实验二、最小的操作系统

* 1. 实验目的：

*熟悉使用INT 0x10中断的各个功能号来调整屏幕、显示字符和字符串。*

* 1. 实验流程：

1. **上次做的操作系统，屏幕比较混乱:**

>make (在Makefile目录下)

>qemu boot.img

A description...

1. **调整屏幕显示，首先清屏:**

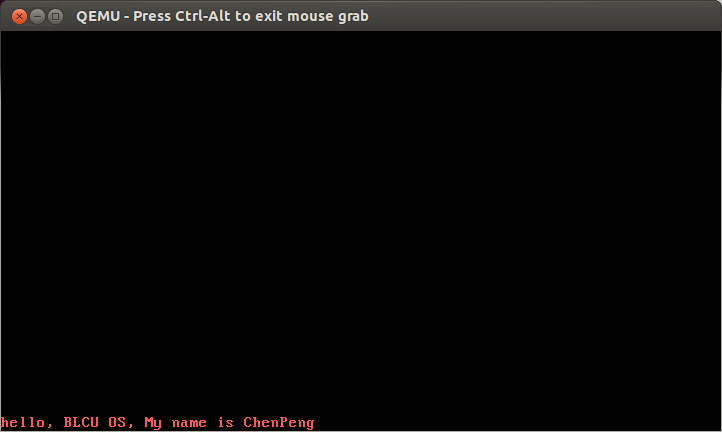
|  |
| --- |
| #清屏函数  #In bootasm.S  ClearScr:  movb $0x06, %ah #功能号06  movb $0, %al #上卷全部行，即清屏  movb $0, %ch #左上角行  movb $0, %cl #左上角列  movb $24, %dh #右下角行  movb $79, %dl #右下角列  movb $0x07, %bh #设置空白区域属性，比如修改上卷后的屏幕与光标颜色  int $0x10  ret |

-----------------------------------------------------------------------------------------------------------

Int 0x10, AH=0x06

INT 10 - VIDEO - SCROLL UP WINDOW  
-----------------------------------------------------------------------------------------------------------

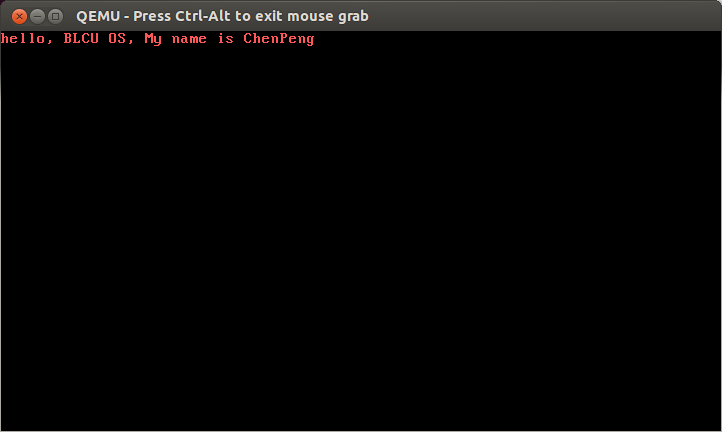
描述：  
上卷窗口  
参数：  
AH 功能号06  
AL 上卷的行数（0表示全部）  
BH 空白区域的视频属性  
CH，CL 窗口左上角的行列位置  
DH，DL 窗口右下角的行列位置  
返回值：  
无  
-----------------------------------------------------------------------------------------------------------



