

计算机与 FX₂ 系列可编程序控制器的通信

徐春辉

(电子与信息工程系)

摘 要 介绍了计算机与 FX₂ 系列可编程序控制器的通信协议,设计了通信接口,并用 C 语言编制了通信程序¹⁹.

关键词 可编程序控制器;通信;接口;通讯协议

分类号 TN 919. 2

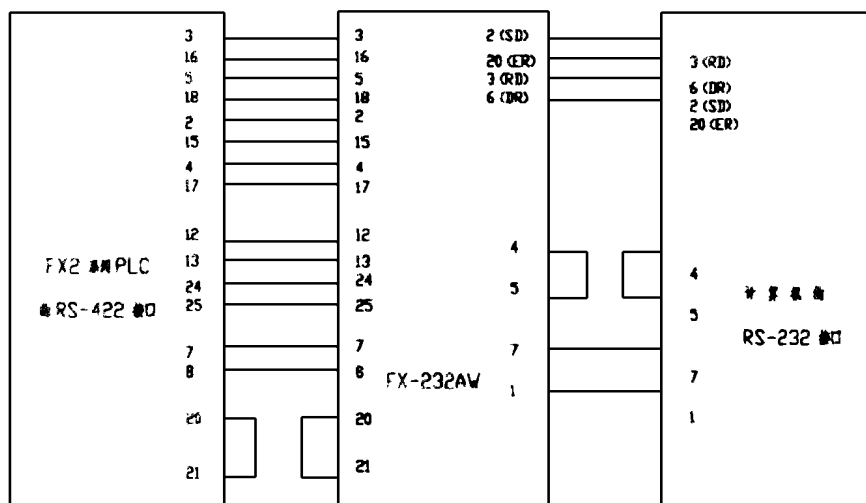
0 引 言

可编程序控制器(以下简称 PLC)由于其高可靠性、编程简单、通用性强、体积小、结构紧凑、安装维护方便等特点,而在工业控制中得到了广泛应用¹⁹.由于计算机在图形显示、数据处理、打印报表、中文显示等方面具有很强的功能¹⁹.因此在诸如柔性制造系统此类的现代化大型控制系统中,常常将多台 PLC 与计算机联接起来构成多级分布式控制系统¹⁹.在此类系统的构成中,关键问题之一是如何实现 PLC 与计算机之间的通信¹⁹.不同系列的 PLC 与计算机的通信方式各不相同,本文以三菱公司的 FX₂ 系列 PLC 为对象,分析了通信接口及通信协议¹⁹并用 C 语言编制了通信程序¹⁹.

1 FX₂ 系列 PLC 与计算机之间的通信接口

FX₂ 系列 PLC 的通信接口采用 RS-422 标准¹⁹.而计算机的串行口采用 RS-232 标准,因此在两者之间必须增加一块 RS-232/RS-422 通信转换板,这样才可以把通信标准从 RS-232 标准变成 RS-422 标准¹⁹.在此采用三菱公司开发的 FX-232AW 接口模块,并按图 1 的方式连接,构成了 FX₂ 系列 PLC 与计算机之间的通信接口¹⁹.

在图 1 中,由于计算机的 RS-232 接口的 4 脚(请求发送端)和 5 脚(清除发送端)短接¹⁹.因此,对计算机发送数据来说 PLC 总是处于数据就绪状态¹⁹.也就是说,计算机在任何时候都可以将数据送到 PLC 内¹⁹.又由于 RS-232 接口的 20 脚(数据终端就绪)和 6 脚(数据装置就绪)交叉连接,因此,对计算机接收数据来说,就必须检测 PLC 是否处于准备就绪的状态¹⁹.如果 6 脚为 1,则可以接收数据;如果 6 脚为 0,则必须等待,直到为 1 时,才可以接收数据¹⁹.

