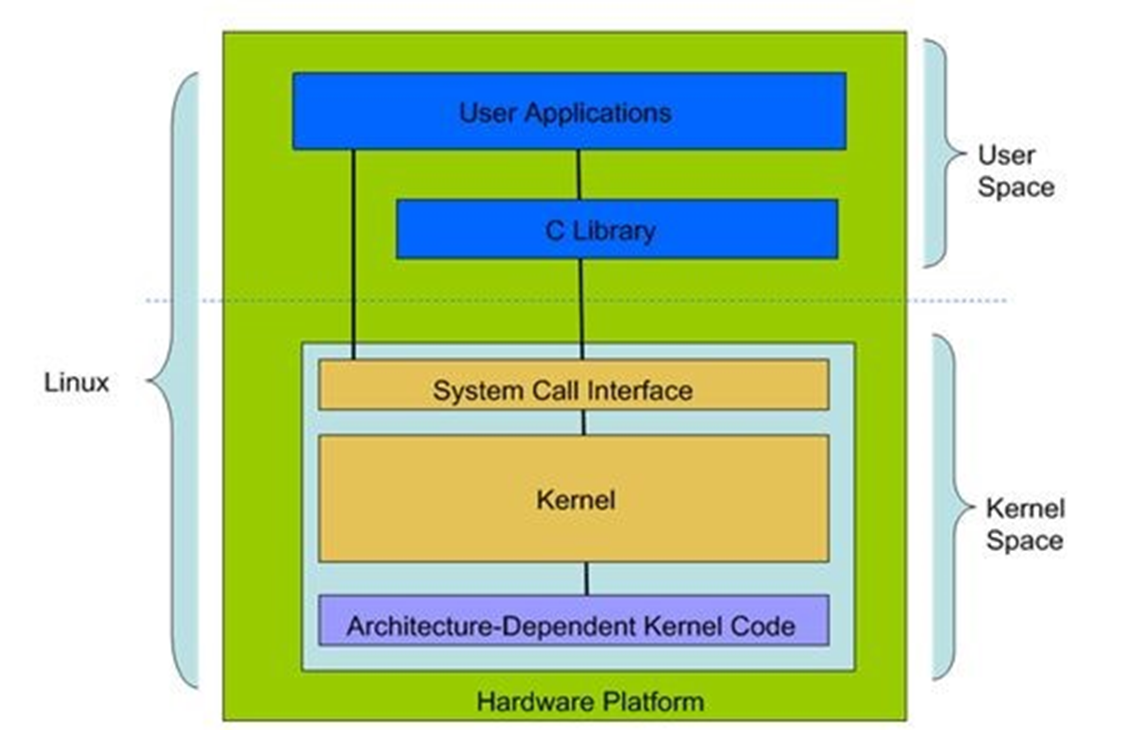
linux体系结构：



linux体系分为用户空间和内核空间：

程序员角度：底层和应用分开，分工产生效率

安全性角度：为了保护内核，CPU通常实现多种工作模式。

-以ARM为例：7种工作模式，不同模式下CPU可以执行的指令

或者访问的寄存器不同：

1.用户模式usr 2.系统模式sys 3.管理模式svc

4.快速中断fiq 5.外部中断irq 6.数据访问终止abt

