**Linux Kernel Source Notes**

（ZL@NHAK Kernel V4.15.0）

**目录**

[1 概述 4](#_Toc519095015)

[1.1 背景与出发点 4](#_Toc519095016)

[1.2 优秀借鉴资源 4](#_Toc519095017)

[2 Linux系统启动(Init) 5](#_Toc519095018)

[2.1 Start\_kernel之前 5](#_Toc519095019)

[2.2 Start\_kernel分析 6](#_Toc519095020)

[2.3 主要结构体 6](#_Toc519095021)

[3 进程管理(Task) 6](#_Toc519095022)

[3.1 概述 6](#_Toc519095023)

[3.2 主要结构体 6](#_Toc519095024)

[3.3 详细分析 7](#_Toc519095025)

[4 内存管理(MM) 7](#_Toc519095026)

[4.1 概述 7](#_Toc519095027)

[4.2 主要结构体 7](#_Toc519095028)

[4.3 详细分析 8](#_Toc519095029)

[5 文件系统(FS) 8](#_Toc519095030)

[5.1 概述 8](#_Toc519095031)

[5.2 主要结构体 9](#_Toc519095032)

[5.3 详细分析 10](#_Toc519095033)

[6 网络通信(Net) 12](#_Toc519095034)

[6.1 概述 12](#_Toc519095035)

[6.2 主要结构体 12](#_Toc519095036)

[6.3 详细分析 12](#_Toc519095037)

[7 Linux定制/优化/定位 12](#_Toc519095038)

[7.1 GNUmake/GCC 12](#_Toc519095039)

[7.2 Kgdb/Oops 12](#_Toc519095040)

[7.3 Kprobe/Ftrace/ELF 12](#_Toc519095041)

[7.4 Perf 12](#_Toc519095042)

[8 总结与展望 12](#_Toc519095043)

# 概述

## 背景与出发点

Linux系统博大精深，但源码之下、了无秘密，善用Makefile和Kconfig探索Kernel源。

## 优秀借鉴资源

优秀书籍：

《Linux设备驱动》[(第三版](http://www.baidu.com/link?url=n9DyC0aAF6aBI5fFzEdztOUb2r1yOOi1dyn9FSobWLvwYr4EW0UG2C59oA2L4zZc))

《Linux内核情景分析》(**毛德操**)

《奔跑吧Linux内核》(张天飞)

《Linux内核修炼之道》(任桥伟)

《深入理解Linux内核》(第三版)

《Debug Hacks 深入调试的技术和工具》

《Linux二进制分析》(ELF)

网上优秀资源：

最经典的Linux内核学习方法论<https://blog.csdn.net/chinaclock/article/details/48293289>

Linux.Kernel.Crash.Book.Everything.you.need.to.know.pdf

Intel64 and IA-32 Architectures Software Developer's Manual.pdf (www.intel.com)

跟我一起写Makefile(陈皓)

疯狂内核<https://blog.csdn.net/yunsongice/article/details/5470283>

Linux进程管理与调度<https://blog.csdn.net/gatieme/article/details/51456569>

Linux内存管理<http://www.cnblogs.com/arnoldlu/p/8051674.html>

Linux内存管理<https://blog.csdn.net/bullbat/article/category/1055017/2>

Linux Kernel Exploration：<http://ilinuxkernel.com>

Linux x86 Memory Address Mapping .pdf

Linux x86\_64 Memory Address Mapping.pdf

LinuxFtrace分析<http://www.cnblogs.com/arnoldlu/p/7211249.html>

Linux内核的测试和调试<https://linux.cn/article-3593-1-rel.html>

LinuxKernel移植 <http://blog.csdn.net/skyflying2012/article/details/43281565>

阿里**R&D** https://102.alibaba.com/downloadFile.do?file=1523962960197/AliTech101\_RD.pdf



# Linux系统启动(Init)

## Start\_kernel之前

基于x86体系结构

处理器芯片手册需要重点关注的几点有：   
（1）地址空间管理，包括上电取指地址，MMU如何配置使用等   
（2）寄存器定义，包括通用寄存器，特殊功能寄存器   
（3）中断异常的定义，异常向量表如何定义和使用   
（4）cache管理   
（5）处理器核reset后如何初始化，处理器核手册会给出一些标准的步骤   
（6）处理器核特定资源介绍

在分析这段代码之前，我们必须先找到汇编代码的入口位置，也就是Bootloader启动linux时所跳转到的地址。内核一般被压缩后保存到FLASH上的，在Bootloader启动Linux之前必须先解压缩内核，关于内核解压缩部分请参照arch/arm/boot/compressed这里面的代码，这里就不细将了。

