

**实验报告**



**题目： 键盘驱动程序的分析与修改**

**班 级： 2020211302**

**学 号： 2020211262**

**姓 名： 鄭毓恒**

**学 院： 计算机学院**

**2021年 12 月 15 日**

一、实验目的

**1、理解 I/O 系统调用函数和 C 标准 I/O 函数的概念和区别；**

**2、建立内核空间 I/O 软件层次结构概念，即与设备无关的操作系统软件、设备驱动程序和中**

**断服务程序；**

**3、了解 Linux-0.11 字符设备驱动程序及功能，初步理解控制台终端程序的工作原理；**

**4、通过阅读源代码，进一步提高 C 语言和汇编程序的编程技巧以及源代码分析能力；**

**5、锻炼和提高对复杂工程问题进行分析的能力，并根据需求进行设计和实现的能力。**

**报告邮寄（最迟时间：2021年12月22日晚23：59）：**

**大一班（1-4班）：**clavicle@bupt.edu.cn

1. 实验环境

**1、硬件：学生个人电脑（x86-64）**

**2、软件：Windows 10，VMware Workstation 15 Player，32 位 Linux-Ubuntu 16.04.1**

**3、gcc-3.4 编译环境**

**4、GDB 调试工具**

三、实验内容

**解压lab4.tar.gz 文件，解压后进入 lab4 目录得到如下文件和目录：**

**\*\*\*\***

**安装gcc编译器：**

**\*\*\*\***

**实验常用执行命令如下：**

** 执行./run ，可启动 bochs 模拟器，进而加载执行 Linux-0.11 目录下的 Image 文件启动linux-0.11 操作系统**

** 进入 lab4/linux-0.11 目录，执行 make 编译生成 Image 文件，每次重新编译（make）前需先执行 make clean**

** 如果对 linux-0.11 目录下的某些源文件进行了修改，执行./run init 可把修改文件回复初始状态**

**本实验包含 2 关，要求如下：**

** Phase 1**

**键入 F12，激活\*功能，键入学生本人的姓名拼音，首尾字母等显示\***

**比如：zhangsan，显示为：\*ha\*gsa\***

** Phase 2**

**键入“学生本人的学号” ：激活\*功能,键入学生本人的姓名拼音,首尾字母等显示\***

**比如：zhangsan，显示为：\*ha\*gsa\*，**

**键入“学生本人的学号-” ：取消显示\*功能**

**提示：完成本实验需要对 lab4/linux-0.11/kernel/chr\_drv/目录下的 keyboard.s、console.c 和 tty\_io.c 源文件进行分析，理解按下按键到回显到显示频上程序的执行过程，然后对涉及到的数据结构进行分析，完成对前两个源程序的修改。修改方案有两种：**

