

# 三菱 PLC 与上位机串行通信的实现

周 鸣<sup>1</sup>, 曲 凌<sup>2</sup>

(1. 上海师范大学 机械与电子工程学院 上海 201418; 2. 上海交通大学 工程训练中心 上海 200240)

**摘 要:** 为了实现对现场生产过程进行监控, PLC 串行通信技术在工业生产中正得到越来越广泛的应用, 介绍了三菱 FX2N 系列 PLC 与上位机进行通信的主要原理, 阐述在 Visual Basic 6.0 环境下上位机与三菱 FX2N PLC 串行通信的具体实现方法。具体介绍了与上位机通信的硬件连接, 三菱 FX2N PLC 的通信协议及 VB 中的 MSCOMM 控件, 对通信实现过程中的主要程序进行简单分析。

**关键词:** PLC; 通信协议; 串行通信; VB; MSCOMM 控件

**中图分类号:** TP271

**文献标识码:** B

**文章编号:** 1004-373X(2006)22-006-03

## Application of Serial Communication between Mitsubishi PLC and Host Computer

ZHOU Ming<sup>1</sup>, QU Ling<sup>2</sup>

(1. Mechanical and Electronical Information Engineering College, Shanghai Normal University, Shanghai, 201418, China;

2. Engineering Training Center, Shanghai Jiaotong University, Shanghai, 200240, China)

**Abstract:** In order to monitor the production process, serial communication method is widely used in industrial production. The article mainly introduces the principle of serial communication with Mitsubishi FX2N series PLC, describes the application of serial communication between Mitsubishi PLC and host computer on Visual Basic 6.0 platform. The hardware connection, communication protocol of FX2N series PLC and the MSCOMM control part in VB 6.0 are discussed. The main program is also analyzed in this article.

**Keywords:** PLC; communication protocol; serial communication; Visual Basic; MSCOMM

可编程控制器(PLC)是专为在工业环境下应用而设计的一种控制器, 具有抗干扰能力强、可靠性好、体积小等特点, 在一些工业生产设备的控制中得到了应用。为了监控现场的信号与工艺, 组成一个友好的人机界面, 在 PLC 的具体应用过程中往往采用上位机与下位 PLC 进行串行通信的方式。本文主要介绍了在 VB6.0 环境下上位计算机与下位三菱 FX2N PLC 进行串行通信的实现方法。

### 1 硬件连接

三菱 FX2N 系列 PLC 本机无串行通信的能力, 为了得到这个功能需外扩 FX232-BD 或 FX485-BD 通信模块, 其中 FX232-BD 为 RS 232 通信方式, FX485-BD 为 RS 485 通信方式。在此使用 FX232-BD 通信模块, 与上位计算机的连接方式如图 1 所示。

### 2 PLC 通信协议

三菱 PLC 有一些计算通信指令用于 PLC 与上位机之间进行通信, 主要有:

BR: 读取位状态指令。

WR: 读取字状态指令。

BW: 写位状态指令。

WW: 写字状态指令。

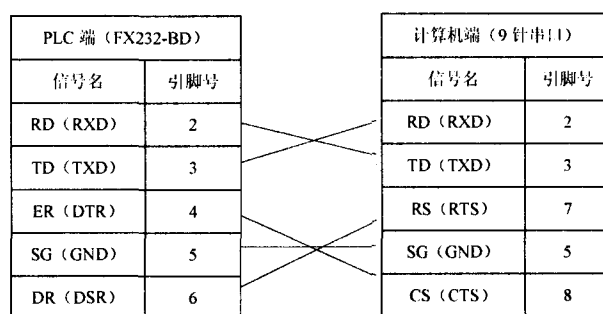


图 1 FX232-BD 与上位机的连接方式

#### 2.1 通信过程

上位机与 PLC 的通信过程如图 2 所示。

#### 2.2 PLC 通信寄存器 D8120 简介

PLC 的寄存器 D8120 为 16 位的通信控制寄存器, 在进行通信之前要对其进行初始化, 其各位参数如表 1 所示。

b0 位定义通信时数据长度, 一般为 7 位或 8 位。b1、b2 组合定义奇偶校验位有 3 种: 无校验、奇校验、偶校验。

b3 定义停止位, 1 位或 2 位。b4, b5, b6, b7 定义通信速率, 具体的定义见表中组合。上述几个参数在进行上位机编程时要和 PLC 端保持一致。b10, b11, b12 定义通信协议及所使用的接口是 RS 232 还是 RS 485, 一般情况下使用计算机通信协议, 所用通信接口根据实际情况设定。b13 用来定义是否在传输帧中增加校验码, 增加校验码可以确保通信传输的准确性。

