南京大学
NANJING UNIVERSITY

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**

- 今天会向大家讲解几个重要的关联概念：
- 1. 系统调用与陷入机制，系统调用的实现流程，系统调用的入口地址表。
- 2. 处理器的模式：内核态(核心态/管态)与用户态(目态)。
- 3. 处理器的指令集：特权指令与非特权指令，特权指令只能在内核态使用，非特权指令在内核态和用户态都可以使用。
- 4. 特别提醒：内核态时处理器在执行内核程序的指令，用户态时处理器执行用户进程的指令，两态合用一个处理器（在单处理器系统中）。
- 核心思想：操作系统内核是中断驱动的，或者讲 中断是激活操作系统的唯一方式。

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

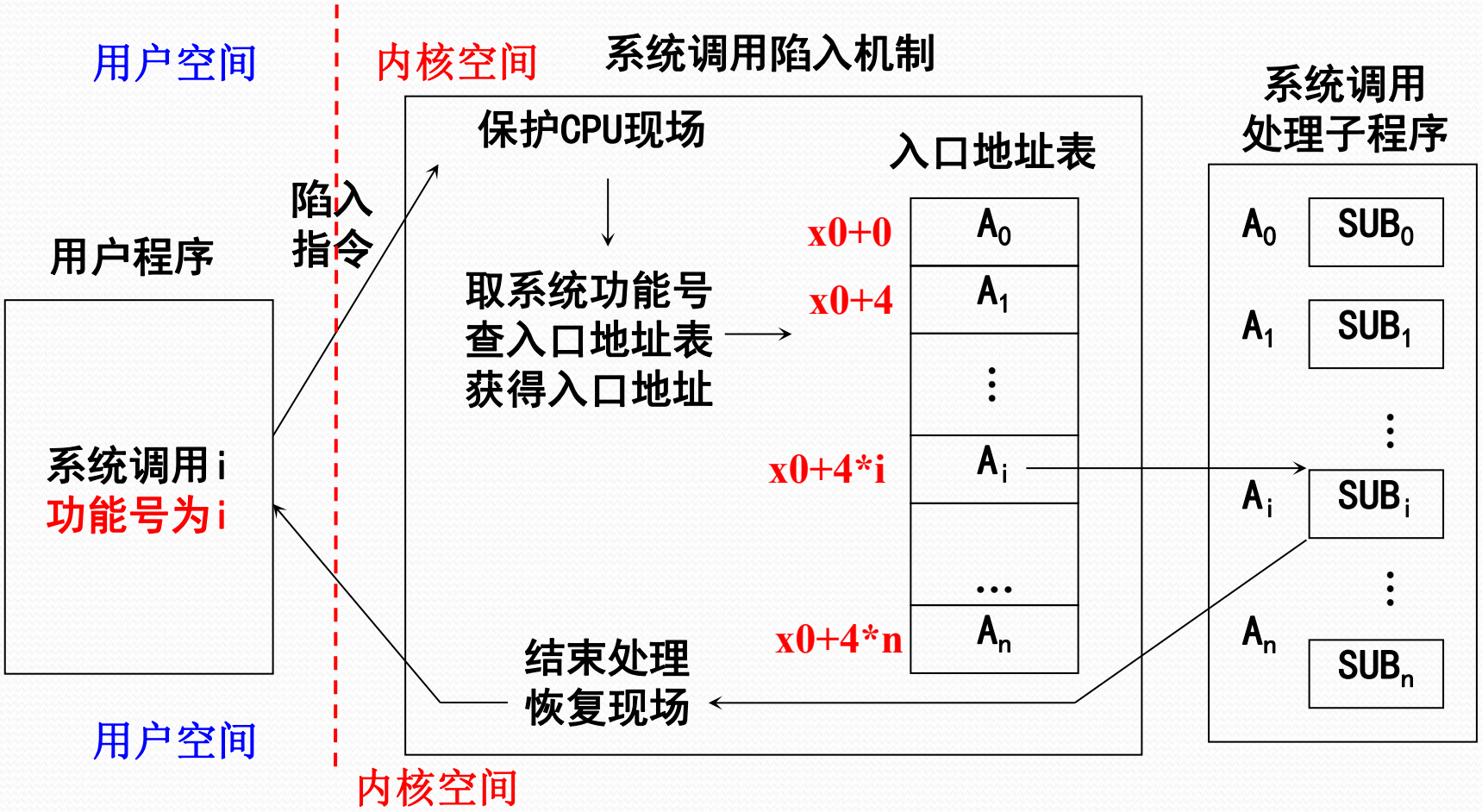
- 编写系统调用处理程序
- 设计一张系统调用入口地址表，每个入口地址指向一个系统调用的处理程序，并包含系统调用自带参数的个数
- 陷入处理机制需开辟现场保护区，以保存发生系统调用时的处理器现场

没有名称空间
(No Name Space)

通过指针的地址计算做定位,
表驱动按号索引

系统调用的实现流程

以32位系统为例, 每个地址占4字节, 以入口地址表的起地址x0为起点,
依据功能号向下偏移4*i=x0+4*i, 读取系统调用处理子程序SUBi的入口地址,
然后执行SUBi, 执行完成之后, 再返回用户空间.