** 在 C 语言源程序层面进行修改**

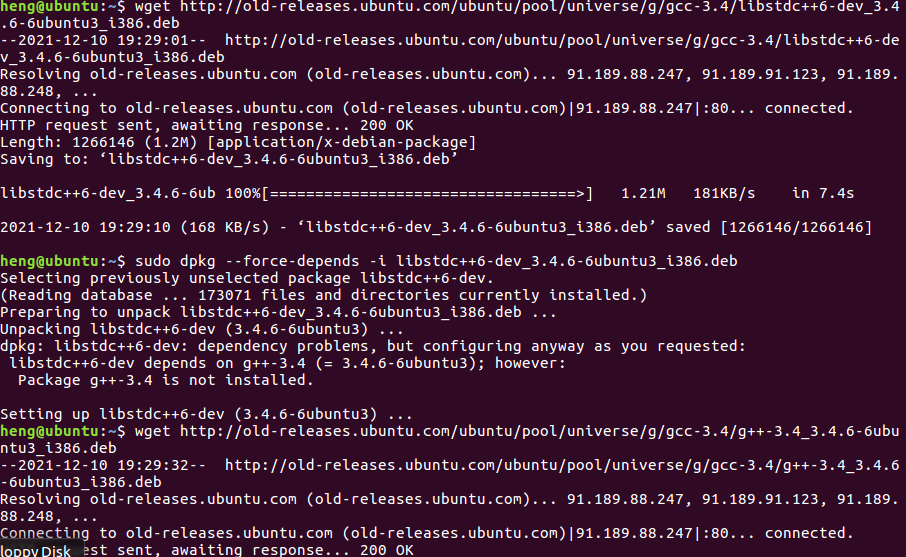
** 在汇编语言源程序层面进行修改**

**其他说明见 实验四.ppt 。linux 内核完全注释(高清版).pdf 一书中对源代码有详细的说明和注释。**

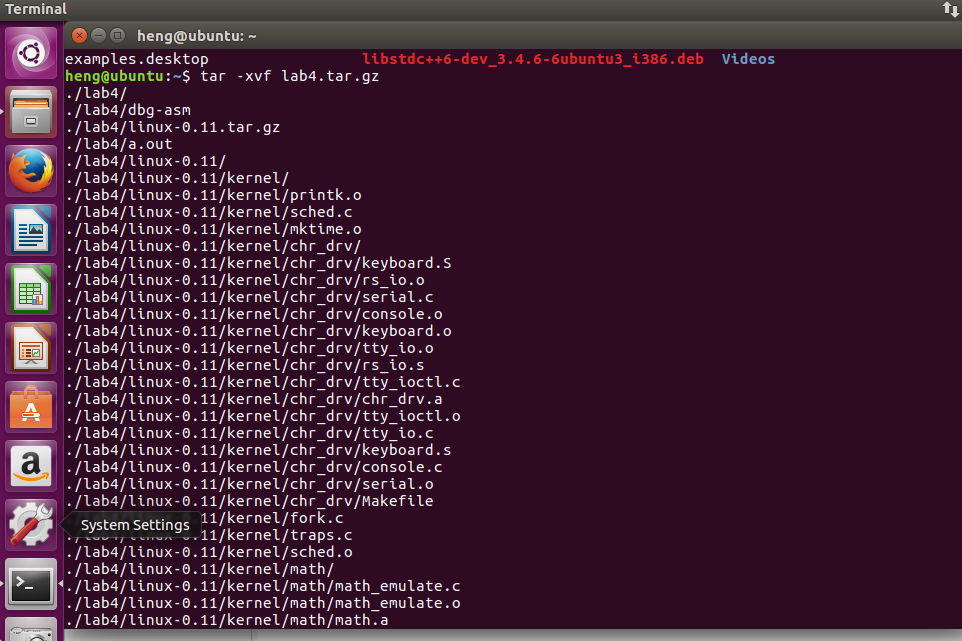
四、实验步骤及实验分析

准备工作

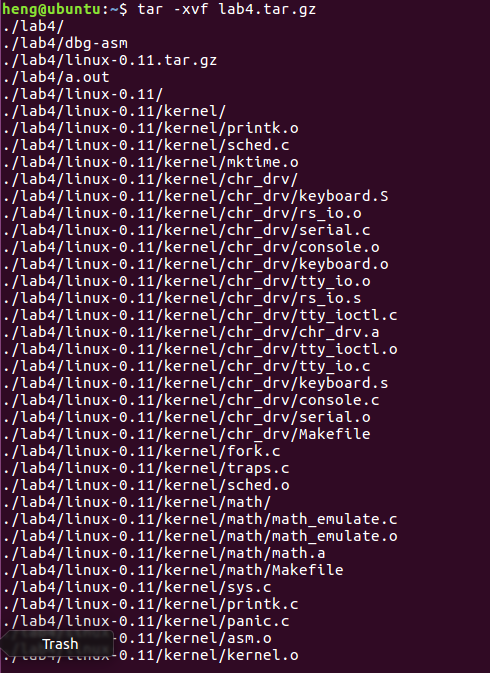
首先安装32位linux-Ubuntu 16.04.1，再安装VMware Workstation 15 Player。创建一个虚拟机，进入虚拟机后根据提示安装VMware Tools for Linux。



