

# 基于 VB 的上位机与欧姆龙 PLC 的串行通信

收藏此信息 打印该信息 添加：佚名 来源：未知

---

## 1 引言

在现代控制系统中，pc 机作为上位机负责系统管理、状态监控、信息处理和打印报表等工作，plc 作为下位机进行面向现场的实时控制已成为一种典型的系统结构。因此，pc 机与 plc 之间的通信问题便成了系统能否实现的关键。串行通信以其接线简单、容易实现的特点得到了广泛的应用。

visual basic 编程软件简单易学、功能强大，利用它可以很方便的实现 pc 机与 plc 间的串行通讯。本文以 omron 公司的 cpm2a plc 为例，介绍了利用 vb6.0 实现 pc 机与 plc 串行通讯的具体方法。

omron 公司的 cpm2a plc 支持 host-link 通信协议，可以与上位计算机进行通信。使用 host-link 通信协议构成的通信网络，可以很方便的实现上位 pc 对下位 plc 的实时监控。

## 2 原理设计

### 2.1 系统结构

系统结构如图 1 所示，计算机作为上位机，plc 作为下位机，因 omron cpm2a 本身带有 rs-232 串口，因此，利用 rs-232 电缆将其与 pc 机的串口(com1 或 com2)相连，便构成了一个 1: 1 的简单的通讯系统[1]。

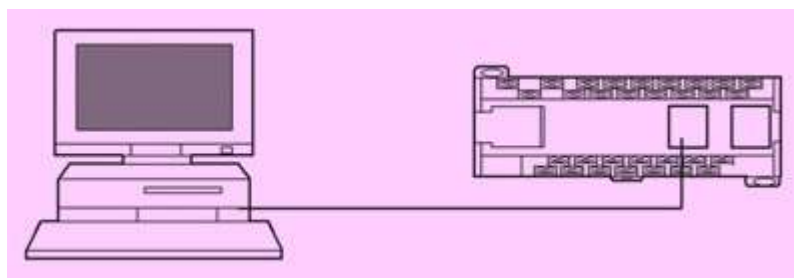


图1 系统结构

## 2.2 host-link 通信协议<sup>[2]</sup>

omron 公司的 host-link 通信系统是由上位计算机 (ibm pc 或兼容机) 通过安装在各台 plc 上的 host-link 单元连接多台 plc 构成的网络。上位机对系统中的 plc 进行集中管理与监控, 通过与 host link 单元的通信, 可以编辑或修改各台 plc 的程序, 实时监控其运行过程, 实现自动化系统的集散控制。对于小型 plc, 也可以通过其 rs 232c 通信端口进行链接。

系统使用 host-link 通信协议进行通信, 上位机具有传送优先权, 总是首先发出命令并启动通信, host link 单元收到命令交由 plc 执行, 然后将执行结果返回上位机, 二者以通信帧为单位, 轮流交换数据。通信时一组传送的数据称为“块”, 它是命令或响应的单位, 从上位机发送到 host-link 单元的数据块称为命令块, 反过来, 从 host-link 单元发送到上位机的数据块称为响应块。多点通信时, 可作为单帧发送的最大数据块为 131 个字符, 因此当一个数据块含有 132 个或更多字符时, 要分成两帧或多帧发送。

每个数据块都以设备号及标题开始, 以校验码 (fcs) 及结束符结束。响应块中还包括反应执行结果的响应码。通信格式如图 2 所示。



图2 通信帧格式

### 3 软件实现

#### 3.1 mscomm 控件介绍<sup>[3]</sup>

mscomm 控件全称为 microsoft communications control，是 microsoft 公司提供的简化 windows 下串行通信编程的 activex 控件，它为应用程序提供了通过串行接口收发数据的简化方法。

mscomm 控件有两种通信方式，查询方式和事件驱动方式，本文采用事件驱动方式。mscomm 控件的属性有许多，其中一些需要在编程时设置，另一些则取默认值即可，具体设置方法可参见相关资料，表 1 列出了 mscomm 控件的常用属性。

| 属性       | 说明                      |
|----------|-------------------------|
| CommPort | 设置并返回通信端口号              |
| Settings | 设置并返回数据传输率、奇偶校验、数据位、停止位 |
| PortOpen | 设置并返回通信端口的状态，也可以打开和关闭端口 |
| Input    | 从接收缓冲区返回和删除字符           |
| Output   | 向传输缓冲区写一个字符串            |
| OnComm   | 返回最近的 OnComm 事件的数字代码    |

#### 3.2 plc 参数设置<sup>[4]</sup>

plc 的通信参数可以采用默认的 host-link 通信形式，即 1 位启动位，7 位字长，偶校验，2 位停止位，9600bit/s。设备号为 00。

#### 3.3 上位机软件设计

使用 vb 通信控件 mscomm 编写 plc 监控程序，通过 rs-232 端口发送并接收数据，关键代码如下：

‘程序初始化并设置串口参数

```
private sub form_load()
```

```

mscom.comport = 1

mscom.settings = "9600, e, 7, 2"    `通信参数设置

mscom.inputmode= cominput

-modetext    `以文本方式接收数据

mscom.portopen = true    `打开串口

end sub

`向 plc 写入数据

private sub butwrite_click()

`检验串口是否关闭, 若串口已关闭, 则打开它

if mscom.portopen = false then

mscom.portopen = true

end if

`plc 必须处于编程状态才可以写入数据

plcend="@00sc00"    `“@”为“开始符”表示数据帧开始; “sc”为 host-link 指令;

“00”表示下位 plc 的设备地址, ascii 码表示[4]。

mscom.output = plcend + fcs(plcend) + "*" + chr$(13)

mscom.inbuffercount = 0

dim outstring as string

outstring="@00wr0010"+ trim(txtcontent.text)

outstring = outstring + fcs(outstring) + "*" + chr$(13) `采用异或校验, 即 fcs
s 校验; "*" + chr$(13) 表示结束符, 表示数据帧的结束, 即 "*↵" [4]。

mscom.inbuffercount = 0

```

```
mscom.output = outstring    `向 plc 发送数据

end sub

`从 plc 读取数据

private sub butread_click()

if mscom.portopen = false then

mscom.portopen = true

end if

dim outstring, instring as string

outstring="@00rr00100001"

outstring=outstring+fcs(outstring) + "*" +chr$(13)

mscom.inbuffercount = 0

mscom.output=outstring    `向 plc 发送读取数据的命令

dim i as integer

do while mscom.inbuffercount < 1

-t <= 14 or i > 9000    `i 为等待的时间

        i = i + 1

loop

        `提取有用数据并显示

if i < 9000 then

        instring = mscom.input

        txtcontent.text = "输出 010 通道的数据为" + mid(instring, 8, 4)

else
```

```
msgbox ("读取失败")

end if

end sub

`fcs 校验

function fcs(byval fcsstr as string) as string

dim slen, i, xorresult as integer

dim tempfcs as string

slen = len(fcsstr)

xorresult = 0

for i = 1 to slen

xorresult = xorresult xor asc(mid$(fcsstr, i, 1))

next i

tempfcs = hex$(xorresult)

if len(tempfcs) = 1 then tempfcs = "0" + tempfcs

fcs = tempfcs

end function
```

#### 4 结束语

本文给出了利用 vb(主要是其中的 mscomm 通信控件)开发通信程序, 实现了 plc 与 pc 普通串口通信的实用方法。实验表明, 该程序运行稳定可靠, 且操作简便。将程序稍加修改, 还可应用到其他工业控制设备, 在小型工业控制系统中具有一定的应用价值。