

三菱 PLC 编程手册

目 录

第一章 FX1N PLC 编程简介

1.1 FX1N PLC 简介.....	1
1.1.1 FX1N PLC 的提出.....	1
1.1.2 FX1N PLC 的特点.....	1
1.1.3 FX1N PLC 产品举例.....	1
1.1.4 关于本手册.....	1
1.2 编程简介.....	1
1.2.1 指令集简介.....	2
1.2.2 资源集简介.....	7
1.2.3 编程及应用简介.....	9

第二章 基本逻辑指令说明及应用

2.1 基本逻辑指令一览表.....	10
2.1 [LD],[LDI],[LDP],[LDF], [OUT] 指令.....	10
2.2.1 指令解说.....	10
2.2.2 编程示例.....	10
2.3[AND], [ANI], [ANDP], [NDF] 指令.....	11
2.3.1 指令解说.....	11
2.3.2 编程示例.....	12
2.4 [OR],[ORI], [ORP],[ORF] 指令.....	13
2.4.1 指令解说.....	13
2.4.2 编程示例.....	13
2.5 [ANB],[ORB] 指令.....	14
2.5.1 指令解说.....	14
2.5.2 编程示例.....	14
2.6 [INV] 指令.....	15
2.6.1 指令解说.....	15
2.6.2 编程示例.....	15
2.7 [PLS],[PLF] 指令.....	16
2.7.1 指令解说.....	16
2.7.2 编程示例.....	17
2.8 [SET], [RST]指令.....	17
2.8.1 指令解说.....	17
2.8.2 编程示例.....	18
2.9 [NOP], [END]指令.....	18
2.9.1 指令解说.....	18
2.9.2 编程示例.....	18
2.10 [MPS], [MRD], [MPP] 指令.....	18
2.10.1 指令解说.....	18
2.10.2 编程示例.....	19
2.11[MC],[MCR]指令.....	21
2.11.1 指令解说.....	21

2.11.2 编程示例.....	21
第三章 步进顺控指令说明及应用	
3.1 步进顺控指令说明.....	22
3.1.1 指令解.....	22
3.1.2 编程示例.....	25
3.2 步进顺控指令应用.....	25
3.2.1 单一流程示例.....	25
3.2.2 选择性分支与汇合示例.....	26
3.2.3 并行分支与汇合示例.....	27
3.2.4 循环和跳转示例.....	29
第四章 功能指令说明及应用	
4.1 功能指令一览表.....	31
4.2 程序流程.....	33
4.2.1 条件跳转[CJ].....	33
4.2.2 子程序调用[CALL]	35
4.2.3 子程序返回[SRET]	35
4.2.4 主程序结束[FEND]	36
4.2.5 循环范围开始 [FOR]	37
4.2.6 循环范围结束 [NEXT]	37
4.3 传送与比较.....	38
4.3.1 比较指令[CM]	39
4.3.2 区域比较 [ZCP]	40
4.3.3 传送指令[MOV].....	41
4.3.4 反向传送 [CML]	43
4.3.5 BCD 转换 [BCD]	44
4.3.6 BIN 转换 [BIN]	45
4.4 四则逻辑运算.....	46
4.4.1 BIN 加法运算[ADD]	46
4.4.2 BIN 减法运算[SUB]	47
4.4.3 BIN 乘法运算[MUL]	48
4.4.4 BIN 除法运算 [DIV]	49
4.4.5 BIN 1 [INC].....	50
4.4.6 BIN 减 1 [DEC]	50
4.4.7 逻辑与[WAND]	51
4.4.8 逻辑或[WOR]	51
4.4.9 逻辑异或[WXOR].....	52
4.4.10 求补 [NEG].....	53
4.4.11 BIN 开方运算[SQR]	53
4.5 循环与移位.....	54
4.5.1 循环右移[ROR]	54
4.5.2 循环左移[ROL]	55
4.5.3 带进位循环右移 [RCR]	56
4.5.4 带进位循环左移[RCL]	58
4.6 浮点数运算.....	59

4.6.1 二进制浮点数比较「DECMP」	59
4.6.2 二进制浮点数区域比较[DEZCP]	60
4.6.3 二进制浮点数转十进制浮点数[DEBCD]	61
4.6.3 十进制浮点数转二进制浮点数[DEBIN]	62
4.6.5 二进制浮点数加法[DEADD]	62
4.6.6 二进制浮点数减法[DESUB]	63
4.6.7 二进制浮点数乘法「DEMUL」	64
4.6.8 二进制浮点数除法「DEDIV」	65
4.6.9 二进制浮点数开方「DESQR」	66
4.6.10 二进制浮点数转 BIN 整数变换「INT」	67
4.6.11 BIN 整数转二进制浮点数「FLT」	68
4.7 触点比较指令.....	69
4.7.1 接点比较指令「LD※」	69
4.7.2 接点比较指令「AND※」	70
4.7.3 接点比较指令「OR※」	72
4.8 功能指令的基本规则.....	73
4.8.1 . 功能指令的表示与执行形式.....	73
4.8.2 功能指令内的数值处理.....	75
4.8.3 利用变址寄存器的操作数修改.....	77
第五章 资源说明及应用	
5.1 变址寄存器 V、Z 说明及应用.....	80
5.1.1 变址寄存器 V、Z 说明.....	80
5.1.2 变址寄存器在梯形图中的应用.....	80
5.1.3 使用变址功能的注意事项.....	81
5.2 输入输出继电器 X、Y 说明及应用.....	82
5.2.1 输入输出继电器 X、Y 说明.....	82
5.2.2 输入输出继电器应用.....	83
5.3 辅助中间继电器 M 说明及应用.....	85
5.3.1 辅助中间继电器 M 说明.....	85
5.3.2 辅助中间继电器 M 应用.....	85
5.4 状态继电器 S 说明及应用.....	87
5.4.1 状态继电器 S 说明.....	87
5.4.2 状态继电器 S 应用.....	88
5.5 定时器 T 说明及应用.....	88
5.5.1 定时器 T 说明.....	88
5.5.2 定时器 T 应用.....	90
5.6 计数器 C 说明及应用.....	92
5.6.1 16 bit 计数器 C 说明.....	92
5.6.2 32 bit 计数器 C 说明.....	93
5.6.3 16 bit 计数器 C 应用.....	95
5.6.4 32 bit 计数器应用.....	96
5.7 数据寄存器 D 说明及应用.....	97
5.7.1 数据寄存器 D 说明.....	97
5.7.2 数据寄存器 D 应用.....	99

