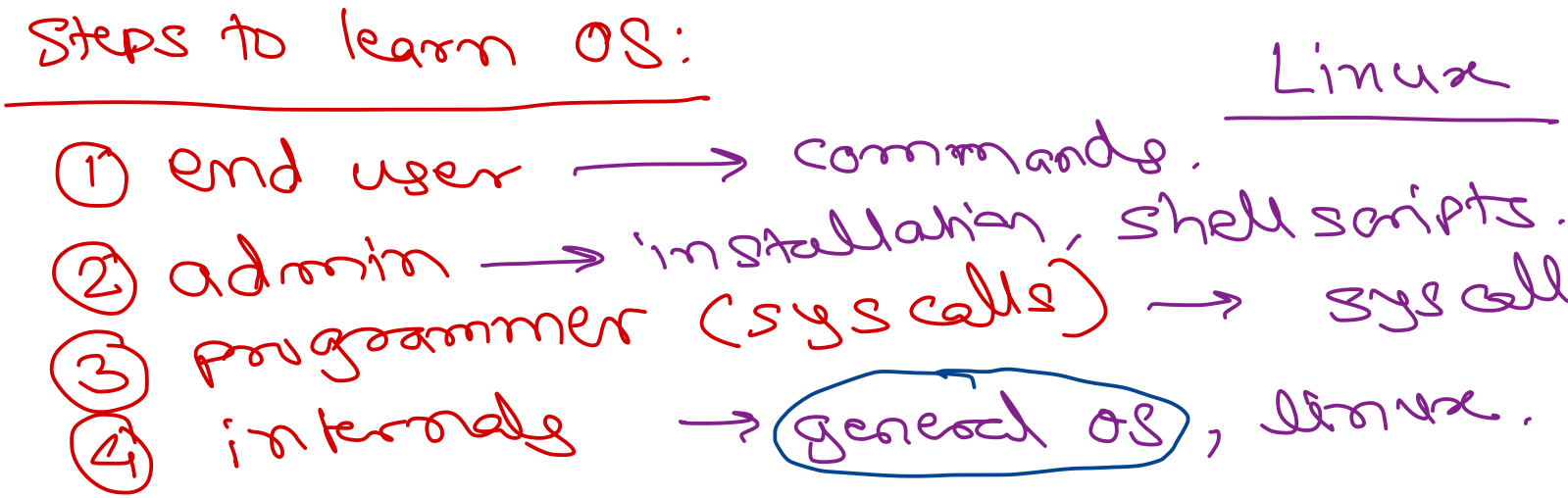
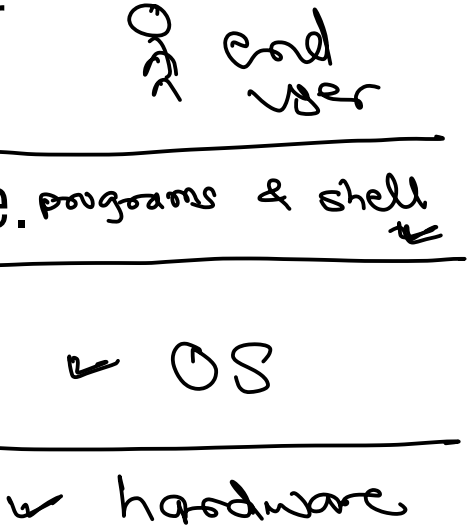
Operating System Introduction

Sunbeam Infotech



What is an Operating System?

