

汉字 FoxBASE 下换页命令 的正确使用

夏吉伟 (贵州遵义师范高等专科学校)

汉字 FoxBASE 是目前用途较广的一种数据库管理系统,但系统中的换页命令 EJECT 却起不到换页的作用,致使一些报表不能连续打印。如果连续打印,就会把报表打印在分页线上,既不美观,又不便于保存,只好在打印时手动调整纸的位置,很不方便。经分析,引起换页命令 EJECT 不起作用的原因是:普通打印机必须使用汉字驱动程序,而该程序修改了 17H 号中断,使打印机工作在图形方式下,致使 EJECT 不能换页。

解决的办法是:如果打印机含汉字库,可以不用汉字打印驱动程序;如果打印机不含汉字库或者要使用打印机的汉字库以外的字体时,首先从中断向量表中取出驱动程序修改后的 17H 号中断的段地址和偏移地址,放入中断向量表中为用户保留的某中断向量,然后把 MS-DOS 的 17H 号中断的段地址 0F000H 和偏移地址 0EFD2H 放入中断向量表中的 17H 号中断向量,执行 EJECT 命令,最后恢复驱动程序修改的 17H 号中断。

用 FoxBASE 系统本身解决该问题存在一定的困难,但可以利用该系统的 RUN!/命令。用 Turbo c 2.0 编制两个小程序,程序 ZDSA.C 保存驱动程序修改的 17H 号中断的段地址和偏移地址,并将 MS-DOS 的 17H 号中断的段地址和偏地址放入中断向量表中;程序 ZDLO.C 恢复驱动程序修改的 17H 号中断。

使用方法是:编写打印报表等文件时,在需要执行换页命令的地方,先执行 RUN ZDSA,再执行 EJECT;最后执行 RUN ZDLO。该方法在兼容的 286 和 386 微机、LQ-1600K 和 PB-9124 打印机、汉字系统 SPDOS、UCDOS 3.0 和 2.13H 上通过。经多次使用,效果较好。

程序如下:

```

/* ZDSA.C */
#include "dos.h"
main()
{
    union REGS in,out;
    struct SREGS segregs;
    in.h.ah = 0x35;
    in.h.al = 0x17;
    intdosx(&in,&out,&segregs);

```

```

    poke(0x0000,0x0180,out.x.bx);
    /* 将偏移地址保存在 0000:0180 和 0000:0181 单元 */
    poke(0x0000,0x0182,segregs.es);
    /* 将段地址保存在 0000:0182 和 0000:0183 单元 */
    in.h.ah = 0x25;
    in.h.al = 0x17;
    segregs.ds = 0xf000;
    /* 0xf000 是 MS-DOS 的 17H 号中断的段地址 */
    in.x.dx = 0xefd2;
    /* 0xefd2 是 MS-DOS 的 17H 号中断的偏移地址 */
    intdosx(&in,&out,&segregs);
}
/* ZDLO.C */
#include "dos.h"
main()
{
    union REGS in,out;
    struct SREGS segregs;
    in.h.ah = 0x25;
    in.h.al = 0x17;
    segregs.ds = peek(0X0000,0X0182);
    in.x.dx = peek(0X0000,0X0180);
    intdosx(&in,&out,&segregs);
}

```

再谈 FoxPRO 下软盘驱动器 状态的检测

何亚群 (中国矿业大学)

自从 FOX 软件公司推出 FOXPRO 系列软件以来,由于其在功能上的突破,编程时体现出巨大的灵活性。然而这样优秀的软件也并非完美无缺,本文所述的就是因为 FOXPRO 系列软件(即便是 FOXPRO2.6)没有提供对软盘驱动器状态的测试函数,造成当软盘驱动器未准备好而对软盘操作时系统中断这一问题。《《计算机系统应用》》95 年第 5 期中,陈学中同志撰写的“FOXPRO 下实现对软盘驱动器状态的检测”一文,对该问题提出一个解决办法,其思想是采用 INT13 的 0、2、3 号功能对软盘进行复位、读写操作,从而实现对软盘驱动器状态的检测。笔者根据文中提供的手段,对软盘驱动器的状态进行测试,在测出如文中所述软盘驱动器的各种状态的同时也发现了一些值得改进的问题。