5.8 程序位置指针 P 说明及应用.....	100
5.8.1 程序位置指针 P 说明.....	100
5.8.2 程序位置指针 P 应用.....	100
5.9 常数标记 K、H 详细说明.....	102
5.9.1 常数标记 K.....	102
5.9.2 常数标记 H.....	103
5.10 特殊软元件说明.....	103
第六章 PID 指令说明及应用	
6.1 PID 运算.....	104
6.1.1.....	104
6.1.2 应用示例.....	110

第一章 FX1N PLC 编程简介

1.1 FX1N PLC 简介

1.1.1 FX1N PLC 的提出

基于以下观点，提出 FX1N PLC 的概念：

①、软件和硬件独立设计。

在规定好硬件和软件接口的前提下，各自独立设计，以提高开发效率。

②、简化硬件设计。

只需进行外形设计和电气接口设计，功能设计由软件设计取代。

③、简化软件设计。

依托功能强大的软件平台，只需设计个体产品与平台间的软件接口。

④、产品应用可二次编程。

根据工艺要求，用标准梯形图语言进行二次编程。

1.2.1 FX1N PLC 的特点

①、什么是 FX1N PLC？

将 PLC 语言（梯形图语言）嵌入到专用芯片中，获取了梯形图编程平台所提供的各种强大的应用功能。我们称用于 PLC 专用芯片产品开发，自身具有强大功能的梯形图语言编程软件为 FX1N PLc。FX1N PLc 能广泛应用于各种工业控制产品中。

②、FX1N PLC 产品有哪些特点？

利用 FX1N PLC 软件开发出的应用产品，我们称之为 FX1N PLC 产品。FX1N PLC 产品具有以下特点：

- 用梯形图语言编写应用程序。
- 能与多家人机界面连接，如台达、EView 等。
- 支持 CANBUS 网络结构。
- 与其它厂家 PLC 并联运行。

1.1.3 FX1N PLC 产品举例

①、可编程控制器 FX1N-40MR

本产品有开关量输入 24 点、开关量输出 16 点，除具有可编程逻辑控制功能之外。每台产品均支持人机界面。

②、空压机控制器

具有用户要求的外观和接口，用户可根据自己的意图，用梯形图编写不同的控制程序，便于工艺保密和系列产品的标准化。每台控制器均可支持人机界面。

③、供水控制器

预留较多的富余接口，可适应各种复杂的供水要求，是供水行业的通用型控制器。应用人员可用梯形图编写控制程序，满足用户的不同需要。该产品支持人机界面。

1.1.4 关于本手册

编写本手册的目的是帮助 FXZN PLC 产品的用户，正确使用梯形图语言编程，充分发挥 FXZN PLC 所提供的强大功能。

“编程

LDP 读取上升沿。

LDF 读取下降沿。

ANDP：上升沿接通，串联连接。

ANDF：下降沿接通，串联连接。

ORP：上升沿接通，并联连接。

ORF：下降沿接通，并联连接。

INV：运算触点取反。

MPS：压栈。

MRD：读栈。

MPP：出栈。

MC：主控。

MCR：主控结束。

NOP：空操作。

END：程序结束。

- 梯形图与指令表：

梯形图是电气控制的专业语言，方便编程人员编程。

专用芯片是按指令表执行控制。

梯形图与指令表二者自动相互转换。下例是二者相互转换示意图。

梯形图

想对基本逻辑指令进一步了解，请参看《第二章基本逻辑指令说明及应用》。

②、步进顺控指令：

- 助记符及名称：

STL：步进梯形图开始。仅对状态继电器 S。步序间状态转移必须使用 SETS，不能用 OUTS。

RET：步进梯形图结束。

- 梯形图与指令表：

梯形图是电气控制的专业语言，方便编程人员编程。

专用芯片是按指令表执行控制。

梯形图与指令表二者自动相互转换。下例是二者相互转换示意图。

想对步进顺控指令进一步了解，请参看《第三章步进顺控指令说明及应用》。

③、基本功能指令：

● 助记符及名称：

CJ：条件跳转。

CALL：子程序调用。

SRET：子程序返回。

FEN_n：主程序结束。

FOR：循环开始。

NEXT：循环结束。

CMP：比较。

ZCP：区域 L 比较。MOV：传送。

CML：取反传送。

BCD：BIN 向 BCD 转换。

BIN：BCD 向 BIN 转换。

ADD：加法。

SUB：减法。

MUL：乘法。

DIV：除法。

INC：自加 1 运算。

DEC：自减 1 运算。

WAND：字与运算（按位）。

WOR：字或运算（按位）。

WXOR：字异或运算（按位）。

NEG：取补运算。

SQR：开方运算。

ROR：循环右移。

ROL：循环左移。
RCR：带进位循环右移。
RCL：带进位循环左移。

DECMP：二进制浮点数比较。
DEZCP：二进制浮点数区域比较。
DEBCD：二进制浮点数向十进制浮点数转换。
DEBIN：十进制浮点数向二进制浮点数转换。
DEADD：二进制浮点数加法。
DESUB：二进制浮点数减法。
DEMUL：二进制浮点数乘法。
DEDIV：二进制浮点数除法。
DEsQR：二进制浮点数开方。

INT：二进制浮点数取整。
FLT：整数转换为二进制浮点数。

LD=：读取“等于 L 比较节点”。
LD>：读取“大于 L 比较节点”。
LD<：读取“小于 L 比较节点”。
LD<>：读取“不等于比较节点”。
LD<=：读取“小于等于比较节点”。
LD>=：读取“大于等于比较节点”。
AND=：串联“等于比较节点”。
AND>：串联“大于比较节点”。
AND<：串联“小于比较节点”。
AND<>：串联“不等于比较节点”。

