

**哈尔滨工业大学（深圳）**

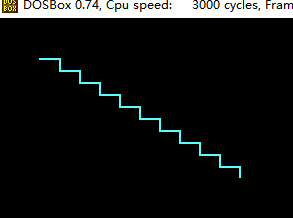
实践教学报告

学 院： 计算机科学与技术学院

|  |  |
| --- | --- |
| 题 目: | 实验四：IO实验 |
| 姓 名: | 李秋阳 |
| 学 号: | 180110527 |
| 专 业: | 计算机科学与技术 |
| 日 期: | 2019年10月31日 |

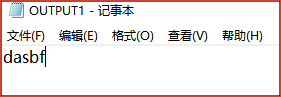
1. **问题描述**

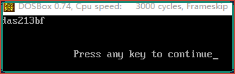
必做题：

题目：通过BIOS和DOS功能调用实现“下楼梯功能”：(提示：用写像素中断画线)从键盘输入一个字符，控制下一个阶梯，一个阶梯由一条水平线和一条垂直线构成，如图最终结果：

通过video中断、像素中断、dos功能等实现动态显示“下楼梯功能”

选做题1：

题目：创建一个文件，从键盘输入任意字母和数字（长度小于等于10），然后删除字符串中的数字，将字母存到文件中。结果如图：



黑色屏幕中输入字符，白色文本中保存了字母

通过汇编语言实现的磁盘文件存取技术对筛选过的信息实现存储。

选做题2：

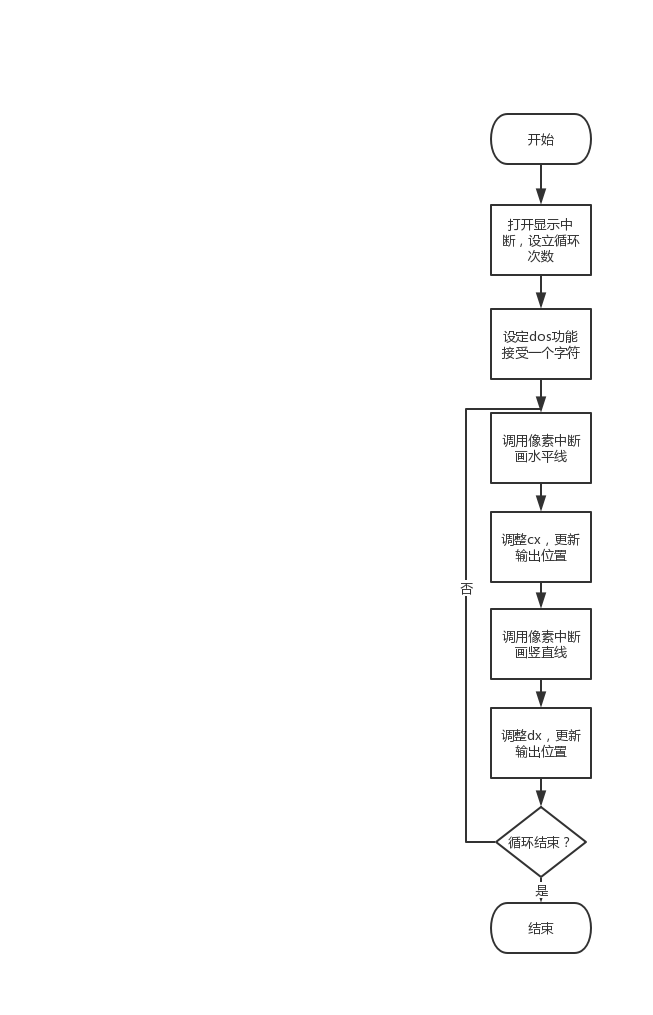
设计一个乐曲程序，通过计算机发出音响并奏出《两只老虎》的乐曲。程序中已经设置了乐谱。结果如图：屏幕显示音符的数字表示，同时扬声器发出对应音符声音



