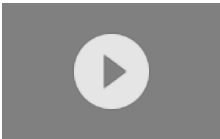




张爱玲的悲剧在于明知故犯

[新浪首页](#) [登录](#) [注册](#)



南城titi的博客

<http://blog.sina.com.cn/u/2736779881> [\[订阅\]](#) [\[手机订阅\]](#)

[首页](#) [博文目录](#) [图片](#) [关于我](#)

个人资料

正文

字体大小：[大](#) [中](#) [小](#)



南城titi

微博

加好友

发纸条

写留言

加关注



博客等级：**13**  
博客积分：**178**  
博客访问：**34,268**  
关注人气：**8**  
获赠金笔：**0**  
赠出金笔：**0**  
荣誉徽章：

相关博文

实拍：古村留守儿童的春节  
大山的孩子  
  
街拍：穿丝袜露大腿最拉风  
曹作兰

[更多>>](#)

推荐博文

日本电视台都从哪儿选拔美女主播  
  
济州联在抢钱吗？  
  
加拿大-美国：雪地里的偷越边境  
  
侯虹斌：不暴富，就对不起这伟大  
  
克里米亚到底有多重要

LINUX中断机制与信号 (2012-05-30 09:10:19)

[转载](#) ▼

标签： 杂谈 分类： linux内核机制

LINUX中断机制与信号 中断和异常

1 中 断（也称硬件中断）

定义：中断是由其他硬件设备依照CPU 时钟周期信号随机产生的（异步中断）。

分类： 可屏蔽中断

非可屏蔽中断

来源： 间隔定时器和I/O

1 异 常（也称软件中断）

定义：当指令执行时由 CPU控制单元 产生的，异常也称为“同步中断”是因为只有在 一条指令终止执行后CPU 才会发出中断。

分类： 处理器探测到的异常

2 故障

2 陷阱

2 异常终止

编程异常（ 也称软中断 ）

2 int指令

来源： 程序的错误产生的

内核必须处理的异常（ 例如：缺页和内核服务的请求 -int）

异常处理

1 当发生异常时，CPU 控制单元产生一个硬件出错码。

1 CPU根据该中断码找到中断向量表内的对应向量，根据该向量转到中断处理程序。

1 中断处理程序处理完之后向当前进程发送一个SIG\*\*\* 信号。

1 若进程定义了相应的信号处理程序则转移到相应的程序执行，若没有，则执行内核定义的操作。

中断处理

1 设备产生中断

1 PIC（可编程中断控制器）会产生一个对应的中断向量

1 和中断向量表中的每一个中断向量进行比较，转到对应的中断处理程序

1 中断处理程序进行保存现场，做相关处理，恢复现场

1 内核调度，返回用户进程



硬件中断的上半部和下半部及实现方式

1 硬件中断的分类

2 紧急的 —— 这类中断必须立即执行

2 非紧急的 —— 也必须立即执行

2 非紧急可延迟的 —— 上半部立即执行，下半部延迟执行

美国总统川普上任一月余，斗媒体

宁泽涛离开国家队是一个两败俱伤

周碧华：江西宁都的

“亲生亲卖”的父母比人贩子罪孽

要像打击电信诈骗那样打击保健品



气质典雅的美女



太阳最先照到的地方



极寒天气洒水成冰



雪后紫禁城美的不要不要的



黑白色调叶梓萱



他们居然这样取野生蜂蜜

[查看更多>>](#)

谁看过这篇博文

扬子	7月3日
Shawngong	6月18日
用户19379...	12月23日
李丰翔	10月3日
wanli15	9月23日
Agoni	3月23日
wavabobo	1月16日
稻田男孩	12月2日
抱抱熊	5月8日
fantasia	3月29日

