



计算机操作系统

1 计算机与操作系统 - 1.3 深入观察操作系统

1.3.5 程序接口的视角

掌握操作系统的程序接口

掌握系统调用的实现机制

掌握系统调用的实现要点

掌握系统调用的实现流程

操作系统的程序接口

- 操作系统的程序接口：操作系统为程序运行扩充的编程接口
- 系统调用：操作系统实现的完成某种特定功能的过程；为所有运行程序提供访问操作系统的接口
- POSIX支持

系统调用的实现机制

- 陷入处理机制：计算机系统中控制和实现系统调用的机制
- 陷入指令：也称访管指令，或异常中断指令，计算机系统为实现系统调用而引起处理器中断的指令
- 每个系统调用都事先规定了编号，并在约定寄存器中规定了传递给内部处理程序的参数

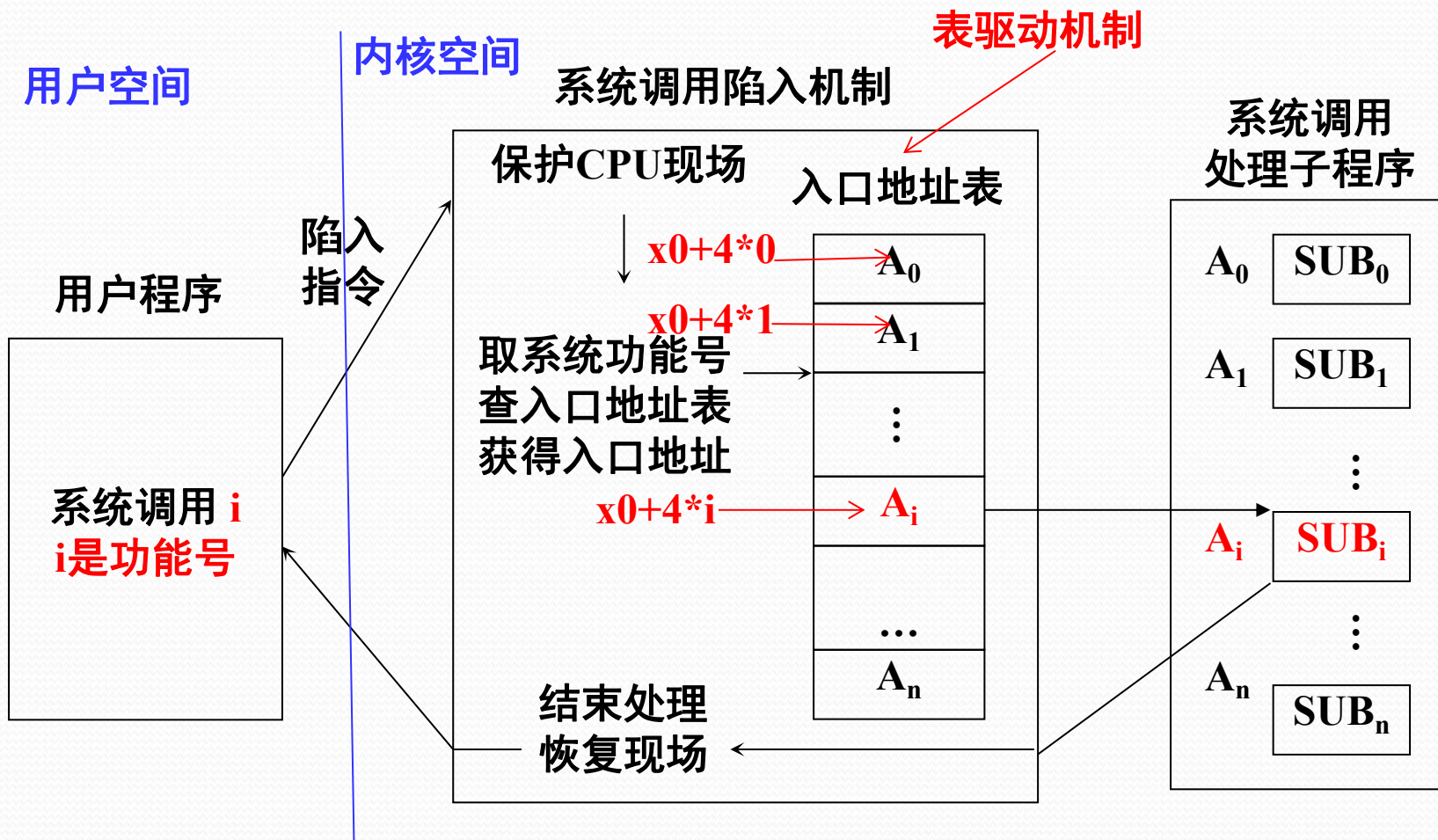
系统调用的实现要点

- 编写系统调用处理程序
- 设计一张系统调用入口地址表，每个入口地址指向一个系统调用的处理程序，并包含系统调用自带参数的个数
- 陷入处理机制需开辟现场保护区，以保存发生系统调用时的处理器现场

