

三菱 FX 系列 PLC 与计算机无协议通讯

云南省电子信息技工学校 王波

[摘 要]

本文主要通过介绍 PLC 通讯的意义和三菱 FX 系列 PLC 的四种通讯方式，并重点介绍 FX 系列 PLC 与计算机无协议通讯，主要从无协议通讯的硬件、配线、数据寄存器设置、PLC 与计算机无协议通讯的指令用法、PLC 程序编写和计算机 VB 程序的编写来说明无协议通讯的过程和一般方法。

[关键词]

PLC 通讯 无协议通讯 寄存器 VB 232 485 求和校验

[Abstract]

My dissertation introduces the significance of PLC communications and the four means of communication of Mitsubishi FX's PLC, And highlights the no protocol communications of FX series PLC and computer, no protocol communications hardware, wiring, Register data set, and the usage of command about no protocol communications, How to write PLC program and computer VB program to illustrate the process of no protocol communications and general method.

[key words]

PLC communications no protocol communications Register VB
232 485 Sum check

前　　言

PLC 问世时间不长，但是随着微处理器的出现，大规模，超大规模集成电路技术的迅速发展和数据通讯技术的不断进步，PLC 得到了迅速的发展；PLC 以高可靠性、丰富的 I/O 接口模块、采用模块化结构、编程简单易学、安装简单、维修方便等优点在自动控制领域从出现到大面积推广到逐步取代继电器控制。

随 PLC 的发展到上世纪 90 年代出现了 PLC 通讯，PLC 不再是孤立的 PLC，PLC 可以相互协作共同完成较大的任务，同时也出现了 PLC 和计算机的通讯，PLC 可以和计算机之间相互交换数据，可以通过计算机来控制 PLC，通过 PLC 把工厂实时数据传输到计算机达到计算机对控制过程的监控实现组态控制，让决策者更快更好的掌握工厂数据和市场数据做出正确的决策，使公司在激烈的市场竞争中立于不败之地。那么 PLC 到底怎么和计算机通讯呢，要实现 PLC 和计算机之间通讯要些什么硬件和软件呢，要如何来组织硬件、怎么编写软件呢？本文作者拟就这些问题以三菱公司 FX 系列的 PLC 和计算机无协议通讯为例来阐述 PLC 和计算机的通讯。

目录

一、 PLC通讯问题提出	4
1、 PLC通讯的应用使PLC能处理更大的控制系统	4
2、 PLC通讯的应用可以节省PLC的输入输出点	4
3、 PLC通讯的应用使PLC作为计算机的一个控制对象（执行机构），让PLC可以处理更复杂的系统	5
4、 PLC通讯的应用使工业组态控制成为可能	5
5、 PLC通讯的应用使PLC的远程控制成为可能	5
二、 FX系列PLC通讯概述	6
1、 N: N网络	6
2、 并行连接	6
3、 RS协议连接	7
4、 计算机无协议连接	7
三、 FX系列PLC和计算机无协议通讯硬件组成	7
1、 计算机通过 232 口和PLC通过 232 模块连接	8
2、 计算机通过 232 口和PLC通过 485 模块的连接	8
四、 FX系列PLC和计算机无协议通讯PLC软元件分配和控制命令	9
1、 PLC软元件分配	9
2、 PLC控制命令	10
五、 FX系列PLC和计算机无协议通讯寄存器设置和PLC程序的编写	12
1、 无协议通讯主要数据寄存器设置	12
2、 PLC程序编写	14
六、 FX系列PLC和计算机无协议通讯计算机程序编写	15
1、 计算机与PLC通讯控制信号代码	15
2、 求和校验码	15
3、 计算机程序编写	16
七、 FX系列PLC和计算机无协议通讯总体构成	23
结束语	24
参考文献	24

一、PLC 通讯问题提出

可编程控制器 PLC 是一种数字运算操作的电子系统，专为在工业环境应用而设计的。它采用一类可编程的存储器，用于其内部存储程序，执行逻辑运算，顺序控制，定时，计数与算术操作等面向用户的指令，并通过数字或模拟式输入/输出控制各种类型的机械或生产过程。PLC 自问世以来经过数十年的发展到现在已经具备高可靠性、丰富的 I/O 接口模块、采用模块化结构、编程简单易学、安装简单、维修方便等优点。近年来为适应工业控制的要求，特别是生产流水线的要求，PLC 出现了通讯功能，通讯功能的出现大大提高了 PLC 的应用范围和功能，主要有以下几个方面：

1、PLC 通讯的应用使 PLC 能处理更大的控制系统

在没有出现 PLC 通讯之前 PLC 只是作为一个单独的控制元器件控制一个简单的系统，如果是系统复杂那么就要更换高档次的 PLC 去控制。有了 PLC 的通讯就变得方便多了，可以用档次不高具备通讯功能的 PLC 组合成一个网络共同去完成系统的控制，特别是对那些分散的系统各个点的控制不复杂，但要求各点间有数据交换，并根据相应的数据协同控制的大系统用 PLC 的通讯功能就容易处理了。

2、PLC 通讯的应用可以节省 PLC 的输入输出点

PLC 的性能指标中输入输出点的数目是 PLC 的一个重要性能指标，扩展输入输出点的价格也是比较贵的，PLC 通讯的应用可以节省 PLC 输入输出点，在分散的系统中并不是每个点上 PLC 的输入输出点都会被用完，有的很少有的很多，如果要扩展可以用相对比较空闲的点上的 PLC 的点来扩展相对应用较多点上的 PLC 的输入输出点，并通过通讯使他们协调工作，犹如在同一台 PLC 上扩展一样；在和计算机通讯的 PLC 系统中还可以利用计算机的鼠标键盘输入来的数据作为 PLC 的输入点，还可以把其他设备输入给计算机的数据也通过 PLC 和计算机通讯传给 PLC，还可以把 Internet 上传给计算机的数据也通过 PLC 和计算机通讯传给 PLC，这样一来 PLC 的输入点和输出点就可以

