Chapter 11
I/O Management and Disk Scheduling

(based on original slides by Pearson)

Categories of I/O Devices

- Human readable
  - Used to communicate with the user
  - Printers and terminals
- Machine readable
  - Used to communicate with electronic equipment
  - Disk drives, USB keys, Sensors, Controllers, Actuators
- Communication
  - Used to communicate with remote devices
  - Digital line drivers, modems

Differences in I/O Devices

- Data rate
  - May be differences of several orders of magnitude between the data transfer rates

I/O Device Data Rates

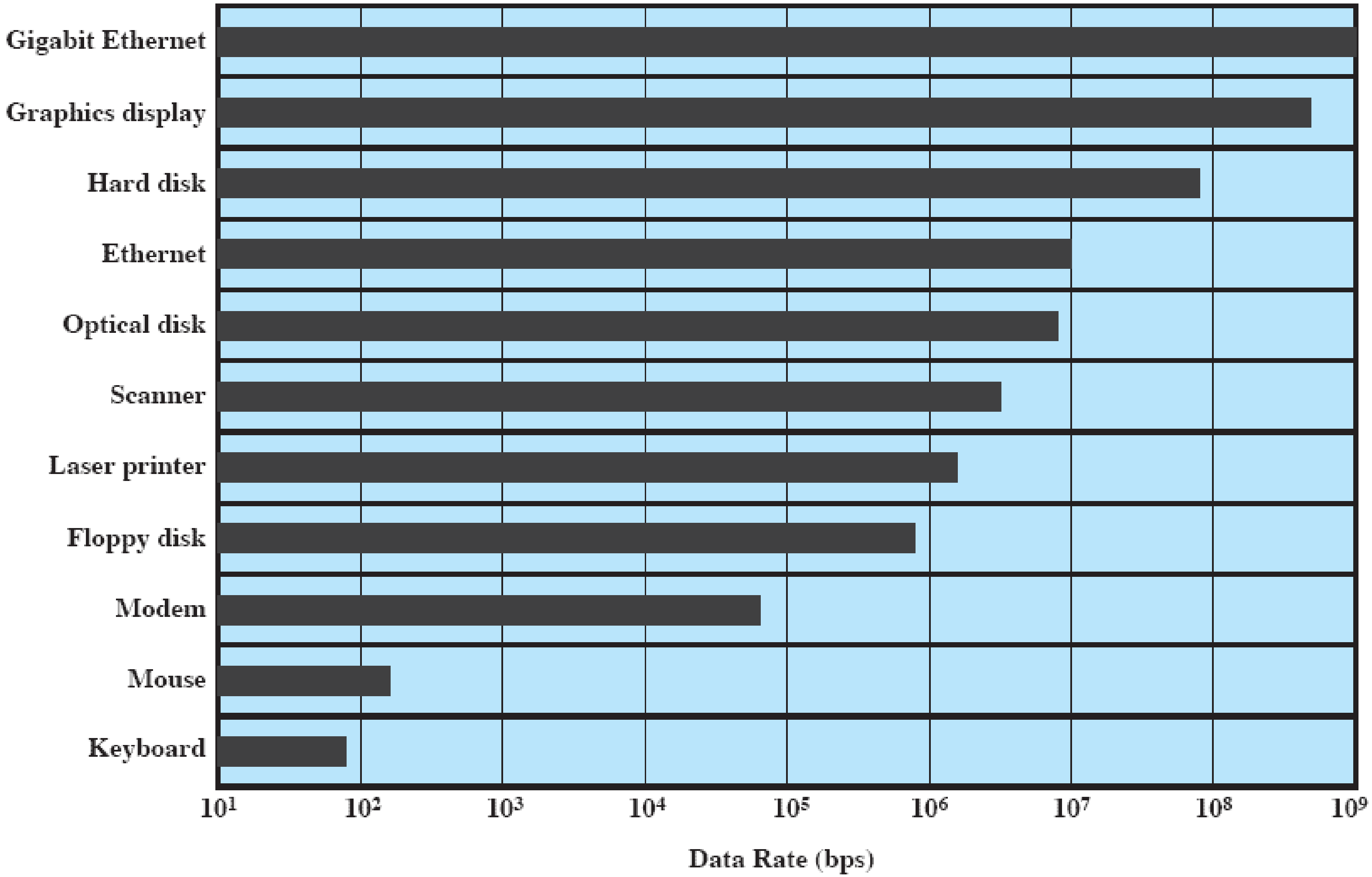


Figure 11.1 Typical I/O Device Data Rates

Differences in I/O Devices

- Application
  - ***The use of the device affects the management software***
  - Example:
    - Disk used to store files requires file management software
    - Disk used to store virtual memory pages needs special hardware and software to support it
    - Terminal used by system administrator may have a higher priority than the regular user’s one

Differences in I/O Devices

- Complexity of control
  - Simple one line on/off
  - Time dependent interaction: duty cycle of stepper motors
  - Complex, protocol-based interaction: most communication devices
- Unit of transfer
  - Data may be transferred as a stream of bytes for a terminal or in larger blocks for a disk

Differences in I/O Devices

- Data representation
  - Encoding schemes, parity

- Error conditions
  - Devices respond to errors differently
    - Motor loses it's torque
