

Lecture 12:
**I/O Introduction: Storage Devices,
Metrics, & Productivity**

Professor David A. Patterson
Computer Science 252
Spring 1998

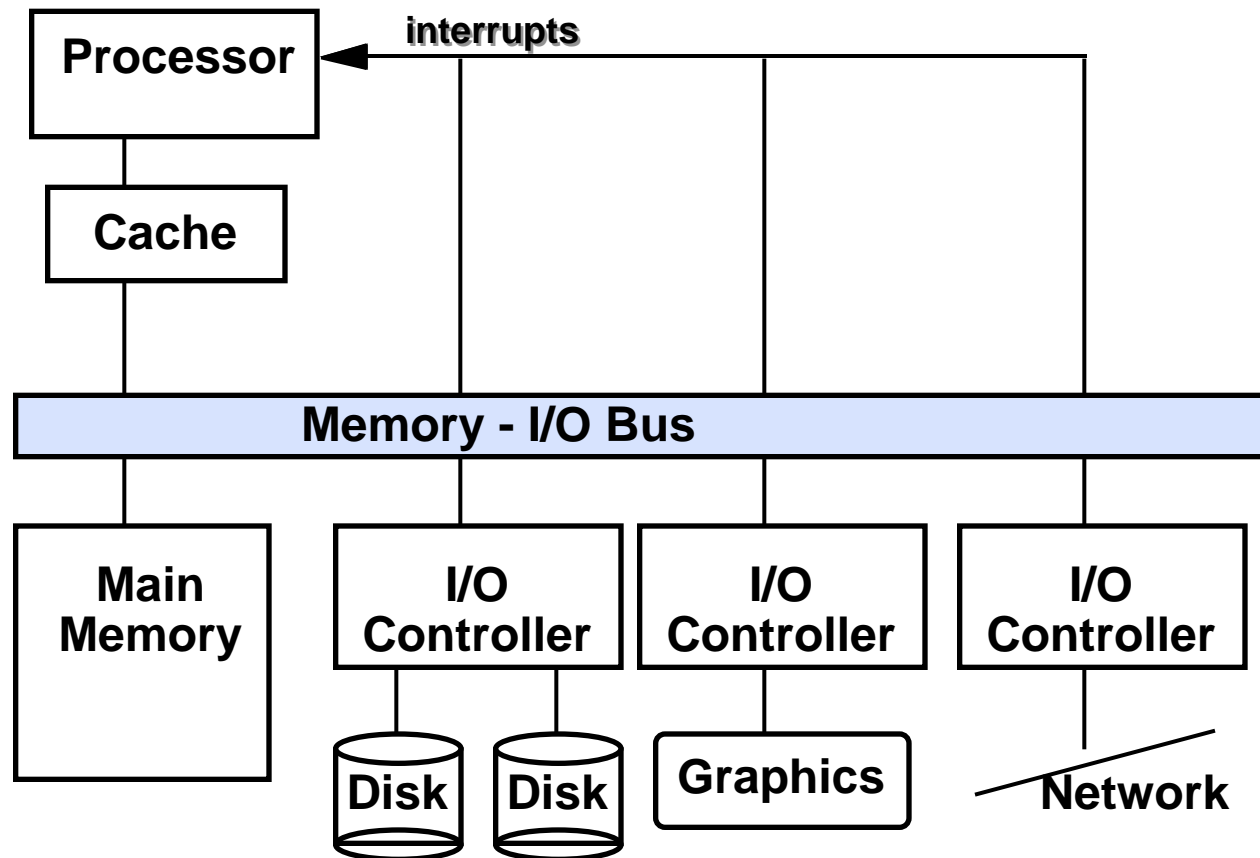
Motivation: Who Cares About I/O?

- CPU Performance: 60% per year
- I/O system performance limited by *mechanical* delays (disk I/O)
 - < 10% per year (IO per sec or MB per sec)
- Amdahl's Law: system speed-up limited by the slowest part!
 - 10% IO & 10x CPU => 5x Performance (lose 50%)
 - 10% IO & 100x CPU => 10x Performance (lose 90%)
- I/O bottleneck:
 - Diminishing fraction of time in CPU
 - Diminishing value of faster CPUs

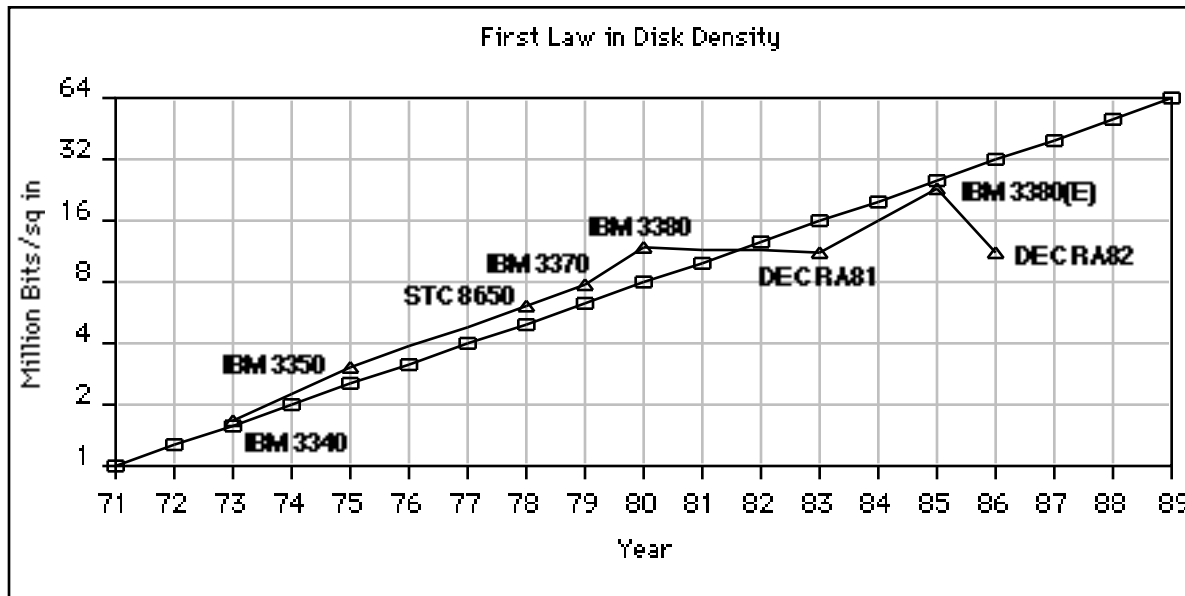
Storage System Issues: 1.5 weeks

- **Historical Context of Storage I/O**
- **Secondary and Tertiary Storage Devices**
- **Storage I/O Performance Measures**
- **Processor Interface Issues**
- **A Little Queuing Theory**
- **Redundant Arrays of Inexpensive Disks (RAID)**
- **I/O Buses**
- **ABCs of UNIX File Systems**
- **I/O Benchmarks**
- **Comparing UNIX File System Performance**

I/O Systems



Technology Trends



Disk Capacity
now doubles
every
18 months; before
1990 every 36 months

- Today: Processing Power Doubles Every 18 months
- Today: Memory Size Doubles Every 18 months(4X/3yr)
- Today: Disk Capacity Doubles Every 18 months
- *Disk Positioning Rate (Seek + Rotate) Doubles Every Ten Years!*

**The I/O
GAP**

Storage Technology Drivers

- **Driven by the prevailing computing paradigm**
 - 1950s: migration from batch to on-line processing
 - 1990s: migration to ubiquitous computing
 - » computers in phones, books, cars, video cameras, ...
 - » nationwide fiber optical network with wireless tails
- **Effects on storage industry:**
 - **Embedded storage**
 - » smaller, cheaper, more reliable, lower power
 - **Data utilities**
 - » high capacity, hierarchically managed storage