Linux系统调用执行流程

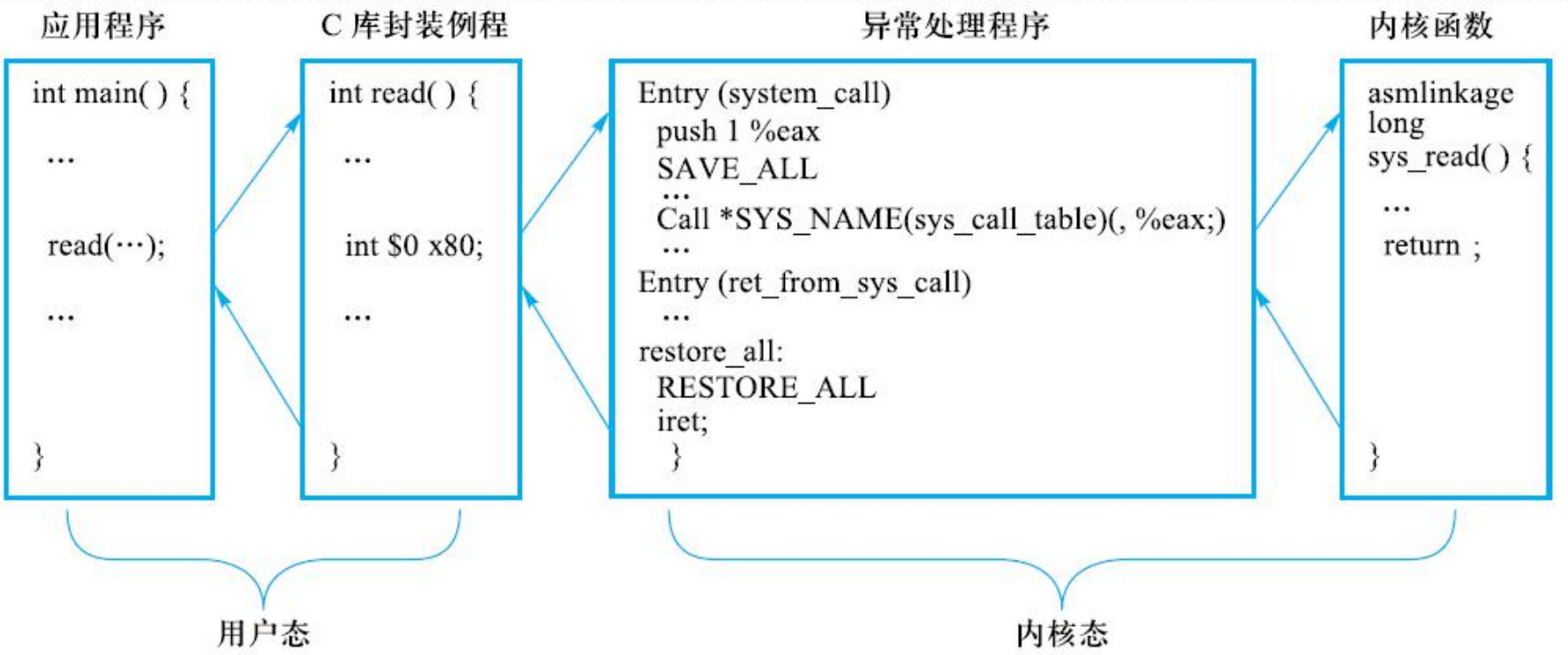
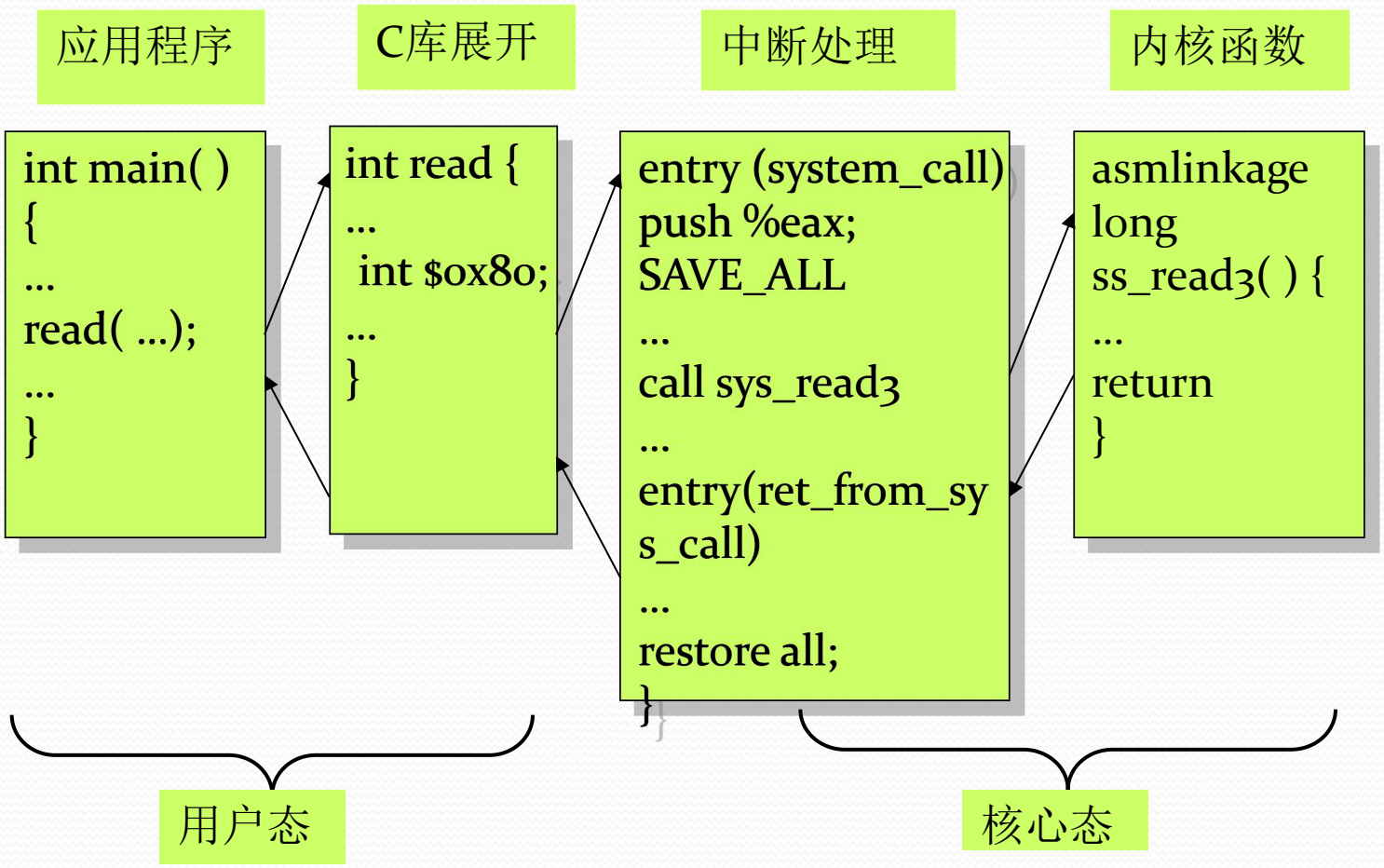


