CS2842 Computer Systems – Lecture VIII

I/O and Peripheral Devices

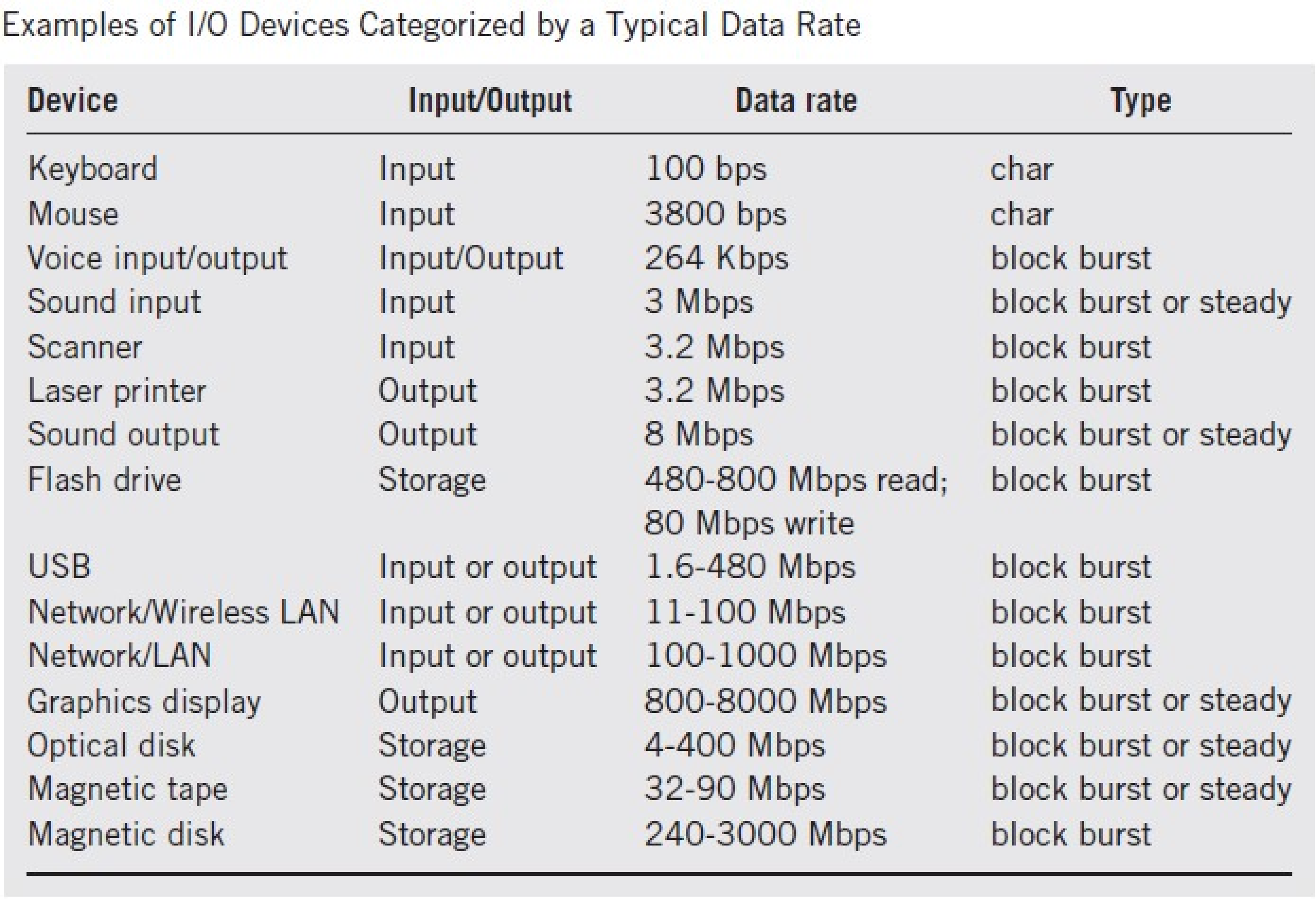
Dr. Sapumal Ahangama
Department of Computer Science and Engineering

TYPICAL I/O DEVICES

- Keyboard
- Mouse
- Printers
- Display Screens
- Disk Drives

- Peripherals Devices – Input, Output and Storage

TYPICAL I/O DEVICES



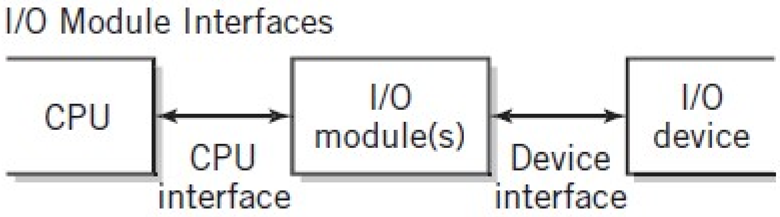
Examples of I/O Devices Categorized by a Typical Data Rate

Device	Input/Output	Data rate	Type
Keyboard	Input	100 bps	char
Mouse	Input	3800 bps	char
Voice input/output	Input/Output	264 Kbps	block burst
Sound input	Input	3 Mbps	block burst or steady
Scanner	Input	3.2 Mbps	block burst
Laser printer	Output	3.2 Mbps	block burst
Sound output	Output	8 Mbps	block burst or steady
Flash drive	Storage	480-800 Mbps read; 80 Mbps write	block burst
USB	Input or output	1.6-480 Mbps	block burst
Network/Wireless LAN	Input or output	11-100 Mbps	block burst
Network/LAN	Input or output	100-1000 Mbps	block burst
Graphics display	Output	800-8000 Mbps	block burst or steady
Optical disk	Storage	4-400 Mbps	block burst or steady
Magnetic tape	Storage	32-90 Mbps	block burst or steady
Magnetic disk	Storage	240-3000 Mbps	block burst

DESIGN CONSIDERATIONS FOR I/O

- It is not practical to directly connect CPU with I/O device
  - Formats of different devices vary
  - Incompatible speed
  - Specialized control requirements (Ex: disk drives)

- I/O modules



PROGRAMMED I/O

- I/O module is directly connected to registers of the CPU
- Full instruction fetch-execute cycle for each and every I/O data word being transferred
  - Slow execution
  - Used in Keyboard

POLLING

- Checking for each input in rotation at frequent intervals
  - Example: Keyboard?
  - Hundreds of I/O devices?