图 1 FX₂ 系列 PLC 与计算机之间的通信接口接线图

2 FX₂ 系列 PLC 与计算机之间的通信协议

2.1 数据通信的方式

FX₂ 系列 PLC 与计算机之间的通信采用异步通信方式,数据传输的速率固定为 9 600 bps,传送或接收的每帧数据由一位起始位、七位数据位、一位奇偶校验位及一位停止位共十位组成,具体格式如图 2 所示¹⁹。



图 2 数据格式的规定

奇偶校验位采用偶校验,七位数据位必须是 ASCII 码,也就是说 FX₂ 系列 PLC 与计算机通信中,所要传送或接收的各种信息都是以 ASCII 码的形式出现的¹⁹。FX₂ 系列 PLC 与计算机的通信中,所用到的 ASCII 码如表 1 所示¹⁹。

表 1 FX₂ 系列 PLC 与计算机通信所用到的字符及其 ASCII 码

字符		ENQ				ACK				NAK				STX				ETX			
ASCII 码		05H ¹⁾				06H ²⁾				15H ³⁾				02H ⁴⁾				03H ⁵⁾			
字符	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F					
ASCII 码	30H	31H	32H	33H	34H	35H	36H	37H	38H	39H	41H	42H	43H	44H	45H	46H					

1) 来自计算机的查询信号;2) 无校验错误时,PLC 对 ENQ 的应答信号;3) 检测到错误时,PLC 对 ENQ 的应答信号;4) 数据块的起始标志;5) 数据块的结束标志¹⁹。

2.2 通信信息的组成形式

FX₂ 系列 PLC 与计算机交流的信息可分为三种类型:连络信号、数据和命令码¹⁹其中连络

信号 ENQ、ACK 和 NAK 作为单个控制字符,可以构成单字符帧¹⁹。其余的信息在发送或接收时必须采用多字符帧的形式¹⁹。

计算机发往 PLC 的一个多字符帧由字符 STX、命令码、数据、字符 ETX 以及和校验五部分组成,如图 3 所示¹⁹。其中和校验值是将命令码到 ETX 之间的所有字符的 ASCII 码相加,取所得和的最低二位数(十六进制数)并将其转换为相应的 ASCII 码¹⁹。命令码部分的字符根据读、写、强制通、强制断等不同的操作而填入不同的命令代码,FX₂ 系列 PLC 与计算机通信所用到的命令码共有四个,如表 2 所示¹⁹。数据部分在读写不同情况下应该填入不同数据,一般前 4 个字节填写软设备的首地址,接着两字节是需要读写的字节数;如果是写命令,那么还要再跟有关的数据;如果是读命令那就无需数据了¹⁹。

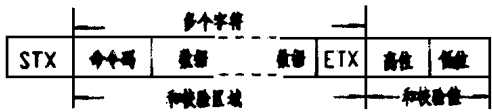


图 3 多字符帧的组成

表 2 FX₂ 系列 PLC 与计算机通信所用的命令码

命令码	操作对象	功 能
0	X,Y,M,S,T,C,D	读位软设备的状态或字软设备内的数据
1	X,Y,M,S,T,C,D	对位软设备置“0”或“1”和对字软设备写数
7	X,Y,M,S,T,C	对位软设备强制置“1”
8	X,Y,M,S,T,C	对位软设备强制置“0”

PLC 对计算机的应答信息也是以多字符帧的形式出现,其格式与图 3 相类似¹⁹。只是没有命令码部分,另外数据部分则是由计算机需要读取的位软设备的状态或字软设备内的数据组成¹⁹。

2.3 FX₂ 系列 PLC 与计算机通信的时序分析

在 FX₂ 系列 PLC 与计算机之间进行的通信中,PLC 始终处于一种“被动响应”的地位,无论是数据的读或写,都是先由计算机发出信号¹⁹。

在刚开始通信时,计算机首先要发送一个控制字符 ENQ,去查询 PLC 是否作好通信的准备,同时也可以检查一下计算机与 PLC 的连接是否正确¹⁹。当 PLC 接到该字符后,如果它处在 RUN 状态,则要等到本次扫描周期结束时才应答;如果它处在 STOP 状态,则马上应答¹⁹。若通信正确,则应答字符 ACK;若通信有错,则应答字符 NAK¹⁹。如果计算机发送了一个控制字符 ENQ,经过 5s 后,什么信号也没有收到,此时计算机应再发送二次控制字符 ENQ,如果还是什么信号也没有收到,则说明连接有错¹⁹。