节省，甚至可以把 PLC 的输入点都省去。

3、PLC 通讯的应用使 PLC 作为计算机的一个控制对象（执行机构），让 PLC 可以处理更复杂的系统

计算机和 PLC 比较可以知道计算机的处理能力要比 PLC 强很多倍，但价格来看计算机又比 PLC 便宜的多，就目前市场上淘汰的计算机的处理能力都要比现在的中型 PLC 能力强。利用 PLC 和计算机通讯可以比较复杂的计算、数据库等 PLC 处理不了的问题留给计算机处理，计算机处理好之后把结果通过通讯传给 PLC 让 PLC 去执行结果。这样 PLC 就变成了计算机的执行机构，还可以大大方便非自动控制专业的能够编写计算机程序的人根据自己的要求来改变控制过程。

4、PLC 通讯的应用使工业组态控制成为可能

很多时候我们都希望从计算机的显示器上看到整个系统的运行过程，当发生故障的时候也可以明确的被标记出来实现组态控制，我们通过 PLC 和计算机的通讯可以把计算机的数据和 PLC 的数据相互交换，在计算机上通过可视化的编程软件编写程序，以图形界面的形式直观的反映 PLC 的软元件实时情况从而来反映整个控制系统的情况，并对系统的数据保存、分析、报警。

5、PLC 通讯的应用使 PLC 的远程控制成为可能

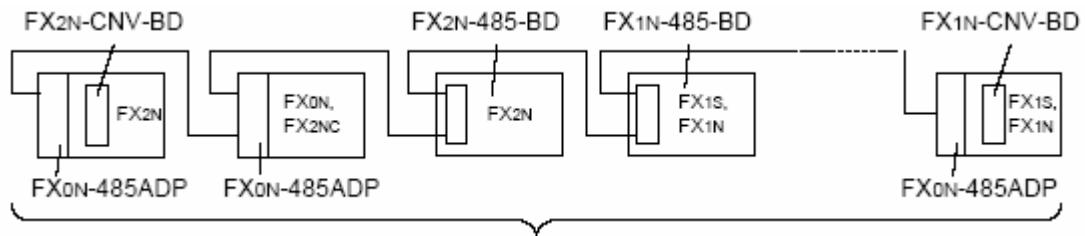
目前能和 Internet 连接的 PLC 价格都比较高，用的也不是太多，但是计算机和 Internet 的连接却是很简单价格也便宜，我们可以把计算机接入 Internet，计算机和 PLC 通过通讯相接，这样我们就可以在能接入 Internet 的任何一个地方对 PLC 进行控制，控制整个生产过程，掌握实时的数据作出正确的决策。

从上面几个方面我们可以看出，PLC 的通讯必将是 PLC 的一个新的发展方向，也必将得到大范围的推广。

二、FX 系列 PLC 通讯概述

三菱 FX 系列 PLC 通讯有 N: N 网络 (N: N Network), 并联连接 (Parallel Link), RS 协议连接 (RS Link) 和计算机无协议连接 (NO Protocol Communication)。

1、N: N 网络



N:N 网络主要是通过 485 模块把 8 台内的 FX 系列的 PLC 连接成一个网络, 有 one pair 和 two pair 两种配线, 通过共享存储单元的形式来相互交换数据; 主要设置的数据寄存器有 M8038, D8176 站号; D8177 从站个数设置 (只有主站设置); D8178 公共数据寄存器范围设置 (只有主站设置), 根据 PLC 的不同共享的存储单元也不相同; D8179 通讯等待时间设置 (只有主站设置) 主站与从站通讯等待时间通常为 0……10, 缺省为 10, 如果超过这个时间则通讯错误; D8180 通讯超时时间设置 (只有主站设置) 主站与从站通讯时间通常为 5……255, 缺省为 5, 如果超过这个时间则通讯错误。

2、并行连接



并行连接是两台同型号的 FX 系列的 PLC 通过 485 模块采用 one pair 或者 two pair 的配线连接起来, 也是通过共享存储单元的形式实现数据的交换, 但共享的范围要比 N:N 网络要多要快, 有普通模式和高速模式, 主要设置的数据寄存器有:

M8070 并行通讯主站驱动

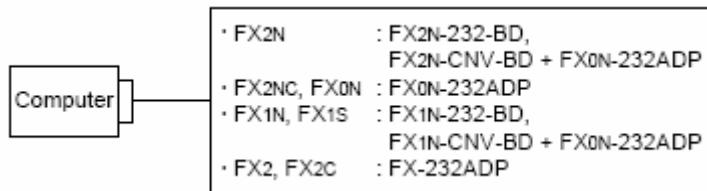
M8071 并行通讯从站驱动

M8072 当 PLC 作并行通讯时为 ON “并行通讯标志”