测试软盘驱动器的状态主要应用在含有对软盘操作的程序中,如管理系统对数据库数据的备份以及已备份数据的恢复等模块的设计中。由于 FOXPRO 未提供对软盘驱动器状态的测试函数,因此当软盘驱动器未准备好时,系统会在 FOXPRO 系统中出现 DOS 错误提示信息,

屏幕下方出现 DISK ERROR:ABORT, RETRY, IGNOR 等提示,同时屏幕画面上移。其后果主要是界面的完整性完全损坏。笔者在使用了陈学中先生提供的程序后发现,尽管该程序能够测出当前软盘驱动器的错误类型,但在屏幕下方仍然出现 DOS 错误提示信息,而且界面依然被破坏。

数据库管理信息系统对软盘驱动器进行操作时,在软盘驱动器未准备好的情况下,不仅应该提示用户注意,而且还应使界面的完整性得以保持。针对这一要求,笔者用 PASCAL 语言编写了一段程序,程序中使用 PASCAL 语言提供的 IORESULT 函数,利用该函数返回最近一次 I/O 操作状态(以整数值表示)的功能,实现对软盘驱动器的测试。为了使 IORESULT 函数报告 I/O 操作出现的错误,需在编译时关闭 I/O 检查开关,即采用 {\$I-} 状态编译。否则当系统发现 I/O 错误时,程序将终止执行,IORESULT 函数将起不到报告错误的作用。IOR-ESULT 函数与软盘驱动器操作有关的返回值如下表:

返回值	错误类型
0	未发现错误
100	读错误
101	写错误
150	磁盘写保护
152	驱动器未准备好
154	CRC 校验错
156	磁盘定位错
157	未知介质类型
158	扇区找不到

本文所举的只是一个使用 PASCAL 语言的 IORESULT 函数判断软盘驱动器是否准备好的简单例子,其思路是向目标软盘驱动器建立一个子目录,然后观察 IORESULT 函数的返回值是否为 0,若 IOR-ESULT < > 0 则说明在软盘上建立子目录这一操作失败,此时在屏幕上提示用户检查当前目标软盘驱动器的状态,采用 REPEAT 循环语句,直到 IORESULT = 0 为止。读者可根据这一设想,举一反三对软盘驱动器的各种状态进行测试。程序的源码如下,该程序用 PASCAL6.0 编译运行通过。用 FOXPRO 的 RUN 语句或将其编译成 BIN 文件由 FOXPRO 程序装载运行皆可。

```
program checka;
uses crt;
```

```
var a:boolean;
    inkey:integer;
Function GetKey:Integer;
Var
    ch      :Char;
    km      :Integer;
Begin
    ch:=ReadKey;
    km:=ord(ch);
    If km=0 Then
    Begin
        ch:=ReadKey;
        km:=256+ord(ch);
    End;
    GetKey:=km;
End;(* GetKey *)
begin
{$I-}
a:=true;
repeat
mkdir('a:\emptdir');
if iorresult < > 0 then
begin
textcolor(white);
textbackground(red);
window(28,11,50,14);
clrscr;
window(1,1,80,24);
gotoxy(30,12);
writeln('软盘未放好或写保护!');
GOTOXY(30,13);
WRITELN(' 按任意键继续。');
inkey:=getkey;
window(28,11,50,14);
clrscr;
a:=false;
end else a:=true;
until a;
rmdir('a:\emptdir');
textcolor(blue);
textbackground(blue);
window(20,9,62,18);
clrscr;
end.
```

该程序有效地解决了由于软盘驱动器未准备好而使 FOXPRO 屏幕遭到破坏这一令人烦恼的问题。但 FOXPRO 在运行 EXE 文件时,FOXPRO 程序中所定义的窗口在测试软驱时会被暂时清除,测试结束后(即外部文件运行结束后)窗口会自动恢复。这是由于在 PASCAL 程序中书写提示信息造成的,因此建议在实践中采用传递参数的方式,用 FOXPRO 写提示信息。更好的办法是先装载编译成 BIN 的文件,再调用运行该模块,这样将会更完善地解决 FOXPRO 系统软件对软盘驱动器测试这一问题。