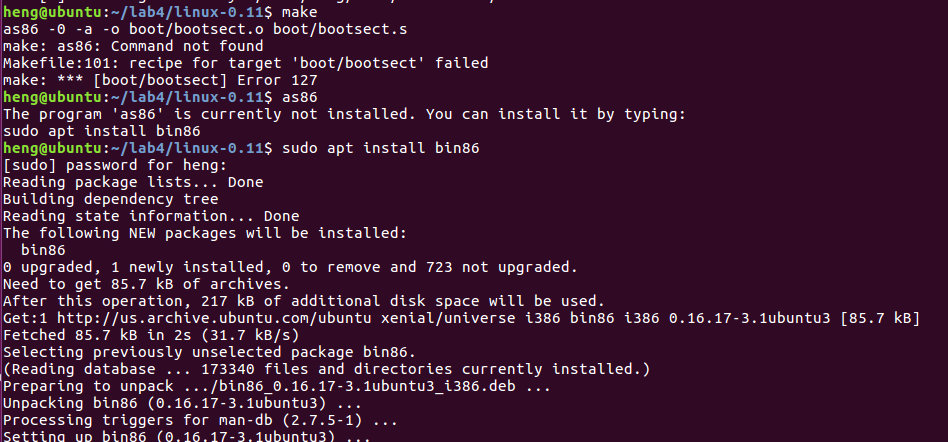
根据实验ppt指示，安装gcc-3.4.



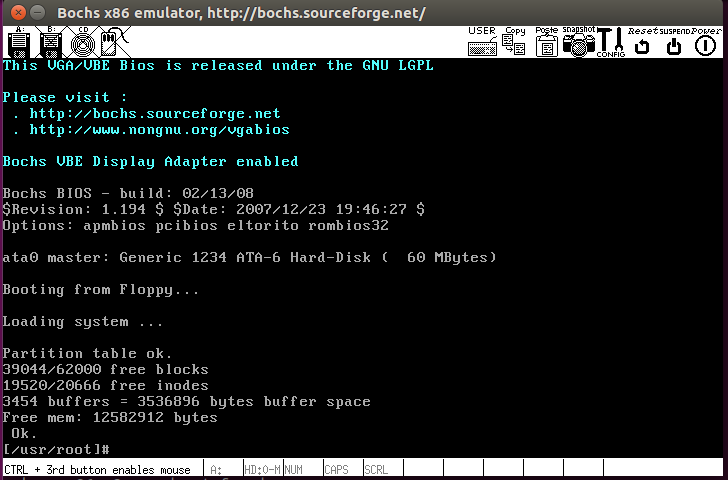
将实验压缩文件夹lab4.tar.gz复制到虚拟机当中，进入终端解压，得到以上文件。



然后对lab4文件夹内的linux-0.11压缩包进行解压，得到以上文件。



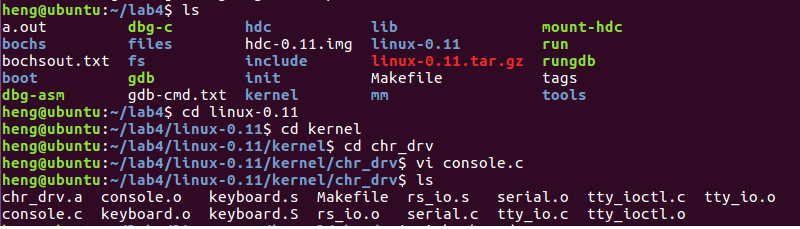
在linux-0.11进行make all会提示错误，没有as86.按照指示安装as86.



在linux-0.11文件夹中执行make clean再make all，然后进入lab4文件夹，进行./run指令，启动bochs模拟器。

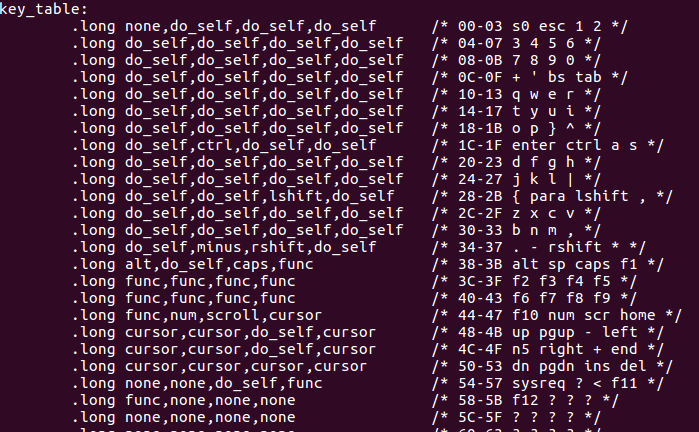
阶段1

将F12看作为激活\*功能的开关，可以透过在驱动程序中加入一个变量实现。变量的值只有0和1两种可能，每按一次F12就将该变量的值改变。取该变量名为f12Flag.

