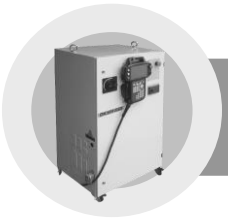
현대중공업

**警告**

应该由合格的安装人员进行安装，并且安装要符合所有国家法规和地方法规

Hyundai 机器人

Hi5EP130301FMCN3



# Hi5 控制器 功能说名书

## 内藏 PLC

現代重工業

現代重工業

本手册内的信息为 HHI 所有。
未经 HHI 书面授权，不得复制全部或部分内容。
本手册不得提供给第三方，不得用于其它用途。

HHI 保留不经过事先通知而修改本手册的权利。

韩国语印刷 – 2013 年 3 月，第 3 版
Hyundai Heavy Industries Co., Ltd. 版权所有© 2013

地址:北京市丰台区卢沟桥南里 2 号
电话:010-83212588
传真:010-83212188
电子邮箱:robot_as@yahoo.com.cn
主页: http://www.hyundai-bj.com

現代重工業

# 目 录


1. 简要 ..... 1-1

2. 控制器设置 ..... 2-1

2.1. 内藏 PLC 的模式设置 ..... 2-2
2.2. 在控制器 TP 中监测 Relay 状态 ..... 2-4
2.3. 内藏 Scan Time ..... 2-4
2.4. 内藏 PLC 启动分配时间调整 ..... 2-4

3. 输入出 Diagram ..... 3-1

4. Relay 规格 ..... 4-1

4.1. Relay 分数 ..... 4-2
4.2. Relay 说明 ..... 4-4
4.3. 计时器&计数器 Relay ..... 4-14

5. 命令语说明 ..... 5-1

5.1. 命令语表 ..... 5-2
5.2. 可使用的 Operands ..... 5-5
5.3. Ladder Diagram 与 Mnemonic 的关系 ..... 5-8

6. 命令语详细说明 ..... 6-1

6.1. XIC(Examine if Closed): Close 检查 ..... 6-2
6.2. XIO(Examine if Open): Open 检查 ..... 6-2
6.3. INV(Inverting): 反转 ..... 6-3
6.4. EQU(Equal): 检查是否相同 ..... 6-3
6.5. NEQ(Not Equal): 检查是否不同 ..... 6-4
6.6. LES(Less Than): 检查是否小于基准值 ..... 6-4
6.7. GRT(Greater Than): 检查是否大于基准值 ..... 6-5
6.8. LEQ(Less Than or Equal): 检查是否比基准值小或相同 ..... 6-5
6.9. GEQ(Greater Than or Equal): 检查是否比基准值大或相同 ..... 6-6
6.10. OTE(Output Energize): 一般输出 ..... 6-6
6.11. OTL(Output Latch): Latch 输出 ..... 6-7
6.12. OTU(Output Unlatch): Unlatch 输出 ..... 6-7
6.13. OSR(One Shot Rising): One Shot 输出 ..... 6-8
6.14. RES(Reset): 复位 ..... 6-8
6.15. TON(Time On Delay): 计时器 ..... 6-9
6.16. CTD(Count Down): 计数器 ..... 6-9


现代重工業

6.17. ADD(Add): 加 ..... 6-10
6.18. SUB(Subtract): 减 ..... 6-11
6.19. MUL(Multiply): 乘 ..... 6-11
6.20. DIV(Divide): 除 ..... 6-12
6.21. POW(Power): 累乘 ..... 6-12
6.22. TOD(Convert to BCD): 变换为 BCD 值 ..... 6-13
6.23. FRD(Convert form BCD to Integer): 变换为 integer ..... 6-14
6.24. SEG(7'Segment): 变换为 7'Segment 值 ..... 6-15
6.25. MOV(Move): 移动 ..... 6-16
6.26. COP(Copy data): Copy ..... 6-17
6.27. CCOP(Conditional Copy data): 按条件复制 ..... 6-18
6.28. ROT(Rotating Output): 按序输出 ..... 6-19
6.29. FOR(FOR): Block 反复 ..... 6-20
6.30. NEXT(NEXT): Block Next ..... 6-21
6.31. LBL(Label): Label 指定 ..... 6-21
6.32. JMP(Jump): Jump ..... 6-22
6.33. CALL(Call): Sub Ladder 呼叫 ..... 6-23
6.34. END(End): 关闭 Ladder ..... 6-23

現代重工業

# 图纸目录

图 1.1 Hi5 I/O Connection ..... 1-2
图 2.1 内藏 PLC 模式设置 ..... 2-2
图 2.2 内藏 PLC Off 状态 图 2.3 内藏 PLC STOP 状态 ..... 2-2
图 2.4 内藏 PLC 启动时间设置 ..... 2-4
图 3.1 输入出 Diagram ..... 3-2

# 表格目录

表 4-1 Relay 分数 ..... 4-2
表 5-1 命令语表 ..... 5-2
表 5-2 可使用的 Operands ..... 5-5

現代重工業

現代重工業

# 1
# 简要

Hi5 控制器的内藏 PLC 是将常用 PLC 的功能体现到控制器的功能。如下图所示、在连接于控制器的 PC 或笔记本电脑启动 HRLadder 后编制/编辑 Ladder 程序、并下载到控制器或上传到控制器的 Ladder 程序、还可监测在控制器启动的程序状况。关于 HRLadder 请参考另外的功能说明书。

Hi5 控制器的 I/O 连接可通过上位固定板和 Fieldbus 连接、利用 Fieldbus- DeviceNet Master 连接 DeviceNet Slave 模块、并通过 BD580 或 BD58A 连接各种装置构成硬连线系统。如上所述、内藏 PLC 可通过 Ladder Logic 控制已连接的输入出信号。

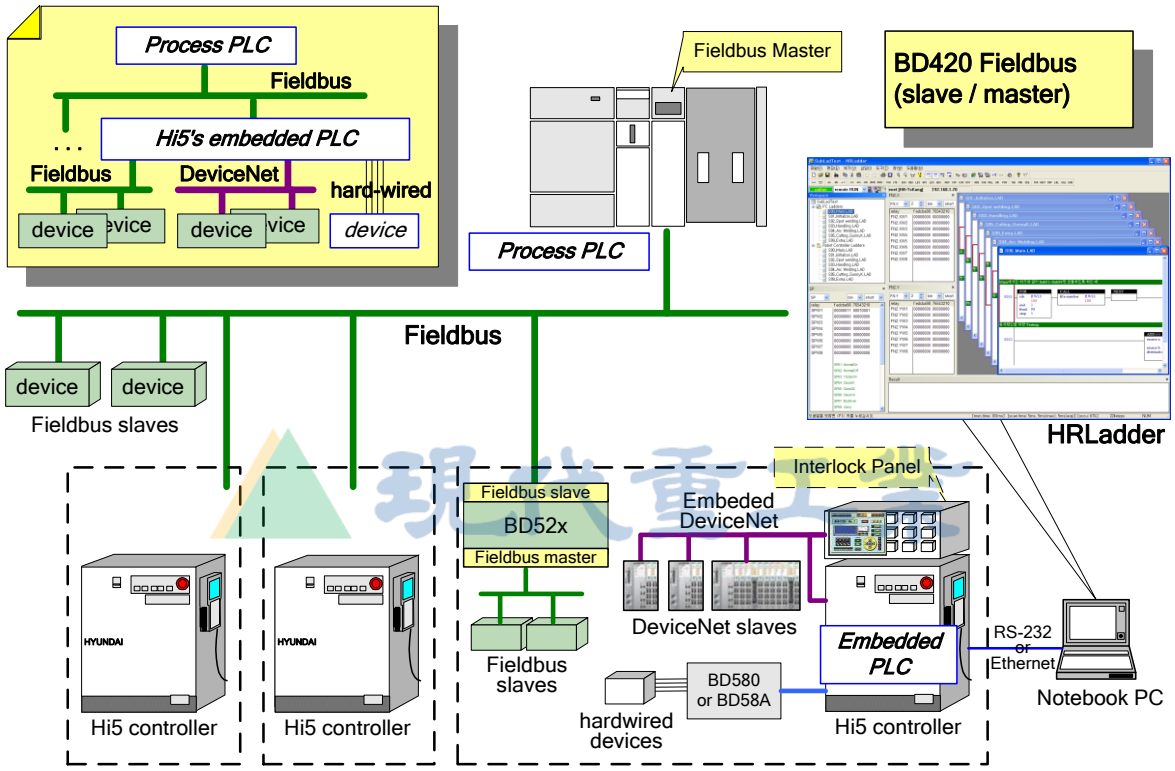



图 1.1 Hi5 I/O Connection

# 2
# 控制器设置

## 2.1. 内藏 PLC 的模式设置

内藏 PLC 的工作模式、可在『[F7]: 条件设置』 → 『[F1]: 应用条件』 → 『5: 内部 PLC 模式』的 <关闭、 停止、 R-停止 、 R-运行 、 运行 >中选一设置。

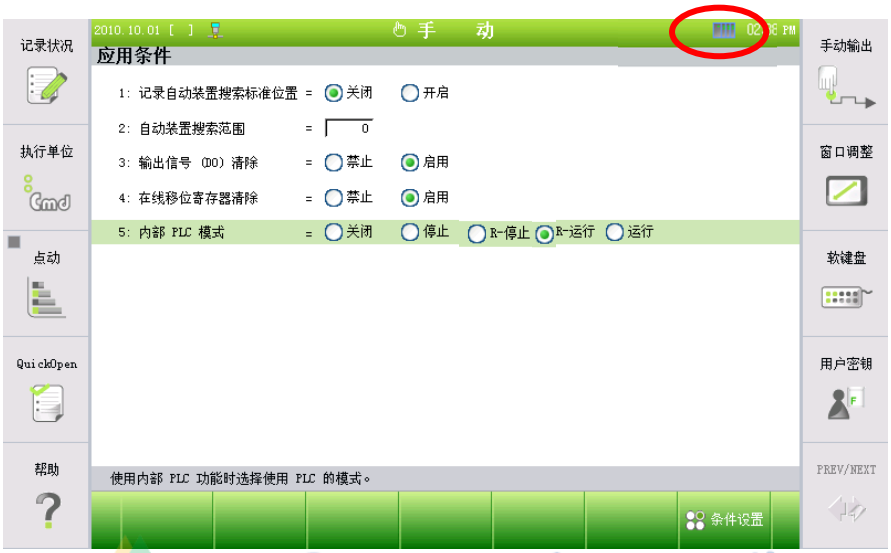



图 2.1 内藏 PLC 模式设置

根据所选模式、在 TP 画面右上方显示图标。如上图所示、如果是[PLC=<R-运行>]或[PLC=<运行>]状态、就会显示 PLC 图标。如下图所示、[PLC=<关闭>]时图标会消失、 [PLC=<停止>]状态时会在 PLC 图显示红色禁止标记。

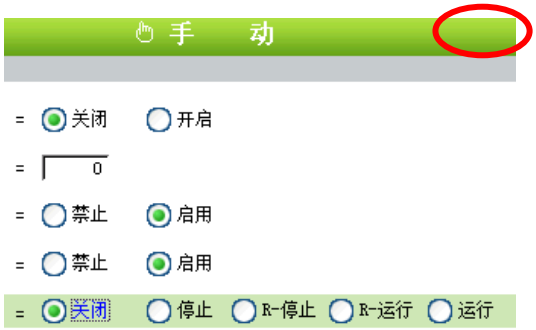



图 2.2 内藏 PLC Off 状态

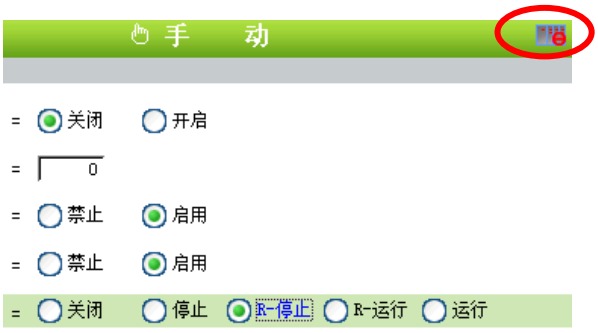



图 2.3 内藏 PLC STOP 状态

- **关闭**
  内藏 PLC 的所有功能都会处于无效状态。这时、机器人控制器的数码输出(Digital Output)-DO1~DO4096 会自动输出物理性输出(Physical Output)-Y1~Y4096、并且物理性输入 X1~X4096 会自动输入 DI1~DI4096。这时各适用 DI、DO、X 和 Y 的属性。
  例如、DO1 为副逻辑、Y1 也属副逻辑时如果用 DO1=1 输出、DO1 会呈副逻辑、0 被输入为 Y1、Y1 的属性会呈副逻辑、因此实际的输出值为 Y1=1。

- **R-停止 / 停止**
  停止内藏 PLC 的工作。R-停止 显示在 HRLadder 可以变更的 Remote 状态。如果设为 STOP、就不能在 HRLadder 变更工作模式。 内藏 PLC 被 Stop 时、PLC 的输出信号 DI、Y、R Relay 会自动变为 0。但、除 PLC 以外的 HRBasic 或分配中、 Y 输出值为“SP11=On”时、Y 输出就会保持之前值。

- **R-运行 / 运行**
  启动内藏 PLC。 R-运行 显示可在 HRLadder 变更的 Remote 状态、如果设为 Run、 就不能在 HRLadder 变更工作模式。

现代重工業

## 2.2. 在控制器 TP 中监测 Relay 状态

可在『[F1]: 服务』 → 『1: 监测』 → 『6: PLC Relay 数据』 → 『1: PLC X 继电器（外部输入）』 → 『11: PLC SW 继电器（系统存储器）』监测 Relay 状态。

## 2.3. 内藏 Scan Time

在 HRLadder 的下方状态栏中显示“scan time”、因 Step 数增加而不能在 5msec 内处理时、scan time 会自动增加(按 5msec 单位)。

## 2.4. 内藏 PLC 启动分配时间调整

在特殊用途中需要加快内藏 PLC Scan 时间时、 请向本公司工程师申请、以便在 Hi5 机器人调整 PLC 所需的分配时间。在手动模式的『[F2]: 系统』 → 『2: 控制参数』 → 『1: 控制环境设置』 → 『7: PLC 执行时间设置』中具有调整分配时间的菜单。 PLC 启动时间设置是指在机器人控制器应处理的作业中设置 PLC 处理时间、默认值为 2.0/20[ms]、如果设置值过大、就不能处理机器人移动所需的动作计算等、必须向本公司工程师提出申请、确认要变更的系统的残余资源后、以此为基础计算分配时间。

