

- speed of components
 - CPU speed increasing rapidly (50% per year)
 - DRAM speed increasing (10% per year)
 - disk speed increasing (4 to 6 % per year)
 - I/O is crucial but lagging in improvement

after n years	CPU time	I/O time	Total time	% I/O time
0	90	10	100	10
1	60	10	70	14
2	40	10	50	20
5	12	10	22	45
10	1	10	11	91

- CPU time improves by 50% per year, i.e., a speed up of 1.5.
- I/O does not improve
- I/O is of major concern

- importance of I/O is increasing
 - graphics for displaying large complicated results
 - parallel and distributed computing
 - internet
 - information processing
- it is often the sophistication of the various networks in a system that determine its cost and place in the PC to supercomputer hierarchy

insert figure 8.1

I/O Issues

- Performance
 - latency (response time)
 - bandwidth (throughput)
- expandability (scalability)
- standard interfaces
- fault tolerance

Diversity of I/O Devices

- behavior: input, output, storage
- partner: human, machine (many different levels)
- data rate: peak, average, worst case rate at which data transfer is supported
- keyboard : input, human, 0.01 KB/s
- graphics display : human, 60MB/s (much higher in some cases)
- modem : input/output, machine, 8KB/s
- magnetic disk : storage, machine, 10 MB/s

Magnetic Disks

- Floppy disk or Hard disk
- advantages of hard disk technology
 - larger due to rigid structure
 - higher density due to more precise control mechanisms
 - higher data rate due to density and spin rate
 - multiple platters each with two recording surfaces

- disadvantage
 - size
 - cost
 - more easily damaged

insert figure 8.4

Hard disks

- platters (1 to 15 per disk)
- tracks (1000 to 5000 per platter surface)
- sectors (64 to 200 per track) each containing:
 - sector number
 - gap
 - data for sector
 - gap

- newer disks allow storage with constant bit density, i.e., more information on outer tracks than inner
- the sector is the smallest unit of transfer
- on input, all information is read into buffer in memory and accessed later for selection of information needed in program address space
- on output information is buffered until buffer full or significant event (close file) then transferred to disk

- seek time – move arm to desired track
- rotational delay – time for sector to appear under head

- 3600 RPM

- average delay –

$$\frac{0.5rotation}{3600RPM} = \frac{0.5rotation}{60RPS} = 8.3ms$$

- transfer time – time to transfer data under head
- control time – overhead to control the above activities
- wait time – time spent waiting while other requests use disk

What is the average time to access a 512 byte sector for a disk at 5400 RPM given the average seek time is 12 ms, transfer rate is 5 MB/s, and controller overhead is 2 ms?

avg access time = avg seek time + controller time + avg rotational latency + transfer time

avg access time = 12 ms + 2 ms + avg rotational latency + transfer time

avg rotational latency =

$$\frac{0.5 \text{ rotation}}{5400 \text{ RPM}} = \frac{0.5 \text{ rotation}}{90 \text{ RPS}} \approx 5.6 \text{ ms}$$

transfer time =

$$\frac{0.5 \text{ KB}}{5.0 \text{ MB/s}} = \frac{2^9 \text{ B}}{5 * 2^{20} \text{ B/s}} = \frac{1}{5 * 2^{11} \text{ B/s}} \approx 0.1 \text{ ms}$$

avg access time = 19.7 ms

- seek time often close to 0 due to locality
- many tricks for improving disk performance
 - for multiple pending accesses schedule the one with the lowest seek distance
 - cache sectors passed over on the way to desired sector
 - cache other information from nearby tracks and platters

Network Characteristics

- distance : 0.01 to 10,000 kilometers
- speed : 0.001 MB/s to 100 MB/s
- topology : bus, ring, star, tree, hypercube etc.
- shared lines : none (point to point) or shared (multiplexed)

- terminal network
 - RS232 standard (0.3 to 19.2 Kbit/s)
 - star (point to point)
 - 10 to 100 meters from central machine
- local area network
 - ethernet (10 Mbit/s)
 - fast ethernet (100 Mbit/s)
 - bus with multiple masters
 - up to a few kilometers in distance (often within one building)
 - multiple networks connected via switches

- long-haul
 - 10 to 10,000 kilometers
 - example, ARPANET
 - uses dedicated leased lines
 - messages are broken into packets
 - packets can be sent by different paths
 - internet means the connection of different long-haul networks

- a bus is a shared communication link between subsystems
- contains control lines (for requests and acknowledgements) and data lines (for data, commands, addresses)
- advantages
 - versatility – new devices can be added to the bus
 - low cost – single set of wires shared in multiple ways
- disadvantage – limited throughput

Types of Buses

- processor-memory bus
 - short
 - high speed
 - synchronous
 - customized based on known interface requirements
- I/O bus
 - long
 - speed to match I/O devices
 - follows a general bus standard
 - supports many types of devices

- asynchronous

- backplane bus

- allows processors, memory, I/O devices to communicate

- often built into the backplane of the machine

- may be used as a processor memory bus

- standard protocol

insert figure 8.9

- (a) is typical of older machines with a standard backplane used for all communication
- (b) processor-memory bus is customized or a backplane bus (PCI), other I/O buses (SCSI) plug into it
- (c) is common for high performance, processor memory bus is customized and high speed and is among the units that plug into a backplane (with other I/O buses)

insert figure 8.7

Memory to Disk Output

1. control signal to disk to prepare for output.
addresses in memory and disk is sent on
data lines
2. memory accesses data
3. memory signals I/O device that data is
ready and exchange occurs using the data
lines

insert figure 8.8

Disk to Memory Input

- control signals sent to request a memory write, data lines contain memory address (read by memory) and disk address (read by disk)
- data transferred disk to memory

Synchronous and Asynchronous Buses

- Synchronous bus
 - includes a clock signal in control lines
 - uses a protocol relative to the clock signal (who does what on which clock cycle)
 - every device on bus runs at same clock rate
 - must be short to avoid clock skew

- Asynchronous bus

- does not use a clock signal

- uses handshaking protocol (information is exchanged about when each is ready for what, e.g., finite state machine control)

- can accommodate devices running at different speeds

Bus Parameters

- bus width – parallel vs. multiplexed address and data lines; parallel is faster and more expensive
- data width – number of parallel bits sent during data transfer; wider is better and more expensive
- number of data sets (each set is data width wide) sent in consecutive cycles without new address information or releasing the bus; longer is faster and more expensive
- bus master – a master (CPU) can initiate a request for activity, a slave can only respond (memory, disk); control is simpler when there are fewer masters.

Bus Arbitration

- arbitration decides which master gets control next
- Factors:
 - priority (relative importance of traffic)
 - fairness (everyone gets control eventually)
- schemes
 - daisy chain
 - centralized (multiple request lines, one decision maker)

- distributed schemes
 - distributed self selection (multiple request lines, device numbers included, everyone makes decision, not fair)
 - distributed by collision detection (multiple request lines, if more than one used
 - collision, resolved typically by delay)