

**微机实验报告**

班级：计算机科学与技术183班

姓名： 王 杰

学号： 201805120507

教师： 舒洪波

目录

[**微机实验报告** 1](#_Toc62317044)

[1.课设目的 3](#_Toc62317045)

[2.程序总体介绍 3](#_Toc62317046)

[3.运用函数模块、部分代码 3](#_Toc62317047)

[键盘中断 3](#_Toc62317048)

[绘制图像 4](#_Toc62317049)

[颜色 7](#_Toc62317050)

[组合键 8](#_Toc62317051)

[4.流程图 10](#_Toc62317052)

[5.实现界面 11](#_Toc62317053)

[6.程序代码 14](#_Toc62317054)

[7.心得体会 21](#_Toc62317055)

## 课设目的

通过本次课程设计，将课上所学知识转化为实践经验，进一步掌握计算机汇编语言编程技术，实现基本的功能;在编程的过程中，提高自己的动手能力，以及分析和解决问题的能力，为将来能够开发出更完整的系统打好基础。

## 2.程序总体介绍

计一个操作界面，开始时显示“开始”，在点击键盘“K”键时出现一个可爱的小熊猫，点击键盘W、S、A、D键可使小熊猫上下左右移动，点击组合键盘移动，比如shift+s等。

## 3.运用函数模块、部分代码

分别运用本学期所学的**颜色，绘制图像，键盘中断（组合键）**等内容。

### 键盘中断

key\_panda:

push ax

push si

mov dx,0x60 ;将端口0x60地址存放到dx中

in al,dx ;从dx的端口中输入一个8位的值放在al

cmp al,0x2a ;用按下左shit键来作为大小写切换摁键

je Transformation1 ;跳转到flag切换大小写

cmp al,0xaa ;用松开左shit表示转化结束

je Transformation2

cmp al,0x25 ;0x25代表k

jnz xs ;结果不为0(键盘按下K)就跳转 画小熊猫

mov si,0

xclear:

cmp si,320\*200

ja x

mov al,48

mov [es:si],al

inc si

jmp xclear

x:

mov word[J],1

mov word[ds:X],50 ;矩阵起点坐标x

mov word[ds:Y],50 ;矩阵起点坐标y

call panda

jmp x14

### 绘制图像

panda:

push ax ;入栈

push si

push bx

push bp

push cx

mov bp,0

mov cx,0

mov ax,[Y] ;第Y行

mov bx,320

mul bx

add ax,[X] ;第X列，Y\*320+X

mov bx,ax ;bx代表图案的起始位置

x1:

add bx,cx

mov si,0

x0: cmp si,29 ;矩阵总列数,从0列开始，30列

ja x2

mov al,[wd0+bp+si]

mov [es:bx+si],al

inc si

jmp x0

x2:

add bp,30 ;矩阵总列数

mov cx,320

cmp bp,889 ;矩阵总像素点

ja xend

jmp x1

xend:

pop cx

pop bp

pop bx

pop si

pop ax

ret

;开始字体

kaishi:

push ax

push si

push bx

push bp

push cx

mov bp,0

mov cx,0

mov ax,[Y]

mov bx,320

mul bx

add ax,[X]

mov bx,ax

xks1:

add bx,cx

mov si,0

xks2:

cmp si,59 ;矩阵总列数

ja xks3

mov al,[wd1+bp+si]

mov [es:bx+si],al

inc si

jmp xks2

xks3:

add bp,60 ;矩阵总列数

mov cx,320

cmp bp,1199 ;矩阵总像素点

ja xks4

jmp xks1

xks4:

pop cx

pop bp

pop bx

pop si

pop ax

ret

### 颜色

;调整显示颜色

color1:

mov al,48

mov dx,0x3c8

out dx,al

mov al,0 ;r

mov dx,0x3c9

out dx,al

mov al,0 ;b

mov dx,0x3c9

out dx,al

mov al,0 ;g

mov dx,0x3c9

out dx,al

mov al,49

mov dx,0x3c8

out dx,al

mov al,255

mov dx,0x3c9

out dx,al

mov al,255

mov dx,0x3c9

out dx,al

mov al,255

mov dx,0x3c9

out dx,al

ret

### 组合键

Transformation1:

mov bx,1

mov word[ds:flag],bx

jmp x14

Transformation2:

mov bx,0

mov word[ds:flag],bx

jmp x14

;按ctrl+s向下移动

xs:

cmp word[J],0

jz x14

cmp al,0x1f ;s

jnz xw

mov bx,word[ds:flag]

cmp bx,0

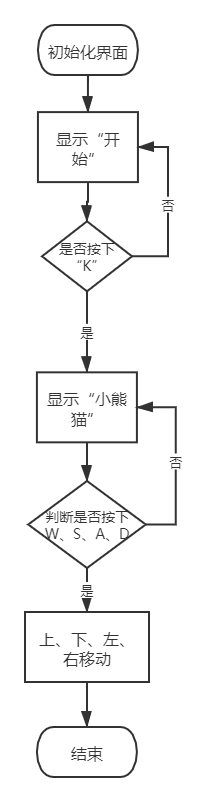
je x14

add word[Y],1

call panda

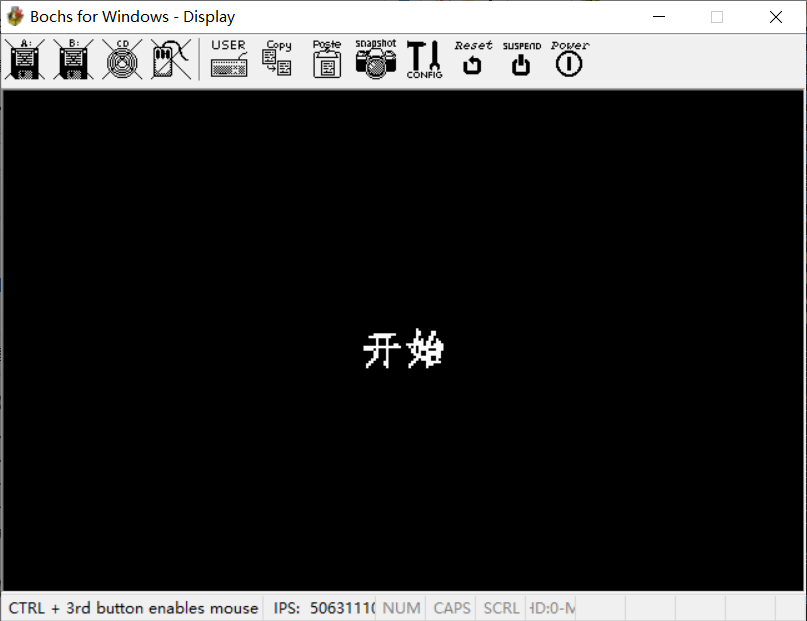
jmp x14

## 4.流程图

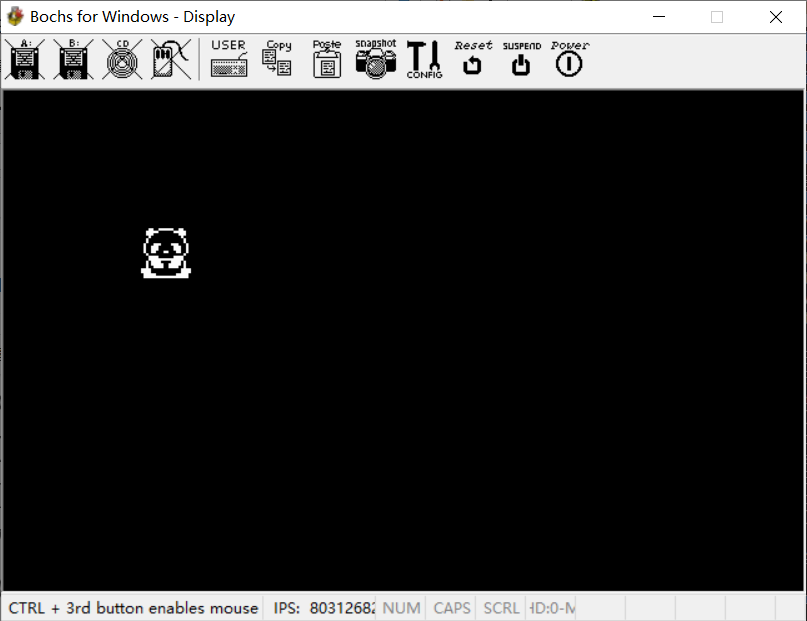
****

## 5.实现界面

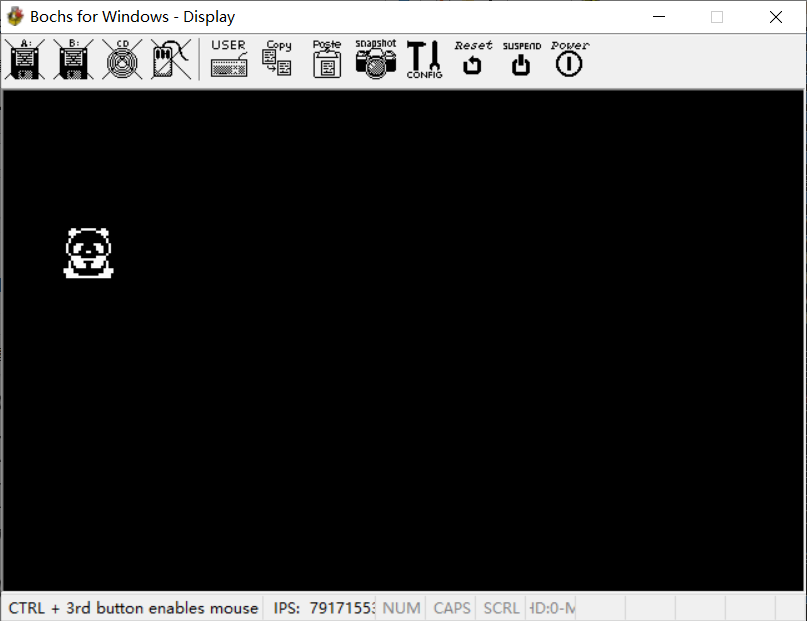
