Lecture 6: Storage Devices, Metrics, RAID, I/O Benchmarks, and Busses

Prof. Fred Chong
ECS 250A Computer Architecture

Winter 1999

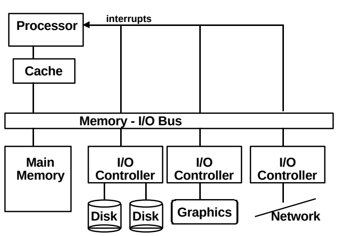
Motivation: Who Cares About I/O?

- 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

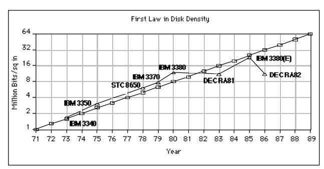
Storage System Issues

- 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)
- ABCs of UNIX File Systems
- I/O Benchmarks
- Comparing UNIX File System Performance
- I/O Busses

I/O Systems



Technology Trends



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

- 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

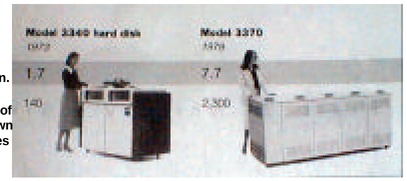
- **1956 IBM Ramac — early 1970s Winchester**
  - Developed for mainframe computers, proprietary interfaces
  - Steady shrink in form factor: 27 in. to 14 in.
- **1970s developments**
  - 5.25 inch floppy disk form factor
  - early emergence of industry standard disk interfaces
    - » ST506, SASI, SMD, 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

FTC.W99 7

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"

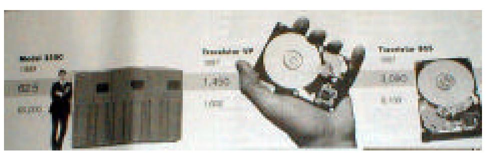
FTC.W99 8

Historical Perspective

- **Late 1980s/Early 1990s:**
  - Laptops, notebooks, (palmtops)
  - 3.5 inch, 2.5 inch, (1.8 inch formfactors)
  - Form factor 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)

FTC.W99 9

Disk History



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

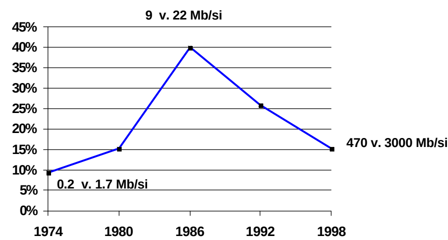
1997:
1450 Mbit/sq. in
2300 MBytes

1997:
3090 Mbit/sq. in
8100 MBytes

source: New York Times, 2/23/98, page C3,
"Makers of disk drives crowd even mroe data into even smaller spaces"

FTC.W99 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 mroe data into even smaller spaces"

FTC.W99 11

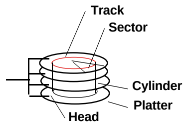
Alternative Data Storage Technologies: Early 1990s

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

FTC.W99 12

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)

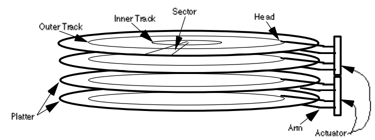


7200 RPM = 120 RPS => 8 ms per rev
ave rot. latency = 4 ms
128 sectors per track => 0.25 ms per sector
1 KB per sector => 16 MB / s

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

Service time

Disk Device Terminology



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

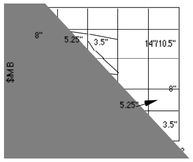
Order of magnitude times for 4K byte transfers:

Seek: 8 ms or less

Rotate: 4.2 ms @ 7200 rpm

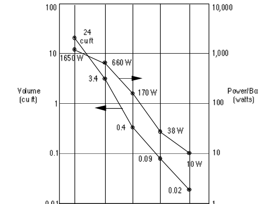
Xfer: 1 ms @ 7200 rpm

Advantages of Small Formfactor Disk Drives



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

Cost and Environmental Efficiencies



	14"	10.5"	8"	5.25"	3.5"
MB	7,500 F	1,320 F	1,000	765	200
MB/watt	4.54	2.0	5.88	20.13	20.0
MB/cu ft	313	388	2,500	8,500	10,000

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)

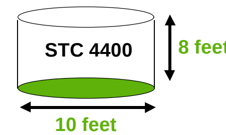
vs.
removable long strips wound on spool
(sequential access, "unlimited" length, multiple / reader)

- New technology trend:
  Helical Scan (VCR, Camcorder, 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 $1985	$0.23/MB $0.22/MB
3.5”	4.3 GB	$1199 $999	$0.27/MB $0.23/MB
2.5”	514 MB 1.1 GB	$299 $345	$0.58/MB $0.33/MB

Optical Disks

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

PCMCIA Cards

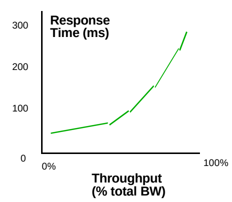
Static RAM	4.0 MB	$700	$175/MB
Flash RAM	40.0 MB	$1300	$32/MB
	175 MB	$3600	$20.50/MB

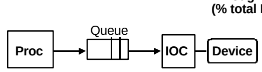
Outline

- 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)
- ABCs of UNIX File Systems
- I/O Benchmarks
- Comparing UNIX File System Performance
- I/O Busses

Disk I/O Performance

Metrics:
Response Time
Throughput

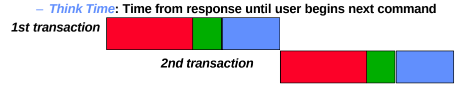




Response time = Queue + Device Service time

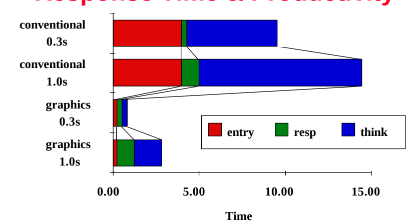
Response Time vs. Productivity

- 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 ms + 0.5/(7200 RPM/60) + 8 KB/4 MB/s + 2 ms
  - 12 + 4.15 + 2 + 2 = 20 ms
- Advertised seek time assumes no locality: typically 1/4 to 1/3 advertised seek time: 20 ms => 12 ms

