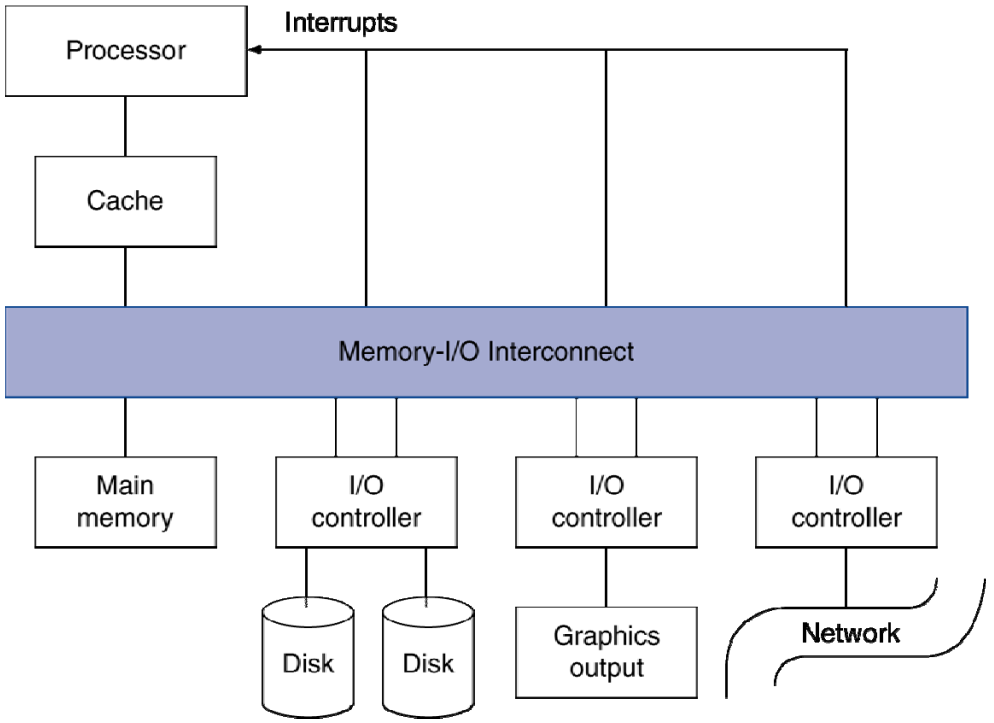
Computer Organization and Design

Storage and Other I/O Topics

Introduction

- I/O devices can be characterized by
  - Behaviour: input, output, storage
  - Partner: human or machine
  - Data rate: peak rate in bytes/sec, transfers/sec



I/O System Characteristics

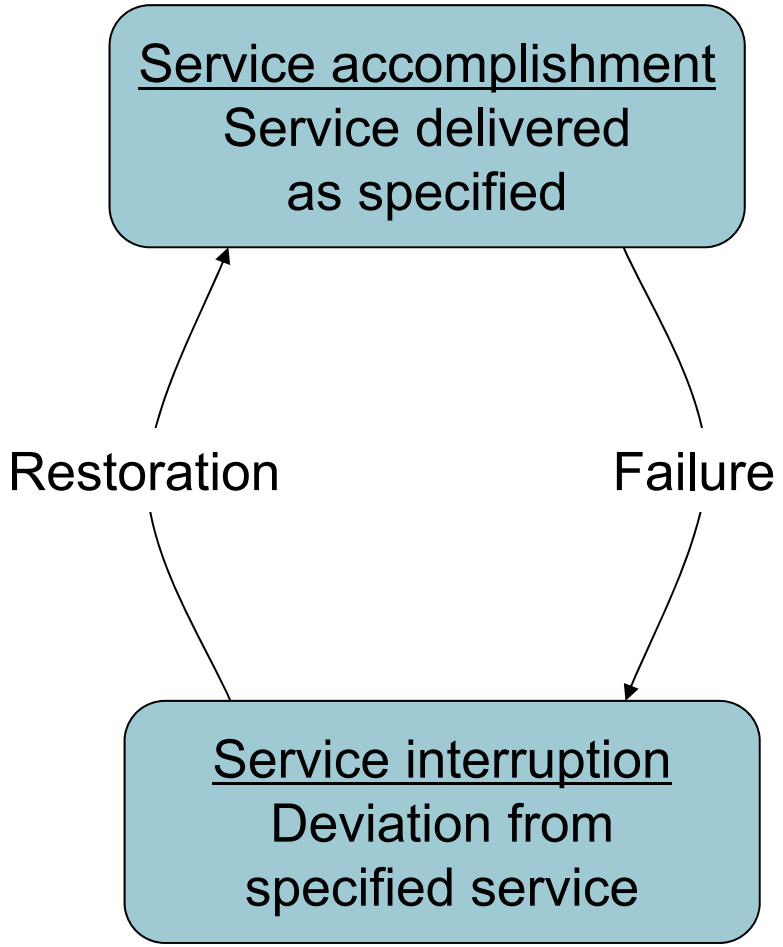
- Dependability and Expandability is important
  - Particularly for storage devices
- I/O systems – dependability and cost critical
  - Processors and memory – performance, power, cost
- Performance measures
  - Latency (response time)
  - Throughput (bandwidth)
- Desktops & Embedded systems
  - Mainly interested in response time & diversity of devices
- Servers
  - Mainly interested in throughput & expandability of devices

