

CSDN

博客 下载 学习 社区 GitCode InsCode

BSP面试

搜索 登录 会员中心 历史 创作中心

原创

kavin.zhu

已于 2022-05-31 10:55:55 修改

39903

收藏 681

版权

分类专栏：

interview

linux

 文章标签：

linux

linux 驱动

linux面试

BSP面试

 华为云开发者联盟 该内容已被华为云开发者联盟社区收录

加入社区

 interview 同时被 2 个专栏收录

9 订阅 3 篇文章 订阅专栏

版权声明：本文为博主原创文章，未经博主允许不得转载。

转载请标明原址：[linux驱动最新面试题（面试题整理，含答案）_不忘初心-CSDN博客](#)
[_linux驱动面试题](#)

前言：这篇文章主要是对linux驱动面试题一个整理跟总结，参考了很多网上的资料，基本涵盖linux驱动相关面试内容。我把他们大概的分为三部分：基础部分，同步相关，还有中断部分。中断，同步相关基本都是必问的。下面也会对这这几个方面的面试题进行详细的解答，你把下面的面试题看懂了，应该可以应付大部分linux驱动面试了。要想真正的理解，还的在实践中多动手调试多总结，如果有什么地方错了或者不全，欢迎小伙伴们留言。

一. 基础题：

1. linux中内核空间及用户空间的区别？用户空间与内核通信方式有哪些？

[linux内核空间和用户空间的是怎样区别的，如何交互，如何从用户空间进入内核空间_bingqingsuimeng的专栏-CSDN博客](#)

2. 字符设备和块设备的区别，请分别列举一些实际的设备说出它们是属于哪一类设备

字符设备：字符设备是个能够像字节流（类似文件）一样被访问的设备，由字符设备驱动程序来实现这种特性。字符设备驱动程序通常至少实现open,close,read和write系统调用。字符终端、串口、鼠标、键盘、摄像头、声卡和显卡等就是典型的字符设备。

块设备：和字符设备类似，块设备也是通过/dev目录下的文件系统节点来访问。块设备上能够容纳文件系统，如：u盘，SD卡，磁盘等。

字符设备和块设备的区别仅仅在于内核内部管理数据的方式，也就是内核及驱动程序之间的软件接口，而这些不同对用户来讲是透明的。在内核中，和字符驱动程序相比，块驱动程序具有完全不同的接口。

3. linux内核的启动过程(源代码级)?

[Linux内核启动过程概述 - CrazyCatJack - 博客园](#)

4. linux中系统调用过程？如:应用程序中read()在linux中执行过程即从用户空间到内核空间？

[linux设备驱动框架_不忘初心-CSDN博客_linux设备驱动](#)

[Linux Read系统调用 - HAOMCU的个人空间 - OSCHINA - 中文开源技术交流社区](#)

5. linux调度原理？

[linux内核调度器 调度原理\(2.6.24笔记整理\)_Janneo.Evans的专栏-CSDN博客](#)

6. 查看驱动模块中打印信息应该使用什么命令？如何查看内核中已有的字符设备的信息？如何查看正在使用的有哪些中断号？

- 1) 查看驱动模块中打印信息的命令：`dmesg`
- 2) 查看字符设备信息可以用`lsmod`和`modprobe`，`lsmod`可以查看模块的依赖关系，`modprobe`在加载模块时会加载其他依赖的 模块。
- 3) 显示当前使用的中断号`cat /proc/interrupt`

7. copy_to_user()和copy_from_user()主要用于实现什么功能？一般用于file_operations结构的哪些函数里面？

由于内核空间和用户空间是不能互相访问的，如果需要访问就必须借助内核函数进行数据读写。`copy_to_user()`:完成内核空间到用户空间的复制，`copy_from_user()`：是完成用户空间到内核空间的复制。一般用于file_operations结构里的read,write,ioctl等内存数据交换作用的函数。当然，如果ioctl没有用到内存数据复制，那么就不会用到这两个函数。

