

**《微型计算机技术及应用》**

**课程设计报告**

专业班级 电子信息工程171班

小组组别 第三组

小组成员 杨琼、徐金辉、李慧敏

2018 年 7 月 5日

1. **功能简介**

设计了一个简单的系统，可以实现图形界面显示（窗口显示）、输入输出显示（文字显示）、多任务（按键中断与定时器中断）等功能。

1. **代码简介**

整体来说，代码分成图形模式、文本模式、按键中断、定时器中断四个部分，图形模式，即首先显示一个窗口；文本模式，即在窗口中显示相应的所设计的文字；按键中断与定时器中断相应的结果根据具体代码而言。

start类似于C的main函数 而Wgm Egm Lgm Cgm ...类似于数组 他们指定了该如何填充显存对应位置的颜色以实现图案效果(如果仔细看Wgm中的1的位置 你会发现像一个W)。在start这个主函数内，执行了一系列的寄存器初始化操作,以方便后续子程序使用;此后start开始执行一个个子程序 Color Color1 实现了颜色码与对应色彩的关联。WGMM等子程序，分别实现了一个个 11列 8行的图案的绘制 其实现思路是将颜色码拷贝到显存对应位置。所有call后面跟的 我们称其为子程序, 调用Call以后 会将当前状态及一些相关数据入栈 并将cs:ip跳到子程序所在位置 开始执行子程序，子程序中有ret 类似于return 会返回调用函数处并将之前保存的状态、数据出栈。因而 call完全可以按照C中函数的思维来理解。

主代码design1：

初始时，在屏幕的中央会出现一个窗口，并且在窗口上方，会有“WELCOME!”的字样。当按下“A”，按键中断1响应，定时器中断0响应，从而进入闭幕模式，即涂色。红色在前，蓝绿色在后，最终显示为红在白中扫射。

副代码design：

初始时，在屏幕的上方，会有“Everyone, welcome to the system!”字样。按下“B”，按键中断1响应，进入睡眠模式，即清平，‘B’与其中断响应次数同时显存，定时器中断0响应，‘Z’与‘H’同时显存。

1. **程序代码**

主代码：

org 0x800

jmp start

num db 0

key\_buf times 100 db 0

xlabel dw 11

rgb db 100, 100, 100

db 100, 0, 100

db 100, 0, 0

db 0, 100, 100

Wgm db 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1

db 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1

db 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1

db 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1

db 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0

db 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0

db 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0

db 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0

Egm db 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1

db 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1

db 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1

db 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Lgm db 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0

db 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0

db 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1

db 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1

db 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Cgm db 0, 0, 0, 1, 0, 1, 1, 1,1, 1, 1

db 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0

db 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0

db 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

db 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0

db 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0

db 0, 0, 0, 1, 0, 1, 1, 1, 1, 1, 1

Ogm db 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0

db 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0

db 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0

db 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1

db 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1

db 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1

db 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0

db 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0

Mgm db 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0

db 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0

db 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0

db 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0

db 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1

db 1, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1

db 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1

db 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1

I1gmdb 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0

db 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0

db 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0

db 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0

db 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0

db 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0

db 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0

db 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0

start:

sti

MOV AL, 0x13 ;VGA320x200x8bit显存

MOV AH, 0x00

INT 0x10

mov ax, 0x0a000 ;往显存(0xa000开始)