系统调用的实现流程

按号定位的前提:

- (1) 各数据结构的尺寸相同;
- (2) 各数据结构按号连续排列



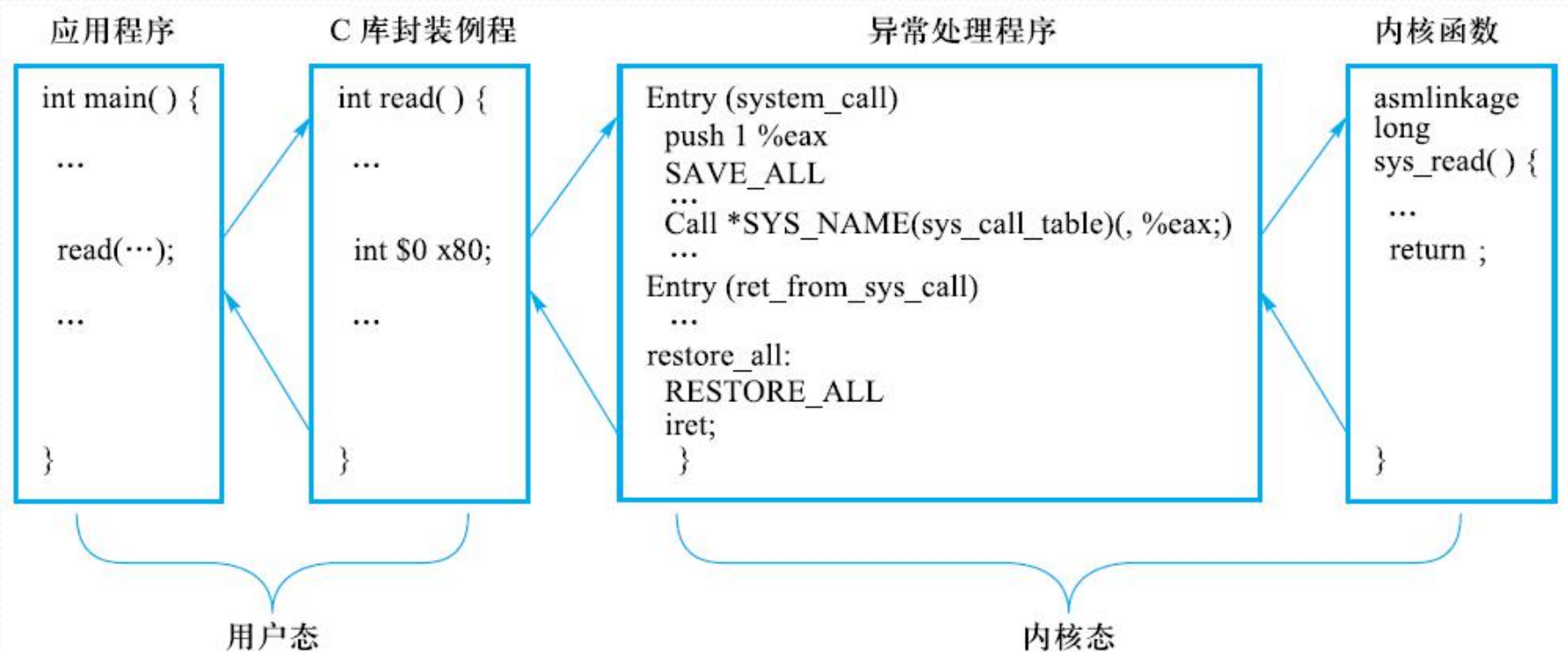
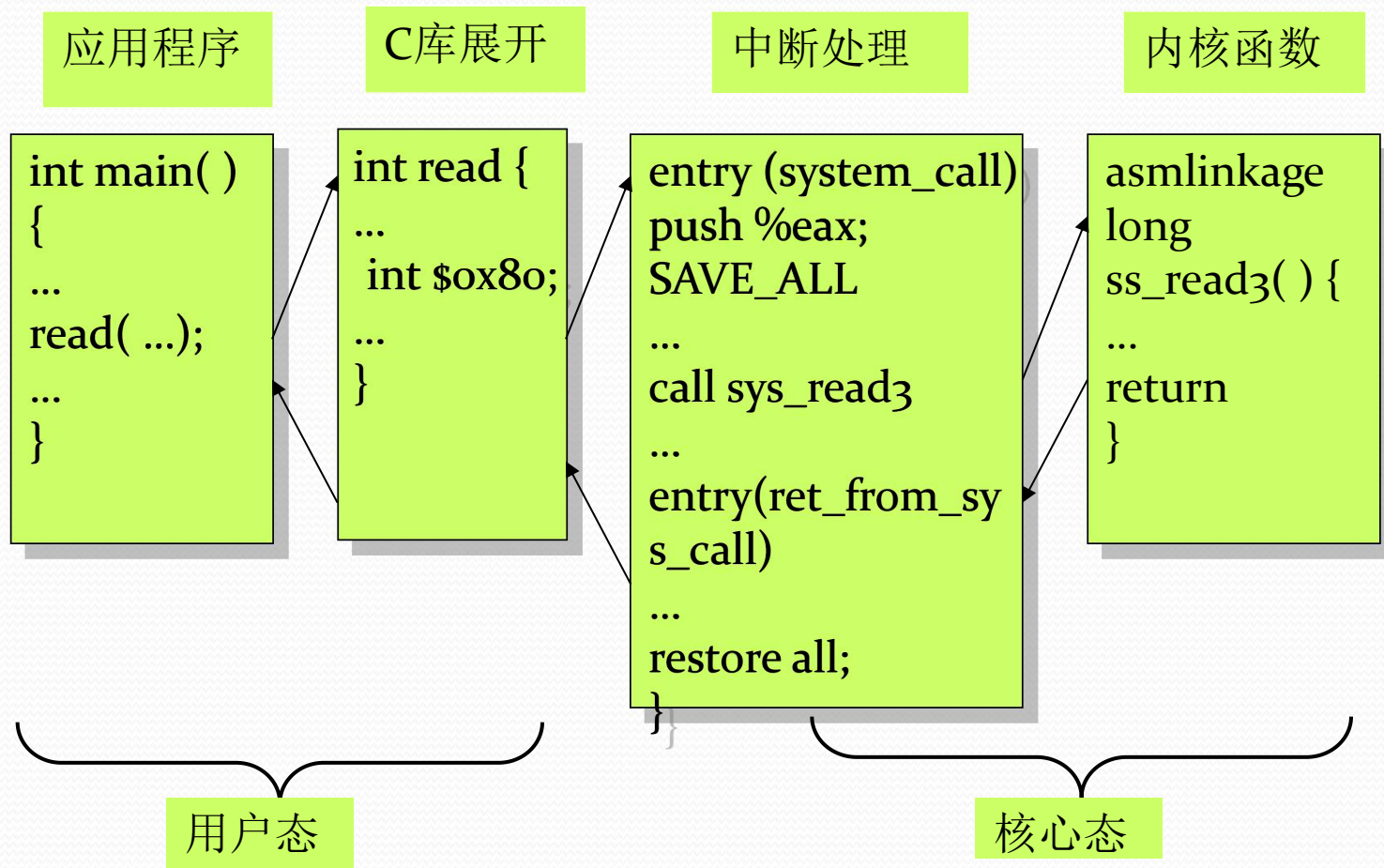