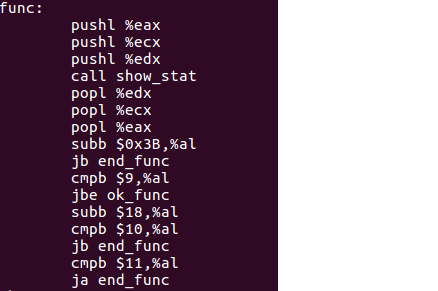


进入/kernel/chr\_drv文件夹中，也就是字符设备驱动程序所在的文件夹，可看到以上文件。

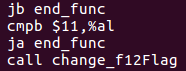
程序console.c为控制台处理程序，keyboard.S为键盘中断处理程序，rs\_io.s为串行中断处理程序，serial.c为串行初始化程序，tty\_io.c为终端IO处理程序，tty\_ioctl.c为终端IO控制程序。



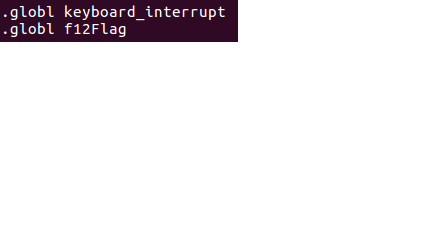
在程序keyboard.S中，可以看出当键盘按键键入时，驱动程序如何实现键盘中断处理过程。在key\_table的部分，可以看出F1至F12的扫描码均用函数func()处理。



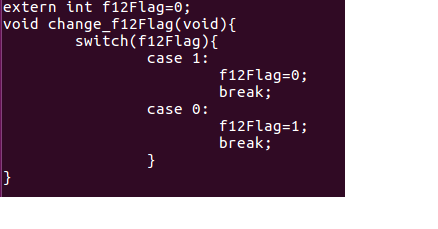
函数func()的内容如上。在key\_table中，F1至F10连续以0x3B至0x44表示，F11和F12以0x57和0x58表示。函数的倒数第四和第二行分别判断是否为F11和F12，如果不是F12则不处理，end\_func。要使F12添加\*功能，则需在函数最后加上一个函数调用语句。



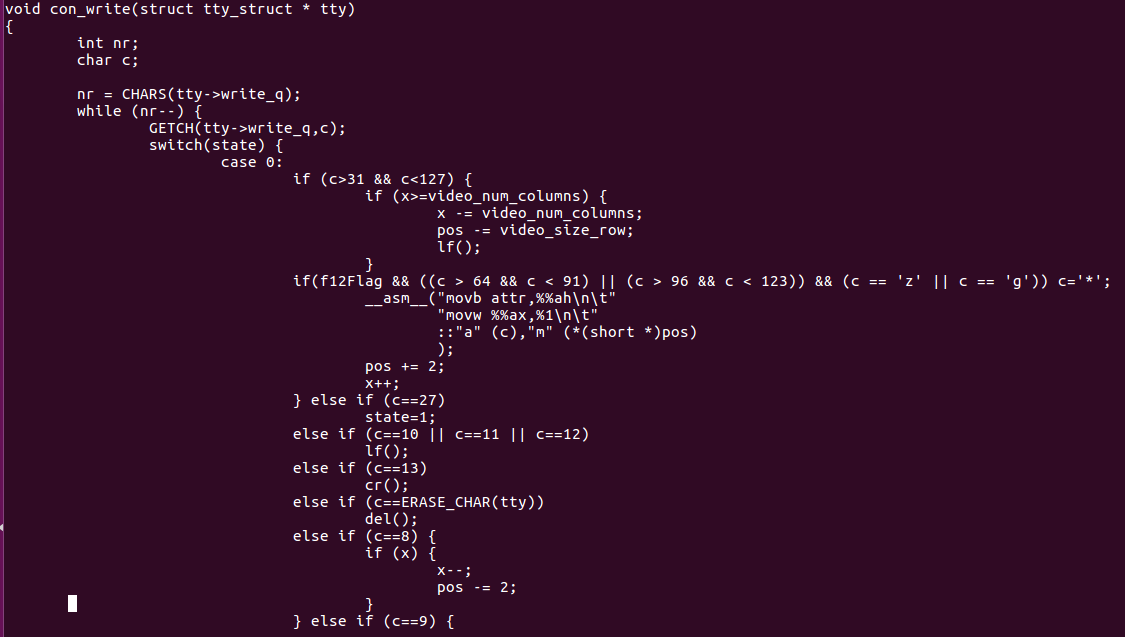
在函数最后添加这一句，当F12键入时，函数change\_f12Flag将会被调用。



