

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: 80620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
文章: 30篇
阅读: 17067



LDD3源码分析
文章: 17篇
阅读: 29965

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llseek分析

分类: LDD3源码分析

2012-03-28 14:36

1923人阅读

评论(1)

收藏

举报

struct

file

测试

ubuntu

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/scull/main.c

本文分析LDD3第6章的llseek函数。

一、用户空间的lseek函数

要理解驱动中llseek函数的实现，必须先清楚对应的用户空间中lseek函数的用法，lseek函数函数原型如下：

```
[cpp]01. off_t lseek(int fd, off_t offset, int whence);
```

第一个参数fd是要操作的文件描述符。
第二个参数指定文件操作指针的偏移量。注意，文件的读和写使用的是同一个文件操作指针。
第三个参数指定移动文件操作指针的参考点。这个参数通常取值为以下宏：

SEEK_SET：表示相对文件起始位置。

SEEK_CUR：表示相对文件操作指针当前位置。

SEEK_END：表示相对文件结束位置。

下面先来看一个用户空间测试程序llseek_test.c的实现，这个程序用来测试scull的定位功能，其代码如下：

```
[cpp]01. 1#include <stdio.h>
02. 2#include <unistd.h>
03. 3#include <fcntl.h>
04. 4#include <string.h>
05. 5#include <sys/types.h>
06. 6#include <sys/stat.h>
07. 7
08. 8#define BUF_SIZE 50
09. 9#define DEVICE_FILE "/dev/scull"
```

<http://blog.csdn.net/liuhaoyutz/article/details/7402857>

1/4

最新评论

- [LDD3源码分析之内存映射](#)
wzw88486969:
@jhlhlonng:unsigned long offset = vma->vm_pgoff <v...
- [Linux设备驱动程序架构分析之l2team](#)
amos:看了你的l2c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](#)
elecfa2011: 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: 省去了不少修改的时间，真是太好了
- [LDD3源码分析之时间与延迟操作](#)
donghuwuwei: jit.c代码需要加上一个头文件。
- [LDD3源码分析之slab高速缓存](#)
棒灰: 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序](#)
棒灰: 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序](#)
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Makefile](#)
fantasyhujian: 写的很详细，对初学者很有帮助！！

阅读排行

- [LDD3源码分析之字符设备](#) (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源码分析之内存映射](#) (2336)
- [LDD3源码分析之与硬件](#) (2333)
- [Android架构分析之Andrc](#) (2093)
- [LDD3源码分析之时间与延迟](#) (1987)
- [LDD3源码分析之poll分析](#) (1972)
- [S3C2410驱动分析之AD](#) (1948)

评论排行

- [LDD3源码分析之字符设备](#) (12)
- [S3C2410驱动分析之触摸](#) (7)
- [LDD3源码分析之内存映射](#) (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高速](#) (4)
- [S3C2410驱动分析之LCI](#) (3)
- [LDD3源码分析之阻塞型I](#) (3)
- [LDD3源码分析之时间与延迟](#) (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)

```
10. 10
11. 11int main(int argc, char *argv[])
12. 12{
13. 13     int fd;
14. 14     int num;
15. 15     char buf[BUF_SIZE];
16. 16
17. 17     fd = open(DEVICE_FILE, O_RDWR);
18. 18     if(fd < 0)
19. 19     {
20. 20         printf("open scull error!\n");
21. 21         return -1;
22. 22     }
23. 23
24. 24     memset(buf, 0, BUF_SIZE);
25. 25     num = read(fd, buf, BUF_SIZE);
26. 26     buf[num] = 0;
27. 27     printf("%s\n", buf);
28. 28
29. 29     lseek(fd, 2, SEEK_SET);
30. 30     write(fd, "aa", 2);
31. 31     num = read(fd, buf, BUF_SIZE);
32. 32     buf[num] = 0;
33. 33     printf("%s\n", buf);
34. 34
35. 35     lseek(fd, 2, SEEK_SET);
36. 36     num = read(fd, buf, BUF_SIZE);
37. 37     buf[num] = 0;
38. 38     printf("%s\n", buf);
39. 39
40. 40     lseek(fd, 0, SEEK_SET);
41. 41     lseek(fd, 2, SEEK_CUR);
42. 42     num = read(fd, buf, BUF_SIZE);
43. 43     buf[num] = 0;
44. 44     printf("%s\n", buf);
45. 45
46. 46     lseek(fd, 0, SEEK_SET);
47. 47     lseek(fd, 0, SEEK_END);
48. 48     memset(buf, 0, BUF_SIZE);
49. 49     printf("read return value is %d.\n", read(fd, buf, BUF_SIZE));
50. 50
51. 51     return 0;
52. 52}
```

