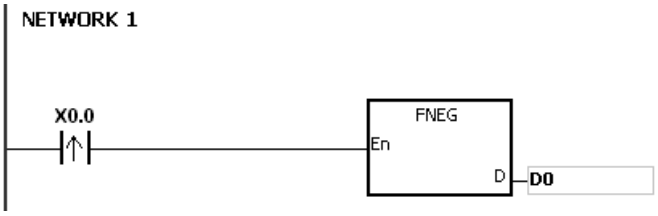
指令说明：

将装置 **D** 中的内容，以单精度浮点数的类型做正负符号反相。

程序范例：

执行前（D1、D0）=16#AE0F9000（负值），当 X0.0 从 OFF→0N 时，（D1、D0）的数据以单精度浮点数做正负符号的反相运作，因此执行后（D1、D0）=16#2E0F9000（正值）。

执行前（D1、D0）=16#2E0F9000（正值），当 X0.0 从 OFF→0N 时，（D1、D0）的数据以单精度浮点数做正负符号的反相运作，因此执行后（D1、D0）=16#AE0F9000（负值）。



API	功能码			操作数	功能
0212		FBCD	P	**S**，**D**	单精度浮点数→十进浮点数

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●		●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

S ： 数据源

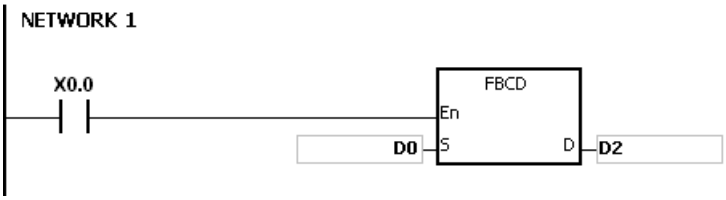
D ： 变换的结果

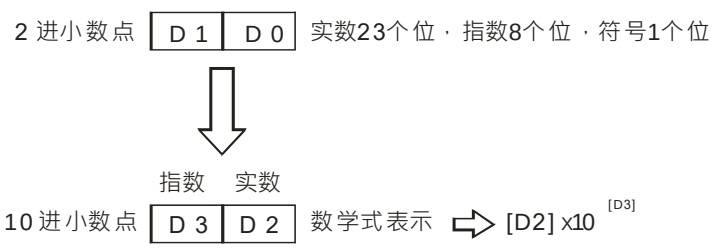
指令说明：

1. 将 **S** 所指定的寄存器中的单精度浮点数变换成十进制浮点数寄存于 **D** 所指定的寄存器当中。
2. PLC 是以单精度浮点数类型作浮点数运算的依据。FBCD 指令就是用来将单精度浮点数变换成十进制浮点数类型的专用指令。
3. 标志：SM600 为零标志，SM601 为借位标志，SM602 为进位标志。
   若转换结果的绝对值大于可表示的最大浮点值，则进位标志 SM602=ON。
   若转换结果的绝对值小于可表示的最小浮点值，则借位标志 SM601=ON。
   若转换结果为 0，则零标志 SM600=ON。

程序范例：

当 X0.0= ON 时（D1、D0）内的单精度浮点数被转换成十进制浮点数存于（D3、D2）。





补充说明：

当 **S** 内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	功能码			操作数	功能
0213		FBIN	P	**S**，**D**	十进浮点数→单精度浮点数

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●		●					
D		●			●	●	●	●			●					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

S ： 数据源

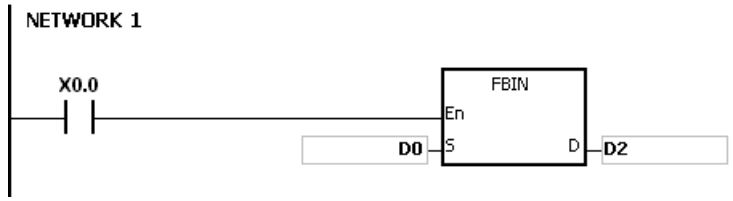
D ： 变换的结果

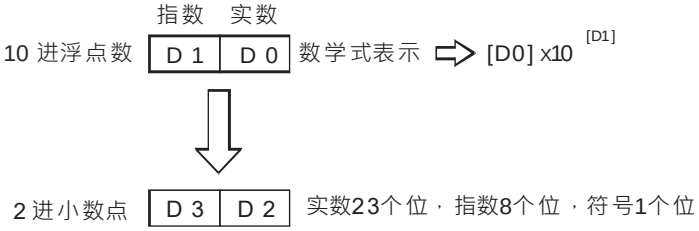
指令说明：

1. 将 **S** 所指定的寄存器中的十进制浮点数变换成单精度浮点数并寄存于 **D** 所指定的寄存器当中。
2. 例如：**S**=1234，**S**+1=3 将变换成 **S**=1.234x10^6。
3. **D** 必须是单精度浮点数形式，**S** 和 **S**+1 中分别用十进制表示实数和指数。
4. FBIN 指令就是用来将十进浮点数变换成单精度浮点数类型的专用指令。
5. 十进浮点数实数范围为-9,999~+9,999，指数范围为-41~+35，实际 PLC 十进浮点数的范围为±1175×10-41 到±3402×10+35。若运算结果为 0，则零标志 SM600=ON。

程序范例一：

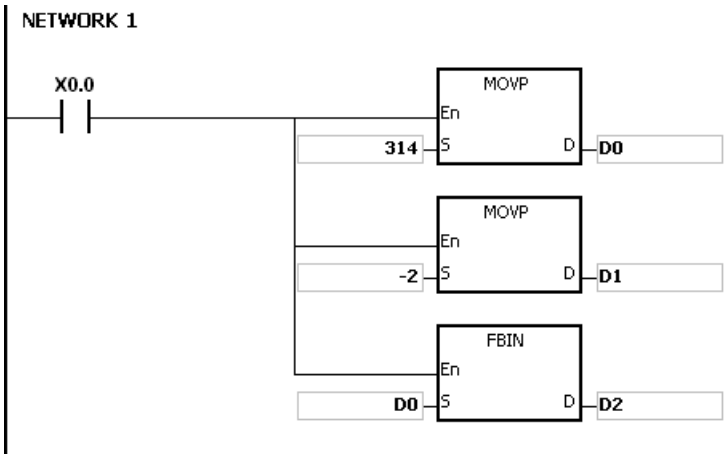
当 X0.0=ON 时，指定寄存器（D1、D0）中的十进制浮点数转换成单精度浮点数并寄存于（D3、D2）中。





程序范例二：

1. 在进行浮点数运算前必须适用 FLT 指令 BIN 整数变换成单精度浮点数 ,变换的前提是被变换值必须是 BIN 整数，然而，FBIN 指令可将浮点数变换成单精度浮点数。
2. 当 X0.0=ON 时，将 K314 搬移到 D0，将 K-2 搬移到 D1，组成十进制浮点数类型（3.14=314×10^{-2}）。



补充说明：

若 **S** 操作数的十进浮点数实数范围不在-9,999~+9,999 之间或指数范围不在-41~+35 之间，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	功能码			操作数	功能
0214		BKBCD	P	**S**，**n**，**D**	连续区块 BIN→BCD 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●							
n	●	●			●	●		●	●				○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 数据源

n ： 欲转换的数据数

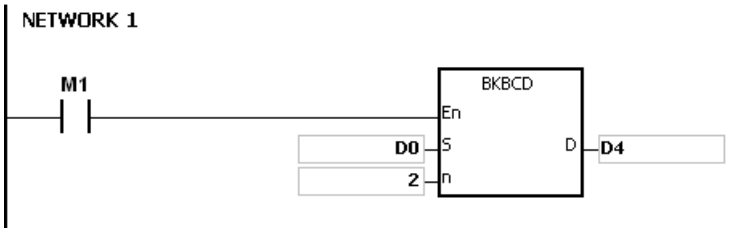
D ： 变换的结果

指令说明：

1. 将 n 笔的数据源 **S** 的内容（BIN 值）作 BCD 的转换，存于 **D**。
2. **n**=1~256。

程序范例：

当 M1=ON 时，将 D0、D1 连续两笔数据中的 BIN 转换成 BCD 值并存于 D4、D5 中。



补充说明：

1. 若 **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. **S+n-1**，**D+n-1**，装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. 在 BCD 变换结果若超过 0~9,999，指令不执行，SM0=ON，SR0 记录错误码 16#200D。（BCD 值以 HEX 表示有任一位数不在 0~9 的范围内）。
4. 若 **S~S+n-1** 的装置与 **D~D+n-1** 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。

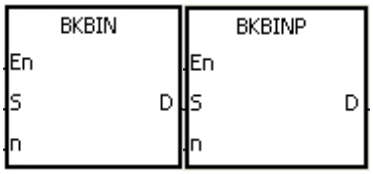
API	功能码		操作数	功能
0215	BKBIN	P	**S**、**n**、**D**	连续区块 BCD→BIN 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●		●	●							
n	●	●			●	●		●	●				○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 数据源

n ： 欲转换的数据数

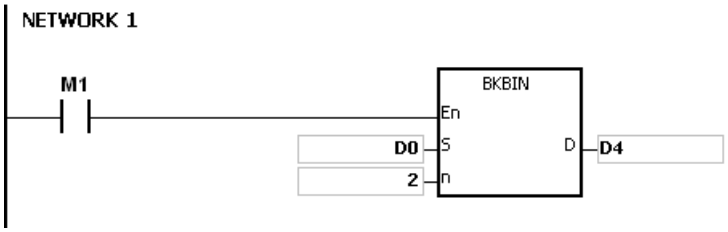
D ： 变换的结果

指令说明：

1. 将 n 笔的数据源 **S** 的内容（BCD：0~9,999）作 BIN 的转换，存于 **D**。
2. 数据源 **S** 的内容有效数值范围：BCD（0~9,999）。
3. **n**=1~256。

程序范例：

当 M1=ON 时，将 D0、D1 连续两笔数据中的 BCD 转换成 BIN 并存于 D4、D5 中。



补充说明：

1. 若 **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. **S+n-1**，**D+n-1**，装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. 当 **S** 的数据内容并非为 BCD 值 (以 Hex 表示有任一位数不在 0~9 的范围内)，则将会产生运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#200D。
4. 若 **S~S+n-1** 的装置与 **D~D+n-1** 的装置有重叠时，指令不执行，SM0=ON，错误码 SR0=16#200C。

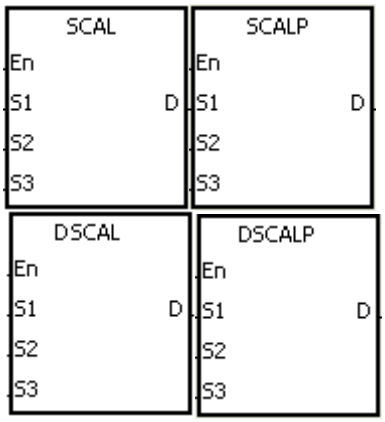
API	功能码			操作数	功能
0216		SCAL	P	**S_1，S_2，S_3，D**	比例值运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2	●	●			●	●		●	●		○	○	○	○		
S_3	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
S_3		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 数值数据源

S_2 ： 斜率

S_3 ： 偏移量

D ： 目的地装置

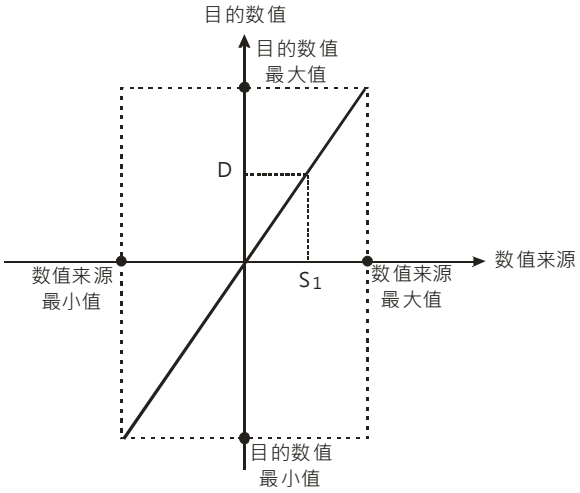
指令说明：

1. 指令内部运算公式为 **D**=（**S_1**×**S_2**）÷1,000+**S_3**。
2. 其中 **S_2** 与 **S_3** 的数值须由用户依下列斜率与偏移量公式先行运算，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–目的数值最小值）÷（数值来源最大值–数值来源最小值）]×1,000

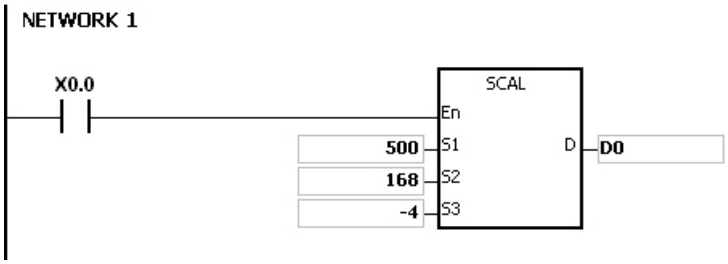
   偏移量公式为 **S_3**=目的数值最小值–数值来源最小值×**S_2**÷1,000

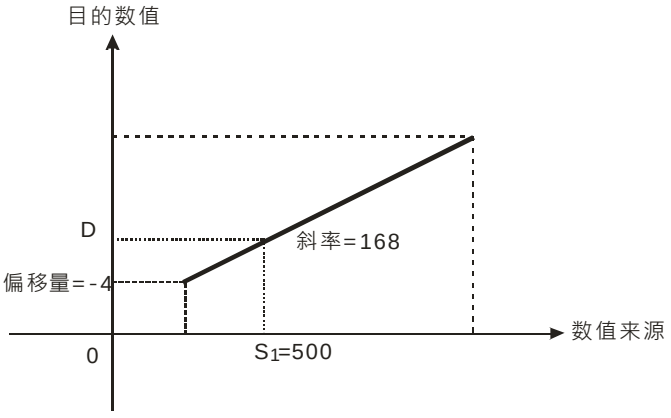
   其输出曲线将如下图所示：



程序范例一：

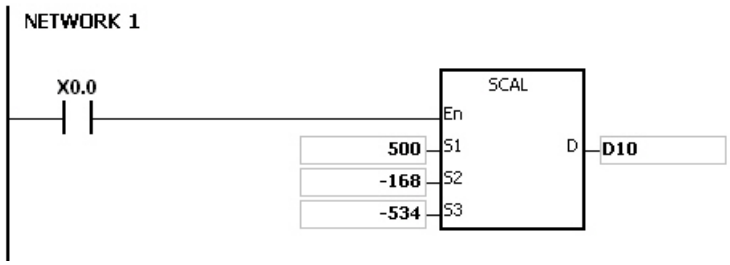
1. 已知 **S_1** 数值数据源为 500，**S_2** 斜率为 168，**S_3** 偏移量为 -4，当 X0.0=ON 时，SCAL 指令执行，可在 D0 得到所要求的比例值。
2. 运算方式：D0=（500×168）÷1,000+（-4）=80

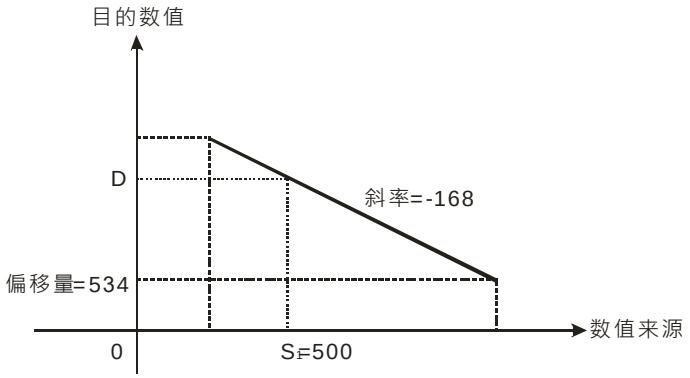




程序范例二：

1. 已知 **S_1** 数值数据源为 500，**S_2** 斜率为-168，**S_3** 偏移量为 534，当 X0.0=ON 时，SCAL 指令执行，可在 D10 得到所要求的比例值。
2. 运算方式：D10=（500×-168）÷1,000+534=450





补充说明：

1. 此 SCAL 指令适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。
2. 16 位指令中，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。
3. 32 位指令中，输入参数 **S_2** 时，其输入数值必须为-2,147,483,648~2,147,483,647 之间的数值（实际数值为-2,147,483,648~2,147,483,647），若是 **S_2** 实际数值超过范围时，请改用 SCLP 指令运算。
4. 用户运用斜率换算公式时，须注意数值来源最大值，必须大于数值来源最小值，而目的数值最大值，并不限制大于目的数值最小值。
5. 16 位指令中，若 **D** 的值>32,767 则 **D**=32,767，若 **D** 的值<-32,768 则 **D**=-32,768。
6. 32 位指令中，若 **D** 的值>2,147,483,647 则 D=2,147,483,647，若 **D** 的值<-2,147,483,648 则 **D**=-2,147,483,648。

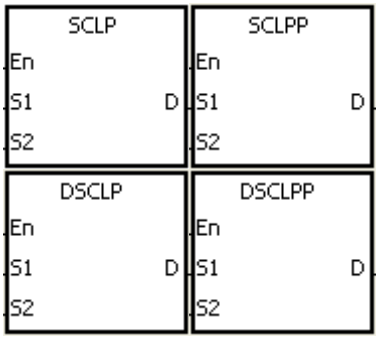
API	功能码			操作数	功能
0217	D	SCLP	P	**S_1 · S_2 · S_3 · D**	参数型比例值运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		○
S_2	●	●			●	●	●	●	●							
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●		●				
S_2		●	●			●	●		●				
D		●	●			●	●		●				

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 数值数据源

S_2 ： 参数

D ： 目的地装置

指令说明：

1. 32 位指令才可以使用 HC 装置，不可使用 E 装置。

2. **S_1** 操作数常数限制使用如下：SM685=ON（采用浮点数做运算）

常数	16 位指令	32 位指令	
		SM685 ON	SM685 OFF
K	○	X	○
16#	○	X	○
F	X	○	X

SM685 标志（是否使用浮点数做运算）仅在 32 位指令有效。

3. 16 位指令 **S_2** 参数设定内容如下：

装置编号	参数名称与说明	设定范围
S_2	数值来源最大值	-32,768~32,767
S_2+1	数值来源最小值	-32,768~32,767
S_2+2	目的数值最大值	-32,768~32,767

装置编号	参数名称与说明	设定范围
S_2+3	目的数值最小值	-32,768~32,767

4. 16 位指令 S_2 操作数将连续占用 4 个装置。

5. 32 位指令 S_2 参数设定内容如下：

装置编号	参数名称与说明	设定范围	
		整数	浮点数
S_2、S_2+1	数值来源最大值	-2,147,483,648~ 2,147,483,647	32 位浮点数范围
S_2+2、3	数值来源最小值		
S_2+4、5	目的数值最大值		
S_2+6、7	目的数值最小值		

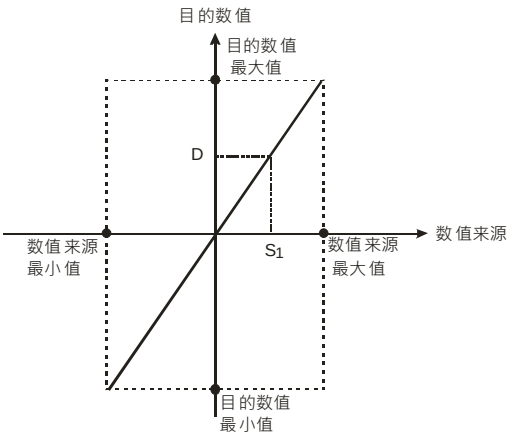
6. 32 位指令 S_2 操作数将连续占用 8 个装置。

7. 标志 SM685 为十进制整数与单精度浮点数使用标志。在 32 位指令中若要使用浮点数做运算可以设定 SM685=ON，若要使用十进制整数做运算可以设定 SM685=OFF。

8. 指令内部运算公式为 D=[（S_1–数值来源最小值）×（目的数值最大值–目的数值最小值）]÷（数值来源最大值–数值来源最小值）+目的数值最小值。

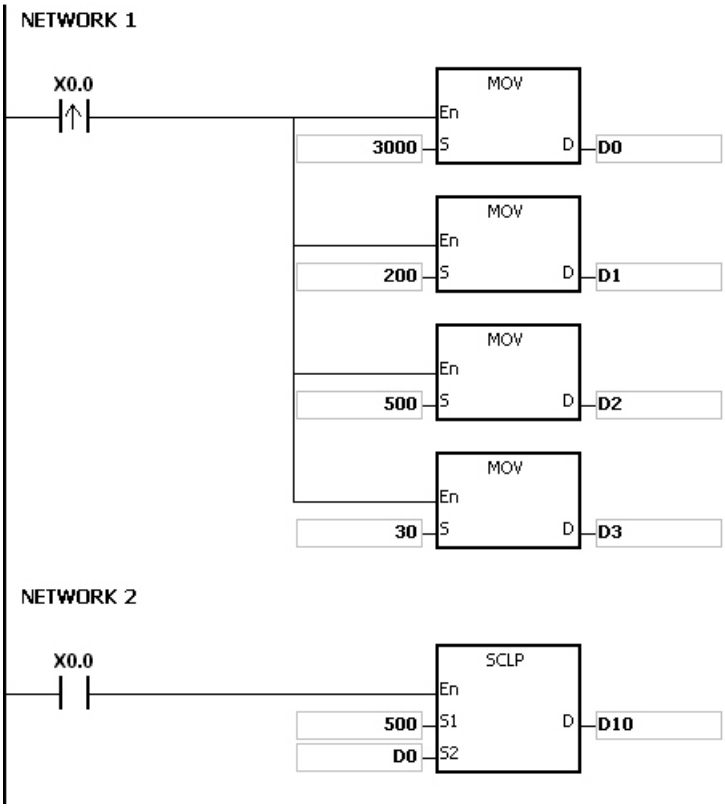
9. 来源数值和目的数值运算关系：
   y=kx+b
   y=目的数值（D）
   k=斜率 =（目的数值最大值–最小值）÷（来源数值最大值–最小值）
   x=来源数值（S_1）
   b=偏移量=目的数值最小值–来源数值最小值×斜率
   将上面的各参数带入公式 y=kx+b，即可推导得出指令内部运算公式：
   y=kx+b=D=kS_1+b=斜率×S_1 + 偏移量 = 斜率×S_1 + 目的数值最小值–来源数值最小值×斜率=斜率×（S_1–来源数值最小值）+ 目的数值最小值=（S_1–来源数值最小值）×（目的数值最大值–目的数值最小值）÷（来源数值最大值–来源数值最小值）+ 目的数值最小值。

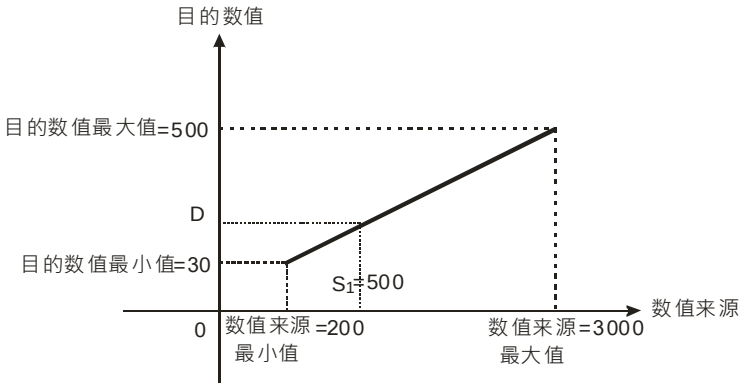
10. 假如 S_1>数值来源最大值，则 S_1=数值来源最大值；假如 S_1<数值来源最小值，则 S_1=数值来源最小值；当输入数值与参数设定完成后，则其输出曲线将如下图所示。



程序范例一：

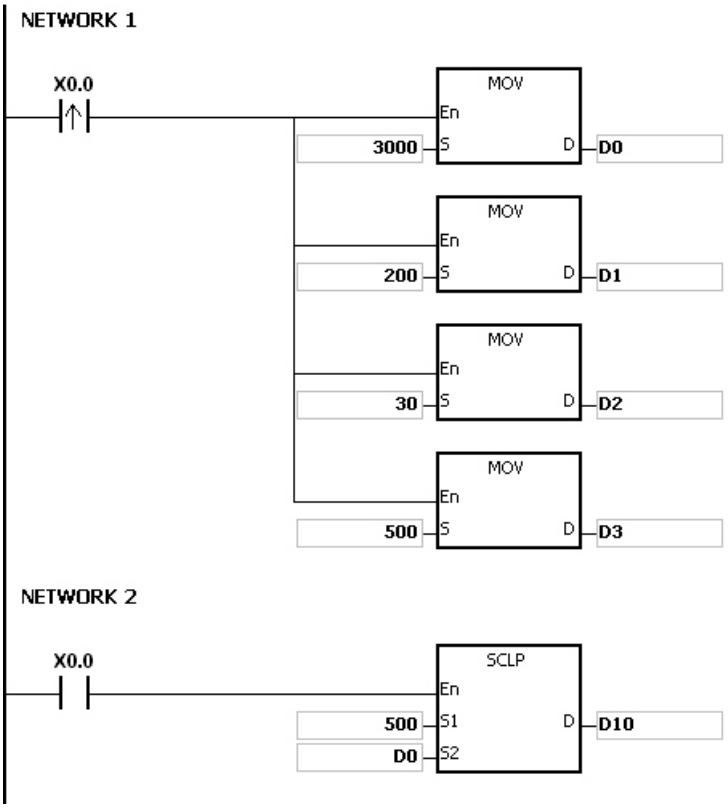
1. 已知 S_1 数值数据源为 500，数值来源最大值 D0=3,000，数值来源最小值 D1=200，目的数值最大值 D2=500，目的数值最小值 D3=30 当 X0.0=ON 时，SCLP 指令执行，可在 D10 得到所要求的比例值。
2. 运算方式：D10=[（500–200）×（500–30）]÷（3,000–200）+30=80.35，取整数，D10=80。

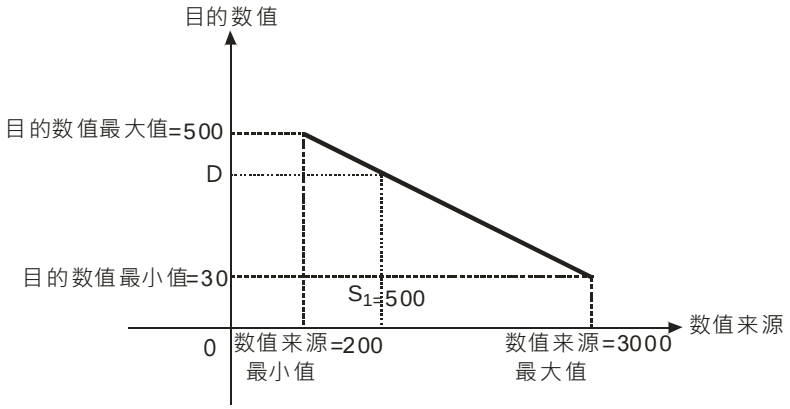




程序范例二：

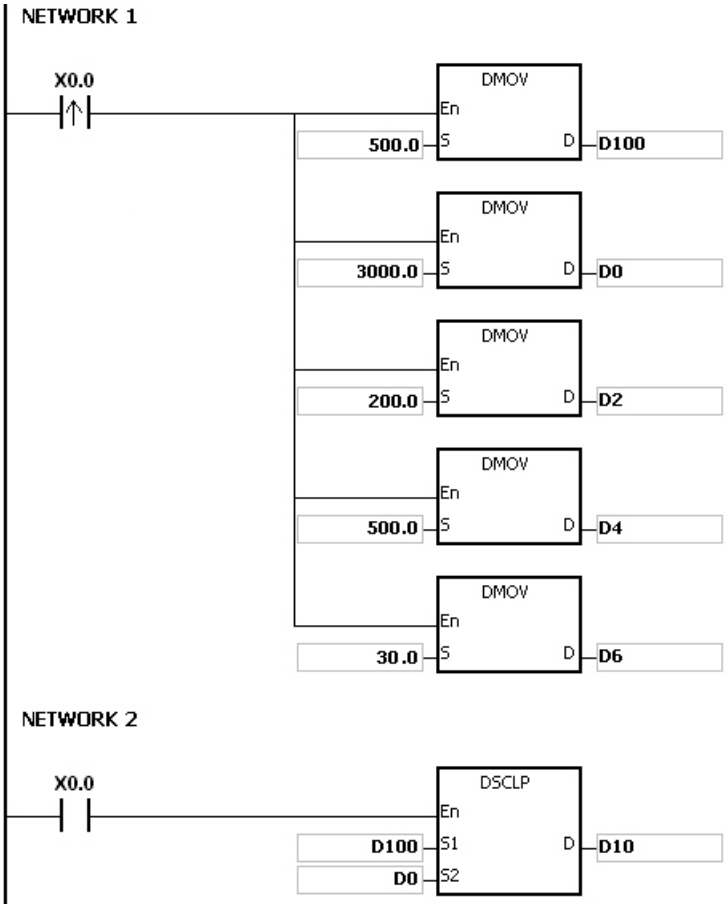
1. 已知S_1数值数据源为500 数值来源最大值=D0=3,000 数值来源最小值D1=200 目的数值最大值D2=30，目的数值最小值 D3=500 当 X0.0=ON 时，SCLP 指令执行，可在 D10 得到所要求的比例值。
2. 运算方式：D10=[（500–200）×（30–500）]÷（3,000–200）+500=449.64。四舍五入取整数，D10=450。

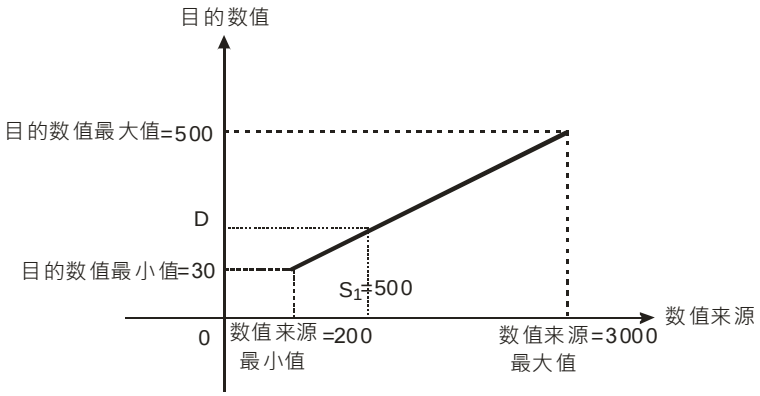




程序范例三：

1. 已知 **S_1** 数值数据源 D100 为 500.0，数值来源最大值 D0 为 3000.0，数值来源最小值 D2 为 200.0，目的数值最大值 D4 为 500.0，目的数值最小值 D6 为 30.0 当 X0.0=ON 时，SET SM685，使用浮点数运算且 DSCLP 指令执行，可在 D10 得到所要求的比例值。
2. 运算方式：D10=[(500.0–200.0)×(500.0–30.0)]÷(3000.0–200.0) +30.0=80.35，取整数，D10=80.0。





补充说明：

1. 16 位 **S_1** 操作数数值设定范围：数值来源最大值≧**S_1**≧数值来源最小值，-32,768~32,767 超出边界值以边界值运算。
2. 32 位 **S_1** 整数操作数数值设定范围：数值来源最大值≧**S_1**≧数值来源最小值，-2,147,483,648~2,147,483,647 超出边界值以边界值运算。
3. 32 位 **S_1** 浮点数操作数数值设定范围：数值来源最大值≧**S_1**≧数值来源最小值，依单精度浮点数范围，超出边界值以边界值运算。
4. 用户运用时，须注意数值来源最大值，必须大于数值来源最小值，而目的数值最大值，并不限制大于目的数值最小值。

5. 数值来源最大值等于数值来源最小值，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2012。
6. 16 位指令 **S_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of WORD。
7. 32 位指令 **S_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of DWORD。

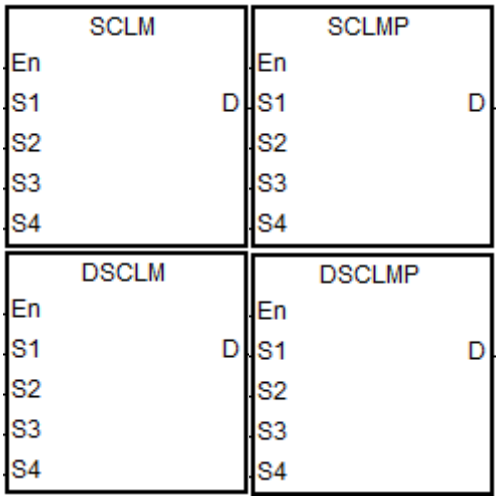
API	功能码			操作数	功能
0222	D	SCLM	P	**S_1 · S_2 · S_3 · S_4 · D**	多点式区域比例运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1					●	●	●	●								
S_2					●	●	●	●					○	○		
S_3					●	●	●	●	●							
S_4					●	●	●	●	●							
D					●	●	●	●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●		●				
S_2		●	●			●	●						
S_3		●	●			●	●						
S_4		●	●			●	●						
D		●	●			●	●		●				

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 数值数据源

S_2 ： 指定多点区域数量

S_3 ： 多点区域比较数值

S_4 ： 转换后对应之基准值

D ： 比例转换后结果

指令说明：

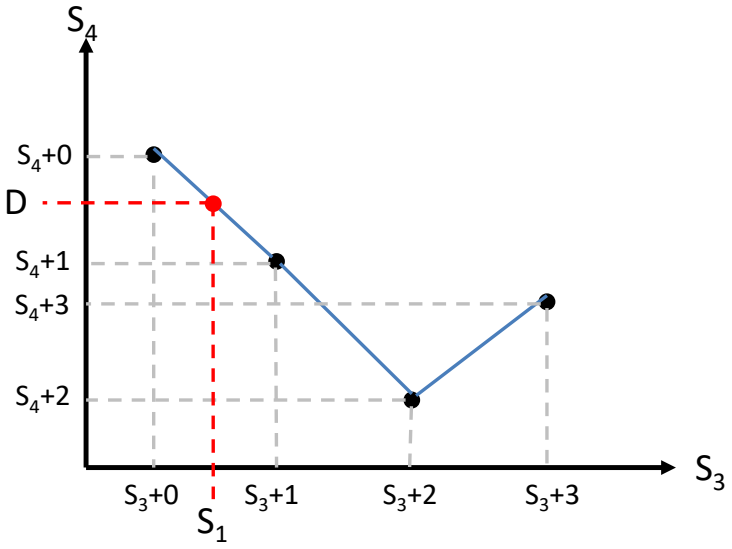
1. 32 位指令才可以使用 HC 装置，不可使用 E 装置。AS300 型主机韧体需为 V1.04.00 版（含）以上支持。

2. **S_1**、**S_3** 与 **S_4** 操作数数值类型使用说明如下：(○为可使用，X 为不可使用)

常数	16 位指令	32 位指令	
		SM685 ON	SM685 OFF
K	○	X	○
16#	○	X	○
F	X	○	X

备注：SM685=ON（采用浮点数做运算）仅在 32 位指令有效。

3. **S_1** 为来源数值；**S_2** 为指定多点区域之数量，数值范围为 2~50 之间，超出此范围时，自动以最小或最大值执行；**S_3** 为多点区域比较设定值；**S_4** 为多点区域数值转换后对应之基准值；举例：当 **S_2** 指定区域数量设为 10，则 **S_3** ~ **S_3**+9 为 10 个区域比较数值，**S_4** ~ **S_4**+9 为 10 个对应后之转换基准值。
4. 多点区域比较顺序为 0，1，2 …**S_2**–1，比较规则为 **S_1** >= **S_3**+0，且 **S_1** < **S_3**+1；若 **S_1** 数值不在此区域内，则往下一个比较区域进行，如：**S_1** >= **S_3**+1，且 **S_1** < **S_3**+2；直到比对次数已达 **S_2** – 1 次。
5. 此指令的比对区域方式为由小到大，因此请设计 **S_3** 区域比较数值也是由小到大的顺序。
6. 多点区域数值转换方式如下图所示：(假设 **S_2** 设定区域数为 4 个)



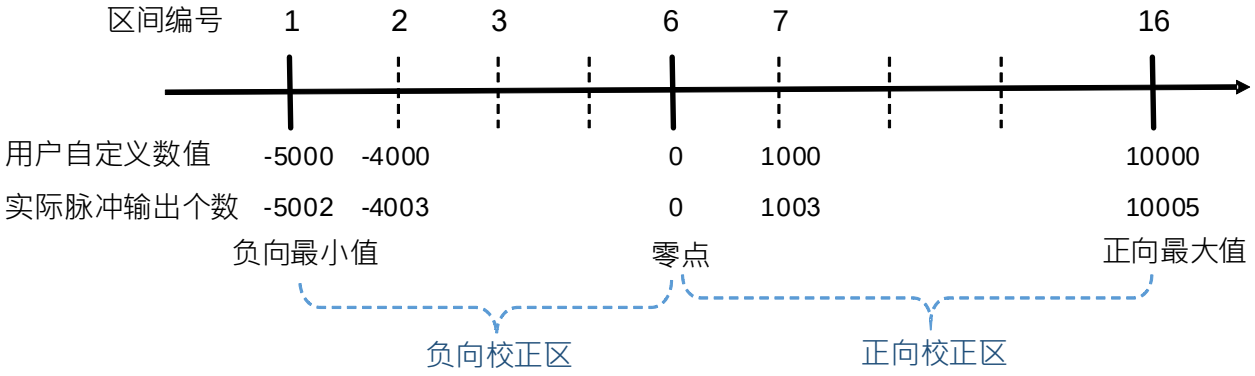
7. 假设 **S_1** 数值位于 **S_3**+0 与 **S_3**+1 之间，则数值转换之数学公式为 **D** = ((**S_1** – **S_3**+0) x (**S_4**+1 – **S_4**+0) / (**S_3**+1 – **S_3**+0)) + **S_4**+0
8. 当 **S_1** 数值比对不在任何一个指定区域内时，**D** 的执行结果说明如下：

   当 **S_1** 数值>最后一个指定区域时，**D** 结果将储存 **S_4** 最后一个转换基准值；例如：在第 6 点范例中，**S_1** 数值>**S_3**+3 数值时，则 **D**=**S_4**+3。

   当 **S_1** 数值<第一个指定区域时，**D** 结果将储存 **S_4** 第一个转换基准值；例如：在第 6 点范例中，**S_1** 数值<**S_3**+0 数值时，则 **D**=**S_4**+0。
9. 16 位指令 **S_3** 与 **S_4** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [**S_2**] of WORD。
10. 32 位指令 **S_3** 与 **S_4** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [**S_2**] of DWORD (SM685=OFF) 或者 ARRAY [**S_2**] of REAL（SM685=ON）。
11. 主机韧体版本 V1.12.10（含）以上，DSCLM 指令（32 位指令）支持启动丝杆螺距校正功能。透过此功能，可以修正因丝杆螺距的公差造成丝杆移动的物理量与脉冲输出个数不匹配的情况。

    应用操作步骤如下：

    A. 实际量测丝杆各段螺距与脉冲输输出个数的对应，填入指令的 S3，S4 操作数。

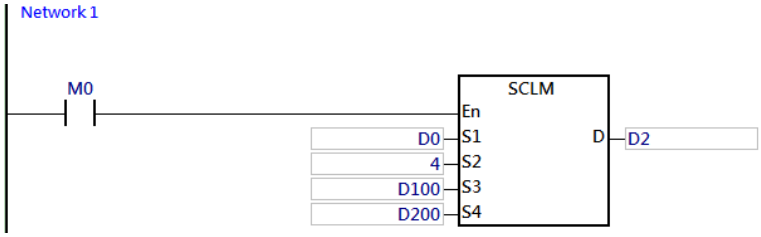


校正区间	目标位置数据 （S3 操作数，ex：D100）	输出轴实际输出个数 （S4 操作数，ex：D200）
1	-5000（D100，D101）	-5002（D200，D201）
2	-4000（D102，D103）	-4003（D202，D203）
. .		
6	0（D110，D111）	0（D210，D211）
7	1000（D112，D113）	1003（D212，D213）
. .		
16	10000（D130，D131）	10005（D230，D231）

B. S2 操作数的下 8 位为指定校正区间数，最大 50 个区间。S2 操作数的上 8 位为指定要启动校正功能的输出轴编号，输入数值 1~12 分别表示 Y0.0~Y0.11 输出点，输入数值为 0，表示执行原本数值转换功能。当执行校正功能，S1 和 D 操作数无功用。

C. 启动 DSCLM 指令 EN 引脚，执行校正功能，输出轴定位指令会依校正参数，调整实际脉冲输出个数；关闭 DSCLM 指令，即关闭校正功能。

程序范例：



给定多点区域比较值 S_3 如下：

装置	D100	D101	D102	D103
内容值	100	200	300	400

给定转换后对应之基准值 S_4 如下：

装置	D200	D201	D202	D203
内容值	4000	3000	1500	2000

根据数值来源（D0）做比例转换后的结果（D2）说明如下：

给定 D0=10

因为 D0<D100（第一个区段），所以 D2=D200=4000（第一个转换基准值）

给定 D0=K150

D0 落在（D100，D101）=（100，200）之间且对应的基准值为（D200，D201）=（4000，3000），所以

D2=（150-100）*（3000-4000）/（200-100）+4000=3500

给定 D0=450

因为 D0>D103（最后一个区段），所以 D2=D203=2000（最后一个转换基准值）

给定 D0=K250

D0 落在（D101，D102）=（200，300）之间且对应的基准值为（D201，D202）=（3000，1500），所以

D2=（250-200）*（1500-3000）/（300-200）+3000=2250

给定 D0=K350

D0 落在（D102，D103）=（300，400）之间且对应的基准值为（D202，D203）=（1500，2000），所以

D2=（350-300）*（2000-1500）/（400-300）+1500=1750

6.4 数据转移指令

6.4.1 数据转移指令一览表

API	指令码（位）		P 指令	功能
	16	32		
0300	MOV	DMOV	✓	数据移动
0302	$MOV	–	✓	字符串移动
0303	CML	DCML	✓	反转传送
0304	BMOV	DBMOV	✓	全部传送
0305	NMOV	DNMOV	✓	多点移动
0306	XCH	DXCH	✓	数据的交换
0307	BXCH	–	✓	全部交换
0308	SWAP	DSWAP	✓	上/下 BYTE 变换
0309	SMOV	–	✓	位数移动
0310	MOVB	–	✓	多位移动

6.4.2 数据转移指令说明

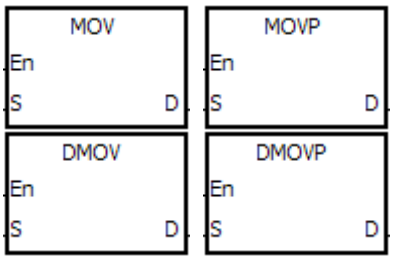
API	指令码			操作数	功能
0300	D	MOV	P	**S**、**D**	数据移动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		○
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●		●		●	●	
D		●	●			●	●		●		●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 数据源

D ： 数据目的地

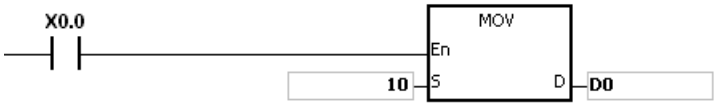
指令说明：

1. 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。
2. 32 位 DMOV 指令 **S** 才可使用 F 浮点数。
3. 32 位 DMOV 指令才可使用 HC 装置，但不可使用 E 装置。

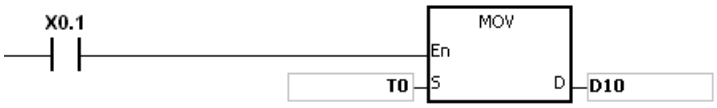
程序范例：

1. 16 位数据搬移，须使用 MOV 指令。
   - 当 X0.0=OFF 时，D0 内容没有变化，若 X0.0=ON 时，将数值 10 传送至 D0 数据寄存器内。
   - 当 X0.1=OFF 时，D10 内容没有变化，若 X0.1=ON 时，将 T0 当前值传送至 D10 数据寄存器内。
2. 32 位数据搬移，须使用 DMOV 指令。
   - 当 X0.2=OFF 时，(D31、D30)、(D41、D40) 内容没有变化，若 X0.2=ON 时，将 (D21、D20) 当前值传送至 (D31、D30) 数据寄存器内。同时，将 HC0 当前值传送至 (D41、D40) 数据寄存器内。
3. F 浮点数搬移，须使用 DMOV 指令。
   - 当 X0.3=OFF 时，(D51、D50) 内容没有变化，若 X0.3=ON 时，将浮点数 3.450 转换为二进浮点值传送至 (D51、D50) 数据寄存器内。

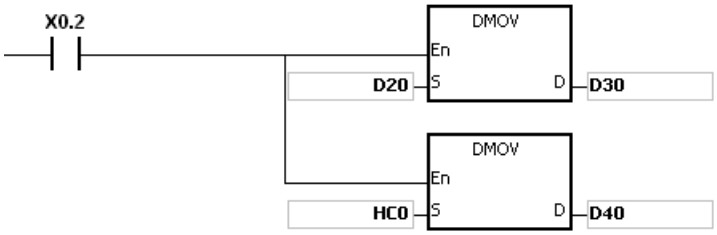
NETWORK 1



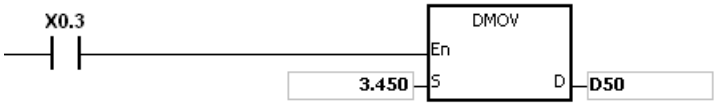
NETWORK 2



NETWORK 3



NETWORK 4



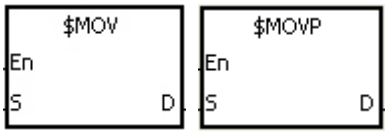
API	指令码			操作数	功能
0302		$MOV	P	**S**，**D**	字符串移动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●						○	
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													●
D													●

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

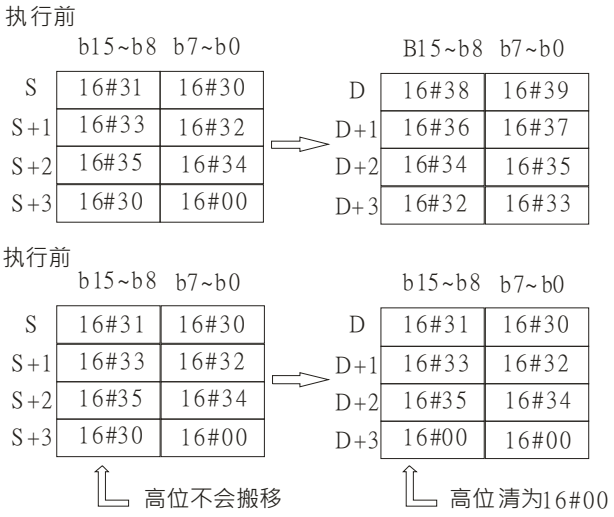


S ： 数据源

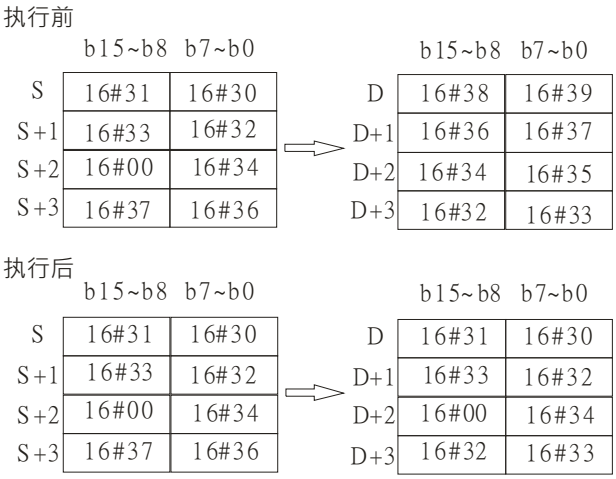
D ： 数据搬移目的地

指令说明：

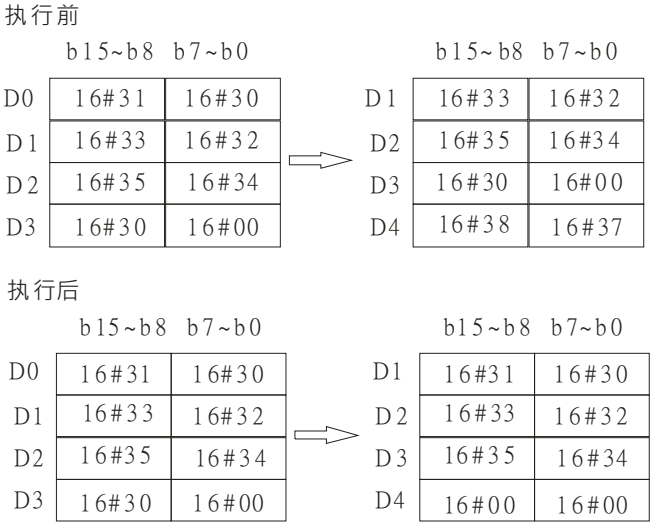
1. 若 **S** 操作数为字符串且该指令执行时，将 **S** 操作数的字符串直接搬移至 **D** 操作数内，并且结尾会自动补上结后缀（16#00）。
2. 若 **S** 操作数要做搬移的不是字符串（$）时，请注意在最后面一定得要有一个结后缀（16#00），且最多 256 个字（包含结束符号 16#00）。
3. 当 **S** 检查连续 256 个字内，或者超出装置范围时，都没找出 16#00 当结尾时，指令不执行。SM0=ON，错误码 SR0=16#200E。
4. 若 **S** 操作数不是字符串，当该指令执行时，将 **S~**16#00 之间的字符串直接搬移至 **D** 内（含结后缀（16#00））。当指令不执行时，**D** 内容不会变化。
5. 当 **D** 装置空间不足以存放 **S** 的字符串长度时，指令不执行，SM0=ON，错误码 SR0=16#2003。
6. 若 **S** 操作数不是字符串且该指令执行时，第一个字符就是结后缀（16#00），则还是会搬移结后缀。
7. 当 16#00 字符串结尾在低位时，执行结果如下：



8. 当 16#00 字符串结尾在高位时，执行结果如下：

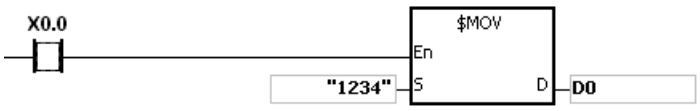


9. 当 **S** 跟 **D** 装置重叠且 **S<D** 时，搬移的顺序从结尾 16#00 开始搬移，一直搬到 **D**，不会发生 **S** 被覆盖的情形。



程序范例一：

若 **S** 操作数为偶数字符串“1234”时，当条件接点 X0.0 启动后，D0~D3 的值如下：



S 操作数

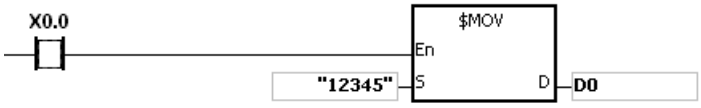
字符串	‘1’	‘2’	‘3’	‘4’
HEX	16#31	16#32	16#33	16#34

执行后，**D** 操作数

装置	**上 8 位**	**下 8 位**	**备注**
D0	16#32	16#31	‘1’=16#31，‘2’=16#32
D1	16#34	16#33	‘3’=16#33，‘4’=16#34
D2	16#00	16#00	下 8 位 16#00 为终止符， 上 8 位 16#00 为自动补零
D3	不变	不变	

程序范例二：

若 **S** 操作数为奇数字符串“12345”时，当条件接点 X0.0 启动后，D0~D3 的值如下：



S 操作数

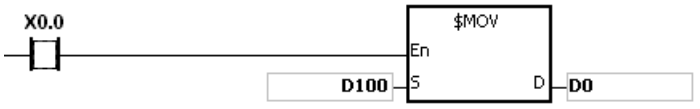
字符串	‘1’	‘2’	‘3’	‘4’	‘5’
HEX	16#31	16#32	16#33	16#34	16#35

执行后，**D** 操作数

装置	**上 8 位**	**下 8 位**	**备注**
D0	16#32	16#31	‘1’=16#31，‘2’=16#32
D1	16#34	16#33	‘3’=16#33，‘4’=16#34
D2	16#00	16#35	上 8 位 16#00 为终止符
D3	不变	不变	

程序范例三：

若 **S** 操作数非字符串且结尾字符 16#00 出现在低位时，执行结果如下：



S 操作数

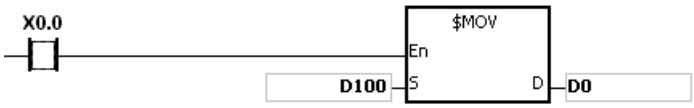
装置	上 8 位	下 8 位	备注
D100	16#31	16#30	‘1’=16#31，‘0’=16#30
D101	16#33	16#32	‘3’=16#33，‘2’=16#32
D102	16#35	16#34	‘5’=16#35，‘4’=16#34
D103	16#30	16#00	‘0’=16#30，16#00 为终止符

执行后，**D** 操作数

装置	上 8 位	下 8 位	备注
D0	16#31	16#30	‘1’=16#31，‘0’=16#30
D1	16#33	16#32	‘3’=16#33，‘2’=16#32
D2	16#35	16#34	‘5’=16#35，‘4’=16#34
D3	16#00	16#00	下 8 位 16#00 为终止符， 上 8 位 16#00 为自动补零
D4	不变	不变	

程序范例四：

若 **S** 操作数非字符串且结尾字符 16#00 出现在高位时，执行结果如下：



S 操作数

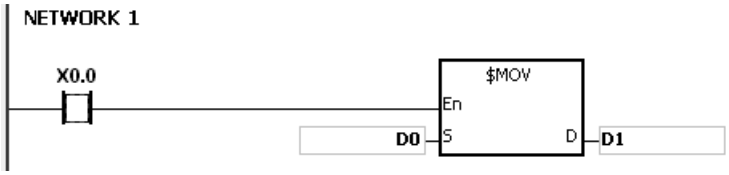
装置	上 8 位	下 8 位	备注
D100	16#31	16#30	‘1’=16#31，‘0’=16#30
D101	16#33	16#32	‘3’=16#33，‘2’=16#32
D102	16#00	16#34	16#00 为终止符，‘4’=16#34
D103	16#37	16#36	‘7’=16#37，‘6’=16#36

执行后，**D** 操作数

装置	上 8 位	下 8 位	备注
D0	16#31	16#30	‘1’=16#31，‘0’=16#30
D1	16#33	16#32	‘3’=16#33，‘2’=16#32
D2	16#00	16#34	16#00 为终止符，‘4’=16#34
D3	不变	不变	

程序范例五：

当 **S** 跟 **D** 装置重叠且 **S<D** 时，搬移的顺序从结尾 16#00 开始搬移，一直搬到 **D**，不会发生 **S** 被覆盖的情形。



S 操作数

装置	上 8 位	下 8 位	备注
D0	16#31	16#30	‘1’=16#31，‘0’=16#30
D1	16#33	16#32	‘3’=16#33，‘2’=16#32
D2	16#35	16#34	‘5’=16#35，‘4’=16#34
D3	16#30	16#00	‘0’=16#30，16#00 为终止符
D4	16#38	16#37	‘8’=16#38，‘7’=16#37

执行后，**D** 操作数

装置	上 8 位	下 8 位	备注
D1	16#31	16#30	‘1’=16#31，‘0’=16#30
D2	16#33	16#32	‘3’=16#33，‘2’=16#32
D3	16#35	16#34	‘5’=16#35，‘4’=16#34
D4	16#00	16#00	下 8 位 16#00 为终止符， 上 8 位 16#00 为自动补零
D5	不变	不变	

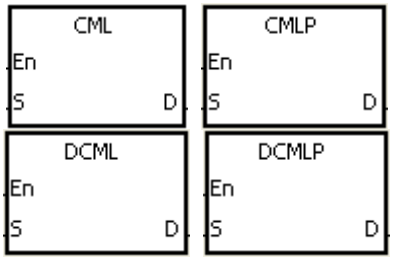
API	指令码			操作数	功能
0303	D	CML	P	**S**、**D**	反转传送

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 数据源

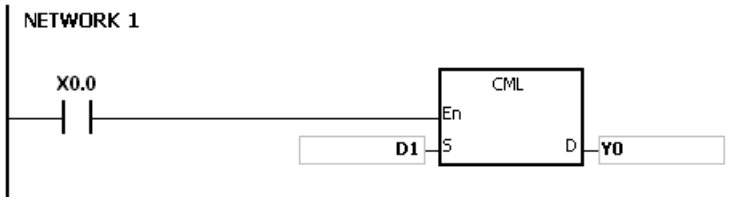
D ： 数据搬移目的地

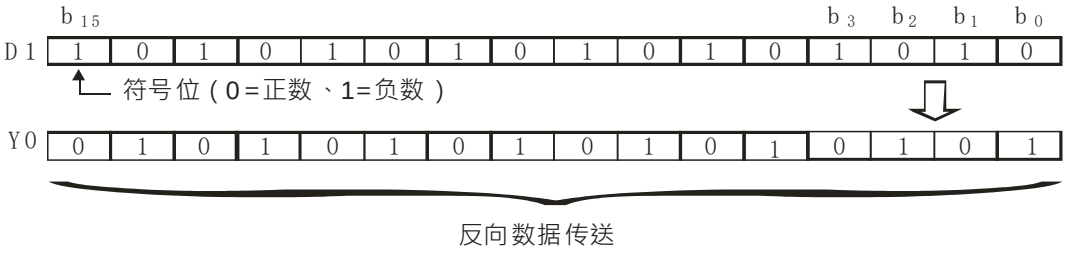
指令说明：

1. 希望作反相输出时，使用本指令。将 **S** 的内容全部反相（0→1、1→0）传送至 **D** 当中。如果内容为常数时，此常数自动被转换成 BIN 值。
2. 32 位指令才可使用 HC 装置，但不可使用 E 装置。

程序范例一：

当 X0.0=ON 时，将 D1 之 b0~b15 内容反相后传送到 Y0.0~Y0.15。

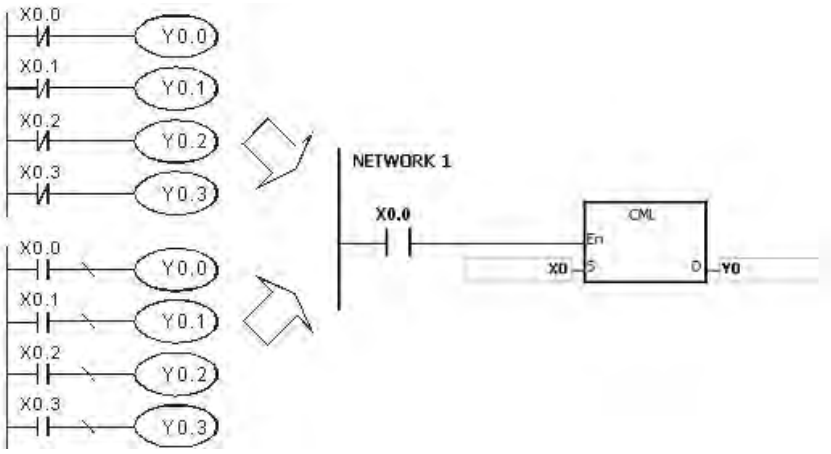




反向数据传送

程序范例二：

下图左边的回路也可以使用 CML 指令来表现，如下图所示



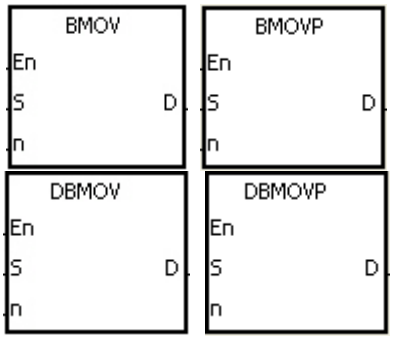
API	指令码			操作数	功能
0304	**D**	BMOV	P	**S**、**D**、**n**	全部传送

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●							
D		●			●	●	●	●								
n	●	●			●	●	●	●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 数据源

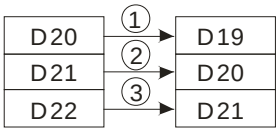
D ： 数据搬移目的地

n ： 传送长度

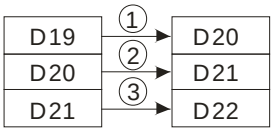
指令说明：

1. 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。
2. **n**=1~256。
3. 32 位指令才可以使用 HC 装置。
4. 为了防止两个操作数所指定传送的号码重叠时，所造成的混乱，请注意两个操作数所指定号码大小的安排，16 位指令说明如下所示：

   当 **S>D** 时，以①→②→③的顺序传送，

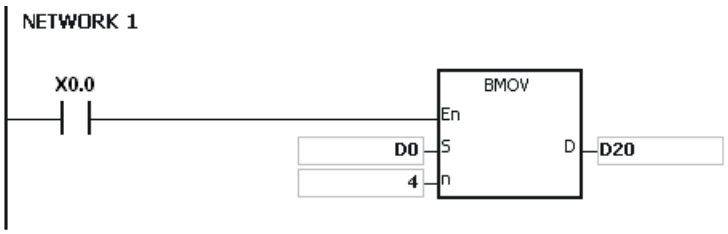


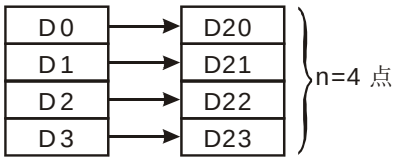
   当 **S<D** 时，是以③→②→①的顺序传送。



程序范例一：

当 X0.0=ON 时，D0~D3 的 4 个寄存器的内容被传送至 D20~D23 的 4 个寄存器内。

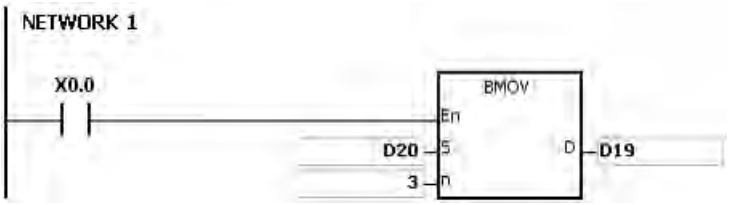


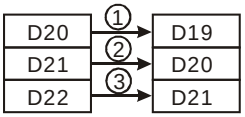


程序范例二：

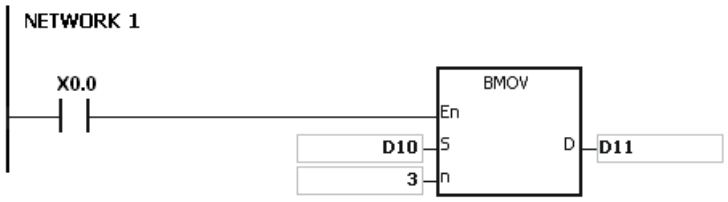
为了防止两个操作数所指定传送的号码重叠时，所造成的混乱，请注意两个操作数所指定号码大小的安排，如下所示：

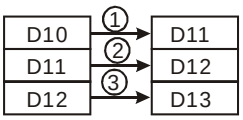
1. 当 **S>D** 时，以①→②→③的顺序传送





2. 当 **S<D** 时，其执行结果以③→②→①的顺序传送。





补充说明：

1. 当 **D**+**n**-1 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 当 **S**+**n**-1 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. 当 **n**>256 或 **n**<1 时，指令不会执行，SM0=ON，错误码 SR0=16#200B。

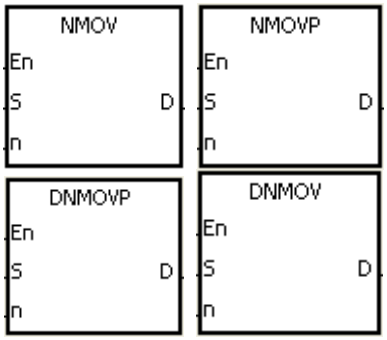
API	指令码			操作数	功能
0305	D	NMOV	P	**S**、**D**、**n**	多点移动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●				○	○		
D		●			●	●	●	●								
n	●	●			●	●	●	●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



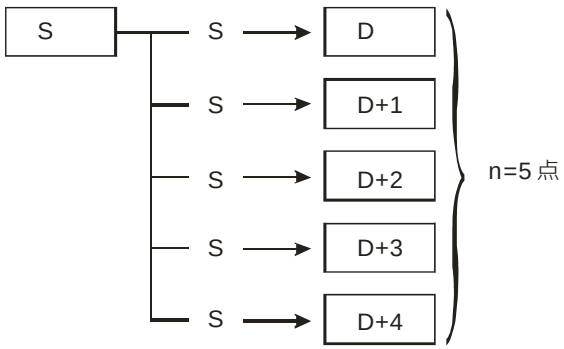
S ： 数据源

D ： 数据搬移目的地

n ： 传送长度

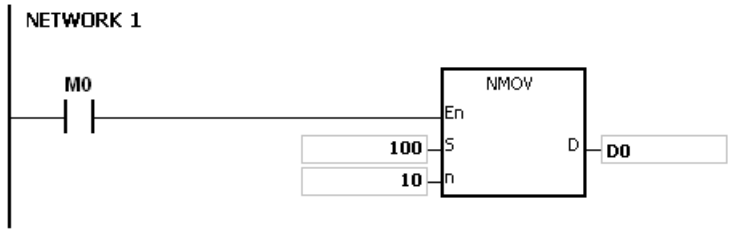
指令说明：

1. 当该指令执行时，将 **S** 的内容直接搬移至 **D** 开始算起的 **n** 个装置中，当指令不执行时，**D** 内容不会变化。
2. 32 位指令才可以使用 HC 装置。
3. NMOV 时 **n**=1~256。



程序范例：

M0 为 ON 时，D0~D9 的值为 100。



补充说明：

1. 当 **D**~**D**+**n**$_{-1}$ 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 当 **n**>256 或 **n**<1 时，指令不会执行，SM0=ON，错误码 SR0=16#200B。

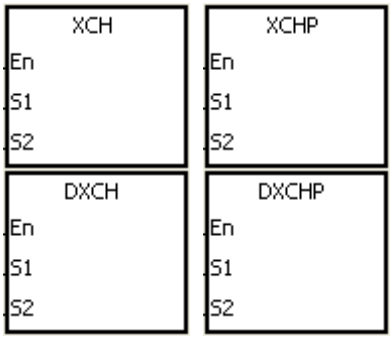
API	指令码			操作数	功能
0306	D	XCH	P	S_1，S_2	数据的交换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1		●			●	●	●	●				○				
S_2		●			●	●	●	●				○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 要交换的数据 1

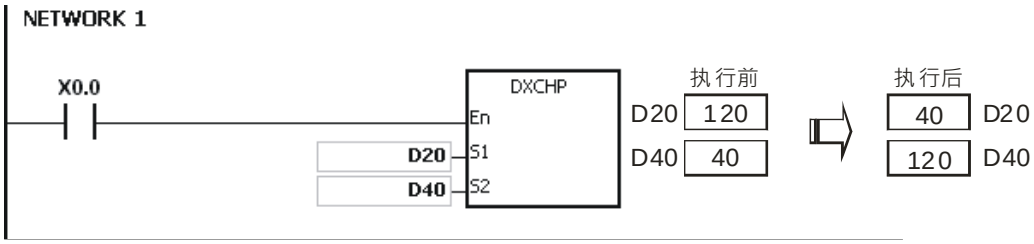
S_2 ： 要交换的数据 2

指令说明：

1. S_1 及 S_2 所指定之装置内容值互相交换。
2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

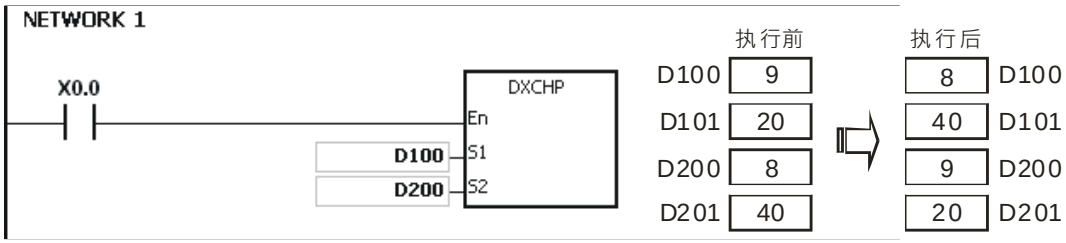
程序范例一：

X0.0=OFF→ON 时，D20 与 D40 的内容互相交换。



程序范例二：

X0.0=OFF→ON 时，D100 与 D200 的内容互相交换。


NETWORK 1
X0.0
DXCHP
En
D100
S1
D200
S2
执行前
执行后
D100 9
D101 20
D200 8
D201 40
8 D100
40 D101
9 D200
20 D201

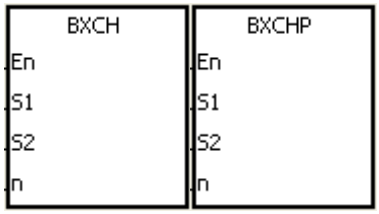
API	指令码			操作数	功能
0307		BXCH	P	**S_1 · S_2 · n**	全部交换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1		●			●	●		●								
S_2		●			●	●		●								
n	●	●			●	●		●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



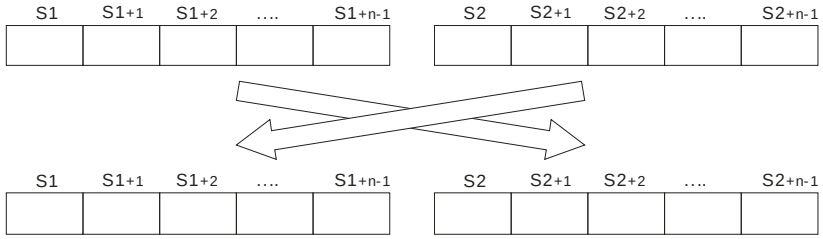
S_1 ： 要交换的数据 1

S_2 ： 要交换的数据 2

n ： 长度

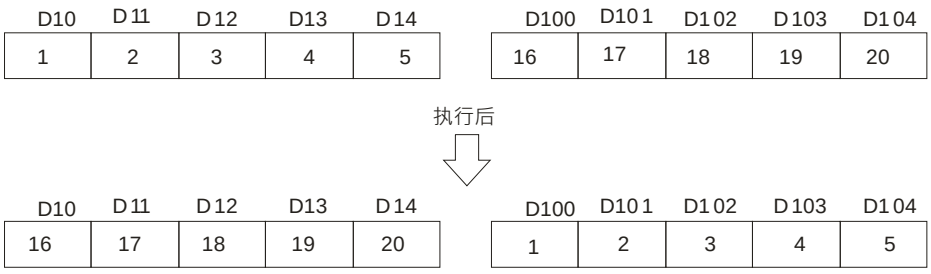
指令说明：

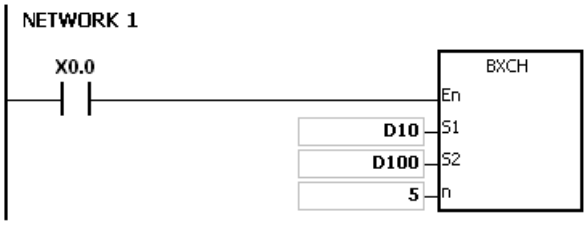
1. **S_1~S_1+n**-1 及 **S_2~S_2+n**-1 所指定之装置内容值互相交换。
2. **n**=1~256。



程序范例：

X0.0=ON 时，D10~D14 的内容与 D100~D104 的内容互相交换。





补充说明：

1. 当 **S_1+n**-1 的范围超出 **S_1** 装置时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 当 **S_2+n**-1 的范围超出 **S_2** 装置时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. 当 **n**>256 或 **n**<1 时，指令不会执行，SM0=ON，错误码 SR0=16#200B。

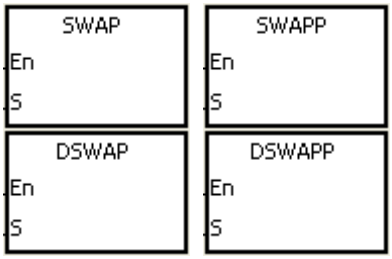
API	指令码			操作数	功能
0308	D	SWAP	P	**S**	上/下 BYTE 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



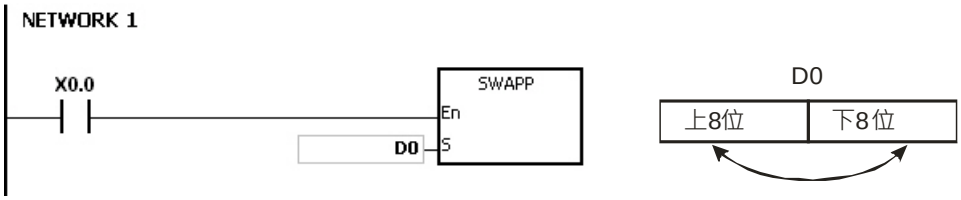
S ：数据源

指令说明：

1. 16 位指令，将 **S** 的上下 8 位的内容互相交换。
2. 32 位指令将 **S** 的上 16 位的上下 8 位内容互相交换，及下 16 位的上下 8 位内容互相交换。
3. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

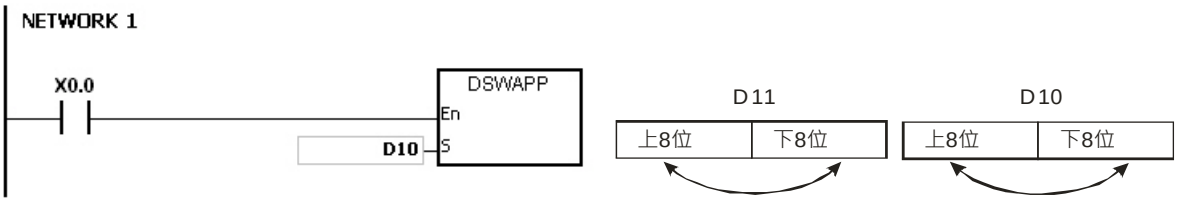
程序范例一：

当 X0.0=ON 时，将 D0 之上位 8 位与下位 8 位的内容互相交换。



程序范例二：

当 X0.0=ON 时，将 D11 之上位 8 位与下位 8 位的内容互相交换，D10 之上位 8 位与下位 8 位的内容互相交换。



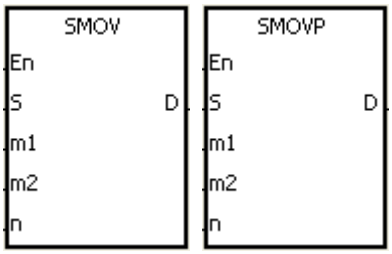
API	指令码			操作数	功能
0309		SMOV	P	**S**，m_1，m_2，**D**，**n**	位数移动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●		●	●				○	○		
m_1	●	●			●	●		●	●				○	○		
m_2	●	●			●	●		●	●				○	○		
D		●			●	●		●								
n	●	●			●	●		●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
m_1		●				●							
m_2		●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 要移动的数据源

m_1 ： 数据源要传送的起始位数

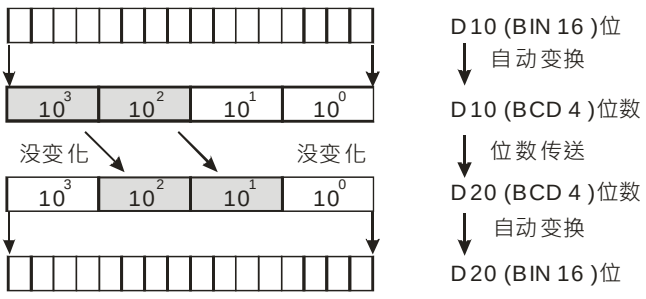
m_2 ： 数据源要传送的位数之个数

D ： 传送之目的地装置

n ： 目的地装置要存放数据源的起始位数

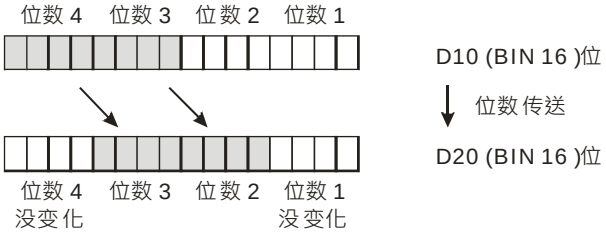
指令说明：

1. 此指令可将数据重新分配或合成。当该指令执行时，指定 **S** 的第 m_1 位数开始往低位计算的 m_2 位数内容传送至 **D** 的第 **n** 位数开始往低位计算的 m_2 位数中。
2. m_1=1~4，m_2=1~m_1，n=m_2~4。（4 个 bit 为一个单位）
3. SM605=OFF 时，为 BCD 模式，指令会自动将 S 的内容当成 BCD 的模式来执行。



若 S=K1234，D=K5678，则执行后 S=1234，D=5128

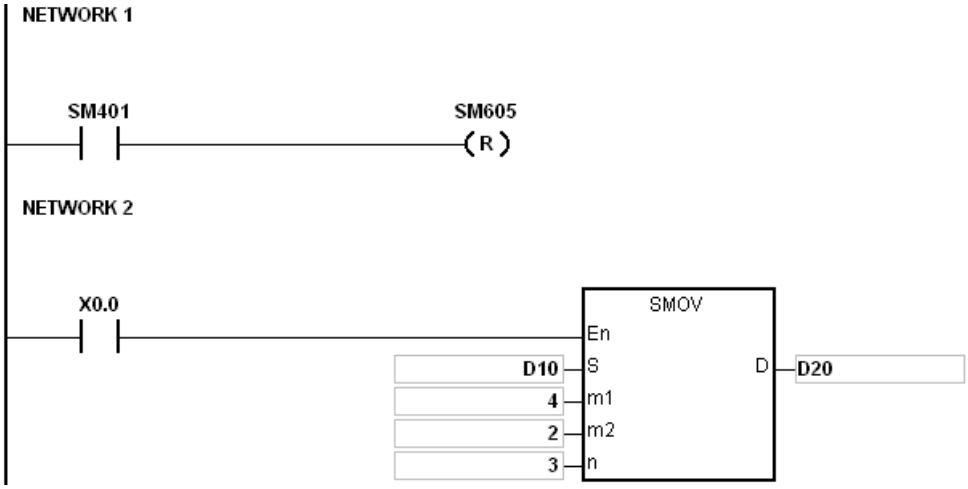
4. SM605=ON 时，为 BIN 模式。

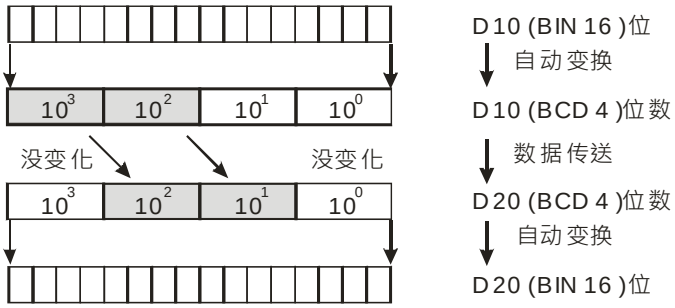


若 S=16#1234，D=16#5678，则执行后 S=16#1234，D=16#5128。

程序范例一：

1. 当 SM605=OFF 时 (BCD 模式)，X0.0=ON，指定 D10 的 10 进制数值的第 4 位数 (亦即千位数) 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(亦即百位数)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。
2. 当 BCD 值超过 0~9,999 的范围时，PLC 判定为运算错误，指令不执行，SM0=ON，错误码 SR0=16#200D。

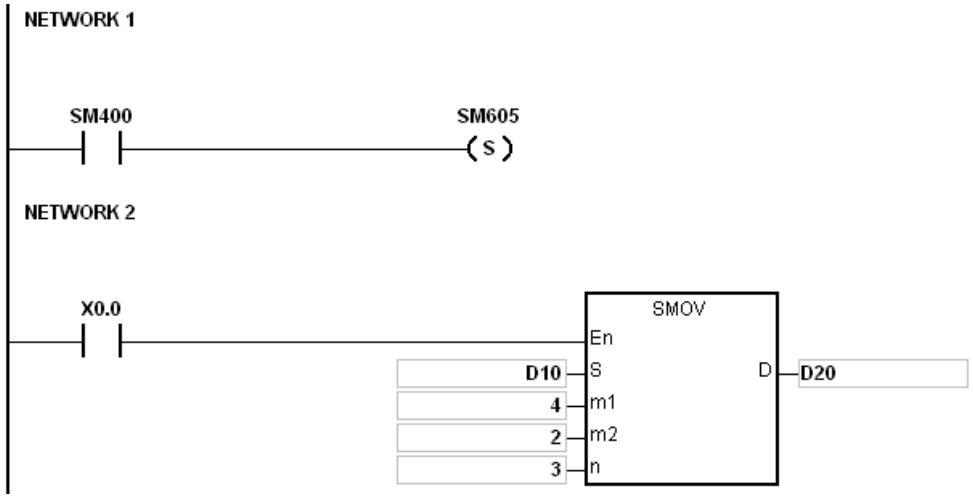


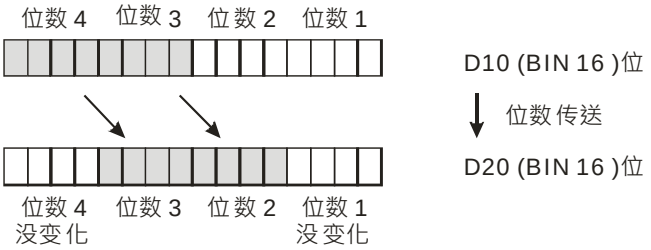


若执行前 D10=1234，D20=5678，执行完毕后，D10 不变，D20=5128。

程序范例二：

当 SM605=ON (BIN 模式) 时，使用 SMOV 指令的话，D10、D20 并不会作 BCD 变换，而是以 BIN 类型 4 个位为一个单位作传送。

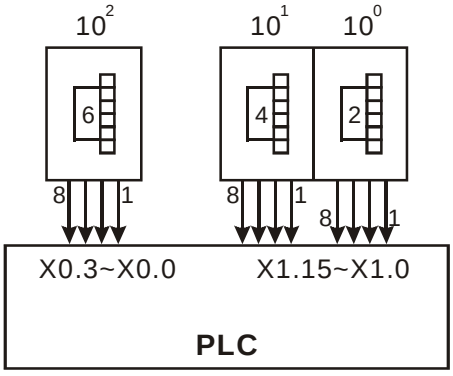


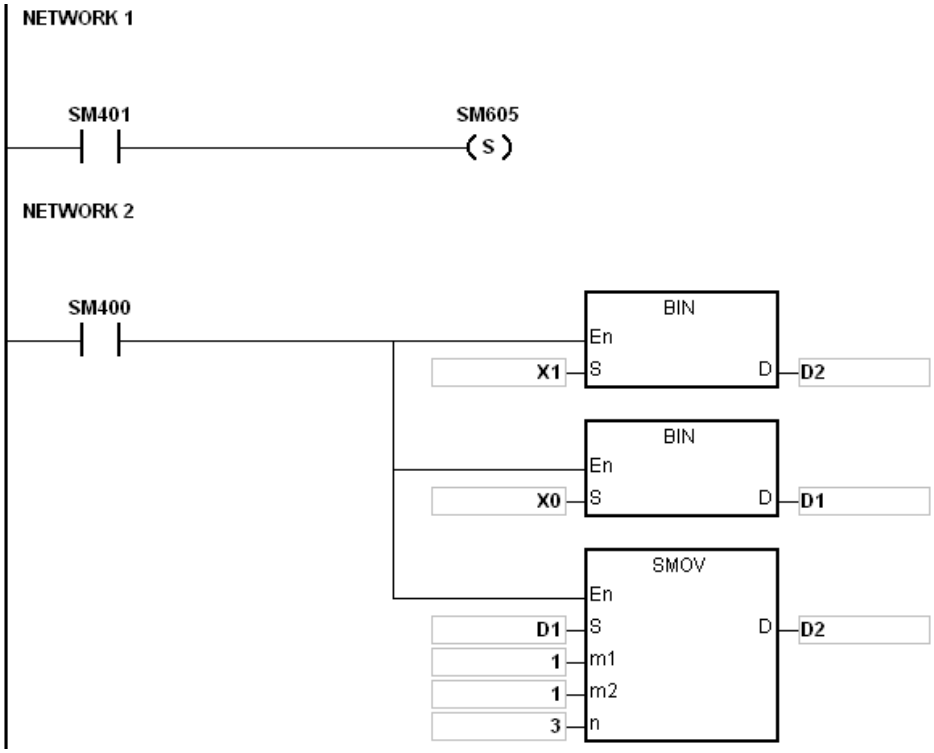


若执行前 D10=16#1234，D20=16#5678，执行完毕后，D10 不变，D20=16#5128。

程序范例三：

1. 连接于非连续编号输入端的指拨开关可使用本指令来合成。
2. 将右 2 位指拨开关传送至 D2 的右 2 位，左 1 位指拨开关传送至 D1 的右 1 位数当中。
3. 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。





补充说明：

1. 当指令为 BCD 模式时，如果 S<0 或 S>9999 或 D<0 或 D>9999，指令不执行，SM0=ON，错误码 SR0=16#200D。
2. m_1<1 或 m_1>4 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 当 m_2<1 或 m_2>m_1，指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 当 n<m_2 或 n>4 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

API	指令码			操作数	功能
0310		MOVB	P	**S**・**n**・**D**	多位移动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●	●	●	●	●	●	●		●						
n	●	●			●	●		●	●		○	○	○	○		
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
n		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 要移动的数据源

n ： 传送笔数

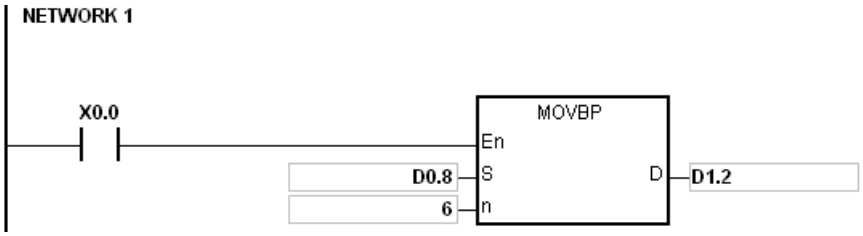
D ： 目的地装置要存放数据源的起始位数

指令说明：

1. 当该指令执行时，指定 **S** 起始位数内容传送至 **D** 开始的 **n** 个位数中。
2. 当 **S** 为 T、C、HC 装置时，只传送装置的状态线圈，而不会传送装置的当前值。
3. **n**=1~256，**n**<1 或 **n**>256 时指令不执行，SM0=ON，错误码 SR0=16#200B。

程序范例：

X0.0 为 ON 时，PLC RUN 后，D0.8~D0.13 之内容传送至 D1.2~D1.7。



补充说明：

1. 当 **D**+**n**-1 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 当 **S**+**n**-1 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

6.5 程序跳转指令

6.5.1 程序跳转指令一览表

API	指令码（位）		P 指令	功能
	16	32		
0400	CJ	–	✓	条件跳转
0401	JMP	–	–	无条件跳转
0402	GOEND	–	–	跳转到 END

6.5.2 程序跳转指令说明

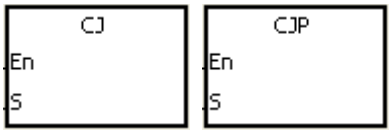
API	指令码		操作数	功能
0400	CJ	P	**S**	条件跳转

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S																

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



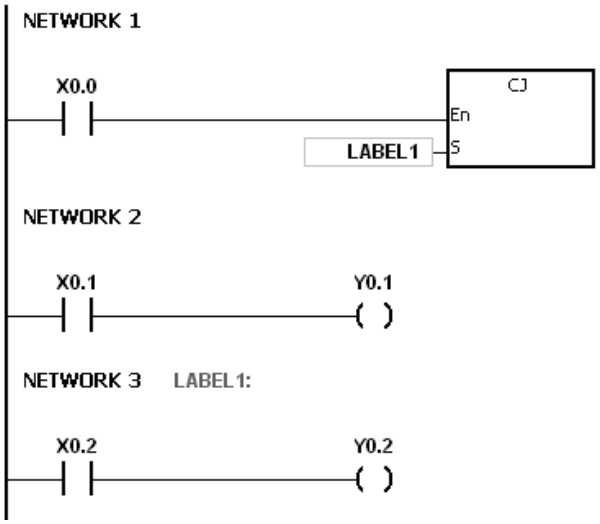
S ： 条件跳转之目的指定标签

指令说明：

1. 当用户希望 PLC 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。
2. 指定卷标 P(LABEL)所指之程序若在 CJ 指令之前，需注意可能发生 WDT 逾时之错误，以至于造成 PLC 停止运转，且错误灯直亮之状况，请注意使用。
3. CJ 指令可重复指定同一指定标签 P。
4. 跳转执行后，当中没有被扫描执行之各指令与装置动作情形说明如下：
   - Y、M、S 装置元件保持跳转发生前之状态。
   - 已启动之 T 定时器将持续计时，且计时到达设定时间将会驱动 T 线圈。
   - MC 与 MCR 指令请参考范例二说明。
   - 一般应用指令不会被执行。

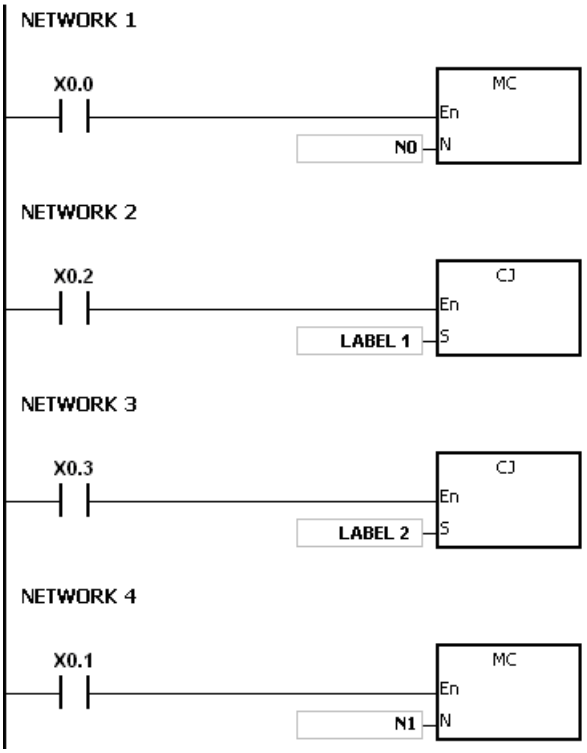
程序范例一：

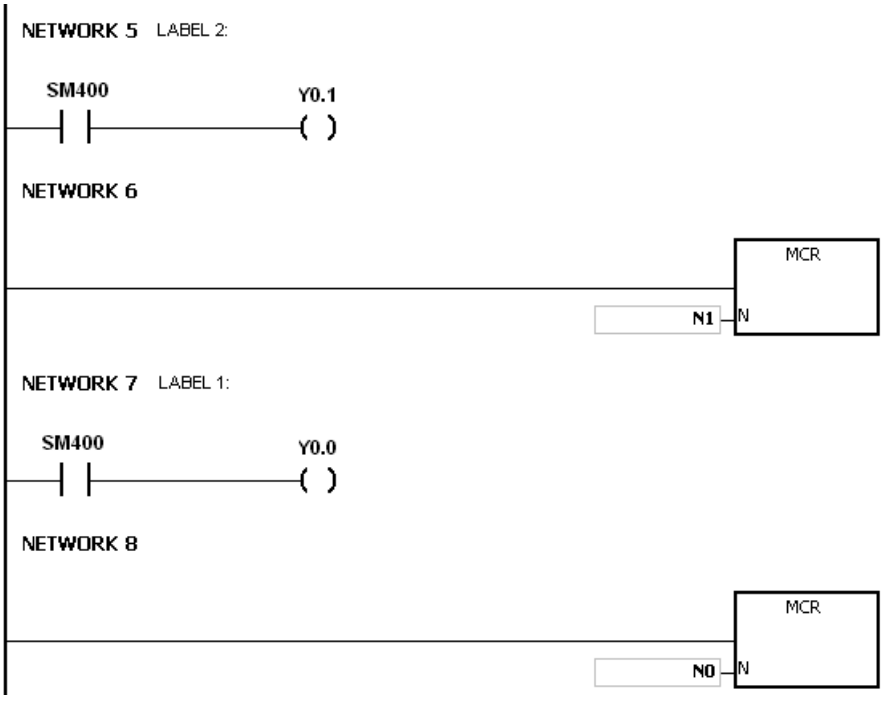
1. 当 X0.0=ON 时，程序自动从 NETWORK1 跳转至指定之标签 LABEL1（NETWORK3）继续执行，中间程序（NETWORK2）跳过不执行。
2. 当 X0.0=OFF 时，CJ 指令不被执行，程序如同一般执行顺序由 NETWORK1~3 继续往下执行。



程序范例二：

1. CJ 指令在 MC、MCR 指令间可使用在下列五种状况：
   (a) 在 MC~MCR 外。
   (b) 在 MC 外至 MC 内，如下图 LABEL1:以下回路有效。
   (c) 同一 N 层 MC 内至 MC 内。
   (d) 在 MC 内至 MCR 外。
   (e) 自 MC~MCR 内跳至另一 MC~MCR 内。
2. 执行 MC 指令时，PLC 会将之前开关接点的状态推入 PLC 内部自定义的堆栈中，而此堆栈由 PLC 自行控制，使用者无法改变；而后当执行到 MCR 指令时，会由堆栈的最上层取出之前的开关接点状态，当上面（b）、（d）、（e）的状况下时，则有可能会发生推入 PLC 内部堆栈与取出堆栈的次数不相同的情况，遇到这种状况时，堆栈最多能堆入 32 层，而另外取出堆栈的值最多取到堆栈为空时则不再取出，所以在搭配 CJ 等跳转指令时须注意堆栈的堆入和取出。


NETWORK 1
X0.0
MC
En
N0
N
NETWORK 2
X0.2
CJ
En
LABEL 1
S
NETWORK 3
X0.3
CJ
En
LABEL 2
S
NETWORK 4
X0.1
MC
En
N1
N

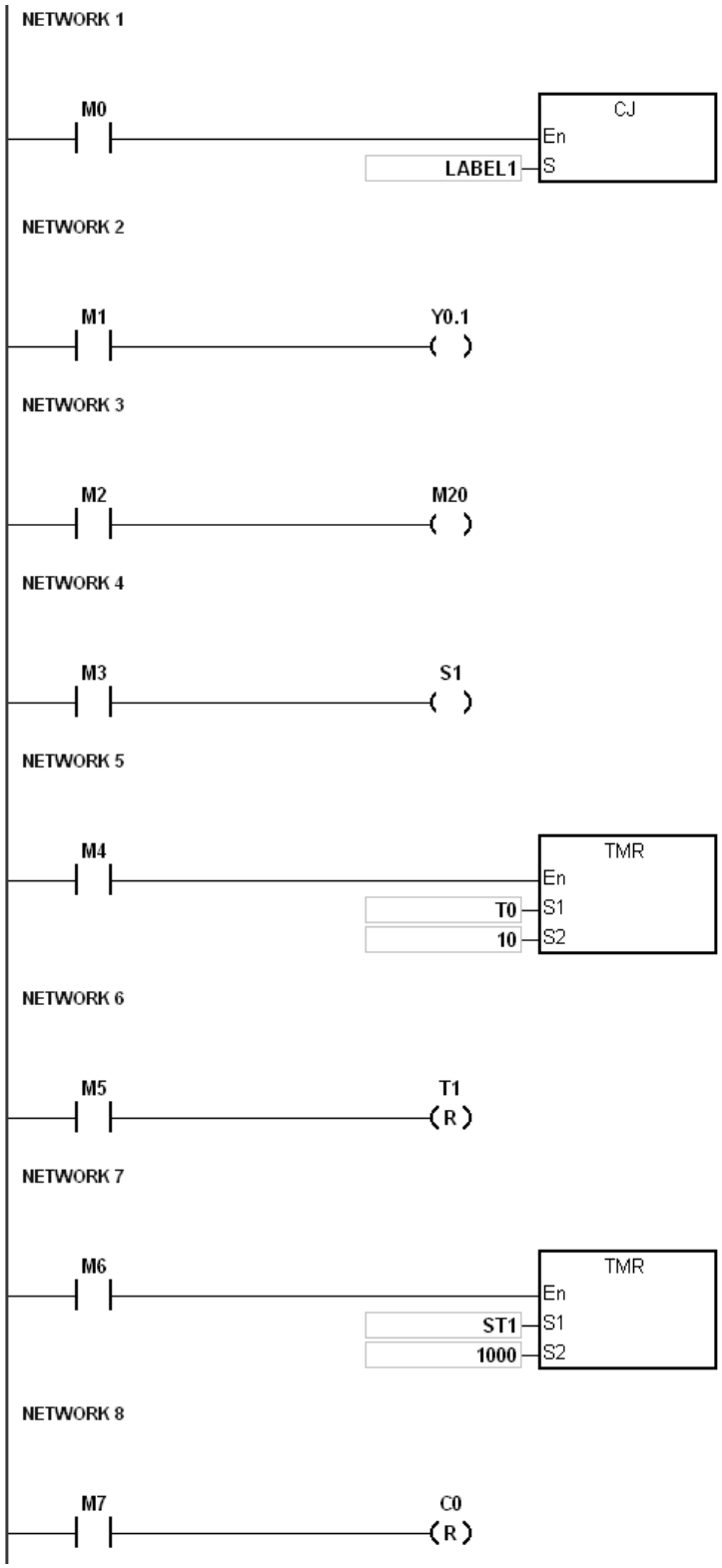

NETWORK 5 LABEL 2:
SM400
Y0.1
NETWORK 6
MCR
N1
N
NETWORK 7 LABEL 1:
SM400
Y0.0
NETWORK 8
MCR
N0
N

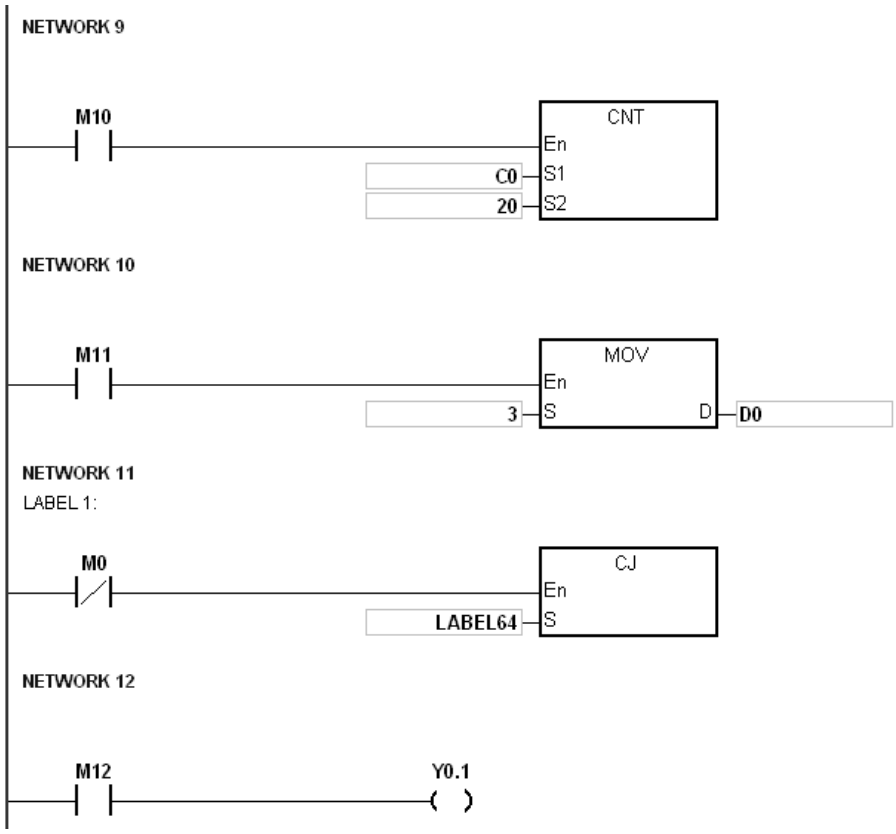
程序范例三：

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 M0=OFF	CJ 执行中接点状态 M0=ON	CJ 执行中输出线圈与指令状态 M0=ON
Y、M、S	M1、M2、M3 OFF	M1、M2、M3 由 OFF→ON	Y0.1 注 1、M20、S1 OFF
	M1、M2、M3 ON	M1、M2、M3 由 ON→OFF	Y0.1 注 1、M20、S1 ON
定时器	M4 OFF	M4 由 OFF→ON	定时器不作计时动作
	M4 ON	M4 由 ON→OFF	定时器保持计时并当计时到达设定值时驱动 T 线圈。
积分型定时器	M6 OFF	M6 OFF→ON	定时器 ST1 不作计时动作
	M6 ON	M6 ON→OFF	积分型定时器保持计时并当计时到达设定值时驱动 T 线圈。
计数器	M7、M10 OFF	M10 OFF→ON 触发	计数器不计数
	M7 OFF，M10 OFF→ON 触发	M10 OFF→ON 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 OFF→ON	应用指令不执行
	M11 ON	M11 ON→OFF	被跳过的应用指令不执行

注 1：Y0.1 为双重输出，M0 为 OFF 时，由 M1 控制，M0 为 ON 时，由 M12 控制。


NETWORK 1
M0
CJ
En
LABEL1
S
NETWORK 2
M1
Y0.1
NETWORK 3
M2
M20
NETWORK 4
M3
S1
NETWORK 5
M4
TMR
En
T0
S1
10
S2
NETWORK 6
M5
T1
R
NETWORK 7
M6
TMR
En
ST1
S1
1000
S2
NETWORK 8
M7
C0
R



补充说明：

指定标签 P（LABEL）在 ISPSoft 中的使用方式，请参考 ISPSoft 的使用手册说明。

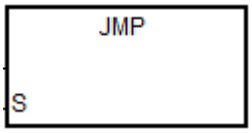
API	指令码		操作数	功能
0401	JMP		**S**	无条件跳转

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S																

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S ： 跳转之目的指定标签

指令说明：

1. 无条件跳转到程序中的某个指定标签 P（LABEL）。
2. 指定卷标 P 所指之程序若在 JMP 指令之前，需注意可能发生 WDT 逾时之错误，以至于造成 PLC 停止运转，且错误灯直亮之状况，请注意使用。
3. 跳转执行中各种装置动作情形说明，请参考 CJ 指令说明。
4. 指定标签 P（LABEL）在 ISPSoft 中的使用方式，请参考 ISPSoft 的使用手册说明。

API	指令码	操作数	功能
0402	GOEND	—	跳转到 END

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：

指令说明：

1. 条件成立时直接跳转到程序中的 END。
2. 功能块跟中断工作（TASK）与 FOR NEXT 中不支持此指令。
3. GOEND 执行时所略过的指令皆不动作，所有装置元件内容值与状态保持不变。

6.6 程序执行控制指令

6.6.1 程序执行控制指令一览表

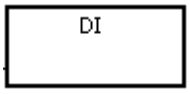
API	指令码（位）		P 指令	功能
	16	32		
0500	DI	–	–	全部中断插入关闭
0501	EI	–	–	全部中断插入启动
0503	EIX	–	–	指定中断启动
0504	DIX	–	–	指定中断关闭

6.6.2 程序执行控制指令说明

API	指令码	操作数	功能
0500	DI	-	中断插入禁止

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



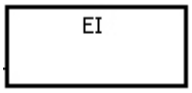
指令说明：

1. 请参考 API0501 EI 指令说明。

API	指令码		操作数	功能
0501	EI		-	中断插入致能

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



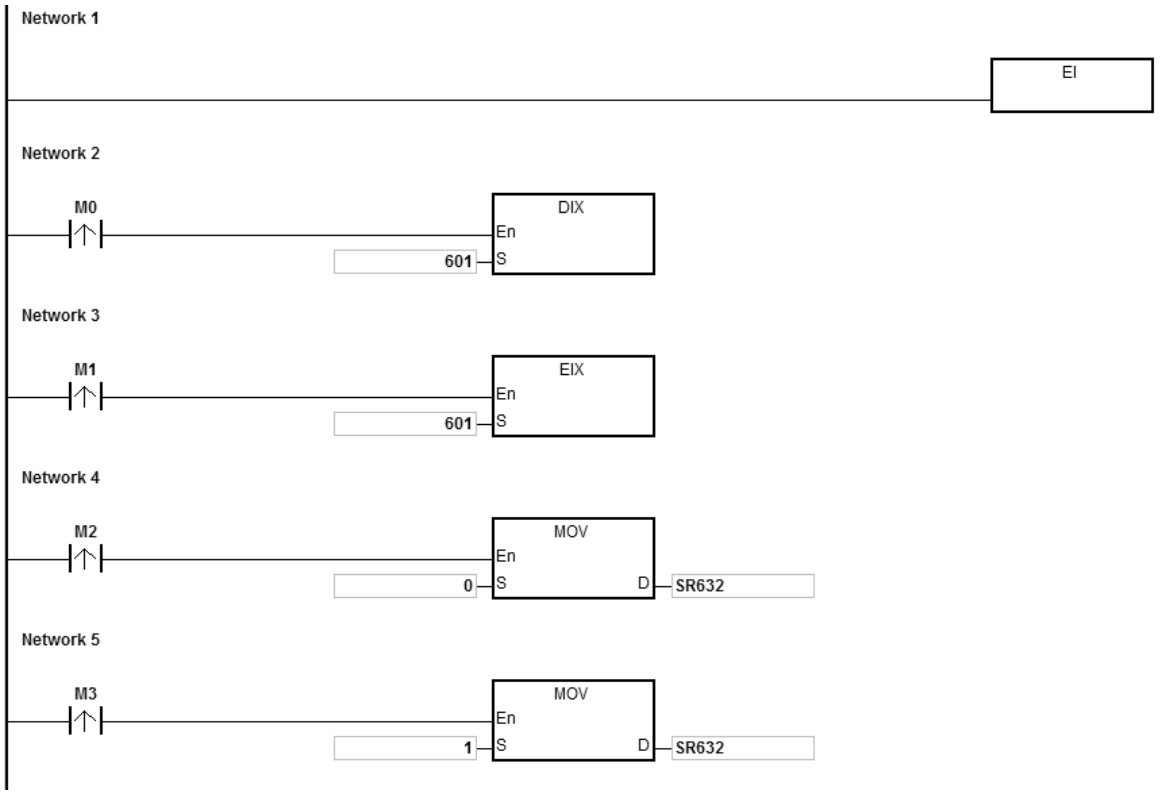
指令说明：

1. EI 表示程序中允许使用中断工作（TASK）(请参考下页补充说明)。
2. 程序中在 EI 指令到 DI 指令间允许使用中断工作（TASK），在程序中若无中断插入禁止之区间时，则可以不使用 DI 指令。
3. 中断工作执行中，不允许再执行其它产生的中断（但会记住有中断产生），会先执行完目前的中断工作，才去执行下一个中断工作。也即执行中断之阶层为一层架构，并且可再记录同一个中断编号一次。举例：当 I0 正在执行中（触发顺序编号 1），此时 I0 又再发生一次触发（触发顺序编号 2），则此次将先记录下来，若触发顺序编号 1 中断还未完成，可是又再触发一次（触发顺序编号 3），那么这第 3 次触发的中断将会被忽略。
4. 当多个中断同时发生时，以指标编号较小者优先执行。
5. 在 DI~EI 指令之间发生的中断要求无法执行，并且会被忽略掉。若是期望 PLC 进入系统内部运行时间时，则建议不要使用 DI 关闭中断。
6. 当中断处理中要实时 I/O 动作时，可在程序中写入 REF 指令或使用 DX/DY 组件立即更新 I/O 状态。
7. 每个中断编号皆配置有暂时遮蔽中断之功能，请参考下页补充说明内中断编号列表。

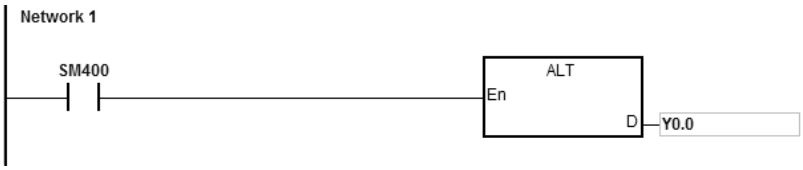
程序范例：

- 于 ISPSoft HWCONFIG 中设定系统参数 I601 时间中断为 500ms。
- PLC 执行时，当 Cyclic_0 程序扫描到 EI 指令之后，时间中断 I601 被启动，则会执行中断工作，而当中断工作（TASK）执行完毕后，则返回主程序并继续往下执行。
- 当 M0=ON 时，则指定 I601 时间中断关闭。
- 当 M1=ON 时，则指定 I601 时间中断启动。
- 当 M2=ON 时，SR632=0，则指定 I601 时间中断关闭。
- 当 M3=ON 时，SR632=1，则指定 I601 时间中断启动。

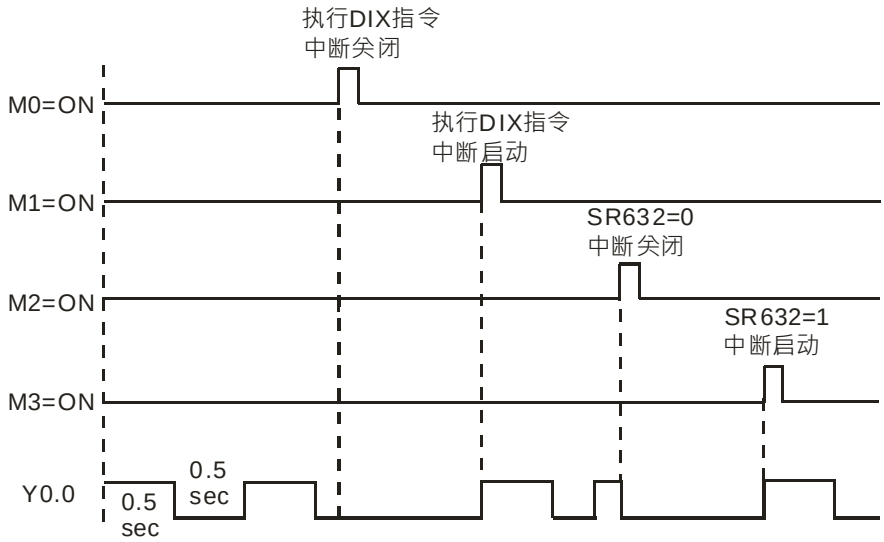
Cyclic_0 程序：



中断工作（TASK）:



时间中断动作时序图：



补充说明：

中断工作（TASK） 共分 7 大类，说明如下：

1. 外部输入中断（ I000~I115）

   I000 表示输入点 X0.0 下降沿触发中断，I100 表示输入点 X0.0 上升沿触发中断，I001 表示输入点 X0.1 下降沿触发中断，I101 表示输入点 X0.1 上升沿触发中断，以此累推。

2. 硬件高速比较中断（I200~I253）

   此比较中断又分为 6 组，每组分别对应一组硬件高速计数器（可参考 DCNT 指令说明）；每一组配置 4 个比较中断编号（请参考 DHSCS 指令说明），例如：第一组比较中断有 I200~I203 共 4 个，第二组为 I210~I213，以此累推。

3. 软件高速比较中断（I260~I267）

   此比较中断共有 8 个，所有软件高速计数器皆共享此中断。

4. 通讯中断

   使用 RS 指令可指定接收某字符时，触发此通讯接收中断请求。详细使用说明请参考 API1812 COM RS 指令。

   COM1：I300。 COM2：I302。Card 1：I304。Card 2：I306。

5. 扩展模块中断（I400~I431）

   每一台模块分别配置一个中断请求，当扩展模块有请求立即中断处理时，可藉由此中断设定之。

6. 高速输出中断（I500~I519）

   定位指令输出完成中断（I500~I505），搭配特殊标志 SM 指定启动，当脉冲输出完毕时，则发出中断请求。

   例如：DDRVI 指令第一轴输出完成时，需要产生中断 I500，则可以设定 SM471 为 ON，启动此功能。

   规画表输出完成中断（I510~I519），使用规划表搭配 TPO 指令，当脉冲输出完成时，则发出中断请求。

7. 定时中断（I601~I604）

   此定时中断之时间须由 HWCONFIG 设定。

   定时中断 I601~I603：默认值 10ms（单位时间为 1ms）（可设定范围 1~2000ms）。

   定时中断 I604：默认值 1ms（单位时间为 0.1ms）（可设定范围 0.1~200ms）。

全部中断编号、说明与遮蔽中断如下所列：

中断编号	中断名称	遮蔽中断 SR	Bit No.
I000	输入点 X0.0 外部中断下降沿触发	SR623	0
I001	输入点 X0.1 外部中断下降沿触发		1
I002	输入点 X0.2 外部中断下降沿触发		2
I003	输入点 X0.3 外部中断下降沿触发		3
I004	输入点 X0.4 外部中断下降沿触发		4
I005	输入点 X0.5 外部中断下降沿触发		5
I006	输入点 X0.6 外部中断下降沿触发		6
I007	输入点 X0.7 外部中断下降沿触发		7
I008	输入点 X0.8 外部中断下降沿触发		8
I009	输入点 X0.9 外部中断下降沿触发		9

中断编号	中断名称	遮蔽中断 SR	Bit No.
I010	输入点 X0.10 外部中断下降沿触发		10
I011	输入点 X0.11 外部中断下降沿触发		11
I012	输入点 X0.12 外部中断下降沿触发		12
I013	输入点 X0.13 外部中断下降沿触发		13
I014	输入点 X0.14 外部中断下降沿触发		14
I015	输入点 X0.15 外部中断下降沿触发		15
I100	输入点 X0.0 外部中断上升沿触发	SR624	0
I101	输入点 X0.1 外部中断上升沿触发		1
I102	输入点 X0.2 外部中断上升沿触发		2
I103	输入点 X0.3 外部中断上升沿触发		3
I104	输入点 X0.4 外部中断上升沿触发		4
I105	输入点 X0.5 外部中断上升沿触发		5
I106	输入点 X0.6 外部中断上升沿触发		6
I107	输入点 X0.7 外部中断上升沿触发		7
I108	输入点 X0.8 外部中断上升沿触发		8
I109	输入点 X0.9 外部中断上升沿触发		9
I110	输入点 X0.10 外部中断上升沿触发		10
I111	输入点 X0.11 外部中断上升沿触发		11
I112	输入点 X0.12 外部中断上升沿触发		12
I113	输入点 X0.13 外部中断上升沿触发		13
I114	输入点 X0.14 外部中断上升沿触发		14
I115	输入点 X0.15 外部中断上升沿触发		15
I200	硬件高速计数器 1 之高速比较中断 1	SR625	0
I201	硬件高速计数器 1 之高速比较中断 2		1
I202	硬件高速计数器 1 之高速比较中断 3		2
I203	硬件高速计数器 1 之高速比较中断 4		3
I210	硬件高速计数器 2 之高速比较中断 1		4
I211	硬件高速计数器 2 之高速比较中断 2		5
I212	硬件高速计数器 2 之高速比较中断 3		6
I213	硬件高速计数器 2 之高速比较中断 4		7
I220	硬件高速计数器 3 之高速比较中断 1		8
I221	硬件高速计数器 3 之高速比较中断 2		9
I222	硬件高速计数器 3 之高速比较中断 3		10
I223	硬件高速计数器 3 之高速比较中断 4		11
I230	硬件高速计数器 4 之高速比较中断 1		12
I231	硬件高速计数器 4 之高速比较中断 2		13

中断编号	中断名称	遮蔽中断 SR	Bit No.
I232	硬件高速计数器 4 之高速比较中断 3		14
I233	硬件高速计数器 4 之高速比较中断 4		15
I240	硬件高速计数器 5 之高速比较中断 1	SR626	0
I241	硬件高速计数器 5 之高速比较中断 2		1
I242	硬件高速计数器 5 之高速比较中断 3		2
I243	硬件高速计数器 5 之高速比较中断 4		3
I250	硬件高速计数器 6 之高速比较中断 1		4
I251	硬件高速计数器 6 之高速比较中断 2		5
I252	硬件高速计数器 6 之高速比较中断 3		6
I253	硬件高速计数器 6 之高速比较中断 4		7
I260	软件高速计数器之高速比较中断 1	SR627	0
I261	软件高速计数器之高速比较中断 2		1
I262	软件高速计数器之高速比较中断 3		2
I263	软件高速计数器之高速比较中断 4		3
I264	软件高速计数器之高速比较中断 5		4
I265	软件高速计数器之高速比较中断 6		5
I266	软件高速计数器之高速比较中断 7		6
I267	软件高速计数器之高速比较中断 8		7
I300	COM1 指定接收字符之通讯中断	SR628	0
I301	保留		1
I302	COM2 指定接收字符之通讯中断		2
I303	保留		3
I304	Card1 指定接收字符之通讯中断		4
I305	保留		5
I306	Card2 指定接收字符之通讯中断		6
I307	保留		7
I500	定位指令轴 1 输出完成中断	SR629	0
I501	定位指令轴 2 输出完成中断		1
I502	定位指令轴 3 输出完成中断		2
I503	定位指令轴 4 输出完成中断		3
I504	定位指令轴 5 输出完成中断		4
I505	定位指令轴 6 输出完成中断		5
I510	定位规划表输出完成中断 1	SR630	0
I511	定位规划表输出完成中断 2		1
I512	定位规划表输出完成中断 3		2
I513	定位规划表输出完成中断 4		3

中断编号	中断名称	遮蔽中断 SR	Bit No.
I514	定位规划表输出完成中断 5		4
I515	定位规划表输出完成中断 6		5
I516	定位规划表输出完成中断 7		6
I517	定位规划表输出完成中断 8		7
I518	定位规划表输出完成中断 9		8
I519	定位规划表输出完成中断 10		9
I601	定时中断 1（单位时间 1ms）	SR632	0
I602	定时中断 1（单位时间 1ms）		1
I603	定时中断 1（单位时间 1ms）		2
I604	定时中断 1（单位时间 0.1ms）		3

注：当多个中断同时发生时，将从中断编号最小的先行执行；但若是已经执行中断工作中，别的中断才发生，则须视 PLC 已经执行编号循环决定。举例：正在执行 I400 中断，但此时有 I500 与 I300 同时发生，则 I400 执行完毕之后，将执行 I500 中断，之后才轮到 I300 中断。

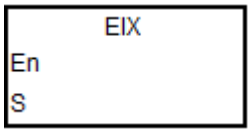
API	指令码			操作数	功能
0503		EIX		**S**	指定中断启动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S													○			

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S ： 数据源

指令说明：

1. **S** 的只可输出 10 进制数值，并且此数值要为 I 中断编号之数值，若输入非中断表内所列之中断编号数值，则指令无效且不会有错误警告发生。举例：期望开启 I500 中断，则要输入 EIX 500。
2. AS 系列主机之中断工作皆默认为开启状态，若程序执行中有使用 DIX 指令关闭中断时，才需使用此 EIX 指令进行重新开启之动作。
3. 执行此指令时系统将自动设定 SR623~SR634 内所对应之中断位状态。
4. 如果没有执行此指令，则中断开启或关闭的控制位会以 SR623~SR634 的内容来决定。
5. 范例请参考 API0501 EI 指令说明。

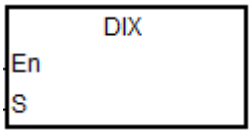
API	指令码	操作数	功能
0504	DIX	**S**	指定中断关闭

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S													○			

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S ： 数据源

指令说明：

1. **S** 的只可输出 10 进制数值，并且此数值要为 I 中断编号之数值，若输入非中断表内所列之中断编号数值，则指令无效且不会有错误警告发生。举例：期望关闭 I500 中断，则要输入 DIX 500。
2. AS 系列主机之中断工作皆默认为开启状态，若程序执行中有需要暂时关闭中断时，则可使用 DIX 指令进行关闭之动作。
3. 执行此指令时系统将自动设定 SR623~SR634 内所对应之中断位状态。
4. 如果没有执行此指令，则中断开启或关闭的控制位会以 SR623~SR634 的内容来决定。
5. 范例请参考 API0501 EI 指令说明。

6.7 I/O 更新指令

6.7.1 I/O 更新指令一览表

API	指令码（位）		P 指令	功能
	16	32		
0600	REF	–	✓	I/O 更新处理
0601	–	DHSRF	✓	高速比较值立即刷新
0602	REFF	–	✓	更新输入点滤波时间

6.7.2 I/O 更新指令说明

API	指令码		操作数	功能
0600	REF	P	**D**，**n**	I/O 更新处理

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
D	○	○														
n								●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号:

D ： 欲 I/O 更新处理之起始装置

n ： I/O 更新处理之数目

指令说明：

1. PLC 一般的 I/O 更新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之前，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 指令之后，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 I/O 数据，则可利用本指令，或者使用 DX/DY 装置元件立即输入/输出。
2. 此 **n** 操作数其数值须为 8 的倍数，不足 8 数值以 8 执行，例如：数值为 20，则将以 24 点执行立即刷新动作。**n** 数值范围为 8, 16, 24, …, 至最大 256。
3. 韧体版本为 V1.04.00(含)以上时，则提供 **D** 指定为正在高速输出点的编号，并且 **n** 指定为 1 时，即表示立即刷新高速输出对应的 SR 当前位置值。若 **n** 指定为 0 时，即表示立即暂停该高速输出、设定暂停输出标志，以及刷新 SR 当前位置值。

   举例：当 X0.0 外部输入中断程序，启动 REF　Y0.0　0 时，则表示 X0.0 发生外部中断时，必须立即暂停 Y0.0 高速输出、设定暂停中标志 SM，并更新当前输出位置 SR。注意：若有搭配设定自动复位标志，则 PLC 只会清除自动复位标志，而不会设定暂停中标志。

5. **n** 及 **D** 参数说明

n 输入数值	**D** 指定装置	REF 指令执行动作说明
n=8 的倍数	Y0.0 或 X0.0	立即更新 I/O 数据，请参考范例一及二说明
n=1#1	高速输出点	立即刷新脉冲位置，请参考范例三说明
n=0#1	高速输出点 无搭配复位标志	立即暂停该高速输出点、设定暂停输出标志，以及刷新当前位置值，请参考范例三说明
n=0#1	高速输出点 有搭配复位标志	清除复位标志，刷新当前位置值 请参考范例三说明
n= -1#2	任意 X 输入点	刷新 DS301 TxPDO 映像区数据（Master <= Slave）
n= -1#2	任意 Y 输出点	刷新 DS301 RxPDO 映像区数据（Master => Slave）
n= -2#3	任意 X 输入点	响应已收到 MODBUS TCP 的通讯命令（PLC 为 Server）

注#1：需搭配韧体版本为 V1.04.00 版以上。

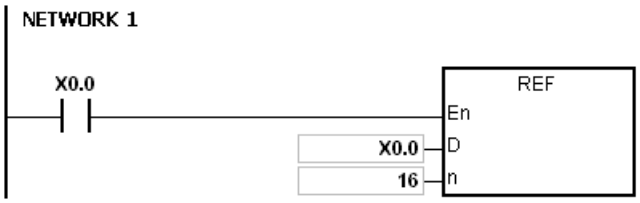
注#2：主站 Master V1.06.40 版以上支持；从站 Slave V1.12 版以上支持。此功能不会加快 PDO 数据交换时间，建议使用在 PLC 扫描时间大于 DS301 的 PDO 更新时间。

注#3：需搭配韧体版本为 V1.10.30 版以上。当 PLC 程序扫描时间过长时，可藉由此功能自行搭配时间中断程序，定时响应上位机透过 MODBUS TCP 下达的通讯命令，藉此加快上位机与 PLC 的通讯速度。建议定时的最快时间为 5ms 即可，若定时中断太快，反而可能拖慢 PLC 扫描程序的速度。

注意：此响应通讯功能会实时处理所有数据，等同于 PLC 程序扫描一个周期中，读取或写入数据会被插断。

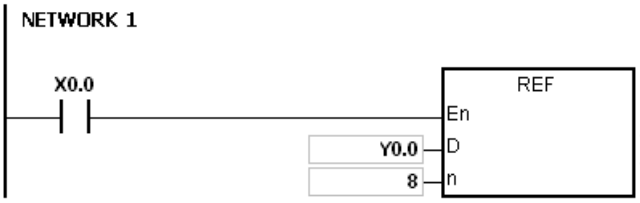
程序范例一：

当 X0.0=ON 时，PLC 会立即读取 X0.0~X0.15 之输入点状态，输入信号立即更新。



程序范例二：

当 X0.0=ON 时，Y0.0~Y0.7 之 8 点输出信号实时被送至输出端，输出信号立即更新，不必到 END 指令才输出。



补充说明：

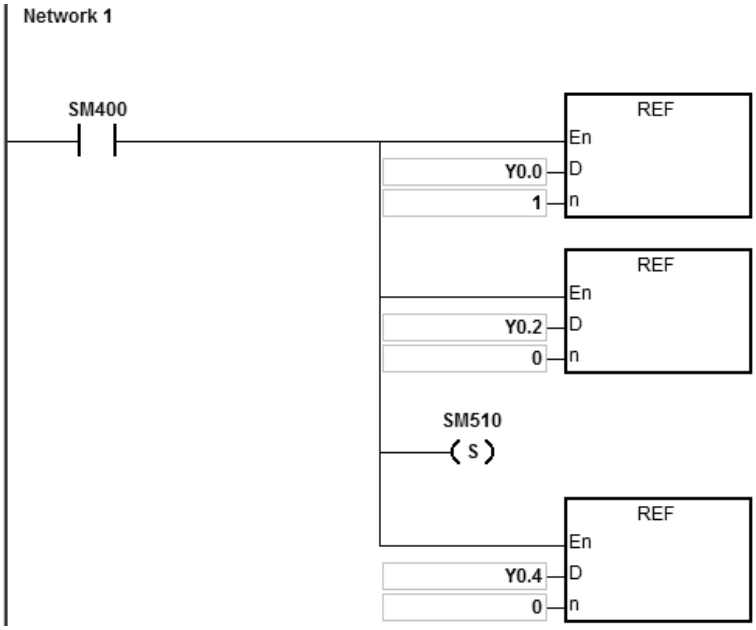
1. **D**+**n**-1 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

程序范例三：

1. 当 X0.0 外部中断发生时，启动 REF 功能

n 输入数值	**D** 指定装置	REF 指令执行动作说明
n=1	高速输出点 Y0.0	立即刷新 Y0.0 的脉冲位置(SR460)
n=0	高速输出点 Y0.2 无搭配复位标志	1. 立即暂停该高速输出 2. 设定暂停输出标志(SET SM483) 3. 刷新 SR480 当前位置值
n=0	高速输出点 Y0.4 且有搭配复位标志 SM510	1. 清除复位标志(RST SM510) 2. 刷新 SR500 当前位置值

外部中断程序 X0.0



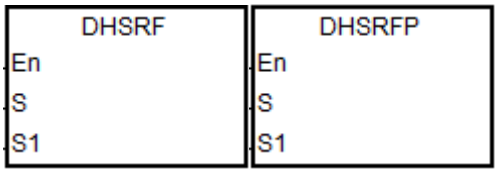
API	指令码			操作数	功能
0601	D	HSRF	P	**S**，**S_1**	高速比较值立即刷新

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S							○									
S_1								○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S												●	
S_1			●				●						

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号:



S ： 欲更新之高速计数器

S_1 ： I/O 更新处理之数目

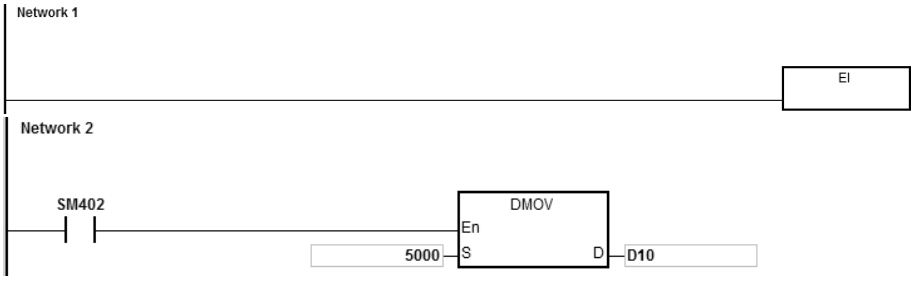
指令说明：

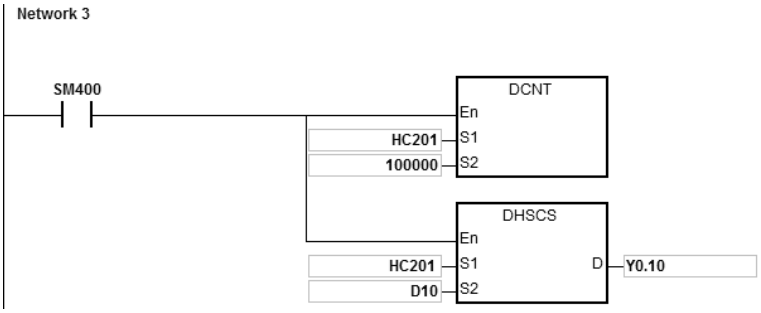
1. PLC 的高速比较器更新比较值的时机点为 DHSCS 或 DHSCR 指令被程序扫描到的时候，因此若是扫描时间太长，或者输入信号速度很快时，则可能会发生来不及比较到达的情况。此时用户可利用本指令，将比较数值立即刷新至 PLC 硬件比较器内，以利比较动作能实时被执行。
2. **S** 操作数为指定刷新的高速计数器编号，**S_1** 为须与 DHSCS 或 DHSCR 指令所指定的相同组件或变量名称，不可采用立即值方式输入。若是 **S_1** 没有与已经启动高速比较指令的操作数相同，则此指令的执行将无效。
3. 此指令需韧体版本为 V1.04.00 版(含)以上才有支持。

程序范例一：

1. PLC RUN 时，DHSCS 的比较值为 5000。
2. 当 X0.0 外部中断发生时，立即将 DHSCS 的比较值设定为 8000。

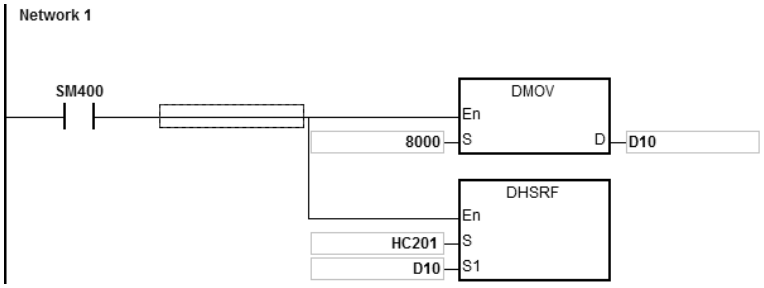
主程序：





外部中断程序 X0.0：

需先将变更的比较值设定至相同变量(如范例：D10)，然后执行 DHSRF 指令进行变更



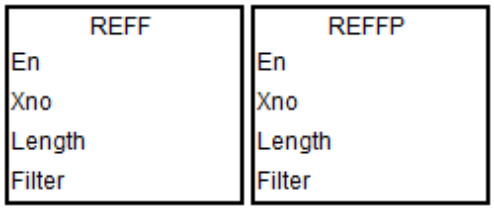
API	指令码			操作数	功能
0602		REFF	P	**Xno**，**Length**，**Filter**	更新输入点滤波时间

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Xno	○															
Length					●	●		●	●				○	○		
Filter					●	●		●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Xno	●												
Length		●				●							
Filter		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号:



Xno ： 更新 X 输入点之起始装置

Length ： 更新 X 输入点之长度

Filter ： 更新滤波时间（单位：微秒）

指令说明：

1. 此指令需搭配主机韧体版本 V1.06.00 版，以及 ISPSoft V3.06 版以上。
2. **Xno** 为起始修改 X 点的输入编号，**Length** 为修改 X 点的长度；若起始 X 点为主机内建的输入点，则假设 **Xno** 设 X0.3，**Length** 设 3，当指令设定之后，其 X0.3~X0.5 的输入滤波时间都将被更新。若起始 X 点为扩展模块的输入点，则假设 **Xno** 设 X1.3，**Length** 设 3（遇到扩展模块输入点时，PLC 自动以 8 为单位），当指令设定之后，其 X1.0~X1.7 这 8 个输入点的输入滤波时间都将被一起更新。
3. **Xno** 与 **Length** 组合后，必须在 PLC 整个系统的输入点范围内，若超出将不会被执行，也不会显示错误信息。
4. 当针对 PLC 主机内建的输入点变更滤波时间，此 **Filter** 为指定修改的滤波时间，单位是微秒（us）。当针对扩展模块的输入点变更滤波时间，此 **Filter** 为指定修改的滤波时间，单位是毫秒（ms）。由于此指令仅供变更当下滤波时间，因此不会连动修改 HWCONFIG 所设定之滤波时间。当设定时间超出 HWCONFIG 显示的范围时，PLC 将自动以最大或最小值设定，并且不会显示任何错误。
5. 主机与扩展模块的滤波时间在 PLC 整个系统上电时，都是依照 HWCONFIG 所设定的数值去执行。
6. 此指令一般建议用脉冲执行型指令 REFFP，若采用 A（或 B）接点方式设定，则 PLC 会自动只执行第一次指令被启动时所设定之滤波时间。

6.8 便利指令

6.8.1 便利指令一览表

API	指令码（位）		P 指令	功能
	16	32		
0700	ALT	–	✓	ON/OFF 交替
0701	TTMR	–	–	教导式定时器
0702	STMR	–	–	特殊定时器
0703	RAMP	DRAMP	–	周期型的倾斜信号
0704	MTR	–	–	矩阵输入
0705	ABSD	DABSD	–	绝对方式凸轮控制
0706	INCD	–	–	相对方式凸轮控制
0708	–	DPIDE	–	PID 运算
0709	XCMP	–	–	多任务位输入比较设定
0710	YOUT	–	–	多任务位比较与输出
0711	–	DSUNRS	✓	日出与日落时间

6.8.2 便利指令说明

API	指令码			操作数	功能
0700		ALT	P	**D**	ON/OFF 交替

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D	●	●	●	●				●		○						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

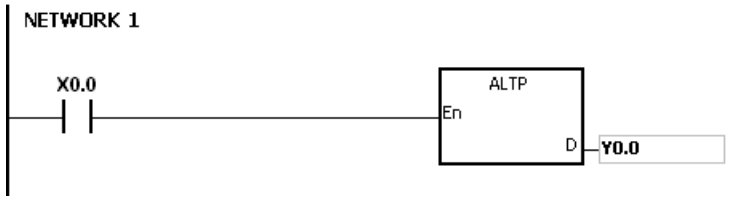
D ：目的地装置

指令说明：

1. ALT 指令执行时，**D** ON/OFF 交换。
2. 本指令一般都是使用脉冲执行型指令（ALTP）。

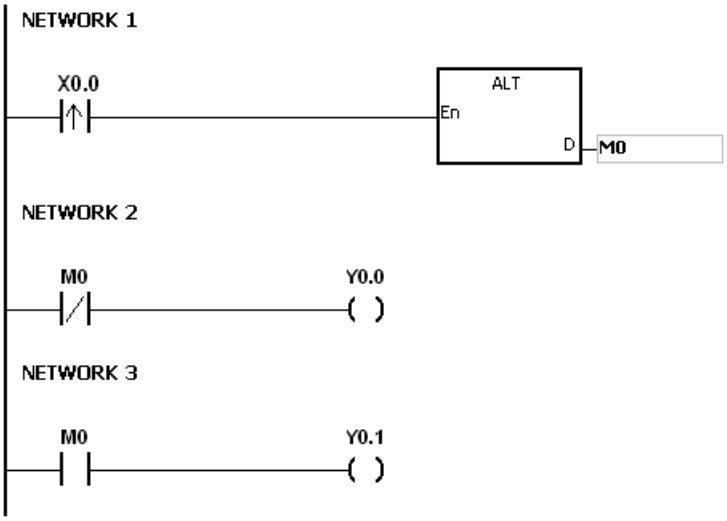
程序范例一：

当第一次 X0.0 从 OFF→ON 时，Y0.0=ON。第二次 X0.0 从 OFF→ON 时，Y0.0=OFF。



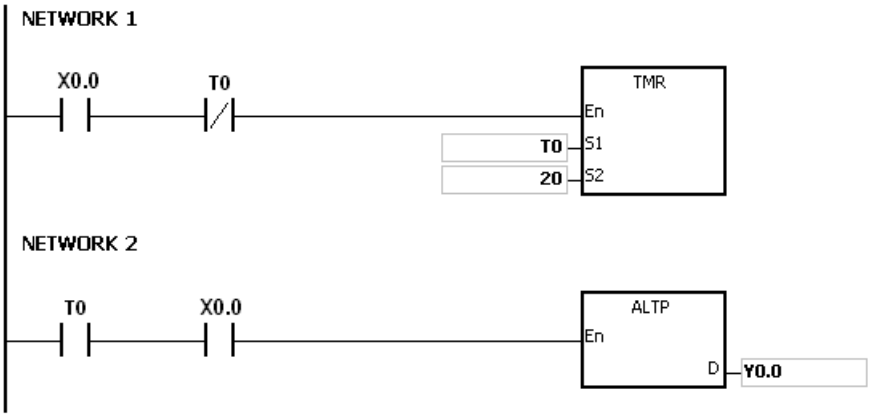
程序范例二：

使用单一开关控制启动与停止。一开始时，M0=OFF 故 Y0.0=ON、Y0.1=OFF，当 X0.0 作第一次 ON/OFF 时，M0=ON 故 Y0.1=ON、Y0.0=OFF，第二次 ON/OFF 时，M0=OFF 故 Y0.0=ON 而 Y0.1=OFF。



程序范例三：

产生闪烁的动作。当 X0.0=ON 时，T0 每隔 2 秒产生一个脉冲，Y0.0 输出会依 T0 脉冲产生 ON/OFF 交替。



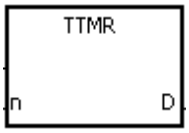
API	指令码	操作数	功能
0701	TTMR	**D**．**n**	教导式定时器

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●						●			○					
n	●	●						●	●		○		○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



D ：储存按钮开关 ON 时间的装置编号

n ：倍数设定

指令说明：

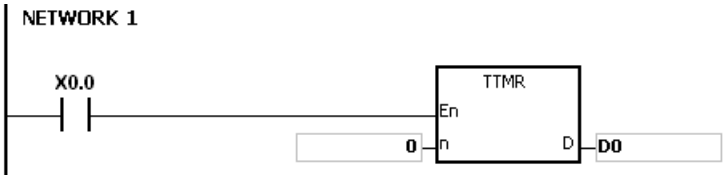
1. 将开关 ON 时间以秒为单位乘以 **n** 倍数后存放于 **D** 内，**D+1** 为系统内部使用，指令执行时不可更改，否则会造成时间计算错误。
2. **D** 的内容值在条件接点为 ON 的时候会先清为 0。
3. 倍数设定：**n**=0 时，**D** 以秒为单位，**n**=1 时，**D** 乘以 10 倍以 100ms 为单位，n=2 时，D 乘以 100 倍以 10ms 为单位。

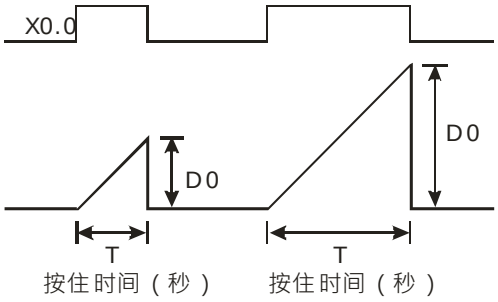
n	D
K0（单位：s）	1×T
K1（单位：100 ms）	10×T
K2（单位：10 ms）	100×T

4. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。
5. **n** 操作数的范围为 0~2。

程序范例一：

1. 按钮开关 X0.0 被按住时间(X0.0 的 ON 时间)，由 n 来指定该时间的倍数，并将位数时间存入于 D0 当中。由此，可使用按钮开关来调整定时器的设定值。
2. 当 X0.0 变成 OFF 时，D0 内容没有变化。





补充说明：

1. **D**+1 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. n<0 或 n>2 时，指令不执行，SM0=ON，SR0 记录错误码=16#200B。
3. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。

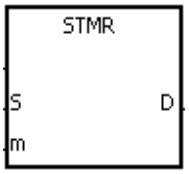
API	指令码	操作数	功能
0702	STMR	**S**，**m**，**D**	特殊定时器

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S					○											
m	●	●						●	●		○		○	○		
D		●	●	●				●		○						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S											●		
m		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S ： 定时器编号（T0~T511）

m ： 定时器设定值

D ： 输出装置的起始编号

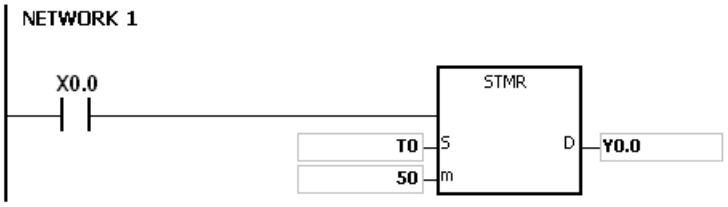
指令说明：

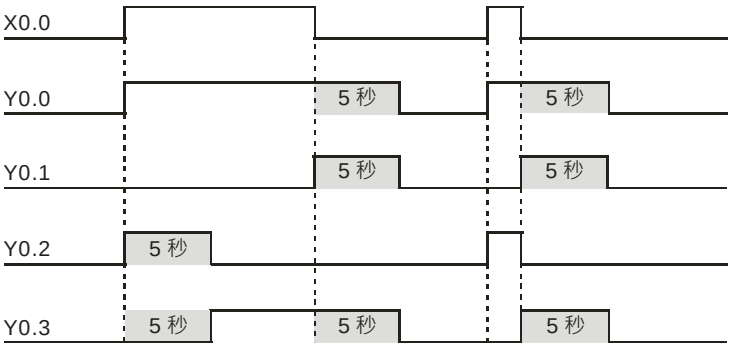
1. STMR 指令，用来产生 OFF 延迟，一次触发及闪烁回路的专用指令。
2. STMR 指令所指定的 T 定时器以 100ms 为单位。
3. STMR 指令所指定的定时器号码不可重复使用。
4. **D** 连续占用 4 个 bit。
5. 执行此指令前，请先 RST D~D+3 的状态。
6. 当条件接点未启动时，如果装置内容为以下两种，则 **D**、**D**+1、**D**+3 的状态会 ON m 秒之后再 OFF，除了以下两种情形之外，**D**~**D**+3 的状态会保持 OFF。
   - T 定时器内容值如果小于或等于 m，且 **D** 的状态为 ON，**D**+1 的状态为 OFF。
   - T 定时器内容值小于 **m**，且 **D**、**D**+1、**D**+3 的状态为 ON，**D**+2 的状态为 OFF。
7. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。
8. **m** 操作数的范围为 1~32767

程序范例：

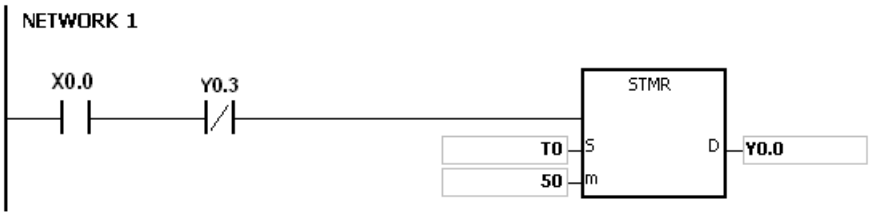
1. 当 X0.0=ON 时，STMR 指令指定定时器 T0，T0 的设定值为 5 秒。
2. Y0.0 为 OFF 延迟接点：当 X0.0 由 OFF 变 ON 时，Y0.0=ON，到 X0.0 由 ON 变 OFF 时，延迟 5 秒后 Y0.0=OFF。
3. Y0.1 于 X0.0 由 ON 变 OFF 时，作一次 5 秒 Y0.1=ON 输出。
4. Y0.2 于 X0.0 由 OFF 变 ON 时，作一次 5 秒 Y0.2=ON 输出。

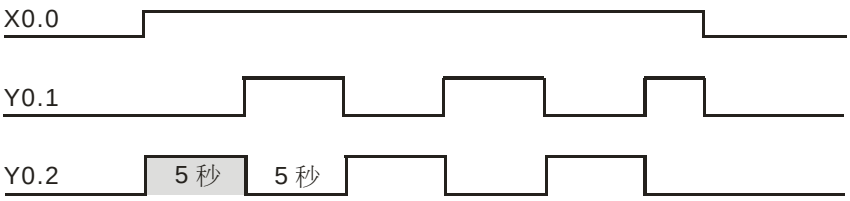
5. Y0.3 于 X0.0 由 OFF 变 ON 时，延迟 5 秒后 Y3=ON，到 X0.0 由 ON 变 OFF 时，延迟 5 秒后 Y0.3=OFF。





6. 在条件接点 X0.0 后面加一个 Y0.3 的 b 接点，则 Y0.1、Y0.2 可作闪烁回路输出。当 X0.0 变成 OFF 时，Y0.0、Y0.1 及 Y0.3 变成 OFF，T0 的内容被复位为 0。





补充说明：

1. **D+3** 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **m**<=0，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of BOOL。

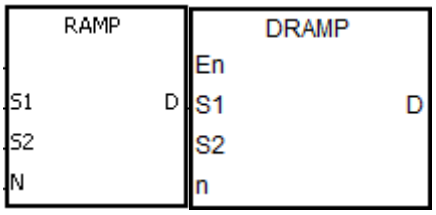
API	指令码			操作数	功能
0703	D	RAMP		**S_1 · S_2 · D · n**	周期型之倾斜信号

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●					●	●	●		○	○	○	○		
S_2	●	●					●	●	●		○	○	○	○		
D		●					●	●								
n	●	●					●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



S_1 ： 倾斜信号的起点设定

S_2 ： 倾斜信号的终点设定

D ： 倾斜信号的经过时间值

n ： 扫描次数

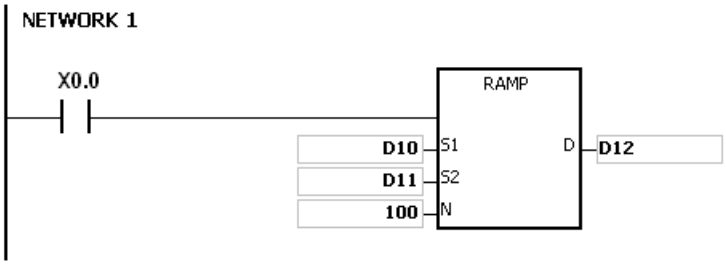
指令说明：

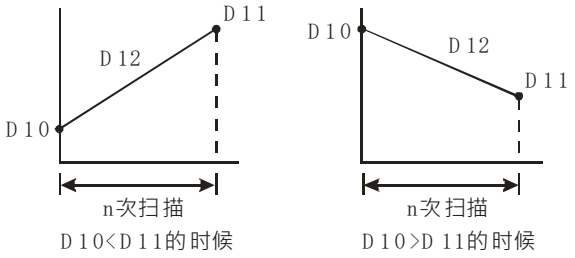
1. 本指令为一具有斜率型输出之指令，其斜率为线性型，并且与 PLC 扫描执行程序有绝对的关系，因此使用本指令时，建议将扫描时间加以固定，或者将此指令写至时间中断程序里。
2. 预先将倾斜信号的起点设定值写入 **S_1** 及倾斜信号之终点设定值写入 **S_2** 内，当条件接点=ON 时， **D** 从 **S_1** 所设定的值往 **S_2** 增加或减少，其扫描次数过程数值存放于 **D**+1 中，当 **D** = **S_2** 或 **D**+1=**n** 已到达扫描次数时，SM687 完成标志=ON。
3. 当指令之条件接点由 ON 变 OFF 时，其 **D** 跟 **D**+1 的内容值将清除为 0，以及 SM687 完成标志=OFF。
4. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。
5. 若需要固定扫描时间，请参考 ISPSoft 手册中的硬件组态工具 说明。
6. **n** 操作数的范围为 1~32767。设定超出此范围值时，指令不执行。
7. 32 位指令才可使用 HC 装置，但不可使用 E 装置。
8. 搭配 SM686 标志，可控制 **D** 内容值的清除动作，详细清除时机请参考范例说明。

程序范例：

本指令若是与模拟信号输出搭配使用时，可执行缓冲启动/停止的动作。

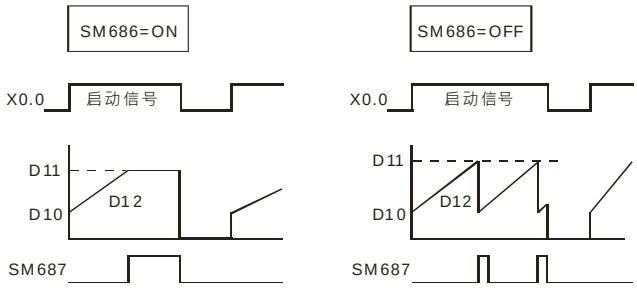
1. 启动信号 X0.0 从 ON 变 OFF 时，指令停止执行，且 D12 将被清除为 0，D13 数值不变，SM687 清除为 OFF；当 X0.0 再度 OFF 变为 ON 时，D12 的内容被设为 D10 的内容值，D13 被清除为 0，且开始重新计算。
2. 当 SM686=OFF 时，则 D12 到达 D11 设定值时，将设定 SM687=ON 一个扫描周期，下次扫描到指令时，D12 的内容会自动变回 D10 的设定值，然后 D13 的内容再次被清除为 0。





扫描次数存放D13当中

3. 当 SM686=ON 时，则 D12 的内容到达 D11 的设定值时，D12 的内容不会被清除为 0，并且 SM687 将被设为 ON，直到条件接点被关闭时，才会自动清除 D12 为 0 与 SM687=OFF。下图为 SM686 标志 ON/OFF 的执行动作比较示意图。



补充说明：

1. **D**+1 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n<=0**，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 16 位指令的 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。
4. 32 位指令的 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of DWORD/DINT。

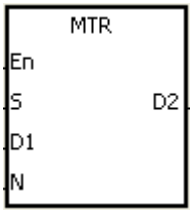
API	指令码		操作数	功能
0704	MTR		**S**，**D_1**，**D_2**，**n**	矩阵输入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	○															
D_1		○														
D_2		○	○	○				○								
n	●	●						●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
D_1	●												
D_2	●												
n		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S ： 矩阵扫描输入起始装置

D_1 ： 矩阵扫描输出起始装置

D_2 ： 矩阵扫描所对应起始装置

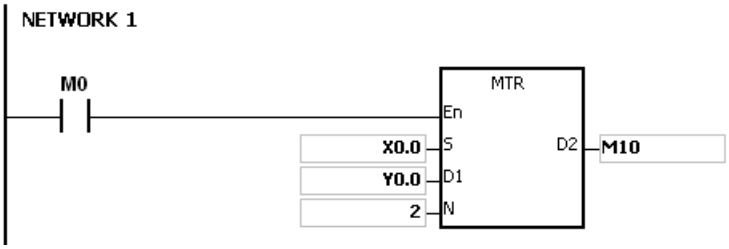
n ： 矩阵扫描的列数

指令说明：

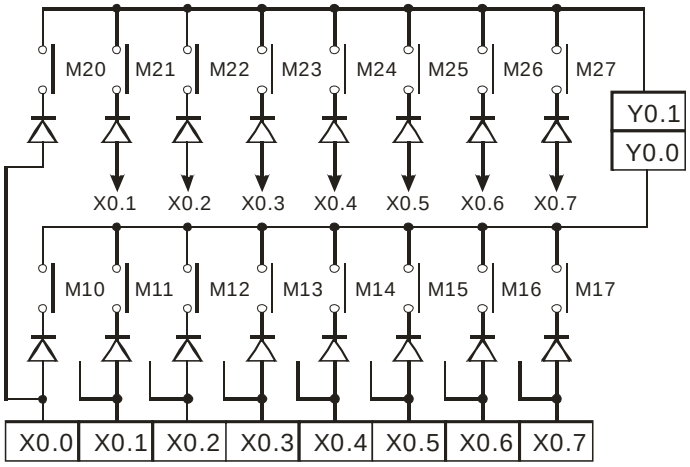
1. **S** 指定矩形所有连接输入端的起始号码，一旦指定后，该号码开始算连续 8 点为矩阵输入端。
2. **D_1** 指定那一个晶体管输出 Y 为矩阵扫描的起始号码，条件接点 OFF 时，**D_1** 起始的 **n** 个装置状态会保持 OFF。
3. 每一次扫描周期可以更新一列的输入点，每一列为 16 点，从 1~**n** 列反复的扫描。
4. 本指令由 **S** 起始的连续 8 个输入端 ,以 **D_1** 起始的 **n** 个外部输出点用矩阵扫描之方式读取 **n** 列的 8 个开关，变成 8×**n** 的多点矩阵输入点。并将扫描读取的开关状态反应在 **D_2** 起始之装置。
5. 使用本指令时，最大可将 8 个输入开关并接 8 列可得 64 个输入点（8×8=64）。
6. 本指令的执行间隔必须比模块 I/O 更新的时间长，否则无法读到正确的输入点。
7. 本指令的条件接点一般都使用常 ON 接点 SM400。
8. n 操作数的范围 2~8。

程序范例一：

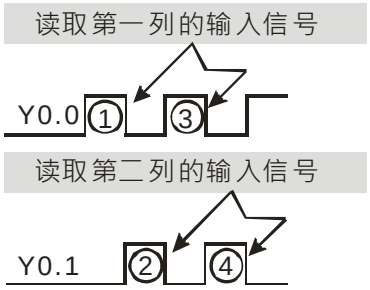
1. 当 M0=ON ,MTR 指令开始执行 ,外部 2 列共 16 个开关的状态被顺序读取并存放在内部继电器 M10~M17，M20~M27。



2. 下图由 X0.0~X0.7 及 Y0.0~Y0.1 构成 2 列矩阵输入回路的外部接线图，16 个开关对应到内部继电器 M10~M17，M20~M27。搭配 MTR 指令使用。



3. 以上图为例，X0.0 开始算的 8 点由 Y0.0~Y0.1（n=2）作矩阵扫描，**D_2** 指定读入结果的起始号码为 M10，代表第一列的开始被读入至 M10~M17，第二列被读入至 M20~M27。



补充说明：

1. 本指令被执行时，当扫描周期太长或太短都可能造成击键不良因此可运用下列技巧来克服。
2. 当扫描周期太短时，可能造成 I/O 来不及反应而无法读取正确的击键，此时，可将扫描时间加以固定。
3. 当扫描周期太长时，可能会使按键反应变得迟钝，可将此指令写在时间中断工作（TASK）内，固定时间执行此指令。
4. **S**+7、**D_1**+**n**-1、**D_2**+（**n***8）-1 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
5. **n**<2 或 **n**>8 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
6. **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [8] of BOOL。

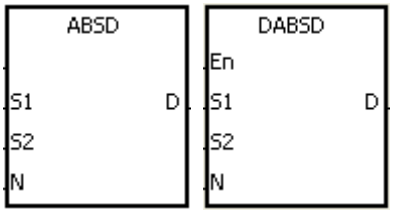
API	指令码			操作数	功能
0705	D	ABSD		**S_1 · S_2 · D · n**	绝对方式凸轮控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●							
S_2	●	●			●	●	●	●	●							
D		●	●	●				●								
n	●	●						●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D	●												
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



S_1 ： 指比较表起始装置

S_2 ： 比较值

D ： 比较结果起始编号

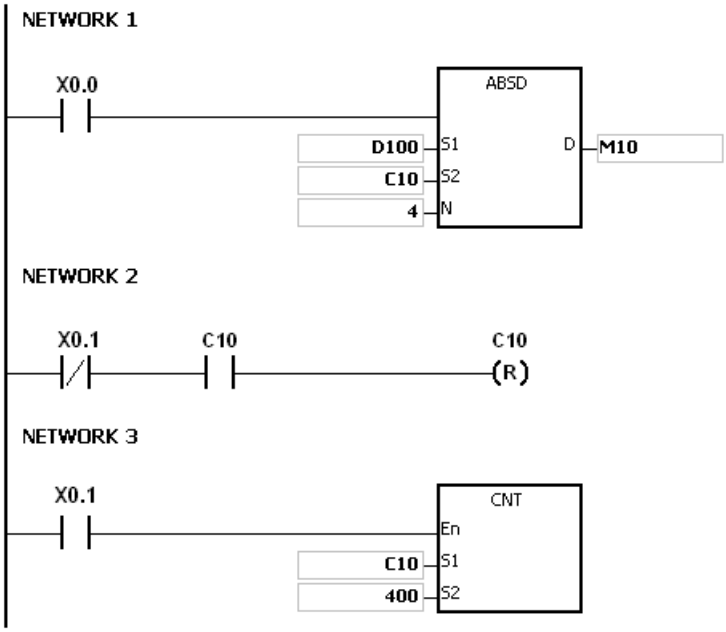
n ： 多段比较的组数

指令说明：

1. ABSD 指令为为对应计数器的当前值产生多个输出波形的指令，通常用来做绝对方式凸轮控制。
2. DABSD 指令才可以使用 HC 装置，但不可使用 E 装置。
3. **n** 操作数范围为=1~256。

程序范例一：

1. 于 ABSD 指令被执行前使用 MOV 指令预先将各设定值写入至 D100~D107 偶数 **D** 号码的内容为下限值，奇数 **D** 号码的内容为上限值。
2. 当 X0.0=ON 时，计数器 C10 的当前值与 D100~D107 等 4 组上下限值作区域比较，结果分别反应在 M10~M13。
3. X0.0=OFF 时，原 M10~M13 的 ON/OFF 状态不会变化。

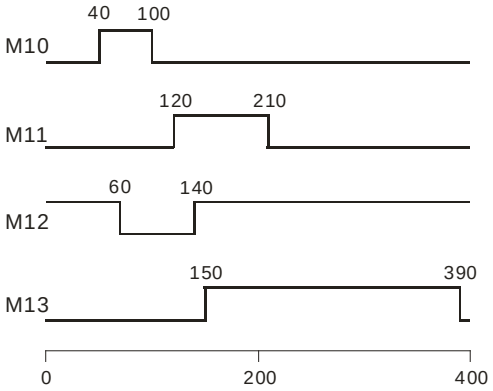


4. 在大于等于下限值且小于等于上限值范围内对应的 M10~M13 会 ON，否则为 OFF。

下限值	上限值	C10 当前值	输出
D100=40	D101=100	40≦C10≦100	M10=ON
D102=120	D103=210	120≦C10≦210	M11=ON
D104=140	D105=170	140≦C10≦170	M12=ON
D106=150	D107=390	150≦C10≦390	M13=ON

5. 若下限值大于上限值时，则小于上限值（C10 < 60）或大于下限值（C10 > 140）时，M12=ON，否则为 OFF。

下限值	上限值	C10 当前值	输出
D100=40	D101=100	40≦C10≦100	M10=ON
D102=120	D103=210	120≦C10≦210	M11=ON
D104=140	D105=60	60≦C10≦140	M12=OFF
D106=150	D107=390	150≦C10≦390	M13=ON



补充说明：

1. ABSD 指令中，**S**+2*__n__-1 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. DABSD 指令中，**S**+4*__n__-1 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. ABSD 指令中，**D**+__n__-1 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. DABSD 指令中，**D**+2*__n__-1 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
5. 若 **n**<1 **或 n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

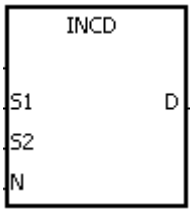
API	指令码	操作数	功能
0706	INCD	**S_1 · S_2 · n · D**	相对方式凸轮控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●							
S_2	●	●			●	●		●	●							
D		●	●	●				●								
n	●	●						●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D	●												
n		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 指比较表起始装置

S_2 ： 计数器编号

D ： 比较结果起始编号

n ： 多段比较的组数

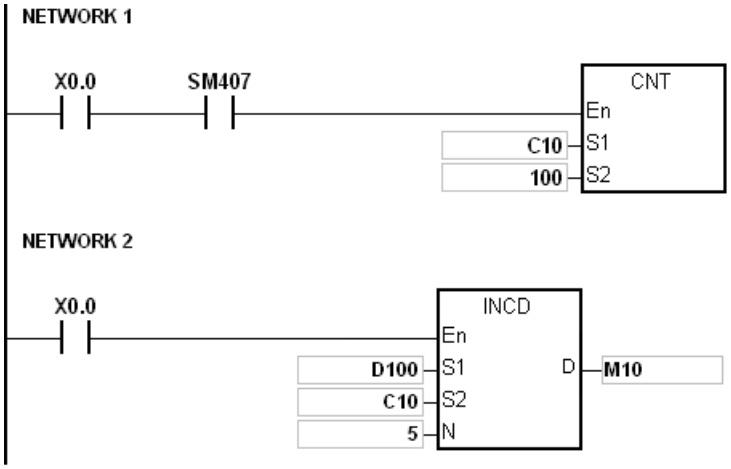
指令说明：

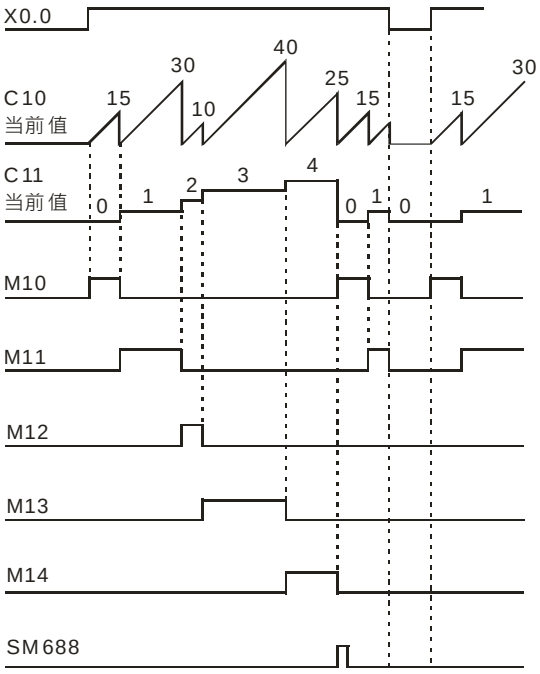
1. INCD 指令为用一对计数器产生多个输出波形的指令，通常用来作相对方式凸轮控制。
2. **S_2** 的当前值与 **S_1** 的设定值作比较，每到达一个设定值，**S_2** 的当前值被复位为 0 重新计数。而目前执行的段数被暂存于 **S_2**+1 当中。
3. **n** 的组数比较完成时，指令执行完毕标志 SM688 会 ON 一次扫描周期。
4. 当条件接点未启动时，**S_2**、**S_2**+1 的内容值为 0，**D~D+n-1** 状态为 OFF，SM688 为 OFF。
5. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。
6. **n** 操作数的范围 1~256。

程序范例：

1. 于 INCD 指令被执行前，使用 MOV 指令预先将各设定值写入至 D100~D104 当中，D100=15、D101=30、D102=10、D103=40、D104=25。
2. 计数器 C10 的当前值与 D100~D104 的设定值作比较，每到达一个设定值，C10 的当前值被复位为 0 重新计数。
3. 而目前执行之段数被暂存于 C11 当中。
4. C11 的内容每变动 1 时，M10~M14 相对应动作，请参考下列时序图。

5. 5 组数比较完成时，指令执行完毕标志 SM688 会 ON 一次扫描周期。
6. 当 X0.0 由 ON 变成 OFF 时，C10 及 C11 全部被复位为 0，M10~M14 亦全部变成 OFF，当 X0.0 再度 ON 时，本指令被从头执行起。





补充说明：

1. **S_2+1** 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **S_1+n-1** 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **D+n-1** 超出装置范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. **n<**1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
5. **S_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。

API	指令码			操作数	功能
0708	D	PIDE		依序如下表列	PID 运算

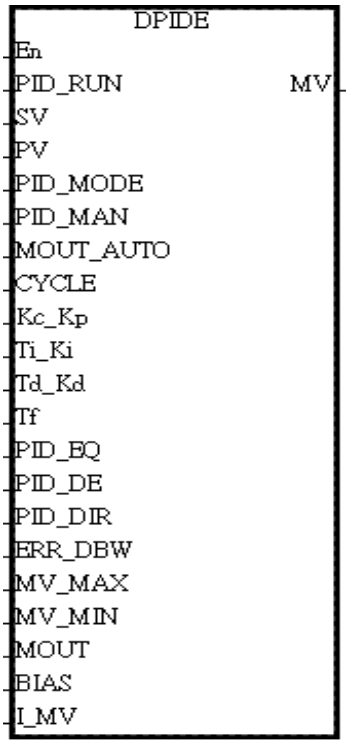
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
PID_RUN	●	●	●	●				●								
SV								●								○
PV								●								○
PID_MODE								●					○	○		
PID_MAN	●	●	●	●				●								
MOUT_AUTO	●	●	●	●				●								
CYCLE								●								
KC_Kp								●								
Ti_Ki								●								
Td_Kd								●								
Tf								●								
PID_EQ	●	●	●	●				●								
PID_DE	●	●	●	●				●								
PID_DIR	●	●	●	●				●								
ERR_DBW								●	●							○
MV_MAX								●	●							○
MV_MIN								●	●							○
MOUT								●								
BIAS								●	●							○
I_MV								●								
MV								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING

数据类型请参考下页操作数说明

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



EN ：启动/关闭指令
PID_RUN ：启动 PID 运算
SV ：目标值
PV ：当前值
PID_MODE ：PID 控制模式
PID_MAN ：PID Auto/Manual 模式
MOUT_AUTO ：手动值（MOUT）自动更新模式
CYCLE ：取样时间（ms）
Kc_Kp ：比例项系数
Ti_Ki ：积分项系数（sec 或 1/sec）
Td_Kd ：微分项系数（sec）
Tf ：微分作用时间常数（sec）
PID_EQ ：PID 计算式选择
PID_DE ：PID 微分项误差计算选择
PID_DIR ：PID 正反向
ERR_DBW ：ERR 的不作用范围
MV_MAX ：MV 输出上限值
MV_MIN ：MV 输出下限值
MOUT ：MV 手动值
BIAS ：前馈控制输出值
I_MV ：误差累积值与系统使用暂存区
MV ：输出值

指令说明：

1. PID 进阶运算控制的专用指令，当指令被主机执行时才会进行 PID 运算与处理。PID 表示“比例、积分和微分”。PID 控制在机械设备、气动设备和电子设备中具有广泛的应用。
2. 参数设定内容如下：

操作数名称	数据类型	功能	设定范围	说明
PID_RUN	BOOL	启动 PID 运算	TRUE：PID 指令开始运算。 FALSE：停止运算，且 MV 值清除为 0。	
SV	REAL	SV 值	单精度浮点数范围	目标值
PV	REAL	PV 值	单精度浮点数范围	当前值

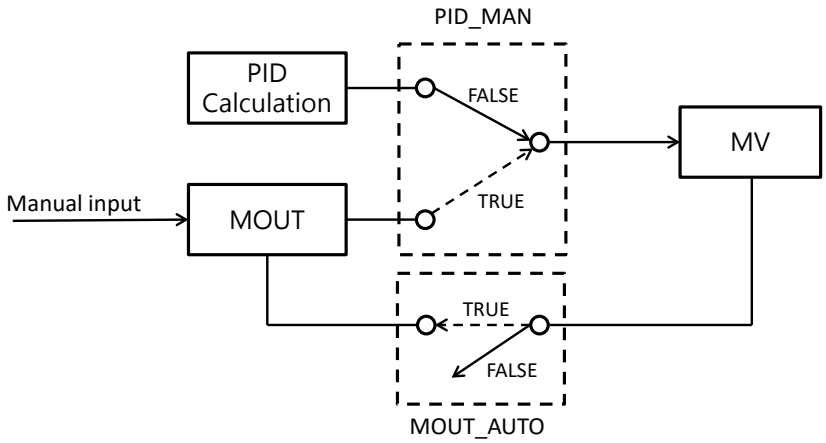
操作数名称	数据类型	功能	设定范围	说明
PID_MODE	DWORD/DINT	PID 控制模式	0：自动控制，当 PID_MAN 由 TRUE 转为 FALSE 时，MV 值会由当时输出的 MV 值开始进行自动运算。当 MV 超出 MV_MAX 或 MV_MIN 时，I_MV 值不会再改变。 1：自动调整参数功能，调整完毕时将自动进入自动控制模式（PID_MODE 改为 0），并且填入最适用的 Kc_Kp、Ti_K_i、Td_K_d 及 Tf 等参数。 2：自动控制(包含 I_MV 修正)，当 PID_MAN 由 TRUE 转为 FALSE 时，MV 值会由当时输出的 MV 值开始进行自动运算。当 MV 超出 MV_MAX 或 MV_MIN 时，I_MV 值会依公式反推数值自动修正，此做法可以加快反向控制的响应时间。 备注：使用自动调整参数功能时，此模式不可使用数值直接设定。	
PID_MAN	BOOL	PID A/M 模式	TRUE：Manual，MV 值会依 MOUT 值输出，但仍在 MV_MAX 与 MV_MIN 之间，当 PID_MODE 为 1 时此设定无效。 FALSE：Auto，MV 值会依 PID 公式计算后输出，输出值在 MV_MAX 与 MV_MIN 之间。	
MOUT_AUTO	BOOL	MOUT 自动更新模式	TRUE：Auto，MOUT 的值会随着 MV 值改变。 FALSE：Normal，MOUT 的值不会随着 MV 值改变。	

操作数名称	数据类型	功能	设定范围	说明
CYCLE	DWORD/DINT	取样时间（T_S）	1~40,000（单位: ms）	每次扫描到本指令时，就会以 CYCLE 设定的取样时间来计算 PID（主机并不会依照此数值来自动判断时间并自动执行），并更新输出值（MV）。 CYCLE 小于 1 则内定为 1，大于 40,000 时则为 40,000。 当 PID 指令用于时间中断时，主机会自动以时间中断的中断时间来计算 PID，此时 CYCLE 的设定无效。
Kc_Kp	REAL	比例项系数（Kc or Kp，依 PID_EQ 参数决定使用何种系数）	正数单精度浮点数范围	P 计算值系数，如果小于 0 则 Kc_Kp 将为 0，在 Independent 下若 Kc_Kp 等于 0 则表示不使用 P 控制 。
Ti_Ki	REAL	积分项系数（Ti or Ki，依 PID_EQ 参数决定使用何种系数）	正数单精度浮点数范围（单位: Ti = sec; Ki = 1/sec）	I 计算值系数，如果小于 0 则 Ti_Ki 将为 0，当 Ti_Ki 等于 0 时则表示不使用 I 控制。
Td_Kd	REAL	微分项系数（Td or K_d，依 PID_EQ 参数决定使用何种系数）	正数单精度浮点数范围（单位: sec）	D 计算值系数，如果小于 0 则 Td_Kd 将为 0，当 Td_Kd 等于 0 时则表示不使用 D 控制。
Tf	REAL	微分作用时间常数（Tf）	正数单精度浮点数范围（单位: sec）	微分作用时间常数，如果小于 0 则 Tf 将为 0，当 Tf 等于 0 时则表示不使用微分作用时间的控制（Derivative Smoothing）。
PID_EQ	BOOL	PID 计算式选择	TRUE：Dependent Formula FALSE：Independent Formula	

操作数名称	数据类型	功能	设定范围	说明
PID_DE	BOOL	PID 微分项误差计算选择	TRUE：使用当前值 (PV) 的变化量来计算微分项的控制值(Derivative of PV) FALSE：使用偏差量 (E) 的变化量来计算微分项的控制值(Derivative of E)	
PID_DIR	BOOL	PID 正反向	TRUE：正向动作 (E=SV-PV) FALSE：反向动作 (E=PV-SV)	
ERR_DBW	REAL	偏差量 (E) 不作用范围	单精度浮点数范围	偏差量 (E) 等于 SV−PV 或 PV-SV ，当 ERR_DBW 设定为 0 时即表示不启动此功能，否则主机会去检查这次的 E 值是否小于 ERR_DBW 的绝对值且是否符合 Cross 状态转换条件，若都有则将 E 值视为 0 之后进行 PID 计算，否则依照正常处理将 E 值代入 PID 计算。
MV_MAX	REAL	输出值 (MV) 饱和上限	单精度浮点数范围	例：设定 1,000，则输出值 (MV) 大于 1,000 时将以 1,000 输出。 此设定值需大于 MV_MIN，否则主机会自动将 MV_MAX 与 MV_MIN 进行互换。
MV_MIN	REAL	输出值 (MV) 饱和下限	单精度浮点数范围	例：设定-1,000，则输出值 (MV) 小于-1,000 时将以-1,000 输出。 此设定值需小于 MV_MAX，否则主机会自动将 MV_MAX 与 MV_MIN 进行互换。

操作数名称	数据类型	功能		设定范围	说明
MOUT	REAL	MV 手动值		单精度浮点数范围	配合 PID_MAN 模式使用，当 PID 设定在手动模式(PID_MAN=True)时，则输出值（MV）会依照手动设定值（MOUT）输出，但是仍然在 MV_MAX 与 MV_MIN 之间。
BIAS	REAL	前馈控制输出值		单精度浮点数范围	使用于 PID 前馈控制。
I_MV(**连续**占用 15 个 DWord 装置)	REAL	I_MV	误差累积值 $\int_0^t Edt$	单精度浮点数范围	为累积的误差值(包含积分项系数)，通常只供参考用，但是用户还是可以依需求清除或修改，不过须以单精度浮点数修改之， PID 控制模式为 0 自动模式时，当 MV 超出 MV_MAX 或 MV_MIN 时，此累积值不会再改变。 PID 控制模式为 2 自动模式(包含 I_MV 修正)时，当 MV 超出 MV_MAX 或 MV_MIN 时，此累积值会依公式反推数值自动修正。
	REAL	I_MV+1	暂存前次 E 值	系统纪录前次偏差量。	
	DWord	I_MV+2 ~ I_MV+5	系统用参数，用户请勿使用。		
	REAL	I_MV+6	系统纪录前次 PV 值。		
	DWord	I_MV+7 ~ I_MV+14	系统用参数，用户请勿使用。		
MV	REAL	MV 输出值	MV 值会介于 MV_MAX 与 MV_MIN 之间。		

3. PID_MAN 与 MOUT_AUTO 切换说明如下图所示：

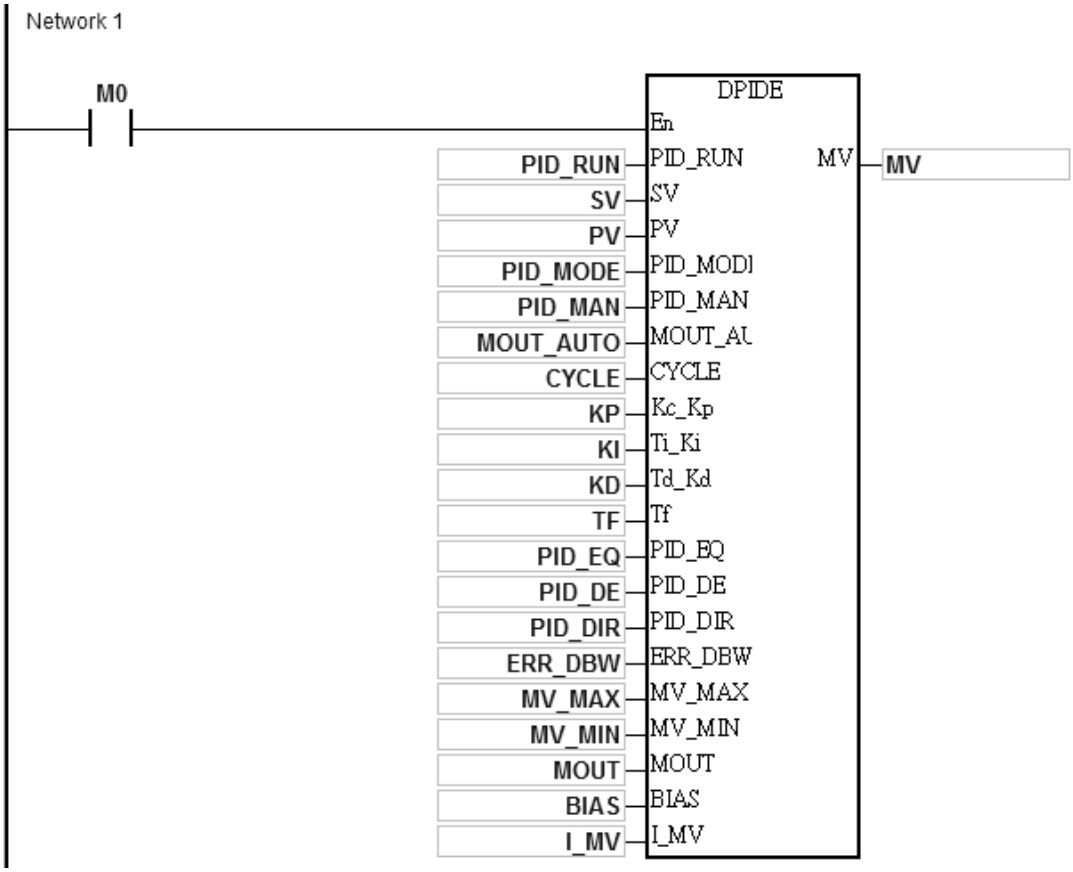


当控制模式(**PID_MAN**=0)执行自动模式要切换到手动模式前，可利用 **MOUT_AUTO**=1 标志，让 **MOUT** 输出值跟随 MV 输出值变化，待切换至手动模式（**PID_MAN**=1）之后，可再关闭 **MOUT_AUTO**=0。

4. 当 **PID_RUN** 由 TRUE→FALSE 时，会自动清除 **MV** 输出值为 0。若希望 **MV** 输出值能够在 PID 不执行时，继续保持最后一次 **MV** 输出值，则可利用清除 **EN** 为 FALSE 去关闭指令，而且不清除 **MV** 输出值。

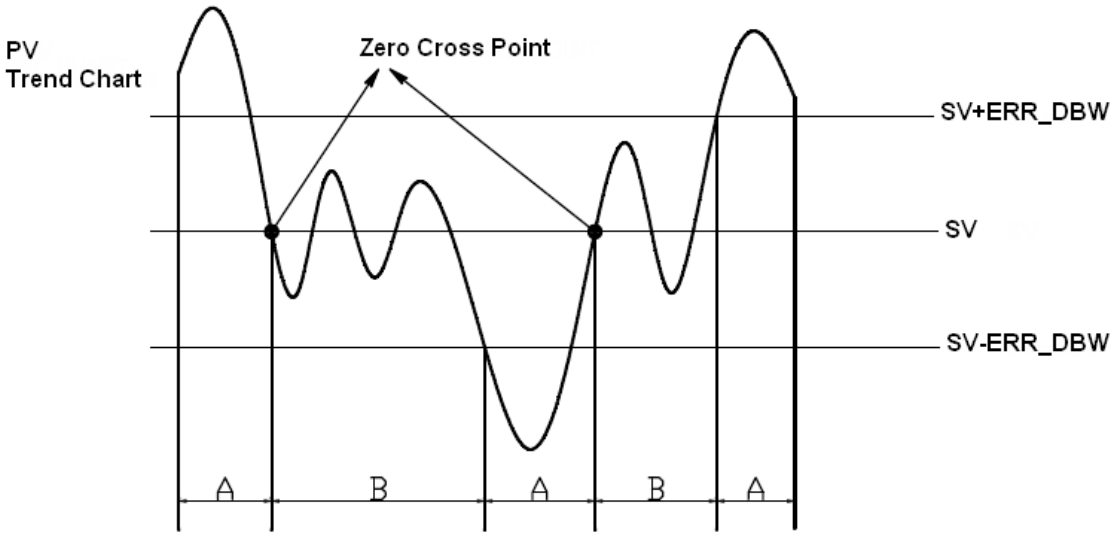
程序范例：

1. 执行 DPIDE 指令前先将参数设定完成。
2. M0=ON 时指令被执行，**PID_RUN**=ON 时 DPIDE 指令开始进行运算，**PID_RUN**=OFF 时 **MV** 值为 0，数值送至 **MV**。当 M0 变成 OFF 时，指令不被执行，原先指令内的参数值没有变化。



补充说明：

1. 指令无使用次数的限制，但是 **I_MV~I_MV**+14 所指定的寄存器号码不可被其它程序重复使用。
2. **I_MV** 占用 30 个 Word 寄存器。
3. DPIDE 指令只能使用在周期性工作（Cyclic Task）与时间中断当中，当指令用于时间中断时，取样时间（**CYCLE**）会自动以时间中断的时间为主。
4. DPIDE 只要被扫描到都会以取样时间（**CYCLE**）所设定的时间来做 PID 运算并直接输出 MV 值，并不会自动计算扫描时间是否有到达取样时间（**CYCLE**）才输出，但如果本指令使用在时间中断程序内，取样时间（**CYCLE**）将与时间中断程序的间隔时间相同，DPIDE 指令会以时间中断程序的中断间隔时间来做 PID 的运算。
5. DPIDE 的当前值（**PV**）在 PID 执行运算动作前必须是一个稳定值。如果要抓取特殊模块的输入值作 PID 运算时，请注意这些模块的 A/D 转换时间。
6. 当 **PV** 值进入 **ERR_DBW** 的范围时，一开始主机仍会依照 E 值进行 PID 计算，直到 **PV** 穿过 **SV** 值时(Zero Cross Point)代表 Cross Status 成立，此时会将 E 值视为 0 代入 PID 计算，一直到 PV 值超出 **ERR_DBW** 的范围时才会恢复将 E 值代入 PID 计算，若 PID_DE=True 则表示使用 PV 值来进行微分项的计算，则在 Cross Status 条件成立后，PLC 会将 **ΔPV** 视为 0 进行 PID 微分项的计算。(**ΔPV**=当前 **PV** - 前次 **PV**) 例如以下的 **PV** 趋势图中，A 的区段主机会依照正常的 PID 进行计算，而 B 的区段主机会将 E 或 **ΔPV** 视为 0 进行 PID 计算。



PID 计算公式：

1. 当 **PID_MODE** 控制模式设定为 0 时，为自动控制模式。

- **Independent Formula & Derivative of E**（**PID_EQ**=False & **PID_DE**=False）

$$MV = K_P E + Ki\int_0^t Edt + K_d * \frac{dE}{dt} + BIAS$$ **其中** $E = SV - PV$ or $E = PV - SV$

- **Independent Formula & Derivative of PV（PID_EQ=False & PID_DE=Ture）**

$$MV = K_P E + Ki\int_0^t Edt - K_d * \frac{dPV}{dt} + BIAS$$ **其中** *E = SV – PV*

Or

$$MV = K_P E + Ki\int_0^t Edt + K_d * \frac{dPV}{dt} + BIAS$$ **其中** *E =PV – SV*

- **Dependent Formula & Derivative of E**（**PID_EQ**=True & **PID_DE**=False）

$$MV = K_c\left[E + \frac{1}{T_i}\int_0^t Edt + T_d * \frac{dE}{dt}\right] + BIAS$$ **其中** *E = SV – PV* or *E = PV – SV*

- **Dependent Formula & Derivative of PV**（**PID_EQ**=True & **PID_DE**=True）

$$MV = K_c\left[E + \frac{1}{T_i}\int_0^t Edt - T_d * \frac{dE}{dt}\right] + BIAS$$ **其中** *E = SV – PV*

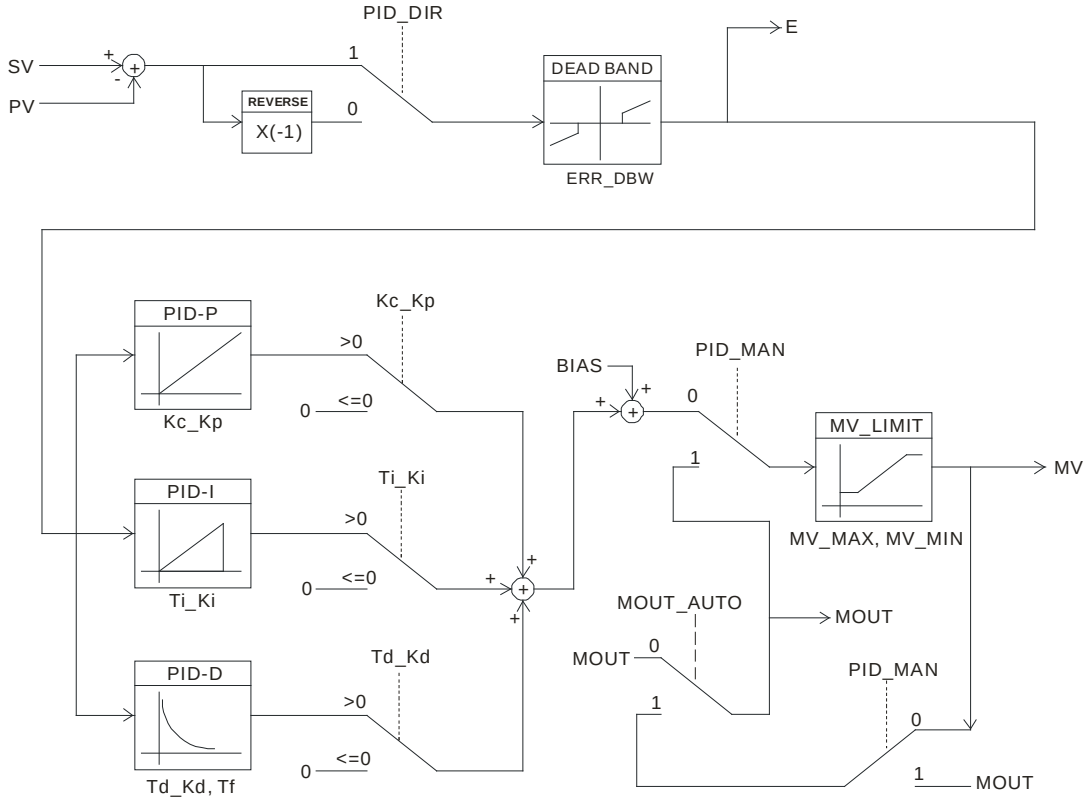
Or

$$MV = K_c\left[E + \frac{1}{T_i}\int_0^t Edt + T_d * \frac{dE}{dt}\right] + BIAS$$ **其中** *E = PV – SV*

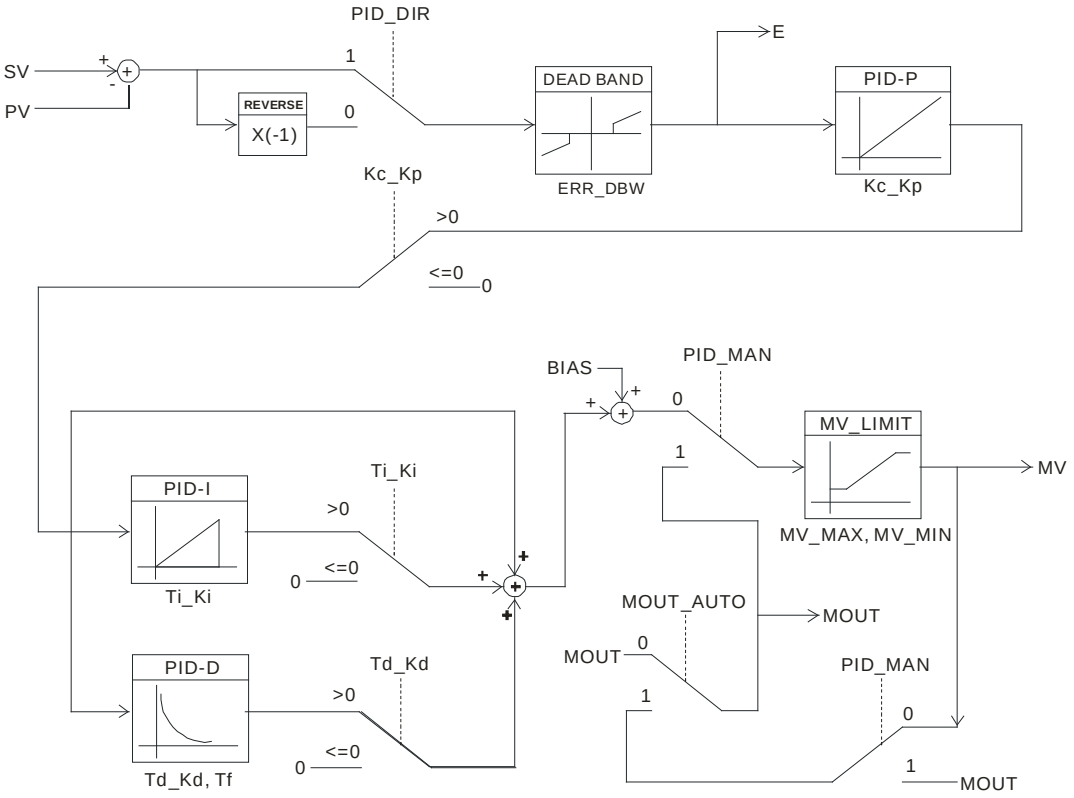
2. 当 **PID_MODE** 控制模式选择为 1，为自动调整模式，当自动调整完成后，**PID_MODE** 会自动变成 0 转换为自动控制模式。

PID 控制方块图：

PID Block Diagram (Independent)



PID Block Diagram (Dependent)



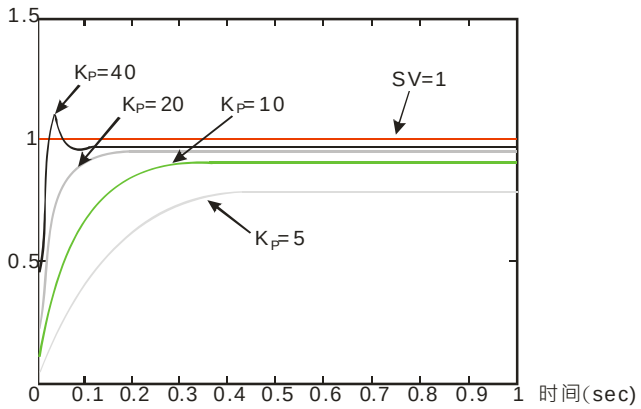
注意事项和建议：

1. 由于可使用 DPIDE 指令的控制环境很多，因此请适当的选取控制功能，例如：当选择自动调整参数（**PID_MODE**=1）功能时，**MV** 值会在 **MAX / MIN** 两个值之间做切换，所以请尽量勿使用于快速反应的马达控制环境中，以免造成在自动调适时系统因快速的剧烈变化，而造成对人员危险或损坏系统的现象发生。
2. 用户于调整 **Kc_Kp**、**Ti_Ki** 及 **Td_Kd** 三个主要参数时（**PID_MODE**=0），请先调整 **Kc_Kp** 值（依经验值设定），而 **Ti_Ki** 及 **Td_Kd** 值先设定为 0，等到调整到大致上可控制时，再依序调整 **Ti_Ki** 值（由小到大）以及 **Td_Kd** 值（由小到大）。其中 **Kc_Kp** 值为 1 则表示 100%，即对偏差值的增益为 1，小于 100%将对偏差值衰减，大于 100%将对偏差值放大。
3. 当用户选用（**PID_MODE**=1）时，建议请使用在停电保持区的 D 寄存器来储存参数，以免自动调整过的参数因停电后而消失。经过自动调整过的参数，并不能保证一定适用于每个控制的环境，因此用户当然可自行修改调整过的参数，不过建议最好只修改 **Ti_Ki** 或 **Td_Kd** 数值就好。
4. 本指令动作须配合许多参数值控制，因此请勿随意设定参数值，以免造成无法控制的现象。

范例 1：PID 指令参数调整建议步骤说明

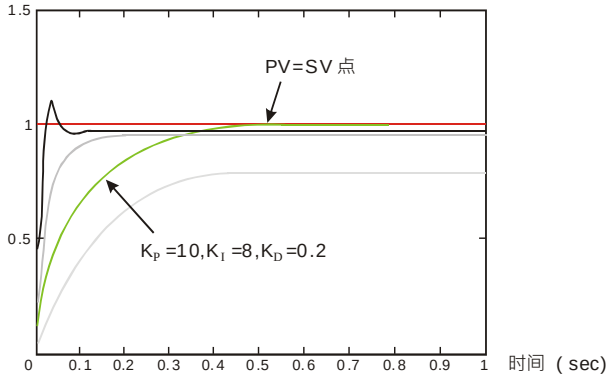
假设控制系统的受控体 G (s) 的转移函数为一阶的函数 $G(s)=\frac{b}{s+a}$（一般马达的模型均为此函数），命令值 SV 为 1，取样时间 **CYCLE** 为 10ms。建议调整步骤如下：

步骤 1：首先将 K_i 及 K_d 值设为 0，接着先后分别设定 K_p 为 5、10、20 及 40，并分别记录其 SV 及 PV 状态，其结果如下图所示。



步骤 2：观察上图后得知 ***Kp*** 为 40 时，其反应会有过冲现象，因此不选用，而 ***Kp*** 为 20 时，其 **PV** 反应曲线接近 **SV** 值且不会有过冲现象，但是由于启动过快，因此输出值 **MV** 瞬间值会很大，所以考虑暂不选用，接着 ***Kp*** 为 10 时，其 **PV** 反应曲线接近 **SV** 值并且是比较平滑接近，因此考虑使用此值，最后 ***Kp*** 为 5 时，其反应过慢，因此也暂不考虑使用。

步骤 3：选定 ***Kp*** 为 10 后，先调整 ***Ki*** 值由小到大（如 1、2、4 至 8），以不超过 ***Kp*** 值为原则，然后再调整 ***Kd*** 由小到大（如 0.01、0.05、0.1 及 0.2），以不超过 ***Kp*** 的 10%为原则，最后可得如下图之 **PV** 与 **SV** 的关系图。

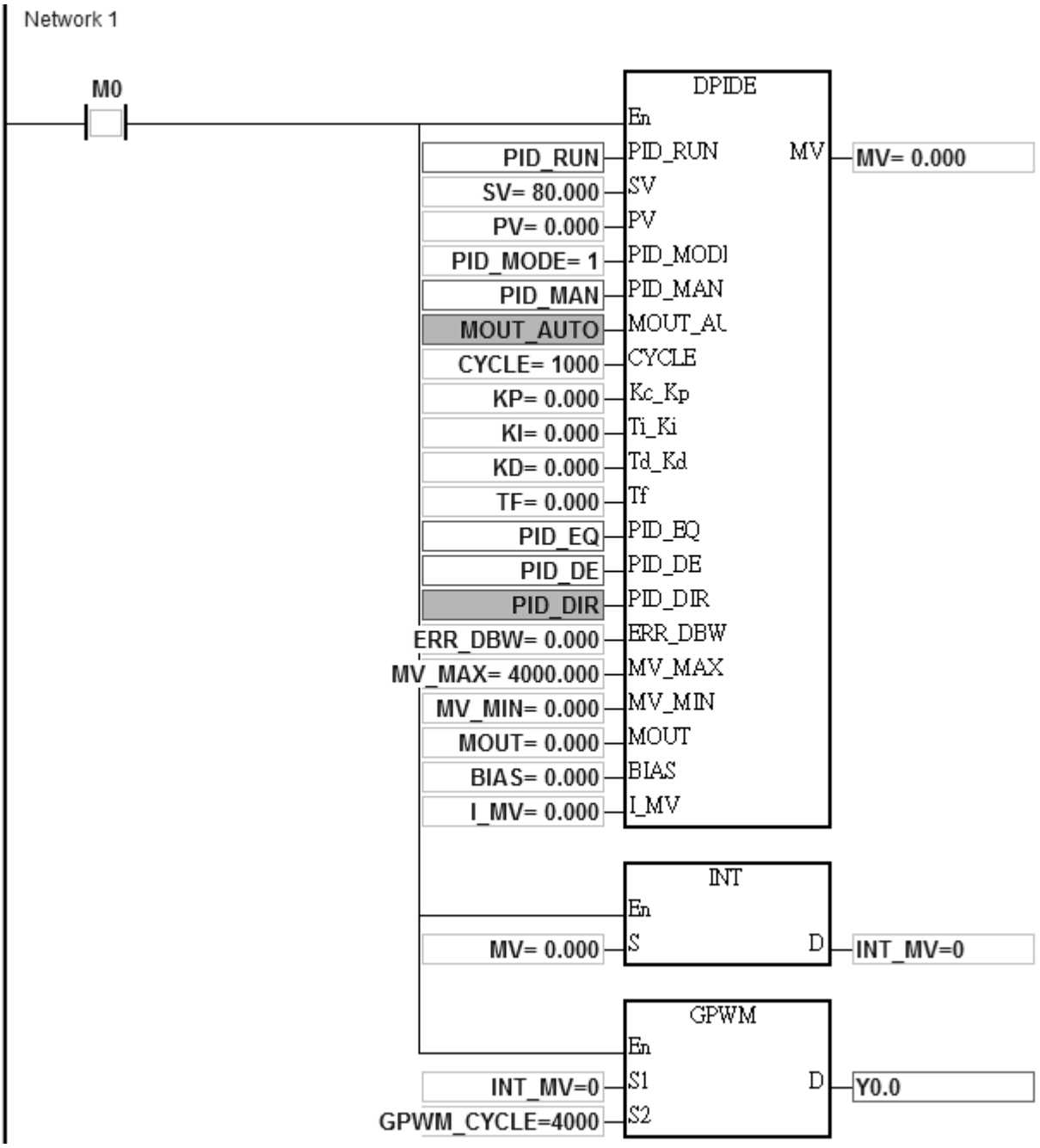


附注：本范例仅供参考，因此用户还需依实际控制系统的状况，自行调整其适合的控制参数。

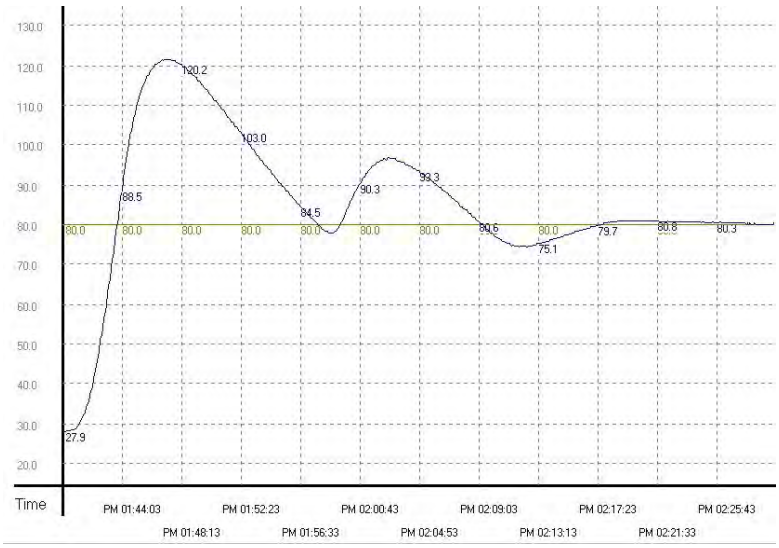
程序范例三：使用自动调整功能控制温度

由于一般用户对于第一次控制的温度环境特性通常不太了解，因此可先使用自动调整功能（**PID_MODE**=1）做一初步调整，待调整完毕后，本指令将自动修改控制模式为（**PID_MODE**=0）。本实例的控制环境为烤箱。

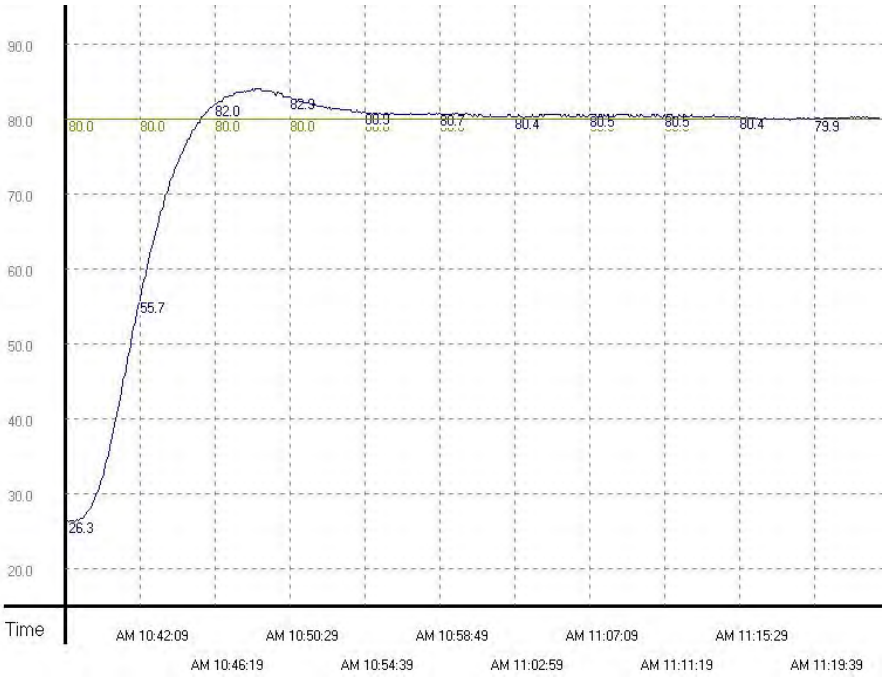
范例程序及参数设定如下图所示：



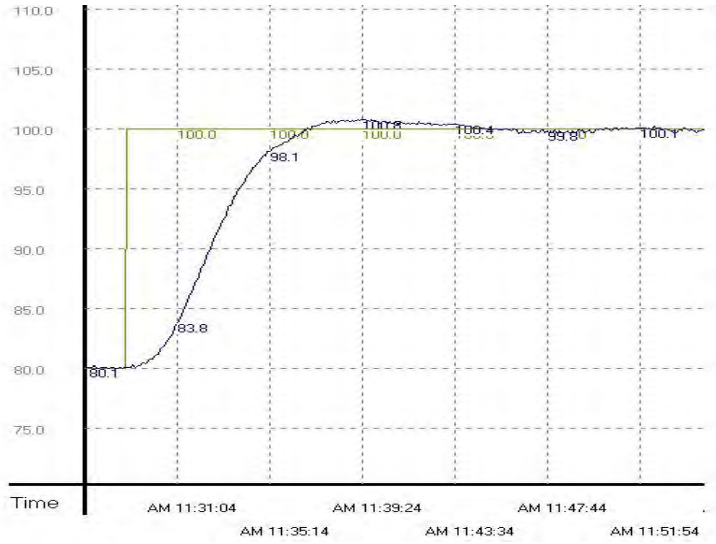
自适应功能的实验结果如下所示：



使用调整后参数做温度控制专用功能的实验结果如下所示：



由上图可看出经过自适应后的温度控制结果还不错，而且控制时间大约只使用了 20 分钟。接着验证目标温度由 80 度修改成 100 度，则得到的结果如下图所示：



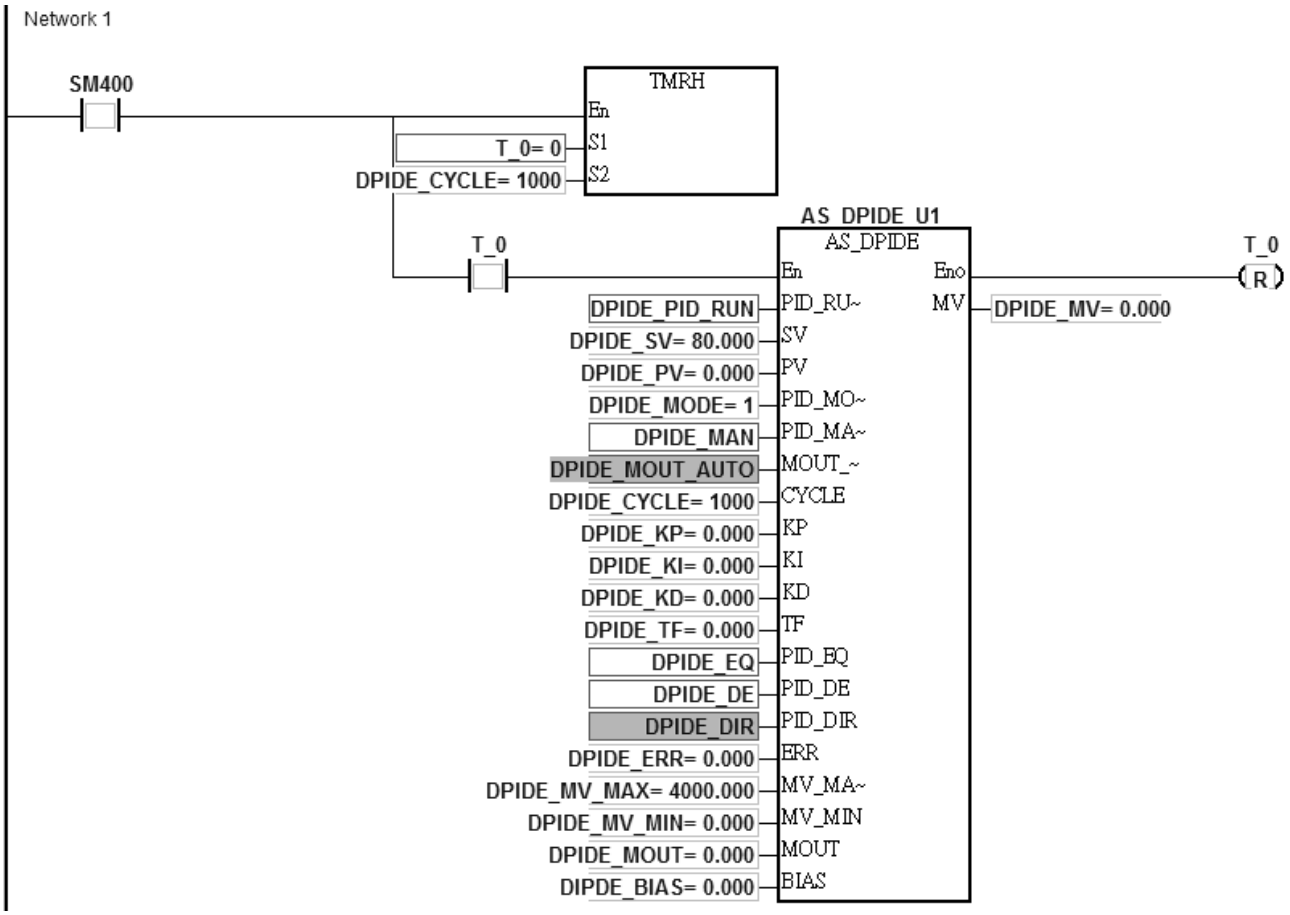
由上图中可看出由 80 度所调整出来的参数使用到 100 度时，还是可以达到控制温度的目的，而且控制时间也不会太长。

程序范例四：DPIDE 指令建立在功能块内，通过周期性工作（Cyclic Task）使用 DPIDE 功能块，达到控制温度的目的。(注：此范例是使用周期性 POU 编写 DPIDE 的方式，其复杂度比范例五高)

1. 主程序中设定取样时间1000ms（DPIDE_CYCLE），通过功能块方式执行DPIDE指令，每次扫描到本功能块时，会以设定的取样时间来计算PID并更新输出值（DPIDE_MV）。
2. 使用自动调整功能（DPIDE_MODE =1）做初步调整，待调整完毕后，本指令将自动修改控制模式为（DPIDE_MODE =0）。范例程序如下图所示：

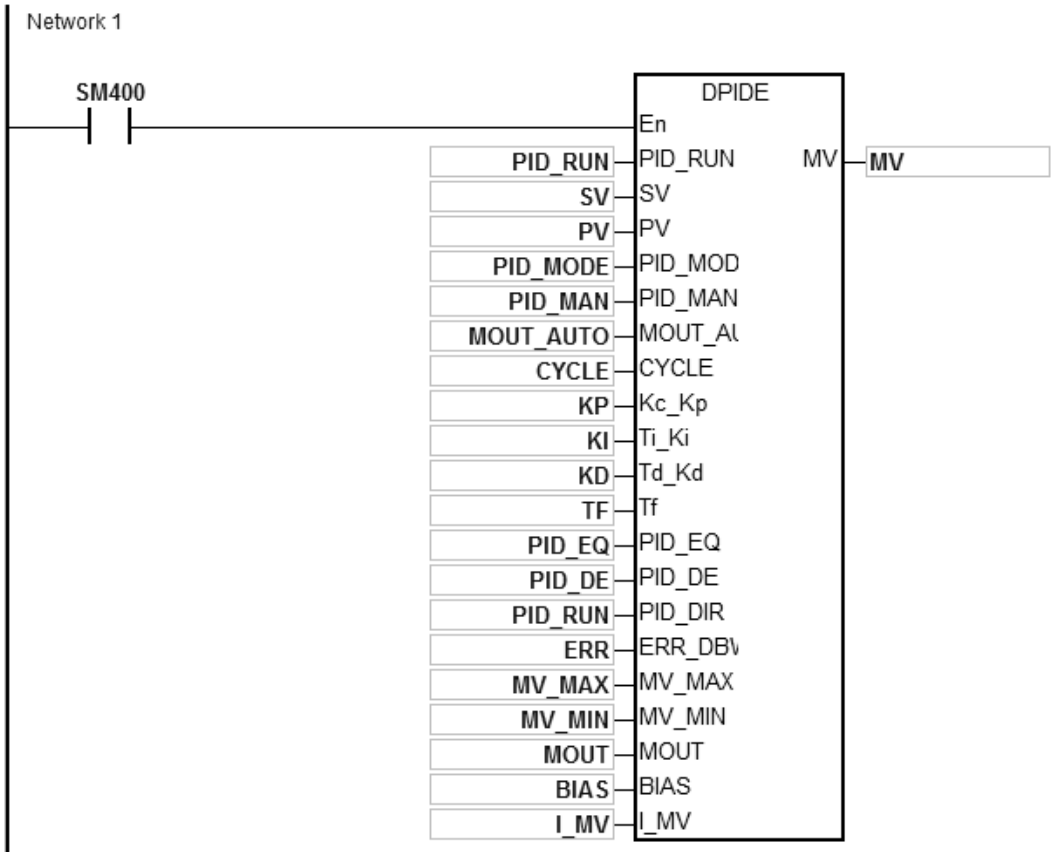
主程序（周期性工作）：

程序内容说明：利用 TMRH 计时 1000ms，在计时到达时，调用一次 AS_DPIDE 这个功能块，范例程序及参数如下图所示。(注：因 DPIDE 指令的所设定的 **CYCLE** 时间，PLC 程序必须配合指定的时间到达时，才可执行一次 DPIDE 指令)



功能块

1. AS_DPIDE 内容说明：执行 DPIDE 指令，并将 PID 运算后的 MV 值输出。（注：如何建立功能块的输入与输出引脚，请自行参考软件手册）
2. 功能块内 DPIDE 指令的 PID_MODE、Kc_Kp、Ti_Ki、Td_Kd、Tf 及 I_MV 等六个引脚，其宣告变量符号之类别必须设定为 VAR_IN_OUT。

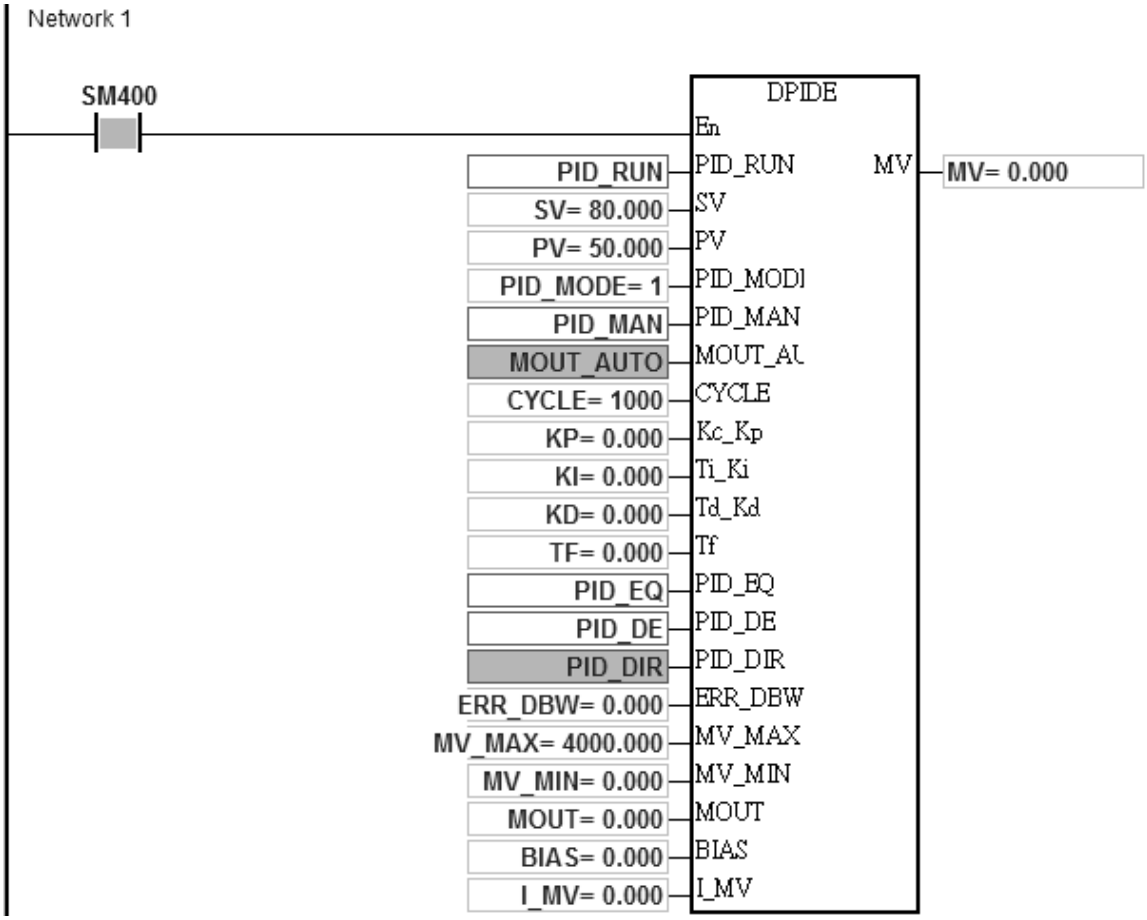


程序范例五：DPIDE 指令建立在定时中断程序内，达到控制温度的目的。(注：直接采用时间中断当做 DPIDE 的 CYCLE 时间设定)

1. HWCONFIG之定时中断0设定为1000ms。
2. DPIDE指令建立在定时中断0程序中，主机自动以时间中断的中断时间来计算PID，此时CYCLE的设定无效。
3. 使用自动调整功能（PID_MODE=1）做初步调整，待调整完毕后，本指令将自动修改控制模式为（PID_MODE=0）。范例程序如下图所示：

主程序（周期性工作）：

定时中断I601程序及参数设定：



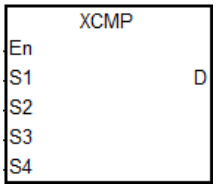
API	指令码	操作数	功能
0709	XCMP	**S_1 · S_2 · S_3 · S_4 · D**	多任务位输入比较设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	○															
S_2							○									
S_3								○								
S_4								○								
D								○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2												●	
S_3		●				●							
S_4			●				●						
D			●				●						

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 实时触发输入点

S_2 ： 高速计数器编号

S_3 ： 工位与对象数量设定

S_4 ： 比较值与偏差值

D ： 储存比较值之堆栈区起始地址

指令说明：

1. V1.04 版（含）以上支持此指令。此指令不建议使用 ST 结构化语言编辑、中断程序以及仅被调用一次性的功能块（FB）程序。
2. 此指令建议与 API0710 YOUT 指令一起搭配使用。一般常被用于一条生产在线安排多处加工位，并且需要可同时进入多个工件依序排列加工之应用，例如：木材加工、喷漆加工...等等加工生产线。
3. **S_1** 操作数为实时触发输入点设定，当使用主机内建 X0.0~X0.15 输入点时，其输入点为立即型触发，若选其它 X 输入点时，则为一般型触发。此指令启动后，立即型触发将自动开启外部输入中断功能，因此建议用户不要使用输入点对应之中断服务程序，若有写，则当此指令启动时，中断服务程序将会被关闭，直到指令被关闭后才能恢复。一般型触发将受扫描时间影响，但适用于输入点容易弹跳的使用环境。
4. **S_2** 操作数为指定搭配之 32 位计数器（HC0~HC255），并且仅限采用累加方式计数。输入点为立即触发型输入点时，建议采用硬件高速计数器，并且自行搭配 DCNT 指令启动。若需要采用高速输出点搭配使用时，则可使用 DMOV 搬移现在输出位置（如：轴 1 的 SR）到 HC0。

5. **在 V1.04.00 版（含）以前**，S_3 操作数连续占用 3 个 16 位装置，S_3+0 为设定工位数量 n 个，S_3+1 为最多可处理对象数量 m 个，S_3+2 为检测物件时的滤波计数个数。上述 n，m 的数值范围为 1~32，超过此范围时以最小或最大值执行。S_3+2 滤波个数范围为 0~32767，小于 0 的数值皆以 0 执行，且 0 表示不进行滤波功能。（建议宣告 3 个 word 类型的矩阵变量，或自行指定 3 个连续 word 类型变量）
6. **在 V1.06.00 版（含）以后**，S_3 操作数连续占用 7 个 16 位装置，S_3+0 为设定工位数量 n 个，S_3+1 为最多可处理对象数量 m 个，S_3+2 为检测物件时的滤波计数个数。上述 n，m 的数值范围为 1~64，超过此范围时以最小或最大值执行。S_3+2 滤波个数范围为 0~32767，小于 0 的数值皆以 0 执行，且 0 表示不进行滤波功能。S_3+3（Low word）及 S_3+4（High word）为上升沿抓取计数值（32 位），S_3+5（Low word）及 S_3+6（High word）为下降沿抓取计数值（32 位）（建议宣告 7 个 word 类型的矩阵变量，或自行指定 7 个连续 word 类型变量）
7. 建议设定 S_3+1 最多可处理对象数量 m 大于工位数量 n，若是设定 m < n，则须特别管控入料数量进入正在制作在线。
8. S_4 操作数连续占用 3n 个 32 位装置（或 6n 个 16 位装置），若使用空间超出 D 装置范围，则指令不执行，其中 n 为 S_3 操作数所设定之工位数量，详细各个装置的功能名称与对应 S_4 编号如下表所示：（建议宣告 3n 个 Dword 类型的矩阵变量，或自行指定 3n 个连续 Dword 类型变量）

功能名称	工位 1	工位 2	・・・	工位 n
基准值（32 位）	S_4 + 0	S_4 + 2	・・・	S_4 + （n-1）x2
对象进入偏差值（32 位）	S_4 + 2n	S_4 + 2n+2	・・・	S_4 + （2n-1）x2
对象离开偏差值（32 位）	S_4 + 4n	S_4 + 4n+2	・・・	S_4 + （3n-1）x2

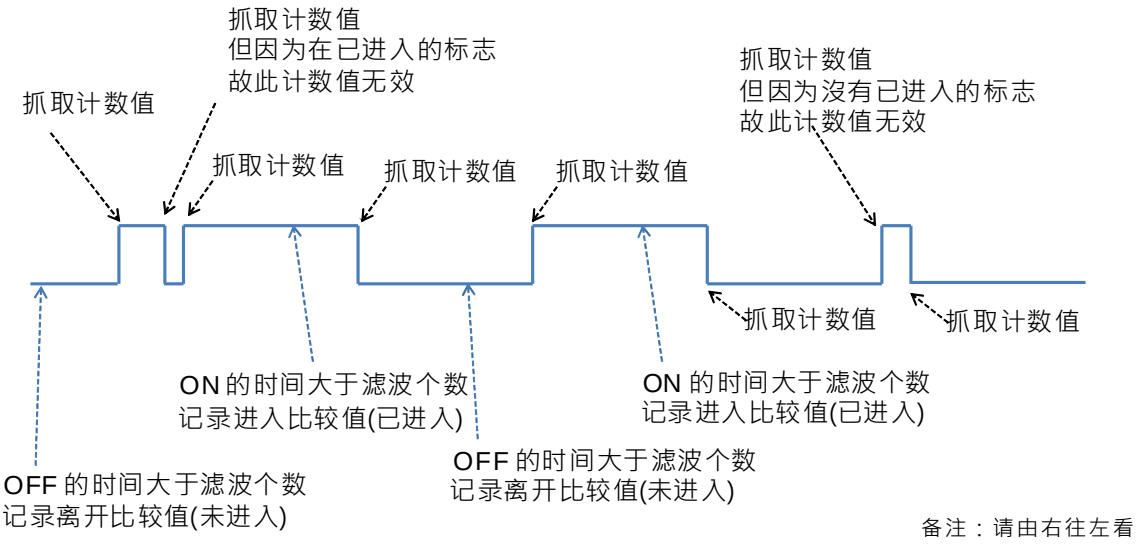
当基准值被设定为 0 时，表示该编号的工位不运作，用户可利用此基准值机动调整机台每次运作的工位。

9. **D** 为指定储存比较值堆栈区之起始地址，此储存堆栈区将连续占用 2n 个 16 位装置，以及 2mxn 个 32 位装置（或 4mxn 个 16 位装置），若使用空间超出 D 装置范围，则指令不执行，详细各个装置的功能名称与对应 **D** 编号如下表所示：

功能名称	工位 1	工位 2	・・・	工位 n
前进索引值（16 位）	**D**+0	**D**+1	・・・	**D**+（n-1）
追随索引值（16 位）	**D**+n	**D**+（n+1）	・・・	**D**+（2n-1）
对象进入比较值 1（32 位）	**D**+2n	**D**+2n+2	・・・	**D**+2n+2（n-1）
对象离开比较值 1（32 位）	**D**+4n	**D**+4n+2	・・・	**D**+4n+2（n-1）
：	：	：	：	：
对象进入比较值 m（32 位）	**D**+4mxn-2n		・・・	**D**+4mxn-2
对象离开比较值 m（32 位）	**D**+4mxn		・・・	**D**+4mxn+2(n-1)

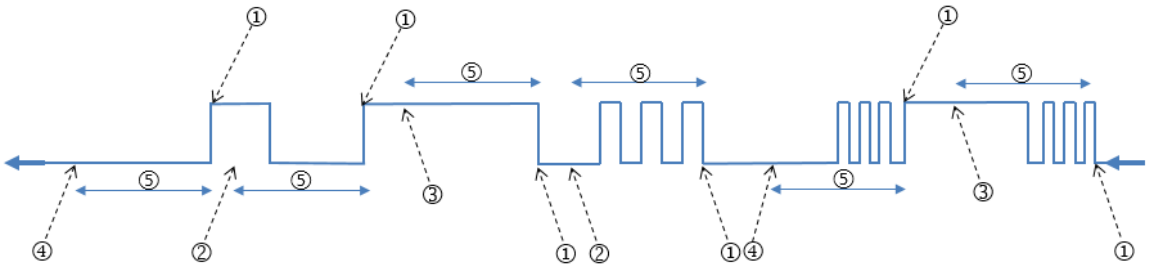
此堆栈区连续占用空间很大，使用前请注意 D 装置空间是否足够使用，若不足使用时，PLC 将只执行有效空间的储存，且不会有警报发生。（建议宣告 2n+4mn 个 word 类型的矩阵变量）

10. 此指令没有使用次数限制，但每次仅限一条指令被启动。
11. 此指令建议搭配 API 0710 YOUT 指令一起使用，并且使用相同的比较值堆栈区（即 **D** 操作数占用的储存空间）。
12. 在 V1.04.00 版（含）以前，抓取高速计数器与滤波功能的时序（从右往左看）说明，如下图所示：



备注：请由右往左看

13. 在 V1.06.00 版（含）以后，抓取高速计数器与滤波功能之时序（从右往左看）说明，如下图所示：



①. PLC 抓取计数值
②. 未达滤波计数个数，释放计数值
③. ON 状态已达滤波计数个数，记录至进入比较堆栈区
④. OFF 状态已达滤波计数个数，记录至离开比较堆栈区
⑤. 滤波计数个数

14. 当上升沿与下降沿触发抓取高速计数器，并且经过滤波个数之后，此时前进索引值加 1 后，分别记录到每个工位对应的进入与离开比较值，而此两个比较值 = 抓取计数值+基准值+偏差值。由于上升沿与下降沿触发都会累加前进索引值，因此前进索引值的最大值为最多对象数量 mx2。

15. 前进索引值在每次上升沿与下降沿触发（默认触发输入点为 OFF 状态），并且滤波个数达到后，将自动累加 1，而且采用循环式累加，例如：对象最多处理数量 m 设为 10，则此前进索引值（初始值为 0）将会以 1，2，3，...20，1，2，3，...20 循环。索引值为 0 时，表示指令启动后尚未有对象进入。备注：前进索引值加 1 之前会判断追随索引值，若加 1 后数值会等于追随索引值，则不累加前进索引值与记录此次比较值。
16. 当指令刚启动且触发输入点为 OFF 时，则后续上升沿/下降沿触发分别对应奇数/偶数的前进索引值；当指令刚启动且触发输入点为 ON 时，则后续下降沿/上升沿触发分别对应奇数/偶数的前进索引值。
17. 指令每次被启动时，其指定之比较堆栈区与索引值都不会做任何默认清除动作，故用户若指定的数据区为停电保持区，且需重新进行启动时，请自行搭配使用 ZRST 指令清除前进与追随索引值。

程序范例：请参考 API 0710 YOUT 指令范例说明。

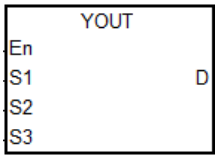
API	指令码	操作数	功能
0710	YOUT	**S_1 · S_2 · S_3 · D**	多任务位比较与输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1							○									
S_2								○								
S_3								○								
D		○	○													

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2		●				●							
S_3			●				●						
D	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 高速计数器编号

S_2 ： 工位与对象数量设定

S_3 ： 比较值的堆栈区起始地址

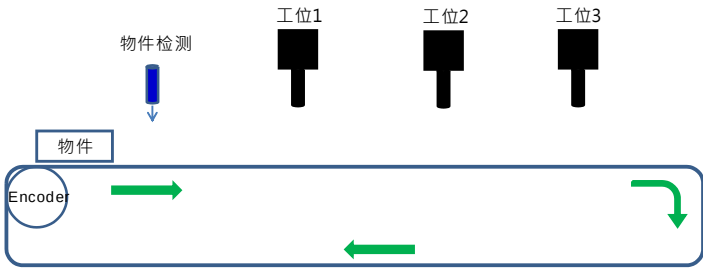
D ： 输出工位的起始地址

指令说明：

1. V1.04.00 版（含）以上支持此指令。此指令不建议使用 ST 结构化语言编辑、中断程序以及仅被调用一次性的功能块（FB）程序。
2. **S_1** 操作数为指定搭配的高速计数器，必须设置与 API0709 XCMP 设定的高速计数器相同。
3. **S_2** 操作数连续占用 2 个 16 位装置，**S_2**+0 为设定工位数量 n 个，**S_2**+1 为最多可处理对象数量 m 个。在 V1.04.00 版（含）以前上述 n，m 的数值范围为 1~32；在 V1.06.00 版（含）以后上述 n，m 的数值范围为 1~64 超过此范围时以最小或最大值执行，并且与 XCMP 指令设置相同。
4. **S_3** 为指定比较值堆栈区之起始地址，此装置连续占用 2n 个 16 位装置，以及 2mxn 个 32 位装置(或 4mxn 个 16 位装置），详细各个装置的功能名称与对应 **D** 编号请参考 XCMP 指令的比较值堆栈区说明。(建议与 XCMP 指令使用相同变量)
5. 此指令没有使用次数限制，但每次仅限一条指令被启动。
6. 此指令建议搭配 API 0709 XCMP 指令一起使用，并且使用相同的比较值堆栈区 (即 **S_3** 操作数连续占用空间 ）。
7. **D** 操作数仅限使用 Y 与 M 装置输出（或 BOOL 类型），并且连续占用工位数量 n 个位。已被选择当选择使用为主机内建之输出点或 M 装置时，则每次判断发生比较设定或清除动作时，指令将自动立即刷新输出状态。

8. 奇数前进索引值（如 1、3、5...）对应的比较值，以下称为前进比较值。偶数前进索引值（如 2、4、6...）对应之比较值，以下称为离开比较值。
9. 当比较堆栈区内的进入与离开的比较值都为 0 时，表示此区域比较动作不执行，对应之工位输出点为 OFF 状态，且追随索引值加 2，但不会超过前进索引值。
10. YOUT 指令执行时，每个工位将依照追随索引值比较相对应的进入与离开比较值；当判断计数值 >= 进入比较值时，则对应的输出点设为 ON，且追随索引值将自动加 1；当判断计数值 >= 离开比较值时，则对应之输出点清除为 OFF，且追随索引值将自动加 1，但不会超过前进索引值。

程序范例：假设多任务位输出架构图如下所示，工位数量 3 个，最多物件数量为 4 个。

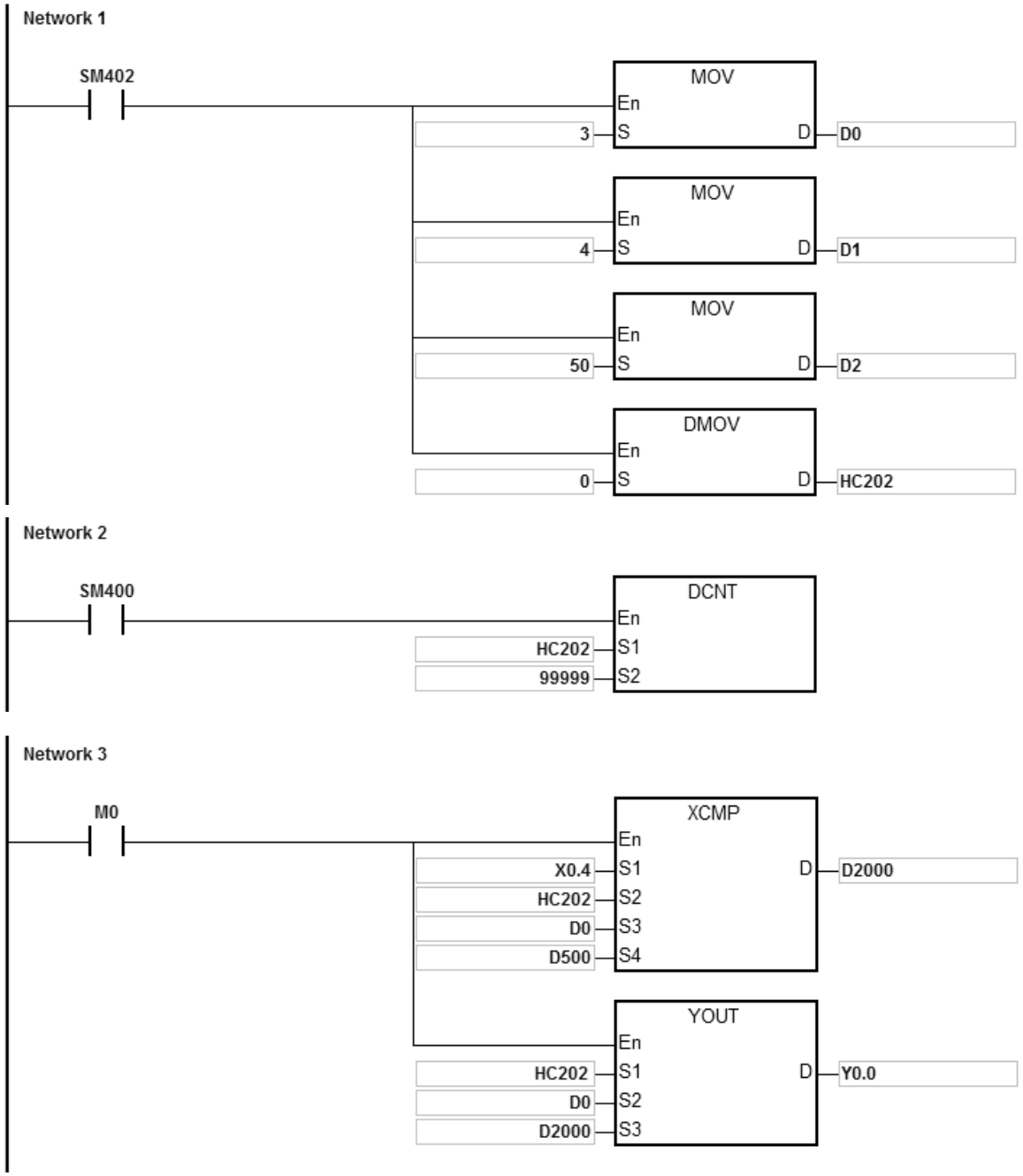


步骤 1：采用 X0.4 输入点为对象检测中断，HC202 为编码器对应的高速计数器，Y0.0 输出为输出工位的起始输出点。

步骤 2：利用寄存器编辑功能，设定基准值、进入偏差值与离开偏差值，如下表所示。

D 编号	D500	D502	D504	
设定 32 位数值	K2000	K3000	K4000	基准值
D 编号	D506	D508	D510	
设定 32 位数值	K100	K120	K130	进入偏差值
D 编号	D512	D514	D516	
设定 32 位数值	K50	K-20	K20	离开偏差值
D 编号	D2000	D2001	D2002	
设定 16 位数值	K0	K0	K0	前进索引值
D 编号	D2003	D2004	D2005	
设定 16 位数值	K0	K0	K0	追随索引值

步骤 3：设定初始值与编写程序如下图所示。



D0 设定 3 组工位，D1 设定 4 组对象，D2 设定滤波个数 50。启动 M0 接点之后，开始执行对象输入检测、设定比较值、各个工位进入/离开比较值判断，以及各个工位输出控制，假设检测了 2 个对象进入共有 4 次触发抓取的 HC202 计数值，分别为 3000、3500、4500、5000，则索引值与堆栈区记录数值如下表所示：(此时 HC202=K5060)

D 编号	D2000	D2001	D2002	
16 位数值	K4	K4	K4	前进索引值
D 编号	D2003	D2004	D2005	
16 位数值	K1	K1	K1	追随索引值

D 编号	D2006	D2008	D2010	
32 位比较值	K5100	K6120	K7130	进入比较值 1
D 编号	D2012	D2014	D2016	
32 位比较值	K5550	K6480	K7520	离开比较值 1
D 编号	D2018	D2020	D2022	
32 位比较值	K6600	K7620	K8630	进入比较值 2
D 编号	D2024	D2026	D2028	
32 位比较值	K7050	K7980	K9020	离开比较值 2
D 编号	D2030	D2032	D2034	
32 位比较值	K0	K0	K0	进入比较值 3
D 编号	D2036	D2038	D2040	
32 位比较值	K0	K0	K0	离开比较值 3

当 HC202 计数数值到达 5200 时，则对应 Y 输出状况如下表所示：

Y 编号	Y0.0	Y0.1	Y0.2	
输出状态	ON	OFF	OFF	
D 编号	D2000	D2001	D2002	
16 位数值	K4	K4	K4	前进索引值
D 编号	D2003	D2004	D2005	
16 位数值	K2	K1	K1	追随索引值

当 HC202 计数数值到达 6200 时，则对应 Y 输出状况以及前进/追随索引值数值如下表所示：

Y 编号	Y0.0	Y0.1	Y0.2	
输出状态	OFF	ON	OFF	
D 编号	D2000	D2001	D2002	
16 位数值	K4	K4	K4	前进索引值
D 编号	D2003	D2004	D2005	
16 位数值	K3	K2	K1	追随索引值

当 HC202 计数数值到达 6800 时，则对应 Y 输出状况以及前进/追随索引值数值如下表所示：

Y 编号	Y0.0	Y0.1	Y0.2	
输出状态	ON	OFF	OFF	
D 编号	D2000	D2001	D2002	
16 位数值	K4	K4	K4	前进索引值
D 编号	D2003	D2004	D2005	
16 位数值	K4	K3	K1	追随索引值

当 HC202 计数数值到达 7300 时，则对应 Y 输出状况以及前进/追随索引值数值如下表所示：

Y 编号	Y0.0	Y0.1	Y0.2
输出状态	OFF	OFF	ON

D 编号	D2000	D2001	D2002	
16 位数值	K4	K4	K4	前进索引值
D 编号	D2003	D2004	D2005	
16 位数值	K4	K3	K2	追随索引值

当 HC202 计数数值到达 7700 时，则对应 Y 输出状况以及前进/追随索引值数值如下表所示：

Y 编号	Y0.0	Y0.1	Y0.2	
输出状态	OFF	ON	OFF	
D 编号	D2000	D2001	D2002	
16 位数值	K4	K4	K4	前进索引值
D 编号	D2003	D2004	D2005	
16 位数值	K4	K4	K3	追随索引值

当 HC202 计数数值到达 8000 时，则对应 Y 输出状况以及前进/追随索引值数值如下表所示：

Y 编号	Y0.0	Y0.1	Y0.2	
输出状态	OFF	OFF	OFF	
D 编号	D2000	D2001	D2002	
16 位数值	K4	K4	K4	前进索引值
D 编号	D2003	D2004	D2005	
16 位数值	K4	K4	K3	追随索引值

当 HC202 计数数值到达 8700 时，则对应 Y 输出状况以及前进/追随索引值数值如下表所示：

Y 编号	Y0.0	Y0.1	Y0.2	
输出状态	OFF	OFF	ON	
D 编号	D2000	D2001	D2002	
16 位数值	K4	K4	K4	前进索引值
D 编号	D2003	D2004	D2005	
16 位数值	K4	K4	K4	追随索引值

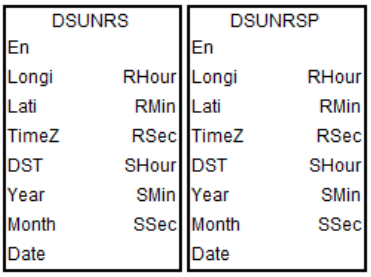
API	指令码			操作数	功能
0711	D	SUNRS	P	**Longi ~ SSec**	日出与日落时间

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Longi								●								○
Lati								●								○
TimeZ								●					○			
DST	●	●	●	●												
Year								●					○			
Month								●					○			
Date								●					○			
RHour								●								
RMin								●								
RSec								●								
SHour								●								
SMin								●								
SSec								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Longi									●				
Lati									●				
TimeZ		●				●							
DST	●												
Year		●				●							
Month		●				●							
Date		●				●							
RHour		●				●							
RMin		●				●							
RSec		●				●							
SHour		●				●							
SMin		●				●							
SSec		●				●							

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



Longi ： 当地经度(REAL 类型数值)

Lati ： 当地纬度(REAL 类型数值)

TimeZ ： 当地时区(整数值)(-12 ~ +14)

DST ： 判断日光节约时间

Year 指定日期的公元年

Month 指定日期的月份

Date 指定日期

RHour 指定日期的日出“时”(24 小时制)

RMin 指定日期的日出“分”

RSec 指定日期的日出“秒”

SHour 指定日期的日落“时”(24 小时制)

SMin 指定日期的日落“分”

SSec 指定日期的日落“秒”

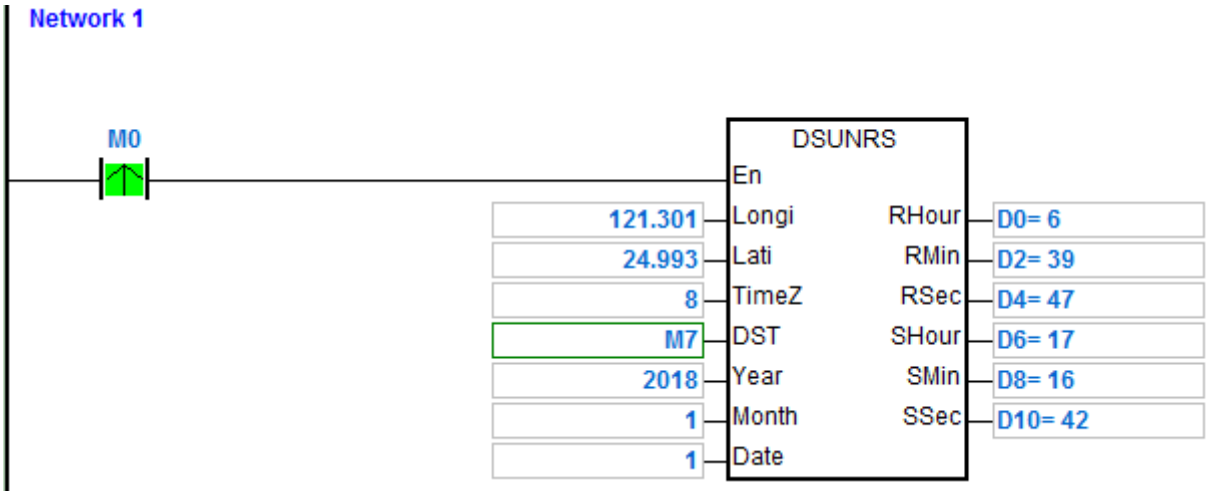
指令说明：

1. V1.04.50 版（含）以上支持此指令。此指令所计算之日出与日落时间，难免会因为输入之当地经/纬度的精度或海拔高度等因素所影响而有所误差，因此若与当地公布之标准时间不符时，请自行再次微调时间。此指令目前自我验证之准确度，其误差范围大约可小于 5 分钟以内。
2. 经度(**Longi**)与纬度(**Lati**)请以符点数值输入，举例：地点为”台湾桃园市”，则其经度为 121.30098，纬度为 24.99363。其中经度为正数时，表示东经度；纬度为正数时，表示北纬度。
3. 时区(**TimeZ**)请依当地时区数值输入，其数值范围为 -12 ~ +14 之间。本指令不会自动依照经度去换算，故输入错误数值或超出范围，本指令将不会显示任何错误。
4. 当日光节约时间(**DST**)设为 On 时，表示指令需判断 PLC 本身的日光节约时间是否已启动，若有启动中，则自动将日出与日落时间补上 1 小时。
5. 当地年(**Year**)、月(**Month**)、日(**Date**)请以 10 进制整数值输入，并且输入正确数值，本指令不会检查数值是否正确。
6. 计算完成后，指令将输出日出之时(**RHour**)、分(**RMin**)、秒(**RSec**)与日落之时(**SHour**)、分(**SMin**)、秒(**SSec**)，其显示数值为 10 进制正整数值，并且为 24 小时制。

程序范例：计算台湾桃园市于 2018 年 1 月 1 日之日出与日落时间，时区为 UTC+8，不需判断 DST

PLC 计算出日出时间：6：39：47， 日落时间 17：16：42，范例程序如下图标。

自行比对官方网站，查询结果为日出时间：06：39：44， 日落时间 17：16：45，误差为+3 ~ -3 秒。



6.9 逻辑操作指令

6.9.1 逻辑操作指令一览表

API	指令码（位）		P 指令	功能
	16	32		
0800	WAND	DAND	✓	逻辑与（AND）运算
0801	MAND	–	✓	矩阵与（AND）运算
0802	WOR	DOR	✓	逻辑或（OR）运算
0803	MOR	–	✓	矩阵或（OR）运算
0804	WXOR	DXOR	✓	逻辑异或（XOR）运算
0805	MXOR	–	✓	矩阵异或（XOR）运算
0808	WINV	DINV	✓	逻辑反向或（INV）运算
0809	LD&	DLD&	–	S_1&S_2
0810	LD|	DLD|	–	S_1|S_2
0811	LD^	DLD^	–	S_1^S_2
0812	AND&	DAND&	–	S_1&S_2
0813	AND|	DAND|	–	S_1|S_2
0814	AND^	DAND^	–	S_1^S_2
0815	OR&	DOR&	–	S_1&S_2
0816	OR|	DOR|	–	S_1|S_2
0817	OR^	DOR^	–	S_1^S_2

6.9.2 逻辑操作指令说明

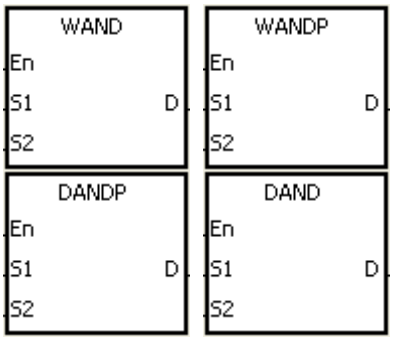
API	指令码			操作数	功能
0800	D	WAND	P	**S_1 · S_2 · D**	逻辑与（AND）运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 数据源装置 1

S_2 ： 数据源装置 2

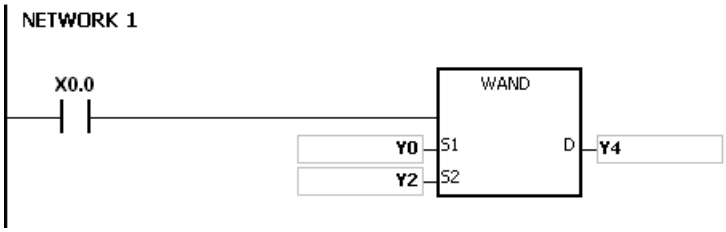
D ： 运算结果

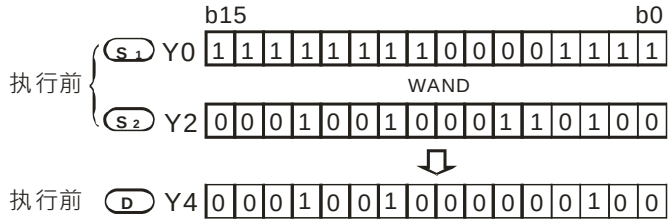
指令说明：

1. 两个数据源：**S_1** 及 **S_2** 作逻辑的"与"（AND）运算并将结果存于 **D**。
2. DAND 才可使用 HC 装置，但不可使用 E 装置。
3. 逻辑的"与"（AND）运算之规则为两位均为 1 结果始为 1，否则为 0。

程序范例一：

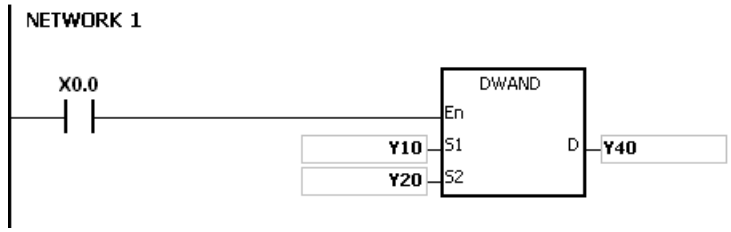
当 X0.0=ON 时，16 位 Y0 与 Y2 作 WAND，逻辑与（AND）运算，将结果存于 Y4 中。

