

UNIX / XENIX 系统 PC 仿真终端软件设计

丁隆厚 (郑州解放军电子技术学院)

摘要:本文介绍了 UNIX / XENIX 系统 PC 仿真终端软件的设计思想,并给出了 UNIX / XENIX 与 DOS 间直接文件传输的实现方法。整个软件包可在各类 PC 机上运行。

UNIX 操作系统以其简洁可移植性好,功能强和效率高等特点受到广泛的重视,已推广到从微型机到大型机的各种计算机上。单是在微机领域,就存在着多种 UNIX 版本。从国内 UNIX 的使用情况来看,虽然为用户提供了 PC 仿真终端软件,但该类软件大部分都没有提供 UNIX 与 DOS 之间的通信功能,这对同时使用 UNIX 和 DOS 的用户来说,不能不说是一大遗憾。笔者通过对 UNIX 终端系统的分析,成功地在 AT&T UNIX 以及 XENIX 上开发了适于各类 PC 机的仿真终端软件,并且开发了 UNIX / XENIX 与 DOS 之间的直接通讯,可实现它们之间的文件传输。其软件设计简单,运行可靠,使用方便,在使用中收到了良好的效果。

一、UNIX 系统终端工作过程简介

终端是用户和系统之间的接口,用户注册进行系统是通过控制终端来进行的。UNIX 系统启动后,生成了 INIT 进程(进程 1),INIT 执行一个无限循环,从文件"/ETC/INITTAB"中得到进入系统状态如"单用户"状态或"多用户"状态时应该执行的指令。在多用户状态下,INIT 允许用户从控制终端上注册,它生成一个叫做 GETTY 的进程,该进程打开一个特定的终端线路并在 OPEN 中睡眠直到机器与该终端在硬件上连通为止。当 OPEN 返回时,GETTY 启动 LOGIN 程序,LOGIN 要求用户输入注册名和口令,如果用户注册成功,LOGIN 执行 SHELL,用户开始工作。为了交互地使用 UNIX 系统,在终端和 UNIX 之间,由终端驱动程序控制终端上的数据传输,终端驱动程序包含一个到内部通信线路规范模块的接口,它负责解释输入和输出。终端对数据的处理可以分为两种方式:①"加工"方式。在

该方式下,通信线路规范在把数据送往接收进程之前先把从键盘上输入的原始数据序列转换成加工好的格式(即用户实际表示的意义);通信线路规范还把进程写出的原始输出序列转换成用户要求的格式。②"原始"方式。在该方式下,通信线路规范不作任何转换地在进程和终端之间传送数据。

通信线路规范的功能是:

- (1)把输入串分成行;
- (2)处理"抹字符"和"抹行"字符;
- (3)把接收到的字符回写到终端上;
- (4)把诸如制表符这样的输出扩展为一个空格串;
- (5)生成到进程的信号,表示终端挂起,输入行间断或用户出入的中断键;
- (6)支持不解释特殊字符如"抹字符"字符"抹行"字符或回车键的原始方式。

用户可以用 IOCTL 系统调用法的应该使用什么样的通信线路规范,并设置终端参数。如设置"抹字符"字符和"抹行"字符;控制终端是否回应其输入,设置终端波特率;清输入和输出字符队列;或者手工开始或结束字符输出等,并可恢复当前设置的值。对于 IOCTL 调用方式,其具体命令和有关数据结构,在此不再叙述,请详细参阅文献[2]。在这里仅介绍一下我们所涉及到的有关部分。

1.速度、字符长度和奇偶性

终端通信线路的速度、字符长度和奇偶性,可由 /USR/INCLUDE/TERMIO.H 中终端信息处结构(STRUCT TERMIO)中 C-CFLAG 的标志位来设置,其中可设波特率(50 到 19200),字符长度(5 位到 8 位),终止位的个数(1 或 2)以及奇偶性(偶数、奇数或无)。

XINIX 内核默认的控制终端波特率为 9600,字符长度为 8 位。

2. 字符映象

输入中的“换行”到“回车”和“回车”到“换行”的映象由 STRUCT TERMIO 中 C-IFLAG 的标志 INLCR 和 ICRNL 控制,对于按下“回车”键时既输入回车符又输入换行符的终端来说,回车被忽略(IGNCR)。

对于输出,通常要把一个“换行”映象成一个“回车-换行”对;C-OFLAG 中的标志 ONLCR 完成这项工作。其它标志则把“回车”变为“换行”,并且抑制在第 0 列上的“回车”。