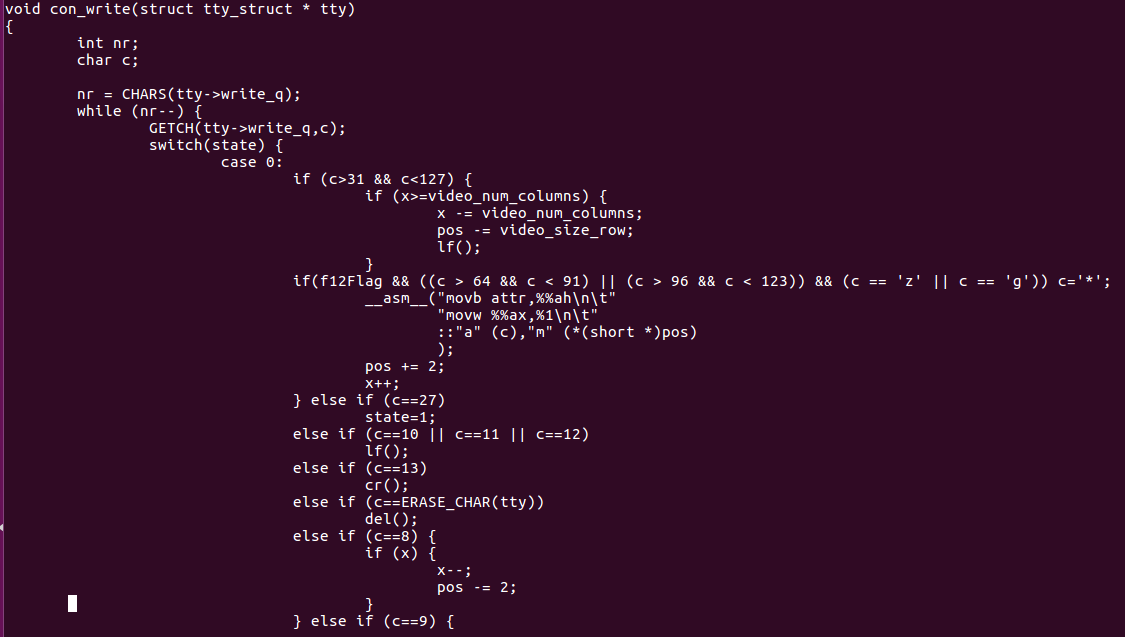
并且在代码开头添加上之前提到的全局变量f12Flag。



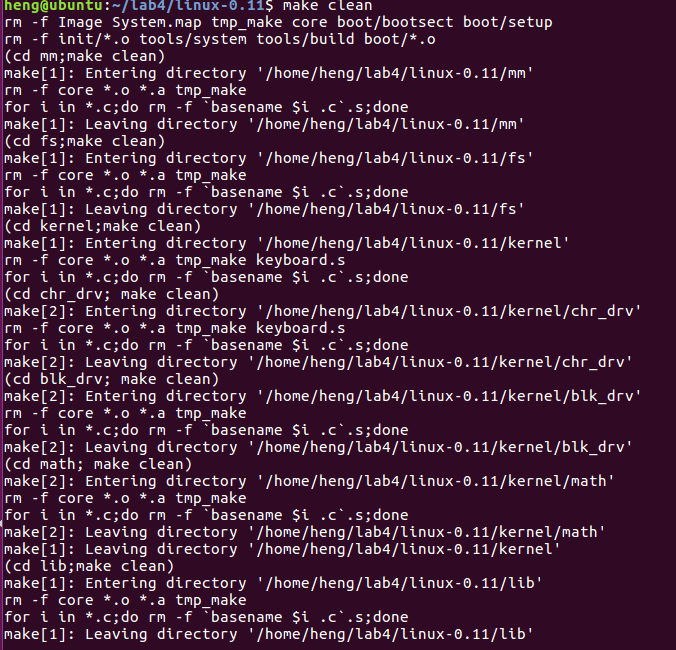
实现\*功能需要改变输入在屏幕的回显，因此进入console.c程序中，加上以上代码。该部分代码表示声明外部变量f12Flag并初始化，然后change\_f12Flag函数则是对该变量进行修改的函数。