 **kavin.zhu**

关注

 63



 681



 4



专栏目录



组织。

次设备号：次设备号由内核使用，用于正确确定设备文件所指的设备。依赖于驱动程序的编写方式，我们可以通过次设备号获得一个指向内核设备的直接指针，也可将此设备号当作设备本地数组的索引。

2) char test 表示设备节点，4表示主设备号，64表示次设备号。（感觉类似于串口终端或者字符设备终端）。

9. 设备驱动程序中如何注册一个字符设备？分别解释一下它的几个参数的含义。

注册一个字符设备驱动有两种方法：

1) void cdev_init(struct cdev *cdev, struct file_operations *fops)

该注册函数可以将cdev结构嵌入到自己的设备特定的结构中。cdev是一个指向结构体cdev的指针，而fops是指向一个类似于file_operations结构（可以是file_operations结构，但不限于该结构）的指针。

2) int register_chrdev(unsigned int major, const char *name, struct file_operations *fops);

该注册函数是早期的注册函数，major是设备的主设备号，name是驱动程序的名称，而fops是默认的file_operations结构（这只是只限于file_operations结构）。对于register_chrdev的调用将为给定的主设备号注册0—255作为次设备号，并为每个设备建立一个对应的默认cdev结构。

10. linux中RCU原理？

[Linux中的RCU机制的分析 - eversliver - 博客园](#)

11. linux内存如何划分以及如何使用？虚拟地址及物理地址的概念以及转换，高端内存的概念？

[linux环境内存分配原理--虚拟内存 mallocinfo - dzqdevin - 博客园](#)

[Linux用户空间与内核空间（理解高端内存） - Jessica程序猿 - 博客园](#)

12. 字符型驱动设备怎么创建设备文件？

手动创建：mknod /dev/led c 250 0 其中dev/led 为设备节点 c 代表字符设备 250代表主设备号 0代表次设备号

还有UDEV/MDEV自动创建设备文件的方式，UDEV/MDEV是运行在用户态的程序，可以动态管理设备文件，包括创建和删除设备文件，运行在用户态意味着系统要运行之后。在/etc/init.d/rcS 脚本文件中会执行 mdev -s 自动创建设备节点。

13. insmod 一个驱动模块，会执行模块中的哪个函数？rmmod呢？这两个函数在设计上要注意哪些？遇到过卸载驱动出现异常没？是什么问题引起的？

答：insmod调用init函数，rmmod调用exit函数。这两个函数在设计时要注意什么？卸载模块时曾出现卸载失败的情形，原因是存在进程正在使用模块，检查代码后发现产生了死锁的问题。

要注意在init函数中申请的资源在exit函数中要释放，包括存储，ioremap，定时器，工作队列等等。也就是一个模块注册进内核，退出内核时要清理所带来的影响，带走一切不留下一点痕迹。

14. 设备驱动模型三个重要成员是？platform总线的匹配规则是？在具体应用上要不要先注册驱动再注册设备？有先后顺序没？

设备驱动模型三个重要成员是 总线、设备、驱动；

platform总线的匹配规则是：要匹配的设备 and 驱动都要注册，设备可以在 设备树 里注册，也可以通过代码注册设备，匹配成功会去调用驱动程序里的probe函数（probe函数在这个platform_driver结构体中注册）。

15. 内核函数mmap的实现原理，机制？

[linux中mmap系统调用原理分析与实现_yinjiabin的博客-CSDN博客_mmap系统调用](#)

[Mmap的实现原理和应用_edwardlulinux的专栏-CSDN博客_mmap原理](#)

16. 在驱动调试过程中遇到过oops没？你是怎么处理的？

[linux中Oops信息的调试及栈回溯—Linux人大家都知道，这是好东西！_Android/Linux的专栏-CSDN博客](#)



kavin.zhu

关注

63



681



4



专栏目录



举报

[ioctl与unlocked_ioctl区别_cbl709的专栏-CSDN博客_unlocked_ioctl](#)

18. 驱动中操作物理绝对地址为什么要先ioremap?