开始界面：



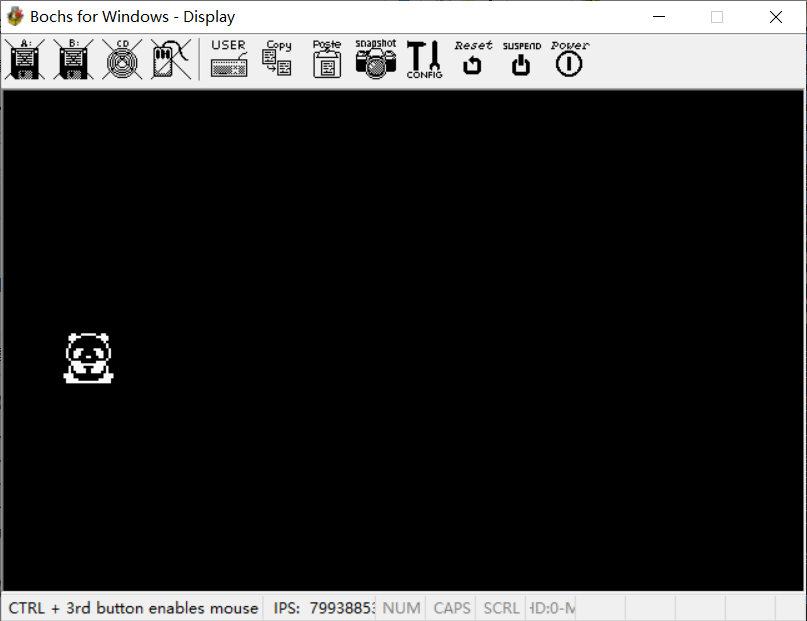
按下“K”后，显示小熊猫：



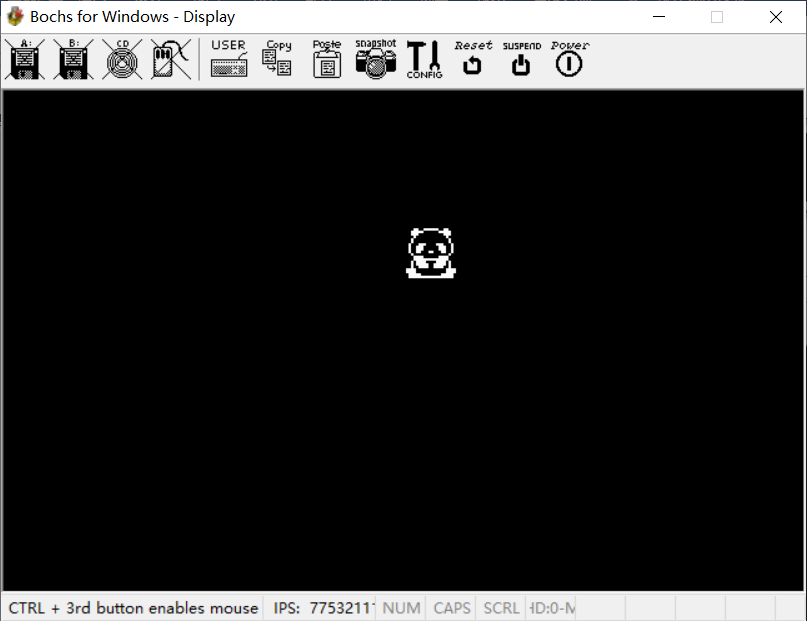
按下“A”后，向左移动：



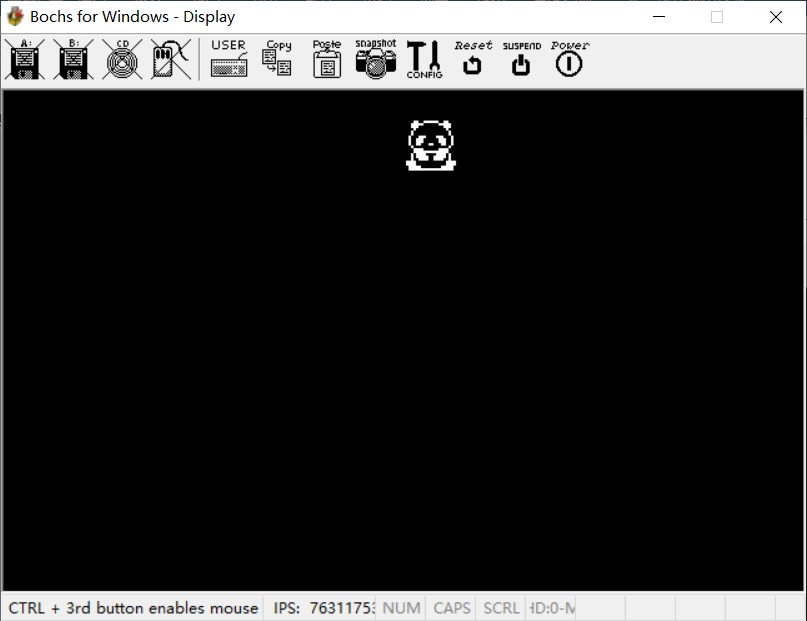
按下“shift+s”后，向下移动：



按下“D”后，向右移动：



按下“W”后，向上移动：



## 6.程序代码

org 0x8400

jmp start

X dw 0

Y dw 0

J dw 0

flag db 0

L dw 0

wd0 db '000000000000000000000000000000' ;30\*30

db '000000000000000000000000000000'