Diversity of I/O Devices

Device	Behavior	Partner	Data rate (Mbit/sec)
Keyboard	Input	Human	0.0001
Mouse	Input	Human	0.0038
Voice input	Input	Human	0.2640
Sound input	Input	Machine	3.0000
Scanner	Input	Human	3.2000
Voice output	Output	Human	0.2640
Sound output	Output	Human	8.0000
Laser printer	Output	Human	3.2000
Graphics display	Output	Human	800.0000–8000.0000
Cable modem	Input or output	Machine	0.1280–6.0000
Network/LAN	Input or output	Machine	100.0000–10000.0000
Network/wireless LAN	Input or output	Machine	11.0000–54.0000
Optical disk	Storage	Machine	80.0000–220.0000
Magnetic tape	Storage	Machine	5.0000–120.0000
Flash memory	Storage	Machine	32.0000–200.0000
Magnetic disk	Storage	Machine	800.0000–3000.0000

I/O Transfer rates quoted in base 10

Dependability



- Fault: failure of a component
  - May or may not lead to system failure
- Failure: permanent or intermittent
  - Harder to diagnose latter when system oscillates b/w 2 states

Summary of Reasons for Failure

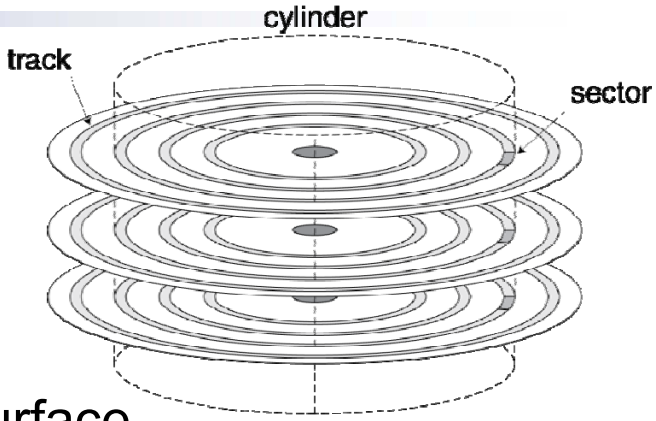
Operator	Software	Hardware	System	Year data collected
42%	25%	18%	Datacenter (Tandem)	1985
15%	55%	14%	Datacenter (Tandem)	1989
18%	44%	39%	Datacenter (DEC VAX)	1985
50%	20%	30%	Datacenter (DEC VAX)	1993
50%	14%	19%	U.S. public telephone network	1996
54%	7%	30%	U.S. public telephone network	2000
60%	25%	15%	Internet services	2002

- Hard to determine causes of failure
  - e.g. operators cause failures, but are also responsible for recording them
- Other categories for outages exist
  - e.g. environmental; generally not major causes