因为内核没有办法直接访问物理内存地址，必须先通过ioremap获得对应的虚拟地址

[Linux 字符设备驱动开发基础（五）—— ioremap\(\) 函数解析_知秋一叶-CSDN博客_ioremap函数](#)

19. 你平常是怎么用C写嵌入式系统的死循环的?

```
for(;;){
```

```
while(1){
```

一般for(;;)性能更优

```
for(;;){
```

这两个;; 空语句，编译器一般会优掉的，直接进入死循环

```
while(1){
```

每循环一次都要判断常量1是不是等于零，在这里while比for多做了这点事

不过从汇编的角度来说，都是一样的代码。

20. 列举最少3种你所知道的嵌入式的体系结构，并请说明什么是ARM体系结构。

arm,mips,x86

[ARM体系结构与编程模型总结_宋铮的博客-CSDN博客_arm体系结构与编程](#)

[ARM体系架构_和蔼的二师兄的专栏-CSDN博客](#)

21. kmalloc和vmalloc的区别

[Linux内核内存管理kmalloc,vmalloc - 慢伴拍的二叉树 - 博客园](#)

[Kmalloc和Vmalloc的区别 - Jessica程序猿 - 博客园](#)

22. IIC原理，总线框架，设备编写方法，i2c_msg

[i2c总线（基本原理）_不忘初心-CSDN博客_i2c](#)

[简单i2c设备驱动实例_不忘初心-CSDN博客](#)

23. kernel panic

[Kernel Panic常见原因以及解决方法 - 浩天之家 - 博客园](#)

24. Linux中的用户模式和内核模式是什么含意?

[linux的用户模式和内核模式_YJF@HZ的博客-CSDN博客](#)

25. 怎样申请大块内核内存?

vmalloc

26. 用户进程间通信主要哪几种方式?

https://blog.csdn.net/wh_sjc/article/details/70283843

27.linux编译时用到的参数含义及?

[linux 中常用编译参数解析_yqtao的博客-CSDN博客_linux编译参数](#)

28. 内核配置编译及Makefile?

[Linux内核配置、编译及Makefile简述 - CrazyCatJack - 博客园](#)

29.谈谈对Volatile关键字的理解?

[Volatile_不忘初心-CSDN博客](#)

30. framebuffer机制?

Linux抽象出FrameBuffer这个设备来供用户态进程实现直接写屏。Framebuffer机制模仿显卡的功能，将显卡硬件结构抽象掉，可以通过Framebuffer的读写直接对显存进行操作。用户可以将Framebuffer看成是显示内存的一个映像，通过mmap将其映射到进程地址空间之后，就可以直接进行读写操作，而写操作可以立即反应在屏幕上。这种操作是抽象的，统



kavin.zhu

关注

63



681



4



专栏目录



举报

1. spinlock与信号量的区别？

[linux 内核同步机制-自旋锁与信号量及其区别_xiaohuima_dong的专栏-CSDN博客](#)

[蜗窝科技 spin lock \(讲的非常不错\) - Red_Point - 博客园](#)

2. linux中的同步机制？

[linux 同步机制 死锁_steviewong的专栏-CSDN博客](#)

[linux同步机制_tong646591的专栏-CSDN博客_linux同步机制](#)

3. linux系统实现原子操作有哪些方法？

[Linux的原子操作与同步机制 - Florian - 博客园](#)

[linux内核原子操作的实现_vivi的技术博客-CSDN博客](#)

4. 自旋锁和信号量在互斥使用时需要注意哪些？在中断服务程序里面的互斥是使用自旋锁还是信号量？还是两者都能用？为什么（答案见1分析）？

答：使用自旋锁的进程不能睡眠,使用执行时间短的任务，使用信号量的进程可以睡眠，适合于执行时间较长的任务。中断服务例程中的互斥使用的是自旋锁，原因是在中断处理例程中，硬中断是关闭的，这样会丢失可能到来的中断。