AND<=：串联“小于等于比较节点”。
AND>=：串联“大于等于比较节点”。
OR=：并联“等于比较节点”。
OR>：并联“大于比较节点”。
OR<：并联“小于 L 比较节点”。
OR<>：并联“不等于比较节点”。
OR<=：并联“小于等于比较节点”。
OR>=：并联“大于等于比较节点”。

● 梯形图与指令表：

梯形图是电气控制的专业语言，方便编程人员编程。
专用芯片是按指令表执行控制。
梯形图与指令表二者自动相互转换。下例是二者相互转换示意图。

想对基本功能指令进一步理解，请参看《第四章基本功能指令说明及应用》。

④、专家功能指令：

- 助记符及名称：

PID : PID 控制算法。

- 梯形图与指令表：

梯形图是电气控制的专业语言，方便编程人员编程。

专用芯片是按指令表执行控制。

梯形图与指令表二者自动相互转换。下例是二者相互转换示意图。

想对专家功能指令进一步了解，请参看《第六章专家功能指令说明及应用》。

1.2.2 资源集简介

①、输入继电器 X：

扩展数量：128 点。

标号范围：X000 — X177；标号为 8 进制。

实际产品的数量和范围：由 FX1N PLC 产品确定。

如 K — 40MR，范围：X000 — X027，数量：24 点

②、输出继电器 Y：

扩展数量：128 点。

标号范围：Y000 — Y177；标号为 8 进制。

实际产品的数量和范围：由 FX1N PLC 产品确定。

如 K — 40MR，范围：Y000 — Y017，数量：16 点。

③、辅助继电器 M：

数量：1536 点

标号范围：M0 — M1535；

标号为十进制。一般用：M0 — M1023，计 1024 点。

停电保持用：M1024 — M1535，计 512 点。

④、状态继电器 S：

数量：1000 点

标号范围：S0 — S999；标号为十进制。

一般用：S0 — S499，计 500 点。

停电保持用：S500 — S999，计 500 点。

⑤、时间继电器 T：

数量：256 点

标号范围：T0 — T255；标号为十进制。三龙电子科技

一般用：T0 — T199, 100 ms 型，计 200 点

T200 — T245, 10 ms 型，计 46 点

累积用：T246 — T249, 1 ms 型，计 4 点

T250 — T255, 100 ms 型，计 6 点

累积用的时间继电器在停电时，计时数据保持，必须用 RST 清零。

⑥、计数器 C：

数量：256 点

标号范围：C0 — C199；C200 — C255；标号为十进制。

一般用：C0 — C99, 16 bit，计 100 点。

停电保持用 C100 -C199, 16 bit，计 100 点。

C200 -C255, 32bit 可逆计数器，计数方向由 M8200 — M8255 确定，ON 时减计数。

⑦、数据寄存器 D：

数量：6000 点

标号范围：D0 — D5999；标号为十进制。

一般用：D0 — D199，计 200

MS000：程序运行时 ON；

MS00Z：程序开运行时第一个扫描周期时 ON；M8020：零标志；

M8021：借位标志；

M8022：进位标志；

M8200—M8255：32 bit 逆计数器方向指定。

想对资源更进一步了解，请参看《第五章资源详细说明及应用》。

1.2.3 编程及应用简介

①、编程软件

- 梯形图编程软件 SLJWin：

支持梯形图编程、下载、监控，可对 FX1N PLC 产品设置加密口令。

- 网络设置软件上位机软件

支持网络构建、下载，经上位机软件设置的主节点与从节点能自动交换网络数据。网络构建支持第三方设备。

②、编程设备

个人计算机：

SLJWin 运行于 WindowS 操作系统。操作系统可以是：

Windows 95，Windows 98，Windows 2000，Windows XP。

③、编程及应用流程说明

-

第二章 基本逻辑指令说明及应用

2.1 基本逻辑指令一览表

助记符、名称 功能 可用软元件 程序步

LD 取 常开触点逻辑运算开始 X,Y,M,S,T,C 1

LDI 取反 常闭触点逻辑运算开始 X,Y,M,S,T,C 1

LDP 取脉冲上升沿 上升沿检出运算开始 X,Y,M,S,T,C 2

LDF 取脉冲下降沿 下降沿检出运算开始 X,Y,M,S,T,C 2

AND 与 常开触点串联连接 X,Y,M,S,T,C 1

ANI 与非 常闭触点串联连接 X,Y,M,S,T,C 1

ANDP 与脉冲上升沿 上升沿检出串联连接 X,Y,M,S,T,C 2

ANDF 与脉冲下降沿 下降沿检出串联连接 X,Y,M,S,T,C 2

OR 或 常开触点并联连接 X,Y,M,S,T,C 1

ORI 或非 常闭触点并联连接 X,Y,M,S,T,C 1

ORP 或脉冲上升沿 上升沿检出并联连接 X,Y,M,S,T,C 2

ORF 或脉冲下降沿 下降沿检出并联连接 X,Y,M,S,T,C 2

ANB 块与 并联回路块的串联连接 1

INV 取反 运算结果的反转 1
NOP 空操作 无动作 1
END 结束 输入输出及返回到开始 1

- 软元件为 Y 和一般 M 的程序步为 1, S 和特殊辅助继电器 M 的程序步为 2, 定时器 T 的程序步为 3, 计数器 C 的程序步为 3-5。
- 软元件为 Y 和一般 M 的程序步为 1, S 和特殊辅助继电器 M、定时器 T、计数器 C 的程序步为 2, 数据寄存器 D 以及变址寄存器 V 和 Z 的程序步为 3。