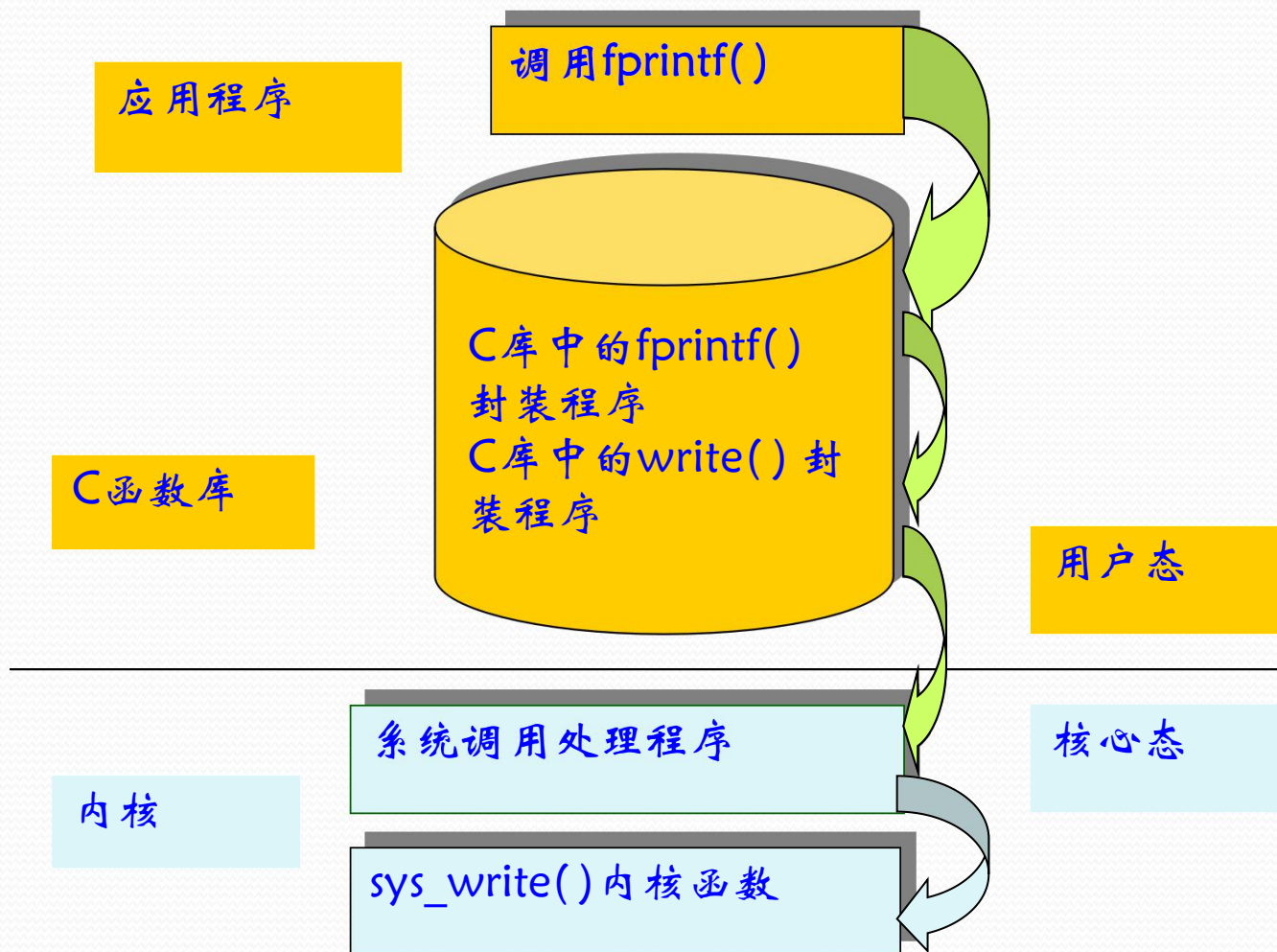
图 1-11 Linux 系统调用执行流程

