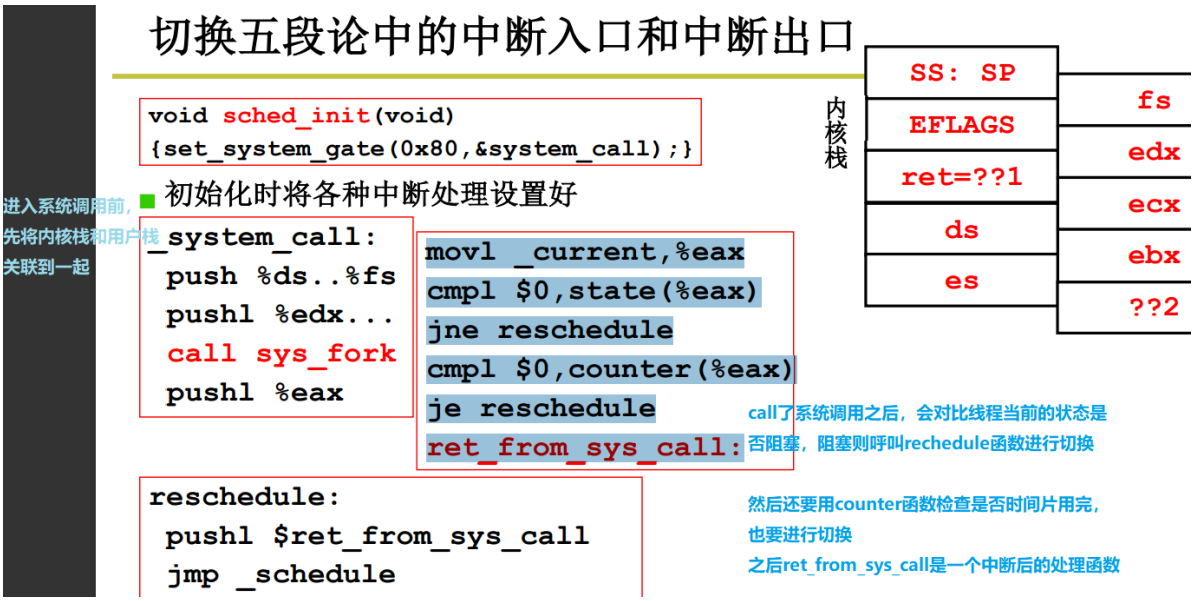
哈工大操作系统-L12核心级线程实现实例

核心级线程的两套栈，核心是内核栈。

1.五段论--1中断入口、5中断出口



- 1.中断入口：在进入中断之前，先把用户栈以及用户态的各种东西保存在内核栈中，建立用户栈和内核栈之间的关联。
- 5.中断出口：reschedule返回之后，一定是回到我们的ret_from_sys_call
  - ret_from_sys_call是中断出口，会把内核栈中的所有东西都pop出来，同时返回到某个线程调用中断的命令之后的第一个命令
- 的

2.五段论--2~4schedule中间发生了什么?

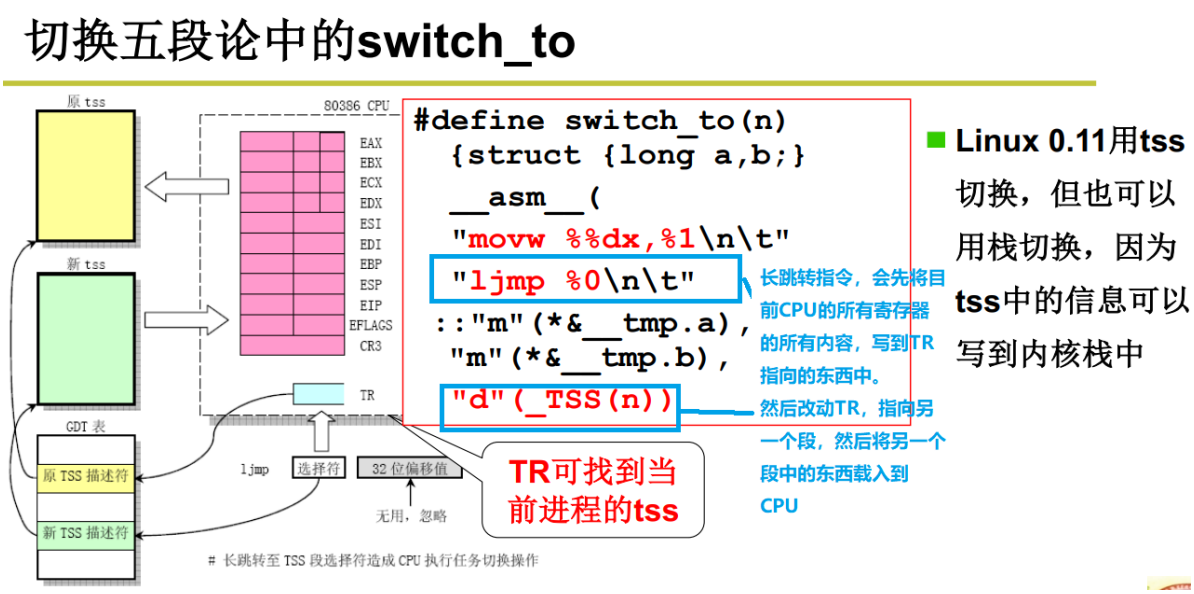
```
reschedule:pushl $ret_from_sys_call
 jmp _schedule
```

```
void schedule(void){ next=i;
  switch_to(next); }
```