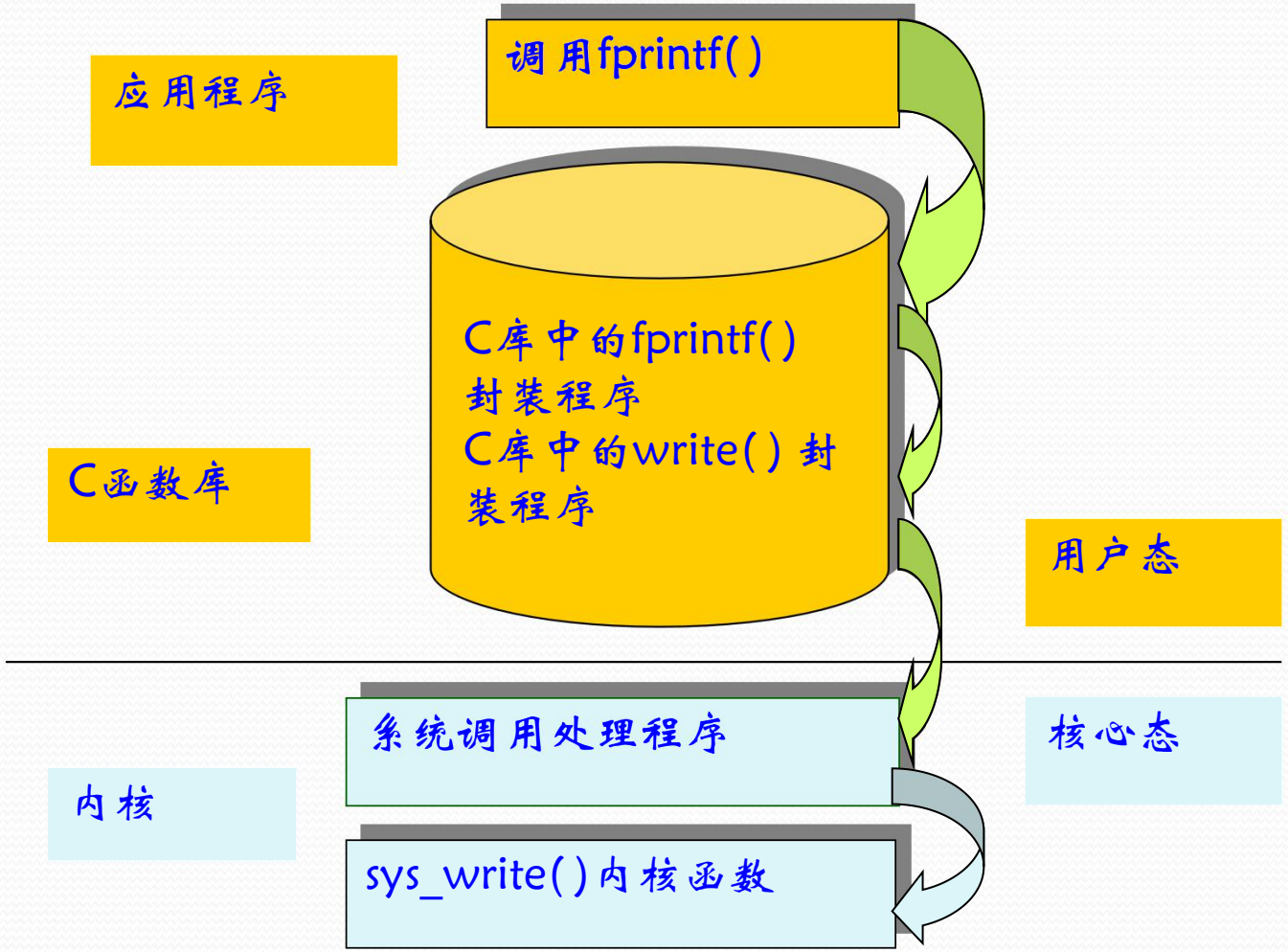
图 1-11　Linux 系统调用执行流程

