

FoxPro 与汇编语言的接口

谢苏琨 (南京东南大学)

摘要:本文介绍了 FoxPro2.5 与汇编语言程序的接口技术,并给出了两个非常有用的实例。

FoxPro2.5 提供了 3 条指令用于二进制程序的调入,执行和释放。

这三条指令是:

LOAD <二进制程序名>

CALL <二进制程序名>[WITH<参数>]

RELEASE MODULE<二进制程序名>

LOAD 指令把二进制子程序从磁盘装入内存,CALL 指令调用已装入内存的二进制子程序,默认的文件名后缀为".BIN"。文件装入时,文件名成为模块名,扩展名不再使用。这些指令的特点是:二进制程序虽以外部文件的形式载入内存,但 FoxPro 把它们当内部过程看待,因此执行速度很快。

FoxPro2.5 最多允许同时把 16 个二进制程序装入内存,每个程序限制在 32KB 以内。供 FoxPro 调用的汇编语言子程序必须严格遵循以下规则:

1.偏移量 0 处必须为一条可执行指令;

2.程序运行所用的空间不得超过实际程序的大小,因为 LOAD 指令是以文件的长度确定所需内存的大小的,也就是说程序不能在运行中动态申请内存;

3.不得改变 DS:BX 传递过来的变量长度;

4.控制返回 FoxPro 前要注意恢复重要寄存器到调用前的状态;

5.程序结束时用远返回把控制交给 FoxPro,即用 RETF 结束程序,或指定子程序的属性为 FAR。

CALL 指令运行已装入内存的二进制模块,它可以带有参数,该参数的起始地址由寄存器 DS:BX 指定,如果没有参数,则 BX 寄存器的内容为 0。二进制子程序通过参数与 FoxPro 传递信息:二进制子程序通过参数获得数据,也可以把数据反馈给某个参数,但要注意反馈的参数长度不得超过原数据长度。参数可以是 C 型表达式,也可以是除 S 型以外的任意类型的内存变量,包括数组(整个数组或数组的单个元素),以字符型表达式最常见。

RELEASE 指令释放指定的二进制模块,即释放该二

进制程序及其数据占用的内存。

利用汇编语言开发具有某些特殊功能的子程序,可以弥补 FoxPro 某些功能的不足,充分利用汇编语言底层功能强大、速度快的优点,使它们相得益彰。下面举两个例子说明,它们均成功地用于笔者为某单位开发的大型管理系统中。

例一:用汇编语言实现 FoxPro 建立子目录的功能

我们知道,FoxPro2.5 中没有建立子目录的指令,而我们有时需要这个功能,例如为自己开发的系统制作安装程序时,需要在程序中建立子目录。下面的汇编子程序实现此功能。

```
;name    mkdir.asm
pushall   macro
    push ax
    push bx
    push cx
    push dx
    push bp
    push si
    push di
    push ds
    push es
    pushf
endm

popall    macro
    popf
    pop es
    pop ds
    pop di
    pop si
    pop bp
    pop dx
    pop cx
    pop bx
    pop ax
endm

cseg segment para 'code'
assume cs:cseg,ds:cseg
mkdir     proc far
start:

    pushall
    lea di,cs:dirpath
    mov si,bx          ;DS:BX为子目录名的地址
zz1:      lodsb
    cmp al,0
    jz zz2
    mov cs:[di],al
    inc di
    jmp zz1

zz2:
```

```

mov byte ptr cs [di],0
mov ah,39h
push cs
pop ds
mov dx,offset dirpath;建立子目录
int 21h
popall
ret
mkdir      endp
dirpath    db 80 dup(0)
cseg       ends
end start

```

说明:(1)二进制模块产生过程:

- ①masm mkdir;
 - ②link mkdir;
 - ③exe2bin mkdir.exe mkdir.com
 - ④ren mkdir.com mkdir.bin
- (2)FoxPro 调用过程:
* 建立子目录的过程

```

proc makedir
para dirname
dirname=dirname+chr(0)
load mkdir
call mkdir with dirname
release module mkdir
return

```

(3)如果要在 C 盘根目录下建立 MYFOX 子目录,只需:

do makedir whth "c:\myfox"即可,如果要在 MYFOX 目录下建立 MYDBF 子目录,只需:

do makedir whth "c:\myfox\mydbf"即可.

例二:快速保存和恢复屏幕

FoxPro 程序中,使用

RUN<外部可执行文件名>或!<外部可执行文件

名>

可以运行外部文件。但是外部文件执行完返回 FoxPro 程序时,原来的屏幕常常被破坏了,下面的程序可以保存和恢复屏幕,它充分利用了汇编语言速度快的优点。

```

;name      savescr.asm
pushall    macro
push ax
push bx
push cx
push dx
push bp
push si
push di
push ds
push es
pushf
endm
popall     macro
popf

```

```

pop es
pop ds
pop di
pop si
pop bp
pop dx
pop cx
pop bx
pop ax
endm

cseg segment para 'code'
assume cs:cseg,ds:cseg
sapi      proc far
start:
pushall
cmp byte ptr ds:[bx], '1'
jz backup_scr

restore_scr:
mov ax,0b800h
mov es,ax      ;b800为文本显示区段地址
mov di,0
push cs
pop ds
mov si,offset scr_buf
mov cx,2000    ;25行*80列=2000个字符
rep movsw      ;ASCII码和属性各一个字节
jmp exit

backup_scr:
mov ax,0b800h
mov ds,ax
mov si,0
push cs
pop es
mov di,offset scr_buf
mov cx,2000
rep movsw

```

exit:

```

popall
ret
sapi      endp
scr_buf    db 4000 dup(0)
cseg       ends
end start

```

说明:(1)二进制模块产生过程同例一;

(2)FoxPro 调用过程:

```

load savescr
call savescr with "1" &&保存屏幕

.
.
.

call savescr with "0" &&恢复屏幕
release module savescr

```