2.2 [LD],[LDI],[LDP],[LDF],[OUT]指令

2.2.1 指令解说

助记符、名称 功能 可用软元件 程序步

LD 取 常开触点逻辑运算开始 X, Y, M, S, T, C 1

LDI 取反 常闭触点逻辑运算开始 X, Y, M, S, T, C 1

LDP 取脉冲上升沿 上升沿检出运算开始 X, Y, M, S, T, C 2

LDF 取脉冲下降沿 下降沿检出运算开始 X, Y, M, S, T, C 2

OUT 输出 线圈驱动 Y, M, S, 主 C 见说明

- LD, LDI, LDP, LDF 指令将触点连接到母线上。多个分支用 ANB, ORB 时也使用。LDP 指令在上升沿（软元件由 OFF 到 ON 变化时）接通一个周期；LDF 指令在下降沿（软元件由 ON 到 OFF 变化时）接通一个周期。
- LD, LDI, LDP, LDF 指令的重复使用次数在 8 次以下。即与后面的 ANB, ORB 指令使用时串并连使用的最多次数

```

0 LD X000
1 OUT Y000
2 OUT C0 K10
5 LDI X001
6 OUT Y001
7 OUT T0 K100
10 LD C0
11 OUT Y002
12 LD T0
13 OUT Y003
14 LDP X002
16 OUT M2
17 LDF X003
19 OUT M3
20 END

```

用 LD , LDI , LDP , LDF 指令与母线连接。输出使用 OUT 指令驱动线圈。使用 OUT 指令驱动定时器的计时线圈或者计数器的计数线圈时，必须设定定时和计数的时间和计数的值，可以是常数 K ， 或者由数据寄存器间接指定数值。每个程序结束必须要有 END 指令，关于 END 指令详见后面的 END 指令介绍。

2.3 [AND], [ANI],[ANDP],[ANDF] 指令

2.3.1 指令解说

助记符、名称 功能 可用软元件 程序步

AND 与 常开触点串联连接 X , Y , M , S , T , C 1

ANI 与非 常闭触点串联连接 X , Y , M , S , T , C 1

ANDP 与脉冲上升沿 上升沿检出握 马联连接 X , Y , M , S , T , C 2

ANDF 与脉冲下降沿 下降沿检出握 马联连接 X , Y , M , S , T , C 2

- AND , ANI , ANDP , ANDF 指令只能串接一个触点，两个以上的并联回路串联时使用后面的 ANB 指令。串联次数不受限制。
- ANDP , ANDF 指令在上升沿（即软元件由 ON 到 OFF 变化时）和下降

14 END

- 实例中 x001 , x003 , Y001 作为串联触点与前面的触点相连。

2.4 [OR],[ORI],[ORP],[ORF]指令

2.4.1 指令解说

助记符、名称 功能 可用软元件 程序步

OR 或 常开触点并联连接 X,Y,M,S,T,C 1

ORI 或非 常闭触点并联连接 X,Y,M,S,T,C 1

ORP 或脉冲上升沿 上升沿检出并联连接 X,Y,M,S,T,C 2

ORF 或脉冲下降沿 下降沿检出并联连接 X,Y,M,S,T,C 2

- OR , ORI , ORP , ORF 指令只能并联一个触点，两个以上的串联回路并联时使用后面的 ORB 指令。
- ORP , ORF 指令在上升沿（即软元件由 OFF 到 ON 变化时）和下降沿（即软元件由 ON 到 OFF 变化时）接通一个周期。
- OR , ORI , ORP , ORF 指令和前面的 LD , LDI , LDP , LDF 指令一起使用，并联次数不受限制。

2.4.2 编程示例

```
0 LD X000
1 ORP X001
3 ORI M0
4 OUT Y000
5 LD X002
6 ORF X010
8 ANI X003
9 ORI X011
10 AND X004
```

11 OR X012
12 LDI X005
13 ORF X013
15 AND X006
16 ORI X014
17 ANB
18 OUT Y001
19 END

- 使用 OR , ORI , ORP , ORF 与前面的 LD , LDI , LDP , LDF 并联连接，在程序步 12 到 16 中，由于是两个并联回路块的串联，所以使用 ANB 指令，关于 ANB 指令详见后面的说明。

2.5 [ANB],[ORB] 指令

2.5.1 指令解说

助记符、名称 功能 可用软元件 程序步

ANB 块与 并联回路块的串联连接 1

ORB 块或 串联回路块的并联连接 1

- 当多分支回路与前面的回路串联连接时，使用 ANB 指令。分支以 LD , LDI , LDP , LDF 指令作为起点，使用 ANB 指令与前面以 LD , LDI , LDP , LDF 指令作为起点的分支串联连接。
- 当 2 个以上的触点串接的串联回路块并联连接时，每个分支使用 LD , LDI 指令开始，ORB 指令结束。
- ANB , ORB 指令都是不带软元件的指令。
- ANB , ORB 使用的并串联回路的个数不受限制，但是当成批使用时，必须考虑 LD . LDI 的使用次数在 8 次以下。

2.5.2 编程示例

0 LD X000
1 ANI X001
2 LDI X002
3 AND X003
4 ORB
5 LD X004
6 AND X005

```
7 ORB
8 OUT Y000
9 LD X006
10 OR X007
11 LD X010
12 ANI X011
13 LDI X012
14 AND X013
15 ORB
16 ORI X014
17 ANB
18 OR X015
19 OUT Y001
20 END
```

- 在每个分支的最后使用 **ORB** 指令，不要在所有的分支后面使用 **ORB** 指令，如程序步 4 和 7 所示。
- **ORB** 和 **ANB** 指令只是对块的连接，如果不是块就不能使用，如程序步 16 和 18 不是块就不能使用。如图所示，串联回路块和并联回路块的示例。

2.6[INV] 指令

2.6.1 指令解说

助记符、名称	功能	可用软元件	程序步
INV 取反	运算结果的反转	1	

- **INV** 指令是将 **INV** 指令之前，**LD** , **LDI** , **LDP** , **LDF** 指令之后的运算结果取反的指令，没有软元件。

2.6.2 编程示例

```
0 LD X000
1 INV
2 OUT Y000
3 LDI X001
4 INV
5 INV
6 OUT Y001
7 END
```

INV 指令的动作范围如下图

2.7[PLS],[PLF]指令

2.7.1 指令解说

助记符、名称 功能 可用软元件 程序步

PLS 上升沿脉冲 上升沿输出 Y,M (特殊 M 除外) 1

PLF 下降沿脉冲 下降沿输出 Y,M (特殊 M 除外) 1

- 使用 PLS 指令时，只在线圈由 OFF 变成 ON 的一个扫描周期内，驱动软元件。。使用 PLF 指令时，只在线圈由 ON 变成 OFF 的一个扫描周期内，驱动软元件。。对具有停电保持功能的软元件，它只在第一次运行时产生脉冲动作。

