

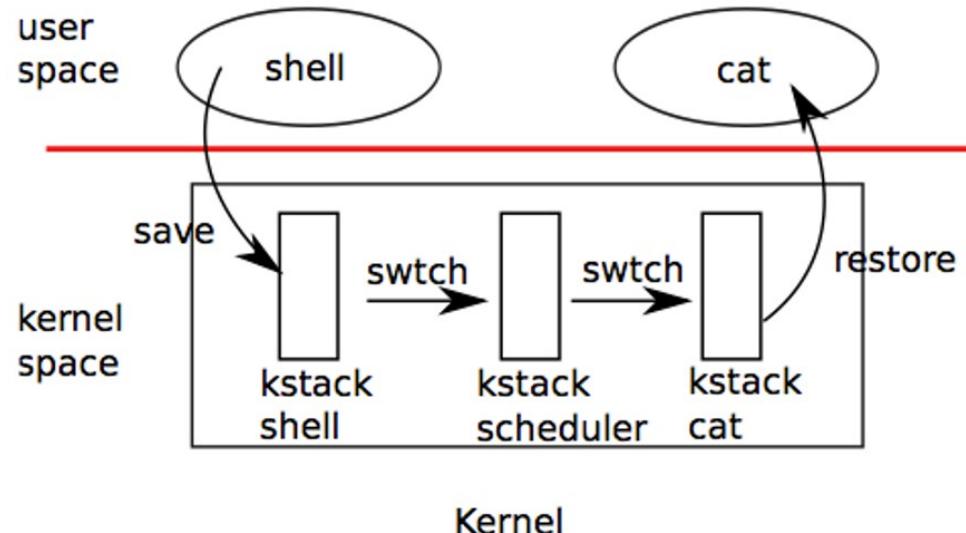


操作系统

Lab3: xv6实践 / 处理机调度
谭舜泉
计算机与软件学院

阅读xv6中文文档第五章“调度”

- xv6 在低层次中实现了两种上下文切换：
 - 从进程的内核线程切换到当前 CPU 的调度器线程；
 - 从调度器线程到进程的内核线程。



阅读xv6中文文档第五章“调度”

- 每个 CPU 都有一个**单独的**调度器线程，调度发生在调度器线程中。
- 进程的内核线程调用 **swtch (swtch.S)** 来保存自己的上下文然后返回到调度器的上下文中。
- (trap.c L105) xv6是分时系统，在收到时钟中断时，一个进程调用**yield**函数。



题目1

- 试解释一下**yield**函数、**scheduler**函数和**sched**函数的用途。
- 结合书本，确定**xv6**使用的是哪种调度算法。给出你的理由（通过分析代码证明你的观点）。
- (**proc.c L268**) 有一个疑问，似乎每次**xv6**都是从进程表开头开始查找**Runnable**的进程。如果刚从CPU切换下来的进程恰好是进程表的第一个PCB，会不会调度器永远都选择它进行调度？



题目2

- 在xv6界面上按Ctrl+P，你会看到终端上会显示当前系统中的进程。
- 仿照相关代码，实现以下功能：
 - 在xv6界面上按Ctrl+R，打印出当前系统中 **sleeping** 的进程，并确定它们对应的等待队列是什么（chan/waiting channel）。



实验报告

- 实验报告要求使用我们所提供的实验报告模板填写。
- 允许两至三人组成一个小组，撰写实验报告。
- 请按照实验要求，完成实验并撰写实验报告。把实验报告电子版上传到BLACKBOARD。本实验分组完成。每个实验小组完成一份实验报告即可。但要求小组中每个成员都把实验报告的副本上传到BLACKBOARD，并注明同组成员的名字和学号。