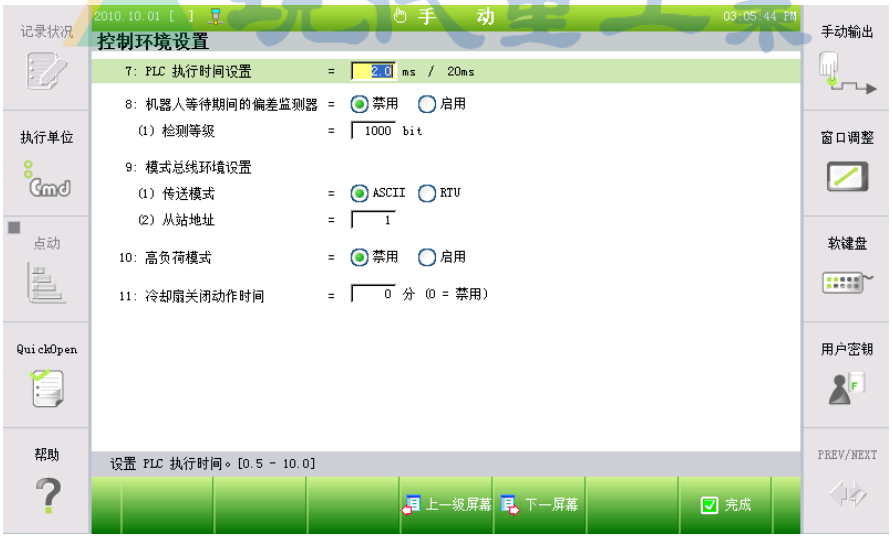



图 2.4 内藏 PLC 启动时间设置

# 3
# 输入出
# Diagram

Hi5 机器人控制器的输入出 Diagram 如下图所示。

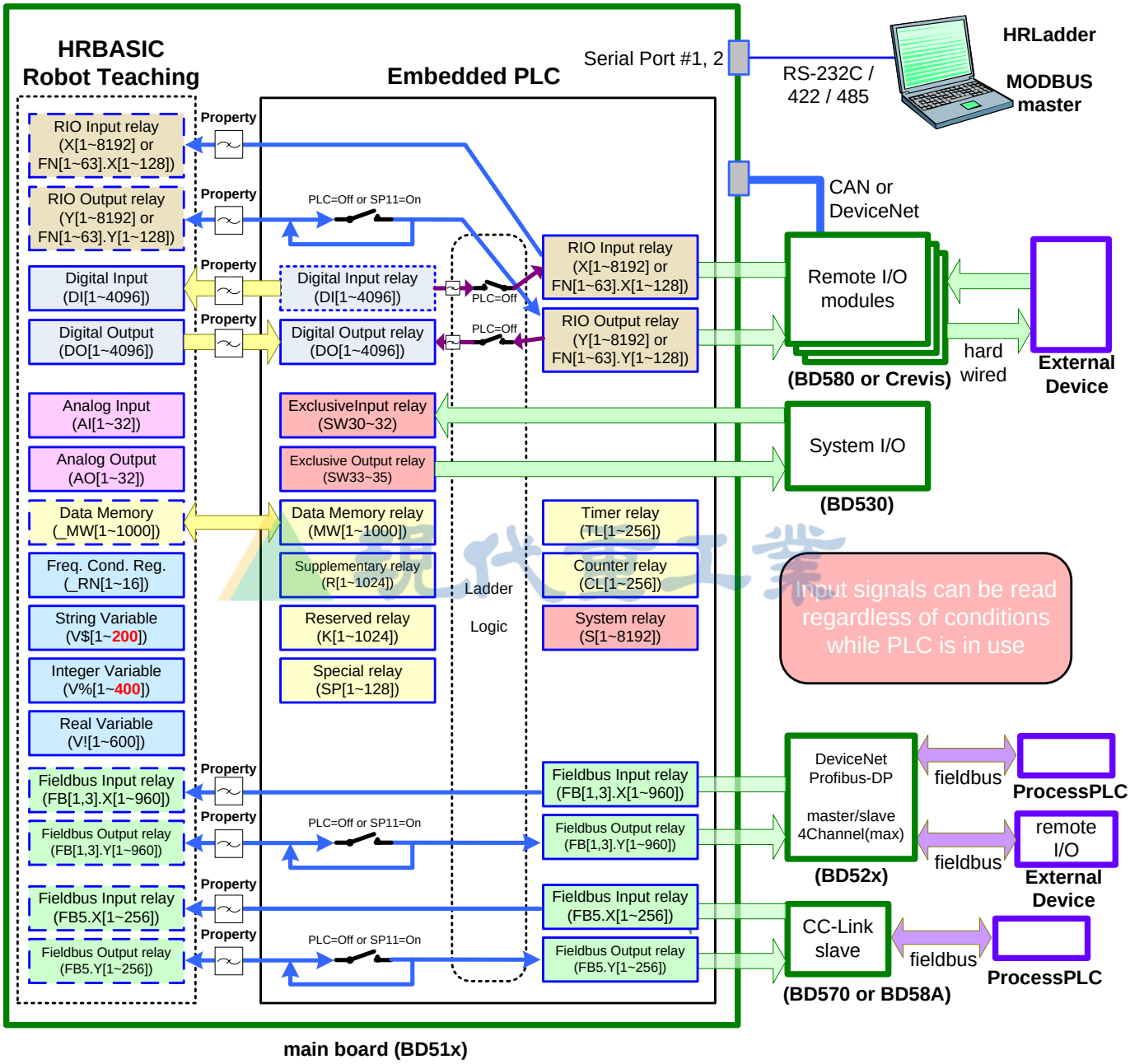



图 3.1 输入出 Diagram

- **上图的系统存储器(System Memory)是为特殊用途预约(reserved)的装置、 今后会根据控制器版本变更用途。**

# 4
# Relay 规格

## 4.1. Relay 分数

表 4-1 Relay 分数

| Relay 名称 | 分数 | Relay(1bit) | Relay(Byte、Word) | (Long、FLoat) |
|---|---|---|---|---|
| 数码输入出 Relay | 输入 4096 分 | DI1~4096 | DIB1~512 | DIL1~128 |
| | | | DIW1~256 | DIF1~128 |
| | 输出 4096 分 | DO1~4096 | DOB1~512 | DOL1~128 |
| | | | DOW1~256 | DOF1~128 |
| RIO 输入出 Relay (BD580/DeviceNet) | 输入 8192 分 每 Node 为 128 分 (Max 64Node) | FN1~64.X1~128 (or X1~8192) | FN1~64.XB1~16 (or XB1~1024) | FN1~64.XL1~4 (or XL1~256) |
| | | | FN1~64.XW1~8 (or XW1~512) | FN1~64.XF1~4 (or XF1~256) |
| | 输出 8192 分 每 Node 为 128 分 (MaX 64Node) | FN1~64.Y1~128 (or Y1~8192) | FN1~64.YB1~16 (or YB1~1024) | FN1~64.YL1~4 (or YL1~256) |
| | | | FN1~64.YW1~8 (or YW1~512) | FN1~64.YF1~4 (or YF1~256) |
| Fieldbus Relay (DeviceNet/ProFiBus-DP) | 输入 960 分 | FB1.X1～960 | FB1.XB1～120 | FB1.XL1～30 |
| | | | FB1.XW1～60 | FB1.XF1～30 |
| | 输出 960 分 | FB1.Y1～960 | FB1.YB1～120 | FB1.YL1～30 |
| | | | FB1.YW1～60 | FB1.YF1～30 |
| | 输入 960 分 | FB3.X1～960 | FB3.XB1～120 | FB3.XL1～30 |
| | | | FB3.XW1～60 | FB3.XF1～30 |
| | 输出 960 分 | FB3.Y1～960 | FB3.YB1～120 | FB3.YL1～30 |
| | | | FB3.YW1～60 | FB3.YF1～30 |
| Fieldbus Relay (CC-Link) | 输入 256 分 | FB5.X1～256 | FB5.XB1～32 | FB5.XL1～8 |
| | | | FB5.XW1～16 | FB5.XF1～8 |
| | 输出 256 分 | FB5.Y1～256 | FB5.YB1～32 | FB5.YL1～8 |

| Relay 名称 | 分数 | Relay(1bit) | Relay(Byte、Word) | (Long、FLoat) |
|---|---|---|---|---|
| | | | FB5.YW1～16 | FB5.YF1～8 |
| 辅助 Relay | 1024 分 | R1~1024 | RB1~128 | RL1~32 |
| | | | RW1~64 | RF1~32 |
| 保存 Relay | 1024 分 | K1~1024 | KB1~128 | KL1~32 |
| | | | KW1~64 | KF1~32 |
| 特殊 Relay | 128 分 | SP1~128 | SPB1~16 | SPL1~4 |
| | | | SPW1~8 | SPF1~4 |
| 计时 Relay | 256 分 | T1~256 | TB1~1024 | TL1~256 |
| | | | TW1~512 | TF1~256 |
| 计数 Relay | 256 分 | C1~256 | CB1~1024 | CL1~256 |
| | | | CW1~512 | CF1~256 |
| 数据存储器 | 16000 分 | M1~16000 | MB1~2000 | ML1~500 |
| | | | MW1~1000 | MF1~500 |
| 系统存储器 | 8192 分 (Reserved) | S1~8192 (Reserved) | SB1~1024 | SL1~256 |
| | | | SW1~512 | SF1~256 |

参考) Relay 形式: Byte=8bit、 Word=16bit、 Long=32bit、 Float=(32bit)

現代重工業

## 4.2. Relay 说明

各 Relay 使用首字母(B/W/L/F)、可以 bit、byte、word、long、float 形式指定。但、计时器和计数器的 bit(T/C)在 long(TL/CL)值为 0 时表示激活状态(activation)、非为 0 时表示非激活状态。

(1) 数码输入出 Relay: 可在 HRBasic 或各种输入出分配中使用的信号。

(2) RIO 输入出 Relay:
BD580 Board 的输入出信号或连接于内藏 Fieldbus 的 Remote I/O Board 的输入出 信号。

(3) Fieldbus Relay:
连接于 BD52XBoard 的 DeviceNet 或 ProfiBus-DP 的输入出信号、或连接于 BD570 或 BD58A 的 CC-Link 的输入出信号。

(4) 辅助 Relay: 在 PLC 程序中使用的辅助 Relay。

(5) 保存 Relay: 即使关闭电源、也会保存 On/Off 状态的 Relay。

(6) 特殊 Relay: 为特殊目的而定义的 Relay

| Relay 编号 | 说明 | 其他 |
|---|---|---|
| **SP01** | 时常 On Relay | Controller states |
| **SP02** | 时常 Off Relay | |
| **SP03** | 在开始运行后的第一个 Scan 时启动的 Relay | Internal timer |
| **SP04** | 0.1 秒 clock (0.05 秒 On → 0.05 秒 Off) | |
| **SP05** | 0.2 秒 clock (0.1 秒 On → 0.1 秒 Off) | |
| **SP06** | 1 秒 clock (0.5 秒 On → 0.5 秒 Off) | |
| **SP07** | 不能在 BCD 计算变换时启动。. | 启动 TOD 或 FRD 时 |
| **SP08** | 计算结果有 carry 时 On. | 执行算数命令时 |
| **SP09** | 2 秒 clock (1 秒 On → 1 秒 Off) | Internal timer |
| **SP10** | 4 秒 clock (2 秒 On → 2 秒 Off) | |
| **SP11** | 如果是 On、就在除 PLC 以外的应用(HRBasic、分配)中允许直接向 YRelay 输出。 | |
| **SP12** | Label 非为常数时 On | |

| Relay 编号 | 说明 | 其他 |
|---|---|---|
| **SP13** | Label 的数量为 100 以上时 On | |
| **SP14** | Label 重复时 On | |
| **SP15** | Label 为 0 以下或没有可 Jump 的 Label 编号时 | |
| **SP16** | 在 Simulation 状态中进行 Modbus Test | |
| **SP17** | Scan Time 超过 5 秒钟时 On<br>(这时处于 5 秒之后的命令不会被启动。 | |
| **SP18** | 没有通过 Call 呼叫的 Subladder 时 On | |
| **SP19~SP128** | Reserved | |

(7) 计时器 Relay:
进行计时器工作的 Relay、 设置值为 0 时接点会 On。 (停电时保存)

(8) 计数器 Relay:
进行计数器工作的 Relay、 设置值为 0 时接点会 On。(停电时保存)

(9) 数据存储器:
保存或读取应用命令的任一数据时使用、在 HRBasic 也可 access、与 HABasic 收发大量数据时可以使用。 (停电时保存)

