

# 工业控制计算机的多机通讯

西安理工大学自动化与信息工程学院 梁莉 李强 刘庆丰 陈军 熊玉亮

本文论述了在计算机工业控制系统中,工控机与多种不同外设之间的通讯软件的设计,分析了几种典型通讯协议下通讯程序的编制,以及串行通讯设备的连接方法。

本系统由一台S7-200PLC实现现场控制,ABB变频器实现速度控制,温度调节器AI708实现温度控制,5台单片机实现PID控制,以上各设备与工控机的通讯均

件驱动方式是可取的。

采用查询方式驱动的串行通讯程序,通过检查InBufferCount的属性值来判定输入缓冲区中是否接收到相应数目的字符或字节。若已接收到相应数目的字符或字节,就可以用Input属性来接收这些字符或字节;否则继续查询InBufferCount的属性值,直到满足条件。本系统根据多机通讯的需要,采用查询方式设计串行通讯程序。

## 2. 系统通讯协议

在系统设计中采用主从式通讯方式,由于几种外设的通讯协议不同,在此分别加以介绍:

(1) 变频器的通讯链路信息由几个信息元素组成。每个元素至少包括一个字符。信息元素和字符数在图2中列出。变频器的通讯链路信息的书写是从主机到变频器以四种不同的信息格式发送的。如果接收到的信息正确,变频器将返回一个ACK-字符;并伴随有信息的地址;类型和数据参数是可接受的。如果接收方发现校验和错误,将返回一个NAK-字符。在接收到的信息正确,但类型或数据参数不可接受时(高级别错误),变频器将返回一个EOT-字符。阅读信息与书写信息相同。阅读信息与书写信息的返回信息都包括一个校验和位(BCC)。校验和(BCC)按以下方式计算:

① 计算从STX到ETX间字符的和

② 如果结果低于32(20H),为了使校验和不与其他控制元素字符相混淆,则在加32(20H)。

(2) 计算机与PLC的通讯遵从MODBUS RTU协议。MODBUS RTU协议是一个主从协议,这就意味着一个网络配置包括一个主设备和一个或多个从设备,每个从设备有不同的地址。主机给从机发送请求,然后等待从机的响应。从机正确接收到请求则回答请求。如果请求没有正确接收到,则有象奇偶错或CRC(校验和)错这一类的传送错误,此时从机就不响应,主机必须在等待一



图1 系统框图

采用RS485接口标准,电量表实现电量的采集,采用POWER MEASUREMENT公司3720ACM。其中ABB变频器采用ASCII码传送字符串的方式进行通讯。系统框图如图1所示。

### 1. VB的串行通讯控件

VB5.0为我们提供了一个很重要的控件MSComm.OCX通讯控件,它通过改变对象属性,向对象发送信息以及为对象事件编写响应代码,就可以很方便的完成用户与应用程序之间的串行通讯。利用VB通讯控件编写程序,关键是要能够准确的理解并设置其属性。对InputMode属性,可以设为ComInput Mode Text或ComInput Mode Binary,即可分别以文本或二进制格式接收传入的数据。这点务必请读者注意。

采用Visual Basic的串行通讯控件编写通讯程序有两种方式:事件驱动方式和查询方式。

采用事件驱动方式的串行通讯程序,当输入缓冲区中接收到字符或是输出缓冲区空时就会触发MSComm控件的OnComm事件,串行驱动程序就立即得到控制权,处理相应程序,然后再将控制权返回到正在运行的应用程序。如果应用程序较小,并且是自保持的,采用这种事