在上述情况中,当计算机接收到来自 PLC 的应答字符 ACK 后,就可以进入数据通信了¹⁹。通信的时序如图 4 所示¹⁹。当计算机发送数据时,其 RS-232 接口上的 ER 端(即 20 脚)为高电平,这样,与其相连的 FX-232AW 接口模块上的 DR 端(即 6 脚)也为高电平,它表

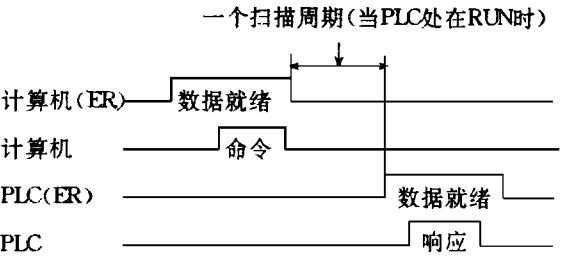


图 4 PLC 与计算机通信的时序图

示计算机的数据已就绪, PLC 可以接收数据了¹⁹。此时, PLC 被强制处于接收数据状态, 即使它做出应答, 也不会被计算机接收, 因此, 当计算机发送完数据后, 必须将 ER 端置为低电平, 这点由计算机 RS-232 接口上的 RS 端(即 4 脚)和 CS 端(即 5 脚)短接来保证, 即计算机发送完数据后, ER 端会自动变为低电平¹⁹。此时, 任何由 PLC 做出的应答都可以被计算机接收, 即计算机处在接收数据状态, 当计算机读取 PLC 对其指令的应答信号后, 复位通信线路, 表示本次通信完成¹⁹。

3 计算机与 FX₂ 系列 PLC 串行通信的实现

本文采用 C 语言实现计算机与 FX₂ 系列 PLC 的串行通信, 其核心是通过库函数 `int86()` 调用系统软中断 INT 14H 来对计算机的串行口进行操作, 从而实现数据的发送和接收¹⁹。

3.1 RS-232C 串行通信端口的初始化

FX₂ 系列 PLC 与计算机之间的通信协议规定: 传输速度固定为 9 600 bps, 奇偶校验位采用偶校验¹⁹。因此, 只需对计算机串行通信端口进行初始化¹⁹。

利用系统软中断 INT 14H 的服务 D7 D6 D5 D4 D3 D2 D1 D0 可以对计算机的串行端口 (COM 1、COM 2) 实现初始化, 其中 DX 寄存器用于存放端口号 (COM 1 为 0, COM 2 为 1), AL 寄存器用于存放初始化参数, 该参数编码为一个字节, 其含义如图 5 所示¹⁹。

根据 FX₂ 系列 PLC 与计算机之间的通信协议确定初始化参数为: 11111010B 即 FAH¹⁹。计算机串行通信端口的初始化可利用下面给出的函数

`portinit(int port, unsigned char code)` 实现¹⁹。其中端口号和初始化参数分别由变量 `port` 和 `code` 提供¹⁹。

```
/* initialize the serial port */
```

```
void portinit(int port, unsigned char code)
```

```
{ union REGS r;
```

```
  r.x.dx = port;
```

```
  r.h.ah = 0;
```

```
  r.h.al = code;
```

```
  int86(0x14, &r, &r);
```

```
}
```

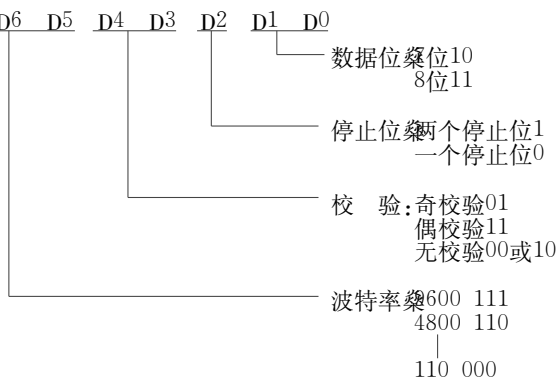


图5 初始化参数的定义

3.2 FX₂ 系列可编程控制器所规定的多字符帧的合成

为编程及操作方便, 本文提供函数 `SCS(char * string, char * SCS)` 和 `chang(char * ch)` 分别完成和检验码的计算与转换以及多字符帧的合成¹⁹。