Dependability Measures

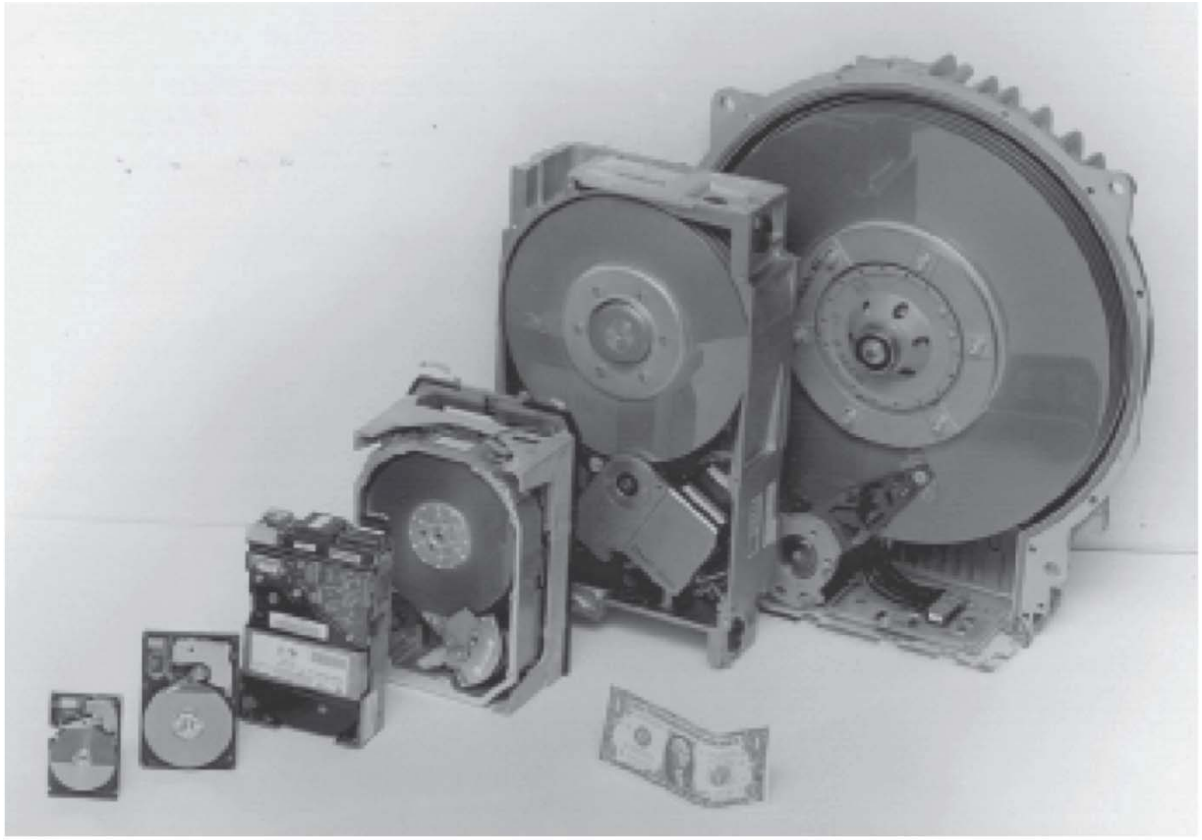
- Reliability: mean time to failure (MTTF)
  - Annual failure rate (AFR): % of devices expected to fail in a year for a given MTTF
- Service interruption: mean time to repair (MTTR)
- Mean time between failures
  - MTBF = MTTF + MTTR
- Availability = MTTF / (MTTF + MTTR)
- How can we Improve Availability?
  - Increase MTTF: fault avoidance, fault tolerance, fault forecasting
  - Reduce MTTR: improved tools and processes for diagnosis and repair

Magnetic Disk Storage



- Purpose
  - Long term, nonvolatile storage
  - Lowest level in the memory hierarchy
- General structure
  - rotating platter coated with a magnetic surface
  - moveable read/write head to access the information on the disk
- Typical numbers
  - 1 to 4 platters (each with 2 recordable surfaces) per disk
  - platters 1” to 3.5” in diameter
  - Rotational speeds of 5,400 to 15,000 RPM
  - 10,000 to 50,000 tracks per surface
    - cylinder - all the tracks under the head at a given point on all surfaces
  - 100 to 500 sectors per track
    - the smallest unit that can be read/written (typically 512B)

Magnetic Disk Evolution



- Diameters: 14 in down to 1.8 in
- Cost (~2010): $0.10 - $0.50 per GB

Disk Sectors and Access

- Each sector records
  - Sector ID
  - Data (512 bytes, 4096 bytes proposed)
  - Error correcting code (ECC)
    - Used to hide defects and recording errors
  - Sector number of next sector
  - Synchronization fields and gaps

- Access to a sector involves
  - Seek time: time to move the head over proper track
  - Rotational latency: time for desired sector to rotate under the head
  - Data transfer time: *f*(sector size, rotation speed, recording density)
    - 70 – 125 MB/s, up to 375 MB/s with built-in cache (~2008)
  - Disk controller overhead: handling control of the disk
  - Queuing delay: if other accesses are pending