mov es, ax

mov ax, 0

mov ds, ax

mov si, 0

mov bp, 400

call Color

call Color1

call Color3

call Color4

call DrawLineOfH

call DrawLineOfH1

call DrawLineOfV

call DrawLineOfV1 ;窗口

call WGMM

call EGMM

call LGMM

call CGMM

call OGMM

call MGMM

call EGMM1

call I1GMM ;欢迎词

mov word [ds : 0x24], int\_key1

mov word [ds : 0x26], 0

jmp $

DrawLineOfH:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;横线

;ax : y1

;bx : x1

;cx : x2

;dx : 320

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

mov ax, 18

mov bx, 38

mov cx, 280

mov dx, 320

sub cx, bx

mul dx

add bx, ax

mov dl, 1

x1: mov [es : bx], dl

inc bx

dec cx

jnz x1

pop dx

pop cx

pop bx

pop ax

ret

DrawLineOfH1:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;横线

;ax : y2

;bx : x1

;cx : x2

;dx : 320

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

mov ax, 180

mov bx, 38

mov cx, 280

mov dx, 320

sub cx, bx

mul dx

add bx, ax

mov dl, 1

x2: mov [es : bx], dl

inc bx

dec cx

jnz x2

pop dx

pop cx

pop bx

pop ax

ret

DrawLineOfV:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;竖线

;ax : y1

;bx : x1

;cx : y2

;dx : 320

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

mov ax, 18

mov bx, 38

mov cx, 180

mov dx, 320

sub cx, ax

mul dx

add bx, ax

mov dl, 1

x3: mov [es : bx], dl

add bx, 320

dec cx

jnz x3

pop dx

pop cx

pop bx

pop ax

ret

DrawLineOfV1:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;竖线

;ax : y1

;bx : x2

;cx : y2

;dx : 320

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

mov ax, 18

mov bx, 280

mov cx, 180

mov dx, 320

sub cx, ax

mul dx

add bx, ax

mov dl, 1

x4: mov [es : bx], dl

add bx, 320

dec cx

jnz x4

pop dx

pop cx

pop bx

pop ax

ret

WGMM:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;图形模式显示

;ax : y1

;bx : x1

;cx : x2-x1+1

;dx : 320

;dl : 颜色号

;bp : ;y2-y1+1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

push bp

push si

mov ax, 21

mov bx, 41

mov dx, 320

mov cx, [ds : xlabel]

mov bp, 8

mul dx

add bx, ax

mov si, Wgm

x22:mov al, [ds : si]

inc si

cmp al, 1

jnz x20

mov dl, 2

x21:mov [es : bx], dl

inc bx

dec cx ;11

jnz x22

add bx, 309 ;bx'=bx-(x2-x1)+320

mov cx, [ds : xlabel]

dec bp

jnz x22

pop si

pop bp

pop dx

pop cx

pop bx

pop ax

ret

x20:mov dl, 1

jmp x21

EGMM:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;图形模式显示

;ax : y1

;bx : x1

;cx : x2-x1+1

;dx : 320

;dl : 颜色号

;bp : y2-y1+1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

push bp

push si

mov ax, 21

mov bx, 53

mov dx, 320

mov cx, [ds : xlabel]

mov bp, 8

mul dx

add bx, ax

mov si, Egm

y2: mov al, [ds : si]

inc si

cmp al, 1

jnz y0

mov dl, 2

y1: mov [es : bx], dl

inc bx

dec cx ;11

jnz y2

add bx, 309 ;bx'=bx-(x2-x1)+320

mov cx, [ds : xlabel]

dec bp

jnz y2

pop si

pop bp

pop dx

pop cx

pop bx

pop ax

ret

y0: mov dl, 1

jmp y1

LGMM:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;图形模式显示

;ax : y1

;bx : x1

;cx : x2-x1+1

;dx : 320

;dl : 颜色号

;bp : y2-y1+1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

push bp

push si

mov ax, 21

mov bx, 65

mov dx, 320

mov cx, [ds : xlabel]

mov bp, 8

mul dx

add bx, ax

mov si, Lgm

z2: mov al, [ds : si]

inc si

cmp al, 1

jnz z0

mov dl, 2

z1: mov [es : bx], dl

inc bx

dec cx ;11

jnz z2

add bx, 309 ;bx'=bx-(x2-x1)+320

mov cx, [ds : xlabel]

dec bp

jnz z2

pop si

pop bp

pop dx

pop cx

pop bx

pop ax

ret

z0: mov dl, 1

jmp z1

CGMM:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;图形模式显示

;ax : y1

;bx : x1

;cx : x2-x1+1

;dx : 320

;dl : 颜色号

;bp : y2-y1+1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

push bp

push si

mov ax, 21

mov bx, 77

mov dx, 320

mov cx, [ds : xlabel]

mov bp, 8

mul dx

add bx, ax

mov si, Cgm

a2: mov al, [ds : si]

inc si

cmp al, 1

jnz a0

mov dl, 2

a1: mov [es : bx], dl

inc bx

dec cx ;11

jnz a2

add bx, 309 ;bx'=bx-(x2-x1)+320

mov cx, [ds : xlabel]