    - CAN does simply not transmit

Performing I/O

- Programmed I/O
  - Process is busy-waiting for the operation to complete
- Interrupt-driven I/O
  - I/O command is issued
  - Processor continues executing instructions
- Direct Memory Access (DMA)
  - DMA module controls exchange of data between main memory and the I/O device
  - Processor interrupted only after entire block has been transferred

Relationship Among Techniques

Table 11.1 I/O Techniques

	No Interrupts	**Use of Interrupts**
I/O-to-memory transfer through processor	Programmed I/O	Interrupt-driven I/O
Direct I/O-to-memory transfer		Direct memory access (DMA)

Evolution of the I/O Function

- Processor directly controls a peripheral device
- Controller or I/O module is added
  - Processor uses programmed I/O without interrupts
  - Processor does not need to handle details of external devices

Evolution of the I/O Function

- Controller or I/O module with interrupts
  - Processor does not spend time waiting for an I/O operation to be performed
- Direct Memory Access
  - Blocks of data are moved into memory without involving the processor
  - Processor involved at beginning and end only

Evolution of the I/O Function

- I/O module is a separate processor
- I/O processor
  - I/O module has its own local memory
  - It is a computer in its own right

Direct Memory Address

- Processor delegates I/O operation to the DMA module
- DMA module transfers data directly to or from memory
- When transfer is complete, DMA module sends an interrupt signal to the processor

