

**实验报告**



**题目：键盘驱动程序的分析与修改**

**班级：2021211321**

**学号：2021213586**

**姓名：郭栩源**

**学院：计算机学院**

**2021年12 月8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。**

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

阶段一：

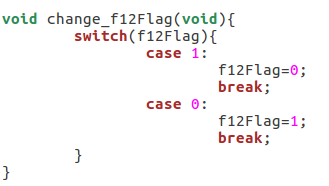
首先，阅读linux内核完全注释，了解linux内核中linux-0.11/kernel/chr\_drv/目录下的keyboard.s,console.c和tty\_io.c的功能。其中keyboard.S是键盘中断处理程序，处理用户键入的字符；console.c是显示控制程序，实现控制台终端收到代码的输出处理。

再分析keyboard.S和console.c的源码。

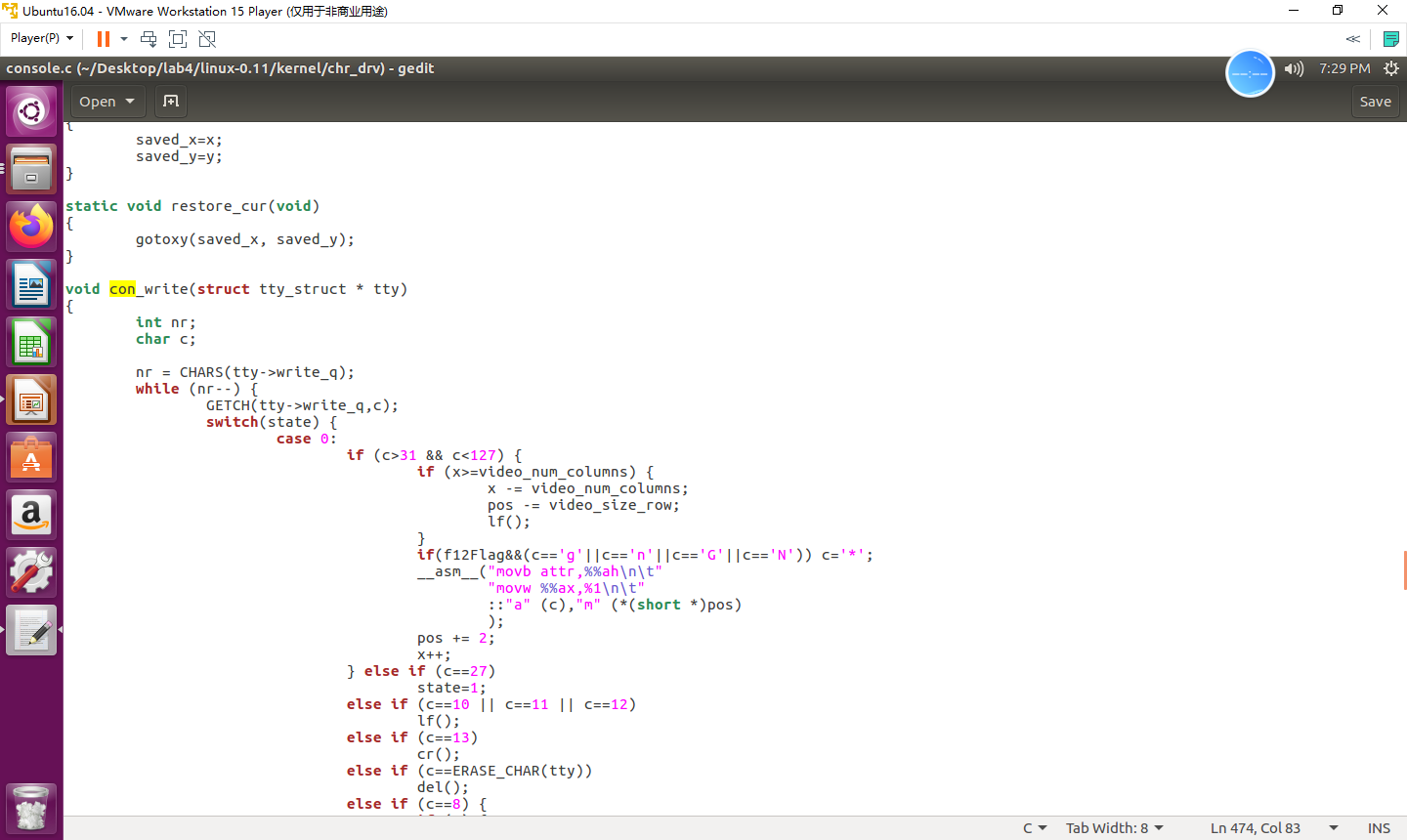
在console.c中，我们首先设立全局变量extern int f12Flag，并赋初值为0。