2.7.2 编程示例

```
0 LD X000
1 PLS M0
3 LD M0
4 SET Y000
5 LD X000
6 PLF M1
8 LD M1
9 RST Y000
10 LDP X001
12 OUT M2
13 LD M2
14 SET Y001
15 LDF X001
17 OUT M3
18 LD M3
19 RST Y001
```

20 END

- 程序段。— 2 和 10 — 12 的动作相同，都是在线圈闭合的上升沿，驱动一个扫描周期的输出。同样，程序段 5 — 7 和 15 — 17 的动作相同，都是在在线圈闭合的下降沿，驱动一个扫描周期的输出。
- 关于 SET , RST 指令的作用详见后面的说明。

2.8 [SET],[RST]指令

2.8.1 指令解说

助记符、名称 功能 可用软元件 程序步

SET 置位 动作保持 Y, M, S 见说明

RST 复位 清除动作保持，寄存器清零 Y, M, S，主 C, D, V, Z

- 软元件为 Y 和一般 M 的程序步为 1, S 和特殊辅助继电器 M、定时器 T、计数器 C 的程序步为 2，数据寄存器 D 以及变址寄存器 V 和 Z 的程序步为 3。SET 指令在线圈接通的时候就对软元件进行置位，只要置位了，除非用 RST 指令复位，否则将保持为 1 的状态。同样，对 RST 指令只要对软元件复位，将保持为 0 的状态，除非用 SET 指令置位。
- 对同一软元件，SET , RST 指令可以多次使用，顺序随意，但是程序最后的指令有效。
- RST 指令可以对数据寄存器 (D)，变址寄存器 (Z)，定时器 (T) 和计数器 (C)，不论是保持还是非保持的都可以复位置零。

2.8.2 编程示例

```
0 LD X000
1 SET Y000
2 LDI X001
3 RST Y000
4 LDP X001
6 SET Y001
7 LDF X001
8 RST Y001
10 END
```

2.9 [NOP],[END]指令

2.9.1 指令解说

助记符、名称 功能 可用软元件 程序步

NOP 空操作 无动作 1

END 结束 输入输出及返回到开始 1

- 程序清除时指令变为 NOP 指令，指令之间加入 NOP 指令，程序对他不做任何事情，继续向下执行，只是增加了程序的步数。
- 每个程序必须有一个且只有一个 END 指令，表示程序的结束。PLC 不断反复进行如下操作：输入处理，从程序的 O 步开始执行直到 END 指令，程序处理结束，接着进行输出刷新。然后开始循环操作。

2.9.2 编程示例

```
0 LD X000
1 AND X001
2 OUT Y000
3 NOP
4 NOP
5 LDI X002
6 ANI X003
7 OUT Y001
8                               END
```

2.10[MPS],[MRD],[MPP]指令

2.10.1 指令解说

助记符、名称 功能 可用软元件 程序步

MPS 压栈 运算存储 1

MRD 读栈 存储读出 1

MPP 出栈 存储读出与复位 1

- 嵌入式 PLC 中有 H 个栈空间，也就是说可以压栈的最大深度为 H 级。每使用一次 **MPS** 将当前结果压入第一段存储，以前压入的结果依次移入下一段。**MPP** 指令将第一段读出，并且删除它，同时以下的单元依次向前移。**MRD** 指令读出第一段，但并不删除它。其他单元保持不变。使用这三条指令可以方便多分支的编程。在进行多分支编程时，**MPS** 保存前面的计算结果，以后的分支可以利用 **MRD** , **MPP** 从栈中读出前面的计算结果，再进行后面的计算。最后一个分支必须用 **MPP**，保证 **MPS** , **MPP** 使用的次数相同。注意，使用 **MPP** 以后，就不能再使用 **MRD** 读出运算结果，也就是 **MPP** 必须放在最后的分支使用。
- **MRD** 指令可以使用多次，没有限制。**MPS** 连续使用的最多次数为 H，但是可以多次使用。每个 **MPS** 指令都有一个 **MPP** 指令对应，**MPP** 的个数不能多于 **MPS** 的个数。

2.10.2 编程示例

实例 1

```
0 LD X000
1 MPS
2 AND X001
3 OUT Y000
4 MRD
5 ANI X002
6 OUT Y001
7 MPP
8 OUT Y002
9 AND X003
10 OUT Y003
11 END
```

```
10 LD X012
11 ANI X013
12 ORB
13 ANB
14 OUT Y005
15 MPP
16 AND X014
17 OUT Y006
18 MPS
19 LDI X015
20 OR X016
21 ANB
22 OUT Y007
23 MPP
24 AND X017
25 OUT Y010
26 END
```

该实例使用一级两段堆栈，并且跟 OR , ORB , ANB 指令混合使用。

实例 3

```
0 LD X000
1 MPS
2 ANI X001
3 MPS
4 ANI X002
5 MPS
6 AND X003
7 OUT Y000
8 MPP
9 ANI X004
10 OUT Y001
11 MPP
12 ANI X005
13 AND X006
14 OUT Y002
15 MPP
16 AND X007
17 MPS
18 ANI X010
19 OUT Y003
20 MPP
21 AND X011
22 OUT Y004
```

23 END

该实例使用三级堆栈，即堆栈嵌套三级。

2.11[MC],[MCR] 指令

2.11.1 指令解说

助记符、名称 功能 可用软元件 程序步

MC 主控 公共串联点的连接线圈指令 Y, M (特殊 M 除外) 3

MCR 主控复位 公共串联点的消除指令 2

- 当前面的触点接通时,就执行 MC 到 MCR 的指令。执行 MC 指令时,母线向 MC 触点后移动,执行 MCR 指令返回母线。
- 使用 MC 指令时,嵌套级 N 的编号按顺序依次增大,也就是说只有使用 NO,才能嵌套 NI。相反使用 MCR 指令时,必须从大往小返回母线。最大嵌套级数为 7 级 (N6)。
- 通过不同的软元件 Y, M, 可以多次使用 MC 指令,如果使用相同的软元件,将同 OUT 指令一样,会出现双线圈输出。