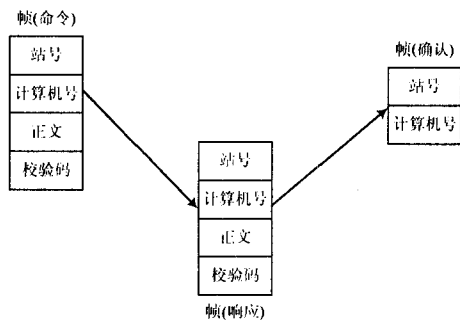


图 2 上位机与 PLC 的通信过程

表 1 寄存器 D8120 的参数

| Bit No.              | Name                          | Description  |   |
|----------------------|-------------------------------|--|---|
|                      |                               | 0 (bit = OFF)  | 1 (bit = ON)  |
| b0                   | Data length                   | 7 bit  | 8 bit   |
| b1<br>b2             | Parity                        | (b2, b1)<br>( 0, 0 ): None<br>( 0, 1 ): Odd<br>( 1, 1 ): Even  |   |
| b3                   | Stop bit                      | 1 bit  | 2 bit   |
| b4<br>b5<br>b6<br>b7 | Baud rate (bps)               | (b7, b6, b5, b4)<br>( 0, 0, 1, 1 ): 300<br>( 0, 1, 0, 0 ): 600<br>( 0, 1, 0, 1 ): 1,200<br>( 0, 1, 1, 0 ): 2,400   | (b7, b6, b5, b4)<br>( 0, 1, 1, 1 ): 4,800<br>( 1, 0, 0, 0 ): 9,600<br>( 1, 0, 0, 1 ): 19,200            |
| b8 <sup>1</sup>      | Header                        | None   | Effective (D8124) Default: STX (02H)  |
| b9 <sup>1</sup>      | Terminator                    | None   | Effective (D8125) Default: ETX (03H)  |
| b10<br>b11<br>b12    | Control line                  | No protocol<br>(b12, b11, b10)<br>( 0, 0, 0 ): No use <RS232C interface><br>( 0, 0, 1 ): Terminal mode <RS232C interface><br>( 0, 1, 0 ): Interlink mode <RS232C interface > (FX2N V2.00 or more)<br>( 0, 1, 1 ): Normal mode 1 <RS232C interface> <RS485 (422) interface> *3<br>( 1, 0, 1 ): Normal mode 2 <RS232C interface> (FX, FX3C only) | Computer link<br>(b12, b11, b10)<br>( 0, 0, 0 ): RS485 (422) interface<br>( 0, 1, 0 ): RS232C interface |
| b13 <sup>2</sup>     | Sum check                     | Sum check code is not added  | Sum check code is added automatically   |
| b14 <sup>2</sup>     | Protocol                      | No protocol  | Dedicated protocol  |
| b15 <sup>2</sup>     | Transmission control protocol | Protocol format 1  | Protocol format 4   |

### 2.3 上位机向 PLC 发送命令帧格式及 PLC 响应帧格式

上位机读 PLC 的命令帧格式和 PLC 的返回帧格式如图 3 所示。

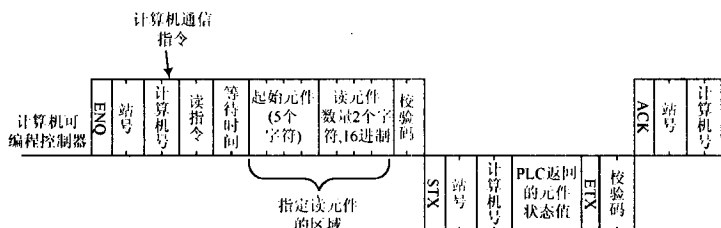


图 3 上位机读 PLC 的命令帧格式和 PLC 的返回帧格式

上位机写 PLC 的命令帧格式和 PLC 的返回帧格式如图 4 所示。

其中 PLC 站号由 PLC 中的 16 位寄存器 D8121 确定,

在 PLC 程序可以设定本机的站点号。计算机号为上位机告诉下位 PLC 本计算机的站点号, 在上位机程序中确定。元件起始号为所要读的第一个元件(包括输入输出和内部寄存器), 元件数量为所要读写的元件个数。站号、计算机号和读写元件数量以 16 进制表示。等待时间为上位告诉下位 PLC 的通信延时时间, 设定值为 0~150 ms, 在数据帧中以 16 进制 0~A 表示, 若设定为 0, 则 PLC 在接到上位机请求后马上回答。有无校验码可以在 PLC 的通信寄存器 D8120 中设定。