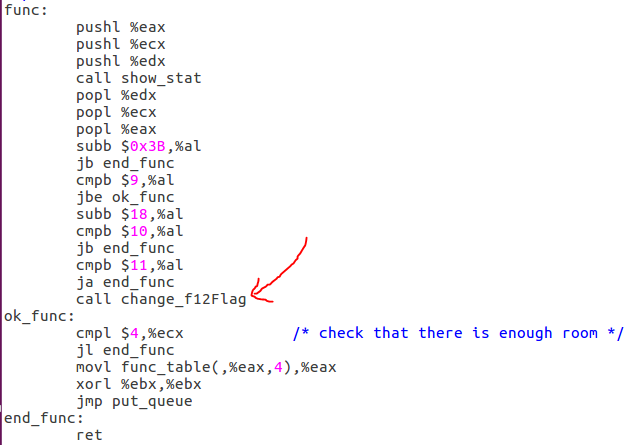
然后编写change\_f12Flag函数，该函数的功能是每次按下f12键时，修改变量f12Flag的值，值为1时修改为0，值为0时修改为1。



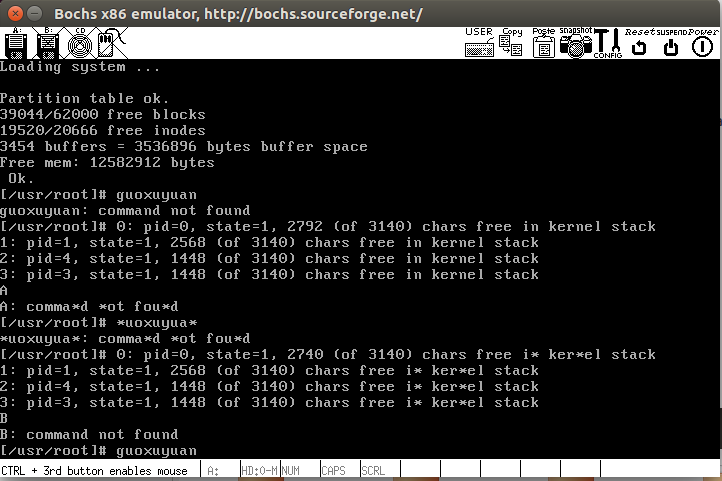
又由linux内核完全注释，了解到con\_write函数为控制台写函数，处理普通字符的回显功能。我的姓名拼音为guoxuyuan，为了让首字母g以及最后一个字母n回显为\*，加入语句“if(f12Flag&&(c=='g'||c=='n'||c=='G'||c=='N')) c='\*';"修改con\_write函数如下：



在keyboard.S中，了解到func函数，用于处理F1~F12功能键，并将功能键扫描码变换成转义字符序列并存放在读队列中。根据linux内核完全注释，我们在下图箭头所指位置插入语句call change\_f12Flag，每次处理f12时进入该函数，改变f12Flag的值。



最后，在linux-0.11目录下使用make指令编译内核，在lab4目录下使用./run指令运行。运行中先输入我的姓名拼音，再按下f12键，重复3次，结果如下：



可知，键盘驱动程序已被成功修改，f12键拥有更改回显为‘\*’的功能。

阶段二：

任务一：

首先对该阶段任务进行分析，我的学号是2021213586，因此阶段二的要求是，输入2021213586时，启动‘\*’回显功能，再次输入2021213586时，取消‘\*’回显功能。故设置一个flag，每次检测到学号输入时，改变flag的值。

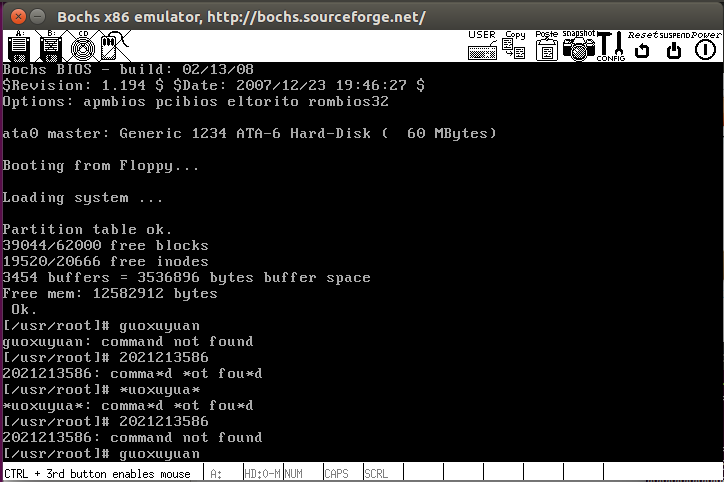
与阶段一相同，我们通过修改console.c中的控制台写函数con\_write，来控制字符的回显功能。

