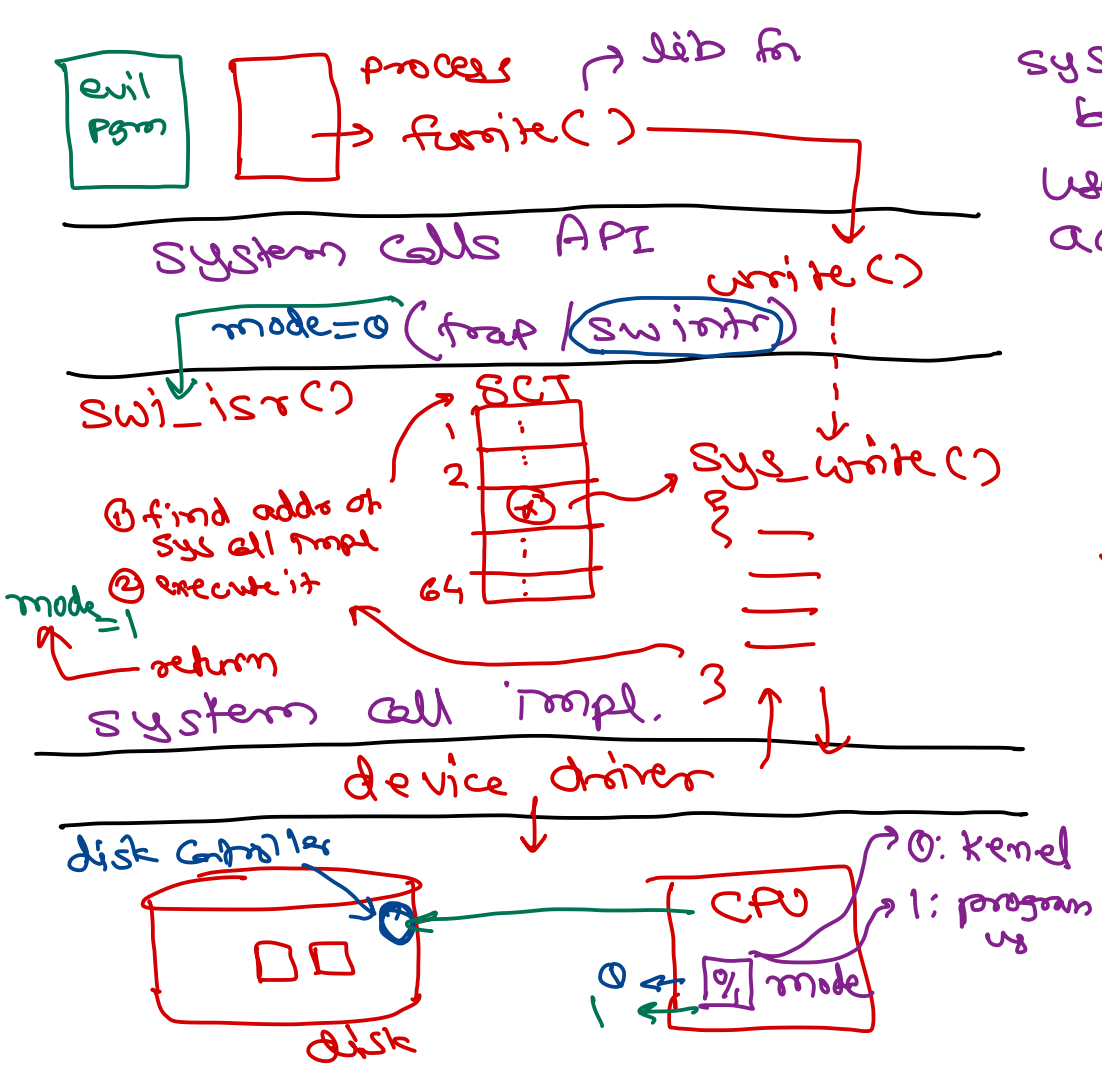
Hardware Protection

- Early operating systems work as resident monitors.
- Then OS start doing additional jobs like I/O, resource allocator, etc.
- In multiprogramming environment, one program could disturb other program in memory by corrupting its data.
- The programming errors are detected by hardware and conveyed to operating system via interrupt. OS should take appropriate action like terminating victim program.
- The following protection mechanisms are available:
  - Dual-Mode Operation
  - I/O Protection
  - Memory Protection
  - CPU Protection



Syscalls are fns exposed by the kernel, so that user programs can access kernel functionality

UNIX : 64 sys calls.

