**设计和功能说明**

**设计**

在控制台之间实现进程通信，模拟广播提示以及聊天信息数据传送。

**功能运行**

　　项目运行之后通过printf不断显示GET\_TICKS消息类型的ticks值。按F1-F3切换控制台，通过进程通信，在循环打印值之后，控制台一能显示二中的内容，二能显示三中的内容。二中进程B和C不断切换，实现B、C重复交替出现的效果。三输入数据之后能在一、二中同步显示。

**程序运行步骤**

　　软盘映像中的 boot 引导loader

　　loader 实现跳入保护模式，通过段、页式存储机制，GDT，LDT数据结构扩充寻址，保护模式可实现中断和异常处理，采用FAT12文件系统在软盘镜像中引导内核进入内存

　　内核通过c 语言编译链接扩充，内核实现操作系统很多主要功能。



**具体实现**

进程切换

　　设置进程表、GDT、esp、TSS、内核栈，可通过添加进程。

键盘中断

　　键入一个值发生一次中断，中断通过设置8259A和建立IDT实现，数组表示扫描码keymap.h，通过函数解析之后显示keyboard.c,通过判断为一些按键加入相应功能

tty控制台

　　通过控制显示属性单独显示，设置３个控制台，in\_process判断不同的按键显示，其中还加入上下滚屏，

框架task\_tty()中不断刷新立即显示得到的字符

进程通信IPC

　　通过核心函数msg\_send()和msg\_receive()发送和处理消息，系统进程SYSTASK接收用户进程消息。

**总结**

　　收获有：实践了操作系统在计算机中具体运行过程的认识。结合了操作系统软件和硬件的了解，对软件硬件的运作方式有了深刻的体会。在键盘中断，tty控制台原理方面体会颇深。

　　这次的项目还是花了很多精力和时间，原本一开始要做的是重写文件系统，后来发现比较困难，当初页想过在内核的基础上编写Ｃ语言程序，但是那样的话和操作系统知识的联系程度并不大，不能通过这门课学习到相应的知识。又转入进程通信，遗憾的是没有时间将功能应用。不过我们学到了很多知识，也拓宽了在操作系统，汇编语言等方面的视野，获得了极大的兴趣和探索的欲望。