这个程序很简单，主要关注一下lseek函数是怎样移动文件操作指针的。

第29行，使用SEEK_SET宏，将文件操作指针移动到文件起始位置加上2个字节处。

第30行，写入两个字符’ a’ 。

第41行，使用SEEK_CUR宏，将文件操作指针移动到文件操作指针当前位置加上2个字节处。

第47行，使用SEEK_END宏，将文件操作指针移动到文件结束处。

第49行，打印read的返回值，当文件操作指针在文件结束处时，read返回0。

下图是使用llseek_test测试scull设备的定位功能的过程：

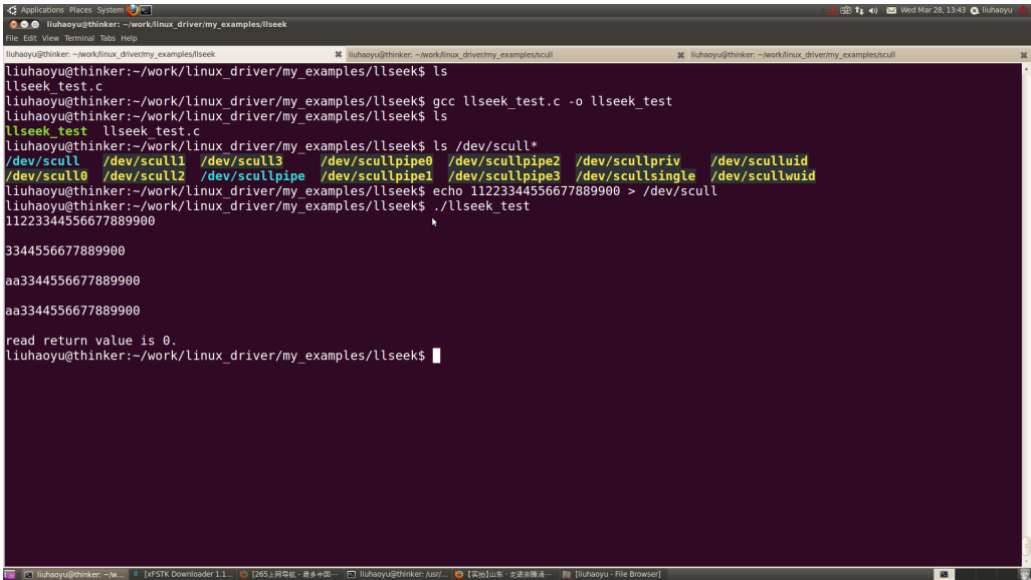
2014年01月 (1)

2013年12月 (6)

展开

文章搜索

推荐文章



这里需要说明的一点是，从上面的输出信息可以看出，对文件的read和write操作使用的是同一个文件操作指针。

二、驱动程序中llseek函数的实现

用户空间的lseek函数的定位功能在驱动程序中是由llseek函数实现的。注意，要完成对文件的定位操作，还需要read、write函数的配合，读写完成后必须更新文件操作指针位置。

即使驱动程序中没有实现llseek函数，有某些情况下，设备也是可以完成定位操作的，内核通过修改filp->f_pos来执行定位，filp->f_pos是文件的当前读写位置。但是，如果定位操作需要涉及设备的物理操作，就必须实现llseek函数了。scull设备的llseek函数代码如下：

```
[cpp]
01. 523loff_t scull_llseek(struct file *filp, loff_t off, int whence)
02. {
03.     525     struct scull_dev *dev = filp->private_data;
04.     526     loff_t newpos;
05.     527
06.     528     switch(whence) {
07.         529         case 0: /* SEEK_SET */
08.             530             newpos = off;
09.             531             break;
10.         532
11.         533         case 1: /* SEEK_CUR */
12.             534             newpos = filp->f_pos + off;
13.             535             break;
14.         536
15.         537         case 2: /* SEEK_END */
16.             538             newpos = dev->size + off;
17.             539             break;
18.         540
19.         541         default: /* can't happen */
20.             542             return -EINVAL;
21.         543     }
22.     544     if (newpos < 0) return -EINVAL;
23.     545     filp->f_pos = newpos;
24.     546     return newpos;
25. }
```

这里唯一与设备相关的操作就是第538行，取得设备文件的大小。同时，我们在前面的文章中分析过，scull的read和write函数读写文件后，总是更新文件操作指针的位置，定位功能需要llseek与read、write的配合。

对于某些设备文件来说，定位功能是没有意义的，例如键盘。对于这些设备，我们不能简单地不实现llseek函数，因为默认方法是允许通过filp->f_pos定位的。我们应该在我们的open函数中调用nonseekable_open，通知内核设备不支持llseek。该函数的函数原型如下：

[cpp]

```
01. int nonseekable_open(struct inode *inode; struct file *filp);
```

另外，为完整起见，我们还应该将file_operations结构中的llseek方法设置为特殊的辅助函数no_llseek。

更多 0

上一篇 LDD3源码分析之异步通知
下一篇 LDD3源码分析之访问控制

顶 1
踩 0

主题推荐 源码 移动 测试 内核 ubuntu

猜你在找

- TI-AM335X GPMC 7个config寄存器重点
- 关于driver_register函数干了什么
- C++第9周(春)项目3 - 分数类
- camera启动调用流程
- Git使用
- MTD 设备驱动 和 NAND Flash 驱动程序分析。
- 如何在windows下面编译u-boot（原发于：2012-07-24
- android 从驱动到应用（二）
- linux内核模块的强制删除~结束rmmod这类disk sleep进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

1楼 雁子依然 2013-06-20 11:33发表



妈妈说，好文章一定要顶

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
- Bootsrap