PLC 编程手册

(LX6V系列) Programming manual

福州富昌维控电子科技有限公司

FUZHOU FUCHANG WECON E AND T LTD

内容提要

这是一本全面介绍 Wecon PLC Editor2 基本功能及实际运用的书。本书完全针对零基础读者而开发，是入门级读者快速而全面掌握维控 PLC 和 Wecon PLC Editor2 的必备参考书。

本书从维控 PLC 的基本产品、Wecon PLC Editor2 的基本概念和基本操作入手，结合大量的案例和图形解析，全面而深入地阐述了 Wecon PLC Editor2 软件的使用，以及 PLC 程序的编写，向读者展示了如何运用 Wecon PLC Editor2 软件编写程序，灵活的运用于自动化控制设备。

本书非常适用于作为初、中级读者的入门及提高参考书，尤其是零基础读者。同时，本书所有解释权归福州富昌维控电子科技有限公司所有。

安 全 注 意 事 项

在进行微型可编程控制的安装、运转、保养检修之前，请务必熟读此手册和其他相关手册，确保正确使用。请在熟练掌握操作方法、安全信息以及全部注意事项之后再进行使用。

（1）设计注意事项

外部电源发生异常、可编程控制器发生故障时，为使整个系统安全运行，请务必在可编程控制器的外部设置安全电路。

①务必在可编程控制器的外部电路中设置紧急制动电路、保护电路、正反转电路等相反操作的互锁电路和防止机器损坏的定位上限、下限的互锁电路等。

②可编程控制器 CPU 通过自诊断检测出 WDT 错误等异常情况时，全部输出被关断。另外当可编程控制器 CPU 不能检测输入输出控制部分等的异常情况时，不能控制输出。这时为使机器能安全运转，请设计外部电路和机构。

③由于输出单元的继电器、晶体管故障，会无法控制输出为 ON 或 OFF 的状态。为使机器能安全运行，对于与重大事故相关的输出信号，请设计外部电路和机构。

（2）安装注意事项

①请在手册中记载的一般规格环境中使用。

请勿在下列场所使用：有灰尘、油烟、导电性尘埃、腐蚀性气体、可燃性气体的场所；暴露于高温、结露、风雨的场所；有振动、冲击的场所。电击、火灾、误操作也会造成产品损坏。

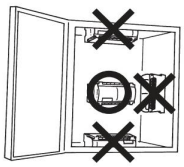
②在进行螺丝孔加工和接线时，不要使铁屑或电线头落入可编程控制器的通风窗口内。可能引起火灾、故障、误操作。

③请把连接电缆、显示模块准确插入规定插口中。接触不良有可能引起误操作。

附记

●为防止温度上升，切勿在底部、顶部及竖直方向安装。
请务必按右图所示在壁面上水平安装。

●主机和其他设备或构造物之间请留出 50mm 以上空间。
尽量避开高压线、高压设备、动力设备。



（3）布线注意事项

可编程控制器的信号输入和输出线不能在同一电缆上通过。

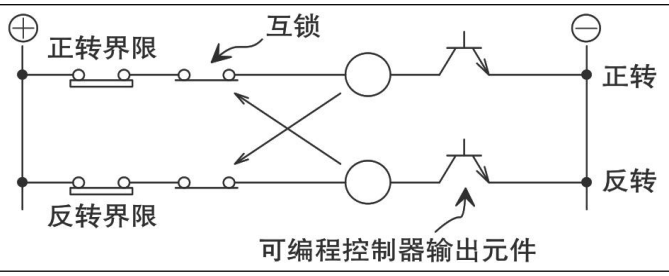
另外，不能将信号输入线和输出线与其他动力线、输出线在同一管道中通过，不能捆扎在一起。

若按上述注意事项执行，输入输出布线即使长达 50～100m，也几乎没有噪音问题。但一般为安全起见，布线长应在 20m 以内。

附记

●必须在外部电源全部切断时进行安装、接线等操作。否则会引起触电或产品损坏。

●在安装、布线等工作结束后，通电运行前，必须先装上端子盖板，以免触电。

! 危险	 正反转接触器同时合上十分危险，像这样的负载，除了可编程控器内部程序设定互锁以外，在可编程控制器外部也必须设置如上图所示的互锁。
! 注意	空端子不要与外部接线，否则会引起产品损坏。

●请按照手册中记载的内容对专用接线端进行 AC 电源的接线。

如果把 AC 电源接入直流输入输出端子或直流电源端子，会烧坏可编程控制器。

●请不要从外部电源对基本单元的 24+端子供电，对空端子，请勿从外部布线，否则会损坏产品。

请把基本单元的接地端子按第三种方式接地。但请不要和强电系统共地。

附记

●电源出现不满 10ms 的瞬时断电，可编程控制器仍会继续工作。

●长时间停电或电压偏低时，可编程控制器会停止工作，输出变为 OFF，但是一旦电源恢复供电，会自动地重新开始运转。

（4）启动．保养注意事项

危险
●请不要在通电时触摸端子，否则可能引起触电、误操作。 ●请在电源关闭后进行端子的清扫和拆卸，在通电时执行有可能引起触电。 ●请在熟读手册、充分确认安全后，再进行机器运转中的程序变更、强制输出、RUN/STOP 等操作。 操作错误会损坏机器，引起事故。

注意
●请不要进行分解和改造，否则会引起故障、误动作、火灾。 ※关于修理事宜请与福州富昌维控电子科技有限公司联系。 ●请在电源关闭之后，进行扩展电缆等连接电缆的装卸工作，否则会引起故障和误动作。

（5）维护检修

●定期检查：可编程控制器是否装有寿命较短的消耗品。

●继电器输出型，如果输出继电器异常高频度工作或驱动大容量负载时，必须注意其对使用寿命的影响。

●和其他设备一起检查，请注意以下要点：

有否由于其他发热体或直射阳光，导致机内温度异常升高；

有无粉尘或导电性尘埃侵入机内；

有无接线和端子松动及其他异常。

目录

1 程序的执行 1
1.1 扫描的构成 1
初始化处理 1
输入输出点刷新 1
程序的运算 1
END 处理 1
1.2 扫描时间 2
初始扫描时间 2
1.3 各程序的流程 2
1.4 程序的执行类型 3
扫描执行型程序 3
事件执行型程序 4
中断执行型程序 7
子程序 14
定位指令 15
2 各软元件说明 17
软元件一览表 17
2.1 用户软元件 18
2.1.1 输入继电器（X） 18
2.1.2 输出继电器（Y） 18
2.1.3 内部继电器（M） 18
2.1.4 状态继电器（S） 18
2.1.5 定时器（T） 18
2.1.6 计数器（C） 19
2.1.7 长计数器（LC） 20
2.1.8 高速计数器（HSC） 20
2.1.9 数据寄存器（D & R） 20
2.2 系统软元件 20
2.2.1 特殊继电器（SM） 20
2.2.2 特殊寄存器（SD） 20
2.3 变址寄存器 21
2.3.1 变址寄存器（[D]） 21
2.4 指针 21
2.4.1 指针（P） 21
2.5 常数 21
2.5.1 10 进制常数（K） 21
2.5.2 16 进制常数（H） 21
2.5.3 实数常数（E） 21
2.5.4 字符串常数 21

2.6 掉电保持设置 22
2.7 软元件的特殊使用方式 22
3 顺控程序指令 23
3.1 触点指令 23
3.1.1 运算开始、串联连接、并联连接 23
3.1.2 脉冲运算开始、脉冲串联连接、脉冲并联连接 26
3.2 结合指令 29
3.2.1 梯形图块串联连接、并联连接 29
3.2.2 运算结果取反 30
3.2.3 运算结果脉冲化 31
3.3 输出指令 32
3.3.1 OUT 指令（定时器、计数器除外） 32
3.3.2 SET 指令 34
3.3.3 RST 指令 36
3.3.4 下降沿输出 PLF 38
3.3.5 上升沿输出 PLS 39
4 程序流程指令 40
4.1 程序跳转 40
CJ/条件跳转 40
4.2 子程序跳转 44
CALL/子程序调用 44
4.3 中断禁止、中断允许 46
DI、EI/中断禁止、允许 46
SIMASK/中断屏蔽 49
4.4 循环指令 50
FOR~NEXT/循环 50
BREAK/跳出循环 52
4.5 看门狗复位 53
WDT/看门狗定时器 53
5 定时器、计数器输出指令 54
5.1 定时器输出指令 54
OUT T/定时器输出 54
5.2 计数器输出指令 55
OUT C/计数器输出 55
OUT LC/长计数器输出 56
6 高速输入计数器 57
6.1 高速计数器的规格 57
高速计数器的类型 57
最高频率 58
高速计数器的分配 58
高速计数器使用步骤 59
6.2 高速计数器的指令 61

OUT HSC/高速计数器开关 61
DHSCS/高速比较置位 63
DHSCR/高速比较复位 65
DHSZ/高速区间比较 66
7 应用指令 68
7.1 传送比较指令 68
MOV/16 位传送 68
DMOV/32 位传送 69
BMOV/成批传送 70
FMOV/16 位多点传送 71
DFMOV/32 位多点传送 72
SMOV/位移动 73
CML/16 位取反传送 75
DCML/32 位取反传送 76
CMP/16 位数据比较输出 77
DCMP/32 位数据比较输出 78
XCH/16 位数据交换 79
DXCH/32 位数据交换 80
ZCP/16 位数据区间比较 81
DZCP/32 位数据区间比较 82
7.2 循环移位指令 83
ROR/16 位循环右移 83
DROR/32 位循环右移 84
RCR/16 位带进位循环右移 85
DRCR/32 位带进位循环右移 86
ROL/16 位循环左移 87
DROL/32 位循环左移 88
RCL/16 位带进位循环左移 89
DRCL/32 位带进位循环左移 90
SFTR/n 位数据的 n 位右移 91
SFTL/n 位数据的 n 位左移 92
WSFR/n 字数据的 n 字右移 93
WSFL/n 字数据的 n 字左移 94
SFR/16 位数据的 n 位右移位 95
DSFR/n 字数据的 1 字右移位 96
SFL/16 位数据的 n 位左移位 97
DSFL/n 字数据的 1 字左移位 98
7.3 算术运算指令 99
ADD/16 位加法运算 99
DADD/32 位加法运算 101
SUB/16 位减法运算 103
DSUB/32 位减法运算 105

MUL/16 位乘法运算 107
DMUL/32 位乘法运算 108
DIV/16 位除法运算 109
DDIV/32 位除法运算 110
INC/16 位数据递增 111
DINC/32 位数据递增 112
DEC/16 位数据递减 113
DDEC/32 位数据递减 114
7.4 逻辑运算指令 115
NEG/16 位补码 115
DNEG/32 位补码 116
WOR/16 位数据逻辑或 117
DOR/32 位数据逻辑或 118
WAND/16 位数据逻辑与 119
DAND/32 位数据逻辑与 120
WXOR/16 位数据逻辑异或 121
DXOR/32 位数据逻辑异或 122
ANS/报警器设置 123
ANR/报警器复位 125
BON/16 位数据位判断 126
DBON/32 位数据位判断 127
ENCO/编码 128
DECO/解码 129
SUM/16 位数据的 ON 位数 130
DSUM/32 位数据的 ON 位数 131
MEAN/16 位数据的平均值 132
DMEAN/32 位数据的平均值 133
SQR/16 位平方根 134
DSQR/32 位平方根 135
WSUM/16 位数据合计值 136
DWSUM/32 位数据合计值 137
SORT/16 位数据排序 138
SORT2/16 位数据排序 141
DSORT2/32 位数据排序 144
SWAP/16 位数据高低字节交换 147
DSWAP/32 位数据高低字节交换 148
BTOW/字节单位数据合并 149
WTOB/字节单位数据分离 151
DIS/16 位数据的 4 位分离 153
UNI/16 位数据的 4 位合并 154
ZRST/数据批量复位 155
ZSET/数据批量置位 156

CRC/循环冗余校验指令 157
7.5 矩阵输入指令 159
MTR/矩阵输入 159
7.6 方便指令 161
ABSD/BIN16 位数据绝对方式 161
DABSD/BIN32 位数据绝对方式 163
SER/16 位数据查找 165
DSER/32 位数据查找 167
ALT/位软元件输出取反 169
INCD/BIN16 位数据相对方式 171
RAMP/控制斜坡信号 173
ROTC/旋转台的就近控制 175
STMR/特殊功能定时器 177
TTMR/示教定时器 179
7.7 外部 IO 指令 181
ARWS/箭头开关 181
DSW/数字键输入 183
HKY/16 进制数字键输入 185
DHKY/32 进制数字键输入 187
PR/ASCII 码打印 189
SEGD/数字键输入 191
SEGL/7SEG 码时分显示 192
TKY/16 进制数字键输入 194
DTKY/32 进制数字键输入 196
7.8 数据转换指令 198
BCD/BIN→BCD 198
BIN/4 位 BCD→BIN 199
DBIN/8 位 BCD→BIN 200
FLT/BIN 整数→2 进制浮点数 201
DFLT/BIN 整数→2 进制浮点数 203
VAL/字符串→BIN16 位数据转换 204
DVAL/字符串→BIN32 位数据转换 205
ASCI/HEX 代码数据→ASCII 转换 206
HEX/ASCII→HEX 代码数据转换 208
CCD/校验码 210
GBIN/格雷码→BIN16 位数据转换 212
DGBIN/格雷码→BIN32 位数据转换 213
GRY/BIN16 位数据→格雷码转换 214
DGRY/BIN32 位数据→格雷码转换 215
DPRUN/8 进制位传送（32 位数据） 216
7.9 浮点指令 217
DACOS/单精度实数 COS^{-1} 运算 217

DASIN/单精度实数 SIN^{-1} 运算 218
DATAN/单精度实数 TAN^{-1} 运算 219
DCOS/单精度实数 COS 运算 220
DCOSH/单精度实数 COSH 运算 221
DSIN/单精度实数 SIN 运算 222
DSINH/单精度实数 SINH 运算 223
DTAN/单精度实数 TAN 运算 224
DTANH/单精度实数 TANH 运算 225
DDEG/单精度实数弧度→角度转换 226
DRAD/单精度实数角度→弧度转换 227
DEADD/单精度实数加法运算 228
DESUB/单精度实数减法运算 229
DEMUL/单精度实数乘法运算 230
DEDIV/单精度实数除法运算 231
DEMOV/单精度实数数据传送 232
DEBCD/2 进制浮点→10 进制浮点转换 233
DEBIN/10 进制浮点→2 进制浮点转换 234
DENEG/单精度实数符号取反 235
DECMP/单精度实数比较 236
DEZCP/2 进制浮点带宽比较 237
DESQR/单精度实数平方根 238
DESTR/单精度实数→字符串转换 239
DEVAL/字符串→单精度实数转换 243
DEXP/单精度实数指数运算 246
INT/单精度实数→带符号 BIN16 位数据 247
DINT/单精度实数→带符号 BIN32 位数据 248
DLOG10/单精度实数常用对数运算 249
DLOGE/单精度实数自然对数运算 250
7.10 触点比较指令 251
有符号 16 位触点比较指令 251
有符号 32 位触点比较指令 253
单精度浮点触点比较指令 255
字符串比较 257
7.11 时钟运算指令 259
TADD/时钟数据的加法运算 259
TSUB/时钟数据的减法运算 261
TRD/时钟数据的读取 263
TWR/时钟数据的写入 264
HTOS/时间数据的 16 位数据转换（时分秒→秒） 265
DHTOS/时间数据的 32 位数据转换（时分秒→秒） 266
HOUR/小时计 16 位 267
DHOUR/小时计 32 位 268

STOH/时间数据的 16 位数据转换（秒→时分秒） 269
DSTOH/时间数据的 32 位数据转换（秒→时分秒） 270
TCMP/时钟数据比较 271
TZCP/时钟数据带宽比较 273
7.12 数据控制指令 275
BAND/BIN16 位数据死区控制 275
DBAND/BIN32 位数据死区控制 276
BINDA/BIN16 位数据→10 进制 ASCII 转换 277
DBINDA/BIN32 位数据→10 进制 ASCII 转换 278
DABIN/10 进制 ASCII→BIN 的转换 279
DDABIN/10 进制 ASCII→BIN32 位数据转换 280
LIMIT/BIN16 位数据上下限限位控制 282
DLIMIT/BIN32 位数据上下限限位控制 283
SCL/BIN16 位单位标度（各点坐标数据） 284
DSCL/BIN32 位单位标度（各点坐标数据） 287
SCL2/BIN16 位单位标度（各 X/Y 坐标数据） 290
DSCL2/BIN32 位单位标度（各 X/Y 坐标数据） 293
ZONE/BIN16 位数据区域控制 296
DZONE/BIN32 位数据区域控制 297
7.13 数据块指令 298
BK+/BIN16 位块数据加法运算 298
DBK+/BIN32 位块数据加法运算 300
BK-/BIN16 位块数据减法运算 302
DBK-/BIN32 位块数据减法运算 304
BKCMP=/BIN16 位块数据比较 306
DBKCMP=/BIN32 位块数据比较 308
BKCMP<>/BIN16 位块数据比较 310
DBKCMP<>/BIN32 位块数据比较 312
BKCMP>/BIN16 位块数据比较 314
DBKCMP>/BIN32 位块数据比较 316
BKCMP>=/BIN16 位块数据比较 318
DBKCMP>=/BIN32 位块数据比较 320
BKCMP</BIN16 位块数据比较 322
DBKCMP</BIN32 位块数据 324
BKCMP<=/BIN16 位块数据比较 326
DBKCMP<=/BIN32 位块数据比较 328
7.14 数据表操作指令 330
SFRD/移位读出 330
POP/从数据表的后入数据读取 332
SFWR/移位写入 334
FINS/数据表的数据插入 336
FDEL/数据表的数据删除 338

7.15 输入输出刷新指令 340
REF/输入输出刷新 340
REFF/输入刷新（带滤波器设定） 342
7.16 时机计测指令 343
DUTY/时钟脉冲发生指令 343
7.17 随机数指令 345
RND/随机数指令 345
7.18 优选指令 346
DEXMN/优选指令 346
8 运动控制 351
8.1 单轴运动控制的管理类功能块 351
1、 MC_POWER 351
2、 MC_READSTATUS 353
3、 MC_READMOTIONSTATE 355
4、 MC_READAXISERROR 357
5、 MC_READAXISINFO 358
6、 MC_READPARAMETER 360
7、 MC_READBOOLPARAMETER 362
8、 MC_READACTUALVELOCITY 364
9、 MC_READACTUALPOSITION 365
10、 MC_RESET 366
11、 MC_SETPOSITION 368
8.2 单轴运动控制的运动类功能块 369
1、 MC_STOP 369
2、 MC_HOME 371
3、 MC_HALT 373
4、 MC_MOVEABSOLUTE 375
5、 MC_MOVERELATIVE 379
6、 MC_MOVEADDITIVE 383
7、 MC_MOVESUPERIMPOSED 385
8.3 凸轮和齿轮多轴运动控制的管理类功能块 387
1、 MC_CAMTABLESELECT 387
2、 MC_GENERATECAMTABLE 389
3、 MC_SAVECAMTABLE 391
8.4 凸轮和齿轮多轴运动控制的运动类功能块 392
1、 MC_CAMIN 392
2、 MC_CAMOUT 397
3、 MC_GEARIN 398
4、 MC_GEAROUT 401
5、 MC_GEARINPOS 402
8.5 轴组运动控制的管理类功能 405
1、 MC_ADDAXISTOGROUP 405

2、 MC_REMOVEAXISFROMGROUP 407
3、 MC_UNGROUPALLAXES 409
4、 MC_GROUPENABLE 410
5、 MC_GROUPDISABLE 411
6、 MC_GROUPREADACTUALPOSITION 412
7、 MC_GROUPREADACTUALVELOCITY 413
8、 MC_GROUPREADERROR 414
9、 MC_GROUPREADSTATUS 415
10、 MC_GROUPRESET 417
11、 MC_GROUPREADACTUALACCELERATION 418
8.6 轴组运动类 419
1、 MC_GROUPSTOP 419
2、 MC_GROUPHALT 421
3、 MC_MOVELINEARABSOLUTE 422
4、 MC_MOVELINEARRELATIVE 424
5、 MC_MOVECIRCULARABSOLUTE 426
6、 MC_MOVECIRCULARRELATIVE 428
7、 MC_MOVEDIRECTABSOLUTE 430
8、 MC_MOVEDIRECTRELATIVE 432
8.7 其他特殊 FB 434
1、 MC_ MOVEJOG 434
2、 MC_ COMBINEAXES 436
3、 MC_ MOVEVELOCITY 438
8.8 SDO 指令 440
1、 MC_READSDO 读取 SDO 数据 440
2、 MC_WRITESDO 写入 SDO 数据 441
8.9 其他 442
8.9.1 单轴运动控制状态图 442
8.9.2 轴组运动控制状态图 443
8.9.3 圆弧模式说明 443
8.9.4 过渡模式说明 443
8.9.5 模式选择说明 444
8.10 错误码对照表 450
9 高速计数器 455
9.1 高速计数器(编码器轴)使用说明 455
9.1.1 高速计数器轴简介 455
9.1.2 创建编码器轴 455
9.1.3 编码器轴用户单位与换算 456
9.1.4 设置工作模式 458
9.1.5 设置计数器参数 460
9.2 HC 轴控 463
1、 HC_COUNTER 高速计数器使能 463

2、 HC_PRESET 高速计数器预置值 466
3、 HC_TOUCHPROBE 高速计数器探针 468
4、 HC_COMPARE 高速计数器比较 472
5、 HC_ARRAYCOMPARE 高速计数器数组比较 474
6、 HC_STEPCOMPARE 高速计数器等间距比较 477
10 通讯指令 480
10.1 通讯口协议设置 480
PROTOCOL/通讯口协议设置 480
10.2 Modbus 串口参数设置 482
PORTPARA/Modbus 串口参数设置 482
10.3 Modbus 站号设置 484
STATION/Modbus 站号设置 484
10.4 RS 指令 486
RS/外部通讯指令 486
10.5 RS 指令对应协议使用说明 489
10.5.1 自定义协议说明 489
10.5.2 Modbus 协议说明 497
10.6 WECON Modbus 协议说明 506
11 PID 控制指令 507
PRUN/8 进制位传送（16 位数据） 507
TRH/干湿球温度与湿度的换算 508
12 字符串指令 510
LEN/字符串的长度检测 510
LEFT/从字符串的左侧开始提取 511
RIGHT/从字符串的右侧开始提取 513
MIDR/字符串中的任意提取 515
$MOV/字符串传送 517
MIDW/字符串中的任意替换 519
STR/BIN16 位数据→字符串转换 522
DSTR/BIN32 位数据→字符串转换 524
$+/字符串的合并 526
INSTR/字符串查找 527
ASC/ASCII 数据输入 529
13 以太网通讯 530
13.1 以太网概述 530
13.1.1 IP 地址概念 530
13.1.2 设置 PC 网络地址 530
13.1.3 测试网络连接状态 531
13.1.4 PLC Editor2 与 PLC 进行以太网连接 532
13.1.5 PLC Editor2 的以太网搜索功能 533
13.2 以太网配置 534
13.2.1 IP 地址设置 534

13.2.2 TCP 协议概念 535
13.3 以太网相关特殊软元件一览表 536
13.4 以太网运算错误码一览表 540
14 内置 FC/FB 543
14.1 内置 FC 函数 543
1、 ADD/加法运算指令 543
2、 SUB/减法运算指令 545
3、 MUL/乘法运算指令 547
4、 DIV/除法运算指令 548
5、 MOD/取余运算指令 549
6、 ABS/求绝对值（整数） 550
7、 FABS/求绝对值（实数） 550
8、 ASIN/SIN^{-1}运算 551
9、 ATAN/TAN^{-1}运算 552
10、 ACOS/COS^{-1}运算 553
11、 COS/COS 运算 554
12、 SIN/SIN 运算 555
13、 TAN/TAN 运算 556
14、 SQRT/平方根运算 557
15、 EXP/指数运算 558
16、 EXPT/幂运算 559
17、 LOG/对数运算 560
18、 LN/自然对数运算 561
19、 LIMIT/上下限位控制 562
20、 MAX/取最大值 563
21、 MIN/取最小值 564
22、 AND/逻辑与 565
23、 OR/逻辑或 566
24、 XOR/异或 567
25、 NOT/取非 568
26、 SHL/左移指令 569
27、 SHR/右移指令 570
28、 ROL/循环左移指令 571
29、 ROR/循环右移指令 572
30、 SEL/二选一指令 573
31、 MUX/多路复用器 574
32、 GT/大于指令 575
33、 LT/小于指令 576
34、 GE/大于等于指令 577
35、 LE/小于等于指令 578
36、 EQ/比较指令（等于） 579
37、 NE/比较指令（不等于） 580

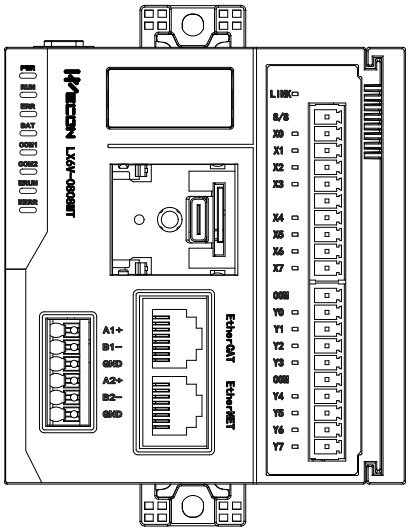
38、 SIZEOF/获取大小指令 581
39、 TRUNC/截尾取整指令 581
40、 MOVE/传送指令 582
41、 BOOL_TO_<TYPE>/BOOL 类型转换指令 583
42、 BYTE_TO_<TYPE>/BYTE 类型转换指令 585
43、 DATE_TO_<TYPE>/DATE 类型转换指令 586
44、 DINT_TO_<TYPE>/DINT 类型转换指令 588
45、 DT_TO_<TYPE>/DT 类型转换指令 590
46、 DWORD_TO_<TYPE>/DWORD 类型转换指令 592
47、 INT_TO_<TYPE>/INT 类型转换指令 593
48、 WORD_TO_<TYPE>/WORD 类型转换指令 595
49、 REAL_TO_<TYPE>/REAL 类型转换指令 596
50、 SINT_TO_<TYPE>/SINT 类型转换指令 597
51、 TIME_TO_<TYPE>/TIME 类型转换指令 598
52、 TOD_TO_<TYPE>/TOD 类型转换指令 599
53、 UDINT_TO_<TYPE>/UDINT 类型转换指令 601
54、 UINT_TO_<TYPE>/UINT 类型转换指令 602
55、 USINT_TO_<TYPE>/USINT 类型转换指令 603
56、 LREAL_TO_<TYPE>/LREAL 类型转换指令 604
57、 LWORD_TO_<TYPE>/LWORD 类型转换指令 605
58、 LINT_TO_<TYPE>/LINT 类型转换指令 606
59、 LTIME_TO_<TYPE>/LTIME 类型转换指令 607
60、 TO_<TYPE>类型转换指令 608
14.2 内置 FB 功能块 610
1、 R_TRIG/上升沿检测 610
2、 F_TRIG/下降沿检测 611
3、 RS/双稳态功能块(复位优先) 612
4、 SR/双稳态功能块(置位优先) 613
5、 CTU/加计数器 614
6、 CTD/减计数器 616
7、 CTUD/加、减计数器 618
8、 TON/ON 延迟定时器 620
9、 TOF/OFF 延迟定时器 621
10、 TP/脉冲定时器 622
11、 RTC/万年历 623
附录 624
附 1 特殊继电器（SM） 624
出错信息 624
系统信息 624
时钟信息 625
扫描信息 625
指令相关 625

中断禁止 626
高速输入输出 627
通讯 628
以太网相关特殊软元件一览表 629
附 2 特殊寄存器（SD） 630
出错信息 630
系统信息 630
时钟信息 631
扫描信息 631
指令相关 632
中断禁止 633
高速输入输出 633
右扩展模块 636
EtherCAT 模块 636
输入滤波 636
通讯 636
以太网相关特殊软元件一览表 638
日志信息 638
附 3 错误码整理 639
PLC 硬件错误 639
梯形图执行错误 639
PLC 参数错误 639
EtherCAT 配置错误 640
PLC 通讯错误 641
PLC 运算错误 643
右扩展模块错误（报通讯错误） 647
附 4 ASCII 代码对照表 648
ASCII 代码对照表 648

1 程序的执行

1.1 扫描的构成

CPU 模块的扫描的构成如下所示。



初始化处理

基于 CPU 模块状态的初始化如下所示。

√：执行、 ×：不执行

处理项目	CPU 模块的状态		
	电源 ON 时	STOP 时	STOP RUN 时
输入输出模块的初始化	√	×	×
CPU 参数的检查	√	×	×
系统参数的检查	√	×	×
软元件初始化	√	×	√
错误的清除	√	×	√

输入输出点刷新

开始程序运算前执行以下内容。

- 将 PLC 实际输入点更新到输入继电器 X。

END 指令执行后执行以下内容。

- 将 PLC 输出继电器 Y 更新到实际输出点。

要点

执行恒定扫描时，恒定扫描的等待时间结束后再进行 I/O 刷新。

程序的运算

根据程序设置，从各程序的步 0 开始执行至 END 指令为止。该程序称为主程序。

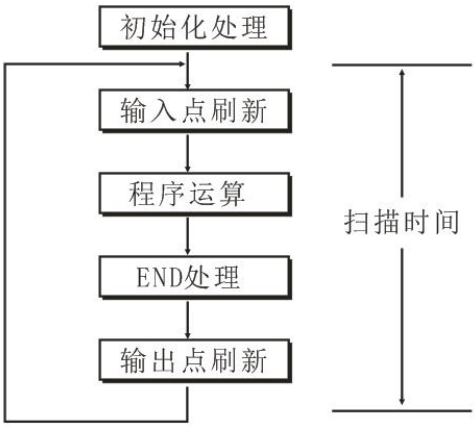
END 处理

进行以下处理。

- 部分指令的完成处理
- 看门狗定时器的复位
- 通信处理
- 对特殊继电器/特殊寄存器的值的设置（设置时机为 END 处理时）

1.2 扫描时间

CPU 模块重复以下处理，扫描时间是以下处理及执行时间的合计。



*1 初始扫描时间表示包含本处理的时间。

初始扫描时间

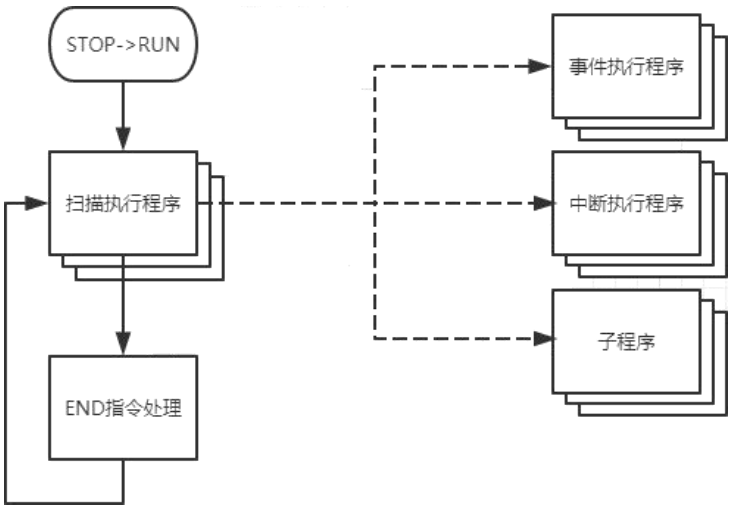
是 CPU 模块在 RUN 时的首次扫描时间。

通过以下方式进行：

- SD134（初始扫描时间（ms 单位））、SD135 （初始扫描时间（s 单位））中存储的值。

1.3 各程序的流程

在 CPU 模块变为 RUN 状态时，按照程序的执行类型及执行顺序的设置，依次执行程序。



要点

程序的执行类型相同时，按照执行顺序中设置的顺序执行。

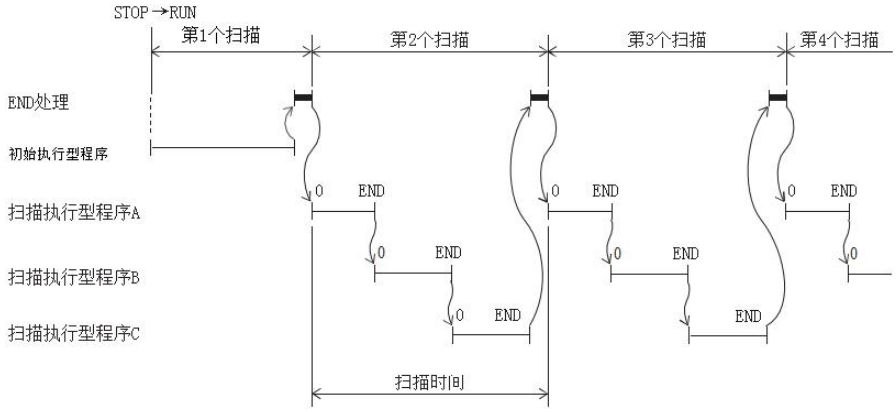
注意事项

使用多周期才能完成的执行指令时（如 OUT　T、RAMP、RS 等），都应在扫描程序编程。若在事件执行型程序、中段执行型程序中使用时，可能会由于多个扫描周期都未执行这些指令，导致实际结果与理想结果不同，所以除非每个扫描周期都可执行到的事件、中断、子程序，其他情况下都不建议使用多周期执行指令。

1.4 程序的执行类型

扫描执行型程序

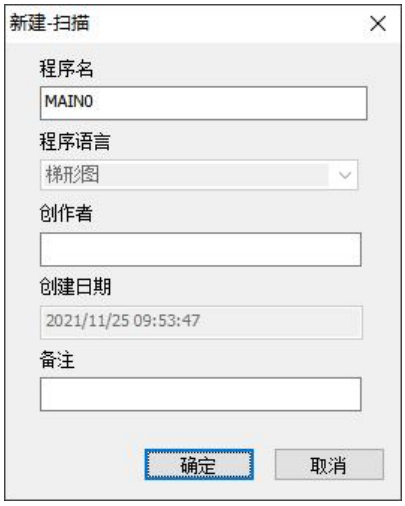
从执行了初始执行型程序的下一个扫描开始 1 个扫描仅执行一次。



执行多个扫描执行型程序时，扫描执行型程序的执行时间为全部扫描执行型程序执行完成为止的时间。此外，在扫描执行型程序的执行完成前，如执行了中断程序/事件执行型程序/子程序，则这些执行时间也将包含在内。

多个扫描程序的创建

[工程管理]→[程序]→[扫描]→[右键新建]

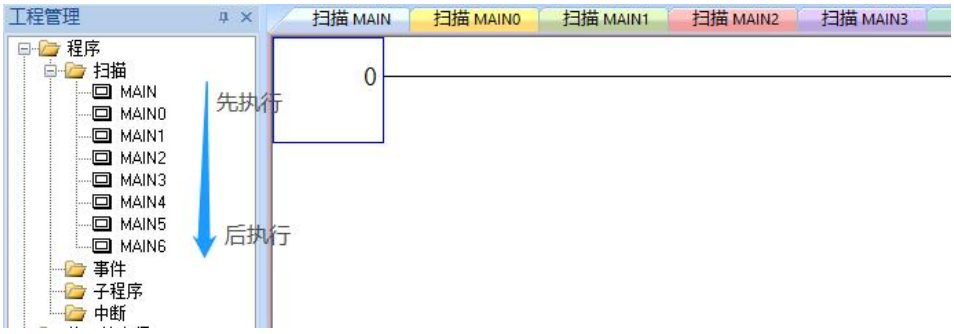


（1）扫描程序名，程序名区分大小写，不能使用与软元件同名的程序名（软元件名不区分大小写）。

（2）不支持/%$@&=~`^<>?:{}[],;!*.\'"字符的输入，不能超过 64 个字符，超出不可输入。默认名为 MAINx。

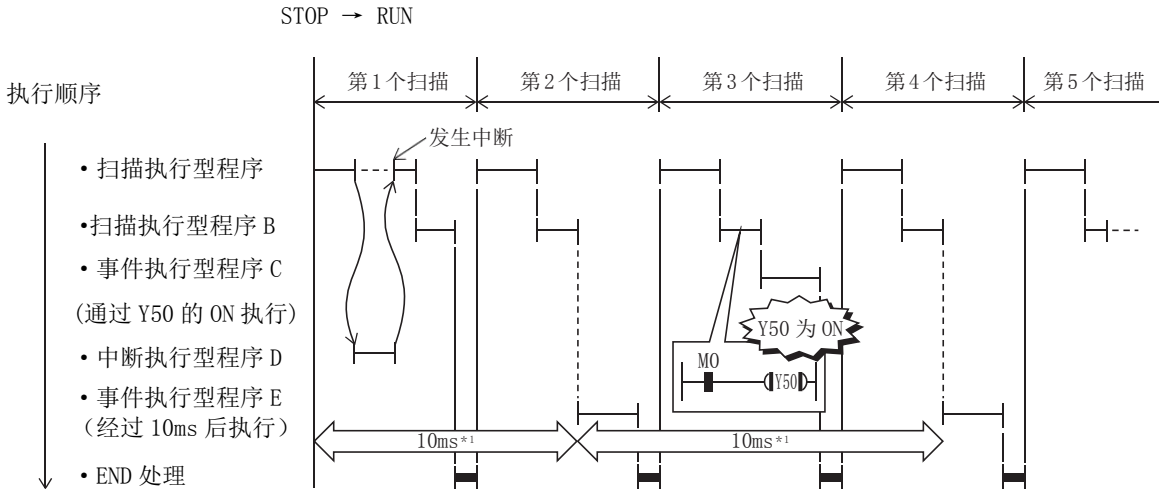
（3）可新建的扫描程序限制为 100 个。

（4）执行顺序按照创建顺序自上而下运行。



事件执行型程序

是将用户指定的事件作为触发开始执行的程序的条件。



触发类型

事件执行型程序的触发如下所示。

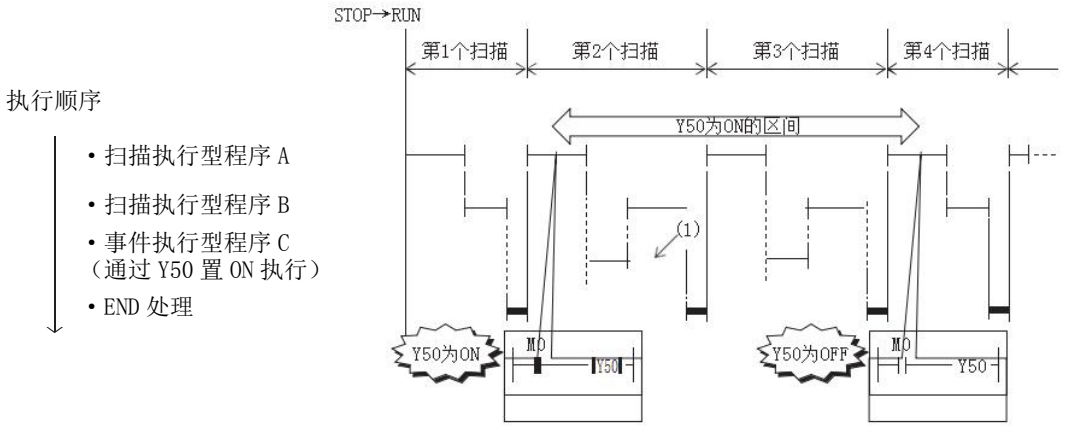
1、位数据的 ON 事件（TRUE）

（1）指定为 ON 事件后，若设置 ON 事件中触发条件的触点在扫描程序时被置 ON，则 ON 事件程序将在扫描程序分页*1 中或 END 指令执行之前执行。

（2）ON 事件程序在单个扫描周期只会执行一次。

（3）ON 事件执行后，可设置是否清除程序内使用的输出(Y)和定时器(T)的当前值。

*1：扫描程序分页：建立了多个扫描程序，各个扫描程序都称为一个分页，在执行完扫描程序 A 后，执行扫描程序 B 之前，会判断是否需要执行事件程序。



（1）轮到事件执行型程序 C 的执行顺序时且 Y50 为 ON 的情况下，执行程序。

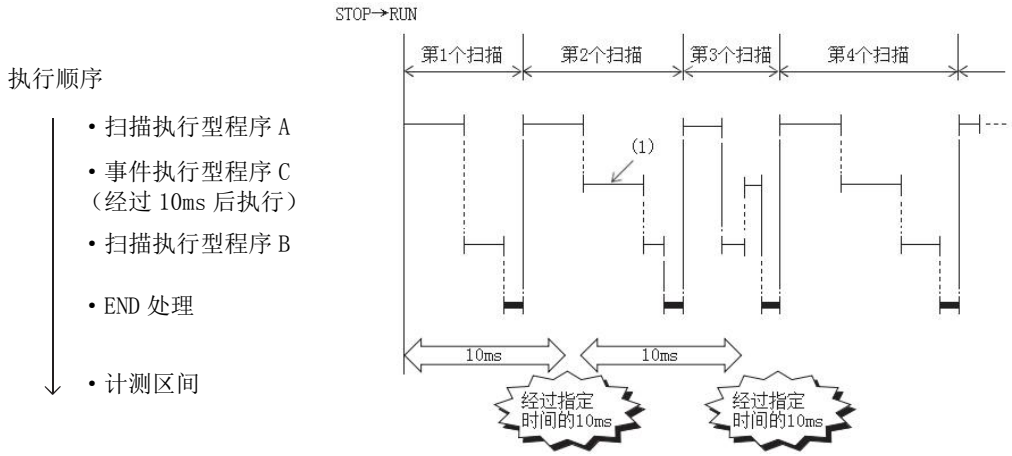
可指定的软元件如下所示：

项目		内容
软元件*1	位软元件	X、Y、M、SM
	字软元件的位指定	D.b

*1 无法指定已进行变址修饰的软元件。

2、TIME 事件

程序运行到 RUN 后，经过指定时间后，轮到首个相应程序的执行顺序时，执行 1 次。对于第 2 次以后的执行，从上次的事件执行型程序的开始重新计测时间。经过指定时间后，轮到首个相应程序的执行顺序时，重复执行程序。此外，在相应程序执行后的下一个扫描中，可清除相应程序内使用的输出（Y）及定时器（T）的当前值。可用于无需响应特别及时的固定周期运行的程序。



(1)经过指定时间后，轮到首个执行顺序时，执行事件执行型程序 C。

要点

设置为清除输出及定时器的当前值，且扫描时间长于经过时间的设置值时，输出及定时器的当前值不会被清除。

操作步骤

1、新建事件

工程管理界面→[程序]→[事件]→[右键新建]。

事件程序名：

（1）程序名区分大小写，不能使用与软元件同名的程序名（软元件名不区分大小写）；

（2）程序名不支持/%$@&=~`^<>?:{}[],;!*.\'"字符的输入；

（3）程序名长度不能超过 64 个字符，超出不可键入。默认名为 EVENTx.

新建事件程序最多可支持 100 个。

2、执行类型

事件执行类型的配置有两种方式：

（1）新建事件程序时配置，如上图所示。

（2）工程管理界面→[程序]→[参数]→[程序参数]→[配置]

配置说明

（1）配置界面：

（2）参数内容：

项目		内容	设置范围	默认
执行类型		选择事件类型	未设置/ON 事件/TIME 事件	未设置
ON 事件	触点	事件类型选择 ON 事件时可设置。设置作为触发条件的位软元件。	X/Y/M/SM/D.b	
	是否清除	当 ON 事件设置的触发条件的位软元件置 OFF 后，下个周期是否清除 ON 事件执行程序中使用的输出(Y)以及定时器(T)的当前值。	True False	False
TIME 事件	时间	设置经过多长时间执行一次事件程序。	[1,2147483647](100us 单位)	
	是否清除	当 TIME 事件执行后，下个扫描周期若没有执行该事件，选择是否清除 TIME 事件执行程序中使用的输出(Y)以及定时器(T)的当前值。	True False	False

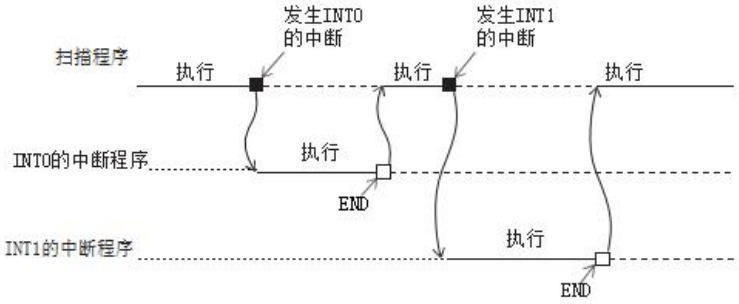
要点

指定“ON 事件”或“TIME 事件”时，若将“是否清除”设置为“清除”，则在一个扫描周期内，都没有执行该事件程序，其程序内部的所有输出（Y）及定时器（T）的当前值将都会被清除（累计型和子程序型的 T 除外）。若 TIME 事件所设置的时间小于扫描周期时，相当于每个扫描周期都能执行都 TIME 事件，即使设置了清除输出，该事件程序内的输出和定时器的数据也不会被清除。

中断执行型程序

在执行扫描程序的过程中，可打断扫描程序优先执行的程序称之为中断执行型程序。

◆ 发生中断原因时，将执行与该中断指针编号相对应的中断程序。但是，执行需要通过 EI 指令设置为中断允许状态。



◆ 一个中断名对应一个中断程序，中断名不能重复。每个中断都有自己的触发条件和执行程序，每个中断程序都以 END 结束。

◆ 中断具有打断原来执行程序优先执行中断的特性，但无法打断正在执行中的中断程序。

◆ 中断程序拥有优先度的概念，优先度数值越小，越优先响应，优先度的可设置范围为 0-2。

发生中断原因时的动作如下所示：

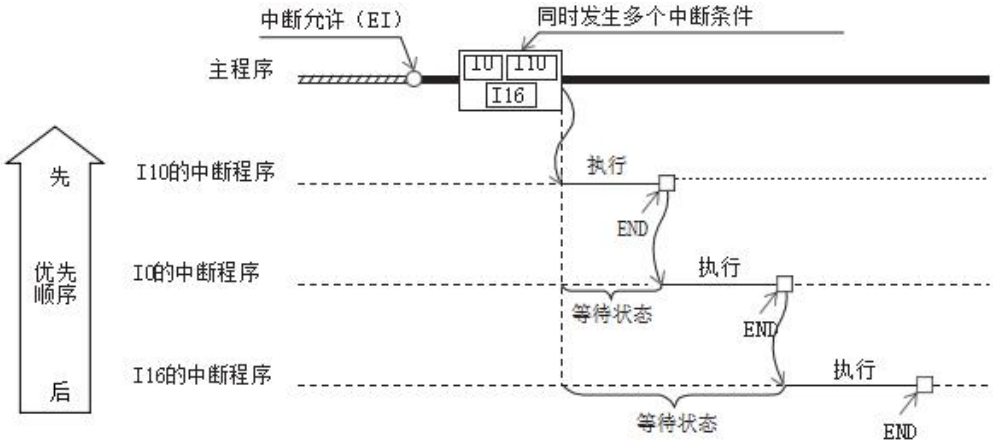
（1）中断禁止中（DI）发生中断原因时；

在被 DI 禁止的程序中触发中断执行的条件，该中断将不被执行，即使之后的程序使用了 EI 指令允许中断，也不会执行之前被屏蔽的中断程序，只有再次触发中断执行条件，该中断程序才会被执行。

（2）中断允许状态中同时发生多个中断原因时；

将从优先度高的中断程序开始依次执行。此外，同时发生多个优先度相同的中断时，按照中断优先顺序执行动作。

若新建 I0、I10、I16 三个中断程序，I0 优先度为 1，I10 优先度为 0，I16 优先度为 1，执行逻辑如下图所示：I10 优先度最小，优先执行；I0 和 I16 的优先度相同，按照程序建立顺序执行。



（3）执行恒定扫描时的等待时间中发生中断时；

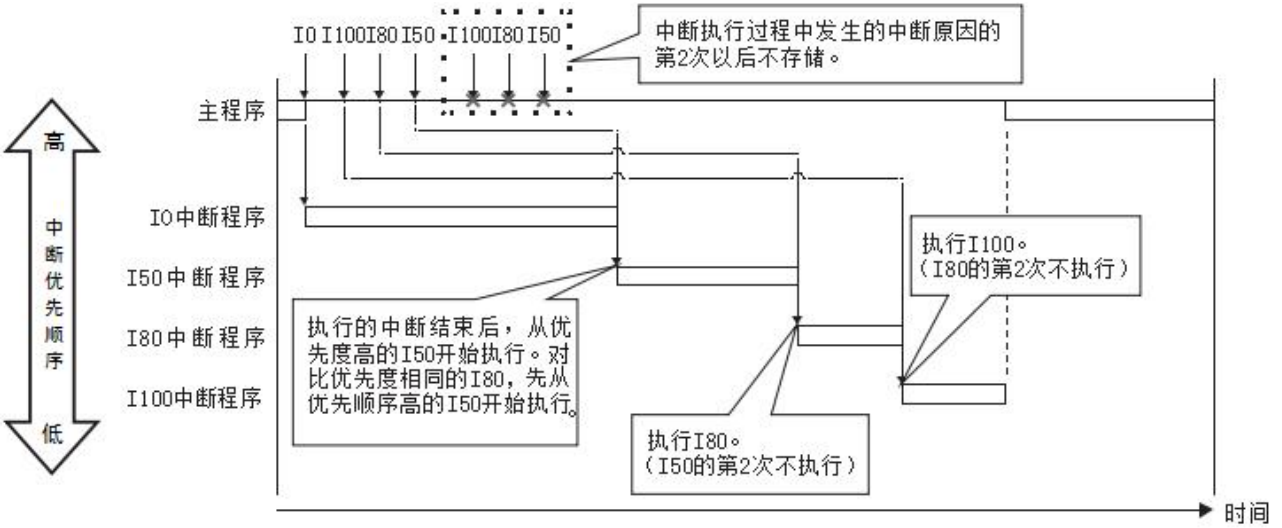
执行该中断的中断程序。

（4）在中断程序执行中发生其他的中断时；

中断程序（包含发生事件执行程序的中断发生时的指定）中，发生了其他中断时，不会打断原中断的执行程序，等待原中断执行程序完成后，执行新的中断程序，执行完成后再返回扫描程序。

（5）中断程序执行中，发生优先度低或优先度相同的中断原因时；

存储发生的中断原因，在执行中的中断程序结束后，执行与存储的中断原因相对应的中断程序。即使多次发生同一中断原因，也仅存储一次该中断原因。



（6）在中断程序执行中发生了同一中断原因时；

存储发生的中断原因，在执行中的中断程序结束后，执行与存储的中断原因相对应的中断程序。即使多次发生同一中断原因，也仅存储一次该中断原因。

中断触发条件分类

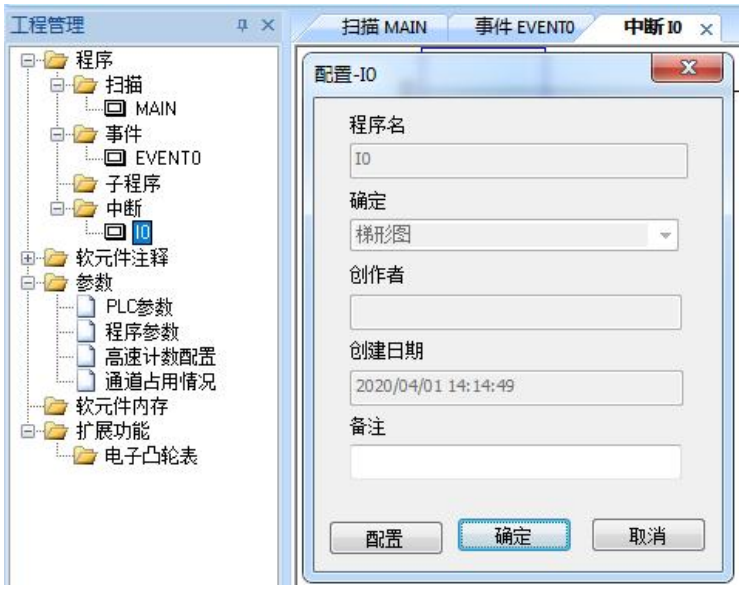
1、外部输入（X）中断

（1）外部输入中断说明

- 外部输入中断由固定 X 点输入的上升沿或下降沿作为触发条件。
- 支持 X0-X7 共 8 个输入点的上升沿和下降沿中断，共支持 16 个外部输入中断。
- 同一个中断触发条件不能建立多个中断程序。
- 外部输入中断与高速计数器无法使用同一个 X 点。
- 必须在扫描程序中使用 EI 允许中断后，中断执行程序才会执行。

（2）外部输入中断步骤

工程管理界面→[程序]→[中断]→[右键新建]

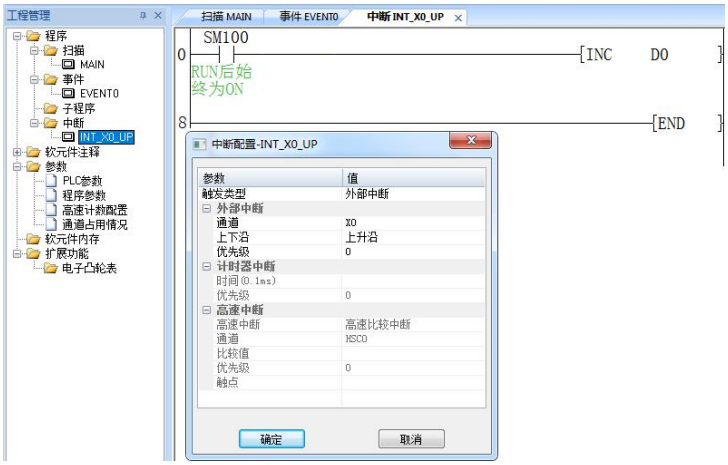


- 中断程序名区分大小写，不能使用与软元件同名的程序名（软元件名不区分大小写），
- 中断程序名不支持/%$@&=~`^<>?:{}[],;!*.\'"字符的输入，
- 中断程序名长度不能超过 64 个字符，超出不可键入。默认名为 INTx。

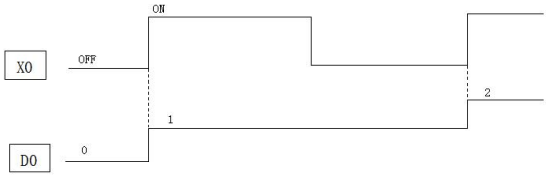
点击配置，执行类型选择外部中断，如下图（也可在工程管理中的“参数”中的“程序参数中配置”）：

项目		内容	设置范围	默认
执行类型		选择中断的类型	未设置/外部输入中断/计时器事件/高速计数中断	未设置
外部输入中断	通道	选择外部输入中断的通道	X0~X7	X0
	触发边沿类型	选择为上升沿触发还是下降沿触发	上升沿；下降沿	上升沿
	优先度	多个中断同时到达时，优先执行的顺序，数值小的先执行	[0,2]	0
	滤波时间(0.01us)	设置 X 点的滤波时间，单位为 0.01us。 注意：X 上升沿中断和 X 下降沿中断是使用同一个 X 滤波，所以在 X 上升沿配置中滤波设置改变后，X 下降沿也会跟随变化。若滤波时间设置为 1000，则必须保证输入的信号高电平和低电平都保持 10us 以上才可触发中断。	[0,1700]	1

（3）编写中断执行程序



双击工程管理中的新创建的中断程序，即可开始编写中断执行程序。如上图新建了一个中断程序为 INT_X0_UP，触发条件配置为 X0 上升沿时执行中断程序，若主程序中使用了 EI 指令允许中断，则每当 X0 从 OFF→ON 时，都会执行 INT_X0_UP 中的所有程序，即 D0 会自增一次。



2、定时器中断

（1）定时器中断说明

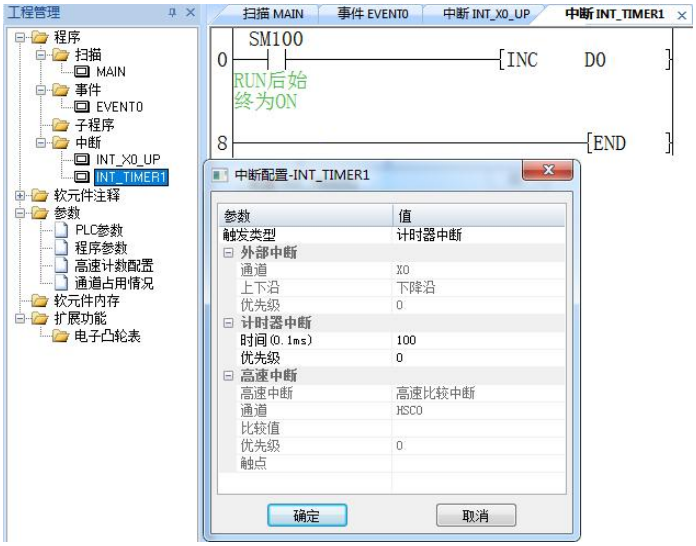
- 定时器中断是根据所设定的时间，每隔该时间执行中断程序一次，最小时间间隔可达 100us；
- 最多可支持创建 100 个定时器中断执行程序；
- 每个定时器中断程序互相独立，互不影响；
- 每个定时器中断程序都应配置优先度，同时触发时，按照优先度顺序执行，但优先度一样时，按照建立程序的先后顺序执行；
- 必须在扫描程序中使用 EI 允许中断后，中断执行程序才会执行。

（2）定时器中断步骤

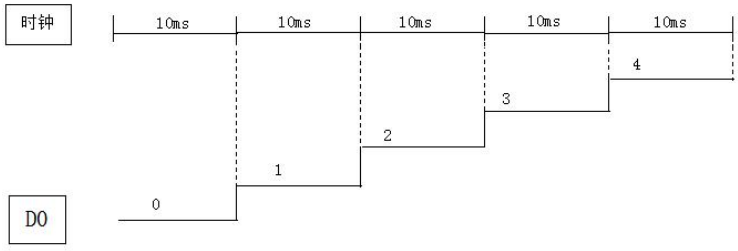
工程管理界面→[程序]→[中断]→[右键新建]。输入程序，点击配置，执行类型选择定时器中断，如下图（也可在工程管理中的“参数”中的“程序参数中配置”）。

项目		内容	设置范围	默认
执行类型		选择中断的类型	未设置/外部输入中断/计时器事件/高速计数中断	未设置
计时器中断	时间	设置中断触发的间隔时间	[1, 2147483647]（100us 单位）	
	优先度	多个中断同时到达时，优先执行的顺序，数值小的先执行	[0, 2]	0

（3）编写中断执行程序



双击工程管理中的新创建的定时器中断程序，即可开始编写中断执行程序。如上图新创建了一个定时器中断程序为 INT0，触发条件配置为每隔 10ms 执行中断程序，若主程序中使用 EI 执行允许中断，则每经过 10ms 都会执行一次 INT0 中所有的指令程序，即 D0 会 10ms 加 1。



3、高速计数中断

（1）高速计数中断说明

◆ 高速计数中断根据 PLC 提供的高速计数器 HSC0-HSC7 经过设定值后触发一次中断 条件，执行中断程序；

◆ 最多可支持 100 个高速计数中断程序，但每个通道可支持的个数无需固定；

◆ 高速计数中断使用时必须配置好高速计数器，并使用 OUT HSC 指令开启相应计数通道计数，才能够使用（具体配置方法查看高速计数器说明章节）；

◆ 每个高速计数中断程序都应配置优先度，同时触发时，按照优先度顺序执行，优先度一样时，按照通道 HSC0-HSC7 顺序执行。通道也一样时，按照创建承诺需顺序执行；

◆ 必须在扫描程序中使用 EI 允许中断后，中断执行程序才会执行。

注意事项：HSC 的通道与外部输入中断的通道都必须使用 PLC 的输入点 X，在配置时需要注意不能复用，具体可参照高速计数器的配置章节。

（2）高速计数中断步骤

工程管理界面→[程序]→[中断]→[右键新建]。输入程序名，点击配置，执行类型选择高速中断，如下图（也可在工程管理中的“参数”中的“程序参数中配置”）。

项目		内容	设置范围	默认
执行类型		选择中断的类型	未设置/外部输入中断/计时器事件/高速计数中断	未设置
高速计数中断	模式	选择高速计数中断的类型： （1）高速比较中断：达到触发条件后执行中断程序。 （2）高速比较置位：达到触发条件后将设置的触点置位。 （3）高速比较复位：达到触发条件后将设置的触点复位。	高速比较中断 高速比较置位 高速比较复位 未设置	高速比较中断
	通道	选择使用的高速计数器通道	HSC0~HSC7	HSC0
	比较值	设置高速计数器的比较值，在所设置通道的高速计数器值经过这个值时，达到触发条件。	[-2147483648, 2147483647]	
	优先度	多个中断同时到达时，优先执行的顺序，数值小的先执行	[0,2]	0
	触点	选择高速比较置位与高速比较复位时可用，达到触发条件后立即将该触点置位或复位。	Y/M/D.b	

（3）触发规则说明

模式	配置	当前值	动作
高速比较中断（INT0）	比较值：10000	9999 → 10000	执行中断 INT0 内的所有程序。
		10001 → 10000	执行中断 INT0 内的所有程序。
高速比较置位（INT1）	比较值：-50000 触点：Y10	-50001 → -50000	Y10 立即置位且映射到实际输出（不受扫描周期影响） INT1 内的程序不会被运行。
		-49999 → -50000	Y10 立即置位且映射到实际输出（不受扫描周期影响） INT1 内的程序不会被运行。
高速比较复位（INT2）	比较值：400000 触点：Y10	399999 → 400000	Y10 立即复位且映射到实际输出（不受扫描周期影响） INT2 内的程序不会被运行。
		400001 → 400000	Y10 立即复位且映射到实际输出（不受扫描周期影响） INT2 内的程序不会被运行。

注意事项：

HSC 的通道与外部输入中断的通道都需要使用到 PLC 的输入点 X，在配置是需要注意不能复用，具体可参照高速计数器的配置章节。

（4）编写中断执行程序

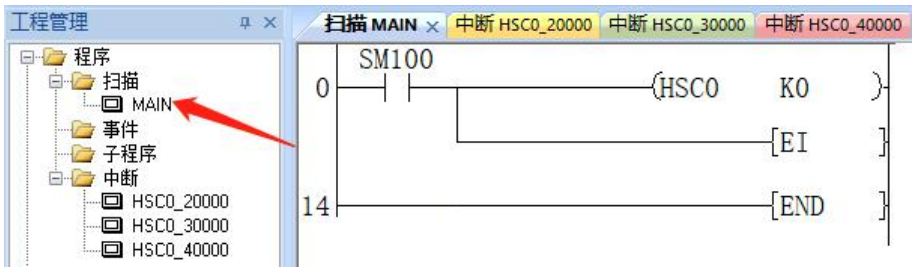
a.新建中断程序

在工程管理的中断下新建三个中断程序分别为 HSC0_20000、HSC0_30000、HSC0_40000，在“程序参数”中进行中断程序的配置，如下图。

b.高速计数的配置

在高速计数配置中将 HSC0 配置为使用，选择工作模式后，点击[检查]按钮，弹出配置正确框后，点击输入即可。

在主程序中调用高速计数器，并使能中断：



程序运行情况：

假设高速计数通道 0 一直接收脉冲：

HSC0 计数值从 0 累加到 20000 时，执行 HSC0_20000 的所有程序；

HSC0 计数值从 20000 累加到 30000 时，执行 HSC0_30000 的所有程序；

HSC0 计数值从 30000 累加到 40000 时，执行 HSC0_40000 的所有程序。

屏蔽中断

1、通过应用指令屏蔽

PLC 的中断在上电时默认是处于屏蔽状态的，只有通过 EI 指令允许中断后才能使用。

中断的屏蔽指令 DI 不带参数时屏蔽所有中断，带参数时可屏蔽部分优先级的中断（详细参考程序流指令 DI/EI）。

2、通过特殊寄存器 SM、SD 屏蔽

（1）外部输入中断屏蔽寄存器

特殊寄存器编号	中断类型	说明	默认值
SM352	X0 上升沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM353	X0 下降沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM354	X1 上升沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM355	X1 下降沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM356	X2 上升沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM357	X2 下降沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM358	X3 上升沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM359	X3 下降沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM360	X4 上升沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM361	X4 下降沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM362	X5 上升沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM363	X5 下降沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM364	X6 上升沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM365	X6 下降沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM366	X7 上升沿中断	ON：屏蔽中断 OFF：允许中断	OFF
SM367	X7 下降沿中断	ON：屏蔽中断 OFF：允许中断	OFF

（2）定时器中断屏蔽寄存器

特殊寄存器编号	中断类型	说明	默认值
SD350	第 1~16 个定时器中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD351	第 17~32 个定时器中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD352	第 33~48 个定时器中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD353	第 49~64 个定时器中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD354	第 65~80 个定时器中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD355	第 81~96 个定时器中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD356	第 97~100 个定时器中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0

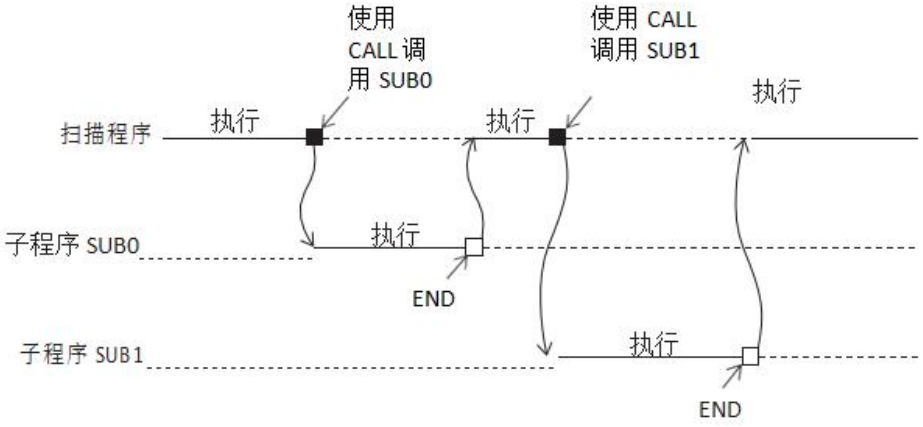
（3）高速计数中断屏蔽寄存器

特殊寄存器编号	中断类型	说明	默认值
SD382	第 1~16 个高速计数中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD383	第 17~32 个高速计数中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD384	第 33~48 个高速计数中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD385	第 49~64 个高速计数中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD386	第 65~80 个高速计数中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD387	第 81~96 个高速计数中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0
SD388	第 97~100 个高速计数中断	每个位可控制一个中断的屏蔽。ON：屏蔽中断； OFF：允许中断。	0

子程序

在执行扫描程序的过程中，可以通过 CALL 指令来调用执行的程序。子程序最多可以新建 100 个。

子程序是将主程序中某一个模块拆分出来，供主程序调用，有利于程序的模块化。如其他高级语言的函数，但是该函数无参数且无返回值。



1、子程序调用说明

新建子程序后，程序内容不执行，只有当扫描、事件、中断程序中使用 CALL(P)指令调用子程序时才执行，且调用一次执行一次。如下图新建的三个子程序 SUB0、SUB1、SUB2。主程序 MAIN 中使用 CALL(P)子程序程序名就可调用到子程序。

通过这种编程方式，对不同条件使用同一逻辑的程序可减少梯形图步数，且对梯形图的阅读性有所提高。

2、注意事项

（1）使用定时器时（OUT　T）需注意在未调用到子程序时，不会将输出复位，需使用特定的子程序寄存器。

（2）子程序间不能递归调用，即在 SUB0 中调用 SUB1，在 SUB1 中又调用 SUB0，这是不被允许的。

（3）子程序最多可嵌套 32 层，超过 32 层将会报严重错误，强行停止梯形图运行。

定位指令

1、事件

（1）ON 事件

若在 ON 事件中开启高速脉冲指令，高速脉冲指令按正常发送，如果脉冲发送中该扫描周期中触发事件 ON 触点变为 OFF，那么根据未扫描处理标志位进行选择脉冲继续发送或者脉冲停止。

输出轴	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
未扫描处理标志位	SM899	SM959	SM1019	SM1079	SM1139	SM1199	SM1259	SM1319

当该标志位为 0（继续发送脉冲）时，若当前扫描周期未扫描到该指令，则继续发送脉冲直到停止。此时需要注意，若将脉冲发送完成后触发事件 OFF 触点变为 ON，那么会重新发送脉冲。

当该标志位为 1（停止发送脉冲）时，若某一扫描周期中触发事件 ON 触点变为 OFF 时，减速停止。

（2）TIME 事件

若在 TIME 事件中开启高速脉冲指令，高速脉冲指令按正常发送，如果脉冲发送中某个扫描周期未扫到该指令，那么根据未扫描处理标志位进行选择脉冲继续发送或者脉冲停止。

输出轴	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
未扫描处理标志位	SM899	SM959	SM1019	SM1079	SM1139	SM1199	SM1259	SM1319

当该标志位为 0（继续发送脉冲）时，若当前扫描周期未扫描到该指令，则继续发送脉冲直到停止。在 TIME 事件中，无法确保每个扫描周期中都扫到该指令，所以应该避免在 TIME 时间中使用高速脉冲指令，否则会出现脉冲发送完成后，重新发送脉冲的情况。

当该标志位为 1（停止发送脉冲）时，若当前扫描周期未扫描到该指令时，那么减速停止。在 TIME 事件中，如果将该标志位设为 1（停止发送脉冲），会出现不进行脉冲发送的现象。

2、子程序

若在子程序中开启高速脉冲指令，高速脉冲指令按正常发送，如果脉冲发送中将扫描周期关闭，那么根据未扫描处理标志位进行选择脉冲继续发送或者脉冲停止。

输出轴	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
未扫描处理标志位	SM899	SM959	SM1019	SM1079	SM1139	SM1199	SM1259	SM1319

当该标志位为 0（继续发送脉冲）时，若当前扫描周期未扫描到该指令，则继续发送脉冲直到停止。此时需要注意，若将脉冲发送完成后重新调用子程序，那么会重新发送脉冲。

当该标志位为 1（停止发送脉冲）时，若高速脉冲发送过程中，子程序被关闭，那么减速停止。如果未开始发送脉冲子程序就被关闭，那么不进行脉冲的发送。

3、中断

（1）外部中断

若在外部中断中开启高速脉冲指令，高速脉冲指令根据未扫描处理标志位进行选择脉冲继续发送或者脉冲停止。

输出轴	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
未扫描处理标志位	SM899	SM959	SM1019	SM1079	SM1139	SM1199	SM1259	SM1319

当该标志位为 0（继续发送脉冲）时，则继续发送高速脉冲直到停止。

当该标志位为 1（停止发送脉冲）时，则高速脉冲减速停止。

（2）计时器中断

若在计时器中断中开启高速脉冲指令，高速脉冲指令按正常发送，如果脉冲发送中某个扫描周期未扫到该指令，那么根据未扫描处理标志位进行选择脉冲继续发送或者脉冲停止。

输出轴	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
未扫描处理标志位	SM899	SM959	SM1019	SM1079	SM1139	SM1199	SM1259	SM1319

当该标志位为 0（继续发送脉冲）时，若当前扫描周期未扫描到该指令，则继续发送脉冲直到停止。在计时器中断

中，无法确保每个扫描周期中都扫到该指令，所以应该避免在 T 计时器中断中使用高速脉冲指令。否则会出现脉冲发送完成后，重新发送脉冲的情况。

当该标志位为 1（停止发送脉冲）时，若当前扫描周期未扫描到该指令时，那么减速停止。在 TIME 事件中，如果将该标志位设为 1（停止发送脉冲），会出现不进行脉冲发送的现象。

（3）高速比较中断

若在高速比较中断中开启高速脉冲指令，那么高速脉冲指令根据未扫描处理标志位进行选择脉冲继续发送或者脉冲停止。

输出轴	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
未扫描处理标志位	SM899	SM959	SM1019	SM1079	SM1139	SM1199	SM1259	SM1319

当该标志位为 0（继续发送脉冲）时，则继续发送高速脉冲直到停止。

当该标志位为 1（停止发送脉冲）时，则高速脉冲减速停止。

2 各软元件说明

软元件一览表

分类	类型	软元件名称	符号	范围	标记
用户软元件	位	输入继电器	X	[0,1777]	8 进制数
	位	输出继电器	Y	[0,1777]	8 进制数
	位	内部继电器	M	[0,7999]	10 进制数
	位	步进继电器	S	[0,4095]	10 进制数
	位/字	定时器	T	[0,511]	10 进制数
	位/字	计数器	C	[0,255]	10 进制数
	位/双字	长计数器	LC	[0,255]	10 进制数
	位/双字	高速计数器	HSC	[0,15]	10 进制数
	字	数据寄存器	D	[0,7999]	10 进制数
	字	数据寄存器	R	[0,29999]	10 进制数
系统软元件	位	特殊继电器	SM	[0,4095]	10 进制数
	字	特殊寄存器	SD	[0,4095]	10 进制数
变址寄存器	字	变址寄存器	[D]	[0,7999]	10 进制数
	字	变址寄存器	V	[0,7]	10 进制数
	双字	长变址寄存器	Z	[0,7]	10 进制数
嵌套	位	嵌套	N	[0,7]	10 进制数
指针	-	指针	P	[0,4095]	10 进制数
常数	-	10 进制常数	K	-	10 进制数
	-	16 进制常数	H	-	16 进制数
	单精度浮点	实数常数	E	-	-

2.1 用户软元件

2.1.1 输入继电器（X）

输入继电器代表 PLC 外部输入信号状态的原件，通过输入 X 端口来检测外部信号状态，0 代表外部信号开路，1 代表外部信号闭合。

用程序指令方法，不能修改输入继电器的状态，其节点信号（常开型、常闭型）在用户程序中都可无限次使用。

继电器信号以 X0、X1、.....X7、X10、X11.....等符号标识，其序号是以 8 进制方式编号。

在有接入扩展模块的情况下，扩展的 X 点也将使用 X 点作为输入信号状态的元件，所占用的 X 点起始位 PLC 所使用 X 点之后以 0 为尾数的 X 点，比如 PLC 占用 17-24 个 X 点（X0-X21、X0-X27）,这时将会以 X30 为起始点开始存放扩展模块的 X 点。

2.1.2 输出继电器（Y）

输出继电器是直接关联到外部用户控制装置的硬件端口的软元件，在逻辑上与 PLC 的物理输出端口一一对应。PLC 每次扫描完用户程序后，会将 Y 继电器的元件状态传送到 PLC 的硬件端口上，0 表示输出端口开路；1 表示输出端口闭合。

Y 继电器编号以 Y0、Y1、.....Y7、Y10、Y11.....等符号标识，其序号是以 8 进制方式编号，Y 继电器元件可在用户承诺需中无限次使用。

硬件上，根据输出元件的不同，可分为继电器型、晶体管型、固态继电器型等，若有输出扩展模块端口，按照由主模块开始，依次序进行编号。

2.1.3 内部继电器（M）

辅助继电器 M 元件用作用户程序执行过程中的中间变量，如同实际电控系统中的辅助继电器，用于状态信息的传递，也可将多个 M 变量组合成字变量使用，M 变量与外部端口没有直接的联系，但可通过程序语句将 X 复制到 M，或将 M 复制到 Y 的方式与外界发生联系，一个 M 变量可无限次使用。

辅助继电器 M 以 M0、M1.....、M7999 等符号标识，其序号是以 10 进制方式编号。

2.1.4 状态继电器（S）

状态继电器 S 用于步进程序的设计和执行处理，利用 STL 步进指令控制步进状态 S 的转移，简化编程设计。若没有采用 STL 编程方式，S 可当作 M 变量来使用。状态 S 变量以 S0、S1.....S4095 等符号标识，其序号是以 10 进制方式编号。

2.1.5 定时器（T）

定时器 T 相当于继电器系统中的时间继电器，用于完成定时功能。定时器为加法运算式，定时器时限到时，当前值与设置值则为相同的值。

定时器的线圈变为 ON 时开始计测。定时器的当前值与设置值一致时将变为时限到，定时器触点将变为 ON。将定时器的线圈置 OFF 时当前值将变为 0，定时器的触点也将变为 OFF。

定时器的 T 值范围是 0-32767。

执行定时器的线圈（OUT T 指令）时，进行定时器线圈的 ON/OFF、当前值的更新及触点的 ON/OFF 处理。

软元件编号	定时器	软元件编号	定时器
T0~T191	100ms 定时器	T246~T249	1ms 累计型定时器
T192~T199	100ms 子程序定时器（累计型，子程序中使用，即使子程序未被调用，依然会更新）	T250~T255	10ms 累计型定时器
		T256~T383	1ms 定时器
T200~T245	10ms 定时器	T384~T511	0.1ms 定时器

（1）通用定时器（T0~T245）

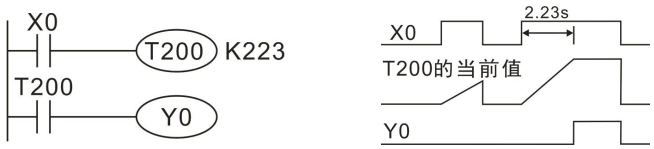


图 a

如上图所示：X0 的常开触点接通时，T200 的当前值计数器从零开始计时，对 10ms 时钟脉冲进行累加计数。当前值等于设定值 223 时，定时器的常开触点接通，常闭触点断开，即 T200 的输出触点在其线圈被驱动 2.23s 后动作。X0 的常开触点断开后，T200 因为线圈断电而被复位，复位后它的常开触点断开，常闭触点接通，当前值恢复为零。

（2）累计型定时器（T246~T255）

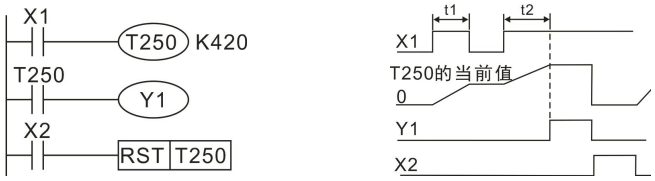
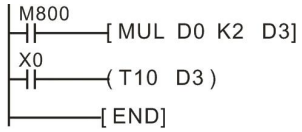


图 b

如图 b 中的 X1 常开触点接通时，T250 的当前值计数器对 10ms 时钟脉冲进行累加计数。X1 的常开触点断开或停止时停止计数，当前值保持不变。当 X1 的常开触点再次接通时继续计数，累计时间 t1+t2 为 4.2s 时，当前值等于设定值 420，T250 的常开触点接通，常闭触点断开。X2 的常开触点接通时，T250 复位（因为累计型定时器的线圈断电时不会复位，需要用 X2 的常开触点和复位指令使 T250 强制复位）。

（3）设定值

定时器的时间可采用程序存储器内的常数（K、H）作为设定值，也可用数据寄存器（D）的内容进行间接指定。



PLC 上电后进行乘法运算，D3=D0*2。将 D3 的数据作为 T10 的定时时间值。

2.1.6 计数器（C）

计数器用于完成计数功能，每个计数器含有线圈、接点、计时时值寄存器，每当计数器线圈的的驱动信号由 OFF→ON 时，计数器读数值增加 1，若计时值达到预设的时间值时，其接点动作，a 接点（NO 接点）闭合，b 接点（NC 接点）断开；若清除计时值，输出 a 接点即断开， b 接点（NC 接点）闭合。部分计时器的具有掉电保持、累计等特性，重新上电后仍维持掉电前的数值。

计数器以 C0, C1,…,C255 进行标识，顺序按 10 进制编号。

计数器（C）为 16 位计数器。

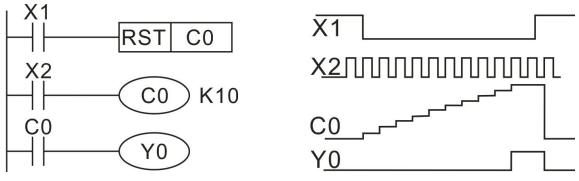


图 c

16 位加计数器的设定值为 1~32767。如图 c 的加计数器的工作过程所示，图中 X1 的常开触点接通后，C0 被复位，它对应的位存储单元被置 0，C0 的常开触点断开，常闭触点接通，同时其计数当前值置 0。X2 提供计数输入信号，当计数器的复位输入电路断开，计数输入电路由断开变为接通（即计数脉冲的上升沿）时，计数器 C0 的当前值加 1。在 10 个计数脉冲后，C0 的当前值等于设定值 10，它对应的位存储单元被置 1，此时 Y0 触点接通。再来计数脉冲时当前值不变，直到复位输入信号接通，计数器的当前值被置为 0。

2.1.7 长计数器（LC）

长计数器（LC）与计数器（C）使用方法大体上相同，但相较于计数器（C），长计数器（LC）为 32 位寄存器，可以计数的值的范围更大。

长计数器以 LC0，LC1,…,LC255 进行标识，顺序按 10 进制编号。

2.1.8 高速计数器（HSC）

高速计数器（HSC）是通过外部输入高速脉冲信号，来进行计数所使用的软元件，HSC 为 32 位寄存器。

相应的参数配置可通过：[工程管理]→[参数]→[高速计数器配置]来进行配置：

2.1.9 数据寄存器（D & R）

寄存器用于数据的运算和存储，如对定时器、计数器、模拟量参数的运算和存储等，每个寄存器的宽度为 16bit。若采用 32bit 指令，则自动将相邻的寄存器组成为 32bit 寄存器使用，地址较低的为低字节，而地址较高的为高字节。

D 寄存器的地址范围：D0~D7999；R 寄存器的地址范围：R0~R29999。

我司系列 PLC 多数指令中参与运算的数据是按有符号数进行处理的，对于 16bit 的寄存器，bit15 为符号位（0 表示正数，1 表示负数）；对于 32bit 的寄存器，高字节的 bit15 为符号位，数值范围为[-32768,+32767]。

当需要处理 32bit 的数据时，可将相邻的 2 个 D 寄存器组成为 32bit 双字，例如以 32bit 格式访问 D100 时，此时将高地址 D101 寄存器作为高字，同时将高字节的 bit15 作为双字的符号位，可处理-2147483648~2147483647 的数值。

2.2 系统软元件

2.2.1 特殊继电器（SM）

特殊继电器 SM 是可编程控制器内部确定规格的内部继电器，因此不能像普通的内部继电器那样用于程序中，可根据需要置 ON/OFF 以控制 PLC。

具体内容请参阅附录 1“特殊继电器（SM）一览表”。

2.2.2 特殊寄存器（SD）

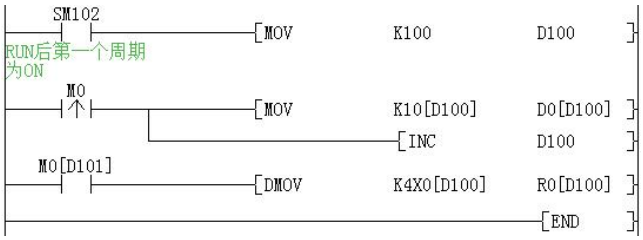
特殊寄存器 SD 是可编程控制器内部确定规格的内部寄存器，因此不能像普通的内部寄存器那样用于程序中，可根据需要写入相应数据以控制 PLC。

具体内容请参阅附录 2“特殊寄存器（SD）一览表”。

2.3 变址寄存器

2.3.1 变址寄存器（[D]）

变址寄存器用于对软元件的变址修饰，[D]变址寄存器实际与数据寄存器 D 是同一个，范围 D0~D7999，输入方法如下，直接在软元件后面加入[D]即可：



支持的变址修饰的软元件如下：

常数 K、H 加变址修饰，如 D0 = 10，K10[D0] 结果 = 10 + 10 = 20。

常数 E、字符串不支持变址修饰。

数据软元件加变址修饰，如 D0 = 10，D10[D0]的结果是 D20 的值。即使 D10[D0]使用在双字指令中，双字取值是取 D20（低字）、D21（高字）的值。

位软元件加变址修饰，如 D0 = 10，M0[D0]的结果是 M10 的值。

位组合成字加变址修饰，如 D0 = 10，K4M10[D0]先取 M10 偏移 10 个地址，再进行组合，结果等同于 K4M20。

是否可以使用变址修饰取决于各个指令是否支持该格式，可查看各个指令的可使用软元件说明中的“偏移修饰”。

2.4 指针

2.4.1 指针（P）

指针是跳转指令（CJ 指令）中使用的软元件。

目前 CALL 指令直接使用子程序名称来进行调用，不再使用 P 指针。

2.5 常数

以下对常数进行说明。

2.5.1 10 进制常数（K）

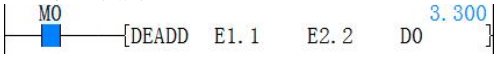
[K]是表示 10 进制整数的符号，以 K□进行指定（例：K123），主要用于指定定时器或计数器的设定值或应用指令操作数中的数值。16bit 指令中，常数 K 的取值范围为[-32768,32767]；32bit 指令中，常数 K 的取值范围为 -2,147,483,648~2,147,483,647。

2.5.2 16 进制常数（H）

[H]是 16 进制数的表示符号，以 H□进行指定（例：H123），主要用于指定应用指令的操作数的数值。常数 H 的取值范围为 0000~FFFF；32bit 指令中，常数 K 的取值范围为 0000,0000~FFFF,FFFF。

2.5.3 实数常数（E）

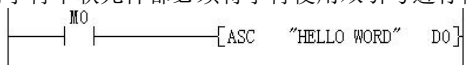
[E]是单精度浮点数表示符号，以 E□进行指定（例：E1.23），主要用于指定应用指令的操作数的数值，单精度浮点数 E 的取值范围为±1.175495*10-38~±3.402823*10+38（±1.175495 E-38~±3.402823 E+38）以及 0（有效位 7 位）。



（其中地址占用了 D1、D0）

2.5.4 字符串常数

字符串常数是指定字符串的软元件，只支持 ASCII 码字符集，任一个字符串都以 NULL 字符（00H）作为字符串的结尾。使用字符串软元件都必须将字符使用双引号进行修饰，如下将字符串转为 ASCII 字符填写到以 D0 开始的软元件中：



2.6 掉电保持设置

用户在软元件范围内，自由配置掉电保存范围。常数配置位于：[工程管理]→[参数]→[PLC 参数]→[软元件锁存]。

	标记	进制	点数	软元件范围	启用锁存	锁存起始	锁存结束	可锁存范围
输入继电器	X	8	1024	0~1777				
输出继电器	Y	8	1024	0~1777				
辅助继电器	M	10	8000	0~7999	☑	500	7999	0~7999
状态继电器	S	10	4096	0~4095	☑	500	4095	0~4095
数据继电器	D	10	8000	0~7999	☑	200	7999	0~7999
数据继电器	R	10	30000	0~29999	☑	0	29999	0~29999
定时器	T	10	512	0~511	☑	200	511	0~511
16位计数器	C	10	256	0~255	☑	100	255	0~255
32位计数器	LC	10	256	0~255	☑	100	255	0~255
高速计数器	HSC	10	16	0~15	☑	0	15	0~15

注：X 和 Y 寄存器不支持掉电保存功能。

2.7 软元件的特殊使用方式

1、用位组合成字

格式：KnB

K 为固定字符。

n 取值 1~8，表示取(n*4)个位组合成字，如 K4M0，将 M0~M15 组合字。

B 是位软元件编号。

示例：同时置位 M0~M31 共 32 个位。

注：KnB 类型也可使用变址修饰。

2、字中取位

格式：D.b

D 为数据软元件 D 的编号（R 不可用）。

b 为需要取的位编号，十六进制，取值范围 0~F。

示例：D2000 中的 bit14 置位，输出 Y0

注：D.b 类型也可使用变址修饰。

3 顺控程序指令

3.1 触点指令

3.1.1 运算开始、串联连接、并联连接

LD、LDI、AND、ANI、OR、ORI

• LD：常开触点运算开始指令，LDI：常闭触点运算开始指令。

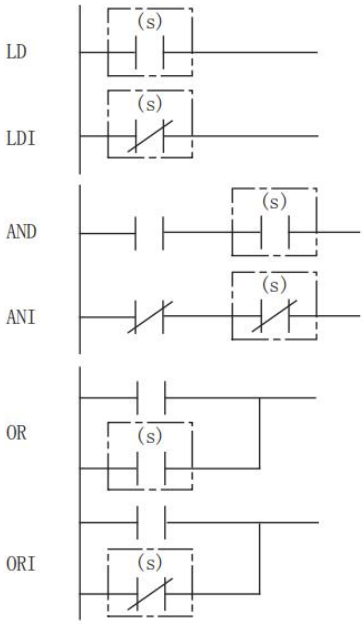
对(s)中指定的软元件的 ON/OFF 信息进行提取，作为运算结果。

• AND：常开触点串联连接指令，ANI：常闭触点串联连接指令。

对(s)中指定的软元件的 ON/OFF 信息进行提取，并与当时为止的运算结果进行 AND 运算，作为运算结果。

• OR：1 个常开触点的并联连接指令,ORI：1 个常闭触点的并联连接指令。

对(s)中指定的软元件的 ON/OFF 信息进行提取，并与当时为止的运算结果进行 OR 运算，作为运算结果。



内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)
(s)	作为触点使用的软元件	---	位	ANY_BOOL

使用的软元件

指令	参数	软元件																								偏移修饰
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]
LD	参数 1	●	●	●	●	●	●	●	●	●	●															●
LDI	参数 1	●	●	●	●	●	●	●	●	●	●															●
AND	参数 1	●	●	●	●	●	●	●	●	●	●															●
ANI	参数 1	●	●	●	●	●	●	●	●	●	●															●
OR	参数 1	●	●	●	●	●	●	●	●	●	●															●
ORI	参数 1	●	●	●	●	●	●	●	●	●	●															●

功能

■LD、LDI

• LD 指令是常开触点运算开始指令，LDI 指令是常闭触点运算开始指令，对指定软元件的 ON/OFF 信息*1 进行提取，作为运算结果。

*1：字软元件的位指定时，根据指定位的 1/0 而 ON/OFF。

■AND、ANI

• AND 指令是常开触点串联连接指令，ANI 指令是常闭触点串联连接指令，对指定位软元件的 ON/OFF 信息*1 进行提取，并与当时为止的运算结果进行 AND 运算，将该值作为运算结果。

*1：字软元件的位指定时，根据指定位的 1/0 而 ON/OFF。

• 串联触点的个数无限制，该指令可以连续任意次使用。

• OUT 指令后，通过触点 OUT 至其他线圈称为级联输出，只要顺序不错，可重复任意次。

■OR、ORI

• OR 指令是 1 个常开触点的并联连接指令，ORI 指令是 1 个常闭触点的并联连接指令，对指定软元件的 ON/OFF 信息*1 进行提取，并与当时为止的运算结果进行 OR 运算，将该值作为运算结果。

*1：字软元件的位指定时，根据指定位的 1/0 而 ON/OFF。

• OR、ORI 指令从该指令所在的步开始，对之前的 LD、LDI 指令所在的步并联连接。

• 并联连接的次数无限制。

要点

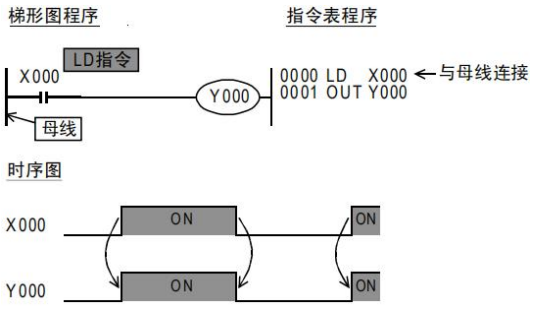
字软元件的位指定时，位的指定是以 16 进制数进行。（例如，D0 的 b11 将变为“D0.B”）

错误码

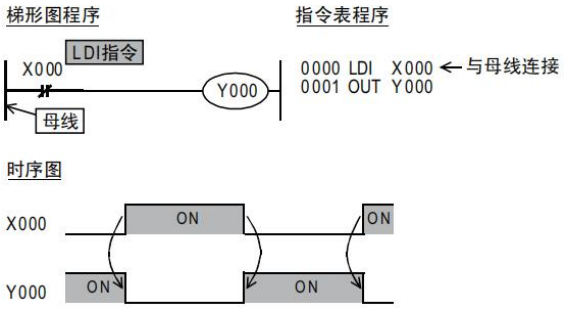
错误码	内容
4085H	(S) 读取地址超出软元件范围。

示例

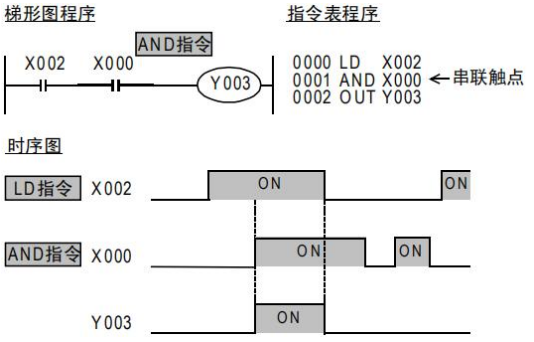
1、LD 指令（a 触点的逻辑运算开始）



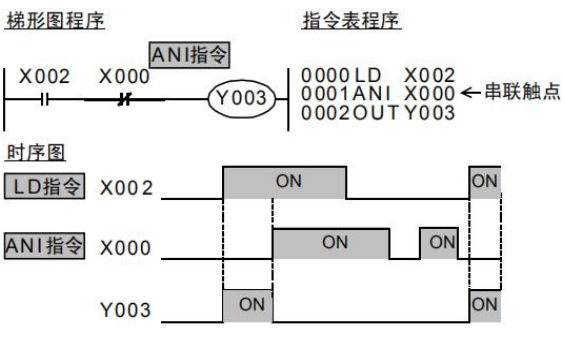
2、LDI 指令（b 触点的逻辑运算开始）



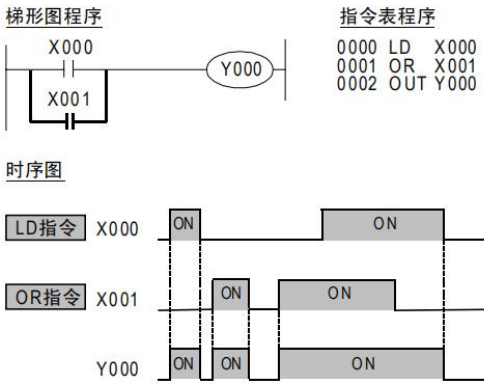
3、AND 指令（串联 a 触点）



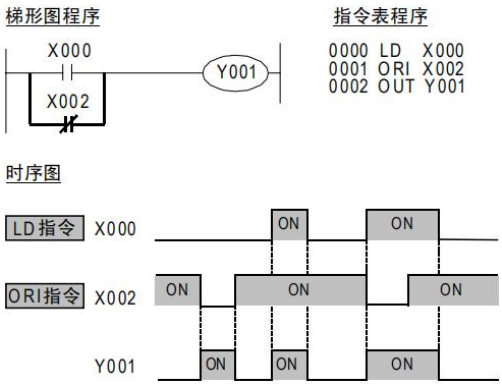
4、ANI 指令（串联 b 触点）



5、OR 指令（并联 a 触点）



6、ORI 指令（并联 a 触点）



7、偏移修饰

LD、LDI、AND、ANI、OR、ORI 指令中使用的软元件，都可用 D 数据软元件进行变址修饰（状态寄存器 S 不可修饰）。

变址修饰中可以使用 D0~D7999。

使用的软元件为输入(X)、输出(Y)的时候，变址寄存器的值换算成 8 进制数后进行加法运算。

例：

D0 的值为 10 的时候，由 X012 决定 LD 触点 ON(导通)/OFF(不导通)。

8、数据寄存器中的位指定

LD、LDI、AND、ANI、OR、ORI 指令中使用的软元件中，可指定数据寄存器(D)的位。

执行数据寄存器的位指定时，请在数据寄存器(D)的编号后输入“.”，然后接着输入位编号(0~F)。可使用的数据寄存器进行位指定，但仅 16 位数据寄存器有效。

请从低位开始按照 0,1,2,…9,A,B,…F 的顺序指定位编号。

例：

D0 的第 3 位决定 LD 触点 ON(导通)/OFF(不导通)。

3.1.2 脉冲运算开始、脉冲串联连接、脉冲并联连接

LDP、LDF、ANDP、ANDF、ORP、ORF

• LDP：上升沿脉冲运算开始指令。

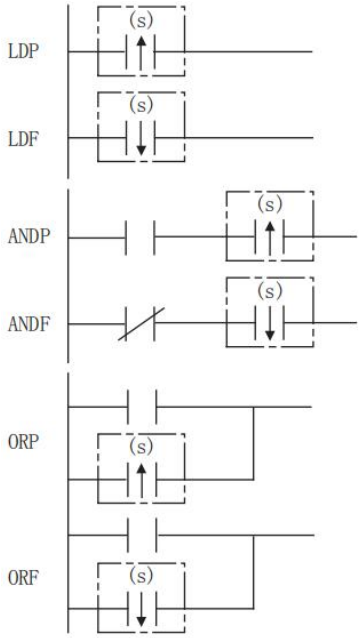
仅在(s)中指定的位软元件的上升沿时(OFF→ON)导通。

• LDF：下降沿脉冲运算开始指令。

仅在(s)中指定的位软元件的下降沿时(ON→OFF)导通。

• ANDP：上升沿脉冲串联连接指令，ANDF：下降沿脉冲串联连接指令。至当时为止的运算结果与(s)中指定的位软元件进行 AND 运算，作为运算结果。

• ORP：上升沿脉冲并联连接指令，ORF：下降沿脉冲并联连接指令。至当时为止的运算结果与(s)中指定的位软元件进行 OR 运算，作为运算结果。



内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	作为触点使用的软元件	--	位	ANY_BOOL	BOOL

使用的软元件

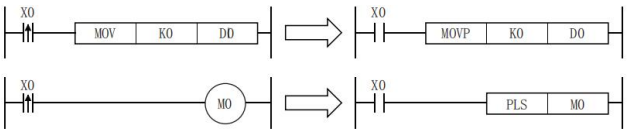
指令	参数	软元件																								偏移修饰
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]
LDP	参数 1	●	●	●	●	●	●	●	●	●	●															
LDF	参数 1	●	●	●	●	●	●	●	●	●	●															
ANDF	参数 1	●	●	●	●	●	●	●	●	●	●															
ANDF	参数 1	●	●	●	●	●	●	●	●	●	●															
ORP	参数 1	●	●	●	●	●	●	●	●	●	●															
ORF	参数 1	●	●	●	●	●	●	●	●	●	●															

功能

■LDP、LDF

• LDP 指令是上升沿脉冲运算开始指令，仅在指定位软元件的上升沿时(OFF→ON)导通。字软元件的位指定时，仅在指定位由 0→1 变化时导通。仅 LDP 指令的情况下，与 ON 中执行指令的脉冲化指令(□P)相同。

将使用了 LDP 指令的回路置换为不使用 LDP 指令的回路时，其情况如下所示。



• LDF 指令是下降沿脉冲运算开始指令，在指定位软元件的下降沿时(ON→OFF)导通。字软元件的位指定时，仅在指定位由 1→0 变化时导通。

■ANDP、ANDF

• ANDP 指令是上升沿脉冲串联连接指令，ANDF 指令是下降沿脉冲串联连接指令，与至当时为止的运算结果进行 AND 运算并作为运算结果。ANDP 指令、ANDF 指令中使用的 ON/OFF 信息如下表所示。

ANDP、ANDF 中指定的软元件		ANDP 的状态	ANDF 的状态
位软元件	字软元件的位指定		
OFF→ON	0→1	ON	OFF
OFF	0	OFF	OFF
ON	1	OFF	OFF
ON→OFF	1→0	OFF	ON

■ORP、ORF

• ORP 指令是上升沿脉冲并联连接指令、ORF 指令是下降沿脉冲并联连接指令，与至当时为止的运算结果进行 OR 运算并作为运算结果。ORP 指令、ORF 指令中使用的 ON/OFF 信息如下表所示。

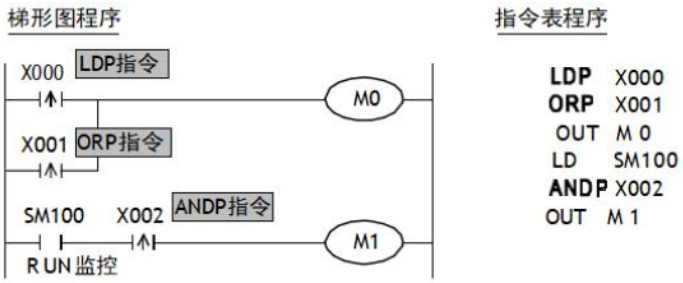
ORP、ORF 中指定的软元件		ORP 的状态	ORF 的状态
位软元件	字软元件的位指定		
OFF→ON	0→1	ON	OFF
OFF	0	OFF	OFF
ON	1	OFF	OFF
ON→OFF	1→0	OFF	ON

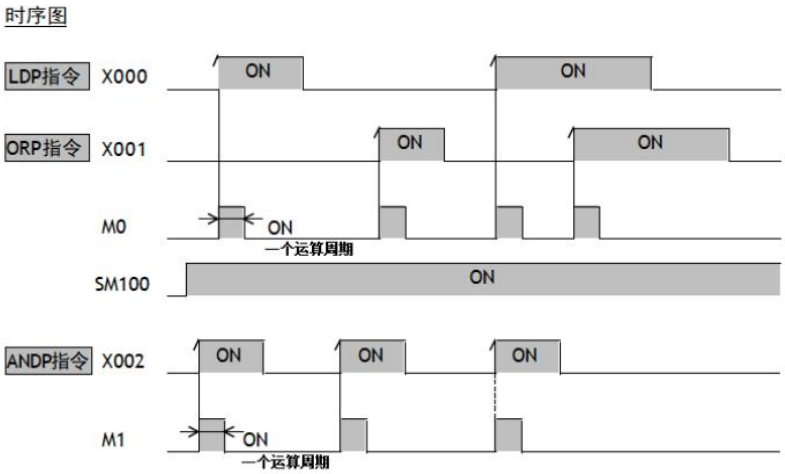
错误码

没有运算出错。

示例

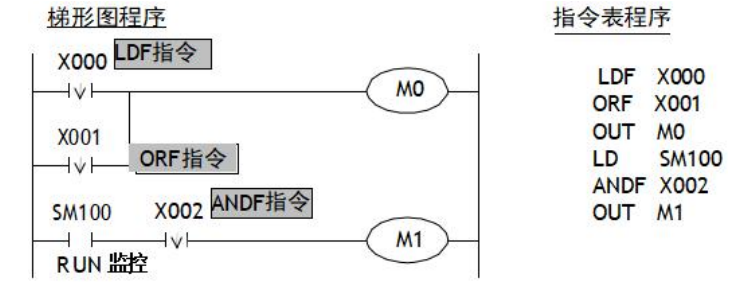
1、LDP、ANDP、ORP 指令（检测到上升沿时运算开始、串联连接、并联连接）

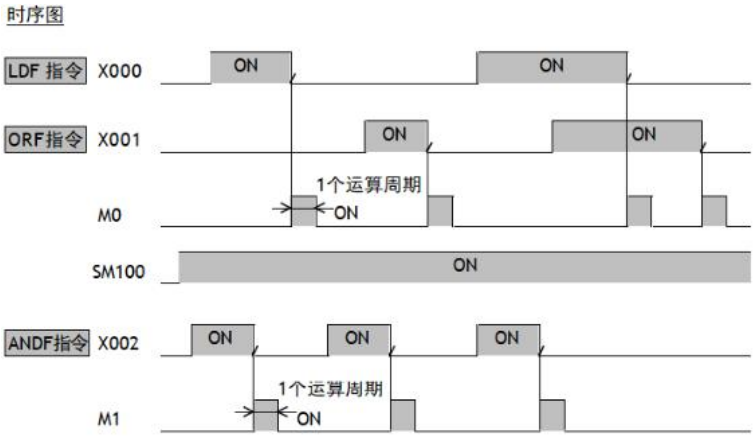




在上图中，X000-X002 从 OFF 变成 ON 时，M0 或 M1 只维持 1 个运算周期为 ON。

2、LDF、ANDF、ORF 指令（检测到下降沿时运算开始、串联连接、并联连接）





在上图中，X000-X002 从 ON 变成 OFF 时，M0 或 M1 只维持 1 个运算周期为 ON。

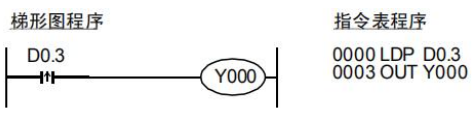
3、数据寄存器（D）的位指定

LDP、LDF、ANDP、ANDF、ORP、ORF 指令用的软元件中，可指定数据寄存器（D）的位。

执行数据寄存器的位指定时，请在数据寄存器（D）的编号后输入“.”，然后输入位编号（0-F）。可使用的数据寄存器进行位指定，但仅 16 位数据寄存器有效。

请从低位开始按照 0,1,2,，…9，A，B，…F 的顺序指定位编号。

例：



D0 的第 3 位决定从 OFF 变 ON 时，LDP 触点 ON（导通）/OFF（不导通）。

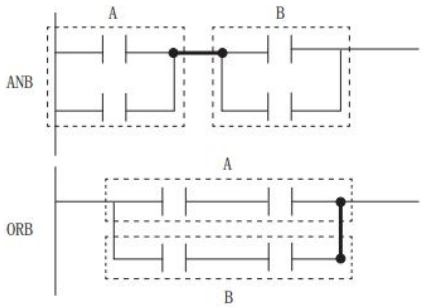
3.2 结合指令

3.2.1 梯形图块串联连接、并联连接

ANB、ORB

进行 A 块与 B 块的 AND 运算或 OR 运算，并作为运算结果。

梯形图



功能

■ANB

- 进行 A 块与 B 块的 AND 运算，并作为运算结果。
- ANB 指令的符号不是触点符号，而是连接符号。

■ORB

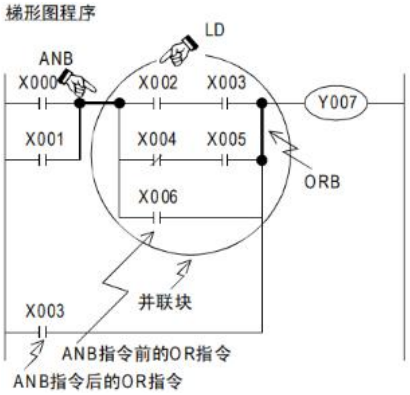
- 进行 A 块与 B 块的 OR 运算，并作为运算结果。
- ORB 指令进行 2 触点以上的回路块的并联连接。仅 1 个触点的并联连接使用 OR 指令、ORI 指令，无需 ORB 指令。
- ORB 指令的符号不是触点符号，而是连接符号。

错误码

没有运算出错。

示例

梯形图程序



指令表程序

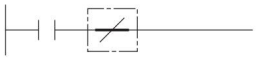
```
0000 LD  X000
0001 OR  X001
0002 LD  X002      ←┐分支起点
0003 AND X003       │
0004 LDI X004      ←┘
0005 AND X005
0006 ORB           ←┐并联块的结束
0007 OR  X006      ←┘
0008 ANB           ←与前面的回路串联连接
0009 OR  X003
0010 OUT Y007
```

3.2.2 运算结果取反

INV

对 INV 指令之前为止的运算结果进行取反。

梯形图



功能

对 INV 指令之前为止的运算结果进行取反。

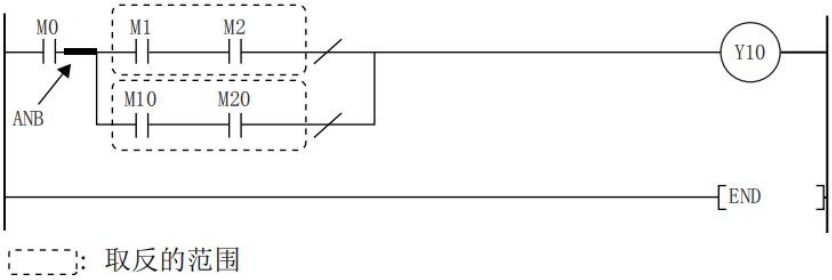
INV 指令之前为止的运算结果	执行 INV 指令后的运算结果
OFF	ON
ON	OFF

错误码

没有运算出错。

要点

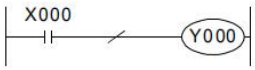
- INV 指令是之前为止的运算结果执行动作，因此应与 AND 指令在同一位置使用。INV 指令不能在 LD 指、OR 指令的位置使用。
- 并用 INV 指令及 ANB 指令是梯形图动作的情况下，应注意取反的范围。



示例

INV 指令（运算结果的反转）

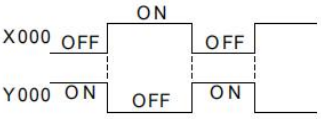
梯形图程序



指令表程序

```
0 LD  X000
1 INV
2 OUT Y000
```

时序图



INV指令执行前的运算结果	INV指令执行后的运算结果
OFF	ON
ON	OFF

反转

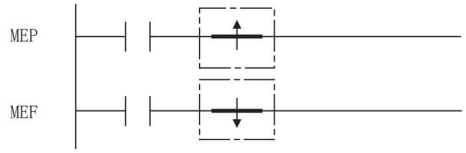
3.2.3 运算结果脉冲化

MEP、MEF

• MEP：MEP 指令之前的运算结果为上升沿时变为 ON，上升沿以外的情况下，变为 OFF。

• MEF：MEF 指令之前的运算结果为下降沿时变为 ON，下降沿以外的情况下，变为 OFF。

梯形图



功能

■MEP

• 在 MEP 指令之前的运算结果为上升沿时(OFF→ON)，变为 ON(导通状态)。MEP 指令之前的运算结果为上升沿以外的情况下，变为 OFF(非导通状态)。

• 使用 MEP 指令时，对多个触点进行了串联连接的情况下，脉冲化处理将易于进行。

■MEF

• 在 MEF 指令之前的运算结果为下降沿时(ON→OFF)，变为 ON(导通状态)。MEF 指令之前的运算结果为下降沿以外的情况下，变为 OFF(非导通状态)。

• 使用 MEF 指令时，对多个触点进行了串联连接的情况下，脉冲化处理将易于进行。

错误码

没有运算出错。

要点

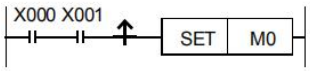
• 对于 MEP 指令、MEF 指令，如果通过子程序及 FOR~NEXT 指令等进行变址修饰后的触点的脉冲化，将可能无法正常动作。

• MEP 指令、MEF 指令是以之前为止的运算结果执行动作，因此应与 AND 指令在同一位置使用。对于 MEP 指令、MEF 指令，不能在 LD 指令、OR 指令的位置使用。

示例

1、MEP 指令（运算结果的上升沿时为 ON）

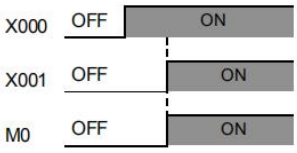
梯形图程序



指令表程序

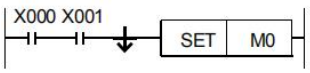
```
0  LD   X000
1  AND  X001
2  MEP
3  SET  M0
```

时序图



2、MEF 指令（运算结果的下降沿时为 ON）

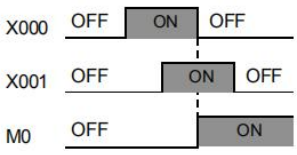
梯形图程序



指令表程序

```
0  LD   X000
1  AND  X001
2  MEF
3  SET  M0
```

时序图

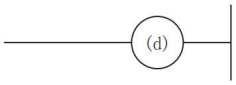


3.3 输出指令

3.3.1 OUT 指令（定时器、计数器除外）

将 OUT 指令之前的运算结果输出到指定的软元件中。

梯形图



内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	ON/OFF 的软元件编号	---	位	ANY_BOOL	BOOL

使用的软元件

指令	参数	软元件																								偏移修饰
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]
OUT	参数 1	●	●	●	●	●					●					●	●				●	●				●

*1：使用 T 的情况参照 OUT T 指令；

*2：使用 C、LC、HSC 的情况参照 OUT C 指令；

*3：使用 S 软元件时无法使用偏移修饰。

功能

将 OUT 指令之前的运算结果输出到指定的软元件中。

条件	运算结果	线圈/指定位
使用位软元件时	OFF	OFF
	ON	ON
使用字软元件时	OFF	0
	ON	1

错误码

错误码	内容
4086H	（d）写入地址超出软元件范围。

示例

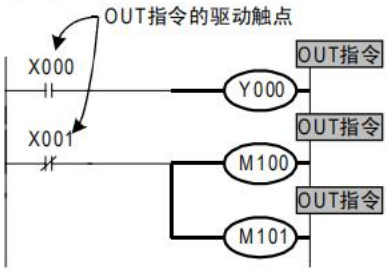
1、使用位软元件时

使用 OUT 指令编写的软元件，根据驱动触点的状态执行 ON/OFF，并联的 OUT 命令能够多次连续使用。

下面的程序举例中，接着 OUT M100 的 OUT M101 就是这个意思。

但是，对同一软元件编号，使用多个 OUT 指令时，会变成双重输出（双线圈），请注意。

梯形图程序

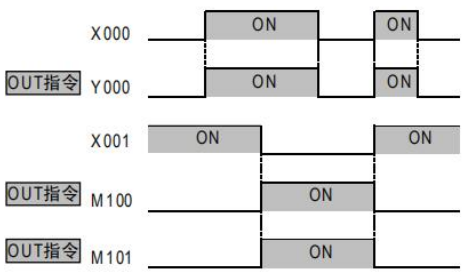


指令表程序

```
0000 LD   X000
0001 OUT Y000
0002 LDI  X001
0003 OUT M100
0004 OUT M101
```

会自动管理程序步号。

时序图



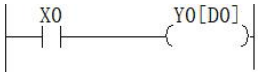
2、偏移修饰

OUT 指令中使用的软元件，都可用 D 数据软元件进行变址修饰（状态寄存器 S 不能修饰）。

变址修饰中可使用 D0~D7999。

使用的软元件为输入（X）、输出（Y）的时候，变址寄存器的值换算成 8 进制数后进行加法运算。

例：



D0 的值为 10 的时候，当 X0 为 ON（导通）时，Y12 触点 ON（导通）。

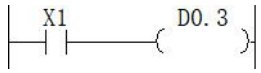
3、数据寄存器中的位指定

OUT 指令中使用的软元件中，可指定数据寄存器（D）的位。

执行数据寄存器的位指定时，请在数据寄存器(D)的编号后输入“.”，然后接着输入位编号(0~F)。可使用的数据寄存器进行位指定，但仅 16 位数据寄存器有效。

请从低位开始按照 0,1,2,…9,A,B,…F 的顺序指定位编号。

例:

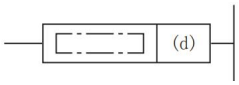


左边的例子中，X1 状态决定 D0 中的第三位 ON(导通)/OFF(不导通)。

3.3.2 SET 指令

执行指令变为 ON 时，(d)中指定的软元件将变为下述状态。

- 位软元件：将线圈、触点置为 ON。
- 字软元件的位指定：将指定位置为 1。



内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	设置(ON)的位软元件编号/字软元件的位指定	---	位	ANY_BOOL	BOOL

使用的软元件

指令	参数	软元件																								偏移修饰
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]
SET	参数 1	●	●	●	●	●					●															●

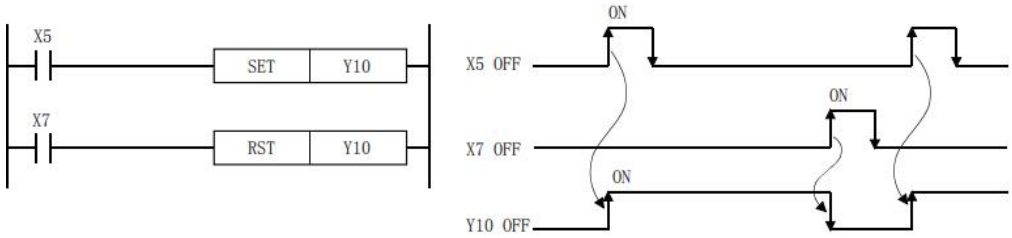
*1：使用 S 软元件时无法使用偏移修饰。

功能

执行指令为 ON 时指定软元件将变为下述状态。

软元件	软元件状态
位软元件	将线圈、触点置为 ON
字软元件的位指定	将指定位置为 1

对于置为 ON 的软元件，即使执行指令变为 OFF 也将保持 ON 不变。对于通过 SET 指令置为 ON 的软元件，可通过 RST 指令置为 OFF。



注意事项

对于输出继电器（Y），在同一运算中执行了 SET 指令和 RST 指令的情况下，将输出接近于 END 指令（程序的结束）的指令结果。

错误码

错误码	内容
4086H	(d)在使用偏移的情况下偏移后的地址超出软元件范围。

示例

1、使用位软元件时

并联的 SET 指令，可连续使用多次。下面的程序举例中，SET Y000 后接着 RST Y000 的程序就是这个情况。

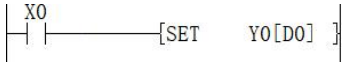
2、偏移修饰

SET 指令中使用的软元件，都可以用 D 数据软元件进行变址修饰（状态寄存器 S 不能修饰）。

变址修饰中可以使用 D0~D7999。

使用的软元件为输入(X)、输出(Y)的时候，变址寄存器的值换算成 8 进制数后进行加法运算。

例:



D0 的值为 10 的时候，X0 为 ON(导通)时，Y12 触点 ON(导通)，X0 为 OFF(不导通)，Y12 触点不变。

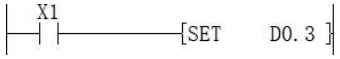
3、数据寄存器中的位指定

SET 指令中使用的软元件中，可以指定数据寄存器(D)的位。

执行数据寄存器的位指定时，请在数据寄存器 (D)的编号后输入“.”，然后接着输入位编号 (0~F)。可使用的数据寄存器进行位指定，但仅 16 位数据寄存器有效。

请从低位开始按照 0,1,2,…9,A,B,…F 的顺序指定位编号。

例:

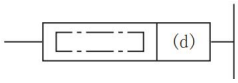


左边的例子中，X1 状态 ON(导通)， D0 中的第三位 ON(导通)。X1 状态 OFF(不导通)，D0 中的第三位状态不变。

3.3.3 RST 指令

RST 输入变为 ON 时，(d) 中指定的软元件将变为下述状态。

- 位软元件：将线圈、触点置为 OFF。
- 定时器、计数器：将当前值置为 0，将线圈、触点置为 OFF。
- 字软元件的位指定：将指定位置为 0。
- 字软元件、模块访问软元件、变址寄存器：将内容置为 0。



内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	复位的位软元件编号/字软元件的位指定或复位的字软元件编号	---	位/字/双字	ANY_ELEMENTARY	BOOL

使用的软元件

指令	参数	软元件																							偏移修饰
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]
RST	参数 1	●	●	●	●	●	●	●	●	●	●					●	●	●	●	●	●	●				●

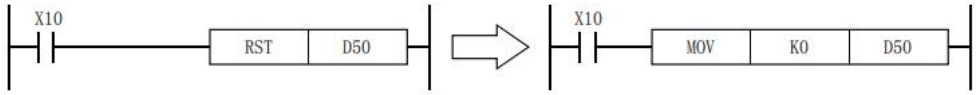
*1：使用 S 软元件时无法使用偏移修饰。

功能

执行指令为 ON 时指定软元件将变为下述状态。

软元件	软元件状态
位软元件	将线圈、触点置为 ON
定时器、计数器	将当前值置为 0，将线圈、触点置为 OFF
字软元件的位指定	将指定位置为 0
字软元件	将内容置为 0

- 执行指令为 OFF 的情况下，软元件的状态不变化。
- 对于通过 RST 指令指定字软元件时的功能，与下述梯形图相同。



注意事项

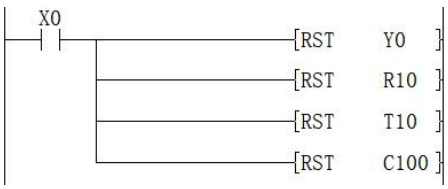
对于定时器、计数器，在 RST 指令被跳转的程序、子程序和中断程序中执行的情况下，定时器和计数器可能会保持复位后的状态不变，定时器和计数器不动作。

错误码

错误码	内容
4086H	(d)写入地址超出软元件范围。

示例

1、使用位软元件



- X0 为 ON（导通）时，Y0 被置为 OFF，R10 被置为 0，T10 的字软元件被置为 0，位软元件被置为 OFF，C100 的字软元件被置为 0，位软元件被置为 OFF。
- X0 为 OFF（不导通）时，所有状态均不变。

2、偏移修饰

RST 指令中使用的软元件，都可以用 D 数据软元件进行变址修饰。（状态寄存器 S 不能修饰）

变址修饰中可以使用 D0~D7999。

使用的软元件为输入(X)、输出(Y)的时候，变址寄存器的值换算成 8 进制数后进行加法运算。

例：

D0 的值为 10 的时候，X0 为 ON(导通)时 Y12 触点 OFF(导通)，X0 为 OFF(不导通)，Y12 触点不变。

3、数据寄存器中的位指定

RST 指令中使用的软元件中，可以指定数据寄存器(D)的位。

执行数据寄存器的位指定时，请在数据寄存器(D)的编号后输入“.”，然后接着输入位编号(0~F)。可使用的数据寄存器进行位指定，但仅 16 位数据寄存器有效。

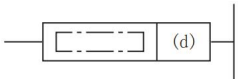
请从低位开始按照 0,1,2,…9,A,B,…F 的顺序指定位编号。

例：

左边的例子中，X1 状态 ON(导通)，D0 中的第三位 OFF(导通)。X1 状态 OFF(不导通)，D0 中的第三位状态不变。

3.3.4 下降沿输出 PLF

PLF 指令 ON→OFF 时使(d)中指定的软元件 1 个扫描 ON，ON→OFF 以外时使其为 OFF。



内容、范围、数据类型

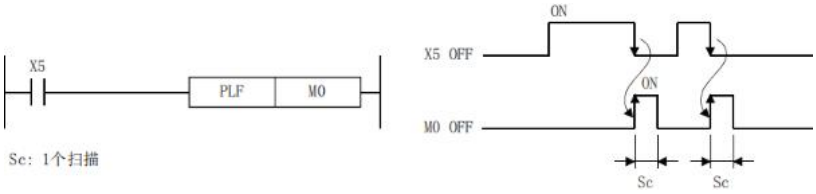
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	脉冲化的软元件	---	位	ANY_BOOL	BOOL

使用的软元件

指令	参数	软元件																							偏移修饰
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]
PLF	参数 1	●	●	●	●	●					●															

功能

• 执行指令 ON→OFF 时将指定软元件置为 ON，执行指令为 ON→OFF 以外时置为 OFF。1 个扫描中(d)中指定的软元件的 PLF 指令为 1 个的情况下，指定软元件将 1 个扫描 ON。



注意事项

• 将 PLF 指令通过 CJ 指令进行跳转，或执行的子程序未通过 CALL(P)指令调用的情况下，(d)中指定的软元件有可能 1 个扫描以上 ON，应加以注意。

错误码

无错误码。

示例

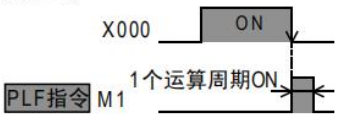
PLF 指令（下降沿的微分输出）

梯形图程序

指令表程序

```
0000  LD   X000
0001  PLF  M 1
```

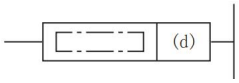
时序图



在上图中，X000 从 ON 变为 OFF 时，只有一个运算周期的 M1 为 ON。

3.3.5 上升沿输出 PLS

PLS 指令 OFF→ON 时使(d)中指定的软元件 1 个扫描 ON，OFF→ON 以外时使其为 OFF。



内容、范围、数据类型

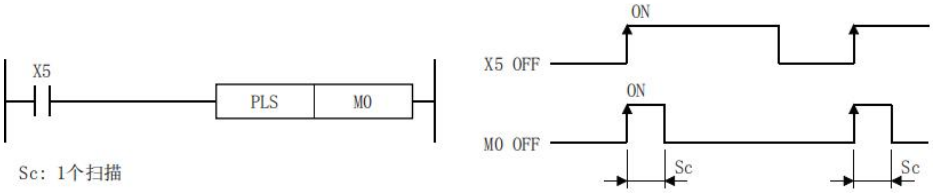
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	脉冲化的软元件	---	位	ANY_BOOL	BOOL

使用的软元件

指令	参数	软元件																							偏移修饰
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]
PLS	参数 1	●	●	●	●	●					●															

功能

• PLS 指令 OFF→ON 时使指定软元件 1 个扫描 ON，OFF→ON 以外时使其为 OFF。1 个扫描中(d)中指定的软元件的 PLS 指令为 1 个的情况下，指定软元件将 1 个扫描 ON。



注意事项

• 将 PLS 指令通过 CJ 指令进行跳转，或执行的子程序未通过 CALL(P)指令调用的情况下，(d)中指定的软元件有可能 1 个扫描以上 ON，应加以注意。

错误码

无错误码。

示例

PLS 指令（上升沿的微分输出）

在上图中，X000 从 OFF 变为 ON 时，只有一个运算周期的 M0 为 ON。

4 程序流程指令

4.1 程序跳转

CJ/条件跳转

跳转指令为 ON 时，执行同一程序文件内指定的指针编号的程序。

-[CJ(P)　(P)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(P)	跳转目标的指针编号	P0~P4095	软元件名	POINTER	-----

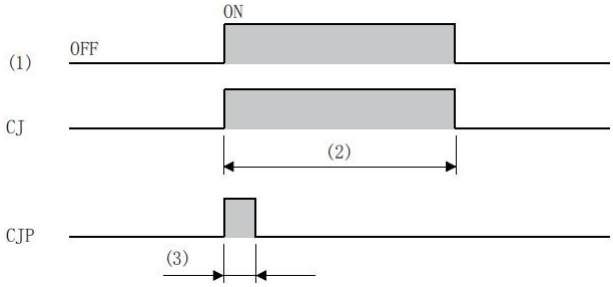
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展	其他
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP	P
CJ	参数 1																									●	●

功能

■CJ(P)

- 执行指令为 ON 时，执行指定的指针编号的程序。
- 执行命令为 OFF 时，执行下一步的程序。



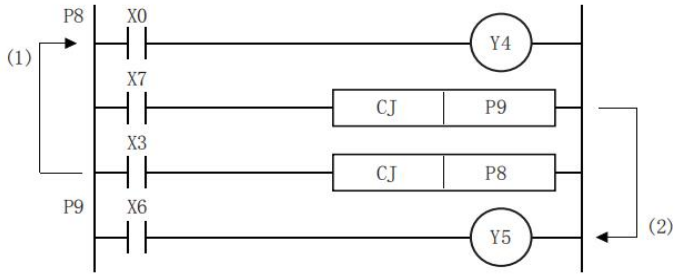
(1)：执行指令；

(2)：每个扫描执行；

(3)：1 个扫描执行。

注意事项

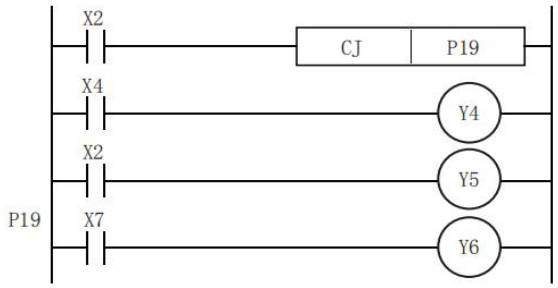
- 将定时器的线圈置为 ON 后，通过 CJ(P)指令对线圈为 ON 的定时器进行了跳转的情况下，将无法正常进行计测。
- 通过 CJ(P)指令对 OUT 指令进行跳转时扫描时间将变短。
- 通过 CJ(P)指令向后跳转时扫描时间将变长。
- 对于 CJ(P)指令，可跳转至比执行中的步号小的步之处。但是，为了避免看门狗定时器时限到，应考虑从这期间环路中跳出的方法。



（1）X3 为 ON 期间，执行环路。

（2）将 X7 置为 ON 时，从环路中跳出。

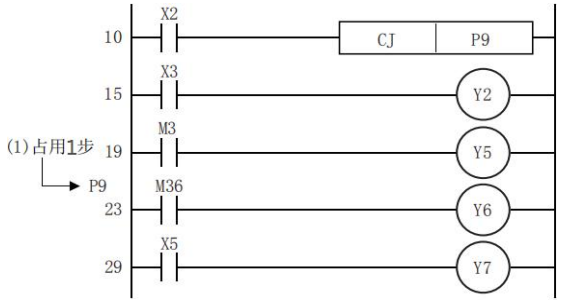
- 通过 CJ(P)指令跳过的软元件不变化。



X2 为 ON 时，跳转至 P19 的标签。

CJ 指令执行中即使 X2、X4 变为 ON/OFF，Y4，Y5 也不变化。

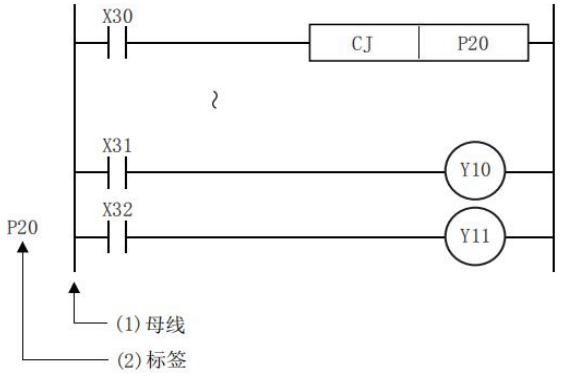
• 标签(P□)占用 1 步。



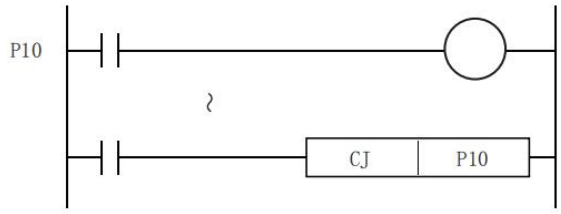
• 跳转指令只能指定同一程序文件内的指针编号。

• 跳转运行中跳转至跳转范围内的指针编号时，执行跳转目标指针编号以后的程序。

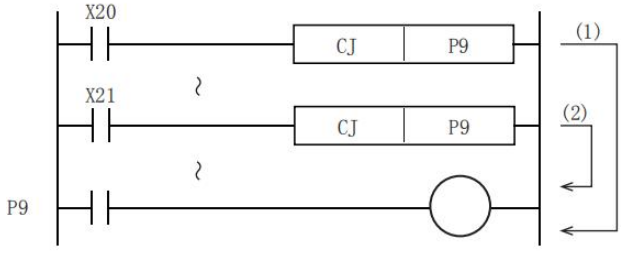
• 标签的程序如下所示。创建回路程序时，将光标移动到梯形图的母线左侧，在回路块起始位置输入标签(P)。



• 也可在步号小于 CJ 指令的位置进行标签编程，但如果扫描时间变为 200ms 以上(默认设置)，将发生看门狗定时器出错，需要注意。



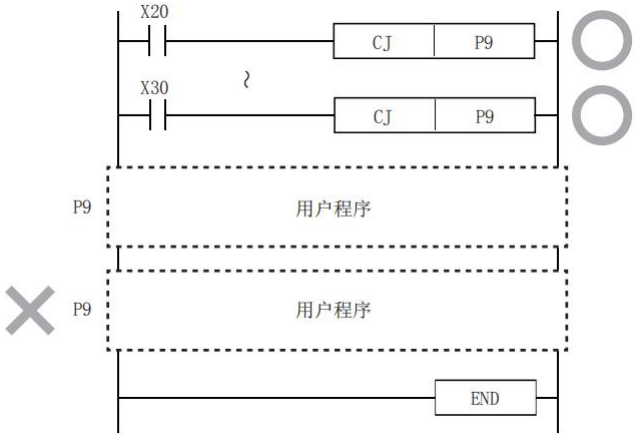
• 操作数中的指针编号为同一编号，且标签为 1 个的情况下，其动作如下所示。



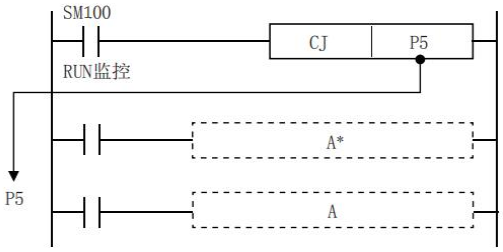
（1）X20 为 ON 时，从 X20 的 CJ 指令跳转至标签 P9。

（2）X20 为 OFF 且 X21 为 ON 时，从 X21 的 CJ 指令跳转至标签 P9。

• 如果重复使用标签编号，将变为出错状态。



• SM100 在 CPU 模块运行中始终为 ON，因此如下所示的使用方法将无条件跳转。



A*：用户程序（跳转而不运算）

A：用户程序

• LX3V 的指针编号 P63 代表跳转到 END 指令，LX6V 的 P63 指针不再提供该功能，若需使用该功能，请使用 GOEND 指令。

错误码

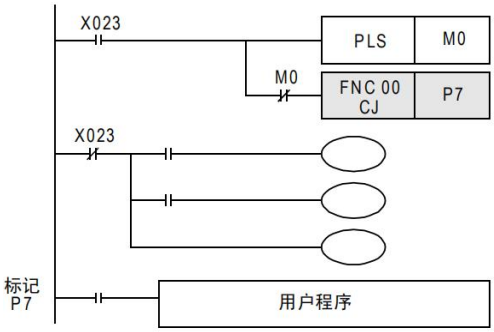
无错误信息。

示例

1. OFF 处理后要跳转的情况

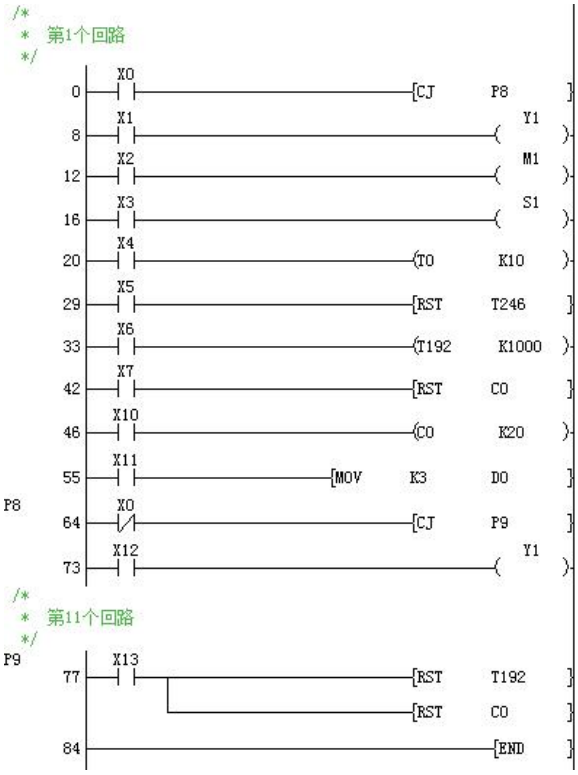
在 X023 从 OFF 变为 ON 的 1 个运算周期后，CJ P7 指令有效。

采用这个方法，可将 CJ P7 指令~标记 P7 之间的输出全部 OFF 后才进行跳转。



2、CJ 指令和触点线圈的动作

下面的程序举例中，当 X000 为 ON 时，从第 1 个回路的 CJ 指令向标记 P8 跳转。当 X000 为 OFF 时不进行跳转，但是从 1 步开始按顺序执行程序，在第 11 个回路的 CJ 指令向标记 P9 跳转。被跳转的指令不执行。



• Y001 输出的双线圈动作：
X000=OFF 时通过 X001 动作，
X000=ON 时通过 X012 动作，
即使是因条件跳转而区分开的程序，在跳转内或是跳转外对同一线圈(Y000)编程 2 次以上时，则视为一般的双重线圈处理。

• 子程序定时器(T192~T199)的动作 ：
线圈驱动后，即使跳转，动作也持续，输出触点也动作。

• 如果使用高速计数器(HSC0~HSC7)的动作
线圈驱动后，即使跳转，动作也持续，输出触点也动作。

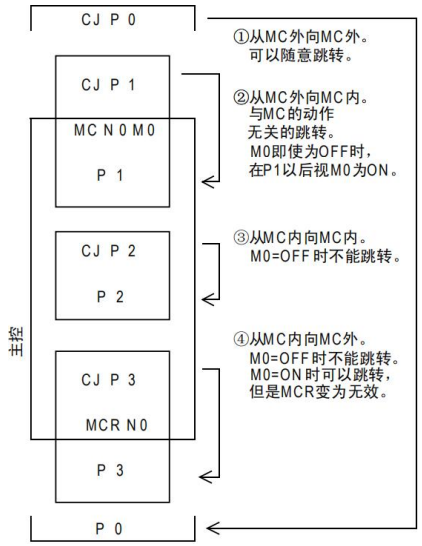
上述的程序中，如跳转中各个输入变化，则各线圈的动作如下表所示。

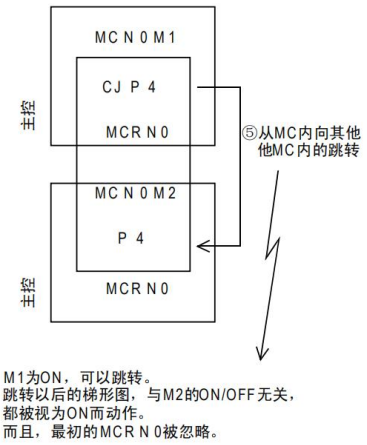
区分	跳转前的触点状态	跳转中的线圈动作
Y,M,S (Y1,M1,S1)	X1、X2、X3 OFF	Y1、M1、S1 OFF
	X1、X2、X3 ON	Y1、M1、S1 ON
1ms、10ms、100ms 定时器 （T0）	X4 OFF	定时器不工作
	X4 ON	计时中断（X0 OFF 后继续）
程序定时器 （T192）	X5 OFF、X6 OFF	定时器不工作，但是，X13 为 ON 时定时器被复位
	X5 OFF、X6 ON	计时继续（X0 OFF 后触点动作）
计数器 (C0)	X7 OFF、X10 OFF	计数中断，但是，X13 为 ON 时被复位
	X7 OFF、X10 ON	计数中断（X0 OFF 后继续）
应用指令 (MOV)	X11 OFF	跳转中不执行单周期应用指令 多周期应用指令部分可执行（如高速脉冲类指令）
	X11 ON	

3、CJ 指令与主控指令 MC~MCR 跳转的关系

主控指令和跳转指令的关系及动作内容如下所示。

但是，由于②、④、⑤的运行会变得复杂，所以请避免使用。





4.2 子程序跳转

CALL/子程序调用

跳转指令为 ON 时，执行同一程序文件内指定的指针编号的程序。

-[CALL(P)　(P)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(P)	子程序名	--	指针	POINTER	--

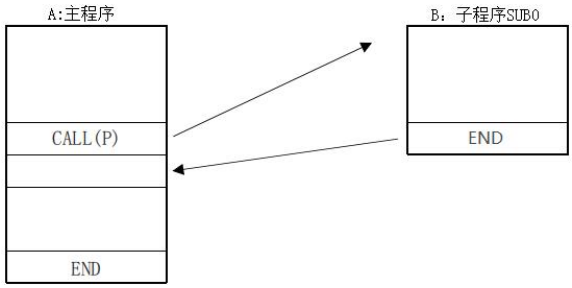
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展	其他
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP	子程序名
CALL	参数 1																									●	●

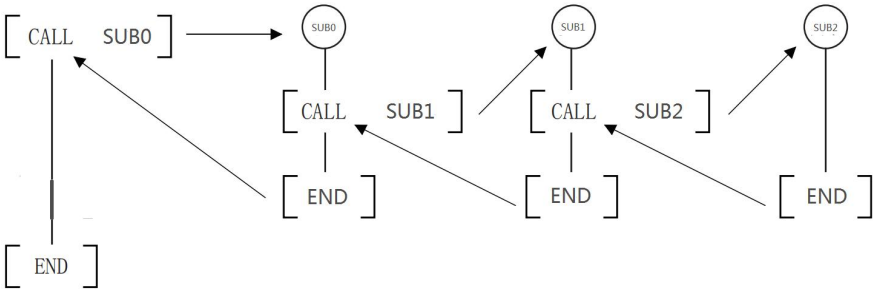
参数 1 只可使用子程序名。

功能

• 执行 CALL(P)指令时，将执行指针(P)的子程序。（P）中只能写入新建的子程序名，若程序名不存在，则梯形图编译不通过。



• CALL(P)指令的嵌套最多可达 32 重。



注意事项

• 可多个 CALL(P)指令调用同一个子程序，但不允许有程序名相同的子程序。

• 子程序内(中断程序内也同样)应使用程序用定时器。该定时器在执行线圈指令或 END 指令时计时。如果到达定时器设置值，在执行线圈指令或 END 指令时输出触点动作。一般的定时器仅在执行线圈指令时计时，因此如果在只有一定

条件下执行线圈指令的子程序内使用，将不计时。

• 如果在子程序内(中断程序内也同样)使用 1ms 累计定时器，当其到达设置值后，执行最先的线圈指令时(执行子程序时)输出触点动作，需要注意。

• 子程序内(中断程序内也同样)置为 ON 的软元件在程序结束后也将保持。因此，这些软元件应在程序结束后的主程序中进行复位。

错误码

错误码	内容
4102H	CALL(P)指令超出 32 层嵌套结构。

示例

1、子程序的新建

[工程管理]→[子程序]→[右键新建]

2、子程序的调用

扫描程序中，开启 M10 则调用子程序 SUB0，执行子程序 SUB0 中的梯形图，直到执行完子程序的 END 指令后，返回到扫描程序 MAIN 中执行 LD M11。

3、子程序嵌套

在上图中，扫描程序中调用子程序 SUB0，在 SUB0 中调用了子程序 SUB1，所以当扫描程序 M10 开启后，执行 CALL 指令后，会先执行子程序 SUB0，执行到 SUB0 的 CALL 指令后，会先执行 SUB1。执行 SUB1 的 END 指令后返回 SUB0 执行，执行完 SUB0 的 END 指令后，返回扫描程序 MAIN。该程序只有 2 层嵌套，注意嵌套层数不能大于 32。

4.3 中断禁止、中断允许

DI、EI/中断禁止、允许

CPU 模块通常为中断禁止状态。该指令可使 CPU 模块变为中断允许状态(EI 指令)，之后再次变为禁止(DI 指令)。

• DI：禁止中断程序的执行。

• EI：解除中断禁止状态。

-[DI (s)]

-[EI]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	禁止中断的优先度	[0,2]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DI	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	

功能

■DI

• 即使在程序中触发了执行中断的条件，在执行 EI 指令之前禁止中断程序的执行。

• PLC 上电时或 STOP 后将变为执行了 DI 指令后的状态，无法执行中断程序。

• DI 指令可选择是否使用参数，无参数时表示禁止所有中断程序。带参数时，根据参数 s1 中的数值，禁止该优先度及更小优先度的中断程序。

• 中断的优先度范围为[0,2]，数值越小中断响应优先级越高，即优先度为 0 的中断，最快被响应。

• 若在 DI 指令之前，没有任何 EI 指令，那该 DI 指令无效。

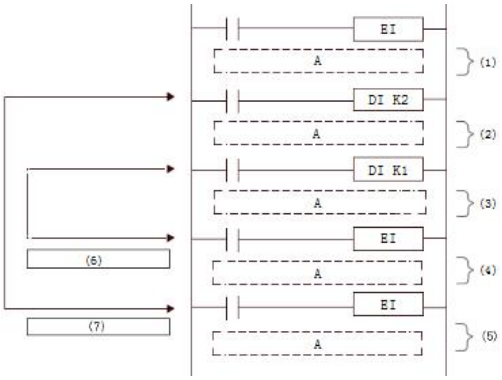
■EI

• 解除执行 DI 指令时的中断禁止状态，允许中断程序运行。

• EI、DI 指令在未使能时，都保持原来的允许或禁止的中断程序执行状态，当前禁止的中断优先度可查看 SD151。

SD151	当前禁止的中断优先级	根据中断禁止指令（DI 指令）、指定优先度以下的中断禁止指令（DI 指令）、中断允许指令（EI 指令），处于中断禁止的优先度将被存储。 0：所有优先度的中断禁止（默认）； 1：优先度 1、2 的中断禁止； 2：优先度 2 的中断禁止； 3：所有优先度的中断允许	R（只读）

■DI、EI 嵌套结构



A：顺控程序

（1）：所有优先度的中断允许区间；

（2）：优先度 2 以下的中断禁止区间（优先度 1 以上的中断允许区）；

（3）：优先度 1 以下的中断禁止区间（优先度 0 以上的中断允许区间）；

（4）：优先度 2 以下的中断禁止区间（优先度 1 以上的中断允许区）；

（5）：所有优先度的中断允许区间；

（6）：和[DI K1]配对的 EI；

（7）：和[DI K2]配对的 EI。

• DI 指令以后发生的中断(请求)在执行 EI 指令后进行处理。

• 多次执行 DI 指令时，自变量的优先度指定为高于当前禁止中的优先度的情况下，将自变量的优先度以下的中断置为禁止。

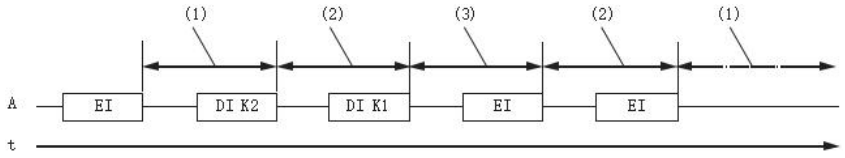
• 多次执行 DI 指令时，自变量的优先度指定为低于当前禁止中的优先度的情况下，中断禁止状态不会被更改。

• DI 指令的嵌套最多可达 16 重。

• 中断指针的中断优先度可通过中断程序的属性进行设置，详细参照中断程序说明。

• 执行 DI 指令及 EI 指令时的中断禁止区间如下所示。

1、多次执行 DI 指令的情况下（对优先度高于当前禁止的中断优先度的中断进行了禁止指定的情况下）



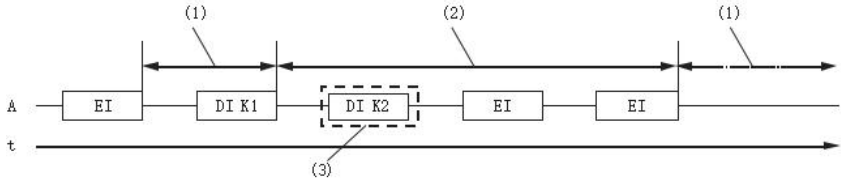
A：扫描执行类型程序

（1）：所有优先度的中断允许区间；

（2）：优先度 2 以下的中断禁止区间（优先度 1 以上的中断允许区间）；

（3）：优先度 1 以下的中断禁止区间（优先度 0 以上的中断允许区间）。

2、多次执行 DI 指令的情况下（对优先度低于当前禁止的中断优先度的中断进行了禁止指定的情况下）



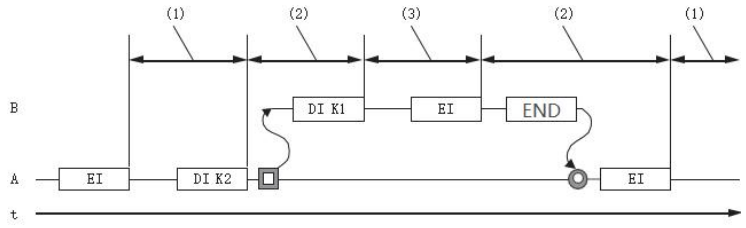
A：扫描执行类型程序

（1）：所有优先度的中断允许区间；

（2）：优先度 1 以下的中断禁止区间（优先度 0 以上的中断允许区间）；

（3）：优先度 1 以下的中断已处于被禁止状态，因此中断禁止优先度不被更改。

3、通过中断程序执行了 DI 指令的情况下



A：扫描执行类型程序

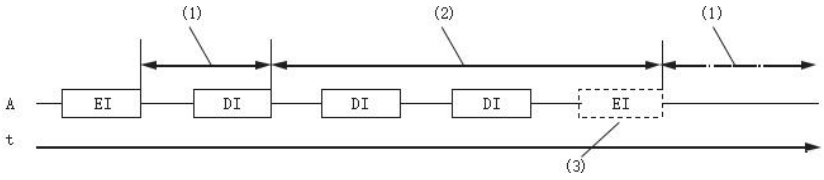
B：中断程序

（1）：所有优先度的中断允许区间；

（2）：优先度 3 以下的中断禁止区间（优先度 1 以上的中断允许区间）；

（3）：优先度 2 以下的中断禁止区间（优先度 0 以上的中断允许区间）。

4、仅执行了无自变量的 DI 指令的情况下



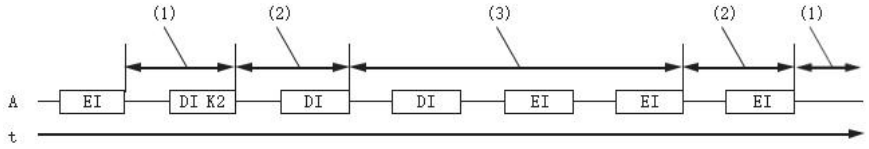
A：扫描执行类型程序

（1）：所有优先度的中断允许区间；

（2）：优先度 1 以下的中断禁止区间（所有的中断禁止区间）；

（3）：由于通过无自变量的 DI 指令置为中断禁止，因此通过执行 1 次 EI 指令，所有优先度的中断均被置为允许。

5、执行带自变量的 DI 指令和无自变量的 DI 指令的情况下（按照带自变量的 DI 指令→无自变量的 DI 指令的顺序执行的情况下）



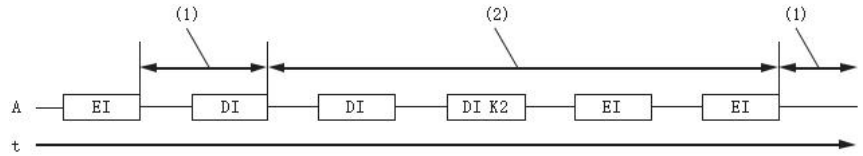
A：扫描执行类型程序

（1）：所有优先度的中断允许区间；

（2）：优先度 2 以下的中断禁止区间（优先度 1 以上的中断允许区间）；

（3）：优先度 1 以下的中断禁止区间（所有的中断禁止区间）。

6、执行带自变量的 DI 指令和无自变量的 DI 指令的情况下（按照带无自变量的 DI 指令→带自变量的 DI 指令的顺序执行的情况下）



A：扫描执行类型程序

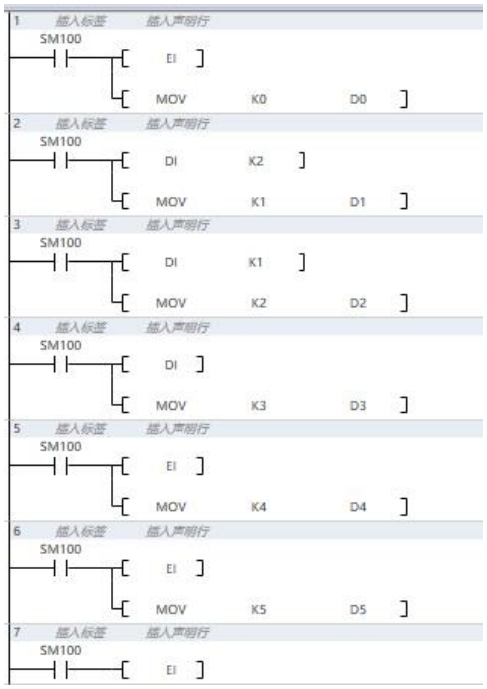
（1）：所有优先度的中断允许区间；

（2）：优先度 1 以下的中断禁止区间（所有的中断禁止区间）。

错误码

错误码	内容
4085H	（S）读取地址超出软元件范围。
4084H	（S）中设置的数据超过 0-2。
4185H	DI 指令的嵌套超过了 16 重的情况下。

示例



可触发所有中断程序

可触发优先度 0 和优先度 1 的中断程序

可触发优先度 0 的中断

无法触发任何中断

可触发优先度为 0 的中断程序

可触发优先度为 0 和优先度为 1 的中断程序

SIMASK/中断屏蔽

将(I)中指定的中断指针 No.按照(s)的值，置为执行允许状态/执行禁止状态。

-[SIMASK (I) (s)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(I)	中断程序名	--	程序名	POINTER	--
(s)	指定中断的允许/禁止	0：允许；1：禁止	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SIMASK	参数 1	只支持中断程序名																									
	参数 2											●	●	●	●	●	●	●	●	●			●	●	●	●	

功能

- 将(I)中指定的中断程序名的中断程序，按照(s)中指定的数据置为执行允许状态/执行禁止状态。
- (s)为 0 的情况下:中断程序的执行允许状态；(s)为 1 的情况下：中断程序的执行禁止状态。
- 关于电源投入时或 STOP→RUN 之后的中断程序，所有中断程序将变为执行状态。
- 设置中断禁止后，即使断开指令也会保存禁止状态，若要恢复需将(S)写 0，再次开启指令，或执行 STOP→RUN.
- 中断的执行允许状态/执行禁止状态将存储到 SM 或 SD 中。情况如下：

1、外部中断

寄存器	内容	寄存器	内容	寄存器	内容	寄存器	内容
SM352	X0 上升沿中断	SM356	X2 上升沿中断	SM360	X4 上升沿中断	SM364	X6 上升沿中断
SM353	X0 下降沿中断	SM357	X2 下降沿中断	SM361	X4 下降沿中断	SM365	X6 下降沿中断
SM354	X1 上升沿中断	SM358	X3 上升沿中断	SM362	X5 上升沿中断	SM366	X7 上升沿中断
SM355	X1 下降沿中断	SM359	X3 下降沿中断	SM363	X5 下降沿中断	SM367	X7 下降沿中断

2、定时器中断

寄存器	内容
SD350~SD356	定时器中断掩码，每个位代表一个中断，共 100 个。

3、高速计数中断

寄存器	内容
SD382~SD388	高速计数中断掩码，每个位代表一个中断，共 100 个。

错误码

错误码	内容
4084H	应用指令中(s)输入了超出非 0 和 1 的数据。
4085H	读应用指令中(s)超出了软元件范围。
4189H	SIMASK 指令指定了未设置的中断程序名。

示例

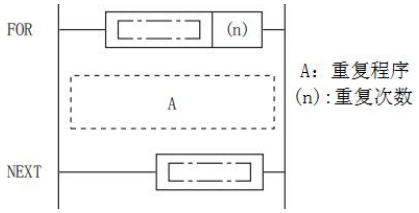
如图所示：

开启 M10 时，禁止 INT10、INT91、INT70 三个中断程序的运行。

4.4 循环指令

FOR~NEXT/循环

将 FOR~NEXT 指令之间的处理无条件执行(n)次时，将进行 NEXT 指令的下一步的处理。



内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(n)	FOR~NEXT 指令之间的重复次数	[1,32767]	有符号 BIN16 位	ANY16	INT

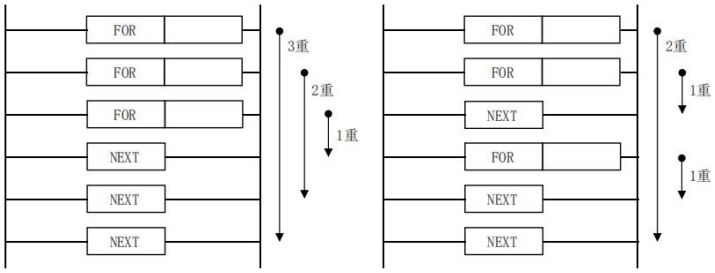
指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
FOR	参数 1											●	●	●	●	●	●	●	●	●			●	●	●	●	

功能

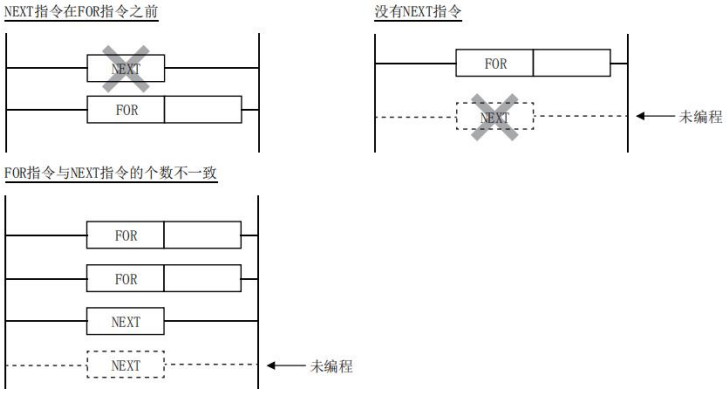
- 将 FOR~NEXT 指令之间的处理无条件执行(n)次时，将进行 NEXT 指令的下一步的处理。
- (n)可在 1~32767 的范围内指定。指定-32768~0 的情况下，将进行与(n)=1 相同的处理。
- 不希望执行 FOR~NEXT 指令之间的处理时，应通过 CJ 指令跳转。
- FOR 指令的嵌套最多可达 5 重。

注意事项

- 在 FOR~NEXT 指令之间以嵌套进行 FOR~NEXT 指令编程的情况下，最多可达 5 重。



- FOR~NEXT 指令之间不得用 IRET、SRET、RET、FEND、END 等指令阻断。
- 重复次数设置较多，使循环时间(运算周期)变长而造成看门狗定时器出错的情况下，需要更改看门狗定时器时间或进行看门狗定时器的复位。
- 下述程序将变为出错。

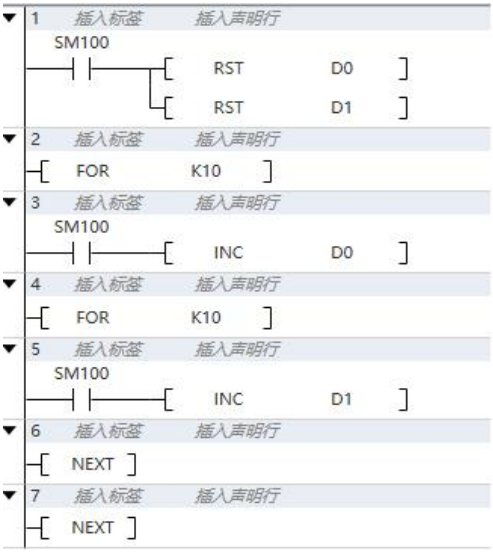


- FOR~NEXT 指令之间的重复执行中，中途结束的情况下，应使用 BREAK 指令。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4100H	FOR~NEXT 指令的嵌套超过了 5 重的情况下或FOR~NEXT 的个数不对应。

示例



程序 INC D0 将执行 10 次，INC D1 将执行 100 次。

执行结束后 D0 将等于 10，D1 等于 100。

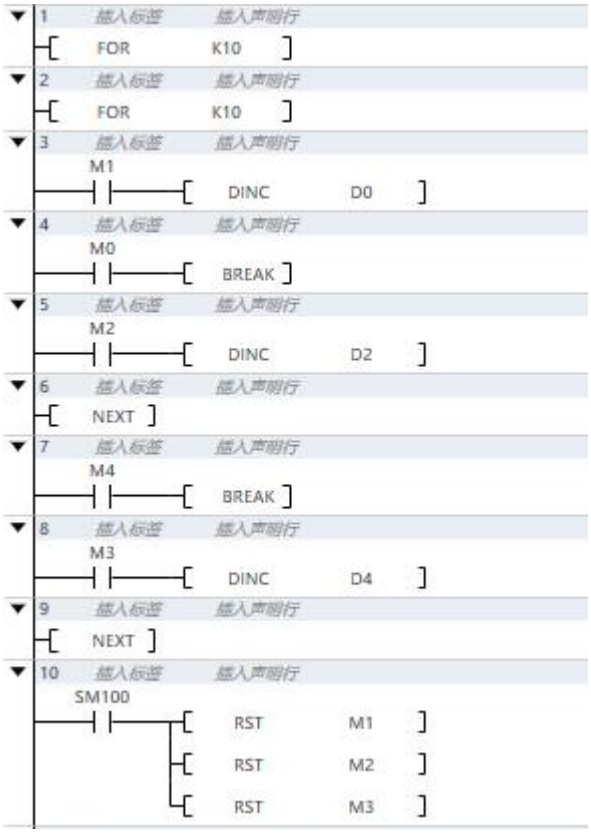
BREAK/跳出循环

将 FOR~NEXT 指令之间的处理无条件执行（n）次时，将进行 NEXT 指令的下一步处理。

-[BREAK]

功能

- 强制结束通过 FOR~NEXT 指令进行的重复处理；
- 该指令只能在 FOR~NEXT 之间，否则将报运算错误；
- BREAK 指令只能跳出指令自己所在的循环嵌套结构；
- 触点接合时，强制结束所在的 FOR~NEXT 指令进行的循环结构，如下图。



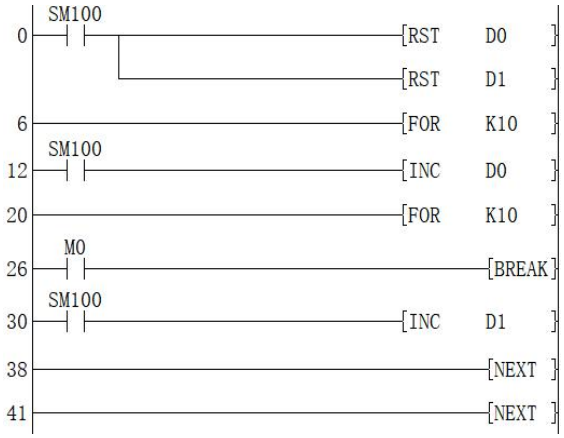
M0 置 ON，无论还剩多少次循环没执行完，直接跳转到第七个网格执行程序执行程序。

M4 置 ON，无论还剩多少次循环没执行完，直接跳转到第十个网格执行程序。

错误码

错误码	内容
4186H	BREAK 指令未在 FOR~NEXT 指令之间使用。

示例



程序 INC D0 将执行 10 次，INC D1 将执行 100 次。

M0 为 OFF 时，执行结束后 D0 将等于 10，D1 等于 100。

M0 为 ON 时，BREAK 指令执行，跳出当前循环，INC D1 指令将不执行，结果 D1=0。

4.5 看门狗复位

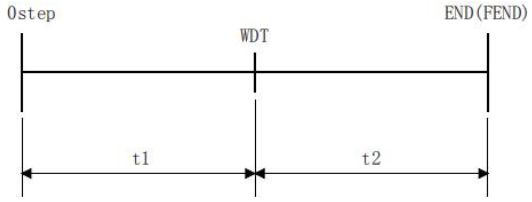
WDT/看门狗定时器

通过程序进行看门狗定时器的复位。

-[WDT]

功能

- 通过程序进行看门狗定时器的复位。
- 在扫描时间根据条件超过了看门狗定时器的设置值的情况下使用。
- 对于从 0 步开始至 WDT 指令为止的 t1，及从 WDT 指令开始至 END 指令为止的，请勿超过看门狗定时器的设置值。



- WDT 指令在 1 个扫描中可以使用 2 次以上。

注意事项

- 看门狗超时时间可在特殊寄存器 SD122 中设置。默认为 200ms。
- 使用特殊继电器 SM122 可控制是否开启看门狗定时器功能。关闭后 WDT 指令将无效。

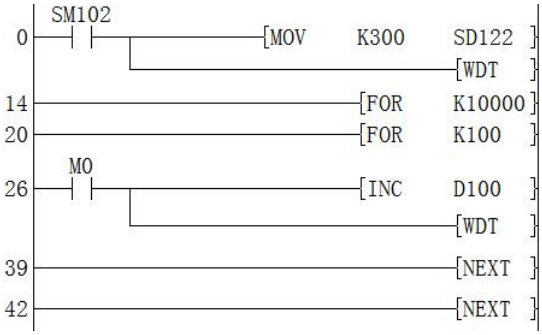
(1) 看门狗定时器时间设为 300ms；

(2) 刷新看门狗定时器。

错误码

没有运算出错。

示例



FOR~NXET 指令循环多次占用扫描周期长，可能会超过所设置的看门狗定时器 300ms，导致 PLC 报错无法继续运行，开启 M0 后，WDT 指令运行，每次循环都会更新看门狗定时器，使其不会报错正常执行程序。

5 定时器、计数器输出指令

5.1 定时器输出指令

OUT T/定时器输出

OUT 指令之前的运算结果为 ON 时，(d)中指定的定时器/累计定时器的线圈将 ON 并进行计测直至设置值为止。如果时限到，常开触点将导通，常闭触点将变为非导通。

-[OUT (d) (value)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	定时器软元件编号	--	计数器	ANY	--
(value)	定时器设置值	[0,32767]	无符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
OUT T	参数 1															●											
	参数 2											●	●	●	●		●	●	●	●			●	●			

功能

- OUT 指令之前的运算结果为 ON 时，(d)中指定的定时器的线圈将 ON 并进行计测直至设置值为止。如果计数到（当前值≥设置值），常开触点将导通，常闭触点将变为非导通。
- OUT 指令之前的运算结果由 ON→OFF 变化时其情况如下所示。

定时器类型	定时器线圈	定时器的当前值	时限到前		时限到后	
			常开触点	常闭触点	常开触点	常闭触点
定时器	OFF	0	非导通	导通	非导通	导通
累计定时器	OFF	保持当前值	非导通	导通	导通	非导通

- 时限到后，通过 RST 指令进行累计定时器的当前值的清除及触点的 OFF。
- 设置值为 0 时，执行 OUT 指令时将时限到。
- OUT T 指令为 ON 中，通过 CJ 指令等跳过了 OUT T 指令的情况下，将不进行当前值的更新及触点的 ON/OFF。
- 同一 OUT T 指令在同一扫描内执行了 2 次以上的情况下，将按照执行的次数进行当前值的更新。
- 各个定时器说明：

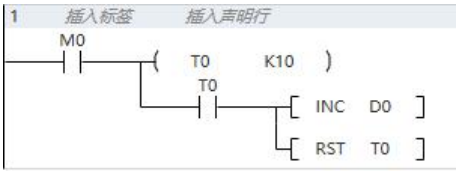
软元件编号	定时器规格	软元件编号	定时器规格
T0~T191	100ms 定时器	T246~T249	1ms 累计型定时器
T192~T199	100ms 子程序定时器（累计型，子程序中使用，即使子程序未被调用，依然会更新）	T250~T255	10ms 累计型定时器
		T256~T383	1ms 定时器
T200~T245	10ms 定时器	T384~T511	0.1ms 定时器

错误码

错误码	内容
4084H	(value)中参数设置超出范围。

示例

利用定时，每 1S 后 D0 加 1：



5.2 计数器输出指令

OUT C/计数器输出

16 位计数器指令：OUT 指令之前的运算结果由 OFF→ON 变化时，将(d)中指定的计数器的当前值+1，如果计数到，常开触点将导通，常闭触点变为非导通。

-[OUT (d) (value)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	计数器软元件编号	--	计数器	ANY	--
(value)	计数器设置值	[0,32767]	无符号 BIN16 位	ANY_INT	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
OUT C	参数 1																●										
	参数 2											●	●	●	●	●		●	●	●			●				

功能

• OUT 指令之前的运算结果由 OFF→ON 变化时，将(d)中指定的计数器的当前值(计数值)+1，如果计数到(当前值≥设置值)，常开触点将导通，常闭触点变为非导通。

• 运算结果为 ON 不变的状况下不进行计数。(计数输入无需脉冲化。)

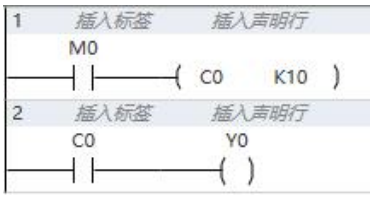
• 计数到后，在执行 RST 指令之前计数值及触点的状态不变化。

• 设置值为 0 的情况下，其处理与 1 时相同。

错误码

错误码	内容
4084H	(value)中参数设置超出范围。
4085H	(value)参数超出软元件范围。

示例



M0 每次从 OFF→ON 时，C0 都会加 1，当 C0 的值加到 K10 时，C0 常开触点闭合，Y0 输出。此时，M0 继续从 OFF→ON，C0 的值也不会再变化。

C0 的触点只可通过 RST/ZRST 指令和通讯能置 OFF。

OUT LC/长计数器输出

32 位计数器指令：OUT 指令之前的运算结果由 OFF→ON 变化时，将(d)中指定的长计数器的当前值+1，如果计数到，常开触点将导通，常闭触点变为非导通。

-[OUT (d) (value)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	长计数器软元件编号	--	计数器	ANY	--
(value)	长计数器设置值	[0,4294967295]	无符号 BIN32 位	ANY_INT	UDINT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
OUT LC	参数 1																				●						
	参数 2											●	●	●	●	●		●	●	●			●				

功能

• OUT 指令之前的运算结果由 OFF→ON 变化时，将(d)中指定的长计数器的当前值(计数值)+1，如果计数到(当前值≥设置值)，常开触点将导通，常闭触点变为非导通。

• 运算结果为 ON 不变的状况下不进行计数。(计数输入无需脉冲化。)

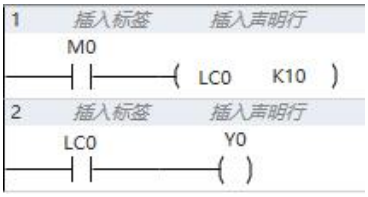
• 计数到后，在执行 RST 指令或 ZRST 指令之前计数值及触点的状态不变化。

• 设置值为 0 的情况下，其处理与 1 时相同。

错误码

错误码	内容
4085H	(value)参数超出软元件范围。

示例



M0 每次从 OFF→ON 时，LC0 都会加 1，当 LC0 的值加到 K10 时，LC0 常开触点闭合，Y0 输出。此时，M0 继续从 OFF→ON，LC0 的值也不会再变化。

LC0 的触点只可通过 RST/ZRST 指令和通讯能置 OFF。

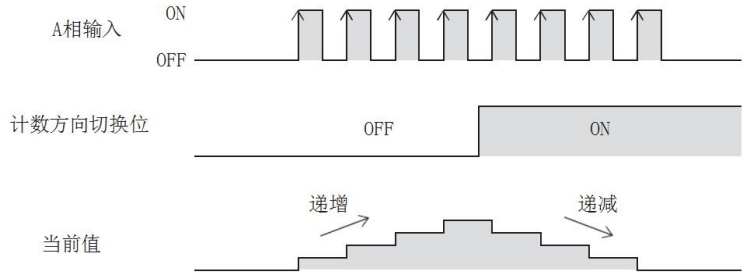
6 高速输入计数器

6.1 高速计数器的规格

高速计数器的类型

1、单相输入计数器（S/W）

单相输入计数器（S/W）的计数方法如下所示：

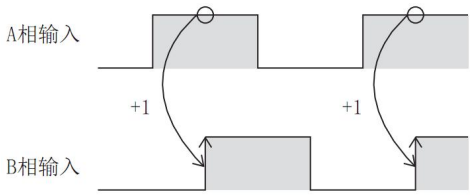


2、AB 相输入计数器[1 倍频]

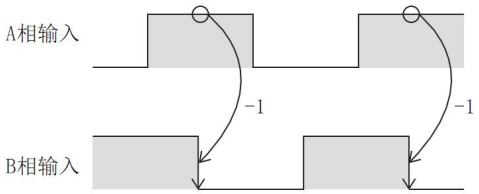
AB 相输入计数器[1 倍频]的计数方法如下所示：

递增/递减动作	计数时机
递增计数时	A 相输入 ON 而 B 相输入 OFF→ON 变化时计数递增 1。
递减计数时	A 相输入 ON 而 B 相输入 ON→OFF 变化时计数递减 1。

递增计数时



递减计数时

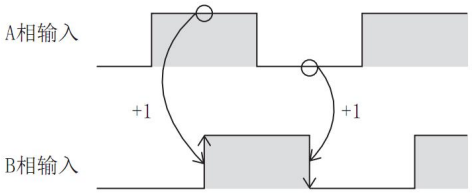


3、AB 相输入计数器[2 倍频]

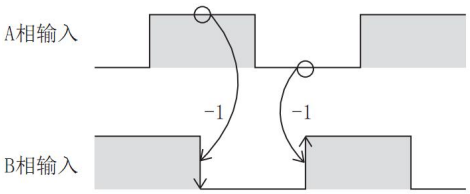
2 相 2 输入计数器[2 倍频]的计数方法如下所示：

递增/递减动作	计数时机
递增计数时	A 相输入 ON 而 B 相输入 OFF→ON 变化时计数递增 1； A 相输入 OFF 而 B 相输入 ON→OFF 变化时计数递增 1。
递减计数时	A 相输入 ON 而 B 相输入 ON→OFF 变化时计数递减 1； A 相输入 OFF 而 B 相输入 OFF→ON 变化时计数递减 1。

递增计数时



递减计数时

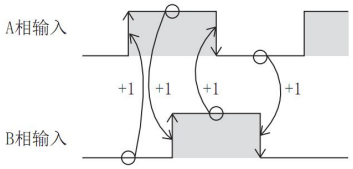


4、AB 相输入计数器[4 倍频]

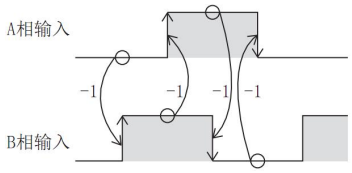
2 相 2 输入计数器[4 倍频]的计数方法如下所示：

递增/递减动作	计数时机
递增计数时	B 相输入 OFF 而 A 相输入 OFF→ON 变化时计数递增 1。 A 相输入 ON 而 B 相输入 OFF→ON 变化时计数递增 1。 B 相输入 ON 而 A 相输入 ON→OFF 变化时计数递增 1。 A 相输入 OFF 而 B 相输入 ON→OFF 变化时计数递增 1。
递减计数时	A 相输入 OFF 而 B 相输入 OFF→ON 变化时计数递减 1。 B 相输入 ON 而 A 相输入 OFF→ON 变化时计数递减 1。 A 相输入 ON 而 B 相输入 ON→OFF 变化时计数递减 1。 B 相输入 OFF 而 A 相输入 ON→OFF 变化时计数递减 1。

递增计数时



递减计数时



最高频率

各类高速计数器的可计数最高频率如下所示：

计数器类型	最高频率
单相输入计数器 （S/W）	150KHz
AB 相输入计数器[1 倍频]	100KHz
AB 相输入计数器[2 倍频]	100KHz
AB 相输入计数器[4 倍频]	100KHz

计数范围：[-2147483648,2147483647]，为带符号 32 位环形计数器。

高速计数器的分配

各类高速计数器的输入软元件为固定分配好，为 HSC0~HSC7 共 8 通道。

每个通道可根据在高速计数器配置中更改为单相输入或 AB 相输入，但需注意占用 X 点不能重复。

通道	高速计数器类型	X0	X1	X2	X3	X4	X5	X6	X7	X10	X11	X12	X13	X14	X15	X16	X17
HSC0	单相输入（S/W）	A															
	AB 相输入	A	B														
HSC1	单相输入（S/W）		A														
	AB 相输入			A	B												
HSC2	单相输入（S/W）			A													
	AB 相输入					A	B										
HSC3	单相输入（S/W）				A												
	AB 相输入							A	B								
HSC4	单相输入（S/W）					A											
	AB 相输入									A	B						
HSC5	单相输入（S/W）						A										
	AB 相输入											A	B				
HSC6	单相输入（S/W）							A									
	AB 相输入													A	B		
HSC7	单相输入（S/W）								A								
	AB 相输入															A	B

A:A 相输入　　B:B 相输入

注意事项：在 HSC0 使用了 AB 相输入后，HSC1 将不能再使用单相输入，因为 HSC0 占用了 X0 和 X1 两点，而 HSC1 若要使用单相输入，需要占用 X1，产生冲突。其他通道也同理。

