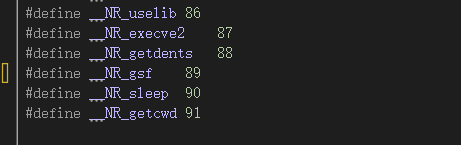
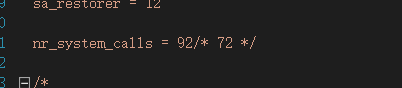
一、在内核添加新的系统调用声明

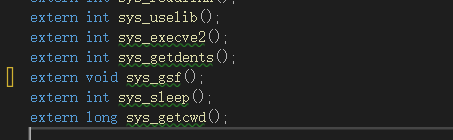
1.在unistd.h中添加系统调用号

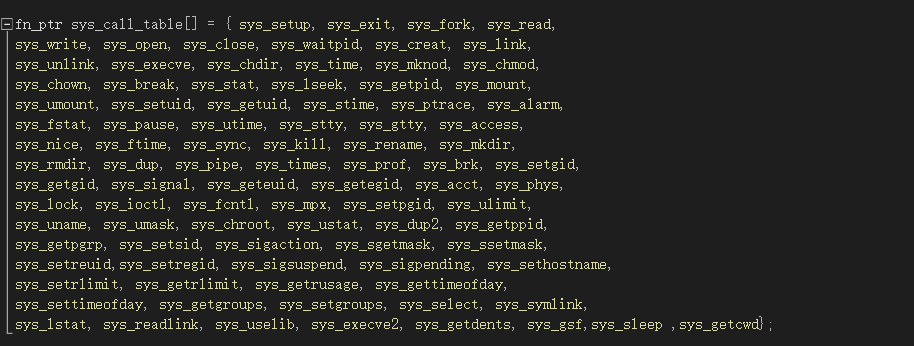


2.修改systemcall.s中的系统调用总数



3.在sys.c中增加函数声明、修改系统调用表



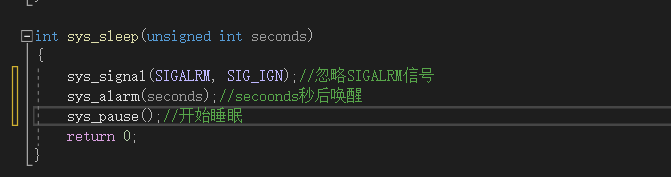


4.直接在main.c中实现4个函数（一开始选择新建一个c文件还需要更改makefile）

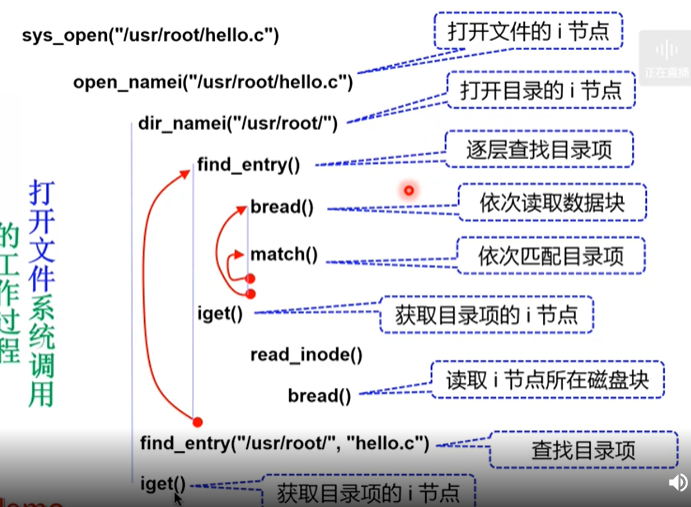
二、函数的具体实现

1.sys\_sleep

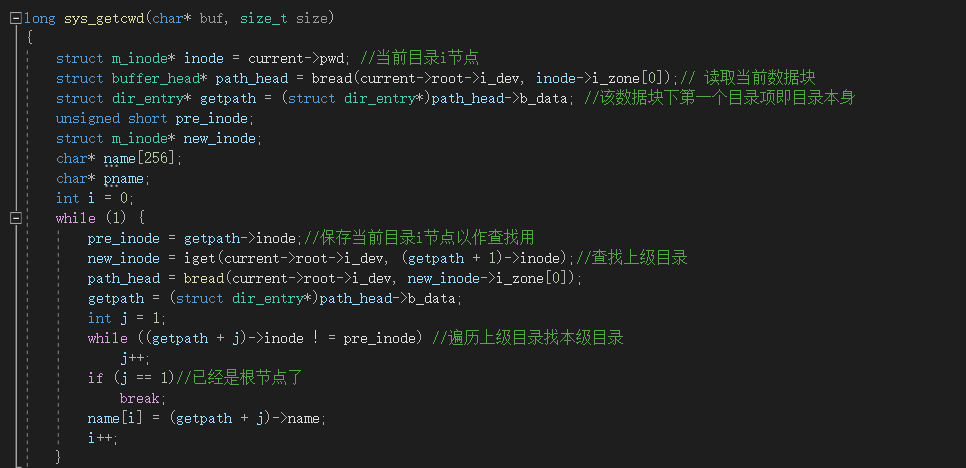
直接调用内核0版本自带的系统调用即可，先使用sys\_signal忽略SIGALRM信号，然后使用sys\_alarm设定seconds秒后唤醒后，调用sys\_pause



