# 实验一 ：接管裸机的控制权

目录

一.实验目的 ..... 1
1.掌握虚拟机的使用方法 ..... 1
2.认识电脑与程序连接的过程 ..... 1
3.掌握在裸机上运行程序的方法 ..... 1
二.实验要求 ..... 2
1.搭建虚拟机环境 ..... 2
2.生成虚拟机与制作软盘 ..... 2
3.设计引导扇区程序 ..... 2
三.实验方案： ..... 2
1.虚拟机配置方法： ..... 2
2.主要软件工具及操作流程： ..... 2
3.相关原理： ..... 3
【引导扇区】 ..... 3
【在屏幕上显示文字】 ..... 4
4.程序流程： ..... 6
5.程序关键模块： ..... 7
四.实验过程与结果 ..... 8
【实验过程】 ..... 8
【运行结果】 ..... 11
五.实验总结： ..... 11
参考文献： ..... 12

# 一.实验目的

## 1.掌握虚拟机的使用方法

## 2.认识电脑与程序连接的过程

## 3.掌握在裸机上运行程序的方法

# 二.实验要求

## 1.搭建虚拟机环境

安装虚拟机程序，并搭建一个可以制作含有自己程序的虚拟软盘的环境

## 2.生成虚拟机与制作软盘

生成一个基本配置的虚拟机 XXXPC 和多个 1.44MB 容量的虚拟软盘，将其中一个虚拟软盘用 DOS 格式化为 DOS 引导盘，用 WinHex 工具将其中一个虚拟软盘的首扇区填满你的个人信息

## 3.设计引导扇区程序

设计一个引导扇区程序，功能是用字符‘A’从屏幕左边某行位置 45 度角下斜射出，保持一个可观察的适当速度直线运动，碰到屏幕的边后产生反射，改变方向运动，如此类推，不断运动；在此基础上，增加你的个性扩展，如同时控制两个运动的轨迹，或炫酷动态变色，个性画面，如此等等，自由不限。还要在屏幕某个区域特别的方式显示你的学号姓名等个人信息。将这个程序的机器码放进放进第三张虚拟软盘的首扇区，并用此软盘引导你的 XXXPC，直到成功。

# 三.实验方案：

## 1.虚拟机配置方法：

使用虚拟机软件(此处使用 VMware)创建 PC 虚拟机裸机。

## 2.主要软件工具及操作流程：

工具：虚拟机，编辑器，汇编环境，映像文件制作工具，映像文件读写工具。

1.用notepad++编辑器编写汇编程序

2.通过nasm编译汇编程序生成二进制文件

具体命令：nasm file.asm -o file.bin

3.使用dd命令生成软盘映像文件

具体命令：dd if=/dev/zero of=diska.img bs=512 count=2880

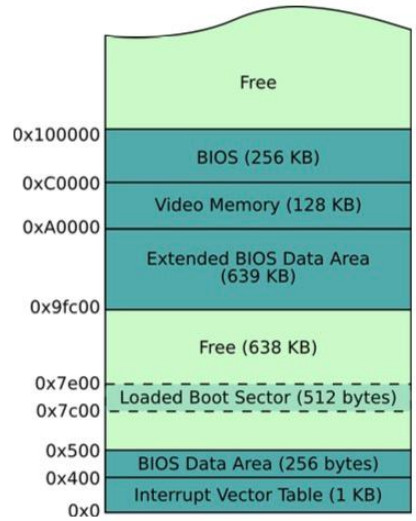
4.使用winhex将二进制文件写入映像文件

5.将映像文件作为软盘加入到vmware所建虚拟机中

# 3.相关原理：

**【引导扇区】**

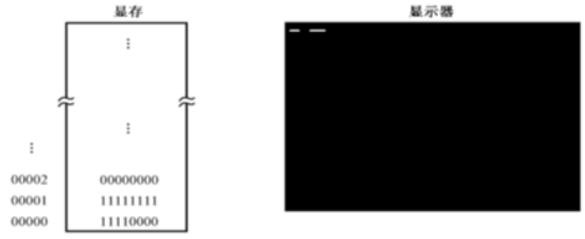
**磁盘的第一个扇区称为引导扇区，其中放置主引导程序，用于加载并转让处理器控制权给操作系统。**系统在开机时会读取磁盘，而且电脑还可以读取虚拟的软盘，可以使用映像文件来作为软盘。系统检测到磁盘或虚拟软盘的存在后，**就会将其首扇区读入到内存中 07c00h 开始的 512 个字节**（如下图一）。**并开始执行主引导程序。**因此我们以无操作系统的虚拟机作为裸机，以映像文件作为软盘来模拟裸机上的程序控制这个过程。