(10) 系统存储器(Reserved): 使用于以下特殊用途、其用途可在未经通报的情况下根据具体情况而变更。

| No. | 说明 | 其它 |
|---|---|---|
| | | |
| **SW3** | PLC 启动模式(4: PLC OFF、 5: 没有程序、 0:STOP、 1:R.STOP、2:R.RUN、3:RUN) | |
| **SW4** | 扩张 IO Board 数 | |
| **SW5** | Main SW Version 的 2nd + 1st | 20.03-10v→ &H03 14 |
| **SW6** | I/O version + Main SW Version 的 3rd | 20.03-10 → &H43 0A |
| | | |
| **SW10** | Scan time | |

| No. | 说明 | 其它 |
|---|---|---|
| **SW11** | 分配时间 | |
| **SW12** | 最大占有时间 | |
| **SW13** | 平均占有时间 | |
| **SW14** | Ladder 的总 Step 数 | |
| **SW15** | Ladder 程序的 CHECKSUM | Ver20.03-14 以后 |
| | | |
| **SW20** | 控制器的当前程序编号 | |
| **SW21** | 控制器的当前 Step 编号 | |
| **SW22** | 控制器的当前功能编号 | |
| **SW23** | 控制器的主程序编号 | Ver20.03-14 以后 |
| **SW24** | FB1~FB4 active | 0th~3rd bit |
| **SW25** | 读取 Hilscher 状态信息(1~4)<br>(1:FB1 信息、2: FB2 信息…) | (since mv20.03-10) |
| **SW26** | LSB: Hilsche COM GlobalBits<br>MSB: reserved | (只有 Master 有效) |
| **SW27** | LSB: 错误 Node 编号<br>MSB: 错误代码 | (只有 Master 有效) |
| **SW28** | Bus 错误计数器 | (只有 Master 有效) |
| **SW29** | DeviceNet: Bus OFF 计数器<br>Profibus-DP: Time Out 计数器 | (只有 Master 有效) |
| **SW30** | 专用输入 1 | |
| **SW31** | 专用输入 2 | |
| | | |
| **SW33** | 专用输出 1 | |
| **SW34** | 专用输出 2 | |

| No. | 说明 | 其它 |
|---|---|---|
| | | |
| **SW37** | LSB: 对指定的 node 编号、每一秒邀请 slave_diag | |
| **SW38** | 对 slave_diag 邀请的回复值 | |
| **SW39** | slave_diag 邀请结果的 Station Status<br>LSB: StationStatus1<br>MSB: StationStatus2 | |
| **SW40** | 读取机器人位置 (0:无、1:当前 base 坐标值、2:指令 base 坐标值、5:当前轴值、6:指令轴值 ) | |
| **SW41** | base 坐标值: X 值<br>轴坐标值: 1 轴值 | 1mm 单位<br>1mm or 0.1deg 单位 |
| **SW42** | base 坐标值: Y 值<br>轴坐标值: 2 轴值 | 1mm 单位<br>1mm or 0.1deg 单位 |
| **SW43** | base 坐标值: Z 值<br>轴坐标值: 3 轴值 | 1mm 单位<br>1mm or 0.1deg 单位 |
| **SW44** | base 坐标值: RX 值<br>轴坐标值: 4 轴值 | 0.1 deg 单位<br>1mm or 0.1deg 单位 |
| **SW45** | base 坐标值: RY 值<br>轴坐标值: 5 轴值 | 0.1 deg 单位<br>1mm or 0.1deg 单位 |
| **SW46** | base 坐标值: RZ 值<br>轴坐标值: 6 轴值 | 0.1 deg 单位<br>1mm or 0.1deg 单位 |
| **SW47~52** | 附加轴位置(T1~T6 轴数) | 0.1 deg or 1mm 单位<br>version 32.04-01 以后 |
| | | |
| **SW60~69** | 指定间接地址地领域 | -1:SW61、-2:SW62、 |
| | | |
| **SW70** | 驱动时间类型<br>(0:无、1:驱动时间(累计)、2:总驱动时间 0)<br>注意) SW(n):MSW(上位)、SW(n+1):LSW | |
| **SW71~SW72** | 检测时间(10msec 单位) | |
| **SW73~SW74** | Cycle 时间(10msec 单位) | |
| **SW75** | Cycle 数 | |
| **SW76~SW77** | 焊接机 1 焊接时间(10msec 单位) | |

| No. | 说明 | 其它 |
|---|---|---|
| **SW78** | 焊接机 1 焊接次数 | |
| **SW79~SW80** | 焊接机 2 焊接时间(10msec 单位) | |
| **SW81** | 焊接机 2 焊接次数 | |
| **SW82~SW83** | 焊接机 3 焊接时间(10msec 单位) | |
| **SW84** | 焊接机 3 焊接次数 | |
| **SW85~SW86** | 焊接机 4 焊接时间(10msec 单位) | |
| **SW87** | 焊接机 4 焊接次数 | |
| **SW88~SW89** | Wait、DI 等待时间(10msec 单位) | |
| **SW90~SW91** | 计时器等待时间(10msec 单位) | |
| **SW70** | 驱动时间类型<br>(0:无、3:最近驱动时间、4:总驱动时间)<br>添加最近驱动时间、Swapping SW、以便使用 SL。<br>即、SW(奇数):LSW(下位)、SW(偶数):MSW | Ver. 30.18.00 以后 |
| **SL36** | 检测时间(10msec 单位) | |
| **SL37** | Cycle 时间(10msec 单位) | |
| **SL38** | Cycle 数 | |
| **SL39** | 焊接机 1 焊接时间(10msec 单位) | |
| **SL40** | 焊接机 1 焊接次数 | |
| **SL41** | 焊接机 2 焊接时间(10msec 单位) | |
| **SL42** | 焊接机 2 焊接次数 | |
| **SL43** | 焊接机 3 焊接时间(10msec 单位) | |
| **SL44** | 焊接机 3 焊接次数 | |
| **SL45** | 焊接机 4 焊接时间(10msec 单位) | |

| No. | 说明 | 其它 |
|---|---|---|
| **SL46** | 焊接机 4 焊接次数 | |
| **SL47** | Wait、DI 等待时间(10msec 单位) | |
| **SL48** | 计时器 等待时间(10msec 单位) | 10msec 单位 |
| **SW70** | 运行时间种类<br>(5:运行信息(电弧))<br>添加最近运行时间、同时为了能够使用 SL、对 SW 进行 Swapping。<br>即 SW(奇数):LSW(下位)、SW(偶数):MSW | |
| **SL36** | 测试时间(10msec 单位) | |
| **SL37** | 循环时间(10msec 单位) | |
| **SL38** | 循环次数 | |
| **SL39** | 电弧焊接进行时间(10msec 单位)、以 1 天的单位设置 | |
| **SL40** | 电弧焊接进行日期(1 天单位) | |
| **SW81** | ARC Retry 进行次数 | |
| **SW82** | ARC Overlap 进行次数. 1:Arc Off | |
| **SW83** | ARC Overlap 进行次数. 2:Wire Off | |
| **SW84** | ARC Overlap 进行次数. 3:Limit Over | |
| **SW85** | ARC Overlap 进行次数. 4:Gas Off | |
| **SW86** | ARC Overlap 进行次数. 5:Coolant Off | |
| **SW87** | ARC 自动熔敷清除 进行次数 | |
| **SL45** | 1 Cycle 期间电弧运行时间 | |
| **SL46** | Tip 使用时间 | |
| **SL47** | Tip 使用时间限制值 | |
| **SW100~SW109** | Program Count 相关 reserved | |

| No. | 说明 | 其它 |
|---|---|---|
| | | |
| **SW110** | 各轴速度 Enable(在 SW110=0xKLM 中)<br>N: 分解能(1=10 倍、2=100 倍、3=1000 倍)<br>M: 其他选项( b4:1=绝对值、 b5:马达速度) | |
| **SW111~SW126** | 各轴(或马达)的速度（16 轴) | |
| **SW130~SW133** | TP KEYPAD 信息 | |
| **SW135** | Get Applet | |
| **SW136** | Set Applet | |
| | | |
| **SW140** | FN Error BitMap1 (Node00~Node15) | 1:Error |
| **SW141** | FN Error BitMap2 (Node16~Node31) | |
| **SW142** | FN Error BitMap3 (Node32~Node47) | |
| **SW143** | FN Error BitMap4 (Node48~Node63) | |
| **SW144** | FN Master State[*1)] | |
| **SW145** | 邀请 FN 错误信息所需的 Node 编号 | |

[*1)] Master State
0x8000(b15): 1(Used)
0x0100(b8): 1(Checking状态：初始化 & Rescan后10秒钟，脱离Node设置画面后2秒)
0x001F(b0~b4):
Master: b4: 1(RUN), 0(IDLE(PLC=STOP))
b0~b3: 0(RUNNING), 1(ResetOutOfBox), 2(InitOutOfBox), 3(ResetNormal), 4(InitNormal), 5(DupMacCheck), 6(NRFault)

| No. | 说明 | 其它 |
|---|---|---|
| **SW146** | 上述应答的 Node State*2) | |
| | | |
| **SW150** | Arc Welding Info | 1=Enable(Digital Only) |
| **SW151** | Arc Welding Real Current | |
| **SW152** | Arc Welding Real Voltage | |
| **SW153** | Arc Welder Error1 | |
| **SW154** | Arc Welder Error2 | |
| | | |
| **SW160** | 0x0001:Request Single<br>0x001X:进行中、0x0000:完成<br>其他: 显示错误状态<br>0x002X: TimeOut Error | 结果 =><br>SW166 ~ SW179 |
| **SW161** | EM Req: NodeID | |
| **SW162** | EM Req: Service | |
| **SW163** | EM Req: Class | |
| **SW164** | EM Req: Instance | |
| **SW165** | EM Req: Attribute | |
| **SW166** | EM Req/Rsp: Size of Data | Max. 26 |

*2) Node State
0x8000(b15): 1(Used Node)
0x4000(b14): 1(Unguaranteed Maker)
0x3000(b12~b13): NodeStatus(1:Standby,2:ConnectionFault,3:ConfigFault)
0x0800(b11): 1(Not IoMode)
0x03FF(b0~b9):
Slave: (Bit-field)
0x000(Online), 0x001(Offline), 0x002(IO Close), 0x004(InSizeErr),
0x008(OutSizeErr), 0x010(VenderMismatch), 0x020(TypeMismatch),0x040(CodeMismatch),
0x080(CcvMismatch), 0x100(IoTooBig), 0x200(NoConnection)

| No. | 说明 | 其它 |
|---|---|---|
| **SW167~SW179** | EM Req/Rsp: Data(SB333~SB358) | Max. 26 bytes |
| | | |
| **SW180** | 获取 SPOT GUN 之磨损量所需的 GUN 编号<br>(0: 无效、1~: GUN 编号) | mv20.04-77 ~<br>mv30.32.33 ~ |
| **SW181** | 移动电极磨损量 | |
| **SW182** | 固定电极磨损量 | |
| **SW183** | 枪搜索状态 | |
| **~SW189** | SPOT GUN reserved | |
| **SW190** | MON_AXIS_CTRL_OFF | 每一个 Bit 为 1 轴 |
| **SW195** | 滚边加压力[N] | |
| | | |
| **SW200** | 轴别累计距离监测模式<br>(支持 dword 是 V31.11-00~、V32.01-00~) | 0:Off、<br>1:read、2:write(word)<br>3:read、4:write(dword) |
| **SW201~SW216** | 轴别累计距离监测 (1 轴~16 轴)<br>dword 的话 SL101~SL108 (1~8 轴) | 单位: km、rad<br>发生 Overflow、就都会成为 0。 |
| **SW220** | MON_SYSTEM_VAR_IDX slot1<br>(0: 功能 Off、10: _SPDRATE(单位: %)<br>、301~316: AI[1~16] (单位: 0.01V)<br>、351~366: AO[1~16]) | [_IDX]<br>0-base: 读取值<br>30000-base: 设定值<br><br>如)<br>在 SW223 设置 352 的话、<br>以 SW230 的值<br>设置_SPDRATE 值<br><br>slot5~: mv31.08-00~<br>slot2~: mv31.01-02~ |
| **SW221** | 系统变数下位 Word | |
| **SW222** | 系统变数上位 Word | |
| **SW223~** | 系统变数 slot2 | |
| **SW226~** | 系统变数 slot3 | |
| **SW229~** | 系统变数 slot4 | |
| **SW232~** | 系统变数 slot5 | |
| **SW235~** | 系统变数 slot6 | |

| No. | 说明 | 其它 |
|---|---|---|
| SW238~ | 系统变数 slot7 | |
| SW241~ | 系统变数 slot8 | |
| SW244~ | 系统变数 slot9 | |
| SW247~ | 系统变数 slot10 | |
| SW250 | (子任务 1) 生成状态 | |
| SW251 | (子任务 1) 当前 程序号码 | |
| SW252 | (子任务 1)当前步骤号码 | |
| SW253 | (子任务 1)当前功能号码 | |
| ~SW259 | 为(子任务 1)进行 reserved | |
| SW260 | (子任务 2)生成状态 | |
| SW261 | (子任务 2)当前程序号码 | |
| SW262 | (子任务 2)当前步骤号码 | |
| SW263 | (子任务 2)当前功能号码 | |
| ~SW269 | 为(子任务 2)进行 reserved | |
| SW270 | (子任务 3)生成状态 | |
| SW271 | (子任务 3)当前程序号码 | |
| SW272 | (子任务 3)当前步骤号码 | |
| SW273 | (子任务 3)当前功能号码 | |
| ~SW279 | 为(子任务 3)进行 reserved | |

現代重工業

## 4.3. 计时器&计数器 Relay

(1) 计时器和计数器 Relay 只支持 down-counting。

- 用户可按 10msec 单位设置计时器 Base。
- 计时器值在内部以 32bit 值处理、可以计时到 2、147、483、647[msec](约 597 小时)。

(2) 计时器 / 计数器值具有以下意义。

| 计时器 & 计数器值 | 说明 |
|---|---|
| 0 | 接点 On (=counting 完毕) |
| -1 | 接点 Off |
| 其外 | 接点 Off; timing & counting (进行中) |

(3) 连接计时器 / 计数器 Relay 的 Rung 处于非激活状态、

- TON: TL(计时器)值就会成为-1。
- CTD: CL(计数器)值就会继续保持。

(4) 连接计时器 / 计数器 Relay 的 Rung 处于激活状态、

- **TON**
  如果 TL 值小于 0、TL 的起初值会保存为“计时器 Base x Freeset x 10”、TL 值大于 0、每 5msec 会减少 5。

- **CTD**
  如果 CL 值小于 0、CL 的起初值会成为 Freeset 值、如果 CL 值大于 0、每次从非激活状态变为激活状态时其数值就会减少 1。