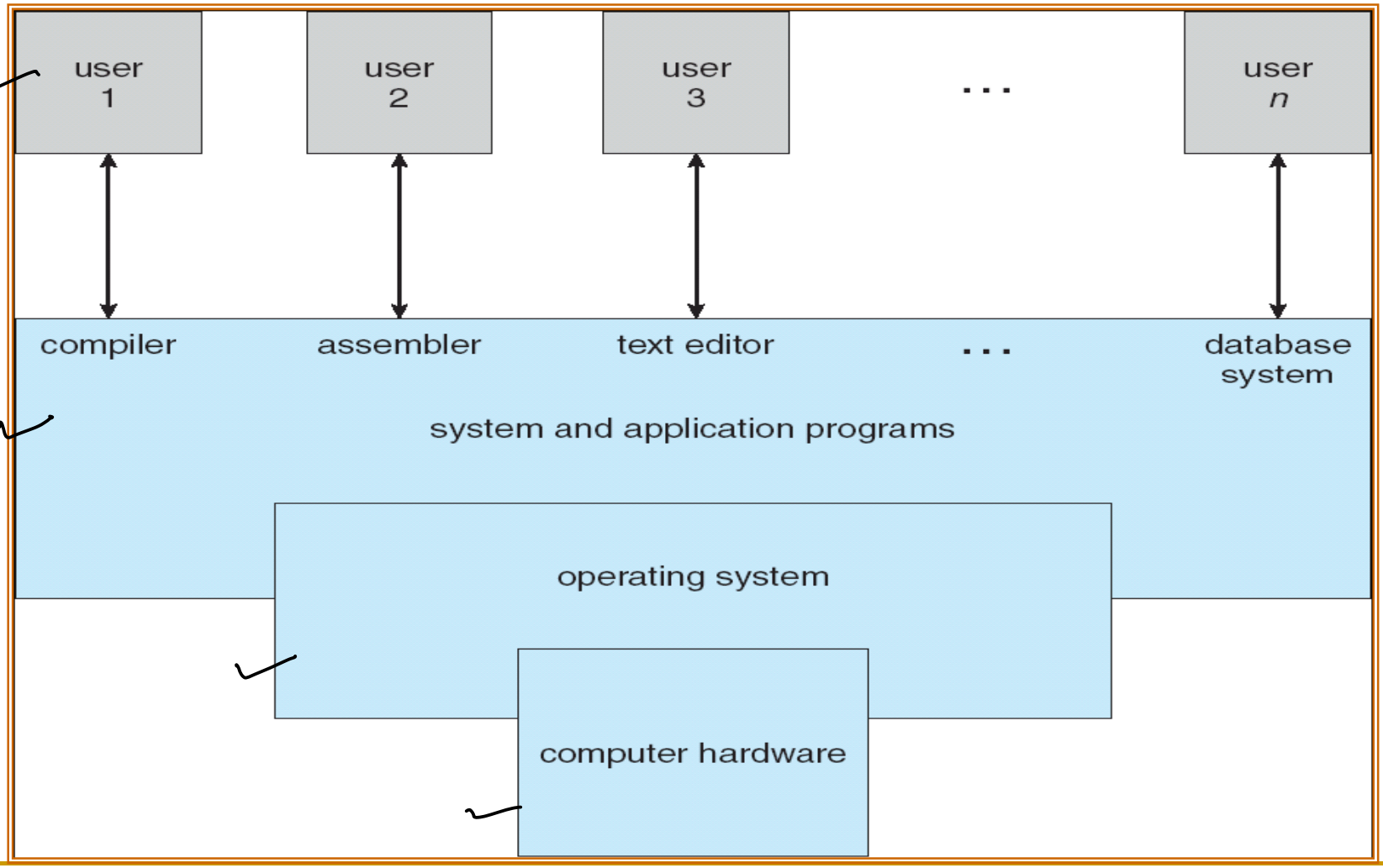
- A program that acts as an intermediary between a user of a computer and the computer hardware.
- Operating system goals:
  - Execute user programs and make solving user problems easier.
  - Make the computer system convenient to use.
- Use the computer hardware in an efficient manner.



Computer System Structure

- Hardware – provides basic computing resources
  - CPU, memory, I/O devices
- Operating system
  - Controls and coordinates use of hardware among various applications and users
- Application programs – define the ways in which the system resources are used to solve the computing problems of the users
  - Word processors, compilers, web browsers, database systems, video games , shell
- Users
  - People, machines, other computers

Four Components of Computer System



Operating System Definition

windows : ntoskrnl.exe
linux: vmlinuz

- OS is a resource allocator
  - Manages all resources and
  - Decides between conflicting requests for efficient and fair resource use
- OS is a control program
  - Controls execution of programs to prevent errors and improper use of the computer
- Everything a vendor ships when you order an operating system