- 首先我们可以看到，进入调度程序之后，会把当前线程的中断出口函数保存在当前线程的内核栈中。
- 而后，真正进入调度程序schedule，找到应该调度的下一个线程的TCB在next中保存，然后进行switch_to
  - **如何找到next是一个调度的问题**

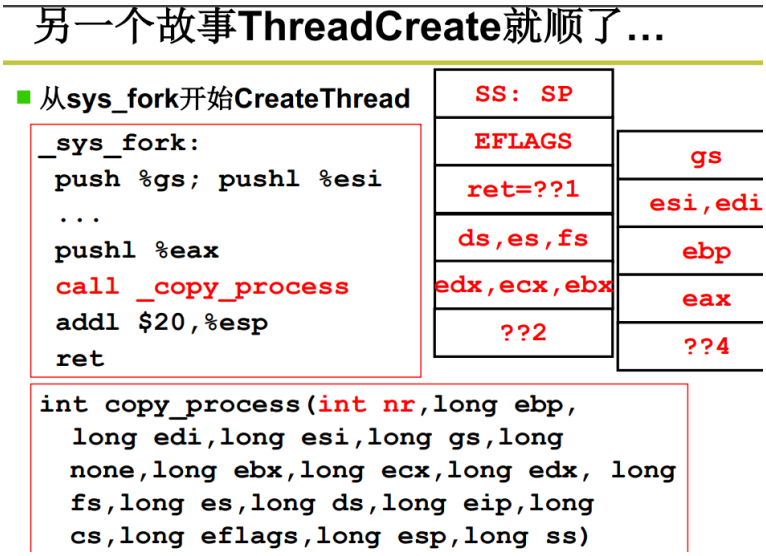
- 然后就可以直接切换到下一个线程的TCB
- 然后内核会根据TCB完成内核栈的切换
- 内核紧接着会根据切换完成的内核栈，完成需要执行的指令序列的切换