M8073 并行通讯中 M8070 和 M8071 设置错误标志, 错误是为 ON

M8162 并行通讯高速模式，两字节读写
 M8070 并行通讯看门狗时间，默然为 500MS

3、RS 协议连接



RS 协议通讯是通过计算机、打印机等的 232 口和 PLC 通过 232BD 模块连接，
 并通过 RS 指令达到通讯的目的，主要设置的数据寄存器有：

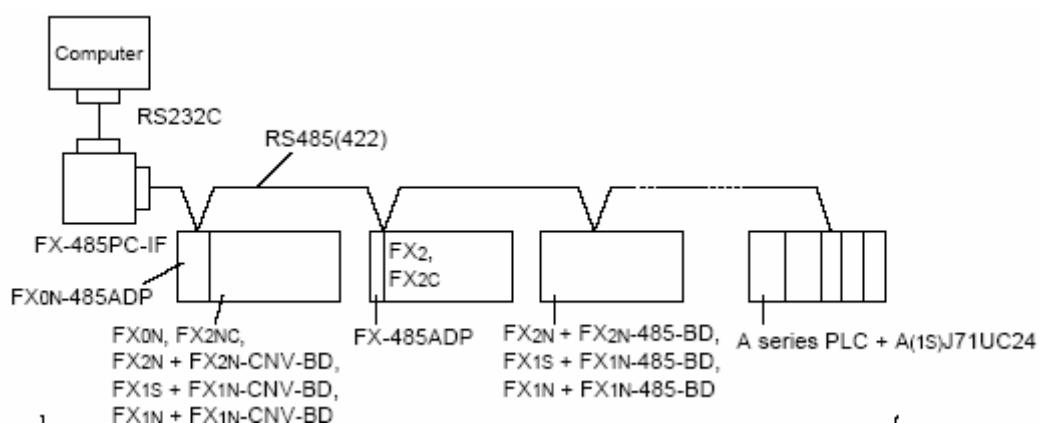
D8120 设置数据长度，校验，停止位，波特率设置等；

M8122 RS-数据传送标志，OFF：不传送，ON：传送，RS-数据被传送时有效。

M8123 结束接收数据，OFF：未结束，ON：结束接收数据，RS-数据接收完时有效。

4、计算机无协议连接

计算机无协议链接是计算机通过 232 口和 PLC 的 232 模块连接通讯，这样一台计算机可以控制一台 PLC，如果计算机通过 232 口，中间经过 232—485 的模块 485PC—IF 再和 PLC 通过 485 模块连接就可以一台计算机控制 16 台 PLC。具体过程后面讲述。

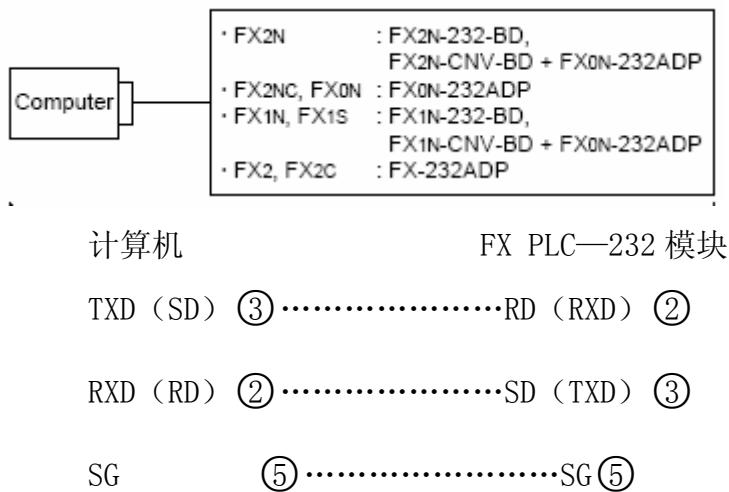


三、FX 系列 PLC 和计算机无协议通讯硬件组成

FX 系列 PLC 和计算机无协议通讯有两种方式，硬件组成如下：

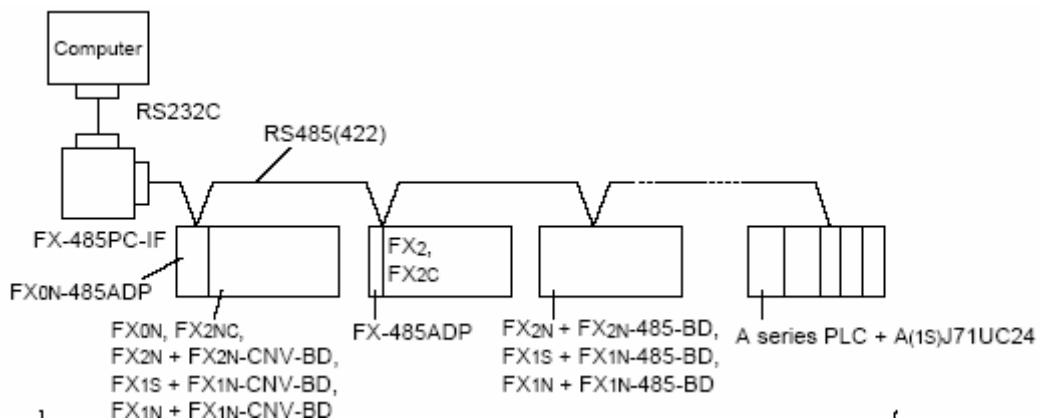
1、计算机通过 232 口和 PLC 通过 232 模块连接

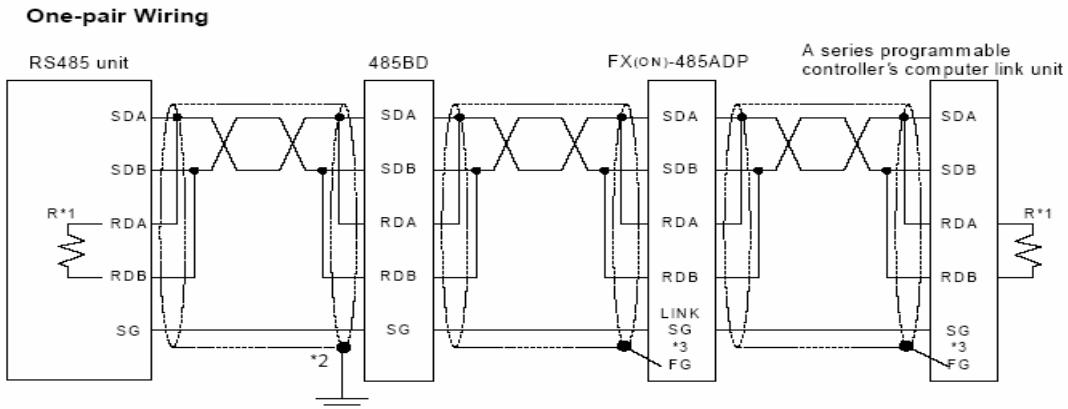
使用的硬件有计算机(带 232 口)一台, FX 系列 PLC 一台(FX2N、FX0N、FX2C 等), PLC 相对应的 232 模块一个, 比如 FX2N 的 PLC 采用 232BD 模块, FX0N 的 PLC 采用 FX0N-232ADP 等, 注意 232 传输距离不超过 15 米。