INTERRUPTS

- Interrupt the normal flow of a program in the computer to react to special events

- Interrupt lines
  - Special control lines to the CPU
  - For example, as many as 32 interrupt lines (IRQ0 –IRQ31)
  - Messages sent on the lines are known as interrupts

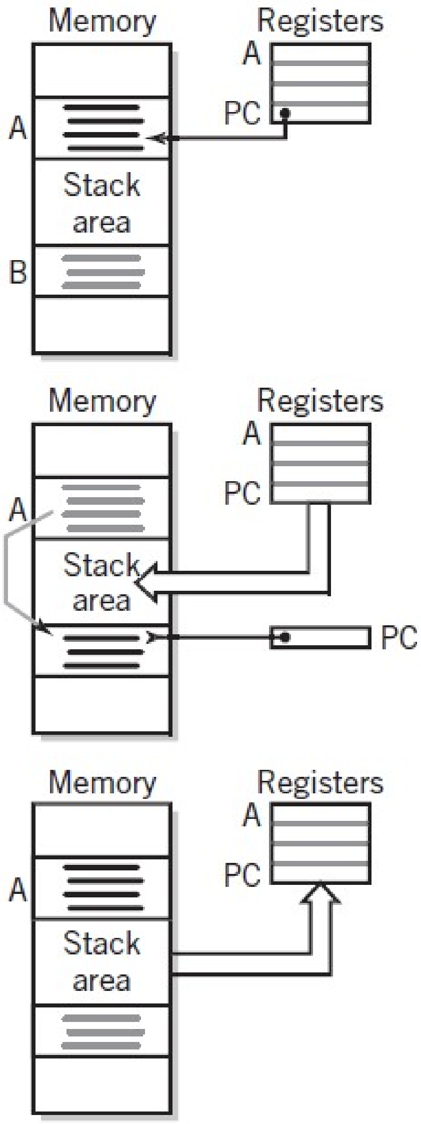
- Example: Keyboard

SERVICING INTERRUPTS

- Temporary suspension of the program in process when an interrupt signal arrive
  - Suspend the current program
  - Saved in the Stack Memory
  - CPU branches to **interrupt handler** program
  - Servicing the interrupt!
  - Restore and resume original program
  - Interrupt may alter original program

SERVICING INTERRUPTS

Servicing an Interrupt

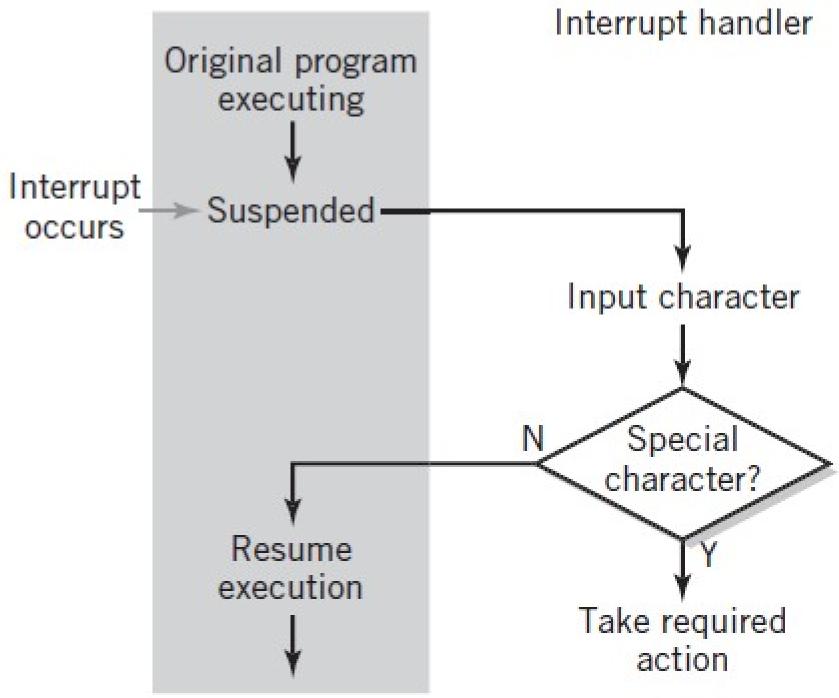


1. Before interrupt arrives, program A is executing. The program counter points to the current instruction.
2. When the interrupt is received by the CPU, the current instruction is completed, all the registers are saved in the stack area (or in a special area known as a process control block). The PC is loaded with the starting location of program B, the interrupt handler program. This causes a jump to program B, which becomes the executing program.
3. When the interrupt routine is complete, the registers are restored, including the program counter, and the original program resumes exactly where it left off.