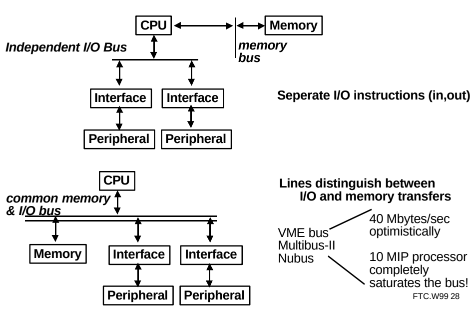
Outline

- 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)
- ABCs of UNIX File Systems
- I/O Benchmarks
- Comparing UNIX File System Performance
- I/O Busses

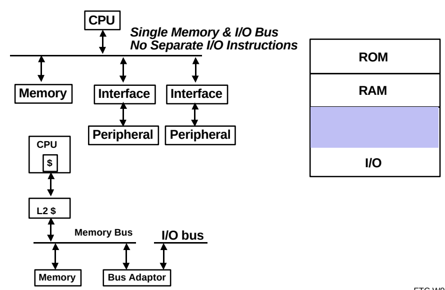
Processor Interface Issues

- 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

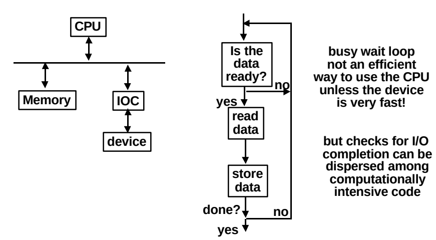
I/O Interface



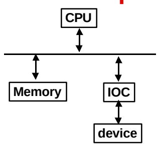
Memory Mapped I/O

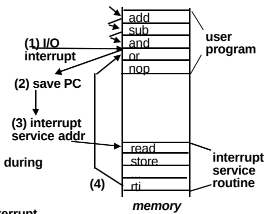


Programmed I/O (Polling)



Interrupt Driven Data Transfer





User program progress only halted during actual transfer

1000 transfers at 1 ms each:
1000 interrupts @ 2 µsec per interrupt
1000 interrupt service @ 98 µsec each = 0.1 CPU seconds

Device xfer rate = 10 MBytes/sec => 0 .1 x 10^{-6} sec/byte => 0.1 µsec/byte
=> 1000 bytes = 100 µsec
1000 transfers x 100 µsecs = 100 ms = 0.1 CPU seconds

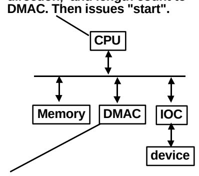
Still far from device transfer rate! 1/2 in interrupt overhead

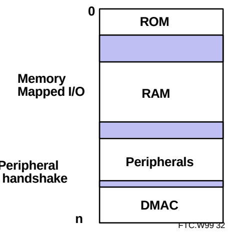
Direct Memory Access

Time to do 1000 xfers at 1 msec each:
1 DMA set-up sequence @ 50 µsec
1 interrupt @ 2 µsec
1 interrupt service sequence @ 48 µsec

.0001 second of CPU time

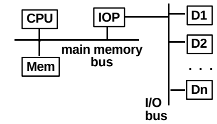
CPU sends a starting address, direction, and length count to DMAC. Then issues "start".

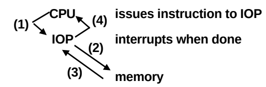




DMAC provides handshake signals for Peripheral Controller, and Memory Addresses and handshake signals for Memory.

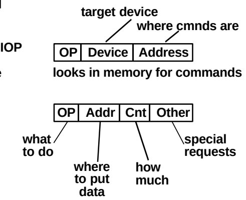
Input/Output Processors





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 <- M [IVA + interrupt number]
  PC <- 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 ,
  - area 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

FTC.W99 37

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)
- ABCs of UNIX File Systems
- I/O Benchmarks
- Comparing UNIX File System Performance
- I/O Busses

FTC.W99 38

Network Attached Storage

Decreasing Disk Diameters

14" » 10" » 8" » 5.25" » 3.5" » 2.5" » 1.8" » 1.3" » . . .
high bandwidth disk systems based on arrays of disks

High Performance *Storage Service* on a High Speed Network

Network provides well defined physical and logical interfaces: *separate CPU and storage system!*

Network File Services

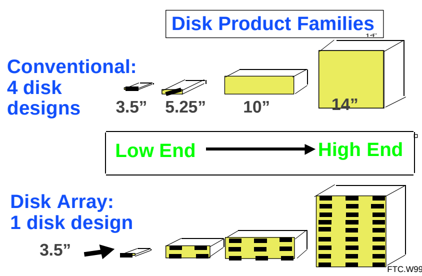
OS structures supporting remote file access

3 Mb/s » 10Mb/s » 50 Mb/s » 100 Mb/s » 1 Gb/s » *10 Gb/s*
networks capable of sustaining high bandwidth transfers

Increasing Network Bandwidth

FTC.W99 39

Manufacturing Advantages of Disk Arrays



FTC.W99 40

Replace Small # of Large Disks with Large # of Small Disks! (1988 Disks)

	IBM 3390 (K)	IBM 3.5" 0061	*x70*
Data Capacity	20 GBytes	320 MBytes	*23 GBytes*
Volume	97 cu. ft.	0.1 cu. ft.	*11 cu. ft.*
Power	3 KW	11 W	*1 KW*
Data Rate	15 MB/s	1.5 MB/s	*120 MB/s*
I/O Rate	600 I/Os/s	55 I/Os/s	*3900 IOs/s*
MTTF	250 KHrs	50 KHrs	*??? Hrs*
Cost	$250K	$2K	*$150K*

Disk Arrays have potential for
- large data and I/O rates
- high MB per cu. ft., high MB per KW
- reliability?