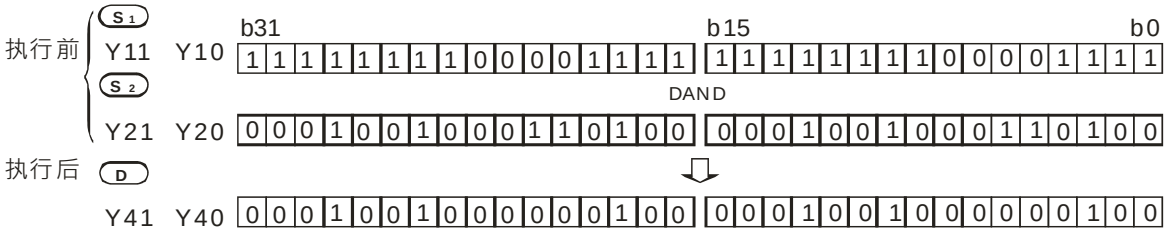




程序范例二：

当 X0.0=ON 时，32 位（Y11、Y10）与（Y21、Y20）作 DAND，逻辑与（AND）运算，将结果存于（Y41、Y40）中。





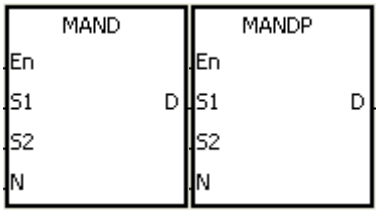
API	指令码			操作数	功能
0801		MAND	P	**S_1 · S_2 · D · n**	矩阵与（AND）运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●							
S_2	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 矩阵来源装置 1

S_2 ： 矩阵来源装置 2

D ： 运算结果

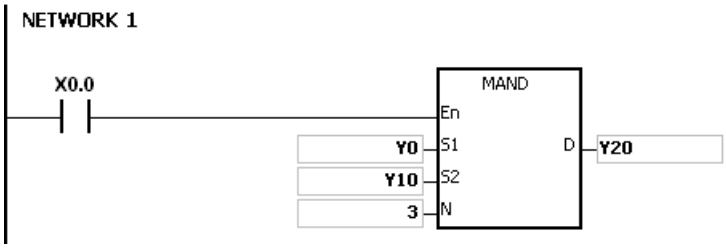
n ： 数组长度

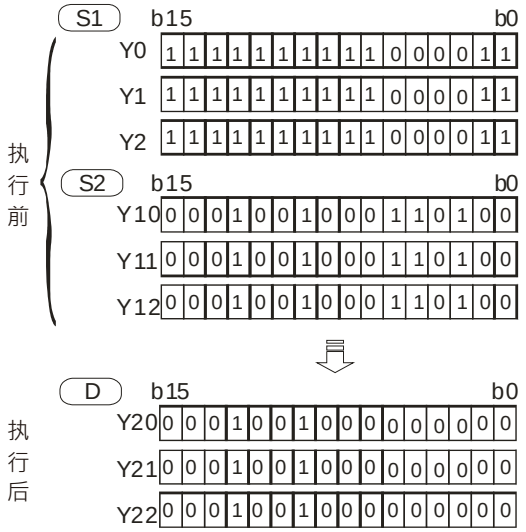
指令说明：

1. 两个矩阵来源依数组长度 **n** 将 **S_1** 及 **S_2** 作矩阵的"与"（AND）运算并将结果存于 **D**。
2. 矩阵的"与"（AND）运算之规则为两位均为 1 结果始为 1，否则为 0。
3. n 操作数的范围值为 1~256。

程序范例：

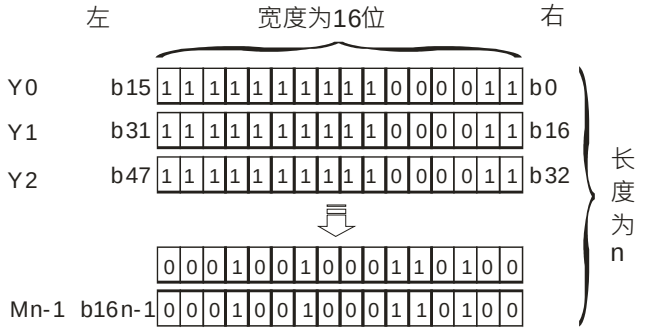
当 X0.0=ON 时，16 位寄存器 Y0~Y2 共 3 列与 16 位寄存器 Y10~Y12 共 3 列作 MAND，矩阵与 (AND) 运算，将结果存于 16 位寄存器 Y20~Y22 共 3 列中。





补充说明：

1. **S_1+n-1**、**S_2+n-1**、**D+n-1** 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 矩阵指令说明：
   - 矩阵是 1 个以上连续之 16 位寄存器所组成，组成矩阵之寄存器个数称为矩阵之长度 n，一个矩阵共有 16×n 个位（点），其运算单位一次只有一个位（点）。
   - 矩阵指令是将 16×n 个矩阵位（序号由 b0～b16n–1）当作一连串单点之集合，而自此集合中指定某一单点作运作，而不将之当作数值看待。
   - 矩阵指令主要在处理单点对多点（矩阵）或多点对多点之状态处理，如搬移、拷贝、比较、搜寻等，为极为方便和重要之应用指令。
   - 在矩阵指令运作中 ,通常需要有一个 16 位寄存器来指定矩阵中 16n 个单点之某个单点当作运算对象，此寄存器称为矩阵之指针 Pr（Pointer），由用户于指令中指定，其有效范围为 0～16n–1，分别对应至矩阵中之位 b0～b16n–1。
   - 矩阵运作中有左、右位移或旋转，我们定义高编号者为左，低编号者为右，如下图示。



   - 矩阵宽度（C）固定为 16 位（bits）。

- Pr：为矩阵之指针，例如 Pr 值为 15 指到 b15 之位。

  范例：以 Y0，n=3 构成的矩阵，Y0=16#AAAA，Y1=16#5555，Y2=16#AAFF

b_{15}	b_{14}	b_{13}	b_{12}	b_{11}	b_{10}	b_9	b_8	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0	
1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	Y0
0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	Y1
1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	Y2

  范例：以 X0，n=3 构成的矩阵，X0=16#37，X1=16#68，X2=16#45

b_{15}	b_{14}	b_{13}	b_{12}	b_{11}	b_{10}	b_9	b_8	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0	
0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	X0
0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	X1
0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	X2

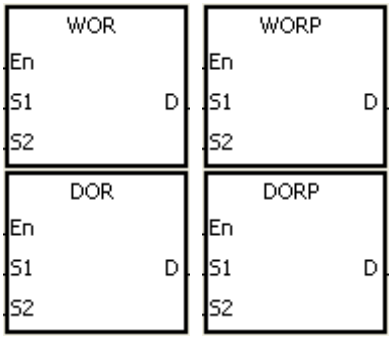
API	指令码			操作数	功能
0802	D	WOR	P	**S_1 · S_2 · D**	逻辑或（OR）运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 数据源装置 1

S_2 ： 数据源装置 2

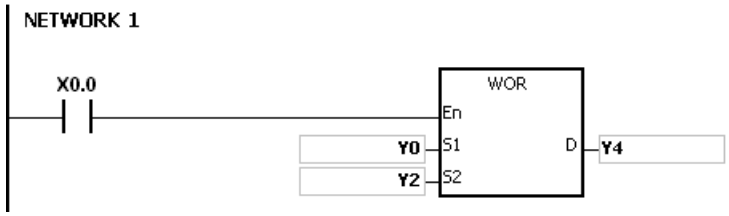
D ： 运算结果

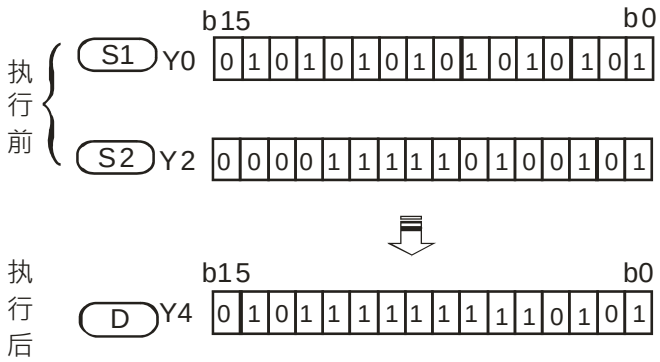
指令说明：

1. 两个数据源：**S_1** 及 **S_2** 作逻辑的“或”（OR）运算结果存于 **D**。
2. DOR 才可使用 HC 装置，但不可使用 E 装置。
3. 逻辑的“或”（OR）运算之规则两位有任一为 1 则结果为 1，两者均为 0 结果才为 0。

程序范例一：

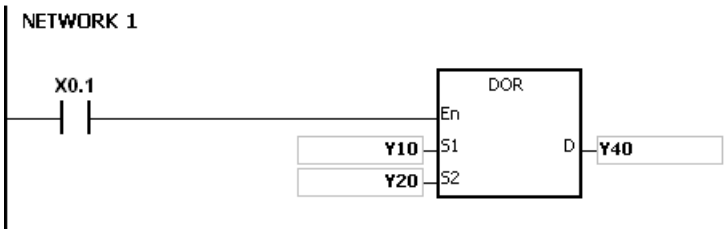
当 X0.0=ON 时，16 位 Y0 与 Y2 作 WOR，逻辑或（OR）运算，将结果存于 Y4 中。

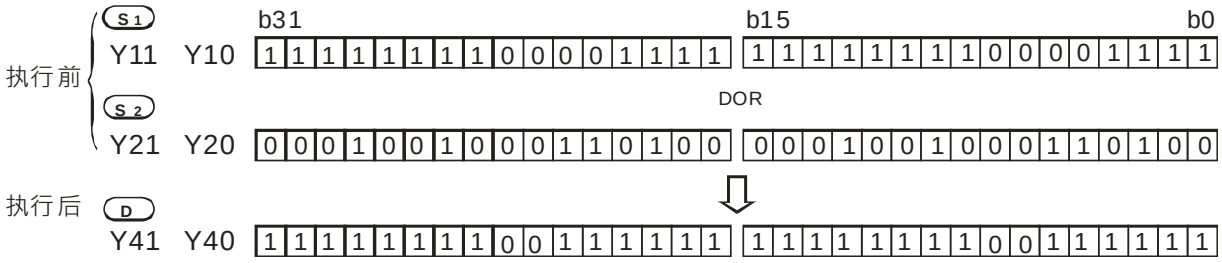




程序范例二：

当 X0.1=ON 时，32 位（Y11、Y10）与（Y21、Y20）作 DOR，逻辑或（OR）运算，将结果存于（Y41、Y40）中。





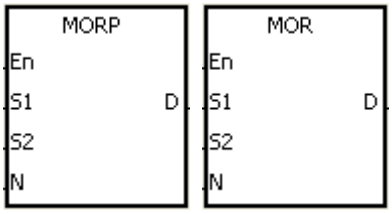
API	指令码			操作数	功能
0803		MOR	P	**S_1 · S_2 · D · n**	矩阵或（OR）运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●							
S_2	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 矩阵来源装置 1

S_2 ： 矩阵来源装置 2

D ： 运算结果

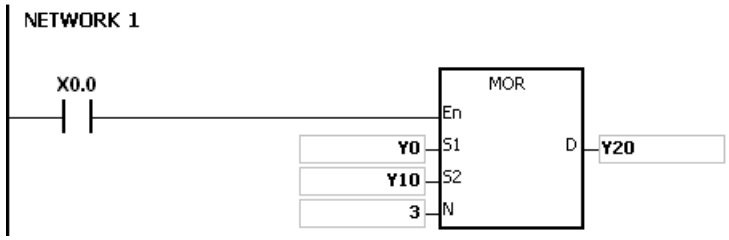
n ： 数组长度

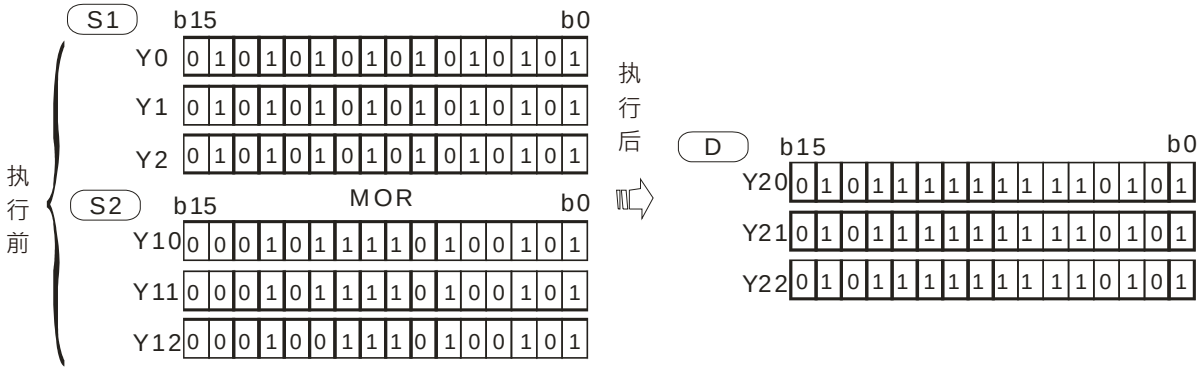
指令说明：

1. 两个矩阵来源依数组长度 **n** 将 **S_1** 及 **S_2** 作矩阵的"或"（OR）运算并将结果存于 **D**。
2. 矩阵的"或"（OR）运算之规则为两位有任一为 1 则结果为 1，两者均为 0 结果才为 0。
3. **n** 操作数的范围值为 1~256。

程序范例：

当 X0.0=ON 时，16 位寄存器 Y0~Y2 共 3 列与 16 位寄存器 Y10~Y12 共 3 列作 MOR，矩阵或（OR）运算，将结果存于 16 位寄存器 Y20~Y22 共 3 列中。





补充说明：

1. **S_1+n-1**、**S_2+n-1**、**D+n-1** 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

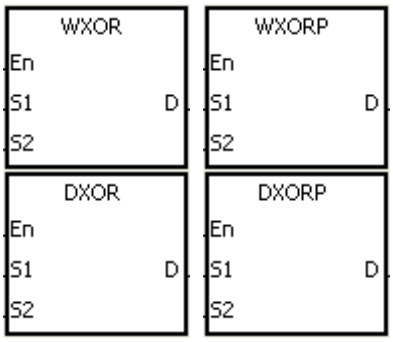
API	指令码			操作数	功能
0804	D	WXOR	P	**S_1 · S_2 · D**	逻辑异或（XOR）运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 数据源装置 1

S_2 ： 数据源装置 2

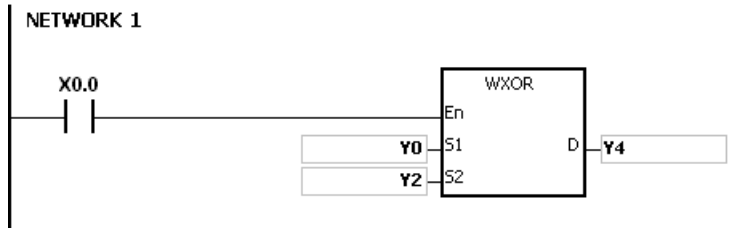
D ： 运算结果

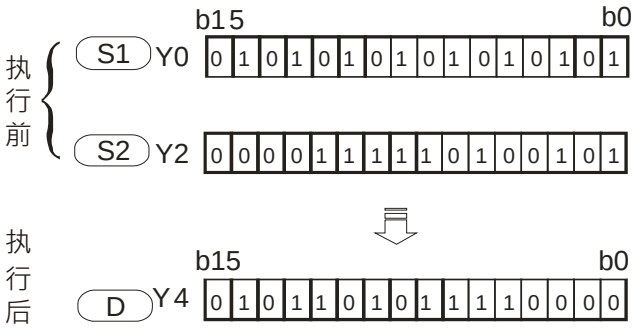
指令说明：

1. 两个数据源：**S_1** 及 **S_2** 作逻辑的“异或”（XOR）运算结果存于 **D**。
2. DXOR 才可使用 HC 装置，但不可使用 E 装置。
3. 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。

程序范例一：

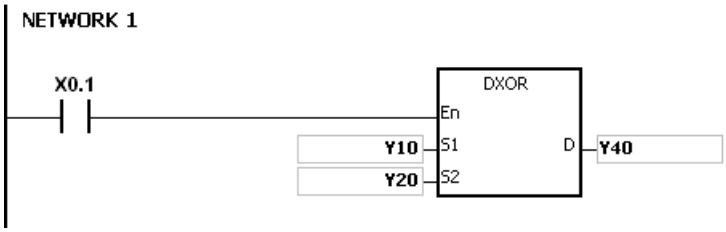
当 X0.0=ON 时，16 位 Y0 与 Y2 作 WXOR，逻辑异或（XOR）运算，将结果存于 Y4 中。

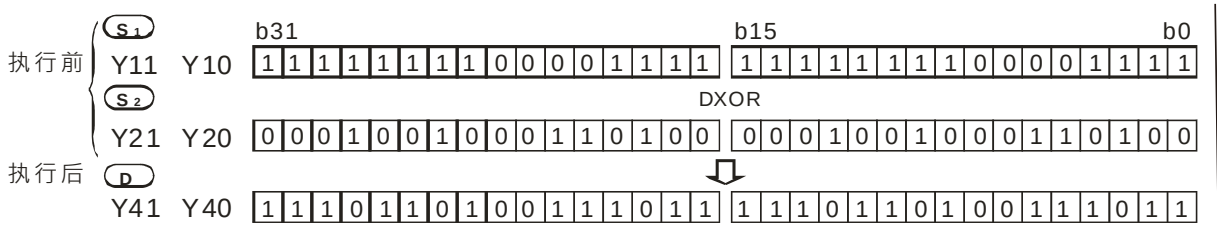




程序范例二：

当 X0.1=ON 时，32 位（Y11、Y10）与（Y21、Y20）作 DXOR，逻辑异或（XOR）运算，将结果存于（Y41、Y40）中。





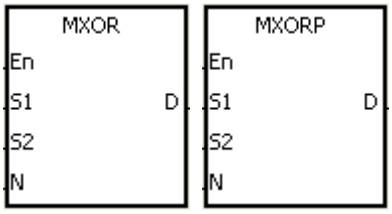
API	指令码			操作数	功能
0805		MXOR	P	**S_1 · S_2 · D · n**	矩阵异或（XOR）运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●							
S_2	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 矩阵来源装置 1

S_2 ： 矩阵来源装置 2

D ： 运算结果

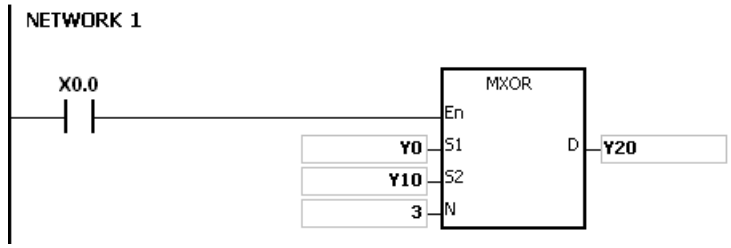
n ： 数组长度

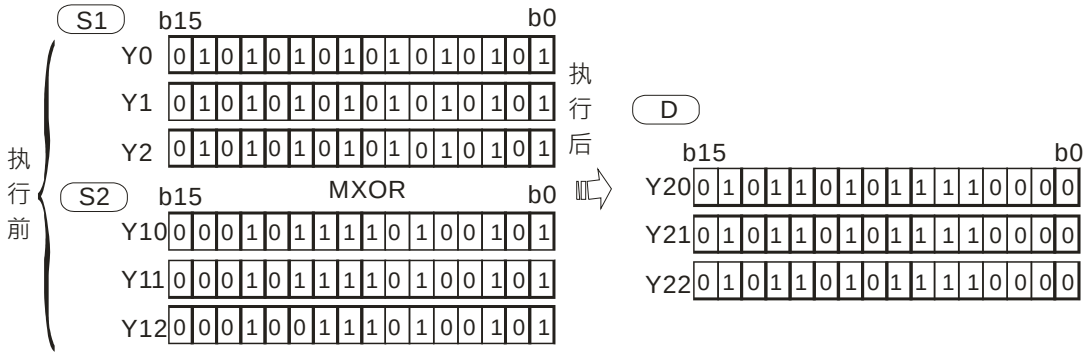
指令说明：

1. 两个矩阵来源依数组长度 **n** 将 **S_1** 及 **S_2** 作矩阵的"异或"（XOR）运算并将结果存于 **D**。
2. 矩阵的"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。
3. **n** 操作数的范围值为 1~256。

程序范例：

当 X0.0=ON 时，16 位寄存器 Y0~Y2 共 3 列与 16 位寄存器 Y10~Y12 共 3 列作 MXOR，矩阵异或（XOR）运算，将结果存于 16 位寄存器 Y20~Y22 共 3 列中。





补充说明：

1. **S_1+n-1**、**S_2+n-1**、**D+n-1** 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

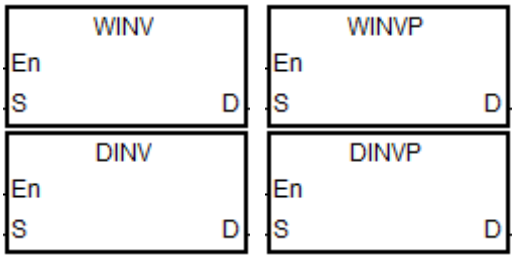
API	指令码			操作数	功能
0808	D	WINV	P	**S**， **D**	反向或（INV）运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D	●	●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 数据源装置

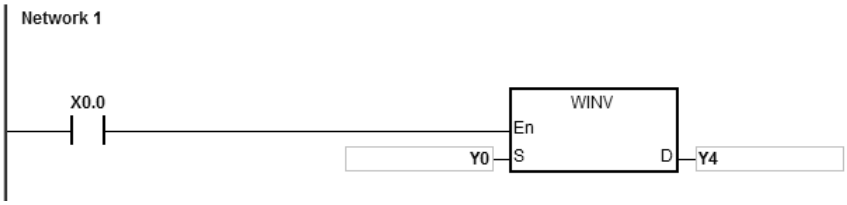
D ： 运算结果

指令说明：

1. 数据源：**S** 作逻辑的“反向或”（INV）运算结果存于 **D**。
2. DINV 才可使用 HC 装置，但不可使用 E 装置。
3. 逻辑的“反向或”（INV）运算之规则为 0 改为 1，1 改为 0。

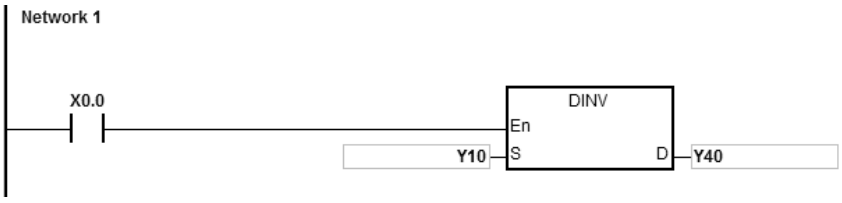
程序范例一：

当 X0.0=ON 时，16 位 Y0 作 WINV，逻辑反向或（INV）运算，将结果存于 Y4 中。



程序范例二：

当 X0.0=ON 时，32 位（Y11、Y10）作 DINV，逻辑反向或（INV）运算，将结果存于（Y41、Y40）中。



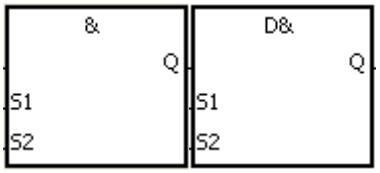
API	指令码			操作数	功能
0809~0811	D	LD＃		S_1，S_2	接点类型逻辑运算 LD＃

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 LD&跟 DLD&为例

S_1 ： 数据源装置 1

S_2 ： 数据源装置 2

指令说明：

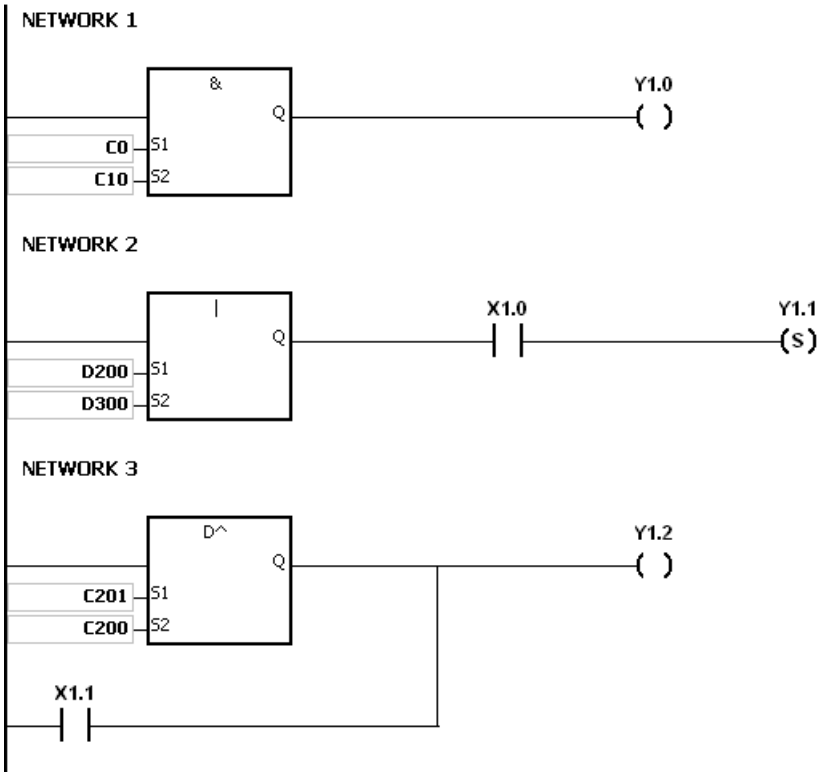
1. S_1 与 S_2 之内容作比较的指令，比较结果不为 0 时，该指令导通，比较结果为 0 时，该指令不导通。
2. DLD＃才可使用 HC 装置，但不可使用 E 装置。
3. LD＃的指令可直接与母线连接使用。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
0809	LD&	**D**LD&	S_1&S_2≠0	S_1&S_2＝0
0810	LD|	**D**LD|	S_1|S_2≠0	S_1|S_2＝0
0811	LD^	**D**LD^	S_1^S_2≠0	S_1^S_2＝0

4. &：逻辑的“与”（AND）运算。
5. |：逻辑的“或”（OR）运算。
6. ^：逻辑的“异或”（XOR）运算。

程序范例：

1. C0 与 C10 的内容做逻辑的“与”（AND）运算不等于 0 时，Y1.0=ON。
2. D200 与 D300 的内容做逻辑的“或”（OR）运算不等于 0 时，而且 X1.0=ON 的时候，Y1.1=ON 并保持住。
3. C201 与 C200 的内容做逻辑的“异或”（XOR）运算不等于 0 时或是 X1.1=ON 的时候，Y1.2=ON。



补充说明：

S_1、**S_2** 装置不合法，接点不导通，SM0=ON，错误码 SR0=16#2003。

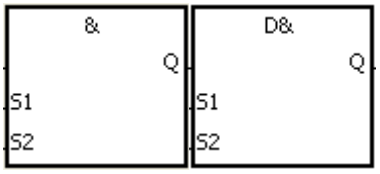
API	指令码			操作数	功能
0812~0814	D	AND#		S_1 · S_2	接点类型逻辑运算 AND#

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 AND&跟 DAND&为例

S_1 ： 数据源装置 1

S_2 ： 数据源装置 2

指令说明：

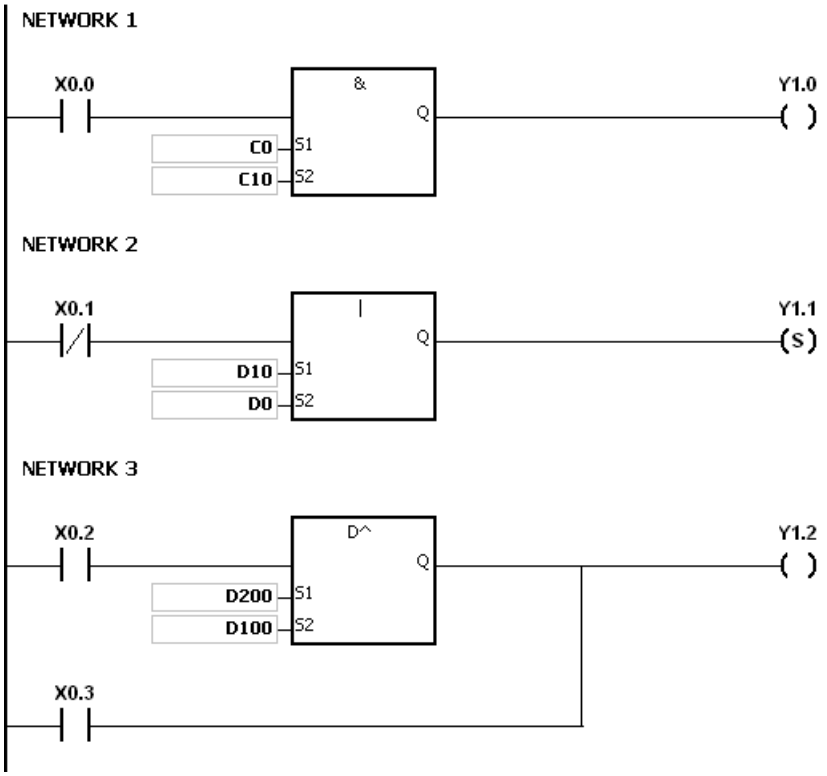
1. S_1 与 S_2 之内容作比较的指令，比较结果不为 0 时，该指令导通，比较结果为 0 时，该指令不导通。
2. DAND# 才可使用 HC 装置，但不可使用 E 装置。
3. AND# 的指令是与接点串接的运算指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
0812	AND&	**D**AND&	S_1&S_2≠0	S_1&S_2=0
0813	AND|	**D**AND|	S_1|S≠0	S_1|S_2=0
0814	AND^	**D**AND^	S_1^S_2≠0	S_1^S=0

4. &：逻辑的"与"（AND）运算。
5. |：逻辑的"或"（OR）运算。
6. ^：逻辑的"异或"（XOR）运算。

程序范例：

1. 当 X0.0=ON 时且 C0 与 C10 的内容做逻辑的"与"（AND）运算不等于 0 时，Y1.0=ON。
2. 当 X0.1=OFF 时且 D10 与 D0 的内容做逻辑的"或"（OR）运算不等于 0 时，Y1.1=ON 并保持住。
3. 当 X0.2=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 X0.3=ON 的时候，Y1.2=ON。



补充说明：

S_1、**S_2** 装置不合法，接点不导通，SM0=ON，错误码 SR0=16#2003。

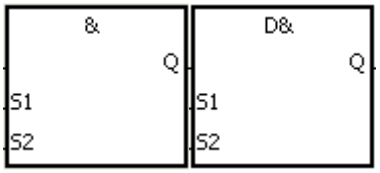
API	指令码			操作数	功能
0815~0817	D	OR#		S_1，S_2	接点类型逻辑运算 OR#

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



以 OR&跟 DOR&为例

S_1 ： 数据源装置 1

S_2 ： 数据源装置 2

指令说明：

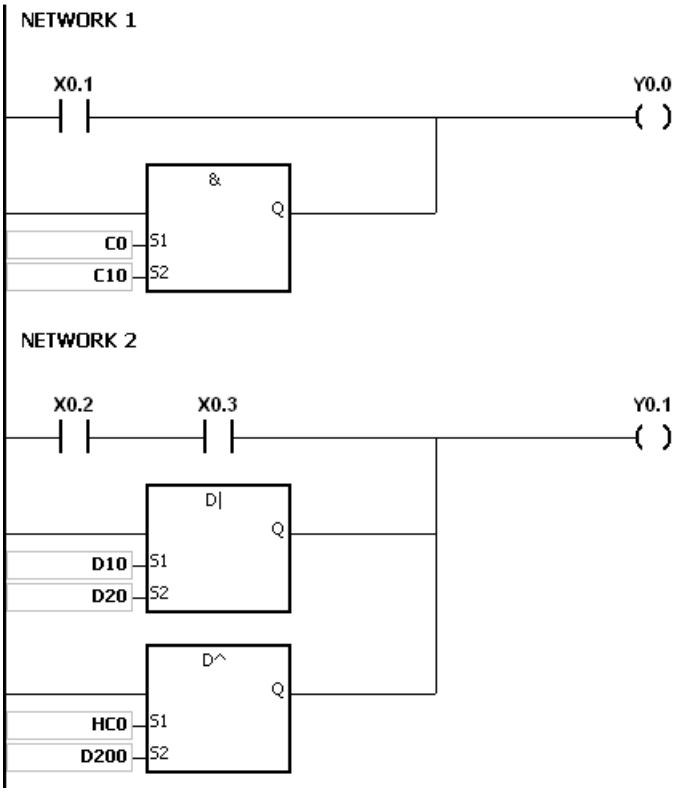
1. S_1 与 S_2 之内容作比较的指令，比较结果不为 0 时，该指令导通，比较结果为 0 时，该指令不导通。
2. DOR# 才可使用 HC 装置，但不可使用 E 装置。
3. OR# 的指令是与接点并接的运算指令。

API No.	16-bit 指令	32-bit 指令	导通条件	非导通条件
0815	OR&	**D**OR&	S_1&S_2 ≠ 0	S_1&S_2 = 0
0816	OR|	**D**OR|	S_1|S_2 ≠ 0	S_1|S_2 = 0
0817	OR^	**D**OR^	S_1^S_2 ≠ 0	S_1^S_2 = 0

4. &：逻辑的"与"（AND）运算。
5. |：逻辑的"或"（OR）运算。
6. ^：逻辑的"异或"（XOR）运算。

程序范例：

1. 当 X0.1=ON 时或 C0 与 C10 的内容做逻辑的"与"（AND）运算不等于 0 时，Y0.0=ON。
2. 当 X0.2 及 X0.3 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 HC0 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，Y0.1=ON。



补充说明：

S_1、**S_2** 装置不合法，接点不导通，SM0=ON，错误码 SR0=16#2003。

6.10 循环指令

6.10.1 循环指令一览表

API	指令码（位）		P 指令	功能
	16	32		
0900	ROR	DROR	✓	右循环
0901	RCR	DRCR	✓	附进位标志右循环
0902	ROL	DROL	✓	左循环
0903	RCL	DRCL	✓	附进位标志左循环
0904	MBR	–	✓	矩阵位循环

6.10.2 循环指令说明

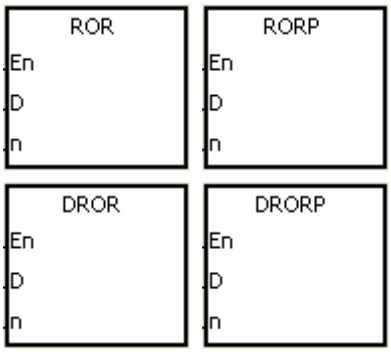
API	指令码			操作数	功能
0900	D	ROR	P	**D**，**n**	右循环

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				
n	●	●			●	●	●	●	●		○	○	○	○		

数据型态	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ： 欲循环之装置

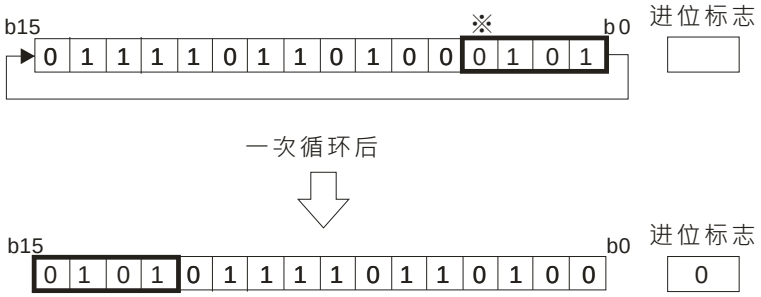
n ： 一次循环之位数

指令说明：

1. 将 **D** 所指定的装置内容一次向右循环 **n** 个位。
2. **D**ROR 才可以使用 HC 装置，但不可使用 E 装置。
3. 16 位指令 **n**=1~16，32 位指令 n=1~32，当 **n**<=0 时不执行，超过范围时，以最大值执行。
4. 本指令一般都是使用脉冲执行型指令（RORP、DRORP）。

程序范例：

当 X0.0 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※的位内容被传送至进位标志信号 SM602 内。



补充说明：

1. 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

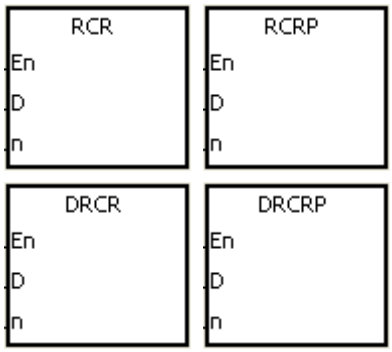
API	指令码			操作数	功能
0901	D	RCR	P	**D**，**n**	附进位标志右循环

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				
n	●	●			●	●	●	●	●		○	○	○	○		

数据型态	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ： 欲循环之装置

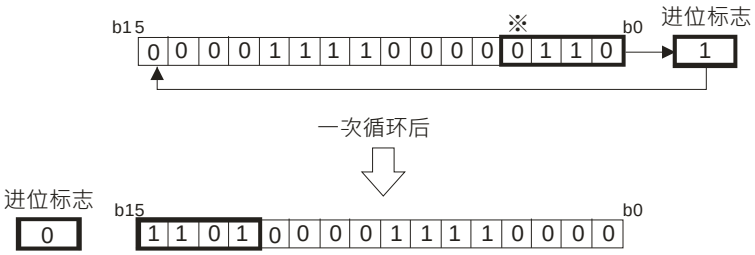
n ： 一次循环之位数

指令说明：

1. 将 **D** 所指定的装置内容连同进位标志 SM602，一次向右循环 **n** 个位。
2. **D**RCR 才可以使用 HC 装置，但不可使用 E 装置。
3. 16 位指令 **n**=1~16，32 位指令 n=1~32，当 **n**<=0 时不执行，超过范围时，以最大值执行。
4. 本指令一般都是使用脉冲执行型指令（RCRP、DRCRP）。

程序范例：

当 X0.0 从 OFF→ON 变化时，D10 的 16 个位连同进位标志 SM602 共 17 个位以 4 个位为一组往右循环，如下图所示标明※的位内容被传送至进位标志信号 SM602 内。



补充说明：

1. 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

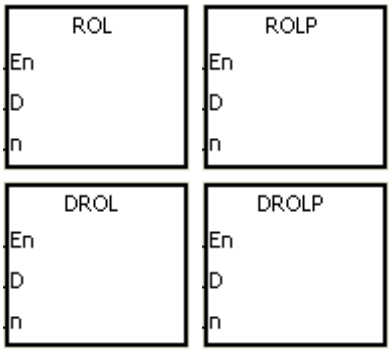
API	指令码			操作数	功能
0902	D	ROL	P	**D**，**n**	左循环

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				
n	●	●			●	●	●	●	●		○	○	○	○		

数据型态	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ： 欲循环之装置

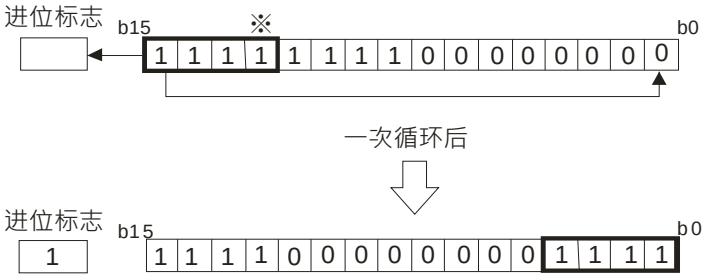
n ： 一次循环之位数

指令说明：

1. 将 **D** 所指定的装置内容一次向左循环 **n** 个位。
2. **D**ROL 才可以使用 HC 装置，但不可使用 E 装置。
3. 16 位指令 **n**=1~16，32 位指令 n=1~32，当 **n**<=0 时不执行，超过范围时，以最大值执行。
4. 本指令一般都是使用脉冲执行型指令（ROLP、DROLP）。

程序范例：

当 X0.0 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 SM602 内。



补充说明：

1. 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

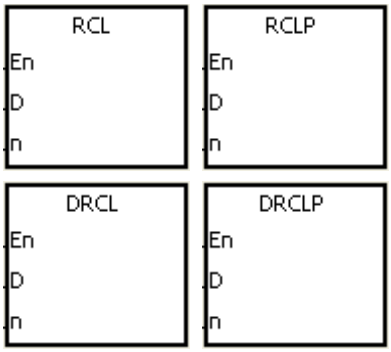
API	指令码			操作数	功能
0903	D	RCL	P	**D**，**n**	附进位标志左循环

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				
n	●	●			●	●	●	●	●		○	○	○	○		

数据型态	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ： 欲循环之装置

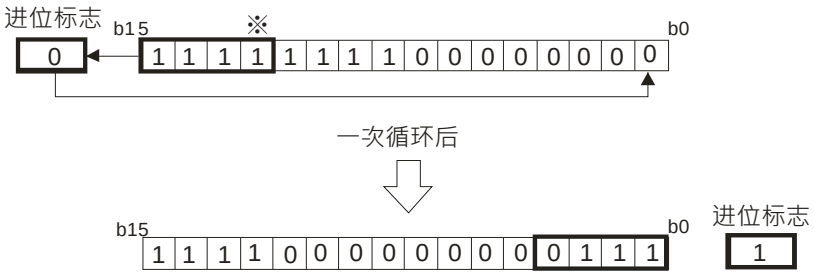
n ： 一次循环之位数

指令说明：

1. 将 **D** 所指定的装置内容连同进位标志 SM602，一次向左循环 **n** 个位。
2. **D**RCL 才可以使用 HC 装置，但不可使用 E 装置。
3. 16 位指令 **n**=1~16，32 位指令 n=1~32，当 **n**<=0 时不执行，超过范围时，以最大值执行。
4. 本指令一般都是使用脉冲执行型指令（RCLP、DRCLP）。

程序范例：

当 X0.0 从 OFF→ON 变化时，D10 的 16 个位连同进位标志 SM602 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 SM602 内。



补充说明：

1. 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

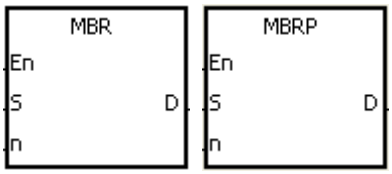
API	指令码			操作数	功能
0904		MBR	P	**S**・**D**・**n**	矩阵位循环

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●				○	○		

数据型态	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 矩阵来源装置

D ： 运算结果

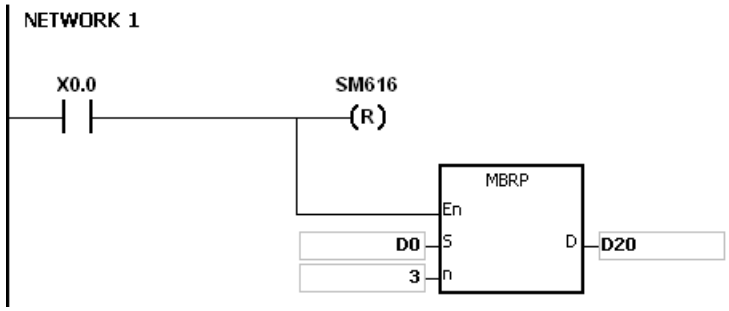
n ： 数组长度

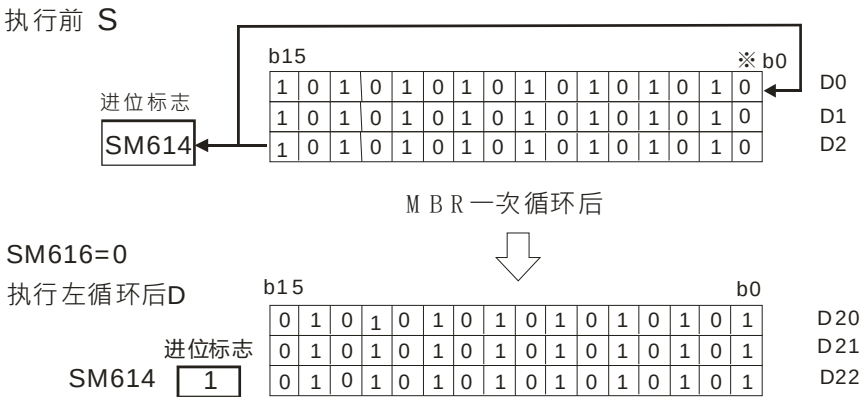
指令说明：

1. 矩阵来源依数组长度 **n** 将 **S** 矩阵位做左右循环控制，SM616=0 决定矩阵位左循环，SM616=1 决定矩阵位右循环。每次循环一位，因循环造成之空位（左循环时为 b0，右循环时为 b16n–1）由旋出位（左循环时为 b16n–1 右循环时为 b0）状态填补。将结果存入 **D**。旋出位不但用以填补前述之空位，同时并将之状态送到进位标志 SM614。
2. **n** 操作数值的范围为 1~256。当 **n**<=0 时，指令不执行，当 **n** 超过最大值时，以最大值执行。
3. 本指令一般都是使用脉冲执行型指令（MBRP）。

程序范例一：

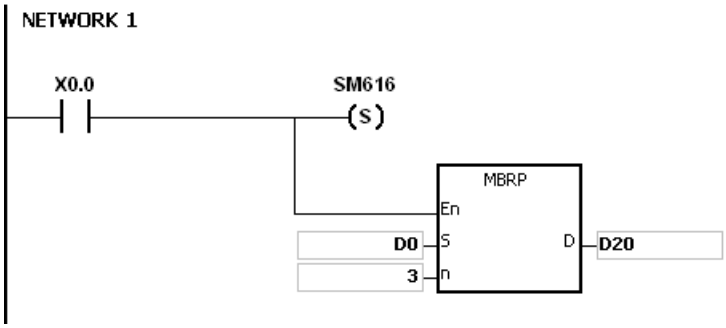
当 X0.0=ON 时 SM616=OFF 决定矩阵为左循环，16 位寄存器 D0~D2 矩阵循环，将结果存于 16 位寄存器 D20~D22 矩阵中，下图标明※位的内容将传送至进位标志 SM614。

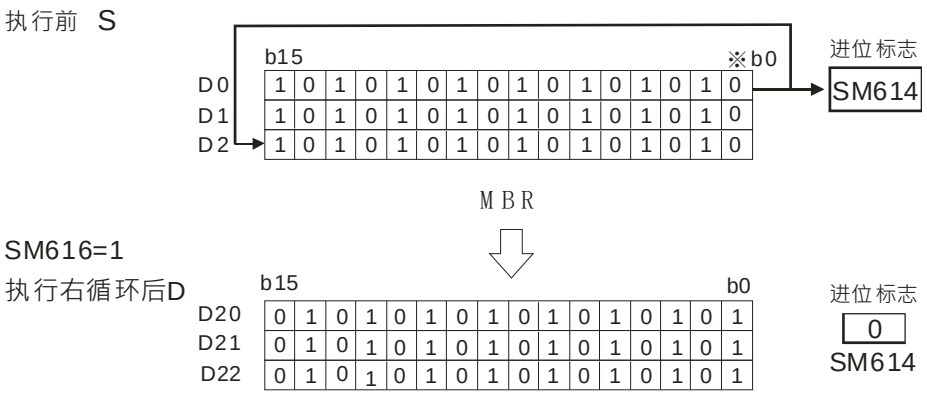




程序范例二：

当 X0.0=ON 时，SM616=ON 作矩阵右循环，16 位寄存器 D0~D2 矩阵作右循环，将结果存于 16 位寄存器 D20~D22 矩阵中，下图标明※位的内容将传送至进位标志 SM614。





补充说明：

1. S+n-1、D+n-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 标志信号说明：

   SM614： 矩阵循环位移输出进位标志

   SM616： 矩阵循环位移方向标志

6.11 计时计数指令

6.11.1 计时计数指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1000	RST	DRST	–	接点或寄存器清除
1001	TMR	–	–	16 位定时器（单位：100ms）
1002	TMRH	–	–	16 位定时器（单位：1ms）
1003	CNT	–	–	16 位计数器
1004	–	DCNT	–	32 位计数器（包含高速计数器使用说明）
1005	–	DHSCS	–	高速比较设定
1006	–	DHSCR	–	高速比较清除
1007	–	DHSZ	–	高速区域比较
1008	–	DSPD	–	速度检测指令
1009	PWD	–	–	脉宽检测指令
1010	–	DCAP	–	外部输入中断获取高速计数值
1011	TMRM	–	–	16 位定时器（单位：10ms）
1012	IETS	–	✓	指令运行时间量测开始
1013	IETE	–	✓	指令运行时间量测结束
1014	–	DHSCY	–	循环式计数搭配比较表格输出
1015	PPDT	–	–	两相输入之相位差时间检测
1016	ATMR	–	–	接点型定时器（100ms）
1017	ATMRH	–	–	接点型定时器（1ms）
1018	ATMRM	–	–	接点型定时器（10ms）

6.11.2 计时计数指令说明

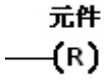
API	指令码			操作数	功能
1000	D	RST		**D**	接点或寄存器清除

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		○	○	○	○	○	○	○		○	○	○				

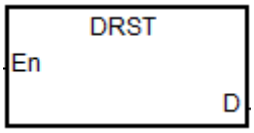
数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●	●	●			●	●		●		●	●	

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



D ：清除元件编号



指令说明：

1. DRST 指令仅支持清除 32 位 HC 装置以及连续两个 16 位 D 装置，其余元件请使用 RST 指令清除。
2. 当 RST 指令被驱动，其指定的元件的动作如下：

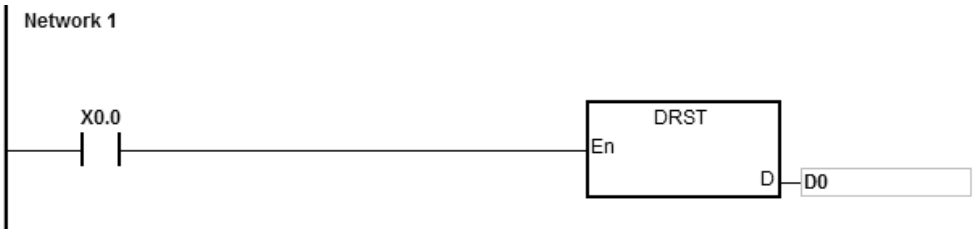
元件	状态
Bit	线圈及接点都会被设定为 OFF。
T，C	目前计时或计数值会被设为 0，且线圈及接点都会被设定为 OFF。
Word	16 位内容值会被清除为 0。
DWord，HC，Real	32 位（含浮点数类型）内容值会被清除为 0。

3. 若 RST 指令没有被执行，其指定元件的状态保持不变。
4. 本指令有支持 DY 装置。

程序范例：

当 X0.0=ON 时，Y0.5 会被设为 OFF。

当 X0.0=ON 时，32 位（D1，D0）会被清除为零。



API	指令码	操作数	功能
1001	TMR	S_1 · S_2	16 位定时器（100ms）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1					○											
S_2								○				○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1											●		
S_2		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：

S_1 ： 定时器编号

S_2 ： 定时器设定值

指令说明：

详细内容说明请参考 API1002 TMRH 指令的指令说明。

API	指令码	操作数	功能
1002	TMRH	S_1 · S_2	16 位定时器（1ms）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1					○											
S_2								○				○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1											●		
S_2		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：

S_1 ： 定时器编号

S_2 ： 定时器设定值

指令说明：

1. TMR 指令所指定的 T 定时器以 100ms 为单位计时，TMRH 指令所指定的 T 定时器以 1ms 为单位计时。
2. TMR 跟 TMRH 的定时器内容值范围为 0~32767。
3. 在程序中同一个 T 定时器如果重复使用（包含使用在不同指令 TMR、TMRH 中），则以最快到达的为主。
4. 当条件接点从 ON→OFF 时，T 计时值将自动归零。
5. 当使用 TMR 指令且需要计时数值持续累积时，只要在 T 之前加上一个 S，即可变成积分型定时器 ST 装置。此累积动作即是条件接点 OFF 时，其 T 的计时数值不会被清除，下次条件接点为 ON 时，T 由目前的数值开始累积计时。积分型 ST 定时器，若需清除计时值，请搭配使用 RST 指令。
6. 在程序中同一个 T 定时器如果重复使用，其中一个条件接点 OFF 时，则 T 会 OFF。
7. 在程序中同一个 T 定时器如果重复使用为 T 与 ST，其中一个条件接点 OFF 时，则 T 会 OFF。
8. 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>=设定值），其 T 接点（BOOL）动作将为 ON。
9. 编号 T0~T411 默认为一般型定时器，编号 T412~T511 默认为子程序型定时器，若有需要变更此范围，请由硬件组态 HWCONFIG 进行变更。
10. 一般型定时器于 TMR 指令扫描时比较，适用于每次皆会扫描到 TMR 指令的状况。子程序型定时器则于 END 指令被执行后，由系统自行计时与比较，故建议设计于不能每次扫描都执行的 TMR 指令，但又需要持续计时与比较时间的应用。

程序范例一：（一般 100ms 定时器使用方式）

当 X0.0 为 ON 时，定时器 T0 会加载设定值 50，当计时到达时 T0=50，T0 接点会 ON。

程序范例二：(积分型 100ms 定时器使用方式)

当 X0.0 为 ON 时，定时器 T0 会加载设定值 50，当定时器数到 T0=25 时，将 X0.0 OFF=>ON，定时器会从 T0=25 继续数到 T0=50，且 T0 接点会 ON。

程序范例三：(一般 1ms 定时器使用方式)

当 X0.0=ON 时，定时器 T5 会加载设定值 1000，当计时到达时 T5=1000，T5=ON。

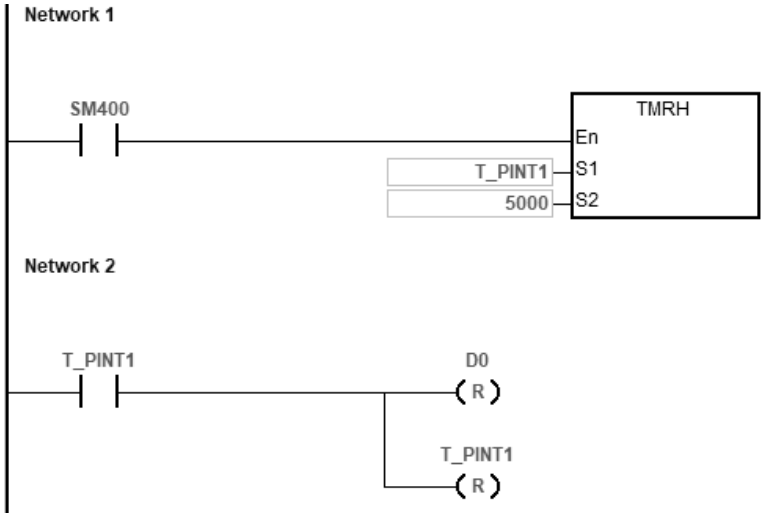
程序范例四：(积分型 1ms 定时器使用方式)

当 X0.0 为 ON 时，定时器 T5 会加载设定值 1000，当定时器数到 T5=500 时，将 X0.0 OFF=>ON，定时器会从 T5=500 继续数到 T5=1000，且 T5 接点会 ON。

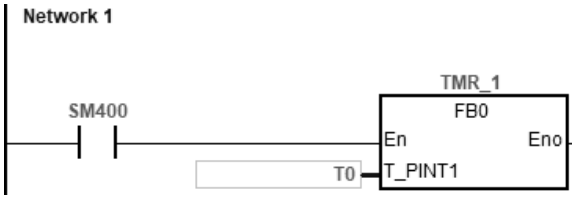
程序范例五：功能块 (FB) 内运用 TMRH 指令

步骤一、在功能块内宣告计数器为 T Pointer

功能块：每 5 sec 清除一次 D0



步骤二、在程序单元指定 T 元件编号（使用 T0）



补充说明：

S_1 操作数在 ISPSoft 宣告变量时，若使用一般 T 定时器，请选择数据类型为 TIMER；若使用积分型 ST 定时器，则需自行指定 ST 元件。

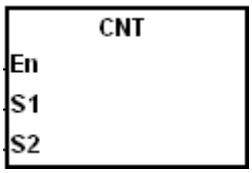
API	指令码	操作数	功能
1003	CNT	S_1，S_2	16 位计数器

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1						○										
S_2								○				○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 计数器编号

S_2 ： 计数器设定值

指令说明：

1. 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数值（计数值=设定值），其 C 接点（BOOL）动作将为 ON。
2. 当计数到达后，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。
3. 当 CNT 指令被运用于程序单元（POU）计数时，请参考范例一。当 CNT 指令被运用于功能块单元（FB）内计数时，请参考范例二。
4. 程序单元（POU）内，CNT 指令所搭配的 C 元件编号，允许重复使用相同编号，其计数值会依据不同条件接点的 OFF=>ON 而分开计数。
5. 功能块（FB）内，CNT 指令所搭配的 C 元件编号，不允许重复使用相同编号，而且建议最多使用 8 个不同 C 元件编号。(注：韧体版本低于 V1.07.00 版（不含），FB 内建议只使用 1 个 C 元件编号。韧体版本 V1.07.00 版（含）以上，FB 内建议最多可使用 8 个 C 元件编号。)

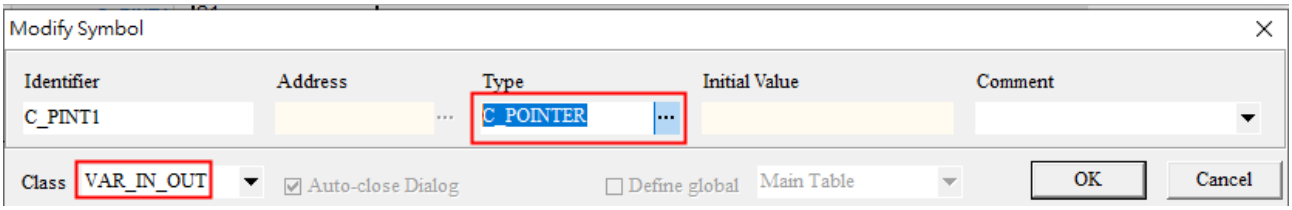
范例一：程序单元（POU）内运用 CNT 指令

当 SM408 的状态第一次 ON 时，计数器 C0 会加载设定值 10 并开始计数，当 SM408 的状态经过 10 次 OFF=>ON 的变化后，计数器 C0 的值计数到达 C0=10，且 C0 的接点为 ON。

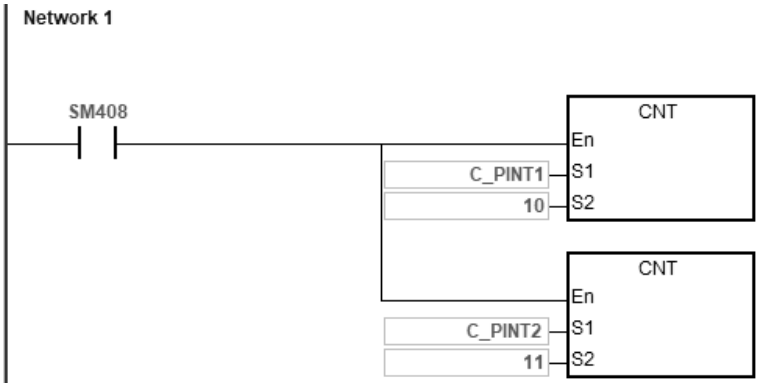
当 C0 为 ON 后，虽然 SM408 继续 OFF=>ON，但 C0 的值已经计数到达就不会再累加。

范例二：功能块（FB）内运用 CNT 指令

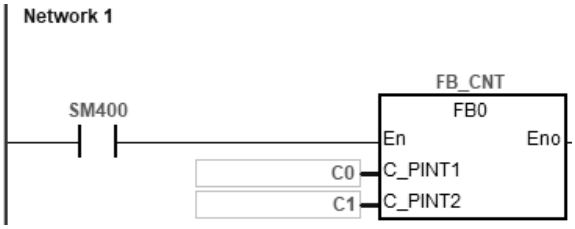
步骤一、在功能块内宣告计数器为 C Pointer



功能块



步骤二、在程序单元指定 C 元件编号（使用 C0 及 C1）



补充说明：

S_1 操作数于 ISPSoft 变量宣告时，请选择数据类型为 COUNTER。

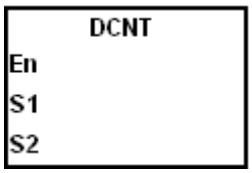
API	指令码	操作数	功能
1004	DCNT	S_1 · S_2	32 位计数器

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							○									
S_2								○					○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													
S_2			●				●						

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 计数器编号

S_2 ： 计数器设定值

指令说明：

1. DCNT 为 32 位计数器 HC0 至 HC255 的启动指令。
2. S_1 操作数于 ISPSoft 变量宣告时，不可选用 CNT 类型，需自行指定 HC 元件编号。
3. 上下数计数器 HC0~HC63，当 DCNT 指令条件接点由 OFF→ON 时，其计数器的数值将依 SM621~SM684 的状态 OFF 执行上数（加一）的动作或状态 ON 下数（减一）的动作。
4. 上数计数器 HC64~HC199，当 DCNT 指令条件接点由 OFF→ON 时，其计数器的数值将固定执行上数(加一)的动作。
5. 当 DCNT 指令 OFF 时，该计数器停止计数，但原有计数值不会被清除，可使用指令 RST 清除计数值及其接点。
6. 当 DCNT 指令被运用于功能块（FB）内计数时，请参考 CNT 指令说明内之使用范例二。
7. 高速计数器 HC200~HC255 使用说明，请参考后两页说明。由于高速计数过程中，PLC 内部将于每次扫描此指令时，进行刷新指定的计数器数值，因此建议使用在程序单元（POU）内。(注：高速计数器的编号，不允许重复使用)
8. AS 系列主机内建输入点因硬件材料限制，其相关建议最高输入频率如下表所示：

   注：下表所列的 Sink/Source 型输入点的最高输入频率，有可能因输出点为开集极输出类型，而无法达到预期的最高频率时，通常都是驱动电流不足所造成，建议可参考 AS 系列硬件及操作手册第 4 章硬件安装与配线建议说明。

型号	输入点类型	输入点编号	建议输入最高频率
AS324MT-A	差动型	X0.0 ~ X0.3	4MHz
	Sink / Source 型	X0.4 ~ X0.11	200KHz

型号	输入点类型	输入点编号	建议输入最高频率
AS332T-A AS332P-A	Sink / Source 型	X0.0 ~ X0.11	200KHz
		X0.12 ~ X0.15	10KHz
AS320T-B AS320P-B	Sink / Source 型	X0.0 ~ X0.7	200KHz
AS300N-A		无内建输入点	
AS228T-A AS228P-A AS228R-A	Sink / Source 型	X0.0 ~ X0.7	200KHz
		X0.8 ~ X0.15	10KHz
AS218TX-A AS218PX-A AS218RX-A	Sink / Source 型	X0.0 ~ X0.7	200KHz

程序范例一：

NETWORK1→

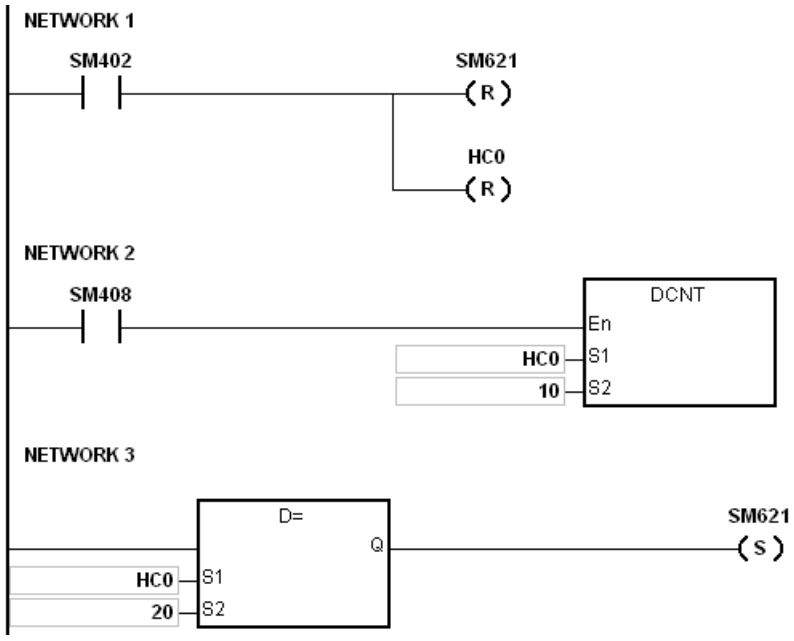
当 PLC RUN 时，定时器 HC0 被清除，且 SM621=OFF 将执行上数，此时 SM408 的状态为第一次 ON，所以计数器 HC0 会加载设定值 10 并开始计数。

NETWORK2→

当 SM408 的状态经过 10 次 OFF=>ON 的变化后，计数器 HC0 的值计数到达 HC0=10，且 HC0 的接点为 ON。当 HC0 为 ON 后，因为 SM408 继续 OFF=>ON，所以虽然 HC0 的值已经计数到达，但还是会继续累加。

NETWORK3→

当计数器继续上数到达 HC0=20 时，程序会设定 SM621=ON 将执行下数，当 SM408 的状态再经过 10 次 OFF=>ON 的变化后，计数器 HC0 的值计数到达由 HC0=10 递减到 HC0=9 时，HC0 的接点会被 OFF。当 HC0 的接点被 OFF 后，因为 SM408 继续 OFF=>ON，所以 HC0 的值还是会继续递减。



补充说明：

1. SM621~SM684 的设定模式，请参考第 2 章的 32 位计数器 HC 的使用说明。

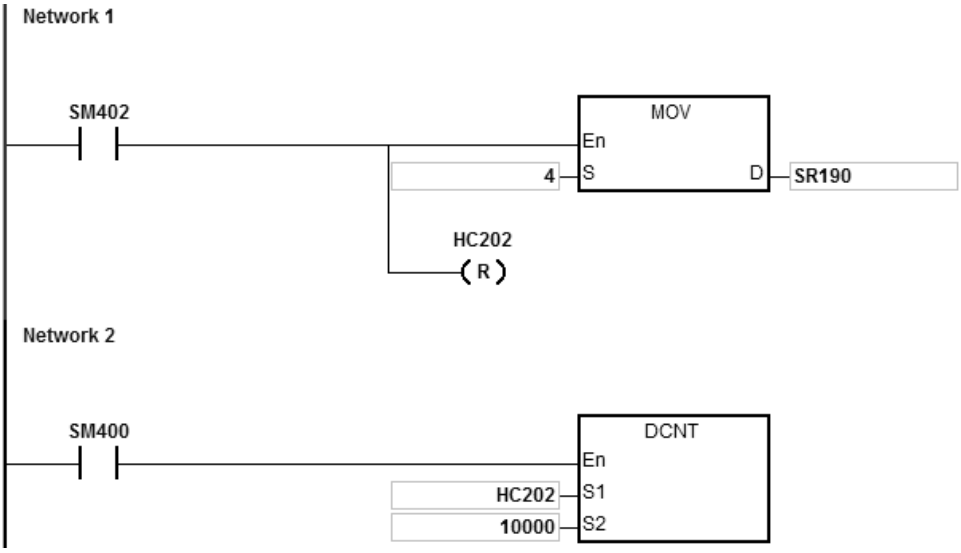
程序范例二：

NETWORK1→

当 PLC RUN 时，将 HC202 设定为 4 倍频计数(模式设定要在 DCNT 指令 ENABLE 之前设定)，定时器 HC202 被清除。

NETWORK2→

当计数器继续上数到达 HC202=10000 时，HC202 为 ON。



高速计数器说明：

AS 系列主机的高速计数器分为两大类，一类为硬件型计数器 (最高可输入达 200kHz)(差动输入点可达 4MHz 输入)，详细各输入点最高可输入频率，请参考各个主机的硬件规格说明，另一类为软件型计数器 (最高可达 10kHz)，其计数器编号可搭配的输入点与计数功能如下两表所示：

硬件型计数器：

Input / HC No.	X0.															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
HC200	P#1												R#4			
HC201	P	D#1											R			
HC202	A#1	B#1											R			
HC203	--#2	--											--			
HC204			P											R		
HC205			P	D										R		
HC206			A	B										R		

Input / HC No.	X0.															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
HC207			--	--										--		
HC208					P										R	
HC209					P	D									R	
HC210					A	B									R	
HC211					--	--									--	
HC212							P									R
HC213							P	D								R
HC214							A	B								R
HC215							--	--								--
HC216									P							
HC217									P	D						
HC218									A	B						
HC219									--	--						
HC220											P					
HC221											P	D				
HC222											A	B				
HC223											--	--				

注 1：P 表示单相脉冲输入、D 表示带方向输入、A 及 B 表示两相两输入、R 表示清除信号输入。以上 4 种输入方式，在 PLC 程序编辑时只能选择 1 种方式，举例：已经编辑 HC200，则 HC201~HC203 将不可再编辑。ISPSoft 编译时不会检查重复编辑的限制条件，但下载至 PLC 时，PLC 将响应重复使用的错误 SR4=16#0E05。

注 2：‘--‘符号表示此计数器模式保留还不可使用，空白处字段则无任何功能。

注 3：上/下数状态选择或显示，以及输入倍频选择，请参考后面 SM/SR 功能对照表说明。

注 4：R 输入点清除功能默认为不启动，若需要使用请参考 SM/SR 功能对照表说明。举例：以 HC200 为例，先起动清除功能(SM291 为 ON)，则 X0.12 为上升沿触发清除一次 HC200。当所选用主机无内建 X0.12~X0.15 输入点时，清除功能请自行挑选外部中断，以及编写清除计数器指令。

软件型计数器：

Input / HC No.	X0.															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
HC232													P			
HC233													P	D		
HC234													A	B		
HC235													UP#5	DN#5		
HC236															P	
HC237															P	D

Input / HC No.	X0.															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
HC238															A	B
HC239															UP	DN
HC240		P														
HC241	UP	DN														
HC242				P												
HC243			UP	DN												
HC244						P										
HC245					UP	DN										
HC246								P								
HC247							UP	DN								
HC248										P						
HC249									UP	DN						
HC250												P				
HC251											UP	DN				
HC252														P		
HC253																P

注 5：UP 表示单相上数输入（如同 CW）、DN 表示单相下数输入（如同 CCW）。

以上 HC200~HC255 范围内未被列出的高速计数器，皆为 PLC 内部保留的装置元件，不建议程序设计者运用。

当所选用主机无内建 X0.12~X0.15 输入点时，则其对应的计数器请勿使用。

高速计数器搭配的 SM/SR 功能对照表：

	上 / 下数功能			清除功能启动	强制反向功能	倍频模式设定
HC 编号	SM 编号	属性	功能说明	SM 编号	SM 编号	SR 编号
HC200	SM300	R/W	显示 / 设定	SM291	SM281 （HC201 适用） （HC202 适用#7）	SR190
HC201	SM301	R	显示			
HC202	SM302	R	显示			
HC204	SM304	R/W	显示 / 设定	SM292	SM282 （HC205 适用） （HC206 适用#7）	SR191
HC205	SM305	R	显示			
HC206	SM306	R	显示			
HC208	SM308	R/W	显示 / 设定	SM293	SM283 （HC209 适用） （HC210 适用#7）	SR192
HC209	SM309	R	显示			
HC210	SM310	R	显示			
HC212	SM312	R/W	显示 / 设定	SM294	SM284 （HC213 适用） （HC214 适用#7）	SR193
HC213	SM313	R	显示			
HC214	SM314	R	显示			

	上 / 下数功能			清除功能启动	强制反向功能	倍频模式设定
HC 编号	SM 编号	属性	功能说明	SM 编号	SM 编号	SR 编号
HC216	SM316	R/W	显示 / 设定	--	SM285 (HC217 适用) (HC218 适用#7)	SR194
HC217	SM317	R	显示			
HC218	SM318	R	显示			
HC220	SM320	R/W	显示 / 设定	--	SM286 (HC221 适用) (HC222 适用#7)	SR195
HC221	SM321	R	显示			
HC222	SM322	R	显示			
HC232	SM332	R/W	显示 / 设定	--	SM287 (HC233 适用) (HC234 适用#7)	SR196
HC233	SM333	R	显示			
HC234	SM334	R	显示			
HC235	SM335	R	显示			
HC236	SM336	R/W	显示 / 设定	--	SM288 (HC237 适用) (HC238 适用#7)	SR197
HC237	SM337	R	显示			
HC238	SM338	R	显示			
HC239	SM339	R	显示			
HC240	SM340	R/W	显示 / 设定	--	--	仅支持 1 倍频且为上升沿计数
HC241	SM341	R	显示			
HC242	SM342	R/W	显示 / 设定			
HC243	SM343	R	显示			
HC244	SM344	R/W	显示 / 设定			
HC245	SM345	R	显示			
HC246	SM346	R/W	显示 / 设定	--	--	仅支持 1 倍频且为上升沿计数
HC247	SM347	R	显示	--	--	仅支持 1 倍频且为上升沿计数
HC248	SM348	R/W	显示 / 设定			
HC249	SM349	R	显示			
HC250	SM350	R/W	显示 / 设定			
HC251	SM351	R	显示			
HC252	SM352	R/W	显示 / 设定			
HC253	SM353	R	显示			

注 1：以上表格所有 SM 特殊标志其默认值皆为 OFF 状态。

注 2：当上/下数功能内的 SM 为 OFF 时，即表示其计数器进行”上数”或显示”上数中”；若 SM 为 ON 时，则表示其计数器进行”下数”或显示”下数中”。

注 3：属性字段内”R”表示”只读”，”W”表示可做”设定”功能。

注 4：倍频模式 SR 特殊寄存器皆默认为 1 倍频，若需使用 2 倍频请输入 2，若需使用 4 倍频则请输入 4，此 4 倍频仅有使用 A/B 两相输入的计数器有效。若是倍频模式不为 1，2，4 这三种数值时，PLC 将自动以 1 倍频执行。

注 5：以上所有单相计数器使用 1 倍频计数时，皆采用上升沿计数模式，即输入点从 OFF→ON。

注 6：强制反向功能，适用于“脉冲输入（P） + 方向（D）” 的计数器，若此 SM 为 ON 时，则表示上/下数行为会被强制反向，举例：默认方向输入为 OFF 时，计数器为累加方式计数，当此 SM 被设为 ON 时，计数器将改为递减方式计数。

注 7：V1.08 版（含）以上支持

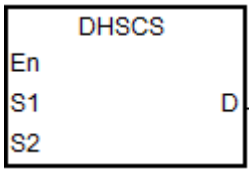
API	指令码			操作数	功能
1005	D	HSCS		**S_1 · S_2 · D**	高速输入比较设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							○									
S_2								●					○	○		
D		○	○	○				○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
D	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 计数器编号

S_2 ： 高速比较值

D ： 设定目标

指令说明：

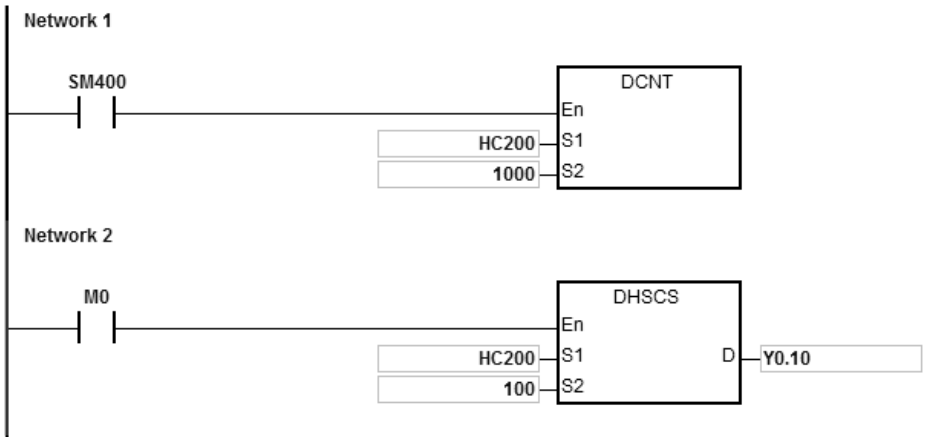
1. 使用此高速比较指令时需搭配启动高速计数器编号 HC200（含）以上，当由 **S_1** 所指定的高速计数器产生加 1 或减 1 变化时，DHSCS 指令会立即做比较动作，当高速计数器当前值等于由 **S_2** 所指定的比较值时，由 **D** 所指定的装置会变为 ON，之后即使比较结果变成不相等，该装置仍然保持 ON 状态。
2. 若 **D** 所指定的装置为 Y0.0~Y0.15 时，当 **S_2**（比较值）与高速计数器当前值相等会实时输出到外部 Y0.0~Y0.15 实际输出端，其余的 Y 装置会受扫描周期影响。而装置均为立即动作，不受扫描周期的影响。V1.06（含）版以上，指定的装置为 Y0.12~Y0.15，也会受扫描周期影响。
3. **D** 操作数也可指定中断 I 装置，其范围 I2□□，□□=00~67。
4. 高速计数器分为软件型与硬件型计数器两种。其可使用高速比较器与中断编号列表及说明请参考下表。

类别	计数器编号范围	内部高速比较器编号	高速比较中断编号
硬件型	HC200 ~ HC203	比较器 HCC00~HCC03	I200~I203
	HC204 ~ HC207	比较器 HCC04~HCC07	I210~I213
	HC208 ~ HC211	比较器 HCC08~HCC11	I220~I223
	HC212 ~ HC215	比较器 HCC12~HCC15	I230~I233
	HC216 ~ HC219	比较器 HCC16~HCC19	I240~I243
	HC220 ~ HC223	比较器 HCC20~HC223	I250~I253
软件型	HC232 ~ HC253	-	I260~I267

5. DHSCS，DHSCR，DHSCY，DHSZ 指令硬件计数比较器说明：
   - 每一组硬件计数器共享 4 个高速比较器，每一个 DHSCS，DHSCR，DHSCY 指令占用 1 个高速比较器，每一个 DHSZ，DHSCY 指令占用 2 个高速比较器。
   - 编写程序时不管是 DHSCS、DHSCR，DHSCY 或是 DHSZ 指令，每一组硬件计数器不可使用超过 4 个高速比较器，否则会有文法检查错误。
6. DHSCS，DHSCR，DHSCY 指令软件计数比较器说明：
   - 软件比较器提供有 8 个比较器来比较设定或清除功能。每 1 个 DHSCS，DHSCR，DHSCY 指令占用 1 个高速比较器。
   - 软件计数比较器的比较中断是依中断编号分别配置一个对应的软件比较器，需注意不可重复使用相同中断编号。。
   - 编写程序时不管是 DHSCS 、DHSCR 或 DHSCY 指令，其三个指令的设定或清除比较器最多不可超过 8 个，否则会有文法检查错误。
7. DHSZ 指令软件计数比较器说明：
   - 软件比较器提供 8 个比较器来执行区域比较功能。每 1 个 DHSZ 指令们用 1 个比较器。
   - 编写程序时 DHSZ 指令最多不可超过 8 个，否则会有文法检查错误。
8. DHSCS，DHSCR，DHSCY，DHSZ 指令不建议使用于 ST 语言编辑、中断服务程序与仅被调用一次型功能块（FB）。

程序范例一：

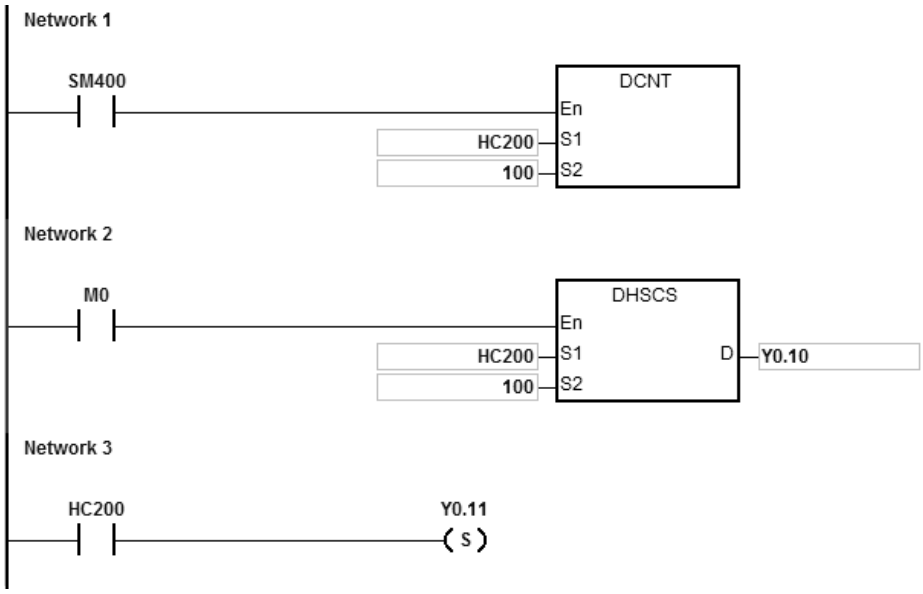
M0=ON 时，执行 DHSCS 指令，当 HC200 的当前值由 99→100 或 101→100 变化时，Y0.10=ON 实时输出到外部 Y0.10 输出端，且一直保持为 ON。



程序范例二：

DHSCS 指令 Y 输出与一般 Y 输出的差异：

1. M0=ON 时，执行 DHSCS 指令，当 HC200 的当前值由 99→100 或 101→100 变化时，Y0.10 是立即输出到外部输出端，与 PLC 扫描时间无关。
2. 当 HC200 的当前值由 99→100 变化时，HC200 接点立即导通，但执行到 SET Y0.11 时，Y0.11 仍会受扫描时间影响，在 END 后才输出。

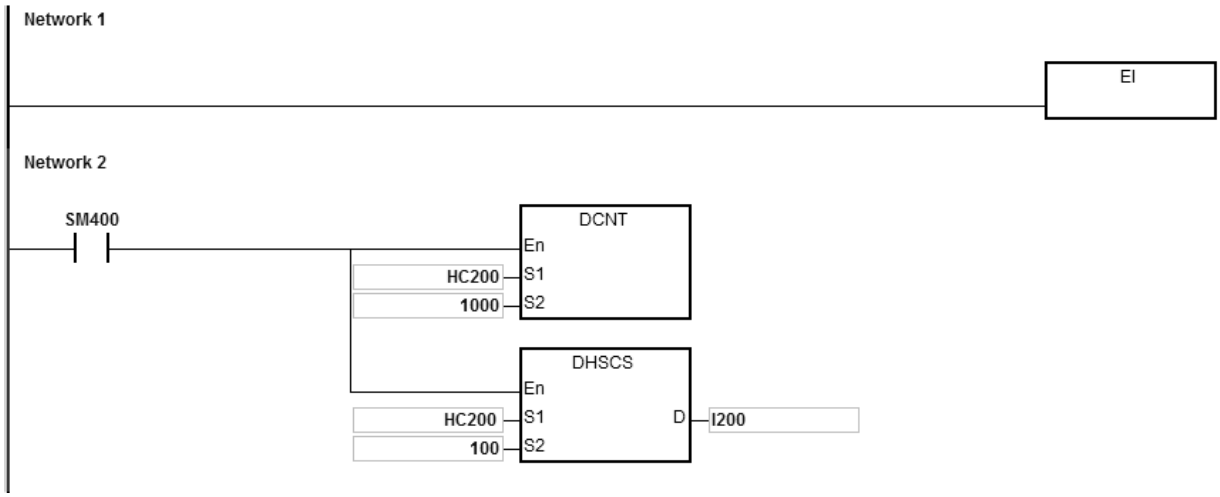


程序范例三：

硬件高速比较中断：

当 HC200 的当前值由 99→100 或 101→100 变化时，程序跳到中断指针 I200 处执行中断程序，Y0.10=ON。

主程序：



I200 中断程序：

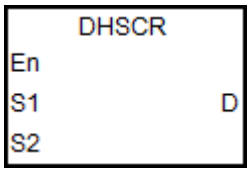
API	指令码			操作数	功能
1006	D	HSCR		**S_1 · S_2 · D**	高速输入比较清除

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							○									
S_2								●					○	○		
D		○	○	○			○	○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
D	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 计数器编号

S_2 ： 高速比较值

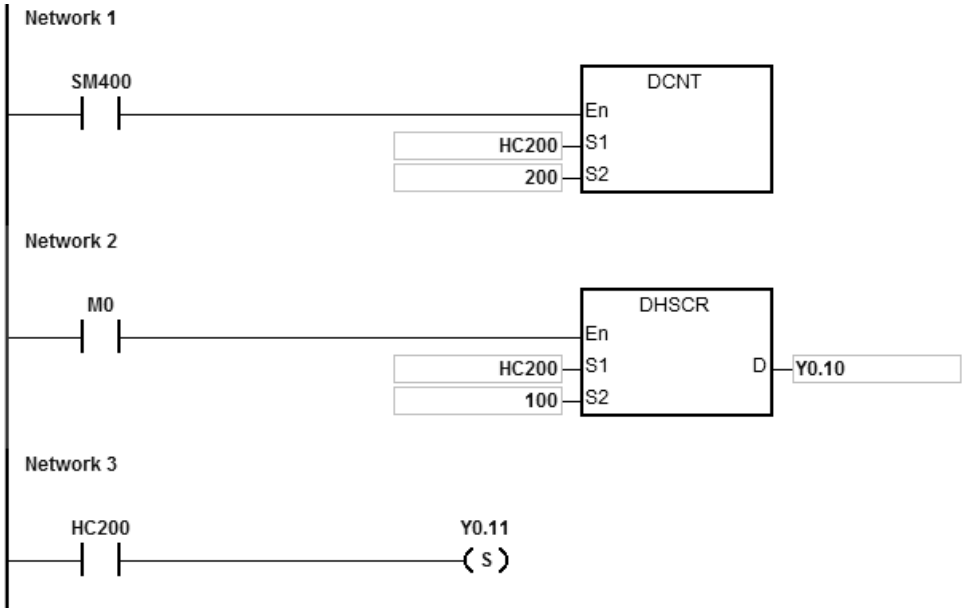
D ： 清除目标

指令说明：

1. 使用此高速比较指令时需搭配启动高速计数器编号 HC200 (含) 以上，当由 **S_1** 所指定的高速计数器产生加 1 或减 1 变化时，DHSCR 指令会立即做比较动作，当高速计数器当前值等于由 **S_2** 所指定的比较值时，由 **D** 所指定的装置会变为 OFF，之后即使比较结果变成不相等，该装置仍然保持 OFF 状态。
2. 若 **D** 所指定的装置为 Y0.0~Y0.15 时，当 **S_2** (比较值) 与高速计数器当前值相等会实时输出到外部 Y0.0~Y0.15 实际输出端，其余的 Y 装置会受扫描周期影响。而装置均为立即动作，不受扫描周期的影响。
   V1.06 (含) 版以上，指定的装置为 Y0.12~Y0.15，也会受扫描周期影响。
3. **D** 操作数也可指定清除 HC 装置，但是仅限制高速计数器编号与 **S_1** 相同。
4. 其余相关使用说明请参考 DHSCS 指令。

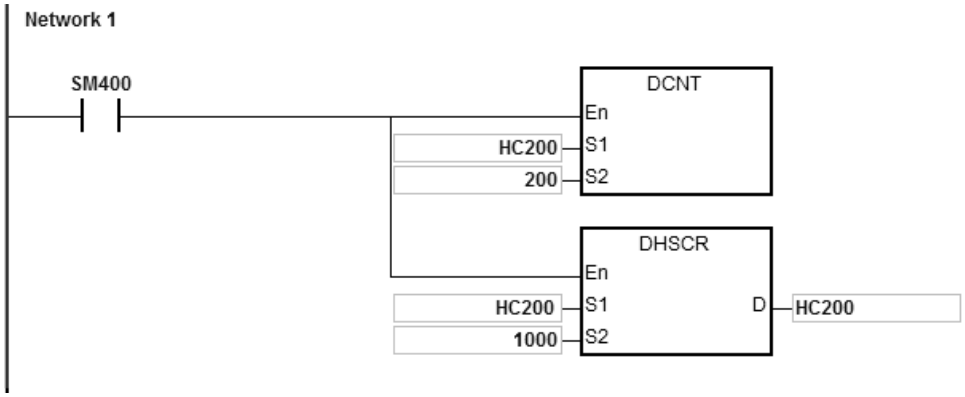
程序范例一：

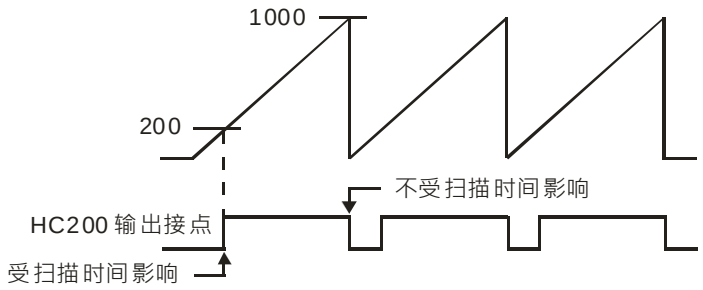
1. 当 M0=ON 且 HC200 的当前值从 99→100 或 101→100 变化时，Y0.10 会被清除 OFF。
2. 当 HC200 的当前值从 199→200 时，HC200 接点会 ON，使 Y0.11=ON，但会有程序扫描时间延迟输出。



程序范例二：

指定为同一编号的硬件高速计数器 HC200，当 HC200 的当前值从 999→1000 或 1001→1000 变化时，HC200 接点会被清除为 OFF。





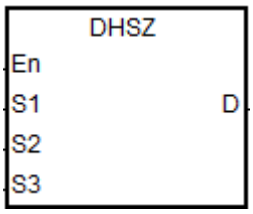
API	指令码			操作数	功能
1007	D	HSZ		**S_1**，**S_2**，**S_3**，**D**	高速输入区域比较

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1							○									
S_2								●					○	○		
S_3								●					○	○		
D		○	○	○				○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
S_3			●				●						
D	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 计数器编号

S_2 ： 区域比较下限值

S_3 ： 区域比较上限值

D ： 设定目标（连续占用 3 点）

指令说明：

1. 使用此高速区域比较指令时需搭配启动高速计数器编号 HC200（含）以上，**S_2** 下限值必须小于 **S_3** 上限值，若此区域限制值设定错误，PLC 将自动调整。
2. **S_1** 使用软件计数器时，当所指定的计数器产生加 1 或减 1 变化时，DHSZ 指令会立即做比较动作。其比较动作与输出状态如下表所示。

比较条件	**D**+0 状态	**D**+1 状态	**D**+2 状态
S_1 计数值 < **S_2** 下限值	ON	OFF	OFF
S_2 下限值 <= **S_1** 计数值 < **S_3** 上限值	OFF	ON	OFF
S_1 计数值 >= **S_3** 上限值	OFF	OFF	ON

注：**S_2** 下限值必须小于 **S_3** 上限值，若此区域限制值设定错误，PLC 将自动调整。

3. **S_1** 使用硬件计数器时，当所指定的计数器数值到达 **S_2** 下限值或是 **S_3** 上限值时，DHSZ 指令会依计数方向（上数/下数）立即做比较动作。其比较动作与输出状态如下表所示。

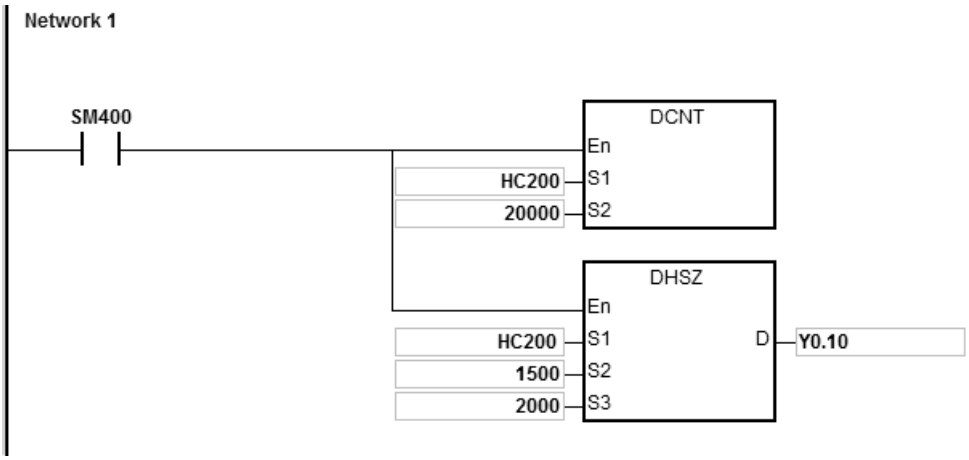
计数方向	比较条件	**D**+0 状态	**D**+1 状态	**D**+2 状态
上数	**S_1** 计数值 == **S_2** 下限值	OFF	ON	OFF
	S_1 计数值 == **S_3** 上限值	OFF	OFF	ON

计数方向	比较条件	**D**+0 状态	**D**+1 状态	**D**+2 状态
下数	**S_1** 计数值 == **S_2** 下限值	ON	OFF	OFF
	S_1 计数值 == **S_3** 上限值	OFF	ON	OFF

4. 若 **D** 所指定的装置为 Y0.0~Y0.15 时，将实时输出到外部 Y0.0~ Y0.15 实际输出端，其余的 Y 装置将会受扫描周期影响。而装置均为立即动作，不受扫描周期的影响。
   V1.06（含）版以上，指定的装置为 Y0.12~Y0.15，也会受扫描周期影响。
5. 高速区域比较相关使用说明请参考 DHSCS 指令。

程序范例：

1. 指定装置为 Y0.10，则自动占有 Y0.11~Y0.12。
2. 当 DHSZ 指令执行时，高速计数器 HC200 在有计数输入时，到达上下限值，Y0.10~Y0.12 其中之一会 ON。
3. 当 HC200 当前值<1500 时，Y0.10=ON，1500<=HC200 当前值<2000 时，Y0.11=ON，HC200 当前值>=2000 时，Y0.12=ON。



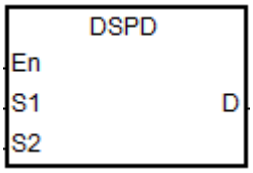
API	指令码			操作数	功能
1008	D	SPD		**S_1，S_2，D**	速度检测

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1							○									
S_2								○	○				○	○		
D								○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
D			●				●						

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 计数器编号

S_2 ： 计时周期时间

D ： 前次周期时间内计数的脉冲个数

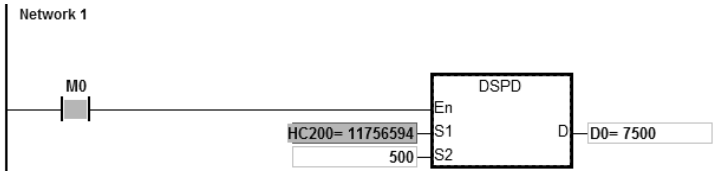
指令说明：

1. 使用此速度检测指令时，**S_1** 需搭配 DCNT 指令启动高速计数器 HC200（含）以上的计数器编号。
2. **S_2** 为计时周期时间的设定值，其计时单位为毫秒（ms），数值范围为 10~10000，超出范围时以最小或最大值执行，PLC 不会发出错误信息。
3. 当由 **S_2** 所指定计时时间到达时，此指令会立即将计数的脉冲个数，存放至 **D** 所指定的装置内，故此指令不受 PLC 扫描动作影响。
4. 此指令不限制编辑次数，但最多可同时启动 8 组速度检测指令，当启动达到 8 组之后，后来启动的指令将无效，也不会有错误信息产生。此指令启动时，将会记录操作数所设定的参数，因此执行过程中，不允许进行参数变更。

程序范例：

当 X0.0 有一输入脉冲信号，可以藉由 DSPD 指令作速度检测，当 M0=On 时，DSPD 指令每 500ms 会将 HC200 数到的脉冲数显示在 D0。

以此范例 D0=7500，实际由 X0.0 输入的脉冲频率为 15kHz(频率=7500/500ms)



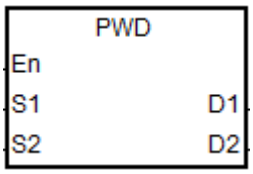
API	指令码	操作数	功能
1009	PWD	**S_1 · S_2 · D**	脉宽检测

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	○															
S_2								○	○				○	○		
D_1								○								
D_2		○	○	○												

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2		●				●							
D_1			●				●						
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 输入点编号

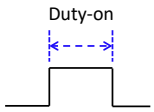
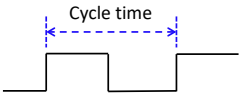
S_2 ： 计时单位设定

D_1 ： 脉宽检测时间（32 位数值）

D_2 ： 已更新标志

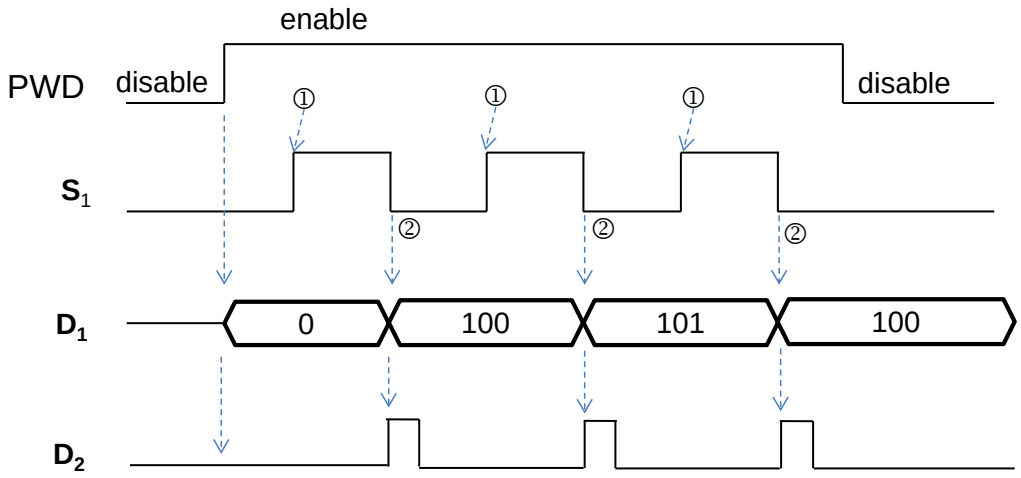
指令说明：

1. **S_1** 仅支援 X0.0~X0.11 这 12 点输入，并且不可与高速计数器共享输入点。
2. **S_2** 为计时单位参数设定值，当 **S_2** 设定值不为下表内的参数代码时，指令将不会执行。

S_2 参数代码	计时单位	检测范围	适用频率范围	备注
0	1us		1Hz ~ 10kHz	
1	1ms		0.02Hz ~ 100Hz	
2	10ns		10Hz ~ 1MHz	奇数点不支持
4	1us		1Hz ~ 10kHz	
5	1ms		0.02Hz ~ 100Hz	
其它数值	指令不执行			

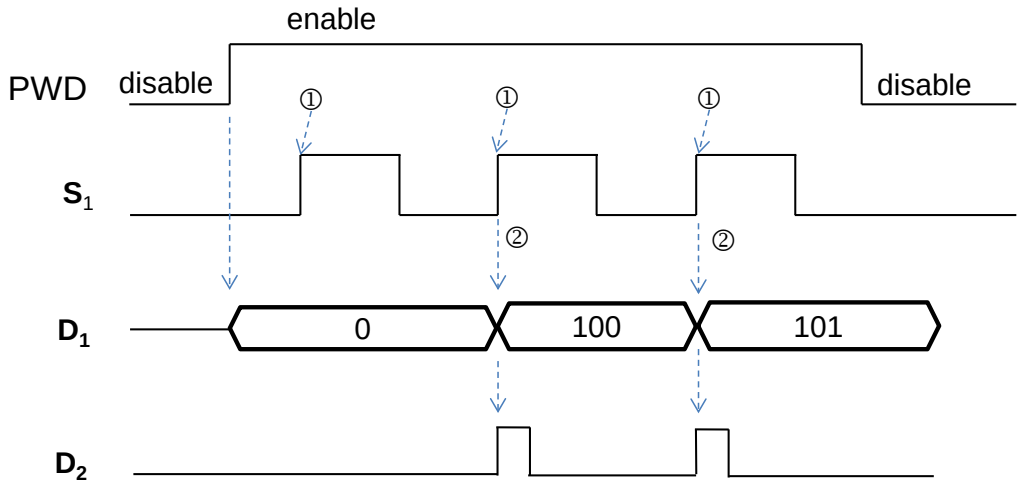
3. **D_1** 为储存脉宽检测时间（32 位数值），其检测数值范围为 0~100,000,000，若是数值已达最大值，则表示已超出计时范围；若是数值为 0，则表示指令启动后，一直未检测到有输入 ON 变 OFF 的状态。

4. **D_2** 为已**更新**标志，当每次 **S_1** 输入点完成一次检测时，并且指令被扫描到的时候，此更新标志将会被设定 ON 一个扫描周期。用户可依据此标志得知检测数值是否已更新为最新检测数值。指令第一次启动时，指定的更新标志也会被清除为 OFF。
5. 当 **S_2** 参数选择 0~2 时，其脉宽检测启动后，相关的储存检测值与更新标志的动作时序图，请参考下图(一)所示。启动计时的时机为 **S_1** 输入点从 OFF→ON 时，如下图内①的位置；储存检测时间的时机为 **S_1** 输入点从 ON→OFF 时，如下图内②的位置。



图(一)参数 0~2 检测模式

6. 当 **S_2** 参数选择 4~5 时，其脉宽检测启动后，相关的储存检测值与更新标志的动作时序图，请参考下图(二)所示。启动计时的时机为 **S_1** 输入点从 OFF→ON 时，如下图内①的位置；储存检测时间的时机为 **S_1** 输入点下一次从 OFF→ON 时，如下图内②的位置。

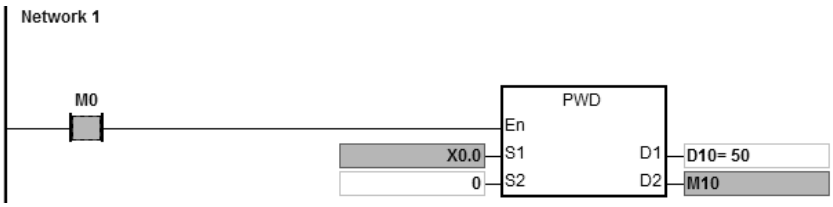


图(二)参数 4~5 检测模式

7. 此指令不限制编辑次数，但最多可同时启动 8 组不同输入点的脉宽检测指令，当启动达到 8 组后，后来启动的指令将无效，也不会有错误信息产生。此指令启动时，将会记录操作数所设定的参数，因此执行过程中，不允许进行参数变更。
8. 采用此检测指令时，请注意搭配的输入点硬件响应时间，以及 HWCONFIG 设定的滤波时间，举例：**S_2** 设定数值为 0 或 2 时，表示采用μs 为基底时间，那么其 HWCONFIG 的指定 **S_1** 输入点滤波时间，应设为 0 关闭滤波功能。

程序范例：

X0.0 有一输入脉冲信号 10kHz，当 M0=On 时，PWD 指令可以针对 X0.0 的输入信号做检测，将信号的脉冲宽度显示在 D10/D11（32 位数据），时间单位设定为 0，此时 D10 检测到的宽度为 50μS



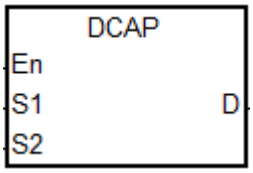
API	指令码			操作数	功能
1010	D	CAP		**S_1 · S_2 · D**	外部输入中断获取高速计数值

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	○															
S_2							○									
D								○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2												●	
D			●				●						

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 外部中断输入点编号

S_2 ： 高速计数器编号

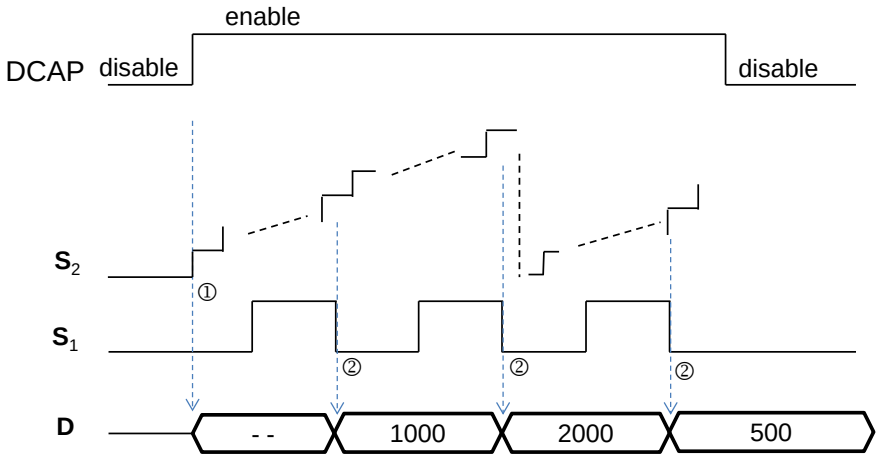
D ： 获取计数值存放位置

指令说明：

1. **S_1** 限制选用主机上的 X0.0~X0.15 这 16 个输入点，并且要自行搭配外部中断服务程序启动此功能。当选择输入点时，请勿与高速计数器共享相同输入点。
2. **S_2** 为选择高速计数器 HC 元件，此计数器需自行搭配 DCNT 指令去启动计数功能。
3. **D** 为储存中断发生时立即获取高速计数器数值的存放位置（32 位数值），其储存时机在中断发生时，并且不受 PLC 程序扫描影响。
4. 此指令的运作动作示意图如下所示：（假设输入中断为下降沿触发）

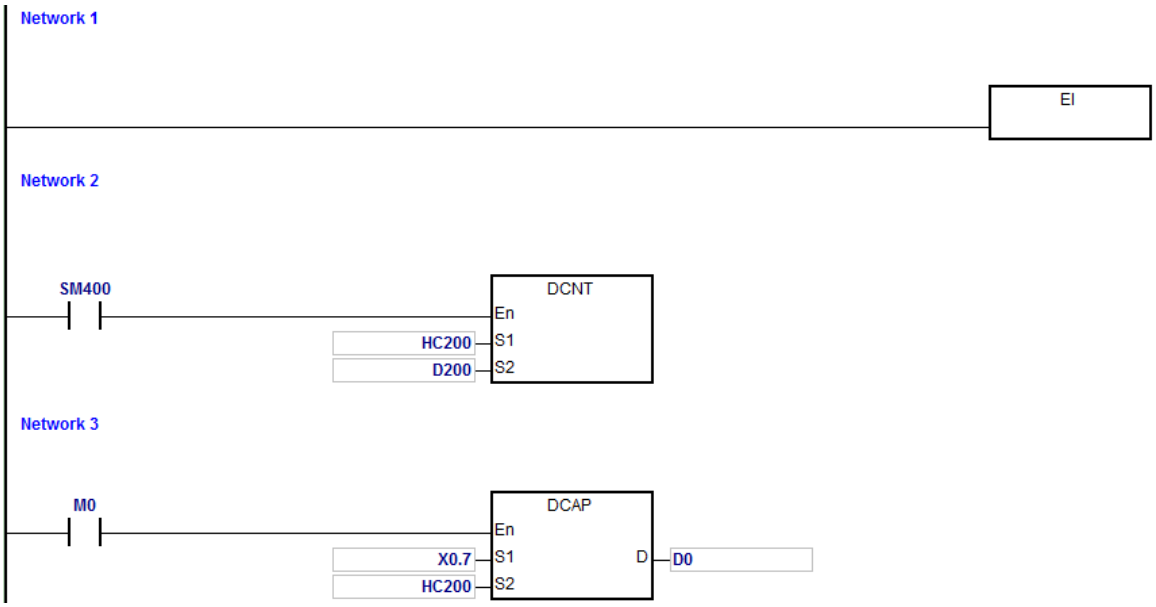
   ① → 指令开始启动，**D** 存放数值不会变动，用户可自行事先填入默认值。

   ② → **S_1** 输入中断发生，立即获取指定 **S_2** 的计数值，并存放置 **D**。

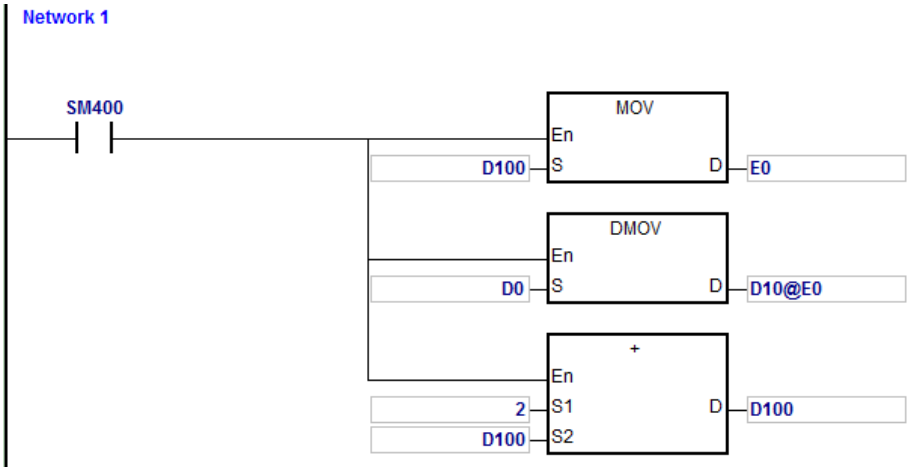


5. 此指令最多可启动 4 个不同输入点的 DCAP 指令，若是一个输入点有安排上升沿与下降沿的外部输入中断，则此获取时机就会分别在上升沿与下降沿触发时，储存一次计数值至指定 **D** 位置。当两个指令指定同一个中断输入点时，以先启动的指令先占用。
6. **S_2** 为 HC 装置元件的计数器编号，建议选用 HC200~HC255 之间的高速计数器。计数器详细说明请参考 API1004 DCNT 指令。

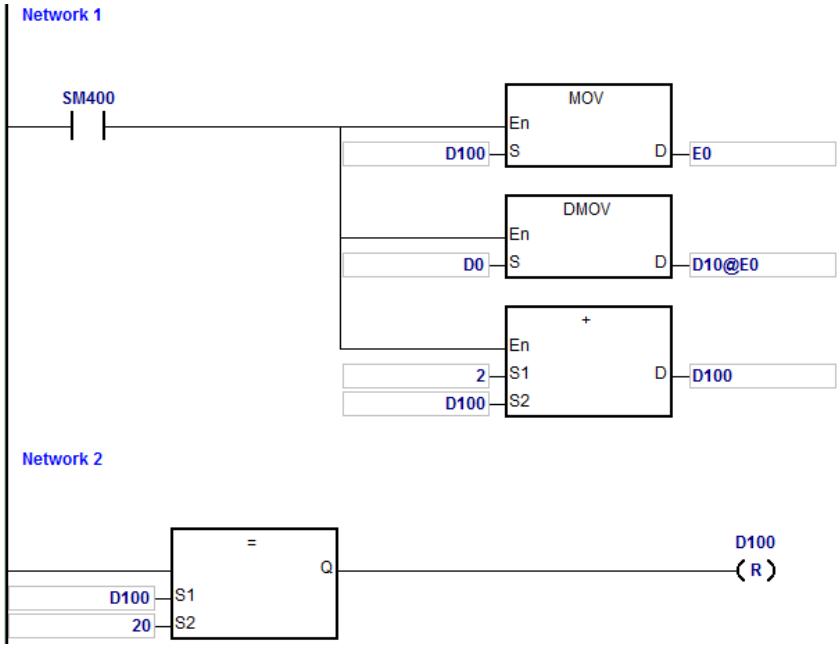
程序范例 1：



外部中断 X0.7_上升沿触发



外部中断 X0.7_下降沿触发



说明：

1. 当 M0 为 ON，启动 DCAP 指令，当 X0.7 有外部中断产生时，将 HC200 计数值获取至 D0(32 位数据)
2. 当外部输入中断上升沿触发一次，先透过 D100 设定间接修饰 E0=0，透过间接修饰 E0，会将 D0 的计数值摆放至 D10；并且 D100=0+2
3. 当外部输入中断下降沿触发一次，透过 D100 设定间接修饰 E0=2，透过间接修饰 E0，会将 D0 的计数值摆放至 D12(10+E0=12)，并且 D100=0+2。当 D100=20 时，将 D100 清除为 0
4. 若外部中断上下触发各 5 次(获取 10 次)，会将获取的数值存放在 D10，D12…D28

   第 1 次获取的值=D10

   第 2 次获取的值=D12

   .

   .

   第 10 次获取的值=D28

   第 11 次获取的值=D10

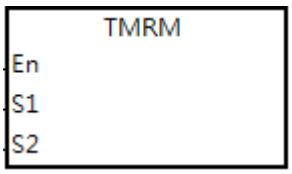
API	指令码		操作数	功能
1011	TMRM		$S_1 \cdot S_2$	16 位定时器（10ms）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1					○											
S_2								○				○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1											●		
S_2		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 定时器编号

S_2 ： 定时器设定值

指令说明：

此指令以 10ms 为基本单位计时，其详细内容说明请参考 API1002 TMRH 指令的指令说明。

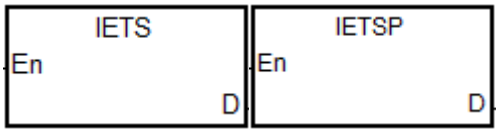
API	指令码			操作数	功能
1012		IETS	P	**D**	指令运行时间量测开始

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



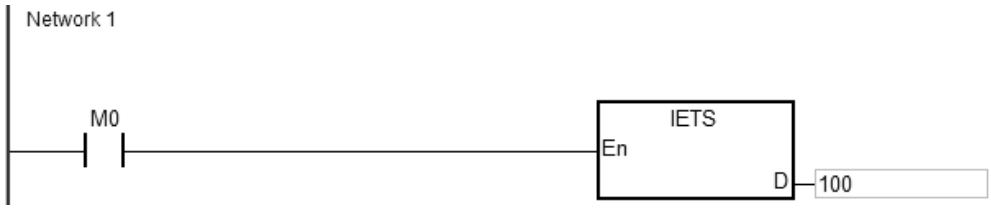
D ： 存放计时结果

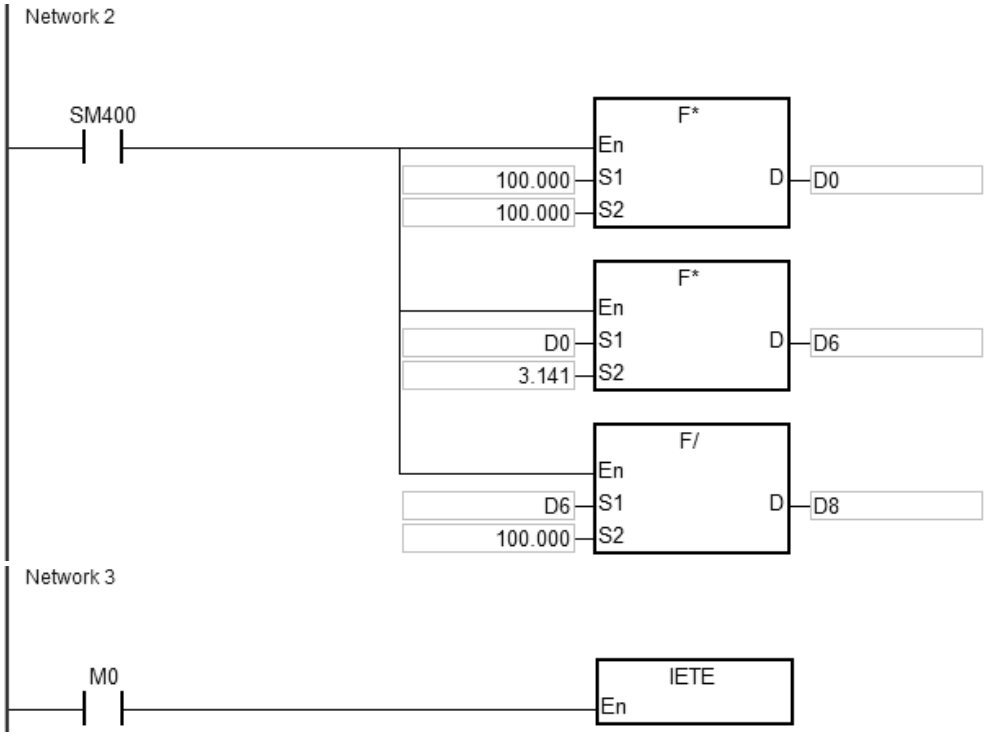
指令说明：

1. 此指令需搭配 API1013 IETE 指令一起使用，目的为量测某一段指定执行的 PLC 程序内的指令所执行的时间。量测的时间单位为 1 微秒（us）。
2. 当 IETS 指令启动后，计时动作立即开始执行，直到遇到 IETE 指令也被执行时，其计时结果将存放于 **D**。
3. 计时结果最小为 0us，最大为 32767us（不会超出此数值），若 IETS 启动后，PLC 扫描至 END 指令时，一直未扫描到 IETE 指令，则 PLC 将自动结束计时，并存放计时结果于 **D** 内。
4. IETS 与 IETE 指令无限制编写次数，但限制每次扫描执行时，最多只能一组被启动，若重复启动 IETS 开始计时，则以最后一个启动 IETS 为计时开始。相反，若多个 IETE 指令都被执行结束计时，则 PLC 以最先关闭计时的地方，做为计时结束点。
5. 此计时量测指令通常被使用于量测某一段 PLC 程序区的运行时间，如：中断服务程序、功能块等。由于此量测功能开启时，会占用 PLC 计时资源，因此建议用户，在量测完成后得要移除这两个指令，避免正常执行过程中占用 PLC 资源。

程序范例：

针对浮点数运算的公式计算指令运行时间，并将计时结果存放于 D100。




Network 2
SM400
F*
En
100.000
S1
D
D0
100.000
S2
F*
En
D0
S1
D
D6
3.141
S2
F/
En
D6
S1
D
D8
100.000
S2
Network 3
M0
IETE
En

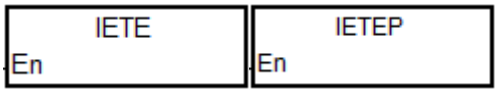
API	指令码			操作数	功能
1013		IETE	P	—	指令运行时间量测结束

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



指令说明：

此指令需搭配 API1012 IETS 指令一起使用，详细说明请参考 API 1012 说明。

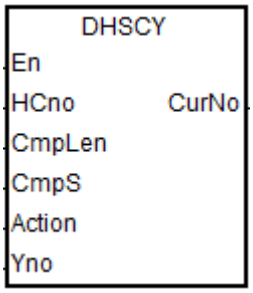
API	指令码			操作数	功能
1014	D	HSCY		**HCno ~ CurNo**	循环式计数搭配比较表格输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
HCno							○									
CmpLen								●					○	○		
CmpS							●	●	●							
Action								●	●							
Yno								●	●							
CurNo								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
HCno												●	
CmpLen		●				●							
CmpS			●				●						
Action		●				●							
Yno		●				●							
CurNo		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



HCno ：32 位输入计数器编号

CmpLen ：比较表格组数长度

CmpS ：比较表的 32 位比较值存放来源

Action ：比较表的比较到达指定动作代码

Yno ：比较表的比较到达 Y 输出点编号

CurNo ：当前比较器已执行的组数编号

指令说明：

1. V1.06.10 版（含）以上支持此指令。
2. **HCno** 限制选用 32 位 HC 计数器编号，当选择计数器为一般型计数器，则比较到达的时机，将会是扫描到指令时，才会进行比对；当选择计数器为高速计数器编号（请参考 DCNT 指令说明），则比较到达时机与控制输出动作，将不会受扫描周期影响。但请注意高速比较的组数限制，请参考 DHSCS 指令说明。
3. **CmpLen** 为选择比较表格的组数，其比较动作将依照设定组数进行循环式的比较功能。此组数长度可设定数值范围为 4~32 之间，当设定数值超过范围时，指令将不会执行，并且设定 SM0 为 On，错误码 SR0 为 0x200B。
4. **CmpS**（假设指定 D100），**Action**（假设指定 D200），**Yno**（假设指定 D300），**CurNo** 操作数说明请参考下表，此表假设 **CmpLen** 数值为 6。

CurNo 组数编号	**CmpS** 来源比较值 (32 位数值)	**Action** 比较到达指定动作	**Yno** 输出点编号
1	D100 = 1000	D200 = 1（Set）	D300 = 16#04 （Y0.4）
2	D102 = 2000	D201 = 1（Set）	D301 = 16#05 （Y0.5）
3	D104 = 3000	D202 = 1（Set）	D302 = 16#06 （Y0.6）
4	D106 = 4500	D203 = 0（Reset）	D303 = 16#04 （Y0.4）
5	D108 = 5500	D204 = 0（Reset）	D304 = 16#05 （Y0.5）
6	D110 = 6500	D205 = 0（Reset）	D305 = 16#06 （Y0.6）

- **CmpS** 为来源比较值，其数值为 32 位整数值，会连续占用 **CmpLen** 个 Dword 空间。建议可采用数组 Dword 类型宣告。比较值要依递增或递减方式排序，升序时，比较值限填正数；递减排序时，比较值限填负整数。如果比较值不依上述规则，指令将不会执行，并且设定 SM0 为 On，错误码 SR0 为 0x200B。
- **Action** 为比较到达时指定的动作代码，其数值为 16 位整数值，会连续占用 **CmpLen** 个 Word 空间；数值 1 表示输出点设定 On，数值 0 表示输出点清除为 Off，数值 16#8000 表示清除计数器数值为 0，重新循环计数；其他数值为保留使用，请勿选用。建议可采用数组 Word 类型宣告。
- **Yno** 为指定输出点的编号，其数值为 16 位整数值，会连续占用 **CmpLen** 个 Word 空间；由于 Y 输出点编号为 16 进制，建议直接采用 16 进制数值设定。举例：16#0004 表示 Y0.4 输出点，16#0016 表示 Y1.6 输出点，16#00AF 表示 Y10.15。建议可采用数组 Word 类型宣告。
- **CurNo** 为显示目前已经比较到达的组数编号，举例：当计数值为 200，则对应比较表内比第一组比较值低，因此 **CurNo** 数值为 0；当计数值到达 1000，则对应比较表等于第一组比较值，因此 **CurNo** 数值变为 1。

5. 当 **Yno** 输出点采用非主机内建的输出点时，其输出点实际输出动作就会受扫描周期所影响。若需要实时控制输出的应用，建议采用主机内建输出点以及 **HCno** 采用硬件型高速计数器。
6. 由于此指令采用循环式比较与输出，因此使用此功能前，必须规划计数器值清除为 0 的机制。建议可采用硬件输入清除方式，或者利用一组比较数值做比较到达清除计数器（注：**CurNo** 也会清除为 0）。举例：上述比较表长度 **CmpLen** 可改为 7 组，然后在第 7 组填入 **Action** 为 16#8000 数值，如下表所示：

组数编号	**CmpS** 来源比较值 (32 位数值)	**Action** 比较到达指定动作	**Yno** 输出点编号
7（**CurNo**=0）	D112 = 8000	D206 = 16#8000 （Clear HC）	D306 = 0 （无作用）

7. 当指令初次启动时，此指令会先将此刻计数器的数值，开始对用户设定的比较表内数值进行比较，自动将比对状况调整至适当的位置。举例：参数设定请参考上述 6 组比较表内容，假设指令启动时，计数值为 2000，则指令被初次启动时，会将 **CurNo** 数值设为 2，并且将 Y0.5 输出设为 On。
8. 此指令的循环比较控制时序图与说明，如下所示：（参数设定请参考上述 6 组比较表内容）

   ① → 指令初次启动，依照 HC202 内的计数值进行比对，并设定 **CurNo** = 1 与 Y0.4 输出为 On。

   ② → 计数值到达 2000 位置，**CurNo** = 2 与 Y0.5 设定为 On。

③ → 计数值到达 3000 位置，**CurNo** = 3 与 Y0.6 设定为 On。

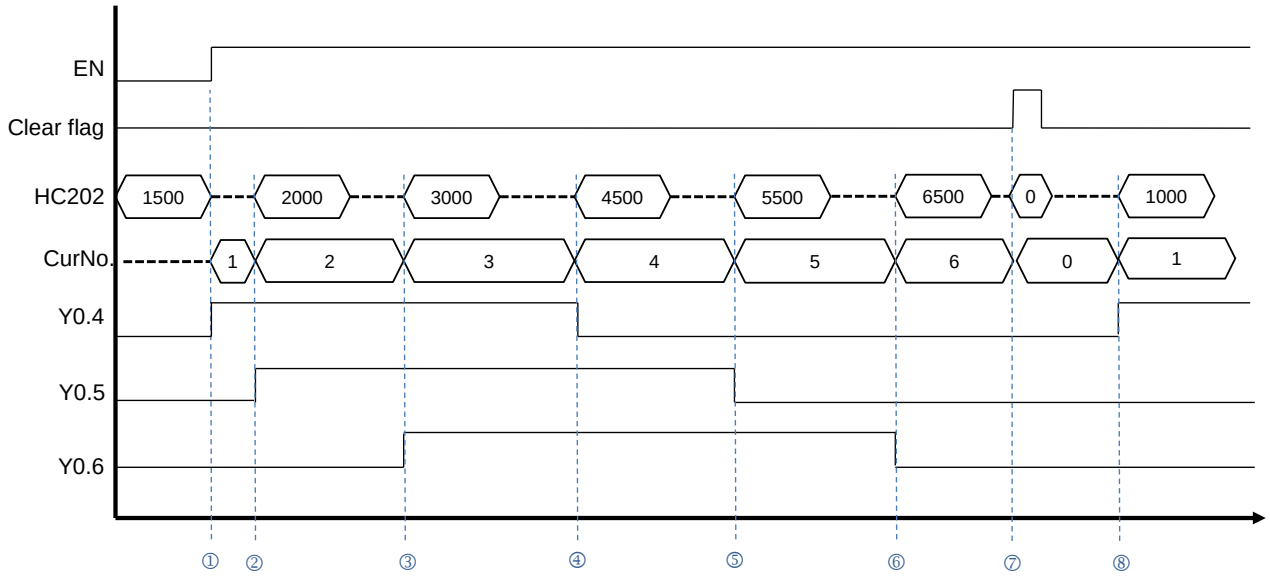
④ → 计数值到达 4500 位置，**CurNo** = 4 与 Y0.4 清除为 Off。

⑤ → 计数值到达 5500 位置，**CurNo** = 5 与 Y0.5 清除为 Off。

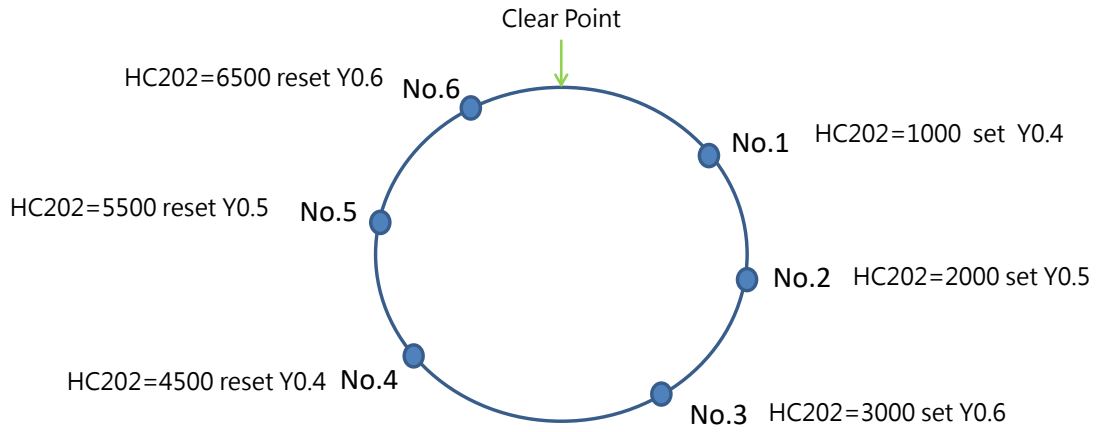
⑥ → 计数值到达 6500 位置，**CurNo** = 6 与 Y0.6 清除为 Off。

⑦ → 外部输入清除讯号触发，**CurNo** = 0 与 HC202 = 0。

⑧ → 计数值到达 1000 位置，**CurNo** = 1 与 Y0.4 设定为 On。



循环式比较示意图如下所示：



程序范例：

1. 用户须先给定下列参数：

CurNo 组数编号	**CmpS** 来源比较值 (32 位数值)	**Action** 比较到达指定动作	**Yno** 输出点编号
1	D100 = 1000	D200 = 1 (Set)	D300 = 16#04 (Y0.4)
2	D102 = 2000	D201 = 1 (Set)	D301 = 16#05 (Y0.5)
3	D104 = 3000	D202 = 1 (Set)	D302 = 16#06

CurNo 组数编号	**CmpS** 来源比较值 (32 位数值)	**Action** 比较到达指定动作	**Yno** 输出点编号
			(Y0.6)
4	D106 = 4500	D203 = 0 (Reset)	D303 = 16#04 (Y0.4)
5	D108 = 5500	D204 = 0 (Reset)	D304 = 16#05 (Y0.5)
6	D110 = 6500	D205 = 0 (Reset)	D305 = 16#06 (Y0.6)

2. 当 M0 为 ON，启动循环比较控制 Y 输出动作流程如下：

① → 指令初次启动，依照 HC202 内的计数值进行比对，并设定 **CurNo** = 1 与 Y0.4 输出为 On。

② → 计数值到达 2000 位置，**CurNo** = 2 与 Y0.5 设定为 On。

③ → 计数值到达 3000 位置，**CurNo** = 3 与 Y0.6 设定为 On。

④ → 计数值到达 4500 位置，**CurNo** = 4 与 Y0.4 清除为 Off。

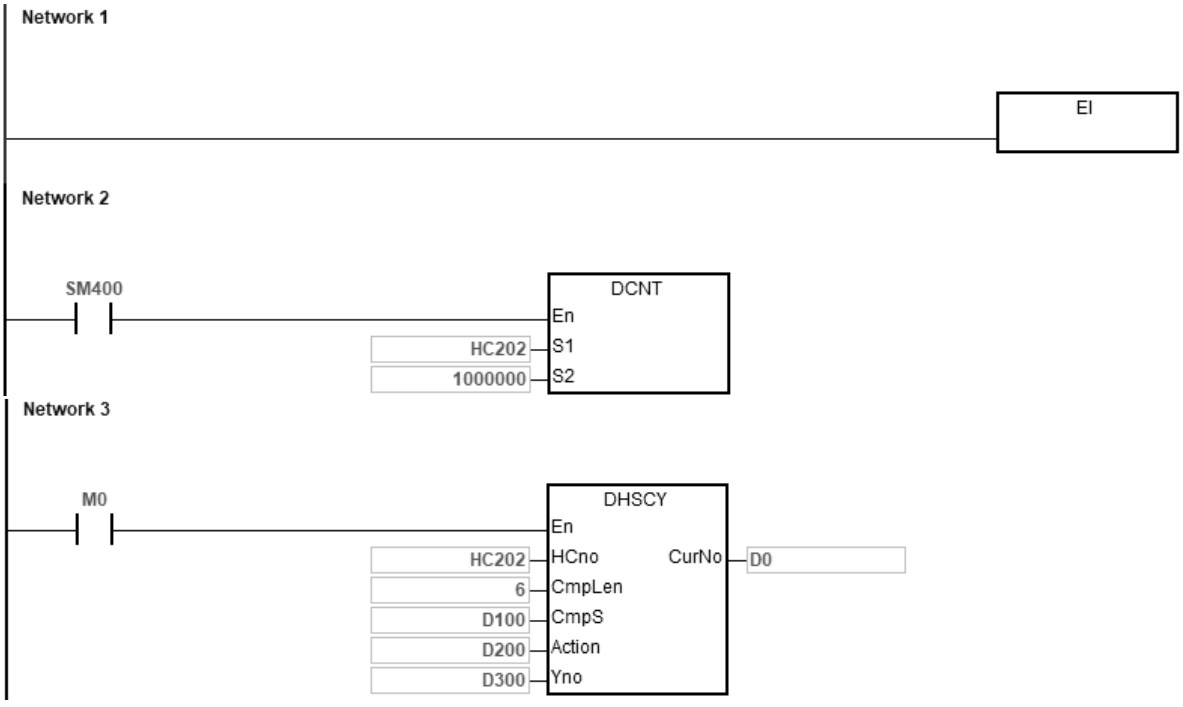
⑤ → 计数值到达 5500 位置，**CurNo** = 5 与 Y0.5 清除为 Off。

⑥ → 计数值到达 6500 位置，**CurNo** = 6 与 Y0.6 清除为 Off。

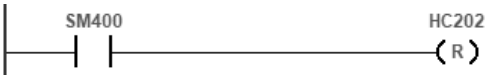
⑦ → 外部输入清除讯号触发，**CurNo** = 0 与 HC202 = 0。

⑧ → 计数值到达 1000 位置，**CurNo** = 1 与 Y0.4 设定为 On。

程序:



外部中断 X0.4



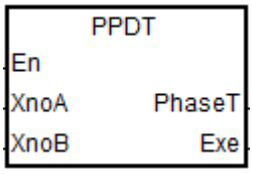
API	指令码		操作数	功能
1015		PPDT	**XnoA**，**XnoB**，**PhaseT**，**Exe**	两相输入之相位差时间检测

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
XnoA	○															
XnoB	○															
PhaseT								●								
Exe		●	●	●												

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
XnoA	●												
XnoB	●												
PhaseT			●				●						
Exe	●												

脉冲执行型	16 位指令	32 位指令
-	AS	

符号：



XnoA ： A 相输入

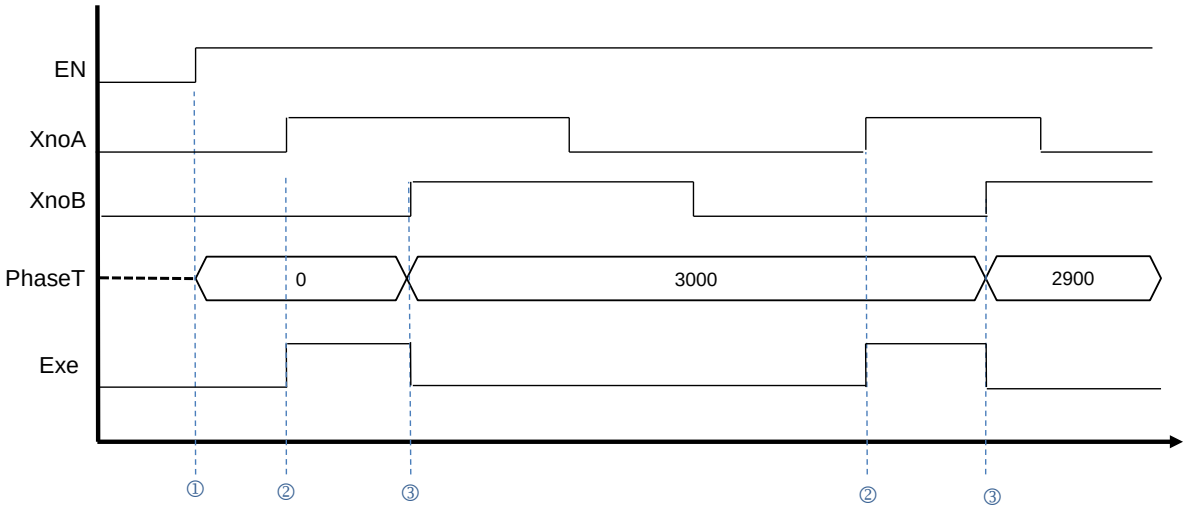
XnoB ： B 相输入

PhaseT ： 两相输入之相位差时间（32 位整数值）

Exe ： 计时中标志

指令说明：

1. V1.06.10 版（含）以上支持此指令。
2. **XnoA** 与 **XnoB** 限制选用主机内建的 X0.0~X0.11 之输入点，并且两个输入点编号不可使用相同编号。当使用非指定的输入点或是两个输入点使用相同编号，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. 此指令的相位差时间量测方式为 **XnoA** 上升沿触发之后，开始启动定时器(此启动动作不受指令扫描影响)，直到 **XnoB** 上升沿触发产生之后，停止计时（此停止动作不受指令扫描影响），并将已经计时的时间（基本单位为 1us）存入 **PhaseT**（此存入动作在指令被扫描到时执行）。
4. 当定时器启动，并且扫描到该启动指令时，**Exe** 标志将会被设为 On；当定时器停止，并且扫描到该启动指令时，**Exe** 标志将会被清除为 Off。
5. 此指令每次启动时，其初始动作固定为清除 **PhaseT** 为 0，以及清除 **Exe** 标志为 Off。
6. 此指令永远计时 **XnoA** 领先 **XnoB** 的时间，若需要量测 **XnoB** 领先 **XnoA**，则请自行变换 **XnoA** 与 **XnoB** 的编号。
7. 由于此指令无使用次数限制，也无重复使用输入点编号限制，但 PLC 是采用 **XnoA** 输入编号做为记录内部定时器的编号，因此不建议同时启动两个 **XnoA** 输入编号相同的指令，否则计时值将会有相互影响的状况。
8. 此指令的计时时序图如下图所示：



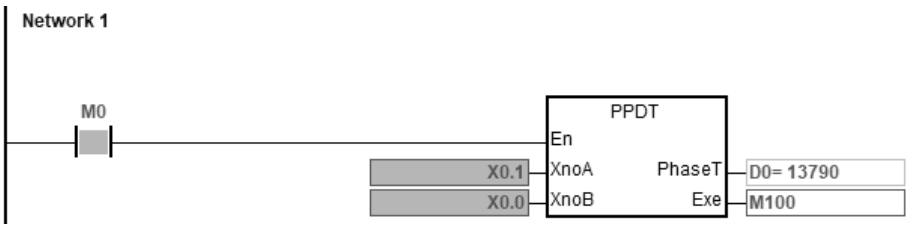
① → 指令初次启动，**PhaseT** 存放数值清除为 0，**Exe** 标志清除为 Off。

② → **XnoA** 输入触发启动计时，**Exe** 标志设为 On。

③ → **XnoB** 输入触发停止计时，**Exe** 标志清除为 Off，并存入计时数值至 **PhaseT** 内。

程序范例：

当 M0 为 ON，PPDT 会计算 X0.1 与 X0.0 的相位差时间，并将计时结果存放于 D0（检测时间为 13790 us）。



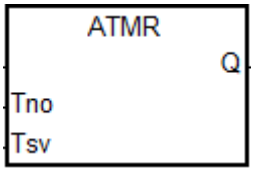
API	指令码	操作数	功能
1016	ATMR	**Tno · Tsv**	接点型定时器（100ms）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Tno					○											
Tsv								○				○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Tno											●		
Tsv		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Tno ： 定时器编号

Tsv ： 定时器设定值

指令说明：

1. V1.12.00 版（含）以上支持此指令。
2. 此指令以 100ms 为基本单位计时，其详细内容说明请参考 API1017 ATMRH 指令的指令说明。

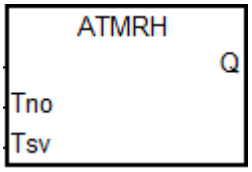
API	指令码	操作数	功能
1017	ATMRH	**Tno，Tsv**	接点型定时器（1ms）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Tno					○											
Tsv								○				○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Tno											●		
Tsv		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Tno ： 定时器编号

Tsv ： 定时器设定值

指令说明：

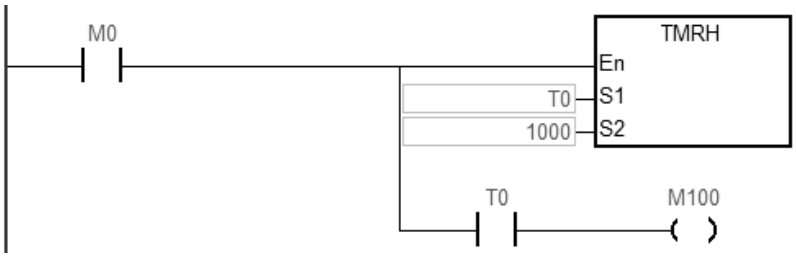
1. V1.12.00 版（含）以上支持此指令。
2. ATMRH 指令相当于 AND + TMRH 指令组合，前面条件接点成立时，指定的定时器线圈受电，定时器开始计时，到达所指定的定时值（计时值 >= 设定值）时，其接点动作如下，同时，AND 接点动作成立；前面条件接点不成立时，则 ATMRH 自动清除计时值。

NO（Normally Open）接点	连续性
NC（Normally Closed）接点	不导通

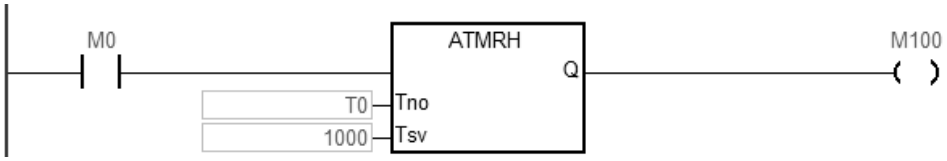
程序范例：

设计程序执行以下动作：当 M0 常开接点 ON 时，T0 定时器开始计时，当计时值大于等于 K1000，M100 常开接点导通。

梯形图使用（TMRH 指令）



梯形图使用（ATMRH 指令）



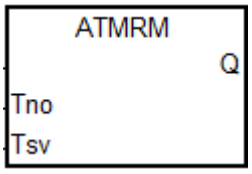
API	指令码	操作数	功能
1018	ATMRM	**Tno，Tsv**	接点型定时器（10ms）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Tno					○											
Tsv								○				○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Tno											●		
Tsv		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Tno ： 定时器编号

Tsv ： 定时器设定值

指令说明：

1. V1.12.00 版（含）以上支持此指令。
2. 此指令以 10ms 为基本单位计时，其详细内容说明请参考 API1017 ATMRH 指令的指令说明。

6.12 移位指令

6.12.1 移位指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1100	SFTR	–	✓	位右移
1101	SFTL	–	✓	位左移
1102	WSFR	DWSFR	✓	寄存器右移
1103	WSFL	DWSFL	✓	寄存器左移
1104	SFWR	DSFWR	✓	位移写入
1105	SFRD	DSFRD	✓	位移读出
1106	SFPO	DSFPO	✓	读出数据串行最新数据
1107	SFDEL	DSFDEL	✓	删除数据串行中的数据
1108	SFINS	DSFINS	✓	插入数据到数据串行中
1109	MBS	–	✓	矩阵位位移
1110	SFR	DSFR	✓	16 位寄存器位右移
1111	SFL	DSFL	✓	16 位寄存器位左移
1112	BSFR	–	✓	n 个位右移 1 个位
1113	BSFL	–	✓	n 个位左移 1 个位
1114	NSFR	DNSFR	✓	n 个寄存器右移
1115	NSFL	DNSFL	✓	n 个寄存器左移

6.12.2 移位指令说明

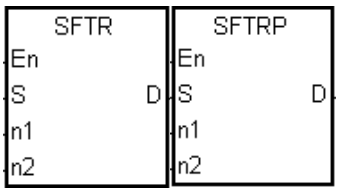
API	指令码			操作数	功能
1100		SFTR	P	**S**、**D**、n_1、n_2	位右移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●	●	●				●								
D		●	●	●				●								
n_1								●	●		○	○	○	○		
n_2								●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
D	●												
n_1		●				●							
n_2		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 移位装置来源之起始编号

D ： 移位装置目的之起始编号

n_1 ： 移位之数据长度

n_2 ： 一次移位之位数

指令说明：

1. 将 **D** 开始之起始编号，具有 n_1 个数字元（位移寄存器长度）的位装置，以 n_2 位个数来右移。而 **S** 开始起始编号以 n_2 位个数移入 **D** 中来填补位空位。
2. 本指令一般都是使用脉冲执行型指令（SFTRP）。
3. n_1 值的范围为 1~1024，n_2 值的范围为 1~n_1。V1.12.00（含）以上支持范围为 1~4096。

程序范例：

1. 当 X0.0 为上升沿时，由 M0~M15 组成 16 位，以 4 位作右移。
2. 扫描一次的位右移动作依照下列编号 1~5 动作。

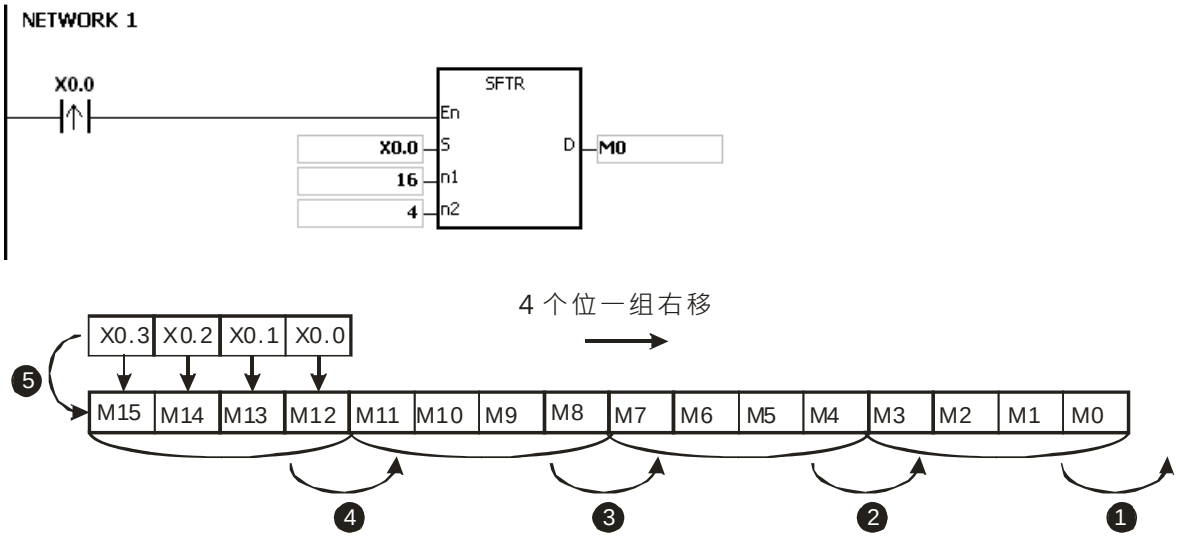
❶ M3~M0 → 进位

❷ M7~M4 → M3~M0

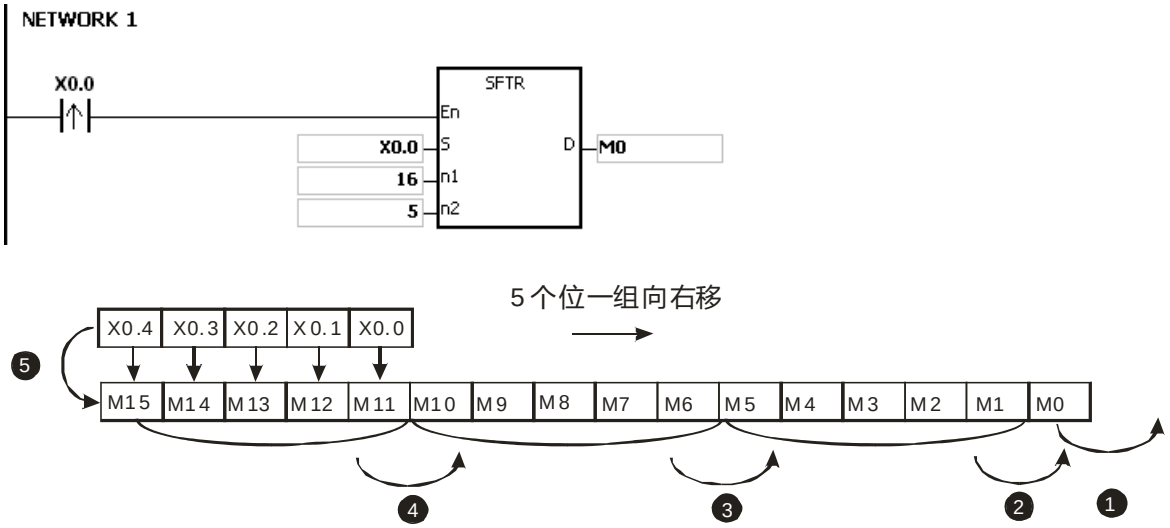
❸ M11~M8 → M7~M4

❹ M15~M12 → M11~M8

❺ X0.3~X0.0 → M15~M12 完成



1. 在 X0.0 上升沿时，由 M0~M15 组成 16 位，以 5 位作右移。
2. 扫描一次的位右移动作依照下列编号 1~5 动作。
   - ❶ M0 → 进位
   - ❷ M5 → M0
   - ❸ M10~M6 → M5~M1
   - ❹ M15~M11 → M10~M6
   - ❺ X0.4~X0.0 → M15~M11 完成



补充说明：

1. **S**+n_2-1、**D**+n_1-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. n_1<1 或 n_1>1024 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. n_2<1 或 n_2>n_1 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

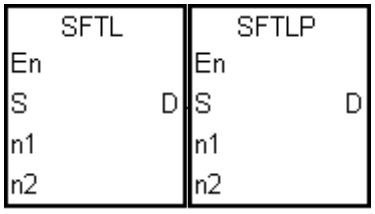
API	指令码			操作数	功能
1101		SFTL	P	**S**・**D**・n_1・n_2	位左移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●	●	●				●								
D		●	●	●				●								
n_1								●	●		○	○	○	○		
n_2								●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
D	●												
n_1		●				●							
n_2		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 移位装置来源之起始编号

D ： 移位装置目的之起始编号

n_1 ： 移位之数据长度

n_2 ： 一次移位之位数

指令说明：

1. 将 **D** 开始之起始编号，具有 n_1 个数字元（位移寄存器长度）的位装置，以 n_2 位个数来左移。而 **S** 开始起始编号以 n_2 位个数移入 **D** 中来填补位空位。
2. 本指令一般都是使用脉冲执行型指令（SFTLP）。
3. n_1 值的范围为 1~1024，n_2 值的范围为 1~n_1。V1.12.00（含）以上支持范围为 1~4096。

程序范例：

1. 在 X0.0 上升沿时，由 M0~M15 组成 16 位，以 4 位作左移。
2. 扫描一次的位左移动作依照下列编号 1~5 动作。

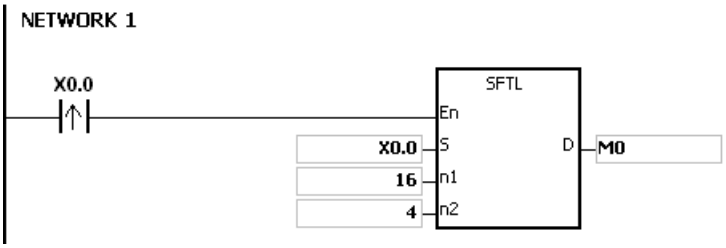
❶ M15~M12 → 进位

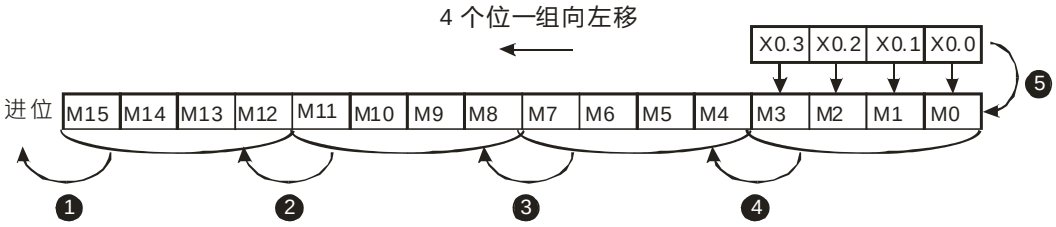
❷ M11~M8 → M15~M12

❸ M7~M4 → M11~M8

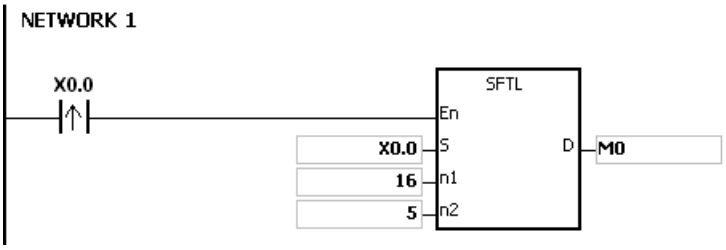
❹ M3~M0 → M7~M4

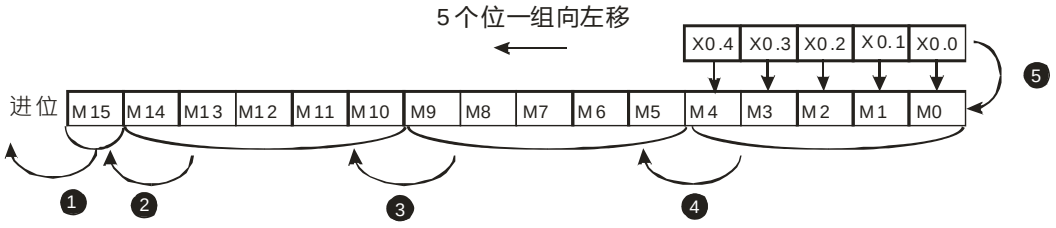
❺ X0.3~X0.0 → M3~M0 完成





3. 在 X0.0 上升沿时，由 M0~M15 组成 16 位，以 5 位作左移。
4. 扫描一次的位左移动作依照下列编号 1~5 动作。
   - ❶ M15 → 进位
   - ❷ M10 → M15
   - ❸ M9~M5 → M14~M10
   - ❹ M4~M0 → M9~M5
   - ❺ X0.4~X0.0 → M4~M0 完成





补充说明：

1. **S**+n_2-1、**D**+n_1-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. n_1<1 或 n_1>1024 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. n_2<1 或 n_2>n_1 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

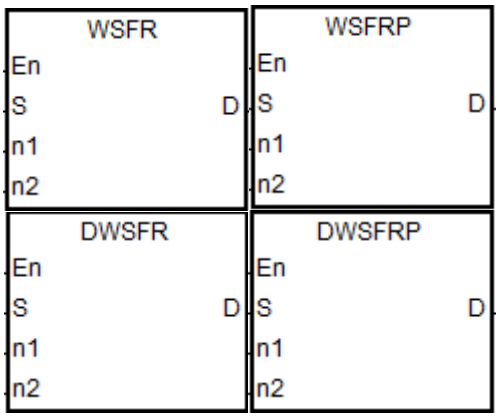
API	指令码			操作数	功能
1102	D	WSFR	P	**S**・**D**・**n_1**・**n_2**	寄存器右移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n_1								●	●		○	○	○	○		
n_2								●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n_1		●	●			●	●						
n_2		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 移位装置来源之起始编号

D ： 移位装置目的之起始编号

n_1 ： 移位之数据长度

n_2 ： 一次移位之位数

指令说明：

1. 将 **D** 开始之起始编号，具有 **n_1** 个字符长度的数据串行，以 **n_2** 个字符来右移。而 **S** 开始起始编号以 **n_2** 字符个数移入 **D** 中来填充字符空位。
2. 本指令一般都是使用脉冲执行型指令（WSFRP）。
3. **n_1** 值的范围，16 位指令为 1~512，32 位指令为 1~256。32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。
4. **n_2** 值的范围为 1~**n_1**。

程序范例 1：

1. M0=OFF→ON 时，由 D20~D35 所组成之 16 个寄存器数据串行为位移区域，以 4 个寄存器来右移。
2. 每扫描一次 M0=OFF→ON 时，其字符右移动作依照下列编号 1~5 动作。

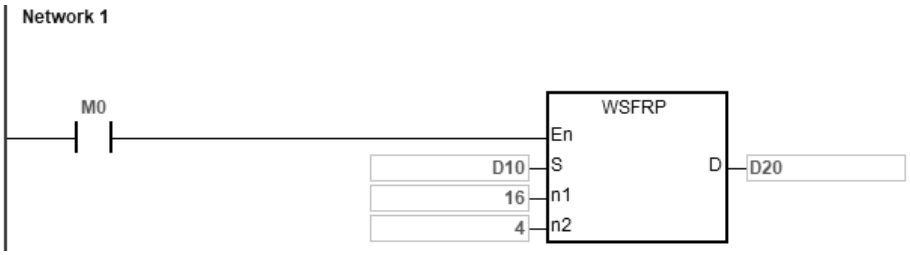
❶ D23~D20 → 进位

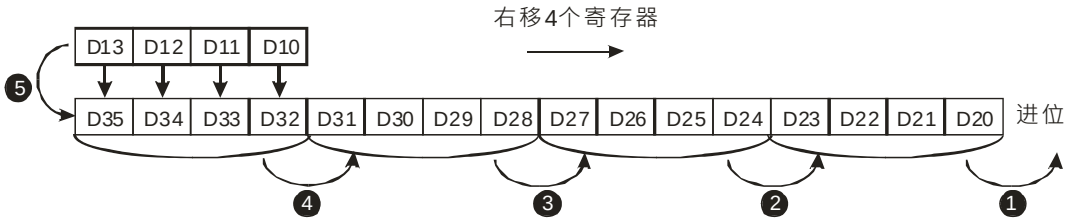
❷ D27~D24 → D23~D20

❸ D31~D28 → D27~D24

❹ D35~D32 → D31~D28

❺ D13~D10 → D35~D32 完成





程序范例 2：

1. M0=OFF→ON 时，由 D20~D35 所组成之 16 个寄存器数据串行为位移区域，以 5 个寄存器来右移。
2. 每扫描一次 M0=OFF→ON 时，其字符右移动作依照下列编号 1~5 动作。

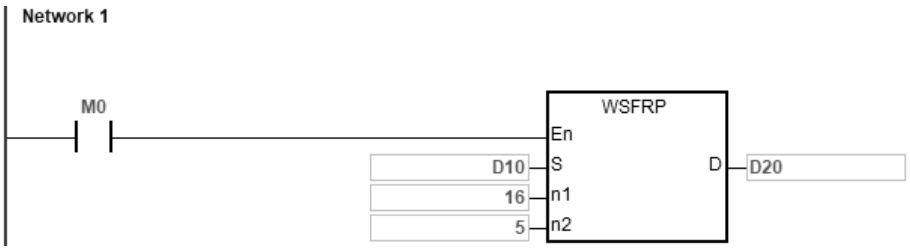
❶ D20 → 进位

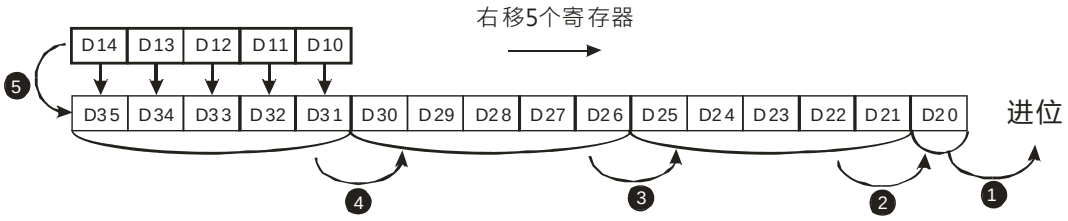
❷ D25 → D20

❸ D30~D26 → D25~D21

❹ D35~D31 → D30~D26

❺ D14~D10 → D35~D31 完成





程序范例 3：

1. M0=OFF→ON 时，由 D20~D35 所组成之 8 笔 32 位寄存器数据串行为位移区域，以 2 笔 32 位寄存器来右移。
2. 每扫描一次 M0=OFF→ON 时，其字符右移动作依照下列编号 1~5 动作。

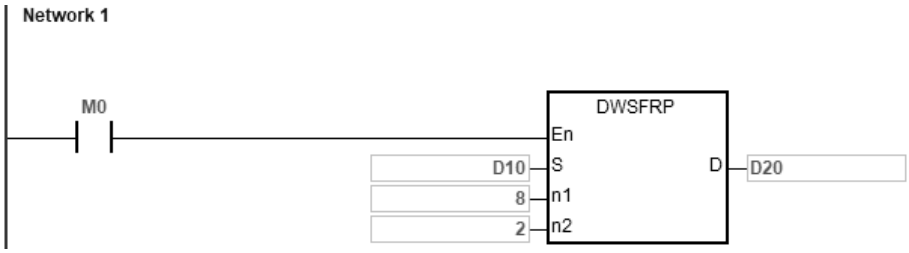
   D23~D20 → 进位

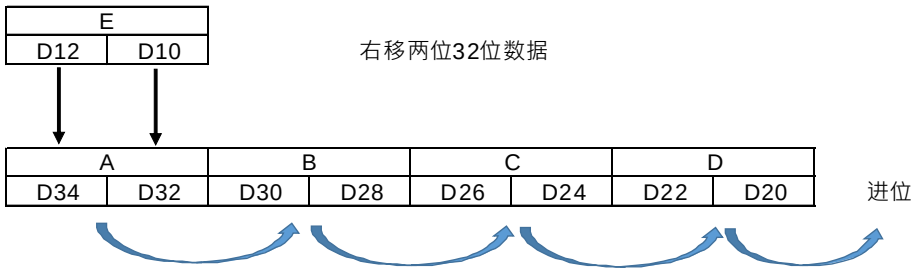
   D27~D24 → D23~D20

   D31~D28 → D27~D24

   D35~D32 → D31~D28

   D13~D10 → D35~D32 完成





补充说明：

1. **S**+**n**$_2$-1、**D**+**n**$_1$-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**$_1$<1 或 **n**$_1$>512(16 位指令)或 **n**$_1$>256(32 位指令)时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. **n**$_2$<1 或 **n**$_2$>**n**$_1$ 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

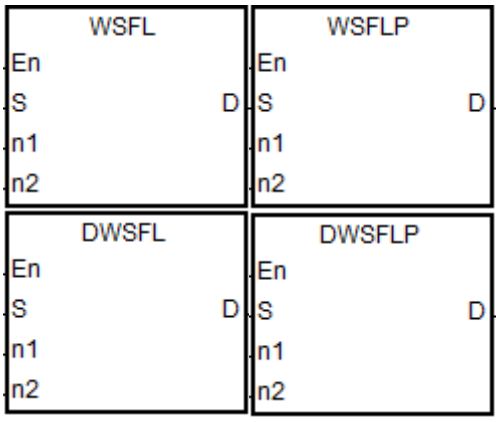
API	指令码			操作数	功能
1103	D	WSFL	P	**S**・**D**・**n_1**・**n_2**	寄存器左移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n_1								●	●		○	○	○	○		
n_2								●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n_1		●	●			●	●						
n_2		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 移位装置来源之起始编号

D ： 移位装置目的之起始编号

n_1 ： 移位之数据长度

n_2 ： 一次移位之位数

指令说明：

1. 将 **D** 开始之起始编号，具有 **n_1** 个字符长度的数据串行，以 **n_2** 个字符来左移。而 **S** 开始起始编号以 **n_2** 字符个数移入 **D** 中来填充字符空位。
2. 本指令一般都是使用脉冲执行型指令（WSFLP）
3. **n_1** 值的范围，16 位指令为 1~512，32 位指令为 1~256。32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。
4. **n_2** 值的范围为 1~**n_1**。

程序范例：

1. 在 M0 上升沿时，由 D20~D35 组成 16 位，以 4 位作左移。
2. 每扫描一次 M0=OFF→ON 时，其位左移动作依照下列编号 1~5 动作。

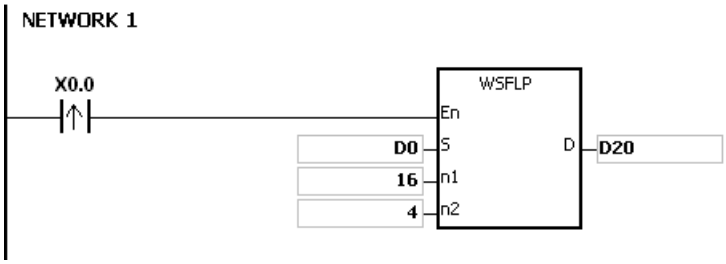
❶ D35~D32 → 进位

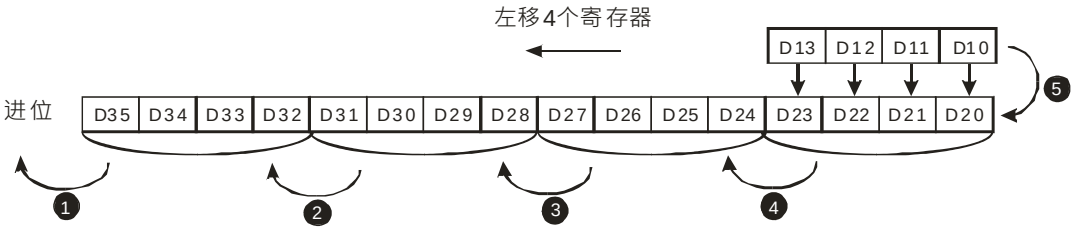
❷ D31~D28 → D35~D32

❸ D27~D24 → D31~D28

❹ D23~D20 → D27~D24

❺ D13~D10 → D23~D20 完成





3. 在 M0 上升沿时，由 D20~D35 组成 16 位，以 5 位作左移。

4. 每扫描一次 M0=OFF→ON 时，其位左移动作依照下列编号 1~5 动作。

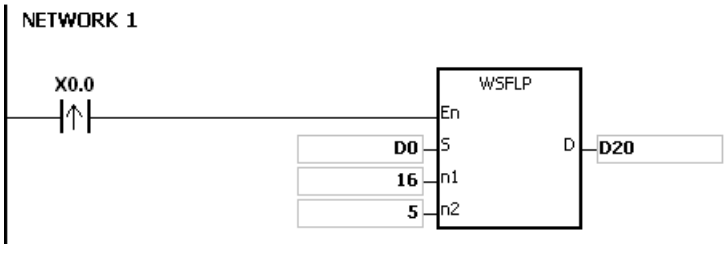
❶ D35 → 进位

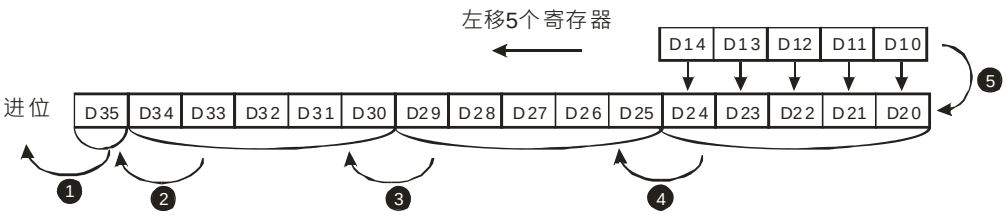
❷ D30 → D35

❸ D29~D25 → D34~D30

❹ D24~D20 → D29~D25

❺ D14~D10 → D24~D20 完成





补充说明：

1. **S**+**n_2**-1、**D**+**n_1**-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n_1**<1 或 **n_1**>512(16 位指令)或 **n_1**>256(32 位指令)时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. n2<1 或 n2>n1 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

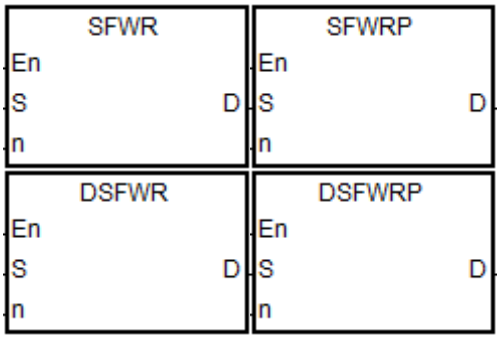
API	指令码			操作数	功能
1104	D	SFWR	P	**S**・**D**・**n**	位移写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 来源位移数据之装置

D ： 目的数据串行之起始编号

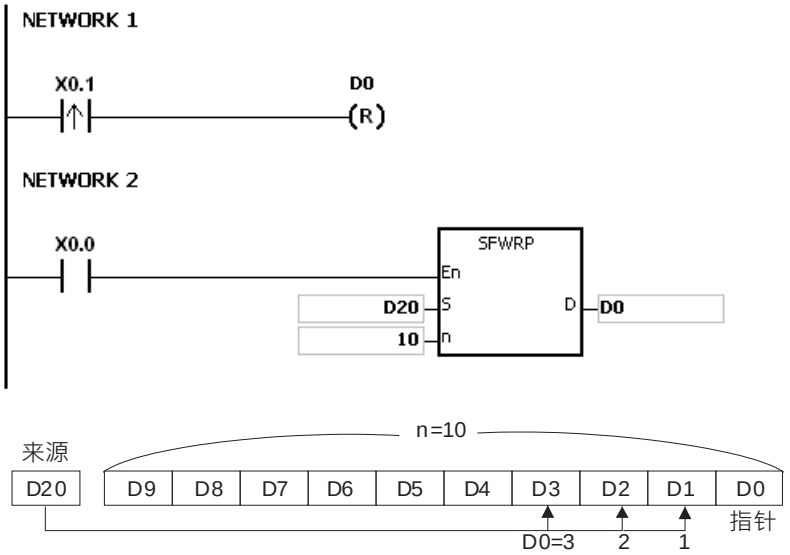
n ： 数据串行之长度

指令说明：

1. 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。当指针的内容大于等于 n-1 时，本指令不再处理写入的新值，进位标志信号 SM602=ON。
2. 本指令一般都是使用脉冲执行型指令（SFWRP）。
3. 此指令常被使用于堆栈功能中的推入（PUSH）功用。若需搭配推出（POP）功用请参考 SFRD 与 SFPO 指令。
4. **n** 值的范围，16 位指令为 2~512，32 位指令为 2~256。32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。

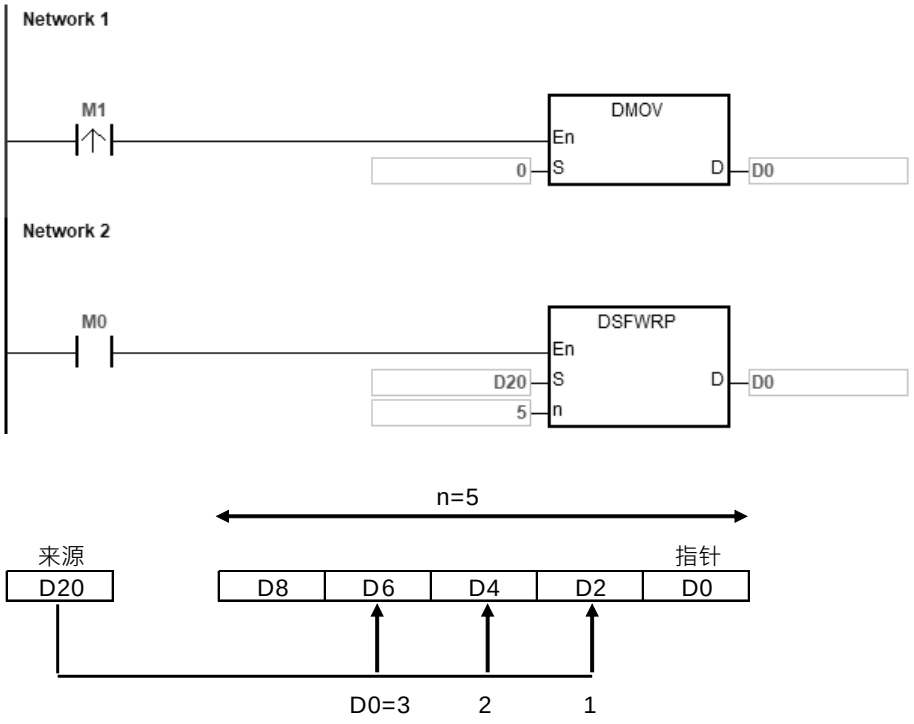
程序范例 1：

1. 开始先将指针 D0 清除为 0，当 M0 = 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 M0 再 OFF→ON 一次，则 D20 的内容被传送至 D2 当中，D0 内容变成 2。
2. 指令执行一次位移写入动作依照下列动作。
   - D20 的内容被传送至 D1 当中。
   - 指针 D0 内容变成 1。



程序范例 2：

1. 开始先将指针 D0/D1 清除为 0，当 M0 = OFF→ON 变化时，D20/D21 的内容被传送至 D2/D3 当中，指针 D0/D1 内容变成 1 变更 D20/D21 的内容后 将 M0 再 OFF→ON 一次 则 D20/D21 的内容被传送至 D4/D5 当中，D0/D1 内容变成 2。
2. 指令执行一次位移写入动作依照下列动作。
   - D20/D21 的内容被传送至 D2/D3 当中。
   - 指针 D0/D1 内容变成 1。



补充说明：

1. **D** 的内容值<0，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **D**+**n**-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **n**<2 或 **n**>512（16 位指令）或 **n**>256（32 位指令），指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 本指令 API1104 SFWR 与 API1105 SFRD 可搭配使用，执行先入先出数据串行的写入读出控制。

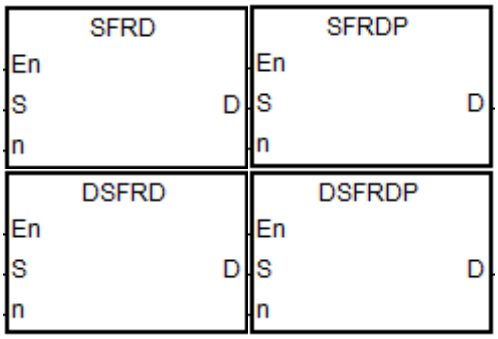
API	指令码			操作数	功能
1105	D	SFRD	P	**S**、**D**、**n**	位移读出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S		●			●	●		●			○					
D		●			●	●		●								
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 来源位移数据之装置

D ： 目的数据串行之起始编号

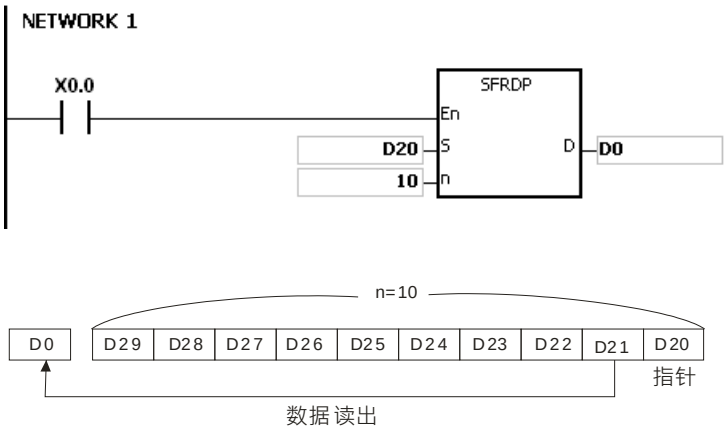
n ： 数据串行之长度

指令说明：

1. 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，**S** 内容值先减 1，之后(**S+1**)所指定的装置其内容值会写入 **D** 所指定的位置，(**S+n-1**)~(**S+2**) 全部右移一个寄存器，(**S+n-1**) 的内容不变，当 **S** 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 SM600=ON。
2. 本指令一般都是使用脉冲执行型指令（SFRDP）。
3. **n** 值的范围，16 位指令为 2~512，32 位指令为 2~256。32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。

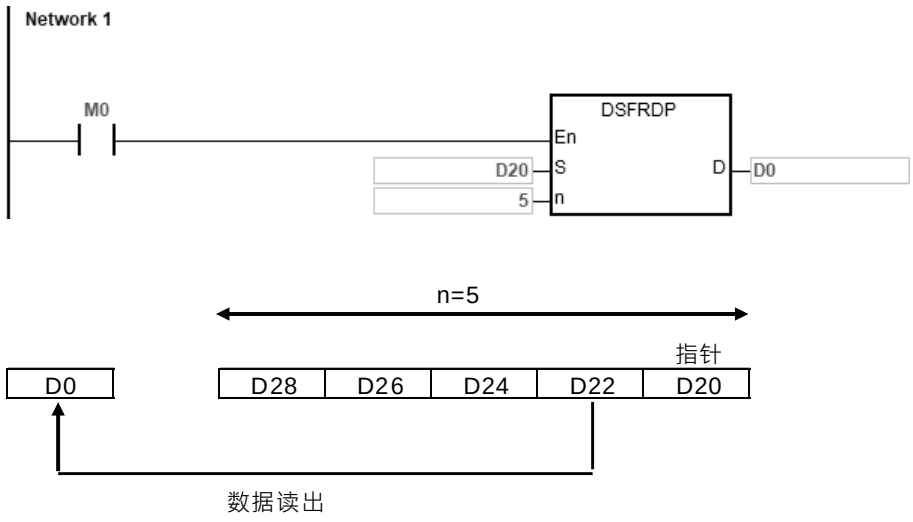
程序范例 1：

1. 给定 D20=10，当 M0 从 OFF→ON 变化时，D21 的内容被传送至 D0 当中，D29~D22 全部往右位移一个寄存器（D29 内容保持不变），D20 内容减 1。
2. 指令执行一次位移读出动作依照下列动作。
   - D21 的内容被读出传送至 D0 当中。
   - D29~D22 全部往右位移一个寄存器。
   - D20 内容减 1。



程序范例 2：

1. 给定 D20/D21=5，当 M0 从 OFF→ON 变化时，D22/D23 的内容被传送至 D0/D1 当中，D29~D22 全部往右位移一笔 32 位数据（D28/D29 内容保持不变），D20/D21 内容减 1。
2. 指令执行一次位移读出动作依照下列动作。
   - D22/23 的内容被读出传送至 D0/D1 当中。
   - D29~D22 全部往右位移一笔 32 位数据。
   - D20 内容减 1。



补充说明：

1. **S**<0，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **S**+**n**-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **n**<2 或 **n**>512（16 位指令）或 **n**>256（32 位指令），指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 本指令 API1104 SFWR 与 API1105 SFRD 可搭配使用，执行先入先出数据串行的写入读出控制。

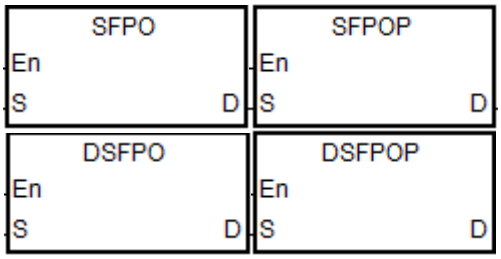
API	指令码			操作数	功能
1106	D	SFPO	P	**S**・**D**	读出数据串行最新数据

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S		●			●	●		●								
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 源数据串行之起始编号

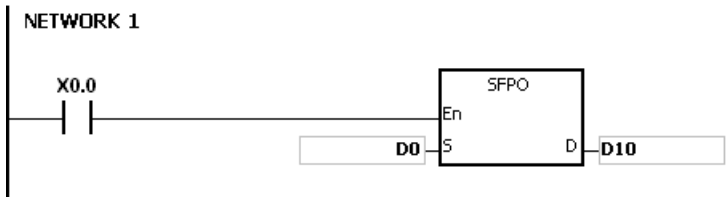
D ： 目的读出数据之装置

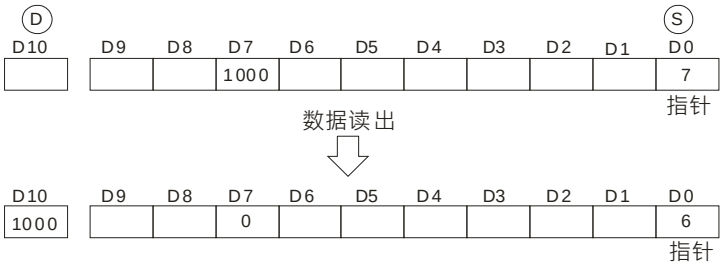
指令说明：

1. 将 **S** 起始编号开始的数据串行，以第一个编号装置作为指针，当指令执行时，**S** 内容值所指定的指针内容值会写入 **D** 所指定的位置，**S** 内容值所指定的指针内容值被清除为 0，**S** 内容值减一，当 **S** 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 SM600=ON。
2. 本指令一般都是使用脉冲执行型指令（SFPOP）。
3. 此指令一般常被应用于推迭功能中的推出（POP）功用。
4. 32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。

程序范例 1：

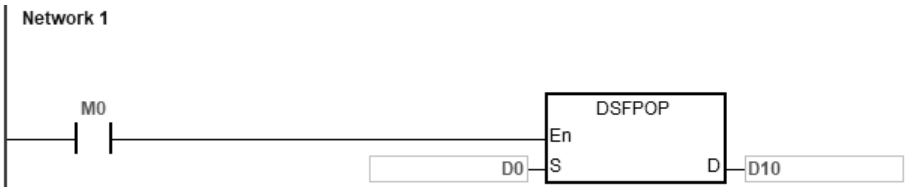
给定 D0=7，当 M0=ON 时，D0 内容所指定的位置内容搬移至 D10 当中，搬移后指定位置清除为 0，D0 内容减一。

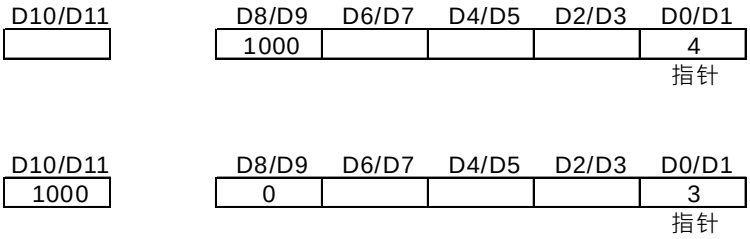




程序范例 2：

给定 D0/D1=4，当 M0=ON 时，D0/D1 内容所指定的位置内容搬移至 D10/D11 当中，搬移后指定位置清除为 0，D0/D1 内容减一。





补充说明：

1. **S** 内容值<0，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **S** 编号+**S** 内容值，装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

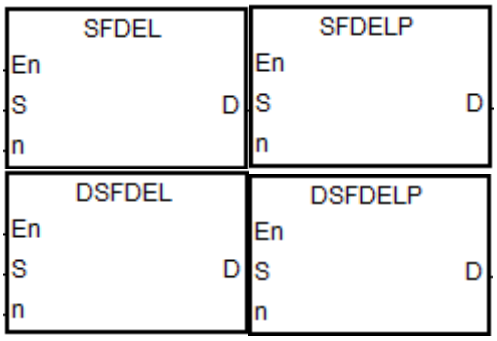
API	指令码			操作数	功能
1107	D	SFDEL	P	**S**、**D**、**n**	删除数据串行中的数据

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S		●			●	●		●								
D		●			●	●		●			○	○				
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ：数据串行之起始编号

D ：储存被删除的数据的装置

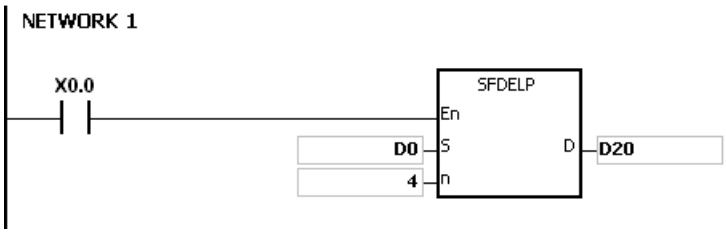
n ：指定删除的位置

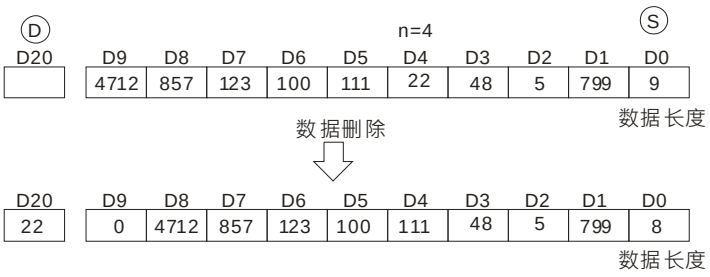
指令说明：

1. **S** 为数据串行的长度，(**S+1**) ~ (**S+S** 内容值) 为数据串行的内容，当指令执行时，**S+n** 的内容值会被储存在 **D** 中，并删除 **S+n** 的内容值，然后(**S+n+1**)~(**S+S** 内容值)的装置内容全部右移一个寄存器，(**S+S** 内容值) 的内容归零，最后将 **S** 内容值减一，当 **S** 的内容等于 0 时，本指令不再处理数据删除的动作，零标志信号 SM600=ON。
2. 本指令一般都是使用脉冲执行型指令（SFDELP）。
3. **n** 操作数数值范围，16 位指令为 1~32767，32 位指令为 1~16383。
4. 32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。

程序范例 1：

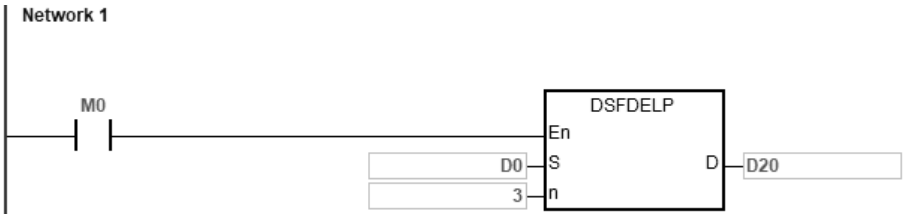
给定 D0=9，当 M0=ON 时，将 n=4 指定的 D4 内容搬移至 D20 当中，并删除 D4 的内容值，然后将 D5~D9 的内容向前递补，最后将 D0 内容减一。

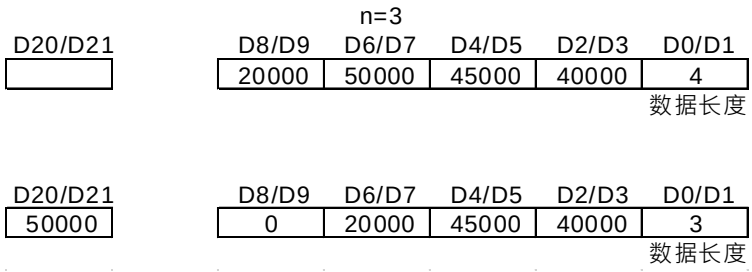




程序范例 2：

给定 D0/D1=4，当 M0=ON 时，将 n=3 指定的 D6/D7 内容搬移至 D20 当中，并删除 D6/D7 的内容值，然后将 D8/D9 的内容向前递补，最后将 D0/D1 内容减一。





补充说明：

1. **S** 内容值<0，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **S**+**n** 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **S**+**S** 内容值，装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. **n**>**S** 内容值，指令不执行，SM0=ON，错误码 SR0=16#200B。
5. **n**<=0 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

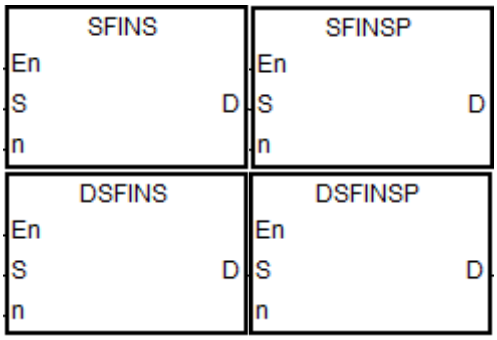
API	指令码			操作数	功能
1108	D	SFINS	P	**S**，**D**，**n**	插入数据到数据串行中

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S		●			●	●		●								
D	●	●			●	●		●	●		○	○	○	○		
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 数据串行之起始编号

D ： 插入数据串行的数据

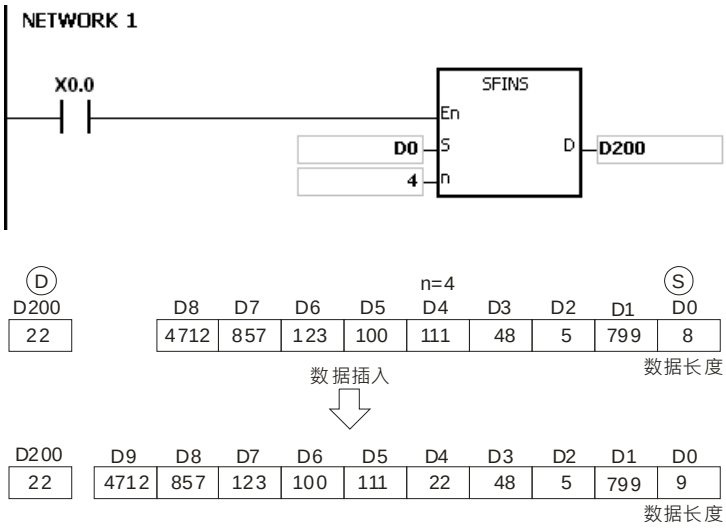
n ： 指定插入的位置

指令说明：

1. **S** 为数据串行的长度，（**S+1**）~（**S+S** 内容值）为数据串行的内容，当指令执行时，**D** 的内容值会被插入 **S+n** 中，原本（**S+n**）~（**S+S** 内容值）的数据全部往下移动一个寄存器，最后将内容值加一，**S** 内容值 =32767 时，本指令不再处理写入的新值，**S** 内容值不再往上增加，进位标志信号 SM602=ON。
2. 本指令一般都是使用脉冲执行型指令（SFINSP）。
3. **n** 操作数数值范围，16 位指令为 1~32767，32 位指令为 1~16383。
4. 32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。

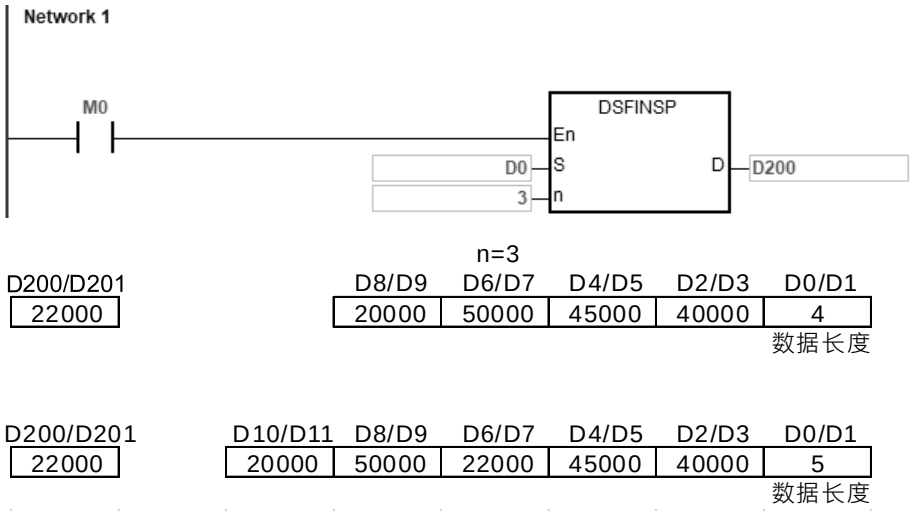
程序范例 1：

给定 D0=8，当 M0=ON 时，先将 D200 的内容值插入 n=4 指定的 D4 中，再将原先 D4~D8 的内容值向下位移填入 D5~D9 中，最后将 D0 内容加一。



程序范例 2：

给定 D0/D1=4，当 M0=ON 时，先将 D200 的内容值插入 n=3 指定的 D6/D7 中，再将原先 D6~D9 的内容值向下位移填入 D8~D11 中，最后将 D0/D1 内容加一。



补充说明：

1. （**S** 内容值）<=0，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **S**+**n** 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **S**+（**S** 内容值）+1，装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. **n**>（**S** 内容值），指令不执行，SM0=ON，错误码 SR0=16#200B。
5. **n**<=0 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

API	指令码			操作数	功能
1109		MBS	P	**S**・**D**・**n**	矩阵位位移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●		○		○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 矩阵来源装置

D ： 运算结果之目的装置

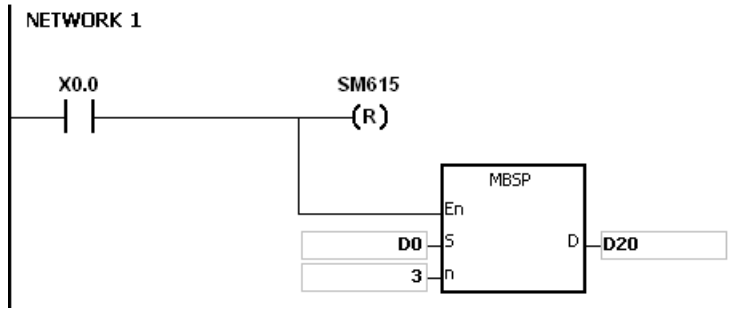
n ： 数组长度

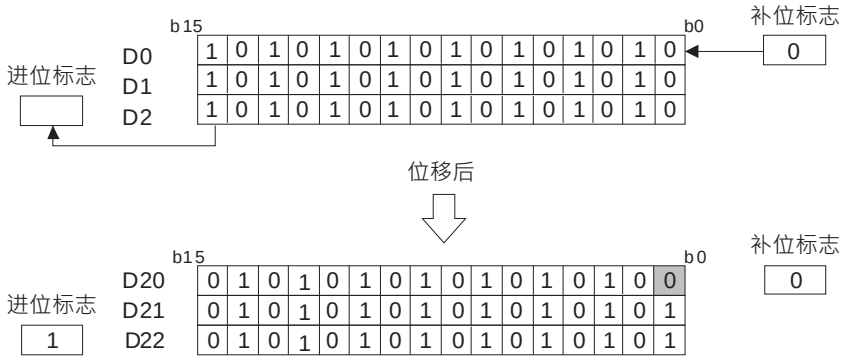
指令说明：

1. 矩阵来源依数组长度 **n** 将 **S** 矩阵位做左右位移控制，SM616=0 决定矩阵位左移，SM616=1 决定矩阵位右移。每次移动一位，因位移而腾出之空位（左移时为 b0，右移时为 b16n–1）则以补位标志 SM615 之状态填补。而因位移而挤出之位（左移时为 b16n–1，右移时为 b0）状态则送到进位标志 SM614 去，然后将结果存入 **D**。
2. n 操作数值的范围为 1~256。
3. 本指令一般都是使用脉冲执行型指令（MBSP）。

程序范例一：

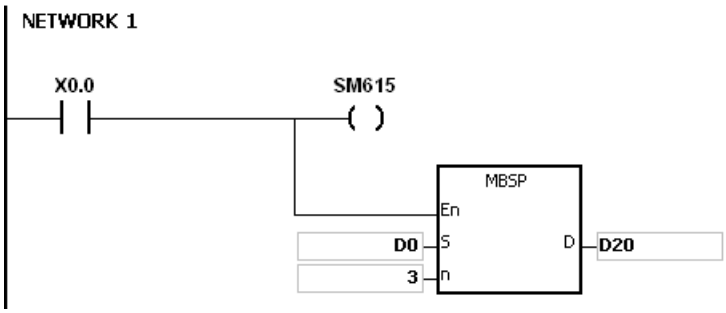
当 X0.0=ON 时，矩阵循环位移方向标志 SM616=OFF 作矩阵左移。设矩阵位移输入补位标志 SM615=0，16 位寄存器 D0~D2 矩阵作左移，将结果存于 16 位寄存器 D20~D22 矩阵中，矩阵循环位移输出进位标志 SM614=1。

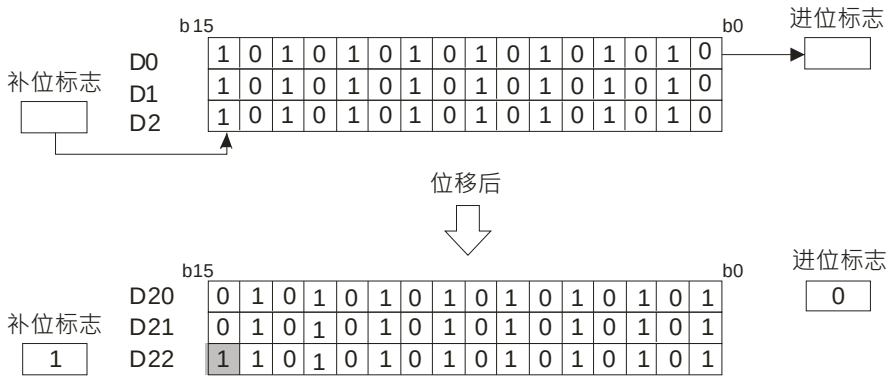




程序范例二：

当 X0.0=ON 时，矩阵循环位移方向标志 SM616=ON 作矩阵右移，设矩阵位移输入补位标志 SM615=1，16 位寄存器 D0~D2 矩阵作右移，将结果存于 16 位寄存器 D20~D22 矩阵中，矩阵循环位移输出进位标志 SM614=0。





补充说明：

1. **S+n-1**、**D+n-1** 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 标志信号说明

   SM614： 矩阵循环位移输出进位标志

   SM615： 矩阵位移输入补位标志

   SM616： 矩阵循环位移方向标志

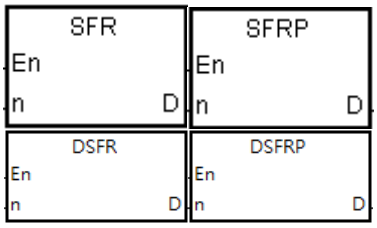
API	指令码			操作数	功能
1110	D	SFR	P	**D**，**n**	寄存器位右移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ： 欲位移之装置

n ： 一次位移之位数

指令说明：

1. 将 **D** 往右移 **n** 个位数，16 位指令因位移而腾出之空位（b_{15}–b_{15-n+1}）则补零；32 位指令因位移而腾出之空位（b_{31}–b_{31-n+1}）则补零。因位移而挤出之第 **n**-1 位（b_{n-1}）状态则送到进位标志 SM602 去。
2. 16 位指令 **n** 内容值的范围为 1~16。32 位指令 n 内容值的范围为 1~32。
3. 本指令一般都是使用脉冲执行型指令（SFRP、DSFRP）。
4. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

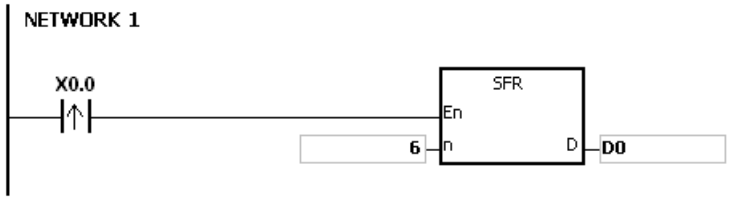
当 X0.0=ON 时，D0 的 b0~b15，往右移动 6 个位数，而 b5 的状态传送到 SM602，位移后，b10~b15 内容清除为 0。

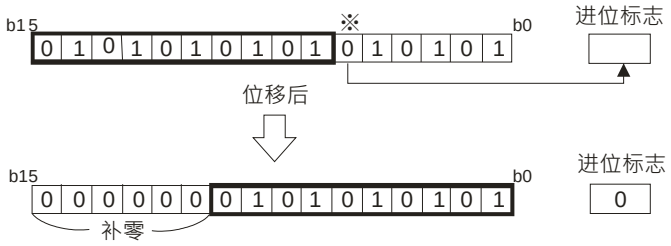
扫描一次的位右移动作依照下列编号 1~3 动作。

❶ b5~b0 → 进位（SM602=b5）

❷ b15~b6 → b9~b0

❸ 0 → b15~b10 完成





补充说明：

n<0 或 **n**>16（16 位指令）或 **n**>32（32 位指令），指令不执行，SM0=ON，错误码 SR0=16#200B。

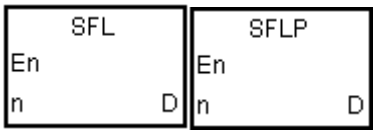
API	指令码			操作数	功能
1111	D	SFL	P	**D**，**n**	寄存器位左移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	AS-

符号：



D ： 欲位移之装置

n ： 一次位移之位数

指令说明：

1. 将 **D** 往左移 **n** 个位数，而因位移而腾出之空位(b_0–b_{n-1})则补零，16 位指令因位移而挤出之第 16-**n** 位(b_{16-n})状态则送到进位标志 SM602；32 位指令因位移而挤出之第 32-**n** 位 (b_{32-n}) 状态则送到进位标志 SM602。
2. 16 位指令 **n** 内容值的范围为 1~16。32 位指令 **n** 内容值的范围为 1~32。
3. 本指令一般都是使用脉冲执行型指令 (SFLP、DSFLP)。
4. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

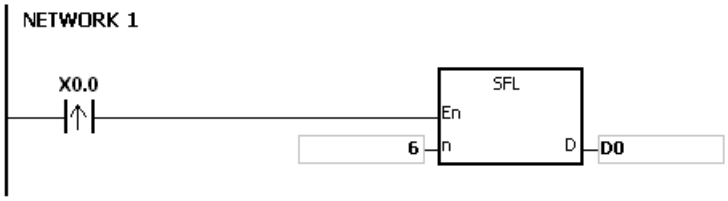
当 X0.0=ON 时，D0 的 b0~b15，往左移动 6 个位数，而 b10 的状态传送到 SM602，位移后，b0~b5 内容清除为 0。

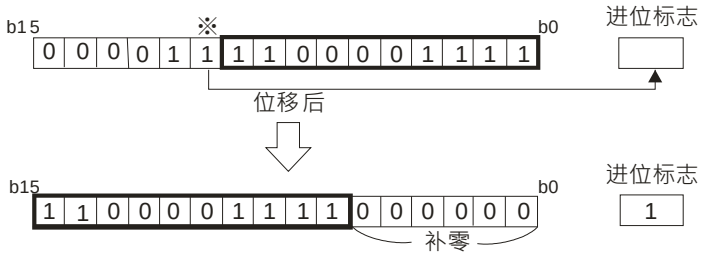
扫描一次的位右移动作依照下列编号 1~3 动作。

❶ b15~b10 → 进位 (SM602=b10)

❷ b9~b0 → b15~b6

❸ 0 → b5~b0 完成





补充说明：

n<0 或 **n**>16（16 位指令）或 **n**>32（32 位指令），指令不执行，SM0=ON，错误码 SR0=16#200B。

API	指令码			操作数	功能
1112		BSFR	P	**D**，**n**	n 个位右移 1 个位（位装置）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●	●	●				●								
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

D ： 移位装置之起始编号

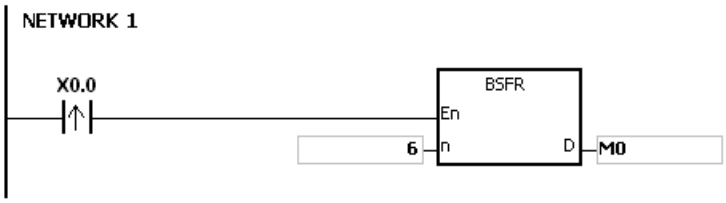
n ： 指定位移位置

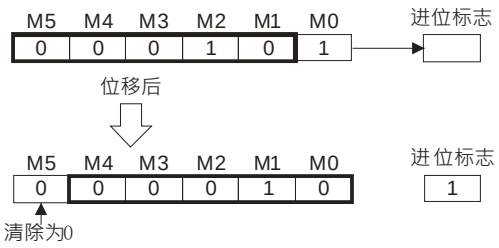
指令说明：

1. 将 **D** 开始的 **n** 个位，右移 1 个位，**D**+**n**-1 位被清除为 0，**D** 位状态则送到进位标志 SM602。
2. 本指令一般都是使用脉冲执行型指令（BSFRP）。
3. **n** 操作数的范围为 1~1024。

程序范例：

当 X0.0=ON 时，M0~M5 依序向右位移 1 个位，M5 的状态清除为 0，M0 的状态传送至进位标志 SM602。





补充说明：

1. **D**+n-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>1024，指令不执行，SM0=ON，错误码 SR0=16#200B。

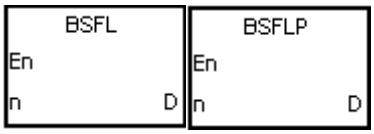
API	指令码		操作数	功能
1113	BSFL	P	**D**，**n**	n 个位左移 1 个位

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●	●	●				●								
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



D ： 移位装置之起始编号

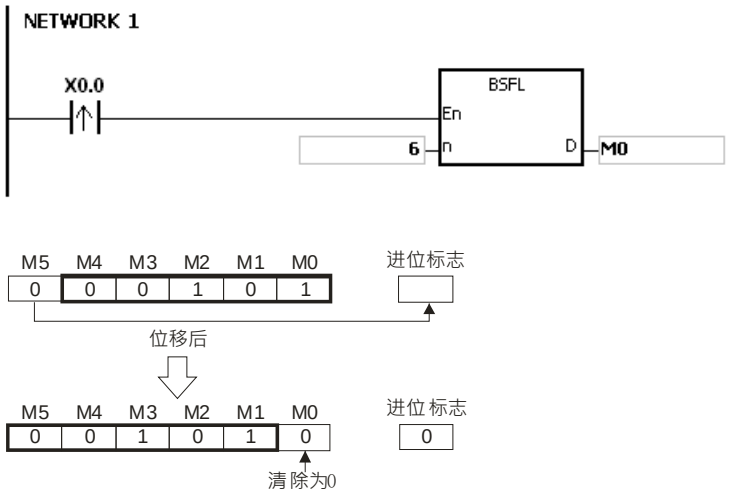
n ： 指定位移位置

指令说明：

1. 将 **D** 开始的 **n** 个位，左移 1 个位，**D** 位被清除为 0，**D**+n-1 位状态则送到进位标志 SM602。
2. 本指令一般都是使用脉冲执行型指令（BSFLP）。
3. **n** 操作数的范围为 1~1024。

程序范例：

当 X0.0=ON 时，M0~M5 内容依序向左位移 1 个位，M0 的状态清除为 0，M5 的状态传送至进位标志 SM602。



补充说明：

1. **D**+n-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>1024，指令不执行，SM0=ON，错误码 SR0=16#200B。

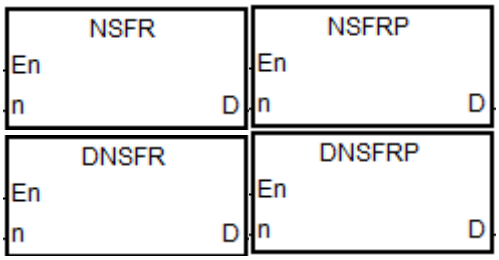
API	指令码			操作数	功能
1114	D	NSFR	P	**D**，**n**	n 个寄存器右移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●		●								
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ： 移位装置之起始编号

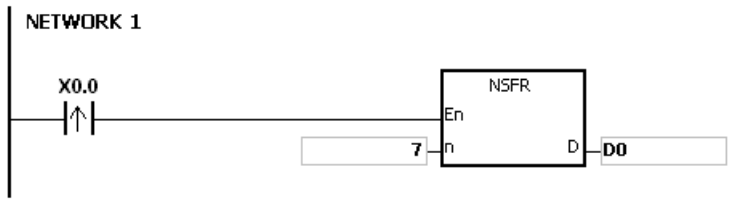
n ： 指定位移数据长度

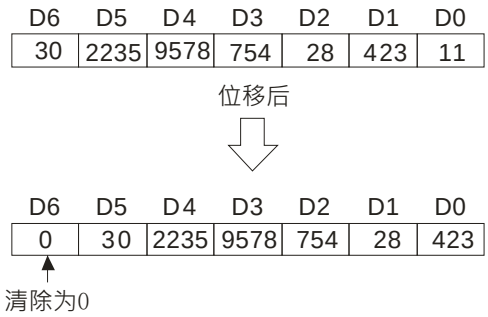
指令说明：

1. 将 **D** 开始的 n 个寄存器右移，**D**+n-1 装置被清除为 0。
2. 本指令一般都是使用脉冲执行型指令（NSFRP）。
3. **n** 操作数的范围，16 位指令为 1~512，32 位指令为 1~256。32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。

程序范例 1：

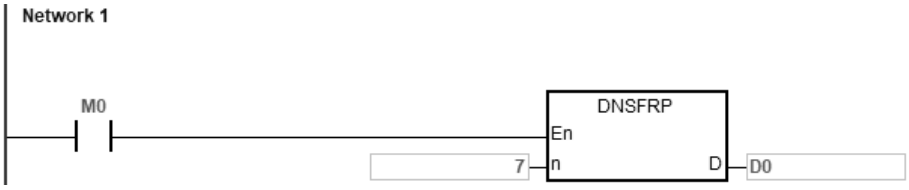
当 M0=OFF→ON 时，D1~D6 内容依序向右位移 1 个寄存器位置，最后 D6 的内容清除为 0。





程序范例 2：

当 M0=OFF→ON 时，D0~D13 为 7 笔 32 位数据，内容依序向右位移 1 笔 32 位位置，最后 D12 的内容清除为 0。



D12	D10	D8	D6	D4	D2	D0
90000	80000	70000	60000	50000	30000	40000

位移后

D12	D10	D8	D6	D4	D2	D0
0	90000	80000	70000	60000	50000	30000

补充说明：

1. **D**+n-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

   n<1 或 **n**>512（16 位指令）或 **n**>256（32 位指令），指令不执行，SM0=**ON**，错误码 SR0=16#200B

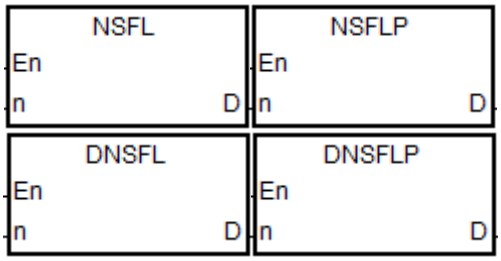
API	指令码			操作数	功能
1115	D	NSFL	P	**D**，**n**	n 个寄存器左移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●		●								
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



D ： 移位装置之起始编号

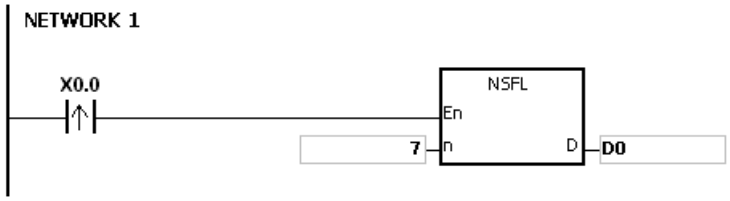
n ： 指定位移数据长度

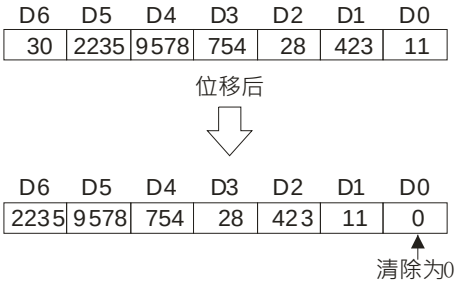
指令说明：

1. 将 **D** 开始的 **n** 个寄存器左移，**D** 装置被清除为 0。
2. 本指令一般都是使用脉冲执行型指令（NSFLP）。
3. **n** 操作数的范围，16 位指令为 1~512，32 位指令为 1~256。32 位指令需搭配 ISPSoft 版本 V3.09.07 版，以及 AS 主机韧体版本为 V1.08 版以上。

程序范例：

当 M0=OFF→ON 时，D0~D5 内容依序向左位移 1 个寄存器位置，最后 D0 的内容清除为 0。





补充说明：

1. **D**+n-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. n<1 或 n>512（16 位指令）或 n>256（32 位指令），指令不执行，SM0=ON，错误码 SR0=16#200B。

6.13 数据处理指令

6.13.1 数据处理指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1200	SER	DSER	✓	多点比较
1201	SUM	DSUM	✓	ON 位数量
1202	DECO	–	✓	译码器
1203	ENCO	–	✓	编码器
1204	SEGD	–	✓	7 段显示器解码
1205	SORT	DSORT	✓	数据排序
1206	ZRST	–	✓	区域清除
1207	BON	DBON	✓	ON 位判定
1208	MEAN	DMEAN	✓	平均值
1209	CCD	–	✓	总和检查
1210	ABS	DABS	✓	绝对值
1211	MINV	–	✓	矩阵反相
1212	MBRD	–	✓	矩阵位读出
1213	MBWR	–	✓	矩阵位写入
1214	MBC	–	✓	矩阵位状态计数
1215	DIS	–	✓	16位数据的4位分组
1216	UNI	–	✓	16位数据的4位链结
1217	WSUM	DWSUM	✓	总和运算
1221	LIMIT	DLIMIT	✓	高低限控制
1222	BAND	DBAND	✓	死区控制
1223	ZONE	DZONE	✓	区域控制
1224	–	FMEAN	✓	浮点数平均值
1225	–	FSUM	✓	浮点数加总
1226	DTM	–	✓	数据转换与搬移
1228	ZSET	–	✓	区域设定
1229	-	FSORT	✓	浮点数数据排序

6.13.2 数据处理指令说明

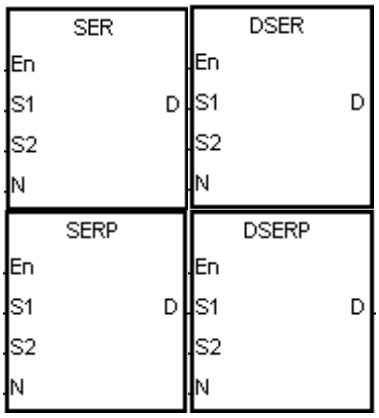
API	指令码			操作数	功能
1200	D	SER	P	**S_1**，**S_2**，**D**，**n**	多点比较

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●							
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●								
n	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ：多点比较之数据区块的起始装置

S_2 ：欲比较之数值数据

D ：存放比较结果之起始装置

n ：被比较之数据区块长度

指令说明：

1. **S_1** 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_2** 所指定的数据，以有号数十进制作比较，比较结果被存放于 **D~D+4** 寄存器当中，并说明如下：

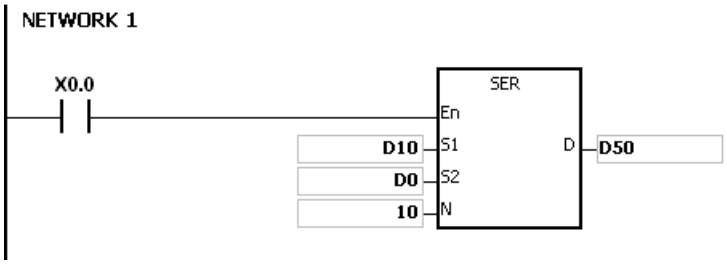
装置	说明
D	相等值的数据数
D+1	第一个相等值的编号
D+2	最后一个相等值的编号
D+3	最小值的编号
D+4	最大值的编号

2. **n** 操作数的内容值范围为 1~256。

3. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

1. 当 X0.0=ON 时，由 D10~D19 组成之数据区块与 D0 作比较，结果存放在 D50~D54 中，当相等值不存在时，D50~D52 的内容全部为 0。
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。



n

S_1	内容值	比较数据	数据编号	结果
D10	88	S_2 D0=100	0	
D11	100		1	相等
D12	110		2	
D13	150		3	
D14	100		4	相等
D15	300		5	
D16	100		6	相等
D17	5		7	最小
D18	100		8	相等
D19	500		9	最大

D	内容值	说明
D50	4	相等值的数据数
D51	1	第一个相等值的编号
D52	8	最后一个相等值的编号
D53	7	最小值的编号
D54	9	最大值的编号

补充说明：

1. **S_1+n**-1 或 **D**+4，超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 16 位指令 **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 32 位指令 **n**<1 或 **n**>128 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 16 位指令 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [5] of WORD/INT。
5. 32 位指令 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [5] of DWORD/DINT。

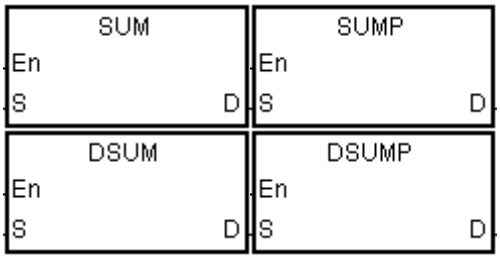
API	指令码			操作数	功能
1201	D	SUM	P	**S**、**D**	ON 位数量

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 来源装置

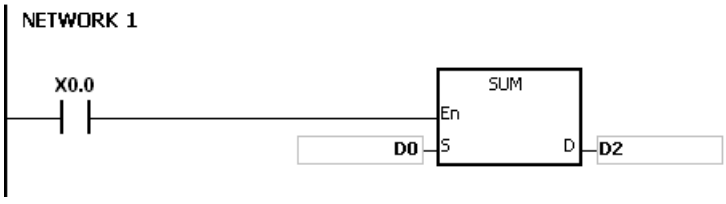
D ： 存放计数值的目的地装置

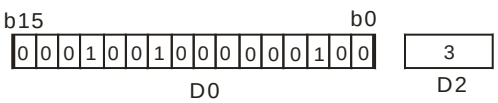
指令说明：

1. 在 **S** 中，所有位内容为“1”的总数将被储存于 **D**。
2. 如果来源装置 **S** 的位全部为“0”时，零标志信号 SM600=ON。
3. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

当 X0.0 为 ON 时，D0 的 16 个位中，内容为“1”的位总数被存于 D2 当中。





补充说明：

装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16# 2003。

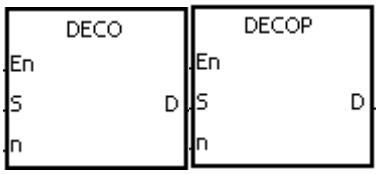
API	指令码			操作数	功能
1202		DECO	P	**S**・**D**・**n**	译码器

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●	●	●	●	●		●	●	○	○	○				
D		●	●	●	●	●		●				○				
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●	●				●							
D	●	●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 译码来源装置

D ： 存放译码结果之装置

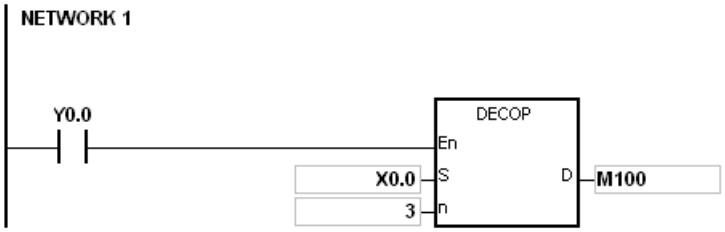
n ： 译码位长度

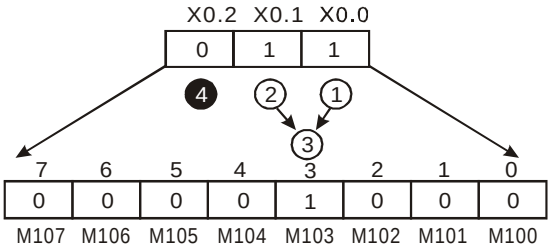
指令说明：

1. 来源装置 **S** 为字符装置时，将以下 8 位之“n”位作译码，并将其“2^n”位长度的结果存于 **D**。
2. 来源装置 **S** 为位装置时，将以连续“n”位作译码，并将其“2^n”位长度的结果存于 **D**。
3. 来源装置 **S** 为选用 T、C 元件时，都视为字符装置。
4. 当 **D** 为位装置时，**n** = 1~8，当 n=8 时，可做最大解码 2^8=256 点。(须注意译码后的装置储存范围，勿重复使用)。
5. **D** 为字符装置时，**n** = 1~4，当 n=4 时，可做最大解码 2^4=16 点。
6. 本指令一般都是使用脉冲执行型指令 (DECOP)。

程序范例一：

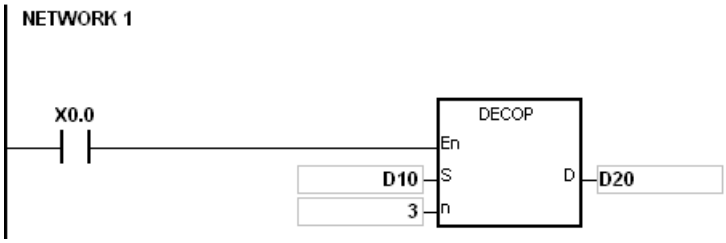
1. Y0.0 = OFF→ON 时，DECO 指令将 X0.0~X0.2 的内容值译码到 M100~M107。
2. 当数据源 X0.0~X0.2 的内容值为 1+2 = 3 时，从 M100 开始算第 3 个位 M103 设定为 1。
3. 当 DECO 指令执行过后，而 Y0.0 变为 OFF，已经做解码输出者照常动作。

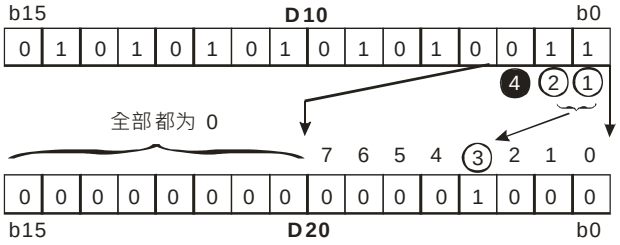




程序范例二：

1. X0.0 = OFF→ON 时，DECO 指令将 D10 中（b2~b0）的内容值解码到 D20 的（b7~b0）。D20 中未被使用之位（b15~b8）全部变为 0。
2. D10 的下位 3 位作译码存放于 D20 之下位 8 位，上 8 位皆为 0。
3. 当 DECO 指令执行过后，而 X0.0 变为 OFF 后，已经做解码输出者照常动作。





补充说明：

1. **D** 为位装置时，若 **n**<1 或 **n**>8，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. **D** 为字符装置时，若 **n**<1 或 **n**>4，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 若 **S** 为位装置时，**S+n**-1 位超过范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. 若 **D** 为位装置时，**D+（2^n）-1** 位超过范围，指令不执行，SM0=ON，错误码 SR0=16#2003。

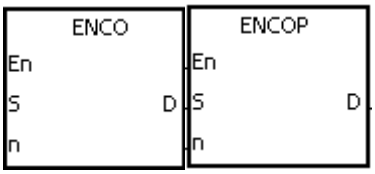
API	指令码			操作数	功能
1203		ENCO	P	**S**、**D**、**n**	编码器

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●	●	●	●	●		●	●	○	○	○				
D		●			●	●		●			○	○				
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●	●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 编码来源装置

D ： 存放编码结果之装置

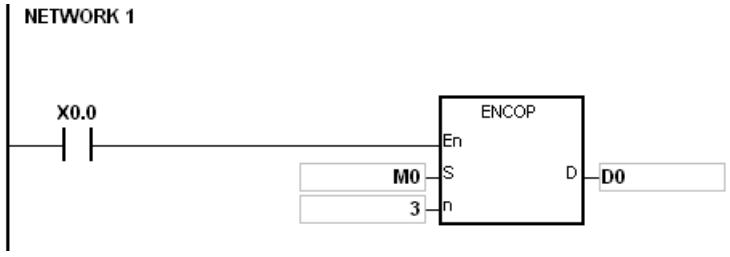
n ： 编码位长度

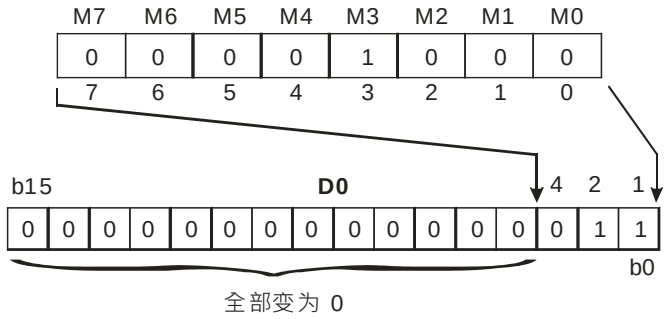
指令说明：

1. 来源装置 **S** 为字符装置时，将以下位“2^n”位数据作编码，并将结果存于 **D**。
2. 来源装置 **S** 有位装置时，将由高位 **S**+（n-1）往低位 **S** 的“2^n”位数据作编码，并将结果存于 **D**。
3. 来源装置 **S** 为选用 T、C 元件时，都视为字符装置。
4. **S** 为位装置时，**n** = 1~8，当 **n**=8 时，可做 2^8=256 点编码。
5. **S** 为字符装置时，**n** = 1~4，当 **n**=4 时，可做 2^4=16 点编码。
6. 本指令一般都是使用脉冲执行型指令（ENCOP）。

程序范例一：

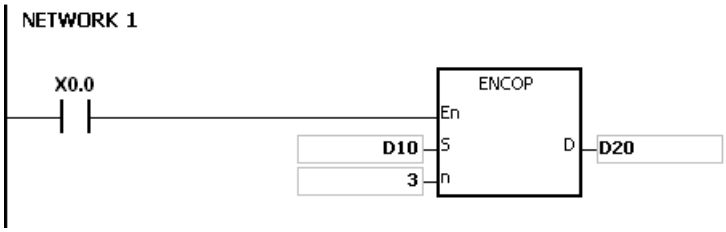
1. 当 X0.0 = OFF→ON 时，ENCO 指令将 2^3 位数据（M0~M7）编码存放于 D0 之下位 3 位（b2~b0）内，D0 中未被使用之位（b15~b3）全部变为 0。
2. 当 ENCO 指令执行过后，而 X0.0 变为 OFF 后，**D** 内数据不变。

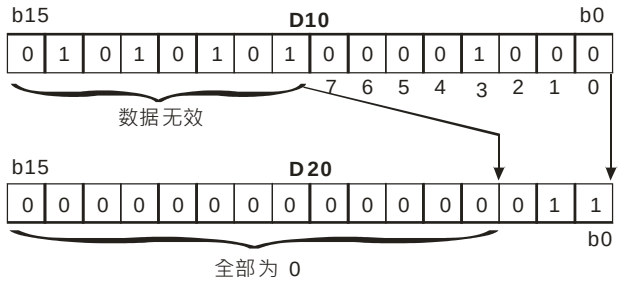




程序范例二：

1. 当 X0.0 = OFF→ON 时，D10 内 2^3 bits 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）
2. 当 ENCO 指令执行过后，而 X0.0 变为 OFF，D 内数据不变。





补充说明：

1. 若数据源 **S** 都没有位为 1 时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **S** 为位装置时，若 **n**<1 或 **n**>8，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. **S** 为字符装置时，若 **n**<1 或 **n**>4，指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 若 **S** 为位装置时，**S+（2^n）-1** 位超过范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
5. 若 **D** 为位装置时，**D+n-1** 位超过范围，指令不执行，SM0=ON，错误码 SR0=16#2003。

API	指令码			操作数	功能
1204		SEGD	P	**S**，**D**	7 段显示器解码

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ：欲译码之来源装置

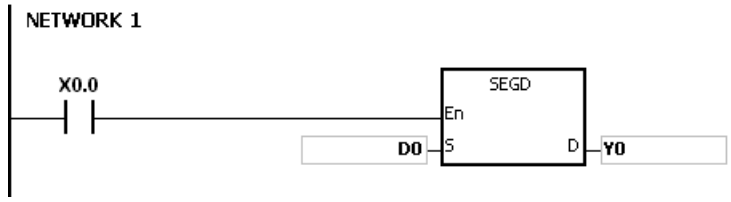
D ：译码后之输出装置

指令说明：

来源装置 **S** 的下位 4 个位（b0~b3），被解码成 7 段显示器的格式，并将结果存于 **D**。

程序范例：

当 X0.0=ON 时，D0 的下位 4 个位（b0~b3）的内容（0~F：16 进制）被解码成 7 段显示器输出，解码的结果暂存于 Y0.0~Y0.15 当中。若指定数据超出 4 个位，仍取下位 4 个位的内容译码。



七段显示器解码表：

16进数	位组合	七段显示器的构成	各节段状态							显示数据
			B0(a)	B1(b)	B2(c)	B3(d)	B4(e)	B5(f)	B6(g)	
0	0000		ON	ON	ON	ON	ON	ON	OFF	
1	0001		OFF	ON	ON	OFF	OFF	OFF	OFF	
2	0010		ON	ON	OFF	ON	ON	OFF	ON	
3	0011		ON	ON	ON	ON	OFF	OFF	ON	
4	0100		OFF	ON	ON	OFF	OFF	ON	ON	
5	0101		ON	OFF	ON	ON	OFF	ON	ON	
6	0110		ON	OFF	ON	ON	ON	ON	ON	
7	0111		ON	ON	ON	OFF	OFF	ON	OFF	
8	1000		ON	ON	ON	ON	ON	ON	ON	
9	1001		ON	ON	ON	ON	OFF	ON	ON	
A	1010		ON	ON	ON	OFF	ON	ON	ON	
B	1011		OFF	OFF	ON	ON	ON	ON	ON	
C	1100		ON	OFF	OFF	ON	ON	ON	OFF	
D	1101		OFF	ON	ON	ON	ON	OFF	ON	
E	1110		ON	OFF	OFF	ON	ON	ON	ON	
F	1111		ON	OFF	OFF	OFF	ON	ON	ON	

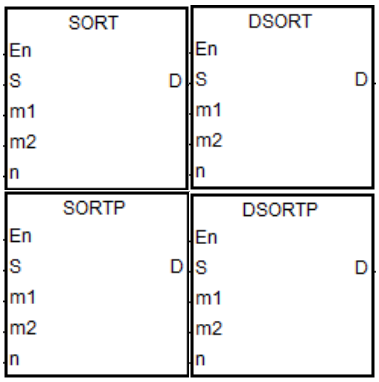
API	指令码			操作数	功能
1205	D	SORT	P	**S**・**m_1**・**m_2**・**D**・**n**	数据排序

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S								●								
m_1								●					○	○		
m_2								●					○	○		
D								●								
n								●					○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
m_1		●	●			●	●						
m_2		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 原始数据区块之起始装置

m_1 ： 被排序之数据组数

m_2 ： 每笔数据之字段数

D ： 存放排序结果数据区块之起始装置

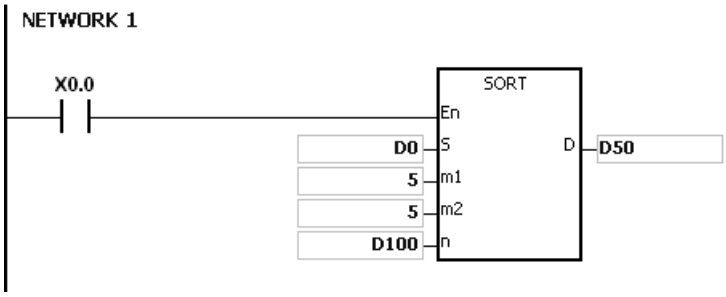
n ： 数据排序的参考值

指令说明：

1. 排序结果显示于 **D** 所指定的起始号码开始算的 **m_1×m_2** 个寄存器当中，当 **S** 与 **D** 指定同一个寄存器时，其排序结果将与原来被排序的数据 **S** 内容相同。
2. **m_1**=1~32，**m_2**=1~6，**n**=1~ **m_2**。
3. SM604=OFF 时为升幂排序，SM604=ON 时为降序排序。
4. 建议使用脉冲型执行指令，避免一直重复排序。
5. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

1. 将 SM604=OFF，当 X0.0=OFF→ON 时，指定执行数据进行升幂排序作业。



2. 排序数据构成例

数据数：m_2 个

数据字段

行 / 列	1	2	3	4	5
	学生编号	国文	英文	数学	理化
1	(D0) 1	(D5) 90	(D10) 75	(D15) 66	(D20) 79
2	(D1) 2	(D6) 55	(D11) 65	(D16) 54	(D21) 63
3	(D2) 3	(D7) 80	(D12) 98	(D17) 89	(D22) 90
4	(D3) 4	(D8) 70	(D13) 60	(D18) 99	(D23) 50
5	(D4) 5	(D9) 95	(D14) 79	(D19) 75	(D24) 69

数据个数：m_1 个

3. D100=3 时的排序后数据

数据数：m_2 个

数据字段

行 / 列	1	2	3	4	5
	学生编号	国文	英文	数学	理化
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

数据个数：m_1 个

4. D100=5 时的排序后数据。

数据数：m_2 个

数据字段

数据个数：m_1 个 ↓	行 / 列	1	2	3	4	5
		学生编号	国文	英文	数学	理化
	1	(D50) 4	(D55) 70	(D60) 60	(D65) 99	(D70) 50
	2	(D51) 2	(D56) 55	(D61) 65	(D66) 54	(D71) 63
	3	(D52) 5	(D57) 95	(D62) 79	(D67) 75	(D72) 69
	4	(D53) 1	(D58) 90	(D63) 75	(D68) 66	(D73) 79
	5	(D54) 3	(D59) 80	(D64) 98	(D69) 89	(D74) 90

补充说明：

1. 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **m_1**、**m_2** 或 **n** 超出范围时，指令不执行，SM0=ON，错误码 SR0=16#200B。

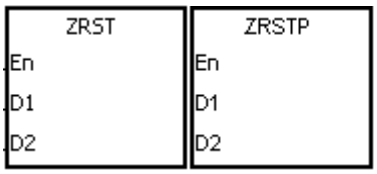
API	指令码		操作数	功能
1206	ZRST	P	D_1 · D_2	区域清除

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
D_1		●	●	●	●	●	●	●		○	○	○				
D_2		●	●	●	●	●	●	●		○	○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D_1	●	●				●							
D_2	●	●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



D_1 ： 区域清除起始装置

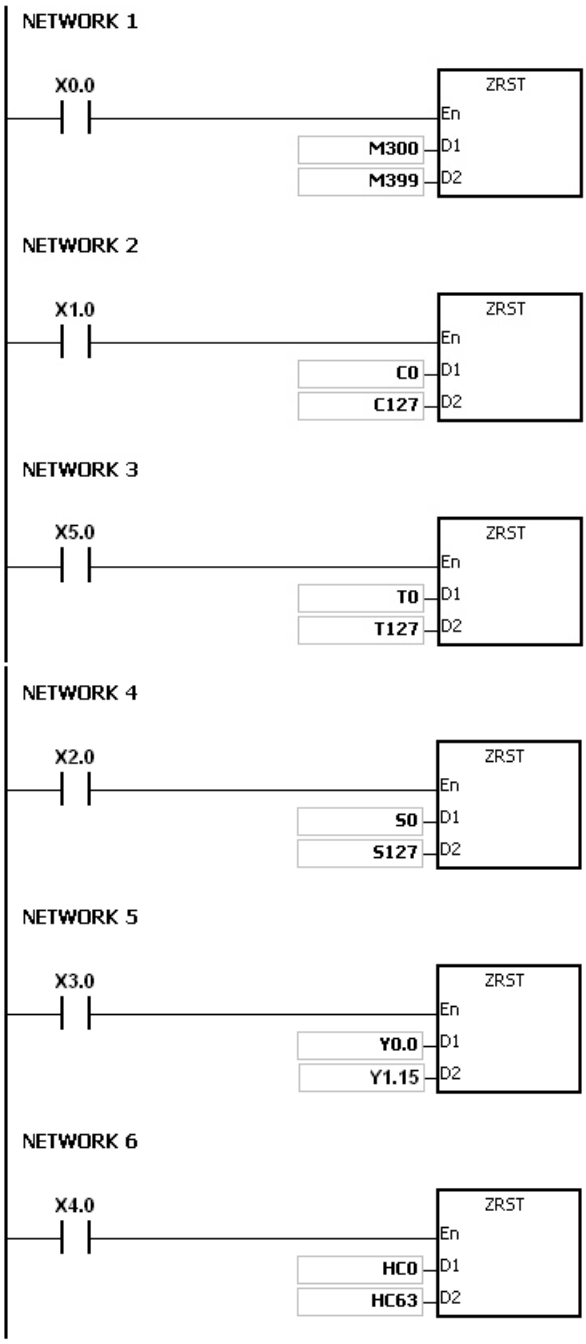
D_2 ： 区域清除结束装置

指令说明：

1. 清除 **D_1~D_2** 间的操作数内容值，故装置类别必须相同，才可进行清除动作。
2. 当 **D_1** 操作数编号 > **D_2** 操作数编号时，只有 **D_2** 指定之操作数被清除。
3. ZRST 指令可直接指定 32 位 HC 装置。

程序范例：

1. 当 X0.0 为 ON 时，辅助继电器 M300~M399 被清除成 OFF。
2. 当 X1.0 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)。
3. 当 X2.0 为 ON 时，步进点 S0~S127 被清除成 OFF。
4. 当 X3.0 为 ON 时，数据寄存器 Y0.0~Y1.15 被清除成 OFF。
5. 当 X4.0 为 ON 时，32 位计数器 HC0~HC63 全部清除。(写入 0，并将接点及线圈清除成 OFF)。
6. 当 X5.0 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)。



补充说明：

1. **D_1**、**D_2** 操作数装置类别不相同，指令不执行，SM0=ON，错误码 SR0=16#2007。
2. **D_1**、**D_2** 操作数数据格式不相同，指令不执行，SM0=ON，错误码 SR0=16#2007。

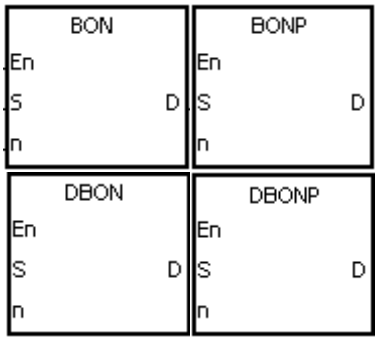
API	指令码			操作数	功能
1207	D	BON	P	**S**・**D**・**n**	ON 位判定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●	●	●				●		○						
n	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D	●												
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ：来源装置

D ：存放判定结果之装置

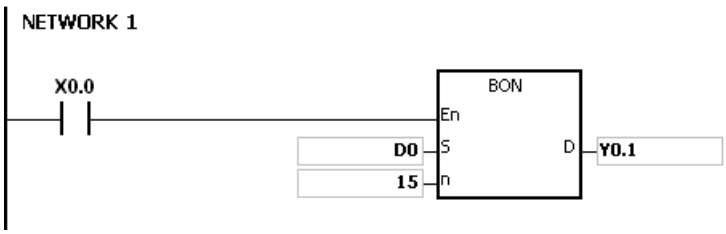
n ：指定判定之位（自 0 开始编号）

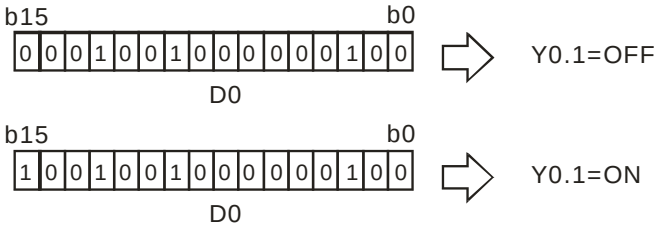
指令说明：

1. 判断 **S** 内容值的第 **n** 个位的状态，并将其状态存放到 **D** 指定位。
2. 16 位指令 **n**=0~15，32 位指令 **n=**0~31。
3. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

1. 当 X0.0=ON 时，若是 D0 的第 15 个位为“1”时，Y0.1=ON，为“0”时，Y0.1=OFF。
2. X0.0 变成 OFF 时，Y0.1 仍保持之前的状态。





补充说明：

n 超出范围时，指令不执行，SM0=ON，错误码 SR0=16#200B。

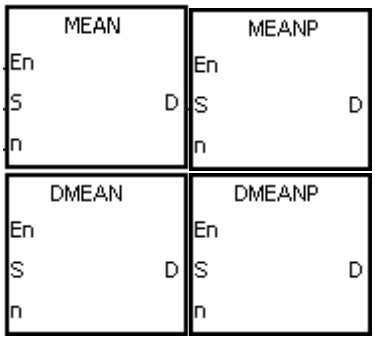
API	指令码			操作数	功能
1208	D	MEAN	P	**S**・**D**・**n**	平均值

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●							
D		●			●	●	●	●			○	○				
n	●	●			●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 欲取平均值之起始装置

D ： 存放平均值之装置

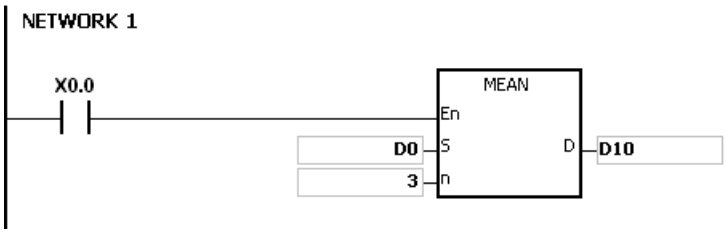
n ： 取平均值之装置个数

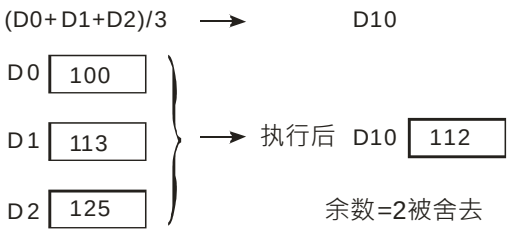
指令说明：

1. 将 **S** 起始之 **n** 个装置内容值相加后取平均值存入 **D** 中。
2. 如果计算中出现余数时，余数会被舍去。
3. **n** 操作数的内容值范围为 1~256。
4. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

当 X0.0=ON 时，D0 开始算的 3 个 (n = 3) 寄存器的内容全部相加，相加之后再除以 3 以求得平均值并存于指定的 D10 当中，余数被舍去。





补充说明：

1. 16 位指令中，**n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. 32 位指令中，**n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 若 **S**+**n**-1 装置，超过装置范围时，SM0=ON，错误码 SR0=16#2003。

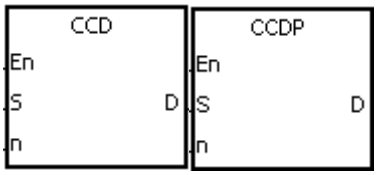
API	指令码			操作数	功能
1209		CCD	P	**S**、**D**、**n**	总和检查

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 数据源起始装置

D ： 存放总和检查之结果

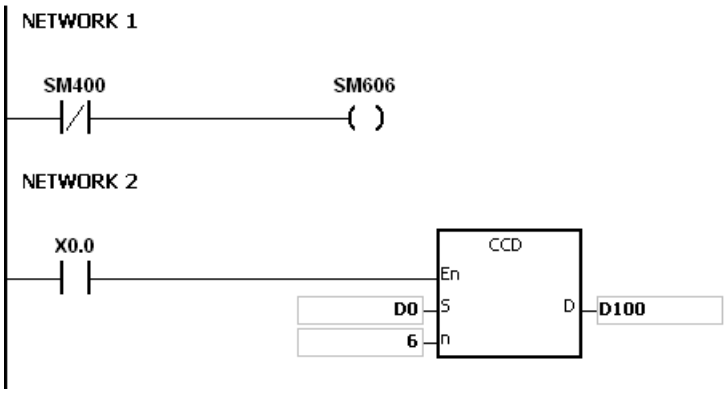
n ： 数据个数

指令说明：

1. 本指令目的为确保数据串之内容正确性，所做的数据总和检查（Sum Check）。
2. 16 位模式：当 SM606=OFF 时，指定为 16 位转换模式。将 **S** 所指定寄存器起始号码开始算的 n 个数据（以 8 位为单位）(以下 8，上 8，下 8，上 8...顺序）内容作加总，最后加总结果存放于 **D** 所指定的寄存器当中，而极性位存放于 **D**+1 当中。
3. 8 位转换模式：当 SM606=ON 时，指定为 8 位转换模式。将 **S** 所指定寄存器起始号码开始算的 n 个数据（以 8 位为单位，且只有下 8 位有效）内容作加总，加总结果存放于 **D** 所指定的寄存器当中，而极性位存放于 **D**+1 当中。
4. **n** 范围为 1~256。

程序范例一：

1. 当 SM606=OFF 时，指定为 16 位转换模式。
2. 当 X0.0=ON 时，将 D0 所指定寄存器起始号码开始算的 6 个数据（以 8 位为单位 n=6 代表指定 D0~D2）内容作加总，加总结果存放于 D100 所指定的寄存器当中，而极性位存放于 D101 当中。



S	数据的内容	
D0 下	100=01100100	
D0 上	111=01101111	←
D1 下	120=01111000	
D1 上	202=11001010	
D2 下	123=01111011	←
D2 上	211=11010011	←
D100	867	合計
D101	00010001	

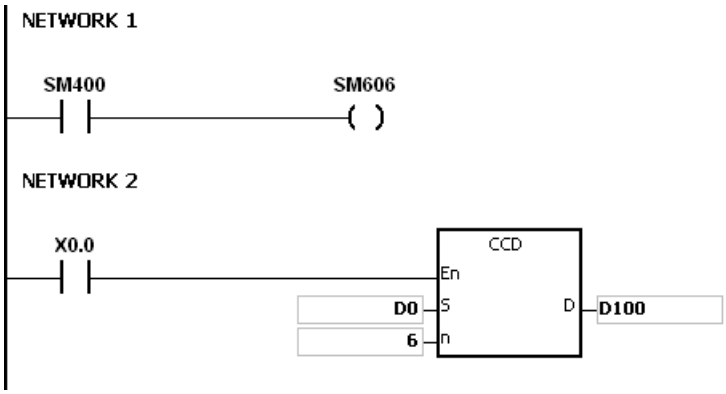
奇数个"1"时,极性为"1"
偶数个"1"时,极性为"0"

D100	0	0	0	0	0	0	1	1	0	1	1	0	0	0	1	1
D101	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

⇦ 极性

程序范例二：

1. 当 SM606=ON 时，指定为 8 位转换模式。
2. 当 X0.0=ON 时，将 D0 所指定寄存器起始号码开始算的 6 个数据（以 8 位为单位 n=6 代表指定 D0~D5）内容作加总，加总结果存放于 D100 所指定的寄存器当中，而极性位存放于 D101 当中。



S	数据的内容	
D0 下	100 = 0 1 1 0 0 1 0 0	
D1 下	111 = 0 1 1 0 1 1 1 (1)	←
D2 下	120 = 0 1 1 1 1 0 0 0	
D3 下	202 = 1 1 0 0 1 0 1 0	
D4 下	123 = 0 1 1 1 1 0 1 (1)	←
D5 下	211 = 1 1 0 1 0 0 1 (1)	←
D100	867	合计
D101	0 0 0 1 0 0 0 (1)	

奇数个"1"时,极性为"1"
偶数个"1"时,极性为"0"

D100	0	0	0	0	0	0	1	1	0	1	1	0	0	0	1	1
D101	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

⇦ 极性

补充说明：

1. SM606=ON（8 位模式）时，S+n-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. SM606=OFF（16 位模式）时，S+n/2-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **n**<1 或 **n**>256 时，指令不执行，SM0= ON，错误码 SR0=16#200B。
4. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。

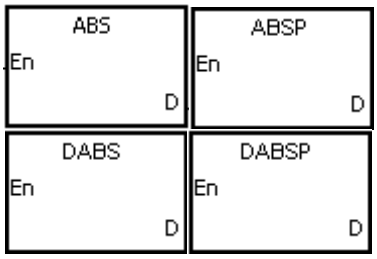
API	指令码			操作数	功能
1210	D	ABS	P	**D**	绝对值

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



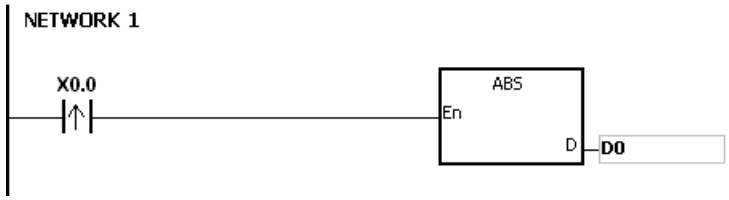
D ： 欲取绝对值之装置

指令说明：

1. 当 ABS 指令执行时，被指定的元件 **D** 取绝对值。
2. 本指令一般都是使用脉冲执行型指令（ABSP）。
3. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。
4. 当指令遇到有号数最小值时(负数最小值)，其绝对值的结果将不会变化，举例 :16 位有号数最小值-32768，其结果还是-32768。以此规则，32 位有号数最小值（负数最小值）也是不变化。