7.未定义指令异常

-以x86为例：4个不同级别的权限，Ring0-Ring 3;Ring0下

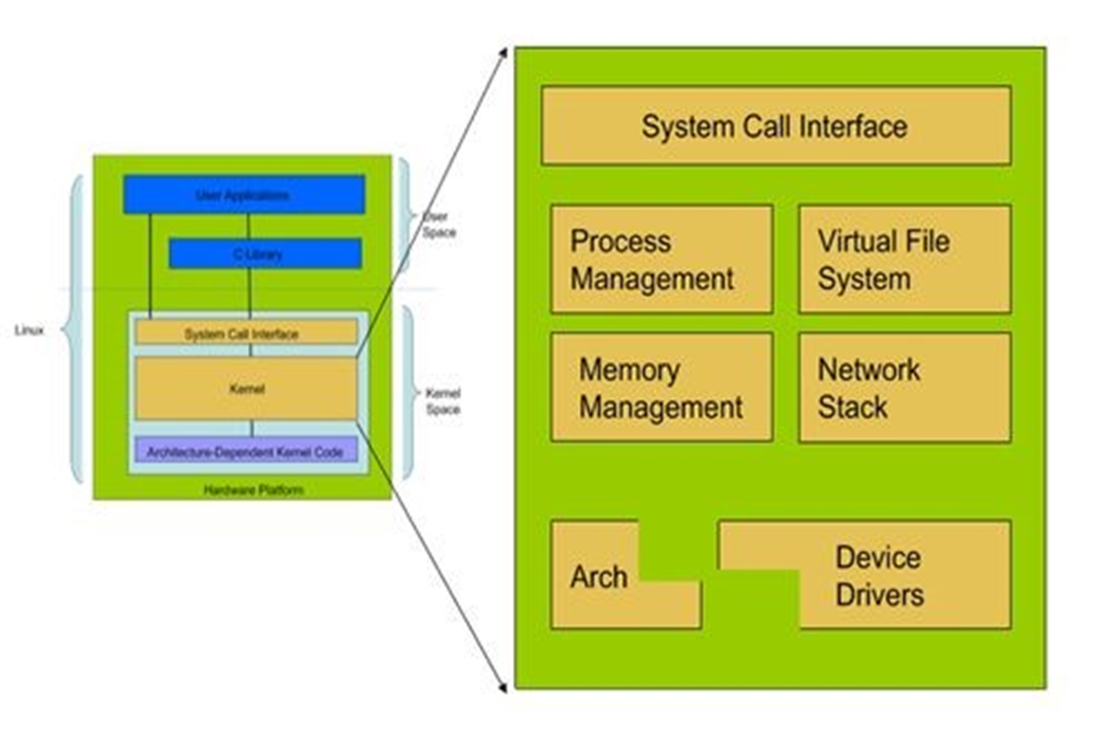
可以执行特权指令，访问io设备；Ring3则有很多的限制

-以Android为例：将应用放到JAVA虚拟机上运行，应用更加远离底层

用户空间和内核空间是程序执行的两种不同状态，我们可以通过

“系统调用”和“硬件中断“来完成用户空间到内核空间的转移

Linux内核结构：