2、计算机通过 232 口和 PLC 通过 485 模块的连接

使用的硬件有计算机(带 232 口)一台, FX 系列 PLC(FX2N、FX0N、FX2C 等)16 台以内型号主要有 FX2N, FX0N, FX1N 等; 232 电平转 485 电平的模块 485PC—IF 一个, 每台 PLC 配置一个 485 模块, 根据具体的 PLC 来选择 485 模块, 比如 FX2N 的 PLC 选择 FX2N—485BD, FX0N 的 PLC 选择 FX0N—485ADP 等。通讯距离计算机 232 口到 485PC—IF 应小于 15 米, 各 PLC 通过 485 模块连接不超过 500 米。PLC 和 PLC 之间通过 485 模块连接使用 one pair 配线和 N: N 网络相同。





四、FX 系列 PLC 和计算机无协议通讯 PLC 软元件分配和控制命令

1、PLC 软元件分配

1) 位元件

元件	PLC 型号					使用命令	
	FXON	FX1S	FX, FX2C	FXIN	FX2N, FX2NC	BR, BW, BT	WR, WW, WT
输入继电器	X0000---	X0000--	X0000---	X0000--	X0000---		
	X0177	-	X0337	-	X0267		
		X0177		X0177			
输出继电器	Y0000--	Y0000--	Y0000--	Y0000--	Y0000--		
	Y0177	Y0015	Y0337	Y0177	Y0267		
辅助继电器	M0000—M0511		M0000—M1535		M0000--		
					M3071		
状态寄存器	S0000--S0127		S0000—S0999				
特殊功能继电器	M8000—M8254		M8000—M8255				
时间继电器	TS000—TS063		TS000—TS255				
计数器	CS000—CS031		CS000—CS255				
	CS235—CS254						

2) 字元件分配

元件	PLC 型号					使用命令		
	FXON	FX1S	FX, FX2C	FXIN	FX2N, FX2NC	BR, BW, BT	WR, WW	WT
时间继电器	TN000—TN063		TN000—TN255					
计数器	CN000—CN031		CN000—CN255					
	CN235—CN254							
数据寄存器	D0000—D0255		D0000—D0999		D0000—D7999			
文件寄存器	D1000—D2499		D1000—D2999					
特殊数据寄存器	D8000—D8255		D8000—D8255					

2、PLC 控制命令

命令类型	命令		所对软元件及描述	控制数	
	标识	ASCII 码		FX0N, FX1S	FX1N, FX2N
位	BR	42H, 52H	读取一组 X、Y、M、S、T、C 一位信息	54 点	256 点
	BW	42H, 57H	写入一组 X、Y、M、S、T、C 一位信息	46 点	160 点
	BT	42H, 54H	选择性写入几组 X、Y、M、S、T、C 的一位信息	10 点	20 点
字	WR	57H, 52H	读取一组 X、Y、M、S 字信息	13 字	32 字
			208 点	512 点	
	WW	57H, 57H	读取一组 T、C、D 字信息	13 点	64 点
			10 字	10 字	
	WT	57H, 54H	写入一组 X、Y、M、S	160 点	160 点
			写入一组 D、T、C 字信息	11 点	64 点
PC	RR	52H, 52H	选择性写入几组 X、Y、M、S 字信息	6 字	10 字
	RS	52H, 53H	选择性写入几组 T、C、D 字信息	96 点	160 点
	PC	50H, 43H	强制 PLC 为 RUN	6 点	10 点
			强制 PLC 为 STOP	—	—
			读取 PLC 型号		

1) BR (计算机读取 PLC 位元件) 指令

PC 侧:	1	2	3	4	5	6	7	8							15	16	17
	ENQ	站号	PLC 型号	BR	延时	首址	地数量	求和校验							ACK	站号	PLC 号
PLC 侧:									STX	站号	PLC 型号	数据	EXT	求和校验			
									NAK	站号	PLC 型号	错误代码					
									9	10	11	12	13	14			

例如读取 PLC 上 X0 到 X7 的状态，计算机先发送：

ENQ + “00FFBR0X000008” +求和校验码 PLC 受到后作出响应，具体看后面的程序。

2) WR (计算机读取 PLC 字元件) 指令

PC 侧:	1	2	3	4	5	6	7	8							15	16	17
	ENQ	站号	PLC 型号	WR 延时	首址	地数量	数	求和校验							ACK	站号	PLC 号
PLC 侧:									STX	站号	PLC 型号	数据	EXT	求和校验			
									NAK	站号	PLC 型号	错误代码					
									9	10	11	12	13	14			

3) BW (计算机写入 PLC 位元件) 指令

PC 侧:	1	2	3	4	5	6	7	8	9								
	ENQ	站号	PLC 型号	BW	延时	首址	地数量	数据	求和校验								
PLC 侧:										STX	站号	PLC 型号					
										NAK	站号	PLC 型号	错误代码				
										10	11	12	13				

4) WW (计算机写入 PLC 字元件) 指令

PC 侧:	1	2	3	4	5	6	7	8	9								
	ENQ	站号	PLC 型号	WW	延时	首址	地数量	数	求和校验								
PLC 侧:										STX	站号	PLC 型号					
										NAK	站号	PLC 型号	错误代码				
										10	11	12	13				

5) BT (计算机测试位元件) 指令

PC 侧:	1	2	3	4	5	6	7	8	9	10	11					
	ENQ	站号	PLC 型号	BT	延时	数量	位元件	数元件	位元件	数元件	求和校验					
PLC 侧:												STX	站号	PLC 型号			
												NAK	站号	PLC 型号	错误代码		
												10	11	12	13		

6) WT (计算机测试字元件) 指令

PC 侧:	1	2	3	4	5	6	7	8	9	10	11				
	ENQ	站 号	PLC 型 号	WW	延 时	数 量	字 元 件	数 据	字 元 件	数 据	求 和				
PLC 侧:													STX	站号	PLC 型 号	
													NAK	站号	PLC 型 号	错 误 代 码