   在进入Linux汇编代码之前，Bootloader的执行影响CPU的状态，其状态一般为：

        1. CPU必须处于SVC(supervisor)模式,并且IRQ和FIQ中断都是禁止的;  
      2. MMU(内存管理单元)必须是关闭的, 此时虚拟地址对物理地址;  
      3. 数据cache(Data cache)必须是关闭的  
      4. 指令cache(Instruction cache)可以是打开的,也可以是关闭的,这个没有强制要求;  
      5. CPU 通用寄存器0 (r0)必须是 0;  
      6. CPU 通用寄存器1 (r1)必须是 ARM Linux machine type（在后面会将到machine type）

        7. CPU 通用寄存器2 (r2) 必须是 kernel parameter list 的物理地址(parameter list 是由boot loader传递给kernel,用来描述设备信息属性的列表）

   Linux汇编的入口地址请看文件linux-4.9.86/arch/x86/kernel/head\_64.S，我们现在来看看文件里面的代码吧。

## Start\_kernel分析

## 主要结构体

# 进程管理(Task)

## 概述

。

## 主要结构体

主要结构体关系俯瞰

主要结构体详细分析

[include/linux/sched.h]

**struct task\_struct**

**{**

}

## 详细分析

。

# 内存管理(MM)

## 概述

。

## 主要结构体

主要结构体关系俯瞰

主要结构体详细分析

[include/linux/mmzone.h]

**type struct pglist\_data**

**{**

**}pg\_data\_t**

全局变量**pgdat\_list**

[include/linux/mmzone.h]

**struct zone**

**{**

**}**

[include/linux/mm\_types.h]

**struct page**

**{**

**}**

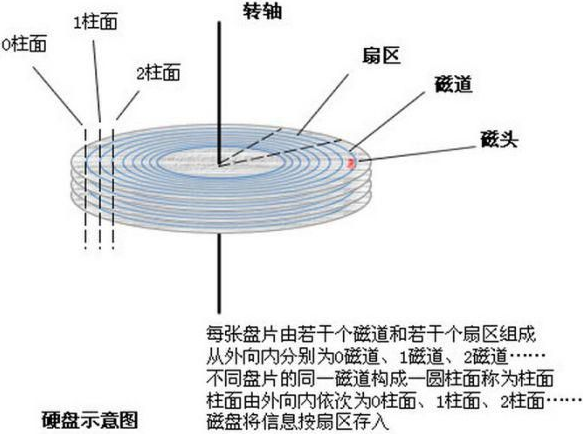
## 详细分析

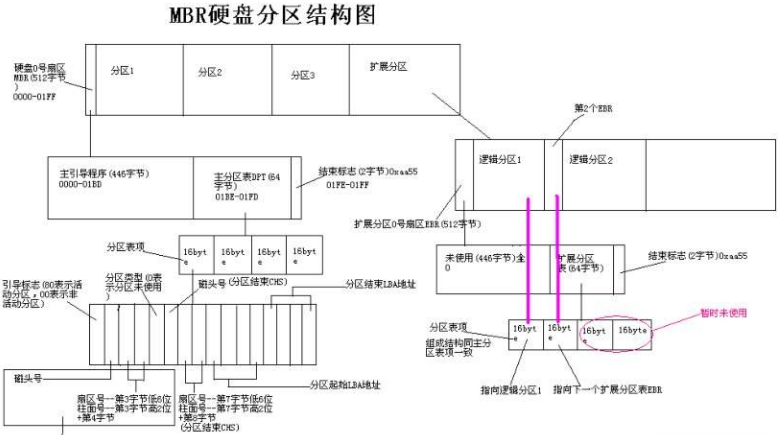
。

# 文件系统(FS)

## 概述

机械磁盘结构示意图





## 主要结构体

主要结构体关系俯瞰

主要结构体详细分析

**struct file\_system\_type**

**{**

}

[fs/filesystems.c: file\_systems]

[fs/ext4/super.c: ext4\_fs\_type]

**struct super\_block**

**{**

**}**

存储一个已安装的文件系统的控制信息，代表一个已安装的文件系统；每次一个实际的文件系统被安装时， 内核会从磁盘的特定位置读取一些控制信息来填充内存中的超级块对象。

**struct vfsmount**

**{**

}

**struct inode**

**{**

}

**struct dentry**

**{**

}

**struct file**

**{**

}

**struct file\_operations**

**{**

}

[fs/ext4/file.c: ext4\_file\_operations]

## 详细分析

**mount**分析

[do\_mount()->do\_new\_mount()->get\_fs\_type()]

- >vfs\_kern\_mount()

mount\_fs()

ext4\_mount()

mount\_bdev()

sget()

sget\_userns()]

**open**分析

[do\_sys\_open()->get\_unused\_fd\_flags()]

->do\_filp\_open()

**read**分析

[vfs\_read()->]

**write**分析

**mmap**分析

**lseek**分析

**close**分析

# 网络通信(Net)

## 概述

。

## 主要结构体

。

## 详细分析

。

# Linux定制/优化/定位

## GNUmake/GCC

。

## Kgdb/Oops

。

## Kprobe/Ftrace/ELF

。

## Perf

。

# 总结与展望

Linux系统博大精深，GNU/Linux生态健壮，扎稳根基，收放自如。