FTC.W99 41

Array Reliability

- Reliability of N disks = Reliability of 1 Disk ÷ N
  - 50,000 Hours ÷ 70 disks = 700 hours
  - Disk system MTTF: Drops from 6 years to 1 month!
- Arrays (without redundancy) too unreliable to be useful!

Hot spares support reconstruction in parallel with access: very high media availability can be achieved

FTC.W99 42

Redundant Arrays of Disks

- Files are "striped" across multiple spindles
- Redundancy yields high data availability

Disks will fail

Contents reconstructed from data redundantly stored in the array

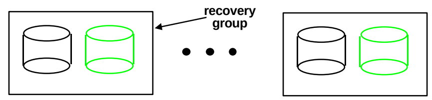
→ Capacity penalty to store it

→ Bandwidth penalty to update

Techniques:
- Mirroring/Shadowing (high capacity cost)
- Horizontal Hamming Codes (overkill)
- Parity & Reed-Solomon Codes
- Failure Prediction (no capacity overhead!) *VaxSimPlus — Technique is controversial*

FTC.W99 43

Redundant Arrays of Disks RAID 1: Disk Mirroring/Shadowing

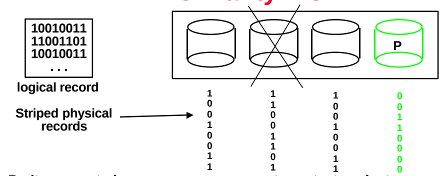


- Each disk is fully duplicated onto its "shadow"
  Very high availability can be achieved
- Bandwidth sacrifice on write:
  Logical write = two physical writes
- Reads may be optimized
- Most expensive solution: 100% capacity overhead

Targeted for high I/O rate , high availability environments

FTC.W99 44

Redundant Arrays of Disks RAID 3: Parity Disk

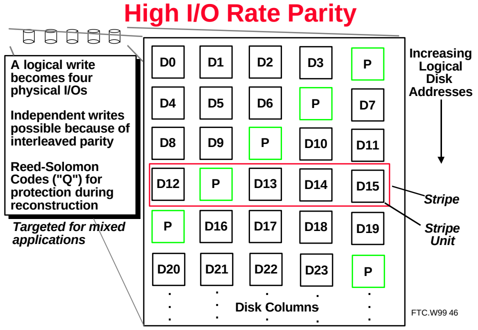


- Parity computed across recovery group to protect against hard disk failures
  - 33% capacity cost for parity in this configuration
  - wider arrays reduce capacity costs, decrease expected availability, increase reconstruction time
- Arms logically synchronized, spindles rotationally synchronized
  - logically a single high capacity, high transfer rate disk

Targeted for high bandwidth applications: Scientific, Image Processing

FTC.W99 45

Redundant Arrays of Disks RAID 5+: High I/O Rate Parity

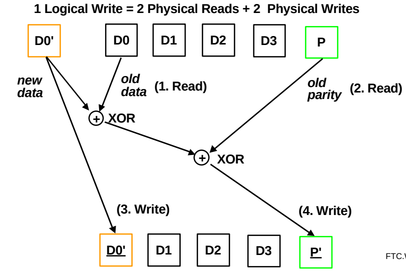


FTC.W99 46

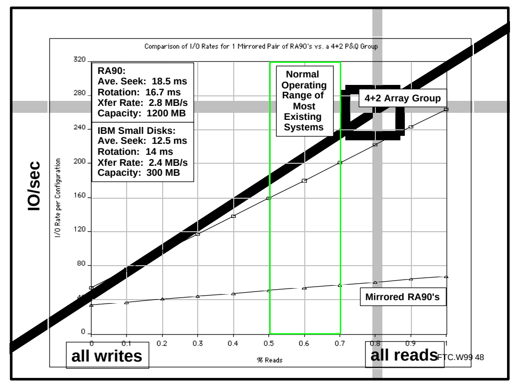
Problems of Disk Arrays: Small Writes

RAID-5: Small Write Algorithm

1 Logical Write = 2 Physical Reads + 2 Physical Writes

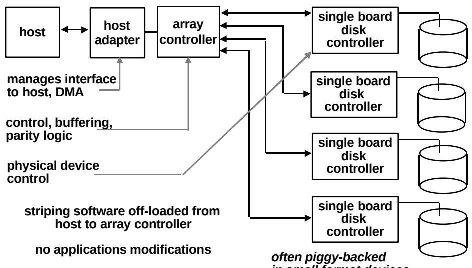


FTC.W99 47



FTC.W99 48

Subsystem Organization



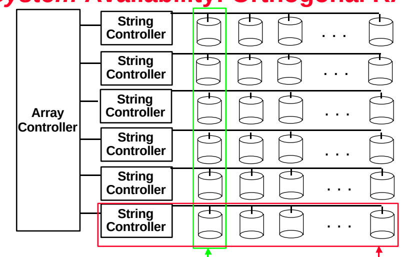
striping software off-loaded from host to array controller

no applications modifications

no reduction of host performance

FTC.W99 49

System Availability: Orthogonal RAIDs



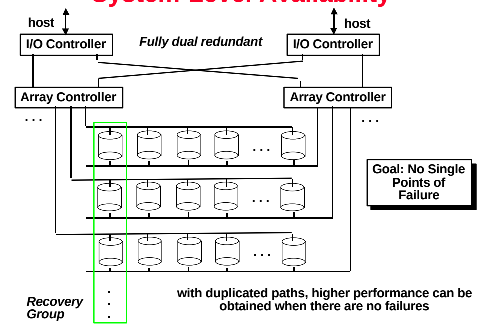
Data Recovery Group: unit of data redundancy