个适当的时间后重新发送请求。

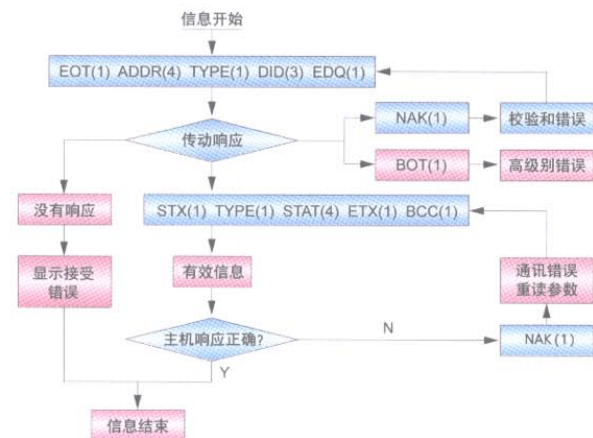


图 2 向变频器读操作流程

MODBUS RTU 协议传送 8 位二进制字符，每个字符包括一个起始位，一个或两个停止位，一个可选择的奇偶校验位。

以读单路/多路输入寄存器状态说明上位机发送、下位机应答的格式

上位机发送请求的帧格式为：

PLC 地址	功能 代码	数据地址 高低	数据长度 高低	CRC 校验码 高低

下位机应答的帧格式为：

PLC 地址	功能 代码	数据地址 高低	数据信息	CRC 校验码 高低

### (3) 3720ACM 电量表

3720ACM 选用 RS485 通信标准，3720ACM 通信协议是采取异步的主从通信为基础。在该协议中，远方设备是从站、仅对主站的请求作出响应。

该协议以 OSI (开放系统相互连) 模型为基础，在通信系统中，OSI 模型采取标准访问。

3720ACM 协议是 OSI 模型的一个简要的版本，它提供物理层、数据链路层和应用层，主要应用于对远方设备进行数据采集和控制。下面就数据链路层作详细介绍。

数据链路层负责错误帧的发送和接收。

帧格式

标有“保留”字样的字节和位作为备用必须置零。

域	Sync	Fmt	Cntl	Len	SRC	DST	Tran	Rsrvd	Data	CRC
字节	1	1	1	1	2	2	1	1	N	2

帧域描述如下：

· 帧同步：帧同步字节值为 14h，发送给主站和从站都如此。

· 帧格式：帧格式字节值为 CDh，该字节规定帧的类型是 3720ACM 型的帧。

· 帧控制：帧控制字由下列位域所构成：

位	含义
7	方向：0= 主站到从站，1= 从站到主站
6	帧类型：0=Data，1=Ack/Nak
5	若帧类型 = Data: 0=Ack/Nak disable, 1=Ack/Nak enable 若帧类型 = Ack/Nak: 0=Nak, 1=Ack
4-0	保留

· 帧的长度：描述帧长度。帧的有效长度从同步字节直到 CRC 字节是帧长度加 5。

· 源地址：源地址定义了帧激励器的地址。

· 目的地址：目的地址定义了帧响应器的地址。

· 执行码：

执行代码域长度为字节，其位域分解如下：

含义	保留	第一帧	计数器
位	7	6	5--0

在多个执行帧中，第一帧设置为 1。计数器从第一帧向下纪录直到 0 帧。单个帧的执行时第一帧的位为 1，计数器设置为 1。

· 帧数据：数据域传输到上层或从上层传输。

· 帧 CRC：CRC 域是双字节域，从帧的控制开始到数据域的最后一个字节为止，对所有位用 CRC-16 进行计算。注：在主机向从机发送时，一个数据的帧数据格式为：

**n 个参数功能码**

(2n 字节)

在从机向主机返回信息时，帧数据格式为：

**参数功能码(2 字节) 数值(4 字节) (以补码表示)**

如果 n 个参数，数据长度为 6n 字节

以上通讯在实际工业现场运行效果良好，考虑到各种通讯过程中可能会出现的一些实际干扰造成的误码问题，为了确保数据的安全性、完整性，实际通讯中应考虑打开端口时，MSComm Port Open=Ture 设定前，首先判断端口是否已打开，若打开，选 MSComm Port Open=False，然后再打开端口。另外采用查寻方式判断 InBufferCount 是否达到所需要的长度，此时一定要加超时处理以防死循环。目前许多组态软件都支持标准的外设通讯，但实时性有时达不到要求，且造价较高。上述的通讯方法也是一种途径，仅供读者参考。■