**System Call Interface （SCI层）**

为用户空间提供了一套标准的系统调用函数来访问Linux内核。

**Procees Management（PM）**

– 进程管理是创建进程（fork、exec）,停止进程（kill、exit）,并控制他们之间的通信（signal等）。还包括进程调度，控制活动进程如何共享CPU

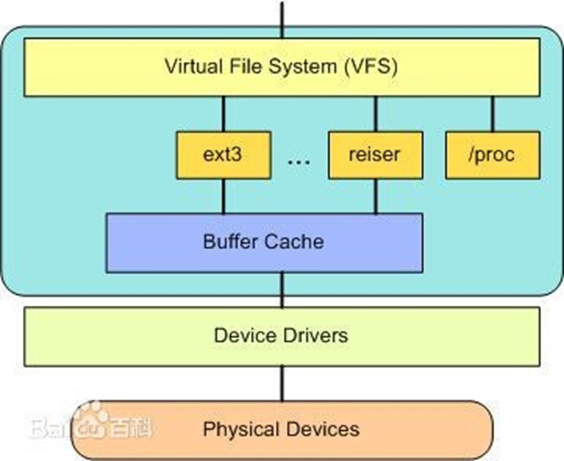
**Memory Management（MM）**

– 内存管理的主要作用是控制多个进程安全的共享内存区域