高速计数器使用步骤

以下对高速计数器的使用步骤进行说明。

［工程管理］→［参数］→［高速计数配置］

1、画面显示

2、显示内容

参数	范围	说明	默认值
是否使用	使用/不使用	设置是否使用计数器。	不使用
脉冲输入模式	单相输入/AB 相输入	选择使用单相输入还是 AB 相输入。	单相输入
计数方向	增计数模式/减计数模式	选择增减计数模式，只有单相输入时有效。	增计数模式
倍频	1 倍频/2 倍频/4 倍频	选择输入计数倍频，只有 AB 相输入时有效。	1 倍频
输入频率测试时间(ms)	[1,32767](ms)	设置间隔多久进行一次输入频率测定，设置时间越短，频率越不准确。该频率测定结果输出在特殊寄存器 SD 中，具体查看特殊寄存器 SD 高速计数器部分说明。	1000ms
滤波时间	[0,1700](0.01us)	设置该通道的 X 点作为高速输入时的滤波时间，滤波设置越小，理论计数越准确，但抗干扰能力降低（该滤波时间只有单向输入时才有效）。 输入为 0 时，为系统所支持的最低滤波时间。	1
最高频率	单相输入：150K AB 相输入：100K	显示各个通道能达到的最高输入频率，只读。	
占用 X 点	--	显示使用该通道后，占用了哪些 X 点，只读。	
检查按钮		检查配置的 X 输入点是否复用，建议设置完成时先点击检查，后再确认输入。	
恢复为默认		恢复为上图相同的默认设置。	
占用输入(X)说明		弹出各个通道的所有模式占用 X 的情况说明表。	
确认输入		配置完成点击确认输入才能够保存配置并且生效。	

3、配置实例

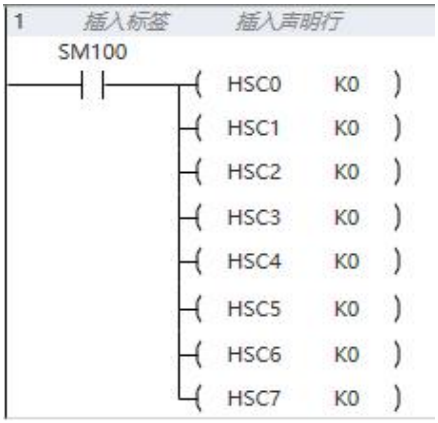
HSC0~HSC3 配置为 4 路单相输入，HSC4~HSC7 配置为 4 路 AB 相输入。

高速计数配置

	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	HSC6	HSC7
是否使用	使用	使用	使用	使用	使用	使用	使用	使用
脉冲输入模式	单相输入	单相输入	单相输入	单相输入	AB相输入	AB相输入	AB相输入	AB相输入
计数方向	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式
倍频	1倍频	1倍频	1倍频	1倍频	4倍频	4倍频	4倍频	4倍频
输入频率测定时间（ms）	1000	1000	1000	1000	1000	1000	1000	1000
滤波时间	1	1	1	1	1	1	1	1
最高频率	150K	150K	150K	150K	100K	100K	100K	100K
占用X点	单向：X0 AB相：X0、X1	单向：X1 AB相：X2、X3	单向：X2 AB相：X4、X5	单向：X3 AB相：X6、X7	单向：X4 AB相：X10、X11	单向：X5 AB相：X12、X13	单向：X6 AB相：X14、X15	单向：X7 AB相：X16、X17

检查　恢复为默认　确认输入　占用输入(X)说明

主程序中使用 OUT HSC 指令开启高速计数。此时，只要外部有脉冲输入即可在 HSC0~HSC7 中观测到脉冲值。



在特殊软元件 SD403，SD402 组成的双字中，可监测当前 HSC0 的输入脉冲频率。其他通道也有相应的寄存器，具体可查看特殊寄存器说明。

若需要停止计数，只需关闭 OUT HSC 指令即可。

6.2 高速计数器的指令

OUT HSC/高速计数器开关

OUT HSC 指令之前的运算结果为 ON 时，开启高速计数器，此时 HSC 寄存器的值记录当前接受到的高速脉冲个数，若计数值达到，则相应的 HSC 的位寄存器变为导通。

-[OUT (d) (value)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	高速计数器的通道	HSC0~HSC7	有符号 BIN32 位	ANY32	--
(value)	高速计数器设置值	[-2147483648,2147483647]	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
OUT HSC	参数 1																					●					
	参数 2											●	●	●	●			●	●	●			●				

功能

开启或关闭高速计数器计数，要使用高速计数器，必须先对高速输入通道进行配置，详细参照高速计数器说明。

指令前运算结果	动作	HSC 数据寄存器状态	HSC 位寄存器状态
ON	开启高速计数	数值根据输入的脉冲进行累加	数值达到设定值置 ON，否则 OFF
OFF	停止高速计数	数值保持不变	状态保持不变

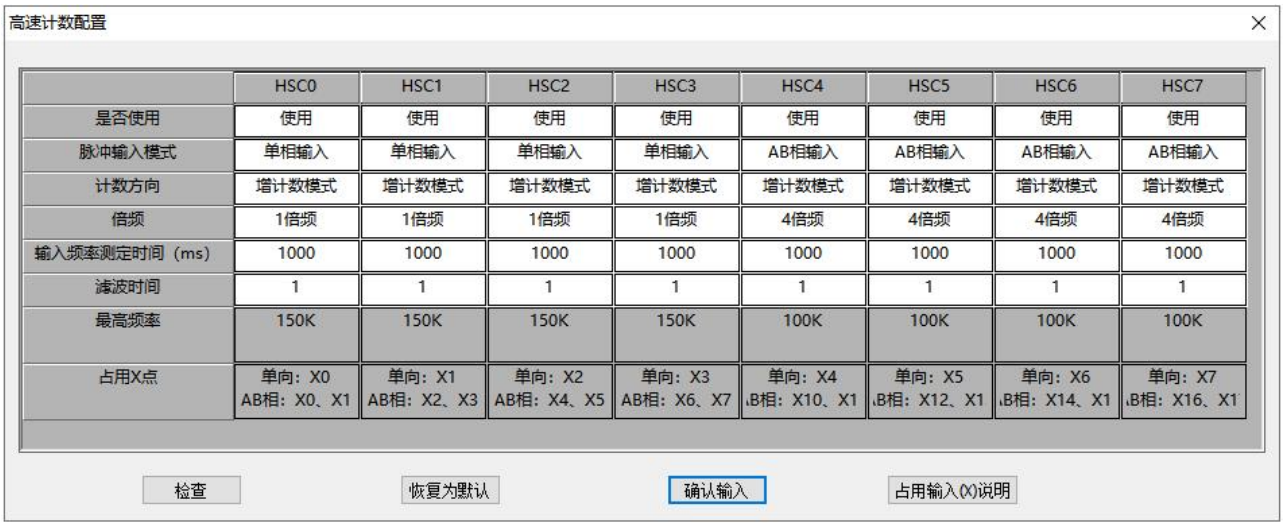
开启高速计数后，软元件 HSC 的值是每 100us 才会更新一次，所以会有最大 100us 的滞后性。若需要获取当前最新的计数值，可以使用 REF 指令对高速计数器进行刷新，具体用法查看 REF 指令。

错误码

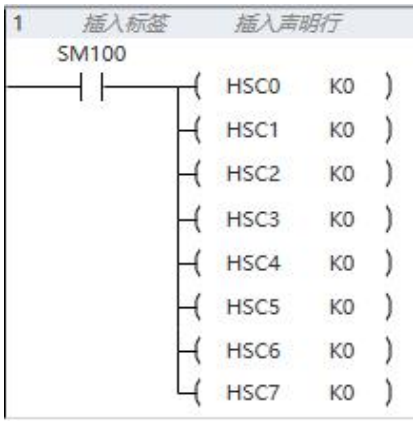
错误码	内容
4085H	(value) 读取地址超出软元件范围。
2580H	高速计数器开启后，但未配置该轴高速计数器使能。

示例

HSC0~HSC3 配置为 4 路单相输入，HSC4~HSC7 配置为 4 路 AB 相输入。



主程序中使用 OUT HSC 指令开启高速计数。此时，只要外部有脉冲输入即可在 HSC0~HSC7 中观测到脉冲值。



在特殊软元件 SD403，SD402 组成的双字中，可监测当前 HSC0 的输入脉冲频率。其他通道也有相应的寄存器，具体可查看特殊寄存器说明。

当 HSC0 的值大于 0 时，HSC0 的触点会置位，其他通道同理。如下梯形图，Y0 将导通。

DHSCS/高速比较置位

每次计数时比较高速计数器中计数的值与指定值，然后立即设置位软元件的指令。

-[DHSCS (s1) (s2) (d)]

内容、范围、数据类型

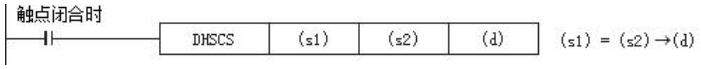
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	与高速计数器的当前值比较的数据,或存储了要比较数据的字软元件编号	[-2147483648, 2147483647]	有符号 BIN32 位	ANY32	DINT
(s2)	高速计数器软元件	HSC0~HSC7	有符号 BIN32 位	ANY32	--
(d)	一致时设置(ON)的位软元件编号		位	ANY_BOOL	--

使用的软元件

指令	参数	软元件																										偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DHSCS	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
	参数 2																					●					
	参数 3		●	●	●	●					●																

功能

• (s2)中指定的通道的高速计数器当前值变为比较值(s1)时(在比较值 K200 的情况下，199→200 及 201→200)，无论扫描时间如何，位软元件(d)都将被设置(ON)。该指令在高速计数器的计数处理后进行比较处理。



• 若(d)中指定的软元件为 Y0~Y20，在(d)被设置时，会直接将 Y 映射到实际硬件输出，不受扫描周期影响。

• DHSCS 参数 3 还可以使用中断函数名作为参数，如下图，在 HSC0 从(19999→20000)或(20001→20000)时都会执行中断程序 INT0。

注意事项

高速计数器中断只支持共 100 个程序，每条 DHSCS 也算在这 100 条内，超过则会报运算错误。

使用 DHSCS 指令必须使用 OUT HSC 指令开启高速计数器。

错误码

错误码	内容
4084H	(S2)中输入软元件超出了 HSC0~HSC7 的范围。
4085H	(s1)(s2) 读取地址超出软元件范围。
4086H	(d) 写入地址超出软元件范围。
2406H	高速计数中断个数超过了 100 条。
4F81H	DHSCS、SHSCR、DHSZ 指令开启但没有使用 OUT HSC 指令开启高速计数器计数。

示例

配置高速计数器，使用 HSC0 为例。

	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	HSC6	HSC7
是否使用	使用	不使用	不使用	不使用	不使用	不使用	不使用	不使用
脉冲输入模式	AB相输入	单相输入	单相输入	单相输入	单相输入	单相输入	单相输入	单相输入
计数方向	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式
倍频	4倍频	1倍频	1倍频	1倍频	1倍频	1倍频	1倍频	1倍频
输入频率测定时间（ms）	1000	1000	1000	1000	1000	1000	1000	1000
滤波时间	0	0	0	0	0	0	0	0
最高频率	100K	150K	150K	150K	150K	150K	150K	150K
占用X点	单向：X0 3相输入：X0、）	单向：X1 3相输入：X2、）	单向：X2 3相输入：X4、）	单向：X3 3相输入：X6、）	单向：X4 相输入：X10、）	单向：X5 相输入：X12、）	单向：X6 相输入：X14、）	单向：X7 相输入：X16、）

扫描 MAIN 中先使用 EI 指令使能中断，再使用 OUT HSC 指令开启高速计数器。

开启 M0 后，HSC0 的值从 19999→20000 时，执行一次 INT0 程序，即 D0 加 1。

HSC0 的值从 20000→20001 时，INT0 程序不执行，即 D0 保持为 1。

HSC0 的值从 20001→20000 时，执行一次 INT0 程序，即 D0 加 1，此时 D0 为 2。

DHSCR/高速比较复位

每次计数时比较高速计数器中计数的值与指定值，然后立即复位位软元件的指令。

-[DHSCR (s1) (s2) (d)]

内容、范围、数据类型

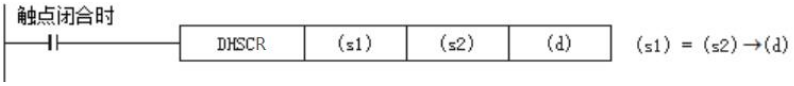
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	与高速计数器的当前值比较的数据,或存储了要比较数据的字软元件编号	[-2147483648, 2147483647]	有符号 BIN32 位	ANY32	DINT
(s2)	高速计数器软元件	HSC0~HSC7	有符号 BIN32 位	ANY32	--
(d)	一致时复位(OFF)的位软元件编号		位	ANY_BOOL	--

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DHSCR	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
	参数 2																					●					
	参数 3		●	●	●	●					●																

功能

• (s2)中指定的通道的高速计数器当前值变为比较值(s1)时(在比较值 K200 的情况下，199→200 及 201→200)，无论扫描时间如何，位软元件(d)都将被复位(OFF)。该指令在高速计数器的计数处理后进行比较处理。



• 若(d)中指定的软元件为 Y0~Y20，在(d)被设置时，会直接将 Y 映射到实际硬件输出，不受扫描周期影响。

注意事项

高速计数器中断只支持共 100 个程序，每条 DHSCR 也算在这 100 条内，超过则会报运算错误。

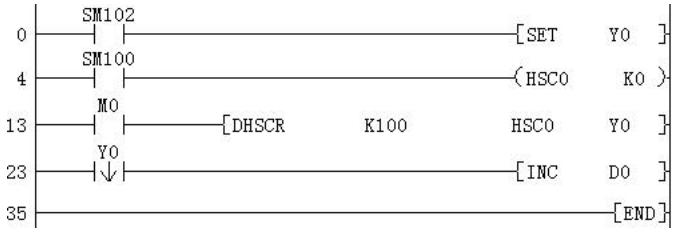
使用 DHSCR 指令必须使用 OUT HSC 指令开启高速计数器。

错误码

错误码	内容
4084H	(S2)中输入软元件超出了 HSC0~HSC7 的范围。
4085H	(s1)(s2) 读取地址超出软元件范围。
4086H	(d) 写入地址超出软元件范围。
2406H	高速计数中断个数超过了 100 条。
4F81H	DHSCS、SHSCR、DHSZ 指令开启但没有使用 OUT HSC 指令开启高速计数器计数。

示例

配置高速计数器，使用 HSC0 为例。



• 扫描 MAIN 中使用 OUT HSC 指令开启高速计数器。

• 开启 M0 后，HSC0 的值从 99→100 时，执复位 Y0，D0 会加 1。

DHSZ/高速区间比较

高速计数的当前值与 2 个值（带宽）进行比较，并输出比较结果。

-[DHSZ (s1) (s2) (s3) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	与高速计数器的当前值比较的数据,或存储了要比较数据的字软元件编号（比较值 1）	[-2147483648, 2147483647]	有符号 BIN32 位	ANY32	DINT
(s2)	与高速计数器的当前值比较的数据,或存储了要比较数据的字软元件编号（比较值 2）	[-2147483648, 2147483647]	有符号 BIN32 位	ANY32	DINT
(s3)	高速计数器软元件	HSC0~HSC7	有符号 BIN32 位	ANY32	--
(d)	在比较值 1 与比较值 2 中输出比较结果的起始位软元件编号		位	ANYBIT_ARRAY（要素数：3）	--

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DHSZ	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
	参数 3																					●					
	参数 4		●	●	●	●					●																

参数 3 还可支持中断程序名。

功能

• 将(s3)中指定的高速计数器当前值与 2 个比较值(比较值 1、比较值 2)进行带宽比较，无论扫描时间如何，(d)、(d)+1、(d)+2 中的一项都将根据比较结果(下、区域内、上)变为 ON。

• 若(d)中指定的软元件为 Y0~Y15，在(d),(d+1),(d+2)被设置时，会直接将 Y 映射到实际硬件输出，不受扫描周期影响。

• 设置[比较值 1]和[比较值 2]时，请保证[比较值 1]<[比较值 2]。如果设置不同，将发生运算错误，DHSZ 指令不执行动作。

注意事项

高速计数器中断只支持共 100 个程序，每条 DHSZ 也算在这 100 条内，且 DHSZ 指令会占用 2 条中断程序空间，超过则会报运算错误。

比较结果占用以（d）起始的 3 个连续地址的单元。请注意不要与其他控制的软元件重复。此外，指定 Y 软元件的情况下，设置请不要超出实际 Y 点输出数。使用 DHSZ 指令必须先使用 OUT HSC 指令开启高速计数器。

错误码

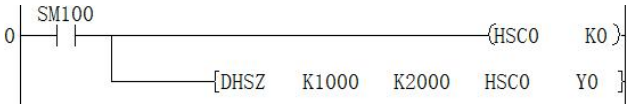
错误码	内容
4084H	(s2)中输入软元件超出了 HSC0~HSC7 的范围。
4085H	(s1)(s2) 读取地址超出软元件范围。
4086H	(d) 写入地址超出软元件范围。
2406H	高速计数中断个数超过了 100 条。
4F81H	DHSCS、SHSCR、DHSZ 指令开启但没有使用 OUT HSC 指令开启高速计数器计数。

示例

配置高速计数器，使用 HSC0 为例。

	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	HSC6	HSC7
是否使用	使用	不使用	不使用	不使用	不使用	不使用	不使用	不使用
脉冲输入模式	AB相输入	单相输入	单相输入	单相输入	单相输入	单相输入	单相输入	单相输入
计数方向	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式	增计数模式
倍频	4倍频	1倍频	1倍频	1倍频	1倍频	1倍频	1倍频	1倍频
输入频率测定时间（ms）	1000	1000	1000	1000	1000	1000	1000	1000
滤波时间	0	0	0	0	0	0	0	0
最高频率	100K	150K	150K	150K	150K	150K	150K	150K
占用X点	单向：X0 3相输入：X0、）	单向：X1 3相输入：X2、）	单向：X2 3相输入：X4、）	单向：X3 3相输入：X6、）	单向：X4 相输入：X10、）	单向：X5 相输入：X12、）	单向：X6 相输入：X14、）	单向：X7 相输入：X16、）

扫描程序：



执行结果：

比较模式	通道 1 的当前值（s3）	输出触点（Y）的变化		
		Y0	Y1	Y3
（s1）＞（s3）	1000>(s3)	ON	OFF	OFF
	999→1000	ON→OFF	OFF→ON	OFF
	1000→999	OFF→ON	ON→OFF	OFF
（s1）≤（s3）≤（s2）	999→1000	ON→OFF	OFF→ON	OFF
	1000→999	OFF→ON	ON→OFF	OFF
	1000≤（s3）≤2000	OFF	ON	OFF
	2000→2001	OFF	ON→OFF	OFF→ON
	2001→2000	OFF	OFF→ON	ON→OFF
（s3）＞（s2）	2000→2001	OFF	ON→OFF	OFF→ON
	2001→2000	OFF	OFF→ON	ON→OFF
	（s3）＞2000	OFF	OFF	ON

7 应用指令

7.1 传送比较指令

MOV/16 位传送

MOV(P)

将(s)中指定的软元件的 BIN16 位数据传送到(d)中指定的软元件。

-[MOV (s) (d)]

内容、范围、数据类型

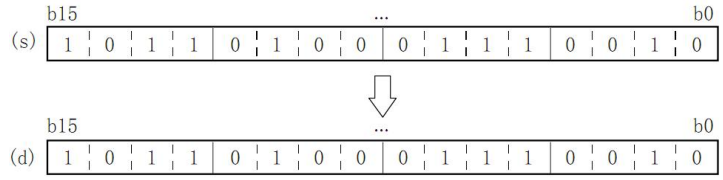
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	传送源数据或存储了数据的软元件编号	[-32768,32767]	有符号 BIN16	ANY16_S	INT
(d)	传送目标软元件编号	--	有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MOV	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

• 将（s）中指定的 BIN16 位数据传送到（d）中指定的软元件。



错误码

错误码	内容
4085H	读应用指令中(s)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将 D0 的值传送到 D2 的值：（D0）→（D2）。

DMOV/32 位传送

DMOV(P)

将(s)中指定的软元件的 BIN32 位数据传送到(d)中指定的软元件。

-[DMOV (s) (d)]

内容、范围、数据类型

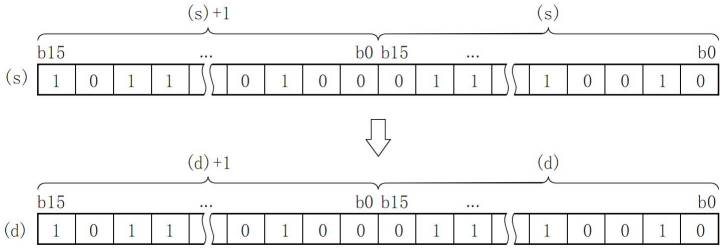
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	传送源数据或存储了数据的软元件编号	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	传送目标软元件编号	--	有符号 BIN32	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DMOV	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●

功能

- 将（s）中指定的 BIN16 位数据传送到（d）中指定的软元件。



错误码

错误码	内容
4085H	读应用指令中(s)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将（D1，D0）的值传送到（D3，D2）的值：（D1，D0）→（D3，D2）。

BMOV/成批传送

BMOV(P)

将(s)中指定的软元件开始的（n）点 BIN16 位数据依次传送到(d)中指定的软元件。

-[BMOV (s) (d) (n)]

内容、范围、数据类型

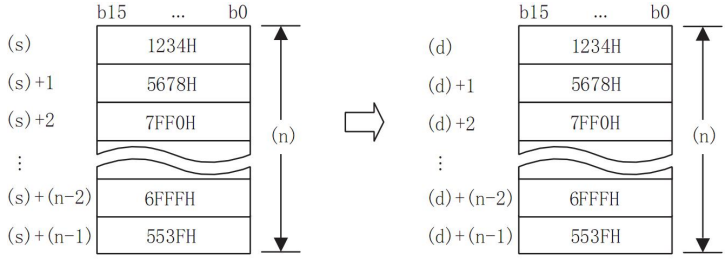
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了传送数据的起始软元件	--	有符号 BIN16	ANY16_S	--
(d)	传送目标的起始软元件	--	有符号 BIN16	ANY16_S	--
(n)	传送数	[1≤n≤512]	有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BMOV	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

- 将(s)中指定的软元件开始的(n)点的 BIN16 位数据批量传送到(d)中指定的软元件。

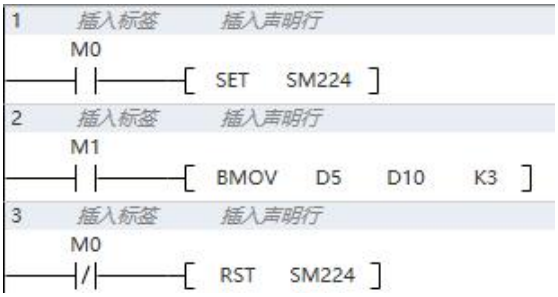


- 软元件编号超出范围时，在允许的范围内进行传送。
- 通过控制 BMOV 指令的方向反转标志位（SM224），即可实现将(d)中指定的软元件开始的(n)点的 BIN16 位数据批量传送到(s)中指定的软元件。

错误码

错误码	内容
4084H	应用指令中(n)输入了超出可指定范围的数据。
4085H	读应用指令中(s) (n)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例



M0 置位时，置位 M1，那么(D5)→(D10)；(D6)→(D11)；(D7)→(D12)；

M0 复位时，置位 M1，那么(D10)→(D5)；(D11)→(D6)；(D12)→(D7)。

FMOV/16 位多点传送

FMOV(P)

将(s1)中指定的软元件的 BIN16 位数据，以（n）点传送到(d)中指定的软元件（即将同个数据传送给多个地址）。

-[FMOV (s) (d) (n)]

内容、范围、数据类型

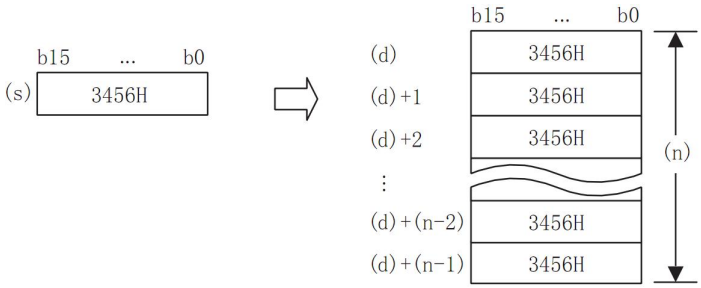
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	传送数据或存储了传送数据的起始软元件	[-32768,32767]	有符号 BIN16	ANY16_S	INT
(d)	传送目标的起始软元件	--	有符号 BIN16	ANY16_S	--
(n)	传送数	[1≤n≤512]	有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
FMOV	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

•将与(s)中指定的软元件的 BIN16 位数据相同的数据，以(n)点传送到(d)中指定的软元件中。



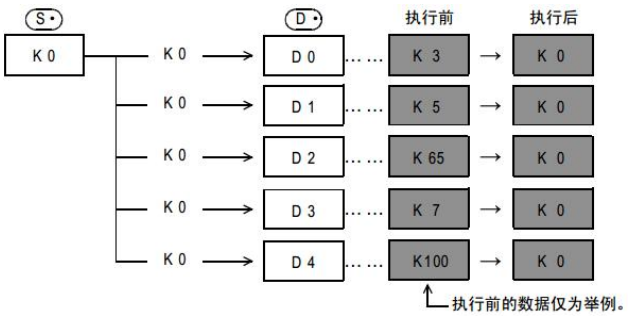
- (n)中指定的个数超出软元件编号范围时，在允许的范围内进行传送。
- 为传送源(s)指定常数(K)时，将自动转换为 BIN。

错误码

错误码	内容
4084H	应用指令中(s)(n)输入了超出可指定范围的数据。
4085H	读应用指令中(s)(n)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将 D0-D4 的值设为 0。



DFMOV/32 位多点传送

DFMOV(P)

将(s1)中指定的软元件的 BIN32 位数据，以（n）点传送到(d)中指定的软元件（即将同个数据传送给多个地址）。

-[DFMOV (s) (d) (n)]

内容、范围、数据类型

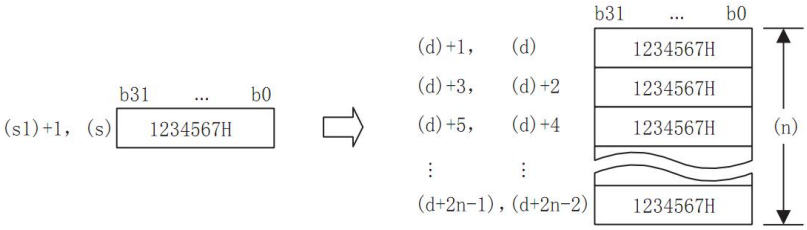
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	传送数据或存储了传送数据的起始软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	传送目标的起始软元件	--	有符号 BIN32	ANY32_S	--
(n)	传送数	[1≤n≤512]	有符号 BIN32	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DFMOV	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

- 将与(s)中指定的软元件的 BIN32 位数据相同的数据，以(n)点传送到(d)中指定的软元件中。



- (n)中指定的个数超出软元件编号范围时，在允许的范围内进行传送。
- 为传送源(s)指定常数(K)时，将自动转换为 BIN。

错误码

错误码	内容
4084H	应用指令中(s)(n)输入了超出可指定范围的数据。
4085H	读应用指令中(s)(n)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将(D1，D0)，(D3，D2)，(D5,D4)，(D7,D6)，(D9,D8)的值设为 0。

SMOV/位移动

SMOV(P)

以位数为单位（4 位）将数据进行分配合成的指令。

-[SMOV (s) (n1) (n2) (d) (n3)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了要移动位数的数据的字软元件编号		有符号 BIN16	ANY16_S	INT
(n1)	传送目标软元件编号	[1,4]	有符号 BIN16	ANY16_S	INT
(n2)	要移动的位数	[1,4]	有符号 BIN16	ANY16_S	INT
(d)	存储进行位移动的数据的字软元件编号		有符号 BIN16	ANY16_S	INT
(n3)	移动目标的起始位置	[1,4]	有符号 BIN16	ANY16_S	INT

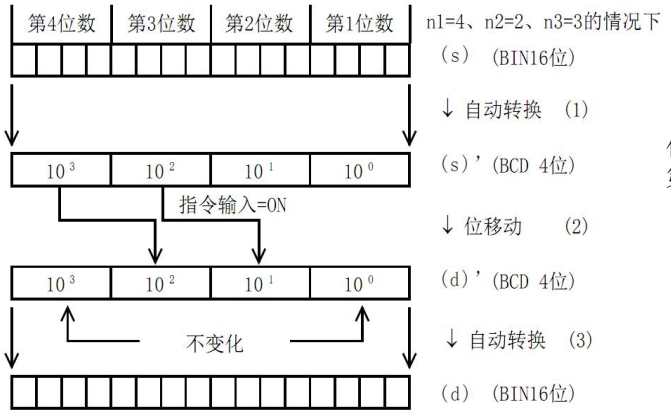
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SMOV	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 4												●	●	●	●	●	●	●	●						●	●
	参数 5											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

将数据以位数单位(4 位)进行分配/合成。传送源(s)与传送目标(d)的内容转换为 4 位数 BCD(0000~9999)，将(n1)位到低位(n2)位、传送目标(d)的(n3)位传送(合成)至起始位置后，转换为 BIN，存储到传送目标(d)中。

- 指令输入 OFF 时，传送目标(d)不发生变化。
- 指令输入 ON 时，传送源(s)的数据及传送目标(d)的传送指定以外的位数不发生变化。



（1）：对（s）执行 BIN→BCD 转换。
（2）：将第（n1）位到低位（n2）、（d）的第（n3）位传送（合成）至从前数起的（n2）位。（d）的第 1 位、第 4 位从（s）开始传送时不受影响。
（3）：将合成的数据（BCD）转换成 BIN，存储到（d）中。

扩展功能

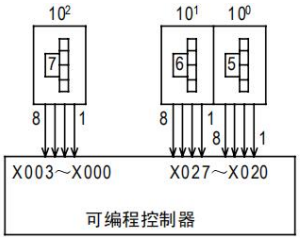
如果将 SM168 置为 ON 后执行 SMOV 指令，则不进行 BIN→BCD 转换。位移动以 4 位单位进行。

错误码

错误码	内容
4084H	应用指令中(n1)(n2)(n3)输入了超出可指定范围的数据或不满足 n2≤n1、n2≤n3 的关系。
4085H	读应用指令中(s)(n1)(n2)(d)(n3)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

合成 3 位数的数字式开关的数据后，以二进制保存到 D2 中。



将非连续的输入端子中连接的 3 个数字式开关额数据进行合成。

M0 置位时，将

(X020~X027)BCD 2 位数→D 2(二进制)；

(X000~X003)BCD 1 位数→D 1(二进制)；

将 D1 的 1 位数存入 D2 的第 3 位，合成 3 位数的值。

CML/16 位取反传送

CML(P)

对(s)中指定的 BIN16 位数据进行逐位取反后，将其结果传送到(d)中指定的软元件。

-[CML (s) (d)]

内容、范围、数据类型

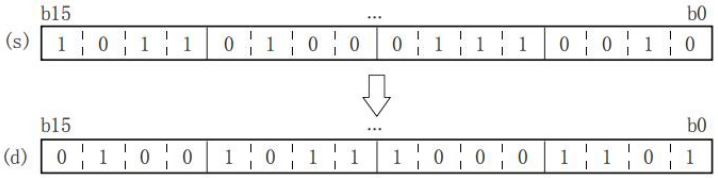
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	取反的数据或存储了数据的软元件编号	[-32768,32767]	有符号 BIN16	ANY16_S	INT
(d)	存储取反结果的软元件编号	--	有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
CML	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

- 对(s)中指定的 BIN16 位数据进行逐位取反后，将其结果传送到(d)中指定的软元件。



- 指定位数的软元件的位数为 4 点的情况，不影响其他位。

错误码

错误码	内容
4085H	读应用指令中(s)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

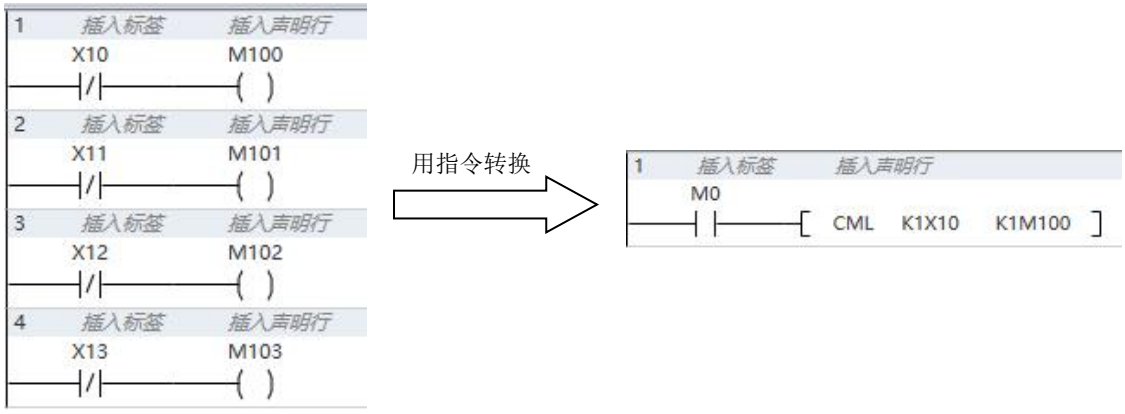
示例

示例 1：

M0 置位时，将 D0 的值取反传送到 D2 的值。

示例 2：

反转输入的获取：



DCML/32 位取反传送

DCML(P)

对(s)中指定的 BIN32 位数据进行逐位取反后，将其结果传送到(d)中指定的软元件。

-[DCML (s) (d)]

内容、范围、数据类型

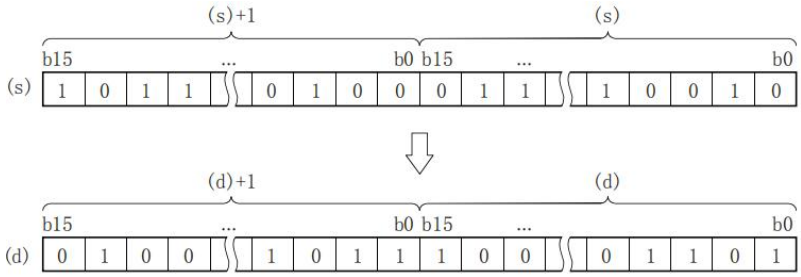
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	取反的数据或存储了数据的软元件编号	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	存储取反结果的软元件编号	--	有符号 BIN32	ANY32_S	DINT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DCML	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●

功能

- 对(s)中指定的 BIN32 位数据进行逐位取反后，将其结果传送到(d)中指定的软元件。



- 指定位数的软元件的位数为 4 点的情况，不影响其他位。

错误码

错误码	内容
4085H	读应用指令中(s)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将(D1，D0)的值取反传送到(D3，D2)的值。

CMP/16 位数据比较输出

CMP(P)

比较(s1)与(s2)中指定的软元件的 BIN16 位数据。

-[CMP (s1) (s2) (d)]

内容、范围、数据类型

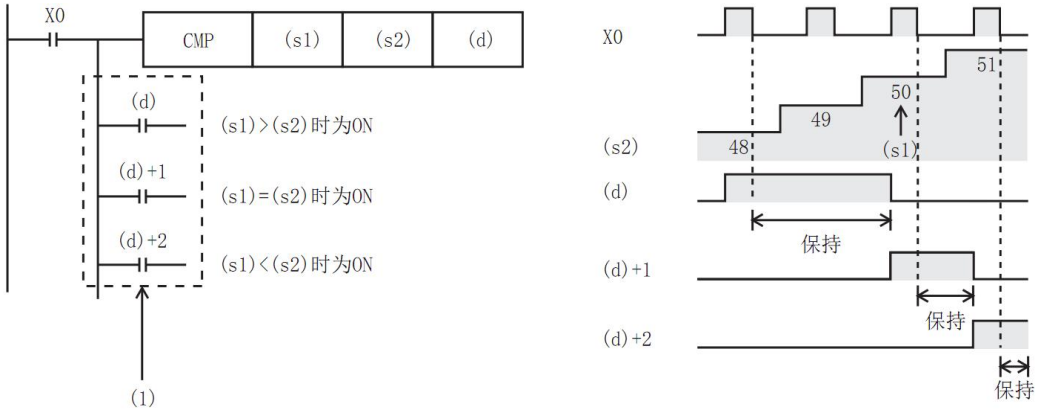
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较值数据或存储了比较值数据的软元件	[-32768,32767]	有符号 BIN16	ANY16_S	INT
(s2)	比较源数据或存储了比较源数据的软元件	[-32768,32767]	有符号 BIN16	ANY16_S	INT
(d)	输出比较结果的起始位软元件		位	ANYBIT_ARRAY	BOOL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
CMP	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3		●	●	●	●					●															●	

功能

- 比较(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据，根据结果(小于、一致、大于)，(d)、(d)+1、(d)+2 中的一项将变为 ON。
- (s1)、(s2)在上述设置数据范围内，作为 BIN 值处理。
- 用代数方法进行大小比较。



(1)：即使指令输入 OFF，不执行 CMP 指令，(d)～(d)+2 也将保持指令输入从 ON 变为 OFF 之前的状态。

注意事项

在起始处占用 3 点（d）中指定的软元件，请注意不要与用于其他控制的软元件重复。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，对比 D0 与 D2 的值：

如果(D0) > (D2) 那么 Y0 为 ON；

如果(D0) = (D2) 那么 Y1 为 ON；

如果(D0) < (D2) 那么 Y2 为 ON。

DCMP/32 位数据比较输出

DCMP(P)

比较(s1)与(s2)中指定的软元件的 BIN16 位数据。

-[DCMP (s1) (s2) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较值数据或存储了比较值数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(s2)	比较源数据或存储了比较源数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	输出比较结果的起始位软元件		位	ANYBIT_ARRAY	BOOL

使用的软元件

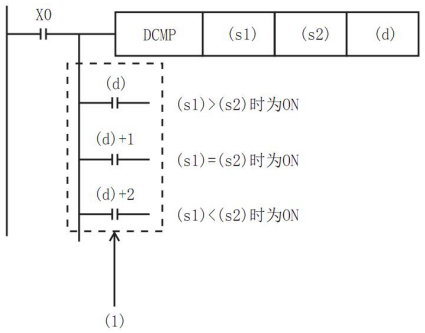
指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DCMP	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●
	参数 3		●	●	●	●					●															●	

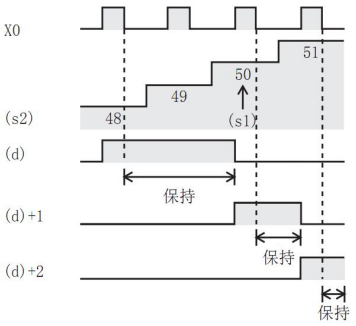
功能

• 比较(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据，根据结果(小于、一致、大于)，(d)、(d)+1、(d)+2 中的一项将变为 ON。

• (s1)、(s2)在上述设置数据范围内，作为 BIN 值处理。

• 用代数方法进行大小比较。





(1)：即使指令输入 OFF，不执行 DCMP 指令，(d)～(d)+2 也将保持指令输入从 ON 变为 OFF 之前的状态。

注意事项

在起始处占用 3 点(d)中指定的软元件。请注意不要与用于其他控制的软元件重复。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，对比(D1，D0)与(D3，D2)的值：

如果(D1，D0) > (D3，D2) 那么 Y0 为 ON;

如果(D1，D0) = (D3，D2) 那么 Y1 为 ON;

如果(D1，D0) < (D3，D2) 那么 Y2 为 ON。

XCH/16 位数据交换

XCH(P)

对(d1)与(d2)的 BIN16 位数据进行交换。

-[XCH (d1) (d2)]

内容、范围、数据类型

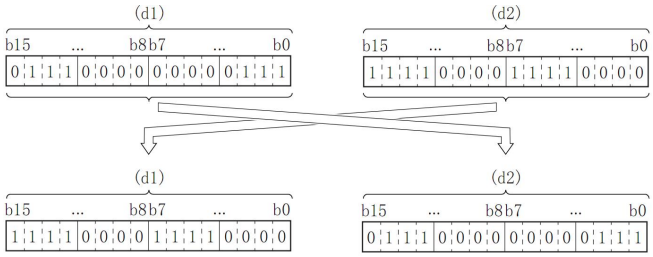
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d1)	存储交换数据的起始软元件	[-32768,32767]	有符号 BIN16	ANY16_S	INT
(d2)	存储交换数据的起始软元件	[-32768,32767]	有符号 BIN16	ANY16_S	INT

使用的软元件

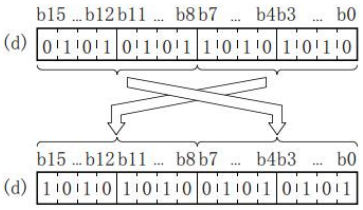
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
XCH	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

- 对(d1)与(d2)的 BIN16 位数据进行交换。



- 在 SM160 为 ON 的状态下执行指令时，如果(d1)和(d2)的软元件编号一致。交换字软元件的高 8 位(字节)和低 8 位(字节)。



注意事项

如果使用连续执行指令，则每个运算周期都将进行转换。

错误码

错误码	内容
4084H	交换模式下,(d1)(d2)中的软元件不同。
4085H	读应用指令中(d1)(d2)输出结果超出软元件范围。
4086H	写应用指令中(d1)(d2)输出结果超出软元件范围。

示例

①当 M0 复位时

置位 M1：D0 的值与 D2 的值互相交换。

②当 M0 置位时

置位 M1：D0 的高 8 位(字节)和低 8 位(字节)相互交换。

福州富昌维控电子科技有限公司

DXCH/32 位数据交换

DXCH(P)

对(d1)与(d2)的 BIN32 位数据进行交换。

-[DXCH (d1) (d2)]

内容、范围、数据类型

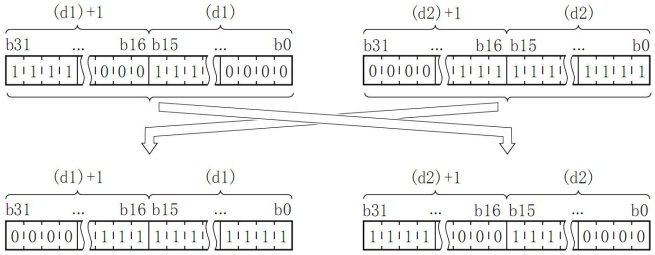
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d1)	存储交换数据的起始软元件	[-2147483647,2147483647]	有符号 BIN32	ANY32_S	DINT
(d2)	存储交换数据的起始软元件	[-2147483647,2147483647]	有符号 BIN32	ANY32_S	DINT

使用的软元件

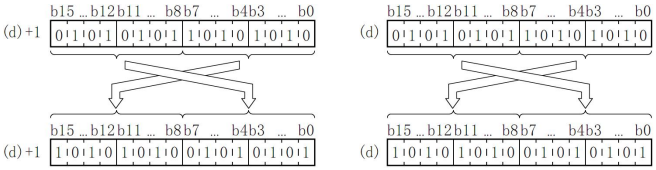
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DXCH	参数 1												●	●	●	●	●	●	●	●	●	●				●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●

功能

- 对(d1)、(d1)+1 与(d2)、(d2)+1 的 BIN32 位数据进行交换。



- 在 SM160 为 ON 的状态下执行指令时，如果(d1)和(d2)的软元件编号一致。交换字软元件(d1)，(d1+1)的高 8 位(字节)和低 8 位(字节)。



注意事项

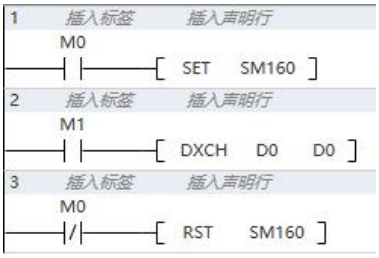
如果使用连续执行指令，则每个运算周期都将进行转换。

错误码

错误码	内容
4084H	交换模式下，(d1)(d2)中的软元件不同。
4085H	读应用指令中(d1)(d2)输出结果超出软元件范围。
4086H	写应用指令中(d1)(d2)输出结果超出软元件范围。

示例

例 1：当 M0 置位时，置位 M1：D0 软元件的高 8 位(字节)和低 8 位(字节)相互交换，D1 软元件的高 8 位(字节)和低 8 位(字节)相互交换。



例 2：当 M0 复位时，置位 M1：(D1，D0)的值与(D3，D2)的值互相交换。

ZCP/16 位数据区间比较

ZCP(P)

将(s1)中指定的软元件的 BIN16 位数据及(s2)中指定的软元件的 BIN16 位数据的值(带宽)，与比较源(s3)中指定的软元件的 BIN16 位数据进行比较，将其结果(下、区域内、上)输出至(d)中指定的软元件以后。

-[ZCP (s1) (s2) (s3) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	下限的比较值数据或存储了比较值数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(s2)	上限的比较值数据或存储了比较值数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(s3)	比较源数据或存储了比较源数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(d)	输出比较结果的起始位软元件		位	ANYBIT_ARRAY	BOOL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ZCP	参数 1											●	●	●	●	●	●	●	●	●			●	●	●	●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●	●	●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●	●	●	●
	参数 4		●	●	●	●					●															●	

功能

• 将(s1)中指定的软元件的 BIN16 位数据及(s2)中指定的软元件的 BIN16 位数据的值(带宽)，与比较源(s3)中指定的软元件的 BIN16 位数据进行比较，根据结果(下、区域内、上)，(d)、(d)+1、(d)+2 中的一项将变为 ON。(s1)、(s2)、(s3)在上述设置数据范围内，作为 BIN 值处理。用代数方法进行大小比较。

• 用代数方法进行大小比较。

(1)：即使指令输入 OFF，不执行 ZCP 指令，(d)～(d)+2 也将保持指令输入从 ON 变为 OFF 之前的状态。

注意事项

• 请将下比较值(s1)、设置为小于上比较值(s2)的值。

• 当(s1)大于(s2)时，将认为(s2)=(s1)进行处理。

• 在起始处占用 3 点(d)中指定的软元件。请注意不要与用于其他控制的软元件重复。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)(s3)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，对比 D0 与是否在 0 到 1000 之间：
如果(D0) > (1000) ，那么 Y2 为 ON;
如果(0) ≤ (D0) ≤(1000)，那么 Y1 为 ON;
如果(D0) < (0)，那么 Y0 为 ON。

DZCP/32 位数据区间比较

DZCP(P)

将(s1)中指定的软元件的 BIN32 位数据及(s2)中指定的软元件的 BIN32 位数据的值(带宽)，与比较源(s3)中指定的软元件的 BIN32 位数据进行比较，将其结果(下、区域内、上)输出至(d)中指定的软元件以后。

-[DZCP (s1) (s2) (s3) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	下限的比较值数据或存储了比较值数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(s2)	上限的比较值数据或存储了比较值数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(s3)	比较源数据或存储了比较源数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	输出比较结果的起始位软元件		位	ANYBIT_ARRAY	BOOL

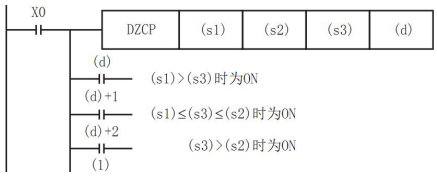
使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DZCP	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
	参数 3											●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
	参数 4		●	●	●	●					●															●	

功能

• 将(s1)中指定的软元件的 BIN32 位数据及(s2)中指定的软元件的 BIN32 位数据的值(带宽)，与比较源(s3)中指定的软元件的 BIN32 位数据进行比较，根据结果(下、区域内、上)，(d)、(d)+1、(d)+2 中的一项将变为 ON。(s1)、(s2)、(s3)在上述设置数据范围内，作为 BIN 值处理。用代数方法进行大小比较。

• 用代数方法进行大小比较。



(1)：即使指令输入 OFF，不执行 ZCP 指令，(d)～(d)+2 也将保持指令输入从 ON 变为 OFF 之前的状态。

注意事项

• 请将下比较值(s1)、设置为小于上比较值(s2)的值。

• 当(s1)大于(s2)时，将认为(s2)=(s1)进行处理。

• 在起始处占用 3 点(d)中指定的软元件。请注意不要与用于其他控制的软元件重复。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)(s3)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，对比 D0 与是否在 0 到 100000 之间：
如果(D0) > (100000)，那么 Y2 为 ON；
如果(0) ≤ (D0) ≤ (100000)，那么 Y1 为 ON；
如果(D0) < (0) ，那么 Y0 为 ON。

7.2 循环移位指令

ROR/16 位循环右移

ROR(P)

将（d）中指定的软元件的 BIN16 位数据，在不包含进位标志的状况下进行（n）位右移。

-[ROR (d) (n)]

内容、范围、数据类型

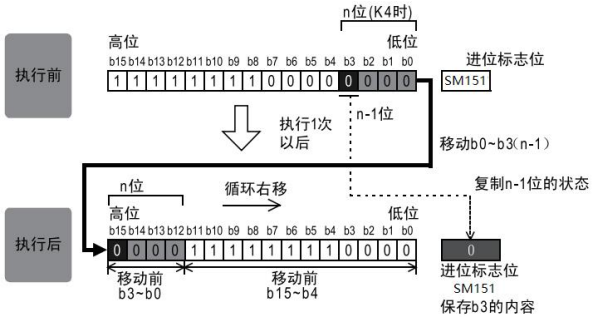
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	循环右移的软元件起始编号	-	有符号 BIN16 位	ANY16	INT
(n)	循环右移的次数	[0,15]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ROR	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件的 BIN16 位数据，在不包含进位标志的状况下进行（n）位右移。进位标志根据 ROR(P)执行前的状态而处于 ON 或 OFF 状态。



• （n）指定[0,15]。（n）中指定了 16 以上的值的情况下，以（n）÷16 的余数值进行右移。例如（n）=18 时，18÷16=1 余 2，因此进行 2 位右移。

相关软元件

软元件	名称	内容
SM151	进位	最后从最低位移出的位为 1 时置 ON。

注意事项

• 请勿将右移的位数（n）设置为负值。

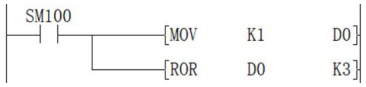
• 连续执行型指令（ROR、RCR）的情况下，每个扫描时间（运算周期）将执行向右移位，应加以注意。

• 在（d）中指定位数指定软元件的场合，仅 K4(16 位指令)或 K8(32 位指令)有效。（例如 K4Y10、K8M0）。

错误码

错误码	内容
4084H	(n)中指定了负值。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



将 D0 软元件中的 1 右移 3 位得到 8192。

DROR/32 位循环右移

DROR(P)

将（d）中指定的软元件的 BIN32 位数据，在不包含进位标志的状况下进行（n）位右移。

-[DROR (d) (n)]

内容、范围、数据类型

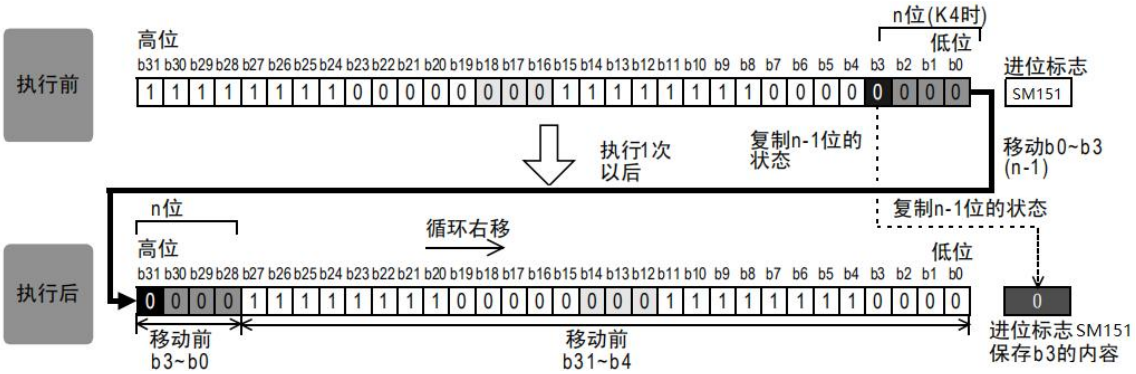
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	循环右移的软元件起始编号	-	有符号 BIN32 位	ANY32	DINT
(n)	循环右移的次数	[0,31]	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DROR	参数 1												●	●	●	●	●	●	●	●	●	●				●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●

功能

• 将（d）中指定的软元件的 BIN32 位数据，在不包含进位标志的状况下进行（n）位右移。进位标志根据 DROR(P) 执行前的状态而处于 ON 或 OFF 状态。



• （n）指定[0,31]。（n）中指定了 32 以上的值的情况下，以（n）÷32 的余数值进行右移。例如（n）=34 时，34÷32=1 余 2，因此进行 2 位右移。

相关软元件

软元件	名称	内容
SM151	进位	最后从最低位移出的位为 1 时置 ON。

注意事项

• 请勿将右移的位数（n）设置为负值。

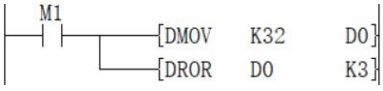
• 连续执行型指令（ROR、RCR）的情况下，每个扫描时间（运算周期）将执行向右移位，应加以注意。

• 在（d）中指定位数指定软元件的场合，仅 K4(16 位指令)或 K8(32 位指令)有效。（例如 K4Y10、K8M0）。

错误码

错误码	内容
4084H	(n)中指定了负值。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M1 上升沿触发后将 D0 软元件的值 32 右移 3 位得到 4。

RCR/16 位带进位循环右移

RCR(P)

将（d）中指定的软元件的 BIN16 位数据，在包含进位标志的状况下进行（n）位右移。

-[RCR (d) (n)]

内容、范围、数据类型

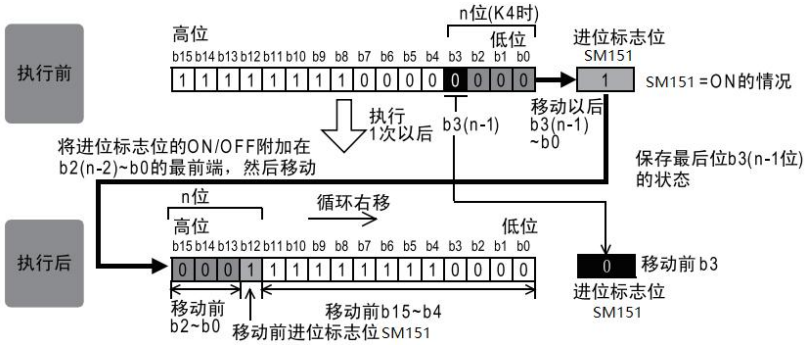
参数	内容	范围	数据类型	数据类型(标签)	自定义变类型
(d)	循环右移的软元件起始编号	-	有符号 BIN16 位	ANY16	INT
(n)	循环右移的次数	[0,15]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RCR	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件的 BIN16 位数据，在包含进位标志的状况下进行（n）位右移。进位标志根据 RCR(P)执行前的状态而处于 ON 或 OFF 状态。



• （n）指定[0,15]。（n）中指定了 16 以上的值的情况下，以（n）÷16 的余数值进行右移。例如（n）=18 时，18÷16=1 余 2，因此进行 2 位右移。

相关软元件

软元件	名称	内容
SM151	进位	最后从最低位移出的位为 1 时置 ON。

注意事项

• 请勿将右移的位数（n）设置为负值。

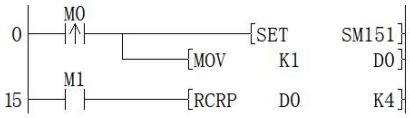
• 连续执行型指令（ROR、RCR）的情况下，每个扫描时间（运算周期）将执行向右移位，应加以注意。

• 在（d）中指定位数指定软元件的场合，仅 K4(16 位指令)或 K8(32 位指令)有效。（例如 K4Y10、K8M0）。

错误码

错误码	内容
4084H	(n)中指定了负值。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M0 上升沿触发后，进位标志 SM151 置 ON，D0 赋值 1。当 M1=ON 时，将 D0 软元件中的值带进位右移 4 位得到 12288。

DRCR/32 位带进位循环右移

DRCR(P)

将（d）中指定的软元件的 BIN32 位数据，在包含进位标志的状况下进行（n）位右移。

-[DRCR (d) (n)]

内容、范围、数据类型

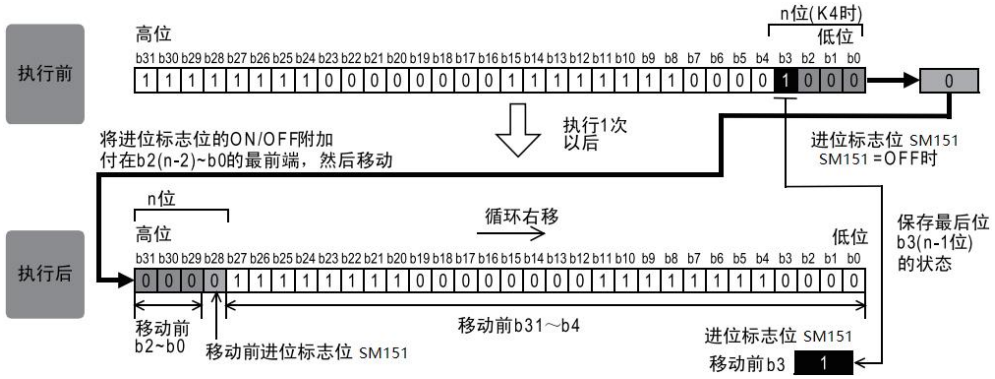
参数	内容	范围	数据类型	数据类型(标签)	自定义变类型
(d)	循环右移的软元件起始编号	-	有符号 BIN32 位	ANY32	DINT
(n)	循环右移的次数	[0,31]	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DRCR	参数 1												●	●	●	●	●	●	●	●	●	●				●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●

功能

• 将（d）中指定的软元件的 BIN32 位数据，在包含进位标志的状况下进行（n）位右移。进位标志根据 DRCR(P)执行前的状态而处于 ON 或 OFF 状态。



• （n）指定[0,31]。（n）中指定了 32 以上的值的情况下，以（n）÷32 的余数值进行右移。例如（n）=34 时，34÷32=1 余 2，因此进行 2 位右移。

相关软元件

软元件	名称	内容
SM151	进位	最后从最低位移出的位为 1 时置 ON。

注意事项

• 请勿将右转的位数（n）设置为负值。

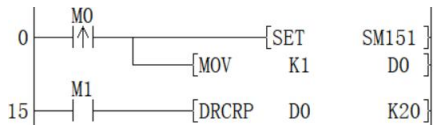
• 连续执行型指令（DROR、DRCR）的情况下，每个扫描时间（运算周期）将执行向右移位，应加以注意。

• 在（d）中指定位数指定软元件的场合，仅 K4(16 位指令)或 K8(32 位指令)有效。（例如 K4Y10、K8M0）。

错误码

错误码	内容
4084H	(n)中指定了负值。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M0 上升沿触发后，进位标志 SM151 置 ON，D0 赋值 1。当 M1=ON 时，将 D0 软元件中的值带进位右移 20 位得到 12288。

ROL/16 位循环左移

ROL(P)

将（d）中指定的软元件的 BIN16 位数据，在不包含进位标志的状况下进行（n）位左移。

-[ROL (d) (n)]

内容、范围、数据类型

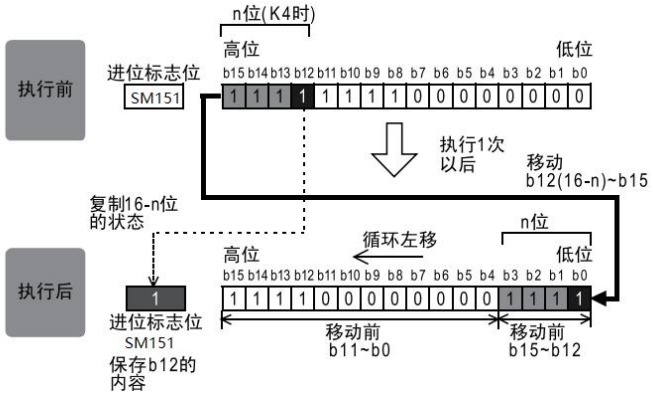
参数	内容	范围	数据类型	数据类型(标签)	自定义变类型
(d)	循环左移的软元件起始编号	-	有符号 BIN16 位	ANY16	INT
(n)	循环左移的次数	[0,15]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ROL	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件的 BIN16 位数据，在不包含进位标志的状况下进行（n）位左移。进位标志根据 ROL(P)执行前的状态而处于 ON 或 OFF 状态。



• （n）指定[0,15]。（n）中指定了 16 以上的值的情况下，以（n）÷16 的余数值进行左移。例如（n）=18 时，18÷16=1 余 2，因此进行 2 位左移。

相关软元件

软元件	名称	内容
SM151	进位	最后从最高位移出的位为 1 时置 ON。

注意事项

• 请勿将左移的位数（n）设置为负值。

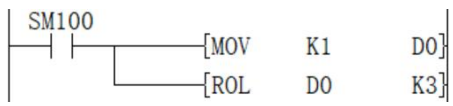
• 连续执行型指令（ROL、RCL）的情况下，每个扫描时间（运算周期）将执行向左移位，应加以注意。

• 在（d）中指定位数指定软元件的场合，仅 K4(16 位指令)或 K8(32 位指令)有效。（例如 K4Y10、K8M0）。

错误码

错误码	内容
4084H	(n)中指定了负值。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



将 D0 软元件中的 1 左移 3 位得到 8。

DROL/32 位循环左移

DROL(P)

将（d）中指定的软元件的 BIN32 位数据，在不包含进位标志的状况下进行（n）位左移。

-[DROL (d) (n)]

内容、范围、数据类型

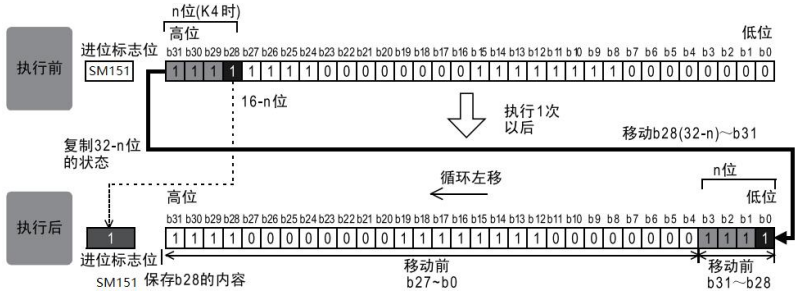
参数	内容	范围	数据类型	数据类型(标签)	自定义变类型
(d)	循环左移的软元件起始编号	-	有符号 BIN32 位	ANY32	DINT
(n)	循环左移的次数	[0,31]	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DROL	参数 1												●	●	●	●	●	●	●	●	●	●				●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●

功能

• 将（d）中指定的软元件的 BIN32 位数据，在不包含进位标志的状况下进行（n）位左移。进位标志根据 DROL(P)执行前的状态而处于 ON 或 OFF 状态。



• （n）指定[0,31]。（n）中指定了 32 以上的值的情况下，以（n）÷32 的余数值进行左移。例如（n）=34 时，34÷32=1 余 2，因此进行 2 位左移。

相关软元件

软元件	名称	内容
SM151	进位	最后从最高位移出的位为 1 时置 ON。

注意事项

• 请勿将左移的位数（n）设置为负值。

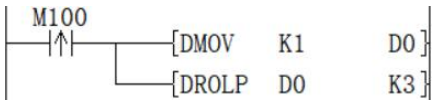
• 连续执行型指令（ROL、RCL）的情况下，每个扫描时间（运算周期）将执行向左移位，应加以注意。

• 在（d）中指定位数指定软元件的场合，仅 K4(16 位指令)或 K8(32 位指令)有效。（例如 K4Y10、K8M0）。

错误码

错误码	内容
4084H	(n)中指定了负值。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



将 D0 软元件中的 1 左移 3 位得到 8。

RCL/16 位带进位循环左移

RCL(P)

将（d）中指定的软元件的 BIN16 位数据，在包含进位标志的状况下进行（n）位左移。

-[RCL (d) (n)]

内容、范围、数据类型

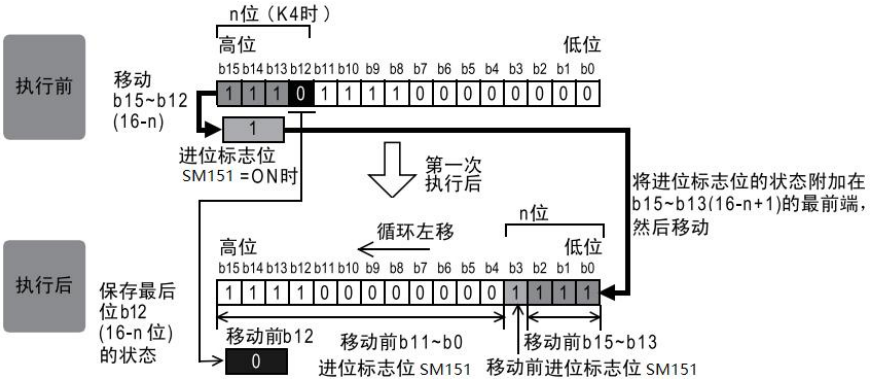
参数	内容	范围	数据类型	数据类型(标签)	自定义变类型
(d)	循环左移的软元件起始编号	-	有符号 BIN16 位	ANY16	INT
(n)	循环左移的次数	[0,15]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RCL	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件的 BIN16 位数据，在包含进位标志的状况下进行（n）位左移。进位标志根据 RCL(P)执行前的状态而处于 ON 或 OFF 状态。



• （n）指定[0,15]。（n）中指定了 16 以上的值的情况下，以（n）÷16 的余数值进行左移。例如（n）=18 时，18÷16=1 余 2，因此进行 2 位左移。

相关软元件

软元件	名称	内容
SM151	进位	最后从最高位移出的位为 1 时置 ON。

注意事项

• 请勿将左移的位数（n）设置为负值。

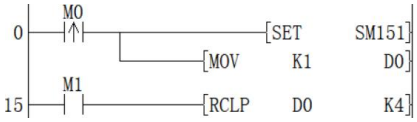
• 连续执行型指令（ROL、RCL）的情况下，每个扫描时间（运算周期）将执行向左移位，应加以注意。

• 在（d）中指定位数指定软元件的场合，仅 K4(16 位指令)或 K8(32 位指令)有效。（例如 K4Y10、K8M0）。

错误码

错误码	内容
4084H	(n)中指定了负值。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M0 上升沿触发后，进位标志 SM151 置 ON，D0 赋值 1。
当 M1=ON 时，将 D0 软元件中的值带进位左移 4 位得到 24。

DRCL/32 位带进位循环左移

DRCL(P)

将（d）中指定的软元件的 BIN32 位数据，在包含进位标志的状况下进行（n）位左移。

-[DRCL (d) (n)]

内容、范围、数据类型

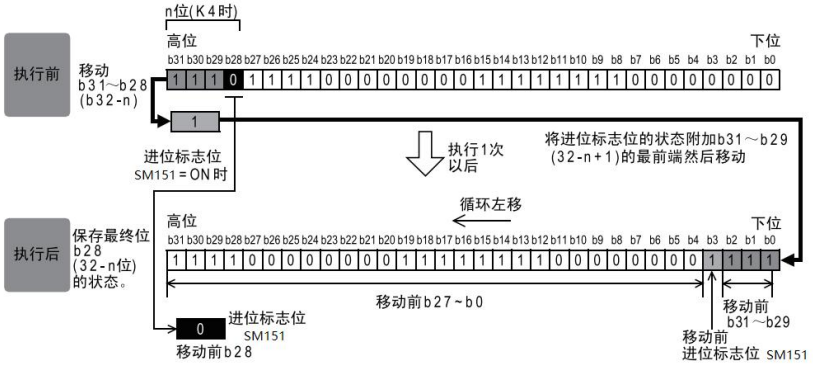
参数	内容	范围	数据类型	数据类型(标签)	自定义变类型
(d)	循环左移的软元件起始编号	-	有符号 BIN32 位	ANY32	DINT
(n)	循环左移的次数	[0,31]	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DRCL	参数 1												●	●	●	●	●	●	●	●	●	●				●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●

功能

• 将（d）中指定的软元件的 BIN32 位数据，在包含进位标志的状况下进行（n）位左移。进位标志根据 RCL(P)执行前的状态而处于 ON 或 OFF 状态。



• （n）指定[0,31]。（n）中指定了 32 以上的值的情况下，以（n）÷32 的余数值进行左移。例如（n）=34 时，34÷32=1 余 2，因此进行 2 位左移。

相关软元件

软元件	名称	内容
SM151	进位	最后从最高位移出的位为 1 时置 ON。

注意事项

• 请勿将左移的位数（n）设置为负值。

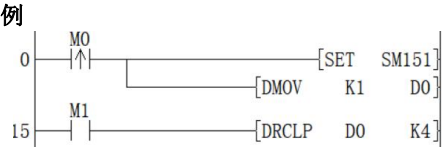
• 连续执行型指令（ROL、RCL）的情况下，每个扫描时间（运算周期）将执行向左移位，应加以注意。

• 在（d）中指定位数指定软元件的场合，仅 K4(16 位指令)或 K8(32 位指令)有效。（例如 K4Y10、K8M0）。

错误码

错误码	内容
4084H	(n)中指定了负值。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M0 上升沿触发后，进位标志 SM151 置 ON，D0 赋值 1。
当 M1=ON 时，将 D0 软元件中的值带进位左旋 4 位得到 24。

福州富昌维控电子科技有限公司

SFTR/n 位数据的 n 位右移

SFTR(P)

将（d）中指定的软元件开始（n1）位的数据向右移位（n2）位。

-[SFTR （s）(d) (n1)（n2）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	移位后存储移位数据的软元件起始编号	-	位	ANY_BOOL	--
(d)	移位的软元件起始编号	-	位	ANY_BOOL	--
(n1)	移位数据的数据长	[0,32767]	有符号 BIN16 位	ANY16	INT
(n2)	移位数	[0,32767]	有符号 BIN16 位	ANY16	INT

使用的软元件

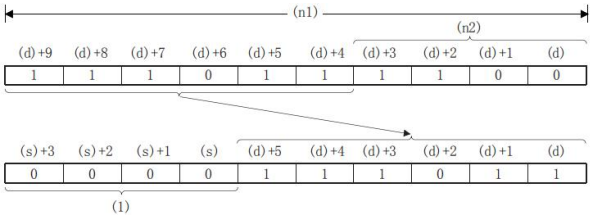
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SFTR	参数 1	●	●	●	●	●					●															●	●
	参数 2		●	●	●	●					●															●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件开始（n1）位的数据向右移位（n2）位。移位后，将（s）开始的（n2）点传送到（d）+（n1~n2）开始的（n2）点中。

• （s）中指定了 K0 的情况下，移位后的（d）+（n1~n2）开始（n2）点的位设置为 0。

• （s）中指定了 K1 的情况下，移位后的（d）+（n1~n2）开始（n2）点的位设置为 1。



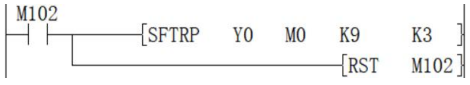
（1）：（s）=K0 的情况下，变为 0。

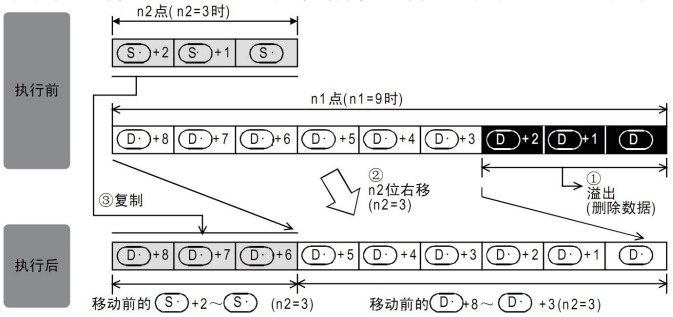
错误码

错误码	内容
4084H	（n1）（n2）中指定的值超出范围时。[0,32767]
	（n1）、（n2）中指定的值为（n1）<（n2）时。
4085H	读应用指令(s)、(d)、(n1)、(n2)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

对于以 M0 起始的 n1=9 位（移位寄存器的长度）数据，右移 n2=3 位。移位后，将 Y0 开始 n2=3 位数据传送到从 M6 开始的 n2=3 位中。





SFTL/n 位数据的 n 位左移

SFTL(P)

将（d）中指定的软元件开始（n1）位的数据向左移位（n2）位。

-[SFTL （s）(d) (n1)（n2）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	移位后存储移位数据的软元件起始编号	-	位	ANY_BOOL	--
(d)	移位的软元件起始编号	-	位	ANY_BOOL	--
(n1)	移位数据的数据长	[0,32767]	有符号 BIN16 位	ANY16	INT
(n2)	移位数	[0,32767]	有符号 BIN16 位	ANY16	INT

使用的软元件

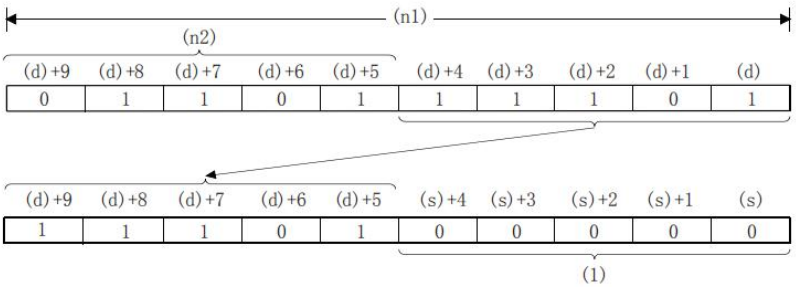
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SFTL	参数 1	●	●	●	●	●					●															●	●
	参数 2		●	●	●	●					●															●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件开始（n1）位的数据向左移位（n2）位。移位后，将（s）开始的（n2）点传送到（d）+开始的（n2）点中。

• （s）中指定了 K0 的情况下，移位后的（d）开始（n2）点的位设置为 0。

• （s）中指定了 K1 的情况下，移位后的（d）开始（n2）点的位设置为 1。



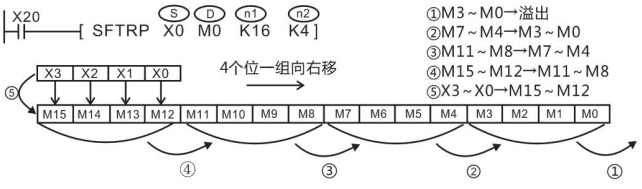
（1）：（s）=K0 的情况下，变为 0。

错误码

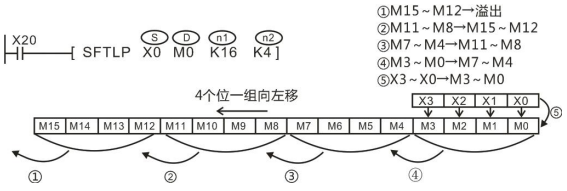
错代码	内容
4084H	（n1）（n2）中指定的值超出范围时。[0,32767]
	（n1）、（n2）中指定的值为（n1）＜（n2）时。
4085H	读应用指令(s)、(d)、(n1)、(n2)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

例 1：



例 2：



WSFR/n 字数据的 n 字右移

WSFR(P)

将（d）中指定的软元件开始（n1）位的数据向右移位（n2）字。

-[WSFR （s）(d) (n1)（n2）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变类型
(s)	移位后存储移位数据的软元件起始编号	-	字	ANY_BOOL	--
(d)	移位的软元件起始编号	-	字	ANY_BOOL	--
(n1)	移位数据的数据长	[0,32767]	有符号 BIN16 位	ANY16	INT
(n2)	移位数	[0,32767]	有符号 BIN16 位	ANY16	INT

使用的软元件

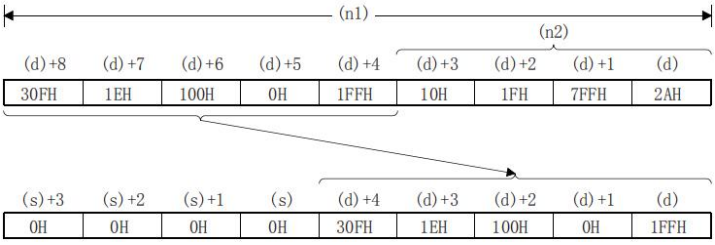
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
WSFR	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件开始（n1）字的数据向右移位（n2）字。移位后，将（s）开始的（n2）点传送到（d）+（n1~n2）开始的（n2）点中。

• （s）中指定了 K 的情况下，移位后的（d）+（n1~n2）开始（n2）点的软元件设置为指定的值。

• (n1)或（n2）中指定的值为 0 的情况下，将变为无处理。

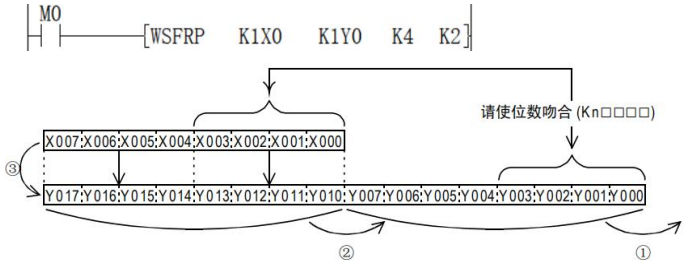


错误码

错误码	内容
4084H	(n1)、(n2)中指定的值超出范围时。[0,32767]
	(n1)、(n2)中指定的值为(n1)<(n2)时。
	(s)、(d)都指定 KnM,KnX,KnS 时，n 取值不等。
4085H	读应用指令(s)、(d)、(n1)、(n2)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

（s）、（d）在位数指定软元件中做相同的倍数指定。该程序实现将 Y0~Y7 位右移去除，Y10~Y17 右移到 Y0~Y7，再将 X0~X7 存储到 Y10~Y17。



WSFL/n 字数据的 n 字左移

WSFL(P)

将（d）中指定的软元件开始（n1）位的数据向左移位（n2）字。

-[WSFL （s）(d) (n1)（n2）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	移位后存储移位数据的软元件起始编号	-	字	ANY_BOOL	--
(d)	移位的软元件起始编号	-	字	ANY_BOOL	--
(n1)	移位数据的数据长	[0,32767]	有符号 BIN16 位	ANY16	INT
(n2)	移位数	[0,32767]	有符号 BIN16 位	ANY16	INT

使用的软元件

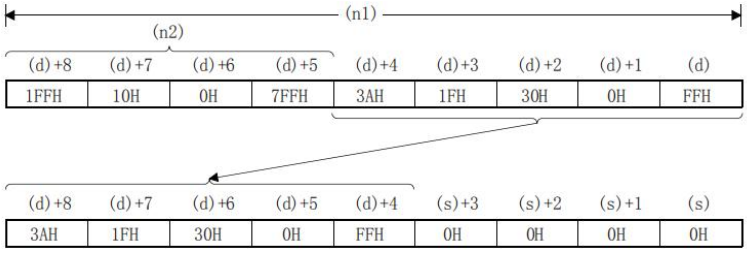
指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
WSFL	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件开始（n1）字的数据向左移位（n2）字。移位后，将（s）开始的（n2）点传送到（d）开始的（n2）点中。

• （s）中指定了 K 的情况下，移位后的（d）开始（n2）点的软元件设置为指定的值。

• (n1)或（n2）中指定的值为 0 的情况下，将变为无处理。

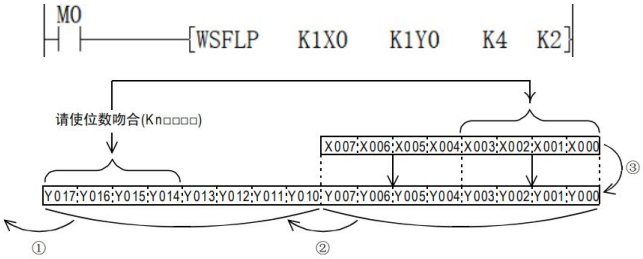


错误码

错误码	内容
4084H	(n1)、(n2)中指定的值超出范围时。[0,32767]
	(n1)、(n2)中指定的值为(n1)<(n2)时。
	(s)、(d)都指定 KnM,KnX,KnS 时，n 取值不等。
4085H	读应用指令(s)、(d)、(n1)、(n2)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

（s）、（d）在位数指定软元件中做相同的倍数指定。该程序实现将 Y10~Y17 高位左移去除，Y0~Y7 左移到 Y10~Y17，再将 X0~X7 存储到 Y0~Y7。



SFR/16 位数据的 n 位右移位

SFR(P)

将（d）中指定的软元件的 16 位数据右移（n）位。

-[SFR (d) (n)]

内容、范围、数据类型

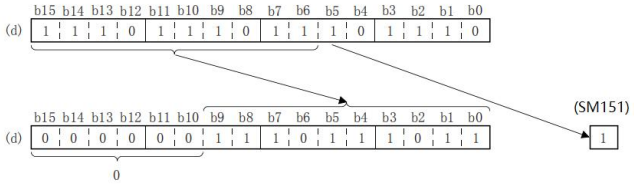
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储移位数据的软元件起始编号	-	有符号 BIN16 位	ANY16	INT
(n)	移位次数	[0, 15]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SFR	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●

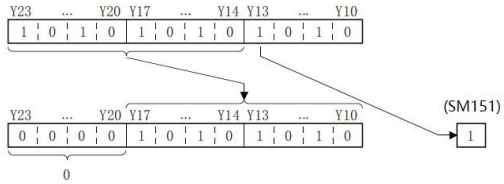
功能

（n）=6 的情况如下



将（d）中指定的软元件的 16 位数据从最高位开始右移（n）位。从最高位开始的（n）位将变为 0。

（n）=4 的情况下



（d）中指定了位软元件的情况下，以位数指定中指定的软元件范围进行右移位。

• （n）指定[0, 15]。（n）中指定了 16 以上的值的情况下，将以（n）÷16 的余数值向左移位。例如，（n）=18 时，18÷16=1 余 2，因此向右移位 2 位。

相关软元件

软元件	名称	内容
SM151	进位	根据 n-1 位的状态（1/0）置为 ON/OFF

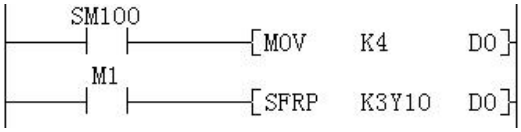
错误码

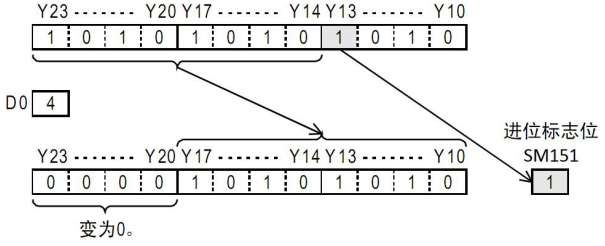
错误码	内容
4084H	(n)中指定了负值时。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

M1 为 ON 时：

将 Y10~Y23 的内容按照 D0 中指定的位数右移。





DSFR/n 字数据的 1 字右移位

DSFR(P)

将（d）中指定的软元件开始（n）点的数据向右移位 1 字。

-[DSFR (d) (n)]

内容、范围、数据类型

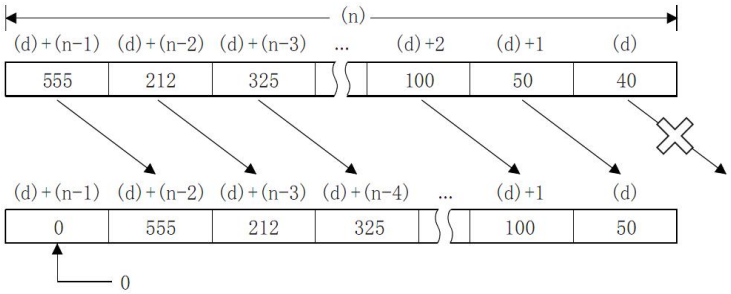
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储移位数据的软元件起始编号	-	有符号 BIN16 位	ANY16	DINT
(n)	移位的软元件数	[0,32767]	有符号 BIN16 位	ANY16	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSFR	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件开始（n）点的数据向右移位 1 字。



• （d）+（n-1）中指定的软元件将变为 0。

注意事项

• 在（d）中通过位软元件的位数指定进行软元件编号指定的情况下，软元件编号应指定 16 的倍数（0、16、32、64……），位数应仅指定 K4。位数不为 K4 时，以 K4 处理。

错误码

错误码	内容
4084H	(n)中指定的数值超出下述范围时。[0,32767]
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

M1 为 ON 时，将 D0~D4 的内容右移 1 字（D1→D0，D2→D1，D3→D2，D4→D3，D4 置 0）。

执行前：

软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
D1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
D2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
D3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4
D4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	5
D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

执行后：

软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
D1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
D2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4
D3	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	5
D4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SFL/16 位数据的 n 位左移位

SFL(P)

将（d）中指定的软元件的 16 位数据左移（n）位。

-[SFL (d) (n)]

内容、范围、数据类型

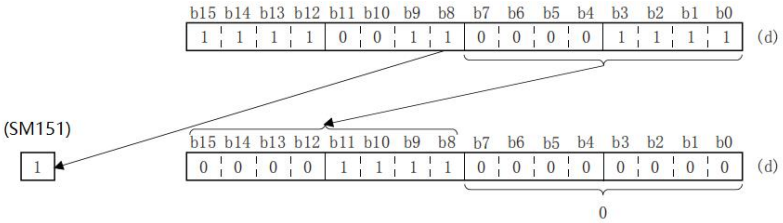
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储移位数据的软元件起始编号	-	有符号 BIN16 位	ANY16	INT
(n)	移位次数	[0,15]	有符号 BIN16 位	ANY16	INT

使用的软元件

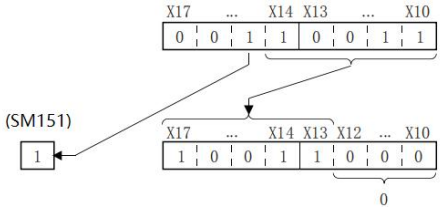
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SFL	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件的 16 位数据从最低位开始左移（n）位。从最低位开始的（n）位将变为 0。（n）=8 的情况如下



• （d）中指定了位软元件的情况下，以位数指定中指定的软元件范围进行左移位。（n）=3 的情况下



• （n）指定[0,15]。（n）中指定了 16 以上的值的情况下，将以（n）÷16 的余数值向左移位。例如，（n）=18 时，18÷16=1 余 2，因此向左移位 2 位。

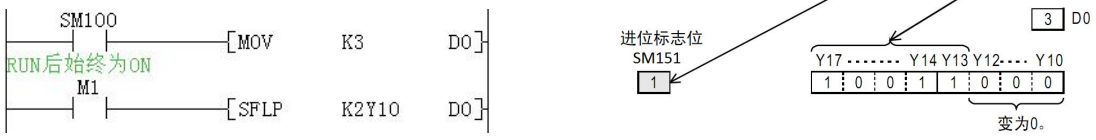
相关软元件

软元件	名称	内容
SM151	进位	根据 n+1 位的状态（1/0）置为 ON/OFF。

错误码

错误码	内容
4084H	(n)中指定了负值时。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例： M1 为 ON 时，将 Y10~Y17 的内容按照 D0 中指定的位数左移。



DSFL/n 字数据的 1 字左移位

DSFL(P)

将（d）中指定的软元件开始（n）点的数据左移 1 字。

-[DSFL (d) (n)]

内容、范围、数据类型

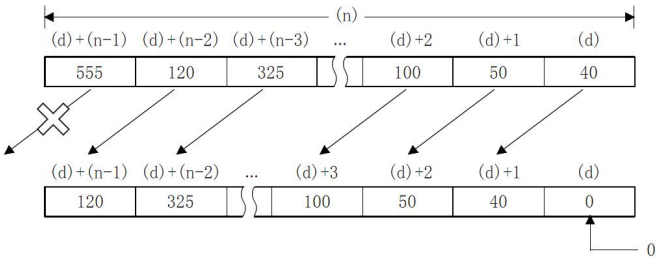
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储移位数据的软元件起始编号	-	有符号 BIN16 位	ANY16	DINT
(n)	移位次数	[0,32767]	有符号 BIN16 位	ANY16	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSFL	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的软元件开始（n）点的数据向左移位 1 字。



• （d）中指定的软元件将变为 0。

注意事项

• 在（d）中通过位软元件的位数指定进行软元件编号指定的情况下，软元件编号应指定 16 的倍数（0、16、32、64……），位数应仅指定 K4。位数不为 K4 时，以 K4 处理。

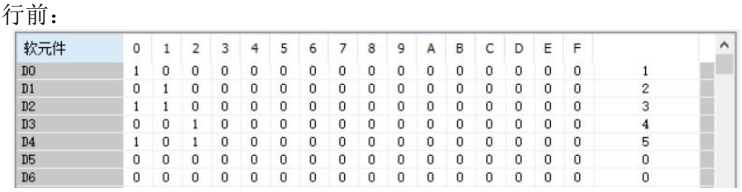
错误码

错误码	内容
4084H	(n)中指定的数值超出下述范围时：[0,32767]。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

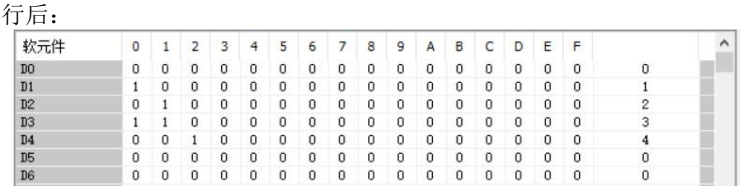
M1 为 ON 时，将 D0~D4 的内容左移 1 字（D3→D4，D2→D3，D1→D2，D0→D1，D0 置 0）。

执行前：



软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
D1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
D2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
D3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4
D4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	5
D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

执行后：



软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
D2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
D3	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
D4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4
D5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

7.3 算术运算指令

ADD/16 位加法运算

ADD(P)

将(s1)中指定的 BIN16 位数据与(s2)中指定的 BIN16 位数据进行加法运算，并将结果存储到(d)中指定的软元件中。

-[ADD (s1) (s2) (d)]

内容、范围、数据类型

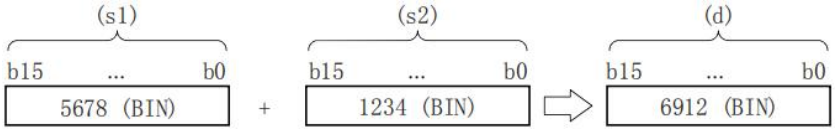
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	加法运算数据或存储了加法运算数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(s2)	加法运算数据或存储了加法运算数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(d)	存储运算结果的软元件		有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ADD	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

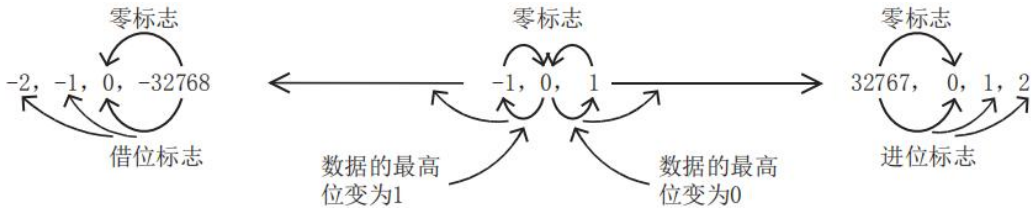
功能

将(s1)中指定的 BIN16 位数据与(s2)中指定的 BIN16 位数据进行加法运算，并将加法运算结果存储到(d)中指定的软元件中。



相关软元件

软元件	名称	内容
SM151	进位	运算结果超过 32,767 时，进位标志将动作(ON)。
SM152	借位	运算结果不满-32,768 时，借位标志将动作(ON)。
SM153	零点	运算结果为 0 时，零标志将动作(ON)。



注意事项

1、源操作数和目标操作数中指定为同一软元件时：

源操作数和目标操作数也可以指定为同一个软元件的编号。

这种情况下，如使用连续执行型的指令(ADD、DADD)，则每个运算周期加法运算的结果都会变化，请注意。

2、采用+1 加法运算程序的 ADD 指令和 INC 指令的区别：

ADD[P]，就是每次 X001 从 OFF 变为 ON 时， D0 的内容上加一运算。

这个指令虽然与后述的 INCP 指令很类似，但是有下面一些内容上的不同。

			ADD/ADDP/DADD/DADDP 指令	INC/INCP/DINC/DINCP 指令
标志位（零、借位、进位）			动作	不动作
运算结果	16 位运算结果	(s) + (+1) = (d)	+32767→0→+1→+2→	+32767→-32768→-32767
		(s) + (-1) = (d)	←-2←-1←0←-32768	——
	32 位运算结果	(s) + (+1) = (d)	+2147483647→0→+1→+2→	+2147483647→-2147483648→-2147483647
		(s) + (-1) = (d)	←-2←-1←0←-2147483648	——

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

将(D0)中的数据加上 10，并将运算结果存储到(D2)，即(D0) + 10 →(D2)。

DADD/32 位加法运算

DADD(P)

将(s1)中指定的 BIN32 位数据与(s2)中指定的 BIN32 位数据进行加法运算，并将结果存储到(d)中指定的软元件中。

-[DADD (s1) (s2) (d)]

内容、范围、数据类型

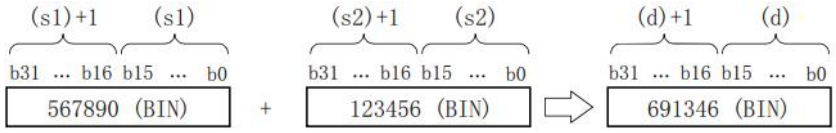
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	加法运算数据或存储了加法运算数据的软元件	[-2147483648,2147483647]	有符号 BIN32	ANY32_S	DINT
(s2)	加法运算数据或存储了加法运算数据的软元件	[-2147483648,2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	存储运算结果的软元件		有符号 BIN32	ANY32_S	DINT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DADD	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

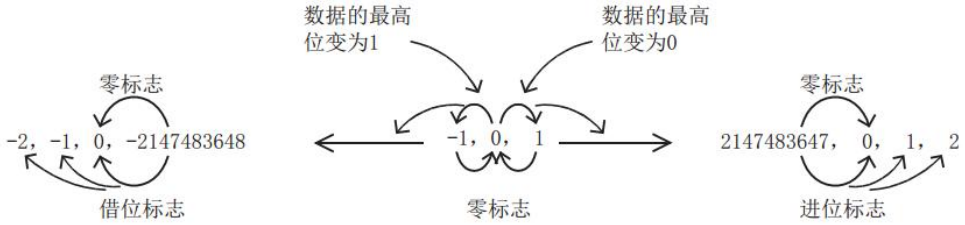
功能

将(s1)中指定的 BIN32 位数据与(s2)中指定的 BIN32 位数据进行加法运算，并将加法运算结果存储到(d)中指定的软元件中。



相关软元件

软元件	名称	内容
SM151	进位	运算结果超过 32,767 时，进位标志将动作(ON)。
SM152	借位	运算结果不满-32,768 时，借位标志将动作(ON)。
SM153	零点	运算结果为 0 时，零标志将动作(ON)。



注意事项

1、源操作数和目标操作数中指定为同一软元件时：

源操作数和目标操作数也可以指定为同一个软元件的编号。

这种情况下，如使用连续执行型的指令(ADD、DADD)，则每个运算周期加法运算的结果都会变化，请注意。

2、采用+1 加法运算程序的 ADD 指令和 INC 指令的区别：

ADD[P]，就是每次 X001 从 OFF 变为 ON 时， D0 的内容上加一运算。

这个指令虽然与后述的 INCP 指令很类似，但是有下面一些内容上的不同。

			ADD/ADDP/DADD/DADDP 指令	INC/INCP/DINC/DINCP 指令
标志位（零、借位、进位）			动作	不动作
运算结果	16 位运算结果	(s) + (+1) = (d)	+32767→0→+1→+2→	+32767→-32768→-32767
		(s) + (-1) = (d)	←-2←-1←0←-32768	——
	32 位运算结果	(s) + (+1) = (d)	+2147483647→0→+1→+2→	+2147483647→-2147483648→-2147483647
		(s) + (-1) = (d)	←-2←-1←0←-2147483648	——

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

将(D1，D0)中的数据加上 100000，并将运算结果存储到(D3，D2)，即(D1，D0) + 100000 →(D3,D2)。

SUB/16 位减法运算

SUB(P)

将(s1)中指定的 BIN16 位数据与(s2)中指定的 BIN16 位数据进行减法运算，并将结果存储到(d)中指定的软元件中。

-[SUB (s1) (s2) (d)]

内容、范围、数据类型

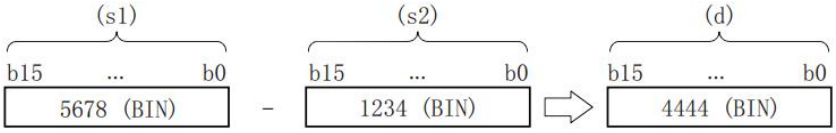
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	减法运算数据或存储了减法运算数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(s2)	减法运算数据或存储了减法运算数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(d)	存储运算结果的软元件		有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SUB	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

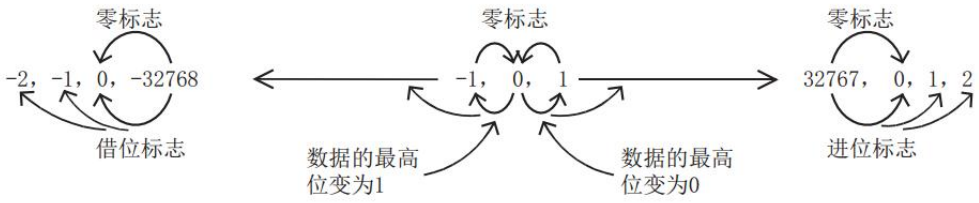
功能

将(s1)中指定的 BIN16 位数据与(s2)中指定的 BIN16 位数据进行减法运算，并将运算结果存储到(d)中指定的软元件中。



相关软元件

软元件	名称	内容
SM151	进位	运算结果超过 32,767 时，进位标志将动作(ON)。
SM152	借位	运算结果不满-32,768 时，借位标志将动作(ON)。
SM153	零点	运算结果为 0 时，零标志将动作(ON)。



注意事项

1、源操作数和目标操作数中指定为同一软元件时：

源操作数和目标操作数也可以指定为同一个软元件的编号。

这种情况下，如使用连续执行型的指令(SUB、DSUB)，则每个运算周期减法运算的结果都会变化，请注意。

2、通过-1 减法程序执行的 SUB(P)指令与-(P)指令及 DEC(P)指令的区别

SUB(P)指令在 X1 每次由 OFF→ON 变化时，D0 内容-1 的程序中，与-(P)指令及后述的 DEC(P)指令相似，但下列内容有区别。

			SUB/SUBP/DSUB/DSUBP 指令	DEC/DECP/DDEC/DDECP 指令
标志位（零、借位、进位）			动作	不动作
运算结果	16 位运算结果	(s)-(+1)=(d)	←-2←-1←0←-32768	-32768→+32767→+32766
		(s)-(-1)=(d)	+32767→0→+1→+2→	——
	32 位运算结果	(s)-(+1)=(d)	←-2←-1←0←-2147483648	-2147483648→+2147483647→+2147483646
		(s)-(-1)=(d)	+2147483647→0→+1→+2→	——

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

将 D0 中的数据减去 10，并将运算结果存储到 D2，即(D0)-10 → (D2)。

DSUB/32 位减法运算

DSUB(P)

将(s1)中指定的 BIN32 位数据与(s2)中指定的 BIN32 位数据进行减法运算，并将结果存储到(d)中指定的软元件中。

-[DSUB (s1) (s2) (d)]

内容、范围、数据类型

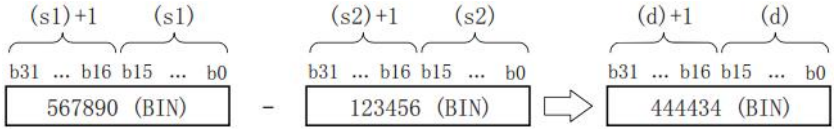
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	减法运算数据或存储了减法运算数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(s2)	减法运算数据或存储了减法运算数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	存储运算结果的软元件		有符号 BIN32	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSUB	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

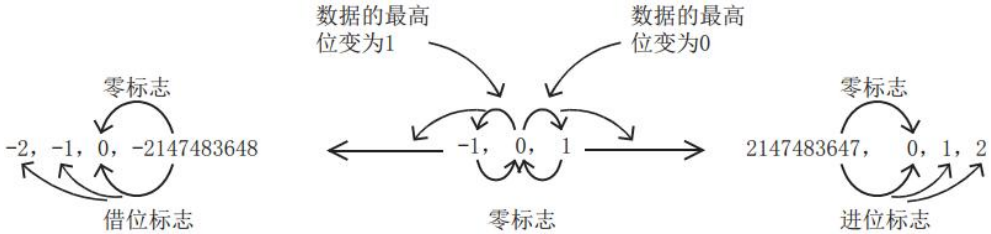
功能

将(s1)中指定的 BIN32 位数据与(s2)中指定的 BIN32 位数据进行减法运算，并将运算结果存储到(d)中指定的软元件中。



相关软元件

软元件	名称	内容
SM151	进位	运算结果超过 2,147,483,647 时，进位标志将动作(ON)。
SM152	借位	运算结果不满-2,147,483,648 时，借位标志将动作(ON)。
SM153	零点	运算结果为 0 时，零标志将动作(ON)。



注意事项

1、源操作数和目标操作数中指定为同一软元件时：

源操作数和目标操作数也可以指定为同一个软元件的编号。

这种情况下，如使用连续执行型的指令(SUB、DSUB)，则每个运算周期减法运算的结果都会变化，请注意。

2、通过-1 减法程序执行的 SUB(P)指令与-(P)指令及 DEC(P)指令的区别

SUB(P)指令在 X1 每次由 OFF→ON 变化时，D0 内容-1 的程序中，与-(P)指令及后述的 DEC(P)指令相似，但下列内容有区别。

			SUB/SUBP/DSUB/DSUBP 指令	DEC/DECP/DDEC/DDECP 指令
标志位（零、借位、进位）			动作	不动作
运算结果	16 位运算结果	(s)-(+1)=(d)	←-2←-1←0←-32768	-32768→+32767→+32766
		(s)-(-1)=(d)	+32767→0→+1→+2→	——
	32 位运算结果	(s)-(+1)=(d)	←-2←-1←0←-2147483648	-2147483648→+2147483647→+2147483646
		(s)-(-1)=(d)	+2147483647→0→+1→+2→	——

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

将(D1,D0)中的数据减去 100000，并将运算结果存储到(D3,D2)，即(D1,D0)-10000 → (D3,D2)。

MUL/16 位乘法运算

MUL(P)

将(s1)中指定的 BIN16 位与(s2)中指定的 BIN16 位进行乘法运算，并将结果存储到(d)中指定的软元件中。

-[MUL (s1) (s2) (d)]

内容、范围、数据类型

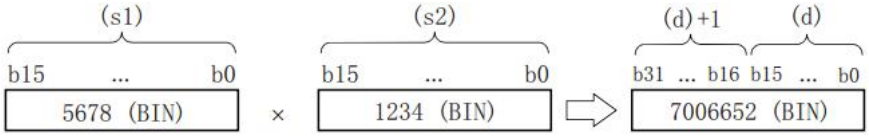
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	乘法运算数据或存储了乘法运算数据的软元件	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(s2)	乘法运算数据或存储了乘法运算数据的软元件	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(d)	存储运算结果的软元件		有符号 BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MUL	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

将(s1)中指定的 BIN16 位数据与(s2)中指定的 BIN16 位数据进行乘法运算，并将运算结果存储到(d)中指定的软元件中。



(d)为位软元件情况下的乘法运算结果

- K1…低位的 4 位(b0~b3)
- K4…低位的 16 位(b0~b15)
- K8…低位的 32 位(b0~b31)

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

将(D0)中的数据乘以(D2)，并将运算结果存储到(D5，D4)，即(D0) × (D2) → (D5，D4)。

DMUL/32 位乘法运算

DMUL(P)

将(s1)中指定的 BIN32 位与(s2)中指定的 BIN32 位进行乘法运算，并将结果存储到(d)中指定的软元件中。

-[DMUL (s1) (s2) (d)]

内容、范围、数据类型

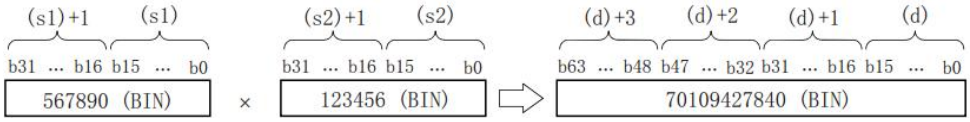
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	乘法运算数据或存储了乘法运算数据的软元件	[-2147483648,2147483647]	有符号BIN32位	ANY32_S	DINT
(s2)	乘法运算数据或存储了乘法运算数据的软元件	[-2147483648,2147483647]	有符号BIN32位	ANY32_S	DINT
(d)	存储运算结果的软元件		有符号BIN64位	ANY64_S	LINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DMUL	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

将(s1)中指定的 BIN32 位数据与(s2)中指定的 BIN32 位数据进行乘法运算，并将运算结果存储到(d)中指定的软元件中。



(d)为位软元件情况下的乘法运算结果

- K1…低位的 4 位(b0~b3)
- K4…低位的 16 位(b0~b15)
- K8…低位的 32 位(b0~b31)

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

将(D1,D0)中的数据乘以(D3，D2)，并将运算结果存储到((D7，D6)，(D5，D4))，即(D1，D0)× (D3，D2) →((D7，D6)，(D5，D4))。

DIV/16 位除法运算

DIV(P)

将(s1)中指定的 BIN16 位数据与(s2)中指定的 BIN16 位数据进行除法运算，并将结果存储到(d)中指定的软元件中。

-[DIV (s1) (s2) (d)]

内容、范围、数据类型

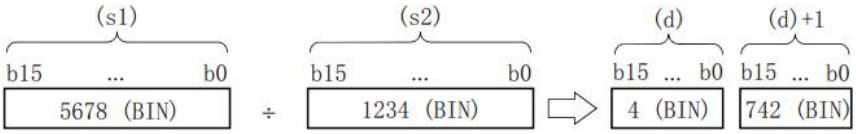
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	除法运算数据或存储了除法运算数据的软元件	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(s2)	除法运算数据或存储了除法运算数据的软元件	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(d)	存储运算结果的软元件		有符号 BIN32 位	ANY32_S	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DIV	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

将(s1)中指定的 BIN16 位数据与(s2)中指定的 BIN16 位数据进行除法运算，并将运算结果存储到(d)中指定的软元件中。



(d)：商 (d)+1：余数

字软元件的情况下，除法运算结果使用 32 位存储商和余数，位软元件的情况下，只使用 16 位存储商。

- 商……被存储到低位 16 位中。
- 余数…被存储到高位 16 位中。(仅可在字软元件的情况下存储。)

注意事项

1、关于运算结果

- 商及余数的最高位表示正(0)负(1)的符号。
- (s1)或(s2)中的一个为负时，商变为负。(s1)为负时，余数变为负。

2、(d)的指定软元件

- 通过位数指定功能，指定位软元件时，无法得出余数。

错误码

错误码	内容
4080H	除数(s2)输入为 0。
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

将(D0)中的数据除以(D2)，并将运算结果：商存储到(D4)，余数存储到(D5)，即(D0)/ (D2) → (D4(商)) (D5(余数))。

DDIV/32 位除法运算

DDIV(P)

将(s1)中指定的 BIN32 位数据与(s2)中指定的 BIN32 位数据进行除法运算，并将结果存储到(d)中指定的软元件中。

-[DDIV (s1) (s2) (d)]

内容、范围、数据类型

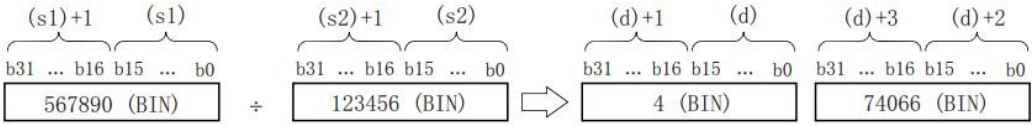
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	除法运算数据或存储了除法运算数据的软元件	[-2147483648, 2147483647]	有符号 BIN32 位	ANY32_S	DINT
(s2)	除法运算数据或存储了除法运算数据的软元件	[-2147483648, 2147483647]	有符号 BIN32 位	ANY32_S	DINT
(d)	存储运算结果的软元件		有符号 BIN64 位	ANY64_S	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DDIV	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

将(s1)中指定的 BIN32 位数据与(s2)中指定的 BIN32 位数据进行除法运算，并将运算结果存储到(d)中指定的软元件中。



字软元件的情况下，除法运算结果使用 BIN64 位存储商及余数。位软元件的情况下，只使用 BIN32 位存储商。

注意事项

1、关于运算结果

- 商及余数的最高位表示正(0)负(1)的符号。
- (s1)或(s2)中的一个为负时，商变为负。(s1)为负时，余数变为负。

2、(d)的指定软元件

- 通过位数指定功能，指定位软元件时，无法得出余数。

错误码

错误码	内容
4080H	除数(s2)输入为 0。
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

将(D1，D0)中的数据除以(D3，D2)，并将运算结果：商存储到(D5，D4)，余数存储到(D7，D6)，即(D1，D0)/(D3，D2) → (D5，D4)(商)(D7，D6)(余数)。

INC/16 位数据递增

INC(P)

对(d)中指定的软元件(BIN16 位数据)进行+1。

-[INC (d)]

内容、范围、数据类型

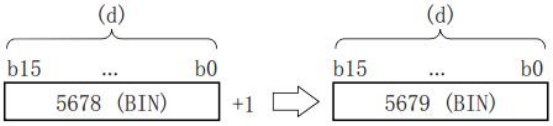
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	保存被加一数据的字软元件编号	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
INC	参数 1												●	●	●	●	●	●	●	●						●	●

功能

对(d)中指定的软元件(BIN16 位数据)进行+1。



- (d)中指定的软元件的内容为 32767 时执行了 INC(P)指令的情况下，-32768 将被存储到(d)中指定的软元件中。
- 标志(零、借位、进位)不执行动作。

注意事项

如果使用连续执行式(INC)指令，则每个运算周期都将进行加法运算，因此应加以注意。

错误码

错误码	内容
4085H	读应用指令中(d)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

对 D0 中指定的软元件数值进行+1，即(D0) + 1 → (D0)。

DINC/32 位数据递增

DINC(P)

对(d)中指定的软元件(BIN32 位数据)进行+1。

-[DINC (d)]

内容、范围、数据类型

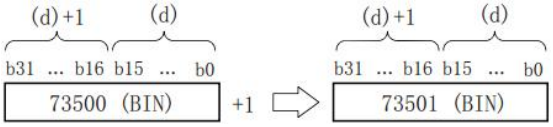
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	保存被加一数据的字软元件编号	[-2147483648, +2147493647]	有符号 BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DINC	参数 1												●	●	●	●	●	●	●	●	●	●				●	●

功能

对(d)中指定的软元件(BIN32 位数据)进行+1。



• (d)中指定的软元件的内容为 2147483647 时执行了 DINC(P)指令的情况下，-2147483648 将被存储到(d)中指定的软元件中。

• 标志(零、借位、进位)不执行动作。

注意事项

如果使用连续执行式(INC)指令，则每个运算周期都将进行加法运算，因此应加以注意。

错误码

错误码	内容
4085H	读应用指令中(d)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

对(D1,D0)中指定的软元件数值进行+1，即(D1,D0) + 1 → (D1,D0)。

DEC/16 位数据递减

DEC(P)

对(d)中指定的软元件(BIN16 位数据)进行-1。

-[DEC (d)]

内容、范围、数据类型

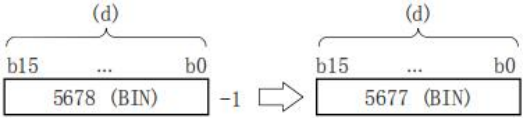
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	保存被减一数据的字软元件编号	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEC	参数 1												●	●	●	●	●	●	●	●						●	●

功能

对(d)中指定的软元件(BIN16 位数据)进行-1。



- (d)中指定的软元件的内容为-32768 时执行了 DEC(P)指令的情况下，32767 将被存储到(d)中指定的软元件中。
- 标志(零、借位、进位)不执行动作。

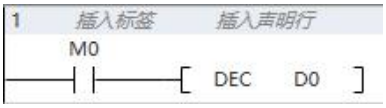
注意事项

如果使用连续执行式(DEC)指令，则每个运算周期都将进行减法运算，因此应加以注意。

错误码

错误码	内容
4085H	读应用指令中(d)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例



M0 每置位一次，对 D0 中指定的软元件数值进行-1，(D0) - 1 → (D0)。

DDEC/32 位数据递减

DDEC(P)

对(d)中指定的软元件(BIN32 位数据)进行-1。

-[DDEC (d)]

内容、范围、数据类型

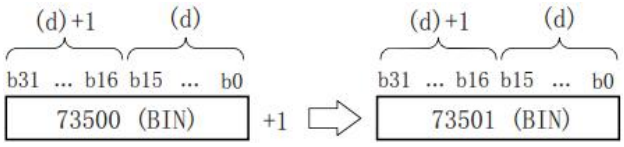
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	保存被减一数据的字软元件编号	[-2147483648, 2147483647]	有符号 BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DDEC	参数 1												●	●	●	●	●	●	●	●	●	●				●	●

功能

对(d)中指定的软元件(BIN32 位数据)进行-1。



- (d)中指定的软元件的内容为 0 时执行了 DDEC(P)指令的情况下，-1 将被存储到(d)中指定的软元件中。
- 标志(零、借位、进位)不执行动作。

注意事项

如果使用连续执行式(DEC)指令，则每个运算周期都将进行减法运算，因此应加以注意。

错误码

错误码	内容
4085H	读应用指令中(d)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

对(D1,D0)中指定的软元件数值进行-1，即(D1,D0) - 1 → (D0)。

7.4 逻辑运算指令

NEG/16 位补码

NEG(P)

对(d)中指定的 BIN16 位软元件的符号进行取反后，存储到(d)中指定的软元件中。

-[NEG (d)]

内容、范围、数据类型

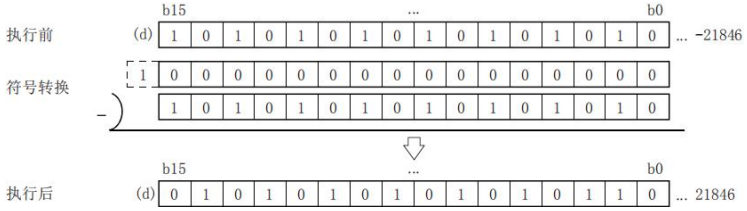
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储进行 2 的补数的数据的起始软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
NEG	参数 1											●	●	●	●	●	●	●	●	●						●	●

功能

- 对(d)中指定的 BIN16 位软元件的符号进行取反后，存储到(d)中指定的软元件中。
- 在对正负符号进行取反时使用。



注意事项

如果使用连续执行式(NEG)指令，则每个运算周期都将被取反，因此应加以注意。

错误码

错误码	内容
4085H	读应用指令中(d)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

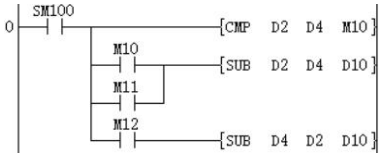
示例

下述两例中假如 D2=K4，D4=K8；或者 D2=K8，D4=K4，D10 的结果均为 K4。

示例 1：M0 每置位一次，对 D0 中指定的软元件数值进行取反。

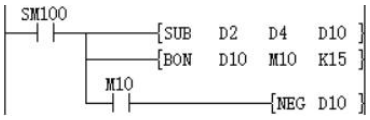
示例 2：（减法运算的差取绝对值）

若 D2>D4 时，M10=On。 若 D2=D4 时，M11=On。若 D2 <D4 时，M12=On。由此可保证 D10 为正值。



也可以用下面的程序来表示：

当 D10 的 bit15 为“1”时（表示 D10 为负数），M10= On，用 NEG 指令将 D10 取补码可得到 D10 的绝对值。



DNEG/32 位补码

DNEG(P)

对(d)中指定的 BIN32 位软元件的符号进行取反后，存储到(d)中指定的软元件中。

-[DNEG　(d)]

内容、范围、数据类型

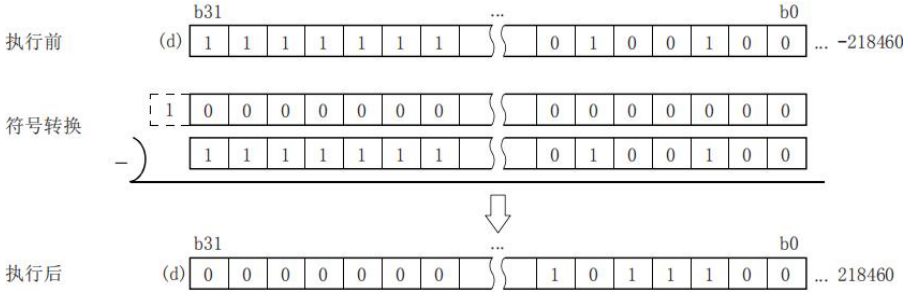
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储进行 2 的补数的数据的起始软元件	[-2147483648, 2147483647]	有符号 BIN16	ANY16_S	DINT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DNEG	参数 1											●	●	●	●	●	●	●	●	●	●	●				●	●

功能

- 对(d)中指定的 BIN32 位软元件的符号进行取反后，存储到(d)中指定的软元件中。
- 在对正负符号进行取反时使用。



注意事项

如果使用连续执行式(DNEG)指令，则每个运算周期都将被取反，因此应加以注意。

错误码

错误码	内容
4085H	读应用指令中(d)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 每置位一次，对(D1，D0)中指定的软元件数值进行取反。

WOR/16 位数据逻辑或

WOR(P)

对(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据的各个位进行逻辑或运算，将结果存储到(d)中指定的软元件中。

-[WOR　(s1)　(s2)　(d)]

内容、范围、数据类型

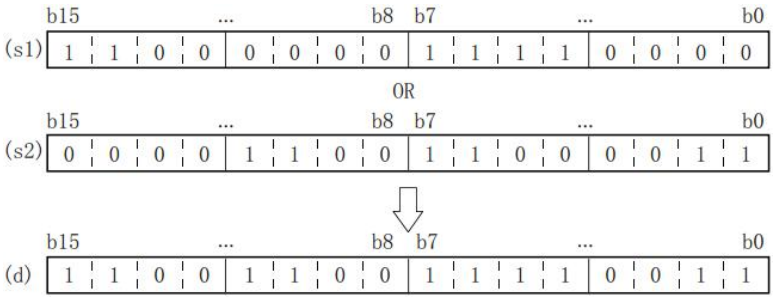
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储进行逻辑或运算的数据或存储了数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(s2)	存储进行逻辑或运算的数据或存储了数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(d)	存储逻辑或结果的软元件		有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
WOR	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

功能

• 对(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据的各个位进行逻辑或运算，将结果存储到(d)中指定的软元件中。



位软元件的情况下，位数指定的点数以后的位软元件将作为 0 进行运算。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将(D0)与(D2)进行逻辑或运算后，将值存入（D4），即(D0)∨(D2) → (D4)

DOR/32 位数据逻辑或

DOR(P)

对(d)中指定的 BIN32 位软元件的符号进行取反后，存储到(d)中指定的软元件中。

-[DOR （s1） （s2） (d)]

内容、范围、数据类型

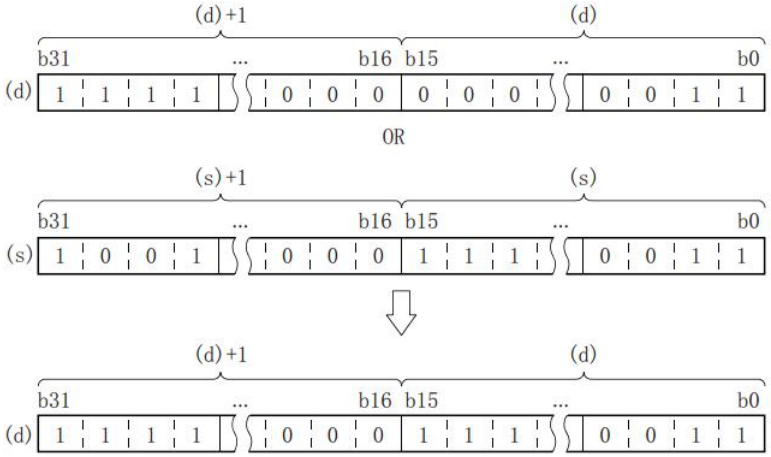
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储进行逻辑或运算的数据或存储了数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(s2)	存储进行逻辑或运算的数据或存储了数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	存储逻辑或结果的软元件		有符号 BIN32	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DOR	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

对(s1)中指定的软元件的 BIN32 位数据与(s2)中指定的软元件的 BIN32 位数据的各个位进行逻辑或运算，将结果存储到(d)中指定的软元件中。



位软元件的情况下，位数指定的点数以后的位软元件将作为 0 进行运算。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将(D1，D0)与(D3，D2)进行逻辑或运算后，将值存入（D5，D4），即(D1，D0)∨(D3，D2) → (D5，D4)。

WAND/16 位数据逻辑与

WAND(P)

对(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据的各个位进行逻辑与运算，将结果存储到(d)中指定的软元件中。

-[WAND （s1） （s2） (d)]

内容、范围、数据类型

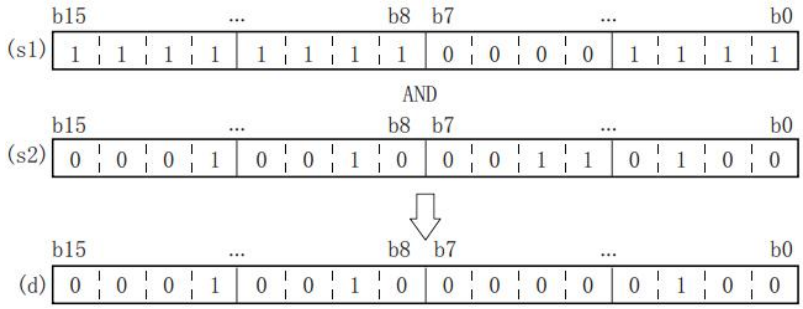
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储进行逻辑与运算的数据或存储了数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(s2)	存储进行逻辑与运算的数据或存储了数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(d)	存储逻辑与结果的软元件		有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
WAND	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

功能

对(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据的各个位进行逻辑与运算，将结果存储到(d)中指定的软元件中。



位软元件的情况下，位数指定的点数以后的位软元件将作为 0 进行运算。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将(D0)与(D2)进行逻辑与运算后，将值存入（D4），即(D0)∧(D2) → (D4)。

DAND/32 位数据逻辑与

DAND(P)

对(s1)中指定的软元件的 BIN32 位数据与(s2)中指定的软元件的 BIN32 位数据的各个位进行逻辑与运算，将结果存储到(d)中指定的软元件中。

-[DAND （s1） （s2） (d)]

内容、范围、数据类型

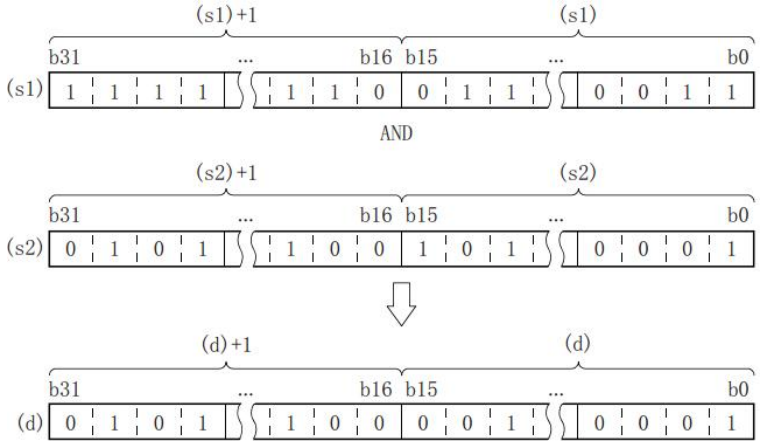
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储进行逻辑与运算的数据或存储了数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(s2)	存储进行逻辑与运算的数据或存储了数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	存储逻辑与结果的软元件		有符号 BIN32	ANY32_S	DINT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DAND	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

对(s1)中指定的软元件的 BIN32 位数据与(s2)中指定的软元件的 BIN32 位数据的各个位进行逻辑与运算，将结果存储到(d)中指定的软元件中。



位软元件的情况下，位数指定的点数以后的位软元件将作为 0 进行运算。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将(D1，D0)与(D3，D2)进行逻辑与运算后，将值存入（D5，D4），(D1，D0)∧(D3，D2) → (D5，D4)。

WXOR/16 位数据逻辑异或

WXOR(P)

对(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据的各个位进行异或运算，将结果存储到(d)中指定的软元件中。

-[WXOR （s1） （s2） (d)]

内容、范围、数据类型

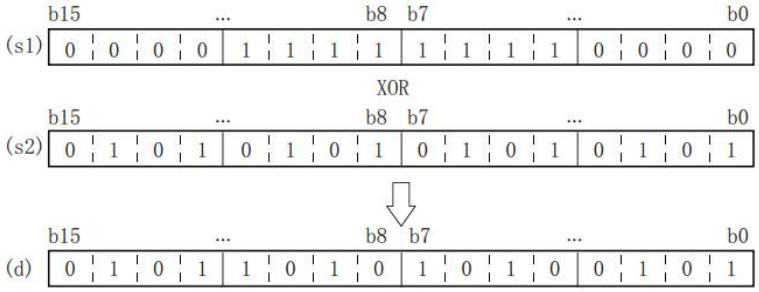
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储进行异或运算的数据或存储了数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(s2)	存储进行异或运算的数据或存储了数据的软元件	[-32768,+32767]	有符号 BIN16	ANY16_S	INT
(d)	存储异或结果的软元件		有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
WXOR	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

功能

• 对(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据的各个位进行逻辑异或运算，将结果存储到(d)中指定的软元件中。



位软元件的情况下，位数指定的点数以后的位软元件将作为 0 进行运算。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

示例 1：

M0 置位时，将(D0)与(D2)进行异或运算后，将值存入（D4），(D0)∀(D2) →(D4)。

示例 2：

与 CML 指令一起使用时实现逻辑异或否（XORNOT）运算。

福州富昌维控电子科技有限公司

DXOR/32 位数据逻辑异或

DXOR(P)

对(s1)中指定的软元件的 BIN32 位数据与(s2)中指定的软元件的 BIN32 位数据的各个位进行异或运算，将结果存储到(d)中指定的软元件中。

-[DXOR （s1） （s2） (d)]

内容、范围、数据类型

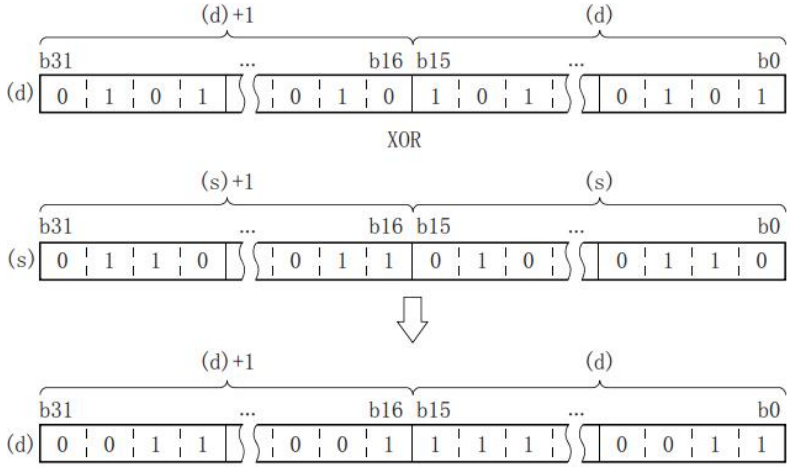
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储进行异或运算的数据或存储了数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(s2)	存储进行异或运算的数据或存储了数据的软元件	[-2147483648, 2147483647]	有符号 BIN32	ANY32_S	DINT
(d)	存储异或结果的软元件		有符号 BIN32	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DXOR	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

对(s1)中指定的软元件的 BIN32 位数据与(s2)中指定的软元件的 BIN32 位数据的各个位进行异或运算，将结果存储到(d)中指定的软元件中。



位软元件的情况下，位数指定的点数以后的位软元件将作为 0 进行运算。

错误码

错误码	内容
4085H	读应用指令中(s1)(s2)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将(D1，D0)与(D3，D2)进行异或运算后，将值存入（D5，D4），即(D1，D0) ∀ (D3，D2) → (D5，D4)。

ANS/报警器设置

ANS

用于设置报警器指令。

-[ANS （s） （n） (d)]

内容、范围、数据类型

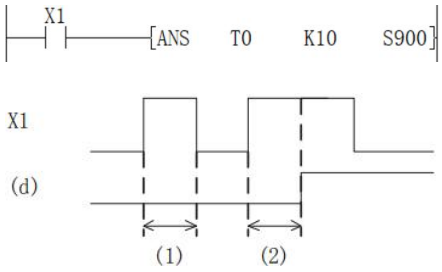
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	判断时间的计时定时器编号	-	有符号 BIN16 位	ANY16	--
(n)	判断时间的数据	[1,32767]	有符号 BIN16 位	ANY16	INT
(d)	设置的报警器软元件	-	位	ANY16_BOOL	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ANS	参数 1															●										●	
	参数 2											●	●	●	●	●	●	●	●				●	●		●	
	参数 3				●																					●	

功能

• 指令输入在判断时间[（n）×100ms，定时器（s）]以上持续 ON 时，设置（d）。指令时间在判断时间[（n）×100ms]以下变为 OFF 的情况下，判断用定时器（s）的当前值复位，不设置（d）。此外，如果指令输入变为 OFF，判断用定时器将复位。



（1）：判断时间以下（（n）X 100ms 以下）

（2）：判断时间以上（含）（（n）X 100ms 以上（含））

相关软元件

软元件	名称	内容
SM249	信号报警器有效	SM249 置 ON 后，下面的 SM248 和 SD249 工作。
SM248	信号报警器动作	SM249 为 ON，状态 S900~S999 中任一动作的时候，SM248 置 ON。
SD249	信号报警器 ON 状态最小编号	保存 S900~S999 中动作的最小编号。

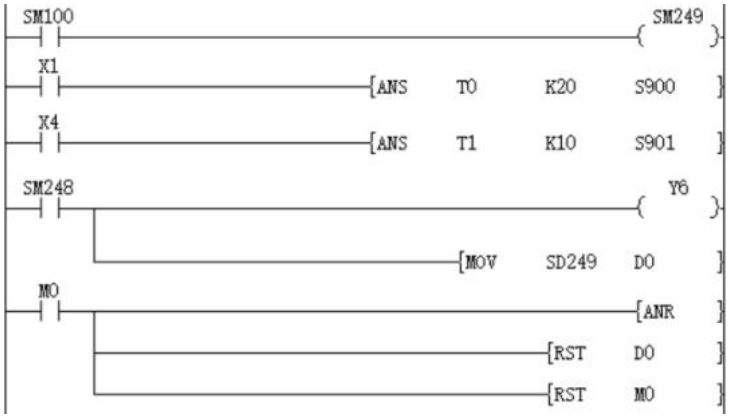
错误码

错误码	内容
4084H	应用指令中(n)输入了超出可指定范围的数据。范围：1~32767。
	定时器编号不在 T0~T199 范围内。
	信号报警器不在 S900~S999 范围内。
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

通过信号报警器显示故障编号。

如下所示，编写诊断外部故障用的程序，如监控 SM249（ON 状态最小编号）的内容时，会显示 S900~S999 中为 ON 的状态的最小编号。同时发生多个故障时，排除了最小编号的故障后可以得知下一个故障编号。



SM249 置 ON 后监控有效

检测到 X1 持续 2 秒置 ON，置 S900

检测到 X4 持续 1 秒置 ON，置 S901

S900~S999 中任意一个为 ON 后 SM248 动作，输出故障显示 YY6 动作

显示故障号至 D0 软元件中

通过外部故障诊断程序，使用复位按钮 M0 使已动作的状态 OFF。M0 每次置 ON 时，新编号的动作状态依次被置位，已经 ON 的新编号复位。

ANR/报警器复位

ANR(P)

将报警器中为 ON 的小编号复位的指令。

-[ANR]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)
无	无参数设定	——	——	——

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ANR	无	无对象软元件																								

功能

• 如果指令输入为 ON，将报警器中动作中的报警器复位。

如果多个报警器正在动作，将小编号的复位。如果将输入指令再次置为 ON，将正在动作的报警器中下一个小编号的复位。

相关软元件

软元件	名称	内容
SM249	信号报警器有效	SM249 置 ON 后，下面的 SM248 和 SD249 工作。
SM248	信号报警器动作	SM249 为 ON，状态 S900~S999 中任一动作的时候，SM248 置 ON。
SD249	信号报警器 ON 状态最小编号	保存 S900~S999 中动作的最小编号。

注意事项

• 如果使用 ANR 指令，每个周期按顺序复位。

• 如果使用 ANRP 指令，仅 1 个运算周期执行。

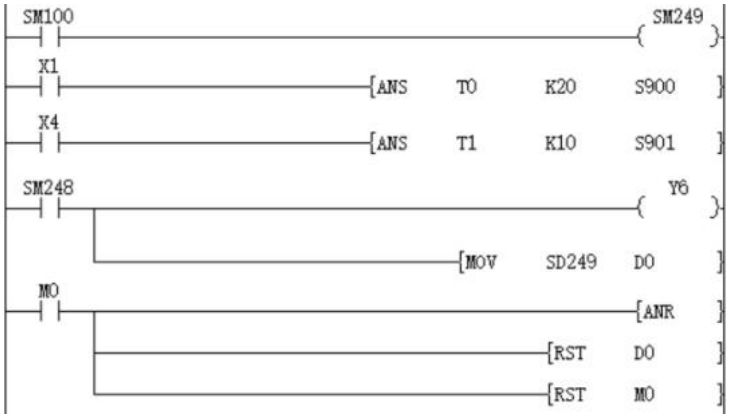
错误码

无运算错误。

示例

通过信号报警器显示故障编号。

如下所示，编写诊断外部故障用的程序，如监控 SM249（ON 状态最小编号）的内容时，会显示 S900~S999 中为 ON 的状态的最小编号。同时发生多个故障时，排除了最小编号的故障后可以得知下一个故障编号。



SM249 置 ON 后监控有效

检测到 X1 持续 2 秒置 ON，置 S900

检测到 X4 持续 1 秒置 ON，置 S901

S900~S999 中任意一个为 ON 后 SM248 动作，输出故障显示 YY6 动作

显示故障号至 D0 软元件中

通过外部故障诊断程序，使用复位按钮 M0 使已动作的状态 OFF。M0 每次置 ON 时，新编号的动作状态依次被置位，已经 ON 的新编号复位。

BON/16 位数据位判断

BON(P)

检查（s）中指定的软元件的 BIN16 位数据（n）位的状态为 ON 还是 OFF，将结果输出到（d）中指定的软元件中。

-[BON （s） （d） (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	数据存储目标字软元件编号	-	有符号 BIN16 位	ANY16	INT
(d)	驱动的位软元件编号	-	位	ANY16_BOOL	INT
(n)	要判断的位的位置	[0,15]	有符号 BIN16 位	ANY16	INT

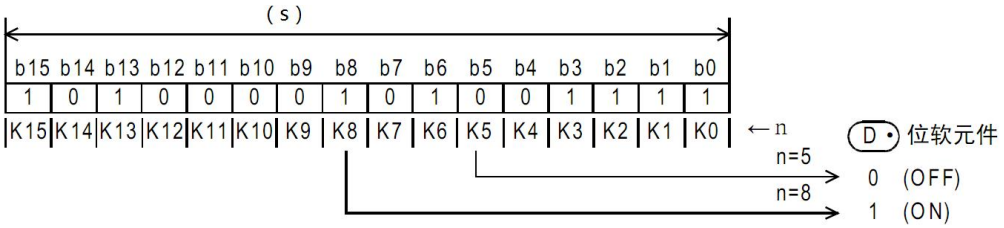
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BON	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2		●	●	●	●					●							●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

检查（s）中指定的软元件的 BIN16 位数据（n）位的状态为 ON 还是 OFF，将结果输出到（d）中指定的软元件中。

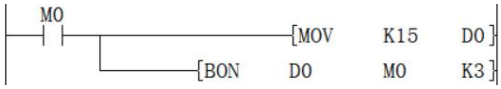
- 上述结果如果为 ON，则（d）=ON，如果为 OFF，则（d）=OFF。
- 如果（s）中指定的软元件中指定常数（K），将自动转换为 BIN。



错误码

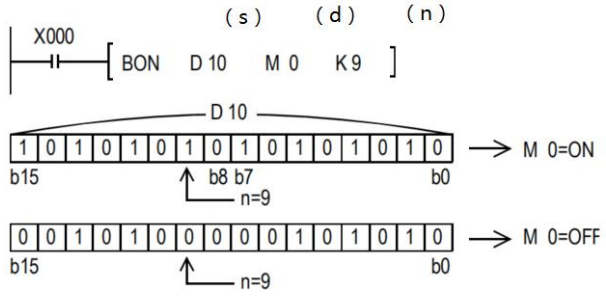
错误码	内容
4084H	(n)中输入了超出指定范围的数据。[0,15]
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



D0 中的 n=第 3 位为 1（ON）时，M0 置 1（ON）。

D10中的n=第9位为1(ON)时，M0为1(ON)。



DBON/32 位数据位判断

DBON(P)

检查（s）中指定的软元件的 BIN32 位数据（n）位的状态为 ON 还是 OFF，将结果输出到（d）中指定的软元件中。

-[DBON （s） （d） (n)]

内容、范围、数据类型

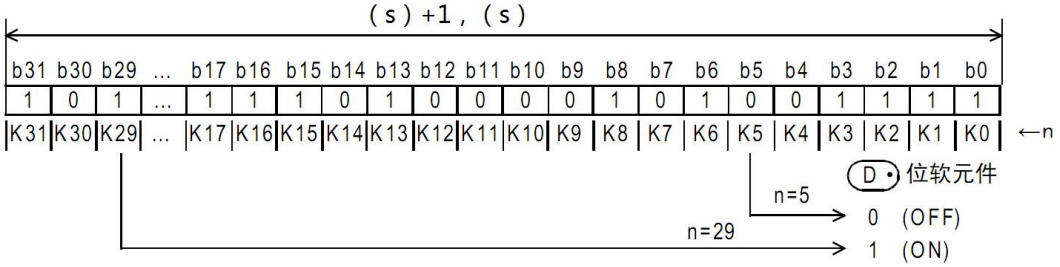
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	数据存储目标字软元件编号	-	有符号 BIN32 位	ANY32	DINT
(d)	驱动的位软元件编号	-	位	ANY32_BOOL	DINT
(n)	要判断的位的位置	[0,31]	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBON	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2		●	●	●	●					●							●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●

功能

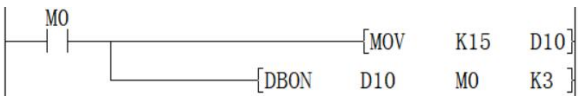
- 检查（s）中指定的软元件的 BIN32 位数据（n）位的状态为 ON 还是 OFF，将结果输出到（d）中指定的软元件中。
- 上述结果如果为 ON，则（d）=ON，如果为 OFF，则（d）=OFF。
- 如果（s）中指定的软元件中指定常数（K），将自动转换为 BIN。



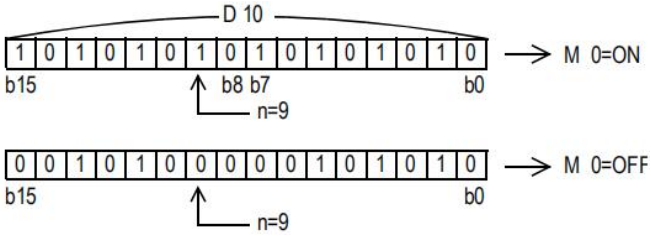
错误码

错误码	内容
4084H	(n)中输入了超出指定范围的数据。数据范围：[0,31]。
4085H	读应用指令(s)、(n)中指定的软 A 元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



D0 中的 n=第 3 位为 1（ON）时，M0 置 1（ON）。



ENCO/编码

ENCO(P)

对（s）开始的 2 的（n）次方位的数据进行编码，并存储到（d）中。

-[ENCO （s） （d） (n)]

内容、范围、数据类型

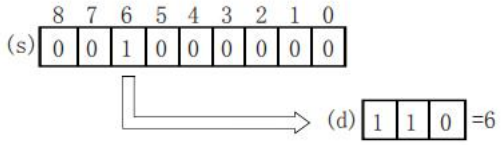
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储编码数据的起始软元件	-	位/有符号 BIN16 位	ANY_ELEMENTARY	INT
(d)	存储编码结果的软元件编号	-	有符号 BIN16 位	ANY_ELEMENTARY	INT
(n)	有效位长	[0,8]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ENCO	参数 1	●	●	●	●	●										●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

- 将从（s）的 $2^{(n)}$ 位的数据变为 1 的位所对应的 BIN 值存储到（d）中。

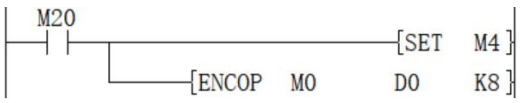


- （n）=0 时将变为无处理，（d）中指定的软元件的内容不变化。
- 位软元件作为 1 位处理，字软元件作为 16 位处理。
- 多个位为 1 时以高位的位置进行处理。

错误码

错误码	内容
4084H	在(s)的位软元件指定中，(n)为 0~8 以外时。
	在(s)的字软元件指定中，(n)为 0~4 以外时。
	从(s)开始的 2（n）位的数据全部为 0 时。
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 M20 置 ON 时，编码后 D0 软元件为 20。

福州富昌维控电子科技有限公司

DECO/解码

DECO(P)

对（s）中指定的软元件的低位（n）位进行解码，将结果存储到（d）中指定的软元件的 2 的（n）次方位中。

-[DECO （s） （d） (n)]

内容、范围、数据类型

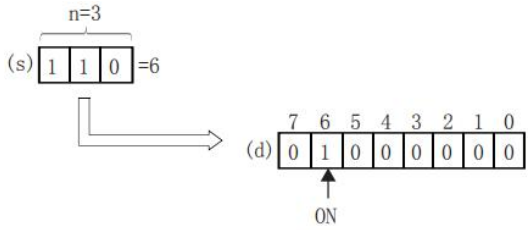
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	解码数据或存储了解码数据的软元件编号	-	位/有符号 BIN16 位	ANY_ELEMENTARY	INT
(d)	存储解码结果的起始软元件	-	有符号 BIN16 位	ANY_ELEMENTARY	INT
(n)	有效位长	[0,8]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DECO	参数 1	●	●	●	●	●							●	●	●	●	●	●	●	●			●	●		●	●
	参数 2		●	●	●	●										●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

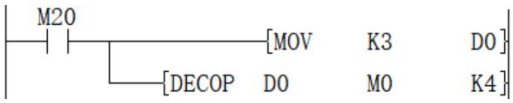
- 将（s）的低位（n）位中指定的 BIN 值对应的（d）的位置为 ON。
- （n）=0 时将变为无处理，（d）中指定的软元件的内容不变化。
- 位软元件作为 1 位处理，字软元件作为 16 位处理。



错误码

错误码	内容
4084H	在(d)的位软元件指定中，(n)为 0~8 以外时。
	在(d)的字软元件指定中，(n)为 0~4 以外时。
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M20 为 ON 时，M3 将置为 ON。

SUM/16 位数据的 ON 位数

SUM(P)

将（s）中指定的软元件的 BIN16 位数据中处于 1 的位的总数存储到（d）中指定的软元件中。

-[SUM （s） (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	对处于 1 的位的总数进行计数的软元件起始编号	-	有符号 BIN16 位	ANY16	INT
(d)	存储位的总数的软元件起始编号	-	有符号 BIN16 位	ANY16	INT

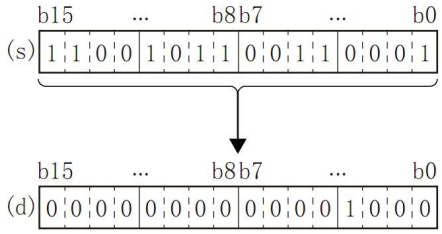
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SUM	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

将（s）中指定的软元件的 BIN16 位数据中处于 1 的位的总数存储到（d）中指定的软元件中。

- （s）中指定的软元件的 BIN16 位数据全部为 0 时，零标志（SM153）变为 ON。

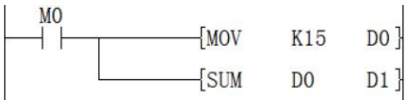


1(ON)的总数以 BIN 存储，左侧的示例中为 8 个。

错误码

错误码	内容
4085H	读应用指令(s)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M0 为 ON 时，对 D0 中 ON 的位数计数后保存到 D1 中，D1 执行后的值为 4。

DSUM/32 位数据的 ON 位数

DSUM(P)

将（s）中指定的软元件的 BIN32 位数据中处于 1 的位的总数存储到（d）中指定的软元件中。

-[DSUM （s） (d)]

内容、范围、数据类型

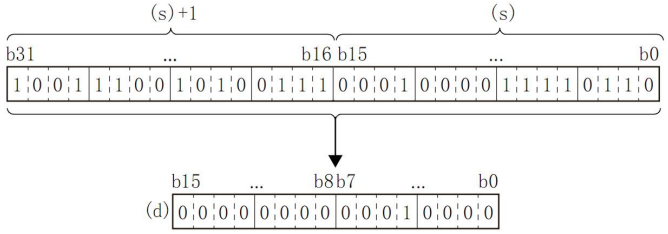
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	对处于 1 的位的总数进行计数的软元件起始编号	-	有符号 BIN32 位	ANY32	DINT
(d)	存储位的总数的软元件起始编号	-	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSUM	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●

功能

- 将（s）中指定的软元件的 BIN32 位数据中，处于 1 的位的总数存储到（d）中指定的软元件中。
- （s）中指定的软元件的 BIN32 位数据全部为 0（OFF）时，零标志（SM153）变为 ON。



1（ON）的总数以 BIN 存储，左述的示例中为 16 个。

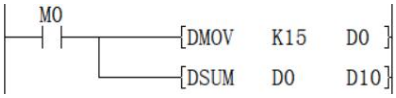
注意事项

指令输入为 OFF 时不执行指令，动作的 ON 位数的输出保持之前的状态不变。

错误码

错误码	内容
4085H	读应用指令(s)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M0 为 ON 时，对 D0 中 ON 的位数计数后保存到 D10 中，D10 执行后的值为 4。

MEAN/16 位数据的平均值

MEAN(P)

对（s）中指定的软元件开始的（n）点（16 位数据单位）的平均值进行计算，将计算结果存储到（d）中。

-[MEAN （s） (d)（n）]

内容、范围、数据类型

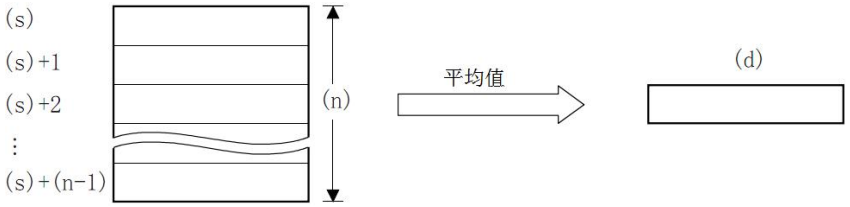
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了进行平均值计算的数据的软元件起始编号	-	有符号 BIN16 位	ANY16	--
(d)	存储平均值的软元件起始编号	-	有符号 BIN16 位	ANY16	INT
(n)	数据数或存储了数据数的软元件编号	[1,32767]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MEAN	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

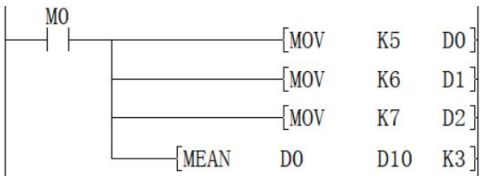
- 对（s）中指定的软元件开始的（n）点的 16 位数据的平均值进行计算后，存储到（d）中指定的软元件中。
- 合计由代数和求出，除以（n）。
- 余数被舍去。



错误码

错误码	内容
4084H	应用指令中(n)输入了超出可指定范围的数据。N≤0
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



将 D0、D1、D2 的数据相加，除以 3 后求得的值保存到 D10 中，其计算后的平均值为 6。

DMEAN/32 位数据的平均值

DMEAN(P)

对（s）中指定的软元件开始的（n）点（32 位数据单位）的平均值进行计算，将计算结果存储到（d）中。

-[DMEAN （s） (d)（n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了进行平均值计算的数据的软元件起始编号	-	有符号 BIN32 位	ANY32	--
(d)	存储平均值的软元件起始编号	-	有符号 BIN32 位	ANY32	DINT
(n)	数据数或存储了数据数的软元件编号	[1,2147483647]	有符号 BIN32 位	ANY32	DINT

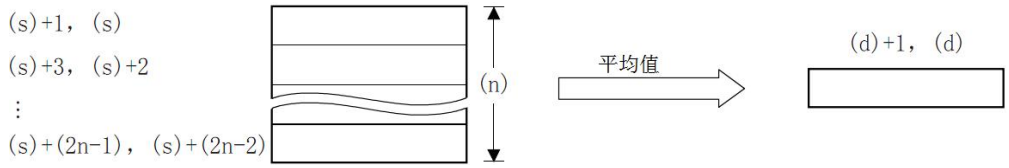
使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DMEAN	参数 1											●	●	●	●	●	●	●	●	●	●	●				●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●
	参数 3											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●

功能

对（s）中指定的软元件开始的（n）点的 BIN32 位数据的平均值进行计算后，存储到（d）中指定的软元件中。

- 合计由代数和求出，除以（n）。
- 余数被舍去。



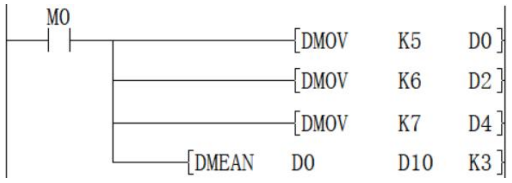
注意事项

软元件编号超出时，在允许范围内将（n）作为较小值处理。

错误码

错误码	内容
4084H	(n)中输入了超出指定范围的数据。N≤0
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



将 D0、D2、D4 的数据相加，除以 3 后求得的值保存到 D10、D11 中，其计算后的平均值为 6。

SQR/16 位平方根

SQR(P)

计算（s）中指定的 BIN16 位数据的平方根，将计算结果存储到（d）中。

-[SQR （s） (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了进行平方根计算的数据软元件	[0,32767]	有符号 BIN16 位	ANY16	INT
(d)	存储算出的平方根的软元件	-	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SQR	参数 1																	●	●	●			●	●		●	●
	参数 2																	●	●	●						●	●

功能

- 计算（s）中指定的 BIN16 位数据的平方根，将计算结果存储到（d）中。

$$\sqrt{(s)} \rightarrow (d)$$

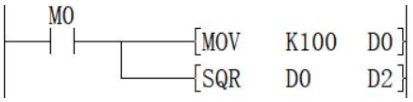
注意事项

- 运算结果小数点将被舍去，变为整数。如果发生舍去，SM152（借位标志）变为 ON。
- 运算结果真为 0 时，SM153(零位标志)变为 ON。

错误码

错误码	内容
4084H	(s)中指定了负值时。
4085H	读应用指令(s)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



D0 的平方根保存到 D2 中，D0 的值为 100，故 D2 软元件中的值为 10。

DSQR/32 位平方根

DSQR(P)

计算（s）中指定的 BIN32 位数据的平方根，将计算结果存储到（d）中。

-[DSQR （s） (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了进行平方根计算的数据软元件	[0,2147483647]	有符号 BIN32 位	ANY32	DINT
(d)	存储算出的平方根的软元件	-	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSQR	参数 1																	●	●	●	●	●	●	●		●	●
	参数 2																	●	●	●	●	●				●	●

功能

- 计算（s）中指定的 BIN32 位数据的平方根，将计算结果存储到（d）中。

$$\sqrt{(s)+1,\ (s)} \rightarrow (d)+1,\ (d)$$

注意事项

- 运算结果小数点将被舍去，变为整数。如果发生舍去，SM152（借位标志）变为 ON。
- 运算结果真为 0 时，SM153(零位标志)变为 ON。

错误码

错误码	内容
4084H	(s)中指定了负值时。
4085H	读应用指令(s)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

D0 的平方根保存到 D2 中，D0 的值为 110，故 D2 软元件中的值为 10（舍去小数部分），借位标志 SM152 置 ON。

WSUM/16 位数据合计值

WSUM(P)

将（s）中指定的软元件开始的（n）点的 BIN16 位数据全部进行加法运算后，存储到（d）中指定的软元件中。

-[WSUM （s） (d) （n）]

内容、范围、数据类型

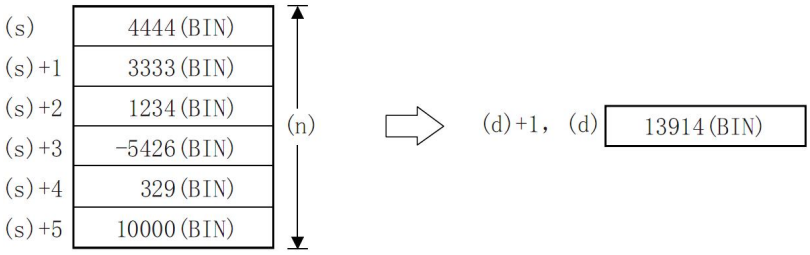
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了进行合计值计算的数据的软元件起始编号	-	有符号 BIN16 位	ANY16	--
(d)	存储合计值的软元件起始编号	-	有符号 BIN32 位	ANY32	--
(n)	数据个数	-	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
WSUM	参数 1															●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

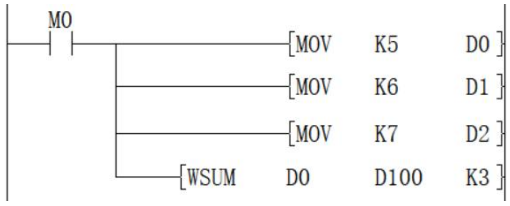
将（s）中指定的软元件开始的（n）点的 BIN16 位数据全部进行加法运算后，存储到（d）中指定的软元件中。



错误码

错误码	内容
4084H	(n)中指定的数值，(n)≤0 时。
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 M0=ON 时，将 D0~D2 的 16 位数据合计保存到[D100,D101]中，其会计的结果为 18。

DWSUM/32 位数据合计值

DWSUM(P)

将（s）中指定的软元件开始的（n）点的 BIN32 位数据全部进行加法运算后，存储到（d）中指定的软元件中。

-[DWSUM （s） (d) （n）]

内容、范围、数据类型

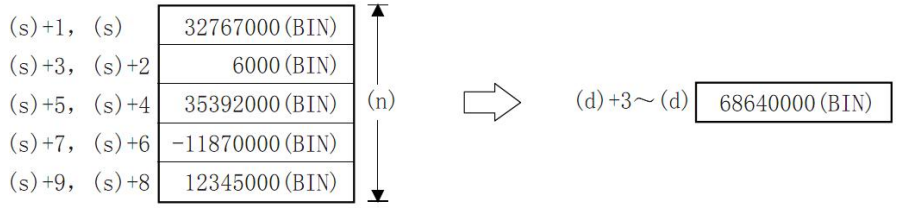
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了进行合计值计算的数据的软元件起始编号	-	有符号 BIN32 位	ANY32	--
(d)	存储合计值的软元件起始编号	-	有符号 BIN64 位	ANY64	--
(n)	数据个数	-	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DWSUM	参数 1															●	●	●	●	●	●	●				●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●
	参数 3											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●

功能

将（s）中指定的软元件开始的（n）点的 BIN32 位数据全部进行加法运算后，存储到（d）中指定的软元件中。



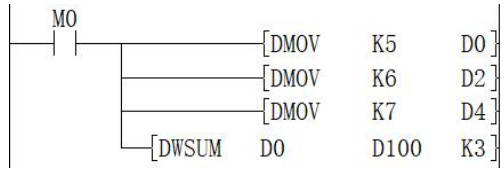
注意事项

在（d）中指定位数指定软元件的场合，n 取值范围 1~8，如 K8(32 位指令，例如 K8M0)而无 K16(64 位指令)。

错误码

错误码	内容
4084H	(n)中指定的数值，(n)≤0 时。
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 M0=ON 时，将[D0,D1]、[D2,D3]、[D4,D5]的 32 位数据合计保存到[D100,D103]的 64 位软元件空间中，其合计的结果为 18。

SORT/16 位数据排序

SORT

将（s）中指定的（n1×n2）点的 BIN16 位数据表（排序源），以（n3）列的组数据为基准，对数据行进行升序排序，并存储到（d）中指定的（n1×n2）点的 BIN16 位数据表（排序后）中。

-[SORT (s) （n1） （n2） （d） （n3）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了数据表的软元件起始编号	-	有符号 BIN16 位	ANY16	--
(n1)	数据（行）数	[1,32]	有符号 BIN16 位	ANY16	INT
(n2)	组数据（列）数	[1,6]	有符号 BIN16 位	ANY16	INT
(d)	存储运算结果的软元件起始编号	-	有符号 BIN16 位	ANY16	--
(n3)	作为排序基准的组数据（列）的行编号	-	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SORT	参数 1															●	●	●	●	●						●	
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 4															●	●	●	●	●						●	
	参数 5											●	●	●	●	●	●	●	●	●			●	●		●	

功能

• 将（s）中指定的（n1×n2）点的 BIN16 位数据表（排序源），以（n3）列的组数据为基准，对数据行进行升序排序，并存储到（d）中指定的（n1×n2）点的 BIN16 位数据表（排序后）中。

• 以排序源中（n1）=K3、（n2）=K4 的情况为例，数据表构成如下所示。当为排序后数据表的情况下，应将（s）替换为（d）。

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=3 的情况下	行号 1	（s）	（s）+3	（s）+6	（s）+9
	行号 2	（s）+1	（s）+4	（s）+7	（s）+10
	行号 3	（s）+2	（s）+5	（s）+8	（s）+11

• 指令输入为 ON 时开始数据对齐，（n1）扫描后数据对齐结束，指令执行结束标志 SM229 置为 ON。

• 根据下述排序源数据，动作示例如下所示。此外，在第 1 列中预先放入管理编号等序号，就可根据该内容判断原来的行号，非常方便。

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（s）	（s）+5	（s）+10	（s）+15
		1	150	45	20
	行号 2	（s）+1	（s）+6	（s）+11	（s）+16
		2	180	50	40
	行号 3	（s）+2	（s）+7	（s）+12	（s）+17
		3	160	70	30

福州富昌维控电子科技有限公司

	行号 4	（s）+3	（s）+8	（s）+13	（s）+18
		4	100	20	8
	行号 5	（s）+4	（s）+9	（s）+14	（s）+19
		5	150	50	45

• 按（n3）=K2(列号 2)执行了指令时的排序结果。

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（d）	（d）+5	（d）+10	（d）+15
		4	100	20	8
	行号 2	（d）+1	（d）+6	（d）+11	（d）+16
		1	150	45	20
	行号 3	（d）+2	（d）+7	（d）+12	（d）+17
		5	150	50	45
	行号 4	（d）+3	（d）+8	（d）+13	（d）+18
		3	160	70	30
	行号 5	（d）+4	（d）+9	（d）+14	（d）+19
		2	180	50	40

• 按（n3）=K3(列号 3)执行了指令时的排序结果。

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（d）	（d）+5	（d）+10	（d）+15
		4	100	20	8
	行号 2	（d）+1	（d）+6	（d）+11	（d）+16
		1	150	45	20
	行号 3	（d）+2	（d）+7	（d）+12	（d）+17
		5	150	50	45
	行号 4	（d）+3	（d）+8	（d）+13	（d）+18
		2	180	50	40
	行号 5	（d）+4	（d）+9	（d）+14	（d）+19
		4	100	70	8

注意事项

• SORT 指令只支持升序排列。

• 动作中请勿改变操作数及数据内容。

• 再次执行时，应将指令输入置为 OFF 一次。

• SORT 指令在程序中最多可驱动 1 条。

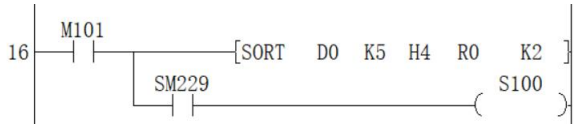
• （s）和（d）中指定同一软元件的情况下，源数据改写为排序后的数据顺序。请特别注意在执行结束前不要改变（s）内容。

错误码

错误码	内容
4084H	(n1)中指定的数值超出下述范围时。[1,32]
	(n2)中指定的数值超出下述范围时。[1,6]
	(n3)中指定的数值超出下述范围时。1~n2
4085H	读应用指令(s)、(n1)、(n2)、(n3)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。
4087H	应用指令中(d)参数使用了不支持的软元件。
4089H	应用指令超出限制使用条数。

示例

参考功能说明示例。



SORT2/16 位数据排序

SORT2(P)

将（s）中指定的（n1×n2）点的 BIN16 位数据表（排序源），以（n3）列的组数据为基准，对数据行进行升序或降序排序，并存储到（d）中指定的（n1×n2）点的 BIN16 位数据表（排序后）中。

-[SORT2 (s) （n1）（n2）（d）（n3）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了数据表的软元件起始编号	-	有符号 BIN16 位	ANY16	--
(n1)	数据（行）数	[1,32]	有符号 BIN16 位	ANY16	INT
(n2)	组数据（列）数	[1,6]	有符号 BIN16 位	ANY16	INT
(d)	存储运算结果的软元件起始编号	-	有符号 BIN16 位	ANY16	--
(n3)	作为排序基准的组数据（行）的列编号	-	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SORT2	参数 1															●	●	●	●	●						●	
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 4															●	●	●	●	●						●	
	参数 5											●	●	●	●	●	●	●	●	●			●	●		●	

功能

• 将（s）中指定的（n1×n2）点的 BIN16 位数据表（排序源），以（n3）列的组数据为基准，对数据行进行升序或降序排序，并存储到（d）中指定的（n1×n2）点的 BIN16 位数据表（排序后）中。

• 以排序源中（n1）=K3、（n2）=K4 的情况为例，数据表构成如下所示。当为排序后数据表的情况下，应将（s）替换为（d）。

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=3 的情况下	行号 1	（s）	（s）+1	（s）+2	（s）+3
	行号 2	（s）+4	（s）+5	（s）+6	（s）+7
	行号 3	（s）+8	（s）+9	（s）+10	（s）+11

• 排序通过 SM165 的 ON/OFF 状态设置

	排序的顺序设置说明
SM165=ON	降序
SM165=OFF	升序

• 指令输入为 ON 时开始数据对齐，（n1）扫描后数据对齐结束，指令执行结束标志 SM229 置为 ON。

• 根据下述排序源数据，动作示例如下所示。此外，在第 1 列中预先放入管理编号等序号，就可根据该内容判断原来的行号，非常方便。

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（s）	（s）+1	（s）+2	（s）+3
		1	150	45	20

福州富昌维控电子科技有限公司

	行号 2	（s）+4	（s）+5	（s）+6	（s）+7
		2	180	50	40
	行号 3	（s）+8	（s）+9	（s）+10	（s）+11
		3	160	70	30
	行号 4	（s）+12	（s）+13	（s）+14	（s）+15
		4	100	20	8
	行号 5	（s）+16	（s）+17	（s）+18	（s）+19
		5	150	50	45

• 按（n3）=K2(列号 2)执行了指令时的排序结果（升序的情况下 SM165=OFF）

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（d）	（d）+1	（d）+2	（d）+3
		4	100	20	8
	行号 2	（d）+4	（d）+5	（d）+6	（d）+7
		1	150	45	20
	行号 3	（d）+8	（d）+9	（d）+10	（d）+11
		5	150	50	45
	行号 4	（d）+12	（d）+13	（d）+14	（d）+15
		3	160	70	30
	行号 5	（d）+16	（d）+17	（d）+18	（d）+19
		2	180	50	40

• 按（n3）=K3(列号 3)执行了指令时的排序结果（降序的情况下 SM165=ON）

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（d）	（d）+1	（d）+2	（d）+3
		3	160	70	30
	行号 2	（d）+4	（d）+5	（d）+6	（d）+7
		2	180	50	40
	行号 3	（d）+8	（d）+9	（d）+10	（d）+11
		5	150	50	45
	行号 4	（d）+12	（d）+13	（d）+14	（d）+15
		1	150	45	20
	行号 5	（d）+16	（d）+17	（d）+18	（d）+19
		4	100	20	8

注意事项

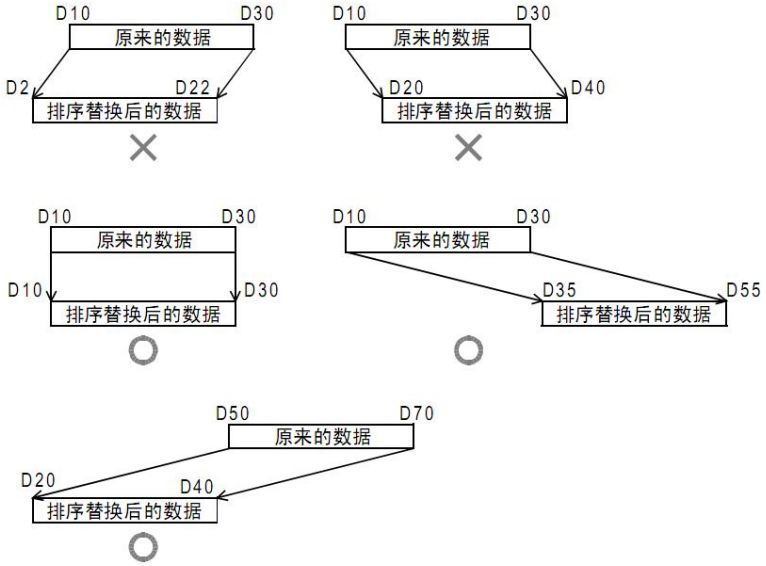
• 动作中请勿改变操作数及数据内容。

• 再次执行时，应将指令输入置为 OFF 一次。

• SORT2 指令在程序中最多只能写入同时驱动 2 条。

• （s）和（d）中指定同一软元件的情况下，源数据改写为排序后的数据顺序。请特别注意在执行结束前不要改变（s）内容。

• 请勿使源数据和排序后的数据错位重叠。



错误码

错误码	内容
4084H	(n1)中指定的数值超出下述范围时。[1,32]
	(n2)中指定的数值超出下述范围时。[1,6]
	(n3)中指定的数值超出下述范围时。1~n2
4085H	读应用指令(s)、(d)、(n1)、(n2)、(n3)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。
4089H	应用指令超出限制使用条数。

示例

参考功能说明示例。

DSORT2/32 位数据排序

DSORT2(P)

将（s）中指定的（n1×n2）点的 BIN32 位数据表（排序源），以（n3）列的组数据为基准，对数据行进行升序或降序排序，并存储到（d）中指定的（n1×n2）点的 BIN32 位数据表（排序后）中。

-[DSORT2 (s) （n1）（n2）（d）（n3）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了数据表的软元件起始编号	-	有符号 BIN32 位	ANY32	--
(n1)	数据（行）数	[1,32]	有符号 BIN32 位	ANY32	DINT
(n2)	组数据（列）数	[1,6]	有符号 BIN32 位	ANY32	DINT
(d)	存储运算结果的软元件起始编号	-	有符号 BIN32 位	ANY32	--
(n3)	作为排序基准的组数据（行）的列编号	-	有符号 BIN32 位	ANY32	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSORT2	参数 1															●	●	●	●	●	●	●				●	
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
	参数 3											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
	参数 4															●	●	●	●	●	●	●				●	
	参数 5											●	●	●	●	●	●	●	●	●	●	●	●	●		●	

功能

• 将（s）中指定的（n1×n2）点的 BIN32 位数据表（排序源），以（n3）列的组数据为基准，对数据行进行升序或降序排序，并存储到（d）中指定的（n1×n2）点的 BIN32 位数据表（排序后）中。

• 以排序源中（n1）=K3、（n2）=K4 的情况为例，数据表构成如下所示。当为排序后数据表的情况下，应将（s）替换为（d）。

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=3 的情况下	行号 1	（s）+1、（s）	（s）+3、（s）+2	（s）+5、（s）+4	（s）+7、（s）+6
	行号 2	（s）+9、（s）+8	（s）+11、（s）+10	（s）+13、（s）+12	（s）+15、（s）+14
	行号 3	（s）+17、（s）+16	（s）+19、（s）+18	（s）+21、（s）+20	（s）+23、（s）+22

• 排序通过 SM165 的 ON/OFF 状态设置

	排序的顺序设置说明
SM165=ON	降序
SM165=OFF	升序

• 指令输入为 ON 时开始数据对齐，（n1）扫描后数据对齐结束，指令执行结束标志 SM229 置为 ON。

• 根据下述排序源数据，动作示例如下所示。此外，在第 1 列中预先放入管理编号等序号，就可根据该内容判断原来的行号，非常方便。

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（s）+1、（s）	（s）+3、（s）+2	（s）+5、（s）+4	（s）+7、（s）+6
		1	150	45	20

福州富昌维控电子科技有限公司

	行号2	（s）+9、（s）+8	（s）+11、（s）+10	（s）+13、（s）+12	（s）+15、（s）+14
		2	180	50	40
	行号3	（s）+17、（s）+16	（s）+19、（s）+18	（s）+21、（s）+20	（s）+23、（s）+22
		3	160	70	30
	行号4	（s）+25、（s）+24	（s）+27、（s）+26	（s）+29、（s）+28	（s）+31、（s）+30
		4	100	20	8
	行号5	（s）+33、（s）+32	（s）+35、（s）+34	（s）+37、（s）+36	（s）+39、（s）+38
		5	150	50	45

• 按（n3）=K2(列号 2)执行了指令时的排序结果（升序的情况下 SM165=OFF）

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（s）+1、（s）	（s）+3、（s）+2	（s）+5、（s）+4	（s）+7、（s）+6
		4	100	20	8
	行号 2	（s）+9、（s）+8	（s）+11、（s）+10	（s）+13、（s）+12	（s）+15、（s）+14
		1	150	45	20
	行号 3	（s）+17、（s）+16	（s）+19、（s）+18	（s）+21、（s）+20	（s）+23、（s）+22
		5	150	50	45
	行号 4	（s）+25、（s）+24	（s）+27、（s）+26	（s）+29、（s）+28	（s）+31、（s）+30
		3	160	70	30
	行号 5	（s）+33、（s）+32	（s）+35、（s）+34	（s）+37、（s）+36	（s）+39、（s）+38
		2	180	50	40

• 按（n3）=K3(列号 3)执行了指令时的排序结果（降序的情况下 SM165=ON）

		组数（n2）个（n2）=K4 的情况下			
		列号 1	列号 2	列号 3	列号 4
		管理编号	身高	体重	年龄
数据数（n1）=5 的情况下	行号 1	（s）+1、（s）	（s）+3、（s）+2	（s）+5、（s）+4	（s）+7、（s）+6
		3	160	70	30
	行号 2	（s）+9、（s）+8	（s）+11、（s）+10	（s）+13、（s）+12	（s）+15、（s）+14
		2	180	50	40
	行号 3	（s）+17、（s）+16	（s）+19、（s）+18	（s）+21、（s）+20	（s）+23、（s）+22
		5	150	50	45
	行号 4	（s）+25、（s）+24	（s）+27、（s）+26	（s）+29、（s）+28	（s）+31、（s）+30
		1	150	45	20
	行号 5	（s）+33、（s）+32	（s）+35、（s）+34	（s）+37、（s）+36	（s）+39、（s）+38
		4	100	20	8

注意事项

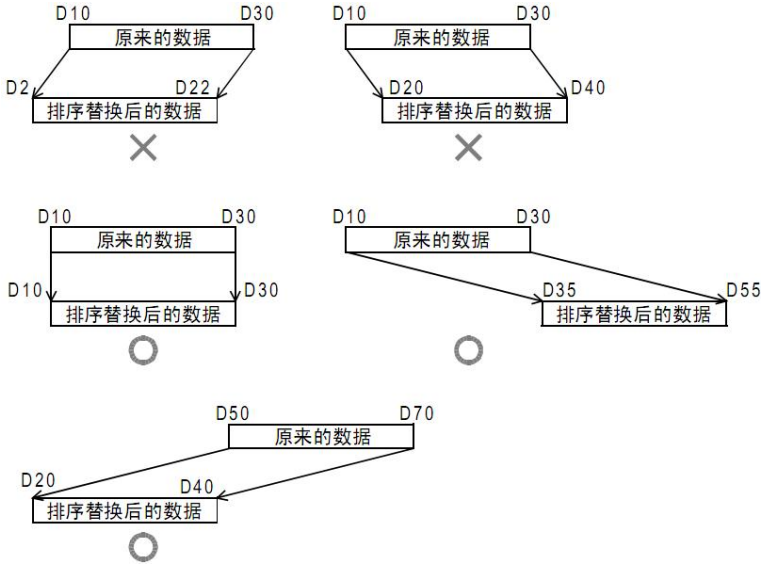
• 动作中请勿改变操作数及数据内容。

• 再次执行时，应将指令输入置为 OFF 一次。

• SORT2 指令在程序中最多只能写入 2 次。

• （s）和（d）中指定同一软元件的情况下，源数据改写为排序后的数据顺序。请特别注意在执行结束前不要改变（s）内容。

• 请勿使源数据和排序后的数据错位重叠。



错误码

错误码	内容
4084H	(n1)中指定的数值超出下述范围时。[1,32]
	(n2)中指定的数值超出下述范围时。[1,6]
	(n3)中指定的数值超出下述范围时。1~n2
4085H	读应用指令(s)、(d)、(n1)、(n2)、(n3)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。
4089H	应用指令超出限制使用条数。

示例

参考功能说明。

SWAP/16 位数据高低字节交换

SWAP(P)

对（d）中指定的软元件的上下各 8 位的值进行变换。

-[SWAP （d）]

内容、范围、数据类型

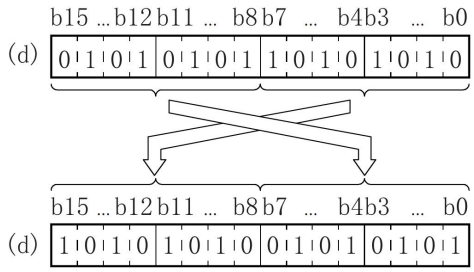
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	高低字节互换的字软元件	-	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SWAP	参数 1												●	●	●	●	●	●	●	●						●	●

功能

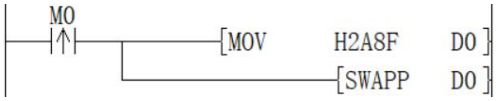
对（d）中指定的软元件的上下各 8 位的值进行变换。



错误码

错误码	内容
4085H	读应用指令(d)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 M0 上升沿触发，将 D0 的低 8 位和高 8 位互换，得到 H8F2A。

DSWAP/32 位数据高低字节交换

DSWAP(P)

（d）及（d）+1 中指定的软元件，将分别转换上下各 8 位的值。

-[DSWAP （d）]

内容、范围、数据类型

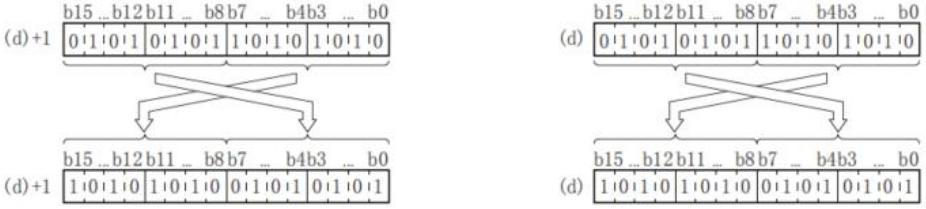
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	高低字节互换的字软元件	-	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSWAP	参数 1												●	●	●	●	●	●	●	●	●	●				●	●

