



















linux 3.6 启动源码分析(六) do_basic_setup

原创 2013年12月16日 16:20:06

1685

在内核init线程中调用了do_basic_setup,这个函数也做了很多内核和驱动的初始化工作

```
[cpp]
    /*好了,设备现在已经初始化完成。 但是还没有一个设备被初始化过,
1.
    但是 CPU 的子系统已经启动并运行,
     且内存和处理器管理系统已经在工作了。
4.
     现在我们终于可以开始做一些实际的工作了...
5.
6.
    static void __init do_basic_setup(void)
7.
8.
       cpuset_init_smp();//针对SMP系统,初始化内核control group的cpuset子系统。如果非SMP,此函数为空。
9.
       /*创建一个单线程工作队列khelper。运行的系统中只有一个,主要作用是指定用户空间的程序路径和环境变量,最终运行
    指定的user space的程序,属于关键线程,不能关闭*/
       usermodehelper_init();
10.
11.
12.
       shmem_init();
       driver_init();//初始化驱动模型中的各子系统,可见的现象是在/sys中出现的目录和文件
13.
       init_irq_proc();//在proc文件系统中创建irq目录,并在其中初始化系统中所有中断对应的目录。
14.
15.
       do_ctors();//调用链接到内核中的所有构造函数,也就是链接进.ctors段中的所有函数。
16.
       usermodehelper_enable();
17.
       do_initcalls();//调用所有编译内核的驱动模块中的初始化函数。
18. }
```

上面的函数调用了driver_init函数,作用是驱动模型子系统的初始化,对于内核驱动工程师来说比较重要, 详解如下:

```
[cpp]
     void __init driver_init(void)
1.
2.
3.
        /* These are the core pieces */
        devtmpfs_init();//初始化devtmpfs文件系统,驱动核心设备将在这个文件系统中添加它们的设备节点。
 4.
        /*初始化驱动模型中的部分子系统和kobject:
 5.
 6.
        devices
 7.
        dev
 8.
        dev/block
9.
        dev/char
10.
        */
11.
        devices_init();
12.
        buses_init();//初始化驱动模型中的bus子系统
13.
        classes_init();//1.初始化驱动模型中的class子系统
14.
        firmware_init();//1.初始化驱动模型中的firmware子系统
        hypervisor_init();//1.初始化驱动模型中的hypervisor子系统
15.
16.
17.
        /* These are also core pieces, but must come after the
18.
         * core core pieces.
         这些也是核心部件,但是必须在以上核心中的核心部件之后调用。
19.
20.
         */
        platform_bus_init();//1.初始化驱动模型中的bus/platform子系统
21.
22.
        cpu_dev_init();//1. 初始化驱动模型中的devices/system/cpu子系统
         memory_dev_init();//初始化驱动模型中的devices/system/memory子系统
24. }
```

而另外一个很主要的函数do_initcalls()调用所有编译内核的驱动模块中的初始化函数。其中按照各个内核模块初始化函数所自定义的启动级别(1~7),按顺序调用器初始化函数。对于同一级别的初始化函数,安装编译是链接的顺序调用,也就是和内核Makefile的编写有关。



严禁讨论涉及中国之军/政相关话题,违者会被禁言、封号!

linux内核的子系统(或者说功能模块的)初始化

内核启动过程中需要完成各个部分的初始化,比如中端、页面管 goodluckwhh 2013年10月16日 22:42 🕮 3223 理、slab分配器、任务调度器、网络、PCI设备等等的初始化,这

些初始化大概可以分为两种: 一种是关键的,必须完成的而且必须以特定的顺序来完成...

devtmpfs文件系统创建设备节点

● u010713604 2014年02月12日 17:44 □ 1960

一、devtmpfs概述 1.devtmpfs 的功用是在 Linux 核心 启动早期建立一个初步的 /dev , 令一般启动程序不用等待 udev , 缩 短 GNU/Linux 的开机时间。 2.重要解...

一个好的前端年薪会有多少?

你认为做前端想拿到高年薪难吗?









他的最新文章	更多文章
Linux设备模型(四)class	
Linux设备模型 (三) platform	
Linux设备模型(二)上层容器	
linux 设备模型(一)对象层	
Linux中断子系统-中断接口	

文章分类	
linux开发	2篇
linux 驱动学习	3篇
linux源码学习	14篇
文章存档	

文章存档		
2014年1月		2篇
2013年12月		12篇
2013年11月		1篇
2013年10月		4篇
2013年9月		9篇
2012年2月		1篇
	展开~	

他的热门文章

linux 3.6 启动源码分析(五) kernel_init进程

□ 3057

linux 3.6 启动源码分析(二) start_kernel 2909

linux 3.6 启动源码分析(七) do_initcalls 2454

linux 3.6 启动源码分析(一) 2078

linux 3.6 启动源码分析(三) setup_arch

☐ 1893

嵌入式linux 运行期间升级u-boot, kern el和文件系统 1792

linux 3.6 启动源码分析(六) do_basic_set up

1681

linux下读写u-boot环境变量 ☐ 1354

Linux中断子系统-中断初始化

1329

linux 3.6 启动源码分析(四) rest_init 1307 linux 3.6 启动源码分析(六) do_basic_setup

但是还没有一个设备被初始化过, 但是 CPU 的子系统已经启动...

qing_ping 2013年12月16日 16:20 🕮 1685









概述 BootROM Bootloader 引导程序 Linux 内核 init 进程 关键路径 作 dearsq 2016年12月14日 21:50 🕮 1266 用细节 Zygote 创建与启动应用 Zygote 是什么 Zygote 创建 启动应用 引导结束...