利用计算机控制发声的原理，实现演奏乐曲的程序。

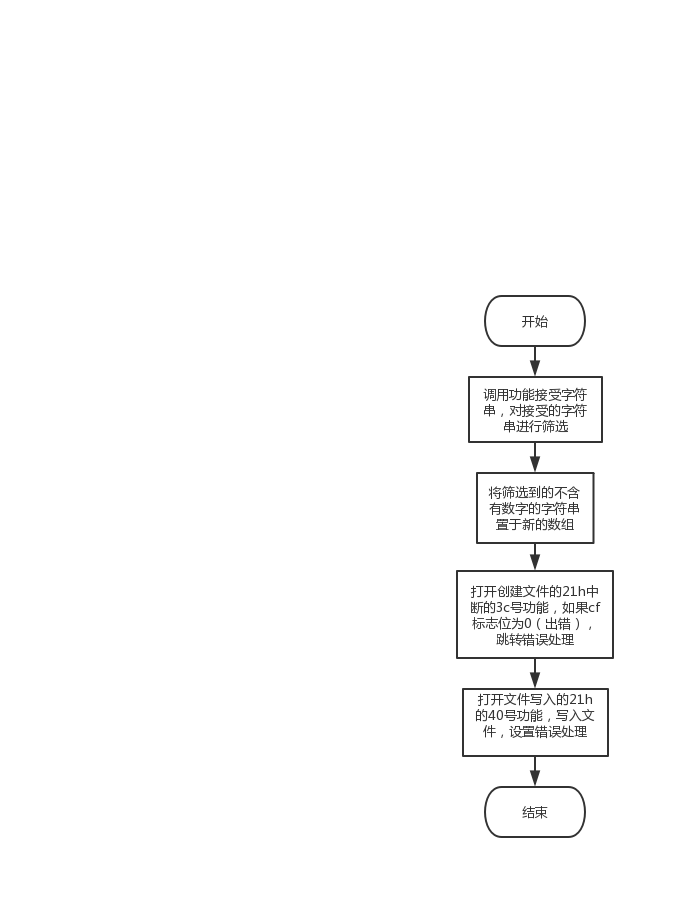
1. **解决方案**

必做题：本题目需要使用的中断是像素中断（在屏幕上画线）和显示中断，通过int 16的0号功能等待接受字符串作为输出的信号，在需要输出的位置用像素中断分别输出一道阶梯的水平线与竖直线。调用循环分别对x轴坐标和y轴坐标进行递增，更新需要输出的位置，实现功能。

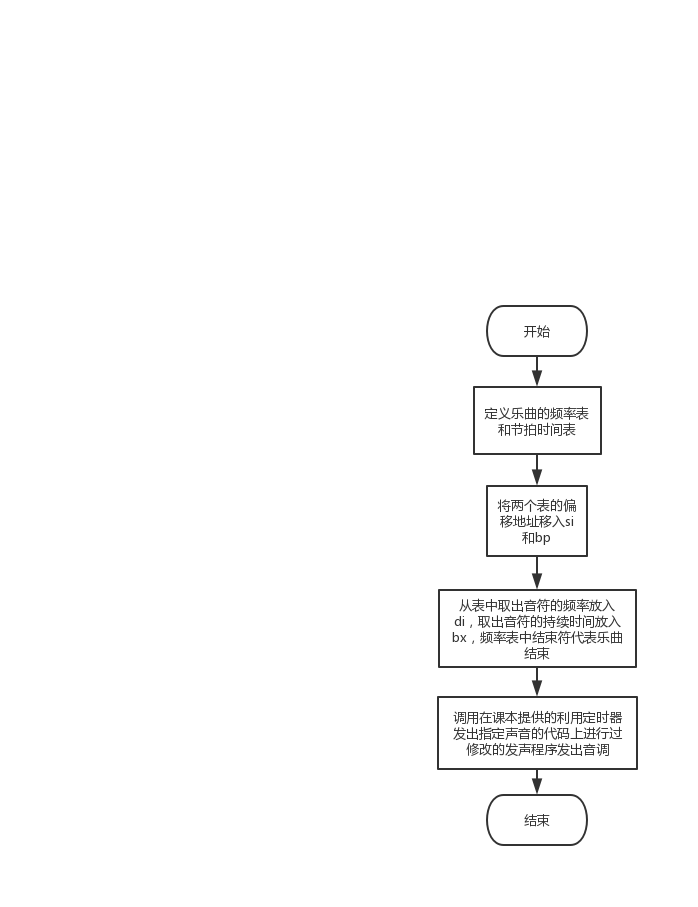


选做题1：文件的创建：通过调用21h中断的3ch号功能实现。在数据段的filename单元中存储文件的asciz串。ah置入功能号3ch，ds:dx指向asciz串，即filename，cx置入文件属性00（一般文件）。调用21h中断后，会改变cf标志位，若创建成功，cf=1，此时ax内存储文件代号，将其放入fh的单元内存储。若失败，则cf=0，可用jc指令跳转至错误处理。

文件的写入：调用21h中断的写文件功能（40h），入口参数：(ah)＝40h，(bx)＝文件代号，文件代号已存储在fh单元中，(cx)＝要写入的字节数，字节数即为字符串长度，已记录在strlen单元中；ds:dx指向str的首地址。若成功，则cf＝0，(ax)＝写入字节数；若失败，则cf＝1，可通过jc指令跳转至错误处理。文件的关闭：写入完成后，必须关闭文件，以确保操作系统将文件记录在磁盘上。其他参数不变，将3eh置入ah，调用21h中断关闭文件。