Redundant Support Components: fans, power supplies, controller, cables

End to End Data Integrity: internal parity protected data paths

FTC.W99 50

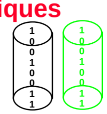
System-Level Availability



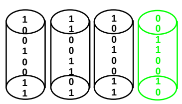
with duplicated paths, higher performance can be obtained when there are no failures

Summary: Redundant Arrays of Disks (RAID) Techniques

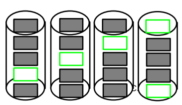
- *Disk Mirroring, Shadowing (RAID 1)*
  - Each disk is fully duplicated onto its "shadow"
  - Logical write = two physical writes
  - 100% capacity overhead



- *Parity Data Bandwidth Array (RAID 3)*
  - Parity computed horizontally
  - Logically a single high data bw disk



- *High I/O Rate Parity Array (RAID 5)*
  - Interleaved parity blocks
  - Independent reads and writes
  - Logical write = 2 reads + 2 writes
  - Parity + Reed-Solomon codes



Outline

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

FTC.W99 53

ABCs of UNIX File Systems

- **Key Issues**
  - File vs. Raw I/O
  - File Cache Size Policy
  - Write Policy
  - Local Disk vs. Server Disk
- **File vs. Raw:**
  - File system access is the norm: standard policies apply
  - Raw: alternate I/O system to avoid file system, used by data bases
- **File Cache Size Policy**
  - % of main memory dedicated to file cache is fixed at system generation (e.g., 10%)
  - % of main memory for file cache varies depending on amount of file I/O (e.g., up to 80%)

FTC.W99 54

ABCs of UNIX File Systems

- Write Policy
  - File Storage should be permanent; either write immediately or flush file cache after fixed period (e.g., 30 seconds)
  - Write Through with Write Buffer
  - Write Back
  - Write Buffer often confused with Write Back
    - » Write Through with Write Buffer, all writes go to disk
    - » Write Through with Write Buffer, writes are asynchronous, so processor doesn't have to wait for disk write
    - » Write Back will combine multiple writes to same page; hence can be called Write Cancelling

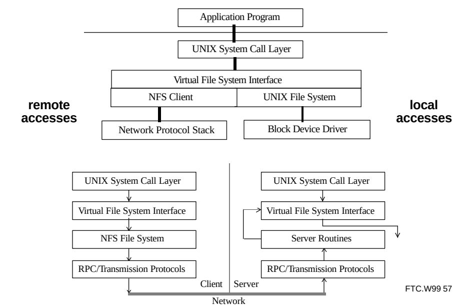
FTC.W99 55

ABCs of UNIX File Systems

- Local vs. Server
  - Unix File systems have historically had different policies (and even file systems) for local client vs. remote server
  - NFS local disk allows 30 second delay to flush writes
  - NFS server disk writes through to disk on file close
  - Cache coherency problem if allow clients to have file caches in addition to server file cache
    - » NFS just writes through on file close
      Stateless protocol: periodically get new copies of file blocks
    - » Other file systems use cache coherency with write back to check state and selectively invalidate or update

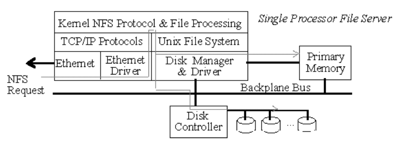
FTC.W99 56

Network File Systems



FTC.W99 57

Typical File Server Architecture



Limits to performance: data copying

read data staged from device to primary memory

copy again into network packet templates

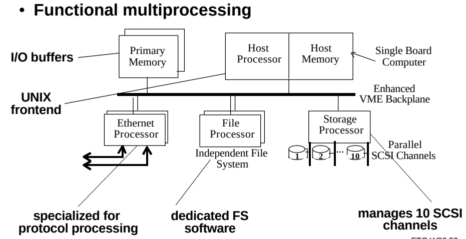
copy yet again to network interface

No specialization for fast processing between network and disk

FTC.W99 58

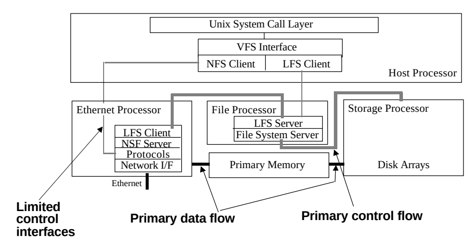
AUSPEX NS5000 File Server

- Special hardware/software architecture for high performance NFS I/O
- Functional multiprocessing



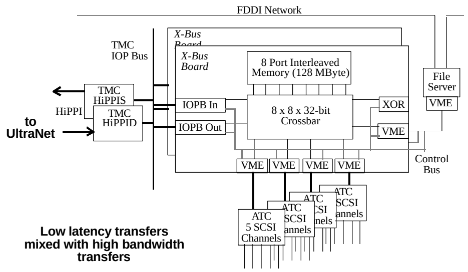
FTC.W99 59

AUSPEX Software Architecture



FTC.W99 60

Berkeley RAID-II Disk Array File Server



Low latency transfers mixed with high bandwidth transfers

"Diskless Supercomputers"

to 120 disk drives

I/O Benchmarks

- For better or worse, benchmarks shape a field
  - Processor benchmarks classically aimed at response time for fixed sized problem
  - I/O benchmarks typically measure throughput, possibly with upper limit on response times (or 90% of response times)
- What if fix problem size, given 60%/year increase in DRAM capacity?