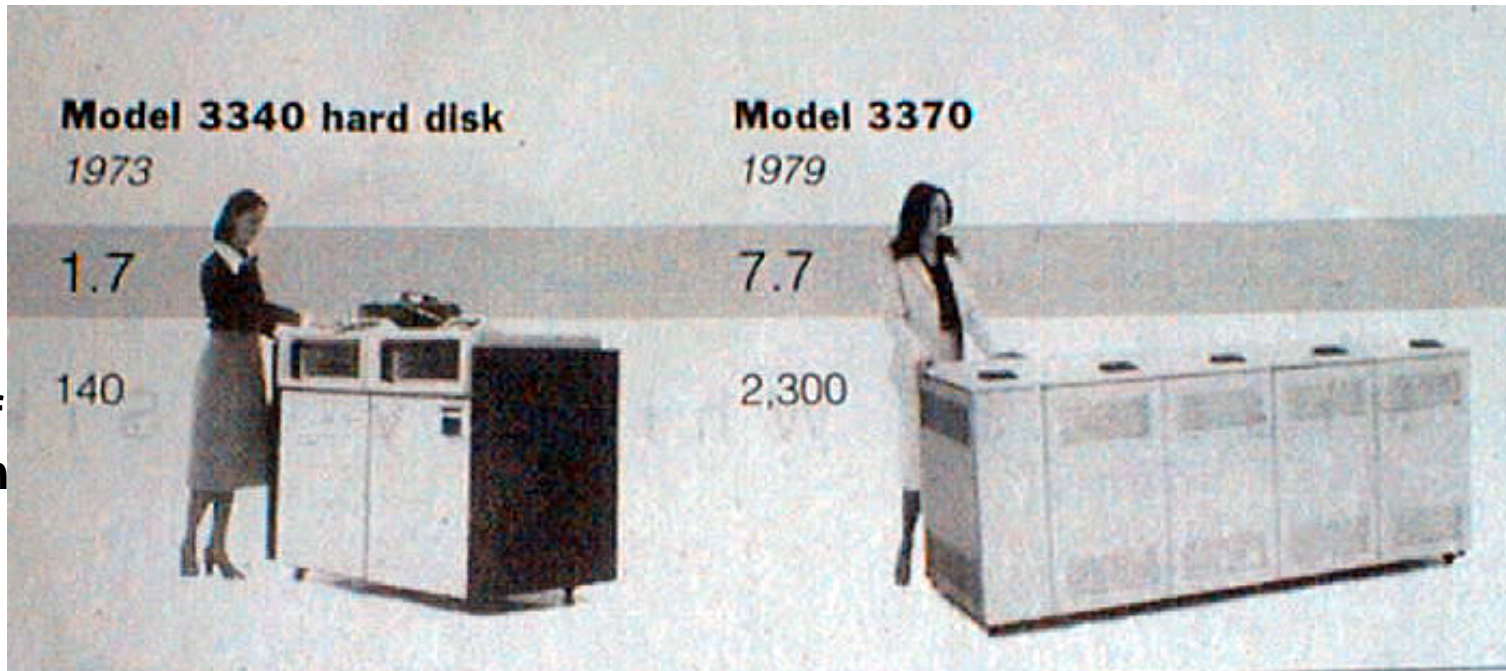
Historical Perspective

- **1956 IBM Ramac — early 1970s Winchester**
 - Developed for mainframe computers, proprietary interfaces
 - Steady shrink in form factor: 27 in. to 14 in.
- **1970s developments**
 - 5.25 inch floppy disk formfactor (microcode into mainframe)
 - early emergence of industry standard disk interfaces
 - » ST506, SASI, SMD, ESDI
- **Early 1980s**
 - PCs and first generation workstations
- **Mid 1980s**
 - Client/server computing
 - Centralized storage on file server
 - » accelerates disk downsizing: 8 inch to 5.25 inch
 - Mass market disk drives become a reality
 - » industry standards: SCSI, IPI, IDE
 - » 5.25 inch drives for standalone PCs, End of proprietary interfaces

Disk History

Data density
Mbit/sq. in.

Capacity of
Unit Shown
Megabytes



1973:
1.7 Mbit/sq. in
140 MBytes

1979:
7.7 Mbit/sq. in
2,300 MBytes

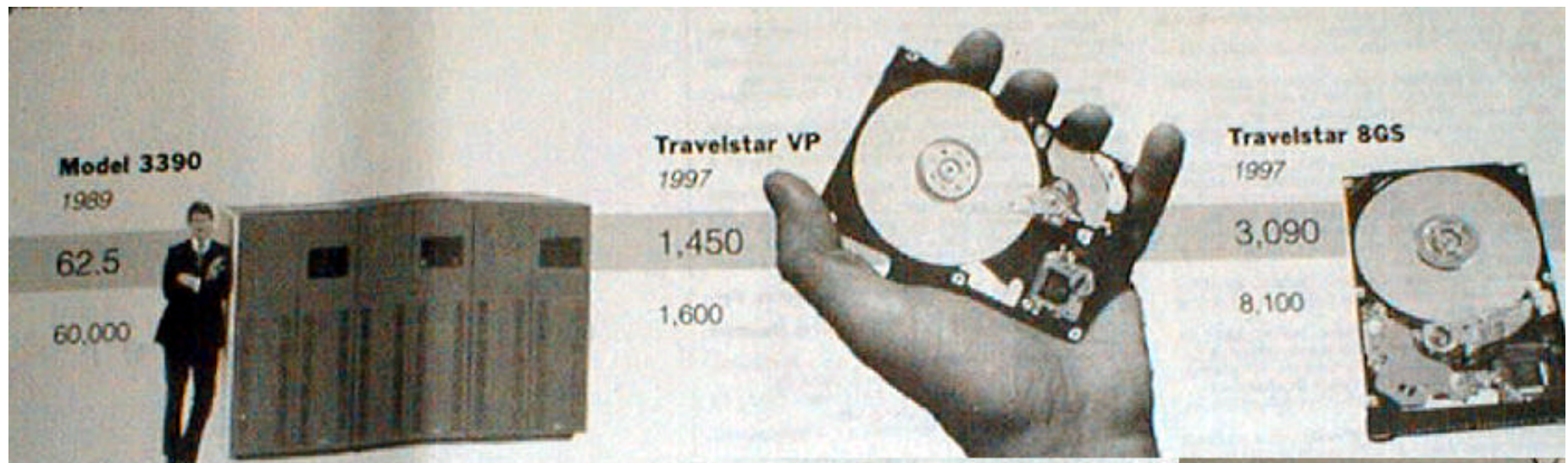
source: *New York Times*, 2/23/98, page C3,

“Makers of disk drives crowd even more data into even smaller spaces” DAP Spr.'98 ©UCB 8

Historical Perspective

- **Late 1980s/Early 1990s:**
 - Laptops, notebooks, (palmtops)
 - 3.5 inch, 2.5 inch, (1.8 inch formfactors)
 - Formfactor plus capacity drives market, not so much performance
 - » Recently Bandwidth improving at 40%/ year
 - Challenged by DRAM, flash RAM in PCMCIA cards
 - » still expensive, Intel promises but doesn't deliver
 - » unattractive MBytes per cubic inch
 - Optical disk fails on performance (e.g., NEXT) but finds niche (CD ROM)