Linux 系统调用执行流程



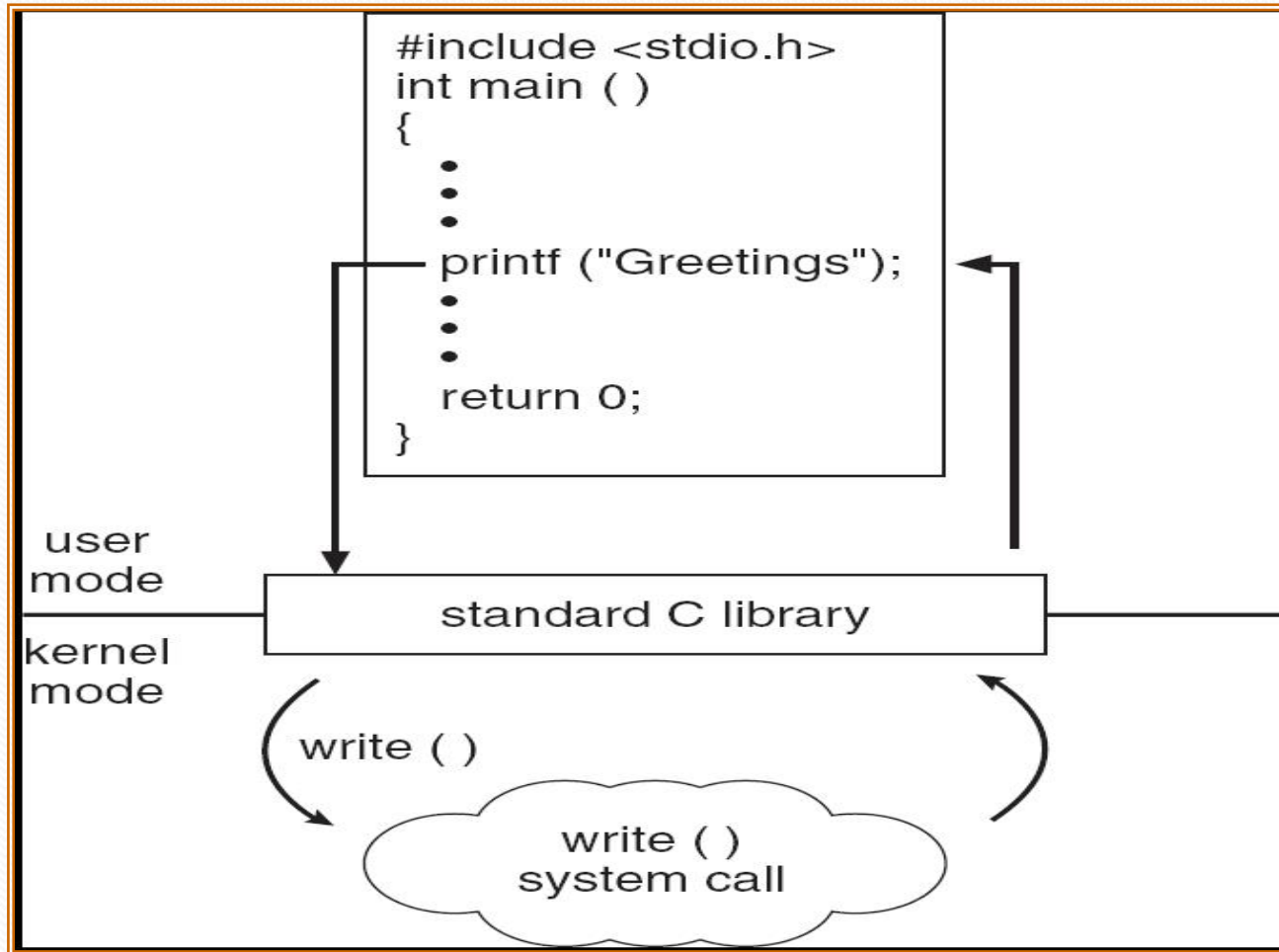
操作系统提供的程序接口(5)