Linux系统调用执行流程



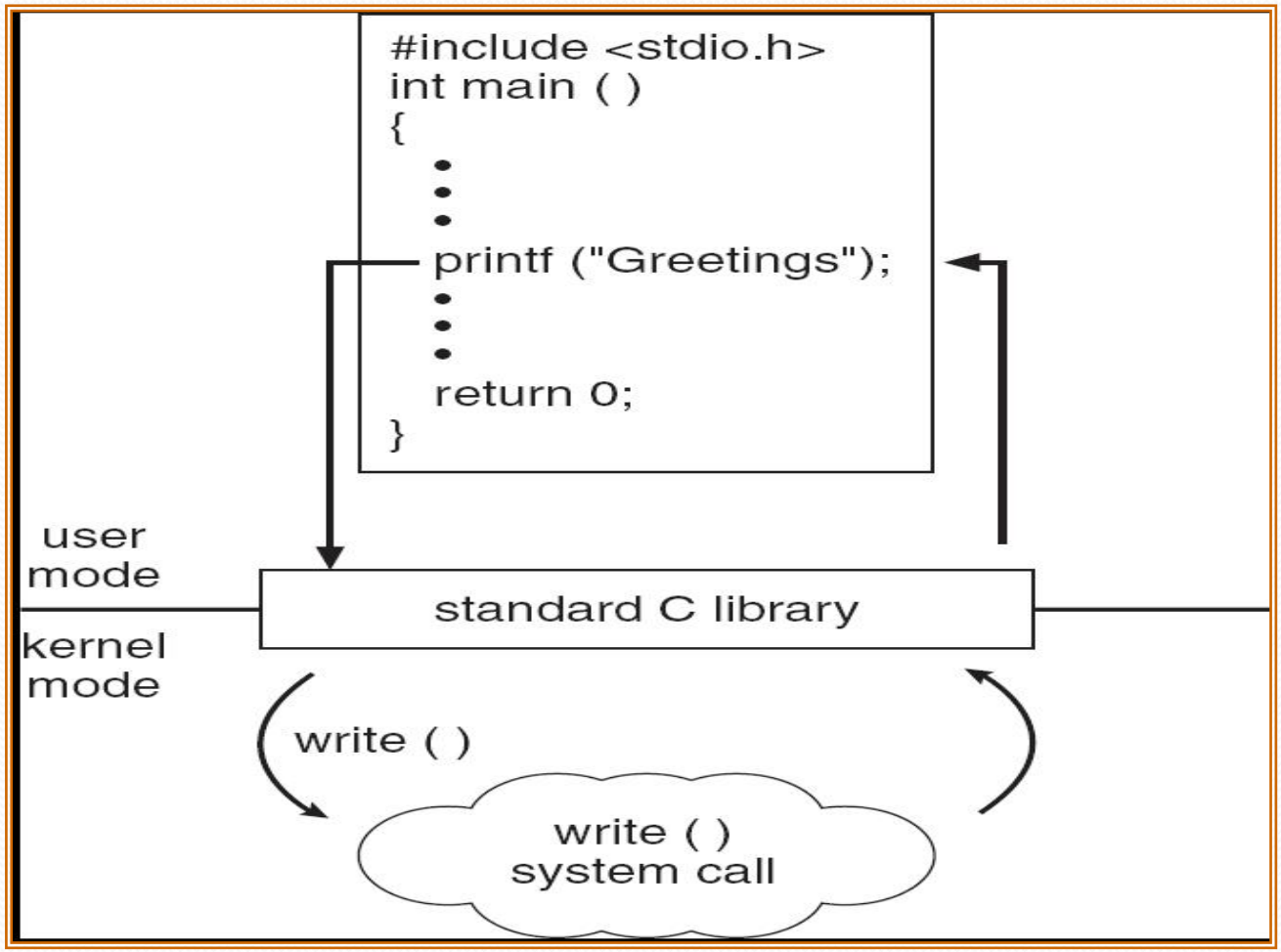
操作系统提供的程序接口(5)