硬件中断任务（处理程序）是一个快速、异步、简单地对硬件做出迅速响应并在最短时间内完成必要操作的中断处理程序。硬中断处理程序可以抢占内核任务并且执行时还会屏蔽同级中断或其它中断，因此中断处理必须要快、不能阻塞。这样一来对于一些要求处理过程比较复杂的任务就不合适在中断任务中一次处理。比如，网卡接收数据的过程中，首先网卡发送中断信号告诉 CPU 来取数据，然后系统从网卡中读取数据存入系统缓冲区中，再下来解析数据然后送入应用层。这些如果都让中断处理程序来处理显然过程太长，造成新来的中断丢失。因此 Linux 开发人员将这种任务分割为两个部分，一个叫上底，即中断处理程序，短平快地处理与硬件相关的操作（如从网卡读数据到系统缓存）；而把对时间要求相对宽松的任务（如解析数据的工作）放在另一个部分执行，这个部分就是我们这里要讲的下半底。

下半底是一种推后执行任务，它将某些不那么紧迫的任务推迟到系统更方便的时刻运行。因为并不是非常紧急，通常还是比较耗时的，因此由系统自行安排运行时机，不在中断服务上下文中执行。内核中实现下半底的手段经过不断演化，目前已经从最原始的BH(bottom thalf) 演生出 BH 、任务队列（Task queues）、软中断（Softirq）、Tasklet、工作队列（Work queues）（2.6 内核中新出现的）。

其中的软中断和异常中提到的软中断的区别：

主要是用来处理非紧急可延迟的硬件中断

Linux系统定义的，不是用户定义，并且个数有限



关于软中断和硬中断的其它解析：

软中断一般是指由指令int 引起的“伪”中断动作——给 CPU 制造一个中断的假象；而硬中断则是实实在在由 8259 的连线触发的中断。因此，严格的讲，int 与 IRQ 毫无关系，但二者均与中断向量有关系。int 引起的中断，CPU 是从指令中取得中断向量号；而 IRQ 引起的中断，CPU 必须从数据线上取回中断号，接下来 CPU 的工作就一样了：保护现场、根据中断号得到中断处理程序地址、执行中断处理、恢复现场继续执行被中断的指令。

在软中断和硬中断之间的区别是什么？

①硬中断是由外部事件引起的因此具有随机性和突发性；软中断是执行中断指令产生的，无面外部施加中断请求信号，因此中断的发生不是随机的而是由程序安排好的。

②硬中断的中断响应周期，CPU 需要发中断回合信号（NMI 不需要），软中断的中断响应周期，CPU 不需发中断回合信号。

③硬中断的中断号是由中断控制器提供的（NMI 硬中断中断号系统指定为 02H）；软中断的中断号由指令直接给出，无需使用中断控制器。

④硬中断是可屏蔽的（NMI 硬中断不可屏蔽），软中断不可屏蔽。

LINUX信号机制

1 信号本质

信号是异步的进程间通讯机制，是在软件层次上对中断机制的一种模拟，在原理上，一个进程收到一个信号与处理器收到一个中断请求可以说是一样的。信号是异步的，一个进程不必通过任何操作来等待信号的到达，事实上，进程也不知道信号到底什么时候到达。

信号是进程间通信机制中唯一的异步通信机制，可以看作是异步通知，通知接收信号的进程有哪些事情发生了。内核也可以因为内部事件而给进程发送信号，通知进程发生了某个事件。注意，信号只是用来通知某进程发生了什么事件，并不给该进程传递任何数据。

1 产生信号的条件主要有：

1. 用户在终端按下某些键时，终端驱动程序会发送信号给前台进程，例如Ctrl-C 产生 SIGINT 信号，Ctrl-/ 产生 SIGQUIT 信号，Ctrl-Z 产生 SIGTSTP 信号。

2. 硬件异常产生信号，这些条件由硬件检测到并通知内核，然后内核向当前进程发送适当的信号。例如当前进程执行了除以0 的指令，CPU 的运算单元会产生异常，内核将这个异常解释为 SIGFPE 信号发送给进程。再比如当前进程访问了非法内存地址，MMU 会产生异常，内核将这个异常解释为 SIGSEGV 信号发送给进程。