**【在屏幕上显示文字】**

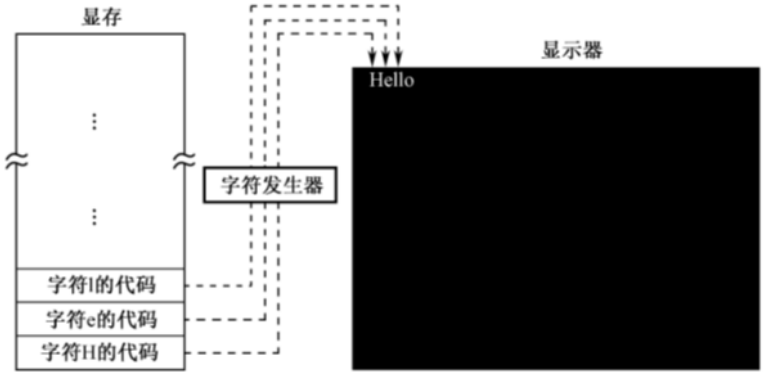
## 在屏幕上显示文字

- 显示器
  - 将那些内容以视觉可见的方式呈现在屏幕上。
- 显示卡
  - 为显示器提供内容，并控制显示器的显示模式和状态
    - 图形方式：最小可控制单位为像素，VGA：640X400
    - 文本方式：最小可控制单位为字符，VGA：25X80
  - 显示卡内存：存放像素或文字及相关属性




## 字符显示原理

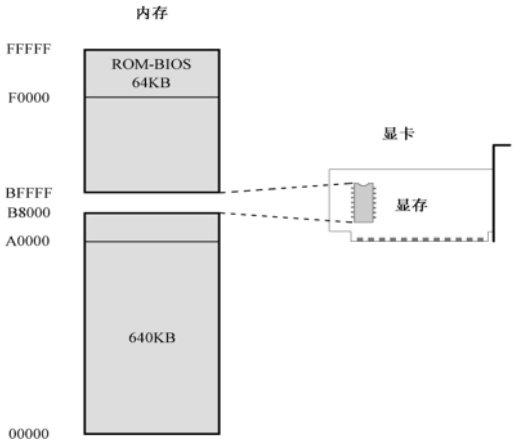
- 字符发生器和控制电路
- 用代码来控制屏幕上的像素，使它们或明或暗以构成字符的轮廓