程序范例：

假设执行前 D0=-1234，当 X0.0=OFF→ON 时，D0 内容取绝对值，执行后 D0=1234。



API	指令码			操作数	功能
1211		MINV	P	**S**・**D**・**n**	矩阵反相

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●		○		○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 矩阵来源装置

D ： 运算结果

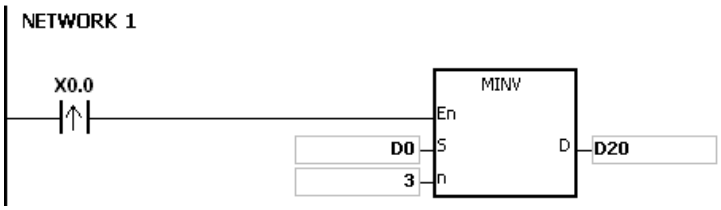
n ： 数组长度

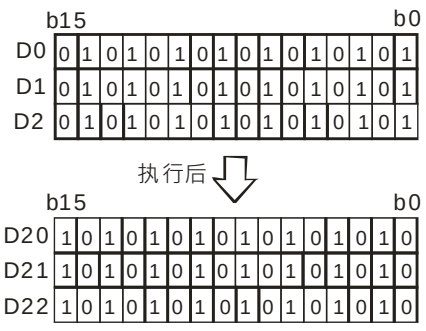
指令说明：

1. 矩阵来源 **S** 依数组长度 **n** 作矩阵的反相运算并将结果存于 **D**。
2. **n** 操作数的范围值为 1~256。

程序范例：

当 X0.0=ON 时，16 位寄存器 D0~D2 共 3 列作 MINV，矩阵反相运算，将结果存于 16 位寄存器 D20~D22 共 3 列中。





补充说明：

1. **S**+**n**-1、**D**+**n**-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

API	指令码			操作数	功能
1212		MBRD	P	**S**、**n**、**D**	矩阵位读出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●		●	●							
n	●	●			●	●		●	●		○		○	○		
D		●			●	●		●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 矩阵来源装置

n ： 数组长度

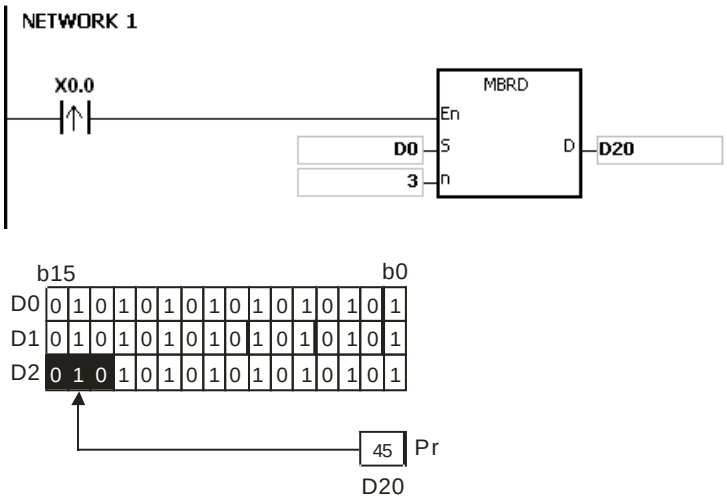
D ： 存放目标指针（Pr）之位置值

指令说明：

1. 当指令执行时，一开始判断矩阵指针清除标志 SM613 是否为 ON，若为 ON，指针 **D** 清除为 0，再将指针 **D** 所指定的 bit 状态读取到矩阵循环位移输出进位标志 SM614，当读取完一个 bit 时，判断矩阵指针递增标志 SM612 是否 ON，若 ON 把指针 **D** 的值加 1，然后结束本指令之执行。
2. 当读到最后一个 bit 时矩阵搜寻结束标志 SM608=ON，指针 **D** 记录着读取的 bit 之编号，然后结束本指令之执行。
3. **n** 操作数的范围值为 1~256。
4. 矩阵之指针 Pr（Pointer），由用户于指令中指定，其有效范围为 0～16**n**–1，分别对应至矩阵中之位 b_0～b_{16n-1}。若 Pr 值超出此范围则矩阵指针错误标志 SM611 设为 1，且本指令不执行。

程序范例：

1. 当 X0.0 由 OFF→ON 时，设指针清除标志 SM613=0、矩阵指针递增标志 SM612=1，所以每读取一次指针 Pr 增加 1。
2. 设指针当时值 D20=45，当 X0.0 由 OFF→ON 时动作 3 次，可得到如❶，❷，❸三个执行结果。
   - ❶ D20=46，矩阵循环位移输出进位标志 SM614=0，矩阵搜寻结束标志 SM608=0。
   - ❷ D20=47，矩阵循环位移输出进位标志 SM614=1，矩阵搜寻结束标志 SM608=0。
   - ❸ D20=47（已达最大 b_{16n-1}，故维持 47），矩阵循环位移输出进位标志 SM614=0，矩阵搜寻结束标志 SM608=1。



补充说明：

1. **S**+**n**-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 标志信号说明：

   SM608： 矩阵搜寻结束标志，当比较到最后一个 bit 时，SM608=1

   SM611： 矩阵指针错误标志，指针 Pr 值超出此范围则 SM611=1

   SM612： 矩阵指针递增标志，将指针目前值+1

   SM613： 矩阵指针清除标志，将指针目前值清除为 0

   SM614： 矩阵循环位移输出进位标志

API	指令码			操作数	功能
1213		MBWR	P	**S**、**n**、**D**	矩阵位写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S		●			●	●		●								
n	●	●			●	●		●	●		○		○	○		
D		●			●	●		●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 矩阵来源装置

n ： 数组长度

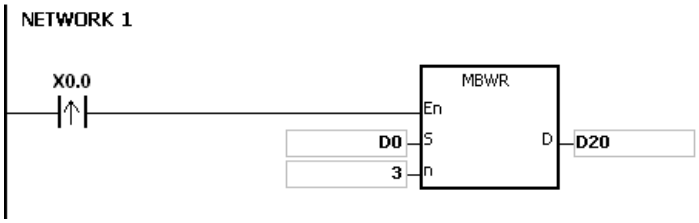
D ： 存放目标指针（Pr）之位置值

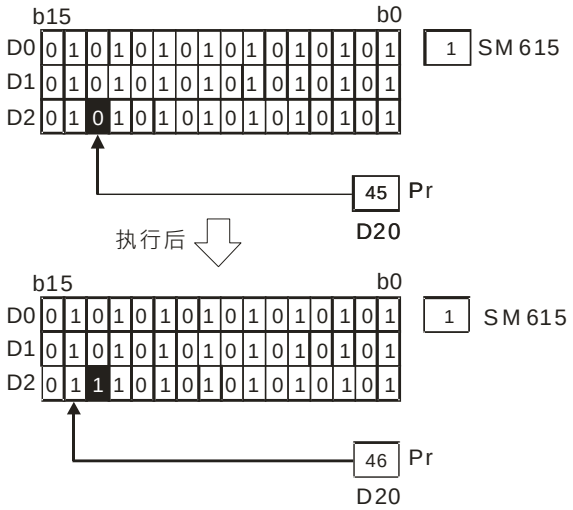
指令说明：

1. 当指令执行时，一开始判断矩阵指针清除标志 SM613 是否为 ON，若为 ON，指针 **D** 清除为 0，把矩阵位移输入补位标志 SM615 的值，由指针 **D** 所指定的位写入 **S** 矩阵，当写完一个 bit 时，判断矩阵指针递增标志 SM612 是否 ON，若 ON 把指针 **D** 的值加 1，然后结束本指令之执行。
2. 当写到最后一个 bit 时矩阵搜寻结束标志 SM608=ON，指针 **D** 记录着读取的 bit 之编号，然后结束本指令之执行。若 **D** 的值超过范围则 SM611=ON。
3. n 操作数的范围值为 1~256。
4. 矩阵之指针 Pr（Pointer），由用户于指令中指定，其有效范围为 0～16**n**–1，分别对应至矩阵中之位 b0～b16n–1。若 Pr 值超出此范围则矩阵指针错误标志 SM611 设为 1，且本指令不执行。

程序范例：

1. 当 X0.0 由 OFF→ON 时，设指针清除标志 SM613=0、矩阵指针递增标志 SM612=1，所以每写入一次指针 Pr 增加 1。
2. 设指针当时值 D20=45，矩阵位移输入补位标志 SM615 状态为 1，当 X0.0 由 OFF→ON 时动作 1 次，可得到执行结果为 D20=46，矩阵位移输入补位标志 SM615=1，矩阵搜寻结束标志 SM608=0。





补充说明：

1. **S**+**n**-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 标志信号说明：

   SM608： 矩阵搜寻结束标志，当比较到最后一个位时，SM608=1

   SM611： 矩阵指针错误标志，指针 Pr 值超出此范围则 SM611=1

   SM612： 矩阵指针递增标志

   SM613： 矩阵指针清除标志

   SM615： 矩阵位移输入补位标志

API	指令码			操作数	功能
1214		MBC	P	**S**、**n**、**D**	矩阵位状态计数

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S		●			●	●		●								
n	●	●			●	●		●	●		○		○	○		
D		●			●	●		●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ：矩阵来源装置

n ：数组长度

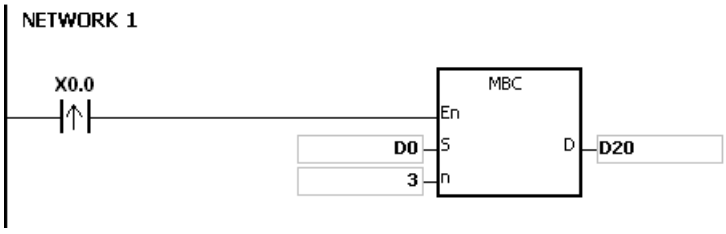
D ：运算结果

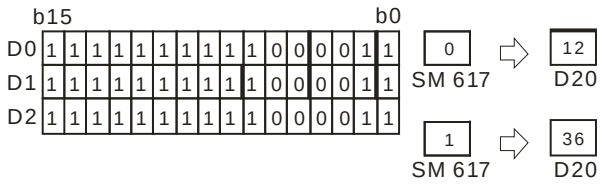
指令说明：

1. 依数组长度 **n** 计算 **S** 矩阵中所有位为 1 或为 0 的个数，并将数目存于 **D** 中。
2. 当 SM617=1 时计算矩阵位为 1 的个数，SM617=0 时，计算矩阵位为 0 的个数。当计算出来的结果为 0 时，SM618=1。
3. **n** 操作数的范围值为 1~256。

程序范例：

当 X0.0 为 ON 时，D0~D2 的矩阵中，当 SM617=1 时计算矩阵位为 1 的位总数被存于 D20 当中。当 SM617=0 时计算矩阵位为 0 的位总数被存于 D20 当中。





补充说明：

1. **S**+**n**-1 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 标志信号说明：

   SM617： 矩阵计数字元为 0 或位为 1 标志

   SM618： 矩阵计数结果为 0 时 ON

API	指令码			操作数	功能
1215		DIS	P	**S**・**n**・**D**	16 位数据的 4 位分组

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●		●	●		○	○				
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

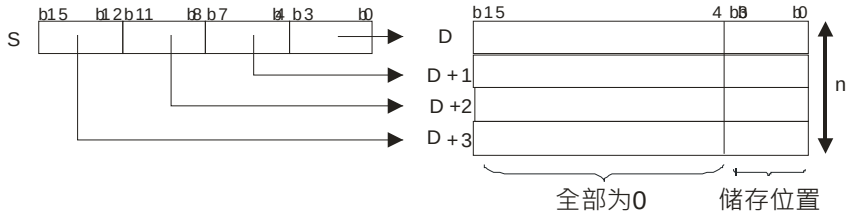
S ： 数据源装置

n ： 储存长度

D ： 运算结果

指令说明：

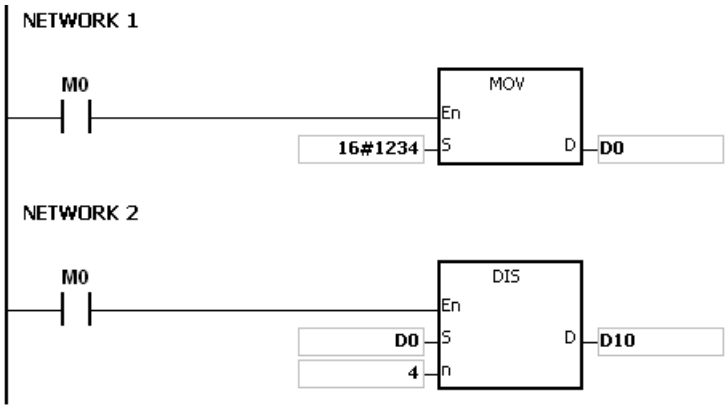
1. 将 **S** 所指定寄存器内容值，从 b_0 开始以 4 个位为 1 组，分别存放于 **D**~**D**+（**n**-1）所指定的寄存器的低 4 位中（b_0 ~ b_3）。

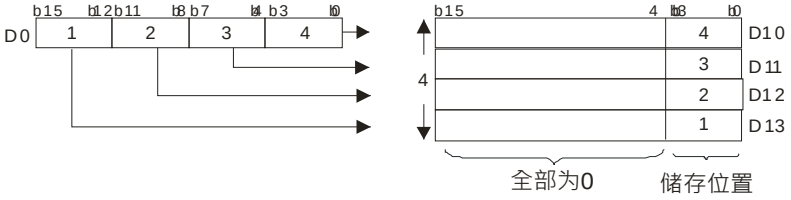


2. **n**=1 ~ 4。

程序范例：

D0 的内容为 16#1234，启动 M0，执行 DIS 指令时，将 D0 的 b0~b15 内容分为 4 个位一组，分别移动至 D10~D13 的 b0~b3 当中。





补充说明：

1. **D**~**D**+（**n**-1）装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>4 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

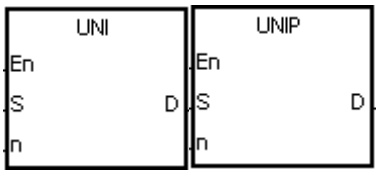
API	指令码		操作数	功能
1216	UNI	P	**S**、**n**、**D**	16 位数据的 4 位链结

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●		●	●							
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



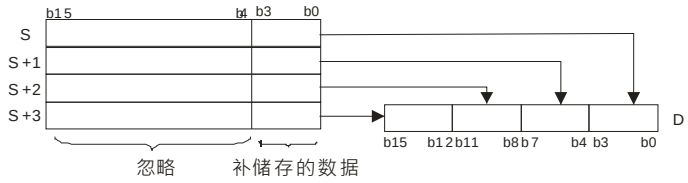
S ： 数据源装置

n ： 数据源长度

D ： 运算结果

指令说明：

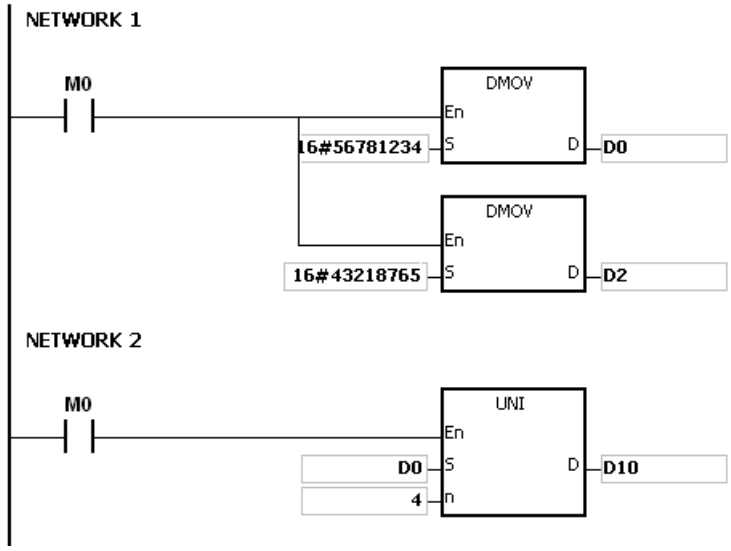
1. 将 **S~S+**（**n**-1）所指定寄存器内容值（$b_0 \sim b_3$），依续存放于 **D**（$b_0 \sim b_{15}$）所指定的寄存器中。

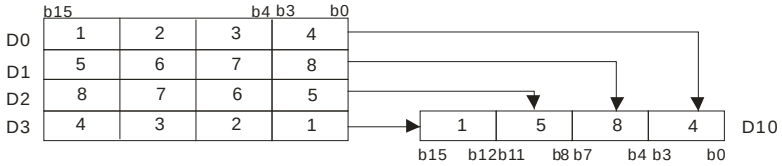


2. **n**=1 ~ 4。

程序范例：

D0~D3 的内容为分别为 16#1234、16#5678、16#8765、16#4321。启动 M0，执行 UNI 指令时，将 D0~D3 的 b0~b3 内容，分别储存至 D10 的 b0~b15 当中。





补充说明：

1. **S**~**S**+（**n**-1）装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **n**<1 或 **n**>4 时，指令不执行，SM0=ON，错误码 SR0=16#200B。

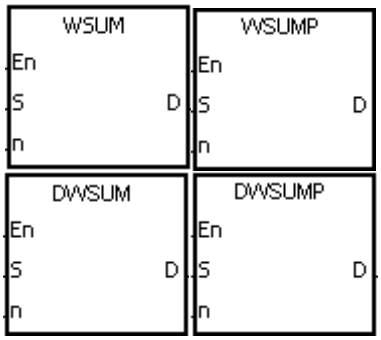
API	指令码			操作数	功能
1217	D	WSUM	P	**S**、**n**、**D**	总和运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●							
n	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
n		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



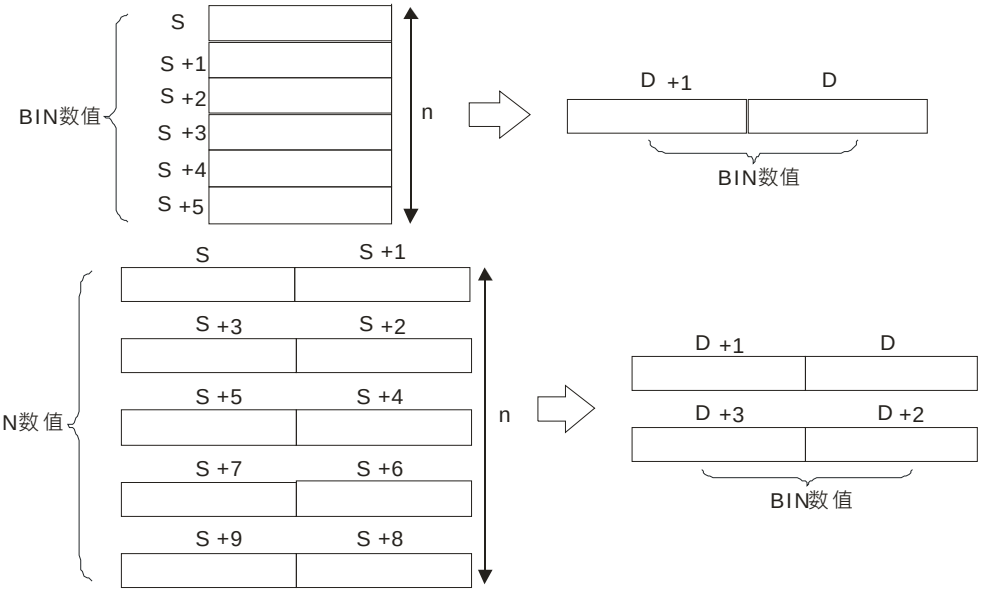
S ： 数据源装置

n ： 数据源长度

D ： 运算结果

指令说明：

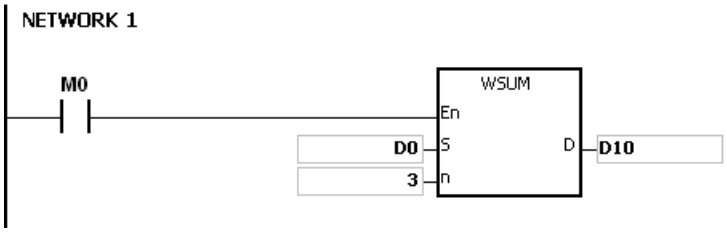
1. 将 **S~S+n-1** 所指定寄存器内容值，以有号数十进制作相加，存放于 **D** 所指定的寄存器中。

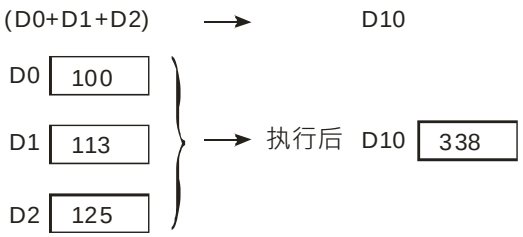


2. **n** 操作数的内容值范围为 1~256。
3. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

当执行 WSUM 指令时，会将 D0~D2 的内容相加后，计算结果（32 位数值）储存至 D10 当中。





补充说明：

1. 16 位指令中，**n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. 32 位指令中，**n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 若 **S**+**n**-1 或 D 装置，超过装置范围时，SM0=ON，错误码 SR0=16#2003。
4. 16 位指令，**D** 装置可宣告数据类型 DWORD 或 ARRAY [2] of WORD。
5. 32 位指令，**D** 装置需宣告数据类型 ARRAY [2] of DWORD。

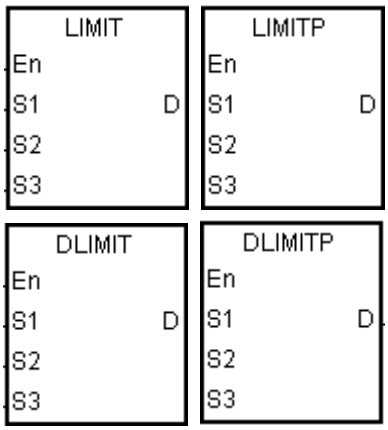
API	指令码			操作数	功能
1221	D	LIMIT	P	S_1 · S_2 · S_3 · **D**	高低限控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
S_3	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
S_3		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 最小输出值

S_2 ： 最大输出值

S_3 ： 输入值

D ： 输出值

指令说明：

1. 输入值 **S_3** 与最小输出值 **S_1** 及最大输出值 **S_2** 作比较，比较后将结果存入输出值 **D**，说明如下：

   若最小输出值 **S_1>**输入值 **S_3**，输出值 **D=**最小输出值 **S_1**。

   若最大输出值 **S_2<**输入值 **S_3**，输出值 **D=**最大输出值 **S_2**。

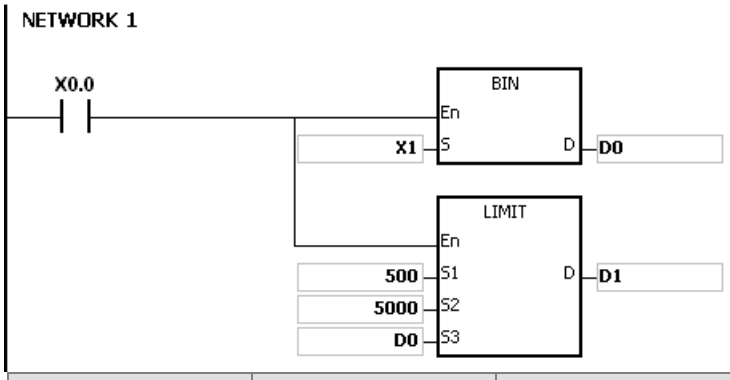
   若最小输出值 **S_1**≦输入值 **S_3**≦最大输出值 **S2**，输出值 **D=**输入值 **S_3**。

   若最小输出值 **S_1>**最大输出值 **S_2**，则指令不执行，输出值 **D** 不变化。

2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

- 当 X0.0=ON 时，X1 的装置状态转为 BIN 值储存在 D0，D0 内的数值经 500~5000 范围限制计算后，将结果储存到 D1。



最小输出值	最大输出值	输入值 D0	功能	输出值 D1
500	5000	499	D0<500	500
		5001	D0>5000	5000
		600	500≦D0≦5000	600

补充说明：

S_1>S_2 时，指令不执行，SM0=ON，错误码 SR0=16#2003。

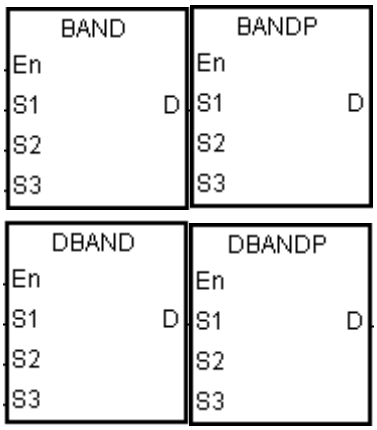
API	指令码			操作数	功能
1222	D	BAND	P	**S_1，S_2，S_3，D**	死区控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
S_3	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
S_3		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 死区下限值

S_2 ： 死区上限值

S_3 ： 输入值

D ： 输出值

指令说明：

1. 输入值 **S_3** 与死区下限值 **S_1** 及死区上限值 **S_2** 作相减，相减后将结果存入 **D**。说明如下：

   若最小输出值 **S_1>**输入值 **S_3**，输出值 **D=S_3-S_1**。

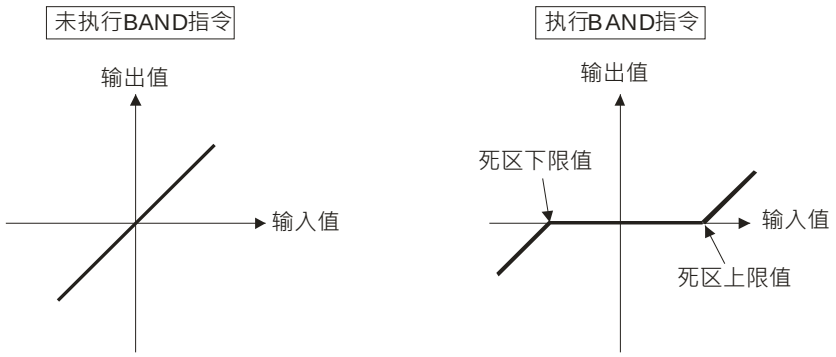
   若最大输出值 **S_2<**输入值 **S_3**，输出值 **D=S_3-S_2**。

   若最小输出值 **S_1**≦输入值 **S_3**≦最大输出值 **S_2**，输出值 **D=**0。

   若最小输出值 **S_1>**最大输出值 **S_2** 则指令不执行，输出值 **D** 不变化。

2. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

3. 图形说明如下：



4. **S_1**、**S_2**和 **S_3**的范围：

- BAND 指令：-32768~32767

  当输出值 **D** 运算结果超过-32768 或 32767，范例说明如下：

  死区下限值 **S_1**=10，输入值 **S_3**=-32768

  输出值 **D**=-32768-10=16#8000-16#000A=16#7FF6=32758

- DBAND 指令：-2147483648~2147483647

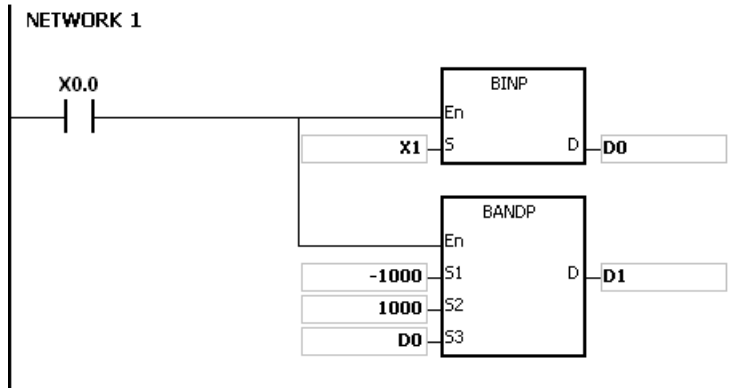
  当输出值 **D** 运算结果超过-2147483648 或 2147483647，范例说明如下：

  死区下限值（**S_1+1**，**S_1**）=1000，输入值（**S_3+1**，**S_3**）=-2147483648

  输出值（**D+1**，**D**）=-2147483648-1000=16#80000000-16#000003E8=16#7FFFFC18
  =2147482648

程序范例一：

当 X0.0 为 ON 时，X1 的 BCD 数值执行从-1000 至 1000 的范围控制，将控制结果储存至 D1 当中。

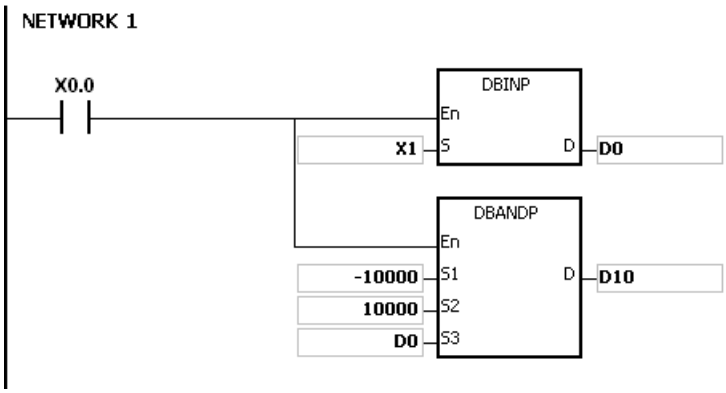


执行结果：

死区下限值	死区上限值	输入值 D0	功能	输出值 D1
-1000	1000	-1200	D0<-1000=>D1=D0-（-1000）	-200
		1200	D0>1000=>D1=D0-1000	200
		500	-1000≦D0≦1000=>D0=0	0

程序范例二：

当 X0.0 为 ON 时，(X2，X1) 的 BCD 数值执行从-10000 至 10000 的范围控制，将控制结果储存至 (D11，D10) 当中。



执行结果：

死区下限值	死区上限值	输入值 (D1，D0)	功能	输出值 (D11，D10)
-10000	10000	-12000	(D1，D0) <-10000 => (D11，D10) = (D1，D0) - (-10000)	-2000
		12000	(D1，D0) >10000 => (D11，D10) = (D1，D0) -10000	2000
		5000	-10000≦ (D1，D0) ≦10000 => (D1，D0) =0	0

补充说明：

S_1>S_2 时，指令不执行，SM0=ON，错误码 SR0=16#2003。

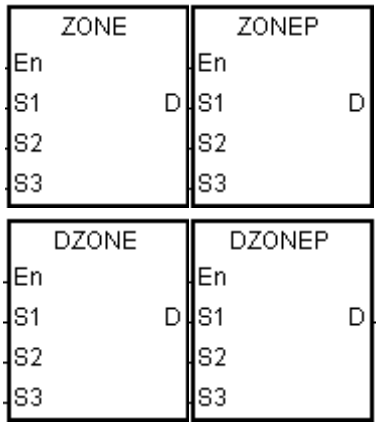
API	指令码			操作数	功能
1223	D	ZONE	P	S_1，S_2，S_3，D	区域控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
S_3	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
S_3		●	●			●	●						
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S_1 ： 负偏差值

S_2 ： 正偏差值

S_3 ： 输入值

D ： 输出值

指令说明：

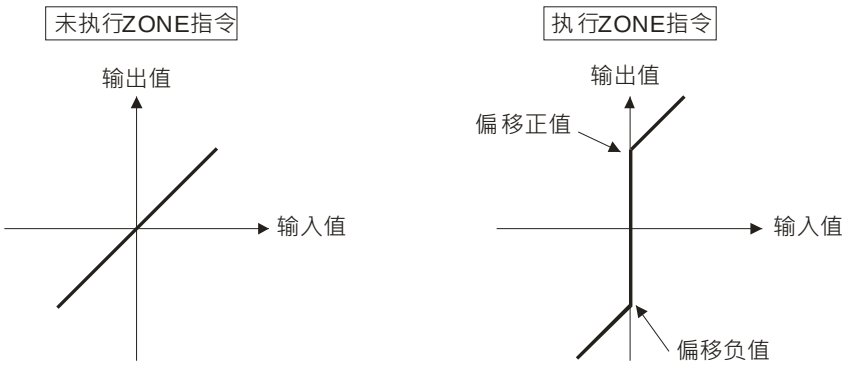
1. 输入值 **S_3** 与负偏差值 **S_1** 或正偏差值 **S_2** 相加，并将结果存入 **D**。说明如下：

   若输入值 **S_3**<0，输出值 **D**=**S_3**+**S_1**。

   若输入值 **S_3**>0，输出值 **D**=**S_3**+**S_2**。

   若输入值 **S_3**=0，输出值 **D**=0。

2. 图形说明如下：



3. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

4. **S_1**、**S_2** 和 **S_3** 的范围：

   - ZONE 指令：-32768~32767

     当输出值 **D** 运算结果超过-32768 或 32767，范例说明如下：

     偏差负值 **S_1**=-100，输入值 **S_3**=-32768

     输出值 **D**=（-32768）+（-100）=16#8000+16#FF9C=16#7F9C=32668

   - DZONE 指令：-2147483648~2147483647

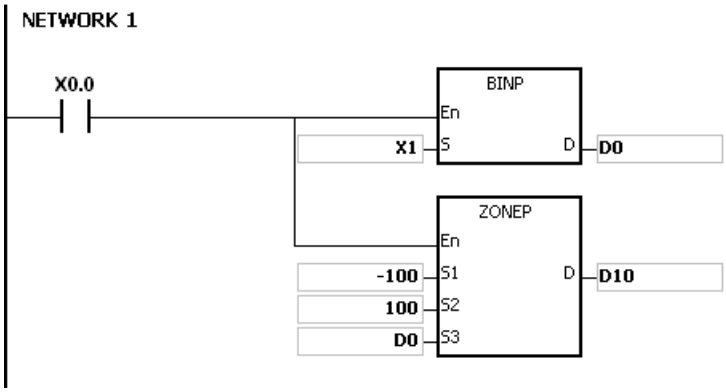
     当输出值 **D** 运算结果超过-2147483648 或 2147483647，范例说明如下：

     偏差负值（**S_1**+1，**S_1**）=-1000，输入值（**S_3**+1，**S_3**）=-2147483648

     输出值（**D**+1，**D**）=-2147483648+（-1000）
     =16#80000000+16#FFFFFC18=16#7FFFFC18=2147482648

程序范例一：

当 X0.0 为 ON 时，X1 的 BCD 数值执行从-100 至 100 的偏差值相加，将偏移结果储存至 D10 当中。

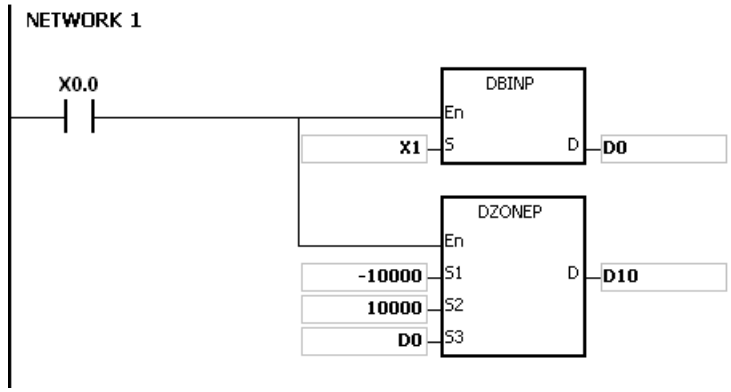


执行结果：

负偏差值	正偏差值	输入值 D0	功能	输出值 D10
-100	100	-10	D0<0=>D10=（-10）+（-100）	-110
		0	D0=0=>D10=0	0
		50	D0>0=>D10=50+100	150

程序范例二：

当 X0.0 为 ON 时，(X2，X1) 的 BCD 数值执行从-10000 至 10000 的偏差值相加，将偏移结果储存至 (D11，D10) 当中。



负偏差值	正偏差值	输入值 (D1，D0)	功能	输出值 (D11，D10)
-10000	10000	-10	(D1，D0) <0 => (D11，D10) = (-10) + (-10000)	-10010
		0	(D1，D0) =0 => (D11，D10) =0	0
		50	(D1，D0) >0 => (D11，D10) =50+10000	10050

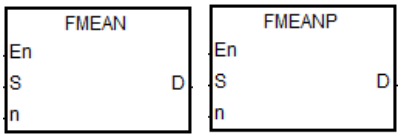
API	指令码			操作数	功能
1224		FMEAN	P	**S**・**D**・**n**	浮点数平均值

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S							●	●								
n							●	●					○	○		
D							●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
n			●				●						
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S ： 欲取平均值之起始装置

D ： 存放平均值之装置

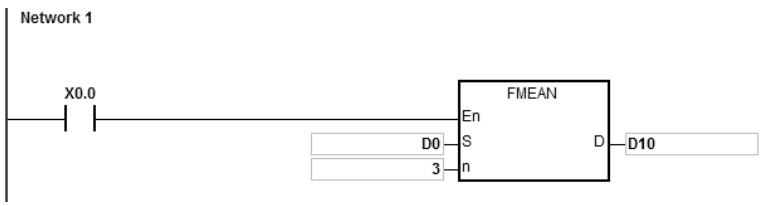
n ： 取平均值之装置个数

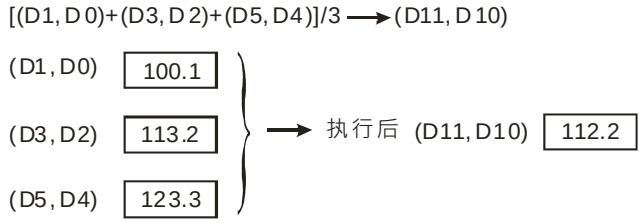
指令说明：

1. 将 **S** 起始之 **n** 个单精度浮点数相加后取平均值存入 **D** 中。
2. **n** 操作数的内容值范围为 1~256。
3. SM600（零标志）、SM601（借位标志）、SM602（进位标志）说明：
   - 运算结果为 0 时，零标志（Zero flag）SM600 为 ON。
   - 若加总的过程中或运算结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000 且借位标志 SM601=ON。
   - 若加总的过程中或运算结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000 且进位标志 SM602=ON。

程序范例：

当 X0.0=ON 时，将（D1，D0）、（D3，D2）、（D5，D4）共 3 笔单精度浮点数全部相加，相加之后再除以 3 以求得平均值并存于指定的（D11，D10）当中。





补充说明：

1. **n**<1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. 若 **S**+2***n**-1 装置，超过装置范围时，SM0=ON，错误码 SR0=16#2003。
3. 当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

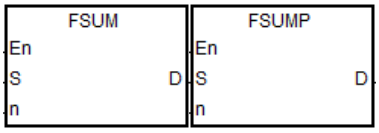
API	指令码			操作数	功能
1225		FSUM	P	**S**、**n**、**D**	浮点数加总

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S							●	●								
n							●	●					○	○		
D							●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
n			●				●						
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



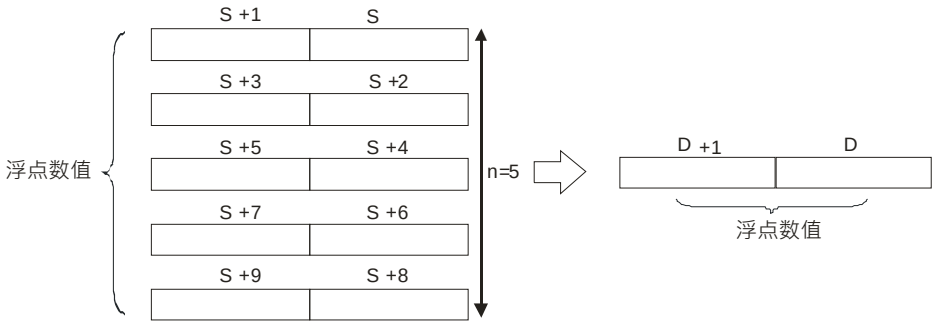
S ： 数据源装置

n ： 数据源长度

D ： 运算结果

指令说明：

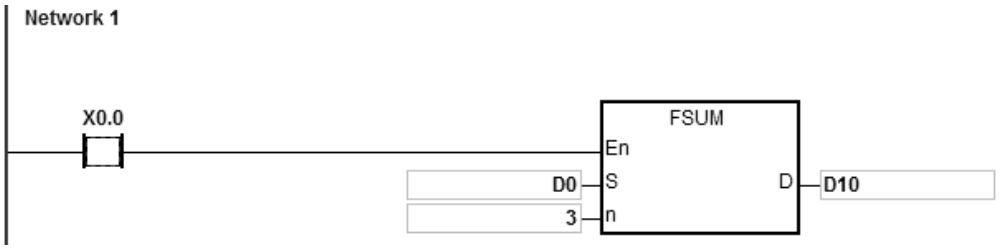
1. 将 **S** 起始之 **n** 笔单精度浮点数相加，存放于 **D** 所指定的寄存器中。



2. **n** 操作数的内容值范围为 1~256。
3. SM600（零标志）、SM601（借位标志）、SM602（进位标志）说明：
   - 运算结果为 0 时，零标志（Zero flag）SM600 为 ON。
   - 若加总的过程中或运算结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000 且借位标志 SM601=ON。
   - 若加总的过程中或运算结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000 且进位标志 SM602=ON。

程序范例：

当执行 FSUM 指令时，会将（D1，D0），（D3，D2），（D5，D4）共 3 笔单精度浮点数相加后，计算结果储存至（D11，D10）当中。



[(D1, D0)+(D3, D2)+(D5, D4)] ⟶ (D11, D10)

(D1, D0) 100.1

(D3, D2) 113.2

(D5, D4) 125.3

⟶ 执行后 (D11, D10) 338.6

补充说明：

1. 当 **n**< 1 或 **n**>256 时，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. 若 **S**+2***n**-1 或 D 装置，超过装置范围时，SM0=ON，错误码 SR0=16#2003。
3. 当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

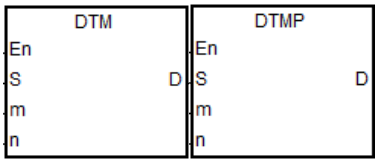
API	指令码		操作数	功能
1226	DTM	P	**S**・**D**・**m**・**n**	数据转换与搬移

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●								
D		●			●	●		●								
m								●					○	○		
n								●					○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
m		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 源数据

D ： 转换与搬移后储存位置

m ： 转换模式选择

n ： 执行长度

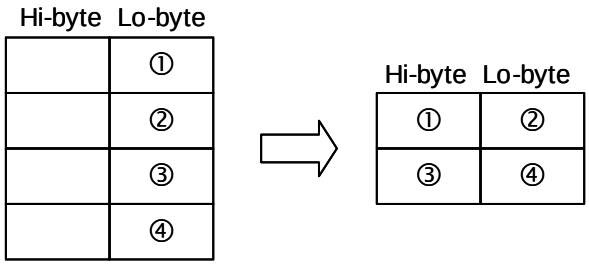
指令说明：

1. **m** 为转换模式选择参数（详见下表说明），详细说明请参照后续各个模式说明，当选择参数代码不是底下介绍的功能，则不会做任何转换或搬移，也无任何错误信息产生。

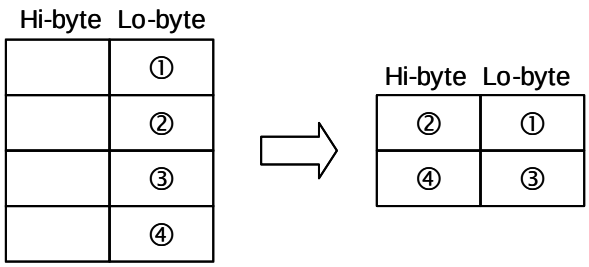
参数	功能说明
0	8 位数据转换为 16 位数据（上 8 位、下 8 位）
1	8 位数据转换为 16 位数据（下 8 位、上 8 位）
2	16 位数据（上 8 位、下 8 位） 转换为 8 位数据
3	16 位数据（下 8 位、上 8 位） 转换为 8 位数据
4	8 位 HEX 数据（上 4 位、下 4 位）转换为 ASCII 数据
5	8 位 HEX 数据（下 4 位、上 4 位）转换为 ASCII 数据
6	8 位 ASCII 数据转换为 HEX 数据（上 4 位、下 4 位）
7	8 位 ASCII 数据转换为 HEX 数据（下 4 位、上 4 位）
18	字符串转浮点数功能 V1.08 版（含）以后支持

参数	功能说明
19	浮点数转字符串功能 V1.08 版（含）以后支持
42	计算相对位置顺时针圆弧插补运行时间
43	计算相对位置逆时针圆弧插补运行时间
44	主机内建输入点滤波时间设定
46	估算相对定位加减速脉冲输出运行时间 V1.08 版（含）以后支持

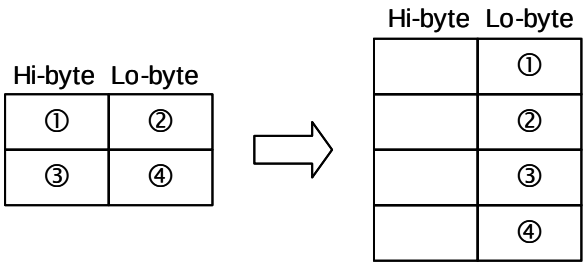
2. **n** 为执行长度设定值，其可设定数值范围为 1~256，若输入超出此范围，将自动以最小或最大值执行。
3. 各个转换与搬移模式说明如下：
   - 当 **m**=0： 假设 **n**=4，则 8 位数据转换为 16 位数据（上 8 位、下 8 位），转换图标说明如下。



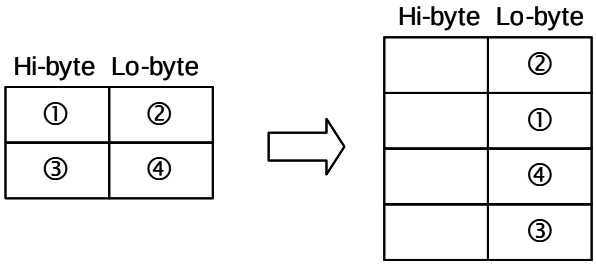
   - 当 **m**=1： 假设 **n**=4，则 8 位数据转换为 16 位数据（下 8 位、上 8 位），转换图标说明如下。



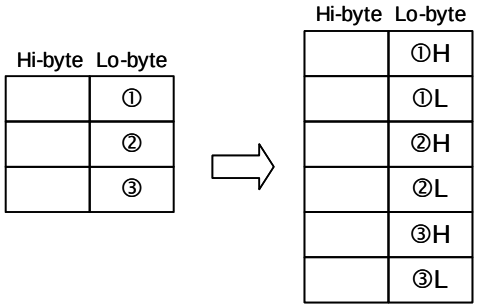
   - 当 **m**=2：假设 **n**=4，则 16 位数据（上 8 位、下 8 位） 转换为 8 位数据，转换图标说明如下



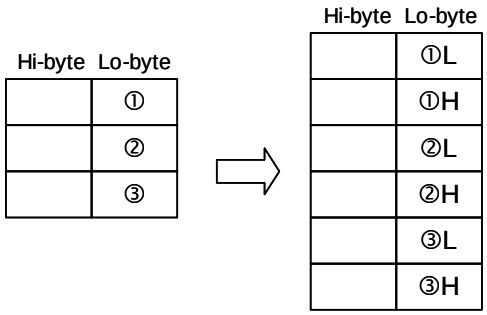
   - 当 **m**=3：假设 **n**=2，则 16 位数据（下 8 位、上 8 位） 转换为 8 位数据，转换图标说明如下



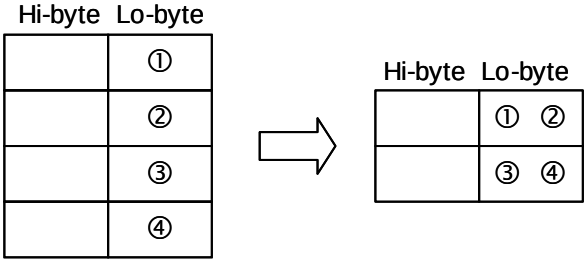
- 当 **m**=4：假设 **n**=3，则 8 位 HEX 数据（上 4 位、下 4 位）转换为 ASCII 数据，转换图标说明如下



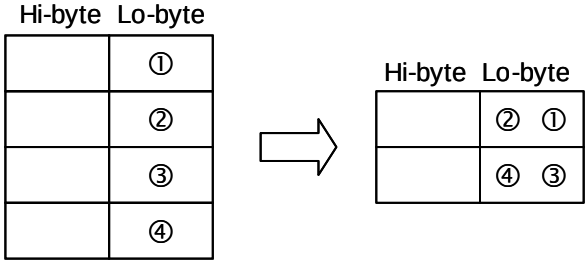
- 当 m=5：假设 **n**=3，则 8 位 HEX 数据（下 4 位、上 4 位）转换为 ASCII 数据，转换图标说明如下



- 当 **m**=6：假设 **n**=4，则 8 位 ASCII 数据转换为 HEX 数据（上 4 位、下 4 位），转换图标说明如下；ASCII 转换数值有 0 ~ 9（0x30~0x39）、A ~ F（0x41~0x46）、a ~ f（0x61~0x66）。



- 当 **m**=7：假设 **n**=4，则 8 位 ASCII 数据转换为 HEX 数据（下 4 位、上 4 位），转换图标说明如下。

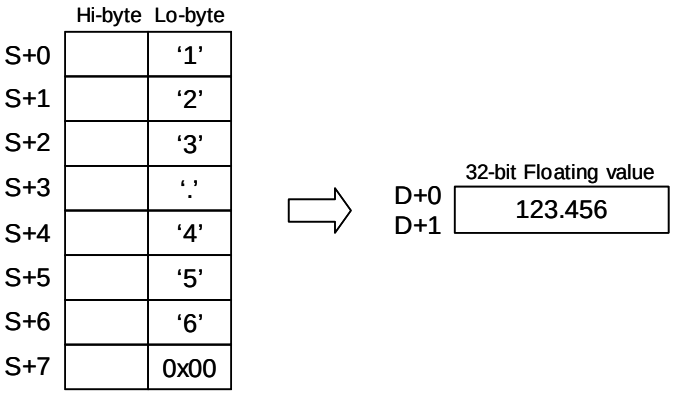


- 当 **m**=18：将来源 S 装置（下 8 位）的 n 个字符（不含小数点）转换成浮点数，并储存在 D 装置。

  注意事项：

  1. 操作数 n 为小数点左右的位数最大设定值（不可超出 8 位数，超出将以 8 位数算），当位数超出最大设定值时，数值将自动被舍去不转换，例如：n = k6，则字符串"123.45678"将会只被转换出 123.456 的数值。
  2. 当来源字符串中有不是 0~9 与小数点的字符出现时，则在小数点之前的会被视为 0，而在小数点之后的会被视为终止符。
  3. 若是一个一个字符转换至 n 个字符中，都未转换到小数点，则自动以 n 个位数之浮点数表示。

     其转换示意图如下：

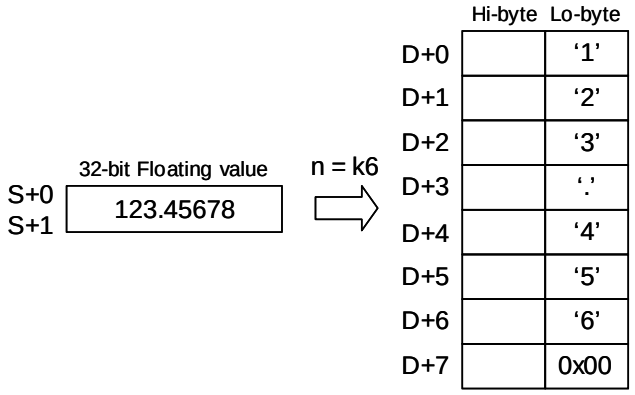


- 当 **m**=19：将来源 S 装置的浮点数转换出最多 n 个字符（不含小数点）的字符串，并储存在 D 装置。

  注意事项：

  1. 操作数 n 为小数点左右的位数最大设定值（不可超出 8 位数，超出将以 8 位数算），当位数超出最大设定值时，数值将自动被舍去不转换，例如：n = k6，则浮点数 F123.45678 将会只被转换出 "123.456" 的字符串。
  2. 当来源数值大于 n 位数的数值时，则在数值最左边的 n 个位数会被转换，例如：浮点数为 F123456.78，而 n = k4，则只会被转换出 "1234" 的字符串。

     其转换示意图如下：



- 当 **m**=42：

S1 操作数：S1+0，S1+1 为 X 轴目标坐标（相对位置）。

S1+2，S1+3 为 Y 轴目标坐标（相对位置）。

S1+4，S1+5 为圆心偏移量（整数类型）或是圆心角度（浮点数类型）。
S1+6，S1+7 为目标速度参考值。（1~200000Hz，超出上下限，以上下限处理）
S1+8 为功能选择（K0，K1，K2 为圆心偏移量模式分别以 10 度，5 度，1 度为单位移动；K10，K11，K12 为圆心角度模式分别以 10 度、5 度、1 度为单位移动）。

D 操作数：D+0，D+1 为估算相对位置顺时针圆弧插补运行时间，单位 ms。

m 操作数：功能码 K42

n 操作数：保留

- 当 **m**=43：

S1 操作数：S1+0，S1+1 为 X 轴目标坐标（相对位置）。
S1+2，S1+3 为 Y 轴目标坐标（相对位置）。
S1+4，S1+5 为圆心偏移量（整数类型）或是圆心角度（浮点数类型）。
S1+6，S1+7 为目标速度参考值。（1~200000Hz，超出上下限，以上下限处理）
S1+8 为功能选择（K0，K1，K2 为圆心偏移量模式分别以 10 度，5 度，1 度为单位移动；K10，K11，K12 为圆心角度模式分别以 10 度、5 度、1 度为单位移动）

D 操作数：D+0，D+1 为估算相对位置逆时针圆弧插补运行时间，单位 ms。

m 操作数：功能码 K43

n 操作数：保留

- 当 **m**=44：

S1 操作数：主机输入点起始编号（K0~K15 ➔ X0.0~X0.15）。

D 操作数：滤波时间，单位 us（范围 0~20000）。

m 操作数：功能码 K44

n 操作数：点数

- 当 **m**=46：计算相对位置加减速脉冲运行时间

S1 操作数：S1+0，S1+1 为脉冲输出数目（相对位置）。
S1+2，S1+3 为脉冲输出频率。（AS324MT Y0.0/Y0.2：1~4MHz，其余 1~200KHz）
S1+4 为脉冲输出装置（K0~K11：Y0.0~Y0.11）

D 操作数：D+0，D+1 为估算相对位置加减速脉冲输出运行时间，单位 ms。

m 操作数：功能码 K46

n 操作数：保留

补充说明：

S 与 D 组件建议采用 word 类型的矩阵变量。

API	指令码			操作数	功能
1228		ZSET	P	**SBit，EBit**	区域设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
SBit		●	●	●				●		○						
EBit		●	●	●				●		○						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
SBit	●												
EBit	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

SBit ： 区域设定起始装置

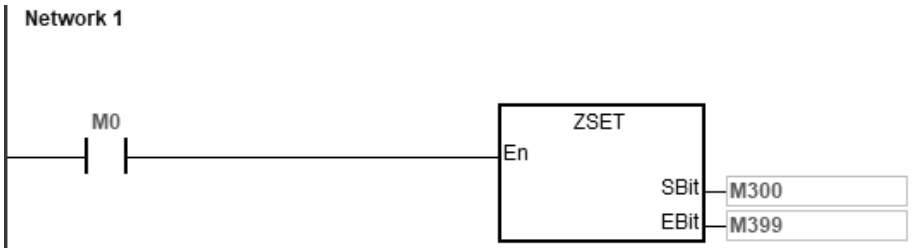
EBit ： 区域设定结束装置

指令说明：

1. 设定 **SBit~EBit** 间的位为 ON，其起始与结束装置类别必须相同，才可进行设定动作。
2. 当 **SBit** 操作数编号 > **EBit** 操作数编号时，只有 **EBit** 指定之操作数被设定。
3. 此指令只对位型装置（T，C ，HC 组件除外）或 BOOL 数据类型进行设定。

程序范例：

1. 当 M0 为 ON 时，辅助继电器 M300~M399 全部被设定成 ON。



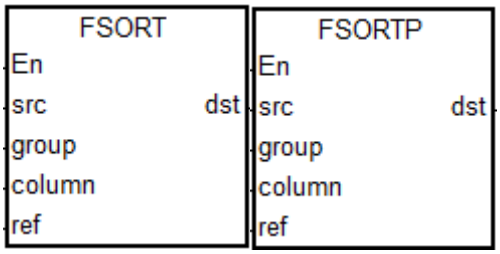
API	指令码			操作数	功能
1229	F	SORT	P	**src ~ ref**	浮点数数据排序

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
src								●								
group								●					○	○		
column								●					○	○		
ref								●					○	○		
dst								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
src									●				
group			●				●						
column			●				●						
ref			●				●						
dst									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



src ： 原始数据区块之起始装置

group ： 被排序之数据组数

column ： 每笔数据之字段数

ref ： 数据排序的参考值

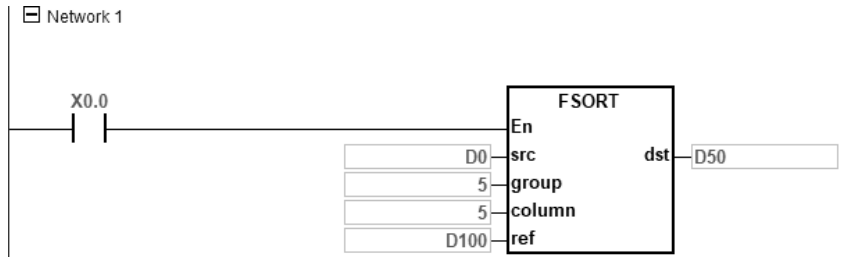
dst ： 存放排序结果数据区块之起始装置

指令说明：

1. 排序结果显示于 **D** 所指定的起始号码开始算的 **group×column**x**2** 个寄存器当中，当 **S** 与 **D** 指定同一个寄存器时，其排序结果将与原来被排序的数据 **S** 内容相同。
2. group=1~32，**column**=1~6，**ref**=1~ **column**。
3. SM604=OFF 时为升幂排序，SM604=ON 时为降序排序。
4. 建议使用脉冲型执行指令，避免一直重复排序。

程序范例：

1. 将 SM604=OFF，当 X0.0=OFF→ON 时，指定执行数据进行升幂排序作业。



2. 排序数据构成例

资料数：column x 2 个

数据字段

行 / 列	1	2	3	4	5
	学生编号	国文	英文	数学	理化
1	(D0,D1)1.0	(D10,D11) 90.0	(D20,D21) 75.0	(D30,D31) 66.0	(D40,D41) 79.0
2	(D2,D3)2.0	(D12,D13) 55.0	(D22,D23) 65.0	(D32,D33) 54.0	(D42,D43) 63.0
3	(D4,D4)3.0	(D14,D15) 80.0	(D24,D25) 98.0	(D34,D35) 89.0	(D44,D45) 90.0
4	(D6,D7)4.0	(D16,D17) 70.0	(D26,D27) 60.0	(D36,D37) 99.0	(D46,D47) 50.0
5	(D8,D9)5.0	(D18,D19) 95.0	(D28,D29) 79.0	(D38,D39) 75.0	(D48,D49) 69.0

资料个数：group x 2 个

3. D100=3 时的排序后数据

资料数：column x 2 个

数据字段

行 / 列	1	2	3	4	5
	学生编号	国文	英文	数学	理化
1	(D50,D51) 4.0	(D60,D61) 70.0	(D70,D71) 60.0	(D80,D81) 99.0	(D90,D91)50.0
2	(D52,D53) 2.0	(D62,D63) 55.0	(D72,D73) 65.0	(D82,D83) 54.0	(D92,D93)63.0
3	(D54,D55) 1.0	(D64,D65) 90.0	(D74,D75) 75.0	(D84,D85) 66.0	(D94,D95)79.0
4	(D56,D57) 5.0	(D66,D67) 95.0	(D76,D77) 79.0	(D86,D87) 75.0	(D96,D97)69.0
5	(D58,D59) 3.0	(D68,D69) 80.0	(D78,D79) 98.0	(D88,D89) 89.0	(D98,D99)90.0

资料个数：group x 2 个

4. D100=5 时的排序后数据。

资料数：column x 2 个

数据字段

资料个数：group x 2 个	行 / 列	1	2	3	4	5
		学生编号	国文	英文	数学	理化
	1	(D50,D51) 4.0	(D60,D61) 70.0	(D70,D71) 60.0	(D80,D81) 99.0	(D90,D91) 50.0
	2	(D52,D53) 2.0	(D62,D63) 55.0	(D72,D73) 65.0	(D82,D83) 54.0	(D92,D93) 63.0
	3	(D54,D55) 5.0	(D64,D65) 95.0	(D74,D75) 79.0	(D84,D85) 75.0	(D94,D95) 69.0
	4	(D56,D57) 1.0	(D66,D67) 90.0	(D76,D77) 75.0	(D86,D87) 66.0	(D96,D97) 79.0
	5	(D58,D59) 3.0	(D68,D69) 80.0	(D78,D79) 98.0	(D88,D89) 89.0	(D98,D99) 90.0

补充说明：

1. 装置超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. $\mathbf{m_1}$、$\mathbf{m_2}$ 或 **n** 超出范围时，指令不执行，SM0=ON，错误码 SR0=16#200B。
3. 当 src 内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

6.14 结构建立指令

6.14.1 结构建立指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1300	FOR	–	–	循环回路起始
1301	NEXT	–	–	循环回路结束
1302	BREAK	–	–	强制结束 FOR-NEXT 循环

6.14.2 结构建立指令说明

API	指令码	操作数	功能
1300	FOR	**S**	循环回路起始

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●			●	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：

S ：回路重复执行的次数

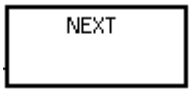
指令说明：

详细内容说明请参考 API1301 NEXT 指令的指令说明。

API	指令码	操作数	功能
1301	NEXT	-	循环回路结束

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：

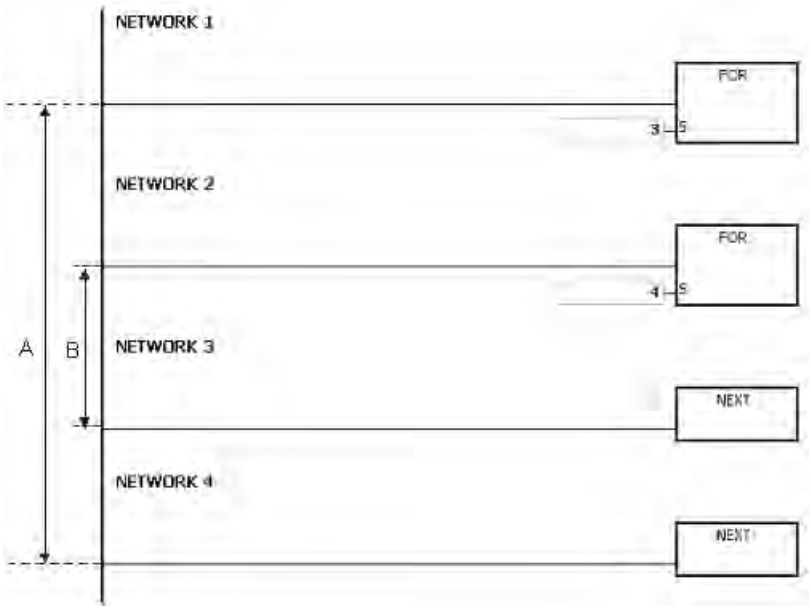


指令说明：

1. 由 FOR 指令指定 FOR~NEXT 循环来回执行 N 次后跳出 FOR~NEXT 循环往下继续执行。
2. 指定次数范围 N=1~32,767，当指定次数范围 N≦1 时，都视为是 1。
3. 当不执行 FOR~NEXT 回路时，可使用 CJ 指令来跳出回路。
4. 下列情形会产生错误：
   - NEXT 指令在 FOR 指令之前。
   - 有 FOR 指令没有 NEXT 指令。
   - FEND 或 END 指令之后有 NEXT 指令时。
   - FOR~NEXT 指令个数不同时。
5. 循环式 FOR~NEXT 回路最多可使用 32 层，但要注意回路次数过多时，会使 PLC 扫描时间增加有可能造成逾时监视定时器动作，而导致错误产生。可使用 WDT 指令来改善。

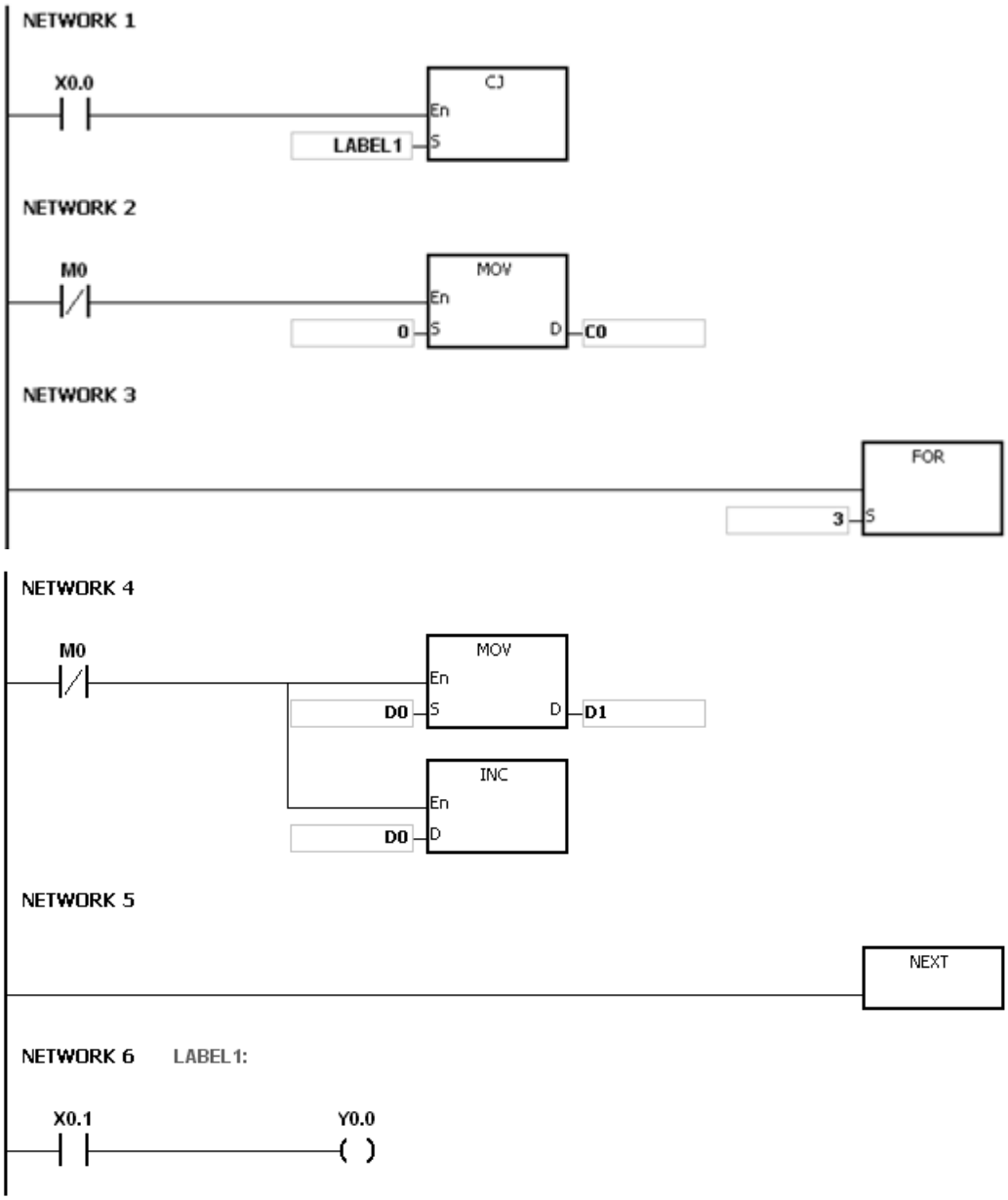
程序范例一：

A 程序执行 3 次后在到 NEXT 指令以后的程序继续执行。而 A 程序每执行一次 B 程序会执行四次，所以 B 程序合计共执行 3×4 = 12 次。



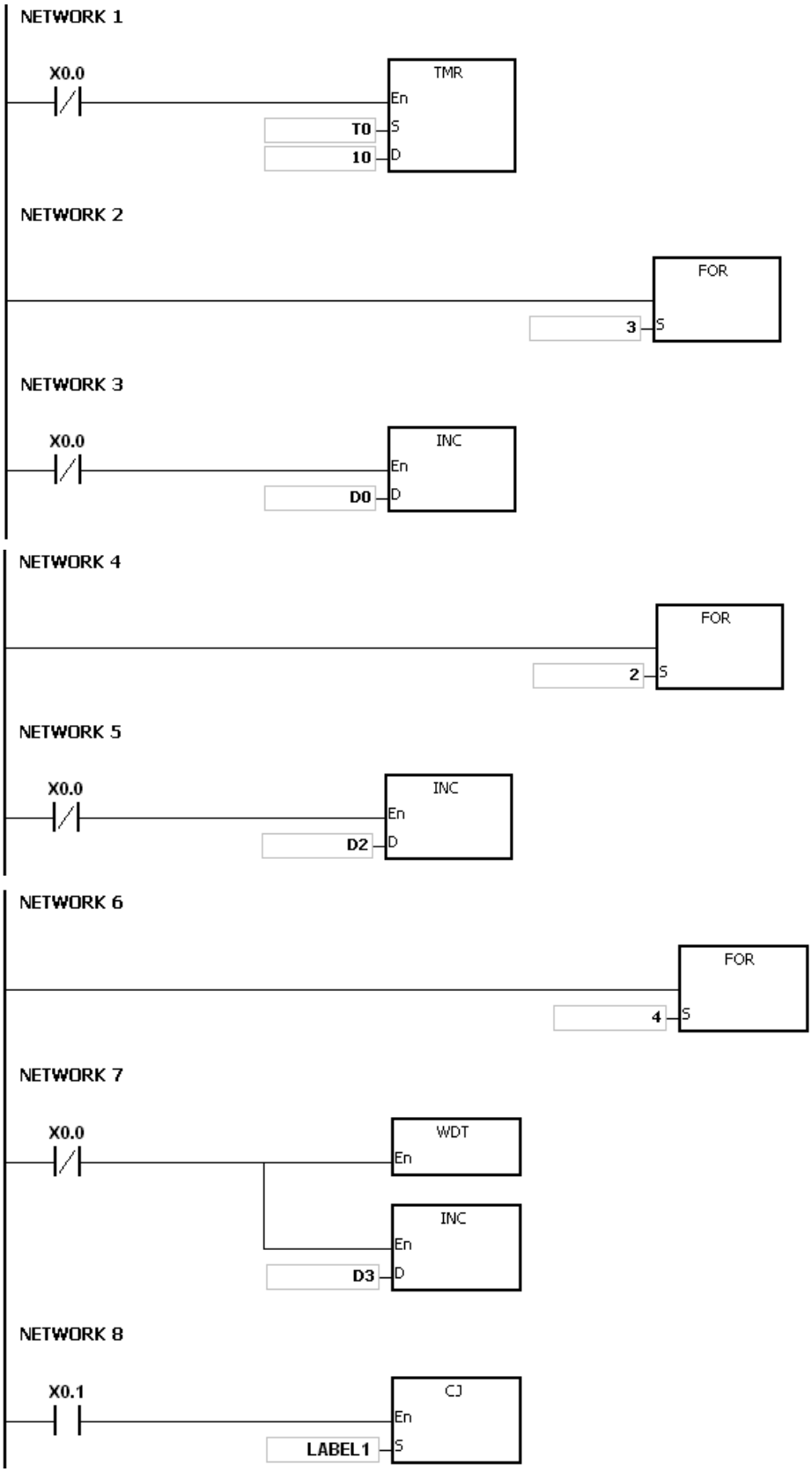
程序范例二：

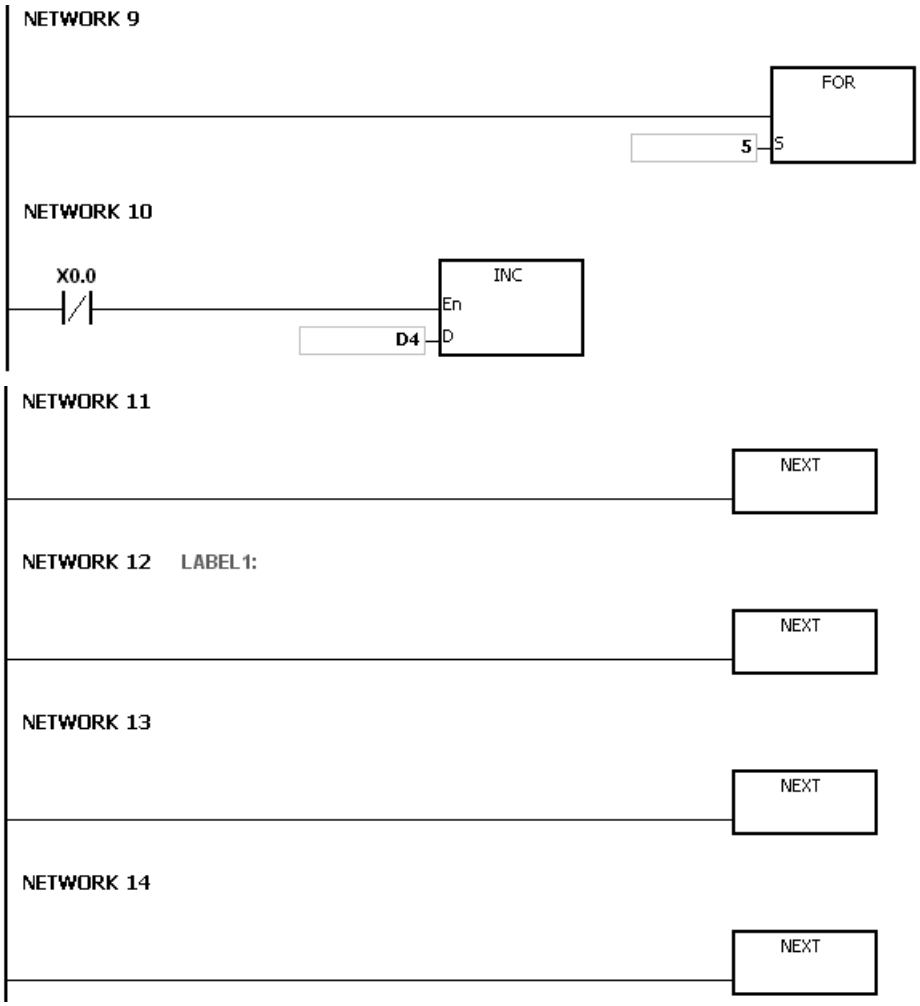
当 X0.0=OFF 时，PLC 会执行 FOR~NEXT 之间的程序，当 X0.0=ON 时，CJ 指令执行跳转至 LABEL 1：即 NETWORK 6 处，FOR~NEXT 之间的程序 NETWORK 4~ NETWORK 5 跳过不执行。



程序范例三：

当不执行 FOR~NEXT 时，可使用 CJ 指令来跳转。NETWORK 8 的 X0.1=ON 时，CJ 指令执行跳转至 Label 1：即 NETWORK 12 处，NETWORK 9~NETWORK 11 跳过不执行。


NETWORK 1
X0.0
TMR
En
T0
S
10
D
NETWORK 2
FOR
3
S
NETWORK 3
X0.0
INC
En
D0
D
NETWORK 4
FOR
2
S
NETWORK 5
X0.0
INC
En
D2
D
NETWORK 6
FOR
4
S
NETWORK 7
X0.0
WDT
En
INC
En
D3
D
NETWORK 8
X0.1
CJ
En
LABEL1
S



补充说明：

指定标签 P（LABEL）在 ISPSoft 中的使用方式，请参考 ISPSoft 的使用手册说明。

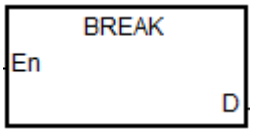
API	指令码	操作数	功能
1302	BREAK	D	强制结束 FOR-NEXT 循环

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



D ： 储存剩余未执行完的循环数

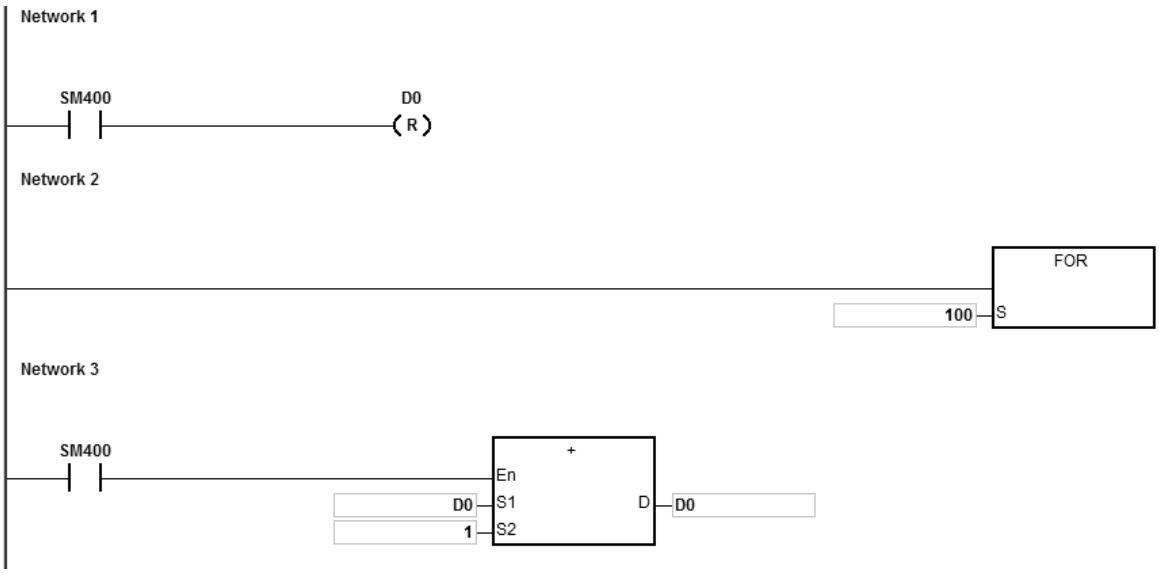
：

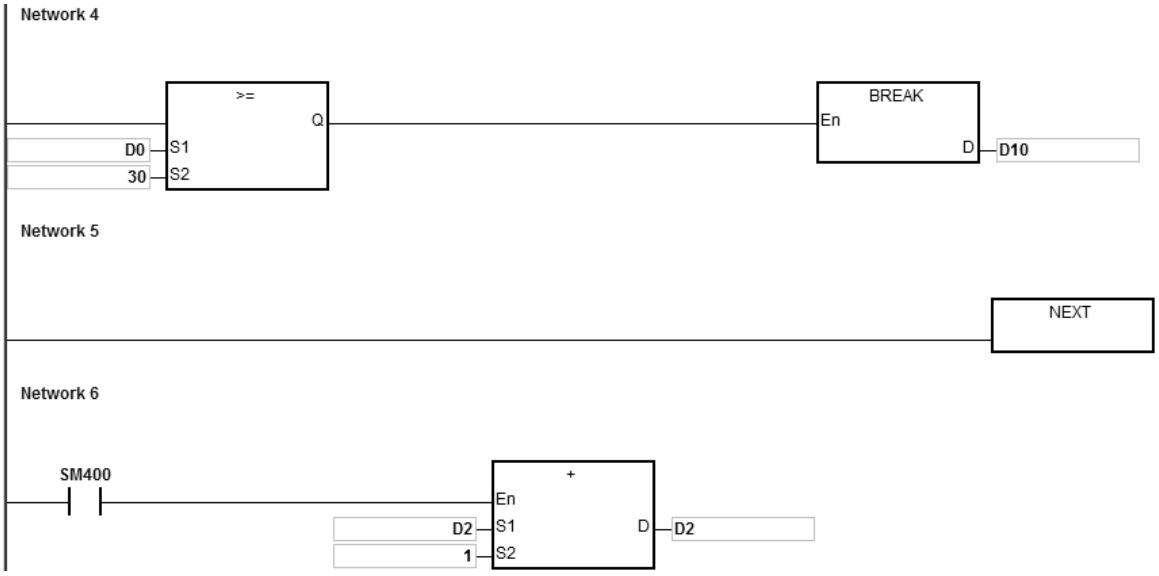
指令说明：

1. 强制结束 FOR-NEXT 的循环，将尚未执行完的循环数存放到 **D**，并跳至最近 NEXT 指令的下一个指令执行。
2. BREAK 指令执行时，将 FOR-NEXT 剩余尚未执行的循环数存放到 **D** 中，包含 BREAK 指令正在执行时的这一次。
3. BREAK 指令在 FOR-NEXT 循环第一次时被启动，则无法立即跳转至 NEXT 下个指令执行，但若是在 FOR-NEXT 执行到第二次以上的循环时，即可立即跳转。

程序范例：

程序开始执行 100 次 FOR-NEXT 循环的程序 D0 内容加 1，当 D0 内容等于 30 时，将强制结束 FOR-NEXT 循环，将剩余的循环数 71 储存至 D10，并跳至 NETWORK 6 处执行 D2 内容加 1。





补充说明：

1. BREAK 指令写在 FOR-NEXT 之外，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2017。
2. 指定标签 P（LABEL）在 ISPSoft 中的使用方式，请参考 ISPSoft 的使用手册说明。

6.15 模块的数据读写指令

6.15.1 模块的数据读写指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1400	FROM	DFROM	✓	扩展模块 CR 数据读出
1401	TO	DTO	✓	扩展模块 CR 数据写入
1402	–	DPUCONF	✓	PU 模块输出控制参数设定
1403	PUSTAT	–	–	PU 模块输出状态读回
1404	–	DPUPLS	–	PU 模块脉冲输出（无加减速）
1405	–	DPUDRI	–	PU 模块相对定位输出（附加减速）
1406	–	DPUDRA	–	PU 模块绝对寻址输出（附加减速）
1407	–	DPUZRN	–	PU 模块原点回归
1408	–	DPUJOG	–	PU 模块寸动输出
1409	–	DPUMPG	–	PU 模块手摇轮控制输出
1410	–	DPUCNT	–	PU 模块高速计数器功能
1415	–	DLCCAL	–	LC 模块通道校正
1416	–	DLCWEI	–	LC 模块重量值读取
1417	–	DMPID	–	温度模块 PID 控制
1418	–	DHCCNT	–	HC 模块计数器控制
1419	–	DHCCAP	–	HC 模块计数值获取
1420	HCDO	–	–	HC 模块输出点控制
1421	–	DHCCMP	–	HC 模块比较输出
1422	–	DHCCMPT	–	HC 模块表格比较输出
1423	–	DHCMEAS	–	HC 模块频率及转速测量
1424	ADLOG	DADLOG	–	模拟输入模块数据记录
1425	ADPEAK	DADPEAK	–	模拟输入模块峰值记录
1426	IOLINKR	–	–	IO-Link 装置参数读出
1427	IOLINKW	–	–	IO-Link 装置参数写入
1430	MLPF	–	–	模拟输入低通滤波器

6.15.2 模块的数据读写指令说明

API	指令码			操作数	功能
1400	D	FROM	P	m_1 . m_2 . m_3 , D_1 , D_2 , n	扩展模块 CR 数据读出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
m_1					●	●	●	●	●		○	○	○	○		
m_2					●	●	●	●	●		○	○	○	○		
m_3					●	●	●	●	●		○	○	○	○		
D_1					●	●	●	●								
D_2					●	●	●	●								
n					●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
m_1		●	●			●	●						
m_2		●	●			●	●						
m_3		●	●			●	●						
D_1		●	●			●	●						
D_2		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：

m_1 ： 主机或远程模块编号

m_2 ： 扩展模块顺序编号

m_3 ： 读取模块之 CR 编号

D_1 ： 存放读取数据的位置

D_2 ： 存放错误代码的位置

n ： 一次读取之数据笔数

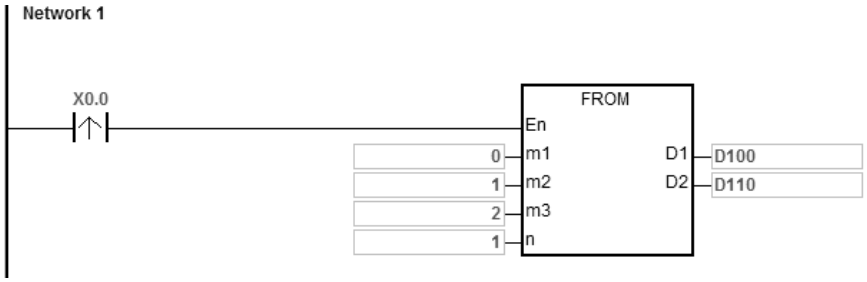
指令说明：

1. 此指令可读取模块之控制寄存器(CR)数据。详细寄存器列表请参考 AS 系列模块手册说明。
2. **m_1** 为 0 时，表示主机，**m_1**=1~16 为远程模块之编号。
3. **m_2** 为连接于主机或远程模块右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
4. **m_3** 欲读取模块之 CR（Controlled Register）编号。

5. **D_2**：开始执行指令时，**D_2** 会被设定为 0（表示无错误），当有通讯错误时，**D_2** 为非 0。详细错误代码请参考补充说明。当指令不执行时，则 **D_2** 不会被写入通讯错误代码。
6. **n**：读取之数据笔数；**n**=1~8。
7. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。

程序范例：

当启动 X0.0=OFF→ON 时，会执行 FROM 的应用指令，读取置放于主机右侧第一个模块的 CR 编号 2，并且将接收的数值储存到 D100，因为执行过程中无错误发生，故 D110=16#0000。



补充说明：

1. **m_1** 与 **m_2** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 当 **D_1**~ **D_1**+**n**-1 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **n** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 由于使用 FROM 会降低主机的执行效能，所以建议使用脉冲型指令单次触发执行，如上述范例。
5. 当指令执行后，若模块响应有错误时，其错误代码将会被记录于 **D_2** 内，详细错误代码说明如下：

错误代码	说明
16#1400	模块无此 CR 编号
16#1401	模块存入数值不合法
16#1402	模块无响应，通讯逾时

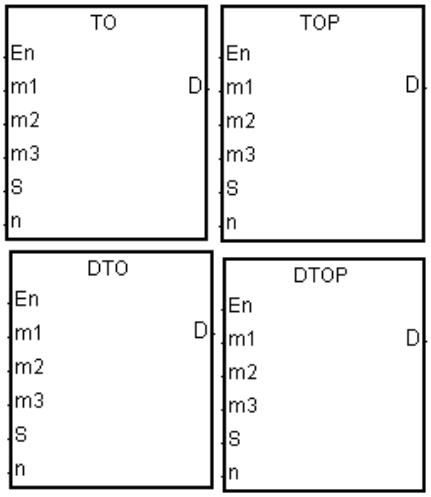
API	指令码			操作数	功能
1401	D	TO	P	**m_1**、**m_2**、**m_3**、**S**、**D**、**n**	扩展模块 CR 数据写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
m_1					●	●	●	●	●		○	○	○	○		
m_2					●	●	●	●	●		○	○	○	○		
m_3					●	●	●	●	●		○	○	○	○		
S					●	●	●	●	●				○	○		
D					●	●	●	●								
n					●	●	●	●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
m_1		●	●			●	●						
m_2		●	●			●	●						
m_3		●	●			●	●						
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



m_1 ： 主机或远程模块编号

m_2 ： 扩展模块顺序编号

m_3 ： 读取模块之 CR 编号

S ： 写入源数据的位置

D ： 存放错误代码的位置

n ： 一次写入之数据笔数

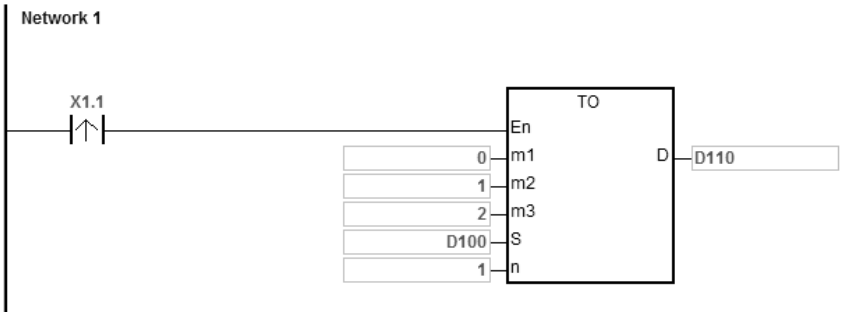
指令说明：

1. 写入数据至扩展模块之控制寄存器(CR)内。详细寄存器列表请参考 AS 系列模块手册说明。
2. **m_1** 为 0 时，表示主机，**m_1**=1~16 为远程模块之编号。
3. **m_2** 为连接于主机或远程模块右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
4. **m_3** 被写入模块之 CR（Controlled Register）编号。

5. **D_2**：开始执行指令时，**D_2** 会被设定为 0（表示无错误），当有通讯错误时，**D_2** 为非 0。详细错误代码请参考补充说明。当指令不执行时，则 **D_2** 不会被写入通讯错误代码。
6. **n**：写入之数据笔数；**n**=1~8。
7. 32 位指令才可以使用 HC 装置，但不可使用 E 装置。
8. 当 S 为 K, 16#数值时，会传送 **n** 个相同数值至指定的模块。例如：**S** 为 16#0001，**n** 为 3，则传送三个 16#0001 给模块。

程序范例：

当启动 X1.1=OFF→ON 时，会执行 TO 的应用指令，并 D100 内容写入到主机旁第 1 台模块的 CR 编号 2 内。因为执行无误所以 D110=16#0000。



补充说明：

1. **m_1** 与 **m_2** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 当 **D_1**~ **D_1**+**n**-1 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **n** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#200B。
4. 由于使用 TO 指令会降低主机的执行效能，所以建议使用脉冲型指令单次触发执行，如上述范例。
5. 当指令执行后，若模块响应有错误时，其错误代码将会被记录于 **D_2** 内，详细错误代码说明如下：

错误代码	说明
16#1400	模块无此 CR 编号
16#1401	模块存入数值不合法
16#1402	模块无响应，通讯逾时

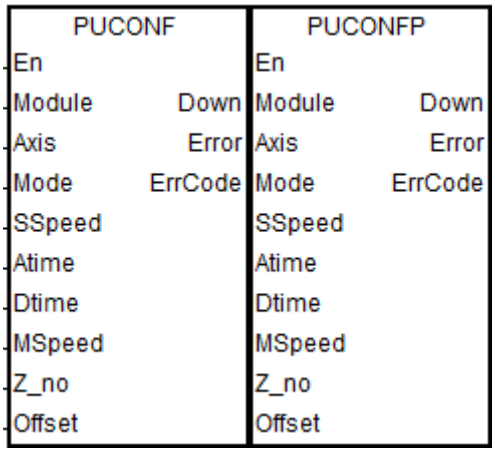
API	指令码			操作数	功能
1402	D	PUCONF	P	**Module ~ Error，ErrCode**	PU 模块输出控制参数设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Axis								●					○	○		
Mode								●					○	○		
SSpeed								●					○	○		
Atime								●					○	○		
Dtime								●					○	○		
MSpeed								●					○	○		
Z_no								●					○	○		
Offset								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
Mode		●				●							
SSpeed		●				●							
Atime		●				●							
Dtime		●				●							
Mspeed			●				●						
Z_no		●				●							
Offset		●				●							
Done	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
AS		AS

符号：



Module ： 模块编号

Axis ： 输出轴编号

Mode ： 输出模式设定

SSpeed ： 启动/结束速度

Atime ： 加速时间

Dtime ： 减速时间

MSpeed ： 最大输出频率

Z_no ： 回到原点后寻找 Z 相次数

Offset ： 回到原点后指定输出个数

Done ： 完成设定标志

Error ： 设定错误标志

ErrCode ： 错误代码

指令说明：

1. 主机韧体版本 V1.08.00（含）之后，支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。此指令设定时机点为 En 由 OFF →ON 时。
2. **Module** 为连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。此 PU 专属指令仅限于主机右侧之 PU 模块使用，不适用连接在远程模块右侧之 PU 模块。若指定之模块不为 PU 模块时，则 **Error** 错误标志会被设为 On。
3. **Axis** 为指定输出之 PU 模块的轴编号，输入数值为 1~4 分别表示指定 PU 模块的轴 1~轴 4 输出，若该 PU 模块无此轴编号时，则 **Error** 错误标志会被设为 On。其轴编号对应的输出点组合如下表所示：

PU 模块名称	轴 1 组合	轴 2 组合	轴 3 组合	轴 4 组合
AS02PU	Y0.0 / Y0.1	Y0.2 / Y0.3	NA	NA
AS04PU	Y0.0 / Y0.1	Y0.2 / Y0.3	Y0.4 / Y0.5	Y0.6 / Y0.7

4. **Mode** 为设定该输出轴的输出模式选择参数，其设定数值如下表所示：

输出模式数值	模式说明	备注
0	单点脉冲输出（仅偶数点输出）	如：Y0.0 或 Y0.2 输出
1	脉冲（偶数点） + 方向（奇数点）	如：Y0.0 为脉冲，Y0.1 为方向；当方向为正，则 Y0.1 为 OFF；当方向为反，则 Y0.1 为 ON。
2	CW （偶数点） + CCW（奇数点）	如：Y0.0 为 CW（正方向），Y0.1 为 CCW（反方向）
3	A 相（偶数点） + B 相（奇数点）	如：Y0.0 为 A，Y0.1 为 B；当 A 领先 B 表示正方向输出；当 B 领先 A 表示反方向输出。
其它数值	自动转为模式 1（默认值）	

5. **SSpeed**~ **Offset** 参数说明如下表所示，且皆为非停电保持数值；若设定超出数值范围，则自动以最小或最大值设定。

参数编号	功能说明	可输入数值范围	默认值	备注
SSpeed	启动/结束速度	0 ~ 10,000（单位 Hz）	100	
Atime	加速时间	0 ~ 10,000（单位 ms）	100	
Dtime	减速时间	0 ~ 10,000（单位 ms）	100	
MSpeed	最大输出频率	AS02PU：100 ~ 200,000（单位 Hz）	200K	32bit 数值
		AS04PU：100 ~ 100,000（单位 Hz）	100K	
Z_no	原点回归功能寻找 Z 相信号次数	-100 ~ 100（单位次数）	0	0 为不启动
Offset	原点回归功能完成且寻找完 Z 相之后，再输出偏移位置	-10,000 ~ 10,000（单位个数）	0	0 为不启动

6. **Done** 为指定之 PU 模块的输出轴，已经被设定参数完成标志，当此完成标志为 On 时，表示参数设定成功，用户可依照此标志状态，进行后续定位输出功能。此 **Done** 标志清除动作需由用户自行清除，此指令只会在完成设定时，设定一次此标志。
7. **Error** 为指定之 PU 模块的输出轴，设定参数发生错误之标志，由于大多参数范围皆由 PLC 自动过滤，因此若有此错误标志发生，则表示无指定之 PU 模块，或者 PU 的模块编号错误，又或者输出轴编号错误。
8. 此参数设定指令为脉冲执行型指令，用户条件接点即使采用 A 接点方式，此指令也只会在启动那一次设定 PU 模块的参数，因此若有变更该轴参数值时，请重新启动指令进行参数重新设定。
9. 由于此设定参数是通过模块通讯命令方式下达，因此每次修改参数时，请确认设定结果的 **Done** 或 **Error** 标志之后，再进行相关后续输出动作。
10. **ErrCode** 为错误代码，其说明请参考下表：

错误代码	说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时

程序范例：请参考 API 1405 DPUDRI 指令范例说明。

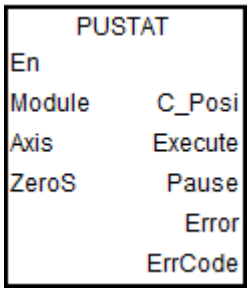
API	指令码			操作数	功能
1403		PUSTAT		**Module ~ ErrCode**	PU 模块输出状态读回

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Axis								●					○	○		
ZeroS	●	●	●	●				●								
C_Posi								●								
Execute		●	●	●				●								
Pause		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
ZeroS	●												
C_Posi			●				●						
Execute	●												
Pause	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Module ： 模块编号

Axis ： 输出轴编号

ZeroS ： 当前输出位置清除为 0

C_Posi ： 当前输出位置

Execute ： 输出执行中标志

Pause ： 输出暂停中标志

Error ： 读回错误标志

ErrCode ： 错误代码

指令说明：

1. 主机韧体版本 V1.08.00（含）之后支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。此指令 **En** 须为 ON 时，才能持续每次被扫描时更新指定轴的状态。

2. **Module** 为连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。此 PU 专属指令仅限于主机右侧之 PU 模块使用，不适用连接在远程模块右侧之 PU 模块。若指定之模块不为 PU 模块时，则 **Error** 错误标志会被设为 On。
3. **Axis** 为指定输出之 PU 模块的轴数编号，输入数值为 1~4 分别表示指定 PU 模块的轴 1~轴 4 输出，若该 PU 模块无此轴数编号时，则 **Error** 错误标志会被设为 On。
4. **C_Posi** 为指定 PU 模块之输出轴的当前位置，此数值为停电保持，并且保存于 PU 模块内。若要清除此数值时，请在指令启动时，设定 **ZeroS** 标志（Off → On）清除为 0。
5. **Execute** 为指定 PU 模块之输出轴的输出执行中标志（只读），当此标志为 On，即表示正在输出中；当此标志为 Off，即表示该输出轴未被使用中，可接受下一个输出指令。
6. **Pause** 为指定 PU 模块之输出轴的输出暂停中标志（只读），当此标志为 On，即表示正在暂停中，当前速度为 0，且当前位置还未输出到达指定输出之目标位置。若用户恢复输出，则此标志将自动被清除。注意：当暂停标志为 On 时，**Execute** 输出执行中标志也会持续为 On。
7. **Error** 为指定之 PU 模块，发生读取错误之标志（只读），当此错误发生时，请参考 **ErrCode** 错误代码说明。
8. **Execute**、**Pause**、**Error** 标志在 PUSTAT 指令下达暂停之后，其属性将变为只读型标志，此时用户无法修改其状态。当 PUSTAT 指令被关闭时，才可执行设定或清除 **Execute**、**Pause**、**Error** 标志。
9. **ErrCode** 为错误代码，其说明请参考下表：

错误代码	说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	PU 模块无此输出轴编号
16#1404	PU 模块之输出频率不合法
16#1405	PU 模块指定之输出轴正在输出当中，不允许重复指定输出

程序范例：请参考 API 1405 DPUDRI 指令范例说明。

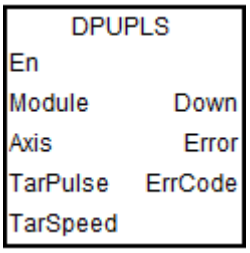
API	指令码			操作数	功能
1404	D	PUPLS		**Module ~ ErrCode**	PU 模块脉冲输出（无加减速）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Axis								●					○	○		
TarPulse								●					○	○		
TarSpeed								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
TarPulse			●				●						
TarSpeed			●				●						
Done	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

Axis ： 输出轴编号

TarPulse ： 目标输出个数

TarSpeed ： 目标输出频率

Done ： 输出完成标志

Error ： 输出错误标志

ErrCode ： 输出错误代码

指令说明：

1. 主机韧体版本 V1.08.00（含）之后支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。此指令有效时机点为 **En**=ON 时，当 **En** 变为 OFF 时，若还未完成输出则立即停止输出。

2. **Module** 为连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。此 PU 专属指令仅限于主机右侧之 PU 模块使用，不适用连接在远程模块右侧之 PU 模块。若指定之模块不为 PU 模块时，则 **Error** 错误标志会被设为 On。
3. **Axis** 为指定输出之 PU 模块的轴数编号，输入数值为 1~4 分别表示指定 PU 模块的轴 1~轴 4 输出，若该 PU 模块无此轴数编号时，则 **Error** 错误标志会被设为 On。
4. **TarPulse** 为指定输出之脉冲个数，其可输入之脉冲个数为有号数 32 位正的数值；当数值为 0 时，表示一直输出，不限制已输出的个数，直到此指令被关闭输出时才停止；当数值小于 0 时，PLC 将自动以 2 的补码方式，转为正整数之个数输出。
5. **TarSpeed** 为指定输出之目标速度(单位为 Hz),可输入数值为有号数 32 位数值 ,其可输入范围为-100,000（-100K） ~ 100,000（100K）之间；用户可在指令启动输出之后，随时再次修改目标频率，PU 模块会自行在输出一个完整脉冲之后，切换至最新设定之目标频率。注意：变换目标频率前，请考虑变化速度与 PLC 扫描时间是否合适。其模块对应的 TarSpeed 设定范围如下表所示：

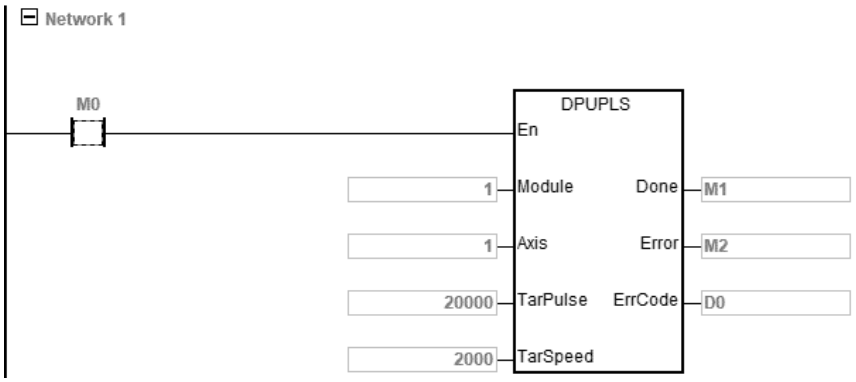
PU 模块名称	TarSpeed 设定范围
AS02PU	-200,000（-200K）~ 200,000（200K）
AS04PU	-100,000（-100K）~ 100,000（100K）

6. 当 **TarSpeed** 目标速度为正数时 (>0), 即表示”正方向”输出点为 Off 状态；当 **TarSpeed** 目标频率为负数时 (<0), 即表示”反方向”输出点为 On 状态；当 **TarSpeed** 目标速度为 0 时，即表示正在执行中的一个完整脉冲输出后，进入暂停输出状态。
7. 此输出指令不提供加速与减速功能，若有需求加减速功能，请改用 DPUDRI 指令。
8. 此输出指令可被用于变速度功能，当指令正在执行输出中，用户可变更 **TarSpeed** 目标频率数值，以达到变换输出速度之目的。
9. 当输出已达指定之 **TarPulse** 脉冲个数时，此 **Done** 完成标志将会被设为 On。此 **Done** 标志清除动作需由用户自行清除，且此指令只会在完成输出时，设定一次此标志。
10. 当启动输出过程中，有发生任何错误状况时，此 **Error** 错误标志将被设为 On，用户请参考 **ErrCode** 的错误代码，进行故障排除。
11. **ErrCode** 错误代码表如下：

错误代码	说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	PU 模块无此输出轴编号
16#1404	PU 模块之输出频率不合法
16#1405	PU 模块指定之输出轴正在输出当中，不允许重复指定输出
16#1406	PU 模块碰到正向极限停止输出
16#1407	PU 模块碰到反向极限停止输出

程序范例：

1. 设定 AS02PU HWCONFIG 输出点为「脉冲 Y0.0+方向 Y0.1」，轴 1 参数皆为默认值。
2. 当 M0=ON 时，执行轴 1 DPUPLS 脉冲输出，以 2kHz 频率从 Y0.0 输出脉冲数目 20,000 个 (无加减速)，Y0.1=OFF 表示为正方向。
3. 脉冲输出完毕后 M1=ON。



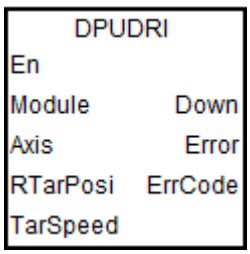
API	指令码			操作数	功能
1405	D	PUDRI		**Module ~ ErrCode**	PU 模块相对定位输出（附加减速）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Axis								●					○	○		
RTarPosi								●					○	○		
TarSpeed								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
RTarPosi			●				●						
TarSpeed			●				●						
Done	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

Axis ： 输出轴编号

RTarPosi ： 相对定位输出个数

TarSpeed ： 目标输出速度

Done ： 输出完成标志

Error ： 输出错误标志

ErrCode ： 输出状态代码

指令说明：

1. **主机韧体版本 V1.08.00（含）之后支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。此指令有效时机点为 En=ON 时，当 En 变为 OFF 时，若还未完成输出则减速停止输出。**

2. **Module** 为连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。此 PU 专属指令仅限于主机右侧之 PU 模块使用，不适用连接在远程模块右侧之 PU 模块。若指定之模块不为 PU 模块时，则 **Error** 错误标志会被设为 On。

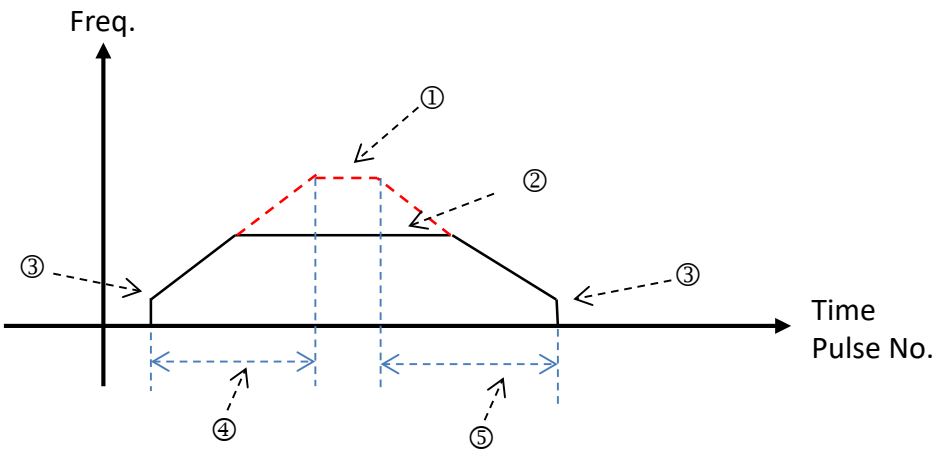
3. **Axis** 为指定输出之 PU 模块的轴数编号，输入数值为 1~4 分别表示指定 PU 模块的轴 1~轴 4 输出，若该 PU 模块无此轴数编号时，则 **Error** 错误标志会被设为 On。
4. **RTarPosi** 为指定输出相对定位之位置，其可输入之脉冲个数为有号数 32 位数值；当数值大于 0 时，表示往正方向输出（方向输出点 Off）；当数值小于 0 时，表示往反方向输出（方向输出点 On）；当数值为 0，此指令立即设定 **Done** 输出完成标志为 On。
5. **TarSpeed** 为指定输出目标速度（单位为 Hz），其可输入之频率值为有号数 32 位正整数的数值；当数值小于 0 时，指令自行以 2 的补码进行转换为正整数；当数值为 0，则指令将通知模块进入暂停模式，实际输出依照减速斜率进行减速，直到输出速度为 0，并设定暂停中标志（请参考 PUSTAT 指令说明）。其模块对应的 TarSpeed 设定范围如下表所示：

PU 模块名称	TarSpeed 设定范围
AS02PU	-200,000（-200K）~ 200,000（200K）
AS04PU	-100,000（-100K）~ 100,000（100K）

6. 在启动输出之后，其目标频率允许随时变更，但实际变换频率时，PLC 将自动依照 DPUCONF 指令所设定之加减速斜率变更频率。
7. 当输出已达指定之 **RTarPosi** 相对定位位置时，此 **Done** 完成标志将会被设为 On。此 **Done** 标志清除动作需由用户自行清除，此指令只会在完成输出时，设定一次此标志。
8. 当启动输出过程中，有发生任何错误状况时，此 **Error** 错误标志将被设为 On，用户请参考 **ErrCode** 的错误代码，进行故障排除。
9. **ErrCode** 错误代码表如下：

错误代码	说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	PU 模块无此输出轴编号
16#1404	PU 模块之输出频率不合法
16#1405	PU 模块指定之输出轴正在输出当中，不允许重复指定输出
16#1406	PU 模块碰到正向极限停止输出
16#1407	PU 模块碰到反向极限停止输出

10. PU 模块定位输出指令之加减速曲线说明如下图：



① ：最大输出频率设定值，此参数请参考 DPUCONF 指令设定，或由 HWCONFIG 设定。

② ：PU 模块输出指令指定之目标频率，此目标频率不可输出超过最大输出频率，若超过，则会被限制在最大输出频率。

③ ：启动/结束输出频率设定值，此参数请参考 DPUCONF 指令设定，或由 HWCONFIG 设定。

④ ：加速时间设定值，此参数请参考 DPUCONF 指令设定，或由 HWCONFIG 设定。

⑤ ：减速时间设定值，此参数请参考 DPUCONF 指令设定，或由 HWCONFIG 设定。

PU 模块之加减速动作为固定式斜率方式，因此实际加减速时间，将跟着指定输出的目标频率而变化。其加减斜率换算公式分别为（最大输出频率 – 启动频率）/ 加速时间，以及（最大输出频率 – 结束频率）/ 减速时间。

程序范例：

1. 于 HWCONFIG 设定 AS02PU 模块参数，设输入点 X0.0 为「轴 1 DOG 上升沿触发」及 X0.1 为「轴 1 Z 相 上升沿触发」，输出点为「A 相 Y0.0+B 相 Y0.1」，其余的轴 1 参数皆为默认值。
2. 当 M10=ON 时，执行轴 1 DPUCONF 指令修改参数，将 **Mode** 输出模式改为 1（脉冲 Y0.0+方向 Y0.1），**SSpeed** 启动速度为 200Hz，**Atime** 加速时间为 200ms，**Dtime** 减速时间为 200ms，**MSpeed** 最高速度为 100kHz，**Z_no** 回原点后寻找 Z 相次数为 1，**offset** 回到原点后输出偏移位置为-100，**Done** 输出完成后设 M11=ON。
3. 当 M0=ON 时，执行轴 1 PUSTAT 指令模块状态读回，若要清除当前位置，可设 M1=ON 清除（上升沿触发）轴 1 D0 当前位置。
4. 当 M20=ON 时，执行轴 1 DPUZRN 指令回原点，DPUSTAT 指令 D0 显示当前位置，以 1kHz 频率从 Y0.0 输出脉冲开始往正方向寻找原点，当碰到原点信号（DOG）X0.0=ON 时，减速变成以寸动速度 100Hz 继续往负方向移动，当 X0.0=OFF 时，再往正向移动寻找 Z 相，直到 X0.1（Z 相）第 1 次上升沿触发信号出现，再往负方向移动 100 个脉冲后结束，**Done** 输出完成后 M21=ON。
5. 当 M30=ON 时 执行轴 1 DPUDRI 相对定位输出，DPUSTAT 指令 D0 显示当前位置以 2kHz 频率从 Y0.0 输出脉冲数目 20,000 个（相对指定），Y0.1=OFF 表示为正方向，DPUSTAT 指令 D0 显示当前位置，**Done** 输出完成后 M31=ON。

输入点设置

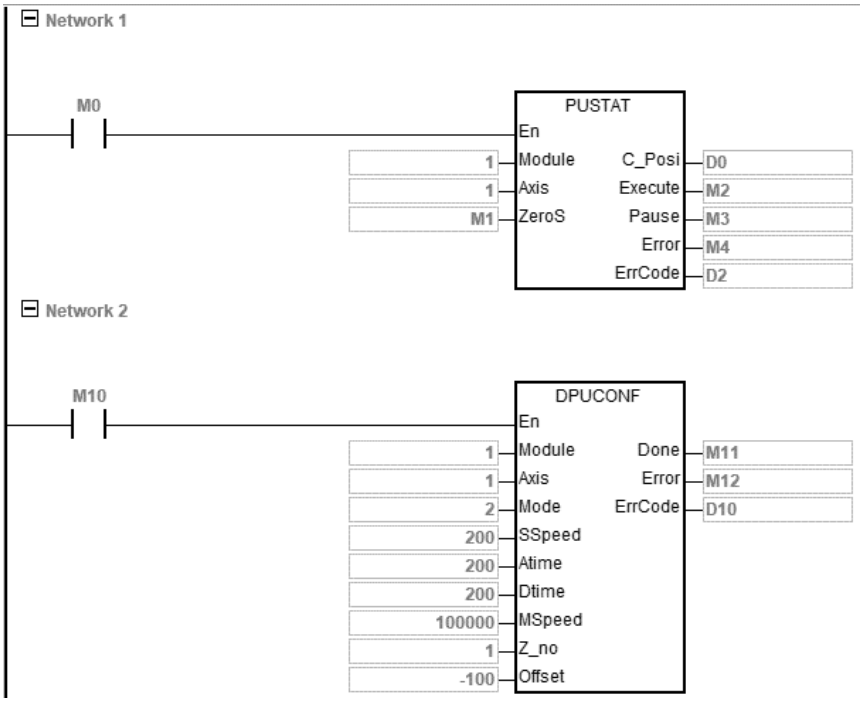
	参数名称	设定值	单位	默认值	最小值	最大值
▸	X0.0 模式	轴 1 DOG 正缘触发		一般输入点	-	-
	X0.1 模式	轴 1 Z 相 正缘触发		一般输入点	-	-
	X0.2 模式	一般输入点		一般输入点	-	-
	X0.3 模式	一般输入点		一般输入点	-	-
	X0.4 模式	一般输入点		一般输入点	-	-
	X0.0 滤波时间	10	ms	10	0	25
	X0.1 滤波时间	10	ms	10	0	25
	X0.2 滤波时间	10	ms	10	0	25
	X0.3 滤波时间	10	ms	10	0	25
	X0.4 滤波时间	10	ms	10	0	25

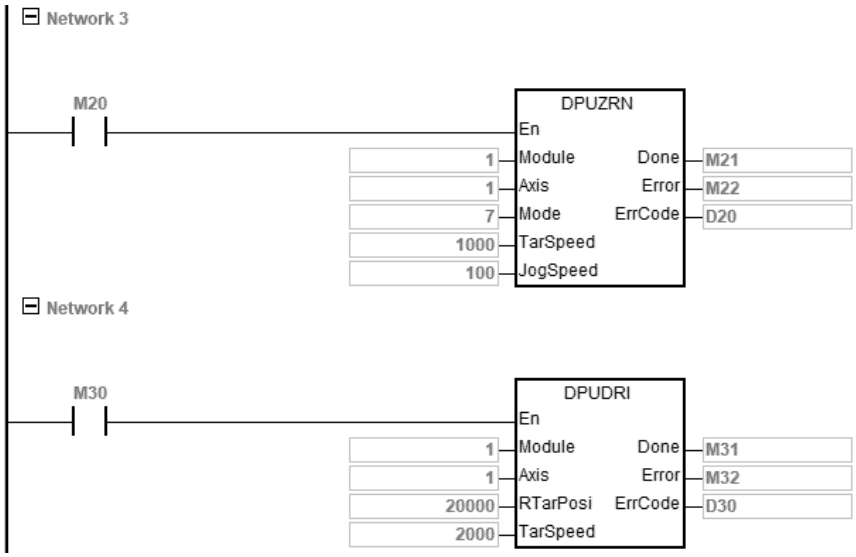
输出点设置

	参数名称	设定值	单位	默认值	最小值	最大值
▸	Y0.0/Y0.1 输出模式	A 相(Y0.0)+B 相(Y0.1)		脉波(Y0.0)+方向(Y...	-	-
	Y0.2/Y0.3 输出模式	脉波(Y0.2)+方向(Y0.3)		脉波(Y0.2)+方向(Y...	-	-

轴 1

	参数名称	设定值	单位	默认值	最小值	最大值
▸	启动速度	100	Hz	100	0	10000
	加速时间	100	ms	100	0	10000
	减速时间	100	ms	100	0	10000
	最高速度	200000	Hz	200000	100	200000
	回到原点后寻找 Z 相次数	0		0	-100	100
	回到原点后输出偏移位置	0	Pulse	0	-10000	10000
	原点的模式选择	模式 1:负向离开原点时停止		模式 1:负向离开原...	-	-
	负软体极限点	0	Pulse	0	-2147483648	2147483647
	正软体极限点	0	Pulse	0	-2147483648	2147483647




Network 3
M20
DPUZRN
En
1
Module
Done
M21
1
Axis
Error
M22
7
Mode
ErrCode
D20
1000
TarSpeed
100
JogSpeed
Network 4
M30
DPUDRI
En
1
Module
Done
M31
1
Axis
Error
M32
20000
RTarPosi
ErrCode
D30
2000
TarSpeed

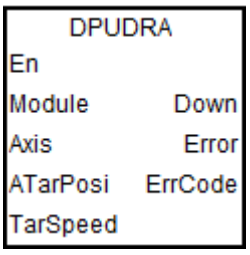
API	指令码			操作数	功能
1406	D	PUDRA		**Module ~ ErrCode**	PU 模块绝对寻址输出（附加减速）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Axis								●					○	○		
ATarPosi								●					○	○		
TarSpeed								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
ATarPosi			●				●						
TarSpeed			●				●						
Done	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

Axis ： 输出轴编号

ATarPosi ： 绝对寻址输出个数

TarSpeed ： 目标输出频率

Done ： 输出完成/暂停标志

Error ： 输出错误标志

ErrCode ： 输出状态代码

指令说明：

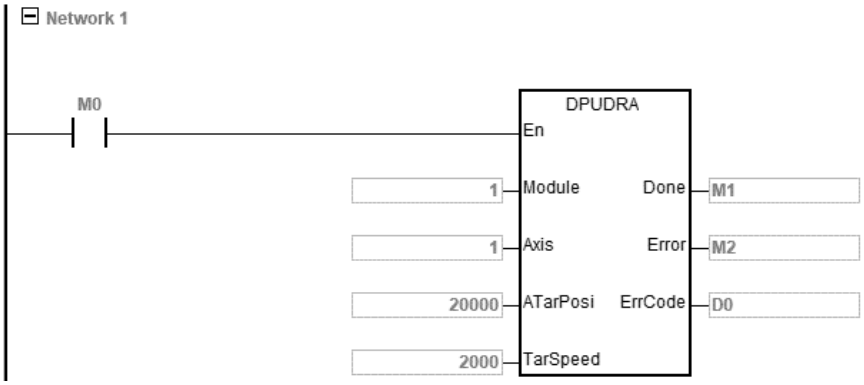
1. 主机韧体版本 V1.08.00（含）之后支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。此指令有效时机点为 **En**=ON 时，当 **En** 变为 OFF 时，若还未完成输出则减速停止输出。

2. **ATarPosi** 为指定输出绝对寻址之位置，其可输入之脉冲个数为有号数 32 位数值，PU 模块将自动比对记录之当前位置；若比对后的数值大于 0 时，表示往正方向输出（方向输出点 Off）；当比对数值小于 0 时，表示往反方向输出（方向输出点 On）；当数值为 0 时，此指令立即设定 **Done** 输出完成标志为 On。

3. 其它参数说明请参考 DPUDRI 指令说明。

程序范例：

1. 设定 AS02PU HWCONFIG 输出点为「脉冲 Y0.0+方向 Y0.1」，轴 1 参数皆为默认值。
2. 当 M0=ON 时，执行轴 1 DPUDRA 脉冲输出，以 2kHz 速度从 Y0.0 输出脉冲，直到当前位置为 20,000（绝对指定），Y0.1=OFF 表示为正方向。
3. 脉冲输出完毕后 M1=ON。



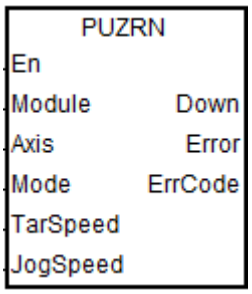
API	指令码			操作数	功能
1407	D	PUZRN		**Module ~ ErrCode**	PU 模块原点回归

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Axis								●					○	○		
Mode								●					○	○		
TarSpeed								●					○	○		
JogSpeed								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
Mode		●				●							
TarSpeed			●				●						
JogSpeed		●				●							
Done	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

Axis ： 输出轴编号

Mode ： 回原点的模式选择

TarSpeed ： 回原点最高输出频率

JogSpeed ： 原点的寸动输出频率

Done ： 输出完成标志

Error ： 输出错误标志

ErrCode ： 错误状态代码

指令说明：

1. 主机韧体版本 V1.08.00（含）之后支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。此指令有效时机点为 **En**=ON 时，当 **En** 变为 OFF 时，若还未完成原点回归则立即停止输出。
2. **Module** 为连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。此 PU 专属指令仅限于主机右侧之 PU 模块使用，不适用连接在远程模块右侧之 PU 模块。若指定之模块不为 PU 模块时，则 **Error** 错误标志会被设为 On。
3. **Axis** 为指定输出之 PU 模块的轴数编号，输入数值为 1~4 分别表示指定 PU 模块的轴 1~轴 4 输出，若该 PU 模块无此轴数编号时，则 **Error** 错误标志会被设为 On。
4. **Mode** 为指定回原点的模式选择，其模式说明如下表所示：

模式参数	功能说明	需选择搭配之输入点（由 HWCONFIG 设定）	备注
0	直接清除当前位置为 0	无	
1	反向离开原点时停止	DOG	HWCONFIG 设定
2	正向离开原点时停止	DOG	HWCONFIG 设定
3	模式 1 完成后，寻找 Z 相次数	DOG 与 Z 相输入	搭配 DPUCONF 设定 Z 相次数。
4	模式 2 完成后，寻找 Z 相次数	DOG 与 Z 相输入	
5	模式 1 完成后，输出偏移位置	DOG	搭配 DPUCONF 设定输出偏移位置
6	模式 2 完成后，输出偏移位置	DOG	
7	模式 1 完成后，先寻找 Z 相次数，在输出偏移位置	DOG 与 Z 相输入	搭配 DPUCONF 设定 Z 相次数和输出偏移位置。
8	模式 2 完成后，先寻找 Z 相次数，在输出偏移位置	DOG 与 Z 相输入	
255	修改轴当前输出位置	无	搭配 **TarSpeed** 参数使用
其它	保留		

注：若模式选择所需之输入点未搭配 HWCONFIG 设定，则有可能无法达成指定之回归原点动作。

注：上述模式 1~4，韧体版本 V1.02.00 版，其动作完成时，该轴当前输出位置不会清除为 0，需自行搭配 PUSTAT 指令清除。韧体版本 V1.02.10 版（含）之后，动作完成时，将会自动清除当前输出位置为 0。

注：上述模式 5~8，韧体版本 V1.02.00 版，其动作完成时，当前输出位置为该次输出结果，须自行搭配模式 255 修改当前位置至指定位置。韧体版本 V1.02.10 版（含）之后，动作完成时，该轴当前输出位置将清除为 0。

5. **TarSpeed** 当模式选择为 1~8 之间，为指定回原点时的最高输出速度，此数值为有号数 32 位数值，正负代表寻找原点的默认启动方向。例如，正代表指令启动从正方向开始寻找原点；其模块对应的 TarSpeed 设定范围如下表所示：

PU 模块名称	当模式选择为 1~8 之间，TarSpeed 设定范围
AS02PU	-200,000 ~ -100（Hz）和 100 ~ 200,000（Hz）
AS04PU	-100,000 ~ -100（Hz）和 100 ~ 100,000（Hz）

若是模式参数指定为 255 时，此 **TarSpeed** 将变为更新 PU 模块当前位置的数值。

6. **JogSpeed** 为指定碰到原点时的寸动输出速度，此数值为有号数 16 位数值，设定范围为 1~10,000（Hz）。
7. 当执行流程已达指定模式之动作流程时，此 **Done** 完成标志将会被设为 On。此 **Done** 标志清除动作需由用户自行清除，此指令只会在完成输出时，设定一次此标志。
8. 当启动输出过程中，有发生任何错误状况时，此 **Error** 错误标志将被设为 On，用户请参考 **ErrCode** 的错误代码，进行故障排除。
9. **ErrCode** 错误代码表如下：

错误代码	说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	PU 模块无此输出轴编号
16#1404	PU 模块之输出频率不合法
16#1405	PU 模块指定之输出轴正在输出当中，不允许重复指定输出
16#1406	PU 模块碰到正向极限停止输出
16#1407	PU 模块碰到反向极限停止输出

程序范例：请参考 API 1405 DPUDRI 指令范例说明。

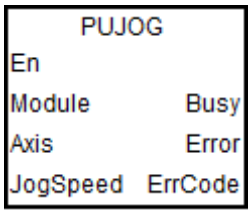
API	指令码			操作数	功能
1408	D	PUJOG		**Module ~ ErrCode**	PU 模块寸动输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Axis								●					○	○		
JogSpeed								●					○	○		
Busy		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
JogSpeed			●				●						
Busy	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ：模块编号

Axis ：输出轴编号

JogSpeed ：寸动输出频率

Busy ：输出执行中

Error ：输出错误标志

ErrCode ：输出错误代码

指令说明：

1. 主机韧体版本 V1.08.00（含）之后支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。此指令有效时机点为 **En**=ON 时，当 **En** 变为 OFF 时，则减速停止输出。
2. **Module** 为连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。此 PU 专属指令仅限于主机右侧之 PU 模块使用，不适用连接在远程模块右侧之 PU 模块。若指定之模块不为 PU 模块时，则 **Error** 错误标志会被设为 On。
3. **Axis** 为指定输出之 PU 模块的轴数编号，输入数值为 1~4 分别表示指定 PU 模块的轴 1~轴 4 输出，若该 PU 模块无此轴数编号时，则 **Error** 错误标志会被设为 On。

4. **JogSpeed** 为指定寸动输出速度，此数值为有号数 32 位数值。当数值大于 0 时，表示往正方向输出（方向输出点 Off）；当数值小于 0 时，表示往反方向输出（方向输出点 On）。当数值为 0 时，表示停止输出。此 JogSpeed 速度可于正在输出中进行速度变更，其可设定范围如下表所示：

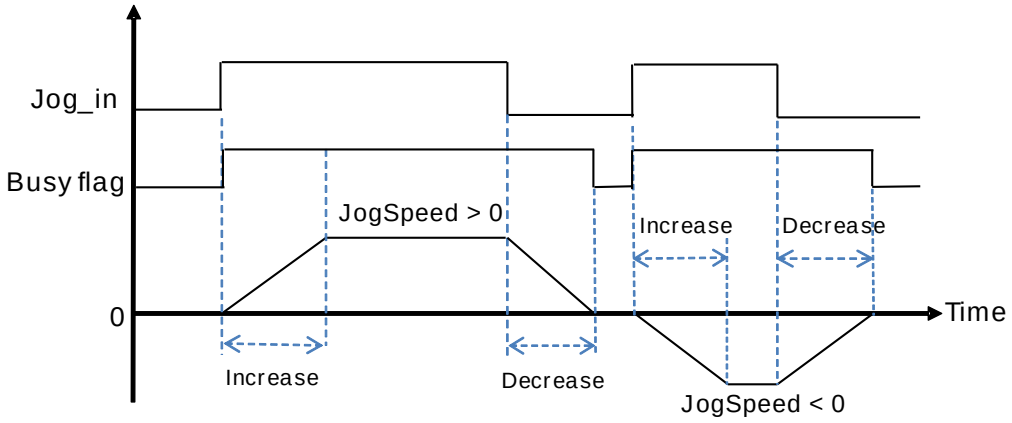
PU 模块名称	JogSpeed 设定范围
AS02PU	-200,000（-200K）~200,000（200K）
AS04PU	-100,000（-100K）~100,000（100K）

5. 当启动输出过程中，有发生任何错误状况时，此 **Error** 错误标志将被设为 On，用户请参考 **ErrCode** 的错误代码，进行故障排除。

6. **ErrCode** 错误代码表如下：

错误代码	说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	PU 模块无此输出轴编号
16#1404	PU 模块之输出频率不合法
16#1405	PU 模块指定之输出轴正在输出当中，不允许重复指定输出
16#1406	PU 模块碰到正向极限停止输出
16#1407	PU 模块碰到反向极限停止输出

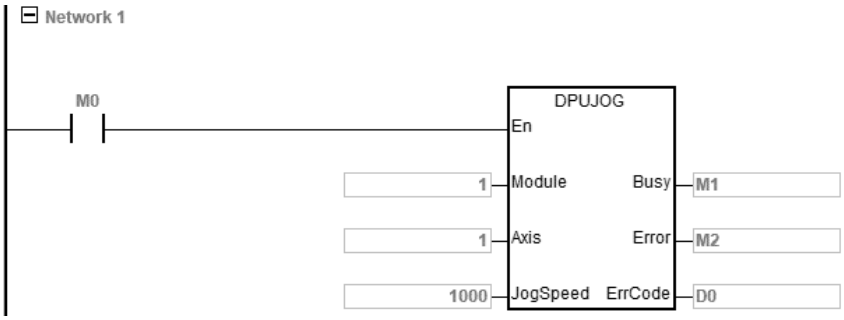
7. JOG 指令输出时序图如下：(其中 Jog_in 即是指令启动开关，Busy flag 即是 **Busy** 执行中标志)



8. JOG 指令关闭之后，需等待 **Busy** 标志已经 OFF 之后，才可再进行其它输出控制。

程序范例：

1. 设定 AS02PU HWCONFIG 输出点为「脉冲 Y0.0+方向 Y0.1」，轴 1 参数皆为默认值。
2. 当 M0=ON 时，执行轴 1 DPUJOG 寸动输出，以 1kHz 频率从 Y0.0 持续输出脉冲，Y0.1=OFF 表示为正方向。
3. 当指令正在运行中时 M1=ON。


Network 1
M0
DPUJOG
En
1
Module
Busy
M1
1
Axis
Error
M2
1000
JogSpeed
ErrCode
D0

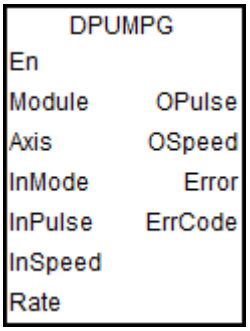
API	指令码			操作数	功能
1409	D	PUMPG		**Module ~ ErrCode**	PU 模块手摇轮输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Axis								●					○	○		
InMode								●					○	○		
InPulse								●								
InSpeed								●								
Rate								●								○
OPulse								●								
OSpeed								●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
InMode		●				●							
InPulse			●				●						
InSpeed			●				●						
Rate									●				
OPulse			●				●						
OSpeed			●				●						
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ：模块编号

Axis ：输出轴编号

InMode ：编码器输入模式与计数倍频设定

InPulse ：已输入个数显示

InSpeed ：已检测输入频率

Rate ：输入/输出比率值（浮点数）

OPulse ：已输出个数

OSpeed ：正在输出频率

Error ：输出错误标志

ErrCode ：错误代码

指令说明：

1. 主机韧体版本 V1.08.20（含）之后支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。
2. 此指令仅 AS02PU 模块，且该模块韧体版本须为 V1.04.00 版（含）以上才支持。此指令有效时机点为 **En**=ON 时，当 **En** 变为 OFF 时，则立即停止输出。
3. **Module** 为连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。此 PU 专属指令仅限于主机右侧之 PU 模块使用，不适用连接在远程模块右侧之 PU 模块。若指定之模块不为 PU 模块时，则 **Error** 错误标志会被设为 On。
4. **Axis** 为指定输出之 PU 模块的轴数编号，输入数值为 1~4 分别表示指定 PU 模块的轴 1~轴 4 输出，若该 PU 模块无此轴数编号时，则 **Error** 错误标志会被设为 On。
5. **InMode** 为编码器来源输入模式与计数倍频设定，其设定参数说明如下表：

InMode 设定说明	
数值	输入模式（设定下述保留数值，模块将以默认值执行）
16#0000	4 倍频 A/B 相输入（默认） A 相领先 B 相表示正向计数，B 相领先 A 相表示反向计数
16#0001	1 倍频 A/B 相输入 计数方向请参考参数值 0
16#0002	2 倍频 A/B 相输入 计数方向请参考参数值 0
16#0003	保留
16#0004	4 倍频 A/B 相输入（请参考参数值 0）
16#0005	脉冲+方向输入（A+/A- 为脉冲输入，B+/B- 为方向输入） B 相为 OFF 时表示正向计数，为 ON 时表示反向计数 A 相采上升沿触发计数

InMode 设定说明	
数值	输入模式（设定下述保留数值，模块将以默认值执行）
16#0006	脉冲+方向输入（A+/A- 为脉冲输入，B+/B- 为方向输入） B 相为 ON 时表示正向计数，为 OFF 时表示反向计数 A 相采上升沿触发计数
16#0007	单相脉冲输入（A+/A- 为脉冲输入） A 相采上升沿触发计数
其它数值	保留

6. **InPulse** 为显示已输入之脉冲计数个数（有号数 32 位数值），此数值在每次启动指令时，PU 模块将自动清除为 0，然后才开始计数。
7. **InSpeed** 为显示已检测之输入频率（有号数 32 位数值），此频率检测的基本时间为 20ms，因此若 20ms 内无计数值，则检测输入频率为 0，若有计数值，则速度将从最低 50Hz 开始输出，即使 **Rate** 比率换算 **OSpeed** 输出速度低于 50Hz，还是以 50Hz 输出。
8. **Rate** 为输入与输出的比率值，其数值为浮点数，实际输出的个数与频率，分别等于输入个数与频率乘以此比率值，举例：输入速度为 100Hz，比率值为 0.5，则输出速度为 100x0.5=50Hz。若换算后的输出最高频率超过 100KHz，则将会限制于 100KHz。备注：请注意最高速输出，若长时间处于最高速输出，将有可能造成手摇轮已经停止，但输出却还有累积的个数，需要继续执行完毕的状况发生。
9. **OPulse** 与 **OSpeed** 分别为显示已输出个数与正在输出中频率（有号数 32 位数值）。
10. 关闭此指令时，请先判断正在输出中的频率值，检查是否已为 0 数值，若还未达 0 即关闭指令，则 PU 模块将立即关闭输出，不再输出比率转换后的指定输出个数。
11. **ErrCode** 错误代码表如下：

错误代码	说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	PU 模块无此输出轴编号
16#1404	PU 模块之输出频率不合法
16#1405	PU 模块指定之输出轴正在输出当中，不允许重复指定输出
16#1406	PU 模块碰到正向极限停止输出
16#1407	PU 模块碰到反向极限停止输出

12. 当指令被启动或关闭时，表示主机得要去通知模块的高速计数器功能开启或关闭，故此指令不可与 API1410 DPUCNT 一起同时被使用，否则可能会发生指令相互开启或关闭模块计数的状况发生。

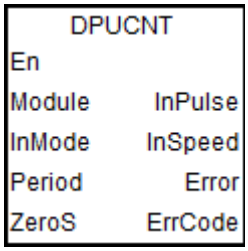
API	指令码			操作数	功能
1410	D	PUCNT		**Module ~ ErrCode**	PU 模块高速计数器功能

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
InMode								●					○	○		
Period								●					○	○		
ZeroS	●	●	●	●				●								
InPulse								●								
InSpeed								●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
InMode		●				●							
Period		●				●							
ZeroS	●												
InPulse			●				●						
InSpeed			●				●						
Error	●												
ErrCode			●				●						

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

InMode ： 编码器输入模式与计数倍频设定

Period ： 速度抓取周期时间

ZeroS ： 计数器清除为 0

InPulse ： 已输入个数显示

InSpeed ： 每周期时间之计数个数

Error ： 输出错误标志

ErrCode ： 错误代码

指令说明：

1. 主机韧体版本 V1.08.00（含）之后支持此指令。ISPSoft 编辑软件建议使用 V3.12 版本（含）以上。
2. 此指令仅 AS02PU 模块支持。此指令有效时机点为 **En**=ON 时，当 **En** 变为 OFF 时，则立即停止计数。

3. **Module** 为连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。此 PU 专属指令仅限于主机右侧之 PU 模块使用，不适用连接在远程模块右侧之 PU 模块。若指定之模块不为 PU 模块时，则 **Error** 错误标志会被设为 On。
4. **InMode** 为编码器来源输入模式与计数倍频设定，其设定参数说明如下表：

   Note：A 相领先 B 相表示正向计数，B 相领先 A 相表示反向计数

InMode 设定说明	
数值	输入模式（设定下述保留数值，模块将以默认值执行）
16#0000	保留
16#0001	1 倍频 A/B 相输入
16#0002	2 倍频 A/B 相输入
16#0003	保留
16#0004	4 倍频 A/B 相输入（默认值）
16#0005	脉冲+方向输入（A+/A- 为脉冲输入，B+/B- 为方向输入） B 相为 OFF 时表示正向计数，为 ON 时表示反向计数 A 相采上升沿触发计数
16#0006	脉冲+方向输入（A+/A- 为脉冲输入，B+/B- 为方向输入） B 相为 ON 时表示正向计数，为 OFF 时表示反向计数 A 相采上升沿触发计数
16#0007	单相脉冲输入（A+/A- 为脉冲输入） A 相采上升沿触发计数
其它数值	保留

5. **Period** 为速度抓取的周期时间设定值，可设定范围为 10ms ~ 1000ms 之间，超出范围数值将自动以最小或最大值设定。
6. **InPulse** 为显示已输入之脉冲计数个数（有号数 32 位数值），此计数的数值为停电保持型，若是有需要清除为 0 时，请在指令启动时，利用 **ZeroS** 设定标志（Off → On）进行清除。
7. **InSpeed** 为显示每一个 **Period** 时间所计数到的数值（有号数 32 位数值），用户若有需要转换以 Hz 为单位的数值，请自行使用计算公式转换。
8. **ErrCode** 错误代码表如下：

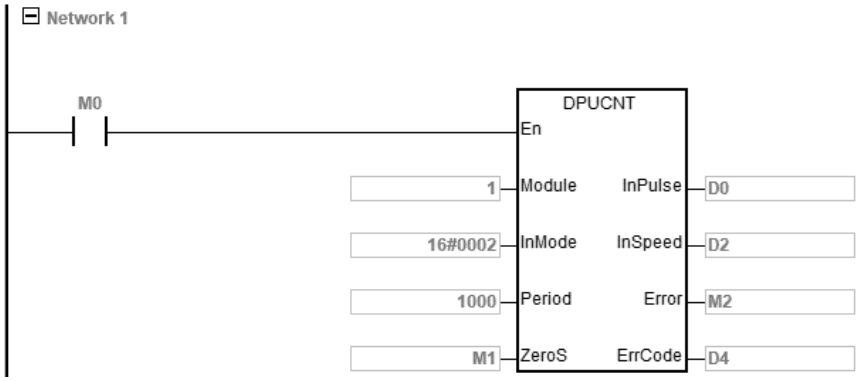
错误代码	说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1406	PU 模块无输入计数功能

9. 当指令被启动或关闭时，表示主机得要去通知模块的高速计数器功能开启或关闭，故此指令不可与

API1409 DPUMPG 一起同时被使用，否则可能会发生指令相互开启或关闭模块计数的状况发生。

程序范例：

1. 当 M0=ON 时，执行 PUCNT 指令计数，**InMode** 为 2 倍频 A/B 相输入
2. M1=ON 清除轴 1 之 **InPulse** 已输入脉冲计数个数
3. 当输入的速度为 10Hz 及脉冲个数为 100 时，则 **InPulse** 及 **InSpeed** 显示为 200 脉冲及 20Hz



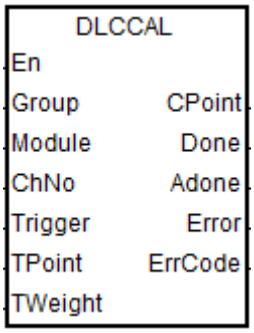
API	指令码			操作数	功能
1415	D	LCCAL		**Group, Module ~ ErrCode**	LC 模块通道校正

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Group								●					○	○		
Module								●					○	○		
ChNo								●					○	○		
Trigger	●	●	●	●				●								
TPoint								●					○	○		
TWeight								●								
CPoint								●								○
Done		●	●	●				●								
ADone		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
ChNo		●				●							
Trigger	●												
TPoint		●				●							
TWeight									●				
CPoint		●				●							
Done	●												
ADone	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Group ：群组编号

Module ：模块编号

ChNo ：通道编号

Trigger ：触发校正

TPoint ：校正总点数

TWeight ：校正重量值

CPoint ：已经校正完成点编号

Done ：单次校正完成标志

ADone ：全部校正完成

Error ：错误标志

ErrCode ：错误代码

指令说明：

1. AS 主机韧体 V1.08.00（含）之后，以及 ISPSoft 软件版本 V3.09 版（含）之后，支持此指令。
2. 此指令仅 AS02LC 模块支持，并且需先从 HWCONFIG 得知目前模块配置之位置。
3. **Group** 为指定 LC 模块连接于主机右侧或者远程模块右侧之群组编号，主机编号为 0，第 1 台远程模块编号为 1，以此原则类推，最大群组数为 15。若指定之模块不为 LC 模块时，则 **Error** 错误标志会被设为 On。
4. **Module** 为连接于主机或远程右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。若指定之模块不为 LC 模块时，则 **Error** 错误标志会被设为 On。
5. **ChNo** 为指定 LC 模块的通道编号，若输入数值不是 LC 模块的通道编号时，则 **Error** 错误标志会被设为 On。
6. **Trigger** 为触发单点校正的命令，当 **Trigger** 状态由 Off 变 On 时，将会通知 LC 模块进行单次校正，完成之后，**Done** 标志将会被设为 On；若是已经完成全部校正点的校正动作，则 **ADone** 标志也会一起 On。用户要再进行下一个校正点之前，需判断 **Done** 标志已经为 On 之后，再自行将 **Trigger** 变为 Off，此时指令也会监看 **Trigger** 由 On 变 Off 时，自动清除 **Done** 标志。
7. **TPoint** 为此次校正的总点数，用户在指令启动之后，就不可再变更此数值，因为此 **TPoint** 数值已经在首次启动时，传送给 LC 模块准备校正。注：校正的总点数限制，请参考 LC 模块说明，此指令将不会进行判断。
8. **TWeight** 为此次校正的各点校正重量值，依校正总点数（N），连续占用 N*2 个装置。用户在指令启动之后，就不可再变更此数值，因为此 **TWeight** 数值已经在首次启动时，传送给 LC 模块准备校正。第一点校正重量值须为 0，若不为 0，则 **Error** 错误标志会被设为 ON。举列：TPoint 校正总点数为 3 点，TWeight 起始装置为 D200，会连续占用 6 个装置 D200~D205，第一点校正重量值为 D200（Low Word）D201（High Word），第二点校正重量值为 D202、D203，第三点校正重量值为 D204、D205。

9. 当 **ADone** 完成标志一旦由 Off 变为 On 时，整个校正动作就会停止，除非 LCCAL 指令被关闭，然后再开启之后，才可重新进行校正动作。

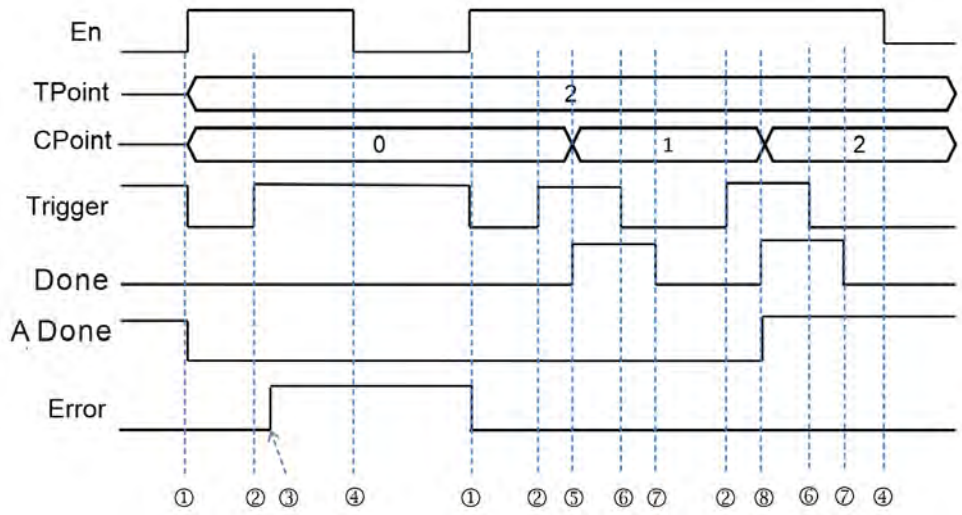
   CPoint 为已经校正的点数编号，用户不可变更此数值，当 **CPoint** >= **TPoint**，指令就会认定已经校正完成，而且自动设定 **ADone** 标志为 On。

10. 此指令首次启动时，**CPoint** 数值会自动被清除为 0，以及 **Trigger**、**Done** 与 **ADone** 标志清除为 Off，表示准备进行校正，当用户设定 **Trigger** 信号（Off→On），并且等 LC 模块完成校正之后，此 **CPoint** 数值将会自动加 1，并且设定单次校正完成 **Done** 信号为 On。用户可藉由此数值观察目前已经完成校正的点位，举例：当 **CPoint** 数值为 2 时，即表示 **Trigger** 触发 2 次，模块已经完成 2 点校正。

11. **ErrCode** 错误代码表如下：

错误代码	说明
16#1410	LC 群组或模块编号错误
16#1411	LC 模块无此通道编号
16#1412	LC 模块数据写入错误或是通讯逾时
16#1413	LC 模块已校正完成，请关闭指令后再重新触发
16#1414	第一点校正重量值不为 0
16#1415	校正重量值不是合法的浮点数
16#1416	超出测量范围
16#1417	超出重量上限
16#1418	校正错误

12. 校正范例的时序图如下所示。



时序图内各时机点说明如下：

① → 指令启动，自动清除 **CPoint** 数值、**Trigger**、**Done**、**ADone** 与 **Error** 标志。

② → 用户触发校正标志。

③ → 指令判断模块编号错误，设定错误代码为"指定 LC 模块编号错误"。

④ → 用户关闭指令。

⑤ → 用户触发之后，LC 模块完成单点校正，**CPoint** 加 1，以及 **Done** 设定为 On。

⑥ → 用户清除 **Trigger** 触发信号。

⑦ → 指令跟随清除 **Done** 信号。

⑧ → 用户触发之后，LC 模块全部完成校正，**CPoint** 加 1，以及 **Done 与 ADone 皆**设定为 On。

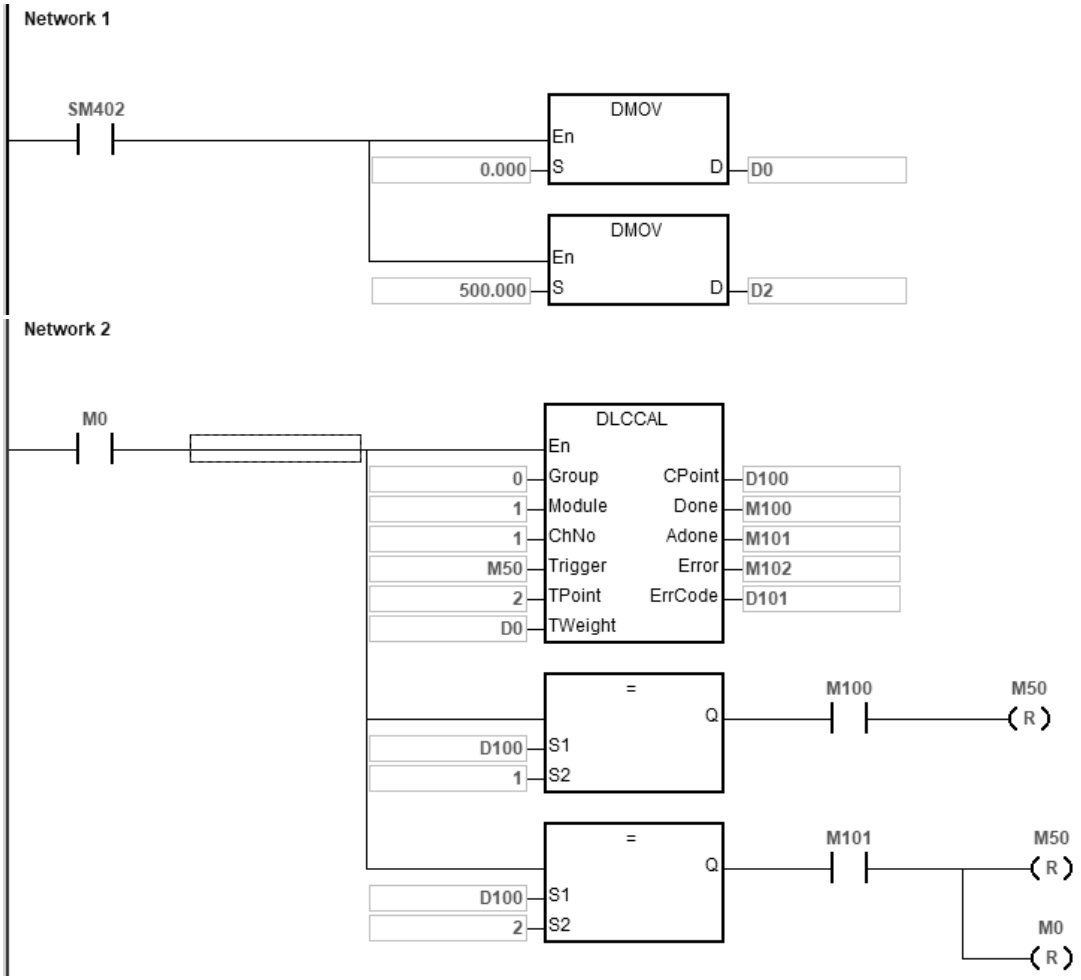
程序范例：

当 PLC RUN 时，会将第一点校正重量值为 0 写入 D0（空载校正），第二点校正重量值 500.0 写入 D2。

① → 当 M0 为 On 时，使用 DLCCAL 指令执行两点校正

② → 确认称台上为空载，SET M50 为 ON，进行第一点校正（空载校正）；当 D100=1 且 M100 为 ON 时即完成第一点校正，M50 为 OFF

③ → 称台上放上 500g 砝码后 SET M50 为 ON，进行第二点校正（500g 校正）；当 D100=2 且 M101 为 ON 时即完成第二点校正，M50 为 OFF 及 M0 为 OFF



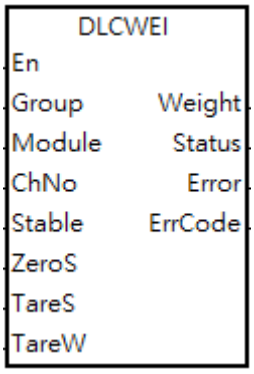
API	指令码			操作数	功能
1416	D	LCWEI		**Group, Module ~ ErrCode**	LC 模块重量值读取

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Group								●					○	○		
Module								●					○	○		
ChNo								●					○	○		
Stable								●					○	○		
ZeroS	●	●	●	●				●								
TareS	●	●	●	●				●								
TareW								●								
Weight								●								
Status								●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
ChNo		●				●							
Stable									●				
ZeroS	●												
TareS	●												
TareW									●				
Weight									●				
Status		●				●							
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Group ：群组编号

Module ：模块编号

ChNo ：通道编号

Stable ：设定重量稳定范围区间（0.0~100000.0）

ZeroS ：设定重量归零

TareS ：设定皮重

TareW ：皮重的重量值

Weight ：当前重量值

Status ：LC 模块状态代码

Error ：错误标志

ErrCode ：错误代码

指令说明：

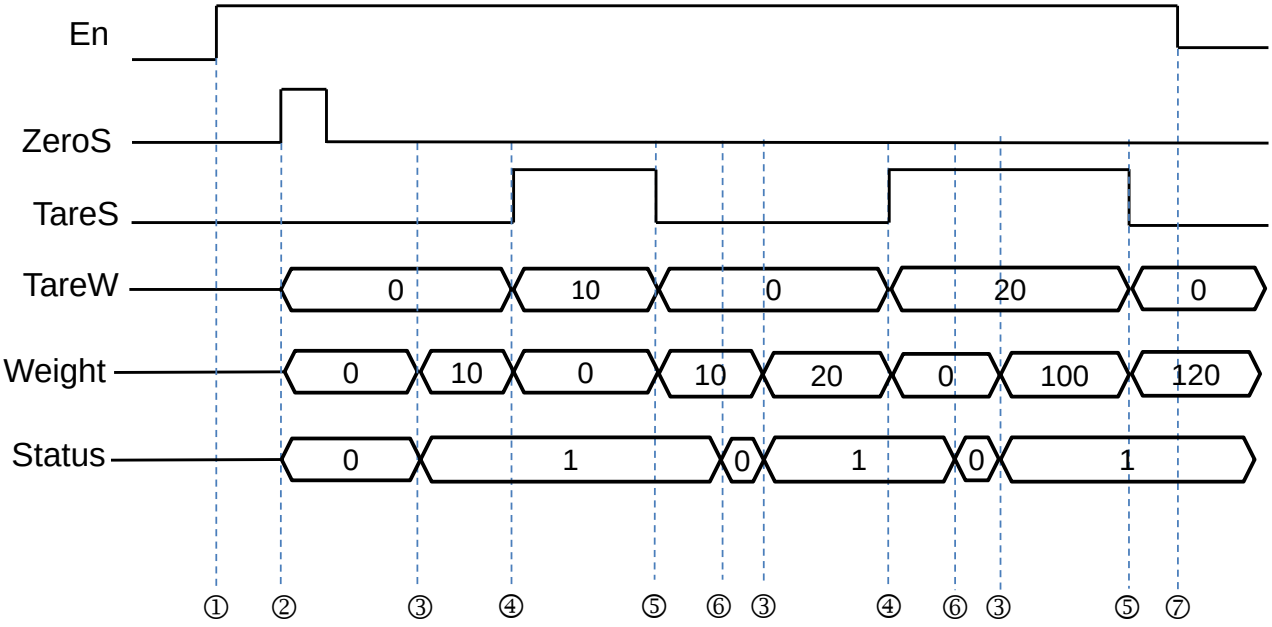
1. AS 主机韧体 V1.06.00（含）之后，以及 ISPSoft 软件版本 V3.08 版之后，支持此指令。
2. 此指令仅供 AS 系列的 LC 模块支持，并且需先从 HWCONFIG 得知目前模块配置之位置。
3. **Group** 为指定 LC 模块连接于主机右侧或者远程模块右侧之群组编号，主机编号为 0，第 1 台远程模块编号为 1，以此原则类推，最大群组数为 15。若指定之模块不为 LC 模块时，则 **Error** 错误标志会被设为 On。
4. **Module** 为连接于主机或远程右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。若指定之模块不为 LC 模块时，则 **Error** 错误标志会被设为 On。
5. **ChNo** 为指定 LC 模块的通道编号，若输入数值不是 LC 模块的通道编号时，则 **Error** 错误标志会被设为 On。
6. **Stable** 为重量稳定范围区间数值，此数值为原始转换值（RAW data），可输入范围为 0.0~100000.0 之间浮点数数值，若是设定超出此范围，则指令自动以最小或最大值设定。此参数设定 LC 模块的时机都是在指令初次启动时，后续若要修改 LC 模块内的此数值，则需先关闭指令后，再设定新的范围值，然后开启指令重新设定。
7. 此指令请启动之后，将会自动将指定的通道改成 ”净重” 显示模式，若客户有需要知道 ”毛重” （总重量）的数值，则可自行将 **TareW** 与 **Weight** 相加即可。
8. **ZeroS** 为设定当前重量为 0 的标志，当此标志由 Off 变 On 时，**TareW** 与 **Weight** 将被清除为 0。
9. **TareS** 为设定皮重的标志，当 **TareS** 标志由 Off 变 On 时，**Weight** 现有重量搬移到 **TareW**，并且将 **Weight** 数值清除为 0。当 **TareS** 标志由 On 变 Off 时，**TareW** 将会回归到 **Weight** 现有重量值，并且将 **TareW** 数值清除为 0。
10. **Weight** 为已经扣除皮重的重量值后，所称的重量值，用户可以监看 **TareW** 是否有数值，以判断是否有启动皮重功能，当数值为 0 时，表示皮重重量还未设定。

11. **Status** 为此指令整合 LC 模块的常用状态代码，其状态整理后如下表说明。

数值	0	1	2	3	4	5
代码说明	测量中 或空载	重量已稳定	硬件/校正错误	校正中	重量超出范围	模块编号/通道错误

12. 读取重量过程中，若 LC 模块发生错误状况时，除了 **Status** 会显示当时错误代码，其 **Error** 标志也会被设为 On，当状态恢复正常时，此 **Error** 会自动清除。若需知道详细错误状况，请参考 LC 模块篇的状态控制寄存器说明。

13. 重量读取范例的时序图如下图：



时序图内各时机点说明如下：

① → 指令启动。

② → 用户下达清除归 0，指令将清除 **TareW** 与 **Weight** 数值，以及 **Status** 状态。

③ → 用户放置被称物于称台上，当量称到稳定的重量值时，**Status** 会变为 1，并且显示 **Weight** 重量值。

④ → 用户下达皮重设定 On，此时 **Weight** 重量移到 **TareW** 内，之后清除 **Weight** 重量。

⑤ → 用户清除皮重设定 Off，此时 **TareW** 重量移回到 **Weight** 内，之后清除 **TareW** 重量。

⑥ → 用户再次放入另一个被称物品于称台上，此时 **Status** 转为测量中。

⑦ → 指令关闭，**TareW**、**Weight** 与 **Status** 保持最后的状态数值。

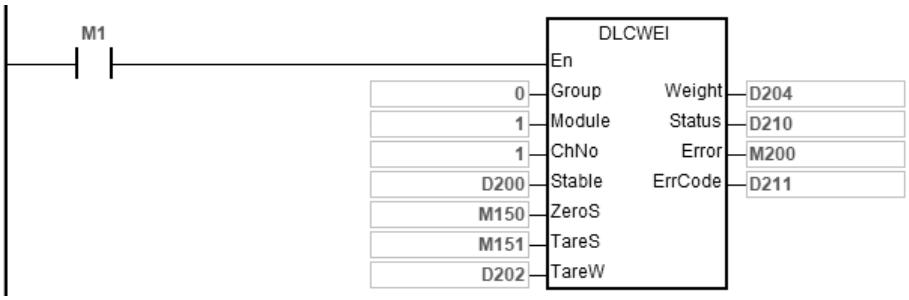
14. **ErrCode** 错误代码表如下：

错误代码	说明
16#1410	LC 群组或模块编号错误
16#1411	LC 模块无此通道编号
16#1414	LC 模块数据写入错误或是通讯逾时

程序范例：

在完成校正后，即可使用 DLCWEI 指令进行重量测量

1. 重量测量：称台上放上 500g 砝码，当 M1 为 On 时，D204 显示当前重量 500.0
2. 皮重设定：
   ① →称台上放上包装材（假设为 100g 重），此时 D204 显示当前重量 100.0
   ② →当 M151 为 ON，此时 **Weight（D204）**重量移到 **TareW（D202）**内，之后清除 **Weight** 重量
   ③ →D204=0.0，D202=100.0，此时皮重设定完成。
3. 清除皮重设定：
   ① →当 M151 为 OFF，此时 **TareW（D202）**重量移回到 **Weight（D204）**内，之后清除 **TareW** 重量
   ② → D204=100.0，D202=0.0 完成清除去皮。
4. 重量稳定范围区间设定（稳定检查功能）:
   ① →指令启动前，须先设定 D200=10.0
   ② →当放上 500g 砝码，前后测量范围在 490~510g 之间，D210=1（重量已稳定）



API	指令码			操作数	功能
1417	D	MPID		**GROUP, MODULE ~ ERROR**	RTD/TC 模块 PID 运算

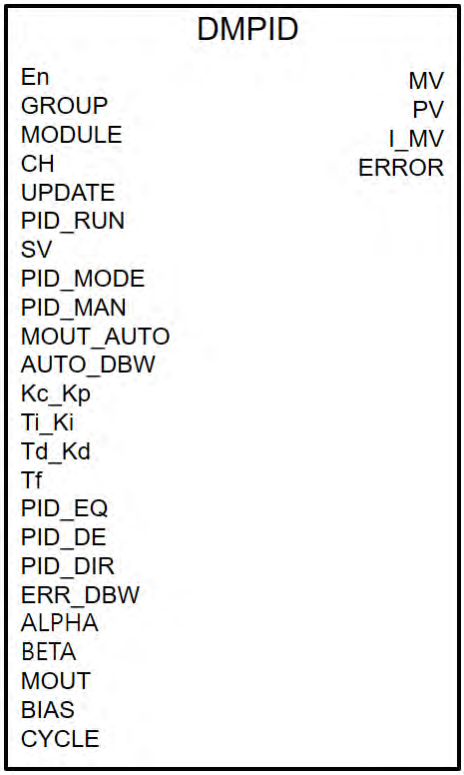
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
GROUP								●					○	○		
MODULE								●					○	○		
CH								●					○	○		
UPDATE	●	●	●	●				●								
PID_RUN	●	●	●	●				●								
SV								●					○	○		
PID_MODE								●					○	○		
PID_MAN	●	●	●	●				●								
MOUT_AUTO	●	●	●	●				●								
AUTO_DBW								●					○	○		
Kc_Kp								●								○
Ti_Ki								●								○
Td_Kd								●								○
Tf								●								○
PID_EQ	●	●	●	●				●								
PID_DE	●	●	●	●				●								
PID_DIR	●	●	●	●				●								
ERR_DBW								●								
ALPHA								●					○	○		
BETA								●					○	○		
MOUT								●								
BIAS								●					○	○		
CYCLE								●					○	○		
MV								●								
PV								●								
I_MV								●								
ERROR								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING

数据类型请参考下列操作数说明

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



GROUP	：	群组编号
MODULE	：	模块编号
CH	：	通道编号
UPDATE	：	更新 PID 参数标志
PID_RUN	：	启动 PID 运算
SV	：	目标值
PID_MODE	：	PID 控制模式
PID_MAN	：	PID Auto/Manual 模式
MOUT_AUTO	：	手动值（MOUT）自动更新模式
AUTO_DBW	：	Auto tuning 不作用区
Kc_Kp	：	比例项系数
Ti_Ki	：	积分项系数
Td_Kd	：	微分项系数
Tf	：	微分作用时间常数
PID_EQ	：	PID 计算式选择
PID_DE	：	PID 微分项误差计算选择
PID_DIR	：	PID 正反向
ERR_DBW	：	ERR 的不作用范围
ALPHA	：	初始积分量补偿参数（加热）
BETA	：	初始积分量补偿参数（冷却）
MOUT	：	MV 手动值
BIAS	：	前馈控制输出值
CYCLE	：	取样时间
MV	：	输出值
PV	：	当前值
I_MV	：	积分项累计值
ERROR	：	错误代码

指令说明：

1. 主机韧体版本 V1.06.00（含）之后，仅支持主机右侧温度模块。

   主机韧体版本 V1.08.00（含）之后，增加支持远程右侧温度模块。AS00SCM-A FW V2.06 以上支持。

2. 此指令仅供 AS 系列的温度模块支持（支持版本为 AS04RTD-A V1.04 以上/AS06RTD-A V1.00 以上；AS04TC-A V1.04 以上/AS08TC-A V1.00 以上）。

3. 参数设定内容如下：

操作数名称	数据类型	功能	设定范围	说明
GROUP	DWORD/DINT	群组编号	指定 RTD/TC 模块连接于主机右侧或者远程模块右侧之群组编号，主机编号为 0，第 1 台远程模块编号为 1，以此原则类推，最大群组数为 15。	
MODULE	DWORD/DINT	模块编号	连接于主机或远程右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。	
CH	DWORD/DINT	通道编号	执行 PID 运算之通道编号，通道 1 编号为 1，通道 2 编号 2，以此原则类推。	
UPDATE	BOOL	更新 PID 参数标志	指令启动 EN 后，若需中途变更参数，请先将参数设定为新数值，再设定 UPDATE 标志为 TRUE；当此指令完成变更之后，指令将会自行清除 UPDATE 标志为 FALSE。	
PID_RUN	BOOL	启动 PID 运算	TRUE：PID 指令开始运算。 FALSE：停止运算，且 MV 值清除为 0。	
SV	DWORD/DINT	SV 值	-32768~32767	目标值
PID_MODE	DWORD/DINT	PID 控制模式	0：自动控制，当 PID_MAN 由 TRUE 转为 FALSE 时，MV 值会由当时输出的 MV 值开始进行自动运算。 1：自动调整参数功能，调整完毕时将自动进入自动控制模式（PID_MODE 改为 0）并且填入最适用的 Kc_Kp、Ti_K_i、Td_K_d、Tf、ALPHA 及 BETA 等参数。 备注：使用自动调整参数功能时，此模式不可使用数值直接设定。	
PID_MAN	BOOL	PID A/M 模式	TRUE：Manual，MV 值会依 MOUT 值输出，当 PID_MODE 为 1 时此设定无效。 FALSE：Auto，MV 值会依 PID 公式计算后输出。	

操作数名称	数据类型	功能	设定范围	说明
MOUT_AUTO	BOOL	MOUT 自动更新模式	TRUE：Auto，MOUT 的值会随着 MV 值改变。 FALSE：Normal，MOUT 的值不会随着 MV 值改变。	
AUTO_DBW	DWORD/DINT	Auto tuning 不作用区	0~32000	应用于 Auto tuning 时 SV ± dead band 范围不作动
Kc_Kp	REAL	比例项系数（Kc or Kp，依 PID_EQ 参数决定使用何种系数）	正数单精度浮点数范围	P 计算值系数，如果小于 0 则 Kc_Kp 将为 0，在 Independent 下若 Kc_Kp 等于 0 则表示不使用 P 控制 。
Ti_Ki	REAL	积分项系数（Ti or Ki，依 PID_EQ 参数决定使用何种系数）	正数单精度浮点数范围（单位：Ti = sec；Ki = 1/sec）	I 计算值系数，如果小于 0 则 Ti_Ki 将为 0，当 Ti_Ki 等于 0 时则表示不使用 I 控制。
Td_Kd	REAL	微分项系数(Td or K_d，依 PID_EQ 参数决定使用何种系数）	正数单精度浮点数范围（单位 :sec）	D 计算值系数，如果小于 0 则 Td_Kd 将为 0，当 Td_Kd 等于 0 时则表示不使用 D 控制。
Tf	REAL	微分作用时间常数（Tf）	正数单精度浮点数范围（单位 :sec）	微分作用时间常数，如果小于 0 则 Tf 将为 0，当 Tf 等于 0 时则表示不使用微分作用时间的控制（Derivative Smoothing）。
PID_EQ	BOOL	PID 计算式选择	TRUE：Dependent Formula FALSE：Independent Formula	
PID_DE	BOOL	PID 微分项误差计算选择	TRUE：使用当前值（PV）的变化量来计算微分项的控制值（Derivative of PV） FALSE：使用偏差量（E）的变化量来计算微分项的控制值（Derivative of E）	

操作数名称	数据类型	功能	设定范围	说明
PID_DIR	BOOL	PID 正反向	FALSE：加热动作（E=SV-PV） TRUE ：冷却动作（E=PV-SV）	
ERR_DBW	DWORD/DINT	偏差量（E）不作用范围	-32768~32767	偏差量(E)等于 SV−PV 或 PV-SV，当 ERR_DBW 设定为 0 时即表示不启动此功能，否则主机会去检查这次的 E 值是否小于 ERR_DBW 的绝对值且是否符合 Cross 状态转换条件，若都有则将 E 值视为 0 之后进行 PID 计算，否则依照正常处理将 E 值代入 PID 计算。
ALPHA	DWORD/DINT	初始积分量补偿参数（加热）	0~100 （单位 :1%）	初始积分量补偿(加热)
BETA	DWORD/DINT	初始积分量补偿参数（冷却）	0~100 （单位 :1%）	初始积分量补偿(冷却)
MOUT	DWORD/DINT	MV 手动值	0~1000 （单位： 0.1%）	配合 PID_MAN 模式使用，当 PID 设定在手动模式(PID_MAN=True) 时，则输出值 (MV) 会依照手动设定值（MOUT）输出。
BIAS	DWORD/DINT	前馈控制输出值	-32768~32767	使用于 PID 前馈控制。
CYCLE	DWORD/DINT	取样时间（T_S）	1~1000 （单位： 100ms）	以 CYCLE 设定的取样时间来计算 PID，并更新输出值（MV）。
MV	REAL	MV 输出值	0.0~100.0 （单位：1%）	
PV	DWORD/DINT/REAL	当前值	数值格式依模块 HWCONFIG 参数设定	

操作数名称	数据类型	功能	设定范围	说明
I_MV	REAL	累积积分值	累积的积分值，供参考用。当 MV 超出 0~100%，I_MV 值不会再改变。	
ERROR	DWORD/DINT	错误代码	16#0000：指令正常运作 16#1400：模块不支持此指令 16#1401：群组编号或模块编号设定错误 16#1402：模块无响应，通讯逾时 16#1403：通道设定错误 16#1404：通讯正在执行 PID 功能，不允许重复设定。	

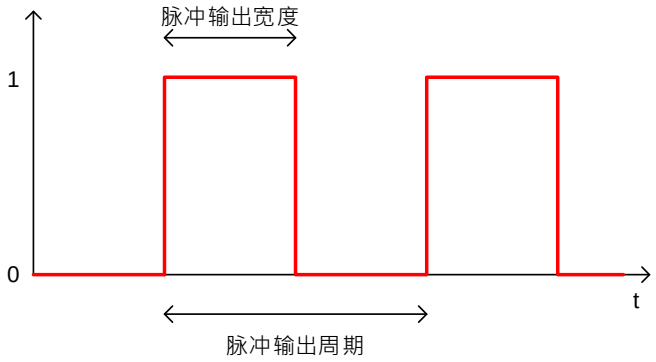
注：PID 参数若写入值超出参数上限，将只会写入上限值给模块；若写入值低于参数下限，将只会写入下限值给模块

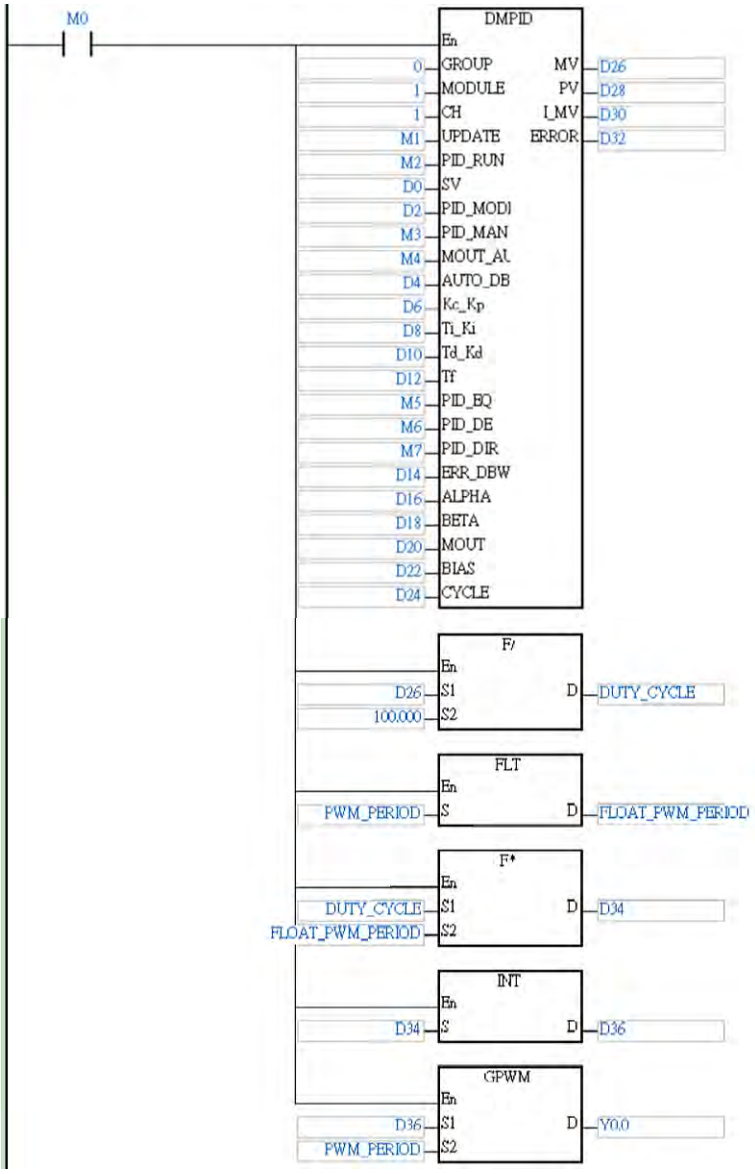
注意：当 PID_RUN 由 TRUE→FALSE 时，会自动清除 MV 输出值为 0。若希望 MV 输出值能够在 PID 不执行时，继续保持最后一次 MV 输出值，则可由 EN 从 TRUE->FALSE 去关闭指令。

程序范例：

1. 执行 DMPID 指令前先将参数设定完成，当指令 En 初次启动时，即会对模块设定好参数。执行中若要变更参数（包含 **PID_RUN** ~ **CYCLE**），可由设定 UPDATE 标志，指令完成参数变更后会自行清除 UPDATE 标志。
2. M0=ON 的时候指令被执行，M2=ON 的时候 DMPID 指令才开始进行运算，M2=OFF 时 MV 值为 0，MV 数值送至 D26。当 M0 变成 OFF 时，指令不被执行，原先指令内的参数值没有变化。
3. 若使用自动调整参数功能（**PID_MODE**=1），调整完毕时将自动进入自动控制模式（**PID_MODE** 自动改为 0），并且将调试出的 **Kc_Kp**、**Ti_Ki**、**Td_Kd**、**Tf**、**ALPHA 及 BETA** 参数填回至 DMPID 输入端对应的数据寄存器（D 装置），也可利用停电保持型之数据寄存器，即可达成保存 PID 控制参数的需求。
4. **MV** 输出值范围为 0.0~100.0，根据需求，用户需自行转换为所需的控制量。此应用例为转换成 PWM 输出。利用浮点数除法 F/指令将 **MV** 输出值转换成范围为 0%~100%的 Duty cycle，接着利用浮点数乘法 F*指令将 Duty cycle 与 PWM 周期相乘，即可计算脉冲输出宽度。
5. 将脉冲输出宽度及脉冲输出周期输入 GPWM 参数，即可在指定的脉冲输出装置实现 PWM 控制。

$$\text{Duty cycle} = \frac{\text{脉冲输出宽度}}{\text{脉冲输出周期}}$$


脉冲输出宽度
1
0
t
脉冲输出周期


M0
DMPID
En
0 GROUP
1 MODULE
1 CH
M1 UPDATE
M2 PID_RUN
D0 SV
D2 PID_MODI
M3 PID_MAN
M4 MOUT_AU
D4 AUTO_DB
D6 Kc_Kp
D8 Ti_Ki
D10 Td_Kd
D12 Tf
M5 PID_BQ
M6 PID_DE
M7 PID_DIR
D14 ERR_DBW
D16 ALPHA
D18 BETA
D20 MOUT
D22 BIAS
D24 CYCLE
MV D26
PV D28
I_MV D30
ERROR D32
F/
D26 S1
100.000 S2
D DUTY_CYCLE
FLT
PWM_PERIOD S
D FLOAT_PWM_PERIOD
F*
DUTY_CYCLE S1
FLOAT_PWM_PERIOD S2
D D34
INT
D34 S
D D36
GPWM
D36 S1
PWM_PERIOD S2
D Y0.0

PID 计算公式：

1. 当 PID_MODE 控制模式设定为 0 时，为自动控制模式。PID 计算公式如下

   - **Independent Formula & Derivative of E**（**PID_EQ**=False & **PID_DE**=False）

     $$MV = K_P E + Ki\int_0^t Edt + K_d * \frac{dE}{dt} + BIAS$$ **其中** *E = SV – PV* or *E = PV – SV*

   - **Independent Formula & Derivative of PV**（**PID_EQ**=False **& PID_DE**=True **)**

     $$MV = K_P E + Ki\int_0^t Edt - K_d * \frac{dPV}{dt} + BIAS$$ **其中** *E = SV – PV*

     Or

     $$MV = K_P E + Ki\int_0^t Edt + K_d * \frac{dPV}{dt} + BIAS$$ **其中** *E =PV – SV*

   - **Dependent Formula & Derivative of E**（**PID_EQ**=True & **PID_DE**=False）

     $$MV = K_c\left[E + \frac{1}{T_i}\int_0^t Edt + T_d * \frac{dE}{dt}\right] + BIAS$$ **其中** *E = SV – PV* or *E = PV – SV*

   - **Dependent Formula & Derivative of PV**（**PID_EQ**=True & **PID_DE**=True）

     $$MV = K_c\left[E + \frac{1}{T_i}\int_0^t Edt - T_d * \frac{dE}{dt}\right] + BIAS$$ **其中** *E = SV – PV*

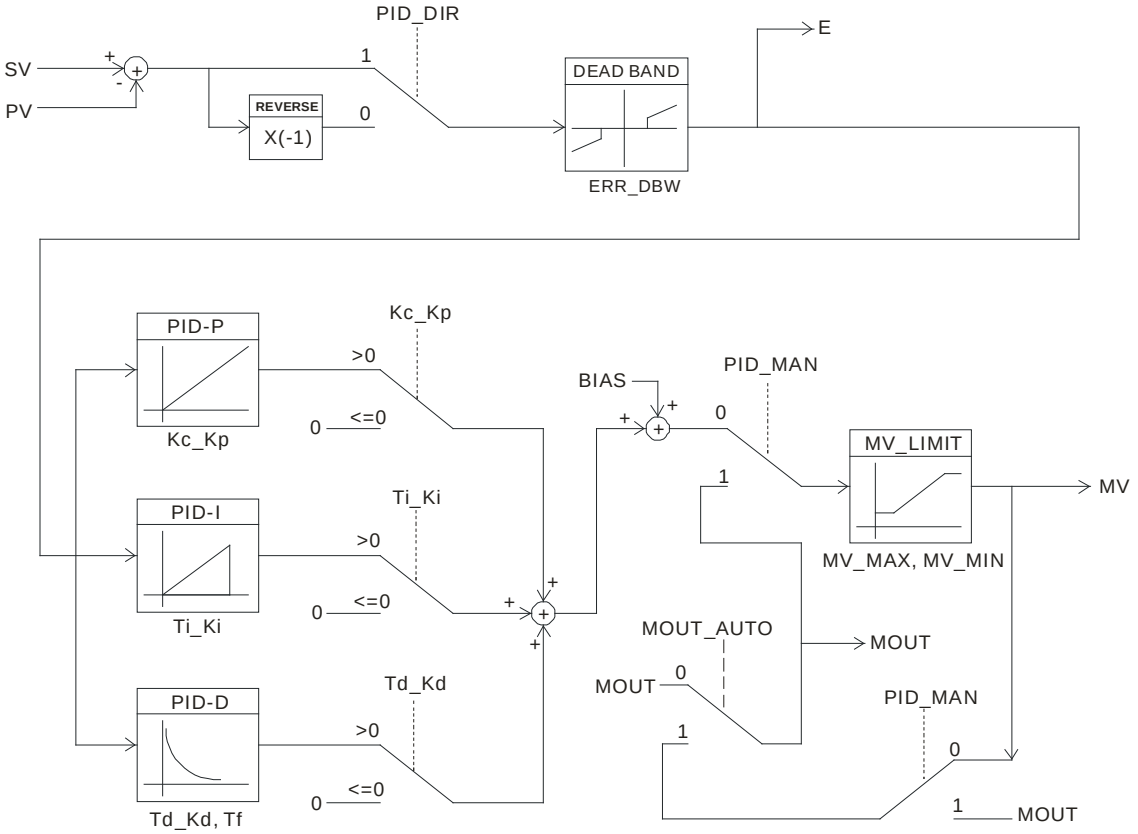
     Or

     $$MV = K_c\left[E + \frac{1}{T_i}\int_0^t Edt + T_d * \frac{dE}{dt}\right] + BIAS$$ **其中** *E = PV – SV*

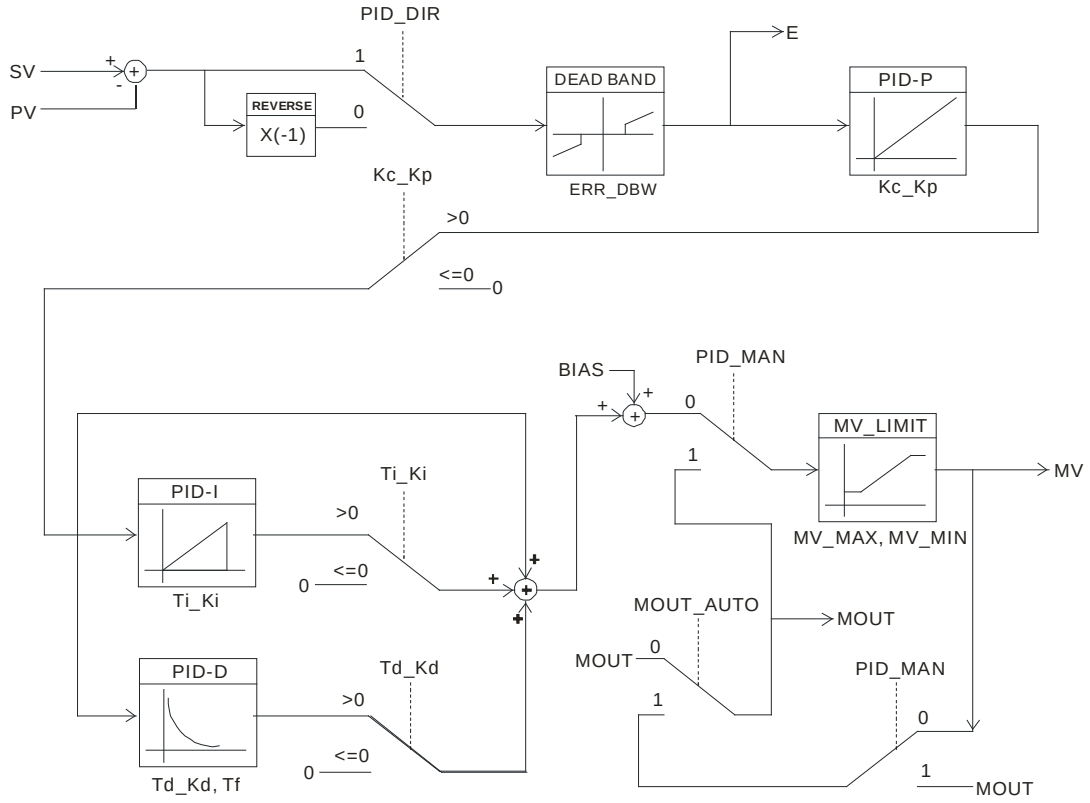
2. 当 PID_MODE 控制模式选择为 1，为自动调整模式，当自动调整完成后，PID_MODE 会自动变成 0 转换为自动控制模式。

PID 控制方块图：

PID Block Diagram (Independent)



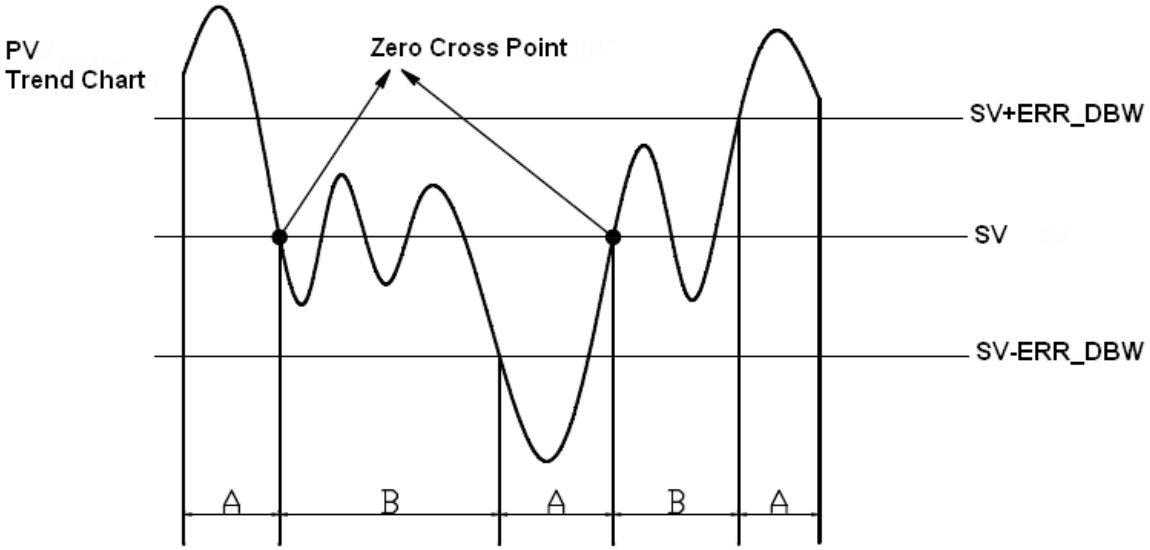
PID Block Diagram (Dependent)



偏差量不作用范围：

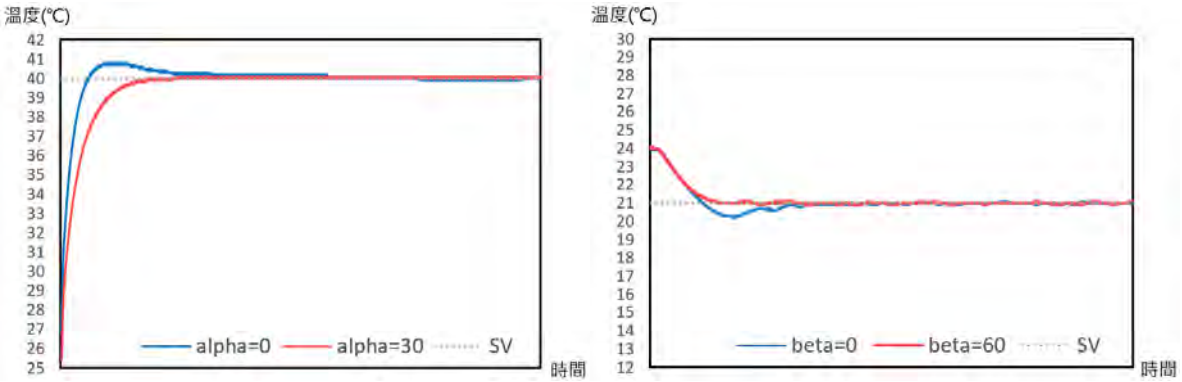
当PV值进入ERR_DBW的范围时，一开始主机仍会依照E值进行PID计算，直到PV穿过SV值时（Zero Cross Point）代表Cross Status成立，此时会将E值视为0代入PID计算，一直到PV值超出ERR_DBW的范围时才会恢复将E值代入PID计算，若PID_DE=True则表示使用PV值来进行微分项的计算，则在Cross Status条件成立后，主机会将Delta PV视为0进行PID微分项的计算。（Delta PV=当前PV-前次PV）

例如：以下的PV趋势图中，A的区段主机会依照正常的PID进行计算，而B的区段主机会将E或Delta PV视为0进行PID计算。



α、β VALUE：

ALPHA 以及 BETA 作用是在 PID 启动时以及 SV 目标值变动时，产生初始积分量的补偿，效果为减少 overshoot 过冲现象，如下图所示，ALPHA 参数作用为减缓上升 overshoot 过冲，BETA 参数作用为减缓下降 overshoot 过冲。



输出周期：

用户依控制环境先决定输出周期（若环境温度变化慢，输出周期可调大些），

脉冲输出宽度如下：

- 脉冲输出宽度= MV 输出值（%）×输出周期

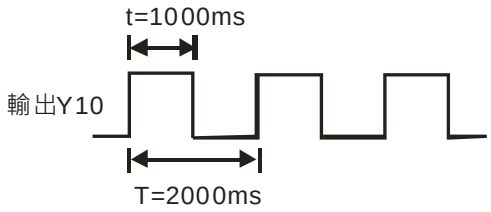
使用 CPU 模块之 GPWM 指令作脉冲输出宽度及输出周期做周期控制。

范例：

设定输出周期为 2000ms，当 PID 运算后，MV 输出值为 50%则

- 脉冲输出宽度= 50 %×2000ms = 1000ms

故 GPWM 参数设定为脉冲输出宽度=1000，输出周期=2000



注意事项和建议：

1. 用户于调整 Kc_Kp、Ti_Ki 及 Td_Kd 三个主要参数时（**PID_MODE**=0），请先调整 Kc_Kp 值（依经验值设定），而 Ti_Ki 及 Td_Kd 值先设定为 0，等到调整到大致上可控制时，再依序调整 Ti_Ki 值（由小到大）以及 Td_Kd 值（由小到大）。其中 Kc_Kp 值为 1 则表示 100%，即对偏差值的增益为 1，小于 100%将对偏差值衰减，大于 100%将对偏差值放大。
2. 经过自动调整过的参数，并不能保证一定适用于每个控制的环境，因此用户当然可自行修改调整过的参数，不过建议最好只修改 Ti_Ki 或 Td_Kd 数值就好。
3. 参数 CYCLE 为 PID 功能多少时间去计算一次，并更新输出值（MV）。
4. 须注意当开启测量的通道数目有变化时，测量值的更新时间将有变化（EX：只开启单一通道做测量时，此测量值每 200ms 更新一次；当另外开启其它三通道做测量时，此测量值为 800ms 更新一次），相同的 Kc_Kp、Ti_Ki 及 Td_Kd 等参数不一定可以适用。

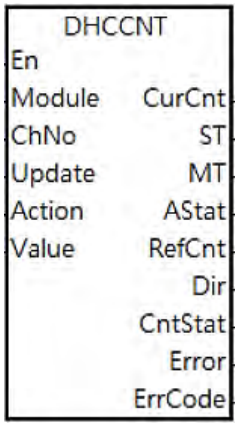
API	指令码			操作数	功能
1418	D	HCCNT		**Module ~ ErrCode**	HC 模块计数器控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
ChNo								●					○	○		
Update		●	●	●				●								
Action								●					○	○		
Value							●	●					○	○		
CurCnt							●	●								
ST								●								
MT								●								
AStat								●								
RefCnt								●								
Dir		●	●	●				●								
CntStat								●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
Update	●												
Action		●				●							
Value			●				●						
CurCnt			●				●						
ST			●				●						
MT			●				●						
AStat		●				●							
RefCnt		●				●							
Dir	●												
CntStat		●				●							
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

ChNo ： 通道编号

Update ： 更新参数标志

Action ： 计数器动作设定

Value ： 新计数值/偏移量存放来源

CurCnt ： 当前计数值显示

ST ： 绝对型编码器 SingleTurn Data

MT ： 绝对型编码器 MultiTurn Data

AStat ： 绝对型编码器 Status Data

RefCnt ： 绝对型编码器数据刷新计数器

Dir ： 计数方向显示

CntStat ： 计数器状态

Error ： 错误标志

ErrCode ： 错误代码

指令说明：

1. 此指令为 AS02HC-A 专用指令，功能包含启动/关闭计数器、以及设定/修改计数值。指令需搭配 PLC 主机韧体 V1.08（含）版以上，以及 ISPSoft 软件 V3.10（含）版以上。
2. 执行此指令前先将 **Action** 与 **Value** 填入设定值，当指令 **En** 初次启动时，将会先对 HC 模块设定 **Action** 与 **Value** 一次；当执行中需要再次变更 **Action** 与 **Value** 时，先将 **Action** 与 **Value** 设定为新数值，再设定 **Update** 标志为 On；当此指令完成变更之后，指令将会自行清除 **Update** 标志为 Off。
3. **Module** 指定模块编号，连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
4. **ChNo** 指定欲控制之计数器通道编号，通道 1 编号为 1，通道 2 编号为 2。
5. **Action** 计数器动作模式设定，以及 **Value** 使用说明如下表

Action	功能说明	批注
0	保持现有模式不变更	
1	设定/变更当前计数值 **CurCnt**	新设定的计数值，请由 **Value** 数值设定。 注意：当使用 SSI 编码器且计数器形式设定为「绝对位置（Absolute Position）」时，此 **Action** 无效。
2	设定 SSI 编码器偏移量为 **Value**	当计数器形式设定为「绝对位置（Absolute Position）」时，可自行设置 SSI 编码器计数值的偏移量，通过 **Value** 设定偏移量，其计数值 **CurCnt** = 编码器原始计数值 + **Value**。

Action	功能说明	批注
2	设定 SSI 编码器偏移量为 **Value**	注意： 1. **Value** 值范围：$-2^{(MT+ST\ 长度-1)} \leqq$ **Value**$<2^{(MT+ST\ 长度-1)}$，超出范围将不允许设定。 2. 当使用 SSI 编码器且计数器形式设定为「环形计数器（Ring counter）」，此 **Action** 无效。 3. 当重新下载 HWCONFIG 配置时，偏移量将被清除为 0。
3	设定/变更 SSI 编码器绝对位置值	当计数器形式设定为「绝对位置（Absolute Position）」时，SSI 编码器计数值将会自动偏移调整至 **Value** 值，其计数值 **CurCnt** = **Value**。 注意： 1. **Value** 值范围：0≦ **Value**$<2^{(MT+ST\ 长度)}$，超出范围将不允许设定。 2. 当使用 SSI 编码器且计数器形式设定为「环形计数器（Ring counter）」，此 **Action** 无效，请使用 **Action=1** 变更计数值。 3. 当重新下载 HWCONFIG 配置时，偏移量将被清除为 0。
4	清除（RESET）当前计数器数值 **CurCnt**	1. 清除 **CurCnt** 为 0。 2. 清除表格比较输出指令 DHCCMPT 的 **CurNo**。 3. 清除比较输出指令 DHCCMP 的 **Match1**、**Match2** 标志。 注意：当使用 SSI 编码器，且计数器形式设定为「绝对位置」时，将无法清除计数值，但会清除 **CurNo**、**Match1**、**Match2** 标志。
5	清除（RESET）**CurCnt** 当前计数器数值 **CurCnt**，以及清除表格比较输出指令/比较输出指令所指定的 Y 输出点	除了上述 **Action**=4 清除内容之外，亦连带清除表格比较输出指令/比较输出指令所指定的 Y 输出点（ON➔OFF）。
6	预置（PRESET）当前计数器数值 **CurCnt**	1. **CurCnt** 被指定修改为 **Value**。 2. 清除表格比较输出指令 DHCCMPT 的 **CurNo**，注意 Preset 之后，DHCCMPT 将会从第一笔比较值等待比较到达，因此若 Preset 后之计数值大于第一笔比较值将无法正确进行表格比较，若须使表格比较功能从 Preset 后之计数值开始做比较，请执行 Preset 后再将 DHCCMPT 的 Update 标志设定为 ON 即可。 3. 清除比较输出指令 DHCCMP 的 **Match1/2** 标志。
6	预置（PRESET）当前计数器数值 **CurCnt**	注意：当使用 SSI 编码器，且计数器形式设定为「绝对位置」时，将无法预置计数值，但会清除 **CurNo**、**Match1**、**Match2** 标志。

Action	功能说明	批注
7	预置（PRESET）当前计数器数值 **CurCnt** ,以及清除表格比较输出指令/比较输出指令所指定的 Y 输出点	除了上述 **Action**=6 清除内容之外，亦连带清除表格比较输出指令/比较输出指令所指定的 Y 输出点（ON➔OFF）。

6. **CurCnt** 为显示指定模块计数通道之当前计数值。
7. **ST** 为绝对型 SSI 编码器 Single-Turn Data 显示（当计数器形式设定为「绝对位置」时，**ST** 将显示经偏移过后之数值；当设定计数器为环形计数器时，**ST** 将显示原始 SSI 编码器数值），当通道模式设为增量型编码器时，此 **ST** 将为 0。
8. **MT** 为绝对型 SSI 编码器 Multi-Turn Data 显示（当计数器形式设定为「绝对位置」时，**MT** 将显示经偏移过后之数值；当设定计数器为环形计数器时，**MT** 将显示原始 SSI 编码器数值），当通道模式设为增量型编码器时，**MT** 将为 0。
9. **AStat** 为绝对型 SSI 编码器 Status Data 显示，当通道模式设为增量型编码器时，**AStat** 将为 0。
10. **RefCnt** 为绝对型 SSI 编码器数据刷新计数器，长度为 16 位，当新获取一笔 SSI 数据后，刷新计数器将加 1，此计数器溢位后将由零重新累计。当通道模式设为增量型编码器时，**RefCnt** 将为 0。
11. **Dir** 为计数方向显示，当 **Dir** 显示 On 表示往正方向，当 **Dir** 显示 Off 表示往反方向。
12. **CntStat** 为计数器状态，各 bit 对应之项目如下表：

位编号	状态名称	状态说明	备注
15 ~ 11	Reserved		保留
10	SSI 绝对位置跨越零点	0：无异常 1：异常发生	由 HWCONFIG 设置开启或关闭检测，默认为不检测。 计数器类型为「绝对位置」时：当用户有设定偏移量时，将以偏移量调整后之位置值执行过零点检测。 计数器类型为「环形计数器」时：由于偏移量将被重设为零，将以编码器原始位置值执行过零点检测。 错误发生原因：过零点发生 清除方式：由 DHCCNT 执行清除（RESET）或预置（PRESET）命令清除，或由 Z 相输入点执行清除（RESET）
9	SSI 通讯异常	0：无异常 1：异常发生	错误发生原因：通讯异常连续 5 次 清除方式：通讯恢复正常
8	SSI 同位检查错误	0：无异常 1：异常发生	错误发生原因：同位检查错误 清除方式：下一笔读值同位检查正确
7	SSI 数据变动大小超出保护设定	0：无异常 1：异常发生	错误发生原因：两笔位置数据变动大小超出保护设定 清除方式：下一笔数据变动大小在合理范围内

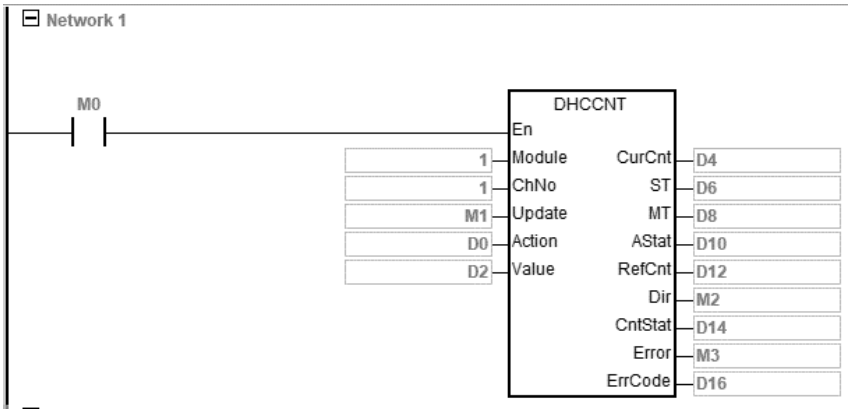
位编号	状态名称	状态说明	备注
6	Reserved	保留	
5	环形计数 Overflow	0：无异常 1：异常发生	由 HWCONFIG 设置开启或关闭检测，默认为不检测 错误发生原因：当计数器 Overflow($>2^{31}-1$)或 Underflow（$<-2^{31}$）发生 清除方式：由 DHCCNT 指令执行清除（RESET）或预置（PRESET)命令清除，或由 Z 相输入点执行清除(RESET)
4	环形计数 Underflow	0：无异常 1：异常发生	
3	线性计数高于设定上限	0：无异常 1：异常发生	错误发生原因：线性计数超出用户设定范围 清除方式：线性计数值回到上下限范围内 注：计数超出上下限后，计数值维持在上限或下限，当计数值回到上下限范围内则恢复计数。
2	线性计数低于设定下限	0：无异常 1：异常发生	
1	线性计数 Overflow	0：无异常 1：异常发生	错误发生原因：线性计数超出上下限后，硬件计数器持续计数，当硬件计数器 Overflow($>2^{31}-1$)或 Underflow($<-2^{31}$) 发生则不再计数，需由下述清除方式执行后，才可恢复计数。 清除方式：由 DHCCNT 指令执行清除（RESET）或预置（PRESET)命令清除，或由 Z 相输入点执行清除(RESET)
0	线性计数 Underflow	0：无异常 1：异常发生	

13. 若指令关闭时，该指定通道将停止更新指令右半部各项数值。
14. 当 **Error** 错误标志为 On 时，请参考 **ErrCode** 错误代码如下表说明：

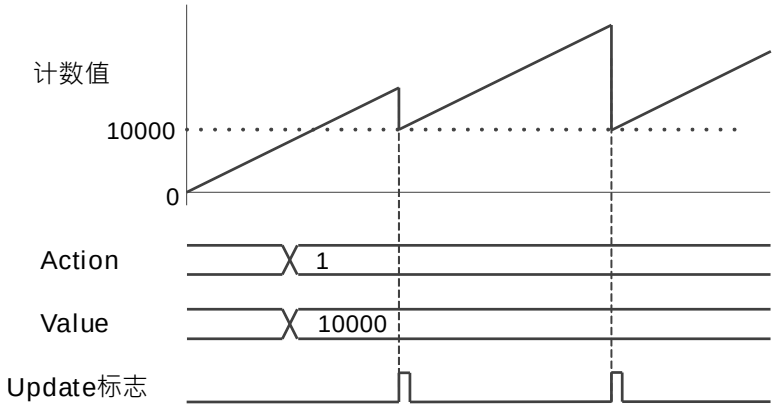
ErrCode 代码	错误状况说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	HWCONFIG 配置错误，指令禁止运作
16#1404	HC 模块无此计数器通道
16#1405	输入接口未选择，指令禁止运作
16#1406	Action 数值不合法
16#1410	HC 模块指定之计数器通道正在执行计数当中，不允许重复指定计数

程序范例 1 - 设定/变更当前计数值（Action = 1）

1. 设定 M0=ON 时，计数器开始计数。
2. 将 DHCCNT 指令的 **Action** 参数设定为 1、**Value** 参数设定为 10000 ;当用户设定 **Update** 标志 M1=ON 时，当前计数值 **CurCnt** 将变更为 10000。
3. 设定完成 **Update** 标志 M1 将自动清除为 OFF。

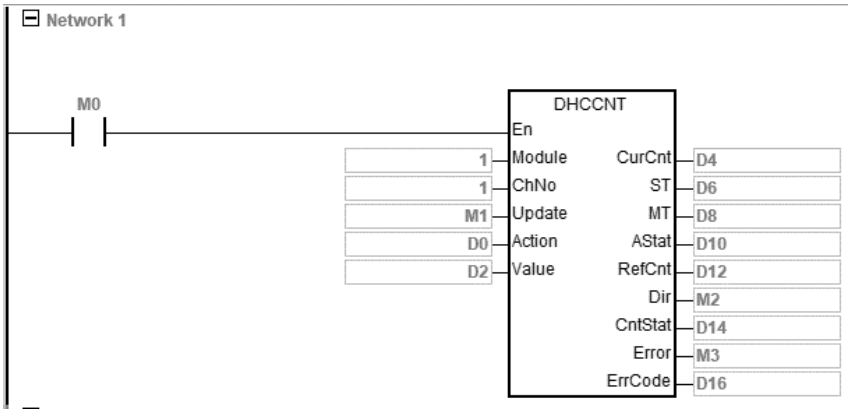


动作时序图：

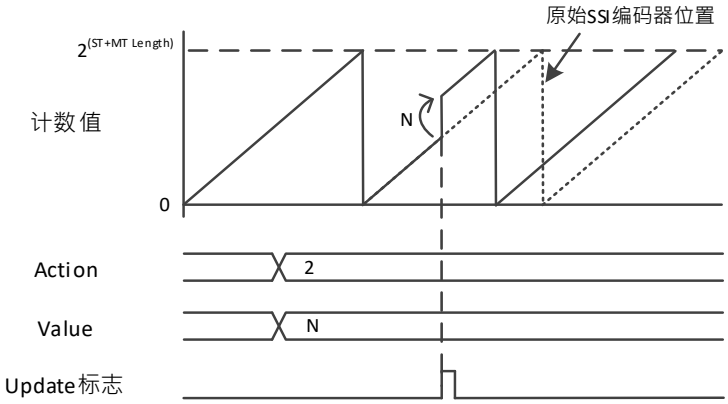


程序范例 2 - 设定绝对型 SSI 编码器偏移量（Action = 2）

1. HWCONFIG 设定通道 1 为「SSI 输入」，且计数器形式选择为「绝对位置」。
2. 设定 M0=ON 时，计数器开始计数。
3. 将 DHCCNT 指令的 **Action** 参数设定为 2、**Value** 参数设定为 500 ;假设当当前计数值 **CurCnt** 为 2500 时，用户设定 **Update** 标志 M1=ON，当前计数值 **CurCnt** 将变更为 3000。
4. 设定完成 **Update** 标志 M1 自动清除为 OFF。

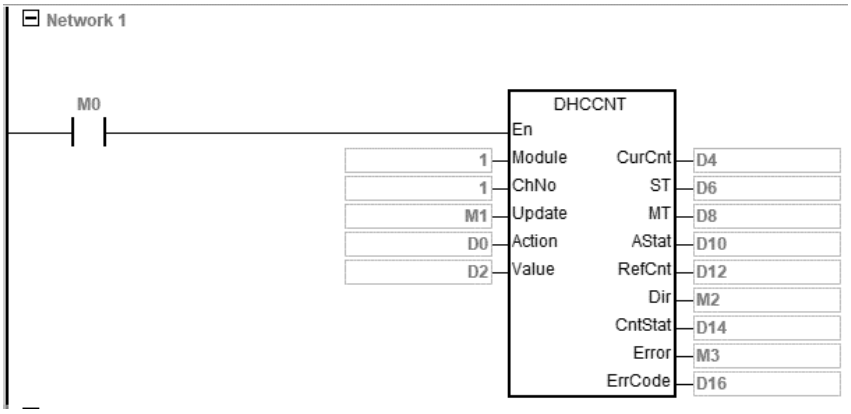


动作时序图：

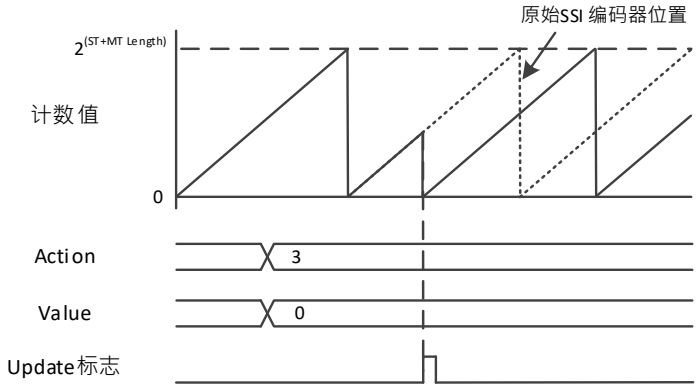


程序范例 3 - 设定/变更 SSI 编码器绝对位置值（Action = 3）

1. HWCONFIG 设定通道 1 为「SSI 输入」，且计数器形式选择为「绝对位置」。
2. 设定 M0=ON 时，计数器开始计数。
3. 将 DHCCNT 指令的 **Action** 参数设定为 3、**Value** 参数设定为 0；当用户设定 **Update** 标志 M1=ON 时，HC 模块将自动计算偏移量，将当前计数值 **CurCnt**（编码器绝对位置）偏移至 0。
4. 设定完成 **Update** 标志 M1 自动清除为 OFF。

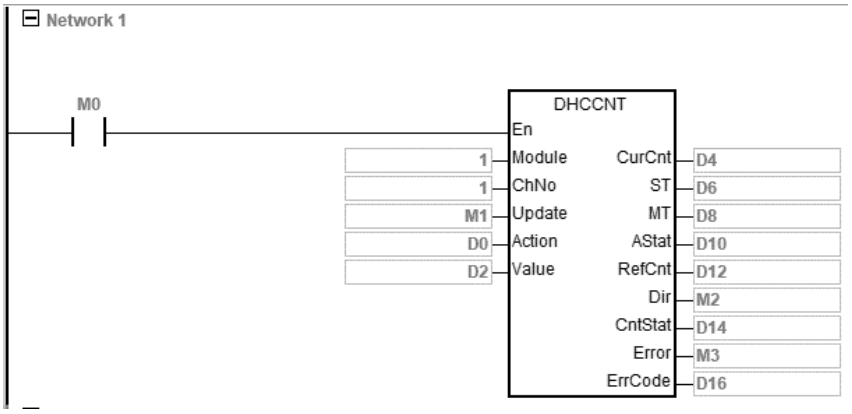


动作时序图：

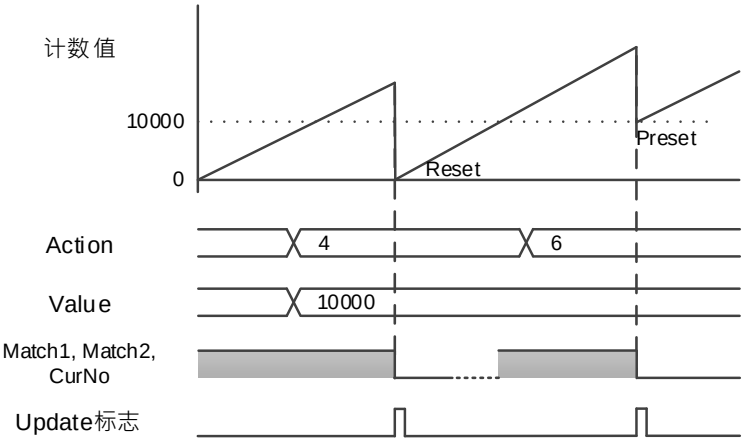


程序范例 4 - 清除（RESET）或预置（PRESET）当前计数器数值（Action = 4、6）

1. 设定 M0=ON 时，计数器开始计数。
2. 将 DHCCNT 指令的 **Action** 参数设定为 4，当用户设定 **Update** 标志 M1=ON 时，当前计数值 **CurCnt** 将被 RESET 为 0，设定完成 **Update** 标志 M1 将自动清除为 OFF。
3. 将 DHCCNT 指令的 **Action** 参数设定为 6、**Value** 参数设定为 10000 ;当用户设定 **Update** 标志 M1=ON 时，当前计数值 **CurCnt** 将被 PRESET 为 10000，设定完成 **Update** 标志 M1 将自动清除为 OFF。



动作时序图：



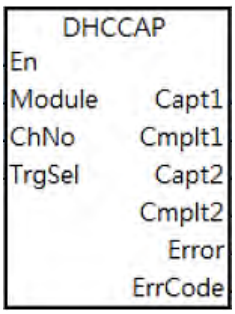
API	指令码			操作数	功能
1419	D	HCCAP		**Module ~ ErrCode**	HC 模块计数值获取

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
ChNo								●					○	○		
TrgSel								●					○	○		
Capt1							●	●								
Cmplt1		●	●	●				●								
Capt2							●	●								
Cmplt2		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
TrgSel		●				●							
Capt1			●				●						
Cmplt1	●												
Capt2			●				●						
Cmplt2	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module : 模块编号

ChNo : 通道编号

TrgSel : TRIGGER 方式选择

Capt1 : Capturer 计数值 1

Cmplt1 : Capt1 获取完成标志

Capt2 : Capturer 计数值 2

Cmplt2 : Capt2 获取完成标志

Error : 错误标志

ErrCode : 错误代码

指令说明：

1. DHCCAP 指令为 AS02HC-A 专用指令，功能为设定获取方式，以及显示获取之计数值。指令需搭配 PLC 主机韧体 V1.08（含）版以上，以及 ISPSoft 软件 V3.10（含）版以上。
2. DHCCAP 需搭配 DHCCNT 指令使用，只有当 DHCCNT 启动时，计数值才会依输入信号计数，DHCCAP 获取之计数值才有效；DHCCNT 关闭时，计数将停止接收输入信号而停止更新计数值，此时获取之计数值将不会变化。
3. 执行此指令前先将 **TrgSel** 设定完成，当指令 **En** 初次启动时，将会先对 HC 模块设定 **TrgSel** 一次；当执行中需要再次变更 **TrgSel** 时，请关闭指令后再重新启动。
4. **Module** 指定模块编号，连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
5. **ChNo** 指定欲控制之计数器通道编号，通道 1 编号为 1，通道 2 编号为 2。
6. **TrgSel** 为计数值获取之触发方式选择

<table>
<tr><th>TrgSel</th><th>计数值获取触发方式</th><th>备注</th></tr>
<tr><td>0</td><td>由指定通道的 Z 相输入点触发</td><td><table><tr><th>ChNo</th><th>触发来源</th></tr><tr><td>1</td><td>CH1 Z 相</td></tr><tr><td>2</td><td>CH2 Z 相</td></tr></table>需搭配 HWCONFIG 配置該通道的 Z 相输入点功能选择为「获取（Capture）」功能</td></tr>
<tr><td>1</td><td>由另一通道之比较输出指令 DHCCMP 的 Match1、Match2 触发</td><td>举例：当通道 2 启动 DHCCAP 且设定 TrgSel=1，而指令通道 1 被使用于比较输出指令 DHCCMP 则当通道 1 比较到达 Match1、Match2 时，将分别触发通道 2 获取当下计数值，并分别储存于 Capt1 或 Capt2 内。
注意：TrgSel=1 时，即使 HWCONFIG 的外部输入点功能选择为「获取（Capture）」，Z 相输入点的获取功能将视为无效。</td></tr>
</table>

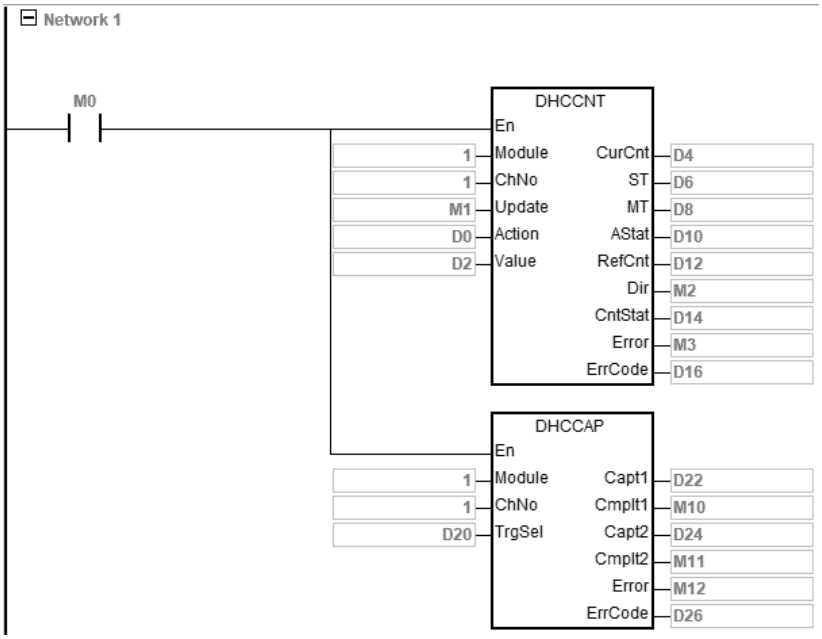
7. **Capt1** 与 **Capt2** 分别为获取计数值 1 与获取计数值 2 显示；当 **TrgSel**=0，**Capt1** 存放 Z 相输入点上升沿触发（Rising edge）时之计数值，**Capt2** 存放 Z 相输入点下降沿触发（Falling edge）时之计数值；当 **TrgSel**=1，**Capt1** 存放另一通道比较输出指令的 Match1 由 Off→On 时之计数值，**Capt2** 存放另一通道比较输出指令的 Match2 由 Off→On 时之计数值。
8. **Cmplt1** 与 **Cmplt2** 分别为完成获取计数值 1、获取计数值 2 的标志，当 **Cmplt1/Cmplt2** 由 Off->On 时，表示 **Capt1/Capt2** 已经为最新获取值，用户读取获取值后须自行清除 **Cmplt1/Cmplt2** 标志，当下一次 **Cmplt1**、**Cmplt2** 由 Off->On 时，即可得知又有新获取值。若用户未清除 **Cmplt1**、**Cmplt2** 标志，模块最新的获取值依然会持续更新到 **Capt1**、**Capt2**。
9. 若 **En** 由 On 变为 Off 时，表示关闭指令的 Capture 功能，此时 **Capt1**、**Capt2** 内容值将保持不变且不再更新，但 **Cmplt1**、**Cmplt2** 标志将被清除。
10. 当 **Error** 错误标志为 On 时，请参考 **ErrCode** 错误代码如下表说明：

ErrCode 代码	错误状况说明
16#1400	模块不支持此功能

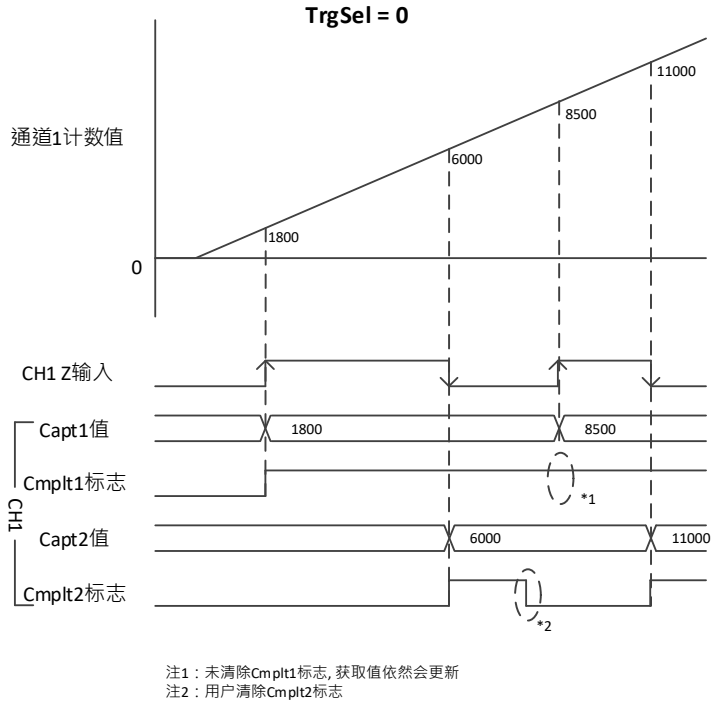
ErrCode 代码	错误状况说明
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	HWCONFIG 配置错误，指令禁止运作
16#1404	HC 模块无此计数器通道
16#1405	输入接口未选择，指令禁止运作
16#1407	TrgSel 数值不合法
16#1411	HC 模块指定之计数器通道正在执行获取当中，不允许重复指定获取

程序范例 1 - 由 Z 相输入点获取计数值

1. 设定 HWCONFIG 中通道 1 的 Z 相输入点功能选择为「获取（Capture）」功能。
2. 将 DHCCAP 指令的 **TrgSel** 参数设定为 0，当 M0=ON 时，DHCCAP 指令将开始等待 Z 相输入点的上升沿/下降沿动作。
3. 当通道 1 的 Z 相输入点的上升沿触发发生，此时计数值 1800 将立即输出到 **Capt1**，同时，**CmpIt1** 标志也被设定为 ON。
4. 当通道 1 的 Z 相输入点的下降沿触发发生，此时计数值 6000 将立即输出到 **Capt2**，同时，**CmpIt2** 标志也被设定为 ON。用户可将 **CmpIt2** 标志清除为 OFF，当下一次下降沿触发再次获取数值时，**CmpIt2** 标志将再次被设定为 ON。
5. 即使 **CmpIt** 标志未清除为 OFF，有新获取值时，新获取值依然会实时输出到 **Capt1**、**Capt2**；例如，当再一次 Z 相输入点的上升沿触发发生，新的获取值 8500 依然会立即输出到 **Capt1**。
6. 而接着当再一次 Z 相输入点的下降沿触发发生，新获取值 11000 将立即输出到 **Capt2**，同时，**CmpIt2** 标志也再次被设定为 ON。



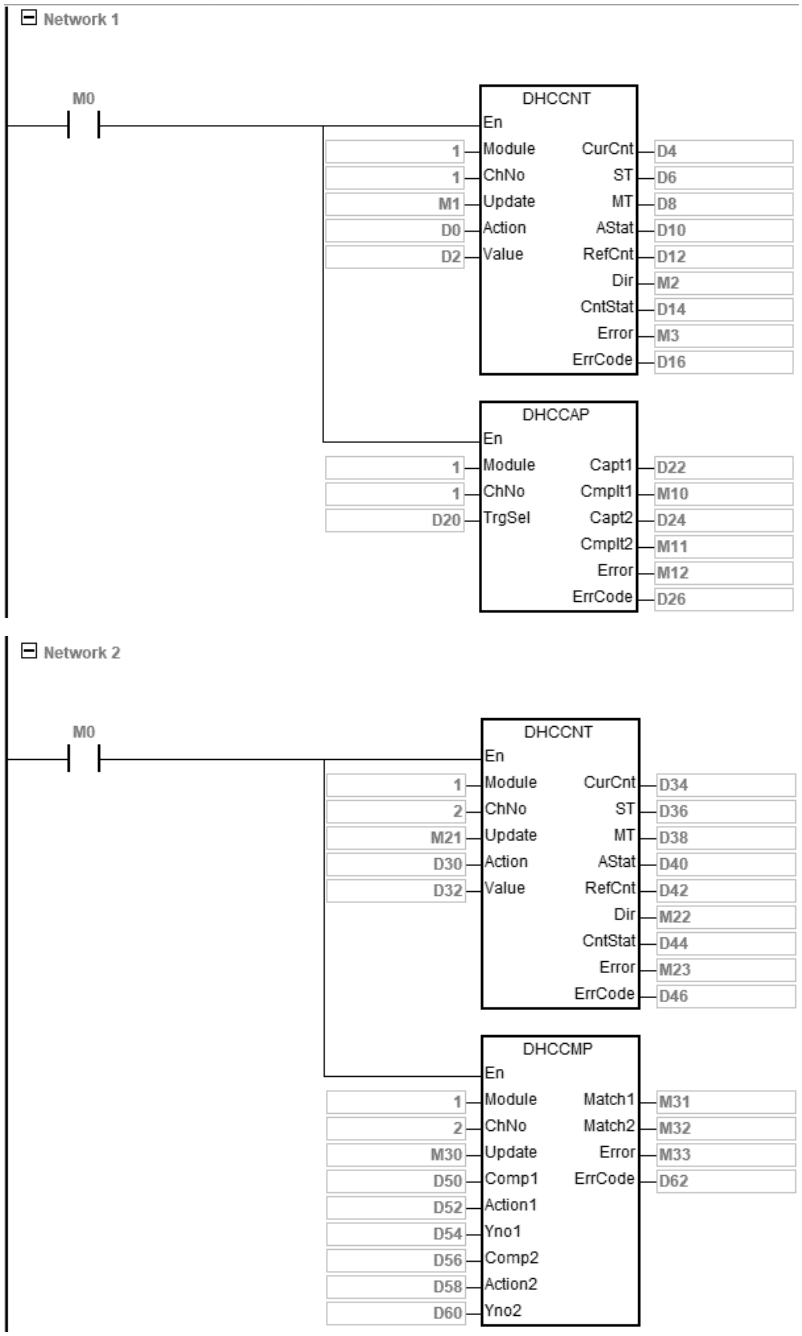
动作时序图：



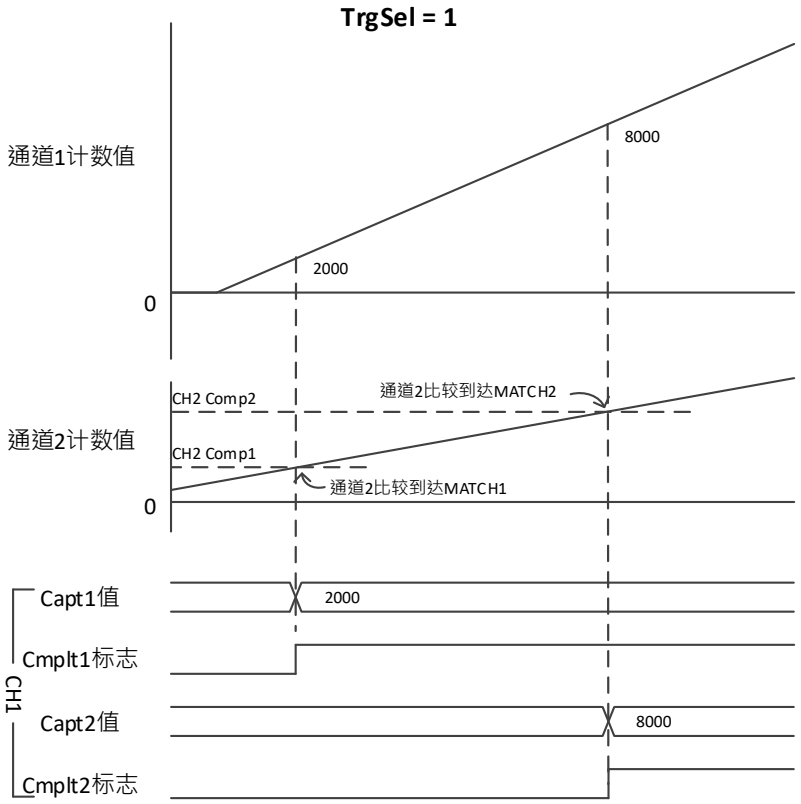
注1：未清除CmpIt1标志, 获取值依然会更新
注2：用户清除CmpIt2标志

程序范例 2 – 另一通道 DHCCMP 比较到达时获取计数值

1. 将通道 1 中 DHCCAP 指令的 **TrgSel** 参数设定为 1，当 M0=ON 时，DHCCAP 指令将开始等待另一通道(通道 2) 的比较到达发生。即使此时 HWCONFIG 中通道 1 的 Z 相输入点功能选择为「获取 (Capture)」，此 Z 相输入点的获取功能将视为无效。
2. 当通道 2 计数值到达比较值 **Comp1**，此时通道 1 的计数值 2000 将立即输出到 **Capt1**，同时，**CmpIt1** 标志也被设定为 ON。
3. 当通道 2 计数值到达比较值 **Comp2**，此时计数值 8000 将立即输出到 **Capt2**，同时，**CmpIt2** 标志也被设定为 ON。
4. 如同范例 1，即使 **CmpIt** 标志未清除为 OFF，当通道 2 再次有比较到达事件发生，新获取值依然会实时输出到 **Capt**。


Network 1
M0
DHCCNT
En
1 Module
1 ChNo
M1 Update
D0 Action
D2 Value
CurCnt D4
ST D6
MT D8
AStat D10
RefCnt D12
Dir M2
CntStat D14
Error M3
ErrCode D16
DHCCAP
En
1 Module
1 ChNo
D20 TrgSel
Capt1 D22
CmpIt1 M10
Capt2 D24
CmpIt2 M11
Error M12
ErrCode D26
Network 2
M0
DHCCNT
En
1 Module
2 ChNo
M21 Update
D30 Action
D32 Value
CurCnt D34
ST D36
MT D38
AStat D40
RefCnt D42
Dir M22
CntStat D44
Error M23
ErrCode D46
DHCCMP
En
1 Module
2 ChNo
M30 Update
D50 Comp1
D52 Action1
D54 Yno1
D56 Comp2
D58 Action2
D60 Yno2
Match1 M31
Match2 M32
Error M33
ErrCode D62

动作时序图：


TrgSel = 1
通道1计数值
8000
2000
0
通道2计数值
CH2 Comp2
通道2比较到达MATCH2
CH2 Comp1
通道2比较到达MATCH1
0
CH1
Capt1值
2000
CmpIt1标志
Capt2值
8000
CmpIt2标志

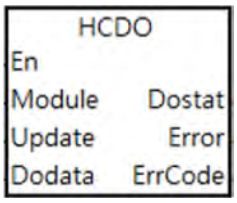
API	指令码	操作数	功能
1420	HCDO	**Module ~ ErrCode**	HC 模块输出点控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
Update		●	●	●				●								
DOdata								●					○	○		
DOstat								●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Update	●												
DOdata		●				●							
DOstat		●				●							
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Module ： 模块编号

Update ： 更新参数标志

DOdata ： 输出点动作设定

DOstat ： 输出点状态显示

Error ： 错误标志

ErrCode ： 错误代码

指令说明：

1. HCDO 指令为 AS02HC-A 专用指令，可以控制输出点 Y0.0~Y0.3 以及显示输出点 Y0.0~Y0.3 状态。指令需搭配 PLC 主机韧体 V1.08（含）版以上，以及 ISPSoft 软件 V3.10（含）版以上。
2. 执行指令前先将 **DOdata** 设定完成，**En** 被启动时将会先对 HC 模块设定 **DOdata** 一次，因此启动前设定之 **DOdata** 即作为输出点 Y0.0~Y0.3 的初始状态；执行中若要变更输出点状态，先将 **DOdata** 设定为新数值，再设定 **Update** 标志为 On，指令完成参数变更后会自行清除 **Update** 标志。
3. **Module** 指定模块编号，连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。

4. **DOdata** 为输出点动作设定值：

b15~b4	b3	b2	b1	b0
NA	Y0.3 动作	Y0.2 动作	Y0.1 动作	Y0.0 动作
NA	0：OFF 1：ON			

5. **DOstat** 为输出点状态显示

b15~b4	b3	b2	b1	b0
NA	Y0.3 状态	Y0.2 状态	Y0.1 状态	Y0.0 状态
NA	0：OFF 1：ON			

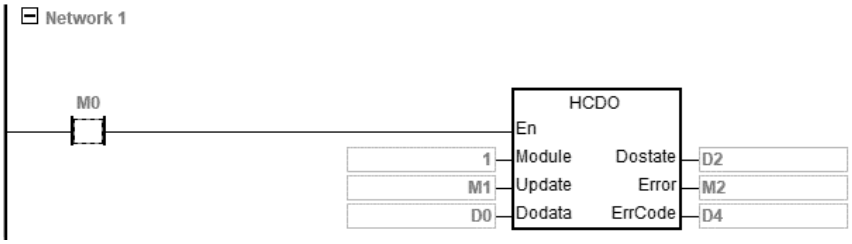
6. 若指令关闭时，将停止更新 **DOstat**。
7. 当 **Error** 错误标志为 On 时，请参考 **ErrCode** 错误代码如下表说明：

ErrCode 代码	错误状况说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	HWCONFIG 配置错误，指令禁止运作
16#1412	HC 模块输出点正在被其他 HC 输出指令控制中，不允许重复控制（**DOstat** 的状态依然可持续更新）

8. 当比较输出指令 DHCCMP 或表格比较输出指令 DHCCMPT 启动时，将无法通过 **DOdata** 变更输出点，但是显示 **DOstat** 的状态可持续更新。
9. 承上，若比较输出指令需要对输出点进行初始值的设定，则可先启动 HCDO 指令，然后再启动 DHCCMP 或 DHCCMPT 指令。若需变更输出点，请先关闭 DHCCMP 或 DHCCMPT 指令，即可由 HCDO 指令之 **DOdata** 更新最新输出动作。

程序范例：设定外部输出点

1. 首先决定 Y0.0~Y0.3 的初始状态，此范例中 Y0.0~Y0.3 的初始状态皆欲设定为 ON，因此在 M0 启动前先将 HCDO 指令的 **DOData** 输出点动作设定为 15。
2. 当 M0=ON 时，Y0.0~Y0.3 立即 ON。
3. DOStat 为输出点状态显示，此时为 15。在 M0=ON 的期间，若 Y0.0~Y0.3 由于 DHCCMP 或 HCCMPT 的控制而变换了状态，DOStat 将实时更新数值。



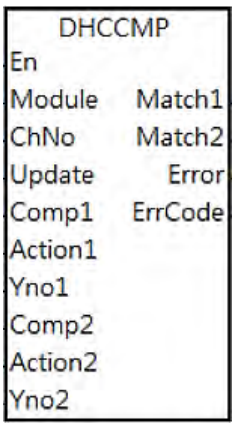
API	指令码			操作数	功能
1421	D	HCCMP		**Module ~ ErrCode**	HC 模块比较输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F	I
Module								●					○	○			
ChNo								●					○	○			
Update		●	●	●				●									
Comp1							●	●					○	○			
Action1								●					○	○			
Yno1								●					○	○			
Comp2							●	●					○	○			
Action2								●					○	○			
Yno2								●					○	○			
Match1		●	●	●				●									●
Match2		●	●	●				●									●
Error		●	●	●				●									
ErrCode								●									

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
Update	●												
Comp1			●				●						
Action1		●				●							
Yno1		●				●							
Comp2			●				●						
Action2		●				●							
Yno2		●				●							
Match1	●												
Match2	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

ChNo ： 通道编号

Update ： 更新参数标志

Comp1 ： 比较值 1

Action1 ： 到达比较值 1 时之动作

Yno1 ： 到达比较值 1 时 Y 输出点编号

Comp2 ： 比较值 2

Action2 ： 到达比较值 2 时之动作

Yno2 ： 到达比较值 2 时 Y 输出点编号

Match1 ： 到达比较值 1 标志

Match2 ： 到达比较值 2 标志

Error ： 错误标志

ErrCode ： 错误代码

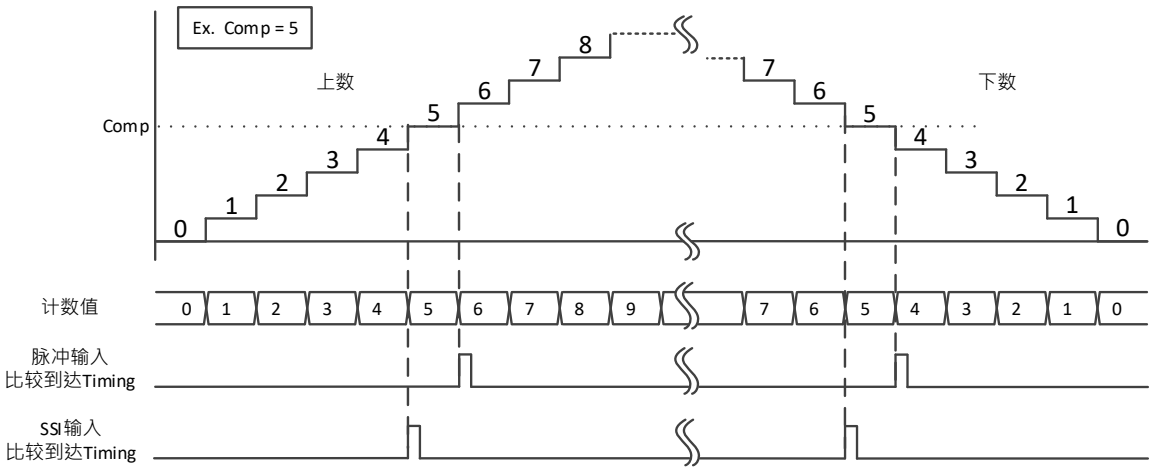
指令说明：

1. DHCCMP 指令为 AS02HC-A 专用指令，当计数值上数/下数到达预先设定之比较值时（称为 MATCH），将依照用户设定对输出点以及计数器执行动作。指令需搭配 PLC 主机韧体 V1.08（含）版以上，以及 ISPSoft 软件 V3.10（含）版以上。
2. DHCCMP 需搭配 DHCCNT 指令使用，只有当 DHCCNT 启动时，计数值才会依输入信号计数，并执行比较动作。
3. 执行指令前先将 **Comp1**、**Comp2**、**Action1**、**Action2**、**Yno1** 及 **Yno2** 参数设定完成，当指令被启动时，将会先对 AS02HC-A 模块设定参数一次。
4. 执行中可以变更 **Comp1**、**Comp2**、**Action1**、**Action2**、**Yno1** 及 **Yno2** 参数，变更方式为先设定为新数值，再设定 **Update** 标志为 On；当此指令完成变更比较值之后，指令将会自行清除 **Update** 标志为 Off，以及 **Match1** 与 **Match2** 标志为 Off。
5. **Module** 指定模块编号，连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
6. **ChNo** 指定执行比较输出之通道编号，通道 1 编号为 1，通道 2 编号为 2。
7. **Comp1** 及 **Comp2** 分别为编号 1 与 2 的比较值（Comp1 及 Comp2 须为不同数值），当该 **ChNo** 通道计数值到达该编号的比较值时，其对应之 **Match1** 及 **Match2** 标志将被设定为 On，以及所指定 **Yno1** 及 **Yno2** 输出点将会依照 **Action1** 及 **Acton2** 指定动作执行。

   当计数器形式选择「线性计数器」时，**Comp1** 及 **Comp2** 需满足「最小计数值 < **Comp** < 最大计数值」；

   当使用 SSI 编码器且计数形式设定为「绝对位置」时，**Comp1** 及 **Comp2** 需满足「$0 \leq$ **Comp** $< 2^{(\text{MT+ST 长度})}$」。

8. 注意比较到达时机点会根据使用之输入接口「脉冲输入」或「SSI 输入」有所不同，说明如下图：



以 Comp=5 为例，SSI 输入比较到达发生时机在计数值由 4→5 瞬间，以及 6→5 瞬间。而对脉冲输入而言，向上计数时，比较到达发生时机在计数值由 5→6 瞬间；向下计数时，比较到达发生时机在计数值由 5→4 瞬间。

9. **Action1**、**Action2** 为比较到达时之指定执行动作，功能说明如下：

Action1	功能说明	**Action2**	功能说明	批注
0	不动作	0	不动作	
1	**Yno1** 指定输出点 Off	1	**Yno2** 指定输出点 Off	
2	**Yno1** 指定输出点 On	2	**Yno2** 指定输出点 On	
3	**Yno1** 指定输出点 Toggle 输出 （ON/OFF 交替）	3	**Yno2** 指定输出点 Toggle 输出（ON/OFF 交替）	
4	**Yno1** 指定输出点 Off + 清除该通道计数值	4	**Yno2** 指定输出点 Off + 清除该通道计数值	因清除计数值时，将连同对 **Match1**、**Match2** 标志清除，故选择这 5 种模式时，不建议判断这两个 Match 标志
5	**Yno1** 指定输出点 On + 清除该通道计数值	5	**Yno2** 指定输出点 On + 清除该通道计数值	
6	**Yno1** 指定输出点 Toggle （ON/OFF 交替） + 清除该通道计数值	6	**Yno2** 指定输出点 Toggle （ON/OFF 交替） + 清除该通道计数值	
7	清除该通道计数值	7	清除该通道计数值	
8	清除该通道计数值 + **Yno1/Yno2** 指定输出点 Off	8	清除该通道计数值 + **Yno1/Yno2** 指定输出点 Off	

注 1：比较到达发生到执行 Action 将存在少许时间延迟，最大延迟时间为 100us。

注 2：当 **Action1/Action2=4/5/6/7/8 时**，

- 因清除计数值时，将连同对 **Match1**、**Match2** 标志清除，故选择这 5 种模式时，不建议判断这两个 Match 标志。

- 当使用 SSI 编码器，且计数器形式设定为「绝对位置」时，将无法清除计数值，但会清除 **Match1**、**Match2** 标志。

10. **Yno1** 及 **Yno2** 分别为比较到达编号 1 与编号 2 分别指定 Y 输出点编号，当比较指令启动之后，HCDO 指令将无法控制输出点：

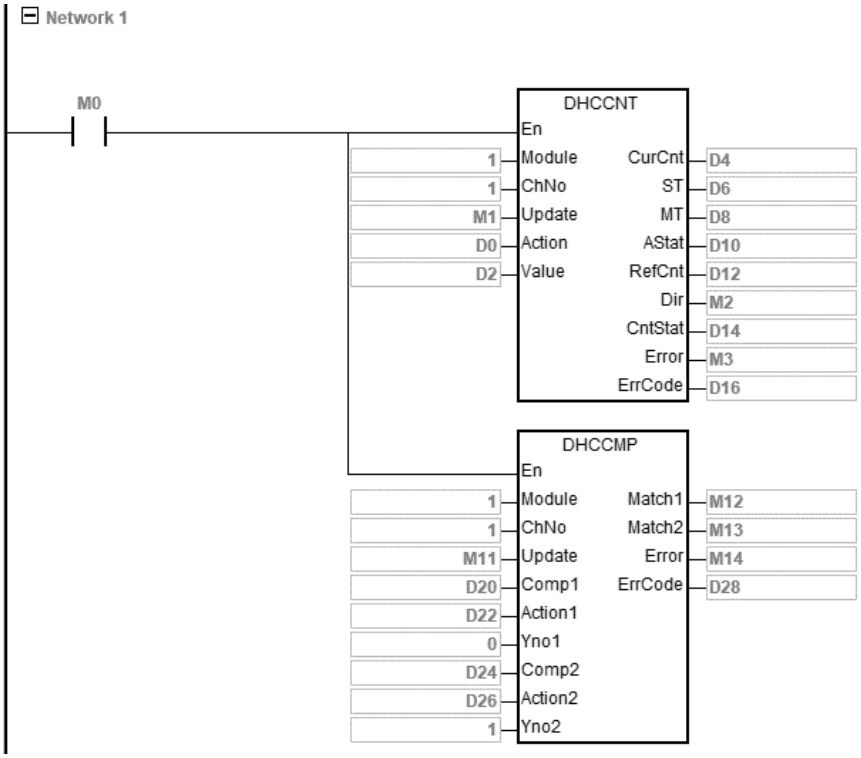
Yno1/Yno2	指定输出点
0	Y0.0
1	Y0.1
2	Y0.2
3	Y0.3

11. **Match1**、**Match2** 分别为比较到达编号 1、编号 2 比较值之状态标志显示，当指定通道的计数值与编号 1 比较值或编号 2 比较值比较到达时，其对应 **Match1** 或 **Match2** 标志将被设定为 On；当主机 STOP、清除计数器、DHCCMP 由 OFF➔ON 或 **Update** 标志被设定时，指令将会对此 **Match1** 与 **Match2** 标志一起清除为 Off。
12. **Match1**、**Match2** 也可指定中断 I 装置，其范围 I400~I431，注意相同编号的 I 装置，发出频率不可高于 800Hz。
13. 若指令关闭时，其相关比较值比对与输出功能都将停止更新。
14. 当 **Error** 错误标志为 On 时，请参考 **ErrCode** 错误代码如下表说明：

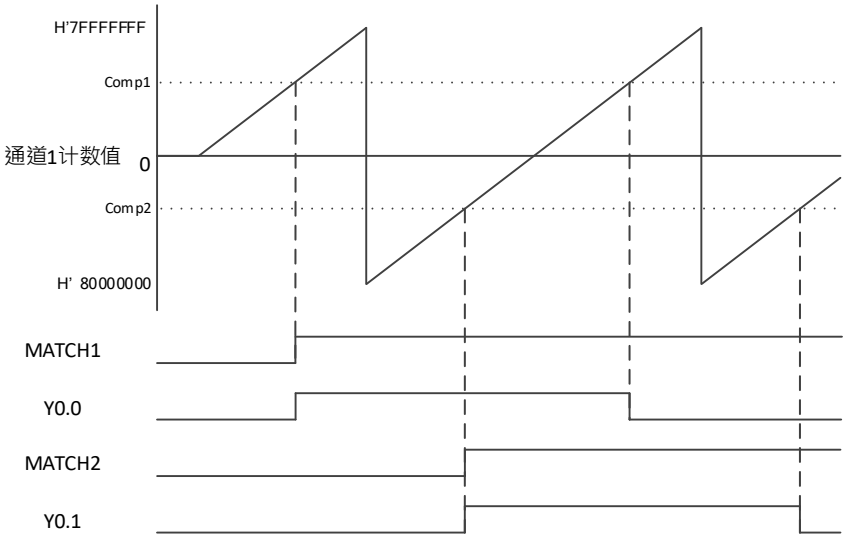
ErrCode 代码	错误状况说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	HWCONFIG 配置错误，指令禁止运作
16#1404	HC 模块无此计数器通道
16#1405	输入接口未选择，指令禁止运作
16#1406	Action 数值不合法
16#1409	Yno 数值不合法
16#1413	HC 模块指定之计数器通道正在执行比较输出功能，不允许重复指定比较输出

程序范例 1 – 比较到达时控制 Y 点输出

1. 设定 HWCONFIG 中通道 1 的输入接口为「脉冲输入」或「SSI 输入」，计数器形式为「环形计数器」。
2. 将 **Comp1** 设定为 16#5000000，**Action1** 设定为 3，**Yno1** 设定为 0；将 **Comp2** 设定为 16#D000000，**Action2** 设定为 3，**Yno2** 设定为 1。
3. 设定 M0=ON 时，计数器开始计数，同时也对模块设定 DHCCMP 参数，开始等待计数器比较到达。
4. 当计数值计数到达 **Comp1** 时，**Match1** 标志被设定为 ON，Y0.0 由 OFF➔ON。
5. 持续计数到达 **Comp2** 时，**Match 2** 标志被设定为 ON，Y0.1 由 OFF➔ON。
6. 当持续计数又再一次到达 **Comp1** 时，Y0.0 将由 ON➔OFF；当计数值又一次到达 **Comp2** 时，Y0.1 将由 ON➔OFF。



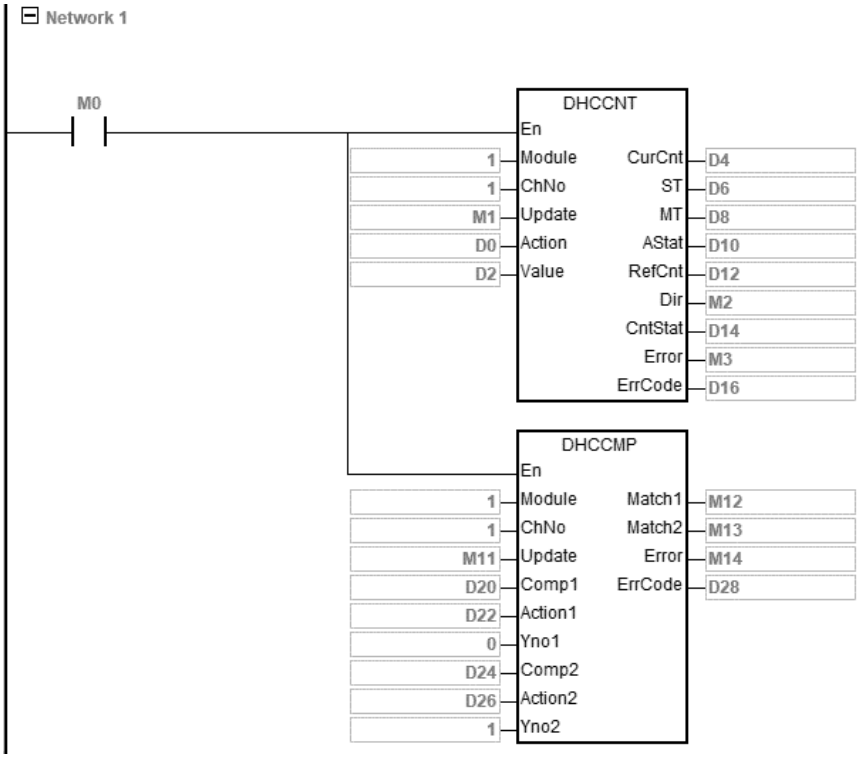
动作时序图：



程序范例 2 – 比较到达时清除计数值

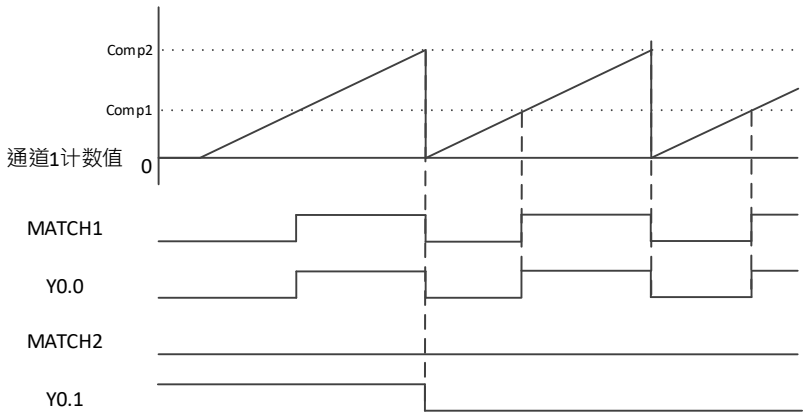
1. 将 **Comp1** 设定为 2500，**Action1** 设定为 2，**Yno1** 设定为 0；将 **Comp2** 设定为 5000，**Action2** 设定为 8，**Yno2** 设定为 1。
2. 设定 M0=ON 时，计数器开始计数，同时也对模块设定 DHCCMP 参数，开始等待计数器比较到达。
3. 当计数值计数到达 **Comp1** 时，**MATCH1** 标志被设定为 ON，Y0.0 被设定为 ON。