**Virtual File Systems（VFS）**

虚拟文件系统，隐藏各种文件系统的具体细节，为文件操作提供统一的接口

Linux提供了一个大的通用模型，使这个模型包含了所有文件系统功能的集合（一切皆文件）



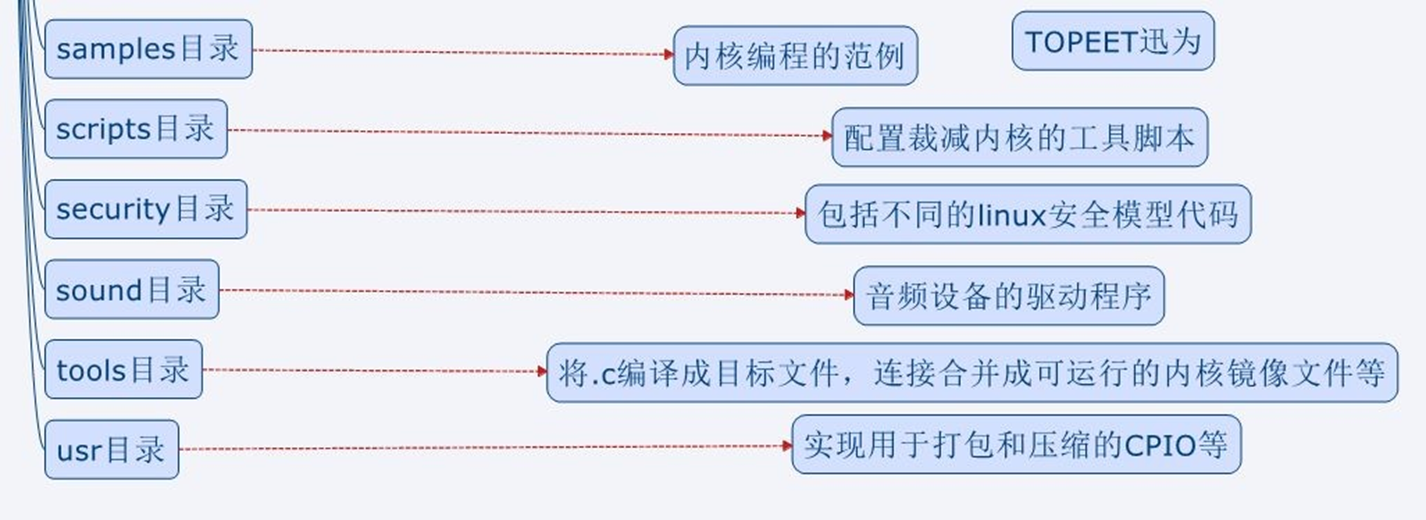
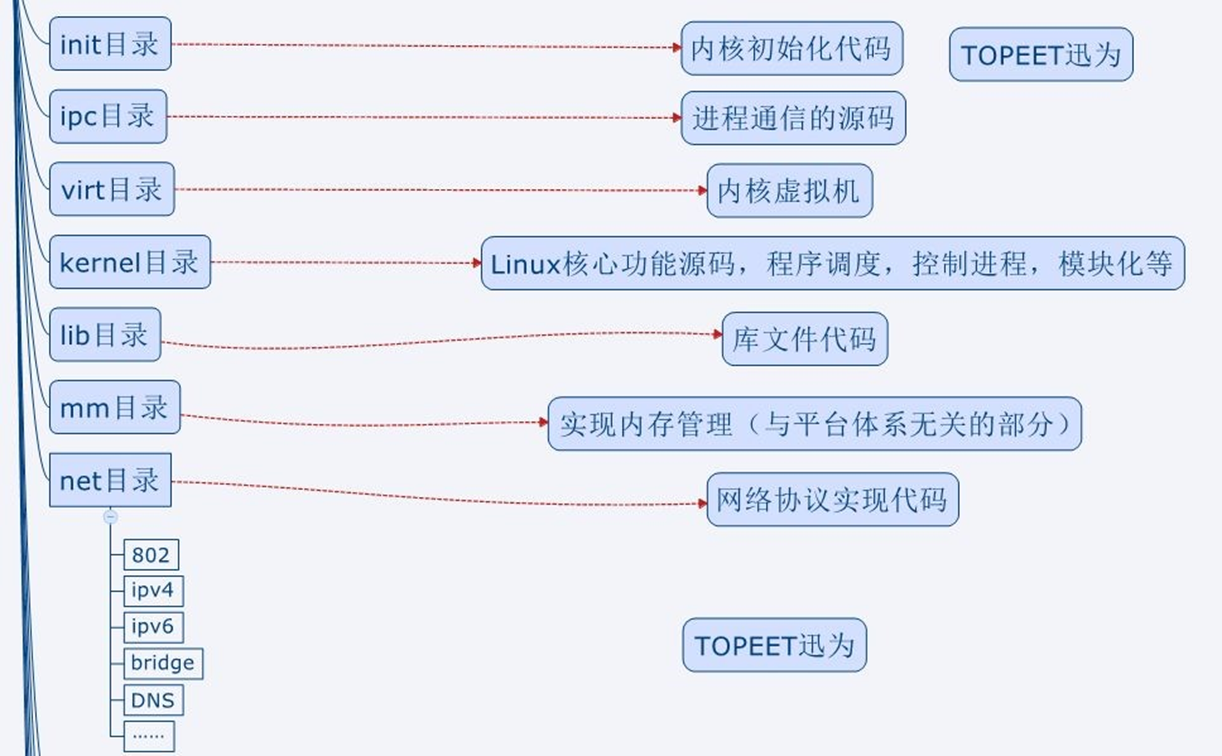