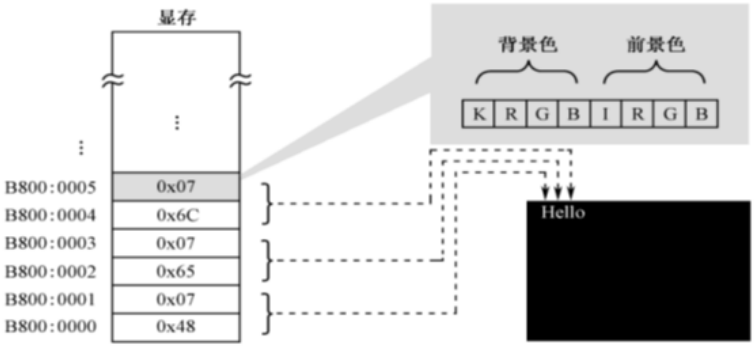
## 字符方式显存地址空间

- 8086可访问的内在1MB
- 地址空间

B8000~BFFFF
共32KB




## 显存与屏幕上字符的对应




## 屏幕上字符的显示属性

- 字符属性0x07
  - 解释为黑底白字，无闪烁，无加亮

| R | G | B | 背景色 | 前景色 | |
|---|---|---|---|---|---|
| | | | K=0 时不闪烁，K=1 时闪烁 | I=0 | I=1 |
| 0 | 0 | 0 | 黑 | 黑 | 灰 |
| 0 | 0 | 1 | 蓝 | 蓝 | 浅蓝 |
| 0 | 1 | 0 | 绿 | 绿 | 浅绿 |
| 0 | 1 | 1 | 青 | 青 | 浅青 |
| 1 | 0 | 0 | 红 | 红 | 浅红 |
| 1 | 0 | 1 | 品（洋）红 | 品（洋）红 | 浅品（洋）红 |
| 1 | 1 | 0 | 棕 | 棕 | 黄 |
| 1 | 1 | 1 | 白 | 白 | 亮白 |

## 显示字符

- 把字符的ASC码和属性编码送到对应的显存中

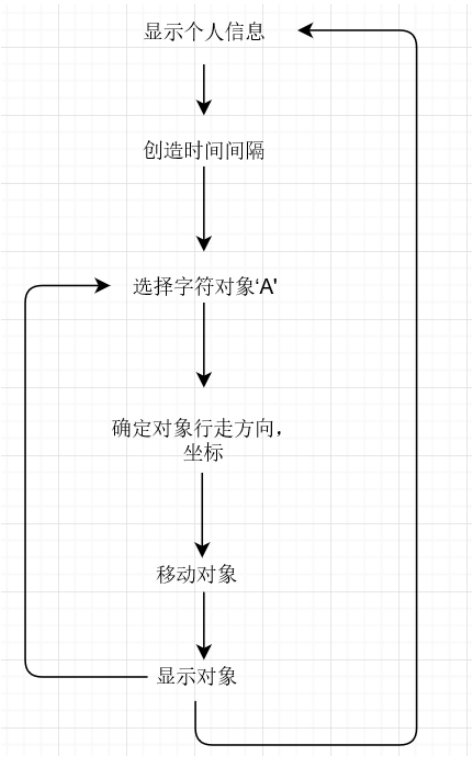
```
mov byte [es:0x00],'L'
mov byte [es:0x01],0x07
```

- 或在屏幕中央显示一个“@”

```
mov byte [es:(12X80+39)X2],'@'；12行39列显示@
mov byte [es: (12X80+39)X2+1],0x07
```

  - 技巧：指令中的常量可于用表达式

## 4.程序流程：




如图所示：程序的流程为：

1，显示个人信息。

2，创造时间间隔，因为电脑运行速度很快所以为了让肉眼看到字符的运动，需要创造时间间隔。

3，选择字符对象，因为有两个字符分别从显示屏左边与右边飞出，所以我们需要设置两个字符对象，此处的用法如同指针/数组的使用，循环的调取两个对象。

4，根据字符对象的坐标，方向对字符对象进行移动，当碰到边界就要让字符对象改变方向，否则沿方向行走。并改变对象的信息

5，进行跳转循环，若第一个对象操作完毕，则跳转到2.重新选择对象。若第二个对象操作完毕则跳转到1.重新更新个人信息的显示，避免被字符遮挡了个人信息。

## 5.程序关键模块：

显示个人信息模块：

```
    mov ax,cs
    mov ds,ax                   ; DS = CS
    mov es,ax                   ; ES = CS
    mov ax,0B800h               ; 文本窗口显存起始地址
    mov gs,ax                   ; GS = B800h
showinf:
    mov si,1980                 ; 偏移
    mov cx,20                   ; string长度
    mov bp,inf
infloop:
    mov bl,byte[es:bp]          ;
    mov byte[gs:si],bl          ;
    mov byte[gs:si+1],11        ;
    inc bp
    add si,2
    loop infloop
```

选择对象模块：

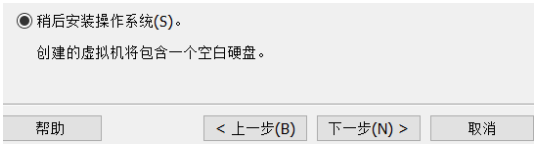
```
    ;循环选择两个对象，将对象的值赋给一个公共对象
    mov cl,0
choose:
    mov al,1
    cmp al,cl
    jz stoneb1
stonea1:
    mov ax,word[xx]
    mov word[x],ax
    mov ax,word[yy]
    mov word[y],ax
    mov al,byte[rdul_]
    mov byte[rdul],al
    jmp dic
stoneb1:
    mov ax,word[xx+2]
    mov word[x],ax
    mov ax,word[yy+2]
    mov word[y],ax
    mov al,byte[rdul_+1]
    mov byte[rdul],al
```

