System Calls

Material by Mark Handley & A. Tanenbaum

Exception Handling

		Exception Vectors	Exception Handlers

Save PC and PSW

Load PC from Exception Vector

CPU switches to kernel-mode
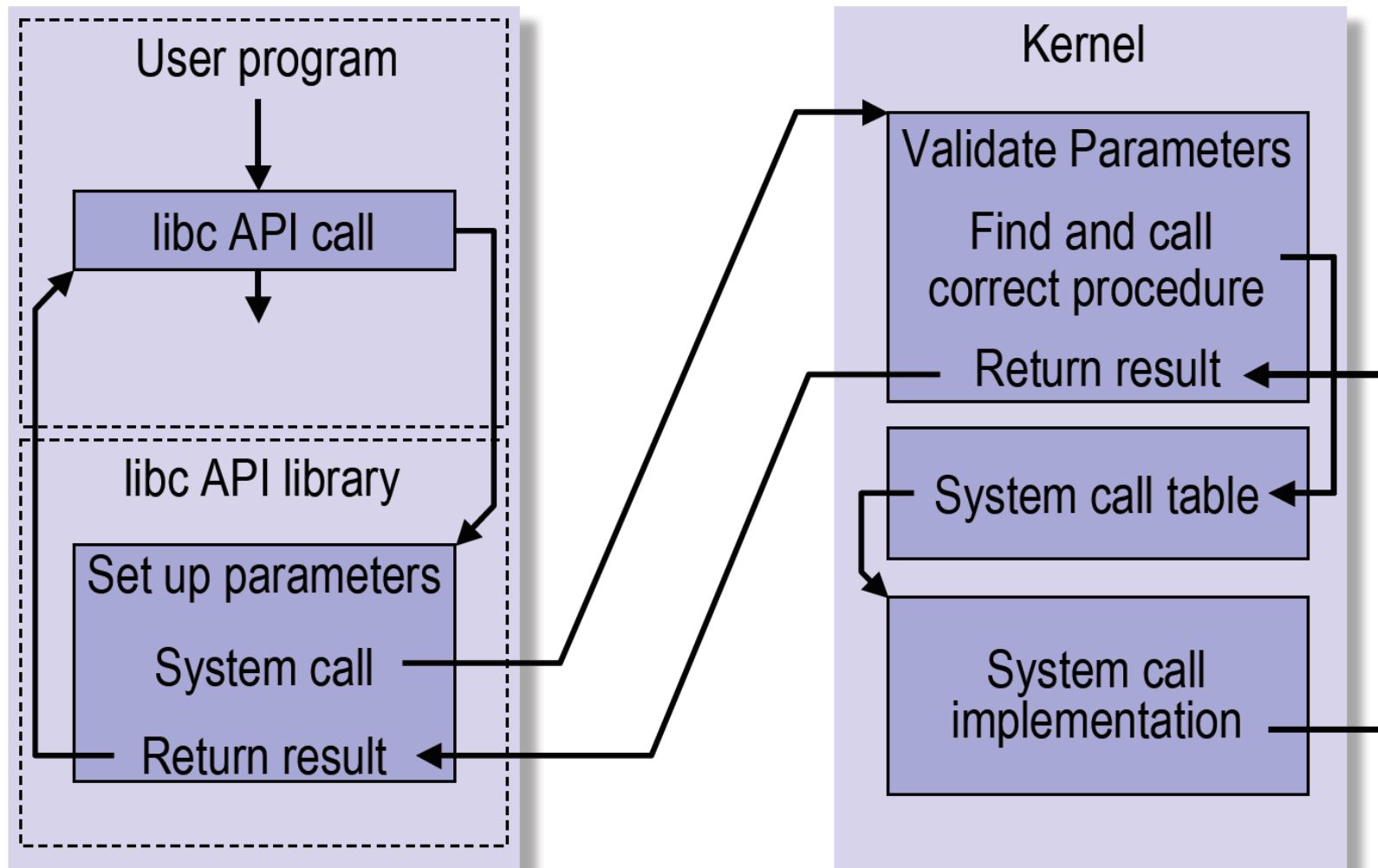
Use PC to fetch next instruction