# 5
# 命令语说明

现代重工業

## 5.1. 命令语表

表 5-1 命令语表

| 命令语 | | | | 说明 |
|---|---|---|---|---|
| 编号 | Mnemonic | 名称 | 标记 | |
| | RUNG | Rung | ├──┤ | 显示 rung。 |
| | BST | Branch Start | ┬── | Branch 的开始 |
| | BND | Branch End | ──┬ | Branch 的结束 |
| | NXB | Nested Branch | └、├ | Branch 的重复 |
| Logic 检查命令 | | | | 检查结果、 是表示 Rung 被激活/(否:非激活) |
| 1 | XIC | Examine if Closed | -\| \|- | 检查接点是否关闭(A 接点) |
| 2 | XIO | Examine if Open | -\|/\|- | 检查接点是否打开(B 接点) |
| 3 | INV | Inverting | -//- | Rung 的结果反转(inverting) |
| 4 | EQU | Equal | -目- | 检查是否相同(=) |
| 5 | NEQ | Not Equal | -目- | 检查是否不同(<>) |
| 6 | LES | Less Than | -目- | 检查是否小于基准值(<) |
| 7 | GRT | Greater Than | -目- | 检查是否大于基准值(>) |
| 8 | LEQ | Less Than or Equal | -目- | 检查是否比基准值(<=)小或相同。 |
| 9 | GEQ | Greater Than or Equal | -目- | 检查是否比基准值(>=)大或相同。 |
| 输出命令 | | | | |
| 10 | OTE | Output Energize | -( )- | 输出 Rung 的状态(激活:ON/非激活:OFF) |
| 11 | OTL | Output Latch | -(L)- | Rung 处于激活状态、 就输出为 ON(high) |
| 12 | OTU | Output Unlatch | -(U)- | Rung 处于激活状态、 就输出为 OFF(low) |

| 命令语 | | | | 说明 |
|---|---|---|---|---|
| 编号 | Mnemonic | 名称 | 标记 | |
| 13 | OSR | One Shot Rising | -(OSR)- | Rung 处于激活状态、 只在 scan 期间输出 ON |
| 14 | RES | Reset | -(RES)- | Rung 处于激活状态、 对计时器或计数器进行复位 |
| 计时器及计数器命令 | | | | |
| 15 | TON | Time On Delay | -[]- | Rung 处于激活状态时计时器会工作 |
| 16 | CTD | Count Down | -[]- | Down Count Rung 的激活时间(非激活→激活) |
| 计算命令 | | | | |
| 17 | ADD | Add | -[]- | Rung 处于激活状态、 (+)计算 |
| 18 | SUB | Subtract | -[]- | Rung 处于激活状态、 (-)计算 |
| 19 | MUL | Multiply | -[]- | Rung 处于激活状态、 (x)计算 |
| 20 | DIV | Divide | -[]- | Rung 处于激活状态、 (/)计算 |
| 21 | POW | Power | -[]- | Rung 处于激活状态、 (^: 累乘)计算 |
| 数据变换命令 | | | | |
| 22 | TOD | Convert int. to BCD | -[]- | Rung 处于激活状态、 变换为 BCD |
| 23 | FRD | Convert form BCD to int. | -[]- | Rung 处于激活状态、 变换为 integer |
| 24 | SEG | 7'Segment | -[]- | Rung 处于激活状态、 变换为 7’ Segment 值 |
| 移动及复制命令 | | | | |
| 25 | MOV | Move | -[]- | Rung 处于激活状态、 复制一个数据 |
| 26 | COP | Copy data | -[]- | Rung 处于激活状态、 复制多个数据 |
| 27 | CCOP | Conditional Copy data | -[]- | 根据 Rung 状态复制多个数据 |
| 应用命令 | | | | |

| 命令语 | | | | 说明 |
|---|---|---|---|---|
| 编号 | Mnemonic | 名称 | 标记 | |
| 28 | ROT | Rotating Output | -[]- | Rung 处于激活状态、按序输出 |
| Block 控制命令 | | | | |
| 29 | FOR | For | -[]- | Rung 处于激活状态、到 NEXT 反复执行 |
| 30 | NEXT | Next | -[]- | 如果是反复次数以内、就用 FOR 文 JUMP |
| 31 | LBL | Label | -[]- | 用 Jump 命令指定要 Jump 的位置 |
| 32 | JMP | Jump | -[]- | Rung 处于激活状态、就向 LabelJump (Label<0 时、 就会越过-n 个 Next) |
| 33 | CALL | Call | -[]- | Rung 处于激活状态、就呼叫 Sub-ladder |
| 34 | END | End | -[]- | Rung 处于激活状态、 Sub-ladder end |

現代重工業

## 5.2. 可使用的 Operands

表 5-2 可使用的 Operands

| inst | Relay Type / arg | Input X,DO n | Input X,DO (B,W,L,F)m | Output Y,DI,R,K,SP n | Output Y,DI,R,K,SP (B,W,L,F)m | Timer T n | Timer T (B,W,L,F)m | Counter C n | Counter C (B,W,L,F)m | Memory M,S n | Memory M,S (B,W,L,F)m | Const (32bit) (L/F) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| XIC | | | x | | x | | x | | x | | x | x |
| XIO | | | x | | x | | x | | x | | x | x |
| EQU | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| LES | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| GRT | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| LEQ | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| GEQ | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| OTF | | x | x | | x | x | x | x | x | x | x | x |
| OTL | | x | x | | x | x | x | x | x | x | x | x |
| OUT | | x | x | | x | x | x | x | x | x | x | x |
| OSR | | x | x | | x | x | x | x | x | x | x | x |
| RES | | x | x | x | x | | x | | x | x | x | x |
| TON | tm | x | x | x | x | | x | x | x | x | x | x |
| | bs | x | | x | | x | | x | | x | | |
| | pst | x | | x | | x | | x | | x | | |
| CTD | cnt | x | x | x | x | x | x | x | x | x | x | x |
| | pst | x | | x | | x | | x | | x | | |
| ADD | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| | dst | x | x | x | | x | | x | | x | | x |
| SUB | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| | dst | x | x | x | | x | | x | | x | | x |
| MUL | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |

現代重工業

| inst | Relay / Type / arg | Input X,DO n | Input X,DO (B,W,L,F)m | Output Y,DI,R,K,SP n | Output Y,DI,R,K,SP (B,W,L,F)m | Timer T n | Timer T (B,W,L,F)m | Counter C n | Counter C (B,W,L,F)m | Memory M,S n | Memory M,S (B,W,L,F)m | Const (32bit) (L/F) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | dst | x | x | x | | x | | x | | x | | x |
| DIV | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| | dst | x | x | x | | x | | x | | x | | x |
| POW | sA | x | | x | | x | | x | | x | | |
| | sB | x | | x | | x | | x | | x | | |
| | dst | x | x | x | | x | | x | | x | | x |
| TOD | src | x | u | x | u | x | u | x | u | x | u | u |
| | dst | x | x | x | u | x | x | x | x | x | u | x |
| FRD | src | x | u | x | u | x | u | x | u | x | u | u |
| | dst | x | x | x | u | x | x | x | x | x | u | x |
| SEG | src | x | u | x | u | x | u | x | u | x | u | u |
| | dst | x | x | x | u | x | x | x | x | x | u | x |
| MOV | src | x | | x | | x | | x | | x | | |
| | dst | x | x | x | | x | | x | | x | | x |
| COP | src | | | | | x | | x | | | | |
| | dst | x | x | | | x | | x | | | | x |
| | len | x | | x | | x | | x | | x | | |
| CCOP | sA | | | | | x | | x | | | | |
| | sB | | | | | x | | x | | | | |
| | dst | x | x | | | x | | x | | | | x |
| | len | x | | x | | x | | x | | x | | |
| ROT | st | x | | x | | x | | x | | x | | |
| | cnt | x | | x | x | x | | x | | x | | |
| | tm | x | x | x | x | | x | x | x | x | x | x |
| | rep | x | | x | | x | | x | | x | | |
| | out | x | x | x | | x | x | x | x | x | | x |
| | rst | | x | | x | | x | | x | x | x | x |
| | tmp | x | x | x | | x | x | x | x | x | | x |
| FOR | idx | x | x | x | | x | x | x | x | x | | x |
| | init | x | | x | | x | | x | | x | | |
| | final | x | | x | | x | | x | | x | | |
| | step | x | | x | | x | | x | | x | | |

現代重工業

| inst | Relay / Type / arg | Input X,DO n | Input X,DO (B,W,L,F)m | Output Y,DI,R,K,SP n | Output Y,DI,R,K,SP (B,W,L,F)m | Timer T n | Timer T (B,W,L,F)m | Counter C n | Counter C (B,W,L,F)m | Memory M,S n | Memory M,S (B,W,L,F)m | Const (32bit) (L/F) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LRI | label | x | x | x | x | x | x | x | x | x | x | |
| JMP | label | x | | x | | x | | x | | x | | |
| CALL | S file | x | | x | | x | | x | | x | | |

(1) 表中‘x’标记表示不能使用。

(2) Relay(relay)的 n 表示 bit 编号、(B、W、L、F)m 是区分 Byte(8bit)、 Word(16bit)、Long(32bit)、Float 形式的字母、 m 表示各形式的数据编号。(例、X1、XB2、XW2、XL2、XF2)

(3) 表中的‘u’标记显示以没有符号的(unsigned) 值处理。

(4) 因此、输入命令语的时可通过选择 Relay 形式(B、W、L、F)来选择数据尺寸(data size)。

(5) 但、在 COP(复制)和 CCOP(按条件复制) 命令中、如果 src 不是常数、src 与 dst 的 Relay 形式应相同。

(6) 输入 Bit 值时可以 Bite(B)或 Word(W)、或 Long Word(L)的相对值输入。即、例如 X34=XB5/2=XW3/2=XL2/2、可以多种形式输入相同的 bit 编号。

(7) 常数(constant)
常数的数据尺寸为 32bit 数据形式具有 2 进数、10 进数、16 进数、以&B 开始、就表示 2 进数；以&H 开始就表示 16 进数；没有词头的数字则表示 10 进数。因此、125 可以&H7D 或&B01111101 等形式输入。

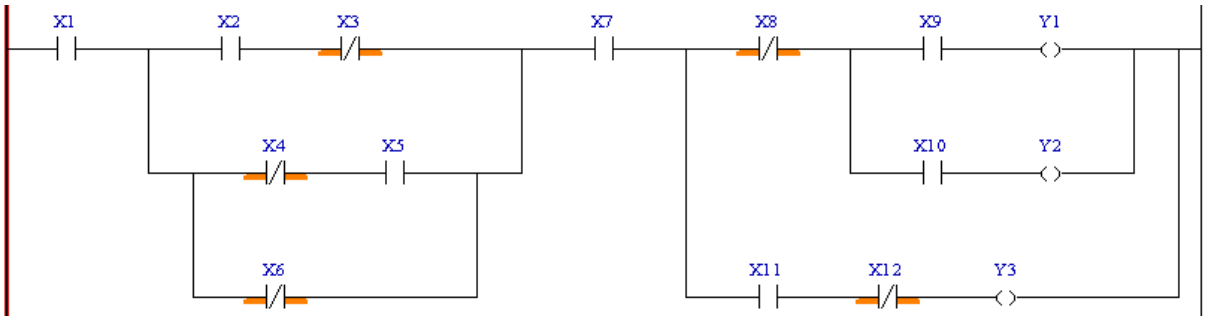
(8) 在表格中、没有‘u’标记的所有 Relay 的(B、W、L、F)都以有符号的(signed) 值处理。
例如、在 MOV 命令中、把 B(8bit)值移动到 W(16bit)时、signed bit 会扩张、应注意。即、如果把 RB(-1=&HFF)移动到 RW、就会成为 0xFFFF(-1)、而不是&H00FF(255)。
但、与 BCD 数据有关的 TOD 和 FRD 命令、变换为了 7’Segment 数据的 SEG 命令会以没有符号的值(unsigned)处理。

(9) CALL/END/LBL/JMP 命令从 mv30.50-00 之后版本开始支持。

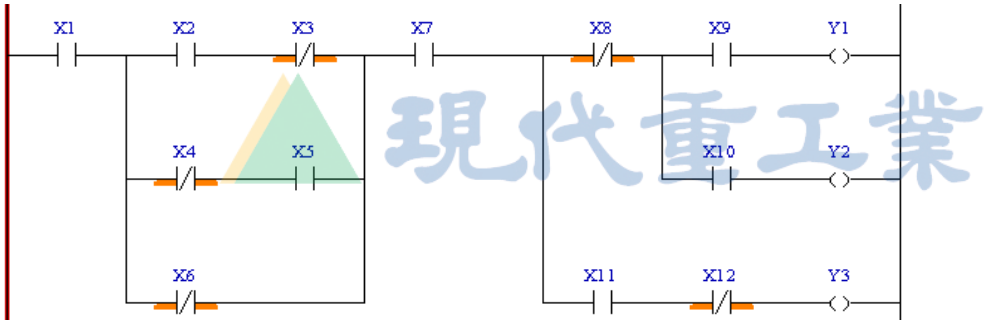
## 5.3. Ladder Diagram 与 Mnemonic 的关系

用 Ladder 编制的文件会自动变换为 Mnemonic、并向控制器发送。

(1) Ladder diagram




或点击 HRLadder 的 branch gap[ ]、显示以下数据。

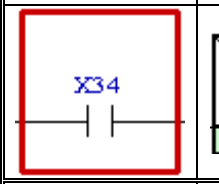



(2) Mnemonic

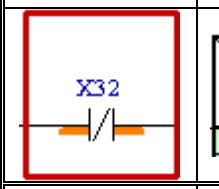
SOR XIC X1 BST XIC X2 XIO X3 NXB BST XIO X4 XIC X5 NXB XIO X6 BND BND XIC X7 BST XIO X8 BST XIC X9 OTE Y1 NXB XIC X10 OTE Y2 BND NXB XIC X11 XIO X12 OTE Y3 BND

# 6
# 命令语详细说明

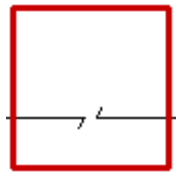
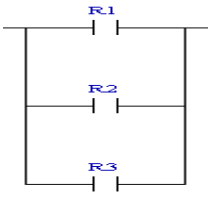
## 6.1. XIC(Examine if Closed): Close 检查

| 标记 | 可使用的 Relay 组合 |
|---|---|
|  | relay / type / inst / arg. 表格：<br>Input: X, DO; Output: Y,DI,R,K,SP; Timer: T; Counter: C; Memory: M, S; const (32bit) (L/F)<br>n (B,W,L,F) m<br>XIC: x x x x x x |
| 说明 | 因数值为 1、就会激活(active)Rung；如果是 0、就会处于非激活状态。 |
| 使用例 | XIC(X2); XIC(DO2); XIC(Y2); XIC(DI2); XIC(R2); XIC(K2); XIC(SP1); XIC(T2); XIC(C2); |
| 输入方法 | 可以把 X18 输入为 XB3/2 或 XW2/2 或 XL1/2。 |

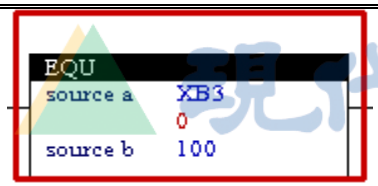
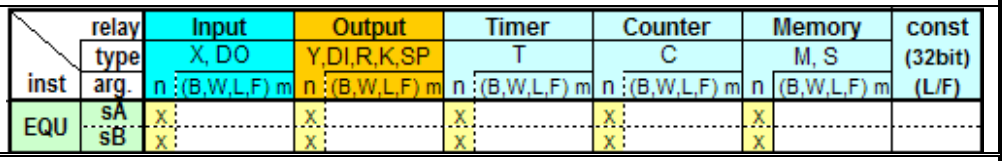
## 6.2. XIO(Examine if Open): Open 检查

| 标记 | 可使用的 Relay 组合 |
|---|---|
|  | relay / type / inst / arg. 表格：<br>Input: X, DO; Output: Y,DI,R,K,SP; Timer: T; Counter: C; Memory: M, S; const (32bit) (L/F)<br>n (B,W,L,F) m<br>XIO: x x x x x x |
| 说明 | 因数值为 0、就会激活(active)Rung；如果是 1、就会处于非激活状态。 |
| 使用例 | XIO(X2); XIO(DO2); XIO(Y2); XIO(DI2); XIO(R2); XIO(K2); XIO(SP1); XIO(T2); XIO(C2); |
| 输入方法 | 可以把 X18 输入为 XB3/2 或 XW2/2 或 XL1/2。 |