5. 驱动里面为什么要有并发、互斥的控制？如何实现？讲个例子？

并发（concurrency）指的是多个执行单元同时、并行被执行，而并发的执行单元对共享资源（硬件资源和软件上的全局变量、静态变量等）的访问则很容易导致竞态（race conditions）。

解决竞态问题的途径是保证对共享资源的互斥访问，所谓互斥访问就是指一个执行单元在访问共享资源的时候，其他的执行单元都被禁止访问。

访问共享资源的代码区域被称为临界区，临界区需要以某种互斥机制加以保护，中断屏蔽，原子操作，自旋锁，和信号量都是linux设备驱动中可采用的互斥途径。

三. 中断相关：

可以先看一下五篇系列文章：<https://blog.csdn.net/droidphone/article/category/1118447>

这篇我收藏的文档详细的叙述了中断上半部及下半部的原理及注意点，如果对其不理解可以下载下来看看，由于CSDN最低没有0积分，那就最低的一个积分吧。下载地址：[linux设备驱动中断之上半部和下半部_中断上半部和下半部,linux中断上半部与下半部-Android文档类资源-CSDN下载](#)

1. linux中软中断的实现原理？

[Linux中断（interrupt）子系统之五：软件中断（softIRQ）_DroidPhone的专栏-CSDN博客](#)

[硬中断与软中断的区别_Linux编程_Linux公社-Linux系统门户网站](#)

2. linux中断响应的执行流程

[linux中断流程详解_yimu13的专栏-CSDN博客_linux中断处理流程](#)

3. linux中断实现机制、tasklet和workqueue的区别和底层实现的区别，为什么要区分中断上半部和中断下半部。

（中断上半部及下半部详细文档：[linux设备驱动中断之上半部和下半部_中断上半部和下半部,linux中断上半部与下半部-Android文档类资源-CSDN下载](#)）

tasklet和workqueue区别？

tasklet运行于中断上下文，不允许阻塞、休眠，而workqueue运行与进程上下文，可以休眠和阻塞。

为什么要区分上半部和下半部？

中断服务程序异步执行，可能会中断其他的重要代码，包括其他中断服务程序。因此，为了避免被中断的代码延迟太长的时间，中断服务程序需要尽快运行，而且执行的时间越短越好，所以中断程序只作必须的工作，其他工作推迟到以后处理。所以Linux把中断处理切为两个部分：上半部和下半部。上半部就是中断处理程序，它需要完成的工作越少越好，执行得越快越好，一旦接收到一个中断，它就立即开始执行。像对时间敏感、与硬件相关、要求保证不被其他中断打断的任务往往放在中断处理程序中执行；而剩下的与中断有相关性但是可以延后的任务，如对数据的操作处理，则推迟一点由下半部完成。下半部分延后执行且执行期间可以相应所有中断，这样可使系统处于中断屏蔽状态的时间尽可能的短，提高了系统



举报

中断的申请request_irq的正确位置：应该是在第一次打开、硬件被告知终端之前。

5. 中断注册函数和中断注销函数

[6.分析request_irq和free_irq函数如何注册注销中断\(详解\) - 诺谦 - 博客园](#)

6. 中断和轮询哪个效率高？怎样决定是采用中断方式还是采用轮询方式去实现驱动？

中断是CPU处于被动状态下来接受设备的信号，而轮询是CPU主动去查询该设备是否有请求。凡事都是两面性，所以，看效率不能简单的说那个效率高。如果是请求设备是一个频繁请求cpu的设备，或者有大量数据请求的网络设备，那么轮询的效率是比中断高。如果是一般设备，并且该设备请求cpu的频率比较底，则用中断效率要高一些。主要是看请求频率。