code to handle exception A

code to handle exception B

code to handle exception C
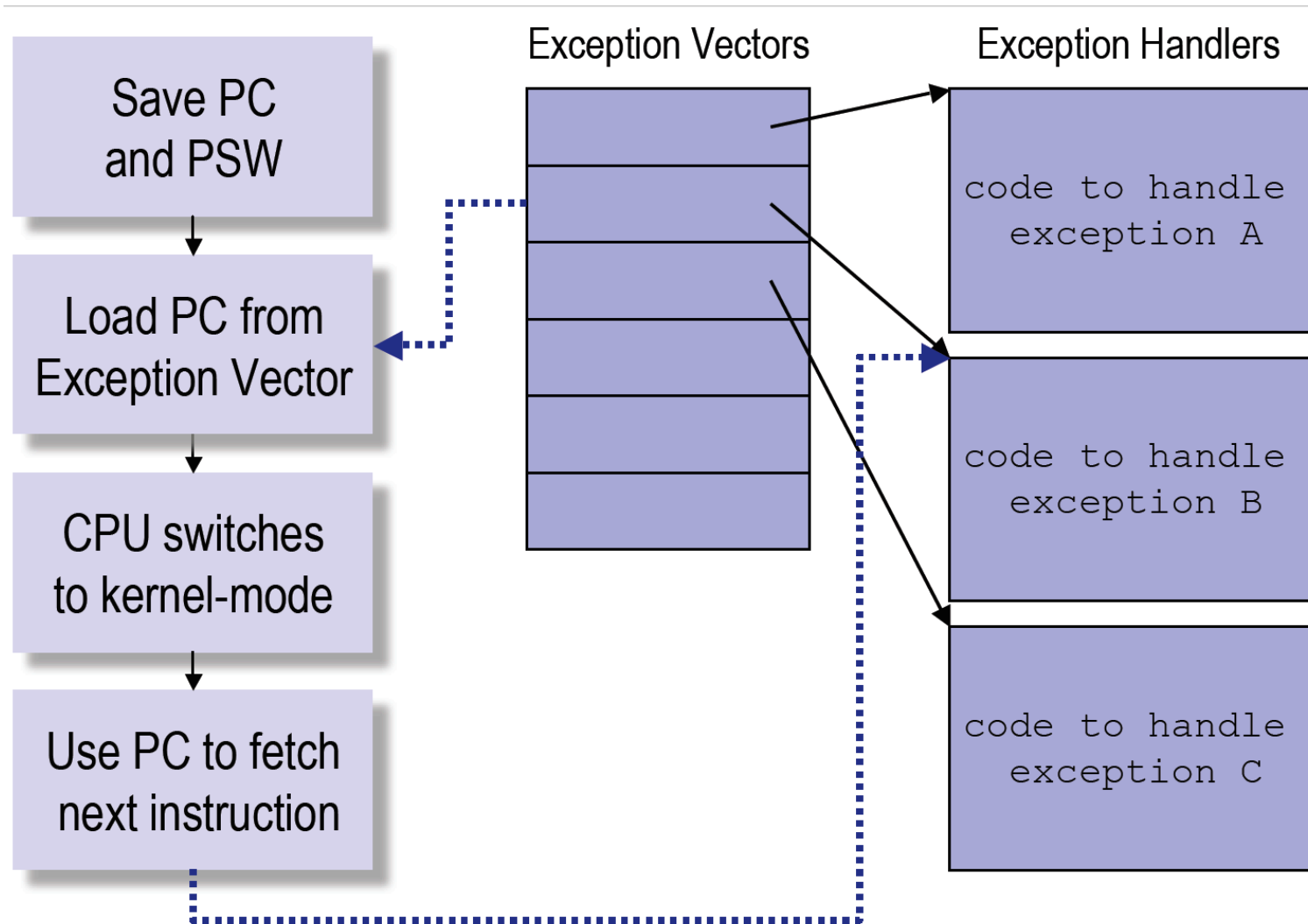
Divide-by-zero, invalid instruction, …