2.11.2 编程示例

- 该实例只使用一个 MC, MCR 指令,嵌套级数也是 1, 可以进行 7 级嵌套。该实例中当 xo00 接通时,执行 MC, MCR 之间的指令,当 xo00 断开时,成为如下两种形式。

现状保持: 累积定时器的值, 计数器的值, 用 SET 或 ST 指令驱动的软元件。变为断开的元件: 非累积定时器的值, 用 OUT 指令驱动的软元件。

0 LD M80

检查出错。

状态转移只能用 SET 指令，不能用 OUT 指令。

使用 OUTS 时，S 作为辅助继电器使用，而不是状态寄存器。

时间继电器 T 可重复使用，但相邻两个状态不能重复使用同一时间继电器。

- 两个矛盾继电器输出时，必需加软件互锁。考虑软件快于硬件，相矛盾的硬件输出也必需互锁。
- 允许同一继电器在不同状态下输出，其实际输出视状态转移的位置确定。

3.2 步进顺控指令应用

3.2.1 单一流程示例

示例说明：

该程序描述一个自行葫芦自进入工位到走出工位的步序过程，若在葫芦升降过程中发生停电，来电后继续停电前的动作，并保证升或降动作总时间不变。

5500 — 5503 为停电保持型状态寄存器；C100 — C101 为停电保持型计数器；TO 延时 2 秒，作信号确认用；T1 作为 500 ms 脉冲发生器；

xo = 0N 时，表示工位上停有自行葫芦；

TO = 0N 时，表明工位上无自行葫芦；因信号由滑触线供给，因而 XO = OFF 时，不一定确定工位无车，需延时确认。

YO 为驱动进车；Y1 驱动葫芦下降；Y2 驱动葫芦上升。

3.2.2

若规定的次数内密码正确，进入密级操作。

流程示意图：钥匙进入 XO = ON

梯形图：

第四章 功能指令说明及应用

4.1 功能指令一览表

分类 指令助记符 功能 页码

程序流程 CJ 条件跳转

CALL 子程序调用

SRET 子程序返回

FEND 主程序结束

FOR 循环范围开始

NEXT 循环范围结束

传送也比较 CMP 比较

ZCP 区域比较

MOV 传送

CML 反向传送

BCD BCD 转换

BIN BIN 转换

四则逻辑运算 ADD BIN 加法

SUB BIN 减法

MUL BIN 乘法

RCL 带进为循环左移

基本指令一览表（续）

分类 指令助记符 功能 页码

浮

点

数

比

- 较 ECMP 2 进制浮点数比较
- EZCP 2 进制浮点数区域比较
- EBCD 2 进制浮点数转 10 进制浮点数
- EBIN 10 进制浮点数转 2 进制浮点数
- EADD 2 进制浮点数加法
- ESUB 2 进制浮点数减法
- EMUL 2 进制浮点数乘法
- EDIV 2 进制浮点数除法
- ESQR 2 进制浮点数开方
- INT 2 进制浮点数转 BIN 整数
- FLT BIN 整数转 2 进制浮点数

节

点

比

较 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)

在上图示例中，如果 X000“ON”，则从 0 步跳到 23 步（标记 P0 的后一步）。X000“OFF”时，不进行跳转，顺序执行。

当 X000“ON”时，进行跳转，跳转中的线圈动作如下：

- Y、M、S 保持以前动作；
- T 在跳转前若没有触发，定时器也不动作。若被触发，时钟继续运行，但触点不动作，当 X000“OFF”时，触点立即动作；
- C 在跳转前若没有触发，跳转后即使触发，计数器不动作。若被触发，计数中断，当 X000“OFF”时继续计数；
- 功能指令跳转后不动作；
- 定时器及计数器的复位指令在跳转外时，技术线圈及跳转的计数线圈复位（接点复位及当前值的清除）有效。
- 对 END 步跳转，需标明标号（P0～P127 都可以），线圈动作如上。
- 主控制指令和跳转指令的关系及动作如下，

从 MC 外向 MC 内跳转时，与 MC 的动作无关，即使 M100 处于“OFF”状态下，P0 以下 M100 视为“ON”；

从 MC 内向 MC 外内跳转时，M100 处于“OFF”状态下，不能跳转；

从 MC 内向 MC 外内跳转时，M100 处于“OFF”状态下，不能跳转，当 M0“ON”时，可跳转，但 MCR 无效

4.2.2 子程序调用『CALL』

16 位指令 CALL（连续执行型）

3 步 CALLP（脉冲执行型）

4.2.3 子程序返回『SRET』

单独指令 SRET

1 步 不需要触点驱动的指令

适用

软元

件 子程序调用的指针（P）

P0～P127；

指针编号可做变址修改；

嵌套最多可为 5 层；

对子程序返回无适用软元件。

功能和动作

- 若 X001“ON”，则执行调用指令跳转到标记 P11 步，执行完通过执行 SRET 指令返回原来的步，再往下执行；
- 在 FEND 指令后对标记（子程序）编程；
- CJ 指令的标记和子程序的标记不能重复编号；
- 在子程序内最多可以允许有四层嵌套，如上例，还可增加 2 层，整体而言可做 5 层；
- 指针编号可作变址修改，如 P0 Z（0+12=12），如果变址得出的编号没有，嵌入式 PLC 停止工作。

4.2.4 主程序结束『FEND』

3 步

适用

软元

件 •字软元件 K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z，
可作变址修改

4.2.6 循环范围结束『NEXT』

单一指令 NEXT

1 步 不需要触点驱动指令

功能和动作

只在 FOR~NEXT 指令之间的处理（利用源数据指定的次数），执行完后，才处理 NEXT 指令以后的程序。

- 上图是通电时对保持用辅助继电器复位的程序；
- 从 4 步至 25 步之间的程序执行了 16 次，执行完后 Z 的值为 512；
- FOR~NEXT 嵌套最多 5 层；
- 循环次数多次周期会延长，请务必注意；
- NEXT 指令在 FOR 指令之前，或无 NEXT 指令，或在 FEND、END 指令以后有 NEXT 指令，或 FOR 指令与 NEXT 个数不相等，都会出错；
- 若不想执行 FOR~NEXT 之间的程序时，利用 CJ 指令，使之跳转。如在上图所例，在 25 步前插入 LDI M0 CJ P50 则 Z 的值为 32，即只执行了一次。