db '000000000000000000000000000000'

db '000000000000000000000000000000'

db '000000000000000000000000000000'

db '000000001100111111001100000000'

db '000000011111000000111110000000'

db '000000011100000000001110000000'

db '000000001000000000000100000000'

db '000000010000000000000010000000'

db '000000010000000000000010000000'

db '000000100011100001110001000000'

db '000000100111000000111001000000'

db '000000100111000000111001000000'

db '000000100110001100011001000000'

db '000000010000000000000010000000'

db '000000001111000000111100000000'

db '000000001111111111111100000000'

db '000000010111100001111010000000'

db '000000010111110011111010000000'

db '000000010011110011110010000000'

db '000001111000000000000111100000'

db '000001111000000000000111100000'

db '000000111110000000011111000000'

db '000000111111111111111111000000'

db '000000000000000000000000000000'

db '000000000000000000000000000000'

db '000000000000000000000000000000'

db '000000000000000000000000000000'

db '000000000000000000000000000000'

wd1 db '000000000000000000000000000000000000000000000000000000000000'

db '000000000000000000000000000000000001000000000000000000000000' ;60\*20

db '000000000000000000000000000000000001100001100000000000000000'

db '000000000000000000111111111000000001100001100000000000000000'

db '000000000000000011111111100000000001000011000000000000000000'

db '000000000000000000010001000000000011010011010000000000000000'

db '000000000000000000010001000000000011110110011000000000000000'

db '000000000000000000010001001100011111110110111100000000000000'

db '000000000000001111111111111110001010111111111100000000000000'

db '000000000000001111010001000000000110111100011100000000000000'

db '000000000000000000110001000000000110100111111000000000000000'

db '000000000000000000110001000000000011100111111000000000000000'

db '000000000000000000110001000000000001111110011000000000000000'

db '000000000000000001100001000000000011011110010000000000000000'

db '000000000000000011000011000000000110000111111000000000000000'

db '000000000000000110000011000000001100000111000000000000000000'

db '000000000000000100000001000000001000000100000000000000000000'

db '000000000000000000000000000000000000000000000000000000000000'

db '000000000000000000000000000000000000000000000000000000000000'

db '000000000000000000000000000000000000000000000000000000000000'

;主函数开始

start:

sti

mov al,0x13

mov ah,0x00

int 0x10

mov ax,0

mov ds, ax

mov bx,0

call color1

mov ax,0x0a000 ;往显存(0xa0000开始)物理地址

mov es,ax ;寄存器寻址

mov cx,0

mov ds,cx ;偏移地址为0的一个字放入寄存器中

mov word[ds:X],130 ;矩阵起点坐标x

mov word[ds:Y],94 ;矩阵起点坐标y

call kaishi

mov word[L],0

mov word[J],0

start1:

mov bx,0

mov word[ds:0x24],key\_panda ;键盘中断服务程序地址

mov word[ds:0x26],0

cmp word[J],0

jz start1 ;结果为零则跳转

jmp $ ;主函数结束

;----------------------------------------------------------

;按k画出小熊猫

key\_panda:

push ax

push si

mov dx,0x60 ;将端口0x60地址存放到dx中

in al,dx ;从dx的端口中输入一个8位的值放在al

cmp al,0x2a ;用按下左shit键来作为大小写切换摁键

je Transformation1 ;跳转到flag切换大小写

cmp al,0xaa ;用松开左shit表示转化结束

je Transformation2

cmp al,0x25 ;0x25代表k

jnz xs ;结果不为0(键盘按下K)就跳转 画小熊猫

mov si,0

xclear:

cmp si,320\*200

ja x

mov al,48

mov [es:si],al

inc si

jmp xclear

x:

mov word[J],1

mov word[ds:X],50 ;矩阵起点坐标x

mov word[ds:Y],50 ;矩阵起点坐标y

call panda

jmp x14

Transformation1:

mov bx,1

mov word[ds:flag],bx

jmp x14

Transformation2:

mov bx,0

mov word[ds:flag],bx

jmp x14

;----------------------------------------------------------

;按s向下移动

xs:

cmp word[J],0

jz x14

cmp al,0x1f ;s

jnz xw

mov bx,word[ds:flag]

cmp bx,0

je x14

add word[Y],1

call panda

jmp x14

;----------------------------------------------------------

;按w向上移动

xw:

cmp al,0x11 ;w

jnz xa

sub word[Y],1 ;第Y-1行，下移

call panda

jmp x14

;----------------------------------------------------------

;按a向左移动

xa:

cmp al,0x1e ;a

jnz xd

sub word[X],1 ;

call panda

jmp x14

;----------------------------------------------------------

;按d向右移动

xd:

cmp al,0x20 ;d

jnz x14

mov si,0

xf: cmp si,320\*200

ja xd1

mov al,48

mov [es:si],al

inc si

jmp xf

xd1:

add word[X],1

call panda

;----------------------------------------------------------

;

x14:

;向端口0x20写入0x61，请求硬盘读

mov dx,0x20

mov al,0x61

out dx,al

pop si

pop ax

iret ;中断返回 中断服务程序的最后一条指令

;----------------------------------------------------------

;小熊猫图案

panda:

push ax ;入栈

push si

push bx

push bp

push cx

mov bp,0

mov cx,0

mov ax,[Y] ;第Y行

mov bx,320

mul bx

add ax,[X] ;第X列，Y\*320+X

mov bx,ax ;bx代表图案的起始位置

x1:

add bx,cx

mov si,0

x0: cmp si,29 ;矩阵总列数,从0列开始，30列

ja x2

mov al,[wd0+bp+si]

mov [es:bx+si],al

inc si

jmp x0

x2:

add bp,30 ;矩阵总列数

mov cx,320

cmp bp,889 ;矩阵总像素点

ja xend

jmp x1

xend:

pop cx

pop bp

pop bx

pop si

pop ax

ret

;----------------------------------------------------------

;开始字体

kaishi:

push ax

push si

push bx

push bp

push cx

mov bp,0

mov cx,0

mov ax,[Y]

mov bx,320

mul bx

add ax,[X]

mov bx,ax

xks1:

add bx,cx

mov si,0

xks2:

cmp si,59 ;矩阵总列数

ja xks3

mov al,[wd1+bp+si]

mov [es:bx+si],al

inc si

jmp xks2

xks3:

add bp,60 ;矩阵总列数

mov cx,320

cmp bp,1199 ;矩阵总像素点

ja xks4

jmp xks1

xks4:

pop cx

pop bp

pop bx

pop si

pop ax

ret

;调整显示颜色

color1:

mov al,48

mov dx,0x3c8

out dx,al

mov al,0 ;r

mov dx,0x3c9

out dx,al

mov al,0 ;b

mov dx,0x3c9

out dx,al

mov al,0 ;g

mov dx,0x3c9

out dx,al

mov al,49

mov dx,0x3c8

out dx,al

mov al,255

mov dx,0x3c9

out dx,al

mov al,255

mov dx,0x3c9

out dx,al

mov al,255

mov dx,0x3c9

out dx,al

ret

;----------------------------------------------------------

end:

mov al,0x20

mov dx,0x20

out dx,al ;EOI=1,中断结束

iret

## 7.心得体会

通过这次的课程设计，我重新回顾了之前学习的颜色，画线画图。键盘中断、组合键等知识，也发现了自己以往学的不够扎实，最后还是实现了想要的功能，但是功能略显单一，希望之后能通过自己的学习继续完善。最后还要感谢舒洪波老师的悉心教导，让我能够在微机这门课中学到更多更深层的知识，谢谢老师！