DMA

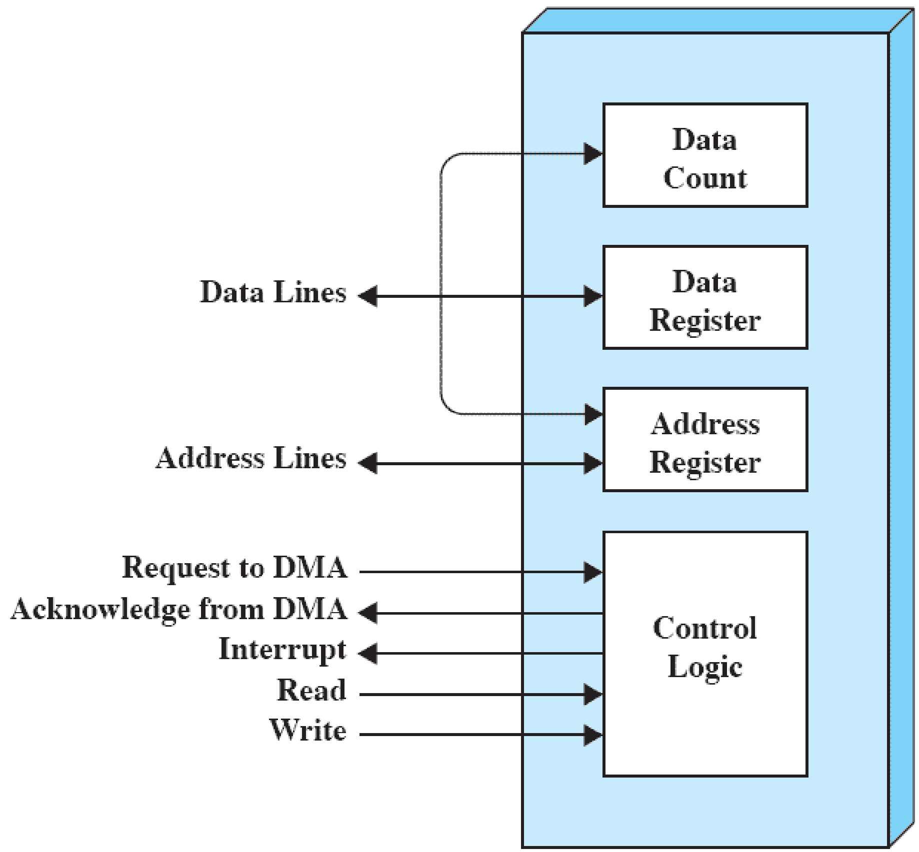
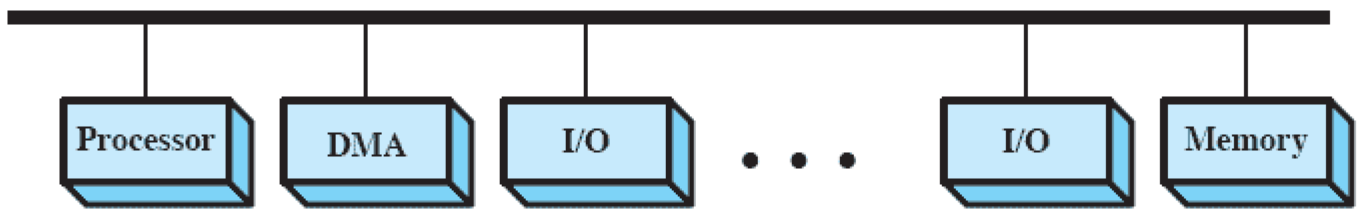