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



书教材 Operating System Concepts (10th), pp.69

Steps in making system call read

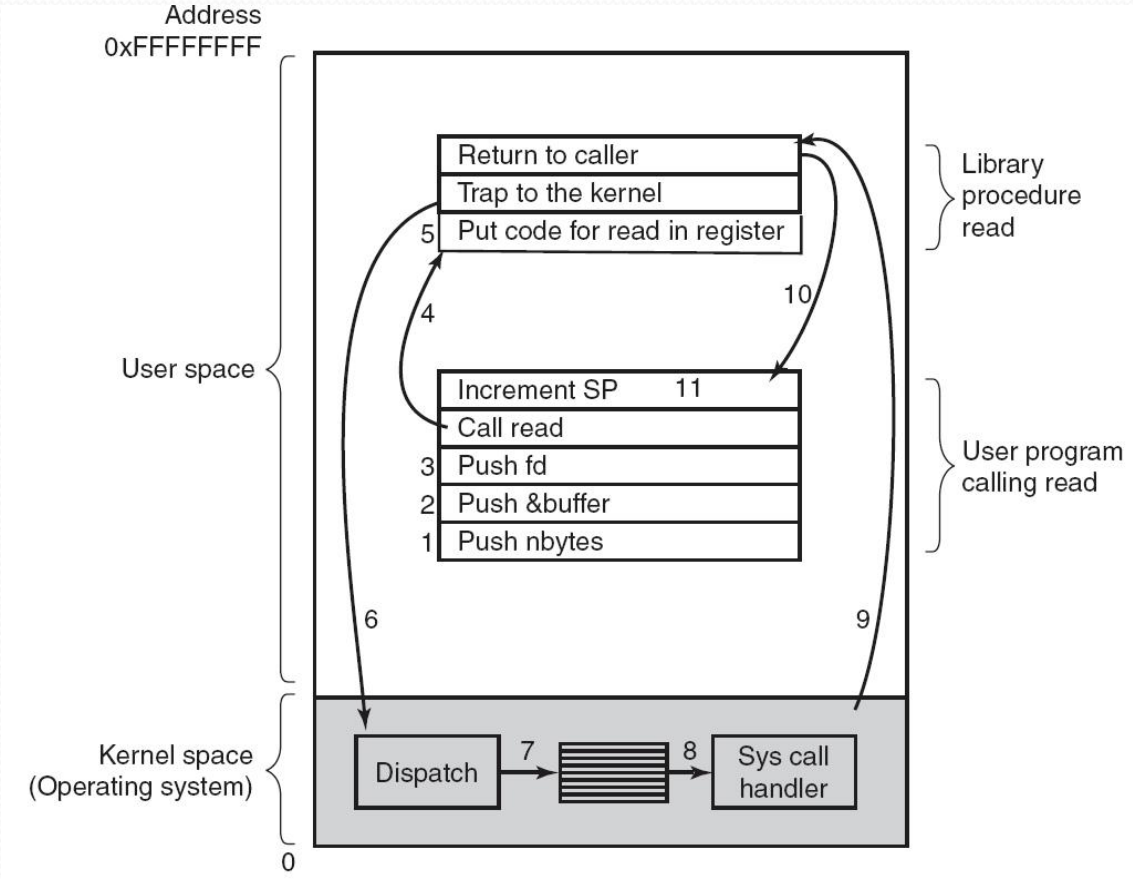
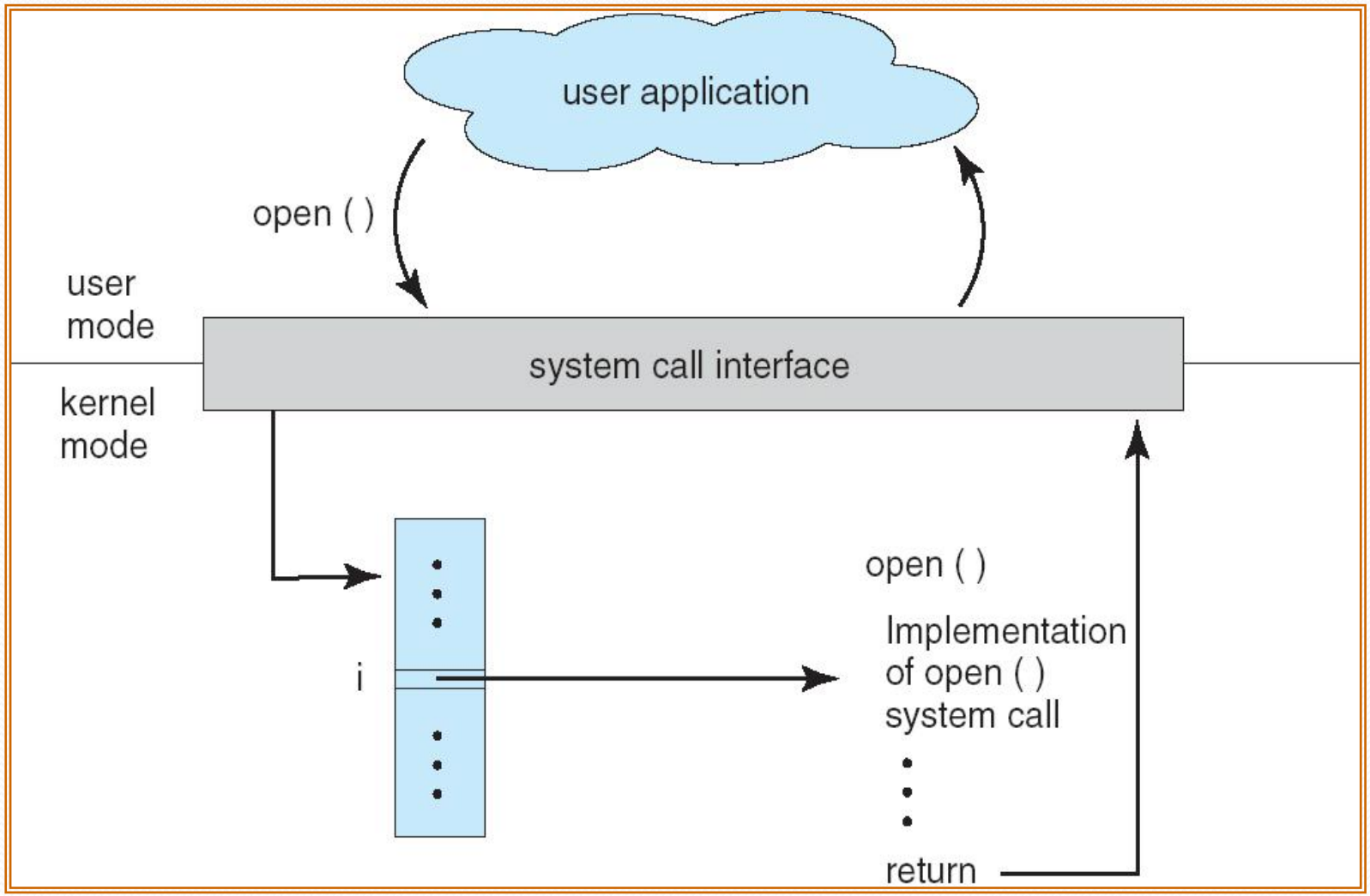