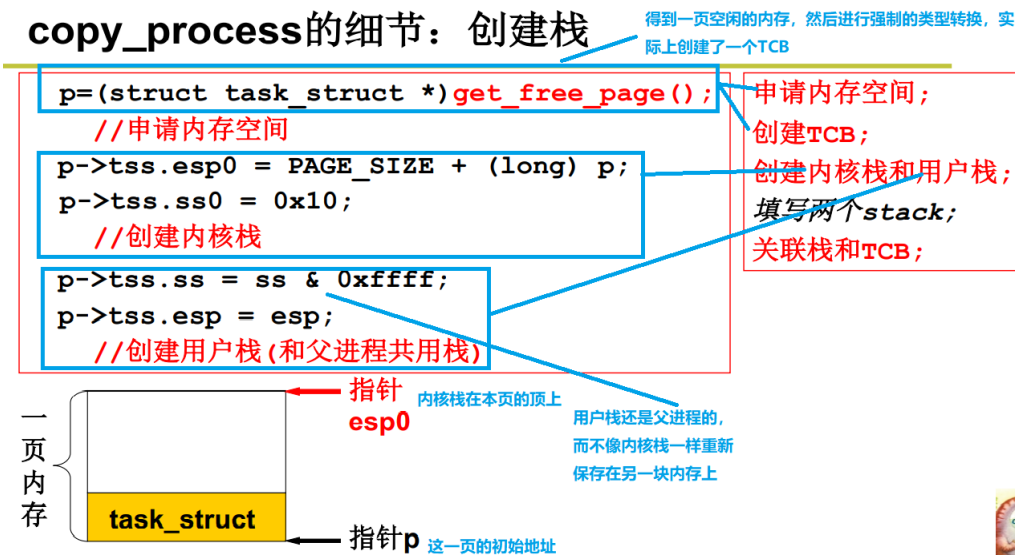
2.1TSS切换



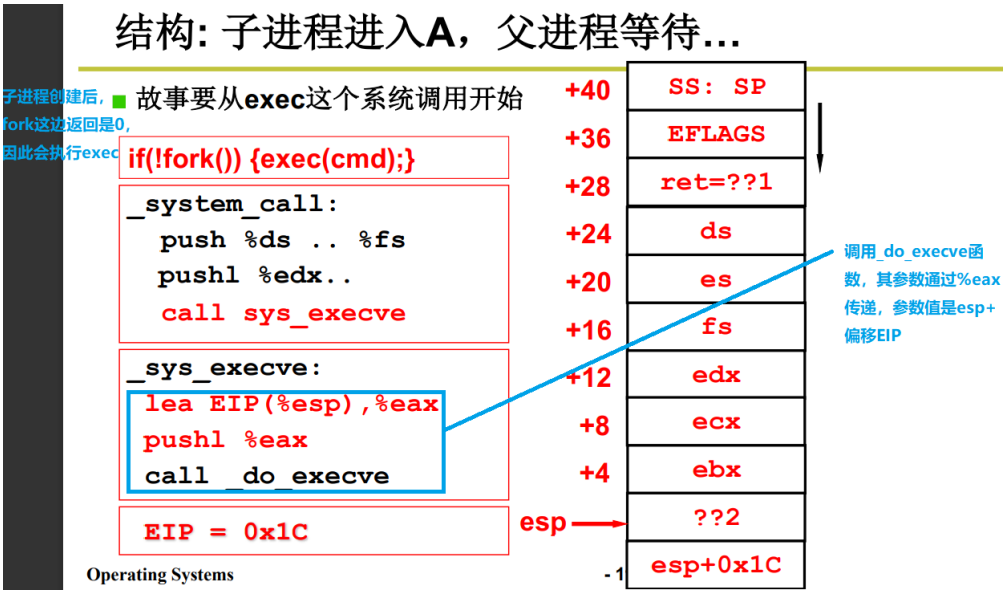
但TSS切换效率低

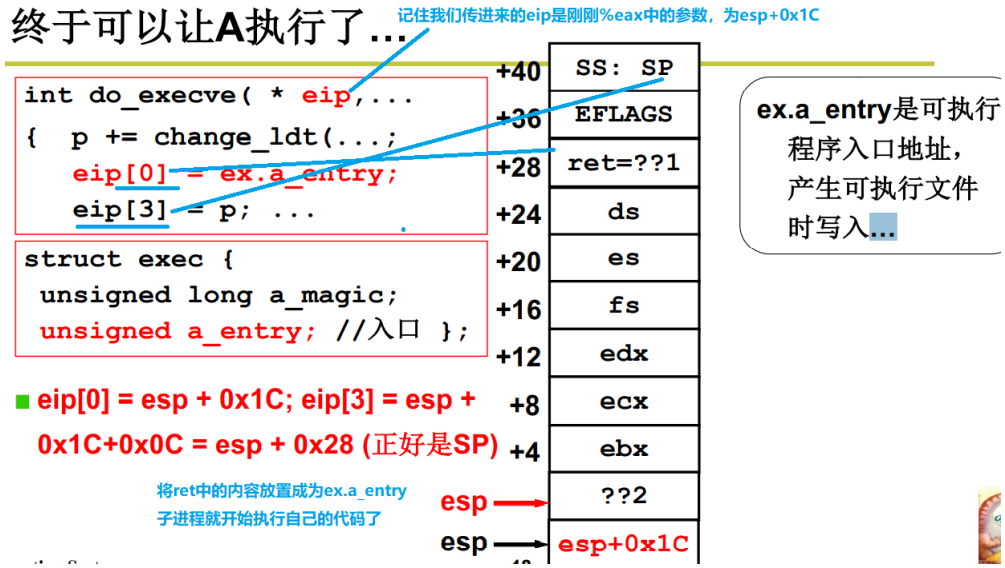
2.2创建一个线程--把线程做成能切换的样子





2.3子进程如何执行自己的代码





每个进程的代码中应该有一句话if(!fork()){}。 这样子就能区分，如果是从该进程创建了子进程，那么子进程会执行什么东西。