只有当F12键入时才会调用该函数。

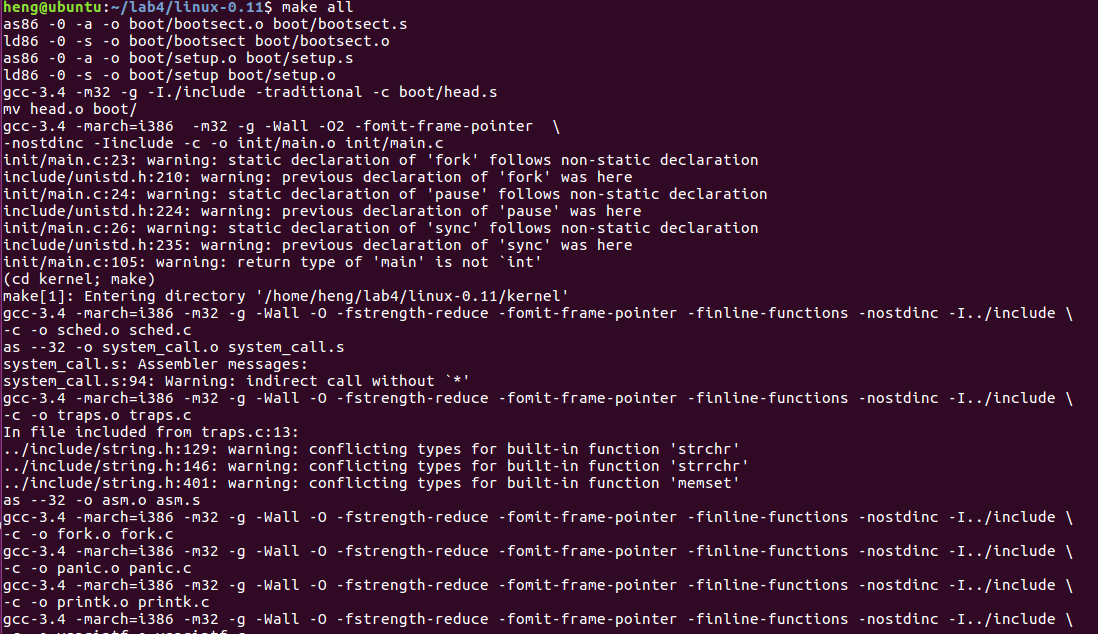


对写控制台函数con\_write进行修改，使回显实现\*功能。该函数会从终端tty\_struct结构的写缓冲队列write\_q中取出字符或字符序列，然后根据字符的性质，把字符显示在终端屏幕上或进行一些光标移动、字符擦除等屏幕控制操作。