SCS(char * string, char * SCS) 是对指针 * string 所指字符串从第一个字符到最后一个字符连续执行求和运算,而后将所得结果转换成两个 ASCII 码字符,并写入指针 * SCS 所指的字符串中,其源程序如下:

```
/* calculate value of scs */
void scs ( char * string, char * scs)
{
    char cha=0x00;
    int i=0;
    while ( * string!= '\0' ) ;
        {cha=cha+( * string);
        string++;}
}
```

```
* ( scs + 1) = ( cha & 0x0f) ;
* scs = ( cha & 0xf0) >> 4;
for ( i=0;i<2;i++)
{
    if ( * ( scs + i) < 0x0a)
        * ( scs + i) = * ( scs + i) + 0x30;
    else * ( scs + i) = * ( scs + i) + 0x37;
}
```

chang(char * ch) 将指针 * ch 所指字符串自动补加和校验码以合成一个完整的多字符帧,源程序如下:

```
/* chang the normal string to the PLC communicate string */
void chang(char * ch)
{
    char * sc = "";
    scs(ch,sc);
    strcat(ch,sc);
}
```

3.3 字符的发送与接收

由于发送或接收的信息其最基本的组成单位为字符,所以只要能够实现字符的发送与接收,就可以将信息按规定的字符串依次发送或接收以实现信息的交流¹⁹。

以下函数 send(int port) 与 receive(int port) 分别调用 INT 14H 的功能 1、功能 3 实现了字符的发送与接收¹⁹。

```
/* send a character out of the port */
unsigned char send(int port, char c)
{
    union REGS r;
    r.x.dx = port;
    r.h.ah = 1;
    r.h.al = c;
    int86( 0x14, &r, &r);
    return(r.h.ah);
}

/* receive a character from the port */
unsigned char receive(int port)
{
    union REGS r;
    while (!check(port))
        if(kbhit())
```

```
{
    getch();
    printf("Forced exit");
    exit(1);
}

r.x.dx = port;
r.h.ah = 2;
int86( 0x14, &r, &r);
if (r.h.ah & 128)
    {printf("read error");
    exit(1);
}

else return(r.h.al);
}
```

其中函数 `check(int port)` 是调用 INT 14H 的功能³ 检查端口接收数据是否准备好, 返回值为 1 即为准备好, 返回值为 0 为尚未准备好¹⁹.

```
/* check the statu of the port */
```

```
int check (int port)
{
    union REGS r;
    r.x.dx = port;
    r.h.ah = 3;
    int 86( 0x14, &r, &r);
    return( r.h.ah & 1);
}
```

4 结 论

本文所有程序均在 Turbo C 2.0 环境下调试通过¹⁹利用以上所给出的函数, 通过改变多字符帧中的命令码及数据即可实现计算机对 FX₂ 系列 PLC 的监视与控制¹⁹.

参 考 文 献

- 1 易传祿, 韩希尧¹⁹. 可编程序控制器应用指南¹⁹. 上海: 上海科学普及出版社, 1992 19.172~193
- 2 Johnson, David G. Programmable Controllers for Factory Automation. Marcel Dekker, Inc, 1987
- 3 Mitsubishi. Handy Manual Fx-Series Programmable Controller. 1991
- 4 区益善¹⁹. 计算机网络工程手册¹⁹. 北京: 电子工业出版社, 1994 19.433~443
- 5 鸿健¹⁹. C 语言高级程序员编程指南¹⁹. 北京: 学苑出版社, 1993
- 6 尹彦芝¹⁹. C 语言高级实用教程¹⁹. 北京: 清华大学出版社, 1992 19.182~222

The Communication between Computers and FX₂-Series Programmable Controller

Xu Chunhui

(Electronic and Information Engineering Department)

Abstract The communication interface and communication protocol between computers and the FX₂ series programmable controller is introduced and the communication program is designed by C language in this paper.

Key words programmable controller; communication; interface; communication protocol