功能

- （d）及（d）+1 中指定的软元件，将分别转换上下各 8 位的值。



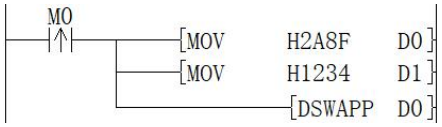
注意事项

- 如果使用连续执行指令，则每个扫描周期都将进行转换。

错误码

错误码	内容
4085H	读应用指令(d)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 M0 上升沿触发，分别将 D0、D1 的低 8 位和高 8 位互换，得到 D0=H8F2A，D1=H3412。

软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D0	0	1	0	1	0	1	0	0	1	1	1	1	0	0	0	1	8F2A
D1	0	1	0	0	1	0	0	0	0	0	1	0	1	1	0	0	3412
D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

BTOW/字节单位数据合并

BTOW(P)

将（s）中指定的软元件编号以后中存储的（n）字节的 BIN16 位数据的低位 8 位，合并为字单位后，存储到（d）中指定的软元件编号以后。

-[BTOW （s） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了以字节单位合并的数据的起始软元件	-	有符号 BIN16 位	ANY16	--
(d)	存储以字节单位合并的结果的起始软元件	-	有符号 BIN16 位	ANY16	--
(n)	合并的字节数据的个数	[0, 32767]	有符号 BIN16 位	ANY16	INT

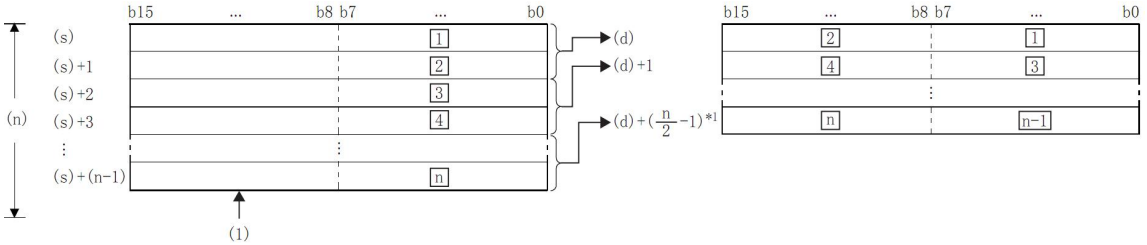
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BTOW	参数 1															●	●	●	●	●						●	●
	参数 2															●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（s）中指定的软元件编号以后中存储的（n）字节的 BIN16 位数据的低位 8 位，合并为字单位后，存储到（d）中指定的软元件编号以后。

• （s）中指定的软元件编号以后中存储的（n）字的数据的高位 8 位将被忽略。此外，（n）为奇数的情况下，存储了第（n）字节的数据的软元件的高位 8 位中将存储 0。



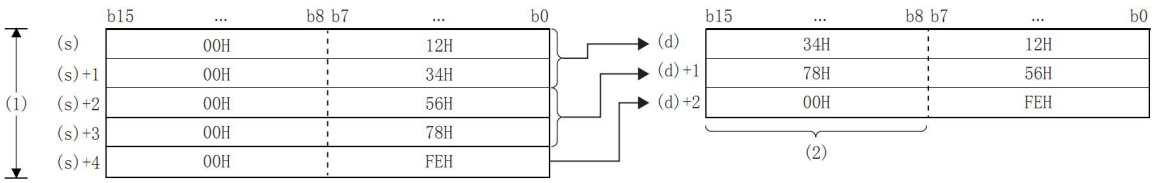
□：第□字节数据；

（1）：忽略高位字节

*1：小数点以下进位。

例

（n）=5 的情况下，将（s）+（s）+4 的低位 8 位为止的数据存储到（d）+（d）+2 中。



（1）：（n）=5 的情况下

（2）：变为 00H

• 通过在（n）中设置字节数，（s）中指定的字节数据的范围及存储（d）中指定的合并数据的软元件的范围将被自动确定。

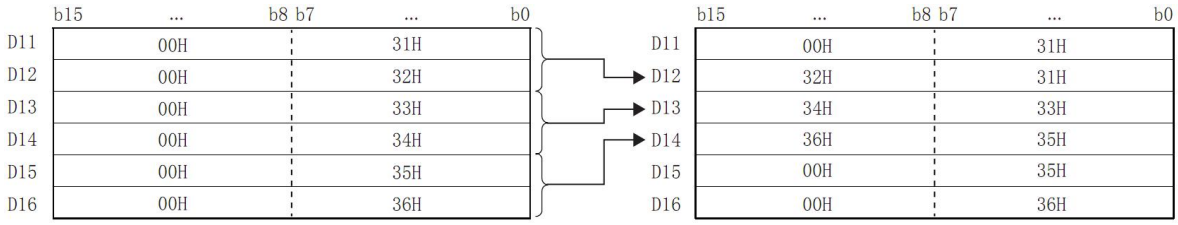
• （n）中指定的字节数为 0 的情况下，不进行处理。

• （s）中指定的字节数据存储软元件的高位 8 位将被忽略，低位 8 位将成为对象。

福州富昌维控电子科技有限公司

例：

将 D11~D16 的低位 8 位存储到 D12~D14 中的情况下。



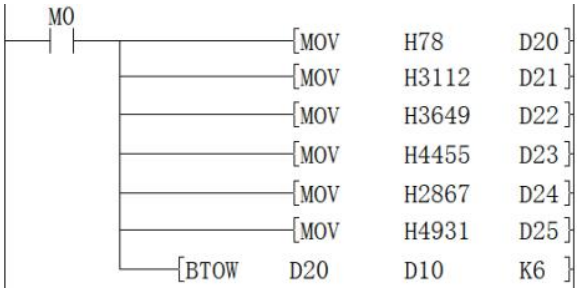
• 即使存储了合并前数据的软元件范围与存储合并后数据的软元件范围重复的情况下，也将作为正常处理。

存储了合并的数据存储的软元件范围	存储合并后数据的软元件范围
(s) +0 ~ (s) + (n) -1	(d) ~ (d) + (n/2-1)

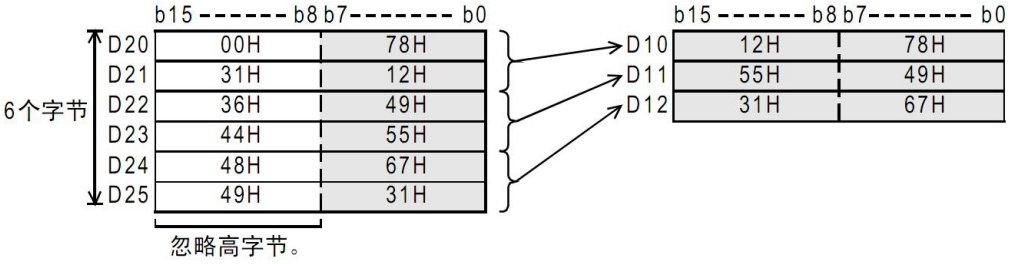
错误码

错误码	内容
4084H	(n)中指定的值超出范围。0~32767。
4085H	读应用指令(s)、(d)、(n)中指定的软元件超出相应软元件的范围时。

示例



当 M0 为 ON 时，将 D20~D25 的数据按照字节单位分离，然后保存到 D10~D12 中。



WTOB/字节单位数据分离

WTOB(P)

将（s）中指定的软元件编号以后中存储的 BIN16 位数据分离为（n）字节后，存储到（d）中指定的软元件编号以后。

-[WTOB （s） （d） （n）]

内容、范围、数据类型

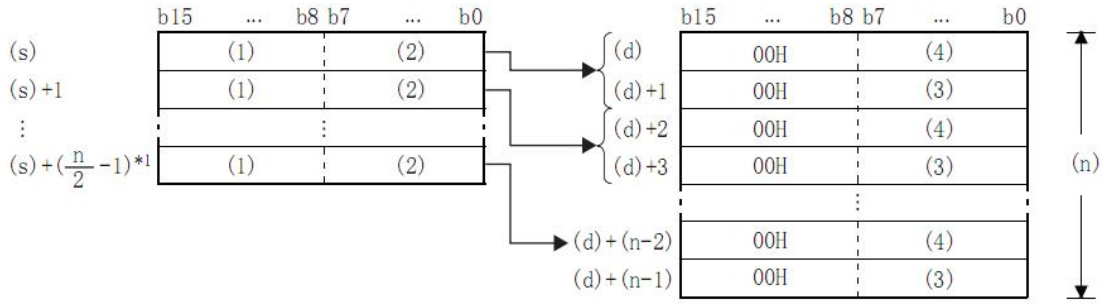
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了以字节单位分离的数据的起始软元件	-	有符号 BIN16 位	ANY16	--
(d)	存储以字节单位分离的结果的起始软元件	-	有符号 BIN16 位	ANY16	--
(n)	分离的字节数据的个数	[0,32767]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
WTOB	参数 1															●	●	●	●	●						●	●
	参数 2															●	●	●	●	●						●	●
	参数 3																	●	●	●			●	●		●	●

功能

• 将（s）中指定的软元件编号以后中存储的 BIN16 位数据分离为（n）字节后，存储到（d）中指定的软元件编号以后。



（1）：高位字节；

（2）：低位字节；

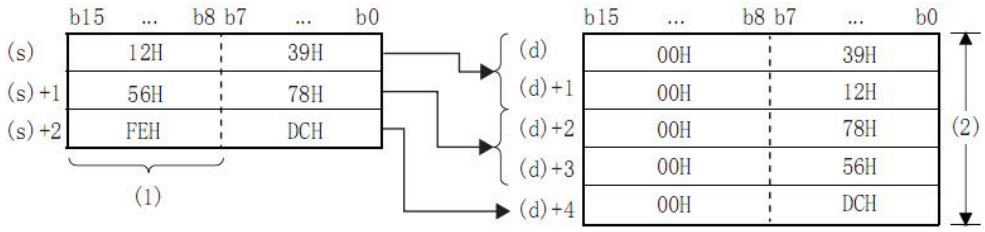
（3）：高位字节数据；

（4）：低位字节数据；

（5）*1：小数点以下进位。

例

（n）=5 的情况下，将（s）～（s）+2 的低位 8 位为止的数据存储到（d）～（d）+4 中：



（1）：（n）=5 的情况下被忽略。

（2）：（n）=5 的情况下。

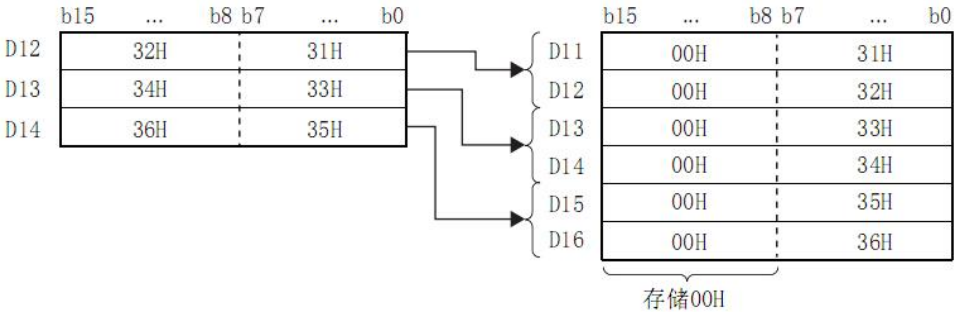
• 通过在（n）中设置字节数，（s）中指定的 BIN16 位数据的范围及存储（d）中指定的字节数据的软元件的范围将被自动确定。

• （n）中指定的字节数为 0 的情况下，不进行处理。

• （d）中指定的字节数据存储软元件的高位 8 位中将被自动存储 00H。

例

将 D12~D14 存储到 D11~D16 的低位 8 位中的情况下：



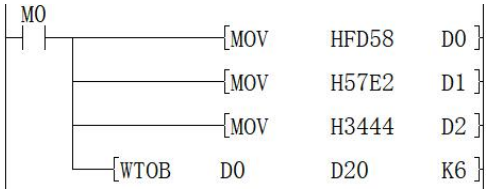
• 即使存储了分离的数据的软元件范围与存储分离后数据的软元件范围重复的情况下，也将作为正常处理。

存储了分离的数据的软元件范围	存储分离后数据的软元件范围
（s）～（s）+（n/2-1）	（d）+0 ～（d）+（n）-1

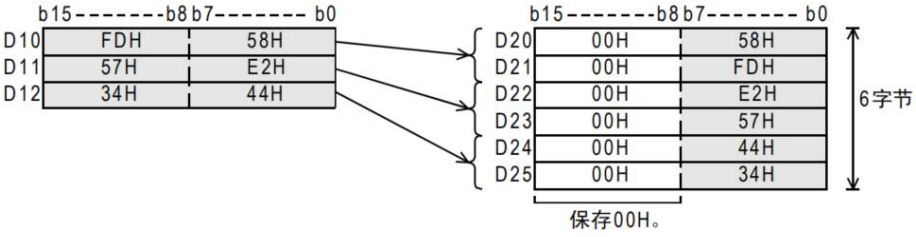
错误码

错误码	内容
4084H	(n)指定的值超出范围。[0,32767]
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 M0 为 ON 时，将 D10~D12 的数据按照字节单位分离，然后保存到 D20~D25 中。



DIS/16 位数据的 4 位分离

DIS(P)

将（s）中指定的 BIN16 位数据的低位（n）位数（1 位数 4 位）的数据存储到（d）中指定的软元件开始的（n）点的低位 4 位中。

-[DIS （s） （d） （n）]

内容、范围、数据类型

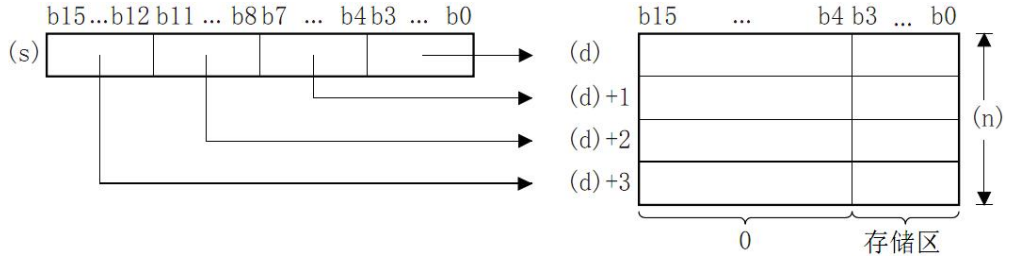
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储分离的数据的起始软元件	-	有符号 BIN16 位	ANY16	--
(d)	存储分离后的数据的起始软元件	-	有符号 BIN16 位	ANY16	--
(n)	分离数（0 代表无处理）	[0,4]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DIS	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（s）中指定的 BIN16 位数据的低位（n）位数（1 位数 4 位）的数据存储到（d）中指定的软元件开始的（n）点的低位 4 位中。



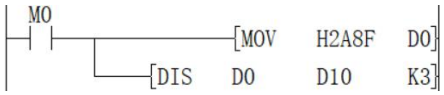
• （s）中指定的软元件开始的（n）点的高位 12 位将变为 0。

• （n）=0 时将变为无处理，（d）的软元件开始的（n）点的内容不变化。

错误码

错误码	内容
4084H	(n)中输入了超出指定范围的数据。[0,4]
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 M0 为 ON 后，将 D0 每隔 4 个位分离后，保存到 D10~D12 中，其结果为 D10 = HF，D11=H8，D12=HA。

UNI/16 位数据的 4 位合并

UNI(P)

将（s）中指定的软元件开始的（n）点的 BIN16 位数据的低位 4 位，合并到（d）中指定的 BIN16 位软元件中。

-[UNI （s） （d） （n）]

内容、范围、数据类型

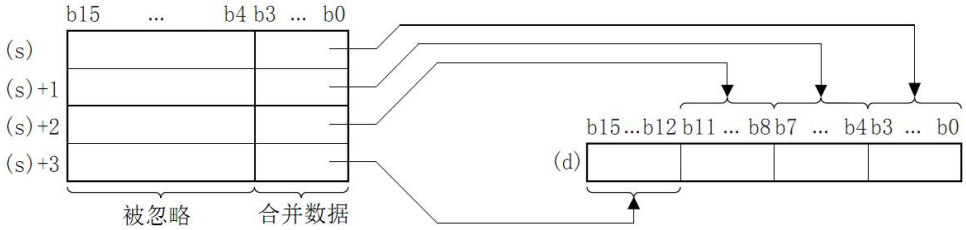
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储合并的数据的起始软元件	-	有符号 BIN16 位	ANY16	--
(d)	存储合并后的数据的起始软元件	-	有符号 BIN16 位	ANY16	--
(n)	合并数	[0,4]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
UNI	参数 1															●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（s）中指定的软元件开始的（n）点的 BIN16 位数据的低位 4 位，合并到（d）中指定的 BIN16 位软元件中。



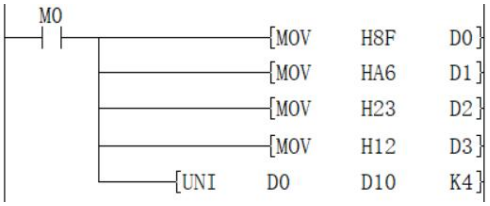
• （d）中指定的软元件的高位（4-n）的位数的位将变为 0。

• （n）=0 时将变为无处理，（d）的软元件的内容不变化。

错误码

错误码	内容
4084H	（n）为 0~4 以外的值时。
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 M0 为 ON 后，将 D0~D3 的低 4 位结合后，保存到 D10 中，其值为 H236F。

ZRST/数据批量复位

ZRST(P)

在相同类型的（d1）与（d2）中指定的软元件之间进行批量复位。中断运行，进行初始运行或对控制数据进行复位等时使用。

-[ZRST （d1）（d2）]

内容、范围、数据类型

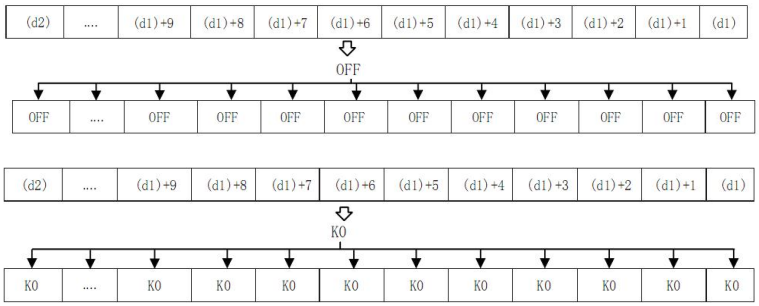
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d1)	批量复位的起始位/字软元件编号	-	位/有符号 BIN16 位	ANY_ELEMENTARY	--
(d2)	批量复位的最终位/字软元件编号	-	位/有符号 BIN16 位	ANY_ELEMENTARY	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ZRST	参数 1		●	●	●	●						●	●	●	●	●	●	●	●	●	●	●				●	●
	参数 2		●	●	●	●						●	●	●	●	●	●	●	●	●	●	●				●	●

功能

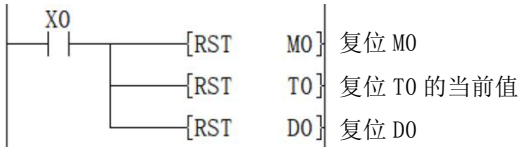
• 在相同类型的（d1）与（d2）中指定的软元件之间进行批量复位。



（d1）、（d2）为位软元件时，在（d1）～（d2）的整个软元件范围内写入 OFF（复位）。

（d1）、（d2）为字软元件时，在（d1）～（d2）的整个软元件范围内写入 K0。

• 作为软元件的单独复位指令，可对位软元件或字软元件使用 RST 指令。



• 常数（例：K0）的批量写入指令有 FMOV（P）指令，可向字软元件（包括位软元件的位数指定）写入 0。

注意事项

• 请为（d1）、（d2）指定相同类型的编号，并使（d1）编号 < （d2）编号。当（d1）编号≥（d2）编号时，对（d1）中指定的软元件，仅复位 1 点。

• ZRST(P)指令为 16 位指令，可为（d1）、（d2）指定（LC）与（HSC）软元件。

错误码

错误码	内容
4084H	(d1)中指定的软元件类型与(d2)中指定的软元件类型不同时。
4085H	读应用指令(d1)、(d2)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d1)中指定的软元件超出相应软元件的范围时。

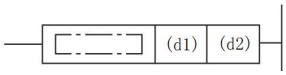
示例

福州富昌维控电子科技有限公司

ZSET/数据批量置位

ZSET(P)

在相同类型的（d1）与（d2）中指定的位软元件之间进行批量置位。



内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d1)	批量置位的起始位软元件编号	-	位	ANY_BOOL	--
(d2)	批量置位的最终位软元件编号	-	位	ANY_BOOL	--

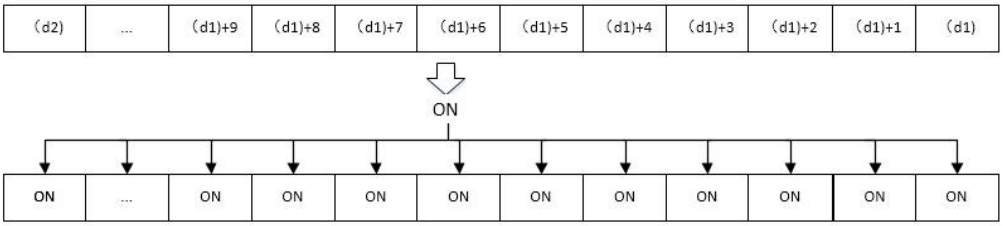
使用的软元件

指令	参数	软元件																								偏移修饰
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]
ZSET	参数 1		●	●	●	●					●															●
	参数 2		●	●	●	●					●															●

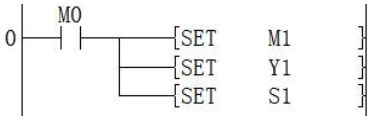
功能

• 在相同类型的（d1）与（d2）中指定的位软元件之间进行批量置位。

• 在（d1）～（d2）的整个软元件范围内写入 ON（置位）。



• 作为软元件的单独置位指令，可对位软元件使用 SET 指令。



（1）置位 M1。

（2）置位 Y1。

（3）置位 S1。

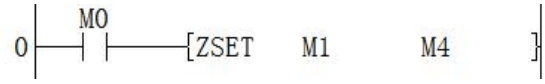
注意事项

• 请为（d1）、（d2）指定相同类型的编号，并使（d1）编号＜（d2）编号。当（d1）编号≥（d2）编号时，对（d1）中指定的软元件，仅置位 1 点。

错误码

错误码	内容
4084H	(d1)中指定的软元件类型与(d2)中指定的软元件类型不同时。
4085H	读应用指令(d1)、(d2)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d1)中指定的软元件超出相应软元件的范围时。
4087H	(d1)中指定的软元件类型与(d2)中指定的软元件类型不为位时。

示例



该梯形图指令的功能是将 M1~M4 软元件的值置 ON。

CRC/循环冗余校验指令

CRC(P)

将(s)中指定的软元件作为起始，生成(n)点的 8 位数据(字节单位)的 CRC 值，并存储到(d)中。

-[CRC(P) (s) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储 CRC 值生成对象的数据的软元件起始编号	-	有符号 BIN16	ANY16	--
(d)	生成的 CRC 值的存储目标软元件编号	-	有符号 BIN16	ANY16	--
(n)	计算 CRC 值的 8 位数据(字节)数或存储了数据数的软元件编号	[1,256]	无符号 BIN16	ANY16_U	UINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
CRC	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●			●

功能

CRC（Cyclic Redundancy Check，简称：CRC）计算为通信中使用的出错检查方法之一。出错检查方法除 CRC 以外，还有奇偶校验以及和校验（校验和），计算水平奇偶校验值及和校验值可使用 CCD（P）指令。

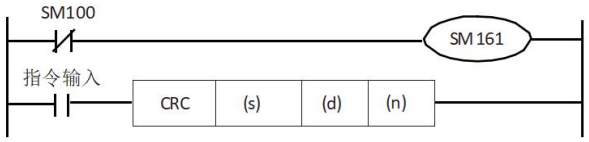
该指令中生成 CRC 值（CRC-16）的多项式使用“X16+X15+X2+1”。

该指令在计算时使用的模式有 16 位转换模式和 8 位转换模式。关于各个模式的动作，请参阅以后的内容。

(1)16 位转换模式(SM161=OFF 时)

对(s)的软元件的高位 8 位(字节)和低位 8 位(字节)进行运算。结果存储到(d)中指定的软元件 1 点的 16 位。

下述程序的情况下，如下所示执行转换。



16 位转换模式

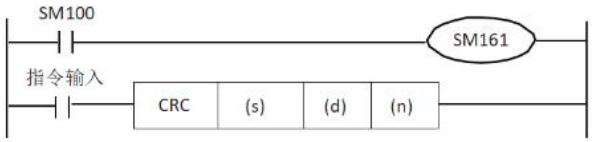
			CRC 值生成对象数据保存目标										CRC 值存储目标	
			(s)		(s)+1		(s)+2		...	(s)+(n)/2-1			(d)	
			低位字节	高位字节	低位字节	高位字节	低位字节	高位字节		低位字节	高位字节		低位字节	高位字节
示例 (s)=D100，(d)=D0，(n)=6	软元件		D100 低	D100 高	D101 低	D101 高	D102 低	D102 高					D0 低	D0 高
	对象数据的内容	8 位	01H	03H	03H	02H	00H	14H					E4H	41H
		16 位	0301H		0203H		1400H						41E4H	

(2)8 位转换模式(SM8161=ON 时)

8 位转换模式中，仅对(s)的软元件的低位 8 位(低位字节)进行运算。结果从(d)中指定的软元件开始使用 2 点，在(d)中存储低位 8 位(字节)，在(d)+1 中存储高位 8 位(字节)。

下述程序的情况下，如下所示执行转换。

福州富昌维控电子科技有限公司



8 位转换模式

		CRC 值生成对象数据保存目标								CRC 值存储目标	
		(s)	(s)+1	(s)+2	(s)+3	(s)+4	(s)+5	...	(s)+(n)-1	(d)	(d)+1
		低位字节	低位字节	低位字节	低位字节	低位字节	低位字节		低位字节	低位字节	低位字节
示例) (s)=D100, (d)=D0, (n)=6	软元件	D100 低	D101 低	D102 低	D103 低	D104 低	D105 低			D0	D1
	对象数据的内容	01H	03H	03H	02H	00H	14H			E4H	41H

• 在 CRC(P)指令中 CRC 值(CRC-16)的生成多项式中使用“X16+X15+X2+1”，但 CRC 值也有很多标准化的生成多项式。如果该生成多项式不同，将变为完全不同的 CRC 值，应加以注意。主要的 CRC 值生成多项式如下所示。

名称	生成多项式
CRC-12	$X^{12}+X^{11}+X^3+X^2+X+1$
CRC-16	$X^{16}+X^{15}+X^2+1$
CRC-32	$X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^{5+}X^4+X^2+X+1$
CRC-CCITT	$X^{16}+X^{12}+X^5+1$

注意事项

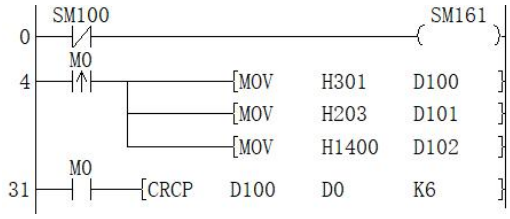
(s1)使用 KnX、KnY、KnM、KnS 时，n 必须指定为 4。

错误码

错误码	内容
4084H	(n)的范围超出 1~256。
4085H	(s)需要转换的数据地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4087H	(s)(d)使用了不支持的软元件类型。

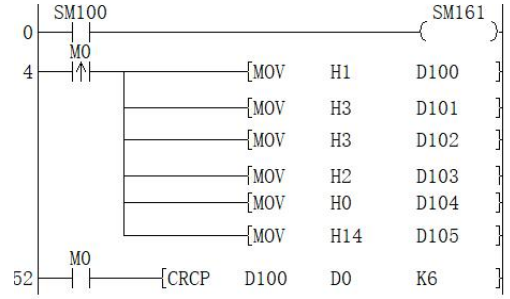
示例

1. 16 位转换模式



M0 置 ON 后，D0 = 41E4H

2. 8 位转换模式



M0 置 ON 后，D0 = E4H，D1 = 41H

7.5 矩阵输入指令

MTR/矩阵输入

MTR

以 8 点输入和(n)点输出(晶体管)的时间分割方式读取 8 点×n 列的输入信号(开关)的指令。

-[MTR (s) (d1) (d2) (n)]

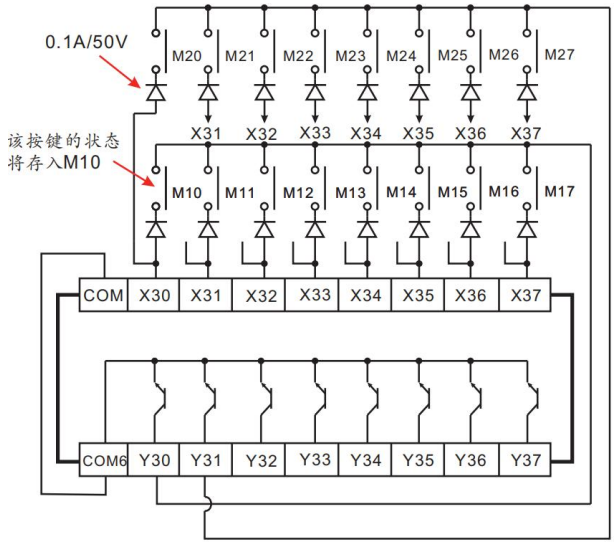
内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	矩阵的行信号输入的起始软元件(X)编号 X000、X010、X020…最终的输入 X 编号为止。连续占用 8 个。	X 最低位的位数编号只能为 0	位	ANY_BOOL	--
(d1)	矩阵的列信号输出的起始软元件(Y)编号 Y000、Y010、Y020…最终的输出 Y 编号为止。连续占用 8 个。	Y 最低位的位数编号只能为 0	位	ANY_BOOL	--
(d2)	ON 输出目标地址的起始软元件(Y,M,S)编号 Y000、Y010、Y020…、M000、M010、M020…、S000、S010、S020…最终的 Y,M,S 编号为止。Y 连续占用 8*(n)个，其他连续占用 10*(n)个。	-	位	ANY_BOOL	--
(n)	设定矩阵输入的列数。	[2,8]	无符号 BIN16 位	ANY16_U	--

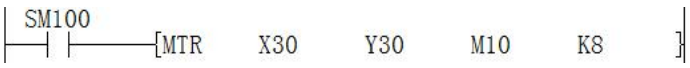
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MTR	参数 1	●																								
	参数 2		●																							
	参数 3		●	●	●	●																				
	参数 4																						●	●		

功能



• 该指令一般使用常 ON 触点 SM100



根据图中的例子说明：

• 当 Y30 和 X30 被接通时 M10 会置 ON ， Y30 和 X34 被接通时 M14 会置 ON ， Y31 和 X36 被接通时 M26 会置 ON；

• (d2)是建议使用最低为 0，主要是当使用 M4 这样的地址的话第一个起始就是 M4，然后会连续占用到 M11，这样不方便计算和查看，所以推荐使用最低为 0 的软元件。

使用的特殊软元件

软元件	内容
SM229	一周期执行完成后 SM229 会置 ON。

注意事项

MTR 指令只能同时运行一条指令。

错误码

错误码	内容
4085H	(s) (n)读取地址超出软元件范围。
	(s)使用低位不是 0 的编号元件。
4086H	(d1) (d2)写入地址超出软元件范围。
	(d1) 使用低位不是 0 的编号元件。
4084H	(n)不在范围 2~8 内。
4089H	MTR 指令在同一时间内执行了多条。

7.6 方便指令

ABSD/BIN16 位数据绝对方式

ABSD

创建与计数器当前（BIN16 位值）对应的多种输出模式。

-[ABSD (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储数据表（上升沿点、下降沿点）的起始软元件编号	-	有符号 BIN16 位	ANY16	--
(s2)	与数据表进行比较的当前值监控用计数器编号	-	有符号 BIN16 位	ANY16	--
(d)	输出的起始位软元件的点数	-	位	ANY16_BOOL	--
(n)	表的行数及输出的位软元件的点数	[1,64]	有符号 BIN16 位	ANY16	--

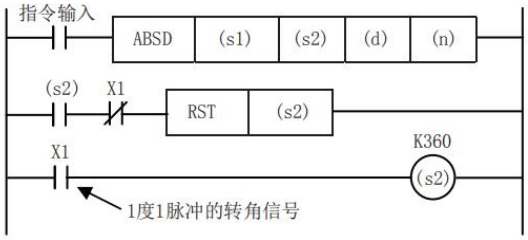
使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ABSD	参数 1											●	●	●	●	●	●	●	●	●						●	
	参数 2																●									●	
	参数 3		●	●	●	●					●															●	
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	

功能

• 以旋转台旋转 1 周（0~360 度）对输出的 ON/OFF 控制为例进行说明。（1 度 1 脉冲的转角信号）

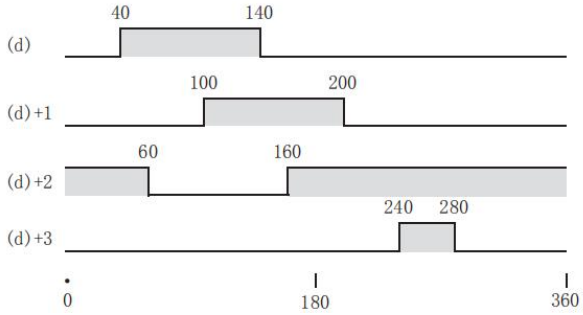
• 将（s1）开始的（n）行的数据表（（n）行×占用 2 点）与计数器当前值（s2）进行比较，在旋转 1 周的过程中从（d）到连续（n）点为止对输出进行 ON/OFF 控制。



• 预先使用传送指令将如下所示的数据写入（s1）~（s1）+2（n）-1。例如，上升沿点数据预先存储 16 位数据到偶数编号的软元件，下降沿点数据预先存储 16 位数据到奇数编号的软元件。

上升沿点		下降沿点		对象输出
-	数据值（示例）	-	数据值（示例）	
（S1）	40	（S1）+1	140	（d）
（S1）+2	100	（S1）+3	200	（d）+1
（S1）+4	160	（S1）+5	60	（d）+2
（S1）+6	240	（S1）+7	280	（d）+3
...	-	...	-	...
（S1）+2（n）-2		（S1）+2（n）-1		（d）+n-1

• 如果指令输入置为 ON，（d）为起始，（n）点为如下所示的输出模式。各上升沿点及下降沿点可通过（s1）~（s1）+2（n）-1 的数据改写单独更改。



注意事项

• 在（s1）中进行位软元件的位数指定的情况下，软元件编号应指定 16 的倍数（0、16、32、64……），位数应仅指定 K4。

• 对象输出点数根据（n）的值确定。（1≤（n）≤64）

• 即使指令输入置为 OFF，输出也不变化。

错误码

错误码	内容
4084H	(n)中指定的数值超出下述范围时。[1,64]
4085H	读应用指令(s1)、(s2)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

参考功能说明中的示例。

DABSD/BIN32 位数据绝对方式

DABSD

创建与计数器当前（BIN32 位值）对应的多种输出模式。

-[DABSD（s1） (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储数据表（上升沿点、下降沿点）的起始软元件编号	-	有符号 BIN32 位	ANY32	--
(s2)	与数据表进行比较的当前值监控用计数器编号	-	有符号 BIN32 位	ANY32	--
(d)	输出的起始位软元件的点数	-	位	ANY16_BOOL	--
(n)	表的行数及输出的位软元件的点数	[1,64]	有符号 BIN32 位	ANY32	--

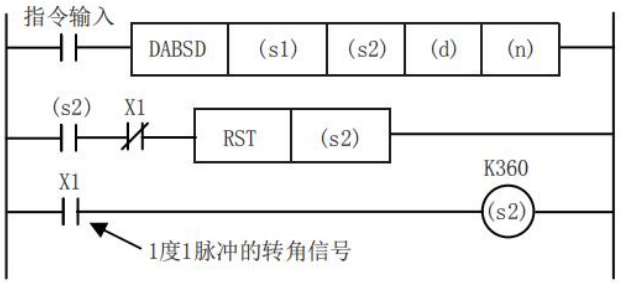
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DABSD	参数 1											●	●	●	●	●	●	●	●	●	●	●				●	
	参数 2																●				●	●				●	
	参数 3		●	●	●	●					●															●	
	参数 4											●	●	●	●	●	●	●	●	●	●	●	●	●		●	

功能

• 以旋转台旋转 1 周（0~360 度）对输出的 ON/OFF 控制为例进行说明。（1 度 1 脉冲的转角信号）

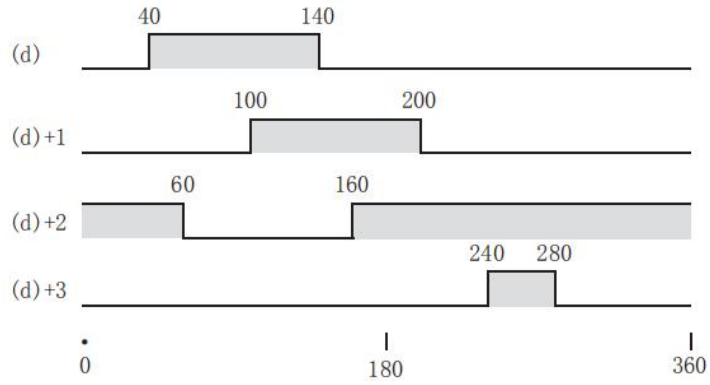
• 将（s1）开始的（n）行的数据表（（n）行×占用 4 点）与计数器当前值（s2）进行比较，在旋转 1 周的过程中从（d）到连续（n）点为止对输出进行 ON/OFF 控制。



• 预先使用传送指令将如下所示的数据写入（s1）、（s1）+1~（s1）+4（n）-2、（s1）+4（n）-1。例如，上升沿点数据预先存储 32 位数据到偶数编号的软元件，下降沿点数据预先存储 32 位数据到奇数编号的软元件。

上升沿点		下降沿点		对象输出
-	数据值（示例）	-	数据值（示例）	
（S1）+1、（S1）	40	（S1）+3、（S1）+2	140	（d）
（S1）+5、（S1）+4	100	（S1）+7、（S1）+6	200	（d）+1
（S1）+9、（S1）+8	160	（S1）+11、（S1）+10	60	（d）+2
（S1）+13、（S1）+12	240	（S1）+15、（S1）+14	280	（d）+3
...	-	...	-	...
（S1）+4（n）-3、（S1）+4（n）-4		（S1）+4（n）-1、（S1）+4（n）-2		（d）+n-1

• 如果指令输入置为 ON，（d）为起始，（n）点为如下所示的输出模式。各上升沿点及下降沿点可通过（s1）~（s1）+2（n）-1 的数据改写单独更改。



注意事项

• DABSD 指令中可指定高速计数器。指定高速计数器的情况下，对于计数器当前值，输出模式中由于扫描循环会有响应延迟。

• 在（s1）中进行位软元件的位数指定的情况下，软元件编号应指定 16 的倍数（0、16、32、64……），位数应仅指定 K8。

• 对象输出点数根据（n）的值确定。（1≤（n）≤64）

• 即使指令输入置为 OFF，输出也不变化。

错误码

错误码	内容
4084H	(n)中指定的数值超出下述范围时。[1,64]
4085H	读应用指令(s1)、(s2)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

参考功能说明中的示例。

SER/16 位数据查找

SER(P)

从数据的表中搜索相同数据及最大值、最小值。

-[SER （s1） （s2） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	搜索相同数据、最大值、最小值的起始软元件编号	-	有符号 BIN16 位	ANY16	--
(s2)	搜索相同数据的值或其存储目标软元件编号	-	有符号 BIN16 位	ANY16	--
(d)	搜索相同数据、最大值、最小值后存储其个数的起始软元件编号	-	有符号 BIN16 位	ANY16	--
(n)	搜索相同数据、最大值、最小值的个数	[1,256]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SER	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3												●	●	●	●	●	●	●	●						●	●
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 对于（s1）为起始的（n）个数据，搜索与（s2）的 BIN16 位数据相同的数据，将结果存储到（d）～（d）+4 中。

• 有相同数据的情况下，在（d）为起始的 5 点的软元件中，存储相同数据的个数、首次/最终位置及最大值、最小值的位置。

• 无相同数据的情况下，在（d）为起始的 5 点的软元件中，存储相同数据的个数、首次/最终位置及最大值、最小值的位置。但是，在（d）为起始的 3 点的软元件（相同数据的个数、首次\最终位置）中，存储 0。

• 搜索结果表的构成和数据示例如下所示。（n=10）

被搜索软元件（S1）	被搜索数据（S1）的值（示例）	比较数据（S2）的值（示例）	数据的位置	搜索结果		
				最大值（d）+4	一致（d）	最小值(d+3)
（S1）	K100	K100	0		○（首次）	
（S1）+1	K111		1			
（S1）+2	K100		2		○	
（S1）+3	K98		3			
（S1）+4	K123		4			
（S1）+5	K66		5			○
（S1）+6	K100		6		○（最终）	
（S1）+7	K95		7			
（S1）+8	210		8	○		
（S1）+9	K88		9			

• 根据上述示例的搜索结果表如下所示。

软元件编号	内容	搜索结果项目
（d）	3	相同数据的个数
（d）+1	0	相同数据的位置（首次）
（d）+2	6	相同数据的位置（最终）
（d）+3	5	最小值的最终位置
（d）+4	8	最大值的最终位置

注意事项

• 进行代数大小比较。（-10<2）

• 数据中的最小值、最大值有多个时，存储各自后侧的位置。

• 如果本指令驱动，搜索结果（d）占用（d）、（d）+1、（d）+2、（d）+3、（d）+4 的 5 点。请注意不要与用于机械控制的软元件重复。

错误码

错误码	内容
4084H	应用指令中(n)输入了超出可指定范围的数据。[0,256]
4085H	读应用指令(s1)、(s2)、(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

参考功能说明中的示例。

DSER/32 位数据查找

DSER(P)

从数据的表中搜索相同数据及最大值、最小值。

-[DSER （s1） （s2） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	搜索相同数据、最大值、最小值的起始软元件编号	-	有符号 BIN32 位	ANY32	--
(s2)	搜索相同数据的值或其存储目标软元件编号	-	有符号 BIN32 位	ANY32	--
(d)	搜索相同数据、最大值、最小值后存储其个数的起始软元件编号	-	有符号 BIN32 位	ANY32	--
(n)	搜索相同数据、最大值、最小值的个数	[1,128]	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSER	参数 1											●	●	●	●	●	●	●	●	●	●	●				●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●
	参数 4											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●

功能

• 对于（s1）+1、（s1）为起始的（n）个数据，搜索与（s2）+1、（s2）的 BIN32 位数据相同的数据，将结果存储到（d）+1、（d）～（d）+9、（d）+8 中。

• 有相同数据的情况下，在（d）+1、（d）为起始的 BIN32 位数据 5 点的软元件中，存储相同数据的个数、首次/最终位置及最大值、最小值的位置。

• 无相同数据的情况下，在（d）+1、（d）为起始的 BIN32 位数据 5 点的软元件中，存储相同数据的个数、首次/最终位置及最大值、最小值的位置。但是，在（d）+1、（d）为起始的 BIN32 位 3 点的软元件（相同数据的个数、首次\最终位置）中，存储 0。

• 搜索结果表的构成和数据示例如下所示。（n=10）

被搜索软元件（S1）	被搜索数据（S1）的值（示例）	比较数据（S2）的值（示例）	数据的位置	搜索结果		
				最大值（d）+4	一致（d）	最小值（d+3）
（S1）+1、（S1）	K100	K100	0		○（首次）	
（S1）+3、（S1）+2	K111		1			
（S1）+5、（S1）+4	K100		2		○	
（S1）+7、（S1）+6	K98		3			
（S1）+9、（S1）+8	K123		4			
（S1）+11、（S1）+10	K66		5			○
（S1）+13、（S1）+12	K100		6		○（最终）	
（S1）+15、（S1）+14	K95		7			
（S1）+17、（S1）+16	210		8	○		
（S1）+19、（S1）+18	K88		9			

• 根据上述示例的搜索结果表如下所示。

软元件编号	内容	搜索结果项目
（d）+1、（d）	3	相同数据的个数
（d）+3、（d）+2	0	相同数据的位置（首次）
（d）+5、（d）+4	6	相同数据的位置（最终）
（d）+7、（d）+6	5	最小值的最终位置
（d）+9、（d）+8	8	最大值的最终位置

注意事项

• 进行代数大小比较。（-10<2）

• 数据中的最小值、最大值有多个时，存储各自后侧的位置。

• 如果本指令驱动，搜索结果（d）占用[（d）+1、（d）]、[（d）+3、（d）+2、]、[（d）+5、（d）+4]、[（d）+7、（d）+6]、[（d）+9、（d）+8]的 5 点。请注意不要与用于机械控制的软元件重复。

错误码

错误码	内容
4084H	应用指令中(n)输入了超出可指定范围的数据。[1,128]
4085H	读应用指令(s1)、(s2)、(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

参考功能说明中的示例。

ALT/位软元件输出取反

ALT(P)

如果输入变为 ON，对位软元件进行取反（ON→OFF）的指令。

-[ALT （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	交替输出的软元件编号	-	位	ANY16_BOOL	BOOL

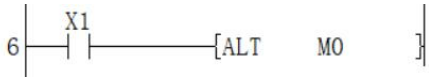
使用的软元件

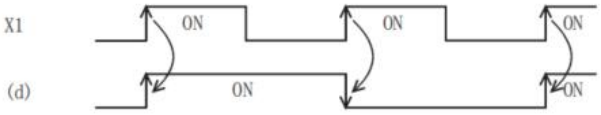
指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ALT	参数 1		●	●	●	●					●															●	●

功能

• 交替输出（1 级）

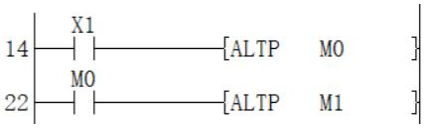
指令输入每次由 OFF→ON 变化时，（d）中指定的位软元件 OFF→ON 取反。

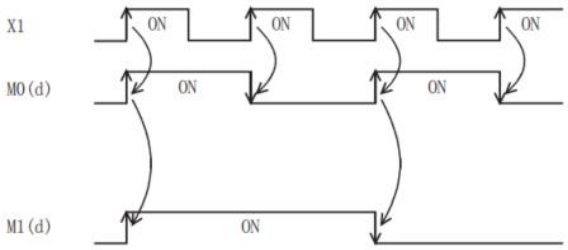




• 分频输出（通过交替输出（2 级））

将多个 ALTP 指令组合使用，可进行分频输出。





注意事项

•如果通过 ALT 指令编程，将在每个运算周期进行动作取反。通过指令 ON→OFF 使动作取反时，应使用 ALT 指令（脉冲执行型）或将指令触点设为 LDP 等（脉冲执行型）。

错误码

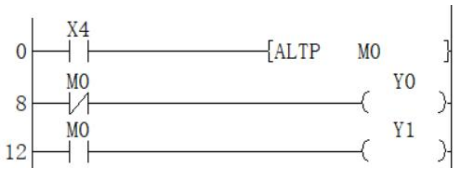
错误码	内容
4085H	读应用指令(d)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

1.通过一个输入启动/停止。

⑴按下按键 X4 后，启动输出 Y1 的动作，停止 Y0 的动作。

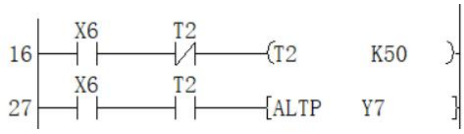
⑵再次按下按键 X4 后，停止输出 Y1 的动作，启动 Y0 的动作。

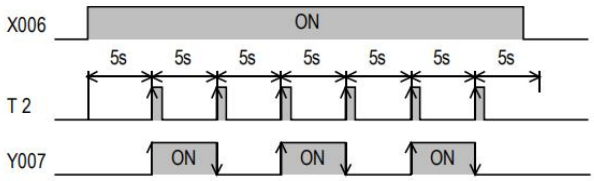


2.闪烁动作

⑴输入 X6 为 ON 时，定时器 T2 的触点每隔 5 秒瞬间动作一次。

⑵T2 的触点，每次 ON 时都使输出 Y7 交替 ON/OFF。





INCD/BIN16 位数据相对方式

INCD

使用一对计数器创建多种输出模式。

-[INCD（s1） （s2） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储设置值的起始软元件编号	-	有符号 BIN16 位	ANY16	--
(s2)	当前值监控用计数器的起始编号	-	有符号 BIN16 位	ANY16	--
(d)	输出的起始位软元件编号	-	位	ANY16_BOOL	--
(n)	输出的位软元件的点数	[1,64]	有符号 BIN16 位	ANY16	INT

使用的软元件

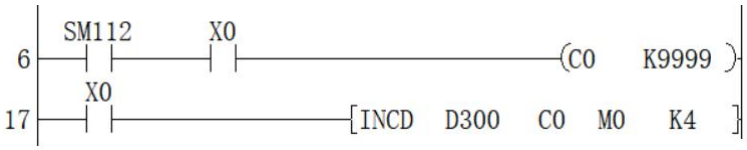
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
INCD	参数 1											●	●	●	●	●	●	●	●	●						●	
	参数 2																●									●	
	参数 3		●	●	●	●					●															●	
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	

功能

• 将（s1）开始的（n）行的数据表（（n）行×占用 2 点）与计数器当前值（s2）进行比较，如果一致则进行复位，并依次对输出进行 ON/OFF 控制。

动作示例

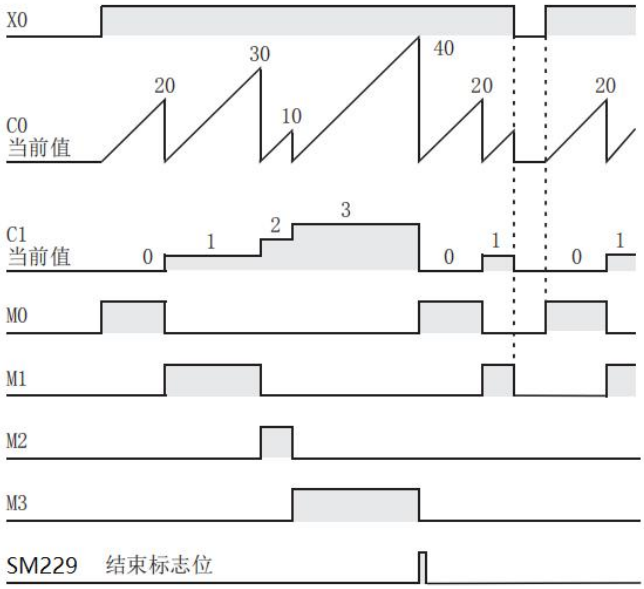
• 通过以下的回路示例对动作进行说明。（s2）占用 2 点。在下述时序图中 C0 和 C1 与此相当。



• 假设预先使用传送指令写入如下所示的数据。

储存软元件		输出	
-	数据值（示例）	-	示例
（S1）	D300=20	（d）	M0
（S1）+1	D301=30	（d）+1	M1
（S1）+2	D302=10	（d）+2	M2
（S1）+3	D303=40	（d）+3	M3
...	....	...	...
（S1）+（n）-1	-	（d）+n-1	-

• 时序图



• 如果指令触点变为 ON，则 M0 输出变为 ON。

• 输出（M0）在 C0 的当前值到达比较值 D300 时复位，工序计数器 C1 的计数值+1，计数器 C0 的当前值也复位。

• 下一个输出 M1 变为 ON。

• 将 C0 当前值与比较值 D301 进行比较，到达该比较值时输出 M1 复位，工序计数器 C1 的计数值+1，计数器 C0 的当前值也复位。

• 同样比较到（n）中指定的点（K4）为止。（1≤（n）≤64）

• (n)中指定的最终工序完成后，执行结束标志 SM229 变为 1 运算周期 ON。SM229 为多个指令中使用的指令执行结束标志，因此应在指令之后作为触点使用，执行该指令专用的结束标志。

• 返回开头重复输出。

注意事项

•在(s1)中通过位软元件的位数指定进行软元件编号指定的情况下，软元件编号应指定 16 的倍数(0、16、32、64……)。

• 程序中最多可同时驱动 4 条 INCD 指令。

错误码

错误码	内容
4084H	(n)中指定的数值超出下述范围。[1,64]
4085H	读应用指令(s1)、(s2)、(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(s2)、(d)中指定的软元件超出相应软元件的范围时。
4089H	指令驱动条数超出限制。

示例

参考功能说明中的示例。

RAMP/控制斜坡信号

RAMP

获得以指定的（n）次在开始（初始值）和结束（目标值）2 个值之间变化的数据。

-[RAMP（s1）（s2）（d）（n）]

内容、范围、数据类型

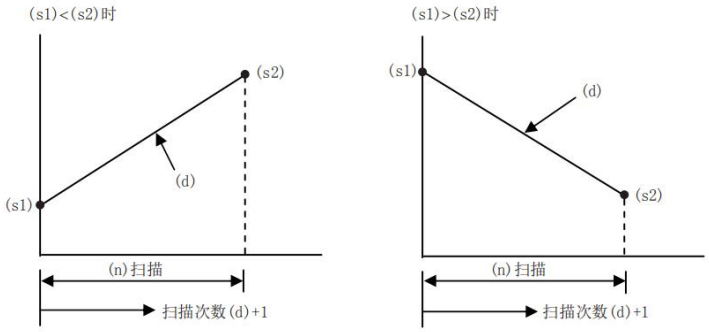
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	保存设定的斜坡初始值的软元件编号	-	有符号 BIN16 位	ANY16	INT
(s2)	保存设定的斜坡目标值的软元件编号	-	有符号 BIN16 位	ANY16	INT
(d)	保存斜坡的当前值数据的软元件编号	-	有符号 BIN16 位	ANY16	INT
(n)	斜坡的转移时间（扫描周期）	[1,32767]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RAMP	参数 1											●	●	●	●	●	●	●	●	●						●	
	参数 2											●	●	●	●	●	●	●	●	●						●	
	参数 3															●	●	●	●	●						●	
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	

功能

• 预先指定开始值（s1）和要结束的值（s2），如果指令输入置为 ON，每个运算周期将依次把以（n）中指定的次数等分后的值加到（s1）后的值存储到（d）中。可将该指令和模拟输出组合后输出柔和启动/停止指令。

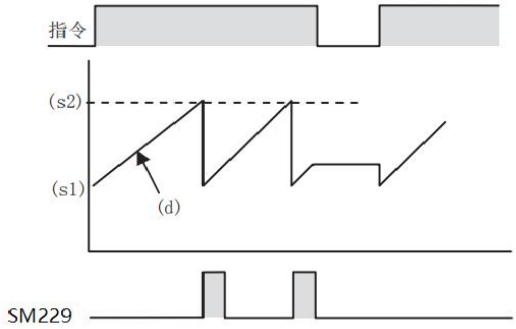


• （d）+1 中存储扫描次数（0→n 次）。

• 从开始到结束值为止的时间需要运行周期×（n）扫描。

• 动作过程中如果输入指令置为 OFF，将变为执行中断状态（（d）:当前值数据保持。（d）+1 扫描次数清除），如果再次置为 ON，（d）被清除，从（s1）重新开始动作。

• 过渡完成后，指令执行完成标志 SM229 动作，（d）的值恢复为（s1）的值。

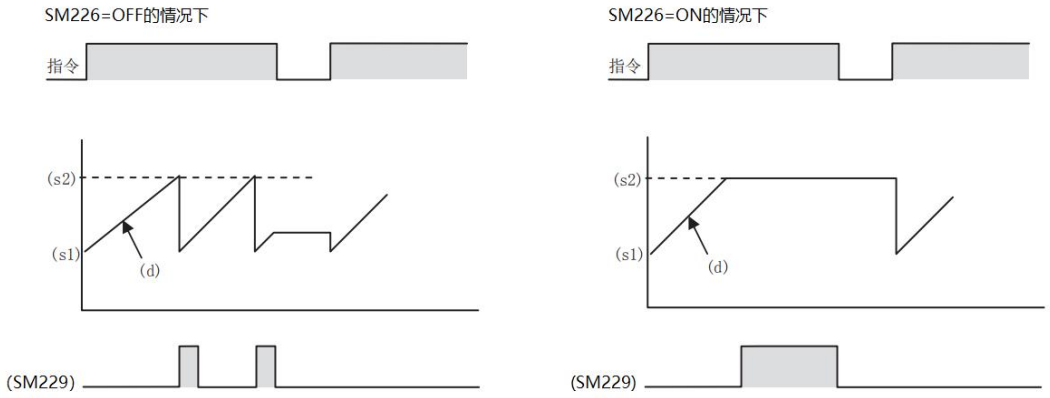


• 以一定的时间间隔获取运算结果的情况（恒定扫描模式）下，将所规定的扫描时间写入 SD120（比实际的扫描时

福州富昌维控电子科技有限公司

间稍长的值），将 SM120 置为 ON。例如将该值指定为 20ms、n=100 次的情况下，在 2 秒（d）的值从（s1）变为（s2）。

- 恒定扫描模式的值也可通过工程工具的参数设置（CPU 参数的恒定扫描执行间隔设置）进行设置。
- 根据模式标志 SM226 的 ON/OFF 动作，（d）的内容如下所示进行更改。



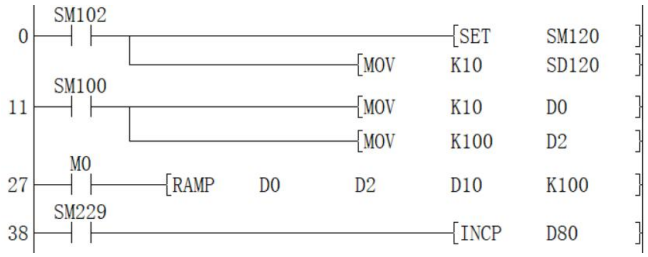
注意事项

- （d）中指定停电保持软元件（保持领域）的情况下，指令输入保持为 ON 不变，将 CPU 模块置为 RUN（开始）时，应预先清除（d）。

错误码

错误码	内容
4084H	应用指令中(n)输入了超出可指定范围的数据。[1,32767]
4085H	读应用指令(s1)、(s2)、(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



如上程序，将 SM120 置 ON，程序以恒定扫描周期（SD120 内的值 10ms）运行。M0=ON 时，在 100×10ms 时间内从 10 变化到 100。

ROTC/旋转台的就近控制

ROTC

为了将旋转台上的物品取出放入，根据要求取出放入的窗口，使旋转台就近旋转。

-[ROTC （s） （n1） （n2） （d）]

内容、范围、数据类型

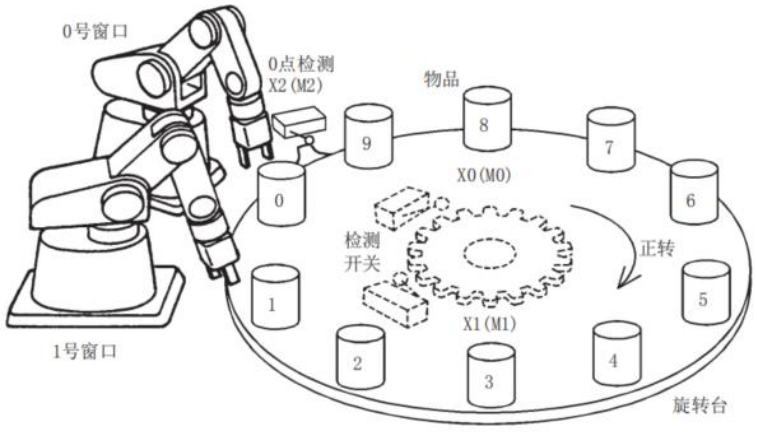
参数	内容		范围	数据类型	数据类型(标签)	自定义变量类型
(s)	调用条件的指定寄存器（根据传送指令预先设置）	(s)+0：计数用寄存器	-	有符号 BIN16 位	ANY16	--
		(s)+1：调用窗口编号的设置				
		(s)+2：调用物品编号的设置				
(n1)	分割数		[2, 32767]	有符号 BIN16 位	ANY16	INT
(n2)	低速区单数		[0, 32767]	有符号 BIN16 位	ANY16	INT
(d)	调用条件的指定位（构成预先从输入信号（X）驱动的内部触点回路）	(d)：A 相信号	-	有符号 BIN16 位	ANY16	--
		(d)+1：B 相信号				
		(d)+2：0 点检测信号				
		(d)+3：高速正转				
		(d)+4：低速正转				
		(d)+5：停止				
		(d)+6：低速反转				
		(d)+7：高速反转				

使用的软元件

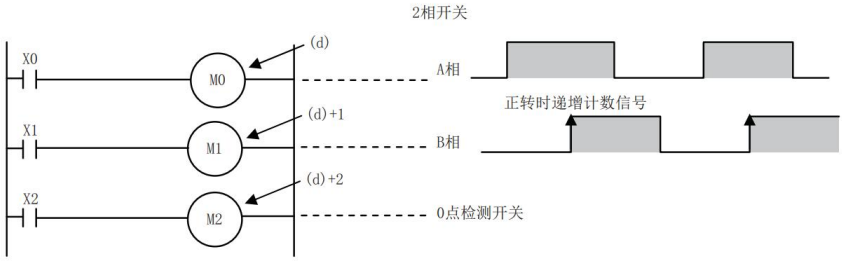
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ROTC	参数 1																	●	●	●						●	
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 4		●	●	●	●					●															●	

功能

为了将如下图所示 n1（=10）分割后的旋转台上的物品取出放入，根据要求取出放入的窗口，在 n2 或（s）、（d）的条件下使旋转台就近旋转。如果指定以下的动作条件，（d）+3~（d）+7 中可获得正/反转、高速/低速/停止等的输出。



• 应设置用于检测旋转台正转/反转的 2 相形状（X0、X1）和 0 号窗口时动作的开关 X2。在 X0~X2 替换为（d）~（d）+2 的内部触点。X 或（d）中指定的起始软元件编号可为任意。



- （s）是计数器，对多少号的物品来到 0 号窗口进行计数。
- （s）+1 中设置要调用窗口的编号。
- （s）+2 中设置调用物品的编号。
- 指定旋转台的分割数（n1）及低速运行区间（n2）。

注意事项

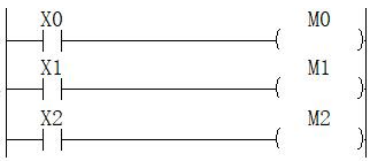
- 如果指令输入置为 ON 驱动该指令，将自动获得（d）+3~（d）+7 的结果。如果指令输入置为 OFF，（d）+3~（d）+7 将变为 OFF。
- 作为示例，旋转检测信号（（d）~（d）+2）置为在 1 分割区间内 10 次动作的情况下，分割数设置、调用窗口编号的设置、物品编号的设置均应为 10 倍的值。由此，低速区间的设置值可设为分割数的中间值等。
- 指令输入为 ON，0 点检测信号（M2）置为 ON 时，计数用寄存器（s）的内容清除为 0.需在预先进行此清除操作之后，再开始运行。
- ROTC 指令最多可驱动 4 条。

错误码

错误码	内容
4084H	(n1)中指定的数值超出下述范围时。[2,32767]
	(n2)中指定的数值超出下述范围时。[0,32767]
	(n1)、(n2)中指定的数值为下述条件时。(n1)<(n2)
	(s)、(s)+1、(s)+2 中某个为负时。
	(s)、(s)+1、(s)+2 中某个为(n1)以上时。
4085H	读应用指令(s)、(n1)、(n2)、(d)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(s)、(d)中指定的软元件超出相应软元件的范围时。
4089H	应用指令超出限制使用条数。

示例

示例说明：

变量	功能	操作说明	变量	功能	操作说明
D200	作为计数寄存器使用	由用户程序事先设定好该 3 个单元	M3	高速正转	当 X0 为 ON 时。可以自动得到 M3~M7 的结果。 当 X0 为 OFF 时，M3~M7 均为 OFF。
D201	调用窗口号码设定		M4	低速正转	
D202	调用工件号码设定		M5	停止	
M0	A 相信号	用户程序每次扫描本语句前执行： 	M6	低速反转	
M1	B 相信号		M7	高反转	
M2	0 点检测信号				

STMR/特殊功能定时器

STMR

使用（d）中指定的软元件开始的 4 点，进行 4 种类型的定时器输出。

-[STMR （s1）（s2）（d）]

内容、范围、数据类型

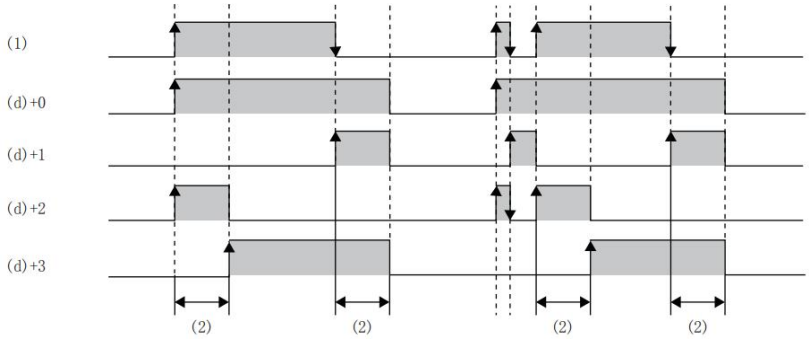
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	使用的定时器编号[T0~T511（100ms 定时器）]	-	软元件名	ANY16	--
(s2)	定时器的设置值	[1,32767]	有符号 BIN16 位	ANY16	INT
(d)	输出的起始位编号（占用 4 点）	-	位	ANYBIT_ARRAY	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
STMR	参数 1															●										●	
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 3		●	●	●	●					●															●	

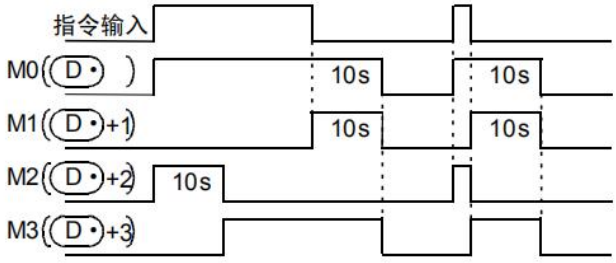
功能

• 使用（d）中指定的软元件开始的 4 点，进行 4 种类型的定时器输出。

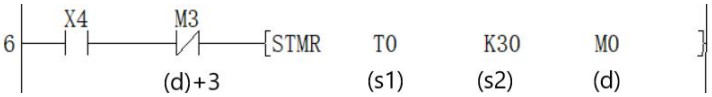


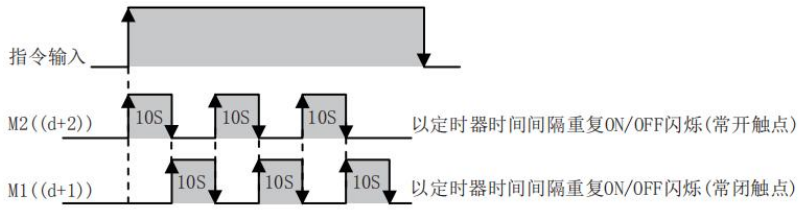
（1）：STMR 指令的指令

（2）：（S2）中指定的设置值



•闪烁将在（d）+3 的常闭触点通过将 STMR 指令置为 ON/OFF 的下述程序（在（s1）中分配 T10，（s2）中分配 K100，（d）中分配 M0）输出到（d）+1、（d）+2 中。





• （S2）的设置值可在 1~32767（1~3276.7 秒）的范围内指定。

注意事项

• 通过该指令指定的定时器编号无法与其他的一般回路（OUT 指令等）重复使用。重复的情况下不能正确执行定时器的动作。

• （s1）中指定的定时器作为 100ms 定时器，从指令触点的上升沿开始计时。

• 在起始处占用 4 点（d）中指定的软元件。请注意不要与用于机械控制的软元件重复。

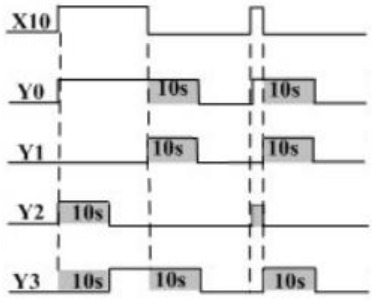
• 指令触点置为 OFF 的情况下，（d）、（d）+1、（d）+3 在设置时间后变为 OFF。（d）+2 和定时器（s1）立即复位。

错误码

错误码	内容
4084H	(s2)中指定的数值，(s2)≤0 时 。
4085H	读应用指令(s2)、(d)中指定的软元件超出相应软元件的范围时。

示例

1. 示例



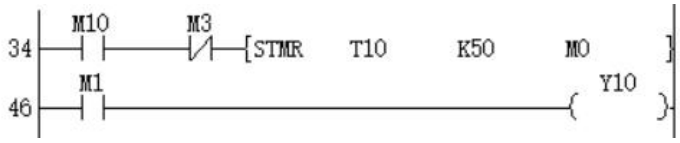
• Y0：当 X10 由 Off→On 时，Y0=On，
到 X10 由 On→Off 时 ，延迟 10 秒后 Y0=Off。

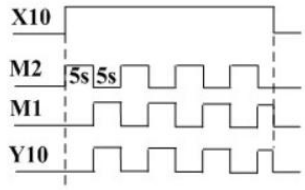
• Y1：当 X10 由 On→Off 时，作一次 10 秒 Y1=On 输出。

• Y2：当 X10 由 Off 变 On 时，作一次 10 秒 Y2=On 输出。

• Y3：当 X10 由 Off 变 On 时，延迟 10 秒后 Y3=On，
到 X10 由 On 变 Off 时，延迟 10 秒后 Y3=Off。

2. 若在指令流中引入（d）+3 的元件，可方便的实现振荡器输出（此功能也可以用 ALT 指令来实现），如下图：





TTMR/示教定时器

TTMR

测试 TTMR 指令为 ON 的时间。通过按钮调整定时器的设置时间的情况下使用。

-[TTMR （d） （s）]

内容、范围、数据类型

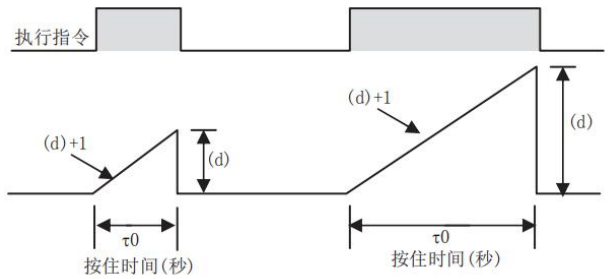
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储示教数据的软元件	-	有符号 BIN16 位	ANY16	--
(s)	示教数据的相乘倍率	[0,2]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TTMR	参数 1															●	●	●	●	●						●	
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	

功能

对执行指令（按钮）的按住时间以秒单位进行测定，将其乘以（s）中指定的倍率（10^S）后存储到（d）中指定的软元件中。



• 关于（d）中存储的时间，按住时间为$\tau 0$（单位：秒)时，根据（s）中指定的倍率，实际的（d）值如下所示。

(S)	倍率	(d)
K0	$\tau 0$	(d) ×1
K1	$10\tau 0$	(d) ×10
K2	$100\tau 0$	(d) ×100

(s)	(d)	(d)+1(单位：100 毫秒)
K0(单位：秒)	$1\times\tau 0$	[(d)+1] =[(d)]×10
K1(单位：100 毫秒)	$10\times\tau 0$	[(d)+1] =[(d)]
K2(单位：10 毫秒)	$100\times\tau 0$	[(d)+1] =[(d)]/10

注意事项

• 如果指令触点由 ON→OFF，按住时间的当前值（d）+1 被清除，示教时间（d）不变化。

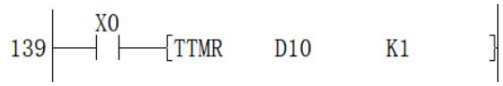
• 在起始处占用 2 点示教时间（d）中指定的软元件。请注意不要与用于机械控制的软元件重复。

错误码

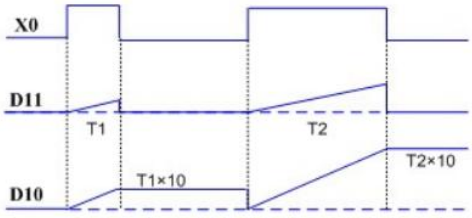
错误码	内容
4084H	(n)中指定的数值超出下述范围时。[0,2]
4085H	读应用指令(d)、(s)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

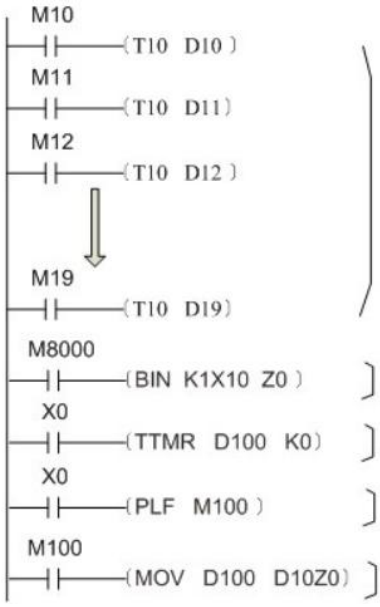
示例 1



在 X0 闭合期间，D10=D11；当 X0 断开后，D10 值保持不变，而 D11 则变为 0。



示例 2



使用 TTMR 指令写入 10 组设置时间，将设置值预先写入 D10~D19，改组定时器位 100ms 型定时器，因此示教数据的 1/10 为实际动作时间（秒）

将 1 位数指拨开关接于 X10~X13，使用 BIN 指令指拨开关的设置值转换成 BIN 值并存放于 Z0 当中

X0 为 On 将时间（秒）存放于 D100 当中

M100 为演示定时器按钮 X0 释放产生的一次扫描周期脉冲

以指拨开关的设置号码当成间接指定的指针，然后将 D100 的内容传送至 D10Z0（D10~D19）当中

7.7 外部 IO 指令

ARWS/箭头开关

ARWS

通过使用位数移动和增减各位数值用的箭头开关，输入数据的指令。

-[ARWS（s）（d1）（d2）（n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	输入的起始位软元件编号	-	BIN16 位	ANY_BOOL	--
(d1)	保存 BCD 换算数据的字软元件编号	-	BIN16 位	ANY_BOOL	INT
(d2)	连接 7 段数码管显示的起始位软元件(Y)	[0,9999]	BIN16 位	ANY16_S	--
(n)	7 段数码管显示的位数指定[设定范围：K0~K3]	[0,3]	BIN16 位	ANY16_S	INT

使用的软元件

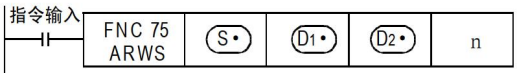
指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ARWS	参数 1	●	●	●	●	●					●															●	
	参数 2															●	●	●	●	●						●	
	参数 3		●																							●	
	参数 4																						●	●		●	

功能

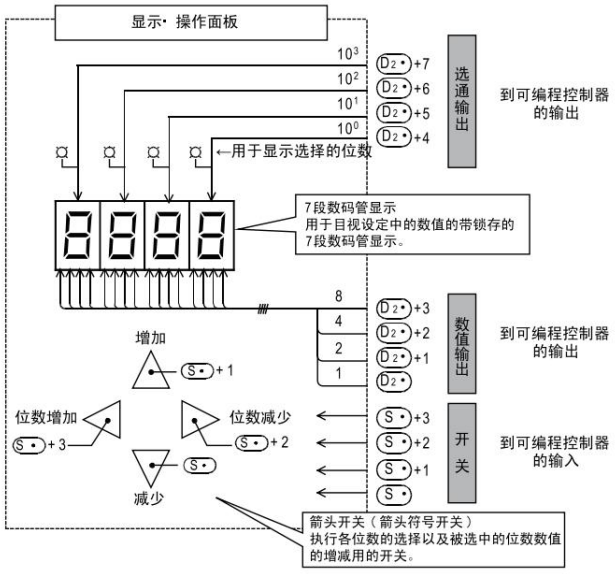
16 位运算（ARWS）

在 D+1 中保存 0~9999 的 16 位 BIN 值，为了方便起见，以下的说明中以 BCD 转换显示。

指令输入为 ON 时，ARWS 指令会如下所示动作：



显示及操作部分的内容



1)带 BCD 译码的 7 段数码管显示的位数指定 n

在下面的动作说明中，以 4 位数(10^3 位数)为例加以说明。

2)位数选择开关的动作(S+2、S+3)

- 位数减少的输入 S+2 为 ON 时的动作，每次按下开关时，位数指定按照 103→102→101→100→103 改变。

• 位数增加的输入 S+3 为 ON 时的动作，每次按下开关时，位数指定按照 103→100→101→102→103 改变。

3)显示选择位数用的 LED 的动作(D2+4~D2+7)，可以通过选通信号 D2+4~D2+7,用 LED 显示指定的位数。

4)以位数为单位的数据变更开关的动作(S、S+1)，针对上述的“位数选择开关”指定的位数，使其数据发生变化。

• S+1 为数据增加开关，增加输入为 ON 时的动作，每次按下开关时，D1 的内容按照 0→1→2→...→8→9→0→1 改变。

• S 为数据减小开关，减少输入为 ON 时的动作，每次按下开关时，D1 的内容按照 0→9→8→7...1→0→9 改变。

这些内容可以在 7 段数码管显示中显示。

如上所示，通过一连串的操作，在查看 7 段数码管显示的同时，可以将目标数值写入 D1 中。

注意事项

1. 参数 n 的设定

请参考 SEGL(FNC 74)指令的参数设定。但是，设定范围为 0~3。

2. 关于可编程控制器的输出形式

请使用晶体管输出型的可编程控制器。

3. 关于扫描时间(运算周期)和显示时序

ARWS 指令与可编程控制器的扫描时间(运算周期)同步执行。

为了执行一连串的显示，可编程控制器的扫描时间需要超出 10ms。

不满 10ms 时，请使用恒定扫描模式，在 10ms 以上的扫描时间下运行。

4. 软元件的占用点数

①软元件 S 的输入占用 4 点。

②软元件 D2 的输出占用 8 点。

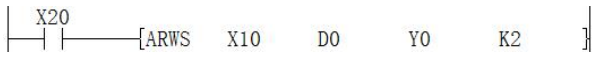
5. 指令的使用次数的限制

ARWS 指令，在程序只可以使用 1 个。

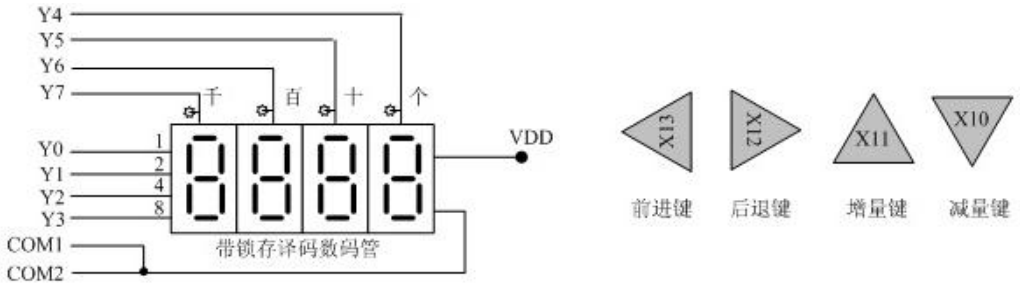
错误码

错误码	内容
4084H	应用指令中(d1)(n)输入了超出可指定范围的数据。
4085H	读应用指令中(s)(d1)(d2)输出结果超出软元件范围。
4086H	写应用指令中(d1)(d2)输出结果超出软元件范围。

示例



对应的硬件接线如下图所示，PLC 应为晶体管输出型：



(1)图中数码管显示 D0 的数值，按下 X10~X13 可修改其中的数值，D0 的数值只能在 0~9999 之间。

(2)当 X20 为 ON 时，光标位为千位。每次按后退键（X12）时，指定位按“千→百→十→个→千”的顺序切换；若按前进键（X13），则切换顺序相反；光标位由选通脉冲信号（Y004~Y007）连接的 LED 指示。

(3)对于光标位，每次按增量键（X11）该位的内容按 0 → 1→ 2→……8→9→0→1 变化，按减量键（X10）时，则按 0→9→8→7→…… 1→0→9 变化，修改的值立即生效。

DSW/数字键输入

DSW

该指令是读取矩阵型设置开关的状态，以 4 个 BCD 设定开关为 1 组，将设定值读取后存放到指定单元，最多可读取 2 组设定开关。

-[DSW （s） （d1） （d2） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	连接数字开关的起始软元件(X)编号(占用 4 点)	-	位	ANY_BOOL	--
(d1)	输出选通信号的起始软元件(Y)编号(占用 4 点)	-	位	ANY_BOOL	--
(d2)	保存数字开关的数值的软元件编号(占用 n 点)	[0,9999]	有符号 BIN16	ANY16_S	INT
(n)	数字开关的组数(4 位数/1 组)[n=1 或是 2]	[1,2]	有符号 BIN16	ANY16_S	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSW	参数 1	●																								●	
	参数 2		●																							●	
	参数 3															●	●	●	●	●						●	
	参数 4																						●	●		●	

功能

该指令是读取矩阵型设置开关的状态，以 4 个 BCD 设定开关为 1 组，将设定值读取后存放到指定单元，最多可读取 2 组设定开关。

1) 有关输入的数值(d1)

— 可以读取 0~9,999 的 4 位数。

— 以 BIN(2 进制数)值保存数据。

— 第 1 组保存到(d2)中，第 2 组保存到(d2)+1 中。

2) 有关组数 n 的指定

—使用 4 位数/1 组×1 时[n=K1]通过选通信号

(s)~[(s)+3]，依次读取(d1)~[(d1)+3]中连接的 BCD4 位数的数字开关，并且将其值作为 BIN 值保存到(d2)中。

—使用 4 位数/1 组×2 时[n=K2]通过选通信号

(s)~[(s)+3]，依次读取(d1)~[(d1)+3]中连接的 BCD4 位数的数字开关，并且将其值作为 BIN 值保存到(d2)中。

通过选通信号(d1)~[(d1)+3]，依次读取(s)+4~[(s)+7]中连接的 BCD4 位数的数字开关，并且将其值作为 BIN 值保存到(d2)+1 中。

注意事项

1. 指令触点为 OFF 时

即使为 OFF，(d2)的内容也不改变，但是(d1)~[(d1)+3]为止都变为 OFF。

2. 软元件的占用点数

1) 使用 4 位数 2 组(n=K2)时，占用从(d2)开始的 2 点。

2) 当为 4 位数 1 组时(s)被占用 4 点，当为 4 位数 2 组时被占用 8 点。

3. 连接不满 4 位数的数字开关时

对于没有使用的位数，选通信号＜指定位数用的输出＞(d1)无需接线，但是即使有未使用的位数，其输出也已经被这个指令占用了，所以也不能用于其他用途。请务必将不使用的输出空出。

4. 推荐使用晶体管输出型

为了能够连续地读取数字开关的值，请务必使用晶体管输出型的可编程控制器。

5. 有关数字开关

请使用 BCD 输出型的数字开关。

6. 关于键盘输入的读取时序

为了防止因键盘输入的滤波器延迟而导致的读取遗漏，请灵活使用[恒定扫描模式]和[定时器中断]功能。

7. 指令限制条数

该指令最多同时使用 2 条。

相关软元件

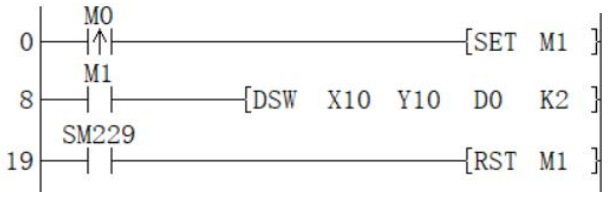
软元件	名称	内容
SM229	指令执行结束	一次读取循环结束，SM229 会置位一个扫描周期。

错误码

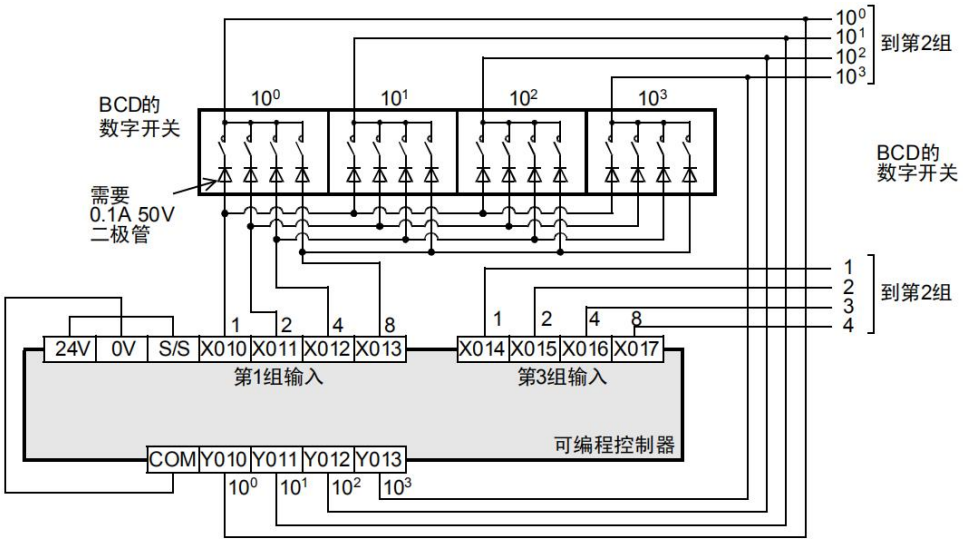
错误码	内容
4084H	应用指令中(n)(d2)输入了超出可指定范围的数据。
4085H	读应用指令中(s)(d2)输出结果超出软元件范围。
4086H	写应用指令中(d1)(d2)输出结果超出软元件范围。
4089H	应用指令超出限制使用条数。

示例

1. 程序



2. 接线图



1) M1(数字开关读取输入)为 ON 期间，DSW 动作。

2) DSW 在其结束 1 个循环的动作，指令执行结束标志位(SM229)变为 ON 之前，一直动作。

HKY/16 进制数字键输入

HKY

通过 0~F 的键盘(16 键)输入，设定数值(0~9)及运行条件(A~F 功能键)等的输入数据用的指令。

当扩展功能为 ON 时，可以使用 0~F 键的 16 进制数进行键盘的输入。

-[HKY（s）（d1）（d2）（d3）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	输入 16 键的起始位软元件(X)编号(占用 4 点)	-	位	ANY_BOOL	--
(d1)	输出的起始软元件(Y)编号(占用 4 点)	-	位	ANY_BOOL	--
(d2)	保存从 16 键输入的数值的软元件编号	[0,9999]	BIN16 位	ANY16_S	INT
(d3)	按键信息为 ON 的起始位软元件编号(占用 8 点)	-	BIN16 位	ANY16_S	--

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
HKY	参数 1	●																								●	
	参数 2		●																							●	
	参数 3															●	●	●	●	●						●	
	参数 4		●	●	●	●					●															●	

功能

16 位运算(HKY)

扫描连接 16 键(0~F)的输入[S~ S+3]和列输出[D1~ D1+3]的信号，按下 0~9 按键，其数值保存在 D2 中，键盘检测输出到 D3+7 中。

此外，按下 A~F 键后，与键盘对应的按键信息[D3~ D3+5]为 ON，键盘检测输出到 D3+6。

1) 关于使用 0~9 的键输入数值 D3、 D3+7

— 若为 9,999 以上时，从高位数溢出。输入的数值，以 BIN(2 进制数)值保存在 D2 中。当 0~9 中任意一个键被按下时，键盘检测输出 D3+7 为 ON。

2) 关于 A~F 键的按键信息 D3~ D3+6

— 与 A~F 键相对应 D3 的开始 6 点为 ON。当 A~F 中任意一个键被按下时，键盘检测输出 D3+6 为 ON。

键盘	按键信息	键盘	按键信息	键盘	按键信息
A	D3	C	D3+2	E	D3+4
B	D3+1	D	D3+3	F	D3+5

扩展功能

SM167 为 ON，扩展功能变为有效后，以 BIN 方式保存 0~F 的 16 进制按键的数据。

除下面以外，都与上述的[功能及动作说明]相同。

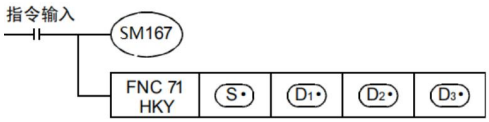
使用 0~F 键输入的 16 进制数据原封不动地写入到 D2 中。

1) 关于使用 0~F 键的数值输入 D2， 当为 FFFF 以上时，从高位数溢出。

例如

输入[1]→[2]→[3]→[B]→[F]时，以 BIN 方式将“23BF”保存到 D2 中。

当输入[F]时，[1]溢出。



注意事项

1. 指令的使用次数的限制

   HKY 指令，在程序中只可以使用其中 1 个。

2. 同时按下键盘时

   同时按下多个键时，先按下的键有效。

3. 指令触点为 OFF 时

   即使为 OFF， D2 的内容也不改变，但是 D3~D3 +7 为止都变为 OFF。

4. 软元件的占用点数

   ①连接 16 键时，占用从输入(X)的起始软元件 S 开始的 4 点。

   ② 连接 16 键时，占用从输出(Y)的起始软元件 D1 开始的 4 点。

   ③ 占用从按键信息输出用的起始软元件 D3 开始的 8 点。

   请注意不要与机器其他控制中使用的软元件重复。

   D3~ D3+5 ： A~F 键的按键信息。 D3+6 ： A~F 键的键盘检测输出。 D3+7 ： 0~9 键的键盘检测输出。

5. 关于键盘输入的读取时序

   HKY 指令与可编程控制器的运算周期同步执行。完成一系列的键盘扫描，需要 8 个运算周期的时间。

   为了防止因键盘输入的滤波器延迟而导致的读取遗漏，请灵活使用[恒定扫描模式]和[定时器中断]功能。

6. 输出形式

   请使用晶体管输出型的可编程控制器。

相关软元件

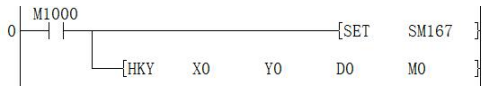
软元件	名称	内容
SM229	指令执行结束	OFF:(d1)~(d1)+3 的扫描中，或指令未执行 ON :(d1)~(d1)+3 循环输出一次的动作(1~4 位数的扫描)后置 ON

错误码

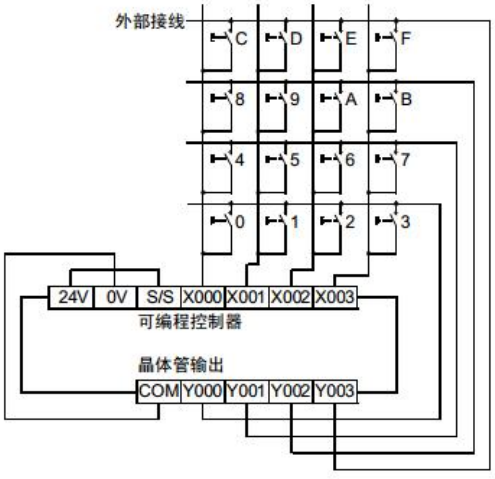
错误码	内容
4085H	读应用指令中(s)(d2)输出结果超出软元件范围
4086H	写应用指令中(d1)(d2)(d3)输出结果超出软元件范围

示例

1. 程序



2. 接线图



当输入输入[1]→[2]→[3]→[B]→[F]时，以 BIN 方式将“23BF”保存到 D0 中。

当输入[F]时，[1]溢出。

DHKY/32 进制数字键输入

DHKY

通过 0~F 的键盘(16 键)输入，设定数值(0~9)及运行条件(A~F 功能键)等的输入数据用的指令。

当扩展功能为 ON 时，可以使用 0~F 键的 32 进制数进行键盘的输入。

-[DHKY（s）（d1）（d2）（d3）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	输入 16 键的起始位软元件(X)编号(占用 4 点)	-	位	ANY_BOOL	--
(d1)	输出的起始软元件(Y)编号(占用 4 点)	-	位	ANY_BOOL	--
(d2)	保存从 16 键输入的数值的软元件编号	[0,99999999]	BIN32 位	ANY32_S	DINT
(d3)	按键信息为 ON 的起始位软元件编号(占用 8 点)	-	BIN16 位	ANY16_S	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DHKY	参数 1	●																							●	
	参数 2		●																						●	
	参数 3															●	●	●	●	●	●	●				●	
	参数 4		●	●	●	●					●															●	

功能

32 位运算(DHKY)

扫描连接 16 键(0~F)的输入[S~ S+3]和列输出[D1~ D1+3]的信号，按下 0~9 按键，其数值保 存在[D2+1， D2]中，键盘检测输出到 D3+7 中。

此外，按下 A~F 键后，与键盘对应的按键信息[D3~ D3+5]为 ON，键盘检测输出到 D3+6。

1) 关于使用 0~9 的键输入数值[D2 +1、 D2]、 D3+7

— 若为 99,999,999 以上时，从高位数溢出。

— 输入的数值，以 BIN(2 进制数)值保存在[D2 +1、 D2]中。

— 当 0~9 中任意一个键被按下时，键盘检测输出 D3+7 为 ON。

2) 关于 A~F 键的按键信息 D3 ~ D3+6

关于键盘按动信息，请参考“HKY16/进制数字键输入”指令。

扩展功能

SM167 为 ON，扩展功能变为有效后，以 BIN 方式保存 0~F 的 16 进制按键的数据。

除下面以外，都与上述的[功能及动作说明]相同。

使用 0~F 键输入的 16 进制数据原封不动地写入到[D2 +1、 D2]中。

1) 关于使用 0~F 键的数值输入[D2+1、 D2]

— 当为 FFFFFFFF 以上时，从高位数溢出。

例如

输入[9]→[2]→[3]→[B]→[F]→[A]→[F]时，以 BIN 方式将“923BFAF”保存到[D2+1、 D2]中。

注意事项

1. 指令的使用次数的限制

DHKY 指令，在程序中只可以使用其中 1 个。

2. 同时按下键盘时

同时按下多个键时，先按下的键有效。

3. 指令触点为 OFF 时

即使为 OFF， D2 的内容也不改变，但是 D3~D3 +7 为止都变为 OFF。

4. 软元件的占用点数

1) 连接 16 键时，占用从输入(X)的起始软元件 S 开始的 4 点。

2) 连接 16 键时，占用从输出(Y)的起始软元件 D1 开始的 4 点。

3) 占用从按键信息输出用的起始软元件 D3 开始的 8 点。

请注意不要与机器其他控制中使用的软元件重复。

— D3~D3+5 ： A~F 键的按键信息

— D3+6 ： A~F 键的键盘检测输出

— D3+7 ： 0~9 键的键盘检测输出

5. 关于键盘输入的读取时序

DHKY 指令与可编程控制器的运算周期同步执行。

完成一系列的键盘扫描，需要 8 个运算周期的时间。

为了防止因键盘输入的滤波器延迟而导致的读取遗漏，请灵活使用[恒定扫描模式]和[定时器中断]功能。

6. 输出形式

请使用晶体管输出型的可编程控制器。

相关软元件

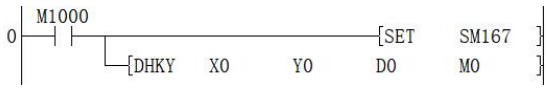
软元件	名称	内容
SM229	指令执行结束	OFF ： (d1)～(d1)+3 的扫描中，或指令未执行。 ON ： (d1)～(d1)+3 循环输出一次的动作(1~4 位数的扫描)后置 ON。

错误码

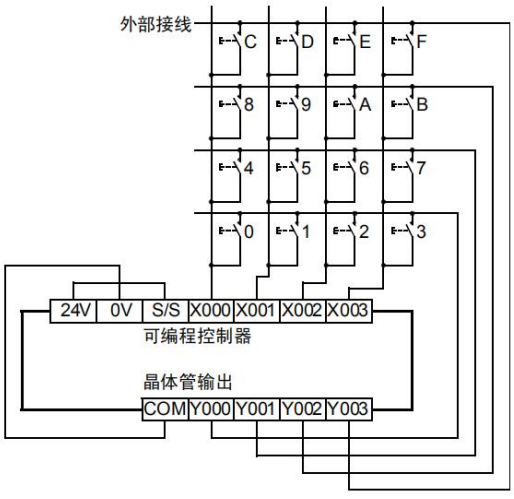
错误码	内容
4085H	读应用指令中(s)(d2)输出结果超出软元件范围。
4086H	写应用指令中(d1)(d2)(d3)输出结果超出软元件范围。

示例

1. 程序



2. 接线图



当输入输入[1]→[2]→[3]→[B]→[F]→[5]→[7]→[6]时，以 BIN 方式将“123BF576”保存到[D1,D0]中。

PR/ASCII 码打印

PR

该指令是将 ASCII 码的数据并行输出到输出(Y)。

-[PR（s） （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	保存 ASCII 码数据的软元件的起始编号		字符串(仅 ASCII 码)	ANY_ASC	--
(d)	输出 ASCII 码数据的起始 Y 编号	-	位	ANY_BOOL	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
PR	参数 1															●	●	●	●	●						●	
	参数 2		●																							●	

功能

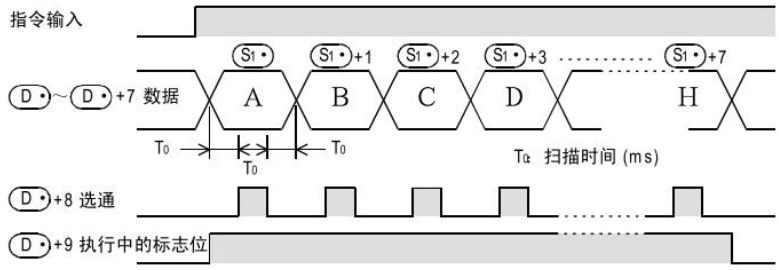
将(S)～(S)+7 的低 8 位（1 个字节）中保存的 ASCII 码，按时分方式逐个字符输出到（D）～(D)+7 中。

中保存的 ASCII 码如下所示，下述时序图即以此为例进行说明。

发送的顺序从（S）=“A”开始，到（S）+7 = “H”为此结束，发送八个字节。

(S·)	(S·)+1	(S·)+2	(S·)+3	(S·)+4	(S·)+5	(S·)+6	(S·)+7
A(H41)	B(H42)	C(H43)	D(H44)	E(H45)	F(H46)	G(H47)	H(H48)

时序图：



输出信号的种类

- (D·)～(D·)+7 ：发送输出　　(D·)为低位侧，(D·)+7为高位侧。
- (D·)+8 ：选通信号
- (D·)+9 ：执行中的标志位　按照上述的时序图动作。

相关软元件

软元件	名称	内容
SM227	PR 模式	OFF ： 8 个字节串行输出(固定为 8 个字符)。 ON ： 16 个字节串行输出(1~16 个字符)。

注意事项

1. 指令输入及指令的动作

   指令输入=ON ： 即使是执行连续为 ON 的指令或者执行脉冲指令，只要循环一次的输出结束，则执行就结束。

   仅当 SM227=ON 时 SM229 动作。

   指令输入=OFF ： 输出全部为 OFF。

2. 与扫描时间(运算时间)的关系

   该指令与扫描时间同步执行。

   扫描时间较短时，可以使用恒定扫描模式驱动；较长时可以使用定时器中断驱动。

福州富昌维控电子科技有限公司

3. 关于可编程控制器的输出

请使用晶体管输出型的可编程控制器。

4. 数据中存在 00H(NUL)时(SM227=ON 时)

指令执行结束，不输出剩余的数据。

此外，SM229 维持一次运算周期为 ON。

5. 指令的使用次数的限制

PR 指令，在程序只可以使用 1 个。

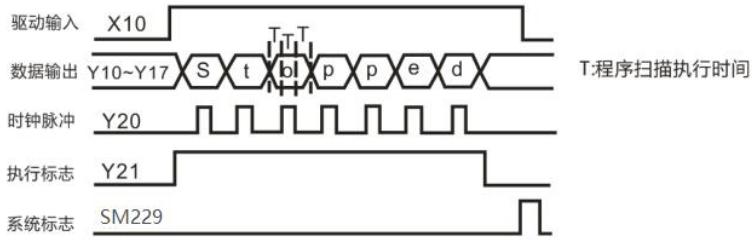
错误码

错误码	内容
4085H	读应用指令中(s)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

1. 程序

若 D200~D203 中的 ASCII 码为“Stopped”，则对应的输出端口信号及其时序如下图：



SEGD/数字键输入

SEGD(P)

点亮 7 段数码管(1 位数)的指令。

-[SEGD （s） （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	译码的起始字软元件	[-32767,+32767]	位	ANY_BOOL	INT
(d)	保存 7 段码显示用数据的字软元件编号	[-32767,+32767]	位	ANY_BOOL	INT

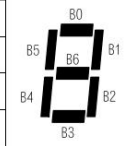
使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SEGD	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

将(S)的低 4 位(1 位数)的 0~F(16 位进制数)译码成 7 段码显示用的数据，并保存到(d)的低 8 位中。

1.7 段码译码表（位软元件的起始，或是字软元件的最低位为 B0）

(S·)					7段码的构成	(D·)											显示数据
16进制数	b3	b2	b1	b0		B15	…	B8	B7	B6	B5	B4	B3	B2	B1	B0	
0	0	0	0	0		—		—	0	0	1	1	1	1	1	1	0
1	0	0	0	1		—		—	0	0	0	0	0	1	1	0	1
2	0	0	1	0		—		—	0	1	0	1	1	0	1	1	2
3	0	0	1	1		—		—	0	1	0	0	1	1	1	1	3
4	0	1	0	0		—		—	0	1	1	0	0	1	1	0	4
5	0	1	0	1		—		—	0	1	1	0	1	1	0	1	5
6	0	1	1	0		—		—	0	1	1	1	1	1	0	1	6
7	0	1	1	1		—		—	0	0	1	0	0	1	1	1	7
8	1	0	0	0		—		—	0	1	1	1	1	1	1	1	8
9	1	0	0	1		—		—	0	1	1	0	1	1	1	1	9
A	1	0	1	0		—		—	0	1	1	1	0	1	1	1	A
B	1	0	1	1		—		—	0	1	1	1	1	1	0	0	b
C	1	1	0	0		—		—	0	0	1	1	1	0	0	1	C
D	1	1	0	1		—		—	0	1	0	1	1	1	1	0	d
E	1	1	1	0		—		—	0	1	1	1	1	0	0	1	E
F	1	1	1	1		—		—	0	1	1	1	0	0	0	1	F

注意事项

软元件的占用点数：软元件(S)的输出开始的低 8 位被占用，高 8 位不变化。

错误码

错误码	内容
4085H	读应用指令中(s)(d)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

M0 置位时，将 D0 内数据低 4 位译码后，输出到 Y10~Y17 端口。翻译用的对应表如上表（7 段码译码表）。该表格不需用户准备，PLC 系统内部已具备该对照表。

SEGL/7SEG 码时分显示

SEGL

控制 1 组或是 2 组 4 位数带锁存的 7 段数码管显示的指令。

-[SEGL（s）（d）（n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	BCD 转换的起始字软元件	[0,9999]	BIN16 位	ANY16	INT
(d)	被输出的起始 Y 编号	-	位	ANY_BOOL	--
(n)	参数编号[设定范围：K0(H0)~K7(H7)]	[0,7]	BIN16/32 位	ANY16_U	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SEGL	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 2		●																							●	
	参数 3																						●	●			

功能

将(s)的 4 位数值转换成 BCD 数据，采用时分方式，依次将每 1 位数输出到带 BCD 译码的 7 段数码管中。(s)为 0~9999 范围的 BIN 数 据时有效。参数（n）应根据可编程控制器侧的正负逻辑和 7 段侧的正负逻辑进行如下设置。

可编程控制器输出逻辑	数据输入	选通信号	参数 n	
			4 位数 1 组时	4 位数 2 组时
负逻辑	负逻辑（一致）	负逻辑（一致）	0	4
		正逻辑（不一致）	1	5
	正逻辑（不一致）	负逻辑（一致）	2	6
		正逻辑（不一致）	3	7
正逻辑	正逻辑（一致）	负逻辑（一致）	0	4
		正逻辑（不一致）	1	5
	负逻辑（不一致）	负逻辑（一致）	2	6
		正逻辑（不一致）	3	7

1. 使用 4 位数 1 组时（n=K0~3）

将（s）的 4 位数值从 BIN→BCD 转换后，采用时分方式，从（d）~（d）+3 依次将每一位数输出。此外，选通信号输出（d）+4~（d）+7 也依次以时分方式输出，锁定为 4 位数第 1 组的 7 端显示

2. 使用 4 位数 2 组时(n=K4~7)

①4 位数第 1 组：将(s)的 4 位数值从 BIN→BCD 转换后，采用时分方式，从(d)~(d)+3 依次将每一位数输出。选通信号输出(d)+4~(d)+7 依次以时分方式输出，锁定为 4 位数第 1 组的 7 段显示。

②4 位数第 2 组：将(s)+1 的 4 位数值从 BIN +BCD 转换后，采用时分方式，从(d)+10~(d)+13 依次将每一位数输出。选通信号输出(d)+4~(d)+7 依次以时分方式输出，锁定为位数第 2 组的 7 段显示。

注意事项

1. 关于更新 7 段码 4 位数的显示所需的时间

更新 4 位数(1 组或 2 组)的显示所需的时间为扫描时间(运算时间)的 12 倍。

2. 指令输入为 OFF 时的动作 ：当指令输入为 ON 时，重复执行动作，但是如在一动作过程中，指令触点变为 OFF，则动作会中断，再次为 ON 时从最初动作开始。

3. 软元件的占用点数

使用 4 位数 1 组时：占用 S 中指定的起始软元件开始的 1 点。

占用 D 中指定的起始软元件开始的 8 点。即使位数少时，占用的点也不能用于其他用途。

使用 4 位数 2 组时：占用 S 中指定的起始软元件开始的 2 点。

占用 D 中指定的起始软元件开始的 12 点。即使位数少时，占用的点也不能用于其他用途。

4. 关于扫描时间(运算周期)和显示时序

SEGL 指令与可编程控制器的扫描时间(运算周期)同步执行。为了执行一连串的显示，可编程控制器的扫描时间需要超出 10ms。不满 10ms 时，请使用恒定扫描模式，在 10ms 以上的扫描时间下运行。

5. 关于可编程控制器的输出形式

请使用晶体管输出型的可编程控制器。

6. 指令限制条数

该指令最多同时使用 2 条。

相关软元件

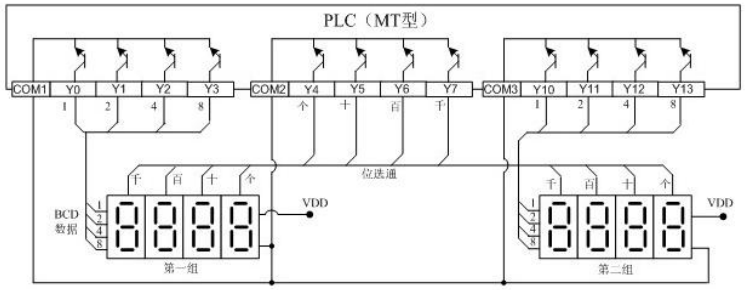
软元件	名称	内容
SM229	指令执行结束	处理完成后，SM229 为 ON 一个扫描周期。

错误码

错误码	内容
4084H	应用指令中(n)输入了超出可指定范围的数据。
4085H	读应用指令中(s)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。。
4089H	应用指令超出限制使用条数。

示例

对应的硬件接线如下图，在第一组数码管上显示 D0 的内容，在第二组数码管上显示 D1 的内容，若 D0 或 D1 的读数超过 9999，程序运行就会出错：



接线图中所使用的数码管，自带显示数据锁存、7 段译码和驱动、负逻辑型（输入端口为低电平时，表示输入的数据为 1，或被选通）的 7 段数码显示管。显示处理中，PLC 的 Y4~Y7 端口会自动循环扫描，每次只有 1 个端口为 ON，作为位选通信号，此时 Y0~Y3 口上的数据即为送给对应位的 BCD 码数据，当位选通信号由 ON→OFF 时，即被锁存到数码管内的锁存器中，经过内部的译码和驱动后，数码管将数字显示出来。PLC 系统依次将 Y4~Y7 循环作同样处理，直到将 4 位都处理完毕。同样的道理，Y10~Y13 是第二组 4 位数码管的数据输出端口，共用 Y4~Y7 的位选通线，处理方法相同，两组的显示处理是同时进行的。范例中若 D0=K2468，D1=K9753，则第一组将会显示 2 4 6 8，第二组显示 9 7 5 3。

完成一次显示刷新需要 12 个扫描周期，处理完成后：根据可编程控制器的正负逻辑、七段码的正负逻辑等因素，按以下原则选择：一组 4 位数的时候 n=0~3。二组 4 位数的时候 n= 4~7。

显示组数	1 组				2 组			
Y 数据输出极性	PNP		NPN		PNP		NPN	
选通与数据极性	相同	相反	相同	相反	相同	相反	相同	相反
n 的取值	0	1	2	3	4	5	6	7

TKY/16 进制数字键输入

TKY

通过 0~9 的键盘(数字键)输入，对定时器和计数器等设定数据的指令。

-[TKY （s） （d1） （d2）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	输入数字键的起始位软元件[占用 10 点]	-	位	ANY_BOOL	--
(d1)	保存数据的字软元件编号	[0,9999]	有符号 BIN16	ANY16_S	INT
(d2)	按键信息为 ON 的起始位软元件编号[占用 11 点]	-	位	ANY_BOOL	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TKY	参数 1	●	●	●	●	●					●															●	
	参数 2												●	●	●	●	●	●	●	●						●	
	参数 3		●	●	●	●					●															●	

功能

对连接的数字键输入[(s)～ +9]按动键盘，将输入的数值保存在(d1)中，在(d2)～ +10 中输出键盘输入信息以及检测到的键盘输出。

1) 有关输入的数值(d1)

— 若为 9,999 以上时，从高位数溢出。

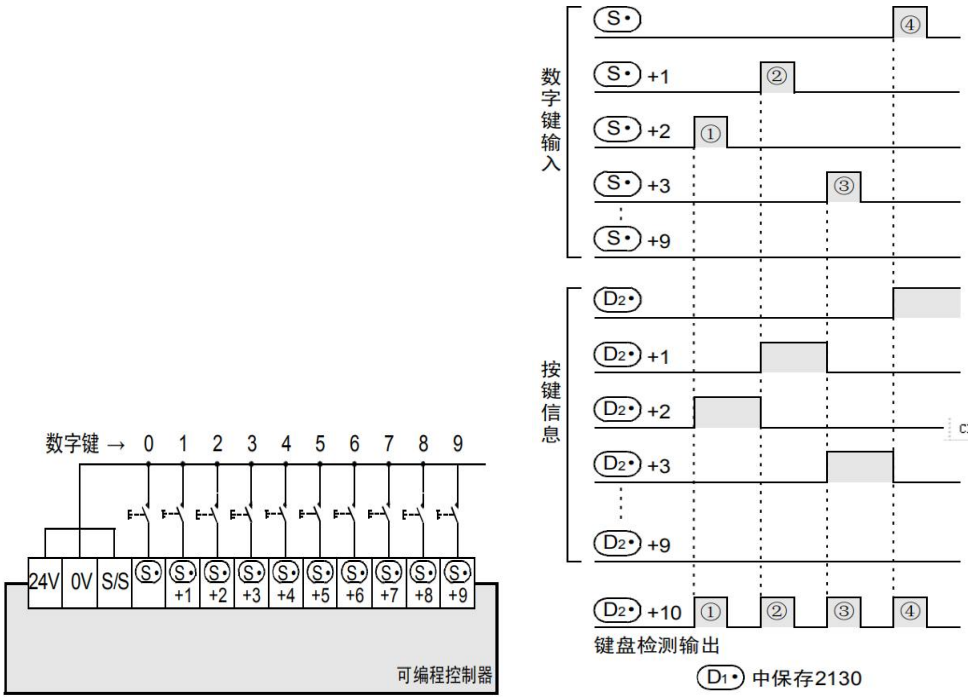
— 输入的数值，以 BIN(2 进制数)保存。

— 按照①、②、③、④的顺序按数字键后，在 (d1)中保存为 2130。

2) 有关按键信息的[(d2)～ +10]

— (d2)～ +9 的按键信息，根据所按下的键 ON/OFF。

— 当 0~9 中任意一个键被按下时，(d2) +10 的键盘检测输出为 ON。



注意事项

1. 同时按下键盘时

   同时按下多个键时，只有先按下的键有效。

2. 指令触点为 OFF 时

   即使为 OFF，（d2）的内容也不改变，但是（d2）~（d2）+10 为止都变为 OFF。

3. 软元件的占用点数

   1）连接数字键的输入，占用从（s）开始的 10 点。

   即使不连接数字键(不使用)时，由于（d2）已被占用，所以不能用作其他用途。

   2）占用从按键信息输出用的起始软元件（d2）开始的 11 点。

   请注意不要与机器其他控制中使用的软元件重复。

   （d2）~（d2）+9：根据数字键 0~9 的输入置 ON。

   （d2）+10： 按下 0~9 之间的任意一个键时为 ON。(键盘检测输出)

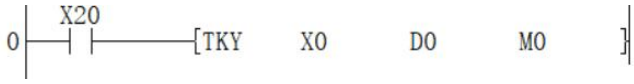
4. 指令的使用次数的限制

   TKY 指令或 DTKY 指令，在程序中只可以使用其中 1 个。

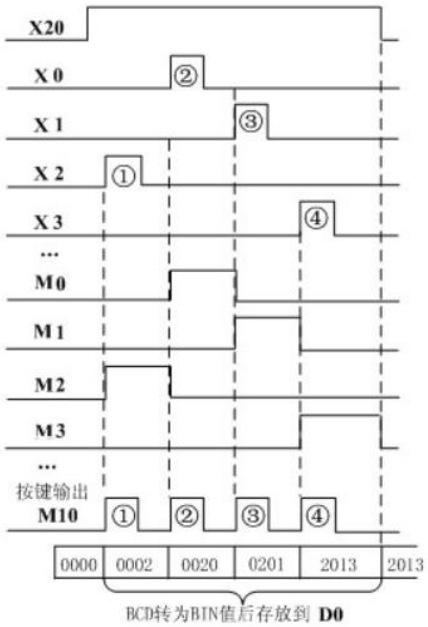
错误码

错误码	内容
4085H	读应用指令中(s)(d1)输出结果超出软元件范围。
4086H	写应用指令中(d1)(d2)输出结果超出软元件范围。

示例



若要输入数字“2013”，按顺序按键 2、0、1、3（X2、X0、X1、X3）即可，PLC 内部变量的动作如下图。



• 由指令中参数的设置，X0~X11 分别对应 0~9 数字键；M0~M9 对应键的状态；当有任何按键按下时，按键输出单元 M10 都会置位；

• 按键值(如 2013)被转换为 BIN 格式后存入指定的 D1 单元 D0；(D0=0x7DD)，即使驱动的能流变为 OFF，D0 也不会改变；

• 当有多个按键按下时，先检测到的按键有效；当输入的数字超过 4 位后，最先输入的数字变化溢出，只留下输入的最后 4 个数字。

DTKY/32 进制数字键输入

DTKY

通过 0~9 的键盘(数字键)输入，对定时器和计数器等设定数据的指令。

-[STMR （s1）（s2）（d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	输入数字键的起始位软元件[占用 10 点]	-	位	ANY_BOOL	--
(d1)	保存数据的字软元件编号	[0,99999999]	有符号 BIN32	ANY32_S	DINT
(d2)	按键信息为 ON 的起始位软元件编号[占用 11 点]	-	位	ANY_BOOL	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DTKY	参数 1	●	●	●	●	●					●															●	
	参数 2												●	●	●	●	●	●	●	●	●	●				●	
	参数 3		●	●	●	●					●															●	

功能

对连接的数字键输入[(s)～ +9]按动键盘，将输入的数值保存在(d1)中，在(d2)～ +10 中输出键盘输入信息以及检测到的键盘输出。

1) 有关输入的数值(d1)

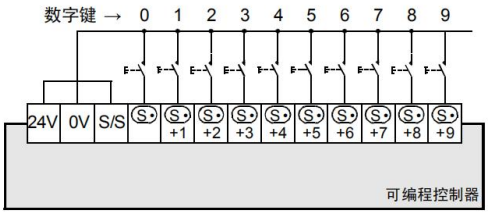
— 若为 99,999,999 以上时，从高位数溢出。

— 输入的数值，以 BIN(2 进制数)保存。

2) 有关按键信息的[(d2)～ +10]

— (D2)～+9 的按键信息，根据所按下的键 ON/OFF。

— 当 0~9 中任意一个键被按下时，(d2) +10 的键盘检测输出为 ON。



注意事项

1. 同时按下键盘时

   同时按下多个键时，只有先按下的键有效。

2. 指令触点为 OFF 时

   即使为 OFF，（d2）的内容也不改变，但是（d2）～（d2）+10 为止都变为 OFF。

3. 软元件的占用点数

   1) 连接数字键的输入，占用从（s）开始的 10 点。

   即使不连接数字键(不使用)时，由于（d2）已被占用，所以不能用作其他用途。

   2) 占用从按键信息输出用的起始软元件（d2）开始的 11 点。

   请注意不要与机器其他控制中使用的软元件重复。

   （d2）～（d2）+9：根据数字键 0~9 的输入置 ON。

   （d2）+10：按下 0~9 之间的任意一个键时为 ON。(键盘检测输出)

4. 指令的使用次数的限制

TKY 指令或 DTKY 指令，在程序中只可以使用其中 1 个。

错误码

错误码	内容
4085H	读应用指令中(s)(d1)输出结果超出软元件范围。
4086H	写应用指令中(d1)(d2)输出结果超出软元件范围。

示例

X20 接通时，若要输入数字“20205689”，按顺序按键 2、0、2、0、5、6、8、9（X2、X0、X2、X0、X5、X6、X10、X11）即可，则（（D1,D0）中的值为 20205689）

7.8 数据转换指令

BCD/BIN→BCD

BCD(P)

将(s)中指定的软元件的 BIN 数据转换为 BCD 后，存储到(d)中指定的软元件中。

CPU 模块的运算采用 BIN(2 进制数)数据进行处理，用于在配有 BCD 译码器的 7 段显示器中显示数值。

-[BCD (s) (d)]

内容、范围、数据类型

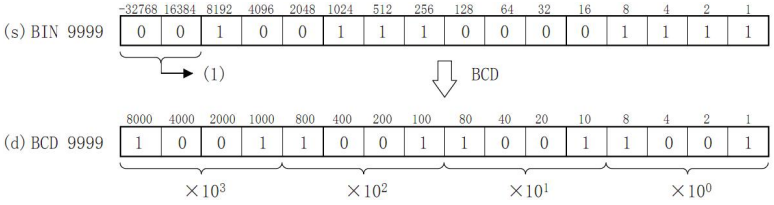
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	BIN 数据或存储了 BIN 数据的起始软元件	[0,9999]	有符号 BIN16	ANY16	INT
(d)	存储 BCD 数据的起始软元件	--	BCD4 位数	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BCD	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

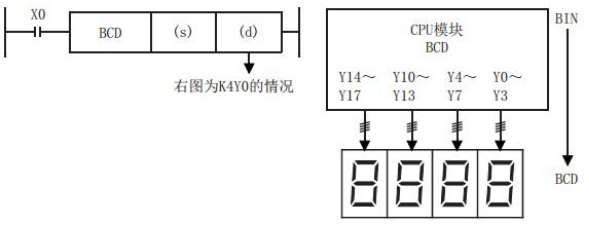
功能

- 将(s)中指定的软元件的 BIN16 位数据(0~9999)转换为 BCD4 位数据后，存储到(d)中指定的软元件中。
- (s)中指定的数据可在 0~9999(BCD)范围内转换。 (s)或(d)中指定的数据为位数指定时，其情况如下表所示。



(1)：必须置为 0。

- (s)中指定的数据通过 BCD(10 进制数)，可在 K0~K9999 范围内转换。
- (s)或(d)中指定的数据为位数指定时，其情况如下表所示。



(d)	位数	数据范围
K1Y0	1 位	0~9
K2Y0	2 位	00~99
K3Y0	3 位	000~999
K4Y0	4 位	0000~9999

注意事项

四则运算(+-×÷)、递增、递减指令等 CPU 模块内的运算均通过 BIN(2 进制数)进行。因此，将 BCD(10 进制数)的数字开关信息发送至 CPU 模块时，请使用 BIN(P)指令(BCD→BIN 转换传送指令)。另外，向 BCD(10 进制数)的 7 段显示器输出时，请使用 BCD(P)指令(BIN→BCD 转换传送)。

错误码

错误码	内容
4084H	应用指令中(s)输入了超出可指定范围的数据。
4085H	读应用指令中(s)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

当 M0 置位时，将 D200 的 BIN 值作 BCD 转换后存入 K1Y0。

福州富昌维控电子科技有限公司

BIN/4 位 BCD→BIN

BIN(P)

将(s)中指定的软元件的 BCD 数据转换为 BIN 后，存储到(d)中指定的软元件中。

与数字开关相同，将 BCD(10 进制数)设定的数值转换为可通过 CPU 模块运算操作的 BIN(2 进制数)，并读取时使用。

-[BIN (s) (d)]

内容、范围、数据类型

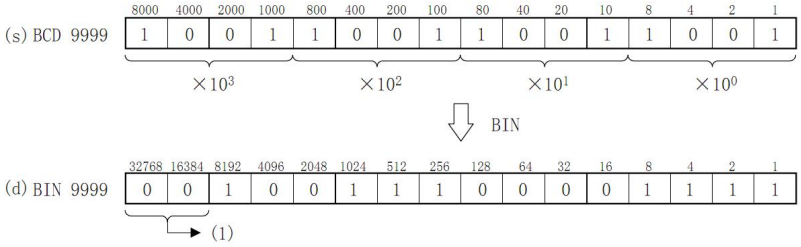
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	BCD 数据或存储了 BIN 数据的起始软元件	[0,9999]	BCD4 位数	ANY16	INT
(d)	存储 BIN 数据的起始软元件	--	有符号 BIN16	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BIN	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

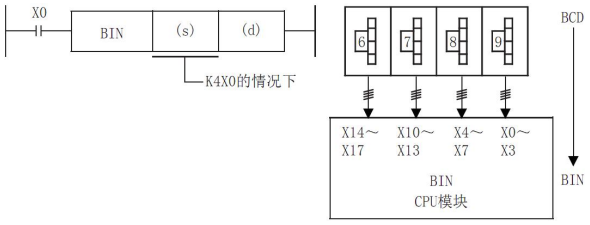
功能

- 将(s)中指定的软元件的 BCD 4 位数据(0~9999)转换为 BIN16 位数据后，存储到(d)中指定的软元件中。



（1）：必须变为 0。

- (s)中指定的数据可在 0~9999(BCD)范围内转换。
- (s)或(d)中指定的数据为位数指定时，其情况如下表所示。



(d)	位数	数据范围
K1X0	1 位	0~9
K2X0	2 位	00~99
K3X0	3 位	000~999
K4X0	4 位	0000~9999

注意事项

四则运算(+-×÷)、递增、递减指令等 CPU 模块内的运算均通过 BIN(2 进制数)进行。因此，将 BCD(10 进制数)的数字开关信息发送至 CPU 模块时，请使用 BIN(P)指令(BCD→BIN 转换传送指令)。另外，向 BCD(10 进制数)的 7 段显示器输出时，请使用 BCD(P)指令(BIN→BCD 转换传送)。

错误码

错误码	内容
4084H	应用指令中(s)输入了超出可指定范围的数据。
4085H	读应用指令中(s1)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例

当 M0 置位时，将 K1Y0 的 BCD 值作 BIN 转换后存入 D200。

DBIN/8 位 BCD→BIN

DBIN(P)

将(s)中指定的软元件的 BCD 数据转换为 BIN 后，存储到(d)中指定的软元件中。

与数字开关相同，将 BCD(10 进制数)设定的数值转换为可通过 CPU 模块运算操作的 BIN(2 进制数)，并读取时使用。

-[DBIN (s) (d)]

内容、范围、数据类型

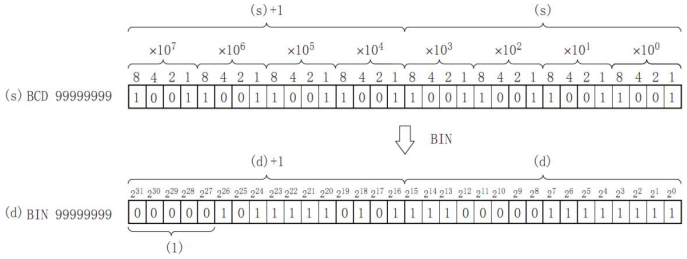
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	BCD 数据或存储了 BIN 数据的起始软元件	[0,99999999]	BCD 8 位数	ANY32	DINT
(d)	存储 BIN 数据的起始软元件	--	有符号 BIN32	ANY32	DINT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBIN	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2												●	●	●	●	●	●	●	●							●	●

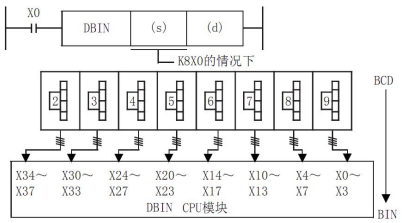
功能

- 将(s)中指定的软元件的 BCD 8 位数据(0~99999999)转换为 BIN32 位数据后，存储到(d)中指定的软元件中。



(1)：必须变为 0。

- (s)中指定的数据可在 0~99999999(BCD)范围内转换。
- (s)或(d)中指定的数据为位数指定时，其情况如下表所示。



(d)	位数	数据范围	(d)	位数	数据范围
K1X0	1 位	0~9	K5X0	5 位	00000~99999
K2X0	2 位	00~99	K6X0	6 位	000000~999999
K3X0	3 位	000~999	K7X0	7 位	0000000~9999999
K4X0	4 位	0000~9999	K8X0	8 位	00000000~99999999

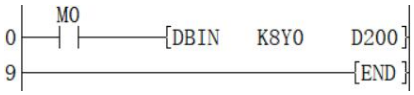
注意事项

四则运算(+-×÷)、递增、递减指令等 CPU 模块内的运算均通过 BIN(2 进制数)进行。因此，将 BCD(10 进制数)的数字开关信息发送至 CPU 模块时，请使用 BIN(P)指令(BCD→BIN 转换传送指令)。另外，向 BCD(10 进制数)的 7 段显示器输出时，请使用 BCD(P)指令(BIN→BCD 转换传送)。

错误码

错误码	内容
4084H	应用指令中(s)输入了超出可指定范围的数据。
4085H	读应用指令中(s1)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例。



当 M0 置位时，将 K8Y0 的 BCD 值作 BIN 转换后存入 D200。

FLT/BIN 整数→2 进制浮点数

FLT(P)

将 BIN16 位整数值转换成 2 进制浮点数（实数）的指令。

-[FLT （s） (d)]

内容、范围、数据类型

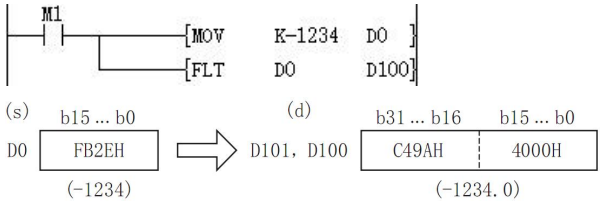
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	保存 BIN 整数值的数据寄存器编号	-	有符号 BIN16 位	ANY16	INT
(d)	保存 2 进制浮点数（实数）的数据寄存器编号	-	单精度实数	ANYREAL_32	——

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
FLT	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

• 将（s）中指定的带符号 16 位数据转换为单精度实数后，存储到（d）+1，（d）中。



注意事项

• 由于在各 2 进制浮点数（实数）运算指令中，指定的 K、H 的值会自动转换成 2 进制浮点数（实数），所以不需要使用 FLT 指令进行转换。

• 这个指令的逆变换指令是 INT（将 2 进制浮点数值转换成 BIN 整数）。

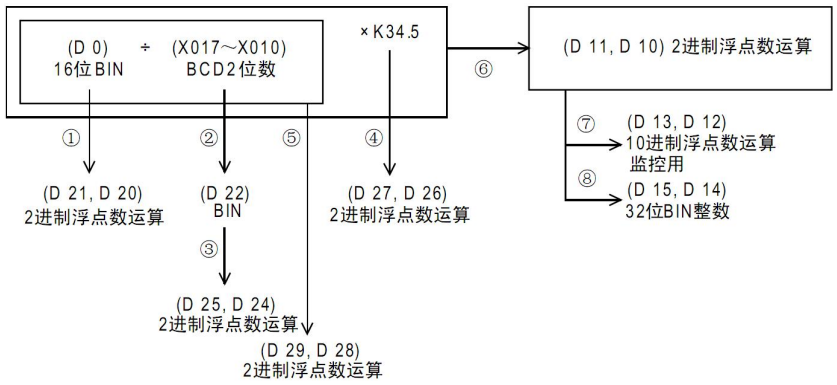
错误码

错误码	内容
4085H	读应用指令(s)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

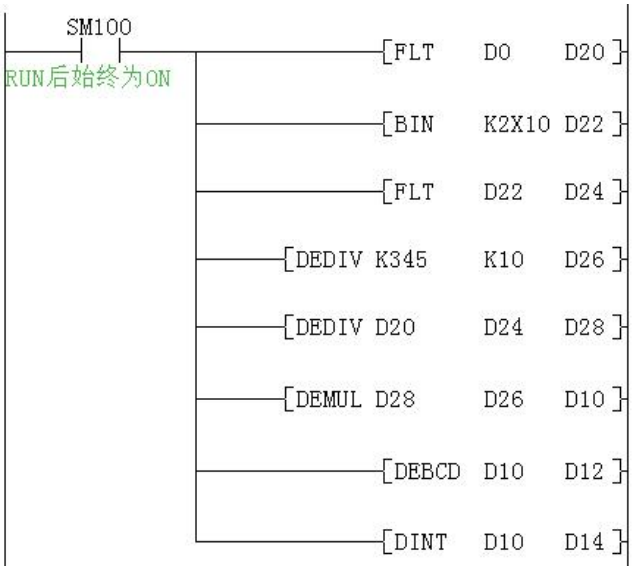
示例

采用二进制浮点数运算的四则运算

（1）计算示例



（2）顺控程序



（D0） → （D21,D20）
BIN 2 进制浮点数运算

（X17~X10） → （D22）
BCD BIN

（D22） → （D25,D24）
BIN 2 进制浮点数运算

（K345÷K10） → （D27,D26）
2 进制浮点数运算

(D21,D20)÷(D25,D24) → （D29,D28）
2 进制浮点数除法运算 2 进制浮点数运算

(D29,D28)×(D27,D26) → （D11,D10）
2 进制浮点数乘法运算

(D11,D10) → （D13,D12）
2 进制浮点数运算 10 进制浮点数运算监控用

(D11,D10) → （D13,D12）
2 进制浮 点数运算 32 位 BIN 整型

DFLT/BIN 整数→2 进制浮点数

DFLT(P)

将 BIN32 位整数值转换成 2 进制浮点数（实数）的指令。

-[DFLT （s） (d)]

内容、范围、数据类型

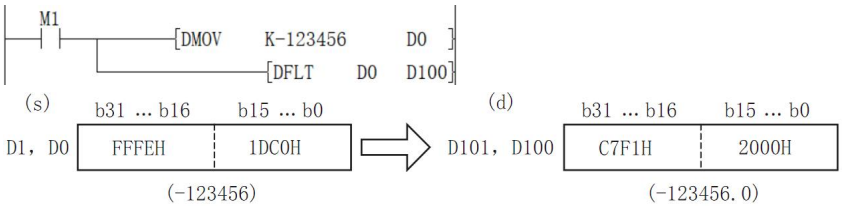
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	保存 BIN32 整数值的数据寄存器编号	-	有符号 BIN32 位	ANY32	DINT
(d)	保存 2 进制浮点数（实数）的数据寄存器编号	-	单精度实数	ANYREAL_32	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DFLT	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

• 将（s）中指定的带符号 BIN32 位数据转换为单精度实数后，存储到（d）+1，（d）中。



注意事项

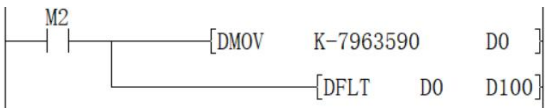
• 由于在各 2 进制浮点数（实数）运算指令中，指定的 K、H 的值会自动转换成 2 进制浮点数（实数），所以不需要使用 DFLT 指令进行转换。

• 这个指令的逆变换指令是 INT（将 2 进制浮点数值转换成 BIN 整数）。

错误码

错误码	内容
4085H	读应用指令(s)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



M2=ON 时，将[D1,D0]中的 BIN32 位整数-7963590 转换成单精度浮点数-7963590.0 存入到[D101，D100]软元件中。

VAL/字符串→BIN16 位数据转换

VAL(P)

将(s)中指定的软元件编号以后中存储的字符串转换为 BIN16 位数据后，将位数存储到(d1)中，将 BIN 数据存储到(d2)中。

[VAL （s） （d1） （d2）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	转换为 BIN 数据的字符串或存储了字符串的起始软元件	-	字符串	ANYSTRING_SINGLE	--
(d1)	存储转换后的 BIN 数据的位数的起始软元件	-	有符号 BIN16 位	ANY16_S_ARRAY	--
(d2)	存储转换后的 BIN 数据的起始软元件	-	有符号 BIN16 位	ANY16_S	--

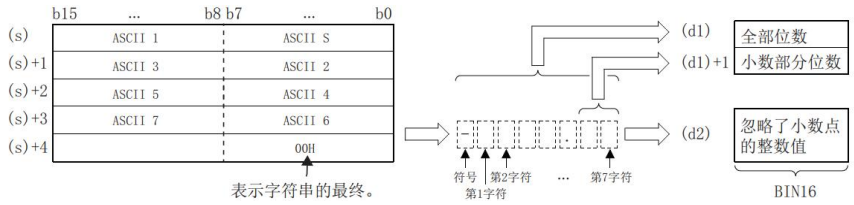
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
VAL	参数 1															●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

功能

将(s)中指定的软元件编号以后中存储的字符串转换为BIN16位数据后，将位数存储到(d1)中，将BIN数据存储到(d2)中。在字符串→BIN 转换中，将(s)中指定的软元件编号开始至存储了 00H 的软元件编号为止的数据作为字符串处理。

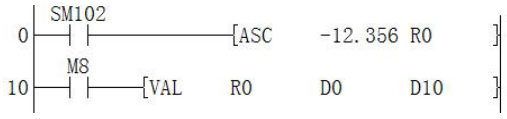
(d1)中存储的全部位数存储表示数值的所有字符(包括符号、小数点)的数。(d1)+1 中存储的小数部位数存储表示 2EH(.)以后的小数部的字符数。对于(d2)中存储的 BIN16 位数据，将忽略小数点的字符串转换为 BIN 值后存储。



错误码

错误码	内容	错误码	内容
4085H	(s) 读取地址超出软元件范围	4082H	(s)指定的字符串不能够转换成数值 例如：第一个字符不是负号或者空格，数字中间出现空格，出现两次小数点，除了第一个字符出现非字符和小数点的符号字符串中的数值去掉小数点后，超出[-32768,32767]这个范围之间。 例如 3.4000 去掉小数点后是 34000 已经超出范围。
408AH	(s)指定的字符串的字符数为 2~8 以外时。		
408BH	(s)取字符串时读取到软元件最大范围但是没有找到 00H 做为结束。		
4086H	(d)在使用偏移的情况下偏移后的地址超出软元件范围 。		

示例



以上得出的结果：D0 对应 str 长度为 7，D1 对应小数点长度为 3。

D0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	7
D1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3

D10 对应忽略小数点的数值为-12356。

软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D10	0	0	1	1	1	1	0	1	1	1	1	1	0	0	1	1	-12356

DVAL/字符串→BIN32 位数据转换

DVAL(P)

将(s)中指定的软元件编号以后中存储的字符串转换为 BIN32 位数据后，将位数存储到(d1)中，将 BIN 数据存储到(d2)中。

-[DVAL　(s)　(d1)　(d2)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	转换为 BIN 数据的字符串或存储了字符串的起始软元件	-	字符串	ANYSTRING_SINGLE	--
(d1)	存储转换后的 BIN 数据的位数的起始软元件	-	有符号 BIN16 位	ANY16_S_ARRAY	--
(d2)	存储转换后的 BIN 数据的起始软元件	-	有符号 BIN32 位	ANY32_S	--

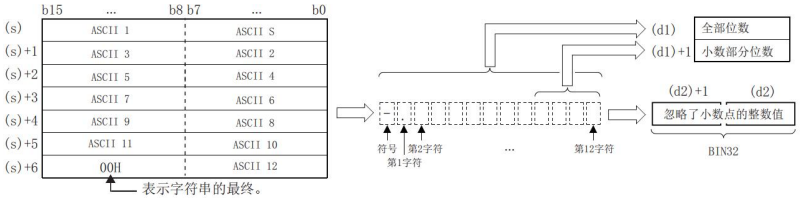
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DVAL	参数 1															●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

将(s)中指定的软元件编号以后中存储的字符串转换为BIN32位数据后，将位数存储到(d1)中，将BIN数据存储到(d2)中。在字符串→BIN 转换中，将(s)中指定的软元件编号开始至存储了 00H 的软元件编号为止的数据作为字符串处理。

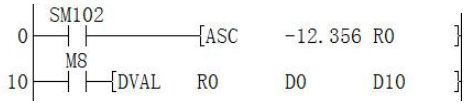
(d1)中存储的全部位数存储表示数值的所有字符(包括符号、小数点)的数。(d1)+1 中存储的小数部位数存储表示 2EH(.)以后的小数部的字符数。对于(d2)中存储的 BIN32 位数据，将忽略小数点的字符串转换为 BIN 值后存储。



错误码

错误码	内容	错误码	内容
4085H	(s) 读取地址超出软元件范围 。	4082H	(s)指定的字符串不能够转换成数值 例如：第一个字符不是负号或者空格，数字中间出现空格，出现两次小数点，除了第一个字符出现非字符和小数点的符号，字符串中的数值去掉小数点后，数值超出-2147483648~2147483647 这个范围之间。 例如：3.000000000 去掉小数点后是 3000000000 已经超出范围。
408BH	(s)读取字符串时读取到软元件最大范围但是没有找到 00H 做为结束。		
408AH	(s)指定的字符串的字符数为 2~13 以外时。		
4086H	(d1) (d2)在使用偏移的情况下偏移后的地址超出软元件范围 。		

示例



左边得出的结果：

D0 对应 str 长度为 7，D1 对应小数点长度为 3。

D10 对应忽略小数点的数值为-12356。

软元件	+0	+1	+2	+3
D0	196615	0	0	0
D8	0	-12356	0	0

ASCI/HEX 代码数据→ASCII 转换

ASCI(P)

将(s)中指定的 HEX 代码数据中的 n 字符(位)转换为 ASCII 码后，存储到(d)中指定的软元件编号以后。

-[ASCI （s） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储要转换的 HEX 代码的软元件起始编号	-	BIN16 位	ANY16	--
(d)	存储转换后的 ASCII 码的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(n)	要转换的 HEX 代码的字符数(位数)	[1,256]	BIN16 位	ANY16_U	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ASCI	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3																	●	●	●			●	●		●	●

功能

• 将通过(s)中指定的 HEX 代码数据中的(n)指定的字符数(位)转换为 ASCII 码后，存储到(d)中指定的软元件编号以后。

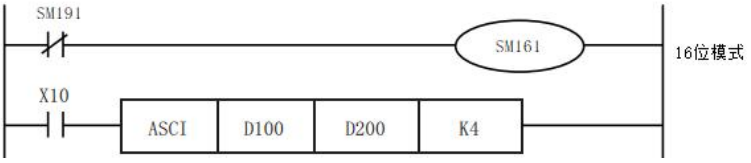
• ASCI(P)指令在转换时使用的模式有 16 位模式和 8 位模式。关于各个模式的动作，请参阅以后的内容。

（1）16 位转换模式(SM161=OFF 时)

•将存储在(s)中指定的软元件以后的 HEX 代码的各位转换为 ASCII，传送到(d)中指定的软元件的高低 8 位(字节)。

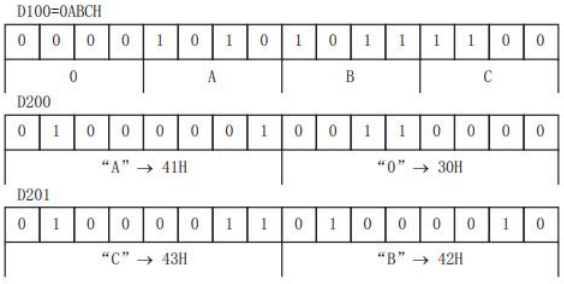
• 在 16 位转换模式中使用的情况下，应将 SM161 始终置为 OFF 使用。

下述程序的情况下，如下所示执行转换。



（s）以后的软元件：D100=0ABCH； D101=1234H； D102=5678H

(n)=K4 的情况下的位构成：



ASCII 码：

"0" =30H "1" =32H "5" =35H

"A" =41H "2" =32H "6" =36H

"B" =42H "3" =33H "7" =37H

"C" =43H "4" =34H "8" =38H

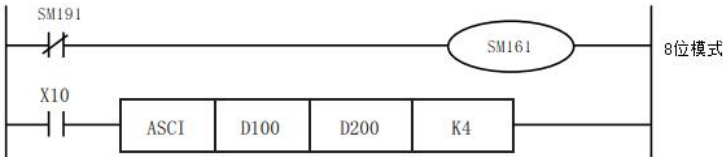
指定位(字符)数和转换结果：

(n) / (d)	K1	K2	K3	K4	K5	K6	K7	K8	K9
D200 下	C	B	A	0	4	3	2	1	8
D200 上		C	B	A	0	4	3	2	1
D201 下			C	B	A	0	4	3	2
D201 上				C	B	A	0	4	3
D202 下					C	B	A	0	4
D202 上	不变化					C	B	A	0
D203 下							C	B	A
D203 上								C	B
D204 下									C

（2）8 位转换模式(SM161=ON 时)

将存储在(s)中指定的软元件以后的 HEX 代码的各位转换为 ASCII，传送到(d)中指定的软元件的下 8 位(字节)。在 8 位转换模式中使用的情况下，应将 SM161 始终置为 ON 使用。

下述程序的情况下，如下所示执行转换。



（s）以后的软元件：D100=0ABCH； D101=1234H； D102=5678H

如果将 SM161 置为 ON，将变为 8 位模式， 进行如下所示的转换处理。

(n)=K2 的情况下的位构成：

ASCII 码：

“0”=30H “1”=31H “5”=35H

“A”=41H “2”=32H “6”=36H

“B”=42H “3”=33H “7”=37H

“C”=43H “4”=34H “8”=38H

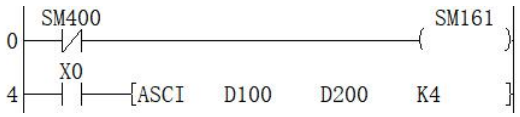
指定位(字符)数和转换结果：

(n) / (d)	K1	K2	K3	K4	K5	K6	K7	K8	K9
D200	C	B	A	0	4	3	2	1	8
D201		C	B	A	0	4	3	2	1
D202			C	B	A	0	4	3	2
D203				C	B	A	0	4	3
D204					C	B	A	0	4
D205	不变化					C	B	A	0
D206							C	B	A
D207								C	B

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(n)中指定的数值在 1~256 以外时。

示例



• 16 位转换模式(SM161=OFF 时)

将存储在 d100 中指定的软元件以后的 HEX 代码的各位转换为 ASCII，传送到 d200 中指定的软元件的高低 8 位(字节)。在 16 位转换模式中使用的情况下，应将 SM161 始终置为 OFF 使用。

HEX/ASCII→HEX 代码数据转换

HEX(P)

将(s)中指定的软元件编号以后，(n)中指定的字符数中存储的 ASCII 数据转换为 HEX 代码后，存储到(d)中指定的软元件编号以后。

-[HEX （s） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了转换为 HEX 代码的 ASCII 数据的起始软元件	-	字符串	ANYSTRING_SINGLE	--
(d)	存储转换后的 HEX 代码的起始软元件	-	BIN16 位	ANY16	--
(n)	转换的 ASCII 数据的字符数(字节数)	[1,256]	BIN16 位	ANY16_U	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
HEX	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3																	●	●	●			●	●		●	●

功能

• 将(s)中指定的软元件编号以后，(n)中指定的字符数中存储的 ASCII 数据转换为 HEX 代码后，存储到(d)中指定的软元件编号以后。HEX(P)指令在转换时使用的模式有 16 位转换模式与 8 位转换模式。关于各种模式的动作，请参阅后文内容。

（1）16 位转换模式(SM161=OFF 时)

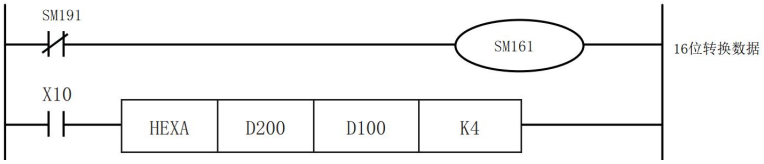
将(s)中指定的软元件的上下 8 位(字节)中存储的 ASCII 数据转换为 HEX 代码后，每 4 位数向(d)中指定的软元件传送。要转换的字符数在(n)中指定。

SM161 与 ASC、ASCI、BCC、CCD、CRC 指令共用。在 16 位转换模式下使用时，请将 SM161 始终置为 OFF。

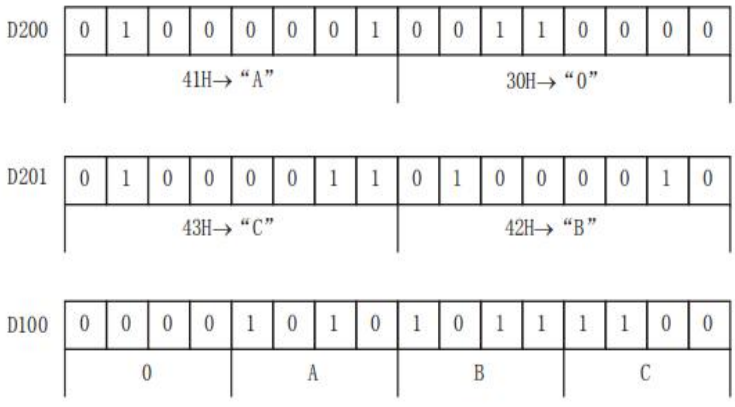
SM161 在 RUN→STOP 时被清除。

另外，还需要将 ASCII 数据存储到 16 位转换模式下，在(s)中指定的软元件的高 8 位中。

在下列程序中，将按照以下方式执行转换。



(n)=K4 的情况下的位构成：



指定字符数与转换结果“ • ”变为 0。

转换源数据：

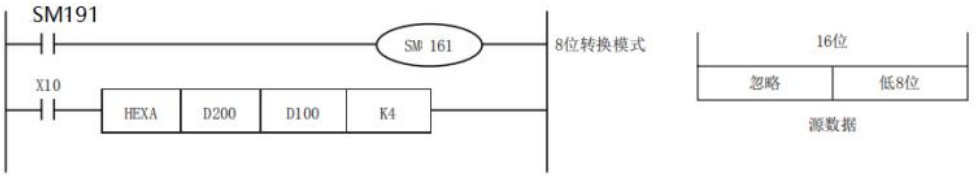
(s)	ASCII 数据	HEX 转换
D200 下	30H	0
D200 上	41H	A
D201 下	42H	B
D201 上	43H	C
D202 下	31H	1
D202 上	32H	2
D203 下	33H	3
D203 上	34H	4
D204 下	35H	5

(n)		1	2	3	4	5	6	7	8	9
(d)	D102	不变化								...OH
	D101					...OH	..OAH	.OABH	OABCH	ABC1H
	D100	...OH	..OAH	.OABH	OABCH	ABC1H	BC12H	C123H	1234H	2345H

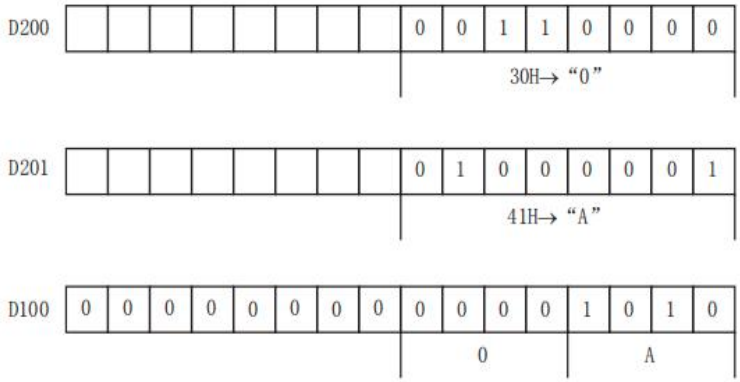
(2) 8 位转换模式(SM161=ON 时)

将(s)中指定的软元件的下 8 位中存储的 ASCII 数据转换为 HEX 代码后，每 4 位数向(d)中指定的软元件传送。

要转换的字符数在(n)中指定。SM161 与 ASC、ASCI、BCC、CCD、CRC 指令共用。在 8 位转换模式下使用时，请将 SM161 始终置为 ON。 SM161 在 RUN→STOP 时被清除。 在下列程序中，将按照以下方式执行转换。



(n)=K2 的情况下的位构成：



转换源数据：

(s)	ASCII 数据	HEX 转换
D200	30H	0
D201	41H	A
D202	42H	B
D203	43H	C
D204	31H	1
D205	32H	2
D206	33H	3
D207	34H	4
D208	35H	5

指定字符数与转换结果 "." 变为 0。

(n)		1	2	3	4	5	6	7	8	9
(d)	D102	不变化								...OH
	D101					...OH	..OAH	.OABH	OABCH	ABC1H
	D100	...OH	..OAH	.OABH	OABCH	ABC1H	BC12H	C123H	1234H	2345H

错误码

错误码	内容
4084H	(n)中指定的数值超范围时。
	(s)中设置了 30H~39H、41H~46H 以外的 ASCII 代码时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

将(s)中指定的软元件的上下 8 位(字节)中存储的 ASCII 数据转换为 HEX 代码后，每 4 位数向(d)中指定的软元件传送。要转换的字符数在(n)中指定。

SM161 与 ASC、ASCI、BCC、CCD、CRC 指令共用。在 16 位转换模式下使用时，请将 SM161 始终置为 OFF。

CCD/校验码

CCD(P)

计算在通信中等使用的出错检查方法的水平奇偶校验值及和校验值。出错检查方法除了这些以外，还有 CRC(Cyclic Redundancy Check)。要计算 CRC 值的情况下，应使用 CRC(P)指令。

-[CCD（s） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	对象软元件起始编号	-	BIN16 位	ANY16	--
(d)	算出的数据的储存目标软元件起始编号	-	BIN16 位	ANY16_ARRAY（要素数：2）	--
(n)	数据数	[1,256]	BIN16 位	ANY16_U	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
CCD	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3																	●	●	●			●	●		●	●

功能

• 算出存储在(s)～(s)+(n)-1 中的数据的加法运算数据与水平奇偶校验数据，并将加法运算数据存储到(d)中，水平奇偶校验。

数据存储到(d)+1 中。该指令在计算时使用的模式有 16 位模式和 8 位模式。关于各个模式的动作，请参阅以后的内容。

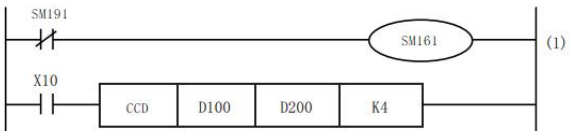
（1）16 位转换模式(SM161=OFF 时)

关于(s)为起始的(n)点的数据，其高低各 8 位数据的加法运算数据和水平奇偶校验数据存储到(d)和(d)+1 的软元件中。

SM161 为与 ASC、ASCI、BCC、CCD、CRC 指令共用。在 16 位中使用的情况下，应始终置为 OFF 使用。

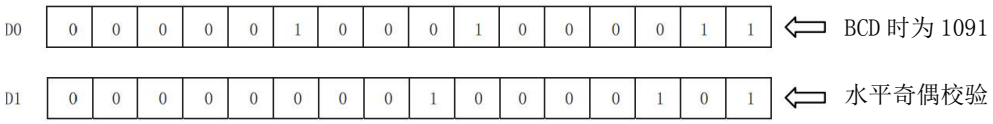
SM161 在 RUN→STOP 时清除。

下述程序的情况下，如下所示执行转换。



16 位转换模式

(s)	数据的内容示例	
D100 下	K100 = 0 1 1 0 0 1 0 0	
D100 上	K111 = 0 1 1 0 1 1 1 ①	←
D101 下	K100 = 0 1 1 0 0 1 0 0	
D101 上	K 98 = 0 1 1 0 0 0 1 0	
D102 下	K123 = 0 1 1 1 1 0 1 ①	←
D102 上	K 66 = 0 1 0 0 0 0 1 0	
D103 下	K100 = 0 1 1 0 0 1 0 0	
D103 上	K 95 = 0 1 0 1 1 1 1 ①	←
D104 下	K210 = 1 1 0 1 0 0 1 0	
D104 上	K 88 = 0 1 0 1 1 0 0 0	
合计	K1091	1 的个数如果是奇数，水平奇偶校验为 1
水平奇偶校验	1 0 0 0 0 1 0 ①	← 1 的个数如果是偶数，水平奇偶校验为 0



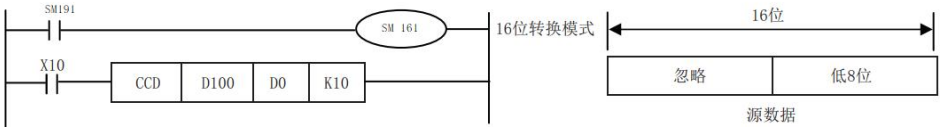
（2） 8 位转换模式(SM161=ON 时)

关于(s)为起始的(n)点的数据(仅低位 8 位)，其加法运算数据和水平奇偶校验数据存储到(d)和(d)+1 的软元件中。

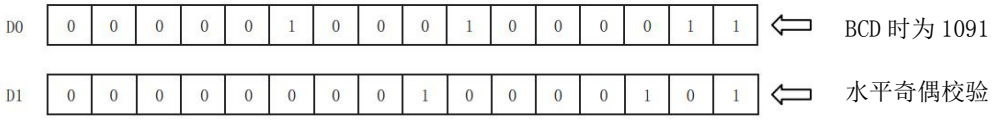
SM161 为与 ASC、ASCI、BCC、CCD、CRC 指令共用。在 8 位中使用的情况下，应始终置为 ON 使用。

SM161 在 RUN→STOP 时清除。

下述程序的情况下，如下所示执行转换。



(s)	数据的内容示例	
D100	K100 = 0 1 1 0 0 1 0 0	
D101	K111 = 0 1 1 0 1 1 1 ①	←
D102	K100 = 0 1 1 0 0 1 0 0	
D103	K 98 = 0 1 1 0 0 0 1 0	
D104	K123 = 0 1 1 1 1 0 1 ①	←
D105	K 66 = 0 1 0 0 0 0 1 0	
D106	K100 = 0 1 1 0 0 1 0 0	
D107	K 95 = 0 1 0 1 1 1 1 ①	←
D108	K210 = 1 1 0 1 0 0 1 0	
D109	K 88 = 0 1 0 1 1 0 0 0	
合计	K1091	1 的个数如果是奇数，水平奇偶校验为 1
水平奇偶校验	1 0 0 0 0 1 0 ①	← 1 的个数如果是偶数，水平奇偶校验为 0



错误码

错误码	内容
4084H	(n)中指定的数值在 1~256 以外时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

关于 D10 为起始的 10 点的数据，其高低各 8 位数据的加法运算数据和水平奇偶校验数据存储到 D0 和 D0+1 的软元件中。

SM161 为与 ASC、ASCI、BCC、CCD、CRC 指令共用。在 16 位中使用的情况下，应始终置为 OFF 使用。

GBIN/格雷码→BIN16 位数据转换

GBIN(P)

将(s)中指定的软元件中存储的 BIN16 位格雷码数据转换为 BIN16 位数据后，存储到(d)中指定的软元件中。

-[GBIN（s） （d）]

内容、范围、数据类型

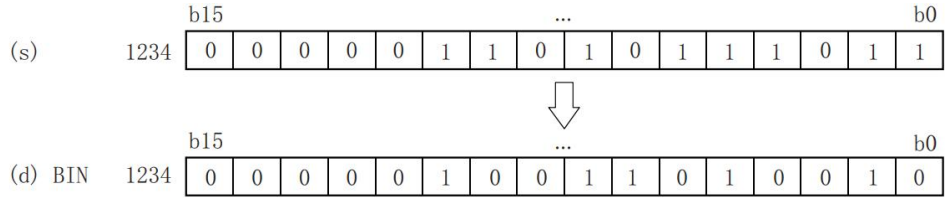
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	格雷码数据或存储了格雷码的起始软元件	[0,32767]	BIN16 位	ANY16_S	INT
(d)	存储转换后的 BIN 数据的起始软元件	-	BIN16 位	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
GBIN	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

• 将(s)中指定的软元件中存储的 BIN16 位格雷码数据转换为 BIN16 位数据后，存储到(d)中指定的软元件中。



• GRY→BIN 数学算法：从左边第二位起，将每位与左边一位解码后的值异或，作为该位解码后的值（最左边一位依然不变）。

错误码

错误码	内容
4084H	(s)中指定的数值超范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

—在使用格雷码方式的编码器检测绝对位置等情况下，可以使用。

—对 S • 而言，0~32767 范围内有效。

DGBIN/格雷码→BIN32 位数据转换

DGBIN(P)

将(s)中指定的软元件中存储的 BIN32 位格雷码数据转换为 BIN32 位数据后，存储到(d)中指定的软元件中。

内容、范围、数据类型

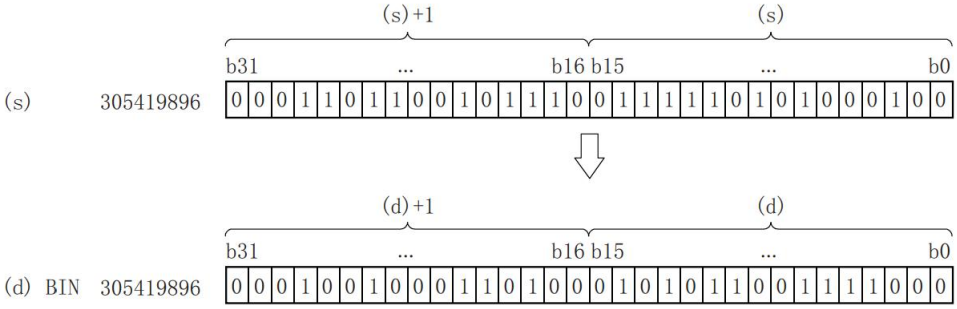
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	格雷码数据或存储了格雷码的起始软元件	[0,2147483647]	BIN32 位	ANY32_S	DINT
(d)	存储转换后的 BIN 数据的起始软元件	-	BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DGBIN	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●

功能

- 将(s)中指定的软元件中存储的 BIN32 位格雷码数据转换为 BIN32 位数据后，存储到(d)中指定的软元件中。



(s)+1：高位 16 位

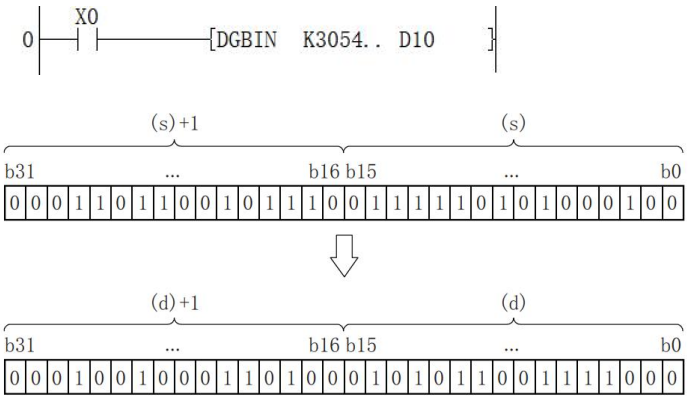
(s)：低位 16 位

• GRY→BIN 数学算法：从左边第二位起，将每位与左边一位解码后的值异或，作为该位解码后的值（最左边一位依然不变）。

错误码

错误码	内容
4084H	(s)中指定的数值超范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



GRY/BIN16 位数据→格雷码转换

GRY(P)

将(s)中指定的软元件的 BIN16 位数据转换为 BIN16 位格雷码数据后，存储到(d)中指定的软元件中。

-[GRY （s） （d）]

内容、范围、数据类型

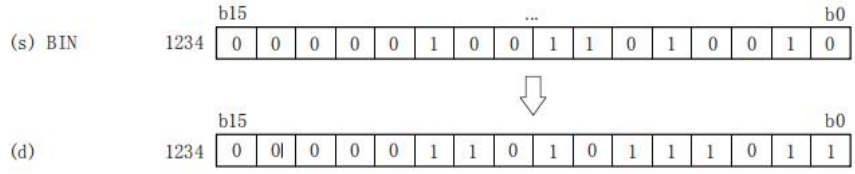
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	BIN 数据或存储了 BIN 数据的起始软元件	[0, 32767]	BIN16 位	ANY16_S	INT
(d)	存储转换后的格雷码的起始软元件	-	BIN16 位	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
GRY	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

- 将(s)中指定的 BIN16 位数据转换为 BIN16 位格雷码后，存储到(d)中指定的软元件中。



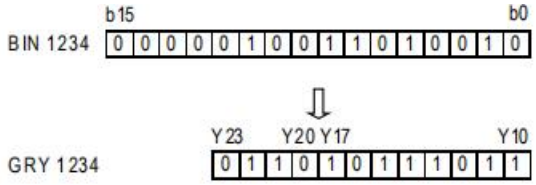
• BIN→GRY 数学算法：从最右边一位起，依次将每一位与左边一位异或（XOR），作为对应 GRY 该位的值，最左边一位不变（相当于左边是 0）。

错误码

错误码	内容
4084H	(s)中指定的数值超范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

如以上梯形图所示。



对 S • 而言，0~32767 范围内有效。

DGRY/BIN32 位数据→格雷码转换

DGRY(P)

将(s)中指定的软元件的 BIN16 位数据转换为 BIN16 位格雷码数据后，存储到(d)中指定的软元件中。

-[GRY（s）（d）]

内容、范围、数据类型

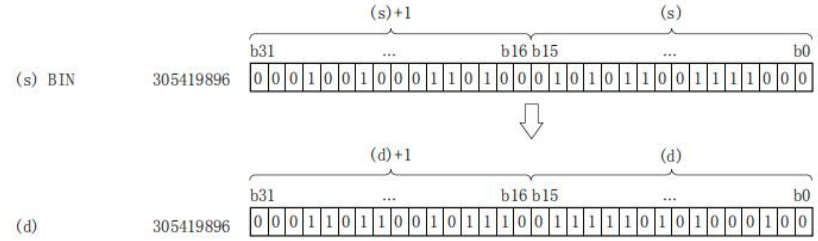
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	BIN 数据或存储了 BIN 数据的起始软元件	[0,2147483647]	BIN32 位	ANY32_S	DINT
(d)	存储转换后的格雷码的起始软元件	-	BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DGRY	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●

功能

• 将(s)中指定的 BIN32 位数据转换为 BIN32 位格雷码后，存储到(d)中指定的软元件中



(s)+1：高位 16 位

(s)：低位 16 位

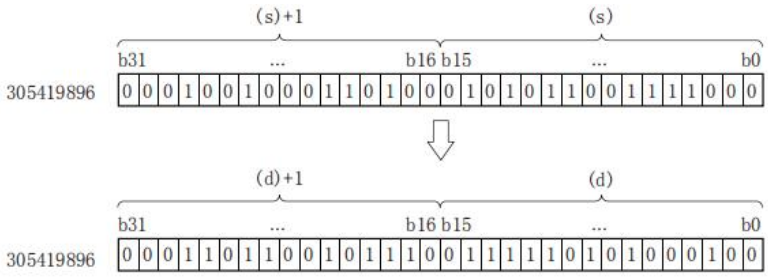
• BIN→GRY 数学算法：从最右边一位起，依次将每一位与左边一位异或（XOR），作为对应 GRY 该位的值，最左边一位不变（相当于左边是 0）。

错误码

错误码	内容
4084H	(s)中指定的数值超范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

如以上梯形图所示：



DPRUN/8 进制位传送（32 位数据）

DPRUN(P)

将指定了位数的(s)与(d)的软元件编号处理为 8 进制数后，传送数据。

-[DPRUN　(s)　(d)　]

内容、范围、数据类型

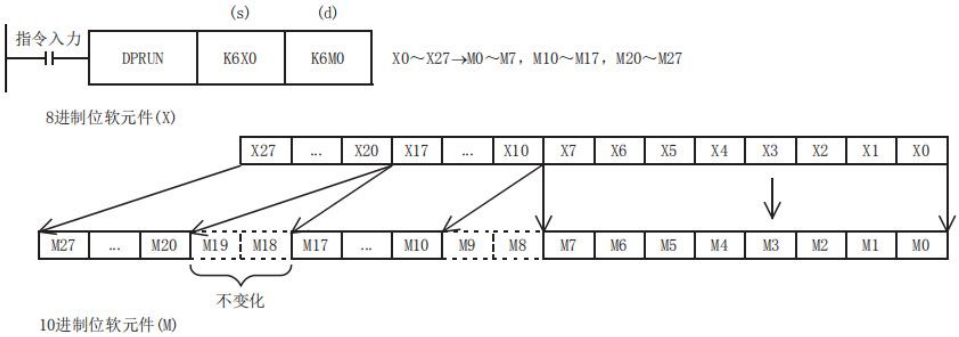
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	位数指定*1	-	BIN32 位	ANY32	--
(d)	传送目标软元件编号*1	-	BIN32 位	ANY32	--

使用的软元件

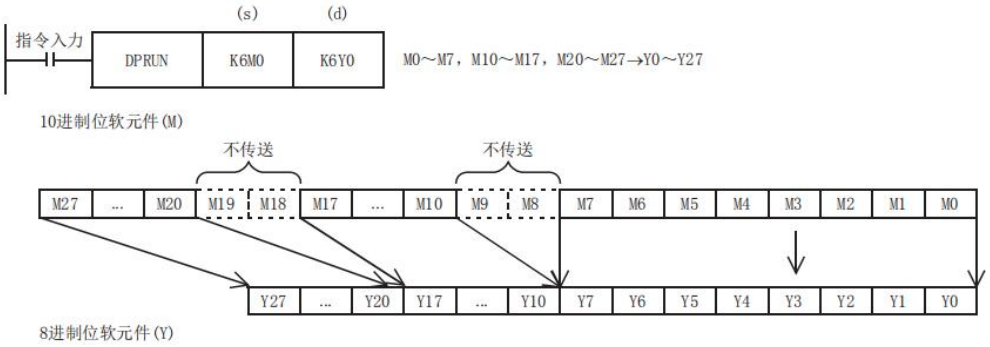
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DPRUN	参数 1											●		●											●	●
	参数 2												●	●											●	●

功能

- 8 进制位软元件→10 进制位软元件



- 10 进制位软元件→8 进制位软元件



错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

如以上梯形图所示：X0~X27 取八进制位的值，传给 M 对应位的软元件。

7.9 浮点指令

DACOS/单精度实数 COS^{-1} 运算

DACOS(P)

对(s)中指定角度的 COS^{-1}(反余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DACOS (s) (d)]

内容、范围、数据类型

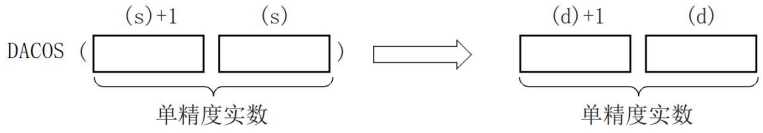
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 COS^{-1}(反余弦)运算的角度数据或存储了角度数据的软元件起始编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 1$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	0~+π	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DACOS	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

对(s)中指定角度的 COS^{-1}(反余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



(s)中指定的 COS 值可在-1.0~1.0 的范围内设置。

(d)中存储的角度(运算结果)以弧度单位存储 0~π 的值。

相关软元件如下所示：

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围 。
4086H	(d)写入地址超出软元件范围 。
4084H	(s)指定软元件的内容为非正规数、非数、±∞、超出-1.0~1.0 时。

示例

计算 0.4 的反余弦值，得出结果：1.159279

DASIN/单精度实数 SIN^{-1} 运算

DASIN(P)

对(s)中指定角度的 SIN^{-1}(反正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DASIN (s) (d)]

内容、范围、数据类型

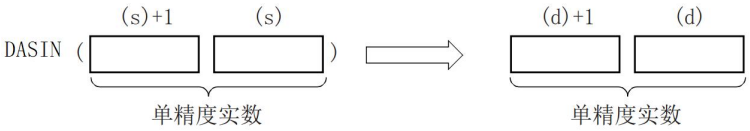
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 SIN^{-1}(反正弦)运算的角度数据或存储了角度数据的软元件起始编号	0、$2^{-126} \leqslant |(s)| < 1$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	$-\pi/2 \sim +\pi/2$	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DASIN	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

- 对(s)中指定角度的 SIN^{-1}(反正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



- (s)中指定的 SIN^{-1} 值可在-1.0~1.0 的范围内设置。
- (d)中存储的角度(运算结果)以弧度单位存储($-\pi/2$)~($\pi/2$)的值。
- 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值<2^{-126}	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围 。
4086H	(d)写入地址超出软元件范围 。
4084H	(s)指定软元件的内容为非正规数、非数、±∞、超出-1.0~1.0 时。

示例

计算 0.4 的反正弦值，得出结果：0.4115168

D0	4.115168E-1

DATAN/单精度实数 TAN^{-1}运算

DATAN(P)

对(s)中指定角度的 TAN^{-1}(反正切)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DATAN (s) (d)]

内容、范围、数据类型

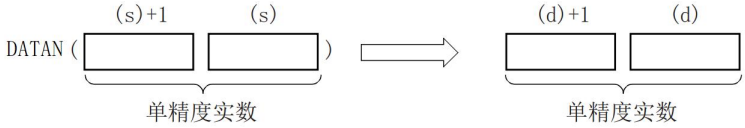
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 TAN^{-1}((反正切)运算的角度数据或存储了角度数据的软元件起始编号	0、$2^{-126} \leqslant |(s)| < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	$-\pi/2 \sim +\pi/2$	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DATAN	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

- 对(s)中指定角度的 TAN^{-1}((反正切)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



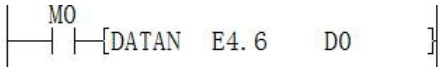
- (d)中存储的角度(运算结果)以弧度单位存储($-\pi/2$)～($\pi/2$)的值。
- 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



计算 4.6 的反正切值，得出结果：1.356736

DCOS/单精度实数 COS 运算

DCOS（P）

对(s)中指定角度的 COS(余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DCOS (s) (d)]

内容、范围、数据类型

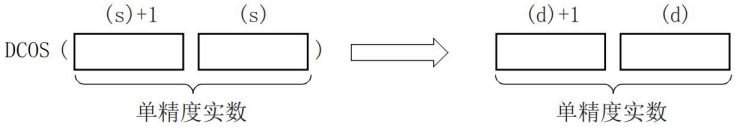
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 COS(余弦)运算的角度数据或存储了角度数据的软元件起始编号	0、$2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DCOS	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

• 对(s)中指定角度的 COS(余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



• 对于(s)中指定的角度，以弧度单位(角度×π÷180)进行设置。

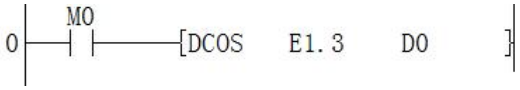
• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



计算 1.3 的余弦值，得出结果 2.674989E-1

DCOSH/单精度实数 COSH 运算

DCOSH(P)

对(s)中指定角度的 DCOSH(双曲余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DCOSH (s) (d)]

内容、范围、数据类型

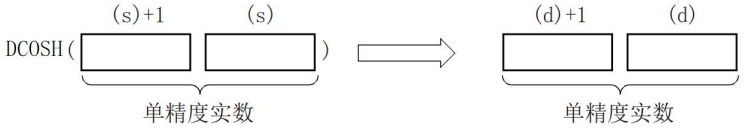
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 DCOSH(双曲余弦)运算的角度数据或存储了角度数据的软元件起始编号	$0、2^{-126} \leq \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DCOSH	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

- 对(s)中指定角度的 DCOSH(双曲余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



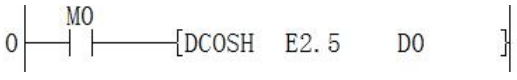
- 该指令是进行 2 进制浮点数取 COSH 值。其计算公式为 cosh 值=$(e^{s}+e^{-s})/2$ 。
- 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM151	进位	运算结果的绝对值>2^{128}	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



计算 2.5 的双曲余弦值，得出结果 6.132289

DSIN/单精度实数 SIN 运算

DSIN(P)

对(s)中指定角度的 SIN(正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DSIN (s) (d)]

内容、范围、数据类型

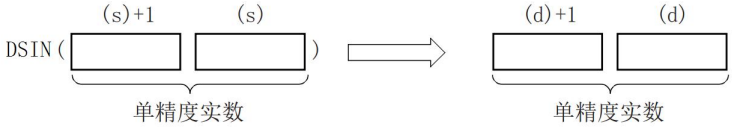
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 SIN(正弦)运算的角度数据或存储了角度数据的软元件起始编号	0、$2^{-126} \leq \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSIN	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

对(s)中指定角度的 SIN(正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



对于(s)中指定的角度，以弧度单位(角度×π÷180)进行设置。

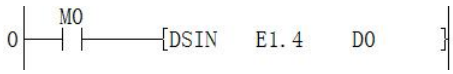
• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围 。
4086H	(d)写入地址超出软元件范围 。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



计算 1.4 的正弦值，得出结果：0.9854497

DSINH/单精度实数 SINH 运算

DSINH(P)

对(s)中指定角度的 SINH(双曲正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DSINH (s) (d)]

内容、范围、数据类型

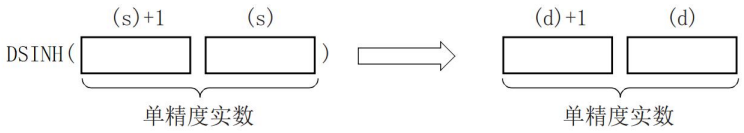
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 SINH(双曲正弦)运算的角度数据或存储了角度数据的软元件起始编号	0、2^{-126}≤|(s)|<2^{128}	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSINH	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

对(s)中指定角度的 SINH(双曲正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



该指令是进行 2 进制浮点数取 SINH 值。其计算公式为 sinh 值 =(e^{s}-e^{-s})/2。

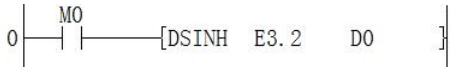
相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值<2^{-126}	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值>2^{128}	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围 。
4086H	(d)写入地址超出软元件范围 。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



计算 3.2 的双曲正弦值，得出结果：12.24588

DTAN/单精度实数 TAN 运算

DTAN(P)

对(s)中指定角度的 TAN(正切)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DTAN (s) (d)]