## 6.3. INV(Inverting): 反转

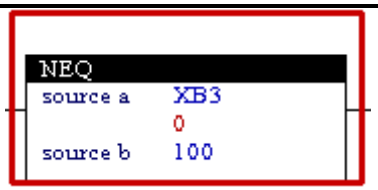
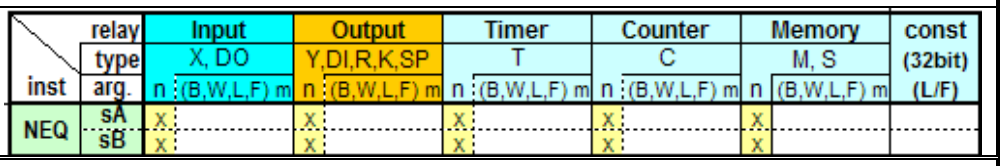
| 标记 | 说明 |
|---|---|
|  | 反转 Rung 之前的结果（激活←→非激活)。<br>此处理方式根据 De Morgan 的规律成为/(AxB)=/A+/B 或/(A+B)=/Ax/B、原本应以多个 Branch 构成 OR Logic、 但如果使用此方式、就无需 Branch 也可简单构成 AND Logic。<br>即、 (R1+R2+R3) = /(/R1x/R2x/R3)、以下两个 Logic 的结果会相同。<br>  <br> |

## 6.4. EQU(Equal): 检查是否相同

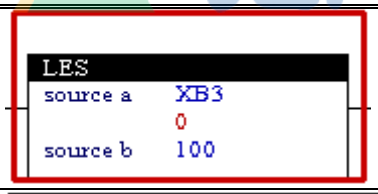
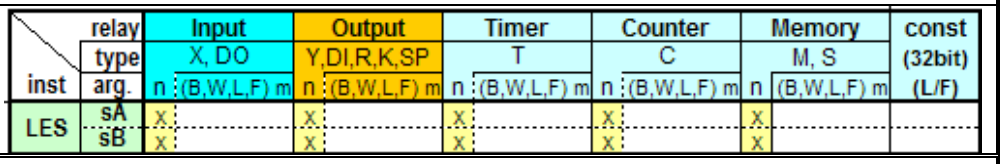
| 标记 |  |
|---|---|
| Relay 组合 |  |
| 因数 | sA: 要比较的第一个 Relay(或数字)<br>sB: 要比较的第二个 Relay(或数字) |
| 说明 | 比较两个值时如果相同、Rung 就会被激活。(激活接点) |
| 使用例 | EQU(XB3、100): XB3 的值为 100、Rung 就会被激活。<br>EQU(XB3、RW4): XB3 与 RW4 的值相同、Rung 就会被激活。 |

| inst \ relay | | Input | Output | Timer | Counter | Memory | const |
|---|---|---|---|---|---|---|---|
| | type | X, DO | Y,DI,R,K,SP | T | C | M, S | (32bit) |
| | arg. | n (B,W,L,F) m | n (B,W,L,F) m | n (B,W,L,F) m | n (B,W,L,F) m | n (B,W,L,F) m | (L/F) |
| EQU | sA | X | X | X | X | X | |
| | sB | X | X | X | X | X | |

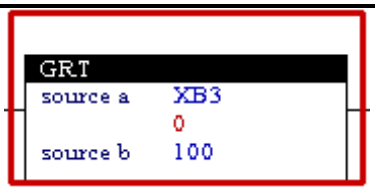
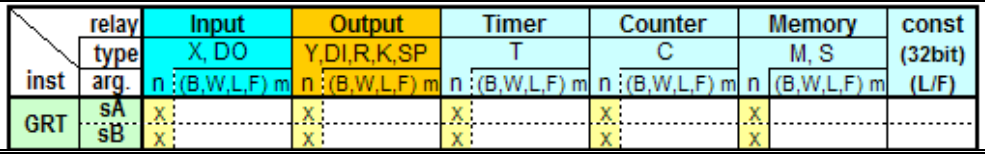
## 6.5. NEQ(Not Equal): 检查是否不同

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 要比较的第一个 Relay(或数字)<br>sB: 要比较的第二个 Relay(或数字) |
| 说明 | 比较两个值时不同、Rung 就会被激活。(激活接点) |
| 使用例 | NEQ(XB3、100): XB3 的值非为 100、 Rung 就会被激活。<br>NEQ(XB3、RW4): XB3 的值与 RW4 的值不同、Rung 就会被激活。 |

## 6.6. LES(Less Than): 检查是否小于基准值

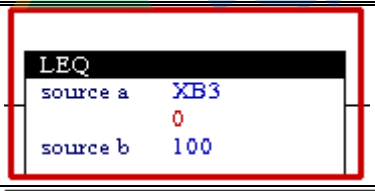
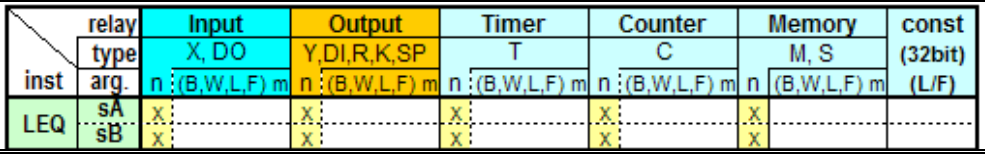
| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 要比较的第一个 Relay(或数字)<br>sB: 要比较的第二个 Relay(或数字) |
| 说明 | 如果 sA 小于 sB、 Rung 就会被激活。 (接点激活) |
| 使用例 | LES(XB3、100): XB3 的值小于 100、Rung 就会被激活。<br>LES(XB3、RW4): XB3 的值小于 RW4 的值、Rung 就会被激活。 |

## 6.7. GRT(Greater Than): 检查是否大于基准值

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 要比较的第一个 Relay(或数字)<br>sB: 要比较的第二个 Relay(或数字) |
| 说明 | sA 大于 sB、 Rung 就会被激活。 (接点激活) |
| 使用例 | GRT(XB3、100): XB3 的值大于 100、Rung 就会被激活。<br>GRT(XB3、RW4): XB3 的值大于 RW4 的值、Rung 就会被激活。 |

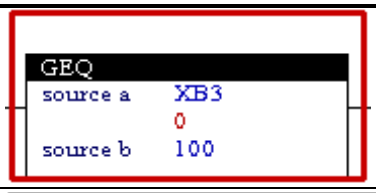
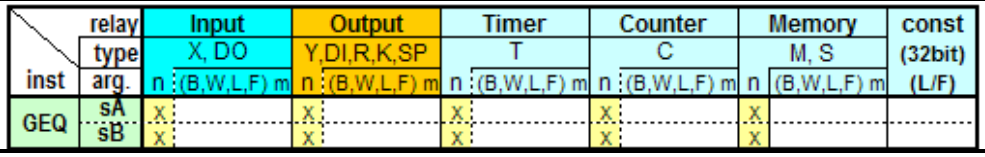
| relay / type / inst / arg. | Input X, DO n (B,W,L,F) m | Output Y,DI,R,K,SP n (B,W,L,F) m | Timer T n (B,W,L,F) m | Counter C n (B,W,L,F) m | Memory M, S n (B,W,L,F) m | const (32bit) (L/F) |
|---|---|---|---|---|---|---|
| GRT sA | X | X | X | X | X | |
| GRT sB | X | X | X | X | X | |

## 6.8. LEQ(Less Than or Equal): 检查是否比基准值小或相同

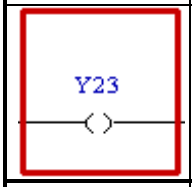
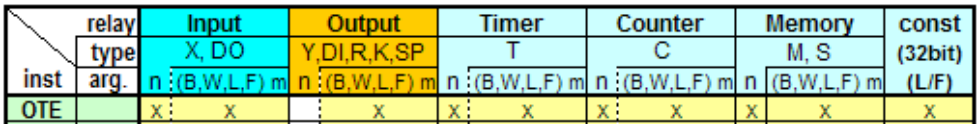
| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 要比较的第一个 Relay(或数字)<br>sB: 要比较的第二个 Relay(或数字) |
| 说明 | sA 小或同于 sB、 Rung 就会被激活。 (接点激活) |
| 使用例 | LEQ(XB3、100): XB3 的值比 100 小或相同、Rung 就会被激活。<br>LEQ(XB3、RW4): XB3 的值比 RW4 的值小或相同、Rung 就会被激活。 |

| relay / type / inst / arg. | Input X, DO n (B,W,L,F) m | Output Y,DI,R,K,SP n (B,W,L,F) m | Timer T n (B,W,L,F) m | Counter C n (B,W,L,F) m | Memory M, S n (B,W,L,F) m | const (32bit) (L/F) |
|---|---|---|---|---|---|---|
| LEQ sA | X | X | X | X | X | |
| LEQ sB | X | X | X | X | X | |

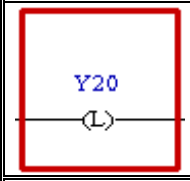
## 6.9. GEQ(Greater Than or Equal): 检查是否比基准值大或相同

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 要比较的第一个 Relay(或数字)<br>sB: 要比较的第二个 Relay(或数字) |
| 说明 | sA 比 sB 大或相同、Rung 就会被激活。 (接点激活) |
| 使用例 | GEQ(XB3、100): XB3 的值比 100 大或相同、Rung 就会被激活。<br>GEQ(XB3、RW4): XB3 的值比 RW4 的值大或相同、Rung 就会被激活。 |

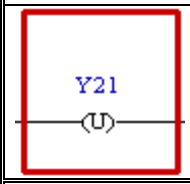
## 6.10. OTE(Output Energize): 一般输出

| | |
|---|---|
| 标记 | 可使用的 Relay 组合 |
|  |  |
| 说明 | 根据 Rung 状态输出输出信号。<br>即、 Rung 处于激活状态时以 On(high)输出信号、 处于非激活状态时以 Off(low) 输出。 |
| 因数例 | Y2; DI2; R2; K2; SP1; |
| 输入方法 | 可以把 Y18 输入为 YB3/2 或 YW2/2 或 YL1/2。 |

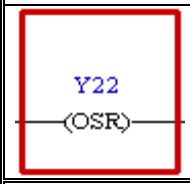
## 6.11. OTL(Output Latch): Latch 输出

| 标记 | 可使用的 Relay 组合 |
|---|---|
|  | relay / type / arg. / inst: Input (X, DO) n (B,W,L,F) m; Output (Y,DI,R,K,SP) n (B,W,L,F) m; Timer (T) n (B,W,L,F) m; Counter (C) n (B,W,L,F) m; Memory (M, S) n (B,W,L,F) m; const (32bit) (L/F)<br>OTL: x x / x / x x / x / x x / x / x x / x / x |
| 说明 | Rung 处于激活状态、 就会以 On(high)输出信号。<br>但如果 Rung 处于非激活状态、输出就不会变更。 |
| 因数例 | Y2; DI2; R2; K2; SP1; |
| 输入方法 | 可以把 Y18 输入为 YB3/2 或 YW2/2 或 YL1/2。 |

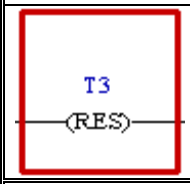
## 6.12. OTU(Output Unlatch): Unlatch 输出

| 标记 | 可使用的 Relay 组合 |
|---|---|
|  | relay / type / arg. / inst: Input (X, DO) n (B,W,L,F) m; Output (Y,DI,R,K,SP) n (B,W,L,F) m; Timer (T) n (B,W,L,F) m; Counter (C) n (B,W,L,F) m; Memory (M, S) n (B,W,L,F) m; const (32bit) (L/F)<br>OTU: x x / x / x x / x / x x / x / x x / x / x |
| 说明 | Rung 处于激活状态、 就会以 Off(low)输出信号。<br>但如果 Rung 处于非激活状态、输出就不会变更。 |
| 使用例 | Y2; DI2; R2; K2; SP1; |
| 输入方法 | 可以把 Y18 输入为 YB3/2 或 YW2/2 或 YL1/2。 |

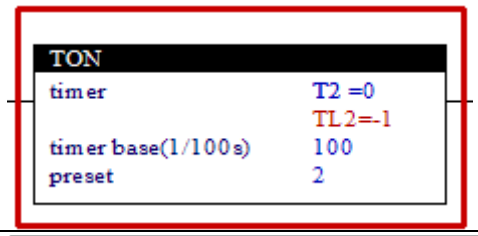
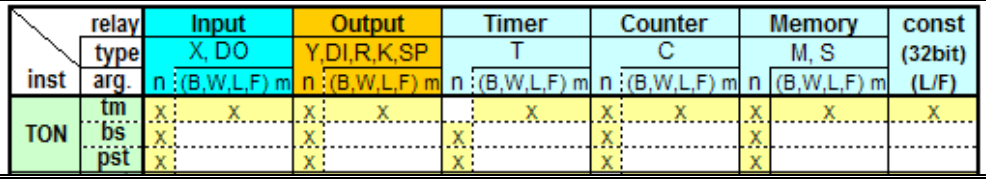
## 6.13. OSR(One Shot Rising): One Shot 输出

| 标记 | 可使用的 Relay 组合 |
|---|---|
|  | relay / type / arg. / inst: Input: X, DO (n, (B,W,L,F) m); Output: Y,DI,R,K,SP (n, (B,W,L,F) m); Timer: T (n, (B,W,L,F) m); Counter: C (n, (B,W,L,F) m); Memory: M, S (n, (B,W,L,F) m); const (32bit) (L/F)<br>OSR: x x x x x x x x x |
| 说明 | Rung 处于激活状态、 只在一次 scan 期间以 On(high)输出。<br>即、 Rung 从非激活状态变更为激活状态时、只在一个 scan 过程中 On 该 Relay。 |
| 因数例 | Y2; DI2; R2; K2; SP1; |
| 输入方法 | 可以把 Y18 输入为 YB3/2 或 YW2/2 或 YL1/2。 |