选做题2：为演奏的乐曲定义一个频率表和一个节拍时间表。分别将两个表的偏移地址移入si和bp，从表中取出音符的频率放入di，取出音符的持续时间放入bx，频率表中结束符代表乐曲结束。调用在课本提供的利用定时器发出指定声音的代码上进行过修改的发声程序发出音调。



1. **具体实现**

必做题

datas segment

w dw 10

h dw 6

turns dw 30

start\_x dw 20

start\_y dw 20

datas ends

stack segment stack

dw 32 dup(?)

stack ends

code segment

assume cs:code,ds:datas,ss:stack

main proc near

start:

mov ax,datas

mov ds,ax

mov ax,stack

mov ss,ax

mov sp,40h

mov ah,0

mov al,4

int 10h

mov cx,turns

loops:

call hor\_ver

mov ah,0

int 16h

loop loops

mov ah,0

mov al,3

int 10h

mov ah,4ch

int 21h

main endp

hor\_ver proc near

push ax

push bx

push cx

push dx

mov cx,w

li:

push cx

mov cx,start\_x

mov dx,start\_y

mov al,1

mov ah,0ch

int 10h

inc start\_x

pop cx

loop li

mov cx,h

ld:

push cx

mov cx,start\_x

mov dx,start\_y

mov al,1

mov ah,0ch

int 10h

inc start\_y

pop cx

loop ld

pop dx

pop cx

pop bx

pop ax

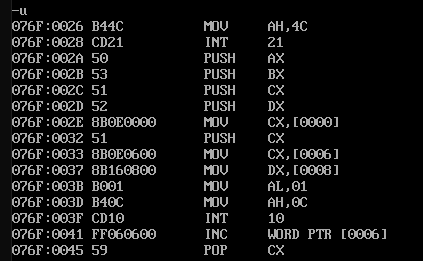
ret

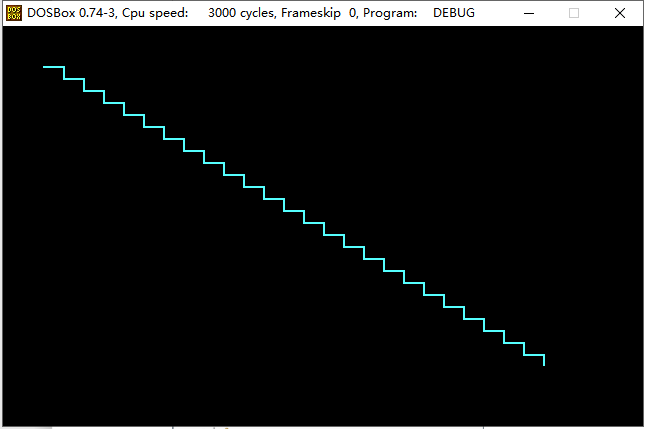
hor\_ver endp

code ends

end start

结果





选做题一

datas segment

buff db 100,0,100 dup(0) ;数据缓存区

outrec db 10 dup(0) ;存放筛选后结果

turns dw 0 ;记录字母个数

file\_name db 'output1.txt',00h ;文件名

create\_err db 0dh,0ah,'error:fail to create file!','$' ;创建文件错误

open\_err db 0dh,0ah,'error:fail to open file!','$' ;打开文件错误

write\_err db 0dh,0ah,'error:fail to write file!','$' ;写文件错误

close\_err db 0dh,0ah,'error:fail to close file!','$' ;关闭文件错误

whandle dw ? ;保存文件句柄

datas ends

stack segment

dw 32 dup(?)

stack ends

codes segment

assume cs:codes,ds:datas,ss:stack

main proc near

start:

mov ax,datas

mov ds,ax

mov ax,stack

mov ss,ax

mov sp,40h

lea dx,buff

mov ah,0ah

int 21h

mov al,buff[1]

mov bx,0

mov bp,0

comp:

cmp bl,al

jz save

cmp buff[bx+2],'a'

jb fai

cmp buff[bx+2],'z'

jna suc

cmp buff[bx+2],'a'

jb fai

cmp buff[bx+2],'z'

jna suc

jmp fai

suc:

inc turns

mov ah,buff[bx+2]

mov outrec[bp],ah

inc bp

fai:

inc bx

jmp comp

save:

mov ah,3ch

mov cx,0

lea dx,file\_name

int 21h

jc error1

mov ah,3dh

mov al,1

lea dx,file\_name

int 21h

jc error2

mov whandle,ax

mov ah,40h

mov bx,whandle

mov cx,turns

lea dx,outrec

int 21h

jc error3

mov ah,3eh

mov bx,whandle

int 21h

jc error4

jmp exit

error1:

lea dx , create\_err

mov ah , 9

int 21h

jmp exit

error2:

lea dx , open\_err

mov ah , 9

int 21h

jmp exit

error3:

lea dx , write\_err

mov ah , 9

int 21h

jmp exit

error4:

lea dx , close\_err

mov ah , 9

int 21h

jmp exit

exit:

mov ah,4ch

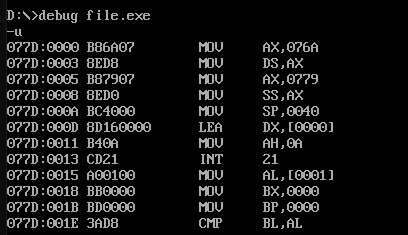
int 21h

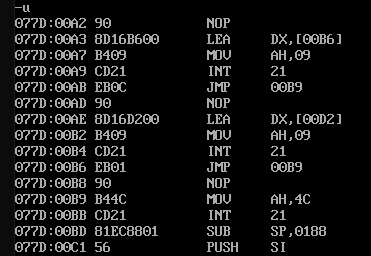
main endp

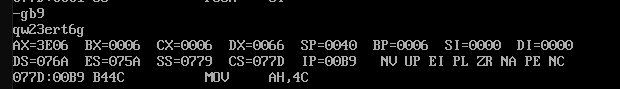
codes ends

end start

结果









选做题2

datas segment

mus\_freq dw 262,294,330,262,262,294,330,262 ;mus\_freq数组存放的是乐谱,每一个数表示一个音符

dw 330,349,392,330,349,392,392,440

dw 392,349,330,262,392,440,392,349

dw 330,262,294,196,262,294,196,262

mus\_time dw 2500,2500,2500,2500,2500,2500,2500,2500,2500,2500 ;mus\_freq中对应音符持续时间

dw 5000,2500,2500,5000,1200,1200,1200,1200,2500,2500

dw 1200,1200,1200,1200,2500,2500,2500,2500,5000,2500,2500,5000

mes db 31h,32h,33h,31h,31h,32h,33h,31h,33h,34h,35h,33h,34h,35h,35h,36h ;乐谱的数字表示

db 35h,34h,33h,31h,35h,36h,35h,34h,33h,31h,32h,35h,31h,32h,35h,31h,'$'

datas ends

stack segment

dw 32 dup(?)

stack ends

codes segment

assume cs:codes,ds:datas,ss:stack

start:

mov ax,datas

mov ds,ax

mov ax,stack

mov ss,ax

mov sp,40h

mov bp,0

mov si,0

in al,61h

and al,0fch

loops:

cmp mes[bp],'$'

jz exit

mov dl,mes[bp]

mov ah,2

int 21h

mov al,0b6h

out 43h,al

mov di,mus\_freq[bp+si]

mov dx,12h

mov ax,348ch

div di

out 42h,al

mov al,ah

out 42h,al

in al,61h

mov ah,al

or al,3

out 61h,al

call del

inc bp

inc si

dec dx

mov al,ah

out 61h,al

jne loops

exit:

mov ah,4ch

int 21h

del proc near

push ax

push bx

push cx

push dx

mov bx,500

wait1:

mov cx,mus\_time[bp+si]

delay:

loop delay

dec bx

jnz wait1

pop dx

pop cx

pop bx

pop ax

ret

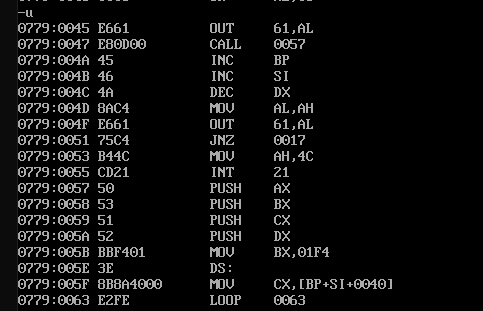
del endp

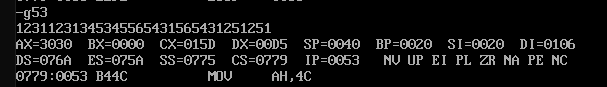
codes ends

end start

结果：







1. **实验总结**

经过四次汇编实验，我开始重新思考汇编语言这一门学科的意义。如何理解一台计算机的运作呢？我们常用的一些编程语言是怎样通过编译器实现的呢？汇编语言实验让我结合理论知识去了解更加低层次的计算机。汇编语言是计算机技术的基础，之所以说汇编重要，其一个重要的原因就是，汇编语言让我更好的理解高级语言。汇编语言对于内存的操作都是基于内存地址的，而高级语言中的指针概念，说白了就是内存的地址。指针的学习和应用中最头疼的就是在指针这个抽象的概念和实际的内存单元之间建立思维映射，而这些恰恰是我们在汇编语言学习中频繁做的一件平常事。另外，对于高级语言中的数据类型、形参实参、函数调用、全局变量、局部变量等概念及操作，我们都可以用汇编语言中的一些操作相关联，把这些抽象的概念和过程，通过汇编语言形成一个具体的映像，深度剖析，这样我们才能真正的学会、学好高级语言。