USES OF INTERRUPTS

- External event notifier
  - Example: Keyboard Input

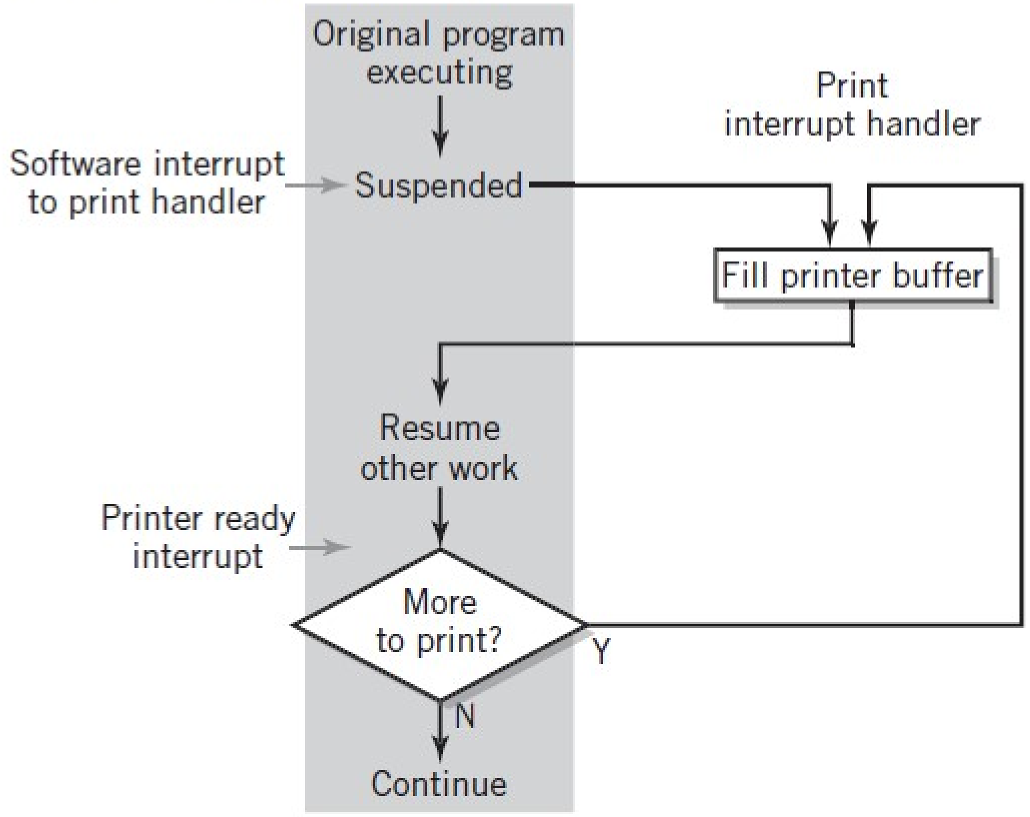
Using a Keyboard Handler Interrupt



USES OF INTERRUPTS

- Completion Signal
  - Example: Printer

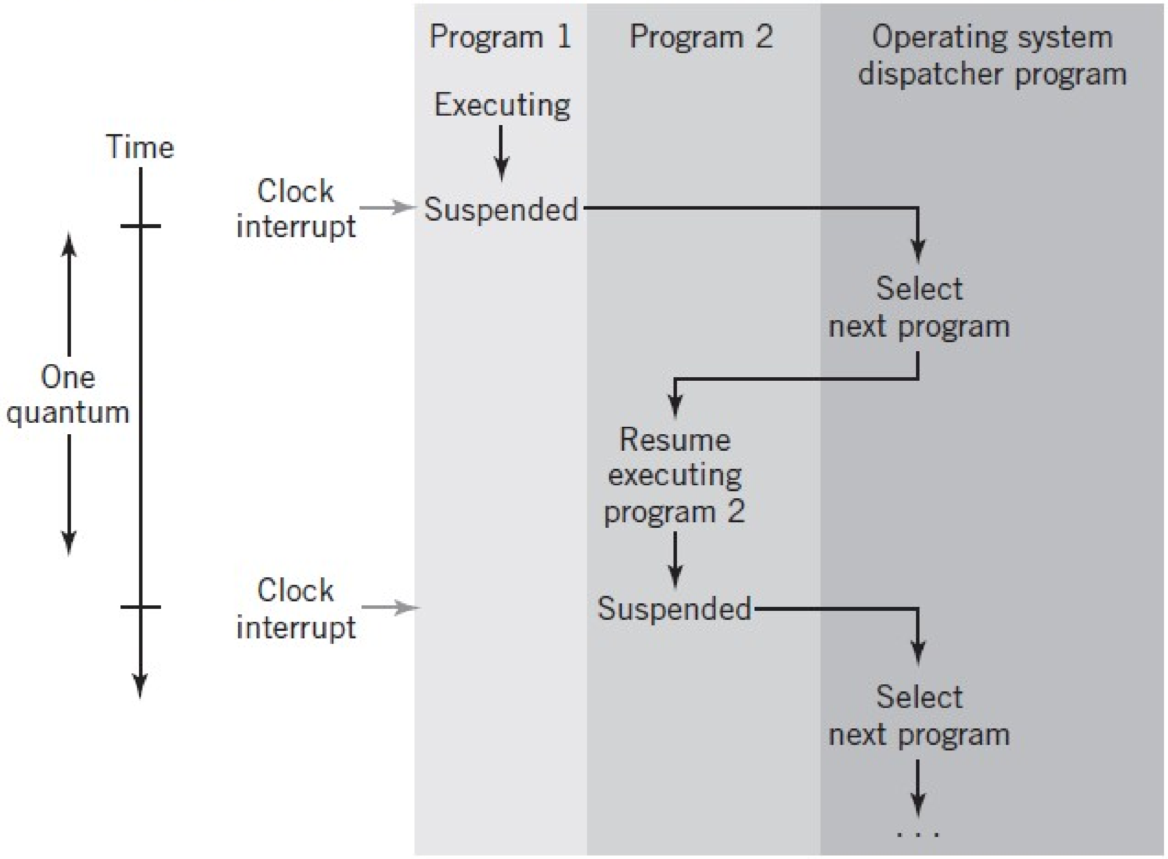
Using a Print Handler Interrupt



USES OF INTERRUPTS

- Allocating CPU time
  - Method of allocating CPU time for different programs or threads sharing the CPU
  - Dispatcher

Using an Interrupt for Time Sharing



USES OF INTERRUPTS

- Abnormal event indicator
  - For the computer to respond to specific course of action during an abnormal event
  - Generated from inside of CPU
  - Example: power failure, divide by 0, nonexistent op code, hardware errors

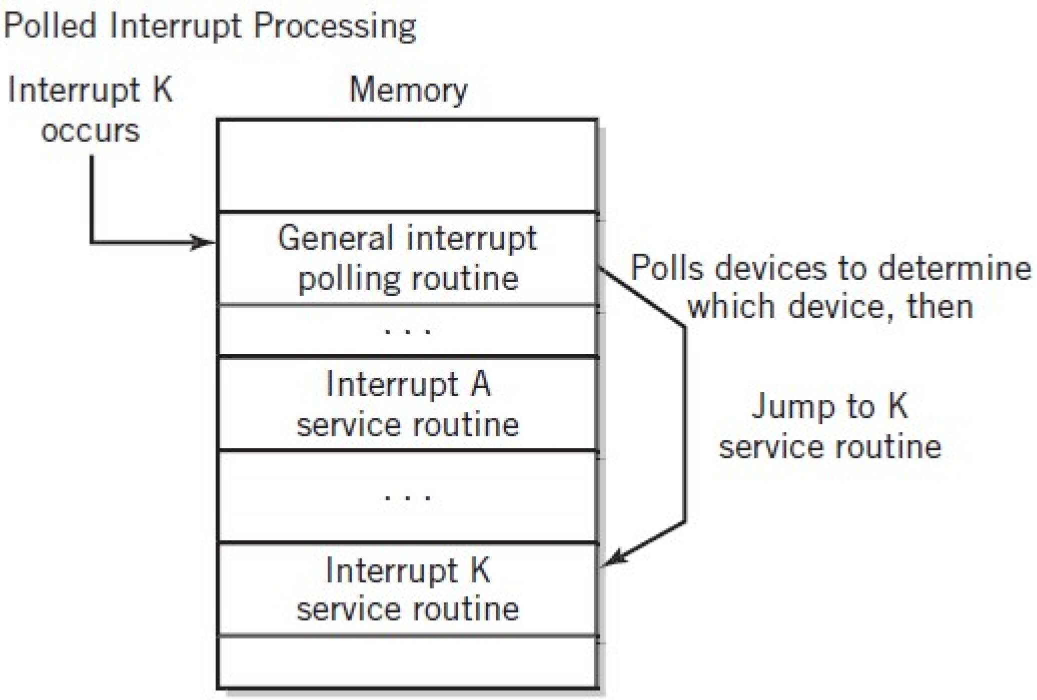
USES OF INTERRUPTS

- Software interrupts
  - CPUs include instructions that simulate an interrupt
  - Example: Software requesting I/O