4.3 传送与比较

4.3.1 比较指令『CMP』

16 位指令 CMP（连续执行型）

Z

位软元件 (Y、M、S)

功能和动作

- 上图示例是 D0 的内容与常数 100 进行比较，大小比较是按代数形式进行的 ($-8 < X$)；
- 所有数据都以 2 进制值处理；
- 当 $D0 > 100$ ，M0“ON”，当 $D0 = 100$ ，M1“ON”，当 $D0 < 100$ ，M2“ON”；
- 目标地址指定 M0，则 M1、M2 被自动占用；
- 当 X001“OFF”时，M0、M1、M2 仍保持以前状态。如当 D0 的内容为 50，则 $50 < 100$ ，M2“ON”，M0、M1 都“OFF”，X001“OFF”时，M2 仍“ON”。

指令不执行时，想要清楚比较结果，可使用复位指令。

4.3.2 区域比较『ZCP』

16 位指令 ZCP（连续执行型）

7 步 ZCPP（脉冲执行型）

32 位指令 DZCP（连续执行型）

13 步 DZCPP（脉冲执行型）

适用软元件 S、字软元件（S1、S2、S）K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z

、位软元件（D）Y、M、S

功能和动作

- 如上例，D3 的内容与 D1、D2 的内容进行比较；
- D1 的内容应小于等于 D2 的内容，若 D1=100，D2=80，比较时 D2 的内容为 100；
- 按代数形式进行比较（-8<0）；
- 当 D1>D3，则 M3“ON”；当 D1≤D3≤D2，则 M4“ON”，当 D2<D3，则 M5“ON”。

4.3.3 传送指令『MOV』

16 位指令 MOV（连续执行型）

7 步 MOVP（脉冲执行型）

32 位指令 DMOV（连续执行型）

13 步 DMOVP（脉冲执行型）

适用软

- 将源 (S•) 向目标 (D•) 传送, X003“OFF”时, 目标 (D•) 的内容不变化;
- 常数 K100 被自动转换成 BIN 码。

《定时、计数器的当前值读出示例》

- 将 T0 当前值传送给 D20。

《定时、计数器设定值间接指定示例》

- T20 定时时间为 5 秒

《位软元件的传送》

- 上图示例可用下面的 MOV 指令来实现

《32 位数据的传说》

运算结果是 32 位的应用指令 (MUL 等)、32 位数据、32 位软元件或 32 位计数器等 32 位数据的传送, 必须适用 DMOV 指令。

- 上例将 (D11、D10) 的内容传送给 (D21、D20),
(C235 的当前值) 传送给 (D31、D30)。

4.3.4 反向传送『CML』

16 位指令 CML (连续执行型)

5 步 CMLP (脉冲执行型)

功能和动作 使数据反向传送的指令

- 将 D0 的内容每位取反（0 取反为 1，1 取反为 0）后，传送到目标地址，常数 K 被自动转换成 2 进制。
如：

D10

- 上例可用 CML 指令来实现。

4.3.5 BCD 转换『BCD』

16 位指令 BCD（连续执行型）

5 步 BCDP（脉冲执行型）

32 位指令 DBCD（连续执行型）

9 步 DBCDP（脉冲执行型）

适用软元件 •字软元件（S•）KnY、KnM、KnS、T、C、D、V、Z

•字软元件（D•）KnY、KnM、KnS、T、C、D、V、Z

功能和动作 将源（BIN）转换为目标（BCD）的指令。

- 使用 BCD、BCDP 指令，转换结果不能超出 0～9999，使用 DBCD，DBCDP 指令，转换结果不能超出 0～99999999；
- 将 PLC 内的 2 进制数变为七段显示等的 BCD 码向外部输出时使用。

4.3.6 BIN 转换『BIN』

32 位指令 DBIN（连续执行型）

9 步 DBINP（脉冲执行型）

适用软元件 S;字软元件（S;）KnX、KnY、KnM、KnS、T、C、D、V、Z
;字软元件（D;）KnY、KnM、KnS、T、C、D、V、Z

功能和动作 将源（BCD）转换为目标（BIN）的指令。

- 使用 BIN、BINP 指令，源数据（S;）不能超出 0～9999，使用 DBIN、DBINP 指令，源数据（S;）不能超出 0～99999999；
- 常数 K 能自动转换成 2 进制。

4.4 四则逻辑运算

4.4.1 BIN 加法运算『ADD』

16 位指令 ADD（连续执行型）

5 步 ADDP（脉冲执行型）

32 位指令 DADD（连续执行型）

13 步 DADDP（脉冲执行型）

适用软元件 S;字软元件（S;）K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z
;字软元件（D;）KnY、KnM、KnS、T、C、D、V、Z

标志位 零 M8020

借位 M8021

进位 M8022

功能和动作

- 两个源数据进行加法后传送到目标处，个数据的最高位是符号位（正数为 0，负数为 1），数据以代数形式进行加法运算($8+(-8)=0$)。
- 运算结果为 0 时，0 标志位 M8020 动作；运算结果超出 32767（16 位运算）或 2147483647（32 位运算）时，进位标志位 M8022 动作运算结果小于-32768（16 位运算）或-2147483648（32 位运算）时，借位标志位 M8021 动作；

- 对于脉冲型指令，每出现一次 OFF 到 ON 的变化，操作数做一次运算，
- 可以将源（S#）和目标（D#）指定为相同的软元件编号。这种情况下，如使用连续执行型指令（ADD、DADD），则每个扫描周期加一次，请务必注意。

