

Linux/Android开发记录

学习、记录、分享Linux/Android开发技术

目录视图

摘要视图

RSS 订阅

个人资料



liuhaoyutz

访问: 80624次

积分: 1673分

排名: 第7877名

原创: 83篇

转载: 0篇

译文: 0篇

评论: 59条

博客声明

本博客文章均为原创，欢迎转载交流。转载请注明出处，禁止用于商业目的。

博客专栏



Android应用开发学习笔记本

文章: 30篇

阅读: 17067



LDD3源码分析

文章: 17篇

阅读: 29970

文章分类

LDD3源码分析 (18)

ADC驱动 (1)

触摸屏驱动 (1)

LCD驱动 (1)

Linux设备模型 (8)

USB驱动 (0)

Android架构分析 (12)

Cocos2d-x (1)

C陷阱与缺陷 (3)

Android应用开发 (30)

Linux设备驱动程序架构分析 (8)

有奖征资源，博文分享有内涵

5月推荐博文汇总

大数据读书汇--获奖名单公布

2014 CSDN博文大赛

LDD3源码分析之vmalloc

分类: LDD3源码分析

2012-03-31 21:08

1468人阅读

评论(0)

收藏

举报

makefile

ubuntu

linux

任务

测试

作者: 刘昊昱

博客: <http://blog.csdn.net/liuhaoyutz>

编译环境: Ubuntu 10.10

内核版本: 2.6.32-38-generic-pae

LDD3源码路径: examples/scullv

一、scullv编译

本文分析LDD3第8章中与vmalloc函数相关代码, 对应源码是examples/scullv目录下的相关文件。

这里首先说明一下, scullv的代码有两部分, 一是main.c, 另一部分是mmap.c, 其中mmap.c主要实现了内存映射相关函数。因为本文主要分析vmalloc的用法, 并且mmap.c涉及的内存映射在新的内核中有了很大变化。在本文中, 我们不讨论mmap.c, 并且为避免编译时出现的错误, 编译时也不编译mmap.c。

要想在编译scullv模块时不编译mmap.c, 只要做如下修改:

将Makefile第18行改为: scullp-objs := main.o

将main.c的第475行屏蔽掉: `//.mmap = scullp_mmap,`

这样就不会编译mmap.c了。

即使不编译mmap.c, scullv在2.6.32-38-generic-pae内核下编译也会遇到很多问题, 但是遇到的问题和解决方法和我在《LDD3源码分析之slab高速缓存》中讨论的是一样的, 这里不详细说明了。

二、vmalloc及其相关函数

vmalloc函数分配的内存存在虚拟地址空间是连续的, 在物理地址上可能不连续。

我们在这里介绍vmalloc函数的原因是, 它是Linux内存分配机制的基础。但是, 在大多数情况下, 我们并不鼓励使用vmalloc, 通过vmalloc函数获得的内存使用效率不高, 而且在某些体系架构上, 用于vmalloc的地址空间总量相对较小。只要可能, 应该直接和单个页面打交道, 而不是使用vmalloc。

vmalloc函数原型及相关函数如下所示:

[cpp]

01. void \*vmalloc(unsigned long size);

02. void vfree(void \* addr);

03. void \*ioremap(unsigned long offset, unsigned long size);

<http://blog.csdn.net/liuhaoyutz/article/details/7417174>

1/3

最新评论

- LDD3源码分析之内存映射  
wzw88486969:  
@fjlhlonng:unsigned long offset  
= vma->vm\_pgoff <v...
- Linux设备驱动程序架构分析之I2  
teamos:看了你的i2c的几篇文  
章，真是受益匪浅，虽然让自己  
写还是ie不出来。非常感谢
- LDD3源码分析之块设备驱动程序  
elecfan2011: 感谢楼主的精彩讲  
解，受益匪浅啊！
- LDD3源码分析之slab高速缓存  
donghuwuwei: 省了不少修改  
的时间，真是太好了
- LDD3源码分析之时间与延迟操作  
donghuwuwei: jitc代码需要加上  
一个头文件。
- LDD3源码分析之slab高速缓存  
捧灰: 今天学到这里了，可是为什  
么我没有修改源码一遍就通过了  
额。。。内核版本是2.6.18-  
53.el5-x...
- LDD3源码分析之字符设备驱动程  
捧灰: 参照楼主的博客在自学~谢  
谢楼主！
- LDD3源码分析之调试技术  
fantasyhujian: 分析的很清楚，  
赞一个！
- LDD3源码分析之字符设备驱动程  
fantasyhujian: 有时间再好好读  
读，真的分析的不错！
- LDD3源码分析之hello.c与Makef  
fantasyhujian: 写的很详细，对  
初学者很有帮助！！