4. 持续计数到达 **Comp2** 时，由于 **Action2**=8 的动作为「清除该通道计数值 + 清除 **Yno1**、**Yno2** 指定输出点」，因此计数值被清除为 0、**MATCH1** 及 **MATCH2** 被清除为 0，Y0.0 与 Y0.1 也同时被清除为 OFF。
5. 当持续计数又再一次到达 Comp1 时，**MATCH1** 标志再次被设定为 ON，Y0.0 也被设定为 ON；依此类推。



6

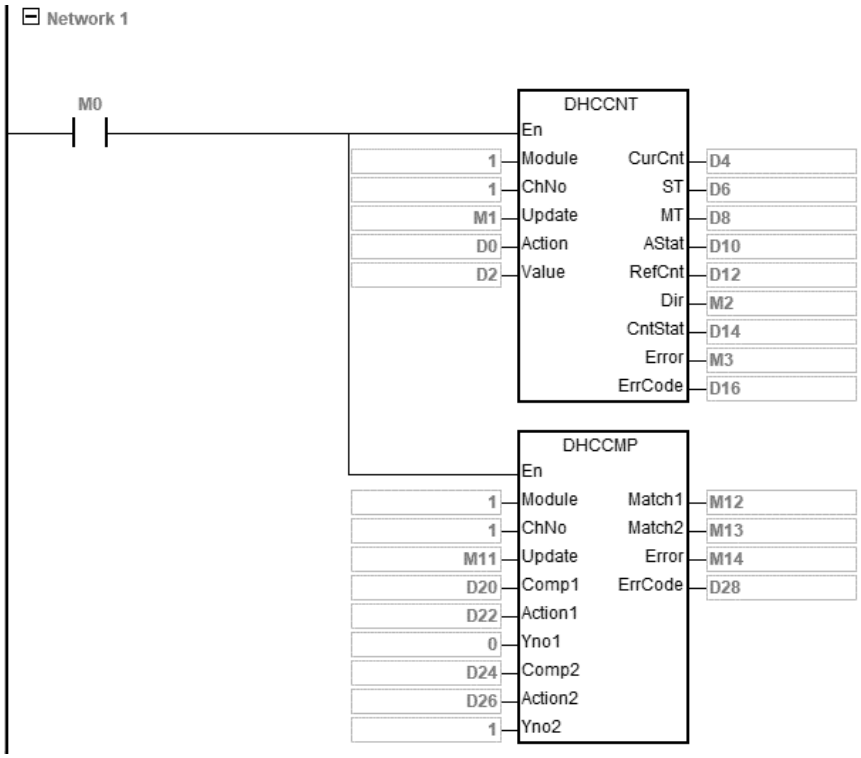
动作时序图：



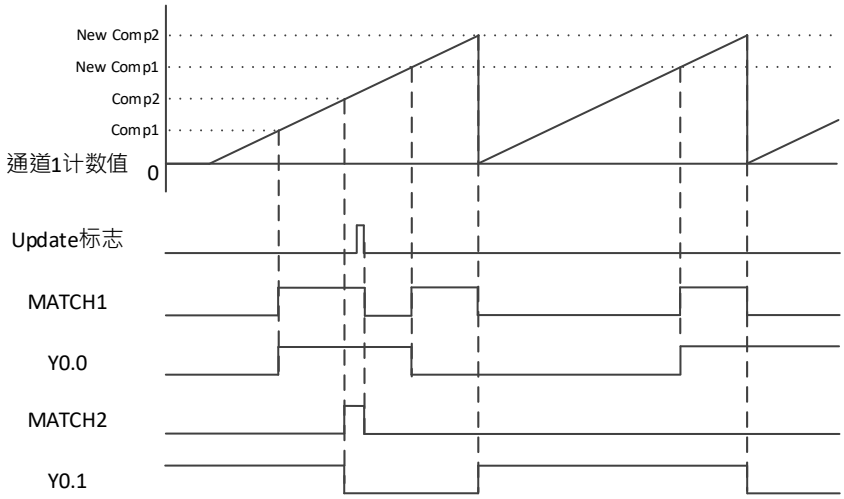
程序范例 3 – 更新比较值

1. 将 **Comp1** 设定为 5000，**Action1** 设定为 3，**Yno1** 设定为 0；将 **Comp2** 设定为 10000，**Action2** 设定为 3，**Yno2** 设定为 1。
2. 设定 M0=ON 时，计数器开始计数，同时也对模块设定 DHCCMP 参数，开始等待计数器比较到达。
3. 当计数值计数到达 **Comp1** 时，**MATCH1** 标志被设定为 ON，Y0.0 由 OFF➔ON。
4. 当计数值计数到达 **Comp2** 时，**MATCH2** 标志被设定为 ON，Y0.1 由 ON➔OFF。

5. 接着设定新比较值：将 **Comp1** 设定为 15000，**Action1** 设定为 3，**Yno1** 设定为 0；将 **Comp2** 设定为 20000，**Action2** 设定为 6，**Yno2** 设定为 1。
6. 设定 **Update** 标志为 ON，设定完成时 **Update** 标志自动清除为 OFF，同时 **MATCH1** 及 **MATCH2** 标志也被清除为 OFF。
7. 当计数值计数到达新比较值 **Comp1** 时，**MATCH1** 标志被设定为 ON，Y0.0 由 ON➔OFF。
8. 当计数值计数到达新比较值 **Comp2** 时，Y0.1 由 ON➔OFF，由于 **Action2**=6 也包含了 Reset，因此计数值清除为 0，同时 **MATCH1** 及 **MATCH2** 标志也被清除为 OFF。

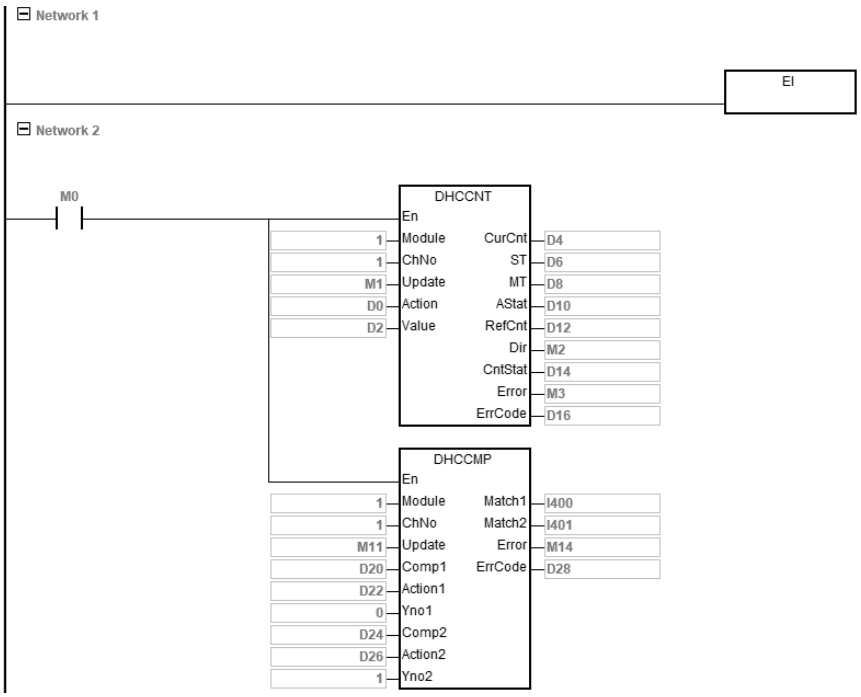


动作时序图：



程序范例 4 – 比较到达中断

1. DHCCMP 也可以设置比较到达中断，在 **MATCH1** 及 **MATCH2** 分别设定为 I400、I401，并配置好 I400、I401 中断程序。
2. 将 **Comp1** 设定为 5000，**Action1** 设定为 3，**Yno1** 设定为 0；将 **Comp2** 设定为 10000，**Action2** 设定为 3，**Yno2** 设定为 1。
3. 设定 M0=ON 时，计数器开始计数，同时也对模块设定 DHCCMP 参数，开始等待计数器比较到达。
4. 当计数值计数到达 **Comp1** 时，I400 对应的中断程序将被执行，Y0.0 被 Toggle（ON/OFF 交替）。
5. 当计数值计数到达 **Comp2** 时，I401 对应的中断程序将被执行，Y0.1 被 Toggle（ON/OFF 交替）。



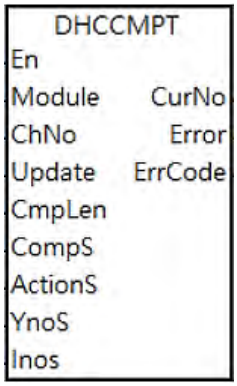
API	指令码			操作数	功能
1422	D	HCCMPT		**Module ~ ErrCode**	HC 模块表格比较输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
ChNo								●					○	○		
Update		●	●	●				●								
CmpLen								●					○	○		
CompS							●	●								
ActionS								●								
YnoS								●								
InoS								●								
CurNo								●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
Update	●												
CmpLen		●				●							
CompS			●				●						
ActionS		●				●							
YnoS		●				●							
InoS		●				●							
CurNo		●				●							
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

ChNo ： 通道编号

Update ： 更新参数标志

CmpLen ： 比较表格组数长度

CompS ： 比较表之 32 位比较值存放来源

ActionS ： 比较表之比较到达之动作存放来源

YnoS ： 比较表之比较到达之 Y 输出点编号存放来源

InoS ： 比较表之比较到达之中断编号存放来源

CurNo ： 当前比较器已执行的组数编号

Error ： 错误标志

ErrCode ： 错误代码

指令说明：

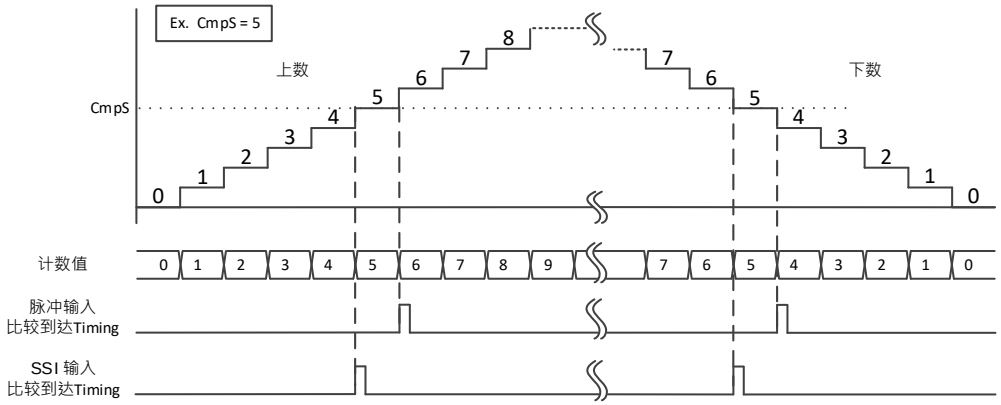
1. DHCCMPT 指令为 AS02HC-A 专用指令，其比较动作将依照设定组数进行「循环式」的比较功能：当比较笔数一步一步比较到达最后一笔比较值时，下一次比较值将重新设置为第一笔比较值；DHCCMPT 指令限定计数器单一方向计数时使用（改变方向将导致动作不正确），且可设定最多 10 个比较值；比较值需依递增或递减方式排序，升序时，计数器需上数使用，且比较值限填正数；递减排序时，计数器需下数使用，且比较值限填负整数。当计数器计数到达预先设定之比较值时，将依照用户设定对输出点及计数器执行动作，以及对主机请求中断。
2. 指令需搭配 PLC 主机韧体 V1.08（含）版以上，以及 ISPSoft 软件 V3.10（含）版以上。
3. DHCCMPT 需搭配 DHCCNT 指令使用，只有当 DHCCNT 启动时，计数值才会依输入信号计数，并执行比较动作。
4. 执行指令前先将 **CmpLen**、**CompS**、**ActionS**、**YnoS** 及 **InoS** 参数设定完成，**En** 初次启动时将会先写入参数一次。
5. 执行中可以变更 **CmpLen**、**CompS**、**ActionS**、**YnoS** 及 **InoS** 参数，变更方式为先设定为新数值，再设定 **Update** 标志为 On；当此指令完成变更之后，指令将会自行清除 **Update** 标志为 Off。
6. **Module** 指定模块编号，连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
7. **ChNo** 指定执行比较输出之通道编号，通道 1 编号为 1，通道 2 编号为 2。
8. **CmpLen** 为比较表格的组数，其比较动作将依照设定组数进行循环式的比较功能。此组数长度可设定数值范围为 2~10 之间，当设定数值超过范围时，指令将不会执行，并显示错误代码。

9. **CompS** 为比较值存放来源，其数值为 32 位整数值，会连续占用 **CmpLen** 个 Dword 空间，比较值需依递增或递减方式排序 (各比较值皆须为不同数值)，升序时，计数器需上数使用，且比较值限填正数；递减排序时，计数器需下数使用，且比较值限填负整数。建议可采用数组 Dword 类型宣告。

   当计数器形式选择「线性计数器」时，**CompS** 需满足「最小计数值 ≤ **CompS** ≤ 最大计数值」;

   当使用 SSI 编码器且计数形式设定为「绝对位置」时，**CompS** 需满足「0 ≤ **CompS** ≤ $2^{(MT+ST\ 长度)}$」。

10. 注意比较到达时机点会根据使用之输入接口「脉冲输入」或「SSI 输入」有所不同，说明如下图：



   以 CompS=5 为例，SSI 输入比较到达发生时机在计数值由 4→5 瞬间，以及 6→5 瞬间。而对脉冲输入而言，向上计数时，比较到达发生时机在计数值由 5→6 瞬间；向下计数时，比较到达发生时机在计数值由 5→4 瞬间。

11. **ActionS** 为比较到达时指定的动作代码，其数值为 16 位整数值，会连续占用 **CmpLen** 个 Word 空间：

Action	功能说明	批注
0	不动作	
1	**YnoS** 指定输出点 Off	
2	**YnoS** 指定输出点 On	
3	**YnoS** 指定输出点 Toggle 输出 (ON/OFF 交替)	
4	**YnoS** 指定输出点 Off + 清除该通道计数值	除了清除该通道计数值之外，也会清除 **CurNo** 为零 注意：当使用 SSI 编码器，且计数器形式设定为「绝对位置」时，将无法清除计数值，但会清除 **CurNo** 为零。
5	**YnoS** 指定输出点 On + 清除该通道计数值	
6	**YnoS** 指定输出点 Toggle (ON/OFF 交替) + 清除该通道计数值	
7	清除该通道计数值	
8	清除该通道计数值 + 所有 **YnoS** 指定输出点 Off	同 **ActionS**=7 的清除 ,并且再增加清除所有被指定的 **YnoS** 输出点。

注：比较到达发生到执行 Action 将存在少许时间延迟，最大延迟时间为 100us。

12. **YnoS** 为指定输出点的编号，其数值为 16 位整数值，会连续占用 **CmpLen** 个 Word 空间。

YnoS	输出点
0	Y0.0
1	Y0.1
2	Y0.2
3	Y0.3

13. **InoS** 为指定中断编号，比较到达时对主机送出指定的中断请求，可用的中断范围为 I400~I431。其数值为 16 位整数值，会连续占用 **CmpLen** 个 Word 空间。

InoS	动作
0	不设定中断
4□□	□□ = 00~31，对应中断编号 I400~I431 比较到达时对主机请求指定的中断

注：相同编号的 I 装置，发出频率不可高于 800Hz。

14. **CurNo** 为显示目前已经比较到达的组数编号，举例：当计数值为 200，则对应下列比较表内比第一组比较值低，因此 **CurNo** 数值为 0；当计数值到达 1000，则对应比较表等于第一组比较值，因此 **CurNo** 数值变为 1。

15. **CompS**（假设指定 D100），**ActionS**（假设指定 D200），**YnoS**（假设指定 D300），**CurNo** 操作数说明请参考下表，此表假设 **CmpLen** 数值为 6。

CurNo 组数编号	**CompS** 来源比较值（32 位数值）	**ActionS** 比较到达指定动作	**YnoS** 输出点编号	**InoS** 中断编号
1	D100 = 1000	D200 = 2（On）	D300 = 0（Y0.0）	D400 = 400（I400）
2	D102 = 2000	D201 = 2（On）	D301 = 1（Y0.1）	D401 = 401（I401）
3	D104 = 3000	D202 = 3（Toggle）	D302 = 0（Y0.0）	D402 = 402（I402）
4	D106 = 4500	D203 = 2（On）	D303 = 2（Y0.2）	D403 = 403（I403）
5	D108 = 5500	D204 = 1（Off）	D304 = 3（Y0.3）	D404 = 404（I404）
6	D110 = 6500	D205 = 1（Off）	D305 = 1（Y0.1）	D405 = 405（I405）

16. 当指令启动时（**En** 由 Off 变为 On）、以及设定 **Update** 标志为 On 执行变更参数时，此指令会根据当前计数值，对用户设定的比较表内所有的比较值先做一次大小比对，小于当前计数值之组数将执行比较到达时的动作。例如：当指令启动时计数值为 3500，则比较表内编号 1~3（比较值<当前计数值）的动作都将被执行过一次比较到达的动作（Y0.0=On+I400 中断请求、Y0.1=On +I401 中断请求、Y0.0 Toggle（ON/OFF 交替）+I402 中断请求），**CurNo** 也将被设定为 3。

17. 若 **En** 由 On 变为 Off 时表示关闭表格比较输出功能，**CurNo** 状态显示将停止更新，输出点状态也不受指令关闭而变动，计数器依然可以持续计数，但不再进行比较。

18. 当 **Error** 错误标志为 On 时，请参考 **ErrCode** 错误代码如下表说明：

ErrCode 代码	错误状况说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	HWCONFIG 配置错误，指令禁止运作
16#1404	HC 模块无此计数器通道
16#1405	输入接口未选择，指令禁止运作
16#1406	Action 数值不合法
16#1408	**CompS** 大小排序错误
16#1409	Yno 数值不合法
16#140C	CmpLen 数值不合法
16#1413	HC 模块指定之计数器通道正在执行比较输出功能，不允许重复指定比较输出
16#1416	Ino 数值不合法

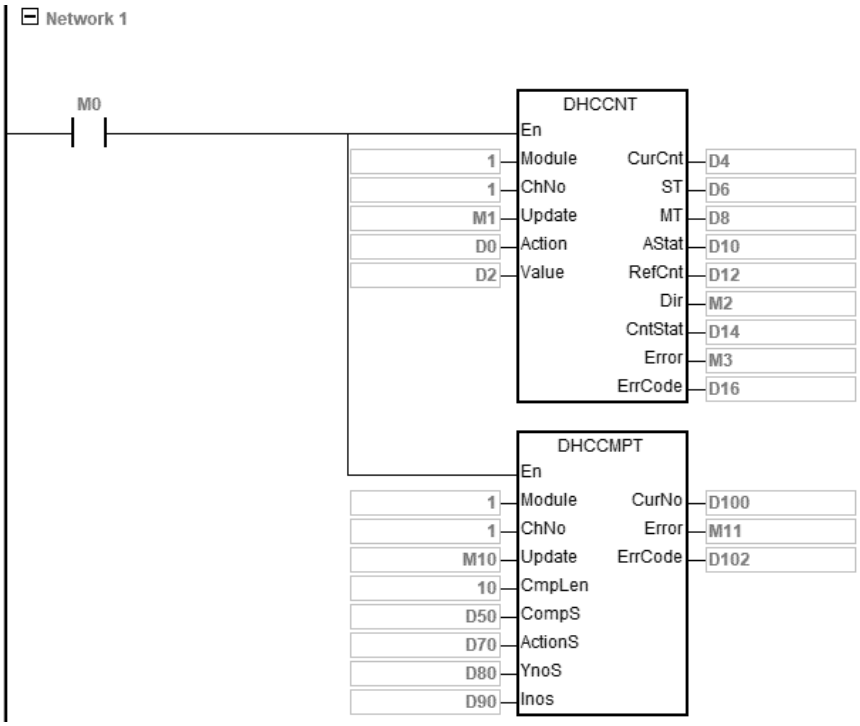
程序范例：

1. 将比较值 1000、2000、3000…依序输入 D 装置中；将 Action 代码 3、3、3…依序输入 D 装置中；将 Yno 代码 0、1、2、3、0、1…依序输入 D 装置中；将 Ino 代码 400、401、402…依序输入 D 装置中。

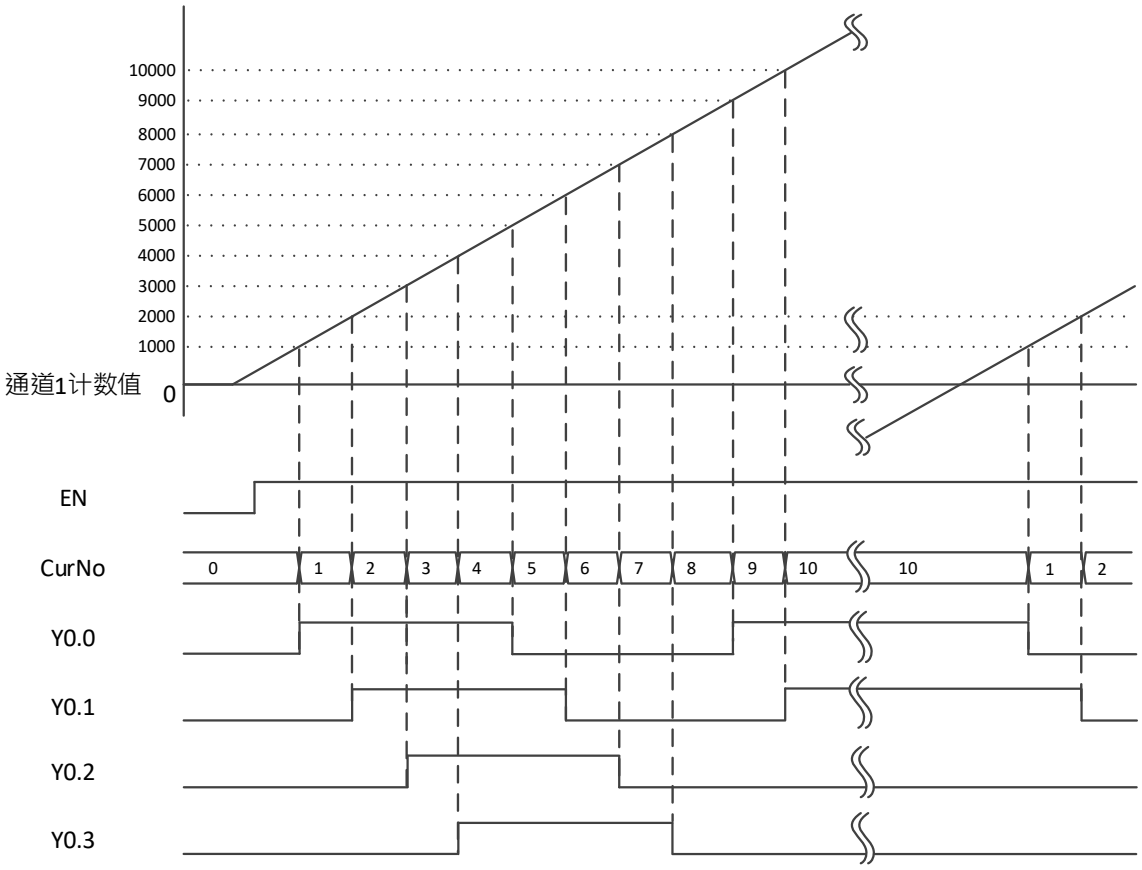
CurNo 组数编号	**CompS** 来源比较值（32 位数值）	**ActionS** 比较到达指定动作	**YnoS** 输出点编号	**InoS** 中断编号
1	D50 = 1000	D70 = 3（Toggle）	D80 = 0（Y0.0）	D90 = 400
2	D52 = 2000	D71 = 3（Toggle）	D81 = 1（Y0.1）	D91 = 401
3	D54 = 3000	D72 = 3（Toggle）	D82 = 2（Y0.2）	D92 = 402
4	D56 = 4000	D73 = 3（Toggle）	D83 = 3（Y0.3）	D93 = 403
5	D58 = 5000	D74 = 3（Toggle）	D84 = 0（Y0.0）	D94 = 404
6	D60 = 6000	D75 = 3（Toggle）	D85 = 1（Y0.1）	D95 = 405
7	D62 = 7000	D76 = 3（Toggle）	D86 = 2（Y0.2）	D96 = 406
8	D64 = 8000	D77 = 3（Toggle）	D87 = 3（Y0.3）	D97 = 407
9	D66 = 9000	D78 = 3（Toggle）	D88 = 0（Y0.0）	D98 = 408
10	D68 = 10000	D79 = 3（Toggle）	D89 = 1（Y0.1）	D99 = 409

2. 当 **En** 由 OFF→ON 时，先对用户设定的比较表内所有的比较值先做一次大小比对，小于当前计数值之组数将执行比较到达时的动作。由于启动时计数值尚小于第一笔比较值 1000，因此不会执行任何一笔比较到达指定动作，CurNo 目前为 0。
3. 当计数值到达 1000 时比较到达第一笔比较值（若为脉冲输入，将在计数值由 1000→1001 时才执行比较到达指定动作），Y0.0 由 OFF➔ON、I400 中断程序被执行、**CurNo** 由 0 变为 1。

4. 当计数值到达 2000 时比较到达第二笔比较值，Y0.1 由 OFF➔ON、I401 中断程序被执行、**CurNo**=2。
5. 当计数值到达 3000 时比较到达第三笔比较值，Y0.2 由 OFF➔ON、I402 中断程序被执行、**CurNo**=3。
6. 当计数值到达 4000 时比较到达第四笔比较值，Y0.3 由 OFF➔ON、I403 中断程序被执行、**CurNo**=4。
7. 依此规则持续比较到达第五~十笔比较值，此时 **CurNo**=10。由于已完成最后一笔比较，下一笔比较值将设定为第一笔比较值 1000。
8. 当环形计数器再次计数到 1000 时，比较到达第一笔计数值，Y0.0 由 ON➔OFF、I400 中断程序被执行，**CurNo**=1。依此规则执行后续比较。



动作时序图：


10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
通道1计数值
0
EN
CurNo
0
1
2
3
4
5
6
7
8
9
10
10
1
2
Y0.0
Y0.1
Y0.2
Y0.3

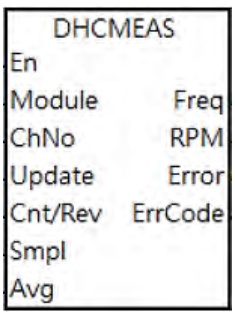
API	指令码			操作数	功能
1423	D	HCMEAS		**Module ~ ErrCode**	HC 模块频率及转速测量

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Module								●					○	○		
ChNo								●					○	○		
Update		●	●	●				●								
Cnt/Rev								●					○	○		
Smpl								●					○	○		
Avg								●					○	○		
Freq								●								
RPM								●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
Update	●												
Cnt/Rev			●				●						
Smpl		●				●							
Avg		●				●							
Freq			●				●						
RPM			●				●						
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Module ： 模块编号

ChNo ： 通道编号

Update ： 更新参数标志

Cnt/Rev ： 一圈计数量

Smpl ： 取样周期

Avg ： 平均次数

Freq ： 频率测量结果

RPM ： 转速测量结果（r/min）

Error ： 错误标志

ErrCode ： 错误代码

指令说明：

1. DHCMEAS 指令为 AS02HC-A 专用指令，功能为频率及转速测量。指令需搭配 PLC 主机韧体 V1.08（含）版以上，以及 ISPSoft 软件 V3.10（含）版以上。
2. DHCMEAS 需搭配 DHCCNT 指令使用，只有当 DHCCNT 启动时，计数值才会依输入信号计数，并由计数值之变化计算出测量结果。
3. 执行指令前先将 **Cnt/Rev**、**Smpl** 与 **Avg** 参数设定完成，**En** 被启动时将会先写入参数一次；执行中若要变更参数，变更方式为先设定为新数值，再设定 **Update** 标志为 On；当此指令完成变更之后，指令将会自行清除 **Update** 标志为 Off。
4. **Module** 指定模块编号，连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
5. **ChNo** 指定欲控制之计数器通道编号，通道 1 编号为 1，通道 2 编号为 2。
6. **Cnt/Rev** 为编码器转动一圈之计数值，设定范围为 1~4294967295（H'00000001~H'FFFFFFFF）。
7. **Smpl** 为取样周期，设定范围为 1~1000，单位为 ms；根据设定的 **Smpl**，**Freq** 频率测量结果输出以及 **RPM** 转速测量结果输出将会有不同的分辨率：

   Freq分辨率=1000 / Smpl (单位：Hz)

   RPM分辨率 = 60000 /（Smpl x Cnt / Rev）(单位：rev / min)

   由于转速计算方式是以取样周期开始以及结束时的 CurCnt 计数值做计算，因此在设计取样周期时需自行排除下列情形：

编码器类型	计数器类型	影响测量正确性之因素
增量型编码器	环形计数器	取样周期内位移超出 2^{31}
	线性计数器	取样周期内位移超出 2^{31}，或计数值超出上下限
绝对型 SSI 编码器	显示绝对位置	取样周期内位移超出 $2^{(MT+ST\ 长度)-1}$
	环形计数器	取样周期内位移超出 2^{31}

8. **Avg** 为平均次数，对测量结果做移动平均，设定范围为 1~10。
9. **Freq** 为平均测量频率结果，单位为 Hz，频率之计算方式为：

$$\text{frequency}\ (\text{Hz}) = \frac{\text{CurCnt}(t + \textbf{Smpl}) - \text{CurCnt}(t)}{\textbf{Smpl}(\text{ms}) \times 10^{-3}}$$

10. **RPM** 为平均转速测量结果，单位为 rev/min，转速之计算方式为：

$$\text{转速}\ (\text{rev/min}) = \frac{(\ \text{CurCnt}(t + \textbf{Smpl}) - \text{CurCnt}(t)\ ) \times 60}{\textbf{Cnt/Rev} \times \textbf{Smpl}(\text{ms}) \times 10^{-3}}$$

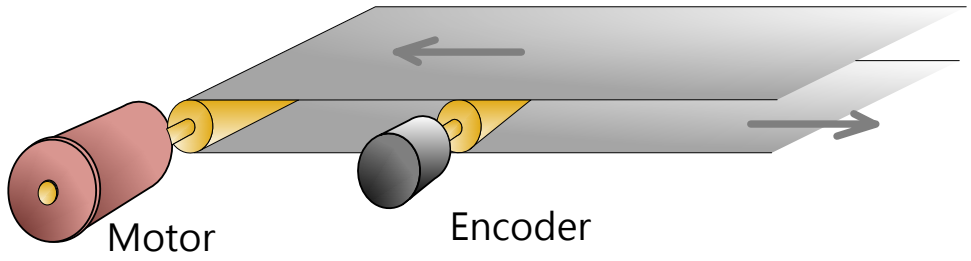
11. 当指令被关闭时，表示关闭测量功能，**Freq**、**RPM** 显示将保持不变且不再更新。
12. 注意当遇到计数值被清除或是用户变更计数值时，将会影响该取样周期的 **Freq** 与 **RPM** 计算结果。
13. 当 **Error** 错误标志为 Off 时，其 **ErrCode** 错误代码为 0；当 **Error** 错误标志为 On 时，请参考 **ErrCode** 错

误代码如下表说明：

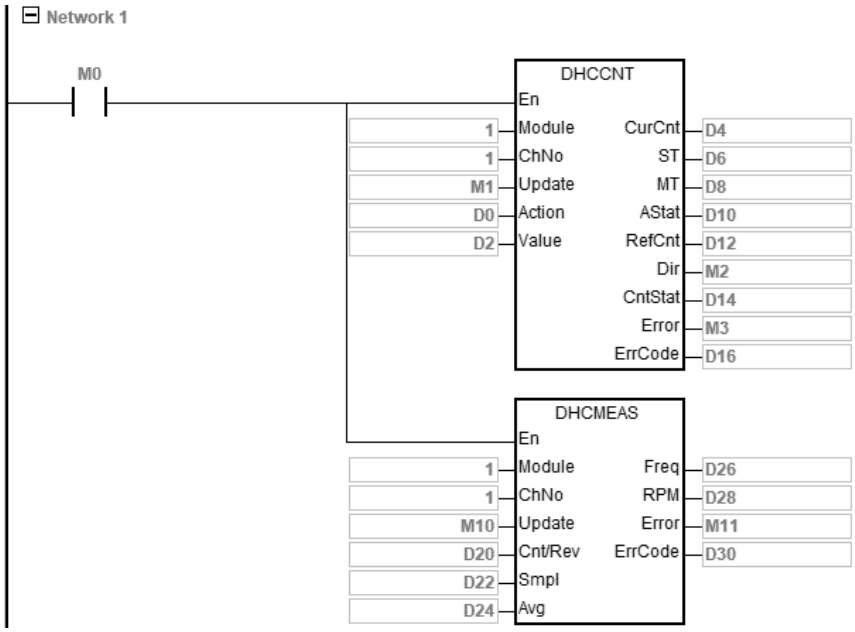
ErrCode 代码	错误状况说明
16#1400	模块不支持此功能
16#1401	模块存入数值不合法或超出范围
16#1402	模块无响应，通讯逾时
16#1403	HWCONFIG 配置错误，指令禁止运作
16#1404	HC 模块无此计数器通道
16#1405	输入接口未选择，指令禁止运作
16#140A	频率测量结果超出范围
16#140B	转速测量结果超出范围
16#140D	**Smpl** 数值不合法
16#140E	**Avg** 数值不合法
16#140F	**Cnt/Rev** 数值不合法
16#1415	HC 模块指定之计数器通道正在执行频率测量功能，不允许重复指定转速测量

程序范例 1：增量型编码器

下图输送带中，假设编码器分辨率为 1000 脉冲/圈，可由下述步骤使用 DHCMEAS 指令监测电机转速：



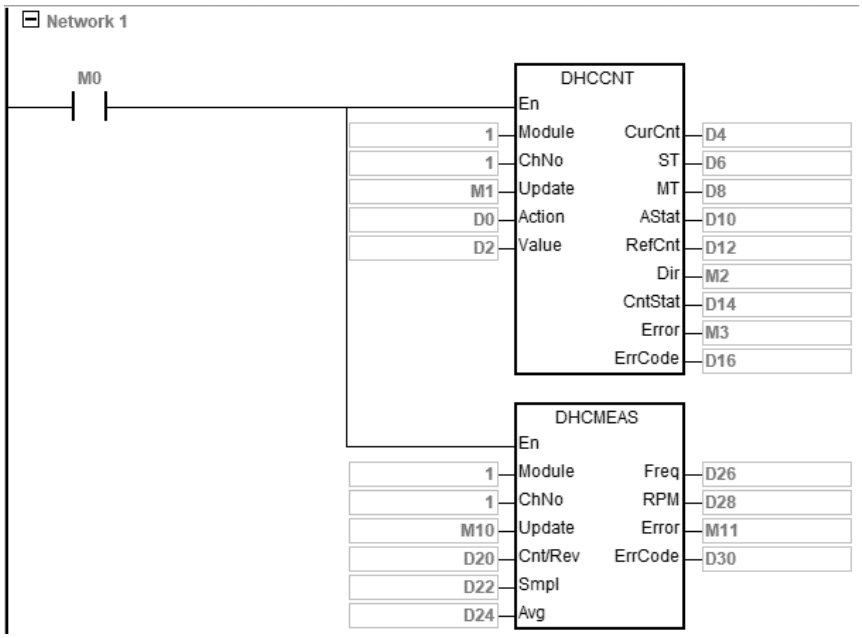
1. 设定 HWCONFIG 中通道 1 的输入接口为「脉冲输入」，计数器形式为「环形计数器」。
2. 将 **Cnt/Rev** 设定为 1000，**Smpl** 设定为 1000，**Avg** 设定为 10。
3. 设定 M0=ON 时，计数器开始计数，同时也对模块设定 DHCMEAS 参数，开始频率及转速量测，每 1000ms 将计算结果显示在 Freq 及 RPM。
4. 当电机运转时，若增量型编码器接收到 5kHz 的脉冲，此时 Freq 计算结果为 5kHz、RPM 计算结果为 300 rev/min。此范例中编码器转速即为电机转速，故得知电机转速亦为 300 rev/min。



程序范例 2：绝对型 SSI 编码器（27-bit）

承范例 1，若输送带中增量型编码器置换为分辨率为 13bit 的绝对型 SSI 编码器，DHCMEAS 的参数设定步骤如下：

1. 设定 HWCONFIG 中通道 1 的输入接口为「SSI 输入」。
2. 将 **Cnt/Rev** 设定为 8192，**Smpl** 设定为 1000，**Avg** 设定为 10。
3. 设定 M0=ON 时，计数器开始计数，同时也对模块设定 DHCMEAS 参数，开始频率及转速量测，每 1000ms 将计算结果更新在 Freq 及 RPM。
4. 当电机运转时，若绝对型 SSI 编码器的位置变化速率为 4095 LSB/秒，此时 Freq 计算结果为 4095Hz、RPM 计算结果为 30 rev/min。此范例中编码器转速即为电机转速，故得知电机转速亦为 30 rev/min。

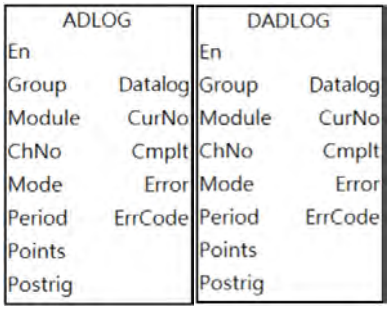


API	指令码			操作数	功能
1424	D	ADLOG		依序如下表列	模拟输入模块数据记录

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Group								●					○	○		
Module								●					○	○		
ChNo								●					○	○		
Mode								●					○	○		
Period								●					○	○		
Points								●					○	○		
PostTrg								●					○	○		
Datalog								●								
CurNo								●								
Cmplt		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●	●			●	●						
Module		●	●			●	●						
ChNo		●	●			●	●						
Mode		●	●			●	●						
Period		●	●			●	●						
Points		●	●			●	●						
PostTrg		●	●			●	●						
Datalog		●	●			●	●		●				
CurNo		●				●							
Cmplt	●												
Error	●												
ErrCode		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
-	AS	AS



Group ： 群组编号

Module ： 模块编号

ChNo ： 通道编号

Mode ： 记录模式

Period ： 记录周期

Points	：	总记录笔数
PostTrg	：	触发后记录笔数
Datalog	：	存放记录值之装置
CurNo	：	累积记录点数
Cmplt	：	记录完成标志
Error	：	错误标志
ErrCode	：	错误代码

指令说明：

1. 此指令为模拟输入模块（AS04AD-A、AS06XA-A、AS02ADH-A）专用指令，功能为启动 / 关闭记录功能，并将记录数据由模块传送至指定 **Datalog** 装置。指令需搭配版本如下所列：

项目	支持版本
PLC 主机韧体	V1.10.00（含）以上
ISPSoft 软件	V3.13（含）以上
AS04AD-A/AS06XA-A 韧体	V1.02.00（含）以上
AS02ADH-A 韧体	V1.00.00（含）以上

2. 各机种对应之记录周期及点数规格如下表说明，详细使用方式请参考 Mode 参数说明。

机种	记录周期	记录笔数
AS04AD-A AS06XA-A	固定周期（Fixed period）模式： 设定范围 1~100，单位时间固定为 10ms	固定为 500 笔
AS02ADH-A	固定周期（Fixed period）模式、触发启动型固定周期（Fixed period + Trigger Start）模式、触发位置指定（Fixed period + Trigger position Assign）记录模式： 设定范围 1~32000，可选择时间单位 20us/40us/80us （注：时间单位为 HWCONFIG 设定之取样周期） 点记录（Point Logging）模式： 由外部输入点每触发一次即记录一点，无固定周期；由 HWCONFIG 设定外部输入点触发时机 通道 1 触发时机为 X0.0 上升沿或下降沿 通道 2 触发时机为 X0.1 上升沿或下降沿	可设定 1~2000 笔

3. **Group** 指定模拟输入模块连接于主机右侧或者远程模块右侧之群组编号，主机编号为 0，第 1 台远程模块编号为 1，以此原则类推，最大群组数为 15。
4. **Module** 指定模块编号，连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
5. **ChNo** 指定通道编号，通道 1 编号为 1，通道 2 编号为 2。

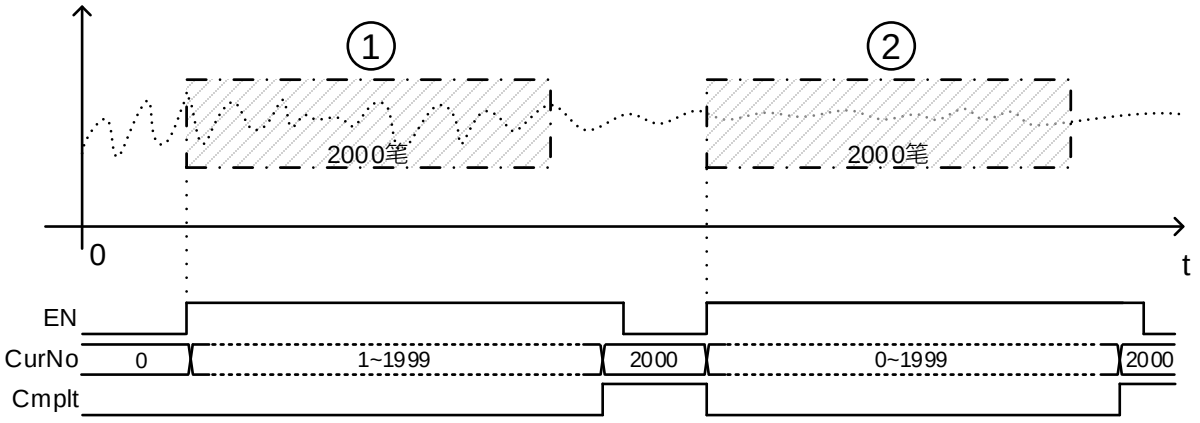
6. **Mode** 为记录模式，各机种支持之模式如下表所示。

适用机种	**Mode** 数值	名称	记录模式说明
AS04AD-A AS06XA-A	K0	固定周期（Fixed period）模式	执行固定周期记录
AS02ADH-A	K0	固定周期（Fixed period）模式	执行固定周期记录
	K1	触发启动型固定周期（Fixed period + Trigger Start）模式	等待外部输入点触发信号，当收到触发后，立即执行固定周期记录
	K2	点记录（Point Logging）模式	无固定周期，由外部输入点每触发一次即记录一笔
	K3	触发位置指定（Fixed period + Trigger position Assign）模式	可指定外部触发时机点，记录前/后固定周期所记录的数据

(1) 固定周期（Fixed period）模式：设定 **Mode**=0，指令 **EN** 开启即以设定之记录周期执行记录，当设定之总记录笔数记录完成，**CmpIt** 标志将自动设定为 High。

范例：

设定 **Points** = 2000

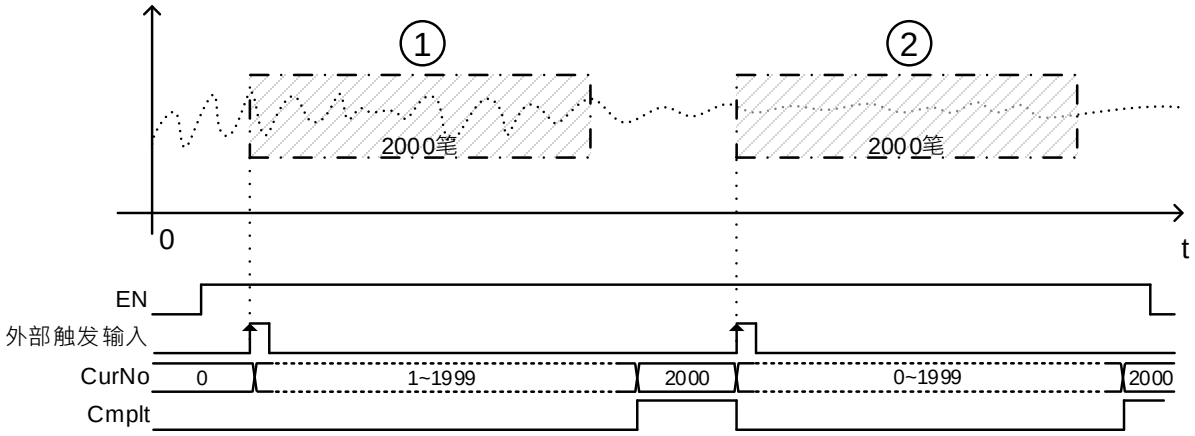


(2) 触发启动型固定周期（Fixed period + Trigger Start）模式：设定 **Mode**=1，开始记录前先将指令 **EN** 开启，外部触发输入点触发时立即以设定之记录周期执行记录，记录完成 **CmpIt** 标志将自动设定为 High。直到设定之总记录笔数记录完成前，外部触发输入点的任何操作将不影响记录的进行；但当记录笔数已完成、**CmpIt** 标志为 High 时，重新触发外部输入点，即可开始新的一轮记录，不须将 **EN** 关闭重启。

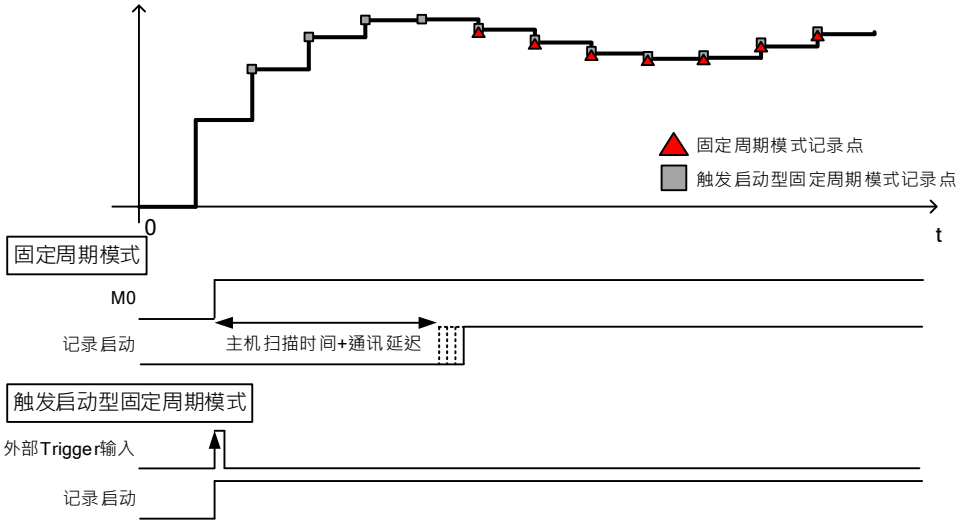
记录通道	对应外部触发输入信号来源 （由 HWCONFIG 设定外部输入点触发时机）
通道 1	X0.0 上升沿或下降沿
通道 2	X0.1 上升沿或下降沿

范例：

设定 **Points** = 2000，外部输入点触发时机设定为上升沿触发



触发启动型固定周期（Fixed period + Trigger Start）模式与固定周期（Fixed period）模式功能类似，但固定周期（Fixed period）模式的记录起始时机将会受主机扫描时间、模块通讯时间的影响造成延迟，如下图所示。固定周期（Fixed period）模式中假设 M0 为控制 ADLOG 指令 **EN** 的装置，可以看出当 M0 由 OFF->ON 时，模块并非立即开始执行记录，而是有一小段延迟。

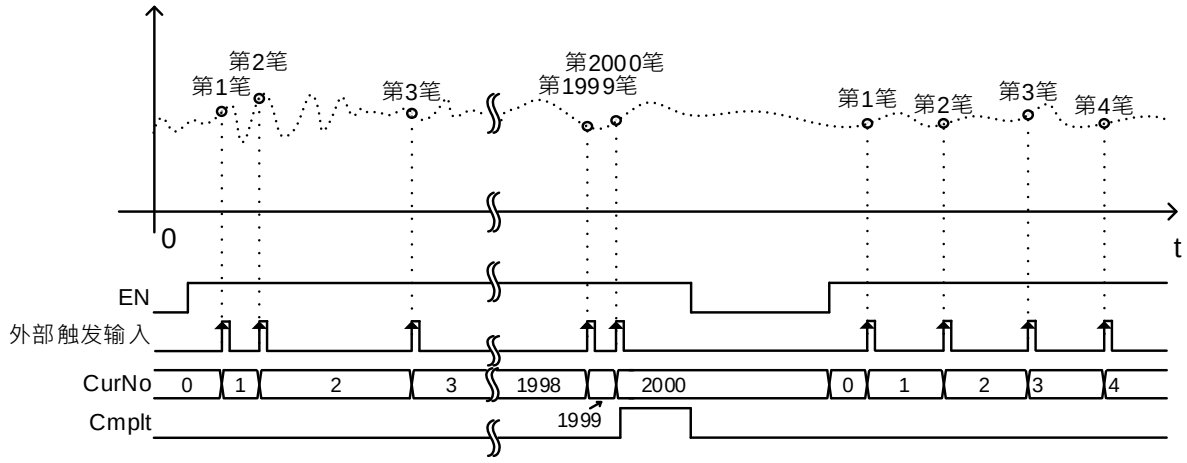


(3) 点记录（Point Logging）模式：设定 **Mode**=2，开始记录前先将指令 **EN** 开启，外部触发输入点每触发一次即记录一笔，达设定之总记录笔数时，**Cmplt** 标志将自动设定为 High；**Cmplt** 标志为 High 后若需继续记录，须将指令重启。

记录通道	对应外部触发输入信号来源 （由 HWCONFIG 设定外部输入点触发时机）
通道 1	X0.0 上升沿或下降沿
通道 2	X0.1 上升沿或下降沿

范例：

设定 **Points** = 2000，外部输入点触发时机设定为上升沿触发

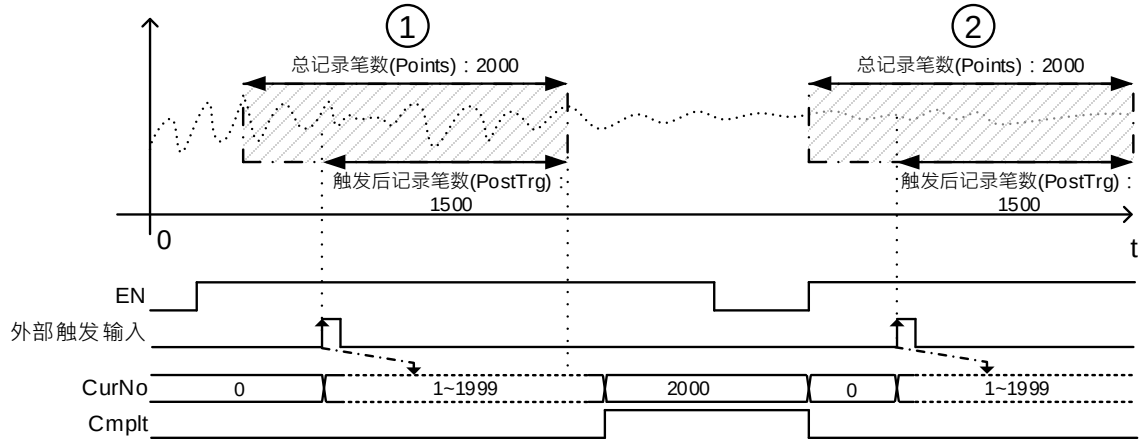


(4) 触发位置指定（Fixed period + Trigger position Assign）记录模式：设定 **Mode**=3，并依需求设定 **Points**、**PostTrg** 参数。此模式藉由外部输入点触发，并且依据设定记录触发前/后特定的笔数。当使用指令 **EN** 开启此记录模式之后，AS02ADH-A 将会开始等待外部触发输入信号，等信号触发之后立即开始采样，直到记录笔数达到设定的笔数后，**Cmplt** 标志将自动设定为 High。**CurNo** 数值在触发前为 0，在触发后模块即开始将触发前的记录数据陆续回传给主机，因此 **CurNo** 数值将逐渐追上累积的记录笔数。

记录通道	对应外部触发输入信号来源 （由 HWCONFIG 设定外部输入点触发时机）
通道 1	X0.0 上升沿或下降沿
通道 2	X0.1 上升沿或下降沿

范例：

设定 **Mode**=3，**Points** = 2000，**PostTrg** = 1500，则表示第 501 点（**Points** - **PostTrg**）的位置，就会是当下外部触发所记录的第 1 笔数据。



7. **Period** 为记录周期设定，说明如下：

机种	Period 设定范围	Period 单位
AS04AD-A AS06XA-A	1~100	固定为 10ms，不可变更
AS02ADH-A	1~32000	须由 HWCONFIG 设定取样周期时间，可选择 20us，40us 及 80us。 若使用「**Mode**=2：点记录（Point Logging）模式」，此 Period 设定值无效。

8. **Points** 为总记录笔数，此参数只对 AS02ADH-A 有效，可设定最多 2000 笔；其他机种不论此参数设定为何，皆固定为 500 笔。

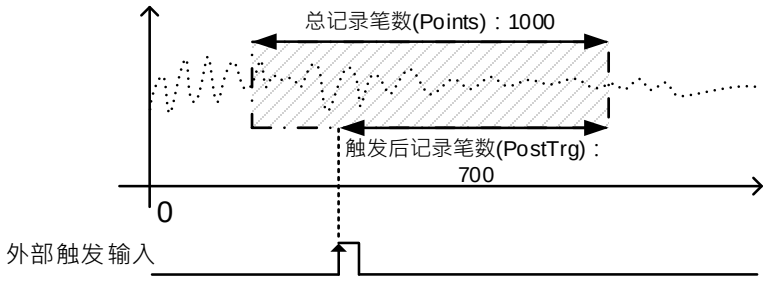
机种	总记录笔数设定范围
AS04AD-A AS06XA-A	设定无效，固定为 500 笔
AS02ADH-A	可设定，1~2000 笔

9. **PostTrg** 为 Trigger 发生后记录之点数，此参数与总记录笔数 **Points** 搭配使用，即可记录触发前后之数据。此参数仅供触发位置指定（Fixed period + Trigger position Assign）记录模式（**Mode**=3）使用，于其它记录模式无效。注意此值不得大于总记录笔数 **Points**，若超出就会自动以 **Points** 数值记录笔数。举例：若设定 **Points** = 100，**PostTrg** = 200，由于 **PostTrg** 大于总记录笔数，将自动把 **PostTrg** 视为 100，因此只会记录触发后的 100 笔数据。

机种	**PostTrg** 设定范围
AS02ADH-A	0 ~2000 笔（不得大于总记录笔数 **Points**）

范例：

设定 **Points** = 1000，**PostTrg** = 700，因此 1000 笔记录中包含了外部触发输入点触发前 300 笔，以及后 700 笔数据。



10. **Datalog** 为指定存放记录值之装置，若 AD 模块输出数值格式选择「整数格数」，建议搭配 16 位指令 ADLOG 使用，此时使用 AS04AD-A、AS06XA-A 机种会连续占用 500 个 word 空间；使用 AS02ADH-A 机种会连续占用 1~2000（依总记录笔数）个 word 空间（若使用 32 位指令 DADLOG，则指令会自动将 16 位数据展成 32 位数据，并多占用一倍的空间）。若 AD 模块输出数值格式选择「浮点数格数」，则须搭配 32 位指令 DADLOG 使用，此时使用 AS04AD-A、AS06XA-A 机种会连续占用 1000 个 word 空间；使用 AS02ADH-A 机种会连续占用 2~4000（依总记录笔数）个 word 空间。

11. **CurNo** 为显示目前模块已回传给主机的记录笔数，当记录正在进行中，**CurNo** 数值由于扫描周期的影响，将有可能无法以连贯的数值显示。
12. **CmpIt** 为记录完成的标志，当 **CmpIt** 由 OFF->ON 时，表示记录值已全数传送到指定之 Datalog 装置，**CmpIt** 标志在 **EN** 由 OFF->ON 时，将自动初始化为 OFF。
13. 若指令关闭时，该指定通道将停止更新指令右半部各项数值。
14. 当 **Error** 错误标志为 ON 时，请参考 **ErrCode** 错误代码如下表说明：

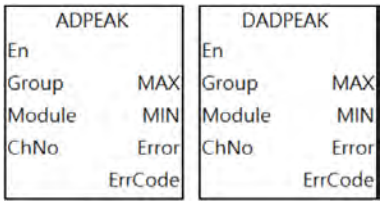
ErrCode 代码	错误状况说明
16#1400	模块不支持此功能
16#1401	群组编号或模块编号设定错误
16#1402	模块存入数值不合法或超出范围
16#1403	模块无响应，通讯逾时
16#1404	AD 模块无此通道
16#1405	AD 模块指定之通道正在执行记录，不允许重复指定
16#1406	AD 模块数据格式与指令类型不匹配
16#1407	记录模式数值超出范围
16#1408	记录周期数值超出范围
16#1409	总记录笔数 **Points** 超出范围

API	指令码			操作数	功能
1425	D	ADPEAK		依序如下表列	模拟输入模块峰值记录

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Group								●					○	○		
Module								●					○	○		
ChNo								●					○	○		
MAX								●								
MIN								●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●	●			●	●						
Module		●	●			●	●						
ChNo		●	●			●	●						
MAX		●	●			●	●		●				
MIN		●	●			●	●		●				
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	AS



Group : 群组编号

Module : 模块编号

ChNo : 通道编号

MAX : 最大值

MIN : 最小值

Error : 错误标志

ErrCode : 错误代码

指令说明：

1. ADPEAK 指令此指令为模拟输入模块（AS04AD-A、AS08AD-B、AS08AD-C、AS06XA-A、AS02ADH-A）专用指令，其功能为启动/关闭模块进行峰值记录功能。指令搭配版本如下所列：

项目	支持版本
PLC 主机韧体	V1.10.00（含）以上
ISPSoft 软件	V3.13（含）以上
AS04AD-A/AS06XA-A 韧体	V1.02.00（含）以上

项目	支持版本
AS02ADH-A 韧体	V1.00.00（含）以上

2. **Group** 指定模拟输入模块连接于主机右侧或者远程模块右侧之群组编号，主机编号为 0，第 1 台远程模块编号为 1，以此原则类推，最大群组数为 15。
3. **Module** 指定模块编号，连接于主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。
4. **ChNo** 指定欲控制之计数器通道编号，通道 1 编号为 1，通道 2 编号为 2。
5. **MAX** 与 **MIN** 分别为最大值与最小值，当 EN 由 OFF->ON 时，**MAX** 与 **MIN** 将被初始化为最新一笔量测值，接着即开始峰值记录功能，该通道将持续检测最大值与最小值并更新到 **MAX** 与 **MIN**。若 AD 模块输出数值格式选择「整数格数」，建议搭配 16 位指令 ADPEAK 使用；若 AD 模块输出数值格式选择「浮点数格数」，则须搭配 32 位指令 DADPEAK 使用。
6. 若 **En** 由 ON 变为 OFF 时表示关闭峰值记录功能，此时 **MAX**、**MIN** 内容值将保持不变且不再更新。
7. 当 **Error** 错误标志为 ON 时，请参考 **ErrCode** 错误代码如下表说明：

ErrCode 代码	错误状况说明
16#1400	模块不支持此功能
16#1401	群组编号或模块编号设定错误
16#1402	模块存入数值不合法或超出范围
16#1403	模块无响应，通讯逾时
16#1404	AD 模块无此通道
16#1405	AD 模块指定之通道正在执行峰值记录，不允许重复指定
16#1406	AD 模块数据格式与指令类型不匹配

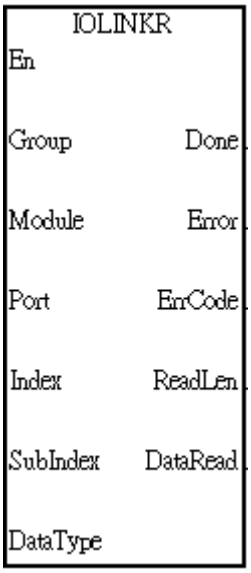
API	指令码	操作数	功能
1426	IOLINKR	**Execute ~ DataRead**	IO-Link 装置参数数据读出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Group								●					○	○		
Module								●					○	○		
Port								●					○	○		
Index								●					○	○		
SubIndex								●					○	○		
DataType								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								
ReadLen								●								
DataRead								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
Port		●				●							
Index		●				●							
SubIndex		●				●							
DataType		●				●							
Done	●												
Error	●												
ErrCode		●				●							
ReadLen		●				●							
DataRead		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Group ： 群组编号

Module ： 模块编号

Port ： 通讯端口编号

Index ： 参数主索引编号

SubIndex ： 参数子索引编号

DataType ： 参数数据类型

Done ： 通讯完成

Error ： 通讯错误

ErrCode ： 错误代码

ReadLen ： 读回数据长度

DataRead ： 读回数据

指令说明：

1. 指令需搭配 PLC 主机韧体 V1.08.50（含）版以上，以及 ISPSoft 软件 V3.12（含）版以上。
2. 用户可以参考 IO-Link 装置的 IODD 文件，通过此指令读取 IO-Link 装置的参数。
3. 此指令无使用次数限制，每台 AS04SIL 模块同一时间只能执行一条通讯指令（IOLINKR、IOLINKW），因此，后面启动的指令将不会被执行。
4. 与 IO-Link 装置的通讯时间，最慢会需要 5 秒，通讯期间请勿关闭或重新执行指令，避免造成错误。再次执行前需先关闭指令再启动，建议使用 Done 或 Error 标志做为关闭与再次启动指令的逻辑条件。
5. **Group** 为指定 AS04SIL 模块连接于主机右侧或者远程模块右侧之群组编号，主机编号为 0，第 1 台远程模块编号为 1，以此原则类推，最大群组数为 15。若指定之模块不为 AS04SIL 模块时，则 Error 错误标志会被设为 ON。
6. **Module** 为连接于主机或远程右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。若指定之模块不为 AS04SIL 模块时，则 Error 错误标志会被设为 ON。
7. **Port** 为指定 AS04SIL 模块的通讯端口编号，数值范围为 1~4，分别代表通讯端口#1~#4。若输入数值不是 AS04SIL 模块的通讯端口编号时，则 Error 错误标志会被设为 ON。
8. **Index** 为欲读出 IO-Link 装置参数的主索引编号。
9. **SubIndex** 为欲读出 IO-Link 装置参数的子索引编号。数值范围为 0~255，若输入数值超过范围，则 Error 错误标志会被设为 ON。
10. **DateType** 为 IO-Link 装置参数的数据类型。用户须设定参数的数据类型，才能正确处理数值类型的数据。若输入数值超过范围，则 Error 错误标志会被设为 ON。

代码	数据类型	说明
0x0000	BooleanT	布尔，数据长度为 1 Byte

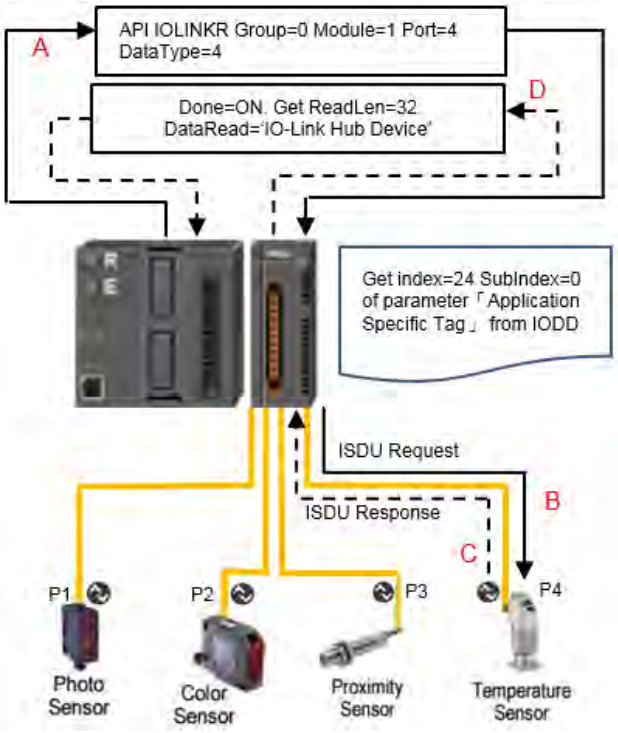
代码	数据类型	说明
0x0001	UIntegerT	无号整数，数据长度为 1、2、4、8 Bytes
0x0002	IntegerT	有号整数，数据长度为 1、2、4、8 Bytes
0x0003	Float32T	浮点数，数据长度为 4 Bytes
0x0004	StringT	不定长度字符串，数据长度为 1~232 Bytes。NULL（0x00）视为字符串结束
0x0005	OctetStringT	固定长度字符串，数据长度依 IODD 属性 fixedLength 定义，为 1~232 Bytes。
0x0006	TimeT	时间，数据长度为 4 Bytes（Resolution：1 second）+ 4 Bytes（Resolution：2^-32 second）
0x0007	TimeSpanT	时间，数据长度为 8 Bytes（Resolution：2^-32 second）
0x0008	ArrayT	需从参数子索引编号 0 读回整个数组
0x0009	RecordT	需从参数子索引编号 0 读回整个记录
其他	保留	保留

11. **Done** 为通讯完成标志。当此标志为 ON 时，代表完成 IO-Link 装置参数数据读取，再依读回数据长度 ReadLen，去解析读回的数据 ReadData。
12. **Error** 为通讯错误标志。当此标志为 ON 时，代表主机、AS04SIL 模块和 IO-Link 装置之间通讯发生错误，可以参考错误代码 ErrCode 来排除错误。
13. **ReadLen** 为 IO-Link 装置参数读回数据长度，以 byte 为单位。
14. **DataRead** 为 IO-Link 装置参数读回数据内容，数据依下 8 位、上 8 位顺序摆放。
15. **ErrCode** 错误代码表如下：

错误代码	说明
0x8000	IO-Link 装置应用错误
0x8011	主索引不存在
0x8012	子索引不存在
0x8020	服务暂时无法取得
0x8021	服务暂时无去取得-本地控制
0x8022	服务暂时无去取得-装置控制
0x8023	拒绝存取
0x8030	参数数值超过范围
0x8031	参数数值超过上限
0x8032	参数数值低于下限
0x8033	参数长度过长
0x8034	参数长度过短
0x8035	功能无法取得

错误代码	说明
0x8036	功能暂时无法取得
0x8040	不合法的参数集
0x8041	不一致的参数集
0x8082	应用尚未就绪
0x8101	IO-Link 装置不在 IO-Link 运行模式
0x8102	通讯端口未连接 IO-Link 装置
0x8103	IO-Link 装置联机建立中
0x8104	通讯端口编号超过范围 1~4
0x8105	参数子索引编号超过范围 0~255
0x8106	读写数据长度超过范围 1~232
0x8107	保留
0x8108	IO-Link 装置通讯逾时 5000ms
0x8109	AS04SIL 群组或模块编号错误
0x810A	内部通讯错误
0x810B	对同一台 AS04SIL 模块，同时执行二个以上 IOLINK 装置读写指令
0x810C	读写装置起始位置+读写数据长度超过读写装置范围
0x810D	参数数据类型数值超过范围
0x810E	参数数据类型与读写数据长度不一致

16. 主机与 AS04SIL 模块、IO-Link 装置通讯流程如下：



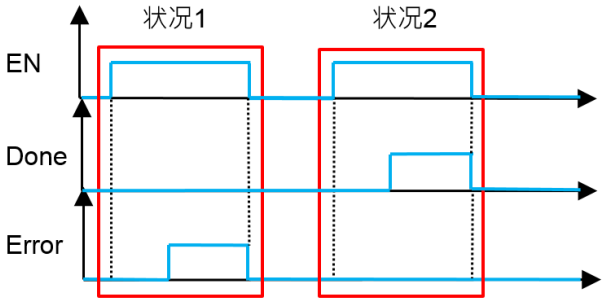
A：主机传送数据信息给 AS04SIL 模块

B：AS04SIL 模块将数据信息传送给目标设备（IO-Link Device）

C：目标设备（IO-Link Device）处理请求信息并将响应信息传送给 AS04SIL 模块

D：主机更新读取状态（Done、Error），接收响应数据

17. EN、Done、Error 标志时序：

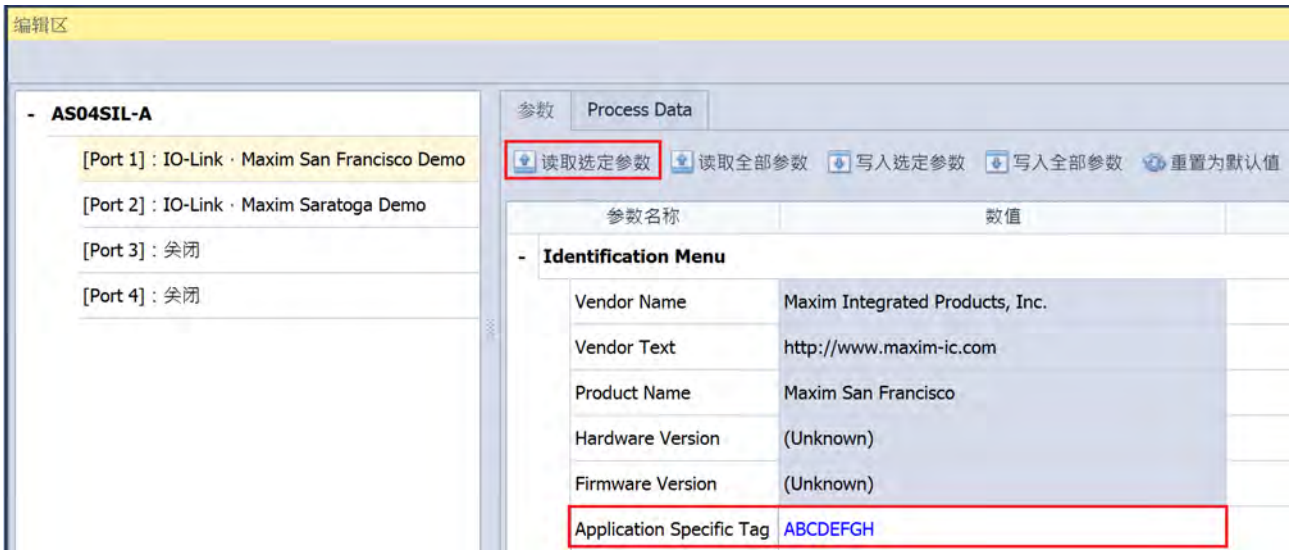


状况 1：通讯发生错误时，输入 EN 与输出 Done、Error 时序图

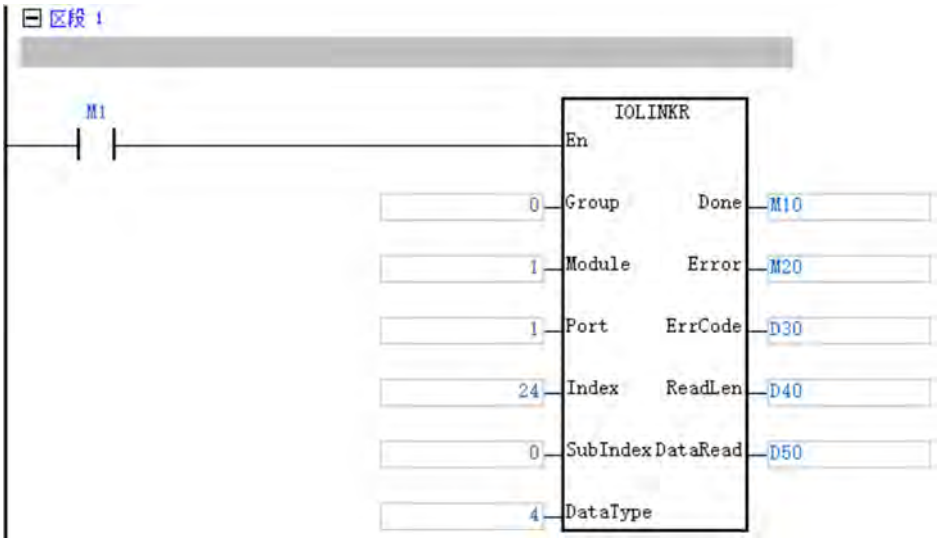
状况 2：通讯正常完成时，输入 EN 与输出 Done、Error 时序图

程序范例 1：读取主机右侧第 1 台通讯端口 1 的装置，参数「Application Specific Tag」

1. 通过 HWCONFIG 读取选定参数得知，装置参数 Application Specific Tag 内存的字符串为「ABCDEFGH」。



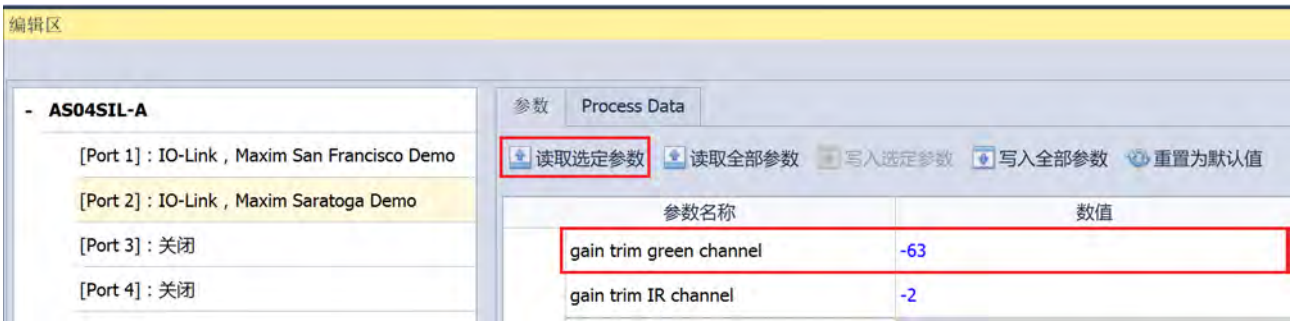
2. 此范例 AS04SIL 配置于 ASCPU 右侧第一台，因此 Group 设定为 0，Module 设定为 1，Port 设为 1。
3. 由 IO-Link 装置使用手册或 IODD 文件得知，其主索引为 24，子索引为 0，数据类型为 StringT，因此 Index 设为 24，SubIndex 设为 0，DataType 设为 4。
4. 设定 M0=ON 时，开始传送读取请求，当 Done（M10）为 ON 时，表示收到装置响应消息且完成读取，此时 ReadLen（D40）值为 32，字符串暂存在 DataRead（D50~D65），IO-Link 装置参数读回数据内容，数据依下 8 位、上 8 位顺序摆放。



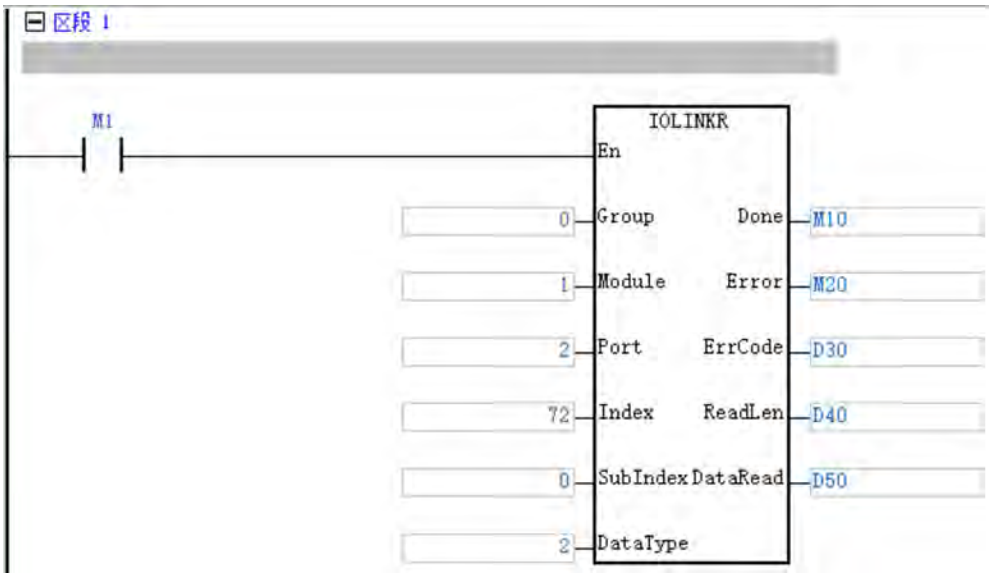
	来源	符号名称	装置名称	状态	数据类型	值(16位)	值(32位)	值(32位浮点数)	数值类型	注释
			M1							
			M10							
			M20							
			D30			0000	00000000	0.0000	十六进制	
▸			D40			32	32	0.0000	有号数十进制	
			D50			4241	44434241	781.0352	十六进制	
			D51			4443	46454443	12625.0654	十六进制	
			D52			4645	48474645	204057.0781	十六进制	
			D53			4847	00004847	0.0000	十六进制	
			D54			0000	00000000	0.0000	十六进制	
			D55			0000	00000000	0.0000	十六进制	
			D56			0000	00000000	0.0000	十六进制	
			D57			0000	00000000	0.0000	十六进制	
			D58			0000	00000000	0.0000	十六进制	
			D59			0000	00000000	0.0000	十六进制	
			D60			0000	00000000	0.0000	十六进制	
			D61			0000	00000000	0.0000	十六进制	
			D62			0000	00000000	0.0000	十六进制	
			D63			0000	00000000	0.0000	十六进制	
			D64			0000	00000000	0.0000	十六进制	

程序范例 2：读取主机右侧第 1 台通讯端口 2 的装置，参数「gain trim green channel」

1. 通过 HWCONFIG 读取选定参数得知，装置参数 gain trim green channel 内存的数值为-63。



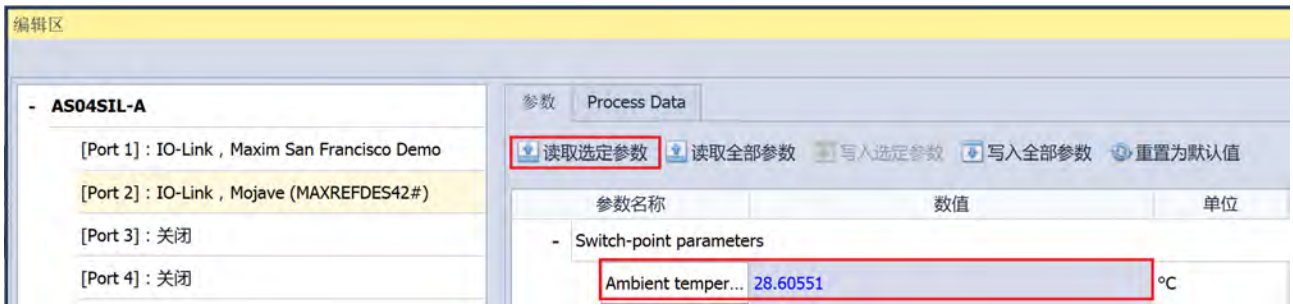
2. 此范例 AS04SIL 配置于 ASCPU 右侧第一台，因此 Group 设定为 0，Module 设定为 1，Port 设为 2
3. 由 IO-Link 装置使用手册或 IODD 文件得知，其主索引为 72，子索引为 0，数据类型为 IntegerT，因此 Index 设为 72，SubIndex 设为 0，DataType 设为 2。
4. 设定 M0=ON 时，开始传送读取请求，当 Done（M10）为 ON 时，表示收到装置响应消息且完成读取，此时 ReadLen（D40）值为 1，数值暂存在 DataRead（D50 Low Byte），IO-Link 装置参数读回数据内容，数据依下 8 位、上 8 位顺序摆放。



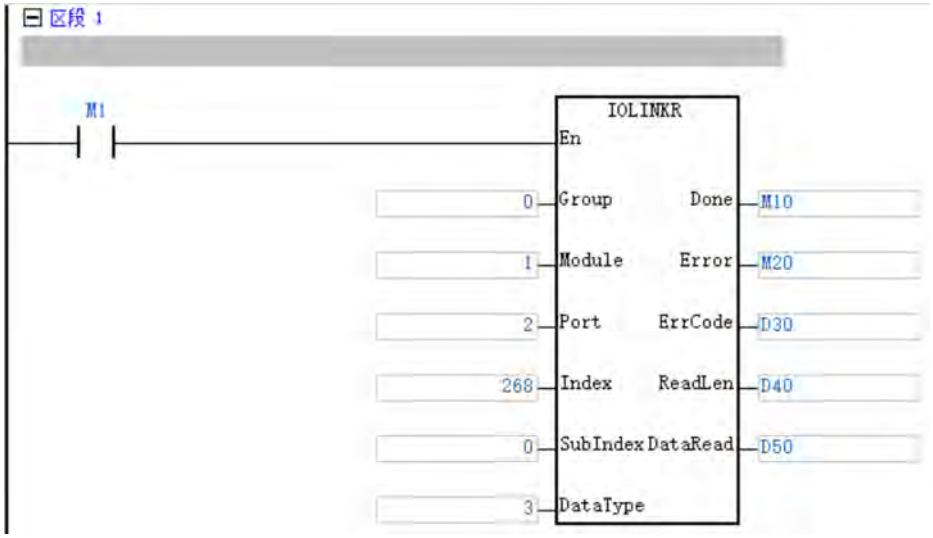
来源	符号名称	装置名称	状态	数据类型	值(16位)	值(32位)	值(32位浮点数)	数值类型	注释
		M1							
		M10							
		M20							
		D30			0000	00000000	0.0000	十六进制	
		D40			1	1	0.0000	有号数十进制	
		D50			00C1	000000C1	0.0000	十六进制	

程序范例 3：读取主机右侧第 1 台通讯端口 2 的装置，参数「Ambient temperature」

1. 通过 HWCONFIG 读取选定参数得知，装置参数 Ambient temperature 内存的数值为 28.60551。



2. 此范例 AS04SIL 配置于 ASCPU 右侧第一台，因此 Group 设定为 0，Module 设定为 1，Port 设为 2
3. 由 IO-Link 装置使用手册或 IODD 文件得知，其主索引为 268，子索引为 0，数据类型为 Float32T，因此 Index 设为 268，SubIndex 设为 0，DataType 设为 3。
4. 设定 M0=ON 时，开始传送读取请求，当 Done (M10) 为 ON 时，表示收到装置响应消息且完成读取，此时 ReadLen (D40) 值为 4，数值暂存在 DataRead (D50~D51)，IO-Link 装置参数读回数据内容，数据依下 8 位、上 8 位顺序摆放。



	来源	符号名称	装置名称	状态	数据类型	值(16位)	值(32位)	值(32位浮点数)	数值类型	注释
			M1							
			M10							
			M20							
			D30			0000	00000000	0.0000	十六进制	
			D40			4	4	0.0000	有号数十进制	
			D50			999F	41E5999F	28.7000	十六进制	
I			D51			41E5	000041E5	0.0000	十六进制	

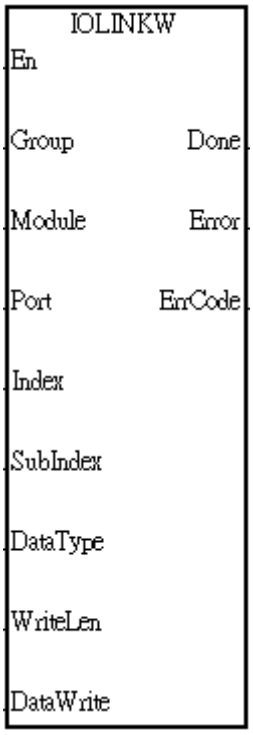
API	指令码	操作数	功能
1427	IOLINKW	**Execute ~ ErrCode**	IO-Link 装置参数数据写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Group								●					○	○		
Module								●					○	○		
Port								●					○	○		
Index								●					○	○		
SubIndex								●					○	○		
DataType								●					○	○		
WriteLen								●					○	○		
DataWrite								●								
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
Port		●				●							
Index		●				●							
SubIndex		●				●							
DataType		●				●							
WriteLen		●				●							
DataWrite		●				●							
Done	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Group ：群组编号

Module ：模块编号

Port ：通讯端口编号

Index ：参数主索引编号

SubIndex ：参数子索引编号

DataType ：参数数据类型

WriteLen ：写入数据长度

DataWrite ：写入数据

Done ：通讯完成

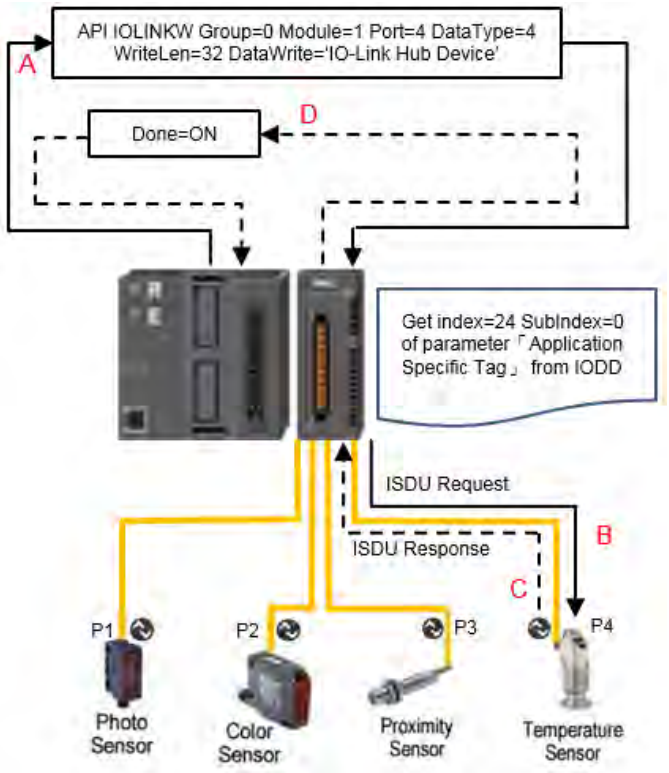
Error ：通讯错误

ErrCode ：错误代码

指令说明：

1. 指令需搭配 PLC 主机韧体 V1.08.50（含）版以上，以及 ISPSoft 软件 V3.12（含）版以上。
2. 用户可以参考 IO-Link 装置的 IODD 文件，通过此指令写入 IO-Link 装置的参数。
3. 此指令无使用次数限制，每台 AS04SIL 模块同一时间只能执行一条读写指令（IOLINKR、IOLINKW），因此，后面启动的指令将不会被执行。
4. 与 IO-Link 装置的通讯时间，最慢会需要 5 秒，通讯期间请勿关闭或重新执行指令，避免造成错误。再次执行前需先关闭指令再启动，建议使用 Done 或 Error 标志做为关闭与再次启动指令的逻辑条件。
5. **Group** 为指定 AS04SIL 模块连接于主机右侧或者远程模块右侧之群组编号，主机编号为 0，第 1 台远程模块编号为 1，以此原则类推，最大群组数为 15。若指定之模块不为 AS04SIL 模块时，则 Error 错误标志会被设为 ON。
6. **Module** 为连接于主机或远程右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数，最大限制台数为 32 台。若指定之模块不为 AS04SIL 模块时，则 Error 错误标志会被设为 ON。
7. **Port** 为指定 AS04SIL 模块的通讯端口编号，数值范围为 1~4，分别代表通讯端口#1~#4。若输入数值不是 AS04SIL 模块的通讯端口编号时，则 Error 错误标志会被设为 ON。
8. **Index** 为欲写入 IO-Link 装置参数的主索引编号。
9. **SubIndex** 为欲写入 IO-Link 装置参数的子索引编号。数值范围为 0~255，若输入数值超过范围，则 Error 错误标志会被设为 ON。
10. **DataType** 为参数数据类型。请参考 API 1426 指令 IOLINKR 之参数数据类型代码表。

11. **WriteLen** 为 IO-Link 装置参数写入数据长度，以 byte 为单位。数值范围为 1~232，若输入数值超过范围，则 Error 错误标志会被设为 ON。
12. **DataWrite** 为 IO-Link 装置参数写入数据内容，数据依上 8 位、下 8 位顺序摆放。
13. **Done** 为通讯完成标志。当此标志为 ON 时，代表完成 IO-Link 装置参数数据写入。
14. **Error** 为通讯错误标志。当此标志为 ON 时，代表主机、AS04SIL 模块、IO-Link 装置之间通讯发生错误，可以参考错误代码 ErrCode 来排除错误。
15. **ErrCode** 错误代码表请参考 API 1426 指令码 IOLINKR 之错误代码表。
16. 主机与 AS04SIL 模块、IO-Link 装置通讯流程如下：



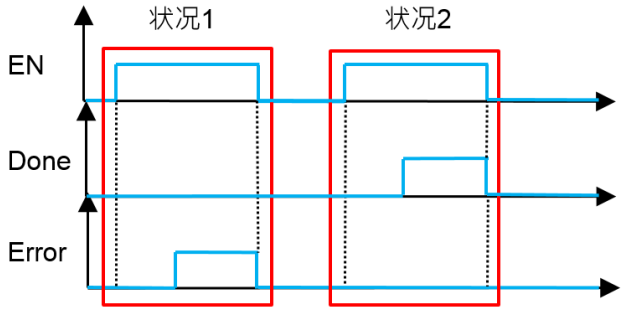
A：主机传送数据信息给 AS04SIL 模块

B：AS04SIL 模块将数据信息传送给目标设备（IO-Link Device）

C：目标设备（IO-Link Device）处理请求信息并将响应信息传送给 AS04SIL 模块

D：主机更新读取状态（Done、Error），接收响应数据

17. EN、Done、Error 标志时序：

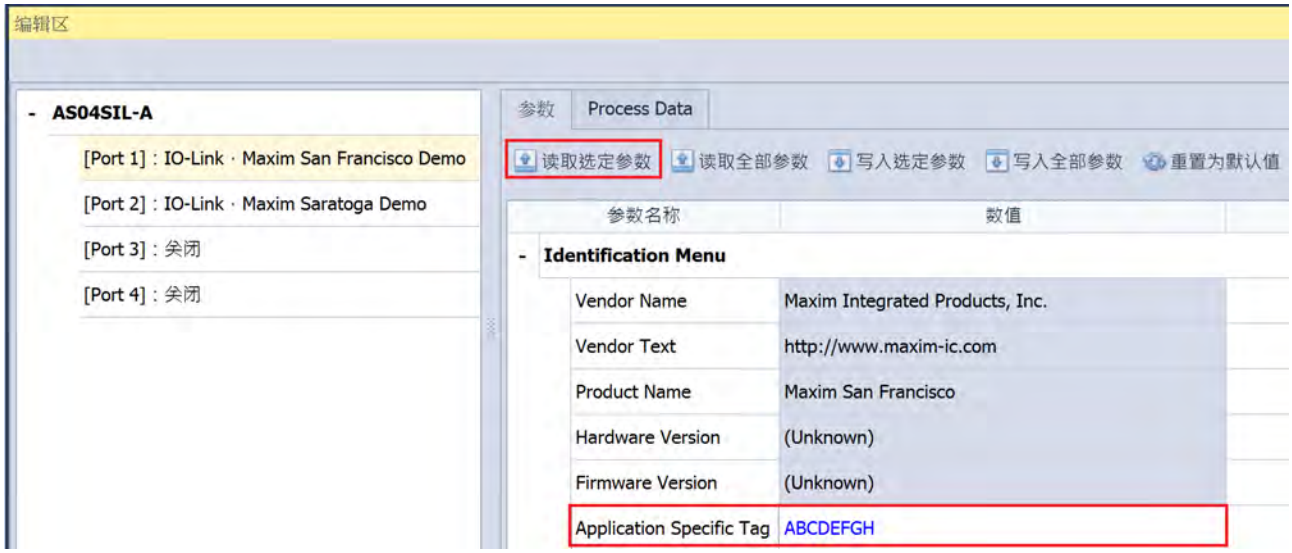


状况 1：通讯发生错误时，输入 EN 与输出 Done、Error 时序图

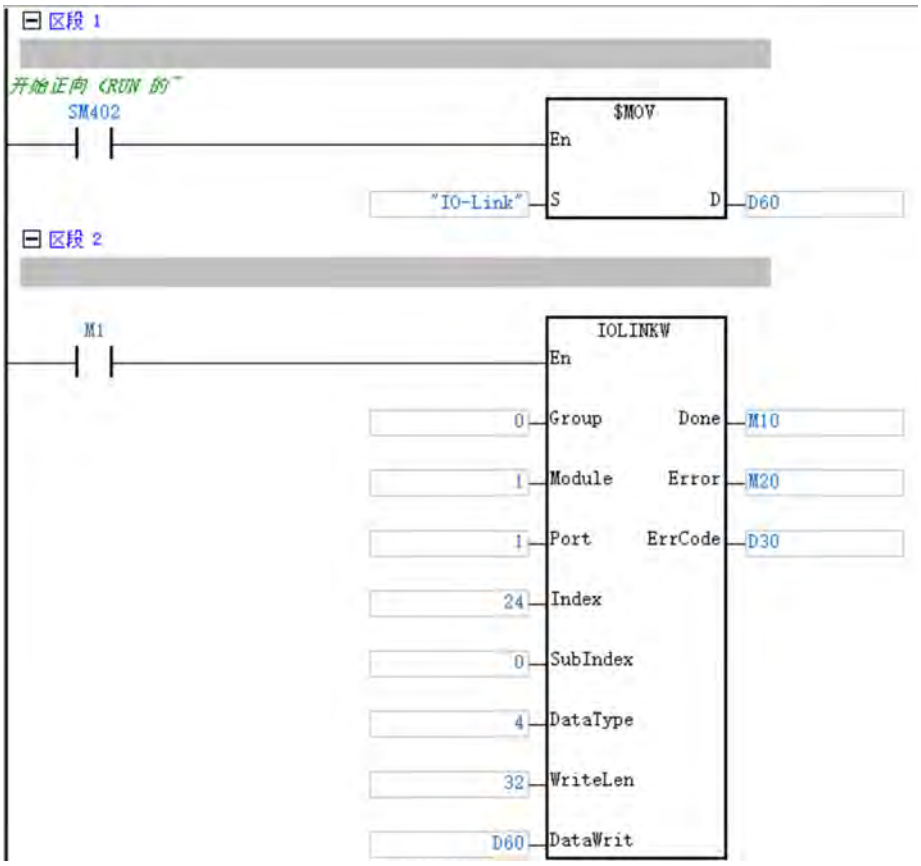
状况 2：通讯正常完成时，输入 EN 与输出 Done、Error 时序图

程序范例 1：写入主机右侧第 1 台通讯端口 1 的装置，参数「Application Specific Tag」

1. 通过 HWCONFIG 读取选定参数得知，装置参数 Application Specific Tag 内存的字符串原本为「ABCDEFGH」。



2. 此范例 AS04SIL 配置于 ASCPU 右侧第一台，因此 Group 设定为 0，Module 设定为 1，Port 设为 1
3. 由 IO-Link 装置使用手册或 IODD 文件得知，其主索引为 24，子索引为 0，数据类型为 StringT，因此 Index 设为 24，SubIndex 设为 0，DataType 设为 4。
4. 用指令$MOV 从 D60 开始写入字符串「IO-Link」。
5. 设定 M1=ON 时，开始传送写入请求，当 Done（M10）为 ON 时，表示收到装置响应消息且完成写入，此时 WriteLen 值为 32，字符串暂存在 DataWrite（D60~D75），IO-Link 装置参数写入数据内容，数据依上 8 位、下 8 位顺序摆放。



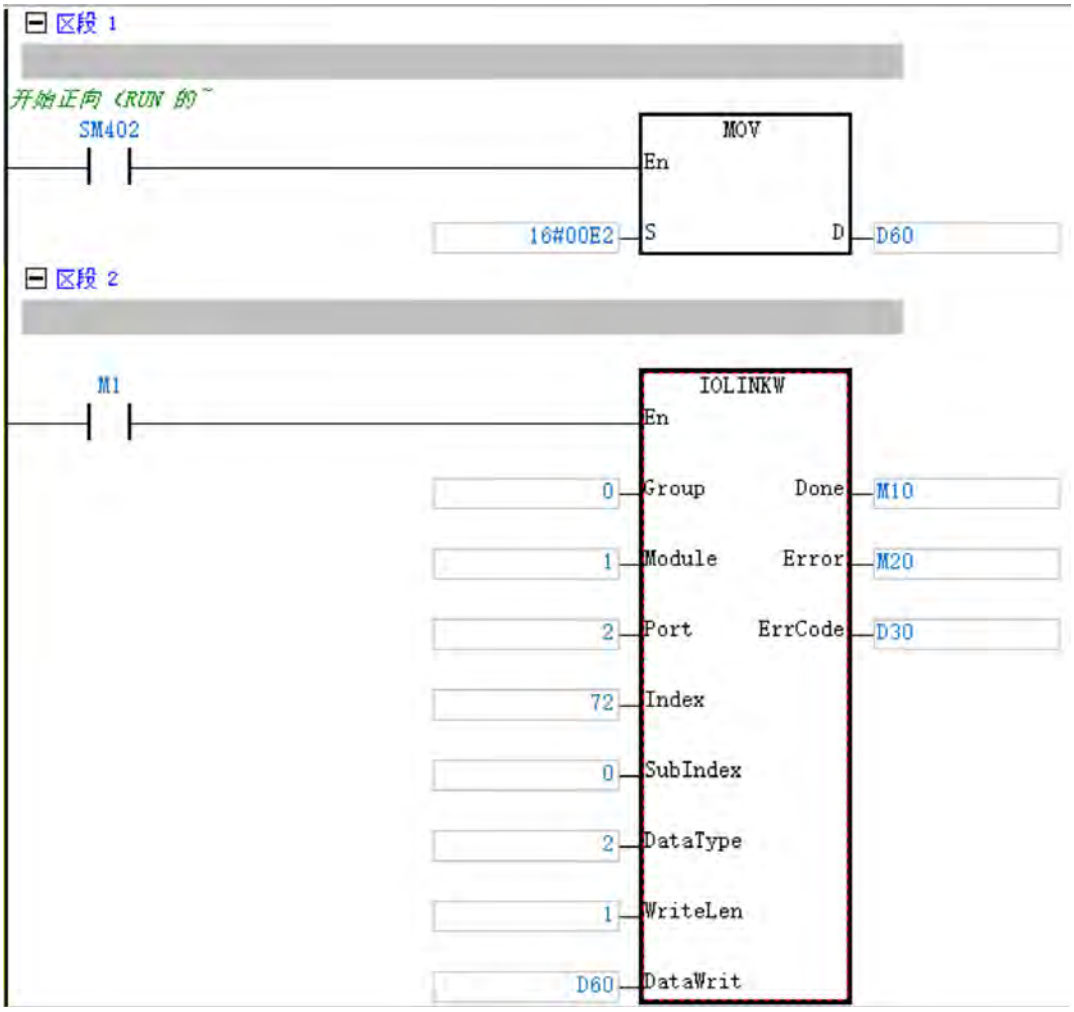
来源	符号名称	装置名称	状态	数据类型	值(16位)	值(32位)	值(32位浮点数)	数值类型	注释
		M1							
		M10							
		M20							
		D30			0000	00000000	0.0000	十六进制	
		D60			4F49	4C2D4F49	45432100.0000	十六进制	
		D61			4C2D	6E494C2D	1557464444607181999940	十六进制	
		D62			6E49	006B6E49	0.0000	十六进制	
		D63			006B	0000006B	0.0000	十六进制	
		D64			0000	00000000	0.0000	十六进制	
		D65			0000	00000000	0.0000	十六进制	
		D66			0000	00000000	0.0000	十六进制	
		D67			0000	00000000	0.0000	十六进制	
		D68			0000	00000000	0.0000	十六进制	
		D69			0000	00000000	0.0000	十六进制	
		D70			0000	00000000	0.0000	十六进制	
		D71			0000	00000000	0.0000	十六进制	
		D72			0000	00000000	0.0000	十六进制	
		D73			0000	00000000	0.0000	十六进制	
		D74			0000	00000000	0.0000	十六进制	

6. 由 HWCONFIG 再次读取选定参数得知，装置参数 Application Specific Tag 内存的字符串已更新为「IO-Link」。

程序范例 2：写入主机右侧第 1 台通讯端口 2 的装置，参数「gain trim green channel」

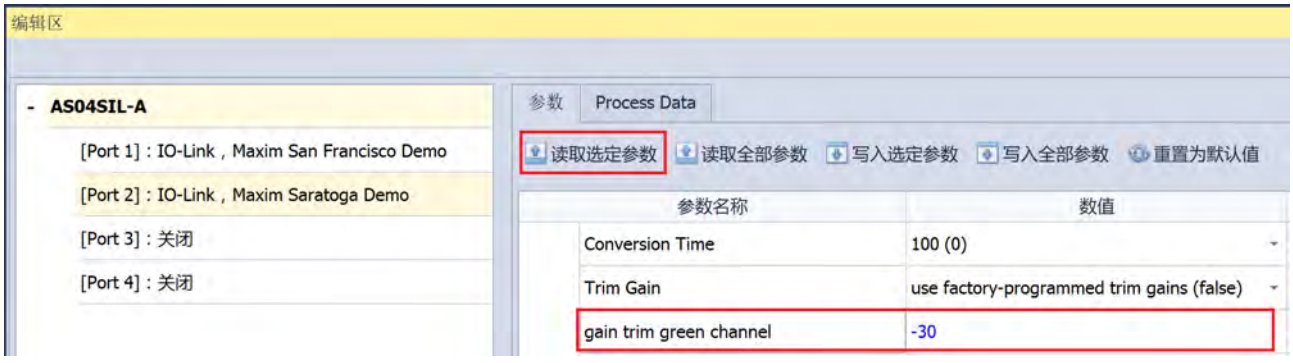
1. 通过 HWCONFIG 读取选定参数得知，装置参数 gain trim green channel 内存的数值原本为-63。

2. 此范例 AS04SIL 配置于 ASCPU 右侧第一台，因此 Group 设定为 0，Module 设定为 1，Port 设为 2
3. 由 IO-Link 装置使用手册或 IODD 文件得知，其主索引为 72，子索引为 0，数据类型为 IntegerT，因此 Index 设为 72，SubIndex 设为 0，DataType 设为 2。
4. 用指令 MOV 在 D60 开始写入数值 0xE2（-30）。
5. 设定 M1=ON 时，开始传送读取请求，当 Done（M10）为 ON 时，表示收到装置响应消息且完成写入，此时 WriteLen 值为 1，数值暂存在 DataRead（D60 Low Byte），IO-Link 装置参数写入数据内容，数据依上 8 位、下 8 位顺序摆放。



	来源	符号名称	装置名称	状态	数据类型	值(16位)	值(32位)	值(32位浮点数)	数值类型	注释
▸			M1							
			M10							
			M20							
			D30			0000	00000000	0.0000	十六进制	
			D60			00E2	000000E2	0.0000	十六进制	

6. 由 HWCONFIG 再次读取选定参数得知，装置参数 gain trim green channel 内存的字符串已更新为-30。



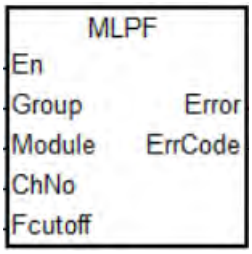
API	指令码	操作数	功能
1430	MLPF	**Group ~ ErrCode**	模拟输入低通滤波器

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Group								●					○	○		
Module								●					○	○		
ChNo								●					○	○		
Fcutoff								●					○	○		
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
ChNo		●				●							
Fcutoff		●				●							
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Group 群组编号

Module ：模块编号

ChNo ：通道编号

Fcutoff ：滤波器截止频率

Error ：错误标志

ErrCode ：错误代码

指令说明：

1. 此指令为提供模拟输入信号进行数字低通滤波器的功能。需搭配 ISPSoft 软件 V3.16（含）版以上，以及需搭配的装置版本如下表所示：

类别	型号	支持韧体版本	备注
AD 功能卡	AS-F2AD	V1.40（含）以上	主机版本需为 V1.14（含）以上。
扩充模块	AS04AD-A AS08AD-B AS08AD-C AS06XA-A	V1.02（含）以上	主机版本需为 V1.14（含）以上。 远程右侧模拟输入模块除了主机版本需为 V1.14（含）以上， AS00SCM-A 版本亦需为 V2.08（含）以上。
	AS02LC-A	V1.06（含）以上	
主机内建 AD 通道	AS218	主核心 V1.12（含）以上 副核心 V1.12（含）以上	

2. 开启此数字滤波器指令后，该通道对应之输入值将由原本显示的平均值改为显示数字滤波器滤波结果。
3. 指定的设备中，任一通道开启此数字滤波器指令时，所有模拟通道之取样时间将被强制指定，原本取样时间设定值将失效；而当数字滤波器指令被关闭，又会以原本取样时间设定值进行取样。

机种名称	AS-F2AD	AS04AD-A AS08AD-B AS08AD-C AS06XA-A	AS02LC-A	主机内建 AD
强制指定之取样时间	2ms	2ms	10ms	2ms

4. **Group** 指定模拟输入模块连接于主机右侧或者远程模块右侧之群组编号，主机编号为 0，第 1 台远程模块编号为 1，以此原则类推，最大群组数为 15。
5. **Module** 为模块编号。若为主机内建之模拟输入通道，此值须设定为 0；若为 AS-F2AD 功能卡上之模拟通道，请依所在之功能卡槽设定为 33 或 34。

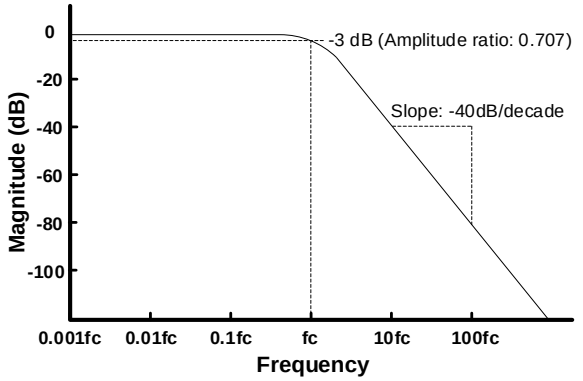
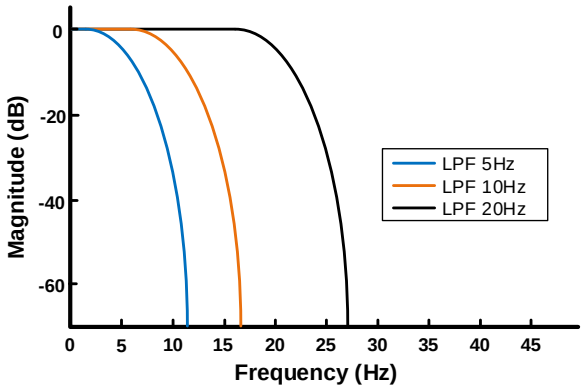
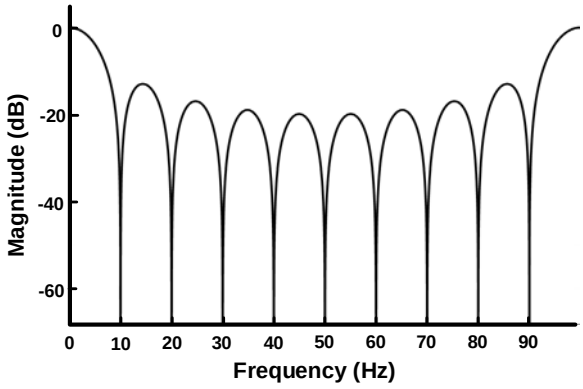
Module	说明
0	主机内建模拟输入通道
1	连接于主机右侧的扩充模块编号 1
2	连接于主机右侧的扩充模块编号 2
3 ~ 32	以上述编排规则计算，最大可连接 32 台扩充模块。
33	功能卡槽 1
34	功能卡槽 2

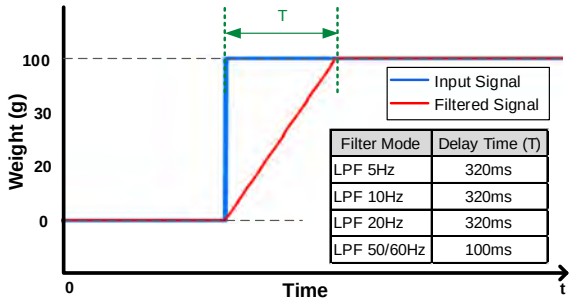
6. **ChNo** 指定启用低通滤波器之通道编号。

Module	ChNo	说明
0	1	主机内建模拟输入通道 1
	2	主机内建模拟输入通道 2
1	1	第 1 台扩充模块的输入通道 1
	2 ~ 8	第 1 台扩充模块的输入通道 2 ~ 8
2	1	第 2 台扩充模块的输入通道 1

Module	ChNo	说明
	2 ~ 8	第 2 台扩充模块的输入通道 2 ~ 8
3 ~ 32		以上述规则累推
33	1	功能卡槽 1　AS-F2AD 之模拟输入通道 1
33	2	功能卡槽 1　AS-F2AD 之模拟输入通道 2
34	1	功能卡槽 2　AS-F2AD 之模拟输入通道 1
34	2	功能卡槽 2　AS-F2AD 之模拟输入通道 2

7.　Fcutoff 为低通滤波器之截止频率（cut-off frequency），设定范围如下表：

机种名称	Fcutoff 设定范围	响应特性
AS-F2AD、AS218 内建 AIO 通道、AS04AD-A、AS08AD-B、AS08AD-C、AS06XA-A	5~100，单位 Hz	
AS02LC-A	0：5Hz 1：10Hz 2：20Hz 3：50/60Hz （针对 50/60Hz 衰减）	LPF 5Hz，10Hz，20Hz 频率响应：  LPF 50/60Hz 频率响应： 

机种名称	Fcutoff 设定范围	响应特性
		输入延迟时间： 

8. 使用中若需变更参数，需先将指令关闭，参数变更完成后再重启指令。
9. 当 **Error** 错误标志为 ON 时，请参考 **ErrCode** 错误代码如下表说明：

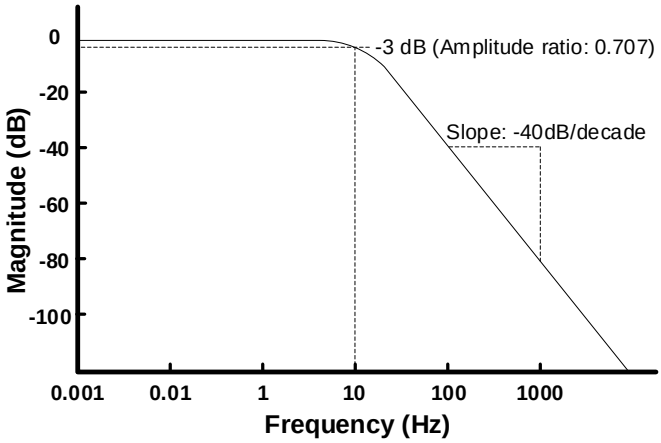
错误代码	说明
0x1400	设备不支持此功能
0x1401	群组编号或模块编号设定错误
0x1402	模块无此通道
0x1403	**Fcutoff** 超出设定范围
0x1404	模块无响应，通讯逾时
0x1405	指定之通道数字滤波功能已开启，不允许重复指定

程序范例：

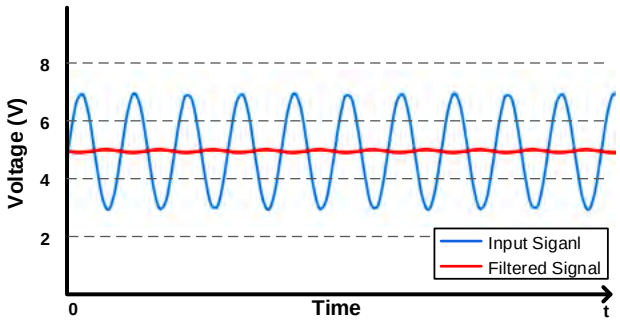
6

范例 1：

1. 以 AS218 内建模拟输入通道 1 为例，设定 **Group = 0**、**Module = 0**、**ChNo = 1**、**Fcutoff = 10（Hz）**。
2. 此数字低通滤波器频率响应如下：

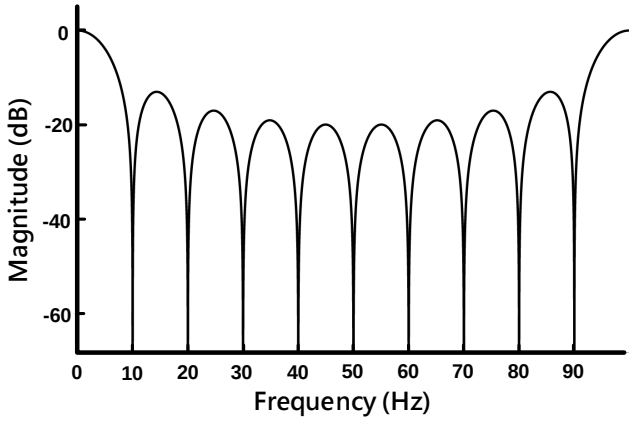


3. 输入一测试信号：5Vdc + 60Hz 的弦波（峰对峰值为 4V），如下图蓝色波形。经过 LPF 低通滤波器后，输出结果为红色波形，可以看出 60Hz 的弦波被衰减，剩下 5Vdc 的直流成分。

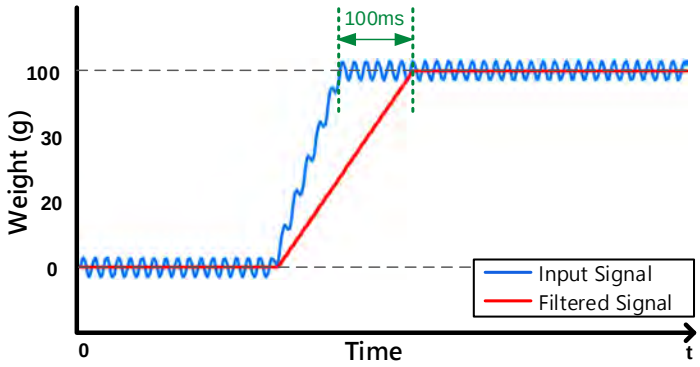


范例 2：

1. 以主机右侧第一台 AS02LC-A 输入通道 2 为例，设定 **Group = 0**、**Module = 1**、**ChNo = 2**、**Fcutoff = 3（50/60Hz）**。
2. 此数字低通滤波器频率响应如下：



3. 输入一测试信号：Load cell 信号加上 50Hz 的弦波噪声，如下图蓝色波形。经过 LPF 低通滤波器后，输出结果为红色波形，可以看出 50Hz 的弦波被明显的衰减。



6.16 浮点数指令

6.16.1 浮点数指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1500	–	FSIN	✓	浮点数 SIN 运算
1501	–	FCOS	✓	浮点数 COS 运算
1502	–	FTAN	✓	浮点数 TAN 运算
1503	–	FASIN	✓	浮点数 ASIN 运算
1504	–	FACOS	✓	浮点数 ACOS 运算
1505	–	FATAN	✓	浮点数 ATAN 运算
1506	–	FSINH	✓	浮点数 SINH 运算
1507	–	FCOSH	✓	浮点数 COSH 运算
1508	–	FTANH	✓	浮点数 TANH 运算
1509	–	FRAD	✓	角度→弧度
1510	–	FDEG	✓	弧度→角度
1511	SQR	DSQR	✓	BIN 开平方根
1512	–	FSQR	✓	浮点数开平方根
1513	–	FEXP	✓	浮点数取指数
1514	–	FLOG	✓	浮点数取对数
1515	–	FLN	✓	浮点数取自然对数
1516	–	FPOW	✓	浮点数权值指令
1517	RAND	–	✓	随机数值

6.16.2 浮点数指令说明

API	指令码		操作数	功能
1500	FSIN	P	**S**，**D**	浮点数 SIN 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

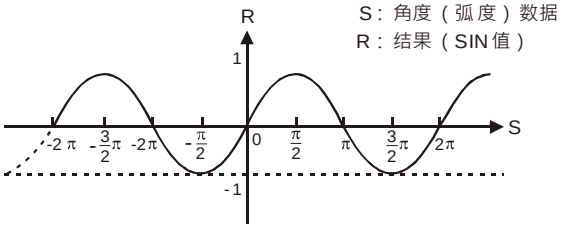
符号：

S ： 指定的来源值

D ： 取 SIN 值结果

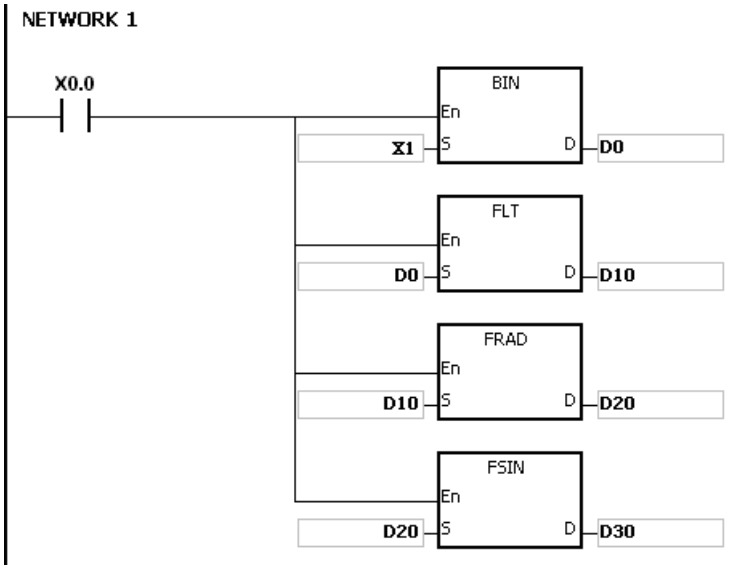
指令说明：

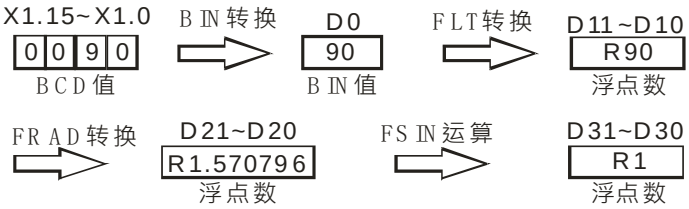
1. **S** 所指定的来源可为弧度或角度，由标志 SM695 决定。
2. 当 SM695=OFF 时，指定为弧度模式，弧度值=角度×π/180。
3. 当 SM695=ON 时，指定为角度模式，角度值=弧度×180/π，角度范围：$0^0 \leqq$角度值$\leqq 360^0$。
4. 若转换结果为 0，则零标号 SM600=ON。
5. 将 S 所指定的来源值，求取 SIN 值后存于 D 所指定的寄存器当中。
6. 下图显示角度（弧度）与结果的关系：



程序范例：

当 X0.0=ON 时，将(X1.15~X1.0)的 BCD 值转成 BIN 值存于 D0，将 D0 做 FLT 运算转成浮点数后存于(D11，D10)，将（D11，D10）做 FRAD 运算转成弧度存于（D21，D20），再将（D21，D20）的弧度做 SIN 运算后存于（D31，D30）当中，内容为浮点数。





补充说明：

1. 当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。
2. 当 SM695=ON，且 **S** 的内容值小于 0 或大于 360，指令不执行，SM0=ON，错误码 SR0=16#2003。

API	指令码			操作数	功能
1501		FCOS	P	**S**，**D**	浮点数 COS 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

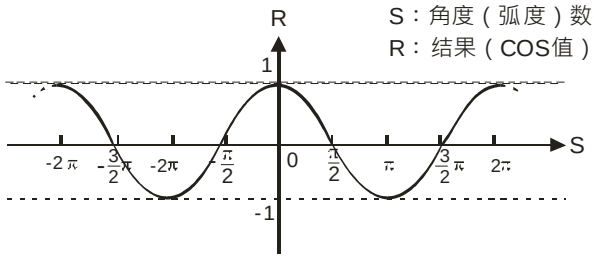
符号：

S ： 指定的来源值

D ： 取 COS 值结果

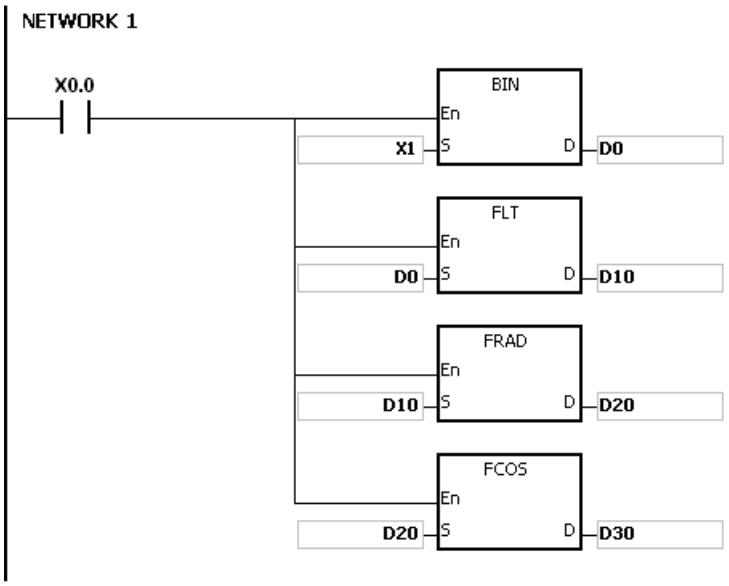
指令说明：

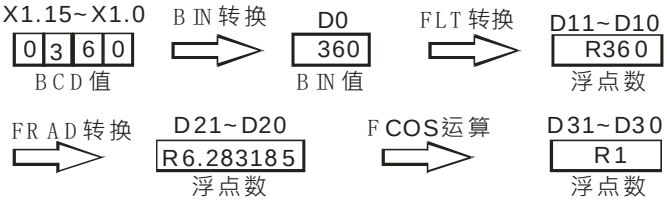
1. **S** 所指定的来源可为弧度或角度，由标志 SM695 决定。
2. 当 SM695=OFF 时，指定为弧度模式，弧度值=角度×π/180。
3. 当 SM695=ON 时，指定为角度模式，角度值=弧度×180/π，角度范围：$0^0 \leqq$角度值$\leqq 360^0$。
4. 若转换结果为 0，则零标号 SM600=ON。
5. 将 **S** 所指定的来源值，求取 COS 值后存于 **D** 所指定的寄存器当中。
6. 下图显示角度与结果的关系：



程序范例：

当 X0.0=ON 时，将(X1.15~X1.0)的 BCD 值转成 BIN 值存于 D0，将 D0 做 FLT 运算转成浮点数后存于(D11，D10)，将（D11，D10）做 FRAD 运算转成弧度存于（D21，D20），再将（D21，D20）的弧度做 COS 运算后存于（D31，D30）当中，内容为浮点数。





补充说明：

1. 当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。
2. 当 SM695=ON，且 **S** 的内容值小于 0 或大于 360，指令不执行，SM0=ON，错误码 SR0=16#2003。

API	指令码			操作数	功能
1502		FTAN	P	**S**，**D**	浮点数 TAN 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

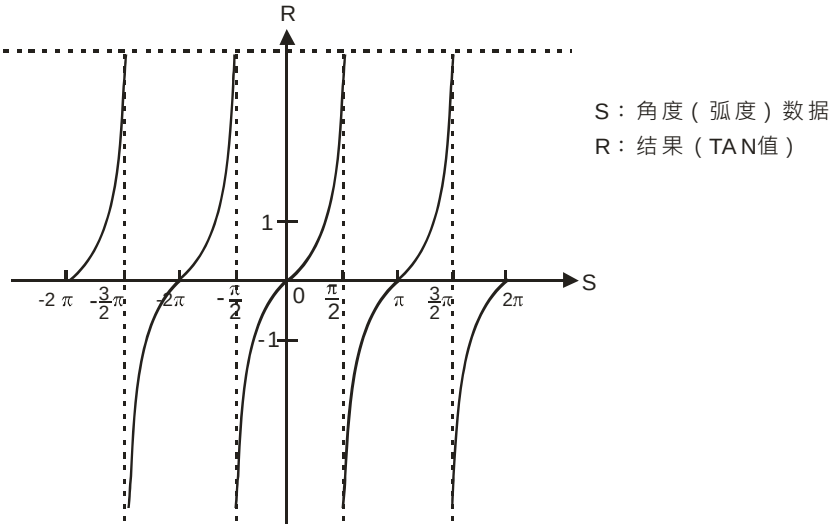
符号：

S ： 指定的来源值

D ： 取 TAN 值结果

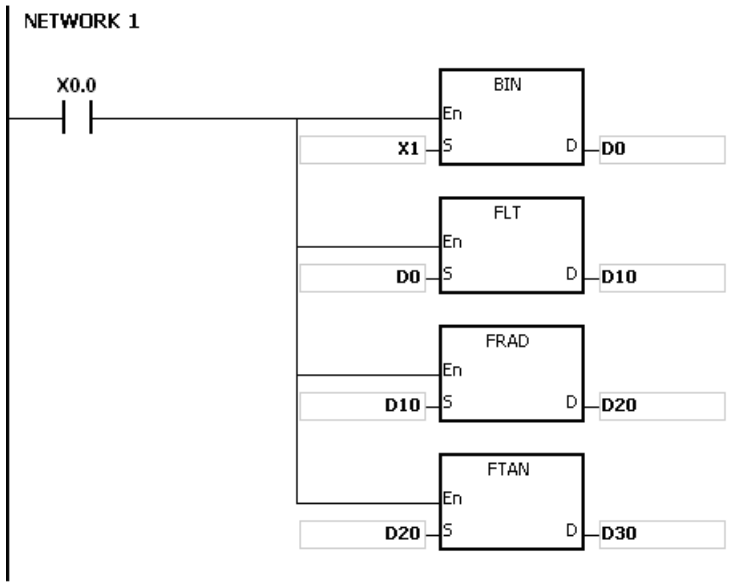
指令说明：

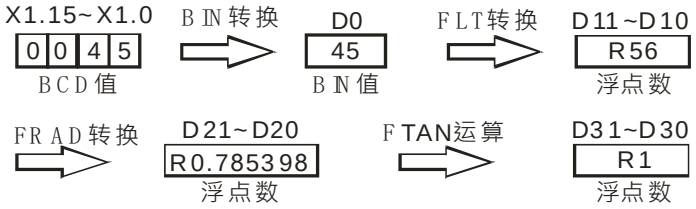
1. **S** 所指定的来源可为弧度或角度，由标志 SM695 决定。
2. 当 SM695=OFF 时，指定为弧度模式，弧度值=角度×π/180。
3. 当 SM695=ON 时，指定为角度模式，角度值=弧度×180/π，角度范围：$0^0 \leqq$ 角度值 $\leqq 360^0$。
4. 若转换结果为 0，则零标号 SM600=ON。
5. 将 **S** 所指定的来源值，求取 TAN 值后存于 **D** 所指定的寄存器当中。
6. 下图显示角度与结果的关系：



程序范例：

当 X0.0=ON 时，将(X1.15~X1.0)的 BCD 值转成 BIN 值存于 D0，将 D0 做 FLT 运算转成浮点数后存于(D11，D10)，将（D11，D10）做 FRAD 运算转成弧度存于（D21，D20），再将（D21，D20）的弧度做 TAN 运算后存于（D31，D30）当中，内容为浮点数。





补充说明：

1. 当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。
2. 当 SM695=ON，且 **S** 的内容值小于 0 或大于等于 360 或等于 90 或等于 270 时，指令不执行，SM0=ON，错误码 SR0=16#2003。

API	指令码			操作数	功能
1503		FASIN	P	**S**，**D**	浮点数 ASIN 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

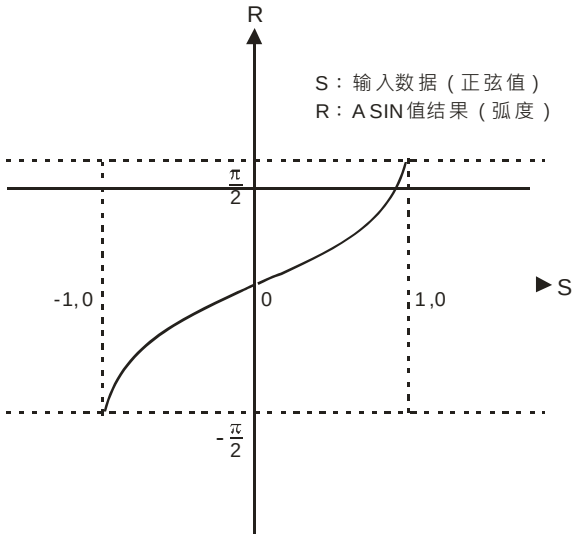
符号：

S ： 指定的来源值

D ： 取 ASIN 值结果

指令说明：

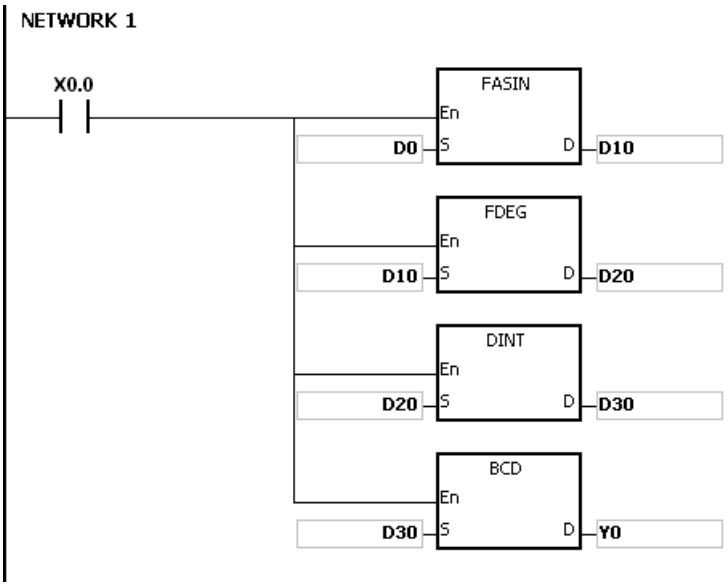
1. ASIN 值=$\sin^{-1}$，下图显示输入数据与结果的关系：

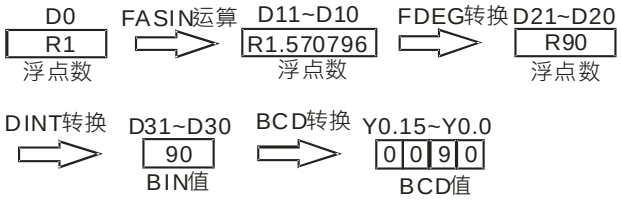


2. 若转换结果为 0，则零标号 SM600=ON。

程序范例：

当 X0.0=ON 时，指定浮点数（D1，D0）求取 ASIN 值后存于（D11，D10）当中，将（D11，D10）的结果做 FDEG 转为角度存于（D21，D20），再将（D21，D20）做 DINT 转换成整数存于（D31，D30），再透过 BCD 指令将求得角度输出于 Y0.15~Y0.0。





补充说明：

1. **S** 操作数指定的正弦值数值之十进浮点值只能介于–1.0~+1.0 之间，若不在此范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

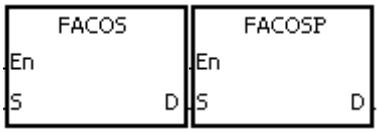
API	指令码			操作数	功能
1504		FACOS	P	**S**，**D**	浮点数 ACOS 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

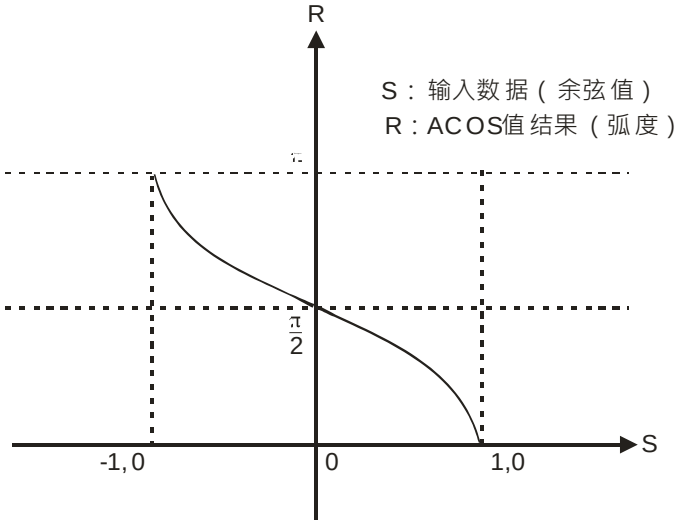


S ： 指定的来源值

D ： 取 ACOS 值结果

指令说明：

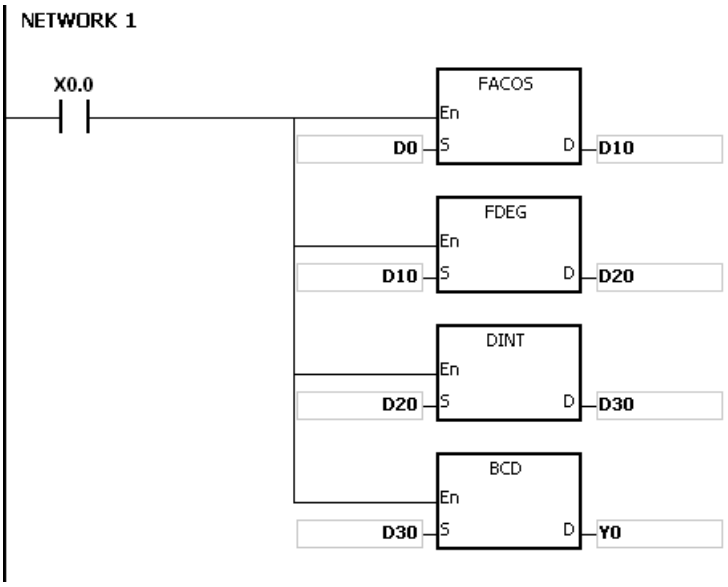
1. ACOS 值=$\cos^{-1}$，下图显示输入数据与结果的关系：

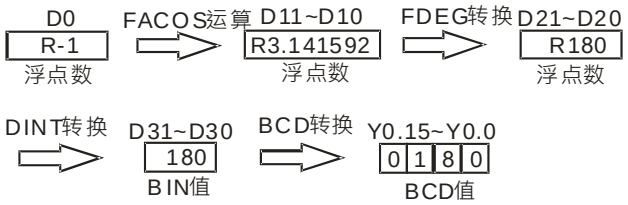


2. 若转换结果的绝对值大于可表示之最大浮点值，则进位标号 SM602=ON。
3. 若转换结果的绝对值小于可表示之最小浮点值，则借位标号 SM601=ON。
4. 若转换结果为 0，则零标号 SM600=ON。

程序范例：

当 X0.0=ON 时，指定浮点数（D1，D0）求取 ACOS 值后存于（D11，D10）当中，将（D11，D10）的结果做 FDEG 转为角度存于（D21，D20），再将（D21，D20）做 DINT 转换成整数存于（D31，D30），再透过 BCD 指令将求得角度输出于 Y0.15~Y0.0。





补充说明：

1. **S** 操作数指定的正弦值数值之十进浮点值只能介于–1.0~+1.0 之间，若不在此范围，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

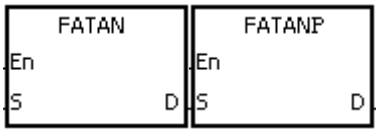
API	指令码			操作数	功能
1505		FATAN	P	**S**、**D**	浮点数 ATAN 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



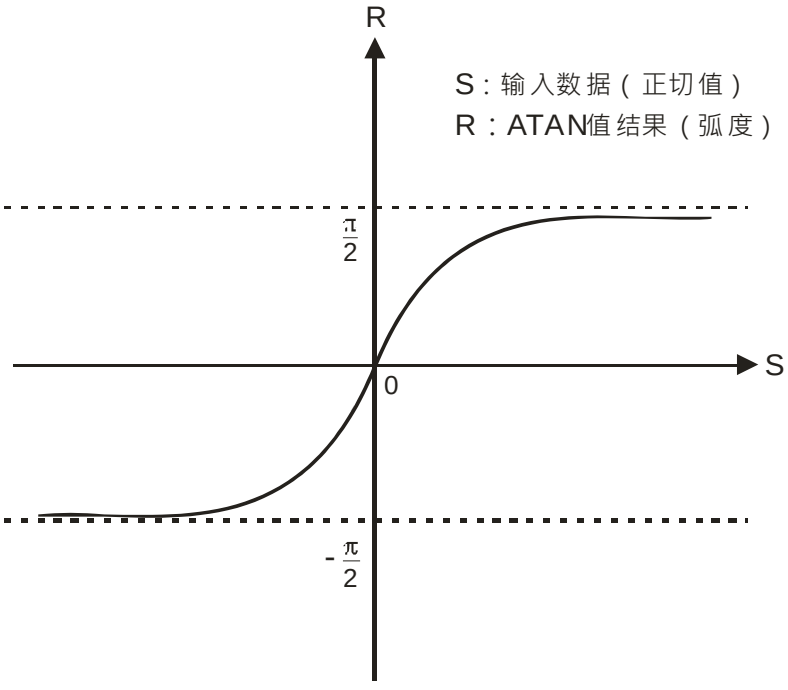
S ： 指定的来源值

D ： 取 ATAN 值结果

指令说明：

1. ATAN 值=$\tan^{-1}$

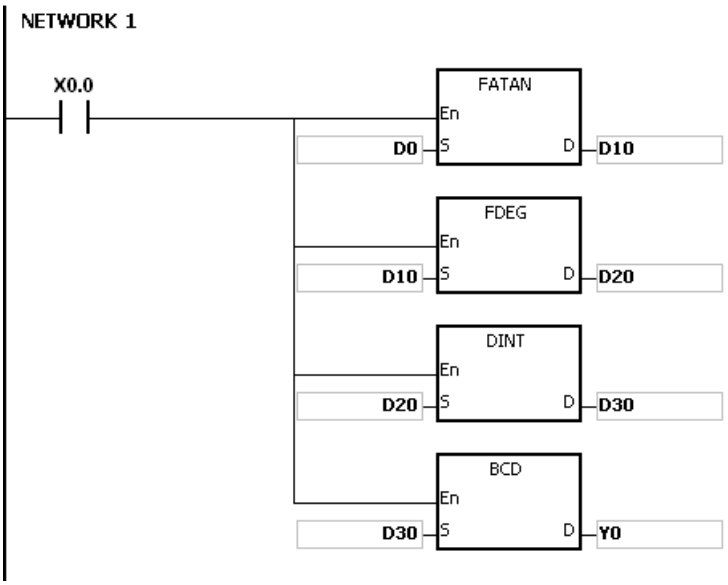
2. 下图显示输入数据与结果的关系：

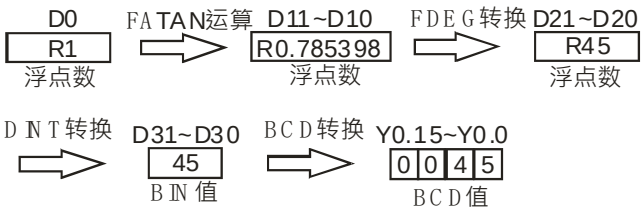


3. 若转换结果为 0，则零标号 SM600=ON。

程序范例：

当 X0.0=ON 时，指定浮点数（D1，D0）求取 ATAN 值后存于（D11，D10）当中，将（D11，D10）的结果做 FDEG 转为角度存于（D21，D20），再将（D21，D20）做 DINT 转换成整数存于（D31，D30），再透过 BCD 指令将求得角度输出于 Y0.15~Y0.0。





补充说明：

当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	指令码			操作数	功能
1506		FSINH	P	**S**，**D**	浮点数 SINH 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

S ： 指定的来源值

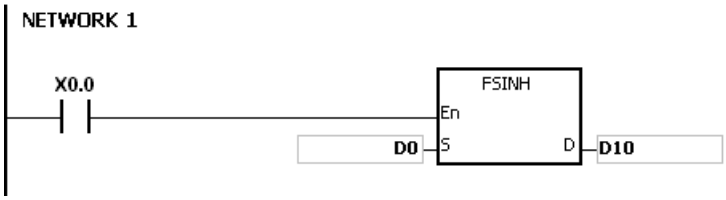
D ： 取 SINH 值结果

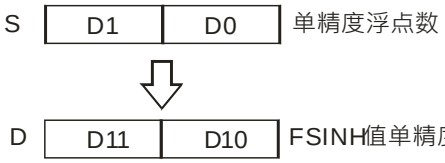
指令说明：

1. SINH 值=（e^{S}-e^{-S}）/2。
2. 若转换结果的绝对值大于可表示之最大浮点值，则 **D**=16#7F800000，进位标号 SM602=ON。
3. 若转换结果的绝对值小于可表示之最小浮点值，则 **D**=16#FF800000，借位标号 SM601=ON。
4. 若转换结果为 0，则零标号 SM600=ON。

程序范例：

1. 当 X0.0=ON 时，指定浮点数（D1，D0）求取 SINH 值后存于（D11，D10）当中，内容为浮点数。





2. 若转换结果的绝对值大于可表示之最大浮点值，则进位标号 SM602=ON。
3. 若转换结果的绝对值小于可表示之最小浮点值，则借位标号 SM601=ON。
4. 若转换结果为 0，则零标号 SM600=ON。

补充说明：

当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

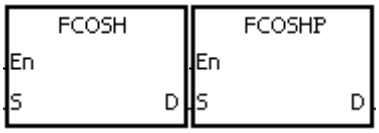
API	指令码			操作数	功能
1507		FCOSH	P	**S**，**D**	浮点数 COSH 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S ： 指定的来源值

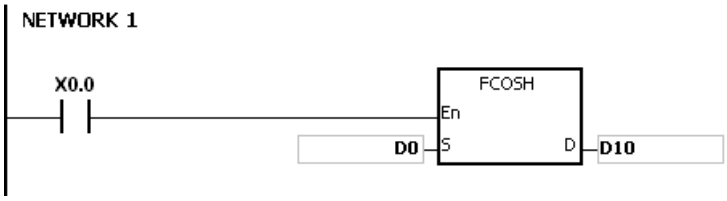
D ： 取 COSH 值结果

指令说明：

1. COSH 值=（$e^{s}+e^{-s}$）/2。
2. 若转换结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000，进位标号 SM602=ON。
3. 若转换结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000，借位标号 SM601=ON。
4. 若转换结果为 0，则零标号 SM600=ON。

程序范例：

1. 当 X0.0=ON 时，指定浮点数（D1，D0）求取 COSH 值后存于（D11，D10）当中，内容为浮点数。



2. 若转换结果的绝对值大于可表示之最大浮点值，则进位标号 SM602=ON。
3. 若转换结果的绝对值小于可表示之最小浮点值，则借位标号 SM601=ON。
4. 若转换结果为 0，则零标号 SM600=ON。

补充说明：

当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

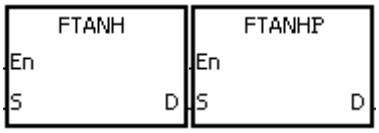
API	指令码			操作数	功能
1508		FTANH	P	**S**，**D**	浮点数 TANH 运算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S ： 指定的来源值

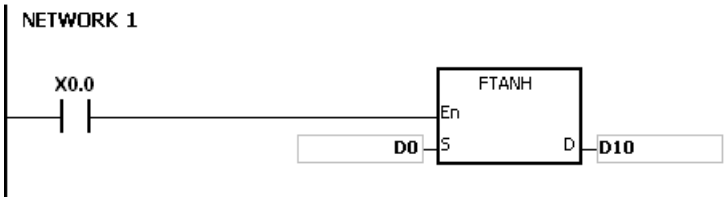
D ： 取 TANH 值结果

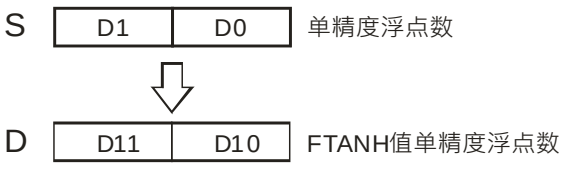
指令说明：

1. TANH 值=（e^{s}-e^{-s}）/（e^{s}+e^{-s}）。
2. 若转换结果为 0，则零标号 SM600=ON。

程序范例：

1. 当 X0.0=ON 时，指定浮点数（D1，D0）求取 TANH 值后存于（D11，D10）当中，内容为浮点数。





2. 若转换结果为 0，则零标号 SM600=ON。

补充说明：

当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	指令码			操作数	功能
1509		FRAD	P	**S**，**D**	角度→弧度

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

S ： 数据源（角度）

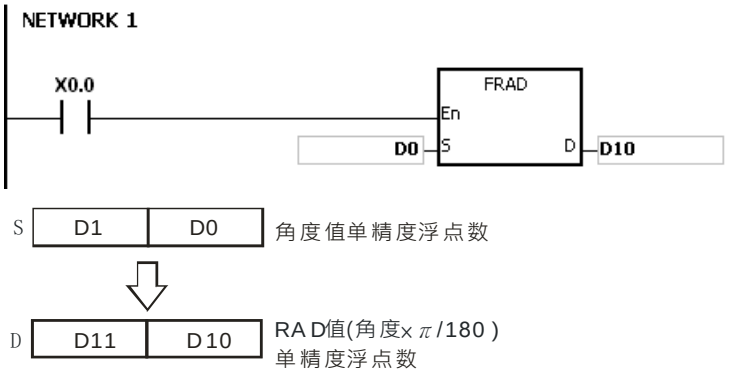
D ： 变换之结果（弧度）

指令说明：

1. 使用下列公式将角度转换成弧度。
2. 弧度 = 角度×（π/180）
3. 若转换结果为 0，则零标号 SM600=ON。

程序范例：

当 X0.0=ON 时，指定浮点数（D1，D0）之角度值，将角度转换成弧度值后存于（D11，D10）当中，内容为浮点数。



补充说明：

当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	指令码			操作数	功能
1510		FDEG	P	**S**，**D**	弧度→角度

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

S ： 数据源（弧度）

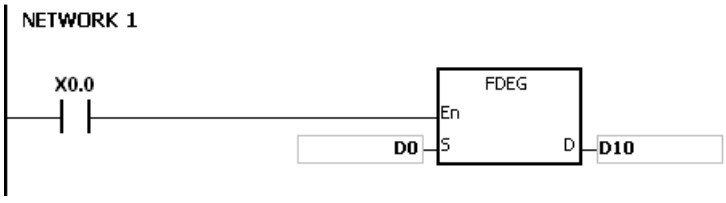
D ： 变换之结果（角度）

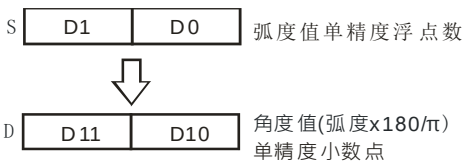
指令说明：

1. 使用下列公式将弧度转换成角度。
2. 角度 = 弧度×（180/π）
3. 若转换结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000，进位标号 SM602=ON。
4. 若转换结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000，借位标号 SM601=ON。
5. 若转换结果为 0，则零标号 SM600=ON。

程序范例：

当 X0.0=ON 时，指定浮点数（D1，D0）之角度值，将弧度值转换成角度后存于（D11，D10）当中，内容为浮点数。





补充说明：

当 **S** 的内容值超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	指令码			操作数	功能
1511	D	SQR	P	**S**，**D**	开平方根

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
		●	●			●	●				●	●	

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：

S ： 欲开平方根之来源装置

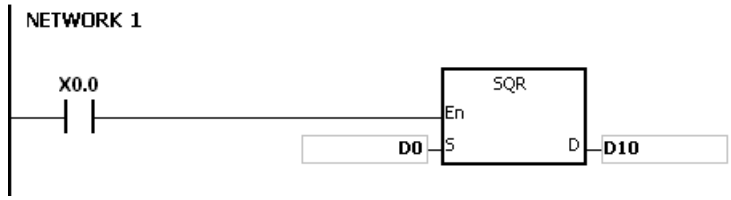
D ： 存放结果之装置

指令说明：

1. 将 **S** 所指定之装置内容值开平方根后，存放于 **D** 所指定之装置。
2. 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 SM601=ON。
3. 运算结果 **D** 为 0 时，零标志信号 SM600=ON。
4. 32 位指令才可使用 HC 装置，但不可使用 E 装置。

程序范例：

当 X0.0=ON 时，将 D0 内容值开平方根后，存放于 D10 内。



补充说明：

S 只可以指定正数，若指定负数时，将视为“指令运算错误”，指令不执行，SM0=ON，错误码 SR0=16#2003。

API	指令码			操作数	功能
1512		FSQR	P	**S**，**D**	浮点数开平方根

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

S ： 欲开平方根之来源装置

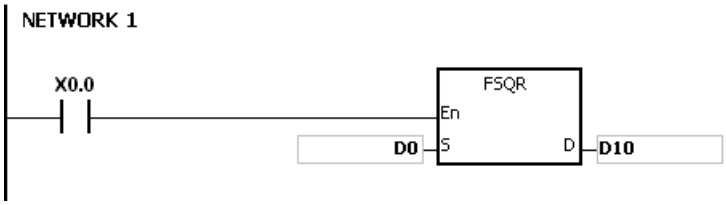
D ： 存放结果之装置

指令说明：

1. **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浮点数类型进行。
2. 若运算结果为 0，则零标号 SM600=ON。

程序范例一：

当 X0.0=ON 时，将浮点数（D1，D0）取开平方根，将结果存放至（D11，D10）所指定的寄存器当中。



补充说明：

S 只可以指定正数，若指定负数时，将视为“指令运算错误”，指令不执行，SM0=ON，错误码 SR0=16#2003。

API	指令码			操作数	功能
1513		FEXP	P	**S**・**D**	浮点数取指数

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

S ： 运算来源装置

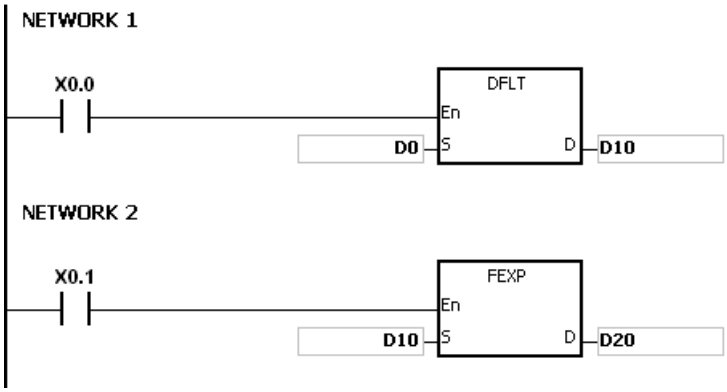
D ： 运算结果装置

指令说明：

1. 以 e=2.71828 为底数，**S** 为指数做 EXP 运算。
2. EXP[**D**+1，**D**]=[**S**+1，**S**]
3. **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** 需转换为浮点数。
4. **D** 操作数内容值=e^S；e=2.71828，**S** 为指定的源数据。
5. 若转换结果的绝对值大于可表示之最大浮点值，则 **D**=16#7F800000，进位标号 SM602=ON。
6. 若运算结果为 0，则零标号 SM600=ON。

程序范例：

1. 当 X0.0 为 ON 时，将（D1，D0）值转成浮点数存于（D11，D10）寄存器中。
2. 当 X0.1 为 ON 时，（D11，D10）为指数做 EXP 运算，其值为浮点数并存放于（D21，D20）寄存器中。



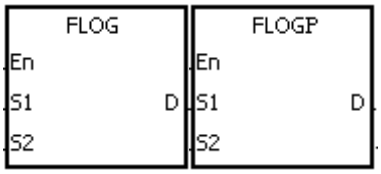
API	指令码			操作数	功能
1514		FLOG	P	**S_1 · S_2 · D**	浮点数取对数

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●		○					○
S_2	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S_1 ： 运算底数装置

S_2 ： 运算来源装置

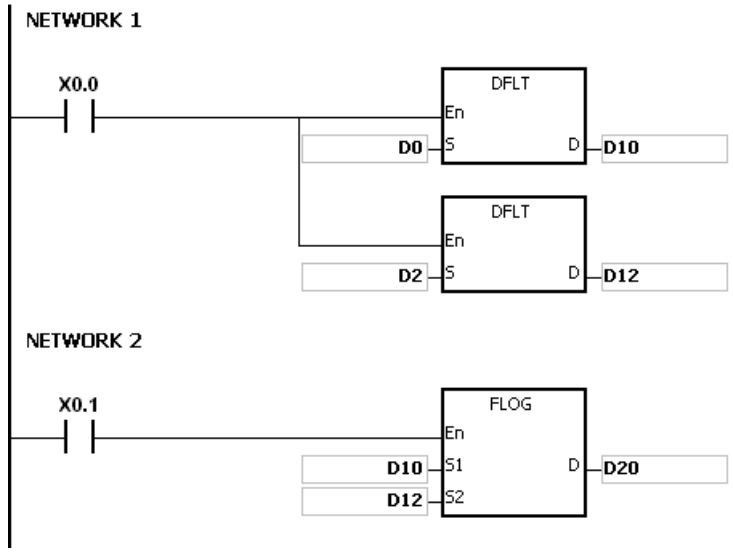
D ： 运算结果装置

指令说明：

1. 将 **S_1** 内容及 **S_2** 内容为操作数做 log 运算（单精度浮点数运算），结果存放于 **D**。
2. **S_1**、**S_2** 内容值只有正数有效。
3. **S_1^D= S_2**，求 D 值→log $_{S1}$ **S_2=D**
4. 例：已知 **S_1**=5，**S_2**=125，求 D=log $_5$ 125 = 3
5. 若运算结果为 0，则零标号 SM600=ON。

程序范例：

1. 当 X0.0 为 ON 时，将（D1，D0）内容及（D3，D2）内容转成浮点数分别存于（D11，D10）及（D13，D12）32 位寄存器中。
2. 当 X0.1 为 ON 时，将（D11，D10）及（D13，D12）32 位寄存器浮点数做 Log 运算并将结果存于（D21，D20）32 位寄存器中。



补充说明：

S_1 内容值<=1 或 **S_2**<=0 时，指令不执行，SM0=ON，错误码 SR0=16#2003。

API	指令码			操作数	功能
1515		FLN	P	**S**，**D**	浮点数取自然对数

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：

S ： 运算来源装置

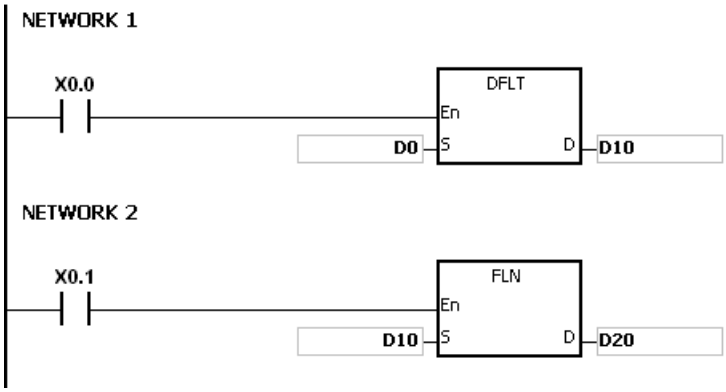
D ： 运算结果装置

指令说明：

1. 以 **S** 为操作数做自然对数 ln 运算（以单精度浮点数运算）。
2. **S** 内容值只有正数有效。
3. e^{D}=**S**→**D** 操作数内容值=ln **S**。
4. 若运算结果为 0，则零标号 SM600=ON。

程序范例：

1. 当 X0.0 为 ON 时，将（D1，D0）值转成浮点数存于（D11，D10）寄存器中。
2. 当 X0.1 为 ON 时，将（D11，D10）寄存器为真数做 ln 运算，其值为浮点数并存放于（D21，D20）寄存器中。



补充说明：

S<=0 时，指令不执行，SM0=ON，错误码 SR0=16#2003。

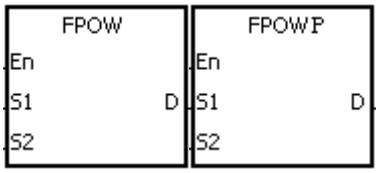
API	指令码			操作数	功能
1516		FPOW	P	**S_1 · S_2 · D**	浮点数权值指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●	●	●	●							○
S_2	●	●			●	●	●	●	●							○
D		●			●	●	●	●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S_1 ： 底数装置

S_2 ： 次幂数装置

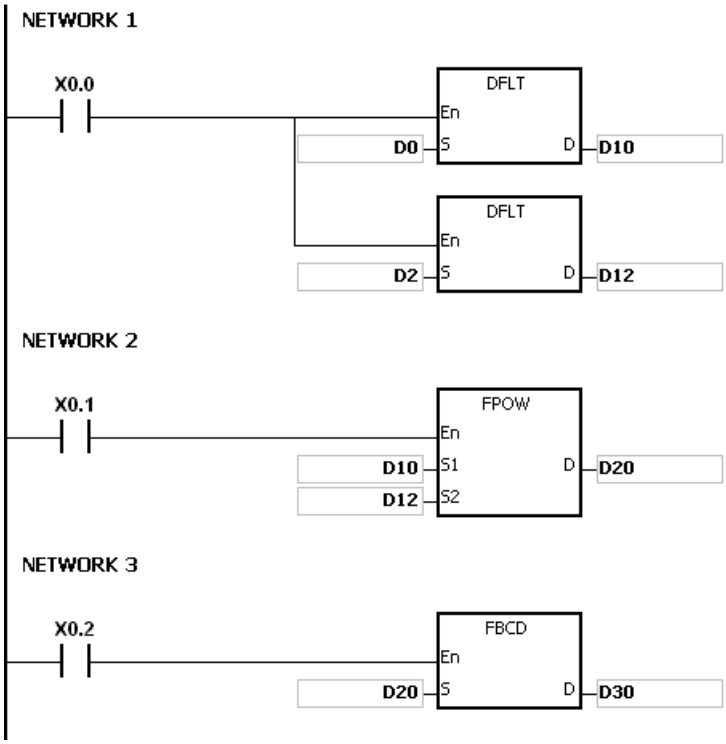
D ： 运算结果装置

指令说明：

1. 将单精度浮点数数据 **S_1** 以 **S_2** 的次幂数相乘后存放于 **D**。以单精度浮点数运算。
2. **D**=POW[**S_1**+1，**S_1**]^[**S_2**+1，**S_2**]
3. 主机韧体版本 V1.10.00 版（含）以下，**S_1** 内容值只有正数有效，**S_2** 内容值正负值都有效。
4. 主机韧体版本 V1.10.10 版（含）以上，**S_1** 内容值为正数，**S_2** 内容值正负值都有效；**S_1** 内容值为负数，**S_2** 内容值正负整数值才有效。
5. 例：S_1^{S2}=**D**，求 **D** 值?
   已知 **S_1**=5，**S_2**=3，则 D=5^3=125
6. 若运算结果的绝对值大于可表示之最大浮点值，则 D=16#7F800000，则进位标号 SM602=ON。
7. 若运算结果的绝对值小于可表示之最小浮点值，则 D=16#FF800000，则借位标号 SM601=ON。
8. 若运算结果为 0，则零标号 SM600=ON。

程序范例：

1. 当 X0.0 为 ON 时，将（D1，D0）内容及（D3，D2）内容转成浮点数分别存于（D11，D10）及（D13，D12）32 位寄存器中。
2. 当 X0.1 为 ON 时，将(D11，D10)及(D13，D12)32 位寄存器浮点数做 POW 运算并将结果存于(D21，D20）32 位寄存器中。
3. 当 X0.2 为 ON 时，将（D21，D20）32 位寄存器浮点数转成十进浮点数并存于（D31，D30）寄存器中。（此时 D31 为表示 D30 的 10 次幂方）



补充说明：

1. 主机韧体版本 V1.10.00 版（含）以下，S_1<0 时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 主机韧体版本 V1.10.10 版（含）以上，S_1<0 而且 S_2 为非整数时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. S_1==0 而且 S_2<=0 时，指令不执行，SM0=ON，错误码 SR0=16#2003。

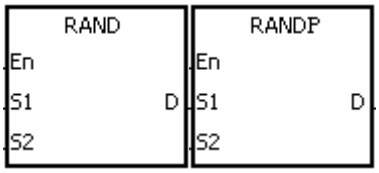
API	指令码			操作数	功能
1517		RAND	P	**S_1，S_2，D**	随机数值

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ：随机数生成的范围下限

S_2 ：随机数生成的范围上限

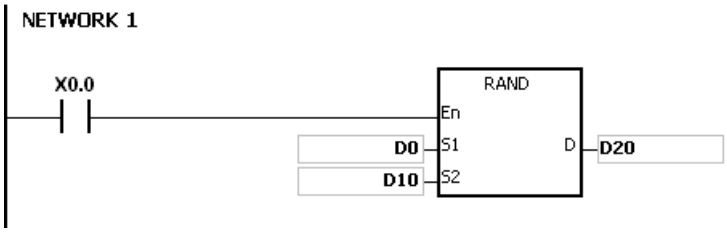
D ：随机数生成的结果

指令说明：

1. 产生介于范围下限 **S_1** 与范围上限 **S_2** 之随机数，将结果存放到 **D** 内。
2. 若用户输入 **S_1>S_2**，指令执行时把 **S_1** 作为上限值，**S_2** 下限值进行比较。

程序范例：

当 X0.0=ON 时，RAND 指令产生介于范围下限 D0 与范围上限 D10 之随机数，将结果存放到 D20 内。



补充说明：

S_1 或 **S_2** 操作数内容值范围为 0~32767，内容值超出范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

6.17 万年历指令

6.17.1 万年历指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1600	TRD	–	✓	万年历数据读出
1601	TWR	–	✓	万年历数据写入
1602	T+	–	✓	万年历数据加算
1603	T-	–	✓	万年历数据减算
1604	HOUR	–	–	运转定时器
1605	TCMP	–	✓	时间比较
1606	TZCP	–	✓	万年历数据区域比较
1607	DST	–	✓	日光节约时间设定
1608	WWON	–	–	每周工作时间控制开关

6.17.2 万年历指令说明

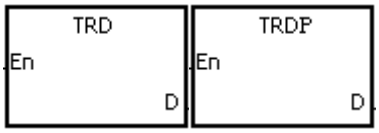
API	指令码		操作数	功能
1600	TRD	P	**D**	万年历数据读出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D					●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



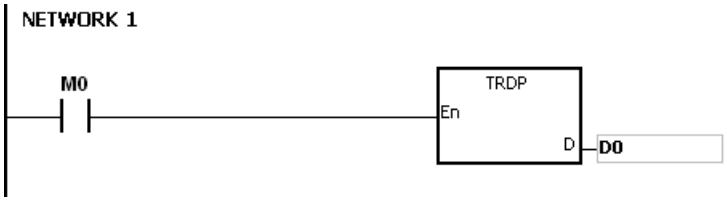
D ： 存放读取结果的装置

指令说明：

1. **D**：万年历当前时间读出后存放的装置。
2. **D** 操作数会占用连续 7 个装置。
3. 主机内建万年历时钟共提供年、星期、月、日、时、分、秒及共 7 组数据存放于 SR391~SR397 当中，TRD 指令的功能就是让程序设计者直接将万年历当前时间读出至指定的 7 个寄存器当中。
4. SR391 只读取公元年份的右 2 位。

程序范例：

当 M0=ON 时，将万年历当前时间读出至指定的 D0~D6 寄存器当中，SR397 之内容 1 表星期一、2 表星期二，类推，7 表星期日。



SR	项目	内容		一般 D	项目
SR391	年 (公元)	00~99	→	D0	年 (公元)
SR392	月	1~12	→	D1	月
SR393	日	1~31	→	D2	日
SR394	时	0~23	→	D3	时
SR395	分	0~59	→	D4	分
SR396	秒	0~59	→	D5	秒

SR	项目	内容
SR397	星期	1~7

→

一般 D	项目
D6	星期

补充说明：

1. **D**+6 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. SM220：±30 秒补正，ON 时作补正。(0~29 秒时归 0，30~59 秒时，分加 1、秒归 0)。
3. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [7] of WORD/INT。

API	指令码			操作数	功能
1601		TWR	P	**S**	万年历数据写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S					●	●		●	●							

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 数据源

指令说明：

1. **S**：存放欲写入万年历新设定值的装置。
2. **S** 操作数会占用连续 7 个装置（请以 24 小时制时间写入）。
3. 要调整主机内建万年历时钟的时候，可使用本指令将正确的当前时间写入至内藏万年历时钟当中。
4. 本指令被执行时，新的设定时间立刻被写入至 PLC 内部的万年历时钟当中，因此，执行本指令时，请注意所写入的新设定时间与写入当时的当前时间是否吻合。
5. 使用方式建议采用脉冲执行型。若条件接点采用常开方式（A 接点）一直启动写入动作，则 PLC 会自动限制只在第一次启动时写入，因此若下次需变更万年历时间，则需将条件接点关闭一个扫描周期后，重新启动指令才可再次更新时间

程序范例：

当 M0=ON 时，将正确的当前时间写入至内藏万年历时钟当中。

新的设定时间

一般 D	项目	内容		SR	项目
D20	年（公元）	00~99	→	SR391	年（公元）
D21	月	1~12	→	SR392	月
D22	日	1~31	→	SR393	日
D23	时	0~23	→	SR394	时
D24	分	0~59	→	SR395	分

万年历时钟

一般 D	项目	内容		SR	项目
D25	秒	0~59	→	SR396	秒
D26	星期	1~7	→	SR397	星期

补充说明：

1. 若 **S** 内容值超出范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 若 **S**+6 超出装置范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [7] of WORD/INT。

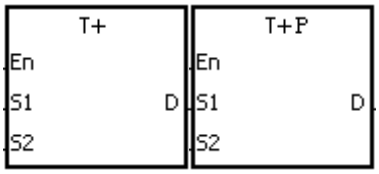
API	指令码			操作数	功能
1602		T+	P	**S_1 · S_2 · D**	万年历数据加算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1					●	●		●	●		○					
S_2					●	●		●	●		○					
D					●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 来源装置 1

S_2 ： 来源装置 2

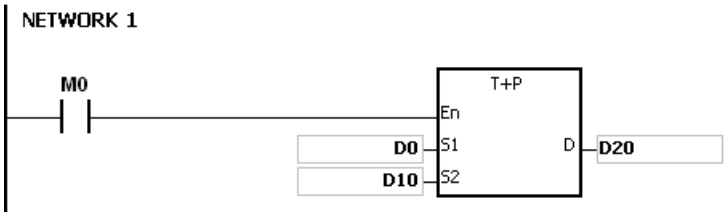
D ： 存放相加后结果的装置

指令说明：

1. 将 **S_1** 所指定的时、分、秒与 **S_2** 所指定的时、分、秒相加，所得到的结果存于指定 **D** 所指定的寄存器时、分、秒当中。
2. **S_1**、**S_2**、**D** 操作数，皆连续占用 3 个装置。
3. 加算结果若大于等于 24 小时的话，进位标志 SM602=ON、**D** 显示加算总值减掉 24 小时所得的结果。
4. 加算结果若是等于 0（0 时 0 分 0 秒），零标志 SM600=ON。

程序范例：

当 M0=ON 时，T+指令执行，将 D0~D2 所指定的万年历数据时、分、秒与 D10~D12 所指定的万年历数据时、分、秒相加，所得到的结果存于 D20~D22 所指定的寄存器中得到加总后之时、分、秒。



8时10分20秒　　6时40分6秒　　14时50分26秒

补充说明：

1. 若 S_1、S_2 内容值超出范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 若 S_1+2、S_2+2、**D**+2 超出装置范围限制，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. S_1 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
4. S_2 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
5. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。

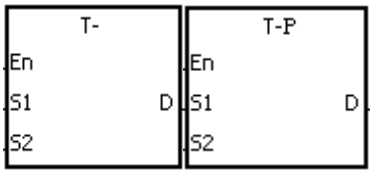
API	指令码			操作数	功能
1603		T-	P	**S_1**，**S_2**，**D**	万年历数据减算

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1					●	●		●	●		○					
S_2					●	●		●	●		○					
D					●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 来源装置 1

S_2 ： 来源装置 2

D ： 存放相加后结果的装置

指令说明：

1. 将 **S_1** 所指定的万年历数据时、分、秒减掉 **S_2** 所指定的万年历数据时、分、秒，所得到的结果存于指定 **D** 所指定的寄存器时、分、秒当中。
2. **S_1**、**S_2**、**D** 操作数，皆连续占用 3 个装置。
3. 减算结果若为负数时，借位标号 SM601=ON、该负数再加上 24 小时所得的结果显示 **D** 所指定的寄存器当中。
4. 减算结果若是等于 0 (0 时 0 分 0 秒)，零标志 SM600=ON。

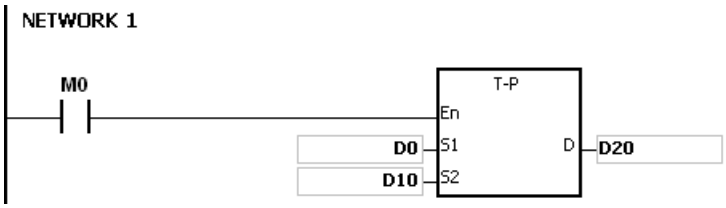
程序范例：

1. 当 M0=ON 时，T-指令执行，将 D0~D2 所指定的万年历数据时、分、秒与 D10~D12 所指定的万年历数据时、分、秒相减，所得到的结果存于指定 D20~D22 所指定的寄存器时、分、秒当中。

D0	20(時)		D10	14(時)		D20	5(時)
D1	20(分)	–	D11	30(分)	→	D21	49(分)
D2	5(秒)		D12	8(秒)		D22	57(秒)
20 時 20 分 5 秒			14 時 30 分 8 秒			5 時 49 分 57 秒		

2. 减算结果若为负数时，借位标号 SM601=ON。

D0	20(时)		D10	14(时)		D20	5(时)
D1	20(分)	-	D11	30(分)	→	D21	49(分)
D2	5(秒)		D12	8(秒)		D22	57(秒)
20时20分5秒			14时30分8秒			5时49分57秒		



补充说明：

1. 若 **S**$_1$、**S**$_2$ 内容值超出范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 若 **S**$_1$+2、**S**$_2$+2、**D**+2 超出装置范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **S**$_1$ 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
4. **S**$_2$ 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
5. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。

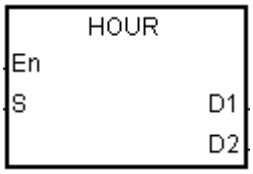
API	指令码	操作数	功能
1604	HOUR	**S**，**D_1**，**D_2**	运转定时器

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S					●	●		●	●		○	○	○	○		
D_1								●								
D_2		●	●	●				●		○						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D_1		●				●							
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	AS	

符号：



S ： 设定导通时间

D_1 ： 测量中的当前时间值

D_2 ： 输出装置

指令说明：

1. **S**：设定导通时间，单位：小时。**D_1**：测量中的当前时间值，单位：小时。**D_2**：输出装置。
2. **S** 为设定导通时间，单位：小时。

   16 位指令：1~32,767
3. HOUR：

   D_1 为测量中之当前时间值（0~32,767），单位：小时。

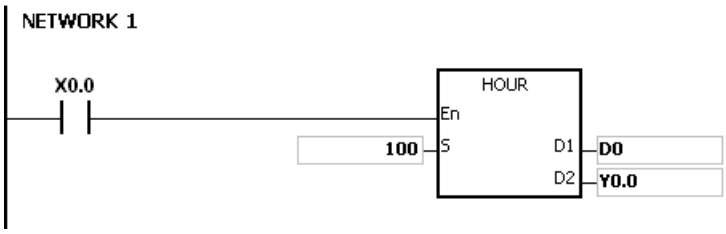
   D_1+1 为未满 1 个小时的当前时间值，设定范围 0~3,599，单位：秒。

   D_1+2 系统内部使用，指令执行时不可更改，否则会造成时间计算错误。

   测量中之当前时间值到达最大数值 32,767 小时、3,599 秒时会停止计时测量，要重新计时须将当前时间值 **D_1**、**D_1**+1 清除为 0。
4. 将输入接点导通时间做计时，当到达设定时间时（以小时为单位），会将输出装置导通，计数未到达时，输出装置不导通。可提供使用者管理机械的运作计时或维修。
5. 当输出装置导通后，定时器会继续计时。
6. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。

程序范例：

16 位指令当 X0.0=ON 时，开始计时，当到达 100 小时 Y0.0 导通，而 D0 会记录测量中之当前时间值（单位：小时），D1 会记录测量中不足 1 小时之当前时间值 0~3599（单位：秒），D2 程序保留使用，执行中不得更改内容值，否则会造成错误。



补充说明：

1. **S**<=0 时指令不执行，输出装置状态不变。
2. HOUR 中 $\mathbf{D}_1$<0 时指令不执行，输出装置状态不变。
3. HOUR 中 $\mathbf{D}_1$+2 超出装置范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. 16 位指令 $\mathbf{D}_1$ 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
5. 32 位指令 $\mathbf{D}_1$ 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of DWORD/DINT。

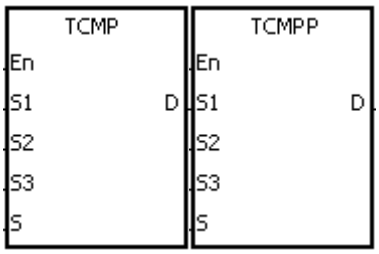
API	指令码			操作数	功能
1605		TCMP	P	**S_1 · S_2 · S_3 · S · D**	时间比较

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2	●	●			●	●		●	●		○	○	○	○		
S_3	●	●			●	●		●	●		○	○	○	○		
S					●	●		●								
D		●	●	●				●		○						

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 来源时间的"时"比较值

S_2 ： 来源时间的"分"比较值

S_3 ： 来源时间的"秒"比较值

S ： 当前时间

D ： 比较结果

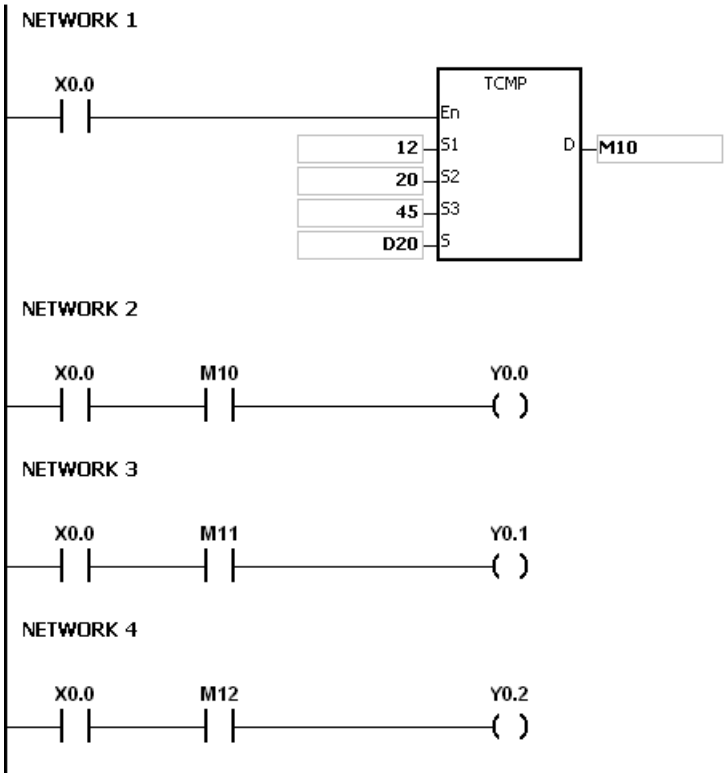
指令说明：

1. 将由 **S_1~S_3** 所指定的时、分、秒设定值与 **S** 起始之时、分、秒时间做比较，其比较结果设定在 **D**。
2. **S** 占用 3 个连续装置。，其 **S** 为"时"，数值范围为 0~23。**S**+1 为"分"，数值范围为 0~59。**S**+2 为"秒"，数值范围为 0~59。
3. 操作数 **D** 占用 3 个连续装置。**D**，**D**+1，**D**+2 用于储存比较结果。
4. **S** 所指定的当前时间，建议先使用 TRD 指令将内存万年历当前时间读出后，再使用 TCMP 指令进行比较。
5. 设定值（**S_1~S_3**）>**S** 当前值 **D**=ON，**D**+1=OFF，**D**+2=OFF。
6. 设定值（**S_1~S_3**）=**S** 当前值 **D**=OFF，**D**+1=ON，**D**+2=OFF。
7. 设定值（**S_1~S_3**）<**S** 当前值 **D**=OFF，**D**+1=OFF，**D**+2=ON。

程序范例：

1. 当 X0.0=ON 时，指令执行，将 D20~D22 万年历当前时间与设定值 12 时 20 分 45 秒做比较，将结果显示到 M10~M12。当 X0.0 由 ON→OFF 变化时，指令不被执行，但是 M10~M12 之前的 ON/OFF 状态仍被保持住。

2. 若需要得到≧、≦、≠之结果时，可将 M10~M12 串并联即可取得。



补充说明：

1. **S**+2 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **D**+2 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. 若 **S** 内容值超出范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. 若 $\mathbf{S}_1$~$\mathbf{S}_3$ 内容值超出范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
5. **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD。
6. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of BOOL。

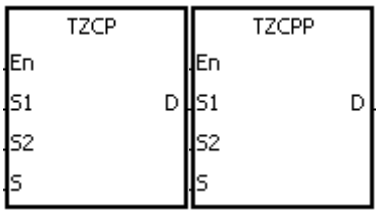
API	指令码			操作数	功能
1606		TZCP	P	**S_1 · S_2 · S · D**	万年历数据区域比较

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1					●	●		●	●							
S_2					●	●		●	●							
S					●	●		●								
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 设定比较时间的下限值

S_2 ： 设定比较时间的上限值

S ： 万年历当前时间

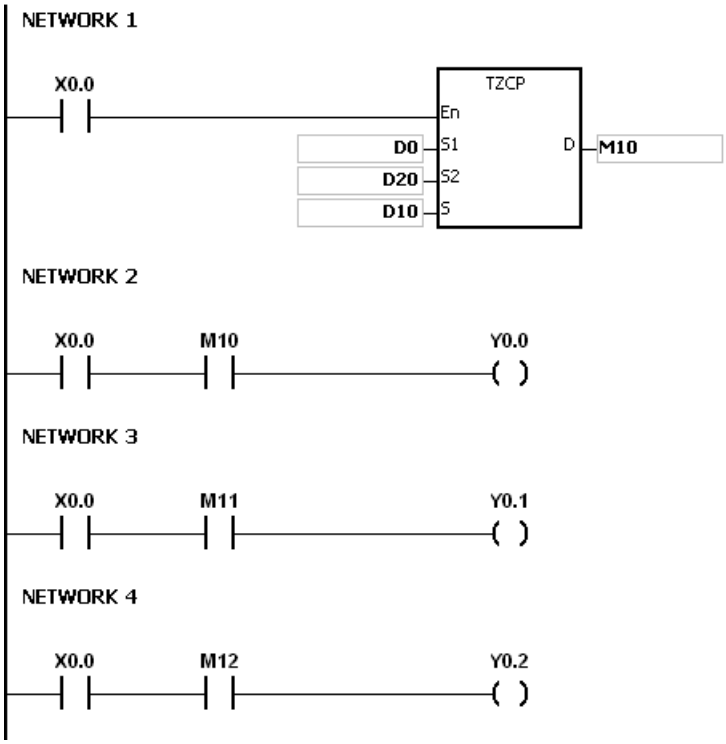
D ： 比较结果

指令说明：

1. 将由 **S** 所指定的万年历当前时间时、分、秒值与 **S_1** 所指定设定比较时间的下限值及 **S_2** 所指定设定比较时间的上限值做区域比较，其比较结果在 **D** 作表示。
2. **S_1**、**S_1**+1、**S_1**+2：设定比较时间下限值的"时"、"分"、"秒"。
3. **S_2**、**S_2**+1、**S_2**+2：设定比较时间上限值的"时"、"分"、"秒"。
4. **S**、**S**+1、**S**+2：为当前时间的"时"、"分"、"秒"。
5. 操作数 **S_1** 时间须比 **S_2** 时间小。当 **S_1**>**S_2**，指令执行时把 **S_1** 时间作为上/下限值进行比较。
6. **S** 所指定的当前时间，建议先使用 TRD 指令将内存万年历当前时间读出后，再使用 TZCP 指令进行比较。
7. 当当前时间 **S** 小于下限值 **S_1** 且 **S** 小于上限值 **S_2** 时，则 **D** 为 ON，当当前时间 **S** 大于下限值 **S_1** 且 **S** 大于上限值 **S_2** 时，则 **D**+2 为 ON，其余状态则 **D**+1 为 ON。

程序范例：

当 X0.0=ON 时，TZCP 指令执行，M10~M12 其中之一会 ON，当 X0.0=OFF 时，TZCP 指令不执行，M10~M12 状态保持在 X0.0=OFF 之前的状态。



补充说明：

1. **S_1**+2、**S_2**+2、**S**+2、**D**+2 超出装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **S_1**、**S_2**、**S** 内容值超出范围，则视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. **S_1** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
4. **S_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
5. **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
6. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of BOOL。

API	指令码			操作数	功能
1607		DST	P	**S**、**S_1**、**S_2**、**S_3**、**S_4**、**S_5**、**D**	日光节约时间

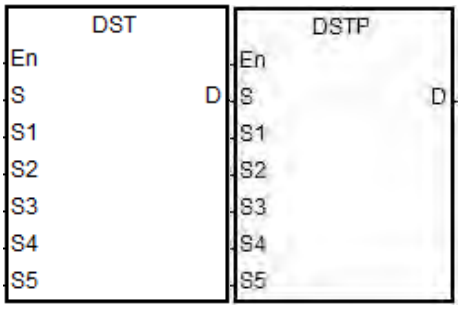
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S					●	●		●					○	○		
S_1					●	●		●					○	○		
S_2					●	●		●					○	○		
S_3					●	●		●					○	○		
S_4					●	●		●					○	○		
S_5					●	●		●					○	○		
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
S_5		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

6

符号：



S ： 日光节约时间功能代码

S_1 ： 节约开始时间的月份

S_2 ： 节约开始时间的日期

S_3 ： 节约结束时间的月份

S_4 ： 节约结束时间的日期

S_5 ： 节约时间（分钟）

D ： 节约功能状态

指令说明：

1. DST 指令中各操作数的说明如下：

   S：日光节约时间功能代码

   V1.04 前（不含）支持的功能代码：

功能代码	功能
0	关闭日光节约时间
1	启动模式 1 日光节约时间

功能代码	功能
2	读取日光节约时间
3~	保留，自动视为读取日光节约时间

V1.04 后（含）支持的功能代码：

功能代码	功能
0	关闭日光节约时间
1	启动模式 1 日光节约时间
2	读取日光节约时间
3	启动模式 2 日光节约时间
4	系统设定，表示目前节约功能为关闭
5	系统设定，表示目前节约功能为启动模式 1
7	系统设定，表示目前节约功能为启动模式 2
6, 8~	保留，自动视为读取日光节约时间

注 1：当 **S**=4、5、7 时，此指令不会执行任何动作

注 2：各功能代码的使用方式后续会有详细说明

S_1：设定日光节约时间（Daylight Saving Time）的开始“月份”

S_2：设定日光节约时间（D.S.T.）的开始日期，会根据 **S**（启动模式）的值而有所不同。

设定 **S**=1（启动模式 1）时，**S_2** 设定节约开始之日期为“日”

设定 **S**=3（启动模式 2）时，**S_2** 设定节约开始之日期为“第 **S_2** 个星期 **S_2+1**”

S_3：设定日光节约时间（Daylight Saving Time）的结束“月份”

S_4：设定日光节约时间（D.S.T. ）的结束日期，会根据 **S**（启动模式）的值而有所不同。

设定 **S**=1（启动模式 1）时，**S_4** 设定节约结束的日期为“日”

设定 **S**=3（启动模式 2）时，**S_4** 设定节约结束的日期为“第 **S_4** 个星期 **S_4+1**”

S_5：设定节约时间，其单位为“分钟”。

D：储存指令执行后之状态，OFF 表示功能关闭，ON 表示功能启动。

2. DST 指令支持 D.S.T（日光节约时间）的功能，其使用说明如下：

D.S.T 功能	**S** 功能代码	说明
关闭	0	关闭日光节约时间
启动	1、3	启动日光节约时间
读取	2	读取日光节约时间

- **关闭日光节约时间功能：**(参考范例一)

  设定 **S** 操作数为 0 表"关闭"功能，其余参数 **S**$_1$~**S**$_5$ 操作数都为无意义，可输入任意值，并将目前的节约状态输出 **D** 操作数为 OFF。

- **启动日光节约时间功能：**(参考范例二、三)

  设定 **S** 操作数为 1 或 3 (决定启动模式) 表"启动"功能，其中 **S**$_1$、**S**$_2$ 操作数为开始的日期，其中 **S**$_3$、**S**$_4$ 操作数为结束的日期，**S**$_5$ 操作数为节约时间，并将目前的节约状态输出 **D** 操作数。

  当日光节约时间为"启动"功能时，并且系统时间之日期第一次进入开始日期 (**S**$_1$、**S**$_2$) 时，其系统内存时间将自动加上节约时间 **S**$_5$ 一次；反之，系统时间之日期第一次进入结束日期 (**S**$_3$、**S**$_4$) 时，其系统内存时间将自动减去节约时间 **S**$_5$ 一次。

  启动日光节约时间的功能有下面几种：

启动模式	**S** 功能代码	比对条件	支持机种版本
模式 1	1	"月"、"日"	V1.00 (含) 以上
模式 2	3	"月"、"星期"	V1.04 (含) 以上

启动模式 1 (**S**=1) : 依"月"、"日"做启动，各操作数设定如下 : (参考范例二)

操作数	说明
S$_1$	节约开始时间之月份 限制输入数值范围 1~12
S$_2$	节约开始时间之日期"日" 限制输入数值范围 1~31
S$_3$	节约结束时间之月份 限制输入数值范围 1~12
S$_4$	节约结束时间之日期"日" 限制输入数值范围 1~31
S$_5$	节约时间 (分钟) 限制输入数值范围 1~1439 (表限制 1 天内)

注 1：若启动成功，目前的节约状态输出 **D** 操作数为 ON。

注 2：若是设定日期错误 (如 : 4 月 31 日)，或者开始日期大于等于结束日期 (如 : 开始 10 月 1 日、结束 4 月 1 日) ，则"启动"功能将无法成功设定，且会设定 SM0=ON，错误码 SR0=16#200B。

注 3：若 **S**$_5$ 超过限制输入数值，则"启动"功能将无法成功设定，且会设定 SM0=ON，错误码 SR0=16#200B。

启动模式 2 (**S**=3) : 依"星期"做启动，各操作数设定如下 : (参考范例三)

操作数	说明
S$_1$	节约开始时间之月份 限制输入数值范围 1~12

操作数	说明
S_2~S_{2+1}	节约开始时间之日期“星期”（第 **S_2** 个星期 **S_2+1**） **S_2** 限制输入数值范围 1~4 或-1 ~ -4 **S_{2+1}** 限制输入数值范围 1~7
S_3	节约结束时间之月份 限制输入数值范围 1~12
S_4~S_{4+1}	节约结束时间之日期“星期”（第 **S_4** 个星期 **S_4+1**） **S_4** 限制输入数值范围 1~4 或-1 ~ -4 **S_{4+1}** 限制输入数值范围 1~7（星期一~星期日）
S_5	节约时间（分钟）

注 1：若启动成功，目前的节约状态输出 **D** 操作数为 ON。

注 2：S_2 与 S_4 限制输入数值为 1~4 或-1 ~ -4。其中-1 表示当月份倒数第一个星期 **S_{2+1}** 与 **S_{4+1}**，-2 表示倒数第 2 个星期 **S_{2+1}** 与 **S_{4+1}**，举例：倒数第 1 个周日是 28 号，那倒数第 2 个周日就是 21 号。若输入数值超出限制范围，则“启动”功能将无法成功设定，且会设定 SM0=ON，错误码 SR0=16#200B。

注 4：若 S_2+1 与 S_4+1 输入数值超出限制范围，皆预设为星期日。

注 5：若 **S_5** 超过限制输入数值，则“启动”功能将无法成功设定，且会设定 SM0=ON，错误码 SR0=16#200B。

注 6：若操作数 S_2 与 S_4 选择装置为 K 或 16#组件，则无法被储存，且会被设定 SM0=ON，错误码 SR0=16#2003。

- **读取日光节约时间功能：**（参考范例一~三）

设定 **S** 操作数为 2 表“读取”功能，其中 **S_1**、**S_2** 操作数将储存开始之日期，**S_3**、**S_4** 操作数将储存结束之日期，**S_5** 操作数将储存节约时间，并将目前的节约状态输出 **D** 操作数。若设定为“读取”后，操作数 **S**, **S_1**~**S_5** 选择装置为 K 或 16#组件，则无法被储存，且会被设定 SM0=ON，错误码 SR0=16#2003。V1.04（含）后，在读取动作完成时，会自动将 DST 现有设定之 **S** 功能码加 4 显示，之后不再执行任何动作。举例：0、1、3 这三种功能码，在被读取完成后将分别显示 4、5、7。

如果读出目前 DST 状态为”关闭”，**S_1**~**S_5** **及 D** 各操作数储存的数据如下：

操作数	说明
S	V1.04（不含）前，功能代码固定 2 V1.04（含）后，系统自动修正功能代码为 4，表示目前节约状态为关闭
S_1~S_5	无效操作数，不用理会
D	节约状态输出 OFF（关闭）

如果读出目前 DST 状态为"启动模式 1"，**S_1~S_5 及 D** 各操作数储存的数据如下：

操作数	说明
S	V1.04（不含）前，功能代码固定 2 V1.04（含）后，系统自动修正功能代码为 5，表示目前节约状态为启动模式 1
S_1	节约开始时间的月份
S_2	节约开始时间的日期"日"
S_3	节约结束时间的月份
S_4	节约结束时间的日期"日"
S_5	节约时间（分钟）
D	节约状态输出 ON（启动）

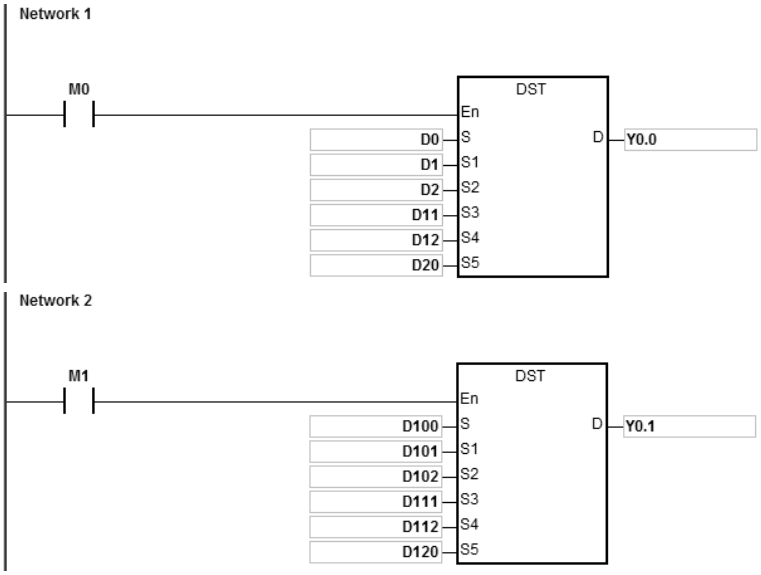
如果读出目前 DST 状态为"启动模式 2"，**S_1~S_5 及 D** 各操作数储存的数据如下：

操作数	说明
S	系统自动修正功能代码为 7，表示目前节约状态为启动模式 2
S_1	节约开始时间的月份
S_2~S_{2+1}	节约开始时间的日期"星期"（第 **S_2** 个星期 **S_2+1**）
S_3	节约结束时间的月份
S_4~S_{4+1}	节约结束时间的日期"星期"（第 **S_4** 个星期 **S_4+1**）
S_5	节约时间（分钟）
D	节约状态输出 ON（启动）

3. 此指令仅做为日光节约功能的开启与关闭，因此指令前面的条件接点被启动或关闭，都不会重复设定日光节约功能的时间（参考范例二中重新启动条件接点 M0 OFF=>ON 的部分）。若功能为已启动状态，并且要修改设定开始与结束时间，则可直接重新执行指令的启动功能即可，不需要先关闭日光节约功能再重新启动日光节约功能。
4. 若设定日光节约时间日期为 4 月 1 日~9 月 1 日，节约 60 分钟，则万年历的时间变化如下：

关闭启动日光节约功能（默认）	启动日光节约功能
3 月 01 日 3 时	3 月 01 日 3 时
3 月 31 日 3 时	3 月 31 日 3 时
4 月 01 日 3 时	4 月 01 日 4 时
5 月 01 日 3 时	5 月 01 日 4 时
6 月 01 日 3 时	6 月 01 日 4 时
7 月 01 日 3 时	7 月 01 日 4 时
8 月 01 日 3 时	8 月 01 日 4 时
8 月 31 日 3 时	8 月 31 日 4 时
9 月 01 日 3 时	9 月 01 日 3 时

范例一、关闭 D.S.T.功能，并读取目前 D.S.T 的状态



给定下面设定值：

装置	内容值	说明
D0	0	关闭 D.S.T.功能
D1	X	无意义操作数，可输入任意值
D2	X	无意义操作数，可输入任意值
D11	X	无意义操作数，可输入任意值
D12	X	无意义操作数，可输入任意值
D20	X	无意义操作数，可输入任意值

启动条件接点 M0

监控 Y0.0=OFF 表示节约功能关闭中

给定 D100=K2， 表示要读取 DST 目前的节约状态

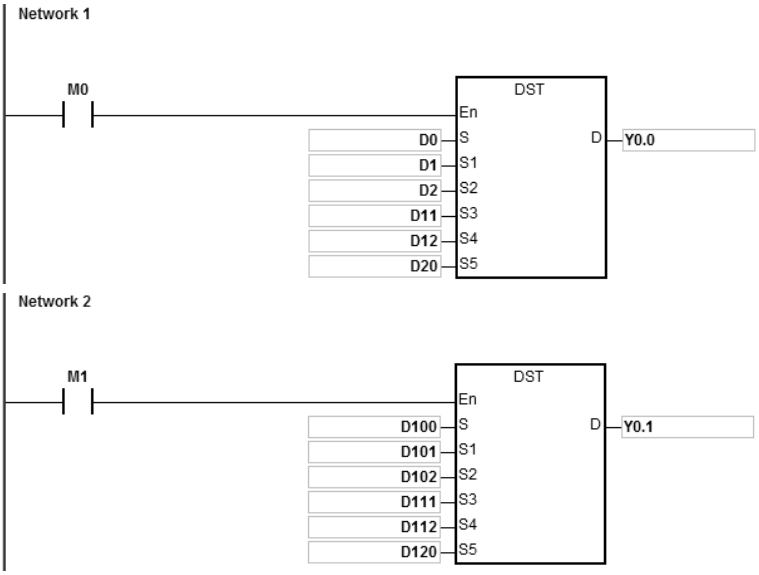
启动条件接点 M1

读出各装置对应的内容值如下：

装置	内容值	说明
D100	2	V1.04（不含）前，此内容值保持不变为 2，表读取功能
	4	V1.04（含）后，系统自行变更为 4，表示已经关闭 D.S.T 功能
D101	X	无意义操作数，不用理会
D102	X	无意义操作数，不用理会
D111	X	无意义操作数，不用理会
D112	X	无意义操作数，不用理会
D120	X	无意义操作数，不用理会
Y0.1	OFF	表示节能状态为关闭

范例二、启动模式 1 开启 D.S.T.功能，并读取目前 D.S.T 的状态

设定日光节约时间为 4 月 1 日开始，9 月 3 日结束（这天会恢复），且节约时间为 60 分钟



给定下面设定值：

装置	内容值	说明
D0	1	启动模式 1 开启 D.S.T.功能
D1	4	起始月份：4 月
D2	1	起始日期：1 日
D11	9	结束月份：9 月
D12	3	结束日期：3 日
D20	60	单位节约时间为 60 分钟

启动条件接点 M0

监控 Y0.0=ON 表示节约功能启动中

也就是，当 PLC 的万年历时间执行到 4 月 1 日的 0 时 0 分 0 秒时会自动增加 60 分钟，但到 9 月 3 日的 0 时 0 分 0 秒时会自动恢复为未节约的时间。

给定 D100=K2， 表示要读取 DST 目前的节约状态

启动条件接点 M1

读出各装置对应的内容值如下：

装置	内容值	说明
D100	2	V1.04（不含）前，此内容值保持不变为 2，表读取功能
	5	V1.04（含）后，系统自行变更为 5，表示启动模式 1 开启 D.S.T.功能
D101	4	起始月份：4 月
D102	1	起始日期：1 日
D111	9	结束月份：9 月
D112	3	结束日期：3 日

装置	内容值	说明
D120	60	单位节约时间为 60 分钟
Y0.1	ON	表示节能状态为 ON

读出日光节约时间的设定，可以使用 DST 指令的方式读出，也可以在 ISPSoft 的 HWCONFIG 设定中读出。

若重新启动条件接点 M0 OFF=>ON，不会因为重复执行条件接点就重复设定日光节约的时间

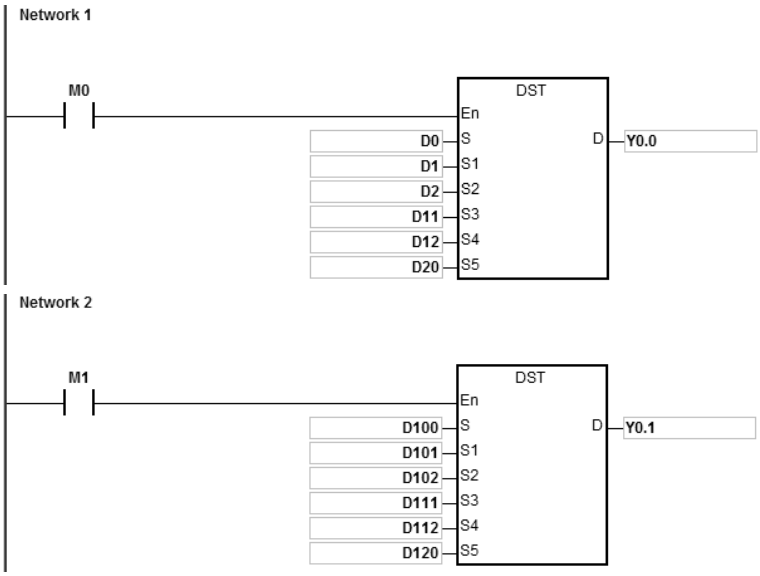
读出各装置对应的内容值如下：

装置	内容值	说明
D100	2	V1.04（不含）前，此内容值保持不变为 2，表读取功能
	5	V1.04（含）后，系统自行变更为 5，表示启动模式 1 开启 D.S.T.功能
D101	4	起始月份：4 月
D102	1	起始日期：1 日
D111	9	结束月份：9 月
D112	3	结束日期：3 日
D120	60	单位节约时间为 60 分钟
Y0.1	ON	表示节能状态为 ON

如果 4 月 1 日 3 时节约后为 4 月 1 日 4 时，此时不管重复几次条件接点 M0 OFF=>ON，则节约后还是 4 月 1 日 4 时，并不会因为重复执行条件接点就重复设定日光节约的时间。

范例三、启动模式 2 开启 D.S.T.功能

设定日光节约时间为 5 月的第 2 个星期三开始，9 月的第 3 个星期五结束（这天会恢复），且节约时间为 60 分钟。



给定下面设定值：

装置	内容值	说明
D0	3	启动模式 2 开启 D.S.T.功能
D1	5	起始月份：5 月
D2	2	起始日期：第 2 个星期
D3	3	起始日期：星期三
D11	9	结束月份：9 月
D12	3	结束日期：第 3 个星期
D13	5	结束日期：星期五
D20	60	单位节约时间为 60 分钟

启动条件接点 M0

监控 Y0.0=ON 表示节约功能启动中

如果是在 2017 年时执行这个命令，则 5 月的第 2 个星期三就是 5 月 10 日，9 月的第 3 个星期五就是 9 月 15 日；也就是，当 PLC 的万年历时间执行到 5 月 10 日的 0 时 0 分 0 秒时会自动增加 60 分钟，但到 9 月 15 日的 0 时 0 分 0 秒时会自动恢复为未节约的时间。

给定 D100=K2， 表示要读取 DST 目前的节约状态

启动条件接点 M1

读出各装置对应的内容值如下：

装置	内容值	说明
D100	2	V1.04（不含）前，此内容值保持不变为 2，表读取功能
	7	V1.04（含）后，系统自行变更为 7，表示启动模式 2 开启 D.S.T.功能
D101	5	起始月份：5 月
D102	2	起始日期：第 2 个星期
D103	3	起始日期：星期三
D111	9	结束月份：9 月
D112	3	结束日期：第 3 个星期
D113	5	结束日期：星期五
D120	60	单位节约时间为 60 分钟
Y0.1	ON	表示节能状态为 ON

读出日光节约时间的设定，可以使用 DST 指令的方式读出，也可以在 ISPSoft 的 HWCONFIG 设定中读出，其中 HWCONFIG 的呈现是系统自动把“星期”转换为“月”“日”的结果。

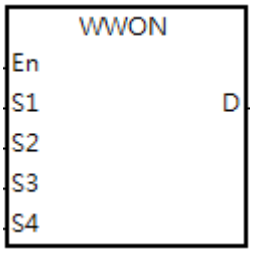
API	指令码	操作数	功能
1608	WWON	S_1，S_2，D	每周工作时间控制开关

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								○								
S_2								○								
S_3								○								
S_4								○								
D		○	○	○												

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 当周工作开始时间的小时（连续占用 7 个）

S_2 ： 当周工作开始时间的分钟（连续占用 7 个）

S_3 ： 当周工作停止时间的小时（连续占用 7 个）

S_4 ： 当周工作停止时间的分钟（连续占用 7 个）

D ： 输出控制开关输出

指令说明：

1. **S_1~ S_1+6** 分别为周日、周一、…至周六的工作开始时间的小时设定值。此参数将连续占用 7 个装置组件，使用 ISPSoft 软件编写 PLC 程序时，可采用矩阵宣告方式的变量。
2. **S_2~ S_2+6** 分别为周日、周一、…至周六的工作开始时间的分钟设定值。此参数将连续占用 7 个装置组件，使用 ISPSoft 软件编写 PLC 程序时，可采用矩阵宣告方式的变量。
3. **S_3~ S_3+6** 分别为周日、周一、…至周六的工作停止时间的小时设定值。此参数将连续占用 7 个装置组件，使用 ISPSoft 软件编写 PLC 程序时，可采用矩阵宣告方式的变量。
4. **S_4~ S_4+6** 分别为周日、周一、…至周六的工作停止时间的分钟设定值。此参数将连续占用 7 个装置组件，使用 ISPSoft 软件编写 PLC 程序时，可采用矩阵宣告方式的变量。
5. **当 S_1** 小时设定值**大于 S_3** 小时设定值时，即表示停止工作时间为隔日的小时，举例：周一工作开始小时设为 18 时，工作停止小时设为 6 时，则此时停止的小时即是指周二的清晨 6 时。

	开始时间	结束时间

	开始	小时		开始	分钟		停止	小时		停止	分钟
周日	**S_1**	24		**S_2**	00		**S_3**	24		**S_4**	00
周一	**S_1**+1	18		**S_2**+1	00		**S_3**+1	06		**S_4**+1	00

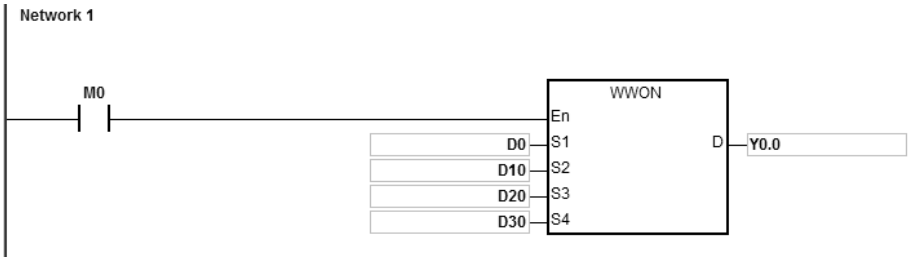
6. 小时设定的数值范围为 0~23，当小时设定值超过此范围时，即表示此时间比较功能不启动；此外，分钟设定的数值范围为 0~59，当分钟设定值超过此范围时，将以 0 为默认值。
7. 当工作开始时间超过一天以上时，则可在小时设定值上输入数值 24，此时开始与停止的比对工作时间将会无效；举例：周一 8 时开始工作，直到周二 20 时停止工作，则设定 **S_1**+1 的小时为 8，**S_3**+1 的小时为 24，+2 的小时为 24，+2 的小时为 20。其工作时间以表格方式呈现如下：

	开始	小时		开始	分钟		停止	小时		停止	分钟
周日	**S_1**	24		**S_2**	00		**S_3**	24		**S_4**	00
周一	**S_1**+1	08		**S_2**+1	00		**S_3**+1	24		**S_4**+1	00
周二	**S_1**+2	24		**S_2**+2	00		**S_3**+2	20		**S_4**+2	00

8. 此指令必须搭配万年历功能执行，在 PLC 开始运行之前，请确认电池是否已经安装。
9. 此指令无使用次数限制，但须注意输出控制开关 **D** 不可重复使用，若重复使用，就以最后执行 WWON 指令控制输出结果。
10. 若是当天有多个工作时段须安排时，请自行规划多个 WWON 指令，但注意输出控制开关 **D** 不可选择重复。

程序范例一：

周一~周五工作时间为 08 时开始，18 时停止，周六与周日都停止。



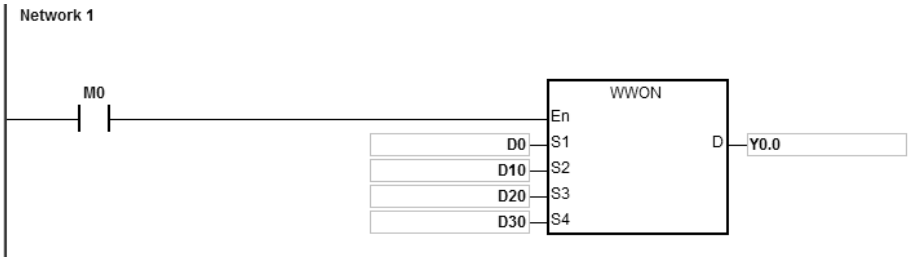
给定个装置默认值如下:

	开始时间						结束时间				
星期	开始	小时		开始	分钟		停止	小时		停止	分钟
周日	D0	24		D10	00		D20	24		D30	00
周一	D1	08		D11	00		D21	18		D31	00
周二	D2	08		D12	00		D22	18		D32	00
周三	D3	08		D13	00		D23	18		D33	00
周四	D4	08		D14	00		D24	18		D34	00
周五	D5	08		D15	00		D25	18		D35	00
周六	D6	24		D16	00		D26	24		D36	00

启动条件接点 M0 后,周一~周五的 08:00~18:00 时 Y0.0=ON, 其它时间 Y0.0 OFF。

程序范例二：

周日休息；周一为 18 时开始，周二 08 时停止，周二 18 时开始，周三 08 时停止，以此顺序到周六早上 08 时停止。



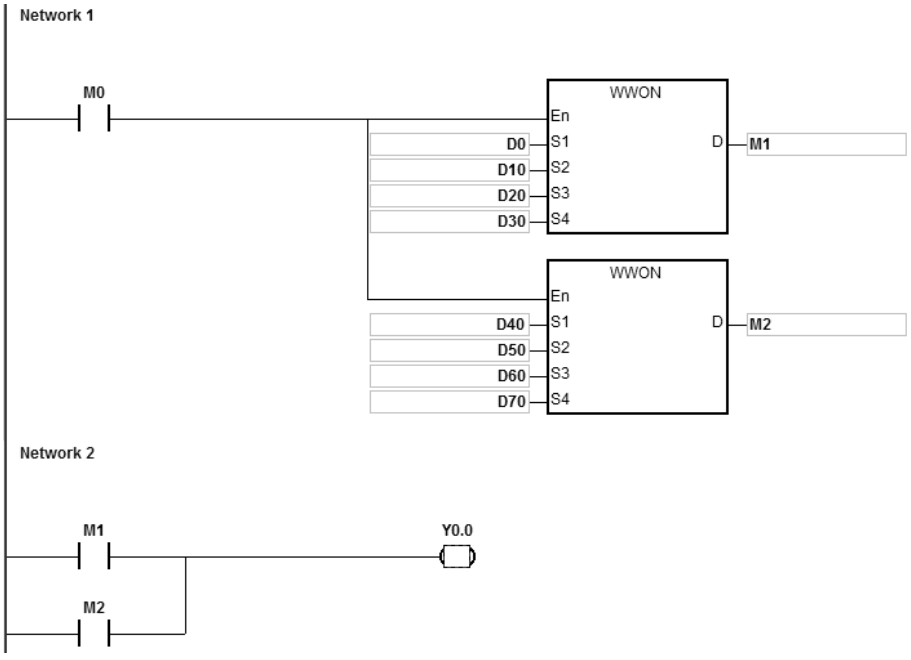
给定个装置默认值如下:

	开始时间						结束时间				
星期	开始	小时		开始	分钟		停止	小时		停止	分钟
周日	D0	24		D10	00		D20	24		D30	00
周一	D1	18		D11	00		D21	08		D31	00
周二	D2	18		D12	00		D22	08		D32	00
周三	D3	18		D13	00		D23	08		D33	00
周四	D4	18		D14	00		D24	08		D34	00
周五	D5	18		D15	00		D25	08		D35	00
周六	D6	24		D16	00		D26	24		D36	00

启动条件接点 M0 后,周一~周五的 18:00 开始~隔天早上的 08:00 时 Y0.0=ON, 其它时间 Y0.0 OFF。

程序范例三：

排定工作时间周一~周五工作时间为早上 08:00~12:00 跟下午 2:00~5:30，周六与周日都停止。



将白天时间给定个装置默认值如下:

	开始时间						结束时间				
星期	开始	小时		开始	分钟		停止	小时		停止	分钟
周日	D0	24		D10	00		D20	24		D30	00
周一	D1	08		D11	00		D21	12		D31	00
周二	D2	08		D12	00		D22	12		D32	00
周三	D3	08		D13	00		D23	12		D33	00
周四	D4	08		D14	00		D24	12		D34	00
周五	D5	08		D15	00		D25	12		D35	00
周六	D6	24		D16	00		D26	24		D36	00

将下午时间给定装置默认值如下:

	开始时间						结束时间				
星期	开始	小时		开始	分钟		停止	小时		停止	分钟
周日	D40	24		D50	00		D60	24		D70	00
周一	D41	14		D51	00		D61	17		D71	30
周二	D42	14		D52	00		D62	17		D72	30
周三	D43	14		D53	00		D63	17		D73	30
周四	D44	14		D54	00		D64	17		D74	30
周五	D45	14		D55	00		D65	17		D75	30
周六	D46	24		D56	00		D66	24		D76	00

启动条件接点 M0 后,周一~周五的早上 08:00~12:00 跟下午 2:00~5:30 时 Y0.0=ON, 其它时间 Y0.0 OFF。

6.18 外围设备指令

6.18.1 外围设备指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1700	TKY	DTKY	–	10 键键盘输入
1701	HKY	DHKY	–	16 键键盘输入
1702	DSW	–	–	指拨开关输入
1703	ARWS	–	–	箭头键盘输入
1704	SEGL	–	–	七段显示器扫描输出

6.18.2 外围设备指令说明

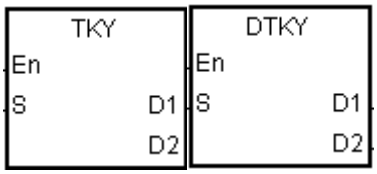
API	指令码			操作数	功能
1700	D	TKY		**S**，**D_1**，**D_2**	10 键键盘输入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	○															
D_1					●	●	●	●								
D_2		○	○	○				○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
D_1		●	●			●	●						
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



S ： 击键起始装置

D_1 ： 击键值存放处

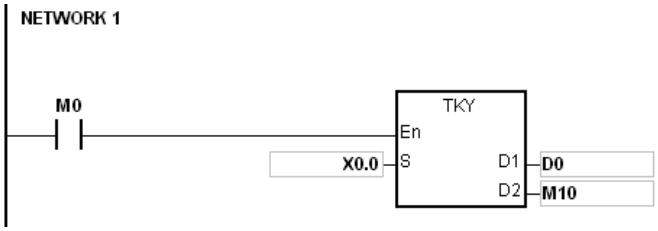
D_2 ： 按键输出信号

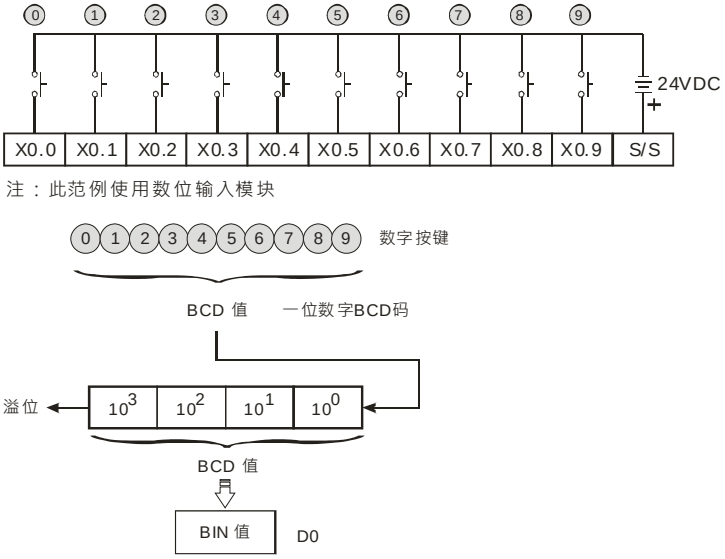
指令说明：

1. 本指令指定由 **S** 开始之 10 个外部输入点，依序代表 10 进位数字之 0~9。这 10 个外部输入点分别接上 10 个按键，依据这 10 个按键被压下之先后顺序可输入 4 位 10 进位数字 0~9,999（16 位指令），或 8 位 10 进位数字 0~99,999,999（32 位指令），并将输入之数值存放在 **D_1**，而 **D_2** 则存放键盘之按键情形。
2. **S** 连续占用 10 个 bit。
3. **D_2** 连续占用 11 个 bit，指令执行中，请勿改变 bit 的状态。
4. 当条件接点未启动时，**D_2** 开始的连续 11 个 bit 输出点会保持 OFF。
5. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。
6. 32 位指令 **D_1** 才可使用 HC 装置。

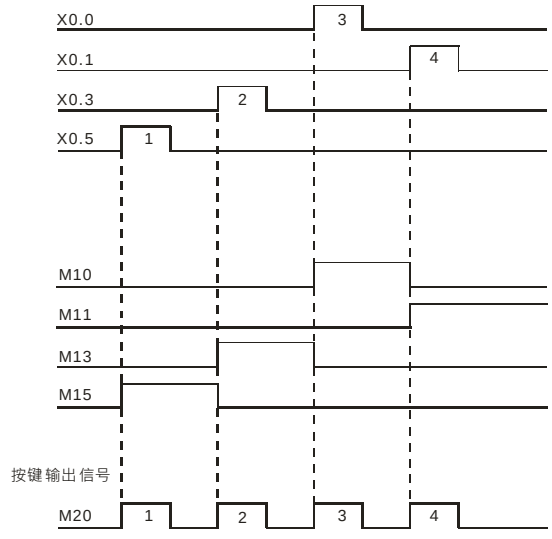
程序范例：

1. 指令指定 X0.0 开始的 10 个输入端与 0~9 的 10 个按键连接，当 M0=ON 时，指令执行，将键盘输入的数值以 BIN 值的形态存入 D0 中，而按键之情况则放在 M10~M19。





2. 如下列时序图所示，连接于数字键盘 X0.5、X0.3、X0.0、X0.1 的 4 个按键以①、②、③、④ 的顺序作打入的动作，结果为 5,301 被暂存于 D0 当中，D0 最大可容纳 9,999，超过 4 位数时，最前面的位数溢位。
3. X0.2 被按下后，至别的按键被按之前，M12=ON，其它的数按键亦相同。
4. 当 X0.0~X0.9 当中任何一个按键被按下时，M10~M19 当中一点对应 ON。
5. 任何一个按键被按下时，M20=ON。
6. 当条件接点 M0 变成 OFF 时，D0 之前的值无变化，但是，M10~M20 全部变成 OFF。



补充说明：

1. **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [10] of BOOL。
2. **D_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [11] of BOOL。

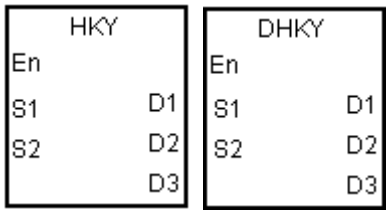
API	指令码			操作数	功能
1701	D	HKY		S_1 · S_2 · D_1 · D_2 · D_3	16 键键盘输入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	○															
S_2								●								
D_1		○														
D_2					●	●	●	●								
D_3		○	○	○				○								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2		●				●							
D_1	●												
D_2		●	●			●	●						
D_3	●												

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



S_1 ： 按键扫描输入起始装置

S_2 ： 系统内部使用

D_1 ： 按键扫描输出起始装置

D_2 ： 击键值存放处

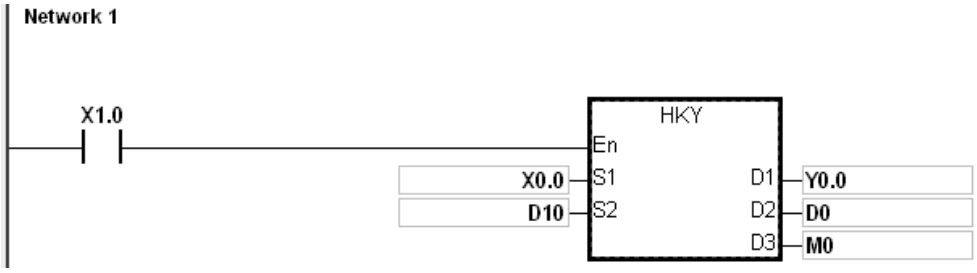
D_3 ： 按键输出信号

指令说明：

1. 本指令指定由 **S_1** 开始之连续 4 个外部输入点及由 **D_1** 开始之连续 4 个外部输出点以矩阵扫描之方式构成 16 键之键盘。键盘输入之数值存放在 **D_2**，而 **D_3** 则存放键盘之按键情形，如果有数个按键同时被按下时，以 **S_1** 的编号小的为优先。
2. 由数字键盘所打入的值被暂存于 **D_2** 当中，使用 16 位指令 HKY 时，**D_2** 最大可容纳 9,999，超过 4 位数时，最前面的位数溢位。使用 32 位指令 DHKY 时，**D_2** 最大可容纳 99,999,999，超过 8 位数时，最前面的位数溢位。
3. 指令执行完毕 SM692 会 ON；每执行完一次矩阵扫描，SM692 会 ON 一个扫描周期。
4. 32 位指令 **D_2** 才可使用 HC 装置。

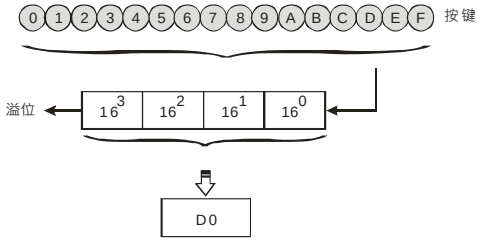
程序范例：

1. 指令指定 X0.0~X0.3 等 4 个输入端与 Y0.0~Y0.3 等 4 个输出端构成扫描 16 键之键盘。当 X1.0=ON 时，指令执行，由键盘输入的数值以 BIN 值的形态存入 D0 中，而按键之情况则放在 M0~M7。

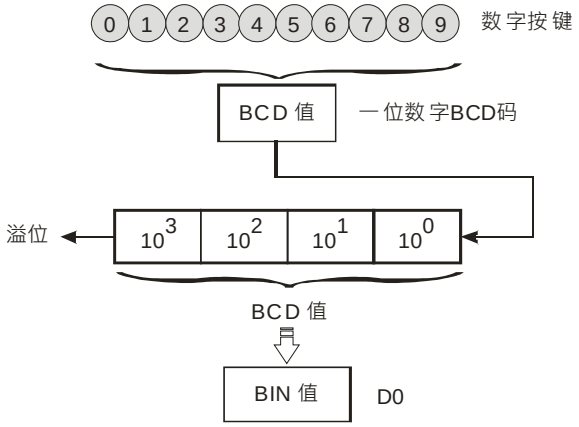


标志 SM691 之功能：

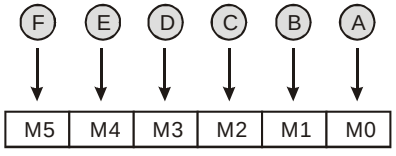
- SM691=ON 时，则 HKY 指令可以输入 0~F 的 16 进位数值。
  - ■ 数字输入：



- SM691=OFF 时，则 HKY 指令 A~F 当成功能键使用。
  - ■ 数字输入：

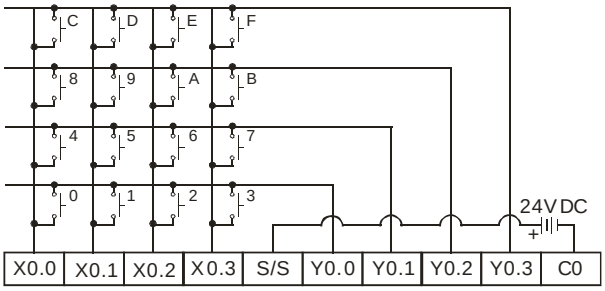


  - ■ 功能键输入：
    - ◆ 按 A 键时，M0=ON 并保持，接着再按 D 键时，M0 变成 OFF、M3=ON 并保持。
    - ◆ 复数个按键同时按，以先按者优先。



2. 按键输出信号：
   - A~F 当中任何一个按键被按时，M6=ON 一次。
   - 0~9 当中任何一个按键被按时，M7=ON 一次。

3. 当条件接点 X1.0 变成 OFF 时，D0 之前的输入值无变化，但是 M0~M7 全部变成 OFF。
4. 外部配线：



注：本范例使用电晶体输出模块

补充说明：

1. 本指令被执行时，当扫描周期太长或太短都可能造成击键不良，因此可运用下列技巧来克服。
2. 当扫描周期太短时，可能造成 I/O 来不及反应而无法读取正确之击键，此时，可将扫描时间加以固定。
3. 当扫描周期太长时，可能会使按键反应变得迟钝，可将此指令写在时间中断工作（TASK）内，固定时间执行此指令。
4. **S_1** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of BOOL。
5. **D_1** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of BOOL。
6. **D_3** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [8] of BOOL。

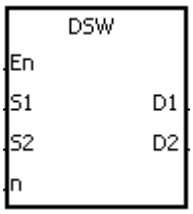
API	指令码	操作数	功能
1702	DSW	**S_1 · S_2 · D_1 · D_2 · n**	指拨开关输入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	○															
S_2								●								
D_1		○														
D_2					●	●		●								
n					●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2		●				●							
D_1	●												
D_2		●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 指拨开关扫描输入起始装置

S_2 ： 系统内部使用

D_1 ： 指拨开关扫描输出起始装置

D_2 ： 指拨开关设定值存放处

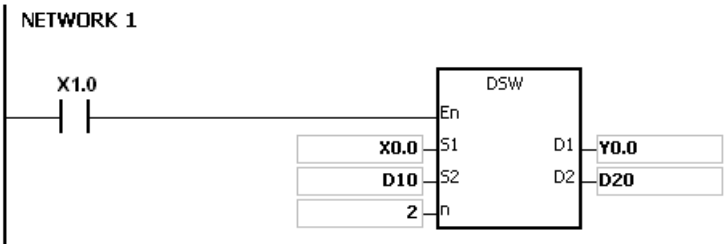
n ： 指拨开关所连接之组数

指令说明：

1. 本指令由 **S_1** 开始的连续 4 个或 8 个外部输入点及由 **D_1** 开始的连续 4 个外部输出点扫描读取 1 组或 2 组 4 位数指拨开关，指拨开关设定值存放在 **D_2**，由 **n** 决定读取 4 位数指拨开关有 1 组或 2 组。
2. 当 **n**=1 时，**D_2** 操作数占用一个寄存器。**n**=2 时，**D_2** 操作数会连续占用 2 个寄存器。
3. **S_2**、**S_{2+1}** 占用两个装置给系统内部使用，用户请勿变更内容值。
4. 每循环扫描一次完毕标志信号 SM694=ON 一个扫描周期。
5. 当条件接点未启动时，**D_1** 开始的连续 4 个外部输出点会保持 OFF。
6. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。

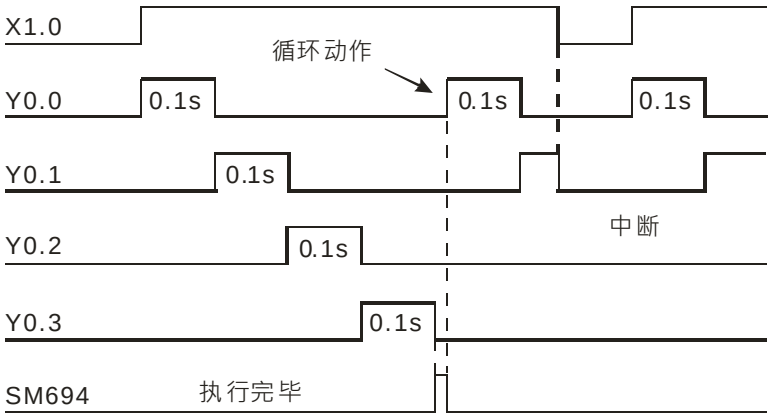
程序范例：

1. 由 X0.0~X0.3 及 Y0.0~Y0.3 组成第一组指拨开关回路，由 X0.4~X0.7 及 Y0.0~Y0.3 组成第二组指拨开关回路。当 X1.0=ON 时，指令开始执行，第一组指拨开关的设定值被读入并转换成 BIN 值后存放至 D20 中，第二组指拨开关的设定值被读入并转换成 BIN 值后存放至 D21 中。

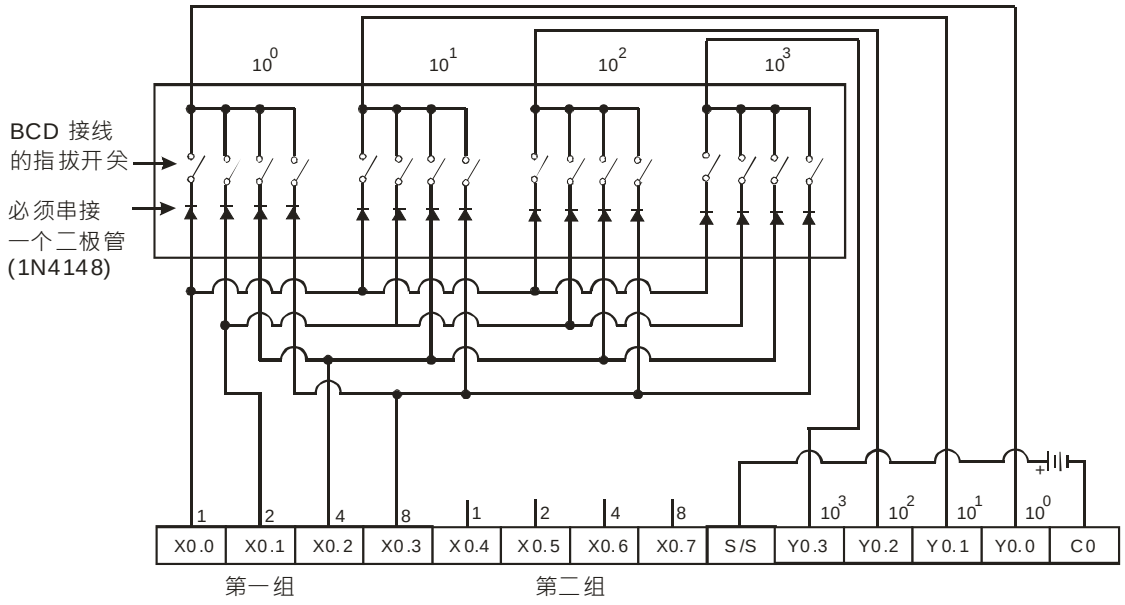


2. 当 X1.0=ON 时，Y0.0~Y0.3 自动循环扫描 ON，每循环一次，执行完毕标志信号 SM694=ON 一个扫描周期。

3. 扫描用输出端 Y0.0~Y0.3 请使用晶体管输出。



4. 指拨开关输入



注：此范例使用晶体管输出模块

补充说明：

1. **n** 超出范围时，指令不执行，SM0=ON，错误码 SR0=16#200B。

2. **D_1** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of BOOL。

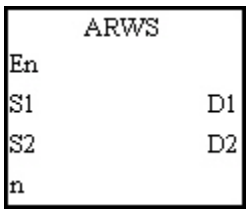
API	指令码	操作数	功能
1703	ARWS	**S_1 · S_2 · D_1 · D_2 · n**	箭头键盘输入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	○															
S_2								●								
D_1					●	●		●								
D_2		○														
n					●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2		●				●							
D_1		●				●							
D_2	●												
n		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 击键起始装置

S_2 ： 系统内部使用

D_1 ： 欲显示于七段显示器之装置

D_2 ： 七段显示器扫描输出起始装置

n ： 输出信号及扫描信号之极性指示

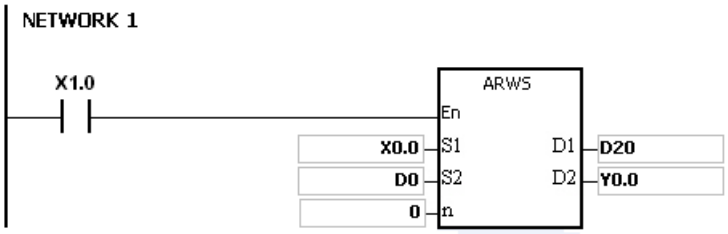
指令说明：

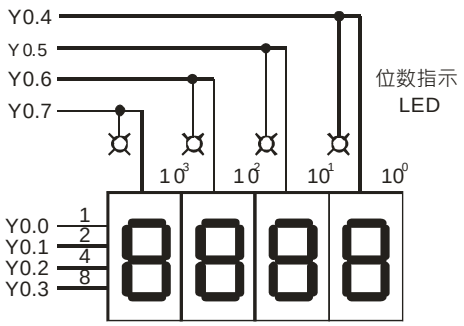
1. 本指令执行，**S_1** 定义为下键，**S_1**+1 定义为上键，**S_1**+2 定义为右键，**S_1**+3 定义为左键，利用上下左右键来执行外部设定值的操作及显示。将设定值存放于 **D_1** 当中，设定值范围：0~9,999。
2. **S_1** 连续占用 4 个 bit 装置。
3. **S_2** 系统内部使用，用户请勿变更内容值。
4. **D_2** 连续占用 8 个 bit 装置。
5. 条件接点未启点时，**D_2** 连续的 8 个 bit 装置保持 OFF。
6. n 的范围：0~3，功能请参考 API1704 SEGL 的补充说明。
7. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。

程序范例：

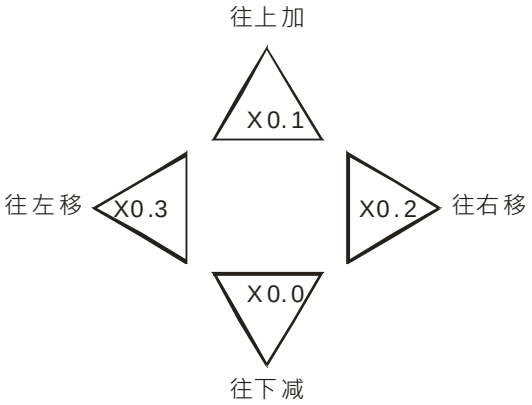
1. 本指令执行，X0.0 定义为下键，X0.1 定义为上键，X0.2 定义为右键，X0.3 定义为左键，利用上下左右键来执行外部设定值的操作及显示。将设定值存放于 D20 当中，设定值范围：0~9,999。

2. 当 X1.0=ON 时，位数 10^3 为有效设定位数，如果按左按键时，则有效设定位数呈现 $10^3 \rightarrow 10^0 \rightarrow 10^1 \rightarrow 10^2 \rightarrow 10^3 \rightarrow 10^0$ 的方向循环跳动。
3. 如果按右移按键，则有效设定位呈现 $10^3 \rightarrow 10^2 \rightarrow 10^1 \rightarrow 10^0 \rightarrow 10^3 \rightarrow 10^2$ 的方向循环跳动。在循环的同时，由 Y0.4~Y0.7 所连接的位数指示灯亦循环 ON 作有效设定位数的指示。
4. 如果按往上加按键时，则有效设定位数的内容由 0→1→2→…8→9→0→1 作变化。如果按往下减按键时，则有效设定位数的内容由 0→9→8→…1→0→9 作变化，同时，变化值亦被显示在七段显示器上。





显示设定值的4位数七段显示器



下面的4个开关用来移动
位数的左右及设定值的加减

补充说明：

1. **n** 超出范围时，指令不执行，SM0=ON，错误码 SR0=16#200B。
2. **S_1** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of BOOL。
3. **D_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [8] of BOOL。

API	指令码	操作数	功能
1704	SEGL	**S_1 · S_2 · D · n**	7 段显示器扫描输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●		○	○				
S_2								●								
D		○														
n					●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D	●												
n		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：

S_1 ： 欲显示于七段显示器之来源装置

S_2 ： 系统内部使用

D ： 七段显示器扫描输出起始装置

n ： 输出信号及扫描信号之正负逻辑设定

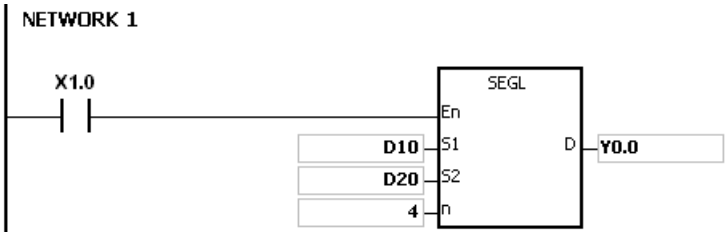
指令说明：

1. 本指令占用 **D** 开始的连续 8 个或 12 个外部输出点，作为 1 组或 2 组 4 位数七段显示器的显示数据及扫描信号输出。每个位数均带有 7 段显示器驱动器，该驱动器是将输入的 BCD 码转换 7 段显示器的驱动信号；驱动器并带有栓锁控制信号，可将 7 段显示器显示保持。
2. **S_1** 中之数值为欲显示到第一组七段显示器之数值，**S_{1+1}** 中之数值为欲显示到第二组七段显示器之数值。
3. **S_2** 系统内部使用，用户请勿变更内容值。
4. **n**=0~7。请参考补充说明。
5. 由 **n** 决定扫描输出 4 位数七段显示器有 1 组或 2 组，且 **n** 也用来指定 PLC 输出端的正负逻辑输出。
6. 4 位数 1 组时，占用输出点 8 个，4 位数 2 组时，占用输出点 12 个。
7. 本指令执行时，扫描输出端顺序循环动作，指令执行中条件接点变成 OFF 再 ON 时，扫描输出端重新执行。
8. 每循环扫描一次完毕标志信号 SM693=ON 一个扫描周期。

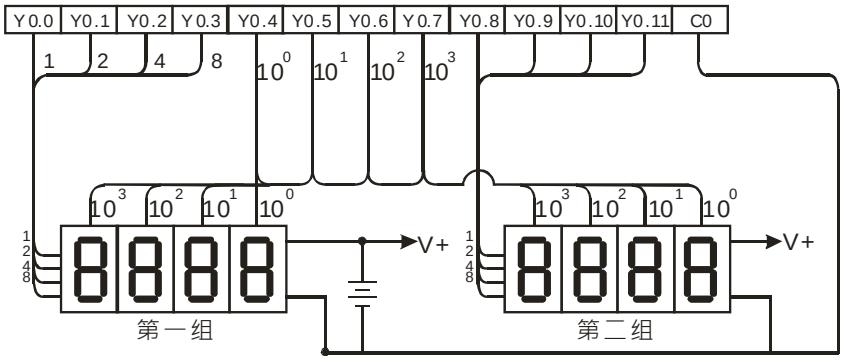
程序范例：

1. 当 X1.0=ON 时，指令开始执行，由 Y0.0~Y0.7 构成七段显示器扫描回路，D10 中之数值被转换成 BCD 码后送到第一组七段显示器显示出来，D11 中之数值被转换成 BCD 码后送到第二组七段显示器显示出来，

若 D10 或 D11 中之数值超过 9,999 将发生运算错误。



2. 当 X1.0=ON 时，Y0.4~Y0.7 会自动循环扫描，每循环扫描一次需 12 个扫描时间，每循环扫描一次完毕标志信号 SM693=ON 一个扫描周期。
3. 一组 4 位数的时候 n=0~3。
   - 将已经译码的 7 段显示模块 1、2、4、8 各端各自并接后连接至 PLC 的 Y0.0~Y0.3，而各位数的 Latch 端单独连接至 PLC 的 Y0.4~Y0.7。
   - 当 X1.0=ON 时，指令被执行，D10 的内容随着 Y0.4~Y0.7 的循环动作被顺序传送至七段显示器作显示。
4. 二组 4 位数的时候 n=4~7。
   - 将已经解碼的七段显示器 1、2、4、8 各端各自并接后连接至 PLC 的 Y0.8~Y0.11，而各位数的 Latch 端与第一组共享 Y0.4~Y0.7。
   - D10 的内容被传送到第一组七段显示器上、D11 的内容被传送到第二组七段显示器作显示。若 D10=1234，D11=4321，则第一组将会显示 1 2 3 4，第二组显示 4 3 2 1。
5. 七段显示器扫描输出接线图：



注：此范例使用晶体管输出模块

补充说明：

1. **n** 的设定值：是用来设定晶体管输出为正极性或负极性回路，连接的七段显示器是一组 4 位数或者是二组 4 位数。
2. PLC 输出点必须选用为晶体管模块，输出为 NPN 型式，采集电极式输出，在电路的连接上，输出必须连接一提升电阻至 DC 电源（小于 30VDC），因此当输出点 Y 导通时，信号输出为低电位。

- BCD 码正逻辑（负极性）输出

BCD 数值				Y 输出（BCD 码）				信号输出			
b_3	b_2	b_1	b_0	8	4	2	1	A	B	C	D
0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	1	0	0	0	1	1	1	1	0
0	0	1	0	0	0	1	0	1	1	0	1
0	0	1	1	0	0	1	1	1	1	0	0
0	1	0	0	0	1	0	0	1	0	1	1
0	1	0	1	0	1	0	1	1	0	1	0
0	1	1	0	0	1	1	0	1	0	0	1
0	1	1	1	0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	0	0	1	1	1
1	0	0	1	1	0	0	1	0	1	1	0

- BCD 码负逻辑（正极性）输出

BCD 数值				Y 输出（BCD 码）				信号输出			
b_3	b_2	b_1	b_0	8	4	2	1	A	B	C	D
0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	1	1	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	0	0	1	0
0	0	1	1	1	1	0	0	0	0	1	1
0	1	0	0	1	0	1	1	0	1	0	0
0	1	0	1	1	0	1	0	0	1	0	1
0	1	1	0	1	0	0	1	0	1	1	0
0	1	1	1	1	0	0	0	0	1	1	1
1	0	0	0	0	1	1	1	1	0	0	0
1	0	0	1	0	1	1	0	1	0	0	1

- 显示扫描栓锁（Latch）信号

正逻辑（负极性）		负逻辑（正极性）	
Y 输出（Latch）	输出控制信号	Y 输出（Latch）	输出控制信号
1	0	0	1

- 参数 n 的设定值

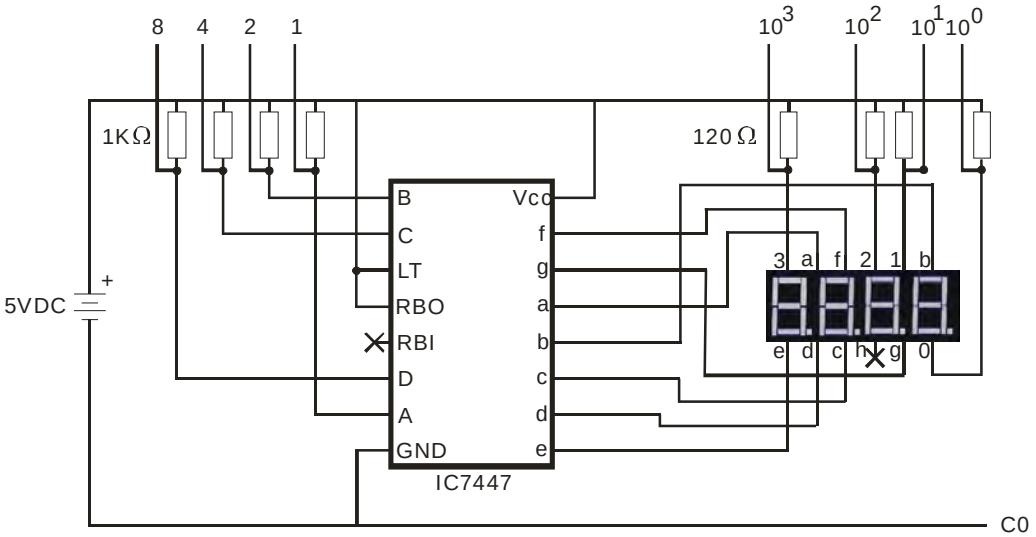
7 段显示器组数	一组				二组			
BCD 码数据 Y 输出	+		-		+		-	
显示扫描栓锁信号	+	-	+	-	+	-	+	-
n	0	1	2	3	4	5	6	7

‘+’：正逻辑（负极性）输出

‘-’：反逻辑（正极性）输出

- PLC 的晶体管输出极性与 7 段显示器的输入极性是否相同或者是不同时，可透过可参数 **n** 的设定值来相互匹配。

- 采用共阳极四位数 7 段显示器搭配 IC7447 配置如下图



6.19 通讯指令

6.19.1 通讯指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1806	LRC	–	–	和检查 LRC 模式
1807	CRC	–	–	和检查 CRC 模式
1808	MODRW	–	–	MODBUS 通讯指令
1812	COMRS	–	–	自定义通讯传送与接收指令
1813	COMDF	–	✓	串行通讯端口格式设定
1814	VFDRW	–	–	台达变频器通讯指令
1815	ASDRW	–	–	台达伺服通讯指令
1816	CCONF	–	✓	通讯端口数据交换表参数设定
1817	MODRWE	–	–	MODBUS 直接指定通讯端口通讯指令
1818	DNETRW	–	–	DeviceNet 通讯读写指令
1819	CANRS	–	–	CAN 自定义通讯传送与接收指令
1820	DMVSH	–	–	台达 DMV 启动检验与通讯指令
1821	DESO	–	–	指定数据交换表通讯编号发送一次

6.19.2 通讯指令说明

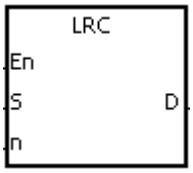
API	指令码	操作数	功能
1806	LRC	**S**、**n**、**D**	和检查 LRC 模式

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S								●	●							
n								●	●				○	○		
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S ： ASCII 模式和检查运算起始装置

n ： 运算组数

D ： 存放运算结果之起始装置

指令说明：

1. LRC 检查码：请参考补充说明。
2. **n**：运算组数须为偶数，范围 1~1000 不在此范围则视为运算错误，指令不执行，SM0、SM1=ON，SR0 记录错误码 16#200B。
3. 16 位转换模式：当 SM606=OFF 时，S 起始装置将其 16 进位数据区分为上 8 位、下 8 位，将每一个位数做 LRC 检查码运算，传送到 D 的上 8 位及下 8 位中，运算的位数以 n 来设定。
4. 8 位转换模式：当 SM606=ON 时，**S** 起始装置将其 16 进位数据区分为上 8 位（无效数据）、下 8 位，将每一个位数做 LRC 检查码运算，传送到 **D** 的下 8 位中占用 2 个寄存器，运算的位数以 n 来设定。(**D** 的上 8 位全部为 0)

程序范例：

1. PLC 与 VFD-S 系列交流伺服电机驱动器联机（ASCII 模式，SM210 =OFF）、(8 位模式，SM606=ON)，传送数据预先写入读取 VFD-S 参数地址 16#2101 开始的 6 笔数据。

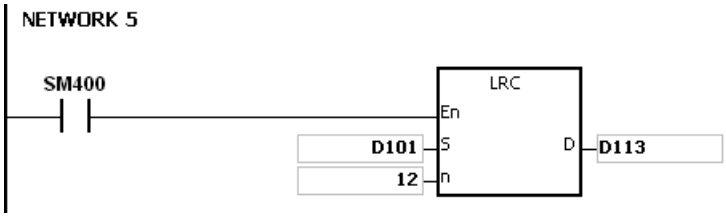
   PLC⇨VFD-S，PLC 传送：“：01 03 2101 0006 D4 CR LF ”

   PLC 传送数据寄存器（PLC 传送信息）

寄存器	DATA		说明	
D100 下	‘：’	16#3A	STX	
D101 下	‘0’	16#30	ADR 1	ADR（10）为交流伺服电机驱动器地址

寄存器	DATA		说明	
D102 下	‘1’	16#31	ADR 0	
D103 下	‘0’	16#30	CMD 1	CMD（10）为命令码
D104 下	‘3’	16#33	CMD 0	
D105 下	‘2’	16#32	起始数据地址 Starting Data Address	
D106 下	‘1’	16#31		
D107 下	‘0’	16#30		
D108 下	‘1’	16#31		
D109 下	‘0’	16#30	数据（word）个数 Number of Data（count by word）	
D110 下	‘0’	16#30		
D111 下	‘0’	16#30		
D112 下	‘6’	16#36		
D113 下	‘D’	16#44	LRC CHK 0	LRC CHK（01）为错误检查码
D114 下	‘4’	16#34	LRC CHK 1	
D115 下	CR	16#0D	END	
D116 下	LF	16#0A		

