# Lab5

## 练习3：阅读分析源代码,理解进程执行 fork/exec/wait/exit 的实现,以及系统调用的实现(不需要编码)

回答问题：

请分析fork/exec/wait/exit在实现中是如何影响进程的执行状态的?

Fork函数，不会去影响当前进程，但是系统调用时会保存当前进程的一些东西（esp寄存器的内容），让后续进程可以调度起来。

Exec：系统调用exec函数时，会保存当前运行程序的长度和大小，然后调用do\_exec，该函数也不会影响进程的执行，但是会修改进程的程序段，而且会保存中断位置，便于返回后可以跳转到正确的程序入口。

Wait：系统调用wait，会把当前程序的id和位置传给do\_wait函数，该函数是否影响当前状态取决于他的孩子进程，如果孩子进程状态为zomble，就释放孩子进程占用的资源，然后父进程继续执行，如果子进程不是这个状态，那么父进程会进入sleep状态。

Exit：系统调用exit时，会返回一个错误值，调用do\_exit然后该程序会影响当前进程，释放当前进程占用的资源，然后标记为zomble，之后还会唤醒父进程，如果有的话。

请给出ucore中一个用户态进程的执行状态生命周期图(包执行状态,执行状态之间的变

换关系,以及产生变换的事件或函数调用)。(字符方式画即可)

Zomble

yield（）

Exit（）

等待运行

正在运行的进程

正常调度

wait（），无zomble子进程

exit（），唤醒父进程

Sleep

执行:make grade。如果所显示的应用程序检测都输出ok,则基本正确。(使用的是qemu-

1.0.1)图片包含 屏幕截图

描述已自动生成