Linux : >300 sys calls.

Software interrupt

a special assembly instruction (arch dependent) → cause execution of ISR.

x86 : INT

ARM : SWI / SVC

(*) mode bit is in (modern) CPU.

when mode bit is 0 → kernel code is running

" " " is 1 → user code is running

(*) mode = 0 → kernel / system / monitor / privileged

mode = 1 → user / non-privileged.

(*) privileged mode special instns are also allowed

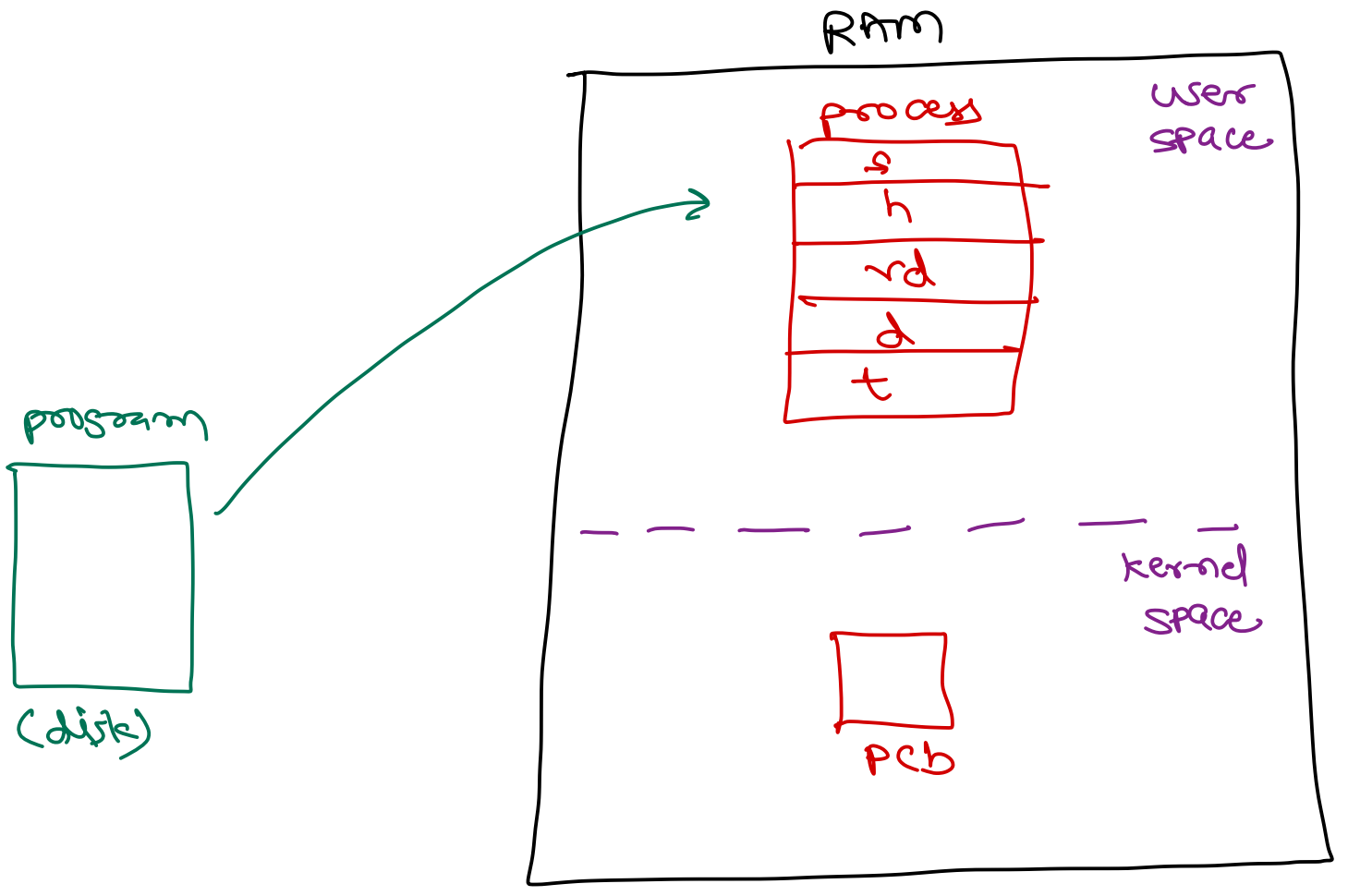
e.g. IO.

non-privileged mode allows only general purpose instructions.