dec bp

jnz a2

pop si

pop bp

pop dx

pop cx

pop bx

pop ax

ret

a0: mov dl, 1

jmp a1

OGMM:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;图形模式显示

;ax : y1

;bx : x1

;cx : x2-x1+1

;dx : 320

;dl : 颜色号

;bp : y2-y1+1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

push bp

push si

mov ax, 21

mov bx, 89

mov dx, 320

mov cx, [ds : xlabel]

mov bp, 8

mul dx

add bx, ax

mov si, Ogm

a5: mov al, [ds : si]

inc si

cmp al, 1

jnz a3

mov dl, 2

a4: mov [es : bx], dl

inc bx

dec cx ;11

jnz a5

add bx, 309 ;bx'=bx-(x2-x1)+320

mov cx, [ds : xlabel]

dec bp

jnz a5

pop si

pop bp

pop dx

pop cx

pop bx

pop ax

ret

a3: mov dl, 1

jmp a1

MGMM:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;图形模式显示

;ax : y1

;bx : x1

;cx : x2-x1+1

;dx : 320

;dl : 颜色号

;bp : y2-y1+1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

push bp

push si

mov ax, 21

mov bx, 101

mov dx, 320

mov cx, [ds : xlabel]

mov bp, 8

mul dx

add bx, ax

mov si, Mgm

a8: mov al, [ds : si]

inc si

cmp al, 1

jnz a6

mov dl, 2

a7: mov [es : bx], dl

inc bx

dec cx ;11

jnz a8

add bx, 309 ;bx'=bx-(x2-x1)+320

mov cx, [ds : xlabel]

dec bp

jnz a8

pop si

pop bp

pop dx

pop cx

pop bx

pop ax

ret

a6: mov dl, 1

jmp a7

EGMM1:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;图形模式显示

;ax : y1

;bx : x1

;cx : x2-x1+1

;dx : 320

;dl : 颜色号

;bp : y2-y1+1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

push bp

push si

mov ax, 21

mov bx, 113

mov dx, 320

mov cx, [ds : xlabel]

mov bp, 8

mul dx

add bx, ax

mov si, Egm

y5: mov al, [ds : si]

inc si

cmp al, 1

jnz y3

mov dl, 2

y4: mov [es : bx], dl

inc bx

dec cx ;11

jnz y5

add bx, 309 ;bx'=bx-(x2-x1)+320

mov cx, [ds : xlabel]

dec bp

jnz y5

pop si

pop bp

pop dx

pop cx

pop bx

pop ax

ret

y3: mov dl, 1

jmp y4

I1GMM:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;图形模式显示

;ax : y1

;bx : x1

;cx : x2-x1+1

;dx : 320

;dl : 颜色号

;bp : y2-y1+1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push cx

push dx

push bp

push si

mov ax, 21

mov bx, 125

mov dx, 320

mov cx, [ds : xlabel]

mov bp, 8

mul dx

add bx, ax

mov si, I1gm

a11:mov al, [ds : si]

inc si

cmp al, 1

jnz a9

mov dl, 2

a10:mov [es : bx], dl

inc bx

dec cx ;11

jnz a11

add bx, 309 ;bx'=bx-(x2-x1)+320

mov cx, [ds : xlabel]

dec bp

jnz a11

pop si

pop bp

pop dx

pop cx

pop bx

pop ax

ret

a9: mov dl, 1

jmp a10

Color:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;白色

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

mov dx, 0x3c8

mov al, 1

out dx, al ;颜色号

mov dx, 0x3c9

mov al, 100

out dx, al ;R/4

mov al, 100

out dx, al ;G/4

mov al, 100

out dx, al ;B/4

pop bx

pop ax

ret

Color1:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;紫红色

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

mov dx, 0x3c8

mov al, 2

out dx, al ;颜色号

mov dx, 0x3c9

mov al, 100

out dx, al ;R/4

mov al, 0

out dx, al ;G/4

mov al, 100

out dx, al ;B/4

pop bx

pop ax

ret

Color3:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;红色

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

mov dx, 0x3c8

mov al, 3

out dx, al ;颜色号

mov dx, 0x3c9

mov al, 100

out dx, al ;R/4

mov al, 0

out dx, al ;G/4

mov al, 0

out dx, al ;B/4

pop bx

pop ax

ret

Color4:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;蓝绿色

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

mov dx, 0x3c8

mov al, 4

out dx, al ;颜色号

mov dx, 0x3c9

mov al, 100

out dx, al ;R/4

mov al, 100

out dx, al ;G/4

mov al, 100

out dx, al ;B/4

pop bx

pop ax

ret

display\_key1:

push ax

push cx

mov al, [ds : key\_buf]

cmp al, 0x9e ;A

jnz key1

mov word [ds : 0x20], int\_time0

mov word [ds : 0x22], 0

key1:pop cx

pop ax

ret

int\_time0:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;定时器中断0

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push dx

mov dl, 3

mov byte [es : bp], dl

inc bp

cmp bp, 64000

je t0

t1: mov al, 0x20

mov dx, 0x20

out dx, al ;EOI=1,中断结束

mov dl, 4

mov byte [es : si], dl

inc si

cmp si, 64000

jae t2

pop dx

pop bx

pop ax

iret

t0: mov bp, 0

jmp t1

t2: mov si, 0

jmp t1

int\_key1:

push ax

push dx

mov dx, 0x60

in al, dx ;采集扫描码

mov [ds : key\_buf], al

call display\_key1

mov al, 0x61

mov dx, 0x20

out dx, al ;键盘中断结束

pop dx

pop ax

iret

副代码：

org 0x8400

jmp start

string db 'Everyone, welcome to the system!'

string1 db 'ZH'

num db 0

key\_buf times 100 db 0

start:

sti ;IF=0 中断响应

mov ax, 0xb800 ;文本模式，往0xb800显存

mov es, ax

mov ax, 0

mov ds, ax ;ds作为数据的段地址

mov bp, 0

mov si, 100

call WTMM ;显存句子

mov word [ds : 0x24], int\_key1 ;按键中断1触发

mov word [ds : 0x26], 0

jmp $

WTMM:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;文本模式，显存“Everyone, welcome to the system!”

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push di

push bx

mov di, 180

mov bx, 0

wtm: mov al, [ds : string+bx]

mov byte [es : di], al

mov byte [es : di+1], 0x0c

inc bx

add di, 2

cmp bx,32

jnz wtm

pop bx

pop di

pop ax

ret

clear:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;清平

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push bx

mov bx, 0

x: mov byte [es : bx], ''

mov byte [es : bx+1], 0x0c

add bx, 2

cmp bx, 4000

jnz x

pop bx

ret

display\_key1:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;按键中断1显示

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push cx

call clear

mov al, [ds : key\_buf]

cmp al, 0xb0 ;B

jnz key1

mov byte [es : 0x100], 'B'

mov byte [es : 0x101], 0x0c

mov cl, [ds : num] ;显示中断次数

add cl, 30h

mov byte [es : 0x110], cl

mov byte [es : 0x111], 0x0c

mov word [ds : 0x20], int\_time0

mov word [ds : 0x22], 0

key1: pop cx

pop ax

ret

int\_time0:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;定时器中断0

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push bx

push dx

mov byte [es : bp], 'Z'

mov byte [es : bp+1], 0x0a

add bp, 2

cmp bp, 4000

jae t0

t1: mov al, 0x20

mov dx, 0x20

out dx, al ;EOI=1,中断结束

mov byte [es : si], 'H'

mov byte [es : si+1], 0x0b

add si, 2

cmp si, 4000

jae t2

pop dx

pop bx

pop ax

iret

t0: mov bp, 0

call clear

jmp t1

t2: mov si, 0

call clear

jmp t1

int\_key1:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;按键中断1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