3. 流控制

可以由用户按下 CONTROL-S 或通过调用带有 TCXONC 命令的 IOCTL 的一个进程来随时停止向终端的输出。可通过用户打入 CONTROL-Q 或通过 IOCTL 重新启动流。

4. 回应

正常情况下,终端以全双工方式工作,这种方式意味着数据可以在两个方向上同时流过通信线。因而,计算机(不是终端)回应敲入的字符以确认这些字符是否被正确接收。通常的输出字符映象也适用,例如,把一个回车映象成一个换行,而当回应时,这个换行被映象成一个回车和一个换行。清除 C-EFLAG 中的 ECHO 标志,可以关闭“回车”。

5. 及时输入

正常情况下,输入字符排成队列直到形成一行为止,该行的结束由一个“换行”或一个 EOT 指明。仅当此时行中的任何字符才对 READ 是可用的,该 READ 可能只是询问或者得到一个字符。在许多应用中,比如屏幕编辑程序和“表格式输入”系统,执行读的进程在字符敲入时就想读它们,而不等待汇集成一行,在这种情况下“行”的概念不再有任何意义。如果清除标志 ICANON,则输入字符中读之前,不需汇集成一行。因此,不能用“删除”和“抹去”符进行编辑。此时,删除和抹去符失去了它们的特定含义。

UNIX 系统控制终端工作可以分为以下四个过程:用户注册过程,输入 SHELL 命令行(或字符)过程,字符输出过程和用户退出过程。当控制终端(非控制台终端)打开电源后,终端首先发一个回车信号,系统收到该信号

后,由 GETTY 进程启动 LOGIN 程序,LOGIN 提示用户输入注册名和口令,如果注册成功,则提示用户输入终端类型,然后 LOGIN 程序就转到该用户的主目录下,启动用户选择的 SHELL 程序,此时,SHELL 即可从标准输入中读入命令行,并根据上述系统所选空的模式来解释它,然后将结果返回终端。当用户退出系统时,可通过在终端上按 CONTROL-D 键,来终止该用户进程。

二、PC 仿真终端软件设计概述

从上面介绍 UNIX 终端的基本情况来看,如果 PC 机能够按照上述要求,选择与之匹配的通信方式,即可通过 RS-232 标准接口实现与 UNIX / XENIX 系统的互连。其不带 MODEN 的硬件连接如图 1 所示。

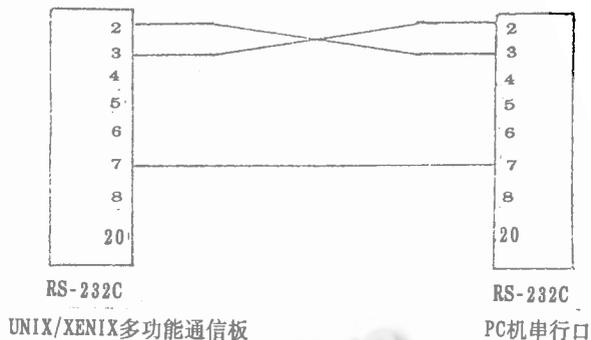


图 1

与 UNIX 系统控制终端相一致,PC 仿真终端也需要四个过程。下面来详细地介绍这几个过程其软件的设计思想。

1. 用户注册过程

用户注册过程由以下几步组成:

(1)RS-232 接口初始化。该过程完成 RS-232C 串行口的初始化工作,实现 PC 机与 UNIX 系统在通信方式的匹配,其汇编程序代码如下:

```
initasy proc nea
mov dx,3fbh;设置9600波特率
mov al,80h;
out dx,al;
mov dx,3f8h;
mov al,12;
```

```

out    dx,al;
xor    al,al;
mov    dx,3f9h;
out    dx,al;
xor    al,3 ;字长选择8位
mov    dx,3fbh;
out    dx,al;
mov    al,0bh ;设置DTR / RTS / OUT2
mov    dx,3fch;
out    dx,al;
mov    al,1 ;允许接收数据在效中断
mov    dx,3f9h;
out    dx,al;
ret

```

initasy endp

·
·
·

(2)发送 PC 仿真终端启动信息。通过 RS-232C 接口发送“回车”(ODH)字符。

(3)用户注册提示信息的接收和进行用户注册。当 PC 机发出“回车”后,处于接收状态,首先接收 LOGIN 提示信息,等到上述信息接收完成后,用户即可输入用户名进行注册。当用户名输入完成后,等到接收:PASSWORD:“信息(如果口令缺省,则自动省略这一步),然后等待用户输入口令。当用户口令输入正确时,则提示用户输入终端类型。一般 PC 机可选择 ANSI 类型,该方式可在 PC 机启动时在 CONFIG.SYS 中设置可安装设备驱动程序命令行“DEVICE=ANSI. STS”即可。用户输入 ANSI 或回车后,即完成了用户注册过程,UNIX 返回 SHELL 状态提示符“\$”(一般用户)或“#”(超级用户)即进入输入 SHELL 命令行过程。