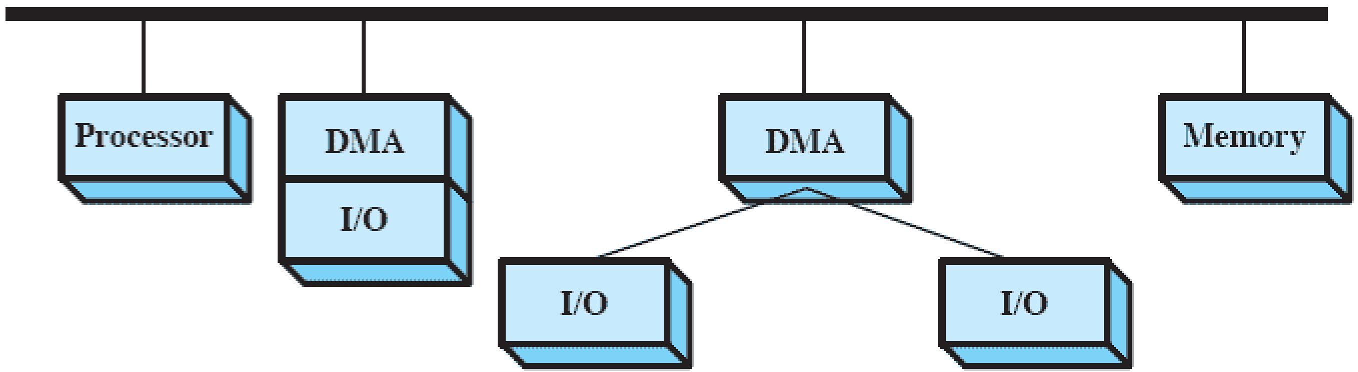
Figure 11.2 Typical DMA Block Diagram

DMA Configurations



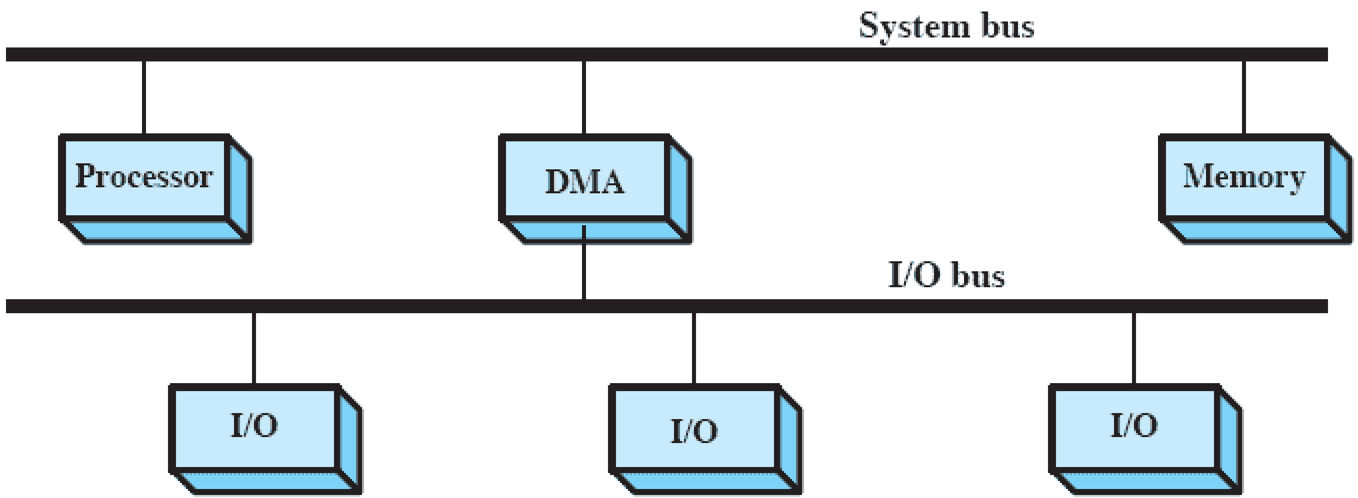
(a) Single-bus, detached DMA

DMA Configurations



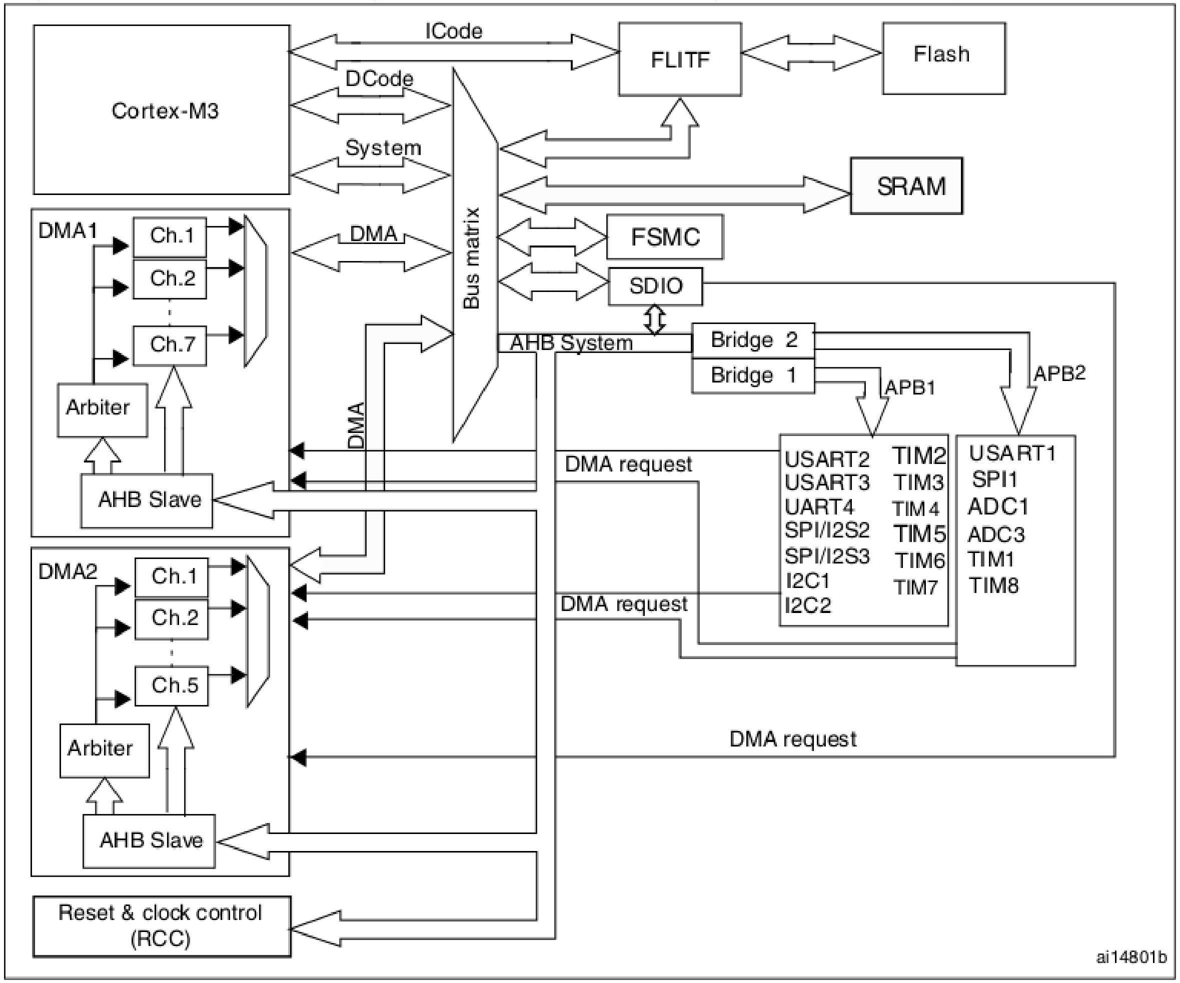
(b) Single-bus, Integrated DMA-I/O

DMA Configurations



(c) I/O bus

DMA Real World



STM32F1xx MCU ~0.30$/unit

OS Design Issues wrt I/O

- Efficiency
  - Most I/O devices are extremely slow compared to main memory
  - Use of multiprogramming allows interleaving of I/O and processing
  - Even in the future: I/O will not keep up with processor throughputs
  - Swapping is used to bring in additional Ready processes; swapping is an I/O operation

OS Design Issues wrt I/O

- Generality
  - Desirable to handle all I/O devices in a uniform manner
  - Hide most of the details of device I/O in lower-level routines

I/O Buffering

- Reasons for buffering
  - Processes must wait for I/O to complete before proceeding
  - Certain pages must remain in main memory during I/O

- Risk of deadlock:
  - Process calls an I/O routine and blocks
  - Process is swapped out
  - Process waits for I/O completion, but I/O waits for the process to be swapped in to access the memory

I/O Buffering

- Block-oriented
  - Information is stored in fixed sized blocks
  - Transfers are made a block at a time
  - Used for disks and USB keys
- Stream-oriented
  - Transfer information as a stream of bytes
  - Used for terminals, printers, communication ports, mouse and other pointing devices, and most other devices that are not secondary storage

- The kernel (I/O routines) handle buffering

Single Buffer

- Operating system assigns a buffer in main memory for an I/O request
- Block-oriented
  - Input transfers made to buffer
  - Block moved to user space when needed
  - Another block is moved into the buffer
    - Read ahead