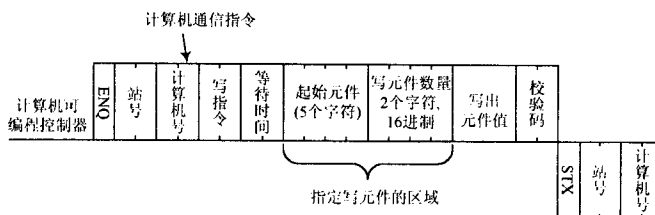


图 4 上位机与 PLC 的命令帧格式和 PLC 的反回帧格式

### 3 程序设计

程序的设计主要采用 VB 中的 MSCOMM 控件传送命令帧, PLC 返回的响应帧也是通过这个控件返回。

#### 3.1 MSCOMM 通信控件

利用 MSComm 控件实现计算机通信的关键是理解并正确设置 MSComm 控件众多属性和方法。以下是 MSComm 控件的常用属性和方法:

CommPort 设置或返回通讯端口号。

格式为: MSComm.CommPort[ = PortNumber ]。在设计时, PortNumber 可以设置成 1~16 的任何数(缺省值为 1)。但是如果用 PortOpen 属性打开一个并不存在的端口时, 就会产生错误。还需注意的是必须在打开端口之前设置 CommPort 属性。例如: MSComm.CommPort = 2, 即设置当前通信串口为 COM2 口。

Settings 以字符串形式设置或返回波特率、奇偶校验、数据位和停止位。

格式为:

MSComm.Settings[ = ParaString ]

ParaString 是一个包含 4 部分的字符串。第一部分为波特率。第二部分为奇偶校验, N 表示不校验, E 表示偶校验, O 表示奇校验, S 表示空格校验, M 表示符号校验。第三部分为数据位位数, 其可选值为 4, 5, 6, 7, 8。第四部分为停止位位数, 其可选值为 1, 1.5, 2。Settings 属性的缺省值为“9 600, N, 8, 1”。

PortOpen 设置并返回通讯端口的状态(打开或关闭), 在设计时无效。

格式为: MSComm.PortOpen[ = TRUE/FALSE ]。

Input 返回并删除接收缓冲区中的数据流。该属性

在设计时无效,在运行时为只读。

格式为:MSComm. Input。

InputLen 设置并返回 Input 属性从接收缓冲区中每次读取的字符数。

格式为:MSComm. InputLen[= CharNumber]。InputLen 属性的缺省值是 0。当设置 InputLen 为 0 时,使用 Input 将使 MSComm 控件读取接收缓冲区中全部的内容。若接收缓冲区中的字符数小于 InputLen 属性设置的字符数,Input 属性返回一个零长度字符串("")。所以在使用 Input 前,用户可以通过检查 InBufferCount 属性来确定缓冲区中是否已有需要数目的字符。

InBufferCount 返回接收缓冲区中已接收的字符数。

格式为:MSComm. InBufferCount[= 0]。该属性在设计时无效,但设置 InBufferCount 属性为 0 可以清除接收缓冲区。

InputMode 设置或返回 Input 属性取回的数据的类型。

格式为:MSComm. InputMode[= ModeValue]。若数据只用 ANSI 字符集,设置 InputMode 属性值为 0(缺省),数据通过 Input 属性以文本形式取回。如数据中有嵌入控制字符、Nulls 等,可设置 InputMode 属性值为 1,数据通过 Input 属性以二进制形式取回。