# 四.实验过程与结果

## 【实验过程】

1.虚拟机配置过程

① vmware下创建新的虚拟机，选择典型虚拟机，选择稍后安装操作系统。




② 选择其他操作系统和MS-DOS版本，创建2G硬盘，添加虚拟软盘。


2.Notepad++编写汇编程序

E:\大二下操作系统\操作系统实验\OStoolsDOS\pro1c.asm - Notepad++

文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?

a.asm | b.asm | pro1c.asm

```
; 程序源代码（pro1c.asm）
; 本程序在文本方式显示器上在中间显示个人姓名学号
; 又分别从显示器左边和右边射出一个'A'符号,以45度向右下和左下运动，撞到边框后反射,如此类推.

    Dn_Rt equ 1                 ;D-Down,U-Up,R-right,L-Left
    Up_Rt equ 2                 ; equ 等价语句
    Up_Lt equ 3                 ;
    Dn_Lt equ 4                 ;
    delay equ 50000             ; 计时器延迟计数,用于控制画框的速度
    ddelay equ 580              ; 计时器延迟计数,用于控制画框的速度
    org 07c00h                  ; 程序加载到07c00h
start:
    mov ax,cs
    mov ds,ax                   ; DS = CS
    mov es,ax                   ; ES = CS
    mov ax,0B800h               ; 文本窗口显存起始地址
    mov gs,ax                   ; GS = B800h
showinf:
    mov si,1980                 ; 偏移
    mov cx,20                   ; string长度
    mov bp,inf
infloop:
    mov bl,byte[es:bp]          ;
    mov byte[gs:si],bl          ;
    mov byte[gs:si+1],11        ;
    inc bp
    add si,2
    loop infloop

    ;制造时延
loop1:
    dec word[count]             ; 递减计数变量
    jnz loop1                   ; >0: 跳转;word[count] = 0
    mov word[count],delay
    dec word[dcount]            ; 递减计数变量
      jnz loop1
    mov word[count],delay
    mov word[dcount],ddelay
```

3.用nasm编译汇编程序生成二进制文件

C:\Windows\system32\cmd.exe

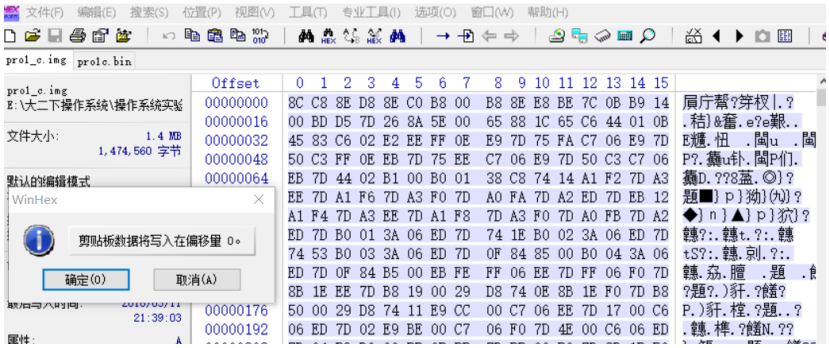
```
E:\大二下操作系统\操作系统实验\OStoolsDOS>nasm pro1c.asm -o pro1c.bin

E:\大二下操作系统\操作系统实验\OStoolsDOS>
```