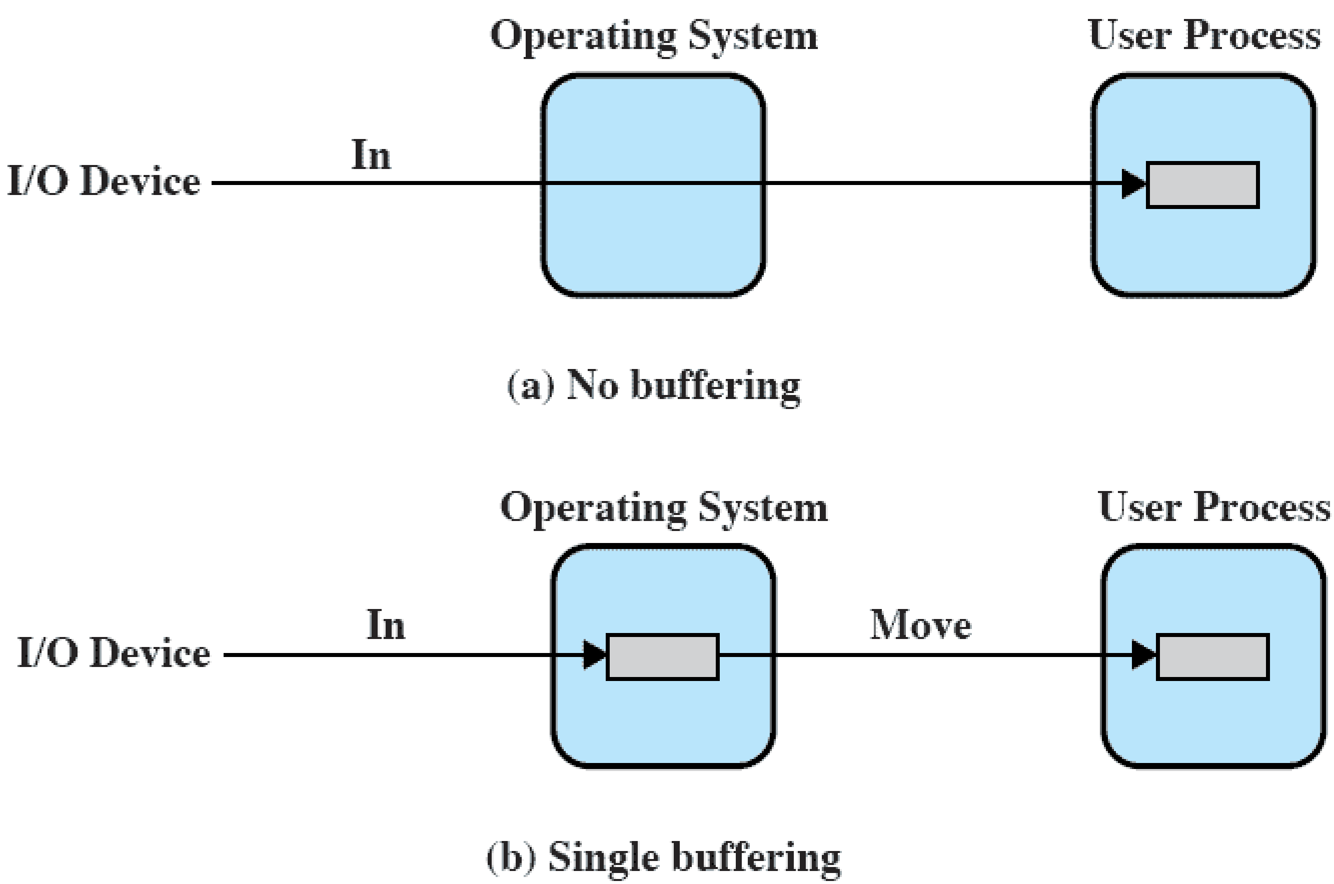
Single Buffer

- Block-oriented
  - User process can process one block of data while next block is read in
  - Swapping can occur since input is taking place in system memory, not user memory
  - Operating system keeps track of assignment of system buffers to user processes

Single Buffer

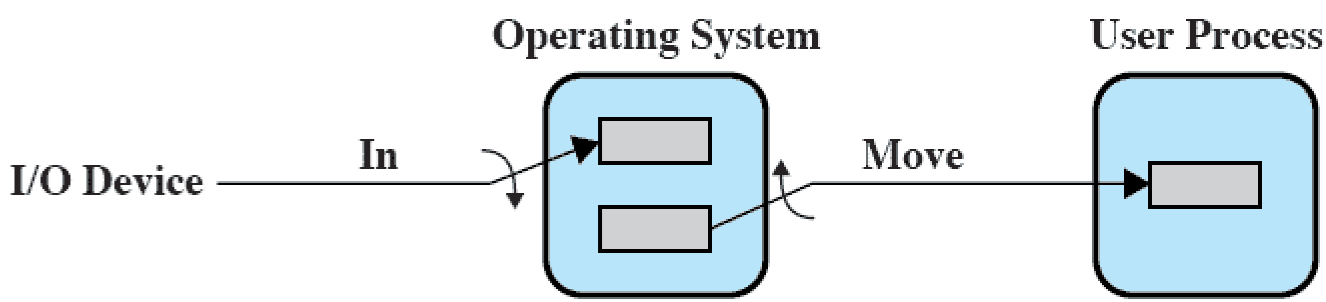
- Stream-oriented
  - Use a line at a time
  - User input from a terminal is one line at a time with carriage return signaling the end of the line
  - Output to the terminal is one line at a time
  - Flush() call in C