Disk History



1989:
63 Mbit/sq. in
60,000 MBytes

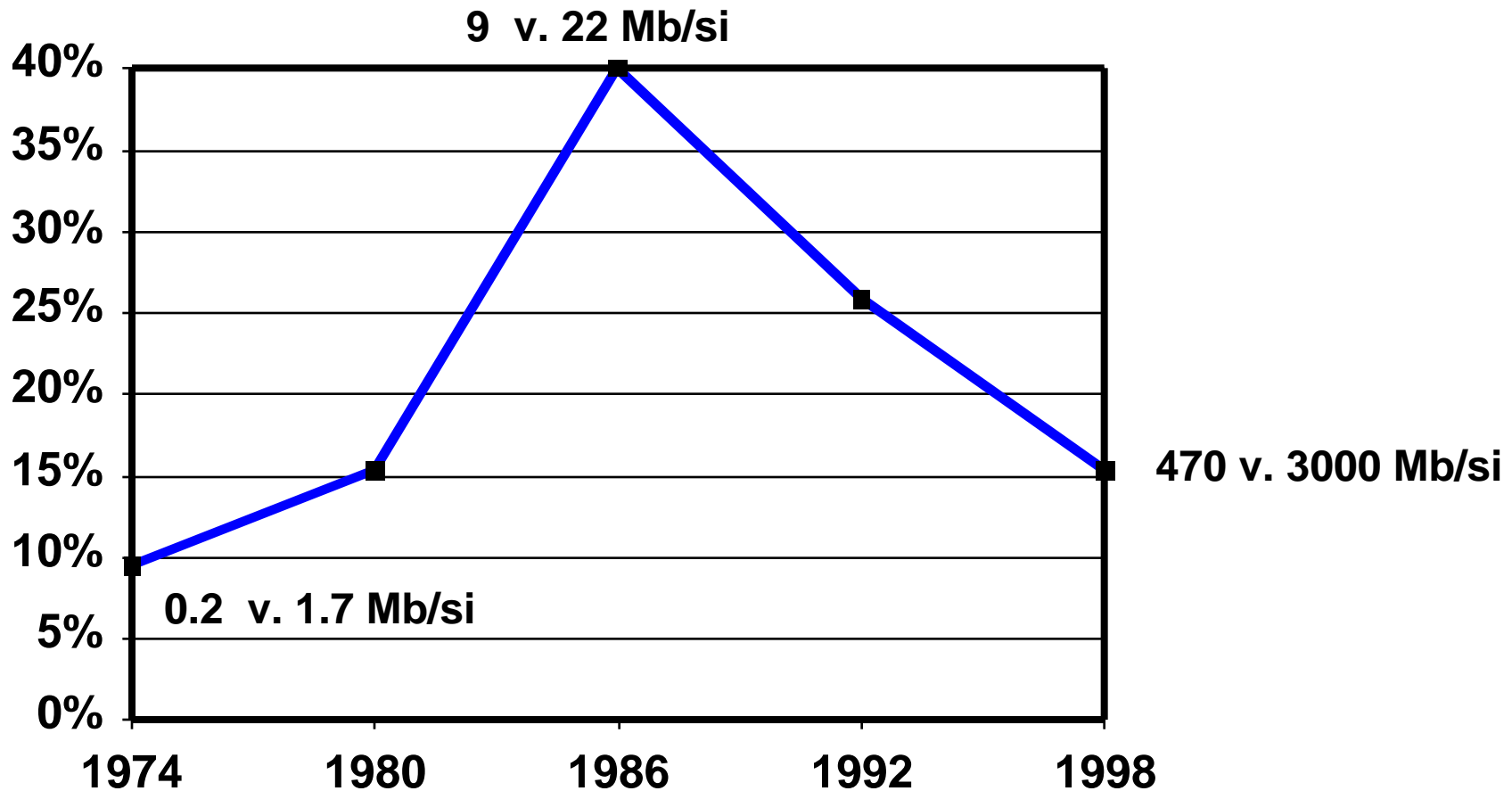
1997:
1450 Mbit/sq. in
2300 MBytes

1997:
3090 Mbit/sq. in
8100 MBytes

source: *New York Times*, 2/23/98, page C3,

"Makers of disk drives crowd even more data into even smaller spaces" DAP Spr.'98 ©UCB 10

MBits per square inch: DRAM as % of Disk over time



source: *New York Times*, 2/23/98, page C3,

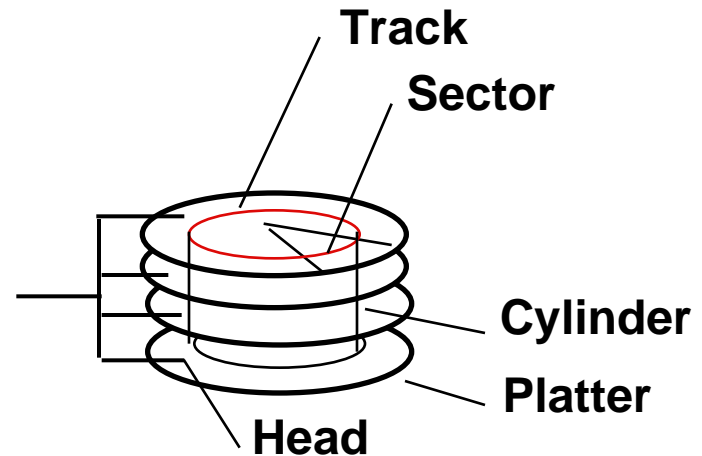
"Makers of disk drives crowd even more data into even smaller spaces" DAP Spr.'98 ©UCB 11

Alternative Data Storage Technologies: Early 1990s

Technology	Cap (MB)	BPI	TPI	BPI*TPI (Million)	Data Xfer (KByte/s)	Access Time
Conventional Tape:						
Cartridge (.25")	150	12000	104	1.2	92	minutes
IBM 3490 (.5")	800	22860	38	0.9	3000	seconds
Helical Scan Tape:						
Video (8mm)	4600	43200	1638	71	492	45 secs
DAT (4mm)	1300	61000	1870	114	183	20 secs
Magnetic & Optical Disk:						
Hard Disk (5.25")	1200	33528	1880	63	3000	18 ms
IBM 3390 (10.5")	3800	27940	2235	62	4250	20 ms
Sony MO (5.25")	640	24130	18796	454	88	100 ms

Devices: Magnetic Disks

- **Purpose:**
 - Long-term, nonvolatile storage
 - Large, inexpensive, slow level in the storage hierarchy
- **Characteristics:**
 - Seek Time (~8 ms avg)
 - » positional latency
 - » rotational latency
- **Transfer rate**
 - About a sector per ms (5-15 MB/s)
 - Blocks
- **Capacity**
 - Gigabytes
 - Quadruples every 3 years (aerodynamics)

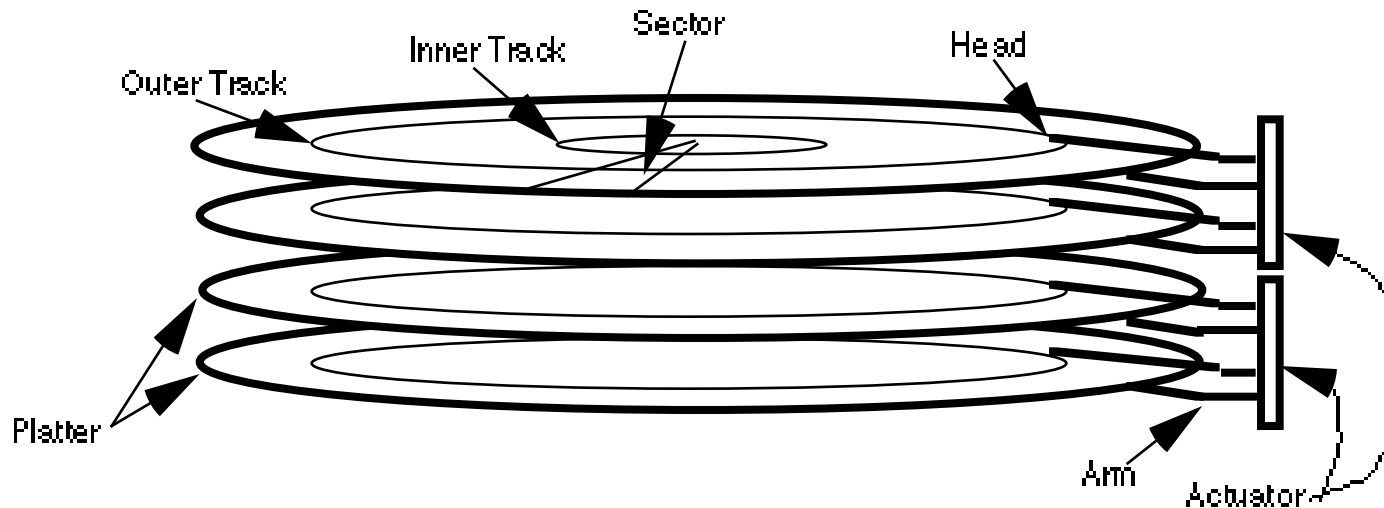


7200 RPM = 120 RPS => 8 ms per rev
ave rot. latency = 4 ms
128 sectors per track => 0.25 ms per sector
1 KB per sector => 16 MB / s

Response time
= Queue + Controller + Seek + Rot + Xfer

Service time

Disk Device Terminology



**Disk Latency = Queuing Time + Controller time +
Seek Time + Rotation Time + Xfer Time**

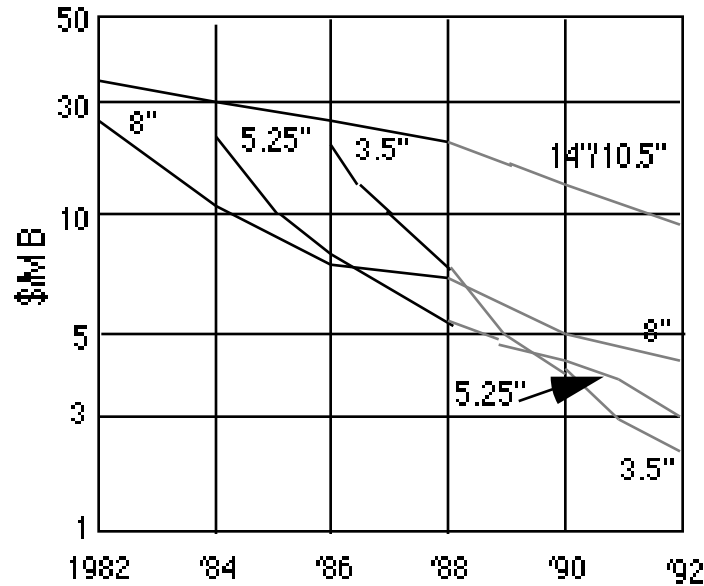
Order of magnitude times for 4K byte transfers:

Seek: 8 ms or less

Rotate: 4.2 ms @ 7200 rpm

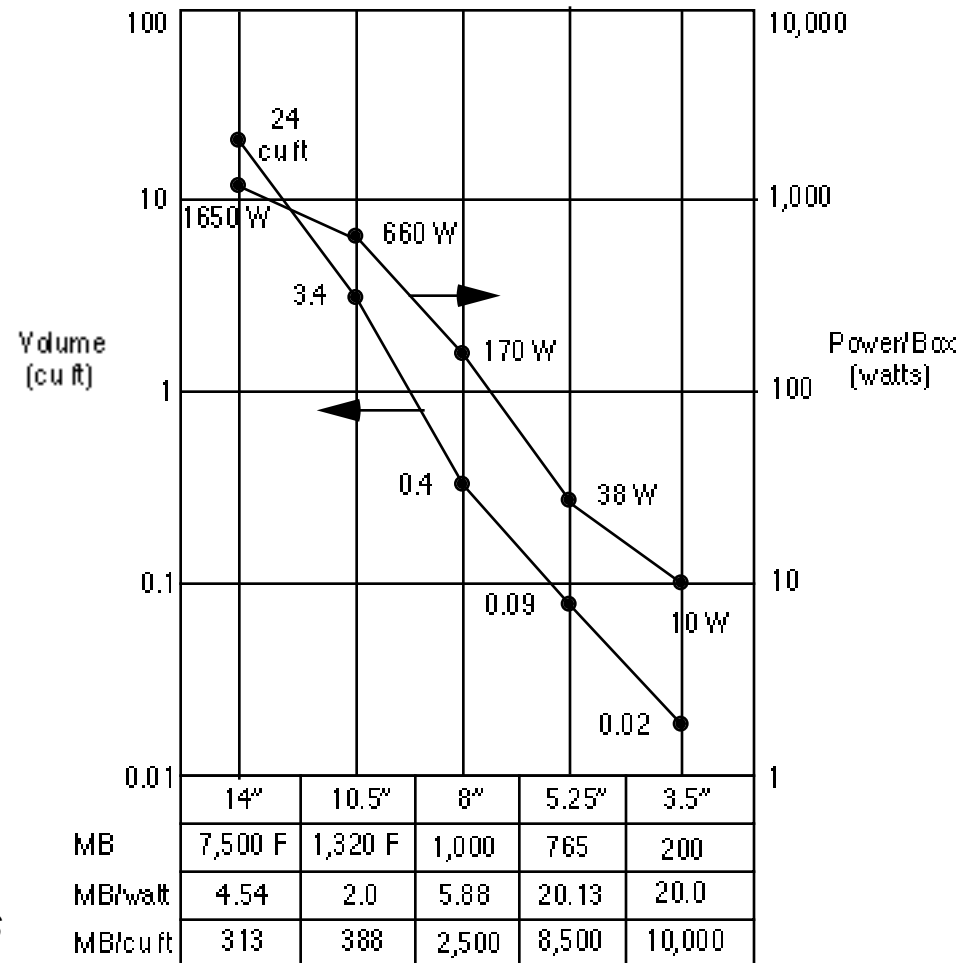
Xfer: 1 ms @ 7200 rpm

Advantages of Small Formfactor Disk Drives



Low cost/MB
High MB/volume
High MB/watt
Low cost/Actuator

Cost and Environmental Efficiencies



CS 252 Administrivia

- **Wed March 4 Quiz 1**
 - Pizza at LaVal's 8:30 – 10PM
- **Email URL of initial project home page to TA?**
 - can share some knowledge gained with other projects
 - allow faculty, TA to make suggestions
 - final “report” will be a URL
 - Limit access to cs.berkeley for now
- **Upcoming events in CS 252**
 - 13-Mar Fri I/O 2: Queuing Theory and Busses**
 - 18-Mar Wed I/O 3: Tertiary Storage & Network Intro**
 - 20-Mar Fri Networks 2: Interface, Switches, Routing**
 - 23-Mar to 27-Mar Spring Break**

Tape vs. Disk

- **Longitudinal tape uses same technology as hard disk; tracks its density improvements**
- **Disk head flies above surface, tape head lies on surface**
- **Disk fixed, tape removable**
- **Inherent cost-performance based on geometries: fixed rotating platters with gaps**
(random access, limited area, 1 media / reader)

VS.

removable long strips wound on spool

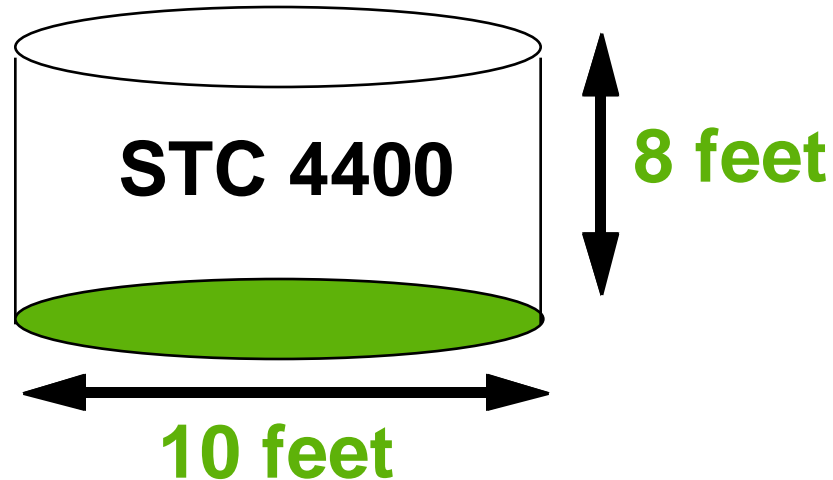
(sequential access, "unlimited" length, multiple / reader)

- **New technology trend: Helical Scan (VCR, Camcoder, DAT)**
Spins head at angle to tape to improve density

Current Drawbacks to Tape

- **Tape wear out:**
 - Helical 100s of passes to 1000s for longitudinal
- **Head wear out:**
 - 2000 hours for helical
- **Both must be accounted for in economic / reliability model**
- **Long rewind, eject, load, spin-up times; not inherent, just no need in marketplace (so far)**
- **Designed for archival**

Automated Cartridge System



6000 x 0.8 GB 3490 tapes = 5 TBytes in 1992
\$500,000 O.E.M. Price

6000 x 10 GB D3 tapes = 60 TBytes in 1998

Library of Congress: all information in the world; in 1992, ASCII of all books = 30 TB

Library vs. Storage

- **Getting books today as quaint as the way I learned to program**
 - punch cards, batch processing
 - wander thru shelves, anticipatory purchasing
- **Cost \$1 per book to check out**
- **\$30 for a catalogue entry**
- **30% of all books never checked out**
- **Write only journals?**
- **Digital library can transform campuses**
- **Will have lecture on getting electronic information**

Relative Cost of Storage Technology—Late 1995/Early 1996

Magnetic Disks

5.25"	9.1 GB	\$2129	\$0.23/MB
		\$1985	\$0.22/MB
3.5"	4.3 GB	\$1199	\$0.27/MB
		\$999	\$0.23/MB
2.5"	514 MB	\$299	\$0.58/MB
	1.1 GB	\$345	\$0.33/MB

