

三菱 PLC 编程口通讯协议

一、三菱 PLC 编程口通讯协议

三菱 PLC 编程口的通讯协议比较简单，只有四个命令，即：

命令	命令码	目标设备
DEVICE READ CMD	"0"	X, Y, M, S, T, C, D
DEVICE WRITE CMD	"1"	X, Y, M, S, T, C, D
FORCE ON CMD	"7"	X, Y, M, S, T, C
FORCE OFF CMD	"8"	X, Y, M, S, T, C

五个标示：

ENQ	05H	请求
ACK	06H	PLC 正确响应
NAK	15H	PLC 错误响应
STX	02H	报文开始
ETX	03H	报文结束

使用累加方式的和校验，帧格式如下：

STX CMD DATA DATA ETX SUM(upper) SUM(lower)

和校验：

SUM= CMD+.....+ETX。 如 SUM=73H, SUM= "73" 。

1、DEVICE READ (读出软设备状态值)

计算机向 PLC 发送：

始	命令	首地址	位数	终	和校验
STX	CMD	GROUP ADDRESS	BYTES	ETX	SUM

PLC 返回

STX 1ST DATA 2ND DATA LAST DATA ETX SUM

2、DEVICE WRITE (向 PLC 软设备写入值)

计算机向 PLC 发送:

始 命令 首地址 位数 数据 终 和校验

PLC 返回

ACK (06H) 接受正确

NAK (15H) 接受错误

3、位设备强制置位/复位

FORCE ON 置位

始 命令 地址 终 和校验

STX CMD ADDRESS ETX SUM

02h 37h address 03h sum

FORCE OFF 复位

始 命令 地址 终 和校验

STX CMD ADDRESS ETX SUM

02h 38h address 03h sum

PLC 返回

ACK(06H) 接受正确

NAK(15H) 接受错误

以上可以看出，协议非常简单，但是由于没有寄存器类型信息，所以地址的计算十分关键，如 D100 和 M100 分别对应哪个地址呢？下面就是三菱 Fx 系列 PLC 地址对应表：

Public Const PLC_D_Base_Address = 4096

```
Public Const PLC_D_Special_Base_AddRes = 3584

Public Const PLC_Y_Group_Base_AddRes = 160

Public Const PLC_PY_Group_Base_AddRes = 672

Public Const PLC_T_Group_Base_AddRes = 192

Public Const PLC_OT_Group_Base_AddRes = 704

Public Const PLC_RT_Group_Base_AddRes = 1216

Public Const PLC_M_SINGLE_Base_AddRes = 2048(命令为 7 或 8 时)

Public Const PLC_M_Group_Base_AddRes = 256

Public Const PLC_PM_Group_Base_AddRes = 768

Public Const PLC_S_Group_Base_AddRes = 0

Public Const PLC_X_Group_Base_AddRes = 128

Public Const PLC_C_Group_Base_AddRes = 448

Public Const PLC_OC_Group_Base_AddRes = 960

Public Const PLC_RC_Group_Base_AddRes = 1472

Public Const PLC_TV_Group_Base_AddRes = 2048

Public Const PLC_CV16_Group_Base_AddRes = 2560

Public Const PLC_CV32_Group_Base_AddRes = 3072
```

当我们用 DEVICE READ 命令时，D100 地址=100*2+4096；M100 地址=100+256；不同的是 D 类型寄存器存放的是字，M 寄存器存放的是位，同样是读两个字节，D100 返回的就是 PLC 中 D100 地址的值，M 类型寄存器返回的是 M100 到 M116 的值。所以当我们用 FORCE ON 命令时，M100 寄存器地址=100+2048；

这也没有什么复杂的，不是吗？可是三菱公司好像不甘于如此，FORCE ON/Off 命令中地址排列与 DEVICE READ/WRITE 不同，是低位在前高位在后。如 Y20，地址是 0510H，代码中 4 个字节地址表示为：1005。（注意：Y 寄存器为八进制，如 Y20 地址=16+1280=0510H）