User Exception Handling

Execution by
user process

User Program

OS Exception Handler

Exception
occurs

Exception
handler called

Exception
handler can
pass control
back to where
the exception
occurred

Control passed to
user program's
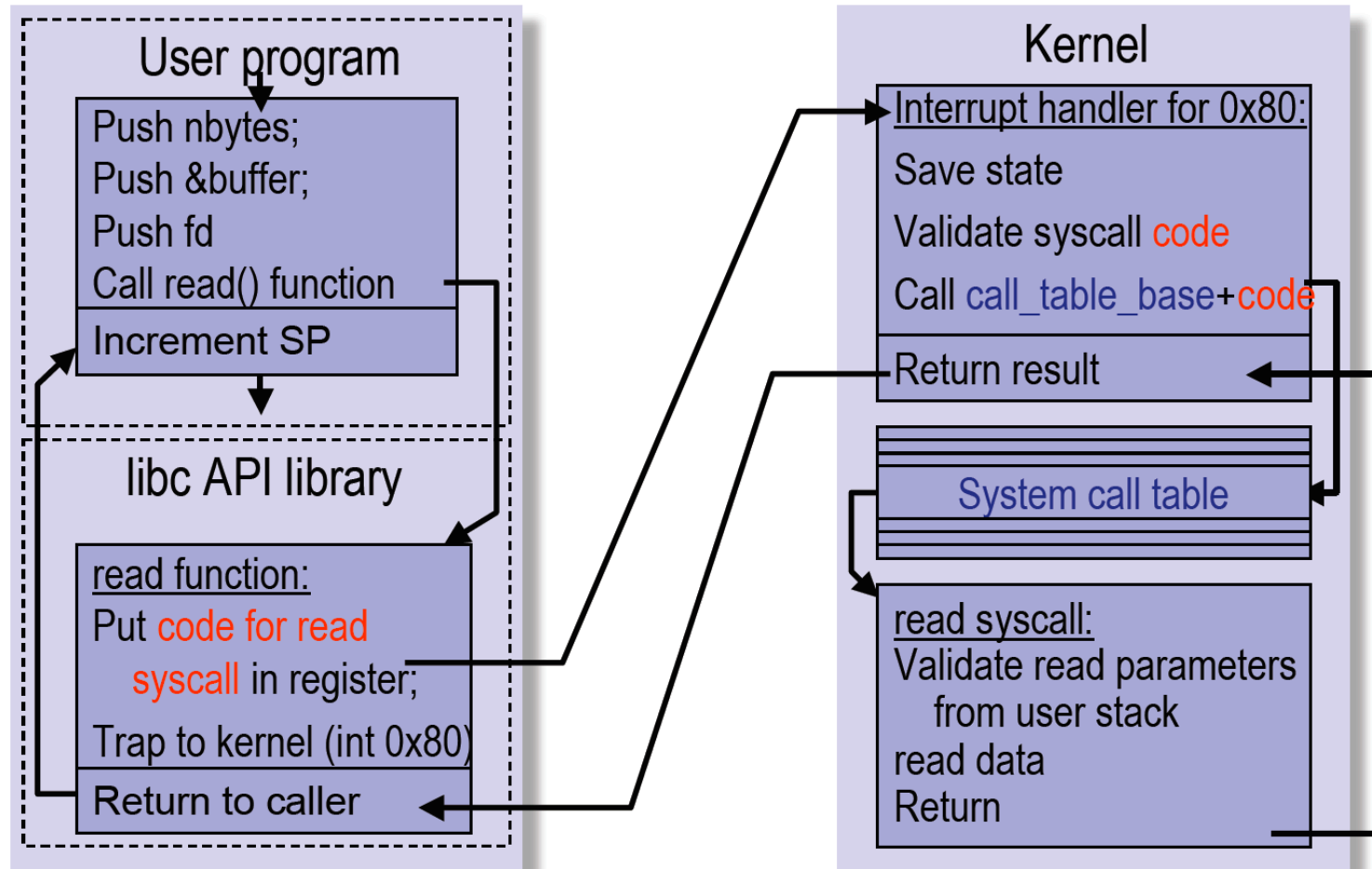exception handler
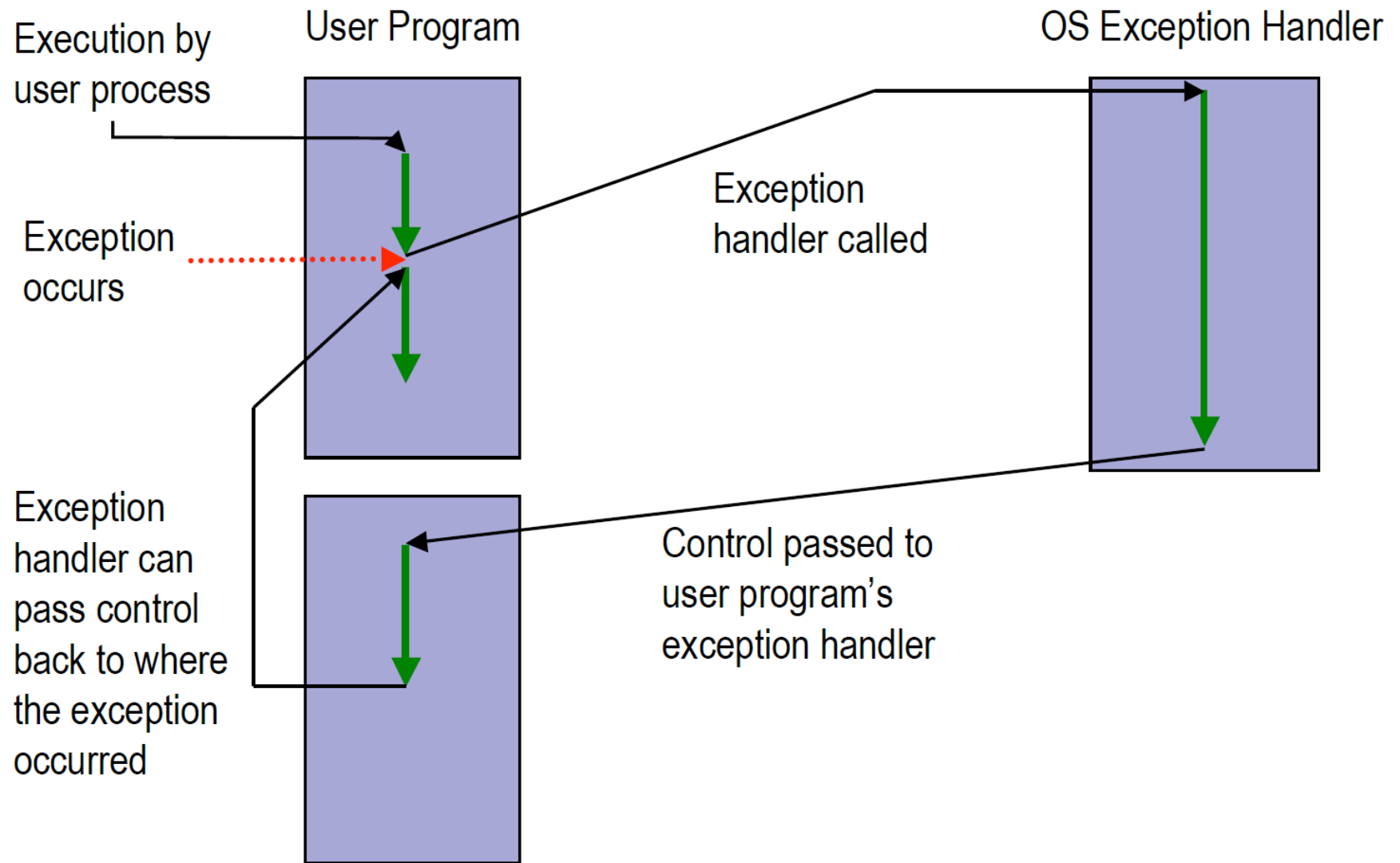
Processing a System Call