Output 向发送缓冲区写数据。该属性在设计时无效,在运行时为只读。

格式为:MSComm. Output=OutData。

OutData 为要发送的数据,可以是文本数据或二进制数据。

MSComm 控件提供下列 2 种处理通讯的方式:查询方式和事件驱动方式。本文中的程序主要采用查询的方式。

### 3.2 应用程序实例

在应用程序中,主要采用 MSCOMM 控件与 PLC 进行通信,加载窗体时初始化通信参数,在此以读取数据为例说明编程的基本过程。

在进行通信之前首先要对 PLC 端进行编程初始化,初始化的程序如图 5 所示。

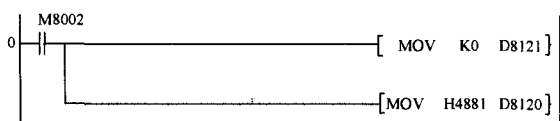


图 5 初始化程序

M8002 为特殊功能寄存器,只在 PLC 转为运行时接通一次。D8121 设定 PLC 站号寄存器,在这里设定 PLC 的站号为 0 号。通信控制寄存器 D8120 中移入 16 进制数 4881,对照参数表,PLC 端的设定为通信速率 9 600 b/s,无校验,8 位数据位,1 位停止位。采用 RS 232 接口,数据

帧中不添加校验位。上位机端要与之相匹配。

上位机读取数据程序如下:

```

Private Sub Form_Load()
    MSComm1.Settings = "9600,n,8,1"
    '设置串行通信参数为 9 600 b/s,偶校验,数据位 7 位,停止位 2 位,与 PLC 断相匹配。
    MSComm1.InputMode = comInputModeText
    '以 ASCII 方式进行收发
    MSComm1.CommPort = 1
    '选择串口 1
    If MSComm1.PortOpen = True Then
        '如果初次化时串口打开着就将他关闭
        MSComm1.PortOpen = False
    End If
End Sub
  
```

串口的初始化工作是在主程序初始化时完成的,除了设置通信参数和选择端口外,其他一些与串口有关的参数都采用默认值。

在按钮事件中进行数据读取,读取程序如下:

```

Private Sub Command9_Click()
    Dim Nowtime As Long
    Dim Count As Integer
    '定义等待延时所用变量
    If MSComm1.PortOpen = False Then
        '若串口未打开,则打开串口
        MSComm1.PortOpen = True
    End If
    Do While Count < 5
        '收发不正常,反复尝试 5 次
        Nowtime = Timer
        MSComm1.InBufferCount = 0
        '清输入缓存区
        MSComm1.OutBufferCount = 0
        '清输出缓存区
        MSComm1.Output = Chr(5) + "00FFBR0M013901"
        '送出读请求命令
    Do While MSComm1.InBufferCount < 7
        '等待接收数据
    If Timer > Nowtime + 1 Then
        Exit Do
    End If
    Loop
    If MSComm1.InBufferCount >= 7 Then
        Inputtemp = MSComm1.Input
        '将接收到的数据赋给中间变量进行处理
        Temp2 = Mid(Inputtemp,6,1)
        Exit Do
    End If
    Count = Count + 1
    Loop
    If Count >= 5 Then
        '若 5 次尝试错误,则认为通信出错
        Dummy = MsgBox("PLC 通讯错误",16,"系统提示")
        Exit Sub
    End If
  
```

程序中请求帧为 Chr(5) + "00FFBR0M013901",chr(5)为 ENQ 05,"00"为 PLC 站号在 PLC 的 D8120 中设定,"FF"为上位机告诉 PLC 的计算机号,"BR"为读取位状态指令,"0"为上位机告诉 PLC 的响应请求时间,表示 PLC 收到正确请求后马上响应请求,"M0139"为读起始元件号 M139,最后两位"01"表示读取的元件个数为 1 个。

(下转第 14 页)

以上 3 个按键重新设置。按下 S2 一次进入温度上限设置状态,再次按下 S2 进入温度下限设置状态,第 3 次按下 S2 则退出温度设置状态。在温度上/下限设定状态下,利用 S3 和 S4 可以完成温度值的设定。LCM1602A 用来显示被测介质的温度和环境温度,同时用来指示温度的变化趋势及系统的工作状态。LCM1602A 是  $16 \times 2$  的字符型 LCD 模块,他与单片机之间的接口有 8 位总线形式和 4 位总线形式 2 种,这里采用 4 位数据总线形式,此时,LCM1602A 的数据输入端中 D7D6D5D4 有效,而 D3D2D1D0 呈高阻状态。P10 脚接蜂鸣器,当按键按下时发出短暂的提示音,同时在 MLX90601 工作不正常时(此时 MLX90601 内部的温度寄存器中的相关位被置位),发出告警信号。