Optical Disks

5.25"	4.6 GB	\$1695+199	\$0.41/MB
		\$1499+189	\$0.39/MB

PCMCIA Cards

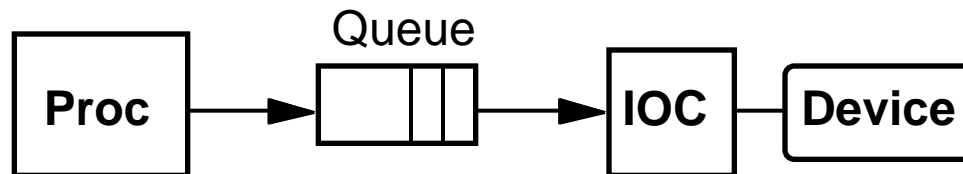
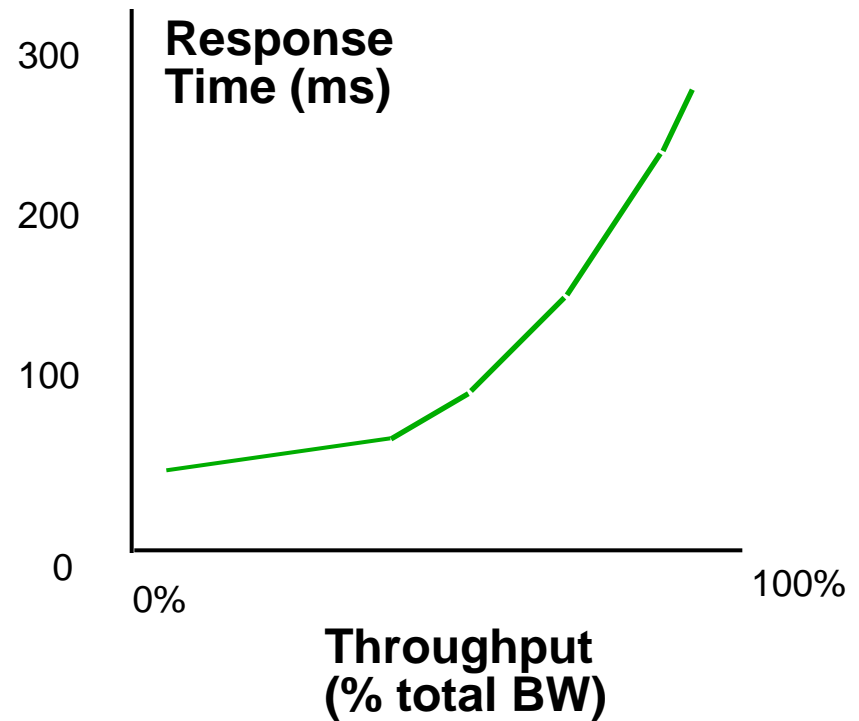
Static RAM	4.0 MB	\$700	\$175/MB
Flash RAM	40.0 MB	\$1300	\$32/MB
	175 MB	\$3600	\$20.50/MB

Lecture Outline

- Historical Context of Storage I/O
- Secondary and Tertiary Storage Devices
- **Storage I/O Performance Measures**
- Processor Interface Issues

Disk I/O Performance

Metrics:
Response Time
Throughput



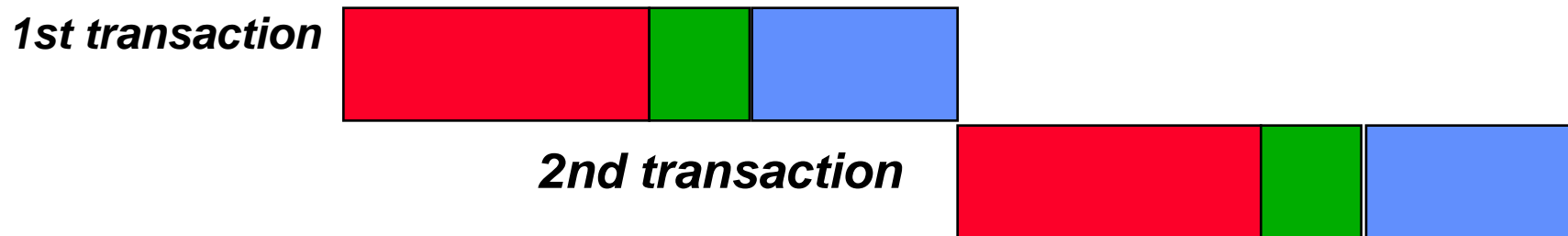
Response time = Queue + Device Service time

Response Time vs. Productivity

- **Interactive environments:**

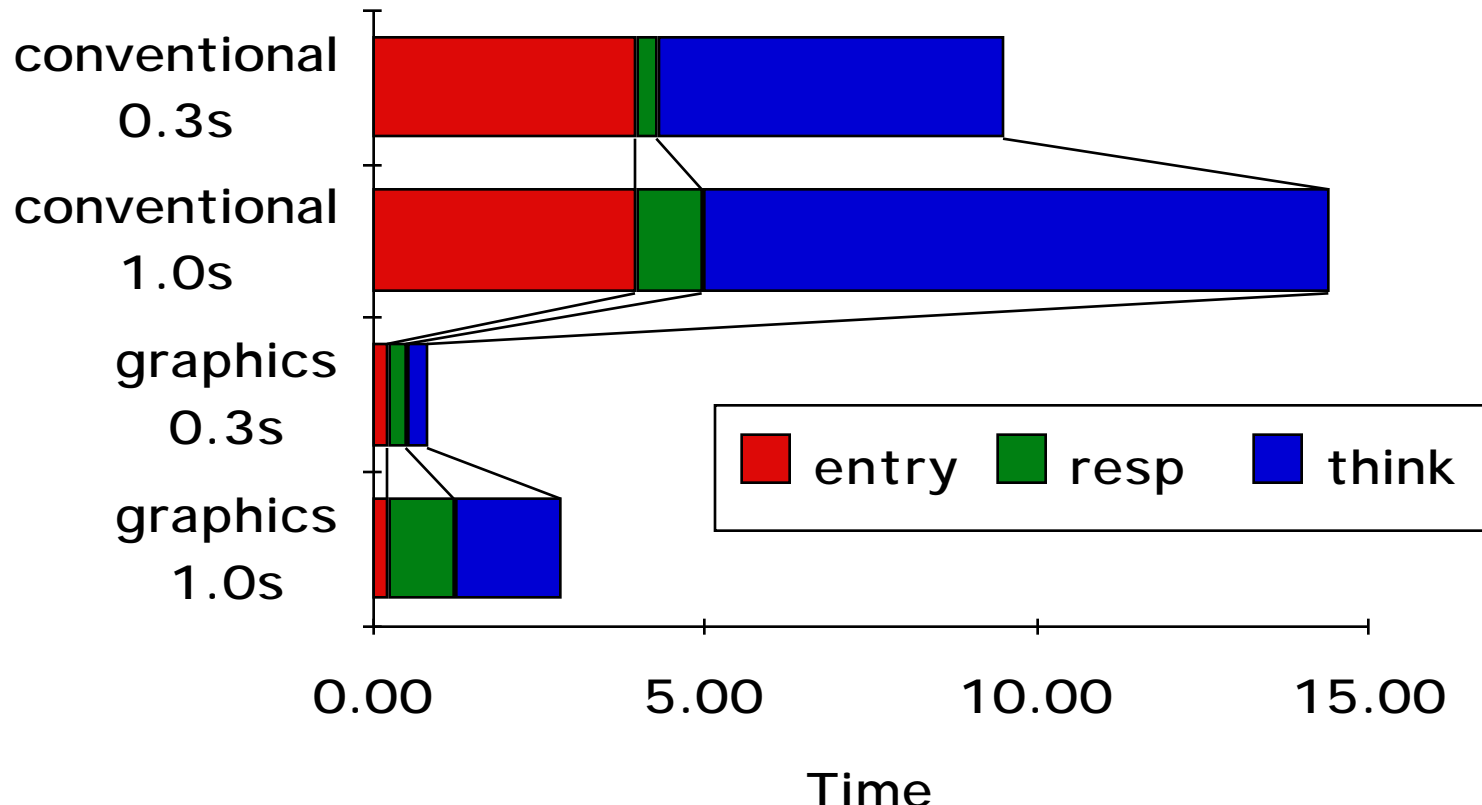
Each interaction or *transaction* has 3 parts:

- **Entry Time**: time for user to enter command
- **System Response Time**: time between user entry & system replies
- **Think Time**: Time from response until user begins next command



- **What happens to transaction time as shrink system response time from 1.0 sec to 0.3 sec?**
 - With Keyboard: 4.0 sec entry, 9.4 sec think time
 - With Graphics: 0.25 sec entry, 1.6 sec think time