阅读排行

- LDD3源码分析之字符设: (3143)
- LDD3源码分析之hello.c: (2701)
- S3C2410驱动分析之LCI (2527)
- Linux设备模型分析之kse (2435)
- LDD3源码分析之内存映! (2336)
- LDD3源码分析之与硬件! (2333)
- Android架构分析之Andrc (2093)
- LDD3源码分析之时间与; (1987)
- LDD3源码分析之poll分析 (1972)
- S3C2410驱动分析之AD ( 1948)

评论排行

- LDD3源码分析之字符设: (12)
- S3C2410驱动分析之触摸 (7)
- LDD3源码分析之内存映! (5)
- LDD3源码分析之hello.c: (4)
- Linux设备模型分析之kot (4)
- LDD3源码分析之slab高; (4)
- S3C2410驱动分析之LCI (3)
- LDD3源码分析之阻塞型I (3)
- LDD3源码分析之时间与; (3)
- LDD3源码分析之poll分析 (2)

文章存档

- 2014年06月 (1)
- 2014年05月 (4)
- 2014年04月 (1)

04. `void iounmap(void * addr);`

需要强调的是，由kmalloc和\_\_get\_free\_pages返回的内存地址也是虚拟地址，其实际值仍然需要由MMU处理才能转换为物理内存地址。vmalloc在如何使用硬件上没有区别，区别在于内核如何执行分配任务上。

kmalloc和\_\_get\_free\_pages返回的虚拟地址范围和物理内存是一一对应的，可能会有基于常量PAGE\_OFFSET的一个偏移。这两个函数不需要为该地址段修改页表。但是，vmalloc和ioremap使用的地址范围是完全虚拟的，每次分配都要通过对页表的适当设置来建立(虚拟)内存区域。

与vmalloc一样，ioremap也建立新的页表，但是和vmalloc不同的是，ioremap并不实际分配内存。ioremap的返回值是一个特殊的虚拟地址，可以用来访问指定的物理内存区域，这个虚拟地址最后要调用iounmap来释放掉。

三、scullv分析

scullv模块的代码和上一篇文章分析的scullp基本上一样，和scull也很类似。这里我们只分析最重要的代码。

scullv中分配内存的代码在scullv\_write函数中：

```
[cpp]
01. 244 /* Allocate a quantum using virtual addresses */
02. 245 if (!dptr->data[s_pos]) {
03. 246     dptr->data[s_pos] = (void *)vmalloc(PAGE_SIZE << dptr->order);
04. 247     if (!dptr->data[s_pos])
05. 248         goto nomem;
06. 249     memset(dptr->data[s_pos], 0, PAGE_SIZE << dptr->order);
07. 250 }
```

scullv中释放内存的代码在scullv\_trim函数中：

```
[cpp]
01. 493 /* Release the quantum-set */
02. 494 for (i = 0; i < qset; i++)
03. 495     if (dptr->data[i])
04. 496         vfree(dptr->data[i]);
```

测试vmalloc模块的过程如下图所示：

2014年01月 (1)

2013年12月 (6)

展开

文章搜索

推荐文章

下一篇 LDD3源码分析之与硬件通信&中断处理

顶

0

踩

0

主题推荐

源码    内存分配    内核    quantum    makefile

猜你在找

- Linux内核地址空间的布局及实现代码
- linux-3.2.36内核启动2-setup\_arch中的内存初始化
- 应届生去大公司与中小公司利弊之个人见解
- like,unlikely宏和GCC内建函数\_\_builtin\_expect()
- [Linux内存]linux内存学习（三）——内存管理基础
- 线程之间的同步与互斥
- 伙伴系统分配器 - \_\_alloc\_pages
- 如何在windows下面编译u-boot（原发于：2012-07-24
- linux中同步例子(完成量completion)
- u-boot编译笔记

免费学习IT4个月,月薪12000

中国[官方授权]IT培训与就业示范基地,学成后名企直接招聘,月薪12000起!

查看评论

暂无评论

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题

Java

VPN

Android

iOS

ERP

IE10

Eclipse

CRM

JavaScript

Ubuntu

NFC

WAP

jQuery

数据库

BI

HTML5

Spring

Apache

Hadoop

.NET

API

HTML

SDK

IIS

Fedora

XML

LBS

Unity

Splashtop

UML

components

Windows Mobile

Rails

QEMU

KDE

Cassandra

CloudStack

FTC

coremail

OPhone

CouchBase

云计算

iOS6

Rackspace

Web App

SpringSide

Maemo

Compuware

大数据

aptech

Perl

Tornado

Ruby

Hibernate

ThinkPHP

Spark

HBase

Pure

Solr

Angular

Cloud Foundry

Redis

Scala

Django

Bootstrap