Example System Call
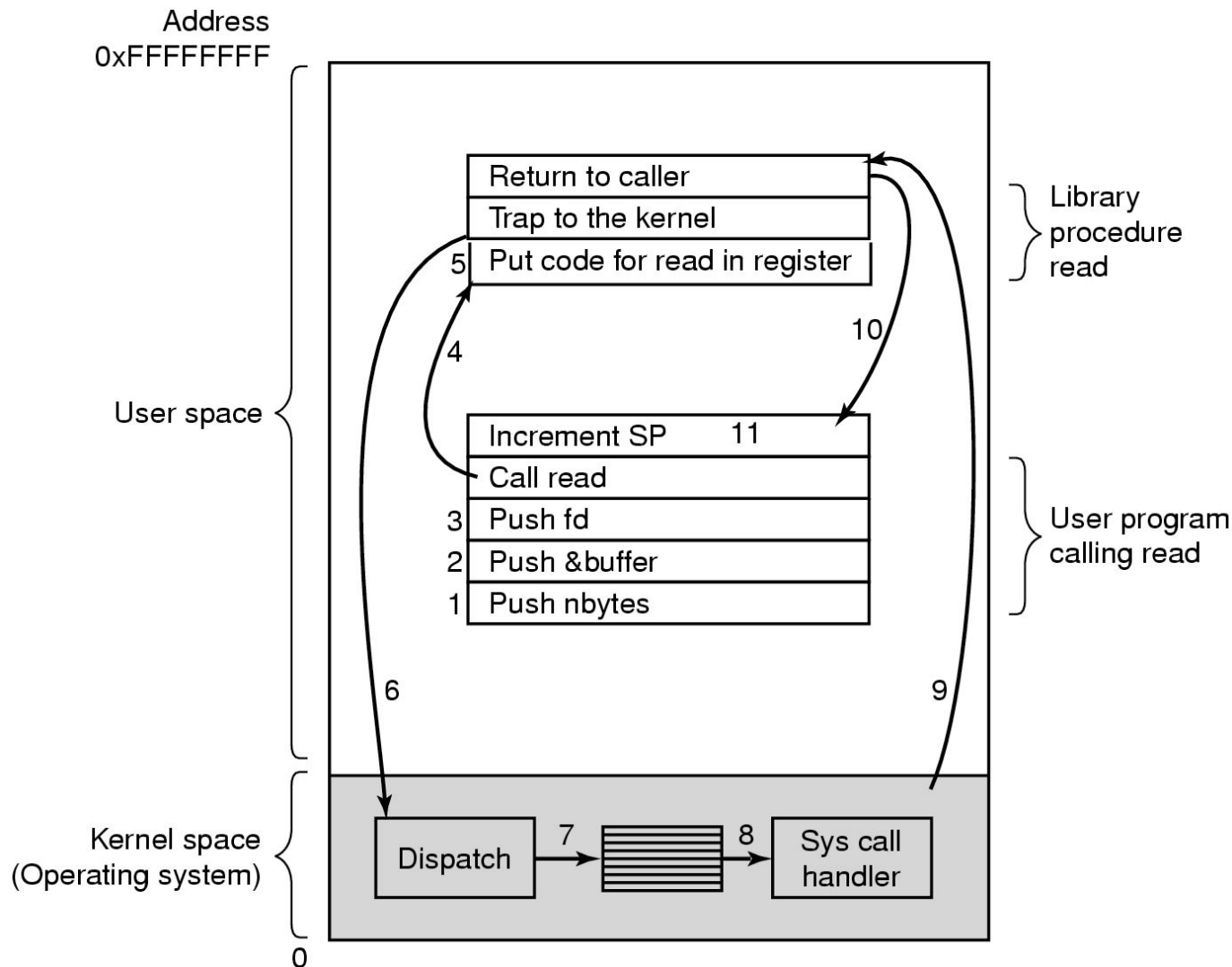
- Call from C program:

count = read(fd, &buffer, length);

  – read up to length bytes of data from from file
  – descriptor fd into buffer, and returns the number of
  – bytes actually read, or -1 if an error occurred.
  – C library call for read() calls the read system call.

Steps in Making a System Call

Steps in Making a System Call

Address
0xFFFFFFFF

User space

Kernel space
(Operating system)

0

Return to caller

Trap to the kernel

5 | Put code for read in register

Library
procedure
read

10

4

Increment SP   11

Call read

3 | Push fd

2 | Push &buffer

1 | Push nbytes

User program
calling read

6

9

Dispatch   7   8   Sys call
handler

There are 11 steps in making the system call
read (fd, buffer, nbytes)

7