

为了避免操作系统和关键数据被用户程序破坏，将处理器的执行状态分为内核态和用户态。

为何要引入内核态和用户态？

- 内核态
内核态是操作系统管理程序执行时所处的状态，能够执行包含特权指令在内的一切指令，能够访问系统内所有的存储空间。
- 用户态
用户态是用户程序执行时处理器所处的状态，不能执行特权指令，只能访问用户地址空间。
- 用户程序运行在用户态,操作系统内核运行在内核态。

- 1启动BIOS固件
- 2.BIOS固件将加载程序从磁盘的引导扇区(512字节) 加载到0x7c00
- 3.将操作系统的代码和数据从硬盘 加载到内存中； 跳到操作系统起始地址
- 拓展：为什么BIOS不直接加载操作系统，而要先执行加载程序？

- 并发性
- 共享性
- 虚拟性
- 不确定性

- 处理机管理
 - 进程控制
 - 进程调度
 - 进程同步
 - 进程通信
- 存储管理
 - 内存分配
 - 内存保护
 - 内存映射
 - 内存扩充
- 设备管理
 - 缓冲管理
 - 设备分配
 - 设备处理
- 文件管理
 - 设备独立性与虚拟设备
 - 文件存储空间的管理
 - 目录管理
 - 文件读写管理
 - 文件存取控制
- 用户接口
 - 命令行
 - 系统调用
 - 图形接口

OS基本概念

双重模式

系统引导

OS特征

OS定义

设置OS的目的

OS分类

通用操作系统

OS结构

OS功能

- 最重要的系统软件
- 资源管理器
- 用户（应用程序）和计算机硬件系统之间的接口

- 提高计算机系统的效率
- 增强系统的处理能力
- 提高系统资源的利用率
- 方便用户使用计算机

- 批处理系统
- 分时操作系统
- 实时操作系统

- 简单结构
- 层次化方法
- 微内核结构
- 模块