

总复习



期末考试范围



第四章 进程管理

UNIX的时钟中断 与 系统调用

UNIX的进程控制



第四章 进程管理

UNIX的时钟中断 与 系统调用

UNIX的进程控制

每次脉冲

每秒计时到

lbolt++
p_cpu++

p_time++, lbolt-60
p_cpu-20

重算p_pri (*SRUN*)
1秒计时到或现运行进程
优先级下降-Run Run++

唤醒0#进程 (@ *RunIn*)
唤醒延时睡眠进程



第四章 进程管理

UNIX的时钟中断

与

系统调用

UNIX的进程控制

每次脉冲

每秒计时到

系统调用号和参数通过现场
保护进入核心栈

lbolt++
p_cpu++

p_time++, lbolt-60
p_cpu-20

重算p_pri (*SRUN*)
1秒计时到或现运行进程
优先级下降-Run Run++

唤醒0#进程 (@ *RunIn*)
唤醒延时睡眠进程

进程入睡

进程被唤醒

重算p_pri (*SRUN*)
现运行进程优先级下降-
RunRun++

被抢占



期末考试范围



第四章 进程管理

UNIX的时钟中断

与 系统调用

UNIX的进程控制

每次脉冲

lbolt++
p_cpu++

每秒计时到

p_time++, lbolt-60
p_cpu-20

重算p_pri (SRUN)
1秒计时到或现运行进程
优先级下降-Run Run++

唤醒0#进程 (@ RunIn)
唤醒延时睡眠进程

系统调用号和参数通过现场
保护进入核心栈

进程入睡

Sleep

设置p_wchan, p_stat
设置p_pri
低睡-唤醒0#进程 (@ RunIn)
下台

进程被唤醒

WakeUpAll

SetRun

清除p_wchan, p_stat
在磁盘-唤醒0#进程 (@ RunOut)
优先级更高-RunRun++

重算p_pri (SRUN)
现运行进程优先级下降-
RunRun++

被抢占



期末考试范围



第四章 进程管理

UNIX的时钟中断

每次脉冲

lbolt++
p_cpu++

每秒计时到

p_time++, lbolt-60
p_cpu-20

重算p_pri (*SRUN*)
1秒计时到或现运行进程
优先级下降-Run Run++

唤醒0#进程 (@ *RunIn*)
唤醒延时睡眠进程

与 系统调用

系统调用号和参数通过现场
保护进入核心栈

进程入睡

Sleep

设置p_wchan, p_stat
设置p_pri
低睡-唤醒0#进程 (@ *RunIn*)
下台

进程被唤醒

WakeUpAll

SetRun

清除p_wchan, p_stat
在磁盘-唤醒0#进程 (@ *RunOut*)
优先级更高-RunRun++

重算p_pri (*SRUN*)
现运行进程优先级下降-
RunRun++

被抢占

UNIX的进程控制

什么样的现场?

Switch

现场保护

切换到0#

选新进程

切换到新进程

现场恢复

return 1



期末考试范围



第四章 进程管理

UNIX的时钟中断

每次脉冲

lbolt++
p_cpu++

每秒计时到

p_time++, lbolt-60
p_cpu-20

重算p_pri (SRUN)
1秒计时到或现运行进程
优先级下降-Run Run++

唤醒0#进程 (@ RunIn)
唤醒延时睡眠进程

与 系统调用

系统调用号和参数通过现场
保护进入核心栈

进程入睡

Sleep

设置p_wchan, p_stat
设置p_pri
低睡-唤醒0#进程 (@ RunIn)
下台

进程被唤醒

WakeUpAll

SetRun

清除p_wchan, p_stat
在磁盘-唤醒0#进程 (@ RunOut)
优先级更高-RunRun++

重算p_pri (SRUN)
现运行进程优先级下降-
RunRun++

被抢占

UNIX的进程控制

Switch

现场保护

切换到0#

选新进程

切换到新进程

现场恢复

return 1

选择的依据?
选不出来怎么办?



期末考试范围



第四章 进程管理

UNIX的时钟中断

每次脉冲

lbolt++
p_cpu++

每秒计时到

p_time++, lbolt-60
p_cpu-20

重算p_pri (SRUN)
1秒计时到或现运行进程
优先级下降-Run Run++

唤醒0#进程 (@ RunIn)
唤醒延时睡眠进程

与 系统调用

系统调用号和参数通过现场
保护进入核心栈

进程入睡

Sleep

设置p_wchan, p_stat
设置p_pri
低睡-唤醒0#进程 (@ RunIn)
下台

进程被唤醒

WakeUpAll

SetRun

清除p_wchan, p_stat
在磁盘-唤醒0#进程 (@ RunOut)
优先级更高-RunRun++

重算p_pri (SRUN)
现运行进程优先级下降-
RunRun++

被抢占

UNIX的进程控制

Switch

现场保护

切换到0#

选新进程

切换到新进程

现场恢复

return 1

如何实现进程的
切换?