Response Time & Productivity



- **0.7sec off response saves 4.9 sec (34%) and 2.0 sec (70%) total time per transaction => greater productivity**
- **Another study: everyone gets more done with faster response, but novice with fast response = expert with slow**

Disk Time Example

- **Disk Parameters:**
 - Transfer size is 8K bytes
 - Advertised average seek is 12 ms
 - Disk spins at 7200 RPM
 - Transfer rate is 4 MB/sec
- **Controller overhead is 2 ms**
- **Assume that disk is idle so no queuing delay**
- **What is Average Disk Access Time for a Sector?**
 - Ave seek + ave rot delay + transfer time + controller overhead
 - $12 \text{ ms} + 0.5 / (7200 \text{ RPM} / 60) + 8 \text{ KB} / 4 \text{ MB/s} + 2 \text{ ms}$
 - $12 + 4.15 + 2 + 2 = 20 \text{ ms}$
- **Advertised seek time assumes no locality: typically 1/4 to 1/3 advertised seek time: 20 ms => 12 ms**

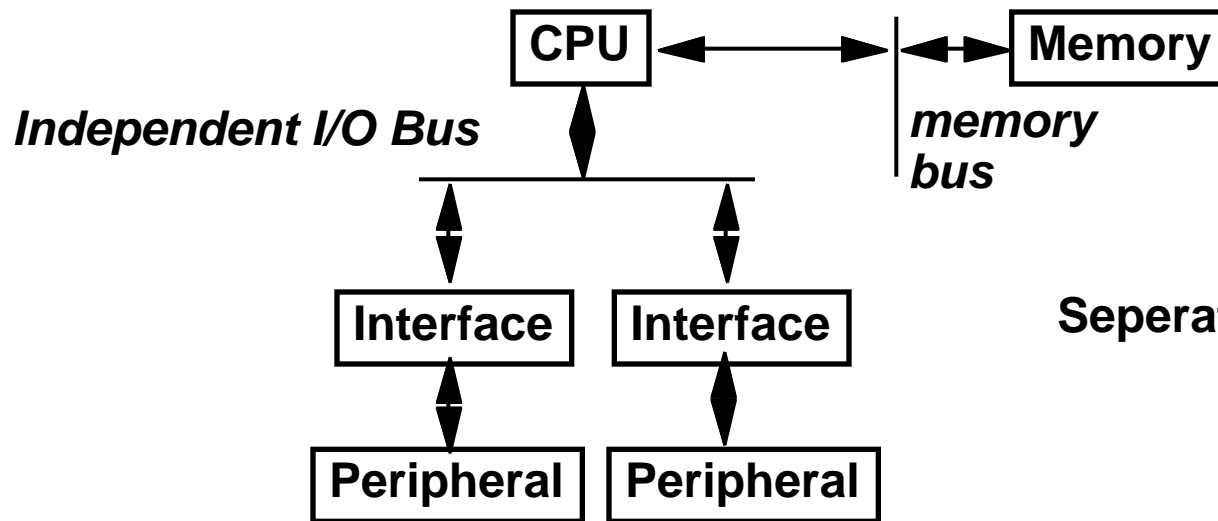
Lecture Outline

- Historical Context of Storage I/O
- Secondary and Tertiary Storage Devices
- Storage I/O Performance Measures
- **Processor Interface Issues**
- Redundant Arrays of Inexpensive Disks (RAID)

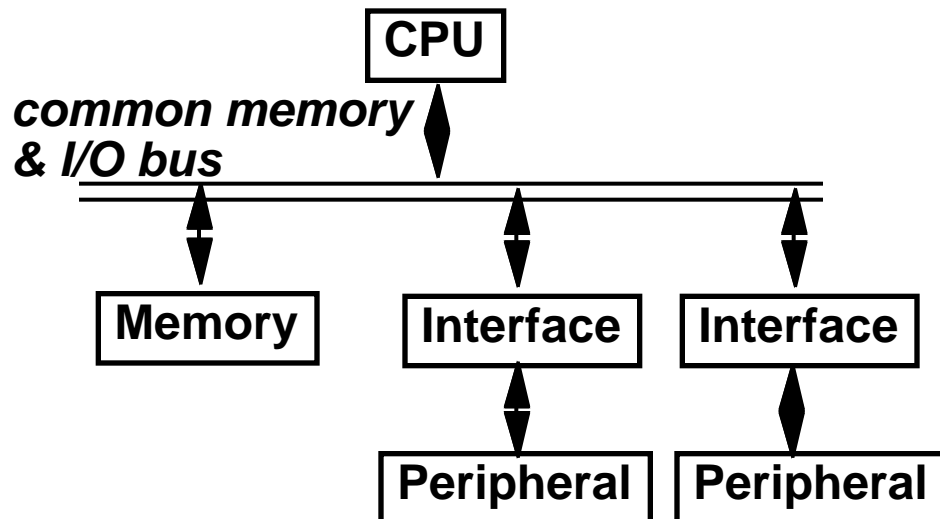
Processor Interface Issues

- **Processor interface**
 - Interrupts
 - Memory mapped I/O
- **I/O Control Structures**
 - Polling
 - Interrupts
 - DMA
 - I/O Controllers
 - I/O Processors
- **Capacity, Access Time, Bandwidth**
- **Interconnections**
 - Busses

I/O Interface



Separate I/O instructions (in,out)



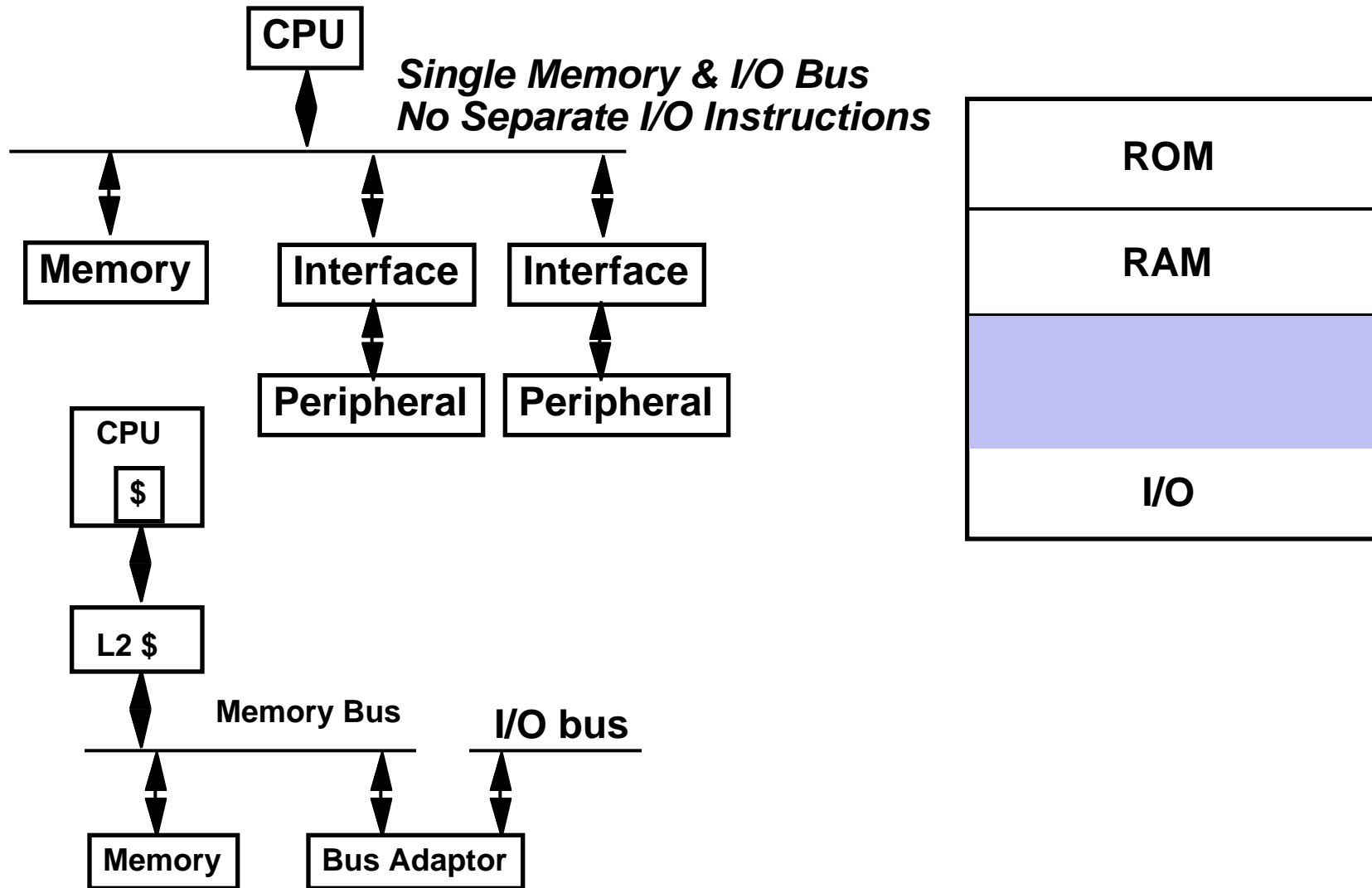
Lines distinguish between I/O and memory transfers