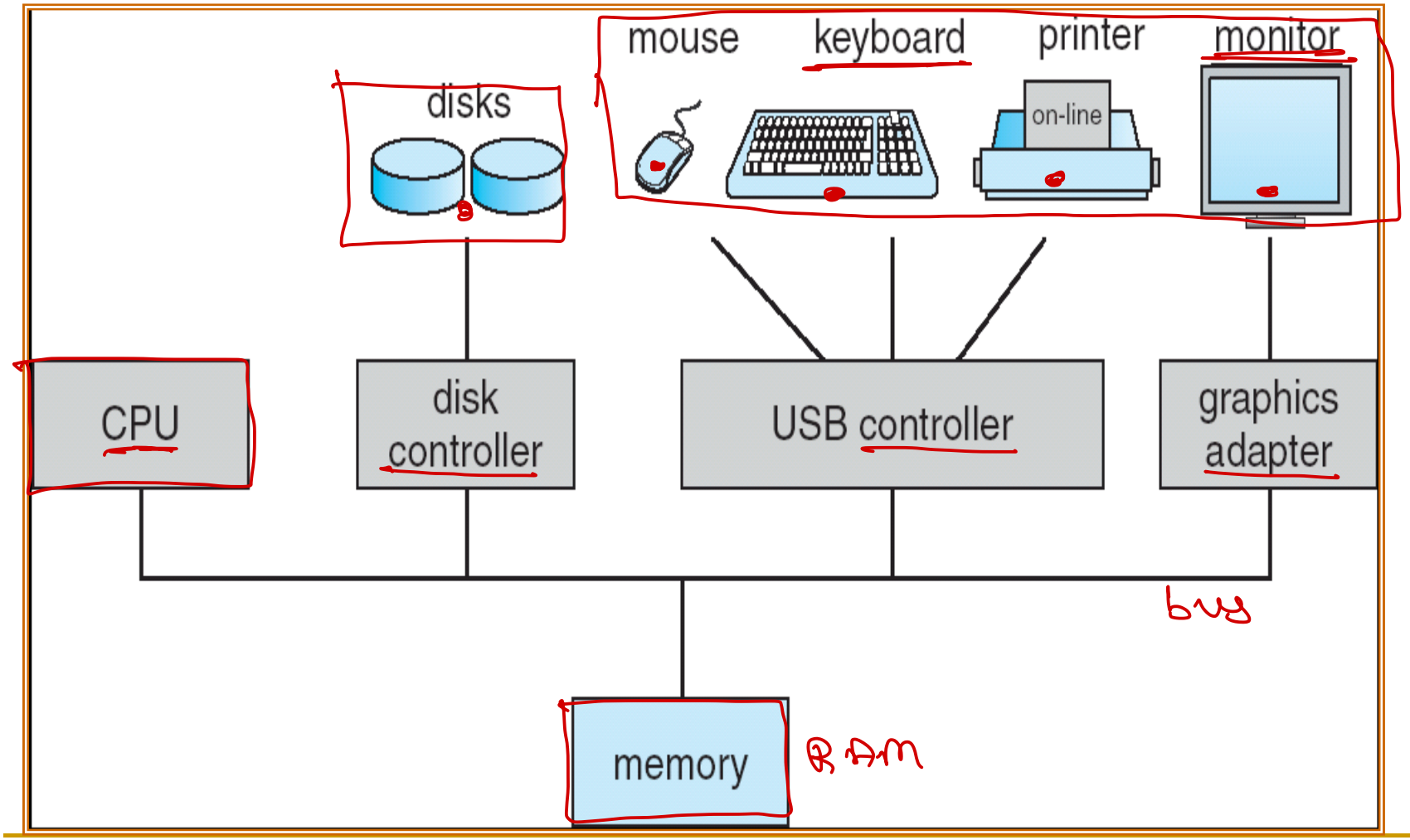
CD/DVD = Core OS + programs + utilities

- "The one program running at all times on the computer" is the kernel. Everything else is either a system program (ships with the operating system) or an application program