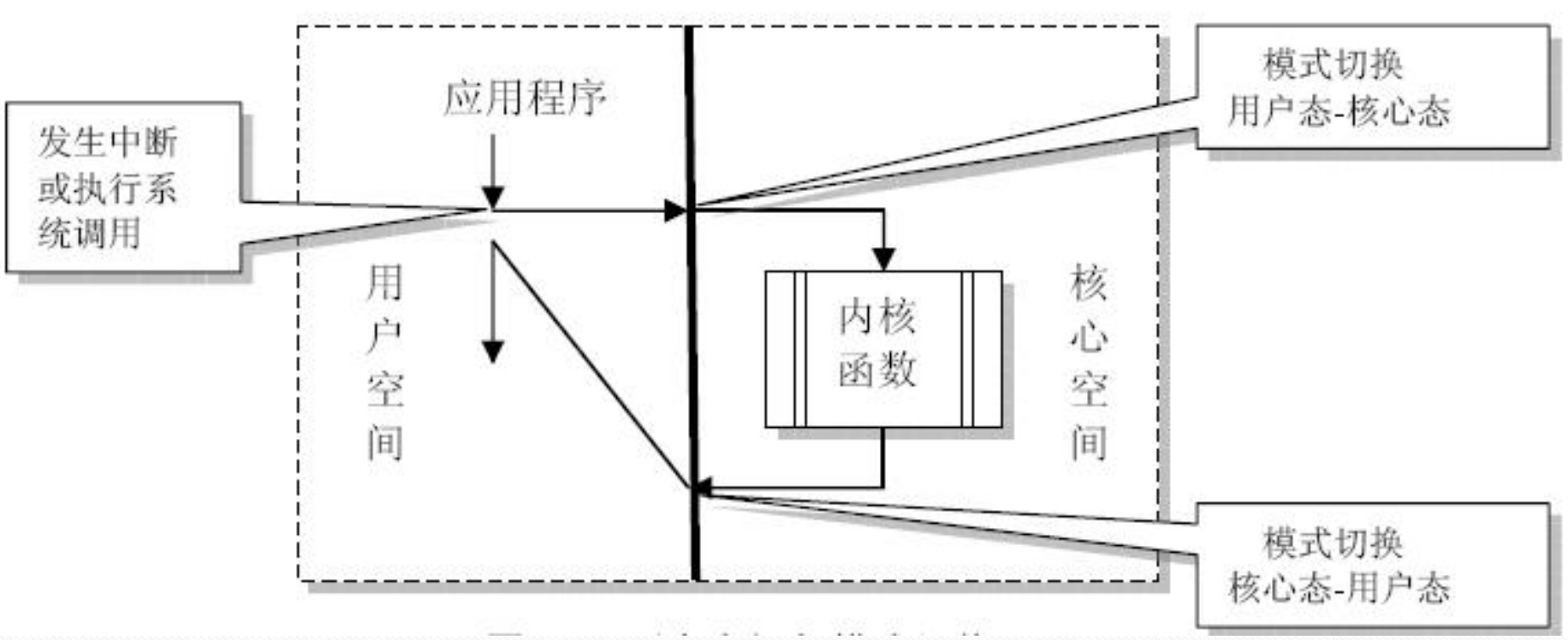
Figure 1-17. The 11 steps in making the system call read(fd, buffer, nbytes).