(*) In hw, when interrupt arrives, mode = 0.

when ISR is completed, mode = 1.

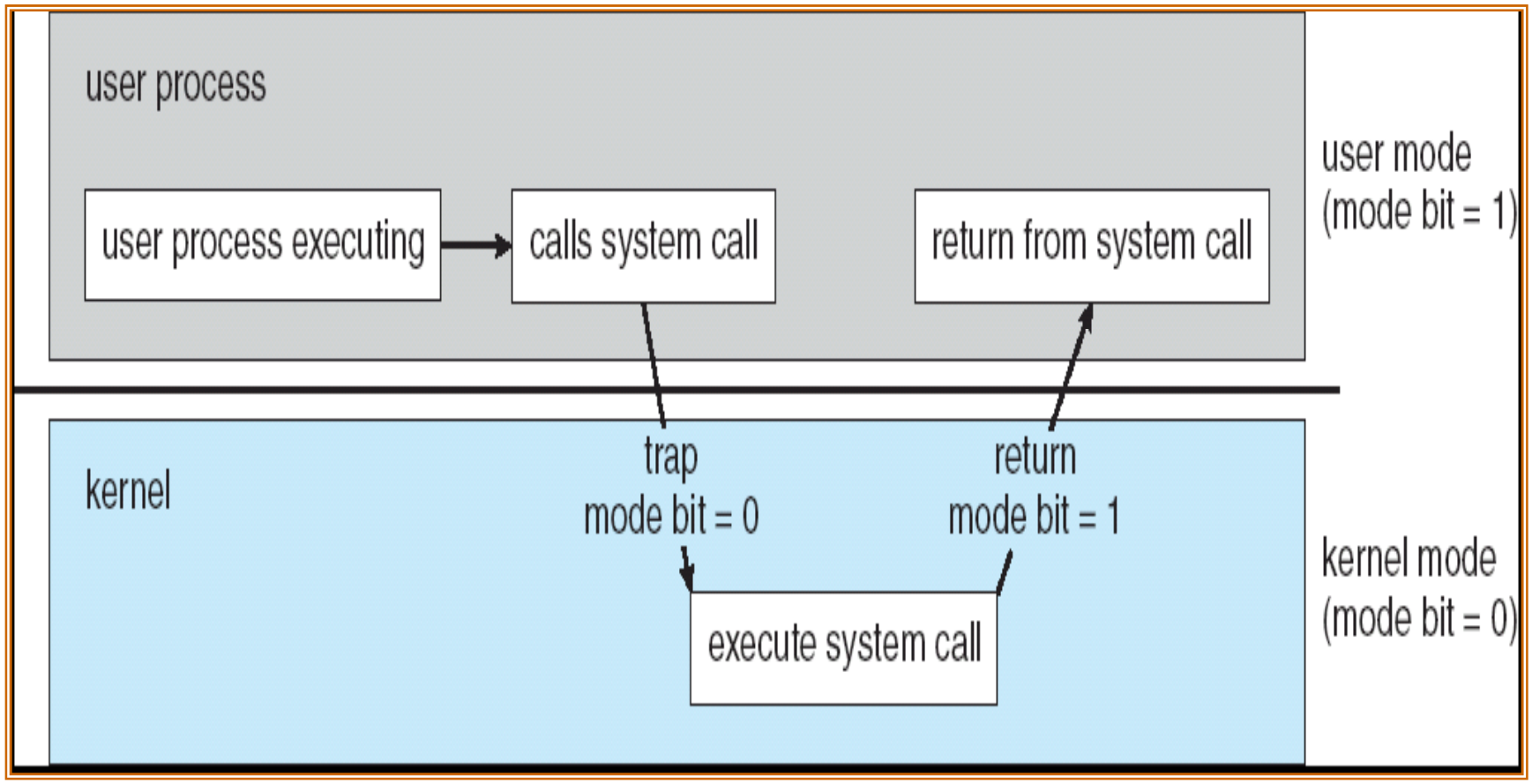
(*) Logical part of RAM in which OS code/data is kept, is called as kernel space. Rest all area in which user programs are executed, is called as user space.