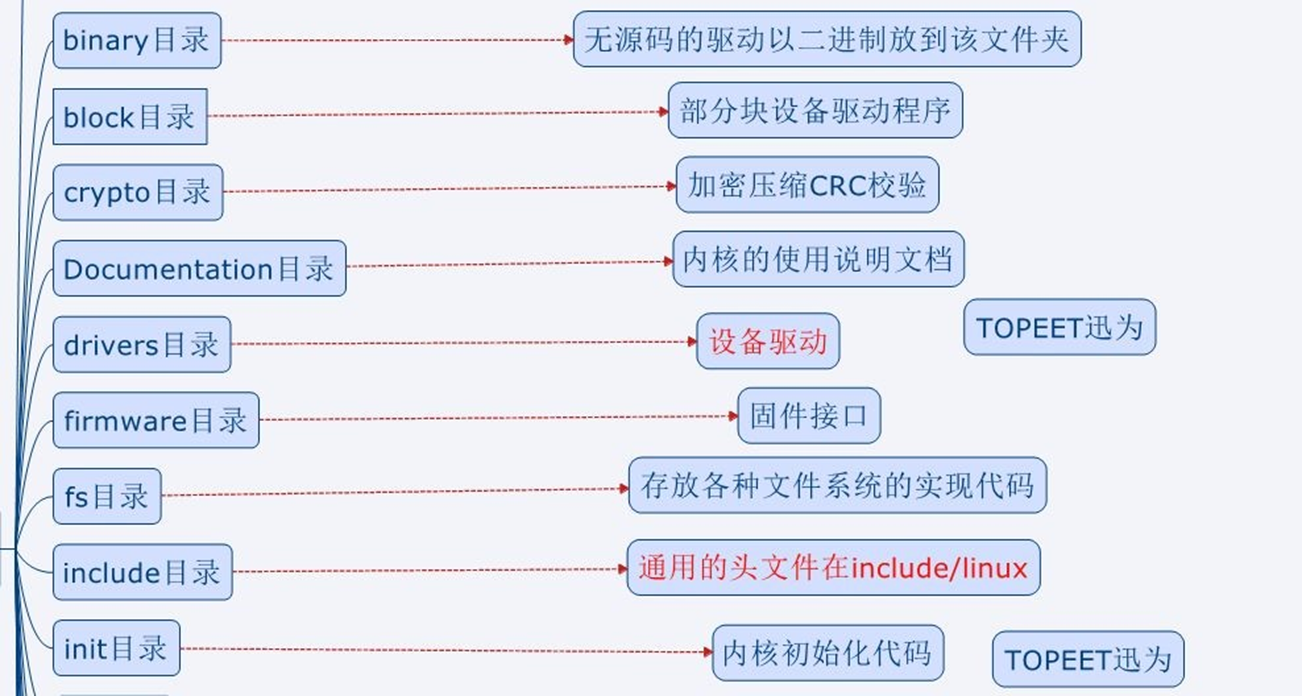
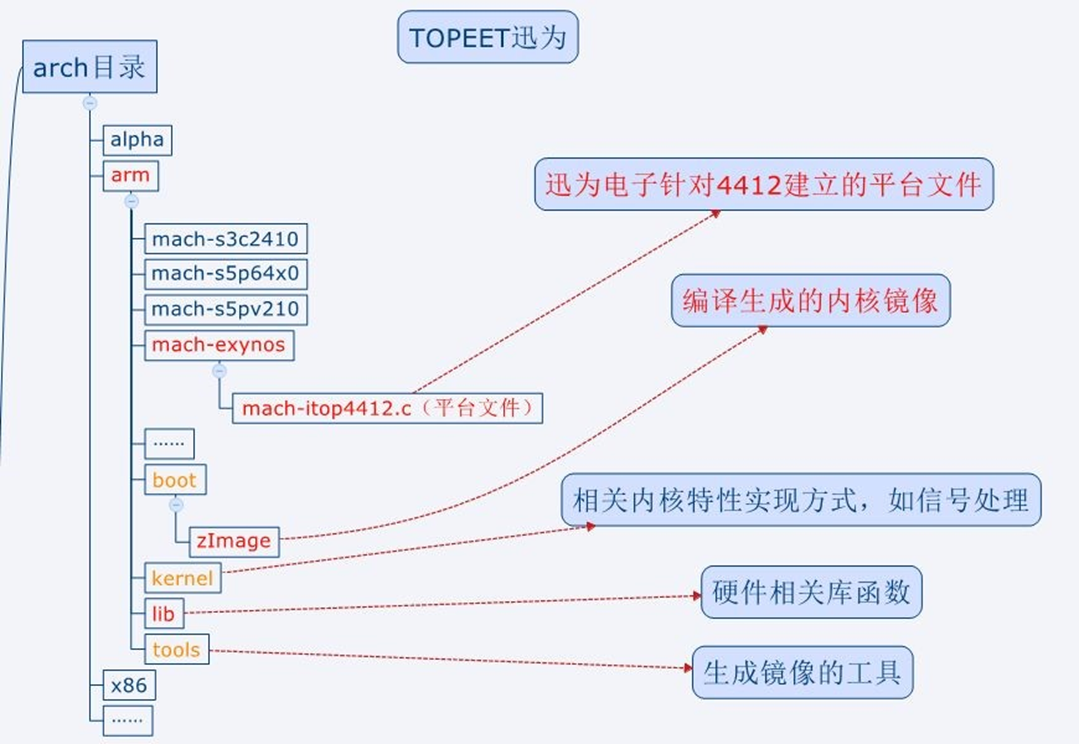
**Device Drivers设备驱动**

Linux内核中有大量的代码在设备驱动程序部分，用于控制特定的硬件设备。

Linux驱动一般分为网络设备、块设备、字符设备、杂项设备

**网络协议栈**

– 内核网络协议栈为Linux提供了丰富的网络协议实现。



内核源码解压编译后有1.3G，通过裁减编译出的linux镜像zImage不到4M