7. 写一个中断服务需要注意哪些？如果中断产生之后要做比较多的事情你是怎么做的？

第一：**中断处理例程应该尽量短**，把能放在后半段(tasklet，等待队列等)的任务尽量放在后半段。

写一个中断服务程序要注意快进快出，在中断服务程序里面尽量快速采集信息，包括硬件信息，然后退出中断，要做其它事情可以使用工作队列或者tasklet方式。也就是中断上半部和下半部。

第二：中断服务程序中不能有阻塞操作。应为中断期间是完全占用CPU的（即不存在内核调度），中断被阻塞住，其他进程将无法操作；

第三：中断服务程序注意返回值，要用操作系统定义的宏做为返回值，而不是自己定义的OK，FAIL之类的。

8. 驱动中操作物理绝对地址为什么要先ioremap？

因为内核没有办法直接访问物理内存地址，必须先**通过ioremap获得对应的虚拟地址**。

9. IRQ和FIQ有什么区别，在CPU里面是是怎么做的？

[FIQ和IRQ的区别及CPU实现_逐波与逆流-CSDN博客_fiq和irq的区别](#)

10. Linux软中断和工作队列的作用是什么？

[Linux内核中的软中断、tasklet和工作队列详解_godleading的专栏-CSDN博客_tasklet](#)

参考文章：[linux驱动工程面试必问知识点](#)

[linux驱动工程面试必问知识点_OpenWrt/WLAN/驱动/嵌入式开发总结-CSDN博客](#)

[Linux 驱动面试题总结_知秋一叶-CSDN博客_linux驱动开发面试题](#)

<https://blog.csdn.net/lhhero701/article/details/51171948>

Linux驱动面试题总结 04-04
各大公司30道Linux驱动面试题的总结，值得一过。Mark。搜集于网络

【Linux驱动开发100问】Linux驱动开发工程师在面试中常被问到的… 让技术没有门槛 660
Linux驱动开发专栏【Linux驱动开发100问】汇总

linux驱动面试题整理_hbcbgcx的博客 4-23
上半部分执行与硬件相关的处理要求快, 而有些驱动在中断处理程序中又需要完成大量工作,这构成矛盾, …

常见linux驱动面试题_KurdtCobain的博客 5-3
1)主设备号:主设备号标识设备对应的驱动程序。虽然现代的linux内核允许多个驱动程序共享主设备号, …

linux系统运维面试题大全（137道题） 最新发布 ma286388309的博客 1567
文章内容涵盖了linux系统运维几乎说有方面的知识，适合linux学习者或面试者阅读。

几个Linux驱动面试题 目 mcuwxlong的专栏 1592
Linux驱动面试题 目

Linux驱动面试题 目及解答汇总_艾特号的博客 4-27
这几天面试几个想做安卓Linux驱动的,总体感觉上驱动基础还是比较薄弱,大部分情况是虽然做过驱动, …

linux驱动基础面试题(一)_初级驱动开发面试题_目_TOWWWW的博客-CSDN博 ... 4-17
Linux设备中字符设备与块设备有什么主要的区别? 字符设备是可以向字节流(类似文件)一样被访问的设…

Linux 驱动常见面试及答案 yaolixiao001的博客 1万+
简单介绍：好久没有面试了，面试之前还是要准备一下的，尤其是对工程师来说，很纠结，不管怎么说…



举报

- 48 个 Linux 面试题和答案 热门推荐

taoxicun的博客 2万+

你在准备 Linux 面试吗？我们准备了一些常见的 Linux 面试题及其答案。如果您是初学者（具有一定…
- Linux驱动开发（面试题常见问题整理）

weixin_49303682的博客 1990

1.Linux设备驱动的分类 -> 字符设备 -> 块设备 -> 网络设备字符设备指那些必须以串行顺序依次进行访…
- 2万字系统总结，带你实现 Linux 命令自由?还不赶紧进来学习