RAM
process
user space
s
h
rd
d
t
program
(disk)
kernel space
PCB

Dual-Mode Operation

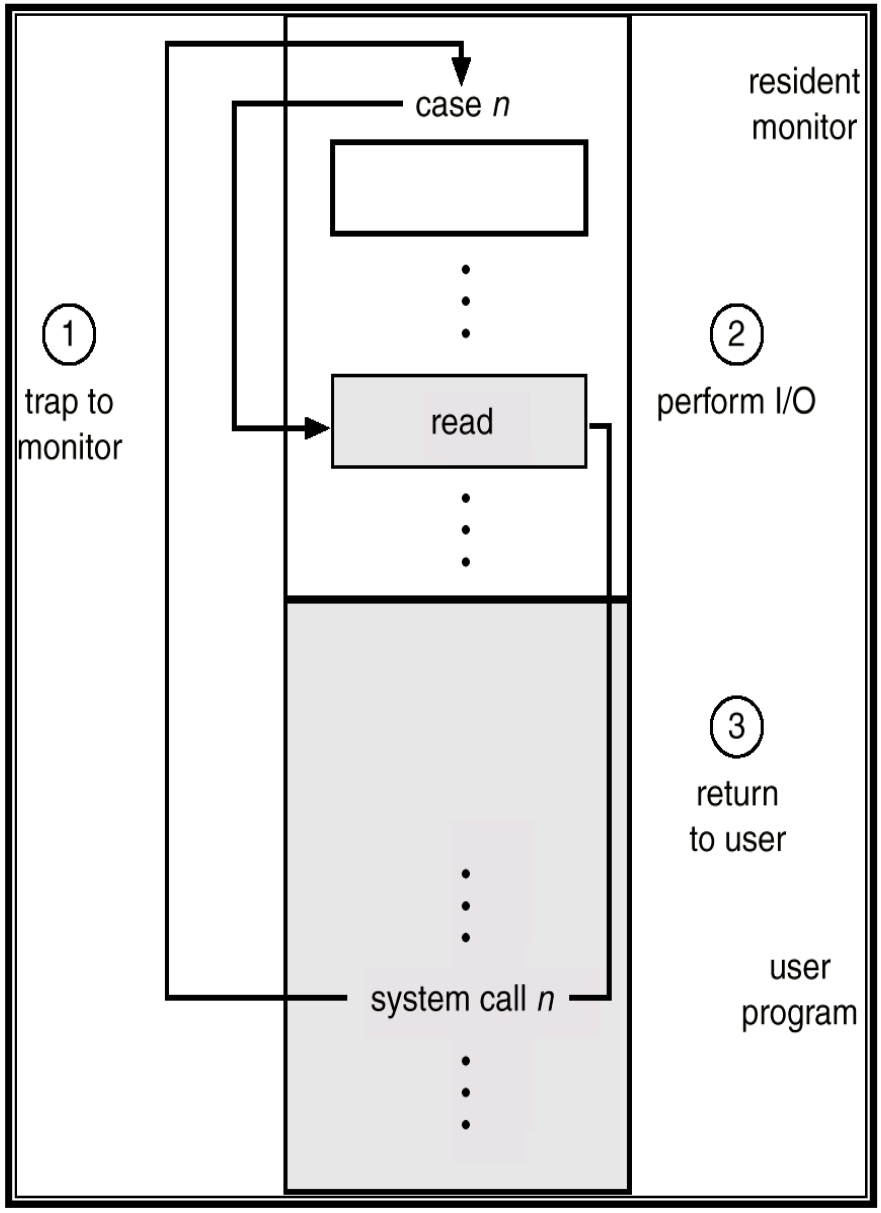
- Sharing system resources requires operating system to ensure that an incorrect program cannot cause other programs to execute incorrectly.
- Provide hardware support to differentiate between at least two modes of operations.
  - User mode – execution done on behalf of a user.
  - Monitor mode (also kernel mode or system mode) – execution done on behalf of operating system.
- Mode bit added to computer hardware to indicate the current mode: monitor (0) or user (1).
- When an interrupt or fault occurs hardware switches to monitor mode.

User mode and Kernel mode



I/O Protection

- All I/O instructions are privileged instructions.
- Since all I/O operations are done through IVT, it must be protected from user programs.
- I/O must be done via system calls.
- Must ensure that a user program could never gain control of the computer in monitor mode.



Thank you!

Source: Galvin OS books/slides

Edited by: Nilesh Ghule