内容、范围、数据类型

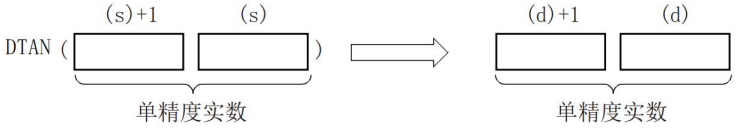
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 TAN(正切)运算的角度数据或存储了角度数据的软元件起始编号	0、$2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DTAN	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

对(s)中指定角度的 TAN(正切)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



对于(s)中指定的角度，以弧度单位(角度×π÷180)进行设置。

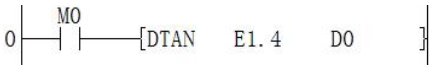
相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围 。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



计算 1.4 的正切值，得出结果：5.797883

DTANH/单精度实数 TANH 运算

DTANH(P)

对(s)中指定角度的 DTANH(双曲正切)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

-[DTANH (s) (d)]

内容、范围、数据类型

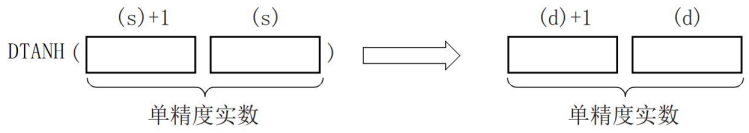
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 DTANH(双曲正切)运算的角度数据或存储了角度数据的软元件起始编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DTANH	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

对(s)中指定角度的 DTANH(双曲正切)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。



该指令是进行 2 进制浮点数取 TANH 值。其计算公式为 tanh 值=$(e^{s}-e^{-s})/(e^{s}+e^{-s})$。

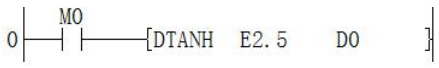
相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围 。
4086H	(d)写入地址超出软元件范围 。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



计算 2.5 的双曲正切值，得出结果 0.9866143

DDEG/单精度实数弧度→角度转换

DDEG(P)

将角度的大小单位从(s)中指定的弧度单位转换为度单位(DEG.单位)后，存储到(d)中指定的软元件编号中。

-[DDEG (s) (d)]

内容、范围、数据类型

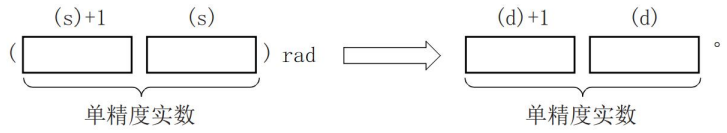
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行度单位转换的弧度角度或存储了弧度角度的软元件起始编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储进行了度单位转换的值的软元件起始编号	$-\pi/2 \sim +\pi/2$	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DDEG	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

• 将角度的大小单位从(s)中指定的弧度单位转换为度单位(DEG.单位)后，存储到(d)中指定的软元件编号中。



• 度单位→弧度单位转换按下述方式进行。

弧度单位 = 度单位* 180/π

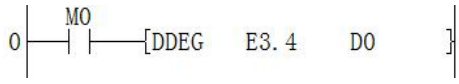
• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零(尾数部分为“0”时)	零标志(SM153)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



得出结果：194.8057

DRAD/单精度实数角度→弧度转换

DRAD(P)

将角度的大小单位从(s)中指定的度单位(DEG.单位)转换为弧度单位后，存储到(d)中指定的软元件编号中。

-[DRAD (s) (d)]

内容、范围、数据类型

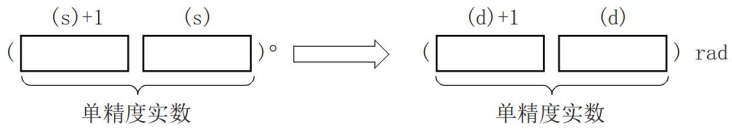
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行弧度单位转换的角度或存储了角度的软元件起始编号	0、2^{-126}≤|(s)|<2^{128}	单精度实数	ANYREAL_32	REAL
(d)	存储进行了弧度单位转换的值的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DRAD	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	

功能

将角度的大小单位从(s)中指定的度单位(DEG.单位)转换为弧度单位后，存储到(d)中指定的软元件编号中。



度单位→弧度单位

转换按下述方式进行。

弧度单位 = 度单位*π/180

• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值<2^{-126}	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例

得出结果：1.047197

DEADD/单精度实数加法运算

DEADD(P)

对(s1)中指定的单精度实数与(s2)中指定的单精度实数进行加法运算，将结果存储到(d)中指定的软元件中。

-[DEADD (s1) (s2) (d)]

内容、范围、数据类型

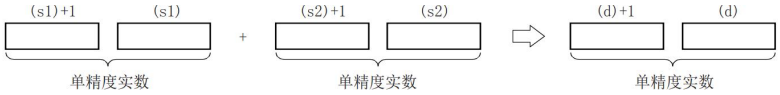
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	被加数据或存储了被加数据的软元件起始编号	0、$2^{-126}\leqslant\lvert(s)\rvert<2^{128}$	单精度实数	ANYREAL_32	REAL
(s2)	加法运算数据或存储了加法运算数据的软元件起始编号	0、$2^{-126}\leqslant\lvert(s)\rvert<2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEADD	参数 1															●	●	●	●	●	●	●	●	●	●	●	●
	参数 2															●	●	●	●	●	●	●	●	●	●	●	●
	参数 3															●	●	●	●	●	●	●				●	●

功能

• 对(s1)中指定的单精度实数与(s2)中指定的单精度实数进行加法运算，将加法运算结果存储到(d)中指定的软元件中。



• 指定了常数(K、H)到(s1)、(s2)的情况下，自动将值转换为单精度实数。

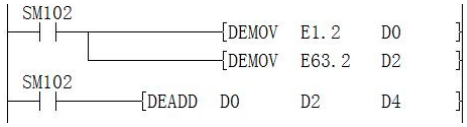
• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

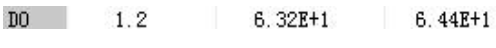
错误码

错误码	内容
4085H	(s1)(s2)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s1)(s2)指定软元件的内容为非正规数、非数、±∞时。

示例



得出计算结果：1.2 + 63.2 = 64.4



DESUB/单精度实数减法运算

DESUB(P)

对(s1)中指定的单精度实数与(s2)中指定的单精度实数进行减法运算，将结果存储到(d)中指定的软元件中。

-[DESUB (s1) (s2) (d)]

内容、范围、数据类型

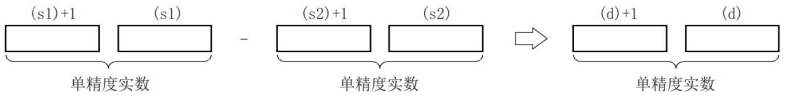
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	被减数据或存储了被减数据的软元件起始编号	0、$2^{-126}\leqslant\lvert(s)\rvert<2^{128}$	单精度实数	ANYREAL_32	REAL
(s2)	减法运算数据或存储了加法运算数据的软元件起始编号	0、$2^{-126}\leqslant\lvert(s)\rvert<2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DESUB	参数 1															●	●	●	●	●	●	●	●	●	●	●	●
	参数 2															●	●	●	●	●	●	●	●	●	●	●	●
	参数 3															●	●	●	●	●	●	●				●	●

功能

• 对(s1)中指定的单精度实数与(s2)中指定的单精度实数进行减法运算，将减法运算结果存储到(d)中指定的软元件中。



• 指定了常数(K、H)到(s1)、(s2)的情况下，自动将值转换为单精度实数。

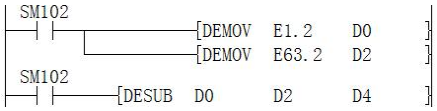
• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s1)(s2)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围 。
4084H	(s1)(s2)指定软元件的内容为非正规数、非数、±∞时。

示例



得出计算结果 1.2 - 63.2 = -62

福州富昌维控电子科技有限公司

DEMUL/单精度实数乘法运算

DEMUL(P)

对(s1)中指定的单精度实数与(s2)中指定的单精度实数进行乘法运算，将结果存储到(d)中指定的软元件中。

-[DEMUL (s1) (s2) (d)]

内容、范围、数据类型

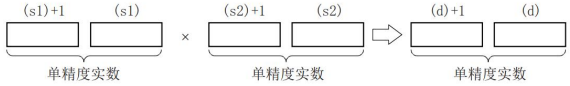
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	被乘数据或存储了被乘数据的软元件起始编号	0、$2^{-126}\leqslant\lvert(s)\rvert<2^{128}$	单精度实数	ANYREAL_32	REAL
(s2)	乘法运算数据或存储了乘法运算数据的软元件起始编号	0、$2^{-126}\leqslant\lvert(s)\rvert<2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEMUL	参数 1															●	●	●	●	●	●	●	●	●	●	●	●
	参数 2															●	●	●	●	●	●	●	●	●	●	●	●
	参数 3															●	●	●	●	●	●	●				●	●

功能

• 对(s1)中指定的单精度实数与(s2)中指定的单精度实数进行乘法运算，将乘法运算结果存储到(d)中指定的软元件中。



• 指定了常数(K、H)到(s1)、(s2)的情况下，自动将值转换为单精度实数。

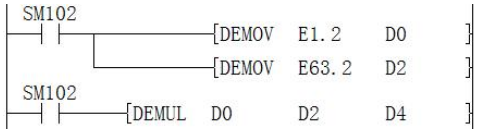
• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s1)(s2)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围 。
4084H	(s1)(s2)指定软元件的内容为非正规数、非数、±∞时。

示例



计算得出结果：1.2*63.2 = 75.84

软元件	+0	+1	+2
D0	1.2	6.32E+1	7.584E+1

DEDIV/单精度实数除法运算

DEDIV(P)

对(s1)中指定的单精度实数与(s2)中指定的单精度实数进行除法运算，将结果存储到(d)中指定的软元件中。

-[DEDIV (s1) (s2) (d)]

内容、范围、数据类型

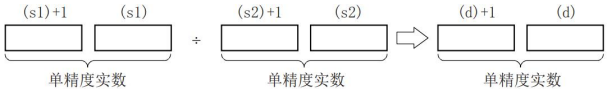
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	被除数据或存储了被除数据的软元件起始编号	0、2^{-126}≤|(s)|<2^{128}	单精度实数	ANYREAL_32	REAL
(s2)	除法运算数据或存储了除法运算数据的软元件起始编号	0、2^{-126}≤|(s)|<2^{128}	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEDIV	参数 1															●	●	●	●	●	●	●	●	●	●	●	●
	参数 2															●	●	●	●	●	●	●	●	●	●	●	●
	参数 3															●	●	●	●	●	●	●				●	●

功能

对(s1)中指定的单精度实数与(s2)中指定的单精度实数进行除法运算，将除法运算结果存储到(d)中指定的软元件中。



• 指定了常数(K、H)到(s1)、(s2)的情况下，自动将值转换为单精度实数。

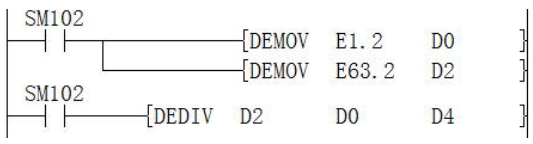
相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为ON。
SM152	借位	运算结果的绝对值<2^{-126}	(d)的值变为32位实数的最小值(2^{-126})，借位标志(SM152)变为ON。
SM151	进位	运算结果的绝对值>2^{128}	(d)的值变为32位实数的最大值(2^{128})，进位标志(SM151)变为ON。

错误码

错误码	内容
4085H	(s1)(s2)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s1)(s2)指定软元件的内容为非正规数、非数、±∞时。
4080H	(s2)值为0。

示例



得出计算结果：63.2 / 1.2 = 52.66666667

D0	1.2	6.32E+1	5.266666E+1

DEMOV/单精度实数数据传送

DEMOV(P)

将(s)中指定的软元件中存储的单精度实数数据，传送至(d)中指定的软元件。

-[DEMOV (s) (d)]

内容、范围、数据类型

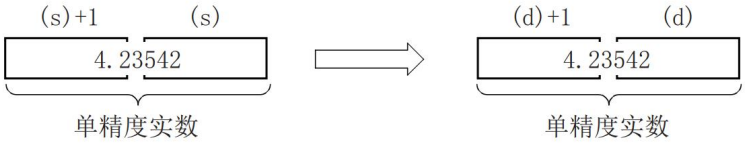
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	传送的数据或存储了传送数据的软元件起始编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储传送目标数据的软元件编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEMOV	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

将(s)中指定的软元件中存储的单精度实数数据，传送至(d)中指定的软元件。



错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围 。

示例

将 3.265 赋值到 R10

软元件	+0	+2
R0		
R8	0.000000E+000	3.256000E+00

DEBCD/2 进制浮点→10 进制浮点转换

DEBCD(P)

将(s)中指定的 2 进制浮点转换为 10 进制浮点后，存储到(d)中指定的软元件。

-[DEBCD (s) (d)]

内容、范围、数据类型

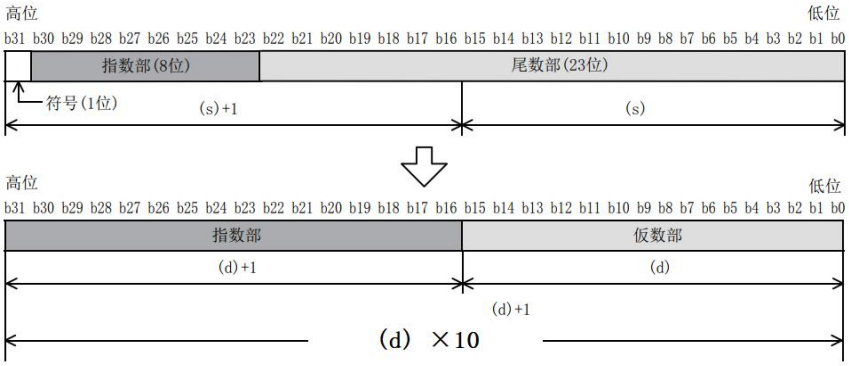
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了 2 进制浮点数据的软元件编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储转换后的 10 进制浮点数据的软元件编号	-	实数	ANY32	REAL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEBCD	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

将(s)中指定的 2 进制浮点转换为 10 进制浮点后，存储到(d)中指定的软元件。



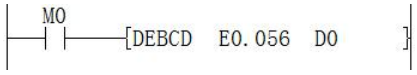
注意事项

浮点运算中全部以 2 进制浮点执行。但时，2 进制浮点是不易懂的数值(专用的监控方法)，因此通过将其转换为 10 进制浮点运算，方便外围设备进行监控等。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



得出结果：5600×10^{-5}

D0	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	0	5600
D1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	-5

DEBIN/10 进制浮点→2 进制浮点转换

DEBIN(P)

将(s)中指定的 10 进制浮点转换为 2 进制浮点后，存储到(d)中指定的软元件。

-[DEBIN　(s)　(d)]

内容、范围、数据类型

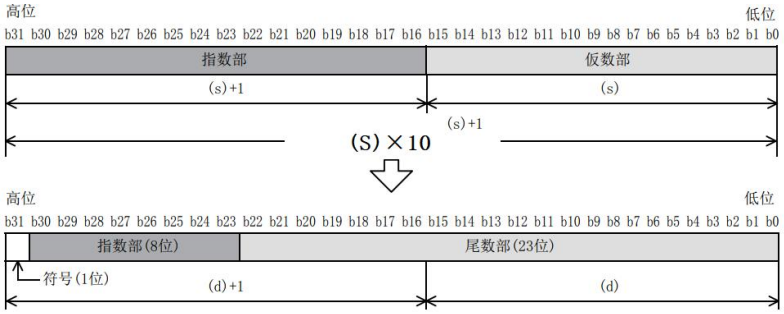
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了 10 进制浮点数据的软元件编号	-	实数	ANY32	REAL
(d)	存储转换后的 2 进制浮点数据的软元件编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEBIN	参数 1															●	●	●	●	●	●	●				●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

• 将(s)中指定的 10 进制浮点转换为 2 进制浮点后，存储到(d)中指定的软元件。



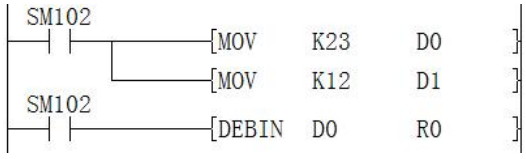
• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值<2^{-126}	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值>2^{128}	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。

示例



转换后得出结果：

R0	2.3E+13

DENEG/单精度实数符号取反

DENEG(P)

对(d)中指定的软元件的单精度实数的符号进行取反后，存储到(d)中指定的软元件中。

-[DEBEG (d)]

内容、范围、数据类型

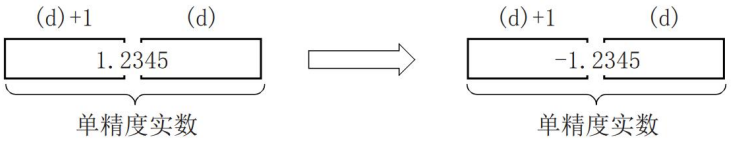
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储了进行符号取反的单精度实数的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DENEG	参数 1															●	●	●	●	●	●	●				●	●

功能

对(d)中指定的软元件的单精度实数的符号进行取反后，存储到(d)中指定的软元件中。

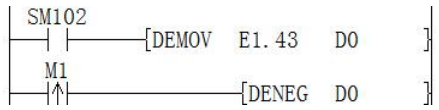


在对正负符号进行取反时使用。

错误码

错误码	内容
4086H	(d)写入地址超出软元件范围 。

示例



转换后变成-1.43

DECMP/单精度实数比较

DECMP(P)

比较 2 个数据(单精度实数)，将其大、小、一致结果输出到位软元件(3 点)。

-[DECMP (s1) (s2) (d)]

内容、范围、数据类型

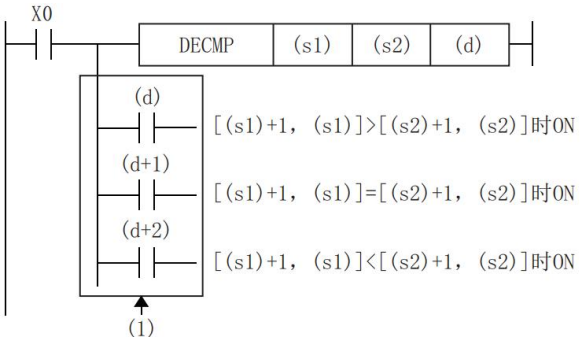
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件编号	0、$2^{-126} \leq \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(s2)	比较数据或存储了比较数据的软元件编号	0、$2^{-126} \leq \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	输出比较结果的起始位软元件编号(占用 3 点)	-	位	ANYBIT_ARRAY	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DECMP	参数 1															●	●	●	●	●	●	●	●	●	●	●	●
	参数 2															●	●	●	●	●	●	●	●	●	●	●	●
	参数 3		●	●	●	●					●															●	●

功能

将比较值(s1)和比较源(s2)作为浮点比较，根据其小、一致、大结果，(d)、(d)+1、(d)+2 中某个的位变为 ON。



(1) ：即使指令输入置为 OFF 而 DECMP 指令不执行，(d)～(d)+2 也保持 X0 置为 OFF 前的状态。

指定了常数(K、H)到(s1)、(s2)中指定的软元件的情况下，自动将值 BIN→单精度实数转换进行处理。

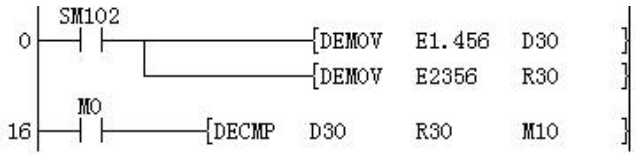
注意事项

(d)中指定的软元件占用 3 点[(d)，(d)+1，(d)+2]。请注意不要与用于其他用途的软元件重复。

错误码

错误码	内容
4085H	(s1)或(s2)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围 。
4084H	(s1)或(s2)指定软元件的内容为非正规数、非数、±∞时。

示例



M10	0
M11	0
M12	1

由于 R30 中的浮点数大于 D30 中的浮点数，所以 M12 置 ON。

DEZCP/2 进制浮点带宽比较

DEZCP(P)

比较高低 2 点的比较范围与数据(2 进制浮点)，将其大、小、带宽结果输出到位软元件(3 点)。

-[DEZCP (s1) (s2) (s3) (d)]

内容、范围、数据类型

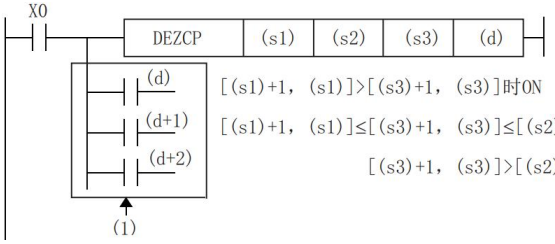
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(s2)	比较数据或存储了比较数据的软元件编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(s3)	比较数据或存储了比较数据的软元件编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	输出比较结果的起始位软元件编号(占用 3 点)	-	位	ANYBIT_ARRAY	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEZCP	参数 1															●	●	●	●	●	●	●	●	●	●	●	●
	参数 2															●	●	●	●	●	●	●	●	●	●	●	●
	参数 3															●	●	●	●	●	●	●	●	●	●	●	●
	参数 4		●	●	●	●					●															●	●

功能

将比较值(s1)、(s2)和比较源(s3)作为浮点比较，根据其小、范围内、大结果，(d)、(d)+1、(d)+2 中某个的位变为 ON。



(1)：即使指令输入置为 OFF 而 DEZCP 指令不执行，(d)～(d)+2 也保持 X0 置为 OFF 前的状态。

指定了常数(K、H)到(s1)、(s2)、(s3)中指定的软元件的情况下，自动将值 BIN→2 进制浮点转换进行处理。

注意事项

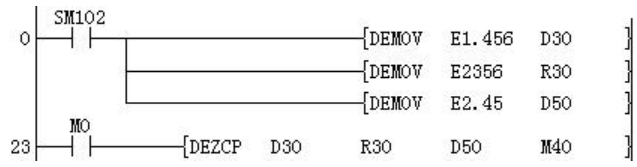
(d)中指定的软元件占用 3 点[(d)，(d)+1，(d)+2]。请注意不要与用于其他用途的软元件重复。

比较数据的大小关系请置为[(s1)+1，(s1)]≤[(s2)+1，(s2)]。[(s1)+1，(s1)]>[(s2)+1，(s2)]的情况下，视为[(s2)+1，(s2)]的值与[(s1)+1，(s1)]相同进行比较。

错误码

错误码	内容
4085H	(s1)、(s2)、(s3) 读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s1)、(s2)、(s3)指定软元件的内容为非正规数、非数、±∞时。

示例



由于 2.45 大于 1.456，同时 2.45 小于 2356，故 M41 置 ON

DESQR/单精度实数平方根

DESQR(P)

对(s)中指定的值的平方根进行运算后，将运算结果存储到(d)中指定的软元件中。

-[DESQR (s) (d)]

内容、范围、数据类型

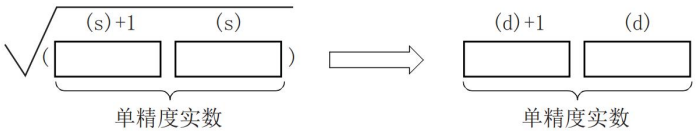
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行平方根运算的数据或存储了数据的软元件起始编号	0、$2^{-126} \leq \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DESQR	参数 1															●	●	●	●	●	●	●	●	●	●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

- 对(s)中指定的值的平方根进行运算后，将运算结果存储到(d)中指定的软元件编号中。



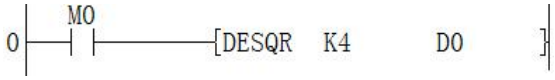
- (s)中指定的值只能设置为正的数。(不能以负的数进行运算。)
- 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。

错误码

错误码	内容
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。

示例



得出结果：D0 为浮点数 2。

软元件	+0	+2
D0	2.000000E+00	0.000000E+000

DESTR/单精度实数→字符串转换

DESTR(P)

将(s1)中指定的软元件中存储的单精度实数数据，按照(s2)中指定的软元件编号以后存储的显示指定转换为字符串后，存储到(d)中指定的软元件编号以后。

-[DESTR (s1) (s2) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	转换的单精度实数数据或存储了数据的软元件起始编号	0、$2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(s2)	存储了转换数值的显示指定的软元件起始编号(s2)中指定的软元件作为起始，用到(s2)+2	-	有符号BIN16位	ANY16_ARRAY（要素数：3）	--
(d)	存储转换后的字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DESTR	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2											●	●	●	●	●	●	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

功能

•将(s1)中指定的软元件中存储的单精度实数数据，按照(s2)中指定的软元件编号以后存储的显示指定转换为字符串后，存储到(d)中指定的软元件编号以后。也可直接指定实数到(s1)。

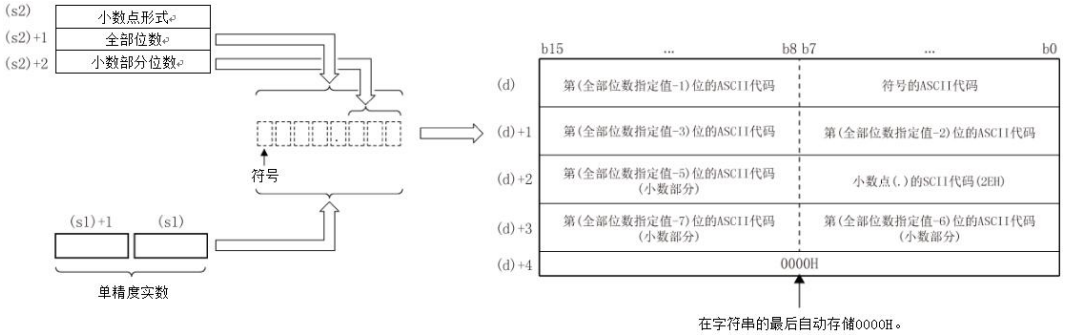
• 根据(s2)中指定的显示指定转换后的数据有所不同。

单元	功能
(s2)	0:小数点形式　　1:指数形式
(s2)+1	全部位数（字符串总数）。范围：2~24
(s2)+2	小数部分位数。范围 0~7

以上表格中的范围，会根据使用的转换形式等信息改变值的范围。

小数点形式

• (s2)中指定了 0 的情况下，将变为小数点形式。

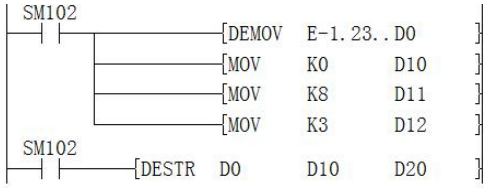


• 小数点形式下对应的位数范围：

单元	功能
(s2)	0:小数点形式
(s2)+1	全部位数（字符串总数）。范围：2~24。　(s2)+2 非 0 时：位数≥（小数部位数+3）。
(s2)+2	小数部分位数。范围 0-7。　(s2)+2 非 0 时：位数≥（小数部位数+3）。

1.例：全部位数为 8，小数部位数为 3，指定-1.235 的情况下，(d)以后将按以下方式存储。

• 字符串显示是为了方便查看以正常从左到右顺序显示字符串。



转换成的字符串：

D20	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	
D21	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	1
D22	0	1	1	1	0	1	0	0	0	1	0	0	1	1	0	0	.2
D23	1	1	0	0	1	1	0	0	1	0	1	0	1	1	0	0	35

对应的 ASCII 码为：

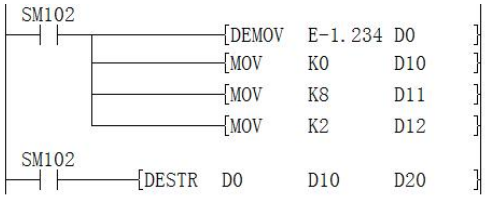
D20	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	2020	自动补充的空格
D21	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	3120	31H(1)20H(空格)
D22	0	1	1	1	0	1	0	0	0	1	0	0	1	1	0	0	322E	32H(2)2EH(.)
D23	1	1	0	0	1	1	0	0	1	0	1	0	1	1	0	0	3533	35H(5)33H(3)

• 第一个为符号位。在符号中，单精度实数数据为正时将存储 20H(空白)，为负时将存储 2DH(-)。

• 转换时如果实际的位数不足全部位数，将会在符号到第一个数字之间补充 20H(空白)

• 小数部位数的范围无法容纳单精度实数数据的小数部的情况下，低位小数部将被四舍五入。

2.例：全部位数为 8，小数部位数为 2，指定-1.234 的情况下，(d)以后将按以下方式存储。



转换成的字符串：

D20	1	0	1	1	0	1	0	0	0	0	0	0	0	1	0	0	-
D21	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	
D22	1	0	0	0	1	1	0	0	0	1	1	1	0	1	0	0	1.
D23	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	23

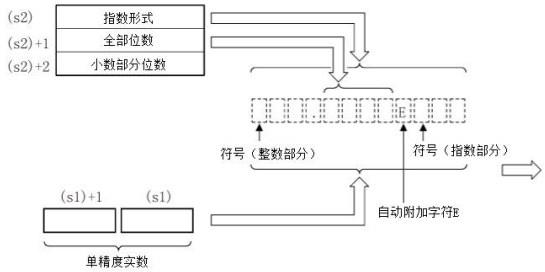
对应的 ASCII 码为：

D20	1	0	1	1	0	1	0	0	0	0	0	0	0	1	0	0	202D
D21	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	2020
D22	1	0	0	0	1	1	0	0	0	1	1	1	0	1	0	0	2E31
D23	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	3332
D24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

• 如上示例中：D20 的低字节存放负号 2DH(-)。然后由于数值的位数不足，D20 的高字节和 D21 都是 20H(空白)。最后 D22 到 D23 存放数值字符 1.23

指数形式

• (s2)中指定了 1 的情况下，将变为指数形式。



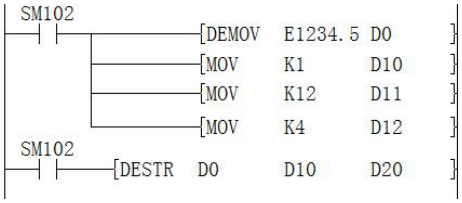
	b15 ... b8	b7 ... b0
(d)	第(全部位数指定值-1)位的ASCII代码	符号的ASCII代码(整数部分)
(d)+1	小数点(.)的SCII代码(2EH)	第(全部位数指定值-2)位的ASCII代码
(d)+2	第(全部位数指定值-5)位的ASCII代码(小数部分)	第(全部位数指定值-4)位的ASCII代码(小数部分)
(d)+3	第(全部位数指定值-7)位的ASCII代码	第(全部位数指定值-6)位的ASCII代码
(d)+4	符号的ASCII代码(指数部分)	45H(E)
(d)+5	第(全部位数指定值-11)位的ASCII代码(指数部分)	第(全部位数指定值-10)位的ASCII代码(指数部分)
(d)+6	0000H	

在字符串的最后自动存储 0000H →

• 指数形式下对应的位数范围：

单元	功能
(s2)	1:指数形式
(s2)+1	全部位数（字符串总数）。范围：2~24 。 (s2)+2 非 0 时：位数≥（小数部位数+7）
(s2)+2	小数部分位数。范围 0-7 (s2)+2 非 0 时：位数≥（小数部位数+7）

• 例 3，全部位数为 12，小数部位数为 4，指定 1234.5 的情况下，(d)以后将按以下方式存储。



转换成的字符串：

D20	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	
D21	1	0	0	0	1	1	0	0	0	1	1	1	0	1	0	0	1.
D22	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	23
D23	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	45
D24	1	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	E+
D25	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	03
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..

对应的 ASCII 码为：

D20	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	2020
D21	1	0	0	0	1	1	0	0	0	1	1	1	0	1	0	0	2E31
D22	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	3332
D23	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	3534
D24	1	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	2B45
D25	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	3330
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

• 整数部的符号中，单精度实数数据为正时将存储 20H(空白)，为负时将存储 2DH(-)。

• 整数部固定为 1 位数。在整数部与符号之间存储 20H(空白)。

• 小数部位数的范围无法容纳单精度实数数据的小数部的情况下，低位小数部将被四舍五入。

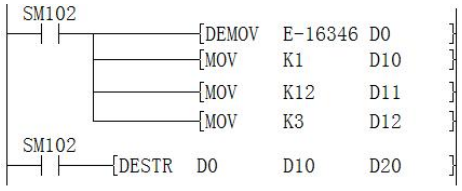
• 将小数部位数设置为 0 以外的情况下，第指定小数部位数+1 位数中将自动存储 2EH(.)。小数部位数为 0 时，不存储 2EH(.)。

• 指数部的符号中，指数为正时将存储 2BH(+)，为负时将存储 2DH(-)。

• 指数部固定为 2 位数。指数部为 1 位数的情况下，在指数部的符号之间将存储 30H(0)。

• 转换后的字符串的最后将自动存储 00H。

例 4：全部位数为 12，小数部位数为 3，指定-16346 的情况下，(d)以后将按以下方式存储。



转换成的字符串：

D20	1	0	1	1	0	1	0	0	0	0	0	0	0	1	0	0	-
D21	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	1
D22	0	1	1	1	0	1	0	0	0	1	1	0	1	1	0	0	.6
D23	1	1	0	0	1	1	0	0	1	0	1	0	1	1	0	0	35
D24	1	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	E+
D25	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	0	04
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..

对应的 ASCII 码为：

D20	1	0	1	1	0	1	0	0	0	0	0	0	0	1	0	0	202D
D21	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	3120
D22	0	1	1	1	0	1	0	0	0	1	1	0	1	1	0	0	362E
D23	1	1	0	0	1	1	0	0	1	0	1	0	1	1	0	0	3533
D24	1	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	2B45
D25	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	0	3430
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

• 如上示例中：

D20 的低字节存放负号 2DH(-)。

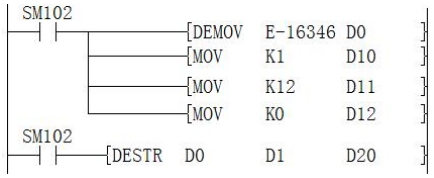
然后由于数值的位数不足，D20 的高字节和 D21 的低位都是 20H(空白)。

16346 变为字符串 1.635E+04，其中 16346 最后一位“6”被四舍五入。

指数部分为 34H(4)只有一位，这时在符号 2DH(-)和 34H(4)之间加上 30H(0)。

最后 D26 自动存储 00H

例 5：全部位数为 12，小数部位数为 0，指定-16346 的情况下，(d)以后将按以下方式存储。



转换成的字符串：

D20	1	0	1	1	0	1	0	0	0	0	0	0	0	1	0	0	-
D21	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	
D22	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	
D23	0	0	0	0	0	1	0	0	0	1	0	0	1	1	0	0	2
D24	1	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	E+
D25	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	0	04
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..

对应的 ASCII 码为：

D20	1	0	1	1	0	1	0	0	0	0	0	0	0	1	0	0	202D
D21	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	2020
D22	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	2020
D23	0	0	0	0	0	1	0	0	0	1	0	0	1	1	0	0	3220
D24	1	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	2B45
D25	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	0	3430
D26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

这个例子主要是可以看出如果小数位数设置 0 时，小数点 2EH(.)会自动省略。

注意事项

单精度实数在进行转换时位数越多越低的位数精度越差，可能会由于进度出现转换数值不准确的情况。

错误码

错误码	内容
4085H	(s1) (s2)读取地址超出软元件范围 。
4086H	(d)写入地址超出软元件范 。
4084H	(s1) (s2)指定软元件的内容为非正规数、非数、±∞时。
	(s2)中指定的形式指定为 0、1 以外时。
	(s1)+1 中指定的全部位数指定了超出 24 的值时。
	(s2)+2 中指定的小数部位数超出下述范围时。0~7
	小数点形式中，(s2)为 0 时。 ①小数部位数为 0 时：[(s2)+1]<2 ②小数部位数为 0 以外时：[(s2)+1]<(小数部位数+3)
	指数形式中，(s2)为 0 时。 ①小数部位数为 0 时：[(s2)+1]<6 ②小数部位数为 0 以外时：[(s2)+1]<(小数部位数+7)

DEVAL/字符串→单精度实数转换

DEVAL(P)

将(s)中指定的软元件编号以后中存储的字符串转换为单精度实数后，存储到(d)中指定的软元件中。

-[DEVAL (s) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行单精度实数转换的字符串数据或存储了字符串数据的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	存储转换后单精度实数的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

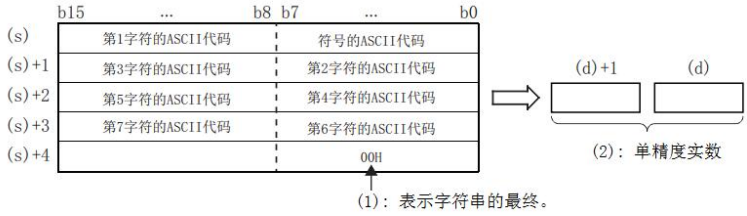
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEVAL	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

将(s)中指定的软元件编号以后中存储的字符串转换为单精度实数后，存储到(d)中指定的软元件中。

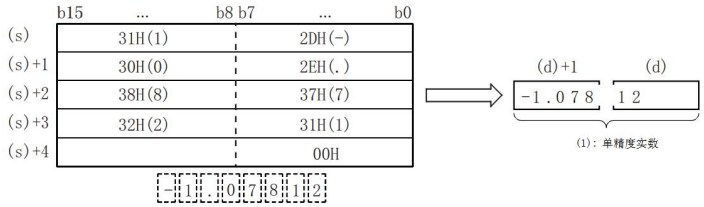
无论指定的字符串为小数点形式还是指数形式均可转换为单精度实数。

字符串最多可设置 24 字符。字符串中的 20H(空白)、30H(0)也作为 1 字符计数。

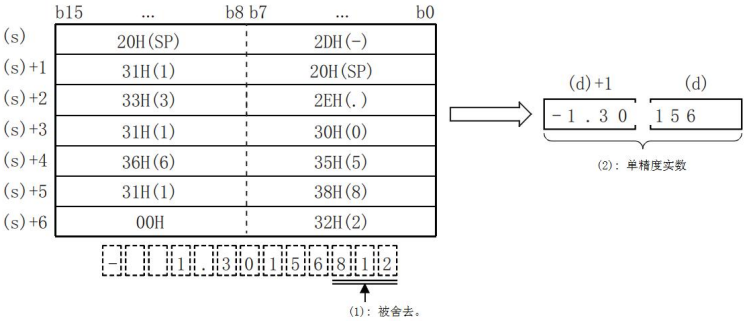


1. 小数点形式的情况下

①、(s)中指定的字符串为小数点形式时的的情况如下所示:

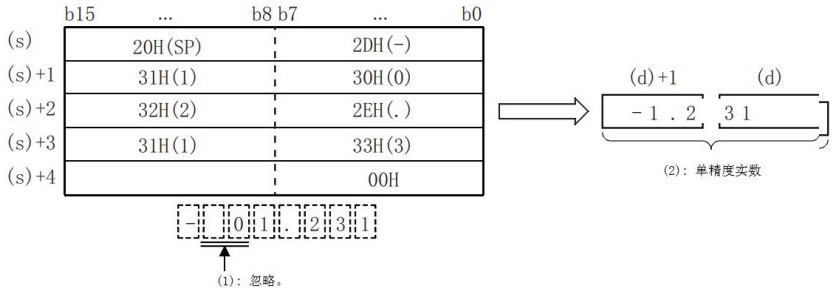


②、(s)中指定的字符串中，对于进行单精度实数转换的字符串，除去符号、小数点、指数部后的 6 位数有效，第 7 位数以后在转换时将被舍去。



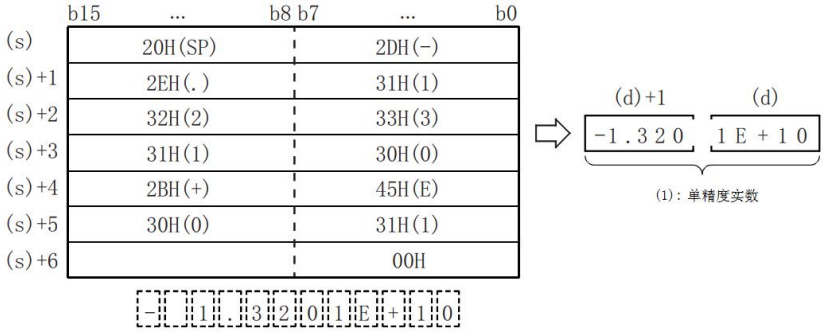
• 在小数点形式中将符号指定为 2BH(+)或省略符号时，将作为正值进行转换。此外，将符号指定为 2DH(-)时，将作为负值进行转换。

③、(s)中指定的字符串中，最初的 0 以外的数值之间存在有 20H(空白)或 30H(0)的情况下，转换时将忽略 20H、30H。

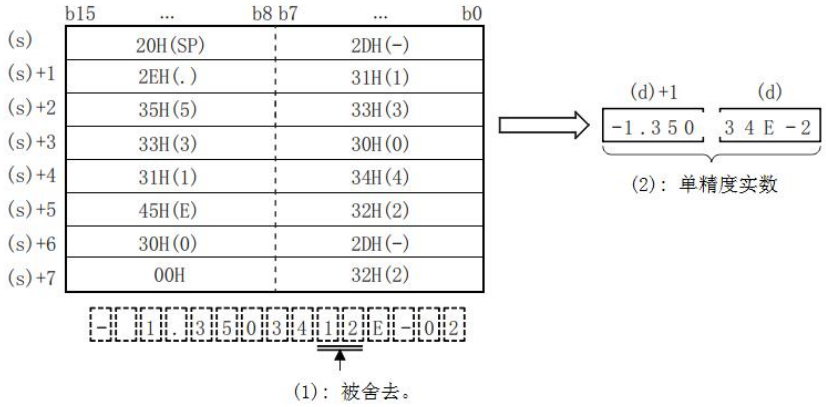


2. 指数形式的情况下

①、(s)中指定的字符串为指数形式的情况下，按下述方式执行：



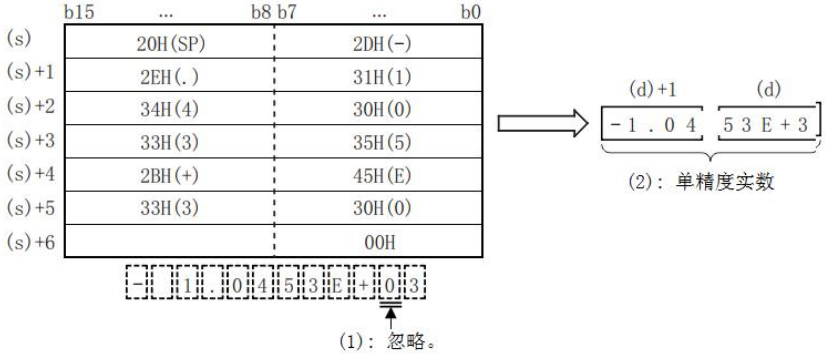
②、(s)中指定的字符串中，对于进行单精度实数转换的字符串，除去符号、小数点、指数部后的 6 位数有效，第 7 位数以后在转换时将被舍去。



在指数形式中将指数部的符号指定为 2BH(+)或省略符号时，将作为正值进行转换。将指数部的符号指定为 2DH(-)时，将作为负值进行转换。

③、(s)中指定的字符串中，最初的 0 以外的数值之间存在有 20H(空白)或 30H(0)的情况下，转换时将忽略 20H、30H。

在指数形式的字符串中，“E”与数值之间存储有 30H(0)的情况下，转换时将忽略 30H。



相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零。	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值<2^{-126}	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值>2^{-128}	(d)的值变为 32 位实数的最大值(2^{-128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
408AH	(s)未读到字符串，或字符串长度超出 24 个。
408BH	(s)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束
4084H	(s)中指定的字符串中出现 2BH (+)、2DH (-) 、20H(空格) 、2EH (.) 、45H (E)、65H(e) 、30 H(0)～ 39H (9) 以外的字符。
	(s)中指定的字符串中有 2 个以上的 2EH(.)时。
	(s)中指定的指数部分中存在有 45H(E)、2BH(+)、2DH(-)、30 H(0)～ 39H (9)以外的字符时或有多个指数部分时，或指数部分出现 2BH(+)或 2DH(-)两次及以上的情况。
	(s)中指定的字符串第一个数字字符之前出现了 2BH(+)或 2DH(-)两次及以上的情况。

示例

软元件	+0	+2	+4	+6
D0	5.246700E+12	0.000000E+000	0.000000E+000	0.000000E+000

D0 中存储的字符串为：5.2467E+12

得出的浮点数为：5.2467E+12

软元件	+0	+2	+4	+6
D96	0.000000E+000	0.000000E+000	0.000000E+000	0.000000E+000

DEXP/单精度实数指数运算

DEXP(P)

进行(s)中指定的值的指数运算后，将运算结果存储到(d)中指定的软元件中。

-[DEXP (s) (d)]

内容、范围、数据类型

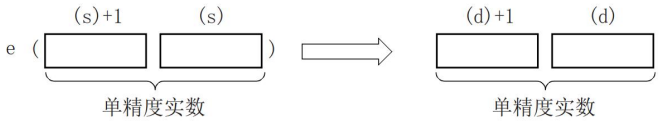
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行指数运算的数据或存储了数据的软元件起始编号	0、$2^{-126} \leq \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEXP	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

进行(s)中指定的值的指数运算后，将运算结果存储到(d)中指定的软元件编号中。



在指数运算中，将底(e)以“2.71828”进行运算。

相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例

计算得出结果：

INT/单精度实数→带符号 BIN16 位数据

INT(P)

将指定的单精度实数转换为带符号 BIN16 位数据。

-[INT (s) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	单精度实数或存储了单精度实数的起始软元件	[-32768,32767]	单精度实数	ANYREAL_32	REAL
(d)	存储 BIN 数据的软元件带符号	-	BIN16 位	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
INT	参数 1															●	●	●	●	●	●	●				●	●
	参数 2															●	●	●	●	●						●	●

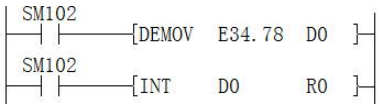
功能

- 将(s)中指定的单精度实数转换为带符号 BIN16 位数据后，存储到(d)中指定的软元件中。
- 对于转换后的数据，将变为(s)中指定的单精度实数的小数点以下第 1 位数被四舍五入后的值。
- 通过工程工具设置输入值的情况下，有可能会发生化整误差。
- 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	转换的时候有舍去小数位	转换的时候((s) - (d))> (2^{-126}) 借位(SM152) 变为 ON
SM151	进位	转换结果超出范围	(s)的值变超出范围[-32768,32767]或的(s)值小于 32 位实数的最小值(2^{-126})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。



得出转换结果：

软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
R0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	34

且借位表示置 ON

SM151	0
SM152	1
SM153	0

DINT/单精度实数→带符号 BIN32 位数据

DINT(P)

将指定的单精度实数转换为带符号 BIN32 位数据。

-[DINT (s) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	单精度实数或存储了单精度实数的起始软元件	[-2147483648,2147483647]	单精度实数	ANYREAL_32	REAL
(d)	存储 BIN 数据的起始软元件	-	带符号 BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DINT	参数 1															●	●	●	●	●	●	●				●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

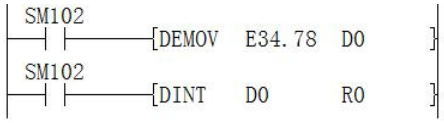
- 将(s)中指定的单精度实数转换为带符号 BIN32 位数据后，存储到(d)中指定的软元件中。
- 对于转换后的数据，将变为(s)中指定的单精度实数的小数点以下第 1 位数被四舍五入后的值。
- 通过工程工具设置输入值的情况下，有可能会发生化整误差。
- 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例



得出转换结果：

软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
R0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	34

且借位表示置 ON

SM151	0
SM152	1
SM153	0

DLOG10/单精度实数常用对数运算

DLOG10(P)

对(s)中指定的值的常用对数(以 10 为底的对数)进行运算，将运算结果存储到(d)中指定的软元件中。

-[DLOG10　(s)　(d)]

内容、范围、数据类型

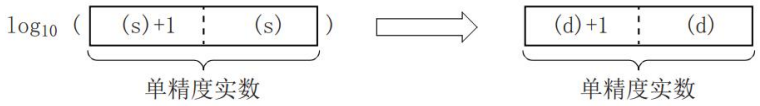
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行常用对数运算的数据或存储了数据的软元件起始编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DLOG10	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

• 对(s)中指定的值的常用对数(以 10 为底的对数)进行运算，将运算结果存储到(d)中指定的软元件编号中。



• (s)中指定的值只能设置为正的数。(不能以负的数进行运算。)

• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s) 读取地址超出软元件范围 。
4086H	(d)写入地址超出软元件范围 。
4084H	(s)指定软元件的内容为非正规数、非数、±∞时。

示例

得出计算结果

DLOGE/单精度实数自然对数运算

DLOGE(P)

对(s)中指定的值的自然对数 e 为底时的对数进行运算后，将运算结果存储到(d)中指定的软元件中。

-[DLOGE (s) (d)]

内容、范围、数据类型

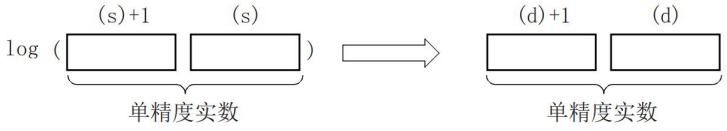
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行常用对数运算的数据或存储了数据的软元件起始编号	$0、2^{-126} \leq \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(d)	存储运算结果的软元件起始编号	-	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DLOGE	参数 1															●	●	●	●	●	●	●			●	●	●
	参数 2															●	●	●	●	●	●	●				●	●

功能

• 对(s)中指定的值的自然对数 e 为底时的对数进行运算后，将运算结果存储到(d)中指定的软元件编号中。



• (s)中指定的值只能设置为正的数。(不能以负的数进行运算。)

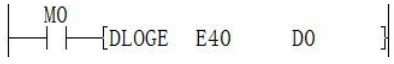
• 相关软元件如下所示。

软元件	名称	内容	
		条件	操作
SM153	零	运算结果真为零	零标志(SM153)变为 ON。
SM152	借位	运算结果的绝对值$<2^{-126}$	(d)的值变为 32 位实数的最小值(2^{-126})，借位标志(SM152)变为 ON。
SM151	进位	运算结果的绝对值$>2^{128}$	(d)的值变为 32 位实数的最大值(2^{128})，进位标志(SM151)变为 ON。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。
4084H	(s)指定软元件的内容为非正规数、非数、+∞时 、负数、0。

示例



得出结果：

7.10 触点比较指令

有符号 16 位触点比较指令

LD□、AND□、OR□

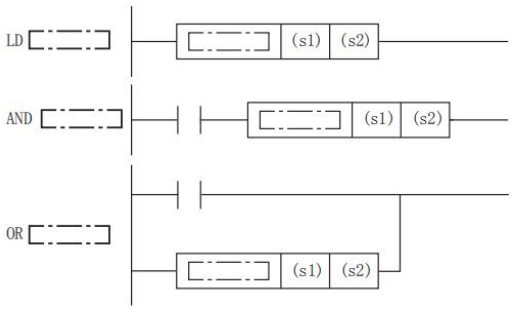
将(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据通过常开触点处理进行比较运算。

LD□：常开触点比较指令

AND□：常开触点串联连接比较指令

OR□：常开触点并联连接比较指令

梯形图



(“□”中可输入“=”，“<>”，“>”，“<”，“>=”，“<=”)

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(s2)	比较数据或存储了比较数据的软元件	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LD=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
LD>	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
LD<	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
LD>=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
LD<=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
LD<>	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
AND=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
AND>	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
AND<	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
AND>=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
AND<=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
AND<>	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
OR=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
OR>	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
OR<	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
OR>=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
OR<=	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	
OR<>	s1、s2											●	●	●	●	●	●	●	●	●			●	●		●	

功能

将(s1)中指定的软元件的 BIN16 位数据与(s2)中指定的软元件的 BIN16 位数据通过常开触点处理进行比较运算。

各指令的比较运算结果如下所示。

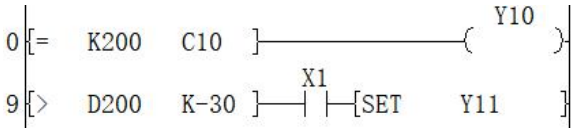
指令符号	条件	比较运算结果	指令符号	条件	比较运算结果
=	(s1)＝(s2)	导通状态	=	(s1)≠(s2)	非导通状态
<>	(s1)≠(s2)		<>	(s1)＝(s2)	
>	(s1)＞(s2)		>	(s1)≤(s2)	
<	(s1)＜(s2)		<	(s1)≥(s2)	
>=	(s1)≥(s2)		>=	(s1)＜(s2)	
<=	(s1)≤(s2)		<=	(s1)＞(s2)	

错误码

错误码	内容
4085H	(S)读取地址超出软元件范围。

示例

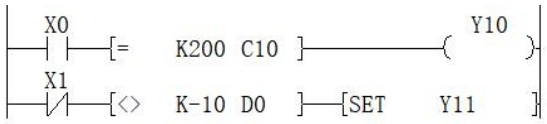
1. LD□指令：



计数器 C10 的当前值为 200 时，Y10 被置位

D200 的内容为-29 以上，且 X1 为 ON 时，Y11 被置位

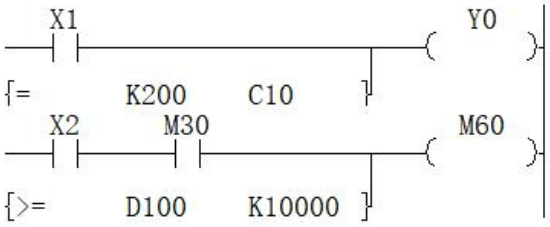
2. AND□指令：



X0 为 ON 时，或者是计数器 C10 的当前值为 200 时，Y10 被置位

X1 为 OFF，且数据计数器 D0 的内容不为-10，Y11 被置位

3. OR□指令：



X1 为 ON 时，或者是计数器 C10 的当前值为 200 时，Y0 被置位

X2 以及 M30 为 ON 时，或是寄存器 D100 的内容为 K10000 及以上时，M60 被置位

有符号 32 位触点比较指令

LDD□、ANDD□、ORD□

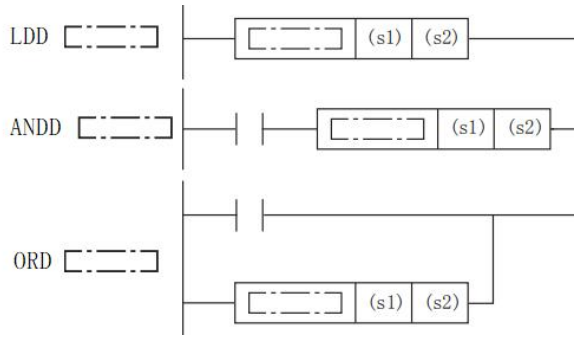
将(s1)中指定的软元件的 BIN32 位数据与(s2)中指定的软元件的 BIN32 位数据通过常开触点处理进行比较运算。

LDD□：常开触点比较指令

ANDD□：常开触点串联连接比较指令

ORD□：常开触点并联连接比较指令

梯形图



（“□”中可输入“D=”，“D<>”，“D>”，“D<”，“D>=”，“D<=”）

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-2147483648,2147483647]	有符号 BIN32 位	ANY32_S	DINT
(s2)	比较数据或存储了比较数据的软元件	[-2147483648,2147483647]	有符号 BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LDD=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
LDD>	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
LDD<	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
LDD>=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
LDD<=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
LDD<>	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ANDD=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ANDD>	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ANDD<	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ANDD>=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ANDD<=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ANDD<>	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ORD=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ORD>	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ORD<	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ORD>=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ORD<=	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	
ORD<>	s1、s2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	

功能

将(s1)中指定的软元件的 BIN32 位数据与(s2)中指定的软元件的 BIN32 位数据通过常开触点处理进行比较运算。

各指令的比较运算结果如下所示。

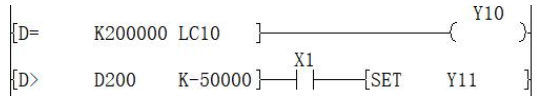
指令符号	条件	比较运算结果	指令符号	条件	比较运算结果
=	(s1)＝(s2)	导通状态	=	(s1)≠(s2)	非导通状态
<>	(s1)≠(s2)		<>	(s1)＝(s2)	
>	(s1)＞(s2)		>	(s1)≤(s2)	
<	(s1)＜(s2)		<	(s1)≥(s2)	
>=	(s1)≥(s2)		>=	(s1)＜(s2)	
<=	(s1)≤(s2)		<=	(s1)＞(s2)	

错误码

错误码	内容
4085H	(S)读取地址超出软元件范围。

示例

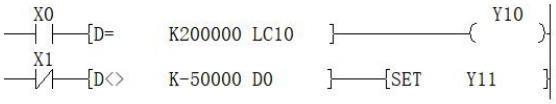
1. LDD□指令：



当 LC10 的数据为 200000 时，Y10 被置位，否则 Y10 被复位。

当 D201、D200 组成的 32 位数据数据超过-5000，并且 X1 为 ON，Y11 置 ON。

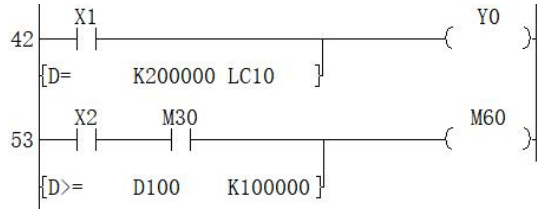
2. ANDD□指令：



当 X0 为 ON 并且 LC10 的值 为 200000 时，Y10 被置位，否则被复位。

当 X1 为 OFF 并且 D1、D0 组成的 32 位数据不等于 K-50000 时，Y11 被置位。

3. ORD□指令：



当 X1 为 ON，或者 LC10 的数据等于 LC10 的数据等于 200000 时，Y0 置位。

当 X2 并且 M30 置位，或者 D101，D100 组成的双字数据大于等于 100000 时，M60 置位。

单精度浮点触点比较指令

LDE□、ANDE□、ORE□

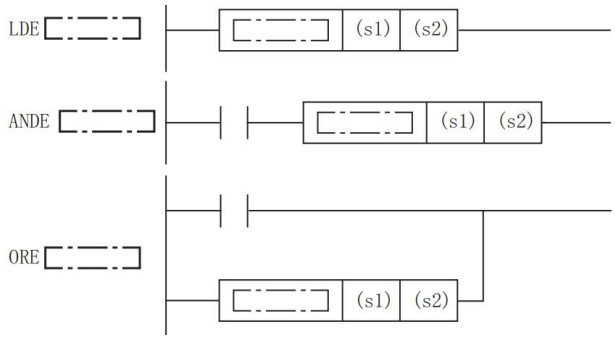
将(s1)中指定的软元件的单精度实数与(s2)中指定的软元件的单精度实数通过常开触点处理进行比较运算。

LDE□：常开触点比较指令

ANDE□：常开触点串联连接比较指令

ORE□：常开触点并联连接比较指令

梯形图



（“□”中可输入“=”，“<>”，“>”，“<”，“>=”，“<=”）

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件起始编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL
(s2)	比较数据或存储了比较数据的软元件起始编号	$0、2^{-126} \leqslant \lvert(s)\rvert < 2^{128}$	单精度实数	ANYREAL_32	REAL

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LDE=	s1、s2															●	●	●	●	●	●	●			●	●	
LDE>	s1、s2															●	●	●	●	●	●	●			●	●	
LDE<	s1、s2															●	●	●	●	●	●	●			●	●	
LDE>=	s1、s2															●	●	●	●	●	●	●			●	●	
LDE<=	s1、s2															●	●	●	●	●	●	●			●	●	
LDE<>	s1、s2															●	●	●	●	●	●	●			●	●	
ANDE=	s1、s2															●	●	●	●	●	●	●			●	●	
ANDE>	s1、s2															●	●	●	●	●	●	●			●	●	
ANDE<	s1、s2															●	●	●	●	●	●	●			●	●	
ANDE>=	s1、s2															●	●	●	●	●	●	●			●	●	
ANDE<=	s1、s2															●	●	●	●	●	●	●			●	●	
ANDE<>	s1、s2															●	●	●	●	●	●	●			●	●	
ORE=	s1、s2															●	●	●	●	●	●	●			●	●	
ORE>	s1、s2															●	●	●	●	●	●	●			●	●	
ORE<	s1、s2															●	●	●	●	●	●	●			●	●	
ORE>=	s1、s2															●	●	●	●	●	●	●			●	●	
ORE<=	s1、s2															●	●	●	●	●	●	●			●	●	
ORE<>	s1、s2															●	●	●	●	●	●	●			●	●	

功能

将(s1)中指定的软元件的单精度实数与(s2)中指定的软元件的单精度实数通过常开触点处理进行比较运算。

各指令的比较运算结果如下所示。

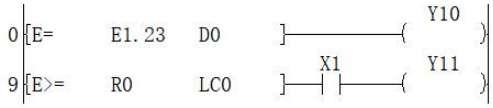
指令符号	条件	比较运算结果	指令符号	条件	比较运算结果
E=	(s1)＝(s2)	导通状态	E=	(s1)≠(s2)	非导通状态
E<>	(s1)≠(s2)		E<>	(s1)＝(s2)	
E>	(s1)＞(s2)		E>	(s1)≤(s2)	
E<	(s1)＜(s2)		E<	(s1)≥(s2)	
E>=	(s1)≥(s2)		E>=	(s1)＜(s2)	
E<=	(s1)≤(s2)		E<=	(s1)＞(s2)	

错误码

错误码	内容
4084H	(s1)(s2)指定软元件的内容为非正规数、非数、±∞时。
4085H	(s1)(s2)读取地址超出软元件范围。

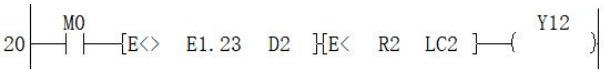
示例

1、LDE□指令：



- 当 D0 中输入的实数等于 E1.23 时，Y10 为 ON，否则 Y10 为 OFF。
- 当 R0 中实数大于等于 LC0 中的实数时，Y11 为 ON，否则为 OFF。
- 若 D0、R0、LC0 中的输入非实数，则会报 H4084 错误。

ANDE□指令：



仅当 M0 为 ON 并且 D2 实数不等于 E1.23 且 R2 实数小于 LC2 实数时，Y12 为 ON，否则都为 OFF。

ORE□指令：

R4 实数小于等于 R15 实数，或者 R6 实数等于 R20 实数时 Y13 为 ON，否则 Y13 为 OFF。

字符串比较

LDS□、ANDS□、 ORS□

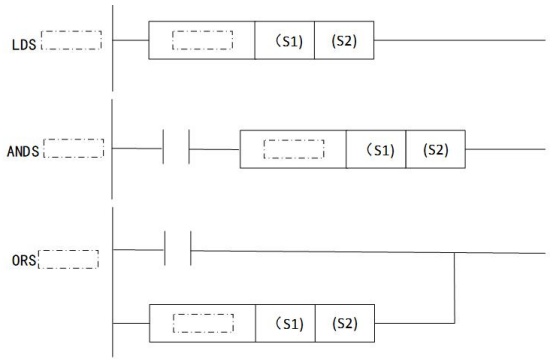
将(s1)中指定的软元件编号以后存储的字符串与(s2)中指定的软元件编号以后存储的字符串进行比较运算。

LDS□： 字符串比较指令

ANDS□： 字符串串联连接比较指令

ORS□：字符串并联连接比较指令

梯形图



（“□”中可输入“=”，“<>”）

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(S1)	连接数据或存储了数据的软元件起始编号或直接指定的字符串	-	字符串	ANYSTRING_SINGLE	--
(S2)	连接数据或存储了数据的软元件起始编号或直接指定的字符串	-	字符串	ANYSTRING_SINGLE	--

使用的软元件

指令	参数	软元件																									偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LDS=	s1、s2											●	●	●	●	●	●	●	●	●						●	
LDS<>	s1、s2											●	●	●	●	●	●	●	●	●						●	
ANDS=	s1、s2											●	●	●	●	●	●	●	●	●						●	
ANDS<>	s1、s2											●	●	●	●	●	●	●	●	●						●	
ORS=	s1、s2											●	●	●	●	●	●	●	●	●						●	
ORS<>	s1、s2											●	●	●	●	●	●	●	●	●						●	

功能

- 将(s1)中指定的软元件编号以后存储的字符串与(s2)中指定的软元件编号以后存储的字符串进行比较运算。
- 各指令的比较运算结果如下所示。

指令符号	条件	比较运算结果	指令符号	条件	比较运算结果
=	(s1)＝(s2)	导通状态	=	(s1)≠(s2)	非导通状态
<>	(s1)≠(s2)		<>	(s1)＝(s2)	

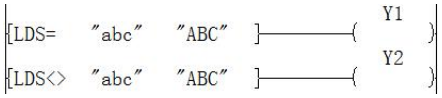
错误码

错误码	内容
4085H	(s1)或(s2)读取地址超出软元件范围。
408AH	(s1)或(s2)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s1)或(s2)读取字符串时读取到软元件最大范围但是没有找到 00H 作为结束。

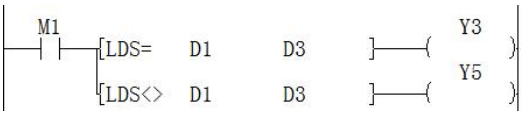
福州富昌维控电子科技有限公司

示例

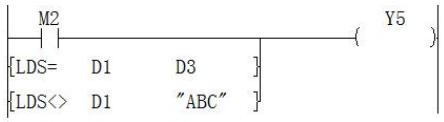
1、LDS□指令：



2、ANDS□指令：



3、ORS□指令：



7.11 时钟运算指令

TADD/时钟数据的加法运算

TADD(P)

将(s1)中指定的软元件编号以后存储的时刻数据与(s2)中指定的软元件编号以后存储的时间数据进行加法运算，将加法运算结果存储到(d)中指定的软元件编号以后。

-[TADD (s1) (s2) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储了被加时刻数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY(要素数:3)	--
(s2)	存储了加法运算时间(时刻)数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY(要素数:3)	--
(d)	存储加法运算结果时刻(时间)数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY(要素数:3)	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TADD	参数 1															●	●	●	●	●						●	●
	参数 2															●	●	●	●	●						●	●
	参数 3															●	●	●	●	●						●	●

功能

将(s1)中指定的时间数据与(s2)中指定的时间数据进行加法运算，将加法运算结果存储到(d)中指定的软元件编号以后。

(s1)	hour	(0～23)		(s2)	hour	(0～23)		(d)	hour	(0～23)
(s1)+1	minute	(0～59)	+	(s2)+1	minute	(0～59)	⇨	(d)+1	minute	(0～59)
(s1)+2	second	(0～59)		(s2)+2	second	(0～59)		(d)+2	second	(0～59)

【例】：将 6 时 32 分 40 秒与 7 时 48 分 10 秒进行了加法运算的情况下

(s1)	6		(s2)	7		(d)	14
(s1)+1	32	+	(s2)+1	48	⇨	(d)+1	20
(s1)+2	40		(s2)+2	10		(d)+2	50

运算结果的时间超过了 24 时的情况下，进位标志变为 ON，减去了 24 小时后的的值将成为运算结果。例如，将 14 时 20 分 30 秒与 20 时 20 分 20 秒进行了加法运算的情况下，其结果不是 34 时 40 分 50 秒，而是 10 时 40 分 50 秒。

(s1)	14		(s2)	20		(d)	10
(s1)+1	20	+	(s2)+1	20	⇨	(d)+1	40
(s1)+2	30		(s2)+2	20		(d)+2	50

运算结果为 0(0 时 0 分 0 秒)时，零标志变为 ON。

对 23 时 59 分 59 秒与 1 秒进行了加法运算的情况下，运算结果为 0 时 0 分 0 秒，进位标志、零标志变为 ON。

相关软元件如下所示：

软元件	名称	内容
SM151	进位	TADD(P)指令的结果超过了时钟数据的最大值 23 时 59 分 59 秒时 ON。
SM153	零	TADD(P)指令的结果为时刻是 0 时 0 分 0 秒时 ON。

注意事项

• (s1)、(s2)、(d)中指定的软元件分别占用 3 点。请注意不要与用于机械控制的软元件重复。

• 利用 CPU 模块内置实时时钟的时钟数据时刻(时、分、秒)的情况下，应使用 TRD(P)指令读取特殊寄存器的值后，将该字软元件指定到各操作数中。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s1)、(s2)中指定的数值在下述以外时。[0,23] (s1)+1、(s2)+1、(s1)+2、(s2)+2 中指定的数值在下述以外时。0~59

示例

设置 D0 时间为 16:30:00 ， D10 时间为 4:30:0

D0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	16
D1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	30
D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4
D11	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	30
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

线圈导通后，D20 时间为 21:0:0

D20	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	21
D21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TSUB/时钟数据的减法运算

TSUB(P)

将(s1)中指定的软元件编号以后存储的时刻数据与(s2)中指定的软元件编号以后存储的时间数据进行减法运算，将减法运算结果存储到(d)中指定的软元件编号以后。

-[TSUB　(s1)　(s2)　(d)]

内容、范围、数据类型

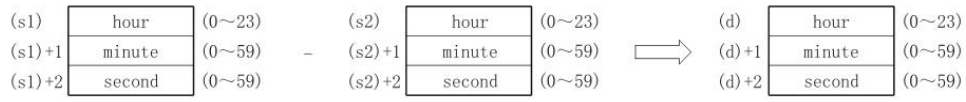
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储了被减时刻数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY(要素数:3)	--
(s2)	存储了减法运算时间(时刻)数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY(要素数:3)	--
(d)	存储减法运算结果时刻(时间)数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY(要素数:3)	--

使用的软元件

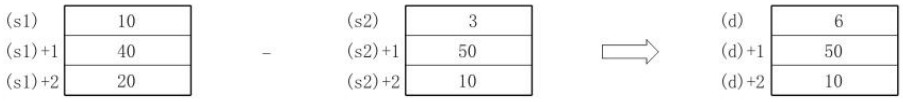
指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TSUB	参数 1															●	●	●	●	●						●	●
	参数 2															●	●	●	●	●						●	●
	参数 3															●	●	●	●	●						●	●

功能

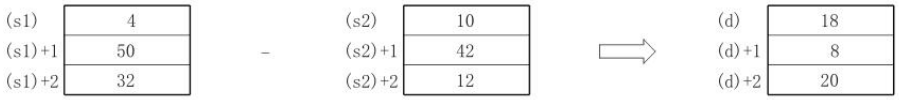
将(s1)中指定的时间数据与(s2)中指定的时间数据进行减法运算，将减法运算结果存储到(d)中指定的软元件编号以后。



【例】：将 10 时 40 分 20 秒中与 3 时 50 分 10 秒进行减法运算的情况下



运算结果的时间为负数的情况下，借位标志变为 ON，该数据+24 的值为运算结果。例如，将 4 时 50 分 32 秒与 10 时 42 分 12 秒进行减法运算的情况下，其结果不是-6 时 8 分 20 秒，而是 18 时 8 分 20 秒。



运算结果为 0(0 时 0 分 0 秒)时，零标志变为 ON。

相关软元件如下所示：

软元件	名称	内容
SM152	借位	TSUB(P)指令的结果为时刻小于 0 时 0 分 0 秒时 ON。
SM153	零	TSUB(P)指令的结果为时刻是 0 时 0 分 0 秒时 ON。

注意事项

- (s1)、(s2)、(d)中指定的软元件分别占用 3 点。请注意不要与用于机械控制的软元件重复。
- 利用 CPU 模块内置实时时钟的时钟数据时刻(时、分、秒)的情况下，应使用 TRD(P)指令读取特殊寄存器的值后，将该字软元件指定到各操作数中。

福州富昌维控电子科技有限公司

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s1)、(s2)中指定的数值在下述以外时。[0,23] (s1)+1、(s2)+1、(s1)+2、(s2)+2 中指定的数值在下述以外时。[0~59]

示例

设置 D0 时间为 16:30:00 ，D10 时间为 4:30:0。

D0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	16
D1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	30
D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

D10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4
D11	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	30
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

线圈导通后，D20 时间为 12:00:00。

D20	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	12
D21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TRD/时钟数据的读取

TRD(P)

读取 CPU 模块内置实时时钟的时钟数据。

-[TRD　(d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	时钟数据的读取目标、起始软元件编号	-	有符号 BIN16 位	ANY16_ARRAY（要素数：7）	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TRD	参数 1															●	●	●	●							●	●

功能

将 CPU 模块内置的实时时钟的时钟数据(SD100~SD106)按以下格式读取到(d)~(d)+6 中。

参数	要素	项目	时钟数据		要素	项目
特殊寄存器	SD105	年(公历)	[2000, 2099]	→	(d)	年(公历)
	SD104	月	[1, 12]	→	(d)+1	月
	SD103	日	[1, 31]	→	(d)+2	日
	SD102	时	[0, 23]	→	(d)+3	时
	SD101	分	[0, 59]	→	(d)+4	分
	SD100	秒	[0, 59]	→	(d)+5	秒
	SD106	星期	0(日)~6(六)	→	(d)+6	星期

- 相关软元件如下所示。这些特殊寄存器的时钟数据通过 END 处理更新。

软元件	内容	软元件	内容
SD100	时钟数据的秒数据以 BIN 码存储。	SD104	时钟数据的月数据以 BIN 码存储。
SD101	时钟数据的分数据以 BIN 码存储。	SD105	时钟数据的年数据以公历 4 位数的 BIN 码存储。
SD102	时钟数据的时数据以 BIN 码存储。	SD106	时钟数据的星期数据以 BIN 码存储。 (0：日、1：一、……、6：六)以 BIN 码存储。
SD103	时钟数据的日数据以 BIN 码存储。		

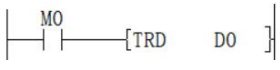
注意事项

- (d)中指定的软元件占用 7 点。请注意不要与用于机械控制的软元件重复。

错误码

错误码	内容
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



M0 线圈导通后读取当前日期和时间为 2020-2-19 13:10:38 星期三

软元件	+0	+1	+2	+3	+4	+5	+6
D0	2020	2	19	13	10	38	3

TWR/时钟数据的写入

TWR(P)

写入 CPU 模块内置实时时钟的时钟数据。

-[TWR (s)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	时钟数据的写入源、起始软元件编号	-	有符号 BIN16 位	ANY16_ARRAY（要素数：7）	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TWR	参数 1															●	●	●	●							●	●

功能

将设置的时钟数据(s)～(s)+6 写入到 CPU 模块内置的实时时钟的时钟数据(SD100~SD106)中。

时刻设置数据				特殊寄存器	
要素	项目	时钟数据		要素	项目
(s)	年(公历)	2000~2099 或 0~99	→	SD105	年(公历)
(s)+1	月	1~12	→	SD104	月
(s)+2	日	1~31	→	SD103	日
(s)+3	时	0~23	→	SD102	时
(s)+4	分	0~59	→	SD101	分
(s)+5	秒	0~59	→	SD100	秒
(s)+6	星期	0(日)~6(六)	→	SD106	星期

• 如果执行 TWR(P)指令，实时时钟的时钟数据立即更改。因此，应预先将几分钟后的时钟数据传送到设置的时钟数据(s)～(s)+6 中，到了正确时刻时执行指令。

• (s)中年份如果范围是 0~99 将被自动当成 2000~2099 年处理。

• 设置了表示不可能存在的时间的数值时，不会更新时钟数据。应设置正确的时钟数据后再次写入。

• 星期(SD100)自动校正。

• 相关软元件如下所示。

软元件	内容	软元件	内容
SD100	时钟数据的秒数据以 BIN 码存储。	SD104	时钟数据的月数据以 BIN 码存储。
SD101	时钟数据的分数据以 BIN 码存储。	SD105	时钟数据的年数据以公历 4 位数的 BIN 码存储。
SD102	时钟数据的时数据以 BIN 码存储。	SD106	时钟数据的星期数据以 BIN 码存储。 (0：日、1：一、……、6：六)以 BIN 码存储。
SD103	时钟数据的日数据以 BIN 码存储。		

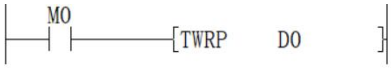
注意事项

(s)中指定的软元件占用 7 点。请注意不要与用于机械控制的软元件重复。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。

示例



提前将 D0 日期和时间设置为 2020-2-19 12:36:00。

软元件	+0	+1	+2	+3	+4	+5	+6
D0	2020	2	19	12	36	0	0

在时间 12:36:00 到来的那一刻，导通 M0 线圈，写入时间。

HTOS/时间数据的 16 位数据转换（时分秒→秒）

HTOS(P)

将(s)中指定的软元件编号以后存储的时间数据，换算为秒并将换算结果以 BIN16 位数据存储到(d)中指定的软元件中。

-[HTOS (s) (d)]

内容、范围、数据类型

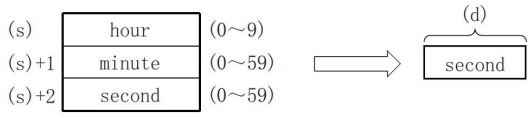
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了被减时刻数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY（要素数：3）	--
(d)	存储转换后时钟数据的软元件起始编号	-	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
HTOS	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

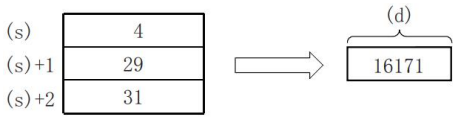
功能

将(s)中指定的软元件编号以后存储的时间数据，换算为秒并将换算结果存储到(d)中指定的软元件中。



例

(s)中指定了 4 小时 29 分 31 秒的情况下



错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	运算结果在 0~32767 以外时； (s)中指定的数值在 0~9 以外时； (s)+1、(s)+2 中指定的数值在 0~59 以外时。

示例

D0 时间设置为 5:36:53

D0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	5
D1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	36
D2	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	53

M0 线圈导通后 D10 的时间为

D10	1	0	1	0	1	1	1	1	0	1	1	1	0	0	1	0	20213

DHTOS/时间数据的 32 位数据转换（时分秒→秒）

DHTOS(P)

将(s)中指定的软元件编号以后存储的时间数据，换算为秒并将换算结果以 BIN32 位数据存储到(d)中指定的软元件中。

-[DHTOS (s) (d)]

内容、范围、数据类型

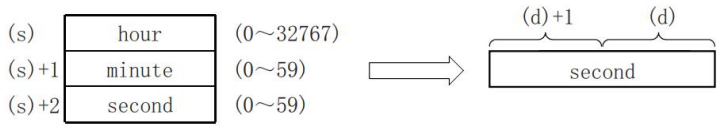
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了被减时刻数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY（要素数：3）	--
(d)	存储转换后时钟数据的软元件起始编号	-	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DHTOS	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

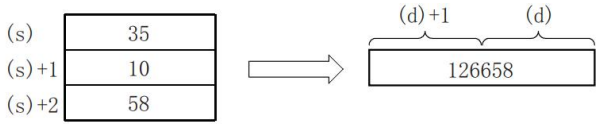
功能

将(s)中指定的软元件编号以后存储的时间数据，换算为秒并将换算结果存储到(d)中指定的软元件中。



【例】

(s)中指定了 35 小时 10 分 58 秒的情况下



错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s)中指定的数值在 0~32767 以外时； (s)+1、(s)+2 中指定的数值在 0~59 以外时。

示例

D0 时间设置为 15:33:24

软元件	+0	+1	+2
D0	15	33	24

M0 线圈导通后 D10 的秒为

软元件	+0	+2
D0	2162703	24
D8	0	56004

HOUR/小时计 16 位

HOUR(P)

将输入触点 ON 的时间以 1 小时为单位，进行加法运算计测。

-[HOUR (s) (d1) (d2)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	报警(d2)置为 ON 的时间(以 1 小时为单位设置)	K0~K32767	有符号 BIN16 位	ANY16	INT
(d1)	存储计测的当前值的软元件 (指定停电保持用数据寄存器)	-	无符号 BIN16 位	ANY16_ARRAY (要素数：2)	--
(d2)	时限到时置为 ON 的软元件(报警输出)	-	位	ANY_BOOL	BOOL

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
HOUR	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2																	●	●	●						●	●
	参数 3		●	●	●	●					●															●	●

功能

将输入触点 ON 的时间以 1 小时为单位，进行加法运算计测，在 ON 累计时间超过(s)中指定的时间(BIN16 位数据)时，将(d2)中指定的软元件置为 ON。

- (s)中以 1 小时为单位设置到报警(d2)置为 ON 为止的时间。
- (d1)中存储以 1 小时为单位的计测的当前值。(d1)中值如果超过 32767，会修改为 32767。
- (d1)+1 中存储不足 1 小时的计测的当前值(1 秒单位)。
- (d2)在当前值(d1)超过(s)中指定的时间时变为 ON。
- 为在 CPU 模块的电源 OFF 后也继续使用当前值数据，应将停电保持用数据寄存器指定到(d1)中。如果使用一般用的数据寄存器，通过 CPU 模块的电源 OFF 及 STOP→RUN 操作，当前值数据将被清除。
- (d2)中指定的报警输出变为 ON 后，计测也将继续进行。
- 当前值达到 16 位的最大值时计测停止。继续计测的情况下，应清除(d1)~(d1)+1 的当前值。

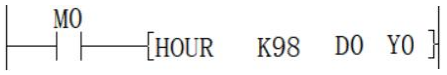
注意事项

- (d1)中指定的软元件占用 2 点。请注意不要与用于机械控制的软元件重复。
- 指令停止运行后，计测停止，输出继续保持。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s)的值为负时。

示例



当 M0＝ON 时，累计该状态的持续时间，将时记录在 D0 中，将不满 1 小时的秒记录在 D1 中，当 D0 累计时间达到 98 小时后，Y0 输出状态为 ON。计时条件满足时，到达指定数值(K98)后，累计计时仍继续进行，读数会继续增大；现在时间值 D0 到达最大数值 32767 小时、D1 达到 3599 秒时会停止计时测量，要重新计时须将现在时间值 D0,D1 清除为 0。

DHOUR/小时计 32 位

DHOUR(P)

将输入触点 ON 的时间以 1 小时为单位，进行加法运算计测。

-[DHOUR (s) (d1) (d2)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	报警(d2)置为 ON 的时间(以 1 小时为单位设置)	[0,2147483647]	有符号BIN32位	ANY32	DINT
(d1)	存储计测的当前值的软元件 (指定停电保持用数据寄存器)	-	无符号BIN32位	ANY32_ARRAY (要素数：2)	--
(d2)	时限到时置为 ON 的软元件(报警输出)	-	位	ANY_BOOL	BOOL

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DHOUR	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2																	●	●	●						●	●
	参数 3		●	●	●	●					●															●	●

功能

•将输入触点 ON 的时间以 1 小时为单位，进行加法运算计测，在 ON 累计时间超过(s)中指定的时间(BIN32 位数据)时，将(d2)中指定的软元件置为 ON。

• (s)+1、(s)中以 1 小时为单位设置到报警(d2)置为 ON 为止的时间。

• (d1)+1、(d1)中存储以 1 小时为单位的计测的当前值。((d1)+1：高位、(d1)：低位)

• (d1)+1、(d1)中值如果超过 2147483647，会修改为 2147483647。

• (d1)+2 中存储不足 1 小时的计测的当前值(1 秒单位)。

• (d2)在当前值(d1)+1、(d1)超过(s)中指定的时间时变为 ON。

• 为在 CPU 模块的电源 OFF 后也继续使用当前值数据，应将停电保持用数据寄存器指定到(d1)中。如果使用一般用的数据寄存器，通过 CPU 模块的电源 OFF 及 STOP→RUN 操作，当前值数据将被清除。

• (d2)中指定的报警输出变为 ON 后，计测也将继续进行。

• 当前值达到 32 位的最大值时计测停止。继续计测的情况下，应清除(d1)～(d1)+2 的当前值。

注意事项

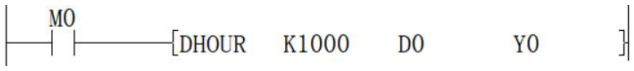
• (d1)中指定的软元件占用 3 点。请注意不要与用于机械控制的软元件重复。

• 指令停止运行后，计测停止，输出继续保持。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s)的值为负时。

示例



当 M0＝ON 时，累计该状态的持续时间，将时记录在 D1、D0 中，将不满 1 小时的秒记录在 D2 中，当 D1、D0 累计时间达到 1000 小时后，Y0 输出状态为 ON。计时条件满足时，到达指定数值(K1000)后，累计计时仍继续进行，读数会继续增大；现在时间值 D1、D0 到达最大数值 2147483647 小时、D2 达到 3599 秒时会停止计时测量，要重新计时须将现在时间值 D0,D1,D2 清除为 0。

STOH/时间数据的 16 位数据转换（秒→时分秒）

STOH(P)

将(s)中指定的软元件编号中存储的秒 16 位数据，换算为时、分、秒，将换算结果存储到(d)中指定的软元件以后。

-[STOH (s) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了转换前时钟数据的软元件起始编号	[0,32767]	有符号 BIN16 位	ANY16	INT
(d)	存储转换后时钟数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY（要素数：3）	--

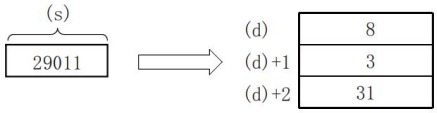
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
STOH	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

将(s)中指定的软元件编号以后存储的秒数据，换算为时、分、秒，将换算结果存储到(d)中指定的软元件以后。

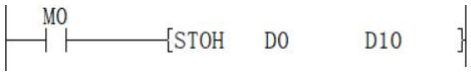
【例】：(s)中指定了 29011 秒的情况下



错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s)的数据超出范围时。

示例



将 D0 秒数的设置为 12537。

M0 线圈导通后 D10 的时、分、秒为

D10	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
D11	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	28
D12	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	57

DSTOH/时间数据的 32 位数据转换（秒→时分秒）

DSTOH(P)

将(s)中指定的软元件编号中存储的秒 32 位数据，换算为时、分、秒，将换算结果存储到(d)中指定的软元件以后。

-[DSTOH (s) (d)]

内容、范围、数据类型

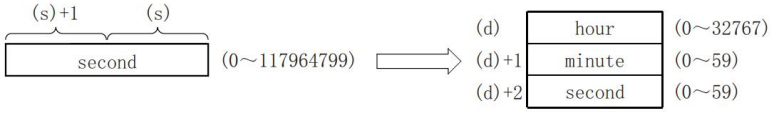
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了转换前时钟数据的软元件起始编号	[0, 117964799]	有符号 BIN32 位	ANY32	DINT
(d)	存储转换后时钟数据的软元件起始编号	-	有符号 BIN16 位	ANY16_ARRAY(要素数：3)	--

使用的软元件

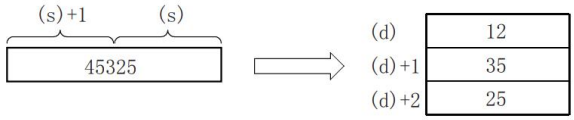
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSTOH	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

将(s)中指定的软元件编号以后存储的秒数据，换算为时、分、秒，将换算结果存储到(d)中指定的软元件以后。



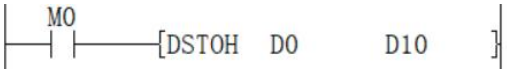
【例】：(s)中指定了 45325 秒的情况下



错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s)的数据超出范围时。

示例



将 D0 秒数的设置为 2152537。

M0 线圈导通后 D10 的时、分、秒为

D10	1	0	1	0	1	0	1	0	0	1	0	0	0	0	0	0	597
D11	1	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	55
D12	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	37

TCMP/时钟数据比较

TCMP(P)

将(s1)、(s2)、(s3)中指定的比较时刻与(s4)中指定的时刻数据进行比较，根据其大小一致情况将(d)中指定的位软元件置为 ON/OFF。

-[TCMP (s1) (s2) (s3) (s4) (d)]

内容、范围、数据类型

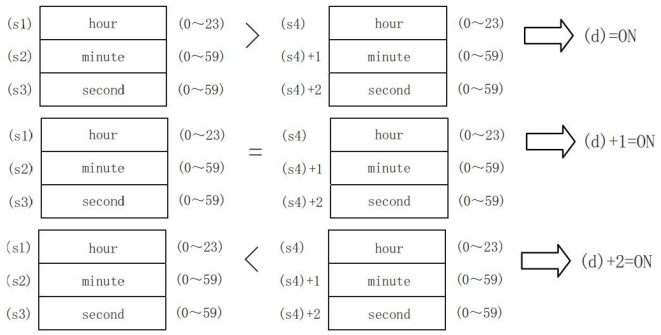
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	指定比较基准时间的“时”。	[0,23]	有符号 BIN16 位	ANY16	INT
(s2)	指定比较基准时间的“分”。	[0,59]	有符号 BIN16 位	ANY16	INT
(s3)	指定比较基准时间的“秒”。	[0,59]	有符号 BIN16 位	ANY16	INT
(s4)	指定时间数据(时、分、秒)的“时”。	-	有符号 BIN16 位	ANY16_ARRAY	--
(d)	根据比较结果 ON/OFF 位软元件。	-	位	ANYBIT_ARRAY	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TCMP	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 4															●	●	●	●	●						●	●
	参数 5		●	●	●	●					●															●	●

功能

将比较基准时间(时、分、秒)[(s1)，(s2)，(s3)]的时间与时间数据(时、分、秒)[(s4)，(s4)+1，(s4)+2]进行大小比较，根据其大小一致的结果 ON/OFF (d)开始的 3 点。



注意事项

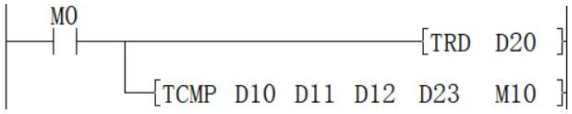
• (s4)、(d)中指定的软元件占用 3 点。请注意不要与用于机械控制的软元件重复。

• 利用 CPU 模块内置实时时钟的时钟数据时刻(时、分、秒)的情况下，应使用 TRD(P)指令读取特殊寄存器的值后，将该字软元件指定到各操作数中。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时 。
4084H	(s1)、(s4)中指定的数值超出[0,23]范围时； (s2)、(s3)、(s4)+1、(s4)+2 中指定的数值超出 0~59 范围时。

示例



设置 D10 为 1，D11 为 30，D12 为 0。

当 M0 导通后，D23 都会来的时间为 0:31:27

D23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D24	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	31
D25	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	27

M10 被置 ON

M10	1
M11	0
M12	0

TZCP/时钟数据带宽比较

TZCP(P)

将(s1)、(s2)中指定的高低 2 点的比较时刻与(s3)中指定的时刻数据进行比较，根据其大小带宽情况将(d)中指定的位软元件置为 ON/OFF。

-[TZCP (s1) (s2) (s3) (d)]

内容、范围、数据类型

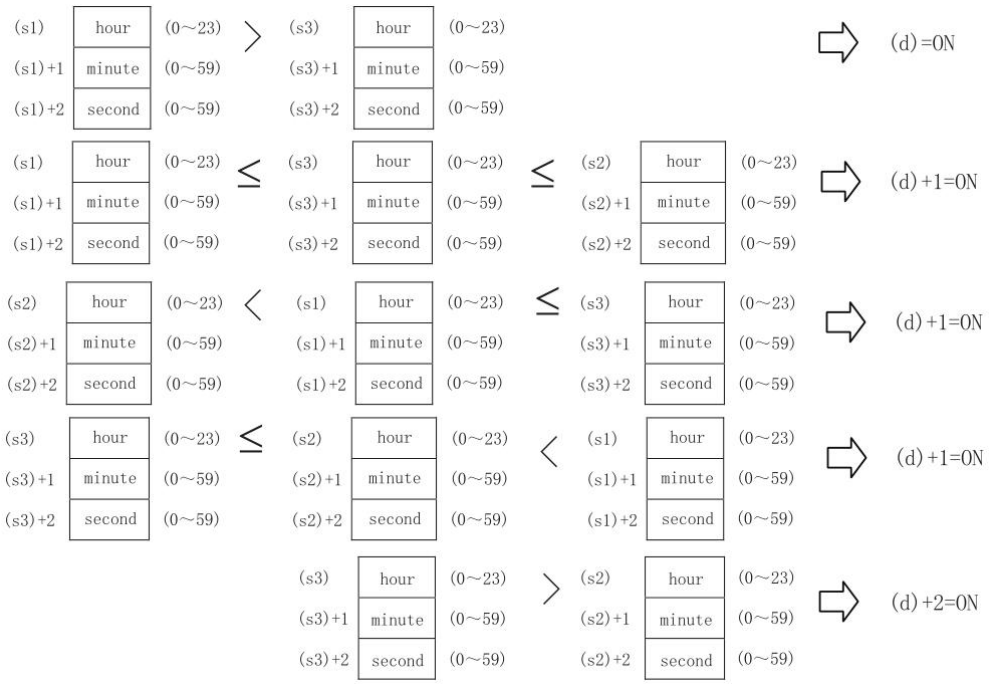
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	指定比较下限时刻(时、分、秒)的“时”	-	有符号 BIN16 位	ANY16_ARRAY（要素数：3）	--
(s2)	指定比较下限时刻(时、分、秒)的“时”	-	有符号 BIN16 位	ANY16_ARRAY（要素数：3）	--
(s3)	指定时刻数据(时、分、秒)的“时”	-	有符号 BIN16 位	ANY16_ARRAY（要素数：3）	--
(d)	根据比较结果 ON/OFF 位软元件。	-	位	ANYBIT_ARRAY（要素数：3）	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TZCP	参数 1															●	●	●	●	●						●	●
	参数 2															●	●	●	●	●						●	●
	参数 3															●	●	●	●	●						●	●
	参数 4		●	●	●	●					●															●	●

功能

将(s1)、(s2)中指定的高低 2 点的比较时刻与(s3)中指定的时刻数据进行比较，根据其大小带宽情况将(d)中指定的位软元件置为 ON/OFF。



注意事项

- (s1)、(s2)、(s3)、(d)中指定的软元件占用 3 点。请注意不要与用于机械控制的软元件重复。
- 利用 CPU 模块内置实时时钟的时钟数据时刻(时、分、秒)的情况下，应使用 TRD(P)指令读取特殊寄存器的值后，将该字软元件指定到各操作数中。
- 当(s1) > (s2)时，(d)、(d)+1、(d)+2 中有两个位置为 ON/OFF。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s1)、(s2)、(s3)中指定的数值超出[0,23]范围时； (s1)+1、(s2)+1、(s3)+1、(s1)+2、(s2)+2、(s3)+2 中指定的数值超出 0~59 范围时。

示例

设置 D0 时间为 16:30:00 ， D10 时间为 4:30:0

D0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	16
D1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	30
D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	4
D11	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	30
D12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

线圈导通后，读取时间到 D20 时间为 8:30:00

D20	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	8
D21	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	30
D22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

M10/M12 被置 ON。

M10	1
M11	0
M12	1

7.12 数据控制指令

BAND/BIN16 位数据死区控制

BAND(P)

(s3)中指定的输入值(BIN16 位值)根据(s1)、(s2)中指定的死区的上下限范围，对(d)中指定的软元件中存储的输出值进行控制。

-[BAND (s1) (s2) (S3) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	死区(无输出区域)的下限值	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(s2)	死区(无输出区域)的上限值	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(S3)	通过死区控制进行控制的输入值	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(D)	存储通过死区控制进行控制的输出值的软元件起始编号	-	有符号 BIN16 位	ANY16_S	INT

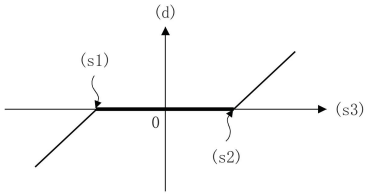
使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BAND	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3											●	●	●	●	●	●	●	●	●						●	●
	参数 4												●	●	●	●	●	●	●	●						●	●

功能

(s3)中指定的输入值(BIN16 位值)根据(s1)、(s2)中指定的死区的上下限范围，对(d)中指定的软元件中存储的输出值进行控制。输出值按下述方式被控制。

条件	输出值中存储的值
死区下限值(s1)>输入值(s3)时	输入值(s3)-死区下限值(s1)
死区上限值(s1)<输入值(s3)时	输入值(s3)-死区上限值(s2)
死区下限值(s1)≤输入值(s3)≤死区上限值(s2)时	0



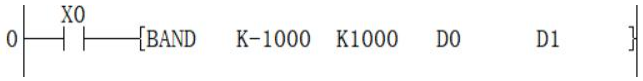
- (d)中存储的输出值为带符号 BIN16 位值的情况下，运算结果超出[-32768,32767]的范围时，其情况如下例所示。

例：(s1)为 10，(s3)为-32768 时，则输出值=-32768-10=8000H-000AH=7FFFH=32758。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	(s1)中指定的下限值大于(s2)中指定的上限值时。

示例



当 X000 为 ON 时

- D0＜(-1000)时，D1 中保存(D0)-(-1000)的值。
- -1000≤D0≤1000 时，D1 中保存 0。
- 1000＜D0 时，D1 中保存(D0)-1000 的值。

DBAND/BIN32 位数据死区控制

DBAND(P)

(s3)中指定的输入值(BIN132 位值)根据(s1)、(s2)中指定的死区的上下限范围，对(d)中指定的软元件中存储的输出值进行控制。

-[DBAND (s1) (s2) (S3) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	死区(无输出区域)的下限值	[-2147483648,2147483647]	有符号 BIN32 位	ANY32_S	DINT
(s2)	死区(无输出区域)的上限值	[-2147483648,2147483647]	有符号 BIN32 位	ANY32_S	DINT
(S3)	通过死区控制进行控制的输入值	[-2147483648,2147483647]	有符号 BIN32 位	ANY32_S	DINT
(d)	存储通过死区控制进行控制的输出值的软元件起始编号	-	有符号 BIN32 位	ANY32_S	DINT

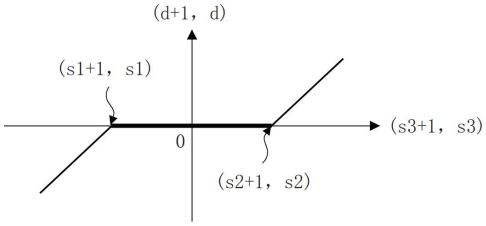
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBAND	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3											●	●	●	●	●	●	●	●	●	●	●				●	●
	参数 4												●	●	●	●	●	●	●	●	●	●				●	●

功能

(s3)中指定的输入值(BIN32 位值)根据(s1)、(s2)中指定的死区的上下限范围，对(d)中指定的软元件中存储的输出值进行控制。输出值按下述方式被控制。

条件	输出值中存储的值
死区下限值((s1)、(s1)+1)>输入值((s3)、(s3)+1)时	输入值((s3)、(s3)+1)-死区下限值((s1)、(s1)+1)
死区上限值((s1)、(s1)+1)<输入值((s3)、(s3)+1)时	输入值((s3)、(s3)+1)-死区上限值((s2)、(s2)+1)
死区下限值((s1)、(s1)+1)≤输入值((s3)、(s3)+1)≤死区上限值((s2)、(s2)+1)时	0



• (d)中存储的输出值为带符号 BIN32 位值的情况下，运算结果超出-2147483648~2147483647 的范围时，其情况如下例所示：(s1)、(s1)+1 为 1000，(s3)、(s3)+1 为-2147483648 时，则输出值=-2147483648-1000=80000000H-000003E8H=7FFFFC18H=2147482648。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时
4086H	写入指定的软元件范围超出相应软元件的范围时
4084H	(s1)中指定的下限值大于(s2)中指定的上限值时。

示例

• (D1、D0)＜(-10000)时，(D11、D10)中保存(D1、D0)-(-10000) 的值。
• -10000≤(D1、D0)≤10000 时，(D11、D10)中保存 0。
• 10000＜(D1、D0)时，(D11、D10)中保存(D1、D0)-10000 的值。

福州富昌维控电子科技有限公司

BINDA/BIN16 位数据→10 进制 ASCII 转换

BINDA(P)

将(s)中指定的 BIN16 位数据、10 进制数表示时的各位的数值转换为 ASCII 码后，存储到(d)中指定的软元件编号以后。

-[BINDA（s） （d）]

内容、范围、数据类型

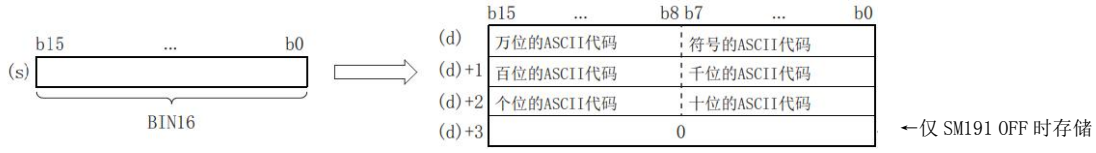
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 ASCII 转换的 BIN 数据	[-32768,+32767]	有符号 BIN16 位	ANY16_S	INT
(d)	存储转换结果的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--

使用的软元件

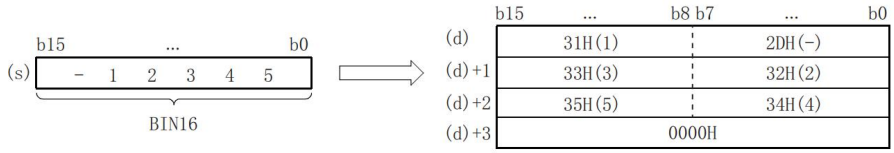
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BINDA	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2																	●	●	●						●	●

功能

将(s)中指定的 BIN16 位数据、10 进制数表示时的各位的数值转换为 ASCII 码后，存储到(d)中指定的软元件编号以后。



例：(s)中指定了-12345 的情况下(指定了带符号的情况下)



(d)中存储的运算结果将变为下述内容。

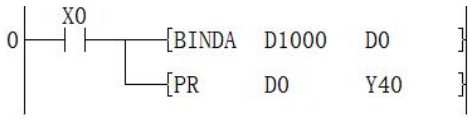
- 在“符号”中，BIN 数据为正时将存储 20H，为负时将存储 2DH。
- 在有效位数的左侧的 0 中，将存储 20H。(进行 0 抑制。)例如“00325”的情况下、“00”将变为 20H，“325”将变为有效位数。
- 至(d)+3 中指定的软元件的数据的存储时，SM191(输出字符数切换信号)为 OFF 的情况下将存储 0，为 ON 的情况下不变化。

注意事项：(d)的占用点数在 SM191 为 ON 时变为 3 点，SM191 为 OFF 时变为 4 点。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。

示例



当 X000 为 ON 后，将 16 位数据(BIN)D1000 的值转换成 10 进制数字的 ASCII 码，然后使用 PR 指令将已经转换的 ASCII 码逐个字符依次时分输出到 Y040~Y057 中的程序。

DBINDA/BIN32 位数据→10 进制 ASCII 转换

DBINDA(P)

将(s)中指定的 BIN32 位数据、10 进制数表示时的各位的数值转换为 ASCII 码后，存储到(d)中指定的软元件编号以后。

-[DBINDA （s） （d）]

内容、范围、数据类型

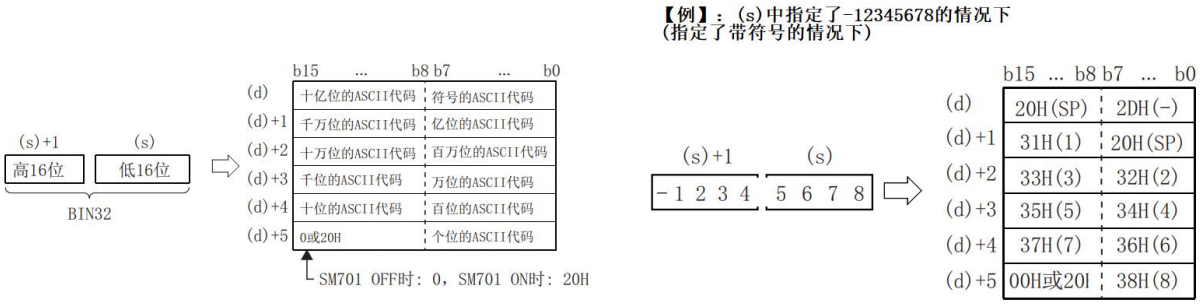
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	进行 ASCII 转换的 BIN 数据	[-2147483648,2147483647]	有符号 BIN32 位	ANY32_S	DINT
(d)	存储转换结果的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBINDA	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2																	●	●	●						●	●

功能

将(s)中指定的 BIN32 位数据、以 10 进制数表示时的各位的数值转换为 ASCII 码后，存储到(d)中指定的软元件编号以后。



(d)中存储的运算结果将变为下述内容。

- 在“符号”中，BIN 数据为正时将存储 20H，为负时将存储 2DH。
- 有效位数的左侧的 0 处将存储 20H。(进行 0 抑制。)例如，“0012034560”的情况下、“00”将变为 20H，“12034560”将变为有效位数。
- 对于(d)+5 中指定的软元件的高位 8 位中存储的数据，在 SM191(输出字符数切换信号)为 OFF 的情况下将存储 0，为 ON 的情况下将存储 20H。

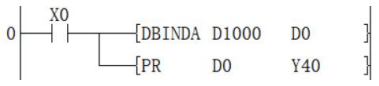
注意事项

(d)占用 6 点。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
4086H	(d)写入地址超出软元件范围。

示例



当 X000 为 ON 后，将 32 位数据(BIN)D1000 的值转换成 10 进制数字的 ASCII 码，然后使用 PR(FNC 77)指令将已经转换的 ASCII 码逐个字符依次时分输出到 Y040~Y051 中的程序。

DABIN/10 进制 ASCII→BIN 的转换

DABIN(P)

数字的 ASCII 码(30H~39H)形式现实的数据转换成 BIN 数据的指令。

-[DABIN（s） （d）]

内容、范围、数据类型

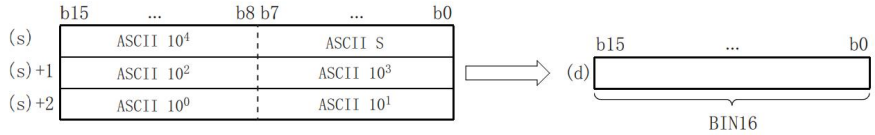
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	保存要转换成 BIN 值的数据(ASCII 码)的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	保存转换结果的软元件编号	-	BIN16 位	ANY16_S	INT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DABIN	参数 1																	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

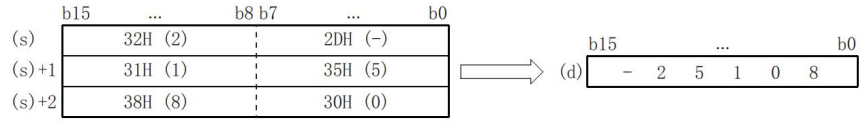
功能

将(s)中指定的软元件编号以后中存储的 10 进制 ASCII 数据转换为 BIN16 位数据后，存储到(d)中指定的软元件中。



ASCII S	ASCII 10^0	ASCII 10^1	ASCII 10^2	ASCII 10^3	ASCII 10^4
ASCII 代码	个位的 ASCII 代码	十位的 ASCII 代码	百位的 ASCII 代码	千位的 ASCII 代码	万位的 ASCII 代码

例：(s)中指定了-25108 的情况下



- 对于(s)～(s)+2 中指定的 ASCII 数据，为[-32768,+32767]的范围内。
- 符号数据中，转换的数据为正时设置 20H，为负时设置 2DH。(设置了 20H、2DH 以外的情况下，将被作为正的数据处理。(DABIN(P))
- 各位中设置的 ASCII 代码的范围为 30H~39H。
- 各位中设置的 ASCII 代码为 20H、00H 时，将作为 30H 处理。

错误码

错误码	内容
4084H	符号数据超出 30H~39H、20H、00H、2DH 的范围时； (s)～(s)+2 中指定的各位的 ASCII 代码超出 30H~39H、20H、00H 的范围时； (s)～(s)+2 中指定的 ASCII 数据为[-32768,+32767]以外时。
4085H	(s)读取地址超出软元件范围 。
4086H	(d)写入地址超出软元件范围 。

示例

当 X000 为 ON 时，将 D20~D22 中设定的符号以及 5 位 10 进制数字的 ASCII 码数据转换成 BIN 值后，保存到 D0 中的程序。

福州富昌维控电子科技有限公司

DDABIN/10 进制 ASCII→BIN32 位数据转换

DDABIN(P)

将(s)中指定的软元件编号以后中存储的 10 进制 ASCII 数据转换为 BIN32 位数据后，存储到(d)中指定的软元件编号中。

-[DDABIN（s）　（d）]

内容、范围、数据类型

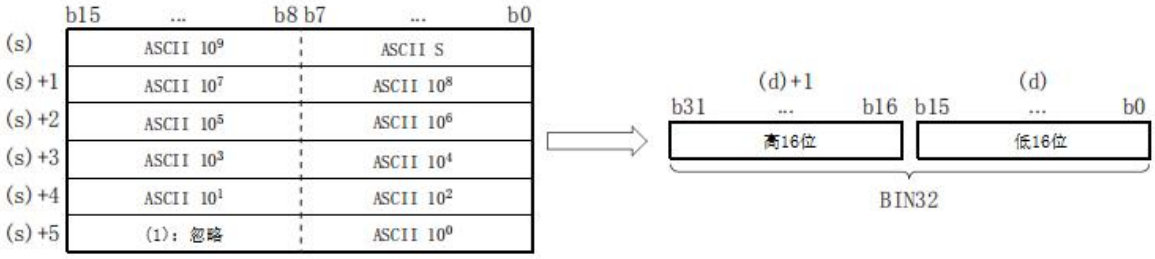
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	保存要转换成 BIN 值的数据(ASCII 码)的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	保存转换结果的软元件编号	-	有符号 BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DDABIN	参数 1																	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●	●	●				●	●

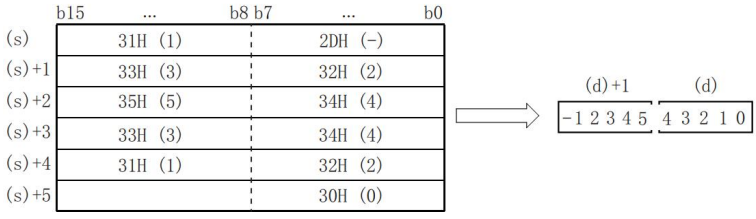
功能

将(s)中指定的软元件编号以后中存储的 10 进制 ASCII 数据转换为 BIN32 位数据后，存储到(d)中指定的软元件中。



ASCII S	ASCII 代码	ASCII S	ASCII 代码
ASCII 10^0	个位的 ASCII 代码	ASCII 10^5	十万位的 ASCII 代码
ASCII 10^1	十位的 ASCII 代码	ASCII 10^6	百万位的 ASCII 代码
ASCII 10^2	百位的 ASCII 代码	ASCII 10^7	千万位的 ASCII 代码
ASCII 10^3	千位的 ASCII 代码	ASCII 10^8	亿位的 ASCII 代码
ASCII 10^4	万位的 ASCII 代码	ASCII 10^9	十亿位的 ASCII 代码

(s)中指定了-1234543210 的情况下(指定了有符号的情况下)



• 对于(s)～(s)+5 中指定的 ASCII 数据，为[-2147483648,2147483647]的范围内。此外，(s)+5 的高位字节中存储的数据将被忽略。

• 符号数据中，转换的数据为正时设置 20H，为负时设置 2DH。（设置了 20H、2DH 以外的情况下，将被作为正的数据处理。(DABIN(P)）

• 各位中设置的 ASCII 代码的范围为 30H~39H。

• 各位中设置的 ASCII 代码为 20H、00H 时，将作为 30H 处理。

福州富昌维控电子科技有限公司

错误码

错误码	内容
4084H	符号数据超出 30H~39H、20H、00H、2DH 的范围时； (s)～(s)+5 中指定的各位的 ASCII 代码超出 30H~39H、20H、00H 的范围时； (s)～(s)+5 中指定的 ASCII 数据为[-2147483648,2147483647]以外时。
4085H	(s) 读取地址超出软元件范围。
4086H	(d) 写入地址超出软元件范围。

示例

当 X000 为 ON 时，将 D20~D25 中设定的符号以及 10 位 10 进制数字的 ASCII 码数据转换成 BIN 值后，保存到 D0~D1 中的程序。

LIMIT/BIN16 位数据上下限限位控制

LIMIT(P)

(s3)中指定的输入值(BIN16 位值)根据(s1)、(s2)中指定的上下限限位值的范围，对(d)中指定的软元件中存储的输出值进行控制。

-[LIMIT (s1) (s2) (s3) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
s1	下限限位值(最小输出极限值)	[-32768,32767]	BIN16 位	ANY16_S	INT
s2	上限限位值(最大输出极限值)	[-32768,32767]	BIN16 位	ANY16_S	INT
s3	通过上下限限位控制进行控制的输入值	[-32768,32767]	BIN16 位	ANY16_S	INT
d	存储通过上下限限位控制进行控制的输出值的软元件起始编号	-	BIN16 位	ANY16_S	INT

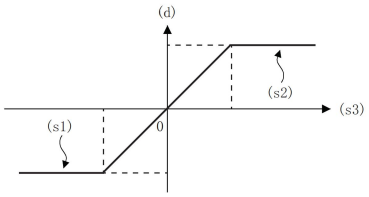
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LIMIT	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3											●	●	●	●	●	●	●	●	●						●	●
	参数 4												●	●	●	●	●	●	●	●						●	●

功能

(s3)中指定的输入值(BIN16 位值)根据(s1)、(s2)中指定的上下限限位值的范围，对(d)中指定的软元件中存储的输出值进行控制。输出值按下述方式被控制。

条件	输出值中存储的值
下限限位值(s1)>输入值(s3)时	下限限位值(s1)
上限限位值(s1)<输入值(s3)时	上限限位值(s2)
下限限位值(s1)≤输入值(s3)≤上限限位值(s2)时	输入值(s3)



- 仅通过上限限位值进行控制的情况下，在(s1)中指定的下限限位值中设置数据范围的最小值。
- 仅通过下限限位值进行控制的情况下，在(s2)中指定的上限限位值中设置数据范围的最大值。

错误码

错误码	内容
4085H	读取地址超出软元件范围。
4086H	写入地址超出软元件范围。
4084H	上限值<下限值。

示例

当 X000 为 ON 时

- D0 ＜ 500 时，D1 为 500。
- 500 ≤ D0 ≤5000 时，D1 为 D0 的值。
- 5000 ＜ D0 时，D1 为 5000。

DLIMIT/BIN32 位数据上下限限位控制

DLIMIT(P)

(s3)中指定的输入值(BIN32 位值)根据(s1)、(s2)中指定的上下限限位值的范围，对(d)中指定的软元件中存储的输出值进行控制。

-[DLIMIT （s1） （s2） （s3） （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
s1	下限限位值(最小输出极限值)	[-2147483648,2147483647]	BIN32 位	ANY32_S	DINT
s2	上限限位值(最大输出极限值)	[-2147483648,2147483647]	BIN32 位	ANY32_S	DINT
s3	通过上下限限位控制进行控制的输入值	[-2147483648,2147483647]	BIN32 位	ANY32_S	DINT
d	存储通过上下限限位控制进行控制的输出值的软元件起始编号	-	BIN32 位	ANY32_S	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DLIMIT	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3											●	●	●	●	●	●	●	●	●	●	●				●	●
	参数 4												●	●	●	●	●	●	●	●	●	●				●	●

功能

(s3)中指定的输入值(BIN32 位值)根据(s1)、(s2)中指定的上下限限位值的范围，对(d)中指定的软元件中存储的输出值进行控制。输出值按下述方式被控制。

条件	输出值中存储的值
下限限位值((s1)、(s1)+1)>输入值((s3)、(s3)+1)时	下限限位值((s1)、(s1)+1)
上限限位值((s2)、(s2)+1)<输入值((s3)、(s3)+1)时	上限限位值((s2)、(s2)+1)
下限限位值((s1)、(s1)+1)≤输入值((s3)、(s3)+1)≤上限限位值((s2)、(s2)+1)时	输入值((s3)、(s3)+1)

- 仅进行上限限位值的控制的情况下，在(s1)中指定的下限限位值中设置数据范围的最小值。
- 仅进行下限限位值的控制的情况下，在(s2)中指定的上限限位值中设置数据范围的最大值。

错误码

错误码	内容
4085H	读取地址超出软元件范围。
4086H	写入地址超出软元件范围。
4084H	上限值<下限值。

示例

动作：

- (D1、D0) ＜ 10000 时，(D11、D10)为 10000。
- 10000 ≤ (D1、D0) ≤1000000 时，(D11、D10)为(D1、D0)的值。
- 1000000 ＜ (D1、D0)时，(D11、D10)为 1000000。

福州富昌维控电子科技有限公司

SCL/BIN16 位单位标度（各点坐标数据）

SCL(P)

对(s2)中指定的标度用转换数据(16 位数据单位)从(s1)中指定的输入值开始进行标度，将运算结果存储到(d)中指定的软元件中。

-[SCL （s1） （s2） （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	进行标度的输入值或存储了输入值的软元件起始编号	[-32768,+32767]	有符号 BIN16 位	ANY16_S	--
(s2)	存储了标度用转换数据的软元件起始编号	-	有符号 BIN16 位	ANY16_S	--
(d)	存储通过标度进行控制的输出值的软元件起始编号	-	有符号 BIN16 位	ANY16_S	--

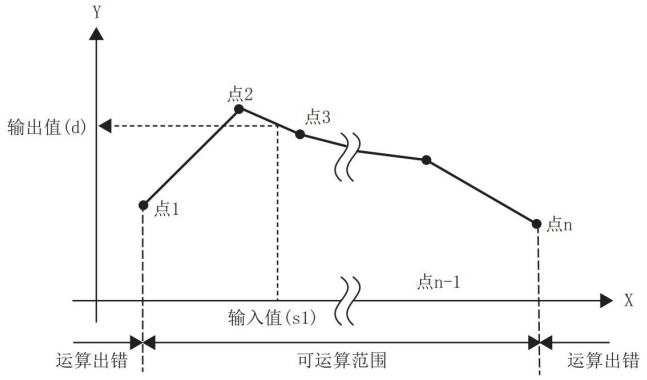
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SCL	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2																	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

功能

对于(s2)中指定的标度用转换数据(16 位数据单位)，通过(s1)中指定的输入值进行标度，将运算结果存储到(d)中指定的软元件编号中。标度转换是基于(s2)中指定的软元件以后存储的标度用转换数据进行。

设置项目(n 表示(s2)中指定的坐标点数)		软元件分配
坐标点数		(s2)
点 1	X 坐标	(s2)+1
	Y 坐标	(s2)+2
点 2	X 坐标	(s2)+3
	Y 坐标	(s2)+4
.....		
点 n	X 坐标	(s2)+2n-1
	Y 坐标	(s2)+2n

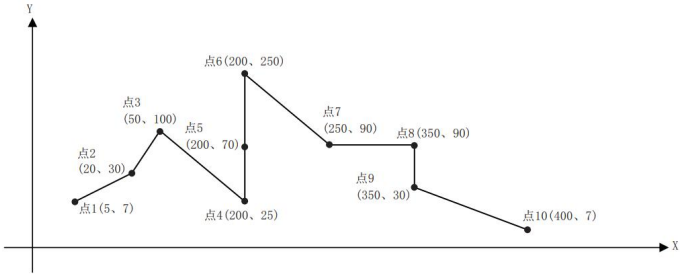


- 运算结果不是整数值的情况下，对小数点以下第 1 位数进行四舍五入。
- 标度用转换数据的 X 坐标数据应设置为升序。

福州富昌维控电子科技有限公司

- (s1)应在标度用转换数据范围内((s2)的软元件值)进行设置。
- 多个点指定了同一 X 坐标的情况下，将输出第 2 点的 Y 坐标的值。
- 应将标度用转换数据的坐标点数设置在 1~32767 的范围内。
- 标度用转换表的设置示例

如下图所示的标度用转换特性的情况下，应如下述数据表进行设置。



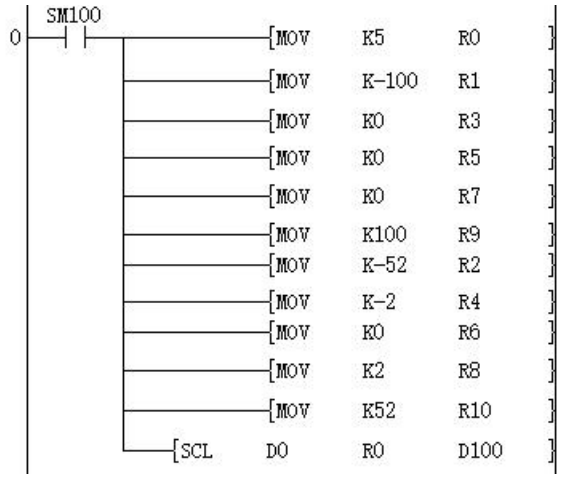
设置项目		设置软元件和设置内容			备注
		(s2)中指定了 R0 的情况下		设定内容	
坐标点数		(s2)	R0	K10	
点 1	X 坐标	(s2)+1	R1	K5	
	Y 坐标	(s2)+2	R2	K20	
点 2	X 坐标	(s2)+3	R3	K30	
	Y 坐标	(s2)+4	R4	K50	
点 3	X 坐标	(s2)+5	R5	K100	
	Y 坐标	(s2)+6	R6	K200	
点 4	X 坐标	(s2)+7	R7	K25	像这样如果通过 3 点指定坐标，可取中间值为输出值。 本示例中，输出值(中间值)通过点 5 的 Y 坐标指定。 而 3 点以上 X 坐标相同的情况下，也输出第 2 点的值。
	Y 坐标	(s2)+8	R8	K200	
点 5	X 坐标	(s2)+9	R9	K70	
	Y 坐标	(s2)+10	R10	K200	
点 6	X 坐标	(s2)+11	R11	K250	
	Y 坐标	(s2)+12	R12	K250	
点 7	X 坐标	(s2)+13	R13	K250	
	Y 坐标	(s2)+14	R14	K90	
点 8	X 坐标	(s2)+15	R15	K350	像这样如果通过 2 点指定坐标，输出值为下一点的 Y 坐标的值。 本示例中，输出值通过点 9 的 Y 坐标指定。
	Y 坐标	(s2)+16	R16	K90	
点 9	X 坐标	(s2)+17	R17	K350	
	Y 坐标	(s2)+18	R18	K30	
点 10	X 坐标	(s2)+19	R19	K400	
	Y 坐标	(s2)+20	R20	K7	

错误码

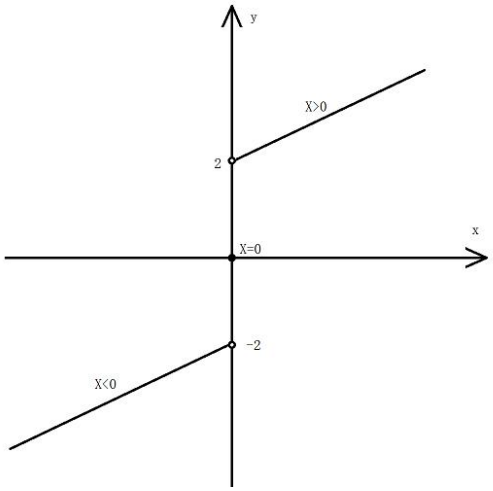
错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	数据表的 Xn 数据未以升序排序时。但是，指令将执行到发生出错的位置为止； (s1)中指定的输入值超出设置的标度用转换数据的范围时； (s2)的软元件开始的坐标点数为 0 以下时。

福州富昌维控电子科技有限公司

示例



当-100≤D0(X)＜0 时，D100(Y)= $\frac{1}{2}X-2$ ；当 D0(X)=0 时，D100(Y)=0；当 0＜D0(X)≤100 时，D100(Y)= $\frac{1}{2}X+2$ 。



$$\begin{cases} Y=\frac{1}{2}X+2 (100\geq X>0) \\ Y=0 (X=0) \\ Y=\frac{1}{2}X-2 (-100\leq X<0) \end{cases}$$

DSCL/BIN32 位单位标度（各点坐标数据）

DSCL(P)

对(s2)中指定的标度用转换数据(32 位数据单位)从(s1)中指定的输入值开始进行标度，将运算结果存储到(d)中指定的软元件中。

-[DSCL（s1） （s2） （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	进行标度的输入值或存储了输入值的软元件起始编号	[-2147483648,2147483647]	有符号BIN32位	ANY32_S	--
(s2)	存储了标度用转换数据的软元件起始编号	-	有符号BIN32位	ANY32_S	--
(d)	存储通过标度进行控制的输出值的软元件起始编号	-	有符号BIN32位	ANY32_S	--

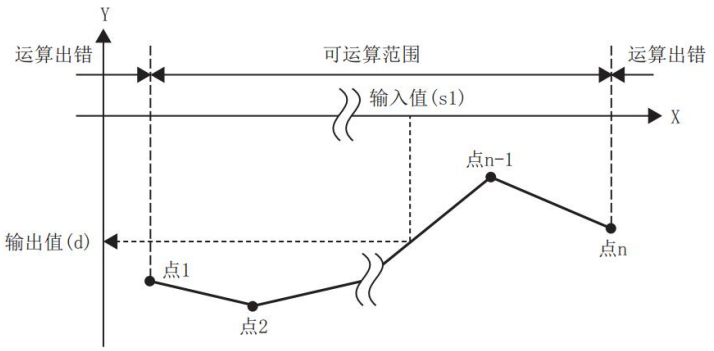
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSCL	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2																	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

对于(s2)中指定的标度用转换数据(32 位数据单位)，通过(s1)中指定的输入值进行标度，将运算结果存储到(d)中指定的软元件编号中。标度转换是基于(s2)中指定的软元件以后存储的标度用转换数据进行。

设置项目(n 表示(s2)中指定的坐标点数)		软元件分配
坐标点数		(s2)+1、(s2)
点 1	X 坐标	(s2)+3、(s2)+2
	Y 坐标	(s2)+5、(s2)+4
点 2	X 坐标	(s2)+7、(s2)+6
	Y 坐标	(s2)+9、(s2)+8
.....		
点 n	X 坐标	(s2)+4n-1、(s2)+4n-2
	Y 坐标	(s2)+4n+1、(s2)+4n

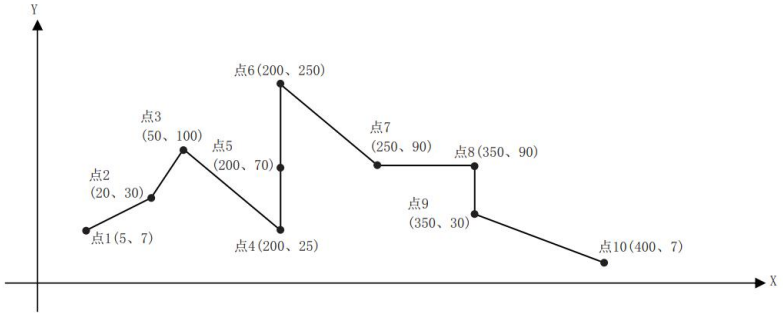


- 运算结果不是整数值的情况下，对小数点以下第 1 位数进行四舍五入。
- 标度用转换数据的 X 坐标数据应设置为升序。
- 对于(s1)应在标度用转换数据范围内((s2)、(s2)+1 的软元件值)进行设置。

福州富昌维控电子科技有限公司

- 多个点指定了同一 X 坐标的情况下，将输出第 2 点的 Y 坐标的值。
- 应将标度用转换数据的坐标点数设置在[1,2147483647]的范围内。
- 标度用转换表的设置示例

如下图所示的标度用转换特性的情况下，应如下述数据表进行设置。



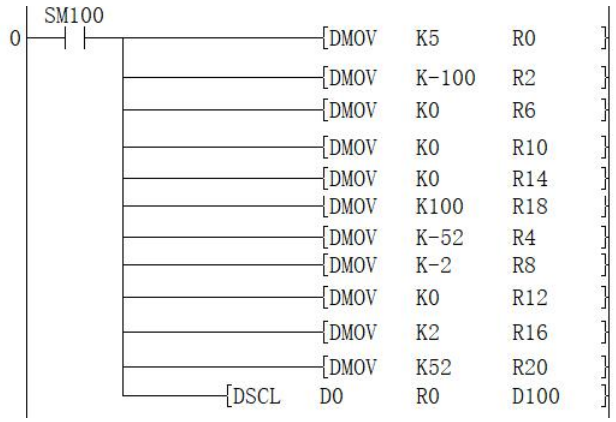
设置项目		设置软元件和设置内容			备注
		(s2)中指定了 R0 的情况下		设定内容	
坐标点数		(s2)+1、(s2)	R1、R0	K10	
点 1	X 坐标	(s2)+3、(s2)+2	R3、R2	K5	
	Y 坐标	(s2)+5、(s2)+4	R5、R4	K7	
点 2	X 坐标	(s2)+7、(s2)+6	R7、R6	K20	
	Y 坐标	(s2)+9、(s2)+8	R9、R8	K30	
点 3	X 坐标	(s2)+11、(s2)+10	R10、R11	K50	
	Y 坐标	(s2)+13、(s2)+12	R13、R12	K100	
点 4	X 坐标	(s2)+15、(s2)+14	R15、R14	K200	像这样如果通过 3 点指定坐标，可取中间值为输出值。 本示例中，输出值(中间值)通过点 5 的 Y 坐标指定。 而 3 点以上 X 坐标相同的情况下，也输出第 2 点的值。
	Y 坐标	(s2)+17、(s2)+16	R17、R16	K25	
点 5	X 坐标	(s2)+19、(s2)+18	R19、R18	K200	
	Y 坐标	(s2)+21、(s2)+20	R21、R20	K70	
点 6	X 坐标	(s2)+23、(s2)+22	R23、R22	K200	
	Y 坐标	(s2)+25、(s2)+24	R25、R24	K250	
点 7	X 坐标	(s2)+27、(s2)+26	R27、R26	K250	
	Y 坐标	(s2)+29、(s2)+28	R29、R28	K90	
点 8	X 坐标	(s2)+31、(s2)+30	R31、R30	K350	像这样如果通过 2 点指定坐标，输出值为下一点的 Y 坐标的值。 本示例中，输出值通过点 9 的 Y 坐标指定。
	Y 坐标	(s2)+33、(s2)+32	R33、R32	K90	
点 9	X 坐标	(s2)+35、(s2)+34	R35、R34	K350	
	Y 坐标	(s2)+37、(s2)+36	R37、R36	K30	
点 10	X 坐标	(s2)+39、(s2)+38	R39、R38	K400	
	Y 坐标	(s2)+41、(s2)+40	R41、R40	K7	

错误码

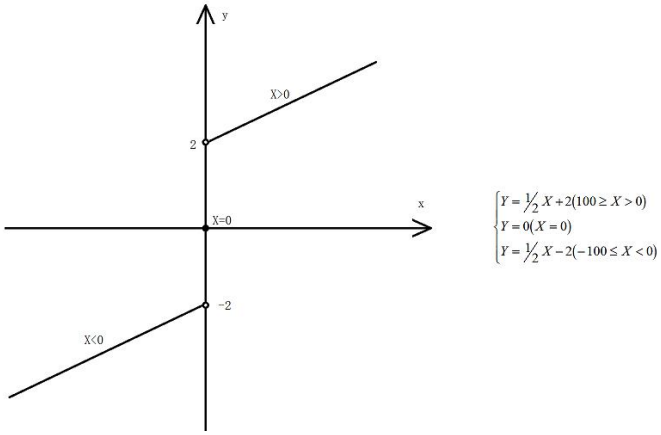
错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	数据表的 Xn 数据未以升序排序时。但是，指令将执行到发生出错的位置为止； (s1)中指定的输入值超出设置的标度用转换数据的范围时； (s2)的软元件开始的坐标点数为 0 以下时。

福州富昌维控电子科技有限公司

示例



当-100≤D0(X)＜0 时，D100(Y)= $\frac{1}{2}X-2$；当 D0(X)=0 时，D100(Y)=0;当 0＜D0(X)≤100 时，D100(Y)= $\frac{1}{2}X+2$ 。



SCL2/BIN16 位单位标度（各 X/Y 坐标数据）

SCL2(P)

对(s2)中指定的标度用转换数据(16 位数据单位)从(s1)中指定的输入值开始进行标度，将运算结果存储到(d)中指定的软元件中。

-[SCL2（s1）（s2）（d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	进行标度的输入值或存储了输入值的软元件起始编号	[-32768,+32767]	有符号 BIN16 位	ANY16_S	--
(s2)	存储了标度用转换数据的软元件起始编号	-	有符号 BIN16 位	ANY16_S	--
(d)	存储通过标度进行控制的输出值的软元件起始编号	-	有符号 BIN16 位	ANY16_S	--

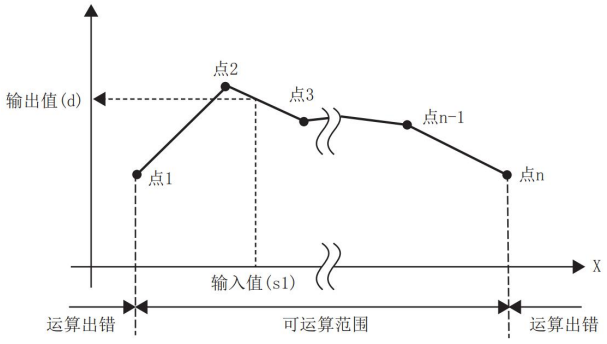
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SCL2	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2																	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

功能

对于(s2)中指定的标度用转换数据(16 位数据单位)，通过(s1)中指定的输入值进行标度，将运算结果存储到(d)中指定的软元件编号中。标度转换是基于(s2)中指定的软元件以后存储的标度用转换数据进行。

设置项目(n 表示(s2)中指定的坐标点数)		软元件分配
坐标点数		(s2)
X 坐标	点 1	(s2)+1
	点 2	(s2)+2
	.....	.....
	点 n	(s2)+n
Y 坐标	点 1	(s2)+n+1
	点 2	(s2)+n+2
	.....	.....
	点 n	(s2)+2n

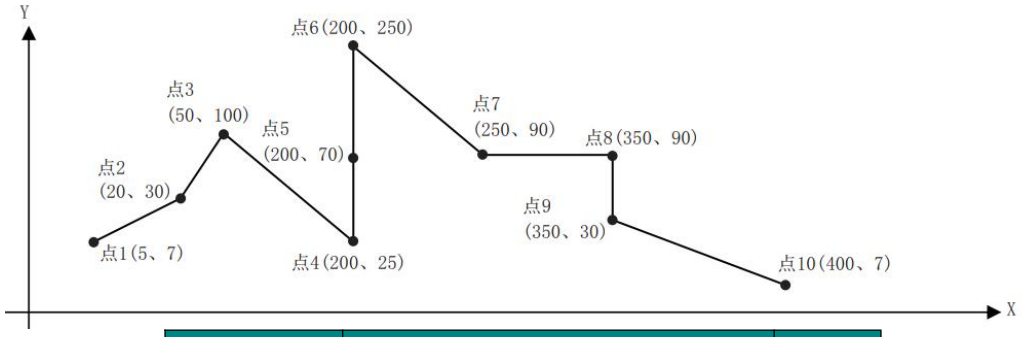


- 运算结果不是整数值的情况下，对小数点以下第 1 位数进行四舍五入。
- 标度用转换数据的 X 坐标数据应设置为升序。

福州富昌维控电子科技有限公司

- (s1)应在标度用转换数据范围内((s2)的软元件值)进行设置。
- 多个点指定了同一 X 坐标的情况下，将输出第 2 点的 Y 坐标的值 。
- 应将标度用转换数据的坐标点数设置在 1~32767 的范围内。
- 标度用转换表的设置示例

如下图所示的标度用转换特性的情况下，应如下述数据表进行设置。



设置项目		设置软元件和设置内容			备注
		(s2)中指定了 R0 的情况下		设定内容	
坐标点数		(s2)	R0	K10	
X 坐标	点 1	(s2)+1	R1	K5	
	点 2	(s2)+2	R2	K20	
	点 3	(s2)+3	R3	K50	
	点 4	(s2)+4	R4	K200	*1 参阅
	点 5	(s2)+5	R5	K200	
	点 6	(s2)+6	R6	K200	
	点 7	(s2)+7	R7	K250	
	点 8	(s2)+8	R8	K350	*2 参阅
	点 9	(s2)+9	R9	K350	
	点 10	(s2)+10	R10	K400	
Y 坐标	点 1	(s2)+11	R11	K7	
	点 2	(s2)+12	R12	K30	
	点 3	(s2)+13	R13	K100	
	点 4	(s2)+14	R14	K25	*1 参阅
	点 5	(s2)+15	R15	K70	
	点 6	(s2)+16	R16	K250	
	点 7	(s2)+17	R17	K90	
	点 8	(s2)+18	R18	K90	*2 参阅
	点 9	(s2)+19	R19	K30	
	点 10	(s2)+20	R20	K7	

*1 像点 4、5、6 那样，如果通过 3 点指定坐标，可取中间值为输出值。

本示例中，输出值(中间值)通过点 5 的 Y 坐标指定。

而 3 点以上 X 坐标相同的情况下，也输出第 2 点的值。

*2 像点 8、9 那样，如果通过 2 点指定坐标，输出值为下一点的 Y 坐标的值。

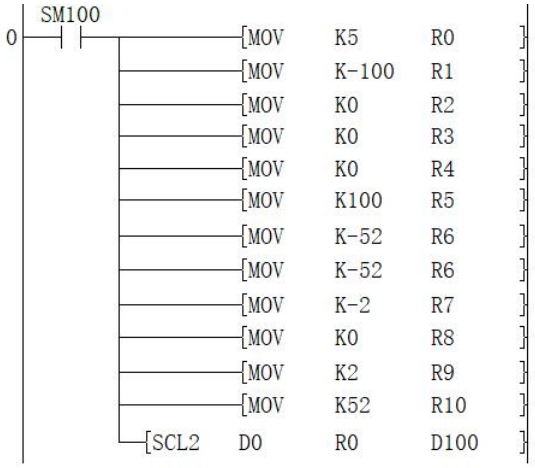
本示例中，输出值通过点 9 的 Y 坐标指定

福州富昌维控电子科技有限公司

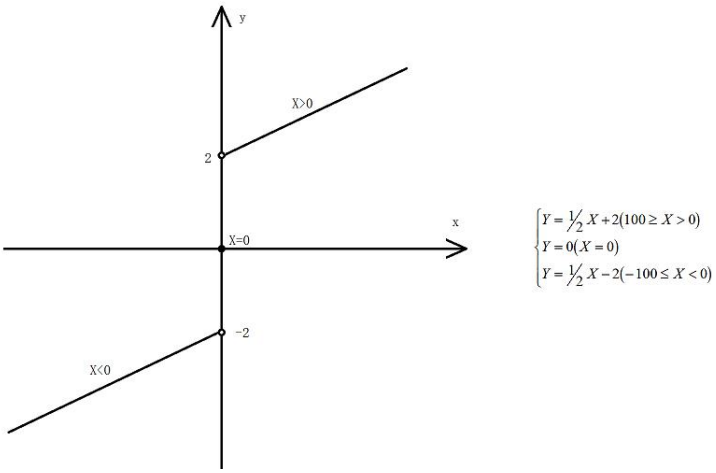
错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	数据表的 Xn 数据未以升序排序时。但是，指令将执行到发生出错的位置为止； (s1)中指定的输入值超出设置的标度用转换数据的范围时； (s2)的软元件开始的坐标点数为 0 以下时。

示例



当-100≤D0(X)＜0 时，D100(Y)= $\frac{1}{2}X-2$ ；当 D0(X)=0 时，D100(Y)=0；当 0＜D0(X)≤100 时，D100(Y)= $\frac{1}{2}X+2$ 。



$$\begin{cases} Y=\frac{1}{2}X+2 (100\geq X>0) \\ Y=0 (X=0) \\ Y=\frac{1}{2}X-2 (-100\leq X<0) \end{cases}$$

DSCL2/BIN32 位单位标度（各 X/Y 坐标数据）

DSCL2(P)

对(s2)中指定的标度用转换数据(32 位数据单位)从(s1)中指定的输入值开始进行标度，将运算结果存储到(d)中指定的软元件中。

-[DSCL2（s1）（s2）（d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	进行标度的输入值或存储了输入值的软元件起始编号	[-2147483648,2147483647]	有符号 BIN32 位	ANY32_S	--
(s2)	存储了标度用转换数据的软元件起始编号	-	有符号 BIN32 位	ANY32_S	--
(d)	存储通过标度进行控制的输出值的软元件起始编号	-	有符号 BIN32 位	ANY32_S	--

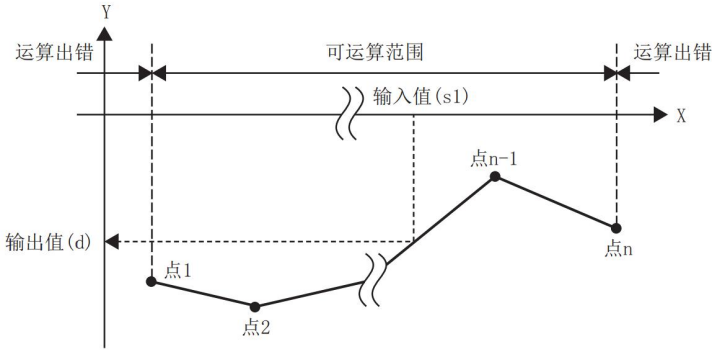
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSCL2	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2																	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●	●	●				●	●

功能

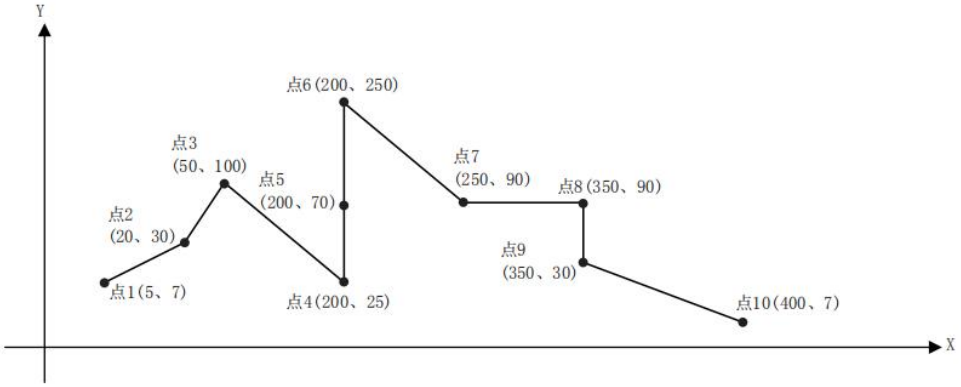
对于(s2)中指定的标度用转换数据(32 位数据单位)，通过(s1)中指定的输入值进行标度，将运算结果存储到(d)中指定的软元件编号中。标度转换是基于(s2)中指定的软元件以后存储的标度用转换数据进行。

设置项目(n 表示(s2)中指定的坐标点数)		软元件分配
坐标点数		(s2)+1、(s2)
X 坐标	点 1	(s2)+3、(s2)+2
	点 2	(s2)+5、(s2)+4
	.....	.....
	点 n	(s2)+2n+1、(s2)+2n
Y 坐标	点 1	(s2)+2n+3，(s2)+2n+2
	点 2	(s2)+2n+5，(s2)+2n+4
	.....	.....
	点 n	(s2)+4n+1、(s2)+4n



- 运算结果不是整数值的情况下，对小数点以下第 1 位数进行四舍五入。
- 标度用转换数据的 X 坐标数据应设置为升序。
- 对于(s1)应在标度用转换数据范围内((s2)、(s2)+1 的软元件值)进行设置。
- 多个点指定了同一 X 坐标的情况下，将输出第 2 点的 Y 坐标的值。
- 应将标度用转换数据的坐标点数设置在[1,2147483647]的范围内。
- 标度用转换表的设置示例

如下图所示的标度用转换特性的情况下，应如下述数据表进行设置。



设置项目		设置软元件和设置内容			备注
		(s2)中指定了 R0 的情况下		设定内容	
坐标点数		(s2)+1、(s2)	R1、R0	K10	
X 坐标	点 1	(s2)+3、(s2)+2	R3、R2	K5	
	点 2	(s2)+5、(s2)+4	R5、R4	K20	
	点 3	(s2)+7、(s2)+6	R7、R6	K50	
	点 4	(s2)+9、(s2)+8	R9、R8	K200	*1 参阅
	点 5	(s2)+11、(s2)+10	R10、R11	K200	
	点 6	(s2)+13、(s2)+12	R13、R12	K200	
	点 7	(s2)+15、(s2)+14	R15、R14	K250	
	点 8	(s2)+17、(s2)+16	R17、R16	K350	*2 参阅
	点 9	(s2)+19、(s2)+18	R19、R18	K350	
	点 10	(s2)+21、(s2)+20	R21、R20	K400	
Y 坐标	点 1	(s2)+23、(s2)+22	R23、R22	K7	
	点 2	(s2)+25、(s2)+24	R25、R24	K30	
	点 3	(s2)+27、(s2)+26	R27、R26	K100	
	点 4	(s2)+29、(s2)+28	R29、R28	K25	*1 参阅
	点 5	(s2)+31、(s2)+30	R31、R30	K70	
	点 6	(s2)+33、(s2)+32	R33、R32	K250	
	点 7	(s2)+35、(s2)+34	R35、R34	K90	
	点 8	(s2)+37、(s2)+36	R37、R36	K90	*2 参阅
	点 9	(s2)+39、(s2)+38	R39、R38	K30	
	点 10	(s2)+41、(s2)+40	R41、R40	K7	

*1 像点 4、5、6 那样，如果通过 3 点指定坐标，可取中间值为输出值。

本示例中，输出值(中间值)通过点 5 的 Y 坐标指定。

而 3 点以上 X 坐标相同的情况下，也输出第 2 点的值。

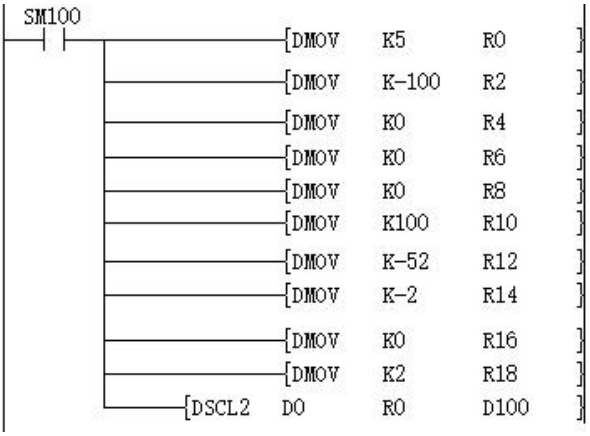
*2 像点 8、9 那样，如果通过 2 点指定坐标，输出值为下一点的 Y 坐标的值。

本示例中，输出值通过点 9 的 Y 坐标指定。

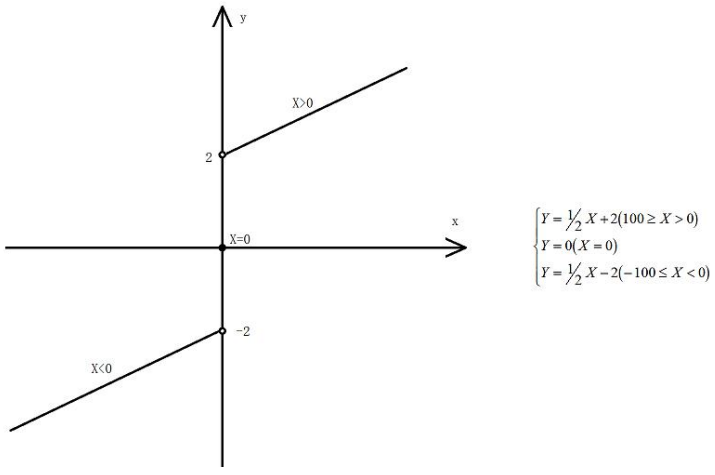
错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。
4084H	数据表的 Xn 数据未以升序排序时。但是，指令将执行到发生出错的位置为止； (s1)中指定的输入值超出设置的标度用转换数据的范围时； (s2)的软元件开始的坐标点数为 0 以下时。

示例



当-100≤D0(X)＜0 时，D100(Y)= $\frac{1}{2}X-2$ ；当 D0(X)=0 时，D100(Y)=0；当 0＜D0(X)≤100 时，D100(Y)= $\frac{1}{2}X+2$ 。



ZONE/BIN16 位数据区域控制

ZONE(P)

在(s3)中指定的输入值中附加(s1)或(s2)中指定的偏置值后，存储到(d)中指定的软元件编号中。

-[ZONE （s1） （s2） （s3） （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
（s1）	输入值中进行加法运算的负的偏置值	[-32768,32767]	BIN16 位	ANY16_S	INT
（s2）	输入值中进行加法运算的正的偏置值	[-32768,32767]	BIN16 位	ANY16_S	INT
（s3）	用于进行区域控制的输入值	[-32768,32767]	BIN16 位	ANY16_S	INT
（d）	存储通过区域控制进行控制的输出值的软元件起 始编号	-	BIN16 位	ANY16_S	INT

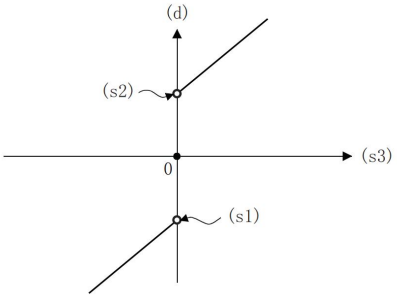
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ZONE	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3											●	●	●	●	●	●	●	●	●						●	●
	参数 4												●	●	●	●	●	●	●	●						●	●

功能

在(s3)中指定的输入值(BIN16 位值)中附加(s1)或(s2)中指定的偏置值后，存储到(d)中指定的软元件编号中。偏置值按下述方式被控制。

条件	输出值中存储的值
输入值(s3)<0 时	输入值(s3)+负的偏置值(s1)
输入值(s3)=0 时	0
输入值(s3)>0 时	输入值(s3)+正的偏置值(s2)



• (d)中存储的输出值为带符号 BIN16 位值的情况下，运算结果超出[-32768,32767]的范围时，其情况如下例所示。

例：(s1)为-100，(s3)为-32768 时，则输出值=-32768+(-100)=8000H-FF9CH=7F9CH=32668。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

当 X000 为 ON 时

•D0＜0 时，D1 中保存(D0)+(-1000)的值。

• D0=0 时，D1 中保存 0。

• 0＜D0 时，D1 中保存(D0)+(1000)的值。

DZONE/BIN32 位数据区域控制

DZONE(P)

在(s3)中指定的输入值中附加(s1)或(s2)中指定的偏置值后，存储到(d)中指定的软元件编号中。

-[DZONE (s1) (s2) (s3) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	输入值中进行加法运算的负的偏置值	[-2147483648,2147483647]	BIN32 位	ANY32_S	DINT
(s2)	输入值中进行加法运算的正的偏置值	[-2147483648,2147483647]	BIN32 位	ANY32_S	DINT
(s3)	用于进行区域控制的输入值	[-2147483648,2147483647]	BIN32 位	ANY32_S	DINT
(d)	存储通过区域控制进行控制的输出值的软元件起 始编号	-	BIN32 位	ANY32_S	DINT

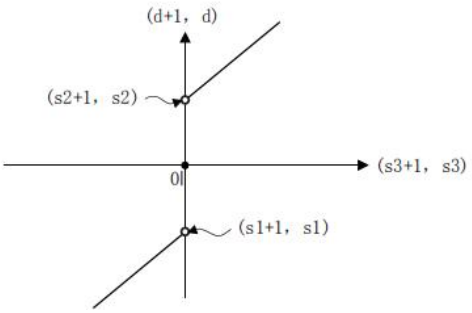
使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DZONE	参数 1											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3											●	●	●	●	●	●	●	●	●	●	●				●	●
	参数 4												●	●	●	●	●	●	●	●	●	●				●	●

功能

在(s3)中指定的输入值(BIN32 位值)中附加(s1)或(s2)中指定的偏置值后，存储到(d)中指定的软元件编号中。偏置值按下述方式被控制。

条件	输出值中存储的值
输入值((s3)、(s3)+1)<0 时	输入值((s3)、(s3)+1)+负的偏置值(s1)、(s1)+1
输入值((s3)、(s3)+1)=0 时	0
输入值((s3)、(s3)+1)>0 时	输入值((s3)、(s3)+1)+正的偏置值(s2)、(s2)+1



• (d)、(d)+1 中存储的输出值为带符号 BIN32 位值的情况下，运算结果超出-2147483648~2147483647 的范围时，其情况如下例所示。

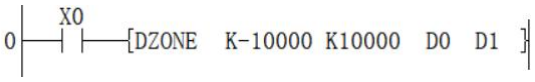
例：(s1)、(s1)+1 为-1000，(s3)、(s3)+1 为-2147483648 时，

则输出值=-2147483648+(-1000)=80000000H+FFFFFC18H=2147482648。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



• (D1、D0)＜0 时，(D11、D10)中保存(D1、D0)+(-10000)的值。

• (D1、D0)=0 时，(D11、D10)中保存 0。

• 0＜(D1、D0)时，(D11、D10)中保存(D1、D0)+10000 的值。

7.13 数据块指令

BK+/BIN16 位块数据加法运算

BK+(P)

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行加法运算，将结果存 储到(d)中指定的软元件中。

-[BK+ (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储了加法运算数据的起始软元件	[-32768,+32767]	BIN16 位	ANY16_S	--
(s2)	加法运算数据或存储了加法运算数据的起始软元件	[-32768,+32767]	BIN16 位	ANY16_S	--
(d)	存储运算结果的起始软元件	-	BIN16 位	ANY16_S	--
(n)	加法运算数据个数	[0,65535]	BIN16 位	ANY16	UINT

使用的软元件

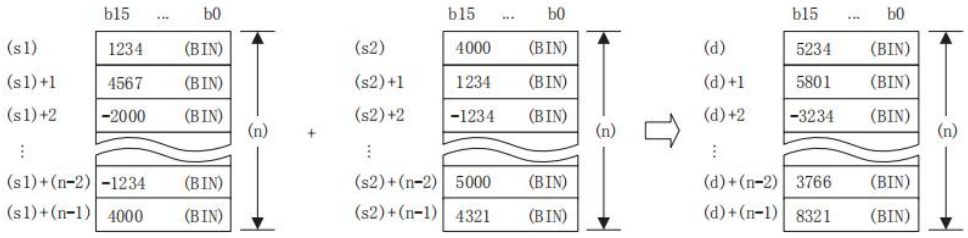
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BK+	参数 1															●	●	●	●	●						●	●
	参数 2															●	●	●	●	●			●	●		●	●
	参数 3															●	●	●	●	●						●	●
	参数 4																	●	●	●			●	●			●

功能

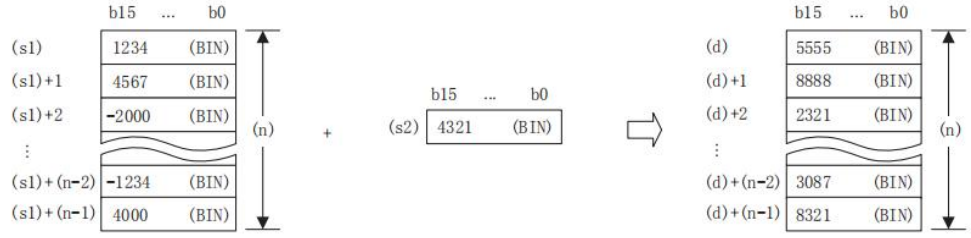
将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行加法运算，将运算结果存储到(d)中指定的软元件中。

- 块加法运算以 16 位单位进行。

(s2)中指定了软元件的情况下(有符号指定时)

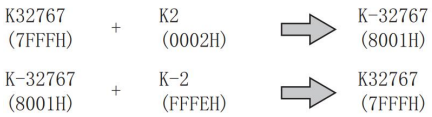


(s2)中指定了常数的情况下(有符号指定时)



- 运算结果发生了下溢及上溢时，其情况如下所示。在此情况下，进位标志不变为 ON。

指定有符号的情况下：



指定 无符号的情况下：

错误码

错误码	内容
4084H	(s1)或(s2)开始的(n)点的软元件范围与(d)开始的(n)点的软元件范围有部分一致(重复)。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

例如以上梯形图

当 X0 为 ON 时，将从 D100 开始的软元件数据(软元件点数为 D0 中所保存的数值)，和从 D150 开始的软元件数据(软元件点数为 D0 中所保存的数值)进行加法运算，并将其结果保存到 D200 以后中的程序。

DBK+/BIN32 位块数据加法运算

DBK+(P)

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行加法运算，将结果存 储到(d)中指定的软元件中。

-[DBK+ (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储了加法运算数据的起始软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(s2)	加法运算数据或存储了加法运算数据的起始软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(d)	存储运算结果的起始软元件	-	BIN32 位	ANY32_S	--
(n)	加法运算数据个数	[0,65535]	BIN32 位	ANY32_S	UDINT

使用的软元件

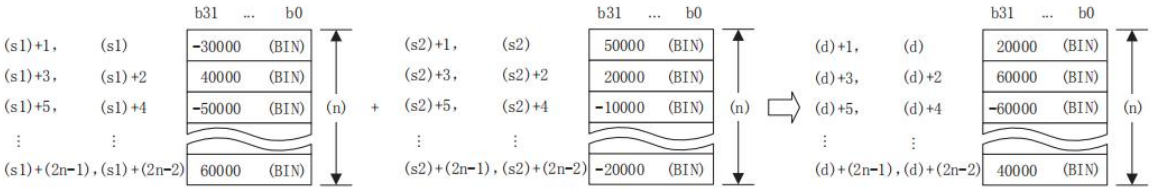
指令	参数	软元件																					偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBK+	参数 1															●	●	●	●	●	●	●				●	●
	参数 2															●	●	●	●	●	●	●	●	●		●	●
	参数 3															●	●	●	●	●	●	●				●	●
	参数 4																	●	●	●			●	●			●

功能

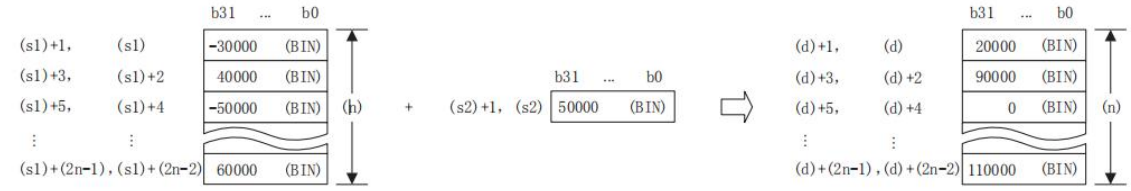
将(s1)中指定的软元件开始的(n)点的BIN32位数据与(s2)中指定的软元件开始的(n)点的BIN32位数据进行加法运算，将运算结果存储到(d)中指定的软元件中。

- 块加法运算以 32 位单位进行。

(s2)中指定了软元件的情况下(有符号指定时)

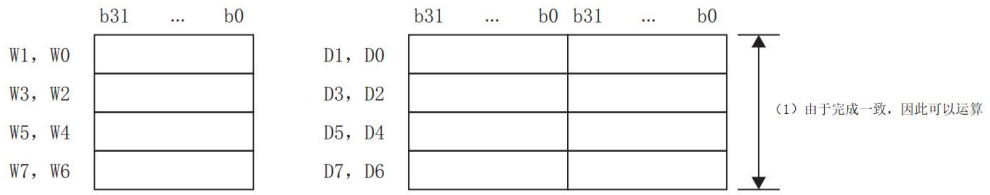


(s2)中指定了常数的情况下(有符号指定时)

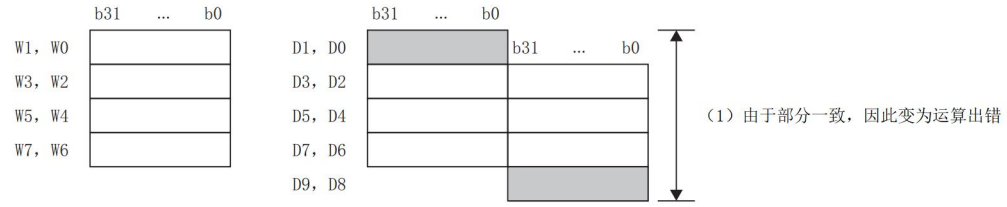


- (s1)或(s2)与(d)以同一软元件(完全一致)指定的情况下，可以运算。但是，(s1)或(s2)开始的(n)点的软元件范围与(d)开始的(n)点的软元件范围有部分一致(重复)的情况下，将变为出错状态。

例：(s2)与(d)开始的 4 点的软元件完全一致的情况下



(s2)、(d)开始的 4 点的软元件有部分一致的情况下



• (n)中指定的值为 0 的情况下将变为无处理。

• 运算结果发生了下溢及上溢时，其情况如下所示。在此情况下，进位标志不变为 ON。

指定有符号的情况下：

K2147483647 (7FFFFFFFH) + K2 (00000002H) ⇨ K-2147483647 (80000001H)

K-2147483647 (80000001H) + K-2 (FFFFFFFEH) ⇨ K2147483647 (7FFFFFFFH)

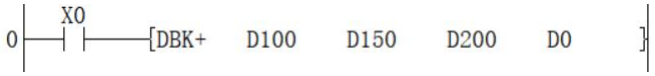
指定无符号的情况下：

K4294967295 (FFFFFFFFH) + K1 (00000001H) ⇨ K0 (00000000H)

错误码

错误码	内容
4084H	(s1)或(s2)开始的(n)点的软元件范围与(d)开始的(n)点的软元件范围有部分一致(重复)； (n)超出范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



例如以上梯形图

当 X0 为 ON 时，将从 D100 开始的软元件数据(软元件点数为 D0 中所保存的数值)，和从 D150 开始的软元件数据(软元件点数为 D0 中所保存的数值)进行加法运算，并将其结果保存到 D200 以后中的程序。

BK-/BIN16 位块数据减法运算

BK-(P)

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行减法运算，将结果存储到(d)中指定的软元件中。

-[BK- (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储了被减数据的起始软元件	[-32768,+32767]	BIN16 位	ANY16_S	--
(s2)	减数数据或存储了减数数据的起始软元件	[-32768,+32767]	BIN16 位	ANY16_S	--
(d)	存储运算结果的起始软元件	-	BIN16 位	ANY16_S	--
(n)	减法运算数据个数	[0,65535]	BIN16 位	ANY16	UINT

使用的软元件

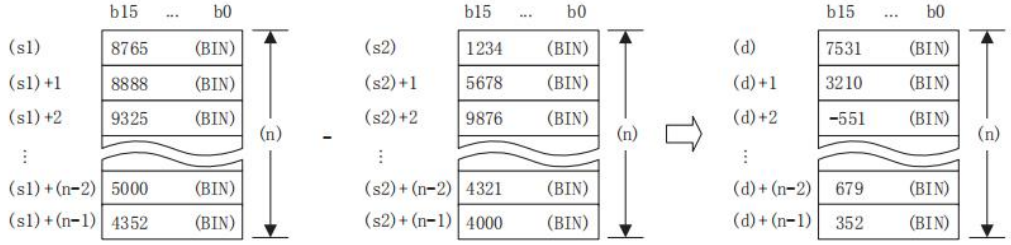
指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BK-	参数 1															●	●	●	●	●						●	●
	参数 2															●	●	●	●	●			●	●		●	●
	参数 3															●	●	●	●	●						●	●
	参数 4																	●	●	●			●	●			●

功能

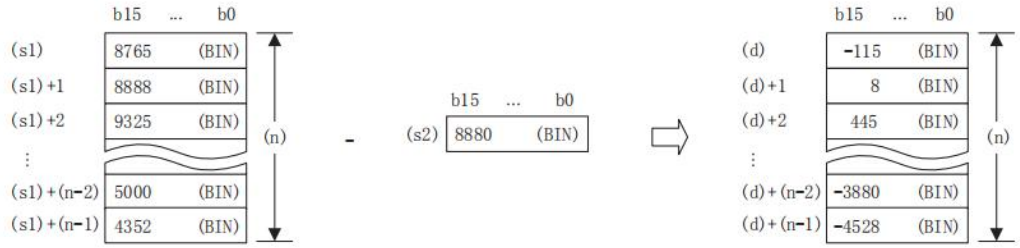
将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行减法运算，将运算 结果存储到(d)中指定的软元件中。

- 块减法运算以 16 位单位进行。

(s2)中指定了软元件的情况下



(s2)中指定了常数的情况下



- 运算结果发生了下溢及上溢时，其情况如下所示。在此情况下，进位标志不变为 ON。

  指定有符号的情况下：

K-32767 (8001H) - K2 (0002H) ⇨ K32766 (7FFEH)

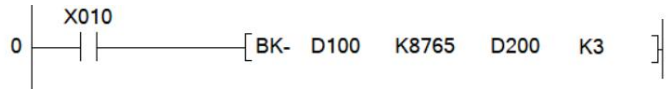
K32767 (7FFFH) - K-2 (FFFEH) ⇨ K-32767 (8001H)

指定无符号的情况下：

错误码

错误码	内容
4084H	(s1)或(s2)开始的(n)点的软元件范围与(d)开始的(n)点的软元件范围有部分一致(重复)。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



如以上梯形图所示：

当 X010 为 ON 时，从 D100 开始的 3 点数据和常数 8765 相减后，将其结果保存到 D200 以后的软元件中的程序。

DBK-/BIN32 位块数据减法运算

DBK-(P)

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行减法运算，将结果存 储到(d)中指定的软元件中。

-[DBK- (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储了被减数据的起始软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(s2)	减数数据或存储了减数数据的起始软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(d)	存储运算结果的起始软元件	-	BIN32 位	ANY32_S	--
(n)	减法运算数据个数	[0,65535]	BIN32 位	ANY32_S	UDINT

使用的软元件

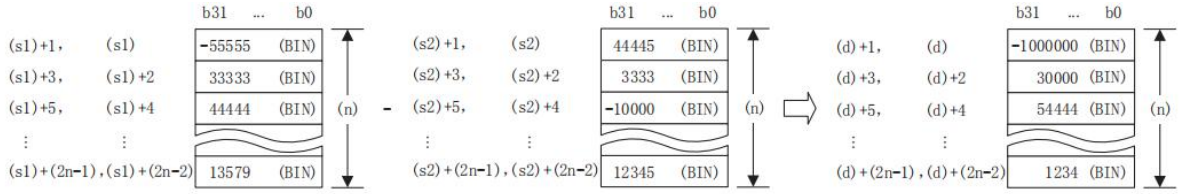
指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBK-	参数 1															●	●	●	●	●	●	●				●	●
	参数 2															●	●	●	●	●	●	●	●	●		●	●
	参数 3															●	●	●	●	●	●	●				●	●
	参数 4																	●	●	●			●	●			●

功能

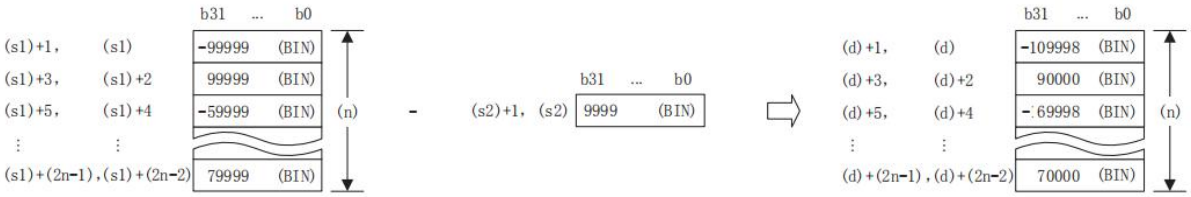
将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行减法运算，将运算结果存储到(d)中指定的软元件中。

- 块减法运算以 32 位单位进行。

(s2)中指定了软元件的情况下(有符号指定时)

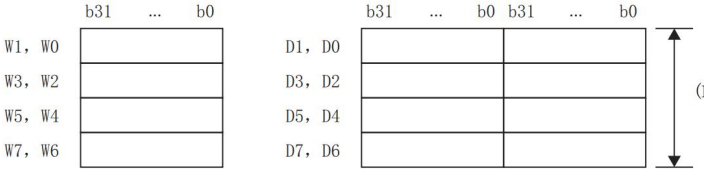


(s2)中指定了常数的情况下(有符号指定时)

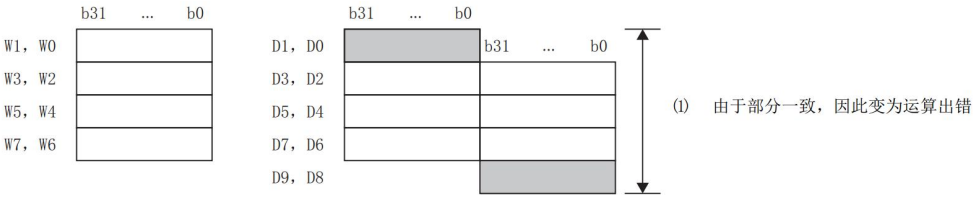


- (s1)或(s2)与(d)以同一软元件(完全一致)指定的情况下，可以运算。但是，(s1)或(s2)开始的(n)点的软元件范围与(d)开始的(n)点的软元件范围有部分一致(重复)的情况下，将变为出错状态。

例： (s2)与(d)开始的 4 点的软元件完全一致的情况下



(s2)、(d)开始的 4 点的软元件有部分一致的情况下



- (n)中指定的值为 0 的情况下将变为无处理。
- 运算结果发生了下溢及上溢时，其情况如下所示。在此情况下，进位标志不变为 ON。

  指定有符号的情况下：

K2147483647 (7FFFFFFFH) − K-2 (FFFFFFFEH) ⇨ K-2147483647 (80000001H)

K-2147483647 (80000001H) − K2 (00000002H) ⇨ K2147483647 (7FFFFFFFH)

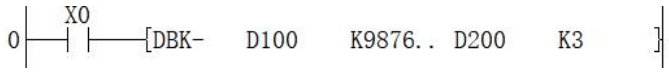
指定无符号的情况下：

K0 (00000000H) − K1 (00000001H) ⇨ K4294967295 (FFFFFFFFH)

错误码

错误码	内容
4084H	(s1)或(s2)开始的(n)点的软元件范围与(d)开始的(n)点的软元件范围有部分一致(重复)； (n)超出范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



如以上梯形图所示：

当 X010 为 ON 时，从 D100 开始的 3 点数据和常数 987654321 相减后，将其结果保存到 D200 以后的软元件中的程序。

BKCMP=/BIN16 位块数据比较

BKCMP=(P)

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[BKCMP= (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件。	[-32768, +32767]	BIN16 位	ANY16_S	--
(s2)	存储比较源数据的软元件。	[-32768, +32767]	BIN16 位	ANY16_S	--
(d)	存储比较结果的起始软元件。	-	位	ANY_BOOL	--
(n)	比较的数据数。	[0, 65535]	BIN16 位	ANY16	UINT

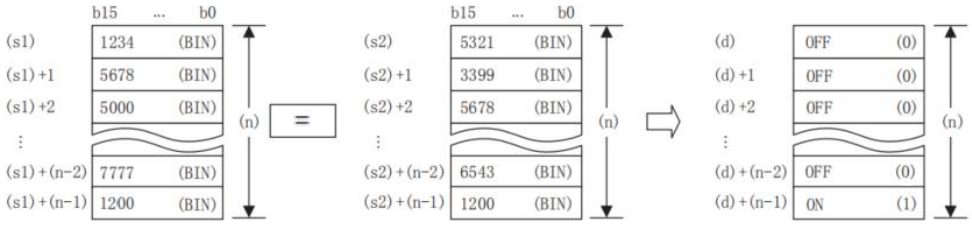
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BKCMP=	参数 1															●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●						●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

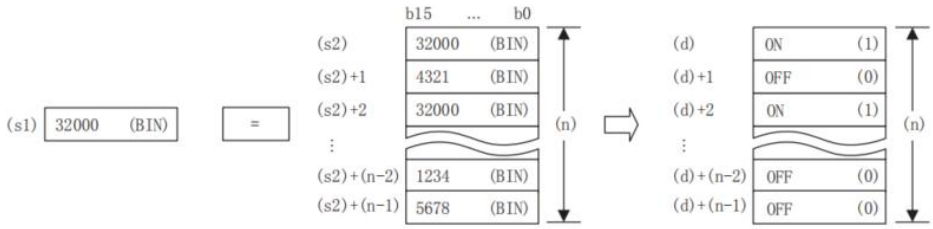
功能

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将比较结果 存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算是以 16 位单位进行。
- (s1)可以指定直接常数。



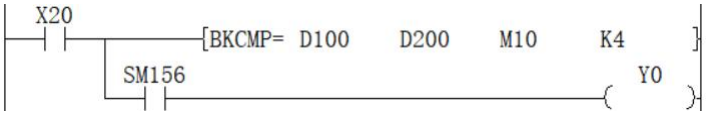
指令符号	条件	比较结果
BKCMP=	(s1)=(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 BKCMP=指令对 D100 开始的 4 点 16 位数据(BIN)和 D200 开始的 4 点 16 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

DBKCMP=/BIN32 位块数据比较

DBKCM=(P)

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[DBKCMP= (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-2147483648, 2147483647]	BIN32 位	ANY32_S	--
(s2)	存储比较源数据的软元件	[-2147483648, 2147483647]	BIN32 位	ANY32_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0, 65535]	BIN32 位	ANY32_S	UDINT

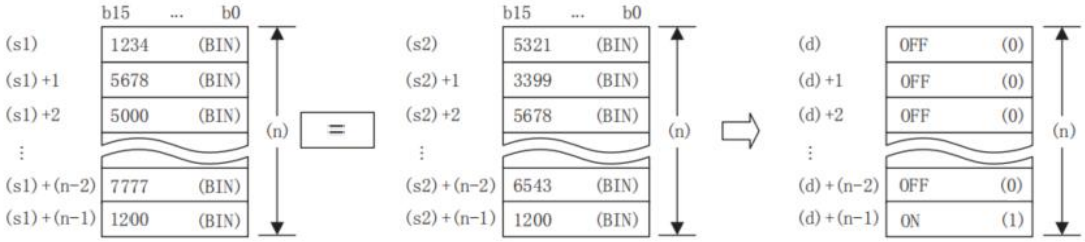
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBKCMP=	参数 1															●	●	●	●	●	●	●	●	●		●	●
	参数 2															●	●	●	●	●	●	●				●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

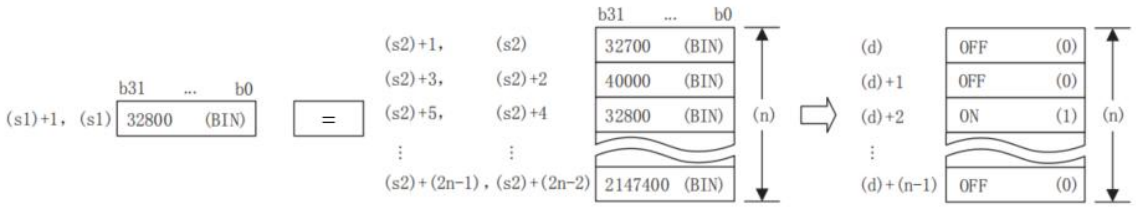
功能

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将比较结果存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算以 32 位单位进行。
- (s1)可以指定直接常数。



- (d)是在(s1)开始的(n)点的软元件范围及(s2)开始的(n)点的软元件范围以外进行指定。
- 各指令的比较运算结果如下所示。

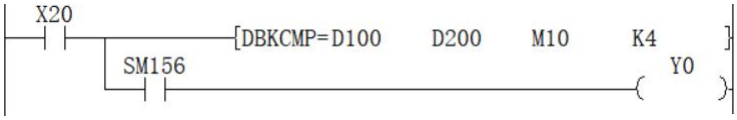
指令符号	条件	比较结果
DBKCMP=	(s1)=(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4084H	(n)超出范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 DBKCMP=指令对 D100 开始的 4 点 32 位数据(BIN)和 D200 开始的 4 点 32 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

BKCMP<>/BIN16 位块数据比较

BKCMP<>(P)