期末考试范围

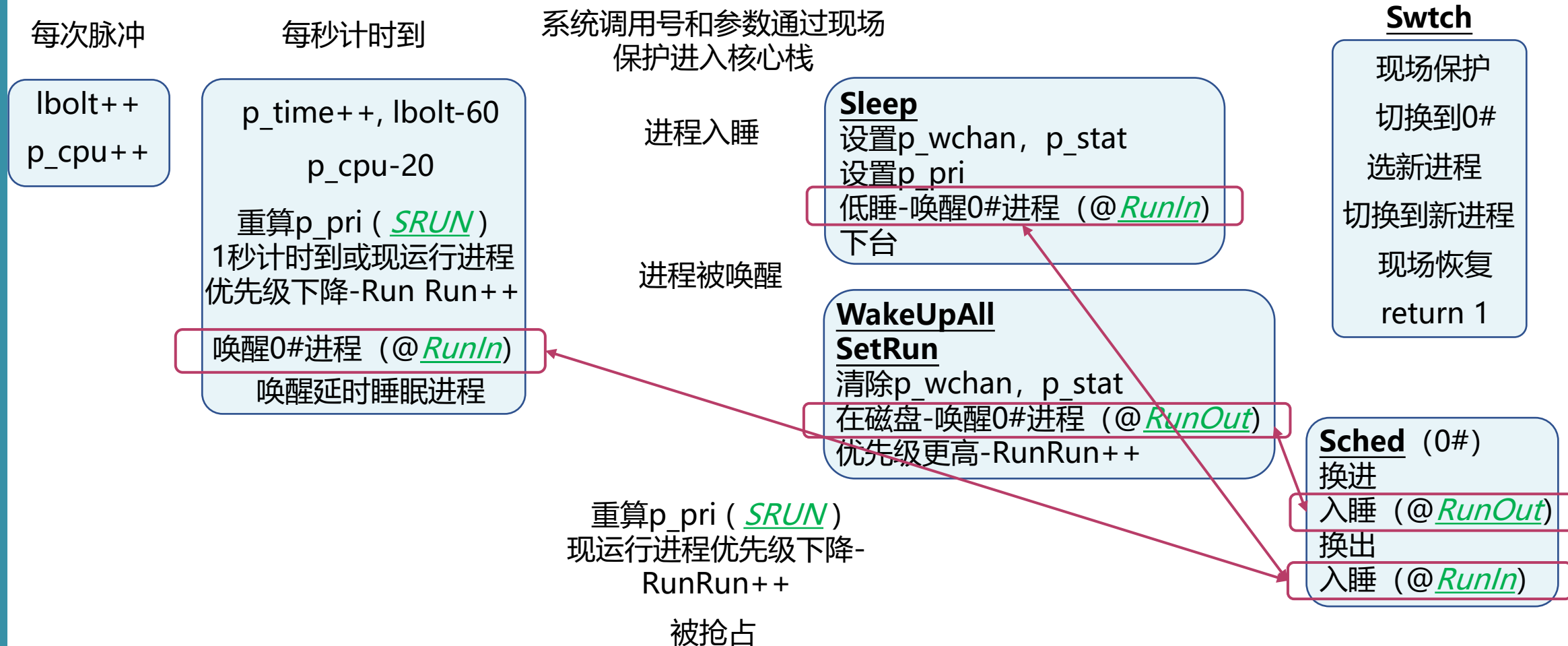


第四章 进程管理

UNIX的时钟中断

与 系统调用

UNIX的进程控制





期末考试范围



第四章 进程管理

UNIX的时钟中断

与 系统调用

UNIX的进程控制

每次脉冲

lbolt++
p_cpu++

每秒计时到

p_time++, lbolt-60
p_cpu-20

重算p_pri (*SRUN*)
1秒计时到或现运行进程
优先级下降-Run Run++

唤醒0#进程 (@ *RunIn*)
唤醒延时睡眠进程

UNIX的动态优先权调度算法

系统调用号和参数通过现场
保护进入核心栈

进程入睡

进程被唤醒

重算p_pri (*SRUN*)
现运行进程优先级下降-
RunRun++

被抢占

Sleep

设置p_wchan, p_stat
设置p_pri
低睡-唤醒0#进程 (@ *RunIn*)
下台

WakeUpAll

SetRun
清除p_wchan, p_stat
在磁盘-唤醒0#进程 (@ *RunOut*)
优先级更高-RunRun++

Switch

现场保护
切换到0#

选新进程

切换到新进程
现场恢复
return 1

Sched (0#)

换进
入睡 (@ *RunOut*)
换出
入睡 (@ *RunIn*)



第四章 进程管理

创建子进程与父子进程同步

NewProc:

创建在盘交换区

return 0

子进程如何上台?
抢占父进程的条件?

wait和exit

父子进程间的同步关系
子进程图像的回收
终止码的传送

UNIX的进程控制

Switch

现场保护

切换到0#

选新进程

切换到新进程

现场恢复

return 1



第四章 进程管理

创建子进程与父子进程同步

NewProc:

创建在盘交换区

return 0

子进程如何上台?
抢占父进程的条件?

wait和exit

子进程图像的回收
终止码的传送

关于0#进程

UNIX的进程控制

Switch

现场保护

切换到0#

选新进程

切换到新进程

现场恢复

return 1



期末考试范围



所有的作业

S25, S31~S33 例图

P05, P07