Computer System

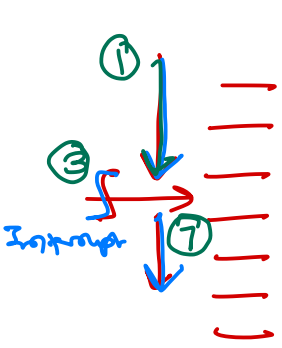
Sunbeam Infotech

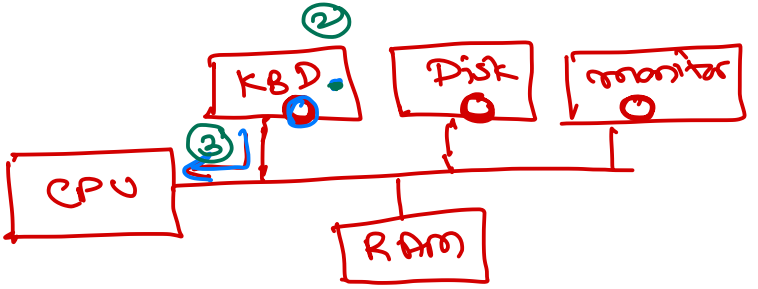
Computer System Structure



Computer-System Operation

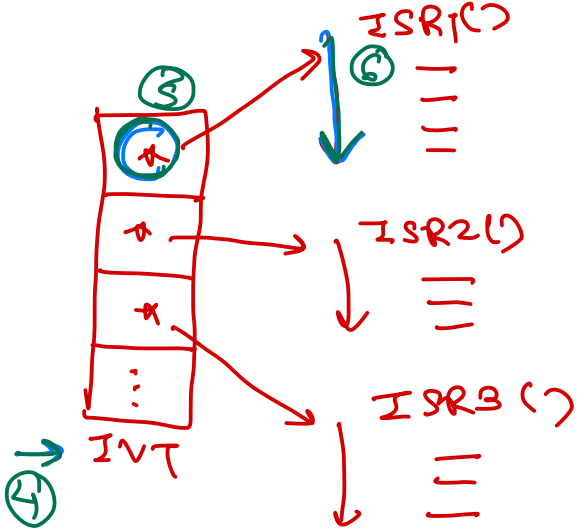
- One or more CPUs, device controllers connect through common bus providing access to shared memory.
- I/O devices and the CPU can execute concurrently.
- Each device controller is in charge of a particular device type and has a local buffer.
- CPU moves data from/to main memory to/from local buffers.
- Actual input and output occurs between the device to local buffer of controller.
- Device controller informs CPU that it has finished its operation by causing an interrupt.



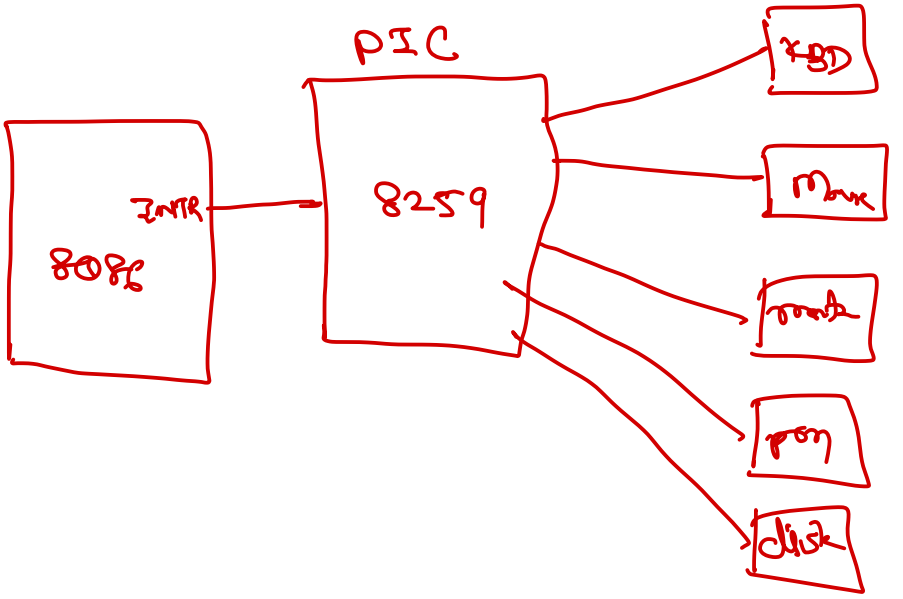


① polling (sync IO)

② interrupt (async IO)