在内核init线程中调用了do_basic_setup,这个函数也做了很多内核和驱动的初始化工作/*好了,设备现在已经初始化完成。

码农不会英语怎么行?英语文档都看不懂!

软件工程出身的英语老师,教你用数学公式读懂天下英文→



linux 3.6 启动源码分析(一)

■ qing_ping 2013年12月16日 13:02 □ 2080

作为需要和硬件打交道的工程师来说,比较关注的是驱动和CPU初始化这一块。所以我沿着启动的路线,重点学习一下和硬 件相关的代码。就从linux解压的入口说起。学习阶段,基本是参考大神文章http://bl...

linux 3.6 启动源码分析(二) start_kernel

(qing ping 2013年12月16日 13:38 🕮 2910

在构架相关的汇编代码运行完之后,程序跳入了构架无关的内核C语言代码:init/main.c中的start kernel函数,在这个函数 中Linux内核开始真正进入初始化阶段,进行一系列与内核相关的初...

网络数据包收发流程(一):从驱动到协议栈

(wdsfup 2016年05月28日 10:55 🕮 1176

一、硬件环境 intel82546: PHY与MAC集成在一起的PCI网卡芯片,很强大 bcm5461: PHY芯片,与之对应的MAC是TSE C TSEC: Three Spee...

Linux内核-协议栈-初始化流程分析

limes feilengcui008 2015年10月31日 10:35 🚨 880

本文主要针对Linux-3.19.3版本的内核简单分析内核协议栈初始化涉及到的主要步骤和关键函数,不针对协议的解析以及数据 包的处理流程做具体分析,后续有机会再详细分析(主要是作者目前这块才涉及...)1....

lperf 源代码分析(六)

Willon tom 2009年08月21日 15:11 🔘 2001

perf中主要的类lperf的实现中主要的类及其相互间的派生关系如下图所示。PerfSocket类PerfSocket类以Socket为基类派生 而来。该类实现了Iperf用于通信的大多数功能,像发送U...

linux 3.6 启动源码分析(四) rest init

在内核初始化函数start_kernel执行到最后,就是调用rest_init函数,这个函数的主要使命就是创建并启动内核线程init。这个 函数虽然意思为剩下的初始化,但是这个"剩下"的可是内容颇多,下...

网上商城系统

电子商务网站建设

百度广告

Linux启动流程-bootloader至kernel的过程--android系统启动流程

1 Bootloader 对于一般的ARM处理器,CPU上电或复位执行第一条指令所在地址,即第一段程序Bootloader的开始地址,B ootloader一般存于Nor-flash(XIP),支持芯片...

🥭 tianshiyalin 2013年12月10日 20:55 👊 1702

学习Linux-4.12内核网路协议栈 (1.1) ——系统的初始化(do_initcalls)

我们知道,不管在什么样的平台上启动linux,它的开始都是以start_kernel()进行系统的初始化,当然网络协议栈的初始化 也是在这个过程中完成,下面从start_kernel()开始跟踪: st...

🏈 lee244868149 2017年07月03日 17:42 👊 493

linux 3.6 启动源码分析(七) do_initcalls

do initcalls()将按顺序从由 initcall start开始,到 initcall end结束的section中以函数指针的形式取出这些编译到内核的 驱动模块中初始化函数起始地址,来依...

linux 3.6 启动源码分析(五) kernel_init进程 **(回 qing_ping** 2013年12月16日 14:58 □ 3058

在start_kernel最后的rest_init函数中内核创建了两个内核线程,一个是内核线程的管理者,另一个是内核初始化线程kernel_i nit. kernel init它将完成设备驱动程序的初始...

init进程【1】——init启动过程

🦚 zhgxhuaa 2014年04月05日 20:41 📖 9362

众所周知, Linux中的所有进程都是有init进程创建并运行的。首先Linux内核启动, 然后在用户空间中启动init进程, 再启动 其他系统进程。在系统启动完成完成后,init将变为守护进程监视系统其他进...

程序员不会英语怎么行?

北大猛男教你:不背单词和语法,一个公式学好英语





联系我们



请扫描二维码联系客服 webmaster@csdn.net **2**400-660-0108

■ QQ客服 ● 客服论坛

关于 招聘 广告服务 百度 ©1999-2018 CSDN版权所有 京ICP证09002463号

经营性网站备案信息 网络110报警服务 中国互联网举报中心

北京互联网违法和不良信息举报中心

Linux内核驱动加载过程



Linux内核驱动加载过程 驱动加载分为两种情况:静态加载和动

态加载。 1. 静态加载 静态加载的方法是把驱动程序直接编译进内核,然后内核在启动过程中由do_initcall()函数加载。 d...





setup_arch()函数是start_kernel阶段最重要的一个函数,每个体系都有自己的setup_arch()函数,是体系结构相关的,具体 编译哪个体系的setup_arch()函数,由顶层Ma...

linux 3.6 启动源码分析(三) setup_arch

Netmap分析 (六)

😭 superbfly 2016年04月23日 10:48 🕮 776

qing_ping 2013年12月16日 14:05 🕮 1894

netmap.c NIOCTXSYNC和NIOCRXSYNC 这两个使用相同的代码,同步ring case NIOCTXSYNC: case NIOCRXSYNC: nifp = priv...

Giraph源码分析(六)——Edge 分析



win_jmail 2014年04月21日 15:42 🕮 2208

在Vertex类中,顶点的存储方式采用邻接表形式。每个顶点有 VertexId、VertexValue、OutgoingEdges和Halt,boolean 型的halt变量用于记录顶点的状态, fals...