3. 一个进程调用kill(2) 函数可以发送信号给另一个进程。

4. 可以用kill(1) 命令发送信号给某个进程，kill(1) 命令也是调用 kill(2) 函数实现的，如果不明确指定信号则发送 SIGTERM 信号，该信号的默认处理动作是终止进程。

5. 当内核检测到某种软件条件发生时也可以通过信号通知进程，例如闹钟超时产生SIGALRM 信号，向读端已关闭的管道写数据时产生 SIGPIPE 信号。

1 进程对信号的处理：

1. 忽略此信号。

2. 执行该信号的默认处理动作。

3. 提供一个信号处理函数，要求内核在处理该信号时切换到用户态执行这个处理函数，这种方式称为捕捉（Catch）一个信号。

## 1 信号与中断的相似点:

- (1) 采用了相同的异步通信方式;
- (2) 当检测出有信号或中断请求时, 都暂停正在执行的程序而转去执行相应的处理程序;
- (3) 都在处理完毕后返回到原来的断点;
- (4) 对信号或中断都可进行屏蔽。

## 1 信号与中断的区别:

- (1) 中断有优先级, 而信号没有优先级, 所有的信号都是平等的;
- (2) 信号处理程序是在用户态下运行的, 而中断处理程序是在核心态下运行;
- (3) 中断响应是及时的, 而信号响应通常都有较大的时间延迟。

## 1 信号机制具有以下三方面的功能:

- (1) 发送信号。发送信号的程序用系统调用 `kill()` 实现;
- (2) 预置对信号的处理方式。接收信号的程序用 `signal()` 来实现对处理方式的预置;
- (3) 收受信号的进程按事先的规定完成对相应事件的处理。

## 1 信号捕获过程



本图出自ULK

注: 从上图可以看出, 信号的处理时机是在当前进程由于系统调用、中断或者异常而进入系统空间以后, 从系统空间返回到用户空间前夕(这样做主要是出于效率的考虑, 和进程的调用时机一致, 从系统调用返回意味着要离开内核态而返回到用户态, 而状态的转换要花费一定的时间, 因此, 在返回到用户态前, 系统把在内核态该处理的事全部做完)。

- 下图为异常和函数产生信号的方式以及进程的处理过程



另外提供两个关于Linux中断机制和信号机制的学习文章

关于软中断: <http://www.ibm.com/developerworks/cn/linux/kernel/interrupt/index.html>

关于信号: <http://www.ibm.com/developerworks/cn/linux/1-ipc/part2/index1.html>

<http://www.ibm.com/developerworks/cn/linux/1-ipc/part2/index2.html>

0

0

喜欢

赠金笔

分享:

阅读(250) | 评论(0) | 收藏(0) | 转载(1) | 喜欢▼ | 打印 | 举报

已投稿到: 排行榜

前一篇: [linux--网络编程](#)

后一篇: [linux信号signal处理函数](#)

评论

重要提示: 警惕虚假中奖信息

[发评论]

评论加载中, 请稍候...

发评论



登录名:  密码:  [找回密码](#) [注册](#) ☒ 记住登录状态

☒ 分享到微博 ☐ 评论并转载此博文

按住左边滑块，拖动完成上方拼图

发评论

以上网友发言只代表其个人观点，不代表新浪网的观点或立场。

< 前一篇

linux—网络编程

后一篇 >

linux信号signal处理函数

新浪BLOG意见反馈留言板 不良信息反馈 电话: 4006900000 提示音后按1键 (按当地市话标准计费) 欢迎批评指正  
新浪简介 | About Sina | 广告服务 | 联系我们 | 招聘信息 | 网站律师 | SINA English | 会员注册 | 产品答疑

Copyright © 1996 - 2017 SINA Corporation, All Rights Reserved  
新浪公司 版权所有