X86 架构内存编址及启动过程

1. 几个名词 X86 架构:

Intel, AMD CPU 采用的结构体系.实模式, 16 位模式: X86 体系 CPU 启动是处于此模式. DOS 启动完成也是此模式, 直接使用物理地址保护模式,虚模式,32 位模式: linux,windows 启动后进入此模式. 使用虚拟地址.

2. 8086CPU 的限制.

X86 架构是从 8086 发展来的, 所以后来的 CPU 都保留了 8086 的限制. 理解内存编址, 要了解 8086.8086 是 16 为 CPU, 地址总线 20 位.8086 采用分段机制. 寻址方式是段基址+偏移量.8086 中,基本的段寄存器是 CS,DS, SS, ES. 都是 16 位寄存器.寻址时, 段寄存器地址左移 4 位, 加上偏移量. 就是需要的物理地址.所以 8086 最大只能访问 0xfffff 以内的空间(1M).

3. 8086 编址

在这种体系结构下, 640K(0xA0000)以下称为基本内存. 这就是系统可用的内存.0xA0000 ~ 0xBFFFF 用于显卡缓存. (640K 开始的部分)0xC0000 开始用于 BIOS, 一般显卡 BIOS 从 0xC0000 开始.系统 BIOS 放到可访问的 1M 内存最后. 中间是一些其他设备的 BIOS, 都有各自的固定起始地址.

4. 8086 启动过程.

系统加电启动的时候, CPU 清 0. 然后 CS 寄存器设为 0xFFFF, IP 寄存器设为 0x0000. 对应的物理地址就是 0xFFFF0.可见这是在系统 BIOS 里面. 一般是一条跳转指令, 跳转到真正的 BIOS 处开始执行.BIOS 首先自检, 此时如果发现严重错误, 比如没有内存,直接鸣喇叭,(因为还没用显卡初始化. 如果没有 CPU 呢? 没有任何反应).然后执行显卡 BIOS, 显示显卡信息. 显示系统 BIOS 自己的信息.如果是从硬盘启动,读取 MBR 到 0x7C00 处.如果引导程序是 GRUB, MBR 里就包括 GRUB 的 stage1 代码.如果不使用 state1.5, 就通过物理扇区直接寻址 stage2(因为没有文件系统).如果是 linux 系统, 就加载 kernel image.如果是用"make zImage"编译的内核, 就加载到 0x10000 处(64K 位置).如果是用"make bzImage"编译的内核, 就加载到 0x100000 处(1M 位置).剩下的就是内核的事情了.