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

梯形图

-[BKCMP<> (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-32768,+32767]	BIN16 位	ANY16_S	--
(s2)	存储比较源数据的软元件	[-32768,+32767]	BIN16 位	ANY16_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0,65535]	BIN16 位	ANY16	UINT

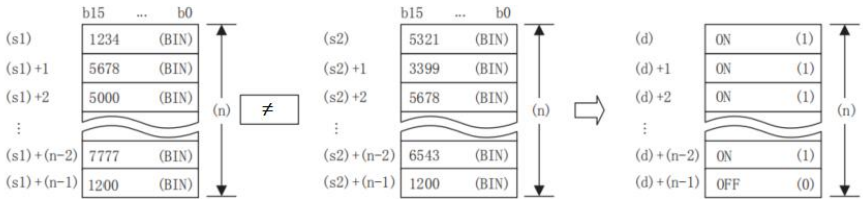
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BKCMP<>	参数 1															●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●						●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

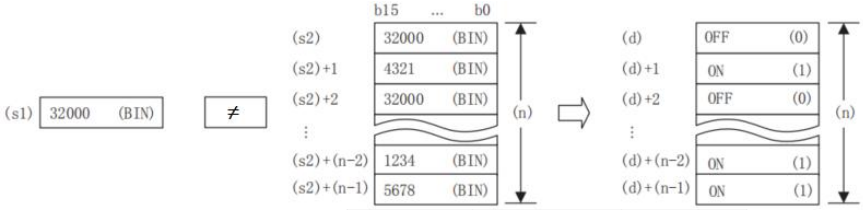
功能

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将比较结果 存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算是以 16 位单位进行。
- (s1)可以指定直接常数。



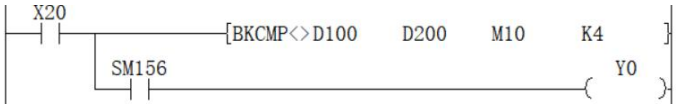
指令符号	条件	比较结果
BKCMP<>	(s1)≠(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 BKCMP<>指令对 D100 开始的 4 点 16 位数据(BIN)和 D200 开始的 4 点 16 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

DBKCMP<>/BIN32 位块数据比较

DBKCMP<>(P)

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[DBKCMP<> (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(s2)	存储比较源数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0,65535]	BIN32 位	ANY32_S	UDINT

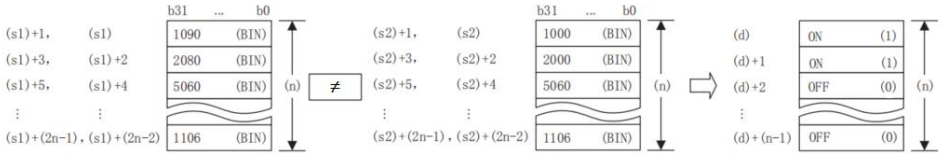
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBKCMP<>	参数 1															●	●	●	●	●	●	●	●	●		●	●
	参数 2															●	●	●	●	●	●	●				●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

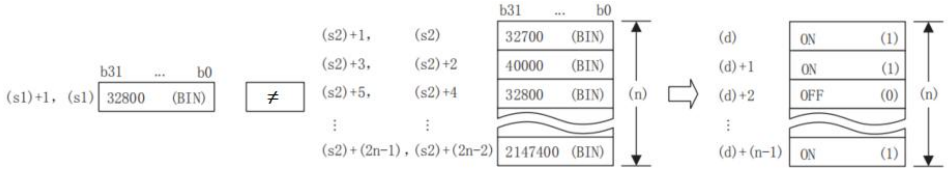
功能

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将比较结果存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算以 32 位单位进行。
- (s1)可以指定直接常数。



- (d)是在(s1)开始的(n)点的软元件范围及(s2)开始的(n)点的软元件范围以外进行指定。
- 各指令的比较运算结果如下所示。

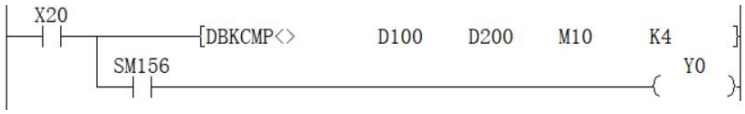
指令符号	条件	比较结果
DBKCMP<>	(s1)≠(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4084H	(n)超出范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 DBKCMP<>指令对 D100 开始的 4 点 32 位数据(BIN)和 D200 开始的 4 点 32 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

BKCMP>/BIN16 位块数据比较

BKCMP>(P)

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[BKCMP> (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-32768,+32767]	BIN16 位	ANY16_S	--
(s2)	存储比较源数据的软元件	[-32768,+32767]	BIN16 位	ANY16_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0,65535]	BIN16 位	ANY16	UINT

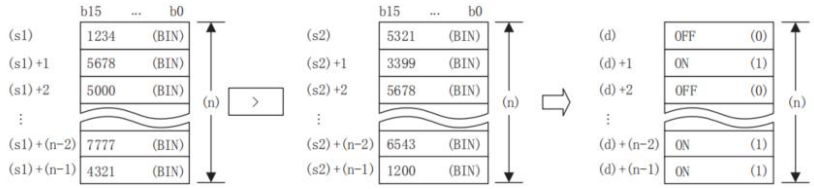
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BKCMP>	参数 1															●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●						●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

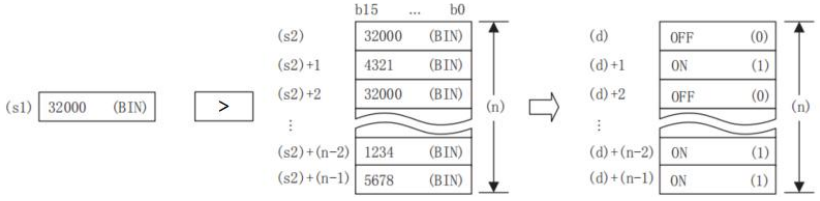
功能

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将比较结果 存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算是以 16 位单位进行。
- (s1)可以指定直接常数。



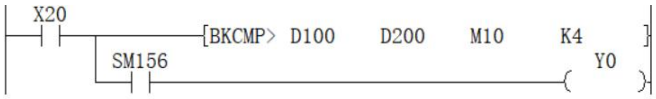
指令符号	条件	比较结果
BKCMP>	(s1)>(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 BKCMP>指令对 D100 开始的 4 点 16 位数据(BIN)和 D200 开始的 4 点 16 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

DBKCMP>/BIN32 位块数据比较

DBKCMP>(P)

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

梯形图

-[DBKCMP> (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(s2)	存储比较源数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0,65535]	BIN32 位	ANY32_S	UINT

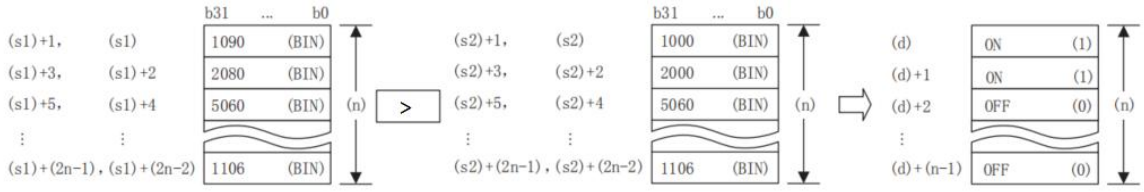
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBKCMP>	参数 1															●	●	●	●	●	●	●	●	●		●	●
	参数 2															●	●	●	●	●	●	●				●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

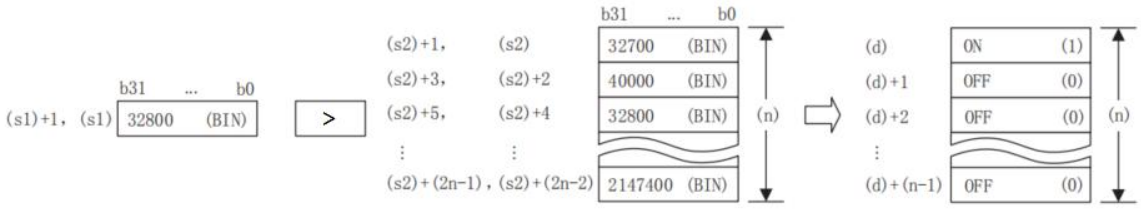
功能

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将比较结果存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算以 32 位单位进行。
- (s1)可以指定直接常数。



- (d)是在(s1)开始的(n)点的软元件范围及(s2)开始的(n)点的软元件范围以外进行指定。
- 各指令的比较运算结果如下所示。

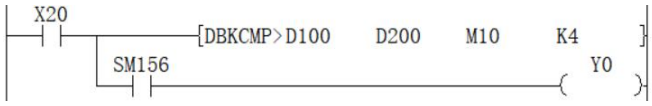
指令符号	条件	比较结果
DBKCMP>	(s1)>(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4084H	(n)超出范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 DBKCMP>指令对 D100 开始的 4 点 32 位数据(BIN)和 D200 开始的 4 点 32 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

BKCMP>=/BIN16 位块数据比较

BKCMP>=(P)

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[BKCMP>= (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-32768, +32767]	BIN16 位	ANY16_S	--
(s2)	存储比较源数据的软元件	[-32768, +32767]	BIN16 位	ANY16_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0, 65535]	BIN16 位	ANY16	UINT

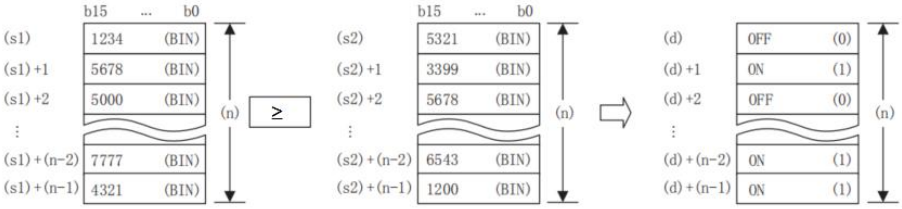
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BKCMP>=	参数 1															●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●						●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

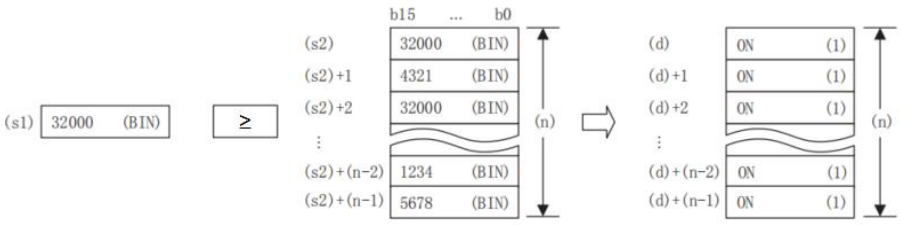
功能

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将比较结果 存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算是以 16 位单位进行。
- (s1)可以指定直接常数。



指令符号	条件	比较结果
BKCMP>=	(s1)>=(s2)	ON

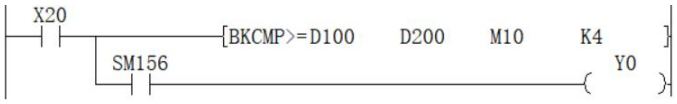
- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

福州富昌维控电子科技有限公司

示例



当 X020 为 ON 时，使用 BKCMP>=指令对 D100 开始的 4 点 16 位数据(BIN)和 D200 开始的 4 点 16 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

DBKCMP>=/BIN32 位块数据比较

DBKCMP>=(P)

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[DBKCMP>= (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(s2)	存储比较源数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0,65535]	BIN32 位	ANY32_S	UDINT

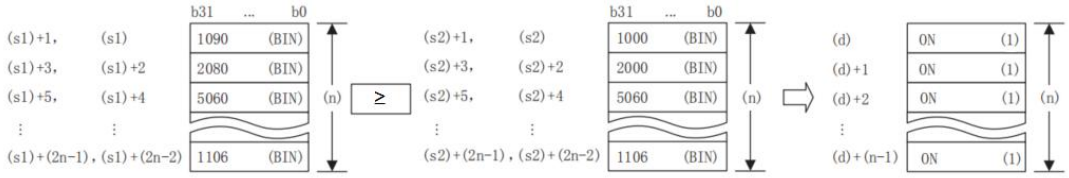
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBKCMP>=	参数 1															●	●	●	●	●	●	●	●	●		●	●
	参数 2															●	●	●	●	●	●	●				●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

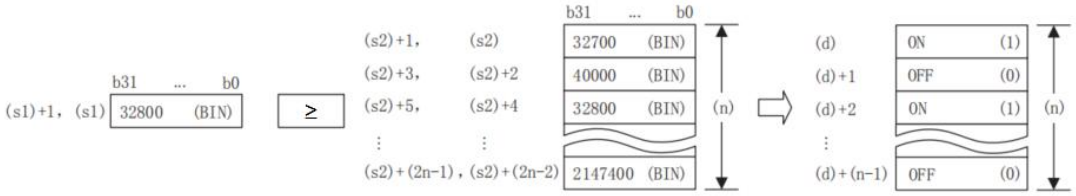
功能

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将比较结果存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算以 32 位单位进行。
- (s1)可以指定直接常数。



- (d)是在(s1)开始的(n)点的软元件范围及(s2)开始的(n)点的软元件范围以外进行指定。
- 各指令的比较运算结果如下所示。

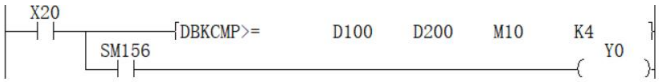
指令符号	条件	比较结果
DBKCMP>=	(s1)>=(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM349(块比较信号)将变为 ON。

错误码

错误码	内容
4084H	(n)超出范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 DBKCMP>=指令对 D100 开始的 4 点 32 位数据(BIN)和 D200 开始的 4 点 32 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

BKCMP</BIN16 位块数据比较

BKCMP<(P)

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[BKCMP< （s1） （s2） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
（s1）	比较数据或存储了比较数据的软元件	[-32768, +32767]	BIN16 位	ANY16_S	--
（s2）	存储比较源数据的软元件	[-32768, +32767]	BIN16 位	ANY16_S	--
（d）	存储比较结果的起始软元件	-	位	ANY_BOOL	--
（n）	比较的数据数	[0, 65535]	BIN16 位	ANY16	UINT

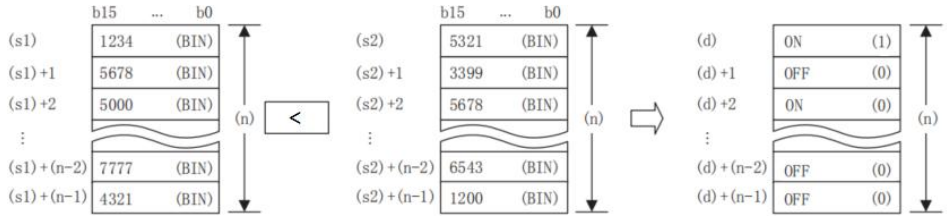
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BKCMP<	参数 1															●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●						●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

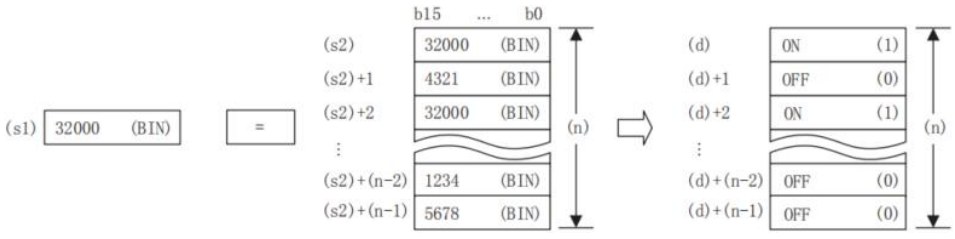
功能

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将比较结果 存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算是以 16 位单位进行。
- (s1)可以指定直接常数。



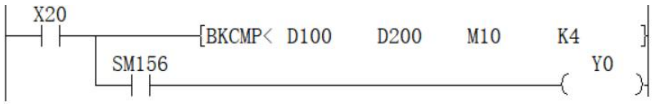
指令符号	条件	比较结果
BKCMP<	(s1)<(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 BKCMP<指令对 D100 开始的 4 点 16 位数据(BIN)和 D200 开始的 4 点 16 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

DBKCMP</BIN32 位块数据

DBKCMP<(P)

将(s)中指定的 HEX 代码数据中的 n 字符(位)转换为 ASCII 码后，存储到(d)中指定的软元件编号以后。

-[DBKCMP<(P) （s） （d） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
（s1）	比较数据或存储了比较数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
（s2）	存储比较源数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
（d）	存储比较结果的起始软元件	-	位	ANY_BOOL	--
（n）	比较的数据数	[0,65535]	BIN32 位	ANY32_S	UINT

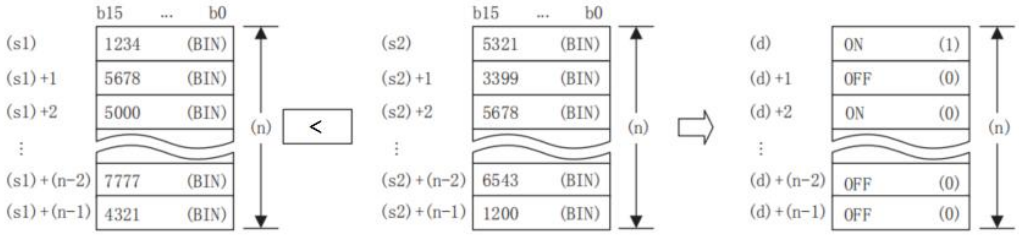
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DBKCMP<	参数 1															●	●	●	●	●	●	●	●	●		●	●
	参数 2															●	●	●	●	●	●	●				●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

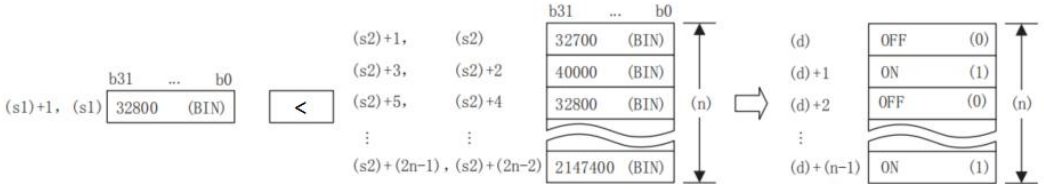
功能

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将比较结果存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算以 32 位单位进行。
- (s1)可以指定直接常数。



- (d)是在(s1)开始的(n)点的软元件范围及(s2)开始的(n)点的软元件范围以外进行指定。
- 各指令的比较运算结果如下所示。

指令符号	条件	比较结果
DBKCMP<	(s1)>=(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM349(块比较信号)将变为 ON。

错误码

错误码	内容
4084H	(n)超出范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

当 X020 为 ON 时，使用 DBKCMP<指令对 D100 开始的 4 点 32 位数据(BIN)和 D200 开始的 4 点 32 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

BKCMP<=/BIN16 位块数据比较

BKCMP<=(P)

将(s1)中指定的软元件开始的(n)点的 BIN16 位数据与(s2)中指定的软元件开始的(n)点的 BIN16 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[BKCMP<= (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-32768, +32767]	BIN16 位	ANY16_S	--
(s2)	存储比较源数据的软元件	[-32768, +32767]	BIN16 位	ANY16_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0, 65535]	BIN16 位	ANY16	UINT

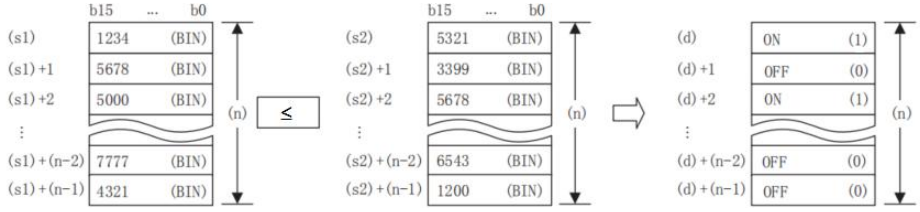
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BKCMP<=	参数 1															●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●						●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

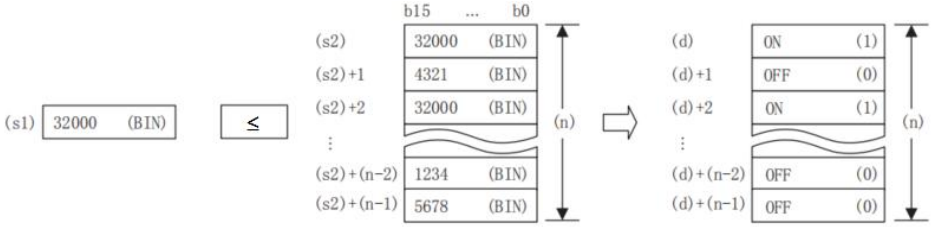
功能

将(s1)中指定的软元件开始的(n)点的BIN16位数据与(s2)中指定的软元件开始的(n)点的BIN16位数据进行比较，将比较结果 存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算是以 16 位单位进行。
- (s1)可以指定直接常数。



- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

指令符号	条件	比较结果
BKCMP<=	(s1)>=(s2)	ON

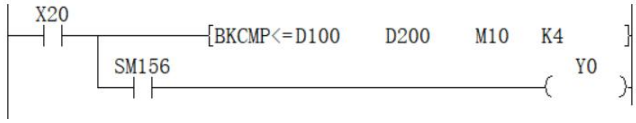
- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

福州富昌维控电子科技有限公司

示例



当 X020 为 ON 时，使用 BKCMP<=指令对 D100 开始的 4 点 16 位数据(BIN)和 D200 开始的 4 点 16 位数据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

DBKCMP<=/BIN32 位块数据比较

DBKCMP<=(P)

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将运算结果存储到(d)中指定的软元件中。

-[DBKCMP<= (s1) (s2) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	比较数据或存储了比较数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(s2)	存储比较源数据的软元件	[-2147483648,2147483647]	BIN32 位	ANY32_S	--
(d)	存储比较结果的起始软元件	-	位	ANY_BOOL	--
(n)	比较的数据数	[0,65535]	BIN32 位	ANY32_S	UDINT

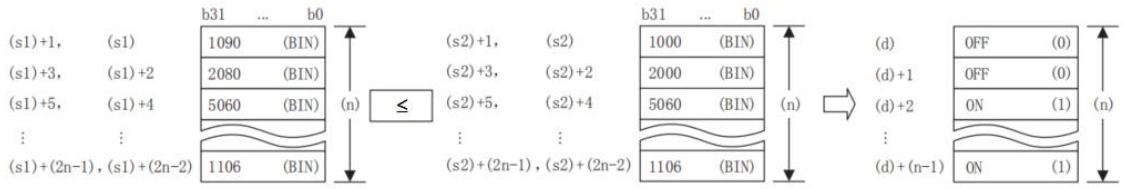
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BKCMP<=	参数 1															●	●	●	●	●	●	●	●	●		●	●
	参数 2															●	●	●	●	●	●	●				●	●
	参数 3		●	●	●	●					●															●	●
	参数 4																	●	●				●	●			●

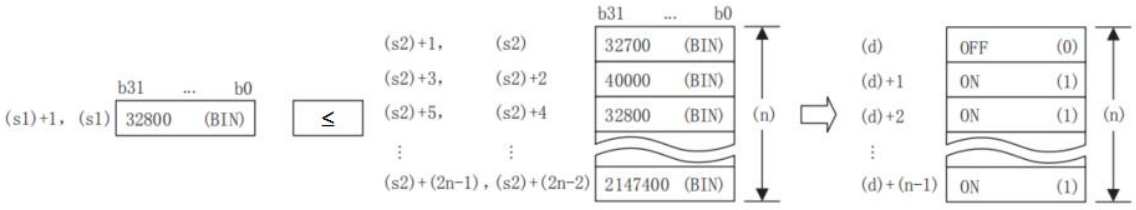
功能

将(s1)中指定的软元件开始的(n)点的 BIN32 位数据与(s2)中指定的软元件开始的(n)点的 BIN32 位数据进行比较，将比较结果存储到(d)中指定的软元件开始的(n)点中。

- (d)中指定的软元件开始的(n)点的相应软元件在比较条件成立时变为 ON、不成立时变为 OFF。



- 比较运算以 32 位单位进行。
- (s1)可以指定直接常数。



- (d)是在(s1)开始的(n)点的软元件范围及(s2)开始的(n)点的软元件范围以外进行指定。
- 各指令的比较运算结果如下所示。

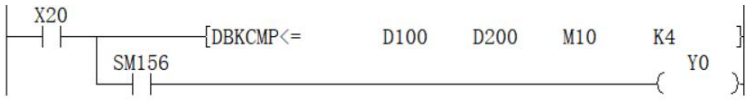
指令符号	条件	比较结果
DBKCMP<=	(s1)>=(s2)	ON

- (d)开始的(n)点中存储的比较运算结果全部为 ON(1)时，SM156(块比较信号)将变为 ON。

错误码

错误码	内容
4084H	(n)超出范围时。
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例



当 X020 为 ON 时，使用 DBKCMP<=指令对 D100 开始的 4 点 32 位数据(BIN)和 D200 开始的 4 点 32 位数。

据(BIN)进行比较，并将其结果保存到 M10 开始的 4 点软元件中的程序。

此外，比较结果(M10 开始的 4 点)全部为 ON(1)时，Y000 置 ON。

7.14 数据表操作指令

SFRD/移位读出

SFRD(P)

用于先入先出及控制的数据读取指令。

-[SFRD （s）(d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了数据的起始字软元件编号 （起始为指针，数据从（s）+1 开始）	-	有符号 BIN16 位	ANY16	--
(d)	存储先出的数据的字软元件编号	-	有符号 BIN16 位	ANY16	INT
(n)	应指定为存储的数据的点数+1 的值。+1 为指针	[2,512]	有符号 BIN16 位	ANY16	INT

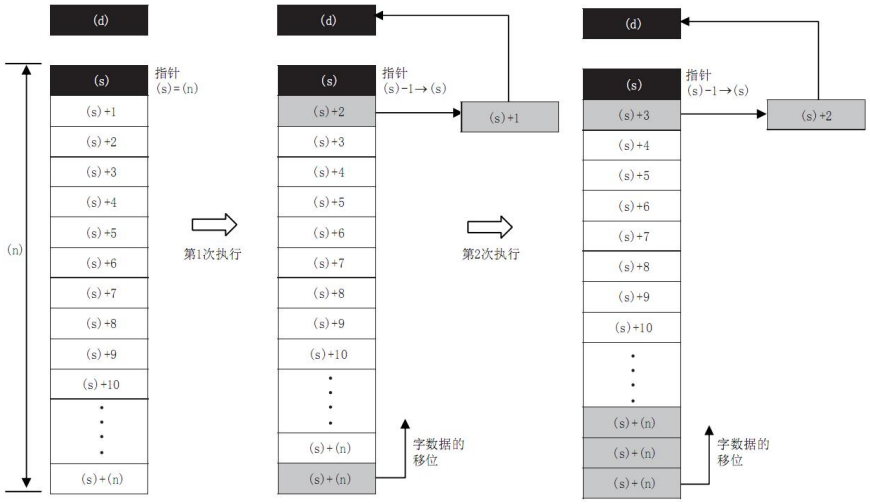
使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SFRD	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将通过 SFWR 指令依次写入的（s）+1 传送到（d）后，从（s）+1 将（n）-1 点各向上移位 1 字。（s）的存储的数据数-1。

• （s）+1 的内容传送（读取）到（d）。与此同时，指针（s）的内容减少，数据各向上移位 1 字。（在连续执行型指令 SFRD 中每个运算周期将移位，因此应使用脉冲执行型指令 SFRDP 编程）。



相关软元件

软元件	名称	内容
SM153	零位	数据的读出，通常从(s)+1 开始执行， 但是指针(s)的内容为 0 时，零位标志 SM153 动作。

注意事项

• 执行读取后的数据不会因为读取而使（s）+（n）的内容变化。

• 连续执行型（SFRD）指令的情况下，每个扫描时间（运算周期）将依次读取，但（s）+（n）的内容不会变化。

• 指针（s）为 0 时，将变为无处理，（d）的内容不会变化。

错误码

错误码	内容
4084H	(n)中设置的数值在下述以外时。2≤(n)≤512
	(s)中指定了负值。
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(s)、(d)中指定的软元件超出相应软元件的范围时。

示例

下面例举使用移位写入（SFWR）、移位读出（SFRD）指令的事例加以说明。

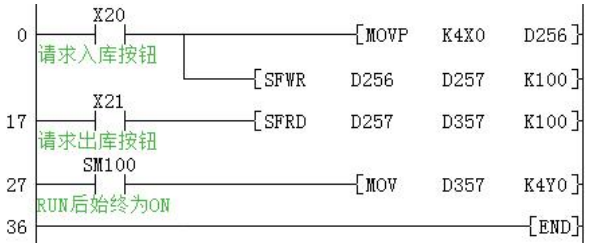
（1）动作内容

①登记产品编号的同时，为了能实现先入库的物品先出库这样的先入先出原则，以下介绍了输出当前应该取的产品编号的梯形图实例。

②产品编号为 4 位数以下的 16 进制数，最大库存量在 99 点以下。

（2）程序

①程序 1

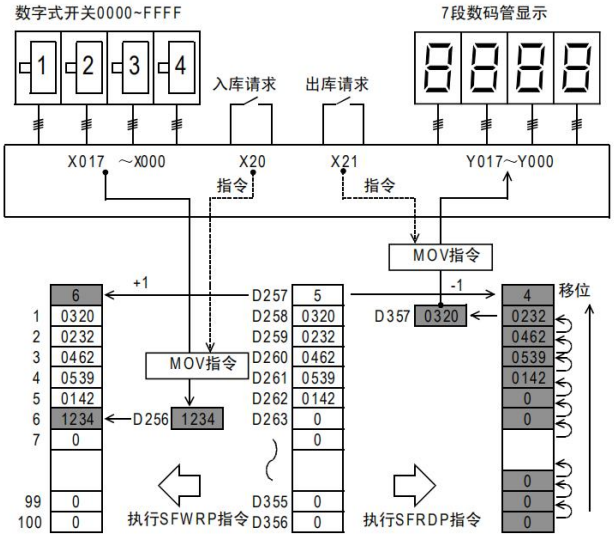


从 X0~X17 输入产品编号，并将其传送到 D256 中

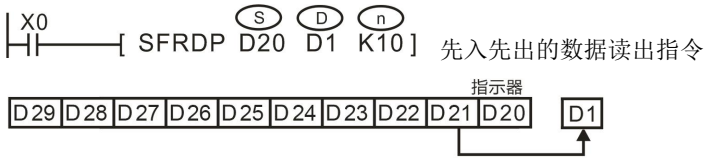
指针 D257 保存产品编号用的数据寄存器 D258~D356（99 点）

根据出库要求将先入的产品编号输出到 D357 中

将应该取出的产品编号以 16 进制 4 位数输出到 Y0~Y17



②程序 2



先入先出的数据读出指令

X0 由 OFF 到 ON 本指令动作依照下列编号 1~3 动作。（D10 内容保持不变），

1：D2 的内容被读出传送至 D20 当中，

2：D10~D3 全部往右移位一个寄存器，

3：指针 D1 内容减 1。

POP/从数据表的后入数据读取

POP(P)

读取通过先入先出/先入后出控制用的移位写入指令（SFWR）写入的最后数据。

-[POP （s）（d）（n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了先入的数据（包含指针数据）的起始软元件编号（存储了数据的起始字软元件编号）	-	有符号 BIN16 位	ANY16	--
(d)	存储后出的数据的软元件编号	-	有符号 BIN16 位	ANY16	--
(n)	存储的数据的点数	[2,512]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
POP	参数 1												●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

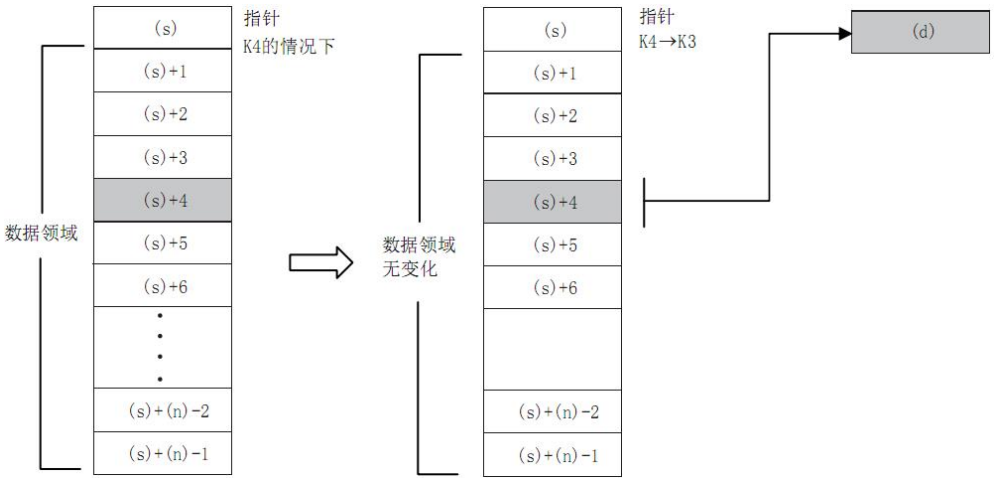
功能

• 对于“（s）～（s）+（n）-1”的字软元件，每次执行指令时会将“（s）+指令数据（s）”的软元件读取到（d）（将通过先入先出控制用的移位写入指令（SFWR）写入的最后数据读取到（d））。（n）指定 2~512。

• 指针数据（s）的值-1。

• 先入后出控制数据

	内容
（s）	指针数据（存储的数据的个数）
（s）+1	数据领域 （通过移位写入指令（SFWR）先入的数据）
（s）+2	
（s）+3	
.....	
（s）+（n）-3	
（s）+（n）-2	
（s）+（n）-1	



注意事项

• 如果以连续执行型进行 POP（P）指令编程，每个运算周期将对指令进行处理。因此有可能无法实现所期望的动作，应加以注意。通常进行 POP（P）指令编程时应使其以“脉冲执行型”或“脉冲化的指定触点”执行。

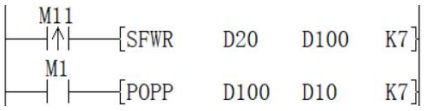
• 指针（s）的当前值为 0 的情况下，零标志 SM153 变为 ON，POP(P)指令将变为无处理。

• 指针（s）的当前为 1 的情况下，写入 0 到（s），零标志 SM153 变为 ON。

错误码

错误码	内容
4084H	(s)>(n)-1 时。
	(s)<0 时。
	(n)中输入了超出指定范围的数据。2≤(n)≤512
4085H	读应用指令(s)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(s)、(d)中指定的软元件超出相应软元件的范围时。

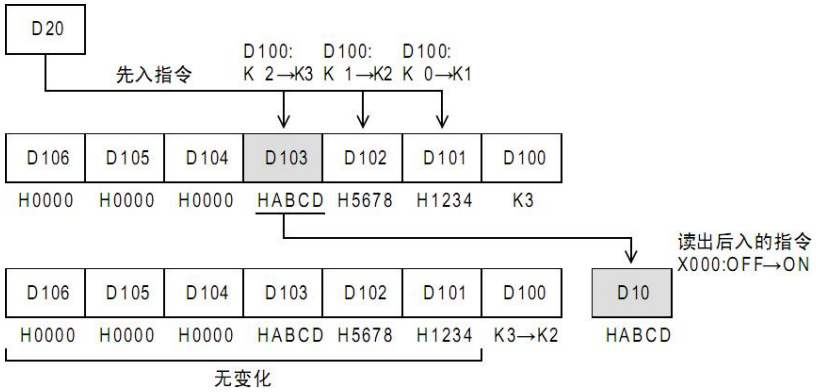
示例



M1 每次为 ON 时，对于 D101~D106 中被先输入的 D20 的值中，最后被保存的值都会保存到 D10 中，然后将数据保存数（指针 D100）减 1。

先输入的数据为下表中的内容时

指针	D100	K3
数据	D101	H1234
	D102	H5678
	D103	HABCD
	D104	H0000
	D105	H0000
	D106	H0000



SFWR/移位写入

SFWR(P)

用于先入先出及控制的数据写入指令。

-[SFWR （s）(d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储了希望先入的数据的字软元件编号	-	有符号 BIN16 位	ANY16	INT
(d)	存储数据并移位的起始字软元件编号（起始为指针，数据从（d）+1 开始）	-	有符号 BIN16 位	ANY16	--
(n)	应指定存储的数据的点数+1 的值	[2,512]	有符号 BIN16 位	ANY16	INT

使用的软元件

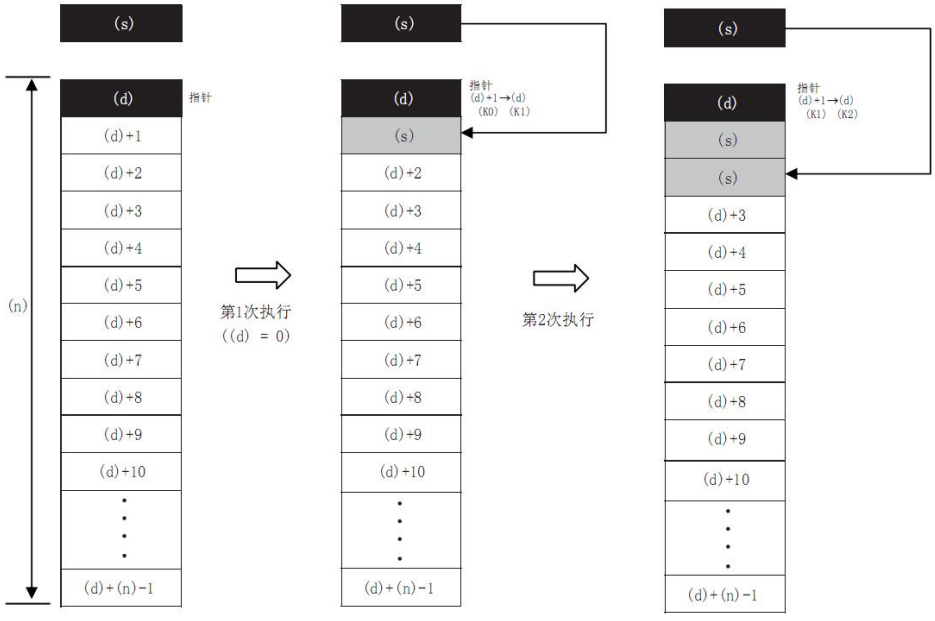
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SFWR	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 从（d）+1 开始，将（s）的内容依次写入（n）-1 点，（d）的存储数据+1.例如，（d）=0 的情况下写入到（d）+1，（d）=1 的情况下写入到（d）+2。

• 通过第 1 次执行，（s）的内容存储到（d）+1，变为（s）的值。

• 如果更改（s）的内容后执行第 2 次，该（s）的内容存储到（d）+2，（d）+2 的内容变为（s）（在连续执行型指令 SFWR 中每个运算周期将依次存储，因此应使用脉冲执行型指令 SFWRP 编程）。之后以此类推，从右侧依次顺延填入，数据的存储点数通过指针（d）的内容显示。



相关软元件

软元件	名称	内容
SM151	进位	指针（d）的内容超过(n)-1 时，变为无处理（不写入），且进位标志位 SM151 置 ON。

注意事项

• 连续执行型（SFWR）指令的情况下，每个扫描时间（运算周期）将依次存储（覆盖），应加以注意。

错误码

错误码	内容
4084H	(n)中设置的数值在 2≤(n)≤512 以外时。
	(d)中指定了负值时。
4085H	读应用指令(s)、(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例

下面例举使用移位写入（SFWR）、移位读出（SFRD）指令的事例加以说明。

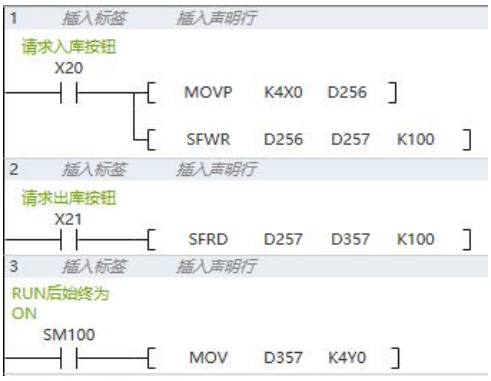
⑴动作内容

①登记产品编号的同时，为了能实现先入库的物品先出库这样的先入先出原则，以下介绍了输出当前应该取的产品编号的梯形图实例。

②产品编号为 4 位数以下的 16 进制数，最大库存量在 99 点以下。

⑵程序

①程序 1

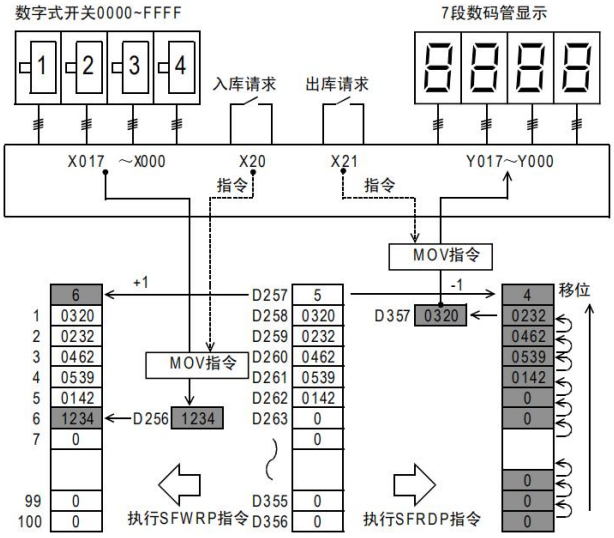


从 X0~X17 输入产品编号，并将其传送到 D256 中

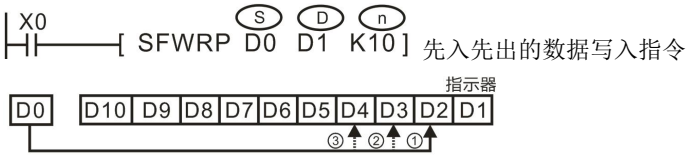
指针 D257 保存产品编号用的数据寄存器 D258~D356（99 点）

根据出库要求将先入的产品编号输出到 D357 中

将应该取出的产品编号以 16 进制 4 位数输出到 Y0~Y17



②程序 2



当 X0＝1 时，D0 的内容被存入 D2，D1 的内容变为 1。当 X0 再次从 OFF→ON 时，这个 D0 的内容被存入 D3, D1 的内容变为 2，以此类推。若 D1 的内容超过 n-1，则指令不处理，而进位标志 M8022 会置 1。

福州富昌维控电子科技有限公司

FINS/数据表的数据插入

FINS(P)

将（s）中指定的 BIN16 位数据插入到（d）中指定的数据表的第（n）号中。执行指令后，数据表的第（n）号开始的数据将逐个向下顺延。

-[FINS （s）（d）（n）]

内容、范围、数据类型

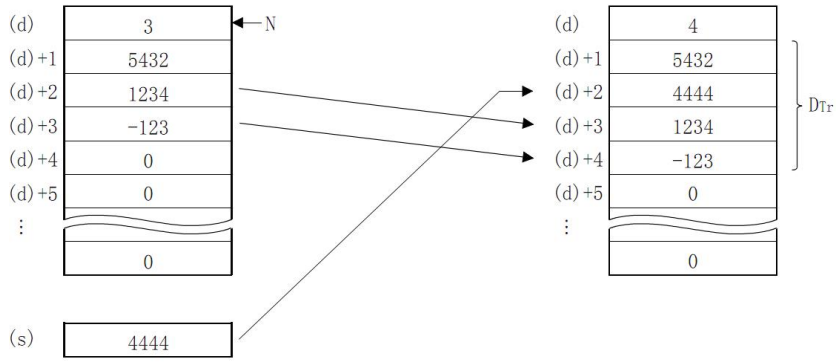
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储插入数据的软元件起始编号	-	有符号 BIN16 位	ANY16	--
(d)	表的起始编号	-	字	ANY16	--
(n)	插入的表位置	[1,512]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
FINS	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 2															●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（s）中指定的 BIN16 位数据插入到（d）中指定的数据表的第（n）号中。执行指令后，数据表的第（n）号开始的数据将逐个向下顺延。



N：数据存储数；

D_{Tr}：数据表范围；

（n）=2 的情况下，将被插入到（d）+2 中。

注意事项

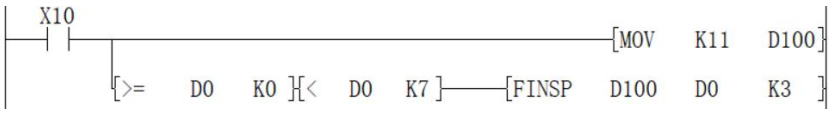
• 数据表中使用的软元件的范围由用户负责管理。

• 数据表的范围为数据存储数（d）之后的软元件（（d）+1）开始（d）个。

错误码

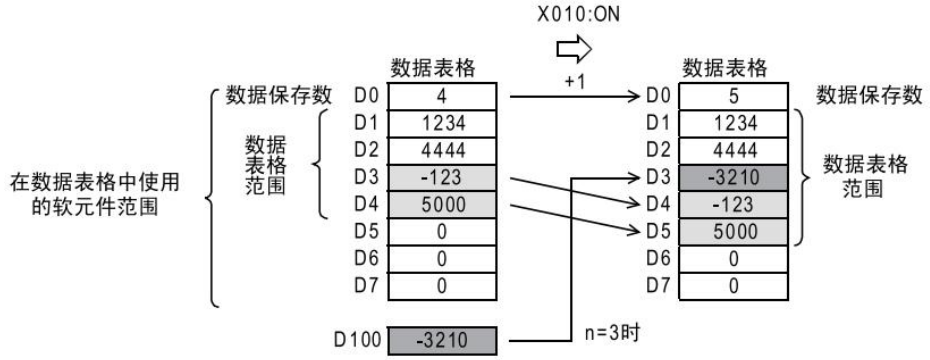
错误码	内容
4084H	(d)的值为 0 的状况下执行 FINS(P)指令时。
	(d)中表的存储数据超过 512。
	(n)中设置的数据在 1≤(n)≤512 以外时。
	执行 FINS(P)指令时，插入数据的表位置(n)大于数据存储数。
4085H	读应用指令(s)、(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(d)中指定的软元件超出相应软元件的范围时。

示例



当 X10=ON 时，在 D0~D4 的数据表格的第 3 号中插入 D100 的数据。

但是，当数据保存数超过 7 时，不执行 FINS（P）指令（数据表格中使用的软元件范围为 D0~D7）。



FDEL/数据表的数据删除

FDEL(P)

将（d）中指定的数据表的第（n）号的数据删除后，存储到（s）中指定的软元件中。执行指令后，数据表的第（n）+1 号以后的数据将逐个向前依次顺延。

-[FDEL （s）（d）（n）]

内容、范围、数据类型

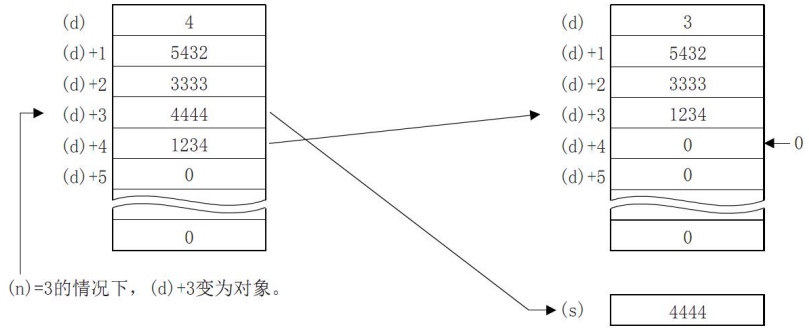
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	存储删除数据的软元件起始编号	-	有符号BIN16位	ANY16	--
(d)	表的起始编号	-	字	ANY16	--
(n)	删除的表位置	[1,512]	有符号BIN16位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
FDEL	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2															●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●			●	●		●	●

功能

• 将（d）中指定的数据表的第（n）号的数据删除后，存储到（s）中指定的软元件中。执行指令后，数据表的第（n）+1 号以后的数据将逐个向前依次顺延。



注意事项

• 数据表中使用的软元件的范围由用户负责管理。

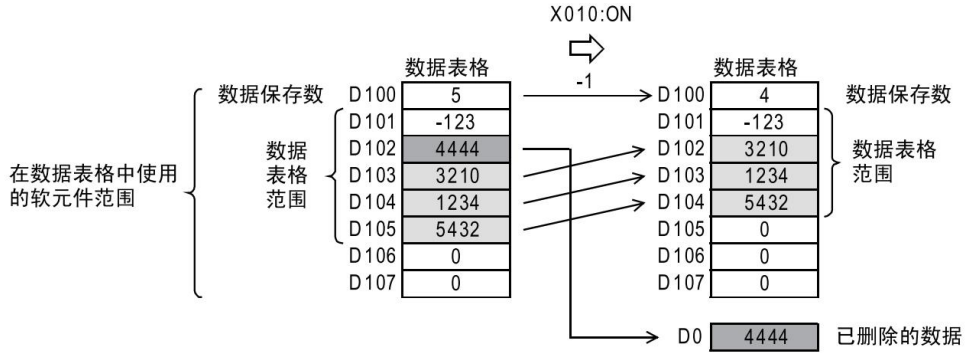
• 数据表的范围为数据存储数（d）之后的软元件（（d）+1）开始（d）个。

错误码

错误码	内容
4084H	(d)的值为0的状况下执行FDEL(P)指令时。
	(d)中表的存储数据超过512。
	(n)中设置的数据在下述以外时，1≤(n)≤512。
	执行FDEL(P)指令时，删除数据的表位置(n)大于数据存储数。
4085H	读应用指令(d)、(n)中指定的软元件超出相应软元件的范围时。
4086H	写应用指令(s)、(d)中指定的软元件超出相应软元件的范围时。

示例

当 X10 为 ON 时，删除 D100~D105 的数据表格中第 2 个数据，将删除的数据保存到 D0 中。但是，当数据保存数为 0 时，请勿执行 FDEL 指令。（数据表格中使用的软元件范围为 D100~D107。）



7.15 输入输出刷新指令

REF/输入输出刷新

REF(P)

对(s)中指定的软元件开始的 n 点进行刷新，进行外部输入的获取或输出。

-[REF (s) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	刷新的软元件起始编号	使用 X、Y 时：最低位的位数编号只能为 0 使用 HSC 时:HSC0~HSC7	位/双字	ANY_BOOL/ANY_32	--
(n)	刷新点数	使用 X、Y 时：只能是 8 的倍数 使用 HSC 时：1~8	无符号 BIN16 位	ANY16_U	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
REF	参数 1	●	●																			●					●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

功能

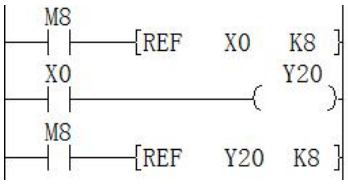
- 是 1 个扫描途中仅对相应的软元件进行刷新，进行外部输入的获取或输出的功能；
- 输入的获取及至外部的输出是仅在执行程序的 END 指令后批量地进行，因此 1 个扫描中不能向外部输出脉冲信号。执行 I/O 刷新指令时，程序执行途中相应的输入(X)或输出(Y)将被强制刷新，因此 1 个扫描中可以向外部输出脉冲信号；
- 可用在 FOR~NEXT 、CJ 等指令之间；
- 可用于中断子程序中进行输入输出刷新一获取最新的输入信息并及时输出运算结果；
- 实际的输入端口状态变化延迟决定于输入元件的滤波时间；
- 实际的输出端口状态变化延迟决定于输出元件（如继电器）的响应时间。输出刷新中的输出接点将在输出继电器（晶体管）应答时间后动作；
- 继电器输出型的应答滞后时间约为 10ms（最大 20ms），晶体管输出型高速输出口约 10 微妙、普通点输出口约 0.5ms；
- X0~X17 滤波时间设为 0 时依然会有一定的延迟；
- REF 指令还可以立即刷新高速计数器 HSC 软元件的值，正常使用时高速计数器的值每 100us 更新一次，若需要立即获取最新的高速计数值，则可以使用 REF 指令来进行一次计数刷新，指令执行完成后，HSC 软元件中的值就是最新的高速输入的脉冲数。

错误码

错误码	内容
4085H	(s) (n)读取地址超出软元件范围，注意(s)+(n)超出(s)对应的软元件的最大范围也会报错。
	(s)使用低位不是 0 的编号元件。
4084H	(s)使用 X、Y 时，(n)不是 8 的倍数或使用或(s)使用 HSC 时，(n)超出 K1~K8 的范围。
2582H	使用了 REF 指令刷新高速计数值，但没有 OUT HSC 指令开启该通道的高速计数器。

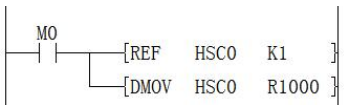
示例

(1) REF 对 X 输入或 Y 输出进行刷新



如上例子中,M8 打开后 X0~X7 就可快速更新输入信号。X0 触发 Y20 后,通过下一条 REF Y20 K8 指令快速输出 Y20~Y27。

(2) REF 对高速计数器 HSC 进行刷新



如上例子中，开启 M0 后，对高速计数器的当前输入脉冲数进行刷新，将最新的高速计数值存入 HSC0 中，并将此时刻的高速计数器值保存在 R1000 地址中。

REFF/输入刷新（带滤波器设定）

REFF(P)

将 X0~X17 的数字滤波的滤波效果临时改为(n)ms。(n)范围是 0~60ms。

-[REFF　(n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(n)	X0~X17 的滤波值	[0,60]	BIN16 位	ANY16	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
REFF	参数 1	●	●															●	●	●			●	●		●	●

功能

▪ 可编程控制器中，X0~X17 使用了数字滤波器，默认的滤波时间常数由 SD2280 SD2281 设定，通过 REFF 指令可将滤波临时改变为 0~60ms。

▪ 当使用了高速计数器，或 X 输入端中断功能，则相关端口的滤波时间自动为最短时间，无关的端口的滤波时间仍为原设定值。

▪ 亦可用 MOV 指令直接对 SD2280 SD2281 进行赋值改变滤波时间。但是这个指令不会改变 SD2280 SD2281 的值。

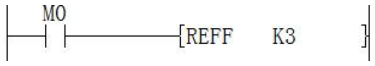
注意事项

▪ 在这条指令之前的 X 点滤波可能会不受控制（SD2280 SD2281 设置为 0 的情况指令之前的 X 点会完全不受控制）。

错误码

错误码	内容
4085H	(n)读取地址超出软元件范围。
4084H	(n)不在 0~60 范围内。

示例



开启 M0 后 REFF 指令之后的梯形图中 X0~X17 的滤波波都会临时变为 3ms，且 SD2280 SD2281 不会改变。

7.16 时机计测指令

DUTY/时钟脉冲发生指令

DUTY

将(d)中指定的用户用时机时钟输出目标(SM340~SM344)按照(n1)中指定的扫描数置为 ON，按照(n2)中指定的扫描数置为 OFF。

-[DUTY (n1) (n2) (d)]

内容、范围、数据类型

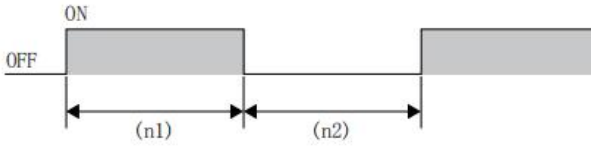
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(n1)	置为 ON 的扫描数	[0,32767]	无符号 BIN16	ANY16	UINT
(n2)	置为 OFF 的扫描数	[0,32767]	无符号 BIN16	ANY16	UINT
(d)	时机时钟输出目标的特殊寄存器	SM340~SM344	位	ANY_BOOL	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DUTY	参数 1											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	
	参数 3					●																				●	

功能

将(d)中指定的用户用时机时钟输出目标(SM340~344)按照(n1)中指定的扫描数置为 ON，按照(n2)中指定的扫描数置为 OFF。



SM340~SM344

(n1)：(n1)扫描

(n2)：(n2)扫描

- (d)中指定的时机时钟的输出目标特殊继电器应指定 SM340~SM344。
- 将扫描数的计数值存储到与(d)中指定的时机时钟的输出目标特殊继电器对应的 SD340~SD344 中。
- 关于扫描数的计数值 SD340~SD344，计数值变为(n1)+(n2)或在指令输入(指令)置为 ON 时复位。

输出时机时钟的特殊继电器(d)	扫描数计数用软元件
SM340	SD340
SM341	SD341
SM342	SD342
SM343	SD343
SM344	SD344

- 在指令输入的上升沿开始动作，时机时钟的输出目标特殊继电器通过 END 指令置为 ON/OFF。而即使指令输入断开，动作也不停止。STOP 时、电源 OFF 时停止。
- 将(n1)、(n2)设置为 0 的情况如下所示。

(n1)、(n2)的状态	(d)的 ON/OFF 状态
(n1)=0、(n2) 0	固定为(d)=OFF
(n1)>0、(n2)=0	固定为(d)=ON

• 相关软元件如下所示。

特殊继电器	名称	内容
SM340	时机时钟输出 1	DUTY 指令的时机时钟输出
SM341	时机时钟输出 2	
SM342	时机时钟输出 3	
SM343	时机时钟输出 4	
SM344	时机时钟输出 5	

特殊寄存器	名称	内容
SD340	时机时钟输出 1 用扫描数计数	DUTY 指令的时机时钟输出 1 用扫描数计数值
SD341	时机时钟输出 2 用扫描数计数	DUTY 指令的时机时钟输出 2 用扫描数计数值
SD342	时机时钟输出 3 用扫描数计数	DUTY 指令的时机时钟输出 3 用扫描数计数值
SD343	时机时钟输出 4 用扫描数计数	DUTY 指令的时机时钟输出 4 用扫描数计数值
SD344	时机时钟输出 5 用扫描数计数	DUTY 指令的时机时钟输出 5 用扫描数计数值

注意事项

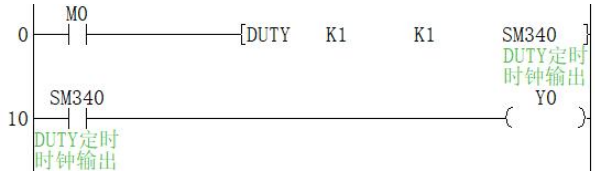
DUTY 指令最多可使用 5 次(点)。但是，无法在多个 DUTY 指令中使用相同的时机时钟输出目标。

错误码

错误码	内容
4084H	(n1)(n2) 写入值超出范围。
4085H	(n1)(n2) 软元件地址超出范围。
4086H	(d) 不在 SM340~SM344 间。
408EH	多条 DUTY 指令的(d)使用了同一 SM 软元件。

示例

使用 DUTY 指令，让 Y0 每周期都翻转一次。



置位 M0，SM340 将一个周期为 ON，一个周期为 OFF。

福州富昌维控电子科技有限公司

7.17 随机数指令

RND/随机数指令

RND(P)

发生 0~32767 的伪随机数，将该值作为随机数存储到(d)中指定的软元件中。

-[RND(P)　(d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d)	存储随机数的软元件起始编号	--	有符号 BIN16	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RND	参数 1												●	●	●	●	●	●	●	●						●	●

功能

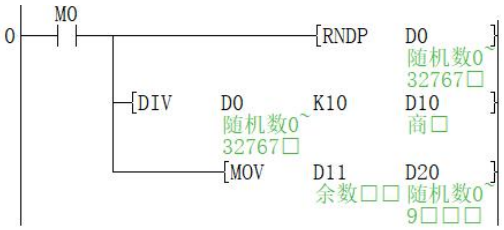
- 发生 0~32767 的伪随机数，将该值作为随机数存储到(d)中指定的软元件中。

错误码

错误码	内容
4086H	(d) 写入地址超出软元件范围。

示例

产生 0~9 的伪随机数。



开启 M0，产生一个 0~9 之间的伪随机数。

7.18 优选指令

DEXMN/优选指令

DEXMN(P)

通过计算，选出多个给定值中的和最接近目标值的给定值的位置。

-[DEXMN (s1) (s2) (s3) (d1) (d2)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	输入数据参数数组起始软元件编号		表格类型	LIST	--
(s2)	选择数据个数的最大值和输出模式起始软元件编号		表格类型	LIST	--
(s3)	目标值	[0,16777215]	无符号BIN32	ANY32_U	--
(d1)	选择结果数组起始软元件编号		表格类型	LIST	--
(d2)	运行结果数组起始软元件编号		表格类型	LIST	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DEXMN	参数 1																	●	●								●
	参数 2																	●	●								●
	参数 3																	●	●								●
	参数 4																	●	●								●
	参数 5																	●	●								●

功能

1、指令功能说明

从 S1 给定的数据组中，选择出个数小于等于 S2（选择数据个数的最大值）且和值最接近 S3（目标值）的数据组合。选择的结果按照 S1 数组对应的位置，存放在 D1 数组中。

指令执行的错误码存放在 D2，选择的个数存放在 D2+2，选择出的数组和值与目标值的差值存放在 D2+4。

2、详细参数说明

输入单参数 S1				
单元	字节数	功能	参数说明	范围
s1	双字	输入数据个数	指定输入的数据个数	1~32
s1+1				
s1+2	双字	第 1 个数据	输入的数据	0~16777215
s1+3				
s1+4	双字	第 2 个数据		
s1+5				
。。。	。。。	。。。		
s1+64	双字	第 32 个数据		
s1+65				

输入参数 S2				
单元	字节数	功能	参数说明	范围
s2	双字	指定选择的数据个数的最大值	指定选择的数据个数的最大值。	1~s1，由于时间限制的范围要参
s2+1				

				照注意点。
s2+2 s2+3	双字	输出模式	输出模式选择： 0、输入数组中的 0 值不加入输出组合。 1、输入数组中的 0 值加入输出组合。	

输入参数 S3				
单元	字节数	功能	参数说明	范围
s3 s3+1	双字	指定目标数据	指定选择的目标数据。	0~16777215

参数 d1				
单元	字节数	功能	参数说明	范围
d1 d1+1	双字	选择结果，每一个位表示一个数据	用位来表示，数据所在的位置 第 0 位对应 S1+2； 第 1 位对应 S1+4。	只读
d1+2 d1+3	双字	选择出的第 1 个数据所在的位置	数据所在的位置，相对 S1+2 的偏移， 0 表示数据在 S1+2； 3 表示数据在 S1+8。 以此类推。	只读
d1+4 d1+5	双字	选择出的第 2 个数据所在的位置		
d1+N*2+2 d1+N*2+3	双字	选择出的第 N 个数据所在的位置		
d1+64 d1+65	双字	选择出的第 32 个数据所在的位置		

参数 d2					备注
单元	字节数	功能	参数说明	备注	
d2 d2+1	双字	选择执行结果 OR 错误码	错误代码	≥0 执行正确 < 0，错误码	-1 输入的个数不在范围内 -2 选择的数据个数不在范围 -3 目标的数据不在范围内 -4 输入的数据不在范围 （有重复错误优先报最上面的错误，比如输入个数为 0 同时目标数据为-30. 这时就报-1 错误）
d2+2 d2+3	双字	选择出来的个数	实际选择出来的数据个数	只读	
d2+4 d2+5	双字	最小偏差	当前结果减去目标值得出的数值	只读	
d2+6 d2+7	双字	运行时间	使用的总时间（ms）		
d2+8 d2+9	双字	保留	组合数 （结果所在的）		
d2+10~ d1+73	双字	内部使用	内部计算使用缓存		

注意事项

在数据个数较大的情况下可能会出现看门狗超时的情况。这是由于计算会占用大量时间导致。

目前给出对应的该指令的时间表，请根据时间表合理使用选择的数据个数的最大值。

数组数量	1	2	3	4	5	6	7	8
选择个数	时间单位（ms）							
1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
2		<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
3			<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
4				<0.1	<0.1	<0.1	<0.1	<0.1
5					<0.1	<0.1	<0.1	<0.1
6						<0.1	<0.2	<0.2
7							<0.2	<0.2
8								<0.2

数组数量	9	10	11	12	13	14	15	16
选择个数	时间单位（ms）							
1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
2	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
3	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.2	<0.2
4	<0.1	<0.1	<0.2	<0.2	<0.2	<0.4	<0.4	<0.4
5	<0.2	<0.2	<0.3	<0.4	<0.5	<0.6	<0.9	1.1
6	<0.2	<0.3	<0.3	<0.6	<0.9	1.2	1.8	2.5
7	<0.2	<0.4	<0.6	<0.9	1.4	2.2	3.4	5.9
8	<0.3	<0.4	<0.7	1.4	2.0	3.3	5.5	8.9
9	<0.3	<0.5	<0.9	1.5	2.6	4.6	7.9	13.4
10		<0.6	<1.0	1.7	3.2	5.8	10.4	18.4
11			1.1	2.2	3.7	6.9	12.8	23.3
12				2.3	4.4	8.2	15.2	28.8
13					4.9	9.3	17.5	32.8
14						10.5	19.8	37.5
15							23.0	43.0
16								46.8

数组数量	17	18	19	20	21	22	23	24
选择个数	时间单位（ms）							
1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
2	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
3	<0.3	<0.3	<0.3	<0.3	<0.3	<0.3	<0.3	<0.3
4	<0.5	<0.6	<0.7	<0.8	<0.9	1.2	1.3	1.5
5	1.4	1.8	2.3	2.9	3.7	4.6	5.7	6.9
6	3.6	5.0	6.8	9.9	12.2	15.9	20.8	26.8
7	7.7	11.4	16.5	23.5	32.9	45.7	62.5	84.4
8	14.2	22.9	33.4	51.2	75.8	110.6	158.9	225.1
9	22.6	37.2	60.2	95.3	148.6	227.6	342.9	

10	32.2	55.4	93.8	156.9	255.2			
11	42.4	75.3	132.2	222.8				
12	51.8	94.9	171.9					
13	61.4	114.3	221.2					
14	70.7	133.2						
15	80.6	151.9						
16	89.4	170.5						
17	98.6	189.7						
18		207.7						

数组数量	25	26	27	28	29	30	31	32
选择个数	时间单位（ms）							
1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
2	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1	<0.1
3	<0.3	<0.3	<0.4	<0.4	<0.5	<0.5	<0.6	<0.6
4	1.7	1.9	2.3	2.6	3.0	3.5	3.8	4.4
5	8.4	10.2	12.3	14.7	17.5	20.6	24.2	28.3
6	34.2	43.3	54.2	15.9	83.3	102.2	124.3	150.5
7	112.6	148.6	194.2	251.4	322.6	410.5	515.4	649.8
8	314.4	433.7	591.6					
9								

*红色字为超过默认扫描周期的极限

错误码

错误码	内容
4084H	数据范围错误,具体可查看参数(d2)的错误码。
4085H	(s1)(s2)(s3)软元件地址超出范围。
4086H	(d1)(d2)软元件地址超出范围。

示例

D200 = 8, D4 = 2,表示从 8 组数据中，取出两个数据，两个数据之和最接近 D8 内的数据。

D200（S1）的数组数据：

D200	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	8
D201	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D202	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	150
D203	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D204	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	99
D205	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D206	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	56
D207	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D208	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	200
D209	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D210	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	86
D211	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D212	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	76
D213	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D214	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	160
D215	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D216	1	0	0	1	1	0	0	1	0	0	0	0	0	0	0	153
D217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

福州富昌维控电子科技有限公司

从以上数据中选出和值接近 300 的组合，D300（D1）选出的结果：

软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D300	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	10
D301	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D302	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
D303	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D304	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
D305	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D306	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D307	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D308	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D309	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D310	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D311	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D312	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

D300 的位 1 和位 3 为 1 以及目前选出 1 和 3 的数据位置。指示的位置是 D204（99）和 D208（200）。

D400（D2）运行结果：

软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D401	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D402	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
D403	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D404	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-1
D405	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-1
D406	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
D407	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D408	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	255
D409	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D410	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D411	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D412	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D413	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D414	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D415	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D416	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D417	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D418	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D419	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

D400 为 0 说明执行正确。

D402 为 2 为选择个数为 2。

D404 为-1 说明 选择出的数据组合和值减去目标值差值为-1。

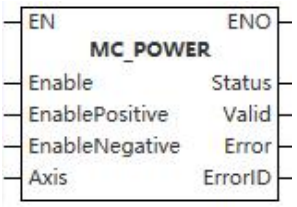
D406 为 2 说明使用的时间为 2ms。

8 运动控制

8.1 单轴运动控制的管理类功能块

1、MC_POWER

MC_POWER — 使能控制指令



该功能块是运动控制系统的上电软件开关。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_POWER	Enable	●	●	●	●	●																					
	EnablePositive	●	●	●	●	●																					
	EnableNegative	●	●	●	●	●																					
	Axis																										
	Status		●	●	●	●																					
	Valid		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_POWER	输入	Enable	BOOL	否	TRUE/FALSE	使能
		EnablePositive	BOOL	否	TRUE/FALSE	允许正方向运动
		EnableNegative	BOOL	否	TRUE/FALSE	允许负方向运动
		Axis	WORD	否	[0,65535]	轴号
	输出	Status	BOOL	能	TRUE/FALSE	使能状态
		Valid	BOOL	能	TRUE/FALSE	数据是否有效
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

- 用于设定轴的使能状态，高电平有效。
- 功能如下：

① Enable 输入为 TRUE，指令启动。驱动器的电源启用，而不是功能块本身的电源启用；

② Enable 输入为 TRUE 时，调用本功能块。如果轴的状态正好是关闭（Disable）状态，则该轴的状态转换到保持静止（StandStill）状态，等待对该轴进行运动控制；

③ Enable 输入为 TRUE 时，有故障发生。则可由一个定时器功能块和一个与逻辑函数使得其轴的状态保持在 FAlSE。它表示在硬件层级有一个故障；

福州富昌维控电子科技有限公司

④ 如果在运动过程中，PLC 供电发生故障。则轴的状态转换到故障停止 ErrorStop 状态；

⑤ 输入信号 EnablePositive 和 EnableNegative 都是电平控制的。它们分别在 TRUE 时，允许轴运动在正方向和负方向，同时可以将该两个输入信号置位，表示允许运动方向既可以为正，也可以为负；

⑥ 每一个轴只能有一个 MC_POWER 功能块用于该轴的上电。如果有多个 MC_POWER 控制同一个轴，则最后一个功能块有效；

⑦ Status 输出信号表示驱动器电源的状态，1 表示已经上电，0 表示未上电。如果轴驱动器是仿真轴，则它也不是仿真的值。

Valid 表示功能块输出信号是否有效，1 表示输出信号有效，0 表示输出信号无效。

Error 表示功能块是否有故障，1 表示有故障，0 表示没故障。ErrorID 是故障代码，当发生功能块故障时，有 16 位数字表示故障的信息。

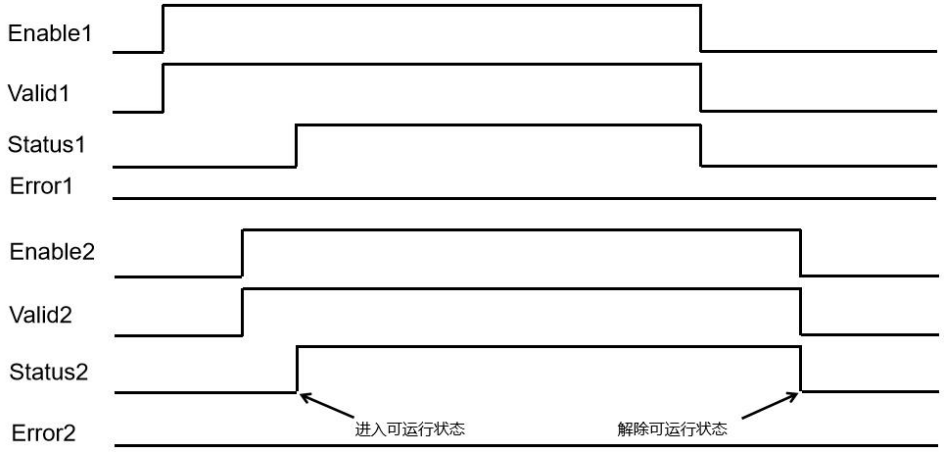
注意事项

- 当轴在使能过程中出现任何错误均会通过 Error 以及 ErrorID 反馈；
- 轴绑定的伺服轴 PDO 参数中一定要有控制字、状态字、目标位置、当前位置。

时序图

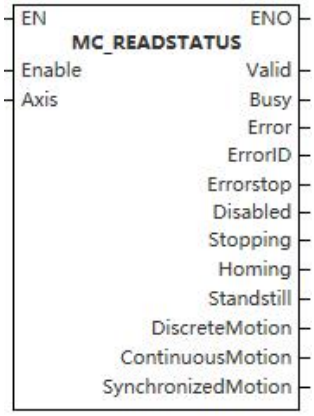
- 当使用一条 MC_Power 指令，轴正常使能时，时序图如下所示：

- 当同一轴使用两条 MC_Power 指令，轴正常使能时，时序图如下所示：



2、MC_READSTATUS

MC_READSTATUS — 读取轴状态指令



该功能块是用于读取被控轴的状态。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READ STATUS	Enable	●	●	●	●	●																				
	Axis																									
	Valid		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	Errorstop		●	●	●	●																				
	Disabled		●	●	●	●																				
	Stopping		●	●	●	●																				
	Homing		●	●	●	●																				
	Standstill		●	●	●	●																				
	DiscreteMotion		●	●	●	●																				
	ContinuousMotion		●	●	●	●																				
	SynchronizedMotion		●	●	●	●																				

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_READ STATUS	输入	Enable	BOOL	否	TRUE/FALSE	使能
		Axis	WORD	否	[0,65535]	轴号
	输出	Valid	BOOL	能	TRUE/FALSE	输出是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）
		Errorstop	BOOL	能	TRUE/FALSE	错误停止标志
		Disabled	BOOL	能	TRUE/FALSE	未激活标志

		Stopping	BOOL	能	TRUE/FALSE	停止中标志
		Homing	BOOL	能	TRUE/FALSE	回零中标志
		Standstill	BOOL	能	TRUE/FALSE	静止标志
		DiscreteMotion	BOOL	能	TRUE/FALSE	断续运动标志
		ContinuousMotion	BOOL	能	TRUE/FALSE	连续运动标志
		SynchronizedMotion	BOOL	能	TRUE/FALSE	同步运动标志

功能

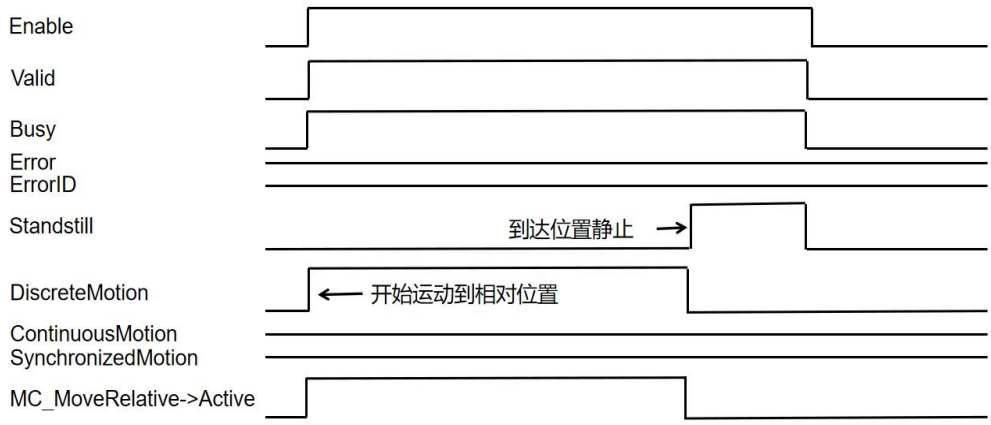
- 用于读取被控轴的状态，分别对应单轴的同步运动（SynchronizedMotion）、连续运动（ContinuousMotion）、断续(DiscreteMotion)、停止（Stopping）、故障停止（Errorstop）、回原点（Homing）、保持静止（Standstill）和关闭（Disabled）；
- 当 MC_ReadStatus 功能块输入 Enable 为 TRUE 时，周期更新轴的状态。

注意事项

- 当轴不存在时，该功能块不可用。

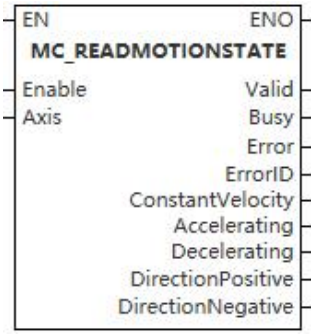
时序图

- 当 MC_READSTATUS 配合 MC_MOVERELELATIVE 指令一起使用时，轴能正常使能时，时序图如下所示：



3、MC_READMOTIONSTATE

MC_READMOTIONSTATE — 读取被控轴的运动状态



该功能块是用于读取被控轴的运动状态，包括恒速、加速、减速、正方向和负方向。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READ MOTION STATE	Enable	●	●	●	●	●																				
	Axis		●	●	●	●																				
	Valid		●	●	●	●																				
	Busy		●	●	●	●																				
	Error																									
	ErrorID																									
	ConstantVelocity		●	●	●	●																				
	Accelerating		●	●	●	●																				
	Decelerating		●	●	●	●																				
	DirectionPositive		●	●	●	●																				
	DirectionNegative		●	●	●	●																				

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_READ MOTIONSTATE	输入	Enable	BOOL	否	TRUE/FALSE	使能
		Axis	WORD	否	[0,65535]	轴号
	输出	Valid	BOOL	能	TRUE/FALSE	输出是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）
		ConstantVelocity	BOOL	能	TRUE/FALSE	恒速
		Accelerating	BOOL	能	TRUE/FALSE	加速
		Decelerating	BOOL	能	TRUE/FALSE	减速
		DirectionPositive	BOOL	能	TRUE/FALSE	正方向
		DirectionNegative	BOOL	能	TRUE/FALSE	负方向

功能

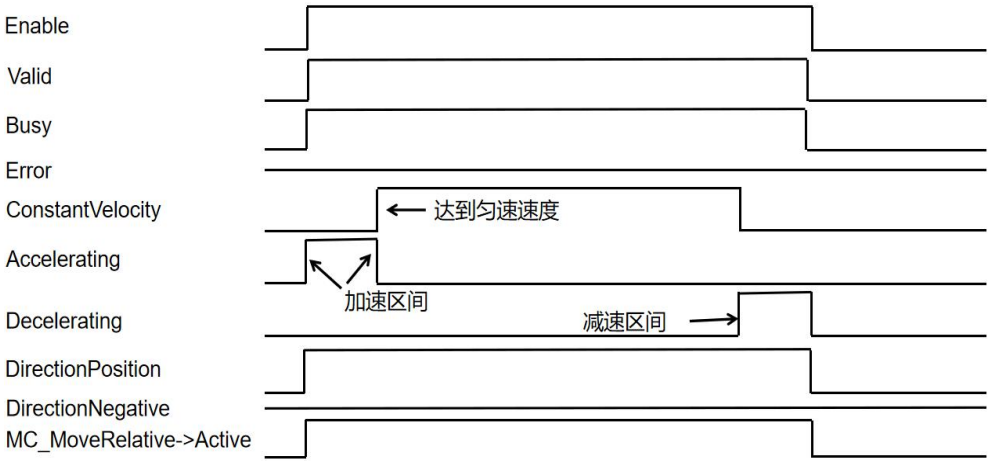
- 用于读取被控轴的状态，包括恒速、加速、减速、正方向和负方向；
- 当 MC_ReadMotionState 功能块输入 Enable 为 True 时，周期更新轴的状态。

注意事项

- 当轴不存在时，该功能块不可用。

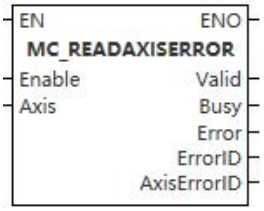
时序图

- 当 MC_READMOTIONSTATE 配合 MC_MOVERELELATIVE 指令一起使用时，轴能正常使能时，时序图如下所示：



4、MC_READAXISERROR

MC_READAXISERROR — 读取轴错误指令



该功能块是提供该轴的故障信息。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READ AXISERROR	Enable	●	●	●	●	●																					
	Axis																										
	Valid		●	●	●	●																					
	Busy		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										
	ConstantVelocity																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_READAXISERROR	输入	Enable	BOOL	否	TRUE/FALSE	使能
		Axis	WORD	否	[0,65535]	轴号
	输出	Valid	BOOL	能	TRUE/FALSE	输出是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码(请参见【8.10 错误码对照表】)
		AxisErrorID	DWORD	能	-	轴错误码

功能

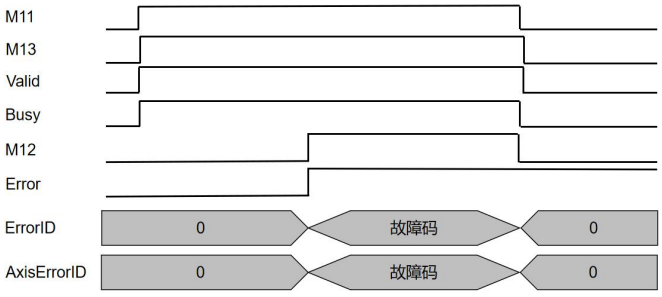
- 提供该轴的故障信息。通常可以将 MC_READSTATUS 功能块的轴状态 ErrorStop 输出连接到 MC_READAXISERROR 的 Enable 输入端，这样可以从 AxisErrorID 获得轴的故障代码。也可以从 ErrorID 获取故障代码。

注意事项

- 当轴不存在时，该功能块不可用。

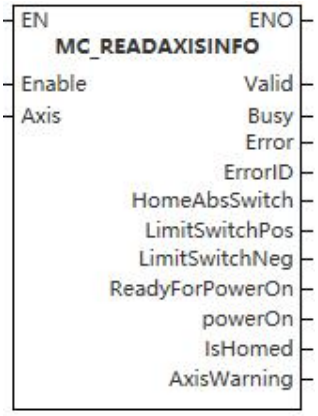
时序图

- 当 MC_READAXISINFO 配合 MC_POWER 和 MC_MOVERELELATIVE 指令一起使用时，其中 MC_POWER->EnablePosition 为 FALSE，MC_POWER->EnableNegative 为 TRUE，手动设置 MC_MOVERELELATIVE 指令 M12 为 TRUE 时，时序图如下图：



5、MC_READAXISINFO

MC_READAXISINFO — 读取被控轴的信息



该功能块是用于提供轴的信息。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READ AXISINFO	Enable	●	●	●	●	●																				
	Axis																									
	Valid		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	HomeAbsSwitch		●	●	●	●																				
	LimitSwitchPos		●	●	●	●																				
	LimitSwitchNeg		●	●	●	●																				
	ReadyForPowerOn		●	●	●	●																				
	powerOn		●	●	●	●																				
	IsHomed		●	●	●	●																				
	AxisWarning		●	●	●	●																				

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_READ AXISINFO	输入	Enable	BOOL	否	TRUE/FALSE	使能
		Axis	WORD	否	[0,65535]	轴号
	输出	Valid	BOOL	能	TRUE/FALSE	输出是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）
		HomeAbsSwitch	BOOL	能	TRUE/FALSE	原点回归绝对位置开关
		LimitSwitchPos	BOOl	能	TRUE/FALSE	正限位开关
		LimitSwitchNeg	BOOl	能	TRUE/FALSE	负限位开关
		ReadyForPowerOn	BOOl	能	TRUE/FALSE	准备好使能

福州富昌维控电子科技有限公司

		powerOn	BOOl	能	TRUE/FALSE	使能
		IsHomed	BOOl	能	TRUE/FALSE	回零成功
		AxisWarning	BOOl	能	TRUE/FALSE	轴警告

功能

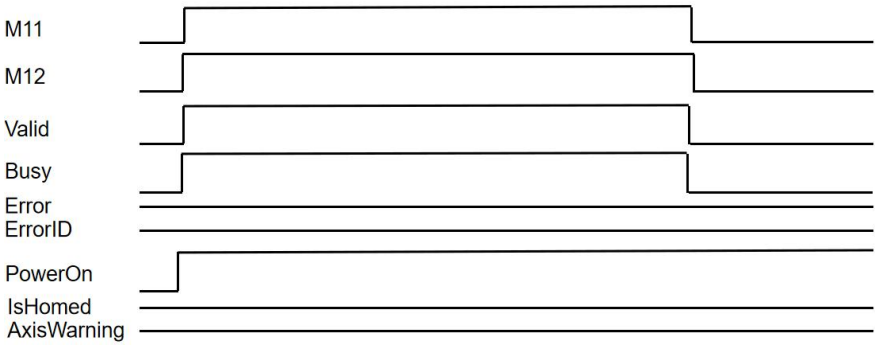
- 用于提供轴的信息，包括回原点的绝对位置开关、正和负限位开关的状态信息、功能块上电开关以及轴是否有警告等信息。通常，可用 MC_READAXISINFO 功能块输出 PowerOn 作为运动控制功能块的执行输入。例如，上电开关 MC_POWER 的 Enable 如果与 MC_MOVEABSOLUTE 功能块共用，则由于 MC_MOVEABSOLUTE 功能块是边沿触发，而 MC_POWER 功能块的执行需要一定时间，实际 MC_POWER 的输出只有在 PowerOn 置位时才表示已经上电，因此常用 MC_READAXISINFO 的输出 PowerOn 连接到 MC_MOVEABSOLUTE 功能块的 Execute 端，才能保证 MC_MOVEABSOLUTE 功能块的边沿触发。而 MC_READAXISINFO 功能块的 Enable 可与 MC_POWER 功能块共用一个输入信号。

注意事项

- 当轴不存在时，该功能块不可用。

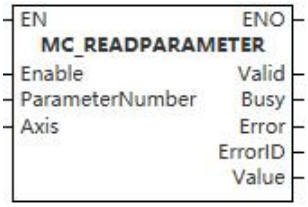
时序图

- 当 MC_READAXISINFO 配合 MC_POWER 指令一起使用时，轴能正常使能时，时序图如下图：



6、MC_READPARAMETER

MC_READPARAMETER — 读取轴参数



该功能块是用于读取轴参数。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READ PARAMETER	Enable	●	●	●	●	●																				
	ParameterNumber																									
	Axis																									
	Valid		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	Value																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_READ PARAMETER	输入	Enable	BOOL	否	TRUE/FALSE	使能
		ParameterNumber	INT	否	[1,17]	参数编号
		Axis	WORD	否	[0,65535]	轴号
	输出	Valid	BOOL	能	TRUE/FALSE	输出是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）
		Value	LREAL	能	-	轴参数值

功能

- 用于读取轴参数，如下表。

参数编号 PN	参数名	数据类型	B/E	读 R/写 W	注释
1	CommandedPosition/指令位置	REAL	B	R	命令位置
2	SWLimitPos/正方向开关限位位置	REAL	E	R/W	正方向软件限位开关的位置
3	SWLimitNeg/反方向开关限位位置	REAL	E	R/W	反方向软件限位开关的位置
4	EnableLimitPos/允许正方向开关限位	BOOL	E	R/W	允许正方向软件限位开关
5	EnableLimitNeg/允许反方向开关限位	BOOL	E	R/W	允许反方向软件限位开关
6	EnablePosLagMonitoring/允许位置偏差监视	BOOL	E	R/W	允许监视位置偏差
7	MaxPositionLag/最大位置偏差	REAL	E	R/W	最大位置偏差
8	MaxVelocitySystem/最大系统速度	REAL	E	R	运动系统中允许的轴最大速度
9	MaxVelocityAppl/最大应用速度	REAL	B	R/W	在应用中允许的轴最大速度

福州富昌维控电子科技有限公司

10	ActualVelocity/实际速度	REAL	B	R	实际速度
11	CommandedVelocitv/所命令的速度	REAL	B	R	所命令的速度
12	MaxAccelerationSystem/最大系统加速度	REAL	E	R	运动系统中允许的轴最大加速度
13	MaxAccelerationAppl/最大应用加速度	REAL	E	R/W	在应用中允许的轴最大加速度
14	MaxDecelerationSvstem/最大系统减速度	REAL	E	R	运动系统中允许的最大轴减速度
15	MaxDecelerationAppl/最大应用减速度	REAL	E	R/W	在应用中允许的最大轴减速度
16	MaxJerkSystem/最大系统加加速度	REAL	E	R	运动系统允许的轴最大加加速度
17	MaxJerkAppl/最大应用加加速度	REAL	E	R/W	应用系统允许的轴最大加加速度

注意事项

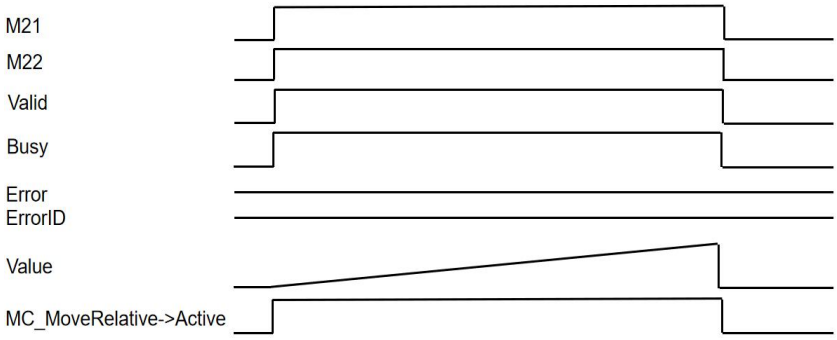
- 当轴不存在时，该功能块不可用。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

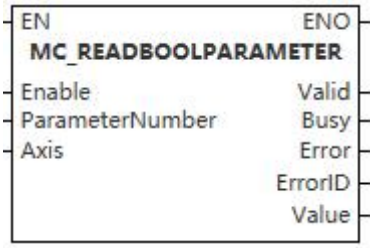
时序图

- 当 MC_READPARAMETER 配合 MC_MOVERELELATIVE 指令一起使用时，其中 ParameterNumber=1，轴能正常使能时，时序图如下所示：



7、MC_READBOOLPARAMETER

MC_READBOOLPARAMETER — 读取轴布尔参数



该功能块是用于读取轴布尔参数。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READBOOL PARAMETER	Enable	●	●	●	●	●																					
	ParameterNumber															●	●	●	●	●							
	Axis																										
	Valid		●	●	●	●																					
	Busy		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										
	Value		●	●	●	●																					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_READBOOL PARAMETER	输入	Enable	BOOL	否	TRUE/FALSE	使能
		ParameterNumber	INT	否	[1,17]	参数编号
		Axis	WORD	否	[0,65535]	轴号
	输出	Valid	BOOL	能	TRUE/FALSE	输出是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）
		Value	BOOL	能	TRUE/FALSE	轴参数

功能

- 用于读取轴布尔参数，如下表：

参数编号 PN	参数名	数据类型	B/E	读 R/写 W	注释
1	EnableLimitPos/允许正方向开关限位	BOOL	E	R/W	允许正方向软件限位开关
2	EnableLimitNeg/允许反方向开关限位	BOOL	E	R/W	允许反方向软件限位开关
3	EnableLimitPosLagMonitoring/允许位置偏差监视	BOOL	E	R/W	允许监视位置偏差

注意事项

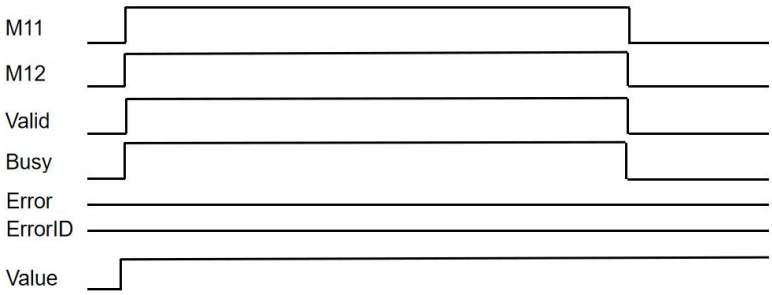
- 当轴不存在时，该功能块不可用。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

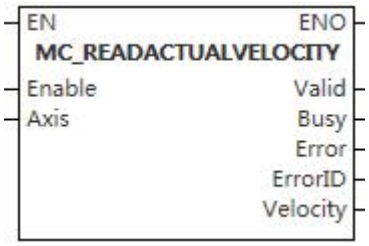
时序图

- 当 MC_READBOOLPARAMETER 配合 MC_MOVERELELATIVE 指令一起使用时，其中 ParameterNumber=4，轴能正常使能时，时序图如下所示：



8、MC_READACTUALVELOCITY

MC_READACTUALVELOCITY — 读取实际速度指令



该功能块是用于输出被控单轴的实际速度。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READACTUALVELOCITY	Enable	●	●	●	●	●																				
	Axis																									
	Valid		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	Velocity																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_READACTUALVELOCITY	输入	Enable	BOOL	否	TRUE/FALSE	使能
		Axis	WORD	否	[0,65535]	轴号
	输出	Valid	BOOL	能	TRUE/FALSE	输出是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）
		Velocity	LREAL	能	-	实际速度

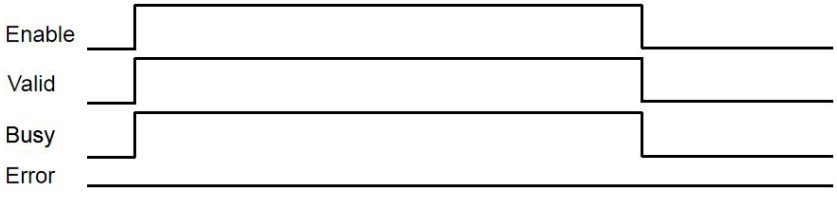
功能

- 当功能块的 Enable 置位，则 MC_READACTUALVELOCITY 输出返回被控单轴的实际速度。

注意事项

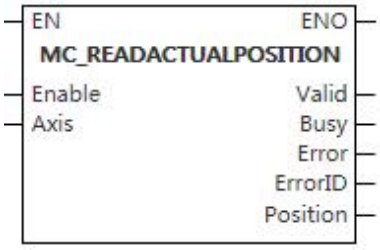
- 当轴不存在时，该功能块不可用；
- 当功能块的 Enable 复位时，输出的值丢失其有效性，Valid 输出复位。

时序图



9、MC_READACTUALPOSITION

MC_READACTUALPOSITION — 读取实际位置指令



该功能块是用于读取轴实际绝对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READ ACTUALPOSITION	Enable	●	●	●	●	●																				
	Axis																									
	Valid		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	Position																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_READ ACTUALPOSITION	输入	Enable	BOOL	否	TRUE/FALSE	使能
		Axis	WORD	否	[0,65535]	轴号
	输出	Valid	BOOL	能	TRUE/FALSE	输出是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）
		Position	LREAL	能	-	实际绝对位置

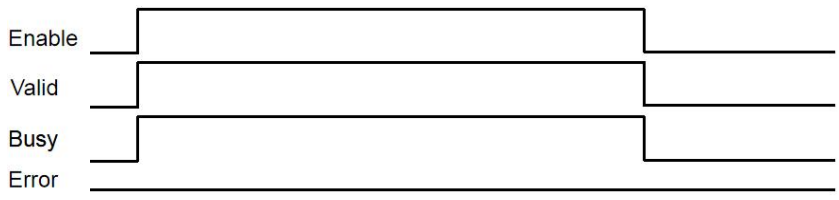
功能

- 当功能块的 Enable 置位，则 MC_READACTUALPOSITION 输出返回被控单轴的实际绝对位置。

注意事项

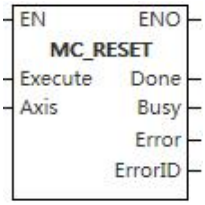
- 当轴不存在时，该功能块不可用；
- 总线轴中需配置 0x6064，否则无效；
- 当功能块的 Enable 复位时，输出的值丢失其有效性，Valid 输出复位。

时序图



10、MC_RESET

MC_RESET — 复位故障指令



该功能块是用于对被控制轴的故障进行清除处理。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_RESET	Execute	●	●	●	●	●																				
	Axis																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_RESET	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

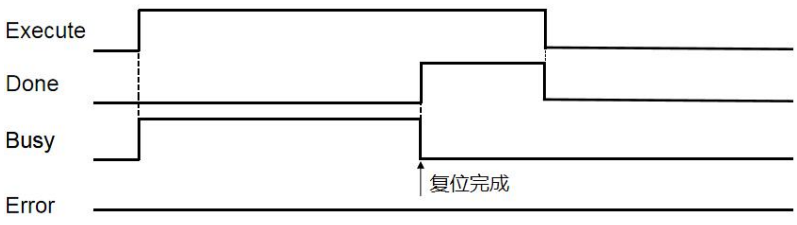
- 当 Execute 输入信号上升沿时，用于对被控制轴的故障进行清除处理。这些故障包括轴故障和驱动器侧故障。对驱动器本身故障，则应先行进行驱动器故障的复位处理。这些故障处理不引起被控轴的运动；
- 在 Execute 信号有效的上升沿，指令尝试复位轴的故障，如果复位成功，则 Done 输出有效，否则 Error 信号有效，ErrorID 将给出复位失败的原因；
- 复位成功后，如果驱动器处于使能状态，则轴的 PLCOpen 状态机进入 StandStill 状态，如果驱动器没有使能，则进入 Disabled 状态。

注意事项

- 当轴不存在时，该功能块不可用。

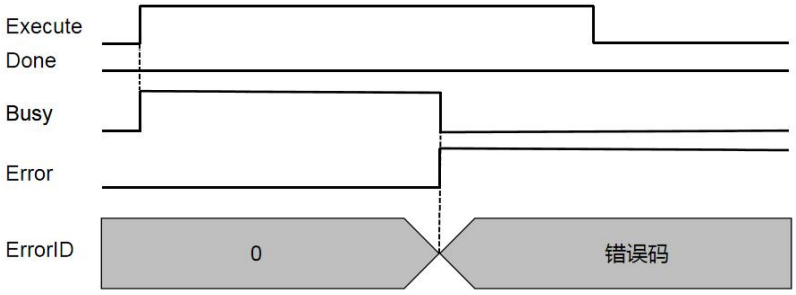
时序图

- 在轴发生故障时调用 MC_Reset 指令且成功复位轴的故障，时序图如下所示：



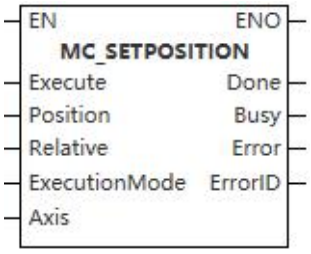
福州富昌维控电子科技有限公司

- 当驱动器发生不可复位的故障时，时序图如下所示：



11、MC_SETPOSITION

MC_SETPOSITION — 设定轴位置



该功能块是用于设置被控制轴的位置（绝对位置或者相对位置）。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_SET POSITION	Execute	●	●	●	●	●																				
	Position																									
	Relative	●	●	●	●	●																				
	ExecutionMode																									
	Axis																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_SET POSITION	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Position	LREAL	否	[-999999999.999999, 999999999.999999]	位置
		Relative	BOOL	否	TRUE/FALSE	设置相对位置或者绝对位置
		ExecutionMode	WORD	否	[0,2]	执行模式
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	否	TRUE/FALSE	完成标志
		Busy	BOOL	否	TRUE/FALSE	忙标志
		Error	BOOL	否	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	否	-	错误码（请参见【8.10 错误码对照表】）

功能

- MC_SETPOSITION 功能块用于设置被控制轴的位置（绝对位置或者相对位置）。它通过移动坐标系的坐标系统来控制轴的设定位置和实际位置，使它们有相同的数值，而不引起运动的发生，通过重新校准以获得相同的跟踪误差。本功能块可用于进行参照情况下的实例；
- 目前该指令暂时不支持轴处于运动状态下触发。

注意事项

- 当轴不存在时，该功能块不可用。

8.2 单轴运动控制的运动类功能块

1、MC_STOP

MC_STOP — 停止指令

该功能块使被控轴的状态转换到减速停止 Stopping 状态。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_STOP	Execute	●	●	●	●	●																				
	Deceleration																									
	Jerk																									
	Axis																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_STOP	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Deceleration	LREAL	否	(0,999999999.999999]	减速度
		Jerk	LREAL	能	[0,999999999.999999]	加加速度，仅在 S 型曲线生效
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		CommandAborted	BOOL	能	TRUE/FALSE	被中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块使被控轴的状态转换到减速停止 Stopping 状态。即当输入 Execute 信号上升沿触发功能块，使被控轴以输出的 Deceleration 值减速，并停止轴运动；
- 减速完成后，Done 信号有效且在 Execute=ON 期间一直保持在 Stopping 状态；
- Execute=OFF 且 Done=ON 时，轴由 Stopping 状态切换到 Standstill 状态。

注意事项

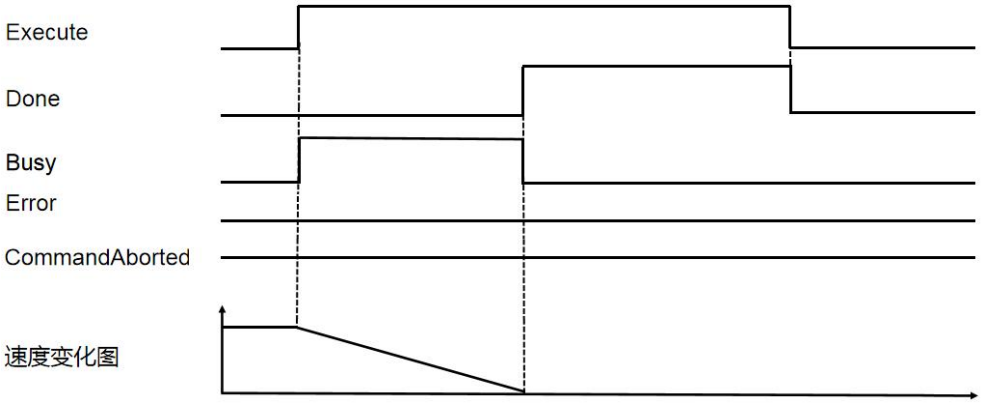
- 当轴不存在时，该功能块不可用。

错误码

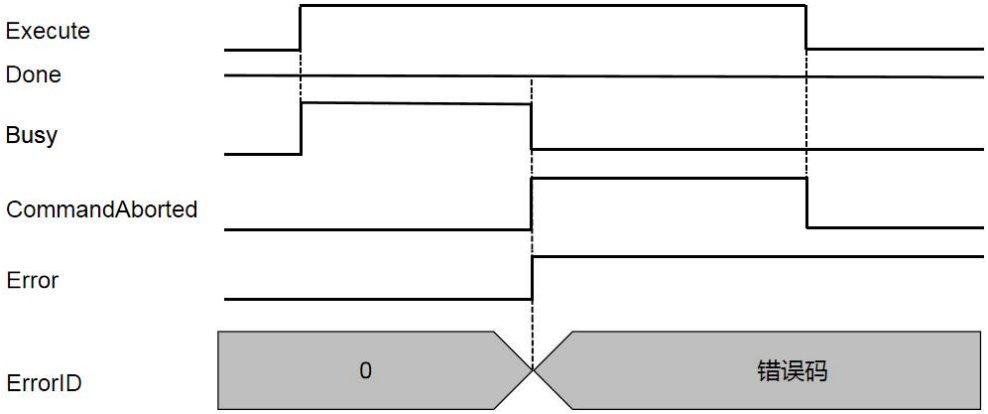
错误码	内容
4084H	输入了超出可指定范围的数据。

时序图

- 调用定位之后，触发 MC_STOP 指令，时序图如下所示：



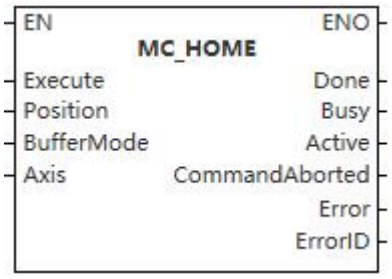
- 指令运行期间驱动器发生故障，时序图如下所示：



福州富昌维控电子科技有限公司

2、MC_HOME

MC_HOME



该功能块使被控轴的状态回到原点状态。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_HMOE	Execute	●	●	●	●	●																				
	Position																									
	Axis																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_HOME	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Position	LREAL	否	[-999999999.999999，999999999.999999]]	位置
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志位
		CommandAborted	BOOL	能	TRUE/FALSE	被中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码(请参见【8.10 错误码对照表】)

功能

- 通过检测回原点的有关限位开关、绝对位置原点开关或发送的序列脉冲信号等，确定被控轴是否回到其原点。

注意事项

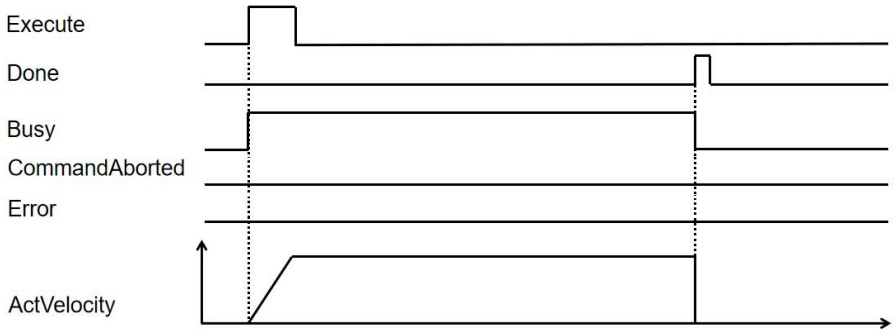
- 当轴不存在时，该功能块不可用；

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

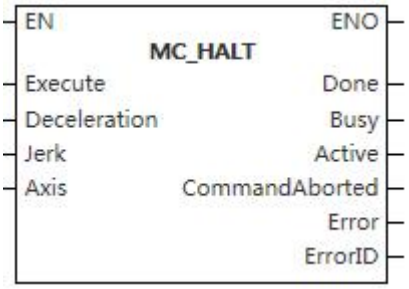
时序图

- 启动原点回归，驱动器正常执行原点回归动作，时序图如下所示：



3、MC_HALT

MC_HALT — 暂停指令



该功能块常用于正常操作条件下停止被控轴的运动。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_HALT	Execute	●	●	●	●	●																					
	Deceleration																										
	Jerk																										
	Axis																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_HALT	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Deceleration	LREAL	否	(0,999999999.999999]	减速度
		Jerk	LREAL	能	[0,999999999.999999]	加加速度，仅在 S 型曲线生效
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	数据是否有效
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志
		CommandAborted	BOOL	能	TRUE/FALSE	被中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

- 如果被控轴在连续运动状态，则 MC_Halt 功能块输入 Execute 信号上升沿触发功能块激活，使被控轴的状态转换到断续运动状态。同样，如果被控轴在断续运动状态或者同步运动状态，MC_HALT 功能块激活，都会使被控轴保持在断续运动状态。因此该功能块常用于正常操作条件下停止被控轴的运动。

注意事项

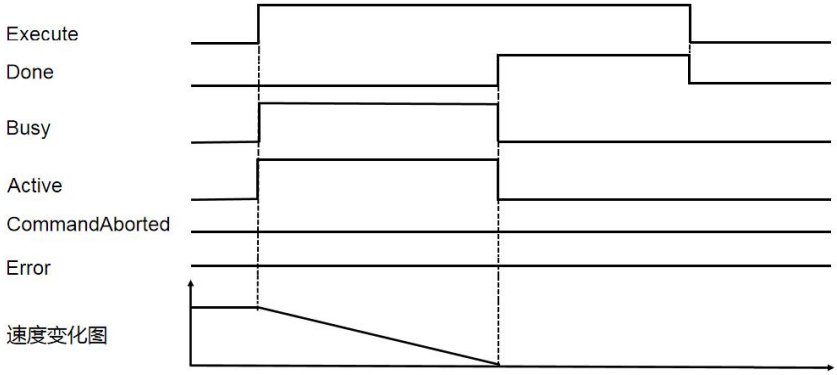
- 当轴不存在时，该功能块不可用。

错误码

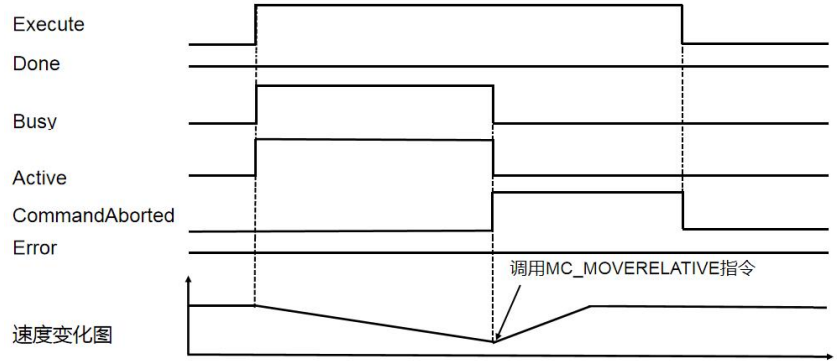
错误码	内容
4084H	输入了超出可指定范围的数据。

时序图

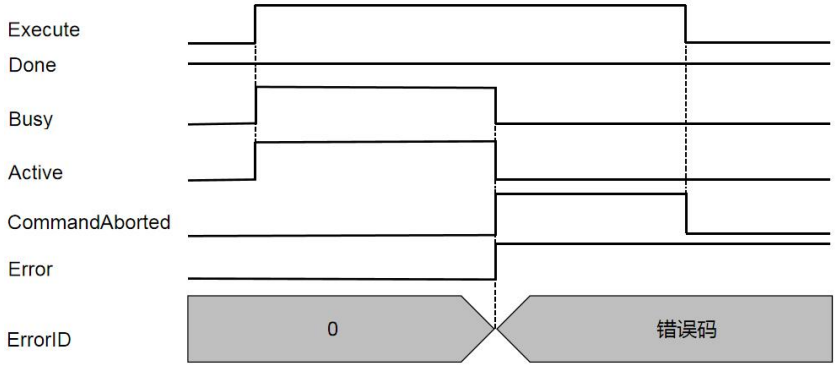
- 调用定位之后，触发 MC_HALT 指令，时序图如下所示：



- 在触发 MC_HALT 指令后，重新调用定位指令打断 MC_HALT 指令的执行，时序图如下所示：



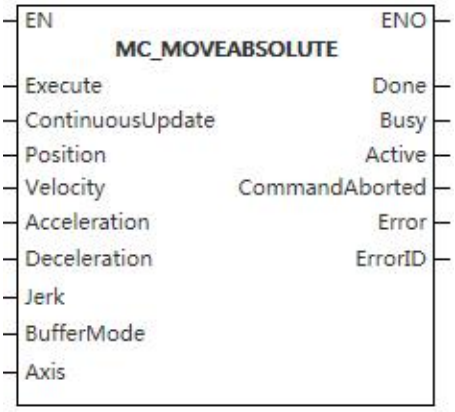
- 在 MC_HALT 指令执行期间，发生故障停机，时序图如下所示：



福州富昌维控电子科技有限公司

4、MC_MOVEABSOLUTE

MC_MOVEABSOLUTE — 绝对定位指令



该功能块使被控轴以输入的参数达到设定的绝对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVE ABSOLUTE	Execute	●	●	●	●	●																				
	ContinuousUpdate	●	●	●	●	●																				
	Position																									
	Velocity																									
	Acceleration																									
	Deceleration																									
	Jerk																									
	BufferMode																									
	Axis																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_MOVEABSOLUTE	输入	Execute	BOOL	否	TRUE/FALSE	使能
		ContinuousUpdate	BOOL	能	TRUE/FALSE	运行时更新
		Position	LREAL	否	[-999999999.999999, 999999999.999999]	绝对位置
		Velocity	LREAL	否	(0, 999999999.999999]	速度
		Acceleration	LREAL	否	(0, 999999999.999999]	加速度
		Deceleration	LREAL	否	(0, 999999999.999999]	减速度
		Jerk	LREAL	能	[0, 999999999.999999]	加加速度，仅在 S 型曲线生效

福州富昌维控电子科技有限公司

		BufferMode	WORD	能	[0,5]	模式选择。 0：中止模式； 1：已缓冲模式； 2：混成低速模式； 3：混成前模块的速度模式； 4：混成后模块的速度模式； 5：混成高速模式。
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块使被控轴以输入的参数（速度、加速度、减速度、加加速度和规定的运动方向）达到设定的绝对位置。

注意事项

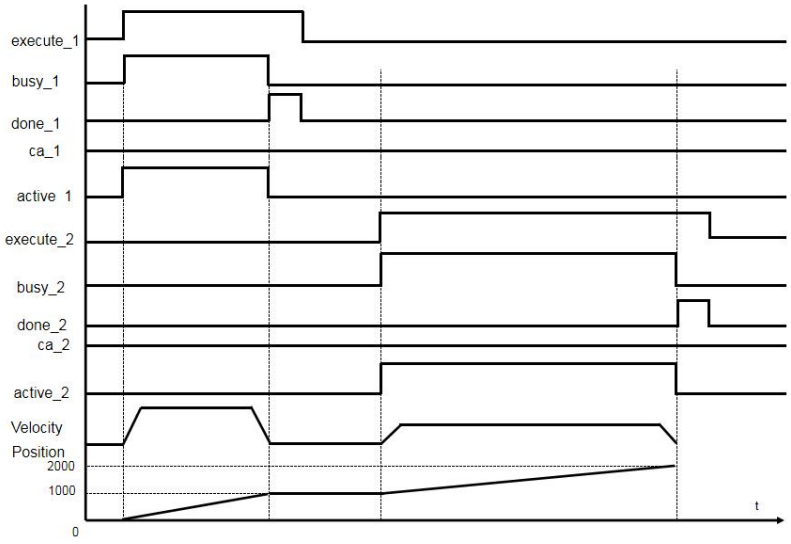
- 当轴不存在时，该功能块不可用。

错误码

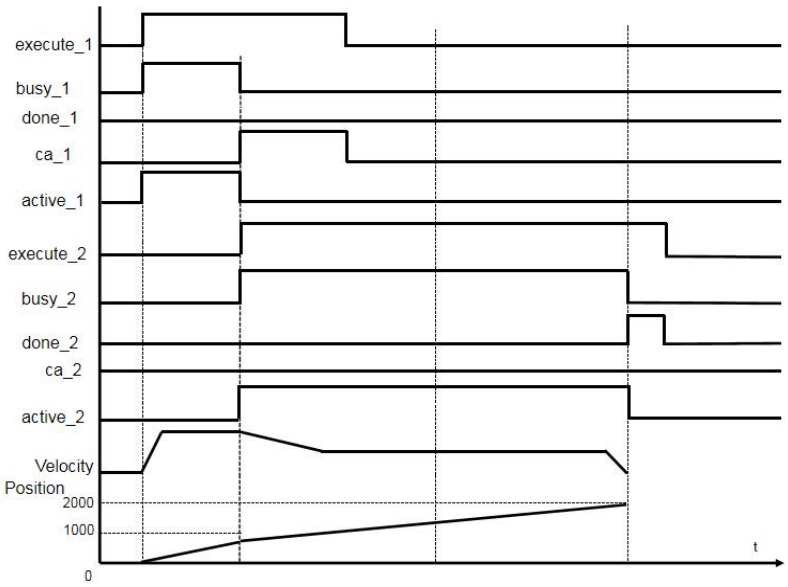
错误码	内容
4084H	输入了超出可指定范围的数据。

时序图

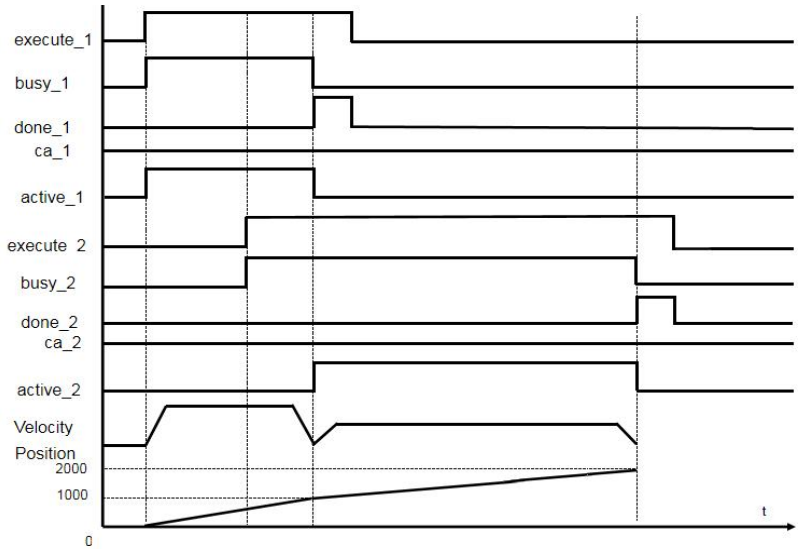
- 正常工况时序图（buffermode_1=0，buffermode_2=0）：



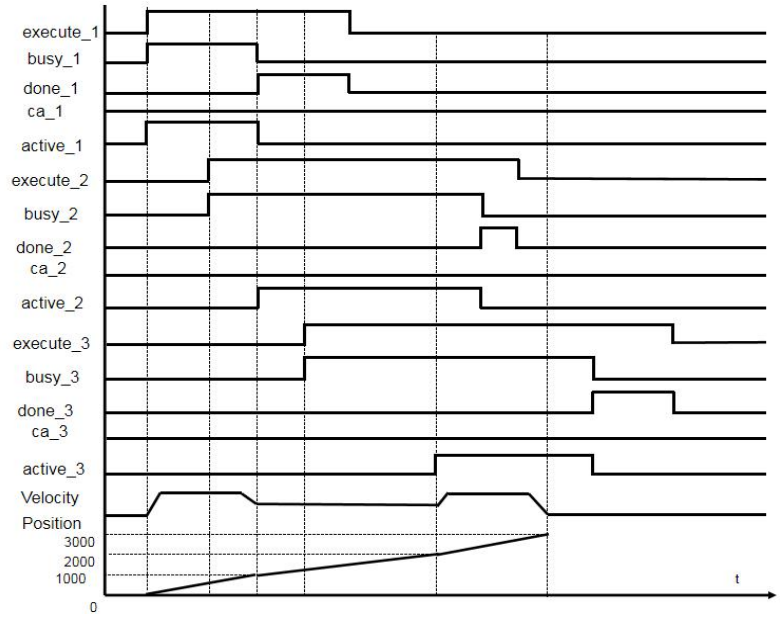
- 中止模式时序图（buffermode_1=0，buffermode_2=0）：



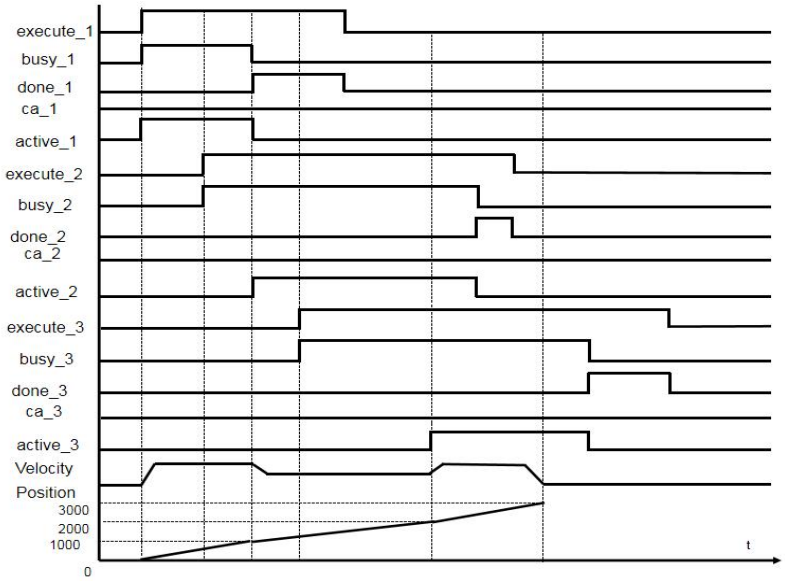
- 已缓冲模式时序图（buffermode_1=0，buffermode_2=1）：



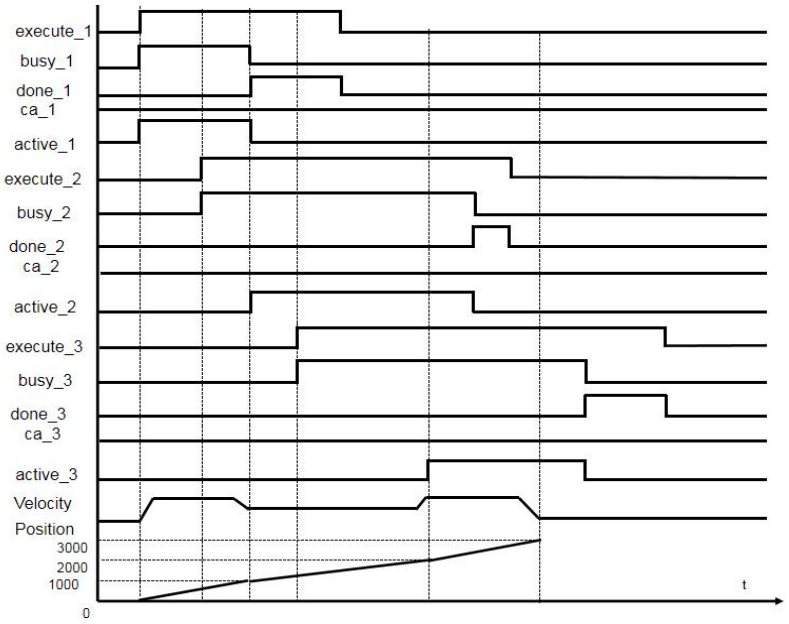
- 混成低速模式时序图（buffermode_1=0，buffermode_2=2，buffermode_3=2）：



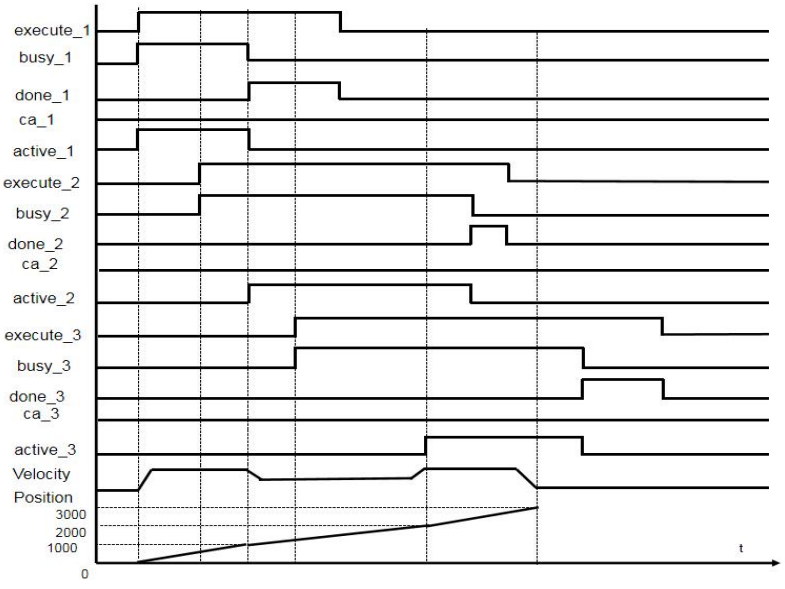
- 混成前模块速度模式时序图（buffermode_1=0，buffermode_2=3，buffermode_3=3）：



- 混成后模块速度模式时序图（buffermode_1=0，buffermode_2=4，buffermode_3=4）：

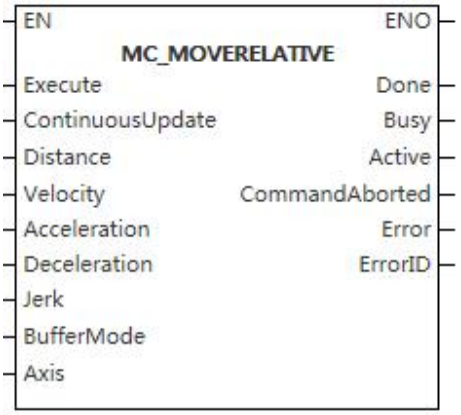


- 混成高速模式时序图（buffermode_1=0，buffermode_2=5，buffermode_3=5）：



5、MC_MOVERELATIVE

MC_MOVERELATIVE — 相对定位指令



该功能块使被控轴以输入的参数达到设定的相对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVE RELATIV	Execute	●	●	●	●	●																				
	ContinuousUpdate	●	●	●	●	●																				
	Distance																									
	Velocity																									
	Acceleration																									
	Deceleration																									
	Jerk																									
	BufferMode																									
	Axis																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_MOVERELATIVE	输入	Execute	BOOL	否	TRUE/FALSE	使能
		ContinuousUpdate	BOOL	能	TRUE/FALSE	运行时更新
		Distance	LREAL	否	[-999999999.999999, 999999999.999999]	相对位置
		Velocity	LREAL	否	(0,999999999.999999]	速度
		Acceleration	LREAL	否	(0,999999999.999999]	加速度
		Deceleration	LREAL	否	(0,999999999.999999]	减速度
		Jerk	LREAL	能	[0,999999999.999999]	加加速度，仅在 S 型曲线生效

福州富昌维控电子科技有限公司

		BufferMode	WORD	能	[0,5]	模式选择。 0：中止模式； 1：已缓冲模式； 2：混成低速模式； 3：混成前模块的速度模式； 4：混成后模块的速度模式； 5：混成高速模式。
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码(请参见【8.10 错误码对照表】)

功能

- 该功能块以当前位置为运动起点，以设定的参数（速度、加速度、减速度和加加速度）控制被控轴达到设定的相对位置 Distance。即以运动起点加 Distance 作为终点的运动控制。

注意事项

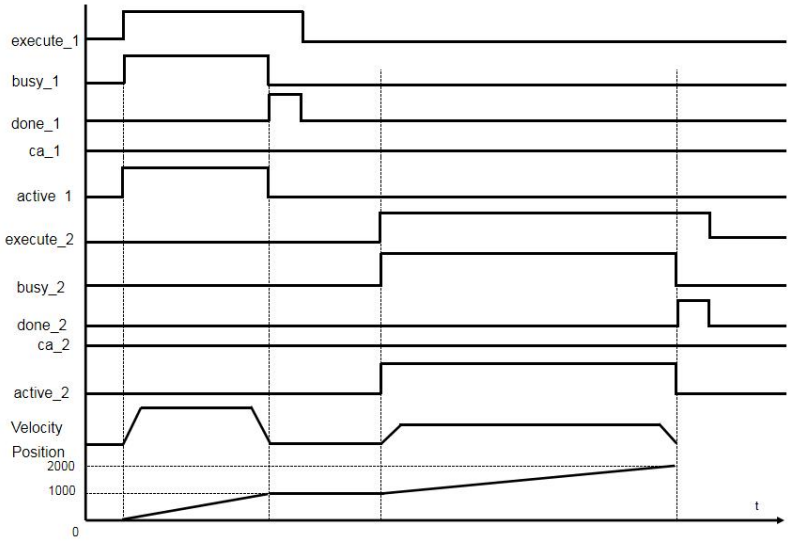
- 当轴不存在时，该功能块不可用。

错误码

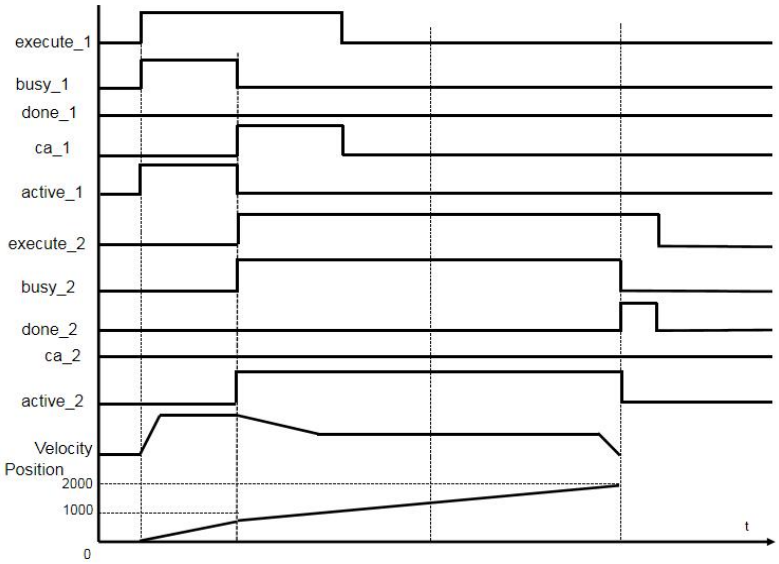
错误码	内容
4084H	输入了超出可指定范围的数据。

时序图

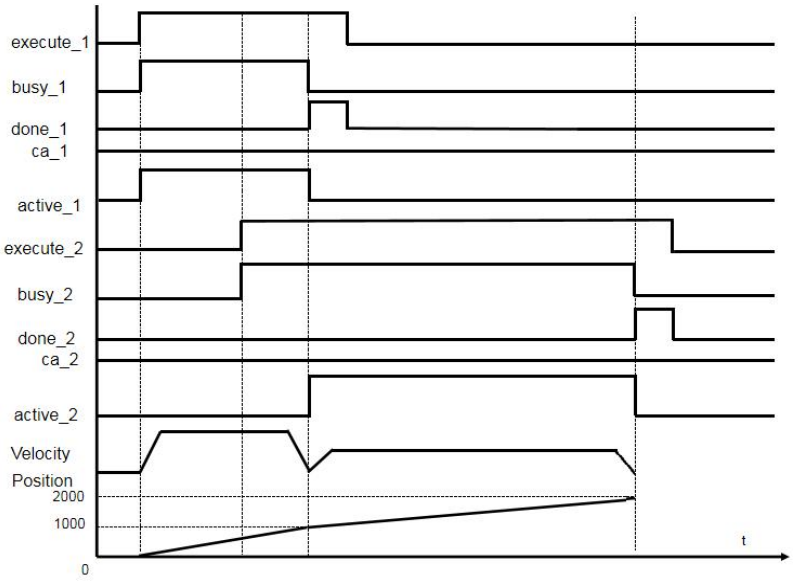
- 正常工况时序图（buffermode_1=0，buffermode_2=0）：



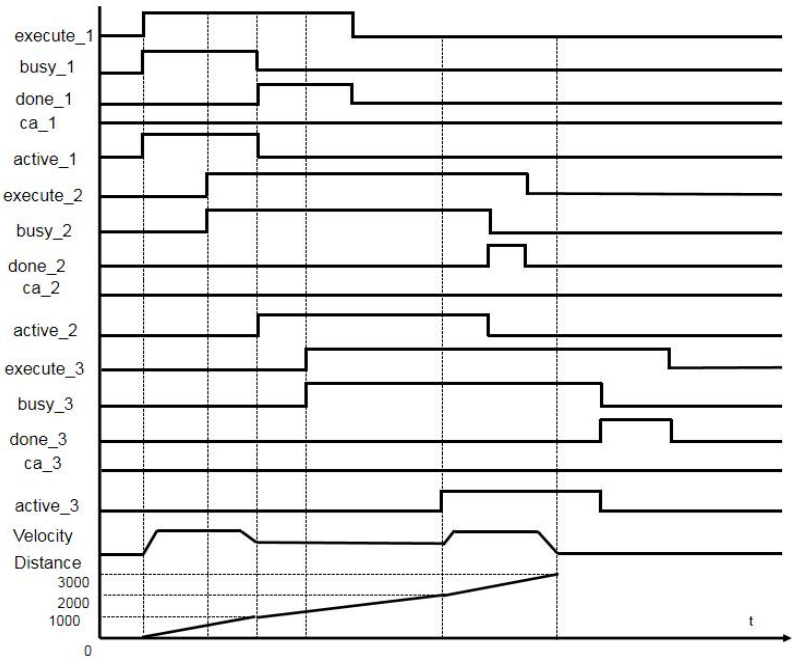
- 中止模式时序图（buffermode_1=0，buffermode_2=0）：



- 已缓冲模式时序图（buffermode_1=0，buffermode_2=1）：

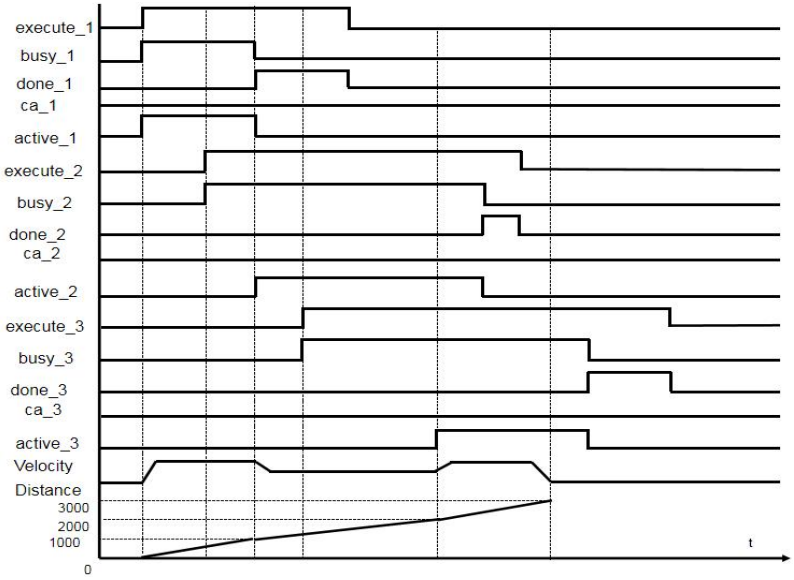


- 混成低速模式时序图（buffermode_1=0，buffermode_2=2，buffermode_3=2）：

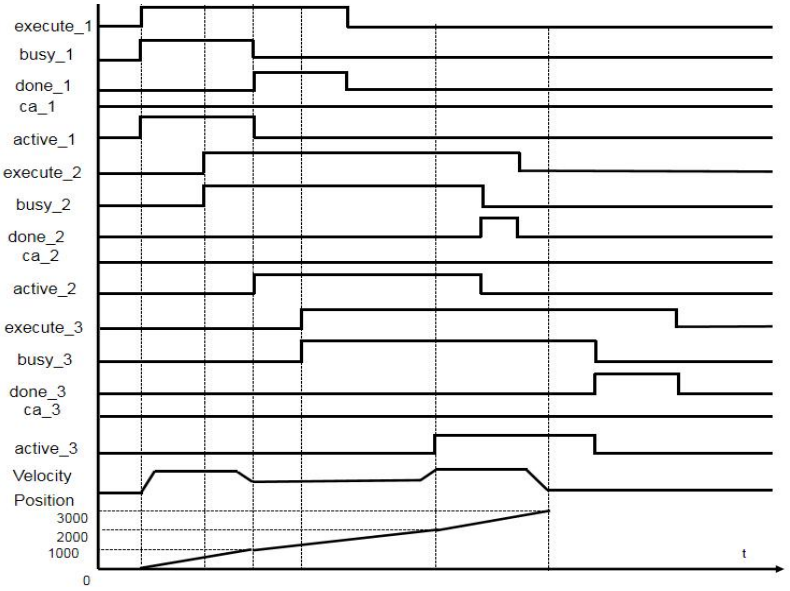


福州富昌维控电子科技有限公司

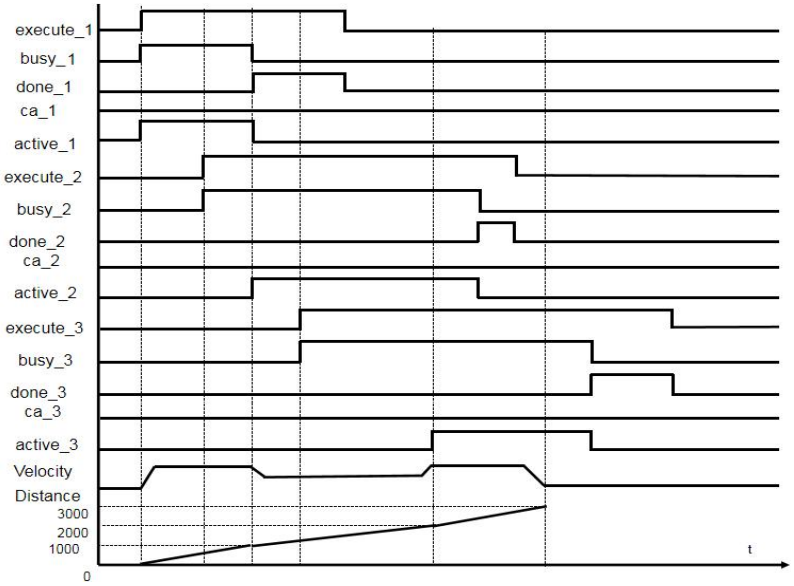
- 混成前模块速度模式时序图（buffermode_1=0，buffermode_2=3，buffermode_3=3）：



- 混成后模块速度模式时序图（buffermode_1=0，buffermode_2=4，buffermode_3=4）：

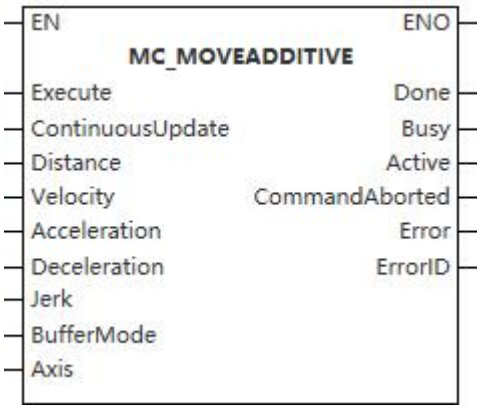


- 混成高速模式时序图（buffermode_1=0，buffermode_2=5，buffermode_3=5）：



6、MC_MOVEADDITIVE

MC_MOVEADDITIVE — 位移叠加指令



该功能块是运动叠加功能块。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVEADDITIVE	Execute	●	●	●	●	●																				
	ContinuousUpdate	●	●	●	●	●																				
	Distance																									
	Velocity																									
	Acceleration																									
	Deceleration																									
	jerk																									
	BufferMode																									
	Axis																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_MOVEADDITIVE	输入	Execute	BOOL	否	TRUE/FALSE	使能
		ContinuousUpdate	BOOL	否	TRUE/FALSE	运行时更新
		Distance	LREAL	否	[-9999999.00,9999999.00]	绝对位置
		Velocity	LREAL	否	[0,9999999.00]	速度
		Acceleration	LREAL	否	[0,9999999.00]	加速度
		Deceleration	LREAL	能	[0,9999999.00]	减速度
		Jerk	LREAL	否	[0,9999999.00]	加加速度

		BufferMode	WORD	能	[0,5]	模式选择。 0：中止模式； 1：已缓冲模式； 2：混成低速模式； 3：混成前模块的速度模式； 4：混成后模块的速度模式； 5：混成高速模式。
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块是在原有持续运动位移位置上增加 Distance 位移、原有的速度重置为 Velocity 速度。

注意事项

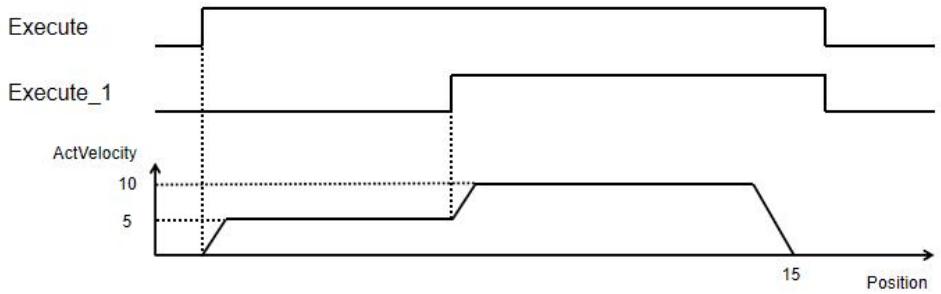
- 当轴不存在时该功能块不可用。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

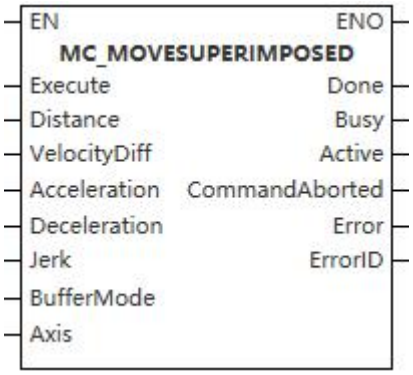
时序图

- 当 MC_MOVEABSOLUTE 和 MC_MOVEADDITIVE 配合正常使用时，当 MC_MOVEABSOLUTE 的目标位置是 10、速度是 5，MC_MOVEADDITIVED 的叠加位置是 15，速度是 10 时，Axis_0 轴的位置速度关系如下：



7、MC_MOVESUPERIMPOSED

MC_MOVESUPERIMPOSED — 速度和位移叠加指令



该功能块是运动叠加功能块。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVESUPERIMPOSED	Execute	●	●	●	●	●																				
	Distance																									
	VelocityDiff																									
	Acceleration																									
	Deceleration																									
	jerk																									
	BufferMode																									
	Axis																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_MOVESUPERIMPOSED	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Distance	LREAL	否	[-999999999.999999, 999999999.999999]	绝对位置
		VelocityDiff	LREAL	否	(0,999999999.999999]	速度
		Acceleration	LREAL	否	(0,999999999.999999]	加速度
		Deceleration	LREAL	能	(0,999999999.999999]	减速度
		Jerk	LREAL	否	[0,999999999.999999]	加加速度，仅在 S 型曲线生效
		BufferMode	WORD	能	[0,5]	模式选择。 0：中止模式； 1：已缓冲模式；

福州富昌维控电子科技有限公司

						2：混成低速模式； 3：混成前模块的速度模式； 4：混成后模块的速度模式； 5：混成高速模式。
		Axis	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块是在原有持续运动位移位置上增加 Distance 位移、原有的速度叠加 VelocityDiff 速度。

注意事项

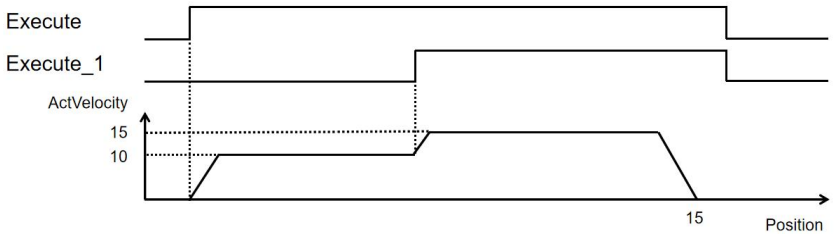
- 当轴不存在时，该功能块不可用。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

时序图

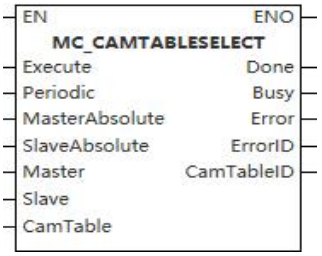
- 当 MC_MOVEABSOLUTE 和 MC_MOVESUPERIMPOSED 配合正常使用时，当 MC_MOVEABSOLUTE 的目标位置是 10、速度是 5，MC_MOVESUPERIMPOSED 的叠加位置是 5、速度是 10 时，Axis_0 轴的位置速度关系如下：



8.3 凸轮和齿轮多轴运动控制的管理类功能块

1、MC_CAMTABLESELECT

MC_CAMTABLESELECT — 凸轮表绑定



该功能块用于绑定凸轮表。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_CAMTABLESELECT	Execute	●	●	●	●	●																				
	Periodic	●	●	●	●	●																				
	MasterAbsolute	●	●	●	●	●																				
	SlaveAbsolute	●	●	●	●	●																				
	Master																									
	Slave																									
	CamTable																	●								
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	CamTableID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_CAMTABLESELECT	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Periodic	BOOL	否	TRUE/FALSE	周期模式设置
		MasterAbsolute	BOOL	能	TRUE/FALSE	主轴位置模式设置
		SlaveAbsolute	BOOL	能	TRUE/FALSE	从轴位置模式设置
		Master	WORD	否	[0,65535]	主轴
		Slave	WORD	否	[0,65535]	从轴
		CamTable	WORD	否	[0,65535]	凸轮表 ID
	输出	Done	BOOL	否	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】
		CamTableID	WORD	能	-	凸轮表索引号

功能

- 该功能块用于绑定凸轮表，即将输入的主轴、从轴和凸轮表绑定。在调用 MC_CAMIN 功能块时，只能使用经过该功能块绑定的凸轮表；
- 主轴采用相对/绝对模式，影响的是在 MC_CAMIN 指令触发时，内部查表的初始位置，在 MC_CAMTABLESELECT 触发时，仅仅给定该凸轮表的属性。主轴最终采用何种模式只由 MasterAbsolute 决定，不受 MC_CAMIN 指令中的 StartMode 影响。需要注意，主轴绝对模式下有可能造成从轴位置的跃迁；
- 从轴采用相对/绝对模式，影响的是在 MC_CAMIN 指令触发时，内部锁存的初始位置，在 MC_CAMTBLSEL 触发时，仅仅给定该凸轮表的属性。此外，从轴最终采用何种模式会受到 MC_CAMIN 中 StartMode 属性影响具体见 MC_CAMIN 指令的介绍。需要注意，从轴绝对模式下有可能造成从轴位置的跃迁。

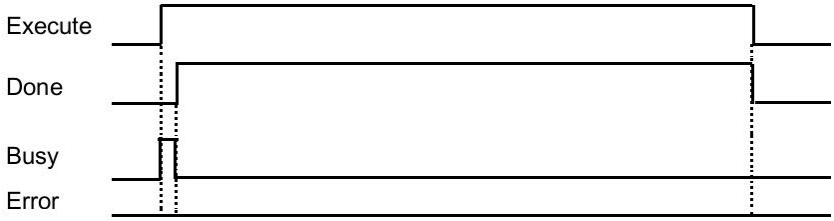
注意事项

- 当轴不存在时，该功能块不可用。

错误码

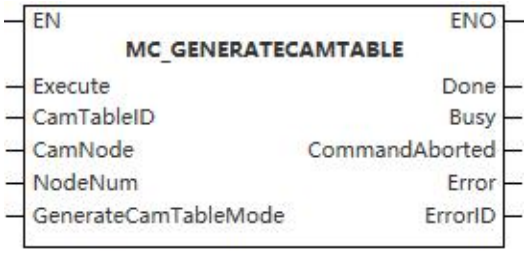
错误码	内容
4084H	输入了超出可指定范围的数据。

时序图



2、MC_GENERATECAMTABLE

MC_GENERATECAMTABLE — 更新凸轮表



该功能块用于更新凸轮表。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GENERATECAMTABLE	Execute	●	●	●	●	●																				
	CamTableID																									
	CamNode																									
	NodeNum																	●								
	GenerateCamTableMode																	●								
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_GENERATECAMTABLE	输入	Execute	BOOL	否	TRUE/FALSE	使能
		CamTableID	WORD	否	[0,65535]	凸轮表索引号
		CamNode	ARRAY (CAM_NODE_T)	否	-	凸轮节点数组
		NodeNum	INT	否	[2,361]	凸轮节点数量
		GenerateCamTableMode	INT	否	0	生效模式。 0：在下一个凸轮周期生效； 其他：保留。
	输出	Done	BOOL	能	TRUE/FALSE	忙标志
		Busy	BOOL	能	TRUE/FALSE	被中止标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	-	是否有错误发生
		ErrorID	DWORD	能	-	错误码(请参见【8.10 错误码对照表】)

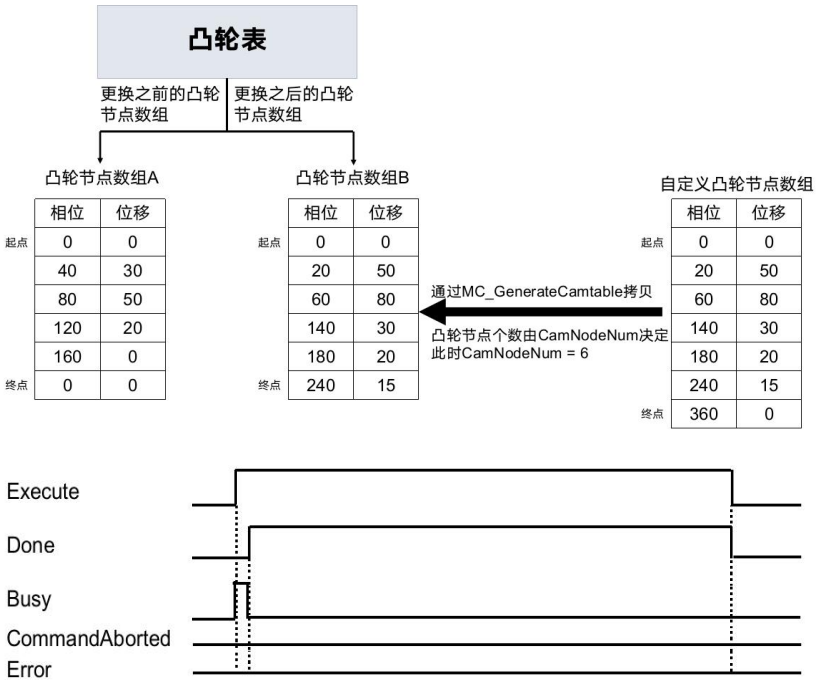
功能

- 该功能块用于更新凸轮表，在启动的上升沿，基于输入变量 CamNode、NodeNum 的值计算出凸轮数据，并更新到 CamTable 指定的凸轮表中，在下一个凸轮周期生效。

注意事项

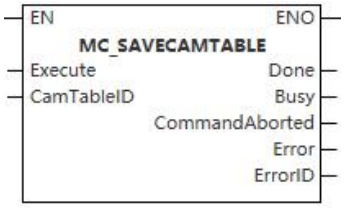
- 当凸轮表不存在时，该功能块不可用。

时序图



3、MC_SAVECAMTABLE

MC_SAVECAMTABLE -保存凸轮表



该功能块用于保存凸轮表。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_SAVECAMTABLE	Execute	●	●	●	●	●																					
	CamTableID																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_SAVECAMTABLE	输入	Execute	BOOL	否	TRUE/FALSE	使能
		CamTableID	WORD	否	[-1,65535]	指定保存凸轮表索引号，-1 代表全部保存
	输出	Done	BOOL	能	TRUE/FALSE	忙标志
		Busy	BOOL	能	TRUE/FALSE	被中止标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	-	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块用于保存指定凸轮表。

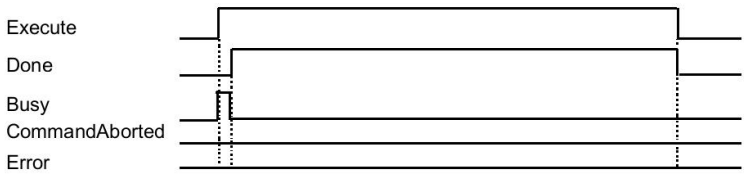
注意事项

- 当凸轮表不存在时，该功能块不可用；
- 在执行本指令的过程中，不能关闭控制器电源，否则会造成数据保存失败，导致凸轮数据丢失。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

时序图

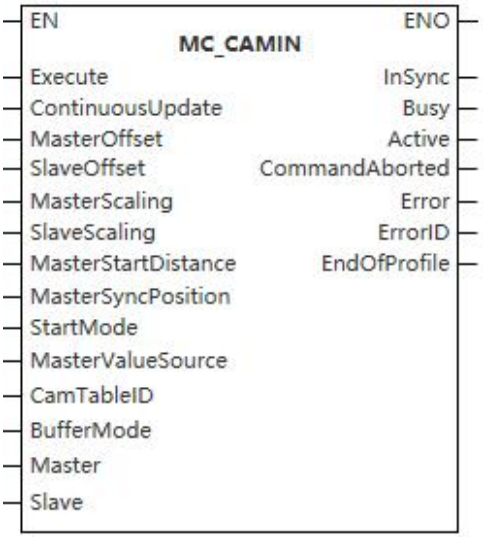


福州富昌维控电子科技有限公司

8.4 凸轮和齿轮多轴运动控制的运动类功能块

1、MC_CAMIN

MC_CAMIN — 凸轮动作开始



该功能块使被控制轴进行凸轮运动。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_CAMIN	Execute	●	●	●	●	●																				
	ContinuousUpdate	●	●	●	●	●																				
	MasterOffset																									
	SlaveOffset																									
	MasterScaling																									
	SlaveScaling																									
	MasterStartDistance																									
	MasterSyncPosition																									
	StartMode																									
	MasterValueSource																									
	CamTableID																									
	Master																									
	Slave																									
	InSync		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	EndOfProfile																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_CAMIN	输入	Execute	BOOL	否	TRUE/FALSE	使能
		ContinuousUpdate	BOOL	能	TRUE/FALSE	持续更新
		MasterOffset	LREAL	能	[-999999999.999999, 999999999.999999]	主轴偏置
		SlaveOffset	LREAL	能	[-999999999.999999, 999999999.999999]	从轴偏置
		MasterScaling	LREAL	否	[-999999999.999999,0)∪(0,999999999.999999]	主轴标度因子
		SlaveScaling	LREAL	否	[-999999999.999999,0)∪(0,999999999.999999]	从轴标度因子
		MasterStartDistance	LREAL	否	[-999999999.999999, 999999999.999999]	主轴开始距离
		MasterSyncPosition	LREAL	否	[-999999999.999999, 999999999.999999]	主轴同步位置
		StartMode	WORD	能	[0,2]	开始模式。 0：绝对； 1：相对； 2：追及。
		MasterValueSource	WORD	能	[0,2]	主轴同步源（命令值、设定值、实际值）
		CamTableID	WORD	否	[0,65535]	凸轮表索引号
		Master	WORD	否	[0,65535]	主轴
		Slave	WORD	否	[0,65535]	从轴
	输出	InSync	BOOL	能	TRUE/FALSE	已同步标识
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）
		EndOfProfile	WORD	能	TRUE/FALSE	电子凸轮表结束信号

功能

该功能以输入的参数 MasterSyncPosition 为起点，在到达 MasterSyncPosition 之前根据设定的参数（主轴绝对相对、从轴绝对相对、开始模式）进行追及或跃迁运动，到达 MasterSyncPosition 位置后，根据 camTableID 获得主轴和从轴之间的位置关系，在设置的主、从轴偏置和标度因子下对该关系进行修正，并进行凸轮运动。

- MasterSyncPosition 在当主轴模式为绝对时，MasterSyncPosition 为绝对位置；
  当主轴模式为相对时，MasterSyncPosition 为相对位置；
- ImSync 状态位在主轴达到 MasterSyncPosition 位置，从轴达到与主轴凸轮表对应的从轴位置时置 ON，一般来说当从轴处于相对模式时，执行 MC_CAMIN 指令，该状态位就会立刻置 ON，当从轴处于绝对或者追及模式时，在从轴跃迁或者追及至与主轴凸轮表对应的从轴位置后置 ON；
- EndOfProfle 状态位在从轴跟随主轴执行完整个凸轮表后会置 ON。StartMode 参数与 MC_CAMTABLESELECT 指令中的 MasterAbsolute/SlaverAbsolute 参数共同决定了主/从轴的运动模式，主轴模式仅由 MasterAbsolute 决定，不受 StartMode 中的值影响。从轴模式表现形式如下：

StartMode	SlaveAbsolute	从轴模式
绝对	相对	相对
绝对	绝对	绝对
相对	相对	相对
相对	绝对	相对
追及	相对	相对追及
追及	绝对	绝对追及

- 主从轴绝对/相对模式在执行 CAMIN 指令时的结果。

主轴模式	从轴模式	结果
相对	相对	CAMIN 执行后，从轴位置跃迁至设置的偏置位置； 主轴运行后，从轴根据凸轮表起始点位进行相对运动。
	绝对	CAMIN 执行后，从轴位置跃迁到凸轮表起始位置（即 0）； 主轴运行后，从轴根据凸轮表对应点位进行运动。
	相对追及	CAMIN 执行后，从轴位置追及至设置的偏置位置； 主轴运行后，从轴根据凸轮表对应点位进行相对运动。
	绝对追及	CAMIN 执行后，从轴位置追及到凸轮表起始位置（即 0）； 主轴运行后，从轴根据凸轮表对应点位进行运动。
绝对	相对	CAMIN 执行后，从轴位置跃迁至设置的偏置位置； 主轴运行后，从轴根据凸轮表对应点位进行相对运动。
	绝对	CAMIN 执行后，从轴位置跃迁到凸轮表主轴当前对应的从轴位置（例：主轴当前位置为 100，凸轮表中主轴点 100 对应的从轴点为 200，CAMIN 执行后从轴跃迁）； 主轴运行后，从轴根据凸轮表对应点位进行相对运动。
	相对追及	CAMIN 执行后，从轴位置追及至设置的偏置位置； 主轴运行后，从轴根据凸轮表对应点位进行相对运动。
	绝对追及	CAMIN 执行后，从轴位置追及到凸轮表主轴当前对应的从轴位置（例：主轴当前位置为 100，凸轮表中主轴点 100 对应的从轴点为 200，CAMIN 执行后从轴跃迁）； 主轴运行后，从轴根据凸轮表对应点位进行相对运动。

- 当主轴为绝对模式时，如果主轴的当前位置不在凸轮表主轴范围内，则自动作周期化处理，例：主轴当前位置为 110，凸轮表中主轴的位置为 0~100，则 CAMIN 执行后默认主轴位置为 10（实际的主轴位置不发生变化）；
- 主从轴倍率、主从轴偏移参数在执行 CAMIN 时生效，不支持过程中修改，不适合的参数会导致从轴位置跃迁。主从中的位置关系为（其中 CAM（）表示主轴在凸轮表上对应的从轴位置）；

  主轴位置 = 主轴位置 * 主轴倍率 + 主轴偏移

  从轴位置 = 从轴倍率 * CAM（主轴位置） + 从轴偏移
- 主从轴倍率不能为 0。

注意事项

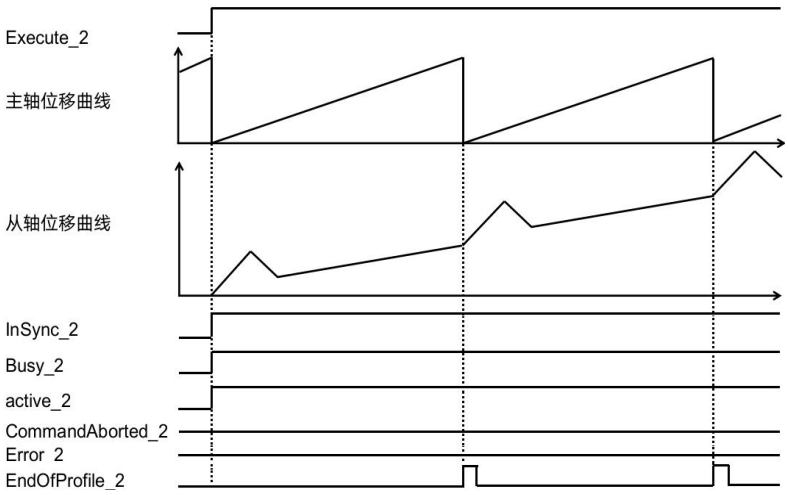
- 当轴不存在时，该功能块不可用。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

时序图

- MC_CAMIN 需和 MC_CAMTABLESELECT 以及单轴运动（如 MC_MOVEABSOLUTE）配合使用，时序图如下所示：



（1）从轴动作因 StartMode 的不同而异

假设凸轮动作的开始条件如下：

- 条件 1 下，主轴会在 20（70~50）的位置进行跃迁，跃迁的位置为 masterSyncPosition 对应的凸轮表位置，在经过 70 的位置时，输出变量 InSync（同步中）变为 TRUE，从轴开始凸轮动作；
- 条件 2 下，主轴会在 20（70~50）的位置进行追及，追及的位置为 masterSyncPosition 对应的凸轮表位置，在经过 70 的位置时，输出变量 InSync（同步中）变为 TRUE，从轴开始凸轮动作；
- 条件 3 下，主轴会在经过 70 的位置时，输出变量 InSync（同步中）变为 TRUE，从轴开始凸轮动作。

输入变量	条件 1	条件 2	条件 3
MasterAbsolute	True（绝对）	True（绝对）	True（绝对）
SlaveAbsolute	True（绝对）	True（绝对）	False（相对）
StartMode	MC_START_MODE_ABSOLUTE	MC_START_MODE_RAMPIN	MC_START_MODE_RELATIVE
MasterSyncPosition	70	70	70
MasterDistance	50	50	50
MasterOffset	0	0	0
SlaveOffset	0	0	0
MasterScaling	1	1	1
SlaveScaling	1	1	1

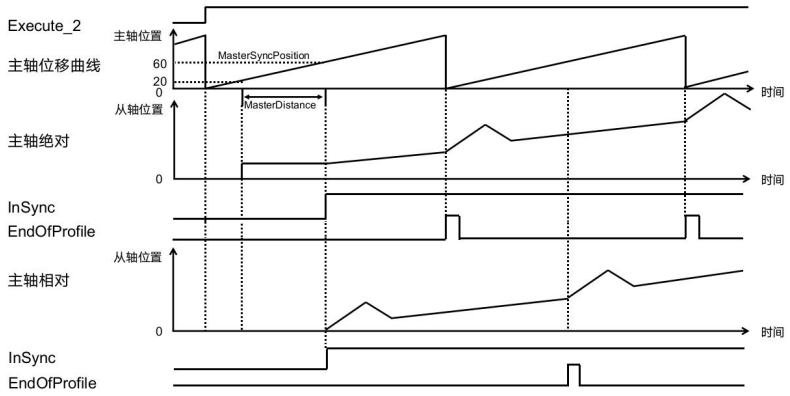
（2）从轴动作因 MasterAbsolute 的不同而异

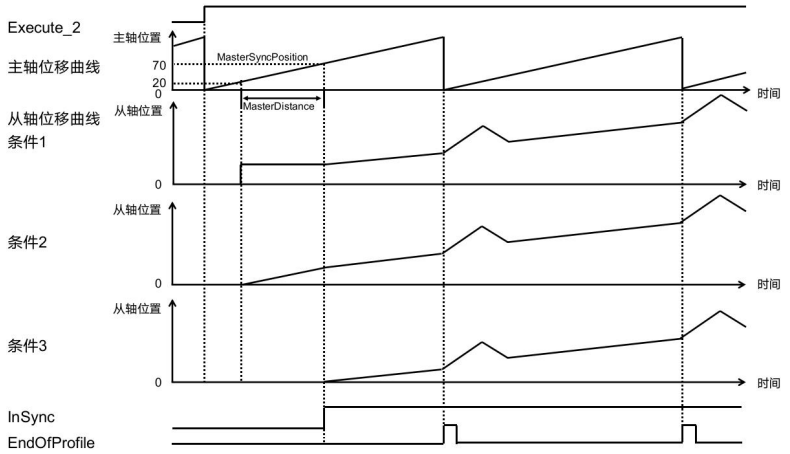
凸轮表

主轴位置	从轴位置
0	0
60	150
120	50
180	180
240	240
300	300
360	360

MasterSyncPosition = 60

主轴绝对		主轴相对	
主轴位置	从轴位置	主轴位置	从轴位置
60	150+从轴在凸轮表起点的位置	0	0+从轴在凸轮表起点的位置
120	50+从轴在凸轮表起点的位置	60	150+从轴在凸轮表起点的位置
180	180+从轴在凸轮表起点的位置	120	50+从轴在凸轮表起点的位置
240	240+从轴在凸轮表起点的位置	180	180+从轴在凸轮表起点的位置
300	300+从轴在凸轮表起点的位置	240	240+从轴在凸轮表起点的位置
360	360+从轴在凸轮表起点的位置	300	300+从轴在凸轮表起点的位置
60	150+从轴在凸轮表起点的位置	360	360+从轴在凸轮表起点的位置





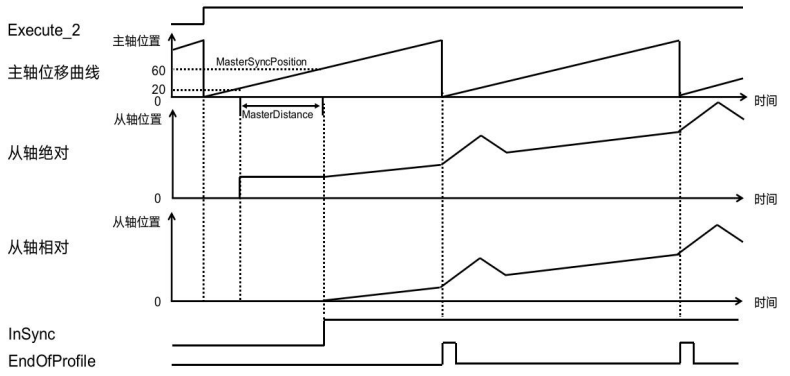
（3）从轴动作因 SlaveAbsolute 的不同而异

凸轮表

主轴位置	从轴位置
0	0
60	150
120	50
180	180
240	240
300	300
360	360

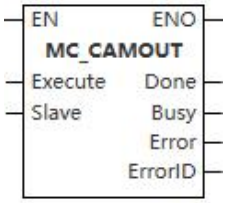
MasterSyncPosition = 60

从轴绝对		从轴相对	
主轴位置	从轴位置	主轴位置	从轴位置
60	150+从轴在凸轮表起点的位置	60	150+从轴在凸轮表起点的位置-150
120	50+从轴在凸轮表起点的位置	120	50+从轴在凸轮表起点的位置-150
180	180+从轴在凸轮表起点的位置	180	180+从轴在凸轮表起点的位置-150
240	240+从轴在凸轮表起点的位置	240	240+从轴在凸轮表起点的位置-150
300	300+从轴在凸轮表起点的位置	300	300+从轴在凸轮表起点的位置-150
360	360+从轴在凸轮表起点的位置	360	360+从轴在凸轮表起点的位置-150



2、MC_CAMOUT

MC_CAMOUT



该功能块使被控制轴以当前速度进行匀速运动。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_CAMOUT	Execute	●	●	●	●	●																					
	Slave																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_CAMOUT	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Slave	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

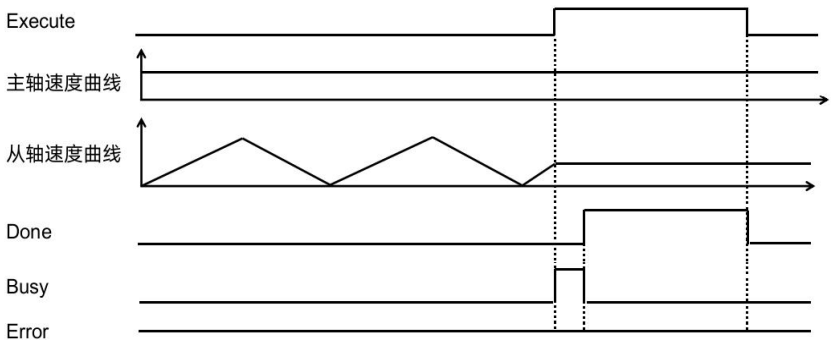
功能

- 该功能将规定的从轴电子凸轮规定的位置关系脱离。执行 MC_CamOut 功能块，使从轴从同步状态转换到连续运动状态。当前从轴的速度被保持。

注意事项

- 轴不存在或不在凸轮状态时，该功能块不可用。

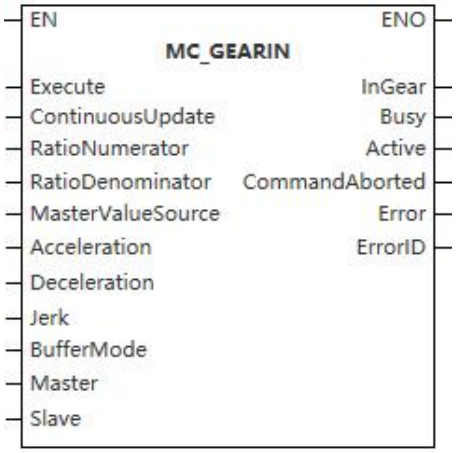
时序图



福州富昌维控电子科技有限公司

3、MC_GEARIN

MC_GEARIN



该功能块使被控制轴进行齿轮运动。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GEARIN	Execute	●	●	●	●	●																				
	ContinuousUpdate	●	●	●	●	●																				
	RatioNumerator																									
	RatioDenominator																									
	MasterValueSource																									
	Acceleration																									
	Deceleration																									
	Jerk																									
	Master																									
	Slave																									
	InGear		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_GEARIN	输入	Execute	BOOL	否	TRUE/FALSE	使能
		ContinuousUpdate	BOOL	能	TRUE/FALSE	
		RatioNumerator	INT16	否	[-32768,32767]	齿轮速度比（分子）
		RatioDenominator	UINT16	否	[0,65535]	齿轮速度比（分母）
		MasterValueSource	WORD	能	[0,2]	主轴同步源

		Acceleration	LREAL	否	(0,999999999.999999]	加速度
		Deceleration	LREAL	否	(0,999999999.999999]	减速度
		Jerk	LREAL	能	[0,999999999.999999]	加加速度,仅在 S 型曲线生效
		Master	WORD	否	[0,65535]	主轴
		Slave	WORD	否	[0,65535]	从轴
	输出	InGear	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块用于建立主轴和从轴之间速度比的同步，一开始从轴以主轴速度乘以速度比为目标速度，并以设定的加速度、加加速度运动，到达目标运动时，按照主轴速度执行齿轮运动。

注意事项

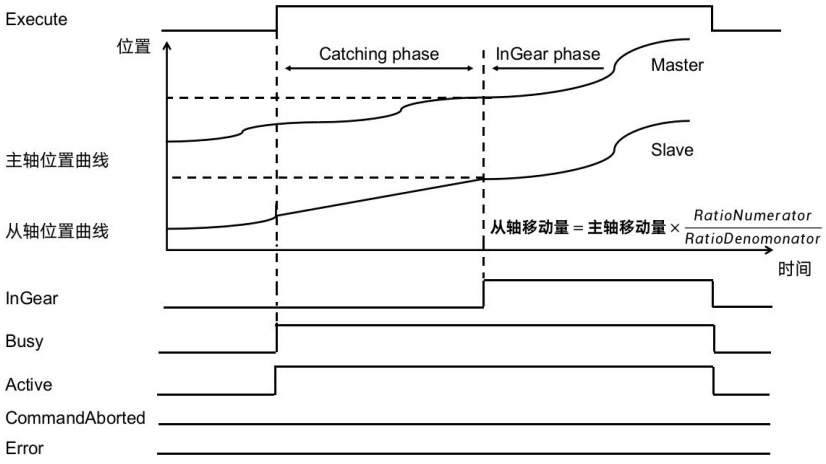
- 当轴不存在时，该功能块不可用。

错误码

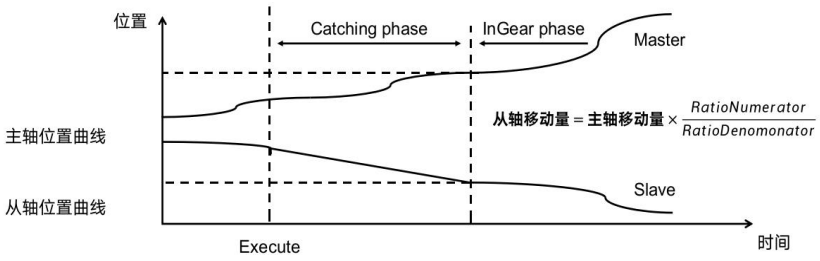
错误码	内容
4084H	输入了超出可指定范围的数据。

时序图

- 开始动作后，从轴以主轴速度乘以齿轮比得到的速度为目标速度，进行加减速动作；
- 到达目标位置之前称为 Catching phase（追赶中）、到达后称为 InGear phase（齿轮同步中）；
- 齿轮比为正，从轴与主轴同方向移动。



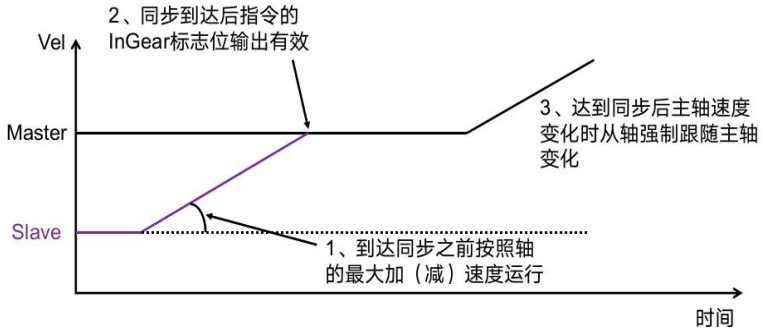
- 齿轮比为负，从轴与主轴反方向移动。



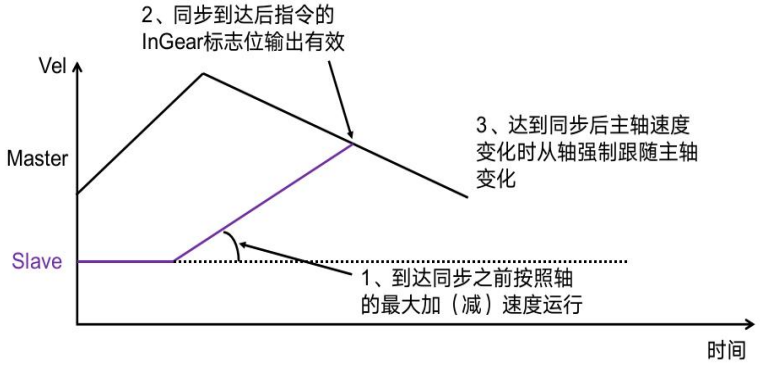
- 在达到同步之前，从轴按照轴设定的最大加速度为加减速度进行运动，从轴速度等于主轴的速度乘以齿轮比时，认为齿轮啮合，此后从轴完全跟随主轴变化。

福州富昌维控电子科技有限公司

① 同步之前主轴保持匀速运动（按照 1：1 的齿轮比触发齿轮指令）

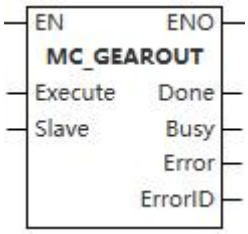


② 同步之前主轴变速运动（按照 1：1 的齿轮比触发齿轮指令）



4、MC_GEAROUT

MC_GEAROUT



该功能块用于将从轴脱离与主轴之间的速度比。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GEAROUT	Execute	●	●	●	●	●																					
	Slave																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_GEAROUT	输入	Execute	BOOL	否	TRUE/FALSE	使能
		Slave	WORD	否	[0,65535]	轴号
	输出	Done	BOOL	能	TRUE/FALSE	完成标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码（请参见【8.10 错误码对照表】）