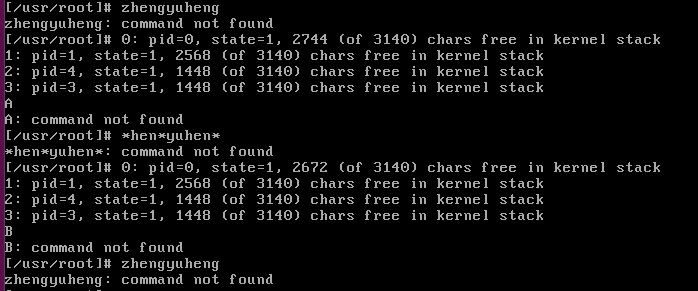


插入以上代码实现。



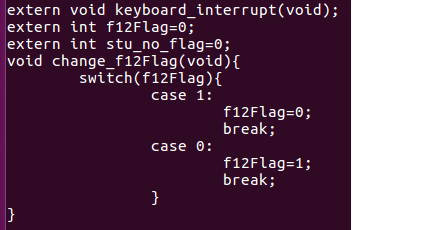


在linux-0.11目录进行make clean和make all。

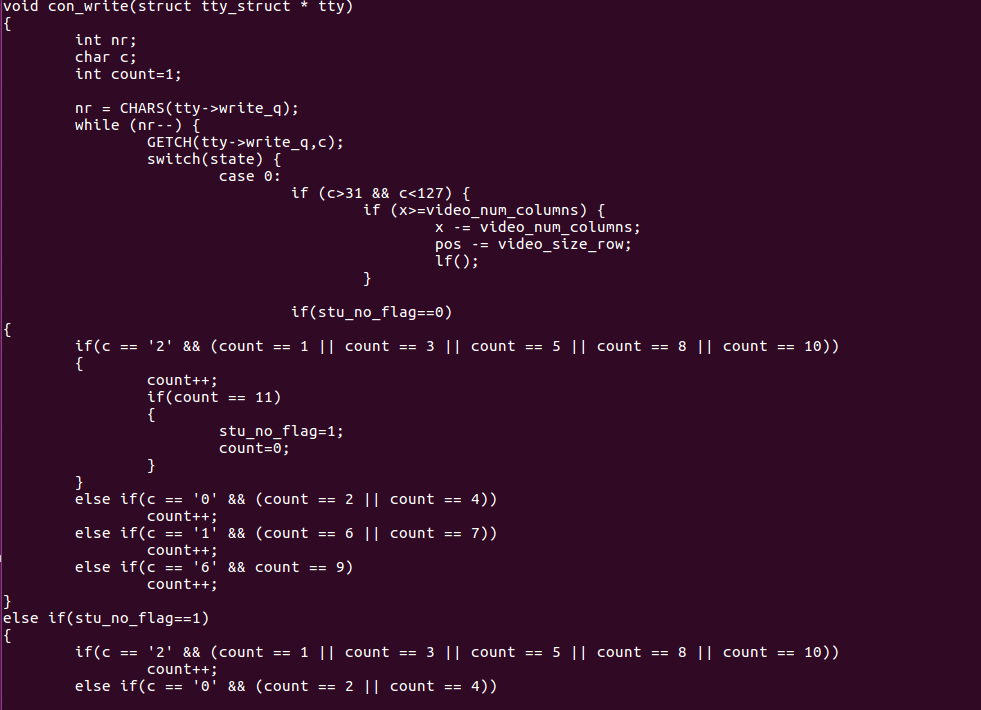


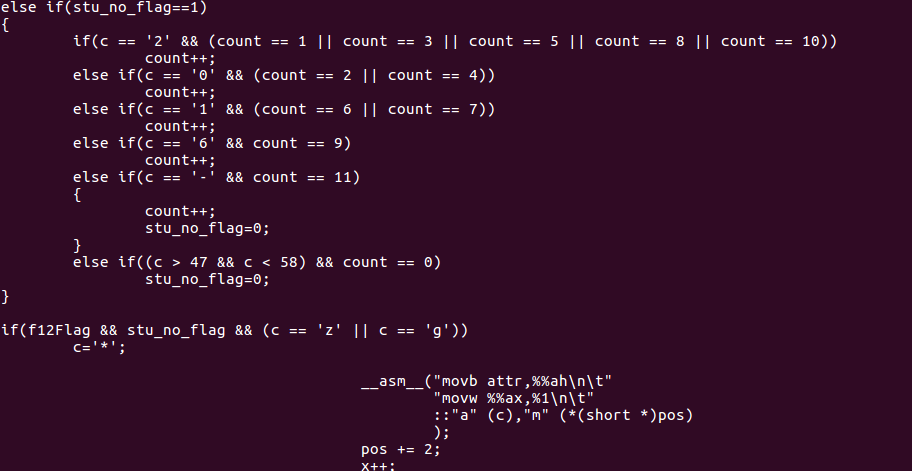
返回lab4文件夹执行./run，启动boch模拟器。先输入一次姓名zhengyuheng，是正常回显。按下F12后，再次输入姓名，可见首字母z和末字母g都变为\*。再次按下F12，输入姓名，回显正常。阶段1完成。