应用程序、库函数、系统调用的调用关系链



Standard C Library Example

- C program invoking printf() library call, which calls write() system call



Steps in making system call read

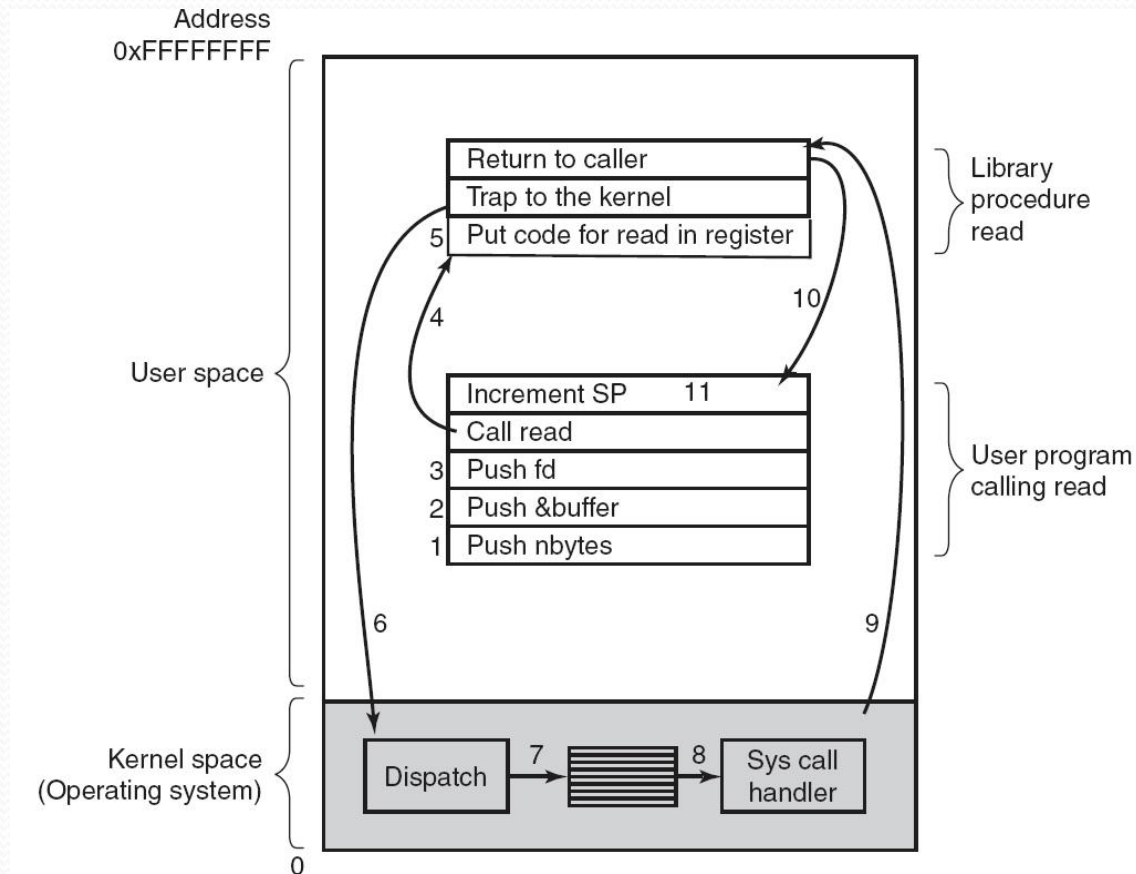
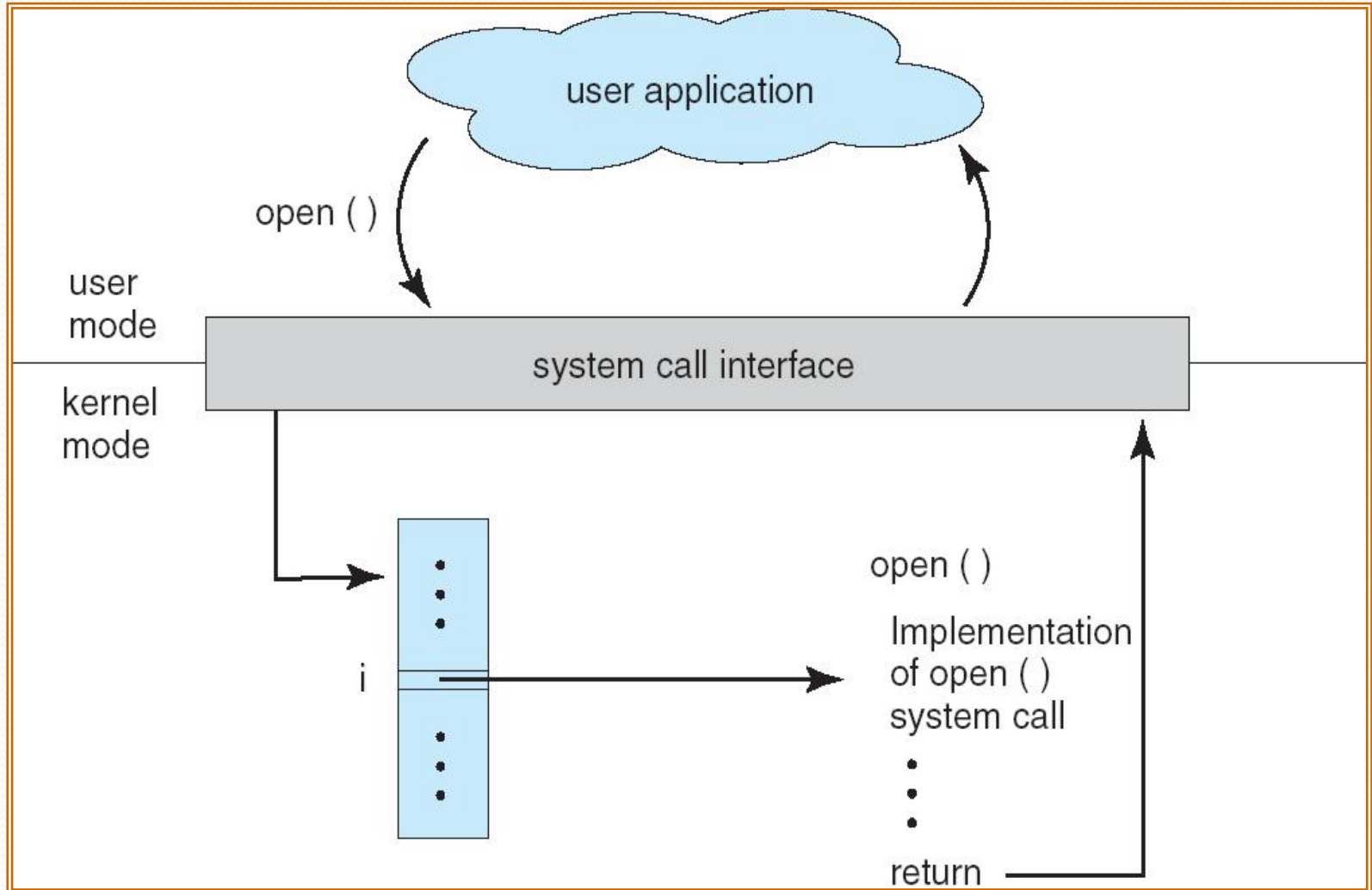
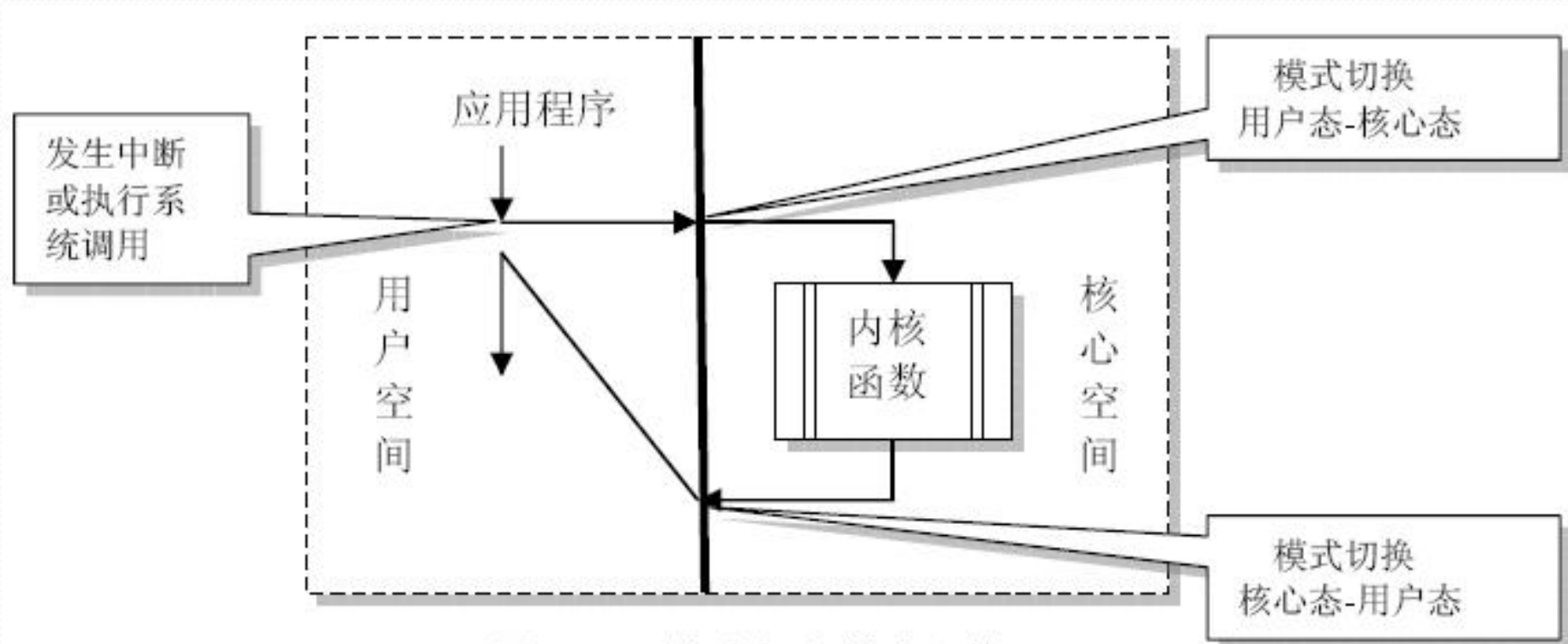