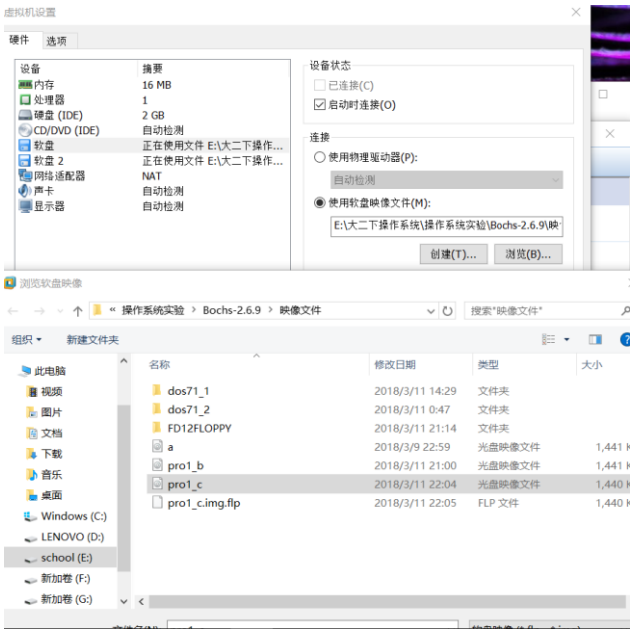
4.使用dd命令制作软盘映像文件

```
E:\GitHub\MyOS\MyOS\Tools>dd if=/dev/zero of=disk.img bs=512 count=2880
rawwrite dd for windows version 0.5.
Written by John Newbigin <jn@it.swin.edu.au>
This program is covered by the GPL.  See copying.txt for details
2880+0 records in
2880+0 records out
```

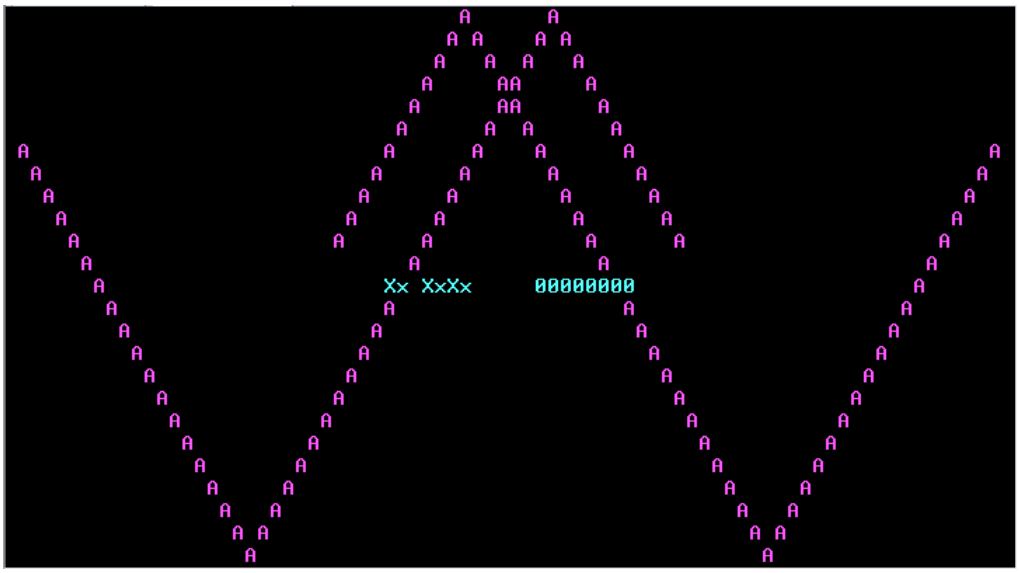
5.用winhex将二进制文件写入映像文件




6.添加软盘驱动器后，将映像文件作为虚拟机软盘



【运行结果】




遇到的问题：符号A将个人信息覆盖

解决方法：只要每次循环更新个人信息的显示便可

# 五.实验总结：

本次实验主要告诉我们载入引导扇区的过程，并且让我们熟悉一下汇编语言。本次实验只使用了引导扇区。因此，本次的实验生成的二进制文件大小不应超过512B，因为引导扇区只是第一个扇区。

在汇编程序上我遇到了一些困难。首先是nasm语法和masm语法的不同，本次实验使用的是nasm：①传送语句中，masm传送的是变量的内容，而nasm传送的是变量的地址。②段值的设置，nasm中不使用segment和assume。然后是x86的语法，我忘记了变量不可以直接传送给变量并且忘记了cs段的作用。

# 参考文献：

1．《80x86 汇编语言程序设计教程》 （杨季文）
2．《nasm 中文手册》