7) PC (读取 PLC 型号) 指令

PC 侧:	1	2	3	4	5	6								13	14	15
	ENQ	站 号	PLC 型 号	PC	延 时	求 和 校 验								ACK	站 号	PLC 号
PLC 侧:							STX	站 号	PLC 型 号	PLC 型号	EXT	求 和 校 验				
							NAK	站 号	PLC 型 号	PLC	错 误 代 码					
							7	8	9	10	11	12				

8) RR (强制运行) RS (强制停止) 指令

PC 侧:	1	2	3	4	5	6										
	ENQ	站号	PLC 型 号	RR 或 RS	延时	求和校验										
PLC 侧:							STX	站号	PLC 型 号							
							NAK	站号	PLC 型 号	PLC	错 误 代 码					
							7	8	9	10						

五、FX 系列 PLC 和计算机无协议通讯寄存器设置和 PLC 程序的编写

1、无协议通讯主要数据寄存器设置

1) D8120 设置

位编号	用途	0	1
	数据长度	7 位	8 位
B2B1	校验	00	无校验
		01	奇校验
		11	偶校验
B3	停止位	1 位	2 位
B7B6B5B4	波特率设置	0011	300
		0100	600
		0101	1200
		0110	2400
		0111	4800
		1000	9600
		1001	19200
B8	起始字符	无	D8124
B9	结束字符	无	D8125
B12B11B10	握手控制线	无协议	000 没有使用 RS232 001 终端模式 RS232 010 连接模式 RS232 011 普通模式 1 RS232 101 普通模式 2 RS232
		计算机连接	000 RS485 010 RS232
B13	求和校验	非求各校验模式	求和校验模式
B14	协议方式	无协议	RS 协议
B15	通讯控制协议	协议模式 1	协议模式 4

模式说明：模式 1 和模式 4 不同之处 D8120 的 B15 位设置上，B15 为 0 时为模式 1，当 B15 为 1 时为模式 4；模式 1 通讯格式后面不加 CR（返回）和 LF（忙）。

2) 其它数据寄存器及中间继电器设置

名称	用法描述
M8121	RS-数据传送延迟 ON: 传送延迟
M8122	RS-数据传送标志 OFF: 不传送 ON: 传送 RS-数据被传送时有效.
M8123	结束接收数据 OFF: 未结束 ON: 结束接收数据 RS-数据接收完时有效
M8124	RS-信号检测到标志 OFF: 信号未检测到 ON: 信号检测到 RS-信号被检测到时有效
M8126	连接标志（计算机连接）
M8127	握手正确（计算机连接）
M8128	握手错误*计算机连接)
M8129	超时标志
M8161	8 位与 18 区别标志
D8120	通信格式

D8121	本地站号 用于无议协 485 数据网络的本地站号
D8122	RS-等待发送的数据量
D8123	RS-已经接收到的数据量
D8124	RS-数据报头 缺省值是 STX(02H)
D8125	232ADP-数据结束符, 缺省值是 ETX(03H)
D8127	RS485-查询主设备寄存器
D8128	RS485-查询数据长度寄存器
D8129	RS485 数据网络通讯超时

2、PLC 程序编写

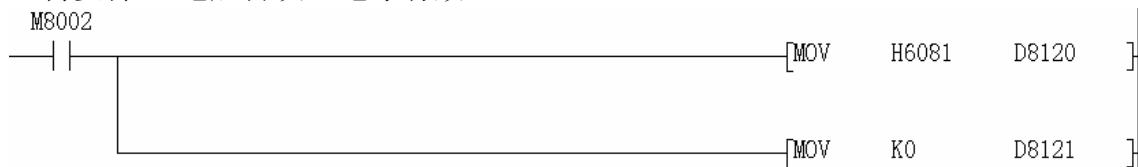
无协议连接 PLC 程序比较简单, PLC 中只要写入通讯方式 D8120, 及 PLC 站号 D8121 就可以, PLC 就能根据从计算机发送来的指令和数据做出相应的响应(发送数据给计算机或者执行计算机的指令), 当然, 也可以通过写些程序来配合计算机工作。注意: 每次改过 D8120 的参数后要掉电一次, FXON 的 PLC 使用无协议通讯是要接通 M8120 。

1) 采用 RS485ADP 及 485PC—IF 连接的 PLC 程序编写

采用 485 后一台计算机能连接 16 台 PLC, 因此要给 PLC 编站号, 定义在 D8121 中;

```
LD      M8002      MOV K0      D8121      设置为 0 号站
LD      M8002      MOV K1      D8121      设置为 1 号站
```

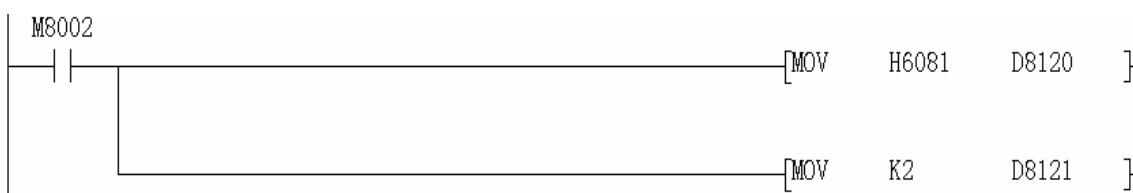
波特率等参数还是写在 D8120 中 (内容见表 D8120), 写完程序输入到 PLC 后记得要掉 电后再次上电才有效。



上面是 0 号站 PLC 程序, H6081 是 16 进制表示, 2 进制为

0 1 1 000 0 0 1000 0 00 1

表达的意思是: 数据长度为 8 位, 无奇偶校验, 停止位 1 位, 波特率 9600, 无报头, 无报尾, RS485 连接, 求和校验方式, 无协议, 协议方式 1, 2 号站程序如下:

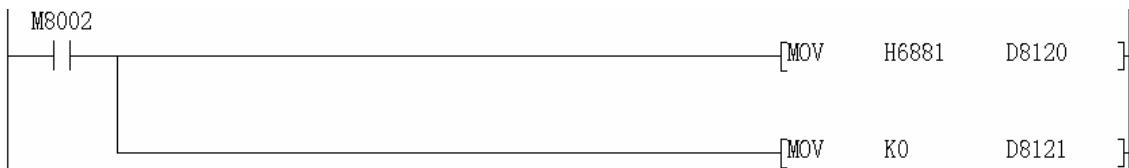


2) 采用 RS232 连接的 PLC 程序编写

采用 RS232 后一台计算机只能连接 1 台 PLC, 因此 PLC 编站号只能为 0, 定义在 D8121 中;

```
LD      M8002      MOV K0      D8121      设置为 0 号站
```

波特率等参数还是写在 D8120 中 (内容见表 D8120), 写完程序输入到 PLC 后记得要掉 电后再次上电才有效。



上面是 0 号站 PLC 程序, H6881 是 16 进制表示, 2 进制为

0 1 1 010 0 0 1000 0 00 1

表达的意思是: 数据长度为 8 位, 无奇偶校验, 停止位 1 位, 波特率 9600, 无报头, 无报尾, RS232 连接, 求和校验方式, 无协议, 协议方式 1。

六、FX 系列 PLC 和计算机无协议通讯计算机程序编写

1、计算机与 PLC 通讯控制信号代码

标识	代码 (16 进制)	VB 格式	描述
STX	02H	Chr(2)	开始发送字符
ETX	03H	Chr(3)	结束发送字符
EOT	04H	Chr(4)	结束传输
ENQ	05H	Chr(5)	询问字符
ACK	06H	Chr(6)	应答字符正确
LF	0AH	Chr(A)	线路忙
CL	0CH	Chr(C)	清除
CR	0DH	Chr(D)	返回
NAK	15H	Chr(15)	应答字符错误

2、求和校验码

求和校验码是保证串行通讯正确的一种方法, 这里所说的求和是指要通讯的数据除报头外 ASCII 码以 16 进制数形式相加, 并取两位 16 进制。

例: 读取 00 站是 PLC 的位存储单元 X0 开始的 5 位元件状态

描述	请求	站号	PLC 型号	命令	等待时间	起始元件	元件数	求和校验码
格式	ENQ	0 0	F F	B R	A	X0000	05	A3
16 进制码	05H	30H, 30H	46H, 46H	42H, 52H	41H	58H, 30H, 30H, 30H, 30H	30H, 35H	41H, 33H

求和校验码:

= 把串行通讯数据的 ASCII 码以 16 进制相加取两位 16 进制数

= 16 进制的 [0+0+F+F+B+R+A+X+0+0+0+0+5]

= 30H+30H+46H+46H+42H+52H+41H+58H+30H+30H+30H++30H+35H

取两位 16 进制得 A3H

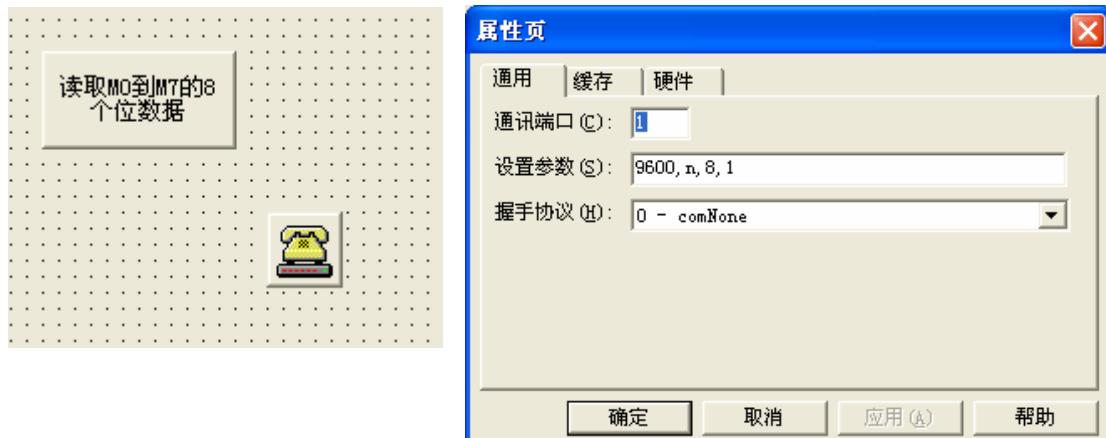
在计算机程序中我们通过下面这个子程序来调用计算：

```
Private Function SumChk(Dats$) As String
    Dim i&
    Dim CHK&
    For i = 1 To Len(Dats)
        CHK = CHK + Asc(Mid(Dats, i, 1))
    Next i
    SumChk = Right(Hex$(CHK), 2)
End Function
```