Benchmark	*Size of Data*	*% Time I/O*	*Year*
I/OStones	1 MB	26%	1990
Andrew	4.5 MB	4%	1988

  - Not much time in I/O
  - Not measuring disk (or even main memory)

I/O Benchmarks

- Alternative: self-scaling benchmark; automatically and dynamically increase aspects of workload to match characteristics of system measured
  - Measures wide range of current & future
- Describe three self-scaling benchmarks
  - Transacition Processing: TPC-A, TPC-B, TPC-C
  - NFS: SPEC SFS (LADDIS)
  - Unix I/O: Willy

I/O Benchmarks: Transaction Processing

- Transaction Processing (TP) (or On-line TP=OLTP)
  - Changes to a large body of shared information from many terminals, with the TP system guaranteeing proper behavior on a failure
  - If a bank's computer fails when a customer withdraws money, the TP system would guarantee that the account is debited if the customer received the money and that the account is unchanged if the money was not received
  - Airline reservation systems & banks use TP
- Atomic transactions makes this work
- Each transaction => 2 to 10 disk I/Os & 5,000 and 20,000 CPU instructions per disk I/O
  - Efficiency of TP SW & avoiding disks accesses by keeping information in main memory
- Classic metric is Transactions Per Second (TPS)
  - Under what workload? how machine configured?

I/O Benchmarks: Transaction Processing

- Early 1980s great interest in OLTP
  - Expecting demand for high TPS (e.g., ATM machines, credit cards)
  - Tandem's success implied medium range OLTP expands
  - Each vendor picked own conditions for TPS claims, report only CPU times with widely different I/O
  - Conflicting claims led to disbelief of all benchmarks=> chaos
- 1984 Jim Gray of Tandem distributed paper to Tandem employees and 19 in other industries to propose standard benchmark
- Published "A measure of transaction processing power," Datamation, 1985 by Anonymous et. al
  - To indicate that this was effort of large group
  - To avoid delays of legal department of each author's firm
  - Still get mail at Tandem to author

I/O Benchmarks: TP by Anon et. al

- Proposed 3 standard tests to characterize commercial OLTP
  - TP1: OLTP test, DebitCredit, simulates ATMs (TP1)
  - Batch sort
  - Batch scan
- Debit/Credit:
  - One type of transaction: 100 bytes each
  - Recorded 3 places: account file, branch file, teller file + events recorded in history file (90 days)
    - » 15% requests for different branches
  - Under what conditions, how report results?

I/O Benchmarks: TP1 by Anon et. al

- DebitCredit Scalability: size of account, branch, teller, history function of throughput

TPS	Number of ATMs	Account-file size
10	1,000	0.1 GB
100	10,000	1.0 GB
1,000	100,000	10.0 GB
10,000	1,000,000	100.0 GB

  - Each input TPS =>100,000 account records, 10 branches, 100 ATMs
  - Accounts must grow since a person is not likely to use the bank more frequently just because the bank has a faster computer!
- Response time: 95% transactions take Š 1 second
- Configuration control: just report price (initial purchase price + 5 year maintenance = cost of ownership)
- By publishing, in public domain

FTC.W99 67

I/O Benchmarks: TP1 by Anon et. al

- Problems
  - Often ignored the user network to terminals
  - Used transaction generator with no think time; made sense for database vendors, but not what customer would see
- Solution: Hire auditor to certify results
  - Auditors soon saw many variations of ways to trick system
- Proposed minimum compliance list (13 pages); still, DEC tried IBM test on different machine with poorer results than claimed by auditor
- Created Transaction Processing Performance Council in 1988: founders were CDC, DEC, ICL, Pyramid, Stratus, Sybase, Tandem, and Wang; 46 companies today
- Led to TPC standard benchmarks in 1990, www.tpc.org

FTC.W99 68

I/O Benchmarks: Old TPC Benchmarks

- TPC-A: Revised version of TP1/DebitCredit
  - Arrivals: Random (TPC) vs. uniform (TP1)
  - Terminals: Smart vs. dumb (affects instruction path length)
  - ATM scaling: 10 terminals per TPS vs. 100
  - Branch scaling: 1 branch record per TPS vs. 10
  - Response time constraint: 90% Š2 seconds vs. 95% Š1
  - Full disclosure, approved by TPC
  - Complete TPS vs. response time plots vs. single point
- TPC-B: Same as TPC-A but without terminals—batch processing of requests
  - Response time makes no sense: plots tps vs. residence time (time of transaction resides in system)
- These have been withdrawn as benchmarks

FTC.W99 69

I/O Benchmarks: TPC-C Complex OLTP

- Models a wholesale supplier managing orders
- Order-entry conceptual model for benchmark
- Workload = 5 transaction types
- Users and database scale linearly with throughput
- Defines full-screen end-user interface
- Metrics: new-order rate (tpmC) and price/performance ($/tpmC)
- Approved July 1992

FTC.W99 70

I/O Benchmarks: TPC-D Complex Decision Support Workload

- OLTP: business operation
- Decision support: business analysis (historical)
- Workload = 17 adhoc transactions
  - e,g., Impact on revenue of eliminating company-wide discount?
- Synthetic generator of data
- Size determined by Scale Factor: 100 GB, 300 GB, 1 TB, 3 TB, 10 TB
- Metrics: “Queries per Gigabyte Hour”
  Power (QppD@Size) = 3600 x SF / Geo. Mean of queries
  Throughput (QthD@Size) = 17 x SF / (time/3600)
  Price/Performance ($/QphD@Size) =
  $/ geo. mean(QppD@Size, QthD@Size)
- Report time to load database (indices, stats) too
- Approved April 1995

FTC.W99 71

I/O Benchmarks: TPC-W Transactional Web Benchmark

- Represent any business (retail store, software distribution, airline reservation, electronic stock trades, etc.) that markets and sells over the Internet/ Intranet
- Measure systems supporting users browsing, ordering, and conducting transaction oriented business activities.
- Security (including user authentication and data encryption) and dynamic page generation are important
- Before: processing of customer order by terminal operator working on LAN connected to database system
- Today: customer accesses company site over Internet connection, browses both static and dynamically generated Web pages, and searches the database for product or customer information. Customer also initiate, finalize and check on product orders and deliveries.
- Started 1/97; hope to release Fall, 1998