Java_Pluto的博客 1046

2万字系统总结，带你实现 Linux 命令自由？##前言 Linux 的学习对于一个程序员的重要性是不言而喻…
- Linux驱动开发面试题整合

qq_40904479的博客 2157

Linux 驱动常见面试及答案八个常见的Linux驱动面试题 Linux驱动面试题总结嵌入式校招面试题 BSP开…
- linux驱动工程师面试题整理

07-05

整理大部分的linux驱动工程师面试题，不算很全，但是还是比较实用！
- linux驱动开发面试题及答案

12-15

linux驱动开发面试题及答案 是一些比较常见的linux面试题 及其对应答案
- linux驱动工程师面试题整理.docx

07-06

//
- linux驱动工程师面试题整理.pdf

07-06

//
- linux驱动面试笔试题.7z

06-13

linux驱动面试笔试题.7z
- linux驱动面试题

monkea123的博客 3962

1、字符型驱动设备你是怎么创建设备文件的，就是/dev/下面的设备文件，供上层应用程序打开使用的…
- 《linux驱动工程师面试题整理》.doc

01-17

1字符型驱动设备你是怎么创建设备文件的就是/dev/下面的设备文件供上层应用程序打开使用的文件 …
- 几道简单的Linux驱动相关面试题，你看你会几题？

一口Linux的专栏 3839

中断：是指CPU在执行程序的过程中，出现了某些突发事件时CPU必须暂停执行当前的程序，转去处理…
- linux驱动工程师面试题

03-16

您好，以下是针对Linux驱动工程师的面试题： 1. 请介绍一下Linux驱动的基本架构和工作原理。 2. 如…

“相关推荐”对你有帮助么？

-  非常没帮助
-  没帮助
-  一般
-  有帮助
-  非常有帮助

关于我们

招贤纳士

商务合作

寻求报道

400-660-0108

kefu@csdn.net

在线客服

工作时间 8:30-22:00

公安备案号11010502030143 京ICP备19004658号 京网文〔2020〕1039-165号 经营性网站备案信息

北京互联网违法和不良信息举报中心 家长监护 网络110报警服务 中国互联网举报中心 Chrome商店下载 账号管理规范

版权与免责声明 版权申诉 出版物许可证 营业执照 ©1999-2023北京创新乐知网络技术有限公司

kavin.zhu VIP

码龄7年 暂无认证

95 原创

3万+ 周排名

1万+ 总排名

98万+ 访问

 等级

6250 积分

567 粉丝

837 获赞

80 评论

4457 收藏











私信

关注

搜博主文章



热门文章

cmake快速入门 222544


linux系统组成及结构 45607

i2c-tools使用及调试 45498

linux驱动最新面试题（面试题整理，含答案） 39896



举报

kavin.zhu

关注

63

681

4

专栏目录



项目调试笔记

付费



XR功能模块

1篇



Native

5篇



安卓系统相关

17篇



Android framework

18篇





Android UI

20篇

最新评论

- 安卓startService：让你彻底理解startSe…
ccccccccc2: 太深奥了我超
- Android硬件服务框架实例之Vibrator(驱…
murphykwu: 整个流程从底层到框架层，再到上层，全部都非常详细。好奇硬件如何…
cmake快速入门
- qq_58817647: 建立一个build文件夹，上级放CMakeLists.txt文件，执行cmke .. | …
- 安卓系统添加dropbear实例
LJ-SEU: 为什么我用网上所有的dropbear源码 mm编译的时候都会报 eng 被废弃 …
- i2c-tools使用及调试
m0_58638026: 没问题，我自己是在zynq上做的，跟着做完全没问题，感谢楼主

您愿意向朋友推荐“博客详情页”吗？



强烈不推荐 不推荐 一般般 推荐 强烈推荐


最新文章


- 1.Android Native Sensor(C++)实例
- 安卓RecyclerView+CardView使用总结
- Android WatchDog

2022年	1篇	2021年	6篇
2020年	13篇	2019年	14篇
2018年	71篇	2017年	32篇
2016年	2篇		


目录

- 一. 基础题：
- 二. 同步相关：
- 三. 中断相关：








举报


kavin.zhu


关注


63



681



4



专栏目录