MULTIPLE INTERRUPTS

- Multiple interrupts can arrive at the same time
  - Priority
  - Which device initiated the interrupt

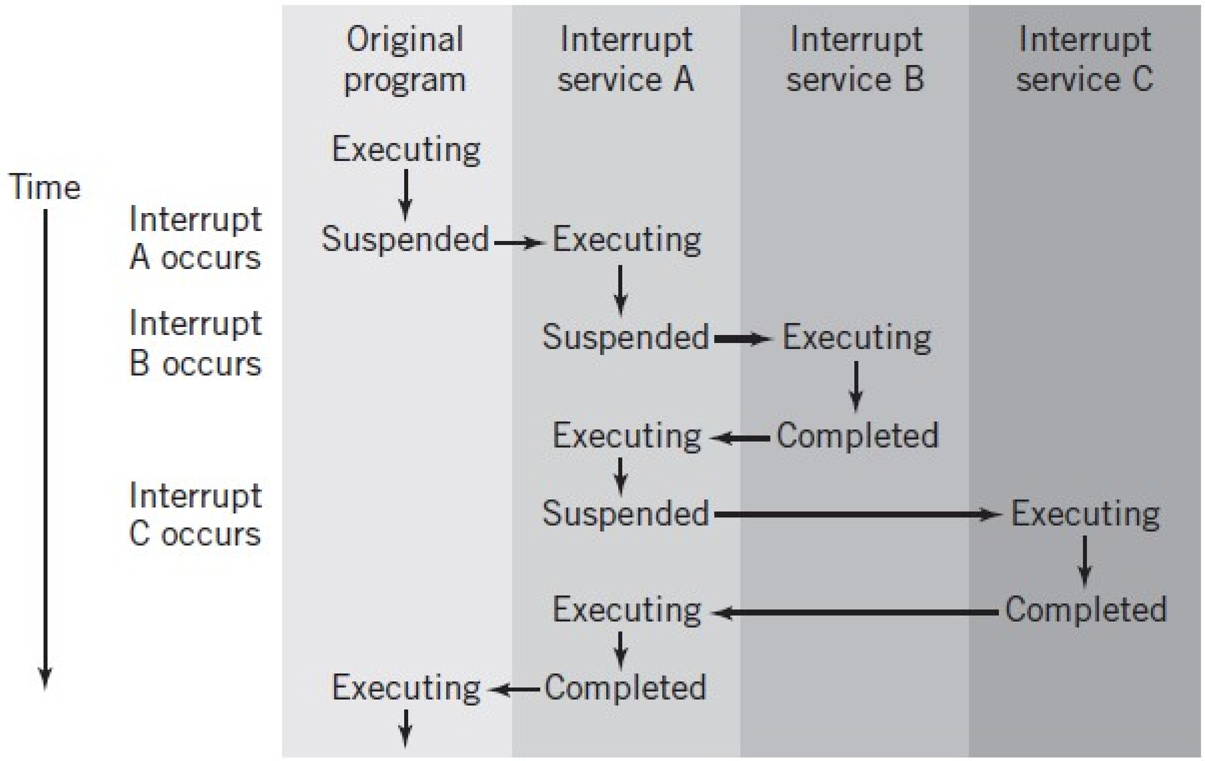
- Which device?
  - Vectored interrupt
  - Polling


Polled Interrupt Processing
Interrupt K occurs
Memory
General interrupt polling routine
. . .
Interrupt A service routine
. . .
Interrupt K service routine
Polls devices to determine which device, then
Jump to K service routine

MULTIPLE INTERRUPTS

- Priority
  - Keyboard Input vs. Task Completion?
  - Non-Maskable vs. Maskable interrupts