2.sys\_getcwd



原理和上图相似不过getcwd是自下而上寻找上级目录，

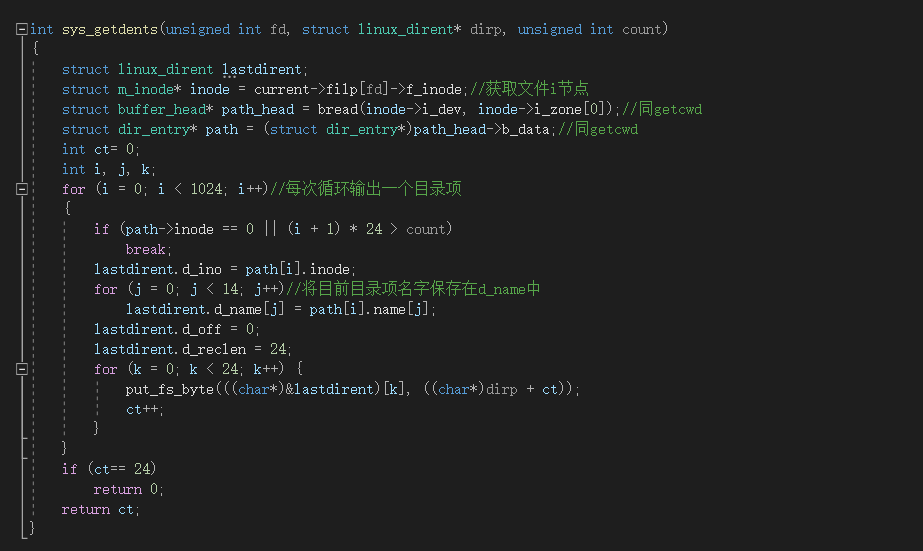


首先由current->pwd得到当前目录索引节点，然后用bread打开相应数据块并找到上级目录，然后得到上级目录索引节点及数据块，然后在上级目录的数据块中通过比较索引节点号找到本目录的名称信息并记录在name字符串数组中，用while循环重复此过程直至本级目录为根目录。



在这之后就是将路径名整理一下变成相应的格式（主要是调顺序和加/），并调用put\_fs\_byte将整理得到的路径名输出。

3.sys\_getdents

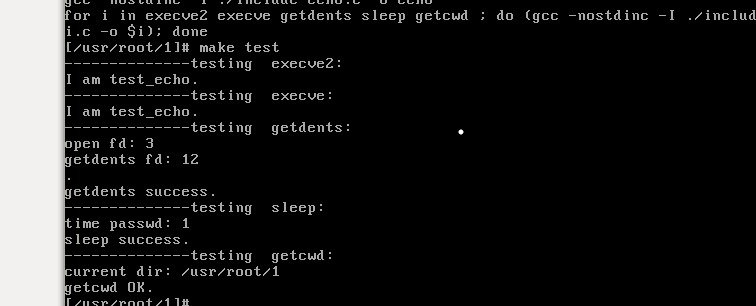


与getcwd类似，不同的是是根据当前目录的文件描述符获取当前目录索引节点，然后根据索引节点获得该目录文件的数据，依次遍历每个页表项，并存进结构体，直到遍历所有目录项。

4.do\_execve2

主要是仿照execve函数， 而do\_execve2函数与do\_execve函数的不同就在于do\_execve2需要提前进行读取代码段和数据段，避免页故障.。

三实验结果



全部执行成功