上列 LRC CHK（0，1）为错误检查码可由指令 LRC 算出（8 位 Mode，SM606=ON）



LRC 检查码：16#01+16#03+16#21+16#01+16#00+16#06=16#2C，然后取 2 的补码=16#D4。此时，‘D’（16#44） 存于 D113 下 8 位内，‘4’（16#34）存于 D114 下 8 位内。

补充说明：

1. 有一通讯数据的 ASCII 模式，格式如下：

STX	‘ : ’	起始字符=‘ : ’（16#3A）
Address Hi	‘ 0 ’	通讯地址： 8-bit 地址由 2 个 ASCII 码组合
Address Lo	‘ 1 ’	
Function Hi	‘ 0 ’	功能码： 8-bit 功能码由 2 个 ASCII 码组合
Function Lo	‘ 3 ’	
DATA（n-1） **…….** **DATA 0**	‘ 2 ’	数据内容： n×8-bit 数据内容由 2n 个 ASCII 码组合
	‘ 1 ’	
	‘ 0 ’	
	‘ 2 ’	
	‘ 0 ’	
	‘ 0 ’	
	‘ 0 ’	
	‘ 2 ’	

LRC CHK Hi	‘D’	LRC 检查码： 8-bit 检查码由 2 个 ASCII 码组合
LRC CHK Lo	‘7’	
END Hi	CR	终止符： END Hi=CR（16#0D），END Lo=LF（16#0A）
END Lo	LF	

2. LRC 检查码：由通讯地址到数据内容结束加起来的值取 2 的补码即为检查码（LRC Check）。例如：16#01+16#03+16#21+16#02+16#00+16#02=16#29，然后取 2 的补码=16#D7。

API	指令码	操作数	功能
1807	CRC	**S**、**n**、**D**	检查 CRC 模式

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S								●	●							
n								●	●				○	○		
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：

S ： RTU 模式和检查运算起始装置

n ： 运算组数

D ： 存放运算结果之起始装置。

指令说明：

1. CRC 检查码计算方式，请参考补充说明。
2. **n**：范围 1~1000 不在此范围则视为运算错误，指令不执行，SM0、SM1=ON，SR0 记录错误码 16#200B。
3. 16 位转换模式：当 SM606=OFF 时，S 起始装置其数据区分为上 8 位、下 8 位，将每一个位数做 CRC 检查码运算，传送到 **D** 的上 8 位及下 8 位中，运算的位数以 n 来设定。
4. 8 位转换模式：当 SM606=ON 时，**S** 起始装置其数据区分为上 8 位 (无效数据)、下 8 位，将每一个位数做 CRC 检查码运算，传送到 **D** 的下 8 位中占用 2 个寄存器，运算的位数以 **n** 来设定。(**D** 的上 8 位全部为 0)

程序范例：

1. PLC 与 VFD-S 系列交流伺服电机驱动器联机 (RTU 模式，SM210=ON)、(16 位模式，SM606=ON)，传送数据预先写入欲写入 VFD-S 参数地址 16#2000 写入内容为 16#12。

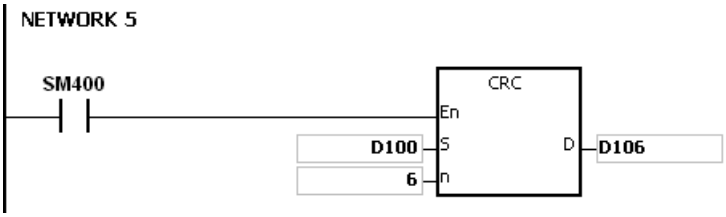
   PLC VFD-S，PLC 传送：01 06 2000 0012 02 07

   PLC 传送数据寄存器 (PLC 传送信息)

寄存器	DATA	说明
D100 下	16#01	Address
D101 下	16#06	Function
D102 下	16#20	数据地址

寄存器	DATA	说明
D103 下	16#00	数据地址
D104 下	16#00	数据内容 Data content
D105 下	16#12	
D106 下	16#02	CRC CHK 0
D107 下	16#07	CRC CHK 1

上列 CRC CHK（0，1）为错误检查码可由指令 CRC 算出（8 位 Mode，SM606=ON）



CRC 检查码：此时，16#02 存于 D106 下 8 位内，16#07 存于 D107 下 8 位内。

补充说明：

1. 有一通讯数据之 RTU 模式，格式如下：

START	时间间隔
Address	通讯地址：8-bit 二进制地址
Function	功能码：8-bit 二进制
DATA（n-1）	数据内容： n×8-bit 数据
…….	
DATA 0	
CRC CHK Low	CRC 检查码： 16-bit CRC 检查码由 2 个 8-bit 二进制组合
CRC CHK High	
END	时间间隔

2. CRC 检查码：检查码由 Address 到 Data content 结束。其运算规则如下：

   步骤一： 令 16-bit 寄存器（CRC 寄存器）=16#FFFF

   步骤二： Exclusive OR 第一个 8-bit Byte 的信息指令与低位 16-bit CRC 寄存器，Exclusive OR，将结果存入 CRC 寄存器内。

   步骤三： 右移一位 CRC 寄存器，将 0 填入高位处。

   步骤四： 检查右移的值，如果是 0 将步骤 3 的新值存入 CRC 寄存器内否则 Exclusive OR 16#A001 与 CRC 寄存器，将结果存入 CRC 寄存器内。

   步骤五： 重复步骤 3～步骤 4，将 8-bit 全部运算完成。

   步骤六： 重复步骤 2～步骤 5，取下一个 8-bit 的信息指令，直到所有信息指令运算完成。最后，得到的 CRC 寄存器的值，即是 CRC 的检查码。值得注意的是 CRC 的检查码必须交换放置于信息指令的检查码中。

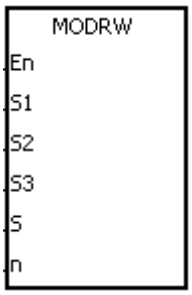
API	指令码	操作数	功能
1808	MODRW	**S_1 · S_2 · S_3 · S · n**	MODBUS 数据读写

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S								●								
n								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S	●	●				●							
n		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 联机装置地址

S_2 ： 通讯功能码

S_3 ： 欲读写数据的地址

S ： 欲读写之数据存放寄存器

n ： 读写数据长度

指令说明：

1. **S_1**：联机装置地址（Unit Address）。指定范围 0~254。0 为广播模式。
2. **S_2**：MODBUS 通讯功能码（Function Code）。

   例如：

功能码	功能说明	数据长度	支持的 AS 从站装置
16#01	读取多笔位装置（Bit）命令	1~1600	X、Y、M、SM、S、T、C、HC
16#02	读取多笔位装置（Bit）命令	1~1600	X、Y、M、SM、S、T、C、HC
16#03	读取多笔字符装置（Word）命令	1~100	X、Y、SR、D、T、C、HC、E
16#04	读取多笔字符装置（Word）命令	1~100	X
16#05	单笔位装置（Bit）状态写入命令	1	Y、M、SM、S、T、C、HC
16#06	单笔字符装置（Word）数据写入命令	1	Y、SR、D、T、C、HC、E
16#0F	多笔位装置（Bit）状态写入命令	1~1600	Y、M、SM、S、T、C、HC

功能码	功能说明	数据长度	支持的 AS 从站装置
16#10	多笔字符装置（Word）数据写入命令	1~100	Y、SR、D、T、C、HC、E

目前仅支持上述功能码，其余功能码将无法执行。请参考下列程序范例。

3. **S_3**：欲读写数据的地址（Device Address）。联机装置的内部装置地址，若地址对于被指定的装置不合法，则联机装置会响应错误信息。（例如：16#8000 通讯地址在 DVP-ES2 为不合法地址）。
4. **S**：欲读写之数据（Source or Destination）。由用户设定寄存器，将欲写入数据长度的数据事先存入寄存器内。或数据读取后存放之寄存器。
5. **n**：读写数据长度（Data Length）。当使用 word 类型通讯功能码时，设定的数据量不可以大于 100 words。当使用 Bit（Bool）相关的通讯功能码，其单位为 Bit，且范围为 1~1600Bits。
6. S_3，S，n 操作数依不同功能码其功能如下：

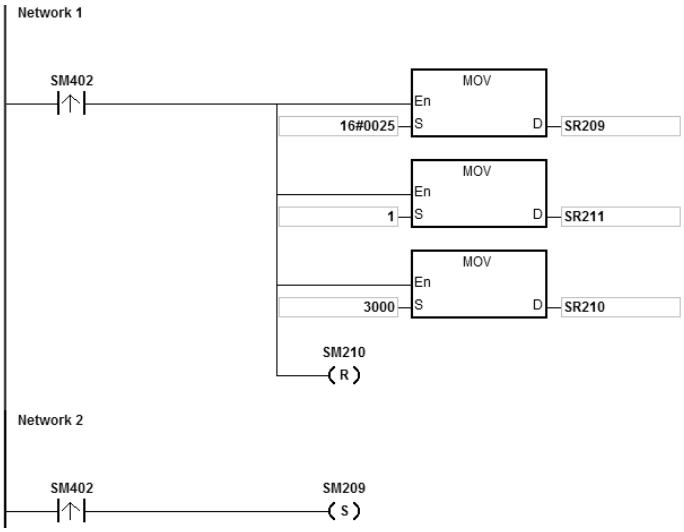
功能码	S_3	S	n
16#01	欲读取数据的地址	欲读取之数据存放寄存器	读取数据长度
16#02	欲读取数据的地址	欲读取之数据存放寄存器	读取数据长度
16#03	欲读取数据的地址	欲读取之数据存放寄存器	读取数据长度
16#04	欲读取数据的地址	欲读取之数据存放寄存器	读取数据长度
16#05	欲写入数据的地址	写入状态值	无意义
16#06	欲写入数据的地址	欲写入数据存放寄存器	无意义
16#0F	欲写入数据的地址	欲写入数据存放寄存器	写入数据长度
16#10	欲写入数据的地址	欲写入数据存放寄存器	写入数据长度

7. 本指令于程序中使用次数并无限制，但是不同通讯指令，在同一时间使用相同的通讯端口时，仅会只有一个指令被执行。另外搭配各通讯口的发送标志时，也需特别注意得要在此指令的前一行设定，否则容易造成 2 个通讯口各自独立的通讯数据互相混用的情况。
8. 当发生通讯逾时时，逾时标志会 ON 起来。若此问题已解除，则可以将此逾时标志清除为 OFF，在使用 MODRW 指令时，逾时时间不可设为 0，必须在 100 ~ 32767ms 区间内；若逾时时间设为 0，则会以 200ms 来执行。
9. 于 MODBUS ASCII 模式时，用户只需设定好，所欲传送之数据（非 ASCII 数值），本指令会自动加入启始字符（ :)、转换 ASCII 码、计算检查码（LRC）和终止符（CR LF）; 同样的，其接收数据 ASCII 字符形式将储存于内部寄存器内，等接收完毕，PLC 内部自动将其数据内容转换为 HEX 数值，并比对检查，若通讯数据皆正确时，将存放在 **S** 中，最后设定通讯完成标志 SM。
10. 于 MODBUS RTU 模式时，用户只需设定好，所欲传送之数据，本指令会自动加入检查码（CRC），其接收数据以 HEX 数值形式储存于 **S** 中。
11. 此指令不可使用于 ST 结构化语言编辑、中断程序或仅被调用一次性的功能块（FB）程序。

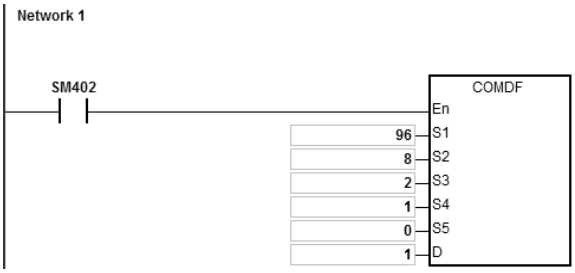
通讯协议设定范例：

1. 这里以 PLC COM1 搭配使用特殊寄存器为例做说明，其它 PLC 通讯端口依此类推。

2. PLC COM1(RS485)通讯格式设定成 ASCII ､9600 ､8 ､E ､1 ､用户可利用 HWCONFIG ､API1813 COMDF 或是特殊寄存器来设定通讯端口。(HWCONFIG 相关设定请参照 ISPSoft 使用手册，通讯相关 SR，SM 寄存器设定可参考第 6.19.3 节)。
3. 设定通讯端口之通讯速率与格式为 9600，8，E，1（SR209=16#0025)。
4. 设定其逾时时间为 3000ms（SR210=3000)。
5. 设定为 ASCII 模式（SM210=OFF ）
6. 通讯端口通讯格式设定生效（SM209=ON)。



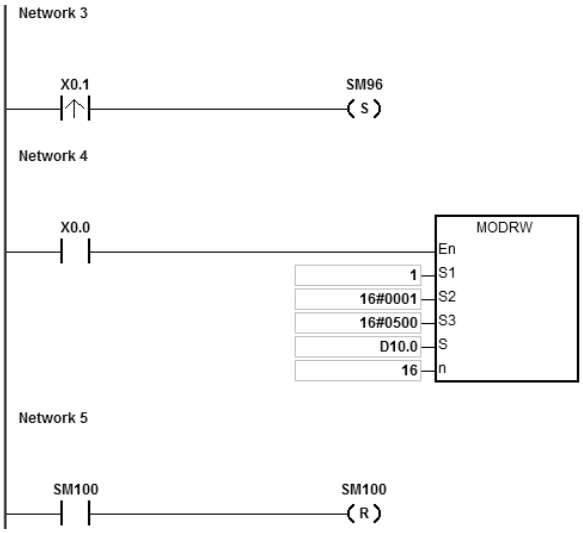
用户也可改用 API1813 COMDF 设定，则可以省略此步骤。



用户也可于 ISPSoft - > HWCONFIG - > COM Port 内直接对通讯端口进行设定，则可以省略此步骤。

程序范例一：

1. 功能码 01 (16#01) : 读取多笔位数据。(此范例读取 16 笔)。(功能码 02 作法与功能码 01 同)



2. AS 主机与 DVP-ES2 主机联机：

当 SM96 为 ON 且 X0.0 也为 ON 的状态时，将会对 DVP-ES2 的 Y0~Y17 进行数据的读取。

DVP-ES2 中 Y0 的地址为 16#0500 且 Y0~Y17 的状态如下：

装置	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
状态	ON	ON	OFF	ON	OFF	OFF	ON	OFF
值	D				2			
装置	Y17	Y16	Y15	Y14	Y13	Y12	Y11	Y10
状态	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
值	0				4			

MODRW 指令内容说明：

操作数	说明	装置
S1	联机装置地址	16#0001
S2	通讯功能码	16#0001
S3	欲读取 Y0 数据的地址	16#0500
S	读回数据存放的起始寄存器	D10.0
n	读取数据长度	16

ASCII 模式做通讯 :(此处 ASCII 码不刻意转换表示，皆以 16#数值表示)

- AS 传送通讯命令 : " : 01 01 05 00 00 10 E9 CR LF"
- AS 接收通讯命令 : " : 01 01 02 D2 04 26 CR LF"

RTU 模式做通讯：

- AS 传送通讯命令："01 01 05 00 00 10 3D 0A"

- AS 接收通讯命令：“01 01 02 D2 04 E4 9F”

因为数据正确无误，所以 SM100=ON。

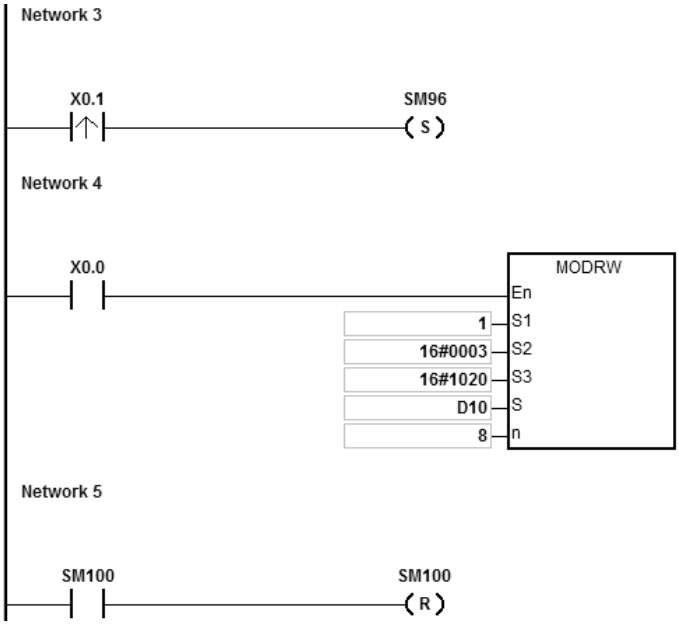
3. DVP-ES2 所回复的内容会储存在 D10.0 到 D10.15。(读回的数据为 D10.15~D10.0=16#04D2)

装置	D10.7	D10.6	D10.5	D10.4	D10.3	D10.2	D10.1	D10.0
状态	ON	ON	OFF	ON	OFF	OFF	ON	OFF
值	D				2			
装置	D10.15	D10.14	D10.13	D10.12	D10.11	D10.10	D10.9	D10.8
状态	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
值	0				4			

4. DVP-ES2 所回传的数据接收完毕后，会对 DVP-ES2 所回传的数据格式进行确认，是否符合数据格式，如果正确无误则 SM100=ON，如果回传数据格式错误则 SM102=ON。

程序范例二：

1. 功能码 03 (16#03) : 读取多笔寄存器数据。(此范例读取 8 笔) (功能码 04 作法与功能码 03 同)



2. AS 主机与 DVP-ES2 联机：

当 SM96 为 ON 且 X0.0 也为 ON 的状态时，将会对 DVP-ES2 的 D32~D39 进行数据读取。

DVP-ES2 中 D32 的地址为 16#1020 且 D32~D239 的内容值：

装置	D32	D33	D34	D35	D36	D37	D38	D39
值 (16#)	1234	5678	1122	3344	5566	7788	99AA	BBCC

MODRW 指令内容说明：

操作数	说明	装置
S1	联机装置地址	16#0001
S2	通讯功能码	16#0003
S3	读取 D32 数据的地址	16#1020
S	读回数据存放的起始寄存器	D10
n	读取数据长度	8

ASCII 模式做通讯：(此处 ASCII 码不刻意转换表示，皆以 16#数值表示)

- AS 传送通讯命令：“：01 03 10 20 00 08 C4 CR LF”
- AS 接收通讯命令：“：01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC AA CR LF”

RTU 模式做通讯：

- AS 传送通讯命令：“01 03 10 20 00 08 41 06”
- AS 接收通讯命令：“01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC 90 FE”

因为数据正确无误，所以 SM100=ON。

3. DVP-ES2 所回传的数据会储存在 D10 至 D17 装置内。

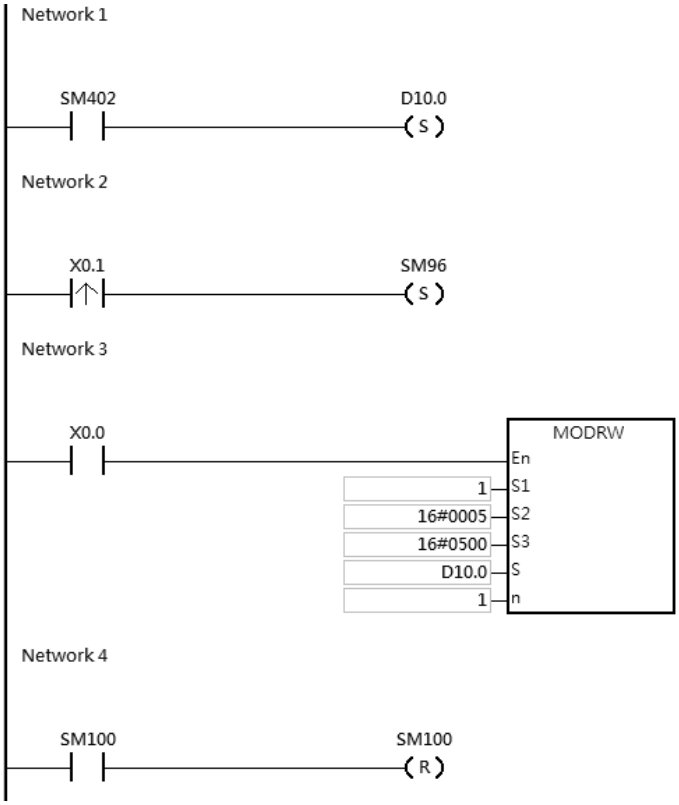
- D10~D17 的内容值：

装置	D10	D11	D12	D13	D14	D15	D16	D17
值（16#）	1234	5678	1122	3344	5566	7788	99AA	BBCC

4. DVP-ES2 所回传的数据接收完毕后，会对 DVP-ES2 所回传的数据格式进行确认，是否符合数据格式，如果正确无误则 SM100=ON，如果回传数据格式错误则 SM102=ON。

程序范例三：

1. 功能码 05（16#05）：单笔位装置状态写入命令。（此例设定装置 ON）



2. AS 主机与 DVP-ES2 联机：

设定 D10.0=ON（准备设定 DVP-ES2 的 Y0=ON）

当 SM96 为 ON 且 X0.0 也为 ON 的状态时，将会对 DVP-ES2 的 Y0 设定状态（Y0 的地址为 16#0500）。

MODRW 指令内容说明：

操作数	说明	装置
S1	联机装置地址	1
S2	通讯功能码	16#0005
S3	写入 Y0 的装置地址	16#0500
S	数据写入 Y0 存放的寄存器	D10.0
n	写入数据长度（在此功能码无意义）	1

ASCII 模式做通讯：（此处 ASCII 码不刻意转换表示，皆以 16#数值表示）

- AS 传送通讯命令：“：01 05 05 00 FF 00 F6 CR LF”
- AS 接收通讯命令：“：01 05 05 00 FF 00 F6 CR LF”

RTU 模式做通讯：

- AS 传送通讯命令：“01 05 05 00 FF 00 8C F6”

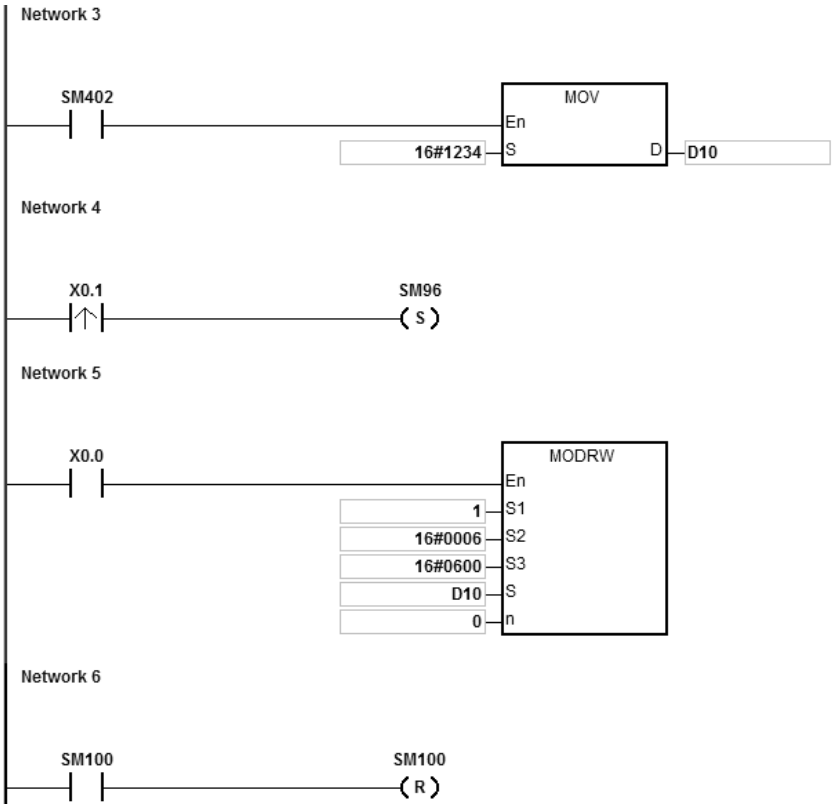
- AS 接收通讯命令：“01 05 05 00 FF 00 8C F6”

因为数据正确无误，所以 SM100=ON。

3. DVP-ES2 所回传的数据接收完毕后，会对 DVP-ES2 所回传的数据格式进行确认，是否符合数据格式，如果正确无误则 SM100=ON，如果回传数据格式错误则 SM102=ON。
4. 当 DVP-ES2 接收到此通讯命令的时候，会强制 Y0 为 ON 的状态。
5. 此功能码为单笔写入功能，所以于本指令中的 n 值，并不会被使用到。

程序范例四：

1. 功能码 06（16#06）：单笔字符装置数据写入命令。



2. AS 与 DVP-ES2 联机：
3. 给定 D10=16#55AA（准备写入 DVP-ES2 T0 的数据内容）

当 SM96 为 ON 且 X0.0 也为 ON 的状态时，将会对 DVP-ES2 的 T0 写入数据（T0 的地址为 16#0600）。

MODRW 指令内容说明：

操作数	说明	装置
S1	联机装置地址	1
S2	通讯功能码	16#0006
S3	写入 T0 的装置地址	16#0600
S	数据写入 T0 存放的寄存器	D10
n	写入数据长度（在此功能码无意义）	0

ASCII 模式做通讯：(此处 ASCII 码不刻意转换表示，皆以 16#数值表示)

- AS 传送通讯命令：“：01 06 06 00 55 AA F4 CR LF”
- AS 接收通讯命令：“：01 06 06 00 55 AA F4 CR LF”

RTU 模式做通讯：

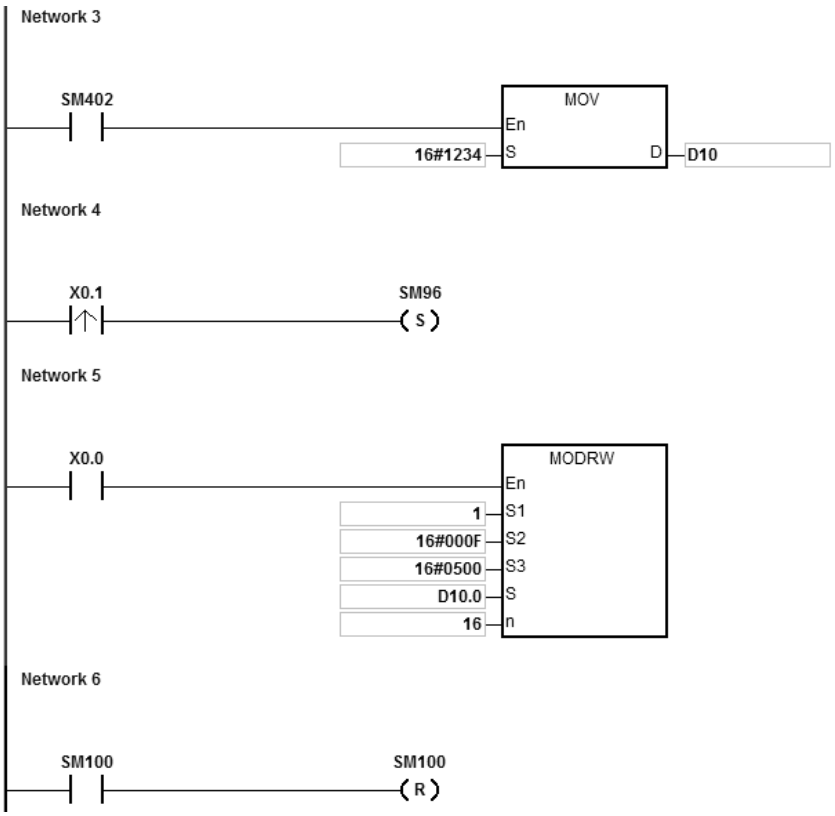
- AS 传送通讯命令：“01 06 06 00 55 AA 36 6D”
- AS 接收通讯命令：“01 06 06 00 55 AA 36 6D”

因为数据正确无误，所以 SM100=ON。

4. DVP-ES2 所回传的数据接收完毕后，会对 DVP-ES2 所回传的数据格式进行确认，是否符合数据格式，如果正确无误则 SM100=ON，如果回传数据格式错误则 SM102=ON。
5. 当 DVP-ES2 接收到此通讯命令的时候，会对 T0 写入存在 AS D10 装置内的数据。
6. 此功能码为写入功能，所以于本指令中的 n 值，并不会被使用到。

程序范例五：

1. 功能码 0F (16#0F) : 多笔位装置状态写入命令。



2. AS 与 DVP-ES2 联机：

- 给定 D10.15~D10.0=16#04D2 (准备写入 DVP-ES2 Y0~Y17 的状态数据)

装置	D10.7	D10.6	D10.5	D10.4	D10.3	D10.2	D10.1	D10.0
状态	ON	ON	OFF	ON	OFF	OFF	ON	OFF
值	D				2			

装置	D10.15	D10.14	D10.13	D10.12	D10.11	D10.10	D10.9	D10.8
状态	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
值	0				4			

当 SM96 为 ON 且 X0.0 也为 ON 的状态时，将会对 DVP-ES2 的 Y0~Y17 写入数据(Y0 的地址为 16#0500)。

MODRW 指令内容说明：

操作数	说明	装置
S1	联机装置地址	1
S2	通讯功能码	16#000F
S3	写入 Y0 的装置地址	16#0500
S	数据写入 Y0~Y17 存放的寄存器	D10.0
n	写入数据长度	16

ASCII 模式做通讯：(此处 ASCII 码不刻意转换表示，皆以 16#数值表示)

- AS 传送通讯命令：“：01 0F 0500 0010 02 D2 04 03 CR LF”
- AS 接收通讯命令：“：01 0F A0 00 00 10 40 CR LF”

RTU 模式做通讯：

- AS 传送通讯命令：“01 0F 05 00 00 10 02 D2 04 EA 43”
- AS 接收通讯命令：“01 0F A0 00 00 10 76 07”

因为数据正确无误，所以 SM100=ON。

3. DVP-ES2 所回传的数据接收完毕后，会对 DVP-ES2 所回传的数据格式进行确认，是否符合数据格式，如果正确无误则 SM100=ON，如果回传数据格式错误则 SM102=ON。

   当 DVP-ES2 接收到此通讯命令的时候，会对 Y0~Y17 写入存在 AS D10.0~D10.15 装置内的数据。

   DVP-ES2 中 Y0~Y17 的状态如下：

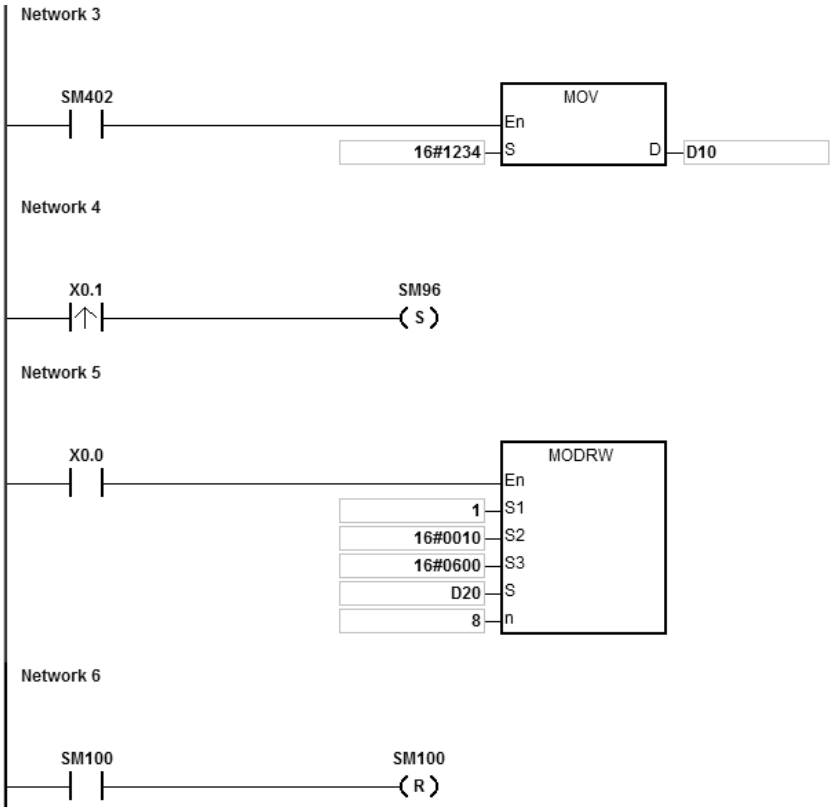
装置	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
状态	ON	ON	OFF	ON	OFF	OFF	ON	OFF
值	D				2			

装置	Y17	Y16	Y15	Y14	Y13	Y12	Y11	Y10
状态	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
值	0				4			

4. 此功能码为写入功能，所以于本指令中的 n 值，并不会被使用到。

程序范例六：

1. 功能码 10（16#10）: 多笔字符装置数据写入命令。



2. AS 与 DVP-ES2 联机：

   给定 D20~D27 的数据内容如下：(准备写入 DVP-ES2 T0~T7 的数据内容)

装置	D20	D21	D22	D23	D24	D25	D26	D27
值（16#）	1234	5678	1122	3344	5566	7788	99AA	BBCC

   当 SM96 为 ON 且 X0.0 也为 ON 的状态时，将会对 DVP-ES2 的 T0~T7 写入数据(T0 的地址为 16#0600)。

   MODRW 指令内容说明：

操作数	说明	装置
S1	联机装置地址	1
S2	通讯功能码	16#0010
S3	写入 T0 的装置地址	16#0600
S	数据写入 T0~T7 存放的寄存器	D20
n	写入数据长度	8

   ASCII 模式做通讯：(此处 ASCII 码不刻意转换表示，皆以 16#数值表示)

   - AS 传送通讯命令：“ :01 10 0600 00 08 10 1234 5678 1122 3344 5566 7788 99AA BBCC 8F CR LF”
   - AS 接收通讯命令：“：01 10 06 00 00 08 E1 CR LF”

RTU 模式做通讯：

- AS 传送通讯命令：“01 10 06 00 00 08 10 1234 5678 1122 3344 5566 7788 99AA BBCC 0B 0C”
- AS 接收通讯命令：“01 10 06 00 00 08 C1 47”

因为数据正确无误，所以 SM100=ON。

3. DVP-ES2 所回传的数据接收完毕后，会对 DVP-ES2 所回传的数据格式进行确认，是否符合数据格式，如果正确无误则 SM100=ON，如果回传数据格式错误则 SM102=ON。

   当 DVP-ES2 接收到此通讯命令时，会对 T0~T7 写入 AS D20~D27 装置内的数据。

   DVP-ES2 中 T0~T7 的内容值：

装置	T0	T1	T2	T3	T4	T5	T6	T7
值（16#）	1234	5678	1122	3344	5566	7788	99AA	BBCC

4. 此功能码为写入功能，所以于本指令中的 n 值，并不会被使用到。

补充说明：

1. **S_1 或 S_2** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
2. **S** 装置不足以写入或读取 **n** 所指定的长度时，指令不执行，SM0=ON，SR0 错误码为 16#2003。
3. **n** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#200B。
4. 当 **S_2** 通讯功能码指定为 Bit 读写时 S 必须指定为 Bit 装置，否则视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
5. 当 **S_2** 通讯功能码指定为字符装置读写时 **S** 必须指定为字符装置，否则视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
6. 当通讯命令为 0x05 与 0x06 时，n 无作用，指令只会写入 1 个 Bit 或 Word。
7. 若执行 MODRW 的时候，送信标志都没有 ON 起来的话，则不执行。
8. 当发生通讯逾时（Timeout）时，通讯逾时标志会被设定为 ON，接收中标志会被清除为 OFF。
9. 当接收指令有错误（Error）时，错误标志会被设定为 ON，接收中标志会被设定为 OFF。
10. 当 **S_2** 为读写 Word 装置，则读写对方设备装置也必须是 Word 装置，当 **S_2** 为读写位装置，则读写对方设备装置也必须为位装置。
11. 相关 SM 标志与 SR 特殊寄存器请参考第 6.19.3 节说明

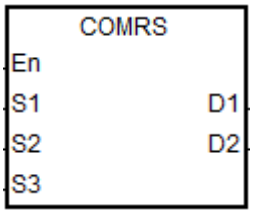
API	指令码	操作数	功能
1812	COMRS	S_1 · S_2 · S_3 · D_1 · D_2	自定义义串行通讯传送与接收指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●							
S_3								●	●				○	○		
D_1								●								
D_2								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
D_1		●				●							
D_2		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 :通讯端口编号（1~2，11~12）

S_2 :传送数据源

S_3 :传送长度设定值

D_1 :接收通讯数据的起始装置组件

D_2 接收数据结束条件参数设定

指令说明：

1. S_1 为通讯端口编号，COM1 编号为 1，COM2 编号为 2，Card1 编号为 11，Card2 编号为 12。若超出通讯端口范围时，则指令将不会执行任何通讯数据接收。
2. 当接收结束条件有指定特定字符时，建议使用于 ASCII 编码之通讯数据，若非 ASCII 编码之通讯数据，则建议改用通讯逾时时间做为结束条件。
3. S_2 通讯数据传送来源与 S_3 传送长度设定值，其范例说明如下：假设 S_2 选择 D100 及 S_3 为 10，并设定为 8 位模式，则指令将会传送 D100~D109 的下 8 位数值共 10 个字，由 S_1 指定的通讯端口传送出去。
4. 当 S_3 传送长度设定值为 0 时，即表示不需要传送字符串；传送长度最大值为 256 个字符。

5. **D_1** 为已接收数据长度值，**D_1+1~ D_1+n** 为存放接收数据地址，说明如下：假设 **D_1** 设为 D200，**D_2**=3 及 **D_2**+1 设定终止符 16#0D0A，并设定为 8 位模式，则指令将会依照接收数据顺序存放于 D201 之后的下 8 位中（上 8 位不变化），直到两个连续结束位为 16#0D 与 16#0A 时，即停止接收数据，并将接收长度填入 D200 内，接着设定完成标志。

6. **D_2**+0 为接收模式参数设定，**D_2**+1 为接收条件参数设定，其详细说明如下表所示：

D_2	**D_2** 接收模式说明	**D_2**+1 参数内容说明	备注
0	不接收通讯数据	不使用	即传送完成后，设定完成标志。
1	通讯数据不连续时间超出 **D_2**+1 设定值时，即为接收完成。	时间长度设定值，时间单位为 1ms，设定值范围为 5~3000。	设定值超出 3000 时，其值为 3000；小于 5 时，其值为 5。
2	接收数据有一个特定字符为结尾字符	特定结尾字符设定值	如特定字符为 16#0A，则 **D_2**+1 设定为 16#000A
3	接收数据有两个连续特定字符为结尾字符	两个特定结尾字符设定值	如两个特定字符顺序为 16#0D 与 16#0A，则 **D_2**+1 设定为 16#0D0A
4	接收数据有一个特定字符为开始字符，以及不连续时间超出 **D_2**+1 设定值时，即为接收完成。	上 8 位为特定开始字符，下 8 位为不连续时间设定值（范围 5~255ms）。	如开始字符为 16#3A 以及不连续时间 15ms，则 **D_2**+1 设定为 16#3A0F
5	接收数据有一个特定字符为开始字符与一个结尾字符	特定开始与结尾字符设定值	如开始字符为 16#3A，结尾字符为 16#0A，则 **D_2**+1 设定为 16#3A0A
6	接收数据有一个特定长度为接收完成	**D_2**+1 为接收长度设定值，设定值范围为 1~256。	如接收 10 个字符，则 **D_2**+1 设定为 10
7	接收数据有一个特定字符为结尾字符，并且产生通讯中断	特定结尾字符设定值	如特定字符为 16#0A，则 **D_2**+1 设定为 16#000A
8	接收数据有一个特定长度为接收完成，并且产出通讯中断	**D_2**+1 为接收长度设定值，设定值范围为 1~256。	如接收 10 个字符，则 **D_2**+1 设定为 10
9	接收数据有一个特定字符为结尾字符或是有一个特定长度为接收完成（其中一个条件到达，即接收完成）	上 8 位为特定结尾字符设定值，下 8 位为接收长度设定值，设定值范围为 1~255。	如结尾字符为 16#0A 以及接收长度为 15 个字符，则 **D_2**+1 设定为 16#0A0F
其它	超出现有支持模式范围，则指令将不会被启动。		

7. **当 D_2** 的接收模式中，除了模式 6,8（指定特定长度为接收完成）外，其它模式接收到达最大接收长度 256 字符，而未完成接收命令时，指令自动结束，并视为接收错误，此时 **D_1**+0=0，已接收的数据不会填入 **D_1**+1~。

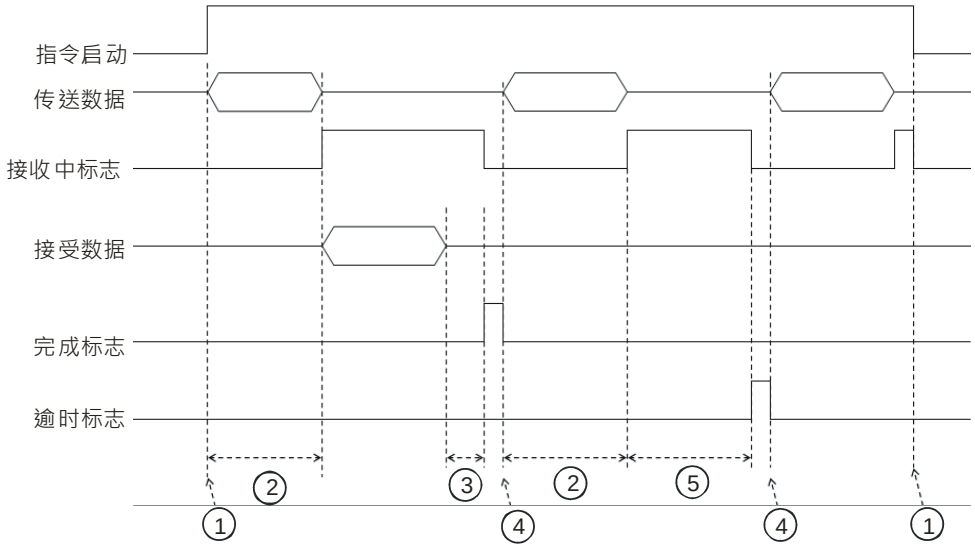
8. 通讯端口对应的特殊标志 SM 与特殊寄存器 SR，请参考第 6.19.3 节。

9. 通讯传送与接收时序图如下:

- 接收模式为 0 时

  需注意数据传送时无法中途取消，此时即使指令启动条件不成立，仍会继续传送数据，惟传送完成后不会将完成标志设为 ON。

- 接收模式为 1 或 4 时



图标符号说明：

① → 由用户设定启动/关闭

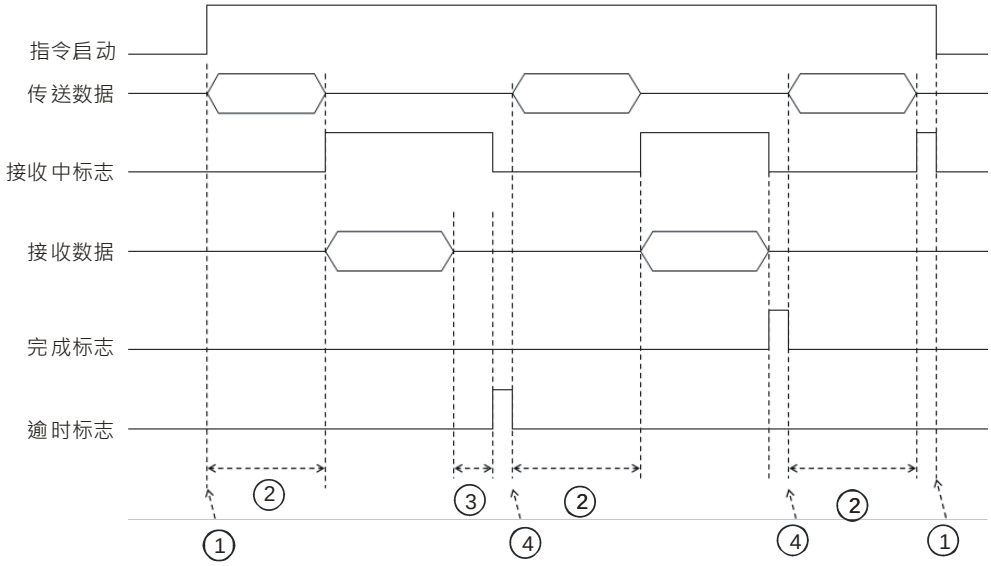
② → 传送数据时间，此期间通讯逾时不会被计时。

③ → 接收到第 1 个字符时，即开始累积计时结束时间，每接收一个字符即清除累积计时值一次，直到累积计时超出 D_2+1 设定值之后，则产生通讯完成标志。

④ → 由用户清除完成或逾时标志，若指令还处于开启状态，则下个周期扫描到指令时，将自动重新传送下一笔通讯数据。

⑤ → 当接收开始时即开始累积计时，直到累积计时超出通讯逾时设定值之后，则产生通讯逾时标志。建议通讯逾时时间最好设定比 D_2+1 结束时间长。

- 接收模式为 2、3、5、6 或 9 时



图标符号说明：

① → 由用户设定启动/关闭

② → 传送数据时间，此期间通讯逾时不会被计时。

③ → 接收开始时即开始累积计时，每接收一个字符即清除累积计时值一次，直到累积计时超出通讯逾时设定值之后，则产生通讯逾时标志。

④ → 由用户清除完成或逾时标志，若指令还处于开启状态，则下个周期扫描到指令时，将自动重新传送下一笔通讯数据。

10. 数据传送/接收模式：

《8 位模式》：用户将编辑好的命令储存在传送装置起始地址上，传送出去的命令包含头后缀。将 16 位数据分成上 8 位，下 8 位，上 8 位被省略，仅下 8 位为有效数据可做数据的传送和接收（以标准 MODBUS 为例）

传送数据：(PLC→外部机器)

D10下	D11下	D12下	D13下	D14下	D15下	D16下
头码	来源数据寄存器由D10下 8位元开始 长度=7				尾码 1	尾码 2

接收数据：（外部机器→PLC）

D100下	D101下	D102下	D103下	D104下	D105下	D106下
头码	来源数据寄存器由D100下 8位开始 长度=7				尾码 1	尾码 2

《16 位模式》：用户将编辑好的命令储存在传送装置起始地址上，传送出去的命令包含头后缀。当 SM106/SM107=OFF 时，指定为 16 位转换模式，将 16 位数据分成上 8 位与下 8 位做数据的传送和接收。

传送数据：(PLC→外部机器)

D10下	D10上	D11下	D11上	D12下	D12上	D13下
头码	来源数据寄存器由D10下 8位开始 长度=7				尾码 1	尾码 2

接收数据：（外部机器→PLC）

D100下	D100上	D101下	D101上	D102下	D102上	D103下
头码	来源数据寄存器由D100下 8位开始 长度=7				尾码 1	尾码 2

PLC 接收数据会将外部机器传入数据报含头码、后缀一起接收，所以长度之设定要注意。

11. 接收模式为 7 与 8 时，各通讯端口所产生之对应通讯中断如下表所示：

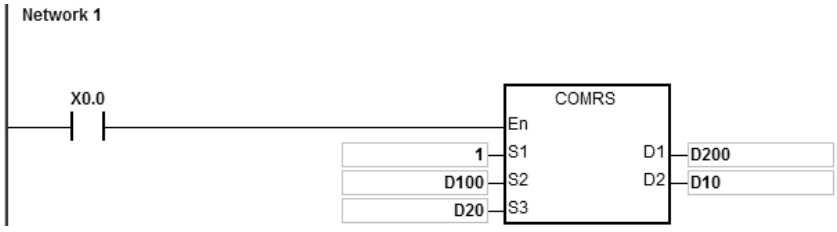
通讯端口编号	COM1	COM2	Card1	Card2
中断编号	I300	I302	I304	I306

12. 此指令不可使用于 ST 结构化语言编辑、中断程序或仅被调用一次性的功能块（FB）程序。

以下程序范例以 AS 的 COM1（RS485）说明：

程序范例 1：

接收模式参数 K0（不接收通讯数据）且设定为 8 位处理模式（SM106=ON）

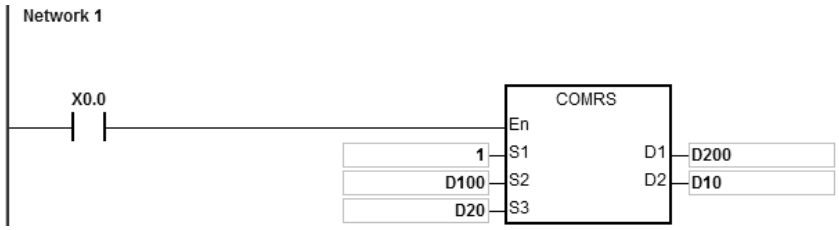


1. 设定要传送的数据长度 D20=4
2. 设定要传送的数据内容 D100=16#0031，D101=16#0032，D102=16#0033，D103=16#0034
3. 设定 D10=16#0000（此参数设定值仅传送数据，不接收数据）
4. 启动条件接点 X0.0
5. PLC 发送出 4 笔数据

   PLC 传送数据→外部机器 31 32 33 34
6. 因为不接收数据，所以 PLC 传送完毕就结束执行，SM100=ON（接收完毕）
7. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，即可再次执行。

程序范例 2：

接收模式参数 1（设定不连续时间 5~3000ms）且设定为 16 位处理模式（SM106=OFF）



1. 设定要传送的数据长度 D20=4
2. 设定要传送的数据内容 D100=16#3231、D101=16#3433
3. 设定 D10=16#0001（模式参数 1），D11=300（不连续时间设定值 300ms）
4. 启动条件接点 X0.0
5. PLC 发送出 4 笔数据

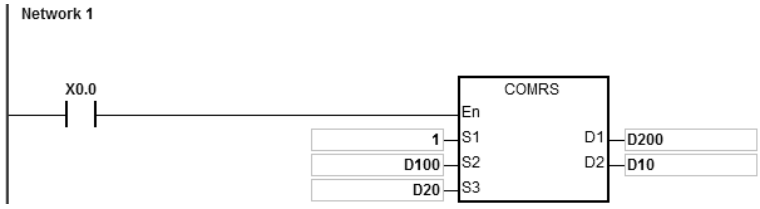
   PLC 传送数据→外部机器 31 32 33 34
6. 外部机器收到 PLC 传送的数据之后，依序发送 5 笔连续数据给 PLC 且每字间隔不超过 20 ms

   外部机器→PLC 35 36 37 38 39
7. 读取 D200=5（接收数据数），接收数据内容为 D201=16#3635、D202=16#3837、D203=16#0039
8. 读取 SM100=ON（接收完毕）
9. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，即可再次执行。

注：当数据传送完毕时，接收中标志 SM98 为 ON，并进入接收状态，准备开始接收数据，当接收到第一个字符时开始累积计时结束时间，每收到一个字符即清除累积计时值，直到累积计时超过 D11 的设定值，完成标志 SM100 为 ON。

程序范例 3：

接收模式参数 2（特定字符为结尾字符）且设定为 8 位处理模式（SM106=ON）



1. 设定要传送的数据长度 D20=0，表示 PLC 不传送数据直接进入接收阶段
2. 设定 D10=16#0002（模式参数 2），D11=16#000A（表示结尾字符 16#0A）
3. 启动条件接点 X0.0
4. PLC 等待接收外部机器回传的数据内容（D20=0 不传送数据给外部机器）
5. 外部机器发送数据给 PLC

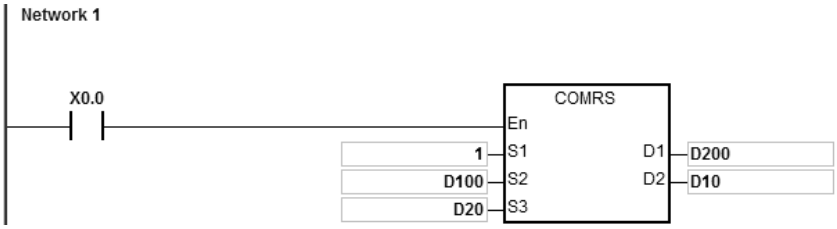
   外部机器→PLC　31 32 33 34 35 0A
6. 读取 D200=6（接收数据数），接收数据内容为 D201=16#0031、D202=16#0032、D203=16#0033、D201=16#0034、D202=16#0035、D203=16#000A
7. 读取 SM100=ON（接收完毕）
8. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，即可再次执行。

注：当 PLC 进入接收状态，接收中标志 SM98 为 ON，开始接收数据到数据后，直至收到设定的特定结尾字符（16#0A）为止，若完成接收则完成标志 SM100 为 ON（若超过通讯逾时时间仍未收到特定结尾字符，则通讯逾时标志 SM104 为 ON）。

程序范例 4：

接收模式参数 3（两个特定字符为结尾字符）且设定为 16 位处理模式（SM106=OFF）

假设外部机器为 DVP-ES2，并要对 DVP-ES2 的 D100 写入 H1234



1. 设定要传送的数据长度 D20=17
2. 设定要传送的数据内容 D100=16#303A、D101=16#3031、D102=16#3136、D103=16#3630、D104=16#3134、D105=16#3332、D106=16#3334、D107=16#0D46、D108=16#000A
3. 给定 D10=16#0003（模式参数 3），D11=16#0D0A（表示结尾字符 16#0D 与 16#0A）
4. 启动条件接点 X0.0
5. PLC 发送出 17 笔数据

   PLC 传送数据→外部机器 3A 30 31 30 36 31 30 36 34 31 32 33 34 33 46 0D 0A

   （ASCII 为：0106106412343FCRLF）
6. 外部机器收到 PLC 传送的数据后，发送 18 笔数据给 PLC 且最后 2 笔是终止符 16#0D 16#0A

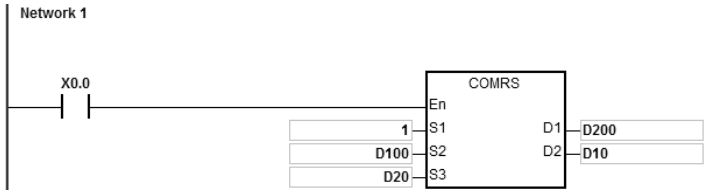
   外部机器→PLC　3A 30 31 30 36 31 30 36 34 31 32 33 34 33 46 0D 0A

   （ASCII 为：0106106412343FCRLF）
7. 读取 D200=17（接收数据数），接收数据内容为 D201=16#303A、D202=16#3031、D203=16#3136、D204=16#3630、D205=16#3134、D206=16#3332、D207=16#3334、D208=16#0D46、D209=16#000A
8. 读取 SM100=ON（接收完毕）
9. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，即可再次执行。

注：当数据传送完毕时，接收中标志 SM98 为 ON，并进入接收状态，当开始接收数据后，直至收到设定的两个特定结尾字符 16#0D0A 为止，若完成接收则完成标志 SM100 为 ON（若超过通讯逾时时间仍未收到特定结尾字符，则通讯逾时标志 SM104 为 ON）

程序范例 5：

接收模式参数 4（特定字符为开始字符以及不连续时间设定 5~255ms）且设定为 8 位处理模式（SM106=ON）



1. 设定要传送的数据长度 D20=4
2. 设定要传送的数据内容 D100=16#0031、D101=16#0032、D102=16#0033、D103=16#0034
3. 设定 D10=16#0004（模式参数 4）、D11=16#3A0F（开始字符为 16#3A、不连续时间设定为 16#0F 即 15ms）
4. 启动条件接点 X0.0
5. PLC 发送出 4 笔数据

   PLC 传送数据→外部机器　31 32 33 34
6. 外部机器收到 PLC 传送的数据后，依序发送 7 个字符连续数据给 PLC 且每个字间隔为 1 ms

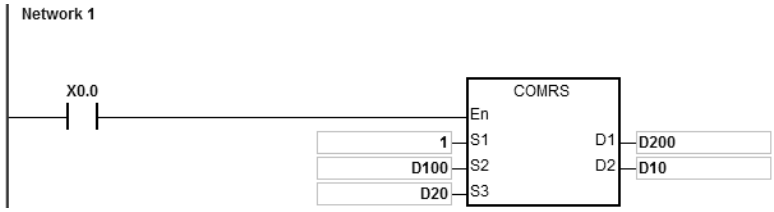
   外部机器→PLC　30 3A 35 36 37 38 39
7. 读取 D200=6（接收数据数）、接收数据内容为 D201=16#003A、D202=16#0035、D203=16#0036、D204=16#0037、D205=16#0038、D206=16#0039
8. 读取 SM100=ON（接收完毕）
9. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，即可再次执行。

注：当数据传送完毕时，接收中标志 SM98 为 ON，并进入接收状态，当接收到特定开始字符（16#3A）开始接收数据并开始累积计时结束时间，收到一个字符即清除累积计时值，直到累积计时超过 D11 的设定值 16#0F（15ms），完成标志 SM100 为 ON。

程序范例 6：

接收模式参数 5（特定字符为开始及结尾字符）且设定为 16 位处理模式（SM106=OFF）

假设外部机器为 DVP-ES2，并要对 DVP-ES2 读取 D100 的内容值



1. 设定要传送的数据长度 D20=17
2. 设定要传送的数据内容 D100=16#303A、D101=16#3031、D102=16#3133、D103=16#3630、D104=16#3034、D105=16#3030、D106=16#3831、D107=16#0D37、D108=16#000A
3. 设定 D10=16#0005（模式参数 5）、D11=16#3A0A（开始字符为 16#3A、终止符为 16#0A）
4. 启动条件接点 X0.0

5. PLC 发送出 17 笔数据

   PLC 传送数据→外部机器 3A 30 31 30 33 31 30 36 34 30 30 30 31 38 37 0D 0A

   （ASCII 为：01031064000187CRLF）

6. 外部机器收到 PLC 传送的数据后，依序发送 15 笔数据给 PLC

   外部机器→PLC 3A 30 31 30 33 30 32 31 32 33 34 42 34 0D 0A

   （ASCII 为：0103021234B4CRLF）

7. 读取 D200=15（接收数据数），接收数据内容为 D201=16#303A、D202=16#3031、D203=16#3033、D204=16#3132、D205=16#3332、D206=16#4234、D207=16#0D34、D208=16#000A

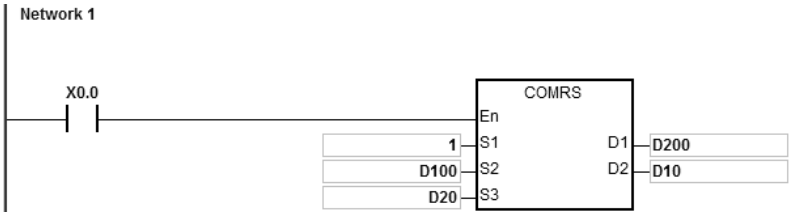
8. 读取 SM100=ON（接收完毕）

9. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，就会自动再次执行。

注：当数据传送完毕时，接收中标志 SM98 为 ON，并进入接收状态，当接收到特定开始字符（16#3A）开始接收数据，直至收到设定的特定结尾字符 16#0A 为止，完成标志 SM100 为 ON。(若超过通讯逾时时间仍未收到特定开始字符（16#3A）或特定结尾字符（16#0A），则通讯逾时标志 SM104 为 ON）

程序范例 7：

接收模式参数 6（接收特定长度数据）且设定为 8 位处理模式（SM106=ON）



1. 设定要传送的数据长度 D20=4

2. 设定要传送的数据内容 D100=16#0031、D101=16#0032、D102=16#0033、D103=16#0034

3. 设定 D10=16#0006（模式参数 6），D11=16#0008（接收 8 笔数据）

4. 启动条件接点 X0.0

5. PLC 发送出 4 笔数据

   PLC 传送数据→外部机器 31 32 33 34

6. 外部机器收到 PLC 传送的数据后，依序发送 8 笔数据给 PLC

   外部机器→PLC 32 33 34 35 36 37 38 39

7. 读取 D200=8（接收数据数），接收数据内容为 D201=16#0032、D202=16#0033、D203=16#0034、D204=16#0035、D205=16#0036、D206=16#0037、D207=16#0038、D208=16#0039

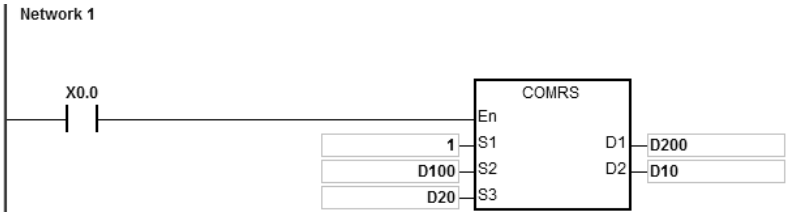
8. 读取 SM100=ON（接收完毕）

9. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，即可再次执行。

注：当数据传送完毕时，接收中标志 SM98 为 ON，并进入接收状态，直至收到设定的长度数据（8 笔）时为止，完成标志 SM100 为 ON。(若超过通讯逾时时间仍未收到设定的长度数据，则通讯逾时标志 SM104 为 ON）

程序范例 8：

接收模式参数 7（接收数据有一个特定字符为结尾字符，并且产生通讯中断）且设定为 8 位处理模式（SM106=ON）



通讯中断程序内容：

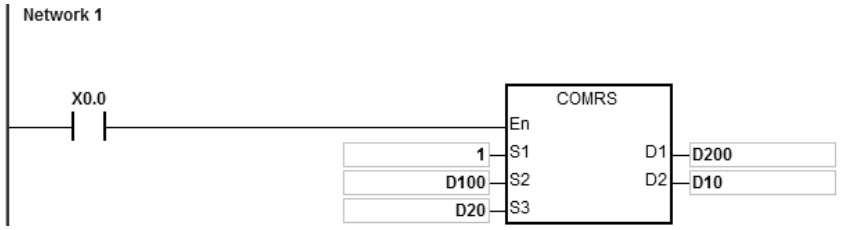
1. 清除中断内的装置 D30=0
2. 设定要传送的数据长度 D20=4
3. 设定要传送的数据内容 D100=16#0031、D101=16#0032、D102=16#0033、D103=16#0034
4. 设定 D10=16#0007（模式参数 7），D11=16#000A（表示结尾字符 16#0A）
5. 启动条件接点 X0.0
6. PLC 发送出 4 笔数据

   PLC 传送数据→外部机器 31 32 33 34
7. 读取 D30=0（中断内的程序尚未被触发执行）
8. 外部机器发送数据给 PLC
9. 外部机器→PLC 31 32 33 34 35 0A
10. 读取 D200=6（接收数据数），接收数据内容为 D201=16#0031、D202=16#0032、D203=16#0033、D201=16#0034、D202=16#0035、D203=16#000A
11. 读取 SM100=ON（接收完毕）
12. 读取 D30=1（中断被触发，因此执行 INC D30）
13. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，即可再次执行。

注：当 PLC 进入接收状态，接收中标志 SM98 为 ON，开始接收数据到数据后，直至收到设定的特定结尾字符（16#0A）为止，若完成接收则完成标志 SM100 为 ON（若超过通讯逾时时间仍未收到特定结尾字符，则通讯逾时标志 SM104 为 ON）

程序范例 9：

接收模式参数 8（接收数据有一个特定长度为接收完成，并且产出通讯中断）且设定为 8 位处理模式（SM106=ON）



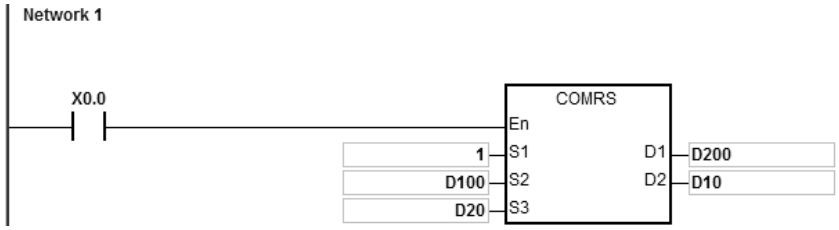
通讯中断程序内容：

1. 清除中断内的装置 D30=0
2. 设定要传送的数据长度 D20=4
3. 设定要传送的数据内容 D100=16#0031、D101=16#0032、D102=16#0033、D103=16#0034
4. 设定 D10=16#0008（模式参数 8），D11=16#0008（接收 8 笔数据）
5. 启动条件接点 X0.0
6. PLC 发送出 4 笔数据
   PLC 传送数据→外部机器 31 32 33 34
7. 读取 D30=0（中断内的程序尚未被触发执行）
8. 外部机器收到 PLC 传送的数据后，依序发送 8 笔数据给 PLC
   外部机器→PLC 32 33 34 35 36 37 38 39
9. 读取 D200=8（接收数据数），接收数据内容为 D201=16#0032、D202=16#0033、D203=16#0034、D204=16#0035、D205=16#0036、D206=16#0037、D207=16#0038、D208=16#0039
10. 读取 SM100=ON（接收完毕）
11. 读取 D30=1（因中断已被触发，因此执行 INC D30）
12. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，即可再次执行。

注：当数据传送完毕时，接收中标志 SM98 为 ON，并进入接收状态，直至收到设定的长度数据（8 笔）时为止，完成标志 SM100 为 ON。（若超过通讯逾时时间仍未收到设定的长度数据，则通讯逾时标志 SM104 为 ON）

程序范例 10：

接收模式参数 9（接收数据有一个特定结尾字符或是达到特定长度，即接收完成）且设定为 8 位处理模式（SM106=ON）



1. 设定要传送的数据长度 D20=4
2. 设定要传送的数据内容 D100=16#0031、D101=16#0032、D102=16#0033、D103=16#0034
3. 设定 D10=16#0009（模式参数 9），D11=16#0A0F（终止符为 16#0A，并设定特定长度为 16#0F）
4. 启动条件接点 X0.0
5. PLC 发送出 4 笔数据

   PLC 传送数据→外部机器 31 32 33 34
6. 外部机器收到 PLC 传送的数据后，依序发送 15 笔数据给 PLC

   外部机器→PLC 31 32 33 34 35 0A 41 42 43 44 45 46 47 48 49
7. 读取 D200=6（接收数据数），接收数据内容为 D201=16#0031、D202=16#0032、D203=16#0033、D204=16#0034、D205=16#0035、D206=16#000A

   因为 PLC 收到第 6 笔数据就收到终止符（16#0A），因此结束接收。
8. 读取 SM100=ON（接收完毕）
9. 如欲再次传送，需手动将完成标志 SM100 设为 OFF，就会自动再次执行。

注：当数据传送完毕时，接收中标志 SM98 为 ON，进入接收状态，开始接收数据后，直至收到设定的特定结尾字符（16#0A）或收到设定的长度数据（15 笔）为止，完成标志 SM100 为 ON。（若超过通讯逾时时间仍未达到结束接收的条件，则通讯逾时标志 SM104 为 ON）

补充说明：

1. 此 COMRS 通讯指令无使用次数限制，但每个通讯端口都仅能被一条通讯指令启动，因此后面启动之通讯指令将不会被执行。
2. 此 COMRS 指令无检查码机制之设计，须此功能者可搭配现有提供之便利指令自行组合运用。
3. D2 内容值为 2、3、5、6、9 时，建议设定通讯逾时。若有设定通讯逾时，当指令没有收到终止符时，才能重新进行重传机制。
4. 每次刚启动指令或重新开始接收新的下一笔通讯数据时，此指令并不会自动清除 $\mathbf{D_1}$+0 ~ $\mathbf{D_1}$+n 装置内的数值，相关的接收数据信息（如接收数据是否有接收？与接收长度为何？）皆以完成标志由 OFF 变为 ON 时为基准。若需清除这些接收数据区域时，建议可搭配 ZRST 指令清除。
5. 当 $\mathbf{S_1}$ 的内容值超出范围时，指令不执行。
6. $\mathbf{S_2}$ 装置不足以读取 $\mathbf{S_3}$ 所指定的长度时，指令不执行，SM0=ON，错误码为 SR0=16#2003。
7. 当 $\mathbf{D_2}$ 的接收模式设定超出范围时（0~9），指令不执行，SM0=ON，SR0 错误码为 16#200B。

8. 当 **D_2** 的接收模式设定为 6 或 8 或 9，**D_1** 装置不足以写入 **D_2**+1 所指定的长度时，指令不执行，SM0=ON，SR0 错误码为 16#2003。
9. 当接收数据超出 **D_1** 最大装置范围，则超出装置范围的接收数据将会被忽略。
10. 当接收完成标志变为 ON 时，其通讯数据将暂停接收，若此期间通讯端口还有数据进入，将不再接收。
11. 当 **S_3** 传送长度设定值为小于 0 或大于 256 时，指令不执行，SM0=ON，SR0 错误码为 16#200B。
12. 当 **D_2** 的接收模式设定为 6 或 8，**D_2**+1 所指定的长度小于 1 或大于 256 时，指令不执行，SM0=ON，SR0 错误码为 16#200B。

API	指令码			操作数	功能
1813		COMDF	P	**S_1 · S_2 · S_3 · S_4 · S_5 · D**	串行通讯端口通讯格式设定指令

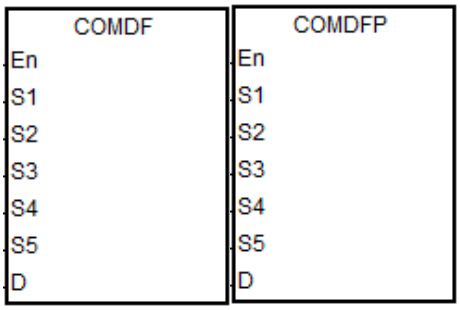
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1													○	○		
S_2													○	○		
S_3													○	○		
S_4													○	○		
S_5													○	○		
D													○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													
S_2													
S_3													
S_4													
S_5													
D													

脉冲执行型	16 位指令	32 位指令
AS	AS	-

6

符号：



S_1 ： 通讯速率（Baud Rate）(基本单位：100 bps）

S_2 ： 通讯数据长度（Data bit no.）

S_3 ： 校验码型式（Parity bit）

S_4 ： 结束位长度（End bit no.）

S_5 ： MODBUS 格式选择

D ： 通讯端口编号

指令说明：

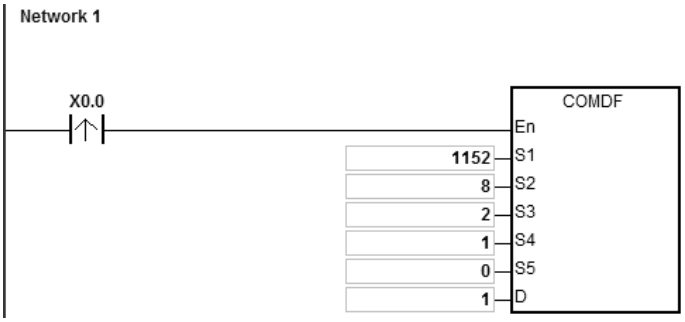
1. 此指令仅提供直接设定参数方式执行，不提供变量宣告方式设定。
2. **S_1** 为通讯速率（Baud Rate）设定，其基本单位为 100bps，例如：输入 96 表示 9600bps。
3. **S_2** 为通讯数据长度设定，其数值 7（预设）表示 7 data bit，8 表示 8 data bit，非此两个数值时，自动以默认值执行。
4. **S_3** 为校验码型式设定，其数值 0 表示无校验码（None），1 表示奇同位（Odd）检查，2（预设）表示偶同位（Even）检查，非此三个数值时，自动以默认值设定。
5. **S_4** 为结束位长度设定，其数值 1（预设）表示 1 bit，2 表示 2 bit，非此两个数值时，自动以默认值执行。

6. **S_5** 为 MODBUS 通讯协议的通讯模式设定，其数值 0（默认）表示 ASCII 模式，1 表示 RTU 模式，非此两个数值时，自动以默认值执行。
7. **D** 为通讯端口编号，COM1 编号为 1，COM2 编号为 2，Card1 编号为 11，Card2 编号为 12。若超出通讯端口范围时，则指令将不会执行任何通讯格式设定。
8. 用户也可利用 ISPSoft - > HWCONFIG - > COM Port 内直接对通讯端口进行设定或是特殊寄存器来设定通讯端口（HWCONFIG 相关设定请参照 ISPSoft 使用手册，通讯相关 SR、SM 寄存器设定可参考第 6.19.3 节）。
9. 使用此指令设定之后，其实际通讯端口将会立即变更通讯，若此时有通讯正在进行，也将会被强制取消；另外其相对应的 SM/SR 设定值将跟着变化，详细 SM/SR 说明请参考下个章节 6.19.3 节。
10. 当通讯格式设定值与前次设定值相同时，指令将不会对实际通讯端口进行任何设定。

通讯协议设定范例：

1. 这里以 PLC COM1 为例做说明，其它 PLC 通讯端口依此类推。
2. 启动条件接点 X0.0。
3. PLC COM1（RS485）通讯格式设定成 115200，8，E，1。
4. PLC COM1（RS485）通讯模式设定成 ASCII。
5. COMDF 指令内容说明：

操作数	说明		内容值
S_1	通讯速率（Baud Rate）	115200 bps	1152
S_2	通讯数据长度（Data bit）	8	8
S_3	校验码型式（Parity bit）	E	2
S_4	结束位长度（End bit）	1	1
S_5	MODBUS 格式选择	ASCII	0
D	通讯端口编号	PLC COM1	1



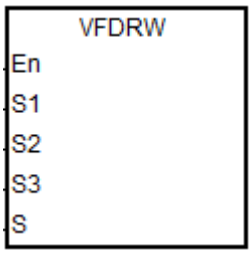
API	指令码		操作数	功能
1814	VFDRW		S_1 · S_2 · S_3 · S	台达变频器专属串行通讯指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●		○	○	○	○		
S_2								●	●		○	○	○	○		
S_3								●	●		○	○	○	○		
S								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 通讯端口编号

S_2 ： 变频器站号

S_3 ： 通讯功能代码

S ： 来源与接收数据

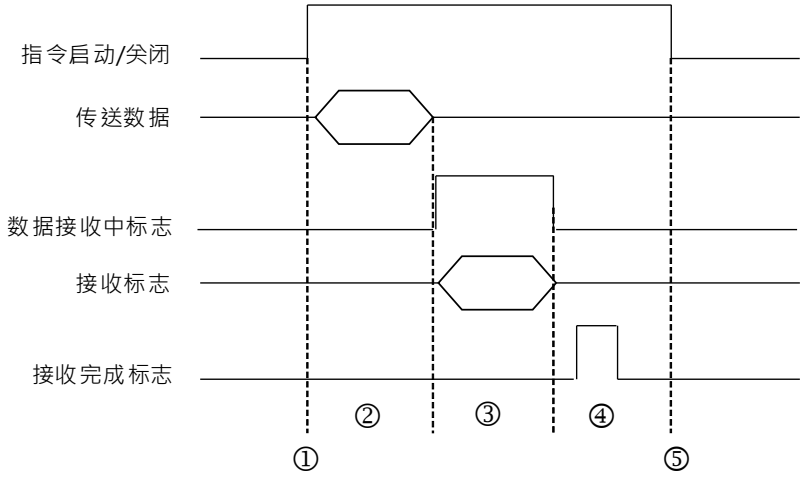
指令说明：

1. **S_1** 为通讯端口编号，COM1 编号为 1，COM2 编号为 2，Card1 编号为 11，Card2 编号为 12。若超出通讯端口范围时，则指令将不会执行任何通讯数据接收。
2. **S_2** 为变频器站号，当输入站号为 0 时，则为广播模式。数值范围为 0~254。超过此范围指令不执行。
3. **S_3** 为通讯功能代码与 S 来源或接收数据说明如下表所示：

S_3 代码	S_3 功能名称	S 来源或接收数据说明	备注
0	异常重置（Reset）	无作用	可输入任意数值
1	正转命令	速度值	设定值与数值单位 ,请参考交流伺服电机驱动器使用手册
2	反转命令	速度值	
3	停止	无作用	可输入任意数值
4	寸动（JOG）正转命令	无作用	设定寸动速度 ,请参考交流伺服电机驱动器使用手册
5	寸动（JOG）反转命令	无作用	

S_3 代码	S_3 功能名称	S 来源或接收数据说明	备注
6	状态读取	接收状态值	读取 VFD 的 H2100 ~ H2104 这 5 个地址状态数值表示意义 ,请参考交流伺服电机驱动器使用手册

4. 通讯传送与接收时序图如下：



图标符号说明：

① → 由用户设定启动/关闭

② → 指令启动后开始传送数据，此期间通讯逾时不会被计时。

③ → 设定接收中标志，并从接收到第 1 个字符时，即开始累积计时，每接收一个字符即清除累积计时值一次，若是累积计时超出通讯逾时设定值，则产生通讯逾时标志。

④ → 接收完成时设定接收完成标志，用户须自行清除此标志。

⑤ → 监看到完成标志后，若需要再次发送通讯指令，则需关闭指令一个周期，然后在下一次周期开启指令。

5. 此通讯指令无使用次数限制，但是每个通讯端口每次只能被一条通讯指令输出执行，若是已完成传送接收时，需自行关闭通讯指令，将通讯控制权释放。

6. 此指令不可使用于 ST 结构化语言编辑、中断程序或仅被调用一次性的功能块（FB）程序

通讯协议设定范例：

1. 使用 HWCONFIG 将 PLC COM1（RS485），站号为 2，通讯格式设定成 ASCII，115200，7，N，2

2. 使用台达变频 C2000 面板进行基本设定，步骤如下：

A. 将 09-00 设置为 1，将变频站号设定为 1

B. 将 09-01 设置为 115.2，将变频 RS485 通讯速率设为 115200

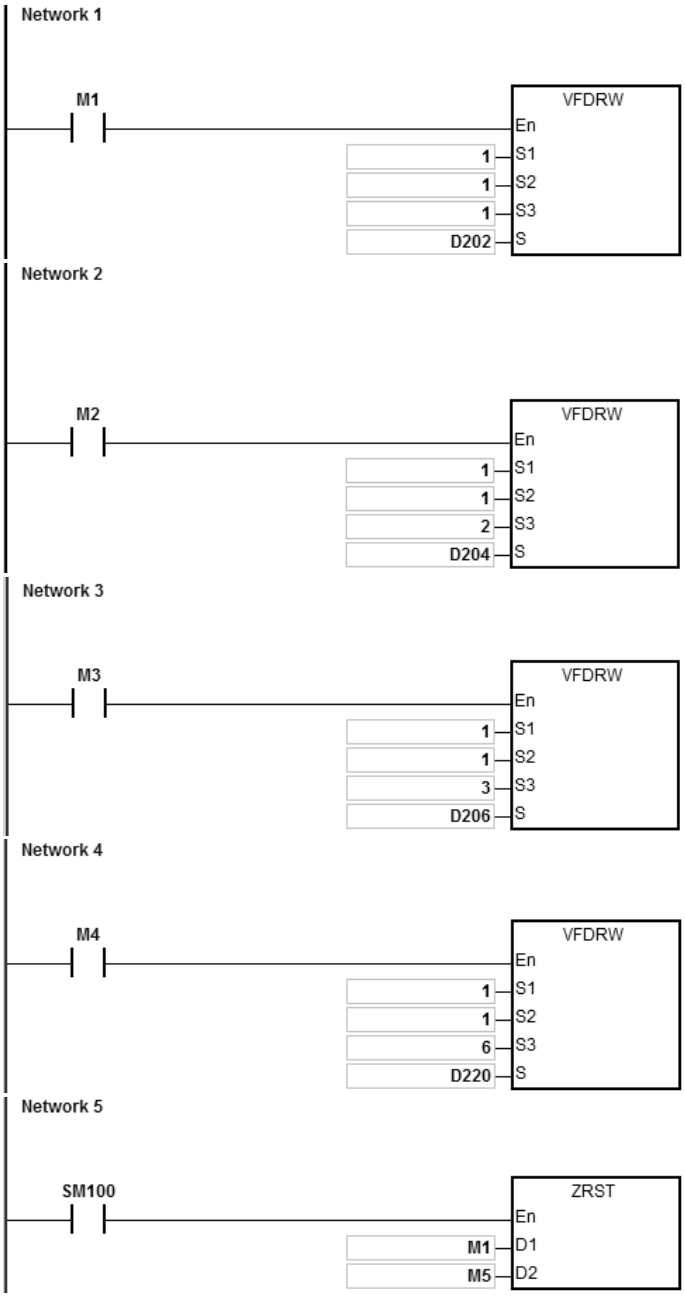
C. 将 09-04 设置为 1，将变频 RS485 通讯格式设为 7，N，2

D. 将 00-20 设置为 1，频率指令由 RS485 输入

E. 将 00-21 设置为 2，运转指令由 RS485 输入

程序范例：

使用 VFDRW 指令，做速度控制，让变频器 VFD 以频率 120Hz 正转；以频率 180Hz 反转，然后停止运转



1. AS 主机与 VFD 联机：
   先给定 D202=12000，当 M1 为 ON 的状态时，VFD 接收到正转命令后，开始加速，持续以频率 120Hz，正转方向运转。
2. 先给定 D204=18000，当 M2 为 ON 的状态时，VFD 接收到反转命令，会先减速停止后，持续以频率 180Hz 反转方向运转。
3. 当 M3 为 ON 的状态时（此时 D206 数值无作用），VFD 接收到停止命令，会减速停止。

4. 当 M4 为 ON 的状态时，将 VFD 的 H2100~H2104 读取到 D220~224

装置	D220	D221	D222	D223	D224
显示内容	Error code	VFD 状态	频率命令	输出频率	输出电流

驱动器状态：当 Bit2 为 1 时，表示 VFD 执行寸动指令；当 Bit4~3 为 11B 时，表示 VFD 的运转方向为反转。频率命令为 18000，表示 VFD 以频率 180Hz，开始运转。

（请参考台达交流伺服电机驱动器手册\通讯协议的参数地址定义 说明）

5. 当接收完毕标志 SM100 为 ON，会将 M1~M5 清除，避免影响下次通讯命令执行。

VFD 回传的数据接收完毕后，会对 VFD 回传的数据格式进行确认，若符合数据格式，SM100=ON；如果回传数据错误则 SM102=ON。

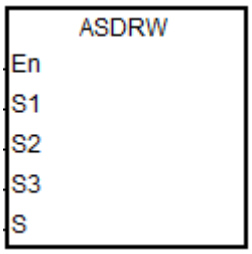
API	指令码		操作数	功能
1815	ASDRW		**S_1 · S_2 · S_3 · S**	台达伺服专属串行通讯指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●		○	○	○	○		
S_2								●	●		○	○	○	○		
S_3								●	●		○	○	○	○		
S								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 通讯端口编号

S_2 ： 伺服站号

S_3 ： 通讯功能代码

S ： 来源与接收数据

指令说明：

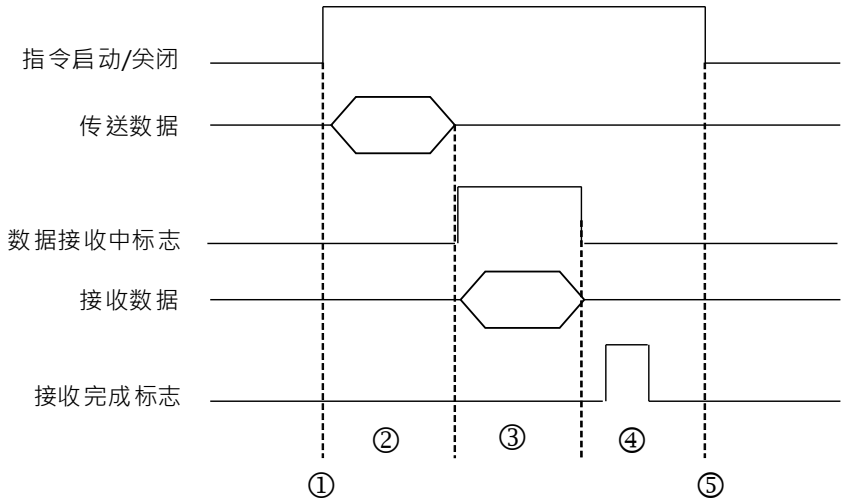
1. **S_1** 为通讯端口编号，COM1 编号为 1，COM2 编号为 2，Card1 编号为 11，Card2 编号为 12。若超出通讯端口范围时，则指令将不会执行任何通讯数据接收。
2. **S_2** 为伺服站号，当输入站号为 0 时，则为广播模式。数值范围为 0~254。超过此范围不执行。
3. 详细伺服参数说明请参考台达伺服操作手册。
4. **S_3** 为通讯功能代码与 S 来源或接收数据说明如下表所示：

适用 A，AB，A+，B 系列			
S_3 代码	**S_3** 功能名称	**S** 来源或接收数据说明	备注
0	读取伺服状态值	连续占用 5 个装置 **S**~**S**+4	读取 P0-04~08 状态值
1	读取伺服寄存器值	连续占用 8 个装置 **S**~**S**+7	读取 P0-09~16 寄存器
2	写入伺服寄存器值	连续占用 8 个装置 **S**~**S**+7	写入 P0-09~16 寄存器
3	JOG 速度输入，正转，反转，停止	数值范围为 1~3000（速度），4999（正转），4998（反转），5000（停止）	写入 P4-05

适用 A，AB，A+，B 系列			
S_3 代码	**S_3** 功能名称	**S** 来源或接收数据说明	备注
4	Servo ON/OFF	1 为 Servo ON，其余 OFF	写入 P2-30
5	写入内部速度命令	数值范围为-5000~5000	写入 P1-09 ~ 11
6	写入内部扭力命令	数值范围为-300~300	写入 P1-12 ~ 14

适用 A2，B2 系列			
S_3 代码	**S_3** 功能名称	**S** 来源或接收数据说明	备注
16	读取伺服状态值	连续占用 10 个装置 **S**~**S**+9	读取 P0-09 ~ 13(32 位数值)
17	写入伺服寄存器	连续占用 8 个装置 **S**~**S**+7	写入 P0-17 ~ 20(32 位数值)
18	写入映射参数值[註 1]	连续占用 8 个装置 **S**~**S**+7	写入 P0-25 ~ 28(32 位数值)
19	JOG 速度输入，正转，反转，停止	数值范围为 1~5000（速度），4999（正转），4998（反转），0（停止）	写入 P4-05
20	Servo ON/OFF	1 为 Servo ON，其余 OFF	写入 P2-30
21	写入内部速度命令(共三组)	连续占用 6 个装置，数值范围为 -60000~60000	写入 P1-09 ~ 11(32 位数值)
22	写入内部扭力命令(共三组)	连续占用 6 个装置，数值范围为 -300~300	写入 P1-12 ~ 14(32 位数值)
23	映像伺服参数`-7-目标设定[註 1]	连续占用 8 个装置 **S**~**S**+7	写入 P0-35 ~ 38(32 位数值)
注 1：B2 系列不支持功能码 18 及 23			

5. 通讯传送与接收时序图如下：



图标符号说明：

① → 由用户设定启动/关闭

② → 指令启动后开始传送数据，此期间通讯逾时不会被计时。

③ → 设定接收中标志，并从接收到第 1 个字符时，即开始累积计时，每接收一个字符即清除累积计时值一次，若是累积计时超出通讯逾时设定值，则产生通讯逾时标志。

④ → 接收完成时设定接收完成标志，用户须自行清除此标志。

⑤ → 监看到完成标志后，若需要再次发送通讯指令，则需关闭指令一个周期，然后在下一次周期开启指令。

6. 此通讯指令无使用次数限制，但是每个通讯端口每次只能被一条通讯指令输出执行，若是已完成传送接收时，需自行关闭通讯指令，将通讯控制权释放。

7. 此指令不可使用于 ST 结构化语言编辑、中断程序或仅被调用一次性的功能块（FB）程序。

通讯协议设定范例：

1. 使用 HWCONFIG 将 PLC COM1（RS485）通讯格式设定成 ASCII，115200，8，E，1

2. 使用台达伺服 ASDA-A2 面板进行基本设定，步骤如下

   A. 将伺服 P2-08 设置为 10，回归原厂设定

   B. 将伺服断电后重新上电

   C. 将 P1-01 设置为 0001（PR 模式）

   D. 将 P3-00 设置为 2，将伺服站号设定为 2

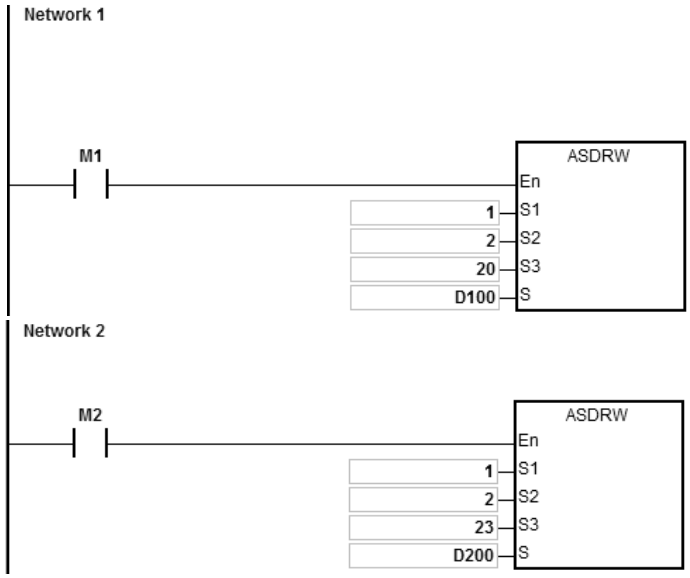
   E. 将 P3-01 设置为 0205，将伺服 RS485 通讯速率设为 115200

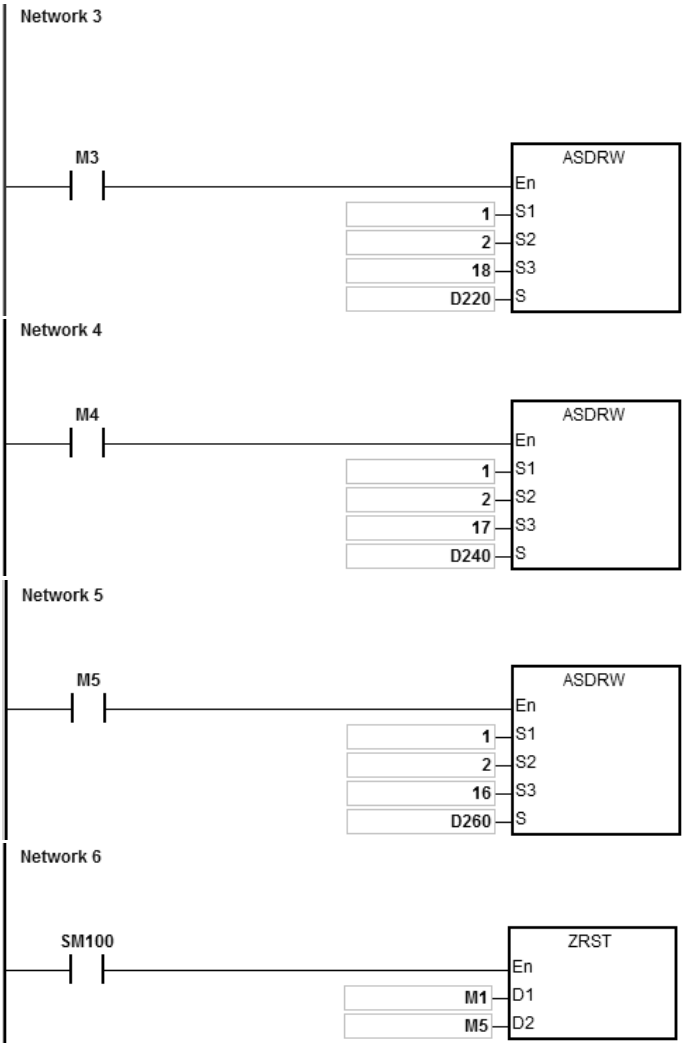
   F. 将 P3-02 设置为 0004，将伺服 RS485 通讯格式设为 8，E，1

   G. 将伺服断电后重新上电，完成基本设定后，即可开始操作

程序范例：

使用 ASDRW 指令，做一段速控制，让伺服以 3000.0 r/min ，加速时间 400ms，减速时间 200ms 移动到相对位置 5000000PUU





1. AS 主机与 ASDA-A2 联机：
   先给定 D100=1，当 M1 为 ON 的状态时，启动伺服 Servo ON

2. 给定 D200~D207 的数据内容如下：(准备写入 ASDA-A2 的 P0-35~38 数据内容)
   连续占用 8 个装置

装置	D200	D201	D202	D203	D204	D205	D206	D207
值 (16#)	05140515		06020602		06030603		053C0507	

当 M2 为 ON 的状态时，将 D200~207 内的数值，写到 ASDA-A2 的 P0-35~38。
P0-35~38 的设定值，是用来设定 P0-25~28 映像目标设定 (请参考台达伺服操作手册，可自行设定要映像目标)

ASDA-A2	P0-35	P0-36	P0-37	P0-38
值 (16#)	0514 0515	0602 0602	0603 0603	053C 0507

当 P0-38 设定为 053C 0507 时，则表示 P0-28 映像参数目标为 P5-60（16BIT）和 P5-07（16BIT）

ASDA-A2	P0-25	P0-26	P0-27	P0-28
映像参数目标	P5-20 P5-21	P6-02	P6-03	P5-60 P5-07
参数名称	加/减速时间 1 加/减速时间 2	路径型式 1	路径 1 数据	目标速度及 PR 命令触发

3. 给定 D220~D227 的数据内容如下：

装置	D220	D221	D222	D223	D224	D225	D226	D227
值（16#）	0190 00C8		0000 1083		4C4B40		7530 0001	

当 M3 为 ON 的状态时，将 D220~227 内的数值，写到 ASDA-A2 的 P0-25~28。

ASDA-A2	P0-25	P0-26	P0-27	P0-28
值（16#）	0190 00C8	0000 1083	4C4B40	7530 0001

设定加速时间=0190（400ms），减速时间=00C8（200ms），路径型式=1083，位置命令=4C4B40（5000000PUU），目标速度=7530（3000.0rpm），PR 命令触发=1 时，则伺服开始动作

4. 给定 D240~D247 的数据内容如下：(准备写入 ASDA-A2 的 P0-17~20 数据内容)

连续占用 8 个装置

装置	D240	D241	D242	D243	D244	D245	D246	D247
值（10#）	41		0		0		0	

当 M4 为 ON 的状态时，将 D240~247 内的数值，写到 ASDA-A2 的 P0-17~20。

P0-17~20 的设定值，是用来设定 P0-09~12 显示的内容（请参考台达伺服操作手册，可自行设定要显示的内容）

ASDA-A2	P0-17	P0-18	P0-19	P0-20
值（10#）	41	0	0	0

当 P0-17 设定为 41 时，则表示 P0-09 内容为驱动器状态

ASDA-A2	P0-09	P0-10	P0-11	P0-12
显示内容	驱动器状态	伺服电机回授脉冲数	同左	同左

5. 当 M5 为 ON 的状态时，将 ASDA-A2 的 P0-17~20 读取到 D260~267

装置	D260	D262	D264	D266
显示内容	驱动器状态	伺服电机回授脉冲数	同左	同左

当驱动器状态， BIT4 为 1 时，表示目标位置到达（请参考台达伺服操作手册，参数 P0-46 说明）

6. 当接收完毕标志 SM100 为 ON，会将 M1~M5 清除，避免影响下次通讯命令执行。
   ASDA-A2 回传的数据接收完毕后，会对 ASDA-A2 回传的数据格式进行确认，若符合数据格式，SM100=ON；如果回传数据错误则 SM102=ON。

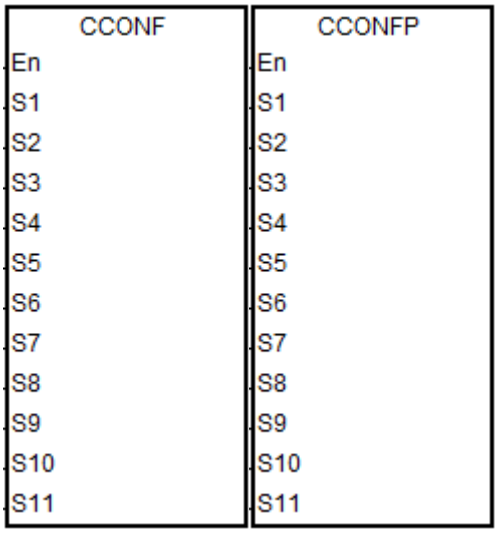
API	指令码			操作数	功能
1816		CCONF	P	S_1~S_{11}	通讯端口数据交换表参数设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●	●				○	○		
S_6								●	●				○	○		
S_7			●					●								
S_8								●	●				○	○		
S_9								●	●				○	○		
S_{10}								●	●				○	○		
S_{11}			●					●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1~S_{11}	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1~S_{11} ：请参考指令说明

指令说明：

1. S_1~S_{11} 装与装置名称与说明如下表所示

装置	名称	使用说明	数据类型	备注
S_1	通讯端口编号	1=COM1，2=COM2，11=通讯功能卡 1，12=通讯功能卡 2，其它数值，指令将不执行参数变更。V1.06.00（含）以后支持通讯功能卡 1/2。	WORD	

装置	名称	使用说明	数据类型	备注
S_2	数据交换表项目编号	数值范围为 1~32，其它数值，指令将不执行参数变更	WORD	
S_3	远程联机站号	数值范围为 0 与 1~240，若超出 240 将自动以 240 执行，若写入 0 或小于 0，则不变更原先设定站号	WORD	负数视为 0
S_4	读取通讯功能码	支持 16#01，02，03，04，17，若超出此范围，指令将不执行参数变更 若 **S_4**=16#17 则 **S_8** 会自动视为 16#17	WORD	
S_5	读取远程通讯地址	16#0000~16#FFFF	WORD	
S_6	读取数据长度	• 位（Bit）功能码 16#01, 02，可读取长度范围为 1~256。 • 字符（Word）功能码 16#03, 04, 17，可读取长度范围为 1~100。 当填入数值为 0（或小于 0)时，即表示不启动读取功能，其它超出范围的数值，皆会以最大值执行。	WORD	
S_7	本地接收数据存放寄存器	位（Bit）功能码：仅能选用 M 字符（Word）功能码：仅能选用 D	BOOL WORD	
S_8	写入通讯功能码	支持 16#05，06，0F，10，若超出此范围，指令将不执行参数变更	WORD	
S_9	写入远程通讯地址	16#0000~16#FFFF	WORD	
S_{10}	写入数据长度	• 位（Bit）功能码 16#0F 的写入长度范围为 1~256；功能码 16#05 只能填入 1。 • 字符（Word）功能码 16#10 的写入长度范围为 1~100；功能码 16#06 只能填入 1。 当填入数值为 0（或小于 0)时，即表示不启动写入功能，其它超出范围的数值，皆会以最大值执行。	WORD	
S_{11}	本地写入远程数据寄存器	位（Bit）功能码：仅能选用 M 字符（Word）功能码：仅能选用 D	BOOL WORD	

2. 使用此指令时，建议使用脉冲型指令方式设定。
3. **S_4** 和 **S_8** MODBUS 通讯功能码详细说明如下：

   AS 的读取多笔位装置（Bit）（非 Discrete Input 装置）命令为 1（16#01）

AS 的读取多笔位装置（Bit）(仅 Discrete Input 装置）命令为 2（16#02）

AS 的读取多笔字符装置（Word）(非 Input Register 装置）命令为 3（16#03）

AS 的读取多笔字符装置（Word）(仅 Input Register 装置）命令为 4（16#04）

AS 的单笔位装置（Bit）状态写入命令为 5（16#05）

AS 的单笔字符装置（Word）数据写入命令为 6（16#06）

AS 的多笔位装置（Bit）状态写入命令为 15（16#0F）

AS 的多笔字符装置（Word）数据写入命令为 16（16#10）

AS 的多笔字符装置（Word）数据同步读写命令为 23（16#17）

仅支持上述功能码，当设定其余功能码(例如：0)，则被视为无效参数(含通讯地址、长度与起始寄存器)，并且以原先通讯参数执行通讯数据交换功能。

4. 当用户在 **S_4**（读取通讯功能码）中选择 16#17（同步读写)，则 **S_8**（写入通讯功能码）操作数视为无效，自动视为 16#17 做写入动作。
5. V1.04.00 版（含）以前当用户在 **S_8**（写入通讯功能码）中选择 16#05 或 16#06，则 **S_{10}**（写入数据长度）操作数视为无效，自动视为 1 笔做写入动作。
6. 此指令所设定之参数值，仅在 PLC 运行过程有效，当 PLC 断电之后，重新上电时，皆以软件 HWCONFIG 所设定之 Data Exchange 表内数值为默认值，若是执行过程中，有需要变换参数内容值时，才需要使用此指令进行修改。
7. 当数据交换功能未启动时，此指令为立即设定通讯联机参数。若是数据交换功能已在执行中，并且刚好变更正在联机编号的通讯参数，则需等到下一个循环周期，才能执行新的通讯参数，举例：数据交换功能正在执行联机编号 3，而指令指定变更联机编号 3 的参数，则需等到下一次循环到联机编号 3，才会执行此次指令所变更之通讯参数。
8. 此指令仅提供通讯参数修改之功能，若需要使用 PLC 程序进行联机编号启动或关闭时，请参考下面启动标志，进行通讯联机的启动与关闭功能。当用户于编辑软件设定自动扫描功能时，则联机编号的启动/关闭标志，将会于数据交换功能执行扫描完成之后，自动更新启动/关闭状态一次。详细搭配之 SM 编号与说明如下表所示：

SM 编号	属性	COM1 数据交换参数内容说明
SM750	R/W	数据交换启动标志
SM752 ~ SM783	R/W	数据交换联机编号 1~32 之启动标志
SM784 ~ SM815	R	数据交换联机编号 1~32 之读取成功标志
SM816 ~ SM847	R	数据交换联机编号 1~32 之错误标志
SM 编号	属性	COM2 数据交换参数内容说明
SM862	R/W	数据交换启动标志
SM864 ~ SM895	R/W	数据交换联机编号 1~32 之启动标志
SM896 ~ SM927	R	数据交换联机编号 1~32 之读取成功标志
SM928 ~ SM959	R	数据交换联机编号 1~32 之错误标志
SM 编号	属性	通讯功能卡 1 数据交换参数内容说明

SM 编号	属性	COM2 数据交换参数内容说明
SM1710	R/W	数据交换启动标志
SM1712 ~ SM1743	R/W	数据交换联机编号 1~32 之启动标志
SM1744 ~ SM1775	R	数据交换联机编号 1~32 之读取成功标志
SM1776 ~ SM1807	R	数据交换联机编号 1~32 之错误标志
SM 编号	属性	通讯功能卡 2 数据交换参数内容说明
SM1822	R/W	数据交换启动标志
SM1824 ~ SM1855	R/W	数据交换联机编号 1~32 之启动标志
SM1856 ~ SM1887	R	数据交换联机编号 1~32 之读取成功标志
SM1888 ~ SM1919	R	数据交换联机编号 1~32 之错误标志

上述读取成功标志为数据接收完成并且检查数据正确时，将会被 PLC 设为 ON，若接收数据有错误发生或通讯逾时发生，则设定错误标志为 ON（可参考错误代码）。由于每一个联机编号之读取成功与错误标志，皆不会同一时间为 ON，因此数据交换过程中，PLC 不会清除任何一条联机读取成功或错误标志。

9. 数据交换功能搭配之 SR 状态说明如下（皆为只读寄存器）：

SR 编号	说明
SR1335	COM1 数据交换实际执行联机 1~32 的周期时间
SR1336	COM1 数据交换目前执行周期联机编号
SR1340 ~ SR1371	COM1 数据交换联机编号 1~32 的错误代码
SR1375	COM2 数据交换实际执行联机 1~32 的周期时间
SR1376	COM2 数据交换目前执行周期联机编号
SR1380 ~ SR1411	COM2 数据交换联机编号 1~32 的错误代码
SR1435	通讯功能卡 1 数据交换实际执行联机 1~32 的周期时间
SR1436	通讯功能卡 1 数据交换目前执行周期联机编号
SR1440 ~ SR1471	通讯功能卡 1 数据交换联机编号 1~32 的错误代码
SR1475	通讯功能卡 2 数据交换实际执行联机 1~32 的周期时间
SR1476	通讯功能卡 2 数据交换目前执行周期联机编号
SR1480 ~ SR1511	通讯功能卡 2 数据交换联机编号 1~32 的错误代码

10. 数据交换功能不提供写入成功标志，若有需要使用时，建议可参考执行中联机编号进行判断，举例：当执行联机编号 SR1336 为 3 时，此通讯动作为先读取通讯数据，接着读取成功后，进行写入通讯，当写入完成之后，此执行联机编号将被修改为 4。

11. 当 **S_6** 长度与 **S_7** 目的起始装置检查超出 D 或 M 组件范围时，则 **S_6** 长度数值将自动被修改为合理范围的长度值，举例：长度值为 100，且目的起始位置为 M8182，则长度值自动会被修改为 10。

12. 以下情况发生时，指令将不会执行，也不会变更 HWCONFIG 中的参数设定，并且设定 SM0=ON，错误代码 SR0=16#200B。

    - 当 **S_1**、**S_2**、**S_4** 与 **S_8** 参数设定数值超出指定范围时，视为输入错误。

- 当 **S_4** 或 **S_8** 功能码选择 bit 做读取或写入功能时，本地端 **S_7** 或 **S_{11}** 存放数据的装置，必须选择 M 装置，若选择错误，视为输入错误。
- 当 **S_4** 或 **S_8** 功能码选择 Word 做读取或写入功能时，本地端 **S_7** 或 **S_{11}** 存放数据的装置，必须选择 D 装置，若选择错误，视为输入错误。

以下程序范例以 AS 的 COM1（RS485）说明：

程序范例：

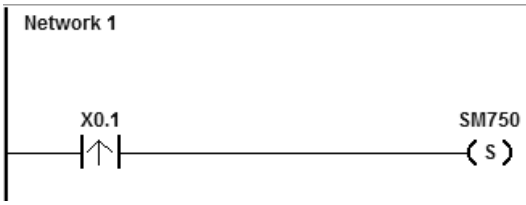
1. AS 主机与 DVP-ES2 主机数据交换的对应，于 ISPSoft HWConfig 中数据交换表-COM1 如下：

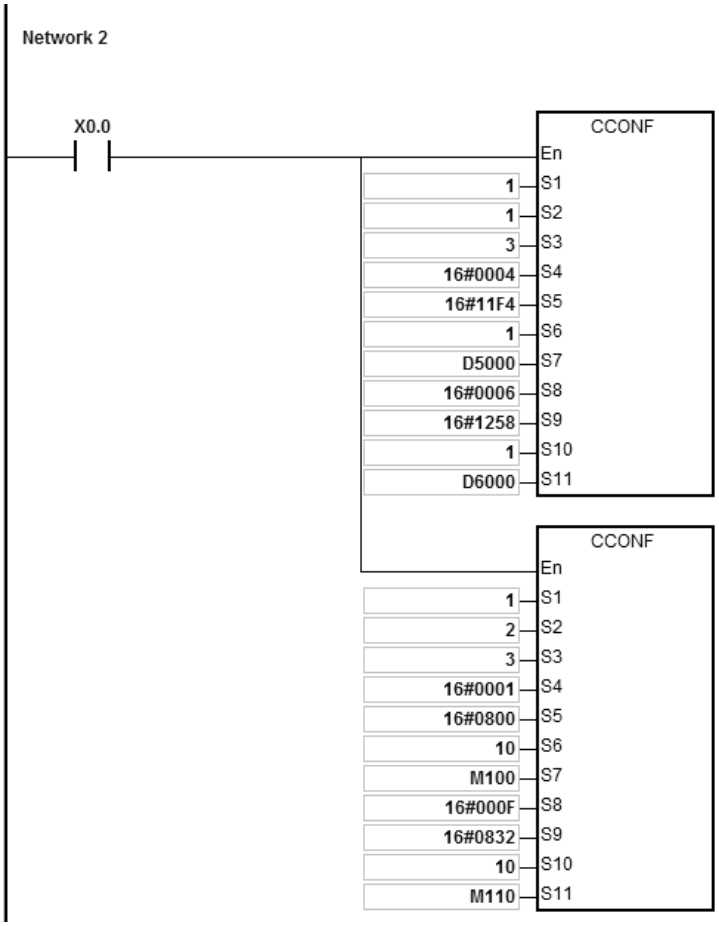
项目	检测方式	远程站号	本地地址		远程地址	长度	功能码	启动方式
1	指定联机	2	D50	←	D50	50	H03	程控
			D100	→	D100	50	H10	
2	指定联机	2	D150	←	D150	50	H03	程控
			D200	→	D200	50	H10	

2. 假设启动前数据交换前，AS 主机与 DVP-ES2 主机对应的数据如下：

AS PLC（Master）	内容值	ES2 PLC（Slave）	内容值
D50~D99	内容全为 0	D50~D99	内容为 1~50
D100~D149	内容为 100~149	D100~D149	内容全为 0
D150~D199	内容全为 0	D150~D199	内容为 150~199
D200~D249	内容为 200~249	D200~D249	内容全为 0

3. X0.1=ON，启动 AS 主机与 DVP-ES2 主机数据交换功能。





4. 启动数据交换后，AS 主机与 DVP-ES2 主机对应的数据变更为：

AS PLC（Master）	内容值	ES2 PLC（Slave）	内容值
D50~D99	内容全为 1~50	D50~D99	内容为 1~50
D100~D149	内容为 100~149	D100~D149	内容全为 100~149
D150~D199	内容全为 150~199	D150~D199	内容为 150~199
D200~D249	内容为 200~249	D200~D249	内容全为 200~249

5. 当 X0.0=ON，变更 AS 主机 COM1 数据交换表参数：

项目	检测方式	远程站号	本地地址		远程地址	长度	功能码	启动方式
1	指定联机	3	D5000	←	D500	1	H04	程控
			D6000	→	D600	1	H06	
2	指定联机	3	M100	←	M0	10	H01	程控
			M110	→	M50	10	H0F	

6. 因 COM1 数据交换表参数的变更，AS 主机与 DVP-ES2 主机对应的数据变更为：

AS PLC（Master）	内容值	ES2 PLC（Slave）	内容值
D5000	内容为 3000	D500	内容为 3000
D6000	内容为 4000	D600	内容为 4000
M100~M109	内容全为 ON	M0~M9	内容全为 ON
M110~M119	内容全为 OFF	M50~M59	内容全为 OFF

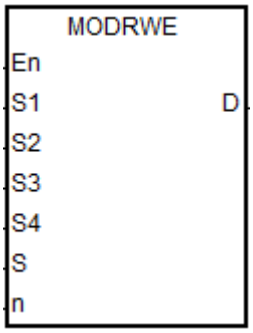
API	指令码	操作数	功能
1817	MODRWE	**S_1 · S_2 · S_3 · S_4 · S · n · D**	MODBUS 直接指定通讯端口通讯指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S								●								
n								●	●				○	○		
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
S	●	●				●							
n		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 通讯端口编号

S_2 ： 联机装置地址

S_3 ： 通讯功能码

S_4 ： 欲读写数据的地址

S ： 欲读写之数据存放或来源寄存器

n ： 读写数据长度

D ： 读写完成设定标志

指令说明：

1. **S_1** 为通讯端口编号，COM1 编号为 1，COM2 编号为 2，Card1 编号为 11，Card2 编号为 12。若超出通讯端口范围时，则指令将不会执行任何通讯数据接收。
2. **S_2** · **S_3** · **S_4** · **S** · **n**：请参考 API1808 MODRW 指令说明。

3. **D**：通讯指令完成通讯时，指定设定之通讯状态标志。此通讯状态标志将占用 3 个连续装置，且须由用户自行清除。通讯标志状态说明如下表：

操作数	说明
D	接收成功标志
D +1	接收数据错误标志
D +2	接收通讯逾时标志

注：此三个状态标志在每次通讯完成时，都会只有其中一个标志被设定为 ON，且连带对应之通讯端口特殊标志（SM）也会被设为 ON。若有需使用通讯特殊标志，请参考第 6.19.3 节。

4. 此指令发送时机为每次启动指令时，当通讯完成之后，用户须关闭指令至少一个扫描周期后，再重新启动指令后，才能发送下一次通讯指令。
5. 此指令相关通讯动作与控制时序，皆与 API1808 MODRW 指令相同，唯有差别在于用户不需自行控制传送标志，也可达成通讯命令的发送。
6. 此指令不可使用于 ST 结构化语言编辑、中断程序或仅被调用一次性的功能块（FB）程序。

程序范例：

MODRW 指令与 MODRWE 指令之程序比较，这里以 PLC COM1 且功能码 03 对 DVP-ES2 的 D20 读取 8 笔数据为例做说明，其它 PLC 通讯端口依此类推，其它功能码的使用请参考 API1808 MODRW 及下面的范例。

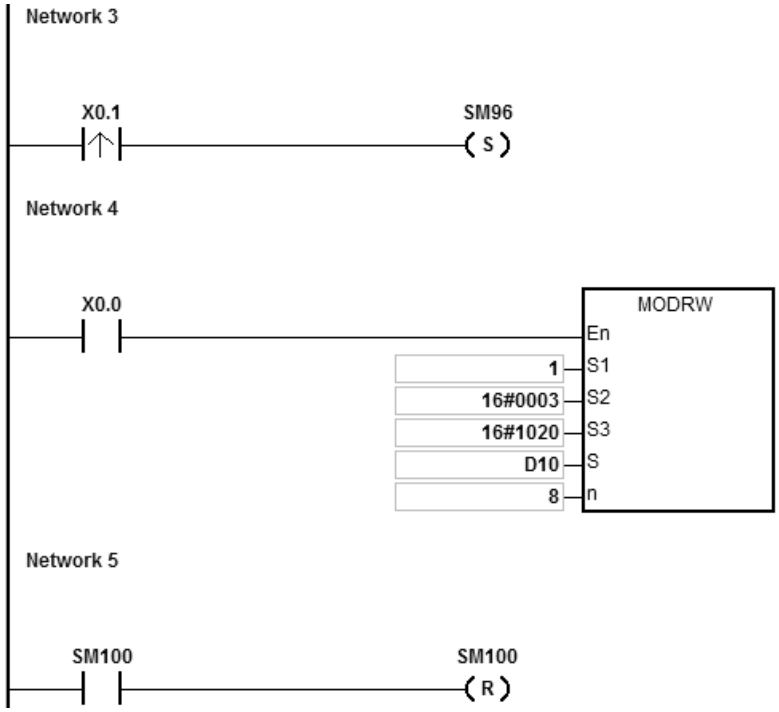
1. 假设 DVP-ES2 主机中 D20 的通讯地址为 16#1020 且 D20~D27 的内容值如下：

装置	D20	D21	D22	D23	D24	D25	D26	D27
值（16#）	1234	5678	1122	3344	5566	7788	99AA	BBCC

2. AS 用通讯方式读取 DVP-ES2 主机中 D20~D27 的内容值指令

方法一：采用 MODRW 指令的写法

当 SM96 为 ON 且 X0.0 也为 ON 的状态时，将会对 DVP-ES2 的 D20~D27 进行数据读取。



MODRW 指令内容说明：

操作数	说明	装置
S_1	联机装置地址	16#0001
S_2	通讯功能码	16#0003
S_3	读取 D20 数据的地址	16#1020
S	读回数据存放的起始寄存器	D10
n	读取数据长度	8

AS 主机与 DVP-ES2 间的通讯响应

ASCII 模式做通讯：(此处 ASCII 码不刻意转换表示，皆以 16#数值表示)

- AS 传送通讯命令：“：01 03 10 20 00 08 C4 CR LF”
- AS 接收通讯命令：“：01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC AA CR LF”

RTU 模式做通讯：

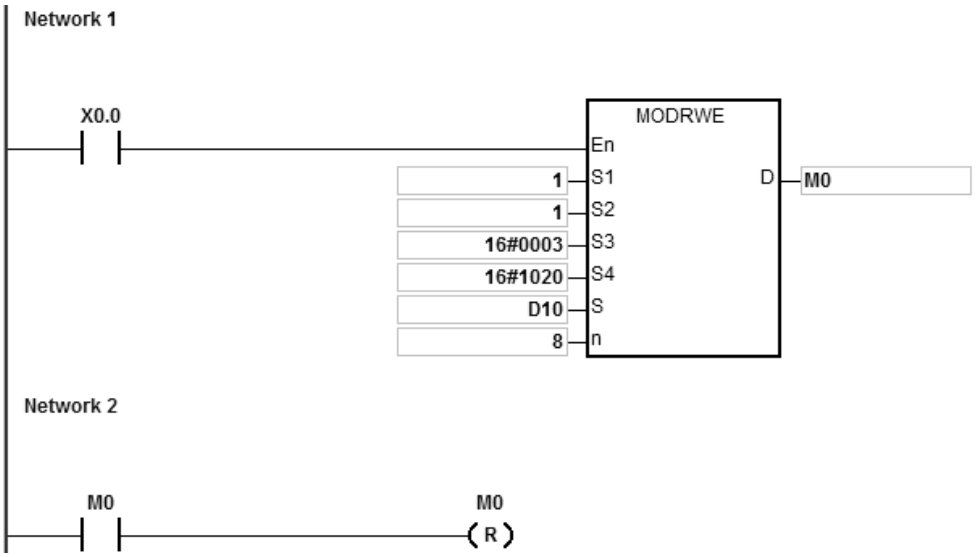
- AS 传送通讯命令：“01 03 10 20 00 08 41 06”
- AS 接收通讯命令：“01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC 90 FE”

因为数据正确无误，所以 SM100=ON。

DVP-ES2 所回传的数据接收完毕后，会对 DVP-ES2 所回传的数据格式进行确认，是否符合数据格式，如果正确无误则 SM100=ON，如果回传数据格式错误则 SM102=ON，如果没有响应则 SM104=ON。

方法二：采用 MODRWE 指令的写法

X0.0=ON 的状态时，将会对 DVP-ES2 的 D20~D27 进行数据读取。



MODRWE 指令内容说明：

操作数	说明	装置
S1	通讯端口编号	16#0001
S2	联机装置地址	16#0001
S3	通讯功能码	16#0003
S4	读取 D20 数据的地址	16#1020
S	读回数据存放的起始寄存器	D10
n	读取数据长度	8
D	读写完成设定标志	M0

AS 主机与 DVP-ES2 间的通讯响应

ASCII 模式做通讯：(此处 ASCII 码不刻意转换表示，皆以 16#数值表示)

- AS 传送通讯命令：“：01 03 10 20 00 08 C4 CR LF”
- AS 接收通讯命令：“：01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC AA CR LF”

RTU 模式做通讯：

- AS 传送通讯命令：“01 03 10 20 00 08 41 06”
- AS 接收通讯命令：“01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC 90 FE”

因为数据正确无误，所以 M0=ON。

DVP-ES2 所回传的数据接收完毕后，会对 DVP-ES2 所回传的数据格式进行确认，是否符合数据格式，如果正确无误则 M0=ON，如果回传数据格式错误则 M1=ON，如果没有响应则 M2=ON；此时对应的通讯特殊标志 SM100、SM102、SM104 也会 ON。

3. AS 主机的 D10~D17 的内容值：

装置	D10	D11	D12	D13	D14	D15	D16	D17
值（16#）	1234	5678	1122	3344	5566	7788	99AA	BBCC

补充说明：

1. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of BOOL。
2. **D**+2 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。

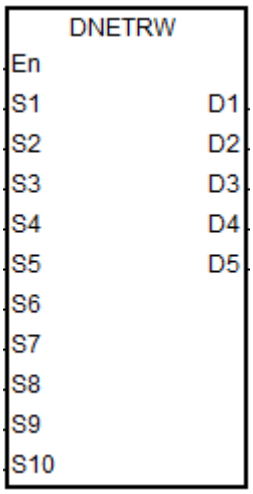
API	指令码	操作数	功能
1818	DNETRW	S_1，S_2，S_3，S_4，S_5，S_6，S_7，S_8，S_9，S_{10}， D_1，D_2，D_3，D_4，D_5	DeviceNet 通讯数据读写

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●	●				○	○		
S_6								●	●				○	○		
S_7								●	●				○	○		
S_8								●								
S_9								●	●				○	○		
S_{10}								●	●				○	○		
D_1		●	●	●												
D_2		●	●	●												
D_3								●								
D_4								●								
D_5								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
S_5		●				●							
S_6		●				●							
S_7		●				●							
S_8		●				●							
S_9		●				●							
S_{10}		●				●							
D_1	●												
D_2	●												
D_3		●				●							
D_4		●				●							
D_5		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 发送 DeviceNet 通讯模块编号

S_2 ： DeviceNet 站号（MAC ID）

S_3 ： 服务代码（Service Code）

S_4 ： 类编号（Class ID）

S_5 ： 实例编号（Instance ID）

S_6 ： 属性编号（Attribute ID）

S_7 ： 写入数据长度

S_8 ： 写入数据存储设备

S_9 ： 通讯逾时时间

S_{10} ： 重传次数

D_1 ： 完成标志

D_2 ： 错误标志

D_3 ： 错误代码

D_4 ： 读取数据长度

D_5 ： 读取数据存储设备

指令说明：

1. **S_1** 为主机右侧模块之顺序编号，第 1 台编号为 1，第 2 台编号为 2，以此原则类推，不分任何种类模块皆须被计数。指定范围为 1~32，超过此范围时，指令将自动以最小（<1 时）或最大（>32 时）值处理。
2. **S_2** 为 DeviceNet 站号，范围为 0~63。可指定主站要读写的从站站号，也可以是主站本身站号，代表要读写主站内部数据。
3. **S_3** 为 DeviceNet 服务代码：

服务代码	说明
0x01	读取所有属性（Get_Attribute_All）
0x02	设定所有属性（Set_Attribute_All）
0x0E	读取单个属性（Get_Attribute_Single）
0x10	设定单个属性（Set_Attribute_Single）

4. **S_4**、**S_5** 与 **S_6** 分别为 DeviceNet 协议 EPATH 组成部分，指定操作参数路径中的类、实例、属性编号。
5. **S_7** 为写入数据长度，以 byte 为单位。
6. **S_8** 为写入数据存储设备起始地址，数据摆放顺序是先放低位，再放高位。
7. **S_9** 为通讯逾时时间，数值范围为 1~100，时间单位为 0.1 秒。
8. **S_{10}** 为重传次数，数值范围为 0~3，当通讯逾时发生，会重新发送通讯。

9. **D_3** 为读写错误代码:

错误代码		说明
代码1（High Byte）	代码2（Low Byte）	
XX	FF	不符合DeviceNet标准
20	01	目标从站不存在
20	02	不能与从站建立联机
20	03	显性报文发送失败
16	00	通讯逾时

10. **D_4** 为读取数据长度，以 byte 为单位。

11. **D_5** 为读取数据存储设备起始地址，数据摆放顺序是先放低位，再放高位。

12. **D_1**、**D_2** 分别为通讯成功/错误设定之标志，详细的发送流程与时序图如下：

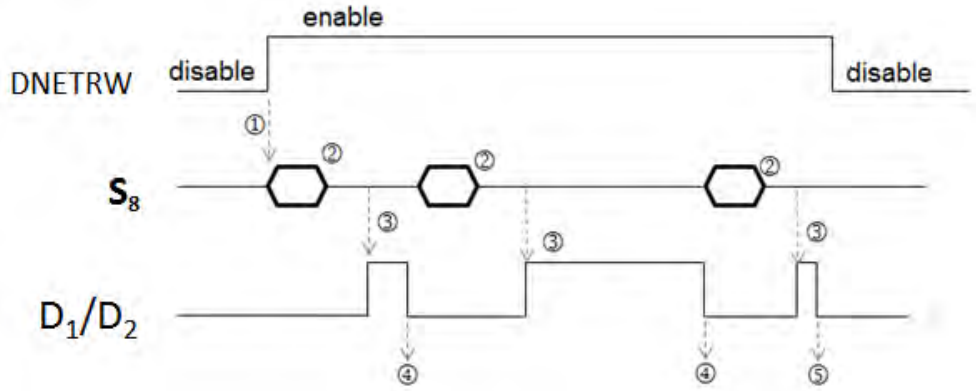
    ① 指令初次启动，若未有其它 DeviceNet 通讯占用，立即发送命令

    ② 正在发送命令

    ③ 发送完毕，依响应设定完成或是错误标志

    ④ 用户变更下一笔发送数据，清除完成/错误标志，即刻发出下一笔命令

    ⑤ 完成发送动作，关闭 DNETRW 指令



13. 此指令不可使用于 ST 结构化语言编辑、中断程序或仅被调用一次性的功能块（FB）程序。

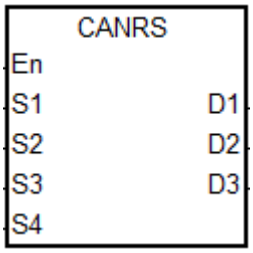
API	指令码	操作数	功能
1819	CANRS	S_1、S_2、S_3、S_4、D_1、D_2、D_3	CAN 自定义通讯传送与接收指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●	●				○	○		
S_2								●	●							
S_3								●	●				○	○		
S_4								●	●							
D_1								●								
D_2		●	●	●				●								
D_3		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
D_1		●				●							
D_2	●												
D_3	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 通讯模式设定

S_2 ： 通讯 ID（MsgID）

S_3 ： 传送长度

S_4 ： 传送来源起始位置

D_1 ： 接收储存起始位置

D_2 ： 通讯完成标志

D_3 ： 通讯错误标志

指令说明：

1. 此指令适用于 AS 系列主机韧体为 V1.04.00 版以上。此指令不建议使用 ST 结构化语言编辑、中断程序以及仅被调用一次性的功能块（FB）程序。
2. 执行前初始化指令前 ·请确认硬件组态(HWCONFIG)是否已经选择功能卡 2 之模式为 CANopen DS301 协议。
3. S_1 为通讯模式设定，其支持模式如下表所示，若设定错误数值，D_3 通讯错误标志将设为 ON，错误代码 SR659 将设为 1。

通讯代码		S1 设定值	通讯格式	功能说明
上 8 位	下 8 位			
0	0	16#0000	2.0A ID 11bits	主站模式，传送后等待接收
	1	16#0001		主站模式，传送广播命令后，进入接收多台从站模式，直到接收超时发生或是接收满 100 笔才结束。
	2	16#0002		从站模式，先接收后再传送
	3	16#0003		从站模式，仅接收
1	0	16#0100	2.0B ID 29bits	主站模式，传送后等待接收
	1	16#0101		主站模式，传送广播命令后，进入接收多台从站模式，直到接收超时发生或是接收满 100 笔才结束
	2	16#0102		从站模式，先接收后再传送
	3	16#0103		从站模式，仅接收

4. **S_2** 为传送 ID，依据 2.0A 或 2.0B 协议，自行决定占用几个 D 寄存器；当选择 2.0A 时，**S_2** 为 11 bits 的 ID code，数据发送格式与内容如下表：

S_2 编号	**S_2**
内容说明	Msg. ID

当选择 2.0B 时，则 **S_2** （Lo-word）与 **S_2**+1（Hi-word）为 29 bits 的 ID code。

S_2 编号	**S_2**	**S_2**+1
内容说明	Msg. ID（Lo-word）	Msg. ID(Hi-word)

5. **S_3** 为传送长度，其范围为 0~8，单位为 byte (8bits)，若是超出此范围之数值，则自动以小于 0 为 0 处理，或大于 8 为 8 处理。当传送长度为 0 时，通讯模式直接自动变更为从站接收模式，不传送任何数据。(此接收模式，可被运用于监看通讯报文的功能)。

6. **S_4** 为传送数据起始位置，且只使用下 8 位数据，例如：传送 4 笔数据，起始位置为 D10，则数据发送顺序如下表：

S_4 编号	D10	D11	D12	D13
内容说明	Data1	Data2	Data3	Data4

7. 当通讯模式 **S_1** 为主站且传送后接收模式或是从站模式时，接收数据将直接存放于指定的 **D_1** 位置（假设 **D_1** 指定 D100)，其存放内容格式如下表所示：

设定 2.0A 模式：

D_1 编号	D100	D101	D102 ~ D109（下 8 位）
内容说明	Msg. ID	Data Length	Data1 ~ Data8

设定 2.0B 模式：

D_1 编号	D100	D101	D102	D103 ~ D110（下 8 位）
内容说明	Msg. ID（Lo-word）	Msg. ID(Hi-word)	Data Length	Data1 ~ Data8

备注：在接收阶段若需要指定接收 Msg. ID 时，请依 2.0A/2.0B 模式先行设定 **D_1/ D_1+1** 数值，若是不指定 ID，则请在接收前清除 **D_1/ D_1+1** 数值为 0。

8. 当通讯模式 **S_1** 为主站广播通讯模式时，则接收数据将存放在指定的 **D_1** 位置（假设 **D_1** 指定 D100），其存放内容如下表所示：

选择 2.0A 模式：(仅介绍接收 2 台信息，其它信息请自行增加 **D_1** 编号)

响应顺序		第一台数据			第二台数据		第三台数据
D_1 编号	D100	D101	D102	D103 ~ D110（下 8 位）	D111	D112~D120	D121~130
内容说明	接收报文数	Msg. ID	Data Length	Data1 ~ Data8	Msg. ID	Length, Data	ID, Length, Data

选择 2.0B 模式：(仅介绍接收 2 台信息，其它信息请自行增加 **D_1** 编号)

响应顺序		第一台数据				第二台数据
D_1 编号	D100	D101	D102	D103	D104 ~ D111（下 8 位）	D112~D122
内容说明	接收报文数	Msg. ID（Lo-word）	Msg. ID（Hi-word）	Data Length	Data1 ~ Data8	ID, Length, Data

9. **D_2** 与 **D_3** 分别为接收完成与错误标志。这两个标志只会在指令被扫瞄时，依据当下通讯状态，判断是否接收完成或错误，若已完成或错误，则设定该标志为 ON。这两个标志会在指令每次启动时，自动清除一次 OFF，用户可不需自行清除。

10. 当指令为主站模式时，建议搭配接收逾时设定 (由 HWCONFIG 设定)，让接收数据未于指定时间内完整接收通讯报文时，可将 **D_3** 错误标志设为 ON，并且记录错误代码 2 于 SR659。此接收逾时若是设为 0，即表示不限制通讯逾时，此状态可运用于当从站模式。注意：采用无通讯逾时的方式，得自行关闭此指令之后，才能释放通讯控制权。

11. 此指令不限制使用次数，但是每次只允许发送一条 CAN 通讯命令，当前一个还在发送或接收中，下一个 CANRS 指令启动就会无法被执行。

12. 下面表格为 CAN BUS 格式中，Msg. ID 每个 bit 内容说明：

当选择 2.0A 协议且假设 **S_2** 数值为 16#0123 时，其 Msg. ID 内容如下表所示

Bit No.	15 ~ 11	10 ~ 8	7 ~ 4	3 ~ 0
S_2 内容（16 位）	-	1	2	3

当选择 2.0B 协议且假设 **S_2** 数值 (Lo-word) 为 16#1234 与 **S_2**+1 数值 (Hi-word) 为 16#0567 时，其 Msg. ID 内容如下表所示

Bit No.	31 ~ 29	28	27 ~ 24	23 ~ 20	19 ~ 16	15 ~ 0
S_2 内容（32 位）	-	0	5	6	7	1234

13. 操作数 **D_1** 采用变量宣告方式时，建议使用 Word 类型的矩阵方式。

14. CANRS 通讯时序图与说明如下：

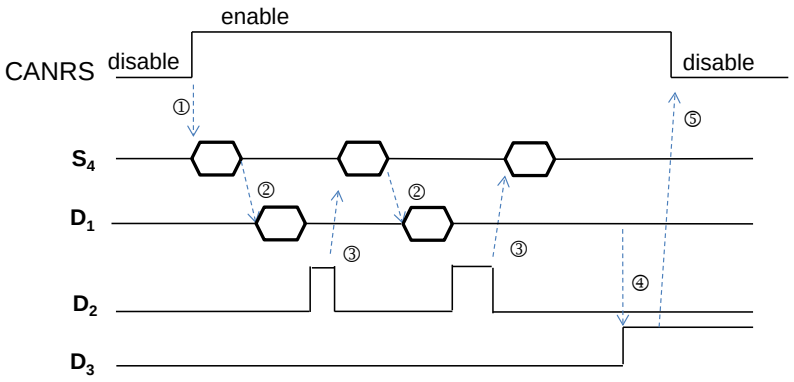
①：指令启动，若未有其它 CANRS 指令占用控制权，则清除 **D_2** 与 **D_3** 标志之后，立即发送数据。

②：发送完成，转为接收数据状态，接收完成后设定 **D_2** 完成标志为 ON。

③：用户清除 **D_2** 完成标志之后，指令再次发送数据。

④：接收时间逾时，设定 **D_3** 错误标志为 ON。

⑤：用户发现已有错误产生，可自行关闭指令，或者可清除 **D_3** 错误标志之后，让指令重新发送。



范例一、主站模式

使用 CANRS 指令，将 D10~D15（6BYTE）数据传送到从站，并且接收对方响应数据储存在 D100 开始的位置，当传送及接收完成后 M100 为 ON，相关参数说明如下：

主站模式_MsgID=2	
通讯模式设定	0=主站模式，传送后等待接收
通讯 ID	传送的 MsgID D0=1
传送长度	6 BYTE
传送来源起始位置	D10
接收储存起始位置	D100 接收的 MsgID D101 接收报文数 D102…. 接收的数据内容
通讯完成标志	M100
通讯错误标志	M101

1. 当 M1 为 ON 时，先将传送的 MsgID 设为 1，接收 MsgID 设为 2，开始传送数据

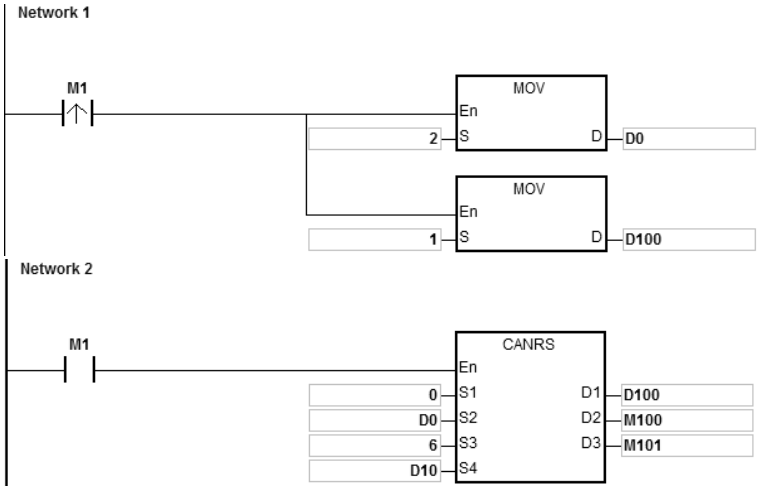
A. 然后开始执行数据传送 D10~D15（6BYTE）的数据到从站

装置	D10~D15
内容	H0A（由用户定义）

B. 接收从站响应的数据后放置在 D100~D109，接收完成后，M100 为 ON

装置	D100	D101	D102~D109
内容	2	8	H0B

2. 主站范例程序



范例二、从站模式:

使用 CANRS 指令，当接收主站传送的数据储存在 D120 开始的位置，并且回传 D20~D27（8BYTE）数据后，当接收及回传完成后 M110 为 ON，相关参数说明如下:

从站模式_MsgID=1	
通讯模式设定	2=从站模式，先接收后再传送
通讯 ID	传送的 MsgID D0=2
传送长度	8 BYTE
传送来源起始位置	D20
接收储存起始位置	D120 接收的 MsgID D121 接收报文数 D122…. 接收的数据内容
通讯完成标志	M110
通讯错误标志	M111

1. 当 M1 为 ON 时，先将传送的 MsgID 设为 2，接收 MsgID 设为 1，然后开始等待接收数据

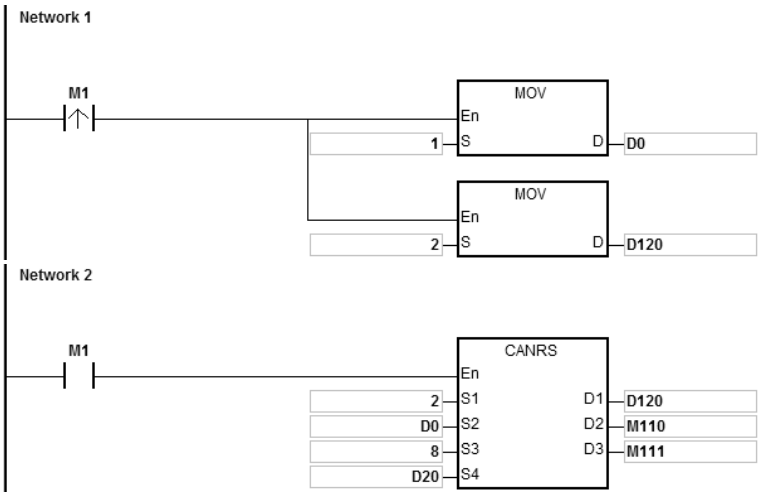
A. 接收主站传送来的 6BYTE 数据后放置在 D120~D127

装置	D120	D121	D122~D127
内容	1	6	H0A

B. 然后开始回传数据 D20~D27（8BYTE）的数据到主站，传送完成后，M110 为 ON。

装置	D10~D15
内容	H0B（由用户定义）

2. 从站范例程序:



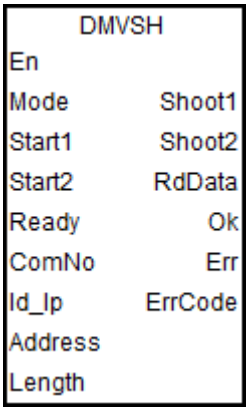
API	指令码	操作数	功能
1820	DMVSH	**Mode ~ ErrCode**	台达 DMV 启动检验与通讯指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Mode								●					○	○		
Start1		●	●	●				●								
Start2		●	●	●				●								
Ready	●															
ComNo								●					○	○		
Id_Ip								●					○	○		
Address								●					○	○		
Length								●					○	○		
Shoot1		●														
Shoot2		●														
RdData								●								
Ok		●	●	●				●								
Err		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Mode		●											
Start1	●												
Start2	●												
Ready	●												
ComNo		●				●							
Id_Ip		●				●							
Address		●				●							
Length		●				●							
Shoot1	●												
Shoot2	●												
RdData		●				●							
Ok	●												
Err	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Mode ： 触发与接收方式

Start1 ： 群组 1 之启动触发信号

Start2 ： 群组 2 之启动触发信号

Ready ： 接收 DMV 响应已完成信号

ComNo ： 选择 PLC 发送/接收的串行或网络通讯口

Id_Ip ： DMV 的串行通讯站号或网络 IP

Address ： 指定读取 DMV 数据的起始位置

Length ： 指定读取长度

Shoot1 ： PLC 通知 DMV 启动检验之群组 1 输出信号

Shoot2 ： PLC 通知 DMV 启动检验之群组 2 输出信号

RdData ： PLC 接收 DMV 回传数据存放的起始位置

Ok ： 通讯成功标志

Err ： 通讯错误标志

ErrCode ： 通讯错误代码

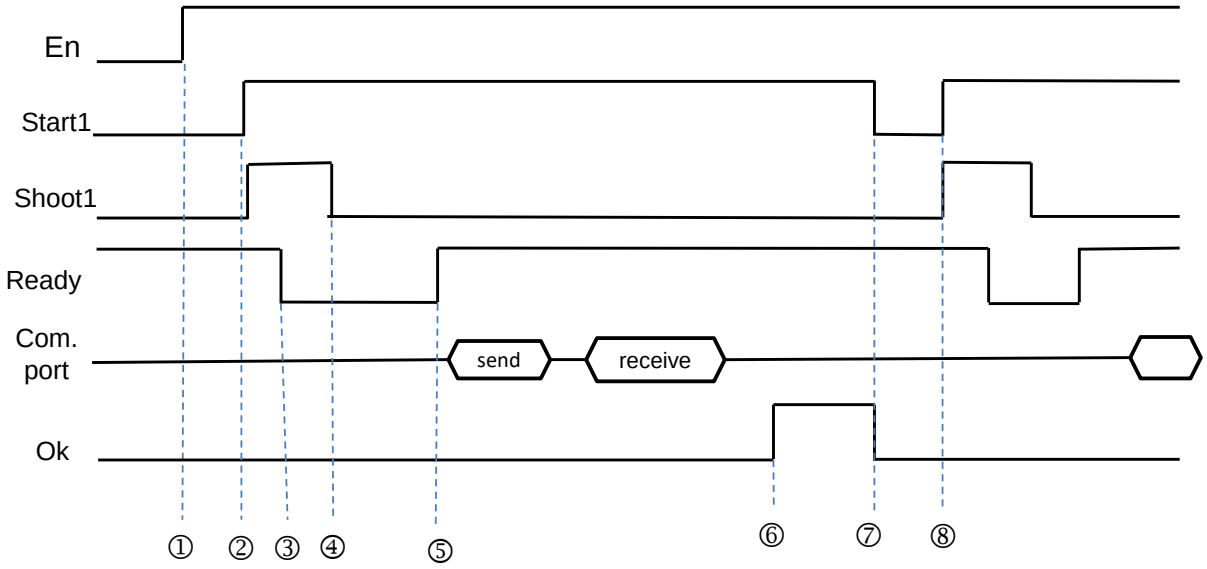
指令说明：

1. 此指令适用于 AS 系列主机韧体为 V1.06.00 版以上。
2. **Mode** 为选择 PLC 对 DMV（包含 DMV1000 与 DMV2000）的触发与接收方式选择，共有 2 种模式（0，1）可设定，当模式设定超出范围时，PLC 自动以模式 0 执行。
3. **Start1**、**Start2**、**Ready**、**Shoot1**、**Shoot2**、**RdData** 与 **Ok** 功能与触发检验动作流程，请参考范例说明。
4. **ComNo** 为选择 PLC 通讯口的编号，数值 1 表示 COM1，数值 2 表示 COM2，数值 11 表示 Card1，数值 12 表示 Card2，数值 4 表示 Ethernet 网络口。超出以上指定数值，或主机本身不支持之通讯口，将会产生通讯错误 **Err** 标志，并设定通讯错误代码为 16#0001。
5. **Id_Ip** 为从站 DMV 的通讯站号或网络 IP。**Address** 为读取 DMV 检验结果的通讯地址。**Length** 为读取检验数据的长度。
6. **ErrCode** 错误代码如下表所示：

错误代码	错误说明	错误排除
16#0001	通讯口指定错误	请重新指定通讯编号
16#0002	DMV 响应通讯错误	请确认通讯地址
16#0003	DMV 无响应通讯	请确认通讯线是否连接或 DMV 已开启运转
16#0004	正在等待 DMV 回应，不可重复启动	请修改启动触发时机

范例一、Mode = 0 模式，仅使用群组 1 触发 DMV 检验与接收 DMV 响应数据。

控制时序图：



控制时序之时机点说明：

① DMVSH 指令启动。

② 用户设定 Start1 通知指令发送 Shoot1 输出信号（约 10ms 为 On），通知 DMV 开始启动检验功能。

③ DMV 接收触发信息之后，DMV 将 Ready 信号由 On 转成 Off。

④ Shoot1 输出信号由 PLC 自动清除为 Off。

⑤ DMV 检验完成后，DMV 回复 Ready 信号由 Off 转成 On。PLC 端须等待 1.5 倍的输入滤波时间之后，才能开始判断 Ready 信号是否为 On？ PLC 接收到 Ready 由 Off 变 On，并发送出通讯读取（采用 MODBUS 的 0x03）命令。

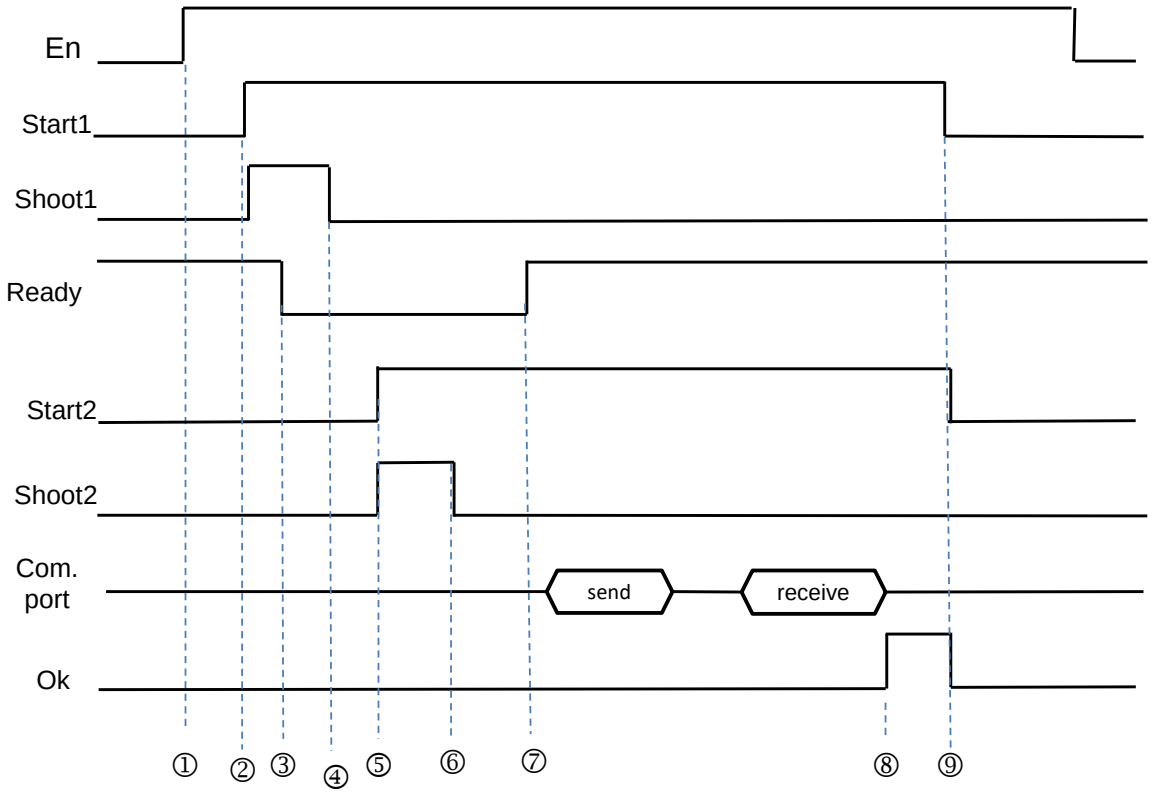
⑥ PLC 接收到 DMV 回复通讯数据之后，设定通讯 Ok 标志。此时用户可处理已接收数据。

⑦ 用户自行清除 Start1 开始与 Ok 标志为 OFF。

⑧ 若需要再次触发时，则可以直接在设定 Start1 开始 DMV 执行下一次检验（回步骤 2）。

范例二、Mode = 1 模式，使用群组 1 与群组 2 分开启动触发，但须等 Ready 信号都响应 ON 之后，才可下通讯读取数据。（此范例适用于 DMV1000 双镜头检验）

控制时序图：



① DMVSH 指令启动。

② 用户设定 Start1 通知指令发送 Shoot1 输出信号（约 10ms 为 On），通知 DMV 开始启动镜头 1 检验功能。

③ DMV 接收触发信息之后，DMV 将 Ready 信号由 On 转成 Off。

④ PLC 自动清除 Shoot1 输出信号为 Off。

⑤ 用户设定 Start2 通知指令发送 Shoot2 输出信号（约 10ms 为 On），通知 DMV 开始启动镜头 2 检验功能。

⑥ PLC 自动清除 Shoot2 输出信号为 Off。

⑦ DMV 检验完成后，DMV 回复 Ready 信号由 Off 转成 On，表示 2 个检验动作都已完成。PLC 接着发送出通讯读取（采用 MODBUS 的 0x03）命令。

⑧ PLC 接收到 DMV 回复通讯数据之后，设定通讯 Ok 标志。此时用户可处理已接收数据。

⑨ 用户自行清除 Start1、Start2 与 Ok 标志为 OFF。

若需要再次执行触发，则可重回步骤 2。

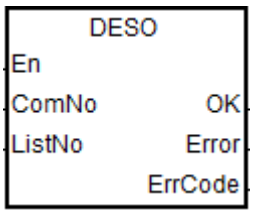
API	指令码	操作数	功能
1821	DESO	**ComNo ~ ErrCode**	指定数据交换表通讯编号发送一次

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
ComNo								●					○	○		
ListNo								●					○	○		
OK		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
ComNo		●				●							
ListNo		●				●							
OK	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



ComNo ： 指定数据交换表之通讯口编号

ListNo ： 指定数据交换表之联机编号

OK ： 传送完成标志

Error ： 传送错误标志

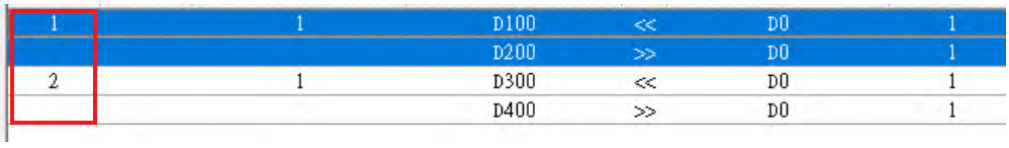
ErrCode ： 错误代码

指令说明：

1. 此指令搭配 ISPSoft 版本 V3.09.07 版及 AS 系列主机韧体为 V1.08.20 版以上。
2. 执行前初始化指令前，请确认主机内建之通讯数据交换表已经建立完成，若未建立，则 **Error** 错误标志为 On，并且 **ErrCode** 错误代码记录为 1。
3. **ComNo** 为指定数据交换表之通讯口编号，请参考下表所列：

通讯口名称	COM1	COM2	Ethernet	Card1	Card2
编号 (10 进制)	1	2	3 主机韧体 V1.12 以上	11	12

4. **ListNo** 为数据交换表所建立之通讯联机编号，请参考如下图的红框处所示。



1	1	D100	<<	D0	1
		D200	>>	D0	1
2	1	D300	<<	D0	1
		D400	>>	D0	1

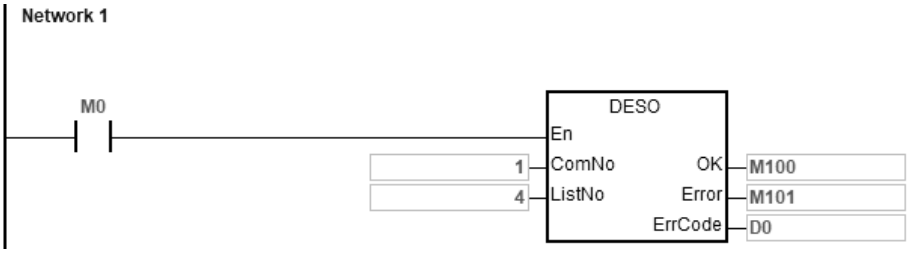
5. 当指令条件接点被启动时，该编号所建立之读取与写入命令，将会被列入数据交换表要启动执行的命令行内，等待通讯完成，并且收到从站响应正确之后，将会设定 **OK** 完成标志（该联机编号所对应之 SM 也会为 On），以及 **ErrCode** 清除为 0。
6. 此指令发送之前，会自动检查该联机编号是否已被启动循环通讯？若已经是启动中，那此指令将不会运作，而且 **Error** 错误标志为 On，**ErrCode** 错误代码记录为 2。
7. 当 **Error** 错误标志为 On 时，请参考下表 **ErrCode** 错误代码说明。

错误代码	错误说明
0	无错误，或已完成发送
1	数据交换表未建立，或该联机编号未建立
2	该联机已在数据交换表启动，不需重复启动
3	从站未回应
4	从站响应通讯错误
5	前一个相同通讯口占用中
6	通讯口不存在或是未插通讯卡 （AS-F232/422/485）
7	数据交换表联机编号超出范围（1~32）

8. 该指定通讯口的通讯逾时时间，将以数据交换表所设定的时间为准。
9. 此指令建议使用在从站设备不允许经常发送写入命令的通讯应用。此指令只在每次启动时，才会开启一次型通讯方式，用户要进行下一次通讯时，必须关闭该指令的条件接点之后，才能再重新启动发送。
10. 该指令无使用次数的限制，但是每个通讯口每次只允许启动一条通讯，若前一个指令还被占用执行中，并且指令还未被关闭，那下一个相同通讯口的通讯命令，就无法被启动，此时 **Error** 错误标志会设为 On，**ErrCode** 错误代码记录为 5。

程序范例：

当 M0 为 ON，DESO 会执行 COM1 数据交换表的第 4 条联机。



COM1 数据交换表

	Enable	Remote Station Address	Local Address	Direction	Remote Address	Quantity
1	☐	1	D400	←	D0	10
			D0	→	D0	10
2	☐	1	D410	←	D10	10
			D10	→	D10	10
3	☐	1	D420	←	D20	10
			D20	→	D20	10
4	☐	1	D430	←	D30	10
			D30	→	D30	10
5	☐	1	D440	←	D40	10
			D40	→	D40	10

6.19.3 通讯标志与寄存器说明

通讯特殊标志 SM 表

标志信号				功能说明	动作
COM1	COM2	Card 1	Card 2		
SM96	SM97	SM76	SM77	送信要求标志。当通讯指令将数据传送与接收时，必须用 P 指令方式，将此标志设为 ON，若通讯指令开始执行，则 PLC 执行数据传送的动作，并且当数据传送完毕后，此标志会自动由 PLC 清除。	用户设定 系统清除
SM98	SM99	SM78	SM79	接收等待或接收中标志。当此此标志为 ON 时，表示通讯端口目前等待或正在接收数据中。	系统设定与清除
SM100	SM101	SM80	SM81	接收完毕标志。当通讯指令接收完毕时，系统将设定此标志为 ON，用户在程序中可利用此标志为 ON 时，处理所接收到的数据。当已经处理完毕数据后，必须自行清除为 OFF。	系统设定 用户清除
SM102	SM103	SM82	SM83	通讯数据接收错误标志。	系统设定 用户清除
SM104	SM105	SM84	SM85	通讯接收逾时标志。用户若有设定通讯逾时 SR，当通讯等待接收中，一直未接收到通讯数据，且超出逾时设定值时，则系统会设定此标志为 ON。若通讯逾时状态已处理，用户必须自行清除为 OFF。	系统设定 用户清除
SM106	SM107	SM86	SM87	8/16 位处理模式选择，ON 为 8 位模式，OFF 为 16 位模式。	用户设定及清除
SM210	SM212	-	-	通讯模式，ON 为 RTU 模式，OFF 为 ASCII 模式。 此通讯模式建议由 HWCONFIG 指定，Card1 与 Card2 不支持 SM 设定方式。	用户设定及清除
SM209	SM211	SM90	SM91	通讯协议变更标志。将此标志设为 ON 时，系统依据通讯特殊寄存器（SR）之设定，重新设置通讯端口之通讯协议、通讯端口站号、通讯逾时之时间设定值，并根据特殊继电器（SM）之设定，重新设置通讯模式。当设定已变更完成时，此标志会自动清除为 OFF。 注：此变更功能不会修改 HWCONFIG 所设定之原始参数；当 PLC 断电后再重新上电，通讯端口将执行 HWCONFIG 所设定之参数。	用户设定 系统清除

注：以上标志皆为非停电保持型

通讯指特殊数据寄存器 SR 表

特殊				功能说明
COM1	COM2	Card 1	Card 2	
SR201	SR202	SR176	SR178	通讯端口站号。
SR209	SR212	SR177	SR179	通讯协议，详细设定说明请参考下面串行通讯协议表格。
SR210	SR213	SR182	SR183	通讯逾时之时间设定值，单位毫秒（ms）。当设定值大于 0 时，则通讯指令执行且进入接收模式时，若在指定的时间内没有收到第一个字符或任二字符之间的等待时间超过此设定值，将会设通讯逾时标志 SM 为 ON，此寄存器可被设为 0，则表示关闭逾时检测功能。使用 MODBUS 通讯指令时，建议设定在 100~32767（ms）区间内。

1. 串行通讯协议，其设定方法如下表：

b0	数据长度			7（内容值=0）		8（内容值=1）
b2~b1	同位			00	：	无（None）
				01	：	奇同位（Odd）
				10	：	偶同位（Even）
b3	stop bits			1 bit（内容值=0）		2 bits（内容值=1）
b7~b4	0001	（16#1）	：	4800		
	0010	（16#2）	：	9600		
	0011	（16#3）	：	19200		
	0100	（16#4）	：	38400		
	0101	（16#5）	：	57600		
	0110	（16#6）	：	115200		
	0111	（16#7）	：	230400		RS-232 不支持
	1000	（16#8）	：	500000		RS-232 不支持
	1001	（16#9）	：	921000		RS-232 不支持
	1111	（16#F）	：	自定义*1		
b8~b15	无定义（保留）					

注*1：自定义通讯速率，请使用 ISPSoft 之硬件组态 HWCONFIG 设定

2. MODBUS 通讯协议之传输速度与 RTU 模式结束时间如下：

通讯速率（bps）	RTU 结束时间间隔（ms）	通讯速率（bps）	RTU 结束时间间隔（ms）
4800	9	115200	1
9600	5	230400	1
19200	3	-	-
38400	2	-	-
57600	1	-	-

6.20 其它指令

6.20.1 其它指令一览表

API	指令码（位）		P 指令	功能
	16	32		
1900	WDT	–	✓	逾时监视定时器
1901	DELAY	–	✓	延迟指令
1902	GPWM	–	–	一般用脉冲宽度调变
1904	EPUSH	–	✓	变址寄存器存入
1905	EPOP	–	✓	变址寄存器读出
1906	INFO	–	✓	系统讯息读出

6.20.2 其它指令说明

API	指令码		操作数	功能
1900	WDT	P	—	逾时监视定时器

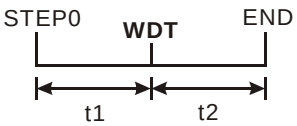
脉冲执行型	16 位指令	32 位指令
AS	AS	—

符号：

指令说明：

1. AS 系列 PLC 系统中有一逾时监视定时器 (Watchdog Timer)，用来监视 PLC 系统的正常运转。
2. WDT 指令可用来清除 PLC 中的监控定时器的计时时间。当 PLC 的扫描程序时间过长，超过用户设定的 WDT 时间 (默认值 200ms) 时，PLC ERROR 的指示灯会亮，PLC 会自动变 STOP，用户只要重新做 STOP=>RUN 就可以恢复正常。
3. 令逾时监视定时器动作的时机：
   - PLC 系统发生异常。
   - 程序运行时间太长，造成扫描时间大于 (Watchdog Timeout) 的内容值。可以下列 2 种方法来改善。

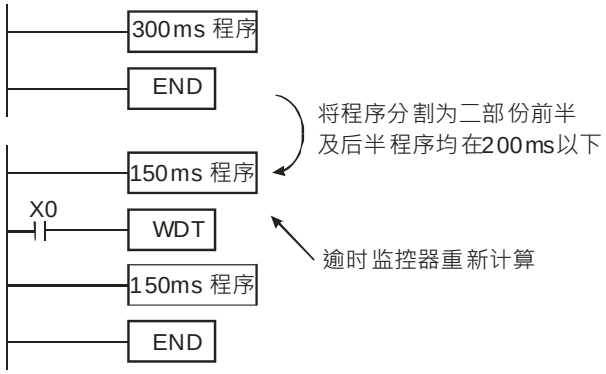
     (a) 使用 WDT 指令。



     (b) 要改变逾时监视时间请参考 ISPSoft 手册中的硬件组态 (HWCONFIG) 工具说明。

程序范例：

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



API	指令码			操作数	功能
1901		DELAY	P	**S**	延迟指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	–

符号：

S ： 延迟时间

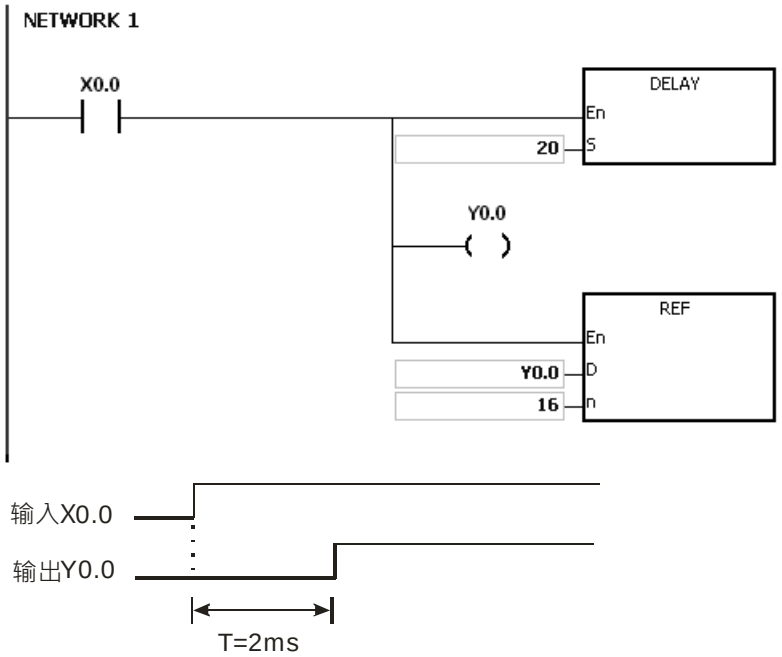
指令说明：

执行 DELAY 指令后，在每次扫描周期 DELAY 指令后面的程序执行会依用户指定的时间作延迟。

S 延迟时间的单位：0.1ms（范围:1~1000）

程序范例：

当 X0.0=ON 时，执行 DELAY 指令延迟 2ms 后才执行后面程序，Y0.0 导通且更新 Y0.0~Y0.15 的状态。



补充说明：

1. **S**<=0 时，不会有延迟时间。
2. **S**>1000，指令不执行，SM0=ON，错误码 SR0=16#2003。

3. 用户可依实际状况来调整延迟时间。
4. DELAY 指令会受到通讯或其它影响而增加延迟时间。

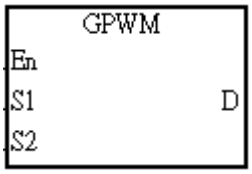
API	指令码	操作数	功能
1902	GPWM	**S_1**，**S_2**，**D**	基本脉冲宽度调变

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●			○	○				
S_2		●			●	●		●								
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
—	AS	—

符号：



S_1 ： 脉冲输出宽度

S_2 ： 脉冲输出周期

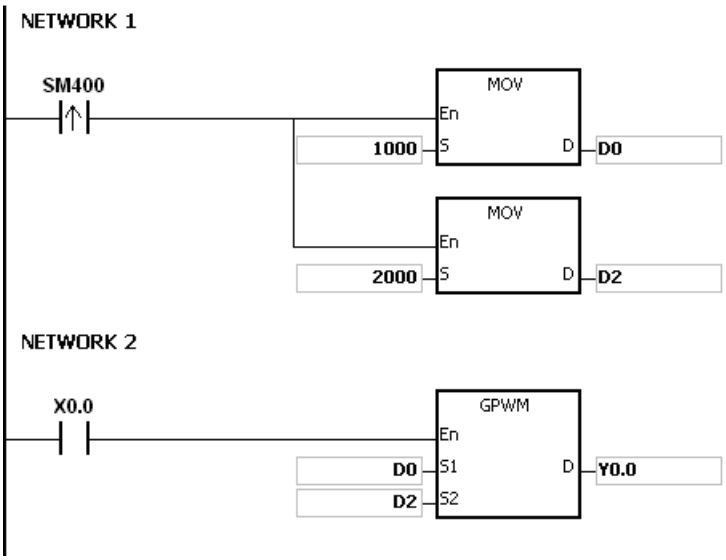
D ： 脉冲输出装置

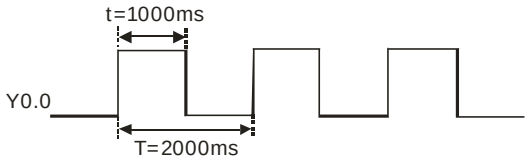
指令说明：

1. GPWM 指令执行时，指定 **S_1** 脉冲输出宽度与 **S_2** 脉冲输出周期，并由 **D** 脉冲输出装置输出。
2. **S_1** 脉冲输出宽度（duty-on）指定 t：0~32，767ms。
3. **S_2** 脉冲输出周期（cycle）指定为 T：1~32，767ms，但 **$S_1 \leqq S_2$**。
4. **S_2**+1、**S_2**+2 为系统使用参数，请勿占用。
5. 当 **$S_1 \leqq 0$** 时，脉冲输出装置无输出，当 **$S_1 \geqq S_2$** 时，脉冲输出装置一直为 ON。
6. **S_1**、**S_2** 可在 GPWM 指令执行时更改。
7. 当条件接点未启动时，**D** 装置输出 0。
8. 使用在线编辑时，请重新启动条件接点，以达到指令的初始化。

程序范例：

程序启动时，D0=1000，D2=2000，当 X0.0=ON 时，Y0.0 输出调变的脉冲，当 X0.0=OFF 时，Y0.0 输出也变成 OFF。





补充说明：

1. 此指令是以扫描周期去计数，因此最大误差为 1 个 PLC 扫描周期。**S_1**、**S_2** 与（**S_2-S_1**）的值必须>PLC 扫描周期，否则 GPWM 输出会有误动作。
2. 若将此指令置于功能块或中断中使用，则会产生 GPWM 输出不准确的情况发生，请特别注意。
3. **S_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。

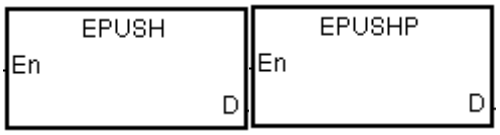
API	指令码			操作数	功能
1904		EPUSH	P	**D**	变址寄存器存入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	–

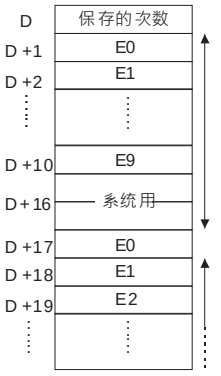
符号：



D ： 要存放变址寄存器的装置编号

指令说明：

1. 将 E0~E9 的内容值存放到 **D** 的内容（保存的次数）所指定的装置。此 **D** 内容值范围为 0~99，超出范围不执行。
2. 每次执行存放需 16 个装置(最后 6 个为系统保留用)，因此若保存次数为 n(**D** 的内容值)，则执行时 E0~E9 会存放在 **D**+（16*n+1）~**D**+（16*n+16）中，且将保存次数累加 1 次（**D**=n+1）。
3. **D** 的容量请保留有 16x100+1 个。
4. 此指令系采用脉冲执行型方式执行堆栈动作，故须执行下一次堆栈动作时，请重新切换条件接点。
5. 配合 EPOP 指令使用，可以达到 E 装置的堆栈存入与读出的功能。



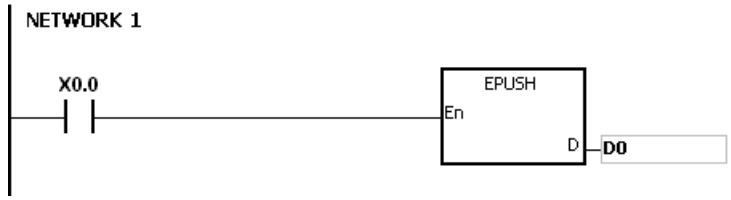
程序范例：

给定 D0=0，当第一次 X0.0=ON 时，E0~E9 的内容会分别传送至 D1~D10，D0=1，

第二次 X0.0 由 OFF→ON 时，E0~E9 的内容会分别传送至 D17~D26，D0=2。

第 n 次 X0.0 由 OFF→ON，E0~E9 的内容会分别传送至：

D 装置编号为（n*16）+1~（n*16）+10



补充说明：

1. **D** 内容值<0 或>99 时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **D**+（（**D** 内容值+1）*16-1）装置容量不足时，指令不执行，SM0=ON，错误码 SR0=16#2003。

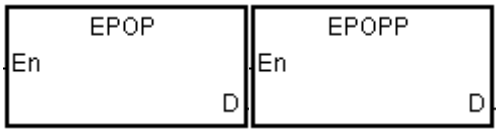
API	指令码			操作数	功能
1905		EPOP	P	**D**	变址寄存器读出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	–

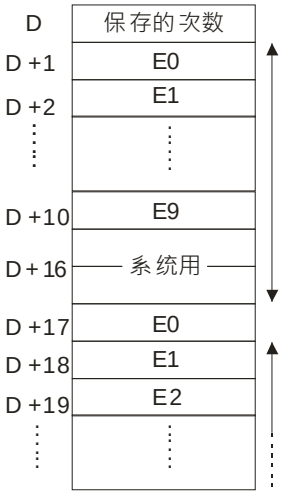
符号：



D ： 要读出内容值到变址寄存器的装置编号

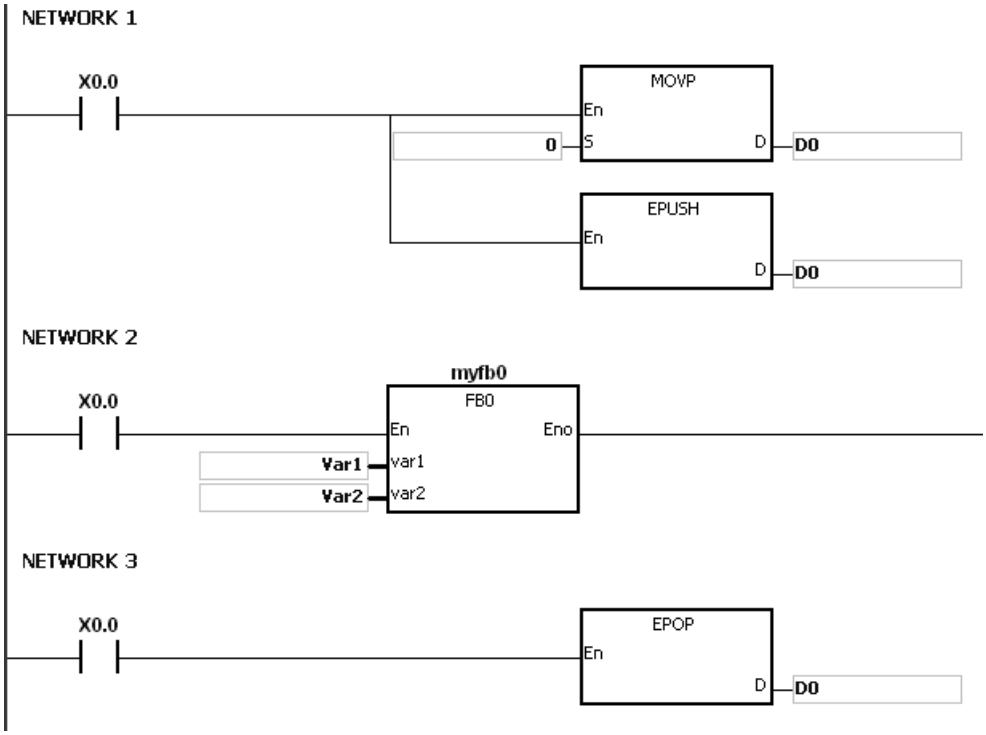
指令说明：

1. 将 **D** 的内容值（保存的次数）减 1，并将 **D** 的内容值读出到装置 E0~E9。此 **D** 内容值范围为 1~100。
2. 每次执行读取需 16 个装置（后 6 个装置为系统保留用），因此若保存次数为 n（D 的内容值），则该次会读取 D+16*（n-1）+1~D+16*（n-1）+10 中的内容值写入 E0~E9，且保存次数 D=n-1。
3. 此指令系采用脉冲执行型方式执行堆栈动作，故须执行下一次堆栈动作时，请重新切换条件接点。
4. 取出的顺序为最后放入 **D** 中的值。



程序范例：

当 X0.0=ON 时，先将 D0 内容值设为 0，然后使用 EPUSH 将 E0~E9 的内容分别传送至 D1~D10，再开始执行功能块 FB0，当功能块 FB0 执行完毕之后，再用 EPOP 将 D1~D10 的值读出至 E0~E9。



补充说明：

1. **D** 内容值<=0 或>100 时，指令不执行，SM0=ON，错误码 SR0=16#2003。
2. **D+**（**D** 内容值*16-1）装置容量不足时，指令不执行，SM0=ON，错误码 SR0=16#2003。

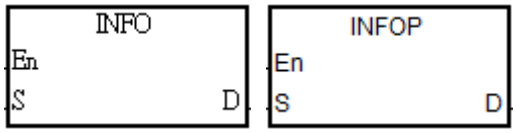
API	脚本			操作数	功能
1906		INFO	P	**S**・**D**	系统讯息读出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S													○			
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	–

符号：



S ： 系统讯息读出代码

D ： 系统讯息读出存放的起始装置位置

指令说明：

1. **S** 为系统讯息代码，其代码如下表所示：

S 代码编号	代码说明	**D** 连续装置数（Word 数目）	备注	支持版本
0	读出产品序号	9	ASCII 码	V1.04 以上
1	以太网络 MAC 地址	3	Hex 数值	V1.04 以上
2	产品已上电时间	2	32 位数值，单位：分钟	V1.04 以上
其他	保留	0	不执行	

2. 假设产品序号为 AS332T-AW16360001，且指令读出代码为 0 以及指定存放至 D0，则讯息存放顺序将如下表所示：

D0	D1	D2	D3	D4	D5	D6	D7	D8
16#5341	16#3333	16#5432	16#412D	16#3157	16#3336	16#3036	16#3030	16#0031
‘SA’	‘33’	‘T2’	‘A-‘	‘1W’	‘36’	‘06’	‘00’	‘1’

3. 假设 MAC 为 16#123456789ABC，且指令读出代码为 1 以及指定存放至 D10，则讯息存放顺序将如下表所示：

D10	D11	D12
16#1234	16#5678	16#9ABC

4. 假设已上电时间为 70000 分钟（16#00011170），指令读出代码为 2，以及指定存放至 D20，则上电时间讯息存放顺序将如下表所示：

D20	D21
16#1170	16#0001

5. 此指令读取讯息之动作，只有在条件接点成立的第一次执行时有效，因此建议使用脉冲执行型的 INFOP 指令；若采用 INFO 指令且搭配常开接点，则指令将会自动仅执行一次。

补充说明：

D 采用变量宣告时，代码 0，1，2 时，请分别宣告为 ARRAY[9]，[3]，[2] of WORD。

6.21 字符串处理指令

6.21.1 字符串处理指令一览表

API	指令码（位）		P 指令	功能
	16	32		
2100	BINDA	DBINDA	✓	有号数十进制→ASCII 变换
2101	BINHA	DBINHA	✓	BIN 十六进制→十六进制 ASCII 变换
2102	BCDDA	DBCDDA	✓	BCD→ASCII 变换
2103	DABIN	DDABIN	✓	有号数十进制 ASCII→有号数十进制 BIN 变换
2104	HABIN	DHABIN	✓	十六进制 ASCII→十六进制 BIN 变换
2105	DABCD	DDABCD	✓	BCD ASCII→BCD 变换
2106	$LEN	–	✓	计算字符串长度
2109	$FSTR	–	✓	Float→String
2110	$FVAL	–	✓	String→Float
2111	$RIGHT	–	✓	从右边获取字符串
2112	$LEFT	–	✓	从左边获取字符串
2113	$MIDR	–	✓	区段获取字符串
2115	$SER	–	✓	字符串搜寻
2116	$RPLC	–	✓	字符串取代
2117	$DEL	–	✓	指定字符串删除
2118	$CLR	–	✓	字符串清除
2119	$INS	–	✓	字符串插入
2122	SPLIT	–	✓	字符串分离
2123	MERGE	–	✓	字符串合并

6.21.2 字符串处理指令说明

API	指令码			操作数	功能
2100	D	BINDA	P	**S**、**D**	有号数十进制→ASCII 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

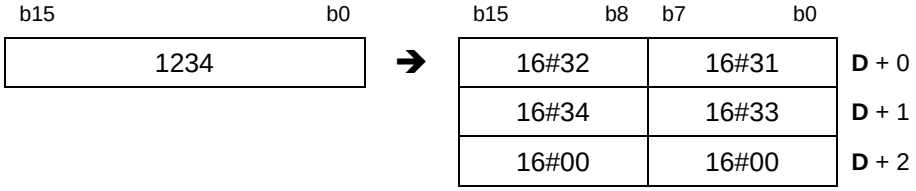
符号：

S ： 转换的来源值

D ： 存放转换结果的装置

指令说明：

1. 将有号数十进制 BIN 值的数据源 **S** 做 ASCII 的转换，并储存于 **D**。
2. 本指令支持字符串结束模式标志 SM690，来控制字符串的终止符。
3. 16 位指令中，S 为 16 位有号数的 BIN 值，其范围-32768~32767；**D** 最多连续占用四个 Word 装置，数据转换方式如图说明：



其中 SM690=OFF 时，**D**+2 填入字符串结束模式 16#0000；SM690=ON 时，不填入字符串结束模式，因此 **D**+2 的值保持不变。

如果 **S** 的内容值为正数，则 **D** 的符号不会被填入，直接填入数值，如上述范例；如果 **S** 的内容值为负数，**D** 的符号”-”会被填入 16#2D。如下范例：S=-1234 且 SM690=OFF 则转换后的值如下说明：

b15 b0

-1234

➔

b15 b8	b7 b0	
16#31	16#2D (-)	**D** + 0
16#33	16#32	**D** + 1
16#00	16#34	**D** + 2

4. 32 位指令中，**S** 为 32 位有号数的 BIN 值，其范围-2147483648~2147483647；**D** 最多连续占用六个 Word 装置，数据转换方式如图说明：

b31 b0

12345678

➔

b15 b8	b7 b0	
16#32	16#31	**D** + 0
16#34	16#33	**D** + 1
16#36	16#35	**D** + 2
16#38	16#37	**D** + 3
16#00	16#00	**D** + 4

其中 SM690=OFF 时，**D**+4 填入 16#0000；SM690=ON 时，**D**+4 不变化。

如果 **S** 的内容值为正数，则 **D** 的符号不会被填入，直接填入数值，如上述范例；如果 **S** 的内容值为负数，则 **D** 的符号"-"会被填入 16#2D。如下范例：S=-12345678 且 SM690=OFF 则转换后的值如下说明：

b31 b0

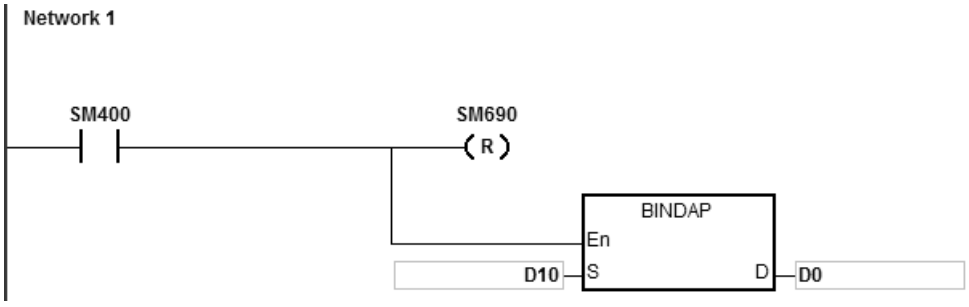
-12345678

➔

b15 b8	b7 b0	
16#31	16#2D	**D** + 0
16#33	16#32	**D** + 1
16#35	16#34	**D** + 2
16#37	16#36	**D** + 3
16#00	16#38	**D** + 4

程序范例一：

给定 D10=5126 且 SM690=OFF，PLC RUN 时 D0=16#3135、D1=16#3632、D2=16#0000。



D10 b15 b8 b7 b0

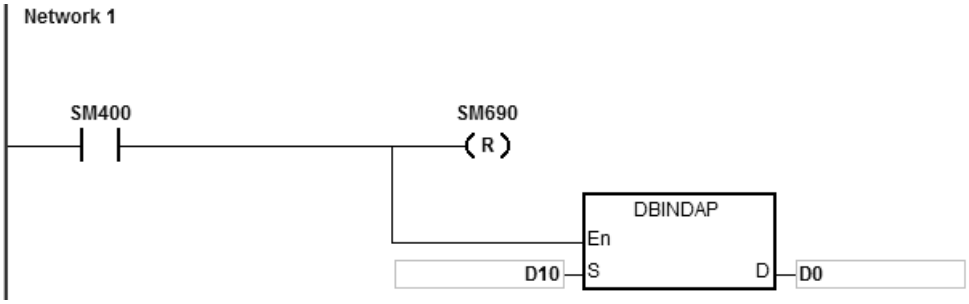
5126

➔

b15 b8	b7 b0	
16#31	16#35	D0
16#36	16#32	D1
16#00	16#00	D2

程序范例二：

给定 D10=-3842563 且 SM690=OFF，PLC RUN 时 **D**0=16#332D、D1=16#3438、D2=16#3532、**D**3=16#3336、**D**4=16#0000。



D10 = -3842563 ➔

b15 b8	b7 b0	
16#33	16#2D	D0
16#34	16#38	D1
16#35	16#32	D2
16#33	16#36	D3
16#00	16#00	D4

补充说明：

1. 若 **D** 操作数的装置不足填入转换的值时，SM0=ON，错误码 SR0=16#2003。
2. 16 位指令 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of WORD/INT。
3. 32 位指令 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [6] of WORD/INT。

API	指令码			操作数	功能
2101	D	BINHA	P	**S**，**D**	BIN 十六进制→十六进制 ASCII 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：

S ： 转换的来源值

D ： 存放转换结果的装置

指令说明：

1. 将十六进制 BIN 值的数据源 **S** 做 ASCII 的转换，并储存于 **D**。
2. 本指令支持字符串结束模式标志 SM690，来控制字符串的终止符。
3. 16 位指令中，**S** 的范围 16#0000~16#FFFF，且固定为 4 位数的 BIN 值；**D** 连续占用三个 Word 装置，数据转换方式如下说明：

b15 ~ b0		b15 ~ b8	b7 ~ b0	
02A6	➔	16#32	16#30	**D** + 0
		16#36	16#41	**D** + 1
		16#00	16#00	**D** + 2

其中 SM690=OFF 时，D+2 填入字符串结束模式 16#0000；SM690=ON 时，不填入字符串结束模式，因此 D+2 的值保持不变。

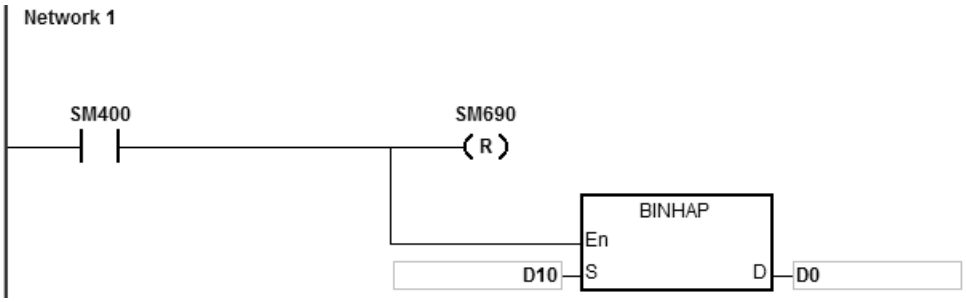
4. 32 位指令中，**S** 的范围 16#00000000~16#FFFFFFFF，且固定为 8 位数的 BIN 值；**D** 连续占用五个 Word 装置，数据转换方式如下说明：

b31 ~ b0		b15 ~ b8	b7 ~ b0	
03AC 6251	➔	16#33	16#30	**D** + 0
		16#43	16#41	**D** + 1
		16#32	16#36	**D** + 2
		16#31	16#35	**D** + 3
		16#00	16#00	**D** + 4

其中 SM690=OFF 时，D+4 填入字符串结束模式 16#0000；SM690=ON 时，不填入字符串结束模式，因此 D+4 的值保持不变。

程序范例一：

给定 D10=16#9C06 且 SM690=OFF，PLC RUN 时 D0=16#4339、D1=16#3630、D2=16#0000。



D10

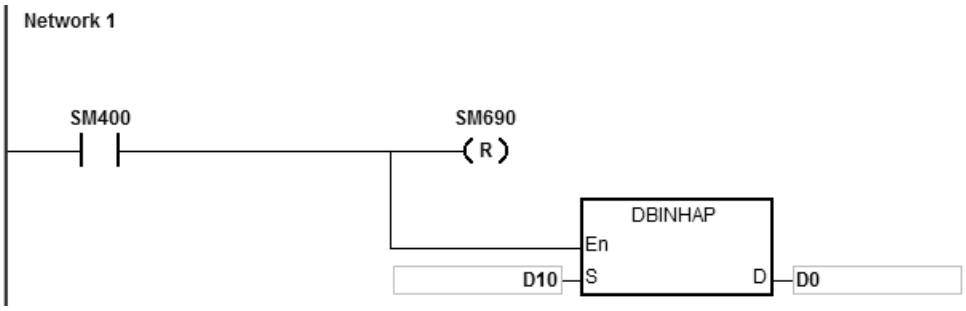
16#9C06

➔

b15 b8	b7 b0	
16#43	16#39	D0
16#36	16#30	D1
16#00	16#00	D2

程序范例二：

给定 D10=16#7B3C581F 且 SM690=OFF，PLC RUN 时 D0=16#4237、D1=16#4333、D2=16#3835、D3=16#4631、D4=16#0000。



D10

16#7B3C 581F

➔

b15 b8	b7 b0	
16#42	16#37	D0
16#43	16#33	D1
16#38	16#35	D2
16#46	16#31	D3
16#00	16#00	D4

补充说明：

1. 16 位指令中 **D**+2 装置，超过装置范围时，SM0=ON，错误码 SR0=16#2003。
2. 32 位指令中 **D**+4 装置，超过装置范围时，SM0=ON，错误码 SR0=16#2003。
3. 16 位指令 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
4. 32 位指令 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [5] of WORD/INT。

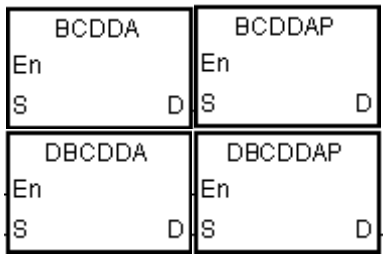
API	指令码			操作数	功能
2102	D	BCDDA	P	**S**，**D**	BCD→ASCII 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 转换的来源值

D ： 存放转换结果的装置

指令说明：

1. 将 BCD 值的数据源 **S** 做 ASCII 的转换，并储存于 **D**。
2. 本指令支持字符串结束模式标志 SM690，来控制字符串的终止符。
3. 16 位指令中，**S** 的范围 BCD 0~9999，且固定为 4 位数的 BCD 值；**D** 连续占用三个 Word 装置，数据转换方式如图说明：

b15~12	b11~8	b7~4	b3~0
0	8	7	6
千	百	十	个

➔

b15 ~ b8	b7 ~ b0	
16#38	16#30	**D** + 0
16#36	16#37	**D** + 1
16#00	16#00	**D** + 2

其中 SM690=OFF 时，**D**+2 填入字符串结束模式 16#0000；SM690=ON 时，不填入字符串结束模式，因此 **D**+2 的值保持不变。

4. 32 位指令中，**S** 的范围 BCD 0~99999999，且固定为 8 位数的 BCD 值；**D** 连续占用五个 Word 装置，数据转换方式如图说明：

b31 ~ b0
0876 5432

➔

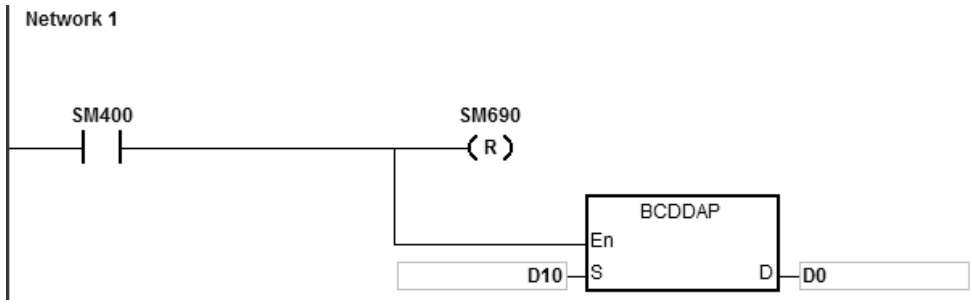
b15 ~ b8	b7 ~ b0	
16#38	16#30	**D** + 0
16#36	16#37	**D** + 1
16#34	16#35	**D** + 2
16#32	16#33	**D** + 3
16#00	16#00	**D** + 4

其中 SM690=OFF 时，**D**+5 填入字符串结束模式 16#0000；SM690=ON 时，不填入字符串结束模式，因此 **D**+5 的值保持不变。

5. **S** 的 BCD 码最大位数即使为 0，也将强制被转换为字符 0（16#30）。

程序范例一：

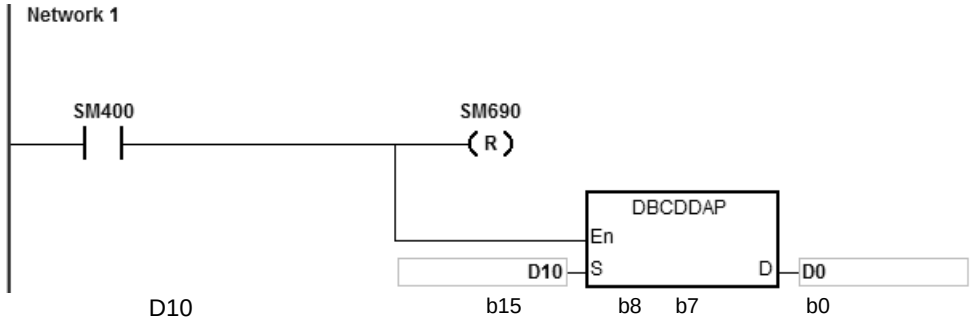
假设 D10 的 BCD 数值为 1295 且 SM690=OFF，PLC RUN 时 D0=16#3231、D1=16#3539、D2=16#0000。



D10	b15 b8	b7 b0	
1295 →	16#32	16#31	D0
	16#35	16#39	D1
	16#00	16#00	D2

程序范例二：

假设 D10 的 BCD 数值为 00578352 且 SM690=OFF，PLC RUN 时 D0=16#3030、D1=16#3735、D2=16#3338、D3=16#3235、D4=16#0000。



D10	b15 b8	b7 b0	
0057 8352 →	16#30	16#30	D0
	16#37	16#35	D1
	16#33	16#38	D2
	16#32	16#35	D3
	16#00	16#00	D4

补充说明：

1. 16 位指令中，若 **S** 内容值不在 0~9999，指令不执行，SM0=ON，错误码 SR0=16#200D。(BCD 值以 Hex 表示有任一位数不在 0~9 的范围内)。
2. 32 位指令中，若 **S** 内容值不在 0~99999999，指令不执行，SM0=ON，错误码 SR0=16#200D。(BCD 值以 Hex 表示有任一位数不在 0~9 的范围内)。
3. 16 位指令中 **D**+2 装置，超过装置范围时，SM0=ON，错误码 SR0=16#2003。
4. 32 位指令中 **D**+4 装置，超过装置范围时，SM0=ON，错误码 SR0=16#2003。
5. 16 位指令 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
6. 32 位指令 **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [5] of WORD/INT。

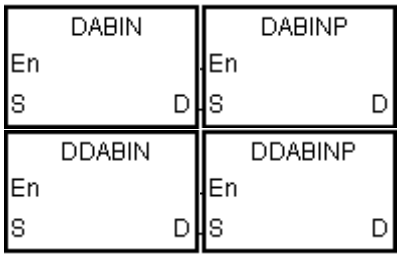
API	指令码			操作数	功能
2103	D	DABIN	P	**S**、**D**	有号数十进制 ASCII→有号数十进制 BIN 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S		●			●	●		●	●						○	
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 转换的来源值

D ： 存放转换结果的装置

指令说明：

1. 将有号十进制 ASCII 值的数据源 **S** 转换成有号数十进制 BIN，并储存于 **D**。
2. 16 位指令中，**S 最多**连续占用三个 Word 装置，且组成的 **S** 内容值范围 ASCII 为-32768~32767；若 **S** 来源输入字符串无结尾 16#00 数值，则最大转换至 5 位数（不含符号）。

	b15 b8	b7 b0
S + 0	" 2 "	" 1 "
S + 1	" 4 "	" 3 "
S + 2	16#00	16#00

➔

b15 b0	
1234	**D**

3. 16 位指令中，**S** 来源为字符串；若第一个字符为数字，即表示正数；若为" - "符号则表示负数。如下范例 "-2345"：

	b15 b8	b7 b0
S + 0	" 2 "	" - "
S + 1	" 4 "	" 3 "
S + 2	16#00	" 5 "

➔

b15 b0	
-2345	**D**

4. 32 位指令中，**S 最多连续**占用六个 Word 装置，且组成的 **S** 内容值范围 ASCII -2147483648~2147483647；若 **S** 来源输入字符串无结尾 16#00 数值，则最大转换至 10 位数（不含符号）。

	b15 b8	b7 b0		b31 b0	
S + 0	“ 2 “	“ 1 “	➔	1234567890	**D**
S + 1	“ 4 “	“ 3 “			
S + 2	“ 6 “	“ 5 “			
S + 3	“ 8 “	“ 7 “			
S + 4	“ 0 “	“ 9 “			
S + 5	16#00	16#00			

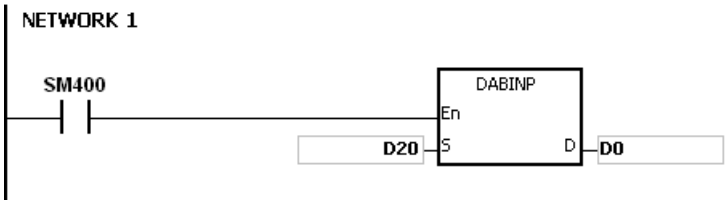
5. 32 位指令中，**S** 来源为字符串，若第一个字符为数字，即表示正数；若为’ - ‘符号则表示负数。如下范例 “-234567890”：

	b15 b8	b7 b0		b31 b0	
S + 0	‘ 2 ‘	‘ - ’	➔	-234567890	**D**
S + 1	‘ 4 ‘	‘ 3 ‘			
S + 2	‘ 6 ’	‘ 5 ‘			
S + 3	‘ 8 ‘	‘ 7 ‘			
S + 4	‘ 0 ’	‘ 9 ‘			
S + 5	16#00	16#00			

6. **S** 字符串内容第 1 个数字未出现时，若遇到空格符（16#20）或’ + ‘符号（16#2B）时，以’0’字符处理，之后第 2 个数字遇到非 0~9 的字符，皆当做字符串结尾处理，且不会显示错误讯息。例如：字符顺序为 16#20→16#31→16#32→16#2B，则转换数值为 12。
7. **S** 来源选择字符串装置时，16 位指令长度范围为 1~6（含正/负号），32 位指令为 1~11（含正/负号）。
8. 32 位指令可使用 HC 装置，但不可使用 E 装置。

程序范例一：

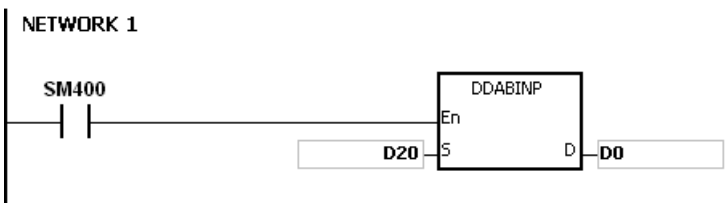
假设 D20=16#302D、D21=16#3433、D22=16#0035，PLC RUN 时，D0=-345。



	b15 b8	b7 b0		b15 b0	
D20	16#30	16#2D	➔	-345	D0
D21	16#34	16#33			
D22	16#00	16#35			

程序范例二：

给定 D20=16#322D、D21=16#3433、D22=16#3635、D23=16#2037、D24=16#0000、D25=16#0000，PLC RUN 时，D0=-234567。



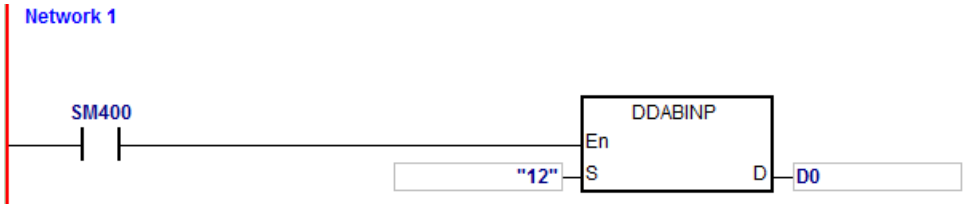
	b15 b8	b7 b0
D20	16#32	16#2D
D21	16#34	16#33
D22	16#36	16#35
D23	16#20	16#37
D24	16#00	16#00
D25	16#00	16#00

➔

b31 b0	
-234567	D0

程序范例三：

S 数据源为”12”（String 类型），PLC RUN 后 D0=12。



补充说明：

1. **S** 所指定的内容值中第一个字为终止符（16#00）时，则指令自动将 **S** 视为’0’字符（16#30）处理。
2. **S** 字符串内容第一个数字为空格符（16#20）或’ + ‘符号（16#2B） 或’ - ‘符号（16#2D），第 2 个数字为 16#00 时，则指令自动将 **S** 视为’0’字符（16#30）处理。
3. **S** 所指定的内容值中第一个字仅支持 ASCII 码 16#30~16#39（0~9）、16#200（空白）、16#2D（负号）、16#2B（正号）、16#00（终止符），若非上述 ASCII 码，则指令不执行，SM0=ON，错误码 SR0=16#2003。
4. 若 **S** 所指定的内容值中第二个以上的字不是 ASCII 码 16#30~16#39 或 16#00 则指令自动以 16#00 处理。
5. **S** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，错误码 SR0=16#2003。
6. 指令中 **S** 装置，使用超过装置范围时，SM0=ON，错误码 SR0=16#2003。
7. 16 位指令 **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [3] of WORD/INT。
8. 32 位指令 **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [6] of WORD/INT。

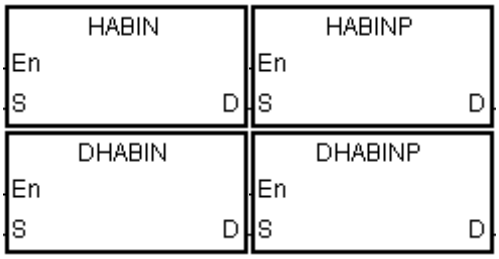
API	指令码			操作数	功能
2104	D	HABIN	P	**S**，**D**	十六进制 ASCII→十六进制 BIN 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S		●			●	●		●	●						○	
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 转换的来源值

D ： 存放转换结果的装置

指令说明：

1. 数据源 **S** 的内容十六进制（ASCII 值）转换成十六进制 BIN，存于 **D**。
2. 16 位指令 **S 最多**连续占用二个 Word 装置，若 **S** 来源输入字符串无结尾 16#00 数值，则最大转换至 4 位数，**S** 内容值范围 ASCII 0000~FFFF；若 **S** 来源如输入字符串装置，范围为“0”~“FFFF”。

	b15 b8	b7 b0		b15 b0	
S + 0	“ A “	“ 5 “	➔	16#5A8D	**D**
S + 1	“ D “	“ 8 “			

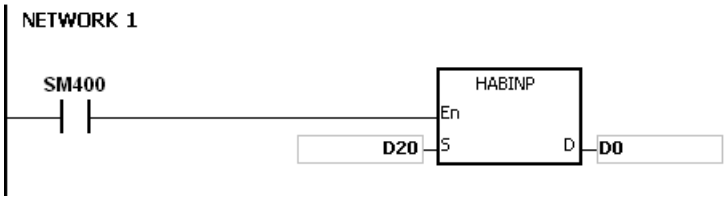
3. 32 位指令 **S 最多**连续占用四个 Word 装置，若 **S** 来源输入字符串无结尾 16#00 数值，则最大转换至 8 位数，**S** 内容值范围 ASCII 00000000~FFFFFFFF；若 **S** 来源如输入字符串装置，范围为“0”~“FFFFFFFF”。

	b15 b8	b7 b0		b31 b0	
S + 0	“ C ”	“ 5 “	➔	16#5CB8 07E1	**D**
S + 1	“ 8 “	“ B “			
S + 2	“ 7 “	“ 0 “			
S + 3	“ 1 “	“ E “			

4. **S** 来源为字符串装置时，16 位指令长度为 1~4，32 位指令长度为 1~8。

程序范例一：

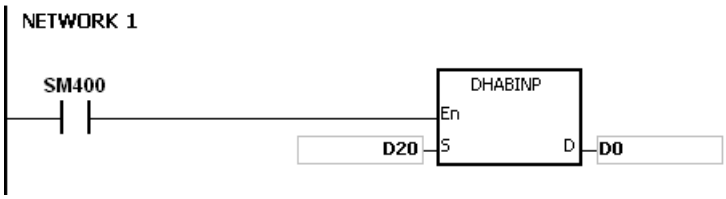
给定 D20=16#3641、D21=16#4633（即 ASCII 16#A63F），PLC RUN 时，D0=-22977



	b15 b8	b7 b0		b15 b0	
D20	16#36	16#41	➔	16#A63F	D0
D21	16#46	16#33			

程序范例二：

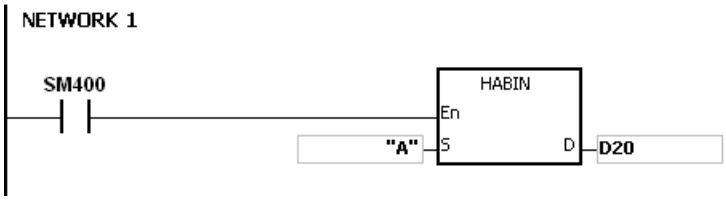
给定 D20=16#4634、D21=16#3244、D22=16#3738、D23=16#3035（即 ASCII 16#4FD28750），PLC RUN 时，（D1，D0）=16#4FD28750。



	b15 b8	b7 b0		b31 b0	
D20	16#46	16#34	➔	16#4FD2 8750	D0
D21	16#32	16#44			
D22	16#37	16#38			
D23	16#30	16#35			

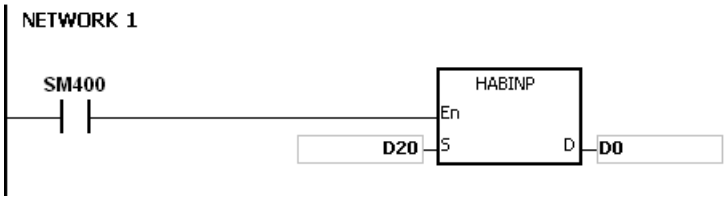
程序范例三：

S 数据源为“A”（String 类型），当 PLC RUN 后 D20=16#A=10。



程序范例四：

给定 D20=16#3641、D21=16#4600（即 ASCII 16#00A6），PLC RUN 时，D0=166



	b15 b8	b7 b0		b15 b0	
D20	16#36	16#41	➔	16#00A6	D0
D21	16#46	16#00			

补充说明：

1. **S** 内容值不是 ASCII 码 16#30~ 16#39（“0”~”9”）或 16#41~ 16#46（“A”~”F”），指令不执行，SM0=ON，错误码 SR0=16#2003。
2. 16 位指令 **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。
3. 32 位指令 **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of WORD/INT。

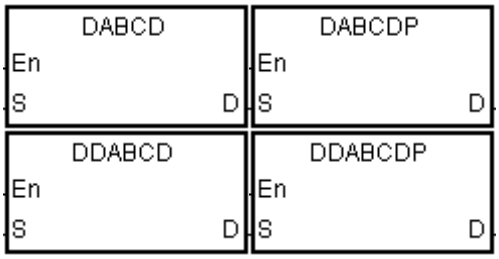
API	指令码			操作数	功能
2105	D	DABCD	P	**S**，**D**	BCD ASCII→BCD 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S		●			●	●		●	●						○	
D		●			●	●	●	●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D		●	●			●	●						

脉冲执行型	16 位指令	32 位指令
AS	AS	AS

符号：



S ： 转换的来源值

D ： 存放转换结果的装置

指令说明：

1. 数据源 **S** 的内容（ASCII 值）转换成 BCD，存于 **D**。
2. **16** 位指令 **S 最多**连续占用二个 Word 装置，**S** 内容值范围 ASCII 0000~9999；若 **S** 来源如输入字符串，范围为"0"~"9999"。

	b15 b8	b7 b0
S + 0	" 8 "	" 9 "
S + 1	" 6 "	" 7 "

➔

b15~b12	b11~b8	b7~b4	b3~b0	
9	8	7	6	**D**
千	百	十	个	

3. **32** 位指令 **S 最多**连续占用四个 Word 装置，**S** 内容值范围 ASCII 0000000~99999999；若 **S** 来源如输入字符串，范围为"0"~"99999999"。例如，**S**~**S**+3 的内容分别为 ASCII 码 98765432，则转换结果如下图所示。

	b15 b8	b7 b0
S + 0	" 8 "	" 9 "
S + 1	" 6 "	" 7 "
S + 2	" 4 "	" 5 "
S + 3	" 2 "	" 3 "

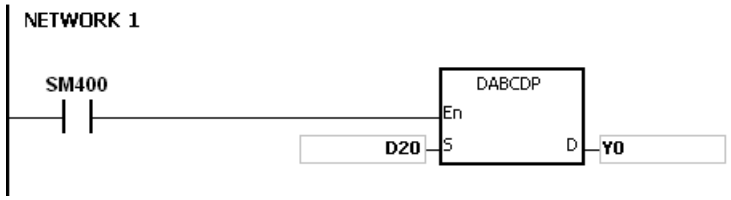
➔

b15~b12	b11~b8	b7~b4	b3~b0	
5	4	3	2	**D**
9	8	7	6	**D**+1

4. **S** 内容为 16#20 时，将当做 16#30 来做处理；若为 16#00 时，将当做终止符处理。
5. **S** 来源为字符串装置时，16 位指令长度为 1~4，32 位指令长度为 1~8。

程序范例一：

设定 D20=16#3820、D21=16#3637（即 ASCII 876），PLC RUN 时，转换为 Y0=16#876。



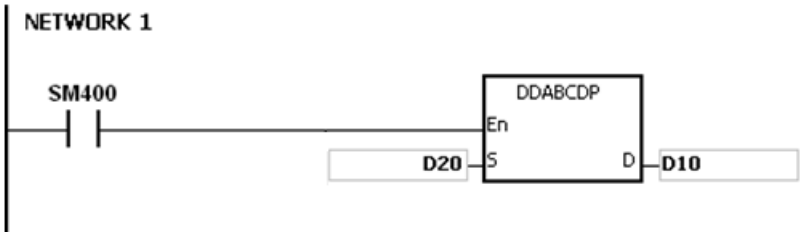
	b15 b8	b7 b0
D20	16#38	16#20
D21	16#36	16#37

➔

b15~b12	b11~b8	b7~b4	b3~b0	
0	8	7	6	Y0

程序范例二：

给定 D20=16#3738、D21=16#3536、D22=16#3334、D23=16#3132（即 ASCII 87654321），PLC RUN 时，转换为（D11，D10）=16#87654321。



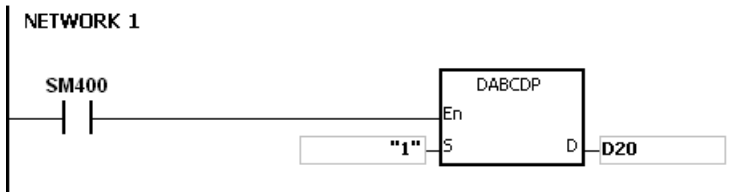
	b15 b8	b7 b0
D20	16#37	16#38
D21	16#35	16#36
D22	16#33	16#34
D23	16#31	16#32

➔

b31 b0	
8765 4321	D0

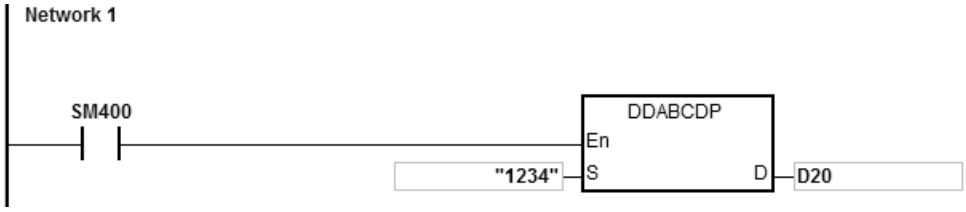
程序范例三：

S 数据源为“1”（String 类型），PLC RUN 时，转换为 D20=16#0001。



程序范例四：

S 数据源为“1234”（String 类型），当 PLC RUN 时，转换为（D21，D20）=16#00001234。



补充说明：

1. **S** 内容值不是 ASCII 码 16#30~16#39，16#20 与 16#00，则指令不执行，且 SM0=ON，SR0 记录错误码 16#2003。
2. **S** 字符串长度超出范围时，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
3. 16 位指令 **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。
4. 32 位指令 **S** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [4] of WORD/INT。

API	指令码			操作数	功能
2106		$LEN	P	**S**・**D**	计算字符串长度

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●								
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

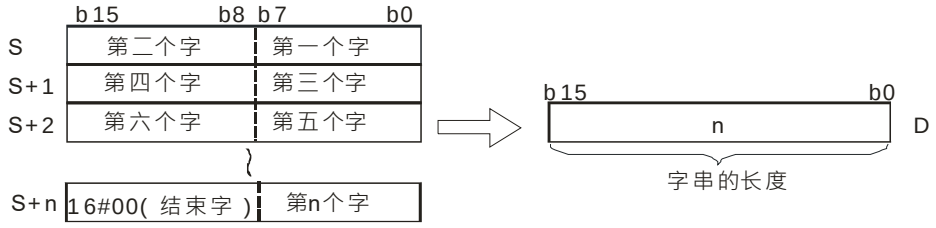
符号：

S ： 字符串来源

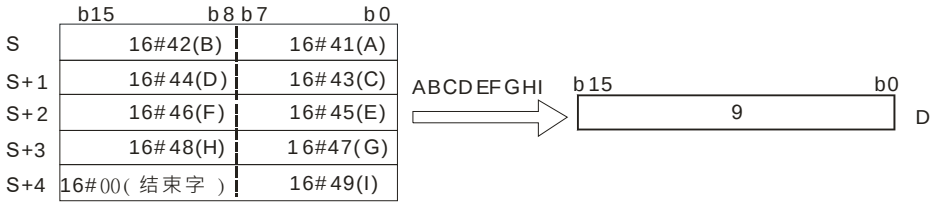
D ： 字符串长度

指令说明：

1. 计算数据源 **S** 的内容（字符串）长度，一直计算到字符串结尾 16#00 之前的长度，并将字符串长度存于 **D**。
2. 字符串长度=0~32767，如果字符串长度>=32767，则 D=32767。



例如 **S**~**S+n** 的内容为 ABCDEFGHI，则计算结果如下图所示。

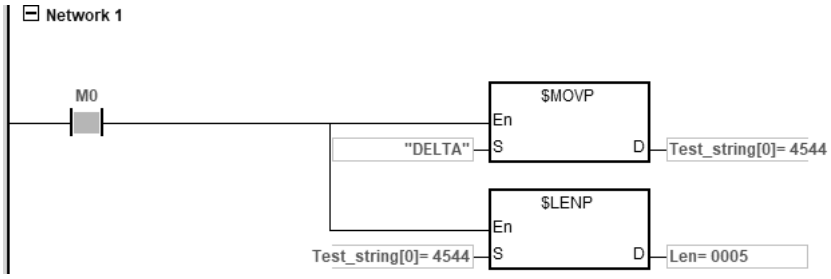


程序范例：

1. 宣告两个变量 Test_string 及 Len

Local Symbols				
Class	Identifiers	Address	Type	Initial Value (Acti...
VAR	Test_string	N/A [Auto]	ARRAY [10] OF WORD	N/A
VAR	Len	N/A [Auto]	WORD	N/A

2. 当 M0 由 OFF→ON，指令$LEN 将字符串长度计算后放存放到 LEN



补充说明：

1. 若字符串的字尾无 16#00 当做结束符号，则指令执行到最大 32767，即结束执行。
2. 若计算长度过程中，遇到超出装置范围时，则当做结束符号处理。

API	指令码			操作数	功能
2109		$FSTR	P	**S_1 · S_2 · D**	Float→String

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●	●							○
S_2	●	●			●	●		●	●							
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S_1 ： 被转换的来源浮点数

S_2 ： 转换的格式与位数设定

D ： 存放转换字符串的装置起始编号

指令说明：

1. 依据 **S_2** 的格式与位数设定，将 **S_1** 的浮点值转换成字符串，并将结果存于 **D**。
2. 将 **S_1** 的浮点数转换成浮点数字符串，并加上结尾字符（16#00）后，将结果存于 **D**。

（S_1+1，S_1）

浮点数

⇨

	上 8 位	下 8 位
D	第二个的 ASCII 码	第一个的 ASCII 码
D + 1	第四个的 ASCII 码	第三个的 ASCII 码
D + 2	第六个的 ASCII 码	第五个的 ASCII 码
……	……	
D +n-1	第 m 个的 ASCII 码	第 m-1 个的 ASCII 码
D +n		16#00 终止符

浮点数字符串

3. 转换的结果依据 **S_2** 的显示设置而有所不同。
4. 总位数 **S_2**+1 可以指定的范围为 2≦**S_2**+1≦20，其包含正负符号、整数字与小数点之后位数，但不包含小数点与指数部份。

操作数	说明
S_2	0: 小数格式 1: 指数格式
S_2+1	浮点数转换总位数

5. 如果来源浮点数 **S_1** 为正数，则 D 会从第一个 ASCII 码开始填入数值；若为负数，则会先填入 16#2D（-）后再从第二个 ASCII 码开始填入数值。

6. 当使用小数格式（**S_2**=0）时：

D运算元转换后的浮点数字串

符号	整数位数	.	小数位数

- 总位数 **S_2**+1 可以指定的范围为 2≦**S_2**+1≦20，其包含正负符号、整数字数与小数点之后位数，但不包含小数点。
- 举例说明 1：

  总位数 8 位，当浮点数为-1.2345678 跟 123456，则运算结果如下所示：

D 编号	浮点数 -1.2345678		浮点数 123456	
	上 8 位	下 8 位	上 8 位	下 8 位
D	16#31（1）	16#2D（-）	16#32（2）	16#31（1）
D + 1	16#32（2）	16#2E（.）	16#34（4）	16#33（3）
D + 2	16#34（4）	16#33（3）	16#36（6）	16#35（5）
D + 3	16#36（6）	16#35（5）	16#00 终止符	
D + 4	16#00 终止符	16#38（8）		

  如果浮点数转换后能在总位数 **S_2**+1 内表达出浮点数得大小，但浮点数的长度超出总位数 **S_2**+1，则超出的小数位数的数值将会被 4 舍 5 入至最后一位数。

  如果转换后的浮点数字符串不足总位数，也不会刻意补满。

- 举例说明 2：

  如果浮点数无法在指定的总位数（**S_2**+1）内显示出指定的数值大小时，则指令将自动以指数格式转换，举例说明：设总位数 5 位数，若浮点数 1234567，则其转换结果变成 1.2346E+06，存放结果如下表。

D 编号	浮点数 1234567	
D	16#2E（.）	16#31（1）
D + 1	16#33（3）	16#32（2）
D + 2	16#36（6）	16#34（4）
D + 3	16#2B（+）	16#45（E）
D + 4	16#32（6）	16#30（0）
D + 5	16#0000 终止符	

- 举例说明 3：

  如果浮点数无法在指定的总位数（**S_2**+1）内显示出指定的数值大小时，则指令将自动以指数格式转换，举例说明：设总位数 2 位数，若浮点数为 0.00012345，则其转换结果分别将变成 1.2E-04，存放结果如下表。

D 编号	浮点数 0.00012345	
D	16#2E（.）	16#31（1）

D 编号	浮点数 0.00012345	
D + 1	16#45（E）	16#32（2）
D + 2	16#30（0）	16#2D（-）
D + 3	16#00 终止符	16#34（4）

- 举例说明 4：

  如果浮点数转换后的绝对值<=10^{-5}（浮点数为纯小数），则指令将自动以指数格式转换，举例说明：设总位数 4 位数，若浮点数分别为 0.00001234，则其转换结果分别将变成 1.234E-05，存放结果如下表。

D 编号	浮点数 0.00001234	
D	16#2E（.）	16#31（1）
D + 1	16#33（3）	16#32（2）
D + 2	16#45（E）	16#34（4）
D + 3	16#30（0）	16#2D（-）
D + 4	16#00 终止符	16#35（5）

7. 当使用指数格式（S_2=1）时：

D运算元转换后的浮点数字串	符号	整数位数	.	小数位数	指数部份

- 总位数 S_2+1 可以指定的范围为 2≦S_2+1≦20，其包含正负符号、整数字数与小数点之后位数，但不包含小数点、指数部份；运算结果的总长度会自动把指数部份（4 位）与小数点（1 位）加入。
- 整数位数固定为 1 个位数。
- 指数部分：固定 4 个字符

  如果指数值为正数，则 **D** 的指数符号位置被填入 16#2B（+），为负数，填入 16#2D（-）。

  指数部份固定为 2 位数，如果只有 1 位数，则前一位被填入 16#30（“0”）。

  举例说明：

  总位数 8 位，浮数值为-123.456789 跟 123456，则运算结果如下所示：

D 编号	浮点数 -123.456789		浮点数 12345	
	上 8 位	下 8 位	上 8 位	下 8 位
D	16#31（1）	16#2D（-）	16#2E（.）	16#31（1）
D + 1	16#32（2）	16#2E（.）	16#33（3）	16#32（2）
D + 2	16#34（4）	16#33（3）	16#35（5）	16#34（4）
D + 3	16#36（6）	16#35（5）	16#2B（+）	16#45（E）
D + 4	16#45（E）	16#38（8）	16#34（4）	6#30（0）
D + 5	16#30（0）	16#2B（+）	16#00 终止符	
D + 6	16#00 终止符	16#32（2）		

  如果浮点数的长度超出总位数 S_2+1，则超出位数的数值将会被 4 舍 5 入至最后一位数。

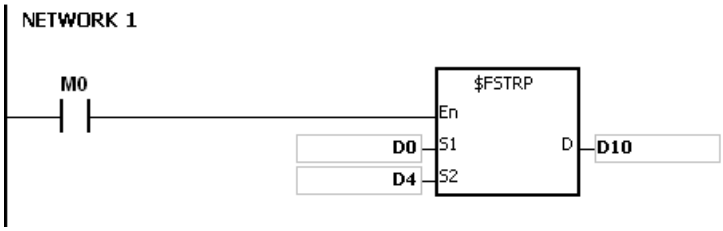
  总位数 8 位的浮点数-123.456789，转换后的字符串长度为 13（不包含终止符）

  总位数 8 位的浮点数 12345，转换后的字符串长度为 10（不包含终止符）

如果转换后的浮点数字符串不足总位数，也不会刻意补满。

程序范例一：

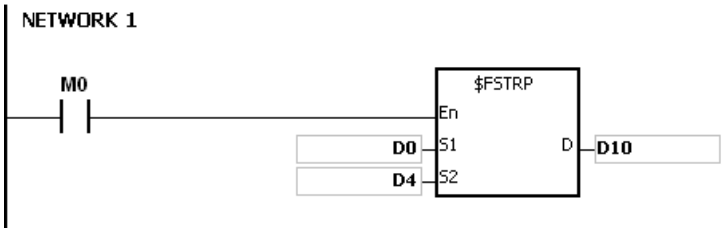
将浮点数内容（D1，D0= 12.3456）转为小数形式（D4=0，D5=8）的字符串，则转换结果如下：



D10	16#32（2）	16#31（1）
D11	16#33（3）	16#2E（.）
D12	16#35（5）	16#34（4）
D13	16#00 终止符	16#36（6）

程序范例二：

将浮点数内容（D1，D0 = 0.0012345678）转为指数形式（D4=1，D5=8）的字符串，则转换结果如下：



D10	16#2E（.）	16#31（1）
D11	16#33（3）	16#30（2）
D12	16#35（5）	16#34（4）
D13	16#37（7）	16#36（6）
D14	16#45（E）	16#38（8）
D15	16#30（0）	16#2D（-）
D16	16#00 终止符	16#33（3）

补充说明：

1. 当 **S_1** 超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。
2. 当 **S_2** 符号所指定的值不是 0 也不是 1，指令不执行，SM0=ON，错误码 SR0=16#2003。
3. 当总位数 **S_2**+1 超出 2≦**S_2**+1≦20 范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. **S_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。

API	指令码			操作数	功能
2110		$FVAL	P	**S**，**D**	String→Float

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●						○	
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D									●				

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 转换的浮点数字符串来源

D ： 存放转换后的浮点数

指令说明：

1. 将 **S** 的浮点数字符串转换成浮点值，并将结果存于 **D**。

	上 8 位	下 8 位
S	第二个的 ASCII 码	第一个的 ASCII 码
S + 1	第四个的 ASCII 码	第三个的 ASCII 码
S + 2	第六个的 ASCII 码	第五个的 ASCII 码
……	……	
S +n-1	第 m 个的 ASCII 码	第 m-1 个的 ASCII 码
S +n		16#00 终止符

浮点数字符串

⇨

（D+1，D）

浮点数

- 各 ASCII 码输入限制请参考说明。
- 不论是小数格式或指数格式，浮点数字符串总长度（m）最多为 24 个字（不含终止符 16#00），因此 n 最多为 13。

2. **S** 的浮点数字符串分为小数格式跟指数格式两种，格式会由主机自动根据 **S** 的输入判别是哪一种。
   - 当使用小数格式时，范例如下：（不含终止符 16#00 的浮点数字符串长度为 9）

	上 8 位	下 8 位
S	16#31（1）	16#32（2）
S + 1	16#32（2）	16#2E（.）
S + 2	16#34（4）	16#33（3）
S + 3	16#36（6）	16#35（5）

⇨

D

21.234567

S + 4	16#00 终止符	16#37（7）

或

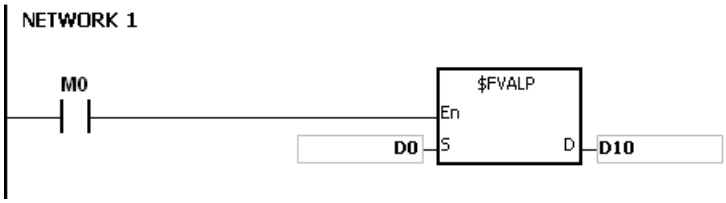
	上 8 位	下 8 位		
S	16#31（1）	16#2D（-）		
S + 1	16#32（2）	16#2E（.）		**D**
S + 2	16#34（4）	16#33（3）	⇨	-1.234568
S + 3	16#36（6）	16#35（5）		
S + 4	16#00 终止符	16#38（8）		

- 当使用指数格式时，范例如下：（不含终止符 16#00 的浮点数字符串长度为 9）

	上 8 位	下 8 位		
S	16#31（1）	16#2D（-）		
S + 1	16#32（2）	16#2E（.）		**D**
S + 2	16#45（E）	16#33（3）	⇨	-1230
S + 3	16#30（0）	16#2B（+）		
S + 4	16#00 终止符	16#32（3）		

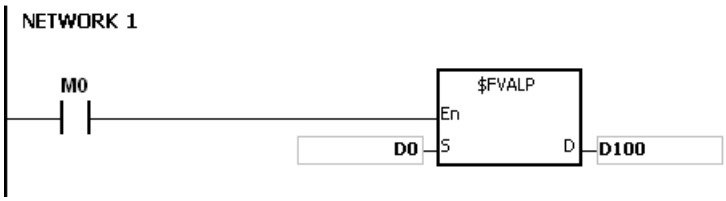
3. 如果 **S** 的第一个 ASCII 码被填入 16#20 或 16#2B，则转换结果为正数，填入 16#2D，则转换结果为负数。

程序范例一：



D0	16#31（1）	16#32（2）		
D1	16#32（2）	16#2E（.）		（D11，D10）
D2	16#34（4）	16#33（3）	⇨	21.234567
D3	16#36（6）	16#35（5）		
D4	16#00 终止符	16#37（7）		

程序范例二：



D0	16#31（1）	16#2D（-）
D1	16#32（2）	16#2E（.）
D2	16#45（E）	16#33（3）
D3	16#30（0）	16#2B（+）
D4	16#00 终止符	16#32（3）

⇨

（D101，D100）
-1230

补充说明：

1. **S** 字符串长度若超过 25 bytes 还没有 16#00 当结尾，SM0=ON，SR0 记录错误码 16#2003。
2. **S** 浮点数字符串输入限制如下说明，如果输入错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。

   第一个的 ASCII 码：可以输入符号、空白、数字。

   符号可输入 16#2B（+）、16#2D（-），空白可输入 16#20（空白），数字可输入 16#30（0）~16#39（9）。

   若第一个 ASCII 码是符号、空白则第二个 ASCII 码一定要视数字。

   第二个（含）之后 ASCII 码：可以输入小数格式、指数格式

   小数格式：

整数部份	或	整数部份	.	小数部份

   若输入小数点“.”（16#2E），则小数点前后都要有数字且小数点仅能出现一次。

   指数格式：

整数部份	指数部份	或	整数部份	.	小数部份	指数部份

   若输入小数点“.”（16#2E），则小数点前后都要有数字且小数点仅能出现一次。

   指数部份的前面一定要有数字。

   整数部分：仅可以是数字“0”（16#30）~“9”（16#39）。

   小数部分：仅可以是数字“0”（16#30）~“9”（16#39）。

   指数部分：四位数 ASCII 码，格式固定如下：

E或e	+或-	数字1	数字2

   一定要有一个“E”（16#45）或”e”（16#65），且仅能出现一次。

   一定要有符号“+”（16#2B）或 “-”（16#2D），且仅能出现一次。

   指一定要有两位数字“0”（16#30）~“9”（16#39），且固定就是要 2 位数。
3. **S** 字符串长度超出范围时，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
   - 字符串的第一个字填的是数字”0”~”9”（16#30~16#39），则浮点数字符串长度限制为 1~24，举例说明最小可输入的字符串长度：”1”
   - 字符串的第一个字填的是 16#20（空白）或符号（16#2B（+）、16#2D（-）），则浮点数字符串长度限制为 2~24，举例说明最小可输入的字符串长度："+1”
4. 当转换结果超出浮点数可以表示的范围时，指令不执行，SM0=ON，错误码 SR0=16#2013。

API	指令码			操作数	功能
2111		$RIGHT	P	**S**・**n**・**D**	从右边获取字符串

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●						○	
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

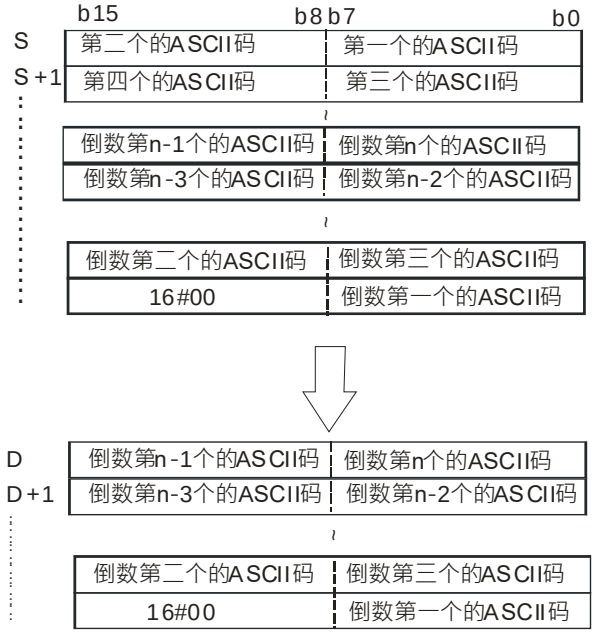
S ： 字符串来源

n ： 获取的长度

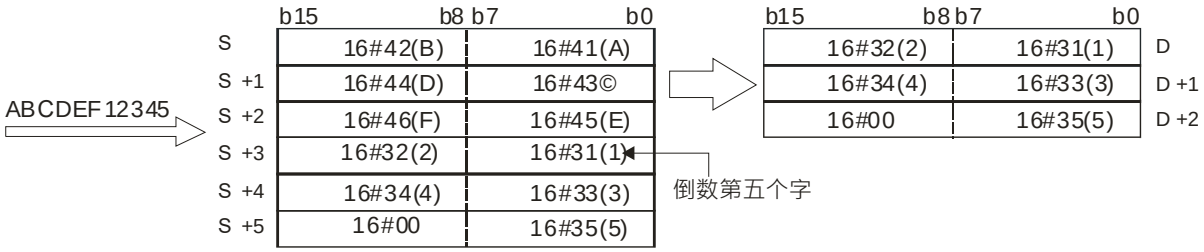
D ： 存放获取到的字符串

指令说明：

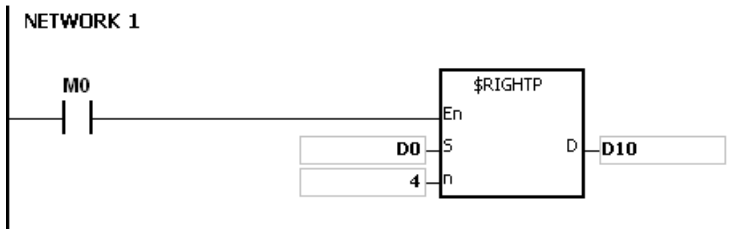
1. 从 **S** 字符串的最右边（最后一个字符）开始获取 **n** 个字，并将获取出来的字符串存放到 **D**。当 **S** 为字符串装置时，其最大长度为 31 个字符；选用其它非字符串装置时，则最大长度为 255 个字符。
2. 当 **n**=0 时，**D** 装置会存放 0。**n** 最大长度为 255。

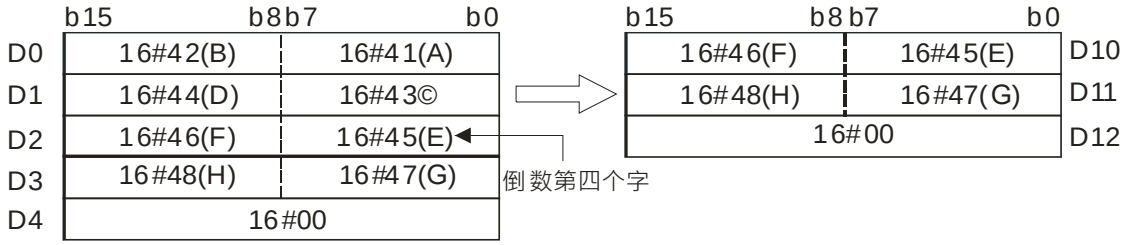


例如，**S** 的内容为 ABCDEF12345，当 **n**=5 时从最右边开始获取 5 个字符，转换结果如下图所示。



程序范例：





6

补充说明：

1. 若 S 操作数不是字符串（$）时，**S** 字符串最多读取 256 个字符（包含结尾字符 16#00），若没有结尾字符（16#00），指令不执行，SM0=ON，错误码 SR0=16#200E。
2. **当 n**<0 以 0 执行；当 **n**>**S** 的字符串长度时，以 **S** 字符串长度执行。
3. **D** 装置范围不足 **n** 个字符填入时，指令不执行，且 SM0=ON，SR0 记录错误码 16#2003。

API	指令码			操作数	功能
2112		$LEFT	P	**S**・**n**・**D**	从左边获取字符串

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●	●						○	
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

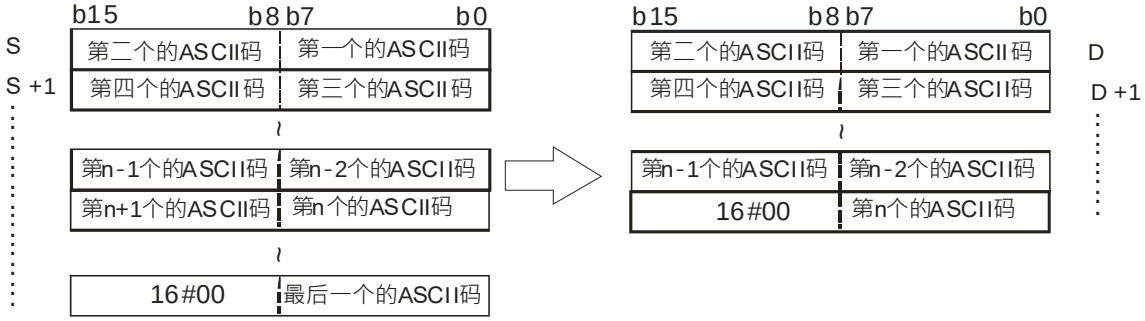
S ： 字符串来源

n ： 要获取的长度

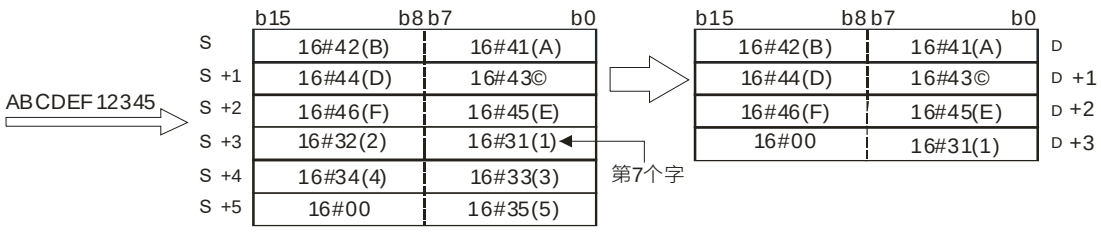
D ： 存放获取到的字符串

指令说明：

1. 从 **S** 字符串的最左边（第一个字符）开始获取 n 个字，并将获取出来的字符串存放到 **D**。。当 **S** 为字符串装置时，其最大长度为 31 个字符；选用其它非字符串装置时，则最大长度为 255 个字符。
2. 当 **n**=0 时，**D** 装置会存放 0。

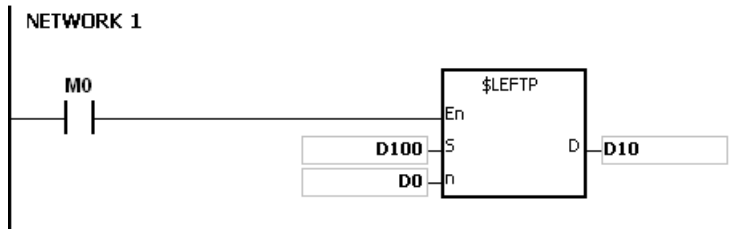


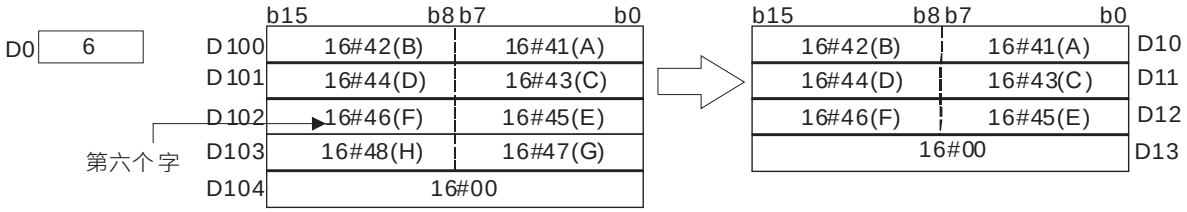
例如，S=ABCDEF12345，当 n=7 时从最左边开始获取 7 个字符，转换结果如下图所示。



程序范例：

M0=ON 时，执行$LEFT 指令，从 D100 开始的数据开始获取，取 6 个字符后将结果储存至 D10。





补充说明：

1. 若 S 操作数不是字符串（$）时，**S** 字符串最多读取 256 个字符（包含结尾字符 16#00），若没有结尾字符（16#00），指令不执行，SM0=ON，错误码 SR0=16#200E。
2. **当 n**<0 以 0 执行；当 **n**>**S** 的字符串长度时，以 **S** 字符串长度执行。
3. **D** 装置范围不足 **n** 个字符填入时，指令不执行，且 SM0=ON，SR0 记录错误码 16#2003。

API	指令码			操作数	功能
2113		$MIDR	P	**S_1 · S_2 · D**	区段获取字符串

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●		○					
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							●
S_2		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

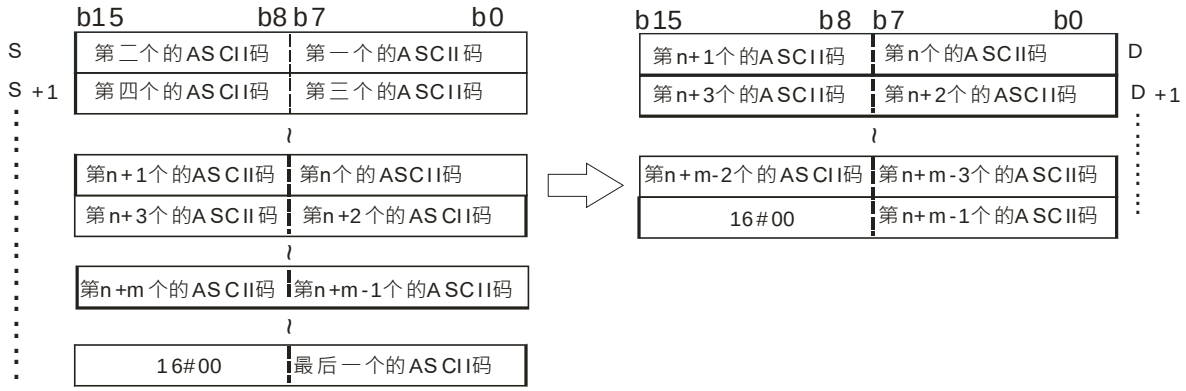
S_1 ： 字符串来源

S_2 ： 获取的区段设定值

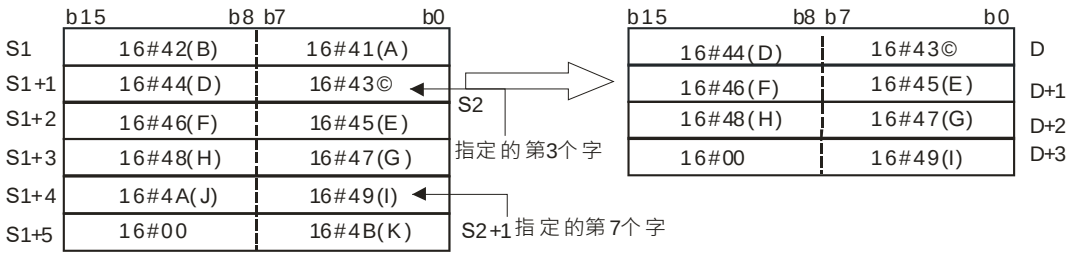
D ： 存放获取到的字符串

指令说明：

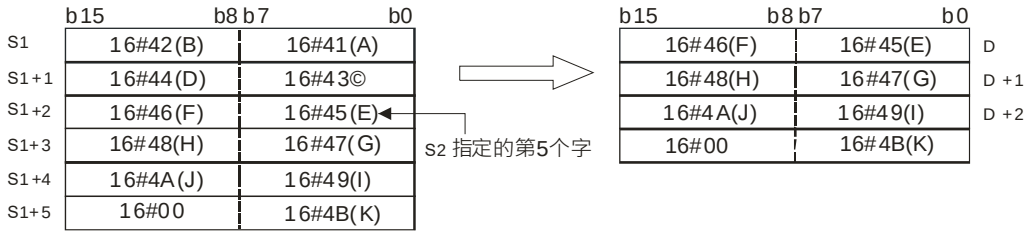
1. 将 **S_1** 字符串从 **S_2** 所指定的字开始获取 **S_2**+1 个字，并将获取出来的字符串存放到 **D**，当 **S_2**=n，**S_2**+1=m 如图说明。



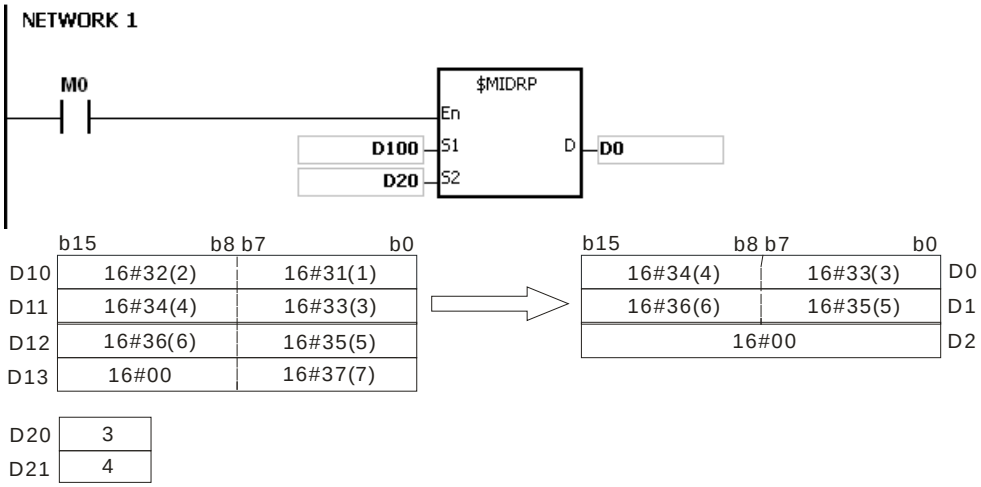
2. 例如 **S**=ABCDEFGHIJK，当 **S_2**=3 且 **S_2**+1=7 时，从最左边第 3 个字符开始获取 7 个字，转换结果如下图所示。



3. 当 **S_2**<=0 或 **S_2**+1<-1 或 **S_2**+1=0 时，指令不动作。
4. 当 **S_2**+1=-1 时，获取 **S_2** 开始到 **S_1** 结尾的所有字符串。
5. 例如 **S_1**=ABCDEFGHIJK，当 **S_2**=5 且 **S_2**+1=-1 时，如下图说明。



程序范例：



补充说明：

1. 若 **S_1** 操作数不是字符串（$）时，**$S_1$** 字符串最多读取 256 个字符（包含结尾字符 16#00），若没有结尾字符（16#00），指令不执行，SM0=ON，错误码 SR0=16#200E。
2. **S_2**<=0 或 **S_2**+1<-1 时，SM0=ON，SR0 记录错误码 16#2003。
3. **当 S_2 > S_1** 字符串长度或（**S_2** + **S_2**+1） > **S_1** 字符串长度，SM0=ON，SR0 记录错误码 16#2003。
4. **S_2**+1 所指定的长度超出 **S_1** 字符串的范围，SM0=ON，SR0 记录错误码 16#2003。
5. **D** 装置范围不足 **S_2**+1 个字符填入时，指令不执行，且 SM0=ON，SR0 记录错误码 16#2003。
6. 16 位指令 **S_2** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。

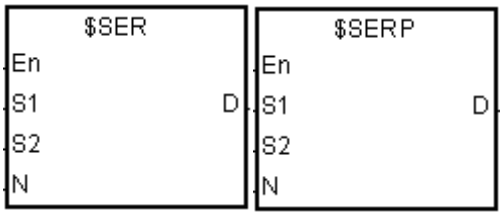
API	指令码			操作数	功能
2115		$SER	P	S_1 · S_2 · **n** · **D**	字符串搜寻

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●						○	
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							●
S_2		●				●							●
n		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 搜寻的目标字符串

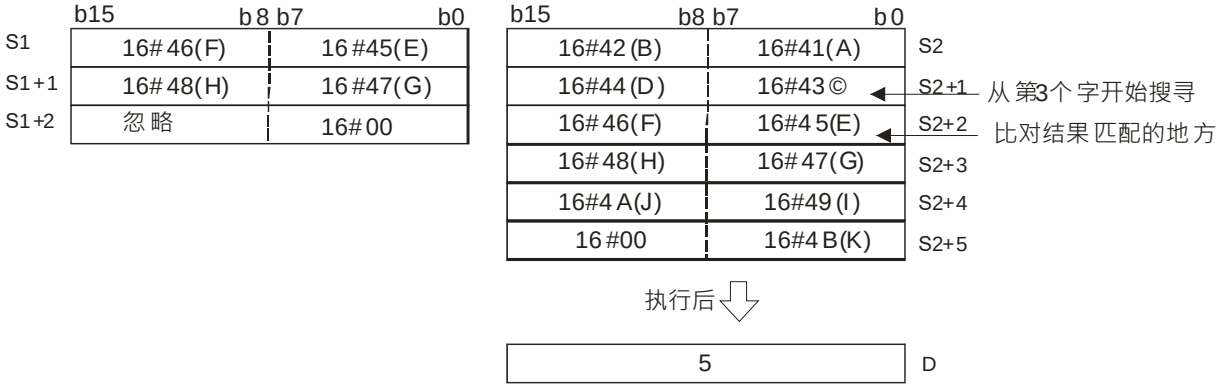
S_2 ： 被搜寻的字符串

n ： 从 **S_2** 的第 **n** 个字开始搜寻

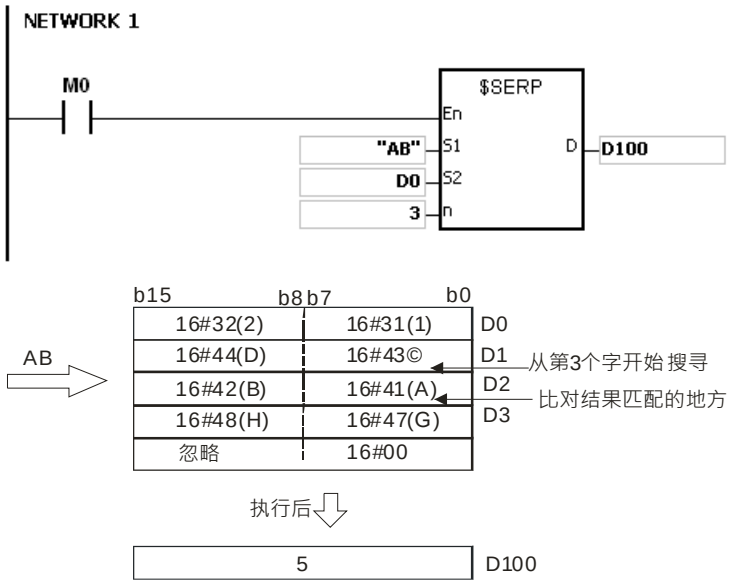
D ： 搜寻结果

指令说明：

1. 从 **S_2** 字符串的第 **n** 个字开始搜寻与 **S_1** 相同的字符串，如果搜寻到相同的字符串，则将此字符串的起始位置位于 **S_2** 字符串的第几个字，存于 **D** 中，如果没搜寻到相同的字符串，则 **D**=0。
2. 当第 **n** 个位置超出 **S_2** 字符串的长度时，或者 **S_1>S_2** 字符串的长度时，则 **D**=0。
3. 例如，在 **S_2**="ABCDEFGHIJK"中搜寻 **S_1**="EFGH"，从 **S_2** 的第 3 个字符（n=3）开始搜寻，搜寻结果为 **S_1** 存在 **S_2** 的起始位置为 5 并存在 **D**=5。