Multiple Interrupts

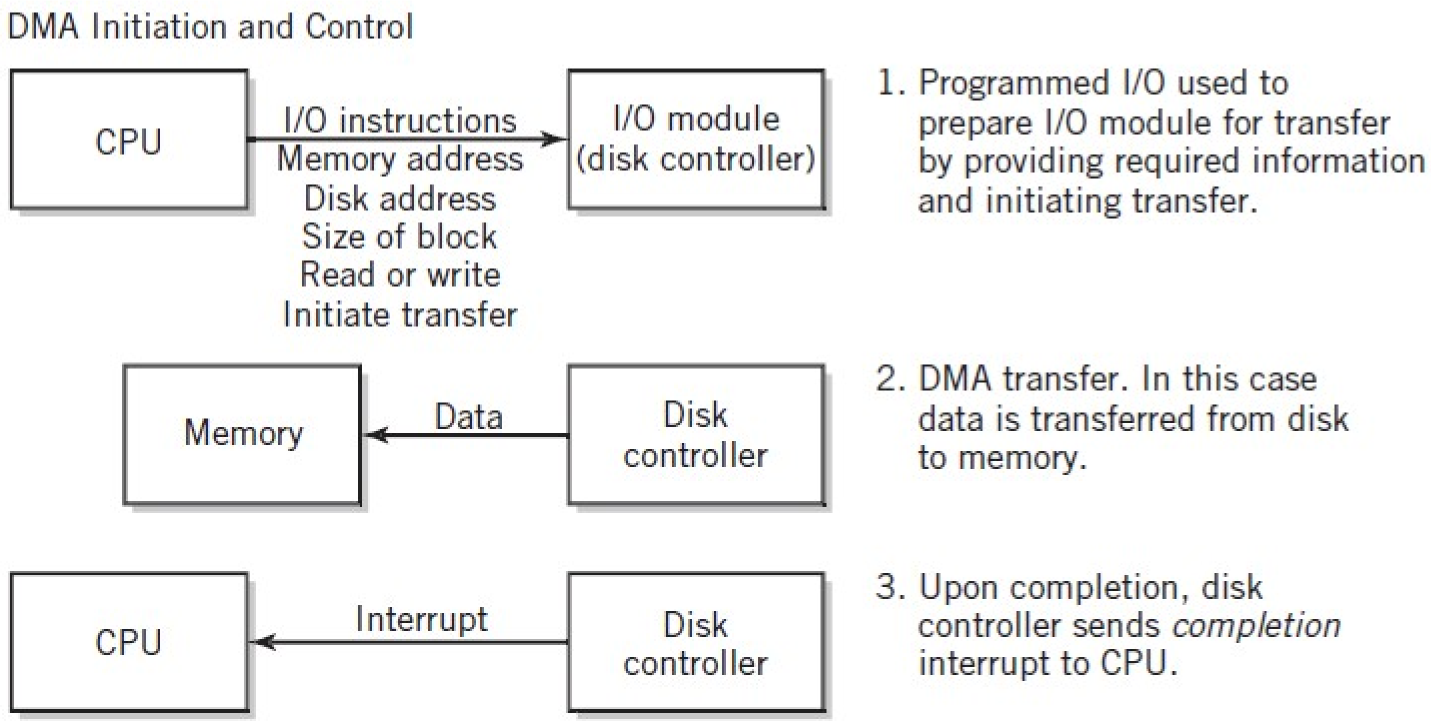


DIRECT MEMORY ACCESS

- Impractical to transfer data to the CPU using programmed I/O
  - Transfer to memory in blocks from devices I/O module

- Direct Memory Access (DMA)
  - Initiated by CPU using programmed I/O
  - Data transfer between I/O module and memory, under the control of I/O module
  - I/O module will notify CPU once done

DIRECT MEMORY ACCESS



1. Programmed I/O used to prepare I/O module for transfer by providing required information and initiating transfer.

2. DMA transfer. In this case data is transferred from disk to memory.

3. Upon completion, disk controller sends *completion* interrupt to CPU.

- Implementation concerns of DMA?

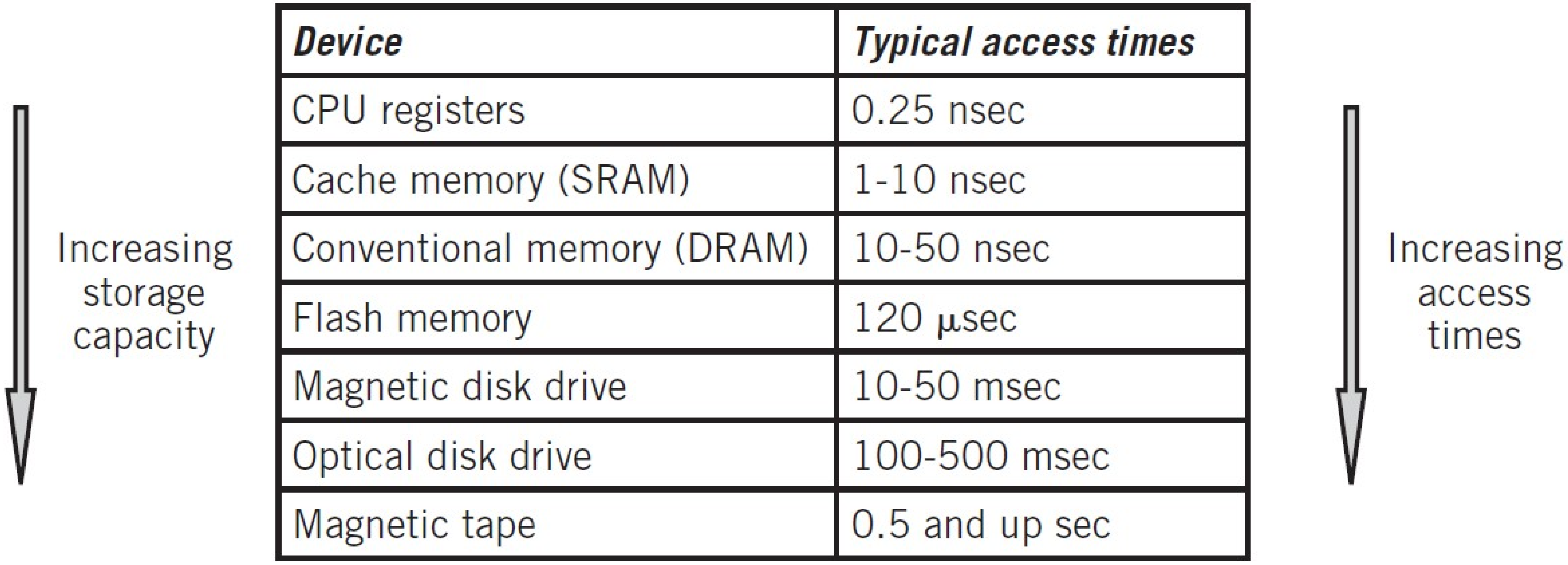
PERIPHERAL DEVICES

- All items external to the main processing function (CPU, Memory, Power Supply)
- Input devices, Output devices, Storage devices

STORAGE DEVICES

- Primary storage
- Secondary storage

The Storage Hierarchy



Device	*Typical access times*
CPU registers	0.25 nsec
Cache memory (SRAM)	1-10 nsec
Conventional memory (DRAM)	10-50 nsec
Flash memory	120 µsec
Magnetic disk drive	10-50 msec
Optical disk drive	100-500 msec
Magnetic tape	0.5 and up sec

Increasing storage capacity

Increasing access times

STORAGE DEVICES

- Secondary storage
  - Data will not immediately available to the CPU
  - Data in secondary storage needs to be copied to memory
  - Online vs. offline storage devices