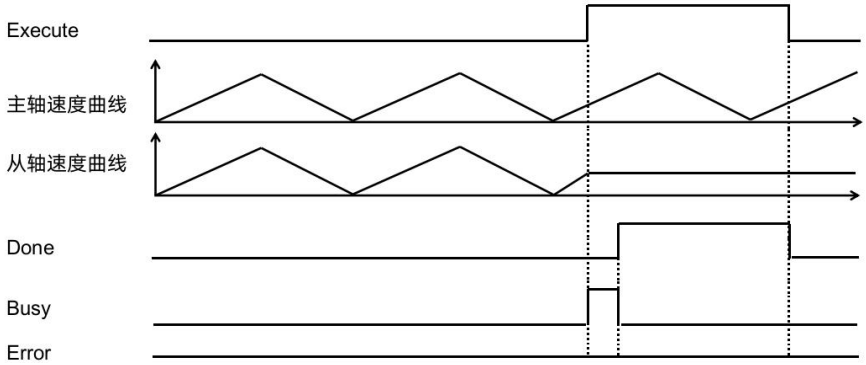
功能

- 该功能块用于将从轴脱离与主轴之间的速度比，并保持其脱离时刻的速度继续运动。

注意事项

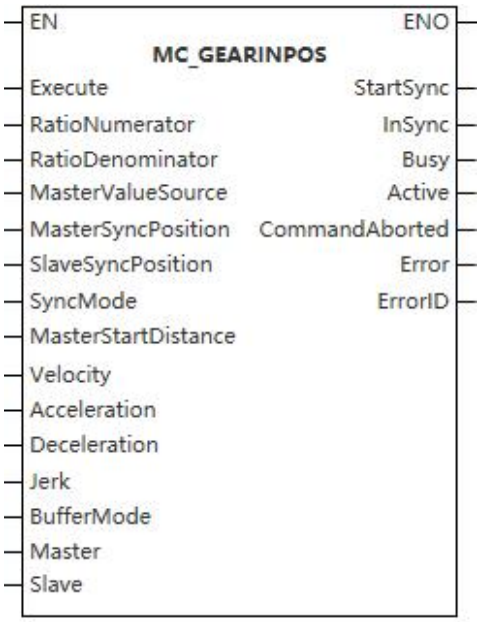
- 从轴不存在或不在齿轮状态时，该功能块不可用。

时序图



5、MC_GEARINPOS

MC_GEARINPOS



该功能块用于设定参数。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GEARINPOS	Execute	●	●	●	●	●																				
	RatioNumerator																									
	RatioDenominator																									
	MasterValueSource																									
	MasterSyncPosition																									
	SlaveSyncPosition																									
	SyncMode																									
	MasterStartDistance																									
	Velocity																									
	Acceleration																									
	Deceleration																									
	Jerk																									
	Master																									
	Slave																									
	StartSync																									
	InSync		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
MC_GEARINPOS	输入	Execute	BOOL	否	TRUE/FALSE	使能
		RatioNumerator	INT16	否	[-32768,32767]	齿轮速度比（分子）
		RatioDenominator	UINT16	否	[0,65535]	齿轮速度比（分母）
		MasterValueSource	WORD	能	[0,2]	主轴同步源
		MasterSyncPosition	LREAL	否	[-999999999.999999, 999999999.999999]	主轴同步位置
		SlaveSyncPosition	LREAL	否	[-999999999.999999, 999999999.999999]	从轴同步位置
		SyncMode	WORD	能	[0,2]	同步模式。 0：shortest； 1：catchup； 2：slowdown。
		MasterStartDistance	LREAL	能	[-999999999.999999, 999999999.999999]	主轴开始距离
		Velocity	LREAL	否	(0,999999999.999999]	速度
		Acceleration	LREAL	否	(0,999999999.999999]	加速度
		Deceleration	LREAL	否	(0,999999999.999999]	减速度
		Jerk	LREAL	能	[0,999999999.999999]	加加速度，仅在 S 型曲线生效
		Master	WORD	否	[0,65535]	主轴
		Slave	WORD	否	[0,65535]	从轴
	输出	StartSync	BOOL	能	TRUE/FALSE	开始同步标识
		InSync	BOOL	能	TRUE/FALSE	已同步标志
		Busy	BOOL	能	TRUE/FALSE	忙标志
		Active	BOOL	能	TRUE/FALSE	有效标志
		CommandAborted	BOOL	能	TRUE/FALSE	中止标志
		Error	BOOL	能	TRUE/FALSE	是否有错误发生
		ErrorID	DWORD	能	-	错误码(请参见【8.10 错误码对照表】)

功能

- 该功能块用于通过设定的最大速度、加减速度、加加速度及同步模式使从轴在特定位置与主轴之间实现规定速度比。

注意事项

- 当轴不存在时，该功能块不可用。

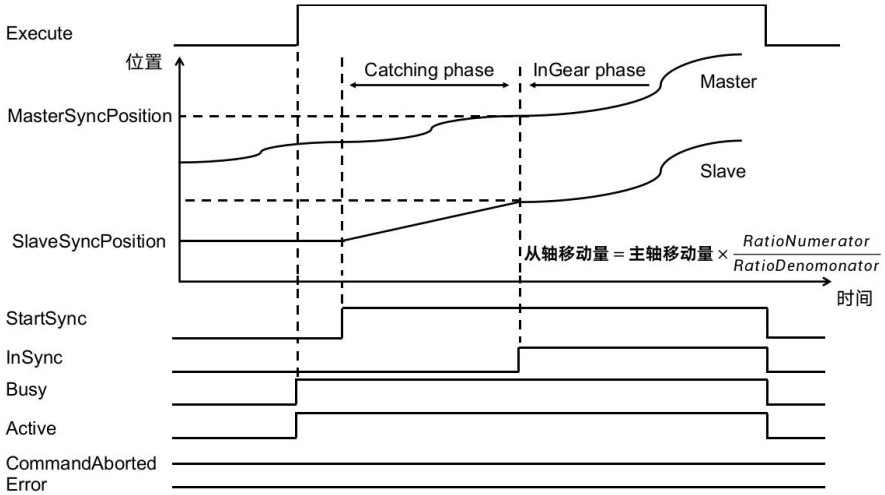
错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

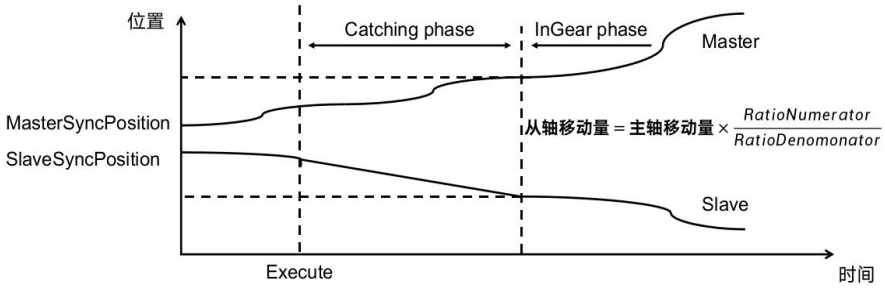
时序图

- 开始动作后，从轴以主轴速度乘以齿轮比得到的速度为目标速度，进行加减速动作；
- （MasterSyncPosition - MasterStartDistance）得到的位置称为开始位置，当主轴到达该位置之后，从轴才开始追赶目标；
- 齿轮比为正，从轴与主轴同方向移动。

福州富昌维控电子科技有限公司

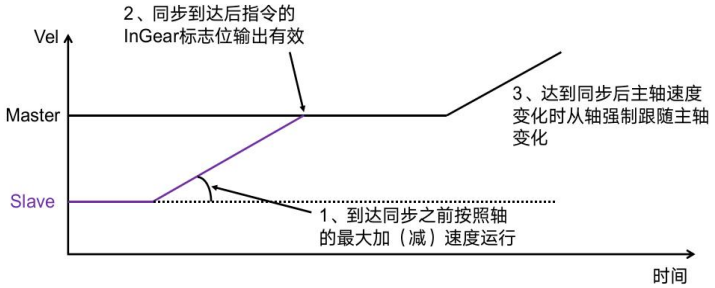


- 齿轮比为负，从轴与主轴反方向移动。

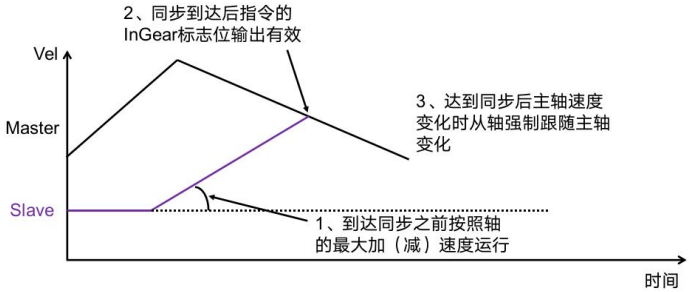


- 在达到同步之前，从轴按照轴设定的最大加速度为加减速度进行运动，从轴速度等于主轴的速度乘以齿轮比时，认为齿轮啮合，此后从轴完全跟随主轴变化。

① 同步之前主轴保持匀速运动（按照 1：1 的齿轮比触发齿轮指令）



② 同步之前主轴变速运动（按照 1：1 的齿轮比触发齿轮指令）



gearInPos 通过以下三种模式进行追及运动，在达到目标速度后，均跟随主轴进行齿轮运动：

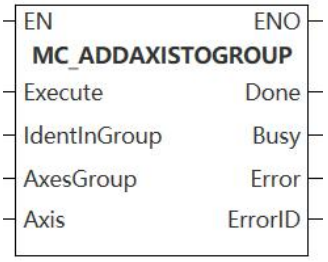
- shortest：以匀加速模式进行追及；
- Catchuo：以 S 型曲线进行追及；
- Slowdown：以 T 型曲线进行追及。

注：追赶中的移动量较短时，可能会出现从轴到目标位置后速度直接突变到目标速度的情况。

8.5 轴组运动控制的管理类功能

1、MC_ADDAXISTOGROUP

MC_ADDAXISTOGROUP



该功能块是添加轴到轴组。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_ADDAXISTOGROUP	Execute	●	●	●	●	●																					
	IdentInGroup																										
	AxesGroup																										
	Axis																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_ADDAXISTOGROUP	Execute	BOOL	否	TRUE/FALSE	使能
	IdentInGroup	WORD	否	[0,2]	轴顺序标识
	AxesGroup	WORD	否	[0,65535]	轴组号
	Axis	WORD	否	[0,65535]	轴号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

- Execute 上升沿触发，将轴 Axis 添加到轴组 AxesGroup。命令不被缓冲，轴不发生运动；
- IndentInGroup 时，轴组中添加轴顺序的标识。

注意事项

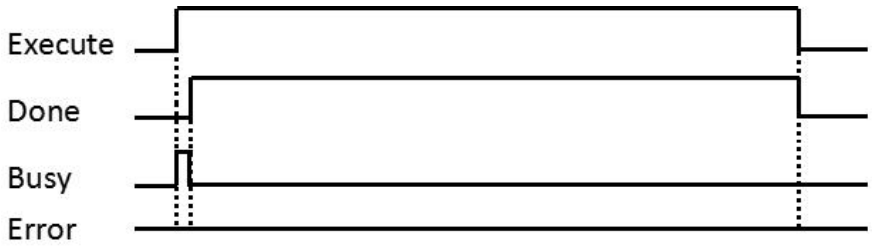
- 轴组序号不存在，会产生报错；
- 轴组使能后，添加会报错；
- 只添加到轴中无法进行运动，需要调用轴组 MC_GroupEnable 使能后，方可进行运动。

福州富昌维控电子科技有限公司

错误码

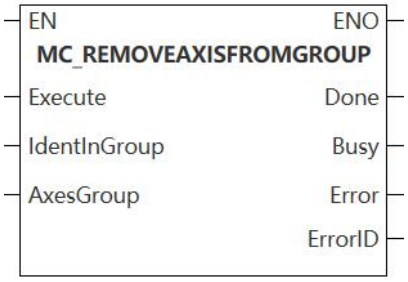
错误码	内容
4084H	输入了超出可指定范围的数据。

时序图



2、MC_REMOVEAXISFROMGROUP

MC_REMOVEAXISFROMGROUP



该功能块是从轴组中删除轴。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_REMOVEAXISFROMGROUP	Execute	●	●	●	●	●																				
	IdentInGroup																									
	AxesGroup																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_REMOVEAXISFROMGROUP	Execute	BOOL	否	TRUE/FALSE	使能
	IdentInGroup	WORD	否	[0,2]	轴顺序标识
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块从轴组 AxexGroup 删除由 IdentInGroup 标志顺序号的轴，不被缓存和不发生运动。

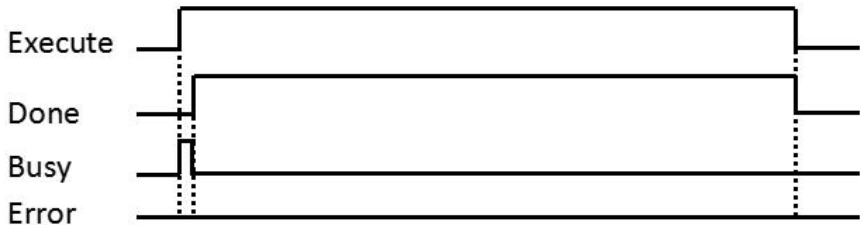
注意事项

- 轴组序号不存在会产生报错；
- 轴组正在运行和未关闭使能，移除会报错；
- 轴组只有关闭使能后，方可进行单轴运行，是否移除不影响单轴运行。

错误码

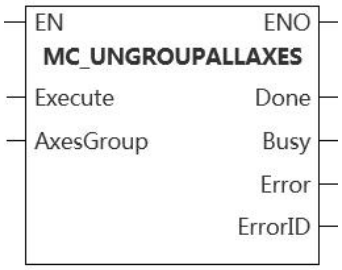
错误码	内容
4084H	输入了超出可指定范围的数据。

时序图



3、MC_UNGROUPALLAXES

MC_UNGROUPALLAXES



该功能块是删除轴组所有轴。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_UNGROUPALLAXES	Execute	●	●	●	●	●																				
	AxesGroup																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_UNGROUPALLAXES	Execute	BOOL	否	TRUE/FALSE	使能
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

- 从轴组 AxesGroup 删除所有轴，不被缓冲和不发生运动。

注意事项

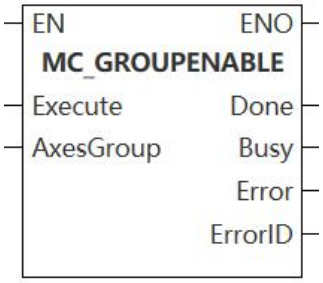
- 轴组序号不存在会产生报错；
- 轴组正在运行和未关闭使能，删除会报错。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

4、MC_GROUPENABLE

MC_GROUPENABLE



该功能块是轴组使能。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPENABLE	Execute	●	●	●	●	●																				
	AxesGroup																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPENABLE	Execute	BOOL	否	TRUE/FALSE	使能
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

- 在 Execute 上升沿，使能轴组运行。

注意事项

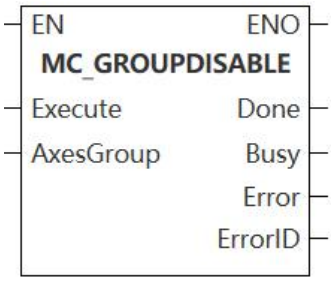
- 轴组运动只有使能才能运行，否则会进行报错；
- 单轴在运行时，使能会报错。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

5、MC_GROUPDISABLE

MC_GROUPDISABLE



该功能块是轴组禁止。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPDISABLE	Execute	●	●	●	●	●																				
	AxesGroup																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPDISABLE	Execute	BOOL	否	TRUE/FALSE	使能
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

- 在 Execute 上升沿，关闭轴组使能。

注意事项

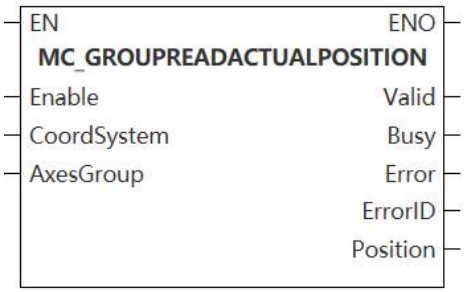
- 轴组关闭使能方可进行单轴运行；
- 轴组在运行时，关闭使能会报错。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

6、MC_GROUPREADACTUALPOSITION

MC_GROUPREADACTUALPOSITION



该功能块是读取轴组各轴实际位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPREADACTUALPOSITION	Enable	●	●	●	●	●																					
	CoordSystem																										
	AxesGroup																										
	Valid		●	●	●	●																					
	Busy		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										
	Position																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPREADACTUALPOSITION	Enable	BOOL	否	TRUE/FALSE	使能
	CoordSystem	WORD	是	0	坐标系
	AxesGroup	WORD	否	[0,65535]	轴组号
	Valid	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）
	Position	LREAL 数组	是		位置

功能

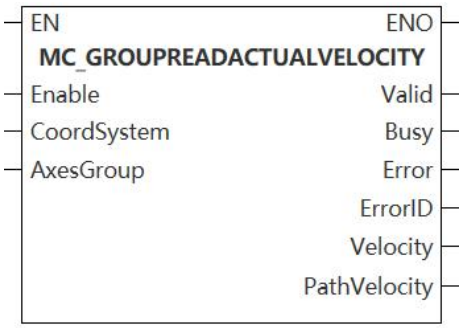
- 当 Enable=true 时，连续返回被选中轴组在被选择坐标系中各轴的位置。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

7、MC_GROUPREADACTUALVELOCITY

MC_GROUPREADACTUALVELOCITY



该功能块是读取轴组各轴实际速度。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPREADACTUALVELOCITY	Enable	●	●	●	●	●																					
	CoordSystem																										
	AxesGroup																										
	Valid		●	●	●	●																					
	Busy		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										
	Velocity																										
	PathVelocity																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPREADACTUALVELOCITY	Enable	BOOL	否	TRUE/FALSE	使能
	CoordSystem	WORD	是	0	坐标系
	AxesGroup	WORD	否	[0,65535]	轴组号
	Valid	BOOL	是	TRUE/FALSE	输出是否有效
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）
	Velocity	LREAL 数组	是	-	速度
	PathVelocity	LREAL	是	-	路径速度

功能

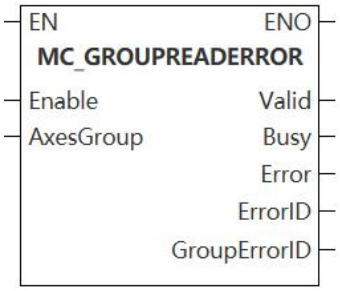
- 当 Enable=true 时，连续返回被选中轴组在被选择坐标系中各轴的速度和当前 TCP 路径速度（组合速度的结果）。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

8、MC_GROUPREADERROR

MC_GROUPREADERROR



该功能块是轴组读取错误。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPREADERROR	Enable	●	●	●	●	●																				
	AxesGroup																									
	Valid		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	GroupErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPREADERROR	Enable	BOOL	否	TRUE/FALSE	使能
	AxesGroup	WORD	否	[0,65535]	轴组号
	Valid	BOOL	是	TRUE/FALSE	输出是否有效
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）
	GroupErrorID	DWORD	是	-	轴组错误码

功能

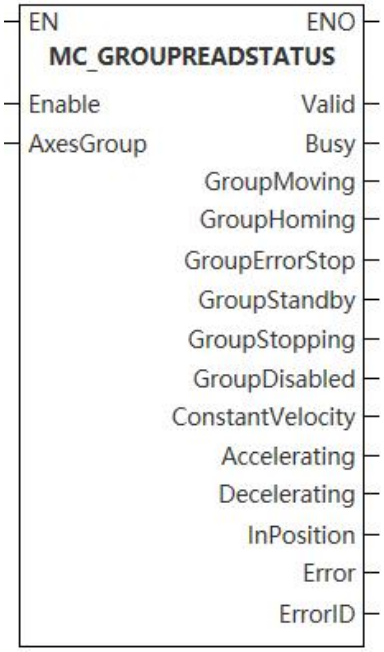
- 当 Enable=true 时，读取轴组出错状态。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

9、MC_GROUPREADSTATUS

MC_GROUPREADSTATUS



该功能块是读取当前已激活轴组状态。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPREADSTATUS	Enable	●	●	●	●	●																					
	AxesGroup																										
	Valid		●	●	●	●																					
	Busy		●	●	●	●																					
	GroupMoving		●	●	●	●																					
	GroupHoming		●	●	●	●																					
	GroupErrorStop		●	●	●	●																					
	GroupStandby		●	●	●	●																					
	GroupStopping		●	●	●	●																					
	GroupDisabled		●	●	●	●																					
	ConstantVelocity		●	●	●	●																					
	Accelerating		●	●	●	●																					
	Decelerating		●	●	●	●																					
	InPosition		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPREADSTATUS	Enable	BOOL	否	TRUE/FALSE	使能
	AxesGroup	WORD	否	[0,65535]	轴组号
	Valid	BOOL	是	TRUE/FALSE	输出是否有效
	Busy	BOOL	是	TRUE/FALSE	忙标志
	GroupMoving	BOOL	是	TRUE/FALSE	轴组运动标志
	GroupHoming	BOOL	是	TRUE/FALSE	轴组回零中标志
	GroupErrorStop	BOOL	是	TRUE/FALSE	轴组错误停止标志
	GroupStandby	BOOL	是	TRUE/FALSE	轴组静止标志
	GroupStopping	BOOL	是	TRUE/FALSE	轴组停止中标志
	GroupDisabled	BOOL	是	TRUE/FALSE	轴组未激活标志
	ConstantVelocity	BOOL	是	TRUE/FALSE	恒速
	Accelerating	BOOL	是	TRUE/FALSE	加速
	Decelerating	BOOL	是	TRUE/FALSE	减速
	InPosition	BOOL	是	TRUE/FALSE	已达目标位置
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

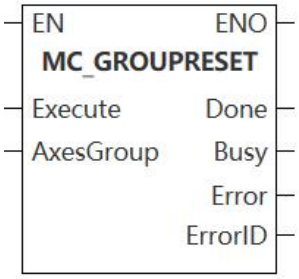
- 当 Enable=true 时，连续返回已经使能轴组的状态，其状态由输出的该项标志置位标记；
- ConstantVelocity 是命令路径上恒定速度移动被置位，Accelerating 和 Decelerating 是命令路径上速度增加或减少时被置位；
- InPosition 表示当前已经到达目标位置被置位。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

10、MC_GROUPRESET

MC_GROUPRESET



该功能块是轴组复位。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPRESET	Execute	●	●	●	●	●																				
	AxesGroup																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPRESET	Execute	BOOL	否	TRUE/FALSE	使能
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

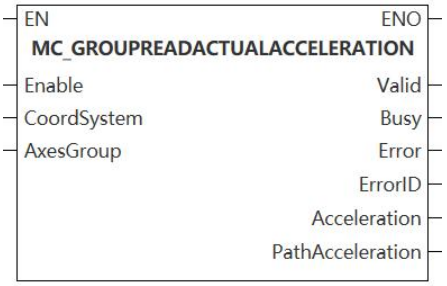
- 当 Excute 上升沿时将轴组从错误状态复位成正常使能状态，与单轴 MC_Reset 类似。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

11、MC_GROUPREADACTUALACCELERATION

MC_GROUPREADACTUALACCELERATION



该功能块是读取轴组各轴实际加速度。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPREADACTUALACCELERATION	Enable	●	●	●	●	●																				
	CoordSystem																									
	AxesGroup																									
	Valid		●	●	●	●																				
	Busy		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									
	Acceleration																									
	PathAcceleration																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPREADACTUALACCELERATION	Enable	BOOL	否	TRUE/FALSE	使能
	CoordSystem	WORD	是	0	坐标系
	AxesGroup	WORD	否	[0,65535]	轴组号
	Valid	BOOL	是	TRUE/FALSE	输出是否有效
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）
	Acceleration	LREAL 数组	是	-	加速度
	PathAcceleration	LREAL	是	-	路径加速度

功能

- 当 Enable=true 时，连续返回被选中轴组在被选择坐标系中各轴的加速度和当前 TCP 路径加速度（组合加速度的结果）。

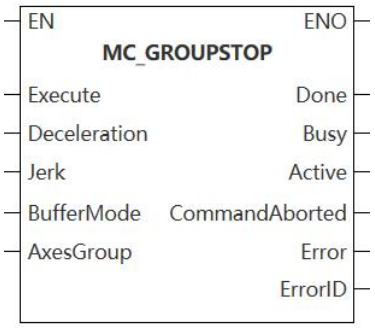
错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

8.6 轴组运动类

1、MC_GROUPSTOP

MC_GROUPSTOP



该功能块是使轴组停止。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPSTOP	Execute	●	●	●	●	●																					
	Deceleration																										
	Jerk																										
	AxesGroup																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPSTOP	Execute	BOOL	否	TRUE/FALSE	使能
	Deceleration	LREAL	否	(0,999999999.999999]	减速度
	Jerk	LREAL	是	[0,999999999.999999]	加加速度
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

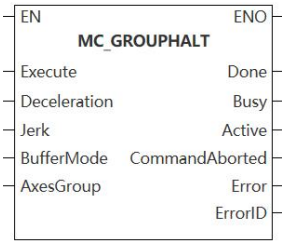
- 当 Execute 上升沿时使轴组减速停止，中止任何正在执行的功能块，Execute=true，轴组处于停止状态，无法运行其余轴组运动指令；
- 轴组速度为零，Done 被置位，如果 Execute=false，回归正常正常轴组使能状态；
- 只能用 MC_GroupDisable 中断轴组。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

2、MC_GROUPHALT

MC_GROUPHALT



该功能块是使轴组暂停。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_GROUPHALT	Execute	●	●	●	●	●																					
	Deceleration																										
	Jerk																										
	AxesGroup																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_GROUPHALT	Execute	BOOL	否	TRUE/FALSE	使能
	Deceleration	LREAL	否	(0,999999999.999999]	减速度
	Jerk	LREAL	是	[0,999999999.999999]	加加速度
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

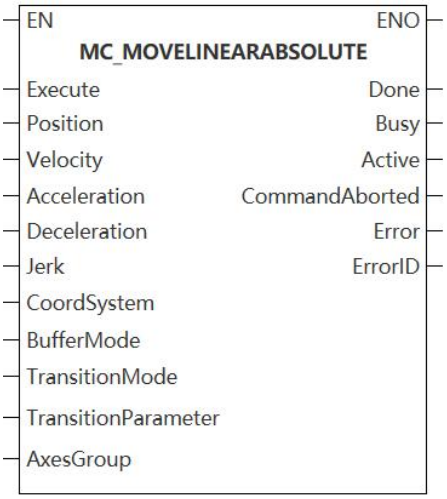
- 当 Execute 上升沿时使轴组减速停止，中止任何正在执行的功能块，轴组速度为零，Done 被置位；
- 非缓冲模式：轴组减速期间，他可能去设置其他运动指令，这些运动指令将中止本功能和被立即执行。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

3、MC_MOVELINEARABSOLUTE

MC_MOVELINEARABSOLUTE



该功能块是直线插补绝对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVELINEARABSOLUTE	Execute	●	●	●	●	●																					
	Position																										
	Velocity																										
	Acceleration																										
	Deceleration																										
	Jerk																										
	CoordSystem																										
	BufferMode																										
	TransitionMode																										
	TransitionParameter																										
	AxesGroup																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_MOVELINEARABSOLUTE	Execute	BOOL	否	TRUE/FALSE	使能
	Position	LREAL 数组	否	-	位置
	Velocity	LREAL	否	(0,999999999.999999]	速度
	Acceleration	LREAL	否	(0,999999999.999999]	加速度
	Deceleration	LREAL	否	(0,999999999.999999]	减速度
	Jerk	LREAL	是	[0,999999999.999999]	加加速度，只有轴组 S 型曲线生效
	CoordSystem	WORD	是	0	坐标系
	BufferMode	WORD	是	[0,1]	模式选择详见【8.9.5 模式选择说明】
	TransitionMode	WORD	是	[0,2]	过渡模式详见【8.9.4 过渡模式说明】
	TransitionParameter	LREAL	是	[0,999999999.999999]	过渡参数详见【8.9.4 过渡模式说明】
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

运动类 FB

- 该功能块执行直线插补运动，将 TCP 实际位置移动到特定 CoordSystem 规定的绝对位置。

注意事项

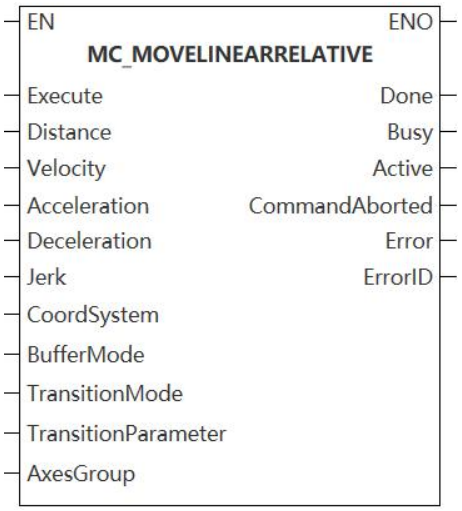
- 轴组使能后方可运动；
- 所有速度、加速度均为安全速度、加速度，以能达到的最大速度、加速度为准，（受 FB 内设置的规划速度、轴内限制速度、加速度影响）若达不到相应进行降低；
- 直线运动允许被打断；
- 加加速度仅在 S 型曲线生效。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

4、MC_MOVELINEARRELATIVE

MC_MOVELINEARRELATIVE



该功能块是直线插补相对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVELINEARRELATIVE	Execute	●	●	●	●	●																					
	Distance																										
	Velocity																										
	Acceleration																										
	Deceleration																										
	Jerk																										
	CoordSystem																										
	BufferMode																										
	TransitionMode																										
	TransitionParameter																										
	AxesGroup																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_MOVELINEARRELATIVE	Execute	BOOL	否	TRUE/FALSE	使能
	Distance	LREAL 数组	否	-	相对位置
	Velocity	LREAL	否	(0,999999999.999999]	速度
	Acceleration	LREAL	否	(0,999999999.999999]	加速度
	Deceleration	LREAL	否	(0,999999999.999999]	减速度
	Jerk	LREAL	是	[0,999999999.999999]	加加速度，只有轴组 S 型曲线生效
	CoordSystem	WORD	是	0	坐标系
	BufferMode	WORD	是	[0,1]	模式选择详见【8.9.5 模式选择说明】
	TransitionMode	WORD	是	[0,2]	过渡模式详见【8.9.4 过渡模式说明】
	TransitionParameter	LREAL	是	[0,999999999.999999]	过渡参数详见【8.9.4 过渡模式说明】
	AxesGroup	WORD	是	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

运动类 FB

- 该功能块执行直线插补运动，将 TCP 实际位置移动到特定 CoordSystem 规定的绝对位置。

注意事项

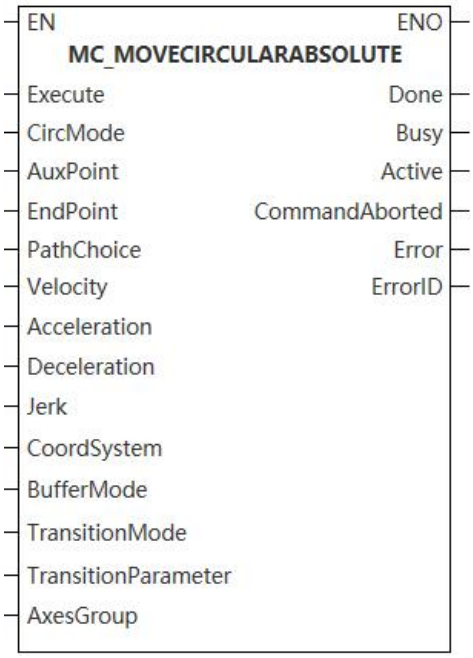
- 轴组使能后方可运动；
- 所有速度、加速度均为安全速度、加速度，以能达到的最大速度、加速度为准，（受 FB 内设置的规划速度、轴内限制速度、加速度影响）若达不到相应进行降低；
- 直线运动允许被打断；
- 加加速度仅在 S 型曲线生效。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

5、MC_MOVECIRCULARABSOLUTE

MC_MOVECIRCULARABSOLUTE



该功能块是圆弧插补绝对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVECIRCULARABSOLUTE	Execute	●	●	●	●	●																				
	CircMode																									
	AuxPoint																									
	EndPoint																									
	PathChoice																									
	Velocity																									
	Acceleration																									
	Deceleration																									
	Jerk																									
	CoordSystem																									
	BufferMode																									
	TransitionMode																									
	TransitionParameter																									
	AxesGroup																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_MOVECIRCULARABSOLUTE	Execute	BOOL	否	TRUE/FALSE	使能
	CircMode	WORD	否	[0,2]	圆弧构建模式详见【8.9.3 圆弧模式说明】
	AuxPoint	LREAL 数组	否	-	辅助点详见【8.9.3 圆弧模式说明】
	EndPoint	LREAL 数组	否	-	终点详见【8.9.3 圆弧模式说明】
	PathChoice	WORD	否	[0.1]	圆弧路径选择详见【8.9.3 圆弧模式说明】
	Velocity	LREAL	否	(0,999999999.999999]	速度
	Acceleration	LREAL	否	(0,999999999.999999]	加速度
	Deceleration	LREAL	否	(0,999999999.999999]	减速度
	Jerk	LREAL	是	[0,999999999.999999]	加加速度，只有轴组 S 型曲线生效
	CoordSystem	WORD	是	0	坐标系
	BufferMode	WORD	是	[0,1]	模式选择详见【8.9.5 模式选择说明】
	TransitionMode	WORD	是	[0,2]	过渡模式详见【8.9.4 过渡模式说明】
	TransitionParameter	LREAL	是	[0,999999999.999999]	过渡参数详见【8.9.4 过渡模式说明】
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是		错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块根据 AuxPoint、EndPoint 和 CircMode 规定的圆弧进行插补，轴组按照圆弧运动到绝对位置 EndPoint，圆弧运动方向由 PathChoice 确定；
- 过渡模式见详解，过渡参数是过渡模式的附加参数，根据过渡模式不同生效；
- AuxPoint、EndPoint 是根据 CoordSystem 规定的从功能块被激活开始的位置到坐标系统中的位置。

注意事项

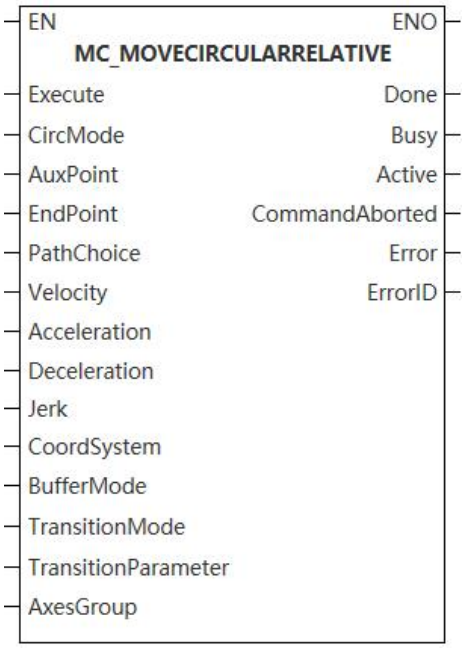
- 轴组使能后方可运动；
- 所有速度、加速度均为安全速度、加速度，以能达到的最大速度、加速度为准，（受 FB 内设置的规划速度、轴内限制速度、加速度影响）若达不到相应进行降低；
- 圆弧运动允许被打断；
- 加加速度仅在 S 型曲线生效。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

6、MC_MOVECIRCULARRELATIVE

MC_MOVECIRCULARRELATIVE



该功能块是圆弧插补相对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVECIRCULARRELATIVE	Execute	●	●	●	●	●																					
	CircMode																										
	AuxPoint																										
	EndPoint																										
	PathChoice																										
	Velocity																										
	Acceleration																										
	Deceleration																										
	Jerk																										
	CoordSystem																										
	BufferMode																										
	TransitionMode																										
	TransitionParameter																										
	AxesGroup																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_MOVECIRCULARRELATIVE	Execute	BOOL	否	TRUE/FALSE	使能
	CircMode	WORD	否	[0,2]	圆弧构建模式详见【8.9.3 圆弧模式说明】
	AuxPoint	LREAL 数组	否	-	辅助点详见【8.9.3 圆弧模式说明】
	EndPoint	LREAL 数组	否	-	终点详见【8.9.3 圆弧模式说明】
	PathChoice	WORD	否	[0,1]	圆弧插补路径详见【8.9.3 圆弧模式说明】
	Velocity	LREAL	否	(0,999999999.999999]	速度
	Acceleration	LREAL	否	(0,999999999.999999]	加速度
	Deceleration	LREAL	否	(0,999999999.999999]	减速度
	Jerk	LREAL	是	[0,999999999.999999]	加加速度，只有轴组 S 型曲线生效
	CoordSystem	WORD	是	0	坐标系
	BufferMode	WORD	是	[0,1]	模式选择详见【8.9.5 模式选择说明】
	TransitionMode	WORD	是	[0,2]	过渡模式详见【8.9.4 过渡模式说明】
	TransitionParameter	LREAL	是	[0,999999999.999999]	过渡速度详见【8.9.4 过渡模式说明】
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块根据 AuxPoint、EndPoint 和 CircMode 规定的圆弧进行插补，轴组按照圆弧运动到相对位置 EndPoint，圆弧运动方向由 PathChoice 确定；
- 过渡模式见详解，过渡参数是过渡模式的附加参数，根据过渡模式不同生效；
- AuxPoint、EndPoint 是根据 CoordSystem 规定的从功能块被激活开始的位置到坐标系统中的位置。

注意事项

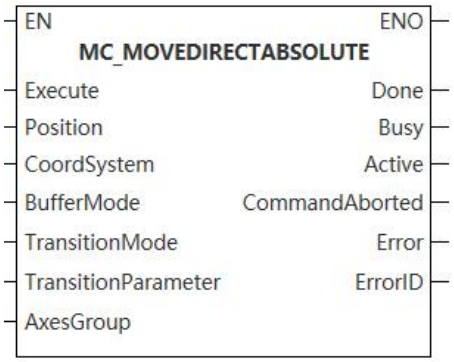
- 轴组使能后方可运动；
- 所有速度、加速度均为安全速度、加速度，以能达到的最大速度、加速度为准，（受 FB 内设置的规划速度、轴内限制速度、加速度影响）若达不到相应进行降低；
- 圆弧运动允许被打断；
- 加加速度仅在 S 型曲线生效。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

7、MC_MOVEDIRECTABSOLUTE

MC_MOVEDIRECTABSOLUTE



该功能块是快速定位绝对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVEDIRECTABSOLUTE	Execute	●	●	●	●	●																					
	Position																										
	CoordSystem																										
	BufferMode																										
	TransitionMode																										
	TransitionParameter																										
	AxesGroup																										
	Done		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_MOVEDIRECTABSOLUTE	Execute	BOOL	否	TRUE/FALSE	使能
	Position	LREAL 数组	否	-	位置
	CoordSystem	WORD	是	0	坐标系
	BufferMode	WORD	是	[0,1]	模式选择详见【8.9.5 模式选择说明】
	TransitionMode	WORD	是	[0,2]	过渡模式详见【8.9.4 过渡模式说明】
	TransitionParameter	LREAL	是	[0,999999999.999999]	过渡速度详见【8.9.4 过渡模式说明】
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志

	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块将轴组移动到特定坐标系统中规定的绝对位置，而不关心是如何到达目标位置；
- 该轴组中的每个轴的性能，例如速度、加速度、减速度、加加速度在移动时不发生改变。

注意事项

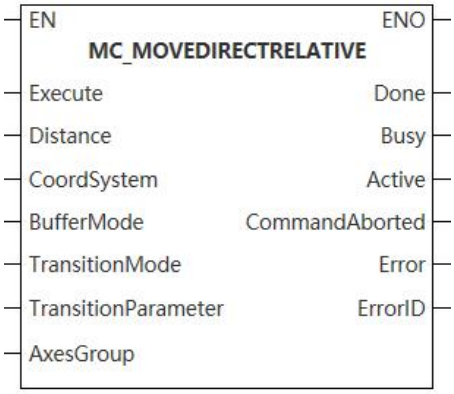
- 轴组使能后方可运动；
- 所有速度均为安全速度、加速度，以能达到的最大速度、加速度为准，（受 FB 内设置的规划速度、轴内限制速度、加速度影响）若达不到相应进行降低；
- 直线运动允许被打断；
- 该指令运行速度、加速度与轴组关联的各轴中的最小值关联；
- 加加速度仅在 S 型曲线生效。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

8、MC_MOVEDIRECTRELATIVE

MC_MOVEDIRECTRELATIVE



该功能块是快速定位相对位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVEDIRECTRELATIVE	Execute	●	●	●	●	●																				
	Distance																									
	CoordSystem																									
	BufferMode																									
	TransitionMode																									
	TransitionParameter																									
	AxesGroup																									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	Active		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_MOVEDIRECTRELATIVE	Execute	BOOL	否	TRUE/FALSE	使能
	Distance	LREAL 数组	否	-	相对位置
	CoordSystem	WORD	是	0	坐标系
	BufferMode	WORD	是	[0,1]	模式选择详见【8.9.5 模式选择说明】
	TransitionMode	WORD	是	[0,2]	过渡模式详见【8.9.4 过渡模式说明】
	TransitionParameter	LREAL	是	[0,999999999.999999]	过渡速度详见【8.9.4 过渡模式说明】
	AxesGroup	WORD	否	[0,65535]	轴组号
	Done	BOOL	是	TRUE/FALSE	完成标志

福州富昌维控电子科技有限公司

	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】）

功能

- 该功能块将轴组移动到特定坐标系统中规定的相对位置，而不关心是如何到达目标位置；
- 该轴组中的每个轴的性能，例如速度、加速度、减速度、加加速度在移动时不发生改变。

注意事项

- 轴组使能后方可运动；
- 所有速度均为安全速度、加速度，以能达到的最大速度、加速度为准，（受 FB 内设置的规划速度、轴内限制速度、加速度影响）若达不到相应进行降低；
- 直线运动允许被打断；
- 该指令运行速度、加速度与轴组关联的各轴中的最小值关联；
- 加加速度仅在 S 型曲线生效。

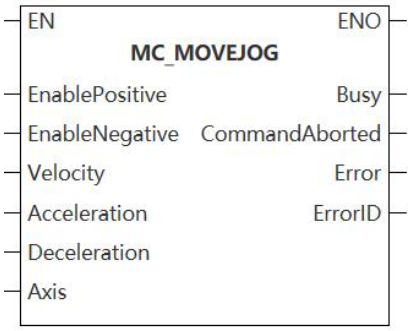
错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

8.7 其他特殊 FB

1、MC_ MOVEJOG

MC_ MOVEJOG



该功能块使被控轴以输入的目标速度来执行微动运动。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVEJOG	EnablePositive	●	●	●	●	●																				
	EnableNegative	●	●	●	●	●																				
	Velocity																									
	Acceleration																									
	Deceleration																									
	Axis																									
	Busy		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_MOVEJOG	EnablePositive	BOOL	否	TRUE/FALSE	正向运动
	EnableNegative	BOOL	否	TRUE/FALSE	反向运动
	Velocity	LREAL	否	(0,999999999.999999]	速度
	Acceleration	LREAL	否	(0,999999999.999999]	加速度
	Deceleration	LREAL	否	(0,999999999.999999]	减速度
	Axis	WORD、AXIS_INFO_T	否	[0,65535]	轴号
	Busy	BOOL	是	TRUE/FALSE	忙标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码（请参见【8.10 错误码对照表】

功能

- 当输入 EnablePositive 信号上升沿时，使被控轴根据输入的 Velocity 执行正向运动；
- 当输入 EnableNegative 信号上升沿时，使被控轴根据输入的 Velocity 执行反向运动；

福州富昌维控电子科技有限公司

- 在正向运动过程中，将 EnablePositive 改为 FALSE，会进行减速运动，直至轴停止。在减速过程中如果将 EnablePositive 改为 TRUE，轴会继续运动直至达到指定速度；在减速过程中如果将 EnableNegative 改为 TRUE，则会先减速到 0，然后再反向运动，直至达到指定速度。

注意事项

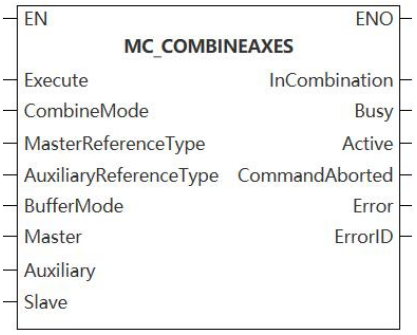
- 当轴不存在时，该功能不可用；
- 在 EnablePositive 或 EnableNegative 为 TRUE 时，此时变更 Velocity 也无法改变速度。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

2、MC_ COMBINEAXES

MC_ COMBINEAXES



该功能块将主轴和辅轴的位置相加、相减作为从轴的位置。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_COMBINEAXES	Execute	●	●	●	●	●																					
	CombineMode																										
	MasterReferenceType																										
	AuxiliaryReferenceType																										
	BufferMode																										
	Master																										
	Auxiliary																										
	Slave																										
	InCombination		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_COMBINEAXES	Execute	BOOL	否	TRUE/FALSE	使能
	CombineMode	WORD	是	[0,1]	加减法运算选择。 0：加法 1：减法
	MasterReferenceType	WORD	是	[1,2]	主轴位置类型。 1：反馈位置 2：指令位置
	AuxiliaryReferenceType	WORD	是	[1,2]	辅轴位置类型。 1：反馈位置 2：指令位置

福州富昌维控电子科技有限公司

	BufferMode	WORD	否	0	模式选择
	Master	WORD、AXIS_INFO_T	否	[0,65535]	主轴轴号
	Auxiliary	WORD、AXIS_INFO_T	否	[0,65535]	辅轴轴号
	Slave	WORD、AXIS_INFO_T	否	[0,65535]	从轴轴号
	InCombination	BOOL	是	TRUE/FALSE	加减运算标志
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	有效标志
	CommandAborted	BOOL	是	TRUE/FALSE	中止标志
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码(请参见【8.10 错误码对照表】)

功能

- 在 Execute 上升沿的时候，执行加减运算定位操作。

注意事项

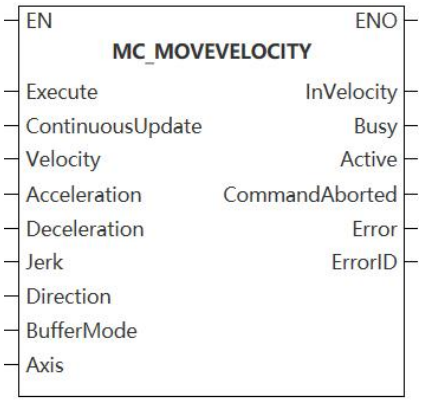
- 无法重启本指令；
- 结束本指令需要使用 MC_Stop 指令；
- 本指定不执行到位检查；
- 使用指令位置时，主轴编号和辅轴编号的设定值必须小于从轴编号；
- 本指令不受主轴和辅轴的异常状态影响，异常状态解除时，从轴继续加减运算定位动作。此外，启动本指令时，从轴发生异常的时，中断本指令，但对主轴和从轴没有影响；
- 对主轴或辅轴执行 MC_SetPosition 指令时，可能出现从轴急速跟踪的情况，非常危险。

错误码

错误码	内容
70000A	轴处于绑定状态
70000B	轴在运行时无法被绑定
800014	主轴和从轴一致
800015	主轴和辅轴一致
800016	辅轴和从轴一致
800017	主轴（辅轴）和从轴轴号非升序

3、MC_ MOVEVELOCITY

MC_ MOVEVELOCITY



使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_MOVEVELOCITY	Execute	●	●	●	●	●																					
	ContinuousUpdate	●	●	●	●	●																					
	Velocity																										
	Acceleration																										
	Deceleration																										
	Jerk																										
	Direction																										
	BufferMode																										
	Axis																										
	InVelocity		●	●	●	●																					
	Busy		●	●	●	●																					
	Active		●	●	●	●																					
	CommandAborted		●	●	●	●																					
	Error		●	●	●	●																					
	ErrorID																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	能否为空	范围	描述
MC_MOVEVELOCITY	Execute	BOOL	否	TRUE/FALSE	使能
	ContinuousUpdate	BOOL	否	TRUE/FALSE	是否持续更新
	Velocity	LREAL	否	[0,999999999.999999]	设定速度
	Acceleration	LREAL	否	[0,999999999.999999]	设定加速度
	Deceleration	LREAL	否	[0,999999999.999999]	设定减速度
	Jerk	LREAL	是	[0,999999999.999999]	设定加加速度，仅在 S 型生效
	Direction	WORD	否	[1]	方向标志位 1:正方向

					2：最短路径 3：负方向 4：当前方向
	BufferMode	WORD	是	[0]	缓冲模式
	Axis	WORD、 AXIS_INFO_T	否	[0,65535]	轴号、轴内置变量
	InVelocity	BOOL	是	TRUE/FALSE	达到设定速度标志位
	Busy	BOOL	是	TRUE/FALSE	忙标志
	Active	BOOL	是	TRUE/FALSE	活动标志
	CommandAborted	BOOL	是	TRUE/FALSE	打断标志位
	Error	BOOL	是	TRUE/FALSE	是否有错误产生
	ErrorID	DWORD	是	-	错误码(请参见【8.10 错误码对照表】)

功能

- MC_MoveVelocity 功能块用于继续前一功能块的运动，并根据功能块规定的速度运动。输入的加速度、减速度和加加速度等数值是监测被控轴的相应数值是否在输入数值范围内；
- 由于没有位置或距离的输入信号，因此，该功能块将一直运动，直到其他功能块发送新的命令来中止该功能块的运动；
- 当被控轴的速度达到输入参数 Velocity 设定值时，输出参数 InVelocity 置位。一旦输入 Execute 信号检测到下降沿或其他功能块使本功能块中止，则输出参数 InVelocity 复位；
- MC_MoveVelocity 功能块应在被控轴处于停止运动状态或者单轴运行状态。该功能块触发后，被控轴转换到连续运动 Continuous Motion 状态。如果被控轴不在上述状态，则该功能块不被执行，其 Error 输出置位；
- MC_MoveVelocity 功能块可多次重复触发。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

8.8 SDO 指令

1、MC_READSDO 读取 SDO 数据

MC_READSDO

该指令用于读取伺服对象字典的值。

-[MC_READSDO （S1）（S2）（S3）（D1）（S4）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)
（S1）	对象字典索引	-	无符号 BIN16	ANY16_U
（S2）	对象字典子索引	-	无符号 BIN16	ANY16_U
（S3）	长度（BYTE 单位）	-	无符号 BIN16	ANY16_U
（D1）	读取到的软元件	-	根据 S3 定义大小	-
（S4）	指定伺服站号	-	有符号 BIN16	ANY16_S

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READSDO	参数 1																	●	●		●	●	●	●			
	参数 2																	●	●		●	●	●	●			
	参数 3																	●	●		●	●	●	●			
	参数 4																	●	●		●	●					
	参数 5																	●	●		●	●	●	●			

功能

- 根据伺服站号（S4）、对象字典索引（S1）、对象字典子索引（S2）、长度（S3）读取值到设定的软元件（D1）。

注意事项

- 如 PDO 中有设置相应的对象字典，可能会出现 SDO 无法读取的情况。

2、MC_WRITESDO 写入 SDO 数据

MC_WRITESDO

该指令用于写入伺服对象字典的值。

-[MC_WRITESDO （S1）（S2）（S3）（S4）（S5）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)
（S1）	对象字典索引	-	无符号 BIN16	ANY16_U
（S2）	对象字典子索引	-	无符号 BIN16	ANY16_U
（S3）	长度（BYTE 单位）	-	无符号 BIN16	ANY16_U
（S4）	需要写入的值	-	根据 S3 定义大小	-
（S5）	指定伺服站号	-	有符号 BIN16	ANY16_S

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MC_READSDO	参数 1																	●	●		●	●	●	●			
	参数 2																	●	●		●	●	●	●			
	参数 3																	●	●		●	●	●	●			
	参数 4																	●	●		●	●					
	参数 5																	●	●		●	●	●	●			

功能

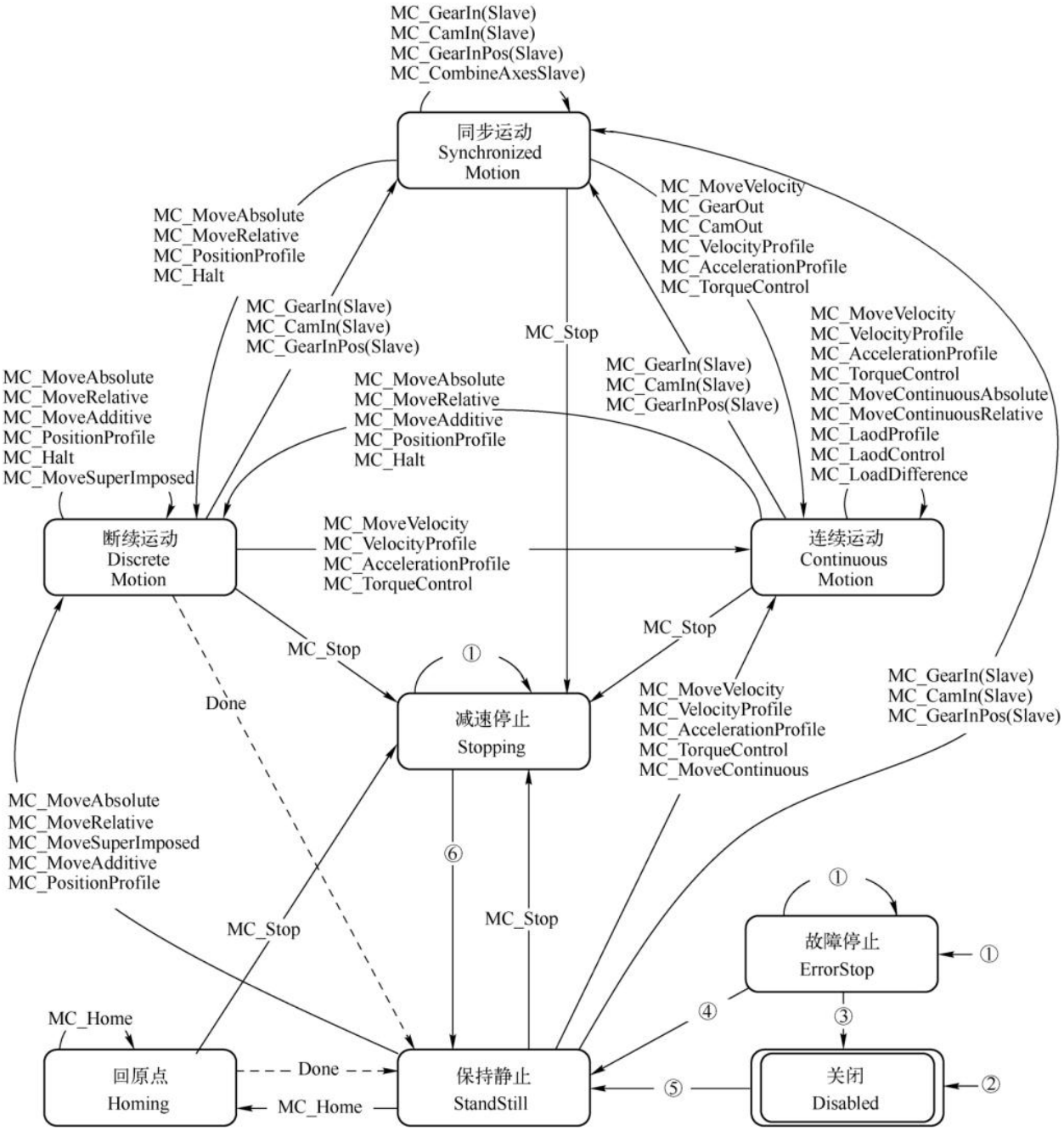
- 将软元件（S4）的值写入给：伺服站号(S5)、对象字典索引（S1）、对象字典子索引（S2）、长度（S3）的位置中。

注意事项

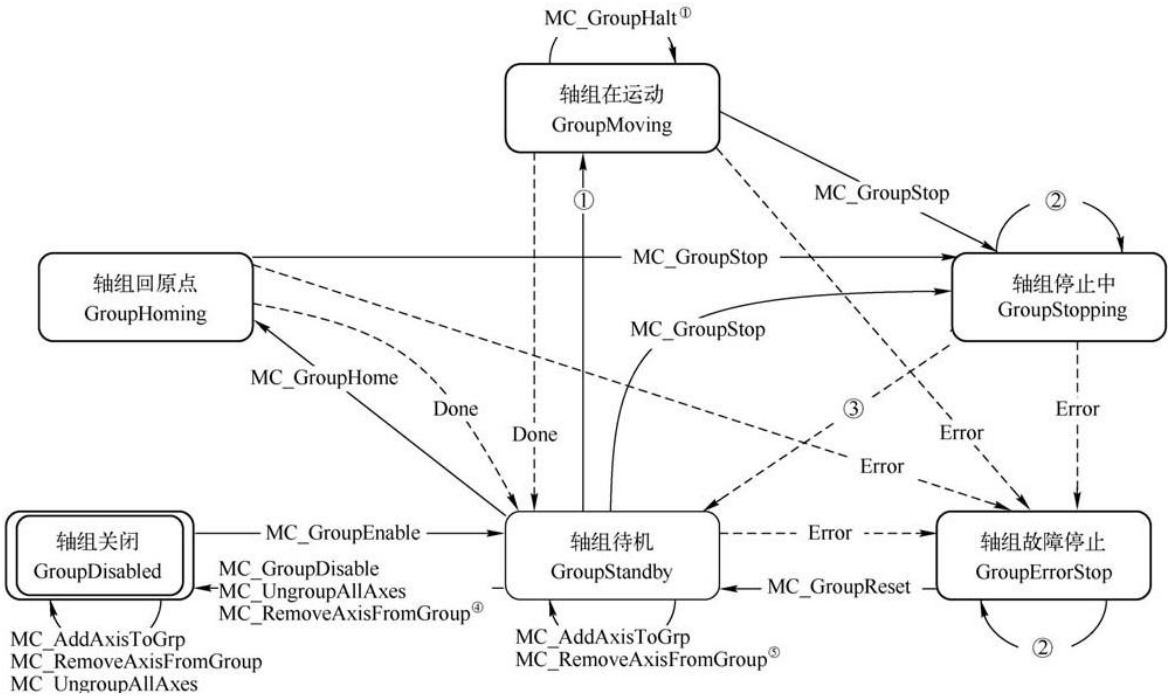
- 如 PDO 中有设置相应的对象字典，可能会出现 SDO 无法写入的情况。

8.9 其他

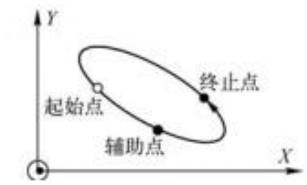
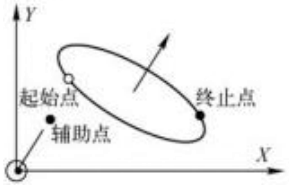
8.9.1 单轴运动控制状态图



8.9.2 轴组运动控制状态图



8.9.3 圆弧模式说明

模式	边界(CircMode = 0)	圆心(CircMode = 1)	半径(CircMode = 2)
功能	把起点、辅助点、终止点组成的圆弧作为运动的轨迹圆	把辅助点定义为运动轨迹的圆心，因此，辅助点应在起点和终止点连线的中垂线上	根据右手螺旋定则，定义终止点和与圆平面垂直的矢量，矢量的长度对应于圆半径。 矢量的方向决定顺逆圆的方向。 圆弧的优劣弧由PathChoice决定。
说明			
辅助点（AuxPoint）	圆弧途径点	圆弧圆心点	空间圆弧所在平面的法向量
路径选择(PathChoice)	无效	顺逆圆 1 逆时针模式 0 顺时针模式	优劣弧 1 劣弧 0 优弧
特点	单命令时限制角度<2π	单命令限制其角度<2π并且≠π	单指令限制其角度<2π 必须计算垂直圆平面的矢量

8.9.4 过渡模式说明

过渡模式详解：过渡参数是过渡模式的附加参数，根据过渡模式不同生效。

TransitionMode	0（无缓冲模式）	1（最大速度缓冲）	2（指定速度缓冲）
TransitionParameter	无效	无效	指定速度

8.9.5 模式选择说明

序　　号	MC_BUFFER_MODE	描　　述
0	mcAborting	约定模式，立刻起动后一功能块的运行
1	mcBuffered	在当前功能块的运动完成后起动后一功能块
2	mcBlendingLow	后一功能块的速度根据前后两个功能块的低速混成
3	mcBlendingPrevious	后一功能块的速度根据前后两个功能块的前一功能快的速度混成
4	mcBlendingNext	后一功能块的速度根据前后两个功能块的后一功能快的速度混成
5	mcBlendingHigh	后一功能块的速度根据前后两个功能块的高速混成

注意

- 需注意，如果正在运动的功能块被另一个功能块中断，会发生由于减速度的限制使制动距离不够的错误；
- 注意在下述解释的所有时序图中，部分状态字反馈的时序图是有误的。例如图 8-2 中，start 信号并非触发后系统自动置位，而是用户控制。功能块将在上升沿时触发执行逻辑。同样的，Done 信号也是，程序并不会自动置位，当功能块结束时，Done 会一直被置位，直到 Execute 的下降沿产生！

(1) 中止模式（mcAborting）

它是没有缓冲的约定模式。当应用轴的状态转换到“ErrorStop”时，所有有缓冲模式的运动控制功能块会进入“中止”模式。这时，在中止模式的功能块输出“Error”被置位。其后续的命令被拒绝，因输出 Error 被置位，因此，Active 被复位。

下图说明前后两个功能块实例在中止模式的功能块连接。

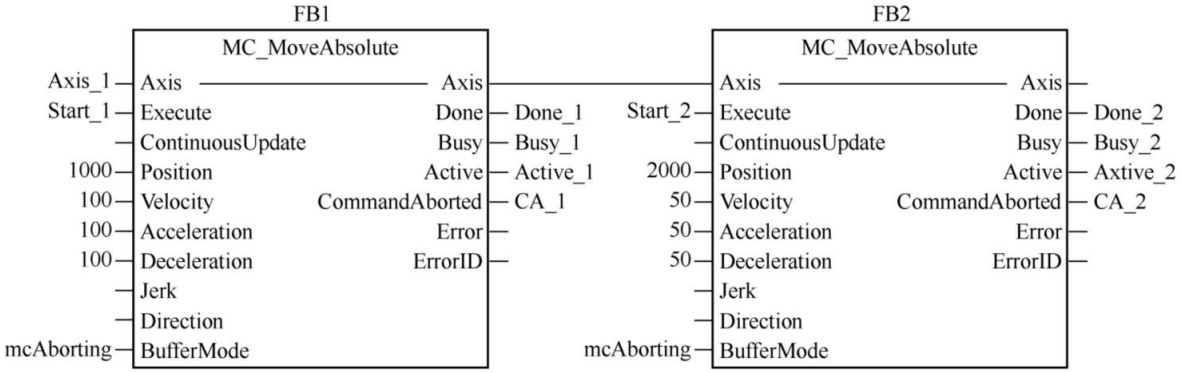


图 8-1

如图 8-1 中，前后两个运动控制功能块都是 MC_MoveAbsolute，且都用中止模式 mcAborting。

下图 8-2 显示功能块实例 FB1 和 FB2 的时序图。图 8-2（a）是两个绝对位置运动功能块在正常工况下的时序图，即 FB1 控制的运动完成后，进行 FB2 控制的运动。

Start_1=TRUE 上升沿时，功能块实例 FB1 激活，同时，输出 Busy_1 和 Active_1 被置位，轴的运动过程是：先根据加速度 100u/s²，使其速度增加，达到 100u/s，经恒速后，再以减速度 100u/s² 减速到 0，并使其位置达到 1000u。这时，输出 Done_1 被置位，而 Busy_1 和 Active_1 复位。轴位置在 1000u。

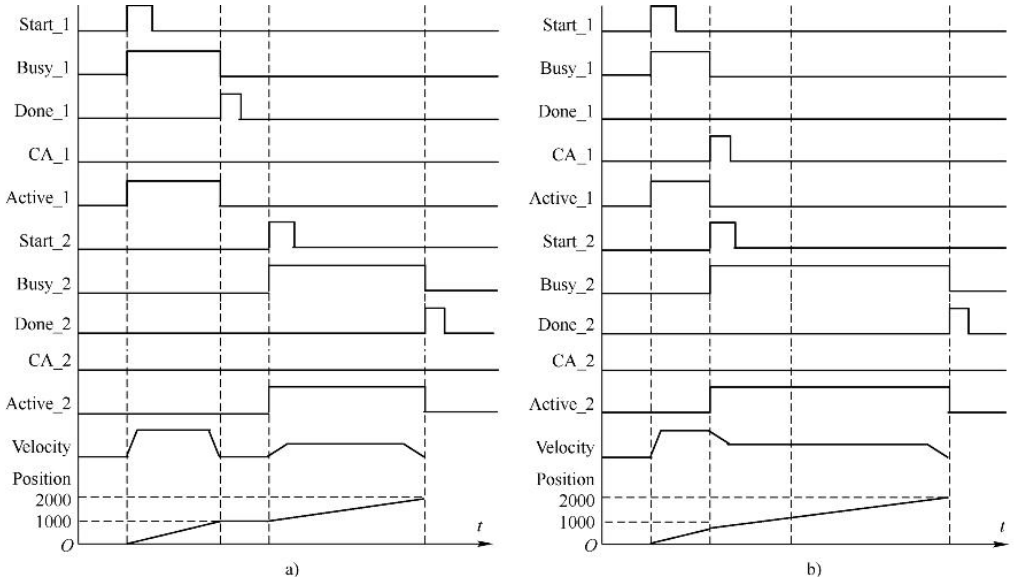


图 8-2 正常工况和中止模式的时序图

等到第二个功能块实例 FB2 在 Start_2=TRUE 上升沿时，该功能块实例 FB2 被激活，输出 Busy_2、Active_2 置位，轴根据加速度 50u/s²，使其速度增加，达到 50u/s，恒速后，再以减速度 50u/s² 减速到 0，并使其位置从 1000u 运动到 2000u。由于绝对位置已经达到，因此，输出 Done_2 被置位，而 Busy_2 和 Active_2 复位。轴位置在 2000u。

图 8-2（b）是两个绝对位置运动功能块在中止模式时的时序图。在 Start_1=TRUE 上升沿时，功能块实例 FB1 激活，同时，输出 Busy_1 和 Active_1 被置位，轴的运动先根据加速度 100u/s²，使其速度升高，但因在该过程中，功能块实例 FB2 的输入 Start_2 被置位，它使前一功能块实例 FB1 中止其运动，这时，轴还没有达到所需位置 1000u 而被中断停止。

由于功能块实例 FB2 的激活，其 Busy_2 和 Active_2 被置位,轴根据功能块实例 FB2 的要求，先根据其减速度 50u/s²，将速度降低到输入数值 50u/s，然后，恒速在 50u/s，最后达到位置 2000u。可见，后一功能块的激活使前一功能块中止其运动，中止命令立刻影响轴的运动。这种缓冲模式称为中止模式。

(2) 已缓冲模式（mcBuffered）

后一功能块的命令被存储在缓冲区域，只有前一功能块的运动完成后，缓冲区的命令才被执行。因此，这种模式称为已缓冲模式。

后一功能块的 BufferMode 输入设置为 mcBuffered 已缓冲模式。图 8-3 是两个功能块连接的设置。

已缓冲模式的时序图如图 8-4 所示。图中，当前一功能块实例 FB1 的输入 Start_1=TRUE 上升沿时，激活 FB1。该功能块实例 FB1 控制轴移动时，后一功能块实例 FB2 的 Start_2 被置位，由于 FB2 的缓冲模式被设置为已缓冲模式，因此，FB2 的激活并不中断 FB1 对轴的控制，虽然，FB2 的输出 Busy_2 被置位，但功能块并没有被激活，即输出 Active_2 仍为 FALSE。因此，FB1 仍继续控制该轴的运动，直到轴到达设定的位置 1000u。然后立刻激活 FB2 工作，而轴的速度没有延迟，直接按加速度 50u/s²，使速度增加，达到 50u/s，然后恒速在 50u/s，最后达到位置 2000u。

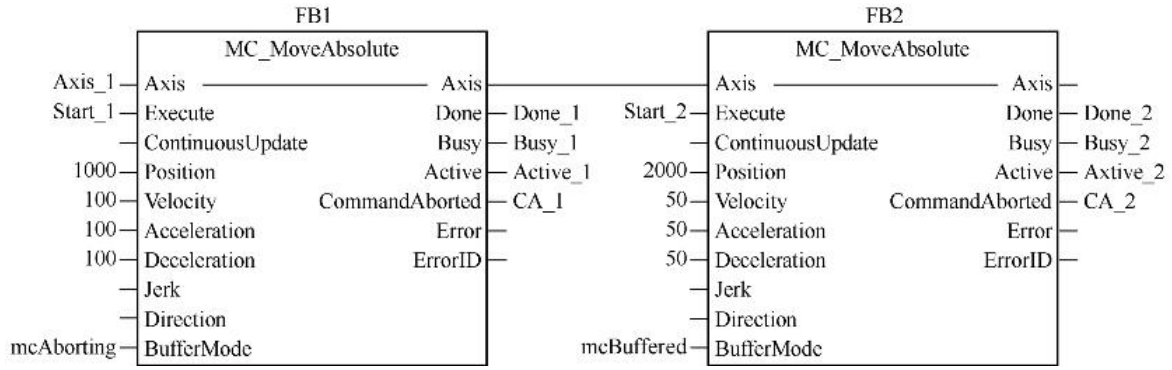


图 8-3 已缓冲模式的连接

从图可见，后一功能块的缓冲模式被设置为已缓冲模式，这表示后一功能块不能中断前一功能块的运动，只有当前一

福州富昌维控电子科技有限公司

功能块完成其任务，Done_1 被置位后，立刻进入后一功能块的工作。

已缓冲模式表示后一功能块的运动命令已经被缓冲区存储，它不会影响前一运动控制功能块对轴的控制和操作。它将等待被执行的功能块输出 Done 或 InXxx 为 TRUE 时，才执行后续命令。

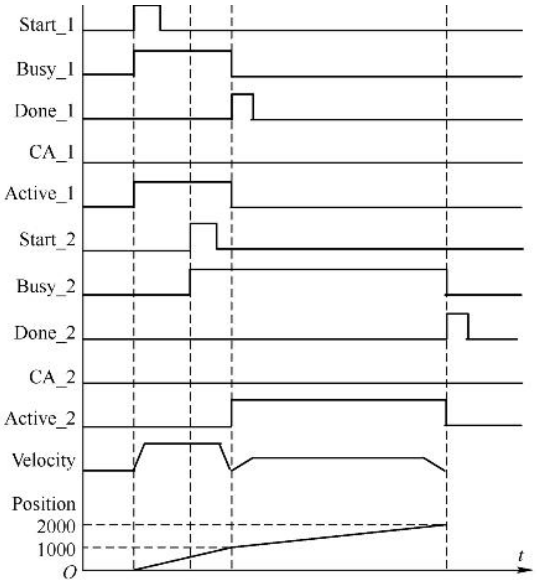


图 8-4 已缓冲模式的时序图

(3) 混成模式

前后两个功能块的混成模式有 4 种选项。混成模式表示前后功能块速度的搅和。这种情况下，前一功能块中断其运动，后续的运动由后一功能块控制，但其初始的速度称为混成速度或搅和速度。

搅和速度既可根据前后两个功能块速度的低者，也可根据两者的高者，或者直接规定采用前一功能块的速度或后一功能块的速度。由此，混成模式分为 4 种。

1）混成低速模式（mcBlendingLow）。

即混成速度是前后两个功能块设定速度的低者。这种混成模式有随机性，前一功能块设定速度低于后一功能块的设定速度，则混成速度选用前一功能块的速度。反之，前一功能块设定速度高于后一功能块的设定速度，则混成速度选用后一功能块的设定低速。

当后一个功能块的缓冲模式设置为混成低速模式时，表示在前一功能块还未完成时后一功能块不会中断前一功能块的工作；当前一功能块速度降到前后两个功能块的低速时，后一功能块才能继续控制轴的运动。

图 8-5 是三个运动功能块的连接图，后两个功能块都被设置为 mcBlendingLow。

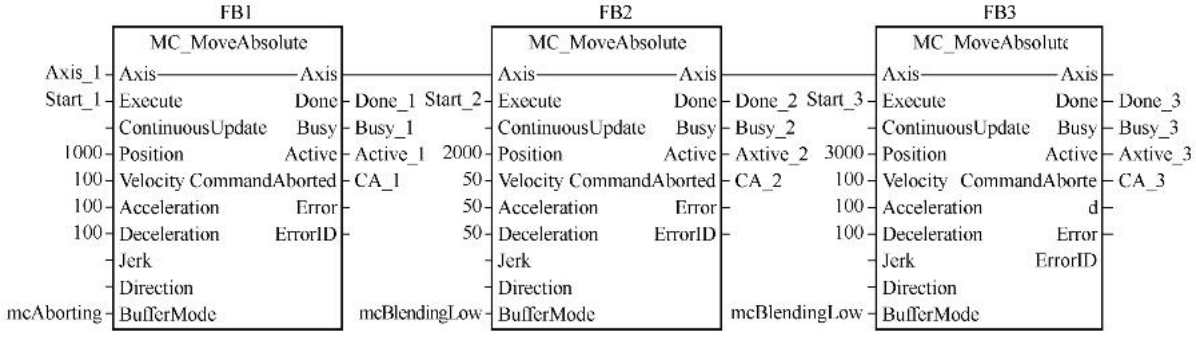


图 8-5 混成低速模式的连接

图 8-6 是混成低速模式的时序图，为说明混成低速，图中采用两个后续的功能块实例 FB2 和 FB3。图中，FB1 在 Strat_1 上升沿置位后，被激活，其 Busy_1 置位，在 FB1 控制轴运动时，FB2 实例的 Start_2 也在上升沿被置位，由于 FB1 还未完成轴的运动，而 FB2 的 Buffer-Mode 被设置为 mcBlendingLow。因此，FB1 按 FB1 和 FB2 混成的最低速，即采用 FB2 的设置速度 50u/s。这表示在 FB1 后续运动中，按 FB2 的减速度将轴的速度降低到 50u/s。当轴速度降低到混成的低速后，FB1 控制轴运动到设置位置 1000u，FB1 的输出 Done_1 置位，表示其控制的运动结束。FB2 立刻控制轴以 50u/s 的恒速运动，轴位置向规定的 2000u 前进。这时，如果功能块实例 FB3 的 Start_3 被置位，由于 FB2 还未到达其位置 2000u，而 FB3

福州富昌维控电子科技有限公司

的 Buffer-Mode 被设置为 mcBlendingLow，因此，在 FB3 的 Start_3 被置位时，该运动轴将根据 FB2 和 FB3 设置的速度，选用两者的低速对轴进行控制，直到轴的位置到达设置位置 2000u，这时，FB2 的激活输出 Active_2 被复位。之后，轴在 FB3 功能块实例的控制下，将速度升高到 100u/s，然后恒速，最后降低速度到 0u/s，这时，轴的位置达到设置位置 3000u。

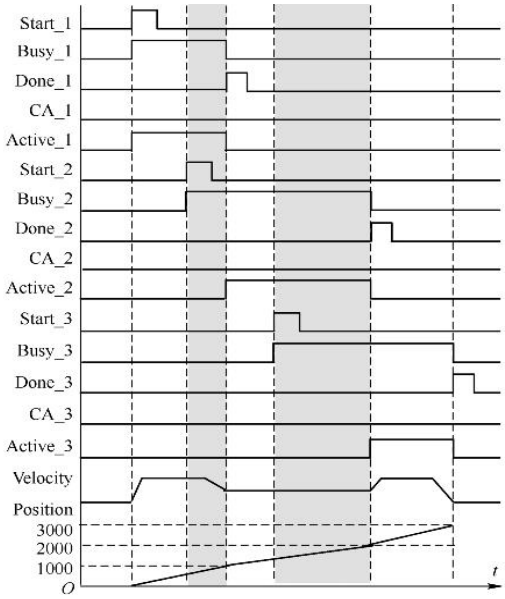


图 8-6 混成低速模式的时序图

混成模式表示由于前后功能块设置的速度不同，当后一功能块的 Execute 上升沿输入时被置位，系统将根据后一功能块设置的 BufferMode 参数，确定这时应采用什么速度对轴进行控制。上述的混成低速模式是根据两者设置速度的低者。前一功能块将在混成速度的控制下完成后续运动，直到达到设置的位置，并由后续功能块继续控制该轴。混成前模块的速度模式是混成速度选用前一功能块设置的速度。

混成前模块的速度模式（mcBlendingPrevious）。如图 8-5 所示，如果 FB2 和 FB3 的 BufferMode 参数设置为 mcBlendingPrevious，则混成速度直接采用前一功能块的速度。图 8-7 是混成前模块的速度模式的时序图。

分析时序图时，需注意混成速度的改变。如图 8-7 所示混成前模块速度模式时序图中，FB1 在缓冲模式时，根据前模块，即 FB1 的速度 100u/s 运动，直到达到设定位置 1000u。而 FB2 在缓冲模式时，根据前模块 FB2 的速度 50u/s 运动，直到达到设定位置 2000u。因此，FB1 输出 Active 复位时的轴速度是 100u/s，FB2 输出 Active 复位时的轴速度是 50u/s。

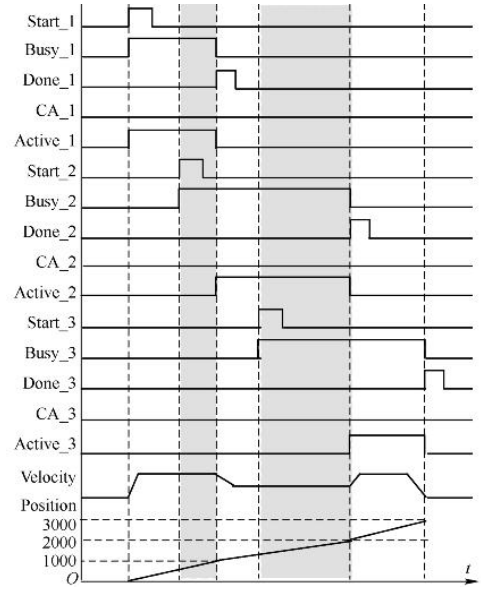


图 8-7 混成前模块速度模式的时序图

混成后模块的速度模式(mcBlendingNext)。如图 8-5 所示，如果 FB2 和 FB3 的 BufferMode 参数设置为 mcBlendingNext，则混成速度直接采用后一功能块的速度。图 8-8 是混成后模块的速度模式的时序图。

如图 8-8 所示混成后模块速度模式的时序图中，FB1 在缓冲模式时，根据后模块，即 FB2 的速度 50u/s 运动，直到达

福州富昌维控电子科技有限公司

到设定位置 1000u。而 FB2 在缓冲模式时，根据后模块 FB3 的速度 100u/s 运动，直到设定位置 2000u 达到。因此，FB1 输出 Active 复位时的轴速度是 50u/s，FB2 输出 Active 复位时的轴速度是 100u/s。

而在图 8-6 所示混成低速模式的时序图中，FB1 在缓冲模式时，根据混成低速缓冲，即用 FB2 的速度 50u/s 运动，直到达到设定位置 1000u。而 FB2 在缓冲模式时，根据混成低速缓冲，用 FB2 的速度 50u/s 运动，直到设定位置 2000u 达到。因此，FB1 输出 Active 复位时的轴速度是 50u/s，FB2 输出 Active 复位时的轴速度 50u/s。

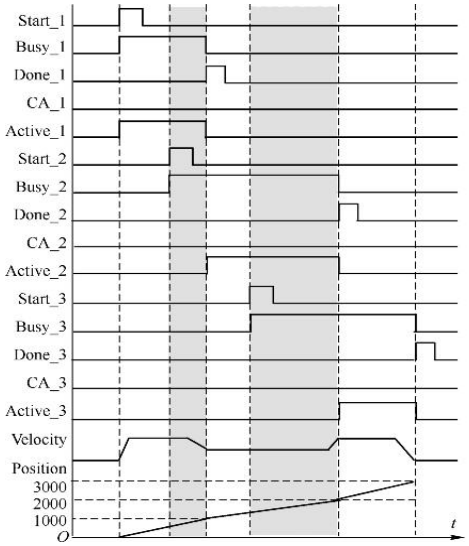


图 8-8 混成后模块速度模式的时序图

2) 混成高速模式（mcBlendingHigh）。

图 8-5 中，如果 FB2 和 FB3 的 BufferMode 参数设置为 mcBlendingHigh，则混成速度直接采用前后两个功能块设定速度的高者。

图 8-9 的混成高速模式时序图中，前一功能块实例 FB1 在缓冲模式时，根据混成高速缓冲，即用 FB1 的速度 100u/s 运动，直到设定位置 1000u 达到。而 FB2 在缓冲模式时，根据混成高速缓冲，用 FB3 的速度 100u/s 运动，直到设定位置 2000u 达到。因此，FB1 输出 Active 复位时的轴速度是 100u/s，然后再减速到 50u/s。FB2 输出 Active 复位时的轴速度 100u/s。

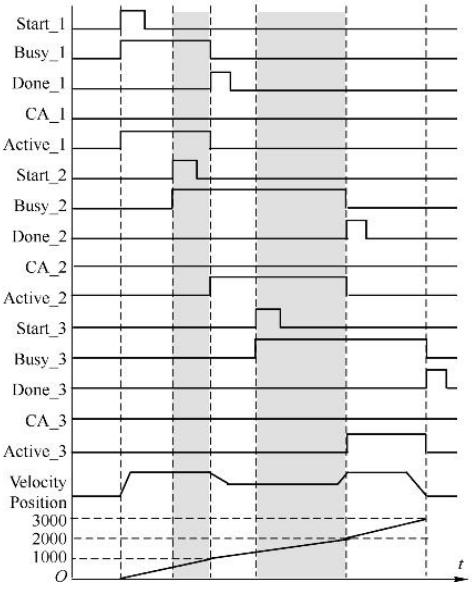


图 8-9 混成高速模式的时序图

电子凸轮中几种模式说明

StartMode 参数与 CAMTABLESELECT 指令中的 MasterAbsolute/SlaveAbsolute 参数共同决定了主/从轴的运动模式，主轴模式仅由 MasterAbsolute 决定，不受 StartMode 中的值的影响。从轴模式表现形式如下：

StartMode	SlaveAbsolute	从轴模式
绝对	相对	相对
绝对	绝对	绝对
相对	相对	相对
相对	绝对	相对
追及	相对	相对追及
追及	绝对	绝对追及

主从轴绝对/相对模式在执行 CAMIN 指令时的结果。

主轴模式	从轴模式	结果
相对	相对	CAMIN 执行后，从轴位置跃迁至设置的偏置位置；主轴运行后，从轴根据凸轮表起始点位进行相对运动
	绝对	CAMIN 执行后，从轴位置跃迁到凸轮表起始位置（即 0）；主轴运行后，从轴根据凸轮表对应点位进行运动
	相对追及	CAMIN 执行后，从轴位置追及至设置的偏置位置；主轴运行后，从轴根据凸轮表对应点位进行相对运动
	绝对追及	CAMIN 执行后，从轴位置追及到凸轮表起始位置（即 0）；主轴运行后，从轴根据凸轮表对应点位进行运动
绝对	相对	CAMIN 执行后，从轴位置跃迁至设置的偏置位置；主轴运行后，从轴根据凸轮表对应点位进行相对运动
	绝对	CAMIN 执行后，从轴位置跃迁到凸轮表主轴当前对应的从轴位置（例：主轴当前位置为 100，凸轮表中主轴点 100 对应的从轴点为 200，CAMIN 执行后从轴跃迁）；主轴运行后，从轴根据凸轮表对应点位进行相对运动
	相对追及	CAMIN 执行后，从轴位置跃迁至设置的偏置位置；主轴运行后，从轴根据凸轮表对应点位进行相对运动
	绝对追及	CAMIN 执行后，从轴位置追及到凸轮表主轴当前对应的从轴位置（例：主轴当前位置为 100，凸轮表中主轴点 100 对应的从轴点为 200，CAMIN 执行后从轴跃迁）；主轴运行后，从轴根据凸轮表对应点位进行相对运动

注意事项

- 当主轴为绝对模式时，如果主轴的当前位置不在凸轮表主轴范围内，则自动作周期化处理，例：主轴当前位置为 110，凸轮表中主轴的位置为 0~100，则 CAMIN 执行后默认主轴位置为 10（实际的主轴位置不发生变化）；
- 主从轴倍率、主从轴偏移参数在执行 CAMIN 时生效，不支持过程中修改，不适合的参数会导致从轴位置跃迁。主从中的位置关系为（其中 CAM（）表示主轴在凸轮表上对应的从轴位置）；

  从轴位置 = 从轴倍率*CAM（（主轴位置*主轴倍率+主轴偏移）） + 从轴偏移
- 主从轴倍率不能为 0。

8.10 错误码对照表

错误码/ErrorCode (16 进制/HEX)	释义	错误码定义/ErrorCode Define	Interpretation
正常			
0x00	没有错误	MC_ERROR_ID_SUCCESS	Success
运行时错误			
0x00200000	计算曲线错误	MC_ERROR_ID_SCURVE_NOTFEASIBLE	Not feasible to compute Scurve
0x00200001	未达到最大速度	MC_ERROR_ID_SCURVE_MAXVEL_NOTREACHED	Maximum velocity is not reached
0x00200002	未达到最大加速度	MC_ERROR_ID_SCURVE_MAXACC_NOTREACHED	Maximum acceleration is not reached
0x00200003	无法找到最大速度	MC_ERROR_ID_SCURVE_FINDMAXACC_FAILED	Fail to find max velocity
0x00200004	无效的曲线规划约束	MC_ERROR_ID_SCURVE_INVALIDINPUT	Maximum acceleration is not reached
PlcOpen 状态机错误			
0x00300000	无效的状态转换通用错误	MC_ERROR_ID_AXIS_STATE_VIOLATION	Invalid constraints for Scurve planning
0x00300001	用户尝试在轴有错误时去使能伺服	MC_ERROR_ID_POWER_ON_OFF_FROM_ERROR_STOP	Try to power on at ErrorStop
0x00300002	在轴处于停止状态的错误状态转换	MC_ERROR_ID_INVALID_STATE_FROM_STOPPING	Invalid state transition at Stopping
0x00300003	在轴处于错误停止状态的错误状态转换	MC_ERROR_ID_INVALID_STATE_FROM_ERROR_STOP	Invalid state transition at ErrorStop
0x00300004	在轴处于未激活状态时的错误状态转换	MC_ERROR_ID_INVALID_STATE_FROM_DISABLED	Invalid state transition at Disabled
0x00300005	在轴处于激活状态的错误状态转换	MC_ERROR_ID_INVALID_STATE_FROM_POWERED	Invalid state transition at Powered
0x00300006	在轴处于同步过程中的错误状态转换	MC_ERROR_ID_INVALID_STATE_FROM_SYNC	Invalid state transition at Sync
0x00300007	在轴处于未回零的错误状态转换	MC_ERROR_ID_INVALID_STATE_NOT_HOME	Invalid state transition when not home
0x00300008	无法将轴切换到未激活状态	MC_ERROR_ID_INVALID_STATE_TO_DISABLED	Invalid state transition to Disabled
0x00300009	无法将轴切换错误停机状态	MC_ERROR_ID_INVALID_STATE_TO_ERRORSTOP	Invalid state transition to ErrorStop
0x0030000A	无法将轴切换到静止状态	MC_ERROR_ID_INVALID_STATE_TO_STANDSTILL	Invalid state transition to StandStill
0x0030000B	无法将轴切换到回零中状态	MC_ERROR_ID_INVALID_STATE_TO_HOMING	Invalid state transition to Homing
0x0030000C	无法将轴切换到正在停止状态	MC_ERROR_ID_INVALID_STATE_TO_STOPPING	Invalid state transition to Stoppping
0x0030000D	无法将轴切换到断续运动状态	MC_ERROR_ID_INVALID_STATE_TO_DISCRETE	Invalid state transition to DiscreteMotion
0x0030000E	无法将轴切换到同步运动状态	MC_ERROR_ID_INVALID_STATE_TO_SYNC	Invalid state transition to SynchronizedMotion
0x0030000F	无法将轴切换到连续运动状态	MC_ERROR_ID_INVALID_STATE_TO_CONTINUOUS	Invalid state transition to ContinuousMotion

错误码/ErrorCode (16 进制/HEX)	释义	错误码定义/ErrorCode Define	Interpretation
由于用户/系统限制导致错误			
0x00400000	速度、加速度或位置超限	MC_ERROR_ID_MOTION_LIMIT_ERROR	Velocity,Acceleration or Position over Limit
0x00400001	不允许正向运动	MC_ERROR_ID_INVALID_DIRECTION_POSITIVE	Velocity in forbiden direction
0x00400002	不允许负向运动	MC_ERROR_ID_INVALID_DIRECTION_NEGATIVE	Velocity in forbiden direction
0x00400003	速度超过限制	MC_ERROR_ID_VELOCITY_OVER_LIMIT	Velocity over limit
0x00400004	加速度超过限制	MC_ERROR_ID_ACCELERATION_OVER_LIMIT	Acceleration over limit
0x00400005	位置超过正方向限制	MC_ERROR_ID_POSITION_OVER_POSITIVE_LIMIT	Position over positive limit
0x00400006	位置超过负方向限制	MC_ERROR_ID_POSITION_OVER_NEGATIVE_LIMIT	Position over negative limit
0x00400007	跟随误差超过限制	MC_ERROR_ID_POSITION_LAG_OVERLIMIT	Position lag over limit
错误停止			
0x00500000	错误停止	MC_ERROR_ID_AXIS_ERROR_STOP	Axis error stopped
伺服上报的错误			
0x00600000	伺服没有错误	MC_ERROR_ID_SERVO_NO_ERROR	Servo has no error
0x00600001	伺服总线初始化错误	MC_ERROR_ID_SERVO_FIELDBUS_INIT_ERROR	Servo fieldbus initializae error
0x00600002	伺服使能错误	MC_ERROR_ID_SERVO_POWER_ERROR	Servo power error
0x00600003	伺服在使能过程中发生错误	MC_ERROR_ID_SERVO_POWERING_ON_ERROR	Servo has error during powering on
0x00600004	伺服在使能后发生错误	MC_ERROR_ID_SERVO_ERROR_WHEN_POWERED_ON	Servo has error after powered on
0x00600005	伺服在取消使能发生错误	MC_ERROR_ID_SERVO_POWERING_OFF_ERROR	Servo has error during powering off
0x00600006	PDO 未配置	MC_ERROR_ID_SERVO_PDO_NOT_MAP	Servo PDO not map
0x00600007	PDO 读取错误	MC_ERROR_ID_SERVO_PDO_READ_ERROR	Servo PDO read error
0x00600008	PDO 写入错误	MC_ERROR_ID_SERVO_PDO_WRITE_ERROR	Servo PDO write error
0x00600009	伺服未上电，无法运行	MC_ERROR_ID_SERVO_POWER_OFF	Servo is in poweroff cant't run
0x0060000A	伺服在回零过程中发生了错误	MC_ERROR_ID_SERVO_ERROR_HOME_STATE	Servo has error state when home
0x0060000B	伺服模式切换错误	MC_ERROR_ID_SERVO_HOME_MODE_SWITCH	Servo mode switch error
0x0060000C	伺服设置回零模式出错	MC_ERROR_ID_SERVO_HOME_CTRLWORD	Servo set operation home mode error
0x0060000D	伺服回零出错	MC_ERROR_ID_SERVO_HOME	Servo home error
0x0060000e	伺服 PDO 访问请求繁忙	MC_ERROR_ID_SERVO_PDO_READ_BUSY	Servo PDO_READ_BUSY
0x0060000f	伺服阀中未配置探针相关 PDO 映射	MC_ERROR_ID_SERVO_TOUCHPROBE_NOTFIND	Servo TouchProbe not fin
0x00600010	伺服参数读错误	MC_ERROR_ID_SERVO_PARAREAD	Servo param read
0x00600011	伺服参数写错误	MC_ERROR_ID_SERVO_PARAWRITE	Servo param write
0x00600012	原点返回过程中超时	MC_LOCAL_PULSE_HOME_TIMEOUT	Timed out during origin return process

错误码/ErrorCode (16 进制/HEX)	释义	错误码定义/ErrorCode Define	Interpretation
0x00600013	运行过程一直未碰到限位（相对运动 9999999 位置）	MC_LOCAL_PULSE_HOME_NOT_TOUCH_LIMIT	The execution process has not encountered the limit
0x006000ff	伺服丢失	MC_ERROR_ID_SERVO_LOST	Servo lost
轴上的错误			
0x00700000	轴上发生了一般错误	MC_ERROR_ID_AXIS_GENERAL_ERROR	Axis's general error
0x00700001	无法将当前功能块添加进执行队列	MC_ERROR_ID_AXIS_ADD_QUEUE_ERROR	Axis's error, can't inser
0x00700002	轴上有错误发生	MC_ERROR_ID_AXIS_IS_UNHEALTHY	Axis is unhealthy
0x00700003	轴指令超过限制	MC_ERROR_ID_AXIS_IS_OVER_LIMIT	Axis's motion cmds is ove
0x00700004	轴在执行下降沿指令时出现错误	MC_ERROR_ID_AXIS_FALLINGEDGE_ERROR	Axis when fallingedge err
0x00700005	轴不存在	MC_ERROR_ID_AXIS_NOTFIND	Axis's not find
0x00700006	轴已经在轴组中	MC_ERROR_ID_AXIS_IS_INGROUP	Axis is in Group
0x00700007	轴在当前状态下不允许设置位置	MC_ERROR_ID_AXIS_CAN_NOT_SETPOS	Axis isn't be setpos forc
0x00700008	轴未使能	MC_ERROR_ID_AXIS_NOT_POWER	Axis isn't power yet
0x00700009	当前轴并未支持绑定的探针号	MC_ERROR_ID_AXIS_NOT_SUPPORTED_TOUCH	
0x00700010	轴已经被结合	MC_ERROR_ID_AXIS_IS_COMBINED	Axis is combined
0x00700011	轴在运行无法连接	MC_ERROR_ID_AXIS_CAN_NOT_COMBINE_WHILE_RUN	Axis can't combine while running
0x0070000C	轴组号未找到	MC_ERROR_ID_AXIS_GROUP_NOTFIND	Axis group not found
通用 FB 错误			
0x00800000	用户功能块执行错误	MC_ERROR_ID_FB_GENERAL_ERROR	User's FB run error
0x00800001	用户功能块读取错误	MC_ERROR_ID_FB_READ_PARAM_ERROR	User's FB read error
0x00800002	用户功能块停止错误	MC_ERROR_ID_FB_STOP_ERROR	User's FB stop error
0x00800003	用户功能块未绑定逻辑轴	MC_ERROR_ID_FB_NOT_BIND_AXIS	User's FB isn't bind the logic axis
0x00800004	用户功能块没有设置访问参数	MC_ERROR_ID_FB_NOT_SET_PARAMETER_ID	User's FB dosen't set parameter number
0x00800005	用户功能块已经结束，无法更新内容	MC_ERROR_ID_FB_NOT_IN_QUEUE	User's FB already done, can't update
0x00800006	用户功能块参数错误	MC_ERROR_ID_FB_PARAMETER_ERROR	User's FB Parameter is error
0x00800007	用户功能块在执行规划时由于参数导致错误	MC_ERROR_ID_FB_PLAN_FAILED	User's FB do replan with parameter failed
0x00800008	用户功能块在运行时发生了绑定轴的切换	MC_ERROR_ID_FB_REPLACE_AXIS_IN_RUN	User's FB Runing Replace Axis
0x00800009	停止运动叠加错误	MC_ERROR_ID_FB_HALTSUPERIMPOSED_ERROR	stop virtual superimposed failed
0x0080000A	用户功能块未绑定轴在轴组上	MC_ERROR_ID_FB_NOT_BIND_AXIS_GROUP	User's FB isn't bind the axis group
0x0080000B	绑定的轴上有错误	MC_ERROR_ID_FB_BIND_AXIS_IS_ERROR	User set Axis is null
0x0080000C	FB 中正在规划的轴有错误	MC_ERROR_ID_FB_MINE_AXIS_IS_ERROR	FB's Axis is null
0x0080000D	FB 没有绑定主轴	MC_ERROR_ID_FB_NOT_BIND_MASTER	FB not bind Master
0x0080000E	FB 没有绑定从轴	MC_ERROR_ID_FB_NOT_BIND_SLAVE	FB not bind Slave

错误码/ErrorCode (16 进制/HEX)	释义	错误码定义/ErrorCode Define	Interpretation
0x0080000F	FB 不能绑定编码器轴	MC_ERROR_ID_FB_CAN_NOT_BIND_ENCODERAXIS	FB can't bind encoder axis
0x00800010	FB 没有绑定探针属性参数	MC_ERROR_ID_FB_NOT_BIND_TRIGGER_INFO	
0x00800011	FB 绑定的探针号不被支持	MC_ERROR_ID_FB_NOT_SUPPORTED_TRINUM	
0x00800012	FB 绑定的参数在运行时不允许修改	MC_ERROR_ID_FB_CAN_NOT_CHANGE_WHILE_RUN	
0x00800013	FB 参数设置错误	MC_ERROR_ID_FB_RANGE_SET_ERROR	FB param range is error, may max value is less than the min value
0x00800014	FB 主从轴一致	MC_ERROR_ID_FB_MASTER_SLAVE_ARE_SAME	FB master as same as slave
0x00800015	FB 主轴辅助轴一致	MC_ERROR_ID_FB_MASTER_AUXILIARY_ARE_SAME	FB master as same as auxiliary
0x00800016	FB 辅助轴和从轴一致	MC_ERROR_ID_FB_AUXILIARY_SLAVE_ARE_SAME	FB auxiliary as same as slave
0x00800017	FB 主从 id 不是升序排列的	MC_ERROR_ID_FB_MASTER_SLAVE_NOT_ASCENDING	FB master and slave id are not in ascending order
通用轴组错误			
0x00900000	轴组通用错误	MC_ERROR_ID_AXIS_GROUP_GENERAL_ERROR	Group general error
0x00900001	轴组功能块无法添加进执行队列	MC_ERROR_ID_AXIS_GROUP_ADD_QUEUE_ERROR	Group fb add error
0x00900002	轴组在移除逻辑轴时发生错误	MC_ERROR_ID_AXIS_GROUP_REMOVE_ERROR	Group remove fb error
0x00900003	轴组在激活过程中发生错误	MC_ERROR_ID_AXIS_GROUP_ENABLE_ERROR	Group enable error
0x00900004	轴组在取消激活过程中发生错误	MC_ERROR_ID_AXIS_GROUP_DISABLE_ERROR	Group disable error
0x00900005	轴组在停止过程中发生错误	MC_ERROR_ID_AXIS_GROUP_STOP_ERROR	Group stop error
0x00900006	轴组想要运行而其中的轴处于被单轴控制的状态	MC_ERROR_ID_AXIS_GROUP_AXIS_INSINGLEMOVE	Group want run, but binded axis is run with single
0x00900007	轴在绑定在轴组中	MC_ERROR_ID_AXIS_GROUP_AXIS_INGROUPMOVE	Group's any axis is error or any error in group
0x00900008	轴组上有任意错误发生	MC_ERROR_ID_AXIS_GROUP_IS_UNHEALTHY	Group can't set position
0x00900009	轴组不允许设置当前位置	MC_ERROR_ID_AXIS_GROUP_CAN_NOT_SETPOT	Group invalid state transition
0x0090000A	轴组想要切换到不被允许的状态	MC_ERROR_ID_AXIS_GROUP_INVALID_STATE_TRANS	Group invalid state transition
0x0090000B	轴组未绑定轴	MC_ERROR_ID_AXIS_GROUP_NOT_BIND_AXIS	Group invalid state transition
0x0090000C	轴组未使能	MC_ERROR_ID_AXIS_GROUP_IS_DISABLED	Group invalid state transition
单轴规划器错误			
0x00A00000	规划器位置为 0	MC_ERROR_ID_PLANNER_DIST_IS_ZERO	Group general error
0x00A00001	规划器速度为 0	MC_ERROR_ID_PLANNER_SPPED_IS_ZERO	Group fb add error
0x00A00002	规划器加速度为 0	MC_ERROR_ID_PLANNER_ACC_DES_IS_ZERO	Group remove fb error
0x00A00003	规划器加加速度为 0	MC_ERROR_ID_PLANNER_JERK_IS_ZERO	Group general error
0x00A00004	规划器同步错误，当前位置与上一次位置不一致	MC_ERROR_ID_PLANNER_CURRENT_DIFF_LAST	Group fb add error
0x00A00005	输入速度超限	MC_ERROR_ID_PLANNER_SPPED_OVER_LIMIT	Planner input speed is over limit

福州富昌维控电子科技有限公司

错误码/ErrorCode (16 进制/HEX)	释义	错误码定义/ErrorCode Define	Interpretation
0x00A00006	输入加减速度超限	MC_ERROR_ID_PLANNER_ACC_DES_OVER_LIMIT	Planner input acc and dec is over limit
0x00A00007	输入加加速度超限	MC_ERROR_ID_PLANNER_JERK_OVER_LIMIT	Planner input Jerk is over limit
0x00AFFFFF	规划器单轴计算错误	MC_ERROR_ID_PLANNER_SG_CULCULATE	Group remove fb error
轴组规划器错误			
0x00B00000	规划器空间曲线，首尾一致	MC_ERROR_ID_PLANNER_SP_SAME_START_END	Group general error
0x00B00001	规划器空间曲线，并不在同一个平面	MC_ERROR_ID_PLANNER_SP_NOT_IN_SAME_PLANE	Group fb add error
0x00BFFFFF	规划器空间曲线计算错误	MC_ERROR_ID_PLANNER_SP_CULCULATE	Group remove fb error

9 高速计数器

9.1 高速计数器(编码器轴)使用说明

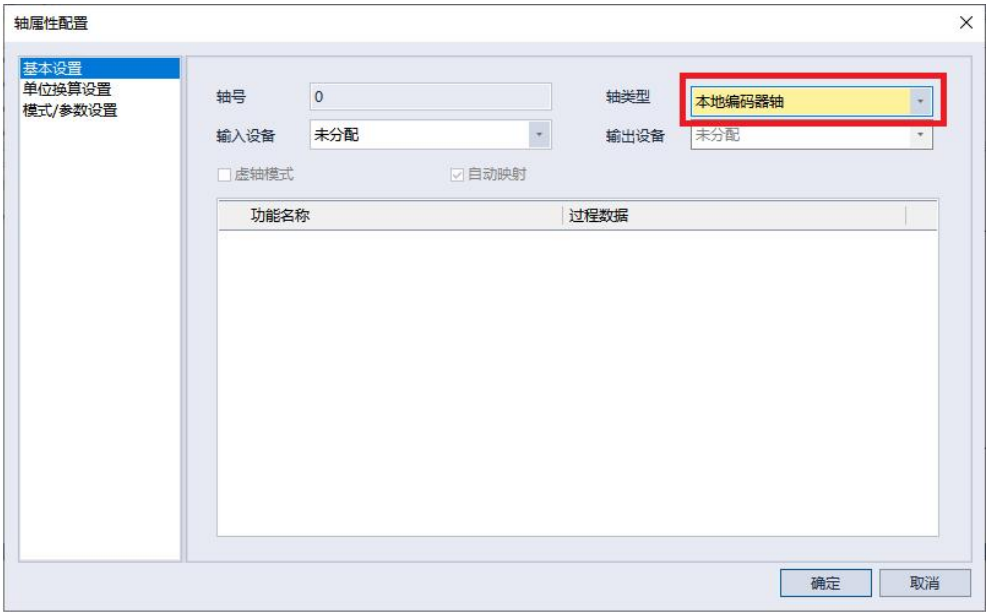
9.1.1 高速计数器轴简介

高速计数器功能在 LX6 系列 PLC 中以编码器轴的形式实现管理应用，后续统称编码器轴，支持 4 轴 32 位高速计数器，可实现 AB 相 1/2/4 倍频、脉冲+方向、单相计数。计数信号可有外部脉冲输入或内部 100us/1ms/10ms 时钟计数，支持探针及高速比较输出功能。

9.1.2 创建编码器轴

【工程管理】->【运动控制轴】->【右键】->【添加轴】；

轴类型选择本地编码器轴。



9.1.3 编码器轴用户单位与换算

编码器轴对编码器信号解码时采用脉冲单位，编码器轴指令则使用例如毫米、度、英寸等常见的度量单位，我们称之为用户单位（Unit）。通过单位换算可将脉冲数转换为为用户单位（Unit），用户单位（Unit）根据实际应用可定义为设备相关单位（毫米、转等）。

单位换算需要设置的参数如下表：

参数名称	功能
电机/编码器旋转一圈的脉冲数	根据编码器分辨率，设定电机转 1 圈的脉冲数
是否使用变速装置	指定是否使用变速装置
电机/编码器旋转一圈的移动量	在不使用变速装置时，电机转 1 圈的工件移动量
工作台旋转一圈的移动量	在使用变速装置时，工作台转 1 圈的工件移动量
齿轮比分子	设定工件侧的齿轮比
齿轮比分母	设定电机侧的齿轮比

在齿轮传动中，通常把驱动轴称为输入轴（或者传动轴），被驱动轴称为输出轴。因此，齿轮比中分子代表输入轴的转速（通常也代表输入轴所带动的齿轮），分母代表输出轴的转速（通常也代表输出轴所带动的齿轮）。如果齿轮比为 a：b，则表示输入轴每转动 a 圈，输出轴就会转动 b 圈。

如伺服电机通过减速机连接丝杆带动工作台运动，通过 PLC 控制器对编码器计数反馈工作台位置，计数器对编码器脉冲计数，以脉冲为单位；计数器轴表示工作台位置，则以毫米为单位。因此，在程序中统一以用户单位（Unit）表示计数器轴的用户单位。用户单位（Unit）和脉冲的转换关系如下：

电机/编码器旋转一圈的脉冲数设置：输入框内 16#表示十六进制数，如编码器旋转一圈的脉冲数为 10000 个脉冲，对应十六进制值为 2710，则输入 16#2710。

工作行程设置：工作行程设置可以不使用变速装置或使用变速装置。

- **不使用变速装置：**

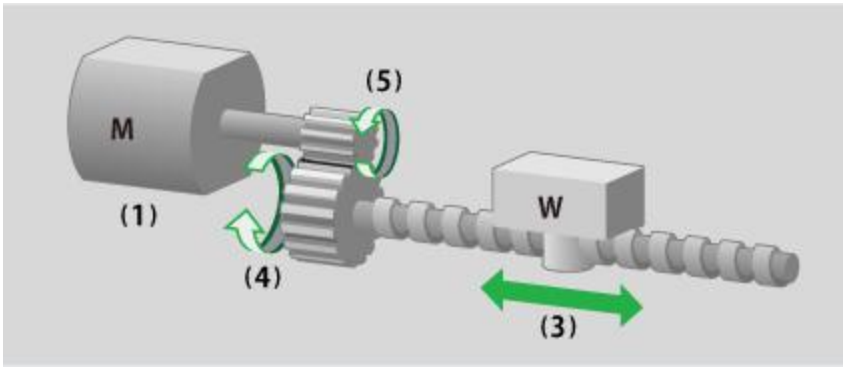
当不使用变速装置时，用户单位到脉冲单位的转换公式如下：

$$脉冲数(pulse)=\frac{电机/编码器旋转一圈的脉冲数[DINT]}{工作台旋转一圈的移动量[REAL]}*移动距离(Unit)$$

如编码器旋转一圈对应的工作轴旋转一圈，用户单位（Unit）为转，则电机/编码器旋转一圈的工作行程设置为 1 即可。

▪ **使用变速装置**

在线型模式下典型工况如下图所示：



其中，(1)为伺服电机，(3)为工件，(4)为齿轮比分子，(5)为齿轮比分母。

由用户单位到脉冲单位的计算公式如下：

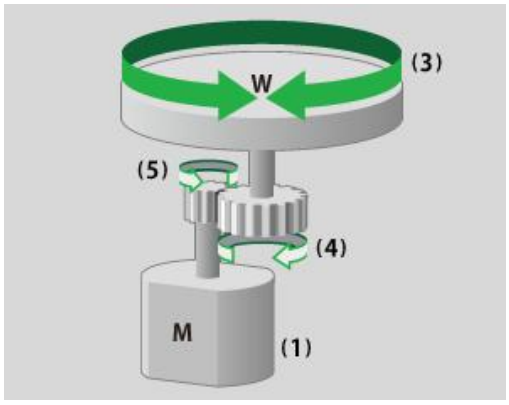
$$脉冲数(pulse)=\frac{电机/编码器旋转一圈的脉冲数[DINT]*齿轮比分子[DINT]}{工作台旋转一圈的移动量[REAL]*齿轮比分母[DINT]}*移动距离(Unit)$$

例如：伺服电机通过减速机连接丝杆带动工作台运动，丝杆旋转一圈的工作行程为 5mm，减速比为 20:10，则输出轴每转动一圈，对应的输入轴（电机）需要旋转 20 圈，因此可以设定齿轮比分子为 20，齿轮比分母为 10，在传递到输出轴（丝杆）之前，先经过齿轮比的放大作用，确保能够使工作台运动所需的距离，设置如下：

工作台旋转一圈的移动量	5.00	Unit
齿轮比分子	20	
齿轮比分母	10	

注意：除了减速比和齿轮比以外，还要根据伺服电机的性能参数和工作台的负载特点确定电机的功率大小，并根据减速比和齿轮比计算出所需的电机转速和扭矩，从而保证系统能够正常工作。

旋转模式下典型工况如下图所示：



其中，(1)为伺服电机，(3)为工件，(4)为齿轮比分子，(5)为齿轮比分母。

由用户单位到脉冲单位的计算公式如下：

$$脉冲数(pulse)=\frac{电机/编码器旋转一圈的脉冲数[DINT]*齿轮比分子[DINT]}{工作台旋转一圈的移动量[REAL]*齿轮比分母[DINT]}*移动距离(Unit)$$

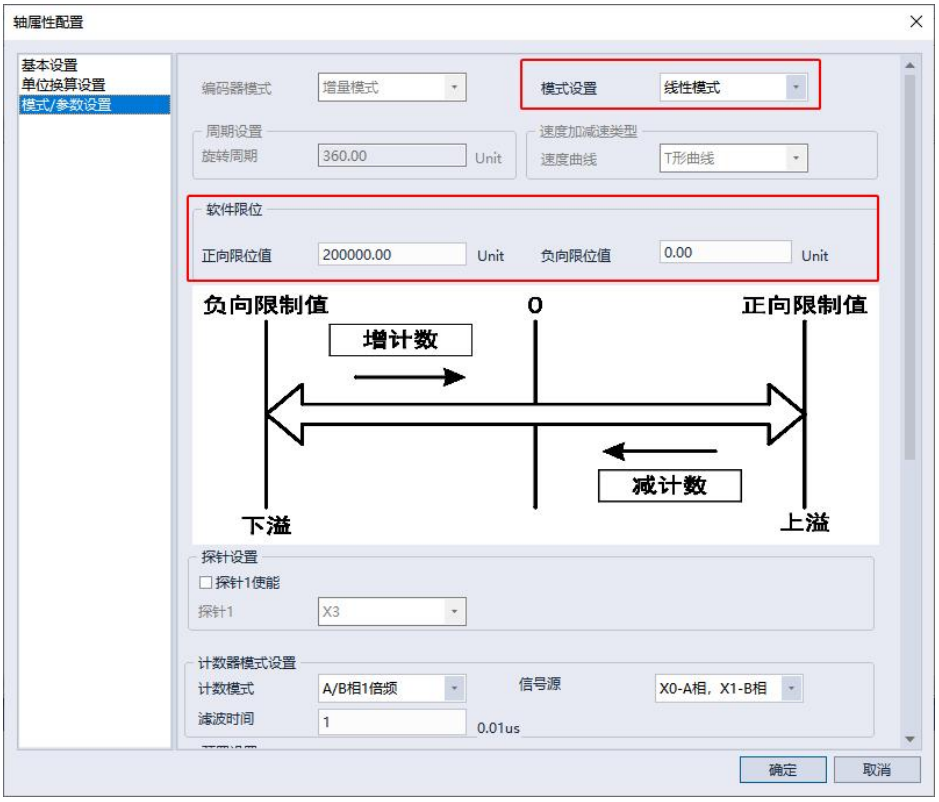
9.1.4 设置工作模式

(1) 线性模式

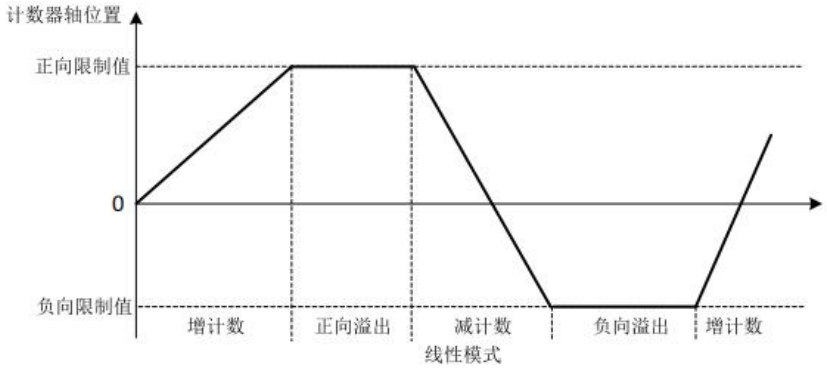
编码器轴的位置在负向限制值和正向限制值之间变化，编码器轴的位置达到限制值后，继续输入同向脉冲，计数器轴报溢出错误，同时计数器轴位置保持不变。编码器轴报溢出后，输入反向脉冲，编码器轴反向计数，溢出错误撤除。

线性模式下，可以在界面中设置编码器轴的负向和正向位置限制值，位置单位为用户单位（Unit）。负向限制值必须小于或等于 0，正向限制值必须大于或等于 0。

由于高速计数器为32位计数器，负向限制值与正向限制值换算为脉冲单位后必须在32位整数范围[-2147483648, 2147483647]。

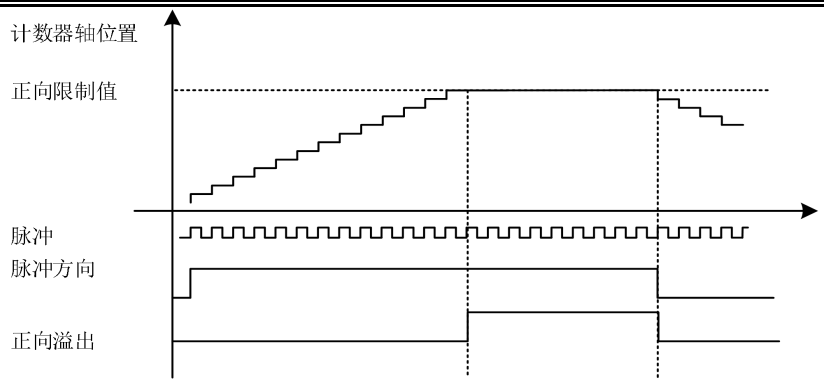


在线性模式下，高速计数器在[负向限制值，正向限制值]的闭区间内工作。当方向为负向时，计数值向负方向减小，到达负向限制值后，计数值不再减小；当方向为正向时，计数值向正方向增加，到达正向限制值后，计数值不再增加。



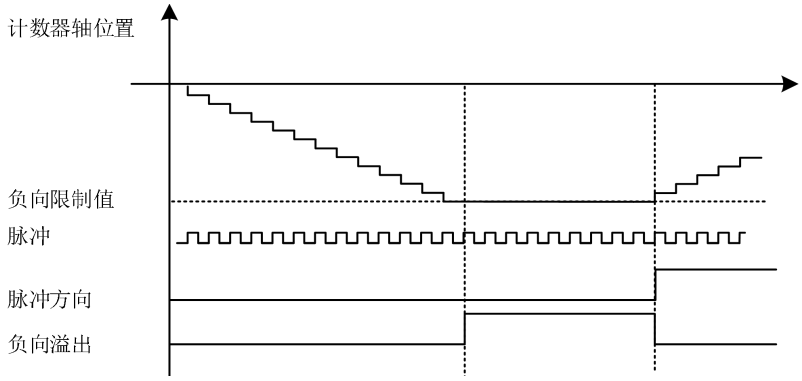
- **正向脉冲计数**

线性模式下，输入正向脉冲，编码器轴位置增计数达到限制值后，继续输入正向脉冲，编码器轴报正向溢出错误，编码器轴位置值保持不变。输入负向脉冲，编码器轴位置减计数，正向溢出错误撤除。



- **负向脉冲计数**

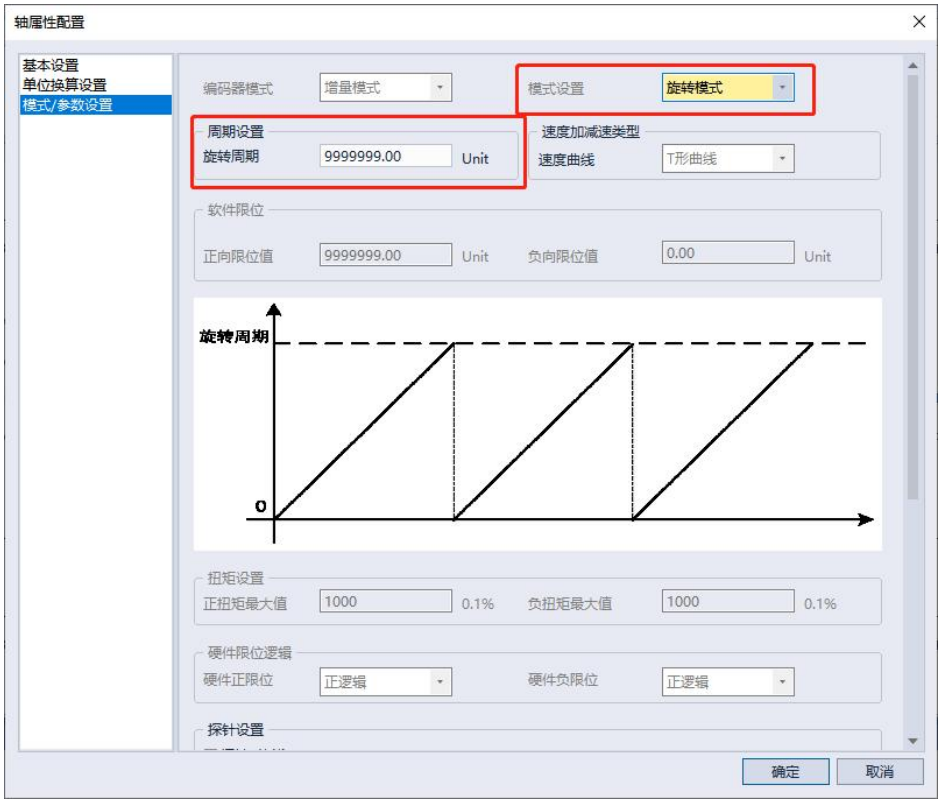
线性模式下，输入负向脉冲，编码器轴位置减计数达到限制值后，继续输入负向脉冲，编码器轴报负向溢出错误，编码器轴位置值保持不变。输入正向脉冲，编码器轴位置增计数，负向溢出错误撤除。



(2) 旋转模式

编码器轴的位置在旋转周期内循环变化，增计数时编码器轴的位置达到旋转周期最大值后变为 0，减计数时编码器轴的位置为 0 后，从旋转周期最大值往下减。旋转模式下，可以在界面中设置计数器轴的旋转周期，周期单位为用户单位（Unit）。

由于高速计数器为 32 位计数器，旋转周期换算为脉冲单位后必需在 32 位整数范围[-2147483648,2147483647]



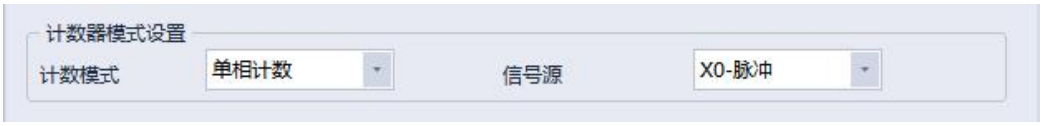
福州富昌维控电子科技有限公司

9.1.5 设置计数器参数

(1) 概述

本地编码器轴支持多种信号计数模式，A/B 相（1/2/4 倍频）、脉冲+方向、单相计数。

信号源：根据不同的计数模式，可以选择不同的信号源。



各计数模式下支持的信号源输入端口如下表所示，不同的本地编码器轴可以选择相同的信号源输入端口。

计数模式	X0	X1	X2	X3	X4	X5	X6	X7	100us	1ms	10ms
AB 相	A 相	B 相	A 相	B 相	A 相	B 相	A 相	B 相	×	×	×
脉冲+方向	脉冲	方向	脉冲	方向	脉冲	方向	脉冲	方向	×	×	×
单相计数	脉冲	脉冲	脉冲	脉冲	脉冲	脉冲	脉冲	脉冲	脉冲	脉冲	脉冲

在使用探针或预置功能时，将编码器轴分为 2 组，即 X0~X3 为计数组一，X4~X7 为计数组二，计数组中同一个信号源只能在一个编码器轴中使用，即当启用了技术组一的探针功能，则 X3 被占用，无法再使用 X3 作为其他计数模式的信号源。内部信号源无法使用探针和预置功能。

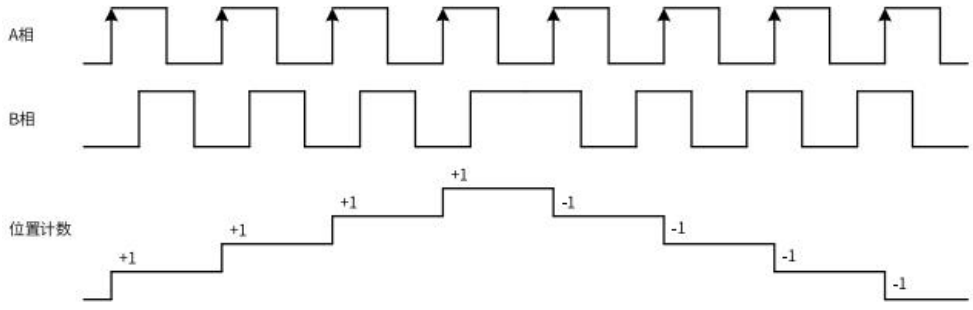
计数模式	X0	X1	X2	X3	X4	X5	X6	X7	100us	1ms	10ms
AB 相	A 相	B 相	探针	预置	A 相	B 相	探针	预置	×	×	×
脉冲+方向	脉冲	方向	探针	预置	脉冲	方向	探针	预置	×	×	×
单相计数	脉冲	×	探针	预置	脉冲	脉冲	探针	预置	脉冲	脉冲	脉冲

(2) AB 相模式

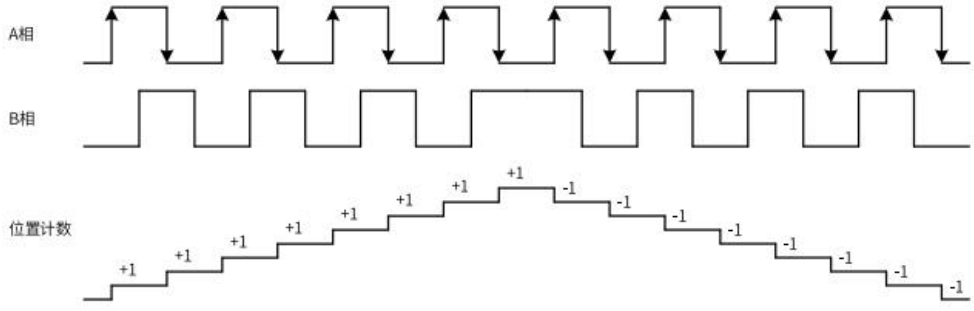
在 A/B 相模式下，编码器产生两个相位相差 90° 的正交相位脉冲信号，即 A 相信号和 B 相信号。当 A 相信号超前 B 相信号时，计数器增计数；当 B 相信号超前 A 相信号时，计数器减计数。

A/B 相脉冲可以设置为 1 倍频、2 倍频或者 4 倍频模式工作。

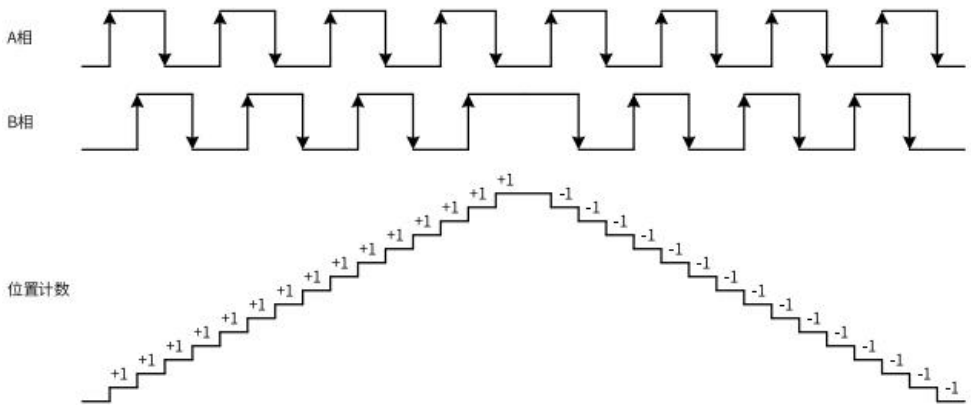
A/B 相 1 倍频模式下，只对 A 相脉冲的上升沿计数，如下图所示：



A/B 相 2 倍频模式下，对 A 相脉冲的上升/下降沿计数，如下图所示：

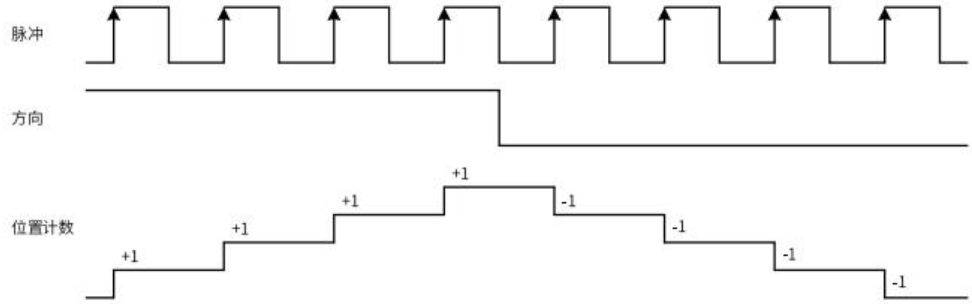


A/B 相 4 倍频模式下，对 A 相脉冲和 B 相脉冲的上升/下降沿计数，如下图所示：



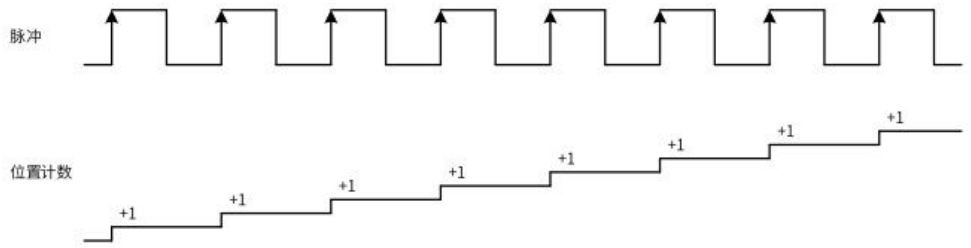
(3) 脉冲+方向模式

在该模式下，方向信号为 ON 时，高速计数器对脉冲信号增计数，方向信号为 OFF 时，高速计数器对脉冲信号减计数，如下图所示：



(4) 单相计数模式

在该模式下，高速计数器对脉冲信号增计数，在输入脉冲上升沿时，位置计数加 1。



(5) 探针设置

每个编码器轴支持 1 路外部输入锁存计数器当前值，实现探针功能。计数组一固定为端子 X3，计数组二固定为端子 X7。

启用探针后，在探针触发后，会将当前计数值保存，通过 HC_TouchProbe 功能块指令读取计数器轴的探针位置。

(6) 预置端子输入

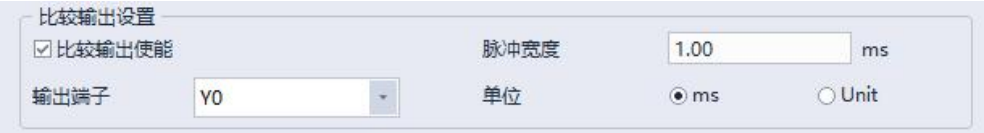
通过勾选预置使能启用外部输入预置计数器值，计数组一固定为端子 X2，计数组二固定为端子 X6。

启用预置功能后，通过 HC_Preset 功能块指令实现外部输入对编码器轴位置预置。

(7) 比较输出端子设置

勾选“比较输出使能”后，不通过软件处理即可实现比较相等时硬件输出，实时性高，输出响应可达微秒级别。

- 启动比较输出功能后，配合功能块指令，比较相等时通过硬件电路控制输出为 ON，输出端子可任意选择 Y0~Y7，输出为 ON 的脉冲宽度可选择为时间单位（ms）或用户单位（Unit）；
- 每个本地编码器轴配备一路比较输出功能，可根据需求配置输入端子与输出脉冲宽度；
- 配置完成后，通过 HC_Compare、HC_ArrayCompare、HC_StepCompare 功能块指令，实现轴位置比较输出；
- 单位选择 ms 时，设置的时间范围为 0.1~6553.5ms。单位选择 Unit 时，应确保设置的值转换为脉冲单位后在 1~65535 的范围。

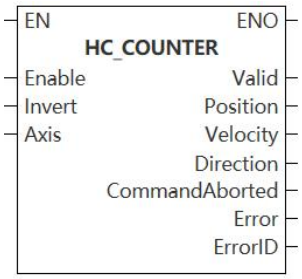


比较输出是直接通过硬件控制端口输出，不通过软件处理，因此不能通过程序中的 Y 软元件显示比较输出的状态。Y 软元件和比较输出对输出端口控制为或关系，若 Y 软元件持续控制为 ON 状态，则实际端口输出保持为 ON。

9.2 HC 轴控

1、HC_COUNTER 高速计数器使能

HC_COUNTER



该功能块用于控制高速计数器启动或停止。

内容、范围、数据类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
HC_COUNTER	输入	Enable	BOOL	否	TRUE/FALSE	高速计数器使能
		Invert	BOOL	是	TRUE/FALSE	计数反向
		Axis	WORD	否	[0,65535]	轴名称/轴号，指定操作的本地编码器轴
	输出	Valid	BOOL	是	TRUE/FALSE	有效状态，进入计数状态为 ON
		Position	LREAL	是	-	当前位置，用户单位
		Velocity	LREAL	是	-	当前速度，用户单位
		Direction	BOOL	是	TRUE/FALSE	计数方向（TRUE：正向，FALSE：反向）
		CommandAborted	BOOL	是	TRUE/FALSE	执行中断
		Error	BOOL	是	TRUE/FALSE	错误
		ErrorID	DWORD	是	详见错误码	错误码

使用的软元件

指令	参数	软元件																					偏移修饰	双字扩展	脉冲型扩展
		X	Y	M	S	SM	T(Bit)	HSC(Bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	DXX	XXP
HC_COUNTER	Enable	●	●	●	●	●																				
	Invert	●	●	●	●	●																				
	Axis															●	●									
	Valid		●	●	●	●																				
	Position																									
	Velocity																									
	Direction		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

- 使能 HC_COUNTER 功能块，可对当前计数器轴位置计数，并实时测量当前速度，显示当前计数方向。
- 计数器轴位置仅根据当前计数模式设置，在计数器轴模式的范围内变化，位置单位为 Unit。

注意

（1）当且仅当功能块 Enable 使能时，更新当前轴位置、速度、计数方向；

（2）若扫描程序中，同时使用两个功能块计数，且指定编码器轴参数 Axis 相同时，先使能的功能块执行将会被后使能的功能块中断，此时前者 CommandAborted 被置为 True，有效状态 Valid 置为 False；

福州富昌维控电子科技有限公司

（3）功能块执行出错后，必须重新使能 Enable 以清除错误，否则功能块不执行计数。

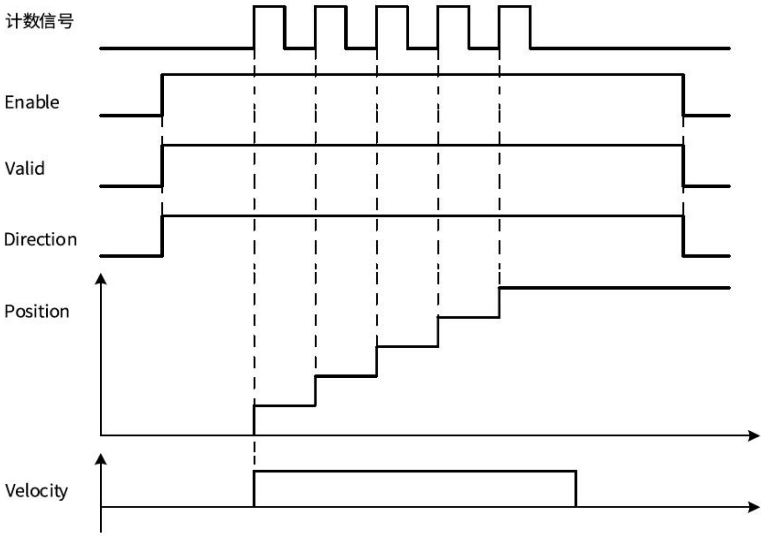
Invert（计数反向）参数

- 通过 Invert 参数设置计数器的计数方向，不同计数模式的计数方向选择定义如下表所示。更改 Invert 的设置后，若功能块已使能可直接生效。

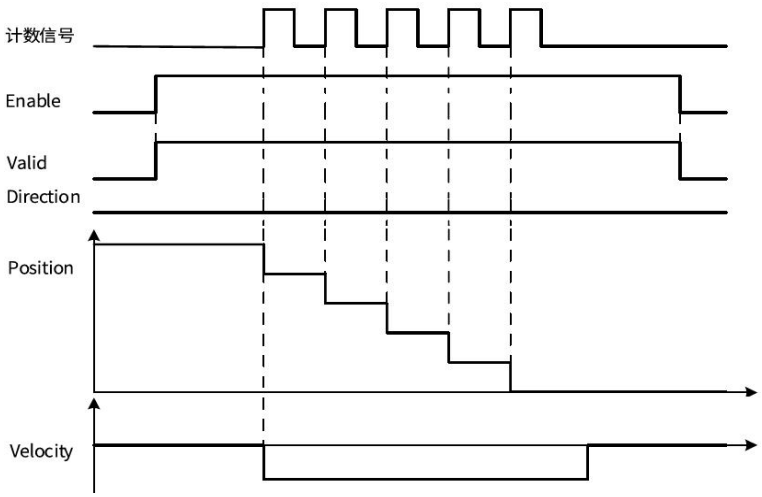
Invert	A/B 相	脉冲＋方向	CW/CCW	单相计数
FALSE	A 相超前 B 相增计数 B 相超前 A 相减计数	方向信号低电平减计数 方向信号高电平增计数	A 相增计数 B 相减计数	增计数
TRUE	A 相超前 B 相减计数 B 相超前 A 相增计数	方向信号低电平增计数 方向信号高电平减计数	A 相减计数 B 相增计数	减计数

说明

- 以“脉冲+方向”模式为例，方向信号=ON/Invert=FALSE 或方向信号=OFF/Invert=TRUE 时，计数器增计数，时序图如下图所示：



- 以“脉冲+方向”模式为例，方向信号=ON/Invert=TRUE 或方向信号=OFF/Invert=FALSE 时，计数器减计数，时序图如下图所示：



错误码

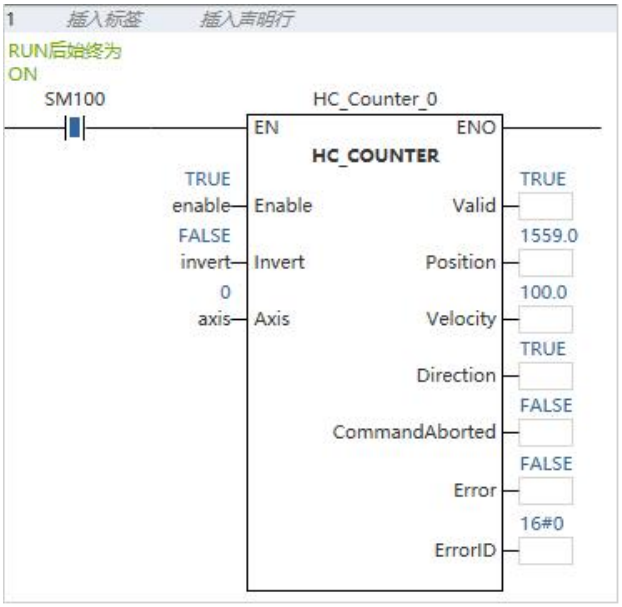
错误码	内容
0x00400005	计数位置超正限制值
0x00400006	计数位置超负限制值
0x00700005	轴未定义或无效轴号
0x00800008	功能块运行时修改轴号

福州富昌维控电子科技有限公司

示例

应用实例

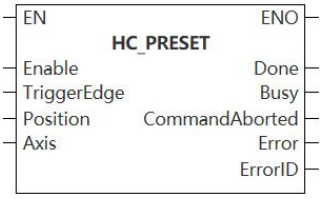
- 计数器轴速度为当前实时速度，速度单位为 Unit/s。计数器轴可测量的最小速度为 1s 时间内计数器 1 个脉冲对应的速度，如计数器一个脉冲对应 0.01Unit，则可测量的最小速度为 0.01Unit/s。



- 例如：上图中当前功能块 enable 为 ON 时，更新当前编码器轴位置、速度、显示计数方向，测量当前 AXIS_0 中输入脉冲速度为 100Unit/s，当前位置为 1559.0Unit。

2、HC_PRESET 高速计数器预置值

HC_PRESET



根据设定的预置触发条件将计数器值设置为预置值。

内容、范围、数据类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
HC_PRESET	输入	Enable	BOOL	否	TRUE/FALSE	高速计数器预置使能
		TriggerEdge	WORD	是	[0,1]	触发边沿。 0：指令 Enable 上升沿触发 1：外部 X 上升沿触发
		Position	LREAL	否	-	预置位置，用户单位
		Axis	WORD	否	[0,65535]	轴名称/轴号，指定操作的本地编码器轴
	输出	Done	BOOL	是	TRUE/FALSE	完成
		Busy	BOOL	是	TRUE/FALSE	执行中
		CommandAborted	BOOL	是	TRUE/FALSE	执行中断
		Error	BOOL	是	TRUE/FALSE	错误
		ErrorID	DWORD	是	详见错误码	错误码

使用的软元件

指令	参数	软元件																					偏移修饰	双字扩展	脉冲型扩展
		X	Y	M	S	SM	T(Bit)	HSC(Bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	DXX	XXP
HC_PRESET	Enable	●	●	●	●	●																				
	TriggerEdge															●	●									
	Position																									
	Axis															●	●									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

功能

- 使能 HC_PRESET 功能块，根据设置的预置触发条件，实现对计数器轴位置赋值。

TriggerEdge（触发边沿）参数

- 可选择指令上升沿触发或外部 X 输入触发预置功能。

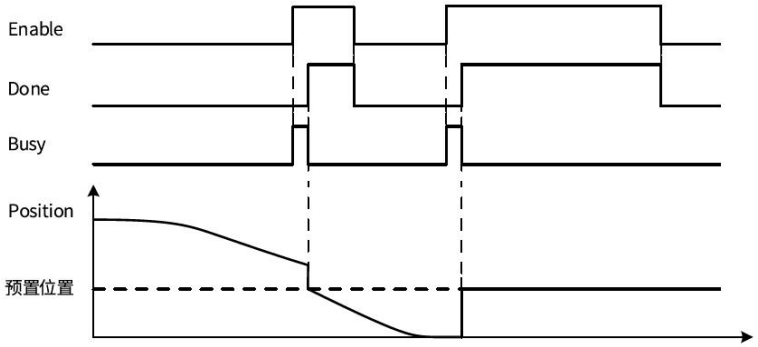
参数	设置	定义
TriggerEdge	0	Enable 上升沿触发
	1	外部 X 上升沿触发

- 预置条件选择外部 X 输入触发时，需要在计数器参数设置勾选预置功能，选择输入端子和触发条件，输入端子可设置选择 X2、X6，触发条件可选择上升沿触发。

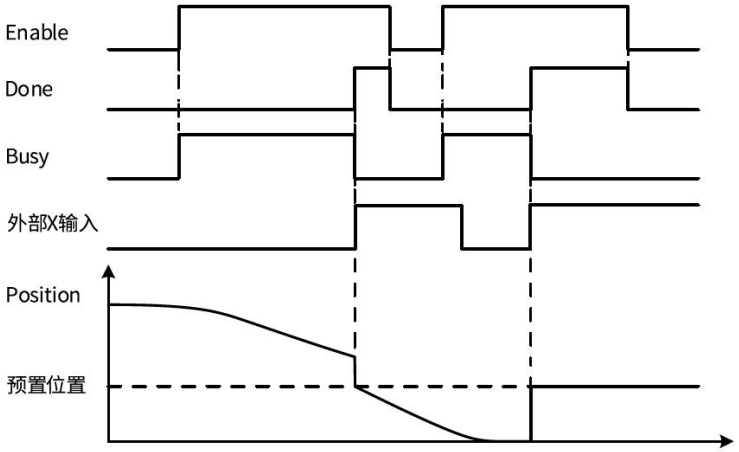
福州富昌维控电子科技有限公司

说明

- 指令 Enable 上升沿触发（TriggerEdge=0），指令时序图如下图所示：



- 外部 X 中断触发，以上升沿触发（TriggerEdge=1）为例，指令时序图入下图所示：

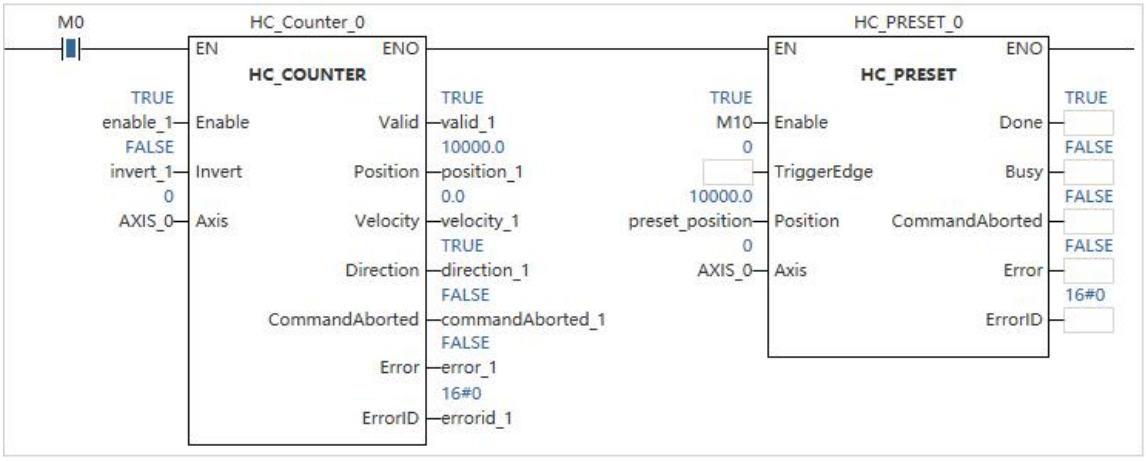


错误码

错误码	内容
0x00700000	设置触发边沿为外部 X 边沿触发，但轴配置界面预置功能未使能
0x00700005	轴未定义或无效轴号
0x00700008	本地编码器轴高速计数器未使能
0x00800006	设置预置位置超限制值/设置触发边沿参数超范围
0x00800008	功能块运行时修改轴号

示例

应用实例

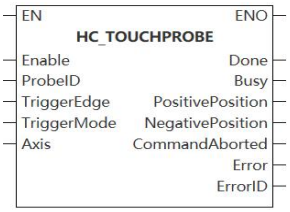


- 设置预置触发条件为Enable上升沿触发(TriggerEdge=0)，预置条件满足后，修改当前HC_COUNTER轴位置(position)为预置位置，此时 HC_COUNTER 功能块读取到当前位置即为预置位置。

福州富昌维控电子科技有限公司

3、HC_TOUCHPROBE 高速计数器探针

HC_TOUCHPROBE



根据设定的触发模式及条件，记录触发信号发生时计数器值。

内容、范围、数据类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
HC_TOUCH PROBE	输入	Enable	BOOL	否	TRUE/FALSE	高速计数器预置使能
		ProbeID	WORD	否	0	探针 ID。 0：探针 1。
		TriggerEdge	WORD	否	[1,3]	触发边沿。 1：外部 X 上升沿触发； 2：外部 X 下降沿触发； 3：外部 X 上升沿或下降沿触发。
		TriggerMode	WORD	是	[0,1]	触发模式。 0：单次触发； 1：连续触发。
		Axis	WORD	否	[0,65535]	轴名称/轴号，指定操作的本地编码器轴
	输出	Done	BOOL	是	TRUE/FALSE	完成
		Busy	BOOL	是	TRUE/FALSE	执行中
		PositivePosition	LREAL	是	-	上升沿锁存位置，用户单位
		NegativePosition	LREAL	是	-	下降沿锁存位置，用户单位
		CommandAborted	BOOL	是	TRUE/FALSE	执行中断
		Error	BOOL	是	TRUE/FALSE	错误
		ErrorID	DWORD	是	详见错误码	错误码

使用的软元件

指令	参数	软元件																						偏移修饰	双字扩展	脉冲型扩展
		X	Y	M	S	SM	T(Bit)	HSC(Bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	DXX	XXP
HC_TOUCH PROBE	Enable	●	●	●	●	●																				
	ProbeID															●	●									
	TriggerEdge															●	●									
	TriggerMode															●	●									
	Axis															●	●									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	PositivePosition																									
	NegativePosition																									
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

功能

- 使能 HC_TOUCHPROBE 功能块指令，可在外部输入触发条件有效时，锁存计数器轴位置值。
- 每个计数器轴仅支持 1 路探针，使用时，需要再计数器参数设置勾选对应的探针功能，选择输入端子和触发模式、条件，输入端子可设置选择 X3、X7。

说明

TriggerEdge（触发边沿）参数

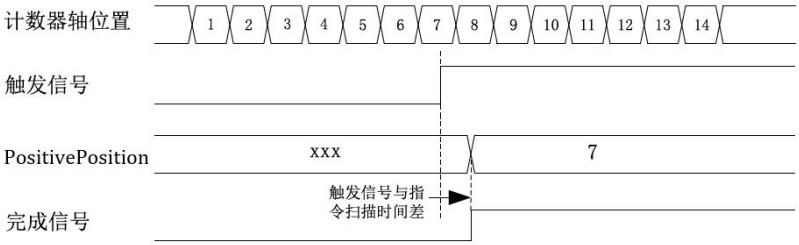
- 可设置探针触发边沿，上升沿触发位置锁存在输出参数 PositivePosition 中，下降沿触发位置锁存在输出参数 NegativePosition 中。

参数	设置	定义
TriggerEdge	1	外部 X 上升沿触发
	2	外部 X 下降沿触发
	3	外部 X 上升沿或下降沿触发

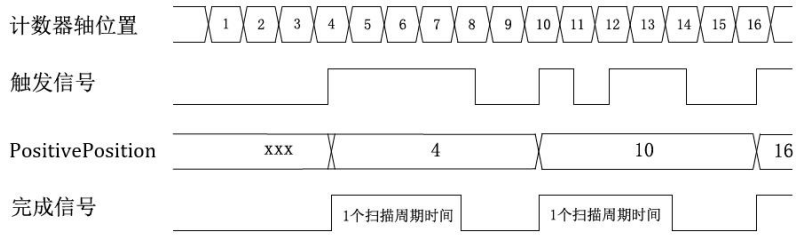
TriggerMode（触发模式）参数

- 可设置单次触发和连续触发模式。

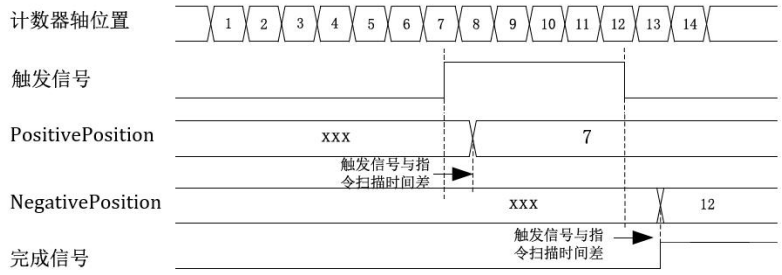
使用单次触发模式，功能块指令 Enable 有效，外部输入触发条件有效时，锁存 1 次计数器轴位置，输出完成信号。探针位置根据触发边沿实时锁存计数器轴位置，不受程序执行影响。程序指令执行时，受扫描周期影响，程序扫描执行到锁存指令时，将锁存位置更新到指令输出参数中。

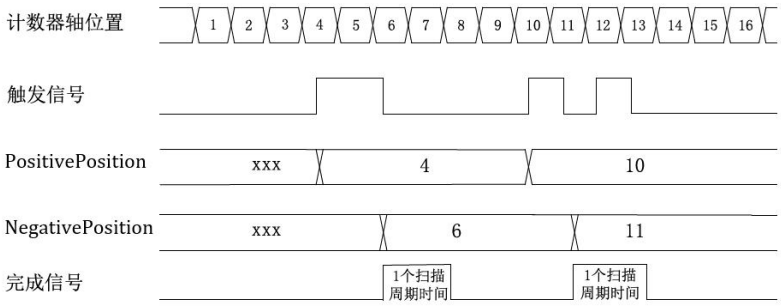


使用连续触发模式，功能块指令 Enable 有效，外部输入触发条件有效时，锁存计数器轴位置，输出完成信号有效时间 1 个扫描周期。完成信号变为 OFF 后，外部输入触发条件有效，会继续锁存计数器轴位置，并输出有效时间为 1 个扫描周期的完成信号。在完成信号有效的 1 个扫描周期时间内，若外部输入触发条件有效，此时不会锁存计数器轴的位置。



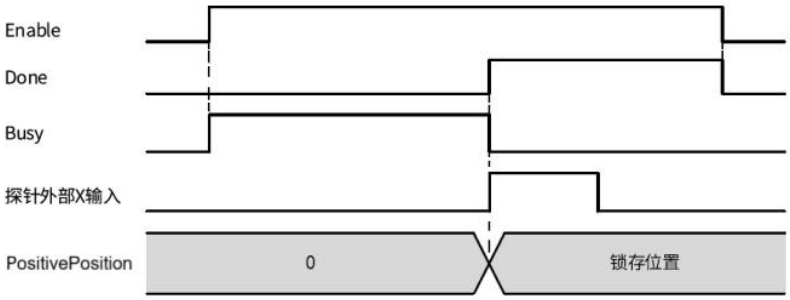
使用双边沿触发模式，当上升沿和下降沿都触发完成锁存后，输出完成信号。单次触发模式时，完成信号持续到指令完成；连续触发模式是，完成信号有效时间 1 个扫描周期，完成信号有效的 1 个扫描周期内，不响应触发锁存信号。



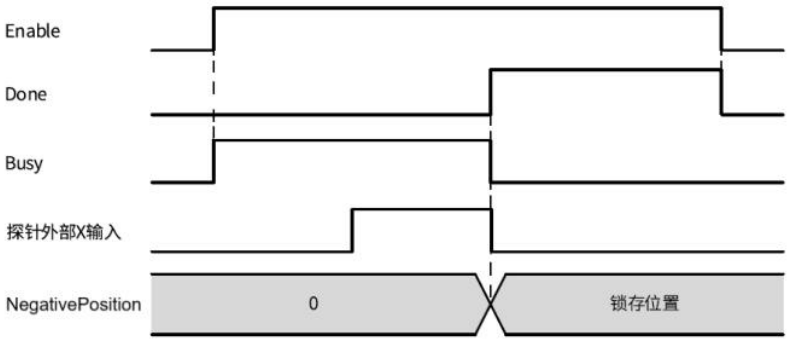


时序图

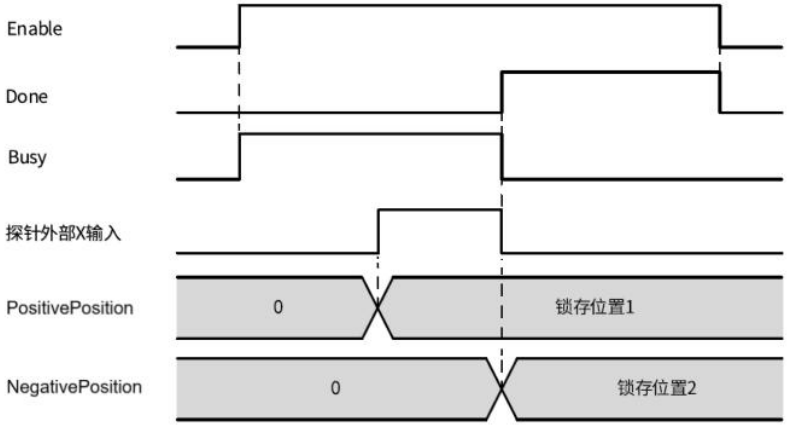
- 外部 X 上升沿触发（TriggerEdge=1），单次触发模式（TriggerMode=0），指令时序图如下图所示：



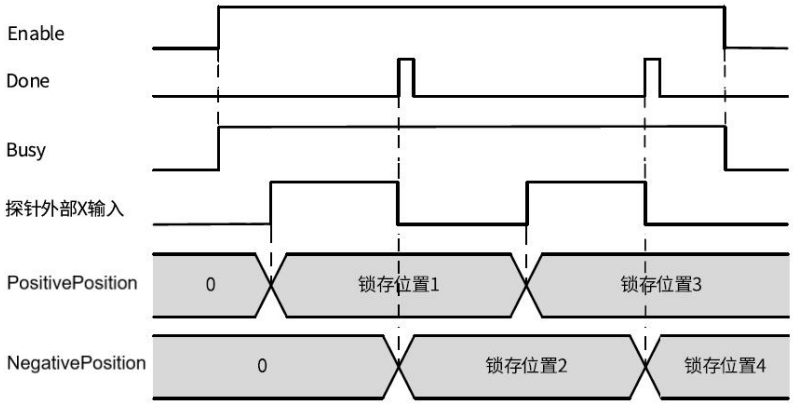
- 外部 X 下降沿触发（TriggerEdge=2），单次触发模式（TriggerMode=0），指令时序图如下图所示：



- 外部 X 上升沿与下降沿触发（TriggerEdge=3），单次触发模式（TriggerMode=0），指令时序图如下图所示：



- 外部 X 上升沿与下降沿触发（TriggerEdge=3），连续触发模式（TriggerMode=1），指令时序图如下图所示：

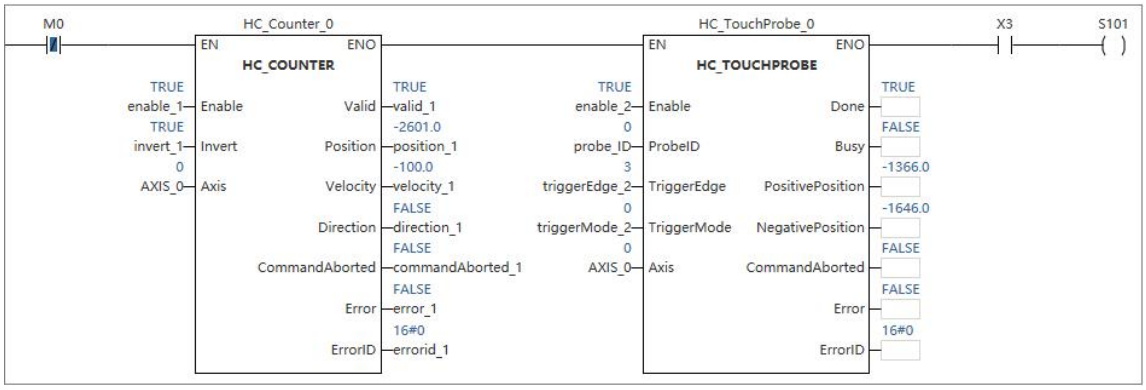


错误码

错误码	内容
0x00700000	轴配置界面探针未使能
0x00700005	轴未定义或无效轴号
0x00800006	设置触发边沿/触发模式/探针 ID 错误
0x00800008	功能块运行时修改轴号

示例

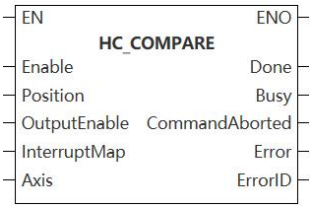
应用实例



- 轴配置界面设置探针使能，配置引脚为 X3，设置触发模式为单次触发(TriggerMode = 0)，触发边沿为外部 X 上升沿和下降沿触发(TriggerEdge = 3)；
- 功能块 Enable 后，Busy 置为 TRUE；
- X3 检测到上升沿时，锁存位置到 PositivePosition；X3 检测到下降沿时，锁存位置到 NegativePosition，此时本次功能块执行完成，Busy 状态置为 False，Done 置为 TRUE。

4、HC_COMPARE 高速计数器比较

HC_COMPARE



探测计数器计数到达指定值。

内容、范围、数据类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
HC_COMPARE	输入	Enable	BOOL	否	TRUE/FALSE	高速计数器比较使能
		Position	LREAL	否	-	比较位置，用户单位
		OutputEnable	BOOL	是	TRUE/FALSE	硬件输出使能。 FALSE：不使用硬件输出； TRUE：使用硬件输出。
		InterruptMap	POINTER	是	-	中断映射，比较值与计数值相匹配时产生中断，并关联中断，可直接输入中断名称，例如： IRQ0：关联中断 IRQ0； 无：不关联中断。
		Axis	WORD	否	[0,65535]	轴名称/轴号，指定操作的本地编码器轴
	输出	Done	BOOL	是	TRUE/FALSE	完成
		Busy	BOOL	是	TRUE/FALSE	执行中
		CommandAborted	BOOL	是	TRUE/FALSE	执行中断
		Error	BOOL	是	TRUE/FALSE	错误
		ErrorID	DWORD	是	详见错误码	错误码

使用的软元件

指令	参数	软元件																					偏移修饰	双字扩展	脉冲型扩展
		X	Y	M	S	SM	T(Bit)	HSC(Bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	DXX	XXP
HC_COMPARE	Enable	●	●	●	●	●																				
	Position																									
	OutputEnable	●	●	●	●	●																				
	InterruptMap																									
	Axis															●	●									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

功能

- 使能 HC_COMPARE 可实现计数器轴的单个位置比较。

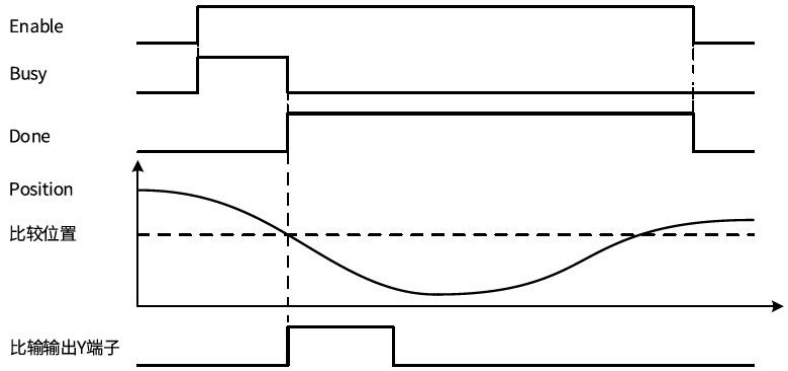
注意

（1）使用本功能块时，InterruptMap 参数若指定中断映射程序名，需在扫描程序中使用 EI 指令使能中断，否则中断程序无法执行；

（2）若扫描程序中存在两个高速比较功能块，且两个功能块 Axis 参数指定本地编码器轴相同时，先执行的功能块将会被后执行的功能块中断，ComandAborted 执行中断将会置 ON。

说明

- 使用硬件输出（OutputEnable=TRUE），指令时序图如下图所示。

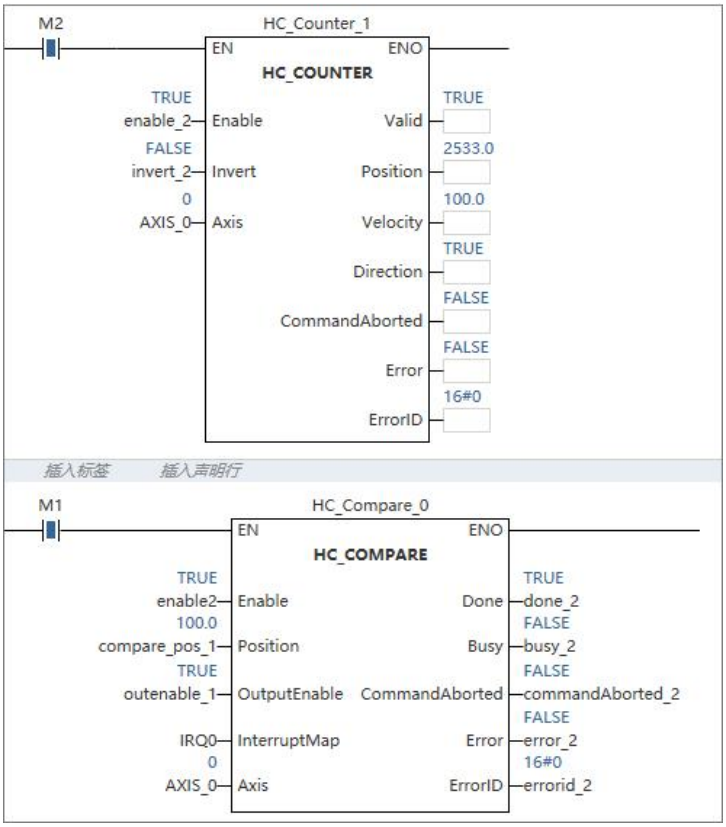


错误码

错误码	内容
0x00700000	OutEnable 为 ON，但轴配置界面比较输出未使能
0x00700005	轴未定义或无效轴号
0x00800006	设置比较位置超限位
0x00800008	功能块运行时修改轴号

示例

应用实例

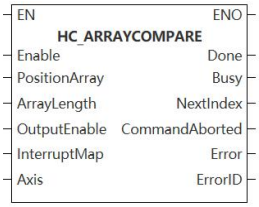


- 设置比较位置为 100，配置中断映射程序（InterruptMap = IRQ0），配置硬件输出使能（outputEnable = TRUE），使能功能块（Enable = TRUE）；
- 当计数器轴位置到达 100 时，触发比较中断，调用中断 IRQ0，若此时硬件输出使能，对应 Y 输出点输出设定脉宽脉冲。

福州富昌维控电子科技有限公司

5、HC_ARRAYCOMPARE 高速计数器数组比较

HC_ARRAYCOMPARE



按次序连续探测计数器计数到达指定数组序列。

内容、范围、数据类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
HC_ARRAY COMPARE	输入	Enable	BOOL	否	TRUE/FALSE	高速计数器比较使能
		PositionArray	ARRAY OF LREAL	否	-	比较位置数组，用户单位
		ArrayLength	WORD	否	[1,32]	数组长度
		OutputEnable	WORD	是	TRUE/FALSE	硬件输出使能。 FALSE：不使用硬件输出； TRUE：使用硬件输出。
		InterruptMap	POINTER	是	-	中断映射，比较值与计数值相匹配时产生中断，并关联中断，可直接输入中断名称，例如： IRQ0：关联中断 IRQ0 无：不关联中断
		Axis	WORD	否	[0,65535]	轴名称/轴号，指定操作的本地编码器轴
	输出	Done	BOOL	是	TRUE/FALSE	完成
		Busy	BOOL	是	TRUE/FALSE	执行中
		NextIndex	WORD	是	[0,32]	下一个比较置索引
		CommandAborted	BOOL	是	TRUE/FALSE	执行中断
		Error	BOOL	是	TRUE/FALSE	错误
		ErrorID	DWORD	是	详见错误码	错误码

使用的软元件

指令	参数	软元件																					偏移修饰	双字扩展	脉冲型扩展
		X	Y	M	S	SM	T(Bit)	HSC(Bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	DXX	XXP
HC_ARRAY COMPARE	Enable	●	●	●	●	●																				
	PositionArray																									
	ArrayLength															●	●									
	OutputEnable	●	●	●	●	●																				
	InterruptMap																									
	Axis															●	●									
	Done		●	●	●	●																				
	Busy		●	●	●	●																				
	NextIndex															●	●									
	CommandAborted		●	●	●	●																				
	Error		●	●	●	●																				
	ErrorID																									

功能

- 使能 HC_ARRAYCOMPARE 可实现计数器轴的多位置连续比较。

说明

- 功能块 Enable 有效时，计数器轴位置与数组第一个位置开始比较，比较相等后，与数组下一个位置比较。功能块中 ArrayLength 参数用于设定数组长度，所有数组长度设定的数组位置比较完成后，完成信号持续输出，完成多位置连续比较。
- 输出参数 NextIndex 指示下一个比较位置索引，即已完成比较相等的个数。

注意

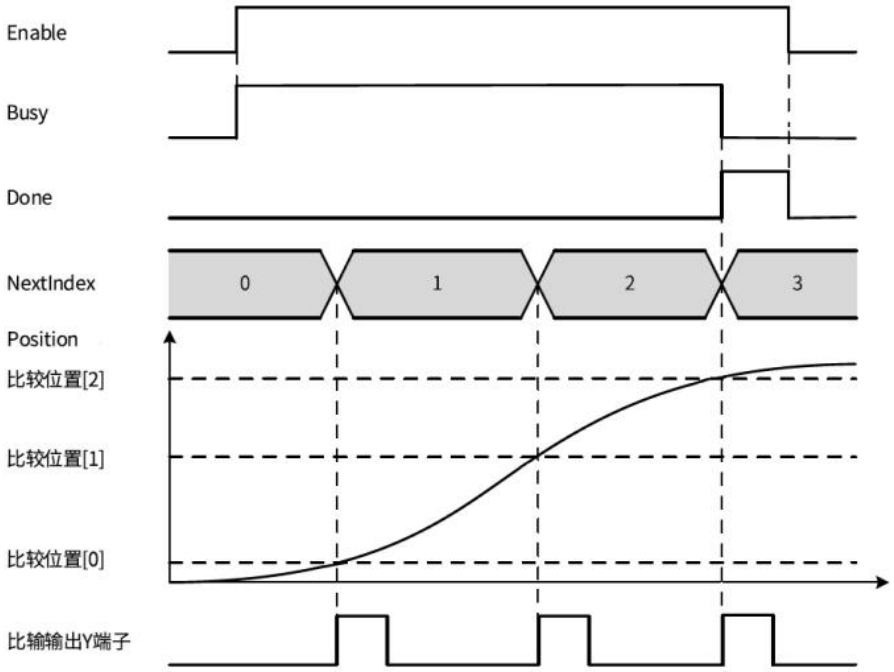
（1）使用本功能块时，InterruptMap 参数若指定中断映射程序名，需在扫描程序中使用 EI 指令使能中断，否则中断程序无法执行；

（2）输入比较位置数组 PositionArray 参数时，必须为单调递增序列数组或单调递减序列数组，且相邻两个比较位置不能重复；

（3）若扫描程序中存在两个高速比较功能块，且两个功能块 Axis 参数指定本地编码器轴相同时，先执行的功能块将会被后执行的功能块中断，ComandAborted 执行中断将会置 ON。

时序图

- 比较 3 个位置（ArrayLength=3），使用硬件输出（OutputEnable=TRUE），指令时序图如下图所示：

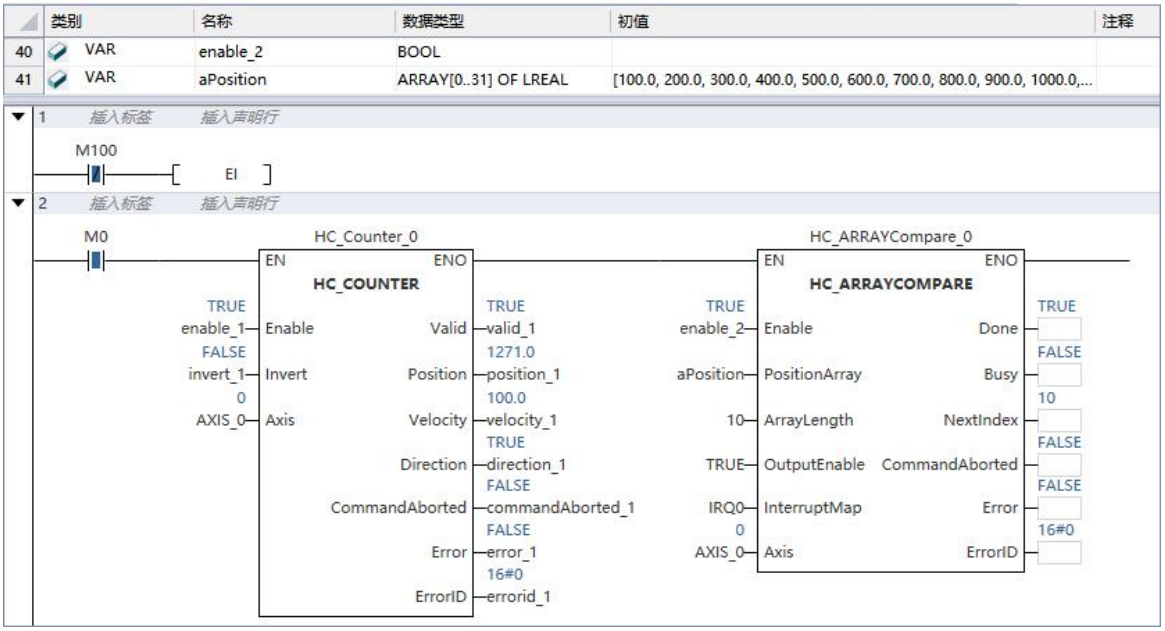


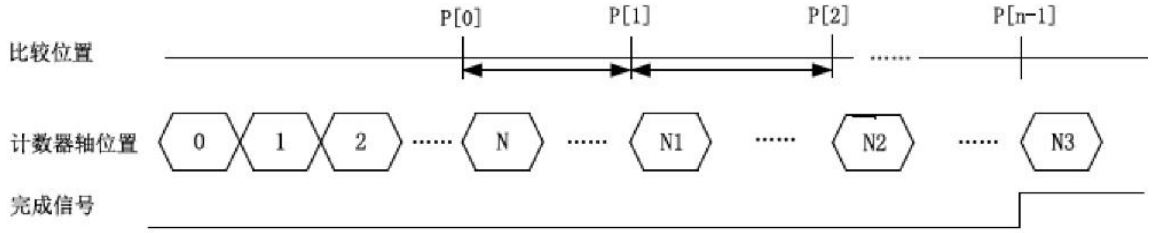
错误码

错误码	内容
0x00700000	OutEnable 为 ON，但轴配置界面比较输出未使能
0x00700005	轴未定义或无效轴号
0x00800006	比较数组未设置或设置数组长度错误
0x00800008	功能块运行时修改轴号

示例

应用实例

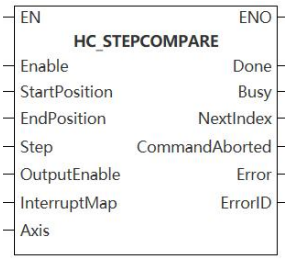




- 设置当前比较位置数组参数及数组长度 n（最长允许设置 ArrayLength=32），功能块 Enable 使能后，当前比较位置开始设置为 PositionArray 索引 0 位置值，当前计数值位置与比较位置相等时，触发比较中断，调用由 InterruptMap 参数指定的中断程序 IRQ0，若配置硬件输出使能（OutputEnable=TRUE），对应 Y 点产生特定脉宽脉冲，若数组中所有比较位置比较完成，Done 完成标志将持续为 TRUE。

6、HC_STEPCOMPARE 高速计数器等间距比较

HC_STEPCOMPARE



按次序连续探测计数器计数到达指定范围与间距的连续序值。

内容、范围、数据类型

指令	引脚类型	参数	变量类型	能否为空	范围	描述
HC_STEP COMPARE	输入	Enable	BOOL	否	TRUE/FALSE	高速计数器比较使能
		StartPosition	LREAL	否	-	起始位置，用户单位
		EndPosition	LREAL	否	-	结束位置，用户单位
		Step	LREAL	否	正长实数	间距，用户单位
		OutputEnable	BOOL	是	TRUE/FALSE	硬件输出使能。 FALSE：不使用硬件输出； TRUE：使用硬件输出。
		InterruptMap	POINTER	是	-	中断映射，比较值与计数值相匹配时产生中断，并关联中断，可直接输入中断名称，例如： IRQ0：关联中断 IRQ0； 无：不关联中断。
		Axis	WORD	否	[0,65535]	轴名称/轴号，指定操作的本地编码器轴
	输出	Done	BOOL	是	TRUE/FALSE	完成
		Busy	BOOL	是	TRUE/FALSE	执行中
		NextIndex	WORD	是	[0,65535]	下一个比较位置索引
		CommandAborted	BOOL	是	TRUE/FALSE	执行中断
		Error	BOOL	是	TRUE/FALSE	错误
		ErrorID	DWORD	是	详见错误码	错误码

使用的软元件

指令	参数	软元件																						偏移修饰	双字扩展	脉冲型扩展
		X	Y	M	S	SM	T(Bit)	HSC(Bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	DXX	XXP
HC_STEPCOMPARE	Enable	●	●	●	●	●																				
	StartPosition																									
	EndPosition															●	●									
	Step	●	●	●	●	●																				
	OutputEnable																									
	InterruptMap															●	●									
	Axis		●	●	●	●																				
	Done		●	●	●	●																				
	Busy															●	●									
	NextIndex		●	●	●	●																				

	CommandAborted		●	●	●	●																								
	Error		●	●	●	●																								
	ErrorID																													

功能

- 使能 HC_STEPCOMPARE 可实现计数器轴的与等间距连续位置比较。

说明

- 功能块 Enable 有效时，计数器轴位置与 StartPosition 位置开始比较，比较相等后，比较位置增加或减小 Step 间距后继续比较。等间距比较完最后 1 个比较位置后，完成信号持续输出。

输出参数 NextIndex

- 指示下一个比较位置索引，索引从 0 开始计算，即 0 表示第 1 个比较位置。因此，该索引正好与已经完成的比较位置个数相等。

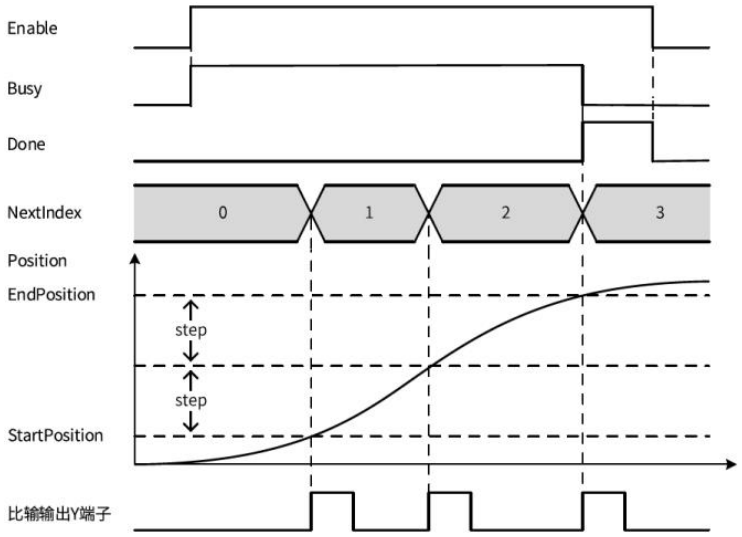
注意

（1）使用本功能块时，InterruptMap 参数若指定中断映射程序名，需在扫描程序中使用 EI 指令使能中断，否则中断程序无法执行。

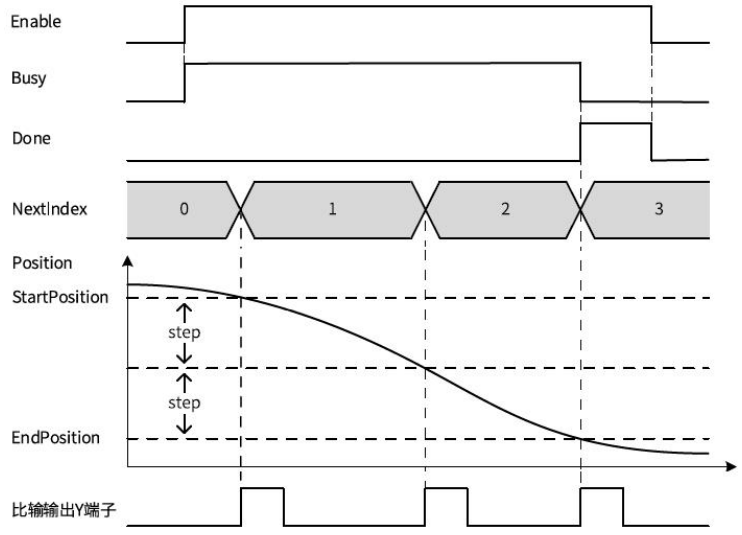
（2）若扫描程序中存在两个高速比较功能块，且两个功能块 Axis 参数指定本地编码器轴相同时，先执行的功能块将会被后执行的功能块中断，ComandAborted 执行中断将会置 ON。