程序范例：



补充说明：

1. 若 **S_1** 操作数不是字符串（$）时，**$S_1$** 字符串最多读取 256 个字符（包含结尾字符 16#00），若没有结尾字符（16#00），指令不执行，SM0=ON，错误码 SR0=16#200E。
2. 若 **S_2** 操作数不是字符串（$）时，**$S_2$** 字符串最多读取 256 个字符（包含结尾字符 16#00），若没有结尾字符（16#00），指令不执行，SM0=ON，错误码 SR0=16#200E。
3. **n**<=0，SM0=ON，记录错误码 SR0=16#2003。

6

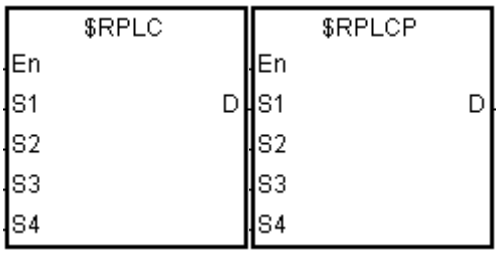
API	指令码			操作数	功能
2116		$RPLC	P	S_1 · S_2 · S_3 · S_4 · D	字符串取代

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●						○	
S_3	●	●			●	●		●	●		○	○	○	○		
S_4	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							●
S_2		●				●							●
S_3		●				●							
S_4		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 被取代的字符串

S_2 ： 新的取代字符串

S_3 ： **S_1** 要被取代掉的长度

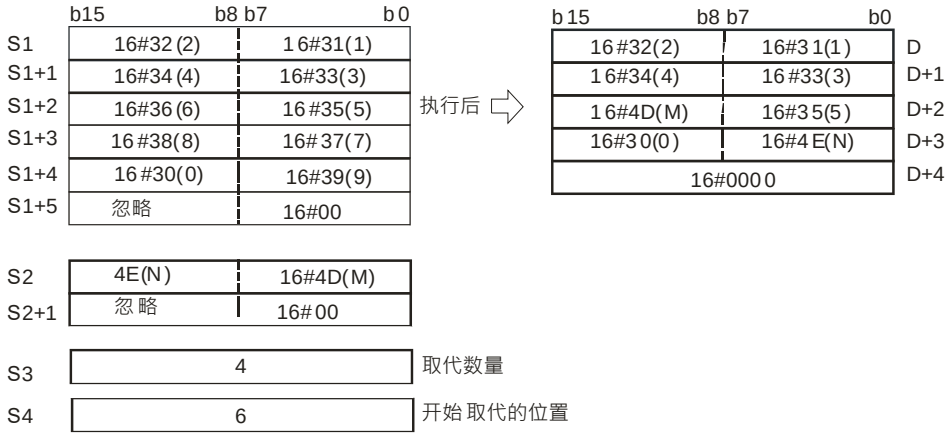
S_4 ： 从 **S_1** 的第 **S_4** 个字开始取代

D ： 取代后的字符串存放位置

指令说明：

1. 从 **S_1** 字符串的第 **S_4** 个字开始，用 **S_2** 的字符串取代 **S_1** 中的字符串，取代的长度为 **S_3**，并将结果存于 **D** 中，如下图范例说明。

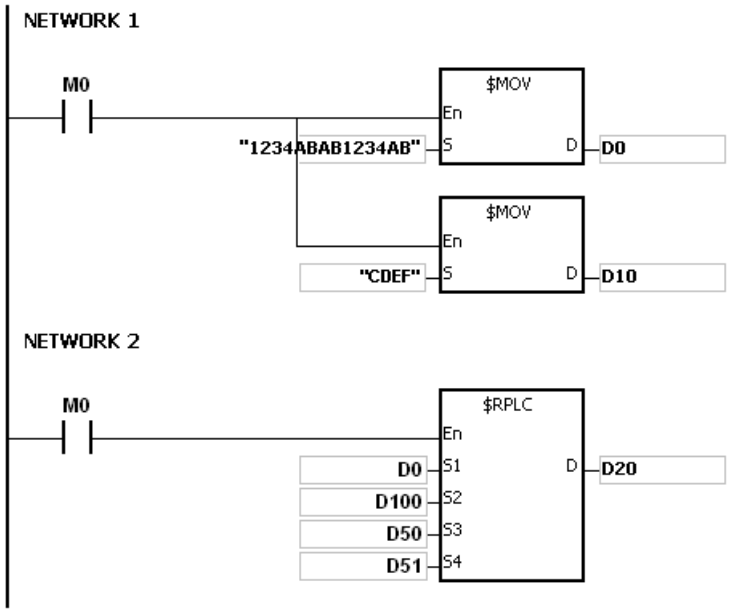
   将字符串"12345**6789**0"中的第 6 个字开始的 4 个字取代为"MN"，取代后的结果为"12345**MN**0"。



2. **S_2** 字符串如果为 16#00，则此指令功能相当于将字符串删除。
3. 若欲取代的字符串长度 **S_3>S_1** 可被取代的长度，则只取代到 **S_1** 的最后一个字。
4. **S_3**=0，指令不动作。

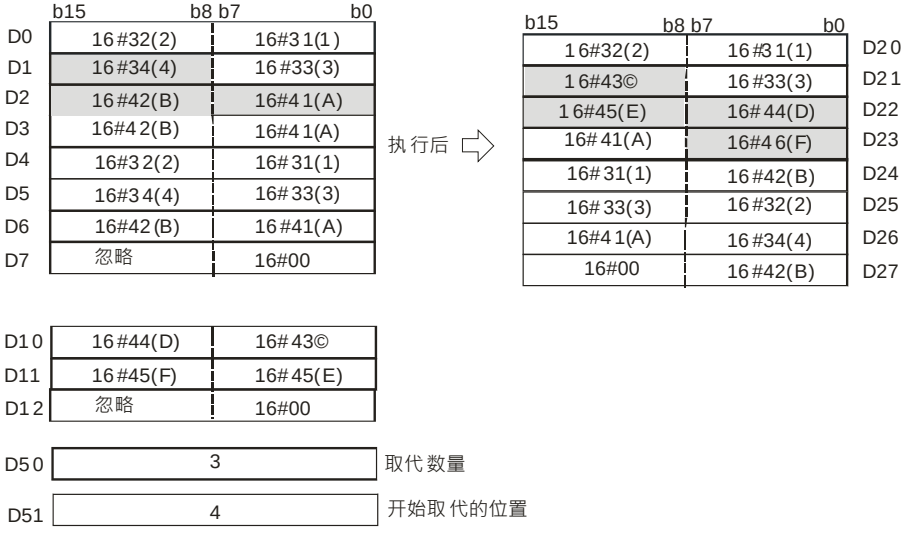
程序范例：

当 M0=ON 时，D0~D7 的内容为“1234ABAB1234AB”，D10~D11 的内容为“CDEF”；当$RPLC 指令执行，将用 D10~D11 的内容取代 D0~D7 中由 D51 指定的起始位置取代 D50 所指定的个数，结果储存于 D20~D27 当中。

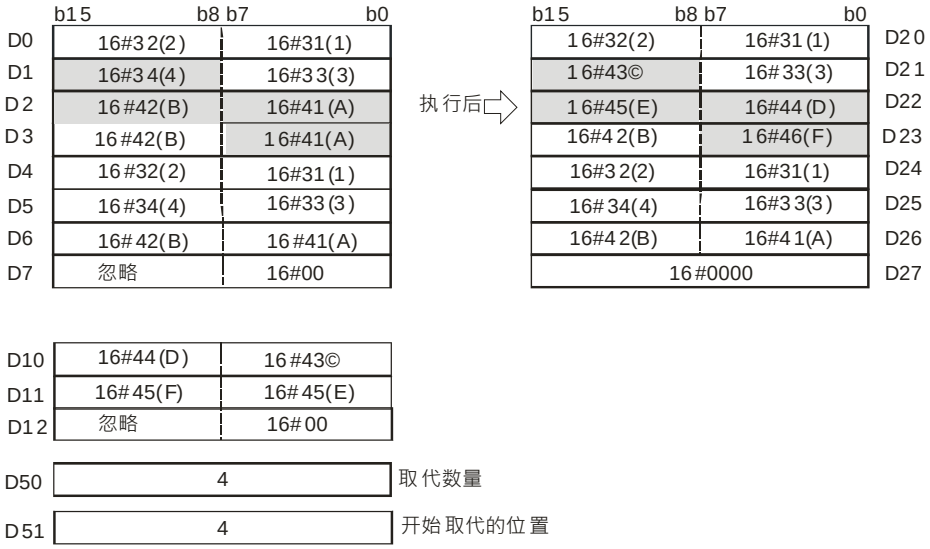


6

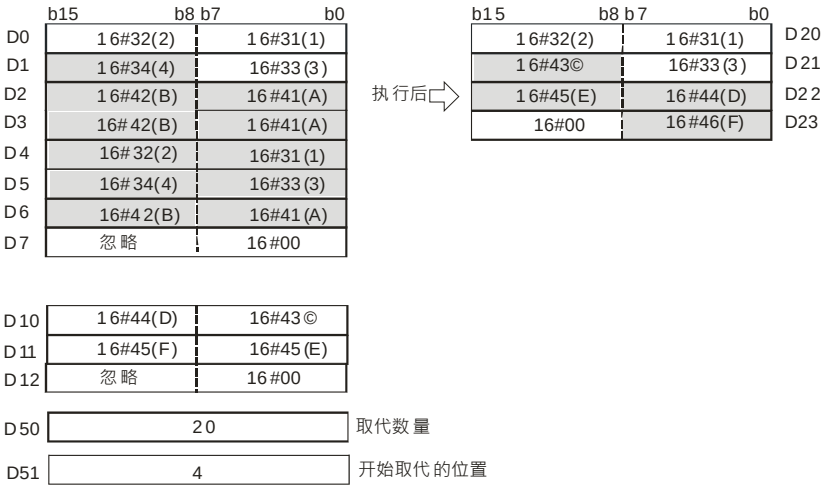
当 D50 指定值为 3，D51 指定值为 4 时，执行的结果如下图所示。



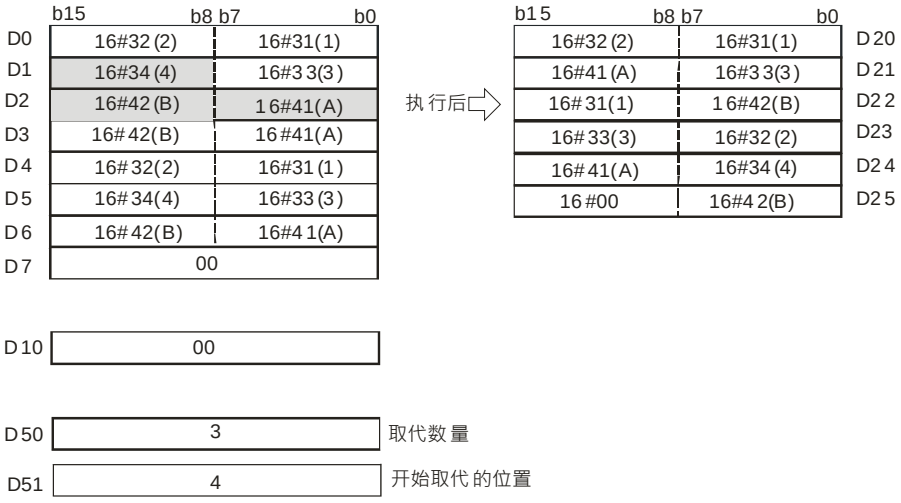
当 D50 指定值为 4，D51 指定值为 4 时，执行的结果如下图所示。



当 D50 指定值为 20，D51 指定值为 4 时，执行的结果如下图所示。



当 D50 指定值为 3，D51 指定值为 4，且 D10=16#00 表示未指定取代字符串时，执行的结果如下图所示。没有指定取代字符串，执行后原本的字符串被删除。



补充说明：

1. **S_1** 字符串中没有 16#00 当结尾，SM0=ON，记录错误码 SR0=16#200E。
2. **S_2** 字符串中没有 16#00 当结尾，SM0=ON，记录错误码 SR0=16#200E。
3. **S_3**<0 或 **S_4**<=0 或 **S_4** 指定的位置超出 **S_1** 的长度，SM0=ON，记录错误码 SR0=16#2003。
4. 若 **S_1** 字符串被取代后>256（含结束符号 16#00）时，指令不执行，SM0=ON，错误码 SR0=16#2003。

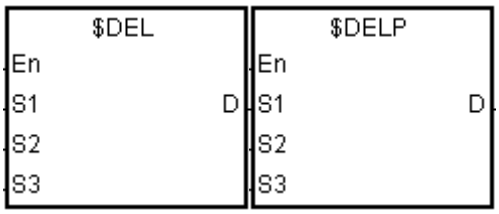
API	指令码			操作数	功能
2117		$DEL	P	S_1 · S_2 · S_3 · **D**	指定字符串删除

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●		○	○	○	○		
S_3	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							●
S_2		●				●							
S_3		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 字符串来源

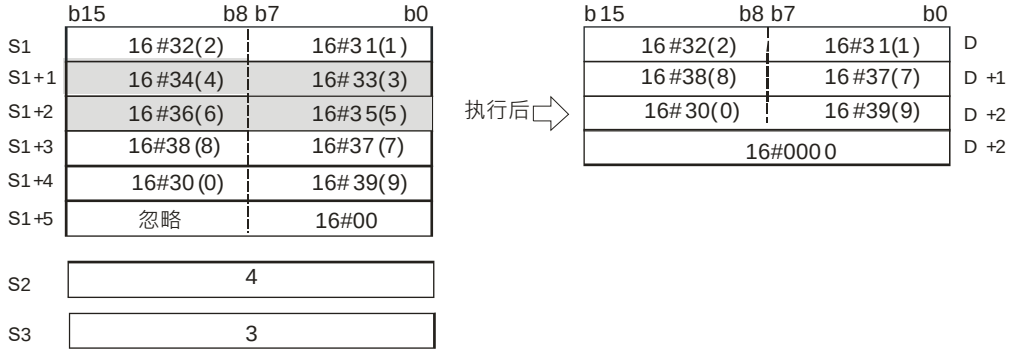
S_2 ： 要删除的个数

S_3 ： 从 **S_1** 的第 **S_3** 个字开始删除

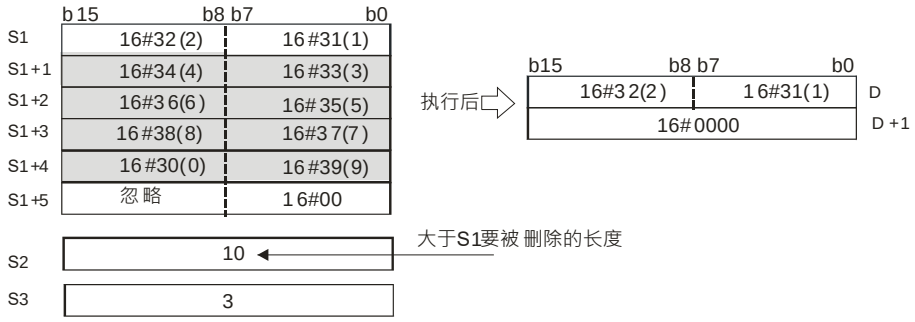
D ： 删除后的字符串

指令说明：

1. 从 **S_1** 字符串的第 **S_3** 个字开始，删除 **S_2** 个字符串，并将结果存于 **D** 中。
2. 如下图说明，将 **S_1** 的字符串"12**345**67890"，删除掉第 3 个开始共 4 个字符串，删除结果为"127890"并存在 **D** 中



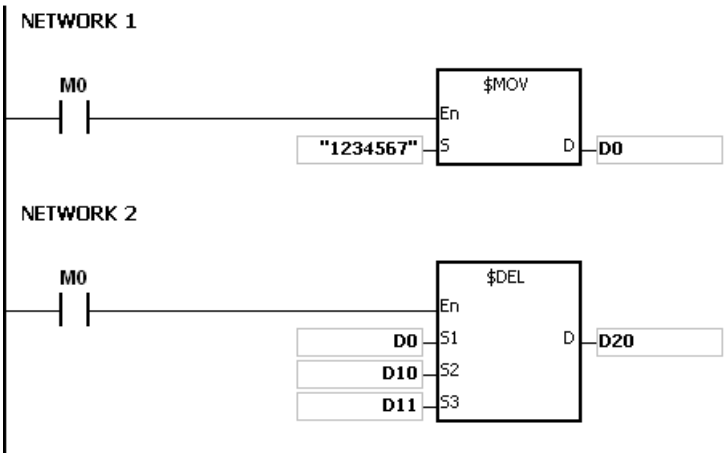
3. 如果欲删除的字符串长度 **S_2>S_1** 要被删除的字符串长度，则只删除到 **S_1** 的最后一个字，并将终止符 16#00 存入 **D** 中。



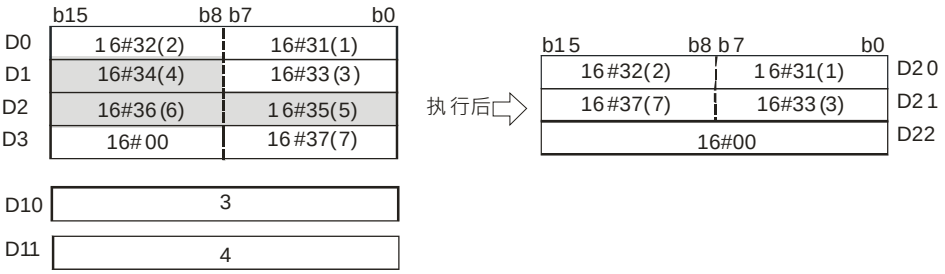
4. S_2=0，指令不动作。

程序范例：

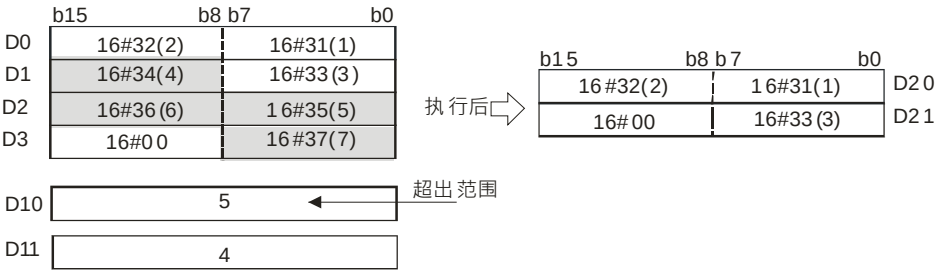
当 M0=ON 时，D0~D3 的内容为“1234567”；当$DEL 指令执行，将从 D11 内容的指定位置开始，删除 D0~D3 内的字符串，删除的字符数为 D10 内容所指定的数值，结果储存于 D20~D22 当中。



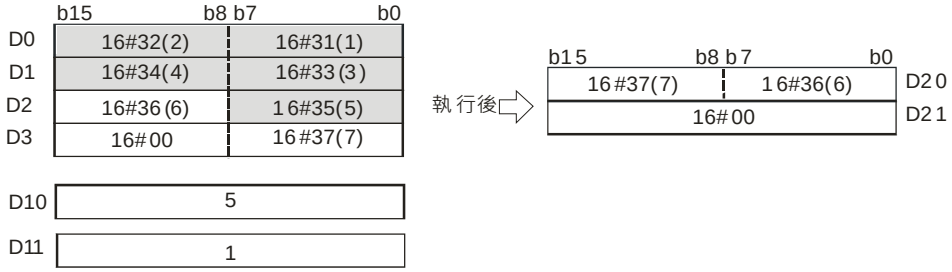
当 D10 指定删除数量为 3，D11 指定从第 4 个字符串开始删除时，执行的结果如下图所示。



当 D10 指定删除数量为 5，D11 指定从第 4 个字符串开始删除时，执行的结果如下图所示。由于指定的删除数量超出范围，故删除到最后一个字后结束。



当 D10 指定删除数量为 5，D11 指定从第 1 个字符串开始删除时，执行的结果如下图所示。



补充说明：

1. **S_1** 字符串中没有 16#00 当结尾，SM0=ON，记录错误码 SR0=16#200E。
2. **S_2**<0 或 **S_3**<=0 或 **S_3** 指定的位置超出 **S_1** 的长度，SM0=ON，记录错误码 SR0=16#2003。

API	指令码			操作数	功能
2118		$CLR	P	**S**	字符串清除

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 要清除的字符串来源

指令说明：

1. 将 **S** 中的字符串全部清除为 0。若无结尾字符 16#00，则最大清除 255 个字。

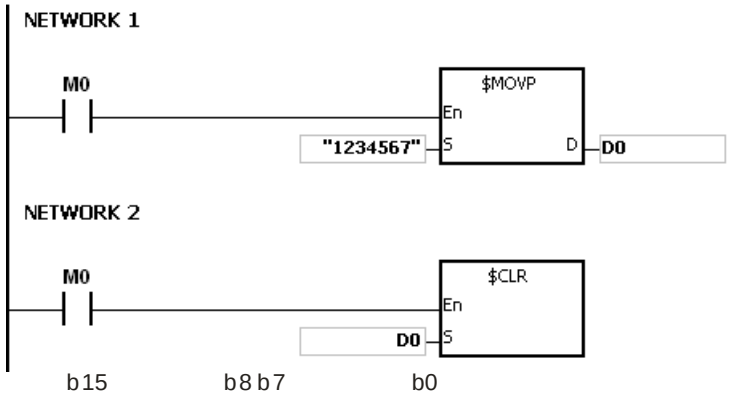
	b15 b8	b7 b0
S	16#31(1)	16#2D(-)
S +1	16#33(3)	16#32(2)
S +2	16#34(4)	16#35(5)
S +3	16#36(6)	16#2E(.)
S +4	16#38(8)	16#37(7)
S +5	16#00	

执行后

S	16#0000
S +1	16#0000
S +2	16#0000
S +3	16#0000
S +4	16#0000
S +5	16#0000

程序范例：

将 D0 的字符串内容清除，如下图说明。



	b15 b8	b7 b0
D0	16#32(2)	16#31(1)
D1	16#34(4)	16#33(3)
D2	16#36(6)	16#35(5)
D3	16#00	16#37(7)

执行后

16#0000	D0
16#0000	D1
16#0000	D2
16#0000	D3

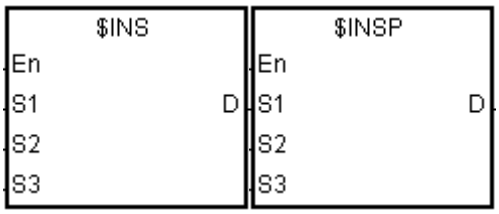
API	指令码			操作数	功能
2119		$INS	P	**S_1 · S_2 · S_3 · D**	字符串插入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●						○	
S_2	●	●			●	●		●	●						○	
S_3	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							●
S_2		●				●							●
S_3		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 来源字符串

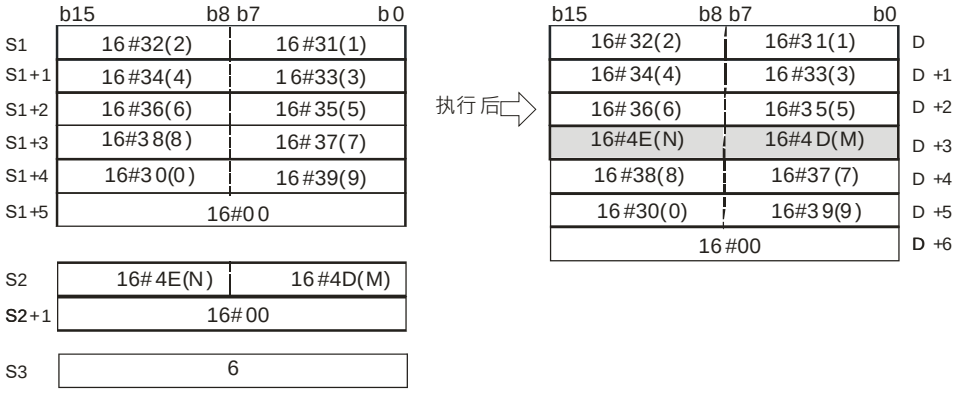
S_2 ： 要插入的字符串

S_3 ： 从 **S_1** 的第 **S_3** 个字之后开始插入

D ： 存放插入之后的字符串

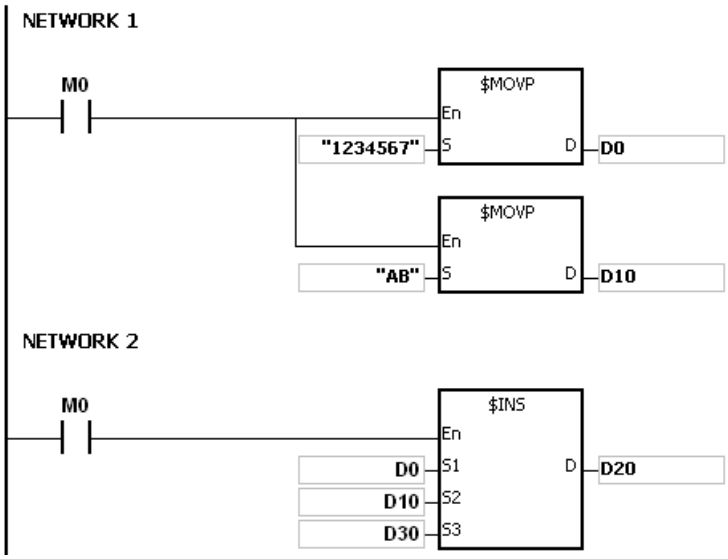
指令说明：

1. 从 **S_1** 字符串的第 **S_3** 个字之后开始，插入 **S_2** 的字符串，并将结果存于 **D** 中。
2. 如果 **S_1** 或 **S_2** 有任何一个字符串为空字符串，则另一个不是空字符串的值会被存于 **D** 中。
3. 如果 **S_1** 和 **S_2** 都为空字符串，则 **D** 为 16#0000。

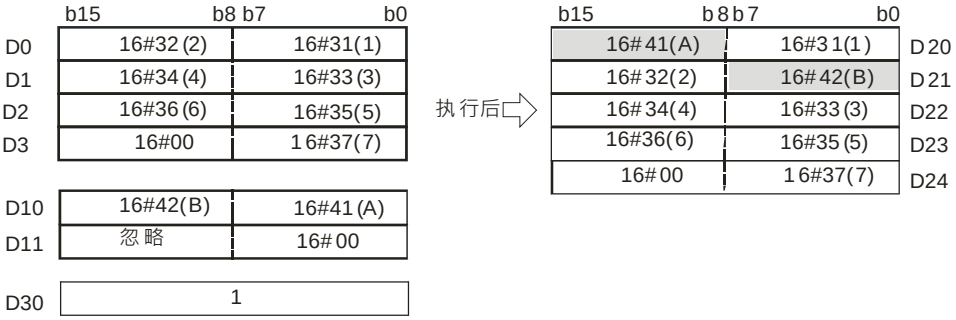


程序范例：

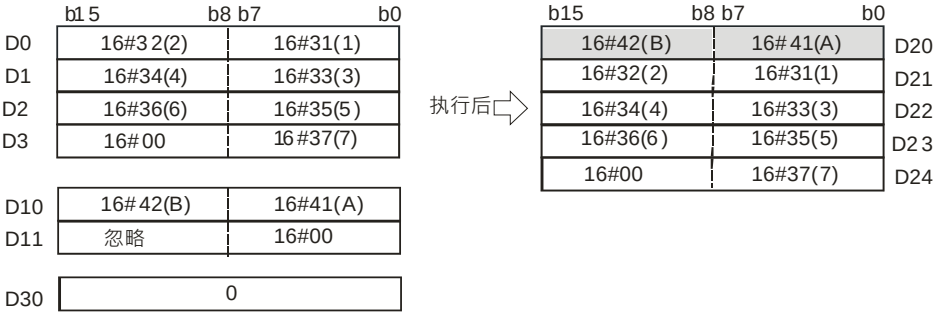
当 M0=ON 时，D0~D3 的内容为"1234567"，D10 的内容为"AB"；当$INS 指令执行，将从 D30 内容所指定的位置之后开始，插入 D10 的内容，结果储存于 D20~D24 当中。



D30 指定从第 1 个字符后开始插入，执行的结果如下图所示。



D30 指定从第 0 个字符后开始插入，执行的结果如下图所示。



补充说明：

1. **S_1** 字符串中没有 16#00 当结尾，SM0=ON，记录错误码 SR0=16#200E。
2. **S_2** 字符串中没有 16#00 当结尾，SM0=ON，记录错误码 SR0=16#200E。
3. **S_3**<0 或 **S_3** 指定的位置超出 **S_1** 的长度，SM0=ON，记录错误码 SR0=16#2003。
4. 若 **S_1** 字符串被插入后>256（含结束符号 16#00）时，指令不执行，SM0=ON，错误码 SR0=16#2003。

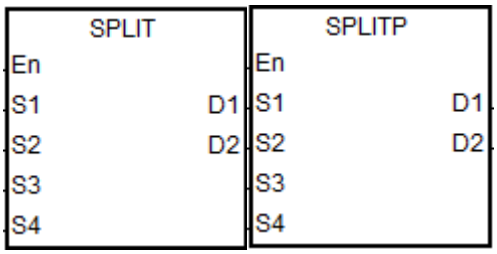
API	指令码			操作数	功能
2122		SPLIT	P	S_1 · S_2 · S_3 · S_4 · D_1 · D_2	字符串分离

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●							
S_2								●					○	○	○	
S_3								●					○	○	○	
S_4								●					○	○		
D_1								●								
D_2								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							●
S_3		●				●							●
S_4		●				●							
D_1		●				●							
D_2		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 字符串来源

S_2 ： 分隔字符

S_3 ： 分离终止符

S_4 ： 字符串分离后每组存放的单位长度

D_1 ： 存放分离后的字符串

D_2 ： 已分离字符串组数

指令说明：

1. 本指令仅支持 V1.04（含）以上的机种。
2. **S_1** 为字符串来源。其有效字符串仅到终止符为止，此终止符为字符串本身的终止符（16#00）或 **S_3** 为指定的分离终止符。
3. **S_2** 为指定的分隔字符，**S_3** 为指定的分离终止符。

   S_2 与 **S_3** 仅下 8 位有效，因此当输入错误时，PLC 不会报错。

   当 **S_2** 与 **S_3** 使用一般 D 装置时，请将字符 ASCII 值存放在下 8 位；若使用数值输入方式时，请直接输入字符 ASCII 数值；当遇到键盘无法输入字符时，建议直接使用数值输入方式。
4. **S_4** 为分离后每组字符串固定占用的单位长度（以 Word 为单位），此长度包含内定终止符（16#00 或 16#0000）。

每组可以存放的最多字数（Byte-含终止符）= **S_4** 单位长度 x 2。举例：**S_4**=2, 则最多可以存 3 个字+1 个终止符（16#00）。

S_4 数值设定范围为 1~100(超出此范围，自动以最小或最大值执行，因此当输入错误时，PLC 不会报错）。

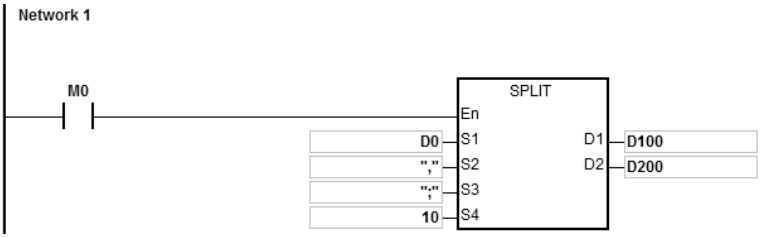
5. **D_2** 为已被分离字符串的组数累加数值，此数值将显示已分离字符串组数。
6. **D_1~ D_1+ S_4 *D_2** –1 以 **S_4** 为一组的单位长度，存放已分离的字符串。

   举例：当 **D_2** 数值为 3 时，则表示 **D_1**+0 开始存放第 1 组字符串，**D_1**+ **S_4** 开始存放第 2 组字符串，**D_1**+ **S_4** x2 开始存放第 3 组字符串。
7. 当指令寻找连续 **S_4** x2 个字符数，一直未发现分隔字符或终止符时，则自动分离为下一条字符串，并且已分离字符串数 **D_2** 累加 1。
8. **S_1** 来源字符串长度最大限制为 256 个字，以及 **D_2** 最多组数为 20 组，当以上两个限制条件任何一个已到达，而且没有终止符出现时，则指令将自动停止分离动作。

程序范例：

欲将存放在 D0~D11 的来源字符串“abcd,1234,5.67,8910;ef”以“,”做分隔并处理到“;”结束，且希望分割出的字符串依序填入以 10 个 Word 为单位的 D100 为起点的寄存器中。

注：上述要分割的字符串不含字符串“”。



6

说明：

1. 给定数据源 D0~D11=“abcd,1234,5.67,89103;edf”数据寄存器内容如下：

装置	内容值	字符串
D0	16#6261	“ba”
D1	16#6463	“dc”
D2	16#312C	“1,”
D3	16#3332	32
D4	16#2C34	“,4”
D5	16#2E35	“.5”
D6	16#3736	“76”
D7	16#382C	“8,”
D8	16#3139	“19”
D9	16#3330	“30”
D10	16#453B	“e;”
D11	16#4644	“fd”

2. 启动条件接点 M0
3. 分割的字符串组数 D200=4
4. 分割后的字符串内容如下：

第 1 组：占用 D100~D109，其中 D100~D101="abcd"

装置	D100	D101	D102	D103	D104	D105	D106	D107	D108	D109
HEX	6261	6463	0000	0000	0000	0000	0000	0000	0000	0000
字符串	"ba"	"dc"	16#0000	-	-	-	-	-	-	-

第 2 组：占用 D110~D119，其中 D110~D111="1234"

装置	D110	D111	D112	D113	D114	D115	D116	D117	D118	D119
HEX	3231	3433	0000	0000	0000	0000	0000	0000	0000	0000
字符串	"21"	"43"	16#0000	-	-	-	-	-	-	-

第 3 组：占用 D120~D129，其中 D120~D121="5.67"

装置	D120	D121	D122	D123	D124	D125	D126	D127	D128	D129
HEX	2E35	3736	0000	0000	0000	0000	0000	0000	0000	0000
字符串	".5"	"76"	16#0000	-	-	-	-	-	-	-

第 4 组：占用 D130~D139，其中 D130~D131="89103"

装置	D130	D131	D132	D133	D134	D135	D136	D137	D138	D139
HEX	3938	3031	0033	0000	0000	0000	0000	0000	0000	0000
字符串	"98"	"01"	下："3" 上：16#00	-	-	-	-	-	-	-

补充说明：

1. **S_1** 字符串最多读取 256 个字符，若没有终止符(16#00)或指令的分离终止符，都以 256 个字符来执行，错误标志 SM0 不会 ON。
2. **D_1** 分离后的字符串最后都将自动补上字符串内定的终止符 16#00 或 16#0000 (请看范例说明)。
3. **D_1** 如果分离后是空字符串也会补上字符串内定的终止符 16#00 或 16#0000。

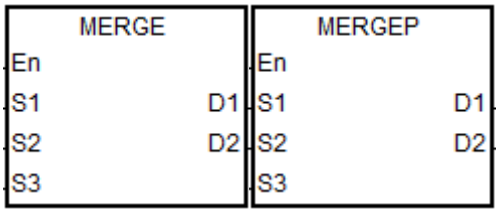
API	指令码			操作数	功能
2123		MERGE	P	**S_1 · S_2 · S_3 · D_1 · D_2**	字符串合并

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●	●						○	
S_2								●	●						○	
S_3								●					○	○	○	
D_1								●								
D_2								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							●
S_2		●				●							●
S_3		●				●							●
D_1		●				●							
D_2		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 来源字符串 1

S_2 ： 来源字符串 2

S_3 ： 分隔字符

D_1 ： 存放合并后的字符串

D_2 ： 合并后字符串的字符数量

指令说明：

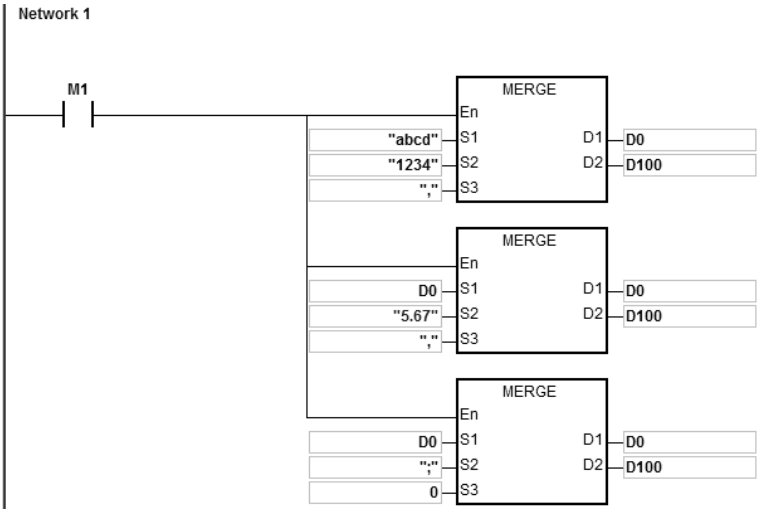
1. 本指令仅支持 V1.04（含）以上的机种。
2. **S_1** 与 **S_2** 分别为来源字符串 1 与 2，**S_3** 为分隔字符且仅下 8 位有效，当 **S_3** 分隔字符设定为 0 数值(16#00) 时，则 **S_1** 与 **S_2** 字符串将直接合并；当 **S_3** 分隔字符设定非 0 数值时，则 **S_1** 字符串会加上分隔字符后，再合并 **S_2** 字符串。
3. **D_1** 为存放合并后之字符串，其合并顺序固定为 ”字符串 1” + 分隔字符 + “字符串 2”，并且固定终止符为 0（16#00），若需要改用别的终止符时，请自行再合并一个终止符即可。
4. **S_1**、**S_2** 与 **S_3** 都为空字符串，则 **D_1** 为 16#0000。
5. **D_2** 为合并后之字符串的字符数量（不含终止符 16#00），其累计的字符串长度为 ”字符串 1 长度” + “0 或 1 个分隔字” + “字符串 2 长度”。

6. **S_1** 与 **S_2** 来源字符串的终止符都为 0，长度最多为 255 个字（不含终止符 16#00）；当字符串合并后，其存放的最大长度为 255 个字（不含终止符 16#00）；举例：字符串 1 为 250 个字，分隔字 1 个字，字符串 2 为 100 个字，则合并后字符串将会是 "字符串 1 的 250 个字" + 分隔字 1 个 + "字符串 2 的最前面 4 个字"。

程序范例：

假设共有 3 个来源字符串分别为"abcd"、"1234"、"5.67"，另外分隔与终止符分别为 "," 与 ";"

则合并程序写法如下：



最后存放于 D0~D7 的结果为 "abcd,1234,5.67;" D100=15

说明：

1. 启动条件接点 M1
2. 第一个 MERGE 指令：会将数据源 1 的字符串 "abcd" 及数据源 2 的字符串"1234，以","为分隔字，合并后存放在 D0 起始的寄存器中。

   合并后 D0~D4= "abcd,1234"，并纪录合并字数为 D100=9，寄存器内容如下：

装置	D0	D1	D2	D3	D4
HEX	6261	6463	312C	3332	0034
字符串	"ba"	"dc"	"1,"	"32"	上 8 位:16#00、下 8 位:"4"

3. 第二个 MERGE 指令：会将数据源 1 的字符串 D0="abcd,1234" 及数据源 2 的字符串"5.67"，以","为分隔字，合并后存放在 D0 起始的寄存器中。

   合并后 D0~D7= "abcd,1234,5.67"，并纪录合并字数为 D100=14，寄存器内容如下：

装置	D0	D1	D2	D3	D4	D5	D6	D7
HEX	6261	6463	312C	3332	2C34	2E35	3736	0000
字符串	"ba"	"dc"	"1,"	"32"	",4"	".5"	"76"	16#0000

4. 第三个 MERGE 指令：会将数据源 1 的字符串 D0="abcd,1234,5.67" 及数据源 2 的字符串";"，以 0 数值做分隔（表数据源 1 与数据源 2 将直接合并），合并后存放在 D0 起始的寄存器中。

   合并后 D0~D7= "abcd,1234,5.67;"，并纪录合并字数为 D100=15，寄存器内容如下：

装置	D0	D1	D2	D3	D4	D5	D6	D7
HEX	6261	6463	312C	3332	2C34	2E35	3736	003B
字符串	“ba”	“dc”	“1,”	“32”	“,4”	“.5”	“76”	上 8 位：16#00、下 8 位：“;”

补充说明：

1. **S_1** 字符串中没有 16#00 当结尾，SM0=ON，记录错误码 SR0=16#200E。
2. **S_2** 字符串中没有 16#00 当结尾，SM0=ON，记录错误码 SR0=16#200E。
3. 若 **D_1**+ （**D_2**+1）/2 - 1 超过装置范围时，指令不执行，SM0=ON，错误码 SR0=16#2003。
4. 合并后的字符串最后都将自动补上字符串内定的终止符 16#00 或 16#0000（请看范例说明）

6.22 以太网控制指令

6.22.1 以太网控制指令一览表

API	指令码（位）		P 指令	功能
	16	32		
2200	SOPEN	–	✓	开启 Socket
2201	SSEND	–	✓	透过已开启的 Socket 传送数据
2203	SCLOSE	–	✓	关闭 Socket
2204	MSEND	–	✓	寄送电子邮件
2206	INTOA	–	✓	IP→字符串变换
2207	IATON	–	✓	字符串→IP 变换
2208	EIPRW	–	–	EtherNet/IP 数据读写
2209	SCONF	–	✓	Socket 参数设定
2210	MCONF	–	✓	MODBUS TCP 数据交换设定
2211	EMCONF1	–	✓	E-mail server 参数设定
2212	EMCONF2	–	✓	E-mail 地址设定
2214	MQTT_Connect	--	--	MQTT Client 联机指令
2215	MQTT_Publish	--	--	MQTT Client 发布指令
2216	MQTT_Subscribe	--	--	MQTT Client 订阅指令

6.22.2 以太网控制指令说明

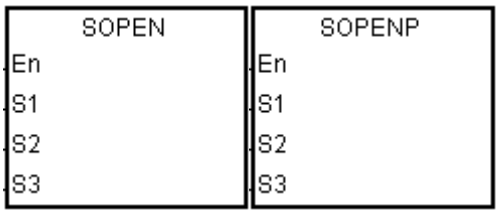
API	指令码		操作数	功能
2200	SOPEN	P	S_1 · S_2 · S_3	开启 Socket

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●			○	○	○		
S_2	●	●			●	●		●	●			○	○	○		
S_3	●	●			●	●		●	●			○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： Socket 模式.

S_2 ： Socket 编号

S_3 ： 设定通讯模式

指令说明：

1. **S_1** 为 Socket 模式，当 **S_1** 为 1 表示开启 TCP Socket，当 **S_1** 为 0 表示开启 UDP Socket。
2. **S_2** 为 Socket 编号，需对应以太网络进阶设定中的 Socket 设定，最多可设定 4 个。
3. 当开启 TCP 联机（**S_1**=1）并且 **S_3** 为 1 或 3，为客户端（Client）模式，由 PLC 主动送出联机要求给服务端（Server）。当开启 TCP 联机（**S_1**=1）并且 **S_3** 为 0 或 2，表示服务端（Server）模式，由 PLC 等待来自客户端（Client）的联机要求。V1.06.30 版（含）以上支持 **S_3** 为 2 或 3。
4. 当开启 UDP 联机（**S_1**=0）时，无客户端与服务端区别，**S_3** 可设为 0~3。
5. **S_1**=0~1(默认值 0)·**S_2**=1~4(默认值 1)·**S_3**=0~3(默认值 0);当超出范围时 ·PLC 会显示错误码(H'2003) 表示装置超过可用范围。
6. 使用本指令前，需用 ISPSoft→HWCONFIG 完成以下设定。
   - 在 PLC 参数设定→ 以太网络基本设定→设定本机的 IP 地址、网络屏蔽。
   - 在 PLC 参数设定进阶设定→通讯接口→启动通讯接口功能。
   - 在 PLC 参数设定→ 以太网络进阶设定→通讯接口→设定欲使用的 TCP/UDP Socket 编号
7. 若程序当中需修改 TCP 与 UDP Socket 内之参数值，请参考 API2209 SCONF 指令说明。

8. V1.06.30 版（含）以上支持 **S**$_3$ 为 2 或 3，此时 PLC 接收地址第一个位置将会保存接收长度，第二个位置开始存放接收数据。欲接收下一笔数据时，须将第一个位置的长度值清除，若无清除，则将于接收到新的数据时，依照顺序往后放，直到将第一个位置的值被写为 0 或超出最大接收数据长度。

9. 开启 TCP Socket（**S**$_1$=1）时，开启模式（**S**$_3$）、远程 IP 和通讯端口编号设定值之说明，如下表所示。

开启模式	**S**$_3$	远程 IP	远程通讯端口	本机通讯端口	说明
主站模式（Client）	1 或 3	特定 IP 地址	不等于 0	不等于 0	指定 IP 地址、本机通讯端口和远程通讯端口
	1 或 3	特定 IP 地址	不等于 0	0	指定 IP 地址但不指定本机通讯端口
	1 或 3	任意值	0	任意值	不合法，本机作为主站时，必须指定远程（Server）通讯端口与 IP 地址
	1 或 3	0.0.0.0	任意值	任意值	
从站模式（Server）	0 或 2	特定 IP 地址	0	不等于 0	指定 IP 但不指定远程通讯端口
	0 或 2	特定 IP 地址	不等于 0	不等于 0	指定 IP 地址和远程通讯端口
	0 或 2	0.0.0.0	0	不等于 0	不指定 IP 地址和远程通讯端口
	0 或 2	0.0.0.0	不等于 0	不等于 0	不指定 IP 地址但指定远程通讯端口
	0 或 2	任意值	任意值	0	不合法，本机作为从站时，必须指定本机（Server）通讯端口

注：新增模式 2 与 3，需搭配使用韧体 V1.06.30 版（含）以上。其差异点为：原先模式 0，1 只接收通讯内容，新增模式 2，3 会将接收的第一个位置显示实际接收的总长度，以供判断该次接收通讯数据是否已经接收完成。

10. 开启 TCP Socket 时，当此指令执行后，若无错误，则开始与远程装置进行联机，此时联机建立标志为 ON，若联机成功则已联机标志为 ON，联机建立中标志为 OFF，反之若有错误则对应错误标志 ON。

TCP Socket 编号	联机建立完成标志	数据接收完成标志	数据传送完成标志	联机已关闭标志	错误标志
1	SM1270	SM1271	SM1272	SM1274	SM1277
2	SM1278	SM1279	SM1280	SM1282	SM1285
3	SM1286	SM1287	SM1288	SM1290	SM1293
4	SM1294	SM1295	SM1296	SM1298	SM1301

11. 开启 UDP Socket 时，当此指令执行后，若无错误，则联机已开启之标志为 ON，反之若有错误则对应错误标志 ON。

UDP Socket 编号	联机建立完成标志	数据接收中标志	数据传送中标志	联机关闭执行标志	错误标志
1	SM1334	SM1335	SM1336	SM1337	SM1338
2	SM1339	SM1340	SM1341	SM1342	SM1343
3	SM1344	SM1345	SM1346	SM1347	SM1348
4	SM1349	SM1350	SM1351	SM1352	SM1353

12. AS 已使用通讯端口定义如下，当 TCP Socket 联机为服务端（Server）模式或使用 UDP Socket，本机通讯端口不允许设定下表所列之通讯端口号码。

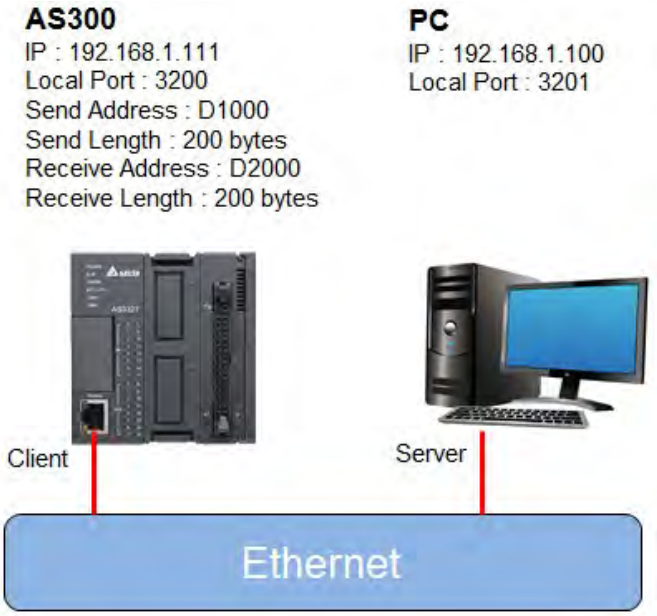
Socket 联机	通讯端口	已使用之通讯协议名称
TCP	80	HTTP
TCP	502	MODBUS TCP
TCP	44818	EtherNet/IP
UDP	68	DHCP/BOOTP
UDP	2222	EtherNet/IP
UDP	44818	EtherNet/IP
UDP	20006	AS 自定义通讯

13. **本指令一般都是使用脉冲执行型指令（SOPENP）。**

程序范例一：(S_3 为模式 1)

1. 此程序范例的系统架构如下图，此范例说明如何以 PC 为服务器端（Server），AS 为客户端（Client）建立一个 TCP 联机。

（备注：PC 需通过设定或搭配工具去开启 Socket 功能，另须注意防火墙或防病毒软件是否会限制通讯端口使用）



2. ISPSoft -> HWCONFIG（以太网络-基本设定 ）

网络设定	
IP 寻址模式	静态
IP 地址	192.168.1.111
网络屏蔽	255.255.255.0
网关地址	192.168.1.1

3. a）ISPSoft -> HWCONFIG（以太网络-进阶设定 -> 通讯接口 -> 1st TCP ）

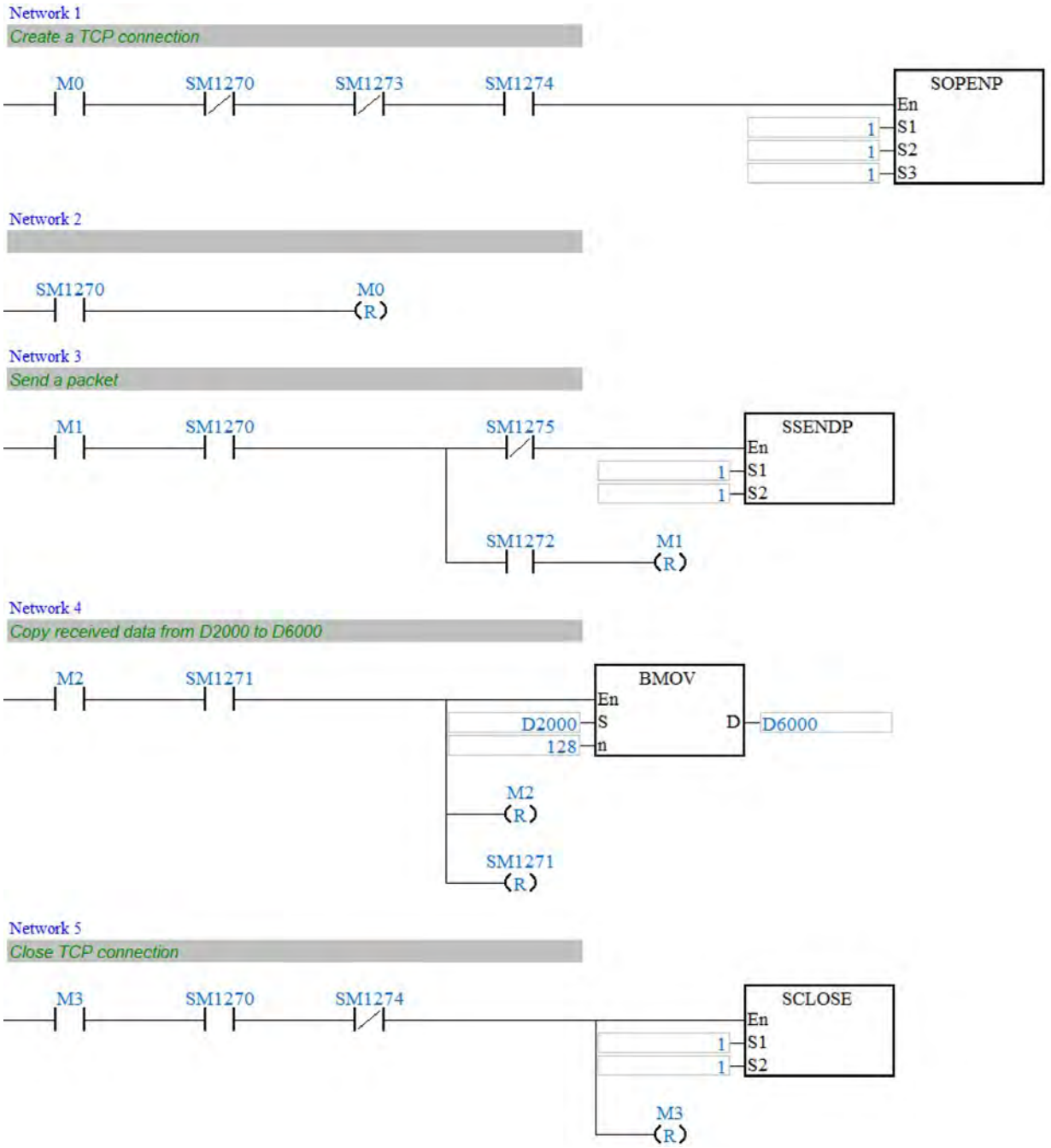
   b）SCONF 指令

1st TCP 设定	
远程 IP 地址	192.168.1.110
远程通讯端口	3201
本机通讯端口	3200
传送数据地址	D1000
传送数据指定长度#1	200
接收数据地址	D2000
接收数据最大长度#1	200
联机保持时间（秒）	10

注#1：此长度数值范围于 V1.06.20 版以下，为 512 bytes。于 V1.06.30 版以上，为 4096 bytes，需使用 SCONF 指令设定。

4. 先行在 ISPSoft→HWCONFIG 参数设定中设定 TCP Socket1 的信息，或使用 SCONF 指令设定。
5. 建立联机（Network 1）：当 M0=ON 时，检查欲联机的 Socket 是否处于关闭中（SM1274=ON）、未联机（SM1270=OFF）以及未建立联机中（SM1273=OFF）的状况，如果条件成立则进入联机程序，联机建立完成后（SM1270=ON），清除 M0（Network 2）。
6. 传送数据(Network 3):当 M1=ON 时 ,开始传送数据 ,检查欲传送的 Socket 是否为已联机(SM1270=ON),并且无数据在传送中（SM1275=OFF）的状态。如果条件成立，则开始传送数据；若否，则此指令无法运行，传送完成后（SM1272=ON），清除 M1。
7. 数据处理（Network 4）：当 M2=ON 时，检查接收中标志（SM1271）是否为 ON？若是，则表示已接收到数据，但是此时并不一定表示已经到达最大接收长度，用户需自行判断接收数据内容。数据处理完毕后清除 M2 及 SM1271。
8. 关闭联机（Network 5）：当 M3=ON 时，准备关闭联机，检查欲关闭的 Socket 是否为已联机状态（SM1270=ON）且不处于联机关闭中（SM1274=OFF）的状态，若条件成立则开始关闭联机，若尚未联机则此指令无法运行。联机关闭后，清除 M3。

范例程序如下所示：

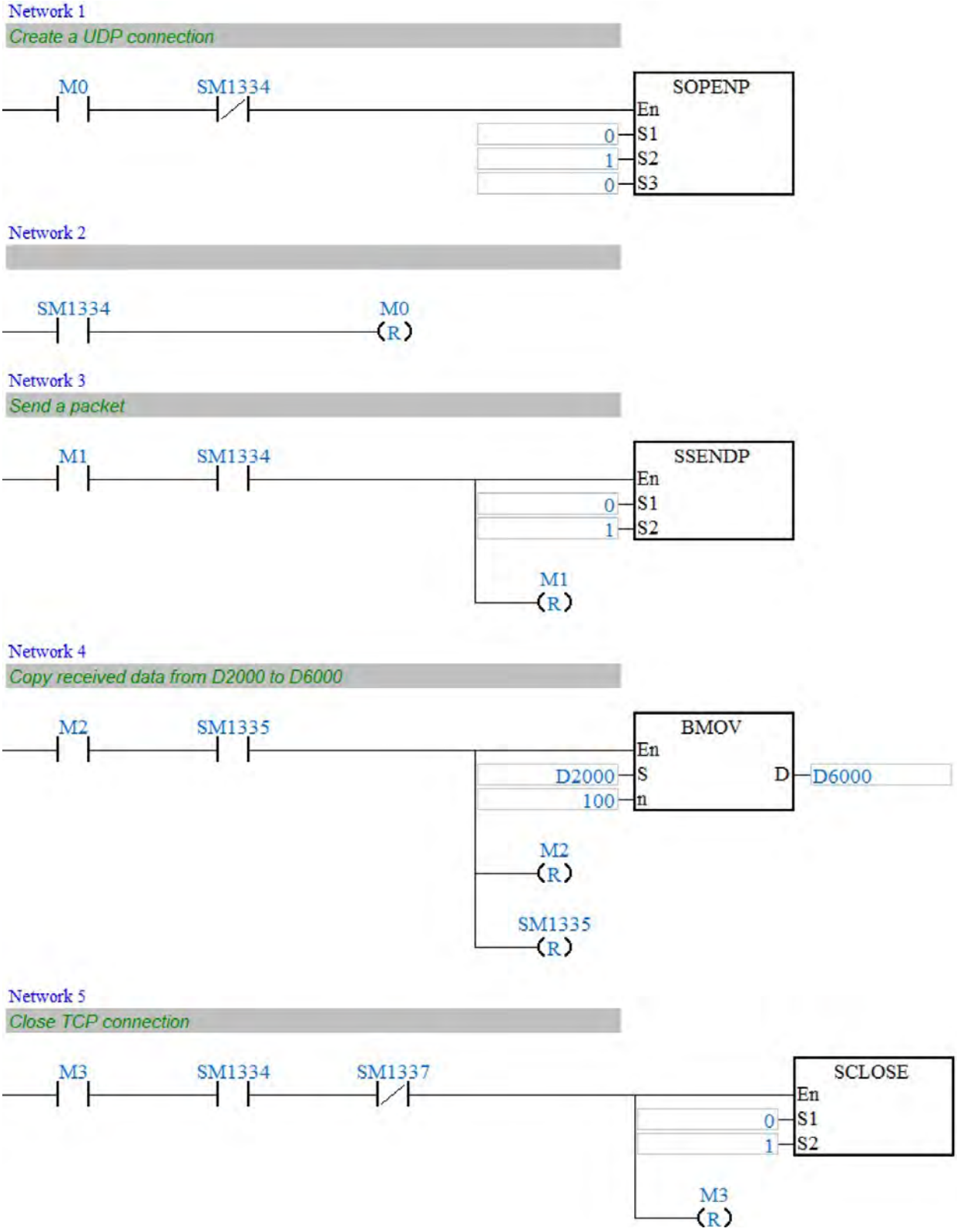


程序范例二：(S_3 模式为 0)

1. 此范例说明如何建立一个 UDP 联机，参考程序范例一步骤 3，将相同内容设定到 ISPSoft -> HWCONFIG（以太网络-进阶设定 -> 通讯接口 -> 1st UDP）。
2. 建立联机（Network 1）：当 M0=ON 时，检查欲联机的 Socket 是否处于关闭中（SM1334=OFF），如果条件成立则进入联机程序，联机建立完成后（SM1334=ON），清除 M0（Network 2）。
3. 传送数据（Network 3）：当 M1=ON 且联机已建立（SM1334=ON）时，开始传送数据，传送后，清除 M1。
4. 数据处理（Network 4）：当 M2=ON 时，检查接收中标志（SM1335）是否为 ON？若是，则表示已接收到数据，但是此时并不一定表示已经到达最大接收长度，用户需自行判断接收数据内容。数据处理完毕后清除 M2 及 SM1335。

5. 关闭联机（Network 5）：当 M3=ON 时，准备关闭联机，检查欲关闭的 Socket 是否为已联机状态（SM1334=ON）且不处于联机关闭中（SM1337=OFF）的状态，若条件成立则开始关闭联机，若尚未联机则此指令无法运行。联机关闭后，清除 M3。

范例程序如下所示：



程序范例三：(S_3 模式为 2)

1. 此范例说明如何建立一个从站（Server）透过 TCP 联机接收通讯数据，使用模式 2 并且接收 2K Byte 的数据。
2. 设定 SCONF

1st TCP 设定	
远程 IP 地址	192.168.1.106
远程通讯端口	任意（0）
本机通讯端口	2020
传送数据地址	D10000
传送数据指定长度#1	4096
接收数据地址	D15000
接收数据最大长度#1	4096
联机保持时间（秒）	30

注#1：此长度数值范围于 V1.06.20 版以下，为 512 bytes。于 V1.06.30 版以上，为 4096 bytes，需使用 SCONF 指令设定。

3. PLC 由 STOP 进入 RUN 状态时，执行 SCONF 指令，修改 Socket 参数，详情请参考 API2209 SCONF 指令说明。
4. 当 M0=ON 时(Network 3)检查欲联机的 Socket 是否处于关闭中(SM1274=ON)未联机(SM1270=OFF)以及未建立联机中(SM1273=OFF)的状况，若条件成立则进入联机程序，联机建立完成后(SM1270=ON)，清除 M0。
5. 远程 Client（IP=192.168.1.106）进行联机，并传送 2K byte 之数据
6. 开启 Monitor Table，并监控起始接受数据地址 D15000，可确认是否确实收到 2K#2 数据如下：

D15000			2000

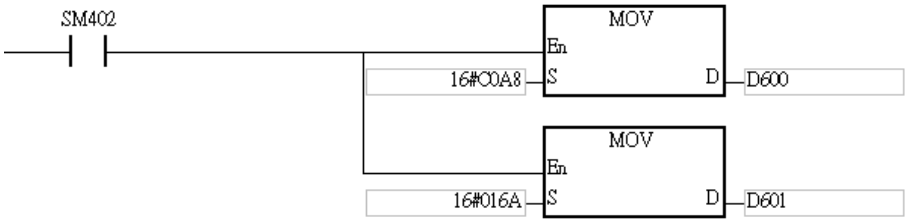
注#2：若数据已经满，需重新传送数据，可重置 D15000 数值为 0，重新接收数据。

7. 当 M3=ON 时（Network 5），准备关闭联机，检查欲关闭的 Socket 是否为已联机状态（SM1270=ON），若为联机中则开始关闭联机。

范例程序如下所示：

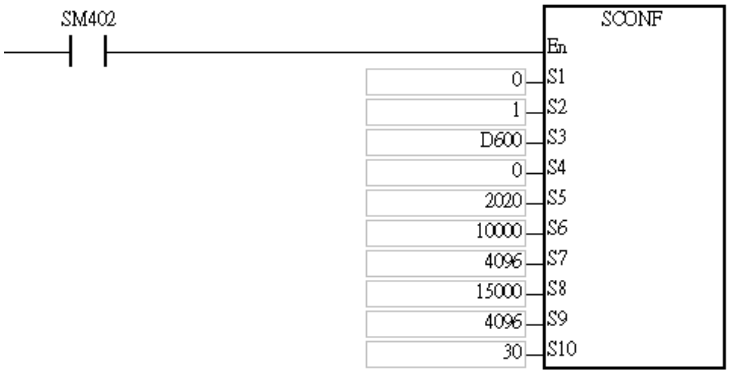
Network 1

Set TCP Client IP



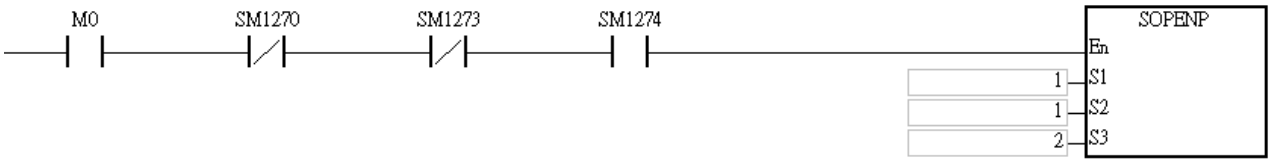
Network 2

Set Sconfig



Network 3

Create TCP Conneciton



Network 4

Network 5

Close TCP connection

补充说明：

1. TCP 联机执行错误说明：

错误码（SR180）	错误标志	错误说明
16#2003	SM0	**S**$_1$、**S**$_2$和 **S**$_3$内容值超出范围
16#600C	SM1109	本机通讯端口已被使用
16#600D	SM1100	Ethernet 网络未连接
16#6200	TCP Socket 错误标志	TCP Socket IP 地址不合法
16#6201	TCP Socket 错误标志	TCP Socket 通讯模式设定不合法
16#6202	TCP Socket 错误标志	TCP Socket 模式设定不合法
16#6203	TCP Socket 错误标志	TCP Socket 传送数据地址不合法
16#6204	TCP Socket 错误标志	TCP Socket 传送数据长度超过范围
16#6205	TCP Socket 错误标志	TCP Socket 传送数据装置超过范围
16#6206	TCP Socket 错误标志	TCP Socket 接收数据地址不合法
16#6207	TCP Socket 错误标志	TCP Socket 接收数据长度超过范围
16#6208	TCP Socket 错误标志	TCP Socket 接收数据装置超过范围
16#6212	TCP Socket 错误标志	TCP Socket 联机通讯逾时
16#6213	TCP Socket 错误标志	TCP Socket 实际接收数据大于设定的接收数据
16#6214	TCP Socket 错误标志	TCP Socket 联机被远程设备拒绝
16#6215	TCP Socket 错误标志	TCP Socket 联机未开启
16#6217	TCP Socket 错误标志	TCP Socket 联机开启已被触发
16#6218	TCP Socket 错误标志	TCP Socket 数据传送已被触发
16#621A	TCP Socket 错误标志	TCP Socket 联机关闭已被触发

2. UDP 联机执行错误说明：

错误码（SR180）	错误标志	错误说明
16#2003	SM0	**S**$_1$、**S**$_2$和 **S**$_3$内容值超出范围
16#600C	SM1109	本机通讯端口已被使用
16#600D	SM1100	Ethernet 网络未连接
16#6209	UDP Socket 错误标志	UDP Socket IP 地址不合法
16#620A	UDP Socket 错误标志	UDP Socket 通讯模式设定不合法
16#620C	UDP Socket 错误标志	UDP Socket 传送数据地址不合法
16#620D	UDP Socket 错误标志	UDP Socket 传送数据长度超过范围
16#620E	UDP Socket 错误标志	UDP Socket 传送数据装置超过范围
16#620F	UDP Socket 错误标志	UDP Socket 接收数据地址不合法

错误码（SR180）	错误标志	错误说明
16#6210	UDP Socket 错误标志	UDP Socket 接收数据长度超过范围
16#6211	UDP Socket 错误标志	UDP Socket 接收数据装置超过范围
16#6213	UDP Socket 错误标志	UDP Socket 实际接收数据大于设定的接收数据
16#6215	UDP Socket 错误标志	UDP Socket 联机未开启
16#6217	UDP Socket 错误标志	UDP Socket 联机开启已被触发
16#6218	UDP Socket 错误标志	UDP Socket 数据传送已被触发
16#621A	UDP Socket 错误标志	UDP Socket 联机关闭已被触发

3. 当客户 (Client) / 服务器 (Server) 端皆为使用 AS 系列 PLC，且都有设定通讯逾时时间时，若是 Server 端先发生逾时，则 Server 端会自行切断联机，而 Client 端不会有错误标志发生；反之，Client 端先发生逾时，则 Client 端会发生错误标志，并主动切断联机。

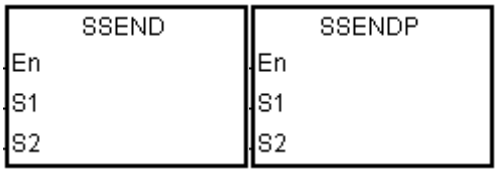
API	指令码			操作数	功能
2201		SSEND	P	S_1 · S_2	透过已开启的 Socket 传送数据

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●			○	○	○		
S_2	●	●			●	●		●	●			○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： Socket 模式.

S_2 ： Socket 编号

指令说明：

1. **S_1** 为 Socket 模式，当 **S_1** 为 1 表示开启 TCP Socket，当 **S_1** 为 0 表示开启 UDP Socket。
2. **S_2** 为 Socket 编号，需对应以太网络中 Socket 设定，最多可设定 4 个。
3. **S_1**=0~1（默认值 0），**S_2**=1~4（默认值 1）；当超出范围时，PLC 会显示错误码（H'2003）表示装置超过可用范围。
4. 使用本指令前，需先以 SOPEN 开启 Socket 联机，当已联机标志（TCP）或是已开启标志（UDP）为 ON 时，方可使用本指令进行数据传输。
5. 透过 TCP Socket 传送数据时，当此指令执行后，若无错误，则开始传送数据，传送中标志为 ON，若传送成功则已传送标志为 ON，传送中标志为 OFF，反之若有错误则对应错误标志 ON。

TCP Socket 编号	数据传送中标志	数据传送完成标志	错误标志
1	SM1275	SM1272	SM1277
2	SM1283	SM1280	SM1285
3	SM1291	SM1288	SM1293
4	SM1299	SM1296	SM1301

6. 透过 UDP Socket 传送数据时，当此指令执行后，若无错误，则数据已传送标志为 ON，反之若有错误则对应错误标志 ON。

UDP Socket 编号	数据传送完成标志	错误标志
1	SM1336	SM1338
2	SM1341	SM1343
3	SM1346	SM1348
4	SM1351	SM1353

7. 本指令一般都是使用脉冲执行型指令（SSENDP）。

程序范例：

请参考 SOPEN 程序范例。

补充说明：

1. TCP 联机执行错误说明：

错误码	错误标志	错误说明
16#2003	SM0	**S**$_1$ 和 **S**$_2$ 内容值超出范围
16#600D	SM1100	Ethernet 网络未连接
16#6202	TCP Socket 错误标志	TCP Socket 模式设定不合法
16#6203	TCP Socket 错误标志	TCP Socket 传送数据地址不合法
16#6204	TCP Socket 错误标志	TCP Socket 传送数据长度超过范围
16#6205	TCP Socket 错误标志	TCP Socket 传送数据装置超过范围
16#6212	TCP Socket 错误标志	TCP Socket 联机通讯逾时
16#6214	TCP Socket 错误标志	TCP Socket 联机被远程设备拒绝
16#6215	TCP Socket 错误标志	TCP Socket 联机未开启
16#6218	TCP Socket 错误标志	TCP Socket 数据传送已被触发

2. UDP 联机执行错误说明：

错误码	错误标志	错误说明
16#2003	SM0	**S**$_1$ 和 **S**$_2$ 内容值超出范围
16#600D	SM1100	Ethernet 网络未连接
16#620A	UDP Socket 错误标志	UDP Socket 通讯模式设定不合法
16#620C	UDP Socket 错误标志	UDP Socket 传送数据地址不合法
16#620D	UDP Socket 错误标志	UDP Socket 传送数据长度超过范围
16#620E	UDP Socket 错误标志	UDP Socket 传送数据装置超过范围
16#6218	UDP Socket 错误标志	UDP Socket 数据传送已被触发

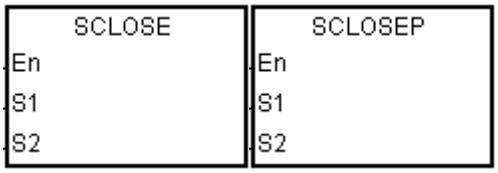
API	指令码			操作数	功能
2203		SCLOSE	P	S_1 · S_2	关闭 Socket

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●			○	○	○		
S_2	●	●			●	●		●	●			○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： Socket 模式

S_2 ： Socket 编号

指令说明：

1. **S_1** 为 Socket 模式，当 **S_1** 为 1 表示开启 TCP Socket，当 **S_1** 为 0 表示开启 UDP Socket。
2. **S_2** 为 Socket 编号，需对应以太网络中 Socket 设定，最多可设定 4 个。
3. **S_1**=0~1（默认值 0），**S_2**=1~4（默认值 1）；当超出范围时，PLC 会显示错误码（H'2003）表示装置超过可用范围。
4. 关闭 Socket 前必须确认此 Socket 为联机状态，否则此指令不会执行。
5. TCP Socket 若由主端执行关闭指令，则从端会继续对本机通讯端口进行监听（联机中标志为 ON）；若由从端执行关闭指令，则从端在关闭完成后也取消对本机通讯端口的监听，除上述之外，关闭后对应的标志皆为 OFF。
6. UDP Socket 执行关闭指令后，对应的标志皆为 OFF。
7. 关闭 TCP Socket 时，当此指令执行后，若无错误，则开始关闭与远程通讯端口的联机，关闭中标志为 ON，若关闭成功则关闭中标志为 OFF，反之若有错误则对应错误标志 ON。

TCP Socket 编号	联机关闭中标志	错误标志
1	SM1274	SM1277
2	SM1282	SM1285
3	SM1290	SM1293
4	SM1298	SM1301

8. 关闭 UDP Socket 时，当此指令执行后，若无错误，则已联机标志为 OFF，反之若有错误则对应错误标志 ON。

UDP Socket 编号	错误标志
1	SM1338
2	SM1343

UDP Socket 编号	错误标志
3	SM1348
4	SM1353

9. 本指令一般都是使用脉冲执行型指令（SCLOSEP）。

程序范例：

请参考 SOPEN 程序范例。

补充说明：

1. TCP 联机执行错误说明：

错误码	错误标志	错误说明
16#2003	SM0	S_1 和 S_2 内容值超出范围
16#600D	SM1100	Ethernet 网络未连接
16#6212	TCP Socket 错误标志	TCP Socket 联机通讯逾时
16#6214	TCP Socket 错误标志	TCP Socket 联机被远程设备拒绝
16#621A	TCP Socket 错误标志	TCP Socket 联机关闭已被触发

2. UDP 联机执行错误说明：

错误码	错误标志	错误说明
16#2003	SM0	S_1 和 S_2 内容值超出范围
16#600D	SM1100	Ethernet 网络未连接
16#621A	UDP Socket 错误标志	UDP Socket 联机关闭已被触发

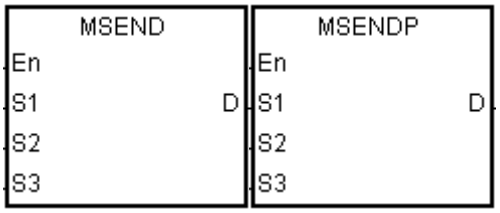
API	指令码			操作数	功能
2204		MSEND	P	**S_1**、**S_2**、**S_3**、**D**	寄送电子邮件

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●	●			○	○	○		
S_2	●	●			●	●		●	●							
S_3	●	●			●	●		●	●							
D	●	●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1 ： 邮件组态

S_2 ： 邮件主旨

S_3 ： 邮件信息

D ： 指令完成标志

指令说明：

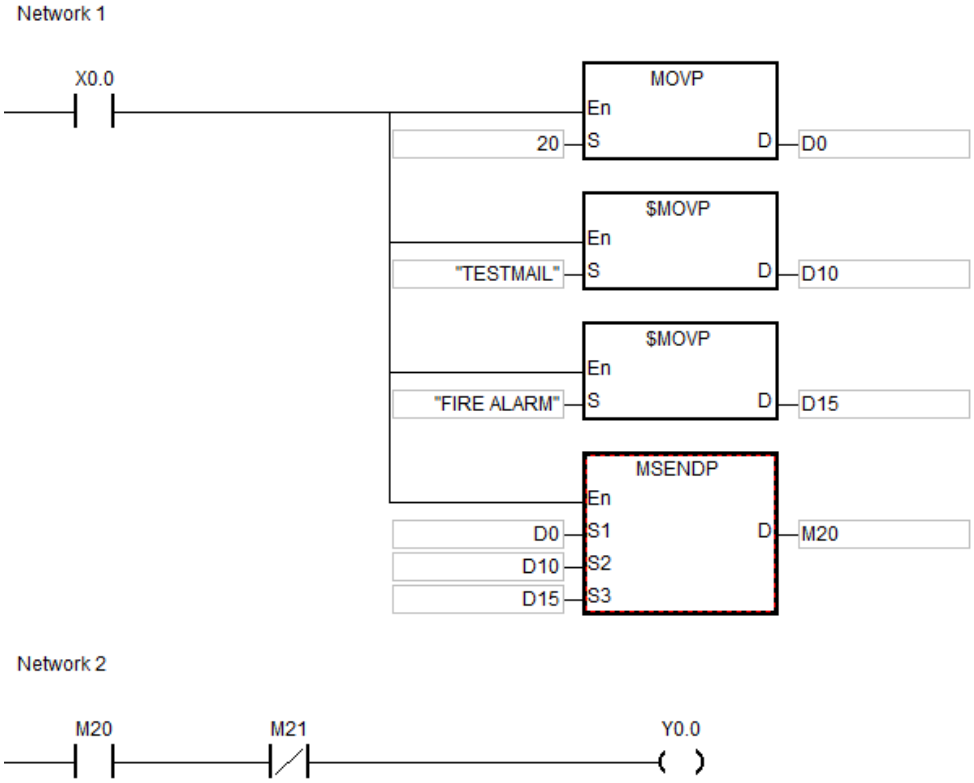
1. 利用 **S_1**、**S_2** 和 **S_3** 的设定转为所需的数据，寄出 Email。
2. 使用本指令前，需用 ISPSoft 完成以下设定。
   - 在 PLC 参数设定 - > 基本设定中设定本机的 IP 地址、网络屏蔽。
   - 在 PLC 参数设定 - > 进阶设定 - > 邮件下设定邮件服务器的 IP 地址、通讯端口、本机邮件地址和主旨。
   - 在 PLC 参数设定 - > 进阶设定 - > 邮件地址与发送条件设定 - > 邮件地址设定要欲寄送的邮件位置。
   - 若外寄服务器需要身分验证，可在 PLC 参数设定 - > 进阶设定 - > 邮件下设定用户名称和密码。
3. 邮件设定如下：

操作数	描述	设定范围
S_1	远程邮件编号	Bit0~Bit3 分别代表在 ISPSoft 中设定的远程邮件地址(ISPSoft 中邮件编号为 4 组)。Bit0 代表远程地址 1，以此类推。 若欲传送至此邮件地址，须建立对应的位。
S_2	主旨	邮件主旨最大支持 16 字符。
S_3	信息	邮件信息最大支持 64 字符。
D	指令完成标志	指令完成后，该位为 ON。若指令完成异常，下一个位 (D+1) 为 ON。

4. 本指令一般都是使用脉冲执行型指令（MSENDP）。

程序范例：

当 X0.0=ON 时，寄一封邮件给远程邮件编号 1 和编号 3（D0=00000101），当与 SMTP 服务器通讯完毕后 M20=ON，若通讯过程中无错误 M21=OFF，输出 Y0.0=ON。



补充说明：

1. **S_2** 或 **S_3** 的字符串长度，系统以抓取终止符（16#00）为主，若达长度上限仍未读取终止符，则字符串长度等于上限值。
2. **D** 操作数，若使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。
3. **S_2** 与 **S_3** 之间请保留一个字符，供本指令产生中断字符使用。
4. 指令执行错误说明：

错误码	错误标志	错误说明
16#2003	SM0	（1）**D+1 超出装置范围** （2）**S_1**<1 或 **S_1**>15
16#600D	SM1100	Ethernet 网络未连接
16#6100	D+1	E-mail 联机忙线中
16#6107	D+1	E-mail 联机通讯逾时
16#6108	D+1	SMTP 服务器账号密码验证错误
16#6111	D+1	远程邮件地址无效

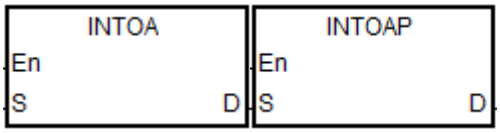
API	指令码			操作数	功能
2206		INTOA	P	**S**、**D**	IP→字符串变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S								●	●							
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



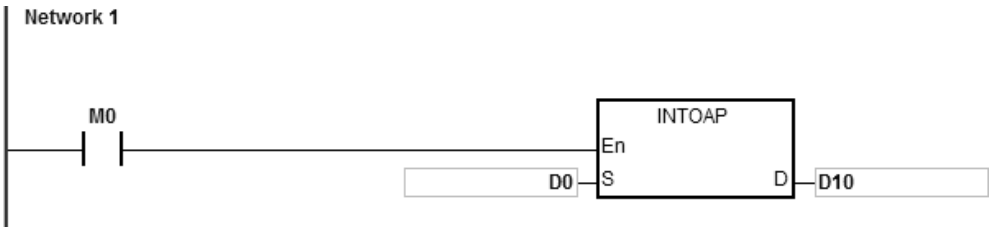
S ： 要转换的来源值

D ： 存放转换后的结果

指令说明：

1. 将 **S** 与 **S**+1 的内容值转换成字符串类型的 IP 地址，并且将结果存到 **D**。
2. **D** 连续占用 8 个装置。

程序范例一：



D0	16#C0(192)	16#A8(168)
D1	16#00(0)	16#01(1)

执行后

D10	″9″	″1″
D11	″.″	″2″
D12	″6″	″1″
D13	″.″	″8″
D14	″0″	″0″
D15	″.″	″0″
D16	″0″	″0″
D17	16#00	″1″

"192.168.0.1"

补充说明：

1. **S** 操作数使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。
2. **D** 操作数使用 ISPSoft 宣告，则数据类型为 ARRAY [8] of WORD/INT。

API	指令码			操作数	功能
2207		IATON	P	**S**、**D**	字符串→IP 变换

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S								●	●						○	
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

S ： 要转换的来源值

D ： 存放转换后的结果

指令说明：

1. 将 **S** 的字符串 IP 地址转换成整数类型，并且将结果存到 **D** 与 **D**+1。
2. **S 最多**连续占用 8 个装置。

"192.168.0.1"

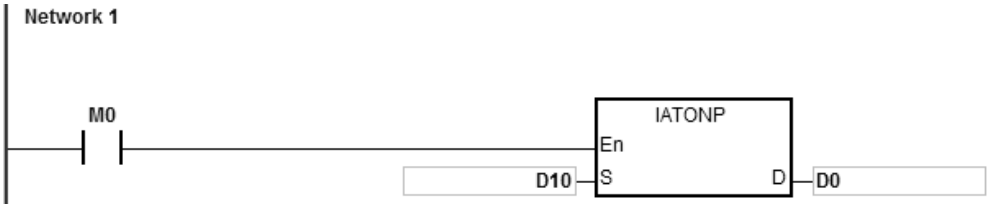
S+0	″9″	″1″
S+1	″.″	″2″
S+2	″6″	″1″
S+3	″.″	″8″
S+4	″0″	″0″
S+5	″.″	″0″
S+6	″0″	″0″
S+7	16#00	″1″

执行后

D+0	16#C0(192)	16#A8(168)
D+1	16#00(0)	16#01(1)

3. **S** 的字符串 IP 地址来源，每个区段数值固定为 1~3 个字符，中间以一个“.”（16#2E）隔开，例如：“192.168.0.1”，只要直接输入即可，不必输入“192.168.000.001”。
4. **S** 的字符串 IP 地址来源，每个区段数值转换出来的结果要在 0~255 的范围内。

程序范例一：



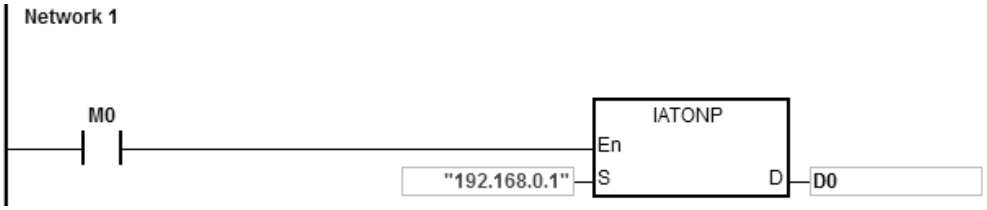
"192.168.0.1"

D10	"9"	"1"
D11	"."	"2"
D12	"6"	"1"
D13	"."	"8"
D14	"0"	"0"
D15	"."	"0"
D16	"0"	"0"
D17	16#00	"1"

執行后⇨

D0	16#C0(192)	16#A8(168)
D1	16#00(0)	16#01(1)

程序范例二：



"192.168.0.1" 執行后⇨

D0	16#C0(192)	16#A8(168)
D1	16#00(0)	16#01(1)

6

补充说明：

1. **S** 字符串中没有 16#00 当结尾，SM0=ON，SR0 错误码为 16#200E。
2. **S** 字符串除了小数点，其余要转成 BIN 值的部份不在 16#30~16#39 范围内，指令不执行，SM0=ON，SR0 错误码为 16#2003。
3. **S** 字符串中小数点“.”符号的个数不等于 3 个时，指令不执行，SM0=ON，SR0 错误码为 16#2003。
4. **S** 的字符串来源，若有任一个区段的数值转换出来的结果不在 0~255 的范围内，指令不执行，SM0=ON，SR0 错误码为 16#2003。
5. **S** 的字符串 IP 地址来源，每个区段数值固定 1~3 个字符，若超过则指令不执行，SM0=ON，SR0 错误码为 16#2003。
6. **S** 操作数使用 ISPSoft 宣告，则数据类型为 ARRAY [8] of WORD/INT。
7. **D** 操作数使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD/INT。

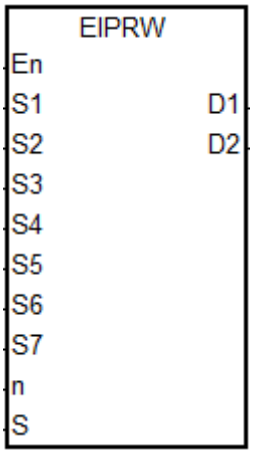
API	指令码		操作数	功能
2208		EIPRW		**S_1~S_7，n，S，D_1，D_2**	EtherNet/IP 读写通讯指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●	●							
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●	●				○	○		
S_6	●	●	●	●												
S_7								●	●				○	○		
n								●	●				○	○		
S								●								
D_1								●								
D_2								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1~D_2	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1~S_7，n，S ：请参考指令说明

D_1~D_2 ：请参考指令说明

指令说明：

1. **S_1~S_7，n，S** 与 **D_1~D_2** 装置名称与说明如下表所示。

装置	名称	使用说明	数据类型	备注
S_1	IP 地址	第 1 个 word 为前两组 IP 地址，第 2 个 word 为后两组 IP 地址。举例：IP=192.168.1.5，则 **S_1**=16#C0A8，**S_1**+1=16#0105	WORD[2]	连续占用 2 个 word
S_2	联机模式	0：UCMM	WORD	

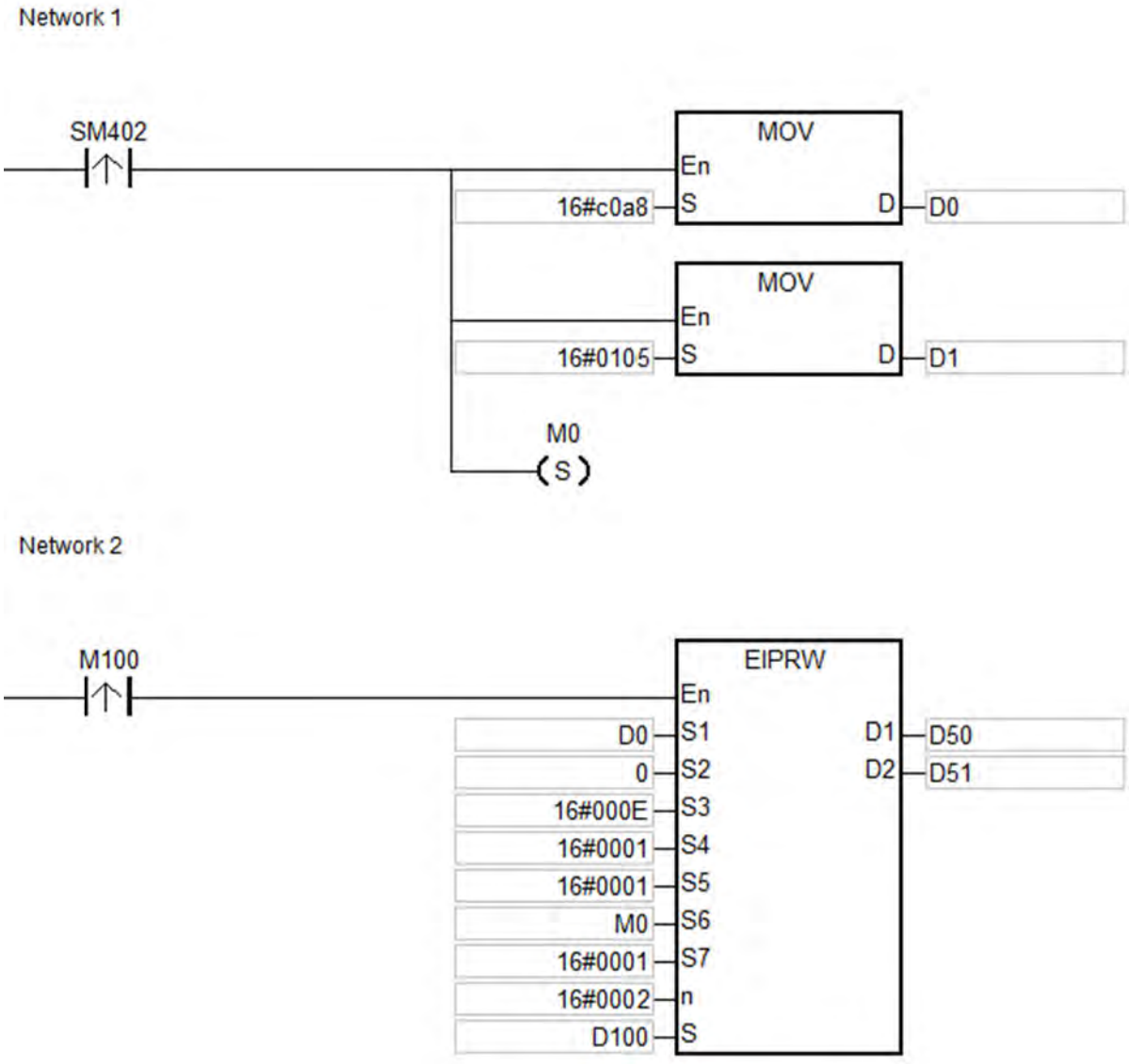
装置	名称	使用说明	数据类型	备注
S_3	功能码（Service Code）	功能码范围 16#0000~00FF，超出此范围不执行	WORD	
S_4	类别（Class）ID	请参考 EtherNet/IP 协议说明	WORD	
S_5	需求（Instance）ID	请参考 EtherNet/IP 协议说明	WORD	
S_6	属性（Attribute）ID 开关	ON 为启动，OFF 为关闭	BOOL	
S_7	属性（Attribute）ID	请参考 EtherNet/IP 协议说明	WORD	
n	读写数据长度 n	读写数据长度设定值，单位 bytes，最大长度限制为 200 bytes	WORD	
S	读写数据目的或来源	发送数据的来源位置，或者接收数据的存放位置	WORD[**n**]	
D_1	通讯执行状态	0：未触发执行 1：通讯执行中 2：通讯完成且无发生错误 3：通讯发生错误 4：参数设定错误	WORD	
D_2	错误码	主错误码与扩展错误码	WORD[2]	连续占用 2 个 word

2. 指令初次被启动时，即表示发送此通讯指令。若此时已达最大联机数时，则 **D_1** 通讯执行状态为 3，扩展错误码 **D_2[1]**会显示 16#6306，表示联机已满。
3. 当设定的参数数值超出限制范围时，此通讯指令不执行，并且设定 **D_1** 通讯执行状态为 4。
4. 此指令建议使用上升沿触发，若持续触发会造成错误，主错误码 **D_2[0]**会显示 16#02，表示指令重复触发，请于指令完成后再执行下一次触发。

程序范例一：

读取设备（192.168.1.5）的厂商代码，数据读取结果存放于 D100，EtherNet/IP Object 参数如下：

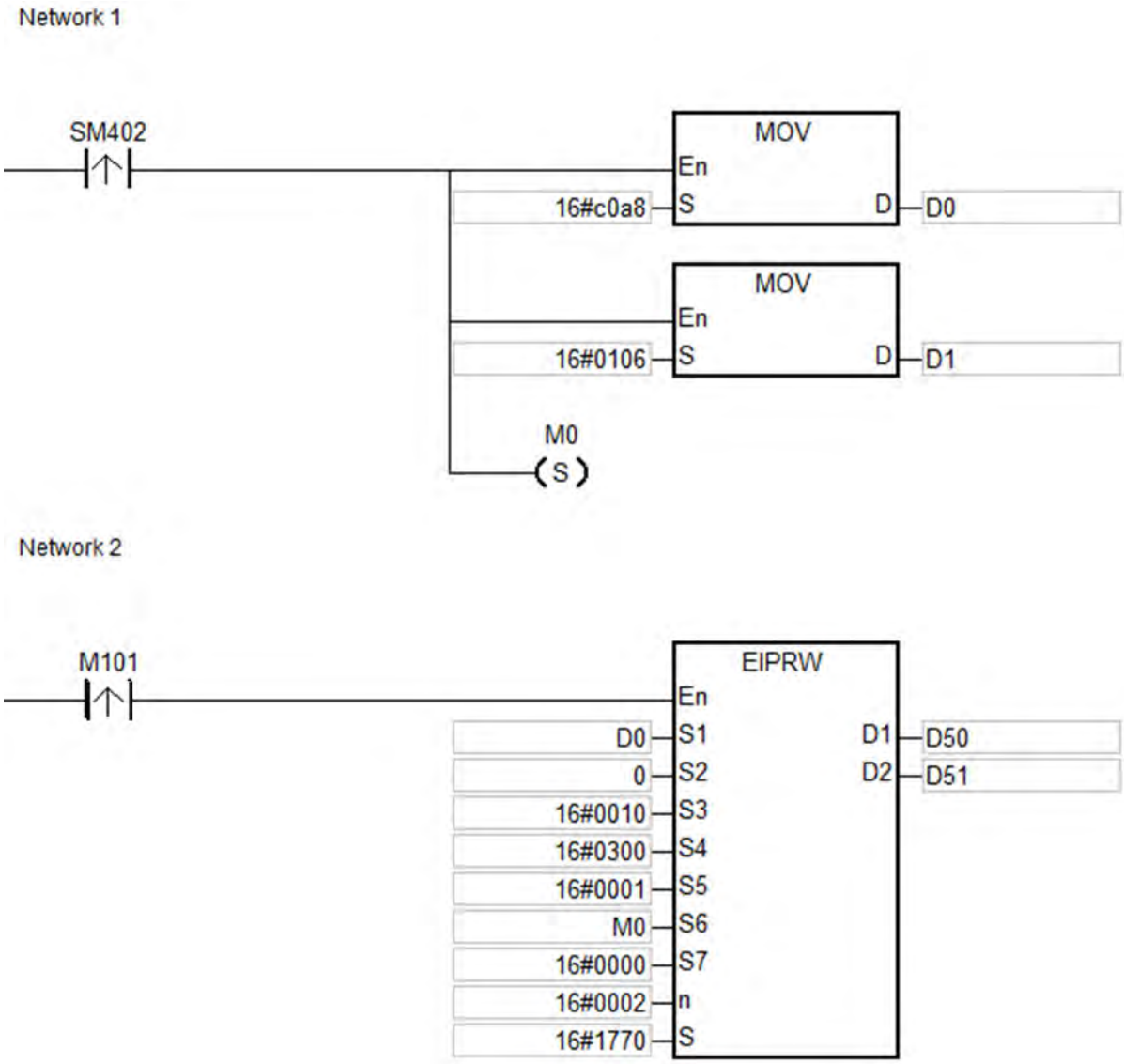
(1) Class ID = 1
(2) Instance ID = 1
(3) Attribute ID = 1



程序范例二：

设定变频器（192.168.1.6）最高操作频率数值（01-00）为 60.00Hz，EtherNet/IP Object 参数定义如下：

(1) Class ID = 16#0300
(2) Instance ID = 16#0001
(3) Attribute ID = 16#0000



补充说明：

1. **S_1** 操作数使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD。
2. **S** 操作数使用 ISPSoft 宣告，则数据类型请以读写长度 **n** 设 ARRAY [**n**] of WORD。
3. **D_2** 操作数使用 ISPSoft 宣告，则数据类型为 ARRAY [2] of WORD。
4. 主错误码 **D_2 [0]**状态说明

错误码 **D_2 [0]**	**状态名称**	错误处置
16#00	成功	
16#01	联机错误	检查从站 EDS File 内容是否正确

错误码 **D_2 [0]**	**状态名称**	错误处置
16#02	资源不可用	(1) 检查主站连接设备是否超过限制 (2) 检查从站连接设备是否超过限制 (3) 指令重复触发，请于前一次操作完成后再执行触发
16#03	参数设定错误	检查读写数据内容(S)是否正确
16#04	路径段落错误	检查类别 ID(S_4)、需求 ID(S_5)和属性 ID(S_7)设定是否正确
16#05	目的路径不存在	检查类别 ID(S_4)、需求 ID(S_5)和属性 ID(S_7)设定是否正确
16#07	联机中断	(1) 检查从站设备 Ethernet 端口连接是否正确 (2) 检查从站设备 Keep alive timer 设定
16#08	服务码不支持	检查功能码(S_3)设定是否正确
16#09	错误属性值	检查读写数据源(S)与内容设定是否正确
16#0A	属性列表错误	检查从站设备对象属性是否允许执行 Get_Attribute_List 和 Set_Attribute_List 功能码
16#0B	发送冲突	检查服务是否有重复设定
16#0C	对象状态冲突	检查是否有建立 Owner IO 联机
16#0D	对象已存在	检查从站是否已支持设定对象，若是已支持则不需执行服务
16#0E	属性不可写	检查对象属性是否支持写入功能
16#0F	权限不足	检查从站设备是否允许执行此服务码
16#10	设备目前状态无法执行服务	检查是否有建立 Owner IO 联机
16#11	响应数据长度过大	检查对象属性数据长度是否超过限制(100 words)
16#12	存取 Tag 时使用，存取数据长度与数据类型不符	检查存取数据长度与数据类型设定是否正确
16#13	传送数据不足	检查读写长度(n)设定是否正确
16#14	属性值不支持	检查属性 ID 开关(S_6)和属性 ID(S_7)设定是否正确
16#15	传送数据过多	检查读写长度(n)设定是否正确
16#16	对象不存在	检查类别 ID(S_4)设定是否正确
16#17	存取 Tag 时使用，存取数据顺序错误	(1) 检查 Ethernet 网络连接是否正确 (2) 检查 Ethernet 通讯环境是否发生封包遗失
16#18	属性值未保存	检查从站设备状态是否发生错误
16#19	属性值执行保存错误	检查从站设备硬件是否发生错误
16#1A	路由错误，需求封包长度超过	检查读写数据长度(n)是否超过路由器限制
16#1B	路由错误，回应封包长度超过	检查读写数据长度(n)是否超过路由器限制

错误码 **D_2 [0]**	**状态名称**	错误处置
16#1F	自定义对象存取错误	请参考从站设备错误定义
16#20	不合法的参数	检查读写数据内容(S)是否正确

5. 扩展错误码 **D_2 [1]**说明：

错误码 **D_2 [1]**	**错误标志**	**错误说明**
16#2003	SM0 / **D_1**	（1）**S_1**、**S_2**、**S_3**、**S_6** 内容值超出范围 （2）**S + n** 超出装置地址范围
16#200B	SM0	**n** 操作数不在范围
16#600D	SM1100	Ethernet 网络未连接
16#6301	**D_1**	远程设备中断联机
16#6302	**D_1**	远程设备响应逾时
16#6303	**D_1**	IP 地址不合法
16#6304	**D_1**	回应命令 Service Code 错误
16#6305	**D_1**	回应命令长度错误
16#6306	**D_1**	所有联机忙碌中

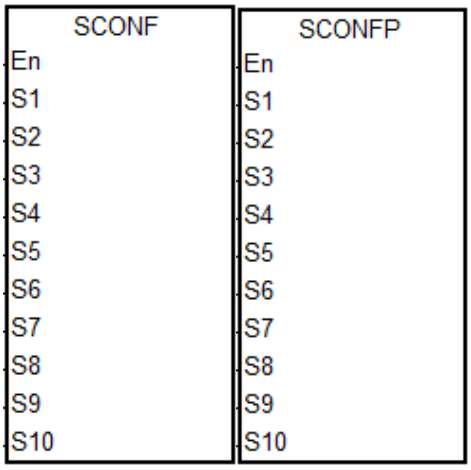
API	指令码			操作数	功能
2209		SCONF	P	S_1~S_{10}	TCP/UDP Socket 参数设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●	●				○	○		
S_6								●	●				○	○		
S_7								●	●				○	○		
S_8								●	●				○	○		
S_9								●	●				○	○		
S_{10}								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1~S_{10}	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1~S_{10} ：请参考指令说明

指令说明：

1. S_1~S_{10} 装置名称与说明如下表所示。

装置	名称	使用说明	数据类型	备注
S_1	TCP/UDP 选择	0：TCP，1：UDP	WORD	
S_2	Socket 编号	数值范围为 1~4	WORD	
S_3	远程 IP 地址	第 1 个 word 为前两组 IP 地址，第 2 个 word 为后两组 IP 地址。例：IP=192.168.1.5，则 S_3=16#C0A8，S_3+1=16#0105	WORD[2]	连续占用 2 个 word

装置	名称	使用说明	数据类型	备注
S_4	远程端口	0~65535，其中 0 表示任意埠	WORD	
S_5	本地端口	0~65535，其中 0 表示任意埠	WORD	
S_6	发送数据源位置	指定 D 装置元件之编号，数值范围为 0~29999，举例：设定 100，即表示由 D100 位置开始发送 **S_7** 个 Byte（下 8 位，上 8 位顺序传送）	WORD	
S_7	发送数据长度	最大长度限制为 512 Byte V1.06.30 版（含）以上，最大值为 4096 bytes。	WORD	
S_8	接收数据目的位置	指定 D 装置元件之编号，数值范围为 0~29999，举例：设定 200，即表示由 D200 位置开始接收存放 **S_9** 个 Byte（下 8 位，上 8 位顺序存放）	WORD	
S_9	接收数据长度	最大长度限制为 512Byte V1.06.30 版（含）以上，最大值为 4096 bytes。	WORD	
S_{10}	联机保持时间设定	数值范围为 1~30000，单位为秒	WORD	仅用于 TCP 模式

2. 使用此指令时，建议运用脉冲型指令方式设定。
3. Socket 设定之参数值，皆以 HWCONFIG 所设定之数值为默认值，若是执行过程中，需要任意变换参数内容值时，才需要使用此指令变更。
4. 当设定参数时，该 Socket 编号正在通讯中，则需等到通讯完成之后，此次设定值才能生效。但建议设定前最好确认该 Socket 未被使用中之后，再进行设定。
5. 当参数设定值其中任何一个不合限制范围内之数值时，指令将不执行，并且设定 SM0=ON，错误代码 SR0=16#2003。

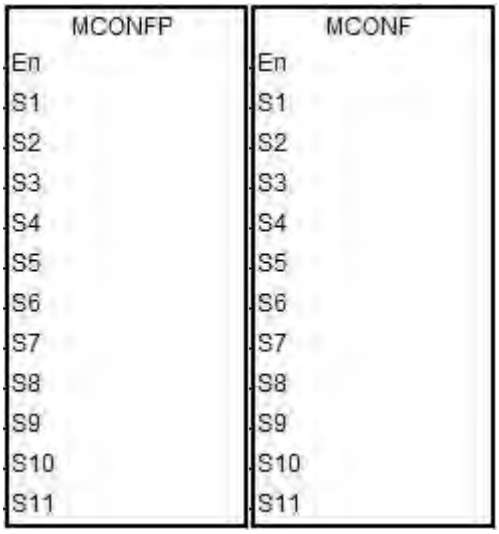
API	指令码			操作数	功能
2210		MCONF	P	**S_1~S_{11}**	MODBUS TCP 数据交换设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●							
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●	●				○	○		
S_6								●	●				○	○		
S_7			●					●								
S_8								●	●				○	○		
S_9								●	●				○	○		
S_{10}								●	●				○	○		
S_{11}			●					●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1~S_{11}	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S_1~S_{11} ：请参考指令说明

指令说明：

1. **S_1~S_{11}** 装与装置名称与说明如下表所示。

装置	名称	使用说明	数据类型	备注
S_1	Data Exchange Table No.	MODBUS TCP 数据交换表编号 数值范围为 1~32	WORD	

装置	名称	使用说明	数据类型	备注
S_2	远程 IP 地址	第 1 个 word 为前两组 IP 地址，第 2 个 word 为后两组 IP 地址。举例：IP=192.168.1.5，则 **S_2**=16#C0A8，**S_2**+1=16#0105	WORD[2]	连续占用 2 个 word
S_3	远程站号	0~255，若超出此范围，将自动以下 8 位数值执行	WORD	
S_4	读取通讯功能码	支持 16#01，02，03，04，17，若超出此范围，则不执行接收参数(含地址与长度)修改。	WORD	
S_5	读取远程通讯地址	16#0000~16#FFFF	WORD	
S_6	读取数据长度	• 位(Bit)功能码 16#01, 02，可读取长度范围为 1~256。 • 字符(Word)功能码 16#03, 04, 17，可读取长度范围为 1~100。 当填入数值为 0(或小于 0)时，即表示不启动读取功能，其它超出范围的数值，皆会以最大值执行。	WORD	
S_7	本地接收数据存放寄存器	位(Bit)功能码：仅能选用 M 字符(Word)功能码：仅能选用 D	BOOL WORD	
S_8	写入通讯功能码	支持 16#05，06，0F，10，若超出此范围，则不执行发送参数(含地址与长度)修改。	WORD	
S_9	写入远程通讯地址	16#0000~16#FFFF	WORD	
S_{10}	写入数据长度	• 位(Bit)功能码 16#0F 的写入长度范围为 1~256；功能码 16#05 只能填入 1，V1.06.00 版(含)以后支持填 0。 • 字符(Word)功能码 16#10 的写入长度范围为 1~100；功能码 16#06 只能填入 1，V1.06.00 版(含)以后支持填 0。 当填入数值为 0(或小于 0)时，即表示不启动写入功能，其它超出范围的数值，皆会以最大值执行。	WORD	
S_{11}	本地写入远程数据寄存器	位(Bit)功能码：仅能选用 M 字符(Word)功能码：仅能选用 D	BOOL WORD	

2. 使用此指令时，建议使用脉冲型指令方式设定。

3. **S_4** 和 **S_8**MODBUS 通讯功能码说明：

   AS 的读取多笔位装置（Bit）(非 Discrete Input 装置）命令为 1（16#01），

   AS 的读取多笔位装置（Bit）(仅 Discrete Input 装置）命令为 2（16#02），

   AS 的读取多笔字符装置（Word）(非 Input Register 装置）命令为 3（16#03），

   AS 的读取多笔字符装置（Word）(仅 Input Register 装置）命令为 4（16#04），

   AS 的单笔位装置（Bit）状态写入命令为 5（16#05），

   AS 的单笔字符装置（Word）数据写入命令为 6（16#06），

   AS 的多笔位装置（Bit）状态写入命令为 15（16#0F），

   AS 的多笔字符装置（Word）数据写入命令为 16（16#10）。

   AS 的多笔字符装置（Word）数据写入命令为 23（16#17）。

   目前仅支持上述功能码，其余功能码将无法执行。

4. 此指令所设定之参数值，仅在 PLC 运行过程有效，当 PLC 断电之后，重新上电时，皆以软件所设定之数据交换表内数值为默认值，若是执行过程中，有需要变换参数内容值时，才需要使用此指令进行修改。

5. 当设定参数时，若是该指定编号正在通讯中，则需等到当次通讯完成之后，下次通讯时新的设定值才能生效。

6. 当参数设定值其中任何一个不合限制范围内之数值时，指令将不执行，并且设定 SM0=ON，错误代码 SR0=16#200B。

7. AS 系列主机不支持类似 DVP 系列的 ETHRW（MODBUS TCP）通讯指令，若需要采用 PLC 程序进行以太网 MODBUS TCP 通讯控制时，请于数据交换表内启动条件选择“程控”模式，并且搭配下表 SM 编号控制与读取：

SM 编号	属性	Ethernet 数据交换参数内容说明
SM1167	R/W	数据交换启动标志
SM1168 ~ SM1199	R/W	数据交换联机编号 1~32 之启动标志
SM1200 ~ SM1231	R	数据交换联机编号 1~32 之成功标志
SM1232 ~ SM1263	R	数据交换联机编号 1~32 之错误标志

   当数据接收正确时，成功标志将会被设为 ON，当接收错误发生时，错误标志将会被设为 ON。每一个联机编号之成功标志与错误标志，绝不会同时间一起为 ON。

8. 数据交换功能搭配之 SR 状态说明如下（皆为只读寄存器）：

SR 编号	说明
SR1120 ~ SR1151	分别为联机编号 1~32 之实际执行通讯时间
SR1152 ~ SR1183	分别为联机编号 1~32 之通讯错误代码

SR1152~ SR1183 错误代码定义如下：

错误代码	说明
16#00XX	远程设备响应错误通讯代码
16#F000	Ethernet 网络未连接
16#F001	远程设备响应逾时
16#F003	TCP 联机通讯逾时
16#F007	回应命令错误
16#F009	远程设备中断联机

API	指令码			操作数	功能
2211		EMCONF1	P	**Server ~ Passw**	Email server 参数设定

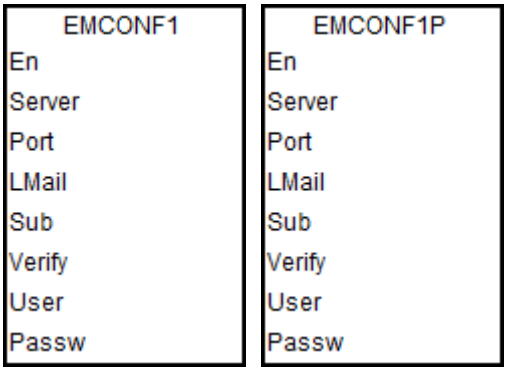
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Server								●	●						○	
Port								●	●				○	○		
LMail								●	●						○	
Sub								●	●						○	
Verify								●	●				○	○		
User								●	●						○	
Passw								●	●						○	

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

Server~Passw ： 请参考指令说明



指令说明：

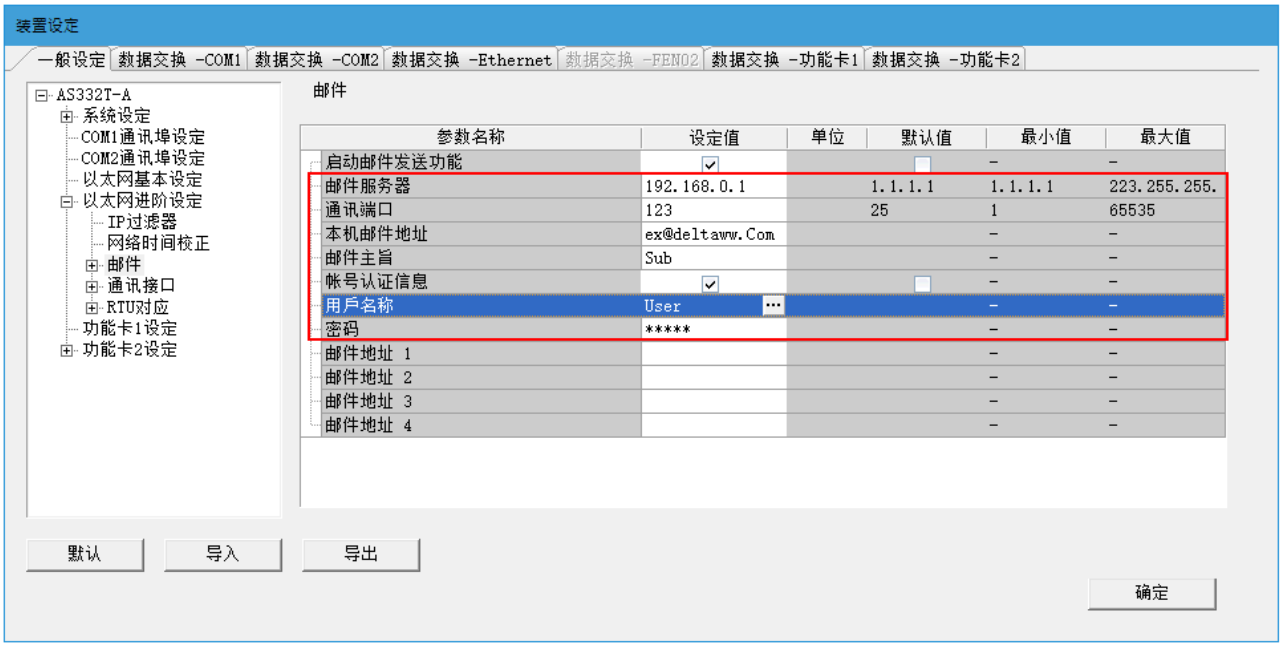
1. V1.06.00 版（含）之后支持此指令。
2. 操作数之名称、使用说明与数据类型如下表所示。

操作数	名称	使用说明	数据类型	备注
Server	邮件服务器	使用装置： **Sever**+0：数据长度，单位 bytes（数值只能为 0 或 4，0 表示不修改） **Sever**+1~**Sever**+2：数据内容（数据从下 8 位开始摆放） 例如：192.168.0.1 ➔ D0 = 16#0004，D1 = 16#0001， D2 = 16#C0A8	WORD[N]	占用地址参考 **Sever**+0，最多连续占用 3 个 word

操作数	名称	使用说明	数据类型	备注
		使用字符串常数：长度最多 15 个字符 每个区段数值固定为 1~3 个字符，中间以一个”.”（16#2E）隔开，例如：”192.168.0.1”	String	
Port	通讯端口	数值范围：1~65535，0：表示不修改	WORD	
LMail	本机邮件地址	使用装置: **LMail**+0 :数据长度，单位 bytes(数值范围为 0~63，0：表示不修改） **LMail**+1~ **LMail**+32：数据内容（数据从下 8 位开始摆放） 例如：ex@deltaww.com D0 = 16#000E，D1 = 16#7865， D2 = 16#6440，D3 = 16#6C65， D4 = 16#6174，D5 = 16#7777， D6 = 16#2E63，D7 = 16#6D6F，	WORD[N]	占用地址参考 **LMail** +0，最多连续占用 33 个 word
		使用字符串常数：长度最多 63 个字符 例如：”ex@deltaww.com”	String	
Sub	邮件主旨	使用装置： **Sub** +0 :数据长度，单位 bytes（数值范围为 0~31，0：表示不修改） **Sub**+1~ **Sub**+16：数据内容（数据从下 8 位开始摆放） 例如：Sub D0 = 16#0003，D1 = 16#7553， D2 = 16#0062，	WORD[N]	占用地址参考 **Sub** +0，最多连续占用 17 个 word
		使用字符串常数：字符串长度最多 31 个字符 例如： “Sub”		
Verify	认证启动	0：表示不修改，1：关闭认证，2：启动认证，	WORD	
User	使用者名称	使用装置： **User**+0 :数据长度，单位 bytes（数值范围为 0~31，0：表示不修改） **User**+1~ **User** +16 数据内容（数据从下 8 位开始摆放） 例如：User	WORD[N]	占用地址参考 **User** +0，最多连续占用 17 个 word

操作数	名称	使用说明	数据类型	备注
		D0 = 16#0004，D1 = 16#7355， D2 = 16#7265，		
		使用字符串常数：长度最多 31 个字符 例如：”User”	String	
Passw	密码	使用装置： **Passw** +0：数据长度，单位 bytes（数值范围为 0~20，0：表示不修改） **Passw**+1~ **Passw**+10：数据内容（数据从下 8 位开始摆放） 例如：Passw D0 = 16#0005，D1 = 16#5061， D2 = 16#7373，D3 = 16#0077	WORD[N]	占用地址参考 **Passw** +0，最多连续占用 11 个 word
		使用字符串常数：长度最多 20 个字符 例如：”Passw”		

上述参数对应 HWCONFIG 以太网进阶设定-邮件



3. Email server 相关参数，将在上电时从 HWCONFIG 加载设定值，之后可借由此指令修改其参数内容，但被修改的内容将不会被停电保持。

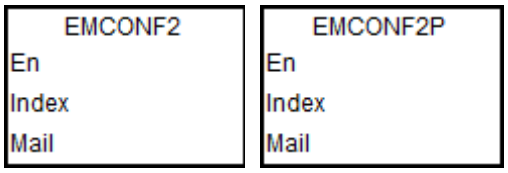
API	指令码			操作数	功能
2212		EMCONF2	P	**Index，Mail**	Email 地址设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Index								●	●				○	○		
Mail								●	●							

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



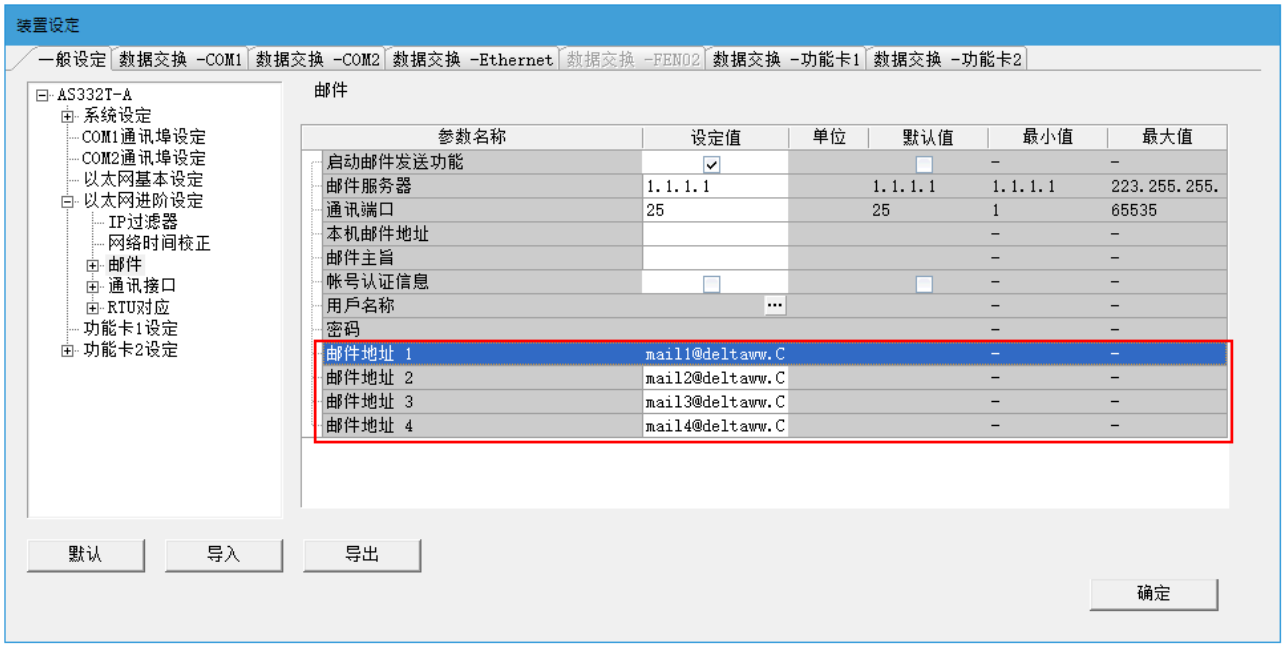
Index，Mail ： 请参考指令说明

指令说明：

1. V1.06.00 版（含）以后支持此指令。
2. 操作数之名称、使用说明与数据类型如下表所示。

操作数	名称	使用说明	数据类型	备注
Index	邮件地址编号	**Index**：欲修改的邮件地址编号（数值范围为 0~4，0：表示邮件地址不修改）	WORD	
Mail	邮件地址	使用装置： **Mail** +0：邮件地址编号 **Index** 数据长度，单位 bytes（数值范围为 0~63，0：表示邮件地址不修改） **Mail** +1~ **Mail** +32 :邮件地址编号 **Index** 数据内容（数据从下 8 位开始摆放） 例如：ex@deltaww.com D0 = 16#000E，D1 = 16#7865， D2 = 16#6440，D3 = 16#6C65， D4 = 16#6174，D5 = 16#7777， D6 = 16#632E，D7 = 16#6D6F	WORD[N]	占用地址参考 **Mail** +0，最多连续占用 33 个 word
		使用字符串常数：长度最多 63 个字符 例如：”ex@deltaww.com”	String	

上述参数对应 HWCONFIG 以太网进阶设定-邮件



3. 邮件地址相关参数，将在上电时从 HWCONFIG 加载设定值，之后可借由此指令修改其参数内容，但被修改的内容将不会被停电保持。

API	指令码	操作数	功能
2214	MQTT_Connect	**Execute ~ Status**	MQTT Client 联机指令

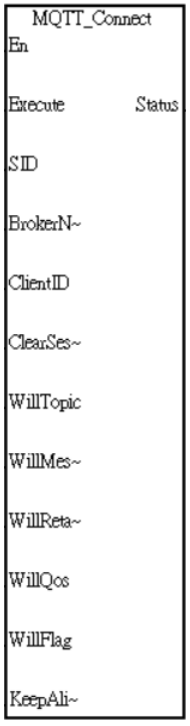
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Execute	●	●	●	●				●		●	●					
SID								●	●		●		●			
BrokerNum								●	●		●		●			
ClientID								●	●		●				●	
CleanSession	●	●	●	●				●		●	●					
WillTopic								●	●		●				●	
WillMessage								●	●		●				●	
WillRetain	●	●	●	●				●		●	●					
WillQos								●	●		●		●			
WillFlag	●	●	●	●				●		●	●					
KeepAlive								●	●		●		●			
Status								●			●					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Execute ~ Status	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
--	AS	-

6

符号：



Execute ~ Status ：请参考指令说明

指令说明：

1. ISPSoft 支持软件版本 V3.16（含）以上，AS300 主机韧体版本 V1.12（含）以上，安装 AS-FFTP01 功能卡时使用。
2. 使用前请先参考 AS 系列模块手册 AS-FFTP01 功能说明，于 HWCONFIG 中将 AS-FFTP01 通讯卡的 MQTT 参数设定完成。
3. 装置名称与说明如下表所示

装置	名称	使用说明	数据类型	备注
Execute	执行	• 上升沿触发执行 1 次联机 • 下降沿触发执行 1 次断线	BOOL	
SID	插槽编号	固定输入 34，代表 AS-FFTP01 功能卡	WORD	
BrokerNum	服务器（Broker）编号	固定输入 1	WORD	
ClientID	客户端名称(Client ID)	数据以先上 8 位，再下 8 位顺序摆放，最大支持长度为 20 个字符，需包含字符串终止符（16#00）	WORD	请参考下方客户端名称说明
CleanSession	清除会话（Session）	• ON：断线后服务器将清除会话 • OFF：断线后服务器将保留会话	BOOL	请参考下方会话说明
WillTopic	Will 信息主题	数据以先上 8 位，再下 8 位顺序摆放，最大支持长度为 100 字符，需包含字符串终止符（16#00）	WORD	请参考下方 Will 信息功能说明
WillMessage	Will 信息	数据以先上 8 位，再下 8 位顺序摆放，最大支持长度为 100 字符，需包含字符串终止符（16#00）	WORD	
WillRetain	保留 Will 信息	设定 Will 信息是否要保留(Retain) 在服务器上。	BOOL	
WillQos	Will 信息服务质量	设定 Will 信息的服务质量（Quality of Service），允许范围设定值：0、1	WORD	
WillFlag	Will 信息功能启动标志	• ON：启动 Will 信息功能 • OFF：关闭 Will 信息功能		
KeepAlive	联机保持时间	此联机将要求服务器，在空闲时间超过此设定值的 1.5 倍时，主动关闭此联机，单位为秒。 • 允许范围设定值：0~65535。 • 建议值：60。	DWORD	

装置	名称	使用说明	数据类型	备注
		• 设定为 0 时，服务器将不被要求主动关闭闲置联机。 实际允许连接时间会根据服务器账户设定而定，请参考服务器平台的开发人员指南进行设定。		
Status	状态	• Status[0] 状态代码 0：联机未建立/断线成功 1：联机成功已建立 2：联机/断线进行中 4：发生错误，将产生错误代码 • Status[1] 错误代码，请参考下方错误代码说明	WORD[2]	连续占用 2 个 word。更新时机请参考下方时序图说明。

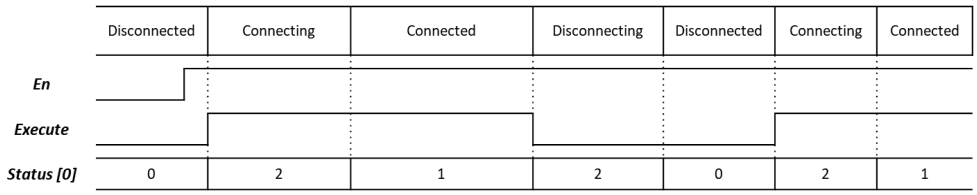
4. 使用 MQTT 发布（Publish）或订阅（Subscribe）前，需确认联机建立成功，建议使用 MQTT_Connect 状态代码作为发布与订阅功能的启动判断，详情请参考 API2215 MQTT_Publish 与 API2216 MQTT_Subscribe 的操作范例。
5. 客户端名称（Client ID）说明：
   (1) 一个客户端使用一个独特的名称，用于服务器辨识客户端，以确认其会话（Session）的状态。必须为字符串，内容限定为数字与英文字母。
   (2) 当客户端名称设定为 0 时，CleanSession 必须设定为 ON。
6. 会话（Session）说明：
   (1) 清除会话（Clean Session）：表示断线后客户端信息会被服务器删除，先前的会话将会被移除。建议使用于不断发布的信息或可接受信息丢失的应用。
   (2) 保留会话（Persistent Session）：表示断线后服务器会保留会话，包含未完成的信息。重新联机后，将自动恢复所有订阅并传送已存放的信息。建议使用于订阅者不可丢失信息、订阅者信息必须被保存的应用。
   (3) 会话中实际存放信息与信息内容会根据服务器帐户限制而定，必须参考服务器平台的开发人员指南进行设定。
7. Will 信息功能（Will Message）说明：
   (1) Will 信息功能包含参数有：Will 信息主题（WillTopic）、Will 信息（WillMessage）、保留 Will 信息（WillRetain）、Will 信息服务质量（WillQos）与 Will 信息启动标志（WillFlag）。
   (2) 启动 Will 信息功能时，需先设定好以上参数，联机前将 WillFlag 设定为 ON，再进行联机。当设备不正常断线时（服务器没有收到断线指令却发现联机中断），服务器会将 Will 信息发送给其他有订阅此 Will 信息主题的设备。
   (3) WillTopic 与 WillMessage 特性与一般信息发布功能相同，请参考 API2215 MQTT_Publish 功能说明。
   (4) WillRetain 特性与一般信息的保留信息功能（Retain）相同，请参考 API2215 MQTT_Publish 保留信息说明。

(5) WillQos 特性与一般信息的服务质量（Quality of Service）相同，请参考 API2215 MQTT_Publish 服务质量说明。

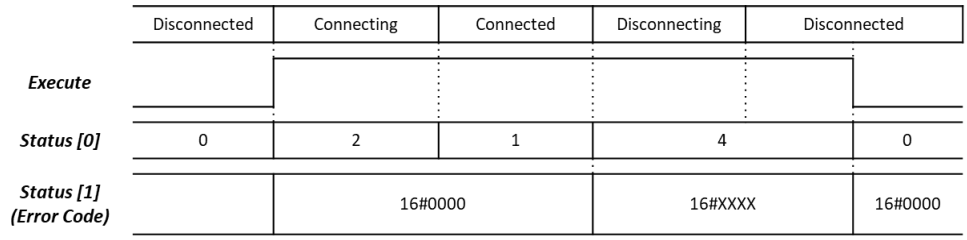
(6) 当 WillFlag 设定为 FALSE 时，WillRetain 必须设定为 OFF、WillQos 必须设定为 0。

8. 时序图

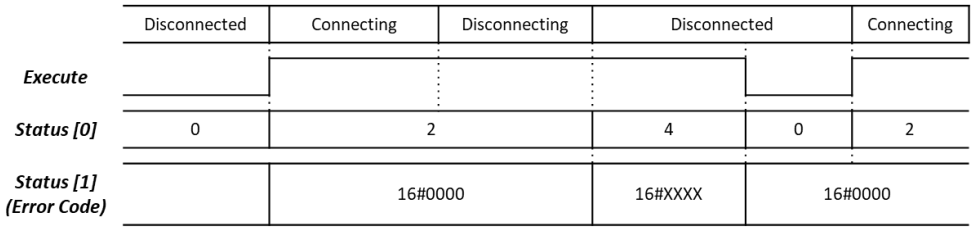
(1) 正常联机与正常断线



(2) 通讯中断线



(3) 尝试联机失败



9. 错误代码与说明如下表所示：（API2214 MQTT_Connect、API2215 MQTT_Publish、API2216 MQTT_Subscribe 共享错误码）

错误代码	错误码含义	处理方式
0x2200	SID 超出范围	修改指令中输入变量 SID 的值在范围内
0x2201	功能卡 2 不是 AS-FFTP01	确认功能卡 2 的卡别是否正确
0x2202	BrokerNum 超出范围	修改指令中输入变量 BrokerNum 的值在范围内
0x2207	MQTT 相关指令正在执行通讯	确认 MQTT 相关指令执行顺序，等待执行完成后(Status[0] = 0、1 或 4)，再执行下一个 MQTT 相关指令
0x2208	主机与 AS-FFTP01 通讯逾时	调整 HWCONFIG 功能卡 2 通讯逾时时间设定
0x2209	AS-FFTP01 忙碌中	请稍后再试

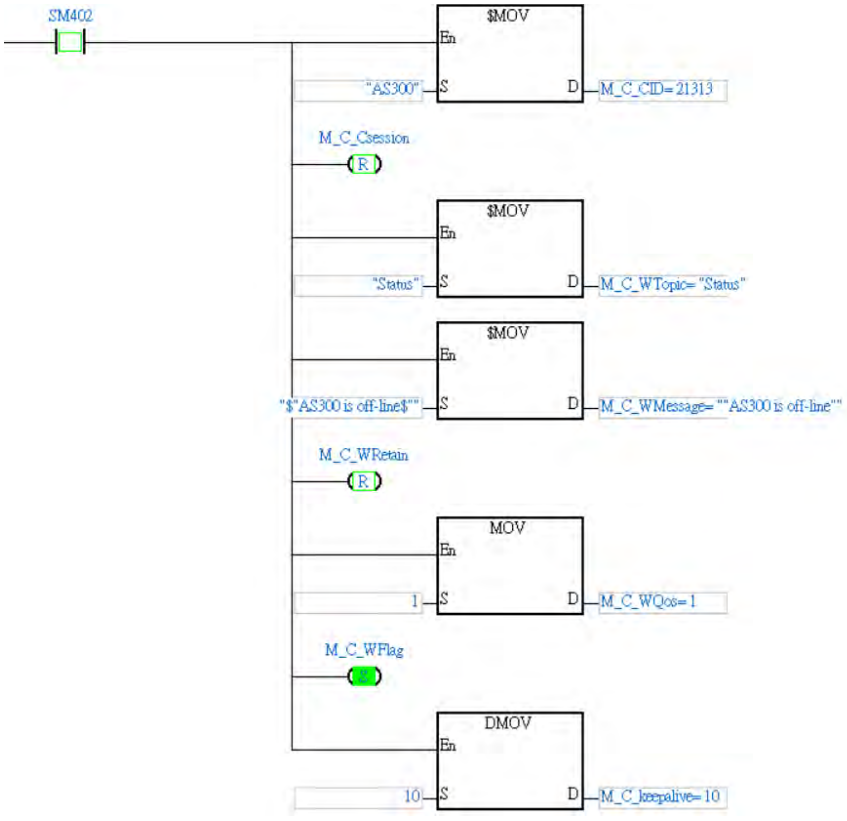
错误代码	错误码含义	处理方式
0x220A	AS-FFTP01 未 READY（SM172 OFF）	透过 AS-FFTP01 网页 System diagnostic 导出系统记录文件，并联络原厂协助分析
0x220B	Topic 字符串长度超出范围	修改指令中输入变量 Topic 字符串长度在范围内
0x220C	Message 字符串长度超出范围	修改指令中输入变量 Message 的字符串长度在范围内
0x220D	Qos 超出范围	修改指令中输入变 Qos 的值在范围内
0x8001	至少存在一笔无效的参数	检查 API 参数是否符合使用规范
0x8002	缓冲区已满	接收到的封包大小超过 1024 bytes
0x8003	MQTT 封包传送失败	检查以太网络是否正常
0x8004	MQTT 封包接收失败	检查以太网络是否正常
0x8005	接收到来自服务器的无效封包	• 确认服务器端运行状态是否正常 • 透过 AS-FFTP01 网页 System diagnostic 导出系统记录文件，并联络原厂协助分析
0x8006	服务器拒绝联机或订阅	确认服务器端运行状态是否正常
0x8008	记录到非法状态	透过 AS-FFTP01 网页 System diagnostic 导出系统记录文件，并联络原厂协助分析
0x8009	收到的封包信息与现有记录冲突	若联机状态不稳定，请尝试执行 MQTT_Connect 断线功能并重新联机。
0x800A	PINGRESP 等待逾时	• 检查以太网络是否正常 • 确认服务器端运行状态是否正常
0x800B	TLS 联机建立失败	• 检查以太网络是否正常 • 检查服务器终端地址信息是否正确 • 检查汇入的跟证书、凭证与私钥是否正确 • 检查服务器端是否仍然开启 • 网络环境防火墙设定是否允许使用通讯端口 8883
0x800C	MQTT 联机建立失败	确认网络状态再重新执行 MQTT_Connect 联机功能。
0x800D	服务器网域、汇入跟证书、凭证与私钥联机信息有缺漏	请于软件 HWCONFIG 的「MQTT 客户端设定」完成服务器终端地址、跟证书、凭证与私钥设定并下载。
0x800E	订阅主题为空	订阅主题不可为空
0x800F	订阅数已达数量上限	解除其他不需要的订阅主题后，再次触发订阅
0x8010	此联机不支持 Qos2	修改 QoS 等级为 1 或 0
0x8011	主题已在订阅中	已订阅的主题不允许再次订阅

程序范例：

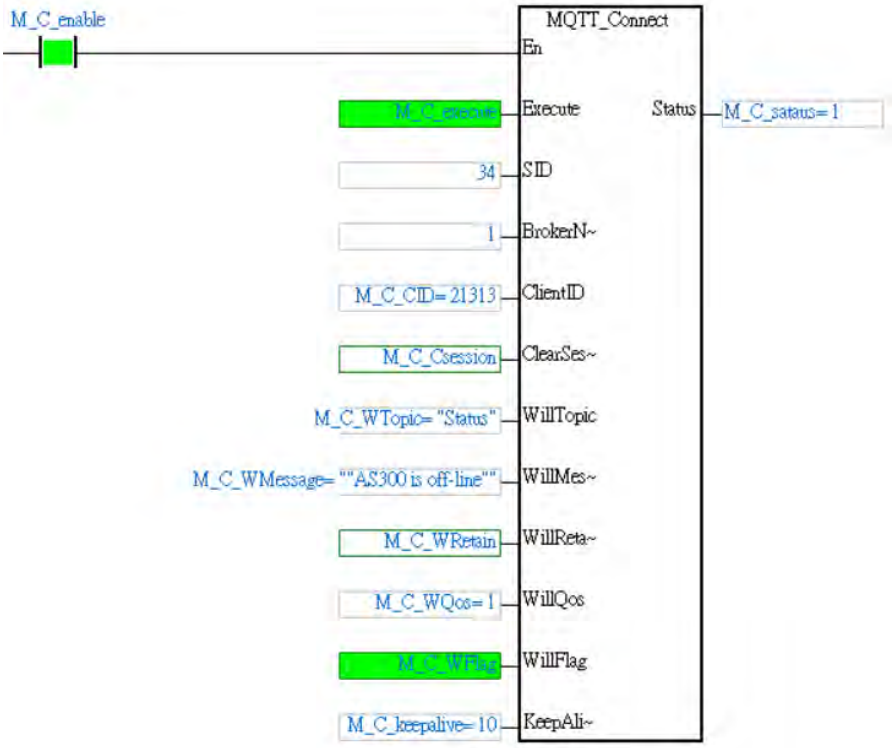
1. 参考 AS 系列模块手册 AS-FFTP01 功能说明，设定服务器与 HWCONFIG 中 MQTT 服务器终端地址、密钥与凭证等内容，并下载 AS-FFTP01 功能卡参数。

2. 参数说明

参数	内容	说明
SID	34	固定参数
BrokerNum	1	
ClientID	“AS300”	服务器会将此联机的会话与 Client ID: “AS300”连结
CleanSession	OFF	不清除会话，重新联机时，保留上次联机时的订阅内容
WillTopic	“Status”	• 有订阅 Will 信息主题为『Status』的客户端，将于通讯断线时收到信息『”AS300 is off-line”』 • 发送字符串时需夹带双引号，因此需在双引号”前使用前缀符号 $ ，才可将双引号作为字符串发送。
WillMessage	"$"AS300 is off-line$""	
WillRetain	OFF	服务器不会保留此 Will 信息
WillQos	1	Will 信息使用 QoS 等级 1 进行通讯 ,订阅者至少会收到一次
WillFlag	ON	启动 Will 信息功能
KeepAlive	10	当断线时间大于 15 秒 (1.5 倍的联机保持时间) 时，服务器将主动关闭联机，有订阅 Will 信息主题的订阅者将收到 Will 信息



3. 触发执行：依序设置 En 与 Execute 为 ON，当 Status[0] = 1 时即表示完成联机。



4. 发布与订阅：可将 Status[0] = 1 作为条件设定于 API2215 MQTT_Publish 与 API2216 MQTT_Subscribe 的 En，当完成联机后才能开启发布与订阅功能，参数设定与操作范例请参考 API2215 MQTT_Publish 与 API2216 MQTT_Subscribe 功能说明。
5. 将 Execute 设置为 OFF 时，AS-FFTP01 通讯卡会发出断线请求，当 Status[0] = 0 时即表示完成断线，此为正常的断线流程。
6. 当 AS-FFTP01 与服务器发生无预警断线时，有订阅主题『Status』的客户端会收到信息『"AS300 is off-line"』。

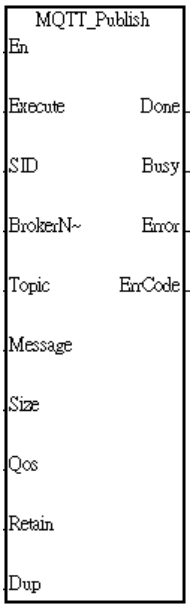
API	指令码	操作数	功能
2215	MQTT_Publish	**Execute ~ ErrCode**	MQTT Client 发布信息指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Execute	●	●	●	●				●		●	●					
SID								●	●		●		●			
BrokerNum								●	●		●		●			
Topic								●	●		●				●	
Message								●	●		●				●	
Size								●	●		●		●			
Qos								●	●		●		●			
Retain	●	●	●	●				●		●	●					
Dup	●	●	●	●				●		●	●					
Done		●	●	●				●		●	●					
Busy		●	●	●				●		●	●					
Error		●	●	●				●		●	●					
ErrCode								●			●					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Execute ~ ErrCode	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Execute ~ ErrCode ：请参考指令说明

指令说明：

1. ISPSoft 支持软件版本 V3.16（含）以上，AS300 主机韧体版本 V1.12（含）以上，安装 AS-FFTP01 功能卡时使用。
2. 使用前请先参考 AS 系列模块手册 AS-FFTP01 功能说明或下方范例，于 HWCONFIG 中将 AS-FFTP01 通讯卡的 MQTT 参数设定完成。
3. 使用 MQTT_Publish 功能前，需先使用 MQTT_Connect 成功建立联机，建议使用联机成功状态代码作为 API 的启动判断，详情请参考下方操作范例。
4. 装置名称与说明如下表所示

装置	名称	使用说明	数据类型	备注
Execute	执行	上升沿触发执行 1 次发布	BOOL	
SID	插槽编号	固定输入 34，代表 AS-FFTP01 功能卡	WORD	
BrokerNum	服务器（Broker）编号	固定输入 1	WORD	
Topic	主题	• 最大支持长度为 100 字符，长度包含字符串终止符（16#00）。 • 主题内容不可为空。	STRING[100]	
Message	信息	• 最大支持长度为 256 个字符，长度上限包含字符串终止符（16#00） 数据传送位顺序说明： • Size = 0：先下 8 位，再上 8 位 • Size ≠ 0：先上 8 位，再下 8 位	WORD[128]	
Size	信息数据长度	• 0：表示信息数据类型为字符串，用终止符判断（16#00） • 1-256：表示信息数据类型为 16 进位表示的原始资料	WORD	
Qos	服务质量（Quality of Service）	信息的服务质量，允许范围设定值：0、1	WORD	请参考服务质量功能说明
Retain	保留	设定信息是否要保留在服务器上。	BOOL	请参考保留信息功能说明
Dup	DUP 标志	允许用户可自行标记发布信息是否为重传信息。一般设定为 OFF。当 QoS 设定为 0 时，Dup 必须设定为 OFF。 • OFF：第一次发布此信息 • ON：标记此信息为重新传送的信息	BOOL	

装置	名称	使用说明	数据类型	备注
Done	完成	通讯完成	BOOL	
Busy	忙碌	通讯忙碌中	BOOL	
Error	错误	发生错误并产生错误代码	BOOL	
ErrCode	错误代码	错误代码请参考 API2214 MQTT_Connect 说明内容	WORD	

5. 服务质量（Quality of Service）功能说明：

   QoS 是一种对称的通讯协议，发布者和订阅者可自行设定自身的 QoS 等级，不同的等级决定其不同的传送或接收行为。AS-FFTP01 通讯卡支持 QoS 等级 1 与等级 2

   (1) QoS=0：不会确保信息是否被接收到，信息被接收与否取决于网络状态。
   - 发布者只送出一次信息。
   - 订阅者不会回复是否收到。
   - 发布 QoS=0 的信息时，Dup 必须设定为 OFF。

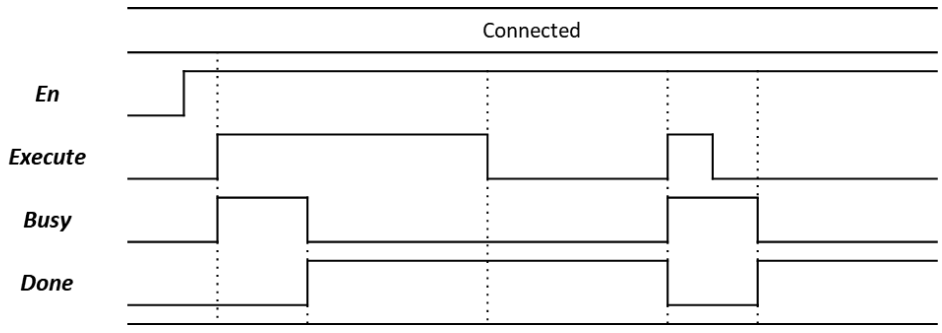
   (2) QoS=1：确保信息至少会被收到一次。
   - 订阅者收到信息后，会回复确认封包（PUBACK Packet）。
   - 发布者与服务器将会保留信息直到收到来自订阅者的确认封包。
   - 信息重传：若信息发布时，订阅者没有收到信息，则当下次订阅者重新联机，且联机时未清除会话（CleanSession = OFF）时，信息将会自动被重新传送

6. 保留（Retain）信息功能说明：

   (1) 当发布的信息 Retain = ON 时，服务器会将此信息以及其 QoS 信息保存下来，并将最后的保留信息（the last retained message）传送给未来所有此主题的新订阅者。

   (2) 当发布的信息 Retain = OFF 时，服务器不会保存此信息，且不会移除或取代任何已保存的保留信息。

7. 时序图：

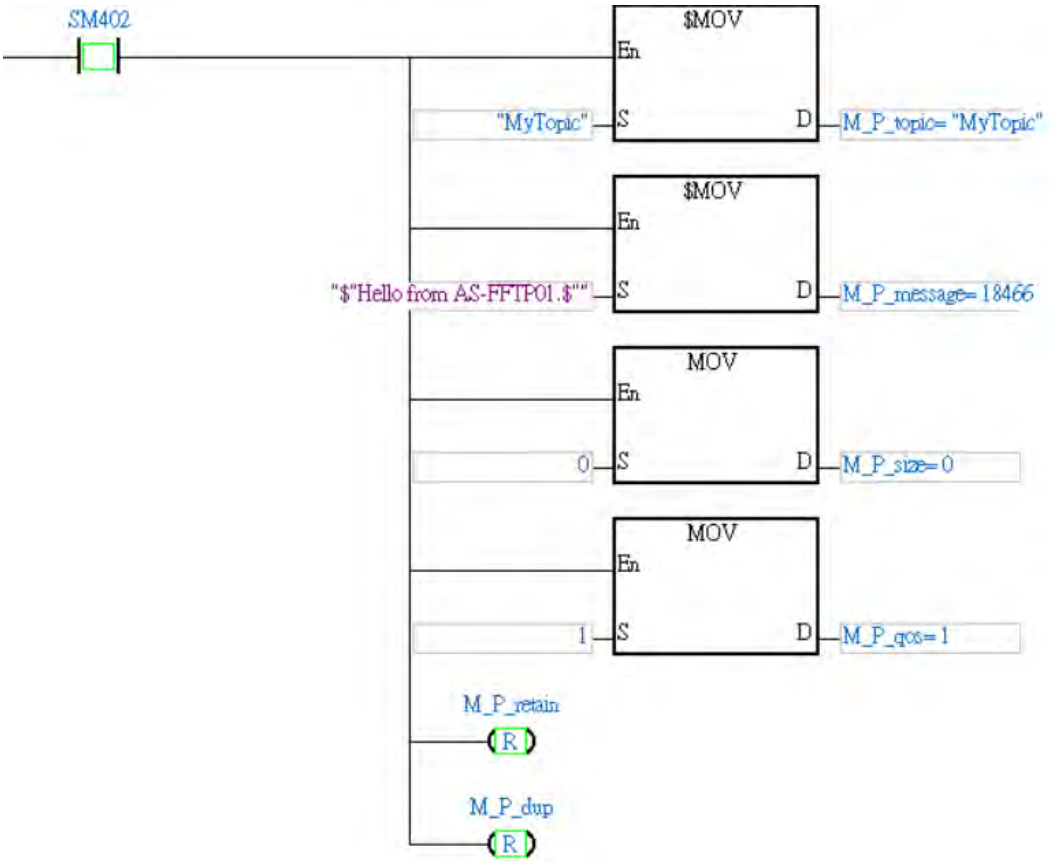


程序范例：

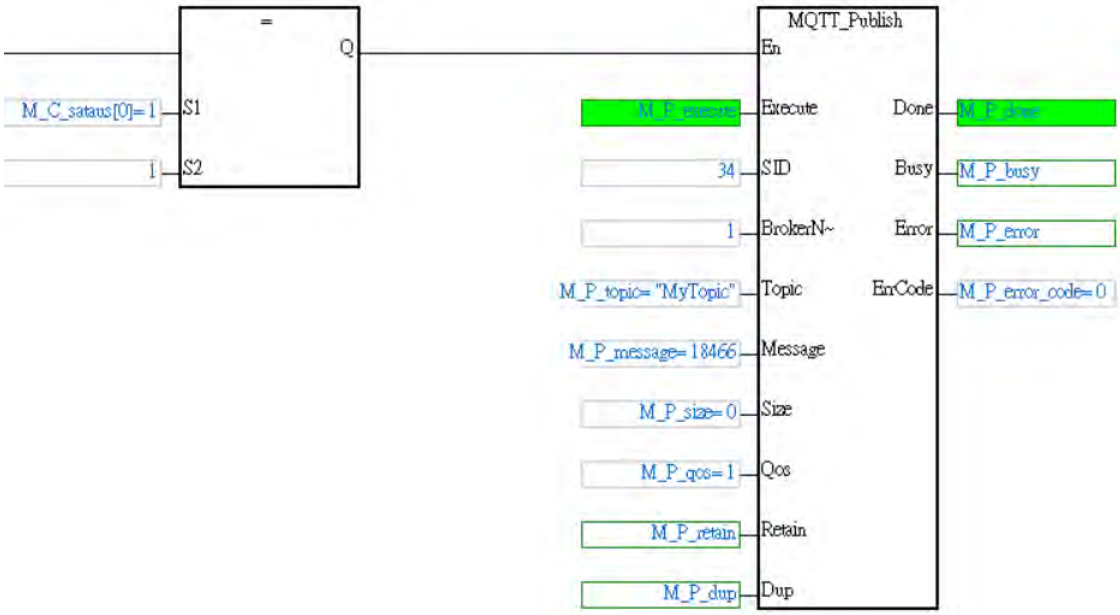
1. 参考 API2214 MQTT_Connect 说明，建立联机。

2. 参数说明

参数	内容	说明
SID	34	固定参数
BrokerNum	1	
Topic	“MyTopic”	• 有订阅主题『MyTopic』的客户端，将收到信息『"Hello from AS-FFTP01."』 • 发送字符串时需夹带双引号，因此需在双引号”前使用前缀符号 $ ，才可将双引号作为字符串发送。
Message	"$"Hello from AS-FFTP01.$""	
Size	0	设定为 0 表示传送字符串
Qos	1	确保信息至少会被收到一次
Retain	OFF	服务器不会保留此信息
Dup	OFF	不标记为重传信息



3. 触发执行：使用 MQTT_Connect 的 Status[0] = 1 作为条件连接 En，确保执行发布时联机是正常的。设定 Execute = ON 后，当 Done = ON 时，表示数据发布完成。



4. 发送字符串时（Size = 0），会先传送数据的下 8 位，再传送上 8 位，如下图为传送的原始数据：

M_P_message[0]		WORD	H"	leH"	ASCII
M_P_message[1]		WORD	le	olle	ASCII
M_P_message[2]		WORD	ol	f ol	ASCII
M_P_message[3]		WORD	f	orf	ASCII
M_P_message[4]		WORD	or	mor	ASCII
M_P_message[5]		WORD	m	SA m	ASCII
M_P_message[6]		WORD	SA	F-SA	ASCII
M_P_message[7]		WORD	F-	TFF-	ASCII
M_P_message[8]		WORD	TF	0PTF	ASCII
M_P_message[9]		WORD	0P	.10P	ASCII
M_P_message[10]		WORD	.1	*".1	ASCII
M_P_message[11]		WORD	*"	***"	ASCII
M_P_message[12]		WORD	**	****	ASCII

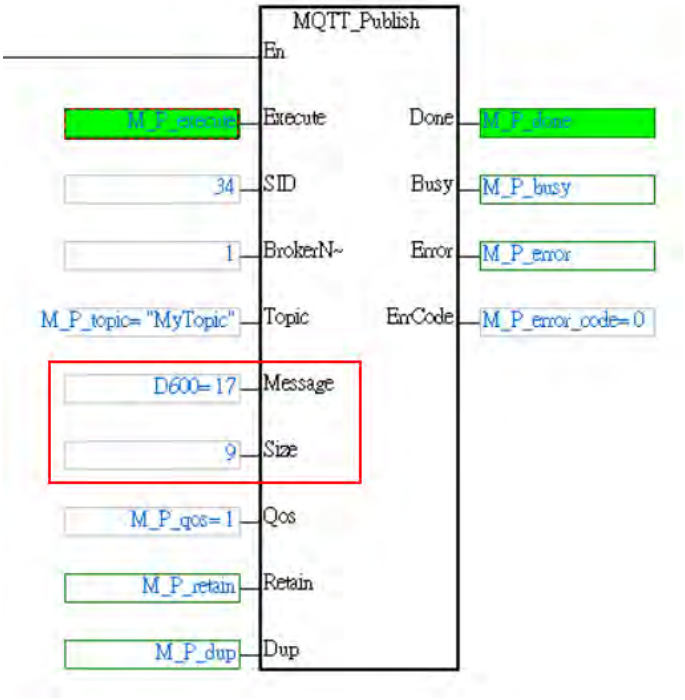
已订阅主题『MyTopic』的客户端将会收到信息『"Hello from AS-FFTP01."』。以使用 MQTT_Subscribe 接收的画面为例，如下图所示：

M_S_R_message[1]		WORD	"H	el"H	ASCII
M_S_R_message[2]		WORD	el	loel	ASCII
M_S_R_message[3]		WORD	lo	flo	ASCII
M_S_R_message[4]		WORD	f	ro f	ASCII
M_S_R_message[5]		WORD	ro	m ro	ASCII
M_S_R_message[6]		WORD	m	ASm	ASCII
M_S_R_message[7]		WORD	AS	-FAS	ASCII
M_S_R_message[8]		WORD	-F	FT-F	ASCII
M_S_R_message[9]		WORD	FT	P0FT	ASCII
M_S_R_message[10]		WORD	P0	1.P0	ASCII
M_S_R_message[11]		WORD	1.	"*1.	ASCII
M_S_R_message[12]		WORD	"*	**"*	ASCII

5. 由于设定发布 QoS 等级为 1，只要订阅者的 QoS 等级也为 1，订阅者就算在发布当下断线导致没有收到信息，只要恢复联机时设定没有清除会话（CleanSession = OFF），信息『"Hello from AS-FFTP01."』就会被重新发送。

6. 若改将 Retain 设定为 ON，当主题『MyTopic』有新订阅者一联机时，就会收到信息『"Hello from AS-FFTP01."』。

7. 当要传送非字符串的原始数据时（Size ≠ 0），会先传送数据的上 8 位，再传送下 8 位。可在 M_P_size 指定数据长度，例如，修改上述范例，改为发布从 D600 开始的 9 个 byte，并比对原始数据与使用 MQTT_Subscribe 接收的数据，如下图所示：



	D600		0011	22330011
	D601		2233	44552233
	D602		4455	66774455
	D603		6677	88006677
	D604		8800	00008800
M_S_R_message[0]		WORD	9	1114121
M_S_R_message[1]		WORD	0011	22330011
M_S_R_message[2]		WORD	2233	44552233
M_S_R_message[3]		WORD	4455	66774455
M_S_R_message[4]		WORD	6677	88006677
M_S_R_message[5]		WORD	8800	00008800
M_S_R_message[6]		WORD	0000	00000000

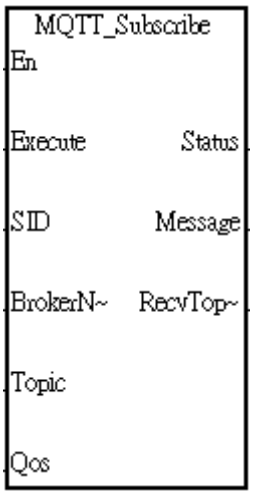
API	指令码	操作数	功能
2216	MQTT_Subscribe	**Execute ~ RecvTopic**	MQTT Client 订阅主题指令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
Execute	●	●	●	●				●		●	●					
SID								●	●		●		●			
BrokerNum								●	●		●		●			
Topic								●	●		●				●	
Qos								●	●		●		●			
Status								●			●					
Message								●			●					
RecvTopic								●			●					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Execute ~ RecvTopic	请参考指令说明												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Execute ~ RecvTopic ：请参考指令说明

指令说明：

1. ISPSoft 支持软件版本 V3.16（含）以上，AS300 主机韧体版本 V1.12（含）以上，安装 AS-FFTP01 功能卡时使用。
2. 使用前请先参考 AS 系列模块手册 AS-FFTP01 功能说明或下方范例，于 HWCONFIG 中将 AS-FFTP01 通讯卡的 MQTT 参数设定完成。

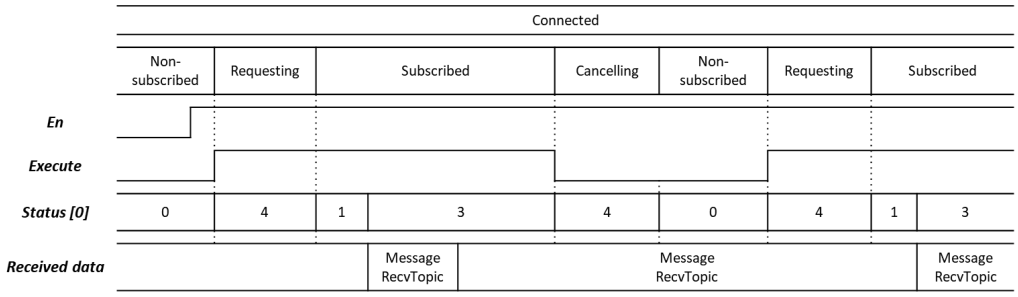
3. 使用 MQTT_Subscribe 功能前，需先使用 MQTT_Connect 成功建立联机，建议使用联机成功状态代码作为 API 的启动判断，详情请参考下方操作范例。

4. 装置名称与说明如下表所示。

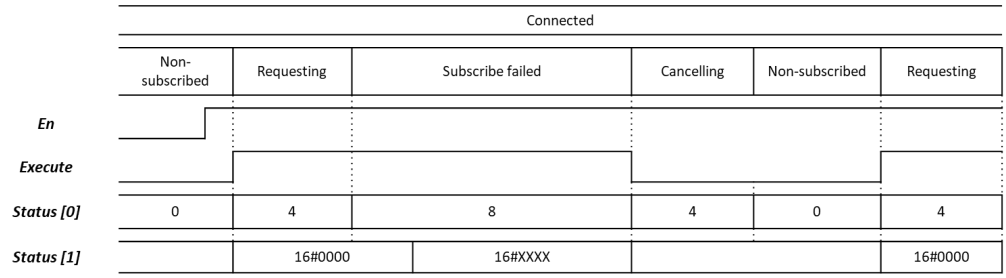
装置	名称	使用说明	数据类型	备注
Execute	执行	上升沿触发执行 1 次订阅 下降沿触发执行 1 次解除订阅	BOOL	
SID	插槽编号	固定输入 34，代表 AS-FFTP01 功能卡	WORD	
BrokerNum	服务器（Broker）编号	固定输入 1	WORD	
Topic	主题	• 最大支持长度为 100 个字符，结尾需包含字符串终止符（16#00）。 • 主题内容不可为空。	STRING[100]	
Qos	服务质量（Quality of Service）	信息的服务质量 允许范围设定值：0、1	WORD	请参考 API2215 MQTT_Publish 说明中的服务质量功能说明
Status	状态	• Status[0] 状态代码 0：解除订阅完成 1：订阅完成 3：收到新的发布信息 4：订阅/解除订阅进行中 8：发生错误，将产生错误代码 • Status[1] 错误代码，请参考 API 2214 MQTT_Connect 说明内容	WORD[2]	连续占用 2 个 word，更新时机请参考下方时序图说明。
Message	信息	• Message[0] 接收到的信息长度 • Message[1] 接收到的信息内容，最大支持 256 字符，数据以先上 8 位，再下 8 位顺序摆放，最后面会包含终止符（16#00）	WORD[129]	
RecvTopic	接收到的主题	同时订阅多个主题时，会需要辨识信息来源主题。 • RcvTopic[0]信息来源主题的长度 • RcvTopic[1] 接收到的主题内容，最大支持 100 字符，数据以先上 8 位，再下 8 位顺序摆放，最后面会包含终止符（16#00）	WORD[51]	

5. 时序图

(1) 正常订阅、接收与正常解除订阅



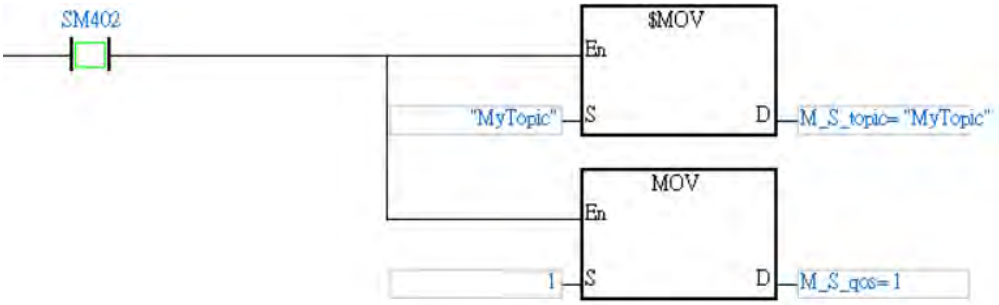
(2) 订阅发生错误



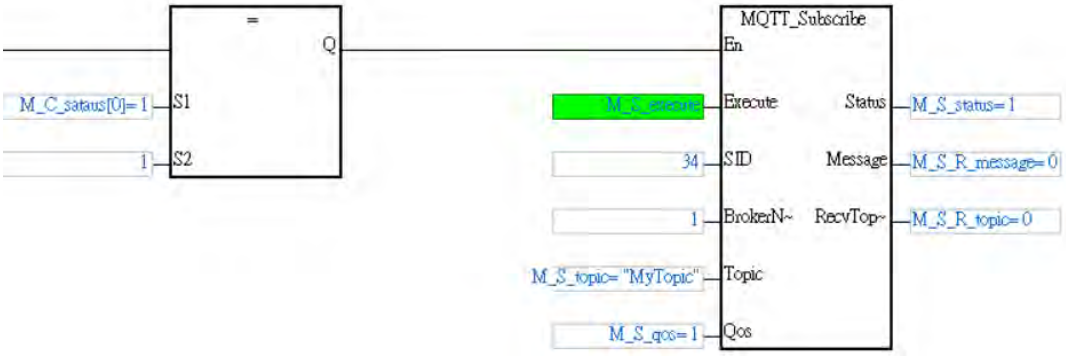
程序范例：

1. 参考 MQTT_Connect 说明，建立联机。
2. 参数说明

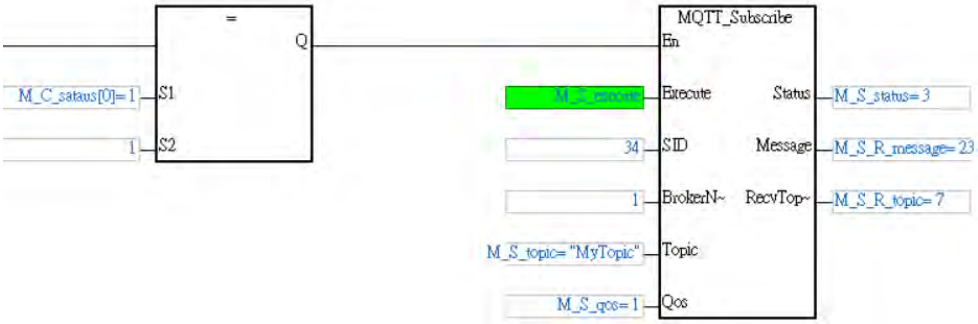
参数	内容	说明
SID	34	固定参数
BrokerNum	1	
Topic	“MyTopic”	订阅主题”MyTopic”
Qos	1	确保至少会收到一次信息



3. 触发执行：使用 MQTT_Connect 的 Status[0] = 1 作为条件连接 En，确保执行订阅时联机是正常的。设定 Execute = ON 后，当 Status[0] = 1 时，表示订阅成功。



4. 接收数据：当发布者传送信息到主题『MyTopic』后（参考 API2215 MQTT_Publish 范例），AS-FFTP01 功能卡就会接收到信息『"Hello from AS-FFTP01."』，主题与信息的数据长度以 byte 为单位放在 Message[0] 与 RecvTopic[0]，主题名称与信息内容则从 Message[1]与 RecvTopic[1]开始依序放置。



MQTT	M_S_R_topic[0]		WORD	7	Signed Decimal
MQTT	M_S_R_topic[1]		WORD	My	ASCII
MQTT	M_S_R_topic[2]		WORD	To	ASCII
MQTT	M_S_R_topic[3]		WORD	pi	ASCII
MQTT	M_S_R_topic[4]		WORD	c*	ASCII
MQTT	M_S_R_topic[5]		WORD	**	ASCII
MQTT	M_S_R_message[0]		WORD	23	Signed Decimal
MQTT	M_S_R_message[1]		WORD	"H	ASCII
MQTT	M_S_R_message[2]		WORD	el	ASCII
MQTT	M_S_R_message[3]		WORD	lo	ASCII
MQTT	M_S_R_message[4]		WORD	f	ASCII
MQTT	M_S_R_message[5]		WORD	ro	ASCII
MQTT	M_S_R_message[6]		WORD	m	ASCII
MQTT	M_S_R_message[7]		WORD	AS	ASCII
MQTT	M_S_R_message[8]		WORD	-F	ASCII
MQTT	M_S_R_message[9]		WORD	FT	ASCII
MQTT	M_S_R_message[10]		WORD	P0	ASCII
MQTT	M_S_R_message[11]		WORD	1.	ASCII
MQTT	M_S_R_message[12]		WORD	"*	ASCII

5. 解除订阅：设定 Execute = OFF，便会针对 Topic 的主题送出解除订阅封包给服务器，当 Status[0] = 0 时表示解除订阅完成。
6. 若未执行解除订阅，而发生了断线或是 PLC 曾进入 STOP 状态，只要 PLC 恢复运行，且重新联机时设定 CleanSession = OFF，就仍会收到先前订阅主题的信息，不需重新订阅。
7. 承上，若此时重新执行订阅，则会收到错误代码 16#8011，此主题已订阅。
8. 订阅 QoS 等级为 1，若要确保在发生断线后，断线期间的信息至少会收到一次，需要在重新联机时保持会话（CleanSession = OFF），且发布信息的 QoS 等级也为 1，信息才会被重新传送。

6.23 储存卡/文件寄存器读写指令

6.23.1 储存卡/文件寄存器读写指令一览表

API	指令码（位）		P 指令	功能
	16	32		
2300	MWRIT	–	✓	将 PLC 数据数据写入到储存卡里
2301	MREAD	–	✓	将数据从储存卡里读取到 PLC
2302	MTWRIT	–	✓	将文字写入到储存卡里
2303	MEMW	–	✓	文件寄存器 FR 写入
2304	MDEL	–	✓	储存卡文件删除
2305	Record_Trigger	–	–	将数据记录并写入文件

6.23.2 储存卡/文件寄存器读写指令说明

API	指令码		操作数	功能
2300	MWRIT	P	**C**、**S**、**S_1**、**S_2**、**S_3**、**S_4**	将 PLC 数据写入到储存卡里

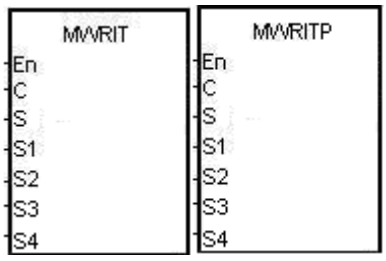
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
C								●	●				○	○		
S								●	●							
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●							
S_4								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
C		●											
S		●											
S_1			●										
S_2		●											
S_3													●
S_4			●										

脉冲执行型	16 位指令	32 位指令
AS	AS	-

6

符号：



C ：控制参数

S ：数据源

S_1 ：欲写入的数据长度

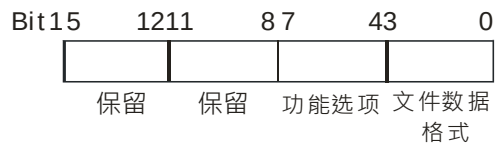
S_2 ：换行

S_3 ：文件名

S_4 ：文件内的数据地址

指令说明：

1. 操作数说明：
   - 控制参数 **C**：



说明项目	代码	说明
文件数据格式	0	二进制。默认值。
		副文件名为.dmd。
		每个数据的单位为 Word。
	1	每个数据之间用逗点做为区隔。
		每个数据的单位为 Word。
		副文件名为.csv。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
文件数据格式	2	每个数据之间用逗点做为区隔。
		每个数据的单位为 DWord。
		副文件名为.csv。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	3	每个数据之间用 TAB 做为区隔。
		每个数据的单位为 Word。
		副文件名为.txt。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	4	每个数据之间用 TAB 做为区隔。
		每个数据的单位为 DWord。
		副文件名为.txt。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	5	数据与数据之间没有区隔符号。
		每个数据的单位为 Word。
		副文件名为.txt。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	6	数据与数据之间没有区隔符号。
		每个数据的单位为 DWord。
		副文件名为.txt。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	7	每个数据之间用逗点做为区隔。(此格式 V1.04.00 版以上支持)

说明项目	代码	说明
		每个数据的单位为 Word。
		副文件名为.csv。
		采用 ASCII 的字符。
		保存的数据采用 10 进制来表示。
功能选项	0	附加（Append）。将用户欲写入的数据，从文件的最后一个数据开始写入。默认值。
		若文件不存在，则会自动产生。
	1	覆写（overwrite）。将用户欲写入的数据，从指定的文件的数据地址开始写入。
		若文件不存在，则会自动产生。
保留	-	固定为 0。

- 数据源 **S**：数据源的起始地址。
- 欲写入的数据长度 **S_1**：用来设定欲写入文件的数据数据长度，0 表示不写入。

项目	说明
计算单位	计算的单位需依照操作数 **C** 的文件数据格式来做为辨别，若文件数据格式为 0、1、3、5 或 7（请参考操作数 **C**），则计算单位为 Word；若文件数据格式为 2、4 或 6（请参考操作数 **C**），则计算单位为 DWord。
参数单位	DWord。
数据数据长度	设定长度不可以超出该装置范围（有关装置范围请参考 Ch2 装置说明），文件写入的数据量不可以超出 4GB。

- 换行 **S_2**：可用来设定写入多少数据（Word/DWord）后，换到下一行，可设定范围为 0~256。（0 表示不换行）
- 韧体版本 V1.04.00 版（含）以上，文件名 **S_3** 支持可直接输入字符串，但不可超过九个字。
- 文件名 **S_3~S_3**+4 :**S_3** 占用连续 5 个装置来指定写入数据的文件名，且文件名不可以超过九个字符（包含终止位 16#00），如果没有填入字符串终止符会有错误。当有读取到终止位时，将会停止读取字符，且检查文件名是否合法。合法命名的字符为 A~Z、a~z 与 0~9。副文件名会依据 **C** 的文件数据格式，而自动给于副文件名，生成的文件会在默认文件夹里。若文件名为“Test1”，则写入装置样式如下：

S3	'e'	'T'
S3 +1	't'	's'
S3+2	NUL	'1'

ASCII代码 ⇨

S3	16#65	16#54
S3 +1	16#74	16#73
S3+2	16#00	16#31

- 默认文件夹路径：

机种名称	文件夹路径
AS300 Series	PLC CARD\AS300\UserProg
AS200 Series	PLC CARD\AS200\UserProg

- 韧体版本 V1.08.30 版（含）以上，文件名 S3 支持最大长度为 200 个字符（包含终止位 16#00）。

- 文件内的数据地址 **S_4**：用来指定写入文件的起始地址。

说明项目	说明
计算单位	计算的单位需依照操作数 C 的文件数据格式来做为辨别，若文件数据格式为 0、1、3、5 或 7，则计算单位为 Word；若文件数据格式为 2、4 或 6，则计算单位为 DWord。(请参考操作数 C)
S_4 的参数单位	此参数单位为 Dword。
S_4 使用方式	若 C 的功能项目为 0（附加），则此操作数不会被使用到。
	若 C 的功能项目为 1（覆写），则会将数据写入到 S4 指定在文件里的数据地址。
	此 S4 指所定的数据地址不可以超出文件所拥有的数据总个数。当 S4 为 0 时，表示为第一个位置。

2. 相关标志

标志	说明
SM450	若储存卡有存在于 CPU 模块上，则此标志会为 ON 的状态。
SM451	储存卡在写保护的模式里
SM452	储存卡正在读/写的动作
SM453	当储存卡有发生错误的时候，此 SM 会设定为 ON。当此 SM 为 ON 时，需自行设定为 OFF。错误代码请参考 SR453。

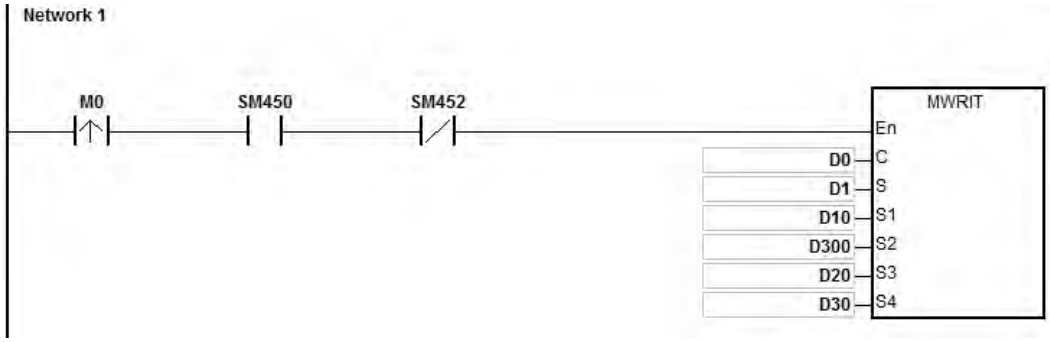
相关错误代码（SR453）

错误代码	说明
16#005E	当 CPU 模块在进行储存卡初始化流程时，发生错误而无法进行初始化流程。
16#005F	路径错误或文件不存在。
16#0060	无法产生默认文件夹。
16#0061	储存卡内存空间不足。
16#0062	储存卡为写保护。
16#0063	当数据写入到文件时，发生错误。
16#0064	当要读取文件数据时，无法读取文件的数据。
16#0065	文件属性为只读状态。

3. 当写入文件数据格式为 0 时，则读取文件数据格式也一定要为 0，不然无法读取，且 SM453 会 ON 起来。其它文件数据格式依此类推。

程序范例：

当储存卡有插入 CPU 模块的时候，SM450 为 ON 的状态；当 MWRIT 开始运作的时候，SM452 为 ON；当 MWRIT 完成写入动作的时候，SM452 为 OFF。可使用脉冲指令 MWRITP，避免指令持续执行，持续写入储存卡，导致写入次数超过储存卡限制而储存卡故障。



操作数	设定值	说明
D0	16#0011	写入文件。 文件数据格式：每个数据之间有用逗点来作为区隔。每个数据的单位为 Word。副文件名为.csv。采用 ASCII 的字符。
D1	-	写入数据内容。
D10、D11	16#00000030	总共写入长度为 48 个 Word。
D300	16#000A	每写入 10 个 Word 换下一行。
D20	D20=16#6554 D21=16#7473 D22=16#0031	写入文件名为“Test1” 。
D30、D31	16#00000000	从文件最初的地址开始写入。

补充说明：

1. **C** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
2. **S_1** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
3. **S_2** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
4. **S_3** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。

API	指令码			操作数	功能
2301		MREAD	P	**C**，**S**，**S_1**，**S_2**，**S_3**，**D**	将数据从储存卡里读取到 PLC

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
C								●	●				○	○		
S								●	●						○	
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
D								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
C		●											
S													●
S_1			●										
S_2		●											
S_3			●										
D		●											

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：

C ： 控制参数

S ： 文件名

S_1 ： 文件的数据地址

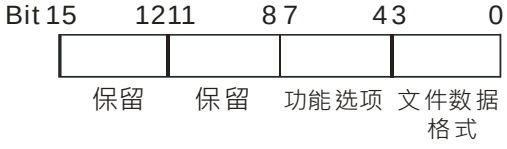
S_2 ： 换行

S_3 ： 欲读取的数据长度

D ： 数据目的地

指令说明：

1. 操作数说明：
   - 控制参数 **C**：



说明项目	代码	说明
文件数据格式	0	二进制（0 为默认值）。
		副文件名为.dmd。
		每个数据的单位为 Word。
	1	每个数据之间用逗点做为区隔。
		每个数据的单位为 Word。
		副文件名为.csv。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
文件数据格式	2	每个数据之间用逗点做为区隔。
		每个数据的单位为 DWord。
		副文件名为.csv。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	3	每个数据之间用 TAB 做为区隔。
		每个数据的单位为 Word。
		副文件名为.txt。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	4	每个数据之间用 TAB 做为区隔。
		每个数据的单位为 DWord。
		副文件名为.txt。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	5	数据与数据之间没有区隔符号。
		每个数据的单位为 Word。
		副文件名为.txt。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	6	数据与数据之间没有区隔符号。
		每个数据的单位为 DWord。
		副文件名为.txt。
		采用 ASCII 的字符。
		保存的数据采用 16 进制来表示。
	7	每个数据之间用逗点做为区隔。(此格式 V1.04.00 版以上支持)
		每个数据的单位为 Word。
		副文件名为.csv。
		采用 ASCII 的字符。

说明项目	代码	说明
		保存的数据采用 10 进制来表示。
功能选项	0	读取数据。(0 为默认值) 读取指定文件内的数据资料。
	1	读取文件内数据长度。 计算文件内有多少个 Words/DWords，将此计算结果会写入 D 与 D+1 里。 当选择此功能代码时，此指令不会将储存卡内的数据读回。 若在操作数 C 中的文件数据格式为 0、1、3、5，则计算单位为 Word； 若文件数据格式为 2、4、6，则计算单位为 DWord。
	其它	以 0 代码执行。
保留	-	固定为 0

- 韧体版本 V1.04.00 版（含）以上，文件名 **S** 支持可直接输入字符串，但不可超过九个字。
- 文件名 **S~S**+4：**S** 占用连续 5 个装置来指定写入数据的文件名，且文件名不可以超过九个字符（包含终止位 16#00），如果没有填入字符串终止符会有错误。当有读取到终止位时，将会停止读取字符，且检查文件名是否合法。合法命名的字符为 A~Z、a~z 与 0~9。副文件名会依据 **C** 的文件数据格式，而自动给于副文件名，生成的文件会在默认文件夹里。若文件名为“Test1”，则写入装置样式如下：

S	'e'	'T'	ASCII代码	S	16#65	16#54
S +1	't'	's'	⇨	S +1	16#74	16#73
S+2	NUL	'1'		S+2	16#00	16#31

- 默认文件夹路径：

机种名称	文件夹路径
AS300 Series	PLC CARD\AS300\UserProg
AS200 Series	PLC CARD\AS200\UserProg

- 韧体版本 V1.08.30 版（含）以上，文件名 S3 支持最大长度为 200 个字符（包含终止位 16#00）。
- 文件的数据地址 **S_1**：用来指定读取文件的起始地址。

项目	说明
计算单位	计算的单位需依照操作数 **C** 的文件数据格式来做为辨别，若文件数据格式为 0、1、3、5 或 7，则计算单位为 Word；若文件数据格式为 2、4 或 6，则计算单位为 DWord。(请参考操作数 **C**)
参数单位	此参数单位为 DWord。
使用方式	所设定的起始数据地址不可以超过文件内所拥有的数据总个数。当 **S_1** 为 0 时，表示为第一个位置。

- 欲读取的数据长度 **S_3**：设定的数据数据长度不可以超出装置最大范围。若超出文件所记载的数据（Word/DWord）个数，读取的数据个数会以文件记载的为主，0 表示不读取，单位为 DWord。
- 数据目的地 **D**：数据（Word/DWord）储存的起始装置地址。

2. 相关标志

标志	说明
SM450	若储存卡有存在于 CPU 模块上，则此标志会为 ON 的状态。
SM451	储存卡在写保护的模式里。
SM452	储存卡正在读/写的动作。
SM453	当储存卡有发生错误的时候，此 SM 会设定为 ON。当此 SM 为 ON 时，需自行设定为 OFF。错误代码请参考 SR453。

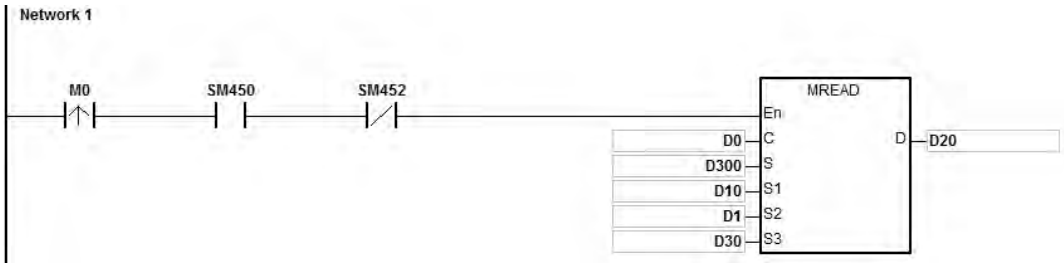
相关错误代码（SR453）:

错误代码	说明
16#005E	当 CPU 模块再进行储存卡初始化流程时，发生错误而无法进行初始化流程。
16#005F	路径错误或文件不存在。
16#0060	无法产生默认文件夹。
16#0061	储存卡内存空间不足。
16#0062	储存卡为写保护。
16#0063	当数据写入到文件时，发生错误。
16#0064	当要读取文件数据时，无法读取文件的数据。

3. 当写入文件数据格式为 0 时，则读取文件数据格式也一定要为 0，不然无法读取，且 SM453 会 ON 起来。其它文件数据格式依此类推。

程序范例：

当储存卡有插入 CPU 模块的时候，SM450 为 ON。当 MREAD 开始运作的时候，SM452 为 ON 的状态，当 MREAD 完成读取动作的时候，SM452 为 OFF。



1. 当 M0 为 ON，PLC 会读取储存卡文件数据保存至 D20。

操作数	设定值	说明
D0	16#0001	读取储存卡文件数据
		文件数据格式：每个数据之间有用逗点来作为区隔。每个数据的单位为 Word。副文件名为.csv。采用 ASCII 的字符。
D300	D300=16#6554 D301=16#7473 D302=16#0031	读取文件名为“Test1”
D10、D11	16#00000000	从文件最初的地址开始读取
D1	16#000A	每读取 10 个 Word 换下一行

操作数	设定值	说明
D30、D31	16#00000020	总共读取长度为 32 个 Word
D20	-	将读取的数据储存至 D20

2. 当 D0=16#0011，M0 为 ON，PLC 会计算文件内有多少个 Words/Dwords，然后保存至 D20。

操作数	设定值	说明
D0	16#0011	计算文件内有多少个 Words/Dwords，将此计算结果会写入 D 与 D+1 里。
D20	-	将读取的数据保存至 D20

补充说明：

1. **C** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
2. **S_2** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
3. **S_3** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
4. **D** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。

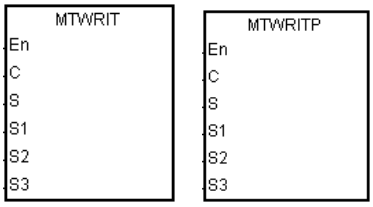
API	指令码			操作数	功能
2302		MTWRIT	P	**C**，**S**，**S_1**，**S_2**，**S_3**	将文字写入到储存卡里

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
C								●	●				○	○		
S								●	●							
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●						○	

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
C		●											
S		●											
S_1		●											
S_2		●											
S_3													●

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



C ： 控制参数
S ： 数据源
S_1 ： 数据长度
S_2 ： 分隔符
S_3 ： 文件名

指令说明：

1. 操作数说明：

   - 控制参数 **C**：

参数数值	说明
0（append）	若文件存在，将装置的数据写在文件的最后一个 byte。
	若文件不存在，会自动产生出来。
1（overwrite）	若文件存在，则新的数据会覆盖到旧的数据，文件的大小以新的数据为主。
	若文件不存在，会自动产生出来。

   - 数据源 **S**：指定要写入储存卡的装置数据源。若欲写入"12345"的字符串到文本文件里的话，则保存在装置里的数据如下图所示。由于以 byte 做为基本单位，所以保存在 Word 的时候，第一个 byte 会在 Word 的 low byte 里，第二个 byte 会在 Word 的 High byte 里，其于以此类推。最后一个 byte 为终止符号"16#00"，表示写入结束。

S300 byte 2	byte 1	S300+1 byte 4	byte 3	S300+2 byte 6	byte 5
16#32	16#31	16#34	16#33	16#00	16#35

- 数据长度 **S_1**：用来决定写入到储存卡的数据长度，单位为 byte。不能超过指定装置的最大范围。每次写入的长度不可以超过 255 bytes，0 表示不写入。
- 分隔符 **S_2**：每次写完数据到储存卡的时候，会将此符号加入到数据里的最后一个 byte 的下两个 bytes。例：**S_1**=N，**S_2** 分隔符之情况如下方表格，则 **S_2** 分隔符写入到储存卡的情况会如下方表格的说明。

S_2 操作数		**说明**
High byte	**Low byte**	
16#00	16#00 或 非 16#00	写入 N 个 Bytes 到文件里。
非 16#00	16#00	写入 N+1 个 Bytes 到文件里，其中 N+1 为 S_2 操作数的 High Byte。
非 16#00	非 16#00	写入 N+2 个 Bytes 到文件里，其中 N+1 为 S_2 操作数的 High Byte，N+2 为 S_2 操作数的 Low Byte。

- 韧体版本 V1.04.00 版（含）以上，文件名 **S_3** 支持可直接输入字符串，但不可超过九个字。
- 文件名 **S_3~S_3**+4：**S_3** 占用连续 5 个装置来指定写入数据的文件名，且文件名不可以超过九个字符(包含终止位 16#00)，如果没有填入字符串终止符会有错误。当有读取到终止位时，将会停止读取字符，且检查文件名是否合法。合法命名的字符为 A~Z、a~z 与 0~9。副文件名会依据 **C** 的文件数据格式，而自动给于副文件名，生成的文件会在默认文件夹里。若文件名为“Test1”，则写入装置样式如下：

S3	'e'	'T'	ASCII代码	S3	16#65	16#54
S3 +1	't'	's'	⇨	S3 +1	16#74	16#73
S3+2	NUL	'1'		S3+2	16#00	16#31

- 默认文件夹路径：

机种名称	文件夹路径
AS300 Series	PLC CARD\AS300\UserProg
AS200 Series	PLC CARD\AS200\UserProg

- 韧体版本 V1.08.30 版（含）以上，文件名 S3 支持最大长度为 200 个字符（包含终止位 16#00)。

2. 相关标志

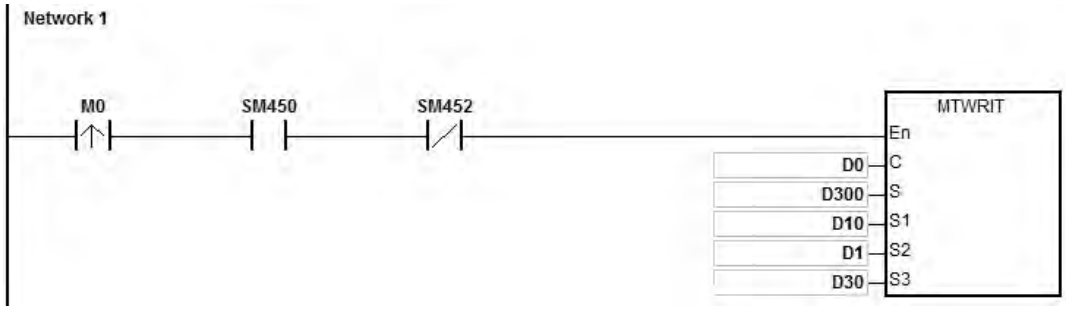
标志	说明
SM450	若储存卡有存在于 CPU 模块上，则此标志会为 ON 的状态。
SM451	储存卡在写保护的模式里
SM452	储存卡正在读/写的动作
SM453	当储存卡有发生错误的时候，此 SM 会设定为 ON。当此 SM 为 ON 时，需自行设定为 OFF。错误代码请参考 SR453。

相关错误代码（SR453）

错误代码	说明
16#005E	当 CPU 模块再进行储存卡初始化流程时，发生错误而无法进行初始化流程。
16#005F	路径错误或文件不存在
16#0060	无法产生默认文件夹
16#0061	储存卡内存空间不足
16#0062	储存卡为写保护
16#0063	当数据写入到文件时，发生错误
16#0064	当要读取文件数据时，无法读取文件的数据
16#0065	文件属性为只读状态

程序范例：

当储存卡有插入 CPU 模块的时候，SM450 会为 ON 的状态，且接点会闭合。当 MTWRIT 开始运作的时候，SM452 会为 ON 的状态，且接点会打开，当 MTWRIT 完成写入动作的时候，SM452 会为 OFF 的状态，且接点会闭合。



操作数	设定值	说明
D0	16#0001	覆写文件 文件数据格式：每个字符的单位为 byte。副文件名为.txt。采用 ASCII 的字符。将 D300 的内容写入到文件里。
D300	-	写入数据内容
D10	16#000A	写入字符串长度为 10 个 bytes。
D1	16#0A00	每次写完数据后，将此符号加入到最后一个 byte
D30	D30=16#6554 D31=16#7473 D32=16#0031	写入的文件名为 Test1

补充说明：

1. **C** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
2. **S_1** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
3. **S_3** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。

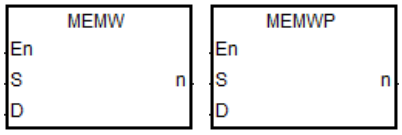
API	指令码		操作数	功能
2303	MEMW	P	**S**・**D**・**n**	文件寄存器 FR 写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●			●	●		●								
D									●							
n	●	●			●	●		●				●	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●											
D		●											
n		●											

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 数据源起始地址

D ： FR 写入起始地址

n ： 写入长度

指令说明：

1. **S** 为数据源之起始地址，若使用自行宣告之变量时，建议使用矩阵型变量。
2. **D** 为写入目的的起始地址，若使用宣告之变量时，请自行寻址使用 FR 组件，并且使用矩阵型变量。
3. **n** 为写入之长度，此长度范围为 1~2048，超出范围时，指令不执行，SM0=ON，SR0 记录错误码 16#200B。
4. 若 S、D 装置超出范围时，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
5. 此指令写入时，将会占用 60~120 毫秒（ms）的运行时间，故建议在控制设备闲置的情况下执行。例如：PLC 暂时无外部中断输入、高速输出或实时事件处理的情况。
6. 写入时间点只有在条件接点 OFF 到 ON 时写入一遍。注意：此文件暂存区只有写入 100,000 次寿命，请小心使用。

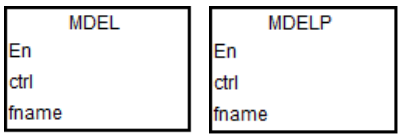
API	指令码			操作数	功能
2304		MDEL	P	**ctrl, fname**	储存卡文件删除

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
ctrl								●					○	○		
fname								●	●						○	

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
ctrl		●				●							
fname													●

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



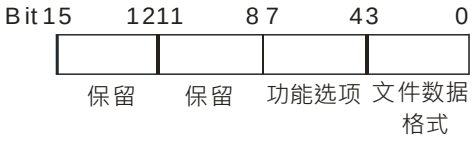
ctrl ：控制参数

fname ：文件名

指令说明：

1. 操作数说明：

- 控制参数 **ctrl**：



说明项目	代码	说明
文件数据格式	0	副文件名为.dmd
	1，2，7	副文件名为.csv
	3，4,，5，6	副文件名为.txt
功能选项，保留	-	固定为 0

- 文件名 **fname** 支持可直接输入字符串，但不可超过九个字。

- 文件名 **fname~fname**+4：**fname** 占用连续 5 个装置来指定写入数据的文件名，且文件名不可以超过九个字符（包含终止位 16#00），如果没有填入字符串终止符会有错误。当有读取到终止位时，将会停止读取字符，且检查文件名是否合法。合法命名的字符为 A~Z、a~z 与 0~9。副文件名会依据 **ctrl** 的文件数据格式，而自动给于副文件名，生成的文件会在默认文件夹里。若文件名为“Test1”，则写入装置样式如下：

S3	'e'	'T'	ASCII代码	S3	16#65	16#54
S3 +1	't'	's'	⇨	S3 +1	16#74	16#73
S3+2	NUL	'1'		S3+2	16#00	16#31

- 默认文件夹路径：

机种名称	文件夹路径
AS300 Series	PLC CARD\AS300\UserProg
AS200 Series	PLC CARD\AS200\UserProg

- 韧体版本 V1.08.30 版（含）以上，文件名 S3 支持最大长度为 200 个字符（包含终止位 16#00）。

2. 相关标志

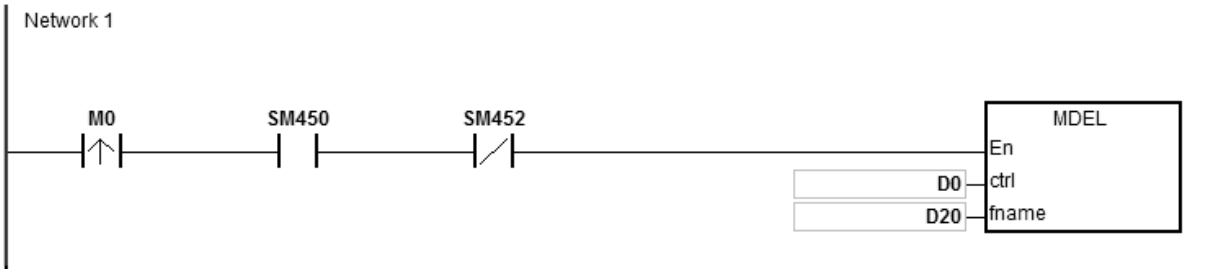
标志	说明
SM450	若储存卡有存在于 CPU 模块上，则此标志会为 ON 的状态。
SM451	储存卡在写保护的模式里
SM452	储存卡正在读/写的动作
SM453	当储存卡有发生错误的时候，此 SM 会设定为 ON。当此 SM 为 ON 时，需自行设定为 OFF。错误代码请参考 SR453。

相关错误代码（SR453）

错误代码	说明
16#005E	当 CPU 模块在进行储存卡初始化流程时，发生错误而无法进行初始化流程。
16#005F	路径错误或文件不存在
16#0060	无法产生默认文件夹
16#0062	储存卡为写保护

程序范例：删除 Test1.csv 文件

当储存卡有插入 CPU 模块的时候，SM450 为 ON 的状态；当 MDEL 开始运作的时候，SM452 为 ON；当 MDEL 完成删除动作的时候，SM452 为 OFF。



操作数	设定值	说明
D0	16#0001	文件数据格式：副文件名为.csv。
D20	D20=16#6554 D21=16#7473 D22=16#0031	写入文件名为“Test1”

补充说明：

1. **ctrl** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。
2. **fname** 内容值超出范围时，视为运算错误，指令不执行，SM0=ON，SR0 记录错误码 16#2003。

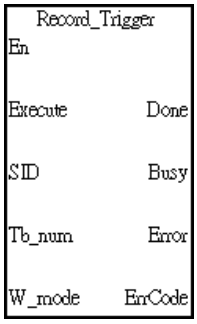
API	指令码	操作数	功能
2305	Record_Trigger	**Execute ~ ErrCode**	将数据记录并写入文件

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Execute	●	●	●	●	●	●	●	●		●	●					
SID	●	●			●	●	●	●	●		●		●			
Tb_num	●	●			●	●	●	●	●		●		●			
W_mode	●	●	●	●	●	●	●	●		●	●					
Done		●	●	●	●	●	●	●		●	●					
Busy		●	●	●	●	●	●	●		●	●					
Error		●	●	●	●	●	●	●		●	●					
ErrCode		●			●	●	●	●			●					

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Execute	●												
SID		●				●							
Tb_num		●				●							
W_mode	●												
Done	●												
Busy	●												
Error	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Execute ~ W_mode ：请参考指令说明

Done ~ ErrCode ：请参考指令说明

指令说明：

1. ISPSoft 支持软件版本 V3.16（含）以上。
2. AS300 主机韧体版本 V1.12（含）以上，安装 AS-FFTP01 功能卡时使用。使用前请先参考 AS 系列模块手册 AS-FFTP01 功能说明或下方范例，于 HWCONFIG 中将 AS-FFTP01 通讯卡的 LOG 参数设定完成。

3. 使用 Record_Trigger 时需先将 HWCONFIG 中的记录模式设定为程控，其搭配参数注意事项如下：

记录模式	记录周期	参数设定
PLC 执行	使用 HWCONFIG 参数	使用 HWCONFIG 参数
永远执行		
程控	由用户自行触发 Execute	「建立新目录时机」、「建立新文件时机」和「保存 SD 卡时机」等参数不支持周期性设定值，如『秒』、『分』、『时』、『日』、『月』等。

4. 装置名称与说明如下表所示。

装置	名称	使用说明	数据类型	备注
Execute	执行标志	触发模块执行命令，上升沿触发执行 1 次记录	BOOL	
SID	插槽编号	固定输入 34，代表 AS-FFTP01 功能卡	WORD	
Tb_num	表格编号	指定执行 HWCONFIG 中设定的表格编号 1~4。	WORD	
W_mode	写入模式	设定写入模式，触发执行标志前需先设定完成： • OFF :触发执行标志后，数据写入缓冲区，待 HWCONFIG 设定的条件达成后，才将数据写入文件。 • ON：触发执行标志后，直接将缓冲区中的数据写入文件。	BOOL	
Done	完成标志	功能块完成标志 • OFF：未触发执行/未完成 • ON：执行完毕且未发生错误	BOOL	
Busy	忙碌标志	功能块忙碌标志 • OFF：非执行中 • ON：执行中	BOOL	
Error	错误标志	功能块错误标志 • OFF：无错误 • ON：发生错误	BOOL	
ErrCode	错误代码	请参考下方错误代码说明。	WORD	

5. 相关错误代码

错误代码	**说明**
16#220B	Tb_num 数值超过范围
16#0001	子目录数量达上限 (500)，无法建立更多子目录
16#0002	SD 卡不存在，请检查并重新插入 SD 卡

错误代码	说明
16#0003	写入文件失败
16#0004	数据记录失败
16#0005	记录模式错误，请于 HWCONFIG 设定记录模式为「程控」
16#0006	参数设定值错误

程序范例：

1. 使用 FTP 客户端软件确认可正确读取 SD 卡，FTP 设定方式参考 AS 模块手册第 10 章 AS-FFTP01 功能说明。
2. 确认 AS300 主机的时区（HWCONFIG > 以太网络进阶设定 > 时区）与万年历（ISPSoft > 工具 > 万年历设定）设置正确。
3. HWCONFIG > AS-FFTP01 参数设定

(1) 于表 1 设定两笔数据，填入名称、寄存器信息与数据类型等参数。字段 1、2 为固定字段，显示日期与时间，自定义字段从编号 3 开始。本范例中分别以整数与浮点数类型记录两个 D 寄存器。

表 1

添加 | 上移 | 下移 | 删除 | 全部删除 | 预览 CSV 文件 | 输入文字以搜索...

栏位编号	栏位名称	寄存器类型	寄存器位址	数据类型	小数点点位数
1	Date				
2	Time				
3	VFD Frequency	D	2103	INT16	0
4	Temperature	D	0	Float	2

(2) 于记录模式处设定程控，可忽视记录周期参数

记录模式

参数名称	设定值	单位	默认值	最小值	最大值
记录模式	程控		程控	-	-
记录周期	100	0.1 秒	100	5	1200

(3) 于文件名中设定父目录名称、子目录名称与文件名。

数据将保存于 SD card/ Delta/ MyLog_日期/ Device_log_日期_时间.csv

文件名称

参数名称	设定值	单位	默认值	最小值	最大值
父目录名称 (SD Card/)	Delta			-	-
子目录名称	MyLog		LOG1	-	-
自动附加子目录名称	日期		关闭	-	-
文件名称 (.csv)	Device_log		log1	-	-
自动附加文件名称	日期 + 时间		关闭	-	-

(4) 于参数设定中设定建立新文件与保存 SD 卡参数。此参数下，单档笔数上限为 10 笔。若数据触发后写入缓冲区 (W_mode = OFF)，则每当记录笔数达 5 笔时，保存一次文件，表示将在记录第 15 笔后自动建立新文件。

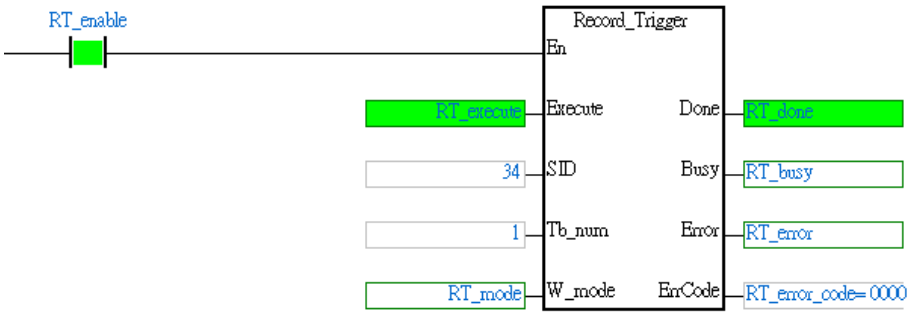
参数设定

参数名称	设定值	单位	默认值	最小值	最大值
建立新目录时机	自动建立新目录		自动建立新目录	-	-
建立新目录参数	1		1	1	31
建立新文件时机	自订单档笔数上区		覆盖文件	-	-
建立新文件参数	10	笔数	100	1	10000
保存 SD 卡时机	笔数		笔数	-	-
保存 SD 卡参数	5	笔数	100	1	500

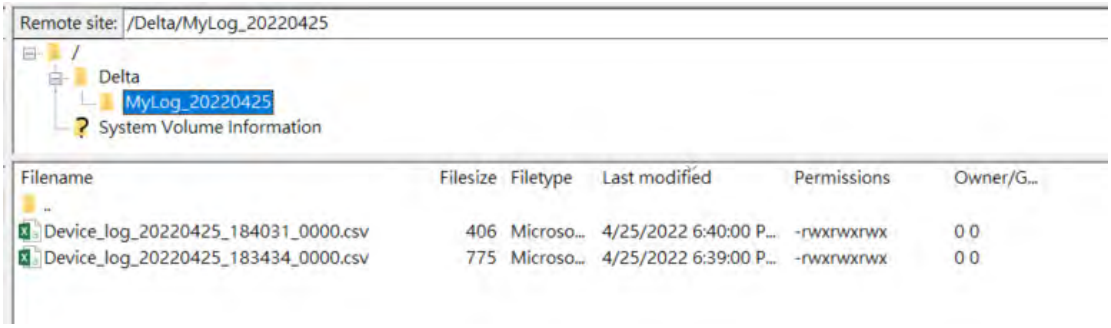
4. 参数说明

参数	内容	说明
SID	34	固定参数
Tb_num	1	执行表 1 记录
W_mode	OFF	

5. 触发执行：将 Execute 设置为 ON，当 Done 为 ON 时，表示数据写入缓冲区成功。欲再次触发前，需先设定 Execute 为 OFF，每次 Execute 从 OFF 到 ON 时，执行一次记录。



6. 本范例手动触发 15 次后，产生 2 份文件，保存为 csv 文件格式。第一份文件中保存 10 笔记录，第二份文件中保存 5 笔记录。透过 FTP 客户端软件可将记录文件传输至计算机。



7. 下图为开启第一份文件后显示的画面，第一列的文字为标题，参考上述步骤 3，于表 1 页面设定。其中 A、B 栏的 Data 与 Time 为固定标题与内容，C、D 栏第 1 列为自定义标题，第 2 列开始记录寄存器数据。

	A	B	C	D	E
1	Date	Time	VFD frequency	Temperature	
2	2022/4/25	18:34:12	5723	20.97	
3	2022/4/25	18:34:16	3128	21	
4	2022/4/25	18:34:28	3631	20.97	
5	2022/4/25	18:34:30	5407	20.97	
6	2022/4/25	18:34:34	4754	20.97	
7	2022/4/25	18:39:24	2220	21.12	
8	2022/4/25	18:39:28	4114	21.16	
9	2022/4/25	18:39:31	5604	21.16	
10	2022/4/25	18:39:33	5061	21.16	
11	2022/4/25	18:39:42	372	21.09	
12					

8. 若设定 W_mode 为 ON，触发 Execute 时，将直接把目前缓冲区中的数据写入文件。可用于重要事件发生且此记录不可遗失时，或是设备异常必须立即保存记录等情况。

   例如：本范例中「保存 SD Card 参数」设定为 5 笔，当触发 Execute 到第 3 次时，共有 3 笔数据存在缓冲区中，尚未存入文件。若于第 4 次触发前设定 W_mode 为 ON，触发 Execute 后就会直接将第 4 次连同前 3 次的记录数据（共 4 笔）一并存入 SD 卡中的记录文件。

6.24 任务控制指令

6.24.1 任务控制指令一览表

API	指令码（位）		P 指令	功能
	16	32		
2400	TKON	–	✓	周期性任务（Cyclic Task）启动
2401	TKOFF	–	✓	周期性任务（Cyclic Task）关闭

6.24.2 任务控制指令说明

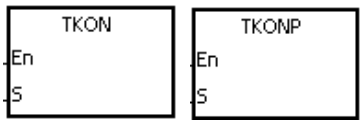
API	指令码		操作数	功能
2400	TKON	P	**S**	周期性任务（Cyclic Task）启动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●						●	●		○	○	○			

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ： 任务编号

指令说明：

1. 启动 **S** 所指定的 Cyclic Task。
2. 当 PLC 由 STOP→RUN 时，会依据 ISPSoft 程序下载时 Cyclic Task 是否启动来执行。
3. 操作数说明：
   - S 的范围为 0~31。
   - Tasks 的建立与启动请参考 ISPSoft 手册。

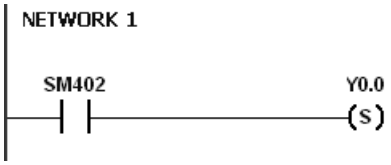
程序范例：

程序写入主机时，只指定启动 Task Cyclic_0 任务，PLC RUN 后，因为程序中指示 TKON 1 表示执行 Task Cyclic_1，所以 Y0.0=ON。

在 ISPSoft 中先建立两个 Cyclic Tasks，只先指定 Cyclic_0 在 PLC RUN 时为 ON，Cyclic_1 先预设为不执行。

Cyclic_0 的 Prog0 中 TKON 1 启动 Cyclic_1

Cyclic_1 的 Prog1 的程序就会被执行



补充说明：

有关任务的说明，请参考 ISPSoft 使用说明。

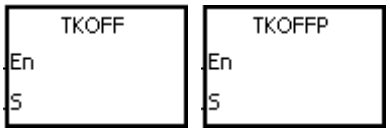
API	指令码			操作数	功能
2401		TKOFF	P	**S**	周期性任务（Cyclic Task）关闭

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S	●	●						●	●		○	○	○			

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

脉冲执行型	16 位指令	32 位指令
AS	AS	-

符号：



S ：任务编号　　Word

指令说明：

1. 关闭 **S** 所指定的 Cyclic Task。
2. 当 PLC 由 STOP→RUN 时，会依据 ISPSoft 程序下载时 Cyclic Task 是否启动来执行。
3. 操作数说明：
   - S 的范围为 0~31。
   - Tasks 的建立与启动请参考 ISPSoft 手册。

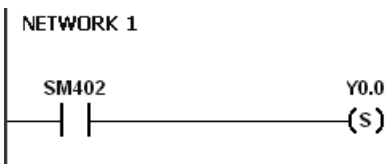
程序范例：

程序写入主机时，就指定启动 Task Cyclic_0 及 Tasks Cyclic_1 任务，PLC RUN 后，因为 TKOFF1 表示关闭 Task Cyclic_1 因此 Y0.0=OFF。

在 ISPSoft 中先建立两个 Cyclic Tasks，先指定 Cyclic_0 和 Cyclic_1 在 PLC RUN 时为执行。

Cyclic_0 的 Prog0 中 TKOFF 1 关闭 Cyclic_1

Cyclic_1 的 Prog1 的程序不会执行



补充说明：

有关任务的说明，请参考 ISPSoft 使用说明。

6.25 SFC 控制指令

6.25.1 SFC 控制指令一览表

API	指令码（位）		P 指令	功能
	16	32		
2500	SFCRUN	–	–	SFC 启动
2501	SFCPSE	–	–	SFC 暂停
2502	SFCSTP	–	–	SFC 停止

6.25.2 SFC 控制指令说明

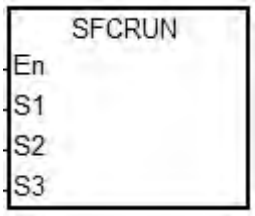
API	指令码	操作数	功能
2500	SFCRUN	S_1 · S_2 · S_3	SFC 启动

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1																
S_2								●					○	○		
S_3																

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													
S_2		●				●							
S_3													

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号:



S_1 ： SFC 名称

S_2 ： 控制参数

S_3 ： 要启动的 STEP 名称

指令说明：

1. 依照 S_2 的设定，启动 S_1 所指定的 SFC 程序。
2. 指令执行时，S_1 所指定的 SFC 程序并不会立即启动，要等到扫描到 SFC 程序时才会启动。
3. 操作数说明：
   - S_1 为 SFC 程序的名称。
   - S_2=0，启动 S_1 指定的 SFC 程序，SFC 中的 SFC/STEP/ACTION/TRANSITION 状态跟参数会被清除，并从初始 STEP 开始执行。
   - S_2=1，启动 S_1 指定的 SFC 程序，SFC 中的 SFC/STEP/ACTION/TRANSITION 状态跟参数会被清除，并从 S_3 指定的 STEP 开始执行。
   - S_2=2，启动 S_1 指定的 SFC 程序，不会清除 SFC 中的 SFC/STEP/ACTION/TRANSITION 状态跟参数，并从原本暂停的位置开始执行。
   - S_3 为 S_1 指定的 SFC 程序中的 STEP 名称。
4. S_2 的范围 0~2，超出范围时，视为 0。
5. 当 SFC 程序状态为 RUN 时，执行此指令无效。

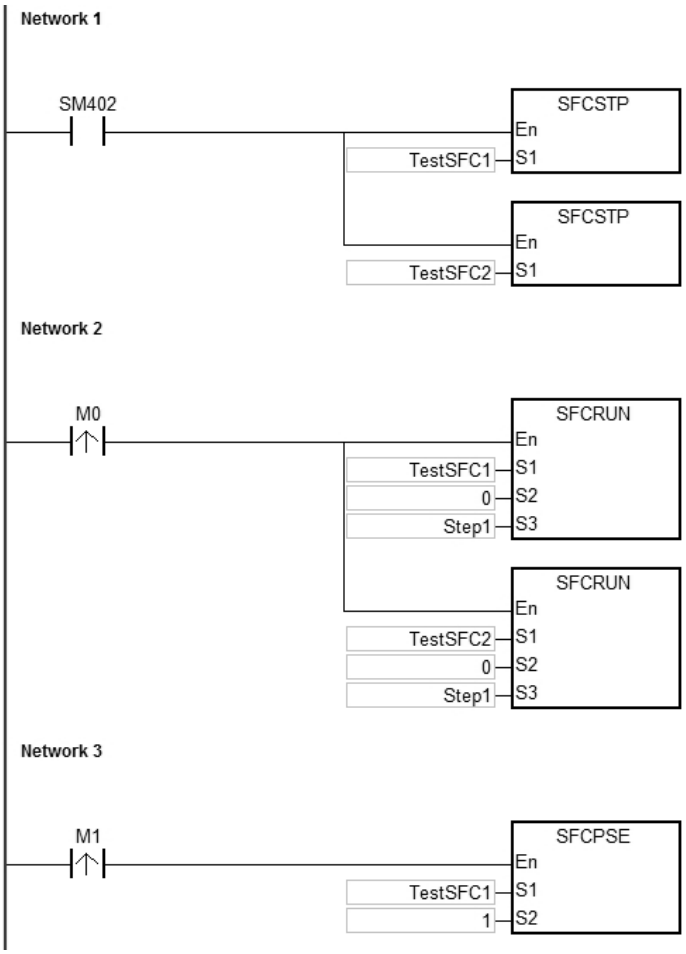
程序范例：

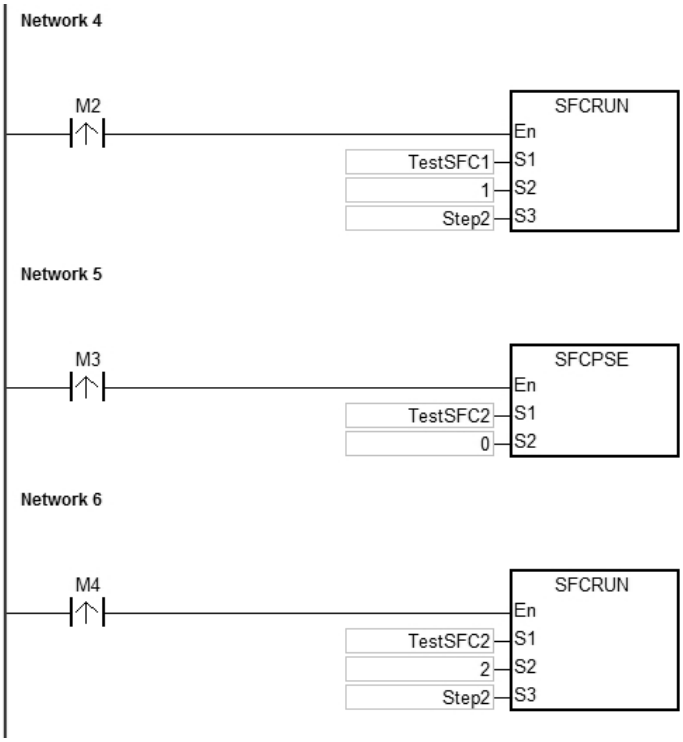
在程序中建立一个 LD 语言 POU 名称为 Main 及两个 SFC 语言 POU 名称为 TestSFC1 与 TestSFC2。

1. 系统执行（RUN）时，TestSFC1 与 TestSFC2 立刻执行 SFCSTP 动作，两个 SFC POU 会停止执行。
2. 当 M0 由 OFF→ON 时，TestSFC1/ TestSFC2 POU 执行动作（SFCRUN）[注一]，执行动作请参考 TestSFC1/ TestSFC2 POU 程序内容。设定 **S_2** 控制参数 0，会清除 SFC 中的状态及参数，并且从 Step1 开始执行。设定 **S_2** 控制参数 1，会清除 SFC 中的状态及参数，并且从 **S_3** 指定的 Step 开始执行。
3. 当 M1 由 OFF→ON 时，TestSFC1 POU 会暂停，控制参数设为 1， SFC 中正在执行中的 ACTION，程序中的输出会被清除（做 Final scan）。
4. 当 M2 由 OFF→ON 时，TestSFC1 POU 执行动作，执行动作请参考 TestSFC1 POU 程序内容。设定控制参数 1，会清除 SFC 中的状态及参数，并且从 Step2 开始执行。
5. 当 M3 由 OFF→ON 时，TestSFC2 POU 会暂停 控制参数设为 0，SFC 中之输出保持不变，不做 Final scan。
6. 当 M4 由 OFF→ON 时，TestSFC2 POU 执行动作，执行动作请参考 TestSFC2 POU 程序内容。设定控制参数 2，会保持 SFC 中的状态及参数，并且从暂停时执行到的 Step 开始执行。

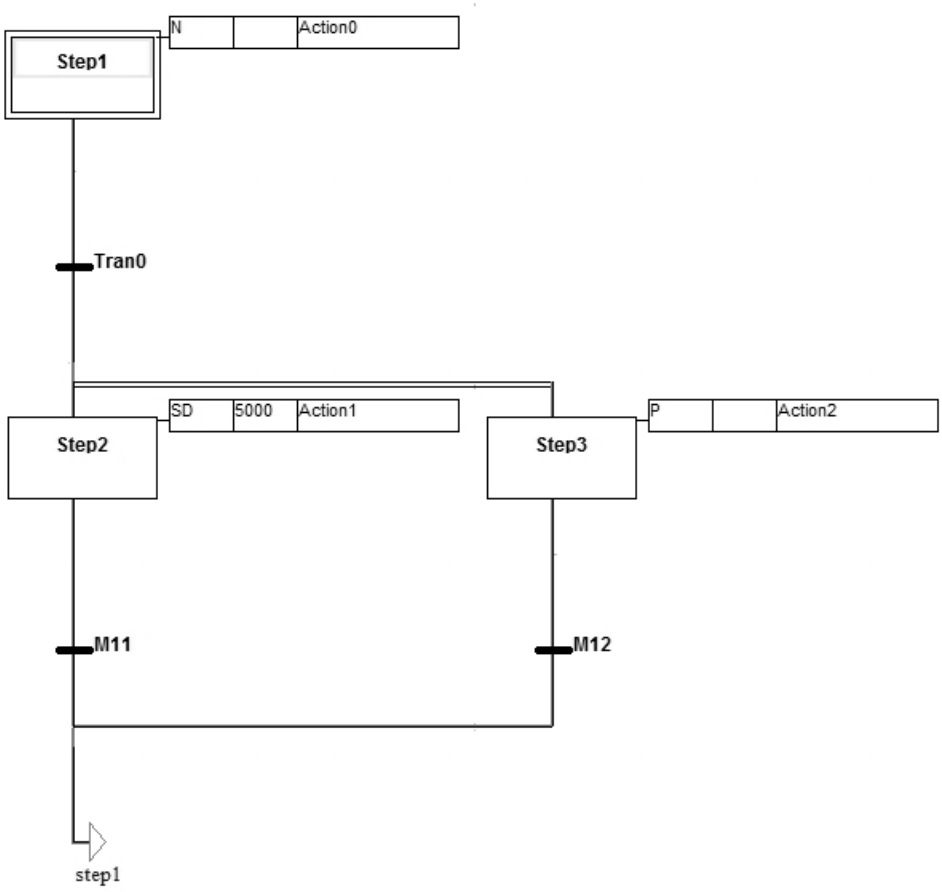
注一：SFCRUN 要等到下一个扫描时间才会启动指定的 SFC POU。

Main POU

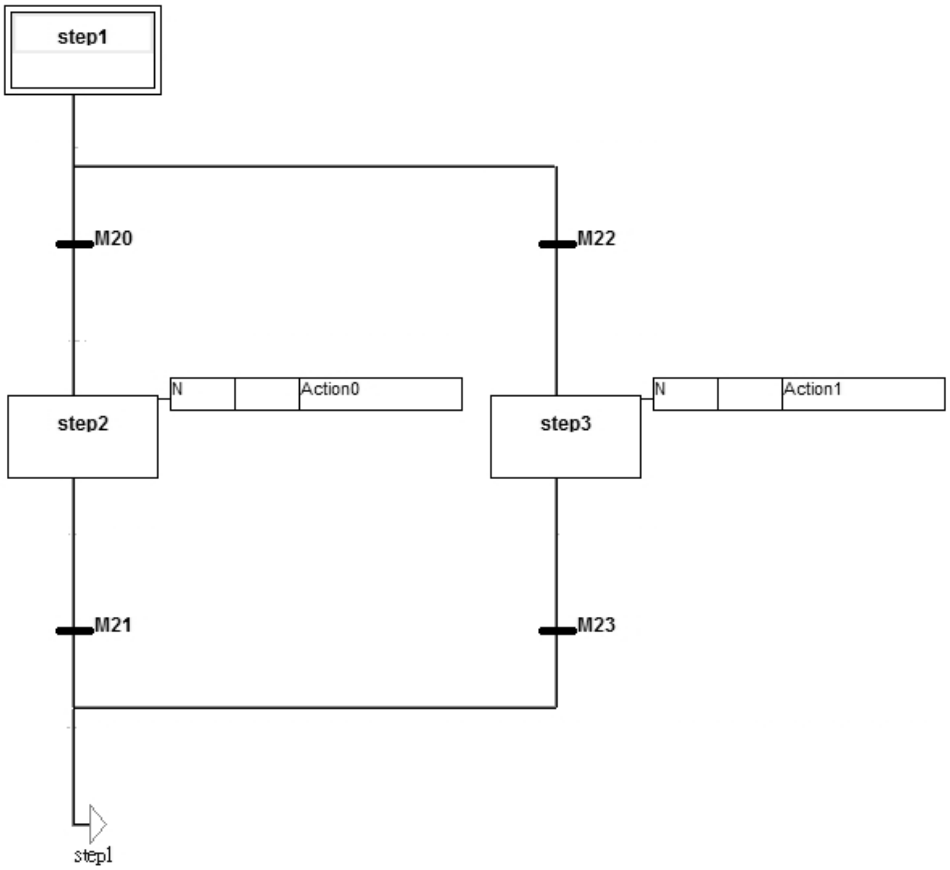




TestSFC1 POU



TestSFC2 POU



补充说明：

有关 SFC 的说明，请参考 ISPSoft 使用说明。

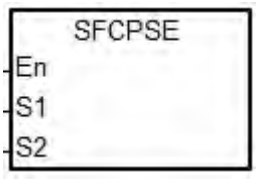
API	指令码	操作数	功能
2501	SFCPSE	S_1 · S_2	SFC 暂停

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1																
S_2								●					○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													
S_2		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： SFC 名称

S_2 ： 控制参数

指令说明：

1. 依照 S_2 的设定，暂停 S_1 所指定的 SFC 程序。
2. 指令执行时，S_1 所指定的 SFC 程序并不会立即暂停，要等到扫描到 SFC 程序时才会暂停。
3. 进入暂停时，SFC 中的 STEP/ACTION/TRANSITON 状态跟参数不变。
4. 操作数说明：
   - S_1 为 SFC 程序的名称。
   - S_2=0 ，暂停 S_1 指定的 SFC 程序 ，SFC 程序中的输出保持不变 ，正在执行中的 ACTION ，不执行 Final Scan。
   - S_2=1 ，暂停 S_1 指定的 SFC 程序 ，正在执行中的 ACTION ，程序中的输出会被清除(执行 Final Scan)。
5. S_2 的范围 0~1，超出范围时，视为 0。
6. 当 SFC 程序状态为 PAUSE/STOP，执行此指令无效。

程序范例：

请参考 SFCRUN 程序范例。

补充说明：

有关 SFC 的说明，请参考 ISPSoft 使用说明。

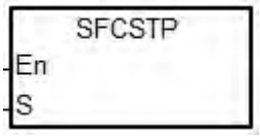
API	指令码	操作数	功能
2502	SFCSTP	**S**	SFC 停止

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S																

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S ：SFC 名称

指令说明：

1. 停止 **S** 所指定的 SFC 程序。
2. 指令执行时，**S** 所指定的 SFC 程序并不会立即停止，要等到扫描到 SFC 程序时才会停止。
3. **S** 指定的 SFC 程序，正在执行中的 ACTION，程序中的输出会被清除（执行 Final Scan）。
4. 当 SFC 程序状态为 STOP，执行此指令无效。

程序范例：

请参考 SFCRUN 程序范例。

补充说明：

有关 SFC 的说明，请参考 ISPSoft 使用说明。

6.26. 高速输出指令

6.26.1 高速输出指令一览表

API	指令码（位）		P 指令	功能	偶數軸#1	奇數軸#2
	16	32				
2700	–	DPLSY	–	高速脉冲输出（无加减速）	V	V
2701	–	DPLSR	–	高速脉冲输出（附加减速）	V	V
2702	PWM	DPWM	–	脉宽调变输出	V	V
2703	JOG	DJOG	–	寸动输出	V	V
2704	–	DZRN	–	原点回归	V	–
2705	–	DPLSV	–	可调变脉冲输出	V	V
2706	–	DDRVI	–	相对地址点对点定位输出	V	V
2707	–	DDRVA	–	绝对地址点对点定位输出	V	V
2708	CSFO	–	–	速度追随输出	V	–
2709	–	DDRVM	–	对标型定位输出	V	–
2710	–	DPPMR	–	相对地址点对点双轴同动输出	V	–
2711	–	DPPMA	–	绝对地址点对点双轴同动输出	V	–
2712	–	DCICR	–	相对位置顺时针圆弧插补	V	–
2713	–	DCICA	–	绝对位置顺时针圆弧插补	V	–
2714	–	DCICCR	–	相对位置逆时针圆弧插补	V	–
2715	–	DCICCA	–	绝对位置逆时针圆弧插补	V	–
2716	–	DCCMR	–	相对位置画圆	V	–
2717	–	DCCMA	–	绝对位置画圆	V	–
2718	TPO	–	–	定位规划表控制输出	V	–
2719	–	DTPWS	✓	定位规划表之单轴输出参数写入	V	–
2720	–	DTPWL	✓	定位规划表之直线插补参数写入	V	–
2721	–	DTPWC	✓	定位规划表之圆弧插补参数写入	V	–
2723		DPPGB	–	点对点来回定位控制	V	–
2724	–	DZRN2	–	原点回归 2（可指定方向输出点）	V	V

注#1：偶数轴表示输出点为 Y0.0、Y0.2、Y0.4、Y0.6、Y0.8、Y0.10

注#2：奇数轴表示输出点为 Y0.1、Y0.3、Y0.5、Y0.7、Y0.9、Y0.11

6.26.2 高速输出指令说明

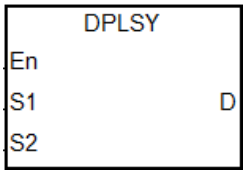
API	指令码			操作数	功能
2700	D	PLSY		**S_1 · S_2 · D**	高速脉冲输出（无加减速）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
D		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
D	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 脉冲输出频率

S_2 ： 脉冲输出个数

D ： 脉冲输出装置

指令说明：

1. **S_1** 指定脉冲输出频率设定范围为 0Hz~200kHz（集电极输出机种）与 4MHz（差动输出机种），基本频率单位为 1Hz，频率误差率范围在 200kHz 约为万分之二，在 100kHz 约为万分之一，依序递减，举例：设定输出 199990Hz 时，其实际输出为 199960Hz；若设定输出 99999Hz，其实际输出为 99990Hz。当输出频率超出限定范围时，PLC 自动以最大或最小频率输出。
2. **S_1** 指定输出频率可在启动执行中变更频率，但不执行任何加减速过程。每次变换频率时机为扫描到执行终指令后，且正在输出中的一个完整脉冲输出完成时。
3. **S_2** 指定脉冲输出个数，可指定范围为 0~2,147,483,647 个。当指定脉冲输出数目设定为 0 时，为不限制脉冲个数，可连续一直输出，直到指令被关闭；当指定脉冲输出数目设定小于 0 时，脉冲不输出。
4. **S_2** 指定脉冲输出个数在启动输出之后，即不可再变更输出个数。
5. **D** 仅限指定 Y0.0~Y0.11 输出，当指令启动后，其输出功能变为高速输出，一般基本指令输出点控制将无效。建议程序使用高速功能之后，就不再使用为一般输出功能。
6. 脉冲输出的 Duty-OFF Time 跟 Duty-ON Time 比例为 1：1。
7. 此指令不限制使用次数，也不限制重复指定相同输出点，但限制每一次扫描周期里，每一个输出点只能被一个高速输出指令启动输出，且依先启动者先占用为原则。
8. 每个输出点之高速输出指令被启动后，皆必须等待该启动之高速输出指令被关闭后，才能让其它指定相同输出的指令被启动。

9. 当高速输出指令被启动于中断程序里，或者未被设计于主流程时，则建议搭配输出完成自动复位功能，由 PLC 于 END 指令里进行输出状态更新。
10. 暂停标志设定后，需等待启动指令再次被执行，并且输出完整一个脉冲之后，PLC 才会暂停输出，并且清除输出中标志。当暂停标志清除并且还有前次剩余个数未输出完毕，则 PLC 将会继续执行。此暂停标志皆由设计者自行设定与清除。
11. 每一个高速输出点搭配之 SM/SR 组件如下表所列。

输出点编号	属性#2	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5
输出中标志	R	SM460	SM472	SM480	SM492	SM500	SM512
完成标志#3	R/W	SM461	SM473	SM481	SM493	SM501	SM513
暂停标志	R/W	SM463	SM474	SM483	SM494	SM503	SM514
输出完成自动复位#4	R/W	SM470	SM475	SM490	SM495	SM510	SM515
当前输出位置#1（32 位）	R/W	SR460 SR461	SR474 SR475	SR480 SR481	SR494 SR495	SR500 SR501	SR514 SR515
输出点编号	属性#2	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
输出中标志	R	SM520	SM532	SM540	SM552	SM560	SM572
完成标志#3	R/W	SM521	SM533	SM541	SM553	SM561	SM573
暂停标志	R/W	SM523	SM534	SM543	SM554	SM563	SM574
输出完成自动复位#4	R/W	SM530	SM535	SM550	SM555	SM570	SM575
当前输出位置#1（32 位）	R/W	SR520 SR521	SR534 SR535	SR540 SR541	SR554 SR555	SR560 SR561	SR574 SR575

注#1：所有当前输出位置皆为停电保持。

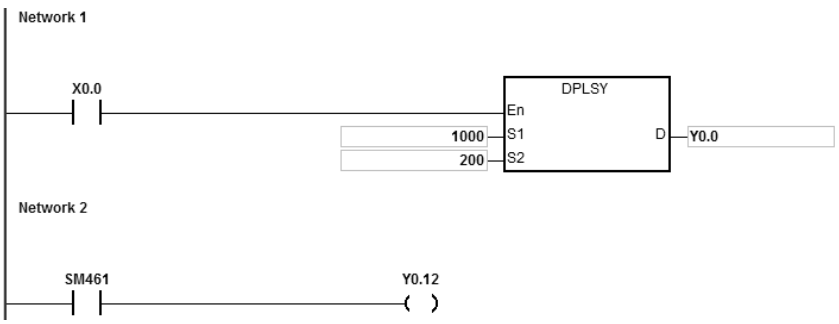
注#2：R 表示只读但不可修改，R/W 表示可读可写。

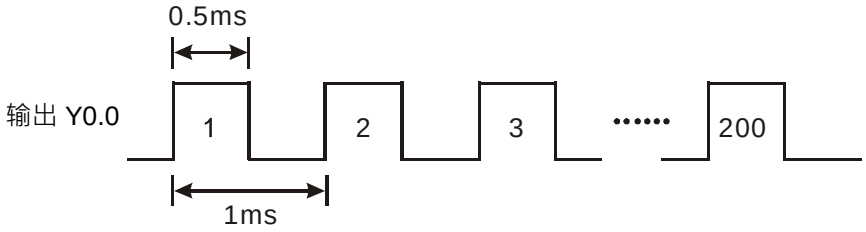
注#3：完成标志建议由设计者自行清除，若完成标志未被清除，则下一次高速输出指令启动时，将会自动被清除。

注#4：自动复位标志由设计者设定，当输出完成后，PLC 将自动清除此标志。

程序范例一：

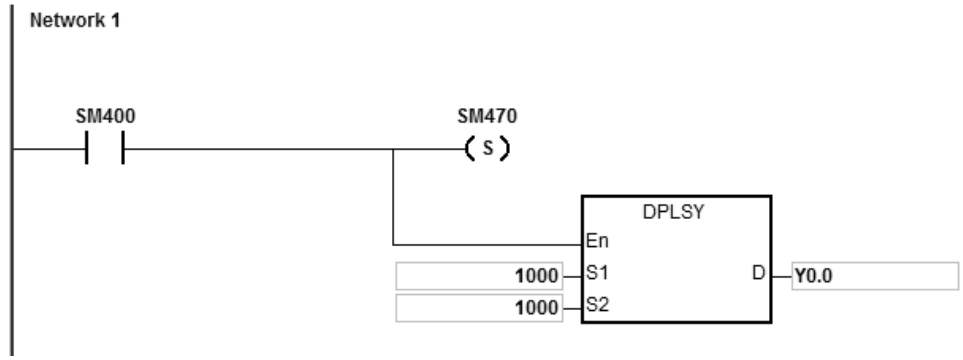
1. 当 X0.0=ON 时，Y0.0 输出 200 个 1kHz 频率脉冲，脉冲输出完毕 SM461=ON，触发 Y0.12=ON。
2. 当 X0.0=OFF 时，脉冲输出 Y0.0 立即停止，当 X0.0 再度 ON 时，脉冲输出会重新开始。



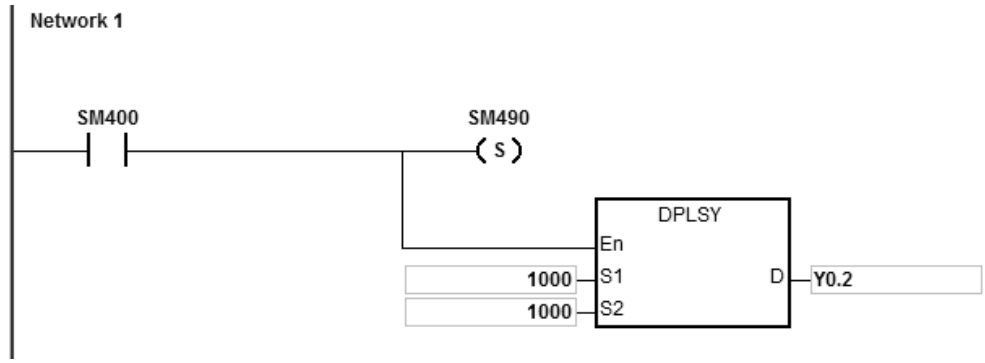


程序范例二：

外部中断 X0.0



外部中断 X0.1



说明：

1. 外部输入中断 X0.0 每次触发一次，Y0.0 就送出 1000 个脉冲；外部输入中断 X0.1 每次触发一次，Y0.2 就送出 1000 个脉冲。
2. 当中断外部输入 X 触发 Y 脉冲输出时，当次 Y 脉冲输出结束与下一次外部输入中断 X 触发的时间，至少必须间隔一次以上的 PLC 扫描时间。

程序范例三（ST 结构化语言 DPLSY 编辑说明）：

```
0001 IF M0 THEN
0002    DPLSY(1000,1000,Y0.0);
0003    M1 := TRUE;
0004 ELSIF M1 AND SM461 THEN
0005    SM470 := TRUE;
0006    M1 := FALSE;
0007 END_IF;
```

动作说明：

1. 当 M0=ON 时，Y0.0 输出 1000 个 1kHz 频率脉冲并且设定 M1=ON，用来记录 Y0.0 高速控制权已经被使用。
2. 当脉冲输出完毕 SM461=ON 并且 M1=ON 时，触发 Y0.0 的自动复位标志 SM470=ON，通知 PLC 释放高速输出控制权。
3. 当 M0 再次由 OFF>ON 时，脉冲输出即会重新启动。
4. 以上程序若有 2 个高速输出程序以上时，请勿重复使用 M1 这个临时性的变量名称。
5. 若输出过程中有可能遇到要求暂停输出的动作，则不建议采用 ST 语言编辑。

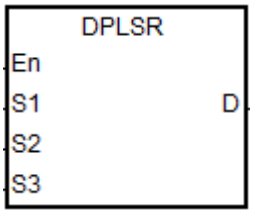
API	指令码			操作数	功能
2701	D	PLSR		**S_1 · S_2 · S_3 · D**	高速脉冲输出（附加减速）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
D		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
D	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 输出目标频率

S_2 ： 脉冲输出个数

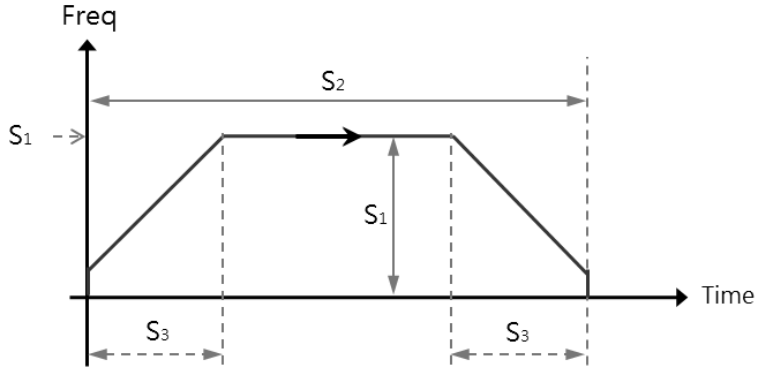
S_3 ： 加减速时间

D ： 脉冲输出装置

指令说明：

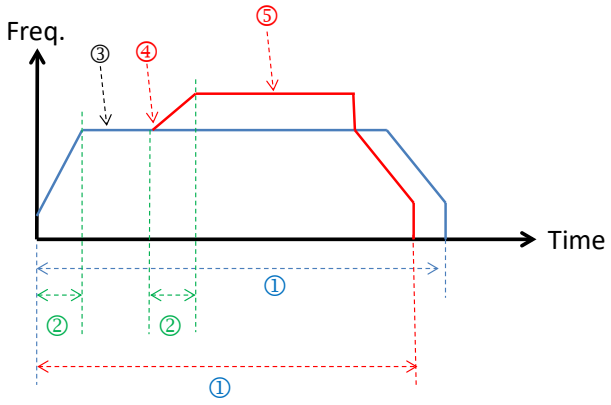
1. **S_1** 指定脉冲输出频率设定范围为 0Hz~200kHz（集电极输出机种）与 4MHz（差动输出机种），基本频率单位为 1Hz，频率误差率在 200kHz 约为万分之二，在 100kHz 约为万分之一，依序递减，举例：设定输出 199990Hz 时，其实际输出为 199960Hz；若设定输出 99999Hz，其实际输出为 99990Hz。当输出频率超出限定范围时，PLC 自动以最大或最小频率输出。
2. **S_1** 指定输出目标频率在指令启动后，可再进行变更目标频率，并且依据 **S_3** 指定时间执行加减速过程。每次变换频率时机为扫描到执行中指令时，且正在输出中的一个完整脉冲输出完成后。
3. **S_2** 指定脉冲输出个数，可指定范围为 0~2,147,483,647 个。当指定脉冲输出数目设定为 0 时，为不限制脉冲个数，可连续一直输出，直到指令被关闭。
4. **S_2** 指定脉冲输出个数在启动输出之后，即不可再变更输出个数。
5. **S_3** 指定加减速时间，其基本单位为 1ms，会在指令第一次被启动时生效，若是执行加速过程中，**S_1** 目标频率有变更，则加减速时间将会被重载执行，但若是已经进入减速区，则变更 **S_1** 目标频率无效。
6. **D** 仅限指定 Y0.0~Y0.11 输出，当指令启动后，其输出功能变为高速输出，一般基本指令输出点控制将无效。建议程序使用高速功能之后，就不再使用为一般输出功能。

7. **S_1** 为输出目标频率、**S_2** 为脉冲输出个数、**S_3** 为加减速时间 设定说明。



8. 脉冲输出的 Duty-OFF Time 跟 Duty-ON Time 比例为 1：1。
9. 此指令不限制使用次数，也不限制重复指定相同输出点，但限制每一次扫描周期里，每一个输出点只能被一个高速输出指令启动输出，且依先启动者先占用为原则。
10. 每个输出点之高速输出指令被启动后，皆必须等待该启动之高速输出指令被关闭后，才能让其它指定相同输出的指令被启动。
11. 当高速输出指令被启动于中断程序里，或者未被设计于主流程时，则建议搭配输出完成自动复位功能，由 PLC 于 END 指令里进行输出状态更新。
12. 暂停标志设定后，需等待启动指令再次被执行，并且输出完整一个脉冲之后，PLC 才会开始进行减速暂停（默认值）或立即停止输出之动作。
13. 每一个高速输出点搭配之 SM/SR 组件请参考 PLSY 指令说明。
14. ST 结构化语言编辑请参考 PLSY 指令的范例程序三说明。
15. 此指令允许执行到目标速度时，用户可变更目标速度，但是由于加减速区域在指令启动时，就已经设定好预计执行的加减速斜率，因此实际变更速度后，就会有如下两种可能情况发生(请参考情况 A 与 B 图示)，故建议一开始的目标速度，最好就是预计最后一段变更的目标速度，可参考底下建议方案图示。

情况 A：下一段目标速度比当前目标速度高(蓝色线为原始设计曲线，红色线为变更后曲线)



情况 A 图示说明：① 输出总个数

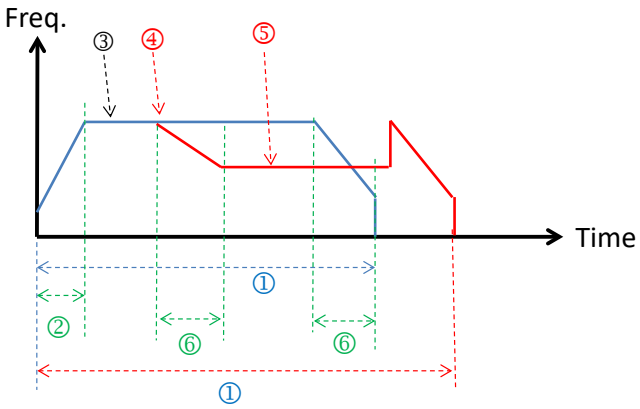
② 加速时间

③ 第一个目标速度

④ 变更目标速度的位置

⑤ 新的目标速度

情况 B：下一段目标速度比当前目标速度低(蓝色线为原始设计曲线，红色线为变更后曲线)



情况 B 图示说明：① 输出总个数

② 加速时间

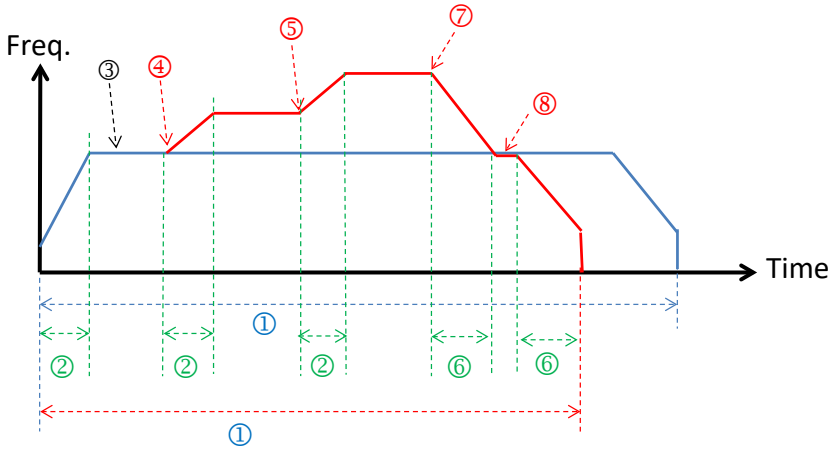
③ 第一个目标速度

④ 变更目标速度的位置

⑤ 新的目标速度

⑥ 减速时间

建议方案：第一段目标速度与最后一段目标速度相同(蓝色线为原始设计曲线，红色线为变更后曲线)



建议方案图示说明：① 输出总个数

② 加速时间

③ 第一个目标速度

④ 第一次变更目标速度的位置

⑤ 第二次变更目标速度的位置

⑥ 减速时间

⑦ 第三次变更目标速度的位置

⑧ 变回与的第一个目标速度相同的速度，然后执行减速停止的动作

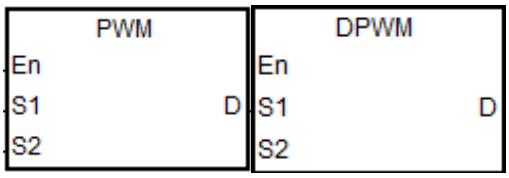
API	指令码			操作数	功能
2702	D	PWM		**S_1 · S_2 · D**	脉宽调变输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●	●	●	●		○		○	○		
S_2	●	●			●	●	●	●	●		○		○	○		
D		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D	●												

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



S_1 ： 脉冲输出宽度

S_2 ： 脉冲输出周期

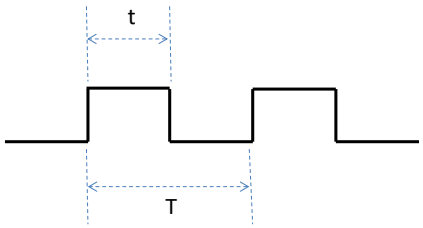
D ： 脉冲输出装置

指令说明：

16 位 PWM 指令输出基本单位为 100us，32 位 DPWM 指令输出基本单位为 1us。

输出指令	PWM	DPWM
S_1 输出数值范围	0 ~ 30000	0 ~ 60000
S_2 输出数值范围	0 ~ 30000	0 ~ 60000

S_1 脉冲输出工作宽度（Duty ON）定义为 t，**S_2** 脉冲输出周期时间（Cycle time）指定义为 T，如下图所示。建议输出数值 **S_1** ≦ **S_2**。



1. **D** 脉冲输出装置限制使用 Y0.0 ~ Y0.11 输出。
2. 本指令于程序中使用次数并无限制，但是同一次扫描周期只能一个相同输出点之输出指令被执行。当程序中有多个高速脉冲输出指令启动相同一个输出点时，则 PLC 会以先执行之指令输出。
3. 当 **S_1**≦0 或 **S_2**≦0 将视为 0 无输出（即 OFF），若 **S_1** > **S_2** 时，则视为 **S_1** = **S_2**；当 **S_1** = **S_2** 且 **S_2** 不为 0 时，则脉冲输出一直为 ON。

4. 脉冲输出宽度 **S_1**. 脉冲输出周期 **S_2** 可在 PWM 指令执行时更改。

5. 相关特殊寄存器 SR 对照表如下：

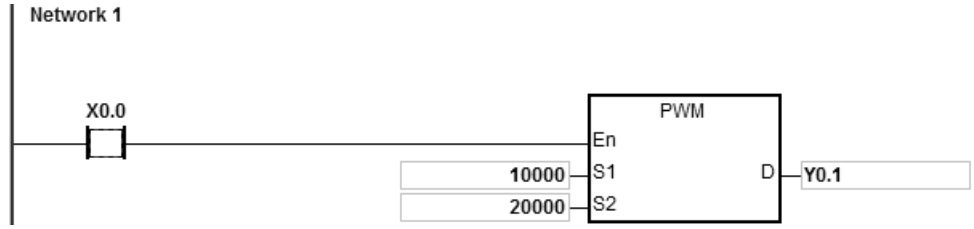
输出点编号	属性	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5
当前输出位置#1 (32 位)	R/W	SR460 SR461	SR474 SR475	SR480 SR481	SR494 SR495	SR500 SR501	SR514 SR515
输出点编号	属性	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
当前输出位置#1 (32 位)	R/W	SR520 SR521	SR534 SR535	SR540 SR541	SR554 SR555	SR560 SR561	SR574 SR575

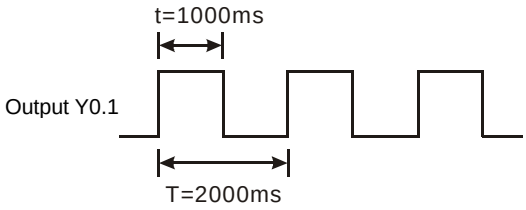
注#1：所有当前输出位置皆为停电保持。

ST 结构化语言编辑请参考下一个 DJOG 指令的程序范例说明。

程序范例：

当 X0.0=ON 时，Y0.1 输出以下脉冲，当 X0.0=OFF 时，Y0.1 输出也变成 OFF。





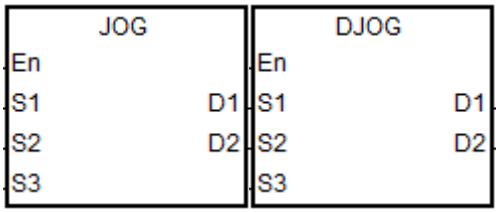
API	指令码			操作数	功能
2703	D	JOG		S_1，S_2，S_3，D_1，D_2	寸动输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1					●	●	●	●	●		○		○	○		
S_2					●	●	●	●	●		○		○	○		
S_3					●	●	●	●	●		○		○	○		
D_1		○														
D_2		○	○													