VME bus
Multibus-II
Nubus

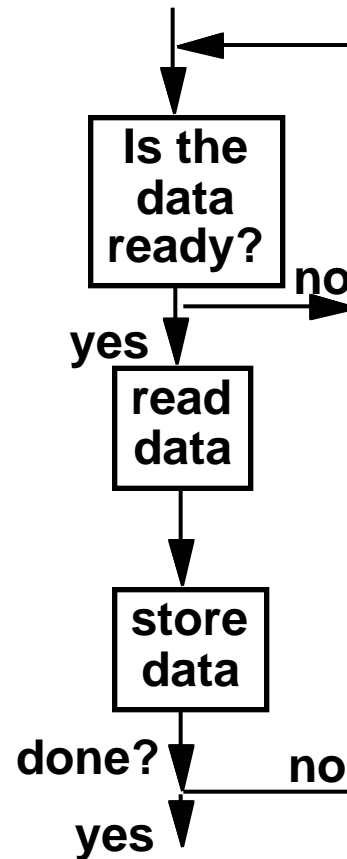
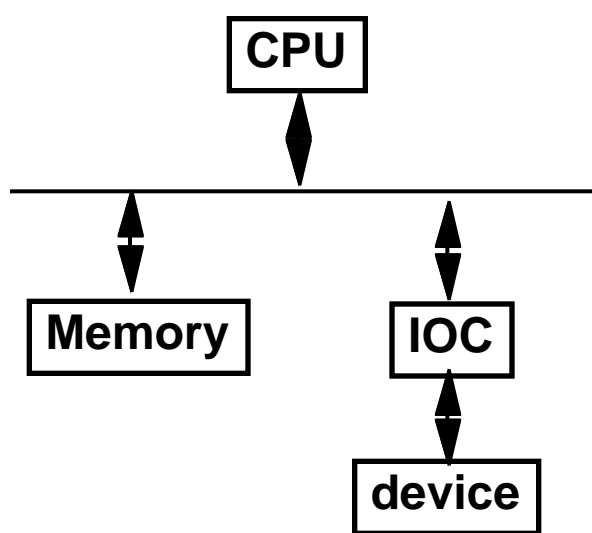
40 Mbytes/sec
optimistically

10 MIP processor
completely
saturates the bus!

Memory Mapped I/O



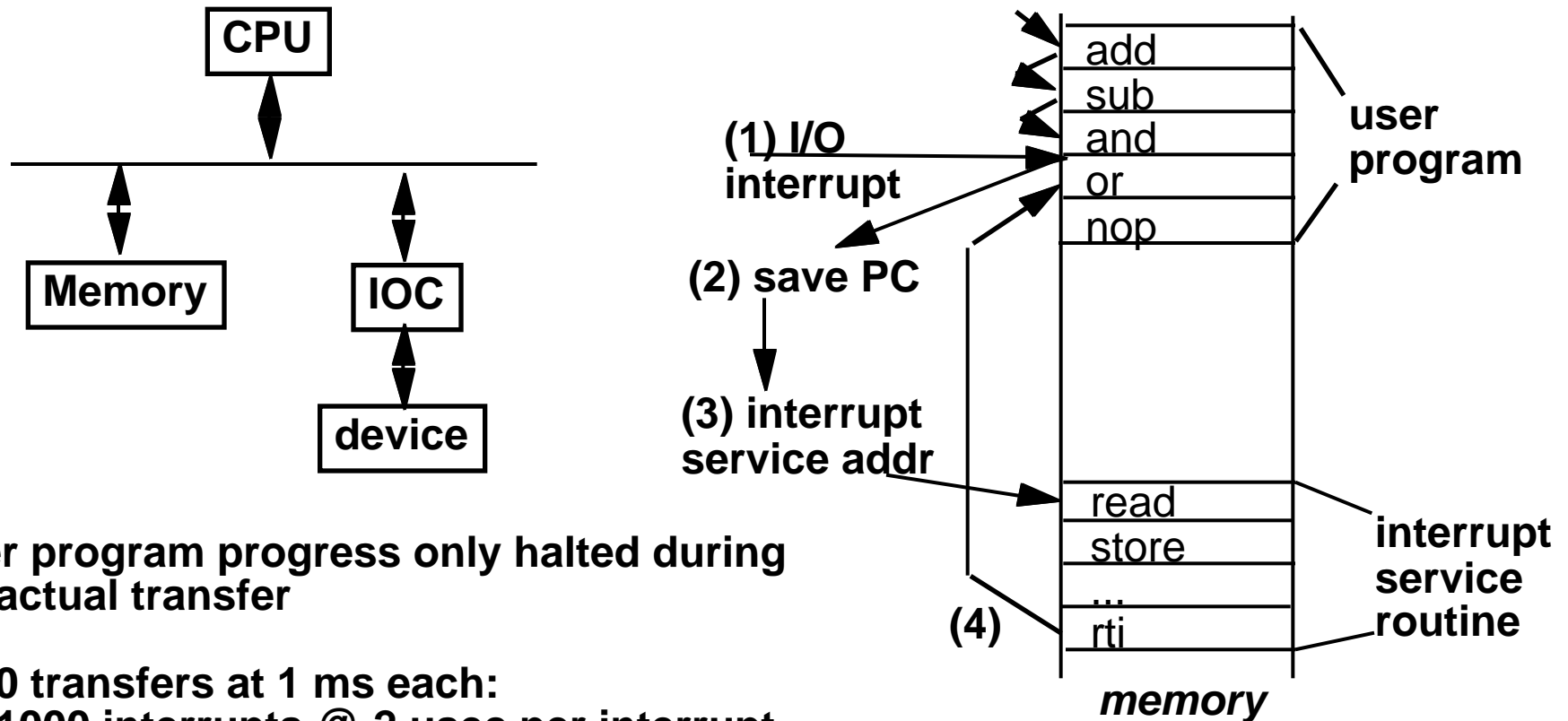
Programmed I/O (Polling)



**busy wait loop
not an efficient
way to use the CPU
unless the device
is very fast!**

**but checks for I/O
completion can be
dispersed among
computationally
intensive code**

Interrupt Driven Data Transfer



User program progress only halted during actual transfer

1000 transfers at 1 ms each:

1000 interrupts @ 2 μ sec per interrupt

1000 interrupt service @ 98 μ sec each = 0.1 CPU seconds

Device xfer rate = 10 MBytes/sec \Rightarrow 0.1×10^{-6} sec/byte \Rightarrow 0.1 μ sec/byte
 \Rightarrow 1000 bytes = 100 μ sec

1000 transfers x 100 μ secs = 100 ms = 0.1 CPU seconds

Still far from device transfer rate! 1/2 in interrupt overhead

Direct Memory Access

Time to do 1000 xfers at 1 msec each:

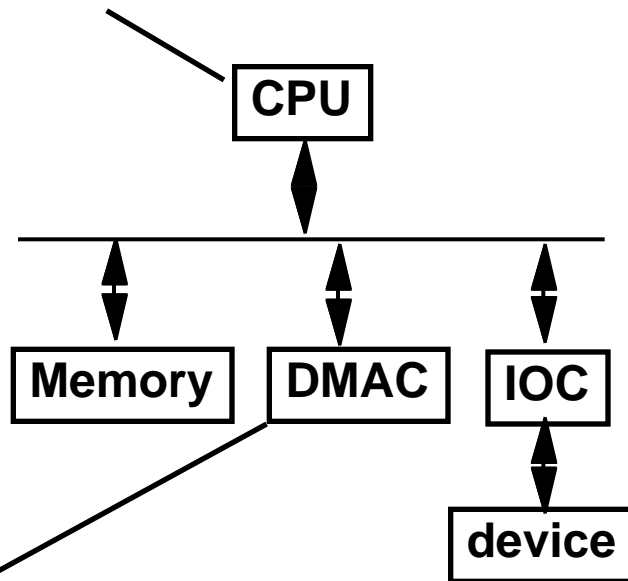
1 DMA set-up sequence @ 50 μ sec

1 interrupt @ 2 μ sec

1 interrupt service sequence @ 48 μ sec

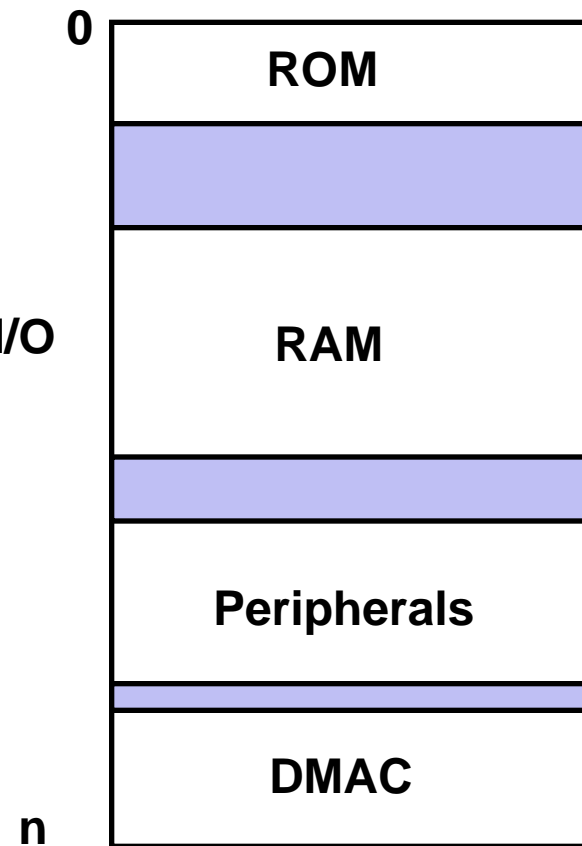
.0001 second of CPU time

CPU sends a starting address, direction, and length count to DMAC. Then issues "start".

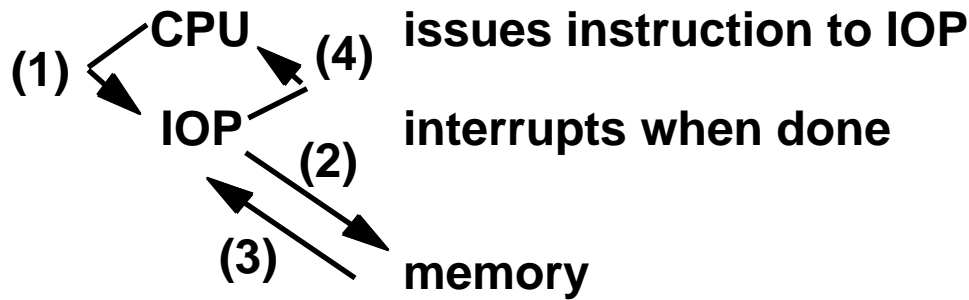
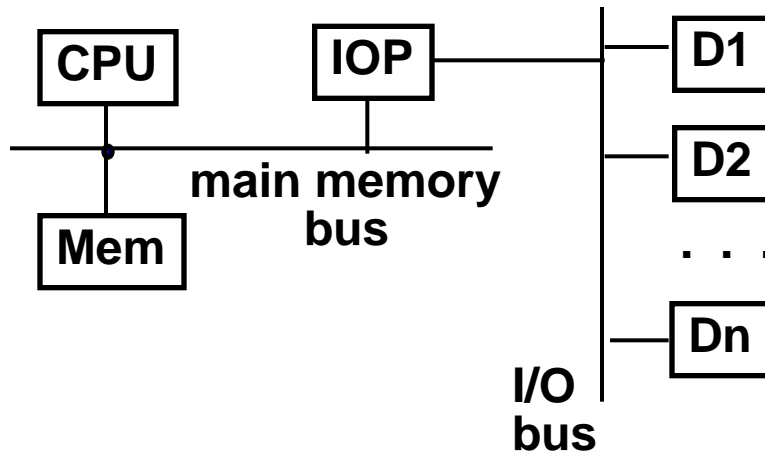


DMAC provides handshake signals for Peripheral Controller, and Memory Addresses and handshake signals for Memory.

Memory Mapped I/O

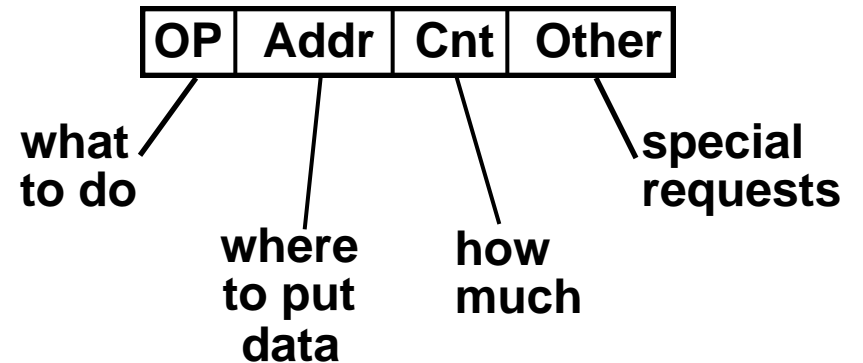
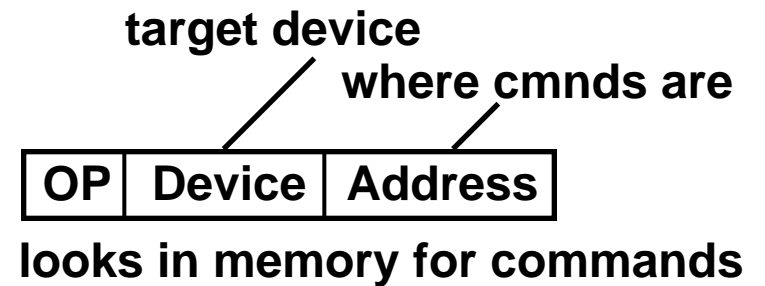


Input/Output Processors



Device to/from memory transfers are controlled by the IOP directly.

IOP steals memory cycles.



Relationship to Processor Architecture

- I/O instructions have largely disappeared
- Interrupt vectors have been replaced by jump tables
PC \leftarrow M [IVA + interrupt number]
PC \leftarrow IVA + interrupt number
- Interrupts:
 - Stack replaced by shadow registers
 - Handler saves registers and re-enables higher priority int's
 - Interrupt types reduced in number; handler must query interrupt controller

Relationship to Processor Architecture

- **Caches required for processor performance cause problems for I/O**
 - Flushing is expensive, I/O pollutes cache
 - Solution is borrowed from shared memory multiprocessors "snooping"
- **Virtual memory frustrates DMA**
- **Load/store architecture at odds with atomic operations**
 - load locked, store conditional
- **Stateful processors hard to context switch**

Summary

- **Disk industry growing rapidly, improves:**
 - bandwidth 40%/yr ,
 - areal density 60%/year, \$/MB faster?
- **queue + controller + seek + rotate + transfer**
- **Advertised average seek time benchmark much greater than average seek time in practice**
- **Response time vs. Bandwidth tradeoffs**
- **Value of faster response time:**
 - 0.7sec off response saves 4.9 sec and 2.0 sec (70%) total time per transaction => greater productivity
 - everyone gets more done with faster response, but novice with fast response = expert with slow
- **Processor Interface: today peripheral processors, DMA, I/O bus, interrupts**

Summary: Relationship to Processor Architecture

- I/O instructions have disappeared
- Interrupt vectors have been replaced by jump tables
- Interrupt stack replaced by shadow registers
- Interrupt types reduced in number
- Caches required for processor performance cause problems for I/O
- Virtual memory frustrates DMA
- Load/store architecture at odds with atomic operations
- Stateful processors hard to context switch

Summary: Storage System Issues

- **Historical Context of Storage I/O**
- **Secondary and Tertiary Storage Devices**
- **Storage I/O Performance Measures**
- **Processor Interface Issues**
- **A Little Queuing Theory**
- **Redundant Arrays of Inexpensive Disks (RAID)**
- **I/O Buses**
- **ABCs of UNIX File Systems**
- **I/O Benchmarks**
- **Comparing UNIX File System Performance**