阶段2

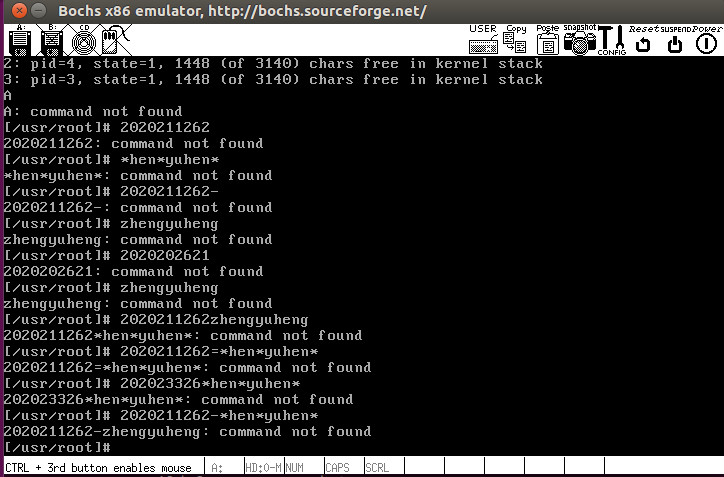


跟阶段1类似，新增一个全局变量，命名为stu\_no\_flag，表示当前是否已经透过学号开启\*功能。由于与键盘中断处理无关，所以只需在console.c添加。





然后更改con\_write函数，插入以上代码。当成功输入学号2020211262，stu\_no\_flag为1。当输入2020211262-，stu\_no\_flag为0.只有当f12Flag和stu\_no\_flag皆为1，也就是键入F12并成功输入学号，才开启\*功能。



在linux-0.11文件夹进行make clean和make all后，返回lab4文件夹进行./run，启动bochs模拟器进行测试。当2020211262（不论单独还是连带姓名）成功输入，姓名首尾字母均被改为\*，否则正常回显。当2020211262-（不论单独还是连带姓名）成功输入，\*功能被关闭，否则仍然启动。阶段2完成。

五、总结体会

有参考书《Linux内核完全注释》和实验ppt的详解，明白各个驱动函数和接口函数的所在文件夹和各自的功能，实验总体难度并不算太大。主要存在两个难点，一是虚拟机内的vi编辑器与过往实验有细微不同，较难使用。第二个难点在于截断2判断是否正确输入学号，代码类似于有限状态自动机，但只能在一个函数内实现。一开始我计划将学号后跟了除完整姓名以外，其他均视为错误输入学号，如2020211262zheng，但在驱动程序内实现难度较大，故简化为输入只要正确输入学号，之前和后续不带其他数字均视为正确，如2020211262zh正确，20202112627错误。

六、诚信声明（不签扣10分）

需要填写如下声明，并在底部给出手写签名的电子版。

在完成本次实验过程中，我参考了以下资料：

1. Linux内核完全注释

在我提交的程序中，还在对应的位置以注释形式记录了具体的参考内容。

我独立完成了本次实验除以上方面之外的所有工作，包括分析、设计、编码、调试与测试。

我清楚地知道，从以上方面获得的信息在一定程度上降低了实验的难度，可能影响起评分。

我从未使用他人代码，不管是原封不动地复制，还是经过某些等价转换。

我未曾也不会向同一课程（包括此后各届）的同学复制或公开我这份程序的代码，我有义务妥善保管好它们。

我编写这个程序无意于破坏或妨碍任何计算机系统的正常运行。

我清楚地知道，以上情况均为本课程纪律所禁止，若违反，对应的实验成绩将按照0分计。

（签名）