## 6.14. RES(Reset): 复位

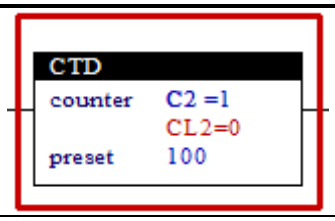
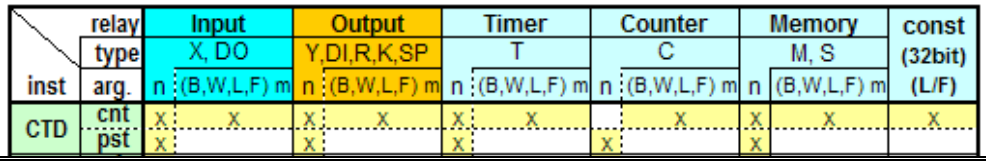
| 标记 | 可使用的 Relay 组合 |
|---|---|
|  | relay / type / arg. / inst: Input: X, DO (n, (B,W,L,F) m); Output: Y,DI,R,K,SP (n, (B,W,L,F) m); Timer: T (n, (B,W,L,F) m); Counter: C (n, (B,W,L,F) m); Memory: M, S (n, (B,W,L,F) m); const (32bit) (L/F)<br>RES: x x x x x x x x x |
| 说明 | 如果 Rung 处于激活状态、 就会清除输出(-1)。 |
| 因数例 | T2; C2; |

## 6.15. TON(Time On Delay): 计时器

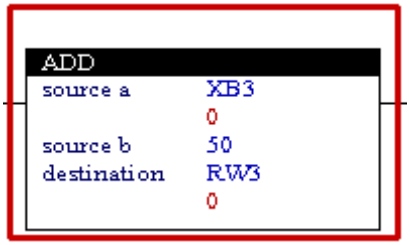
| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | tm: 指定计时器 Relay 编号。<br>bs: 把计时器的单位(如果是 100、就按 1 秒单位设置；如果是 10、就按 0.1 秒 单位设置、 ……)设为 Relay 或数字。<br>pst: 把 Freeset 值(时间[msec]=bs*pst*10)设置为 Relay 或数字。 |
| 说明 | 计算 Rung 的激活时间、过设置时间(bs x pst x 10) [msec]后该计时器 Relay 就会 On(high)。<br>但如果 Rung 处于非激活状态、就会即时清除(-1)。<br>参考) TL 的值为 1msec 单位。 |
| 使用例 | TON(T2、100、5): 利用计时器把 T2 设置为 (100x5x10[msec]=5[sec])秒。<br>TON(T3、RW3、RW4): 利用计时器把 T3 设置为与(RW3)x(RW4)x10 的值相同。 |



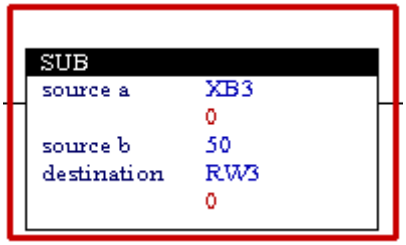
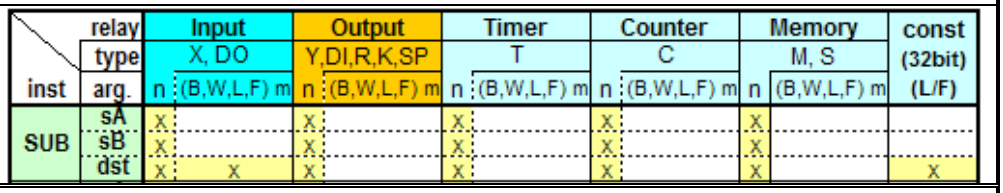
## 6.16. CTD(Count Down): 计数器

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | cnt: 指定计数器 Relay 编号。<br>pst: 用 Replay 或数字设置 Freeset 值。 |
| 说明 | Down Count Rung 的上升(从非激活变为激活)。<br>如果 CL 值为 0、 该计数器就会 On(high)、不再进行 Count。<br>Rung 处于激活时如果该 CL 值为负数、就把 pst 值保存于 CL。<br>参考) 即使 Rung 处于非激活状态、CL 也不会清除(-1)。 |
| 使用例 | CTD(C2、100): 将 C2 设置为(100)个 Down-Counter。<br>CTD(C3、RW3): 将 C3 设置为(RW3)值 Down-Counter。 |

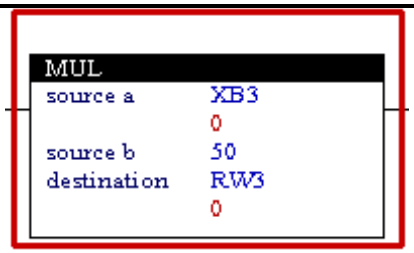
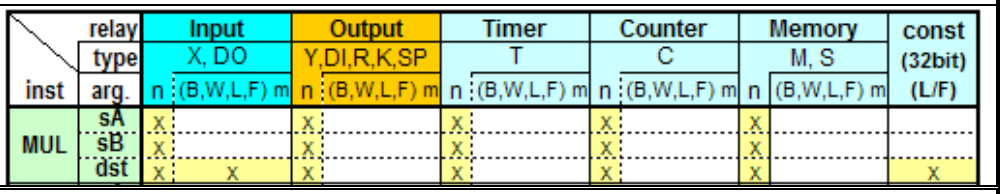
## 6.17. ADD(Add): 加

<table>
<tr><td>标记</td><td> 
</td></tr>
<tr><td>Relay 组合</td><td>
<table>
<tr><th rowspan="3">inst</th><th>relay</th><th colspan="2">Input</th><th colspan="2">Output</th><th colspan="2">Timer</th><th colspan="2">Counter</th><th colspan="2">Memory</th><th rowspan="2">const<br>(32bit)<br>(L/F)</th></tr>
<tr><th>type</th><th colspan="2">X, DO</th><th colspan="2">Y,DI,R,K,SP</th><th colspan="2">T</th><th colspan="2">C</th><th colspan="2">M, S</th></tr>
<tr><th>arg.</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th></tr>
<tr><td rowspan="3">ADD</td><td>sA</td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td></td></tr>
<tr><td>sB</td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td></td></tr>
<tr><td>dst</td><td>X</td><td>X</td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td></tr>
</table>
</td></tr>
<tr><td>因数</td><td>sA: 计算所需的第一个 Relay(或数字)<br>sB: 计算所需的第二个 Relay(或数字)<br>dst: 保存结果的 Relay</td></tr>
<tr><td>说明</td><td>Rung 处于激活状态、 在 sA 值添加 sB 的值后保存于 dst。<br>如果计算结果发生 Overflow、就设置为 SP8=1。</td></tr>
<tr><td>使用例</td><td>ADD(XB3、RB2、RW3): (RW3)=(XB3)+(RB2)<br>ADD(XB3、RB2、RB2): (RB2)=(XB3)+(RB2)<br>ADD(XB3、50、RW3): (RW3)=(XB3)+50</td></tr>
</table>

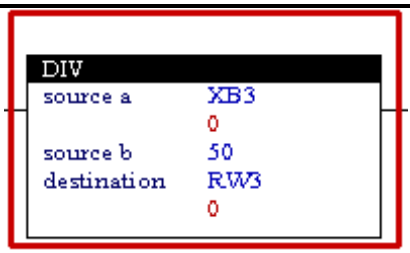
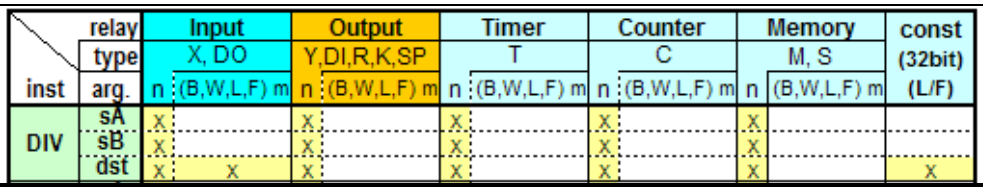
## 6.18. SUB(Subtract): 减

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 计算所需的第一个 Relay(或数字)<br>sB: 计算所需的第二个 Relay(或数字)<br>dst: 保存结果的 Relay |
| 说明 | Rung 处于激活状态、 从 sA 值减去 sB 的值后保存于 dst。 |
| 使用例 | SUB(XB3、RB2、RW3): (RW3)=(XB3)-(RB2)<br>SUB(XB3、RB2、RB2): (RB2)=(XB3)-(RB2)<br>SUB(XB3、50、RW3): (RW3)=(XB3)-50 |

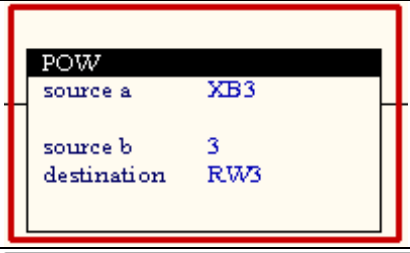
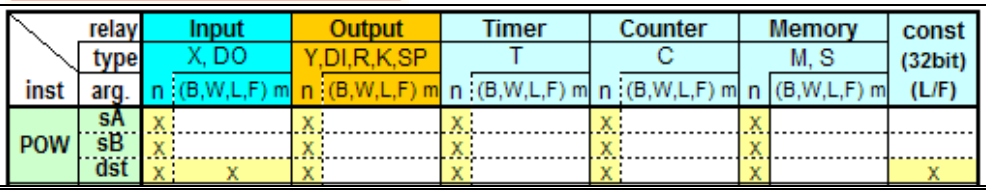
## 6.19. MUL(Multiply): 乘

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 计算所需的第一个 Relay(或数字)<br>sB: 计算所需的第二个 Relay(或数字)<br>dst: 保存结果的 Relay |
| 说明 | Rung 处于激活状态、在 sA 值乘于 sB 的值后保存于 dst。<br>如果在计算结果发生 Overflow、就设置为 SP8=1。 |
| 使用例 | MUL(XB3、RB2、RW3): (RW3)=(XB3)x(RB2)<br>MUL(XB3、RB2、RB2): (RB2)=(XB3)x(RB2)<br>MUL(XB3、50、RW3): (RW3)=(XB3)x50 |

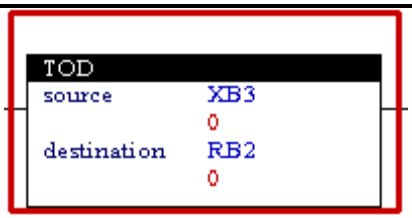
## 6.20. DIV(Divide): 除

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 计算所需的第一个 Relay(或数字)<br>sB: 计算所需的第二个 Relay(或数字)<br>dst: 保存结果的 Relay |
| 说明 | Rung 处于激活状态、 将 sA 值除以 sB 值后保存于 dst。<br>如果 sB 为 0 或计算结果发生 Overflow 就设置为 SP8=1。 |
| 使用例 | DIV(XB3、RB2、RW3): (RW3)=(XB3)/(RB2)<br>DIV(XB3、RB2、RB2): (RB2)=(XB3)/(RB2)<br>DIV(XB3、50、RW3): (RW3)=(XB3)/50 |

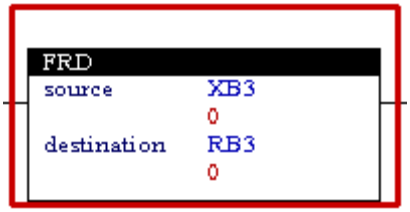
## 6.21. POW(Power): 累乘

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | sA: 计算所需的第一个 Relay(或数字)<br>sB: 计算所需的第二个 Relay(或数字)<br>dst: 保存结果的 Relay |
| 说明 | Rung 处于激活状态、 在 sA 值累乘 sB 值后保存于 dst。<br>如果计算结果发生 Overflow、就设置为 SP8=1。 |
| 使用例 | POW(2、3、RW3): (RW3)=2^3=2*2*2=8<br>POW(XB3、RB2、RW3): (RW3)=(XB3)^(RB2)<br>POW(XB3、RB2、RB2): (RB2)=(XB3)^(RB2)<br>POW(XB3、3、RW3): (RW3)=(XB3)^3 |

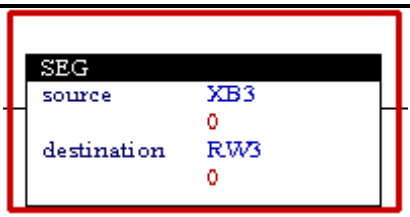
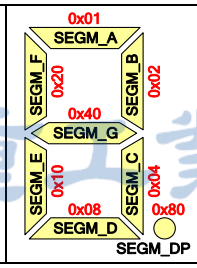
## 6.22. TOD(Convert to BCD): 变换为 BCD 值

<table>
<tr><td>标记</td><td> 
</td></tr>
<tr><td>Relay 组合</td><td>
<table>
<tr><td rowspan="3">inst</td><td>relay</td><td colspan="2">Input</td><td colspan="2">Output</td><td colspan="2">Timer</td><td colspan="2">Counter</td><td colspan="2">Memory</td><td>const</td></tr>
<tr><td>type</td><td colspan="2">X, DO</td><td colspan="2">Y,DI,R,K,SP</td><td colspan="2">T</td><td colspan="2">C</td><td colspan="2">M, S</td><td rowspan="2">(32bit)<br>(L/F)</td></tr>
<tr><td>arg.</td><td>n</td><td>(B,W,L,F) m</td><td>n</td><td>(B,W,L,F) m</td><td>n</td><td>(B,W,L,F) m</td><td>n</td><td>(B,W,L,F) m</td><td>n</td><td>(B,W,L,F) m</td></tr>
<tr><td rowspan="2">TOD</td><td>src</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>u</td></tr>
<tr><td>dst</td><td>x</td><td>x</td><td>x</td><td>u</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>u</td><td>x</td></tr>
</table>
</td></tr>
<tr><td>因数</td><td>src: 原本(source) Relay(或数字)<br>dst: 目的地(destination) Relay</td></tr>
<tr><td>说明</td><td>Rung 处于激活状态、 将 src 的值变换为 CD 后保存于 dst。<br>此命令为 BCD 形式、使用在在 7’ 显示值的装置时方便。<br>如果 dst 为 Bite(B)形式、就变换为两个 Digit、 如果是 Word(W)形式、就会变换为 4 个 Digit。<br>如果 Src 的值大于 Digit 数、就设置为 SP7=1。</td></tr>
<tr><td>使用例</td><td>TOD(XB3、RB2):将 XB3 的值变换为 BCD 后保存于 RB2。<br>(参考、 BCD(Binary Coded Decimal)表示 4bit 的代码值可以具备 0~9 范围值得数字。即、在 BCD 以 4bit 显示的数字 0~F 中不使用 A~F。)<br>如果是(XB3)=&H7B(123)、就把&H23(35)保存于 RB2、因为是&H7B(123)>&H63(99)、设置为 SP7=1。</td></tr>
</table>

