

PART I

进程问题集锦

F AQ, 本章记录课间和课后宋老师以及同学们答疑

CHAPTERS IN THIS PART:

课后答疑

以下问题为宋老师在微信答疑群中回答记录，微信群问题不定期提出，此部分内容会随之更新。

Q: rps 后，非多队列网卡，中断会再给每个核发中断来派发软中断？

A: 中断只发一个 core，这个 core 自己给别的 core 发核间中断

1.

Q: rt 补丁，是不是只有 RT_FULL 支持优先级反转？

A: 不是，不需要 rt 补丁就支持优先级继承，早就 Merge 到了 mainline。

不叫支持优先级反转，反转是个问题，继承是解决它的方法。反转是个现象，不存在支持不支持。

你的问题是错误的。

2.

Q: 所以 softirq 的优先级都是相同的吗？

A: 不是的，它是一个 bit 的设置，检查哪个 bit 被设置，肯定是有先后顺序。挨个检查哪个 bit 的。不过这个不是关键点。你关心延迟的实时性的时候，你根本就消灭了 softirq。

3.

Q: 内核管理多核好理解，但内核如何管理多 CPU？

A: 异构多 os 和 Linux 没关系，那是多 os 之间的问题，不是 Linux 管理范畴，几个 OS 一个通信方法即可。

4.

Q: cfs 调度单位是 task_struct, 那像 HMP EAS 这些调度器单位是什么?

A: 调度单元与调度算法无关, 你说的调度器和我们说的调度器不一定是一个意思。调度单元就算线程。这个不以任何操作系统, 任何算法为转移。

5.

Q: 课上说多核可以运行 rtos+linux, 那是不是单核各自运行一个 OS, 如何实现, 是启动 linux 后, 再启动 rtos 运行到某个核吗?

A: 多个 core 单独各玩各的, 谁先启动没有讲究, 取决于产品, 这个和 linux 没有关系, 两个 CPU 各自运行自己的一套软件。你直接想象成两个电脑就好了。

6.

Q: rt 的 Linux 如何对付内存的 lazy 问题? 能否改用内核的 api 解决用户态内存申请的实时性问题。

A: 内核和用户态都是用相同的算法来进行调度, 内核的调度没有比用户态更优越, 用户态的调度策略和优先级比内核高时也可以抢占内核的。内存方面 lazy 有它的解法, 主要是先申请 malloc, 立即释放, 稳住堆; 提前调用一个很大的临时变量的函数, 稳住栈; 提前把线程都创建; mlockall 屏蔽交换。这些都有套路, 看文档即可。

7.

Q: 有人说一但调用了 `|_dofirq|`

A: 该怎么抢怎么抢, 该抢的你让它抢, 你只需保证抢了后的数据不出错即可。

8.