8086 arch



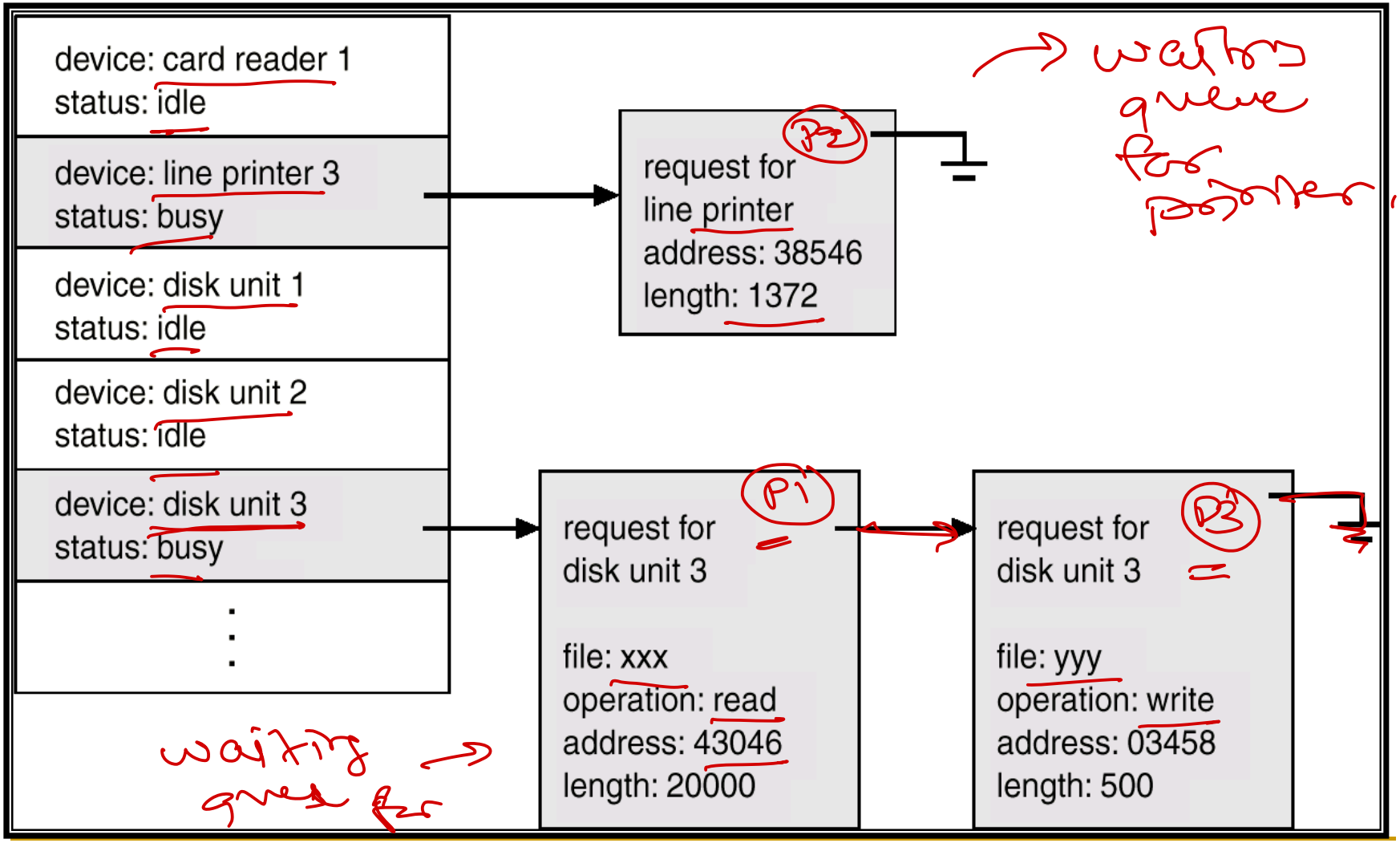
Common Functions of Interrupts

- Interrupt transfers control to the interrupt service routine generally, through the interrupt vector, which contains the addresses of all service routines.
- Interrupt architecture must save the address of the interrupted instruction. The OS preserves the state of the CPU by storing registers and the program counter.
- Incoming interrupts are disabled while another interrupt is being processed to prevent a lost interrupt.
- A trap is a software-generated interrupt caused either by an error or a user request.
- An operating system is interrupt driven.

I/O Structure

- In synchronous I/O method, after I/O starts, control returns to user program only upon completing I/O
  - wait instruction or loop idles the CPU until the next interrupt
  - At most one I/O request is outstanding at a time, no simultaneous I/O processing.
- In asynchronous method, after I/O starts, control return to user program without waiting for I/O completion
  - System call – request to the operating system to allow user to wait for I/O completion.
  - Device-status table contains entry for each I/O device indicating its type, address, and state.
  - Operating system indexes into I/O device table to determine device status and to modify table entry to include interrupt.

Device-Status Table



Operating System Concepts

Source: Galvin OS books/slides

Edited by: Nilesh Ghule