时序图

- 使用硬件输出（OutputEnable=TRUE），StartPosition ＜ EndPosition，指令时序图如下图所示：



- 使用硬件输出（OutputEnable=TRUE），StartPosition ＞ EndPosition，指令时序图如下图所示：



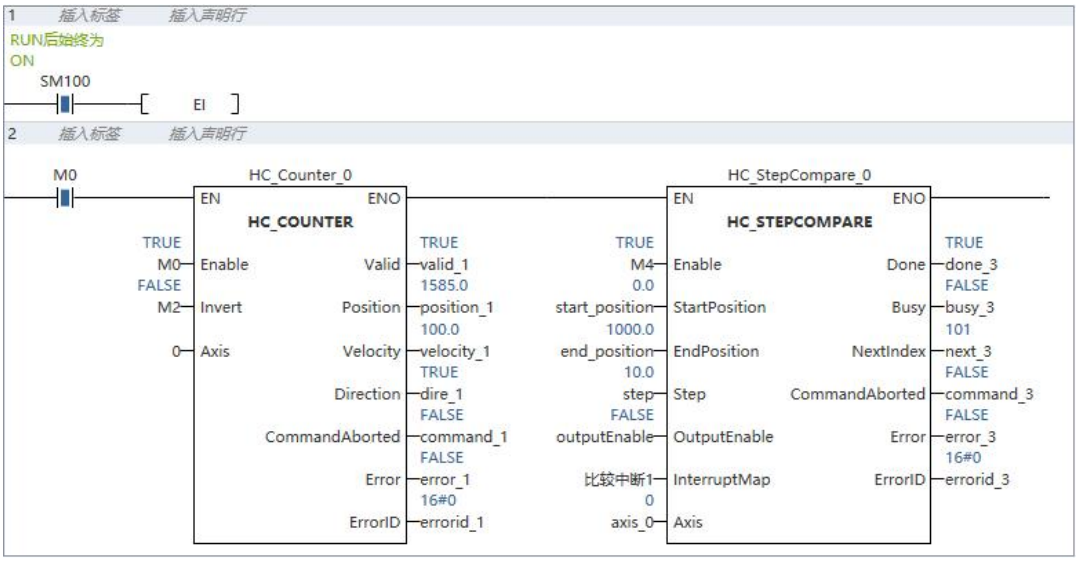
福州富昌维控电子科技有限公司

错误码

错误码	内容
0x00700000	OutEnable 为 ON，但轴配置界面比较输出未使能
0x00700005	轴未定义或无效轴号
0x00800000	比较位置个数超过 65535，且等间距比较未完成
0x00800006	设置起始位置、结束位置、间距错误
0x00800008	功能块运行时修改轴号

示例

应用实例



- 功能块 Enable 使能后，当前比较位置开始设置为 StartPosition 指定位置值，当前计数值位置与比较位置相等时，触发比较中断，调用由 InterruptMap 参数指定的中断程序，若配置硬件输出使能（OutputEnable=TRUE），对应 Y 点产生特定脉宽脉冲，当前位置超过结束位置后，持续输出完成信号。

10 通讯指令

10.1 通讯口协议设置

PROTOCOL/通讯口协议设置

PROTOCOL(P)

对(n)号通讯口设置(s)协议。

-[PROTOCOL　(s)　(n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	要设置的协议编号	-	无符号 BIN16 位	ANY16	INT
(n)	要设置的第几个通讯口，0 表示 COM1、1 表示 COM2	0、1	无符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
PROTOCOL	参数 1												●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

功能

- 这条指令主要用于 RUN 的时候设置协议，也可以在 RUN 第一周期使用；
- 能直接将协议地址（COM1 SD2542、COM2 SD2592）以及对协议修改标识（COM1 SD2543、COM2 SD2593）的值根据指令参数进行设置；
- 协议修改标识（COM1 SD2543、COM2 SD2593）的设置具体的计算公式是：

(参数设置值+对应串口的起始特殊软元件的偏移+10)*2。

  · 例如设置协议为 2 就是（2 +2593 - 2590 +10）*2 也就是（2+3+10）*2=30 。这时候会给 SD2592 设置 2，SD2593 设置 30；

  · 提示：无论是通过自己计算然后向协议修改标识（COM1 SD2543、COM2 SD2593）设置值，还是用这个指令来设置，都是能够在 PLC 处于 RUN 状态下修改协议的；

- 关于协议修改标识（COM1 SD2543、COM2 SD2593）：RUN 过程中，设置协议和修改标识正确时判断（END 时候判断）到正确的第一周期在修改完成时会将协议修改标识（COM1 SD2543、COM2 SD2593）清空，而后再设置相同的值将不作处理。主要为了避免多次的重复设置。

协议编号

协议编号	内容
0H	Wecon Modbus 从站
2H	Modbus RTU 从站
3H	Modbus ASCII 从站
10H	自定义协议
20H	Modbus RTU 主站
30H	Modbus ASCII 主站

相关的软元件

软元件	内容
SD2542	COM1 协议设置
SD2543	COM1 协议修改标识
SD2592	COM2 协议设置
SD2593	COM2 协议修改标识

福州富昌维控电子科技有限公司

注意事项

通讯参数的设置会影响整体的通讯，PLC 中的处理是在没有通讯情况下或者一轮通讯完成之后再进行修改。这一点需要注意。

如果设置的协议与提供的协议编号不符，则会出现设置后协议修改标识（COM1 SD2543、COM2 SD2593）不会被清空的情况。这时协议不会被设置成功，还是会按照原来的协议来运行。

错误码

错误码	内容
4085H	(s) (n)读取地址超出软元件范围。
4084H	(n)不是 1。

示例

如上示例，在 RUN 过程中 M1 由 OFF→ON，

SD2592 会被设置为 32（20H），SD2593 会被设置为 90 之后会被清空。这时表示已经设置成功了。

SD2592	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	32
SD2593	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

如果再次将 M1 由 OFF→ON，

SD2592 会被设置为 32（20H），SD2593 会被设置为 90 但是不会被清空。

SD2592	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	32
SD2593	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	90

10.2 Modbus 串口参数设置

PORTPARA/Modbus 串口参数设置

PORTPARA(P)

对(n)号通讯口设置(s)串口参数。

-[PORTPARA (s) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	要设置的串口参数	-	无符号 BIN16 位	ANY16	INT
(n)	要设置的第几个通讯口，0 表示 COM1、1 表示 COM2	0、1	无符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
PORTPARA	参数 1												●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

功能

- 这条指令主要用于 RUN 的时候设置串口参数，也可以在 RUN 第一周期使用。
- 能直接将串口参数地址（COM1 SD2540、COM2 SD2590）以及对串口参数修改标识（COM1 SD2541、COM2 SD2591）的值根据指令参数进行设置。
- 串口参数修改标识（COM1 SD2541、COM2 SD2591）的设置具体的计算公式是：

(参数设置值+对应串口的起始特殊软元件的偏移+10)*2。

· 例如设置 COM1 串口参数为 193(HC1)就是(193+2541-2540+10)*2 也就是(193+1+10)*2=408。这时候会给 SD2540 设置 193(HC1)，SD2541 设置 408 ；

· 提示：无论是通过自己计算然后向串口参数修改标识（COM1 SD2541、COM2 SD2591）设置值，还是用这个指令来设置，都是能够在 PLC 处于 RUN 状态下修改串口参数的；

- 关于串口参数修改标识（COM1 SD2541、COM2 SD2591）：RUN 过程中，设置串口参数和修改标识正确时判断（END 时候判断）到正确的第一周期在修改完成时会将串口参数修改标识（COM1 SD2541、COM2 SD2591）清空，而后再设置相同的值将不作处理。主要为了避免多次的重复设置。

串口参数设置表

位编号	名称	内容	
		0（位 OFF）	1（位 ON）
b0	数据长度	7 位	8 位
b1 b2	奇偶校验	b2，b1 (0,0)：无 (0,1)：奇校验（ODD） (1,1)：偶校验（EVEN）	
b3	停止位	1 位	2 位
b4 b5 b6 b7 b8 b9	波特率（bps）	b9,b8,b7,b6,b5,b4 (0, 0, 0, 1, 1, 1)：4800 (0, 0, 1, 0, 0, 0)：9600 (0, 0, 1, 0, 0, 1)：19200 (0, 0, 1, 0, 1, 0)：38400 (0, 0, 1, 0, 1, 1)：57600	b9,b8,b7,b6,b5,b4 (0, 0, 1, 1, 0, 0)：115200 (0, 0, 1, 1, 0, 1)：187500 (0, 0, 1, 1, 1, 0)：230400 (0, 0, 1, 1, 1, 1)：460800 (0, 1, 0, 0, 0, 0)：921600

b10	STX	关闭 STX 功能	开启 STX 功能，具体值由 D8124 设置
b11	ETX	关闭 ETX 功能	开启 ETX 功能，具体值由 D8125 设置
b12	--	--	--
b13	--	--	--
b14	--	--	--
b15	--	--	--

相关的软元件

软元件	内容
SD2540	COM1 串口参数设置。
SD2541	COM1 串口参数修改标识。
SD2590	COM2 串口参数设置。
SD2591	COM2 串口参数修改标识。

注意事项

• 通讯参数的设置会影响整体的通讯，PLC 中的处理是在没有通讯情况下或者一轮通讯完成之后再进行修改。这一点需要注意。

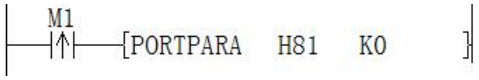
• STX 功能和 ETX 功能只有在自定义协议情况下才有用。

• 如果设置的协议与提供的协议编号不符，则会出现设置后串口参数修改标识（COM1 SD2541、COM2 SD2591）不会被清空的情况。这时协议不会被设置成功，还是会按照原来的协议来运行。

错误码

错误码	内容
4085H	(s)(n)读取地址超出软元件范围。
4084H	(n)不是 0 或者 1。

示例



如上示例

在 RUN 过程中 M1 由 OFF→ON

SD2540 会被设置为 129（H81，波特率：9600、停止位：1、数据位：8、校验位：无），SD2541 会被设置为 280 之后会被清空。这时已经设置成功了

SD2540	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	129
SD2541	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

如果再次将 M1 由 OFF→ON

SD2540 会被设置为 129（H81），SD2541 会被设置为 280 但是不会被清空。

SD2540	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	129
SD2541	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	280

10.3 Modbus 站号设置

STATION/Modbus 站号设置

STATION(P)

Modbus 从站协议下。对(n)号通讯口设置站号(s)。

-[STATION　(s)　(n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	要设置的站号	[0,255]	无符号 BIN16 位	ANY16	INT
(n)	要设置的第几个通讯口 0 表示 COM1、 1 表示 COM2	0、1	无符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
STATION	参数 1												●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

功能

- 这条指令主要用于 RUN 的时候设置站号，也可以在 RUN 第一周期使用。
- 能直接将站号地址（COM1 SD2544、COM2 SD2594）以及对站号修改标识（COM1 SD2545、COM2 SD2595）的值根据指令参数进行设置。
- 站号修改标识（COM1 SD2545、COM2 SD2595）的设置具体的计算公式是：

(参数设置值+对应串口的起始特殊软元件的偏移+10)*2。

· 例如设置 COM2 站号为 1 就是（1 +2595 - 2590 +10）*2 也就是（1 +5+10）*2=32 。这时候会给 SD2594 设 1，SD2595 设置 32；

· 提示：无论是通过自己计算然后向站号修改标识（COM1 SD2545、COM2 SD2595）设置值，还是用这个指令来设置，都是能够在 PLC 处于 RUN 状态下修改站号的。

- 关于站号修改标识（COM1 SD2545、COM2 SD2595）：RUN 过程中，设置站号和修改标识正确时判断（END 时候判断）到正确的第一周期在修改完成时会将站号修改标识（COM1 SD2545、COM2 SD2595）清空，而后再设置相同的值将不作处理。主要为了避免多次的重复设置。

相关的软元件

软元件	内容
SD2544	COM1 站号设置
SD2545	COM1 站号修改标识
SD2594	COM2 站号设置
SD2595	COM2 站号修改标识

错误码

错误码	内容
4085H	(s) (n)读取地址超出软元件范围。
4084H	(s)不在[0,255]范围内。
	(n)不是 0 或者 1。

注意事项

通讯参数的设置会影响整体的通讯，PLC 中的处理是在没有通讯情况下或者一轮通讯完成之后再进行修改。这一点需要注意。

示例

如上示例

在 RUN 过程中 M1 由 OFF→ON

SD2594 会被设置为 2，SD2595 会被设置为 34 之后会被清空。这时已经设置成功了。

SD2594	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
SD2595	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

如果再次将 M1 由 OFF→ON

SD2594 会被设置为 2，SD2595 会被设置为 34 但是不会被清空。

SD2594	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
SD2595	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	34

10.4 RS 指令

RS/外部通讯指令

RS

Modbus 主站协议情况下：该指令是主站发送协议帧的设置接口。根据指令设置的站号功能码(s)、从站地址(m)、长度(d)，以及功能码判断是否需要 n 的数据自动组合发送和接收协议帧。如果是读类型功能码会将数据写入到(n)中。

-[RS (s) (m) (d) (n)]

内容、范围、数据类型

自定义协议情况下：

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	为待发送数据存放的寄存器区的起始地址。	-	无符号 BIN16 位	ANY16	--
(m)	为待发送数数据的长度（字节数）。	[0,523]	无符号 BIN16 位	ANY16	--
(d)	存储写入的数据的软元件起始编号。	-	无符号 BIN16 位	ANY16	--
(n)	写入数据数量（字节数）。	[0,523]	无符号 BIN16 位	ANY16	--

Modbus 协议情况下：

参数	内容	范围	数据类型	数据类型(标签)
(s)	高字节存储从站的站号，低字节存储 Modbus 功能码。	-	无符号 BIN16 位	ANY16
(m)	从站地址。由从站提供出来的地址，会从从站的这个地址中读取或者写入数据。	-	无符号 BIN16 位	ANY16
(d)	长度，Modbus 读取或者写入的长度，单位由功能码来确定。	-	无符号 BIN16 位	ANY16
(n)	读取或写入数据的起始地址。	-	无符号 BIN16 位	ANY16

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RS	参数 1															●	●	●	●				●	●			
	参数 2															●	●	●	●				●	●			
	参数 3															●	●	●	●				●	●			
	参数 4															●	●	●	●				●	●			

(1)自定义协议

当通讯协议设置为自定义协议时。RS 指令前面的触点被开启同时 SM2591 也被置 ON 的时候，发送长度(m)非 0 值的情况下，就会将(s)的数据发送(m)个字节出去，然后进入等待接收模式。待数据接收回来时将会存储(n)个字节长度的数据到(d)中。接收到(n)个数据后就会将 SM2593 置 ON。

此外在发送长度(m)为 0 的时候就会处于只接收模式。在接收长度(n)为 0 的时候就会处于只发送模式。

开启 RS 指令的起始符（STX）和结束符（ETX）模式，必须设置特殊地址 SD2600 的第 10 位和第 11 位状态。设置如下：

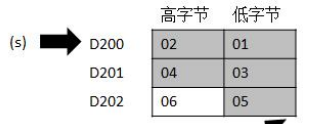
位编号	名称	内容	
		0（位 OFF）	1（位 ON）
b0	数据长度	7 位	8 位
b1 b2	奇偶校验	b2，b1 (0,0)：无 (0,1)：奇校验（ODD） (1,1)：偶校验（EVEN）	
b3	停止位	1 位	2 位
b4 b5 b6	波特率（bps）	b9,b8,b7,b6,b5,b4 （0，0，0，1，1，1）：4800 （0，0，1，0，0，0）：9600	b9,b8,b7,b6,b5,b4 （0，0，1，1，0，0）：115200 （0，0，1，1，0，1）：187500

福州富昌维控电子科技有限公司

b7 b8 b9		(0, 0, 1, 0, 0, 1)：19200 (0, 0, 1, 0, 1, 0)：38400 (0, 0, 1, 0, 1, 1)：57600	(0, 0, 1, 1, 1, 0)：230400 (0, 0, 1, 1, 1, 1)：460800 (0, 1, 0, 0, 0, 0)：921600
b10	STX	关闭 STX 功能	开启 STX 功能，具体值由 D8124 设置
b11	ETX	关闭 ETX 功能	开启 ETX 功能，具体值由 D8125 设置
b12	--	--	--
b13	--	--	--
b14	--	--	--
b15	--	--	--

示例

当 M1 为 ON 时，指令执行后通讯的收发数据存放如下图：



如图所示发送的数据依次为：01H、02H、03H、04H、05H

深色背景使用的数量为发送的长度（m）

如图所示接收到的数据依次为:21H、22H、23H、24H

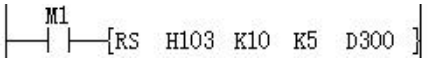
深色背景使用的数量为接收的长度（n）

实际编程时，需要做一些串行通讯的配置和准备，如设定串口的收发模式、波特率、位数、校验位、软件协议的设定、超时判断条件、收发缓冲区的数据准备、收发标志处理等，才能按预期的要求进行通讯。下面展示一个比较完整的 RS 通讯设置程序：

详细的自定义协议的使用说明请参阅本文档“9.5.1 自定义协议说明”。

⑵Modbus 协议

当协议设置成 Modbus 主站协议（不论是 RTU 还是 ASCII）的情况下。RS 指令前面的触点被置 ON 时。RS 指令会根据 站号功能码(s)、从站地址(m)、长度(d)并且根据功能码判断是否需要(n)的数据来进行组合数据帧发送。



参数	内容
(s)	高字节存储从站的站号，低字节存储功 Modbus 能码。
(m)	从站地址。由从站提供出来的地址，会从从站的这个地址中读取或者写入数据。
(d)	长度，Modbus 读取或者写入的长度。单位由功能码来确定。
(n)	读取或写入数据的起始地址。

福州富昌维控电子科技有限公司

如上所示梯形图：

当 M1 置 ON 后 PLC 将会从 PLC 的 COM2 发出数据（16 进制）：01 03 00 0A 00 05 A5 CB

01：代表从机地址，(s)的高 8 位；

03：Modbus 命令码，(s)的低 8 位，意义为读从机寄存器；

00 0A：所要读取的从机寄存器地址，(m)的值；

00 05：所要读取的寄存器个数，(d)的值，

A5 CB：CRC 校验码。

详细的自定义协议的使用说明请参阅本文档“9.5.2 Modbus 协议说明”。

注意事项

1. RS 指令虽然目前允许指令全部参数都可以用 K 和 H 这样的常量，但是根据协议的不同有不同的限制。
2. 当协议是自定义协议时 S 和 D 不能是常量，否则会报（3189H）错。
3. 当协议是 Modbus 协议时 n 不能是常量，否则会报（3189H）错。
4. 不能设置 RTU 协议和 7 位数据位组合。
5. 串口参数设置不同的情况下有可能会出现依然能通讯得上的情况，目前来看这个是正常情况。
6. 目前 RS 指令不能在中断和事件中使用。

错误码

错误码	内容
4085H	(s)(m)(d)(n)读取地址超出软元件范围。
3180H	COM2 数据接收错误。通讯线上可能有干扰，建议接上地线。
3181H	COM2 数据接收超时。检查接线，检查串口参数设置是否主从对应，检查是否有出现干扰。检查是否从站响应来不及导致的。建议尝试将发送间隔 SD2546 加大。
3182H	COM2 CRC 校验错误。通讯线上可能有干扰，建议接上地线。
3183H	COM2 LRC 校验错误。通讯线上可能有干扰，建议接上地线。
3184H	COM2 站号配置错误。检查从站站号设置，以及检查从站接收发送机制是否有问题。
3185H	COM2 发送缓冲区溢出。出现该错误提示，请联系技术人员。
3186H	COM2 功能码错误。检查设置的功能码是否是 PLC 支持的功能码。
3187H	COM2 地址错误。检查从站是否有这个地址（请查阅 Modbus 异常 02）
3188H	COM2 长度错误。检查通讯长度是否超出 Modbus 协议规定的长度范围，或是否超出自定义协议的规定长度范围。
3189H	COM2 数据错误。检查指令的参数是否有错误。检查从站是否支持这个值的设置。（请查阅 Modbus 异常 03）
318AH	COM2 从站忙。从站返回信息：从站忙（请查阅 Modbus 异常 06）
318BH	COM2 从站不支持功能码。检查从站是否支持这个功能码（请查阅 Modbus 异常 01）
318CH	COM2 从站故障。从站返回信息：从站故障，请检查从站是否故障（请查阅 Modbus 异常 04）
318DH	COM2 从站确认。从站返回信息：从站确认（请查阅 Modbus 异常 05）
318EH	COM2 目前不支持这个指令或功能。当设置为从站协议时不能使用主站相关的通讯指令或者主站功能。请更改协议或者关闭对应指令前的触点或者关闭对应的通讯功能。
318FH	COM2 发送超时。出现该错误提示，请联系技术人员。
31A0H	COM2 不可用网关。从站返回信息：不可用网关（请查阅 Modbus 异常 0A）
31A1H	COM2 指示没有从目标设备中获得响应。从站返回信息：设备未在网络中（请查阅 Modbus 异常 0B）

10.5 RS 指令对应协议使用说明

10.5.1 自定义协议说明

10.5.1.1 简介

自定义协议其功能为，可以不经过任何处理直接与设备传输数据或由客户自己设定对应的协议与其他设备通讯。

目前 6V PLC 的自定义协议是由 COM2 来配置发送，使用的指令是 RS 指令。需要通过软元件配置协议和串口参数。

目前的自定义协议大体上与 3V 的接近。

10.5.1.2 基础配置

(1)指令

RS 指令本身与以前 3V 时候的 RS 指令用法相同，且可以使用 R 软元件。

-[RS (s) (m) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	为待发送数据存放的寄存器区的起始地址	-	无符号 BIN16 位	ANY16	--
(m)	为待发送数数据的长度（字节数）	[0,528]	无符号 BIN16 位	ANY16	--
(d)	存储写入的数据的软元件起始编号	-	无符号 BIN16 位	ANY16	--
(n)	写入数据数量	[0,528]	无符号 BIN16 位	ANY16	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RS	参数 1															●	●	●	●				●	●			
	参数 2															●	●	●	●				●	●			
	参数 3															●	●	●	●				●	●			
	参数 4															●	●	●	●				●	●			

注意：RS 指令虽然目前允许指令全部参数都可以用 K 和 H 这样的常量，但是根据协议的不同有不同的限制。

当协议是自定义协议时，S 和 D 不能是常量，否则会报错。

(2)特殊软元件设置

特殊地址表

COM2 特殊 D 软元件（SD）		COM2 特殊 M 软元件（SM）	
2590	通讯口设置	2590	发送控制开启
2591	串口参数修改标识	2591	发送控制/发送提示
2592	协议设置	2592	接收控制开启
2593	协议修改标识	2593	接受控制/接收提示
2594	站号设置	2594	8 位模式 （自定义协议用）
2595	站号修改标识	2595	
2596	发送间隔 （0.1ms） 0~32767 设置 0 时为 10（1ms）	2596	
2597	通讯超时设置 （10ms） 0~32767 设置 0 时为 10（100ms）	2597	
2598	超时重试次数 0~32767	2598	
2599	字符间隔超时设置 （自定义协议用）（0.1ms） 0~32767 设置 0 时为 10（1ms）	2599	
2600	STX 值	2600	

福州富昌维控电子科技有限公司

2601	**ETX 值**	2601	
...		...	
2610	**接收到的数据量**	2610	**通讯完成标志**
2611	**最后一次错误**	2611	**接收中标识**
2612	**当前错误**	2612	**发生重试**
2613	**出错步数**	2613	**通讯错误**
2614	**出错站号**	2614	**通讯超时**
2615	**出错累计次数**	2615	

这里自定义协议会使用到并且要设置的内容已经用“下划线”和“加粗”标记出来了。

另外作为判断条件要使用的软元件也已经在表格中用“加粗”标记出来了。

注：COM1 目前不支持自定义协议。

①通讯口设置 SD2590

位编号	名称	内容	
		0（位 OFF）	1（位 ON）
b0	数据长度	7 位	8 位
b1 b2	奇偶校验	b2，b1 (0,0)：无 (0,1)：奇校验（ODD） (1,1)：偶校验（EVEN）	
b3	停止位	1 位	2 位
b4 b5 b6 b7 b8 b9	波特率（bps）	b9,b8,b7,b6,b5,b4 (0, 0, 0, 1, 1, 1)：4800 (0, 0, 1, 0, 0, 0)：9600 (0, 0, 1, 0, 0, 1)：19200 (0, 0, 1, 0, 1, 0)：38400 (0, 0, 1, 0, 1, 1)：57600	b9,b8,b7,b6,b5,b4 (0, 0, 1, 1, 0, 0)：115200 (0, 0, 1, 1, 0, 1)：187500 (0, 0, 1, 1, 1, 0)：230400 (0, 0, 1, 1, 1, 1)：460800 (0, 1, 0, 0, 0, 0)：921600
b10	STX	关闭 STX 功能	开启 STX 功能，具体值由 D8124 设置
b11	ETX	关闭 ETX 功能	开启 ETX 功能，具体值由 D8125 设置

根据上表提供的位设置来确定要设置的串口参数设置到 SD2590，目前设置大体与 3V 相同主要由于波特率可以设置更高的波特率的情况下 STX 和 ETX 的开启使用的位需要后移两位。

例：要设置串口参数：波特率 115200、停止位 1、数据位 8、校验位无、开启 STX。

那么要在 SD2590 上设置值 H4C1（K1217），目前直接设置到 SD2590 的参数只有在 PLC RUN 第一周期有效，而需要在 RUN 时候修改的话可以使用 PORTPARAM 指令进行设置。

②协议设置 SD2592

协议设置		协议设置	
0 H	Wecon Modbus 从站	10 H	自定义协议
2 H	Modbus RTU 从站	20 H	Modbus RTU 主站
3 H	Modbus ASCII 从站	30 H	Modbus ASCII 主站

通过在 SD2592 中设置相应的数值即可设定对应的协议。

目前直接设置到 SD2592 的参数只有在 PLC RUN 第一周期有效。

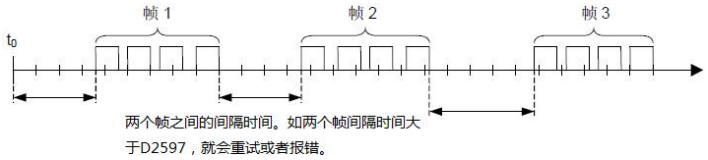
而需要在 RUN 时候修改的话可以使用 PROTOCOL 指令进行设置。

③发送间隔 SD2596

发送间隔主要功能是：当一条指令完成之后等待多长时间发送下一条指令。设置 0 的情况下基本不会等待间隔发送，但会受扫描周期影响。

发送间隔的单位是 0.1ms 也就是设置 100 情况下间隔时间是 10ms。

④通讯超时 SD2597 与超时重试 SD2598



通讯超时主要功能是：当 PLC 发送过后等待多长时间没有接收到数据要重试或者报错。设置 0 的情况下默认 100ms。

通讯超时的单位是 10ms，也就是设置 10 的情况下超时判断时间是 100ms。

当出现接收超时时，会判断是否有重试次数以及当前的重试次数，如果重试次数大于等于 SD2598 则会报错。

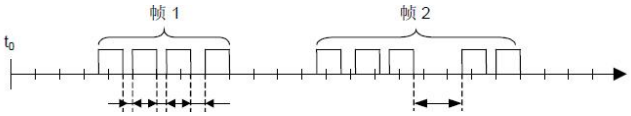
SD2598 设为 0 则不重试。

设为 1 情况下发送一次超时后再发送一次。

⑤字符间隔超时 SD2599

目前这个设置是自定义协议才有的。

字符间隔超时主要功能是：在未通讯超时前接收到至少一个字符后，判断之后的两个字符之间的间隔时间是否超出，超出情况下报错，并结束通讯。



这个设置主要是考虑部分设备可能发送较慢或者有其他特定发送需求才做的。字符间隔超时不会重试。

⑶串口参数设置指令

①上位机界面设置

通过上位机界面中的 PLC 参数设置，协议切换为无协议就是自定义协议。

串口参数内容主要设置波特率、数据位等串口参数。

起始字符启用后能够发送起始/结束字符。接收时也会区分起始/结束字符。详细内容见发送接收流程中的说明。

需要注意的一点是这里的起始字符是 10 进制的数值。

②PORTPARA 指令

-[PORTPARA (s) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	要设置的串口参数	[0,256]	无符号 BIN16 位	ANY16	--
(n)	要设置的第几个通讯口 0 表示 COM1、1 表示 COM2	0、1	无符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
PORTPARA	参数 1												●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

功能：用于 RUN 的时候设置通讯参数。

详细请参阅“PORTPARA 指令说明”。

③PROTOCOL 指令

-[PROTOCOL　(s)　(n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	要设置的协议编号	[0,65535]	无符号 BIN16 位	ANY16	--
(n)	要设置的第几个通讯口 0 表示 COM1、1 表示 COM2*1	1	无符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
PROTOCOL	参数 1												●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

功能

用于 RUN 的时候设置通讯协议，以上指令实际是对参数修改标识设定特定的值。

具体的计算公式是：(参数设置值+对应串口的起始特殊软元件的偏移+10)*2

例如设置协议为 Modbus RTU 主站（H20）就是（0x20 +2593 - 2590 +10）*2 也就是 （32 +3+10） *2=90，这样做的目的主要是为了避免中途串口参数被随意修改。

自定义协议情况下设置参数的触发是要在一条指令完全完成后才会改变数值。

Modbus 主站协议情况下设置参数的触发是要在一条指令完全完成后才会改变数值。

Modbus 从站协议情况下设置参数的触发只要没有在处理接收数据的时候就能够切换。

详细请参阅“PROTOCOL 指令说明”。

④串口参数的优先级说明

以下列出串口参数设置的优先级，目前串口参数是由掉电保存的。

串口参数设置指令设置 = 梯形图 MOV 指令设置对应的 SD 软元件〉上位机下载参数〉之前的断电保存数据。

⑷基础梯形图

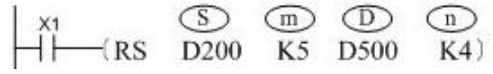
10.5.1.3 发送接收流程

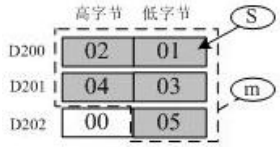
⑴发送接收机制

①无起始符（STX） 结束符（ETX）情况

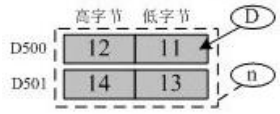
• 16 位情况（SM2594 为 OFF）

会将软元件分为高低直接然后由低地址先发送再发送高地址的顺序发数据。





例如发送的数据为：01、02、03、04、05

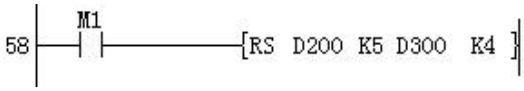


当 X1 为 ON 时，指令执行后通讯的收发数据存放如图。

例如接收的数据为：11、12、13、14

• 8 位情况（SM2594 为 ON）

直接获取软元件低地址的值然后发送



软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D200	0	0	0	0	1	1	1	0	1	0	1	1	1	1	1	0	7D70
D201	1	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	00E1
D202	1	0	0	0	1	1	0	0	1	0	1	0	1	0	0	0	1531
D203	0	1	0	0	0	1	1	0	0	0	0	1	1	0	0	0	1862
D204	0	0	1	1	0	1	1	0	0	1	1	1	1	0	0	0	1E6C
D205	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000
D206	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

如以上梯形图和以上数据情况下发送的数据应该是：70 E1 31 62 6C。

D300	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0002
D301	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0096
D302	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0042
D303	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0053
D304	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

接收也都会存放在低地址。

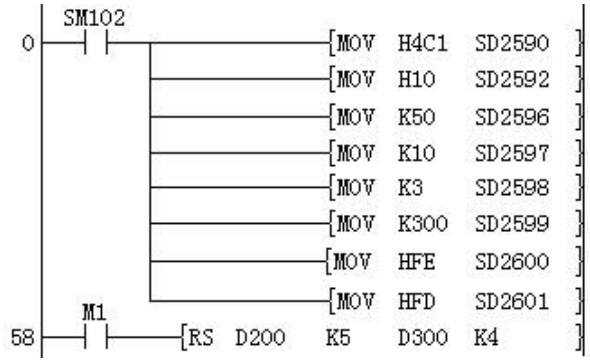
②打开并设置起始符 STX 情况

起始符 SD2600 的值只有低位有效。

开启 STX 时发送会在开头带有 STX，接收时以 STX 为开始进行接收但是不显示 STX。

在接收到一半有接收到 STX 则会重新开始接收。

没有收到第一个字符时超时时间以第一个字符超时判断超时，收到一个字符，无论收到的是什么数据都会从收到的最后一个字符开始算字符间超时。



软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D200	0	0	0	0	1	1	1	0	1	0	1	1	1	1	1	0	7D70
D201	1	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	00E1
D202	1	0	0	0	1	1	0	0	1	0	1	0	1	0	0	0	1531
D203	0	1	0	0	0	1	1	0	0	0	0	1	1	0	0	0	1862
D204	0	0	1	1	0	1	1	0	0	1	1	1	1	0	0	0	1E6C
D205	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000
D206	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

如以上梯形图和以上数据发送的结果是 FE 70 7D E1 00 31 。

这时给 PLC 发送 FE 51 26 34 15 将会收到以下数据。

而如果发送 FE 25 63 FE 51 26 34 15 也会收到以下数据。

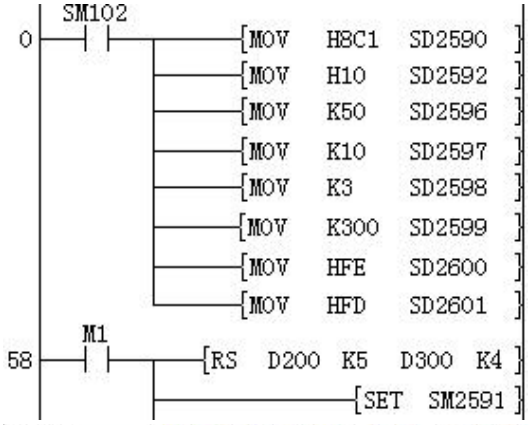
软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D300	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	0	2651
D301	0	0	1	0	1	1	0	0	1	0	1	0	1	0	0	0	1534

③打开并设置结束符 ETX 情况

结束符 SD2601 的值只有低位有效。

开启 ETX 是发送的结尾会带上 ETX。

当收到 ETX 时 SM2610 和 SM2593 立刻置 ON，无论是否收到足够的长度。



软元件	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
D200	0	0	0	0	1	1	1	0	1	0	1	1	1	1	1	0	7D70
D201	1	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	00E1
D202	1	0	0	0	1	1	0	0	1	0	1	0	1	0	0	0	1531
D203	0	1	0	0	0	1	1	0	0	0	0	1	1	0	0	0	1862
D204	0	0	1	1	0	1	1	0	0	1	1	1	1	0	0	0	1E6C

如以上梯形图和以上数据发送的结果是 70 7D E1 00 31 FD。

如以上梯形图给他发送 01 06 FD 会接收到以下数据。如果直接发送 FD 会直接判断为结束。

D300	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0601
D301	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000
D302	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000
D303	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000

④起始符 STX 和结束符 ETX 都被打开

都被打开的情况下大体上是上述单个打开状态的结合。

但如果都打开时只接收到 ETX 数据的话不会立刻结束，会转为判断字符间隔时间并且没有开始接收数据。

⑵发送后接收（类似 Modbus 主站机制）

①发送控制开关（SM2590）和接收控制开关（SM2592）OFF（默认）的情况

当 RS 指令的触点被置 ON 后，将 SM2591 置 ON 就立刻发送，发送完成后会立刻自动将 SM2591 置 OFF。

发送完成后判断 SM2593，如果是 ON 状态，则会一直停止在这条指令，等待被置 OFF 然后接收数据；如果是 OFF 状态会开始接收数据。

经过首字符超时时间（SD2597）没有收到字符判断为超时，如果有设置重试次数会进行重试。但重试过后依然通讯不上报错，同时 SM2593 和 SM2614 置 on。

收到第一个字符后，判断字符间的间隔时间是否超过 SD2599 所设置的时间，否则就会报错。（详细相关内容在介绍“字符间隔超时”有说明。）

有两条或者多条指令的情况下，SM2591 置 ON 后会发送第一个置 ON 的 RS 指令，之后如果这一条 RS 指令完成[*1]，再置 ON 后，SM2591 会切换成梯形图顺序中下一条 RS 指令[*2]。

有两条或者多条指令的情况下，在数据接收到一半时换成另外一条发送，继续停留在原来的指令进行接收直到完成。

注*1：完成指的是接收到数据或者报错。

注*2：如果梯形图顺序之后没有其他开启的 RS 指令，会回到梯形图顺序第一条被开启的 RS 指令进行执行。

②发送控制开关（SM2590）ON 情况下的发送方式

当发送控制开关（SM2590）ON 时，发送就不需要再控制 SM2591 置 ON。

这时候只要 RS 指令的触点被触发即可发送，同时自动将 SM2591 置 ON。

③接收控制开关（SM2592）ON 情况下的接收方式

当接收控制开关（SM2592）ON 时，会在指令执行时自动将 SM2593 转为 OFF 状态。

接收时也不会再判断 SM2593 的状态，接收后依然会将 SM2593 置 ON。

④发送控制开关（SM2590）和接收控制开关（SM2592）ON 情况

综合上述这种情况下，只需要将 RS 指令前面的触点触发即可发送数据，会自动转为等待接收状态，接收完成就会转向下一条被开启的 RS 指令。

⑶只发送

当发送长度设为非 0 值，接收长度设为 0 。此时就是只发送模式。

①送控制开关（SM2590）置 OFF 情况

SM2591 置 ON 就发送数据，不接收。

有两条或者多条指令的情况下，SM2591 置 ON 后，会发送第一个置 ON 的 RS 指令，如果这一条 RS 指令完成，再置 ON SM2591 会切换成梯形图中下一条 RS 指令。

②发送控制开关（SM2590）ON 情况

RS 指令被触发就会发送。

多条指令被触发会循环发送，循环发送期间会根据设定的发送间隔进行间隔发送。

⑷只接收

当发送长度为 0 接收长度非 0 时候为只接收。

①SM2592 接收控制开关 OFF 情况

接收时 SM2593 就会置 ON，需要梯形图控制置 OFF 后才能继续接收。没有收到数据 SM2593 就不会置 ON。

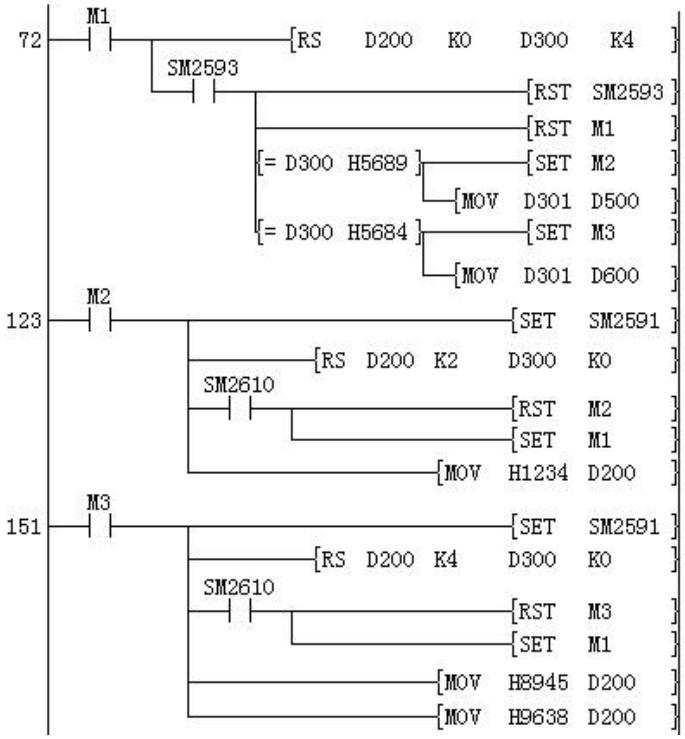
有多条指令同时开启时，只接收的接收位置是在第一条开启的指令先接收，收完或者超时后 SM2593 置 ON，再将 SM2593 置 OFF 会循环接收。

②SM2592 接收控制开关 ON 情况：

能够正常接收不会受到标志位的控制。SM2593 接收标识作为接收提示标识，没有收到数据 SM2593 就不会置 ON。

多条 RS 只接收指令同时开启，会由于超时判断切换到下一条，这一点导致接收位置不确定。

⑸用只接收和只发送方式作出接收后发送（类似 Modbus 从站机制）



只接收：当判断为接收完成的情况时，根据收到的内容进行判断要触发的指令，同时将接收到的数据保存。

只发送：M2 被触发后和 M3 被触发发送的内容不同。他们在触发结束后又会返回去触发 M1 的只接收部分。

10.5.1.4 错误信息

目前错误码会在 SD7 和 SD0 以及 SD2611 和 SD2612 上进行显示。

自定义协议会出现的错误码主要有以下这些

错误码	内容
3181H	数据接收超时。
3188H	长度错误。
3189H	COM2 数据错误。“检查指令的参数是否有错误。检查从站是否支持这个值的设置。（请查阅 Modbus 异常 03）”
318EH	COM2 目前不支持这个指令或功能。当设置为从站协议时不能使用主站相关的通讯指令或者主站功能。请更改协议或者关闭对应指令前的触点或者关闭对应的通讯功能。
318FH	COM2 发送超时。出现该错误提示，请联系技术人员。
3190H	COM2 接收数据超出缓存限制。

在重置协议或者通讯参数后会清空报错。

另外通讯完成/通讯错误/通讯超时，标志位的置位会在被执行的指令之后进行置位。

10.5.1.5 与三菱的差别

目前已知的差异：

⑴STX 和 ETX，三菱可以设置最大 4 个字节，我们只有一个字节。

⑵能够加上和校验以及 CR、LF。

⑶三菱的指令控制是最先被触发的指令就固定执行这一条指令。中途不会被切换成其他指令，除非这条指令的触点被关闭。

10.5.2 Modbus 协议说明

10.5.2.1 简介

目前的 Modbus 主站协议大体上与 LX3V 的接近。从站部分对地址进行了修改。

10.5.2.2 基础配置

(1)指令

RS 指令本身与以前 3V 时候的 RS 指令用法相同。且可以使用 R 软元件。

-[RS (s) (m) (d) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)
(s)	高字节存储从站的站号，低字节存储功 Modbus 能码。	-	无符号 BIN16 位	ANY16
(m)	从站地址。由从站提供出来的地址，会从从站的这个地址中读取或者写入数据。	-	无符号 BIN16 位	ANY16
(d)	长度，Modbus 读取或者写入的长度。单位由功能码来确定。	-	无符号 BIN16 位	ANY16
(n)	读取或写入数据的起始地址。	-	无符号 BIN16 位	ANY16

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RS	参数 1															●	●	●	●				●	●			
	参数 2															●	●	●	●				●	●			
	参数 3															●	●	●	●				●	●			
	参数 4															●	●	●	●								

注意：RS 指令虽然目前允许指令全部参数都可以用 K 和 H 这样的常量。

但是根据协议的不同有不同的限制。当协议是 Modbus 主站协议时 n 不能是常量，否则会报错。

(2)特殊软元件设置

特殊地址表

COM1 特殊 D 软元件（SD）	
2540	通讯口设置
2541	串口参数修改标识
2542	
2543	
2544	站号设置
2545	站号修改标识

*注 COM1 不支持修改协议。

COM2 特殊 D 软元件（SD）		COM2 特殊 M 软元件（SM）	
2590	通讯口设置	2590	发送控制开启
2591	串口参数修改标识	2591	发送控制/发送提示
2592	协议设置	2592	接收控制开启
2593	协议修改标识	2593	接受控制/接收提示
2594	站号设置	2594	8 位模式 （自定义协议用）
2595	站号修改标识	2595	
2596	发送间隔 （0.1ms） 0-32767 设置 0 时为 10（1ms）	2596	
2597	通讯超时设置 （10ms） 0-32767 设置 0 时为 10（100ms）	2597	

福州富昌维控电子科技有限公司

2598	**超时重试次数　0-32767**	2598	
2599	字符间隔超时设置 （自定义协议用）（0.1ms） 0-32767 设置 0 时为 10（1ms）	2599	
2600	**STX 值**	2600	
2601	**ETX 值**	2601	
...		..	
2610	**接收到的数据量**	2610	**通讯完成标志**
2611	**最后一次错误**	2611	**接收中标识**
2612	**当前错误**	2612	**发生重试**
2613	**出错步数**	2613	**通讯错误**
2614	**出错站号**	2614	**通讯超时**
2615	**出错累计次数**	2615	

这里 Modbus 协议会使用到并且要设置的内容已经用 “下划线” 和 “加粗” 标记出来了。

另外作为判断条件要使用的软元件也已经在表格中用“加粗”标记出来了。

①通讯口设置 SD2590

位编号	名称	内容	
		0（位 OFF）	1（位 ON）
b0	数据长度	7 位	8 位
b1 b2	奇偶校验	b2，b1 (0,0)：无 (0,1)：奇校验（ODD） (1,1)：偶校验（EVEN）	
b3	停止位	1 位	2 位
b4 b5 b6 b7 b8 b9	波特率（bps）	b9,b8,b7,b6,b5,b4 (0, 0, 0, 1, 1, 1)：4800 (0, 0, 1, 0, 0, 0)：9600 (0, 0, 1, 0, 0, 1)：19200 (0, 0, 1, 0, 1, 0)：38400 (0, 0, 1, 0, 1, 1)：57600	b9,b8,b7,b6,b5,b4 (0, 0, 1, 1, 0, 0)：115200 (0, 0, 1, 1, 0, 1)：187500 (0, 0, 1, 1, 1, 0)：230400 (0, 0, 1, 1, 1, 1)：460800 (0, 1, 0, 0, 0, 0)：921600

根据上表提供的位设置来确定要设置的串口参数设置到 SD2590。

目前设置大体与 3V 相同主要由于波特率可以设置更高的波特率的情况下 STX 和 ETX 的开启使用的位需要后移两位。

例：要设置串口参数：波特率 115200、停止位 1、数据位 8、校验位无。

那么要在 SD2590 上设置值 HC1（K1217）。

目前直接设置到 SD2590 的参数只有在 PLC RUN 第一周期有效。

而需要在 RUN 时候修改的话可以使用 PORTPARAM 指令进行设置。

②协议设置 SD2592

协议设置			
0H	Wecon Modbus 从站	10H	自定义协议
2H	Modbus RTU 从站	20H	Modbus RTU 主站
3H	Modbus ASCII 从站	30H	Modbus ASCII 主站

通过在 SD2592 中设置相应的数值即可设定对应的协议。

目前直接设置到 SD2592 的参数只有在 PLC RUN 第一周期有效。

而需要在 RUN 时候修改的话可以使用 PROTOCOL 指令进行设置。

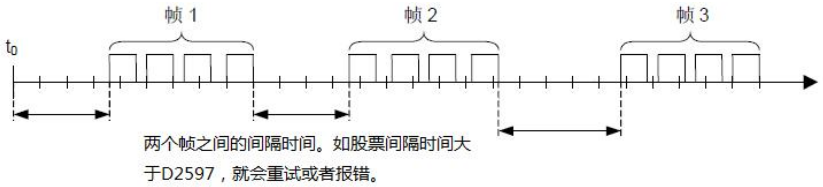
目前 COM1 不能使用除 Wecon Modbus 从站以外的协议。

③发送间隔 SD2596

发送间隔主要功能是：当一条指令完成之后等待多长时间发送下一条指令。设置 0 的情况下基本不会等待间隔发送，但会受扫描周期影响。

发送间隔的单位是 0.1ms 也就是设置 100 情况下间隔时间是 10ms。

④通讯超时 SD2597 与超时重试 SD2598



通讯超时主要功能是：当 PLC 发送过后等待多长时间没有接收到数据要重试或者报错。设置 0 的情况下默认 100ms。

通讯超时的单位是 10ms，也就是设置 10 的情况下超时判断时间是 100ms。

当出现接收超时时，会判断是否有重试次数以及当前的重试次数，如果重试次数大于等于 SD2598 则会报错。

SD2598 设为 0 则不重试。

设为 1 情况下发送一次超时后再发送一次。

(3)串口参数设置

① PORTPARA 指令

-[PORTPARA (s) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	要设置的串口参数[0,256]	-	无符号BIN16位	ANY16	INT
(n)	要设置的第几个通讯口，0 表示 COM1、1 表示 COM2	[-214748364,2147483647]	无符号BIN16位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
PORTPARA	参数 1												●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

功能

用于 RUN 的时候设置通讯参数。

详细请参阅“PORTPARA 指令说明”。

② PROTOCOL 指令

-[PROTOCOL (s) (n)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	要设置的协议编号	[0,65535]	无符号 BIN16 位	ANY16	INT
(n)	要设置的第几个通讯口,0 表示 COM1、1 表示 COM2	1	无符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
PROTOCOL	参数 1												●	●	●	●	●	●	●	●			●	●		●	●
	参数 2												●	●	●	●	●	●	●	●			●	●		●	●

功能

用于 RUN 的时候设置通讯协议，以上指令实际是对参数修改标识设定特定的值。

具体的计算公式是：(参数设置值+对应串口的起始特殊软元件的偏移+10)*2

例如设置协议为 Modbus RTU 主站 （H20）就是（0x20 +2593 - 2590 +10）*2 也就是 （32 +3+10） *2=90，这样做的目的主要是为了避免中途串口参数被随意修改。

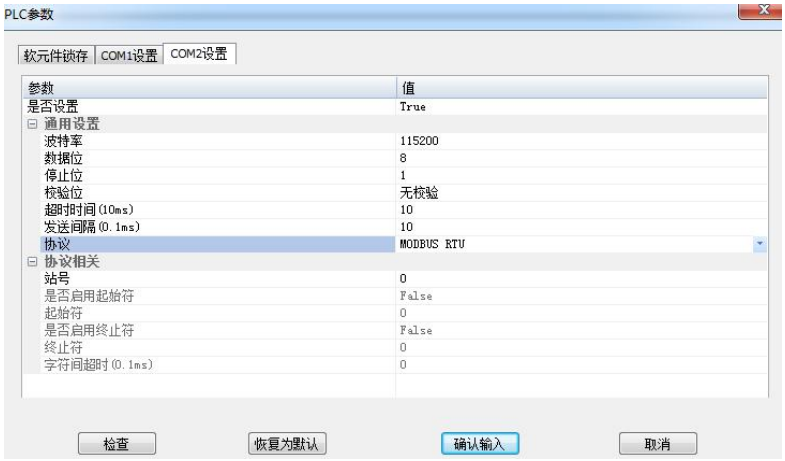
Modbus 主站协议情况下设置参数的触发是要在一条指令完全完成后才会改变数值。

Modbus 从站协议情况下设置参数的触发只要没有在处理接收数据的时候就能够切换。

详细请参阅“PROTOCOL 指令说明”。

③上位机设置

上位机的 PLC 参数设置可以设置对应的串口参数.



专用协议站号不能为 0

Modbus RTU 和 Modbus ASCII 协议下站号为 0：协议设置主站。

Modbus RTU 和 Modbus ASCII 协议下站号为非 0：协议设置从站。

串口参数按照表格中的内容进行填写。

注意：RTU 协议不能设置 7 位数据位。

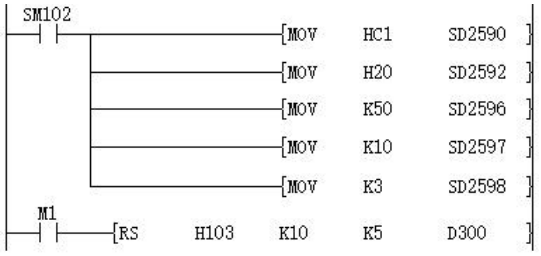
④串口参数的优先级说明

以下列出串口参数设置的优先级

目前串口参数是由掉电保存的。

串口参数设置指令设置 = 梯形图 MOV 指令设置对应的 SD 软元件 〉 上位机下载参数 〉之前的断电保存数据。

(4)基础梯形图



串口参数设置
波特率 115200、数据位 8、校验位无、停止位 1
通讯协议设置：Modbus RTU 主站协议
发送间隔：5ms
接收超时：100ms
超时重试次数：3 次
站号 01H、功能码 03H、从站地址 10、长度 05

10.5.2.3 发送接收流程

(1)Modbus 主站

编程时，在每个 RS（Modbus 模式）指令的前面，要根据要进行的通讯操作对象地址、操作类型、操作寄存器地址、数据个数、发送或接收的单元等各操作数单元赋值完毕，一旦开始执行，系统程序会自动计算 CRC 校验，组织通讯帧，完成发送数据、接收应答操作。

若使用 Modbus-ASC 协议通讯（将 SD2592 设定为 H30），其中收发数据的 HEX-ASC 格式变换由 PLC 系统程序自动完成，用户使用 RS（Modbus 模式）指令的方法与使用 Modbus-RTU 协议的方法完全相同。

在 plc 程序中，若有多个 RS（Modbus 模式）指令被驱动，系统程序在执行时，仍是将一个 RS 指令的“发送、等待回答、接收、校验解析存放”等环节进行完毕后，再对下一个 RS 指令的同样处理，直到所有 RS 指令执行完毕，重新开始，用户无需关心其执行的时序和过程，简化了 PLC 编程设计，这是 Modbus 指令的优点所在。

Modbus 主站功能一览表

功能码	功能名	详细内容
0x01	线圈读出	线圈读出（可以多点）
0x02	输入读出	输入读出（可以多点）
0x03	保持寄存器读出	保持寄存器读出（可以多点）
0x04	输入寄存器读出	输入寄存器读出（可以多点）
0x05	1 线圈写入	线圈写入（仅 1 点）
0x06	1 寄存器写入	保持寄存器写入（仅 1 点）
0x0F	批量线圈写入	多点的线圈写入
0x10	批量寄存器写入	多点的保持寄存器写入

示例：

如上所示梯形图：

当 M1 置 ON 后 PLC 将会从 PLC 的 COM2 发出数据（16 进制）：01 03 00 0A 00 05 A5 CB

01：代表从机地址，(s)的高 8 位；

03：Modbus 命令码，(s)的低 8 位，意义为读从机寄存器；

00 0A ：所要读取的从机寄存器地址，(m)的值；

00 05 ：所要读取的寄存器个数， (d)的值，

A5 CB ：CRC 校验码。

注意：

使用 1 线圈写入（0x05）时，要写入 1 使用的值为 0xFF00 写入 0 的话使用的值为 0x0000。

使用批量线圈写入（0x0F）时，注意高低字节调换。例如写 1 个长度时候第四个参数值是 1（0x0001）是发送 0，需要写 256（0x0100）才会给目标地址写 1。

(2)Modbus 从站

plc 作为 Modbus 从站时，支持 Modbus 的 0x01，0x03，0x05，0x06，0x 0f，0x10 等通讯操作命令；通过这些命令，可读写 PLC 的线圈有 M，S，T，C，X（只读）SM，Y，LC，HSC 等变量；寄存器变量有 D，T，C，R，SD，LC，HSC。

Modbus 通讯主机在访问（读取或改写）PLC 从机的内部变量时，必须遵循如下的通讯命令帧定义，以及变量地址索引方法，才能进行正常的通讯操作。

①命令码 0x01（01）：读线圈

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x01（命令码）	1 个字节	读线圈
3	线圈起始地址	2 个字节	高位在前，低位在后，见线圈编址
4	线圈数量	2 个字节	高位在前，低位在后（N）
5	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x01+字节数+线圈状态+CRC 检验

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x01（命令码）	1 个字节	读线圈
3	字节数	1 个字节	值：[（N+7）/8]
4	线圈状态	[（N+7）/8] 个字节	每 8 个线圈合为一个字节，最后一个若不足 8 位，未定义部分填 0。前 8 个线圈在第一个字节，最地址最小的线圈在最低位。依次类推。
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧。

②命令码 0x03（03）：读寄存器

请求帧格式：从机地址+0x03+寄存器起始地址+寄存器数量+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x03（命令码）	1 个字节	读寄存器
3	寄存器起始地址	2 个字节	高位在前，低位在后，见寄存器编址
4	寄存器数量	2 个字节	高位在前，低位在后（N）
5	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x03+字节数+寄存器值+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x03（命令码）	1 个字节	读寄存器
3	字节数	1 个字节	值：N*2
4	寄存器值	N*2 个字节	每两字节表示一个寄存器值，高位在前低位在后。寄存器地址小的排在前面
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧。

③命令码 0x05（05）：写单线圈

请求帧格式：从机地址+0x05+线圈地址+线圈状态+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x05（命令码）	1 个字节	写单线圈
3	线圈地址	2 个字节	高位在前，低位在后，见线圈编址
4	线圈状态	2 个字节	高位在前，低位在后。非 0 即为有效
5	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x05+线圈地址+线圈状态+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x05（命令码）	1 个字节	写单线圈
3	线圈地址	2 个字节	高位在前，低位在后，见线圈编址
4	线圈状态	2 个字节	高位在前，低位在后。非 0 即为有效
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧。其中线圈状态 0xFF00 表示置 ON ，0x0000 表示置 OFF。

④命令码 0x06（06）：写单个寄存器

请求帧格式：从机地址+0x06+寄存器地址+寄存器值+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x06（命令码）	1 个字节	写单寄存器
3	寄存器地址	2 个字节	高位在前，低位在后，见寄存器值编址
4	寄存器值	2 个字节	高位在前，低位在后。非 0 即为有效
5	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x06+寄存器地址+寄存器值+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x06（命令码）	1 个字节	写单寄存器
3	寄存器地址	2 个字节	高位在前，低位在后，见寄存器编址
4	寄存器值	2 个字节	高位在前，低位在后。非 0 即为有效
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧。

⑤命令码 0x0f（15）：写多个线圈

请求帧格式：从机地址+0x 0f+线圈起始地址+线圈数量+字节数+线圈状态+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x 0f（命令码）	1 个字节	写多个单线圈
3	线圈起始地址	2 个字节	高位在前，低位在后，见线圈编址
4	线圈数量	2 个字节	高位在前，低位在后。N，最大为 1968
5	字节数	1 个字节	值：[（N+7）/8]
6	线圈状态	[（N+7）/8] 个字节	每 8 个线圈合为一个字节，最后一个若不足 8 位，未定义部分填 0。 前 8 个线圈在第一个字节，最地址最小的线圈在最低位，依次类推
7	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x 0f +线圈起始地址+线圈数量+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x 0f（命令码）	1 个字节	写多个单线圈
3	线圈起始地址	2 个字节	高位在前，低位在后，见线圈编址
4	线圈数量	2 个字节	高位在前，低位在后。
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧。

⑥命令码 0x10（16）：写多个寄存器

请求帧格式：从机地址+0x10+寄存器起始地址+寄存器数量+字节数+寄存器值+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x10（命令码）	1 个字节	写多个寄存器
3	寄存器起始地址	2 个字节	高位在前，低位在后，见寄存器编址
4	寄存器数量	2 个字节	高位在前，低位在后。N，最大为 120
5	字节数	1 个字节	值：N*2
6	寄存器值	N*2（N*4）	
7	CRC 校验	2 个字节	高位在前，低位在后

响应帧格式：从机地址+0x10+寄存器起始地址+寄存器数量+CRC 检验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	0x10（命令码）	1 个字节	写多个寄存器
3	寄存器起始地址	2 个字节	高位在前，低位在后，见寄存器编址
4	寄存器数量	2 个字节	高位在前，低位在后。N，最大为 120
5	CRC 校验	2 个字节	高位在前，低位在后

错误响应：见错误响应帧。

⑦错误响应帧

错误响应：从机地址+（命令码+0x80）+错误码+CRC 校验。

序号	数据(字节)意义	字节数量	说明
1	从机地址	1 个字节	取值 1~247，由 SD2544、SD2594 设定
2	命令码+0x80	1 个字节	错误命令码
3	错误码	1 个字节	1~4
4	CRC 校验	2 个字节	高位在前，低位在后

序号	错误码	说明
1	01	不支持的功能码
2	02	错误的地址或者功能码
3	03	长度错误
4	04	指令不完善
5	05	不允许的地址

⑧从站地址表

字地址				
地址类型	占用	地址范围	10 进制地址	总共预留地址大小
T0~T511	512 WORD	0x0000-0x01ff	0	1536
C0~C255	256 WORD	0x0600-0x06ff	1536	1024
LC0~LC255	512 WORD	0x0A00-0x0BFF	2560	1024
HSC0~HSC15	128 WORD	0x0E00-0x0E1F	3584	512
D0~D7999	8000 WORD	0x1000-0x2F3F	4096	16384
SD0~SD4095	4096 WORD	0x5000-0x5FFF	20480	12288
R0~R30000	30000 WORD	0x8000-0xF52F	32768	30000

位地址				
地址类型	占用	地址范围	10 进制地址	总共预留地址大小
T0~T511	512 bit	0x0000-0x01ff	0	1536
C0~C255	256 bit	0x0600-0x06ff	1536	1024
LC0~LC255	256 bit	0x0A00-0x0AFF	2560	1024
HSC0~HSC15	64 bit	0x0E00-0x0E0F	3584	512
M0~M8000	8192 bit	0x1000-0x2F3F	4096	16384
SM0~SM4095	4096 bit	0x5000-0x5FFF	20480	12288
预留		0x8000-0xBFFF		16383
S0~S4095	4096 bit	0xC000-0xCFFF	49152	8192
X0~X1023	1024 bit	0xE000-0xE3FF	57344	4096
Y0~Y1023	1024 bit	0xF000-0xF3FF	61440	4096

10.5.2.4 错误信息

目前错误码会在 SD7 和 SD0 以及 SD2611 和 SD2612 上进行显示。

Modbus 协议会出现的错误码主要有以下这些。

错误码	内容
4085H	(s)(m)(d)(n)读取地址超出软元件范围 （这个错误只在 SD7 和 SD0 上显示）
3180H	COM2 数据接收错误。通讯线上可能有干扰，建议接上地线。
3181H	COM2 数据接收超时。检查接线，检查串口参数设置是否主从对应，检查是否有出现干扰。检查是否是从站响应来不及导致的。建议尝试将发送间隔 SD2546 加大。
3182H	COM2 CRC 校验错误。通讯线上可能有干扰，建议接上地线。
3183H	COM2 LRC 校验错误。通讯线上可能有干扰，建议接上地线。
3184H	COM2 站号配置错误。检查从站站号设置，以及检查从站接收发送机制是否有问题。
3185H	COM2 发送缓冲区溢出。出现该错误提示，请联系技术人员。
3186H	COM2 功能码错误。检查设置的功能码是否是 PLC 支持的功能码。
3187H	COM2 地址错误。检查从站是否有这个地址（请查阅 Modbus 异常 02）
3188H	COM2 长度错误。检查通讯长度是否超出 Modbus 协议规定的长度范围，或是否超出自定义协议的规定长度范围。
3189H	COM2 数据错误。检查指令的参数是否有错误。检查从站是否支持这个值的设置。（请查阅 Modbus 异常 03）
318AH	COM2 从站忙。从站返回信息：从站忙（请查阅 Modbus 异常 06）
318BH	COM2 从站不支持功能码。检查从站是否支持这个功能码（请查阅 Modbus 异常 01）
318CH	COM2 从站故障。从站返回信息：从站故障，请检查从站是否故障（请查阅 Modbus 异常 04）
318DH	COM2 从站确认。从站返回信息：从站确认（请查阅 Modbus 异常 05）
318EH	COM2 目前不支持这个指令或功能。当设置为从站协议时不能使用主站相关的通讯指令或者主站功能。请更改协议或者关闭对应指令前的触点或者关闭对应的通讯功能。
318FH	COM2 发送超时。出现该错误提示，请联系技术人员。
3190H	COM2 接收数据超出缓存限制。
31A0H	COM2 不可用网关。从站返回信息：不可用网关（请查阅 Modbus 异常 0A）
31A1H	COM2 指示没有从目标设备中获得响应。从站返回信息：设备未在网络中（请查阅 Modbus 异常 0B）

在重置协议或者通讯参数后会清空报错。

另外通讯完成/通讯错误/通讯超时 标志位的置位会在被执行的指令之后进行置位。

10.6 WECON Modbus 协议说明

目前 WECON Modbus 协议说明（专用协议）是基于 Modbus RTU 协议进行修改的。

所以串口参数部分不能使用 7 位数据位。

这个协议完全兼容 Modbus RTU 协议，地址也与 LX6V PLC 的 Modbus RTU 协议默认地址相同。

扩展的功能主要是用于和 PLC Editor2 通讯。

11 PID 控制指令

PRUN/8 进制位传送（16 位数据）

PRUN(P)

将指定了位数的(s)与(d)的软元件编号处理为 8 进制数后，传送数据。

-[PRUN （s） （d）]

内容、范围、数据类型

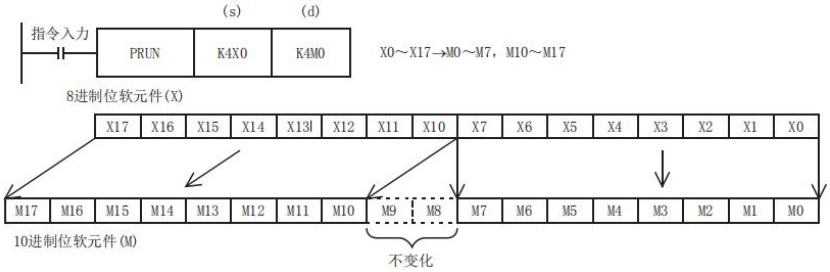
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	位数指定*1	-	BIN16 位	ANY16	--
(d)	传送目标软元件编号*1	-	BIN16 位	ANY16	--

使用的软元件

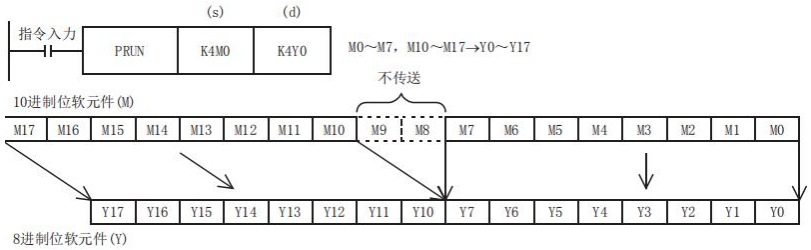
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
PRUN	参数 1											●		●											●	●
	参数 2												●	●											●	●

功能

- 8 进制位软元件→10 进制位软元件



- 10 进制位软元件→8 进制位软元件

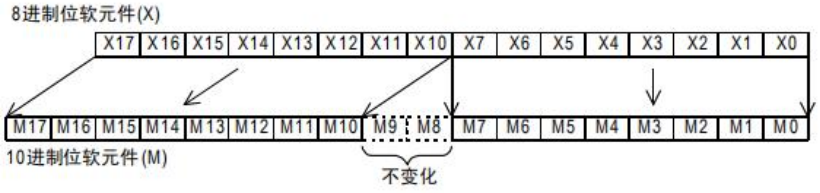


错误码

错误码	内容
4085H	读取指定的软元件范围超出相应软元件的范围时。
4086H	写入指定的软元件范围超出相应软元件的范围时。

示例

如以上梯形图所示：X0~X17 取八进制位的值，传给 M 对应位的软元件。



TRH/干湿球温度与湿度的换算

TRH

该指令完成干球温度、湿球温度与对应湿度的换算。

-[TRH（d1） （s） （d2） （n）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(d1)	湿度	[0,100]	单精度浮点型	ANYREAL_32	REAL
(s)	干球温度	-	单精度浮点型	ANYREAL_32	REAL
(d2)	湿球温度	-	单精度浮点型	ANYREAL_32	REAL
(n)	模式	[0,1]	有符号 BIN32 位	ANY32	DINT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TRH	参数 1															●	●	●	●	●						●	
	参数 2															●	●	●	●	●						●	
	参数 3															●	●	●	●	●						●	
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	

功能

• （n）中有两种模式可供选择：

模式 0：通过湿球温度、干球温度计算对应的湿度。

模式 1：通过干球温度、湿度计算对应的湿球温度。

• 换算过程公式如下：

假设湿球温度为 A，干球温度为 B，对应当前湿度为 C，三者满足以下条件：

$$EXP\{(A\times 17.27)/(A+237.36)\}\times 611=x \qquad (1)$$

$$EXP\{(B\times 17.27)/(B+237.36)\}\times 611=y \qquad (2)$$

$$z=x-C\times y/100 \qquad (3)$$

$$A=B-z/65.566 \qquad (4)$$

注意事项

• 湿球温度不大于干球温度，当两者相同时，湿度达到最大 100%。

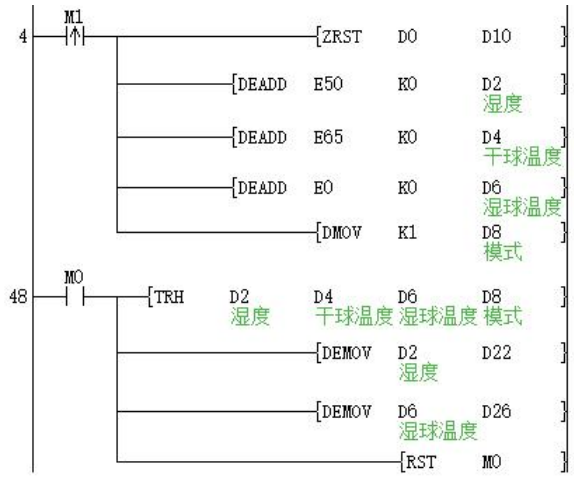
• 干湿球温度的单位为（℃）。

• 干球一般取值范围在 0~100℃之间，指令没有对其范围进行判别，因此在使用此指令时要特别注意。

错误码

错误码	内容
4084H	(n)中指定的数值超出下述范围。0~1
	(d1)中指定的数值超出以下范围。0~100
	(s)中指定了负值。
	(d2)中指定了负值。
4085H	读应用指令中(d1)(s)(d2)(n)输出结果超出软元件范围。
4086H	写应用指令中(d1)(d2)输出结果超出软元件范围。

示例



干湿球温度/湿度换算表

湿球温度 \ 湿度 / 干球温度	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
10.0	2.55	3.06	3.58	4.09	4.58	5.07	5.54	6.02	6.49	6.95	7.41	7.86	8.29	8.73	9.16	9.59	10.00
11.0	3.24	3.78	4.32	4.85	5.37	5.88	6.38	6.87	7.36	7.84	8.31	8.77	9.24	9.69	10.13	10.57	11.00
12.0	3.94	4.50	5.06	5.62	6.15	6.68	7.21	7.72	8.23	8.72	9.21	9.70	10.17	10.64	11.10	11.56	12.00
13.0	4.62	5.21	5.79	6.38	6.93	7.49	8.04	8.57	9.09	9.61	10.12	10.62	11.12	11.59	12.07	12.54	13.00
14.0	5.30	5.92	6.53	7.13	7.72	8.29	8.85	9.42	9.96	10.50	11.02	11.54	12.05	12.55	13.05	13.52	14.00
15.0	5.98	6.62	7.26	7.89	8.50	9.10	9.68	10.26	10.83	11.38	11.93	12.47	12.99	13.50	14.02	14.51	15.00
16.0	6.64	7.32	7.99	8.64	9.28	9.90	10.51	11.11	11.69	12.27	12.83	13.38	13.93	14.47	14.98	15.50	16.00
17.0	7.31	8.02	8.72	9.39	10.05	10.70	11.34	11.95	12.56	13.16	13.73	14.31	14.87	15.42	15.95	16.48	17.00
18.0	7.98	8.72	9.43	10.13	10.82	11.50	12.15	12.80	13.42	14.03	14.64	15.23	15.80	16.37	16.93	17.46	18.00
19.0	8.64	9.40	10.15	10.89	11.59	12.29	12.97	13.64	14.28	14.92	15.54	16.15	16.75	17.33	17.90	18.45	19.00
20.0	9.30	10.09	10.87	11.63	12.37	13.09	13.79	14.49	15.16	15.81	16.45	17.07	17.69	18.28	18.87	19.44	20.00
21.0	9.95	10.78	11.59	12.38	13.14	13.89	14.61	15.33	16.02	16.69	17.35	17.99	18.62	19.24	19.84	20.43	21.00
22.0	10.60	11.47	12.31	13.12	13.92	14.69	15.44	16.17	16.88	17.58	18.26	18.92	19.56	20.19	20.81	21.41	22.00
23.0	11.25	12.14	13.02	13.86	14.68	15.48	16.26	17.02	17.75	18.46	19.16	19.84	20.50	21.15	21.77	22.40	23.00
24.0	11.89	12.83	13.73	14.61	15.46	16.28	17.08	17.86	18.61	19.35	20.06	20.76	21.44	22.11	22.75	23.39	24.00
25.0	12.53	13.51	14.44	15.35	16.22	17.08	17.90	18.70	19.48	20.24	20.97	21.68	22.38	23.06	23.73	24.37	25.00
26.0	13.18	14.18	15.15	16.09	16.99	17.87	18.73	19.54	20.34	21.13	21.88	22.62	23.33	24.02	24.70	25.36	26.00
27.0	13.82	14.86	15.83	16.84	17.76	18.67	19.55	20.39	21.21	22.01	22.79	23.53	24.26	24.98	25.67	26.35	27.00
28.0	14.46	15.53	16.57	17.57	18.54	19.46	20.37	21.24	22.08	22.90	23.70	24.46	25.20	25.94	26.64	27.33	28.00
29.0	15.10	16.21	17.28	18.31	19.31	20.26	21.20	22.09	22.95	23.79	24.61	25.39	26.15	26.90	27.61	28.32	29.00
30.0	15.73	16.88	17.99	19.05	20.08	21.07	22.02	22.94	23.82	24.68	25.51	26.31	27.10	27.85	28.58	29.30	30.00
31.0	16.37	17.56	18.70	19.80	20.85	21.87	22.84	23.78	24.69	25.57	26.42	27.24	28.04	28.82	29.56	30.29	31.00
32.0	17.00	18.22	19.41	20.54	21.62	22.67	23.67	24.63	25.56	26.47	27.33	28.17	28.99	29.76	30.54	31.27	32.00
33.0	17.63	18.90	20.12	21.28	22.40	23.47	24.50	25.48	26.43	27.35	28.24	29.10	29.93	30.73	31.51	32.27	33.00
34.0	18.26	19.58	20.83	22.02	23.18	24.28	25.32	26.33	27.31	28.25	29.15	30.03	30.87	31.69	32.49	33.25	34.00

12 字符串指令

LEN/字符串的长度检测

LEN(P)

检测(s)中指定的字符串的长度后，存储到(d)中指定的软元件编号以后。

将(s)中指定的软元件编号开始至存储了 00H 的软元件编号为止的数据作为字符串处理。

-[LEN （s） （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	字符串或存储了字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	存储检测的字符串长度的软元件编号	-	有符号 BIN16 位	ANY16	INT

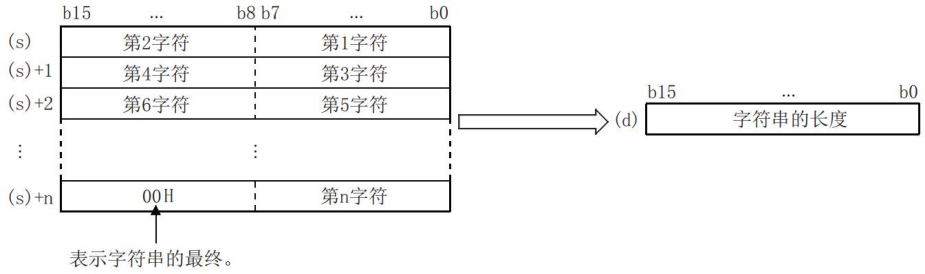
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LEN	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

检测(s)中指定的字符串的长度后，存储到(d)中指定的软元件编号以后。

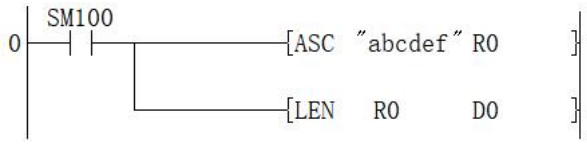
将(s)中指定的软元件编号开始，至存储的 00H 的软元件编号为止的数据作为字符串处理。



错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
408AH	(s)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束。
4086H	(d)在使用偏移的情况下偏移后的地址超出软元件范围。

示例



例如以上梯形图

使用 asc 指令给 R0 起始的地址写入字符串 abcdef。

然后使用 LEN 指令判断长度。此时 D0 将会显示 6。

LEFT/从字符串的左侧开始提取

LEFT(P)

对于(s)中指定的软元件编号以后中存储的字符串数据，将字符串的左侧(字符串的起始)开始的(n)字符的数据，存储到(d)中指定的软元件编号以后。

-[LEFT (s) (d) (n)]

内容、范围、数据类型

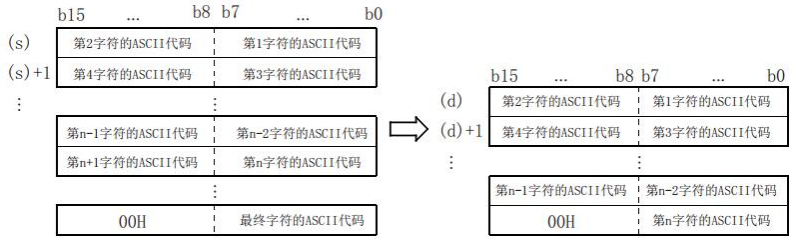
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	字符串或存储了字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	存储从(s)的左侧开始的(n)字符的字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(n)	提取的字符数	[1,400]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LEFT	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●			●	●		●	●

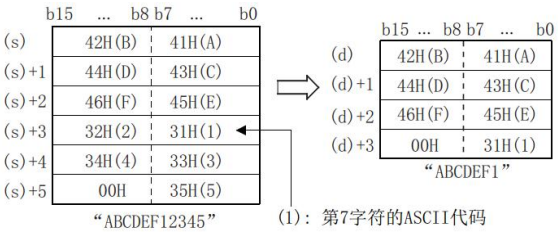
功能

对于(s)中指定的软元件编号以后中存储的字符串数据，将字符串的左侧(字符串的起始)开始的(n)字符的数据，存储到(d)中指定的软元件编号以后。



(s)中指定的字符串是从指定的软元件开始，以字节单位至首个检测到“00H”的位置为止的数据。

（n）=7 的情况下：



表示字符串的最终的 NULL 代码(00H)将被自动附加到字符串数据的最后。

提取的字符数为奇数的情况下，存储最终字符的软元件的高位字节中将存储“00H”。提取的字符数为偶数的情况下，存储最终字符的软元件之后的软元件中将存储“0000H”。

(n)中指定的字符数为 0 的情况下，(d)中将存储 NULL 代码(00H)。

注意事项

处理 ASCII 码以外的字符代码时，应注意以下几点。

字符数以字节单位(8 位)处理。因此，像移位 JIS 代码那样用 2 字节表示 1 字符的字符代码，1 个字符的字符数为“2”。

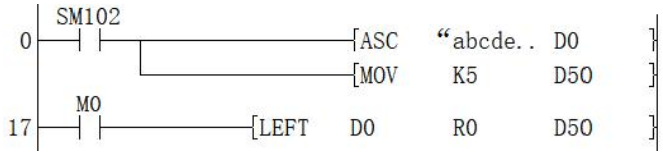
从包含像移位 JIS 代码那样用 2 字节表示 1 字符的字符代码的字符串中提取字符串时，应以 1 字符的字符代码为单

位考虑提取的字符数。如果只提取了 2 字节的字符代码中的 1 字节，将不会是所期望的字符代码，应加以注意。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
408AH	(s)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束。
4084H	(n)<1 或(n)>字符串长度。
4086H	(d)写入地址超出软元件范围 。

示例



从 D0 起始的“abcdef”中从左开始取出 5 个字符到 R0 中，R0 的字符串为“abcde”。

R0	1	0	0	0	0	1	1	0	0	1	0	0	0	1	1	0	ab
R1	1	1	0	0	0	1	1	0	0	0	1	0	0	1	1	0	cd
R2	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	e.
R3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

RIGHT/从字符串的右侧开始提取

RIGHT(P)

对于(s)中指定的软元件编号以后中存储的字符串数据，将字符串的右侧(字符串的最终)开始的(n)字符的数据，存储到(d)中指定的软元件编号以后。

-[RIGHT (s) (d) (n)]

内容、范围、数据类型

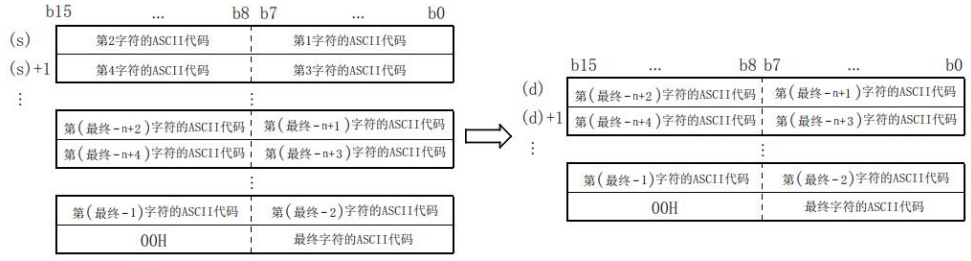
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	字符串或存储了字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	存储从(s)的右侧开始的(n)字符的字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(n)	提取的字符数	[1,400]	有符号BIN16位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RIGHT	参数1											●	●	●	●	●	●	●	●	●						●	●
	参数2												●	●	●	●	●	●	●	●						●	●
	参数3												●	●	●	●	●	●	●	●			●	●		●	●

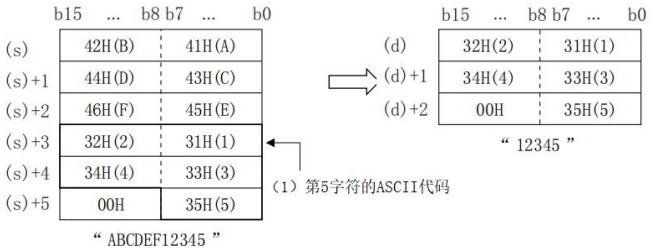
功能

对于(s)中指定的软元件编号以后中存储的字符串数据，将字符串的右侧(字符串的最终)开始的(n)字符的数据，存储到(d)中指定的软元件编号以后。



(s)中指定的字符串是从指定的软元件开始，以字节单位至首个检测到“00H”的位置为止的数据。

（n）=5 的情况下：



表示字符串的最终的 NULL 代码(00H)将被自动附加到字符串数据的最后。

提取的字符数为奇数的情况下，存储最终字符的软元件的高位字节中将存储“00H”。提取的字符数为偶数的情况下，存储最终字符的软元件之后的软元件中将存储“0000H”。

(n)中指定的字符数为 0 的情况下，(d)中将存储 NULL 代码(00H)。

注意事项

处理 ASCII 码以外的字符代码时，应注意以下几点。

字符数以字节单位(8 位)处理。因此，像移位 JIS 代码那样用 2 字节表示 1 字符的字符代码，1 个字符的字符数为“2”。

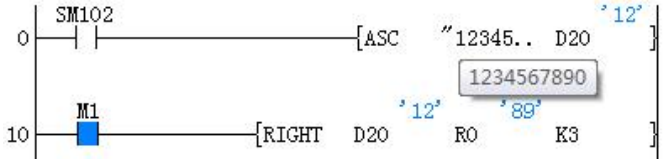
福州富昌维控电子科技有限公司

从包含像移位 JIS 代码那样用 2 字节表示 1 字符的字符代码的字符串中提取字符串时，应以 1 字符的字符代码为单位考虑提取的字符数。如果只提取了 2 字节的字符代码中的 1 字节，将不会是所期望的字符代码，应加以注意。

错误码

错误码	内容
4085H	(s)、(n)读取地址超出软元件范围 。
408AH	(s)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束。
4084H	(n)<1 或(n)>字符串长度。
4086H	(d)写入地址超出软元件范围 。

示例



字符串“1234567890”中从右开始获取 3 个字符“890”存入 R0 中。

R0	0	0	0	1	1	1	0	0	1	0	0	1	1	1	0	0	89
R1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0.

MIDR/字符串中的任意提取

MIDR(P)

将字符串数据中的任意位置的数据存储到(d)中指定的软元件编号以后。

-[MIDR (s1) (d) (s2)]

内容、范围、数据类型

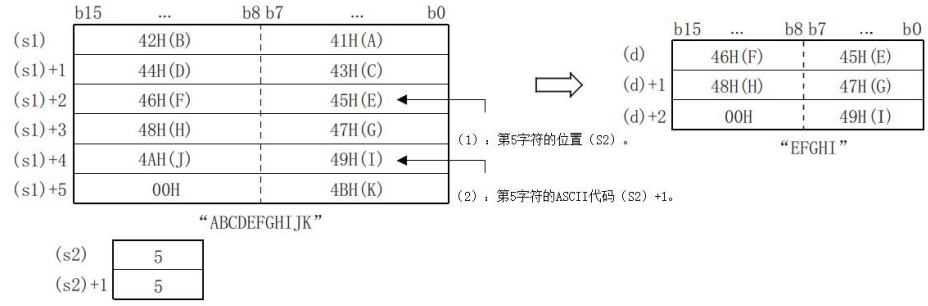
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	字符串或存储了字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	存储运算结果字符串数据的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(s2)	存储起始字符位置及字符数的软元件起始编号(s2)+0：起始字符的位置，(s2)+1:字符数有符号	-	有符号 BIN16 位	ANY16_ARRAY	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MIDR	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●						●	●

功能

对于(s1)中指定的软元件编号以后中存储的字符串数据，将(s2)中指定位置开始的(s2)+1 中指定字符的数据，存储到(d)中指定的软元件编号以后。



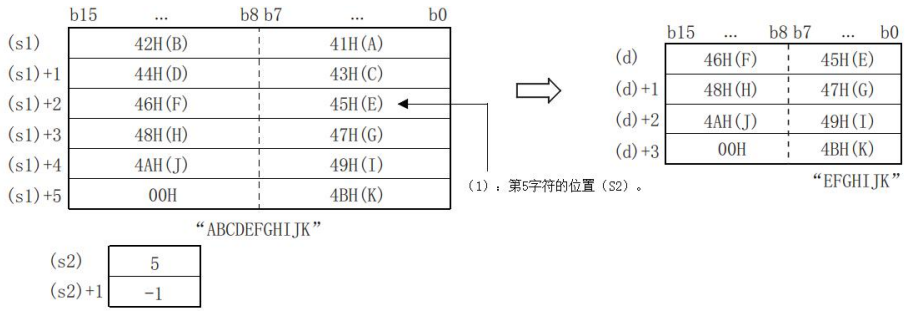
(s1)中指定的字符串是从指定的软元件开始，以字节单位至首个检测到“00H”的位置为止的数据。

表示字符串的最终的 NULL 代码(00H)将被自动附加到字符串数据的最后。

提取的字符数“(s2)+1”为奇数的情况下，存储最终字符的软元件的高位字节中将存储“00H”。提取的字符数“(s2)+1”为偶数的情况下，存储最终字符的软元件之后的软元件中将存储“0000H”。

(s2)+1 中指定的字符数为 0 的情况下不进行处理。

(s2)+1 中指定的字符数为-1 的情况下，将(s1)中指定的最终字符数据为止的数据存储到(d)中指定的软元件以后。



注意事项

处理 ASCII 码以外的字符代码时，应注意以下几点：

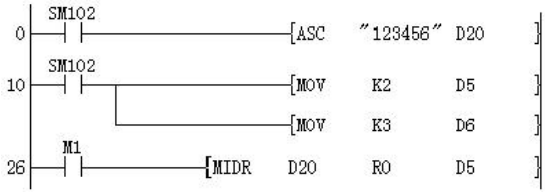
① 字符数以字节单位(8 位)处理。因此，像移位 JIS 代码那样用 2 字节表示 1 字符的字符代码，1 个字符的字符数为“2”；

② 从包含像移位 JIS 代码那样用 2 字节表示 1 字符的字符代码的字符串中提取字符串时，应以 1 字符的字符代码为单位考虑提取的字符数。如果只提取了 2 字节的字符代码中的 1 字节，将不会是所期望的字符代码，应加以注意。

错误码

错误码	内容
4085H	(s1)、(s2)读取地址超出软元件范围。
408AH	(s1)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s1)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束。
4084H	(s2)+1 的值为-2（包括-2）以下时； (s2)的值超过了(s1)的字符数时； (s2)的值为负时； (s2)+1 的值超过了(s1)的字符数时； (s2)与(s2)+1 的加法运算后的值超过了(s1)的字符数时。
4086H	(d)写入地址超出软元件范围。

示例



从字符串“123456”第二个字符起始获取三个字符“234”到 R0 中。

R0	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	23
R1	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	4.

$MOV/字符串传送

$MOV(P)

将(s)中指定的字符串数据传送到(d)中指定的软元件编号以后。

-[$MOV (s) (d)]

内容、范围、数据类型

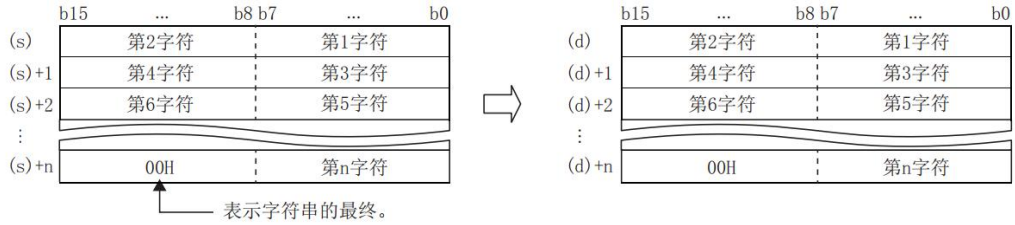
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	传送字符串(最大255字符)或存储了字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	存储传送字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--

使用的软元件

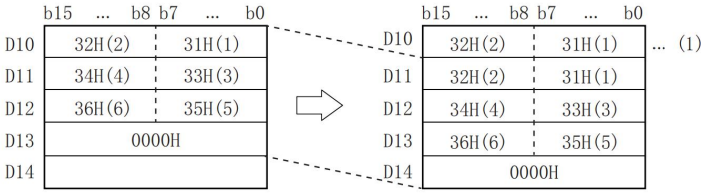
指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
$MOV	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●

功能

将(s)中指定的字符串数据传送到(d)中指定的软元件编号以后。在字符串的传送中，对(s)中指定的“ ”(双引号)围住的字符串，或从软元件编号开始至存储了00H的软元件编号为止的字符串进行一次传送。

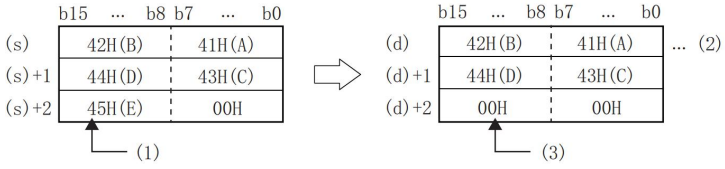


即使存储了传送的字符串数据的软元件范围(s)～(s)+n与存储传送后的字符串数据的软元件范围(d)～(d)+n重复的情况下，也将正常进行处理。例如，将D10~D13中存储的字符串传送到D11~D14中时，其情况如下所示。



(1)：直接变为传送前的字符串。

(s)+n的低位字节中存储了00H的情况下，(d)+n的高位字节、低位字节均将存储00H。



(1)：不能传送高位字节。

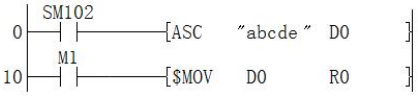
(2)：直接变为传送前的字符串。

(3)：高位字节自动存储00H。

错误码

错误码	内容
4085H	(s)读取地址超出软元件范围。
408AH	(s)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束。
4086H	(d)写入地址超出软元件范围。

示例



将 D0 中的字符串“abcde”复制到 R0 中。

R0	1	0	0	0	0	1	1	0	0	1	0	0	0	1	1	0	ab
R1	1	1	0	0	0	1	1	0	0	0	1	0	0	1	1	0	cd
R2	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	e.

MIDW/字符串中的任意替换

MIDW(P)

对于(s1)中指定的软元件编号以后中存储的字符串数据，将(s2)+1 中指定的字符的数据，存储到(d)中指定的软元件编号以后中存储的字符串数据的(s2)中指定的位置以后。

-[MIDW (s1) (d) (s2)]

内容、范围、数据类型

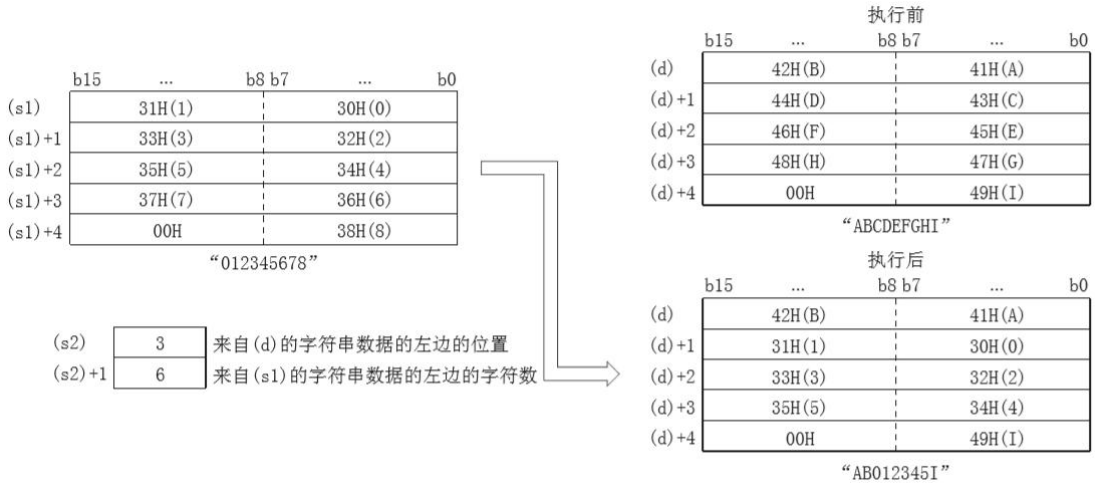
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	字符串或存储了字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	存储运算结果字符串数据的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(s2)	存储起始字符位置及字符数的软元件起始编号 (s2)+0：起始字符的位置，(s2)+1：字符数有符号	-	有符号 BIN16 位	ANY16_ARRAY	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MIDW	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2												●	●	●	●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●						●	●

功能

对于(s1)中指定的软元件编号以后中存储的字符串数据，将(s2)+1 中指定的字符的数据，存储到(d)中指定的软元件编号以后中存储的字符串数据的(s2)中指定的位置以后。

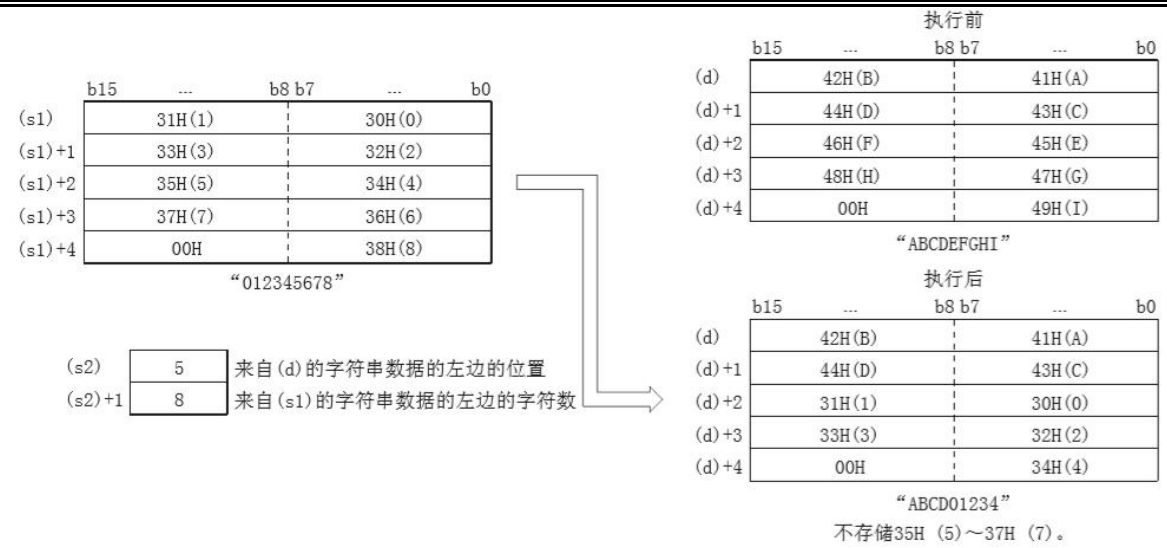


(s1)或(d)中指定的字符串是从指定的软元件开始，以字节单位至首个检测到“00H”的位置为止的数据。

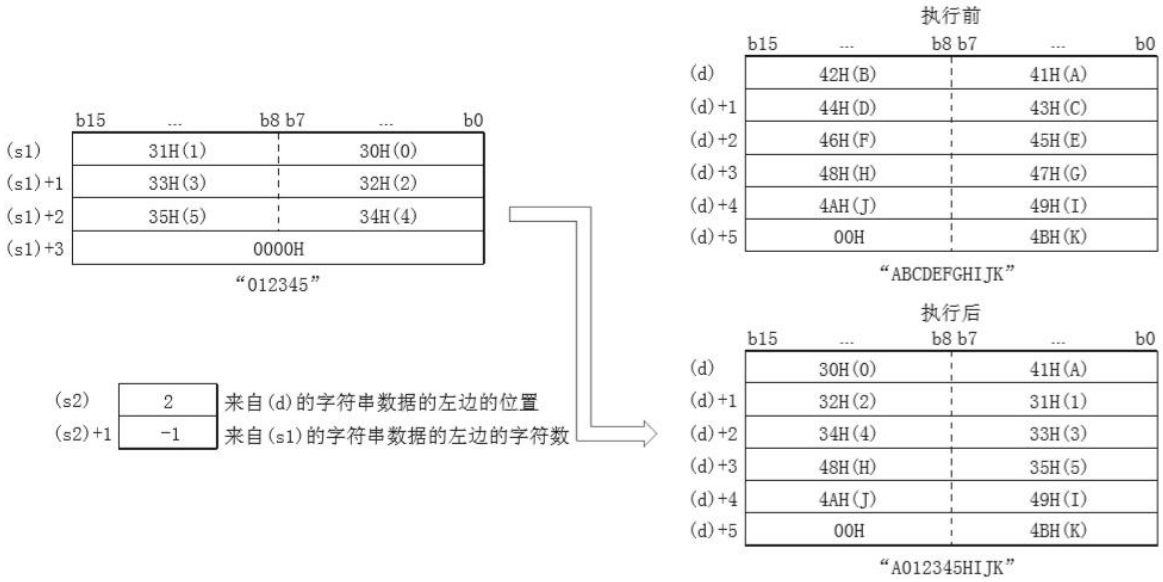
表示字符串的最终的 NULL 代码(00H)将被自动附加到字符串数据的最后。

(s2)+1 中指定的字符数为 0 的情况下不进行处理。

(s2)+1 中指定的字符数超过了(d)中指定的字符串数据的最终字符的情况下，存储(d)的最终字符为止的数据。



(s2)+1 中指定的字符数为-1 的情况下，将(s1)中指定的最终字符数据为止的数据存储到(d)中指定的软元件以后。



注意事项

处理 ASCII 码以外的字符代码时，应注意以下几点。

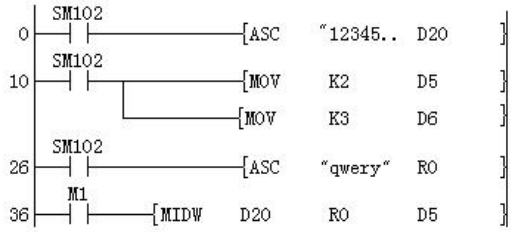
字符数以字节单位(8 位)处理。因此，像移位 JIS 代码那样用 2 字节表示 1 字符的字符代码，1 个字符的字符数为“2”。

从包含像移位 JIS 代码那样用 2 字节表示 1 字符的字符代码的字符串中提取字符串时，应以 1 字符的字符代码为单位考虑提取的字符数。如果只提取了 2 字节的字符代码中的 1 字节，将不会是所期望的字符代码，应加以注意。

错误码

错误码	内容
4085H	(s1)(s2) (d) 读取地址超出软元件范围。
408AH	(s1) (d)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s1) (d)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束。
4084H	(s2)+1 的值为-2（包括-2）以下时； (s2)的值超过了(d)的字符数时； (s2)的值为负时； (s2)+1 的值超过了(s1)的字符数时。
4086H	(d)写入地址超出软元件范围。

示例



将 R0 中存储的字符串“qwery”中的第二个字符起始的三个字符长度的字符替换成 D20 中的前三个字符。R0 的结果为“q123y”。

R0	1	0	0	0	1	1	1	0	1	0	0	0	1	1	0	0	q1
R1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	23
R2	1	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	y.

STR/BIN16 位数据→字符串转换

STR(P)

将(s2)中指定的 BIN16 位数据，在(s1)中指定的位置处附加小数点后转换为字符串，存储到(d)中指定的软元件编号以后。

-[STR　(s1)　(s2)　(d)]

内容、范围、数据类型

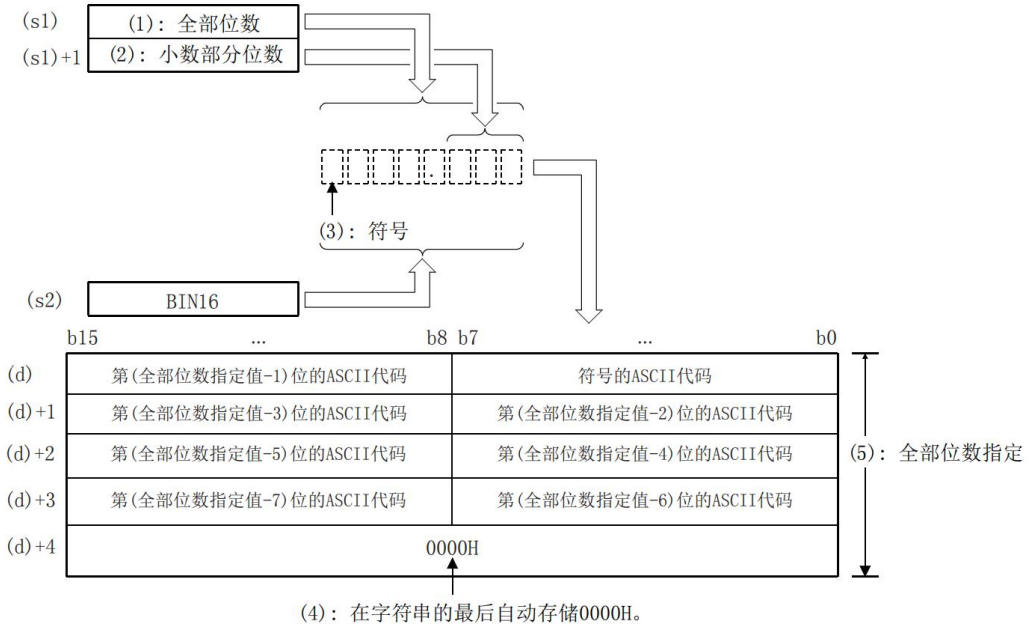
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储了转换数值的位数的软元件起始编号	-	有符号BIN16位	ANY16_S_ARRAY	--
(s2)	转换的 BIN 数据	[-32768,+32767]	有符号BIN16位	ANY16_S	INT
(d)	存储转换后的字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
STR	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●			●	●		●	●
	参数 3															●	●	●	●	●						●	●

功能

将(s2)中指定的 BIN16 位数据，在(s1)中指定的位置处附加小数点后转换为字符串，存储到(d)中指定的软元件编号以后。



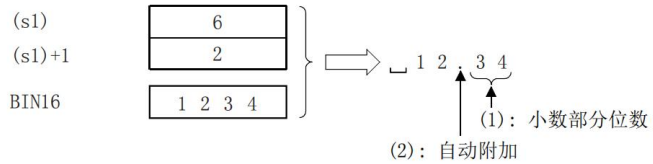
(s1)中可指定的全部位数为 2~8 位数。

(s1)+1 中可指定的小数部位数为 0~5 位数。但是，设置时应满足小数部位数≤(全部位数-3)的条件。

转换后的字符串数据将按下述方式被存储到(d)以后的软元件编号中。

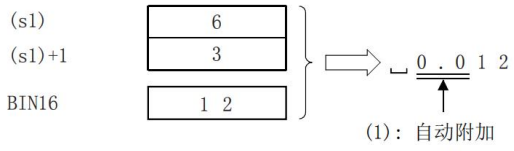
•在符号中，BIN16 位数据为正时将存储 20H(空白)，为负时将存储 2DH(-)。

•将小数部位数设置为 0 以外的情况下，第指定位数+1 位数中将自动存储 2EH(.)。小数部位数为 0 时，不存储 2EH(.)。



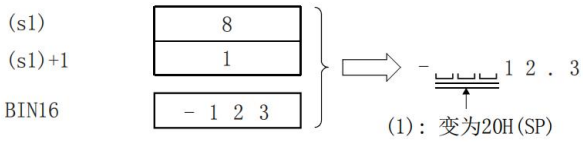
31H(1)	20H(空格)
2EH(.)	32H(2)
34H(4)	33H(3)
00H	00H

小数部位数的值大于 BIN16 位数据的位数的情况下，将自动附加 0 后，向右对齐转换为“0.***”。



30H(0)	20H(空格)
30H(0)	2EH(.)
32H(2)	31H(1)
00H	00H

全部位数的值中除去符号、小数点后的位数大于 BIN16 位数据的位数的情况下，在符号与数值之间将存储 20H(空白)。BIN16 位数据的位数一方大的情况下，将变为出错状态。



20H(空格)	2DH(-)
20H(空格)	20H(空格)
32H(2)	31H(1)
33H(3)	2EH(.)
00H	00H

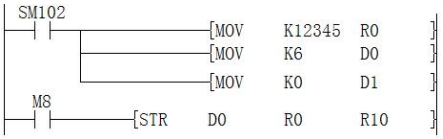
转换后的字符串的最后将自动存储 00H。

•总位数为偶数位数的情况下，存储最终字符的软元件之后的软元件中将存储“0000H”。而奇数位数的情况下，存储最终字符的软元件的高位字节(8 位)中将存储“00H”。

错误码

错误码	内容
4085H	(s1) 、 (s2) 读取地址超出软元件范围。
4084H	(s1)或(s1+1)参数设置值超出范围。 例如： 1、(s1)的值不在 2-8 的范围内。 2、(s1+1)的值不在 0-5 的范围内。 3、(s1+1)的值大于(s1)的值减去 3。 4、(s1+1) 为 0 时(s1)中指定的位数小于(s2)中指定的 BIN16 位数据的位数+1 时。 (s1+1) 非 0 时(s1)中指定的位数小于(s2)中指定的 BIN16 位数据的位数+2 时。 ((s1)的位数 < (s2)的不包含符号的 BIN16 位数据的位数 + 符号(+或-)的位数 + 小数点(.)的位数)
4086H	(d)在使用偏移的情况下偏移后的地址超出软元件范围。

示例



M8 置 ON 后，根据设置的全部位数 6 位小数点 0 位的设置转换成字符串“ 12345”（1 前面有空格）。

R10	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	1
R11	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	23
R12	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	45

DSTR/BIN32 位数据→字符串转换

DSTR(P)

将(s2)中指定的 BIN32 位数据，在(s1)中指定的位置处附加小数点后转换为字符串，存储到(d)中指定的软元件编号以后。

-[DSTR　(s1)　(s2)　(d)]

内容、范围、数据类型

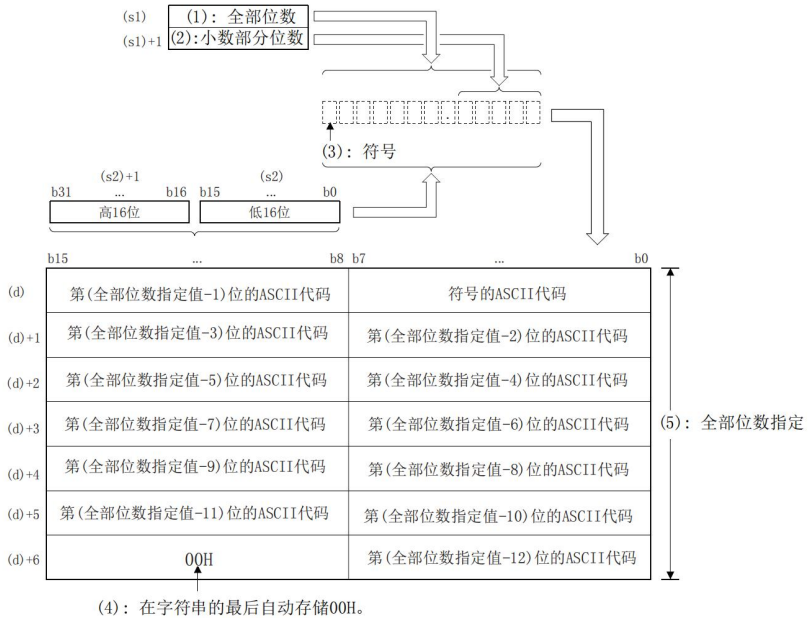
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	存储了转换数值的位数的软元件起始编号	-	有符号 BIN16 位	ANY16_S_ARRAY	--
(s2)	转换的 BIN 数据	[-2147483648,2147483647]	有符号 BIN32 位	ANY32_S	DINT
(d)	存储转换后的字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--

使用的软元件

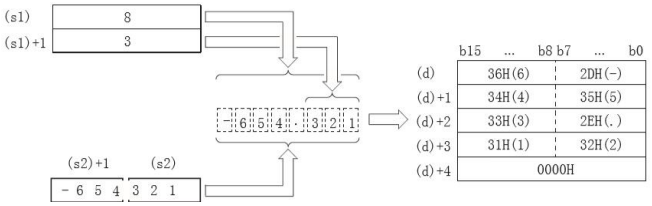
指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DSTR	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●	●	●	●	●		●	●
	参数 3															●	●	●	●	●						●	●

功能

将(s2)中指定的 BIN32 位数据，在(s1)中指定的位置处附加小数点后转换为字符串，存储到(d)中指定的软元件编号以后。



(S2) 中指定了-654.321 的情况下：



(s1)中可指定的全部位数为 2~13 位数。

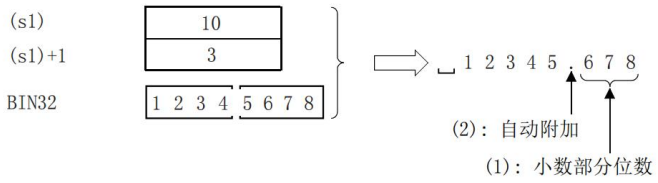
(s1)+1 中可指定的小数部位数为 0~10 位数。但是，设置时应满足小数部位数≤(全部位数-3)的条件。

福州富昌维控电子科技有限公司

转换后的字符串数据将按下述方式被存储到(d)以后的软元件编号中。

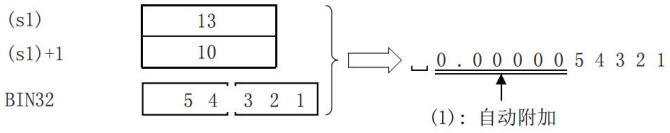
•在符号中，BIN32 位数据为正时将存储 20H(空白)，为负时将存储 2DH(-)。

•将小数部位数设置为0以外的情况下，第指定位数+1位数中将自动存储2EH(.)。小数部位数为0时，不存储2EH(.)。



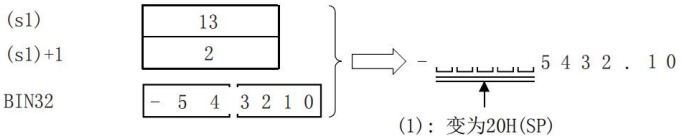
31H(1)	20H(空格)
33H(3)	32H(2)
35H(5)	34H(4)
36H(6)	2EH(.)
38H(8)	37H(7)
00H	00H

•小数部位数的值大于 BIN32 位数据的位数的情况下，将自动附加 0 后，向右对齐转换为“0.***”。



30H(0)	20H(空格)
30H(0)	2EH(.)
30H(0)	30H(0)
30H(0)	30H(0)
34H(4)	35H(5)
32H(2)	33H(3)
00H	31H(1)

•全部位数的值中除去符号、小数点后的位数大于 BIN32 位数据的位数的情况下，在符号与数值之间将存储 20H(空白)。BIN16 位数据的位数一方大的情况下，将变为出错状态。



20H(空格)	2DH(-)
20H(空格)	20H(空格)
20H(空格)	20H(空格)
34H(4)	35H(5)
32H(2)	33H(3)
31H(1)	2EH(.)
00H	30H(0)

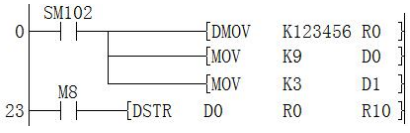
•转换后的字符串的最后将自动存储 00H。

•总位数为偶数位数的情况下，存储最终字符的软元件之后的软元件中将存储“0000H”。而奇数位数的情况下，存储最终字符的软元件的高位字节(8 位)中将存储“00H”。

错误码

错误码	内容
4085H	(s1)、 (s2) 读取地址超出软元件范围
4084H	(s1)或(s1+1)参数设置值超出范围。例如： 1、(s1)的值不在 2-13 的范围内。 2、(s1+1)的值不在 0-10 的范围内。 3、(s1+1)的值大于(S1)的值减去 3。 4、(s1+1) 为 0 时(s1)中指定的位数小于(s2)中指定的 BIN16 位数据的位数+1 时。 (s1+1) 非 0 时(s1)中指定的位数小于(s2)中指定的 BIN16 位数据的位数+2 时。 ((s1)的位数 < (s2)的不包含符号的 BIN16 位数据的位数 + 符号(+或-)的位数 + 小数点(.)的位数)
4086H	(d)在使用偏移的情况下偏移后的地址超出软元件范围 。

示例



如例子中所示，我们需要将 123456 转换成有 9 个长度小数点后有 3 个长度的浮点数字符串，

转换得出的结果应该是：123.456 ，前面的数值将会有两个空格来补充不足的数字。

R10	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	
R11	1	0	0	0	1	1	0	0	0	1	0	0	1	1	0	0	12
R12	1	1	0	0	1	1	0	0	0	1	1	1	0	1	0	0	3.
R13	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	45
R14	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	6.

福州富昌维控电子科技有限公司

$+/字符串的合并

$+(P)

将(s2)中指定的软元件编号以后存储的字符串连接到(s1)中指定的软元件编号以后存储的字符串数据的后面，存储到(d)中指定的软元件编号以后。

-[$+ (s1) (s2) (d)]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	连接数据或存储了数据的软元件起始编号或直接指定的字符串	-	字符串	ANYSTRING_SINGLE	--
(s2)	连接的数据或存储了连接的数据的软元件起始编号或直接指定的字符串	-	字符串	ANYSTRING_SINGLE	--
(d)	存储连接结果的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--

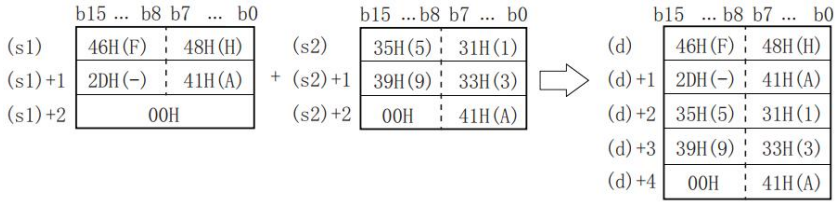
使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
$+	参数 1											●	●	●	●	●	●	●	●	●						●	●
	参数 2											●	●	●	●	●	●	●	●	●						●	●
	参数 3												●	●	●	●	●	●	●	●						●	●

功能

将(s2)中指定的软元件编号以后存储的字符串连接到(s1)中指定的软元件编号以后存储的字符串数据的后面，存储到(d)中指定的软元件编号以后。

(s1)、(s2)的字符串是从指定的软元件编号开始至存储了 00H 的软元件编号为止。



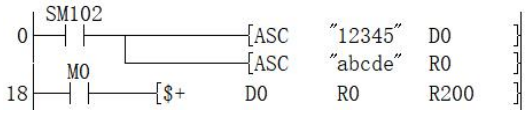
字符串合并时，将忽略表示(s1)中指定的字符串的结束的 00H，在(s1)的最终字符处连接(s2)中指定的字符串。

如果进行字符串的合并，将在最后自动附加 00H。 连接后的字符数为奇数的情况下，存储了最终字符的软元件的高位字节中将存储 00H，连接后的字符数为偶数的情况下，存储了最终字符的软元件之后的软元件中将存储 0000H。

错误码

错误码	内容
4085H	(s1)或(s2)读取地址超出软元件范围。
408AH	(s1)或(s2)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s1)或(s2)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束。
4086H	(d)写入地址超出软元件范围。

示例



将字符串“12345”和字符串“abcde”进行拼合后的结果是“12345abcde”

R200	1	0	0	0	1	1	0	0	0	1	0	0	1	1	0	0	12
R201	1	1	0	0	1	1	0	0	0	0	1	0	1	1	0	0	34
R202	1	0	1	0	1	1	0	0	1	0	0	0	0	1	1	0	5a
R203	0	1	0	0	0	1	1	0	1	1	0	0	0	1	1	0	bc
R204	0	0	1	0	0	1	1	0	1	0	1	0	0	1	1	0	de

福州富昌维控电子科技有限公司

INSTR/字符串查找

INSTR(P)

从(s2)中指定的软元件编号以后中存储的字符串数据的左侧第(s3)字符开始，搜索(s1)中指定的软元件编号以后中存储的字符串数据，将搜索结果存储到(d)中指定的软元件中。

-[INSTR (s1) (s2) (d) (s3)]

内容、范围、数据类型

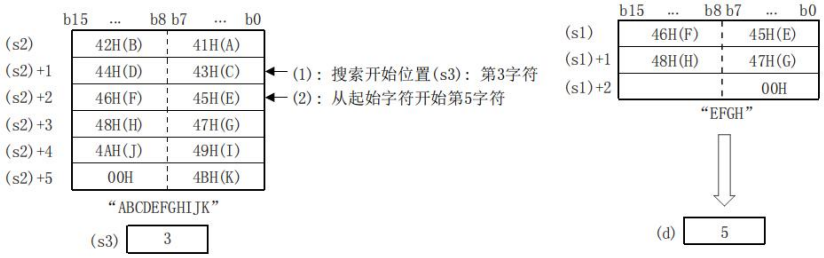
参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s1)	搜索字符串或存储了搜索字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(s2)	搜索的字符串或存储了搜索的字符串的软元件起始编号	-	字符串	ANYSTRING_SINGLE	--
(d)	存储搜索结果的软元件起始编号	-	有符号 BIN16 位	ANY16	--
(s3)	搜索开始位置	[1,400]	有符号 BIN16 位	ANY16	INT

使用的软元件

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
INSTR	参数 1															●	●	●	●	●						●	●
	参数 2															●	●	●	●	●						●	●
	参数 3											●	●	●	●	●	●	●	●	●						●	●
	参数 4											●	●	●	●	●	●	●	●	●			●	●		●	●

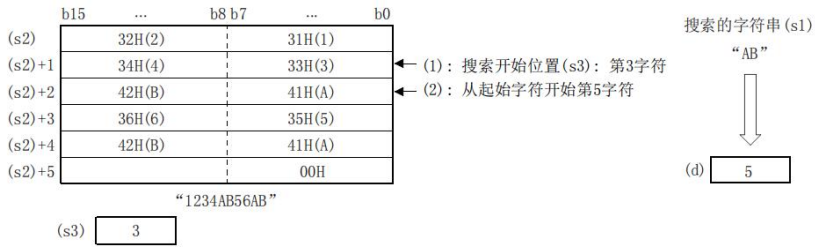
功能

从(s2)中指定的软元件编号以后中存储的字符串数据的左侧第(s3)字符开始，搜索(s1)中指定的软元件编号以后中存储的字符串数据，将搜索结果存储到(d)中指定的软元件中。搜索结果将存储(s2)中指定的字符串数据的起始字符开始的第几字符。



没有一致的字符串数据的情况下，(d)中将存储 0。

搜索开始位置(s3) “0”的情况下，不进行处理，搜索的字符串(s1)可直接指定字符串。

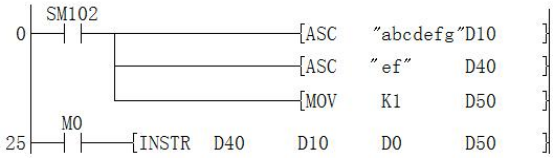


福州富昌维控电子科技有限公司

错误码

错误码	内容
4085H	(s1)、(s2)、(s3) 读取地址超出软元件范围。
408AH	(s1)、(s2)读取字符串长度超出，字符串连续长度超过 400 个字符。
408BH	(s1)、(s2)读取字符串时读取到软元件最大范围但是没有找到 00H 最为结束。
4086H	(d)写入地址超出软元件范围。
4084H	(s3) <0 或(s3)>=字符串长度。

示例



连续字符串“abcdefg”中查找“ef”字符串从第一个开始查找位置在第 5 位置上。

ASC/ASCII 数据输入

ASC

将半角/英文数字字符串转换成 ASCII 码的指令。 用于在外部显示器中选择显示多个消息。

-[ASC（s） （d）]

内容、范围、数据类型

参数	内容	范围	数据类型	数据类型(标签)	自定义变量类型
(s)	从计算机输入的 32 个字符的半角英文数字	-	字符串(仅 ASCI 码)	ANY_ASC	--
(d)	保存 ASCII 数据的起始字软元件编号	-	BIN16 位	ANY16_S	--

使用的软元件

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ASC	参数 1																										
	参数 2															●	●	●	●	●						●	

功能

1. 16 位运算(ASC)

•将(S)中指定的半角、英文、数字字符串转换成 ASCII 码后，依次传送到(D)中。

（1）在(S)中处理 A～Z、 0～9、符号的半角字符。 (不处理全角字符串。)在用编程工具编程时，输入字符串。

（2）转换后的 ASCII 码按照低 8 位、高 8 位的顺序，每 2 个字符/1 字节地保存在(D)中。

2. 扩展功能

•SM161 置 ON 后，扩展功能变为有效，此时将 S 中指定的半角/英文数字字符串转换成 ASCII 码，然后将其依次传送到 D 的低 8 位(1 个字节)中。

注意事项

1. 软元件的占用点数

（1） 扩展功能 OFF 时 ，D 占用字符数÷2 点(不能整除时，则将小数点进位。)

（2） 扩展功能 ON 时； D 占用的点数和字符数占用的数目相同。

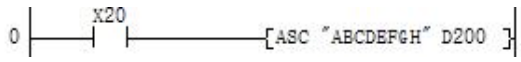
2. 使用 ASC、ASCI、BCC、CCD、CRC 等指令时，扩展功能标志位 SM161 是与其他指令通用的标志位。

使用上述指令和 ASC 指令时，请注意，在 ASC 指令的前面编写 SM161 ON 或 OFF 的程序，以不造成影响。

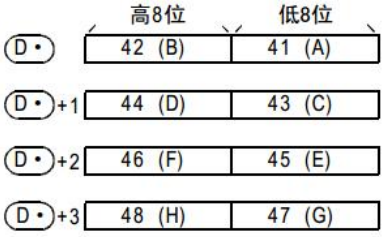
错误码

错误码	内容
4085H	读应用指令中(s)输出结果超出软元件范围。
4086H	写应用指令中(d)输出结果超出软元件范围。

示例



当 X20 = ON 时，D200~D203 的赋值：



若将特殊寄存器 SM161 置为 ON，则每个 ASCII 字符在转换后占用 1 个 16bit 变量

如下图，每个变量的高字节填 0 处理(16 进制)：

	D•		S
	高 8 位	低 8 位	字符串
D•	00	41	A
D•+1	00	42	B
D•+2	00	43	C
D•+3	00	44	D
D•+4	00	45	E
D•+5	00	46	F
D•+6	00	47	G
D•+7	00	48	H

13 以太网通讯

13.1 以太网概述

13.1.1 IP 地址概念

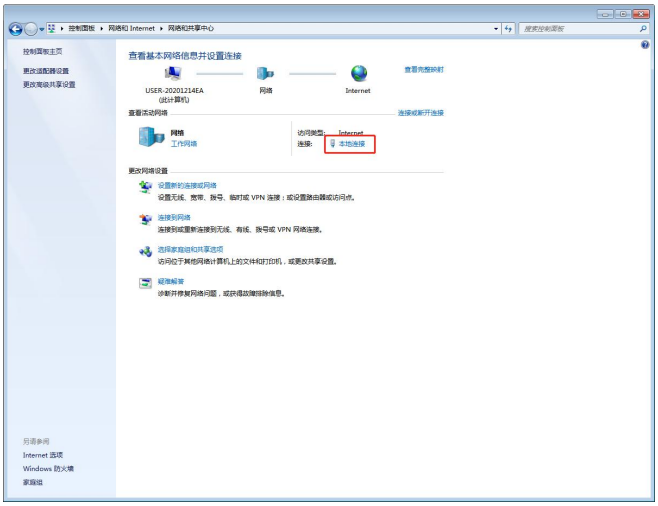
IP 地址由网络地址和主机地址组成，由子网掩码进行区分。若编程设备（如 PC）使用网卡需要连接到工厂局域网，则编程设备和 PLC 必须处于同一子网中。IP 地址与子网掩码相结合即可指定设备的子网。

将 IP 地址与子网掩码进行逻辑与运算，可以计算出网络地址，若地址处于同一网络，则表示可以通讯。如：

编号	网络设备 1			网络设备 2			网络连通
	IP	子网掩码	网络地址	IP	子网掩码	网络地址	
1	192.168.0.1	255.255.255.0	192.168.0.0	192.168.0.10	255.255.255.0	192.168.0.0	是
2	192.168.0.1	255.255.255.0	192.168.0.0	192.168.1.10	255.255.255.0	192.168.1.0	否
3	192.168.0.1	255.255.255.1	192.168.0.1	192.168.0.10	255.255.255.1	192.168.0.0	否

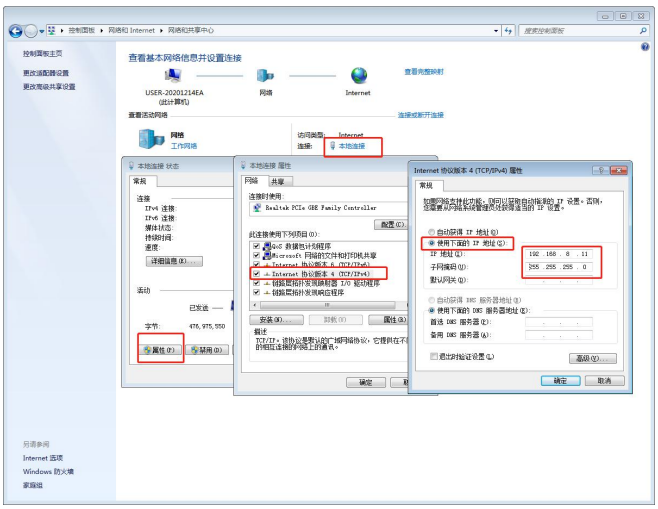
13.1.2 设置 PC 网络地址

1. 打开控制面板->网络和 Internet->网络和共享中心；



2. 点击本地链接->属性->Internet 协议版本 4；

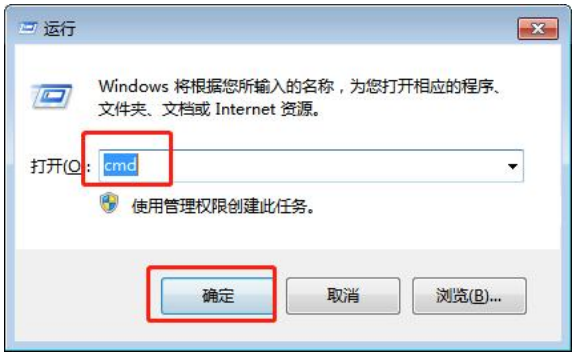
3. 设置与 PLC 在同一网络地址的 IP 地址和子网掩码，注意不能设置局域网内已经被使用的 IP 地址。如 PLC 的 IP 为 192.168.8.8，子网掩码为 255.255.255.0，可以设置如下图 IP 地址，可以使 PC 连接上 PLC。



13.1.3 测试网络连接状态

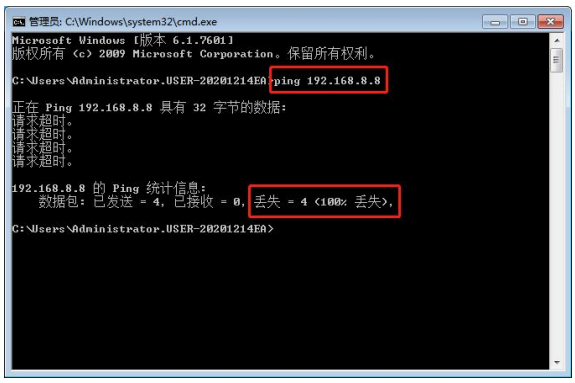
通过 ping 命令可以测试 PC 与 PLC 的连接状态。

1. win+R 键，输入 cmd；

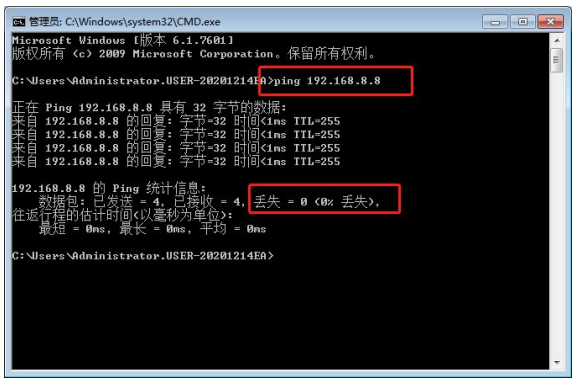


2. 若 PLC 的 IP 地址为 192.168.8.8；

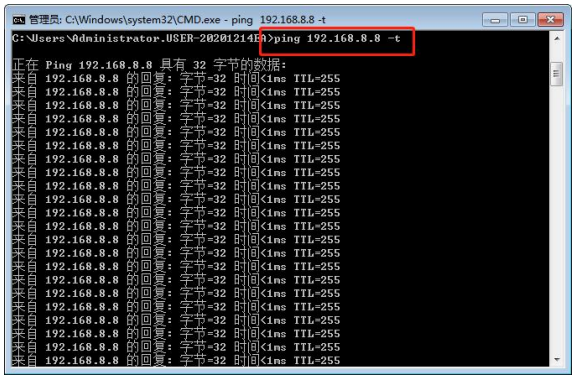
(1) 输入 ping 192.168.8.8 命令，按回车。若显示 100%丢失，则说明无法与 PLC 连接。



(2) 输入 ping 192.168.8.8 命令，按回车。若显示 0%丢失，则说明可以与 PLC 连接。



(3) “ping 网络设备 ip”命令仅可以 ping 四次，若想一直 ping，可以使用“ping 网络设备 ip -t”命令，如下图所示：



13.1.4 PLC Editor2 与 PLC 进行以太网连接

1. 传输设置->以太网配置->输入 PLC 的 IP 地址；（注意：网卡的地址要与 PLC 在同一网段）

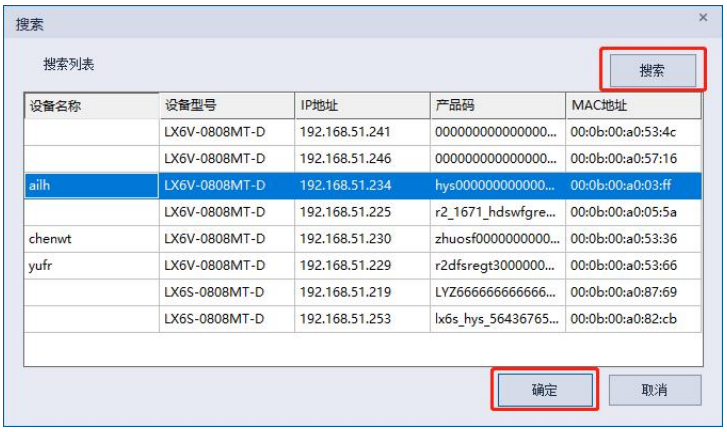
2. 点击通讯测试进行通讯确认；

3. 连接成功后点击选择，就可以操作 PLC 了。

13.1.5 PLC Editor2 的以太网搜索功能

1．传输设置->网卡选择->设备搜索（注意：网卡的地址要与 PLC 在同一网段），在 1 处选择与 PLC 相连，或与 PLC 在同一网段的网卡，点击 2 处进行局域网内的 PLC 设备搜索。

2．搜索界面如下所示，点击搜索后即可显示局域网内的 PLC 设备，选中相应设备后点击确定，即可进行通讯。



3．1 处 IP 地址自动进行填充，点击通讯测试测试通讯是否正常，正常后点选择按钮完成连接配置。

13.2 以太网配置

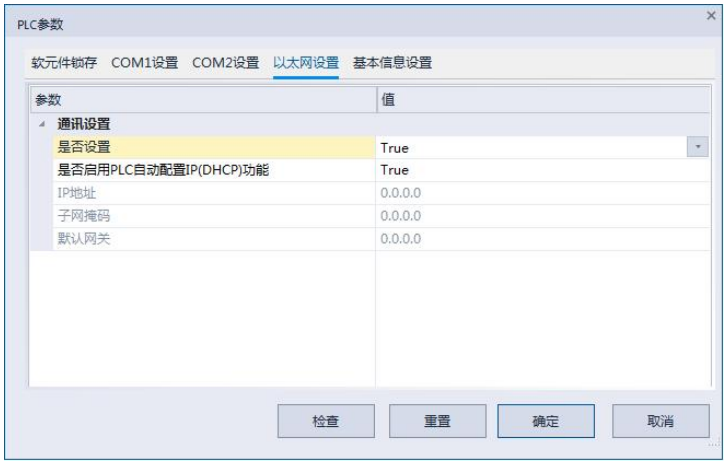
13.2.1 IP 地址设置

（1）通过编程软件设置

工程管理->参数->PLC 参数->以太网配置。

修改后通过 PLC 下载勾选参数项，下载完成 STOP->RUN 后生效。

注:Modbus TCP 服务器支持最大连接数用于设置可支持外部 Modbus TCP 客户端来同时连接 PLC 的最大个数,范围 2~8。



（2）通过特殊软元件设置

在 SD2680~SD2691 中写入 IP 地址、子网掩码、默认网关；

SM2680 设置为 ON 开启静态 IP 功能；

SM2683 设置为 ON 标识 IP 可修改；

STOP->RUN 或者重新上电后新 IP 地址生效。

SM 标号	名称	内容	R/W	SD 标号	名称
SM2680	静态设置 IP 开关	ON：静态设置 OFF：通过路由器 DHCP 自动配置 IP 地址，且无法修改 IP STOP->RUN 生效	R/W	SD2680	ip 地址第一字节
SM2681	显示当前网络信息	ON 后刷新当前 IP 网关子网掩码。刷新完成后 OFF	R/W	SD2681	ip 地址第二字节
SM2682	显示当前 MAC 信息（预留）	ON 后刷新当前 MAC。刷新完成后 OFF	R/W	SD2682	ip 地址第三字节
SM2683	IP、子网掩码、网关修改标志	ON：可修改 OFF：不可修改 （设置为 ON 后，STOP->RUN 时进行修改，修改完成后自动变为 OFF）	R/W	SD2683	ip 地址第四字节
SM2684				SD2684	子网掩码第一字节
SM2685				SD2685	子网掩码第二字节
SM2686				SD2686	子网掩码第三字节
SM2687				SD2687	子网掩码第四字节
SM2688				SD2688	默认网关第一字节
SM2689				SD2689	默认网关第二字节
SM2690				SD2690	默认网关第三字节
SM2691				SD2691	默认网关第四字节

13.2.2 TCP 协议概念

TCP 协议为传输控制协议（Transport Control Protocol），是一种面向连接的、可靠的传输层协议。面向连接是指一次正常的 TCP 传输需要通过在 TCP 客户端和 TCP 服务端建立特定的虚电路连接来完成。要通过 TCP 传输数据，必须在两端主机之间建立连接。

13.3 以太网相关特殊软元件一览表

SM 标号	名称	内容	R/W	掉电保存
SM2681	显示当前网络信息	ON 后刷新当前 IP、子网掩码、默认网关刷新完成后 OFF	R/W	×
SM2682	显示当前 MAC 信息	ON 后刷新当前 MAC。刷新完成后 OFF	R/W	×
SM2683	IP、子网掩码、网关修改标志	ON：可修改 OFF：不可修改 （设置为 ON 后，STOP->RUN 时进行修改，修改完成后自动变为 OFF）	R/W	√
SM2684	网卡连通状态	ON：网络连通 OFF：网络未连通，请检查网线是否链接	R	√
SM2692	MAC 地址修改标志	ON：可修改 OFF：不可修改 （设置为 ON 后，STOP->RUN 时进行修改，修改完成后自动变为 OFF）	R/W	√
SM2700	Modbus TCP 保活机制	ON:开启 OFF：关闭（默认）	R/W	√
SM2701	Modbus TCP 服务器强制关闭	ON:开启 OFF：关闭（默认）（启用后关闭成功后自动变为 OFF）	R/W	×
SM2710	以太网出错标志	ON：以太网出现出错，请查看 SD2710 和 SD2711 OFF：以太网无错误	R	×
SM2740	Modbus TCP 服务器链接状态 1	ON：客户端已链接 OFF：客户端未链接	R	×
SM2760	Modbus TCP 服务器链接状态 2	ON：客户端已链接 OFF：客户端未链接	R	×
SM2780	Modbus TCP 服务器链接状态 3	ON：客户端已链接 OFF：客户端未链接	R	×
SM2800	Modbus TCP 服务器链接状态 4	ON：客户端已链接 OFF：客户端未链接	R	×
SM2820	Modbus TCP 服务器链接状态 5	ON：客户端已链接 OFF：客户端未链接	R	×
SM2840	Modbus TCP 服务器链接状态 6	ON：客户端已链接 OFF：客户端未链接	R	×
SM2860	Modbus TCP 服务器链接状态 7	ON：客户端已链接 OFF：客户端未链接	R	×
SM2880	Modbus TCP 服务器链接状态 8	ON：客户端已链接 OFF：客户端未链接	R	×

SD 标号	名称	内容	R/W	掉电保存
SD2680	ip 地址第一字节	本机 IP 地址	R/W	√
SD2681	ip 地址第二字节		R/W	√
SD2682	ip 地址第三字节		R/W	√
SD2683	ip 地址第四字节		R/W	√
SD2684	子网掩码第一字节	本机子网掩码	R/W	√

SD2685	子网掩码第二字节		R/W	√
SD2686	子网掩码第三字节		R/W	√
SD2687	子网掩码第四字节		R/W	√
SD2688	默认网关第一字节	本机默认网关	R/W	√
SD2689	默认网关第二字节		R/W	√
SD2690	默认网关第三字节		R/W	√
SD2691	默认网关第四字节		R/W	√
SD2692	MAC 地址第一字节	本机 MAC 地址	R/W	√
SD2693	MAC 地址第二字节		R/W	√
SD2694	MAC 地址第三字节		R/W	√
SD2695	MAC 地址第四字节		R/W	√
SD2696	MAC 地址第五字节		R/W	√
SD2697	MAC 地址第六字节		R/W	√
SD2700	通讯速度显示	0：100Mbps/半双工 1：100Mbps/全双工 2：10Mbps/半双工 3：10Mbps/全双工	R	×
SD2702	Modbus TCP 服务器支持最大链接数	本机 Modbus TCP 服务器最大支持多少个客户端同时链接	R/W	×
SD2703	Modbus TCP 已被链接数	本机 Modbus TCP 已被链接数量	R	×
SD2710	错误码	以太网出错错误码	R	×
SD2711	本次出错的套接字 ID	-1：系统默认 Modbus TCP 服务器 0~5：客户自定义套接字错误	R	×
SD2720	输入 ping 请求个数低位	外部输入 ping 命令计数	R	×
SD2721	输入 ping 请求个数高位		R	×
SD2722	输入 ping 应答个数低位	收到外部 ping 命令后回复个数	R	×
SD2723	输入 ping 应答个数高位		R	×
SD2724	输出 ping 请求个数低位	发送 ping 命令计数	R	×
SD2725	输出 ping 请求个数高位		R	×
SD2726	输出 ping 应答个数低位	收到发送的 ping 命令后回复计数	R	×
SD2727	输出 ping 应答个数高位		R	×
SD2728	arp 包发送个数	arp 包发送条数计数	R	×
SD2729	arp 包接收个数	arp 包接收条数计数	R	×
SD2730	ip 包发送个数	ip 包发送条数计数	R	×
SD2731	ip 包接收个数	ip 包接收条数计数	R	×
SD2732	tcp 包发送个数	tcp 包发送条数计数	R	×
SD2733	tcp 包接收个数	tcp 包接收条数计数	R	×
SD2734	udp 包发送个数	udp 包发送条数计数	R	×
SD2735	udp 包接收个数	udp 包接收条数计数	R	×
SD2740	链接 1 本地端口号	Modbus TCP 客户端第 1 个来链接本 PLC 的链接信息和错误信息。	R	×
SD2741	链接 1 IP 地址第一字节		R	×
SD2742	链接 1 IP 地址第二字节		R	×
SD2743	链接 1 IP 地址第三字节		R	×

SD2744	链接 1 IP 地址第四字节		R	×
SD2745	链接 1 对端端口号		R	×
SD2746	保留		R	×
SD2747	保留			
SD2748	链接 1 出错错误码		R	×
SD2749	链接 1 错误通讯次数低字		R	×
SD2750	链接 1 错误通讯次数高字		R	×
SD2760	链接 2 本地端口号	Modbus TCP 客户端第 2 个来链接本 PLC 的链接信息和错误信息。	R	×
SD2761	链接 2 IP 地址第一字节		R	×
SD2762	链接 2 IP 地址第二字节		R	×
SD2763	链接 2 IP 地址第三字节		R	×
SD2764	链接 2 IP 地址第四字节		R	×
SD2765	链接 2 对端端口号		R	×
SD2766	保留		R	×
SD2767	保留			
SD2768	链接 2 出错错误码		R	×
SD2769	链接 2 错误通讯次数低字		R	×
SD2770	链接 2 错误通讯次数高字		R	×
SD2780	链接 3 本地端口号	Modbus TCP 客户端第 3 个来链接本 PLC 的链接信息和错误信息。	R	×
SD2781	链接 3 IP 地址第一字节		R	×
SD2782	链接 3 IP 地址第二字节		R	×
SD2783	链接 3 IP 地址第三字节		R	×
SD2784	链接 3 IP 地址第四字节		R	×
SD2785	链接 3 对端端口号		R	×
SD2786	保留		R	×
SD2787	保留			
SD2788	链接 3 出错错误码		R	×
SD2789	链接 3 错误通讯次数低字		R	×
SD2780	链接 3 错误通讯次数高字		R	×
SD2800	链接 4 本地端口号	Modbus TCP 客户端第 4 个来链接本 PLC 的链接信息。	R	×
SD2801	链接 4 IP 地址第一字节		R	×
SD2802	链接 4 IP 地址第二字节		R	×
SD2803	链接 4 IP 地址第三字节		R	×
SD2804	链接 4 IP 地址第四字节		R	×
SD2805	链接 4 对端端口号		R	×
SD2806	保留		R	×
SD2807	保留			
SD2808	链接 4 出错错误码	Modbus TCP 客户端第 4 个来链接本 PLC 的错误信息。	R	×
SD2809	链接 4 错误通讯次数低字		R	×
SD2810	链接 4 错误通讯次数高字		R	×
SD2820	链接 5 本地端口号	Modbus TCP 客户端第 5 个来链接本 PLC 的链接信息和错误信息。	R	×
SD2821	链接 5 IP 地址第一字节		R	×

SD2822	链接 5 IP 地址第二字节		R	×
SD2823	链接 5 IP 地址第三字节		R	×
SD2824	链接 5 IP 地址第四字节		R	×
SD2825	链接 5 对端端口号		R	×
SD2826	保留		R	×
SD2827	保留			
SD2828	链接 5 出错错误码		R	×
SD2829	链接 5 错误通讯次数低字		R	×
SD2830	链接 5 错误通讯次数高字		R	×
SD2840	链接 6 本地端口号	Modbus TCP 客户端第 6 个来链接本 PLC 的链接信息和错误信息。	R	×
SD2841	链接 6 IP 地址第一字节		R	×
SD2842	链接 6 IP 地址第二字节		R	×
SD2843	链接 6 IP 地址第三字节		R	×
SD2844	链接 6 IP 地址第四字节		R	×
SD2845	链接 6 对端端口号		R	×
SD2846	保留		R	×
SD2847	保留			
SD2848	链接 6 出错错误码		R	×
SD2849	链接 6 错误通讯次数低字		R	×
SD2850	链接 6 错误通讯次数高字		R	×
SD2860	链接 7 本地端口号	Modbus TCP 客户端第 7 个来链接本 PLC 的链接信息和错误信息。	R	×
SD2861	链接 7 IP 地址第一字节		R	×
SD2862	链接 7 IP 地址第二字节		R	×
SD2863	链接 7 IP 地址第三字节		R	×
SD2864	链接 7 IP 地址第四字节		R	×
SD2865	链接 7 对端端口号		R	×
SD2866	保留		R	×
SD2867	保留			
SD2868	链接 7 出错错误码		R	×
SD2869	链接 7 错误通讯次数低字		R	×
SD2870	链接 7 错误通讯次数高字		R	×
SD2880	链接 8 本地端口号	Modbus TCP 客户端第 8 个来链接本 PLC 的链接信息和错误信息。	R	×
SD2881	链接 8 IP 地址第一字节		R	×
SD2882	链接 8 IP 地址第二字节		R	×
SD2883	链接 8 IP 地址第三字节		R	×
SD2884	链接 8 IP 地址第四字节		R	×
SD2885	链接 8 对端端口号		R	×
SD2866	保留	Modbus TCP 客户端第 8 个来链接本 PLC 的链接信息和错误信息。		
SD2867	保留		R	×
SD2888	链接 8 出错错误码		R	×
SD2889	链接 8 错误通讯次数低字		R	×
SD2890	链接 8 错误通讯次数高字		R	×

13.4 以太网运算错误码一览表

错误码	说明	动作	处理方案	检测时机
3680	以太网数据接收错误	继续运行	检查环境是否存在干扰	指令执行时
3681	以太网数据接收超时	继续运行	检查网线是否松动 检查网络对端是否故障无法发出数据。 检查网络对端是否有响应不及时导致数据来不及，这个原因可以尝试加大套接字配置中的接收超时时间。	指令执行时
3684	Modbus TCP 站号配置错误	继续运行	检查从站站号设置。 以及检查从站接收发送机制是否有问题。	指令执行时
3685	Modbus TCP 发送缓冲区溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
3686	Modbus TCP 功能码错误	继续运行	检查设置的功能码是否是 PLC 支持的功能码	指令执行时
3687	Modbus TCP 地址错误	继续运行	检查从站是否有这个地址（请查阅 Modbus 异常 02）	指令执行时
3688	Modbus TCP 长度错误	继续运行	检查通讯长度是否超出 Modbus 范围	指令执行时
3689	Modbus TCP 数据错误	继续运行	检查指令的参数是否有错误。 检查从站是否支持这个值的设置。（请查阅 Modbus 异常 03）	指令执行时
368A	Modbus TCP 从站忙	继续运行	从站返回信息：从站忙（请查阅 Modbus 异常 06）	指令执行时
368B	Modbus TCP 从站不支持功能码	继续运行	检查从站是否支持这个功能码（请查阅 Modbus 异常 01）	指令执行时
368C	Modbus TCP 从站故障	继续运行	从站返回信息：从站故障，请检查从站是否故障（请查阅 Modbus 异常 04）	指令执行时
368D	Modbus TCP 从站确认	继续运行	从站返回信息：从站确认（请查阅 Modbus 异常 05）	指令执行时
368E	Modbus TCP 目前协议不支持这个指令	继续运行	当设置为从站协议时不能使用 RS 指令，请更改协议或者关闭 RS 指令前的触点	指令执行时
368F	网口发送超时	继续运行	出现该错误提示，请联系技术人员	指令执行时
3690	接收缓存溢出	继续运行	检查对方设备是否在一直发送数据	指令执行时
36A0	Modbus TCP 不可用网关	继续运行	从站返回信息：不可用网关（请查阅 Modbus 异常 0A）	指令执行时
36A1	Modbus TCP 指示没有从目标设备中获得响应。通常意味着设备未在网络中	继续运行	从站返回信息：设备未在网络中（请查阅 Modbus 异常 0B）	指令执行时
36C0	Modbus TCP 事务处理标识符错误	继续运行	检查是否存在网络拥堵，数据来不及接收。	指令执行时
36C1	Modbus TCP 服务器可用链接满	继续运行	查看 SD2702 和 SD2703 是否有过多的客户端来链接	指令执行时
36C8	以太网协议栈堆空间不足	继续运行	出现该错误提示，请联系技术人员	指令执行时
36C9	链接数超限制	继续运行	确认总链接数是否超限制	指令执行时
36CA	上次发送未完成	继续运行	使用发送完成标记，判断当前发送完成后，再发下一个	指令执行时

36CB	TCP 写入异常	继续运行	使用标记位软元件判断当前链接状态是否正常，不正常时，不进行数据发送，例如关闭中标记置位后，不进行数据发送动作。	指令执行时
36CC	TCP 输出异常	继续运行	出现该错误提示，请联系技术人员	指令执行时
36CD	IP 地址信息已使用	继续运行	请检查是否存在使用相同地址信息的链接存在	指令执行时
36CE	服务器接受链接错误	继续运行	出现该错误提示，请联系技术人员	指令执行时
36CF	TCP 接收缓存溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
36D0	TCP 链接失败	继续运行	可能在网线未链接的状态下开启 TCP 客户端导致	指令执行时
36D1	主动关闭链接时异常	继续运行	出现该错误提示，请联系技术人员	指令执行时
36D2	协议栈内部发生异常关闭	继续运行	可能为长时间得不到应答被关闭。检查对端是否在线，是否能 ping 通。	指令执行时
36D3	对端发起 RST 链接	继续运行	1、检查对端是否发起异常关闭。2、作为客户端，对端链接数已满，或对方端口未打开导致。	指令执行时
36D4	协议栈发生单端关闭	继续运行	出现该错误提示，请联系技术人员	指令执行时
36D5	存在 IP 地址冲突	继续运行	局域网内存在相同 IP 设备,请更改 IP 地址	指令执行时
36D6	存在 MAC 地址冲突	继续运行	局域网内存在相同 MAC 设备，请更改 MAC 地址	指令执行时
36D7	TCP 发送缓存溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
36D8	UDP 链接异常	继续运行	可能为 IP 地址，端口号已被使用导致	指令执行时
36D9	UDP 发送缓存溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
36DA	UDP 发送时内存空间不足	继续运行	出现该错误提示，请联系技术人员	指令执行时
36DB	UDP 发送失败	继续运行	出现该错误提示，请联系技术人员	指令执行时
36DC	UDP 内存释放失败	继续运行	出现该错误提示，请联系技术人员	指令执行时
36DD	UDP 接收缓存溢出	继续运行	UDP 接收的数据长度超过限制值 512 导致	指令执行时
4084	应用指令中输入了超出可指定范围的数据(如参数 1 规定为 0~1，设置 2)	继续运行	修改应用指令参数	应用指令执行时
4085	读应用指令中输出结果超出软元件范围（如 D 软元件最大 D7999，使用到 D8000）	继续运行	修改应用指令参数	应用指令执行时
4086	写应用指令中输出结果超出软元件范围（如 D 软元件最大 D7999，使用到 D8000）	继续运行	修改应用指令参数	应用指令执行时
5080	以太网套接字已经建立链接，无法重复打开	继续运行	检查 SOCOPEN 指令是否重复执行。	应用指令执行时
5081	以太网套接字未打开，无法操作	继续运行	检查 SOCOPEN 指令（d2）参数的已链接标志位是否置位。	应用指令执行时
5082	以太网指令输入的套接字 ID 超出范围	继续运行	修改应用指令参数	应用指令执行时
5083	创建 TCP 服务器失败	继续运行	检查链接是否占满	应用指令执行时

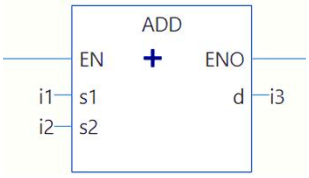
5084	创建链接失败	继续运行	检查链接是否占满	应用指令执行时
5086	以太网指令使用的套接字 ID 未在上位机进行配置或配置后未使能	继续运行	检查上位机以太网配置	应用指令执行时
5087	SOCRECV 指令接收超时	继续运行	1、检查网络链接是否正常 2、检查网络对端是否有数据发出	应用指令执行时
5088	SOCMTCP 指令指定套接字使用了非 TCP 客户端的配置模式	继续运行	检查上位机以太网配置	应用指令执行时
5089	以太网套接字配置 TCP 服务器时，指定本机端口为 502	继续运行	502 端口已被内部占用，请修改本机端口号	应用指令执行时
508A	UDP 端口制定为 1092	继续运行	UDP 端口 1092 被系统内部占用无法使用，请修改本机端口号	应用指令执行时
5090	网线链接异常	继续运行	检查网线是否链接上	应用指令执行时

14 内置 FC/FB

14.1 内置 FC 函数

1、ADD/加法运算指令

ADD



将 s1、s2 中指定的软元件或者变量相加输出到 d。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ADD	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

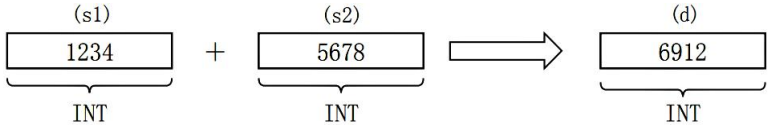
- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
ADD	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME

功能

- 将(s1)中指定的数据与(s2)中指定的数据进行加法运算，并将加法运算结果存储到(d)中，将运算结果从(d)以与(s)相同的数据类型进行输出。
- 数据类型为 INT 型的情况下：



注意事项

- 输出最好与输入为相同类型，否则可能出现高位数据丢失。若是输出的结果长度大于（d）的长度，将会进行截断，将高位舍弃，留下（d）长度的数据；
- 整数类型在运算时发生了上溢或者下溢出会进行循环计数；
- 实数类型在运算时发生了上溢或者下溢时运算出错，输出不定值；
- NAN：not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生；
- INF：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF；

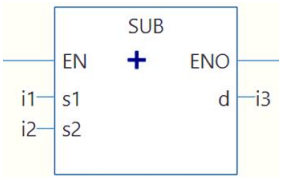
  计算后指数（e）和尾数（f）都为 0 时，等于标志变 ON，当结果的绝对值小于浮点数据能够表达的最小值，计算结果也输出 0，这时下溢出标志变 ON。

福州富昌维控电子科技有限公司

▪ 时间类型之间的相加规则：TIME+TIME = TIME，TOD+TIME = TOD，DT+TIME = DT，DATE + TIME=DATE。除了 DATE、DT 类型底层是精确到秒(s)的，其他的 TIME、TOD 类型是精确到毫秒(ms)，因此精确度不同的类型相加的时候，需要把以毫秒(ms)为单位的换算成秒(s)为单位的，再进行加法运算。然后看输出类型的精确度，看是否再需要转换单位进行输出。

2、SUB/减法运算指令

SUB



将(s1)中指定的数据与(s2)中指定的数据进行减法运算，并将结果存储到(d)中。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SUB	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

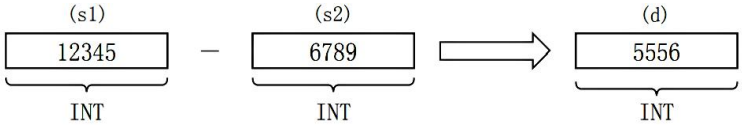
- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
SUB	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME

功能

- 将(s1)中指定的数据与(s2)中指定的数据进行减法运算，并将结果存储到(d)中。将运算结果从(d)以与(s)相同的数据类型进行输出。
- 数据类型为 INT 型的情况下：



注意事项

- 输出最好与输入为相同类型，否则可能出现高位数据丢失。若是输出的结果长度大于（d）的长度，将会进行截断，将高位舍弃，留下（d）长度的数据；
- 整数类型在运算时发生了上溢或者下溢出会进行循环计数；
- 实数类型在运算是发生了上溢或者下溢时运算出错，输出不定值；
- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生；
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF；
- 时间类型之间的相减规则：TIME-TIME = TIME，DATE-DATE = TIME，DATE-TIME = DATE，TOD-TIME = TOD，TOD-TOD = TIME，DT-TIME = DT，DT-DT = TIME 。DATE、DT 类型底层是精确到秒(s)的，TIME、TOD 类型是精确到毫秒(ms)。分为三种：

福州富昌维控电子科技有限公司

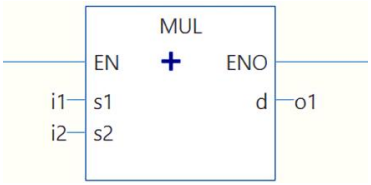
① 输入跟输出精确度全部都一样。例如：TIME-TIME = TIME，TOD-TIME = TOD，TOD-TOD = TIME，不需要进行单位转换，直接相减，直接输出；

② 输入精确度一样，输出不一样。例如：DATE-DATE = TIME,DT-DT = TIME，这个相减完需要判断类型的精确度，DATE 和 DT 类型以秒(s)为单位的需要转换成毫秒(ms)为单位进行输出；

③ 输入精确度不一样。例如：DT-TIME = DT，DATE-TIME = DATE。两个需要相减的类型的精确度不一致时，需要把以毫秒(ms)为单位的转换成秒(s)为单位，进行减法运算，然后再判断输出类型的精确度，看是否需要转换单位进行输出。

3、MUL/乘法运算指令

MUL



将(s1)中指定的数据与(s2)中指定的数据进行乘法运算，并将结果存储到(d)中。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MUL	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

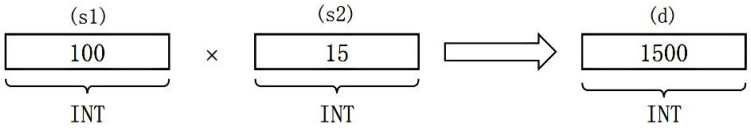
- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
MUL	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME

功能

- 将(s1)中指定的数据与(s2)中指定的数据进行乘法运算，并将结果存储到(d)中。
- 数据类型为 INT 型的情况下：



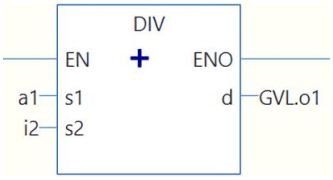
注意事项

- 在 INT 类型下运算即使运算结果超过了 INT 型数据范围的情况下，也输出 INT 型的数据。(运算结果变为 DINT 型，但是输出时以删除了高 16 位的 INT 型的数据被输出。或者使用 DINT 类型去接收输出值；
- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生；
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF。

福州富昌维控电子科技有限公司

4、DIV/除法运算指令

DIV



将(s1)中指定的数据与(s2)中指定的数据进行除法运算，并将结果存储到(d)中。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DIV	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

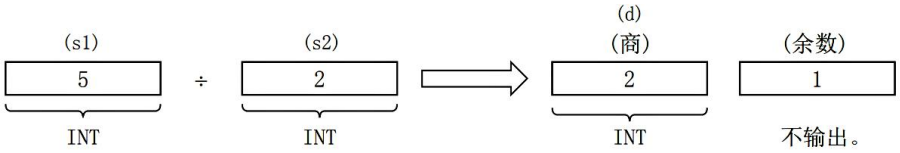
- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
DIV	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME

功能

- 将(s1)中指定的数据与(s2)中指定的数据进行除法运算，将运算结果的商从(d)以与(s)相同的数据类型进行输出。
- 数据类型为 INT 型的情况下：



注意事项

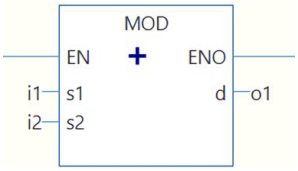
- 在 INT/DINT 类型下运算下只会输出整数的商；
- 在 REAL 类型的运算下，会输出实数的运算结果；
- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生；
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF。

错误码

错误码	内容
4080H	除数(s2)输入为 0。

5、MOD/取余运算指令

MOD



输出输入值的余数((s1) (s2))。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MOD	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

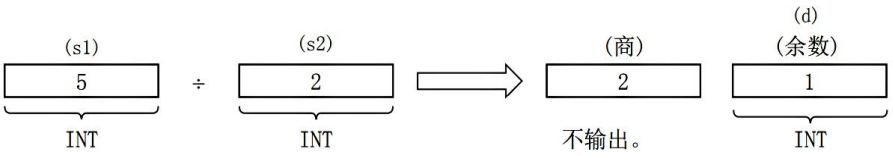
- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
MOD	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT

功能

- 进行(s1)、(s2)中输入的 INT 型/DINT 型数据的除法运算((s1) (s2))，将运算结果的余数从(d)以与(s)相同的数据类型进行输出。



注意事项

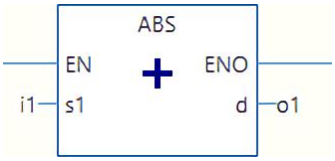
- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生；
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF。

错误码

错误码	内容
4080H	除数(s2)输入为 0。

6、ABS/求绝对值（整数）

ABS



计算 s1 的绝对值，输出到 d 中。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ABS	输入 s1															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
ABS	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT

功能

- 计算 s1 的绝对值，输出到 d 中。

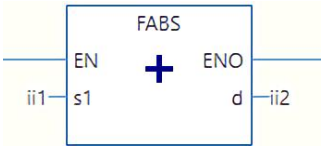
d = |s1|

注意事项

- (s1)的数据类型为 INT 型且输入了-32768 的情况下，从(d)输出-32768；
- (s1)的数据类型为 DINT 型且输入了-2147483648 的情况下，从(d)输出-2147483648。

7、FABS/求绝对值（实数）

FABS



计算 s1 的绝对值，输出到 d 中。

不支持使用软元件

使用的变量类型

指令	参数	变量类型
FABS	输入 s1	REAL、LREAL
	输出 d	REAL、LREAL

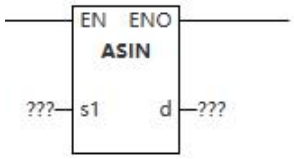
功能

- 计算 s1 的绝对值，输出到 d 中。

d = |s1|

8、ASIN/SIN^{-1}运算

ASIN



对(s1)中指定角度的 SIN^{-1}(反正弦)值进行运算后，将运算结果存储到(d)中。

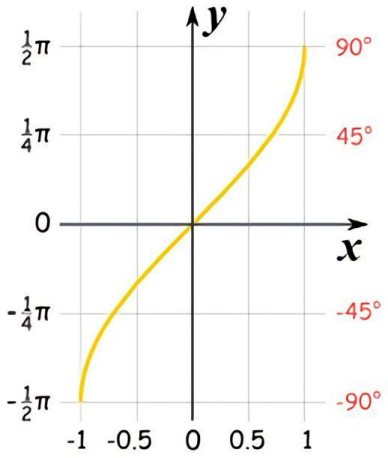
不支持使用软元件

使用的变量类型

指令	参数	变量类型
ASIN	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

- 对(s1)中指定角度的 SIN^{-1}(反正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中；
- (s1)中指定的 SIN^{-1} 值可在-1.0~1.0 的范围内设置；
- (d)中存储的角度(运算结果)以弧度单位存储(-π/2)~(π/2)的值。



ASIN 图形

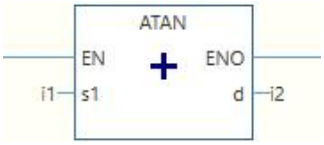
错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

福州富昌维控电子科技有限公司

9、ATAN/TAN^{-1}运算

ATAN



对(s1)中指定角度的 TAN^{-1}(反正切)值进行运算后，将运算结果存储到(d)中。

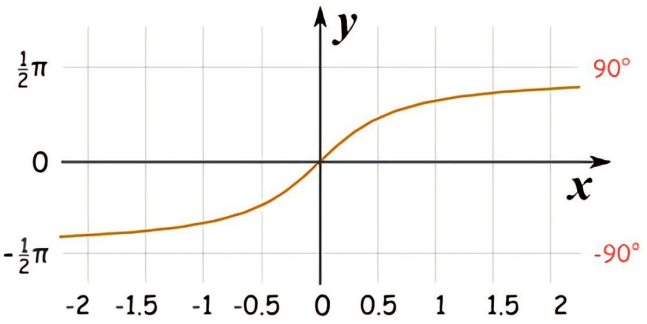
不支持使用软元件

使用的变量类型

指令	参数	变量类型
ATAN	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

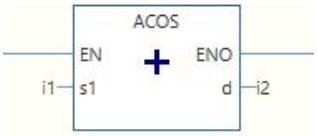
- 对(s1)中指定角度的 TAN^{-1}((反正切)值进行运算后，将运算结果存储到(d)中；
- (s1)中指定的 TAN^{-1}值可在 1.17549 -38~3.40282+38 的范围内设置；
- (d)中存储的角度(运算结果)以弧度单位存储(-π/2)～(π/2)的值。



ATAN 图形

10、ACOS/COS^{-1}运算

ACOS



对(s1)中指定角度的 COS^{-1}(反余弦)值进行运算后，将运算结果存储到(d)中。

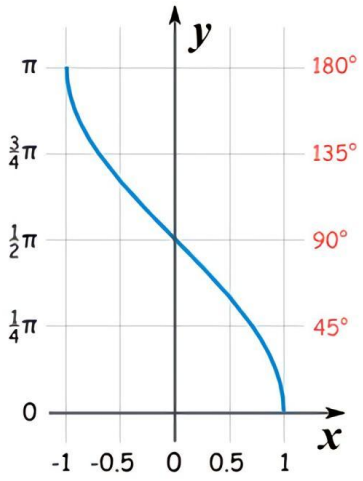
不支持使用软元件

使用的变量类型

指令	参数	变量类型
ACOS	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

- 对(s)中指定角度的 COS^{-1}(反余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中；
- (s)中指定的 COS 值可在-1.0~1.0 的范围内设置；
- (d)中存储的角度(运算结果)以弧度单位存储 0~π 的值。



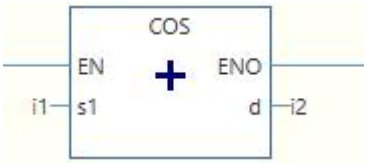
ACOS 图形

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

11、COS/COS 运算

COS



对(s1)中指定角度的 COS(余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

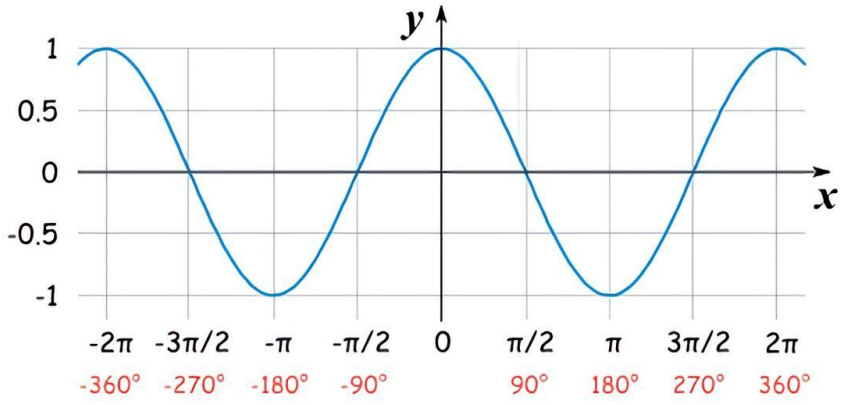
不支持使用软元件

使用的变量类型

指令	参数	变量类型
COS	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

- 对(s1)中指定角度的 COS(余弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。
- 对于(d)中指定的角度，以弧度单位(角度×π÷180)进行设置。



COS 的图形

12、SIN/SIN 运算

SIN

对(s1)中指定角度的 SIN(正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

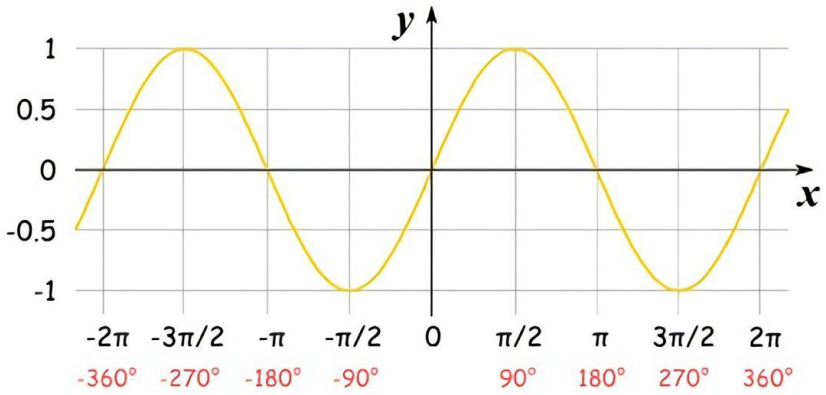
不支持使用软元件

使用的变量类型

指令	参数	变量类型
SIN	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

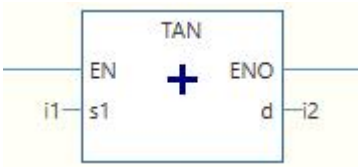
- 对(s1)中指定角度的 SIN(正弦)值进行运算后，将运算结果存储到(d)中指定的软元件编号中；
- 对于(s)中指定的角度，以弧度单位(角度×π÷180)进行设置。



SIN 的图形

13、TAN/TAN 运算

TAN



对(s1)中指定角度的 TAN(正切)值进行运算后，将运算结果存储到(d)中指定的软元件编号中。

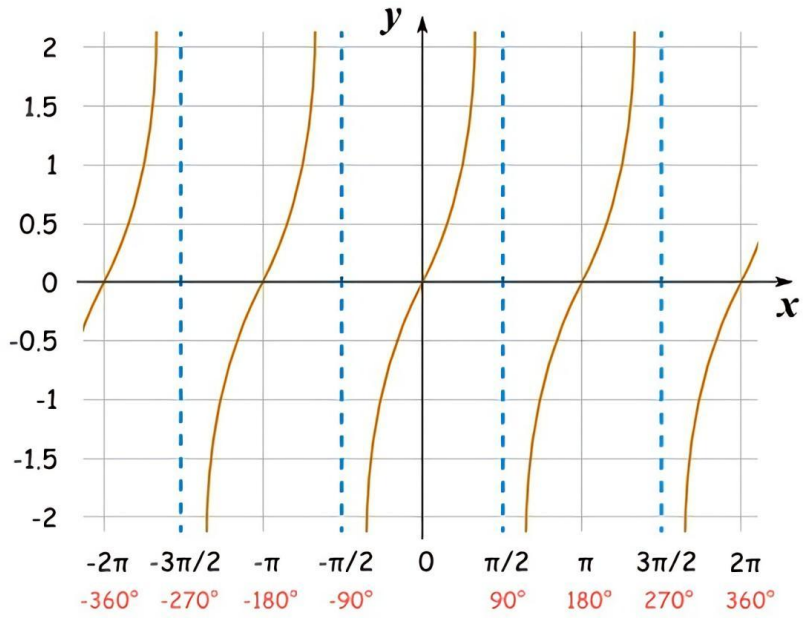
不支持使用软元件

使用的变量类型

指令	参数	变量类型
SIN	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

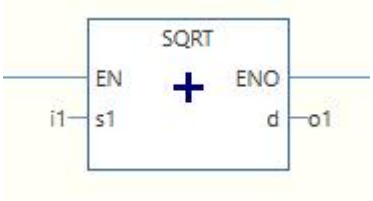
- 对(s1)中指定角度的 TAN(正切)值进行运算后，将运算结果存储到(d)中指定的软元件编号中；
- 对于(s1)中指定的角度，以弧度单位(角度×π÷180)进行设置。



TAN 的图形

14、SQRT/平方根运算

SQRT



对(s1)进行平方根运算。

不支持使用软元件

使用的变量类型

指令	参数	变量类型
SQRT	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

- 从(d)输出(s1)中输入的 REAL 型数据的平方根；
- 如果将输入值置为 A，运算输出值置为 B，其情况如下所示：

$$B=\sqrt{A}$$

- (s1)的输入值为 REAL 或长 LREAL 型的数据值且在正数范围内。

注意事项

- s1 不能为负值，否则可能会出现运算出错；
- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生；
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

15、EXP/指数运算

EXP

进行(s1)中指定的值的指数运算后，将运算结果存储到(d)中指定的软元件中。

不支持使用软元件

使用的变量类型

指令	参数	变量类型
EXP	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

- 进行(s)中指定的值的指数运算后，将运算结果存储到(d)中指定的软元件编号中；
- 在指数运算中，将底(e)以“2.718282”进行运算；
- 如果将输入值置为 A，运算输出值置为 B，其情况如下所示。

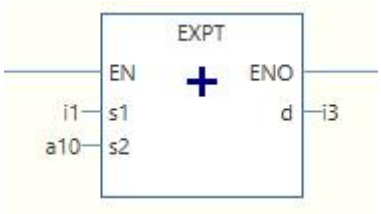
$B=e^{A}$

注意事项

- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生。
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF。

16、EXPT/幂运算

EXPT



进行(s1)中指定的值进行（s2）的幂运算后，将运算结果存储到(d)中指定的软元件中。

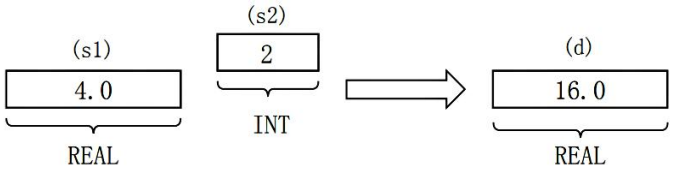
不支持使用软元件

使用的变量类型

指令	参数	变量类型
EXPT	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

- 进行(s1)中指定的值进行（s2）的幂运算后，将运算结果存储到(d)中指定的软元件中。

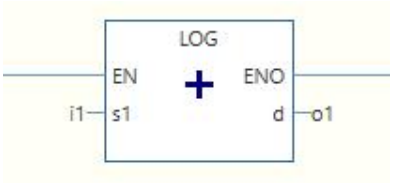


注意事项

- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生。
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF。

17、LOG/对数运算

LOG



(s1)中指定的值进行以 10 为底的对数运算后，将运算结果存储到(d)中。

不支持使用软元件

使用的变量类型

指令	参数	变量类型
LOG	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

- 进行(s1)中指定的值进行以 10 为底的对数运算后，将运算结果存储到(d)中；
- 如果将输入值置为 A，运算输出值置为 B，其情况如下所示：

  $B = \log 10^{A}$

- (s1)的输入值是 REAL 型的数据值；
- (s1)中指定的值只能设置为正的数；(负数不能进行运算)
- 运算结果为-0 或发生了下溢时，将运算结果作为 0 进行输出。

注意事项

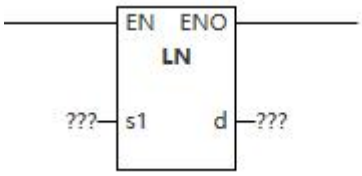
- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生；
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

18、LN/自然对数运算

LN



进行(s1)中指定的值进行以 e 为底的对数运算后，将运算结果存储到(d)中。

不支持使用软元件

使用的变量类型

指令	参数	变量类型
LN	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL
	输出 d	REAL、LREAL