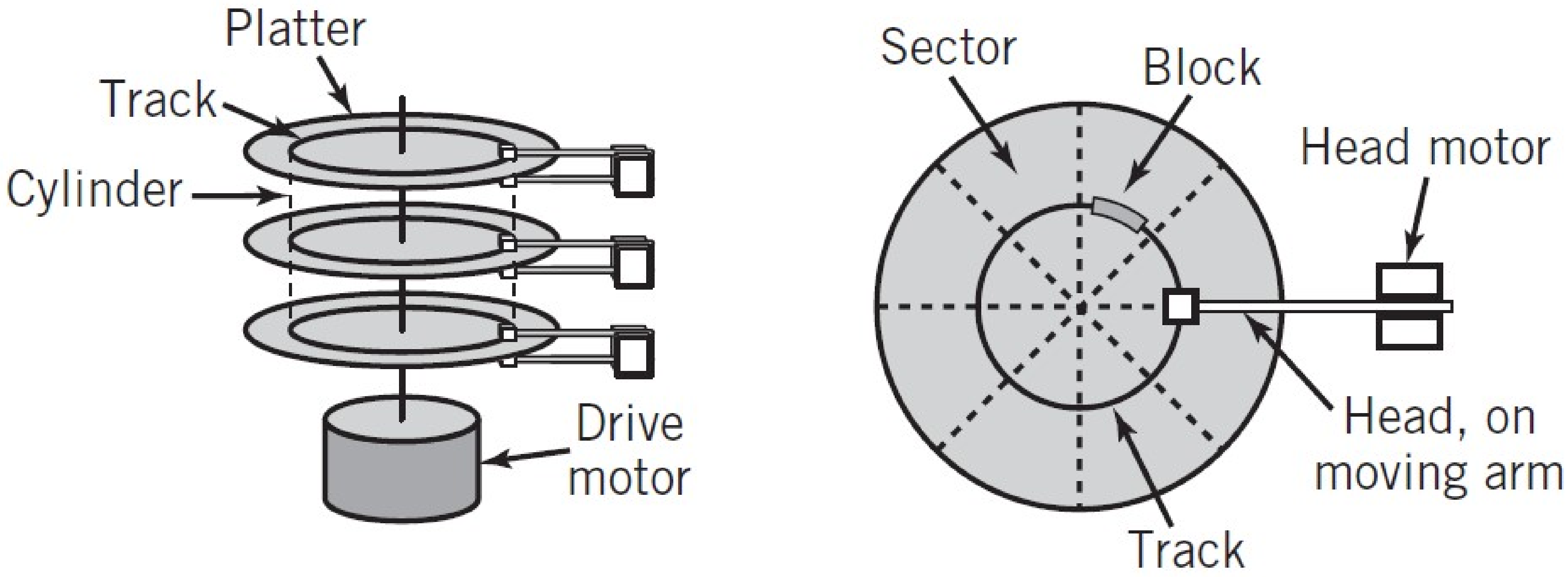
SOLID STATE MEMORY

- Flash Memory
  - Nonvolatile integrated circuit memory
  - ROM vs. Flash Memory?
  - Needs to be written in blocks

- Solid State Drives (SSDs)
  - Less weight, low power consumption and small size
  - Relatively immune to physical shock and vibration
  - Generate little heat and no noise