Figure 1-17. The 11 steps in making the system call `read(fd, buffer, nbytes)`.

API – System Call – OS Relationship

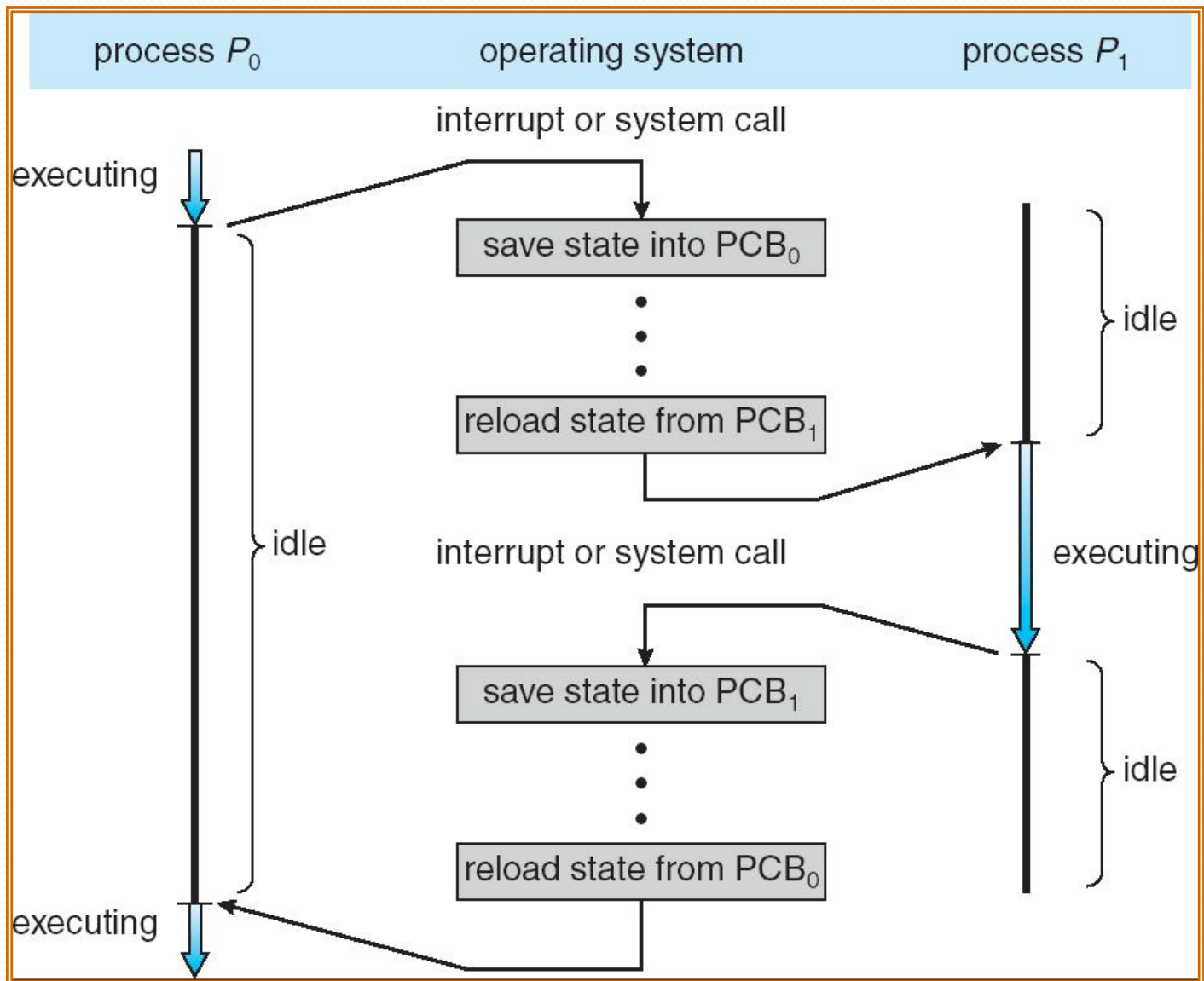


两个空间与模式切换



两个空间与模式切换

CPU Switch From Process to Process



Transition from user to kernel mode

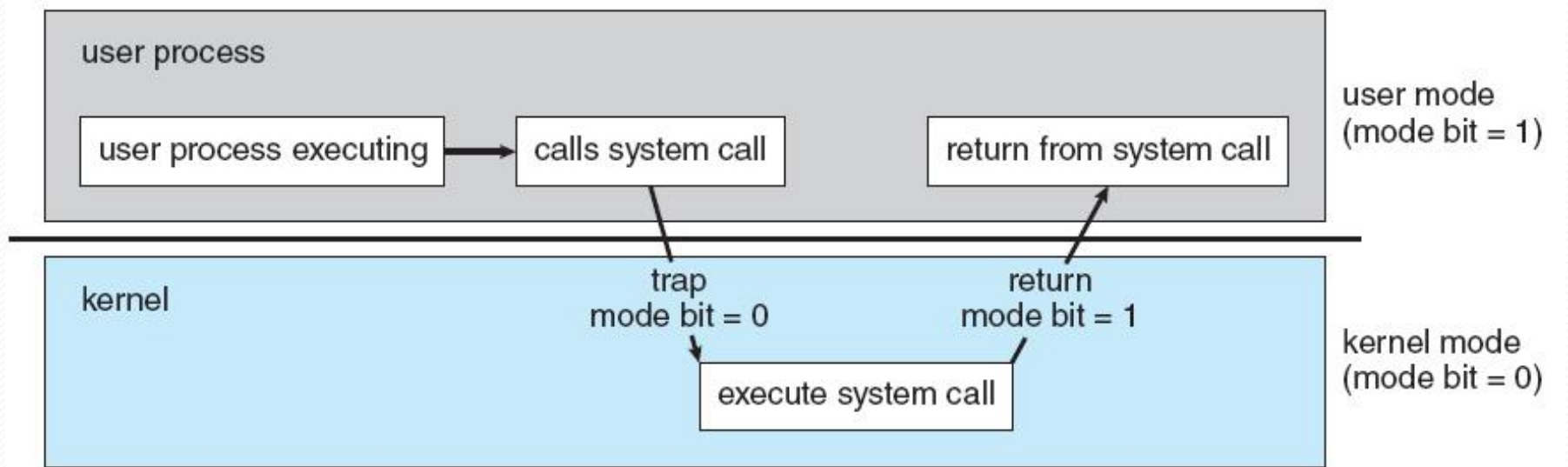


Figure 1.13 Transition from user to kernel mode.