在console.c中建立全局变量flag，其作用与阶段一中的f12Flag相同，flag为1时启用‘\*’回显，flag为0时停用‘\*’回显。建立全局变量total，total是输入的字符串与学号相同的位数，假设输入的前total为与学号的前total位相同，若total+1位输入与学号total+1位相同，则total++，否则total置0。当total等于10时，可知输入了”2021213586“，于是改变flag的值。

修改console.c代码如下图：



最后，在linux-0.11目录下使用make指令编译内核，在lab4目录下使用./run指令运行。首先输入我的姓名拼音”guoxuyuan"，此时为正常回显；输入“2021213586”后，再输入“guoxuyuan”，回显变为“\*uoxuyua\*”；再次输入“2021213586“，回显恢复正常。



可知，键盘驱动程序已被成功修改，输入学号具有打开或关闭 ‘\*’回显的功能。

任务二：

首先对该阶段任务进行分析，我的学号是2021213586，因此阶段二的要求是，输入2021213586时，启动‘\*’回显功能，输入2021213586-时，取消‘\*’回显功能。

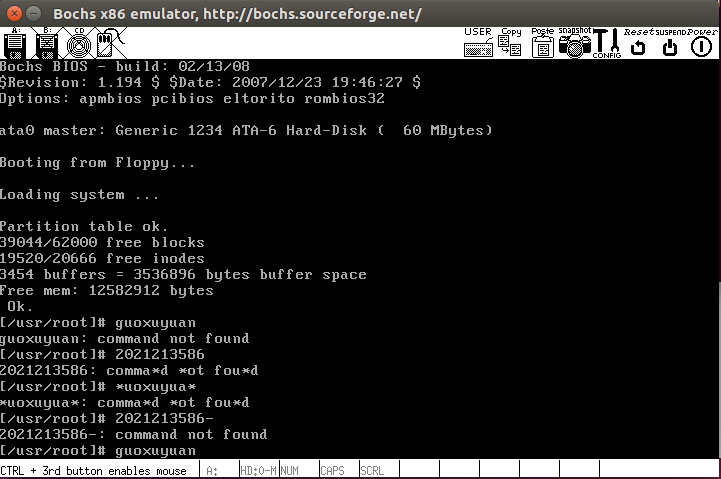
与阶段一相同，我们通过修改console.c中的控制台写函数con\_write，来控制字符的回显功能。

在console.c中建立全局变量flag，其作用与阶段一中的f12Flag相同，flag为1时启用‘\*’回显，flag为0时停用‘\*’回显。建立全局变量total，total是输入的字符串与学号相同的位数，假设输入的前total为与学号的前total位相同，若total+1位输入与学号total+1位相同，则total++，否则total置0。当total等于10时，可知输入了”2021213586“，于是让flag置1，启用‘\*’回显；当total为11时，可知输入了”2021213586-“，于是让flag置0，停用‘\*’回显。

修改console.c代码如下图：



最后，在linux-0.11目录下使用make指令编译内核，在lab4目录下使用./run指令运行。首先输入我的姓名拼音”guoxuyuan"，此时为正常回显；输入“2021213586”后，再输入“guoxuyuan”，回显变为“\*uoxuyua\*”；再输入“2021213586-”，回显恢复正常。



可知，键盘驱动程序已被成功修改，输入学号（学号-）具有打开（关闭）‘\*’回显的功能。

五、总结体会

本次实验与前三次实验不同，它不是在MobaXterm\_Personal\_22.1平台中完成的，而是在桌面级操作系统下运行linux内核完成的。因此还涉及了安装ubuntu桌面级操作系统的步骤。我在本次实验中选择使用虚拟机。

本次实验加深了我对linux中io的了解，尤其是结合linux内核完全注释了解了键盘驱动程序的原理，实现了通过修改linux内核中的键盘驱动程序改变linux控制台中的回显。

总而言之，如果说前三次实验加深了我对linux命令行和汇编语言的理解，本次实验则更多地加深了我对linux操作系统以及linux下io的理解，这些都是非常宝贵的经验。

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，我曾分别与以下各位同学就以下方面做过交流：

1、简单描述交流内容，例如：来自\*\*\*的建议，采用\*\*\*方式\*\*\*

2、

此外，我还参考了以下资料：

1. 网址等
2. 《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）