3、计算机程序编写

计算机程序采用 VB 来编写，通常先画控件，定义控件的各种属性和方法，定义计算机串口，编写代码，调试等几个过程。下面以几个例子来说明计算机程序的编写。

1) 通过 BR 指令来读取 00 站 PLC 上 M0 到 M7 的状态：PLC 程序同前面的。



```
Private Function
SumChk(Dats$) As String; 16 进制求和校验码程序
    Dim i&
    Dim CHK&
    For i = 1 To Len(Dats)
        CHK = CHK + Asc(Mid(Dats, i, 1))
    Next i
    SumChk = Right(Hex$(CHK), 2)
End Function
```

```
Private Sub Command1_Click() ' 读写码 ENQ 00 FF BR 0 M0000 08 计
算机通过串口发送到 PLC 读取 M0 到 M7 的状态
Dim sd As String
If MSComm1.PortOpen = False Then
    MSComm1.PortOpen = True ' 打开串行端口'
End If
MSComm1.InputLen = 0 ' 串行数据接收缓冲区初始化'
```

```

sd = "00FFBROM000008" '注意区分大小写'
MSComm1.Output = Chr(5) + sd + SumChk(sd) '向 PLC 发送的读取数据命令字符串'
WT: If MSComm1.InBufferCount < 16 Then GoTo WT; 等待收数据
rd = MSComm1.Input '读取接收缓冲区数据'
pd = Mid(rd, 1, 5) '取 PLC 应答字符串的前 5 位'
If pd = Chr(2) + "00FF" Then '判断读取是否正确'
js = Chr(6) + "00FF" '向 PLC 发送已正确接收数据信号'
MSComm1.Output = js
rd = Mid(rd, 6, 8) '检出数据以二进制表示, M0M1……M1
Print "收到数据: "
Print rd
X = "读取正常!"
Else
X = "读取错误!"
End If
Print X
MSComm1.PortOpen = False
End Sub

```

- 2) 通过 BW 指令是 Y0 到 Y7 为 ON 和 OFF, PLC 程序不变, 计算机程序如下:
用 VB 先画两个 command 控件, 点击发送数据给 PLC 改变 PLC 上 Y0—Y7 状态, 再画一个串口控件, 属性和 1) 相同。VB 代码如下:

```

Private Function SumChk(Dats$) As String
Dim i&
Dim CHK&
For i = 1 To Len(Dats)
    CHK = CHK + Asc(Mid(Dats, i, 1))
Next i
SumChk = Right(Hex$(CHK), 2)
End Function

```

```

Private Sub Command1_Click()
Dim sd As String
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.InputLen = 0
sd = "00FFBW0Y00000811111111"
MSComm1.Output = Chr(5) + sd + SumChk(sd)
WT: If MSComm1.InBufferCount < 5 Then GoTo WT
pd = Mid(rd, 1, 5)
If pd = Chr(2) + "00FF" Then
X = "位写入正确!"
Else

```

```

X = "位写入错误！"
End If
Print X
MSComm1.PortOpen = False
End Sub

Private Sub Command2_Click()
Dim sd As String
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.InputLen = 0
sd = "00FFBW0Y00000800000000"
MSComm1.Output = Chr(5) + sd + SumChk(sd)
WT: If MSComm1.InBufferCount < 5 Then GoTo WT
pd = Mid(rd, 1, 5)
If pd = Chr(2) + "00FF" Then
X = "位写入正确"
Else
X = "位写入错误！"
End If
Print X
MSComm1.PortOpen = False
End Sub

```

3) 通过BT指令改写Y0为ON, M100为OFF, M200为1

PLC程序不变,计算机VB程序画一个command控件,串口同前面,代码如下:

```

Private Function SumChk(Dats$) As String
Dim i&
Dim CHK&
For i = 1 To Len(Dats)
CHK = CHK + Asc(Mid(Dats, i, 1))
Next i
SumChk = Right(Hex$(CHK), 2)
End Function

```

```

Private Sub Command1_Click()
Dim sd As String
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.InputLen = 0
sd = "00FFBT003Y00001M010000M020001"
MSComm1.Output = Chr(5) + sd + SumChk(sd)
WT: If MSComm1.InBufferCount < 5 Then GoTo WT

```

```

pd = Mid(rd, 1, 5)
If pd = Chr(2) + "00FF" Then
    X = "测试位写入正确"
Else
    X = "测试位写入错误！"
End If
Print X
MSComm1.PortOpen = False
End Sub

```

4) 通过 WR 多 D0 到 D2 的数据

PLC 程序不变，VB 程序画一个 command 空间和一个串口控件，注意收到的字元件数据是十六进制的，要变成其他进制还要转换，代码如下：

```

Private Function SumChk(Dats$) As String
    Dim i&
    Dim CHK&
    For i = 1 To Len(Dats)
        CHK = CHK + Asc(Mid(Dats, i, 1))
    Next i
    SumChk = Right(Hex$(CHK), 2)
End Function

Private Sub Command1_Click()
    Dim sd As String
    If MSComm1.PortOpen = False Then
        MSComm1.PortOpen = True
    End If
    MSComm1.InputLen = 0
    sd = "00FFWR0D000003"
    MSComm1.Output = Chr(5) + sd + SumChk(sd)
    WT: If MSComm1.InBufferCount < 20 Then GoTo WT
    rd = MSComm1.Input
    pd = Mid(rd, 1, 5)
    If pd = Chr(2) + "00FF" Then
        js = Chr(6) + "00FF"
        MSComm1.Output = js
        rd = Mid(rd, 6, 12)
        Print "收到 D0 到 D2 数据: "
        Print rd
        X = "读取正确"
    Else
        X = "读取错误！"
    End If
    Print X
    MSComm1.PortOpen = False
End Sub

```

```
End Sub
```

- 5) 通过WW令使3号站PLC上D100到D103的数据改成H1234, HABCD, H0011, HFF00, 代码如下:

```
Private Function SumChk(Dats$) As String
    Dim i&
    Dim CHK&
    For i = 1 To Len(Dats)
        CHK = CHK + Asc(Mid(Dats, i, 1))
    Next i
    SumChk = Right(Hex$(CHK), 2)
End Function
```

```
Private Sub Command1_Click()
    Dim sd As String
    If MSComm1.PortOpen = False Then
        MSComm1.PortOpen = True
    End If
    MSComm1.InputLen = 0
    sd = "03FFWW0D0100041234ABCD0011FF00"
    MSComm1.Output = Chr(5) + sd + SumChk(sd)
    WT: If MSComm1.InBufferCount < 5 Then GoTo WT
    pd = Mid(rd, 1, 5)
    If pd = Chr(2) + "03FF" Then
        X = "字写入正确!"
    Else
        X = "字写入错误!"
    End If
    Print X
    MSComm1.PortOpen = False
End Sub
```