FTC.W99 72

TPC-C Performance tpm(c)

Rank		Config	tpmC	$/tpmC	Database
1	IBM	RS/6000 SP (12 node x 8-way)	57,053.80	$147.40	Oracle8 8.0.4
2	HP	HP 9000 V2250 (16-way)	52,117.80	$81.17	Sybase ASE
3	Sun	Ultra E6000 c/s (2 node x 22-way)	51,871.62	$134.46	Oracle8 8.0.3
4	HP	HP 9000 V2200 (16-way)	39,469.47	$94.18	Sybase ASE
5	Fujitsu	GRANPOWER 7000 Model 800	34,116.93	$57,883.00	Oracle8
6	Sun	Ultra E6000 c/s (24-way)	31,147.04	$108.90	Oracle8 8.0.3
7	Digital	AlphaS8400 (4 node x 8-way)	30,390.00	$305.00	Oracle7 V7.3
8	SGI	Origin2000 Server c/s (28-way)	25,309.20	$139.04	INFORMIX
9	IBM	AS/400e Server (12-way)	25,149.75	$128.00	DB2
10	Digital	AlphaS8400 5/625 (10-way)	24,537.00	$110.48	Sybase SQL

FTC.W99 73

TPC-C Price/Performance $/tpm(c)

Rank		Config	$/tpmC	tpmC	Database
1	Acer	AcerAltos 19000Pro4	$27.25	11,072.07	M/S SQL 6.5
2	Dell	PowerEdge 6100 c/s	$29.55	10,984.07	M/S SQL 6.5
3	Compaq	ProLiant 5500 c/s	$33.37	10,526.90	M/S SQL 6.5
4	ALR	Revolution 6x6 c/s	$35.44	13,089.30	M/S SQL 6.5
5	HP	NetServer LX Pro	$35.82	10,505.97	M/S SQL 6.5
6	Fujitsu	teamserver M796i	$37.62	13,391.13	M/S SQL 6.5
7	Fujitsu	GRANPOWER 5000 Model 670	$37.62	13,391.13	M/S SQL 6.5
8	Unisys	Aquanta HS/6 c/s	$37.96	13,089.30	M/S SQL 6.5
9	Compaq	ProLiant 7000 c/s	$39.25	11,055.70	M/S SQL 6.5
10	Unisys	Aquanta HS/6 c/s	$39.39	12,026.07	M/S SQL 6.5

FTC.W99 74

TPC-D Performance/Price 300 GB

Rank	Config.	Qppd	QthD	$/QphD	Database
1	NCR WorldMark 5150	9,260.0	3,117.0	2,172.00	Teradata
2	HP 9000 EPS22 (16 node)	5,801.2	2,829.0	1,982.00	Informix-XPS
3	DG AViiON AV20000	3,305.8	1,277.7	1,319.00	Oracle8 v8.0.4
4	Sun - Ultra Enterprise 6000	3,270.6	1,477.8	1,553.00	Informix-XPS
5	Sequent NUMA-Q 2000 (32 way)	3,232.3	1,097.8	3,283.00	Oracle8 v8.0.4

Rank	Config.	Qppd	QthD	$/QphD	Database
1	DG AViiON AV20000	3,305.8	1,277.7	1,319.00	Oracle8 v8.0.4
2	Sun Ultra Enterprise 6000	3,270.6	1,477.8	1,553.00	Informix-XPS
3	HP 9000 EPS22 (16 node)	5,801.2	2,829.0	1,982.00	Informix-XPS
4	NCR WorldMark 5150	9,260.0	3,117.0	2,172.00	Teradata
5	Sequent NUMA-Q 2000 (32 way)	3,232.3	1,097.8	3,283.00	Oracle8 v8.0.4

FTC.W99 75

TPC-D Performance 1TB

Rank	Config.	Qppd	QthD	$/QphD	Database
1	Sun Ultra E6000 (4 x 24-way)	12,931.9	5,850.3	1,353.00	Infomix Dyn
2	NCR WorldMark (32 x 4-way)	12,149.2	3,912.3	2103.00	Teradata
3	IBM RS/6000 SP (32 x 8-way)	7,633.0	5,155.4	2095.00	DB2 UDB, V5

- **NOTE: Inappropriate to compare results from different database sizes.**

FTC.W99 76

TPC-D Performance 1TB

Rank	Config.	Qppd	QthD	$/QphD	Database
1	Sun Ultra E6000 (4 x 24-way)	12,931.9	5,850.3	1,353.00	Infomix Dyn
2	NCR WorldMark (32 x 4-way)	12,149.2	3,912.3	2103.00	Teradata
3	IBM RS/6000 SP (32 x 8-way)	7,633.0	5,155.4	2095.00	DB2 UDB, V5

FTC.W99 77

SPEC SFS/LADDIS Predecessor: NFSstones

- **NFSStones: synthetic benchmark that generates series of NFS requests from single client to test server: reads, writes, & commands & file sizes from other studies**
  - Problem: 1 client could not always stress server
  - Files and block sizes not realistic
  - Clients had to run SunOS

FTC.W99 78

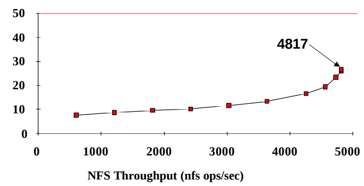
SPEC SFS/LADDIS

- 1993 Attempt by NFS companies to agree on standard benchmark: Legato, Auspex, Data General, DEC, Interphase, Sun. Like NFSstones but
  - Run on multiple clients & networks (to prevent bottlenecks)
  - Same caching policy in all clients
  - Reads: 85% full block & 15% partial blocks
  - Writes: 50% full block & 50% partial blocks
  - Average response time: 50 ms
  - Scaling: for every 100 NFS ops/sec, increase capacity 1GB
  - Results: plot of server load (throughput) vs. response time & number of users
    - » Assumes: 1 user => 10 NFS ops/sec