Single Buffer



Double Buffer

- Use two system buffers instead of one
- A process can transfer data to or from one buffer while the operating system empties or fills the other buffer

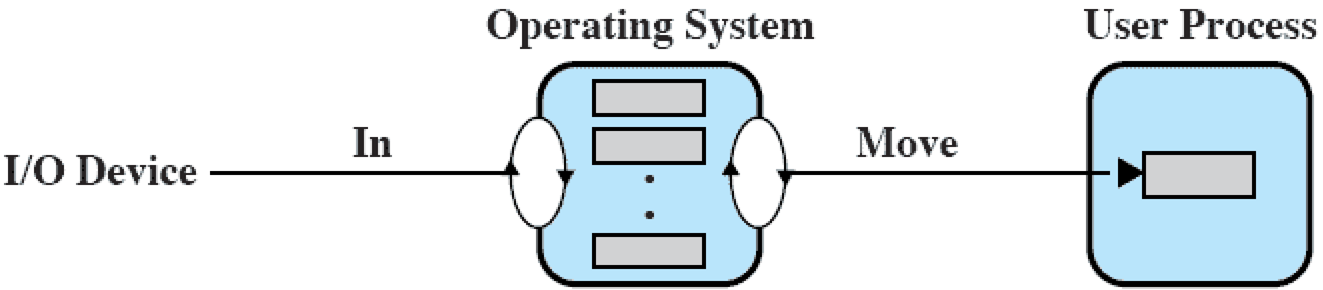


(c) Double buffering

- Process does not have to wait for I/O

Circular Buffer

- More than two buffers are used
- Each individual buffer is one unit in a circular buffer



(d) Circular buffering

- Good for bursty I/O

Disk Performance Parameters

- To read or write, the disk head must be positioned at the desired track and at the beginning of the desired sector
- Seek time
  – Time it takes to position the head at the desired track
- Rotational delay or rotational latency
  – Time its takes for the beginning of the sector to reach the head

Timing of Disk I/O Transfer

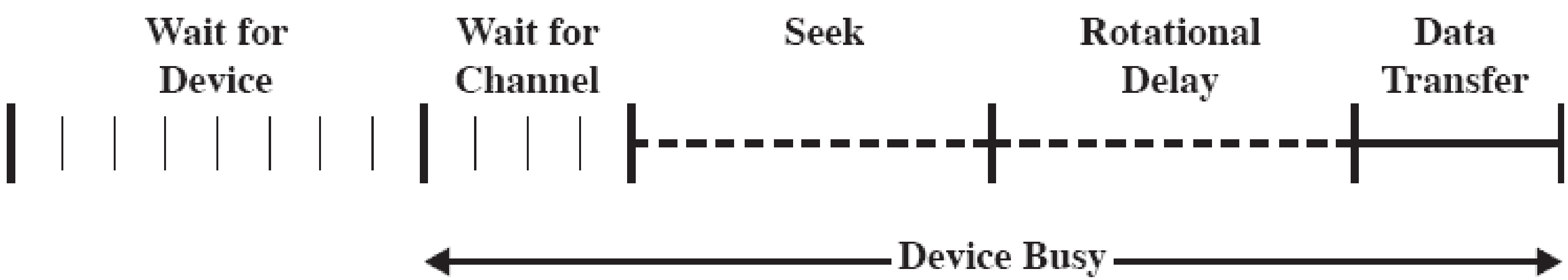


Figure 11.6 Timing of a Disk I/O Transfer

Disk Performance Parameters

- Access time
  - Sum of seek time and rotational delay
  - The time it takes to get in position to read or write

- There can be a huge difference between sequential read access and random read access.