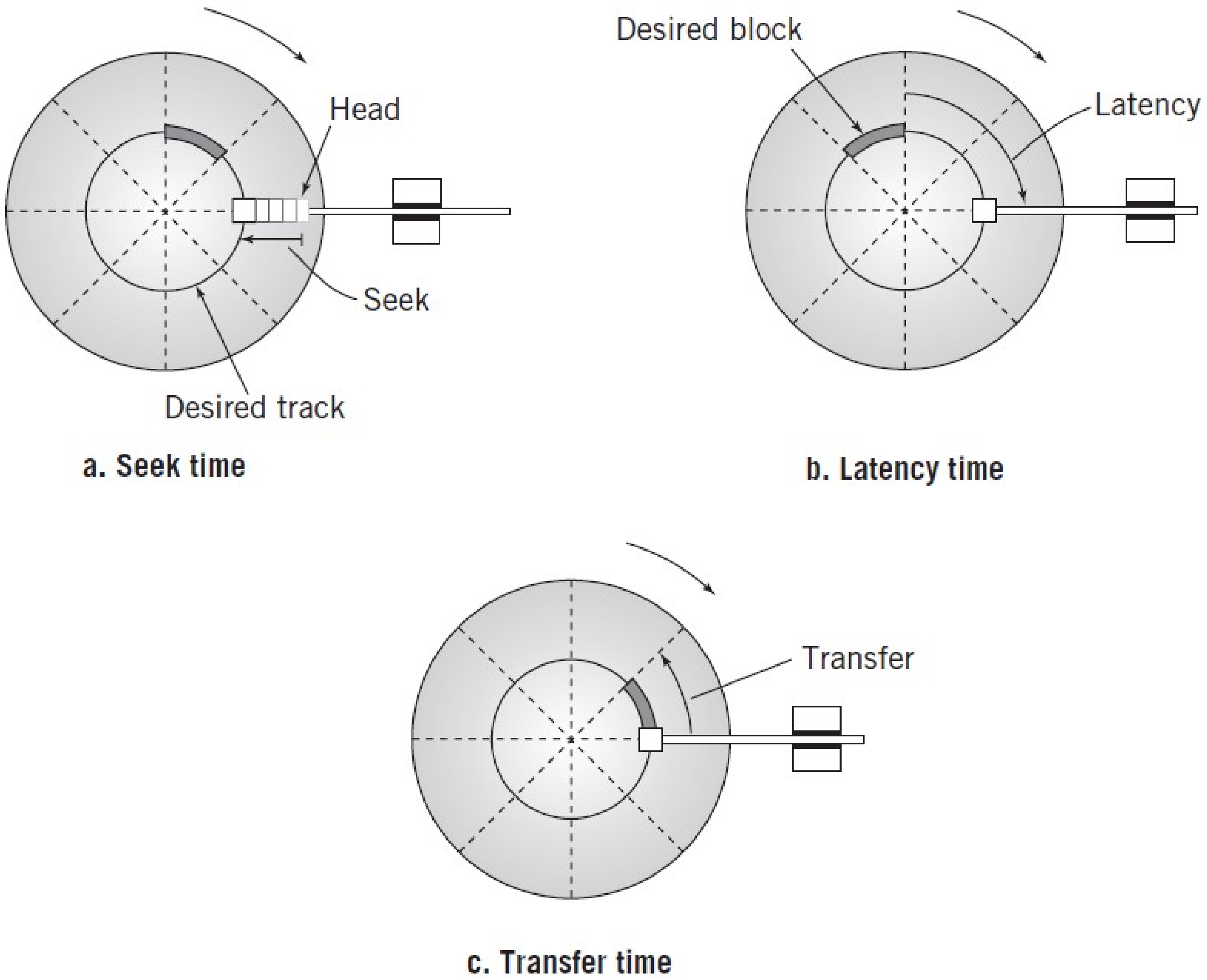
MAGNETIC DISKS

A Hard Disk Layout



MAGNETIC DISKS

Locating a Block of Data: (a) Seek Time, (b) Latency Time, (c) Transfer Time



a. Seek time

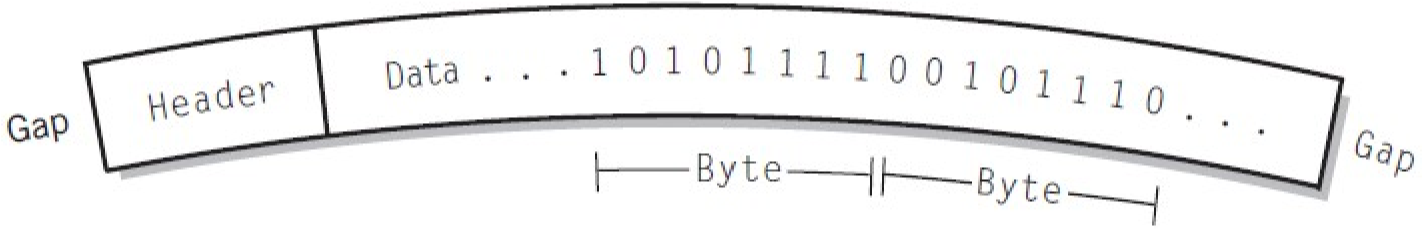
b. Latency time

c. Transfer time

MAGNETIC DISKS

- Data blocks

A Single Data Block

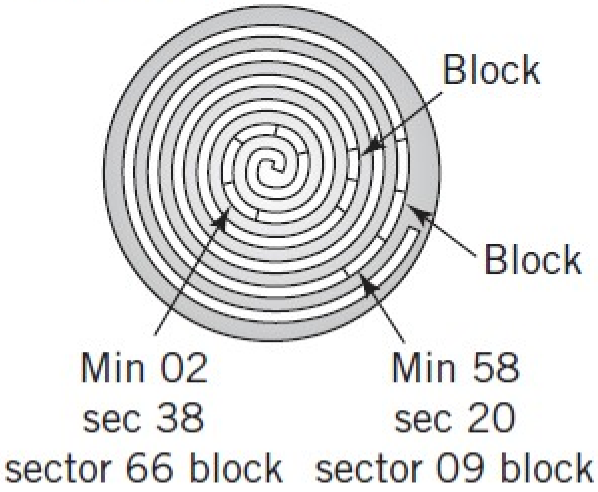


- Constant Angular Velocity (CAV)
- Constant Linear Velocity (CLV)

OPTICAL DISKS

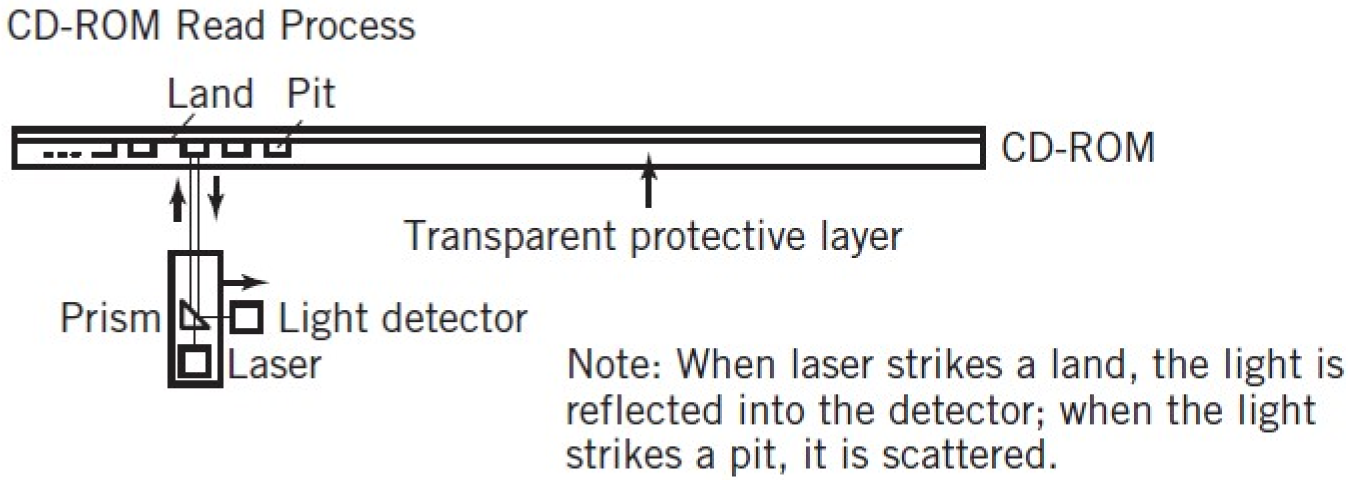
- Optical storage technologies include various types of CDs and DVDs

Layout of a CD-ROM versus a Standard Disk



OPTICAL DISKS

- Read process of optical disks



CD-ROM Read Process

Note: When laser strikes a land, the light is reflected into the detector; when the light strikes a pit, it is scattered.