4.4.2 BIN 减法运算『SUB』

16 位指令 SUB（连续执行型）

7 步 SUBP（脉冲执行型）

32 位指令 DSUB（连续执行型）

13 步 DSUBP（脉冲执行型）

适用软元件 字软元件（S1、S2） K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z

适用软元件 S1、S2、K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z（V、Z 仅限 16 位计算）

字软元件（D）KnY、KnM、KnS、T、C、D、V、Z

功能和动作

《16 位运算》

《32 位运算》

- 32 位运算不能指定 Z 作为（D）；
- 除数作为 0 时，如果被除数为整数，商为 32767（16 位）或 2147483647（32 位）；如果被除数为 0，商为 0；如果被除数为负数，商为-32768（16 位）或-2147483648（32 位）；
- 商和余数的最高位为符号位，0 为正，1 为负，当被除数或除数中的一方为负数时，商为负，当别除数为负时，余数则为负。

4.4.5 BIN 增 1『INC』

16 位指令 INC（连续执行型）

3 步 INCP（脉冲执行型）

32 位指令 DINC（连续执行型）

5 步 DINCP（脉冲执行型）

适用软元件 S、字软元件（D）KnY、KnM、KnS、T、C、D、V、Z

5 步 DDECP（脉冲执行型）

适用软元件 •字软元件（D•）KnY、KnM、KnS、T、C、D、V、Z

功能和动作

- X001 每置“ON”一次，D0 的内容减 1，在连续执行指令中，每个扫描周期执行减 1 运算，所以务必引起注意；
- -32768 或-2147483648 减 1 变为 32767 或 2147483647，标志位不动作

4.4.7 逻辑与『WAND』

16 位指令 WAND（连续执行型）

7 步 WANDP（脉冲执行型）

32 位指令 DWAND（连续执行型）

13 步 DWANDP（脉冲执行型）

功能和动作

《逻辑与》

三龙电子科技

4.4.8 逻辑或『WDR』

16 位指令 WDR（连续执行型）

7 步 WDRP（脉冲执行型）

32 位指令 DW

适用软元件 S1~S2、K、H、KnX、KnY、KnM、KnS、T、C、D、V、Z

~字软元件（D~）KnY、KnM、KnS、T、C、D、V、Z

功能和动作

《逻辑异或》

4.4.10 求补『NEG』

16 位指令 NEG（连续执行型）

3 步 NEGP（脉冲执行型）

32 位指令 DNEG（连续执行型）

5 步 DNEGP（脉冲执行型）

适用软元件 S1~S2、字软元件（D~）KnY、KnM、KnS、T、C、D、V、Z

功能和动作

- 将（D~）指定的软元件内容中各位先去反（0 变 1，1 变 0），然后再加 1，将其结果存入原来的软元件中；
- 使用连续执行指令则在每一个扫描周期执行一次，务必引起注意。

4.4.11 BIN 开方运算『SQR』

16 位指令 SQR（连续执行型）

5 步 SQRP（脉冲执行型）

9 步 DSQRP（脉冲执行型）

适用软元件 •字软元件（S1•）K、H、D
•字软元件（D•）D

功能和动作

- 进行开方运算的指令；
- 仅在（S1•） ≥ 0 时有效，如果（S1•） < 0 ，结果为 0；
- 运算结果舍去小数取整数；
- 无标志位。

4.5 循环与移位

4.5.1 循环右移〔ROR〕

16 位指令 ROR（连续执行型）

5 步 RORP（脉冲执行型）

32 位指令 DROR（连续执行型）

9 步 DRORP（脉冲执行型）

适用软元件 •字软元件（D•）KnY、KnM、KnS、T、C、D、V、Z

•字软元件（n）K、H

回转量：n ≤ 16 （16 位指令）

n ≤ 32 （32 位指令）

16 位指令无

32 位指令 DEBCD（连续执行型）

9 步 DZEBCDP（脉冲执行型）

适用软元件 $\#8226$; 字软元件 (S $\#8226$;) D

$\#8226$; 位软元件 (D $\#8226$;) Y、M、S (D $\#8226$;占有连续的 3 点)

功能和动作

- 该指令把二进制浮点数的源操作数转换为十进制浮点数的目标操作数。
- 浮点数的运算在嵌入式 PLC 内部是以二进制浮点数为基础执行的。
- 二进制浮点数数据格式：尾数部分 23 位，指数部分 8 位，符号位 1 位。
- 二进制浮点数所表示的范围：最小绝对值 1×10^{-37} ，最大绝对值 1×10^{37} 。
- 十进制浮点数数据格式：尾数部分为低 16 位，指数部分为高 16 位。
- 十进制浮点数所表示的范围：尾数的输入范围任意，但是不能超过二进制浮点数所能表示的范围，即最小绝对值 1×10^{-37} ，最大绝对值 1×10^{37} ，指数 $= -37 \sim +37$ 。

4.6.4 十进制浮点数转二进制浮点数 [DEBIN]

4.6.8 二进制浮点数除法「DEDIV」

16 位指令 无

32 位指令 DEDIV（连续执行型）

13 步 DEDIVP（脉冲执行型）

适用软元件 S1、S2；字软元件（S1、S2）K、H、D

（D）D

位软元件

功能和动作

- 两个二进制浮点数源操作数相除的结果作为二进制浮点数存入目的地址中。
- 将常数 K、H 作为源操作数时，自动转换为二进制浮点数处理。
- 正确运算的数据范围：最小绝对值 1×10^{-37} ，最大绝对值 1×10^{37} 。不论是源操作数还是目的操作数和中间结果都不能超过此范围，否则导致运算结果不正确。因此也不存在零的二进制浮点数，当除数 S2 为 0 时，则运算结果错误。

- 当 X000 处于“ON”时，且计数器 C10 的当前值等于 100 时，驱动 Y3。
- 当 X001 处于“ON”，时，且 DO 的内容不等于 10 时，置位 Y4。
- 当 X002 处于“ON”且（D3, D2）的内容大于 13579 时，或者 M100 处于“ON 时，驱动 Y5。

注意事项

- 当源数据的最高位（16 位指令：b15,32 位指令：b31）为 1 时，将该数值作为负数进行比较；
- 32 计数器（CZ00 — C255）的比较，必须以 32 位指令来进行。