2.SHELL 命令行及字符输入过程

正常情况下,UNIX 系统终端工作在“回应(ECHO)”方式下,因此命令行字符的输入可采用不带回车的字符输入方式(DOS 系统功能调用 8),输入一个字符,立即通过 RS-232C 口发送,UNIX 收到该字符后,应回应一个同样的字符,即 PC 机通过 RS-232C 接口接收该字符,如果收到的字符与所发送的字符一致,则在 CRT 上显示

该字符,否则重发。其字符的发送过程汇编程序代码如下:

```

send   proc    near;发送一个字符
sel:   mov     dx,3fdh
       in      al,dx
       test    al,20h
       jz     sel
       mov    dx,3f8h
       mov    al,bl
       out   dx,al
       call  rece ;接收回应字符
       cmp   al,bl
       jne   sel
       ret

```

send endp

rece proc near ;接收一个字符

```

se0:   mov     dx,3fdh
       in      al,dx
       test    al,1
       jz     re0
       mov    dx,3f8h
       in     al,dx
       ret

```

rece endp

3.字符输出过程

当终端输入一个“回车”字符后,表示命令行的结束,此时 UNIX 系统实现对 SHELL 命令行的解释和处理,然后将处理结果进行输出。由于 SHELL 命令的执行结果是不可预测其输出长度的,因此在 PC 机进行字符接收处理时,当接收完系统送来的字符后,要能够适时地退出字符接收过程,以便于重新输入新的命令行或字符信息。鉴于此,我们通过查寻方式来实现字符的接收处理,在实现字符的接收时,当接收完一个字符后,若在一定的时间间隔内没有收到任何字符,则退出字符接收过程。下面子程序 RECEIVE 给出了字符接收部分的汇编代码。程序中,等待时间间隔由 COUNT0 和 COUNT1 这

两个参数决定。如果正常接收一个字符,则将 C 标志清 0,如果在规定的时间内没有收到任何字符,则表示字符接收过程完成,将标志 C 置 1,退出。对于一般的 SHELL 命令而言,当 COUNT0 和 COUNT1 分别为 2 和 Q8000H 时,即能很好地满足我们的要求。

```

receive proc near
    bush cx
    mov bl,count0 ;等待时间控制数Q
re2:    mov xc,count1 ;等待时间控制数1
re1:    mov dc,3fdh
        in al,dx
re3:    test al,1
        jnz re1
        dec bl
        jz re5 ;等待时间到
        jmp re2
re4:    mov dx,3f8h
        in al,dx
        cld ;置接收字符标志
        pop cx
        ret
re5:    stc ;置接收完成标志
        pop cx
        ret
receive endp

```

由于字符的显示需要一空的时间,如果接收一个字符显示一个字符,则可能会丢失当前所送来的字符。因此,字符的接收,可采用缓冲技术,将所接收的字符暂存于内存,当接收过程完成或已接收的字符已达到缓冲区最大容量时方可进行显示处理。对大量字符信息的接收,可采用流控制技术,当接收的字符接近缓冲区最大容量时,可发送一个 CONTROL-S(13H)字符,通知 UNIX 系统暂停向终端的输出。当目前缓冲区的信息显示完成后,再发任一字符重新启动系统向终端的输出。在这里必须注意,在发送 CONTROL-S 后,还必须进行接收处理,因为在发送该字符的同时,系统仍然可能有字符送出。

4. 用户退出过程

当仿真 PC 终端用户完成所请求的工作时,即可按 CONTRLO-D(04H)键,退出 UNIX 的使用,此时 PC 终

端返回到 DOS 状态。

三、UNIX / XENIX-DOS 通信功能实现

上述 PC 仿真终端的环境建立好后,即可实现 UNIX / XENIX 与 DOS 之间的通信。在此环境中我们实现了 UNIX / XENIX-DOS 之间 ASCII 码源文件的传输,以实现 UNIX / XENIX 文件拷贝到 DOS 和 DOS 文件拷贝到 UNIX / XENIX。

1. UNIX / XENIX 文件到 DOS 系统的拷贝

UNIX 提供了 CAT 命令可以实现文件的显示,CAT 顺序地读文件的内容,并把它写到标准输出上。如果在 PC 终端上,将接收的输出信息以给定 DOS 文件名存盘,即可实现文件从 UNIX 到 DOS 的拷贝。该功能实现比较简单,故在此不再详叙。