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
S_3		●	●			●	●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



S_1 ： 加速时间

S_2 ： 目标输出频率

S_3 ： 减速时间

D_1 ： 脉冲输出装置

D_2 ： 辅助输出装置

指令说明：

1. S_1 加速时间，其基本单位为 10ms，例如：输入 10 表示加速时间为 100ms。当指令启动之后，输出频率将每分成 10 次且每 10ms 加速一次，并且在 S_1 加速时间到达时，加速至 S_2 指定之 JOG 目标速度。
2. 当指令的被关闭时，其输出将会依指定减速时间每 10ms 减速一次，直到减速时间到达停止输出，但若需要立即停止输出，则可设定减速时间为 0。
3. 加速 S_1 与减速 S_3 时间之设定值数值范围为 0~3000，也即是 0~30 秒，当超过此范围时，以最小或最大值输出。此加减速时间会受扫描时间影响，若需精准时间切换其加减速过程，则建议使用 DDRVI 指令这类有加减速设计的输出指令。
4. S_2 目标输出频率范围为-200K~200KHz，当超过此范围时，以最小或最大值输出。当输出频率为正数，表示往正方向输出；当输出频率为负数，即表示往反方向输出。

5. **D_1** 输出点必须按照参考下表选择 Y0.0~Y0.11。**D_2** 辅助输出点建议参考下表所列之输出点，若自行选择其它输出点或 M 装置时，则输出模式选择 SR 将无效，并且默认为"脉冲+方向"模式，此 **D_2** 为方向输出点。

D_1 选择偶数输出点编号：

D_1 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
建议 **D_2** 方向输出点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
输出中标志	SM460	SM480	SM500	SM520	SM540	SM560
输出模式选择	SR462	SR482	SR502	SR522	SR542	SR562
当前输出位置	SR460 SR461	SR480 SR481	SR500 SR501	SR520 SR521	SR540 SR541	SR560 SR561
背隙补偿	SR478	SR498	SR518	SR538	SR558	SR578

D_1 选择奇数输出点编号：

D_1 输出点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
D_2 方向输出点	Y1.0 以上输出点或任意 M 组件（BOOL 类型）					
输出中标志	SM472	SM492	SM512	SM532	SM552	SM572
当前输出位置	SR474 SR475	SR494 SR495	SR514 SR515	SR534 SR535	SR554 SR555	SR574 SR575
背隙补偿	SR479	SR499	SR519	SR539	SR559	SR579

6. 输出模式选择 SR 为 0 时，表示"脉冲+方向"方式输出，当 SR 数值为 1 时，则为"A/B 相"方式输出。（备注：**D_1** 选用偶数输出点且 **D_2** 选用建议方向输出点，才可选择输出模式）

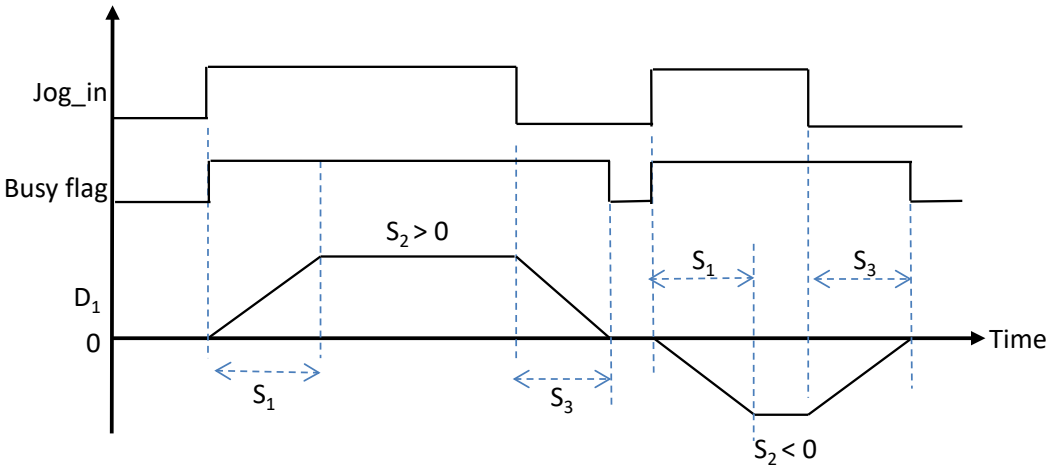
7. 当方向输出点未采用内定之输出点时，用户可参考下表 SR，设定方向比脉冲提前输出时间，以确保方向切换后，脉冲才开始输出。此 SR 的时间单位为 ms，可设定范围为 0~20ms，默认值为 0 表示不启动。

D_1 输出点	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5
SR 编号	SR640	SR641	SR642	SR643	SR644	SR645
D_1 输出点	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
SR 编号	SR646	SR647	SR648	SR649	SR650	SR651

注：此功能之计时来源时间为 PLC 扫描时间。（韧体支持版本须为 V1.02.20 以上版本）

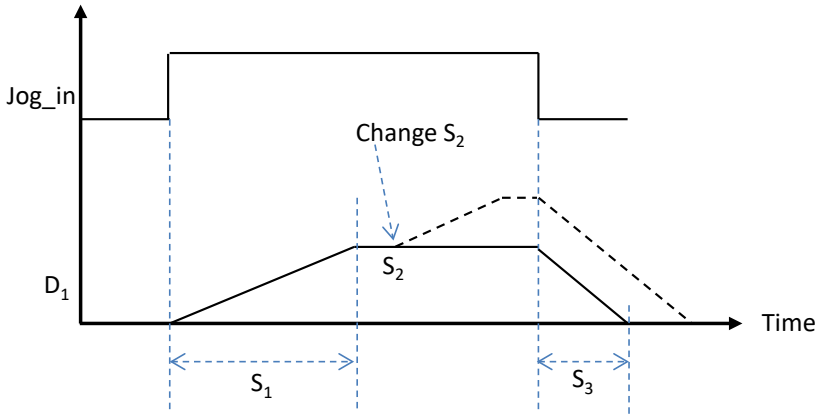
8. 当开始输出时，其输出中标志 SM 将被设定为 ON 状态，当输出完毕时，输出中标志将会自动清除为 OFF 状态，并且不会产生输出完成标志。

9. 其输出时序图如下图所示：（图中 Busy flag 请参考各轴输出中标志）



10. 指令执行过程中，其目标输出频率可被修改，但修改加减速时间无效。当新的目标频率比旧的高，则使用加速斜率；反之，则使用减速斜率。

11. 加速及减速之斜率在指令被启动时，依据其设定时间与目标速度进行换算后而设定，因此斜率在指令输出过程中将不会跟随目标频率变化，例如：原先目标频率 1kHz 输出，之后修改为 2kHz 输出，因此实际减速时间将与一开始设定时间不同。下图内的虚线为目标速度被变更后的加速与减速时序图。



12. 加减速过程，都是在前一个脉冲输出完成之后，才允许下一个输出变换速度，举例：目标速度是 100Hz，加速时间是设 100(即 1 秒) 则第一个输出会是 1Hz 速度 但等到 1Hz 输出完成时 ,下一个将会是 100Hz。（注：虽然每 10ms 切换一次目标速度，但因 1 个脉冲未输出完毕，而无法变速度）

程序范例：(以 ST 结构化语言编辑)

一般使用此指令时，优先建议采用梯形图语言编辑，若采用 ST 语言时，则必须搭配自动复位标志，以通知 PLC 归还高速输出控制权。以 Y0.0 输出为例的建议程序写法如下图。

```
0001 IF M0 THEN
0002    DJOG(1000,5000,100,Y0.0,Y0.1);
0003    M1 := TRUE;
0004 ELSIF (NOT M0) AND M1 THEN
0005    SM470 := TRUE;
0006    M1 := FALSE;
0007 END IF;
```

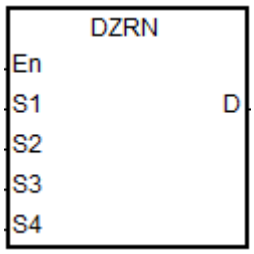
API	指令码			操作数	功能
2704	D	ZRN		**S_1，S_2，S_3，S_4，D**	原点回归

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
S_4	○															
D		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4	●												
D	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 回归目标频率

S_2 ： 原点（DOG）寸动（JOG）频率

S_3 ： 回原点模式选择

S_4 ： 原点(DOG)输入设备

D ： 脉冲输出装置

指令说明：

1. **S_1** 目标频率范围 1Hz~200kHz，**S_2** 寸动频率需比 **S_1** 目标频率小，且 **S_2** 寸动频率即为启动频率。若 **S_1** 小于 **S_2** 频率，则 **S_1** 自动被修正，与 **S_2** 频率相同。
2. **S_4** 与 **D** 操作数须参照输入与输出装置搭配组合使用，请勿执行中变更；**S_4** 操作数输入点建议使用 X0.0 ~ X0.15 这 16 点高速输入点(不受 PLC 扫描指令影响)，若是采用一般 X1.0 之后输入点或 M 装置时(韧体版本需 V1.02.10 以上)，则执行停止动作将受到 PLC 扫描时间影响。请搭配 **D** 操作数输出点与方向输出点建议选择如下表所示；当 **D** 输出选择不为默认”脉冲+方向”输出(默认值 0)时，请修改模式选择 SR 为 1，将模式改为 A/B 相输出。

轴编号	轴 1	轴 2	轴 3	轴 4	轴 5	轴 6
S_4 输入点	可任意选择 X0.0 ~ X0.15 输入点，但不同轴输出不可选择相同输入点 若 DOG 点有震动或开关弹跳的状况，请由 HWCONFIG 设定输入点滤波时间					
D 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10

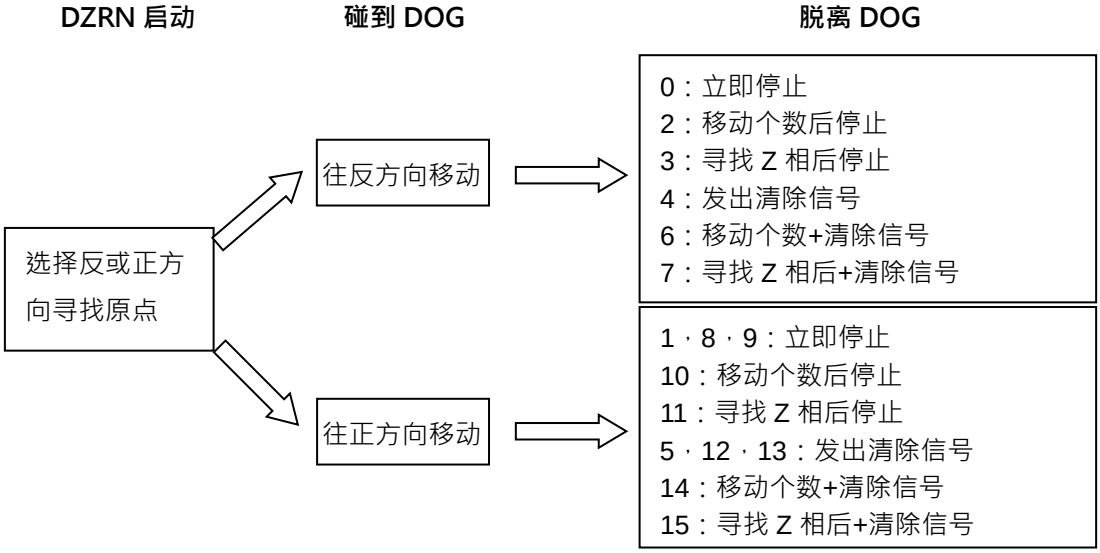
自动搭配输出方向点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
输出模式选择	SR462	SR482	SR502	SR522	SR542	SR562
输出中标志	SM460	SM480	SM500	SM520	SM540	SM560
输出完成标志	SM461	SM481	SM501	SM521	SM541	SM561
输出当前位置	SR460 SR461	SR480 SR481	SR500 SR501	SR520 SR521	SR540 SR541	SR560 SR561

3. **S_3** 为原点回归模式选择，其参数与功能代码分为上、下 16 位两个设定值，说明如下表

S3 为原点回归模式选择				
上 16 位	下 16 位			
b31~b16	b15~b6	b5	b4	b3~b0
移动个数 或 寻找 Z 相个数	保留	启动时寻找方向#1 0：往反方向 1：往正方向	设定 DOG 信号 0：A 接点 1：B 接点	模式设定 0~15(F)

注#1：此功能韧体支持版本为 V1.04.30 版以上。

请依照下图说明，并且根据脱离 DOG 后，欲执行的动作来选择模式



4. **S_3** 为原点回归模式选择，其功能代码分为上、下 16 位两个参数设定，其详细说明请参考下表：

功能项目	代码		说明
	高 16 位	低 16 位	
反向离开原点停止 （模式 0）	0	0	指令启动时，固定往反方向的目标速度寻找原点，当碰到原点时（OFF→ON）减速至寸动速度，并且持续以反方向行进，直到原点信号由 ON 变 OFF 时停止。
正向离开原点停止 （模式 1）	0	1	指令启动时，固定往反方向的目标速度寻找原点，当碰到原点时（OFF→ON）立即减速停止，然后以寸动速度往正方向行进，直到原点信号由 ON 变 OFF 时停止。

功能项目	代码		说明
	高 16 位	低 16 位	
模式 0 回原点后再移动	移动个数	2	回原点动作如同低 16 位代码 0 的动作，当回到原点之后，将再以移动个数指定的数量移动。当高 16 位个数为正的数值往正方向移动，反之，负的数值就往反方向移动。
模式 0 回原点后寻找 Z 相（Z 相输入点请由 HWCONFIG 设定）	Z 相次数	3	回原点动作如同低 16 位代码 0 的动作，当回到原点之后，将再依据寻找 Z 相个数移动。当高 16 位次数为正的数值往正方向寻找，反之，负的数值就往反方向寻找。假如：HWCONFIG 指定 X0.1 上升沿触发为 Z 相输入条件，则每次 X0.1 上升沿触发，即计数 1 次。
模式 0 回原点后输出清除信号（输出清除点请由 HWCONFIG 设定）	移动个数 或 Z 相次数	4+0=4 4+1=5 4+2=6 4+3=7 bit2=ON）	选择 4~7 分别表示选择代码 0~3 功能执行完成时，再依指定的输出点送出一个 ON 约 20 毫秒（ms）宽度的信号。此输出点指定范围为 Y0.12~0.15 及 Y1.0 ~ Y1.15，例如：HWCONFIG 指定 Y1.12，即表示 Y1.12 为输出清除信号。
正向离开原点停止（模式 1）	0	8+0=8 8+1=9 （bit3=ON）	同代码 1（模式 1）的执行动作
模式 1 回原点后再移动个数	移动个数	8+2=10 （bit3=ON）	回原点动作如同低 16 位代码 1 的动作，当回到原点之后，将再以移动个数指定的数量移动。当高 16 位个数为正的数值往正方向移动，反之，负的数值就往反方向移动。
模式 1 回原点后寻找 Z 相（Z 相输入点请由 HWCONFIG 设定）	Z 相次数	8+3=11 （bit3=ON）	回原点动作如同低 16 位代码 1 的动作，当回到原点之后，将再依据寻找 Z 相个数移动。当高 16 位次数为正的数值往正方向寻找，反之，负的数值就往反方向寻找。假如：HWCONFIG 指定 X0.1 上升沿触发为 Z 相输入条件，则每次 X0.1 上升沿触发，即计数 1 次。
模式 1 回原点后输出清除信号（输出清除点请由 HWCONFIG 设定）	0 或 移动个数或 Z 相次数	12 ~ 15 （bit3=bit2 =ON）	回到指定模式 1 的原点后，再输出 20ms 的清除信号
DOG 点为 B 接点		+16 （bit4=ON）	当低 16 位代码 bit4 为 ON 时 表示 DOG 点 ON□OFF 时为碰到原点，OFF□ON 表示离开原点。

5. 上表内的低 16 位代码数值判断与执行顺序为：判断 bit5 决定寻找原点方向→判断 bit4 决定原点（DOG）信号 → 判断 bit3 选择模式 0 或 1 方式回原点 → 执行 bit1 与 bit0 的回原点动作 → 执行 bit2 指定清除动作。
6. 当定位控制系统架构需要设置正反极限输入点时，请于编辑软件之 HWCONFIG 设定选择输入点与上升沿、下降沿触发条件，并且注意极限输入点不可与原点或 Z 相输入点相同。

   韧体 V1.10（含）以下，于执行回原点过程中，并且有启动极限点检查功能时，若发生碰极限的情况，不会设定极限警报标志。韧体 V1.12（含）以上，将会自动设定极限警报提示，供用户自行判断此警告标志，决定后续运行动作。（注：警告标志设定后，须由用户自行清除。）
7. 当指令运行完成指定功能之后，其输出结束标志才会产生，例如：选择代码 6 的功能，则会在 Z 相寻找完成时，PLC 才会设定完成标志。
8. 当 DZRN 指令启动之后，若是指定原点输入点与程序中外部输入中断同一个时，则中断服务程序将暂时不能被执行；且须直到 DZRN 指令被关闭之后，才能恢复使用。
9. 当 HWCONFIG 已经启动极限开关，且有外部输入中断服务程序时，此中断程序将会一起被执行。
10. 此 ZRN 指令不建议使用于 ST 语言编辑、中断程序以及仅被执行一次性的功能块编辑。

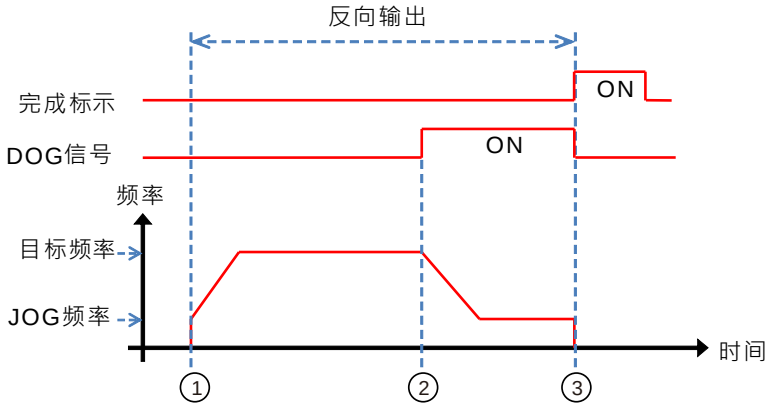
    韧体 V1.12（含）以上支持 ST 语言编辑，请参考范例说明。
11. 各功能之动作示意图如下：

    功能代码 0：

    ① 启动 DZRN 功能，固定往反方向的目标速度寻找原点

    ② 碰到近点 DOG 信号，减速至 JOG 速度，并且持续以反方向行进，直到原点信号由 ON 变 OFF 时停止

    ③ 离开近点 DOG 信号，信号由 ON 变 OFF 时停止

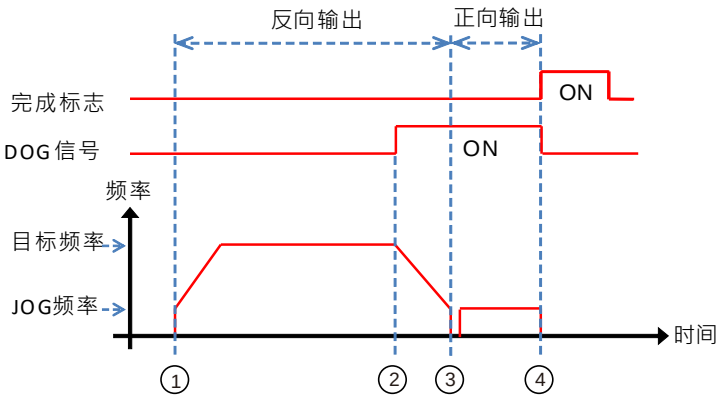


功能代码 1：

① 启动 DZRN 功能，固定往反方向的目标速度寻找原点

② 碰到近点 DOG 信号，减速并且在反向后，以 JOG 速度正方向行进，直到原点信号由 ON 变 OFF 时停止

③ 离开近点 DOG 信号，信号由 ON 变 OFF 时停止



功能代码 2：

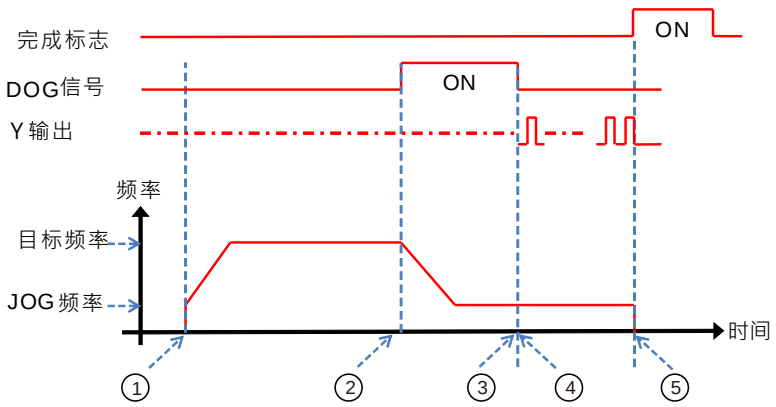
① 启动 DZRN 功能，固定往反方向的目标速度寻找原点

② 碰到近点 DOG 信号，减速至 JOG 速度，并且持续往反方向行进

③ 离开近点 DOG 信号，信号由 ON 变 OFF 时，将再以移动个数指定的数量移动

④ 第 1 个脉冲开始输出

⑤ 第 100 个脉冲输出完毕后停止，且完成标志为 ON



功能代码 3：

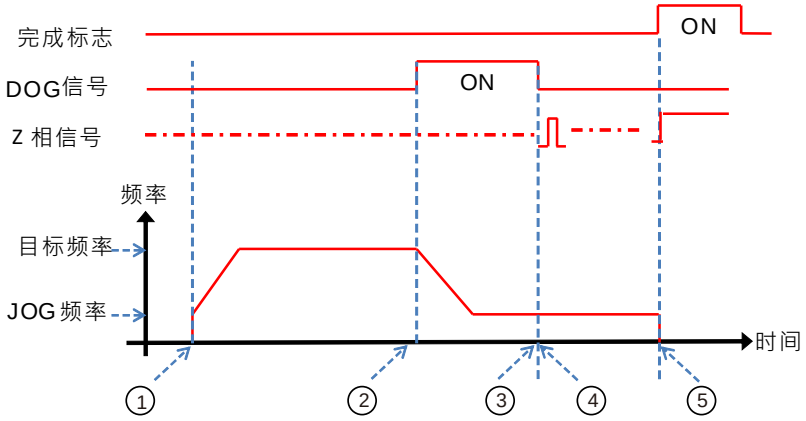
① 启动 DZRN 功能，固定往反方向的目标速度寻找原点

② 碰到近点 DOG 信号，减速至 JOG 速度，并且持续往反方向行进

③ 离开近点 DOG 信号，信号由 ON 变 OFF 时，将再依据寻找 Z 相个数移动

④ 第 1 个 Z 相脉

⑤ 第 2 个 Z 相脉完毕后停止，且完成标志为 ON

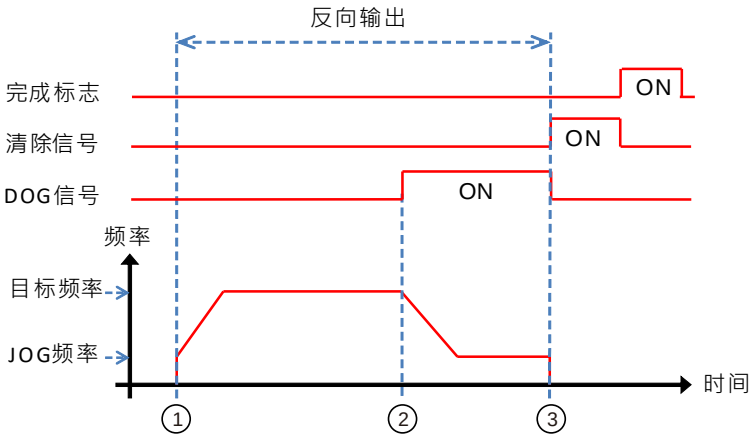


功能代码 4：

① 启动 DZRN 功能，固定往反方向的目标速度寻找原点

② 碰到近点 DOG 信号，减速至 JOG 速度，并且持续以反方向行进，直到原点信号由 ON 变 OFF 时停止

③ 离开近点 DOG 信号，信号由 ON 变 OFF 时停止，此时清除信号为 ON（约 20 毫秒）



功能代码 0 + 启动反极限功能：

从 HWCONFIG 设定反极限输入点，并下载至 PLC 内。(注：指令启动时会自动判断反极限功能)

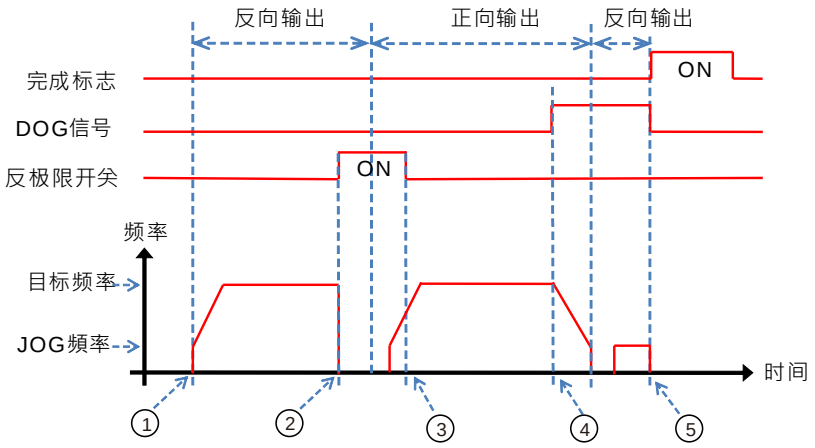
① 启动 DZRN 功能，固定往反方向的目标速度寻找原点

② 碰到反极限开关后，停止行进，反向后，开始往正方向行进

③ 离开反极限开关，持续正方向行进

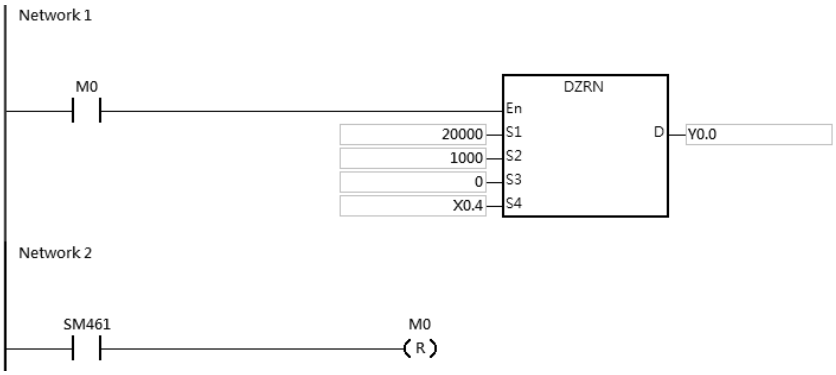
④ 碰到近点 DOG 信号，减速并且在反向后，以 JOG 速度反方向行进

⑤ 离开近点 DOG 信号，信号由 ON 变 OFF 时，停止输出



程序范例一：

当 M0=ON 时，以 20kHz 频率从 Y0.0 输出脉冲开始往反方向寻找原点，当碰到原点信号（DOG）X0.4=ON 时变成以寸动速度 1kHz 继续往反方向移动，直到 X0.4 从 ON 变 OFF 时立即停止。



说明：

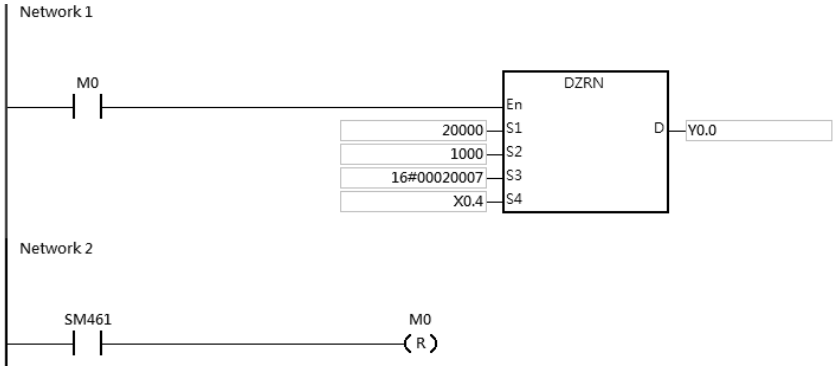
S_3 设定为 0

高 16 位[0000]=不启动移动个数或寻找 Z 相功能

低 16 位[0000]=模式 0，碰到 DOG 往反方向移动，脱离 DOG 立即停止

程序范例二：

当 M0=ON 时，以 20kHz 频率从 Y0.0 输出脉冲，开始往反方向寻找原点，当碰到原点信号（DOG）X0.4=ON 时，减速变成以寸动速度 1kHz 继续往反方向移动，直到 X0.4=OFF 后，再往正向移动寻找 Z 相，直到 X0.5（Z 相）输入脉冲到第 2 次上升沿触发信号出现，并再从 Y1.4 输出一个脉冲（ON 宽度 20ms）后结束。



说明：

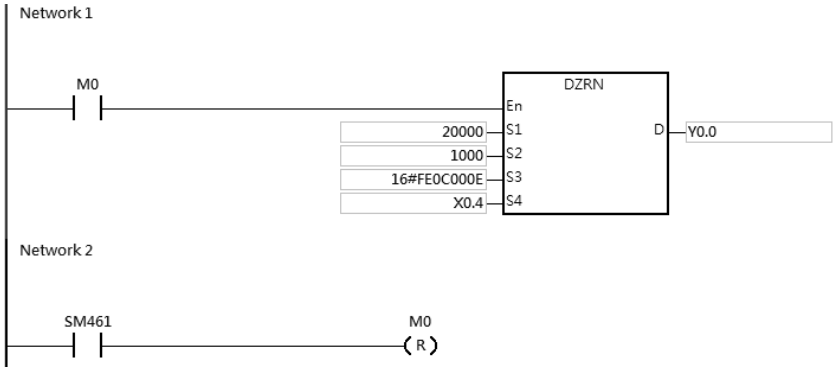
1. 在 HWCONFIG 指定 X0.5 上升沿触发为 Z 相输入条件，则每次 X0.5 上升沿触发，即计数 1 次。
2. 在 HWCONFIG 指定 Y1.4 为输出清除信号。
3. S_3 设定为 16#00020007，高 16 位=02 表示寻找 Z 相 2 次；低 16 位=07 表示模式设定为寻找 Z 相后，输出清除信号（20ms）。

   高 16 位[0002]=脱离 DOG，往正寻找 Z 相 2 次

   低 16 位[0007]=模式 7，碰到 DOG 往反方向移动，脱离 DOG，寻找 Z 相后，输出清除信号（20ms）。

程序范例三：

当 M0=ON 时，以 20kHz 频率从 Y0.0 输出脉冲，开始往反方向寻找原点，当碰到原点信号（DOG）X0.4=ON 时，减速变成以寸动速度 1kHz 继续往正方向移动，直到 X0.4=OFF 后，再往反方向移动 500 个脉冲，并再从 Y1.4 输出一个脉冲（ON 宽度 20ms）后，停止输出。



说明：

1. 在 HWCONFIG 指定 Y1.4 为输出清除信号。
2. S_3 设定为 16# FE0C000E

   高 16 位[FE0C=-500]=脱离 DOG，往反方向移动 500 个脉冲

   低 16 位[000E]=模式 14 ·碰到 DOG 往正方向移动 ·脱离 DOG ·输出指定脉冲数后 ·输出清除信号(20ms)。

程序范例四：(以 ST 结构化语言编辑)

一般使用此指令时，建议优先采用梯形图语言编辑，若采用 ST 语言时 (V1.12 以上支持)，须等到完成回原点后 (ex：SM461 完成标志 ON)，再关闭指令，而且必须搭配自动复位标志，以通知 PLC 归还高速输出控制权。以 Y0.0 输出为例的建议程序写法如下图。

```
0001 IF M0 THEN
0002     DZRN(1000,100,0,X0.0,Y0.0);
0003     M1:=TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
```

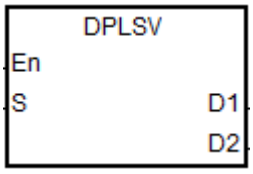
API	指令码			操作数	功能
2705	D	PLSV		**S**，**D_1**，**D_2**	可调变脉冲输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S							●	●	●		○		○	○		
D_1		○														
D_2		○	○													

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S ： 脉冲输出目标频率

D_1 ： 脉冲输出装置

D_2 ： 脉冲方向输出装置

指令说明：

1. **S** 指定脉冲输出频率，使用集电极输出机种时，可指定范围为-200kHz ~ +200k Hz，使用差动输出机种时，可指定范围为-4MHz ~ +4M Hz。其中正负号代表正反方向。而在脉冲输出中仍可任意变更脉冲输出频率，但变更频率与前次频率为不同方向时，则此指令将自动停止输出一个扫描周期后再重新输出目标频率。
2. **D_1** 与 **D_2** 脉冲输出装置之选择请参考下表所示；当 **D_1** 与 **D_2** 输出模式选择不为默认”脉冲+方向”输出（默认值 0）时，请修改模式选择 SR 为 1，将模式改为 A/B 相输出。

轴编号	轴 1	轴 2	轴 3	轴 4	轴 5	轴 6
D_1 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
D_2 输出方向点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
输出模式选择	SR462	SR482	SR502	SR522	SR542	SR562
输出中标志	SM460	SM480	SM500	SM520	SM540	SM560
输出当前位置	SR460 SR461	SR480 SR481	SR500 SR501	SR520 SR521	SR540 SR541	SR560 SR561

3. **D_2** 脉冲方向输出装置，若指定不为建议搭配之输出点，则将受扫描时间影响。
4. **D_2** 脉冲方向输出装置，对应 **S** 的正反做动作，当 **S** 为正（+）时 **D_2** 为 OFF，当 **S** 为负（-）时 **D_2**：ON。

5. 此指令无加减速之设定，因此无法执行开始之加速与停止之减速动作，若是必须达到加减速之功能请利用 API 0703 DRAMP 指令来做脉冲输出频率的加减。
6. 当指令执行脉冲输出中，若驱动条件变为 OFF 则直接停止输出。
7. 当方向输出点未采用内定之输出点时，用户可参考下表 SR，设定方向比脉冲提前输出时间，以确保方向切换后，脉冲才开始输出。此 SR 的时间单位为 ms，可设定范围为 0~20ms，默认值为 0 表示不启动。

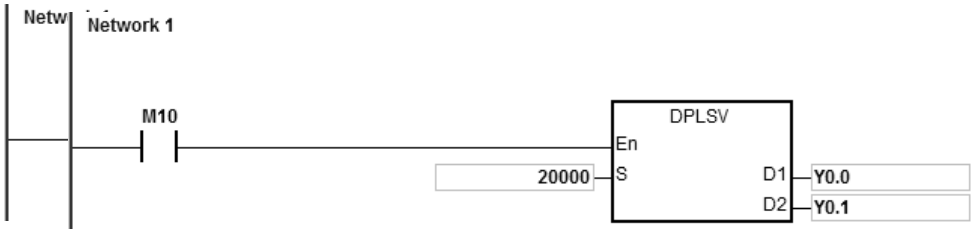
D_1 输出点	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5
SR 编号	SR640	SR641	SR642	SR643	SR644	SR645
D_1 输出点	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
SR 编号	SR646	SR647	SR648	SR649	SR650	SR651

注：此功能之计时来源时间为 PLC 扫描时间。（韧体支持版本须为 V1.02.20 以上版本）

8. ST 结构化语言编辑请参考 DJOG 指令的程序范例说明。

程序范例：

当 M10=ON 时，以 20kHz 频率从 Y0.0 输出脉冲，Y0.1=OFF 表示为正方向。



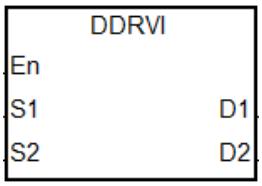
API	指令码			操作数	功能
2706	D	DRVI		**S_1 · S_2 · D_1 · D_2**	相对地址高速脉冲输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
D_1		○														
D_2		○	○													

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 脉冲输出数目（相对位置）

S_2 ： 脉冲输出频率

D_1 ： 脉冲输出装置

D_2 ： 脉冲方向输出装置

指令说明：

1. **S_1** 指定脉冲输出数目（相对定位），可指定范围为-2,147,483,648 ~ +2,147,483,647 个，其中正负号代表正反方向。

   当指令启动且 **S_1** =0 时，即表示不输出，且输出完成标志不会被设定为 ON。若用户设计之输出个数，有可能出现 0 数值，请自行增加判断式决定执行动作。

2. **S_2** 指定脉冲输出频率，使用集电极输出机种时，可指定范围为 0Hz ~ +200k Hz，使用差动输出机种时，可指定范围为 0Hz ~ +4M Hz。当 **S_2** 输出频率设定低于启动/停止频率（请参考下页 SR 列表）时，PLC 将自动限制在启停频率。

3. **D_1** 与 **D_2** 脉冲输出装置之选择请参考下表所示；当 **D_1** 与 **D_2** 输出模式选择不为默认"脉冲+方向"输出（默认值 0）时，请修改模式选择 SR 为 1，将模式改为 A/B 相输出。

轴编号	轴 1	轴 2	轴 3	轴 4	轴 5	轴 6
D_1 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
D_2 输出方向点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
输出模式选择	SR462	SR482	SR502	SR522	SR542	SR562

DDRVI 与 DDRVA 支持 **D_1** 输出装置可选择奇数点（如下表）输出，并且仅支持"脉冲+方向"模式输出。建议方向输出点不要选择 Y0.0, Y0.2…Y0.10 这些高速输出点。

轴编号	轴 7	轴 8	轴 9	轴 10	轴 11	轴 12
D_1 输出点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
D_2 方向输出点	Y1.0 以上输出点或任意 M 组件（BOOL 类型）					

4. **D_2** 脉冲方向输出装置，若指定不为上表建议搭配之输出点，则其输出动作将受扫描时间影响。
5. **D_2** 脉冲方向输出装置使用输出模式为"脉冲+方向"时，对应 **S_1** 的正反做动作，当 **S_1** 为负(-)时 **D_2**=ON。当 **S_1** 为正(+)时 **D_2** 为 OFF，反向脉冲输出结束后 **D_2** 并不会立即 OFF，须等指令执行接点开关 OFF 时 **D_2** 才会切为 OFF。
6. 此指令输出时需搭配启动/结束频率、加速时间与减速时间之参数，其各轴对应之 SM/SR 请参考下表所示。

D_1 输出装置选择偶数点时

D_1 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
输出中标志	SM460	SM480	SM500	SM520	SM540	SM560
输出完成标志	SM461	SM481	SM501	SM521	SM541	SM561
输出方向脚反向设定	SM462	SM482	SM502	SM522	SM542	SM562
输出暂停标志	SM463	SM483	SM503	SM523	SM543	SM563
固定斜率加减速	SM469	SM489	SM509	SM529	SM549	SM569
输出完成自动复位	SM470	SM490	SM510	SM530	SM550	SM570
输出完成设定 I 中断	SM471	SM491	SM511	SM531	SM551	SM571
对应之 I 中断编号	I500	I501	I502	I503	I504	I505
指令关闭或暂停时立即停止输出标志	SM476	SM496	SM516	SM536	SM556	SM576
输出当前位置	SR460 SR461	SR480 SR481	SR500 SR501	SR520 SR521	SR540 SR541	SR560 SR561
启动/结束频率（Hz）	SR463	SR483	SR503	SR523	SR543	SR563
加速时间（ms）	SR464	SR484	SR504	SR524	SR544	SR564
减速时间（ms）	SR465	SR485	SR505	SR525	SR545	SR565
固定斜率之目标频率	SR472 SR473	SR492 SR493	SR512 SR513	SR532 SR533	SR552 SR553	SR572 SR573
当前输出频率	SR610 SR611	SR612 SR613	SR614 SR615	SR616 SR617	SR618 SR619	SR620 SR621

D_1 输出装置选择奇数点时

D_1 输出点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
输出中标志	SM472	SM492	SM512	SM532	SM552	SM572
输出完成标志	SM473	SM493	SM513	SM533	SM553	SM573
输出暂停标志	SM474	SM494	SM514	SM534	SM554	SM574

输出完成自动复位	SM475	SM495	SM515	SM535	SM555	SM575
指令关闭或暂停时立即停止输出标志	SM477	SM497	SM517	SM537	SM557	SM577
输出当前位置	SR474 SR475	SR494 SR495	SR514 SR515	SR534 SR535	SR554 SR555	SR574 SR575
启动/结束频率（Hz）	SR476	SR496	SR516	SR536	SR556	SR576
加速/减速时间（ms）	SR477	SR497	SR517	SR537	SR557	SR577

奇数点输出不支持:固定斜率加减速及输出完成中断

若输出正在执行中，当指令关闭或暂停时，默认皆为减速停止输出，若需要立即停止输出，请自行设定立即停止输出标志。

7. V1.06.00 版（含）以上，新增 SR652/SR653=加速区脉冲个数显示；SR654/SR655=减速区脉冲个数显示。各轴对应之 SM/SR 参数属性、默认值与停电保持请参考下表所示。

项目	读写属性	Stop→Run	Run→Stop	Power ON	出厂值	停电保持
输出中标志	R	OFF	OFF	OFF	OFF	否
输出完成标志	R / W	OFF	OFF	OFF	OFF	否
输出暂停标志	R / W	OFF	OFF	OFF	OFF	否
输出完成自动复位	R / W	OFF	OFF	OFF	OFF	否
固定斜率加减速	R / W	OFF	OFF	OFF	OFF	否
输出完成设定 I 中断	R / W	OFF	OFF	OFF	OFF	否
指令关闭或暂停时立即停止输出标志	R / W	OFF	OFF	OFF	OFF	否
输出当前位置	R / Wd	不变化	不变化	不变化	0	是
启动/结束频率（Hz）	R / Wd	不变化	不变化	不变化	200	是
加速时间（ms）	R / Wd	不变化	不变化	不变化	200	是
减速时间（ms）	R / Wd	不变化	不变化	不变化	200	是
固定斜率之目标频率	R / Wd	不变化	不变化	不变化	0	是
当前输出频率	R	0	0	0	0	否

注：R 表示可读取，W 表示可写入，Wd 表示可写入但是高速输出执行中不可写入

8. 当输出完成自动复位功能被启动之后，此标志将会在每次输出完成时自动清除，故下次输出动作若是持续需要自动复位功能，则需再次设定此标志。建议使用上升沿或下降沿方式执行单次设定。此功能通常被运用于输出指令被启动后，无法再被 PLC 程序扫描执行的状况下，如：指令于中断程序里启动。

9. 当使用输出完成设定 I 中断功能之后，此标志将会在输出完成且进入中断程序时自动清除，故下次输出动作若是持续需要产生中断，则需再次设定此标志。建议使用上升沿或下降沿方式执行单次设定。

10. 当方向输出点未采用内定之输出点时，用户可参考下表 SR，设定方向比脉冲提前输出时间，以确保方向切换后，脉冲才开始输出。此 SR 的时间单位为 ms，可设定范围为 0~20ms，默认值为 0 表示不启动。

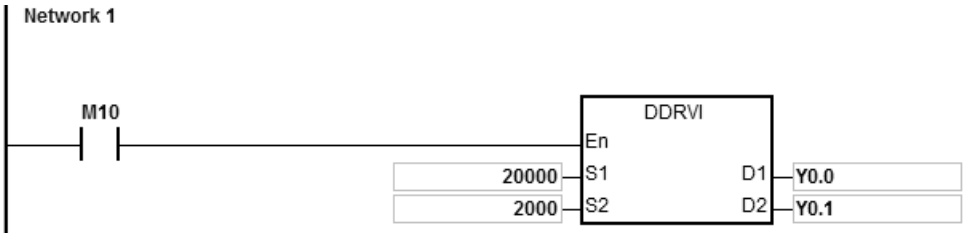
D_1 输出点	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5

SR 编号	SR640	SR641	SR642	SR643	SR644	SR645
D_1 输出点	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
SR 编号	SR646	SR647	SR648	SR649	SR650	SR651

注：此功能之计时来源时间为 PLC 扫描时间。（韧体支持版本须为 V1.02.20 以上版本）

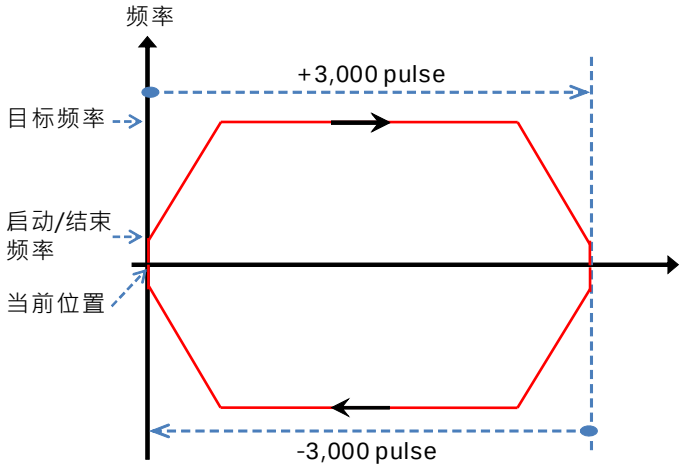
程序范例 1：

当 M10=ON 时，以 2kHz 频率从 Y0.0 输出脉冲数目 20,000 个（相对指定），Y0.1=OFF 表示为正方向。



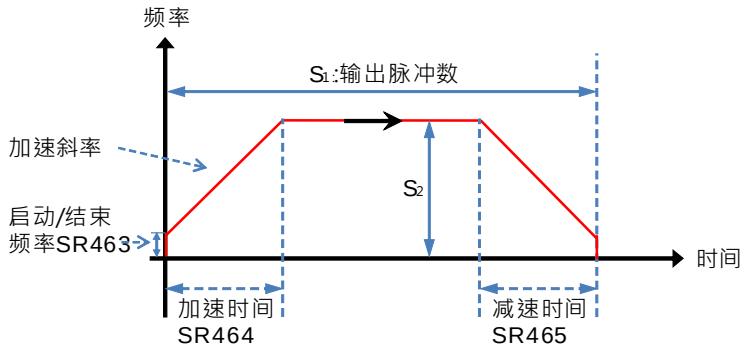
补充说明：

1. 相对定位方式的动作说明： 指将从当前位置开始的移动距离以正/负符号来指定的方式，也可叫做相对定位方式。



2. 相对定位的设定项目与加减速的设定：

a) 以 Y0.0 输出为例之输出曲线图：

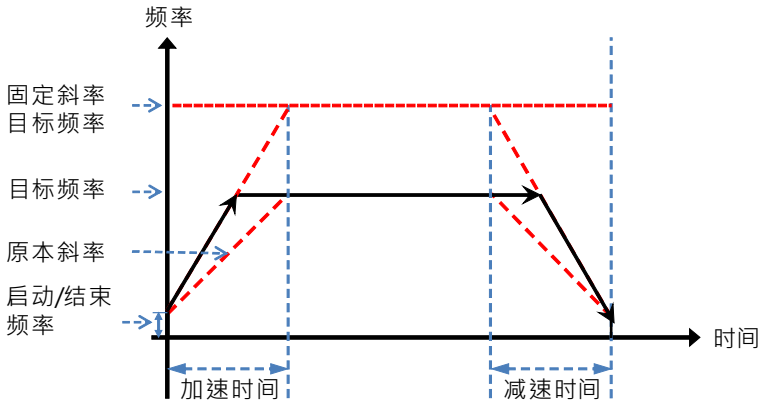


b) 本指令可同时编写多个使用同组输出于用户程序中，但是每次 PLC 程序执行时只能启动一组指令，例如已有指令起动 Y0.0 这一组输出时，则其它使用同 Y0.0 输出的指令将不会被执行；所以启动指令之顺序是由先启动者先执行为原则。

c) 指令启动后，其所有参数将不接受修改，直到指令关闭。

3. 固定斜率加减速设定：

a) 以 Y0.0 输出为例之输出曲线图：



b) 启动固定斜率标志 SM469，将固定斜率之目标频率写入 SR472，

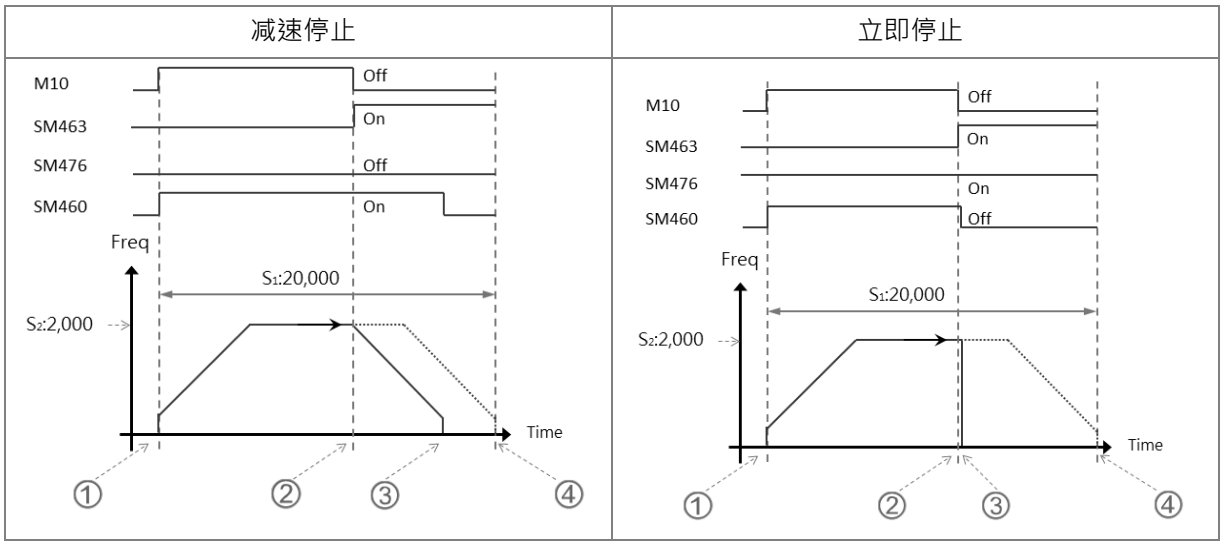
c) 如上图，DDRVI 将以新的斜率取代原本的斜率，执行定位动作

d) 一般斜率是依照起动/结束频率、目标频率和加减速时间来决定的
固定斜率是依照起动/结束频率、固定斜率目标频率和加减速时间来决定的

e) 当固定斜率之目标频率小于目标频率，则不启动固定斜率功能

4. 输出暂停标志与立即停止标志动作说明：

a) 以程序范例 1 为例的输出曲线图：



b) 减速停止动作说明

① M10 为 ON，启动 DDRVI 指令，开始输出脉冲并且 SM460 输出中标志为 ON

② 当指令被关闭（M10）为 OFF 或暂停标志被启动（SM463）为 ON 时，会确认立即停止标志（SM476）的状态

③ 如果立即停止标志（SM476）为 OFF 时，则减速停止并且 SM460 输出中标志为 OFF

④指令启动后，若 M10 没有 OFF 及暂停标志 SM463 没有 ON，则 Y0.0 输出脉冲 20,000 个后停止

c) 立即停止动作说明硬件极限功能说明：

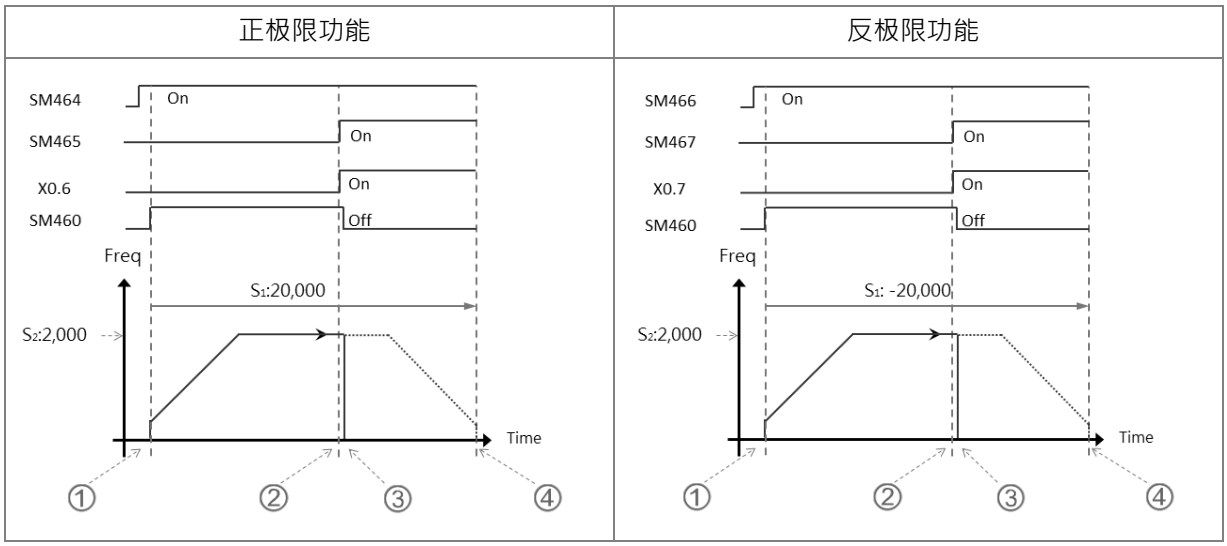
① M10 为 ON，启动 DDRVI 指令，开始输出脉冲并且 SM460 输出中标志为 ON。

② 当指令被关闭（M10）为 OFF 或暂停标志被启动（SM463）为 ON 时，会确认立即停止标志（SM476）的状态。

③ 如果立即停止标志（SM476）为 ON 时，则立即停止并且 SM460 输出中标志为 OFF。

5. 硬件极限功能说明：

a) 以轴 1（Y0.0/Y0.1）输出为例的输出曲线图：



b) 正极限动作说明

⓪从 HWCONFIG 设定正极限输入点（以 X0.6 上升沿触发为例），并下载至 PLC 内。当 SM464 设定为 ON，即表示启动正极限功能。

① DDRVI 往正方向输出脉冲 20,000 个。（注：若刚启动时正极限点成立，则指令不会正方向输出）

②当外部机构触发极限输入点 X0.6 为 ON 时，代表硬件正极限功能条件成立

③ DDRVI 指令立即停止输出，并且 SM465 正极限警报为 ON

④若硬件极限功能条件未成立，则 Y0.0 输出脉冲 20,000 个后停止

c) 反极限动作说明

⓪从 HWCONFIG 设定反极限输入点（以 X0.7 上升沿触发为例），并下载至 PLC 内。当 SM466 设定为 ON，即表示启动反极限功能。

① DDRVI 往反方向输出脉冲 20,000 个。（注：若刚启动时反极限点成立，则指令不会反方向输出）

②当外部机构触发极限输入点 X0.7 为 ON 时，代表硬件反极限功能条件成立

③ DDRVI 立即停止输出，并且 SM467 反极限警报为 ON

④若硬件极限功能条件未成立，则 Y0.0 输出脉冲-20,000 个后停止

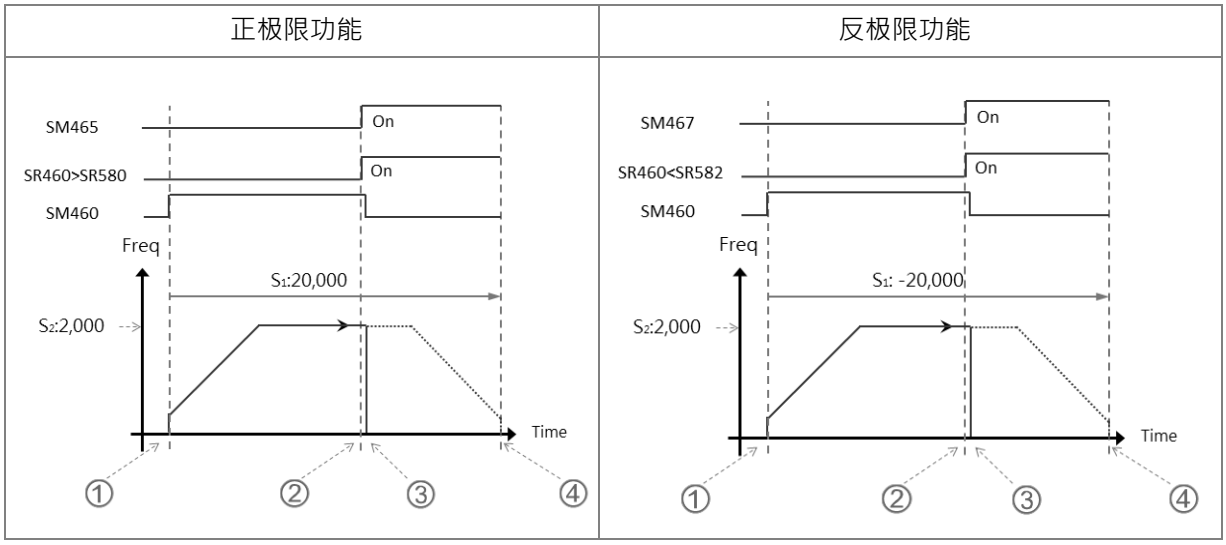
d) 极限功能相关 SM 标志

轴编号	轴 1	轴 2	轴 3	轴 4	轴 5	轴 6
硬件正极限启动标志	SM464	SM484	SM504	SM524	SM544	SM564
正极限警报标志	SM465	SM485	SM505	SM525	SM545	SM565
硬件反极限启动标志	SM466	SM486	SM506	SM526	SM546	SM566
反极限警报标志	SM467	SM487	SM507	SM527	SM547	SM567

备注：极限点警报标志由 PLC 自动依据启动标志状态，进行状态设定或清除。

6. 软件极限功能说明：

a) 以轴 1（Y0.0/Y0.1）输出为例的输出曲线图：



b) 正极限动作说明

⓪先由以 HWCONFIG 设定正向软件极限位置（以 SR580=11,000 为例），下载至 PLC 后，代表正极限功能被启动。

① DDRVI 开始往正方向输出脉冲 20,000 个。（注：若刚启动时已超出正极限，则指令不会正方向输出）

② 当现在位置 SR460>极限位置 SR580 时，代表软件正极限功能条件成立。

③ DDRVI 立即停止输出，并且 SM465 正极限警报为 ON。

④ 若极限功能条件未成立，则 Y0.0 输出脉冲 20,000 个后停止。

c) 反极限动作说明

⓪先由以 HWCONFIG 设定负向软件极限位置（以 SR582=-11,000 为例），下载至 PLC 后，代表反极限功能被启动。

① DDRVI 开始往反方向输出脉冲 20,000 个。（注：若刚启动时已超出反极限，则指令不会反方向输出）

② 当现在位置 SR460 < 极限位置 SR582 时，代表硬件反极限功能条件成立。

③ DDRVI 立即停止输出，并且 SM467 反极限警报为 ON。

④ 若极限功能条件未成立，则 Y0.0 输出脉冲-20,000 个后停止。

d) 下降沿触发说明

极限开关	A 接点	B 接点
上下降沿触发选择	上升沿触发	下降沿触发
极限开关动作说明	启动时机=由 low 到 high 及维持在 high	启动时机=由 high 到 low 及维持在 low

e) 极限功能相关 SR

轴编号	轴 1	轴 2	轴 3	轴 4	轴 5	轴 6
正向软件极限位置	SR580 SR581	SR584 SR585	SR588 SR589	SR592 SR593	SR596 SR597	SR600 SR601
反向软件极限位置	SR582 SR583	SR586 SR587	SR590 SR591	SR594 SR595	SR598 SR599	SR602 SR603

正向输出极限定义：输出当前位置大于正向极限位置时，即立即停止输出。

反向输出极限定义：输出当前位置小于反向极限位置时，即立即停止输出。

当正向与反向两个极限值皆为 0 时，表示软件极限功能不启动。由于此软件极限功能为输出指令被扫描时，才会进行检查输出位置之动作，故停止动作将会受 PLC 程序扫描影响，若需更快且实时停止之动作，请改用外部输入点为极限点方式。

7. **ST 结构化语言 DDRVI 编辑说明：**

```
0001 IF M0 THEN
0002    DDRVI(1000,1000,Y0.0,Y0.1);
0003    M1 := TRUE;
0004 ELSIF M1 AND SM461 THEN
0005    SM470 := TRUE;
0006    M1 := FALSE;
0007 END_IF;
```

动作说明：

7.1. 当 M0=ON 时，Y0.0 输出 1000 个 1kHz 频率脉冲，并且设定 M1=ON，用来记录 Y0.0 高速控制权已经被使用。

7.2. 当脉冲输出完毕 SM461=ON 并且 M1=ON 时，触发 Y0.0 的自动复位标志 SM470=ON，通知 PLC 释放高速输出控制权。

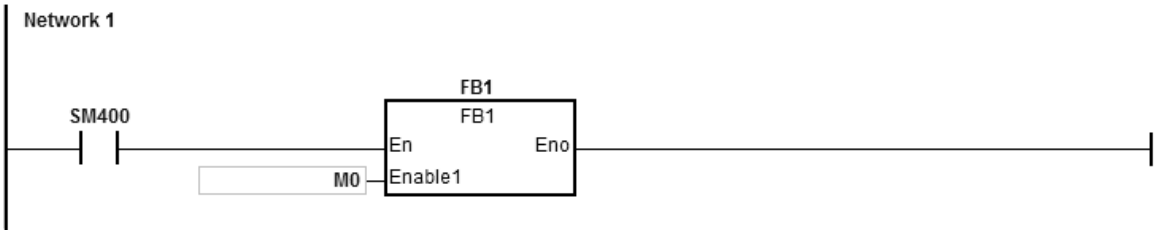
7.3. 当 M0 再次由 OFF>ON 时，脉冲输出即会重新启动。

7.4. 以上程序若有 2 个高速输出程序以上时，请勿重复使用 M1 这个临时性的变量名称。

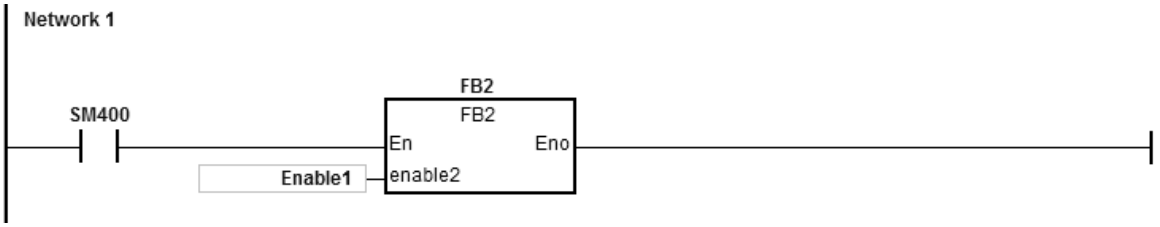
7.5. 若输出过程中有可能遇到要求暂停输出的动作，则不建议采用 ST 语言编辑。

8. **2 层功能块 DDRVI 编辑说明：**

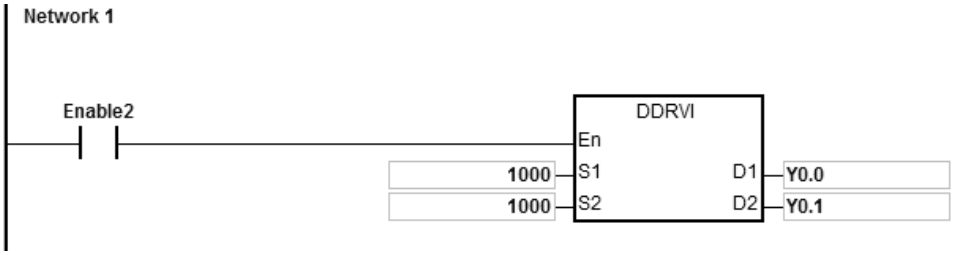
主程序：



功能块 FB1：



功能块 FB2：



动作说明：

当 M0=ON 时，Y0.0 输出 1000 个 1kHz 频率脉冲，Y0.1=OFF 表示为正方向。

9. 输出执行中变更目标速度说明，请务必先参考 PLSR 指令说明。请注意，每次变更目标速度，皆须到达目标速度之后，才可再进行下一次目标速度变更。

10. **输出执行中变更目标位置功能说明：**

A. 此功能需搭配 V1.02.30 版以上韧体才可使用；并且不允许变更输出方向之变换新位置，如：原输出目标位置为正方向，则只可变更往正方向继续输出。

B. 当用户设定变更标志之后，其实际变更时机将于扫描到该输出指令时才会进行变更，并且于变更完成后清除变更标志。若用户设定的目标位置不正确，则此变更标志将持续保持为 ON。

C. 各输出轴（点）所对应之变更标志 SM，请参考下表：

输出编号	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5
SM 编号	SM478	SM479	SM498	SM499	SM518	SM519
输出编号	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
SM 编号	SM538	SM539	SM558	SM559	SM578	SM579

D. 此变更目标位置功能，适用于 DDRVI 与 DDRVA 指令。当采用 DDRVI 指令时，一旦有设定变更标志，则每次都会执行变更，即使变更的输出个数为 0，也会立即执行停止输出的动作；当采用 DDRVA 指令时，且变更的目标位置与前次相同时，则不执行变更目标位置的动作。

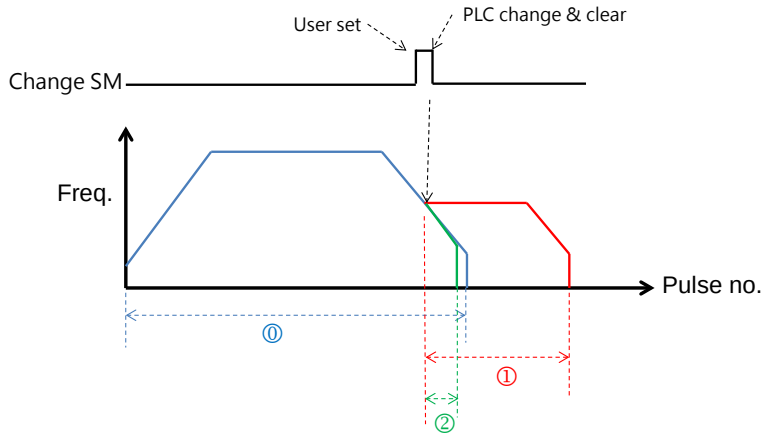
E. 输出执行中变更目标位置，将有可能发生底下三种情况与输出时序图：

情况 A：变更时机点发生于输出减速输出区

⓪：初次启动时，设定的目标位置（蓝色）

①：当新变更的目标输出个数 >= 剩余输出的个数时，将会输出的曲线（红色）

②：新变更的目标输出个数 < 剩余输出的个数时，将会输出的曲线（绿色）

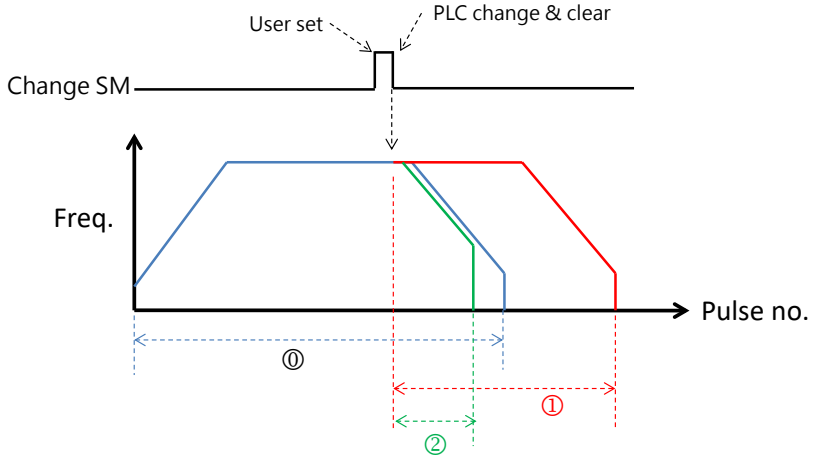


情况 B：变更时机点发生目标速度输出区

⓪：初次启动时，设定的目标位置（蓝色）

①：当新变更的目标输出个数 >= 减速区输出的个数时，将会输出的曲线（红色）

②：新变更的目标输出个数 < 减速区输出的个数时，将会输出的曲线（绿色）

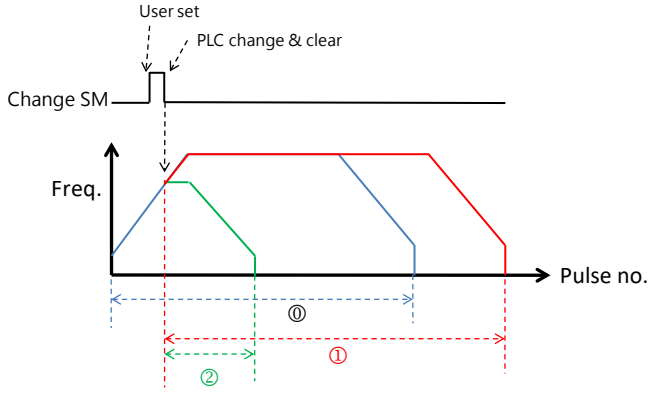


情况 C：变更时机点发生于输出加速输出区

⓪：初次启动时，设定的目标位置（蓝色）

①：当新变更的目标输出个数 >= 减速区输出的个数时，将会输出的曲线（红色）

②：新变更的目标输出个数 < 减速区输出的个数时，将会输出的曲线（绿色）



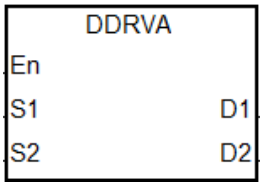
API	指令码			操作数	功能
2707	D	DRVA		**S_1，S_2， D_1，D_2**	绝对地址高速派波输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
D_1		○														
D_2		○	○													

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 脉冲输出数目（绝对位置）

S_2 ： 脉冲输出频率

D_1 ： 脉冲输出装置

D_2 ： 脉冲方向输出装置

指令说明：

1. **S_1** 指定脉冲输出数目（绝对寻址），可指定范围为-2,147,483,648 ~ +2,147,483,647 个，其中正负号代表正反方向。

   当指令启动时，指令将自动比对 **S_1** 与当前位置，若比对 **S_1=0** 时，则不输出任何脉冲且设定完成标志为 ON。

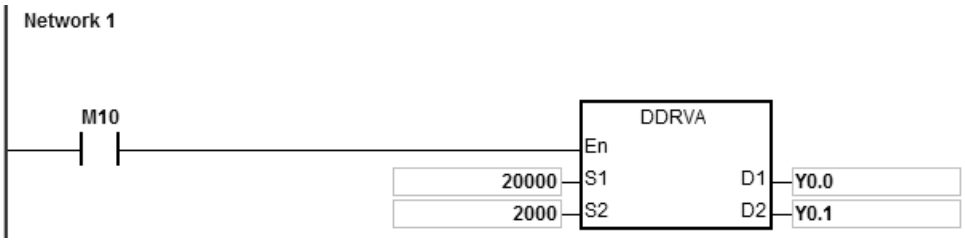
2. **S_2** 指定脉冲输出频率，使用集电极输出机种时，可指定范围为 Hz ~ +200k Hz，使用差动输出机种时，可指定范围为 Hz ~ +4M Hz。当 **S_2** 输出频率设定低于启动/停止频率(请参考 DRVI 指令之 SR 列表)时，PLC 将自动限制在启停频率。

此绝对定位是指输出运行到指定之输出当前位置为其目标位置。例如：输出当前位置为 100，而 **S_1** 目标位置设定为 1000，则此指令实际输出的个数为 1000-100=900 个。

其余相关说明请参考上一个 DDRVI 指令说明。

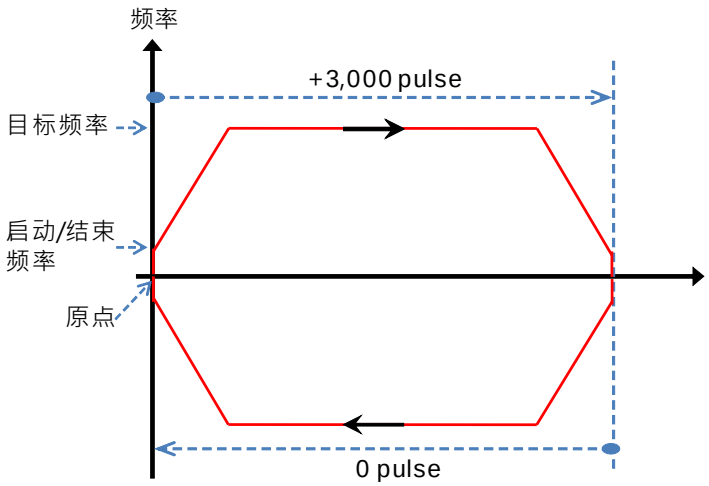
程序范例：

假设 SR460 输出当前位置（32 位）为 100，当 M10=ON 时，DDRVA 指令将以 2kHz 速度从 Y0.0 输出脉冲，并且输出直到 SR460 输出当前位置为 20,000（绝对指定），Y0.1=OFF 表示为正方向。



补充说明：

1. 绝对定位方式的动作说明： 指将自圆点（0 点）开始的距离指定的方式，也可以叫做绝对寻址方式。



API	指令码	操作数	功能
2708	CSFO	S_1 · S_2 · S_3 · S_4 · S_5 · D_1 · D_2	速度追随

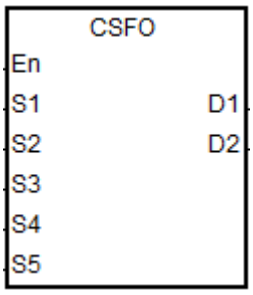
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	○															
S_2								●								
S_3								●								
S_4								●								
S_5								●								
D_1		○														
D_2								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2			●				●						
S_3			●				●						
S_4		●											
S_5		●											
D_1	●												
D_2			●				●						

脉冲执行型	16 位指令	32 位指令
-	AS	-

6

符号：



S_1 ： 输入点来源

S_2 ： 输入脉冲个数显示

S_3 ： 输入速度显示

S_4 ： 输入与输出速度比率乘数设定

S_5 ： 输入与输出速度比率除数设定

D_1 ： 脉冲输出装置

D_2 ： 输出速度显示

指令说明：

1. **S_1** 输入点来源仅可选择 X0.0, X0.2, X0.4, X0.6, X0.8, X0.10 这 6 点，且会占用连续 2 点输入，选择非指定输入点时皆无法执行。当选择好输入点，其高速计数器也将自动被指定，若用户有写相同高速计数器之 DCNT 指令，则 PLC 将以先启动计数器之指令为优先执行。这 6 个输入点所占用输入点与对应之高速计数器如下表所示：

输入组别	1	2	3	4	5	6
S_1+0 输入点(A 相)	X0.0	X0.2	X0.4	X0.6	X0.8	X0.10
S_1+1 输入点(B 相)	X0.1	X0.3	X0.5	X0.7	X0.9	X0.11
高速计数器编号	HC202	HC206	HC210	HC214	HC218	HC222
输入反向标志	SM270	SM271	SM272	SM273	SM274	SM275

2. 当此指令之高速计数器只能使用于 A/B 相输入模式，当连接手摇轮之后，在没有转动状况下，PLC 的输入点灯号为 ON 时，请设定输入反向标志，通知 PLC 检测有效工作周期改为 OFF 状态。
3. **S_2** 为显示输入脉冲个数，请采用 32 位变量宣告。
4. **S_3** 为显示输入脉冲频率，请采用 32 位变量宣告，其基本单位为 1Hz。
5. **S_4** 为输入与输出比率值乘数，**S_5** 为输入与输出比率值除数，**S_4**、**S_5 数值范围** 1~255，超过此范围自动最大或最小值处理。输出脉冲频率=输入脉冲频率* **S_4/ S_5**。举例：输入与输出为 5:3，则输入乘数为 K3，除数为 K5，若比率为 1:2，则输入乘数为 K2，除数为 K1。
6. **D_1** 将输出装置仅可选择 Y0.0, Y0.2, Y0.4, Y0.6, Y0.8, Y0.10 这 6 点，且会占用连续 2 点输出。其输出组合与输出模式 SR 如下表所示：

输出轴编号	1	2	3	4	5	6
D_1+0 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
D_1+1 输出点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
输出模式选择	SR462	SR482	SR502	SR522	SR542	SR562

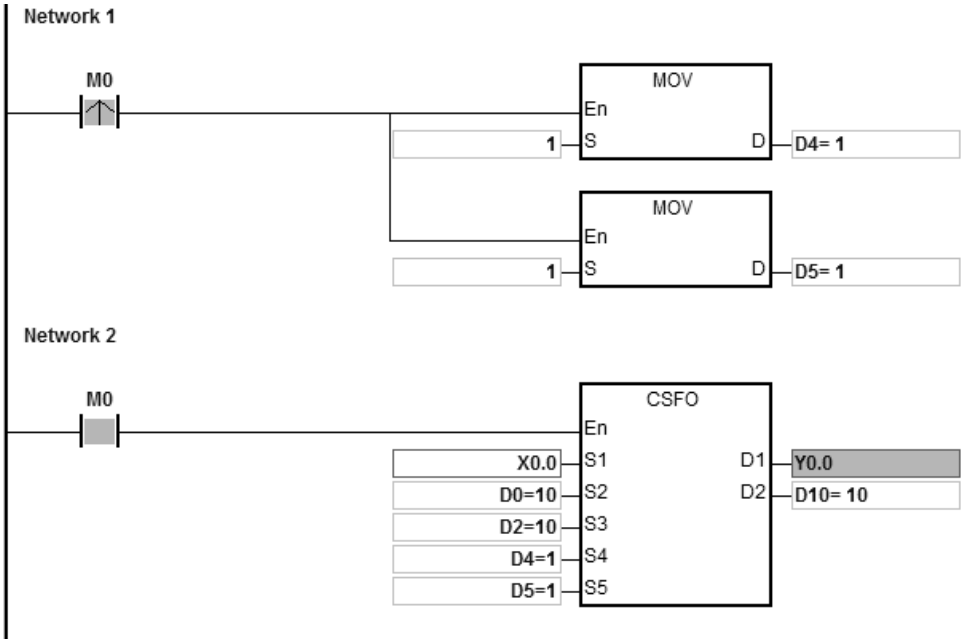
7. **D_2** 为显示输出脉冲频率，请采用 32 位变量宣告，其基本单位为 1Hz。
8. 此指令无使用次数限制，但每次指令启动时，其高速输入与输出，皆不可被其它指令启动占用，否则将会无法执行。
9. 此指令不建议使用于 ST 语言编辑、中断程序以及仅被执行一次性的功能块编辑。

注意事项：

1. 输入脉冲频率是以输入脉冲之正半周脉冲宽度（ON）为计算依据，若脉冲 ON 与 OFF 宽度不为 1:1 时，默认还是以 ON 宽度为准。
2. 输入脉冲 ON 即是指输入点灯亮，若搭配手摇轮使用时，请注意没有转动时，输入点灯号状态是否为 OFF，是否需要启动输入反向标志。
3. 韧体版本 V1.04.70(含) 版以上，此指令最低输入与输出频率皆限制为 50Hz，举例：当实际输入频率为 10Hz，其 **S_3** 显示为 10Hz，但指令内部将自动认定为 50Hz，然后经过比率(1:2)转换之后得到 100Hz 输出频率。若是比率为(2:1)转换，则输出频率还是为 50Hz(非 25Hz 或 5Hz)。
4. **S_4** 与 **S_5** 比率值允许在指令执行中进行变更，但是一旦变更之后，其输入 **S_2** 显示个数将无法依照比率值换算出该轴的 SR 当前输出位置。
5. 当比率换算之后的输出频率超出该 PLC 所允许的最高输出频率时，此状况将会被记录错误代码为 16#2030 于错误历史记录内，并且设定 SM29=ON 与 SR29=该轴编号。韧体版本 V1.06.20(含)版以上支持。
6. 当输入频率为 0 并且输出脉冲还持续输出达 2 秒以上时，PLC 将自动关闭输出，然后记录错误代码 16#2031 于错误历史记录内，并且设定 SM29=ON 与 SR29=该轴编号。韧体版本 V1.06.20(含)版以上支持。

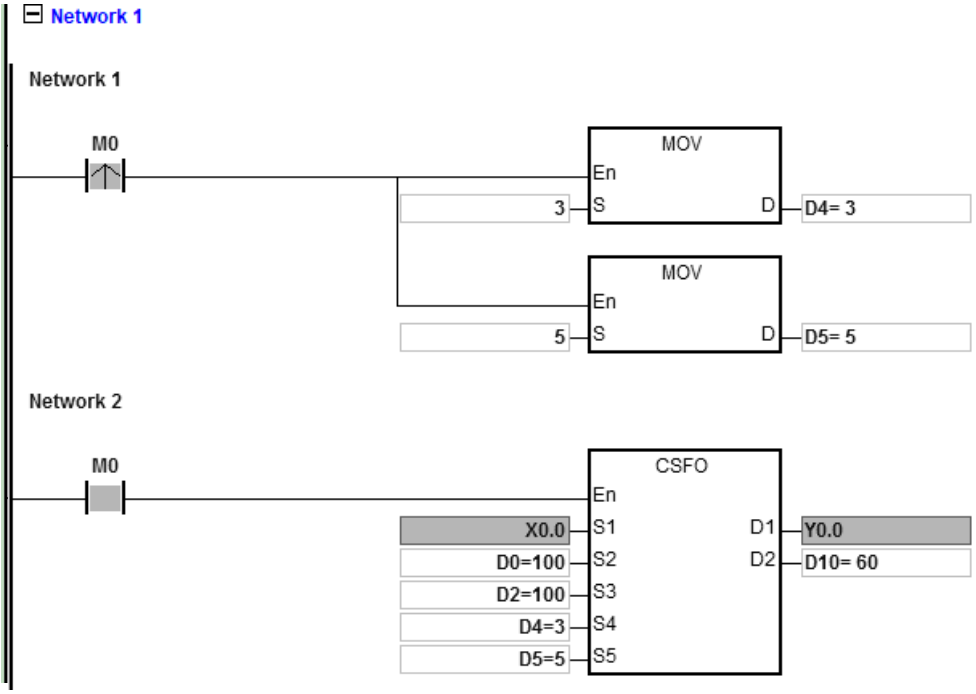
程序范例一：

X0.0/X0.1 为检测手摇轮所输入的脉冲，当 M0=ON 时，输入与输出速度比率乘数/除数的设定值皆为 1（D4=1，D5=1），当手摇轮所输入的速度为 10Hz（D2，D3=10）及脉冲个数为 10 时（D0，D1=10），则输出轴（Y0.0/Y0.1）将会以 10Hz（D10，D11=10）输出 10 个脉冲（SR460，SR461=10）。



程序范例二：

当 M0=ON 时，输入与输出速度比率乘数/除数的设定值各为 3 及 5 （D4=3，D5=5），当手摇轮输入的速度为 500Hz（D2，D3=500）及脉冲个数为 2500 时（D0，D1=2500），则输出轴（Y0.0/Y0.1）将会以 300Hz（D10，D11=300）输出 1500 个脉冲（SR460，SR461=1500）。



API	指令码			操作数	功能
2709	D	DRVM		**S_1 · S_2 · S_3 · S_4 · S_5 · S_6 · D_1 · D_2**	对标型定位输出

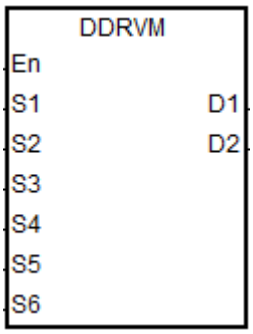
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3	○		○													
S_4								●	●				○	○		
S_5								●	●				○	○		
S_6								●	●				○	○		
D_1		○														
D_2		○	○													

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3	●												
S_4			●				●						
S_5			●				●						
S_6			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

6

符号：



S_1 ： 目标输出总个数

S_2 ： 目标输出频率

S_3 ： 外部中断对标输入点编号

S_4 外部输入中断前段遮蔽结束的输出个数

S_5 外部输入中断后段遮蔽开始的输出个数

S_6 对标中断产生后之指定减速输出个数

D_1 ： 脉冲输出装置

D_2 方向脉冲输出装置

指令说明：

1. 当 **S_1** 输出总个数输入为 0 时，即表示以最长 32 位数值 (有号数) 输出，直到对标发生，再减速输出停止。若输出个数不为 0，并且外部中断未发生，则输出目标总个数。**S_1** 指定脉冲输出总个数，可指定范围为 -2,147,483,648 ~ +2,147,483,647 个，其中正负号代表正反方向。

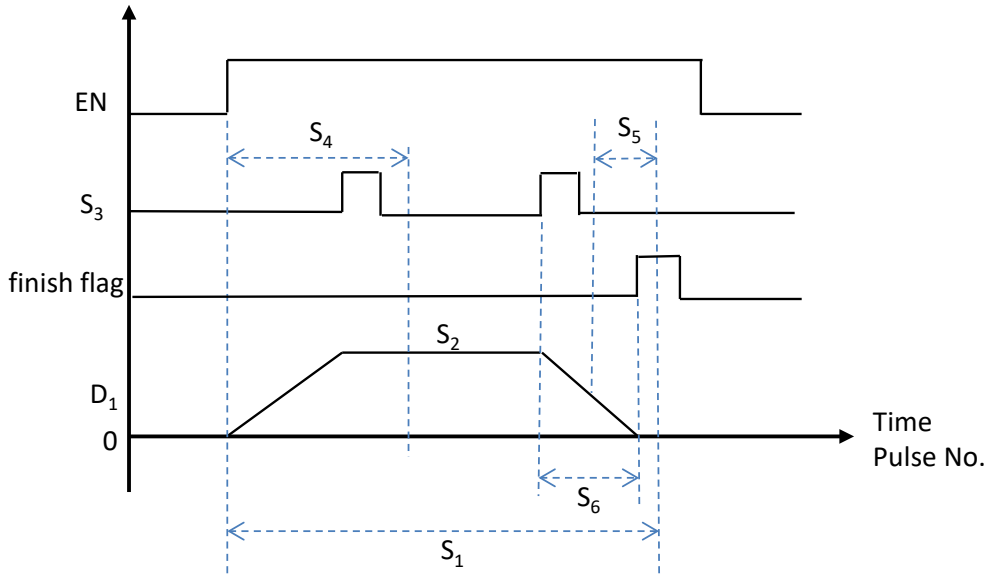
2. 当目标输出频率 **S_2** 输入小于等于 0 时，表示不启动输出；当输出频率大于最高频率限制时，则以最高频率输出。
3. 当 **S_3** 输入点可以选择 M 或 X 装置，当选择不是主机内建 X0.0~X0.15 输入点编号时，此输入来源信号将会受扫描时间影响。
4. 使用此指令时，必须依据输入点编号搭配外部输入中断程序，才可达到实时执行减速输出的动作。此外部中断上升沿或下降沿触发之动作，请于编辑外部中断程序时一同选择。
   A.有使用外部中断（如：I0xx, I1xx）：当外部中断发生时，执行对标减速停止。若外部中断未开启，且无动作。
   B.没有使用外部中断：执行指令时，若 X 点与 M 装置有上升沿变化，则执行对标减速停止，注意此触发动作将会受扫描时间影响。（注：不需搭配中断程序）
5. 前段遮蔽区域为输出个数 1 到 **S_4** 之间，当设定为 0 时，表示不启动前段遮蔽功能；当遮蔽个数超过输出总个数 **S_1** 时，指令将以输出总个数为遮蔽个数条件。当外部输入触发动作发生在遮蔽个数内，则此外部输入中断将自动失效。
6. 后段遮蔽区域为输出个数 **S_5** 到总输出个数 **S_1** 之间，当 **S_5** 设定为 0 或者>=**S_1** 时，则表示不启动后遮蔽功能。当外部输入触发动作发生在遮蔽区域内，则此外部输入中断将自动失效。当 **S_4** 的数值大于 **S_5** 时，即表示指令输出过程中外部输入中断无效。
7. 若前段与后段遮蔽区域皆有设定时，且 **S_4** < **S_5** 设定数值，则表示输入中断有效区域为 **S_4**+1 ~ **S_5**-1 之间。
8. V1.04.00 版（含）以前，当对标后指定输出个数 **S_6** 设定 0 时，表示减速停止动作要以设定的减速时间执行；当对标后指定输出个数 **S_6** 设定-1 或< 0 时，表示停止动作以立即停止方式执行；若是指定个数 **S_6** 不足以减速时间内减速停止，则指令将会限制目标速度，让减速停止的输出动作符合减速时间的要求。若指定个数 **S_6** 大于减速时间所需之输出个数，则指令将以中断触发时的输出速度送出多余的输出个数，最后再进行减速动作。
9. V1.06.00 版（含）以后，当对标后指定输出个数 **S_6** 设定 0 时，表示减速停止动作要以设定的减速时间执行；当对标后指定输出个数 **S_6** 设定-1 时，表示停止动作以立即停止方式执行；当对标后指定输出个数 **S_6** 设定> 0 时，若是指定个数 **S_6** 不足以减速时间内减速停止，则指令将会限制目标速度，让减速停止的输出动作符合减速时间的要求，若指定个数 **S_6** 大于减速时间所需之输出个数，则指令将以中断触发时的输出速度送出多余的输出个数，最后再进行减速动作；当对标后指定输出个数 **S_6** 设定< -1 时，对标减速过程以指定个数为优先，实际减速输出个数请去除负号，若是指定个数 **S_6** 不足以减速时间内减速停止，则不限制最高频率，若指定个数 **S_6** 大于减速时间所需之输出个数，则指令将以中断触发时的输出速度送出多余的输出个数，最后再进行减速动作。

减速输出个数 **S_6**	功能说明
>0	指定对标减速过程顺畅由优先，指令自动限制目标速度。 A. **S_6** 大于减速时间所需之输出个数，则指令将以中断触发时的输出速度送出多余的输出个数，最后再进行减速动作 B. **S_6** 不足以减速时间内减速停止，则指令将会限制目标速度，让减速停止的输出动作符合减速时间的要求
=0	指定对标减速过程以减速时间为主 **S_6** 设定 0 时，表示减速停止动作要以设定的减速时间执行

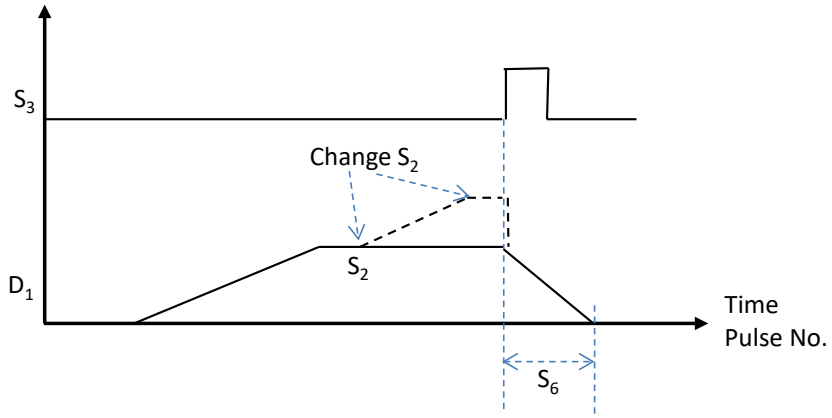
减速输出个数 **S_6**	功能说明
=-1	**S_6** 设定-1 时，表示停止动作以立即停止方式执行
<-1	对标减速过程以指定个数为优先，实际减速输出个数请去除负号 A. **S_6** 大于减速时间所需之输出个数，则指令将以中断触发时的输出速度送出多余的输出个数，最后再进行减速动作 B. **S_6** 不足以减速时间内减速停止，则不限制目标速度，输出完毕 **S_6** 脉冲后，立即停止

- SR652/SR653=加速区脉冲个数显示；SR654/SR655=减速区脉冲个数显示

10. 其输出、中断触发、遮蔽与相关标志如下列时序图中所示：（图中 finish flag 请参考 DDRVI 指令之各轴输出完成标志 SM）



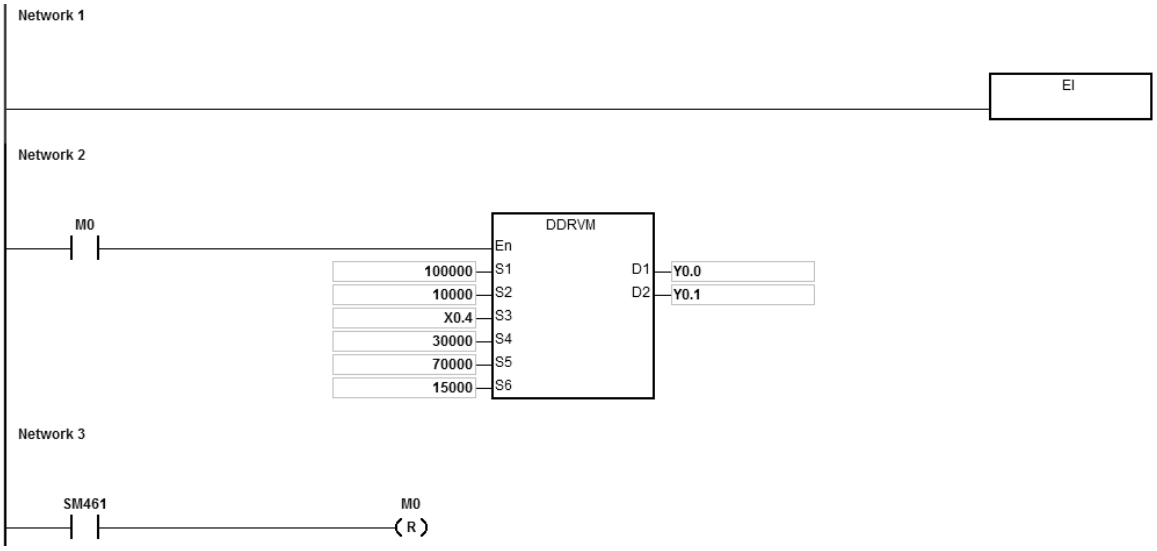
11. 当输出过程当中，此指令允许临时变更目标速度 **S_2**，但由于减速区段的输出个数，已经在指令启动时就已被设定，因此若有变更速度的时序图将如下图虚线所示：



12. **D_1** 与 **D_2** 脉冲输出装置之选择请参考 DDRVI 指令说明，但此指令 **D_1** 仅可使用偶数编号的 Y 输出点（如：Y0.0，Y0.2…Y0.10）；**D_2** 脉冲方向输出装置，若指定不为建议搭配之输出点，则方向输出将受扫描时间影响。

13. 此指令不**建议**使用于 ST 语言编辑、中断程序以及仅被执行一次性的功能块编辑

程序范例 1：



X0.4 外部中断程序

说明：

1. 当 M0 由 OFF 变 ON 时，Y0.0 开始输出脉冲，等待输出个数超过 30,000 个后，外部中断 X0.4 进来，D0 会加 1，开始行走 15,000 个脉冲后停止输出（S_6=15,000）。如果中断都未发生，则脉冲输出 100,000 才停止输出
2. 如果脉冲输出个数在 1~30,000，则属于前段遮蔽区域，此时外部中断 X0.4 进来，将不会执行减速停止动作。
3. 如果脉冲输出个数在 70,000~100,000 个，则属于后段遮蔽区域，此时外部中断 X0.4 进来，一样不会执行减速停止动作。
4. 脉冲输出完毕 SM461=ON，清除 M0

补充说明：

1. 对标功能及遮蔽功能作说明：

   ① 启动 DDRVM 功能，开始输出脉冲

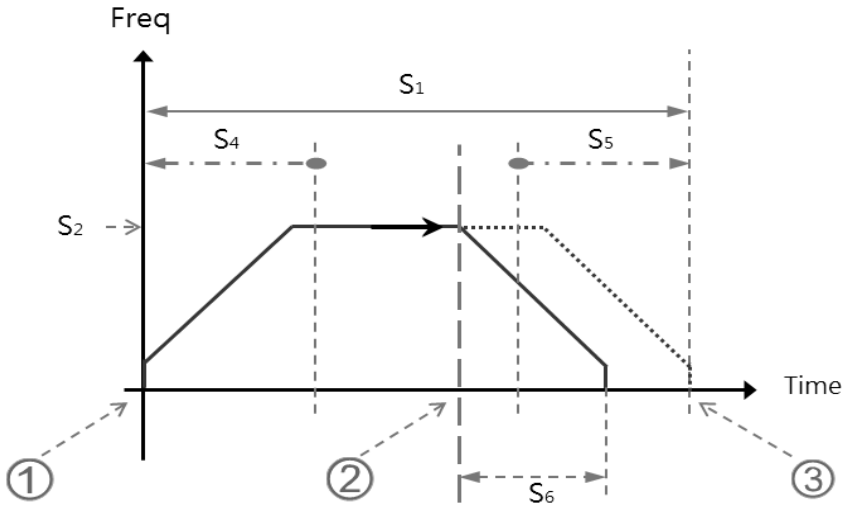
   ② 对标中断产生，开始执行减速动作，当输出完 **S_6** 指定减速输出个数脉冲后停止

   ③ 当对标中断未产生或无效时，DDRVM 则依照 **S_1** 目标输出总个数输出完毕后停止

   S_4:为前段遮蔽的输出个数，当对标中断在此区间发生时，则此次对标中断无效

   S_5:为后段遮蔽的输出个数，当对标中断在此区间发生时，则此次对标中断无效

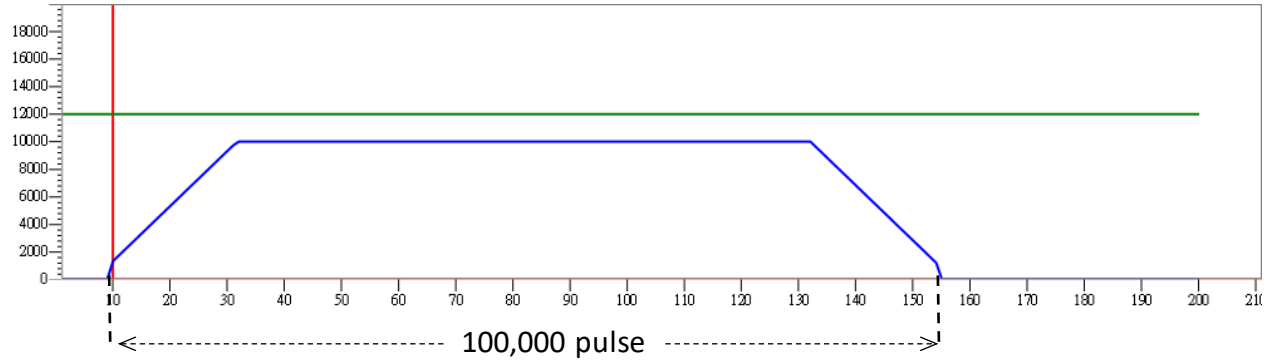
   S_2:为目标输出频率



2. 对标功能发生在加速区、全速区及减速区的动作说明：

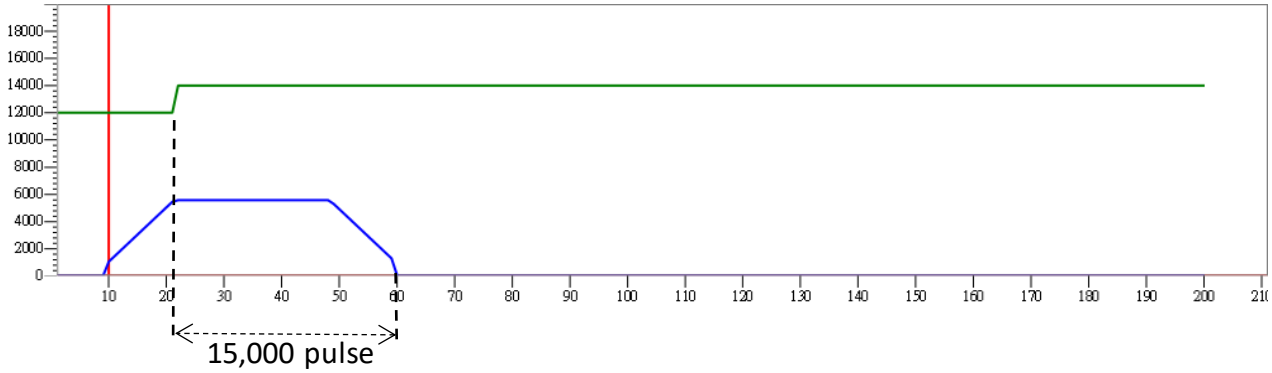
未产生对标动作时： (下图纵轴为频率值，横轴为时间，绿色线表示对标信号，蓝色线表示实际输出速度)

对标动作未发生，脉冲输出 S_1=100,000 个脉冲之后停止



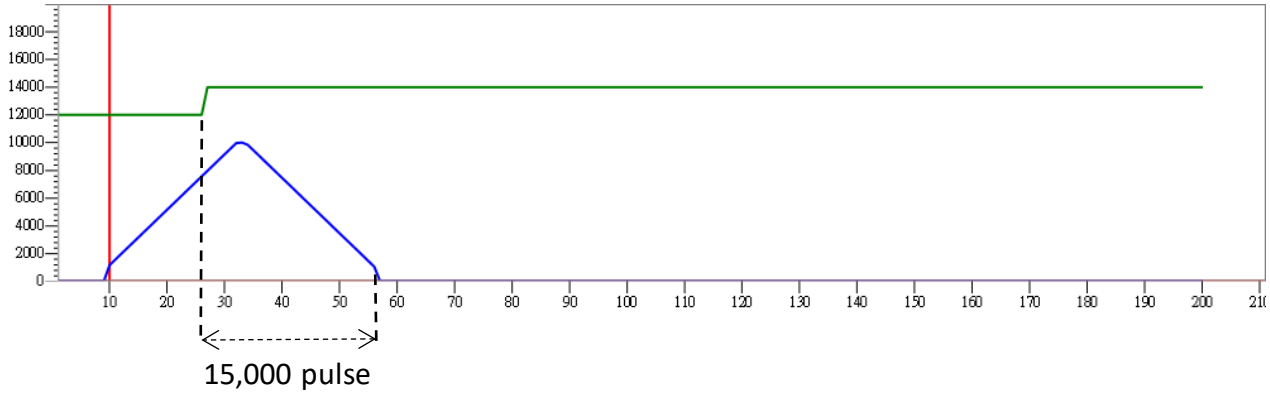
当对标动作于加速区域发生时之状况一：(下图纵轴为频率值，横轴为时间，绿色线表示对标信号，蓝色线表示实际输出速度)

对标动作(绿色线 Off→On)在脉冲个数为 3000 发生，虽然输出 S_6=15,000 个脉冲，但输出过程中无法达到目标速度 10kHz



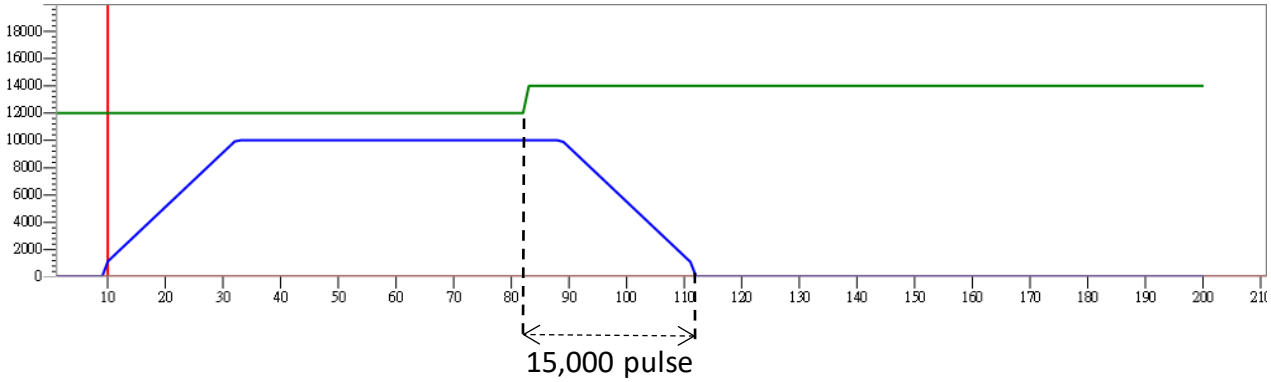
当对标动作于加速区域发生时之状况二：(下图纵轴为频率值，横轴为时间，绿色线表示对标信号，蓝色线表示实际输出速度)

对标动作(绿色线 Off→On)在脉冲个数为 6000 发生 ,输出 **S_6**=15,000 个脉冲 ,并且过程中可达到目标速度 10kHz



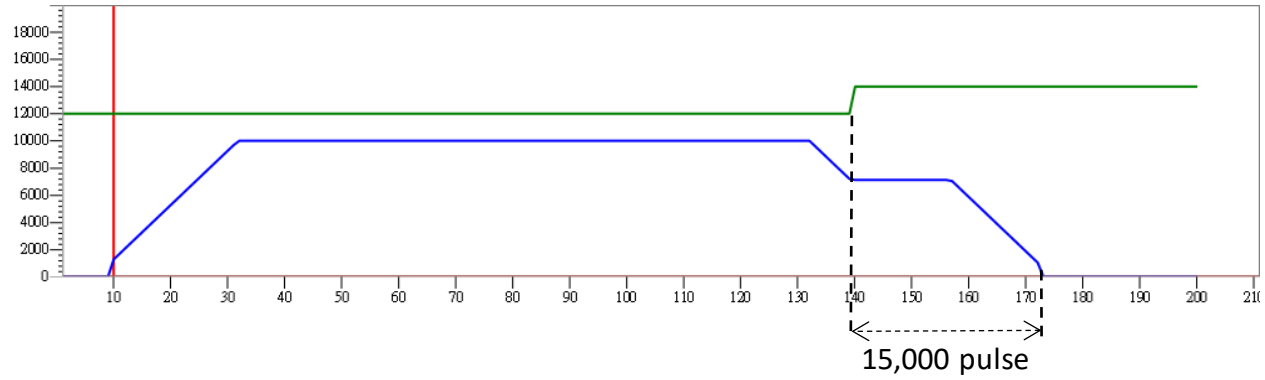
当对标动作于目标速度区发生时：(下图纵轴为频率值，横轴为时间，绿色线表示对标信号，蓝色线表示实际输出速度)

对标动作(绿色线 Off→On)在脉冲个数为 50,000 发生，输出完 **S_6**=15,000 个脉冲后停止。



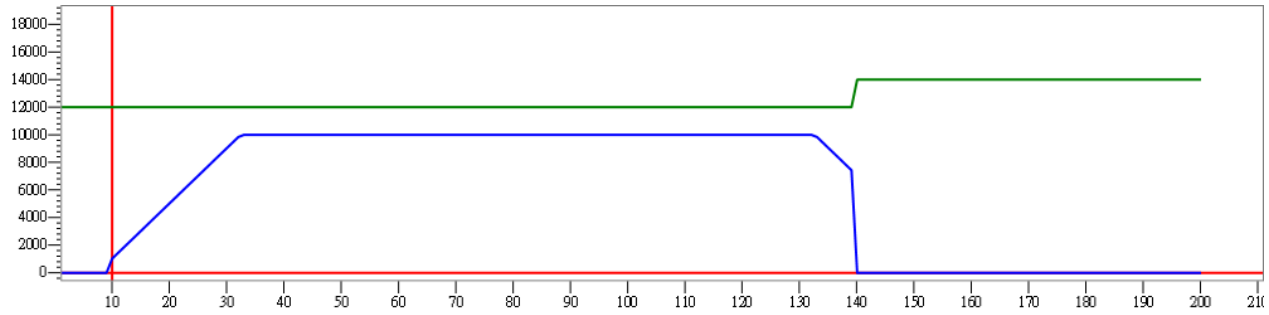
当对标动作于减速区域发生时之状况一：(下图纵轴为频率值，横轴为时间，绿色线表示对标信号，蓝色线表示实际输出速度)

对标动作(绿色线 Off→On)在脉冲个数为 95,000 发生，以当下输出速度，输出完 **S_6**=15,000 个脉冲后停止。



当对标动作于减速区域发生时之状况二：(下图纵轴为频率值，横轴为时间，绿色线表示对标信号，蓝色线表示实际输出速度)

对标动作(绿色线 Off→On)在脉冲个数为 95,000 发生，因 **S_6** 对标后输出个数设为 -1，于是立即停止输出


18000
16000
14000
12000
10000
8000
6000
4000
2000
0
10
20
30
40
50
60
70
80
90
100
110
120
130
140
150
160
170
180
190
200

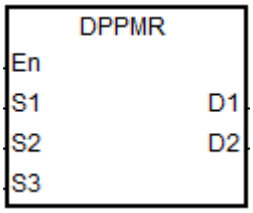
API	指令码			操作数	功能
2710	D	PPMR		**S_1，S_2，S_3， D_1，D_2**	相对地址点对点双轴同动输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
D_1		○														
D_2		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： X 轴脉冲输出数目

S_2 ： Y 轴脉冲输出数目

S_3 ： 点到点之间的最高输出频率

D_1 ： X 轴脉冲输出装置

D_2 ： Y 轴脉冲输出装置

指令说明：

1. **S_1** 与 **S_2** 分别代表 X 轴与 Y 轴指定脉冲输出数目（相对定位），其输出数目范围为-2,147,483,648~ +2,147,483,647 个，其中正负号代表正反方向。
2. **S_3** 指定点到点之间的最高输出频率，指定范围为 1Hz ~ +200k Hz。
3. V1.08.70 版（含）以上，支持执行到最高输出频率时，用户可变更最高输出频率。依指令启动时，计算的间隔频率和间隔时间，执行变更最高输出频率后的加减速。由于加减速区域在指令启动时，就已经设定好预计执行的加减速斜率，因此实际变更速度后，减速区的动作，请参考 DPLSR 指令说明。
4. **D_1** 与 **D_2** 分别 X 轴与 Y 轴脉冲输出装置，可任意指定下表这 6 轴输出，但内定搭配方向输出点不可变更。此方向信号定义 OFF 为正方向，ON 为反方向。若不采用默认的 ”脉冲+方向” 输出模式（默认 0），则请变更输出模式选择 SR 为 1。

轴编号	轴 1	轴 2	轴 3	轴 4	轴 5	轴 6
D_1 与 **D_2** 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10

内定搭配方向输出点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
输出模式选择	SR462	SR482	SR502	SR522	SR542	SR562

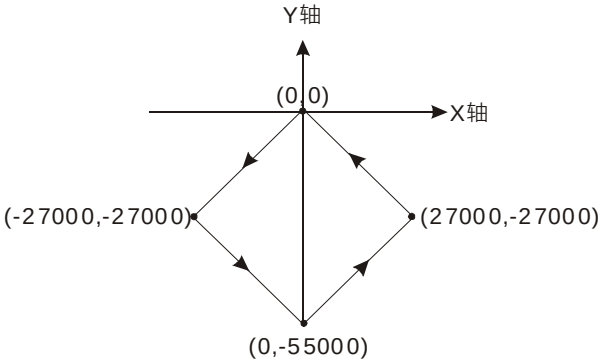
5. **S_3** 最高输出频率 PLC 将自动配置给 X 或 Y 轴输出最多个数的其中一轴 但若是这两轴输出个数相差过大，无法换算出合理的输出频率时，PLC 将自动调降指定的最高输出频率，且不会显示任何警报信息。
6. 此指令输出过程会搭配各轴启动/结束频率以及加减速时间的参数，并且都是以 X 轴为参考参数来源，而不会以各轴原本配置的参数。举例：当 X 轴选择轴 1，Y 轴选择轴 3 输出，则启动速度与加减速时间都会以轴 1 的参数为来源数值。
7. 各轴标志与参数对应之 SM/SR 如下表所示。

轴编号	轴 1	轴 2	轴 3	轴 4	轴 5	轴 6
输出中标志	SM460	SM480	SM500	SM520	SM540	SM560
输出完成标志	SM461	SM481	SM501	SM521	SM541	SM561
输出当前位置	SR460 SR461	SR480 SR481	SR500 SR501	SR520 SR521	SR540 SR541	SR560 SR561
启动/结束频率	SR463	SR483	SR503	SR523	SR543	SR563
加速时间	SR464	SR484	SR504	SR524	SR544	SR564
减速时间	SR465	SR485	SR505	SR525	SR545	SR565

8. 此指令无使用次数限制，但指令刚启动时，假设 Y 轴输出已被使用中，则指令将会无法输出，须等到 Y 轴已经完成且被释放后，才会开始执行。
9. 当两轴同动输出结束时，其输出完成标志将分别设定各轴对应之完成标志，由于此同动完成的时机无法保证每次都一定能同时完成，因此建议用户程序，最好同时判断两轴对应的输出完成标志之后，再往下一流程执行后续动作。

程序范例一（梯形图编辑）:

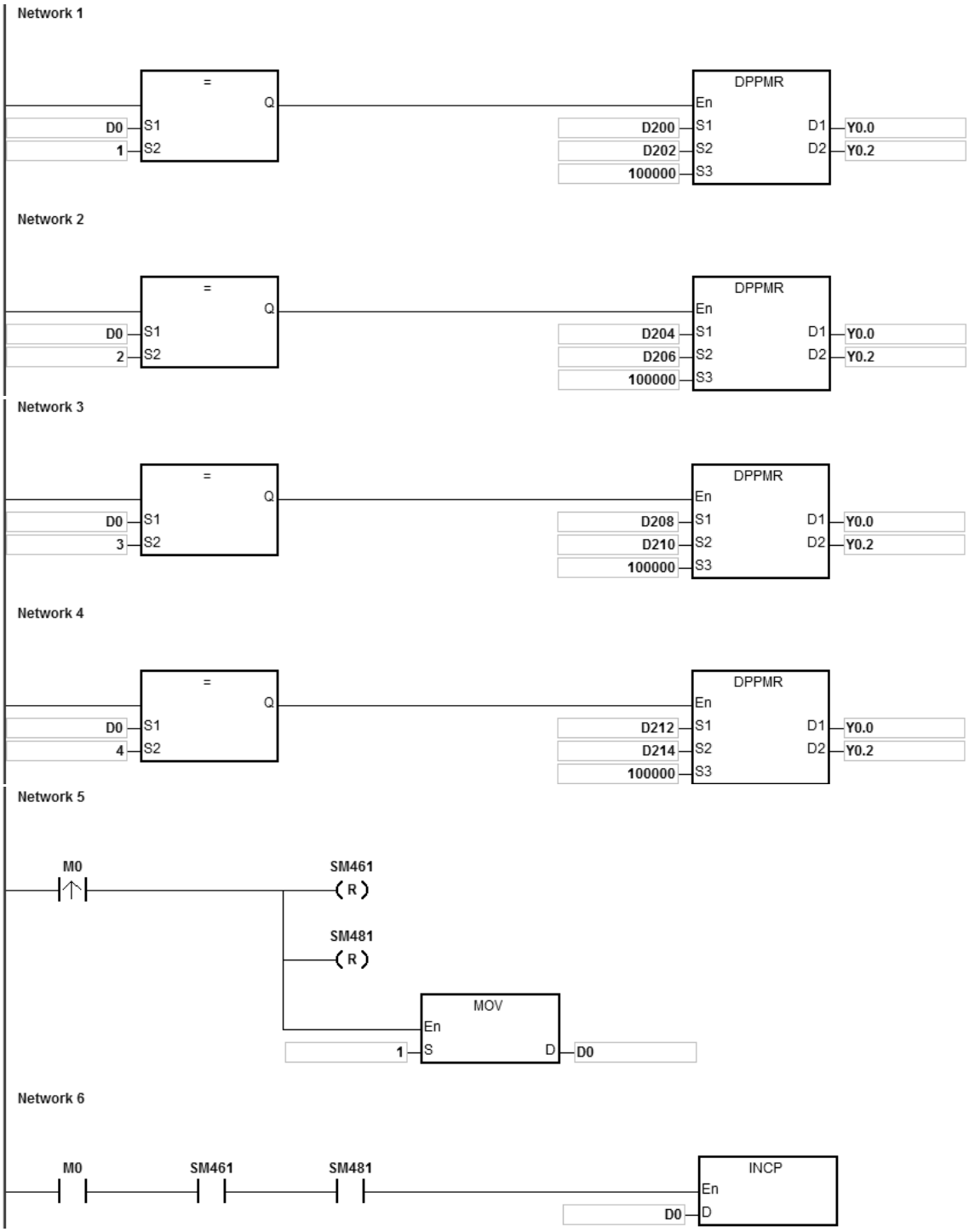
1. 画一个菱型如下图



2. 操作步骤：

- 规划四点的坐标如下（0，0）、(-27000，-27000)、(0，-55000）及（27000，-27000）如上图，计算四点的相对地址的坐标如下(-27000，-27000)、(27000，-28000)、(27000，28000)及(-27000，27000)，分别放在 32bit（D200，D202）、(D204，D206)、(D208，D210)、(D212，D214)。

- PLC RUN，并设定 M0 为 ON，则开始两轴画线。



3. 动作说明：

当 PLC RUN，M0=ON 时，开始以频率 100kHz 执行第一段点对点运动，在每一段点对点运动结束后，D0 加 1，自动执行第二段点对点运动，以此类推，直到执行完第四段点对点运动。

程序范例二（ST 结构化语言编辑）：

```
0001 IF M0 THEN
0002    DPPMR(1000,1000,1000,Y0.0,Y0.2);
0003    M1 := TRUE;
0004 ELSIF M1 AND SM461 AND SM481 THEN
0005    SM470 := TRUE;
0006    SM490 := TRUE;
0007    M1 := FALSE;
0008 END_IF;
```

动作说明：

1. 当 M0=ON 时，Y0.0 与 Y0.2 同时输出 1000 个 1kHz 频率脉冲，并且设定 M1=ON，用来记录 Y0.0 与 Y0.2 高速控制权已经被使用。
2. 当 Y0.0 输出完毕 SM461=ON，以及 Y0.2 输出完毕 SM481=ON，并且 M1=ON 时，触发 Y0.0 的自动复位标志 SM470=ON 以及 Y0.2 自动复位标志 SM490=ON，通知 PLC 释放高速输出控制权。此时也顺带清除 M1。
3. 当 M0 再次由 OFF>ON 时，脉冲输出即会重新启动。
4. 以上程序若有 2 个高速输出程序以上时，请勿重复使用 M1 这个临时性的变量名称。
5. 若输出过程中有可能遇到要求暂停输出的动作，则不建议采用 ST 语言编辑。

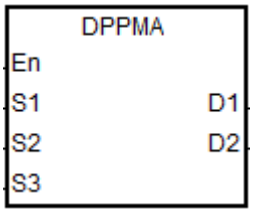
API	指令码			操作数	功能
2711	D	PPMA		**S_1，S_2，S_3， D_1，D_2**	绝对地址点对点双轴同动输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
D_1		○														
D_2		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： X 轴脉冲输出数目

S_2 ： Y 轴脉冲输出数目

S_3 ： 点到点之间的最高输出频率

D_1 ： X 轴脉冲输出装置

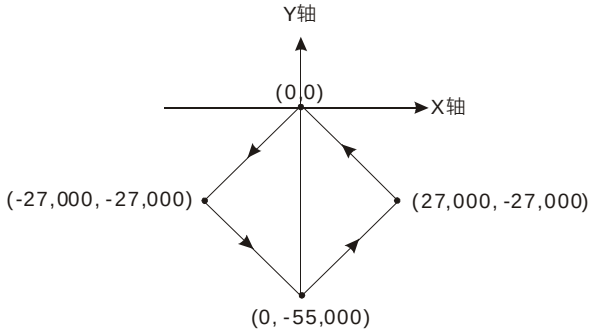
D_2 ： Y 轴脉冲输出装置

指令说明：

1. **S_1**、**S_2** 分别代表 X 轴与 Y 轴指定脉冲输出数目（绝对地址）。
2. 其余相关指令、各轴参数与标志说明，以及 ST 结构化语言编辑，请参照前一个 DPPMR 指令。

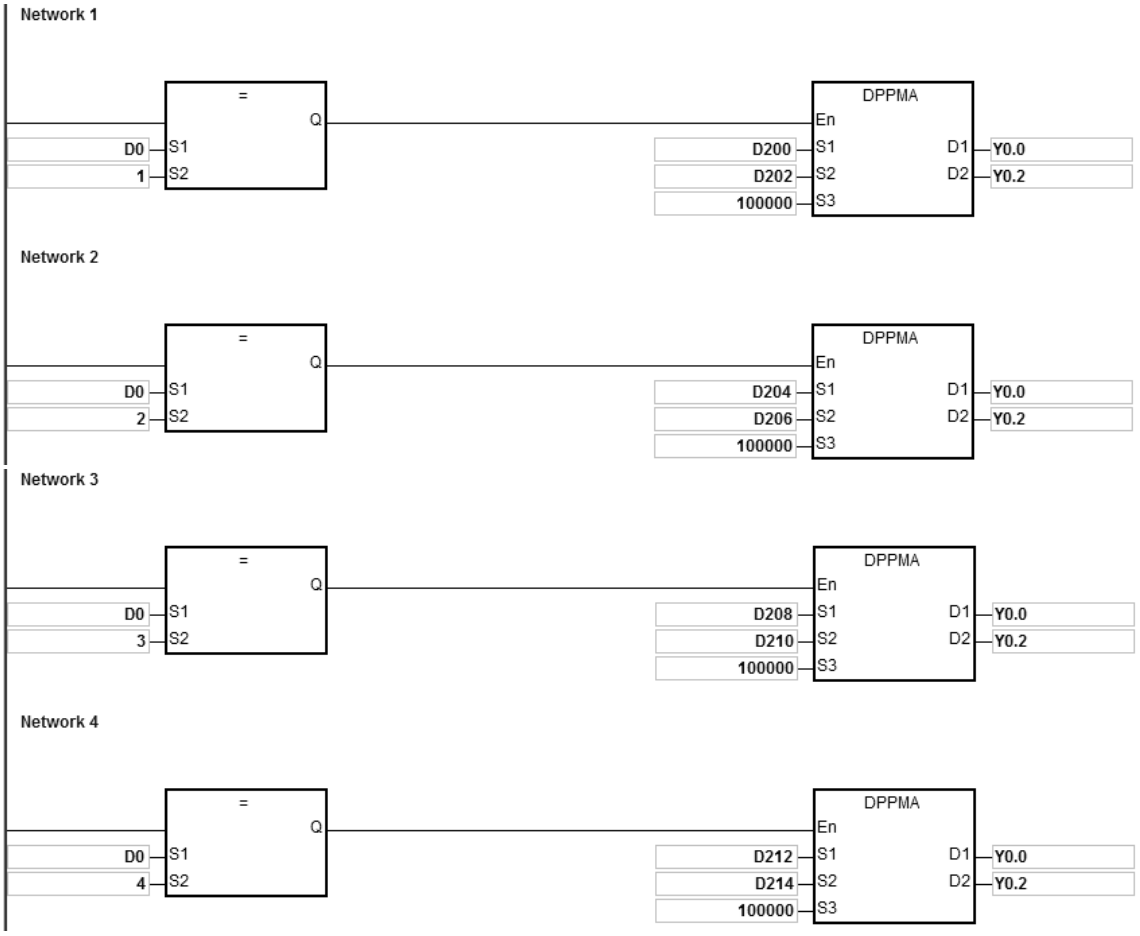
程序范例：

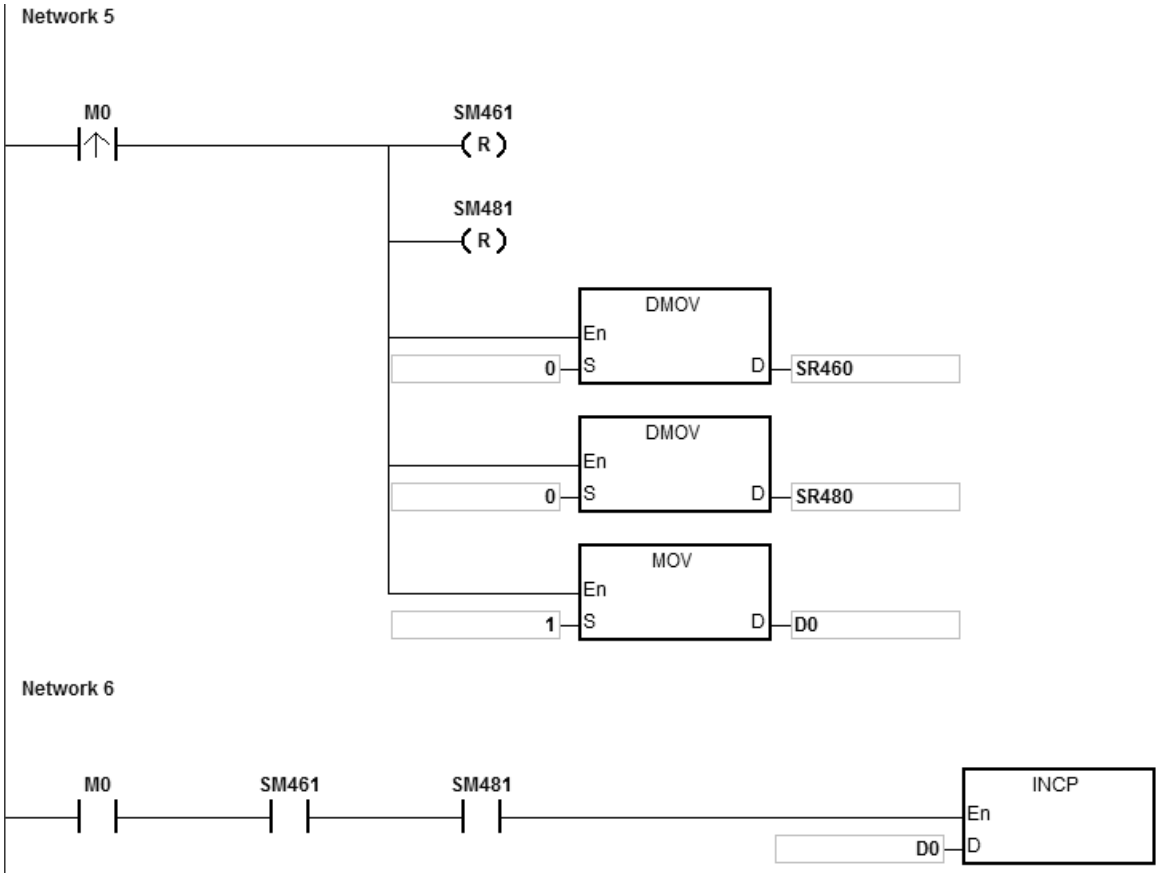
1. 画一个菱型如下图



2. 操作步骤：

- 规划四点的绝对坐标如下（-27000，-27000）、（0，-55000）、（27000，-27000）及（0，0）如上图，分别放在 32bit（D200，D202）、（D204，D206）、（D208，D210）、（D212，D214）。
- PLC RUN，并设定 M0 为 ON，则开始双轴画线。





3. 动作说明：

当 PLC RUN，M0=ON 时，开始以频率 100kHz 执行第一段点对点运动，在每一段点对点运动结束后，D0 加 1，自动执行第二段点对点运动，以此类推，直到执行完第四段点对点运动。

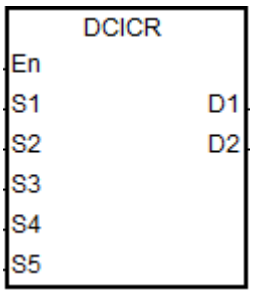
API	指令码			操作数	功能
2712	D	CICR		S_1、S_2、S_3、S_4、S_5、D_1、D_2	相对位置顺时针圆弧插补

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		●
S_4							●	●	●		○		○	○		
S_5							●	●	●		○		○	○		
D_1		○														
D_2		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●		●				
S_4			●				●						
S_5			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： X 轴目标坐标（相对位置）

S_2 ： Y 轴目标坐标（相对位置）

S_3 ： 圆心位移量或是圆心角度

S_4 ： 目标速度参考值

S_5 ： 功能选择

D_1 ： X 轴输出装置

D_2 ： Y 轴输出装置

指令说明：

1. S_1、S_2 与 S_3 分别为 X 轴、Y 轴指定目标坐标地址（相对位置）与圆心位移量。其详细动作请参考下页的顺时针圆弧动作说明。
2. D_1、D_2 分别为 X 轴与 Y 轴之指定脉冲输出装置，其各轴输出点与模式选择请参考 DPPMR 指令说明。
3. S_4 为设定目标速度之参考值，当指令启动时，PLC 会在规划圆弧行进路径时，搭配此目标速度先行计算，但若是计算过程已经预估无法达成圆弧行进效果时，则会自动帮忙降低输出速度，以达成同动画圆弧之功能为优先。

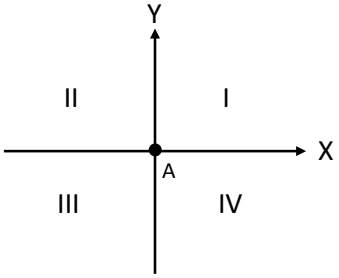
4. **S_5** 为功能选择设定值，详细说明请参考下表：

S_5 设定值	**S_3** 功用	功能说明
0	圆心偏移量(32 位整数)	以角度 10 度为基本运行角度
1	圆心偏移量(32 位整数)	以角度 5 度为基本运行角度
2	圆心偏移量(32 位整数)	以角度 1 度为基本运行角度 (韧体版本需为 V1.04.60 以上)
10	圆心角度 (浮点数)	以角度 10 度为基本运行角度 (韧体版本需为 V1.04.40 以上)
11	圆心角度 (浮点数)	以角度 5 度为基本运行角度 (韧体版本需为 V1.04.40 以上)
12	圆心角度 (浮点数)	以角度 1 度为基本运行角度 (韧体版本需为 V1.04.60 以上)
20	无效	以基本角度 10 度与目标 X，Y 坐标运行固定 90 度圆弧 (韧体版本需为 V1.10.10 以上)
21	无效	以基本角度 5 度与目标 X，Y 坐标运行固定 90 度圆弧 (韧体版本需为 V1.10.10 以上)
22	无效	以基本角度 1 度与目标 X，Y 坐标运行固定 90 度圆弧 (韧体版本需为 V1.10.10 以上)
其他	圆心偏移量(32 位整数)	皆以 **S_5** 设定值 0 执行

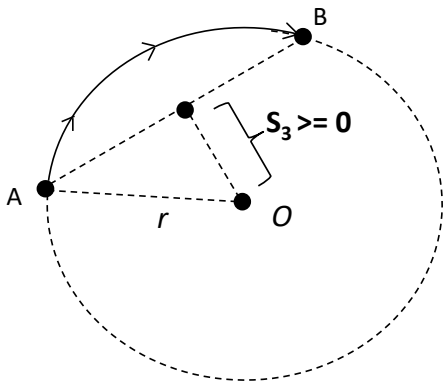
5. ST 结构化语言编辑，请参考 DPPMR 指令的范例二说明。

顺时针之圆弧动作说明 (适用 S_5 设定值 0~12) :

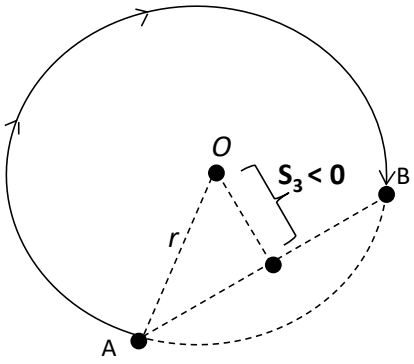
1. 定义坐标象限如下图所示。其坐标分成四个象限，分别为第 I, II, III 及 IV 象限。图中 A 表示 X,Y 轴当前坐标位置。



2. 定义 B 点为 **S_1** 与 **S_2** 目标坐标位置。定义 *O* 点为 A 与 B 两点间连成直线之中心点，也即是所谓圆心点。
3. **S_3** 圆心位移量即是指圆心 *O* 点指定之偏移量。在 V1.04.60 版 (含) 以上，若功能选择为圆心角度，**S_3 圆心角度即是∠AOB 之夹角。**
4. 当目标 B 点位于当前 A 点的第 I, IV 象限，则 **S_3** 圆心位移量设定值将执行如下两图之行进路径 (如：实线部分) ：(图中 *r* 表示半径)

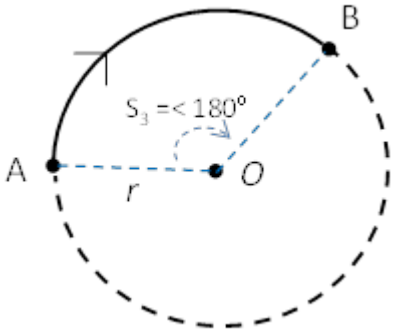


当 S_3>= 0

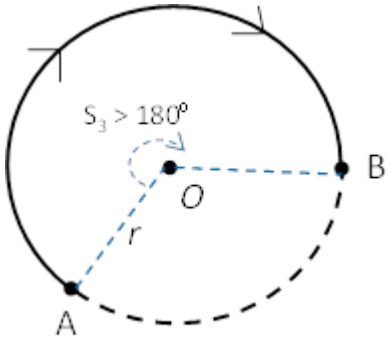


当 S_3< 0

5. 当目标 B 点位于当前 A 点的第 I 及 IV 象限，则 S_3 圆心角度设定值将执行如下两图之行进路径（如：实线部分）：（图中 *r* 表示半径）

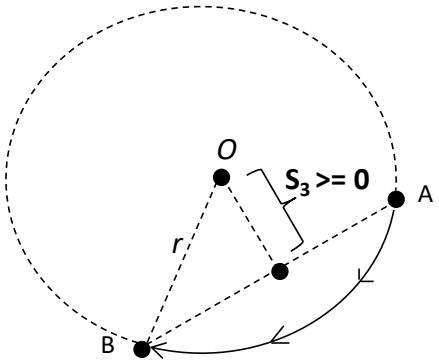


当 S_3<= 180.0

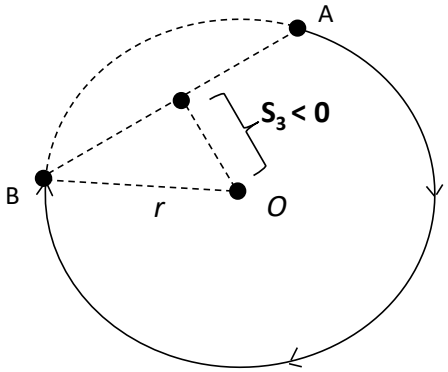


当 S_3> 180.0

6. 当目标 B 点位于现在 A 点的第 II 及 III 象限，则 S_3 圆心位移量设定值将执行如下两图之行进路径：

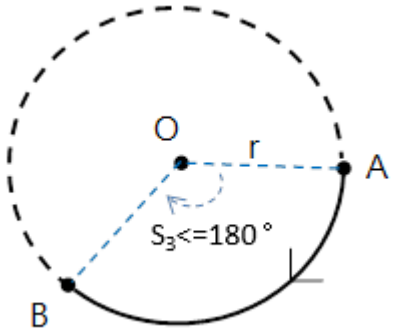


当 S_3>= 0

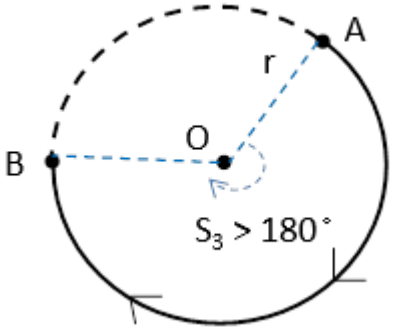


当 S_3< 0

7. 当目标 B 点位于当前 A 点的第 II 及 III 象限，则 S_3 圆心位移量设定值将执行如下两图之行进路径：



当 **S3**<=180.0

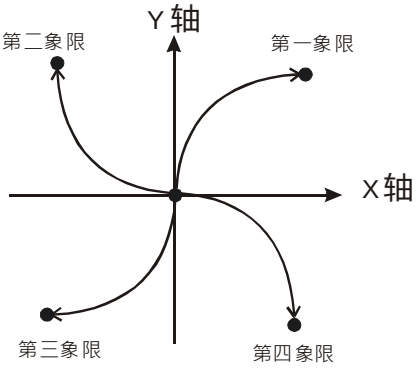


当 **S3**>180.0

8. 当目标 B 点之 X 轴为 0，若 Y 轴目标位置 >= 0 时，则此 B 点定义位于 A 点的第 I 象限；相反的，若 Y 轴目标位置 < 0，则此 B 点定义位于 A 点的第 III 象限。

固定 90 度之顺时针圆弧动作说明（适用 S5 设定值 20~22）：

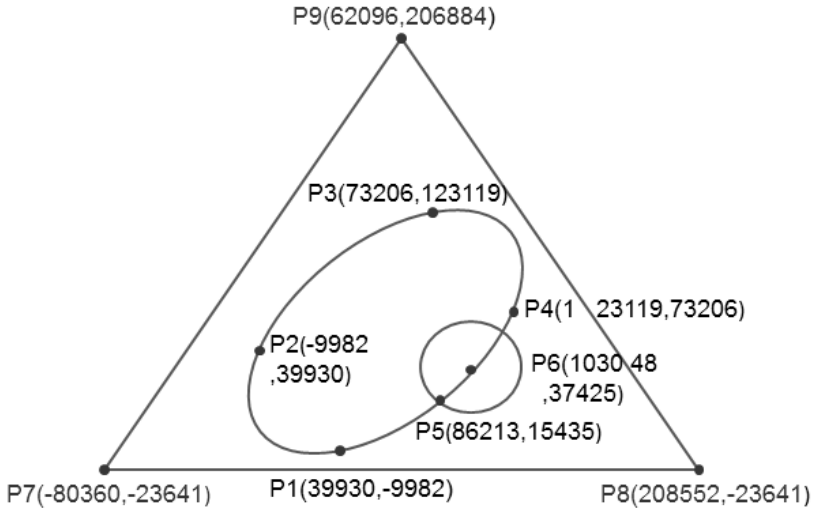
1. 坐标象限同上述模式说明，此功能将固定以顺时针方向进行 90 度圆弧运行。注：此功能需搭配韧体 V1.10.10 版（含）以上。
2. **S1** 与 **S2** 参数所指定之四个象限的目标位置，相对于当下 X 与 Y 轴现在位置（下图中心点），其运行路径将如下图所示之四种情况。



3. 由于每次都以 90 度运行，因此当 X/Y 轴的目标位置不是相同距离时，其圆弧将不会形成四分之一的正圆形，适合被使用于画椭圆形之应用。

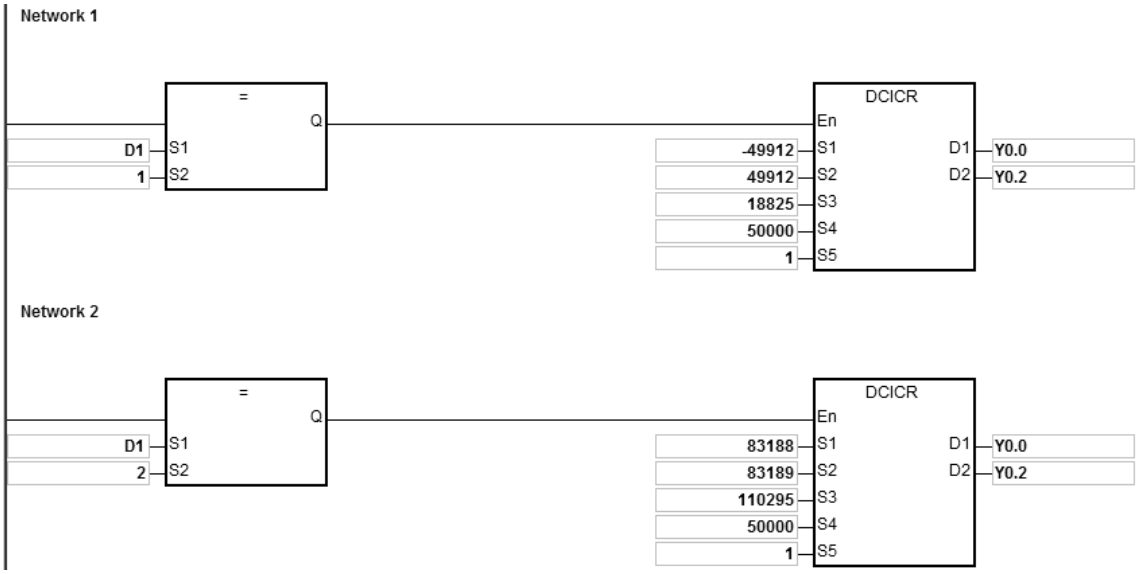
程序范例：

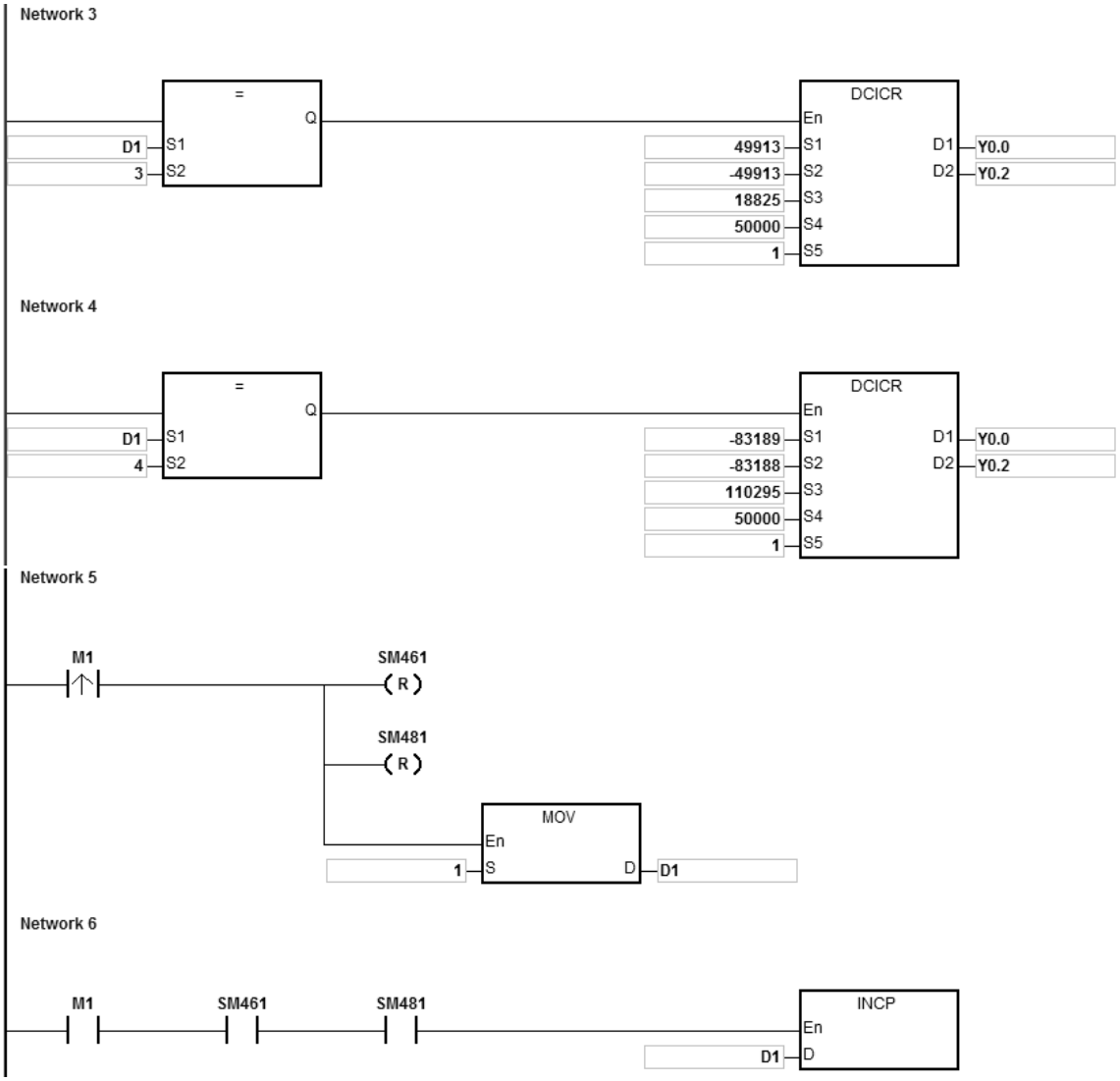
1. 画一个 DELTA LOGO 如下图



2. 操作步骤：将 LOGO 分为三个部分

- 椭圆部分=使用 DCICR 相对位置顺时针画弧插补
- 椭圆形绝对坐标 P1 (39930，-9982)、P2 (-9982，39930)、P3 (73206，123119)、P4 (123119，73206)
- 以 (39930，-9982) 为起点，计算后的相对坐标为 (-49912，49912)、(83188，83189)、(49913，-49913)、(-83189，-83188)





当 PLC RUN，M1=ON 时，开始以频率 50kHz 执行第一段顺时针画弧插补运动，在每一段顺时针画弧插补运动结束后，D1 加 1，自动执行第二段顺时针画弧插补运动，以此类推，直到执行完第 4 段顺时针画弧插补运动。

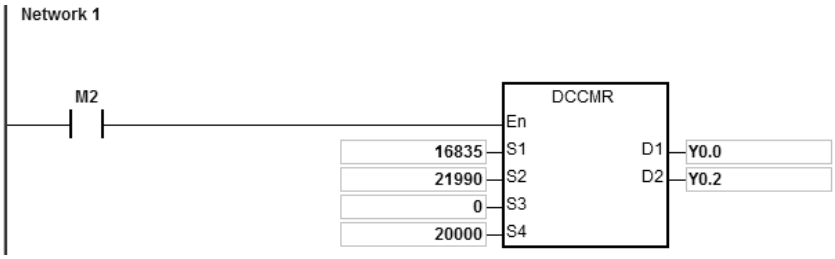
当 D1=1 时，由 P1 到 P2，DCICR 以圆心偏移量 18825，圆弧分辨率 5，开始画弧

当 D1=2 时，由 P2 到 P3，DCICR 以圆心偏移量 110295，圆弧分辨率 5，开始画弧

当 D1=3 时，由 P3 到 P4，DCICR 以圆心偏移量 18825，圆弧分辨率 5，开始画弧

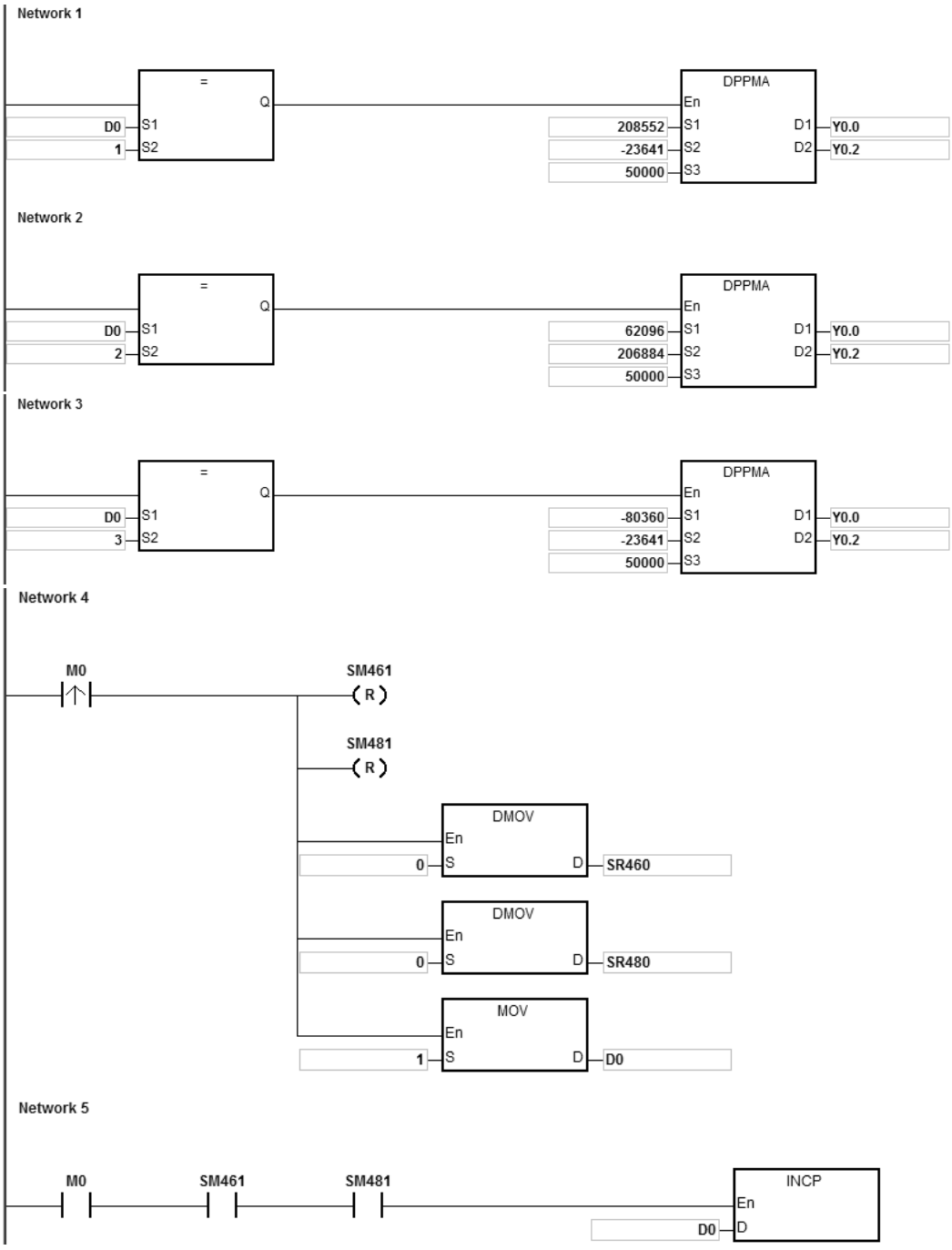
当 D1=4 时，由 P4 到 P1，DCICR 以圆心偏移量 110295，圆弧分辨率 5，开始画弧

- 圆形部分=使用 DCCMR 相对位置画圆
- 圆形绝对坐标 P5（86213，15435）、P6（103048，37425），以（86213，15435）为起点，计算后相对坐标圆心坐标为（16835，21990）



当 PLC RUN，M2=ON 时，开始以目标速度参考值 20kHz 执行相对位置画圆运动。

- 三角形部分=使用 DPPMA 绝对位置双轴同动
- 三角形绝对坐标=起点 P7（-80360，23641）、P8（208552，-23641）、P9（62096，206884）



当 PLC RUN，M0=ON 时，开始以频率 50kHz 执行第一段位置双轴同动运动，在每一段双轴同动运动结束后，D1 加 1，自动执行第二段双轴同动运动，以此类推，直到执行完第 3 段双轴同动运动。

当 D1=1 时，由 P7 到 P8，DPPMA，开始画线

当 D1=2 时，由 P8 到 P9，DPPMA，开始画线

当 D1=3 时，由 P9 到 P7，DPPMA，开始画线

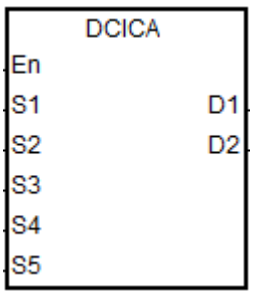
API	指令码			操作数	功能
2713	D	CICA		S_1 · S_2 · S_3 · S_4 · S_5 · D_1 · D_2	绝对位置顺时针圆弧插补

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		○
S_4							●	●	●		○		○	○		
S_5							●	●	●		○		○	○		
D_1		○														
D_2		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●		●				
S_4			●				●						
S_5			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： X 轴目标坐标（绝对位置）

S_2 ： Y 轴目标坐标（绝对位置）

S_3 ： 圆心位移量或是圆心角度

S_4 ： 目标速度参考值

S_5 ： 功能选择

D_1 ： X 轴输出装置

D_2 ： Y 轴输出装置

指令说明：

1. S_1、S_2 分别为 X 轴、Y 轴指定目标坐标地址（绝对位置）。其它相关说明请参考上一个 DCICR 指令说明。
2. ST 结构化语言编辑，请参考 DPPMR 指令的范例二说明。

API	指令码			操作数	功能
2714	D	CICCR		S_1 · S_2 · S_3 · S_4 · S_5 · D_1 · D_2	相对位置逆时针圆弧插补

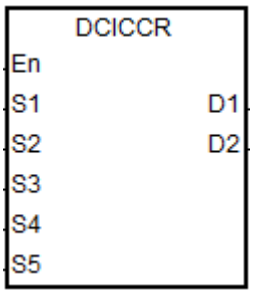
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		○
S_4							●	●	●		○		○	○		
S_5							●	●	●		○		○	○		
D_1		○														
D_2		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●		●				
S_4			●				●						
S_5			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

6

符号：



S_1 ： X 轴目标坐标（相对位置）

S_2 ： Y 轴目标坐标（相对位置）

S_3 ： 圆心位移量或是圆心角度

S_4 ： 目标速度参考值

S_5 ： 功能选择

D_1 ： X 轴输出装置

D_2 ： Y 轴输出装置

指令说明：

1. S_1、S_2 与 S_3 分别为 X 轴、Y 轴指定目标坐标地址（相对位置）与圆心位移量。其详细动作请参考下页的逆时针圆弧动作说明。
2. D_1、D_2 分别为 X 轴与 Y 轴之指定脉冲输出装置，其各轴输出点与模式选择请参考 DPPMR 指令说明。
3. S_4 为设定目标速度之参考值，当指令启动时，PLC 会在规划圆弧行进路径时，搭配此目标速度先行计算，但若是计算过程已经预估无法达成圆弧行进效果时，则会自动帮忙降低输出速度，以达成同动画圆弧之功能为优先。

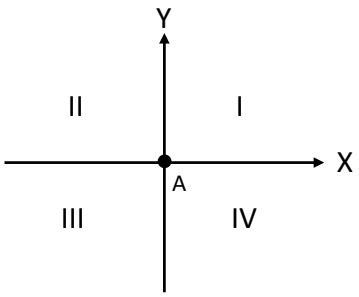
4. **S_5** 为功能选择设定值，详细说明请参考下表：

S_5 设定值	**S_3** 功用	功能说明
0	圆心偏移量(32 位整数)	以角度 10 度为基本运行角度
1	圆心偏移量(32 位整数)	以角度 5 度为基本运行角度
2	圆心偏移量(32 位整数)	以角度 1 度为基本运行角度 (韧体版本需为 V1.04.60 以上)
10	圆心角度 (浮点数)	以角度 10 度为基本运行角度 (韧体版本需为 V1.04.40 以上)
11	圆心角度 (浮点数)	以角度 5 度为基本运行角度 (韧体版本需为 V1.04.40 以上)
12	圆心角度 (浮点数)	以角度 1 度为基本运行角度 (韧体版本需为 V1.04.60 以上)
20	无效	以基本角度 10 度与目标 X，Y 坐标运行固定 90 度圆弧 (韧体版本需为 V1.10.10 以上)
21	无效	以基本角度 5 度与目标 X，Y 坐标运行固定 90 度圆弧 (韧体版本需为 V1.10.10 以上)
22	无效	以基本角度 1 度与目标 X，Y 坐标运行固定 90 度圆弧 (韧体版本需为 V1.10.10 以上)
其他	圆心偏移量(32 位整数)	皆以 **S_5** 设定值 0 执行

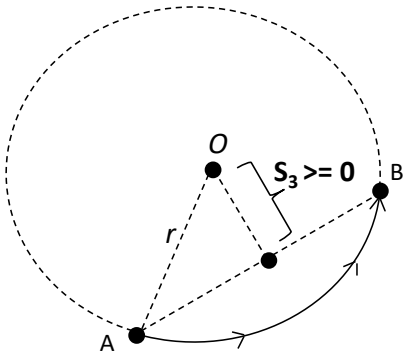
5. ST 结构化语言编辑，请参考 DPPMR 指令的范例二说明。

逆时针之圆弧动作说明 (适用 S_5 设定值 0~12) :

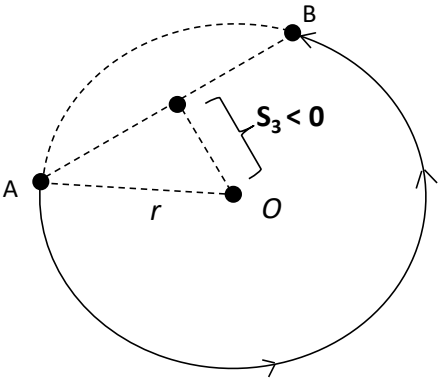
1. 定义坐标象限如下图所示。其坐标分成四个象限，分别为第 I, II, III 及 IV 象限。图中 A 表示 X,Y 轴当前坐标位置。



2. 定义 B 点为 **S_1** 与 **S_2** 目标坐标位置。定义 *O* 点为 A 与 B 两点间连成直线之中心点，也即是所谓圆心点。
3. **S_3** 圆心位移量即是指圆心 *O* 点指定之偏移量。V1.04.60 版 (含) 以上，功能选择圆心角度即是指∠AOB 之夹角。
4. 当目标 B 点位于当前 A 点的第 I, IV 象限，则 **S_3** 圆心位移量设定值将执行如下两图之行进路径 (如：实线部分) ：(图中 *r* 表示半径)

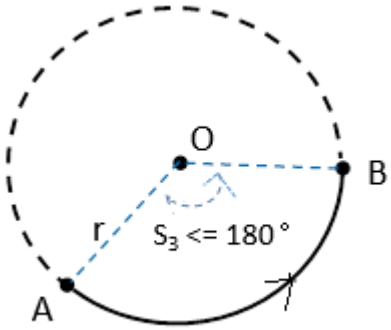


当 **S3**>= 0

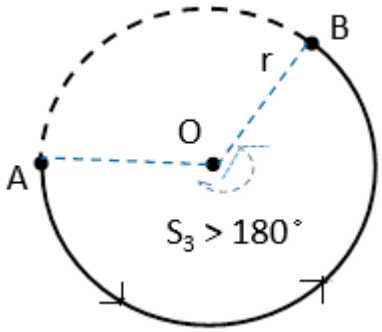


当 **S3**< 0

5. 当目标 B 点位于当前 A 点的第 I、IV 象限，则 **S3** 圆心角度设定值将执行如下两图之行进路径（如：实线部分）：（图中 *r* 表示半径）

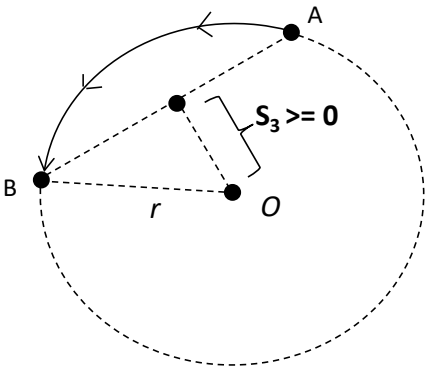


当 **S3**<= 180.0

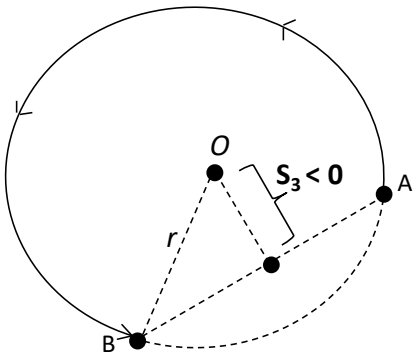


当 **S3** > 180.0

6. 当目标 B 点位于当前 A 点的第 II、III 象限，则 **S3** 圆心位移量设定值将执行如下两图之行进路径：

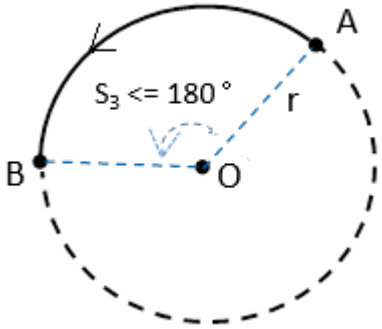


当 **S3**>= 0

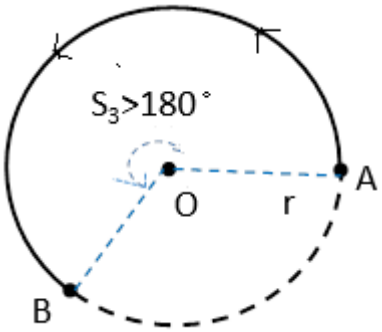


当 **S3**< 0

7. 当目标 B 点位于当前 A 点的第 II、III 象限，则 **S3** 圆心角度设定值将执行如下两图之行进路径：



当 **S_3**<= 180.0

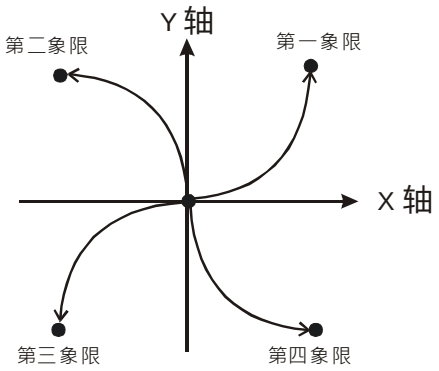


当 **S_3**> 180.0

8. 当目标 B 点之 X 轴为 0，若 Y 轴目标位置 >= 0 时，则此 B 点定义位于 A 点的第 I 象限；相反的，若 Y 轴目标位置 < 0，则此 B 点定义位于 A 点的第 III 象限。

固定 90 度之逆时针圆弧动作说明（适用 S_5 设定值 20~22）：

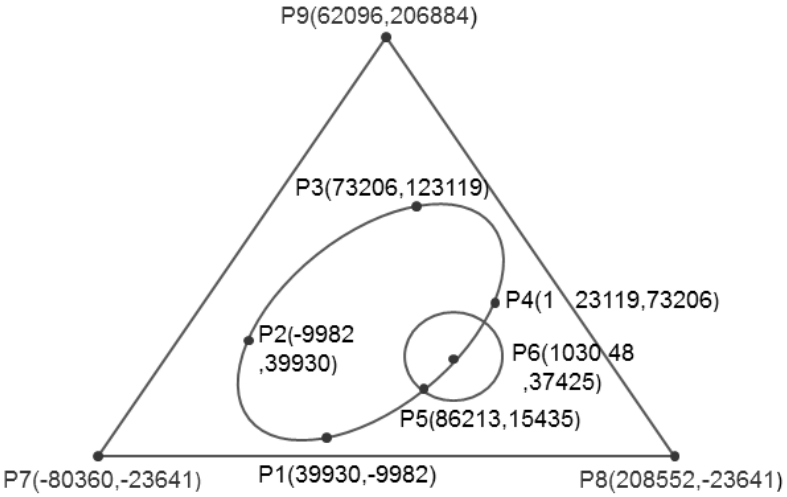
1. 坐标象限同上述模式说明，此功能将固定以逆时针方向进行 90 度圆弧运行。注：此功能需搭配韧体 V1.10.10 版（含）以上。
2. **S_1** 与 **S_2** 参数所指定之四个象限的目标位置，相对于当下 X 与 Y 轴现在位置（下图中心点），其运行路径将如下图所示之四种情况。



3. 由于每次都以 90 度运行，因此当 X/Y 轴的目标位置不是相同距离时，其圆弧将不会形成四分之一的正圆形，适合被使用于画椭圆形之应用。

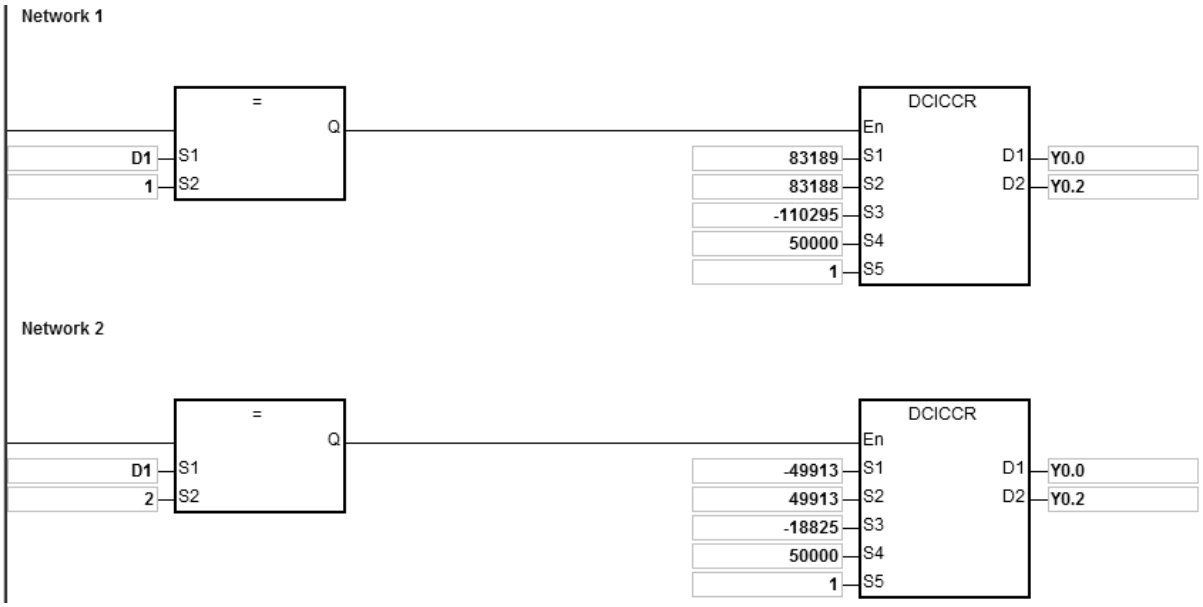
程序范例：

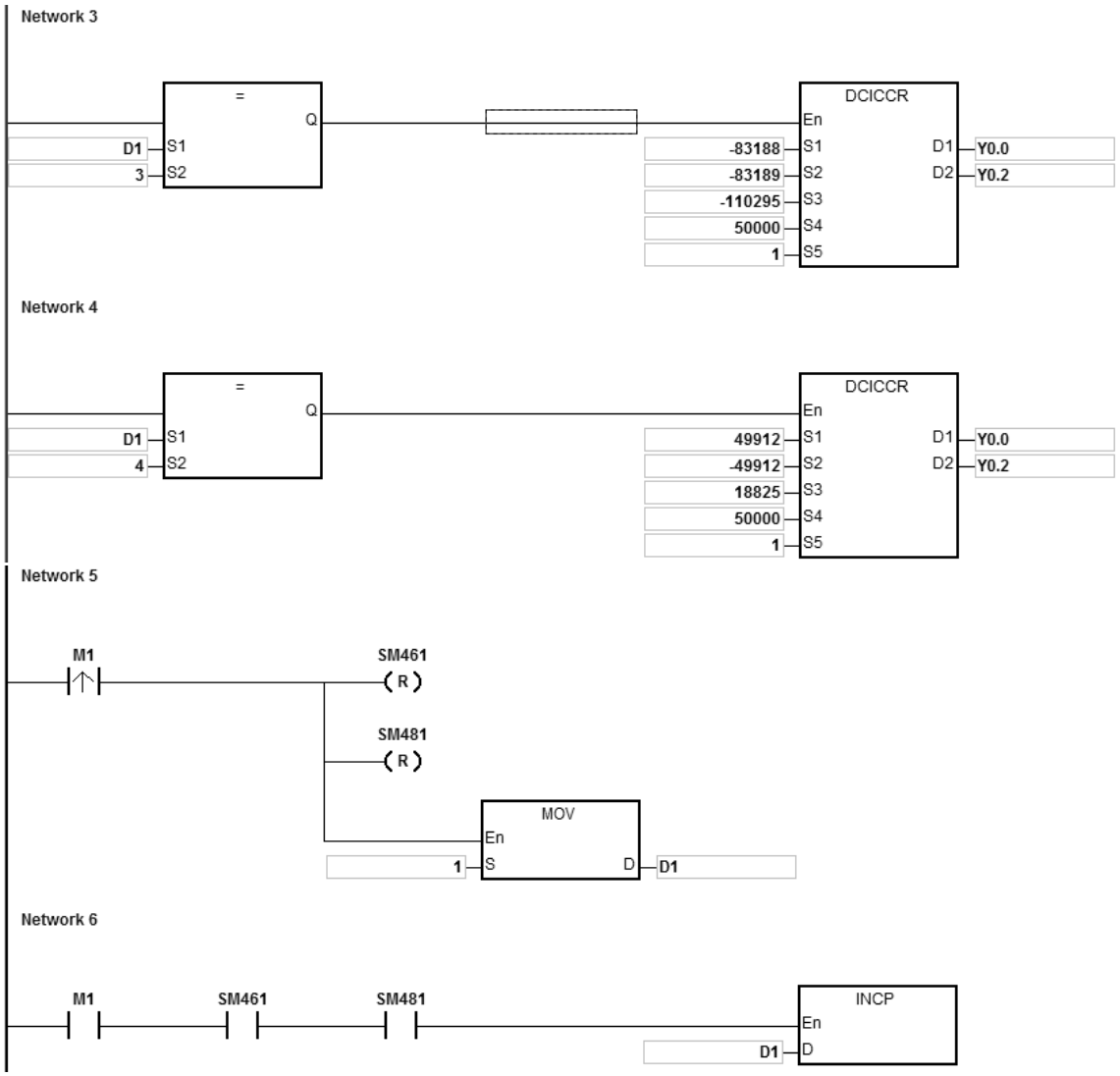
1. 画一个 DELTA LOGO 如下图



2. 操作步骤：将 LOGO 分为三个部分

- 椭圆部分=使用 DCICCR 相对位置逆时针画弧插补
- 椭圆形绝对坐标 P1（39930，-9982）、P4（123119，73206）、P3（73206，123119）、P2（-9982，39930）
- 以（39930，-9982）为起点，计算后的相对坐标为（83189，83188）、（-49913，49913）、（-83188，-83189）、（49912，-49912）





当 PLC RUN，M1=ON 时，开始以频率 50kHz 执行第一段顺时针画弧插补运动，在每一段顺时针画弧插补运动结束后，D1 加 1，自动执行第二段顺时针画弧插补运动，以此类推，直到执行完第 4 段顺时针画弧插补运动。

当 D1=1 时，由 P1 到 P4，DCICCR 以圆心偏移量-110295，圆弧分辨率 5，开始画弧

当 D1=2 时，由 P4 到 P3，DCICCR 以圆心偏移量-18825，圆弧分辨率 5，开始画弧

当 D1=3 时，由 P3 到 P2，DCICCR 以圆心偏移量-110295，圆弧分辨率 5，开始画弧

当 D1=4 时，由 P2 到 P1，DCICCR 以圆心偏移量-18825，圆弧分辨率 5，开始画弧

- 圆形部分=使用 DCCMR 相对位置画圆，请参考 API 2712 DCICR 范例
- 三角形部分=使用 DPPMA 绝对位置双轴同动，请参考 API 2712 DCICR 范例

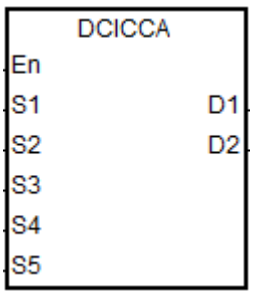
API	指令码			操作数	功能
2715	D	CICCA		**S_1 · S_2 · S_3 · S_4 · S_5 · D_1 · D_2**	绝对位置逆时针圆弧插补

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		○
S_4							●	●	●		○		○	○		
S_5							●	●	●		○		○	○		
D_1		○														
D_2		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●		●				
S_4			●				●						
S_5			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： X 轴目标坐标（绝对位置）

S_2 ： Y 轴目标坐标（绝对位置）

S_3 ： 圆心位移量或是圆心角度

S_4 ： 目标速度参考值

S_5 ： 功能选择

D_1 ： X 轴输出装置

D_2 ： Y 轴输出装置

指令说明：

1. **S_1**、**S_2** 分别为 X 轴、Y 轴指定目标坐标地址（绝对位置）。其它相关说明请参考上一个 DCICCR 指令说明。
2. ST 结构化语言编辑，请参考 DPPMR 指令的范例二说明。

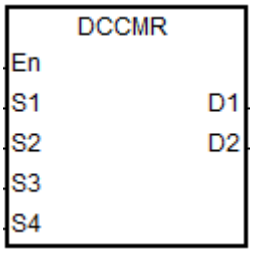
API	指令码			操作数	功能
2716	D	CCMR		S_1 · S_2 · S_3 · S_4 · D_1 · D_2	相对位置画圆

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
S_4							●	●	●		○		○	○		
D_1		○														
D_2		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 圆心 X 轴坐标地址（相对位置）

S_2 ： 圆心 Y 轴坐标地址（相对位置）

S_3 ： 功能选择

S_4 ： 目标速度参考值

D_1 ： X 轴输出装置

D_2 ： Y 轴输出装置

指令说明：

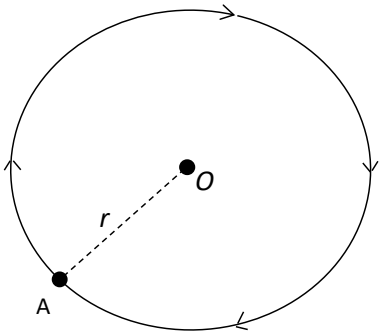
1. S_1、S_2 分别为 X 轴、Y 轴指定圆心坐标地址（相对位置）。
2. S_3 为功能选择设定值，详细说明如下表：

S_3 设定值	功能说明
0	以顺时针方式，间隔角度 10 度为基本运行角度
1	以顺时针方式，间隔角度 5 度为基本运行角度
2	以逆时针方式，间隔角度 10 度为基本运行角度
3	以逆时针方式，间隔角度 5 度为基本运行角度
4	以顺时针方式，间隔角度 1 度为基本运行角度（韧体版本需为 V1.04.60 以上）

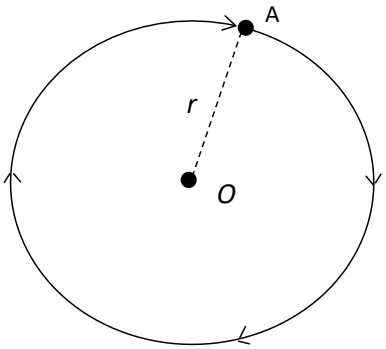
S_3 设定值	功能说明
5	以逆时针方式，间隔角度 1 度为基本运行角度（韧体版本需为 V1.04.60 以上）
其他	皆以默认值 0 执行

3. D_1、D_2 分别为 X 轴与 Y 轴之指定脉冲输出装置，其各轴输出点、模式选择与输出轴参数，请参考 DPPMR 指令说明。
4. S_4 为设定目标速度之参考值，当指令启动时，PLC 会在规划圆弧行进路径时，搭配此目标速度先行计算，但若是计算过程已经预估无法达成圆弧行进效果时，则会自动帮忙降低输出速度，以达成同动画圆之功能为优先。
5. 当圆心 *O* 点之 X 轴坐标为 0，若 Y 轴目标位置 >= 0 时，则此 *O* 点定义位于 A 点的第 I 象限；相反的，若 Y 轴目标位置 < 0，则此 *O* 点定义位于 A 点的第 III 象限。
6. ST 结构化语言编辑，请参考 DPPMR 指令的范例二说明。

顺时针画圆动作说明：下列图中 A 为当前位置坐标，*O* 为目标圆心坐标，*r* 为圆的半径。

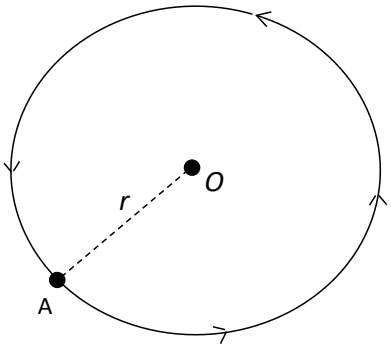


圆心 *O* 位于 I，IV 象限

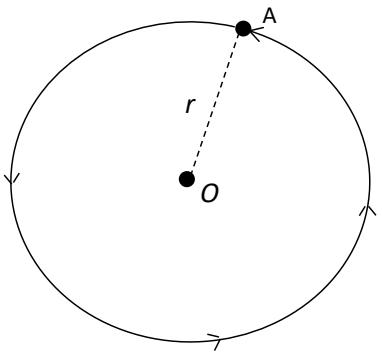


圆心 *O* 位于 II，III 象限

逆时针画圆动作说明：



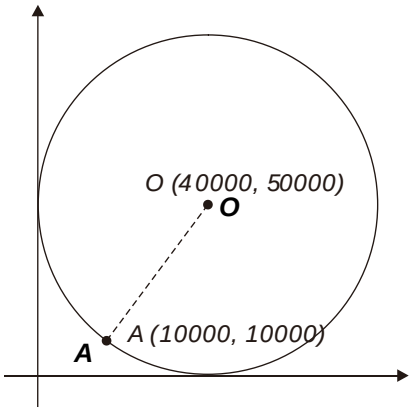
圆心 *O* 位于 I，IV 象限



圆心 *O* 位于 II，III 象限

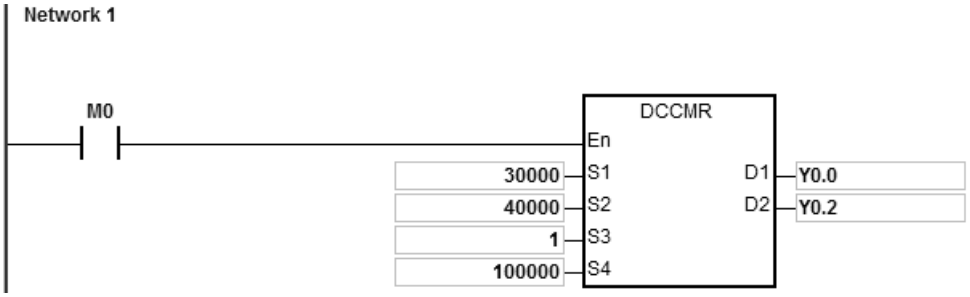
程序范例：

1. 以坐标（40000，50000）为圆心 *O* 点，坐标（10000，10000）为 *A* 点，画一个圆型，如下图



2. 操作步骤：

- 计算 *O* 点的相对地址的坐标如下（30000，40000）
- A 点坐标为当前位置，SR460=10000，SR480=10000
- S_3=1，顺时针且间隔每 5 度为基本行进角度
- PLC RUN，并设定 M0 为 ON，则开始相对位置画圆动作。



3. 动作说明：

- 当 PLC RUN，M0=ON 时，开始以频率 100kHz，执行顺时针画圆。
- 当画圆动作完成时，完成标志 SM461 及 SM481 会设定为 ON。

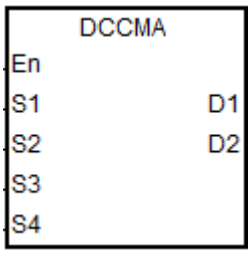
API	指令码			操作数	功能
2717	D	CCMA		**S_1 · S_2 · S_3 · S_4 · D_1 · D_2**	绝对位置画圆

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
S_4							●	●	●		○		○	○		
D_1		○														
D_2		○														

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4			●				●						
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 圆心 X 轴坐标地址（绝对位置）

S_2 ： 圆心 Y 轴坐标地址（绝对位置）

S_3 ： 功能选择

S_4 ： 目标速度

D_1 ： X 轴输出装置

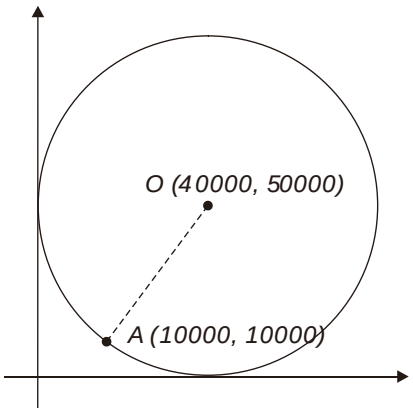
D_2 ： Y 轴输出装置

指令说明：

S_1、**S_2** 分别为 X 轴、Y 轴指定圆心坐标地址（绝对位置）。其余执行动作请参考 DCCMR 指令说明。

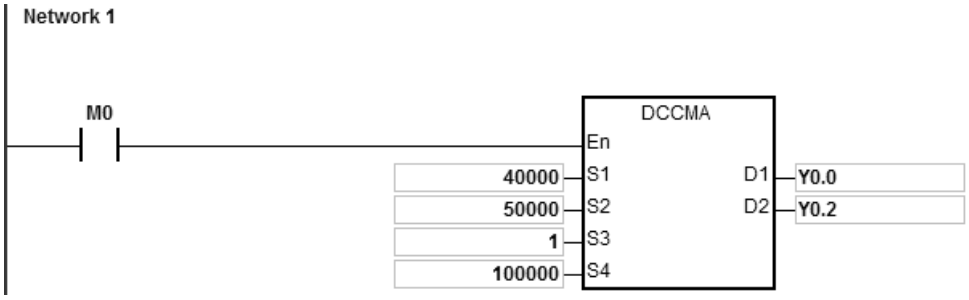
程序范例：

1. 以坐标（40000，50000）为圆心 *O* 点，坐标（10000，10000）为 *A* 点，画一个圆型，如下图。
2. ST 结构化语言编辑，请参考 DPPMR 指令的范例二说明。



3. 操作步骤：

- 在指令的 X，Y 坐标填入绝对地址的坐标如下（40000，50000）
- A 点坐标为当前位置，SR460=10000，SR480=10000
- S_3=1，顺时针且间隔每 5 度为基本行进角度
- PLC RUN，并设定 M0 为 ON，则开始绝对位置画圆动作。



4. 动作说明：

- 当 PLC RUN，M0=ON 时，开始以频率 100kHz，执行顺时针画圆。
- 当画圆动作完成时，完成标志 SM461 及 SM481 会设定为 ON。

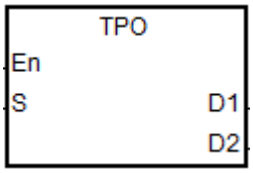
API	指令码	操作数	功能
2718	TPO	**S**・**D_1**・**D_2**	定位规划表控制输出

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S								●				○	○	○		
D_1								●								
D_2		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D_1		●				●							
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



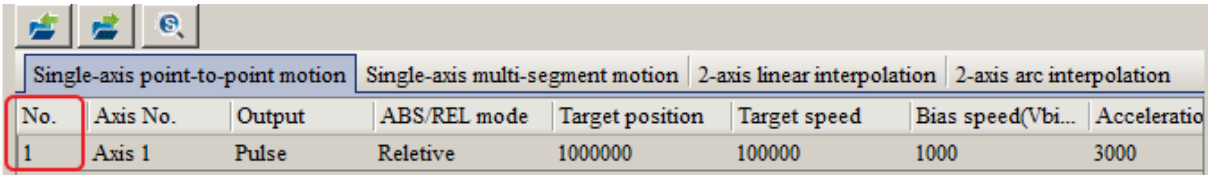
S ： 定位规划表之编号

D_1 ： 显示执行中之输出表编号

D_2 ： 连续编号输出时切换标志

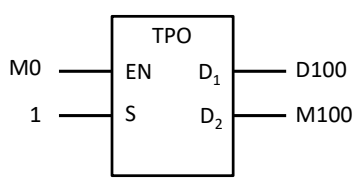
指令说明：

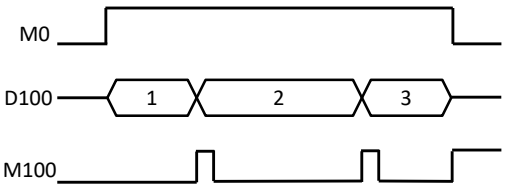
1. **S_1** 为定位规划表所编排之编号，若表格内无此编号，则指令将不会被执行，并且设定 SM0=ON 及记录错误码 SR0=16#2027。



2. **S_1** 编号为启动时的第一个输出之编号，若输出过程中有设定连续编号输出之动作，请参考 **D_1** 显示之输出编号。

3. 当单个编号输出或者连续编号第一个输出时，指定之 **D_2** 切换标志将会被设置为 OFF，当连续编号输出执行到输出编号切换时，**D_2** 切换标志将会被 ON 一个扫描周期，直到最后一个编号输出完成时再设为 ON。指令范例与标志切换时序图如下图所示：





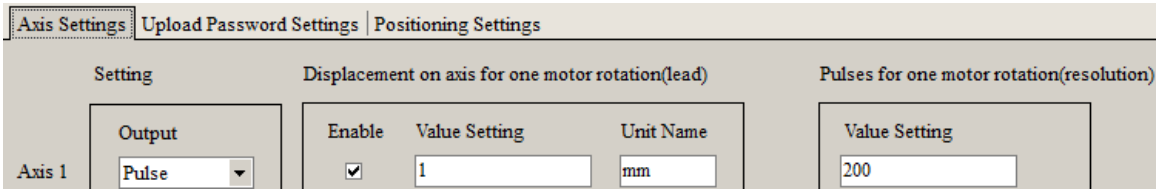
4. 此 **D_2** 切换标志将会受 PLC 程序扫描时间影响，若切换下一段输出的时间快于扫描时间，则那一段的切换标志将可能来不及清除为 OFF。

5. 此指令不建议使用于 ST 结构化语言编辑、中断程序以及仅被呼叫一次性功能块（FB）程序。
6. 当采用单轴多段速输出方式，并且韧体版本为 V1.07.10 版（含）以下时，该输出方式将不可搭配暂停标志目标功能，若有需要输出过程中进行立即停止输出动作，请采用直接关闭 TPO 输出指令。韧体版本为 V1.08.00 版（含）以上时，若搭配暂停标志进行暂停功能，则 TPO 指令将会暂停下一段的接续动作。举例：原先规划编号 1，2，3 的编号顺序执行，但当执行编号 2 时，用户启动暂停标志，则输出会执行完编号 2，但不会执行编号 3 的指定动作；然后等到用户关闭暂停标志之后，TPO 指令才会开始执行编号 3。
7. 当采用单轴定位或双轴直线插补输出方式时，若搭配暂停标志进行暂停功能，则 TPO 指令将会立即执行暂停动作，待关闭暂停标志时，将重新执行剩余之输出个数。
8. 当采用双轴圆弧插补输出方式时，若搭配暂停标志进行暂停功能，则 TPO 指令将会在执行一个基本圆弧单位（角度 1 度）之后暂停，待关闭暂停标志时，将重新执行剩余之圆弧角度的输出个数。
9. 主机韧体版本 V1.12.10（含）以上，支持暂停切换到下一段或是切换到特定的定位规划表编号。各输出轴搭配下列 SM/SR 使用。TPO 指令执行中，当前定位规划表编号输出完毕，要切换到下一段时，若暂停切换 SM 标志为 ON，会依特定定位规划表编号 SR，执行不同动作：
   A. 特定定位规划表编号 SR 的数值为存在的定位规画表编号：会切换到特定的定位规划表编号，同时，会清除暂停切换 SM 标志和特定定位规划表编号 SR。
   B. 特定定位规划表编号 SR 的数值为 0 或是不存在的定位规画表编号：会暂停切换到下一段，直到用户自行清除暂停切换 SM 标志，才会切换到下一段，恢复输出。

输出轴	暂停切换标志	特定定位规划表编号
轴 1（Y0.0/Y0.1）	SM591	SR1291
轴 2（Y0.2/Y0.3）	SM592	SR1292
轴 3（Y0.4/Y0.5）	SM593	SR1293
轴 4（Y0.6/Y0.7）	SM594	SR1294
轴 5（Y0.8/Y0.9）	SM595	SR1295
轴 6（Y0.10/Y0.11）	SM596	SR1296

程序范例一：

1. 开启机械单位转换功能

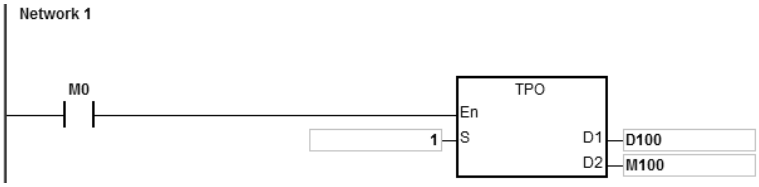


2. 设定定位规划表与下载

此时的目标位置 2000 是机械单位(Unit:mm)，并且经由 "定位规划表" 协助转换后，下载实际输出 400,000 个 PULSE 至 PLC 主机内储存。

No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	Bias speed(Vbi...	Acceleration ti...
1	Axis 1	Relative	2000	100000	200	200

3. 当 M0 为 ON 时，Y0.0 以频率 100kHz 输出 400,000 个脉冲，脉冲输出完毕 M100=ON。



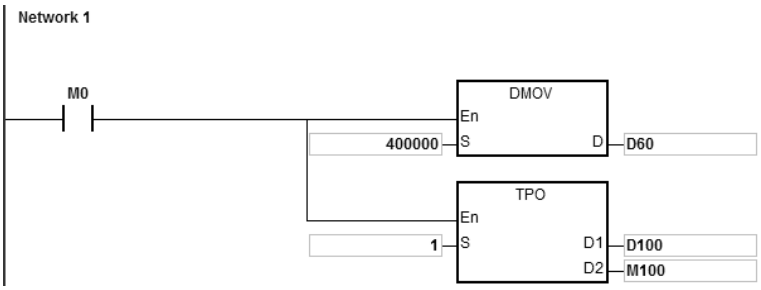
程序范例二：目标位置由全局变量或是 D 装置填入

1. 设定定位规划表，以 D 装置填入目标位置并下载

   由于 PLC 的目标位置(D60)内定为 PULSE 单位，因此当用户采用 D 装置当目标位置，并且想要用机械单位当命令时，则需自行额外设计机械单位转换 PULSE 单位的转换程序。

No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	Bias speed(Vbi...	Acceleration ti...
1	Axis 1	Relative(Symbol/Address)	D60	100000	200	200

2. 当 M0 为 ON 时，D60=400,000，Y0.0 以频率 100kHz 输出 400,000 个脉冲，脉冲输出完毕 M100=ON。



注:目标位置由全局变量、D 装置填入或是由指令(DTPWS/DTPWL/DTPWC)写入时，都是以 PULSE 为单位

3. 若目标位置使用全局变量时

   A.全局变量需指定地址

Global Symbols

Class	Identifiers	Address	Type...	Initial Value (Activ...
VAR	Target_Axis1	D60	DWORD	N/A

   B. 定位规划表中，在目标位置使用全局变量

Single-axis point-to-point motion | Single-axis multi-segment motion | 2-axis linear interpolation | 2-axis arc interpolation

No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	Bias speed(Vbi...	Acceleration ti...
1	Axis 1	Relative(Symbol/Address)	Target_axis1	100000	200	200

程序范例三：中断信号及完成标志说明

1. 设定定位规划表，如下

Single-axis point-to-point motion | Single-axis multi-segment motion | 2-axis linear interpolation | 2-axis arc interpolation

No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	ACC/DEC time(ms)	Abort signal(I/M)	Completion action	Completion flag(I/M)	No. of next segment
3	Axis 1	Relative	10000	5000	300	M100	Go to next segment		4
4	Axis 1	Relative	15000	8000	100		Completion	M777	---

2. 中断信号：当 No3 输出时尚未完成时，M100 为 ON，则立即输出 No4
   中断信号支持：外部中断、高速比较中断或 M 装置（M0~M8191）
3. 完成标志：当 No4 输出完成时，M777 为 ON
   当此表格执行完毕时，会呼叫中断或是启动 M 装置
   支持的定位规划表中断编号为 I510~519；M 装置（M0~M8191）

备注：采用中断编号变更时，将不受 PLC 扫描所影响，若采用 M 装置，则会有影响。

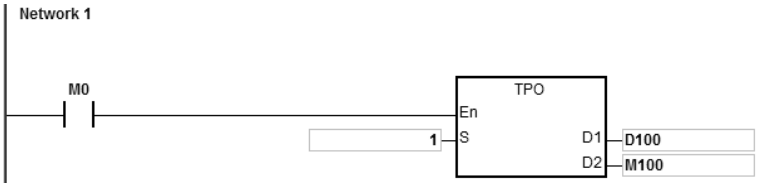
程序范例四：单轴多段输出

1. 设定定位规划表与下载

Single-axis point-to-point motion | Single-axis multi-segment motion | 2-axis linear interpolation | 2-axis arc interpolation

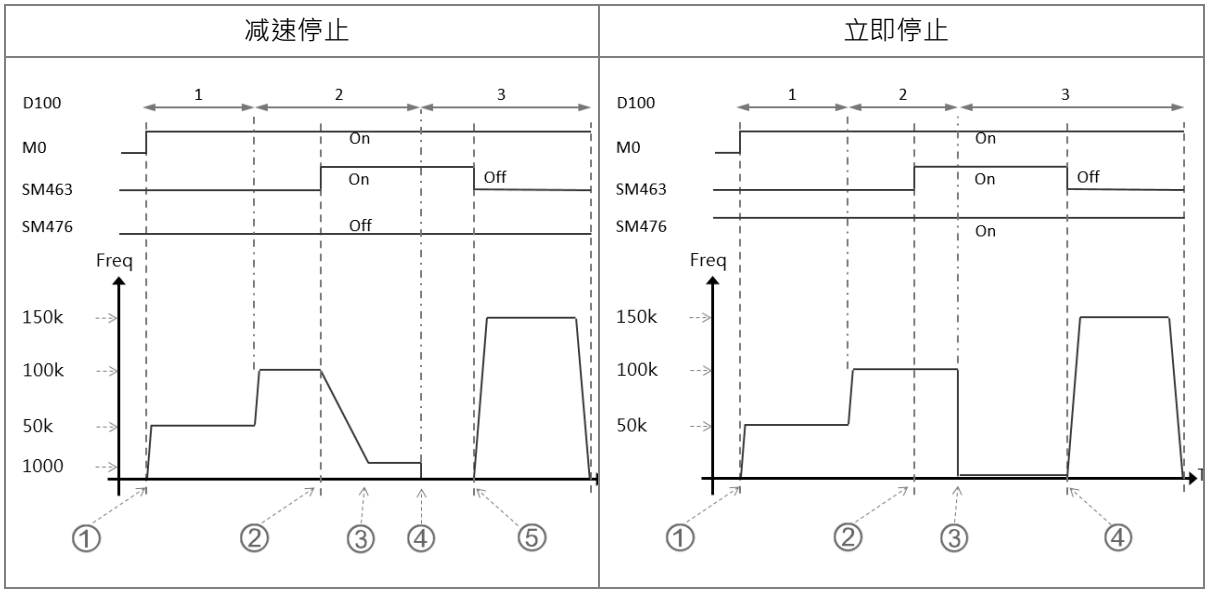
No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	ACC/DEC time(...	Abort signal(I/...	Completion acti...
1	Axis 1	Relative	200000	50000	200		Go to next segmer
2	Axis 1	Relative	400000	100000	200		Go to next segmer
3	Axis 1	Relative	600000	150000	200		Go to next segmer

2. 当 M0 为 ON 时，Y0.0 依续执行 No.1~No.3 三段输出，脉冲输出完毕 M100=ON。



3. 减速停止即立即停止动作说明

a) 输出曲线图：



b) 减速停止动作说明

① M0 为 ON，启动 TPO 指令，开始第一段 No.1 输出。

② 在执行第二段 No.2 输出，当暂停标志(SM463)为 ON 时，会确认立即停止标志(SM476)的状态。

③ 如果立即停止标志（SM476）为 OFF 时，以轴 1 的减速时间 SR465 执行减速；直到速度降到最低频率 1000Hz。

④ 以最低频率 1000hz 输出至第二段 No.2 输出完成后停止（Pulse=600K）。

⑤ 当暂停标志被启动（SM463）为 OFF 时，TPO 开始执行第三段 No.3 输出，输出完成后（Pulse=1200K）。

c) 立即停止动作说明

① M0 为 ON，启动 TPO 指令，开始第一段 No.1 输出。

② 在执行第二段 No.2 输出，当暂停标志(SM463)为 ON 时，会确认立即停止标志(SM476)的状态。

③ 如果立即停止标志（SM476）为 ON 时，TPO 会将第二段 No.2 输出完成后立即停止（Pulse=600K）。

④ 当暂停标志被启动（SM463）为 OFF 时，TPO 开始执行第三段 No.3 输出，输出完成后（Pulse=1200K）。

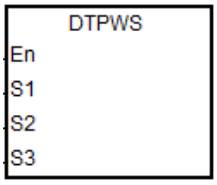
API	指令码			操作数	功能
2719	D	TPWS	P	S_1、S_2、S_3	定位规划表之单轴输出参数写入

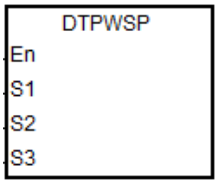
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：





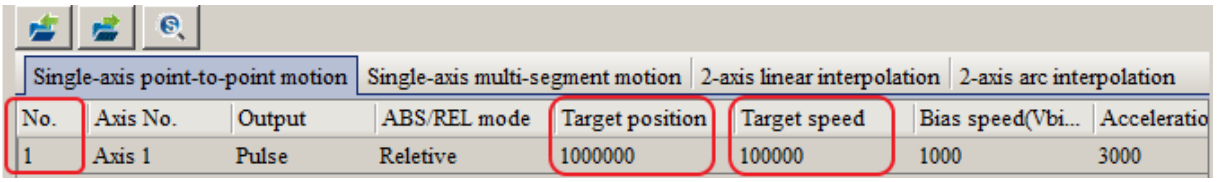
S_1 ： 输出表编号

S_2 ： 目标位置

S_3 ： 目标速度

指令说明：

1. S_1 为定位规划表所编排之编号，若表格内无此编号或编号指定模式不为单轴输出 (点对点与多段速) 模式，则指令将不会被执行，并且设定 SM0=ON 及记录错误码 SR0=16#2027。



2. S_2 为目标位置。此参数只可输入 32 位整数，若软件编辑规划表时，有搭配使用机械单位转换时，请先使用转换指令进行修改。
3. S_3 为目标速度，此速度设定值范围 1Hz~200,000Hz。

   注 1：AS324MT 机种可设定 Y0, Y2 输出轴最高为 4MHz)

   注 2：数值超出范围时，自动以最小或最大值设定。
4. 当指令启动修改且被指定修改单轴正在输出时，此修改参数会被修改进入表格内，但是得等到下一次输出被启动时才会有效。
5. 此指令修改之参数，仅适用于 PLC 运行中修改，断电时最后一笔写入之参数将不会被保存。定位规划表每次上电时，皆以软件编辑且下载之定位规划表参数为默认值。
6. V1.06.00 版(含)以后支持 ,此指令默认不修改加减速时间 ,若用户须连带变更加减速时间 ,请先设定 SM585 标志为 On，并且变更该轴配置的加速 (单轴点对点及单轴多段) 与减速 (单轴点对点) 时间 SR 之后，再

启动此指令进行变更。

7. 若 TPO 指令正在输出执行中，请注意搭配此指令修改参数的时机点，建议要避开修改正在或即将执行输出的已编排之编号。

范例一：轴 3 已经建好规画表如下图，正在执行过程中，临时变更编号 2 的目标位置、速度与加减速时间。

单轴点对点 | 单轴多段输出 | 双轴直线补间 | 双轴圆弧补间

No.	轴编号	地址模式	目标位置	目标速度(Hz)	加减速时间(ms)	中断讯号(I/M)	完成后动作	完成旗标(...	下一段编号
1	轴 3	相对模式	1000	1000	100		执行下一段		2
2	轴 3	相对模式	2000	2000	50		执行下一段		3
3	轴 3	相对模式	1500	1500	2000		结束	M100	---

变更步骤如下：

1) 使用 DMOV 指令指定新的目标位置与速度。
2) 使用 MOV 指令指定轴 3 的加速时间 SR504（单位 ms）。批注：不管加速或减速，单轴多段的加减速时间只采用加速时间的 SR。
3) 设定 SM585 标志为 On。
4) 启动 DTPWS 指令执行一次，变更规划表参数内容。
5) 变更完成后，主机将会自动清除 SM585 为 Off。

范例二：轴 3 已经建好规画表如下图，正在执行过程中，临时变更编号 2 的所有参数。

单轴点对点 | 单轴多段输出 | 双轴直线补间 | 双轴圆弧补间

No.	轴编号	地址模式	目标位置	目标速度(Hz)	起停速度(Hz)	加速度时间(ms)	减...	完成后动作	完成旗标(...	下一段编号
1	轴 2	相对模式	1000	1000	100	100	200	执行下一段		2
2	轴 3	相对模式	2000	2000	50	50	100	结束		---

变更步骤如下：

1) 使用 DMOV 指令指定新的目标位置与速度。
2) 使用 MOV 指令指定轴 3 的起停速度 SR503、加速时间 SR504 以及减速时间 SR505。批注：单轴点对点输出时，起停速度、加速与减速时间的 SR 都会一起被更新。
3) 设定 SM585 标志为 On。
4) 启动 DTPWS 指令执行一次，变更规划表参数内容。
5) 变更完成后，主机将会自动清除 SM585 为 Off。

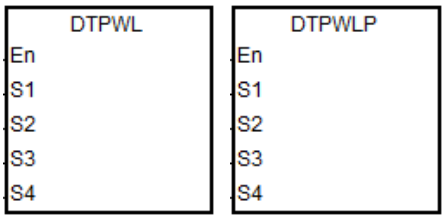
API	指令码			操作数	功能
2720	D	TPWL	P	S_1、S_2、S_3、S_4	定位规划表之直线插补参数写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4			●				●						

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S_1 ： 输出表编号

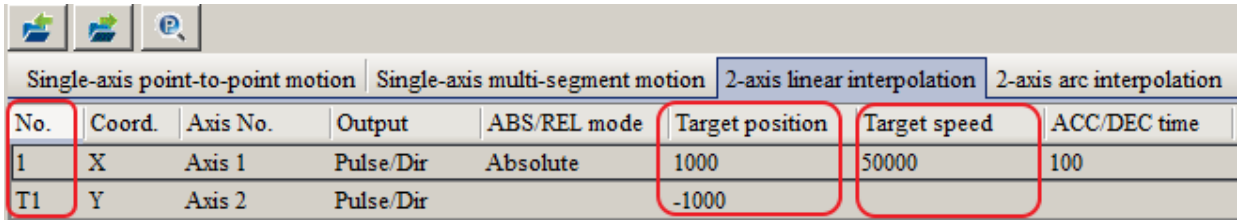
S_2 ： X 轴目标位置

S_3 ： Y 轴目标位置

S_4 ： 目标速度

指令说明：

1. S_1 为定位规划表所编排之编号，若表格内无此编号或编号内指定模式不为直线插补模式，则指令将不会被执行，并且设定 SM0=ON 及记录错误码 SR0=16#2027。



2. S_2 与 S_3 分别为 X 轴与 Y 轴目标位置。此两个参数只可输入 32 位整数，若软件编辑规划表时，有搭配使用机械单位转换时，请先使用转换指令进行修改。
3. S_4 为目标速度，此速度设定值范围 1Hz~200,000Hz。(注：数值超出范围时，自动以最小或最大值设定)
4. 当直线插补的动作执行时，此 S_4 目标速度将自动对应到距离较长的一轴输出，若 X 与 Y 轴无法配对执行同时到达时，则 PLC 内部将自动降速，已两轴同时到达为最终目标。

5. 当指令启动修改且被指定修改之任一轴正在输出时，此双轴参数会被修改进入表格内，但是得等到下一次双轴输出被启动时才会有效。
6. 此指令修改之参数，仅适用于 PLC 运行中修改，断电时最后一笔写入之参数将不会被保存。定位规划表每次上电时，皆以软件编辑且下载之定位规划表参数为默认值。
7. 若需修改加减速时间，请先修改指定 X 轴输出轴所对应的加速时间 SR，再设定 SM585 标志，然后再启动此指令，当完成相关参数变更后，会自动清除 SM585 为 OFF。举例：X 轴指定为 Y0.0 输出轴，则修改加减速时间就必须修改 SR464（Y0.0 输出的加速时间）。详细使用步骤请参考 TPWS 指令范例一说明。

   注 1：此加减速时间仅供 PLC 运算双轴同动输出参考，若计算之后无法达成同时到达，PLC 将自动以最适合时间执行，且不会产生任何错误信息。

   注 2：此功能 PLC 韧体支持版本 V1.12.00（含）以上。
8. 若 TPO 指令正在输出执行中，请注意搭配此指令修改参数的时机点，建议要避开修改正在或即将执行输出的已编排之编号。

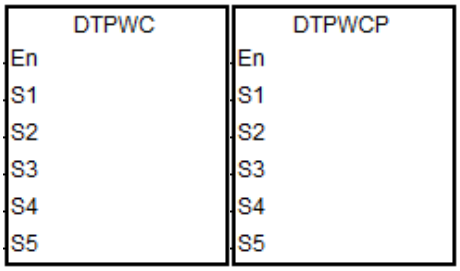
API	指令码			操作数	功能
2721	D	TPWC	P	S_1、S_2、S_3、S_4、S_5	定位规划表之圆弧插补参数写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4			●				●						
S_5			●				●						

脉冲执行型	16 位指令	32 位指令
AS	-	AS

符号：



S_1 ： 输出表编号

S_2 ： X 轴目标位置

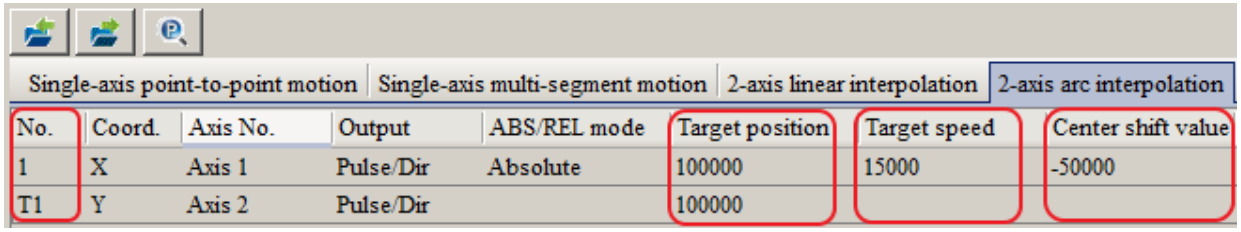
S_3 ： Y 轴目标位置

S_4 ： 圆心偏移位置

S_5 ： 目标速度

指令说明：

1. S_1 为定位规划表所编排之编号，若表格内无此编号或编号指定模式不为圆弧插补模式，则指令将不会被执行，并且设定 SM0=ON 及记录错误码 SR0=16#2027。



2. S_2 与 S_3 分别为 X 轴与 Y 轴目标位置。S_4 为圆心偏移位置。此三个参数只可输入 32 位整数，若软件编辑规划表时，有搭配使用机械单位转换时，请先使用转换指令进行修改。
3. S_5 为目标速度，此速度设定值范围 1Hz~200,000Hz。此实际输出之参考速度，若无法达成双轴同步输出之效果，PLC 将自动降速执行。

4. 当指令启动修改且被指定修改之任一轴正在输出时，此双轴参数会被修改进入表格内，但是得等到下一次双轴输出被启动时才会有效。
5. 此指令修改之参数，仅适用于 PLC 运行中修改，断电时最后一笔写入之参数将不会被保存。定位规划表每次上电时，皆以软件编辑且下载之定位规划表参数为默认值。
6. 此指令不支持顺时针与逆时针之行进方向变更，若需变更行进方向，则请从软件之定位规划表内，新增或修改圆弧插补输出参数。
7. 若 TPO 指令正在输出执行中，请注意搭配此指令修改参数的时机点，建议要避开修改正在或即将执行输出的已编排之编号。

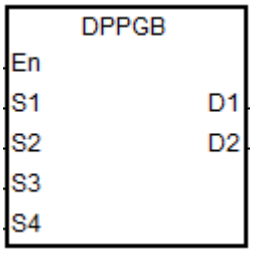
API	指令码			操作数	功能
2723	D	PPGB		S_1 · S_2 · S_3 · S_4 · D_1 · D_2	点对点来回定位控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●								
S_2								●								
S_3								●								
S_4								●								
D_1		○														
D_2			●													

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4									●				
D_1	●												
D_2	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 相对位置 A

S_2 ： 相对位置 B

S_3 ： 目标速度

S_4 ： 目标速度比率调整（浮点数）

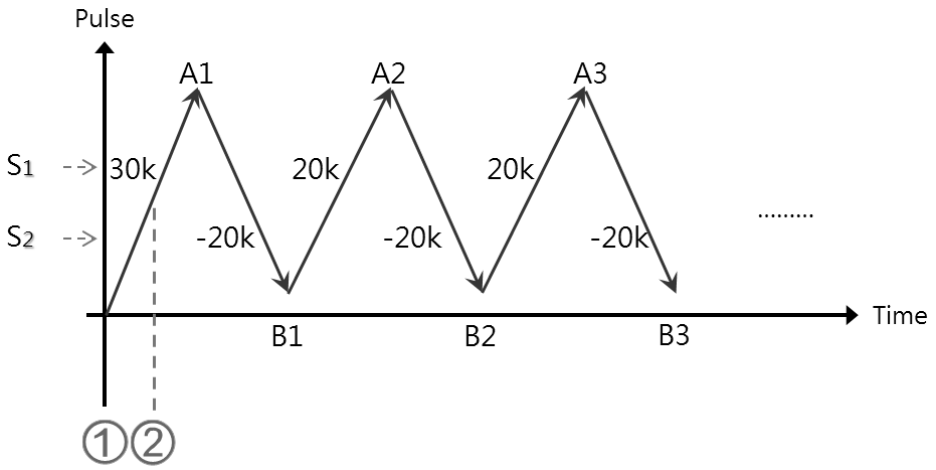
D_1 ： 脉冲输出装置

D_2 ： 可变目标速度区指示

指令说明：

1. 此指令是依照两个相对位置 S_1 与 S_2 计算出两个目标位置 在两个目标位置来回定位控制之高速输出指令，适合使用于整线形式之控制场合，如：纺织业的整经机台，或者电缆线整线机台等等。

2. 当指令初次启动输出时，**S_1** 与 **S_2** 一开始的相对位置需先指定，以利指令提前运算切换下一个输出。当启动输出之后，用户可修改即将执行之目标位置，但正在输出的目标位置已经不可变更。



①启动 DPPGB 指令，当 **S_1**:30,000；**S_2**:-30,000，往目标位置 A1=30,000 移动

②若在脉冲位置>15,000 时，变更相对位置 **S_1**:20,000；**S_2**:-20,000。

此时目标位置 A1=30,000，因为正在输出的目标位置已经不可变更，计算后的目标位置 B1=10,000（A1+ **S_2**），A2=30,000（B1+ **S_1**）

3. **S_3** 为目标速度来源（32 位整数），**S_4** 为目标速度微调比率值（浮点数），**S_3** 与 **S_4** 数值相乘后的数值（去小数取 32 位整数）范围为 1~200kHz，当超过此范围时，指令自动以最大或最小值处理。比率计算公式为 实际目标速度 = 目标速度 x **S_4**。举例：目标速度为 1kHz，微调比率浮点数为 1.2345，则实际输出目标速度为 1234Hz。

4. 当输出正在执行中，目标速度与微调比率可以被变更，而且将会在指令被扫描到时，变更实际输出速度。建议变更目标速度的变化幅度不可太大，以避免影响已经事先运算好的减速输出动作。

5. **D_1** 输出装置仅可选择 Y0.0、Y0.2、Y0.4、Y0.6、Y0.8 及 Y0.10 这 6 点，且会占用连续 2 点输出。其输出组合与输出模式 SR 如下表所示：

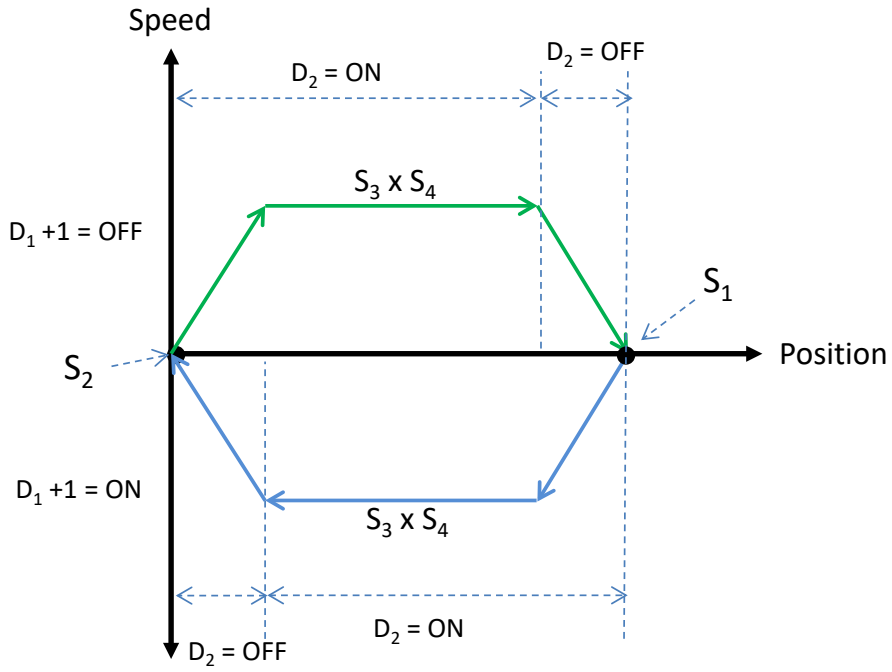
输出轴编号	1	2	3	4	5	6
D_1+0 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
D_1+1 输出点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
输出模式选择	SR462	SR482	SR502	SR522	SR542	SR562

6. **D_2** 为目标速度与微调比率可变化之标志指示，当此标志为 ON 时，即表示可变更正在执行的目标速度，当标志由 ON 变 OFF 时，即表示进入减速区，并且此时的目标速度已经被记录为下一阶段输出的目标速度。

7. 此指令无使用次数限制，但每次指令启动时，其指定高速输出轴不可被其它指令启动占用，否则将会无法执行。

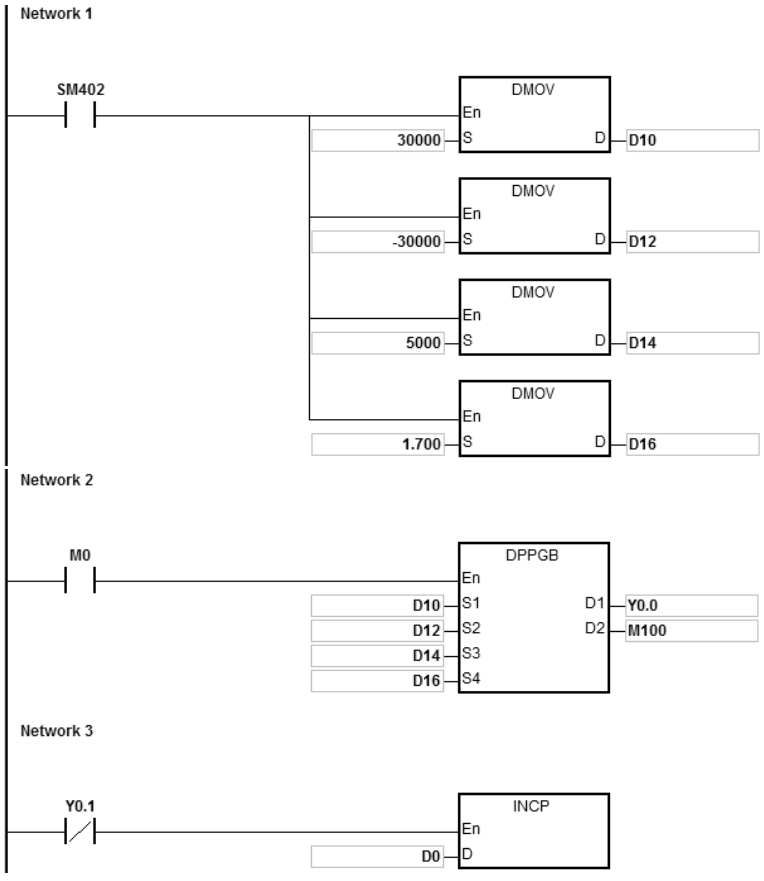
8. 此指令不建议使用于 ST 结构化语言编辑程序、中断程序以及仅被呼叫一次性功能块（FB）程序。

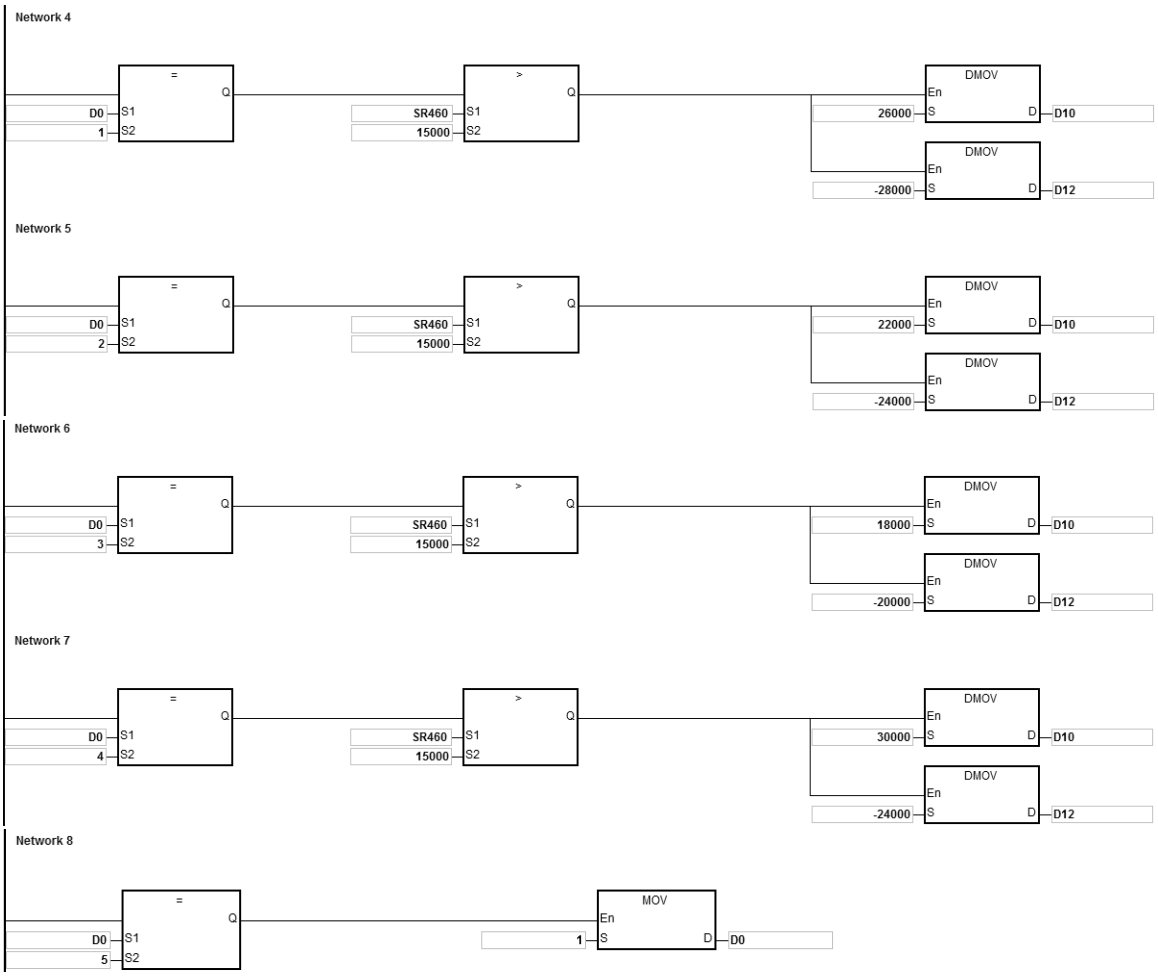
输出时序图：



范例程序 1：

当 M0=ON 时，DPPGB 以实际目标速度为 8500Hz (5000*1.7)，从 Y0.0 依照相对位置 **S₁** 及 **S₂** 计算出目标位置，开始往返运动，Y0.1=OFF 时表示为正方向。



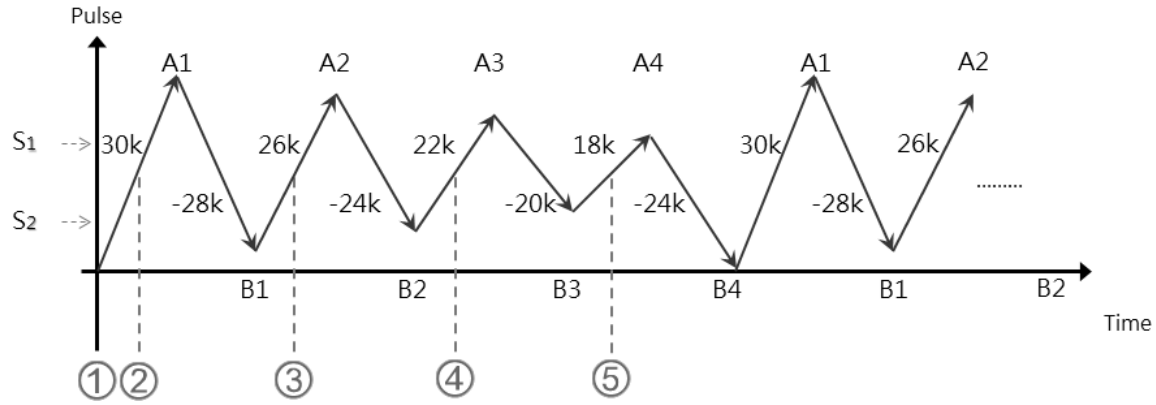


6

说明：

1. 在指令输出后，变更 **S_1** 相对目标位置 A 与 **S_2** 相对目标位置 B：

   a) Y0.0 输出曲线图：



   b) 说明:

   ①当 **S_1**:30,000；**S_2**:-30,000，M0 启动 DPPGB 指令，往位置 A1=30,000 移动

   ②当 D0=1，且 SR460 脉冲位置>15,000 时，变更相对位置 **S_1**:26,000；**S_2**:-28,000。

   此时目标位置 A1=30,000，因为正在输出的目标位置已经不可变更

   PLC 计算后的目标位置 B1=2,000（A1-28,000），A2=28,000（B1+26,000）

③当 D0=2，且 SR460 脉冲位置>15,000 时，变更相对位置 **S_1**:22,000；**S_2**:-24,000。

此时目标位置 A2=28,000，因为正在输出的目标位置已经不可变更

PLC 计算后的目标位置 B2=4,000（A2-24,000），A3=26,000（B2+22,000）

④当 D0=3，且 SR460 脉冲位置>15,000 时，变更相对位置 **S_1**:18,000；**S_2**:-20,000。

PLC 计算后的目标位置 B3=6,000（A3-20,000），A4=24,000（B3+18,000）

⑤当 D0=4，且 SR460 脉冲位置>15,000 时，变更相对位置 **S_1**:30,000；**S_2**:-24,000。

PLC 计算后的目标位置 B4=0（A4-24,000），A1=30,000（B4+30,000）

步骤	S_1:相对位置	S_2:相对位置	目标位置 B	目标位置 A
①	30,000	-30,000	-	A1=30,000
②	26,000	-28,000	B1=2,000	A2=28,000
③	22,000	-24,000	B2=4,000	A3=26,000
④	18,000	-20,000	B3=6,000	A4=24,000
⑤	30,000	-24,000	B4=0	A1=30,000

API	指令码			操作数	功能
2724	D	ZRN2		**TFreq**、**JFreq**、**Mode**、**DOG**、**NL**、**Pulse**、**Dir**	原点回归 2（可指定方向脚）

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
TFreq							●	●	●		○		○	○		
JFreq							●	●	●		○		○	○		
Mode							●	●	●		○		○	○		
DOG	○		○													
NL	○		○													
Pulse		○														
Dir		○	○													

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
TFreq			●				●						
JFreq			●				●						
Mode			●				●						
DOG	●												
NL	●												
Pulse	●												
Dir	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

6

符号：

TFreq ： 回归目标频率

JFreq ： 碰原点（DOG）后的寸动（JOG）频率

Mode ： 回原点模式选择

DOG ： 原点（DOG）输入设备

NL ： 反极限输入设备

Pulse ： 脉冲输出装置

Dir ： 方向输出装置

指令说明：

1. 此功能韧体支持版本为 V1.04.90 版以上。
2. **TFreq** 回归目标频率与 **JFreq** 寸动频率范围 1Hz~200kHz，**JFreq** 寸动频率需比 **TFreq** 目标频率小，且 **JFreq** 寸动频率即为启动频率。若 **TFreq** 小于 **JFreq** 频率，则 **TFreq** 自动被修正，与 **JFreq** 频率相同。

3. **DOG**、**NL**、**Pulse**、**Dir** 操作数请勿执行中变更；**DOG**、**NL** 操作数输入点建议使用 X0.0 ~ X0.15 这 16 点高速输入点（不受 PLC 扫描指令影响），若是采用一般 X1.0 之后输入点或 M 装置时，则执行停止、极限动作将受到 PLC 扫描时间影响。

4. **DOG**、**NL** 及 **Dir** 说明如下表

DOG 输入点	可任意选择 X，M 装置，但不同轴输出不可选择相同输入点 若 **DOG** 点使用 X0.0~X0.15，有震动或开关弹跳的状况，请由 HWCONFIG 设定输入点滤波时间
NL 输入点	可任意选择 X，M 装置，但不同轴输出不可选择相同输入点
Dir 方向脚	可任意选择 Y、M 装置

韧体 V1.10（含）以下，于执行回原点过程中，并且有启动极限点检查功能时，若发生碰极限的情况，不会设定极限警报标志。韧体 V1.12（含）以上，将会自动设定极限警报提示，之后由用户自行判断此警告标志，决定后续运行动作。（注：当警告标志设定后，须由用户自行清除）

注意事项：DZRN2 不支持正极限功能。

5. 选择 **Pulse** 输出点，请参考底下相对应之 SR 与 SM 参数表

Pulse 输出点	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
方向脚提前输出时间设定（单位 1ms）	SR640	SR642	SR644	SR646	SR648	SR650
输出中标志	SM460	SM480	SM500	SM520	SM540	SM560
输出完成标志	SM461	SM481	SM501	SM521	SM541	SM561
输出当前位置	SR460 SR461	SR480 SR481	SR500 SR501	SR520 SR521	SR540 SR541	SR560 SR561

Pulse 输出点	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
方向脚提前输出时间设定（单位 1ms）	SR641	SR643	SR645	SR647	SR649	SR651
输出中标志	SM472	SM492	SM512	SM532	SM552	SM572
输出完成标志	SM473	SM493	SM513	SM533	SM553	SM573
输出当前位置	SR474 SR475	SR494 SR495	SR514 SR515	SR534 SR535	SR554 SR555	SR574 SR575

6. **Mode** 为原点回归模式选择，其参数与功能代码分为上、下 16 位两个设定值，说明如下表

	Mode 为原点回归模式选择				
上 16 位	下 16 位				
b31~b16	b15~b7	b6	b5	b4	b3~b0
移动个数	保留	DOG 点遮罩功能设定 0：关闭 1：启动	设定 **NL** 信号 0：A 接点 1：B 接点	设定 **DOG** 信号 0：A 接点 1：B 接点	模式设定 0~2， 8~10

其功能代码分为上、下 16 位两个参数设定，其详细说明请参考下表：

功能项目	代码		说明
	高 16 位	低 16 位	
反向离开原点停止（模式 0）	0	0	指令启动时，固定往反方向的目标速度寻找原点，当碰到原点时（OFF→ON）减速至寸动速度，并且持续以反方向行进，直到原点信号由 ON 变 OFF 时停止。
正向离开原点停止（模式 1）	0	1	指令启动时，固定往反方向的目标速度寻找原点，当碰到原点时（OFF→ON）立即减速停止，然后以寸动速度往正方向行进，直到原点信号由 ON 变 OFF 时停止。
模式 0 回原点后再移动	移动个数	2	回原点动作如同低 16 位代码 0 的动作，当回到原点之后，将再以移动个数指定的数量移动。当高 16 位个数为正的数值往正方向移动，反之，负的数值就往反方向移动。
正向离开原点停止（模式 1）	0	8+0=8 8+1=9 （bit3=ON）	同代码 1（模式 1）的执行动作
模式 1 回原点后再移动个数	移动个数	8+2=10 （bit3=ON）	回原点动作如同低 16 位代码 1 的动作，当回到原点之后，将再以移动个数指定的数量移动。当高 16 位个数为正的数值往正方向移动，反之，负的数值就往反方向移动。
DOG 点为 B 接点		+16 （bit4=ON）	当低 16 位代码 bit4 为 ON 时，表示 **DOG** 点 ON□OFF 时为碰到原点，OFF□ON 表示离开原点。
NL 点为 B 接点		+32 （bit5=ON）	当低 16 位代码 bit5 为 ON 时，表示 **NL** 反极限点 ON□OFF 时为碰到极限，OFF□ON 表示离开原点。

7. 上表内的低 16 位代码数值判断与执行顺序为：判断 bit5，bit4 决定 **NL** 反极限与原点（DOG）信号 → 判断 bit3 选择模式 0 或 1 方式回原点 → 执行 bit1 与 bit0 的回原点动作

8. DOG 遮罩功能设定，韧体 V1.12.（含）以上支持，当遮罩功能启动（bit6 设定为 1），Mode+1 为 DOG 点遮罩个数设定，数值小于等于 0，视为不启动 DOG 点遮罩。以 DZRN2 指令启动后的输出脉冲个数（不分正负向）作为遮罩区的判断依据。由于每次找原点的起始位置可能不一样，因此遮罩区的设置很可能会出现错过原点的情况；故请依实际应用情况评估是否要设置遮罩功能，或设置负极限功能防范。

   EX：将 MODE 参数设为 D0=64（bit6 设定为 1）；将 D2=120000（遮罩个数设定），当 DZRN2 启动后，输出脉冲小于 120000 内的 DOG 讯号都会被遮蔽（无作用）。

9. 当指令运行完成指定功能之后，其输出结束标志才会产生，例如：选择代码 2 的功能，则会在移动个数输出完成时，PLC 才会设定完成标志。
10. 当 DZRN2 指令启动之后，若是指定原点输入点与程序中外部输入中断同一个时，则中断服务程序将暂时不能被执行；且须直到 DZRN2 指令被关闭之后，才能恢复使用。
11. 此 DZRN2 指令不建议使用于 ST 语言编辑、中断程序以及仅被执行一次性的功能块编辑。

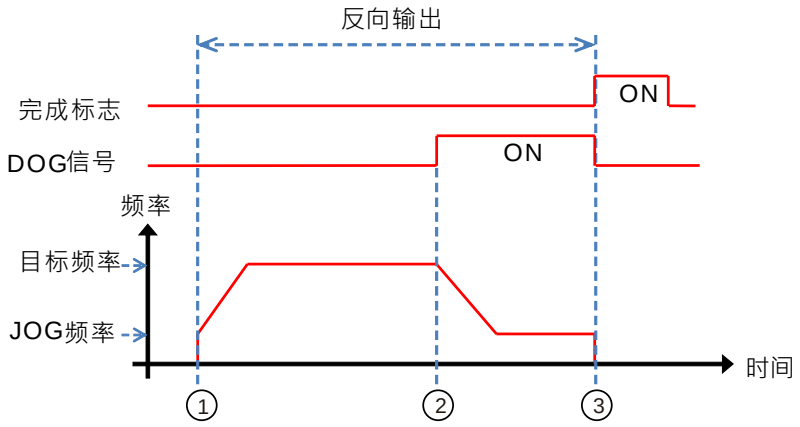
    韧体 V1.12（含）以上支持 ST 语言编辑，请参考范例说明。
12. 各功能之动作示意图如下：

    功能代码 0：

    ① 启动 DZRN2 功能，固定往反方向的目标速度寻找原点

    ② 碰到近点 DOG 信号，减速至 JOG 速度，并且持续以反方向行进，直到原点信号由 ON 变 OFF 时停止

    ③ 离开近点 DOG 信号，信号由 ON 变 OFF 时停止

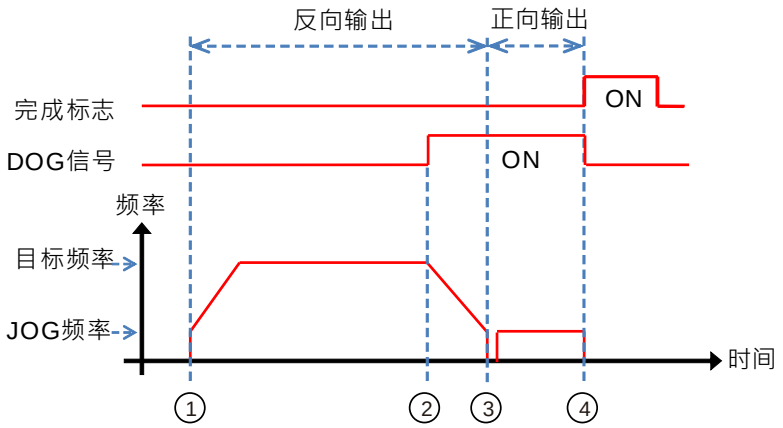


    功能代码 1：

    ① 启动 DZRN2 功能，固定往反方向的 **TFreq** 目标速度寻找原点。

    ② 碰到近点 **DOG** 信号之后减速停止，并且在停止之后，再以 **JFreq** 速度往正方向行进，直到 **DOG** 信号由 ON 变 OFF 时停止。

    ③ 离开 **DOG** 信号，信号由 ON 变 OFF 时立即停止。



    功能代码 2：

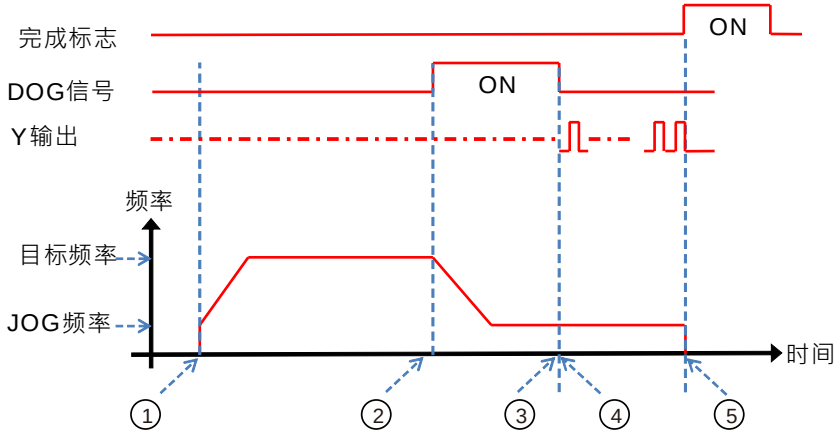
① 启动 DZRN2 功能，固定往反方向的目标速度寻找原点。

② 碰到 **DOG** 信号，减速至 **JFreq** 速度，并且持续往反方向行进。

③ 离开 **DOG** 信号，信号由 ON 变 OFF 时，将再以移动个数指定的数量移动。

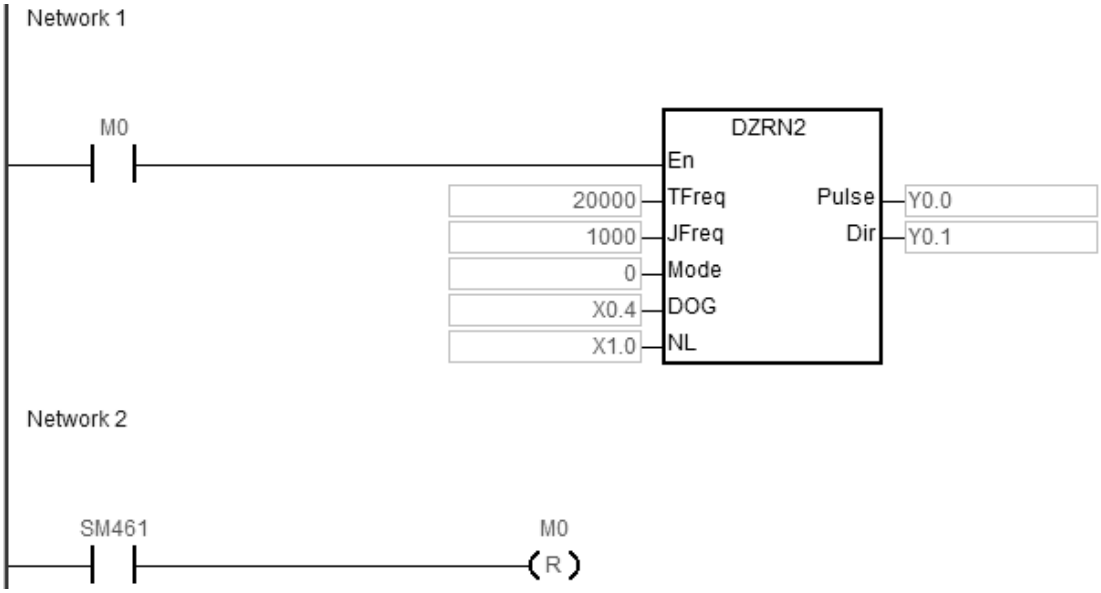
④ 第 1 个脉冲开始输出

⑤ 第 100 个脉冲输出完毕后停止，且完成标志为 ON。



程序范例一：

当 M0=ON 时，以 20kHz 频率从 Y0.0 输出脉冲开始往反方向寻找原点，当碰到原点信号（DOG）X0.4=ON 时变成以寸动速度 1kHz 继续往反方向移动，直到 X0.4 从 ON 变 OFF 时立即停止。



说明：

Mode 设定为 0

高 16 位[0000]=不启动移动个数

低 16 位[0000]=模式 0，碰到 DOG 往反方向移动，脱离 DOG 立即停止

程序范例二：(以 ST 结构化语言编辑)

一般使用此指令时，建议优先采用梯形图语言编辑，若采用 ST 语言时 (V1.12 以上支持)，须等到完成回原点之后 (ex：SM461 完成标志 ON)，再关闭指令，而且必须搭配自动复位标志，以通知 PLC 归还高速输出控制权。以 Y0.0 输出为例的建议程序写法如下图。

```
0001 IF M0 THEN
0002     DZRN2(1000,100,0,X0.0,X1.0,Y0.0,Y0.1);
0003     M1:=TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
```

6.27 台达专用 CANopen 与 EtherCAT 通讯指令

6.27.1 通讯指令一览表

API	指令码（位）		P 指令	功能	CANopen		EtherCAT	
	16	32			伺服	变频	伺服	变频
2800	INITC	–	–	专用 CANopen 通讯初始化	V	V	–	–
2801	ASDON	–	–	驱动器启动与停止	V	V	V	V
2802	CASD	–	–	驱动器加减速设定	V	V	V	V
2803	–	DDRVIC	–	伺服相对位置定位控制	V	–	V	–
2804	–	DDRVAC	–	伺服绝对位置定位控制	V	–	V	–
2805	–	DPLSVC	–	驱动器速度控制	V	V	V	V
2806	ZRNC	DZRNC	–	伺服原点回归	V	–	V	–
2807	COPRW	–	–	CANopen 通讯读写数据	V	V	–	–
2808	COPWL	DCOPWL	–	CANopen 多笔参数写入	V	V	–	–
2809	RSTD	–	–	发送重置或 NMT 命令	V	V	V	V
2810	ZRNM	–	–	伺服原点模式设定	V	–	V	–
2811	EMER	–	–	读取 Emergency 信息	V	V	V	V
2812	–	DCSFOC	–	通讯型控制伺服追随功能	V	–	V	–
2813	ECAMD	–	–	E-CAM 表下载至从站	V	–	–	–
2814	–	DECAMS	–	E-CAM 参数设定与规划	V	–	–	–
2815	ECAMA	–	–	E-CAM 启动与状态显示	V	–	–	–
2816	ECAMC	–	–	变更 E-CAM 表起始位置与长度	V	–	–	–
2817	–	DTQC	–	扭矩控制	V	V	V	–
2818	–	DTQLC	–	扭矩限制之速度控制	V	V	V	–
2820	INITEC	–	–	专用 EtherCAT 通讯初始化	–	–	V	V
2821	ECATRW	–	–	EtherCAT SDO 数据读写	–	–	V	V
2822	MOTST	–	–	EtherCAT 从站 Motion 状态读取	–	–	V	V
2823	ECATST	–	–	EtherCAT 从站状态读取	–	–	V	V

6.27.2 CANopen 与 EtherCAT 通讯指令说明

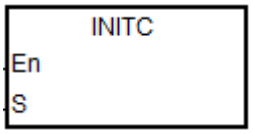
API	指令码	操作数	功能
2800	INITC	**S**	专用 CANopen 通讯初始化

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S													○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●											

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



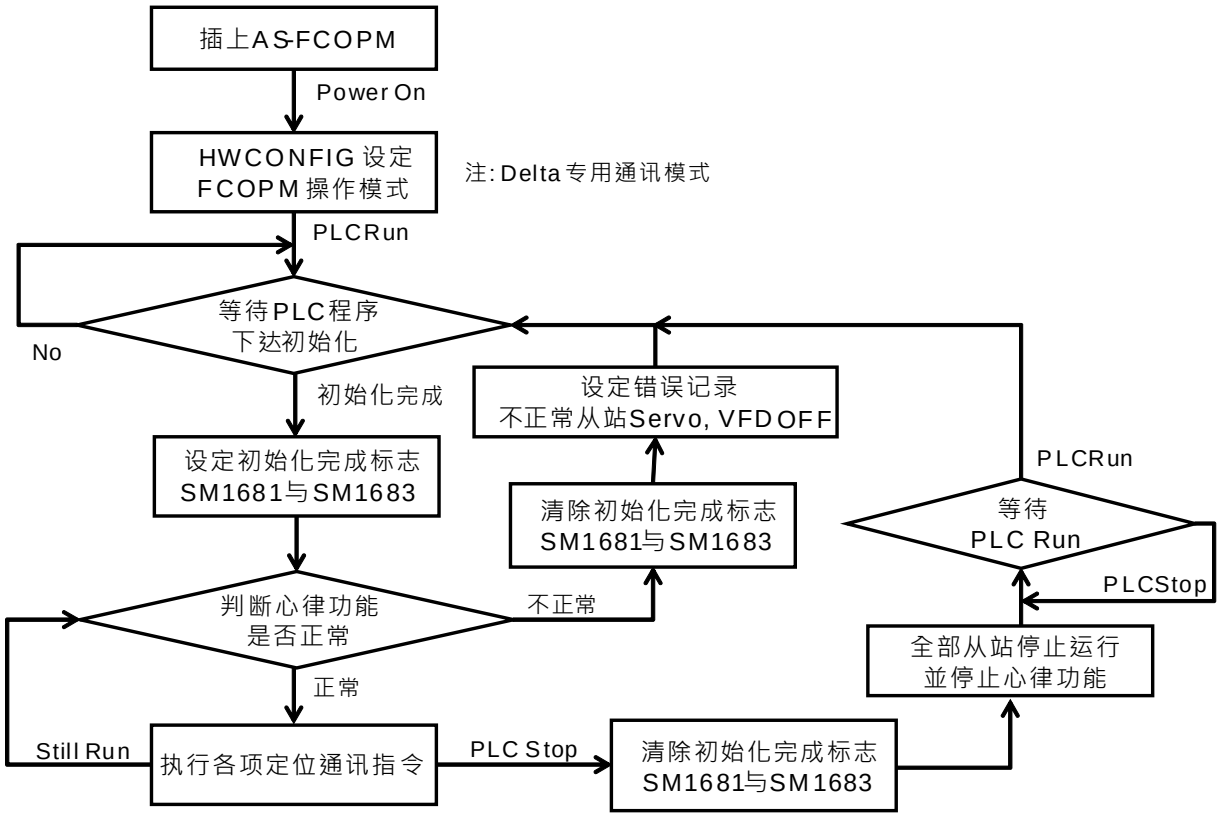
S ： 初始化伺服轴数

指令说明：

1. 执行前初始化指令前，若主机为 AS300 型主机，请确认硬件配置（HWCONFIG）是否已经选择功能卡 2 的工作模式为台达专属驱动器模式。若主机为 AS200 或 AS100 型主机，则请确认内建 CAN 通讯口的工作模式是否已选择台达专属驱动器模式。
2. **S** 可指定范围为 1~8，(此范围为伺服专用站号区)，当数值超出此范围时，则 PLC 将再判断是否为变频器专用区。注意：伺服站号设定只能从站号 1 开始编号，其中间站号不可以跳过或保留。
3. 主机韧体版本为 V1.02.00 版以上，新增 **S** 可指定范围 21~28 (此范围为变频器专用站号区)，当数值不为此范围时，此指令将不会进行任何初始化动作。注意：变频器站号一样必须连续。
4. 当初始化站号设为 5 时，PLC 会针对站号 1~5 进行初始化，当站号设为 23 时，PLC 会对站号 21~23 进行初始化。举例：若是设备上有 5 台伺服与 3 台变频器时，则需先下达 INITC 5，等完成初始化后，再下 INITC 23。
5. 指令刚启动（SM1686 为 OFF)，并且 **S** 指定范围为 1~8，则 SM1681 会被清除为 OFF，以及所有从站 1~8 都会全部被重置为未初始化状态，之后等被指定初始化的最后一台从站，已经响应完成初始化，则 SM1681 会被设置为 ON。当 **S** 指定范围为 21~28，则 SM1683 会被清除为 OFF，以及所有从站 21~28 都会全部被重置为未初始化状态，之后等被指定初始化的最后一台从站回应已完成初始化，则 SM1683 会被设置为 ON。注意：这两个初始化完成标志，仅是通知当下初始动作已完成，若后续有使用参数设定指令时，这两个 SM 将会变成该指令的完成标志。
6. 若执行通讯过程中发生错误，SM1682 将被设定为 ON；另外，在 SR658 会记录发生错误的站号及 SR659 会记录错误代码。(此处通讯错误标志与代码皆为所有台达专用 CANopen 通讯指令共享)