FTC.W99 79

Example SPEC SFS Result: DEC Alpha

- 200 MHz 21064: 8KI + 8KD + 2MB L2; 512 MB; 1 Gigaswitch
- DEC OSF1 v2.0
- 4 FDDI networks; 32 NFS Daemons, 24 GB file size
- 88 Disks, 16 controllers, 84 file systems



FTC.W99 80

Willy

- UNIX File System Benchmark that gives insight into I/O system behavior (Chen and Patterson, 1993)
- Self scaling to automatically explore system size
- Examines five parameters
  - Unique bytes touched: - data size; locality via LRU
    - » Gives file cache size
  - Percentage of reads: %writes = 1 – % reads; typically 50%
    - » 100% reads gives peak throughput
  - Average I/O Request Size: Bernoulli, C=1
  - Percentage sequential requests: typically 50%
  - Number of processes: concurrency of workload (number processes issuing I/O requests)
- Fix four parameters while vary one parameter
- Searches space to find high throughput

FTC.W99 81

Example Willy: DS 5000

	Sprite	Ultrix
Avg. Access Size	32 KB	13 KB
Data touched (file cache)	2MB, 15 MB	2 MB
Data touched (disk)	36 MB	6 MB

- % reads = 50%, % sequential = 50%
- DS 5000 32 MB memory
- Ultrix: Fixed File Cache Size, Write through
- Sprite: Dynamic File Cache Size, Write back (Write cancelling)

FTC.W99 82

Sprite's Log Structured File System

Large file caches effective in reducing disk reads

Disk traffic likely to be dominated by writes

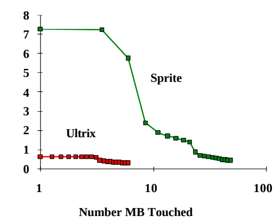
Write-Optimized File System
- Only representation on disk is log
- Stream out files, directories, maps without seeks

Advantages:
- Speed
- Stripes easily across several disks
- Fast recovery
- Temporal locality
- Versioning

Problems:
- Random access retrieval
- Log wrap
- Disk space utilization

FTC.W99 83

Willy: DS 5000 Number Bytes Touched



- Log Structured File System: effective write cache of LFS much smaller (5-8 MB) than read cache (20 MB)
  - Reads cached while writes are not => 3 plateaus

FTC.W99 84

Summary: I/O Benchmarks

- Scaling to track technological change
- TPC: price performance as nomalizing configuration feature
- Auditing to ensure no foul play
- Throughput with restricted response time is normal measure

Outline

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

Interconnect Trends

- Interconnect = glue that interfaces computer system components
- High speed hardware interfaces + logical protocols
- Networks, channels, backplanes

	Network	Channel	Backplane
Distance	>1000 m	10 - 100 m	1 m
Bandwidth	10 - 100 Mb/s	40 - 1000 Mb/s	320 - 1000+ Mb/s
Latency	high (>ms)	medium	low (<µs)
Reliability	low Extensive CRC	medium Byte Parity	high Byte Parity

message-based / narrow pathways / distributed arb ←→ memory-mapped / wide pathways / centralized arb

Backplane Architectures

Metric	VME	FutureBus	MultiBus II	SCSI-I
Bus Width (signals)	128	96	96	25
Address/Data Multiplexed?	No	Yes	Yes	na
Data Width	16 - 32	32	32	8
Xfer Size	Single/Multiple	Single/Multiple	Single/Multiple	Single/Multiple
# of Bus Masters	Multiple	Multiple	Multiple	Multiple
Split Transactions	No	Optional	Optional	Optional
Clocking	Async	Async	Sync	Either
Bandwidth, Single Word (0 ns mem)	25	37	20	5, 1.5
Bandwidth, Single Word (150 ns mem)	12.9	15.5	10	5, 1.5
Bandwidth Multiple Word (0 ns mem)	27.9	95.2	40	5, 1.5
Bandwidth Multiple Word (150 ns mem)	13.6	20.8	13.3	5, 1.5
Max # of devices	21	20	21	7
Max Bus Length	.5 m	.5 m	.5 m	25 m
Standard	IEEE 1014	IEEE 896	ANSI/IEEE 1296	ANSI X3.131

Distinctions begin to blur:

SCSI channel is like a bus

FutureBus is like a channel (disconnect/reconnect)

HIPPI forms links in high speed switching fabrics

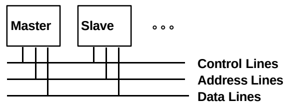
Bus-Based Interconnect

- Bus: a shared communication link between subsystems
  - Low cost: a single set of wires is shared multiple ways
  - Versatility: Easy to add new devices & peripherals may even be ported between computers using common bus
- Disadvantage
  - A communication bottleneck, possibly limiting the maximum I/O throughput
- Bus speed is limited by physical factors
  - the bus length
  - the number of devices (and, hence, bus loading).
  - these physical limits prevent arbitrary bus speedup.

Bus-Based Interconnect

- Two generic types of busses:
  - I/O busses: lengthy, many types of devices connected, wide range in the data bandwidth), and follow a bus standard (sometimes called a *channel*)
  - CPU–memory buses: high speed, matched to the memory system to maximize memory–CPU bandwidth, single device (sometimes called a *backplane*)
  - To lower costs, low cost (older) systems combine together
- Bus transaction
  - Sending address & receiving or sending data

Bus Protocols



Multibus: 20 address, 16 data, 5 control, 50ns Pause

Bus Master: has ability to control the bus, initiates transaction

Bus Slave: module activated by the transaction

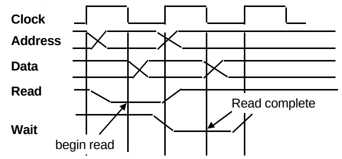
Bus Communication Protocol: specification of sequence of events and timing requirements in transferring information.