功能

- 进行(s1)中指定的值进行以“e”为底的对数运算后，将运算结果存储到(d)中；
- 如果将输入值置为 A，运算输出值置为 B，其情况如下所示：

  $B = \log e^{A}$

- 自然对数运算中，将底“e”作为 2.71828 进行运算。

注意事项

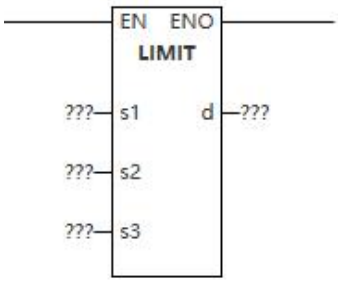
- nan:not a number，表示“非数值”，计算如 0.0/0.0，∞/∞，或∞－∞结果不等于一个数字或无穷大时产生；
- inf：infinite，表示“无穷大”，当结果的绝对值大于浮点数数据能够表达的最大值，则为上溢出，结果输出±INF。如果结果为正，输出为 + INF；如果为负，输出为 - INF。

错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

19、LIMIT/上下限位控制

LIMIT



判断输入数据是否在最小值和最大值之间，若输入数据在二者之间，则直接把输入数据作为输出数据进行输出。若输入数据大于最大值，则把最大值作为输出值。若输入数据小于最小值，则把最小值作为输出值。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LIMIT	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输入 s3															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
LIMIT	输入 s1	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s3	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME

功能

- 判断输入数据是否在最小值和最大值之间，若输入数据在二者之间，则直接把输入数据作为输出数据进行输出。若输入数据大于最大值，则把最大值作为输出值。若输入数据小于最小值，则把最小值作为输出值；
- (s1)为最小值，(s3)为最大值，（s2）为比较值；
- (s2)的输入值＞(s3)的输入值的情况下，从(d)输出(s3)的输入值；
- (s2)的输入值＜(s1)的输入值的情况下，从(d)输出(s1)的输入值；
- (s1)的输入值＜(s2)的输入值＜(s3)的输入值的情况下，从(d)输出(s2)的输入值。

注意事项

- 输入和输出的类型需要相同，否则可能会出现数据缺失的情况；
- (s3)的值需要大于(s1)的值，否则将运算出错，d1 将不输出值。

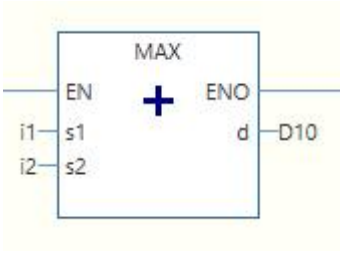
错误码

错误码	内容
4084H	输入了超出可指定范围的数据。

福州富昌维控电子科技有限公司

20、MAX/取最大值

MAX



在两个输入数据中选择最大值作为输出。

使用的软元件使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MAX	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
MAX	输入 s1	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME

功能

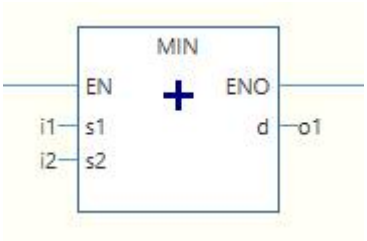
- 在两个输入数据中选择最大值作为输出；
- 输出和输入同类型。

注意事项

- 请使用同类型进行比较。

21、MIN/取最小值

MIN



在两个输入数据中选择最小值作为输出。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MIN	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
MIN	输入 s1	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME

功能

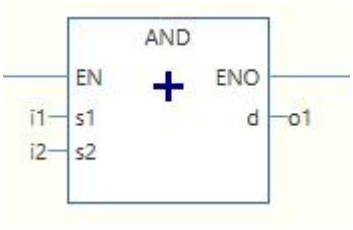
- 在两个输入数据中选择最小值作为输出；
- 输出和输入同类型。

注意事项

- 请使用同类型进行比较。

22、AND/逻辑与

AND



变量或常量的相与运算。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
AND	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

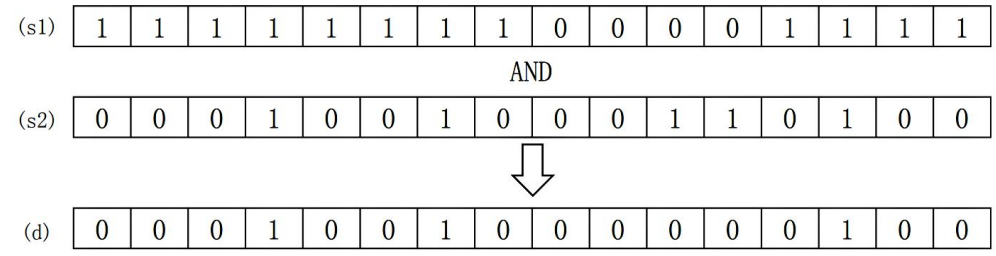
使用的变量类型

指令	参数	变量类型
AND	输入 s1	BOOL、BYTE、WORD、DWORD、LWORD
	输入 s2	BOOL、BYTE、WORD、DWORD、LWORD
	输出 d	BOOL、BYTE、WORD、DWORD、LWORD

功能

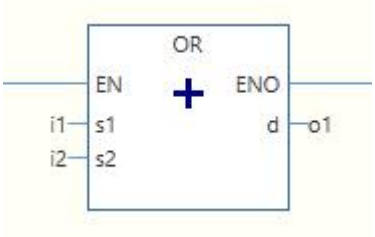
- 对(s1)（s2）中输入的 BOOL 型/WORD 型/DWORD 型数据按各个位进行逻辑积运算，将运算结果从(d)以与(s)相同的数据类型进行输出。

数据类型为 WORD 型的情况下:



23、OR/逻辑或

OR



变量或常量的“或”运算。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
OR	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

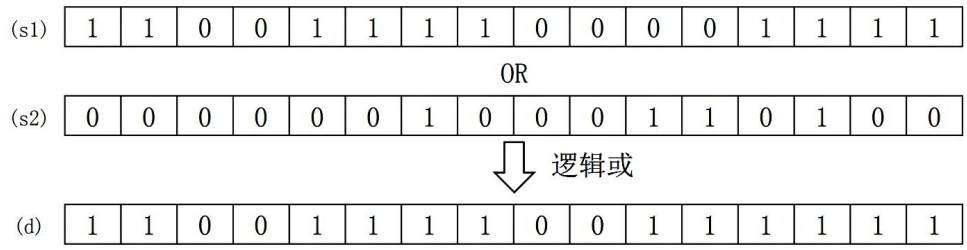
使用的变量类型

指令	参数	变量类型
OR	输入 s1	BOOL、BYTE 、WORD、DWORD、LWORD
	输入 s2	BOOL、BYTE 、WORD、DWORD、LWORD
	输出 d	BOOL、BYTE 、WORD、DWORD、LWORD

功能

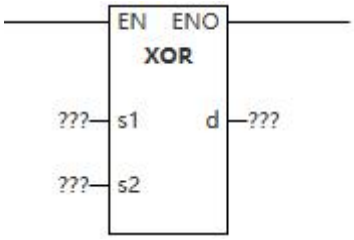
- 对(s1)(s2)中输入的 BOOL 型/WORD 型/DWORD 型数据按各个位进行逻辑或运算，将运算结果从(d)以与(s)相同的数据类型进行输出。

数据类型为 WORD 型的情况下:



24、XOR/异或

XOR



变量或常量的异或运算。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
XOR	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

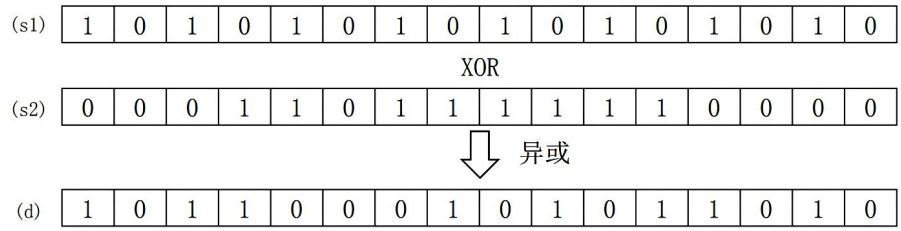
使用的变量类型

指令	参数	变量类型
XOR	输入 s1	BOOL、BYTE、WORD、DWORD、LWORD
	输入 s2	BOOL、BYTE、WORD、DWORD、LWORD
	输出 d	BOOL、BYTE、WORD、DWORD、LWORD

功能

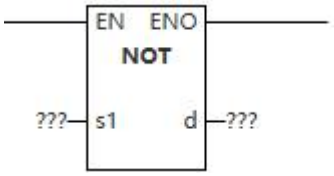
- 对(s1)(s2)中输入的 BOOL 型/WORD 型/DWORD 型数据按各个位进行异或运算，将运算结果从(d)以与(s)相同的数据类型进行输出。

数据类型为 WORD 型的情况下：



25、NOT/取非

NOT



变量或常量的取非运算。逐位取非。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
NOT	输入 s1															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
NOT	输入 s1	BOOL、BYTE 、WORD、DWORD、LWORD
	输出 d	BOOL、BYTE 、WORD、DWORD、LWORD

功能

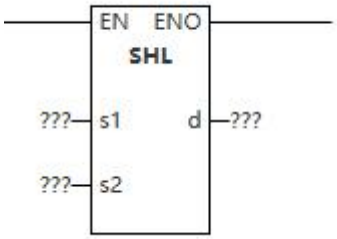
- 对(s)中输入的 BOOL 型/WORD 型/DWORD 型数据按各个位进行逻辑否运算，将运算结果从(d)以与(s)相同的数据类型进行输出。

数据类型为 WORD 型的情况下：

(s)	0	1	1	0	1	0	1	1	0	0	0	0	1	1	1	1
	NOT															
(d)	1	0	0	1	0	1	0	0	1	1	1	1	0	0	0	0

26、SHL/左移指令

SHL



对操作数进行按位左移，左边移出的位不作处理，右边自动补 0。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SHL	输入 s1																										
	输入 s2															●	●	●	●	●							
	输出 d																										

软元件使用注意事项

- T、C、D、R、SD 为有符号 16 位数据类型。

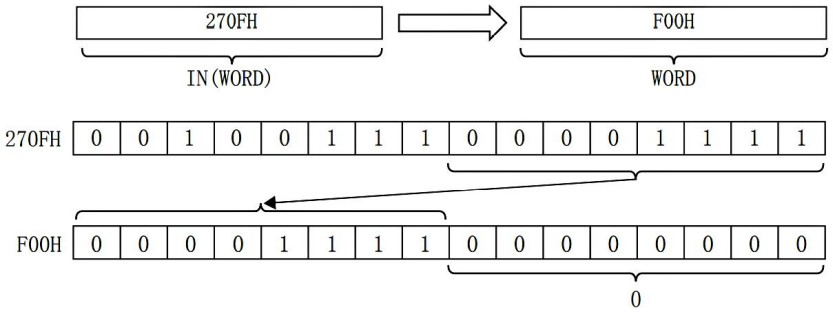
使用的变量类型

指令	参数	变量类型
SHL	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT
	输入 s2	INT
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT

功能

- 将(s1)中输入的 16 位数据/32 位数据(WORD 型/DWORD 型)左移(s2)位后，从(d)以与(s1)相同的数据类型进行输出；
- 从最低位开始的(s2)位将变为 0；
- (s1)的输入值为 16 位数据/32 位数据(WORD 型/DWORD 型)的数据。

(s1)的数据为 16 位数据（WORD 型），（s2）的输入值为 8 的情况下：



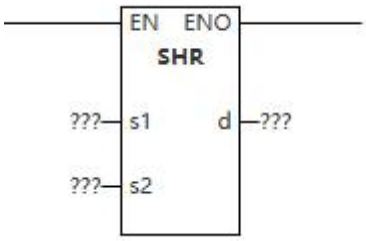
注意事项

- 至(n)的输入值(移位数指定)为 INT 型的数据值且下述所示范围内的值：
  ① 输入为 16 位数据时（s2）的范围是[0,15]。超出的部分会进行取余处理；
  ② 输入为 32 为数据时（s2）的范围是[0,31]。超出的部分会进行取余处理。
- 当位移的值 s2<0，可以转成 s2 = s2 + 输出数据类型位数。例如，当输入输出类型为 16 位的 int 类型时，右移 s2 =-3，即等价于右移 s2 =16+(-3) = 13。

福州富昌维控电子科技有限公司

27、SHR/右移指令

SHR



对操作数进行按位右移，右边移出的位不作处理，左边自动补 0。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SHR	输入 s1																										
	输入 s2															●	●	●	●	●							
	输出 d																										

软元件使用注意事项

- T、C、D、R、SD 为有符号 16 位数据类型。

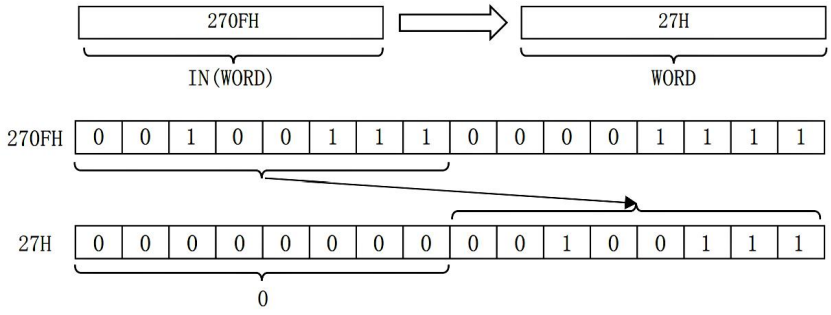
使用的变量类型

指令	参数	变量类型
SHR	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT
	输入 s2	INT
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT

功能

- 将(s1)中输入的 16 位数据/32 位数据(WORD 型/DWORD 型)右移(s2)位后，从(d)以与(s1)相同的数据类型进行输出；
- 右移的位数(n)为(s2)的值；
- 从最高位开始的(s2)位将变为 0；
- 至(s1)的输入值为 16 位数据/32 位数据(WORD 型/DWORD 型)的数据。

(s1)的数据为 16 位数据（WORD 型），（s2）的输入值为 8 的情况下：



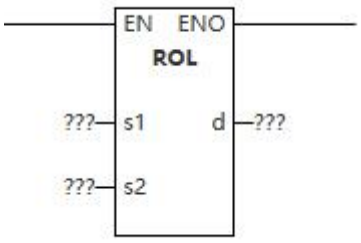
注意事项

- 至(s2)的输入值(移位数指定)为 INT 型的数据值且下述所示范围内的值：
  ① 输入为 16 位数据时（s2）的范围是[0,15]。超出的部分会进行取余处理；
  ② 输入为 32 为数据时（s2）的范围是[0,31]。超出的部分会进行取余处理。
- 当位移的值 s2<0，可以转成 s2 = s2 + 输出数据类型位数。例如，当输入输出类型为 16 位的 int 类型时，右移 s2 =-3，即等价于右移 s2 =16+(-3) = 13。

福州富昌维控电子科技有限公司

28、ROL/循环左移指令

ROL



对操作数进行按位循环左移，左边移出的位直接补充到右边最低位。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ROL	输入 s1																									
	输入 s2															●	●	●	●	●						
	输出 d																									

软元件使用注意事项

- T、C、D、R、SD 为有符号 16 位数据类型

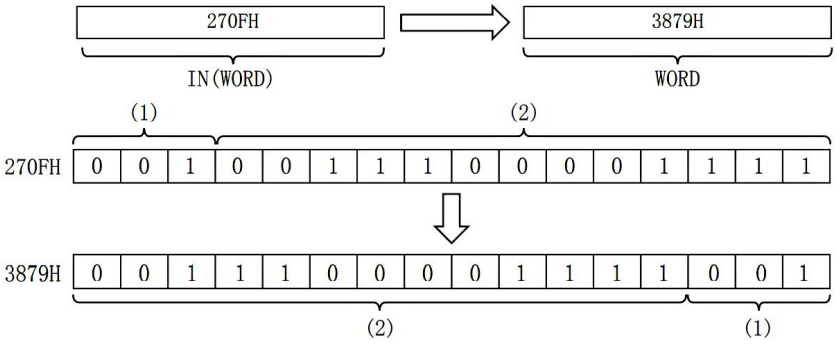
使用的变量类型

指令	参数	变量类型
ROL	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT
	输入 s2	INT
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT

功能

- 将(s1)中输入的 16 位数据/32 位数据(WORD 型/DWORD 型)向左(s2)位回转(旋转)后，从(d)以与(s1)相同的数据类型进行输出；
- 向左回转的(s2)位数为输入至(n)的值；
- 至(s1)的输入值为 16 位数据/32 位数据(WORD 型/DWORD 型)的数据。

（s1）的数据为 16 位数据（WORD 型），（s2）的输入值为 3 的情况下。（向左回转 3 位。）



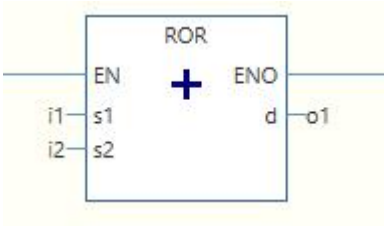
注意事项

- 至(s2)的输入值(移位数指定)为 INT 型的数据值且下述所示范围内的值：
  ① 1、输入为 16 位数据时（s2）的范围是[0,15]。超出的部分会进行取余处理；
  ② 2、输入为 32 为数据时（s2）的范围是[0,31]。超出的部分会进行取余处理。
- 当位移的值 s2<0，可以转成 s2 = s2 + 输出数据类型位数。例如，当输入输出类型为 16 位的 int 类型时，右移 s2 =-3，即等价于右移 s2 =16+(-3) = 13。

福州富昌维控电子科技有限公司

29、ROR/循环右移指令

ROR



对操作数进行按位循环右移，右边移出的位直接补充到左边最高位。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
ROR	输入 s1																										
	输入 s2															●	●	●	●	●							
	输出 d																										

软元件使用注意事项

- T、C、D、R、SD 为有符号 16 位数据类型。

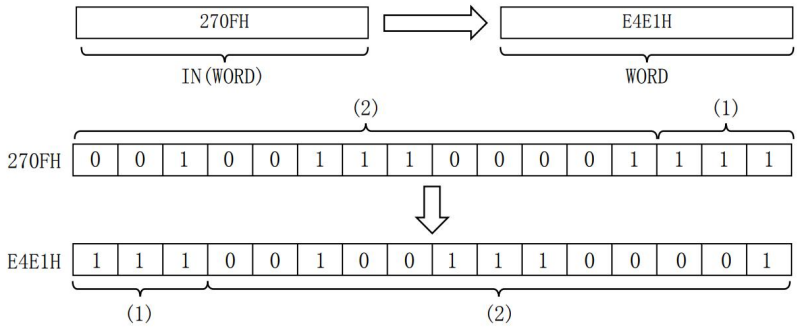
使用的变量类型

指令	参数	变量类型
ROR	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT
	输入 s2	INT
	输出 d	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT

功能

- 将(s1)中输入的 16 位数据/32 位数据(WORD 型/DWORD 型)向右(s2)位回转(旋转)后，从(d)以与(s1)相同的数据类型进行输出；
- 向循环右移转的(s2)位数为输入至(s2)的值；
- 至(s1)的输入值为 16 位数据/32 位数据(WORD 型/DWORD 型)的数据。

(s1)的数据为 16 位数据(WORD 型)，至(s2)的输入值为 3 的情况下。(向右回转 3 位。)

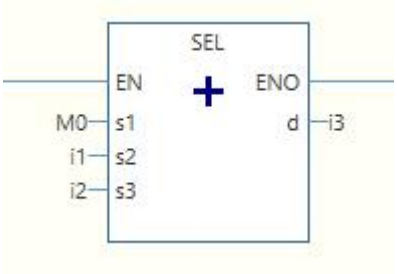


注意事项

- 至(s2)的输入值(移位数指定)为 INT 型的数据值且下述所示范围内的值：
  ① 输入为 16 位数据时（s2）的范围是[0,15]。超出的部分会进行取余处理；
  ② 输入为 32 为数据时（s2）的范围是[0,31]。超出的部分会进行取余处理。
- 当位移的值 s2<0，可以转成 s2 = s2 + 输出数据类型位数。例如，当输入输出类型为 16 位的 int 类型时，右移 s2 =-3，即等价于右移 s2 =16+(-3) = 13。

30、SEL/二选一指令

SEL



通过选择开关在两个输入数据中选择一个作为输出，选择开关为 FALSE 时输出为第一个输入数据，选择开关为 TRUE 时输出为第二个输入数据。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SEL	输入 s1	●	●	●	●	●																				
	输入 s2															●	●	●	●	●	●	●					
	输入 s3															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- X、Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型。

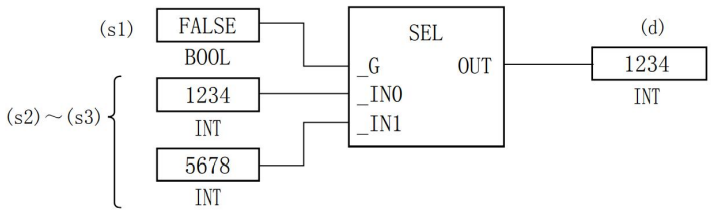
使用的变量类型

指令	参数	变量类型
SEL	输入 s1	BOOL
	输入 s2	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s3	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME

功能

- 按照输入至(s1)的输入值，将输入至(s2)、(s3)的值的某个，从(d)以与(s2)、(s3)相同的数据类型进行输出；
- (s1)的输入值为 FALSE(=0)的情况下，从(d)输出(s2)的输入值；
- (s1)的输入值为 TRUE(=1)的情况下，从(d)输出(s3)的输入值。

(s2)、(s3)的数据类型为 INT 型的情况下(自变量名的(s2)、(s3)对应于(s1)的位值(0 或 1)。)

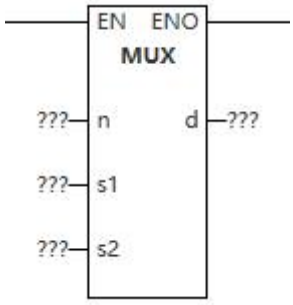


注意事项

- 若 d 的数据与 s2、s3 不同，可能会导致数据缺失。

31、MUX/多路复用器

MUX



按照输入至(n)的输入值，将输入至(s1)～(s28)的值的某个从(d)以与(s)相同的数据类型进行输出。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MUX	输入 n															●	●	●	●	●							
	输入 s1	●	●	●	●	●										●	●	●	●	●	●	●					
	输入 s2	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d		●	●	●	●										●	●	●	●	●	●	●					

软元件使用注意事项

- X、Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型。

使用的变量类型

指令	参数	变量类型
MUX	输入 n	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT
	输入 s1	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME

功能

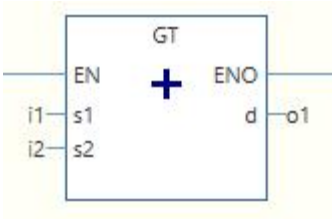
- 按照输入至(n)的输入值，将输入至(s1)～(s28)的值的某个从(d)以与(s)相同的数据类型进行输出；
- n=0，不输出；
- n≠0，输出 s(n)。

注意事项

- 若 d 的数据与输入不同，可能会导致数据缺失。

32、GT/大于指令

GT



判断两个操作数的大小，当第一个数大于第二个数时输出 TRUE，否则输出为 FALSE。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
GT	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d		●	●	●	●																					

软元件使用注意事项

- Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型。

使用的变量类型

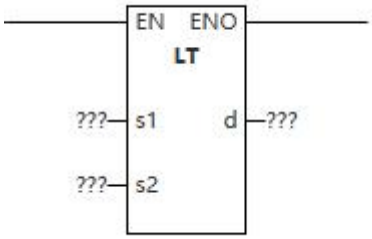
指令	参数	变量类型
GT	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL

功能

- 判断两个操作数的大小，当第一个数大于第二个数时输出 TRUE，否则输出为 FALSE。

33、LT/小于指令

LT



判断两个操作数的大小，当第一个数小于第二个数时输出 TRUE，否则输出为 FALSE。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LT	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d		●	●	●	●																					

软元件使用注意事项

- Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型。

使用的变量类型

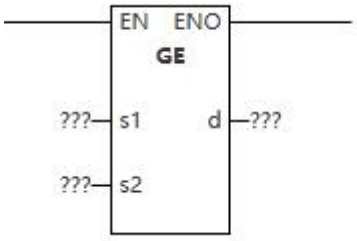
指令	参数	变量类型
LT	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL

功能

- 判断两个操作数的大小，当第一个数小于第二个数时输出 TRUE，否则输出为 FALSE。

34、GE/大于等于指令

GE



判断两个操作数的大小，当第一个数大于等于第二个数时输出 TRUE，否则输出为 FALSE。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
GE	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d		●	●	●	●																					

软元件使用注意事项

- Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型。

使用的变量类型

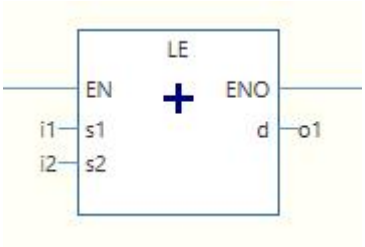
指令	参数	变量类型
GE	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL

功能

- 判断两个操作数的大小，当第一个数大于等于第二个数时输出 TRUE，否则输出为 FALSE。

35、LE/小于等于指令

LE



判断两个操作数，当第一个数小于等于第二个数时输出 TRUE，否则输出为 FALSE。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LE	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d		●	●	●	●																					

软元件使用注意事项

- Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型。

使用的变量类型

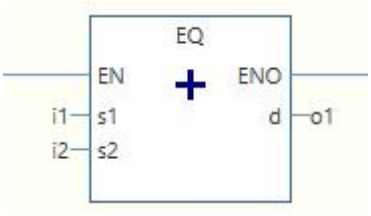
指令	参数	变量类型
LE	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL

功能

- 判断两个操作数，当第一个数小于等于第二个数时输出 TRUE，否则输出为 FALSE。

36、EQ/比较指令（等于）

EQ



判断两个操作数，当第一个数等于第二个数时输出 TRUE，否则输出为 FALSE。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
EQ	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d		●	●	●	●																					

软元件使用注意事项

- Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型。

使用的变量类型

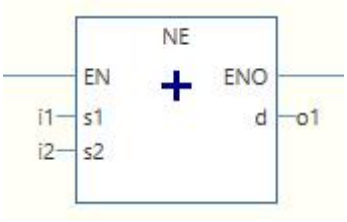
指令	参数	变量类型
EQ	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL

功能

- 判断两个操作数的大小，当第一个数等于第二个数时输出 TRUE，否则输出为 FALSE。

37、NE/比较指令（不等于）

NE



判断两个操作数，当第一个数不等于第二个数时输出 TRUE，否则输出为 FALSE。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
NE	输入 s1															●	●	●	●	●	●	●					
	输入 s2															●	●	●	●	●	●	●					
	输出 d		●	●	●	●																					

软元件使用注意事项

- Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型。

使用的变量类型

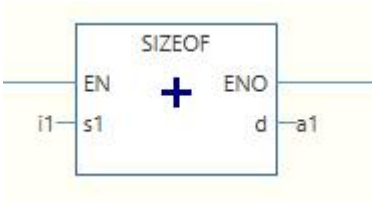
指令	参数	变量类型
NE	输入 s1	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输入 s2	BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL

功能

- 判断两个操作数，当第一个数不等于第二个数时输出 TRUE，否则输出为 FALSE。

38、SIZEOF/获取大小指令

SIZEOF



获取（s1）指定变量的大小。

不支持使用软元件

使用的变量类型

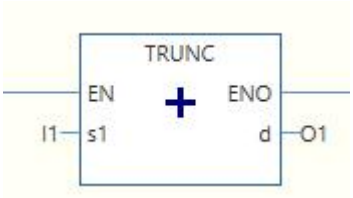
指令	参数	变量类型
SIZEOF	输入 s1	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	WORD

功能

- 获取（s1）指定变量的大小。

39、TRUNC/截尾取整指令

TRUNC



取整。

不支持使用软元件

使用的变量类型

指令	参数	变量类型
TRUNC	输入 s1	REAL、LREAL
	输出 d	INT、DINT

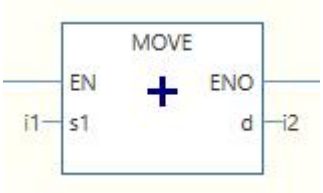
功能

- 将 s1 的值取整，将小数点后的数据截去。

福州富昌维控电子科技有限公司

40、MOVE/传送指令

MOVE



将(s1)中指定的变量的值代入至(d)中指定的变量中。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
MOVE	输入 s1															●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●	●	●					

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型
MOVE	输入 s1	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME
	输出 d	BOOL、BYTE、WORD、DWORD、LWORD、SINT、USINT、INT、UINT、DINT、UDINT、LINT、REAL、LREAL、TIME、TIME_OF_DAY、DATE、DATE_AND_TIME、LTIME

功能

- 将(s1)中指定的变量的值代入至(d)中指定的变量中；
- 对于(s1)、(d)，可以指定 BOOL 型/INT 型/DINT 型/WORD 型/DWORD 型/LWORD 型/REAL 型/TIME 型。此外，(s1)与(d)仅可以指定相同的数据类型。

41、BOOL_TO_<TYPE>/BOOL 类型转换指令

BOOL_TO_<TYPE>

功能

- 将 BOOL 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BOOL_TO_BYTE	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_DATE	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_DINT	输入 s1	●	●	●	●	●																				
	输出 d																				●	●				
BOOL_TO_DT	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_DWORD	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_INT	输入 s1	●	●	●	●	●																				
	输出 d															●	●	●	●	●						
BOOL_TO_REAL	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_SINT	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_TIME	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_TOD	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_UDINT	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_UINT	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_USINT	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_WORD	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_LREAL	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_LWORD	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_LINT	输入 s1	●	●	●	●	●																				
	输出 d																									
BOOL_TO_LTIME	输入 s1	●	●	●	●	●																				
	输出 d																									

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
BOOL_TO_BYTE	输入 s1	BOOL	BOOL_TO_TOD	输入 s1	BOOL
	输出 d	BYTE		输出 d	TIME_OF_DAY
BOOL_TO_DATE	输入 s1	BOOL	BOOL_TO_UDINT	输入 s1	BOOL
	输出 d	DATE		输出 d	UDINT
BOOL_TO_DINT	输入 s1	BOOL	BOOL_TO_UINT	输入 s1	BOOL
	输出 d	DINT		输出 d	UINT
BOOL_TO_DT	输入 s1	BOOL	BOOL_TO_USINT	输入 s1	BOOL
	输出 d	DATE_AND_TIME		输出 d	USINT
BOOL_TO_DWORD	输入 s1	BOOL	BOOL_TO_WORD	输入 s1	BOOL
	输出 d	DWORD		输出 d	WORD
BOOL_TO_INT	输入 s1	BOOL	BOOL_TO_LREAL	输入 s1	BOOL
	输出 d	INT		输出 d	LREAL
BOOL_TO_REAL	输入 s1	BOOL	BOOL_TO_LWORD	输入 s1	BOOL
	输出 d	REAL		输出 d	LWORD
BOOL_TO_SINT	输入 s1	BOOL	BOOL_TO_LINT	输入 s1	BOOL
	输出 d	SINT		输出 d	LINT
BOOL_TO_TIME	输入 s1	BOOL	BOOL_TO_LTIME	输入 s1	BOOL
	输出 d	TIME		输出 d	LTIME

使用说明

- 若 BOOL 等于 FALSE，输出的值为 0，若 BOOL 为 TRUE 输出的值为 1；
- BOOL 转 DATE 时，DATE 在底层是以秒进行计时。所以不管 BOOL 输入为何值，显示的都是 1970-01-01；
- BOOL 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当 BOOL 为 TRUE 时，输出为 1970-01-01-0:0:1。

42、BYTE_TO_<TYPE>/BYTE 类型转换指令

BYTE_TO_<TYPE>

功能

- 将 BYTE 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
BYTE_TO_BOOL	输入 s1																										
	输出 d		●	●	●	●																					
BYTE_TO_DINT	输入 s1																										
	输出 d																				●	●					
BYTE_TO_INT	输入 s1																										
	输出 d															●	●	●	●	●							

其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
BYTE_TO_BOOL	输入 s1	BYTE	BYTE_TO_TOD	输入 s1	BYTE
	输出 d	BOOL		输出 d	TIME_OF_DAY
BYTE_TO_DATE	输入 s1	BYTE	BYTE_TO_UDINT	输入 s1	BYTE
	输出 d	DATE		输出 d	UDINT
BYTE_TO_DINT	输入 s1	BYTE	BYTE_TO_UINT	输入 s1	BYTE
	输出 d	DINT		输出 d	UINT
BYTE_TO_DT	输入 s1	BYTE	BYTE_TO_USINT	输入 s1	BYTE
	输出 d	DATE_AND_TIME		输出 d	USINT
BYTE_TO_DWORD	输入 s1	BYTE	BYTE_TO_WORD	输入 s1	BYTE
	输出 d	DWORD		输出 d	WORD
BYTE_TO_INT	输入 s1	BYTE	BYTE_TO_LREAL	输入 s1	BYTE
	输出 d	INT		输出 d	LREAL
BYTE_TO_REAL	输入 s1	BYTE	BYTE_TO_LWORD	输入 s1	BYTE
	输出 d	REAL		输出 d	LWORD
BYTE_TO_SINT	输入 s1	BYTE	BYTE_TO_LINT	输入 s1	BYTE
	输出 d	SINT		输出 d	LINT
BYTE_TO_TIME	输入 s1	BYTE	BYTE_TO_LTIME	输入 s1	BYTE
	输出 d	TIME		输出 d	LTIME

使用说明

- BYTE 转 BOOL 时，若输入为 0 则输出的值为 FALSE，反之输入非 0 值，则输出 TRUE；
- BYTE 转 DATE 时，DATE 在底层是以秒进行计时。所以不管 BYTE 输入为何值，显示的都是 1970-01-01；
- BYTE 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当 BYTE 输入为 61 时，输出为 1970-01-01-0:1:1。

43、DATE_TO_<TYPE>/DATE 类型转换指令

DATE_TO_<TYPE>

功能

- 在内部，日期保存为一个 DWORD（自 1970 年 1 月 1 日以来的秒数）；
- 将 DATE 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DATE_TO_BOOL	输入 s1																										
	输出 d		●	●	●	●																					
DATE_TO_DINT	输入 s1																										
	输出 d																				●	●					
DATE_TO_INT	输入 s1																										
	输出 d															●	●	●	●	●							

其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
DATE_TO_BYTE	输入 s1	DATE	DATE_TO_TOD	输入 s1	DATE
	输出 d	BYTE		输出 d	TIME_OF_DAY
DATE_TO_BOOL	输入 s1	DATE	DATE_TO_UDINT	输入 s1	DATE
	输出 d	BOOL		输出 d	UDINT
DATE_TO_DINT	输入 s1	DATE	DATE_TO_UINT	输入 s1	DATE
	输出 d	DINT		输出 d	UINT
DATE_TO_DT	输入 s1	DATE	DATE_TO_USINT	输入 s1	DATE
	输出 d	DATE_AND_TIME		输出 d	USINT
DATE_TO_DWORD	输入 s1	DATE	DATE_TO_WORD	输入 s1	DATE
	输出 d	DWORD		输出 d	WORD
DATE_TO_INT	输入 s1	DATE	DATE_TO_LREAL	输入 s1	DATE
	输出 d	INT		输出 d	LREAL
DATE_TO_REAL	输入 s1	DATE	DATE_TO_LWORD	输入 s1	DATE
	输出 d	REAL		输出 d	LWORD
DATE_TO_SINT	输入 s1	DATE	DATE_TO_LINT	输入 s1	DATE
	输出 d	SINT		输出 d	LINT
DATE_TO_TIME	输入 s1	DATE	DATE_TO_LTIME	输入 s1	DATE
	输出 d	TIME		输出 d	LTIME

使用说明

- DATE 转 BOOL 时 DATE_TO_BOOL(D#1970-01-01)时输出为 FALSE；
- DATE 转 INT 时 DATE_TO_INT(D#1970-01-15)时输出为 29952；

▪ DATE 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当 DATE 为 TRUE 时，输出为 1970-01-01-0:0:1。

▪ DATE 转 TIME、DATE 转 TOD、DATE 转 LTIME 时，DATE 在底层是以秒进行计，TIME 和 TIME_OF_DAY 是以毫秒(ms)计时的，LTIME 是以纳秒(ns)进行计时的，所以转换需要作相应的单位转换。

注意事项

▪ 若是 DATE 是有其他类型转化过来，可能会出现内部有存值的情况。如当一个 INT 型数据 10 通过 INT_TO_DATE 转化成 DATE 这时的输出显示为 1970-01-01，但是其实是在内部进行了 10 秒保存，当这个值通过 DATE_TO_INT 时，显示的是 10 而不是 0，这个值通过 DATE_TO_BOOL 时，显示的值是 TRUE。

44、DINT_TO_<TYPE>/DINT 类型转换指令

DINT_TO_<TYPE>

功能

- 将 DINT 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DINT_TO_BYTE	输入 s1																				●	●					
	输出 d																										
DINT_TO_DATE	输入 s1																				●	●					
	输出 d																										
DINT_TO_BOOL	输入 s1																				●	●					
	输出 d		●	●	●	●																					
DINT_TO_DT	输入 s1																				●	●					
	输出 d																										
DINT_TO_DWORD	输入 s1																				●	●					
	输出 d																										
DINT_TO_INT	输入 s1																				●	●					
	输出 d															●	●	●	●	●							
DINT_TO_REAL	输入 s1																				●	●					
	输出 d																										
DINT_TO_SINT	输入 s1																				●	●					
	输出 d																										
DINT_TO_TIME	输入 s1																				●	●					
	输出 d																										
DINT_TO_TOD	输入 s1																				●	●					
	输出 d																										
DINT_TO_UDINT	输入 s1																				●	●					
	输出 d																										
DINT_TO_UINT	输入 s1																				●	●					
	输出 d																										
DINT_TO_USINT	输入 s1																				●	●					
	输出 d																										
DINT_TO_WORD	输入 s1																				●	●					
	输出 d																										
DINT_TO_LREAL	输入 s1																				●	●					
	输出 d																										
DINT_TO_LWORD	输入 s1																				●	●					
	输出 d																										

DINT_TO_LINT	输入 s1																				●	●					
	输出 d																										
DINT_TO_LTIME	输入 s1																				●	●					
	输出 d																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
DINT_TO_BYTE	输入 s1	DINT	DINT_TO_TOD	输入 s1	DINT
	输出 d	BYTE		输出 d	TIME_OF_DAY
DINT_TO_DATE	输入 s1	DINT	DINT_TO_UDINT	输入 s1	DINT
	输出 d	DATE		输出 d	UDINT
DINT_TO_BOOL	输入 s1	DINT	DINT_TO_UINT	输入 s1	DINT
	输出 d	BOOL		输出 d	UINT
DINT_TO_DT	输入 s1	DINT	DINT_TO_USINT	输入 s1	DINT
	输出 d	DATE_AND_TIME		输出 d	USINT
DINT_TO_DWORD	输入 s1	DINT	DINT_TO_WORD	输入 s1	DINT
	输出 d	DWORD		输出 d	WORD
DINT_TO_INT	输入 s1	DINT	DINT_TO_LREAL	输入 s1	DINT
	输出 d	INT		输出 d	LREAL
DINT_TO_REAL	输入 s1	DINT	DINT_TO_LWORD	输入 s1	DINT
	输出 d	REAL		输出 d	LWORD
DINT_TO_SINT	输入 s1	DINT	DINT_TO_LINT	输入 s1	DINT
	输出 d	SINT		输出 d	LINT
DINT_TO_TIME	输入 s1	DINT	DINT_TO_LTIME	输入 s1	DINT
	输出 d	TIME		输出 d	LTIME

使用说明

- DINT 转 BOOL 时，非 0 的输出为 TRUE；
- DINT 转 DATE 时，DATE 在底层是以秒进行计时。 DINT_TO_DATE 输入为 86400 时输出为的 1970-01-02；
- DINT 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当 DINT 为 61 时，输出为 1970-01-01-0:01:1。

45、DT_TO_<TYPE>/DT 类型转换指令

DT_TO_<TYPE>

功能

- 在内部，日期保存为一个 DWORD（自 1970 年 1 月 1 日以来的秒数）；
- 将 DATE_AND_TIME 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DT_TO_DINT	输入 s1																									
	输出 d																				●	●				
DT_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				
DT_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
DT_TO_BYTE	输入 s1	DATE_AND_TIME	DT_TO_UINT	输入 s1	DATE_AND_TIME
	输出 d	BYTE		输出 d	UINT
DT_TO_DATE	输入 s1	DATE_AND_TIME	DT_TO_USINT	输入 s1	DATE_AND_TIME
	输出 d	DATE		输出 d	USINT
DT_TO_DINT	输入 s1	DATE_AND_TIME	DT_TO_WORD	输入 s1	DATE_AND_TIME
	输出 d	DINT		输出 d	WORD
DT_TO_BOOL	输入 s1	DATE_AND_TIME	DT_TO_LREAL	输入 s1	DATE_AND_TIME
	输出 d	BOOL		输出 d	LREAL
DT_TO_DWORD	输入 s1	DATE_AND_TIME	DT_TO_USINT	输入 s1	DATE_AND_TIME
	输出 d	DWORD		输出 d	USINT
DT_TO_INT	输入 s1	DATE_AND_TIME	DT_TO_WORD	输入 s1	DATE_AND_TIME
	输出 d	INT		输出 d	WORD
DT_TO_REAL	输入 s1	DATE_AND_TIME	DT_TO_LREAL	输入 s1	DATE_AND_TIME
	输出 d	REAL		输出 d	LREAL
DT_TO_SINT	输入 s1	DATE_AND_TIME	DT_TO_LWORD	输入 s1	DATE_AND_TIME
	输出 d	SINT		输出 d	LWORD
DT_TO_TIME	输入 s1	DATE_AND_TIME	DT_TO_LINT	输入 s1	DATE_AND_TIME
	输出 d	TIME		输出 d	LINT
DT_TO_TOD	输入 s1	DATE_AND_TIME	DT_TO_LTIME	输入 s1	DATE_AND_TIME
	输出 d	TIME_OF_DAY		输出 d	LTIME
DT_TO_UDINT	输入 s1	DATE_AND_TIME			
	输出 d	UDINT			

福州富昌维控电子科技有限公司

使用说明

- DT 转 BOOL 时输入为 1970-01-01-0:0:0，输出的值为 FALSE，反之输出为 TRUE；
- DT 转 DATE 时，输入为 1970-01-01-1:1:1，输入为 1970-01-01，但是其内部有存值；
- DT 转 TOD、DT 转 TIME、DT 转 LTIME 时，DATE_AND_TIME 在底层是以秒进行计，TIME 和 TIME_OF_DAY 是以毫秒(ms)计时的，LTIME 是以纳秒(ns)进行计时的，所以转换需要作相应的单位转换。

46、DWORD_TO_<TYPE>/DWORD 类型转换指令

DWORD_TO_<TYPE>

功能

- 将 DWORD 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
DWORD_TO_DINT	输入 s1																									
	输出 d																				●	●				
DWORD_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				
DWORD_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
DWORD_TO_BYTE	输入 s1	DWORD	DWORD_TO_TOD	输入 s1	DWORD
	输出 d	BYTE		输出 d	TIME_OF_DAY
DWORD_TO_DATE	输入 s1	DWORD	DWORD_TO_UDINT	输入 s1	DWORD
	输出 d	DATE		输出 d	UDINT
DWORD_TO_DINT	输入 s1	DWORD	DWORD_TO_UINT	输入 s1	DWORD
	输出 d	DINT		输出 d	UINT
DWORD_TO_DT	输入 s1	DWORD	DWORD_TO_USINT	输入 s1	DWORD
	输出 d	DATE_AND_TIME		输出 d	USINT
DWORD_TO_BOOL	输入 s1	DWORD	DWORD_TO_WORD	输入 s1	DWORD
	输出 d	BOOL		输出 d	WORD
DWORD_TO_INT	输入 s1	DWORD	DWORD_TO_LREAL	输入 s1	DWORD
	输出 d	INT		输出 d	LREAL
DWORD_TO_REAL	输入 s1	DWORD	DWORD_TO_LWORD	输入 s1	DWORD
	输出 d	REAL		输出 d	LWORD
DWORD_TO_SINT	输入 s1	DWORD	DWORD_TO_LINT	输入 s1	DWORD
	输出 d	SINT		输出 d	LINT
DWORD_TO_TIME	输入 s1	DWORD	DWORD_TO_LTIME	输入 s1	DWORD
	输出 d	TIME		输出 d	LTIME

使用说明

- DWORD 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE；
- DWORD 转 DATE 时，DATE 在底层是以秒进行计时。输入为 86400 时输出 1970-01-02；
- DWORD 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当输入为 86401 时，输出为 1970-01-02-0:0:1。

47、INT_TO_<TYPE>/INT 类型转换指令

INT_TO_<TYPE>

功能

- 将 INT 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
INT_TO_BYTE	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_DATE	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_DINT	输入 s1															●	●	●	●	●							
	输出 d																				●	●					
INT_TO_DT	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_DWORD	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_BOOL	输入 s1															●	●	●	●	●							
	输出 d		●	●	●	●																					
INT_TO_REAL	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_SINT	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_TIME	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_TOD	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_UDINT	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_UINT	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_USINT	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_WORD	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_LREAL	输入 s1															●	●	●	●	●							
	输出 d																										
INT_TO_LWORD	输入 s1															●	●	●	●	●							
	输出 d																										

INT_TO_LINT	输入 s1																●	●	●	●	●						
	输出 d																										
INT_TO_LTIME	输入 s1																●	●	●	●	●						
	输出 d																										

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
INT_TO_BYTE	输入 s1	INT	INT_TO_TOD	输入 s1	INT
	输出 d	BYTE		输出 d	TIME_OF_DAY
INT_TO_DATE	输入 s1	INT	INT_TO_UDINT	输入 s1	INT
	输出 d	DATE		输出 d	UDINT
INT_TO_DINT	输入 s1	INT	INT_TO_UINT	输入 s1	INT
	输出 d	DINT		输出 d	UINT
INT_TO_DT	输入 s1	INT	INT_TO_USINT	输入 s1	INT
	输出 d	DATE_AND_TIME		输出 d	USINT
INT_TO_DWORD	输入 s1	INT	INT_TO_WORD	输入 s1	INT
	输出 d	DWORD		输出 d	WORD
INT_TO_BOOL	输入 s1	INT	INT_TO_LREAL	输入 s1	INT
	输出 d	BOOL		输出 d	LREAL
INT_TO_REAL	输入 s1	INT	INT_TO_LWORD	输入 s1	INT
	输出 d	REAL		输出 d	LWORD
INT_TO_SINT	输入 s1	INT	INT_TO_LINT	输入 s1	INT
	输出 d	SINT		输出 d	LINT
INT_TO_TIME	输入 s1	INT	INT_TO_LTIME	输入 s1	INT
	输出 d	TIME		输出 d	LTIME

使用说明

- NT 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE；
- INT 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当输入为 10 时，输出为 1970-01-01-0:0:10。

48、WORD_TO_<TYPE>/WORD 类型转换指令

WORD_TO_<TYPE>

功能

- 将 WORD 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
WORD_TO_DINT	输入 s1																										
	输出 d																				●	●					
WORD_TO_INT	输入 s1																										
	输出 d															●	●	●	●	●							
WORD_TO_BOOL	输入 s1																										
	输出 d		●	●	●	●																					

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
WORD_TO_BYTE	输入 s1	WORD	WORD_TO_TOD	输入 s1	WORD
	输出 d	BYTE		输出 d	TIME_OF_DAY
WORD_TO_DATE	输入 s1	WORD	WORD_TO_UDINT	输入 s1	WORD
	输出 d	DATE		输出 d	UDINT
WORD_TO_DINT	输入 s1	WORD	WORD_TO_UINT	输入 s1	WORD
	输出 d	DINT		输出 d	UINT
WORD_TO_DT	输入 s1	WORD	WORD_TO_USINT	输入 s1	WORD
	输出 d	DATE_AND_TIME		输出 d	USINT
WORD_TO_DWORD	输入 s1	WORD	WORD_TO_BOOL	输入 s1	WORD
	输出 d	DWORD		输出 d	BOOL
WORD_TO_INT	输入 s1	WORD	WORD_TO_LREAL	输入 s1	WORD
	输出 d	INT		输出 d	LREAL
WORD_TO_REAL	输入 s1	WORD	WORD_TO_LWORD	输入 s1	WORD
	输出 d	REAL		输出 d	LWORD
WORD_TO_SINT	输入 s1	WORD	WORD_TO_LINT	输入 s1	WORD
	输出 d	SINT		输出 d	LINT
WORD_TO_TIME	输入 s1	WORD	WORD_TO_LTIME	输入 s1	WORD
	输出 d	TIME		输出 d	LTIME

使用说明

- WORD 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE；
- WORD 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当 WORD 为 TRUE 时，输出为 1970-01-01-0:0:1。

49、REAL_TO_<TYPE>/REAL 类型转换指令

REAL_TO_<TYPE>

功能

- 将 REAL 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
REAL_TO_DINT	输入 s1																									
	输出 d																				●	●				
REAL_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						
REAL_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
REAL_TO_BYTE	输入 s1	REAL	REAL_TO_TOD	输入 s1	REAL
	输出 d	BYTE		输出 d	TIME_OF_DAY
REAL_TO_DATE	输入 s1	REAL	REAL_TO_UDINT	输入 s1	REAL
	输出 d	DATE		输出 d	UDINT
REAL_TO_DINT	输入 s1	REAL	REAL_TO_UINT	输入 s1	REAL
	输出 d	DINT		输出 d	UINT
REAL_TO_DT	输入 s1	REAL	REAL_TO_USINT	输入 s1	REAL
	输出 d	DATE_AND_TIME		输出 d	USINT
REAL_TO_DWORD	输入 s1	REAL	REAL_TO_WORD	输入 s1	REAL
	输出 d	DWORD		输出 d	WORD
REAL_TO_INT	输入 s1	REAL	REAL_TO_LREAL	输入 s1	REAL
	输出 d	INT		输出 d	LREAL
REAL_TO_BOOL	输入 s1	REAL	REAL_TO_LWORD	输入 s1	REAL
	输出 d	BOOL		输出 d	LWORD
REAL_TO_SINT	输入 s1	REAL	REAL_TO_LINT	输入 s1	REAL
	输出 d	SINT		输出 d	LINT
REAL_TO_TIME	输入 s1	REAL	REAL_TO_LTIME	输入 s1	REAL
	输出 d	TIME		输出 d	LTIME

使用说明

- REAL 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE；
- REAL 转 INT 时，当 REAL 为 1.5 时，输出为 2，当 REAL 为 1.4 时，输出为 1；

  当 REAL 为-1.5 时，输出为-2，当 REAL 为-1.4 时，输出为-1。

50、SINT_TO_<TYPE>/SINT 类型转换指令

SINT_TO_<TYPE>

功能

- 将 SINT 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SINT_TO_DINT	输入 s1																									
	输出 d																				●	●				
SINT_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						
SINT_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
SINT_TO_BYTE	输入 s1	SINT	REAL_TO_TOD	输入 s1	REAL
	输出 d	BYTE		输出 d	TIME_OF_DAY
SINT_TO_DATE	输入 s1	SINT	REAL_TO_UDINT	输入 s1	REAL
	输出 d	DATE		输出 d	UDINT
SINT_TO_DINT	输入 s1	SINT	REAL_TO_UINT	输入 s1	REAL
	输出 d	DINT		输出 d	UINT
SINT_TO_DT	输入 s1	SINT	REAL_TO_USINT	输入 s1	REAL
	输出 d	DATE_AND_TIME		输出 d	USINT
SINT_TO_DWORD	输入 s1	SINT	REAL_TO_WORD	输入 s1	REAL
	输出 d	DWORD		输出 d	WORD
SINT_TO_INT	输入 s1	SINT	REAL_TO_LREAL	输入 s1	REAL
	输出 d	INT		输出 d	LREAL
SINT_TO_REAL	输入 s1	SINT	REAL_TO_LWORD	输入 s1	REAL
	输出 d	REAL		输出 d	LWORD
SINT_TO_BOOL	输入 s1	SINT	REAL_TO_LINT	输入 s1	REAL
	输出 d	BOOL		输出 d	LINT
SINT_TO_TIME	输入 s1	SINT	REAL_TO_LTIME	输入 s1	REAL
	输出 d	TIME		输出 d	LTIME
SINT_TO_TOD	输入 s1	SINT			
	输出 d	TIME_OF_DAY			

使用说明

- SINT 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE；
- SINT 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当 SINT 为 61 时，输出为 1970-01-01-0:1:1。

51、TIME_TO_<TYPE>/TIME 类型转换指令

TIME_TO_<TYPE>

功能

- TIME 单位是毫秒；
- 将 TIME 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TIME_TO_DINT	输入 s1																									
	输出 d																				●	●				
TIME_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						
TIME_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
TIME_TO_BYTE	输入 s1	TIME	TIME_TO_UDINT	输入 s1	TIME
	输出 d	BYTE		输出 d	UDINT
TIME_TO_DATE	输入 s1	TIME	TIME_TO_UINT	输入 s1	TIME
	输出 d	DATE		输出 d	UINT
TIME_TO_DINT	输入 s1	TIME	TIME_TO_USINT	输入 s1	TIME
	输出 d	DINT		输出 d	USINT
TIME_TO_DT	输入 s1	TIME	TIME_TO_WORD	输入 s1	TIME
	输出 d	DATE_AND_TIME		输出 d	WORD
TIME_TO_DWORD	输入 s1	TIME	TIME_TO_LREAL	输入 s1	TIME
	输出 d	DWORD		输出 d	LREAL
TIME_TO_INT	输入 s1	TIME	TIME_TO_LWORD	输入 s1	TIME
	输出 d	INT		输出 d	LWORD
TIME_TO_REAL	输入 s1	TIME	TIME_TO_ LINT	输入 s1	TIME
	输出 d	REAL		输出 d	LINT
TIME_TO_SINT	输入 s1	TIME	TIME_TO_LTIME	输入 s1	TIME
	输出 d	SINT		输出 d	LTIME
TIME_TO_BOOL	输入 s1	TIME	TIME_TO_UDINT	输入 s1	TIME
	输出 d	BOOL		输出 d	UDINT
TIME_TO_TOD	输入 s1	TIME			
	输出 d	TIME_OF_DAY			

使用说明

- TIME 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE。
- DT 转 TOD、DT 转 TIME、DT 转 LTIME 时，DATE_AND_TIME 在底层是以秒进行计，TIME 和 TIME_OF_DAY 是以毫秒(ms)计时的，LTIME 是以纳秒(ns)进行计时的，所以转换需要作相应的单位转换。

福州富昌维控电子科技有限公司

52、TOD_TO_<TYPE>/TOD 类型转换指令

TOD_TO_<TYPE>

功能

- 将 TIME_OF_DAY 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TOD_TO_DINT	输入 s1																									
	输出 d																				●	●				
TOD_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						
TOD_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
TOD_TO_BYTE	输入 s1	TIME_OF_DAY	TOD_TO_UDINT	输入 s1	TIME_OF_DAY
	输出 d	BYTE		输出 d	UDINT
TOD_TO_DATE	输入 s1	TIME_OF_DAY	TOD_TO_UINT	输入 s1	TIME_OF_DAY
	输出 d	DATE		输出 d	UINT
TOD_TO_DINT	输入 s1	TIME_OF_DAY	TOD_TO_USINT	输入 s1	TIME_OF_DAY
	输出 d	DINT		输出 d	USINT
TOD_TO_DT	输入 s1	TIME_OF_DAY	TOD_TO_WORD	输入 s1	TIME_OF_DAY
	输出 d	DATE_AND_TIME		输出 d	WORD
TOD_TO_DWORD	输入 s1	TIME_OF_DAY	TOD_TO_LREAL	输入 s1	TIME_OF_DAY
	输出 d	DWORD		输出 d	LREAL
TOD_TO_INT	输入 s1	TIME_OF_DAY	TOD_TO_ LWORD	输入 s1	TIME_OF_DAY
	输出 d	INT		输出 d	LWORD
TOD_TO_REAL	输入 s1	TIME_OF_DAY	TOD_TO_ LINT	输入 s1	TIME_OF_DAY
	输出 d	REAL		输出 d	LINT
TOD_TO_SINT	输入 s1	TIME_OF_DAY	TOD_TO_LTIME	输入 s1	TIME_OF_DAY
	输出 d	SINT		输出 d	LTIME
TOD_TO_TIME	输入 s1	TIME_OF_DAY	TOD_TO_UDINT	输入 s1	TIME_OF_DAY
	输出 d	TIME		输出 d	UDINT
TOD_TO_BOOL	输入 s1	TIME_OF_DAY	TOD_TO_UINT	输入 s1	TIME_OF_DAY
	输出 d	BOOL		输出 d	UINT

使用说明

- TOD 转 BOOL 时，输入为 00:00:00 时，输出为 FALSE，反之为 TRUE；
- TOD 转 INT 时，输入为 01:05:05 时，输出为 3905；
- TOD 转 DT 时，DATE_AND_TIME 在底层是以秒进行计时。当输入为 01:05:05 时，输出为 1970-01-01-01:05:05；
- TOD 转 DATE、TIME 转 DT、TOD 转 LTIME 时，TOD 是以毫秒(ms)计时的，LTIME 是以纳秒(ns)进行计时的，DATE_AND_TIME、DATE 在底层是以秒进行计，所以转换需要作相应的单位转换。

53、UDINT_TO_<TYPE>/UDINT 类型转换指令

UDINT_TO_<TYPE>

功能

- 将 UDINT 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
UDINT_TO_DINT	输入 s1																									
	输出 d																				●	●				
UDINT_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						
UDINT_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
TOD_TO_BYTE	输入 s1	TIME_OF_DAY	TOD_TO_UDINT	输入 s1	TIME_OF_DAY
	输出 d	BYTE		输出 d	UDINT
TOD_TO_DATE	输入 s1	TIME_OF_DAY	TOD_TO_UINT	输入 s1	TIME_OF_DAY
	输出 d	DATE		输出 d	UINT
TOD_TO_DINT	输入 s1	TIME_OF_DAY	TOD_TO_USINT	输入 s1	TIME_OF_DAY
	输出 d	DINT		输出 d	USINT
TOD_TO_DT	输入 s1	TIME_OF_DAY	TOD_TO_WORD	输入 s1	TIME_OF_DAY
	输出 d	DATE_AND_TIME		输出 d	WORD
TOD_TO_DWORD	输入 s1	TIME_OF_DAY	TOD_TO_LREAL	输入 s1	TIME_OF_DAY
	输出 d	DWORD		输出 d	LREAL
TOD_TO_INT	输入 s1	TIME_OF_DAY	TOD_TO_ LWORD	输入 s1	TIME_OF_DAY
	输出 d	INT		输出 d	LWORD
TOD_TO_REAL	输入 s1	TIME_OF_DAY	TOD_TO_ LINT	输入 s1	TIME_OF_DAY
	输出 d	REAL		输出 d	LINT
TOD_TO_SINT	输入 s1	TIME_OF_DAY	TOD_TO_LTIME	输入 s1	TIME_OF_DAY
	输出 d	SINT		输出 d	LTIME
TOD_TO_TIME	输入 s1	TIME_OF_DAY	TOD_TO_UDINT	输入 s1	TIME_OF_DAY
	输出 d	TIME		输出 d	UDINT
TOD_TO_BOOL	输入 s1	TIME_OF_DAY	TOD_TO_UINT	输入 s1	TIME_OF_DAY
	输出 d	BOOL		输出 d	UINT

使用说明

- UDINT 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE。

54、UINT_TO_<TYPE>/UINT 类型转换指令

UINT_TO_<TYPE>

功能

- 将 UINT 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
UINT_TO_DINT	输入 s1																										
	输出 d																				●	●					
UINT_TO_INT	输入 s1																										
	输出 d															●	●	●	●	●							
UINT_TO_BOOL	输入 s1																										
	输出 d		●	●	●	●																					

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
UINT_TO_BYTE	输入 s1	UINT	UINT_TO_UDINT	输入 s1	UINT
	输出 d	BYTE		输出 d	UDINT
UINT_TO_DATE	输入 s1	UINT	UINT_TO_BOOL	输入 s1	UINT
	输出 d	DATE		输出 d	BOOL
UINT_TO_DINT	输入 s1	UINT	UINT_TO_USINT	输入 s1	UINT
	输出 d	DINT		输出 d	USINT
UINT_TO_DT	输入 s1	UINT	UINT_TO_WORD	输入 s1	UINT
	输出 d	DATE_AND_TIME		输出 d	WORD
UINT_TO_DWORD	输入 s1	UINT	UINT_TO_LREAL	输入 s1	UINT
	输出 d	DWORD		输出 d	LREAL
UINT_TO_INT	输入 s1	UINT	UINT_TO_LWORD	输入 s1	UINT
	输出 d	INT		输出 d	LWORD
UINT_TO_REAL	输入 s1	UINT	UINT_TO_ LINT	输入 s1	UINT
	输出 d	REAL		输出 d	LINT
UINT_TO_SINT	输入 s1	UINT	UINT_TO_LTIME	输入 s1	UINT
	输出 d	SINT		输出 d	LTIME
UINT_TO_TIME	输入 s1	UINT	UINT_TO_UDINT	输入 s1	UINT
	输出 d	TIME		输出 d	UDINT
UINT_TO_TOD	输入 s1	UINT			
	输出 d	TIME_OF_DAY			

使用说明

- UINT 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE。

55、USINT_TO_<TYPE>/USINT 类型转换指令

USINT_TO_<TYPE>

功能

- 将 USINT 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
USINT_TO_DINT	输入 s1																										
	输出 d																				●	●					
USINT_TO_INT	输入 s1																										
	输出 d															●	●	●	●	●							
USINT_TO_BOOL	输入 s1																										
	输出 d		●	●	●	●																					

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
USINT_TO_BYTE	输入 s1	USINT	USINT_TO_TOD	输入 s1	USINT
	输出 d	BYTE		输出 d	TIME_OF_DAY
USINT_TO_DATE	输入 s1	USINT	USINT_TO_UDINT	输入 s1	USINT
	输出 d	DATE		输出 d	UDINT
USINT_TO_DINT	输入 s1	USINT	USINT_TO_UINT	输入 s1	USINT
	输出 d	DINT		输出 d	UINT
USINT_TO_DT	输入 s1	USINT	USINT_TO_BOOL	输入 s1	USINT
	输出 d	DATE_AND_TIME		输出 d	BOOL
USINT_TO_DWORD	输入 s1	USINT	USINT_TO_WORD	输入 s1	USINT
	输出 d	DWORD		输出 d	WORD
USINT_TO_INT	输入 s1	USINT	USINT_TO_LREAL	输入 s1	USINT
	输出 d	INT		输出 d	LREAL
USINT_TO_REAL	输入 s1	USINT	USINT_TO_LWORD	输入 s1	USINT
	输出 d	REAL		输出 d	LWORD
USINT_TO_SINT	输入 s1	USINT	USINT_TO_LINT	输入 s1	USINT
	输出 d	SINT		输出 d	LINT
USINT_TO_TIME	输入 s1	USINT	USINT_TO_LTIME	输入 s1	USINT
	输出 d	TIME		输出 d	LTIME

使用说明

- USINT 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE。

56、LREAL_TO_<TYPE>/LREAL 类型转换指令

LREAL_TO_<TYPE>

功能

- 将 LREAL 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LREAL_TO_DINT	输入 s1																									
	输出 d																				●	●				
LREAL_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						
LREAL_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
LREAL_TO_BYTE	输入 s1	LREAL	LREAL_TO_TOD	输入 s1	LREAL
	输出 d	BYTE		输出 d	TIME_OF_DAY
LREAL_TO_DATE	输入 s1	LREAL	LREAL_TO_UDINT	输入 s1	LREAL
	输出 d	DATE		输出 d	UDINT
LREAL_TO_DINT	输入 s1	LREAL	LREAL_TO_UINT	输入 s1	LREAL
	输出 d	DINT		输出 d	UINT
LREAL_TO_DT	输入 s1	LREAL	LREAL_TO_USINT	输入 s1	LREAL
	输出 d	DATE_AND_TIME		输出 d	USINT
LREAL_TO_DWORD	输入 s1	LREAL	LREAL_TO_WORD	输入 s1	LREAL
	输出 d	DWORD		输出 d	WORD
LREAL_TO_INT	输入 s1	LREAL	LREAL_TO_REAL	输入 s1	LREAL
	输出 d	INT		输出 d	REAL
LREAL_TO_BOOL	输入 s1	LREAL	LREAL_TO_LWORD	输入 s1	LREAL
	输出 d	BOOL		输出 d	LWORD
LREAL_TO_SINT	输入 s1	LREAL	LREAL_TO_LINT	输入 s1	LREAL
	输出 d	SINT		输出 d	LINT
LREAL_TO_TIME	输入 s1	LREAL	LREAL_TO_LTIME	输入 s1	LREAL
	输出 d	TIME		输出 d	LTIME

使用说明

- LREAL 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE；
- LREAL 转 INT 时，当 LREAL 为 1.5 时，输出为 2,当 LREAL 为 1.4 时，输出为 1；

  当 LREAL 为-1.5 时，输出为-2,当 LREAL 为-1.4 时，输出为-1。

57、LWORD_TO_<TYPE>/LWORD 类型转换指令

LWORD_TO_<TYPE>

功能

- 将 LWORD 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LWORD_TO_DINT	输入 s1																									
	输出 d																				●	●				
LWORD_TO_INT	输入 s1																									
	输出 d															●	●	●	●	●						
LWORD_TO_BOOL	输入 s1																									
	输出 d		●	●	●	●																				

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
LWORD_TO_BYTE	输入 s1	LWORD	LWORD_TO_TOD	输入 s1	LWORD
	输出 d	BYTE		输出 d	TIME_OF_DAY
LWORD_TO_DATE	输入 s1	LWORD	LWORD_TO_UDINT	输入 s1	LWORD
	输出 d	DATE		输出 d	UDINT
LWORD_TO_DINT	输入 s1	LWORD	LWORD_TO_UINT	输入 s1	LWORD
	输出 d	DINT		输出 d	UINT
LWORD_TO_DT	输入 s1	LWORD	LWORD_TO_USINT	输入 s1	LWORD
	输出 d	DATE_AND_TIME		输出 d	USINT
LWORD_TO_DWORD	输入 s1	LWORD	LWORD_TO_WORD	输入 s1	LWORD
	输出 d	DWORD		输出 d	WORD
LWORD_TO_INT	输入 s1	LWORD	LWORD_TO_REAL	输入 s1	LWORD
	输出 d	INT		输出 d	REAL
LWORD_TO_BOOL	输入 s1	LWORD	LWORD_TO_LREAL	输入 s1	LWORD
	输出 d	BOOL		输出 d	LREAL
LWORD_TO_SINT	输入 s1	LWORD	LWORD_TO_LINT	输入 s1	LWORD
	输出 d	SINT		输出 d	LINT
LWORD_TO_TIME	输入 s1	LWORD	LWORD_TO_LTIME	输入 s1	LWORD
	输出 d	TIME		输出 d	LTIME

使用说明

- LWORD 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE。

58、LINT_TO_<TYPE>/LINT 类型转换指令

LINT_TO_<TYPE>

功能

- 将 LINT 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LINT_TO_DINT	输入 s1																										
	输出 d																				●	●					
LINT_TO_INT	输入 s1																										
	输出 d															●	●	●	●	●							
LINT_TO_BOOL	输入 s1																										
	输出 d		●	●	●	●																					

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
LINT_TO_BYTE	输入 s1	LINT	LINT_TO_TOD	输入 s1	LINT
	输出 d	BYTE		输出 d	TIME_OF_DAY
LINT_TO_DATE	输入 s1	LINT	LINT_TO_UDINT	输入 s1	LINT
	输出 d	DATE		输出 d	UDINT
LINT_TO_DINT	输入 s1	LINT	LINT_TO_UINT	输入 s1	LINT
	输出 d	DINT		输出 d	UINT
LINT_TO_DT	输入 s1	LINT	LINT_TO_USINT	输入 s1	LINT
	输出 d	DATE_AND_TIME		输出 d	USINT
LINT_TO_DWORD	输入 s1	LINT	LINT_TO_WORD	输入 s1	LINT
	输出 d	DWORD		输出 d	WORD
LINT_TO_INT	输入 s1	LINT	LINT_TO_REAL	输入 s1	LINT
	输出 d	INT		输出 d	REAL
LINT_TO_BOOL	输入 s1	LINT	LINT_TO_LREAL	输入 s1	LINT
	输出 d	BOOL		输出 d	LREAL
LINT_TO_SINT	输入 s1	LINT	LINT_TO_LWORD	输入 s1	LINT
	输出 d	SINT		输出 d	LWORD
LINT_TO_TIME	输入 s1	LINT	LINT_TO_LTIME	输入 s1	LINT
	输出 d	TIME		输出 d	LTIME

使用说明

- LINT 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE。

59、LTIME_TO_<TYPE>/LTIME 类型转换指令

LTIME_TO_<TYPE>

功能

- 将 LTIME 数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																								偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
LTIME_TO_DINT	输入 s1																										
	输出 d																				●	●					
LTIME_TO_INT	输入 s1																										
	输出 d															●	●	●	●	●							
LTIME_TO_BOOL	输入 s1																										
	输出 d		●	●	●	●																					

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
LTIME_TO_BYTE	输入 s1	LTIME	LTIME_TO_TOD	输入 s1	LTIME
	输出 d	BYTE		输出 d	TIME_OF_DAY
LTIME_TO_DATE	输入 s1	LTIME	LTIME_TO_UDINT	输入 s1	LTIME
	输出 d	DATE		输出 d	UDINT
LTIME_TO_DINT	输入 s1	LTIME	LTIME_TO_UINT	输入 s1	LTIME
	输出 d	DINT		输出 d	UINT
LTIME_TO_DT	输入 s1	LTIME	LTIME_TO_USINT	输入 s1	LTIME
	输出 d	DATE_AND_TIME		输出 d	USINT
LTIME_TO_DWORD	输入 s1	LTIME	LTIME_TO_WORD	输入 s1	LTIME
	输出 d	DWORD		输出 d	WORD
LTIME_TO_INT	输入 s1	LTIME	LTIME_TO_REAL	输入 s1	LTIME
	输出 d	INT		输出 d	REAL
LTIME_TO_BOOL	输入 s1	LTIME	LTIME_TO_LREAL	输入 s1	LTIME
	输出 d	BOOL		输出 d	LREAL
LTIME_TO_SINT	输入 s1	LTIME	LTIME_TO_LWORD	输入 s1	LTIME
	输出 d	SINT		输出 d	LWORD
LTIME_TO_TIME	输入 s1	LTIME	LTIME_TO_LINT	输入 s1	LTIME
	输出 d	TIME		输出 d	LINT

使用说明

- LTIME 转 BOOL 时若输入等于 0，输出的值为 FALSE，反之输出的值为 TRUE；
- LTIME 转 DATE、LTIME 转 DT、LTIME 转 TOD、LTIME 转 TIME 时，LTIME 是以纳秒(ns)进行计时的，TOD 是以毫秒(ms)计时的 DATE_AND_TIME、DATE 在底层是以秒进行计，所以转换需要作相应的单位转换。

60、TO_<TYPE>类型转换指令

TO_<TYPE>

功能

- 将某个数据类型转化成其他数据类型。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TO_BYTE	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_DATE	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_DINT	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																				●	●					
TO_DT	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_DWORD	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_INT	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d															●	●	●	●	●							
TO_BOOL	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d		●	●	●	●																					
TO_SINT	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_TIME	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_TOD	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_UDINT	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_UINT	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_USINT	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_WORD	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_LREAL	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_REAL	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										

TO_LWORD	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_LINT	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										
TO_LTIME	输入 sl	●	●	●	●	●										●	●	●	●	●	●	●					
	输出 d																										

- 其余指令的不支持软元件使用。

软元件使用注意事项

- LC、HSC 为有符号 32 位数据类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- ANY：BOOL，BYTE，DATE，DATE_AND_TIME，SINT，TIME，TIME_OF_DAY，UDINT，UINT，USINT，WORD，DWORD，INT，DINT，REAL，LREAL，LWORD，LINT，LTIME。

使用的变量类型

指令	参数	变量类型	指令	参数	变量类型
TO_BYTE	输入 s1	ANY	TO_UDINT	输入 s1	ANY
	输出 d	BYTE		输出 d	UDINT
TO_DATE	输入 s1	ANY	TO_UINT	输入 s1	ANY
	输出 d	DATE		输出 d	UINT
TO_DINT	输入 s1	ANY	TO_USINT	输入 s1	ANY
	输出 d	DINT		输出 d	USINT
TO_DT	输入 s1	ANY	TO_WORD	输入 s1	ANY
	输出 d	DATE_AND_TIME		输出 d	WORD
TO_DWORD	输入 s1	ANY	TO_REAL	输入 s1	ANY
	输出 d	DWORD		输出 d	REAL
TO_INT	输入 s1	ANY	TO_LREAL	输入 s1	ANY
	输出 d	INT		输出 d	LREAL
TO_BOOL	输入 s1	ANY	TO_LWORD	输入 s1	ANY
	输出 d	BOOL		输出 d	LWORD
TO_SINT	输入 s1	ANY	TO_LINT	输入 s1	ANY
	输出 d	SINT		输出 d	LINT
TO_TIME	输入 s1	ANY	TO_LTIME	输入 s1	ANY
	输出 d	TIME		输出 d	LTIME
TO_TOD	输入 s1	ANY			
	输出 d	TIME_OF_DAY			

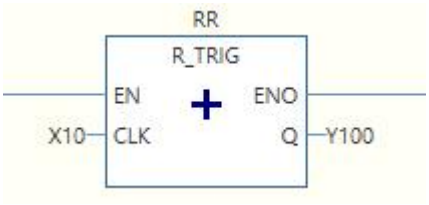
使用说明

- 根据输入的变量类型，内部自动转化成相应的数据类型转化，详细转化功能同上。

14.2 内置 FB 功能块

1、 R_TRIG/上升沿检测

R_TRIG



上升沿检测

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
R_TRIG	CLK	●	●	●	●	●																				
	Q		●	●	●	●																				

软元件使用注意事项

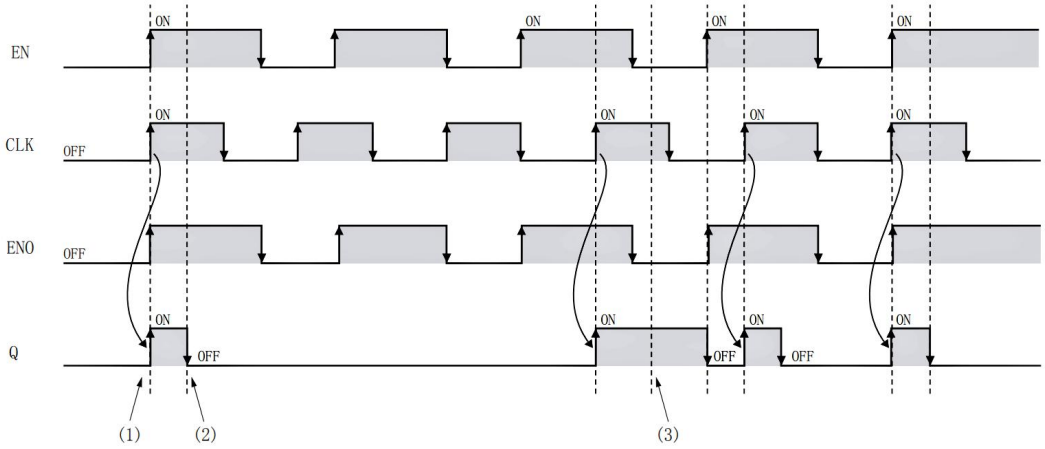
- X、Y、M、S、SM 为 BOOL 类型；
- X 不能作为输出。

使用的变量类型

指令	参数	变量类型
R_TRIG	CLK	BOOL
	Q	BOOL

功能

- (CLK)变为 ON 时，仅 1 个扫描将(Q)置为 ON。
- 时序图

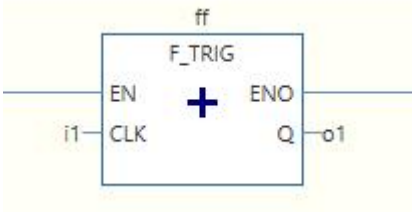


(1):通过 EN=ON 且(CLK)的上升沿将(Q)置为 ON；

(2):在下一次的扫描中，将(Q)置为 OFF；

(3): EN=OFF 的情况下，(Q)保持上次的扫描的输出。

2、 F_TRIG/下降沿检测



F_TRIG 下降沿检测

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
F_TRIG	CLK	●	●	●	●	●																				
	Q		●	●	●	●																				

软元件使用注意事项

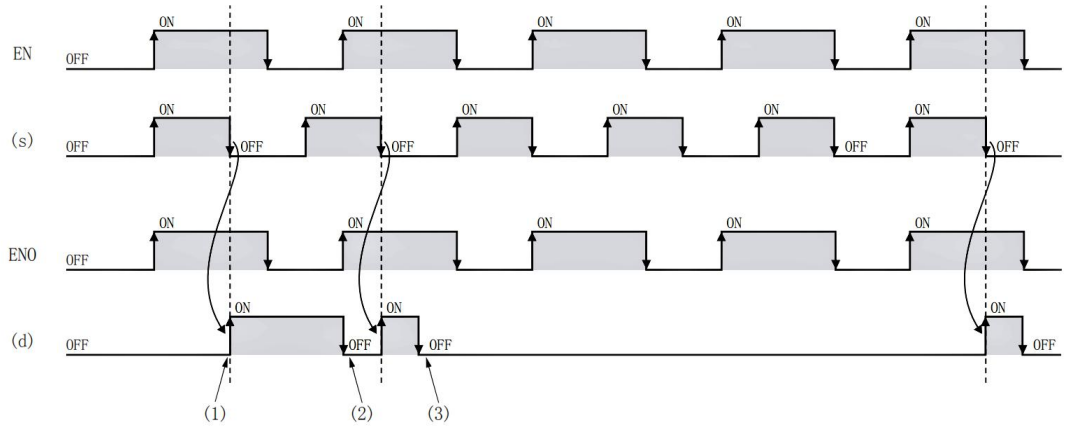
- X、Y、M、S、SM 为 BOOL 类型；
- X 不能作为输出。

使用的变量类型

指令	参数	变量类型
F_TRIG	CLK	BOOL
	Q	BOOL

功能

- (CLK)变为 OFF 时，仅 1 个扫描将(Q)置为 ON。
- 时序图



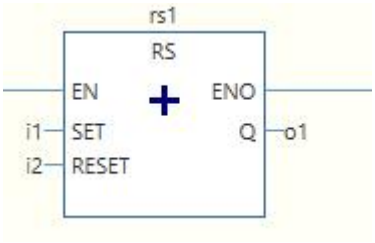
(1):在 EN=ON 且(CLK) 的下降沿将(Q)置为 ON;

(2):在下一次的扫描中，将(Q)置为 OFF;

(3): EN=OFF 的情况下，(Q)保持上次的扫描的输出。

3、 RS/双稳态功能块(复位优先)

RS



复位优先的双稳态功能块。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RS	SET	●	●	●	●	●																					
	RESET	●	●	●	●	●																					
	Q		●	●	●	●																					

软元件使用注意事项

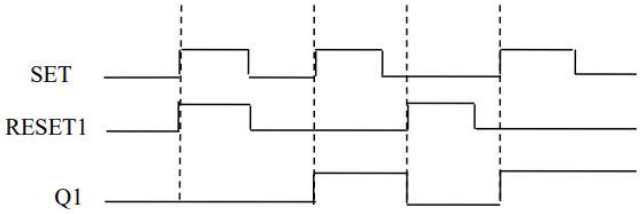
- X、Y、M、S、SM 为 BOOL 类型；
- X 不能作为输出。

使用的变量类型

指令	参数	变量类型
RS	SET	BOOL
	RESET	BOOL
	Q	BOOL

功能

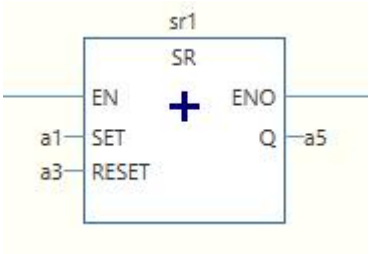
- 当 RESET 为“TRUE”时，不管 SET 为何值，Q 的输出都是“FALSE”；当 RESET 为“FALSE”，并且 SET 为“TRUE”时，Q 输出“TRUE”；当 RESET 和 SET 都为“FALSE”时 Q 保持原状态。



SET	RESET1	Q1 输出
0	0	保持原状态
1	0	1
0	1	0
1	1	0

4、 SR/双稳态功能块(置位优先)

SR



置位优先的双稳态功能块。

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
SR	SET	●	●	●	●	●																				
	RESET	●	●	●	●	●																				
	Q		●	●	●	●																				

软元件使用注意事项

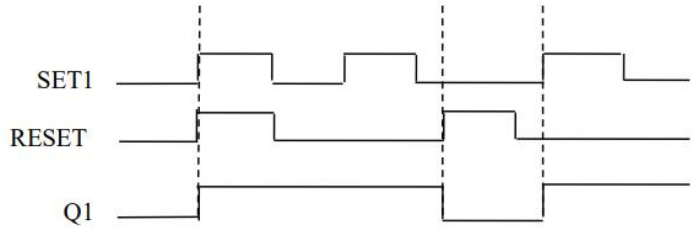
- X、Y、M、S、SM 为 BOOL 类型；
- X 不能作为输出。

使用的变量类型

指令	参数	变量类型
SR	SET	BOOL
	RESET	BOOL
	Q	BOOL

功能

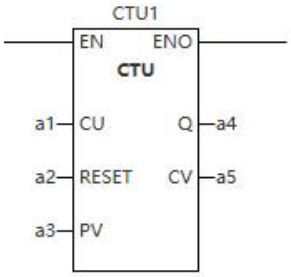
- 当 SET 为“TRUE”时，不管 RESET 为何值，Q 的输出都是“TRUE”；当 SET 为“FALSE”，并且 RESET 为“TRUE”时，Q 输出“FALSE”；当 RESET 和 SET 都为“FALSE”时 Q 保持原状态。



SET1	RESET	Q1 输出
0	0	保持原状态
1	0	1
0	1	0
1	1	1

5、 CTU/加计数器

CTU



加计数器

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
CTU	CU	●	●	●	●	●																					
	RESET	●	●	●	●	●																					
	PV															●	●	●	●	●							
	Q		●	●	●	●																					
	CV															●	●	●	●	●							

软元件使用注意事项

- X、Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型；
- X 不能作为输出使用。

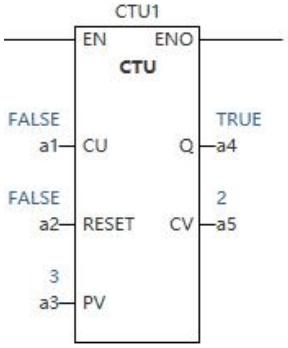
使用的变量类型

指令	参数	变量类型
CTU	CU	BOOL
	RESET	BOOL
	PV	WORD
	Q	BOOL
	CV	WORD

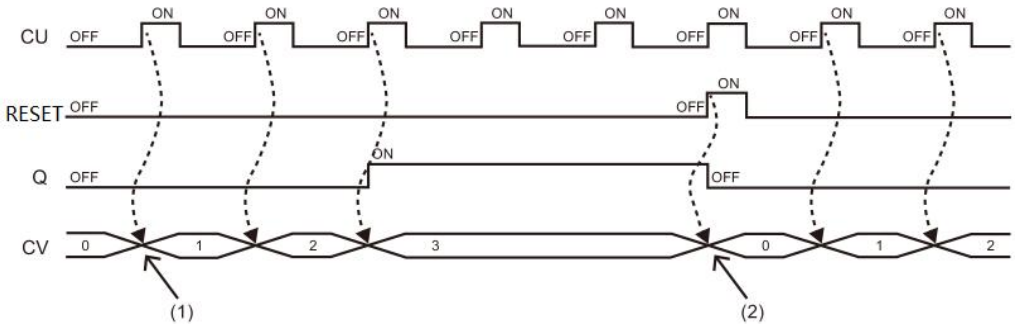
功能

- 如果(CU)变为 OFF→ON，对(CV)进行加法计数(+1)；
- 如果(CV)大于等于计数器的(PV)，则(Q)变为 ON，加法计数继续直到加到最大值 32767 时停止计数；
- (PV)设置计数器的目标值。如果将(RESET)置为 ON，(CV)被设置为 0。

例：PV 设置为 3



时序图：



(1)CU 上升沿，CV 进行增计数；

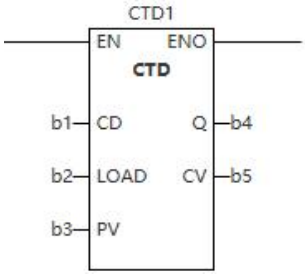
(2)RESET 置 ON，CV 复位。

注意事项

- 计数器范围是 0~32767；
- 若是 PV 的值设置的范围不在计数器范围内，Q 的输出将一直为 ON。

6、 CTD/减计数器

CTD



减计数器

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																						偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
CTD	CD	●	●	●	●	●																					
	LOAD	●	●	●	●	●																					
	PV															●	●	●	●	●							
	Q		●	●	●	●																					
	CV															●	●	●	●	●							

软元件使用注意事项

- X、Y、M、S、SM 为 BOOL 类型；
- T、C、D、R、SD 为有符号 16 位数据类型；
- LC、HSC 为有符号 32 位数据类型；
- X 不能作为输出使用。

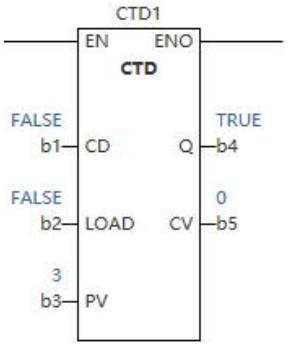
使用的变量类型

指令	参数	变量类型
CTD	CD	BOOL
	LOAD	BOOL
	PV	WORD
	Q	BOOL
	CV	WORD

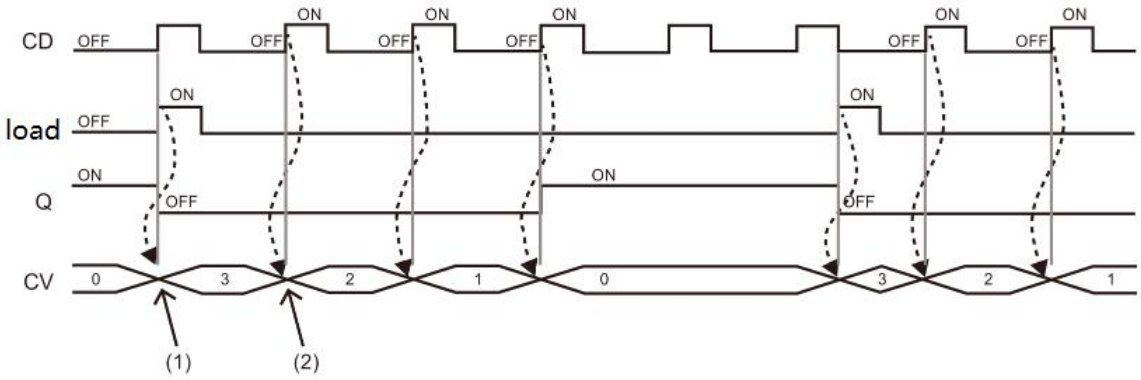
功能

- 如果(CD)变为 OFF→ON，对(CV)进行减法计数(-1)；
- 如果(CV)等于 0 时，则(Q)变为 ON，停止计数；
- (PV)设置计数器的初始值。如果将(LOAD)置为 ON，(CV)被设置(PV)的值。

例：PV 设置为 3



时序图：



(1)LOAD 置 ON，将 PV 的值赋给 CV；

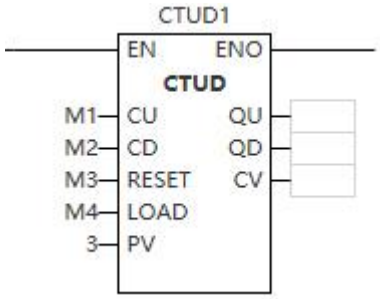
(2)CD 上升沿，CV 进行减计数。

注意事项

- 计数器范围是 0~32767；
- 若是 CV 的值设置的范围不在计数器范围内，Q 的输出将一直为 ON；
- 若是 CV 的值设置的范围不在计数器范围内，计数无效。

7、 CTUD/加、减计数器

CTUD



加、减双向计数器

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
CTUD	CU	●	●	●	●	●																				
	CD	●	●	●	●	●																				
	RESET	●	●	●	●	●																				
	LOAD	●	●	●	●	●																				
	PV															●	●	●	●	●						
	QU		●	●	●	●																				
	QD		●	●	●	●																				
	CV															●	●	●	●	●						

软元件使用注意事项

- T、C、D、R、SD 为有符号 16 位数据类型；
- X、Y、M、S、SM 为 BOOL 类型；
- LC、HSC 为有符号 32 位数据类型；
- X 不能作为输出使用。

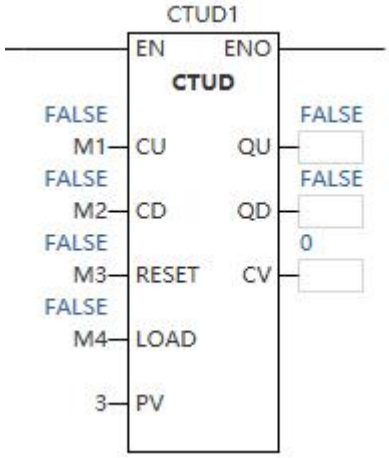
使用的变量类型

指令	参数	变量类型
CTUD	CU	BOOL
	CD	BOOL
	RESET	BOOL
	LOAD	BOOL
	PV	WORD
	QU	BOOL
	QD	BOOL
	CV	WORD

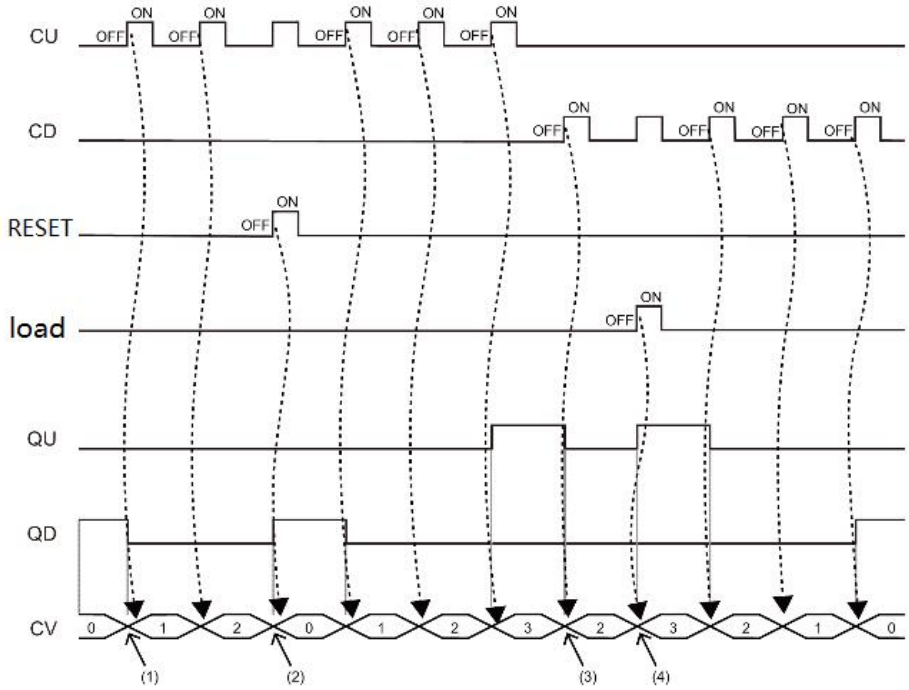
功能

- 如果(CU)变为 OFF→ON，对(CV)进行加法计数(+1)；
- 如果(CV)大于等于计数器的(PV)，则(QU)变为 ON，加法计数继续直到加到最大值 32767 时停止加计数；
- (PV)设置计数器的初始值。如果将(RESET)置为 ON，(CV)被设置为 0；
- 如果(CD)变为 OFF→ON，对(CV)进行减法计数(-1)；
- 如果(CV)等于 0 时，则(QD)变为 ON，停止减计数；
- (PV)设置计数器的目标值。如果将(LOAD)置为 ON，(CV)被设置(PV)的值。

例：PV 设置为 3



时序图：



(1) CU 上升沿，CV 进行增计数；

(2)RESET 置 ON，CV 复位；

(3)CD 上升沿，CV 进行减计数；

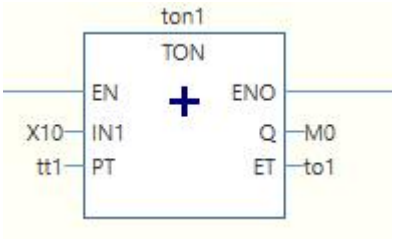
(4)LOAD 置 ON，将 PV 的值赋给 CV。

注意事项

- 计数器范围是 0~32767；
- 若是 PV 的值设置的范围不在计数器范围内，QU 的输出将一直为 ON；
- 若是 CV 的值设置的范围不在计数器范围内，加减计数无效,QD 的输出为 ON。

8、 TON/ON 延迟定时器

TON



ON 延迟定时器

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TON	IN1	●	●	●	●	●																				
	PT																									
	Q		●	●	●	●																				
	ET																									

软元件使用注意事项

- X、Y、M、S、SM 为 BOOL 类型；
- X 不能作为输出。

使用的变量类型

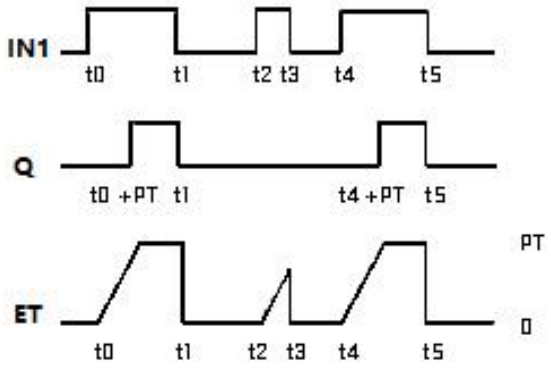
指令	参数	变量类型
TON	IN1	BOOL
	PT	TIME
	Q	BOOL
	ET	TIME

功能

- IN1 从“FALSE”变为“TRUE”，定时器启动，当达到 PT 指定时间并且期间 IN1 一直处于“TRUE”状态，则输出 Q 置为“TRUE”。若在期间 IN1 的状态变为“FALSE”则定时器复位，等待下一个 IN1 上升沿信号，定时器重启；

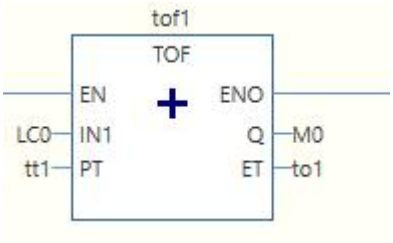
- ET 为定时器启动后经过的时间，延迟从 0 开始到 PT 设置的时间结束，然后保持 ET 直到 IN1 变为“FALSE”状态，ET 将复位为 0。

时序图



9、 TOF/OFF 延迟定时器

TOF



OFF 延迟定时器

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TOF	IN1	●	●	●	●	●																				
	PT																									
	Q		●	●	●	●																				
	ET																									

软元件使用注意事项

- X、Y、M、S、SM 为 BOOL 类型；
- X 不能作为输出。

使用的变量类型

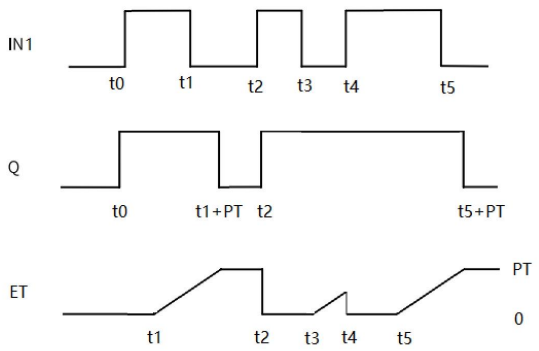
指令	参数	变量类型
TOF	IN1	BOOL
	PT	TIME
	Q	BOOL
	ET	TIME

功能

- IN1 从“FALSE”变为“TRUE”，定时器 Q 的输出信号为“TRUE”，定时器的启动输入端变为“FALSE”时，定时器启动，只要当定时器在运行，其输出 Q 一直为“TRUE”，当达到定时时间时输出端 Q 复位，在达到定时器时间之前，如果定时器的输入 IN1 变为“TRUE”状态，定时器 ET 复位，输出 Q 保持“TRUE”状态；

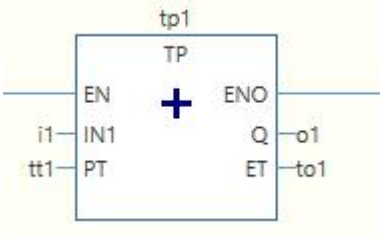
- ET 为定时器启动后经过的时间，延迟从 0 开始到 PT 设置的时间结束，然后保持 ET 直到 IN1 变为“TRUE”状态，ET 将复位为 0。如果在达到 PT 设定的时间之前，IN1 变为“TRUE”，输出 ET 将立即复位。

时序图



10、 TP/脉冲定时器

TP



脉冲定时器

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
TP	IN1	●	●	●	●	●																				
	PT																									
	Q		●	●	●	●																				
	ET																									

软元件使用注意事项

- X、Y、M、S、SM 为 BOOL 类型；
- X 不能作为输出。

使用的变量类型

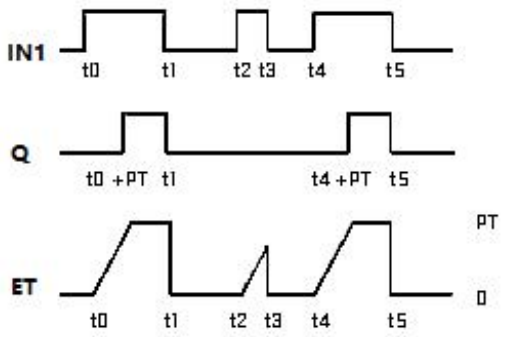
指令	参数	变量类型
TP	IN1	BOOL
	PT	TIME
	Q	BOOL
	ET	TIME

功能

- IN1 从“FALSE”变为“TRUE”时，定时器则启动，无论定时器输入端 IN1 如何变化，定时器的实际运行时间都是客户定义的 PT，在定时器运行时，去输出端 Q 的输出信号为“TRUE”；

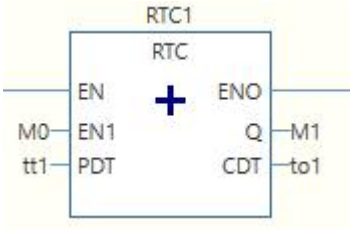
- 输出端 ET 为输出端 Q 提供定时时间。定时时间从 0 开始，到 PT 结束，当 PT 到达时 ET 会保持定时时间，直到 IN1 变为“FALSE”。

时序图



11、 RTC/万年历

RTC



万年历

使用的软元件（只能使用单一软元件，不支持软元件拼接以及偏移）

指令	参数	软元件																							偏移修饰	脉冲型扩展
		X	Y	M	S	SM	T(bit)	C(bit)	LC(bit)	HSC(bit)	D.b	KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	E	[D]	XXP
RTC	EN1	●	●	●	●	●																				
	PDT																									
	Q		●	●	●	●																				
	CDT																									

软元件使用注意事项

- X、Y、M、S、SM 为 BOOL 类型；
- X 不能作为输出。

使用的变量类型

指令	参数	变量类型
RTC	EN1	BOOL
	PDT	DATE_AND_TIME
	Q	BOOL
	CDT	DATE_AND_TIME

功能

- 当 EN1 为“FALSE”，输出 Q 以及 CDT 都为“0”相关时间为 DT#1970-1-1-00:00:00。一旦 EN1 变为“TRUE”，将会以 PDT 设置的时间开始，以秒计数并从 CDT 进行返回。当 EN 复位为 FALSE，CDT 将会复位成 DT#1970-1-1-00:00:00。

注意事项：

- 只有上升沿才会触发开始计数。

福州富昌维控电子科技有限公司

附录

附 1 特殊继电器（SM）

出错信息

SM 标号	名称	内容	R/W	掉电保存
SM0	最新出错信息	OFF:无错误 ON:有出错	R	×
SM1	保留			
SM2	出错解除	OFF→ON:清除错误请求 ON→OFF:错误清除完成	R/W	×
SM3	电池电压过低	OFF：正常 ON：电池电压过低	R	×
SM4	电池电压过低锁存	OFF：正常 ON：电池电压过低	R	×
SM5	保留		R	×
SM6	PLC 硬件错误	OFF:无错误 ON:有出错	R	×
SM7	PLC 通讯错误	OFF:无错误 ON:有出错	R	×
SM10	参数错误	OFF:无错误 ON:有出错	R	×
SM11	运算错误	OFF:无错误 ON:有出错	R	×
SM14	运算错误锁存	OFF:无错误 ON:有出错	R	×

系统信息

SM 标号	名称	内容	R/W	掉电保存
SM30	电池低电警告屏蔽	OFF:关闭（默认） ON:打开	R/W	×
SM31	清除所有非保持寄存器	OFF→ON:清除请求 ON→OFF:清除完成	R/W	×
SM32	清除所有保持寄存器	OFF→ON:清除请求 ON→OFF:清除完成	R/W	×
SM33	停机状态所有软元件不变	OFF:关闭（默认） ON:打开	R/W	×
SM34	PLC 所有的输出都为 OFF 状态	OFF:关闭（默认） ON:打开	R/W	×
SM35	电池低电 BAT 灯状态	OFF:BAT 灯无效（默认） ON:电池低电 BAT 灯亮	R/W	×
SM36~SM49	保留			
SM53~SM99	保留			

时钟信息

SM 标号	名称	内容	R/W	掉电保存
SM100	RUN 后始终为 ON		R	×
SM101	RUN 后始终为 OFF		R	×
SM102	RUN 后第一个周期为 ON		R	×
SM103	RUN 后第一个周期为 OFF		R	×
SM104	USB 供电模式	ON 时为 USB 供电模式。此时仅支持下载、时钟设置、密码设置等功能	R	×
SM105~SM106	保留		R	×
SM107	时钟停止和预置	停止时钟运行和显示	R/W	×
SM108	时钟读取显示停止	时钟内部运行，停止显示	R/W	×
SM109	1min 振荡时钟	每 30s 切换状态	R	×
SM110	1ms 振荡时钟	每 0.5ms 切换状态	R	×
SM111	10ms 振荡时钟	每 5ms 切换状态	R	×
SM112	100ms 振荡时钟	每 50ms 切换状态	R	×
SM113	1s 振荡时钟	每 500ms 切换状态	R	×
SM114	nms 振荡时钟	每(n/2)ms 切换状态，n 由 SD114 设置	R	×
SM115	ns 振荡时钟	每（n/2）s 切换状态 n 由 SD115 设置	R	×
SM116	±30 秒修正	时钟秒数小于 30S 时清零，大于 30S 时进位	R/W	×
SM117~SM119	保留		R	×

扫描信息

SM 标号	名称	内容	R/W	掉电保存
SM120	恒定扫描周期	OFF:未开启（默认） ON：开启	R/W	×
SM121	RUN、STOP 控制	OFF:STOP ON：RUN	R/W	×
SM122	梯形图看门狗功能开关	OFF:未开启 ON：开启（默认）	R/W	×

指令相关

SM 标号	名称	内容	R/W	掉电保存
SM151	进位标志	OFF：计算未发生进位 ON：计算发生进位	R	×
SM152	退位标志	OFF：计算未发生退位 ON：计算发生退位	R	×
SM153	零位标志	OFF：计算结果不为零 ON：计算结果为零	R	×
SM160	XCH 交换模式	OFF：参数 1 与参数 2 交换 ON：参数本身高八位和第八位交换	R/W	×
SM161	位处理模式（ASC、ASCI、BCC、CCD、CRC）	OFF：16 位处理模式 ON：8 位处理模式	R/W	×
SM165	SORT/SORT2 指令升降序选择	OFF:升序 ON：降序	R/W	×
SM167	HKY 指令 HEX 数据处理	OFF：数字键+功能键 ON ：16 进制键	R/W	×

SM168	SMOV 指令 16 进制处理	OFF：执行 BIN→BCD 转换 ON：不执行 BIN→BCD 转换	R/W	×
SM191	BINDA 输出字符数切换信号	OFF：输出 00H ON：无变化	R/W	×
SM224	BMOV 指令方向	OFF：正向传送 ON：反向传送	R/W	×
SM226	RAMP 指令模式	OFF：循环执行模式 ON：完成后保持	R/W	×
SM227	PR 指令模式	OFF ：8 个字节串行输出(固定为 8 个字符) ON ：16 个字节串行输出(1～16 个字符)	R/W	×
SM229	部分应用指令执行完成标志	OFF：指令未执行或执行中 ON：指令执行完成	R/W	×
SM240	STL 用：禁止转移	OFF：普通动作 ON：禁止发生状态转移	R/W	×
SM241	IST 指令转移开始	OFF：IST 指令未动作 ON：IST 指令转移开始	R/W	×
SM242	IST 指令对应启动输入的脉冲输出	OFF：未启动 ON：启动	R/W	×
SM243	IST 指令原点回归状态的结束标志用户程序控制	OFF：原点回归未结束 ON：原点回归结束	R/W	×
SM244	IST 指令检测到机械原点动作用户程序控制	OFF:非机械原点 ON：机械原点	R/W	×
SM245	STL 指令用：禁止模式切换时的全部输出复位用户程序控制	OFF:状态切换时输出全部复位 ON：状态切换时输出不动作		×
SM246	IST 指令用：有 STL 状态 ON	OFF:STL 监控有效为 OFF 时，或 STL 监控有效为 ON 且步进继电器(S 软元件)的全部为 OFF 时 ON：STL 监控有效为 ON，且步进继电器(S 软元件)的任意一个为 ON 时		×
SM247	STL 监控有效	OFF:无效 ON：使 STL 监控变为有效，动作中的状态编号(S0～S4095)按照从小到大的顺序被保存在特殊辅助继电器 SD240～SD247 中。	R/W	×
SM248	ANS 指令信号报警器动作	OFF：报警器未动作 ON：报警器动作	R/W	×
SM249	ANS 指令信号报警器有效	OFF：报警器无效 ON：报警器有效	R/W	×
SM340	DUTY 定时时钟输出 1	DUTY 指令的时机时钟输出	R	×
SM341	DUTY 定时时钟输出 2		R	×
SM342	DUTY 定时时钟输出 3		R	×
SM343	DUTY 定时时钟输出 4		R	×
SM344	DUTY 定时时钟输出 5		R	×

中断禁止

SM 标号	名称	内容	R/W	掉电保存
SM352	X0 上升沿中断	OFF：X0 上升沿中断有效 ON：禁用 X0 上升沿中断	R/W	×
SM353	X0 下降沿中断	OFF：X0 下降沿中断有效 ON：禁用 X0 下降沿中断	R/W	×
SM354	X1 上升沿中断	OFF：X1 上升沿中断有效 ON：禁用 X1 上升沿中断	R/W	×
SM355	X1 下降沿中断	OFF：X1 下降沿中断有效 ON：禁用 X1 下降沿中断	R/W	×

SM356	X2 上升沿中断	OFF：X2 上升沿中断有效 ON：禁用 X2 上升沿中断	R/W	×
SM357	X2 下降沿中断	OFF：X2 下降沿中断有效 ON：禁用 X2 下降沿中断	R/W	×
SM358	X3 上升沿中断	OFF：X3 上升沿中断有效 ON：禁用 X3 上升沿中断	R/W	×
SM359	X3 下降沿中断	OFF：X3 下降沿中断有效 ON：禁用 X3 下降沿中断	R/W	×
SM360	X4 上升沿中断	OFF：X4 上升沿中断有效 ON：禁用 X4 上升沿中断	R/W	×
SM361	X4 下降沿中断	OFF：X4 下降沿中断有效 ON：禁用 X4 下降沿中断	R/W	×
SM362	X5 上升沿中断	OFF：X5 上升沿中断有效 ON：禁用 X5 上升沿中断	R/W	×
SM363	X5 下降沿中断	OFF：X5 下降沿中断有效 ON：禁用 X5 下降沿中断	R/W	×
SM364	X6 上升沿中断	OFF：X6 上升沿中断有效 ON：禁用 X6 上升沿中断	R/W	×
SM365	X6 下降沿中断	OFF：X6 下降沿中断有效 ON：禁用 X6 下降沿中断	R/W	×
SM366	X7 上升沿中断	OFF：X7 上升沿中断有效 ON：禁用 X7 上升沿中断	R/W	×
SM367	X7 下降沿中断	OFF：X7 下降沿中断有效 ON：禁用 X7 下降沿中断	R/W	×

高速输入输出

SM 标号	名称	内容	R/W	掉电保存
SM400	HSC0 触点状态	OFF:计数未达到设定值 ON：计数达到设定值	R	×
SM401	HSC0 的运动方向	OFF:正方向 ON：反方向	R	×
SM405	HSC0 计数方向	OFF:递增计数 ON：递减计数	R/W	×
SM406~SM429	保留			
SM430	HSC1 触点状态	OFF:计数未达到设定值 ON：计数达到设定值	R	×
SM431	HSC1 的运动方向	OFF:正方向 ON：反方向	R	×
SM435	HSC1 计数方向	OFF:递增计数 ON：递减计数	R/W	×
SM436~SM459	保留			
SM460	HSC2 触点状态	OFF:计数未达到设定值 ON：计数达到设定值	R	×
SM461	HSC2 的运动方向	OFF:正方向 ON：反方向	R	×
SM465	HSC2 计数方向	OFF:递增计数 ON：递减计数	R/W	×
SM466~SM489	保留			
SM490	HSC3 触点状态	OFF:计数未达到设定值 ON：计数达到设定值	R	×
SM491	HSC3 的运动方向	OFF:正方向 ON：反方向	R	×
SM495	HSC3 计数方向	OFF:递增计数 ON：递减计数	R/W	×

福州富昌维控电子科技有限公司

SM496~SM519	保留			
SM520	HSC4 触点状态	OFF:计数未达到设定值 ON：计数达到设定值	R	×
SM521	HSC4 的运动方向	OFF:正方向 ON：反方向	R	×
SM525	HSC4 计数方向	OFF:递增计数 ON：递减计数	R/W	×
SM526~SM549	保留			
SM550	HSC5 触点状态	OFF:计数未达到设定值 ON：计数达到设定值	R	×
SM551	HSC5 的运动方向	OFF:正方向 ON：反方向	R	×
SM555	HSC5 计数方向	OFF:递增计数 ON：递减计数	R/W	×
SM556~SM579	保留			
SM580	HSC6 触点状态	OFF:计数未达到设定值 ON：计数达到设定值	R	×
SM581	HSC6 的运动方向	OFF:正方向 ON：反方向	R	×
SM585	HSC6 计数方向	OFF:递增计数 ON：递减计数	R/W	×
SM586~SM609	保留			
SM610	HSC7 触点状态	OFF:计数未达到设定值 ON：计数达到设定值	R	×
SM611	HSC7 的运动方向	OFF:正方向 ON：反方向	R	×
SM615	HSC7 计数方向	OFF:递增计数 ON：递减计数	R/W	×
SM616~SM639	保留			
SM1902	EtherCAT 运行标识（0：未启动或者正在启动　1：启动完成可运行 MC_POWER 指令）			

通讯

SM 标号	名称	内容	R/W	掉电保存
通讯 COM1				
SM2540	COM1 发送控制开启		R/W	×
SM2541	COM1 发送控制/发送提示		R/W	×
SM2542	COM1 接收控制开启		R/W	×
SM2543	COM1 接受控制/接收提示		R/W	×
SM2544	COM1 8 位模式		R/W	×
SM2560	COM1 通讯完成标志		R/W	×
SM2561	COM1 接收中标识		R/W	×
SM2562	COM1 发生重试		R/W	×
SM2563	COM1 通讯错误		R/W	×
SM2564	COM1 通讯超时		R/W	×
SM2565	保留			×
SM2566	保留			×

福州富昌维控电子科技有限公司

SM2567	保留			×
SM2568	保留			×
SM2569	保留			×
SM2570	保留			×
SM2571	保留			×
通讯 COM2				
SM2590	COM2 发送控制开启	OFF：数据发送将由 SM2591 控制 ON：数据自动发送	R/W	×
SM2591	COM2 发送控制/发送提示	OFF→ON：数据发送开始 ON→OFF：数据发送结束	R/W	×
SM2592	COM2 接收控制开启	OFF：数据接收满自动停止 ON：正常接收，不受标志 SM2593 影响	R/W	×
SM2593	COM2 接受控制/接收提示	OFF：数据未接收满或未收到数据（根据 SM2592 的状态） ON：数据接收满或收到数据（根据 SM2592 的状态）	R/W	×
SM2594	COM2 的 8 位模式（RS 自定义协议使用）	OFF：16 位模式 ON：8 位模式		×
SM2610	通讯完成标志	OFF：通讯未完成 ON：通讯完成	R/W	×
SM2611	接收中标识	OFF：未接收到数据 ON：数据正在接收	R/W	×
SM2612	发生重试	OFF：未发生重试 ON：发生重试	R/W	×
SM2613	通讯错误	OFF：无错误 ON：发生通讯错误	R/W	×
SM2614	通讯超时	OFF：正常通讯 ON：通讯超时	R/W	×
SM2615	保留			

以太网相关特殊软元件一览表

SM 标号	名称	内容	R/W	掉电保存
SM2681	显示当前网络信息	ON 后刷新当前 IP、子网掩码、默认网关。刷新完成后 OFF	R/W	×
SM2682	显示当前 MAC 信息	ON 后刷新当前 MAC。刷新完成后 OFF	R/W	×
SM2683	IP、子网掩码、网关修改标志	ON：可修改 OFF：不可修改 （设置为 ON 后，STOP->RUN 时进行修改，修改完成后自动变为 OFF）	R/W	√
SM2684	网卡连通状态	ON：网络连通 OFF：网络未连通，请检查网线是否链接	R	√
SM2692	MAC 地址修改标志	ON：可修改 OFF：不可修改 （设置为 ON 后，STOP->RUN 时进行修改，修改完成后自动变为 OFF）	R/W	√

附 2 特殊寄存器（SD）

出错信息

SD 标号	名称	内容	R/W	掉电保存
SD0	最新出错信息错误码	最新自诊断出错代码将被存储	R	X
SD1	保留			
SD2	设置电池最低电压	默认值：26（2.6V）	R/W	X
SD3	当前电池电压	单位：0.1V	R	X
SD4	电池电压锁存值	发生电池电压过低警告时，锁存出错时的电池电压值，单位 0.1V	R	X
SD5	AC/DC 掉电次数	记录当前电源掉电且自动重启的次数	R	X
SD6	PLC 硬件错误的错误代码	硬件错误的错误代码将被存储	R	X
SD7	PLC 通讯错误代码	通讯错误的错误代码将被存储	R	X
SD8	PLC 通讯错误步号低字	通讯错误的梯形图步号将被存储，双字	R	X
SD9	PLC 通讯错误步号高字		R	X
SD10	参数错误代码	参数错误的错误代码将被存储	R	X
SD11	运算错误代码	运算错误的错误代码将被存储	R	X
SD12	运算错误程序步号低字	运算错误的梯形图步号将被存储，双字	R	X
SD13	运算错误程序步号高字		R	X
SD14	运算错误的代码锁存	运算错误的错误代码将被存储，该错误无法被出错解除功能功能清除	R	X
SD15	运算错误程序的步号锁存低字	运算错误的梯形图步号将被存储，无法被出错解除功能清除，双字	R	X
SD16	运算错误程序的步号锁存高字		R	X
SD17	程序出错错误代码	PLC 发生不可预知的错误	R	X
SD18	AC220V 掉电恢复时间	AC220V 掉电后重新恢复的时间将被存储，单位：ms	R	X
SD19~SD29	保留			

系统信息

SD 标号	名称	内容	R/W	掉电保存
SD30	型号识别码	存储 PLC 型号标识码，无法修改	R	X
SD31	软件版本号	存储 PLC 软件版本号，无法修改	R	X
SD32	硬件版本号	存储 PLC 硬件版本号，无法修改	R	X
SD33	输入点数	存储 PLC 的输入点数，无法修改	R	X
SD34	输出点数	存储 PLC 的输出点数，无法修改	R	X
SD35	高速输入轴数	存储 PLC 的高速输入轴数，无法修改	R	X
SD36	高速输出轴数	存储 PLC 的高速输出轴数，无法修改	R	X
SD37、SD38	继电器标识	标识有多少输出点为继电器类型，使用掩码方式，每个位标识一个输出点数，为 1 代码是继电器类型	R	X
SD40~SD47	产品唯一 ID（16 个字节）	存储产品唯一识别码，无法修改	R	X
SD48	编译链接版本	存储 PLC 编译链接模块版本，无法修改	R	X
SD49	生产信息字符串	存储生产信息，ASCII 码保存		X
SD50~SD99	型号识别码	存储 PLC 型号标识码，无法修改	R	X

时钟信息

SD 标号	名称	内容	R/W	掉电保存
SD100	实时时钟秒(0~59)	PLC 内置 RTC 时钟	R	X
SD101	实时时钟分(0~59)		R	X
SD102	实时时钟小时(0~23)		R	X
SD103	实时时钟日(1~31)		R	X
SD104	实时时钟月(1~12)		R	X
SD105	实时时钟公历年(2000～2099)		R	X
SD106	实时时钟星期		R	X
SD107~SD113	保留			
SD114	nms 振荡时钟的 n 值	设置 SM114 振荡时钟 n 的值，默认 500ms	R/W	X
SD115	ns 振荡时钟的 n 值	设置 SM115 振荡时钟 n 的值，默认 2s	R/W	X
SD116~SD119	保留			

扫描信息

SM 标号	名称	内容	R/W	掉电保存
SD120	恒定扫描周期时间设定（ms）	默认:10ms	R/W	X
SD122	看门狗定时器时间设置值	单位 ms，默认 200ms	R/W	X
SD128	扫描周期当前值 ms 部分	• 当前扫描时间将被存储到 SD128、SD129 中。(计测是以 1μs 单位进行) SD128：存储 ms 的位(存储范围：0～65535) SD129：存储 μs 的位(存储范围：0～999) (例)当前扫描时间为 23.6ms 的情况下，按以下方式存储。 SD128=23 SD129=600 • STOP→RUN 时将被一次清零。	R	X
SD129	扫描周期当前值 us 部分		R	X
SD130	扫描周期最大值 ms 部分	去除初始执行程序的扫描时间后的扫描时间的最大值将被存储到 SD130、SD131 中。(计测是以 1μs 单位进行) SD130：存储 ms 的位(存储范围：0～65535) SD131：存储 μs 的位(存储范围：0～999) • STOP→RUN 时将被一次清零。	R	X
SD131	扫描周期最大值 us 部分		R	X
SD132	扫描周期最小值 ms 部分	去除初始执行程序的扫描时间后的扫描时间的最小值将被存储到 SD133、SD134 中。(计测是以 1μs 单位进行) SD130：存储 ms 的位(存储范围：0～65535) SD131：存储 μs 的位(存储范围：0～999) • STOP→RUN 时将被一次清零。	R	X
SD133	扫描周期最小值 us 部分		R	X
SD134	初始扫描时间 ms 部分	• 初始扫描时间将被存储到 SD134、SD135 中。(计测是以 1μs 单位进行) SD134：存储 ms 的位(存储范围：0～65535) SD135：存储 μs 的位(存储范围：0～999) • STOP→RUN 时将被一次清零。	R	X
SD135	初始扫描时间 us 部分		R	X
SD136	END 处理时间 ms 部分	• 扫描程序结束后，至下一个扫描开始为止的时间将被存储到 SD136、SD137 中。(计测是以 1μs 单位进行) SD136：存储 ms 的位(存储范围：0～65535) SD137：存储 μs 的位(存储范围：0～999) • STOP→RUN 时将被一次清零。	R	X
SD137	END 处理时间 us 部分		R	X

福州富昌维控电子科技有限公司

SD138	程序执行时间 ms 部分	恒定扫描等待时间(ms 单位) • 恒定扫描设置时的等待时间被存储在 SD138、SD139 中。(计测是 以 1μs 单位进行) SD138：存储 ms 的位(存储范围：0～65535) SD149：存储μs 的位(存储范围：0～999) • STOP→RUN 时将被一次清零。	R	X
SD139	程序执行时间 us 部分		R	X
SD140	恒定扫描周期等待时间 ms	• 1 个扫描中的扫描程序的执行时间被存储在 SD140、SD141 中。(计测是以 1μs 单位进行) SD140：存储 ms 的位(存储范围：0～65535) SD141：存储μs 的位(存储范围：0～999) • STOP→RUN 时将被一次清零。	R	X
SD141	恒定扫描周期等待时间 us		R	X
SD150	当前中断优先级	在中断程序的执行中，该中断存在的优先度将被存储。 0～2：执行中的中断程序的中断指针优先度	R	X
SD151	当前禁止的中断优先级	根据中断禁止指令(DI 指令)、指定优先度以下的中断禁止指令(DI 指令)、中断允许指令(EI 指令),处于中断禁止的优先度将被存储。 0：所有优先度的中断禁止(默认) 1：优先度 1、2 的中断禁止 2：优先度 2 的中断禁止 3：所有优先度的中断允许	R	X

指令相关

SD 标号	名称	内容	R/W	掉电保存
SD150	当前中断优先级	在中断程序的执行中，该中断存在的优先度将被存储。 1~3： 执行中的中断程序的中断指针优先度 0：未执行中断(默认)	R	X
SD151	当前禁止的中断优先级	根据中断禁止指令(DI 指令)、指定优先度以下的中断禁止指令(DI 指令)、中断允许指令(EI 指令),处于中断禁止的优先度将被存储。 0：所有优先度的中断禁止(默认) 1：优先度 1、2 的中断禁止 2：优先度 2 的中断禁止 3：所有优先度的中断允许	R	X
SD240	STL 用：ON 状态编号 1	STL 中 ON 状态的 S 软元件编号将被保存，最多保存 8 个，超出 8 个存储编号小的前 8 个。	R	X
SD241	STL 用：ON 状态编号 2		R	X
SD242	STL 用：ON 状态编号 3		R	X
SD243	STL 用：ON 状态编号 4		R	X
SD244	STL 用：ON 状态编号 5		R	X
SD245	STL 用：ON 状态编号 6		R	X
SD246	STL 用：ON 状态编号 7		R	X
SD247	STL 用：ON 状态编号 8		R	X
SD249	信号报警器 ON 状态最小编号	存储信号报警器 ON 的最小编号	R/W	X
SD340	时机时钟输出 1 用扫描数计数	DUTY 指令的时机时钟输出 1 用扫描数计数值	R/W	X
SD341	时机时钟输出 2 用扫描数计数	DUTY 指令的时机时钟输出 2 用扫描数计数值	R/W	X
SD342	时机时钟输出 3 用扫描数计数	DUTY 指令的时机时钟输出 3 用扫描数计数值	R/W	X
SD343	时机时钟输出 4 用扫描数计数	DUTY 指令的时机时钟输出 4 用扫描数计数值	R/W	X
SD344	时机时钟输出 5 用扫描数计数	DUTY 指令的时机时钟输出 5 用扫描数计数值	R/W	X

中断禁止

SD 标号	名称	内容	R/W	掉电保存
SD350~SD381	定时器中断禁止掩码	SIMASK 指令中断掩码，每个 bit 代表一个中断，具体查看 SIMAK 指令	R/W	X
SD382~SD388	高速计数器中断禁止掩码	SIMASK 指令中断掩码，每个 bit 代表一个中断，具体查看 SIMAK 指令	R/W	X

高速输入输出

SD 标号	名称	内容	R/W	掉电保存
SD400	HSC0 当前计数值低位	存储通道高速计数器的当前计数值,每 100us 更新一次	R	X
SD401	HSC0 当前计数值高位		R	X
SD402	HSC0 当前频率低位	存储通道高速计数器的当前频率，每 100us 更新一次	R	X
SD403	HSC0 当前频率高位		R	X
SD405	HSC0 模式(显示) 0:普通 IO 1:单相计数 2:AB 相计数	默认：普通 IO	R	X
SD420	HSC0 倍频（显示） 1：1 倍频 2：2 倍频 4：4 倍频	默认：1 倍频	R	X
SD421	HSC0 频率采样时间（ms）	计算通道高速计数器频率的采样时间，默认 1000ms	R/W	X
SD422	HSC0 输入滤波设置(0.01us)	范围 0~1700，默认 0	R	X
SD423	DHSCS、DHSCR、DHSZ 指令使用 HSC0 通道的优先级设置	[0,2]，最高优先级为 0，默认值 0	R/W	X
SD430	HSC1 当前计数值低位	存储通道高速计数器的当前计数值,每 100us 更新一次	R	X
SD431	HSC1 当前计数值高位		R	X
SD432	HSC1 当前频率低位	存储通道高速计数器的当前频率，每 100us 更新一次	R	X
SD433	HSC1 当前频率高位		R	X
SD435	HSC1 模式(显示) 0:普通 IO 1:单相计数 2:AB 相计数	默认：普通 IO	R	X
SD450	HSC1 倍频 1：1 倍频 2：2 倍频 4：4 倍频	默认：1 倍频	R	X
SD451	HSC1 频率采样时间（ms）	计算通道高速计数器频率的采样时间，默认 1000ms	R/W	X
SD452	HSC1 输入滤波设置(0.01us)	范围 0~1700，默认 0	R	X
SD453	DHSCS、DHSCR、DHSZ 指令使用 HSC1 通道的优先级设置	[0,2]，最高优先级为 0，默认值 0	R/W	X
SD460	HSC2 当前计数值低位	存储通道高速计数器的当前计数值,每 100us 更新一次	R	X
SD461	HSC2 当前计数值高位		R	X
SD462	HSC2 当前频率低位	存储通道高速计数器的当前频率，每 100us 更新一次	R	X
SD463	HSC2 当前频率高位		R	X
SD464	保留			

SD465	HSC2 模式(显示) 0:普通 IO 1:单相计数 2:AB 相计数	默认：普通 IO	R	X
SD480	HSC2 倍频 1：1 倍频 2：2 倍频 4：4 倍频	默认：1 倍频	R	X
SD481	HSC2 频率采样时间（ms）	计算通道高速计数器频率的采样时间，默认 1000ms	R/W	X
SD482	HSC2 输入滤波设置(0.01us)	范围 0~1700，默认 0	R	X
SD483	DHSCS、DHSCR、DHSZ 指令使用 HSC2 通道的优先级设置	[0,2]，最高优先级为 0，默认值 0	R/W	X
SD490	HSC3 当前计数值低位	存储通道高速计数器的当前计数值,每 100us 更新一次	R	X
SD491	HSC3 当前计数值高位		R	X
SD492	HSC3 当前频率低位	存储通道高速计数器的当前频率，每 100us 更新一次	R	X
SD493	HSC3 当前频率高位		R	X
SD494	保留			X
SD495	HSC3 模式(显示) 0:普通 IO 1:单相计数 2:AB 相计数	默认：普通 IO	R	X
SD510	HSC3 倍频 1：1 倍频 2：2 倍频 4：4 倍频	默认：1 倍频	R	X
SD511	HSC3 频率采样时间（ms）	计算通道高速计数器频率的采样时间，默认 1000ms	R/W	X
SD512	HSC3 输入滤波设置(0.01us)	范围 0~1700，默认 0	R	X
SD513	DHSCS、DHSCR、DHSZ 指令使用 HSC3 通道的优先级设置	[0,2]，最高优先级为 0，默认值 0	R/W	X
SD521	HSC4 当前计数值高位		R	X
SD522	HSC4 当前频率低位	存储通道高速计数器的当前频率，每 100us 更新一次	R	X
SD523	HSC4 当前频率高位		R	X
SD524	保留			
SD525	HSC4 模式(显示) 0:普通 IO 1:单相计数 2:AB 相计数	默认：普通 IO	R	X
SD540	HSC4 倍频 1：1 倍频 2：2 倍频 4：4 倍频	默认：1 倍频	R	X
SD541	HSC4 频率采样时间（ms）	计算通道高速计数器频率的采样时间，默认 1000ms	R/W	X
SD542	HSC4 输入滤波设置(0.01us)	范围 0~1700，默认 0	R	X
SD543	DHSCS、DHSCR、DHSZ 指令使用 HSC4 通道的优先级设置	[0,2]，最高优先级为 0，默认值 0	R/W	X
SD550	HSC5 当前计数值低位	存储通道高速计数器的当前计数值,每 100us 更新一次	R	X
SD551	HSC5 当前计数值高位		R	X
SD552	HSC5 当前频率低位	存储通道高速计数器的当前频率，每 100us 更新一次	R	X

福州富昌维控电子科技有限公司

SD553	HSC5 当前频率高位		R	X
SD554	保留			
SD555	HSC5 模式(显示) 0:普通 IO 1:单相计数 2:AB 相计数	默认：普通 IO	R	X
SD570	HSC5 倍频 1：1 倍频 2：2 倍频 4：4 倍频	默认：1 倍频	R	X
SD571	HSC5 频率采样时间（ms）	计算通道高速计数器频率的采样时间，默认 1000ms	R/W	X
SD572	HSC5 输入滤波设置(0.01us)	范围 0~1700，默认 0	R	X
SD573	DHSCS、DHSCR、DHSZ 指令使用 HSC5 通道的优先级设置	[0,2]，最高优先级为 0，默认值 0	R/W	X
SD580	HSC6 当前计数值低位	存储通道高速计数器的当前计数值,每 100us 更新一次	R	X
SD581	HSC6 当前计数值高位		R	X
SD582	HSC6 当前频率低位	存储通道高速计数器的当前频率，每 100us 更新一次	R	X
SD583	HSC6 当前频率高位		R	X
SD584	保留			
SD585	HSC6 模式(显示) 0:普通 IO 1:单相计数 2:AB 相计数	默认：普通 IO	R	X
SD600	HSC6 倍频 1：1 倍频 2：2 倍频 4：4 倍频	默认：1 倍频	R	X
SD601	HSC6 频率采样时间（ms）	计算通道高速计数器频率的采样时间，默认 1000ms	R/W	X
SD602	HSC6 输入滤波设置(0.01us)	范围 0~1700，默认 0	R	X
SD603	DHSCS、DHSCR、DHSZ 指令使用 HSC6 通道的优先级设置	[0,2]，最高优先级为 0，默认值 0	R/W	X
SD610	HSC7 当前计数值低位	存储通道高速计数器的当前计数值,每 100us 更新一次	R	X
SD611	HSC7 当前计数值高位		R	X
SD612	HSC7 当前频率低位	存储通道高速计数器的当前频率，每 100us 更新一次	R	X
SD613	HSC7 当前频率高位		R	X
SD614	保留			
SD615	HSC7 模式(显示) 0:普通 IO 1:单相计数 2:AB 相计数	默认：普通 IO	R	X
SD630	HSC7 倍频 1：1 倍频 2：2 倍频 4：4 倍频	默认：1 倍频	R	X
SD631	HSC7 频率采样时间（ms）	计算通道高速计数器频率的采样时间，默认 1000ms	R/W	X
SD632	HSC7 输入滤波设置(0.01us)	范围 0~1700，默认 0	R	X
SD633	DHSCS、DHSCR、DHSZ 指令使用 HSC7 通道的优先级设置	[0,2]，最高优先级为 0，默认值 0	R/W	X

右扩展模块

SD 标号	名称	内容	R/W	掉电保存
SD2081		预留	R	×
SD2082		预留	R	×
SD2083	扩展模块数	当前连接的扩展模块个数	R	×
SD2084~SD2115	扩展模块型号、版本号信息	bit0~bit9：版本号 bit10~bit15：型号 1/2：LX6_16EX 模块 3/4：LX6_16EYT 模块	R	×

EtherCAT 模块

SD 标号	名称	内容	R/W	掉电保存
SD1901	已连接伺服数量			
SD1902	上位机配置伺服数量			
SD2083	扩展模块数			
SD1904	Ethercat 激活时间 s			
SD1905	Ethercat 激活时间 ms			

输入滤波

SD 标号	名称	内容	R/W	掉电保存
SD2280	输入滤波点设置，默认 10ms	低字节 X0~X3，高字节 X4~X7	R/W	×
SD2281	输入滤波点设置，默认 10ms	低字节 X10~X13，高字节 X14~X17	R/W	×
SD2282	输入滤波点设置，默认 10ms	低字节 X20~X23，高字节 X24~X27	R/W	×
SD2283	输入滤波点设置，默认 10ms	低字节 X30~X33，高字节 X34~X37	R/W	×
SD2284	输入滤波点设置，默认 10ms	低字节 X40~X43，高字节 X44~X47	R/W	×
SD2285	输入滤波点设置，默认 10ms	低字节 X50~X53，高字节 X54~X57	R/W	×
SD2286~SD2287	输入滤波点设置，默认 10ms	.....	R/W	×

通讯

SD 标号	名称	内容	R/W	掉电保存
通讯 COM1				
SD2540	COM1 通讯口设置	默认：波特率 115200,停止位 1,数据位 8,校验位无。 具体设置方法查看 PORTPARA 指令说明。	R/W	√
SD2541	COM1 串口参数修改标识	串口参数若需要在 RUN 中修改，必须要计算出修改标识,写入正确的标识才能修改成功,修改成功后，该值自动清零，具体计算方式查看 PORTPARA 指令说明	R/W	√
SD2542	COM1 协议设置	0H ： Modbus 从站 2H ： Modbus RTU 从站 3H ： Modbus ASCII 从站 10H ： 自定义协议 20H ： Modbus RTU 主站 30H ： Modbus ASCII 主站	R/W	√
SD2543	COM1 协议修改标识	通讯协议若需要在 RUN 中修改，必须要计算出修改标识,写入正确的标识才能修改成功,修改成功后，该值自动清零，具体计算方式查看 PROTOCOL 指令。	R/W	√

SD2544	COM1 站号设置	范围：[0,255]，默认：0	R/W	√
SD2545	COM1 站号修改标识	通讯站号若需要在 RUN 中修改，必须要计算出修改标识，写入正确的标识才能修改成功，修改成功后，该值自动清零，具体计算方式查看 STATION 指令。	R/W	√
SD2546	COM1 发送间隔 0.1ms		R/W	√
SD2547	COM1 通讯超时设置 10ms		R/W	√
SD2548	COM1 超时重试次数		R/W	√
SD2549	COM1 字符间隔超时设置 0.1ms		R/W	√
SD2550	COM1 STX 值		R/W	√
SD2551	COM1 ETX 值		R/W	√
SD2555	PLC 上传下载超时时间，超时没有继续传输将会中断上传下载。	单位：100ms，默认：300（30s）		√
SD2560	COM1 接收到的数据量		R	√
SD2561	COM1 最后一次错误		R	√
SD2562	COM1 当前错误		R	√
SD2563	COM1 出错步数		R	√
SD2564	COM1 出错站号		R	√
SD2565	COM1 出错累计次数		R	√
SD2566	COM1 出错步数（双字）		R	√
SD2567			R	√
SD2568	保留		R	√
SD2569	保留		R	√
SD2570	保留		R	√
SD2571	保留		R	√
通讯 COM2				
SD2590	COM2 通讯口设置	默认：波特率 115200,停止位 1,数据位 8,校验位无 具体设置方法查看 PORTPARA 指令说明	R/W	√
SD2591	COM2 串口参数修改标识	串口参数若需要在 RUN 中修改，必须要计算出修改标识，写入正确的标识才能修改成功，修改成功后，该值自动清零，具体计算方式查看 PORTPARA 指令说明	R/W	√
SD2592	COM2 协议设置	0H ： Modbus 从站 2H ： Modbus RTU 从站 3H ： Modbus ASCII 从站 10H ： 自定义协议 20H ： Modbus RTU 主站 30H ： Modbus ASCII 主站	R/W	√
SD2593	COM2 协议修改标识	通讯协议若需要在 RUN 中修改，必须要计算出修改标识，写入正确的标识才能修改成功，修改成功后，该值自动清零，具体计算方式查看 PROTOCOL 指令	R/W	√
SD2594	COM2 站号设置	范围：[0,255]，默认：0	R/W	√
SD2595	COM2 站号修改标识	通讯站号若需要在 RUN 中修改，必须要计算出修改标识，写入正确的标识才能修改成功，修改成功后，该值自动清零，具体计算方式查看 STATION 指令	R/W	√
SD2596	COM2 发送间隔	单位：0.1ms，默认值:0	R/W	√
SD2597	COM2 通讯超时设置	单位：10ms,默认值：100ms	R/W	√
SD2598	COM2 超时重试次数	默认值：0	R/W	√
SD2599	COM2 字符间隔超时设置	单位：0.1ms，默认值:10（1ms）	R/W	√
SD2600	COM2 自定义协议起始符	默认值：0	R/W	√
SD2601	COM2 自定义协议结束符	默认值：0	R/W	√
SD2610	COM2 接收到的数据量	存储串口接收到的数据量	R	×
SD2611	COM2 最后一次错误	存储最后一次出现的通讯错误码	R	×

SD2612	COM2 当前错误	存储当前出现的通讯错误码	R	×
SD2613	COM2 出错步数	存储当前出现的通讯错误的梯形图步数	R	×
SD2614	COM2 出错站号	存储当前出现的通讯错误的站号	R	×
SD2615	COM2 出错累计次数	存储累计通讯错误次数	R	×

以太网相关特殊软元件一览表

SD 标号	名称	内容	R/W	掉电保存
SD2680	ip 地址第一字节	本机 IP 地址	R/W	√
SD2681	ip 地址第二字节		R/W	√
SD2682	ip 地址第三字节		R/W	√
SD2683	ip 地址第四字节		R/W	√
SD2684	子网掩码第一字节	本机子网掩码	R/W	√
SD2685	子网掩码第二字节		R/W	√
SD2686	子网掩码第三字节		R/W	√
SD2687	子网掩码第四字节		R/W	√
SD2688	默认网关第一字节	本机默认网关	R/W	√
SD2689	默认网关第二字节		R/W	√
SD2690	默认网关第三字节		R/W	√
SD2691	默认网关第四字节		R/W	√
SD2692	MAC 地址第一字节	本机 MAC 地址	R/W	√
SD2693	MAC 地址第二字节		R/W	√
SD2694	MAC 地址第三字节		R/W	√
SD2695	MAC 地址第四字节		R/W	√
SD2696	MAC 地址第五字节		R/W	√
SD2697	MAC 地址第六字节		R/W	√

日志信息

SD 标号	名称	内容	R/W	掉电保存
SD4000	写入梯形图次数低位	存储梯形图的下载总次数，掉电保存	R	√
SD4001	写入梯形图次数高位		R	√
SD4002	写入 PLC 参数次数低位	存储参数下载的总次数，掉电保存	R	√
SD4003	写入 PLC 参数次数高位		R	√
SD4004	写入密码次数低位	存储写入密码的总次数，掉电保存	R	√
SD4005	写入密码次数高位		R	√
SD4006	写入注释次数低位	存储注释下载的总次数，掉电保存	R	√
SD4007	写入注释次数高位		R	√
SD4008	总开机次数低位	存储 PLC 开机总次数，掉电保存	R	√
SD4009	总开机次数高位		R	√
SD4010	总开机时间低位	存储 PLC 开机总时间，掉电保存，单位 s	R	√
SD4011	总开机时间高位		R	√
SD4012	开机 RUN 总时间低位	存储 PLC 开机后 RUN 的总时间,掉电保存,单位 s	R	√
SD4013	开机 RUN 总时间高位		R	√
SD4014	本次开机 RUN 时间低位	存储 PLC 本次开机后 RUN 时间，单位 s	R	×
SD4015	本次开机 RUN 时间高位		R	×

附 3 错误码整理

PLC 硬件错误

错误码	说明	动作	处理方案	检测时机
1000	PLC 供电电压异常	停止运行 PWR 灯灭	提示供电电源异常，请更换电源	始终
1100	看门狗超时	停止运行	程序扫描时间超过了看门狗超时时间设置，修改超时时间(SD122)的设置值或程序。	始终
1200	FLASH 写入次数超限(上位机读取次数信息显示，SD4000 双字)	停止运行	FLASH 的写入次数超过 2 万次，需更换 PLC	下载程序时
1201	读取生产信息失败	停止运行	FLASH 损坏，需更换 PLC	STOP→RUN 时
1380	检测出电池电压过低，会影响 PLC 时钟运行	继续运行	确认电池的连接情况。 及时更换电池。	END 指令执行时
1382	用户自定义异常	继续运行	无需处理	END 指令执行时
1600	硬件错误	停止运行	返厂维修	STOP→RUN 时

梯形图执行错误

错误码	说明	动作	处理方案	检测时机
1400	STOP→RUN 时引起的程序异常	停止运行	检测参数配置是否配置错误、梯形图是否使用了不支持的指令。	STOP→RUN 时
1401	STOP 时引起的程序异常	停止运行		STOP 时
1402	梯形图执行时引起的程序异常	停止运行		梯形图运行时
1403	RUN→STOP 引起的程序异常	停止运行		RUN→STOP 时
1500	梯形图转换在 END 指令中执行 OUT T 的标号错误	停止运行	检测梯形图中 OUT T 指令	END 指令执行时
1501	空指针错误	停止运行	检测是否使用了未定义的程序名	初始化时

PLC 参数错误

错误码	说明	动作	处理方案	检测时机
2000	程序分配的 I/O 点数与实际硬件 I/O 点数不同	停止运行	检查 I/O 点数配置	STOP→RUN 时
2001	对高速脉冲输入输出模块设置了标准输入输出模块的参数	停止运行	检查输入点参数配置	STOP→RUN 时
2002	安装的扩展模块超出最大个数	停止运行	减少扩展模块的安装	STOP→RUN 时
2003	X 点复用，既把同一个点作为 AB 相高速输入，又作为单向输入或中断输入	停止运行	检查输入模式配置	STOP→RUN 时
2004	配置高速输入 IO 错误,CNTCFG 指令参数写入	停止运行	检查 CNTCFG 的参数 1 的值	指令运行时
2100	存储器容量的设置错误	停止运行	检查存储器容量的设置	STOP→RUN 时
2101	保持区域的设置错误	停止运行	检查保持寄存器的设置	STOP→RUN 时
2102	注释区域的设置错误	停止运行	检查注释的设置	STOP→RUN 时
2103	文件寄存器的区域设置错误	停止运行	检查文件存储器区域设置	STOP→RUN 时
2200	程序校验不一致	停止运行	上下位机程序不一致，请重新上传或下载	STOP→RUN 时
2201	特殊参数的校验和不一致	停止运行	上下位机参数不一致，请重新上传或下载	STOP→RUN 时
2202	特殊参数设置错误	停止运行	检查特殊参数的设置	STOP→RUN 时

2203	PLC Editor2 和 PLC 固件版本不一致	停止运行	请通过查 PLC Editor2 与固件版本的对应表，重新安装 PLC Editor2 或升级固件	STOP→RUN 时
2380	当前扫描周期超出了恒定扫描周期设定值	继续运行	修改恒定扫描周期设置	END 指令执行时
2400	事件超出最大范围	停止运行	检查事件设置是否超过 100 个	STOP→RUN 时
2401	事件执行程序为空	停止运行	是否建立正确的事件程序	STOP→RUN 时
2402	事件清除程序为空	停止运行	是否建立正确的事件程序	STOP→RUN 时
2403	定时中断超出最大范围	停止运行	检查定时中断设置是否超过 100 个	STOP→RUN 时
2404	定时中断执行程序为空	停止运行	是否建立正确的定时中断程序	STOP→RUN 时
2405	定时中断优先级设置错误	停止运行	检查定时中断优先级是否设置[0,2]	STOP→RUN 时
2406	高速计数中断超出最大范围	停止运行	检查高速计数中断设置是否超过 100 个	STOP→RUN 时
2407	高速计数中断执行程序为空	停止运行	是否建立正确的高速计数中断程序	STOP→RUN 时
2408	高速计数优先级设置错误	停止运行	检查高速计数中断优先级是否设置[0,2]	STOP→RUN 时
2409	外部中断超出最大范围	停止运行	检查外部中断设置是否超过 16 个	STOP→RUN 时
240A	外部中断执行程序为空	停止运行	是否建立正确的外部中断程序	STOP→RUN 时
240B	外部中断优先级设置错误	停止运行	检查外部中断优先级是否设置[0,2]	STOP→RUN 时
2500	高速计数器通道超出最大范围	停止运行	检测高速输入配置	STOP→RUN 时
2501	高速计数器的模式设置错误	停止运行	检测高速输入配置	STOP→RUN 时
2502	高速计数器的倍频设置错误	停止运行	检测高速输入配置	STOP→RUN 时
2503	高速计数器的计数方向设置错误	停止运行	检测高速输入配置	STOP→RUN 时
2504	使用了高速计数器中断，但没有使用 OUT HSC 指令开启高速计数器，且去修改了 HSC 软元件的值	继续运行	查看是否有关闭 OUT HSC 指令的触点	100us 中断执行时
2580	高速计数器开启后，但未配置该轴高速计数器使能	继续运行	查看工程管理→参数→高速计数配置	OUT HSC 指令执行时
2581	使用了高速计数器中断，但没有使用 OUT HSC 指令开启高速计数器，且去修改了 HSC 软元件的值	继续运行	查看是否有关闭了 OUT HSC 指令的触点	100us 中断执行时
2582	使用了 REF 指令刷新高速计数值，但没有 OUT HSC 指令开启该通道的高速计数器	继续运行	1、查看工程管理→参数→高速计数配置→是否使用 2、查看是否有关闭了 OUT HSC 指令的触点	REF 指令执行时
2600	高速计数器与中断的 X 点有冲突	停止运行	检测高速输入或外部中断配置	STOP→RUN 时

EtherCAT 配置错误

错误码	说明	动作	处理方案	检测时机
2781	EtherCAT 配置文件加载失败。文件被损坏	继续运行	请重新下载 EtherCAT 配置文件	EtherCAT 初始化时
2782	无法找到 EtherCAT 配置文件	继续运行	请重新下载 EtherCAT 配置文件	EtherCAT 初始化时
2783	打开 EtherCAT 配置文件失败	继续运行	请重新下载 EtherCAT 配置文件	EtherCAT 初始化时
2784	解析 EtherCAT 配置失败，找不到对应的值或者参数	继续运行	请重新下载 EtherCAT 配置文件，或检查 EtherCAT 上下位机版本是否匹配	EtherCAT 初始化时
2785	初始化失败	继续运行	需要联系厂商	EtherCAT 初始化时
2786	获取从站设备失败	继续运行	请检查是否有链接 EtherCAT 从设备	EtherCAT 初始化时
2787	从站参数设置失败	继续运行	检查配置是否正确	EtherCAT 初始化时

2788	EtherCAT 从设备激活失败	继续运行	检查配置是否正确	EtherCAT 初始化时
2789	EtherCAT 启动失败	继续运行	检查配置是否正确，或联系厂商	EtherCAT 初始化时
278A	配置的 EtherCAT 从站个数与链接的从站个数不匹配	继续运行	检查设置以及链接的 EtherCAT 从设备个数	EtherCAT 初始化时
278B	接入的伺服机型与 EtherCAT 配置表中的机型不一致	继续运行	检查使用的配置是否与实际机型匹配	EtherCAT 初始化时
278C	初始化失败	继续运行	需要联系厂商	EtherCAT 初始化时
278D	初始化失败	继续运行	需要联系厂商	EtherCAT 初始化时
278E	ETHERCAT 停止错误	继续运行	需要联系厂商	EtherCAT 初始化时
278F	初始化失败	继续运行	需要联系厂商	EtherCAT 初始化时
2790	初始化失败	继续运行	需要联系厂商	EtherCAT 初始化时
2791	初始化超时	继续运行	需要联系厂商	EtherCAT 初始化时
2792	初始化超时	继续运行	需要联系厂商	EtherCAT 初始化时
2793	初始化超时	继续运行	需要联系厂商	EtherCAT 初始化时

PLC 通讯错误

错误码	说明	动作	处理方案	检测时机
3080	COM1 数据接收错误	继续运行	通讯线上可能有干扰，建议接上地线。	指令执行时
3081	COM1 数据接收超时	继续运行	1、检查接线，检查串口参数设置是否主从对应。 2、检查是否有出现干扰。 3、检查是否从站响应来不及导致的。建议尝试将发送间隔 SD2546 加大。	指令执行时
3082	COM1 CRC 校验错误	继续运行	通讯线上可能有干扰，建议接上地线。	指令执行时
3083	COM1 LRC 校验错误	继续运行	通讯线上可能有干扰，建议接上地线。	指令执行时
3084	COM1 站号配置错误	继续运行	检查从站站号设置，以及检查从站接收发送机制是否有问题。	指令执行时
3085	COM1 发送缓冲区溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
3086	COM1 功能码错误	继续运行	检查设置的功能码是否是 PLC 支持的功能码	指令执行时
3087	COM1 地址错误	继续运行	检查从站是否有这个地址（请查阅 Modbus 异常 02）	指令执行时
3088	COM1 长度错误	继续运行	检查通讯长度是否超出 Modbus 范围	指令执行时
3089	COM1 数据错误	继续运行	检查指令的参数是否有错误。 检查从站是否支持这个值的设置。（请查阅 Modbus 异常 03）	指令执行时
308A	COM1 从站忙	继续运行	从站返回信息：从站忙（请查阅 Modbus 异常 06）	指令执行时
308B	COM1 从站不支持功能码	继续运行	检查从站是否支持这个功能码（请查阅 Modbus 异常 01）	指令执行时
308C	COM1 从站故障	继续运行	从站返回信息：从站故障，请检查从站是否故障（请查阅 Modbus 异常 04）	指令执行时
308D	COM1 从站确认	继续运行	从站返回信息：从站确认（请查阅 Modbus 异常 05）	指令执行时
308E	COM1 目前不支持这个指令或功能	继续运行	当设置为从站协议时不能使用主站相关的通讯指令或者主站功能。请更改协议或者关闭对应指令前的触点或者关闭对应的通讯功能。	指令执行时
308F	COM1 发送超时	继续运行	出现该错误提示，请联系技术人员	指令执行时
3090	接收缓存溢出	继续运行	检查对方设备是否在一直发送数据	指令执行时

30A0	COM1 不可用网关	继续运行	从站返回信息：不可用网关（请查阅 Modbus 异常 0A）	指令执行时
30A1	COM1 指示没有从目标设备中获得响应。通常意味着设备未在网络中	继续运行	从站返回信息：设备未在网络中（请查阅 Modbus 异常 0B）	指令执行时
3180	COM2 数据接收错误	继续运行	通讯线上可能有干扰，建议接上地线。	指令执行时
3181	COM2 数据接收超时	继续运行	1、检查接线，检查串口参数设置是否主从对应。 2、检查是否有出现干扰。 3、检查是否从站响应来不及导致的。建议尝试将发送间隔 SD2546 加大。	指令执行时
3182	COM2 CRC 校验错误	继续运行	通讯线上可能有干扰，建议接上地线。	指令执行时
3183	COM2 LRC 校验错误	继续运行	通讯线上可能有干扰，建议接上地线。	指令执行时
3184	COM2 站号配置错误	继续运行	检查从站站号设置，以及检查从站接收发送机制是否有问题。	指令执行时
3185	COM2 发送缓冲区溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
3186	COM2 功能码错误	继续运行	检查设置的功能码是否是 PLC 支持的功能码	指令执行时
3187	COM2 地址错误	继续运行	检查从站是否有这个地址（请查阅 Modbus 异常 02）	指令执行时
3188	COM2 长度错误	继续运行	检查通讯长度是否超出 Modbus 范围	指令执行时
3189	COM2 数据错误	继续运行	检查指令的参数是否有错误。 检查从站是否支持这个值的设置。（请查阅 Modbus 异常 03）	指令执行时
318A	COM2 从站忙	继续运行	从站返回信息：从站忙（请查阅 Modbus 异常 06）	指令执行时
318B	COM2 从站不支持功能码	继续运行	检查从站是否支持这个功能码（请查阅 Modbus 异常 01）	指令执行时
318C	COM2 从站故障	继续运行	从站返回信息：从站故障，请检查从站是否故障（请查阅 Modbus 异常 04）	指令执行时
318D	COM2 从站确认	继续运行	从站返回信息：从站确认（请查阅 Modbus 异常 05）	指令执行时
318E	COM2 目前不支持这个指令或功能	继续运行	当设置为从站协议时不能使用主站相关的通讯指令或者主站功能。请更改协议或者关闭对应指令前的触点或者关闭对应的通讯功能。	指令执行时
318F	COM2 发送超时	继续运行	出现该错误提示，请联系技术人员	指令执行时
31A0	COM2 不可用网关	继续运行	从站返回信息：不可用网关（请查阅 Modbus 异常 0A）	指令执行时
31A1	COM2 指示没有从目标设备中获得响应。通常意味着设备未在网络中	继续运行	从站返回信息：设备未在网络中（请查阅 Modbus 异常 0B）	指令执行时
31C0	PLCLINK 表头异常。	继续运行	重新下载程序。	应用指令或功能执行时
31C1	通讯口暂不支持 PLCLINK 这个功能。	继续运行	升级固件	应用指令或功能执行时
31C2	PLCLINK 表格版本不兼容。	继续运行	重新下载程序。	应用指令或功能执行时
31C3	PLCLINK 命令条数超出范围。目前限制为 1~255 条。	继续运行	检查命令条数	应用指令或功能执行时
31C4	PLCLINK 表格中站号超出范围。	继续运行	检查表格中的站号。	应用指令或功能执行时

31C5	PLCLINK 表格超出软元件范围。	继续运行	检查表格对应的软元件范围。	应用指令或 功能执行时
31C6	PLCLINK 表格命令使用的软元件超出范围。	继续运行	检查表格中每条命令使用的软元件.	应用指令或 功能执行时

PLC 运算错误

错误码	说明	动作	处理方案	检测时机
3680	以太网数据接收错误	继续运行	检查环境是否存在干扰	指令执行时
3681	以太网数据接收超时	继续运行	检查网线是否松动 检查网络对端是否故障无法发出数据。 检查网络对端是否有响应不及时导致数据来不及， 这个原因可以尝试加大套接字配置中的接收超时时间。	指令执行时
3684	Modbus TCP 站号配置错误	继续运行	检查从站站号设置。 以及检查从站接收发送机制是否有问题。	指令执行时
3685	Modbus TCP 发送缓冲区溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
3686	Modbus TCP 功能码错误	继续运行	检查设置的功能码是否是 PLC 支持的功能码	指令执行时
3687	Modbus TCP 地址错误	继续运行	检查从站是否有这个地址（请查阅 Modbus 异常 02）	指令执行时
3688	Modbus TCP 长度错误	继续运行	检查通讯长度是否超出 Modbus 范围	指令执行时
3689	Modbus TCP 数据错误	继续运行	检查指令的参数是否有错误。 检查从站是否支持这个值的设置。 （请查阅 Modbus 异常 03）	指令执行时
368A	Modbus TCP 从站忙	继续运行	从站返回信息：从站忙（请查阅 Modbus 异常 06）	指令执行时
368B	Modbus TCP 从站不支持功能码	继续运行	检查从站是否支持这个功能码 （请查阅 Modbus 异常 01）	指令执行时
368C	Modbus TCP 从站故障	继续运行	从站返回信息：从站故障，请检查从站是否故障（请查阅 Modbus 异常 04）	指令执行时
368D	Modbus TCP 从站确认	继续运行	从站返回信息：从站确认 （请查阅 Modbus 异常 05）	指令执行时
368E	Modbus TCP 目前协议不支持这个指令	继续运行	当设置为从站协议时不能使用 RS 指令，请更改协议或者关闭 RS 指令前的触点	指令执行时
368F	网口发送超时	继续运行	出现该错误提示，请联系技术人员	指令执行时
3690	接收缓存溢出	继续运行	检查对方设备是否在一直发送数据	指令执行时
36A0	Modbus TCP 不可用网关	继续运行	从站返回信息：不可用网关 （请查阅 Modbus 异常 0A）	指令执行时
36A1	Modbus TCP 指示没有从目标设备中获得响应。通常意味着设备未在网络中	继续运行	从站返回信息：设备未在网络中 （请查阅 Modbus 异常 0B）	指令执行时
36C0	Modbus TCP 事务处理标识符错误	继续运行	检查是否存在网络拥堵，数据来不及接收。	指令执行时

福州富昌维控电子科技有限公司

36C1	Modbus TCP 服务器可用链接满	继续运行	查看 SD2702 和 SD2703 是否有过多的客户端来链接	指令执行时
36C8	以太网协议栈堆空间不足	继续运行	出现该错误提示，请联系技术人员	指令执行时
36C9	连接数超限制	继续运行	确认总连接数是否超限制	指令执行时
36CA	上次发送未完成	继续运行	使用发送完成标记，判断当前发送完成后，再发下一个	指令执行时
36CB	TCP 写入异常	继续运行	使用标记位软元件判断当前连接状态是否正常，不正常时，不进行数据发送，例如关闭中标记置位后，不进行数据发送动作。	指令执行时
36CC	TCP 输出异常	继续运行	出现该错误提示，请联系技术人员	指令执行时
36CD	IP 地址信息已使用	继续运行	请检查是否存在使用相同地址信息的连接存在	指令执行时
36CE	服务器接受连接错误	继续运行	出现该错误提示，请联系技术人员	指令执行时
36CF	TCP 接收缓存溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
36D0	TCP 连接失败	继续运行	可能在网线未连接的状态下开启 TCP 客户端导致	指令执行时
36D1	主动关闭连接时异常	继续运行	出现该错误提示，请联系技术人员	指令执行时
36D2	协议栈内部发生异常关闭	继续运行	可能为长时间得不到应答被关闭。检查对端是否在线，是否能 ping 通。	指令执行时
36D3	对端发起 RST 链接	继续运行	1、检查对端是否发起异常关闭。2、作为客户端，对端连接数已满，或对方端口未打开导致。	指令执行时
36D4	协议栈发生单端关闭	继续运行	出现该错误提示，请联系技术人员	指令执行时
36D5	存在 IP 地址冲突	继续运行	局域网内存在相同 IP 设备,请更改 IP 地址	指令执行时
36D6	存在 MAC 地址冲突	继续运行	局域网内存在相同 MAC 设备，请更改 MAC 地址	指令执行时
36D7	TCP 发送缓存溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
36D8	UDP 连接异常	继续运行	可能为 IP 地址，端口号已被使用导致	指令执行时
36D9	UDP 发送缓存溢出	继续运行	出现该错误提示，请联系技术人员	指令执行时
36DA	UDP 发送时内存空间不足	继续运行	出现该错误提示，请联系技术人员	指令执行时
36DB	UDP 发送失败	继续运行	出现该错误提示，请联系技术人员	指令执行时
36DC	UDP 内存释放失败	继续运行	出现该错误提示，请联系技术人员	指令执行时
36DD	UDP 接收缓存溢出	继续运行	UDP 接收的数据长度超过限制值 512 导致	指令执行时
4080	除法指令中除数为 0	继续运行	修改应用指令参数	应用指令执行时
4081	应用指令计算数据溢出	继续运行	修改应用指令参数	应用指令执行时
4082	应用指令中输入了无法转换的数据类型	继续运行	修改应用指令参数	应用指令执行时
4083	应用指令中输入了-0、非规格化数、非数、±∞的任意一个数据	继续运行	修改应用指令参数	应用指令执行时
4084	应用指令中输入了超出可指定范围的数据(如参数 1 规定为 0~1，设置 2)	继续运行	修改应用指令参数	应用指令执行时

4085	读应用指令中输出结果超出软元件范围（如 D 软元件最大 D7999，使用到 D8000）	继续运行	修改应用指令参数	应用指令执行时
4086	写应用指令中输出结果超出软元件范围（如 D 软元件最大 D7999，使用到 D8000）	继续运行	修改应用指令参数	应用指令执行时
4087	应用指令参数使用了不支持的软元件	继续运行	修改应用指令参数	应用指令执行时
4088	多条应用指令同时使用了同一个轴并且都已经启动	继续运行	修改应用指令参数	应用指令执行时
4089	应用指令超出限制使用条数	继续运行	检查梯形图中是否使用了有限制使用条数的指令，并且超出限制	应用指令执行时
408A	字符串读取长度超出，字符串连续长度超过限制（目前是 400）或是超出了指令内部的限制	继续运行	修改读取字符串长度	应用指令执行时
408B	字符串读取时读取到软元件最大范围但是没有找到 00H 最为结束	继续运行	查看字符串结束符	应用指令执行时
408E	多条应用指令参数使用了同一软元件，但该指令不允许软元件复用	继续运行	查看 DUTY 指令是否使用了相同的 SM 进行输出	应用指令执行时
408F	使用的固件不支持该指令，请升级到最新固件	继续运行	升级到包含该指令的固件	应用指令执行时
4090	指定范围的两个参数先后顺序异常，例 D10 D100 指定范围是允许的但 D100 D10 不允许	继续运行	检查并修改指令的参数。	应用指令执行时
4100	FOR~NEXT 指令使用个数不对应或 FOR~NEXT 超出最大嵌套层次	停止运行	修改梯形图 FOR~NEXT 指令对应关系	NEXT 和 END 指令执行时
4180	没有 CJ、 CALL 的跳转目标地址,变址修饰的结果、标签未定义，以及在 P0~P4095 以外时在 CALL 指令中执行了 P63。因为 P63 是向 END 跳转的标签，所以不能在 CALL 指令中使用	继续运行	修改应用指令参数	应用指令执行时
4181	CJ 指令超出最大嵌套层次	继续运行	修改应用指令参数	应用指令执行时
4102	CALL 指令超出最大嵌套层次	继续运行	修改应用指令参数	应用指令执行时
4183	中断超出最大嵌套层次	继续运行	修改应用指令参数	应用指令执行时
4185	EI 指令出栈错误	继续运行	修改应用指令参数	应用指令执行时
4186	BREAK 不在 FOR~NEXT 指令内	继续运行	修改应用指令参数	应用指令执行时
4187	MC~MCR 超出了最大嵌套范围	继续运行	查看 MC、MCR 的嵌套关系	应用指令执行时
4188	MC 嵌套结构使用 N 时，未按照从小到大的顺序	继续运行	修改 MC 对应的 N 嵌套	应用指令执行时
4189	SIMASK 指令指定了未设置的中断	继续运行	修改 SIMASK 指定的中断名或该中断配置	应用指令执行时
4D80	采样时间(Ts)超出了对象范围(Ts≤0)	继续运行	修改应用指令参数	应用指令执行时
4D81	输入滤波器常数(α)超出了对象范围(α＜0 或 1023 ＜ α)	继续运行	修改应用指令参数	应用指令执行时
4D82	最大上升率(deltaT)超出了对象范围(deltaT＜0 或 32000 ＜ deltaT)	继续运行	修改应用指令参数	应用指令执行时
4D83	比例增益(Kp)超出了对象范围(Kp＜0)	继续运行	修改应用指令参数	应用指令执行时
4D84	积分增益(Ki)超出了对象范围(Ki＜0)	继续运行	修改应用指令参数	应用指令执行时

4D85	微分增益(Kd)超出了对象范围(Kd<0)	继续运行	修改应用指令参数	应用指令执行时
4D86	采样时间(Ts)<运算周期	继续运行	修改应用指令参数	应用指令执行时
4D87	比例增益(Kp)超出了对象范围(Kp＜1 或 Kp>3000)	继续运行	修改应用指令参数	应用指令执行时
4D88	积分时间(Ti)超出了对象范围(Ti<0 或 Ti>3600)	继续运行	修改应用指令参数	应用指令执行时
4D89	微分时间(Td)超出了对象范围(Td＜0 或 Td>1000)	继续运行	修改应用指令参数	应用指令执行时
4D90	PID 输出上限小于下限	继续运行	修改应用指令参数	应用指令执行时
4E80	电子凸轮表格载入错误	继续运行	修改应用指令参数	指令运行时
4E81	当前编号的表格有凸轮正在使用	继续运行	修改应用指令参数	指令运行时
4E82	表格地址出错	继续运行	修改应用指令参数	指令运行时
4E83	表格超出软元件范围	继续运行	修改应用指令参数	指令运行时
4EC0	电子齿轮比设置错误	继续运行	修改应用指令参数	指令运行时
4F80	DHSZ 指令最小范围≥最大范围	继续运行	修改应用指令参数	指令运行时
4F81	DHSCS、DHSCR、DHSZ 指令开启但没有使用 OUT HSC 指令开启高速计数器计数	继续运行	1、查看工程管理→参数→高速计数配置→是否使用 2、查看是否有关闭了 OUT HSC 指令的触点	指令运行时
4F90	半径模式下，半径不在范围内。	继续运行	修改应用指令参数	指令运行时
4F92	圆心/半径模式下，设置的终点坐标与理论圆上终点偏差较大引起象限计算错误。	继续运行	修改应用指令参数	指令运行时
4F93	半径模式下，弦长大于直径。	继续运行	修改应用指令参数	指令运行时
4F95	圆心模式下，圆心与起点距离，圆心与终点距离两者相距不在[-10-10]范围内。	继续运行	修改应用指令参数	指令运行时
4F97	圆心模式下，计算出来的半径距离大于半径最大表示范围正负 800000 脉冲。	继续运行	修改应用指令参数	指令运行时
4F98	螺旋插补错误,Z 轴为主轴（Z 轴坐标大于圆平面虚拟主轴个数）。	继续运行	修改应用指令参数	指令运行时
4F99	螺旋插补错误,Z 轴为 0。	继续运行	修改应用指令参数	指令运行时
4F9B	导程设置超范围（即出现导程 $K \le 0$ ）。	继续运行	修改应用指令参数	指令运行时
5080	以太网套接字已经建立连接，无法重复打开	继续运行	检查 SOCOPEN 指令是否重复执行。	应用指令执行时
5081	以太网套接字未打开,无法操作	继续运行	检查 SOCOPEN 指令（d2）参数的已连接标志位是否置位。	应用指令执行时
5082	以太网指令输入的套接字 ID 超出范围	继续运行	修改应用指令参数	应用指令执行时
5083	创建 TCP 服务器失败	继续运行	检查连接是否占满	应用指令执行时
5084	创建链接失败	继续运行	检查连接是否占满	应用指令执行时
5086	以太网指令使用的套接字 ID 未在上位机进行配置或配置后未使能	继续运行	检查上位机以太网配置	应用指令执行时

5087	SOCRECV 指令接收超时	继续运行	1、检查网络连接是否正常 2、检查网络对端是否有数据发出	应用指令执行时
5088	SOCMTCP 指令指定套接字使用了非 TCP 客户端的配置模式	继续运行	检查上位机以太网配置	应用指令执行时
5089	以太网套接字配置 TCP 服务器时，指定本机端口为 502	继续运行	502 端口已被内部占用，请修改本机端口号	应用指令执行时
508A	UDP 端口制定为 1092	继续运行	UDP 端口 1092 被系统内部占用无法使用，请修改本机端口号	应用指令执行时
5090	网线连接异常	继续运行	检查网线是否连接上	应用指令执行时

右扩展模块错误（报通讯错误）

错误码	说明	动作	处理方案	检测时机
7080	扩展模块和校验出错	继续运行	检测扩展模块与主机的链接或是否有外部干扰	指令运行时
7081	扩展模块通讯报文异常	继续运行	检测扩展模块与主机的链接或是否有外部干扰	指令运行时
7082	FROM/TO 指令错误	继续运行	检测扩展模块与主机的链接	指令运行时
7083	扩展模块访问异常	继续运行	检测扩展模块与主机的链接	指令运行时

附 4 ASCII 代码对照表

ASCII 代码对照表

Bin (二进制)	Oct (八进制)	Dec (十进制)	Hex (十六进制)	缩写/字符	解释
0000 0000	0	0	0x00	NUL(null)	空字符
0000 0001	1	1	0x01	SOH(start of headline)	标题开始
0000 0010	2	2	0x02	STX (start of text)	正文开始
0000 0011	3	3	0x03	ETX (end of text)	正文结束
0000 0100	4	4	0x04	EOT (end of transmission)	传输结束
0000 0101	5	5	0x05	ENQ (enquiry)	请求
0000 0110	6	6	0x06	ACK (acknowledge)	收到通知
0000 0111	7	7	0x07	BEL (bell)	响铃
0000 1000	10	8	0x08	BS (backspace)	退格
0000 1001	11	9	0x09	HT (horizontal tab)	水平制表符
0000 1010	12	10	0x0A	LF (NL line feed, new line)	换行键
0000 1011	13	11	0x0B	VT (vertical tab)	垂直制表符
0000 1100	14	12	0x0C	FF (NP form feed, new page)	换页键
0000 1101	15	13	0x0D	CR (carriage return)	回车键
0000 1110	16	14	0x0E	SO (shift out)	不用切换
0000 1111	17	15	0x0F	SI (shift in)	启用切换
0001 0000	20	16	0x10	DLE (data link escape)	数据链路转义
0001 0001	21	17	0x11	DC1 (device control 1)	设备控制 1
0001 0010	22	18	0x12	DC2 (device control 2)	设备控制 2
0001 0011	23	19	0x13	DC3 (device control 3)	设备控制 3
0001 0100	24	20	0x14	DC4 (device control 4)	设备控制 4
0001 0101	25	21	0x15	NAK (negative acknowledge)	拒绝接收
0001 0110	26	22	0x16	SYN (synchronous idle)	同步空闲
0001 0111	27	23	0x17	ETB (end of trans. block)	结束传输块
0001 1000	30	24	0x18	CAN (cancel)	取消
0001 1001	31	25	0x19	EM (end of medium)	媒介结束
0001 1010	32	26	0x1A	SUB (substitute)	代替
0001 1011	33	27	0x1B	ESC (escape)	换码(溢出)
0001 1100	34	28	0x1C	FS (file separator)	文件分隔符
0001 1101	35	29	0x1D	GS (group separator)	分组符
0001 1110	36	30	0x1E	RS (record separator)	记录分隔符
0001 1111	37	31	0x1F	US (unit separator)	单元分隔符
0010 0000	40	32	0x20	(space)	空格
0010 0001	41	33	0x21	!	叹号
0010 0010	42	34	0x22	″	双引号
0010 0011	43	35	0x23	#	井号
0010 0100	44	36	0x24	$	美元符
0010 0101	45	37	0x25	%	百分号
0010 0110	46	38	0x26	&	和号
0010 0111	47	39	0x27	’	闭单引号

0010 1000	50	40	0x28	(	开括号
0010 1001	51	41	0x29	)	闭括号
0010 1010	52	42	0x2A	*	星号
0010 1011	53	43	0x2B	+	加号
0010 1100	54	44	0x2C	,	逗号
0010 1101	55	45	0x2D	-	减号/破折号
0010 1110	56	46	0x2E	.	句号
0010 1111	57	47	0x2F	/	斜杠
0011 0000	60	48	0x30	0	字符 0
0011 0001	61	49	0x31	1	字符 1
0011 0010	62	50	0x32	2	字符 2
0011 0011	63	51	0x33	3	字符 3
0011 0100	64	52	0x34	4	字符 4
0011 0101	65	53	0x35	5	字符 5
0011 0110	66	54	0x36	6	字符 6
0011 0111	67	55	0x37	7	字符 7
0011 1000	70	56	0x38	8	字符 8
0011 1001	71	57	0x39	9	字符 9
0011 1010	72	58	0x3A	:	冒号
0011 1011	73	59	0x3B	;	分号
0011 1100	74	60	0x3C	<	小于
0011 1101	75	61	0x3D	=	等号
0011 1110	76	62	0x3E	>	大于
0011 1111	77	63	0x3F	?	问号
0100 0000	100	64	0x40	@	电子邮件符号
0100 0001	101	65	0x41	A	大写字母 A
0100 0010	102	66	0x42	B	大写字母 B
0100 0011	103	67	0x43	C	大写字母 C
0100 0100	104	68	0x44	D	大写字母 D
0100 0101	105	69	0x45	E	大写字母 E
0100 0110	106	70	0x46	F	大写字母 F
0100 0111	107	71	0x47	G	大写字母 G
0100 1000	110	72	0x48	H	大写字母 H
0100 1001	111	73	0x49	I	大写字母 I
1001010	112	74	0x4A	J	大写字母 J
0100 1011	113	75	0x4B	K	大写字母 K
0100 1100	114	76	0x4C	L	大写字母 L
0100 1101	115	77	0x4D	M	大写字母 M
0100 1110	116	78	0x4E	N	大写字母 N
0100 1111	117	79	0x4F	O	大写字母 O
0101 0000	120	80	0x50	P	大写字母 P
0101 0001	121	81	0x51	Q	大写字母 Q
0101 0010	122	82	0x52	R	大写字母 R
0101 0011	123	83	0x53	S	大写字母 S

0101 0100	124	84	0x54	T	大写字母 T
0101 0101	125	85	0x55	U	大写字母 U
0101 0110	126	86	0x56	V	大写字母 V
0101 0111	127	87	0x57	W	大写字母 W
0101 1000	130	88	0x58	X	大写字母 X
0101 1001	131	89	0x59	Y	大写字母 Y
0101 1010	132	90	0x5A	Z	大写字母 Z
0101 1011	133	91	0x5B	[	开方括号
0101 1100	134	92	0x5C	\	反斜杠
0101 1101	135	93	0x5D	]	闭方括号
0101 1110	136	94	0x5E	^	脱字符
0101 1111	137	95	0x5F	_	下划线
0110 0000	140	96	0x60	`	开单引号
0110 0001	141	97	0x61	a	小写字母 a
0110 0010	142	98	0x62	b	小写字母 b
0110 0011	143	99	0x63	c	小写字母 c
0110 0100	144	100	0x64	d	小写字母 d
0110 0101	145	101	0x65	e	小写字母 e
0110 0110	146	102	0x66	f	小写字母 f
0110 0111	147	103	0x67	g	小写字母 g
0110 1000	150	104	0x68	h	小写字母 h
0110 1001	151	105	0x69	i	小写字母 i
0110 1010	152	106	0x6A	j	小写字母 j
0110 1011	153	107	0x6B	k	小写字母 k
0110 1100	154	108	0x6C	l	小写字母 l
0110 1101	155	109	0x6D	m	小写字母 m
0110 1110	156	110	0x6E	n	小写字母 n
0110 1111	157	111	0x6F	o	小写字母 o
0111 0000	160	112	0x70	p	小写字母 p
0111 0001	161	113	0x71	q	小写字母 q
0111 0010	162	114	0x72	r	小写字母 r
0111 0011	163	115	0x73	s	小写字母 s
0111 0100	164	116	0x74	t	小写字母 t
0111 0101	165	117	0x75	u	小写字母 u
0111 0110	166	118	0x76	v	小写字母 v
0111 0111	167	119	0x77	w	小写字母 w
0111 1000	170	120	0x78	x	小写字母 x
0111 1001	171	121	0x79	y	小写字母 y
0111 1010	172	122	0x7A	z	小写字母 z
0111 1011	173	123	0x7B	{	开花括号
0111 1100	174	124	0x7C	|	垂线
0111 1101	175	125	0x7D	}	闭花括号
0111 1110	176	126	0x7E	~	波浪号
0111 1111	177	127	0x7F	DEL (delete)	删除

好团队·好产品·助力智能制造

福州富昌维控电子科技有限公司
FUZHOU FUCHANG WECON E AND T LTD
网站：www.we-con.com.cn
Email：info@we-con.com.cn
电话：400-799-8189
传真：0591-87843899
福建省福州市软件园E区10号楼（生产总部）
福建省福州市软件园F区6号楼5-6层（管理和研发中心）

培训视频　公众号　抖音