push ax

push dx

mov dx, 0x60

in al, dx ;采集扫描码

mov [ds : key\_buf], al

inc byte[ds : num] ;input编码的个数

call display\_key1

mov al, 0x61

mov dx, 0x20

out dx, al ;键盘中断结束

pop dx

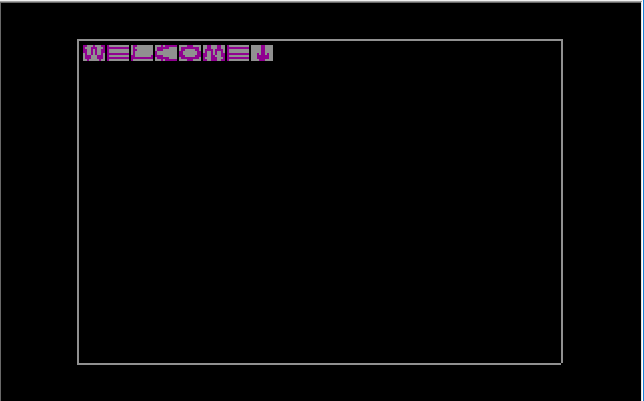
pop ax

iret

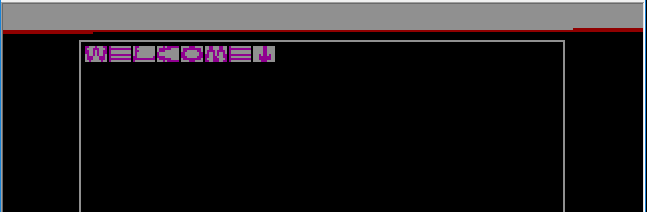
1. **测试结果**

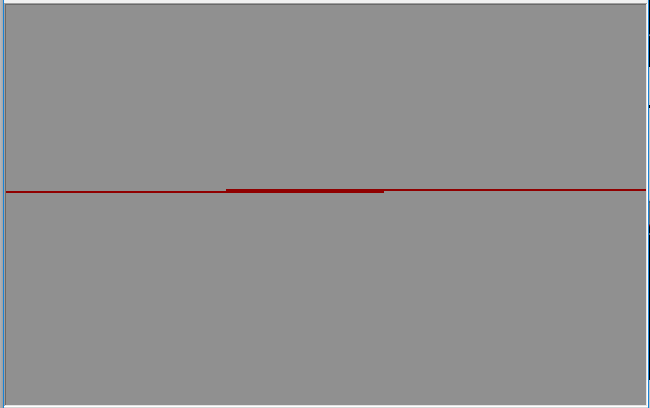
执行design1.bin文件

初始界面



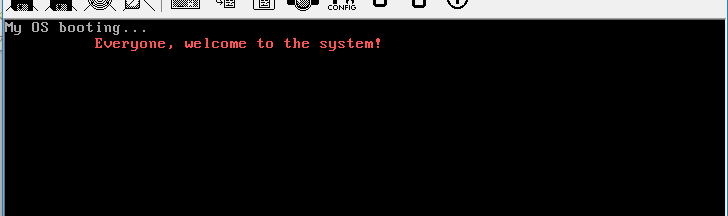
按键中断



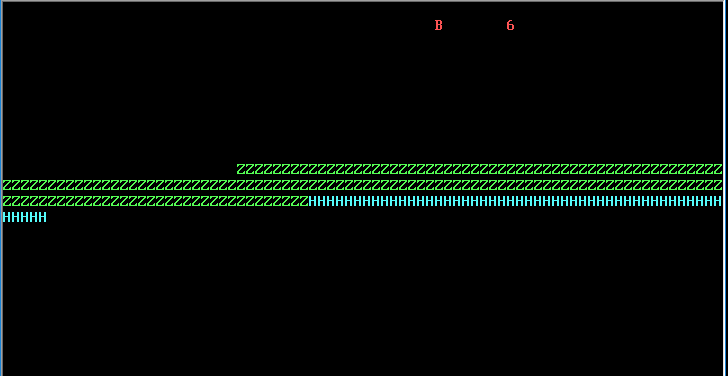


执行design.bin文件

初始界面



按键中断



1. **心得体会**
2. 要有良好的逻辑思考能力。虽然会突然卡顿，一时无法解决，以至于走许多的弯路，浪费了一些时间，但在这个过程中学习到了自己的组合逻辑思考方法。
3. 要有足够的耐心。在编写过程中，总会出现大大小小的问题，使人烦躁，这时候静下心来去认真思考问题的出处以及如何解决才是最重要的。
4. 要有综合的能力。单个的函数模块如果综合起来要怎么实现，这个过程需要良好的综合分析问题的能力。
5. **组员分工**

杨琼：主代码撰写

徐金辉：副代码撰写

李慧敏：文本工作者