Mordern Operating Systems (4th edition) pp.52 Chap1 Fig.1-17

API – System Call – OS Relationship

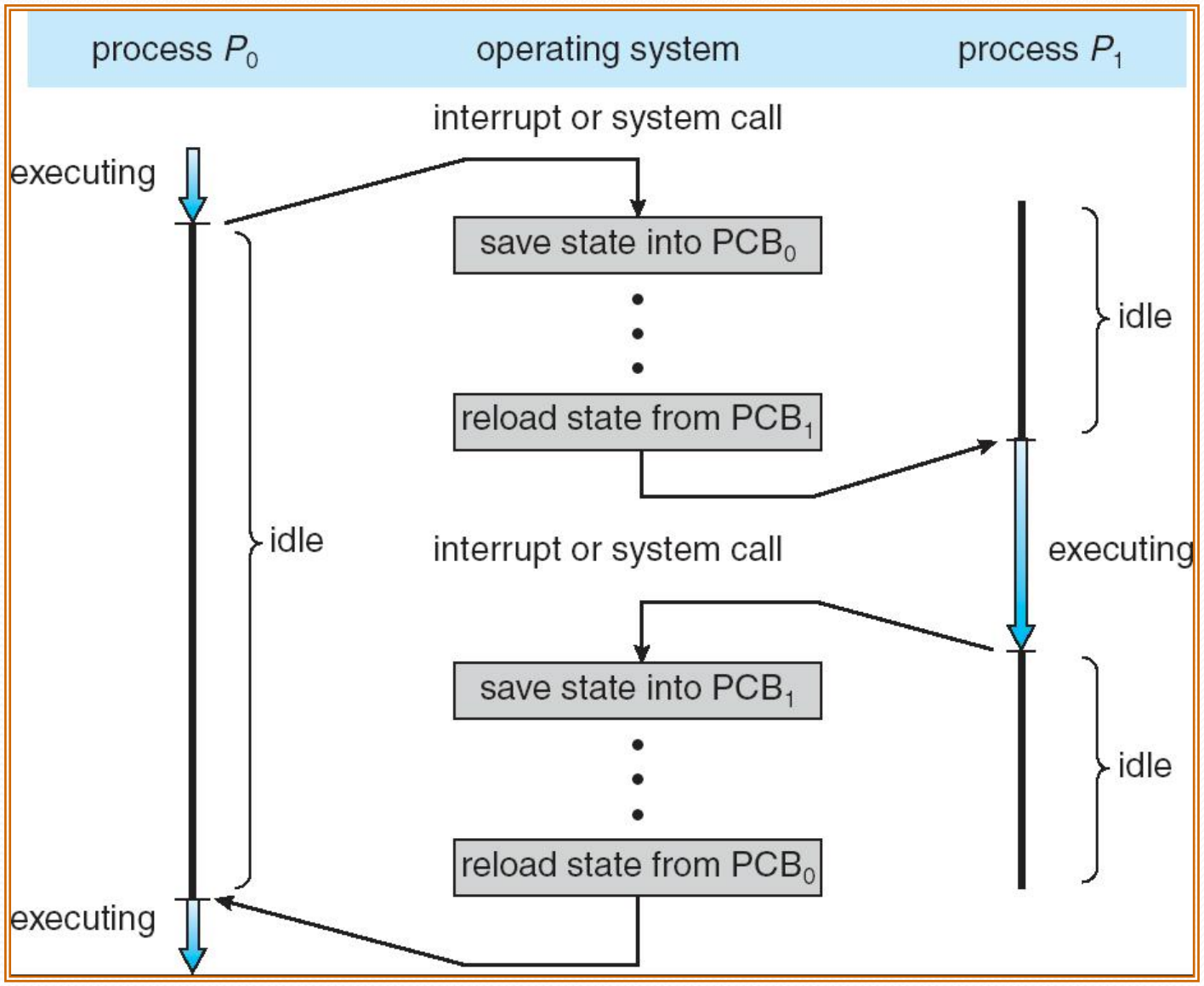


两个空间与模式切换



两个空间与模式切换

CPU Switch From Process to Process



Transition from user to kernel mode

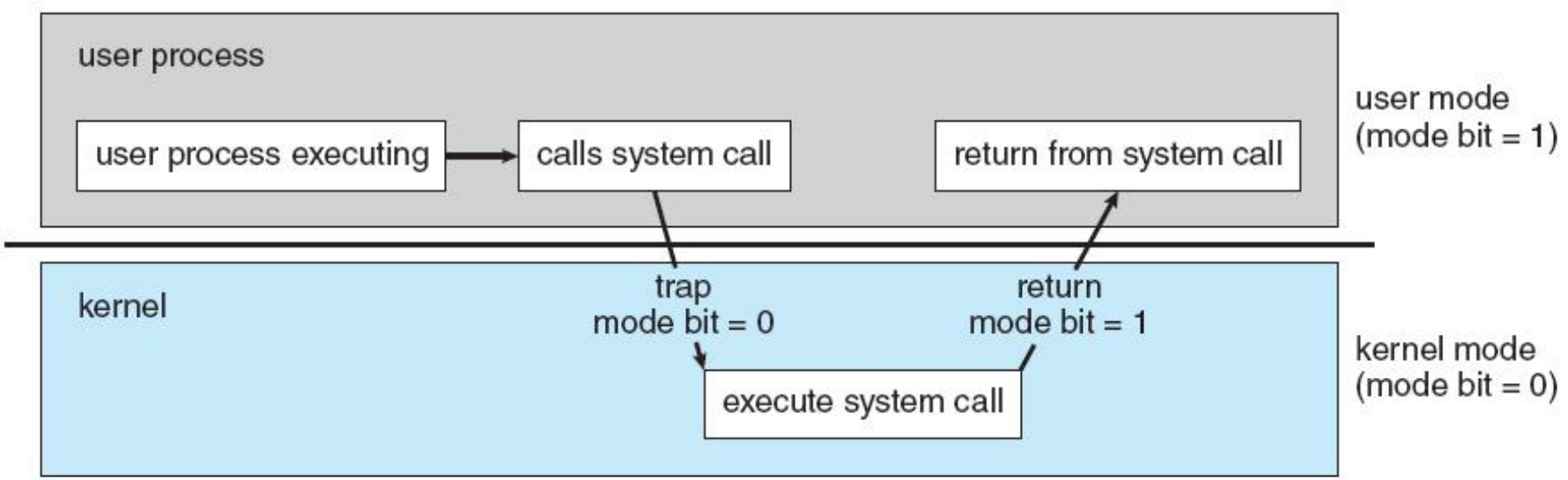


Figure 1.13 Transition from user to kernel mode.

书教材 Operating System Concepts (10th), pp.25