Asynchronous Bus Transfers: control lines (req., ack.) serve to orchestrate sequencing

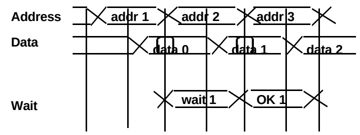
Synchronous Bus Transfers: sequence relative to common clock

FTC.W99 91

Synchronous Bus Protocols



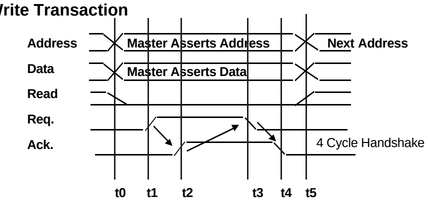
Pipelined/Split transaction Bus Protocol



FTC.W99 92

Asynchronous Handshake

Write Transaction



t0 : Master has obtained control and asserts address, direction, data
Waits a specified amount of time for slaves to decode target\

t1: Master asserts request line

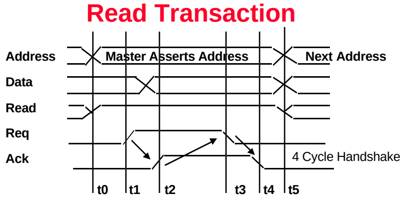
t2: Slave asserts ack, indicating data received

t3: Master releases req

t4: Slave releases ack

FTC.W99 93

Read Transaction



t0 : Master has obtained control and asserts address, direction, data
Waits a specified amount of time for slaves to decode target\

t1: Master asserts request line

t2: Slave asserts ack, indicating ready to transmit data

t3: Master releases req, data received

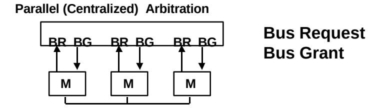
t4: Slave releases ack

Time Multiplexed Bus: address and data share lines

FTC.W99 94

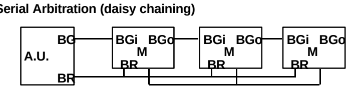
Bus Arbitration

Parallel (Centralized) Arbitration

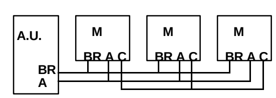


Bus Request
Bus Grant

Serial Arbitration (daisy chaining)



Polling



FTC.W99 95

Bus Options

Option	*High performance*	*Low cost*
Bus width	Separate address & data lines	Multiplex address & data lines
Data width	Wider is faster (e.g., 32 bits)	Narrower is cheaper (e.g., 8 bits)
Transfer size	Multiple words has less bus overhead	Single-word transfer is simpler
Bus masters	Multiple (requires arbitration)	Single master (no arbitration)
Split transaction?	Yes—separate Request and Reply packets gets higher bandwidth (needs multiple masters)	No—continuous connection is cheaper and has lower latency
Clocking	Synchronous	Asynchronous

FTC.W99 96

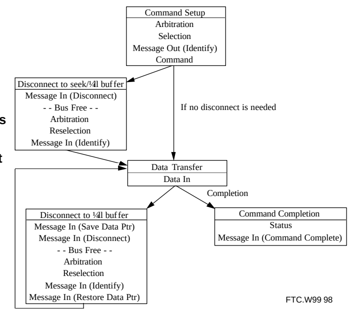
SCSI: Small Computer System Interface

- Clock rate: 5 MHz / 10 MHz (fast) / 20 MHz (ultra)
- Width: n = 8 bits / 16 bits (wide); up to n – 1 devices to communicate on a bus or "string"
- Devices can be slave ("target") or master("initiator")
- SCSI protocol: a series of "phases", during which specific actions are taken by the controller and the SCSI disks
  - Bus Free: No device is currently accessing the bus
  - Arbitration: When the SCSI bus goes free, multiple devices may request (arbitrate for) the bus; fixed priority by address
  - Selection: informs the target that it will participate (Reselection if disconnected)
  - Command: the initiator reads the SCSI command bytes from host memory and sends them to the target
  - Data Transfer: data in or out, initiator: target
  - Message Phase: message in or out, initiator: target (identify, save/restore data pointer, disconnect, command complete)
  - Status Phase: target, just before command complete

FTC.W99 97

SCSI "Bus": Channel Architecture

peer-to-peer protocols
initiator/target
linear byte streams
disconnect/reconnect



FTC.W99 98

1993 I/O Bus Survey (P&H, 2nd Ed)

Bus	SBus	TurboChannel	MicroChannel	PCI
Originator	Sun	DEC	IBM	Intel
Clock Rate (MHz)	16-25	12.5-25	async	33
Addressing	Virtual	Physical	Physical	Physical
Data Sizes (bits)	8,16,32	8,16,24,32	8,16,24,32,64	8,16,24,32,64
Master	Multi	Single	Multi	Multi
Arbitration	Central	Central	Central	Central
32 bit read (MB/s)	33	25	20	33
Peak (MB/s)	89	84	75	111 (222)
Max Power (W)	16	26	13	25

FTC.W99 99

1993 MP Server Memory Bus Survey

Bus	Summit	Challenge	XDBus
Originator	HP	SGI	Sun
Clock Rate (MHz)	60	48	66
Split transaction?	Yes	Yes	Yes?
Address lines	48	40	??
Data lines	128	256	144 (parity)
Data Sizes (bits)	512	1024	512
Clocks/transfer	4	5	4?
Peak (MB/s)	960	1200	1056
Master	Multi	Multi	Multi
Arbitration	Central	Central	Central
Addressing	Physical	Physical	Physical
Slots	16	9	10
Busses/system	1	1	2
Length	13 inches	12? inches	17 inches

FTC.W99 100

Summary: I/O Benchmarks

- Scaling to track technological change
- TPC: price performance as normalizing configuration feature
- Auditing to ensure no foul play
- Throughput with restricted response time is normal measure

FTC.W99 101