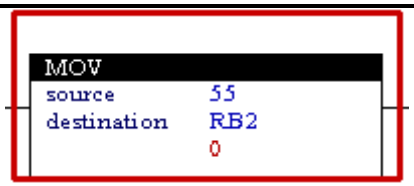
## 6.23. FRD(Convert form BCD to Integer): 变换为 integer

<table>
<tr><td>标记</td><td> 
</td></tr>
<tr><td>Relay 组合</td><td>
<table>
<tr><td rowspan="3">inst</td><td>relay</td><td colspan="2">Input</td><td colspan="2">Output</td><td colspan="2">Timer</td><td colspan="2">Counter</td><td colspan="2">Memory</td><td>const</td></tr>
<tr><td>type</td><td colspan="2">X, DO</td><td colspan="2">Y,DI,R,K,SP</td><td colspan="2">T</td><td colspan="2">C</td><td colspan="2">M, S</td><td rowspan="2">(32bit)<br>(L/F)</td></tr>
<tr><td>arg.</td><td>n</td><td>(B,W,L,F) m</td><td>n</td><td>(B,W,L,F) m</td><td>n</td><td>(B,W,L,F) m</td><td>n</td><td>(B,W,L,F) m</td><td>n</td><td>(B,W,L,F) m</td></tr>
<tr><td rowspan="2">FRD</td><td>src</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>u</td></tr>
<tr><td>dst</td><td>x</td><td>x</td><td>x</td><td>u</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>u</td><td>x</td></tr>
</table>
</td></tr>
<tr><td>因数</td><td>src: 原本(source) Relay(或数字)<br>dst: 目的地(destination) Relay</td></tr>
<tr><td>说明</td><td>如果 Rung 处于激活状态、 就把 src 的 BCD 值变换为 integer 后保存于 dst。<br>此命令在接收以 BCD 形式输出的 Cam 开关值时可方便使用。<br>如果 src 的值非为 BCD、就设置为 SP7=1。<br>并且、 src 为 Word(W)形式、dst 为 Bite(B)形式时、要变换的 src 最大值为&H9999、因此 integer 的变换结果成为 9999(&H270F)、超过 Bite 的范围&Hff、从而发生 Over flow、这时设置为 SP7=1。</td></tr>
<tr><td>使用例</td><td>把 FRD(XB3、RB2): XB3 的值(BCD)变换为 integer 后保存于 RB2。<br>如果是(XB3)=&H23(35)、 就把 23=&H17 保存于 RB2。</td></tr>
</table>

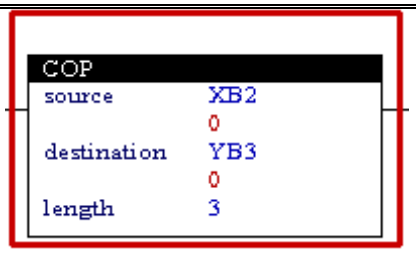
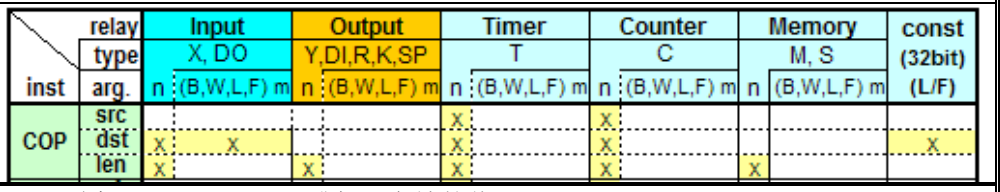
## 6.24. SEG(7'Segment): 变换为 7’ Segment 值

<table>
<tr><td>标记</td><td> 
</td></tr>
<tr><td>Relay 组合</td><td>
<table>
<tr><th rowspan="3">inst</th><th>relay</th><th colspan="2">Input</th><th colspan="2">Output</th><th colspan="2">Timer</th><th colspan="2">Counter</th><th colspan="2">Memory</th><th rowspan="3">const<br>(32bit)<br>(L/F)</th></tr>
<tr><th>type</th><th colspan="2">X, DO</th><th colspan="2">Y,DI,R,K,SP</th><th colspan="2">T</th><th colspan="2">C</th><th colspan="2">M, S</th></tr>
<tr><th>arg.</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th></tr>
<tr><td rowspan="2">SEG</td><td>src</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>x</td><td>u</td><td>u</td></tr>
<tr><td>dst</td><td>x</td><td>x</td><td>x</td><td>u</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>u</td><td>x</td></tr>
</table>
</td></tr>
<tr><td>因数</td><td>src: 原本(source) Relay(或数字)<br>dst: 目的地(destination) Relay</td></tr>
<tr><td>说明</td><td>Rung 处于激活状态、将 src 的值变换为 7’ Segment 值(8bit)后保存于 dst。<br>Dst 为 Word(W)形式、就把 2 个 7’ Segment 值(8bit)保存于 dst。</td></tr>
<tr><td>7’ Segment 数据</td><td>#define SEGM_A 0x01<br>#define SEGM_B 0x02<br>#define SEGM_C 0x04<br>#define SEGM_D 0x08<br>#define SEGM_E 0x10<br>#define SEGM_F 0x20<br>#define SEGM_G 0x40<br>#define SEGM_DP 0x80<br> 
<br>#define SEGD_0 (SEGM_A|SEGM_B|SEGM_C|SEGM_D|SEGM_E|SEGM_F)<br>#define SEGD_1 (SEGM_B|SEGM_C)<br>#define SEGD_2 (SEGM_A|SEGM_B|SEGM_G|SEGM_E|SEGM_D)<br>#define SEGD_3 (SEGM_A|SEGM_B|SEGM_C|SEGM_D|SEGM_G)<br>#define SEGD_4 (SEGM_B|SEGM_C|SEGM_F|SEGM_G)<br>#define SEGD_5 (SEGM_A|SEGM_C|SEGM_D|SEGM_F|SEGM_G)<br>#define SEGD_6 (SEGM_A|SEGM_C|SEGM_D|SEGM_E|SEGM_F|SEGM_G)<br>#define SEGD_7 (SEGM_A|SEGM_B|SEGM_C)<br>#define SEGD_8 (SEGM_A|SEGM_B|SEGM_C|SEGM_D|SEGM_E|SEGM_F|SEGM_G)<br>#define SEGD_9 (SEGM_A|SEGM_B|SEGM_C|SEGM_F|SEGM_G)<br>#define SEGD_A (SEGM_A|SEGM_B|SEGM_C|SEGM_E|SEGM_F|SEGM_G)<br>#define SEGD_B (SEGM_C|SEGM_D|SEGM_E|SEGM_F|SEGM_G)<br>#define SEGD_C (SEGM_A|SEGM_D|SEGM_E|SEGM_F)<br>#define SEGD_D (SEGM_B|SEGM_C|SEGM_D|SEGM_E|SEGM_G)<br>#define SEGD_E (SEGM_A|SEGM_D|SEGM_E|SEGM_F|SEGM_G)<br>#define SEGD_F (SEGM_A|SEGM_E|SEGM_F|SEGM_G)</td></tr>
<tr><td>使用例</td><td>SEG(XB3、RW3): 相当于 XB3 值的 7’ Segment 值保存于 RW3。<br>如果(XB3)=(&H17)、<br>就把结合上述 SEGD_1(SEGM_B|SEGM_C=0x02|0x04=0x06)=&H06 和<br>上述 SEGD_7(SEGM_A|SEGM_B|SEGM_C=0x01|0x02|0x04=0x07)=&H07 的&H0607 保存于 RW3。</td></tr>
</table>

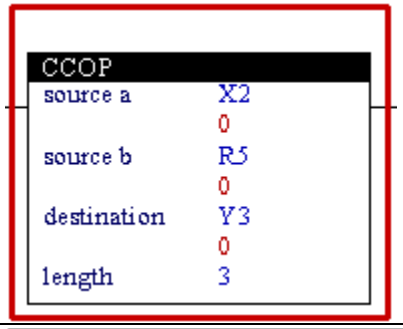
## 6.25. MOV(Move): 移动

<table>
<tr><td>标记</td><td> 
</td></tr>
<tr><td>Relay 组合</td><td>
<table>
<tr><th rowspan="3">inst</th><th>relay</th><th colspan="2">Input</th><th colspan="2">Output</th><th colspan="2">Timer</th><th colspan="2">Counter</th><th colspan="2">Memory</th><th rowspan="3">const<br>(32bit)<br>(L/F)</th></tr>
<tr><th>type</th><th colspan="2">X, DO</th><th colspan="2">Y,DI,R,K,SP</th><th colspan="2">T</th><th colspan="2">C</th><th colspan="2">M, S</th></tr>
<tr><th>arg.</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th></tr>
<tr><td rowspan="2">MOV</td><td>src</td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td></td></tr>
<tr><td>dst</td><td>X</td><td>X</td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td></tr>
</table>
</td></tr>
<tr><td>因数</td><td>src: 原本(source) Relay(或数字)<br>dst: 目的地(destination) Relay</td></tr>
<tr><td>说明</td><td>Rung 处于激活状态、 就把 src 的值复制到 dst。<br>如果、 src 为 Word(W)形式、dst 为 Bite(B)形式、在 src 的值中只把下位 Bite 复制到 dst。<br>并且、 内藏 PLC 的所有数据都以有符号的数据处理、src 为 Bite(B)形式、值为 -1(&Hff)时复制为 Word(W)形式的 dst、就会复制为-1(&HFFFF)。(&H00ff 成为 255 的值。)</td></tr>
<tr><td>使用例</td><td>MOV(XB3、RB2): XB3 的值复制到 RB2。<br>MOV(55、RB2): 将 55 复制到 RB2。<br>MOV(-30、RB2): 将-30 复制到 RB2。</td></tr>
</table>

## 6.26. COP(Copy data): Copy

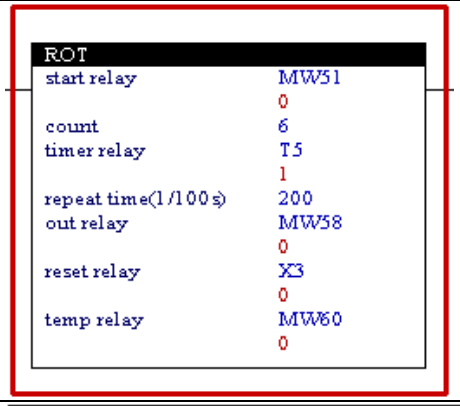
| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | src: 原本(source) Relay 或想要充填的值(const)<br>dst: 目的地(destination) Relay<br>len: 数量(length) Relay(或数字) |
| 说明 | Rung 处于激活状态、 从 src 位置开始按 len 数复制到 det 位置。<br>Src 为数字时、 从 dst 开始按 len 数充填 src 值。这时如果 dst 为 bit 形式、 src 的数字为 0 就用 OFF 充填、 如果不是 0 就用 ON 充填。<br>Src 为 Relay 时、 src 和 dst 的数据形式应相同。<br>即、 如果 src 为 Bit、dst 也应为 Bit;如果 src 为 Bite(B)、dst 也应为 Bite(B)；src 为 Word(W)、dst 也应是(W)。<br>如果 src+len 大于 src Relay 最大数量或 dst+len 大于 dst Relay 最大数量、 就按 Relay 最大数量进行复制。 |
| 使用例 | 按指定数复制的例)<br>COP(X2、Y3、4): Y3=X2、 Y4=X3、 Y5=X4、 Y6=X5<br>COP(XB2、YB3、3): YB3=XB2、 YB4=XB3、 YB5=XB4<br>COP(XW2、YW3、2): YW3=XW2、 YW4=XW3<br><br>用指定值(const)充填的例)<br>COP(0、Y3、4): Y3=OFF、 Y4=OFF、 Y5=OFF、 Y6=OFF<br>COP(1、Y3、4): Y3=ON、 Y4=ON、 Y5=ON、 Y6=ON<br>COP(25、Y3、4): Y3=ON、 Y4=ON、 Y5=ON、 Y6=ON<br>COP(25、YB3、3): YB3=25、 YB4=25、 YB5=25<br>COP(&H55AA、YW3、2): YW3=&H55AA、 YW4=&H55AA<br>COP(0、MW3、50): MW3=0、 MW4=0、 MW5=0、 MW6=0、 ~、 MW52=0 |

## 6.27. CCOP(Conditional Copy data): 按条件复制

| | |
|---|---|
| 标记 |  |
| Relay 组合 | relay / type / inst / arg. 对照表：<br>Input: X, DO — n (B,W,L,F) m<br>Output: Y,DI,R,K,SP — n (B,W,L,F) m<br>Timer: T — n (B,W,L,F) m<br>Counter: C — n (B,W,L,F) m<br>Memory: M, S — n (B,W,L,F) m<br>const (32bit) (L/F)<br>CCOP: sA, sB, dst, len |
| 因数 | sA: Rung 处于激活状态时、要复制的原本(source) Relay 或要充填的值(const)<br>sB: Rung 处于非激活状态时、 要复制的原本(source) Relay 或要充填的值(const)<br>dst: 目的地(destination) Relay<br>len: 数量(length) Relay(或数字) |
| 说明 | 根据 Rung 状态、从 sA 或 sB 的位置按 len 数复制到 det 位置。<br>Src 为数字时、从 dst 开始按 len 数以 sA/sB 值充填。这时如果 dst 为 bit 形式、 sA /sB 的数字为 0 就以 OFF 充填、如果不是 0、就以 ON 充填。<br>sA/sB 为 Relay 时、 sA/sB 与 dst 的数据形式应相同。<br>即、 sA/sB 为 Bit、 dst 也应为 Bit； 如果 sA/sB 为 Byte(B)、dst 也应为 Byte(B); 如果 sA/sB 为 Word(W)、dst 也应为 Word(W)。<br>如果(sA/sB)+len 比 sA/sB Relay 最大数量大或 dst+len 大于 dst Relay 最大数量时、只复制 Relay 的最大数量。 |
| 使用例 | 按指定数量复制的例)<br>CCOP(X2、R5、Y3、3): Rung 处于激活状态、 Y3=X2、 Y4=X3、 Y5=X4<br>Rung 处于非激活状态、 Y3=R5、 Y4=R6、 Y5=R7<br>CCOP(1、0、Y3、3): Rung 处于激活状态、 Y3=ON、 Y4=ON、 Y5=ON<br>Rung 处于非激活状态、 Y3=OFF、 Y4=OFF、 Y5=OFF<br>参考) 因数使用例与 COP 命令类似、请参考 COP 命令。 |