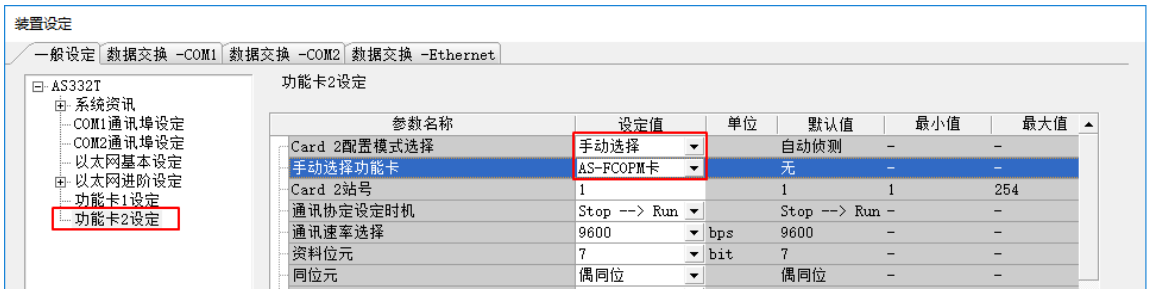
7. 主机韧体版本 V1.06.40（含）以上，新增 SM1686 标志为 ON 时，此指令初始化动作变为指定单台初始化模式，举例：原先 SM1686=OFF 时，INITC 3 表示初始化从站站号 1、2、3 共三台，接着，当 SM1686=ON 时，INITC 6 表示只初始化从站站号 6；经过上述两个初始化动作之后，表示站号 1，2，3，6 共四台从站已经被启动。补充说明：专用模式所对应的从站信息，都是在初始化完成后开始进行。
8. 连续站号初始化的动作，请务必于指定单台初始化前启用，若无法先连续再单台初始化时，请全部都采用单台初始化模式
9. 当初始化完成后，一旦发生心律保护错误（Heartbeat error）的状况（SR0 = 0x1900~0x191C，后 2 码为从站 ID），并且为连动模式（默认值）（SM1684=OFF）时，则初始完成标志（SM1681 & SM1683）将自动清除为 OFF，并且暂停其它从站的相关动作；若用户在故障排除后，想再重新启动，则须重新对所有从站进行初始化动作，然后才可再进行后续执行动作。PLC 将在确认初始化完成后，自动关闭慢闪的错误灯号，但会留下错误记录。
10. 若需各站独立运行，不受其它站故障影响（前提通讯线正常连接），则可以在初始化后，设定 SM1684 为 ON 通知 PLC 仅记录错误与设定对应的错误标志即可，不需让其它通讯从站停止运行。
11. 驱动器的通讯站号与速率，用户须先行手动设定后，再将驱动器断电并重新上电一次，此时驱动器才会以最新设定值运行。
12. 此指令不建议使用于 ST 结构化语言编程程序、中断程序以及仅被一次性调用的功能块（FB）程序。

初始化与执行之示意流程图：

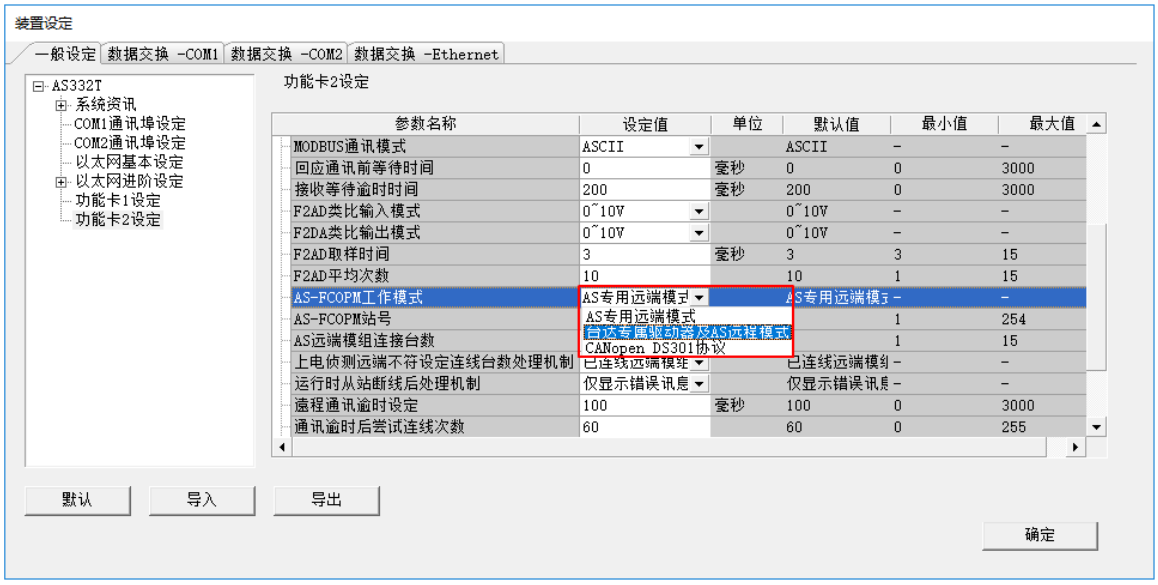


程序范例：(与台达伺服 ASD-A2M 通讯)

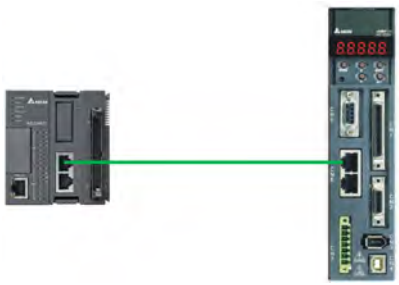
1. 当使用台达 CANopen 专用指令前，请将 AS-FCOPM 功能卡安装至 AS 主机上 (仅支持卡槽 2)，并确认硬件配置 (HWCONFIG) 是否已经选择功能卡 2 的工作模式为[台达专属驱动器与 AS 远程模式]；并且将通讯速率设定为 1000kbps，或其它通讯速率(支持四种通讯速率，分别为 125k、250k、500k 及 1000kbps)



图一、功能卡选择画面



2. 使用 CANopen 通讯电缆，连接 AS 系列与 ASDA 系列，如下列示意图。(AS-FCOPM 通讯卡上 120Ω需切为 ON 状态，表示启用终端电阻)。



3. 使用 ASDA 面板进行基本设定，步骤如下：

   a. 将伺服 P2-08 设置为 10，恢复原厂设定。

   b. 将伺服断电后重新上电

   c. 将 P1-01 设置为 0001 (PR 模式)

   d. P3-01 设置为 0403，将伺服 CAN 速率设为 1.0Mbit/s。

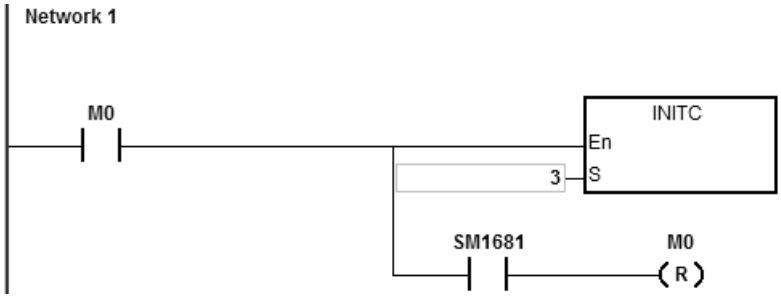
   e. 站号设定=依照需要的台数，分别设置每台伺服的 P3-00，请依序设定为 1、2、3。最多可设定 8 台。

f. 将伺服断电后重新上电

g. 基本设定完成后，即可开始操作。

4. 范例程序下载，并将 M0 设为 ON 时（SM1686 默认为 OFF），指令将开始自动对伺服站号 1~3 下达初始化设定，直到 SM1681 被设为 ON，表示已经初始化完成。(伺服成功进入 CANopen 模式时，也会显示一下 Co-Ld 的信息)

梯形图：



ST 语言：

```
0001 IF M0 = TRUE AND SM1681 = FALSE THEN
0002    INITC(3);
0003 END_IF;
0004 IF M0 = TRUE AND SM1681 = TRUE THEN
0005    M0 := FALSE;
0006 END IF;
```

5. AS 初始化流程对伺服驱动器所设定的内容如下：

A. 设定 P2-30(辅助机能)为 5，通知伺服不需回存参数至 EEPROM 做永久保持，以保护伺服寿命。

B. 清除 P6-02(PATH#1)为 0，以及 P6-06(PATH#3)为 0。(表示清除 PR 模式的 PATH#1 & #3)

C. 设定 P3-06(SDI 来源选择)为 16#0100。(表示 DI1~DI8 由硬件控制，EDI9 由软件控制，EDI10~EDI14 由硬件控制)

D. 清除 P4-07(手动控制 SDI 状态)为 0。

E. 设定 P2-36(选择 EDI9 功能)为 16#0101。(表示 EDI9 为 Servo ON 功能)

F. 设定 P0-17(映射参数 CM1A)为 1。(表示映像参数为脉冲命令输出寄存器 CMD_O)

G. 设定 P0-18(映射参数 CM2A)为 64。(表示映像参数为脉冲命令寄存器 CMD_E)

H. 设定 P5-20~P5-35(加减速时间)为 1。(表示加减速时间为 1ms)

I. 设定 P5-60~P5-75(目标速度)为 1。(表示目标速度为 0.1rpm)

J. 设定 PDO1 对应 P5-07(PR 命令)、P0-01(警报代码)与 P0-46(DO 点状态) 与 P4-07（DI 点状态）。

K. 设定 PDO2 对应 P0-09(CM1 状态为 CMD_O)与 P0-10(CM2 状态为 CMD_E)。

L. 设定 PDO3 对应 P0-11（CM3 状态为当前扭矩）。

M. 设定心律（Heartbeat）功能的时间与 PLC 扫描逾时相同（默认值 200ms），当主站与从站之间没有通讯状况的时间超过逾时时间，则 PLC 发出警告信息（错误灯慢闪）(参考错误信息说明)，伺服自动 Servo OFF。(主机韧体版本 V1.04.00（含）以上支持。)

N. 依照 SM1685 设定状况，判定是否启动 PDO3 读取伺服 P0-12 自定义参数数值（用户须事先设定伺服 P0-20 的 CM4A），若有启动（SM1685=ON）读取，PLC 将自动读取至 SR791~SR806 存放。(每一

站伺服皆保留 2 个 word 空间，实际自定义参数类型，由用户自行决定）

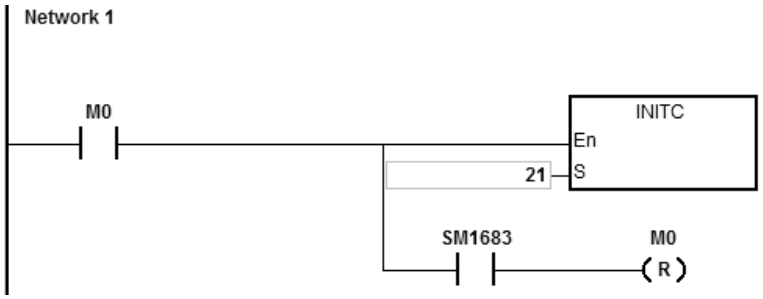
注：主机韧体版本 V1.04.10（含）以上支持扭矩（上述说明 PDO3）与 DI 状态（上述说明 PDO1）。主机韧体版本 V1.04.60（含）以上支持读取自定义数值（上述说明 PDO3）。

6. 上述的项次编号 A、B、F、G、J、K、L、N 这 8 项伺服设定参数，用户请勿于初始化完成之后，继续使用 COPRW 指令进行修改。
7. 当用户选用绝对型伺服时，必须自行再增加 COPRW 通讯指令，去针对 P3-12 写入 16#0100 数值，以通知伺服在断电瞬间将相关绝对型有关的参数写入 EEPROM 内。
8. 如需更改 ASD-A2 硬件 DI 信号配置，可自行设定 ASD-A2 硬件 DI 信号配置相关参数或透过 COPRW 指令变更；当采用 COPRW 变更时，请在 INITC 完成后，并且伺服启动前，进行配置修改。
9. 当初始化完成之后，伺服将会进入 PR 模式，之后用户请勿再对伺服 P5-18 命令位置参数进行通讯控制。
10. 以上介绍的伺服参数，仅提供部分说明，若用户须了解更多详细信息，请自行参考伺服操作手册。

范例二：（与台达变频器通讯）

1. 手动设定变频器 P09-36 站号（范围限定 21~28 之间），P09-37 通讯速率 （变频默认为 1MHz），P09-40 为标准 CANopen 协议（变频默认值）。
2. 手动设定命令来源 P00-20 为 6，以及 P00-21 为 3，设定由 CANopen 通讯控制。
3. AS 初始化流程对变频器所设定的内容如下：
   a. 设定 PDO 交换区，将变频器当前运转频率、当前状态、当前转矩与 DI 状态对应至 PLC 指定的 SR。
   b. 设定心律（Heartbeat）功能的时间与 PLC 扫描逾时相同（默认值 200ms），当主站与从站之间没有通讯状况的时间超过逾时时间，则 PLC 发出警告信息（错误灯慢闪）（参考错误信息说明），变频器自动停止运转。
4. 以上介绍的变频器参数，仅提供部分说明，若用户须了解更多详细信息，请自行参考变频器手册。

范例程序：



程序说明：

当 M0 设为 ON 时，指令将开始自动对变频器站号 21 下达初始化设定，直到 SM1683 被设为 ON，表示已经初始化完成。

补充说明：

初始化过程相关特殊标志与寄存器：

功能项目	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
台达专属驱动器功能	R/W	请参照下述范例之硬件配置（HWCONFIG）设定，Stop→Run 时启动							
INITC 与 CASD 通讯完成标志	R	SM1681							
通讯错误标志	R	SM1682							
单台初始化模式启动标志（V1.06.40 版以上支持）	R/W	SM1686=OFF（默认）为连续站号初始化模式 SM1686=ON 为指定单台初始化模式							
通讯错误站编号	R	SR658							
通讯错误代码	R	SR659							

初始化错误时，SR659 可能发生的通讯错误代码：

错误代码	发生原因
0x0002	从站未响应 SDO 信息。
0x0003	从站接收信息错误，或从站响应通讯错误。
0x0006	INITC 初始化时，有其中一站正被使用。
0x0007	从站站号超出 INITC 指令的指定范围。
0x0008	指定单台从站重置过程失败，需再确认通讯线与从站是否正常运行。

使用台达专属驱动器模式，并且开始通讯之后，其站号 1~15 以及站号 21~28 的 PDO 数据更新时间，可参考下表计算：(举例：1M 速度时，PDO 更新周期时间为 10ms*5 次 = 50ms)

通讯速率（bps）	1M	500K	250K	125K	50K
内定同步时间（ms）#1	10	15	30	60	80
内定同步次数（次）#2	5	5	5	5	5

注#1：若选择是台达专属+DS301 混合模式，可使用 CANopen Builder 软件修改从站的同步时间，但只能大于上列之内定同步时间，若设定值低于上列时间，则以内定同步时间执行。

注#2：若选择是台达专属+DS301 混合模式，可使用 CANopen Builder 软件指定 DS301 表内的同步次数，上列专属驱动器的内定同步次数不可变更。

使用台达专属驱动器模式，其 COPRW 指令（SDO 通讯）内定通讯逾时如下表所示之时间：

通讯速率（bps）	1M	500K	250K	125K	50K
内定通讯逾时（ms）	100	100	200	400	500

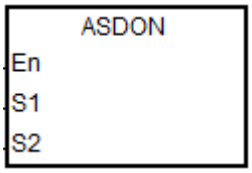
API	指令码	操作数	功能
2801	ASDON	**S_1 · S_2**	驱动器启动与停止

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2													○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 驱动器站号

S_2 ： 启动与停止模式

指令说明：

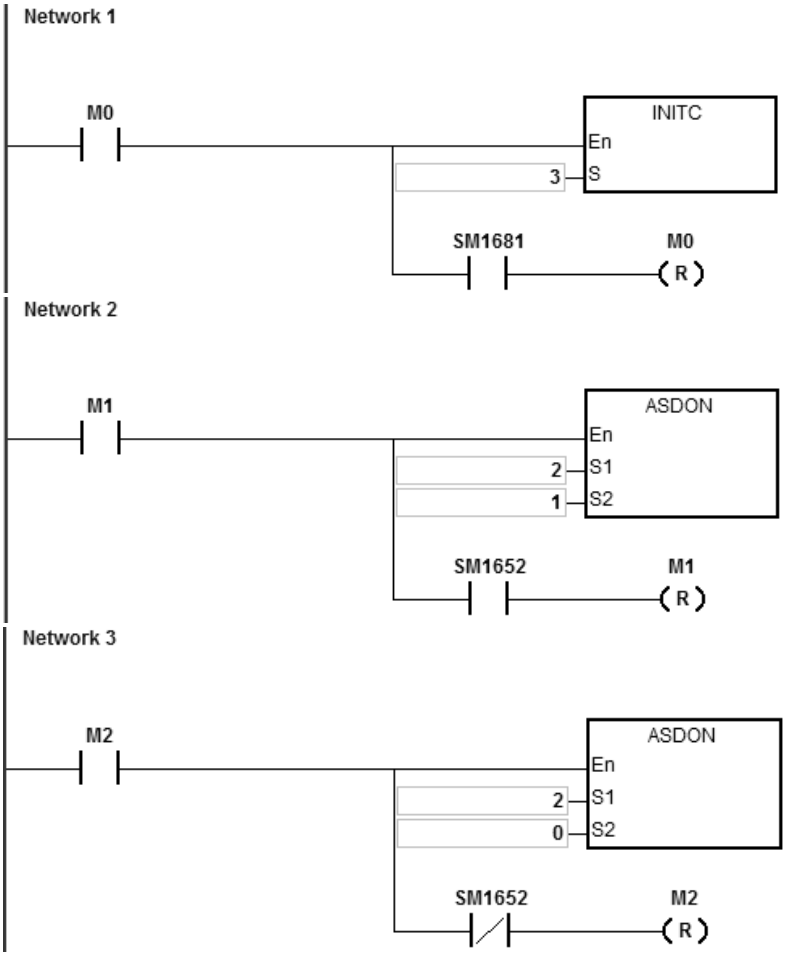
1. 主机韧体版本 V1.14.00 (含) 以上，ISPSoft 编辑软件需 V3.16 版以上，支持 EtherCAT (以下简称 ECAT) 通讯，须搭配 AS-FECAT 功能卡。
2. 执行此指令前，使用 CAN 通讯，必须先完成指令 INITC；使用 ECAT 通讯时，必须先完成指令 INITEC。
3. 使用 CAN 通讯时，**S_1** 可指定范围为 1~8 (伺服) 或 21~28 (变频)，超出此范围指令不执行；使用 ECAT 通讯，S_1 可指定范围为 1~16，超出此范围指令不执行，指定的站号，其装置种类，依 INITEC 指令设定。
4. 当 **S_1** 为指定伺服 (1~8 (CAN) / 1~16 (ECAT))，并且 **S_2** 输入非零的数值，皆表示执行启动 SERVO-ON 动作 (固定为位置控制模式)；输入 0 则表示执行停止 SERVO-OFF 动作。
5. 当 **S_1** 为指定变频 (21~28 (CAN) / 1~16 (ECAT))，并且 **S_2** 输入数值为 1，即表示执行变频器的速度控制模式；输入数值为 2 时，即表示执行转矩控制模式，输入数值为 0，则表示执行停止运转。变频器的速度与转矩模式可直接切换，不需先下达停止命令后，才再进行变换控制模式。备注：采用直接切换运转模式之前，请先确认相关变频器参数已经设定完成。
6. 上述 2 点说明整理如下表：

驱动器种类	**S_1** 站号范围	**S_2** 功能说明
伺服	1~8 (CAN) 1~16 (ECAT)	1=SERVO-ON 0=SERVO-OFF
变频器	21~28 (CAN) 1~16 (ECAT)	0=执行停止运转 1=速度控制模式 2=转矩控制模式

7. 使用 CAN 通讯，伺服各站（1~8）皆有分配显示状态标志（SM1651~M1658）；使用 ECAT 通讯，伺服各站（1~16）皆有分配显示状态标志（SM1651~SM1658，SM1621~SM1628）。由此标志可得知实际伺服的状态，标志 ON 表示 SERVO-ON，标志 OFF 表示 SERVO-OFF。
8. 使用 CAN 通讯，变频各站（21~28）皆有分配显示启动中标志（SM1621~M1628）；使用 ECAT 通讯，变频各站（1~16）皆有分配显示启动中标志（SM1651~SM1658，SM1621~SM1628），由此标志可得知变频的运转状态，标志 ON 表示运转中，标志 OFF 表示停止。
9. 若执行通讯过程中发生错误，SM1682 将被设定为 ON；另外，在 SR658 会记录发生错误的站编号及 SR659 会记录错误代码。

程序范例一（梯形图编程）：

1. 当 M0 = OFF → ON 时，指令开始对伺服站号 1~3 下达初始化设定，直到 SM1681 被设为 ON。
2. 当 M1 = OFF → ON 时，指令开始对伺服站号 2，启动伺服，SM1652 标志为 ON，表示 SERVO-ON。
3. 当 M2 = OFF → ON 时，指令开始对伺服站号 2，停止伺服，SM1652 标志为 OFF，表示 SERVO-OFF。



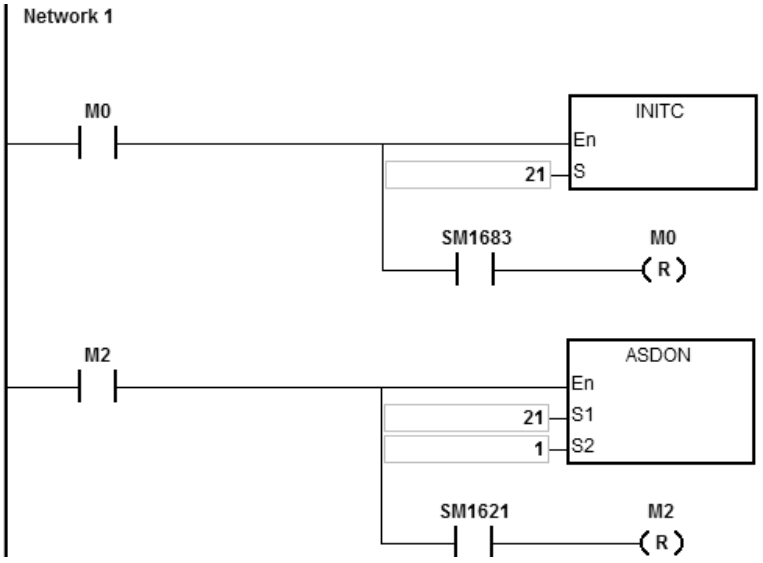
程序范例二（ST 语言编程）：

```
0001 IF M1 THEN
0002    ASDON(2,1);
0003 END_IF;
0004 IF M1 AND SM1652 THEN
0005    SM1582 := TRUE;
0006    M1 := FALSE;
0007 END_IF;
0008
0009 IF M2 THEN
0010    ASDON(2,0);
0011 END_IF;
0012 IF M2 AND (not SM1652) THEN
0013    SM1582 := TRUE;
0014    M2 := FALSE;
0015 END_IF;
```

1. 当 M1 为 ON 时，启动伺服站号 2 为 Servo ON 状态。当 SM1652=ON 时，即表示伺服已经 Servo ON。
2. 确认 M1 与 SM1652 同时为 ON 时，设定伺服站号 2 的自动复位标志 SM1582=ON，释放其通讯控制权，并让之后同样对站号 2 的通讯指令，可以继续执行。
3. 当 M2 为 ON 时，清除伺服站号 2 为 Servo OFF 状态。当 SM1652=OFF 时，即表示伺服已经 Servo OFF。
4. 确认 M2 为 ON 并且 SM1652 为 OFF 时，设定伺服站号 2 的自动复位标志 SM1582=ON，释放其通讯控制权，并让之后同样对站号 2 的通讯指令，可以继续执行。

程序范例三：(与台达变频器通讯)

1. 当 M0 = OFF → ON 时，指令开始对变频器站号 21 下达初始化设定，直到 SM1683 被设为 ON。
2. 当 M2 = OFF → ON 时，指令开始对变频器站号 21，启动变频器的速度控制模式，SM1621 标志为 ON，表示变频器运转中。



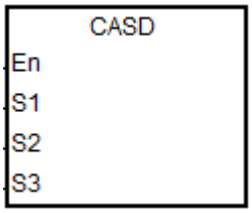
API	指令码	操作数	功能
2802	CASD	S_1 ， S_2 ， S_3	驱动器加减速设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2	●	●			●	●		●	●		○	○	○	○		
S_3	●	●			●	●		●	●		○	○	○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 驱动器站号

S_2 ： 加速时间（ms）

S_3 ： 减速时间（ms）

指令说明：

1. 主机韧体版本 V1.14.00(含)以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
2. 执行此指令前，使用 CAN 通讯，必须先完成指令 INITC；使用 ECAT 通讯，必须先完成指令 INITEC。
3. 使用 CAN 通讯，S_1 可指定范围为 1~8（伺服）或 21~28（变频），超出此范围指令不执行；使用 ECAT 通讯，S_1 可指定范围为 1~16，超出此范围指令不执行，指定的站号，其装置种类，依 INITEC 指令设定。
4. S_2 与 S_3 说明整理如下表：

驱动器种类	S_1 站号范围	S_2 与 S_3 数值范围	S_2 与 S_3 加减速说明
伺服	1~8 （CAN） 1~16 （ECAT）	1~32767（ms）	S_2 加速时间为伺服从 0 速度，开始加速到 3000.0 rpm 所需要的时间。 S_3 减速时间为伺服从 3000.0 rpm 速度减速到 0 所需要的时间。
变频器	21~28 （CAN） 1~16 （ECAT）	100~32700（ms） 须以 100 倍数输入	S_2 加速时间为变频器从 0.0Hz 加速到最高操作频率所需要的时间 S_3 减速时间为变频器从最高操作频率减速到 0Hz 所需要的时间

5. 当指令刚启动时，使用 CAN 通讯，若 S_1 指定站号为 1~8 之间，则 SM1681 会被清除为 OFF (表示修改参

数中)，当伺服已经响应接收命令，则 SM1681 会被设置为 ON。(表示参数修改成功)。若 **S_1** 指定站号为 21~28 之间，则 SM1683 会被清除为 OFF (表示修改参数中)，当变频器已经响应接收命令，则 SM1683 会再被设置为 ON (表示参数修改成功)；使用 ECAT 通讯，若 S_1 指定站号为 1~16，则 SM1681 会被清除为 OFF (表示修改参数中)，当伺服/变频已经响应接收命令，则 SM1681 会被设置为 ON。(表示参数修改成功)。

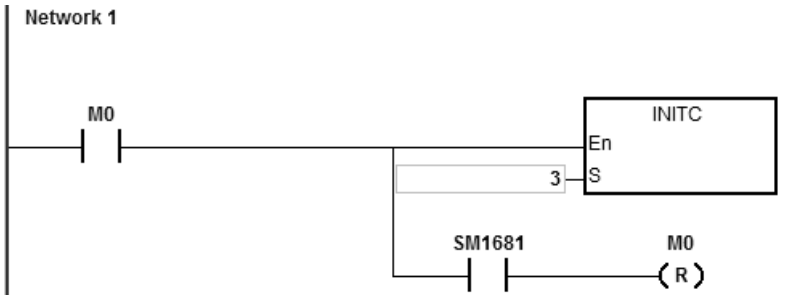
6. 若执行通讯过程中发生错误，SM1682 将被设定为 ON；另外，在 SR658 会记录发生错误的站编号及 SR659 会记录错误代码。

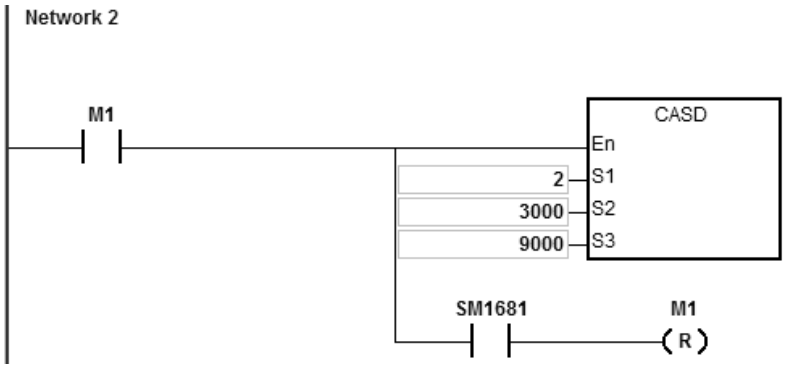
程序范例一 (与台达伺服 ASD-A2M 通讯)：

1. 当 M0 = OFF → ON 时，指令开始对伺服站号 1~3 下达初始化设定，直到 SM1681 被设为 ON。
2. 当 M1 = OFF → ON 时，当伺服站号 2 的目标速度为 3000 rpm 时，设定伺服站号 2 的加速时间 3000ms，减速时间 9000ms。
3. 若伺服站号 2 的目标速度为 1000 rpm，则加减速时间如下：

   加速时间：[3000ms/3000rpm]*1000rpm=1000ms

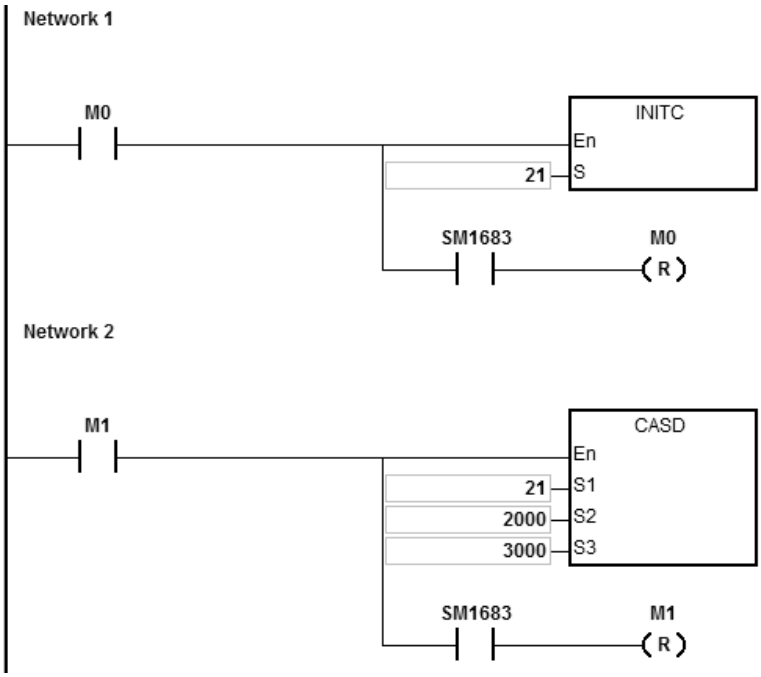
   减速时间：[9000ms/3000rpm]*1000rpm=3000ms





程序范例二：(与台达变频器通讯)

1. 当 M0 = OFF → ON 时，指令开始对变频器站号 21 下达初始化设定，直到 SM1683 被设为 ON。
2. 当 M1 = OFF → ON 时，设定变频器站号 21 的加速时间 2000ms，减速时间 3000ms。


Network 1
M0
INITC
En
21
S
SM1683
M0
R
Network 2
M1
CASD
En
21
S1
2000
S2
3000
S3
SM1683
M1
R

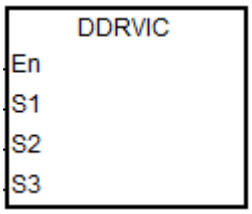
API	指令码			操作数	功能
2803	D	DRVIC		S_1 ， S_2 ， S_3	伺服相对位置定位控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●					○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 伺服站号

S_2 ： 相对型目标地址

S_3 ： 目标速度

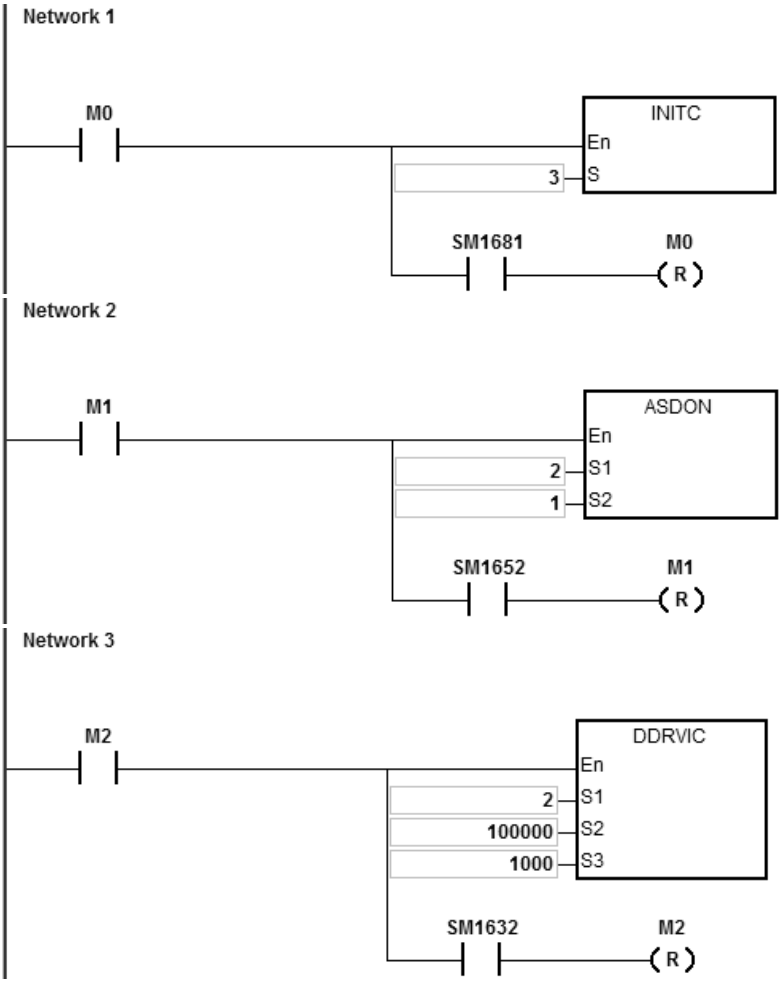
指令说明：

1. 主机韧体版本 V1.14.00(含)以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
2. 执行此指令前，使用 CAN 通讯，必须先完成指令 INITC 与 ASDON（SERVO ON）；使用 ECAT 通讯，必须先完成指令 INITEC 与 ASDON（SERVO ON）。
3. 使用 CAN 通讯，**S_1** 可指定范围为 1~8，超出此范围指令不执行；使用 ECAT 通讯，S_1 可指定范围为 1~16，超出此范围或是指定站号的装置种类不为伺服，指令不执行。
4. **S_2** 指定范围为 -2147483648 ~ 2147483647，正反号代表正反转方向。此目标地址为相对型位置。
5. **S_3** 单位为 0.1 rpm，输入数值范围为 1~60000，其表示 0.1~6000.0 rpm。
6. 当输出到达定位之后，使用 CAN 通讯，站号 1~8 各轴所对应的完成标志（SM1631~SM1638）将会被设置为 ON；使用 ECAT 通讯，站号 1~16 各轴所对应的完成标志（SM1631~SM1638，SM1921~SM1928）。定位完成标志在各轴启动输出时，PLC 会自动清除，若用户需要对相同输出轴进行连续定位动作，则请自行清除定位标志，以避免下一个连续输出定位指令在还未开始输出，即判断到定位完成标志。
7. 详细各轴对应的 SM 与 SR 表，请参考下页说明。
8. 若执行通讯过程中发生错误，SM1682 将被设定为 ON ;另外，在 SR658 会记录发生错误的站编号及 SR659 会记录错误代码。

9. 此定位指令乃是运用 CANopen 协议的 SDO 通讯方式发送，并且每次只能发送一条 SDO 通讯，因此不建议运用于不同轴且须同动输出定位之应用。
10. 主机韧体版本 V1.12.10（含）以上，支持指令执行中变更目标地址和目标速度。变更目标速度时，伺服依加减速时间设定，进行加减速。若新的目标速度为 0，减速停止后，视为暂停；当新的目标速度由 0 变为非 0 时，会恢复动作。变更目标位置时，不允许输出反向的目标位置变更。相对定位，当新的目标位置为 0 时，立即减速停止，视为定位到达。

程序范例一（梯形图编程）：

1. 当 M0 = OFF → ON 时，指令开始对伺服站号 1~3 下达初始化设定，直到 SM1681 被设为 ON。
2. 当 M1 = OFF → ON 时，指令开始对伺服站号 2，启动伺服，SM1652 标志为 ON，表示 SERVO-ON。
3. 当 M2 = OFF → ON 时，伺服站号 2 以 100.0 r/min 移动到相对位置 100000 PUU，当输出到达定位之后完成标志 SM1632 为 ON。



程序范例二（ST 语言编程）：

```
0001 IF M2 THEN
0002    DDRVIC(2,100000,1000);
0003 END_IF;
0004 IF M2 AND SM1632 THEN
0005    SM1582 := TRUE;
0006    M2 := FALSE;
0007 END_IF;
```

1. 当 M2 = OFF → ON 时，伺服站号 2 以 100.0 r/min 移动到相对位置 100000 PUU，当输出到达定位之后完成标志 SM1632 为 ON。
2. 当 M2 与 SM1632 皆为 ON 时，设定伺服站号 2 自动复位标志，释放站号 2 的通讯控制权，让之后同样对站号 2 的通讯指令，可以继续执行。

台达专属驱动器使用之 CAN 通讯指令相关特殊标志（SM）与寄存器（SR）说明：

ASD-A2 CAN 通讯过程相关特殊标志与寄存器

功能项目	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
专用功能启动标志	R/W	由硬件配置（HWCONFIG）设定，Stop→Run 时启动							
伺服 INITC 与 CASD 通讯完成标志	R	SM1681							
通讯错误标志	R	SM1682							
单台初始化模式启动标志（V1.06.40 版以上支持）	R/W	SM1686=OFF（默认）为连续站号初始化模式 SM1686=ON 为指定单台初始化模式							
伺服定位完成标志#1	R/W	SM1631	SM1632	SM1633	SM1634	SM1635	SM1636	SM1637	SM1638
伺服暂停标志	R/W	SM1641	SM1642	SM1643	SM1644	SM1645	SM1646	SM1647	SM1648
伺服启动标志（Servo ON）	R	SM1651	SM1652	SM1653	SM1654	SM1655	SM1656	SM1657	SM1658
往返功能启动（DRVAC 支持）	R/W	SM1661	SM1662	SM1663	SM1664	SM1665	SM1666	SM1667	SM1668
往返方向标志（DRVAC 支持）	R	SM1671	SM1672	SM1673	SM1674	SM1675	SM1676	SM1677	SM1678
通讯控制权自动复位标志	R/W	SM1581	SM1582	SM1583	SM1584	SM1585	SM1586	SM1587	SM1588
Heartbeat 错误标志	R	SM1691	SM1692	SM1693	SM1694	SM1695	SM1696	SM1697	SM1698
Heartbeat 错误发生处理方式	R/W	SM1684=OFF（默认）为全部驱动器停止运转 SM1684=ON 为仅发生错误的驱动器停止运转							
通讯错误站编号	R	SR658							
通讯错误代码	R	SR659							

注#1：定位完成标志在各轴启动输出时，PLC 会自动清除，若用户需要对该轴连续执行定位动作，则请自行清除定位标志，以避免下一个连续输出定位指令还未开始输出，即判断到定位完成标志。

台达伺服 CAN 通讯各轴指定交握参数的特殊寄存器对应表

伺服参数名称（编号）	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
PR 命令（P5-07）	R	SR661	SR662	SR663	SR664	SR665	SR666	SR667	SR668
异警号码（P0-01） 请用 16 进制显示	R	SR671	SR672	SR673	SR674	SR675	SR676	SR677	SR678
DO 状态（P0-46）	R	SR681	SR682	SR683	SR684	SR685	SR686	SR687	SR688
伺服当前位置（P0-09）	R	SR691 SR692	SR693 SR694	SR695 SR696	SR697 SR698	SR699 SR700	SR701 SR702	SR703 SR704	SR705 SR706
命令目标位置（P0-10）	R	SR711 SR712	SR713 SR714	SR715 SR716	SR717 SR718	SR719 SR720	SR721 SR722	SR723 SR724	SR725 SR726
DI 状态（P4-07）	R	SR731	SR732	SR733	SR734	SR735	SR736	SR737	SR738
当前扭矩（P0-11） （单位 0.1%）	R	SR741	SR742	SR743	SR744	SR745	SR746	SR747	SR748
自定义数值(（P0-12)） 请搭配 SM1685 启动/关闭	R	SR791 SR792	SR793 SR794	SR795 SR796	SR797 SR798	SR799 SR800	SR801 SR802	SR803 SR804	SR805 SR806
定位完成允许范围	R/W	SR811	SR812	SR813	SR814	SR815	SR816	SR817	SR818

台达变频器 CANopen 通讯过程相关特殊标志与寄存器

功能项目	R/W	ID. 21	ID. 22	ID. 23	ID. 24	ID. 25	ID. 26	ID. 27	ID. 28
变频 INITC 与 CASD 通讯完成标志	R	SM1683							
变频启动中标志	R	SM1621	SM1622	SM1623	SM1624	SM1625	SM1626	SM1627	SM1628
变频 Heartbeat 错误标志	R	SM1611	SM1612	SM1613	SM1614	SM1615	SM1616	SM1617	SM1618
通讯控制权自动复位标志	R/W	SM1601	SM1602	SM1603	SM1604	SM1605	SM1606	SM1607	SM1608
变频状态显示 （Index 6041H-00H）	R	SR751	SR752	SR753	SR754	SR755	SR756	SR757	SR758
变频当前运转速度 （Index 6043H-00H）	R	SR761	SR762	SR763	SR764	SR765	SR766	SR767	SR768
变频当前转矩（单位 0.1%） （Index 2021H-0CH）	R	SR771	SR772	SR773	SR774	SR775	SR776	SR777	SR778
变频 DI 点状态 （Index 2022H-11H）	R	SR781	SR782	SR783	SR784	SR785	SR786	SR787	SR788

台达专用驱动器 CAN 通讯错误代码表（SR659 显示的错误数值）

错误代码	发生原因
0x0002	从站未响应 SDO 信息。

错误代码	发生原因
0x0003	从站接收信息错误。较容易发生在 COPRW 信息编程错误，使从站无法辨认。
0x0004	未接收到从站 PDO 信息。
0x0005	指令操作数使用错误。
0x0006	INITC 初始化时，有其中一站正被使用。
0x0007	从站站号超出 INITC 指令的指定范围。
0x0008	指定单台从站重置过程失败，需再确认通讯线与从站是否正常运行。

台达专属驱动器使用的 ECAT 通讯指令相关特殊标志（SM）与寄存器（SR）说明：

ECAT 通讯过程相关特殊标志与寄存器

功能项目	R/W	ID. 1 ~ ID. 16
CASD 通讯完成标志	R/W	SM1681
通讯错误标志	R/W	SM1682
断线错误发生处理方式	R/W	SM1684=OFF（默认）为全部驱动器停止运转 SM1684=ON 为仅发生错误的驱动器停止运转
ECAT SDO abort code（32-bit）	R/W	SR656 SR657
通讯错误站号	R/W	SR658
通讯错误代码	R/W	SR659

功能项目	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
伺服定位完成标志#2	R/W	SM1631	SM1632	SM1633	SM1634	SM1635	SM1636	SM1637	SM1638
伺服暂停标志	R/W	SM1641	SM1642	SM1643	SM1644	SM1645	SM1646	SM1647	SM1648
伺服/变频启动中标志	R	SM1651	SM1652	SM1653	SM1654	SM1655	SM1656	SM1657	SM1658
往返功能启动 (DRVAC 支持)	R/W	SM1661	SM1662	SM1663	SM1664	SM1665	SM1666	SM1667	SM1668
往返方向标志 (DRVAC 支持)	R	SM1671	SM1672	SM1673	SM1674	SM1675	SM1676	SM1677	SM1678
通讯控制权自动复位标志	R/W	SM1581	SM1582	SM1583	SM1584	SM1585	SM1586	SM1587	SM1588
断线标志#1	R/W	SM1691	SM1692	SM1693	SM1694	SM1695	SM1696	SM1697	SM1698

功能项目	R/W	ID. 9	ID. 10	ID. 11	ID. 12	ID. 13	ID. 14	ID. 15	ID. 16
伺服定位完成标志#2	R/W	SM1921	SM1922	SM1923	SM1924	SM1925	SM1926	SM1927	SM1928
伺服暂停标志	R/W	SM1931	SM1932	SM1933	SM1934	SM1935	SM1936	SM1937	SM1938
伺服/变频启动中标志	R	SM1621	SM1622	SM1623	SM1624	SM1625	SM1626	SM1627	SM1628
往返功能启动 (DRVAC 支持)	R/W	SM1941	SM1942	SM1943	SM1944	SM1945	SM1946	SM1947	SM1948
往返方向标志 (DRVAC 支持)	R	SM1951	SM1952	SM1953	SM1954	SM1955	SM1956	SM1957	SM1958
通讯控制权自动复位标志	R/W	SM1601	SM1602	SM1603	SM1604	SM1605	SM1606	SM1607	SM1608

功能项目	R/W	ID. 9	ID. 10	ID. 11	ID. 12	ID. 13	ID. 14	ID. 15	ID. 16
断线标志#1	R/W	SM1611	SM1612	SM1613	SM1614	SM1615	SM1616	SM1617	SM1618

注#1：因 ECAT 没有 Heartbeat 机制，故以通讯封包内的 working counter（WKC）作为判断从站掉线依据。

注#2：定位完成标志在各轴启动输出时，PLC 会自动清除，若用户需要对该轴连续执行定位动作，则请自行清除定位标志，以避免下一个连续输出定位指令还未开始输出，即判断到定位完成标志。

台达驱动器 ECAT 通讯各轴指定交握参数的特殊寄存器对应表

驱动器参数名称（编号）	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
伺服 PR 命令（P5-07）/ 变频状态显示（Index 6041H-00H）	R	SR661	SR662	SR663	SR664	SR665	SR666	SR667	SR668
伺服异警号码（P0-01）	R	SR671	SR672	SR673	SR674	SR675	SR676	SR677	SR678
伺服 DO 状态（P0-46）	R	SR681	SR682	SR683	SR684	SR685	SR686	SR687	SR688
伺服现在位置（P0-09）	R	SR691 SR692	SR693 SR694	SR695 SR696	SR697 SR698	SR699 SR700	SR701 SR702	SR703 SR704	SR705 SR706
伺服命令目标位置（P0-10）	R	SR711 SR712	SR713 SR714	SR715 SR716	SR717 SR718	SR719 SR720	SR721 SR722	SR723 SR724	SR725 SR726
伺服 DI 状态（P4-07）	R	SR731	SR732	SR733	SR734	SR735	SR736	SR737	SR738
伺服当前扭矩（P0-11）	R	SR741	SR742	SR743	SR744	SR745	SR746	SR747	SR748
伺服自定义数值（P0-12）	R	SR791 SR792	SR793 SR794	SR795 SR796	SR797 SR798	SR799 SR800	SR801 SR802	SR803 SR804	SR805 SR806
伺服定位完成允许范围	R/W	SR811	SR812	SR813	SR814	SR815	SR816	SR817	SR818

驱动器参数名称（编号）	R/W	ID. 9	ID. 10	ID. 11	ID. 12	ID. 13	ID. 14	ID. 15	ID. 16
伺服 PR 命令（P5-07）/ 变频状态显示（Index 6041H-00H）	R	SR751	SR752	SR753	SR754	SR755	SR756	SR757	SR758
伺服异警号码（P0-01）	R	SR761	SR762	SR763	SR764	SR765	SR766	SR767	SR768
伺服 DO 状态（P0-46）	R	SR1191	SR1192	SR1193	SR1194	SR1195	SR1196	SR1197	SR1198
伺服现在位置（P0-09）	R	SR1201 SR1202	SR1203 SR1204	SR1205 SR1206	SR1207 SR1208	SR1209 SR1210	SR1211 SR1212	SR1213 SR1214	SR1215 SR1216
伺服命令目标位置（P0-10）	R	SR1221 SR1222	SR1223 SR1224	SR1225 SR1226	SR1227 SR1228	SR1229 SR1230	SR1231 SR1232	SR1233 SR1234	SR1235 SR1236
伺服 DI 状态（P4-07）	R	SR781	SR782	SR783	SR784	SR785	SR786	SR787	SR788
伺服当前扭矩（P0-11）	R	SR771	SR772	SR773	SR774	SR775	SR776	SR777	SR778
伺服自定义数值（P0-12）	R	SR1241 SR1242	SR1243 SR1244	SR1245 SR1246	SR1247 SR1248	SR1249 SR1250	SR1251 SR1252	SR1253 SR1254	SR1255 SR1256
伺服定位完成允许范围	R/W	SR1261	SR1262	SR1263	SR1264	SR1265	SR1266	SR1267	SR1268

台达专用驱动器 ECAT 通讯错误代码表（SR659 显示的错误数值）

错误代码	发生原因
0x0002	从站未响应 SDO 信息
0x0003	从站接收信息错误。较容易发生在 COPRW 信息编辑错误，使从站无法辨认。
0x0004	未接收到从站 PDO 信息
0x0005	指令操作数使用错误
0x0006	INITC 初始化时，其中有一站正被使用。
0x0007	从站站号超出 INITC 指令指定范围
0x0008	指定单台从站重置过程失败，需再确认通讯线与从站是否正常运行。
0x0009	AS-FECAT 功能卡未响应通讯
0x000A	DCSOFC 指令同时控制超过四台从站

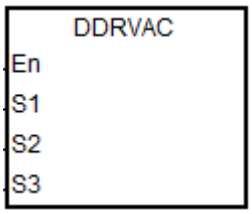
API	指令码			操作数	功能
2804	D	DRVAC		S_1 ， S_2 ， S_3	伺服绝对位置定位控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●					○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 伺服站号

S_2 ： 绝对型目标地址

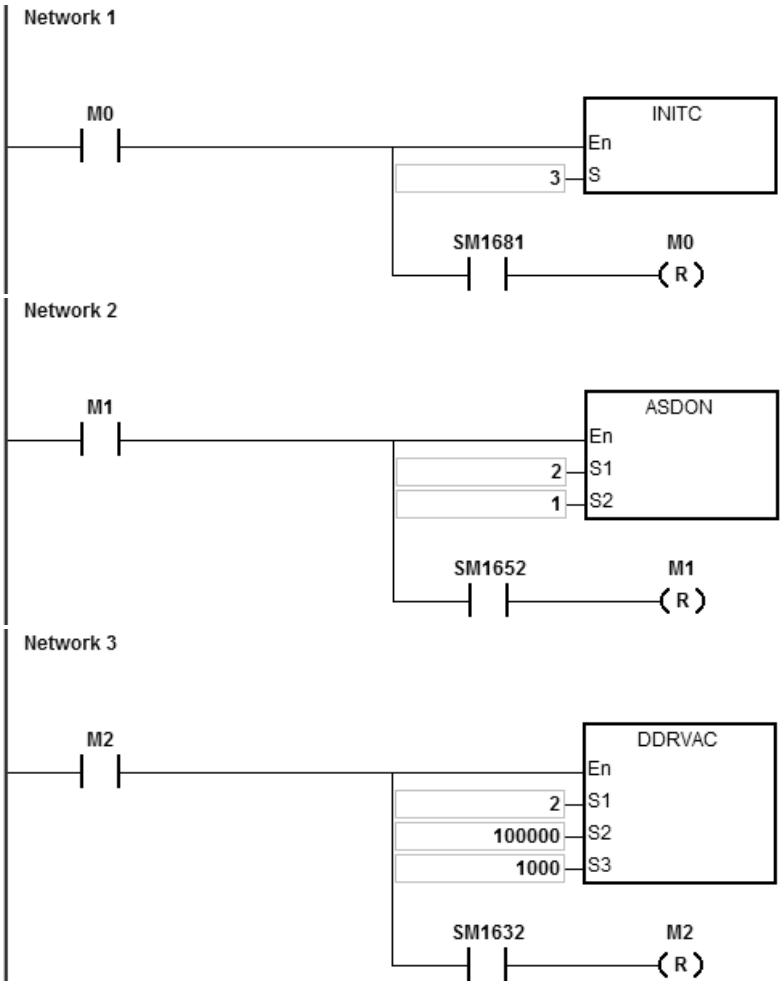
S_3 ： 目标速度

指令说明：

1. 主机韧体版本 V1.14.00(含)以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
2. 执行此指令前，使用 CAN 通讯，必须先完成指令 INITC 与 ASDON（SERVO ON）；使用 ECAT 通讯，必须先完成指令 INITEC 与 ASDON（SERVO ON）。
3. 使用 CAN 通讯，**S_1** 可指定范围为 1~8，超出此范围指令不执行；使用 ECAT 通讯，S_1 可指定范围为 1~16，超出此范围或是指定站号的装置种类不为伺服，指令不执行。
4. **S_2** 指定范围为 -2147483648 ~ 2147483647，正反号代表正反转方向。此目标地址为绝对型位置。
5. 其余内容与范例说明，请参考 DRVIC 指令说明。

程序范例 1：

1. 当 M0 = OFF → ON 时，指令开始对伺服站号 1~3 下达初始化设定，直到 SM1681 被设为 ON。
2. 当 M1 = OFF → ON 时，指令开始对伺服站号 2，启动伺服，SM1652 标志为 ON，表示 SERVO-ON。
3. 当 M2 = OFF → ON 时，伺服站号 2 以 100.0 r/min 从当前位置移动到绝对位置 100000 PUU，当输出到达定位之后完成标志 SM1632 为 ON。



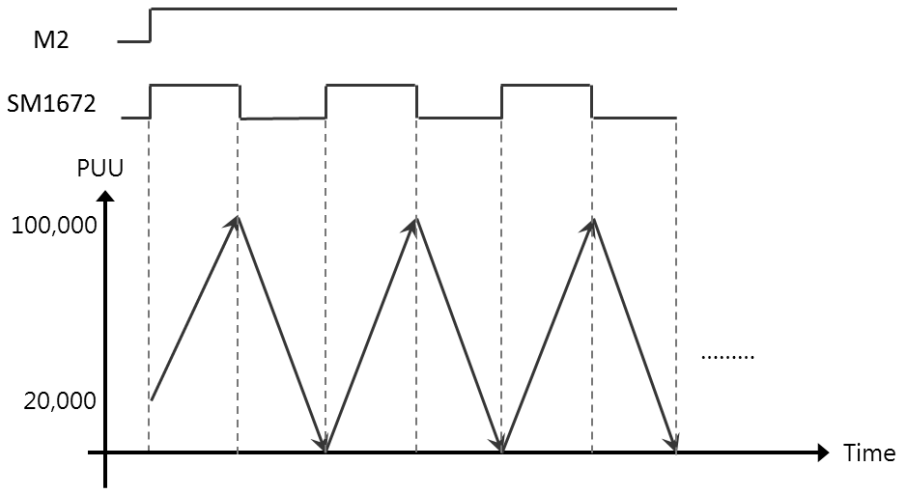
程序范例 2：

1. 延续程序范例 1，并且增加一行程序，在 PLC RUN 时，SET SM1662，启动伺服站号 2 的往返功能

2. 如下图，M2 启动后，伺服由当前位置=20,000 先移动到目标绝对位置 100,000，之后在绝对位置 100,000 与 0 之间做往返运动

   往返方向指示标志 SM1672=当伺服启动后往第一次目标位置移动时为 ON，之后会重复 ON->OFF 变化。

3. 正在运行过程中，目标位置可随时修改，但生效时机皆为下一次往返周期。


M2
SM1672
PUU
100,000
20,000
.....
Time

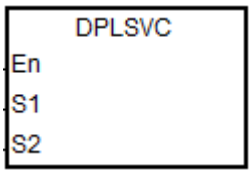
API	指令码			操作数	功能
2805	D	PLSVC		**S_1** ， **S_2**	驱动器速度控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●					○	○		
S_2								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



S_1 ： 驱动器站号

S_2 目标速度

指令说明：

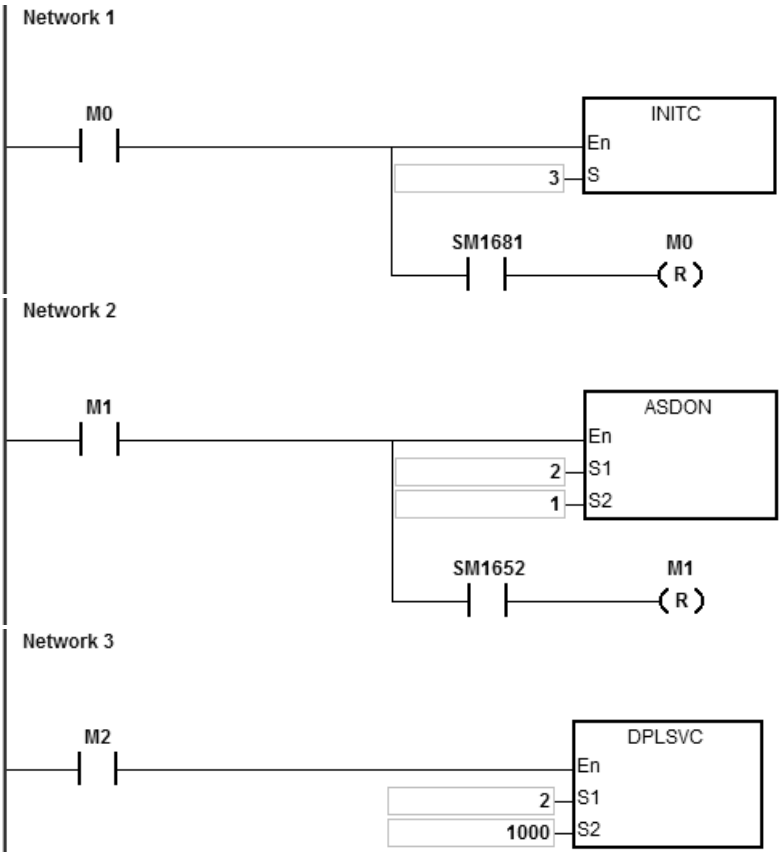
1. 主机韧体版本 V1.14.00(含)以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
2. 执行此指令前，使用 CAN 通讯，必须先完成指令 INITC 与 ASDON（SERVO ON）；使用 ECAT 通讯，必须先完成 INITEC 与 ASDON（SERVO ON）。
3. 使用 CAN 通讯，**S_1** 可指定范围为 1~8（伺服）或 21~28（变频），超出这些范围指令不执行；使用 ECAT 通讯，S_1 可指定范围为 1~16，超出这些范围指令不执行
4. **S_2** 说明整理如下表：

驱动器种类	**S_1** 站号范围	**S_2** 数值范围	**S_2** 说明
伺服	1~8（CAN） 1~16（ECAT）	-60000 ~ 60000 （0.1rpm）	正反号代表正反转方向，举例：输入 6005 等于 600.5rpm
变频器	21~28（CAN） 1~16（ECAT）	-32768 ~ 32767 （1rpm）	正反号代表正反转方向，举例：输入 605 等于 605rpm

5. 详细各轴对应的 SM 与 SR 表，请参考 DRVIC 指令说明。
6. 若执行通讯过程中发生错误，SM1682 将被设定为 ON ;另外，在 SR658 会记录发生错误的站编号及 SR659 会记录错误代码。

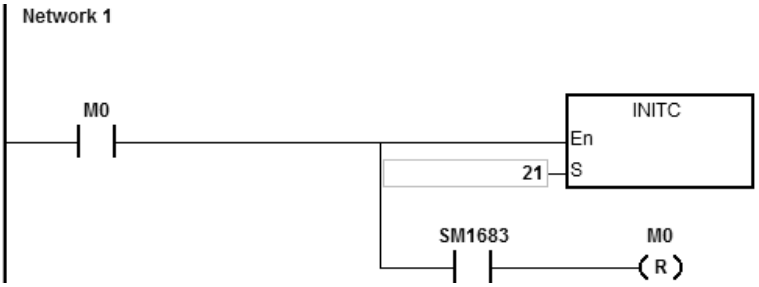
程序范例一（与台达伺服通讯）：

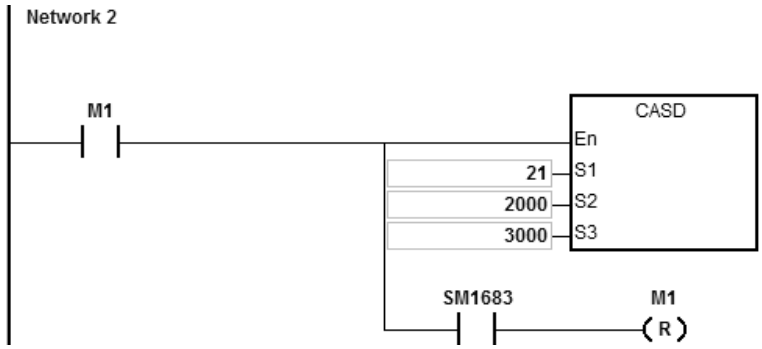
1. 当 M0 = OFF → ON 时，指令开始对伺服站号 1~3 下达初始化设定，直到 SM1681 被设为 ON。
2. 当 M1 = OFF → ON 时，指令开始对伺服站号 2，启动伺服，SM1652 标志为 ON，表示 SERVO-ON。
3. 当 M2 = OFF → ON 时，伺服站号 2 以 100.0 r/min 速度移动，直到 M2 为 OFF。



程序范例二：(与台达变频器通讯)

1. 当 M0 = OFF → ON 时，指令开始对变频站号 21 下达初始化设定，直到 SM1683 被设为 ON。
2. 当 M1 = OFF → ON 时，指令开始对变频站号 21，写入加减速时间。
3. 当 M2 = OFF → ON 时，对变频站号 21 启动速度控制模式。
4. 当 M3 = OFF → ON 时，变频器以 2500rpm 速度移动，直到 M3 为 OFF。




Network 2
M1
CASD
En
21
S1
2000
S2
3000
S3
SM1683
M1
R

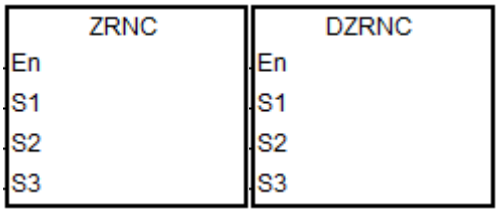
API	指令码			操作数	功能
2806	D	ZRNC		S_1 ， S_2 ， S_3	伺服原点回归

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●					○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●	●			●							
S_3		●	●			●							

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



S_1 ： 伺服站号

S_2 ： 第一段速度

S_3 ： 第二段速度

指令说明：

1. 主机韧体版本 V1.14.00(含)以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
2. 执行此指令前，使用 CAN 通讯，必须先完成指令 INITC 与 ASDON (SERVO ON)；使用 ECAT 通讯，必须先完成指令 INITEC 与 ASDON (SERVO ON)。并且建议可先运用 API 2810 ZRNM 指令设定一次伺服各轴的回原点模式。
3. 使用 CAN 通讯，S_1 可指定范围为 1~8 (伺服)，超出此范围指令不执行；使用 ECAT 通讯，S_1 可指定范围为 1~16，超出此范围或是指定站号的装置种类不为伺服，指令不执行。
4. S_2 指定范围为 10 ~ 20000，其速度单位为 0.1 rpm。(对应伺服参数 P5-05)。若是使用线性电机的伺服，并且第一段速超过 32767 以上时，请改用 DZRNC 指令发送命令。(DZRNC 需搭配 V1.04 版以上主机韧体版本)
5. S_3 指定范围为 10 ~ 5000，其速度单位为 0.1 rpm。(对应伺服参数 P5-06)
6. 当伺服回到原点后，其各轴对应的完成标志将会被设定为 ON，详细特殊标志 (SM) 与寄存器 (SR) 说明，请参考 DRVIC 指令。
7. 若执行通讯过程中发生错误，SM1682 将被设定为 ON ;另外，在 SR658 会记录发生错误的站编号及 SR659 会记录错误代码。

程序范例：

1. 当 M0 = OFF → ON 时，指令开始对伺服站号 1 下达初始化设定，直到 SM1681 被设为 ON。
2. 当 M1 = OFF → ON 时，指令开始对伺服站号 1，设定加速时间（10000）/减速时间（20000）。并且启动伺服，当 SM1651 标志为 ON，表示 SERVO-ON。
3. 当 M2 = OFF → ON 时，

   对站号 1 伺服的 P5-04 写入 2 字节的数据，数据内容在 D100，写入完成后 M100 被设置为 ON。

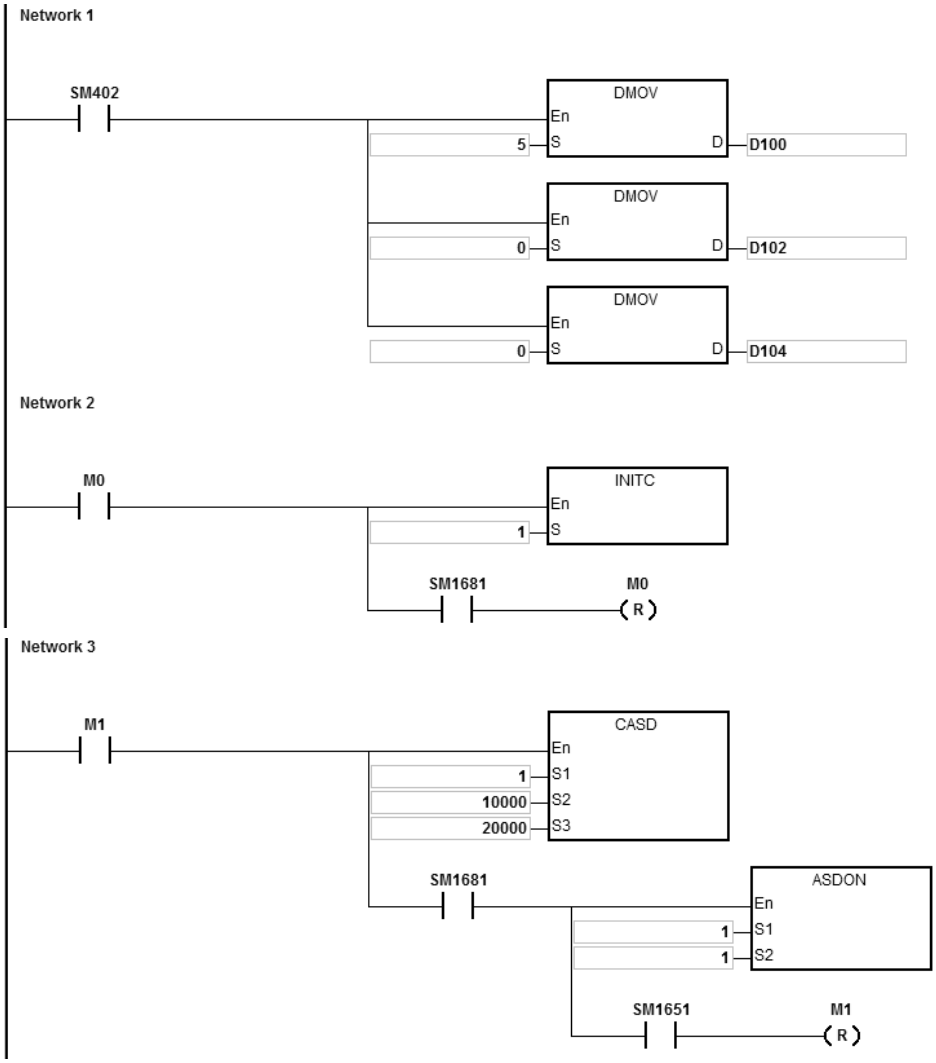
   对站号 1 伺服的 P6-00 写入 4 字节的数据，数据内容在 D102，写入完成后 M101 被设置为 ON。

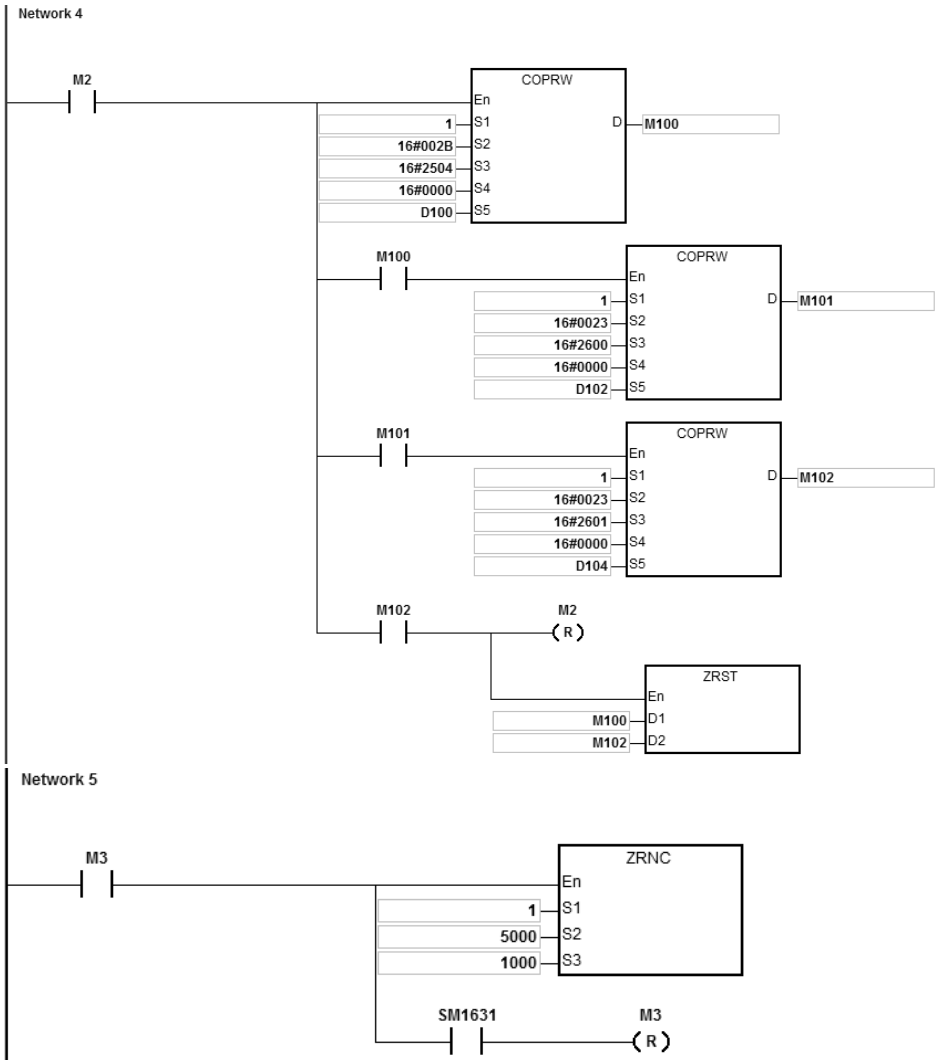
   对站号 1 伺服的 P6-01 写入 4 字节的数据，数据内容在 D104，写入完成后 M102 被设置为 ON。

   伺服的 P5-04 为原点回归模式设定

   伺服的 P6-00，P6-01 为原点回归定义

4. 当 M3 = OFF → ON 时，对站号 1 伺服启动原点回归功能。




Network 4
M2
COPRW
En
1
S1
D
M100
16#002B
S2
16#2504
S3
16#0000
S4
D100
S5
M100
COPRW
En
1
S1
D
M101
16#0023
S2
16#2600
S3
16#0000
S4
D102
S5
M101
COPRW
En
1
S1
D
M102
16#0023
S2
16#2601
S3
16#0000
S4
D104
S5
M102
M2
(R)
ZRST
En
M100
D1
M102
D2
Network 5
M3
ZRNC
En
1
S1
5000
S2
1000
S3
SM1631
M3
(R)

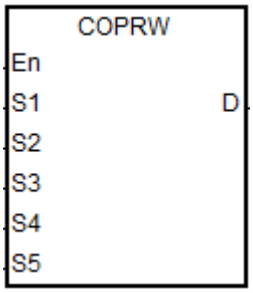
API	指令码	操作数	功能
2807	COPRW	S_1 · S_2 · S_3 · S_4 · S_5 · D	CANopen 通讯读写数据

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●					○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●								
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
S_5		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



S_1 ： 伺服或从站的站号

S_2 ： 请求码

S_3 ： 索引

S_4 ： 子索引

S_5 ： 读取或写入的装置

D ： 通讯完成标志

指令说明：

1. S_1 可指定范围为 1~127，超过此范围时，指令将自动以最小(<1 时)或最大(>127 时)值发送。
2. S_2 只能指定下表的请求码，若设定代码错误时，则此指令不执行，并设定 SM1682 将被设定为 ON，SR659 为 16#0005。

S_2 代码(下 8 位数值有效)	功能说明
16#00H23	写 4 个字节（Byte）数据，加速型(（Expedited)）SDO
16#00H2B	写 2 个字节数据，加速型(（Expedited)）SDO
16#00H2F	写 1 个字节数据，加速型(（Expedited)）SDO

S_2 代码(下 8 位数值有效)	功能说明
16#00H40	读取数据，数据长度包含在 SDO 响应的信息中，加速型(（Expedited)）SDO
16#0021	由 S_5 指定写入长度与数据，标准型(（Normal)）SDO
16#0041	读取数据，并将读取长度与数据储存于 S_5，标准型(（Normal)）SDO

3. S_3 与 S_4 请直接参考台达伺服/变频器手册，或者第三方符合 CANopen DS301 设备的对象字典设定说明。
4. 使用台达专属通讯模式时，建议先执行初始化通讯 INITC 指令后，再使用此 COPRW 指令，避免写入参数有可能会被初始化指令覆盖。COPRW 指令为 CANopen 的 SDO 通讯方式，若从站的站号超出初始化规范的 1~8 或 21~28 站号范围时，用户可自行使用 COPRW 指令，进行从站相关参数的设定或读取。

   在台达专属驱动模式时，其 COPRW 指令（SDO 通讯）内定通讯逾时如下表所示的时间：

通讯速率（bps）	1M	500K	250K	125K	50K
内定通讯逾时（ms）	100	100	200	400	500

   在 CANopen DS301 模式时，其 COPRW 指令通讯逾时时间，可自行从 HWCONFIG 的 CANopen 通讯逾时参数变更，其默认逾时时间为 100ms。
5. 若执行通讯过程中发生错误，SM1682 将被设定为 ON，并且 SR658 会记录发生错误的站编号及 SR659 会记录错误代码（可参考 DRVIC 指令说明）。备注：使用通讯指令时，用户必须自行编写通讯错误的处理流程，以避免不可预期的通讯错误发生时，造成通讯流程不正确。
6. COPRW 时序图如下图所示：

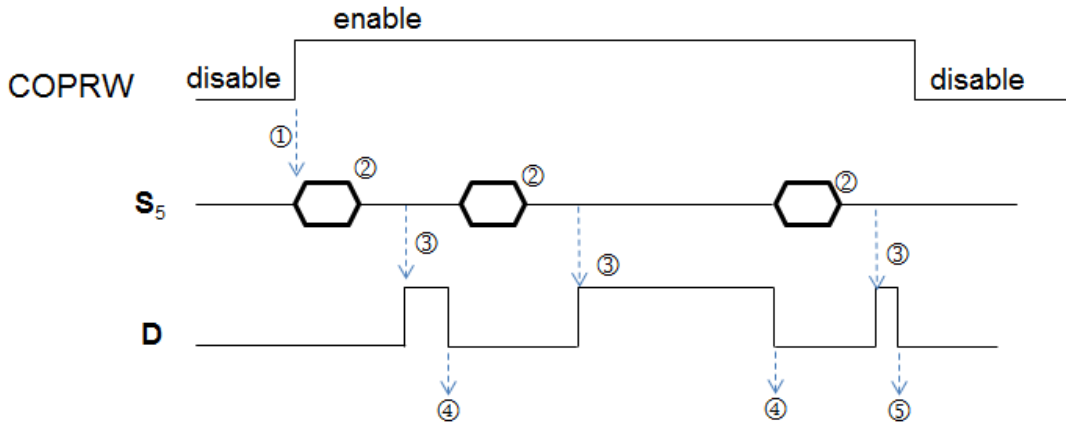
   ① 指令初次启动，若未有其它 CANopen 通讯占用，立即发送命令码

   ② 正在发送命令码

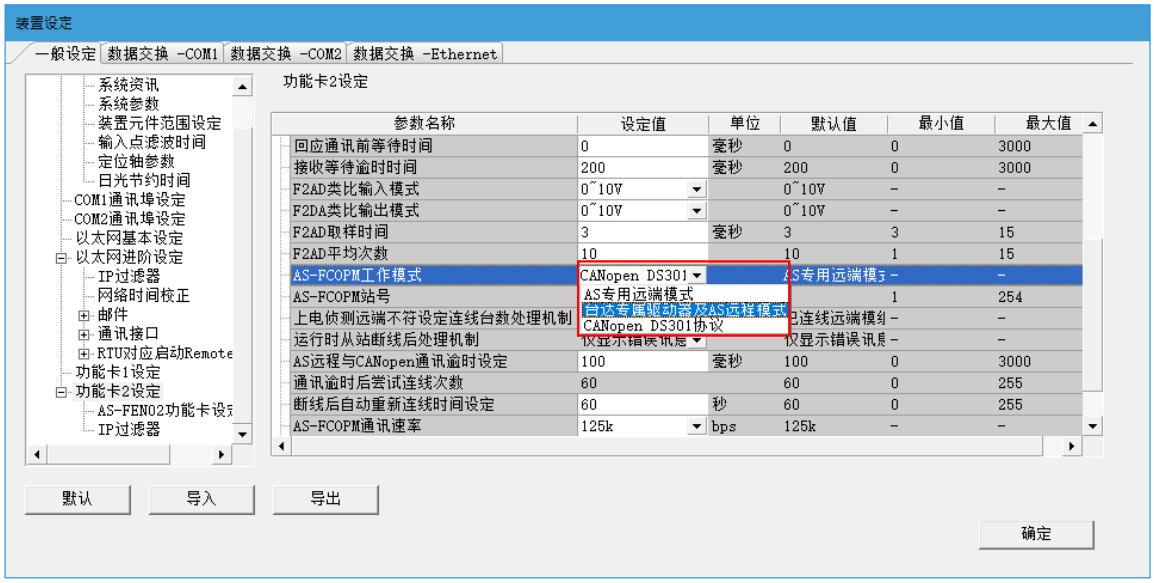
   ③ 发送完成，并设定完成标志

   ④ 用户变更下一笔发送数据，清除完成标志，即刻发出下一笔命令码

   ⑤ 完成发送动作，关闭 COPRW 指令



7. COPRW 指令目前仅提供硬件配置设定为 “台达专属驱动器及 AS 远程模式”的模式” 或者 ”CANopen DS301 协议”的模式，此模式设定如下图



8. 台达 ASDA-A2 的参数大多都是以十进制数值表示，用户可将下列范例做参数转换为索引地址，至于子索引地址请固定填入 0。举例 :PX-YY 的索引地址 = 0x2000 + (X << 8) + YY

   范例：P2-10 = 0x2000 + (0x0002 << 8) + 0x000A = 0x220A

   P5-04 = 0x2000 + (0x0005 << 8) + 0x0004 = 0x2504

   P1-44 = 0x2000 + (0x0001 << 8) + 0x002C = 0x212C

9. 台达变频器的参数也是以十进制数值表示，但参数转换为索引与子索引需用下列公式转换 ：

   PXX-YY 的索引地址 = 0x2000 + XX (16 进制)，子索引地址 = YY+1 (16 进制)

   范例：P10-15 ➔ 索引为 0x2000 + 0x000A = 0x200A，子索引为 0x0F+1= 0x10

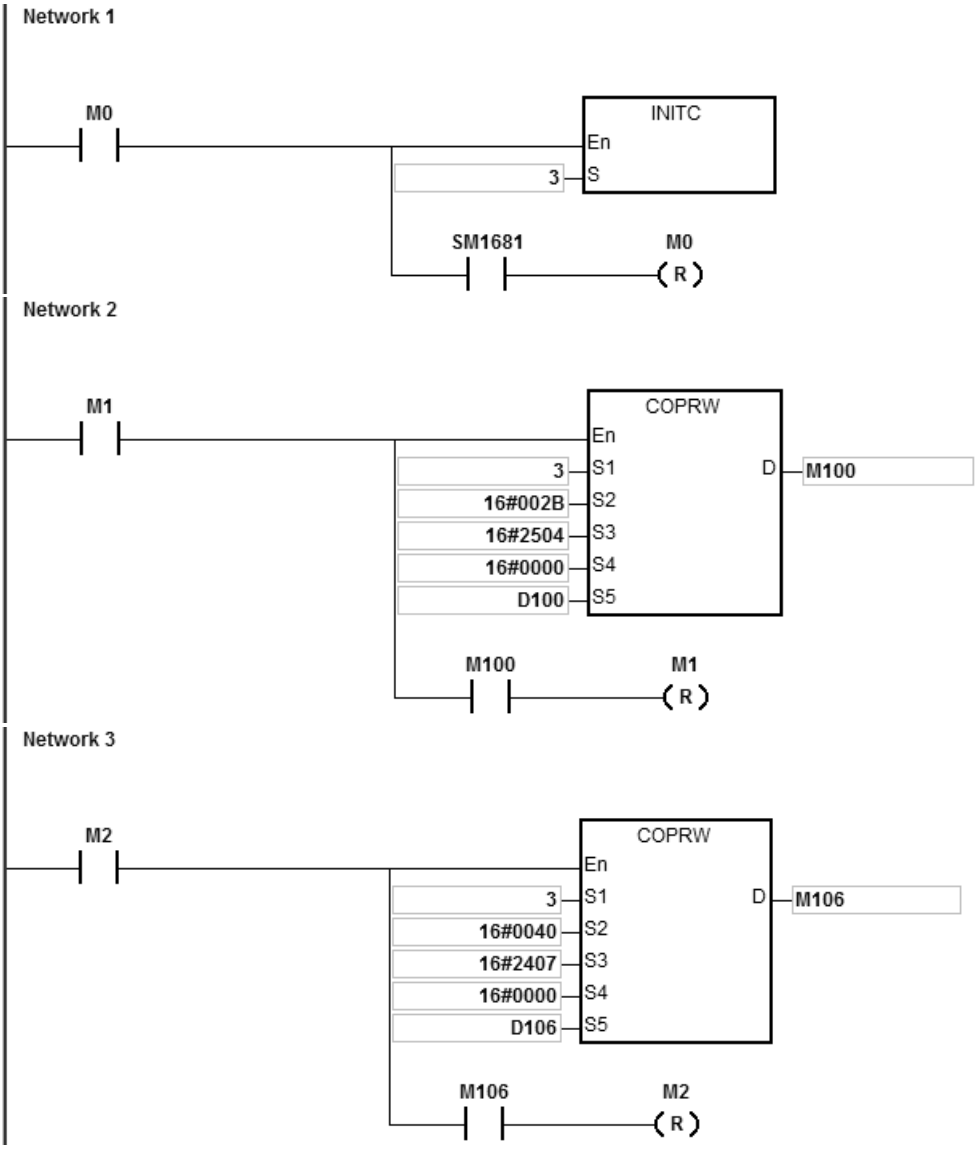
10. 由于此 COPRW 通讯指令乃是运用 SDO 方式进行通讯，(即必须一去一回的通讯方式)，并且是与本章节介绍的台达专属通讯定位指令共享 SDO 通讯控制权，因此请用户编写 PLC 程序时，务必注意以下事项：
    - 不建议在同一个扫描周期里，同时启动发送两个以上 COPRW 指令或通讯定位指令。
    - 不建议将 COPRW 指令采用一直启动通讯的方式。
    - 不建议写于功能块内，若必须写于功能块内时，则 PLC 程序有 2 个 POU (程序单元) 调用该功能块，将会有同时启用两个指令的状况出现。
11. 标准型 (Normal) SDO 的通讯请求码 (16#0021 & 16#0041)，必须使用主机韧体版本为 V1.04.80 (含) 以上版本。
12. 当通讯请求码设定为 16#0021 时，其 **S_5** 表示为写给从站的 Byte 长度值 n，以及 **S_5**+1~ **S_5**+ ((n+1) / 2) 为写入的资料，举例：**S_5** 指定为 D100，并且内容值设定为 16，则此指令将会把 D101~D108 的内容值，以 D101 下 8 位、D101 上 8 位、D102 下 8 位、D102 上 8 位…的顺序传送给从站。
13. 当通讯请求码设定为 16#0041 时，当指令接收到从站响应的信息时，将会把 Byte 长度值 n 存放于 **S_5**，并且将接收数据按照接收顺序储存于 **S_5**+1~ **S_5**+ ((n+1) / 2)，举例：指令接收到 15 个 Byte 长度，则 D100 储存 15，并且将数据串 Byte[0]，Byte[1].. Byte[14] 分别储存于 D101 下 8 位、D101 上 8 位…、D107 上 8

位以及最后 D108 下 8 位。

14. 为避免 Normal SDO 传送或接收过多数据，而造成通讯数据缓冲区不足，因此 PLC 将自动限制最大 Byte 长度值为 255 个，若发生超出范围时，就直接以最大长度 255 个传送或接收。

程序范例一：(采用 Expedited SDO 方式)

1. 当 M0 = OFF → ON 时，指令开始对伺服站号 1~3 下达初始化设定，直到 SM1681 被设为 ON。
2. 当 M1 = OFF → ON 时，对站号 3 伺服的 P5-04 (S_3=16#2504) 写入 2 字节的数据 (S_2=16#002B)，数据内容在 D100，写入完成后 M100 被设置为 ON。
3. 当 M2 = OFF → ON 时，对站号 3 伺服的 P4-07 (S_3=16#2407) 进行读取 (S_2=16#0040)，读取到的内容会摆放在 D106，读取完成后 M106 被设置为 ON。



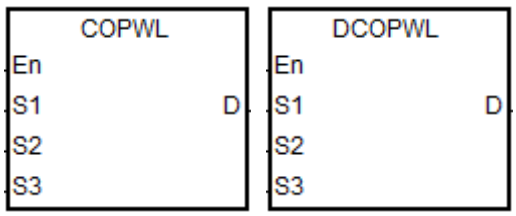
API	指令码			操作数	功能
2808	D	COPWL		**S_1 · S_2 · S_3 · D**	CANopen 多笔参数写入

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
S_1								●					○	○		
S_2								●	●							
S_3								●					○	○		
D		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●	●			●	●						
S_3		●				●							
D	●												

脉冲执行型	16 位指令	32 位指令
-	AS	AS

符号：



S_1 ： 站号

S_2 ： 写入数据源装置的起始位置

S_3 ： 连续写入的笔数

D ： 通讯完成标志

指令说明：

1. **S_1** 站号可指定范围为 1~127，超过此范围时，指令将自动以最小（<1 时）或最大（>127 时）值发送。
2. **S_2** 与 **S_3** 分别为写入数据源装置的起始位置与写入笔数，举例：**S_2** 指定起始装置为 D10，以及写入笔数为 3 笔，则各个数据列表如下表说明：

指令名称	笔数编号	索引地址	子索引地址	写入来源数值
COPWL（16 位数值写入）	1	D10	D11	D12
	2	D13	D14	D15
	3	D16	D17	D18
DCOPWL（32 位数值写入）	1	D10	D11	D12 · D13
	2	D14	D15	D16 · D17
	3	D18	D19	D20 · D21

S_3 输入范围为 1~100 笔

3. 台达伺服与变频器的索引地址与子索引地址，请参考 COPRW 指令说明。原则上台达伺服与变频器的参数写入皆为 16 位或 32 位数值（包含浮点数），若用户有需要 8 位数值写入时，请采用 COPRW 指令。

4. **D** 为通讯完成设定的标志，当多笔通讯传送完成后，**D** 将会被设为 ON，详细的发送流程与时序图如下：

   ① 用户启动 COPWL 指令开始传送数据。

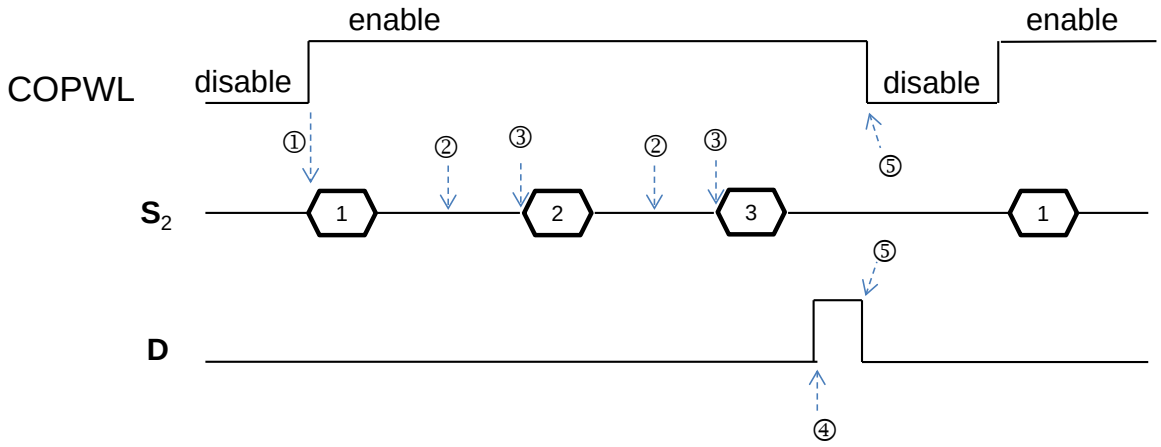
   ② COPWL 指令传送一笔数据之后，继续进行下一个 PLC 指令。

   ③ 等待下一次扫描至 COPWL，并且确定从站已经收到时，发送下一笔数据。

   ④ 当数据传送完成最后一笔指定的写入参数之后，指令将设定指定的完成标志。

   ⑤ 用户程序收到完成标志已设定时，需自行关闭 COPWL 指令，以利后续 COPWL 或 COPRW 指令运行。

   备注：当用户关闭指令时，完成标志也会自动被清除一次。



   备注：此时序图以传送 3 笔写入参数为例。

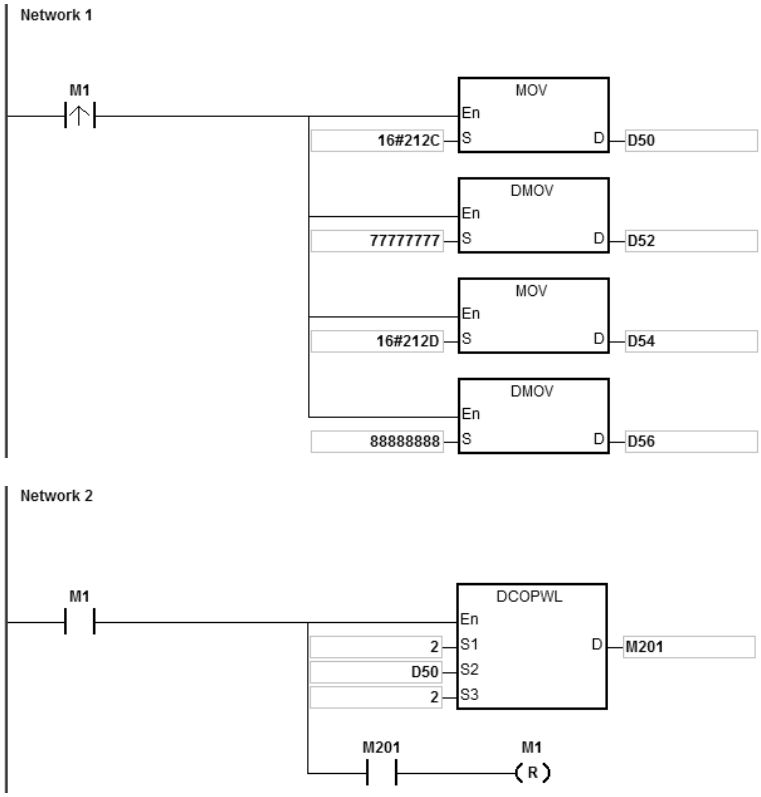
5. 指令启动之后，请等待写入完成被设定之后，再关闭指令；若执行过程有通讯错误发生，请排除故障原因之后，重新再启动指令全部写入一次。

程序范例：

1. 当 M1 = OFF → ON 时，将数据填入 D 装置。

指令名称	笔数编号	索引地址	子索引地址	写入来源数值
DCOPWL (32 位数值写入)	1	D50 = 16#212C (电子齿轮比分子)	D51 = 0	D52，D53 = 77777777
	2	D54 = 16#212D (电子齿轮比分母)	D55 = 0	D56，D57 = 88888888

2. 当 M1 = OFF → ON 时，对站号 2 伺服的 P1-44 写入 32 字节的数据，数据内容在 D52=77777777；P1-45 写入 32 字节的数据，数据内容在 D56=88888888，写入完成后 M201 被设置为 ON。



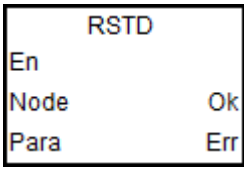
API	指令码	操作数	功能
2809	RSTD	**Node**，**Para**，**Ok**，**Err**	发送重置或 NMT 命令

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
Node								●					○	○		
Para								●					○	○		
Ok		●	●	●												
Err		●	●	●												

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●											
Para		●											
Ok	●												
Err	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Node ： 重置的站号

Para ： 参数设定值

Ok ： 重置完成标志

Err ： 重置错误标志

指令说明：

1. 使用 CAN 通讯，用于台达专属指令模式时，在执行重置指令前，请先确认所有台达驱动器已经使用 INITC 指令初始化过一遍，并且是曾经正常运行过。
2. 使用 CAN 通讯，用于 CANopen DS301 模式时，此指令将做为 NMT 通讯功能，可利用 **Para** 参数进行网络状态切换。
3. 使用 ECAT 通讯，在执行重置指令前，请先确认所有台达驱动器已使用 INITEC 指令初始化过一遍，并且是曾经正常运行过。
4. 此指令适用于主机韧体版本 V1.04.10（含）以上，ISPSoft 编辑软件需 V3.03.08 版以上。主机韧体版本 V1.14.00（含）以上，ISPSoft 编辑软件需为 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
5. 使用 CAN 通讯，当 CAN 通讯口被指定运行于台达专属驱动器模式时，**Node** 可指定范围为 0（广播）、1~8（此范围为伺服专用站号区），以及 21~28 之间（此范围为变频器专用站号区），当站号超出此范围时，则 PLC 将不执行重置动作，并且会设定 **Err** 错误标志。（错误代码请参考 SR659 说明）
6. 使用 CAN 通讯，当 CAN 通讯口被指定运行于 CANopen DS301 模式时，**Node** 可指定范围为 1~64 站号，以及站号 0 为广播功能，当数值超出此范围时，则 PLC 将不执行 NMT 通讯动作，并且会设定 **Err** 错误标志。（错误代码请参考 SR659 说明）

7. 使用 ECAT 通讯，Node 可指定范围为 0（广播）、1~16（此范围为伺服/变频专用站号区），当站号超出此范围时，则 PLC 将不执行重置动作，并且会设定 Err 错误标志。（错误代码请参考 SR659 说明）
8. **Para** 参数设定值，仅适用于 CANopen DS301 模式，若在台达专属驱动器模式与 AS 远程模式时，此参数无意义。此 **Para**（NMT 服务码）可设定如下表所示，若输入数值非表内指定值，则会设定 **Err** 错误标志。

NMT 服务码	16#01	16#02	16#80	16#81	16#82
功能说明	从站启动	从站停止	进入预运行	应用重置	通讯重置

9. 此指令每次只能对一台驱动器或从站进行命令动作，若同时启动多个 RSTD 指令时，则 PLC 自动以先启动的指令先执行。
10. 此指令的执行时机为每次启动时送出命令，若在未收到 **Ok** 完成标志之前即关闭指令，则 PLC 将不设定 **Ok** 完成标志。
11. 此指令执行的内容，除了通知指定的驱动器清除错误状态之外，也会连带重新确认相关通讯参数是否正确，并重新建立正确的通讯参数。举例：原先正常运行中，因从站站号 2 那一台故障断线，造成整个系统停止运行，此时现场人员排除故障之后，用户只需针对站号 2 重新下此指令重置，即可让此从站恢复原先可被控制的状态，省下全部驱动器皆须重新初始化的时间。
12. 通讯过程中，若从站响应给主机任何一个通讯命令错误，此指令都会设定 **Err** 标志，并停止后续动作。（错误代码请参考 SR659 说明）

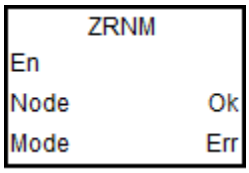
API	指令码	操作数	功能
2810	ZRNM	**Node**，**Mode**，**Ok**，**Err**	台达伺服驱动器回原点模式设定

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Node								●					○	○		
Mode								●					○	○		
Ok		●	●	●												
Err		●	●	●												

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●											
Mode		●											
Ok	●												
Err	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Node ： 指定的站号

Mode ： 回原点模式代码

Ok ： 完成标志

Err ： 错误标志

指令说明：

1. 执行重置指令前，请先确认所有台达驱动器已经初始化完成。
2. 此指令适用于主机韧体版本 V1.04.10（含）以上，ISPSoft 编辑软件需 V3.03.08 版以上。主机韧体版本 V1.14.00（含）以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
3. 使用 CAN 通讯，Node 可指定范围为 1~8（此范围为伺服专用站号区）；使用 ECAT 通讯，Node 可指定范围为 1~16，当数值超出此范围时或是指定站号的装置种类不为伺服，则 PLC 将不执行回原点模式设定动作，并且会设定 **Err** 错误标志。（错误代码请参考 SR659 说明）
4. 此指令每次只能对一台驱动器进行模式设定，若同时启动多个指令时，则 PLC 自动以先启动的指令先执行。
5. 此指令的执行时机为每次启动时送出命令，若在未收到完成标志之前即关闭指令或指令不再被扫描，则 PLC 将无法设定 **Ok** 完成或 Err 错误标志。
6. 此指令只会在条件接点 En 从 OFF 变为 ON 时，触发一次通讯发送，因此强烈建议采用梯形图方式编辑此指令。当采用 SFC 或 ST 语言编辑时，若无法让指令的条件接点 En 由 ON 变 OFF，则此指令将只能被触发一次，之后将无法再被发送。
7. **Mode** 回原点模式代码选择如下表所示，若设定数值超出此范围，则 PLC 还是会传送，然后由服务器本身决定是否接收命令。此设定模式即为 ASDA 伺服参数 P5-04。

- 台达伺服回原点模式设定值为 16 进制数值编码，其数值字段定义为 0xWZYX，各代码说明如下表：

代码名称	数值范围	功能与代码说明	备注
W	0 ~ 1	选择其最终停止位置 0 = 离开原点减速停止后，再自动回到真实原点 1 = 离开原点减速停止后，就不再动作	
Z	0 ~ 1	碰到极限时处理机制 0 = 停止输出 1 = 反向输出	
Y	0 ~ 2	Z 相信号处理模式（适用 X 代码 0~8） 0 = 返回时寻找 Z 相，往前不寻找 1 = 返回时不找 Z 相，往前时寻找 Z 相 2 = 一律不寻找 Z 相 Z 相信号处理模式（适用 X 代码 9~A） 0 = 返回时寻找 Z 相 1 = 一律不寻找 Z 相	
X	0 ~ A	回归模式选择 0~8 0 = 正转方向原点回归，PL 为原点 1 = 反转方向原点回归，NL 为原点 2 = 正转方向原点回归，ORG 由 OFF 变 ON 为原点 3 = 反转方向原点回归，ORG 由 OFF 变 ON 为原点 4 = 正转方向原点回归，寻找 Z 相脉冲为原点 5 = 反转方向原点回归，寻找 Z 相脉冲为原点 6 = 正转方向原点回归，ORG 由 ON 变 OFF 为原点 7 = 反转方向原点回归，ORG 由 ON 变 OFF 为原点 8 = 直接以目前位置为原点 回归模式选择 9~A 9 = 正转方向原点回归，碰撞为原点 A = 反转方向原点回归，碰撞为原点	

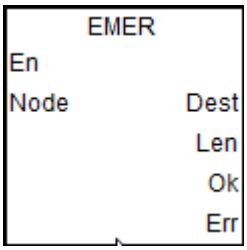
API	指令码	操作数	功能
2811	EMER	**Node**，**Dest**，**Len**，**Ok**，**Err**	读取 Emergency 信息

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Node								●					○	○		
Dest								●								
Len								●								
Ok		●	●	●												
Err		●	●	●												

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●											
Dest		●											
Len		●											
Ok	●												
Err	●												

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Node ： 指定的站号

Dest ： 存放目的装置

Len ： 已读回的笔数

Ok ： 完成标志

Err ： 错误标志

指令说明：

1. 此指令适用于 CANOpen DS301 通讯模式，台达专属模式和 ECAT 通讯。
2. 此指令适用于主机韧体版本 V1.04.10（含）以上，ISPSoft 编辑软件需 V3.03.08 版以上。主机韧体版本 V1.14.00（含）以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
2. 当收到从站 **Node** 响应 Emergency 信息之后，将会自动存放于 **Dest** 指定的存放位置，并且会设定 **Ok** 完成标志。
3. **Node** 可指定范围建议为已存在的从站号码，当数值超出此站号范围，或者从站已断线时，则 PLC 将无法收到任何信息，并且将会设定 **Err** 错误标志与通讯超时错误代码。(错误代码请参考 SR659 说明)
4. 此指令读取动作与 AS 操作手册内 Emergency 通讯方式一样，因此操作时，请挑选其中一种通讯方式即可，不可同时操作。

5. 此指令最多可读取 5 笔 Emergency 信息，每次读取 **Ok** 成功之后，**Len** 会存放读回的总笔数。用户可依照此长度，判断 **Dest** 已经连续占用几个 word，每一笔信息占用 4 个 word，存放顺序为下 8 位、上 8 位，存放格式如下表格 : (举例：**Dest** 为 D10，**Len** 为 D5 得到长度 2 笔)

D 装置编号	数值
D5	2

D 装置编号	上 8 位	下 8 位
D10	第 1 笔第 2 个 byte	第 1 笔第 1 个 byte
D11	第 1 笔第 4 个 byte	第 1 笔第 3 个 byte
D12	第 1 笔第 6 个 byte	第 1 笔第 5 个 byte
D13	第 1 笔第 8 个 byte	第 1 笔第 7 个 byte
D14	第 2 笔第 2 个 byte	第 2 笔第 1 个 byte
D15	第 2 笔第 4 个 byte	第 2 笔第 3 个 byte
D16	第 2 笔第 6 个 byte	第 2 笔第 5 个 byte
D17	第 2 笔第 8 个 byte	第 2 笔第 7 个 byte

API	指令码			操作数	功能
2812	D	CSFOC		**Xno ~ OutSpd**	通讯型控制伺服追随功能

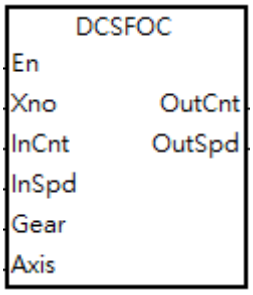
装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Xno	○															
InCnt								●								
InSpd								●								
Gear								●								●
Axis								●					●	●		
OutCnt								●								
OutSpd								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Xno	●												
InCnt			●				●						
InSpd			●				●						
Gear									●				
Axis		●											
OutCnt			●				●						
OutSpd			●				●						

脉冲执行型	16 位指令	32 位指令
-	-	AS

6

符号：



Xno ： 输入点来源

InCnt ： 输入脉冲个数显示

InSpd ： 输入速度显示

Gear ： 输入与输出速度比率值设定（浮点数）

Axis ： CANopen 通讯控制面板达伺服的站号

OutCnt ： 输出位置显示（单位 PUU）

OutSpd ： 输出速度显示

指令说明：

1. 主机韧体版本 V1.14.00（含）以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
2. 此指令使用 CAN 通讯适用于主机韧体版本 V1.06.00（含）以上，以及 ISPSoft 软件 V3.07（含）版以上。。
3. 执行此指令前，使用 CAN 通讯，必须先完成指令 INITC 与 ASDON（SERVO ON）；使用 ECAT 通讯，必须先完成指令 INITEC 与 ASDON（SERVO ON）。

4. **Xno** 输入点来源仅可选择 X0.0、X0.2、X0.4、X0.6、X0.8 及 X0.10 这 6 点，且会占用连续 2 点输入，选择非指定输入点时皆无法执行。当选择好输入点，其高速计数器也将自动被指定，若用户有写相同高速计数器之 DCNT 指令，或者 CSFO 指令（API 2708），则 PLC 将以先启动计数器之指令为优先执行。这 6 个输入点所占用输入点与对应之高速计数器如下表所示：

输入组别	1	2	3	4	5	6
Xno+0 输入点（A 相）	X0.0	X0.2	X0.4	X0.6	X0.8	X0.10
Xno+1 输入点（B 相）	X0.1	X0.3	X0.5	X0.7	X0.9	X0.11
高速计数器编号	HC202	HC206	HC210	HC214	HC218	HC222
输入反向标志	SM270	SM271	SM272	SM273	SM274	SM275

5. 当此指令之高速计数器只能使用于 A/B 相输入模式，当连接手摇轮之后，在没有转动状况下，PLC 的输入点灯号为 ON 时，请设定输入反向标志，通知 PLC 检测有效工作周期改为 OFF 状态。
6. **InCnt** 为显示输入脉冲个数，请采用 32 位变量宣告。
7. **InSpd** 为显示输入脉冲频率，请采用 32 位变量宣告，其基本单位为 1Hz。
8. **Gear** 为输入与输出比率值（浮点数）。其比率转换公式为 “输出脉冲频率” = “输入脉冲频率” * **Gear**。举例：**Gear** 设为 2.5，则当输入速度为 50 时，其输出速度将为 50 x 2.5 = 125。注：输出个数与输出速度都以整数值（去小数方式）输出。
9. **Axis** 输出轴的站号选择，使用 CAN 通讯，此站号范围为 1~8，并且是 INITC 指令初始化范围内的站号；使用 ECAT 通讯，此站号范围为 1~16，并且是 INITEC 指令初始化范围内的站号，指定站号的装置种类须为伺服。
10. **OutCnt** 为显示输出脉冲个数，请采用 32 位变量宣告，其基本单位为伺服的 PUU。
11. **OutSpd** 为显示输出速度，请采用 32 位变量宣告，其基本单位为 Hz。
12. 此指令无使用次数限制，也可多个指令采用同一组输入点当输入计数来源，但每次指令初次启动时，皆会先自动清除输入计数器内容值，因此采用同一个输入带动多个输出时，请注意此清除动作。另外此指令是依照输出站号，记录相关输入以及输出轴相关信息，因此不可同时启动相同输出轴，若有重复使用时，PLC 将以先启动先输出为输出原则。使用 ECAT 通讯，最多同时控制 4 台从站，超过 4 台限制，指令不执行，SM0 = ON，错误代码 SR0 = 16#2003。
13. 此指令不建议使用于 ST 语言编辑、中断程序以及仅被执行一次性的功能块编辑。

注意事项：

1. 输入脉冲频率是以输入脉冲之正半周脉冲宽度（ON）为计算依据，若脉冲 ON 与 OFF 宽度不为 1：1 时，默认还是以 ON 宽度为准。
2. 输入脉冲 ON 即是指输入点灯亮，若搭配手摇轮使用时，请注意没有转动时，输入点灯号状态是否为 OFF，是否需要启动输入反向标志。
3. 此指令会有大约 20ms 以及 CANopen/ECAT 通讯时间的落后反应，若无法满足需求，请考虑改用 CSFO 指令，改用脉冲型输出控制伺服。

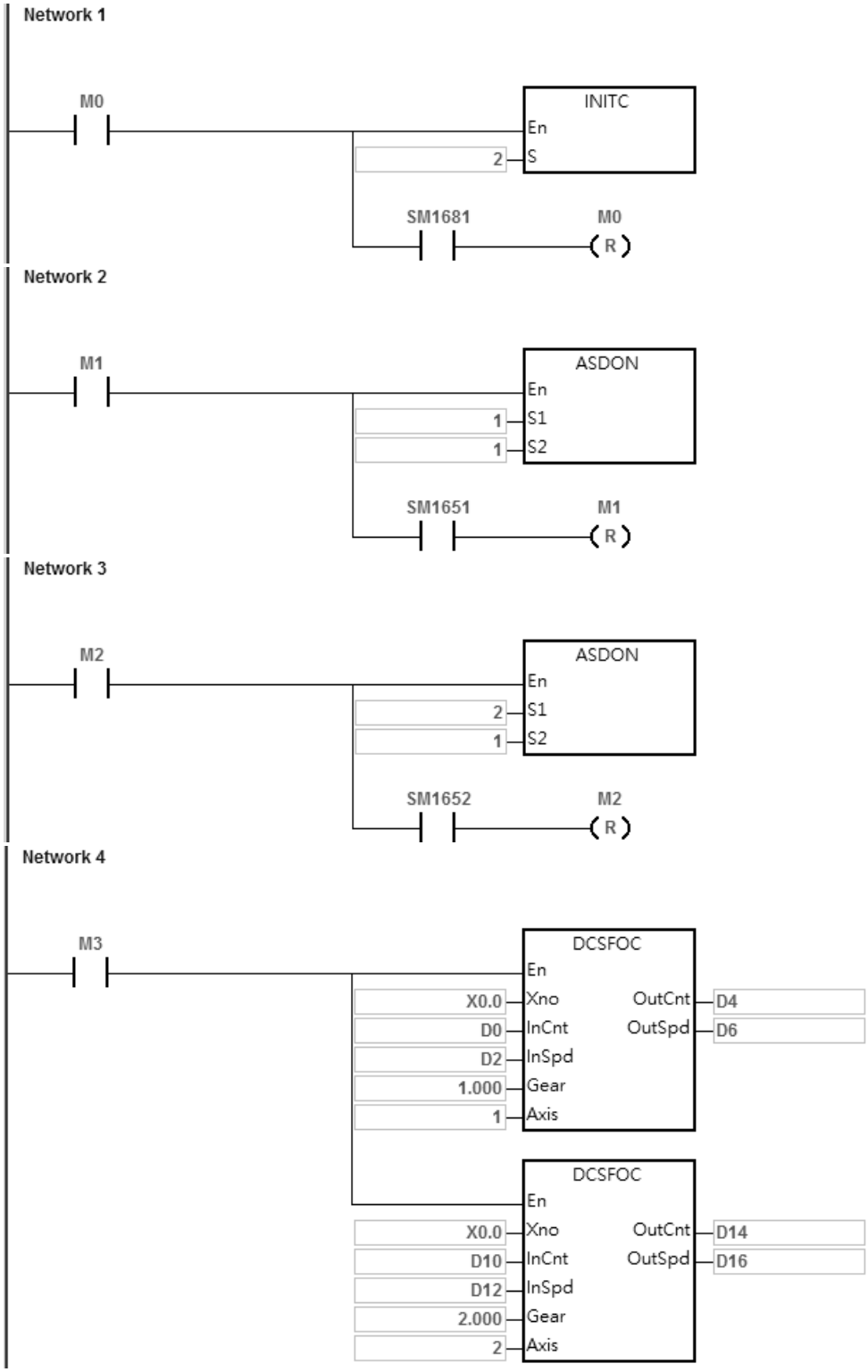
程序范例：

当 M0 = OFF → ON 时，指令开始对伺服站号 1~2 下达初始化设定，直到 SM1681 被设为 ON。

当 M1 = OFF → ON 时，指令开始对伺服站号 1，启动伺服，SM1651 标志为 ON，表示 SERVO-ON。

当 M2 = OFF → ON 时，指令开始对伺服站号 1，启动伺服，SM1652 标志为 ON，表示 SERVO-ON。

当 M3 = OFF → ON 时，由 X0.0 与 X0.1 连接手摇轮控制 2 台伺服，比率值 1 台为 1，另一台为 2。



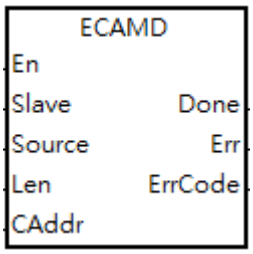
API	指令码	操作数	功能
2813	ECAMD	**Slave ~ ErrCode**	E-CAM 表下载至从站

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Slave								●					○	○		
Source								●	●				○	○		
Len								●					○	○		
CAddr								●					○	○		
Done		●	●	●												
Err		●	●	●												
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
Source		●				●							
Len		●				●							
CAddr		●				●							
Done	●												
Err	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Slave ：从轴伺服站号

Source ：电子凸轮表编号或装置元件起始位置

Len ：下载表格之段数长度

CAddr ：下载至伺服电子凸轮表格之起始地址

Done ：下载完成标志

Err ：下载错误标志

ErrCode ：错误代码

指令说明：

1. 此指令需搭配 PLC 主机韧体 V1.06（含）版以上，以及 ISPSoft 软件 V3.07（含）版以上。
2. **Slave** 为下载目的伺服之通讯站号，其站号范围为 1~8，若超出此范围时，指令不会执行，并且设定错误标志 **Err** 为 ON。当下载完成之后，**Done** 下载完成标志将会被设定为 ON。
3. 当 **Source** 为直接指定数值，指定下载电子凸轮表时的表格编号，其数值范围 1~16，若超出范围时，PLC 会设定错误标志 **Err** 为 ON。当用户指定从 PLC 内部寄存器编号，则会从指定装置编号开始连续读取 **Len+1** 个 32 位数值，举例：**Source** 为 D10，**Len** 为 10，则将会从 D10/D11，D12/D13，….D30/D31 读取 11 个 32 位数值，并且传送至指定伺服内。

4. **Len** 为下载电子凸轮表之段数长度，此长度的范围为 1~720，超出范围时设定错误标志 **Err** 为 ON。当下载 C**Addr** 起始地址加上 **Len+1** 长度，超出伺服最大的 800 时，PLC 指令会自动停止下载，并且设定错误标志 **Err** 为 ON。
5. **CAddr** 为下载至伺服内电子凸轮表的起始地址，此地址的范围为 0~798，若超出此范围时，指令不会执行，并且设定错误标志 **Err** 为 ON。
6. 当内定的 16 张表格不够使用时，用户可在 CAM 表编辑画面中，选择下载至 PLC 档案寄存器（FR）内，增加储存容量。
7. 建议建立电子凸轮表时，若表格内使用点数不多时，则可善用伺服内部 800 个点的空间，以节省 PLC 下载至伺服时所花的通讯时间。举例：用户规划凸轮点编号 1~200 为表 A，以及 201~400 为表 B，则可一次一起下载这 400 点至伺服内。
8. 当指令启动时，其动作为清除 **Done** 标志，然后传送 E-CAM 表内容，若传送与接收过程一切正常，则会设定 **Done** 标志。若需重新传送时，请至少关闭指令一个扫描周期之后，再重新启动指令。若是传送过程中，从轴有任何错误回应，则会设定错误标志 **Err**。
9. 当 **Err** 为 ON 时，请参考下列 **ErrCode** 数值，以利错误诊断。

ErrCode 数值	状态说明	备注
0000	无错误发生	
0001	从轴站号超出范围	
0002	ECAM 表编号超出范围	
0003	下载地址（**CAddr**）或长度（**Len**）超出范围	
0004	从轴通讯无响应	
0005	下载或设定过程中，从站回应错误	
0006	主机不在驱动器专属模式	

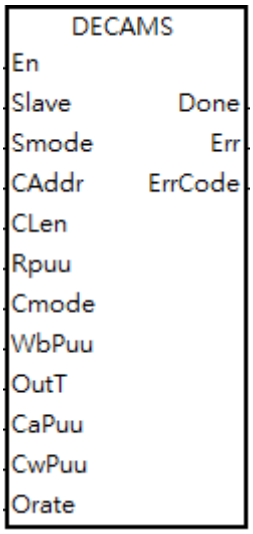
API	指令码			操作数	功能
2814	D	ECAMS		**Slave ~ ErrCode**	E-CAM 参数设定与规划

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Slave								●					○	○		
Smode								●					○	○		
CAddr								●					○	○		
CLen								●					○	○		
Rpuu								●					○	○		
Cmode								●					○	○		
WbPuu								●					○	○		
OutT								●					○	○		
CaPuu								●					○	○		
CwPuu								●					○	○		
Orate								●					○	○		
Done		●	●	●												
Err		●	●	●												
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
Smode		●				●							
CAddr		●				●							
CLen		●				●							
RPuu			●				●						
Cmode		●				●							
WbPuu			●				●						
OutT		●				●							
CaPuu			●				●						
CwPuu			●				●						
ORate			●				●						
Done	●												
Err	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Slave ： 指定从轴的站号

Smode ： 主轴编码器来源选择

CAddr ： E-CAM 表格之参考起始地址

CLen ： E-CAM 表格之使用区段长度

RPuu ： 旋转一圈脉冲个数（32 位数值）

Cmode ： 啮合模式设定

WbPuu ： 第一次啮合前等待输入个数（32 位数值）

OutT ： 脱离啮合时机设定

CaPuu ： 啮合后输入多少个数后脱离（32 位数值）

CwPuu ： 脱离后等待多少个数后再啮合（32 位数值）

ORate ： 凸轮表输出比例设定（32 位数值）

Done ： 下载完成标志

Err ： 下载错误标志

ErrCode ： 错误代码

指令说明：

1. 此指令需搭配 PLC 主机韧体 V1.06（含）版以上，以及 ISPSoft 软件 V3.07（含）版以上。
2. **Slave** 为指定伺服之通讯站号，其站号范围为 1~8，若超出此范围时，指令不会执行，并且设定 **Err** 下载错误标志为 ON。当下载完成，且从站伺服也响应正常之后，**Done** 下载完成标志将会设为 ON。
3. **Smode** 为伺服追随主轴之编码器讯号来源选择，可选择之项目如下表所示。

主轴来源设定值	功能说明
0	CAP 轴
1	AUX ENC
2	Pulse Cmd（A/B 相输入）
3	PR 命令
4	时间轴（1ms）
5	同步抓取修正轴（P5-77）
6	模拟通道 1（虚拟轴，单位每 10V 对应频率 1M pulse/s）
12	Pulse Cmd（CW/CCW 输入）
22	Pulse Cmd（Pulse/Dir 输入）
其它数值	以主轴来源编号 2 方式处理

4. 主机会依主轴来源设定值，修改伺服 P5-88.Y 命令来源。当主轴来源选择 Pulse Cmd，会再修改伺服 P1-00 的脉冲型式。伺服 P1-00 须在伺服 servo off 时才可以设定。

5. **CAddr** 为指定伺服内电子凸轮表的起始地址，此地址的范围为 0~794，若设定超出此范围时，指令不会执行，并且设定 **Err** 下载错误标志。主机会设定伺服 P5-81。
6. **CLen** 为指定执行之电子凸轮表区段长度，此长度的范围为 5~720 段，N 段会需要 N+1 点数据，当设定"起始地址 + 长度"超出最大 800 时，指令不会执行，并且设定 **Err** 下载错误标志。举例：起始地址设为 501，长度设定为 400，此时 PLC 将不会执行。主机会设定伺服 P5-82。伺服 P5-82 须在伺服 ECAM 停止（P5-88.X0=0） 时才可以设定。
7. **RPuu** 为设定主轴的电子凸轮表旋转一圈 360 度之脉冲个数（伺服的单位名称 Puu），此为 32 位数值。数值范围 10 ~ 1073741823，超过上下限，以上下限处理。主机会设定伺服 P5-83，P5-84。伺服 P5-83，P5-84 须在伺服 ECAM 停止（P5-88.X0=0） 时才可以设定。
8. **Cmode** 为啮合模式选择，其可设定之模式如下表所示。

模式设定值	功能说明
0	立即
1	DI: CAM ON
2	Capture 任一点动作
其它数值	以模式 0 方式啮合

9. 主机会设定伺服 P5-88.Z 啮合时机。
10. **WbPuu** 为第一次啮合前等待输入脉冲个数（32 位数值），当啮合启动之后，ASDA 伺服将等待此输入个数到达之后，才开始进行啮合动作。数值范围-1073741824 ~ +1073741823，超过上下限，以上下限处理。当此数值=0 时，即表示不开启啮合前等待功能。此参数也可自行参考伺服 P5-87 参数说明。主机会设定伺服 P5-87。
11. **OutT** 为啮合脱离模式选择，其可设定之模式如下表所示。相加表示复选，但 2、4、6 不可同时选。

模式 设定值	功能说明	脱离后动作
0	不脱离	-
1	DI: CAM OFF	进入停止状态
2	Master 轴达到设定位移量 CaPuu（P5-89）	进入停止状态
4	Master 轴超出设定位移量 CaPuu（P5-89）	回到前置状态，前置量为 CwPuu（P5-92）
6	同 2，但脱离时维持速度连续，啮合长度会超过 CaPuu（P5-89）一点点，适合脱离后立即连接 PR 命令之用	进入停止状态
8	搭配模式 1，2，或 6 脱离后关闭凸轮	设定 P5-88.X0 = 0
	搭配模式 4 避免回到前置状态的命令震动	无
其它数值	以模式 0 方式啮合	

12. 主机会设定伺服 P5-88.U 脱离时机。

13. **CaPuu** 为指定啮合后输入脉冲个数（32 位数值）到达后，执行脱离啮合状态，数值范围-1073741824 ~ +1073741823，超过上下限，以上下限处理。当此数值=0 时，即表示立即脱离。此参数也可自行参考伺服 P5-89 参数说明。主机会设定伺服 P5-89。
14. **CwPuu** 为指定脱离后输入脉冲个数（32 位数值）到达后，重新执行啮合状态，当此数值=0 时，即表示立即进入啮合状态。此参数也可自行参考伺服 P5-92 参数说明。**主机会设定伺服 P5-92。**
15. **ORate** 为凸轮表输出比例值设定，此数值为 32 位整数值，数值范围-2147000000 ~ +2147000000，超过上下限，以上下限处理。请参考伺服 P5-19 说明，举例：当比例值设为 123000 时，则伺服内部会再除以 1000000，因此相当于比例值为 0.123；当比例值为为 1500000 时，则伺服内部会再除以 1000000，因此相当于比例值为 1.5。主机会设定伺服 P5-19。
16. 当指令启动时，其动作为清除 **Done** 标志，然后发送设定 E-CAM 相关参数命令，若传送与接收过程一切正常，则会设定 **Done** 标志。若需再次设定时，请关闭指令一个扫描周期之后，再重新启动指令。若是设定过程中，从轴有任何错误回应，则会设定错误标志 **Err**。
17. 此指令适合为初次 E-CAM 功能的参数设定，或者首次啮合前设定 E-CAM 表功能。若从轴正在啮合执行中，想要变换 E-CAM 表格内的内容，则请采用 ECAMC 指令变更。
18. 当 **Err** 为 ON 时，请参考下列 **ErrCode** 数值，以利错误诊断。

ErrCode 数值	状态说明	备注
0000	无错误发生	
0001	从轴站号超出范围	
0003	下载地址（**CAddr**）或长度（**CLen**）超出范围	
0004	从轴通讯无响应	
0005	下载或设定过程中，从站回应错误	
0006	主机不在驱动器专属模式	

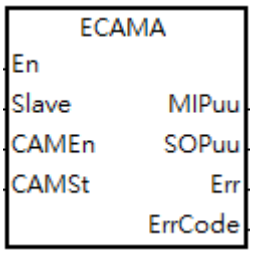
API	指令码	操作数	功能
2815	ECAMA	**Slave ~ ErrCode**	E-CAM 启动与状态显示

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Slave								●					○	○		
CAMEn	●	●	●	●												
CAMSt								●								
MIPuu								●								
SOPuu								●								
Err		●	●	●												
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
CAMEn	●												
CAMSt		●				●							
MIPuu			●				●						
SOPuu			●				●						
Err	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Slave ： 从轴伺服站号

CAMEn ： E-CAM 启动/关闭设定

CAMSt E-CAM 状态显示

MIPuu ： 显示主轴的现在输入脉冲数（单位 PUU）

SOPuu ： 显示该轴的现在输出脉冲数（单位 PUU）

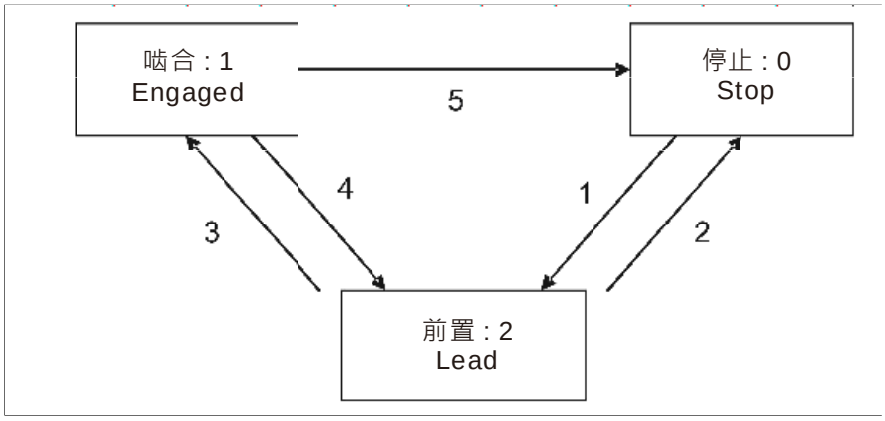
Err ： 错误标志

ErrCode ： 错误代码

指令说明：

1. 执行此指令，必须先完成 INITC 指令。
2. 此指令需搭配 PLC 主机韧体 V1.06（含）版以上，以及 ISPSoft 软件 V3.07（含）版以上。
3. **Slave** 为指定接收此命令之从轴伺服驱动器的站号，此站号范围为 1~8 之间数值。当站号超出此范围数值，指令不会进行传送动作，并且设定 **Err** 错误标志 ON；若指定站号之伺服无法正常通讯，也会设定 **Err** 通讯错误为 ON。
4. **CAMEn** 为 E-CAM 启动设定开关。当 **CAMEn** 的 BOOL 值为 ON，，表示启动 E-CAM 控制；当 **CAMEn** 的 BOOL 值为 OFF，，表示关闭 E-CAM 控制。指令执行中，主机会依 ECAMEn 状态去设定伺服 P5-88.X0。

5. **CAMSt** 为目前伺服 E-CAM 功能的状态，依伺服 P5-88.S 显示 0 代表停止状态，1 代表啮合状态，2 代表前置状态。



6. 状态说明：
   - 停止：为凸轮的初始状态，凸轮不会随主动轴脉冲运动。当凸轮功能关闭时（P5-88.X0 = 0）一律回到本状态。
   - 前置：当啮合条件（路径 1）成立后，进入本状态，凸轮依然不随主动轴脉冲而运动。
   - 啮合：当前置量到达（路径 3）后，进入本状态，凸轮开始随主动轴脉冲而运动。
7. 路径说明：
   - 路径 1：啮合条（P5-88.Z）成立时，状态可由停止->前置。前置量由 P5-87 指定。
   - 路径 2：当凸轮功能关闭（P5-88.X0 = 0），则回到停止状态。
   - 路径 3：当前置量到达后，状态可由前置->啮合。
   - 路径 4：脱离条件（P5-88.U = 4）成立，状态可由啮->前置。前置量由 P5-92 指定。
   - 路径 5：脱离条件（P5-88.U = 1，2，6）成立，或是凸轮功关闭（P5-88.X0 = 0），状态可由啮合->停止。
8. **MIPuu** 为显示指定 **Slave** 从站接收到主轴所输入之脉冲个数，单位为伺服的 PUU；当此指令被关闭时，此数值将不会再刷新。此数值是依伺服 P5-86 显示。
9. **SOPuu** 为显示指定 **Slave** 从站已经输出的脉冲个数，单位为伺服的 PUU；当此指令被关闭时，此数值将不会再刷新。此数值是依伺服 CMD_O 显示。
10. 当 **Err** 为 ON 时，请参考下列 **ErrCode** 数值，以利错误诊断。

ErrCode 数值	状态说明	备注
0000	无错误发生	
0001	从轴站号超出范围	
0004	从轴通讯无响应	
0005	下载或设定过程中，从站回应错误	
0006	主机不在驱动器专属模式	
0007	从轴未完成初始化	

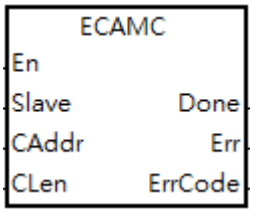
API	指令码	操作数	功能
2816	ECAMC	**Slave ~ ErrCode**	变更 E-CAM 表起始位置与长度

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Slave								●					○	○		
CAddr								●					○	○		
CLen								●					○	○		
Done		●	●	●												
Err		●	●	●												
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
CAddr		●				●							
CLen		●				●							
Done	●												
Err	●												
ErrCode		●				●							

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Slave ： 从轴伺服站号

CAddr ： E-CAM 表格之参考起始地址

CLen ： E-CAM 表格之使用区段长度

Done ： 参数变更完成标志

Err ： 参数变更错误标志

ErrCode ： 错误代码

指令说明：

1. 此指令需搭配 PLC 主机韧体 V1.06（含）版以上，以及 ISPSoft 软件 V3.07（含）版以上。
2. **Slave** 为指定接收此命令之从轴伺服驱动器的站号，此站号范围为 1~8 之间数值。当站号超出此范围数值，指令不会进行传送动作，并且设定 **Err** 错误标志 ON；若指定站号之伺服无法正常通讯，也会设定 **Err** 通讯错误为 ON。当变更 E-CAM 表参数，且收到指定从站响应正确接收之后，**Done** 完成标志将会被设为 ON。
3. **CAddr** 为指定伺服内电子凸轮表的起始地址，此地址的范围为 0~794，若设定超出此范围时，指令不会执行，并且设定 **Err** 下载错误标志。主机会设定伺服 P5-81。
4. **CLen** 为指定执行之电子凸轮表区段长度，此长度的范围为 5~720 段，N 段会需要 N+1 点数据，当设定 “起始地址 + 长度” 超出最大 800 时，指令将不会执行，并且设定 **Err** 下载错误标志。举例：起始地址设为 501，长度设定为 400，此时 PLC 不会下达命令。主机会设定伺服 P5-82。伺服 P5-82 须在伺服 ECAM 停止 （P5-88.X0=0） 时才可以写。

5. 当指令启动时，其动作为清除 **Done** 标志，然后发送变更命令，之后指令将等待从轴响应，直到收到从轴响应之后，才会设定 **Done** 标志。若需再次变更 E-CAM 起始地址与长度时，请关闭指令一个扫描周期之后，再重新启动指令。
6. 此指令适合于快速变更 E-CAM 的应用，但用户须先下载有效之 E-CAM 内容于指定的伺服内，才可开始应用。
7. 当 **Err** 为 ON 时，请参考下列 **ErrCode** 数值，以便错误诊断。

ErrCode 数值	状态说明	备注
0000	无错误发生	
0001	从轴站号超出范围	
0003	下载地址（**CAddr**）或长度（**CLen**）超出范围	
0004	从轴通讯无响应	
0005	下载或设定过程中，从站回应错误	
0006	主机不在驱动器专属模式	

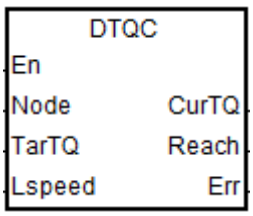
API	指令码			操作数	功能
2817	D	TQC		**Node、TarTQ、…、Err**	扭矩命令控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Node								●					○	○		
TarTQ								●					○	○		
LSpeed								●					○	○		
CurTQ								●								
Reach		●	●	●				●								
Err		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●				●							
TarTQ		●				●							
LSpeed			●				●						
CurTQ		●				●							
Reach	●												
Err	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Node ：驱动器站号

TarTQ ：目标扭矩命令

LSpeed ：限制最大速度命令

CurTQ ：当前扭矩数值

Reach ：目标扭矩已到达标志

Err ：通讯错误标志

指令说明：

1. 执行此指令前，请先确认所有台达驱动器已经初始化完成。
2. 此指令适用于 AS 主机韧体版本 V1.06.20（含）以上，以及 ISPSoft 编辑软件 V3.07 版（含）以上。主机韧体版本 V1.14.00（含）以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
3. 伺服要设定 P1-01=3 为扭矩控制模式（T），P1-01 设定须重新上电才会生效。变频要设定为扭矩模式，可用 ASDON 指令设定，再设定 P11-33（转矩命令来源）为 3（CANOpen）。
4. 使用 CAN 通讯，**Node** 为指定台达伺服与变频驱动器之站号，可使用的站号范围为 1~8（伺服驱动器）与 21~28（变频驱动器）之间；使用 ECAT 通讯，Node 可使用的站号范围为 1~16，指定从站的装置种类须为伺服，若设定超出此范围，则 PLC 将回应 **Err** 错误标志为 ON。

5. **TarTQ** 为指定之扭矩命令，建议用户事先参考驱动器操作手册说明之后，再进行数值设定。此扭矩命令允许执行中进行变更。

驱动器种类	**Node** 站号范围	**TarTQ** 说明
伺服	1~8（CAN） 1~16（ECAT）	主机设定伺服 P1-12（内部扭矩指令 1），数值范围为-300~+300，单位为 1 %，超出上下限范围，以上下限数值处理，V1.08.40 版（含）以上支持伺服 ASDA-A3/B3 高精度扭矩模式，数值范围为-3000~+3000，单位为 0.1 %，超出上下限范围，以上下限数值处理；正负号代表正反转，伺服依 P1-01（扭矩输出方向控制）动作；并且须设定伺服 DI：TCM0=ON，TCM1=OFF 以选择内部扭矩指令 1。由于台达伺服 P1-12 初值非 0%，若采用扭矩模式时，不希望发生伺服 Servo On 时，就开始进行运转，请记得 Servo on 之前，自行先清除伺服 P1-12 的数值。（可利用 COPRW 指令写入）
变频器	21~28（CAN）	主机设定变频 P11-27（最大转矩命令）、index 6071-00（目标转矩）和 index 6072-00（最大输出转矩），数值范围为-500~+500，单位为 1%，超过上下限范围，以上下限数值处理。

6. **LSpeed** 为限制最大速度命令，此命令为避免驱动器在无法达到指定扭矩命令之状况时，可限制驱动器的速度不再往上增加。此限制速度命令允许执行中进行变更。

驱动器种类	**Node** 站号范围	**LSpeed** 说明
伺服	1~8（CAN） 1~16（ECAT）	主机设定伺服 P1-55（最大速度限制），数值范围 10~伺服电机之最大可运转速度，单位为 1 r/min。
变频器	21~28（CAN）	主机设定变频 1. P01-00=LSpeed（最高操作频率）、数值范围0~59900，单位为0.01Hz，超过上下限范围，以上下限数值处理。 2. P11-37=100%（转矩模式正方向速度限制） 3. P11-38=100%（转矩模式反方向速度限制） 用户须自行设定P11-36（速度限制选择）为0，此时操作频率限制依照参数P11-37和P11-38做限制。

7. **CurTQ** 为当前驱动器的扭矩数值，单位为 0.1 %，此数值会与特殊 SR 显示之该站号扭矩值相同。注：详细特殊 SR 编号与说明请参考 DRVIC 指令说明。

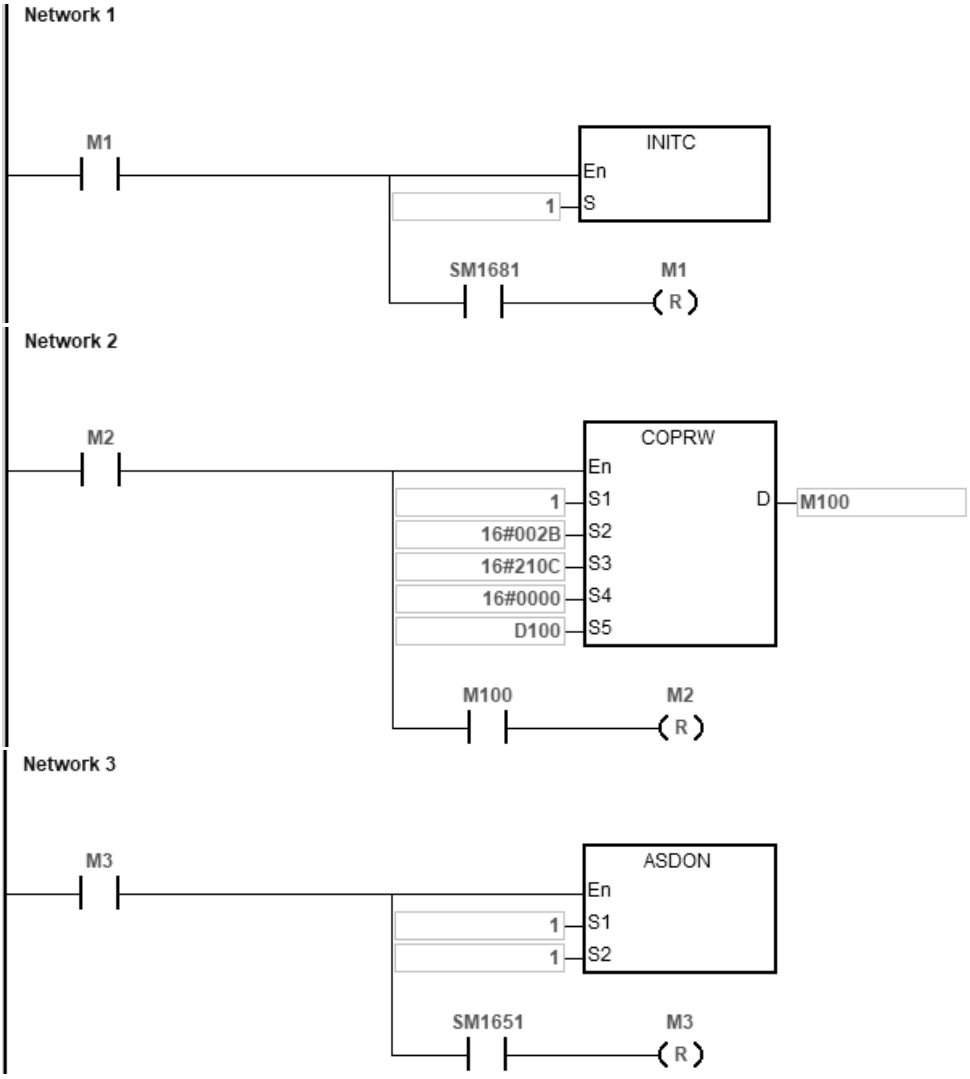
8. **Reach** 为 **CurTQ** 与 **TarTQ** 数值以 0.1 %为单位的比较结果，当 **Reach** 标志为 Off 并且 **CurTQ** >= **TarTQ** 时，此 **Reach** 标志将会被设定为 On；当 **Reach** 标志为 On 并且 **CurTQ** <（**TarTQ** – 5）时，此 **Reach** 标志将会被设为 Off；此指令之到达比较区间内定 5 为迟滞反映区，若此区间不符合用户需求，请自行另外编写比较判断区间。

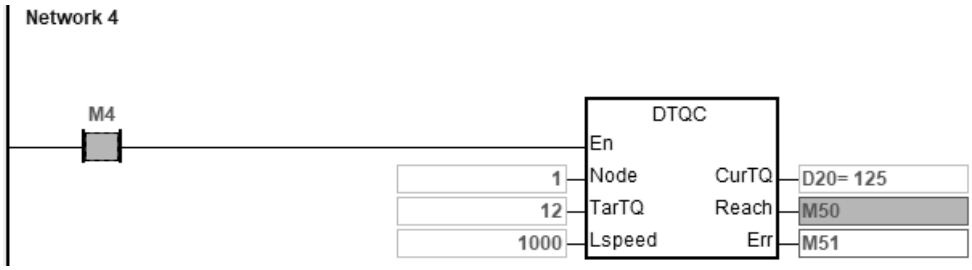
9. 若驱动器 **Node** 响应错误或者无回应时，PLC 将会设定 **Err** 错误标志为 On。若有需要查看错误站号与代码时，请参考 DRVIC 指令说明内 SR658 与 SR659 说明。
10. 由于扭矩与速度命令之数值范围，会因驱动器的型号与马力不同，而会有些不同，因此建议用户须事先参考台达驱动器操作手册说明。注意：本指令不会对用户输入之扭矩与速度命令进行检查与限制。

程序范例一（与台达伺服ASD-A2M通讯）：

1. ASD-A2 设定
   A. 将 P1-01 设置为 0003（T 扭矩控制模式）
   B. 须设定伺服 DI：TCM0，TCM1 选择内部扭矩指令 1，以默认状态下伺服的 DI3 及 DI4 是选择扭矩命令的设定，须将 DI3 设定为 ON

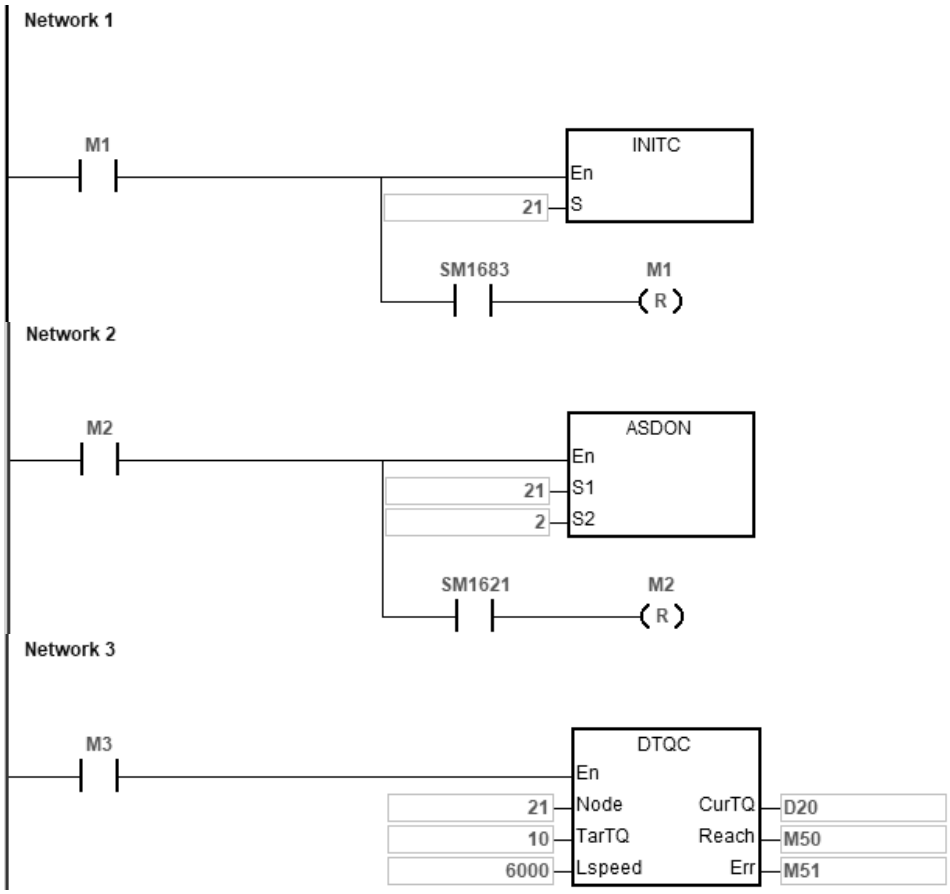
2. 当 M1 = OFF → ON 时，INITC 指令开始对伺服站号 1 下达初始化设定，直到 SM1681 被设为 ON。
3. 当 M2 = OFF → ON 时，设定将伺服的 P1-12 设定为 0
   COPRW 指令会将 D100 的内容值，写到 P1-12（**S**$_3$=16#210B），由于台达伺服 P1-12 初值非 0%，若采用扭矩模式时，不希望发生伺服 Servo On 时，就开始进行运转，请记得 Servo on 之前，自行先清除伺服 P1-12 的数值。
4. 当 M3 = OFF → ON 时，ASDON 指令开始对伺服站号 1，启动伺服，SM1651 标志为 ON，表示 SERVO-ON。
5. 当 M4 = OFF → ON 时，伺服以扭矩命令 P1-12=12%，开始转动，最大转速限制在 1000rpm，直到 M4 为 OFF。





程序范例二：(与台达变频器通讯)

1. VFD-C2000 设定
   A. 手动设定变频器 P09-36 站号 (范围限定 21~28 之间) ，P09-37 通讯速率 (变频默认为 1MHz) ，P09-40 为标准 CANopen 协议 (变频默认值) 。
   B. 设定命令来源 P00-20 为 6，P00-21 为 3，以及 P11-33 (转矩命令来源) 为 3 设定由 CANopen 通讯控制。
   C. 手动设定 P11-36 (速度限制选择) 为 0，此时操作频率限制依照参数 P11-37 和 P11-38 做限制
2. 当 M1 = OFF → ON 时，指令开始对变频站号 21 下达初始化设定，直到 SM1683 被设为 ON。
3. 当 M2 = OFF → ON 时，对变频站号 21 启动扭矩控制模式。
4. 当 M3 = OFF → ON 时，变频器以扭矩命令 10%，最高操作频率 60.00Hz，开始转动，直到 M3 为 OFF。



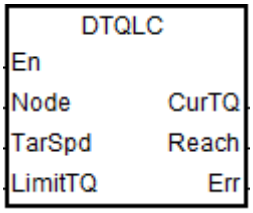
API	指令码			操作数	功能
2818	D	TQLC		**Node、TarSpd、…、Err**	扭矩限制之速度控制

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Node								●					○	○		
TarSpd								●					○	○		
LimitTQ								●					○	○		
CurTQ								●								
Reach		●	●	●				●								
Err		●	●	●				●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●				●							
TarSpd			●				●						
LimitTQ		●				●							
CurTQ		●				●							
Reach	●												
Err	●												

脉冲执行型	16 位指令	32 位指令
-	-	AS

符号：



Node ：驱动器站号

TarSpd ：目标速度命令

LimitTQ ：限制扭矩命令

CurTQ ：当前扭矩数值

Reach ：限制扭矩已到达标志

Err ：通讯错误标志

指令说明：

1. 执行此指令前，请先确认所有台达驱动器已经初始化完成。
2. 此指令适用于 AS 主机韧体版本 V1.06.20（含）以上，以及 ISPSoft 编辑软件 V3.07 版（含）以上。主机韧体版本 V1.14.00（含）以上，ISPSoft 编辑软件需 V3.16 版以上，支持 ECAT 通讯，须搭配 AS-FECAT 功能卡。
3. 伺服要设定 P1-01 为位置控制模式（PR），P1-01 设定须重新上电才会生效。变频要设定为速度模式，可用 ASDON 指令设定，再设定 P00-20（频率指令来源设定（AUTO））为 6（CANOpen）。
4. 使用 CAN 通讯，**Node** 为指定台达伺服与变频驱动器之站号，可使用的站号范围为 1~8（伺服驱动器）与 21~28（变频驱动器）之间；使用 ECAT 通讯，Node 可使用的站号范围为 1~16，指定站号的装置种类须为伺服，若设定超出此范围，则 PLC 将回应 **Err** 错误标志为 ON。

5. **TarSpd** 为目标速度控制命令，当限制扭矩数值未达到限制之目标数值时，用户可对此目标速度进行变更。

驱动器种类	**Node** 站号范围	TarSpd 数值范围	**TarSpd** 说明
伺服	1~8（CAN） 1~16（ECAT）	-60000 ~ 60000 （0.1rpm）	正负号代表正反转方向，举例：输入 6005 等于 600.5rpm
变频器	21~28（CAN）	-32768 ~ 32767 （1rpm）	正负号代表正反转方向，举例：输入 605 等于 605rpm

6. **LimitTQ** 为限制扭矩数值，此数值于指令启动时，通知驱动器设定一次，之后指令执行中，不允许变更。

   主机设定伺服 P1-12（内部扭矩限制 1），数值范围为-300~+300，单位为 1 %，超出上下限以上下限处理。V1.08.40 版（含）以上支持伺服 ASDA-A3/B3 高精度扭矩模式，数值范围为-3000~+3000，单位为 0.1 %，超出上下限范围，以上下限数值处理。伺服须设定 P1-02（速度及扭矩限制设定）开启扭矩限制，透过伺服 DI：TCM0，TCM1 选择内部扭矩限制 1。举例：P1-12 输入 30，则表示如下表所示

允许扭矩范围	正向运转扭矩限制	反向运转扭矩限制
-30% ~ +30%	30%	-30%

   主机设定变频设定 P11-17（正转电动转矩限制）和 P11-19（反转电动转矩限制），数值范围为 0~+500，单位为 1 %，超出上下限以上下限处理。举例输入如下表：

P11-17 设定值	P11-19 设定值	允许扭矩范围	正向运转扭矩限制	反向运转扭矩限制
+30	+30	-30% ~ +30%	30%	-30%

7. **CurTQ** 为当前驱动器的扭矩数值，单位为 0.1 %，此数值会与特殊 SR 显示之该站号扭矩值相同。注：详细特殊 SR 编号与说明请参考 DRVIC 指令说明。
8. **Reach** 为 **CurTQ** 与 **TarTQ** 数值以 0.1 %为单位的比较结果，当 **Reach** 标志为 Off 并且 **CurTQ** >= **LimitTQ** 时，此 **Reach** 标志将会被设定为 On；当 **Reach** 标志为 On 并且 **CurTQ** <（**LimitTQ** – 5）时，此 **Reach** 标志将会被设为 Off；此指令之到达比较区间内定 5 为迟滞反映区，若此区间不符合用户需求，请自行另外编写比较判断区间。
9. 若驱动器 **Node** 响应错误或者无回应时，PLC 将会设定 **Err** 错误标志为 On。若有需要查看错误站号与代码时，请参考 DRVIC 指令说明内 SR658 与 SR659 说明。
10. 由于扭矩与速度命令之数值范围，会因驱动器的型号与马力不同，而会有些不同，因此建议用户须事先参考台达驱动器操作手册说明。注意：本指令不会对用户输入之扭矩与速度命令进行数值范围检查与限制。
11. 在指令关闭的当下，将会自动送出恢复扭力限制值为 100%的命令，以利后续驱动器能进行其他定位命令。

程序范例一（与台达伺服ASD-A2M通讯）：

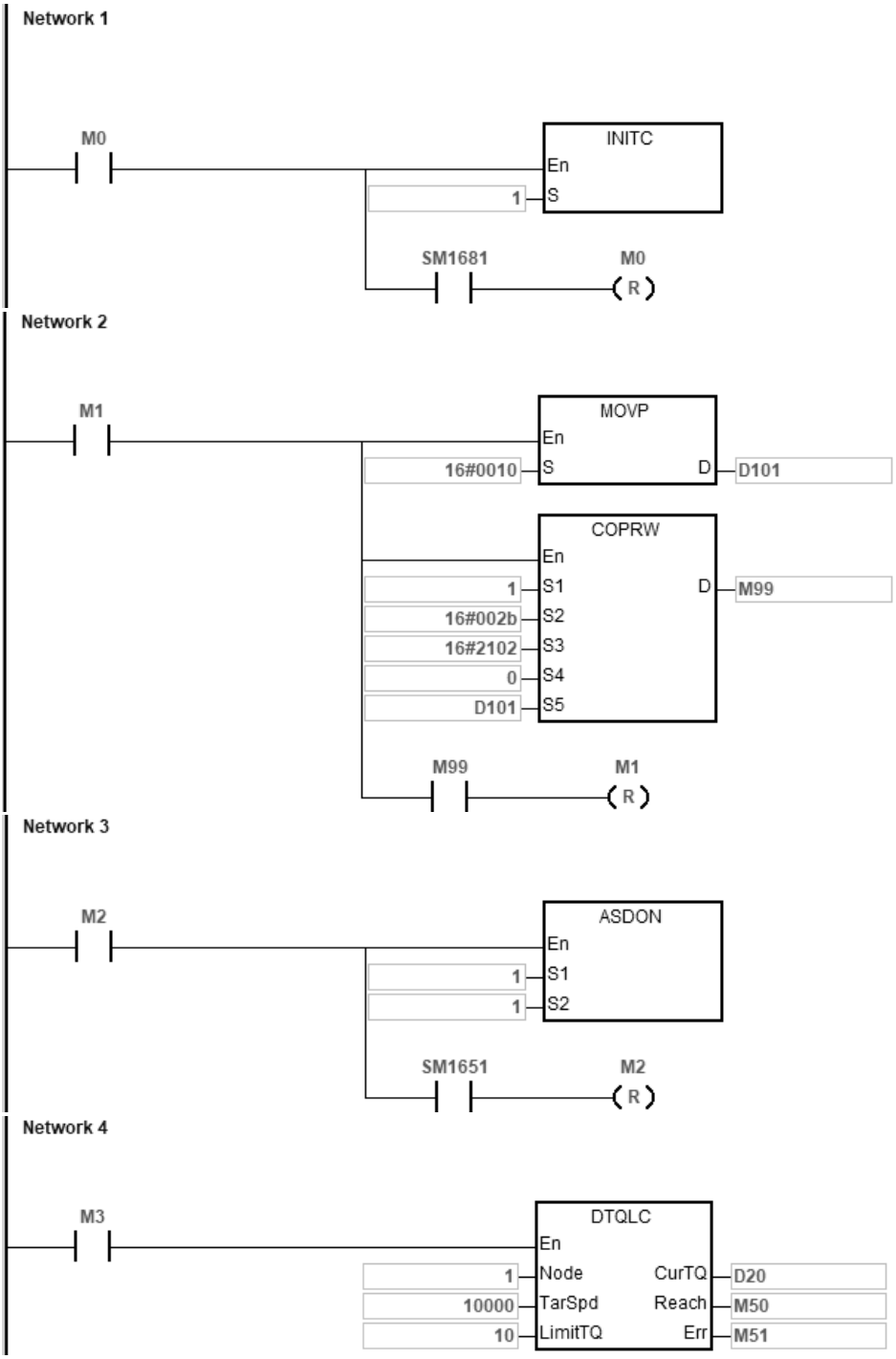
1. ASD-A2 设定

   A. 将 P1-01 设置为 0001（位置控制模式）

   B. 将 P1-02 设置为 0010（开启扭矩限制）或使用 COPRW 设定此参数

   C. 须设定伺服 DI：TCM0，TCM1 选择内部扭矩指令 1，以默认状态下伺服的 DI3 及 DI4 是选择扭矩命令的设定，须将 DI3 设定为 ON。

2. 当 M0 = OFF → ON 时，指令开始对伺服站号 1 下达初始化设定，直到 SM1681 被设为 ON。
3. 当 M1 = OFF → ON 时，透过 COPRW 指令将 P1-02 设置为 0010（开启扭矩限制）。
4. 当 M2 = OFF → ON 时，指令开始对伺服站号 1，启动伺服，SM1651 标志为 ON，表示 SERVO-ON。
5. 当 M3 = OFF → ON 时，伺服以 1000.0 RPM，开始转动，最大扭矩限制在 10%，直到 M3 为 OFF。



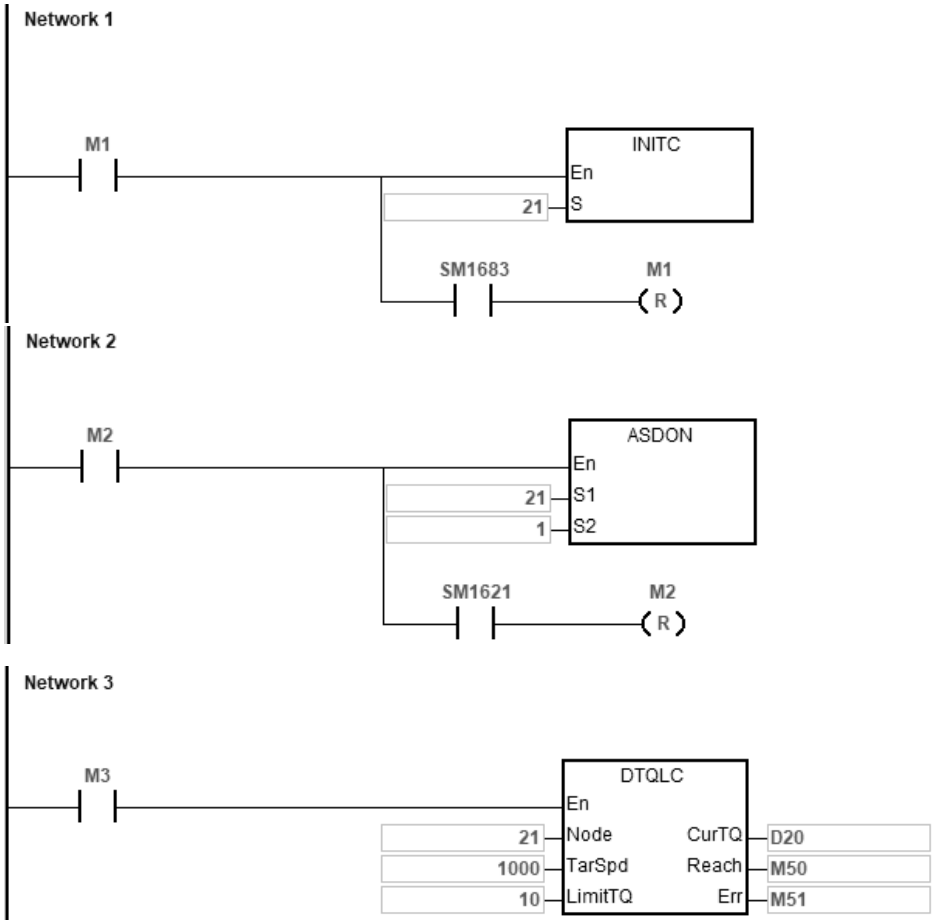
程序范例二：(与台达变频器通讯)

VFD-C2000 设定

A. 手动设定变频器 P09-36 站号(范围限定 21~28 之间)，P09-37 通讯速率(变频默认为 1MHz)，P09-40 为标准 CANopen 协议 (变频默认值) 。

B. 设定命令来源 P00-20 为 6，P00-21 为 3，设定由 CANopen 通讯控制。

1. 当 M1 = OFF → ON 时，指令开始对变频站号 21 下达初始化设定，直到 SM1683 被设为 ON。
2. 当 M2 = OFF → ON 时，对变频站号 21 启动速度控制模式。
3. 当 M3 = OFF → ON 时，变频器以 1000rpm 速度运转，最高扭矩限制在 10%，直到 M3 为 OFF。



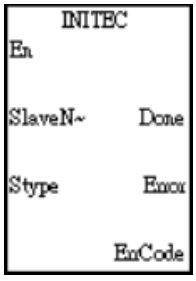
API	指令码	操作数	功能
2820	INITEC	**SlaveNo**、**SType**、**Done**、**Error**、**ErrCode**	专用 EtherCAT 通讯初始化

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
SlaveNo								●					○	○		
SType								●	●							
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
SlaveN		●				●							
SType		●				●							
Done	●												
Error	●												
ErrCod		●											

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



SlaveNo ： 初始化从站数量

SType ： 从站类型

Done ： 完成标志

Error ： 错误标志

ErrCode ： 错误代码

指令说明：

1. 执行初始化指令前，请确认是否安装 AS-FECAT 通讯卡。
2. 此指令适用于主机韧体版本 V1.14.00 (含) 以上，ISPSoft 编辑软件需 V3.16 版以上，须搭配 AS-FECAT 功能卡。
3. **SlaveNo** 可指定范围为 1~16 (此范围为 EtherCAT 支持最大从站数)，当数值超出此范围时，则 PLC 将认为初始化错误。注意：从站站号设定只能从站号 1 开始编号，其中间站号不可以跳过或保留。
4. **SType** 为指定从站的类型，并且为 Word 的 Array 类型变量，举例：若 **SlaveNo** 设定为 8，则此变量也必须设为 Array[8]的变量。台达驱动器的类型代码定义如下表所示：

类型代码	1	2	3	其余数值
驱动器型号	ASDA-A2-E	ASDA-A3-E ASDA-B3-E	C2000 CH2000	保留

注意：当类型代码与实际联机的驱动器比对不符合时，PLC 将设定错误标志与错误代码。

5. **Done** 为初始化完成标志。建议在执行初始化动作完成后，再开始进行后续的定位控制。

6. **Error** 为初始化错误标志，**ErrCode** 为错误代码。其错误代码如下表所示：

错误代码	错误说明
16#0000	无错误
16#0001	从站数量超出指定范围
16#0003	从站类型代码设定错误
16#0004	与 AS-FECAT 通讯逾时
16#0005	从站 SDO 正被其他指令使用中
16#0006	未安装 AS-FECAT 通讯卡
16#0007	从站状态初始化错误（AL STATUS UNKNOWN）
16#0008	从站状态初始化错误（AL STATUS ERROR）
16#0009	实际从站数量少于 INITEC 设定数量
16#000A	AS-FECAT 通讯卡与从站无实体接线
16#000B	AS-FECAT 通讯卡初始化逾时

7. 当初始化完成后，一旦发生断线错误的状况（SR0 = 0x1901~0x1910，后 2 码为从站 ID），并且为连动模式（默认值）（SM1684=OFF）时，暂停其他从站的相关动作；若用户在故障排除后，想再重新启动，则须重新对所有从站进行初始化动作，然后才可再进行后续执行动作。PLC 将在确认初始化完成后，自动关闭慢闪的错误灯号，但会留下错误记录。

8. 若需各站独立运行，不受其他站故障影响（前提通讯线正常连接），则可以在初始化后，设定 SM1684 为 ON 通知 PLC 仅记录错误与设定对应的错误标志即可，不需让其他通讯从站停止运行。

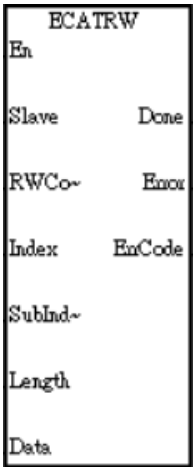
API	指令码	操作数	功能
2821	ECATRW	**Slave**、**RWCode**、**… ErrCode**	EtherCAT SDO 数据读写

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Slave								●					○	○		
RWCode								●					○	○		
Index								●								
SubIndex								●								
Length								●					○	○		
Data								●								
Done		●	●	●												
Error		●	●	●												
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
RWCode		●				●							
Index		●				●							
SubIndex		●				●							
Length		●				●							
Data		●	●			●	●						
Done	●												
Error	●												
ErrCode		●											

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Slave ： 从站站号

RWCode ： 读写代码

Index ： 索引

SubIndex 子索引

Length ： 读写长度

Data ： 读写数据的目的或来源

Done ： 完成标志

Error ： 错误标志

ErrCode ： 错误代码

指令说明：

1. 此指令乃是使用 SDO 通讯方式，对 EtherCAT 网络上的从站，读写其参数数值。
2. 此指令适用于主机韧体版本 V1.14.00 (含) 以上，ISPSoft 编辑软件需 V3.16 版以上，须搭配 AS-FECAT 功能卡。
3. **Slave** 可指定范围为 1~16 （此范围为 AS-FECAT 支持的最大从站数)，当数值超出初始化所设定的范围时，则 PLC 将认为设定错误。
4. **RWCode** 可指定设定的代码表如下所示：

代码编号	16#0000	16#0001	16#1000	16#1001
功能说明	单笔读取	多笔读取	单笔写入	多笔写入

注：若设定非上表定义之代码，PLC 将会设定为错误。当使用单笔读取或写入时，则指令的 **Length** 参数将会无效。当使用多笔读取或写入时，则会依据 **Length** 参数执行。

5. **Index / SubIndex** 为 SDO 通讯的索引和子索引。**Length** 为多笔读写时的长度值，其范围为 1~50 笔，若超过范围，自动修正为上下限。若为多笔读写，Index / SubInex 的数据变量需宣告为 Array 类型
6. **Data** 为读取时的目的存放数据位置，或写入时的源数据位置。若采用多笔的读写通讯，需注意该 Data 的数据变量需宣告为 Array 类型。每笔读写数据固定占用 2 个 word。
7. **Done** 为读写完成标志。建议执行读取动作时，需等待完成标志产生，才能使用已读取的数据。
8. **Error** 为通讯错误标志，**ErrCode** 为错误代码。其错误代码如下表所示：

错误代码	错误说明
16#0000	无错误
16#0001	从站站号超出指定范围或指定的从站站号未完成初始
16#0002	从站通讯无响应
16#0003	读写代码设定错误
16#0004	依读写长度，Index / SubIndex 装置超过范围
16#0005	依读写长度，data 装置超过范围
16#0006	未安装 AS-FECAT 功能卡
16#0007	从站接收信息错误，或从站响应通讯错误

API	指令码	操作数	功能
2822	MOTST	**Slave**，**Reset**， **… ErrCode**	EtherCAT 从站 Motion 状态读取

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Slave								●					○	○		
Reset			●													
Position								●								
State								●								
Done		●	●	●												
Error		●	●	●												
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
Reset	●												
Position			●				●						
State		●				●							
Done	●												
Error	●												
ErrCode		●											

脉冲执行型	16 位指令	32 位指令
-	AS	-

6

符号：

Slave ： 从站站号

Reset ： 清除定位控制权

Position ： 从站现在位置值（32bit）

State ： 从站现在状态值

Done ： 完成标志

Error ： 错误标志

ErrCode ： 错误代码

指令说明：

1. 此指令乃是使用 PDO 通讯方式，定时刷新从站的状态与现在位置，并且于扫描至指令所指定的站号，将自动刷新至指令设定的变量内。建议在初始化完成后，就一直启动执行刷新从站最新状态。
2. 此指令适用于主机韧体版本 V1.14.00（含）以上，ISPSoft 编辑软件需 V3.16 版以上，须搭配 AS-FECAT 功能卡。
3. **Slave** 可指定范围为 1~16 （此范围为 AS-FECAT 支持的最大从站数），当数值超出初始化所设定的范围时，则 PLC 将认为设定错误。

4. **Reset** 为 On 时，可清除指定的从站释放已被占用的定位控制权，以利其它定位指令(DDRVIC / DDRVAC / DPLSVC / ZRNC / DZRNC / DCSFOC / DTQC / DTQLC) 可重新启动该从站执行定位控制。当释放完控制权后，Reset 将会被自动清除为 Off。注意：正常定位控制中，若使用此清除动作后，会停止原本定位动作。

5. 指定从站站号的装置类型为伺服，**Position** 为该指定从站的现在位置值 (32bit) ; 指定从站站号的装置类型为变频，**Position** 数值固定为 0。

6. **State** 为该从站现在运行状态值，其状态值每个位 (bit) 为 On 时的定义如下表所示：

位编号	Bit 15~4	Bit 3	Bit 2	Bit 1	Bit 0
状态说明	保留	Servo On	暂停中	定位已到达	定位输出中

7. **Done** 为状态已更新完成标志。此标志于正常状况应都持续为 On 的状态，若出现为 Off 时，则上述现在状态、位置与速度值，都将维持最后一次更新的信息。

8. **Error** 为通讯错误标志，**ErrCode** 为错误代码。其错误代码如下表所示：

错误代码	错误说明
16#0000	无错误
16#0001	从站站号超出指定范围或指定从站站号未完成初始
16#0003	未安装 AS-FECAT 功能卡

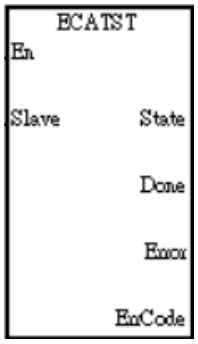
API	指令码	操作数	功能
2823	ECATST	**Slave**、**State**、**Done**、**Error**、**ErrCode**	EtherCAT 从站通讯状态读取

装置	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	“$”	F
Slave								●					○	○		
State								●								
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

数据类型	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
State		●				●							
Done	●												
Error	●												
ErrCod		●											

脉冲执行型	16 位指令	32 位指令
-	AS	-

符号：



Slave ： 从站站号

State ： EtherCAT 从站状态

Done ： 完成标志

Error ： 错误标志

ErrCode ： 错误代码

指令说明：

1. 执行前先完成 INITEC 初始化指令。
2. 此指令适用于主机韧体版本 V1.14.00 (含) 以上，ISPSoft 编辑软件需 V3.16 版以上，须搭配 AS-FECAT 功能卡。
3. 指令无使用次数限制，但是，同一台从站只允许同时启动一条指令更新状态。
4. **Slave** 可指定范围为 1~16 (此范围为 EtherCAT 支持最大从站数)，当数值超出此范围时，或指定的从站未完成初始，指令不执行。

5. **State**：**EtherCAT** 从站通讯状态

位编号	状态说明
Bit 0~3	1：初始化状态（Init State）
	2：预运行状态（Pre-Operational State）
	3：启动状态（Bootstrap State）
	4：安全运行状态（Safe-Operational State）
	8：运行状态（Operational State）
Bit 4	0：装置状态与要求状态一致或由命令清除
	1：装置未进入要求状态或由从站本机改变
Bit 5	0：Device Identification 未生效
	1：Device Identificaiton 生效
Bit 6~15	保留

6. **Done**为状态已更新完成标志，此标志于正常状况应都持续为On的状态，若出现为Off时，则上述EtherCAT从站状态，都将维持最后一次更新的信息。

7. **Error** 为通讯错误标志，**ErrCode** 为错误代码。其错误代码如下表所示：

错误代码	错误说明
16#0000	无错误
16#0001	从站站号超出指定范围或指定从站未完成初始
16#0002	从站通讯无响应
16#0003	未安装 AS-FECAT 功能卡
16#0004	相同从站站号启动两条以上的 ECATST 指令

6.27.3 台达专用 CANopen 通讯之常见问题与排除方法

常见问题与排除方法如下说明：

问题一：PLC 启动 INITC 初始化指令后，未看到伺服对应的初始化完成标志 SM1681 被设为 ON！

排除方法：

1) 重新确认 AS HWCONFIG 设定 Card 2 为 AS-FCOPM 通讯卡 并且通讯模式为台达专用驱动器模式？确认后再确认通讯速率是否为指定的速率？以上都正确时，请再重新下载一次硬件配置表，并再确认是否正常？若还未正常，则往下一步进行。
2) 重新确认是否已经手动设定伺服初始化一次？并且断电后再上电，若已执行，请往下一步排除。
3) 重新确认是否已经手动设定伺服的站号（P3-00）、通讯速率与模式（P3-01）与通讯模式（P1-01）为 PR 模式？并且已经有断电后再上电一次。若已确定，请往下一步进行排除。
4) 重新确认 CAN 通讯线是否连接正确？主机的 AS-FCOPM 卡上的终端电阻是否为 On 状态？（表示开启终端电阻）是否最后一台伺服有连接一颗终端电阻？若都有，则往下一步。
5) 观看 PLC 在启动初始化指令时，伺服上的数字显示器是否短暂出现 code 这字样？若无，则请重回第一步，或更换一台伺服。若有，则往下一步进行。
6) 检查伺服上的 DI0~DI7（P2-10 ~ P2-17）实际输入接脚状态与伺服参数设定功能是否相符？例如：DI0 为反极限输入脚且为 A 接点，则观看实际输入状态是否已经为 Off（B 接点）状态？若 DI 功能已经启动，则请先排除启动状态。若没有被启动，则往下一步进行。
7) 将 HWCONFIG 里的 AS-FCOMP 通讯卡的通讯模式切回 DS301 模式，利用 CANopen Builder 寻找从站，观看哪一台从站未被扫描出来？未被扫描到的那一台伺服，请先暂时移除，并且重新调整伺服站号与 PLC 初始化台数后，再进行一次初始化建立。

7

第7章 故障排除

目录

7.1 故障排除 7-3
7.1.1 基本检查 7-3
7.1.2 清除错误状态 7-3
7.1.3 故障排除程序 7-4
7.1.4 查询错误记录与位置 7-5
7.2 CPU 模块故障排除 7-7
7.2.1 ERROR 灯号常亮 7-7
7.2.2 ERROR 灯号一般闪烁（亮 0.5 秒、暗 0.5 秒） 7-7
7.2.3 ERROR 灯号快速闪烁（亮 0.2 秒、暗 0.2 秒） 7-9
7.2.4 ERROR 灯号慢速闪烁（亮 1 秒、暗 3 秒） 7-10
7.2.5 BAT. LOW 灯号常亮 7-10
7.2.6 BAT. LOW 灯号一般闪烁（亮 0.5 秒、暗 0.5 秒） 7-11
7.2.7 RUN 与 ERROR 灯号同时闪烁（亮 0.5 秒、暗 0.5 秒） 7-11
7.2.8 RUN 与 ERROR 灯号交替闪烁（RUN 亮 0.5 秒、换 ERROR 亮 0.5 秒） 7-11
7.2.9 其它错误但无灯号显示 7-11
7.3 其它模块故障排除 7-18
7.3.1 模拟（AD/DA/XA）及温度（RTD/TC）模块故障排除 7-18
7.3.1.1 ERROR 灯常亮 7-18
7.3.1.2 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒） 7-18
7.3.2 AS02/04PU 定位模块故障排除 7-19
7.3.2.1 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒） 7-19
7.3.3 AS02HC 高速计数模块故障排除 7-19
7.3.3.1 ERROR 灯常亮 7-19
7.3.3.2 ERROR 灯闪烁（亮 0.5 秒、暗 0.5 秒） 7-19
7.3.4 AS02LC 称重模块故障排除 7-20

7.3.4.1 ERROR 灯常亮 7-20
7.3.4.2 ERROR 灯闪烁 (亮 0.2 秒、暗 0.2 秒) 7-20
7.3.5 AS04SIL IO-Link 通讯模块故障排除 7-21
7.3.5.1 MS 灯闪烁 (亮 0.2 秒、暗 0.2 秒) 7-21
7.3.5.2 NS 灯恒亮 7-21
7.3.5.3 NS 灯闪烁 (亮 1 秒、暗 1 秒) 7-21
7.3.6 AS00SCM 为串行通讯模块故障排除 7-23
7.3.6.1 ERROR 灯常亮 7-23
7.3.6.2 ERROR 灯一般闪烁 (亮 0.5 秒、暗 0.5 秒) 7-23
7.3.7 AS00SCM 为 AS 远程模块故障排除 7-24
7.3.7.1 ERROR 灯常亮 7-24
7.3.7.2 ERROR 灯慢速闪烁 (亮 1 秒、暗 1 秒) 7-24
7.3.7.3 ERROR 灯一般闪烁 (亮 0.5 秒、暗 0.5 秒) 7-25
7.3.7.4 ERROR 灯快速闪烁 (亮 0.2 秒、暗 0.2 秒) 7-25
7.3.8 AS01DNET 为通讯模块故障排除 7-25
7.4 模块错误代码对应灯号及状态说明 7-26
7.4.1 CPU 模块错误代码对应灯号及状态 7-26
7.4.2 模拟与温度模块错误代码对应灯号 7-32
7.4.3 AS02/04PU 定位模块错误代码对应灯号 7-33
7.4.4 AS02HC 高速计数模块错误代码对应灯号 7-33
7.4.5 AS02LC 称重模块错误代码对应灯号 7-33
7.4.6 AS04SIL IO-Link 通讯模块错误代码对应灯号 7-34
7.4.7 AS00SCM 为串行通讯模块错误代码对应灯号 7-35
7.4.8 AS00SCM 为 AS 远程模块错误代码对应灯号 7-35
7.4.9 AS01DNET 为主从站模式时的数字显示器显示说明 7-35
7.4.10 AS01DNET 为 RTU 模式时的数字显示器显示说明 7-37

7.1 故障排除

7.1.1 基本检查

本章说明系统运行时可能发生的各种类型的故障，以及这些故障的原因和处理方法。排除故障须先确认故障的原因。在分析故障原因之前请先确认以下三点：

(1) 检查以下项目

- PLC 系统必须处于规范的条件下运作（包括环境、电气、机械震动…等条件）
- 设备电源妥善接通，输入至 PLC 系统的电源有正常开启
- 各模块、端子与电缆之间有妥善安装
- 各个显示灯号为正常状态
- 各种设定开关位于正确位置

(2) 按照下述操作 AS 系统，以检查运行情况

- 切换 RUN/STOP 开关
- 确认主机所设定的 RUN/STOP 条件
- 确认并排除外部装置可能造成的影响
- 以 ISPSoft 监控系统的运作状况及错误记录

(3) 依照以上（1）（2）两点判断故障发生原因的范围

- AS 系统或者外部装置
- 主机或扩展模块
- 参数设定或控制程序

7.1.2 清除错误状态

当系统处于错误状态时，可尝试以下述几种方式清除系统的错误状态，但在清除之后，若错误来源仍然存在，则系统还是会再次进入错误状态。

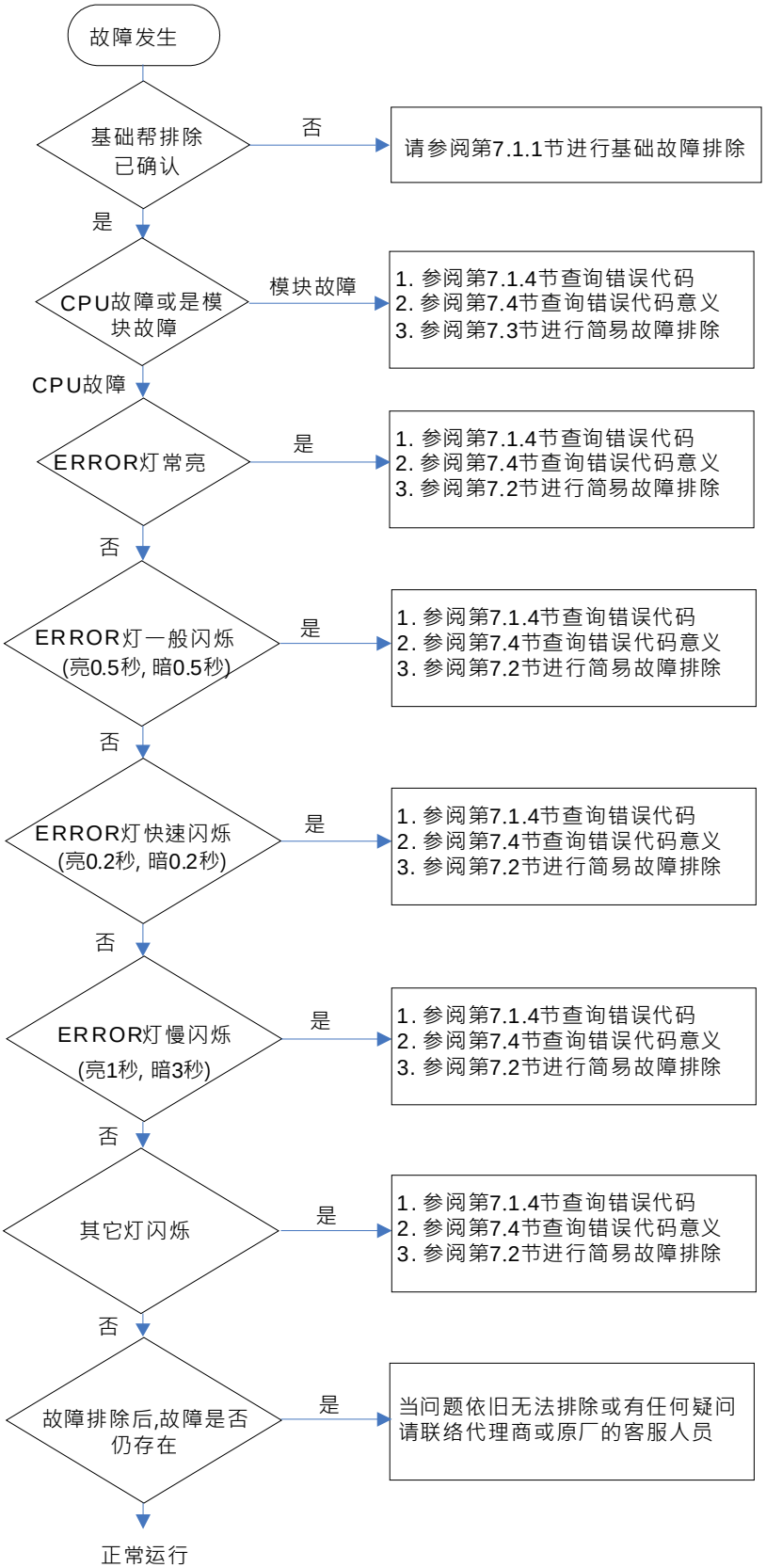
(1) 将主机状态切换至 STOP 后再切换至 RUN

(2) 将主机断电后重开

(3) 以 ISPSoft 清除错误记录

(4) 重置主机或将主机回复至出厂默认值后，重新下载项目并执行。

7.1.3 故障排除程序

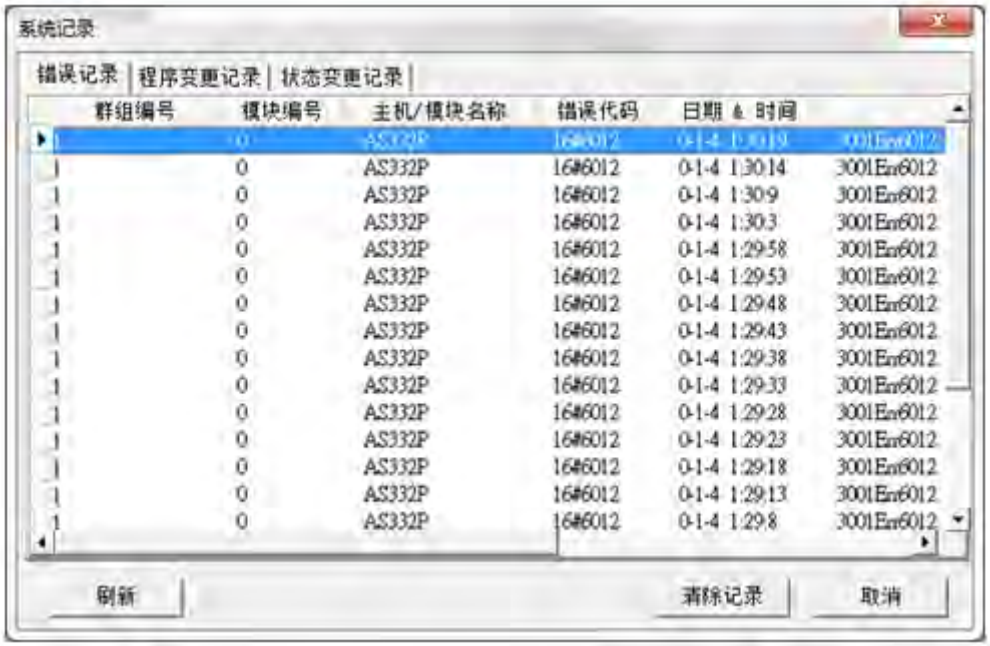

故障发生
基础帮排除已确认
否
请参阅第7.1.1节进行基础故障排除
是
CPU故障或是模块故障
模块故障
1. 参阅第7.1.4节查询错误代码
2. 参阅第7.4节查询错误代码意义
3. 参阅第7.3节进行简易故障排除
CPU故障
ERROR灯常亮
是
1. 参阅第7.1.4节查询错误代码
2. 参阅第7.4节查询错误代码意义
3. 参阅第7.2节进行简易故障排除
否
ERROR灯一般闪烁(亮0.5秒, 暗0.5秒)
是
1. 参阅第7.1.4节查询错误代码
2. 参阅第7.4节查询错误代码意义
3. 参阅第7.2节进行简易故障排除
否
ERROR灯快速闪烁(亮0.2秒, 暗0.2秒)
是
1. 参阅第7.1.4节查询错误代码
2. 参阅第7.4节查询错误代码意义
3. 参阅第7.2节进行简易故障排除
否
ERROR灯慢闪烁(亮1秒, 暗3秒)
是
1. 参阅第7.1.4节查询错误代码
2. 参阅第7.4节查询错误代码意义
3. 参阅第7.2节进行简易故障排除
否
其它灯闪烁
是
1. 参阅第7.1.4节查询错误代码
2. 参阅第7.4节查询错误代码意义
3. 参阅第7.2节进行简易故障排除
否
故障排除后,故障是否仍存在
是
当问题依旧无法排除或有任何疑问请联络代理商或原厂的客服人员
正常运行

7.1.4 查询错误记录与位置

错误发生时，系统将产生一个对应的错误代码并记录在主机中。主机至多可储存 20 笔错误记录，而当超出 20 笔的记录产生时，最旧的该笔记录将被最新的记录覆盖，但若主机有安装储存卡时，在记录被覆盖前，系统将自动将主机的 20 笔记录备份至储存卡中，而储存卡至多可储存 1000 笔记录，同样的，当储存卡的错误记录超出 1000 笔时，最旧的 20 笔记录将被最新的 20 笔记录覆盖。

(1) 启动 ISPSoft 后，在功能工具栏中点选 **PLC 主机（P）> 系统记录**。

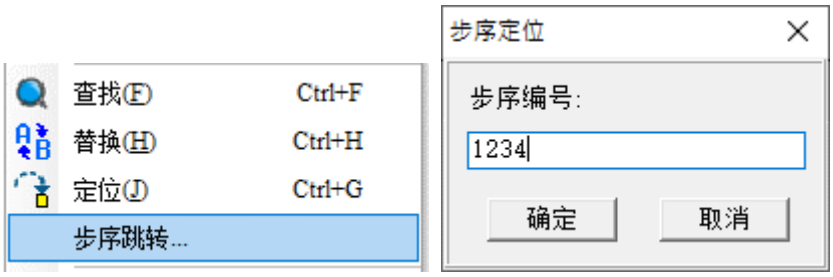
(2) 错误记录显示画面如下所示，按下更新键可更新记录显示，而按下**清除记录**按钮则可清除主窗口及主机中的错误记录。



- 群组编号：若该记录为”1“，则表示记录 CPU 主机或主机右侧模块编号，若编号显示为 2~16 时，则代表产生错误为第 1~15 台远程模块的编号。
- 模块编号：若该记录为”0“，则代表产生错误的是 CPU 主机或远程模块本身，若编号显示为 1~32 时，则代表产生错误为主机或远程模块的右侧模块编号（最靠近主机或远程模块为 1，由左往右累加计数）。备注：远程右侧最多为 8 台扩展模块。
- 主机/模块名称：此字段为 CPU 主机、远程或扩展模块的识别型号名称。
- 错误代码：此字段显示的为该笔记录的错误代码。
- 日期&时间：错误发生的日期与时间，越近期的错误将显示在最上层。
- 最后的字段为该错误的相关简述。

(3) 当下载或执行 PLC 程序过程中，遇到 PLC 回复文法检查（SM5=ON）（错误码纪录于 SR4）或装置使用超出范围（SM0=ON）（错误码纪录于 SR0）的错误时，可经由底下手法找出错误位置。

- 当 SM0=ON 时，请查看 SR1 的 32 位数值，此 SR1 记录最近一次发生错误时的程序位置。
- 当 SM5=ON 时，请查看 SR5 的 32 位数值，此 SR5 记录最近一次发生错误时的程序位置。
- 得知错误位置之后，可再藉由 ISPSoft 软件→编辑→步序定位，输入该错误位置编号，让编辑画面跳至 PLC 程序被系统记录错误的位置。



7.2 CPU 模块故障排除

请依据 CPU 模块上的 LED 指示灯号状态及错误代码，从以下表格中获知故障排除方式。以下表格中【记录】一栏，标记说明如下：「V」表此错误代码有记录在错误记录表内；「X」表错误发生后不会记录在错误记录表；「H」表可由 HWCONFIG 设定是否储存至错误记录表内。(备注：历史错误记录同时也会被保存于 SR 内，请参考第 2 章的 SR 说明)

7.2.1 ERROR 灯号常亮

错误代码 (16#)	错误说明	排除方式	错误标志	记录
000A	扫描逾时	1. 确认 HWCONFIG 中，CPU 模块参数的 WDT 设定 2. 确认程序是否有造成扫描时间过长的设计	SM8	V

7.2.2 ERROR 灯号一般闪烁 (亮 0.5 秒、暗 0.5 秒)

错误代码 (16#)	错误说明	排除方式	错误标志	记录
000C	PLC 程序毁损	重新下载项目程序	SM9	V
000D	装置初始值与 PLC 程序未完整下载	重新下载项目程序	SM9	V
0010	主机内部存储器检测错误	请联络原厂	SM9	V
0020	CPU 执行速度未达最高速	请联络原厂	SM9	V
0021	台达设备参数还原错误	1. 确认要还原参数的台达设备与参数备份文件所记录的设备是否一致。 2. 确认主机与要还原参数的台达设备网络联机是否正常。 3. 确认 HWCONFIG 台达设备参数还原的兼容性检查设定。	SM9	H
002E	主机外部内存检测错误	请联络原厂	SM9	V
002F	PLC 程序与系统记录比对不符	重新下载项目程序	SM34	V
0070	功能卡与手动配置不符	确认 HWCONFIG 中的功能卡配置设定与实际的功能卡配置是否吻合。	SM10	V
0102	中断编号使用超过范围	查询错误位置、修改程序并重新编译后，再次下载项目至 PLC。	SM5	X
0202	MC 指令超过有效范围		SM5	X
0302	MCR 指令超过有效范围		SM5	X
0D03	DHSCS 指令操作数使用不当		SM5	X

错误代码（16#）	错误说明	排除方式	错误标志	记录
0E05	DCNT 指令操作数 HCXXX 使用不当		SM5	X
1000 ~ 10FF	系统性错误	请联系技术人员	-	V
1300 ~ 130F	远程模块发出错误信息	请参考远程模块第 7.3.4 节错误代码说明	SM30	V
1402	实际 IO 模块不符合配置设定	确认 HWCONFIG 中的模块配置设定与实际的模块配置是否吻合	SM10	V
140B	通讯模块超出 4 台限制	确认通讯模块台数	SM10	V
140C	定位模块超出 8 台限制	确认定位模块台数	SM10	V
140D	所有扩展模块超出 32 台限制	确认扩展模块台数	SM10	V
140E	远程模块右侧连接超出 8 台限制	确认远程模块右侧连接台数	SM30	V
1600	扩展模块 ID 配置超出范围	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。	SM10	V
1601	扩展模块无法配置 ID	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。	SM10	V
1602	扩展模块 ID 配置重复	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。	SM10	V
1603	扩展模块无法进入运行模式	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。	SM10	V
1604	扩展模块通讯逾时	1. 确认模块与主机连接，重新上电。 2. 若一再出现此信息，请联络原厂。	SM10	V
1605	扩展模块硬件错误	请返回原厂维修	SM10	V
1606	通讯模块功能卡配置错误	请确认功能卡是否正确或确实连接	SM10	V
1607	扩展模块外部电源错误	请确认外部 24VDC 供电是否正常	SM10	V
1608	扩展模块校正值或 CJC 补偿异常	请返回原厂重新进行校验	SM10	V
1609 ~ 160F	保留（扩展模块错误代码）			
200A	无效的指令	查询错误位置、修改程序并重新编译后，再次下载项目至 PLC。	SM5	V
6010	MODBUS TCP 联机数超出范围	检查上位设备数量是否超过 32 台	SM 1092	V

错误代码（16#）	错误说明	排除方式	错误标志	记录
6011	EtherNet/IP 联机数超出范围	检查设备连接数量是否超过 16 台	SM 1093	V
C000 ~ CFFF	PLC 程序有语法不合法错误	请将 PLC 程序存盘，并交由代理(经销)商反映给原厂。	-	V

7.2.3 ERROR 灯号快速闪烁（亮 0.2 秒、暗 0.2 秒）

表示 CPU 模块的输入电源 24VDC 正在断电进行中，或者电源供应不足或不稳定，造成无法运行。

错误代码（16#）	错误说明	排除方式	错误标志	记录
002A	曾经发生过 24V 低电压又恢复	确认电源系统是否稳定	SM7	V

7.2.4 ERROR 灯号慢速闪烁（亮 1 秒、暗 3 秒）

错误代码（16#）	错误说明	排除方式	错误标志	记录
0040	功能卡 1 AS-F2AD 或是内建 AD 输入通道 1 数值转换超过范围	确认实际输入信号	SM27	H
0041	功能卡 1 AS-F2AD 或内建 AD 输入通道 2 数值转换超过范围	确认实际输入信号	SM27	H
0042	功能卡 2 AS-F2AD 输入通道 1 数值转换超过范围	确认实际输入信号	SM27	H
0043	功能卡 2 AS-F2AD 输入通道 2 数值转换超过范围	确认实际输入信号	SM27	H
0044	功能卡 1 AS-F2AD 或内建 AD 输入通道 1 电流输入 4mA~20mA 模式断线	确认接线	SM27	H
0045	功能卡 1 AS-F2AD 或内建 AD 输入通道 2 电流输入 4mA~20mA 模式断线	确认接线	SM27	H
0046	功能卡 2 AS-F2AD 输入通道 1 电流输入 4mA~20mA 模式断线	确认接线	SM27	H
0047	功能卡 2 AS-F2AD 输入通道 2 电流输入 4mA~20mA 模式断线	确认接线	SM27	H
1500	远程模块断线警告信息	请检查通讯线是否脱落或接触不良	SM30	V
1502 ~ 150F	远程模块发出警告信息	参考第 7.3.4 节远程模块警告信息	SM30	V
1800 ~ 180F	扩展模块发出警告信息	参考第 7.3 节模块警告信息	SM10	V
1900 ~ 191C	CANopen/ECAT 专用通讯模式的从站发生心律（Heartbeat）/断线错误信息	请检查 CANopen/ECAT 通讯线是否正常连接，或者从站是否正常运作中（后两码为错误从站 ID 编号（16 进制须转 10 进制））	-	V

7.2.5 BAT. LOW 灯号常亮

此灯号表示未安装电池（CR1620）或电池低电压的警告，若用户不须 CPU 断电后持续让万年历计时，则可透过硬件组态（HWCONFIG→CPU→系统参数）选择”关闭”电池低电压灯号显示。（预设为启动）

错误代码（16#）	错误说明	排除方式	错误标志	记录
0027	电池低电压	更换电池或由 HWCONFIG 关闭灯号显示	SM219	X

7.2.6 BAT. LOW 灯号一般闪烁（亮 0.5 秒、暗 0.5 秒）

此闪烁灯号表示万年历计时功能异常，须回厂进行维修。

错误代码（16#）	错误说明	排除方式	错误标志	记录
0026	万年历无法计时	请联络原厂	SM218	V
0028	万年历读写无回应	请联络原厂	SM217	V

7.2.7 RUN 与 ERROR 灯号同时闪烁（亮 0.5 秒、暗 0.5 秒）

此状况为显示 CPU 韧体正在更新烧录中。若此状态是发生在 CPU 刚上电时，则表示上次韧体更新失败需重新烧录韧体提示，故请重新烧录 CPU 韧体，或联络代理商服务人员。

7.2.8 RUN 与 ERROR 灯号交替闪烁（RUN 亮 0.5 秒、换 ERROR 亮 0.5 秒）

此状况为显示 CPU 模块正在进行 SD 卡的 PLC 程序备份、还原动作，以及储存其它信息。

若未使用 SD 卡且 CPU 模块为 AS218 系列，则表示主机 AIO 韧体正在更新烧录中。若此状态是发生于 CPU 刚上电时，则表示上次主机内建 AIO 韧体更新失败需重新烧录韧体提示，故请重新烧录 AIO 韧体。若未更新 AIO 韧体，则表示主机内部检测 AIO 单元异常，请联络代理商服务人员。

7.2.9 其它错误但无灯号显示

错误代码（16#）	错误说明	排除方式	错误标志	记录
0011	PLC 标识符错误	确认 PLC 标识符	SM34	V
0012	PLC 主密码错误	确认 PLC 主密码	SM34	V
002D	PLC 主密码错误次数达到上限	无法解除主密码设定，请恢复出厂值	SM34	V
0050	停电保持区 SM 储存区块异常	1. 恢复出厂值后，重新下载程序与参数。 2. 若一再出现此信息，请联络原厂。	SM6	V
0051	停电保持区 SR 寄存器异常	1. 恢复出厂值后，重新下载程序与参数。 2. 若一再出现此信息，请联络原厂。	SM6	V
0052	停电保持区 M 储存区块异常	1. 恢复出厂值后，重新下载程序与参数。 2. 若一再出现此信息，请联络原厂。	SM6	V
0054	停电保持区 C 寄存器异常	1. 恢复出厂值后，重新下载程序与参数。 2. 若一再出现此信息，请联络原厂。	SM6	V
0055	停电保持区 HC 寄存器异常	1. 恢复出厂值后，重新下载程序与参数。 2. 若一再出现此信息，请联络原厂。	SM6	V
0056	停电保持区 S 储存区块异常	1. 恢复出厂值后，重新下载程序与参数。 2. 若一再出现此信息，请联络原厂。	SM6	V
0059	停电保持区 D 寄存器异常	1. 恢复出厂值后，重新下载程序与参数。 2. 若一再出现此信息，请联络原厂。	SM6	V

错误代码（16#）	错误说明	排除方式	错误标志	记录
005D	CPU 模块检测不到储存卡	确认储存卡是否正确插入 CPU 模块中	SM453	V
005E	储存卡的初始程序错误	确认储存卡是否故障	SM453	V
0063	数据写入储存卡的文件时有错误	确认文件路径是否正确，或储存卡是否故障。	SM453	V
0064	储存卡的文件无法被读取	确认文件路径是否正确，文件是否损毁。	SM453	V
1950	CANopen 专用通讯模式未初始化完成，指令禁止执行	请检查 CANopen 通讯线是否正常连接，从站是否正常运作，确认正常后可重新执行初始化指令。	-	V
19B0	从站模式之心跳信息逾时	检查 CANopen 网络中网络连接是否正常	-	V
19B1	从站模式之 PDO 数据长度不符	重新设定从站的 PDO 数据长度并下载	-	V
注：错误代码 19E1~19E8，请比对 SR830~SR893 记录从站 1~64 的错误代码，查询哪一台从站发生错误				
19E1	从站返回的 PDO 数据长度与扫描列表中配置的不符 备注：错误代码 19E1~19E8，建议另外参考操作手册之 CANopen 通讯说明	设定从站的 PDO 数据长度重新下载	-	V
19E2	未接收到从站 PDO	检查并确认设定正确	-	V
19E3	初次启动时的自动 SDO 下载失败	检查并确认自动 SDO 内容是否正确	-	V
19E4	PDO 参数配置失败	确认 PDO 参数设定合法	-	V
19E5	关键参数不匹配	确认所连接的从站与所设定的从站一致	-	V
19E6	网络中不存在此从站	确认从站工作电源正常，确认网络连接正常	-	V
19E7	从站错误控制逾时	确认从站工作电源正常，确认网络连接正常	-	V
19E8	主从站站号重复	重新设定主站或从站站号，确认重新设定后的站号不重复	-	V
19F3	配置表错误	1. 重新下载配置表。 2. 若一再出现此信息，请联络原厂。	-	V
19F4	CAN 通讯 BUS-OFF 状态	1. 检查 CANopen 网络中网络接线与终端电阻是否正确？ 2. 确认网络上所有的节点都有相同的波特率？	-	V
19FB	发送寄存器超出最大容量	请调整同步时间（建议延长时间）	-	V
19FC	接收寄存器超出最大容量	请调整同步时间（建议延长时间）	-	V
2001	使用台达专用 CANopen 通讯指令时，未插 FCOMP 卡或不是在 ASDA-A2 专用模式	1. 确认功能卡 2 的卡别及工作模式。 2. 查询错误位置、修改程序并重新编译后，再次下载项目至 PLC。	SM0	V

错误代码（16#）	错误说明	排除方式	错误标志	记录
2003	程序中使用的装置超过可用范围或不合法	查询错误位置、修改程序并重新编译后，再次下载项目至 PLC。	SM0	V
200B	n 操作数或其它 K/H 操作数超出范围		SM0	V
200C	部份指令不充许操作数发生重叠		SM0	V
200D	BIN 转成 BCD 时发生错误		SM0	V
200E	字符串没有 00 当做结尾	查询错误位置、修改程序并重新编译后，再次下载项目至 PLC。	SM0	V
2012	除法运算错误		SM0	V
2013	浮点数格式错误，超出可转换范围		SM0	V
2014	TKON/TKOFF 指令所指定的 TASK 编号错误或超出范围		SM0	V
2017	BREAK 指令写在 FOR-NEXT 之外		SM0	V
2027	无此定位规划表编号或是定位规划表模式不对	1.查询错误位置、修改程序并重新编译后，再次下载项目至 PLC。 2. 确认定位规划表设定	SM0	V
2028	高速输出指令正在输出中，不可重复输出	请参考 SR28 记录的轴数编号，重新调整输出的流程。	-	V
2030	CSFO 指令之输出频率超出最高限制频率	请调整输入与输出比率值，或调降输入频率	-	V
2031	CSFO 指令于输入频率为 0 时，该轴输出超过 2 秒还未输出完已换算的输出个数	请调整输入与输出比率值，或调降输入频率	-	V
6004	以太网络的 IP 地址过滤设置错误	重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/IP 过滤器。	SM1108	X
600D	RJ45 埠未连接	检查通信电缆	SM1100	X
6012	网络上有重复 IP 设备	1. 检查网络上是否有相同设备 IP 存在。 2. 检查网络上是否有一个以上 DHCP 服务器或 BOOTP 服务器。	SM1101	V
6100	E-mail 联机忙碌	稍后重试（此错误不会造成 PLC 停机，因此请在程序中利用相关标志，建立对应的处置方式。）	SM1113	X

错误代码（16#）	错误说明	排除方式	错误标志	记录
6103	E-mail 附件中的装置地址设定错误	重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/邮件/邮件发送条件。	SM1113	X
6104	E-mail 附件不存在	确认附件文件是否存在在储存卡中。	SM1113	X
6105	E-mail 附件超过容量	确认储存卡中指定为附件的文件大小，若超过 2MB 将无法指定为附件。	SM1113	X
6106	SMTP 服务器地址错误	确认正确的地址后，重设 HWCONFIG 中，CPU 模块/以太网进阶设定/邮件。	SM1113	X
6107	SMTP 服务器逾时	1. 检查 SMTP 服务器的状态是否正常。 2. 稍后重试（此错误不会造成 PLC 停机，因此请在程序中利用相关标志，建立对应的处置方式。）	SM1113	X
6108	寄件服务器验证错误	确认账号密码，并重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/邮件。	SM1113	X
6200	TCP 通讯接口（Socket）功能的远程 IP 地址不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/TCP 设定。	-	X
6201	TCP 通讯接口（Socket）功能的本地端口不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/TCP 设定。	-	X
6202	TCP 通讯接口（Socket）功能的远程端口不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/TCP 设定。	-	X
6203	TCP 通讯接口（Socket）功能的传送数据地址不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/TCP 设定。	-	X
6206	TCP 通讯接口（Socket）功能的接收数据地址不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/TCP 设定。	-	X
6208	TCP 通讯接口（Socket）功能的接收数据装置超出范围	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/TCP 设定。	-	X
6209	UDP 通讯接口（Socket）功能的远程 IP 地址不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/UDP 设定。	-	X

错误代码（16#）	错误说明	排除方式	错误标志	记录
620A	UDP 通讯接口（Socket）功能的本地端口不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/UDP 设定。	-	X
620C	UDP 通讯接口（Socket）功能的传送数据地址不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/UDP 设定。	-	X
620F	UDP 通讯接口（Socket）功能的接收数据地址不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/UDP 设定。	-	X
6210	UDP 通讯接口（Socket）功能的接收数据长度不合法	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/UDP 设定。	-	X
6212	远程装置响应逾时	确认远程装置仍保持联机。	-	X
6213	接收数据超过限制	1. 检查程序与相关的 SR。 2. 重新设定 HWCONFIG 中，CPU 模块/以太网进阶设定/通讯接口/TCP 设定和 UDP 设定。	-	X
6214	远程装置拒绝联机	确认远程装置运作正常	-	X
6215	目前通讯接口(Socket)未开启	检查程序在操作时序的设计上是否正确	-	X
6217	目前通讯接口(Socket)已开启	检查程序在操作时序的设计上是否正确	-	X
6218	目前通讯接口(Socket)已传送	检查程序在操作时序的设计上是否正确	-	X
6219	目前通讯接口(Socket)已接收	检查程序在操作时序的设计上是否正确	-	X
621A	目前通讯接口(Socket)已关闭	检查程序在操作时序的设计上是否正确	-	X
7011	COM1 通讯功能码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7012	COM1 装置组件通讯地址错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7013	COM1 需求组件已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7014	COM1 需求通讯长度已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7017	COM1 通讯数据串的检查码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7021	COM2 通讯功能码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H

错误代码（16#）	错误说明	排除方式	错误标志	记录
7022	COM2 装置组件通讯地址错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7023	COM2 需求组件已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7024	COM2 需求通讯长度已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7027	COM2 通讯数据串的检查码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7031	Ethernet 通讯功能码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7032	Ethernet 装置组件通讯地址错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7033	Ethernet 需求组件已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7034	Ethernet 需求通讯长度已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7037	Ethernet 通讯数据串的检查码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7041	USB 通讯功能码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7042	USB 装置组件通讯地址错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7043	USB 需求组件已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7044	USB 需求通讯长度已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7047	USB 通讯数据串的检查码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70B1	功能卡 1 通讯功能码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70B2	功能卡 1 装置组件通讯地址错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70B3	功能卡 1 需求组件已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H

错误代码（16#）	错误说明	排除方式	错误标志	记录
70B4	功能卡 1 需求通讯长度已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70B7	功能卡 1 通讯数据串的检查码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70C1	功能卡 2 通讯功能码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70C2	功能卡 2 装置组件通讯地址错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70C3	功能卡 2 需求组件已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70C4	功能卡 2 需求通讯长度已超出限制范围	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
70C7	功能卡 2 通讯数据串的检查码错误	1. 检查两方的通信设置。 2. 检查通信电缆。	-	H
7203	无效通讯功能码	1. 检查通讯协议所定义之通讯功能码 2. 检查软件与韧体版本是否为最新版本 3. 请记下操作流程与错误画面 ,并转交代理(经销）商反映给原厂。	-	H
8105	下载的项目程序内容有误 :下载的程序语法错误	1. 重新编译后再下载程序 2. 请确认通讯线是否确实连接牢固 3. 请将项目全部储存 ,并整理压缩成一个档案 ,然后交由代理（经销）商反映给原厂	-	H
8106	下载的项目程序内容有误 :执行码超过限制长度	1. 重新编译后再下载程序 2. 请将项目全部储存 ,并整理压缩成一个档案 ,然后交由代理（经销）商反映给原厂	-	H
8107	下载的项目程序内容有误 :原始码超过限制长度	1. 重新编译后再下载程序 2. 请将项目全部储存 ,并整理压缩成一个档案 ,然后交由代理（经销）商反映给原厂	-	H
8000 ~ 8FFF	此区间内之错误代码皆为编辑软件与 PLC 间专属的错误码	1. 请与代理商人员洽询软件与韧体最新版本 2. 请记下操作流程与错误画面 ,并转交代理(经销）商反映给原厂。	-	H

7.3 其它模块故障排除

•模块简介与错误代码类型说明

AS 系统可以安装数字、模拟、温度、定位、高速计数、称重及网络模块，详细规格请参阅各模块使用手册。模块错误代码主要分为两大类，一类为错误型之代码，当发生此类错误时，主机会跟随模块进行停止运行的动作；另一类为警告型之代码，当发生此类警告信息时，主机将持续运行。详细代码与故障排除方式表列如下：

7.3.1 模拟（AD / DA / XA）及温度（RTD / TC）模块故障排除

7.3.1.1 ERROR 灯常亮

下表错误代码须由模块的硬件组态（HWCONFIG）设定为“报警“错误信息显示，则模块才会以错误信息回报，否则将以一般警告信息回报。

错误代码	错误说明	排除方式
16#1605	模块内部硬件错误	退回原厂检修
16#1607	模块外部电源错误	检查电源
16#1608	校正值或 CJC 补偿异常	请联络原厂

7.3.1.2 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒）

为有利于 PLC 运行（RUN）时不受模块的错误信息影响整个系统，因此模块默认下表这些错误代码为警告类信息，若用户期望模块发生下表前 4 项错误时，PLC 须立即停止（STOP），则请由模块的硬件组态内报警设定勾选为”报警”信息（默认为”警告”）。

错误代码	错误说明	排除方式
16#1801	模块外部电源错误	检查电源
16#1802	模块硬件错误	退回原厂检修
16#1804	校正值异常	请联络原厂
16#1807	CJC 温度错误	退回原厂检修
16#1808	模拟（温度）输入通道 1 超出范围值	检查通道 1 输入信号
16#1809	模拟（温度）输入通道 2 超出范围值	检查通道 2 输入信号
16#180A	模拟（温度）输入通道 3 超出范围值	检查通道 3 输入信号
16#180B	模拟（温度）输入通道 4 超出范围值	检查通道 4 输入信号
16#180C	模拟（温度）输入通道 5 超出范围值	检查通道 5 输入信号
16#180D	模拟（温度）输入通道 6 超出范围值	检查通道 6 输入信号
16#180E	模拟（温度）输入通道 7 超出范围值	检查通道 7 输入信号
16#180F	模拟（温度）输入通道 8 超出范围值	检查通道 8 输入信号
-	上电时未收到主机检测命令	检查主机与模块连接是否确实或重新组合

7.3.2 AS02/04PU 定位模块故障排除

7.3.2.1 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒）

错误代码	错误说明	排除方式
16#1802	模块硬件错误	退回原厂检修

7.3.3 AS02HC 高速计数模块故障排除

7.3.3.1 ERROR 灯常亮

错误代码	错误说明	排除方式
16#1605	计数值停电保持异常（严重错误）	计数值记录已遗失，请将模块断电重启（重启后，错误代码将自行清除）。 若一再出现此信息，请联络原厂。
16#1606	模块设定值停电保持异常（严重错误）	模块设定已遗失并恢复出厂设定，模块断电重启或重新下载 HWCONFIG 可清除错误代码。 若一再出现此信息，请联络原厂。
16#1607	模块设定值配置内容错误（严重错误）	检查 HWCONFIG 配置内容并重新下载。 若一再出现此信息，请联络原厂。

7.3.3.2 ERROR 灯闪烁（亮 0.5 秒、暗 0.5 秒）

为有利于 PLC 运行（RUN）时不受模块的错误信息影响整个系统，因此模块默认下表这些错误代码为警告类信息。

错误代码	错误说明	排除方式
16#1800	CH1 计数器溢位 / 借位	检查计数值，若不须此报警请由 HWCONFIG 的报警设定页面中关闭。 此错误状态可由下述任一方式清除：清除计数器（Reset）、预置计数器（Preset）、重启模块、重启 DHCCNT 指令。
16#1801	CH2 计数器溢位 / 借位	
16#1802	CH1 线性计数超出设定上下限	检查通道一 / 通道二计数值。 硬件内部计数值依然持续计数，当内部硬件计数值回到最大计数值至最小计数值范围内，错误代码将自行清除。
16#1803	CH2 线性计数超出设定上下限	
16#1804	CH1 SSI 编码器位置变化量超出限制	检查是否受干扰、检查最大位移量限制设定值是否与实际操作转速匹配；当下一笔读取位置值回到正常变化量范围内，错误代码将自行清除。
16#1805	CH2 SSI 编码器位置变化量超出限制	

错误代码	错误说明	排除方式
16#1806	CH1 SSI 通讯异常	检查 DHCCNT 计数器状态，若为同位检查错误，检查是否有干扰存在，以及数据格式是否正确设定；若为 SSI 通讯异常，检查配线方式是否有断线情形、编码器供电是否正常以及数据格式设定是否正确。
16#1807	CH2 SSI 通讯异常	
16#1808	CH1 SSI 绝对位置跨越零点	检查 SSI 绝对型编码器操作范围，若不须此报警请由 HWCONFIG 的报警设定页面中关闭。 此错误状态可由下述任一方式清除：清除计数器（Reset）、预置计数器（Preset）、重启模块、重启 DHCCNT 指令。
16#1809	CH2 SSI 绝对位置跨越零点	

7.3.4 AS02LC 称重模块故障排除

7.3.4.1 ERROR 灯常亮

下表错误代码须由模块的硬件组态（HWCONFIG）设定为“报警“错误信息显示，则模块才会以错误信息回报，否则将以一般警告信息回报。

错误代码	错误说明	排除方式
16#1605	模块内部硬件（含驱动板）错误	退回原厂检修
16#1607	模块外部电源错误	检查电源

7.3.4.2 ERROR 灯闪烁（亮 0.2 秒、暗 0.2 秒）

为有利于 PLC 运行（RUN）时不受模块的错误信息影响整个系统，因此模块默认下表这些错误代码为警告类信息，若用户期望模块发生前 3 项错误时，PLC 立即停止（STOP），则请由模块的硬件组态内报警设定勾选为”报警”信息（默认为”警告”）。

错误代码	错误说明	排除方式
16#1801	模块外部电源错误	检查电源
16#1802	模块硬件错误	退回原厂检修
16#1807	模块驱动板错误	检查端子线是否受到干扰或短路（如 EXC+与 EXC-短路），如排除上述问题之后，还是持续出现此错误，请退回原厂检修
16#1808	通道 1 输入超出量测范围或 SEN 电压错误	检查通道 1 输入信号与接线
16#1809	通道 1 超出重量上限	检查通道 1 输入重量值与最大重量值设定
16#180A	通道 1 调校错误	检查通道 1 调校重量值与调校步骤
16#180B	通道 2 输入超出量测范围或 SEN 电压错误	检查通道 2 输入信号与接线
16#180C	通道 2 超出重量上限	检查通道 2 输入重量值与最大重量值设定

错误代码	错误说明	排除方式
16#180D	通道 2 调校错误	检查通道 2 调校重量值与调校步骤
-	上电时未收到主机检测命令	检查主机与模块连接是否确实或重新组合

7.3.5 AS04SIL IO-Link 通讯模块故障排除

7.3.5.1 MS 灯闪烁（亮 0.2 秒、暗 0.2 秒）

此表为 SIL 模块安装于主机右侧或是 RTU 右侧，可能发生之错误代码。

错误代码	错误说明	排除方式
16#1605	模块本身硬件错误	更换 AS04SIL-A 模块或联络代理商送回原厂维修
16#1606	24VDC 电源曾经不稳定，且发生过低电压状况超过 10ms 之后又恢复	检查一次电源（24V）

7.3.5.2 NS 灯恒亮

此表为 SIL 模块安装于主机右侧或是 RTU 右侧，可能发生之警告代码。

警告代码	警告说明	排除方式
16#1802	二次电源未上电	检查二次电源

7.3.5.3 NS 灯闪烁（亮 1 秒、暗 1 秒）

此表为 SIL 模块安装于主机右侧或是 RTU 右侧，可能发生之警告代码。

警告代码	警告说明	排除方式
16#1800	发生来自 IO-Link 主站的诊断事件	根据发生的 IO-Link 事件代码，参照对应的建议排除方式
16#1801	发生来自 IO-Link 装置的诊断事件	根据发生的 IO-Link 事件代码，参照对应的建议排除方式
16#1803	IO-Link 装置映像表下载错误	软件重新下载配置
16#1804	过程数据参数集切换失败	检查连接装置与软件配置是否相符
16#1805	通讯端口 1 IO-Link 联机发生异常	1. 外部电源，断电 3 秒后再次上电 2. 重新下载配置
16#1806	通讯端口 2 IO-Link 联机发生异常	
16#1807	通讯端口 3 IO-Link 联机发生异常	
16#1808	通讯端口 4 IO-Link 联机发生异常	
16#1809	装置扫描异常强制终止扫描	1. 外部电源，断电 3 秒后再次上电 2. 再次执行「扫描所有装置」

IO-Link 事件代码如下，记录于模块常态交换区通讯端口 1 – 4 装置状态。事件来源为 IO-Link 装置时，请同时参考 IO-Link 装置操作手册。

IO-Link 事件代码	类别			事件说明	建议排除方式	事件来源	
	警告	错误	通知			IO-Link 主站	IO-Link 装置
16#4000		V		装置的温度过高	降低负载		V
16#4210	V			装置的温度过高	移除热源		V

IO-Link 事件代码	类别			事件说明	建议排除方式	事件来源	
	警告	错误	通知			IO-Link 主站	IO-Link 装置
16#5101		V		装置的熔断器故障	更换熔断器		V
16#5110	V			电源电压过高	确认耐受性		V
16#5111	V			电源电压过低	确认耐受性		V
16#6320		V		参数错误	核对装置规格		V
16#6321		V		参数遗失	核对装置规格		V
16#7710		V		装置短路	检查配置		V
16#8C10	V			超出过程变量范围	确认数据稳定		V
16#8C20		V		测量范围超限	确认应用规格		V
16#8C30	V			低于过程变量范围	确认数据稳定		V
16#8CA0	V			未连接任何 IO-Link 装置	检查设备	V	
16#8CA1	V			装置支持的 IO-Link 协议版本与配置不符	用相符的 IODD 文件版本重新配置	V	
16#8CA2	V			连接装置与配置不符	核对软件配置与实际装置	V	
16#8CA3				保留		V	
16#8CA4		V		IO-Link 装置上的过程电缆短路	检查安装	V	
16#8CA5	V			主站的温度超过 135°C	移除热源	V	
16#8CA6		V		主站的温度超过 160°C	移除热源并降低负载	V	
16#8CA7	V			装置的电源电压过低警告 L+（<18V）	检查二次电源	V	
16#8CA8		V		装置的电源电压过低错误 L+（<9V）	检查二次电源	V	
16#8CA9	V			装置 ID 不合法	核对装置规格	V	
16#8CAA	V			超出 IO-Link 装置的过程数据长度	核对装置规格	V	
16#8CAB	V			超出过程数据长度	重新扫描装置并下载配置	V	
16#8CAC		V		数据备份储存错误	联系原厂或代理商支持	V	
16#FF21			V	新联机装置		V	
16#FF22			V	装置断线	检查安装	V	

IO-Link 事件代码	类别			事件说明	建议排除方式	事件来源	
	警告	错误	通知			IO-Link 主站	IO-Link 装置
16#FF23			V	备份数据的 ID 与实际连接装置不符	将 DS 状态设定为『关闭』后，再设定为『备份/还原』，依照实际配置重新进行备份	V	
16#FF24			V	数据备份空间不足	核对装置规格	V	
16#FF25			V	数据备份存取被拒绝	核对装置规格	V	

7.3.6 AS00SCM 为串行通讯模块故障排除

7.3.6.1 ERROR 灯常亮

此表为 SCM 模块安装在主机右侧，当通讯模块时可能发生的错误代码。

错误代码	错误说明	排除方式
16#1605	模块本身硬件错误	1. 检查模块与主机之间是否正确安装 2. 更换 AS00SCM 模块或联络代理商送回原厂维修
16#1606	功能卡配置错误	1. 检查通讯卡插槽是否正确连接功能卡 2. 更换功能卡或联络代理商送回原厂维修 3. 检查硬件配置表（HWCONFIG）是否和功能卡实体相同 4. 更换 AS00SCM 模块或联络代理商送回原厂维修

7.3.6.2 ERROR 灯一般闪烁（亮 0.5 秒、暗 0.5 秒）

此表为 SCM 模块安装在主机右侧，当通讯模块时可能发生的警告代码

错误代码	错误说明	排除方式
16#1802	参数警告	重新下载模块参数
16#1803	通讯逾时警告	1. 检查模块连接器是否确实连接 2. 站号及通讯格式是否设定正确 3. 确认功能卡连接状态是否正常
16#1804	UD Link 配置警告	1. 检查 UD Link 是否设定正确 2. 检查 PLC 程序触发内容是否正确

下表为 SCM 模块当通讯模块时，透过 SCMSoft 显示的错误代码。这些错误状况将不会产生 ERROR 灯号闪烁，也不会回传错误码给主机。

错误代码	错误说明	排除方式
16#0107	功能卡 1 和手动设定不一致	检查 CARD1 硬件配置表是否和功能卡实体相同
16#0108	功能卡 2 和手动设定不一致	检查 CARD2 硬件配置表是否和功能卡实体相同
16#0201	模块参数错误	检查参数值并重新下载模块参数

错误代码	错误说明	排除方式
16#0301	功能卡 1 通讯逾时	1. 站号及通讯格式是否设定正确 2. 确认功能卡的通讯线连接状态是否正常
16#0302	功能卡 2 通讯逾时	1. 站号及通讯格式是否设定正确 2. 确认功能卡的通讯线连接状态是否正常
16#0400	功能卡 1 UD Link Group ID 无效	1. 检查 UD Link 是否设定正确 2. 检查 PLC 程序触发内容是否正确
16#0401	功能卡 2 UD Link Group ID 无效	1. 检查 UD Link 是否设定正确 2. 检查 PLC 程序触发内容是否正确
16#0402	功能卡 1 UD Link Command 无效	1. 检查 UD Link 是否设定正确 2. 检查 PLC 程序触发内容是否正确
16#0403	功能卡 2 UD Link Command 无效	1. 检查 UD Link 是否设定正确 2. 检查 PLC 程序触发内容是否正确

7.3.7 AS00SCM 为 AS 远程模块故障排除

此处远程模块显示的错误信息，对于 AS 主机而言，皆为警告类别之错误状态，主机只会慢闪 ERROR 灯，但不会影响主机正常程序运行状况，因此若用户有必要处理远程模块的错误时，请在 PLC 程序内搭配 SM30 标志进行判断与设计处理流程。

7.3.7.1 ERROR 灯常亮

此表为错误类型的错误代码表。

错误代码	错误说明	排除方式
16#1301	模块本身硬件错误	1. 检查模块与主机之间是否正确安装 2. 更换 AS00SCM 模块或联络代理商送回原厂维修
16#1302	功能卡配置错误	1. 检查通讯卡插槽 2 是否正确连接 AS-FCOPM 卡 2. 更换功能卡或联络代理商送回原厂维修 3. 检查硬件配置表（HWCONFIG）是否和功能卡实体相同 4. 更换 AS00SCM 模块或联络代理商送回原厂维修
16#1304	远程模块右侧连接超出 8 台限制	确认远程模块右侧扩展 IO 模块不得超过 8 台

7.3.7.2 ERROR 灯慢速闪烁（亮 1 秒、暗 1 秒）

此表为警告类型之代码表。

错误代码	错误说明	排除方式
16#1506	远程模块进入停机状态	此状态表示远程模块停机条件曾经被触发，且目前状况已排除，请将设备重新上电。停机条件包括：主站断线、主站断线后恢复联机、IO 模块发生警报以及 IO 模块断线，详细设定请参考 AS 系列模块手册第 9.4.3 节说明。

7.3.7.3 ERROR 灯一般闪烁（亮 0.5 秒、暗 0.5 秒）

此表为警告类型之代码表。

错误代码	错误说明	排除方式
16#1500	远程模块通讯逾时警告	检查远程模块的通讯线是否确实连接
16#1502	参数警告	重新下载模块参数
16#1503	扩展模块通讯逾时警告	检查模块连接器是否确实连接
16#1505	实际扩展模块配置与设定配置不符	检查实际模块配置是否与设定配置不同
16#1604	扩展 IO 模块通讯逾时	此代码将显示于扩展 IO 模块字段，请将远程模块断电，确认扩展 IO 模块与远程模块有正确连接，再重新上电。

7.3.7.4 ERROR 灯快速闪烁（亮 0.2 秒、暗 0.2 秒）

此状态表示供应远程模块之 24VDC 电源供应不足，请检查供应电源，若供应电源确认正常，则请先移除扩展模块，再次确认是否为 SCM 远程模块损坏。此表为警告类型的错误代码。

错误代码	错误说明	排除方式
16#1303	24VDC 电源曾经不稳定，且发生过低电压状况超过 10ms 之后又恢复	检查 24V 电源

7.3.8 AS01DNET 为通讯模块故障排除

请参考第 7.4.9 节与第 7.4.10 节说明。

7.4 模块错误代码对应灯号及状态说明

7.4.1 CPU 模块错误代码对应灯号及状态

- **字段简介**

a. 错误代码：该错误发生时系统所产生的错误代码。

b. 说明：该错误的说明。

c. CPU 状态：该错误发生时，CPU 主机的状态变化。

- 停止：发生该错误时 CPU 停止运行。
- 持续：发生该错误时 CPU 持续运行。

d. 灯号状态：该错误发生时的主机灯号变化。

- ERROR：系统错误灯号。

- **各灯号说明**

模块名称	灯号	说明
CPU	Error LED	指示 CPU 的错误或警告状态，其灯号显示共有底下五种状态，前三种直亮、快速与一般显示方式为错误类别信息，当此类错误发生时，使用者须排除后，才可再重新让 CPU 运行；最后一种慢速显示为警告类型信息，用户不需立即处理，但最好在设备机台未运行状态下，进行排除的动作。 错误类： 常亮：系统运行过程中发生严重错误。 快速闪烁：电源不稳定或硬件组件自我检查错误。(亮 0.2 秒、暗 0.2 秒) 一般闪烁：程序错误或系统无法执行的错误发生。(亮 0.5 秒、暗 0.5 秒) 警告类： 慢速闪烁：警告信息发生，系统可持续运行。(亮 1 秒、暗 3 秒) 无：有警告信息发生但不影响运行，仅由编辑软件提示或 SM/SR 显示警告信息。

错误代码 (16#)	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
000A	扫描逾时	停止	V				
000C	PLC 程序毁损	停止			V		
000D	装置初始值与 PLC 程序未完整下载	停止			V		
0010	主机内部存储器检测错误	停止			V		
0011	PLC ID 错误	持续					V
0012	PLC 密码错误	持续					V
0020	CPU 执行速度未达最高速	停止			V		
0021	台达设备参数还原错误 (依 HWCONFIG 设定是否停止)	H			H		

错误代码（16#）	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
注：16#0021 依 HWCONFIG 设定，可以选择是否停止。若选择不停止，只会有 ERROR LOG 记录。若选择停止，CPU 会停止，ERROR 灯一般闪烁。							
0026	万年历无法计时（电池灯号闪烁）	持续					
0027	电池低电压（电池灯号常亮）	持续					
0028	万年历读写无回应	持續					V
002A	曾经发生过 24V 低电压又恢复	持续		V			
002D	PLC 密码错误次数达到上限	持续					V
002E	主机外部内存检测错误	停止			V		
002F	PLC 程序与系统记录比对不符	停止			V		
0040	功能卡 1 AS-F2AD 或是内建 AD 输入通道 1 数值转换超过范围	持续				V	
0041	功能卡 1 AS-F2AD 或内建 AD 输入通道 2 数值转换超过范围	持续				V	
0042	功能卡 2 AS-F2AD 输入通道 1 数值转换超过范围	持续				V	
0043	功能卡 2 AS-F2AD 输入通道 2 数值转换超过范围	持续				V	
0044	功能卡 1 AS-F2AD 或内建 AD 输入通道 1 电流输入 4mA~20mA 模式断线	持续				V	
0045	功能卡 1 AS-F2AD 或内建 AD 输入通道 2 电流输入 4mA~20mA 模式断线	持续				V	
0046	功能卡 2 AS-F2AD 输入通道 1 电流输入 4mA~20mA 模式断线	持续				V	
0047	功能卡 2 AS-F2AD 输入通道 2 电流输入 4mA~20mA 模式断线	持续				V	
0050	停电保持区 SM 储存区块异常	持续					V
0051	停电保持区 SR 寄存器异常	持续					V
0052	停电保持区 M 储存区块异常	持续					V
0054	停电保持区 C 寄存器异常	持续					V
0055	停电保持区 HC 寄存器异常	持续					V
0056	停电保持区 S 储存区块异常	持续					V
0059	停电保持区 D 寄存器异常	持续					V
005D	CPU 模块检测不到储存卡	持续					V
005E	储存卡的初始程序错误	持续					V
0063	数据写入储存卡的文件时有错误	持续					V

错误代码（16#）	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
0064	储存卡的文件无法被读取	持续					V
0070	功能卡与手动配置不符	停止			V		
0102	中断编号使用超过范围	停止			V		
0202	MC 指令超过有效范围	停止			V		
0302	MCR 指令超过有效范围	停止			V		
0D03	DHSCS 指令操作数使用不当	停止			V		
0E05	DCNT 指令操作数 HCXXX 使用不当	停止			V		
1000 ~ 10FF	系统性错误	停止			V		
1300 ~ 130F	远程模块发出错误信息	持续				V	
1402	实际 IO 模块不符合配置设定	停止			V		
140B	通讯模块超出 4 台限制	停止			V		
140C	定位模块超出 8 台限制	停止			V		
140D	所有扩展模块超出 32 台限制	停止			V		
140E	远程模块的扩展模块超出 8 台限制	停止			V		
1500	远程模块断线警告信息	持续				V	
1502 ~ 150F	远程模块发出警告信息	持续				V	
1600	扩展模块 ID 配置超出范围	停止			V		
1601	扩展模块无法配置 ID	停止			V		
1602	扩展模块 ID 配置重复	停止			V		
1603	扩展模块无法进入运行模式	停止			V		
1604	扩展模块通讯逾时	停止			V		
1605	扩展模块硬件错误	停止			V		
1606	通讯模块功能卡配置错误	停止			V		
1607	扩展模块外部电源错误	停止			V		
1608	扩展模块校正值或 CJC 补偿异常	停止			V		
1609 ~ 160F	保留（扩展模块错误代码）	停止			V		
1800 ~ 180F	扩展模块发出警告类信息	持续				V	

错误代码（16#）	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
1900 ~ 191C	CANopen/ECAT 专用通讯模式之从站发生心律/断线错误信息	持续				V	
1950	CANopen 专用通讯模式未初始化完成，指令禁止执行	持续					V
19B0	从站模式之心跳信息逾时	持续					V
19B1	从站模式之 PDO 数据长度不符	持续					V
19E1	从站返回的 PDO 数据长度与扫描列表中配置的不符 备注：错误代码 19E1~19E8，建议另外参考操作手册之 CANopen 通讯说明	持续					V
19E2	未接收到从站 PDO	持续					V
19E3	初次启动时的自动 SDO 下载失败	持续					V
19E4	PDO 参数配置失败	持续					V
19E5	关键参数不匹配	持续					V
19E6	网络中不存在此从站	持续					V
19E7	从站错误控制逾时	持续					V
19E8	主从站站号重复	持续					V
19F3	配置表错误	持续					V
19F4	CAN 通讯 BUS-OFF 状态	持续					V
19FB	发送寄存器超出最大容量	持续					V
19FC	接收寄存器超出最大容量	持续					V
2001	使用台达专用 CANopen 通讯指令时，未插 FCOMP 卡或不是在 ASDA-A2 专用模式	持续					V
2003	程序中使用的装置超过可用范围或不合法	持续					V
200A	无效的指令	停止			V		
200B	n 操作数或其它 K/H 操作数超出范围	持续					V
200C	部份指令不充许操作数发生重迭	持续					V
200D	BIN 转成 BCD 时发生错误	持续					V
200E	字符串没有 00 当做结尾	持续					V
2012	除法运算错误	持续					V
2013	浮点数格式错误，超出可转换范围	持续					V

错误代码（16#）	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
2014	TKON/TKOFF 指令所指定的 TASK 编号错误或超出范围	持续					V
2017	BREAK 指令写在 FOR-NEXT 之外	持续					V
2027	无此定位规划表编号或是定位规划表模式不对	持续					V
2028	高速输出指令正在输出中，不可重复输出	持续					V
2030	CSFO 指令之输出频率超出最高限制频率	持續					V
2031	CSFO 指令于输入频率为 0 时，该轴输出超过 2 秒还未输出完已换算的输出个数	持續					V
6004	以太网络的 IP 地址过滤设置错误	持续					V
600D	RJ45 接口未连接	持续					V
6010	MODBUS TCP 联机数超出范围	持续			V		
6011	EtherNet/IP 联机数超出范围	持续			V		
6012	网络上有重复 IP 设备	持续					V
6100	E-mail 联机忙碌	持续					V
6103	E-mail 附件中的装置地址设定错误	持续					V
6104	E-mail 附件不存在	持续					V
6105	E-mail 附件超过容量	持续					V
6106	SMTP 服务器地址错误	持续					V
6107	SMTP 服务器逾时	持续					V
6108	寄件服务器验证错误	持续					V
6200	TCP 通讯接口（Socket）功能的远程 IP 地址不合法	持续					V
6201	TCP 通讯接口（Socket）功能的本地端口不合法	持续					V
6202	TCP 通讯接口（Socket）功能的远程端口不合法	持续					V
6203	TCP 通讯接口（Socket）功能的传送数据地址不合法	持续					V
6206	TCP 通讯接口（Socket）功能的接收数据地址不合法	持续					V
6208	TCP 通讯接口（Socket）功能的接收数据装置超出范围	持续					V
6209	UDP 通讯接口（Socket）功能的远程 IP 地址不合法	持续					V
620A	UDP 通讯接口（Socket）功能的本地端口不合法	持续					V
620C	UDP 通讯接口（Socket）功能的传送数据地址不合法	持续					V
620F	UDP 通讯接口（Socket）功能的接收数据地址不合法	持续					V
6210	UDP 通讯接口（Socket）功能的接收数据长度不合法	持续					V
6212	远程装置响应逾时	持续					V
6213	接收数据超过限制	持续					V

错误代码（16#）	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
6214	远程装置拒绝联机	持续					V
6215	目前通讯接口（Socket）未开启	持续					V
6217	目前通讯接口（Socket）已开启	持续					V
6218	目前通讯接口（Socket）已传送	持续					V
6219	目前通讯接口（Socket）已接收	持续					V
621A	目前通讯接口（Socket）已关闭	持续					V
7011	COM1 通讯功能码错误	持续					V
7012	COM1 装置组件通讯地址错误	持续					V
7013	COM1 需求组件已超出限制范围	持续					V
7014	COM1 需求通讯长度已超出限制范围	持续					V
7017	COM1 通讯数据串的检查码错误	持续					V
7021	COM2 通讯功能码错误	持续					V
7022	COM2 装置组件通讯地址错误	持续					V
7023	COM2 需求组件已超出限制范围	持续					V
7024	COM2 需求通讯长度已超出限制范围	持续					V
7027	COM2 通讯数据串之检查码错误	持续					V
7031	Ethernet 通讯功能码错误	持续					V
7032	Ethernet 装置组件通讯地址错误	持续					V
7033	Ethernet 需求组件已超出限制范围	持续					V
7034	Ethernet 需求通讯长度已超出限制范围	持续					V
7037	Ethernet 通讯数据串的检查码错误	持续					V
7041	USB 通讯功能码错误	持续					V
7042	USB 装置组件通讯地址错误	持续					V
7043	USB 需求组件已超出限制范围	持续					V
7044	USB 需求通讯长度已超出限制范围	持续					V
7047	USB 通讯数据串的检查码错误	持续					V
70B1	功能卡 1 通讯功能码错误	持续					V
70B2	功能卡 1 装置组件通讯地址错误	持续					V
70B3	功能卡 1 需求组件已超出限制范围	持续					V
70B4	功能卡 1 需求通讯长度已超出限制范围	持续					V
70B7	功能卡 1 通讯数据串的检查码错误	持续					V
70C1	功能卡 2 通讯功能码错误	持续					V
70C2	功能卡 2 装置组件通讯地址错误	持续					V

错误代码（16#）	错误说明	CPU 状态	ERROR 灯号状态				
			常亮	快闪	一般	慢闪	无
70C3	功能卡 2 需求组件已超出限制范围	持续					V
70C4	功能卡 2 需求通讯长度已超出限制范围	持续					V
70C7	功能卡 2 通讯数据串的检查码错误	持续					V
7203	无效通讯功能码	持续					V
8105	下载的项目程序内容有误：下载的程序语法错误	持续					V
8106	下载的项目程序内容有误：执行码超过限制长度	持续					V
8107	下载的项目程序内容有误：原始码超过限制长度	持续					V
8000 ~ 8FFF	此区间内之错误代码皆为编辑软件与 PLC 间专属的错误码	持续					V
C000 ~ CFFF	PLC 程序有语法不合法错误	停止			V		

7.4.2 模拟与温度模块错误代码对应灯号

错误代码	说明	灯号状态	
		A → D / D → A / A ↔ D	ERROR
16#1605	模块内部硬件错误	OFF	常亮
16#1607	模块外部电源错误	OFF	常亮
16#1608	校正值或 CJC 补偿异常	OFF	常亮
16#1801*1	模块外部电源错误	OFF	闪烁
16#1802*1	模块硬件错误	OFF	闪烁
16#1804*1	内部错误，出厂校正异常	RUN：闪烁 STOP：OFF	闪烁
16#1807*1	CJC 温度错误	OFF	闪烁
16#1808	模拟（温度）输入通道 1 超出范围值	RUN：闪烁 STOP：OFF	闪烁
16#1809	模拟（温度）输入通道 2 超出范围值		
16#180A	模拟（温度）输入通道 3 超出范围值		
16#180B	模拟（温度）输入通道 4 超出范围值		
16#180C	模拟（温度）输入通道 5 超出范围值		
16#180D	模拟（温度）输入通道 6 超出范围值		
16#180E	模拟（温度）输入通道 7 超出范围值		
16#180F	模拟（温度）输入通道 8 超出范围值		

错误代码	说明	灯号状态	
		A → D / D → A / A ↔D	ERROR
-	上电时未收到主机检测命令	OFF	闪烁一次或两次，两秒后重复闪烁

*1：此四个错误码默认为警告类信息，若需更改为错误类信息，请由模块硬件组态进行修改。

7.4.3 AS02/04PU 定位模块错误代码对应灯号

错误代码	说明	灯号状态	
		A ↔D	ERROR
16#1802	模块内部硬件错误	OFF	闪烁

7.4.4 AS02HC 高速计数模块错误代码对应灯号

错误代码	说明	ERROR 灯号状态	
		常亮	一般闪烁
16#1605	计数值停电保持异常	V	
16#1606	模块设定值停电保持异常	V	
16#1607	模块设定值配置内容错误	V	
16#1800	CH1 计数器溢位 / 借位		V
16#1801	CH2 计数器溢位 / 借位		V
16#1802	CH1 线性计数超出设定上下限		V
16#1803	CH2 线性计数超出设定上下限		V
16#1804	CH1 SSI 编码器位置变化量超出限制		V
16#1805	CH2 SSI 编码器位置变化量超出限制		V
16#1806	CH1 SSI 通讯异常		V
16#1807	CH2 SSI 通讯异常		V
16#1808	CH1 SSI 绝对位置跨越零点		V
16#1809	CH2 SSI 绝对位置跨越零点		V

7.4.5 AS02LC 称重模块错误代码对应灯号

错误代码	说明	灯号状态	
		A → D	ERROR
16#1605	模块内部硬件（含驱动板）错误	OFF	常亮

错误代码	说明	灯号状态	
		A → D	ERROR
16#1607	模块外部电源错误	OFF	常亮
16#1801*1	模块外部电源错误	OFF	闪烁
16#1802*1	模块内部硬件错误	OFF	闪烁
16#1807*1	模块驱动板错误	OFF	闪烁
16#1808	通道 1 输入超出量测范围或 SEN 电压错误	RUN：闪烁 STOP：OFF	闪烁
16#1809	通道 1 超出重量上限		
16#180A	通道 1 调校错误		
16#180B	通道 2 输入超出量测范围或 SEN 电压错误		
16#180C	通道 2 超出重量上限		
16#180D	通道 2 调校错误		
-	上电时未收到主机检测命令	OFF	闪烁一次或两次，两秒后重复闪烁

*1：此三个错误码默认为警告类信息，若需更改为错误类信息，请由模块硬件组态进行修改。

7.4.6 AS04SIL IO-Link 通讯模块错误代码对应灯号

错误代码	说明	MS 灯号状态	
		常亮	快闪
16#1605	模块本身硬件错误		V
16#1606	24VDC 电源曾经不稳定，且发生过低电压状况超过 10ms 之后又恢复		V

错误代码	说明	NS 灯号状态	
		常亮	一般闪烁
16#1800	发生来自 IO-Link 主站的诊断事件		V
16#1801	发生来自 IO-Link 装置的诊断事件		V
16#1802	二次电源未上电	V	
16#1803	IO-Link 装置映像表下载错误		V
16#1804	过程数据参数集切换失败		V
16#1805	通讯端口 1 IO-Link 联机发生异常		V
16#1806	通讯端口 2 IO-Link 联机发生异常		V

错误代码	说明	NS 灯号状态	
		常亮	一般闪烁
16#1807	通讯端口 3 IO-Link 联机发生异常		V
16#1808	通讯端口 4 IO-Link 联机发生异常		V
16#1809	装置扫描异常强制终止扫描		V

7.4.7 AS00SCM 为串行通讯模块错误代码对应灯号

错误代码	说明	ERROR 灯号状态	
		常亮	一般闪烁
16#1605	模块本身硬件错误	V	
16#1606	功能卡配置错误	V	
16#1802	模块参数错误		V
16#1803	通讯逾时		V
16#1804	UD Link 错误		V

7.4.8 AS00SCM 为 AS 远程模块错误代码对应灯号

错误代码	说明	ERROR 灯号状态			
		常亮	一般闪烁	慢闪	快闪
16#1301	模块本身硬件错误	V			
16#1302	功能卡配置错误	V			
16#1303	24VDC 电源曾经不稳定，且发生过低电压状况超过 10ms 之后又恢复				V
16#1304	远程模块右侧连接超出 8 台限制	V			
16#1500	远程模块通讯逾时警告		V		
16#1502	参数警告		V		
16#1503	扩展模块通讯逾时警告		V		
16#1505	实际扩展模块与配置不符		V		
16#1506	远程模块进入停机状态			V	
16#1604	扩展模块通讯逾时		V		

7.4.9 AS01DNET 为主从站模式时的数字显示器显示说明

代码	显示说明	处理方法
0~63	模块的节点地址（正常工作时）	无需处理
80	模块处于停止状态	将 PLC 主机拨至 RUN 状态，进行 IO 数据交换
F0	AS01DNET 扫描模块的站号与	1. 确认模块的节点地址是唯一的

代码	显示说明	处理方法
	其它节点重复，或超出范围	2. 将模块重新上电
F1	扫描列表内没有配置从站	配置扫描列表，配置完成后下载至模块
F2	工作电源电压过低	检查模块以及 PLC 主机的工作电源是否正常
F3	模块进入测试模式	将功能开关的 IN1 切换为 OFF 状态，并对模块重新上电
F4	BUS-OFF	1. 检查网络通讯线是否正常、屏蔽线是否接地 2. 确认所有网络上的节点设备波特率是否一致 3. 检查网络的首尾两端是否都接有 121Ω 的终端电阻 4. 将扫描模块重新上电
F5	没有检测到网络电源	1. 检查网络电缆是否正常 2. 确认网络电源正常
F6	内部错误，内部存储器检测出错	将模块重新上电，如果错误依然存在，退回工厂进行修复
F8	内部错误，工厂制造流程出错	将模块重新上电，如果错误依然存在，退回工厂进行修复
F9	内部错误，配置数据存储器访问出错	将模块重新上电，如果错误依然存在，退回工厂进行修复
FA	配置数据无效	i. 正确配置网络后重新下载至模块 i. 检查扫描列表内从站节点地址是否与模块节点地址重复
E0	从站返回的识别参数与配置数据不一致	1. 确认总在线从站的节点站号是否变化 2. 确认总在线的节点设备是否被更换 3. 重新对网络进行配置
E1	从站返回的 IO 数据长度与扫描列表中配置的不一致	重新配置从站的 IO 数据长度，并下载至模块，运行 PLC 主机
E2	主站模式时，扫描列表中配置的从站断线或不存在 从站模式时，AS01DNET 模块（从站）与主站的 IO 连接中断	1. 检查从站的节点站号是否变化 2. 检查网络通讯电缆是否正常，如断路、松动等 3. 检查总线通讯电缆长度是否超过最远传输距离。超过最远传输距离后，将不能保证系统稳定
E3	模块发送数据失败	1. 检查模块与网络连接是否正常 2. 确认模块波特率与网络上其它节点的波特率设定一致
E4	从站传送的 IO 分段数据序列有错误	检查从站是否工作正常
E5	模块与从站建立连接时，从站返回错误信息	检查从站是否工作正常
E6	从站返回的 IO 数据长度超出扫描列表中配置的长度	确认从站的 IO 数据长度与扫描列表中配置的 IO 数据长度一致
E7	模块正在进行重复地址检测	若长时间显示该代码，请按如下方法排除错误： 1. 保证网络中有至少两个正常工作的节点 2. 检查网络的首尾两端是否都接有 121Ω 的终端电阻 3. 确认网络上的节点设备波特率是否一致 4. 检查网络通讯电缆是否正常，如断路、松动等 5. 检查总线通讯电缆长度是否超过最远传输距离。超过最远传输距离后，将不能保证系统稳定 6. 检查网络通讯电缆的屏蔽线是否接地 7. 将 AS01DNET 扫描模块重新上电

7.4.10 AS01DNET 为 RTU 模式时的数字显示器显示说明

代码	显示说明	处理方法
0~63	扫描模块的节点站号（正常工作时）	无需处理
F0	站号与其它节点重复，或超出范围	1. 确认 AS01DNET（RTU）节点站号在 DeviceNet 网络中是唯一的并且在 0~63 之间 2. 更改节点站号后将其重新上电
F1	DeviceNet Builder 软件中没有 I/O 模块配置到 AS01DNET（RTU）内	在 DeviceNet Builder 软件中增加 I/O 模块到 AS01DNET（RTU）内，配置完成后下载至 AS01DNET（RTU）
F2	AS01DNET（RTU）模块的工作电压过低	检查 AS01DNET（RTU）模块的工作电源是否正常
F3	AS01DNET（RTU）模块进入测试模式	将 AS01DNET（RTU）重新上电
F4	AS01DNET（RTU）模块进入 Bus-Off 状态	1. 检查网络通讯电缆是否正常、屏蔽线是否接地 2. 确认所有网络上的节点设备波特率是否一致 3. 检查网络的首尾两端是否都接有 121Ω 的终端电阻 4. 将扫描模块重新上电
F5	AS01DNET（RTU）模块没有网络电源	1. 检查网络电缆是否正常 2. 检测网络电源正常（AS01DNET（RTU） V+（红色）和 V-（黑色）之间需外接直流 24V 网络电源）
F6	内部错误，AS01DNET（RTU）模块的内部存储单元出错	将 AS01DNET（RTU）重新上电，如果错误依然存在，请退回工厂进行修复
F7	内部错误，AS01DNET（RTU）模块的数据交换单元出错	将 AS01DNET（RTU）重新上电，如果错误依然存在，请退回工厂进行修复
F8	出厂制造错误	将 AS01DNET（RTU）重新上电，如果错误依然存在，请退回工厂进行修复
F9	内部错误，AS01DNET（RTU）模块 FLASH 存取出错	将 AS01DNET（RTU）重新上电，如果错误依然存在，请退回工厂进行修复
E4	模块发生错误	检查 AS01DNET（RTU）右侧连接模块是否报错、不存在或者现在模块与软件配置的模块不一致，及增加没有配置的模块。

代码	显示说明	处理方法
E7	AS01DNET（RTU）重复地址检测	若长时间显示该代码，请按如下方法排除错误： 1. 保证网络中有至少两个正常工作的节点 2. 检查网络的首尾两端是否都接有 121Ω 的终端电阻 3. 确认网络上的节点设备波特率是否一致 4. 检查网络通讯电缆是否正常，如断路、松动等 5. 检查总线通讯电缆长度是否超过最远传输距离。超过最远传输距离后，将不能保证系统稳定 6. 检查网络通讯电缆的屏蔽线是否接地 7. 将 AS01DNET（RTU）模块重新上电
E9	AS01DNET（RTU）连接 I/O 模块台数超出 8 台限制	检查 AS01DNET（RTU）连接 I/O 模块台数是否超出 8 台
80	AS01DNET（RTU）处于停止状态（STOP）	1. 和 DeviceNet 主站相连的 PLC 的 RUN/STOP 开关拨至 RUN 2. 检查 AS01DNET（RTU）控制字的值是否为 1，详细请参考 AS 系列模块手册第 11.5.4.3.4 节的说明
83	正在下载软件中的 AS01DNET（RTU）配置	等待软件中 AS01DNET（RTU）配置数据下载完成