2. DOS 文件到 UNIX / XENIX 的拷贝

由于 UNIX 没有提供直接从串行口接收信息的命令,因此,如果要实现 DOS 文件拷贝到 UNIX,则必须在 UNIX 系统下编制一程序,能够实现从串行口直接接收信息。然后,由于 PC 仿真终端可以挂接在任一串行口上,而其串行口什么时候激活,什么时候进行的是所拷贝的文件信息传输,都必须进行判断。要实现这样的功能是极其复杂的。那么能否通过别的方式来实现 DOS 文件到 UNIX 和拷贝呢?由于 PC 仿真终端是 UNIX 系统下的一个终端,如果把 DOS 文件内容看成是直接从终端上所输入的信息,然后再将它们以给定的文件名存盘,这样也就实现了 DOS 文件到 UNIX 的拷贝。我们在 UNIX / XENIX 系统下编制了下面程序 DSO COPY .C,通过在 PC 机终端上发送命令 DOS COPY FILENAME ↓,然后在 PC 机终端上从串行口将文件 FILENAME 发送,UNIX / XENIX 即可将所送信息以 FILENAME 为文件名存盘。这样也就实现了文件从 DOS 到 UNIX / XENIX 的拷贝。

```

/* doscopy.c */
#include <stdio.h>
#include <fcntl.h>
#include <termio.h>
#define MAXBUFF 2048
main(argc,argv)

```

```

int atgc;
char * atgv[];
{
int fd;
int i;
char c,d;
char buff[MAXBUFF];
struct termio tbuf,tbufsave;
if ((fd = creat(atgv[1],0666)) == -1)
    syserr("creat");
if (ioctl(O,TCGETA,&tbuf) == -1)
    syserr("ioctl1");
tbufsave = tbuf;
tbuf.c_lflag &= ECHO;
if (ioctl(O,TCGETA,&tbuf) == -1)
    syserr("ioctl2");
nbuf:
while(((c = getc(stdin)) != EOF)&&(i < MAXBUFF)){
    if(c == oxa)
        if((d = getc(stdin)) == Oxa){
            ;
        }else{
            buff[i] = c;
            i++;
            buff[d] = d;
            i++;
        }
    buff[i] = c;
    i++;
}
if(i > 0)
    write(fd,buff,i);
i = 0;
if(c != EOF)
    goto nbuf;
close(fd);
if(ioctl(0,TCSETAF,&tbufsave) == -1)

```

```

    syserr("ioctl3");
}

```

在 DOS COPY.C 程序中,首先以命令行参数 FILENAME 为名创建一文件,然后保存当前终端状态,这时清除终端 ECHO 标志,使终端工作在无回应方式,然后再接收从终端送来的信息,并将信息存盘。当收到终端文件发送结束的信号 CONTROL-D 时,退出信息的接收,关闭所写文件。而后恢复原终端状态。

在 PC 机终端一侧,若发送了 DOS COPY FILENAME ↓ 命令行后,在一定的时间间隔内,没有收到任何信息(出错),则可读 FILENAME 文件,实现其发送处理。

3. UNIX / XENIX 与 DOS 文件的兼容性

UNIX / XENIX ASCII 码源文件除了“回车”和“换行”的字符映象特征外,其系与 DOS ASCII 码文件完全兼容。在正常方式下,UNIX / XENIX 文件拷贝到 DOS 系统,其一个“换行”要映象成一个“回车-换行”,而 DOS 文件拷贝到 UNIX / XENIX 系统,则要将“回车”变成“换行”,因此,在上面 DOS COPY.C 程序中,当出现了 2 个 QXA(“换行”)字符时,要去掉一个 QXA 字符,当然也可以通过设置 IGNCR(忽略回车)标志来实现“回车”的忽略。

四、结束语

利用上述设计思想,我们成功地用汇编语言实现了 PC 仿真终端软件包。除了 VI 编辑程序,由于现有 DOS ANSI 控制字符不能与 UNIX / XENIX ANSI 字符完全兼容而不支持部分编辑功能外,其余部分使用都很正常。上述缺陷可通过修改 DOS 系统 ANSI.SYS 设备驱动程序来加以完善。

参考文献:

- [1][美]M.J.巴克,UNIX 操作系统教程原理与设计,《计算机研究与发展》编辑部,1989
- [2]中国科学院软件研究所编译,XENIX 系统 V 丛书,北京科学技术出版社,1989
- [3]段小航等编译,UNIX 系统高级程序设计,中国铁道出版社,1991