Linux启动过程全接触 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/251/2021_2022_Linux_E5_90_AF_E5_8A_c103_251782.htm 关于Windows启动过程介绍的文章可谓多如牛毛，而对于Linux的介绍却是凤毛麟角。凡是曾经使用过Linux的用户可能都会注意到，当计算机启动时，屏幕上会出现很多信息。一般情况下，这些信息我们可以通过以下的命令看到：cat /var/log/dmesg | more这些信息究竟有什么含义？这个问题看起来似乎很容易回答，因为只要在Linux参考书里查找一下，就会找出一个类似于这样的答案：“ 这是一些内核启动信息…… ”。但是“ 内核启动信息 ”到底是什么意思呢？ 要想对Linux内部工作有所了解，就必须要对Linux内核的体系结构有一个全面的了解。下面我们就去揭开它的秘密。在此，我不想解释Linux内核的体系结构，只想解释（或者说是试图去解释）计算机系统启动进程中一些最基本的概念。这里所说的启动过程是指从按下开关到提示符出现的整个过程。 启动指的是什么 在操作系统的词汇里，启动是指通过处理器执行一些指令，把操作系统的一部分放入到主存中。在启动过程中，Linux内部的数据结构会被初始化，会被赋给一些初始值，并且某些进程会被创建。因为当计算机电源打开时，所有的硬件设备都处于一种不可预知的状态，内存也处于一种不活动的随机状态，所以，计算机的启动过程可以说是一个长且复杂的任务。因此，我们必须知道，之所以叫“ 启动 ”主要是因为计算机体系结构的原因。 在此提请读者注意： 1.对计算机内部的工作和内核的操作有一个基本的了解，对自己非常有益。 2.这篇文章中提到的所有

文件，指的都是Linux内核2.4.2-2版本里的文件。这些文件对于所有的Linux内核来说都是相同的，并且可以在任何一个Linux系统里找到它们，此处我使用的是Red Hat 7.1。 3.在本文里，讨论范围限于IBM PC体系结构。 BIOS及其功能 当计算机打开电源时，内存里包含的是一些随机的数据，所有的东西都没有被初始化，操作系统也没有被加载。开始整个启动过程的是一个特殊的硬件电路，它触发CPU的Reset脚的逻辑值。然后，一些CPU的寄存器比如CS（一个分段寄存器：代码段寄存器，它指向含有程序指令的段），eip（在执行指令过程中，当CPU检测到一个意外事故发生时，它会做出三种类型的判断：错误、陷阱、中止，这取决于eip寄存器的值，它存储在内核模块栈里）就会被给定一个值。接着，物理地址为0xffffff0的代码将被执行。这个地址被存储在一个只读存储器（ROM）里。BIOS（基本输入/输出系统）实际上是一段存储在ROM里的程序。它包含了一系列可以被某些操作系统调用，用于处理计算机各种硬件设备的中断驱动和低级程序。其中微软的DOS就是这样的一种操作系统。 Linux是否使用附于计算机系统的BIOS来初始化硬件设备？或者说，是否有其它的东西来完成同样的任务？不过这个问题没有那么简单，必须要了解一些知识。我们从80386模式开始。Intel微处理器实现地址翻译（从逻辑地址->线性地址->物理地址）有两种不同的途径，分别称作实模式和保护模式。实模式存在主要是为了使得处理器可以和较老的处理相兼容。事实上，所有的BIOS程序都是在实模式下运行的。但是，Linux内核是在保护模式下运行，而不是在实模式下。因此，一旦初始化完成后，Linux就不再使用BIOS，而是完全由自己来为计

算机上的所有硬件提供驱动程序（这点和DOS是不一样的）。 那么什么时候Linux使用保护模式？为什么BIOS不能使用相同的模式？BIOS使用实模式是因为其在操作过程中使用的是实模式地址，并且在计算机刚打开电源时，只有实模式地址可用。一个实模式地址由段地址和偏移地址组成，因此，相应的物理地址就为段地址×（2×8） 偏移。 那么，这是不是意味着在整个启动过程中，Linux就从来不使用BIOS了呢？答案是否定的。在启动阶段，Linux从硬盘或者其它外部设备加载内核时，需要使用BIOS。 让我们来看一下启动时BIOS主要做了哪些操作： 1.BIOS要对硬件进行一系列彻底的检测。这个步骤主要是检查系统安装有哪些设备,以及它们工作是否正常。通常把这个步骤叫做自检（Power-On Self-Test,POST），这时会显示版本及其它很多相关的硬件信息。 2.BIOS要对硬件进行初始化。这一步非常重要，因为它要保证所有的硬件设备在IRQ（中断请求）和I/O端口操作时都没有冲突。等这步完成以后，它会显示一个已经安装的PCI设备表。 3.接着到了操作系统，BIOS将查找一个可以引导的操作系统。这取决于BIOS的设置，它可以从软盘、硬盘或者光盘启动。 4.一旦发现一个合法的设备，BIOS就会把其第一扇区的内容复制到物理地址，即从0x00007c00开始的内存中，然后跳至刚加载的地址并执行之。 到此为止，BIOS所要做的工作就全部完成了。