## 6.28. ROT(Rotating Output): 按序输出

**标记**




**Relay 组合**

| inst | relay type arg. | Input X, DO n | Input X, DO (B,W,L,F) m | Output Y,DI,R,K,SP n | Output Y,DI,R,K,SP (B,W,L,F) m | Timer T n | Timer T (B,W,L,F) m | Counter C n | Counter C (B,W,L,F) m | Memory M, S n | Memory M, S (B,W,L,F) m | const (32bit) (L/F) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ROT | st | X | | X | | X | | X | | X | | X |
| | cnt | X | | X | X | X | | X | | X | | |
| | tm | X | X | X | X | | X | X | X | X | X | X |
| | rep | X | | X | | X | | X | | X | | |
| | out | X | X | X | | X | X | X | X | X | | X |
| | rst | | X | | X | | X | | X | X | | X |
| | tmp | X | X | X | | X | X | X | X | X | | X |

**因数**

st: 开始(start) Relay
cnt: 数量(count) Relay(或数字)
tm: 计时器(timer) Relay(1/100sec 单位)
rep: 指定反复(repeat)时间的 Relay(或数字)
out: 输出(output)结果的 Relay
rst: 复位(reset)输出状态的 Relay
tmp: 显示数值输出方向的 Relay

**说明**

Rung 处于激活状态、 在 rep 时间内、从 st Relay 到 cnt 范围内的 Relay 值中非为 0 的值反复输出到 out Relay。
如果、 向 rst Relay 输入信号、从 st Relay 按 cnt 数充填 0、 将计时器值初始化为 rep 值、outRelay 也以 0 输出。
输出错误编号的装置只有一个、且所发生的错误类型多时可方便使用此命令、 以便在指定时间内输出错误编号。

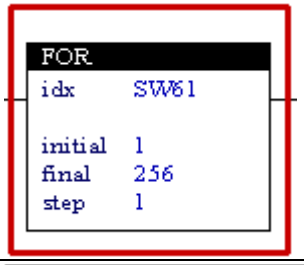
**使用例**

```
|-------(错误条件 1)------------------[MOV(21、MW51)]---|
|-------(错误条件 2)------------------[MOV(22、MW52)]---|
|-------(错误条件 3)------------------[MOV(23、MW53)]---|
|-------(错误条件 4)------------------[MOV(24、MW54)]---|
|-------(错误条件 5)------------------[MOV(25、MW55)]---|
|-------(错误条件 6)------------------[MOV(26、MW56)]---|
|-----------[ROT(MW51、6、T5、200、MW58、X3、MW60)]---|
|-------------------------------------[TOD(MW58、YB3)]---|
```

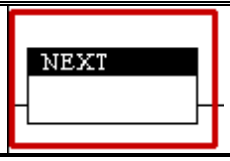
上述 Logic 在错误条件 1~6 中发生一个以上错误、 错误编号就会保存于 MW51~MW56、根据 ROT 命令发生的错误编号会在 MW58 保存 2 秒钟、并通过 TOD 命令变换为 BCD 值、按序显示于与 YB3 连接的显示装置。
如果、 信号被输入到连接于外部错误复位的 X3、 保存错误编号的 MW51~MW56 内容就会被清除为 0、 MW58 和 MW60 也被清除为 0、在显示装置显示 0。


現代重工業

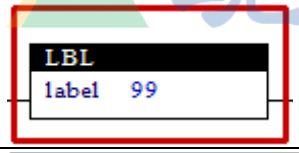
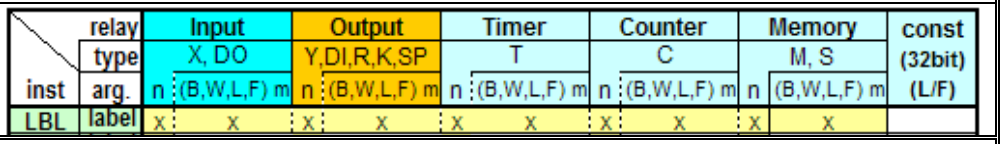
## 6.29. FOR(FOR): Block 反复

<table>
<tr><td>标记</td><td> 
</td></tr>
<tr><td>Relay 组合</td><td>
<table>
<tr><th colspan="2">relay</th><th colspan="2">Input</th><th colspan="2">Output</th><th colspan="2">Timer</th><th colspan="2">Counter</th><th colspan="2">Memory</th><th>const</th></tr>
<tr><th colspan="2">type</th><th colspan="2">X, DO</th><th colspan="2">Y,DI,R,K,SP</th><th colspan="2">T</th><th colspan="2">C</th><th colspan="2">M, S</th><th>(32bit)</th></tr>
<tr><th>inst</th><th>arg.</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>n</th><th>(B,W,L,F) m</th><th>(L/F)</th></tr>
<tr><td rowspan="4">FOR</td><td>idx</td><td>X</td><td>X</td><td>X</td><td></td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td><td></td><td>X</td></tr>
<tr><td>init</td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td></td></tr>
<tr><td>final</td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td></td></tr>
<tr><td>step</td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td>X</td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>因数</td><td>idx: 目录(index) Relay<br>init: 初始值(initial value) Relay 或数字<br>final: 结束值(final value) Relay 或数字<br>step: 作为步骤(step)的 Relay 或数字</td></tr>
<tr><td>说明</td><td>Rung 处于激活状态、从 init 到 final 按 Step 增加 idx Relay 值并反复执行到 NEXT Block。<br>执行 FOR 文时、在无条件 idx 代入 init 值。<br>FOR/NEXT 文最大可以重叠至 10 个。<br>→ FOR() FOR() FOR() ….NEXT NEXT NEXT<br>step>0 时、如果 init>final、就不执行、直接 Jump 到 NEXT。<br>step<0 时、如果 init<final 、就不执行、直接 Jump 到 NEXT。<br>Final 和 step 即使指定变数、也只适用首次开始 for 文时的值。<br>特别在 FOR 文的中间脱离时、可以使用 JMP(负数)命令。(请参考 JMP 命令说明)<br>注意) FOR 命令不进行 Branch 所需的另外处理。</td></tr>
<tr><td>使用例</td><td>FOR(RB10、1、4、1): 从 RB10=1 到 4 逐一增加、反复执行到 NEXT 命令。<br>{|--[FOR(SW61、1、256、1)]-—[XIC(X-1)]—-[OTL(Y-1)]—-[NEXT]--|}: 从 SW61=1 到 256 逐一增加、反复执行{XIC(X-1)、 OTL(Y-1)}命令。<br>即、 idx 使用相对 Addressing(SW61~SW69)所需的 Relay、 其 XIC 的 XRelay 和 OTL 的 YRelay 为 “-1”、适用 SW61 值的编号、因此只把在 X1~X256 中属于 High 信号编号的 YRelay 编号输出为 High、未输入编号的 Y 输出保持之前状态。<br>参考) 相对 Addressing 是指、不管任何形式的 Relay 将其编号设为-1~-9 值、就按保存于 SW61~SW69 的值指定 Relay 编号的方式。</td></tr>
</table>

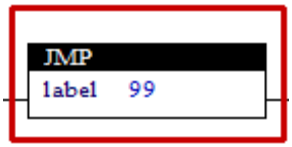
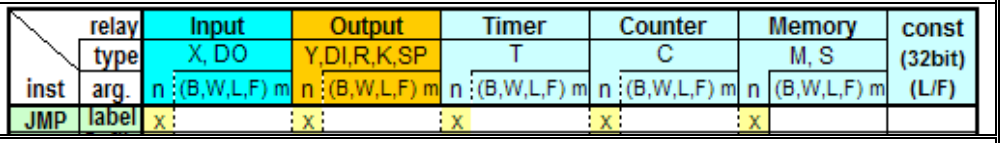
## 6.30. NEXT(NEXT): Block Next

| | |
|---|---|
| 标记 |  |
| 因数 | 无 |
| 说明 | 按 FOR 命令的 step 工作如下。<br>step>0 时、反复执行到 idx Relay 值比 final 小或相同为止、<br>step<0 时、 反复执行到 idx Relay 值比 final 大或相同为止。<br>在没有 FOR 文的情况下执行 NEXT 文时、NEXT 命令会被忽视。<br>注意)<br>FOR/NEXT 命令不会另行 Branch 处理、 如果在 Branch 内记录 FOR 命令、 在 Branch 外部或其它 Branch 内记录 NEXT、FOR 命令就会进行误工作。 |
| 使用例 | 请参考 FOR 命令的使用例。 |

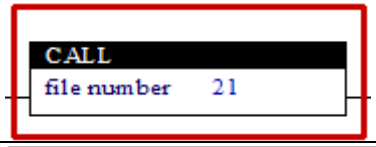
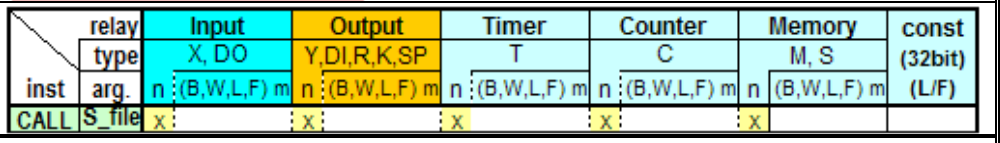
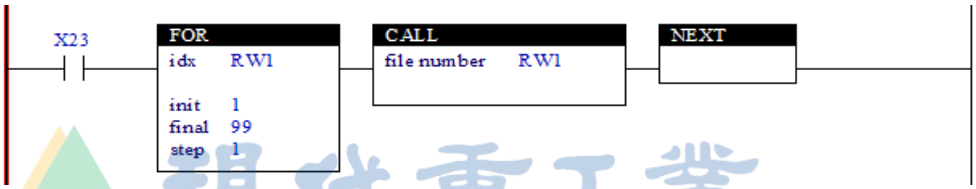
## 6.31. LBL(Label): Label 指定

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | label: Label 编号、 用大于 0 的数字(const)指定编号。 |
| 说明 | 利用 JMP 命令指定要 jump 的 Label 位置。<br>在 LBL 命令、指定位置与 Rung 的激活/非激活无关。<br>在一个 file 内有相同编号的 LBL 命令时、 |
| 使用例 | 与 JMP 命令一起使用、请参考 JMP 命令说明。 |

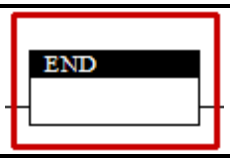
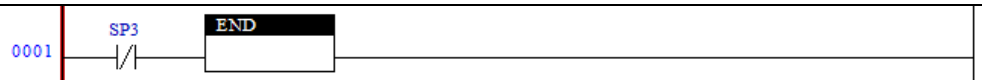
## 6.32. JMP(Jump): Jump

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | label: Label 编号数字(const)或 Relay |
| 说明 | Rung 处于激活状态、Jump 到有与用 label 指定的 Label 值相同之 LBL 命令的位置。特别 label<0 时、可在 FOR 文的中间脱离时使用。 (按负数指定值 skip)<br>注意 1)<br>Label 的位置处于比 JMP 命令高的位置、而且 JMP 命令前面没有条件时、可能会陷入无限 Roof、应注意。这时、Scan Time 会超过 5 秒、因此被设置为 SP17=On。<br>注意 2)<br>在 FOR/NEXT 命令 Block 内使用 JMP(正数)、不考虑脱离 Block 时的情况、因此 Block 控制可能会发生错误。这时应采取措施、以便利用 JMP(负数)Jump 到 NEXT 命令。 |
| 使用例 | {\|--[XIC(X21)]—-[**JMP(25)**]-—[……]—-[**LBL(25)**]--\|}: 如果 X21Relay 为 1、 就会根据 JMP(25)命令 Jump 到有 LBL(25) 命令的位置。LBL 命令位于其他 Rung 也会正常工作。 |

## 6.33. CALL(Call): Sub Ladder 呼叫

| | |
|---|---|
| 标记 |  |
| Relay 组合 |  |
| 因数 | file number: 指定 Sub Ladder 文件编号(1~99)的 Relay 或数字 |
| 说明 | Rung 处于激活状态、 就呼叫指定为(file number)的编号之 Sub Ladder。<br>Sub Ladder 的文件名为 S01xxxx.LAD ~ S99xxxx.LAD、最大具有 99 个、 用户可任一设置文件名的 xxxx、最大可设置 15 字符。 |
| 使用例 |  <br>如果 X23 信号被输入、 就会按序呼叫从 S01xxxx.LAD 到 S99xxxx.LAD 的文件。<br>CALL 命令执行结果、如果该编号的 Sub Ladder 不存在或数值范围超出 1~99、就会被设置为 SP18=On；如果正常执行、就会成为 SP18=Off。因此、根据需要须有 SubLadder 时、可在呼叫后利用 SP18 检测错误。<br>从 Ladder 通过 CALL 命令呼叫 1~99 号的 SubLadder、 按应用编号安排 SubLadder 编号、 并用控制器导入应用所需的 SubLadder、就可以期待该应用 Ladder 自动启动的效果。 |

## 6.34. END(End): 关闭 Ladder

| | |
|---|---|
| 标志 |  |
| 人数 | 无 |
| 说明 | Rung 处于激活状态、 就关闭现在启动的 Ladder。<br>如果现在启动的 Ladder 为 Sub Ladder、就 Return 为 Main Ladder；如果当前启动的 Ladder 为 Main Ladder、 就关闭后重新启动。 |
| 使用例 |  <br>需要初始化步骤时、用 S01_Initialize.LAD 文件编制处理化处理过程、然后在 MainLadder 呼叫 1 号、在 S01 的第一个 rung 添加、 SP3 是在开始运行后的第一个 Scan 启动的 Relay、S01 只执行一次、下一个 Scan 开始会在第一个 Rung Return。 |

現代重工業

■ **Head Office**
1, Jeonha-dong, Dong-gu, Ulsan, Korea
TEL : 82-52-230-7901 / FAX : 82-52-230-7900

■ **BEIJING HYUNDAI**
JINGCHENG MACHINERY CO.,LTD.
NO.2NANLI,LUGOUQIAO, FENGTAI DISTRICT,BEIJING
TEL : 86-010-8321-2588 / FAX : 86-010-8321-2188
E-Mail : robot_as@yahoo.com.cn
POST CODE : 100072

■ **韩国现代重工业本部**
蔚山市东区田下洞 1 番地
TEL : 82-52-230-7901 / FAX : 82-52-230-7900

■ **北京现代京城工程机械有限公司**
北京市丰台区卢沟桥南里 2 号
电话 ： 86-010-8321-2588 / 传真 ： 86-010-8321-2188
电子邮箱 ： robot_as@yahoo.com.cn
邮编 ：100072