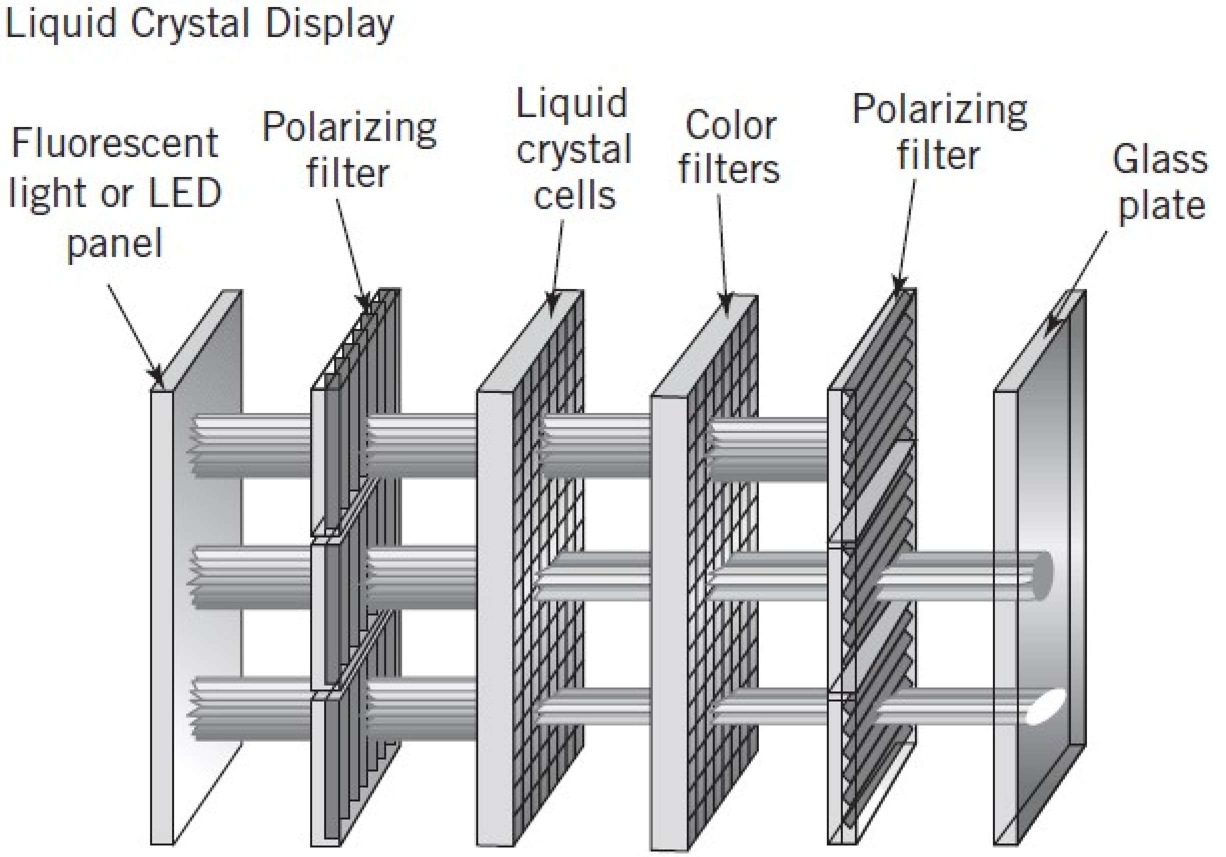
DISPLAY

- True color system
  - 24 bits per pixel
  - RGB layers with 8 bits per color
  - 256 × 256 × 256, or more than 16 million different colors

- 1024-pixel by 768-pixel graphic image would require over 2.3MB

DISPLAY

- Liquid Crystal Display Technology

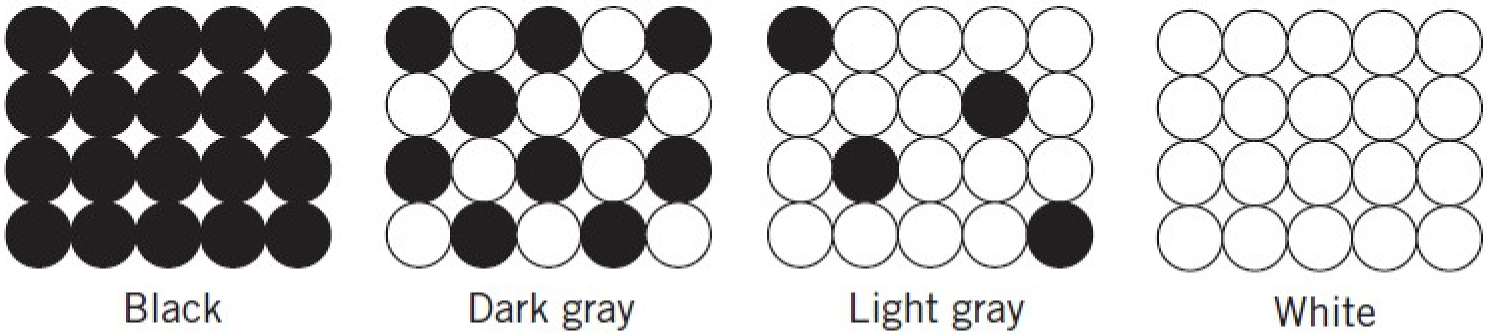


DISPLAY

- Organic Light-Emitting Diode (OLED)
  - Consists of a thin display panel that contains red, green, and blue LEDs for each pixel
  - Transistors for each LED will generate electrical current to light the LED

PRINTERS

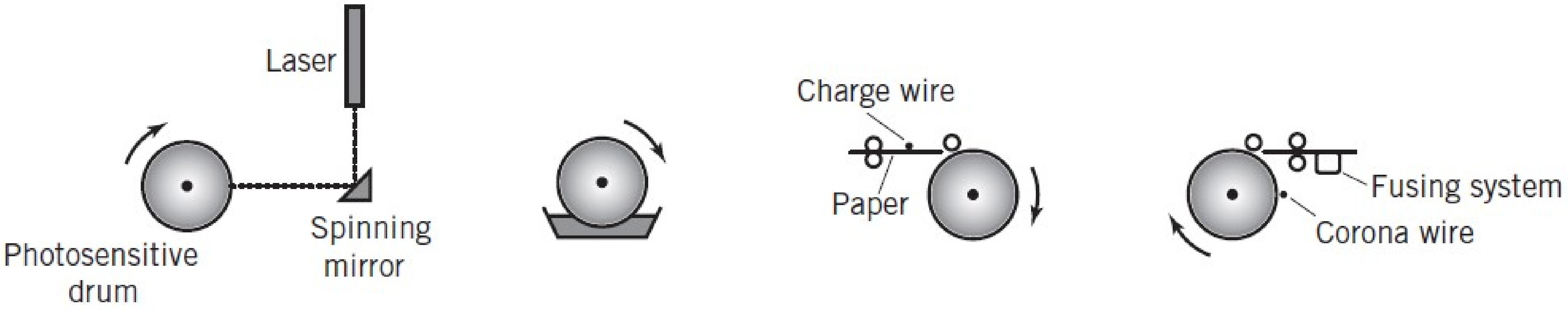
- Nearly all modern computer printers produce their output as a combination of dots



PRINTERS

- Laser printer operation

Operation of a Laser Printer



THANK YOU

REFERNCES

- Chapter 9 and 10: The Architecture of Computer Hardware, Systems Software & Networking: An Information Technology Approach -4th Edition, Irv Englander -John Wiley and Sons