1. **小练习：显示字符串在屏幕开始，如下图：**

提示：设置光标位置为（0,0）,之后再DispStr

http://www.siliconkit.com/dav/work/helphtml/ix/10/06.html



1. **INT 10H 中断简介**

|  |  |  |  |
| --- | --- | --- | --- |
| AH | 功 能 | 调用参数 | 返回参数 / 注释 |
| 1 | 置光标类型 | （CH）0―3 = 光标开始行  （CL）0―3 = 光标结束行 |  |
| 2 | 置光标位置 | BH = 页号  DH = 行  DL = 列 |  |
| 3 | 读光标位置 | BH = 页号 | CH = 光标开始行  CL = 光标结束行  DH = 行  DL = 列 |
| 4 | 置显示页 | AL = 显示页号 |  |
| 5 | 屏幕初始化或上卷 |  | 此功能只能在文字模式下才能发生作用 |
| 6 | 屏幕初始化或上卷 | AL = 上卷行数  　　AL =0全屏幕为空白  　　BH = 卷入行属性 　　CH = 左上角行号  　　CL = 左上角列号  　　DH = 右下角行号  　　DL = 右下角列号 |  |
| 7 | 屏幕初始化或下卷 | AL = 下卷行数 　　AL = 0全屏幕为空白  　　BH = 卷入行属性 　　CH = 左上角行号  　　CL = 左上角列号  　　DH = 右下角行号  　　DL = 右下角列号 |  |
| 8 | 读光标位置的属性和字符 | BH = 显示页 | AH = 属性 　AL = 字符 |
| 9 | 在光标位置显示字符及其属性 | BH = 显示页 　　AL = 字符 　　BL = 属性 　　CX = 字符重复次数 |  |
| A | 在光标位置只显示字符 | BH = 显示页 　　AL = 字符  　　CX = 字符重复次数 |  |
| E | 显示字符(光标前移) | AL = 字符 　　BL = 前景色 | 光标跟随字符移动 |
| 13 | 显示字符串 | ES:BP = 串地址  　　CX = 串长度  　　DH， DL = 起始行列  　　BH = 页号 　　AL = 0，BL = 属性  　　串：Char，char，……，char 　　AL = 1，BL = 属性  　　串：Char，char，……，char  　　AL = 2  　　串：Char，attr，……，char，attr  　　AL = 3  　　串：Char，attr，……，char，attr | 光标返回起始位置 　光标跟随移动 　光标返回起始位置 光标跟随串移动 |

键盘服务--INT 16H

2014-01-21 15:32:30

00H、10H —从键盘读入字符   
03H —设置重复率   
01H、11H —读取键盘状态   
04H —设置键盘点击   
02H, 12H —读取键盘标志   
05H —字符及其扫描码进栈   
  
  
(2)、功能   
  
(3)、   
  
(4)、功能   
出口参数：无   
  
(5)、功能   
出口参数：无   
  
(6)、

1. **INT 16H 中断简介**

|  |  |  |  |
| --- | --- | --- | --- |
| AH | 功 能 | 调用参数 | 返回参数 / 注释 |
| 00H和10H | 从键盘读入字符 | AH＝00H——读键盘  ＝10H——读扩展键盘，可根据0000:0496H单元的内容判断：扩展键盘是否有效 | AH＝键盘的扫描码 AL＝字符的ASCII码 |
| 01H和11H | 读取键盘状态 | AH＝01H——检查普通键盘  ＝11H——检查扩展键盘 | ZF＝1——无字符输入，否则，AH＝键盘的扫描码，AL＝ASCII码。 |
| 02H和12H | 读取键盘标志 | AH＝02H——普通键盘的移位标志  ＝12H——扩展键盘的移位标志 | AL＝键盘标志(02H和12H都有效)，其各位之值为1时的含义如下：  位7—INS开状态  位3—ALT键按下 位6—CAPS LOCK开状态位2—CTRL键按下 位5—NUM LOCK开状态位1—左SHIFT键按下 位4—SCROLL LOCK开状态  位0—右SHIFT键按下 AH＝扩展键盘的标志(12H有效)，其各位之值为1时的含义如下： 位7—SysReq键按下  位3—右ALT键按下 位6—CAPS LOCK键按下位2—右CTRL键按下 位5—NUM LOCK键按下位1—左ALT键按下 位4—SCROLL键按下  位0—左CTRL键按下 |
| 03H | 设置重复率 | AH＝03H 对于PC/AT和PS/2：  AL＝05H BH＝重复延迟 BL＝重复率 对于PCjr：  AL＝00H——装入缺省的速率和延迟  ＝01H——增加初始延迟  ＝02H——重复频率降低一半  ＝03H——增加延迟和降低一半重复频率  ＝04H——关闭键盘重复功能 |  |
| 04H | 设置键盘点击 | AH＝04H  AL＝00H——关闭键盘点击功能  ＝01H——打开键盘点击功能 |  |
| 05H | 字符及其扫描码进栈 | AH＝05H CH＝字符的描述码 CL＝字符的ASCII码 | CF＝1——操作成功，AL＝00H，否则，AL＝01H |