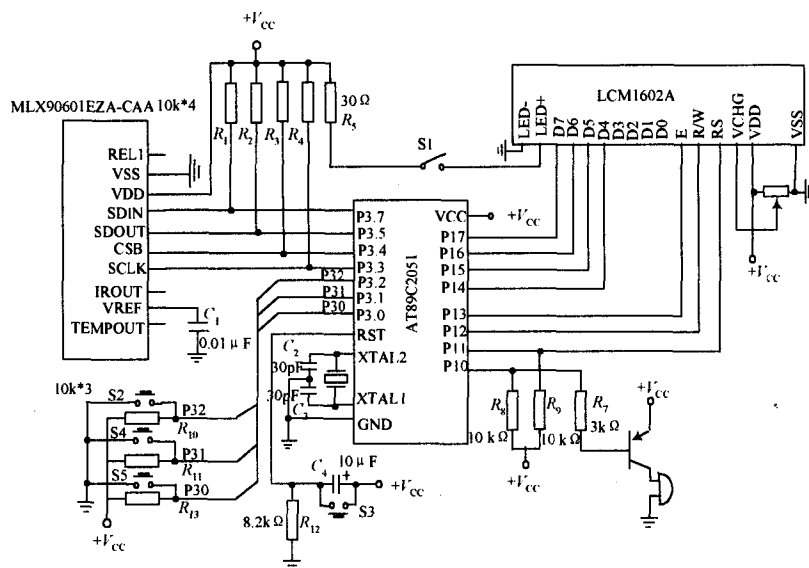


图 4 红外温度测量仪的硬件电路

### 3.2 软件编程

编程中需要注意的是在对内部 E<sup>2</sup>PROM、寄存器等进行操作时,应该严格按照其操作时序进行。例如,向

作者简介 杜晓 盐城师范学院物理系教师,主要从事电子技术方面的教学和科研工作。

(上接第 8 页)

### 4 结语

基于 VB6.0 环境下的 PLC 串行通信方式,能够应用于小型监控系统当中。由下位 PLC 作为现场控制器进行实时控制,上位计算机进行高级语言编程组成友好的人机界面进行操作控制,有实际的应用意义。

#### 参考文献

[1] 李玉东. Visual Basic 6.0 中文版控件大全[M]. 北京:电子

作者简介 周鸣男,1966 年出生,讲师。现从事计算机应用的教学及研究工作。

内部 E<sup>2</sup>PROM 中写入数据,应按照图 5 所示的流程图进行。该红外温度测量仪程序设计比较简单,在此不再赘述。

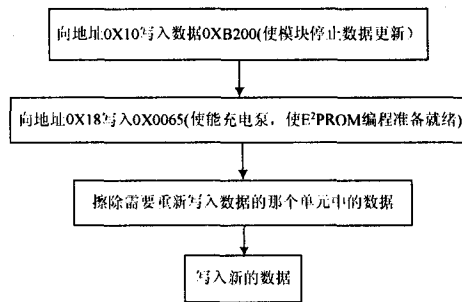


图 5 向模块内部 E<sup>2</sup>PROM 写数据流程图

需要注意的是对 E<sup>2</sup>PROM 不可使用块擦除和块写入命令,否则会将整个 E<sup>2</sup>PROM 中的数据全部擦除掉,可能导致 MLX90601 工作不正常。

### 4 结语

MLX90601 系列红外测温模块具有多种输出方式,能够实现非接触式测温,使用简单、方便,工作可靠,可广泛应用于电烤箱、微波炉及其他需要进行温度控制的场合。

#### 参考文献

- [1] MLX90601 DataSheet[Z]. Melexis Corporation, 2003, 1: 21-30.
- [2] 余永权. ATMEL89 系列单片机应用技术[M]. 北京:北京航空航天大学出版社,2002.
- [3] 李维隰,郭强. 液晶显示应用技术[M]. 北京:电子工业出版社,2001.

工业出版社,2000.

- [2] 郜焕平,马希荣. Visual Basic 6.0 程序设计[M]. 北京:机械工业出版社,2003.
- [3] 仵浩,宋燕杰,宋文. Visual Basic 串口通信工程开发实例导航[M]. 北京:人民邮电出版社,2003.
- [4] 蔡锦达,张文国,刘菁冉,等. 用 VB 实现 PC 机与永宏 PLC 的通信[J]. 计算机应用,2002, 22(7): 57-59.
- [5] 三菱公司. 三菱 PLC 通信手册.