- 6) 通过WT改写5号站上D100为H3456, D200为HAAAA, 代码如下:

```
Private Function SumChk(Dats$) As String
    Dim i&
    Dim CHK&
    For i = 1 To Len(Dats)
        CHK = CHK + Asc(Mid(Dats, i, 1))
    Next i
    SumChk = Right(Hex$(CHK), 2)
End Function
```

```
Private Sub Command1_Click()
    Dim sd As String
    If MSComm1.PortOpen = False Then
        MSComm1.PortOpen = True
```

```

End If
MSComm1.InputLen = 0
sd = "05FFWT002D01003456D0200AAAA"
MSComm1.Output = Chr(5) + sd + SumChk(sd)
WT: If MSComm1.InBufferCount < 5 Then GoTo WT
pd = Mid(rd, 1, 5)
If pd = Chr(2) + "05FF" Then
X = "测试字写入正确!"
Else
X = "测试字写入错误!"
End If
Print X
MSComm1.PortOpen = False
End Sub

```

7) 通过 PC 指令读取 3 号站 PLC 型号, 代码如下:

```

Private Function SumChk(Dats$) As String
Dim i&
Dim CHK&
For i = 1 To Len(Dats)
CHK = CHK + Asc(Mid(Dats, i, 1))
Next i
SumChk = Right(Hex$(CHK), 2)
End Function

Private Sub Command1_Click()
Dim sd As String
If MSComm1.PortOpen = False Then
MSComm1.PortOpen = True
End If
MSComm1.InputLen = 0
sd = "03FFPC0"
MSComm1.Output = Chr(5) + sd + SumChk(sd)
WT: If MSComm1.InBufferCount < 10 Then GoTo WT
rd = MSComm1.Input
pd = Mid(rd, 1, 5)
If pd = Chr(2) + "00FF" Then
js = Chr(6) + "00FF"
MSComm1.Output = js
rd = Mid(rd, 6, 2)
Print "PLC 型号是: "
Print rd
X = "读取正确"
Else
X = "读取错误!"
End If
End Function

```

```

End If
Print X
MSComm1.PortOpen = False
End Sub

```

说明：PLC 型号根据 PLC 的不同而不同，比如：

FX1S.....F2H	FXON.....8EH
FX2C.....8DH	FX1N.....9EH
FX2N, FX2NC.....9DH	

8) 通过 RR 和 RS 指令使 0 号站 PLC 强制运行和停止，代码如下：

```

Private Function SumChk(Dats$) As String
    Dim i&
    Dim CHK&
    For i = 1 To Len(Dats)
        CHK = CHK + Asc(Mid(Dats, i, 1))
    Next i
    SumChk = Right(Hex$(CHK), 2)
End Function

```

```

Private Sub Command1_Click()
Dim sd As String
If MSComm1.PortOpen = False Then
    MSComm1.PortOpen = True
End If
MSComm1.InputLen = 0
sd = "00FFRR0"
MSComm1.Output = Chr(5) + sd + SumChk(sd)
WT: If MSComm1.InBufferCount < 5 Then GoTo WT
pd = Mid(rd, 1, 5)
If pd = Chr(2) + "00FF" Then
    X = "强制运行正确！"
Else
    X = "强制运行错误！"
End If
Print X
MSComm1.PortOpen = False
End Sub

```

```

Private Sub Command2_Click()
Dim sd As String
If MSComm1.PortOpen = False Then
    MSComm1.PortOpen = True
End If
MSComm1.InputLen = 0
sd = "00FFRS0"

```

```

MSComm1.Output = Chr(5) + sd + SumChk(sd)
WT: If MSComm1.InBufferCount < 5 Then GoTo WT
pd = Mid(rd, 1, 5)
If pd = Chr(2) + "00FF" Then
X = "强制停止正确!"
Else
X = "强制停止错误!"
End If
Print X
MSComm1.PortOpen = False
End Sub

```

七、FX 系列 PLC 和计算机无协议通讯总体构成

总体说来 FX 系列 PLC 和计算机无协议通讯主要包括这 7 个方面内容：

- 1、硬件组成：PLC、通讯模块、计算机。
- 2、硬件连接有两种，通过 232 还是 485PC—IF 连接，485 采用 one pair 配线。
- 3、PLC 程序编写，注意 PLC 的型号和站号，FX0N 的 PLC 注意 M8120 接通，修改 D8120 后要掉电一次，D8120 中的装置要和计算机串口设置相同。
- 4、计算机程序编写，计算机的程序根据控制的要求进行编写，前面给出的都是具体控制某些软元件的方法，具体控制过程还得具体分析，程序流程基本相似，大体如下：
 - 1) 打开串口
 - 2) 准备控制命令和内容数据
 - 3) 发送命令或者命令加内容数据给 PLC
 - 4) 等待 PLC 返回数据
 - 5) 收到 PLC 返回数据
 - 6) 判断数据是否正确
 - 7) 根据判断发送数据给 PLC
 - 8) 读取数据
 - 9) 关闭串口
- 5、程序联机调试
- 6、生成可执行文件或者安装文件
- 7、编写说明书

结束语

本文简单的介绍了三菱 FX 系列 PLC 的计算机无协议通讯，主要从硬件配置、软件编写等几个方面进行了讲述，也对比了 FX 系列 PLC 的几种通讯方式；希望对从事自动控制的同行、在校学生、PLC 爱好者等在 PLC 通讯方面有所帮助和启发，本文只是抛砖引玉罢了。由于时间仓促和本人水平有限文中错误难免，希望读者批判指正。

参考文献

- | | | | |
|---|-------|-------------|-----------|
| 1 | 张万忠 | 可编程控制器应用技术 | 化学工业出版社 |
| 2 | 夏幸明 | 可编程控制器技术及应用 | 北京理工大学出版社 |
| 3 | 三菱公司 | FX 系列通讯手册 | |
| 4 | 三菱公司 | FX 系列编程手册 | |
| 5 | 三菱公司 | GPPW 使用手册 | |
| 6 | 思南工控网 | | |