Disk Access Example

- What is the average sector read time, given:
  - 512B sector, 15,000rpm, 4ms average seek time, 100 MB/s transfer rate, 0.2ms controller overhead, idle disk
- Average read time
  - 4ms seek time
    + 0.5 rotations / (15,000/60) = 2ms rotational latency
    + 512 / 100MB/s = 0.005ms transfer time
    + 0.2ms controller delay
    = 6.2ms
- If actual average seek time is 1ms
  - Average read time = 3.2ms
  - Manufacturers quote average seek time
    - Based on all possible seeks
    - Locality, OS scheduling lead to smaller actual avg. seek times

Disk Performance Issues

- Smart disk controller allocates physical sectors on disk
  - Organize disks more like tapes than random access devices
    - Speed up sequential transfers
  - ATA (Advanced Technology Attachment) – A command set and interface standard for the connection of storage devices such as hard disks, solid-state drives, and CD-ROM drives. Parallel ATA has been largely replaced by serial ATA (SATA).
  - SCSI (Small Computer Systems Interface) – A set of standards (commands, protocols, and electrical and optical interfaces) for physically connecting and transferring data between computers and peripheral devices. Most commonly used for hard disks and tape drives.
- Disk drives include caches
  - Prefetch sectors in anticipation of access
  - Avoid seek and rotational delay

Magnetic Disks: Real World Data

Feature	Seagate ST31000340NS	Seagate ST973451SS	Seagate ST9160821AS
Disk diameter (inches)	3.5	2.5	2.5
Capacity (GB)	1000	73	160
# of surfaces (heads)	4	2	2
Rotation speed (RPM)	7,200	15,000	5,400
Transfer rate (MB/sec)	105	79-112	44
Minimum seek (ms)	0.8r-1.0w	0.2r-0.4w	1.5r-2.0w
Average seek (ms)	8.5r-9.5w	2.9r-3.3w	12.5r-13.0w
MTTF (hours@25°C)	1,200,000	1,600,000	??
Dim (inches), Weight (lbs)	1x4x5.8, 1.4	0.6x2.8x3.9, 0.5	0.4x2.8x3.9, 0.2
GB/cu.inch, GB/watt	43, 91	11, 9	37, 84
Power: op/idle/sb (watts)	11/8/1	8/5.8/-	1.9/0.6/0.2
Price in 2008, $/GB	~$0.3/GB	~$5/GB	~$0.6/GB

Aside: Media Bandwidth Requirements

- High quality video
  - Digital data = (30 frames/s) × (640 x 480 pixels) × (24-b color/pixel) = 221 Mb/s (27.625 MB/s)
    - Some drives may not be able to stream at this rate!
- High quality audio
  - Digital data = (44,100 audio samples/s) × (16-b audio samples) × (2 audio channels for stereo) = 1.4 Mb/s (0.175 MB/s)
- Compression *reduces* the bandwidth requirements considerably
  - H.264, MPEG2/4, DIVX, MP3, AAC, …

Flash Storage

- Nonvolatile semiconductor storage
  - 100× to 1000× faster than magnetic disk
  - Smaller, lower power, more robust
  - Large market (cell phones, cameras, MP3 players) to motivate investment and improvements in flash technology
  - But more $/GB (2 to 40 times higher than disk; dropping fast)

Characteristics	Kingston SecureDigital (SD) SD4/8 GB	Transend Type I CompactFlash TS16GCF133	RiDATA Solid State Disk 2.5 inch SATA
Formatted data capacity (GB)	8	16	32
Bytes per sector	512	512	512
Data transfer rate (read/write MB/sec)	4	20/18	68/50
Power operating/standby (W)	0.66/0.15	0.66/0.15	2.1/—
Size: height × width × depth (inches)	0.94 × 1.26 × 0.08	1.43 × 1.68 × 0.13	0.35 × 2.75 × 4.00
Weight in grams (454 grams/pound)	2.5	11.4	52
Mean time between failures (hours)	> 1,000,000	> 1,000,000	> 4,000,000
GB/cu. in., GB/watt	84 GB/cu.in., 12 GB/W	51 GB/cu.in., 24 GB/W	8 GB/cu.in., 16 GB/W
Best price (2008)	~ $30	~ $70	~ $300

Flash Types

- NOR flash: bit cell like a NOR gate
  - Random read/write access
- NAND flash: bit cell like a NAND gate
  - Denser (bits/area), but block-at-a-time access
    - Wiring for random access not present
  - Cheaper per GB

Characteristics	NOR Flash Memory	NAND Flash Memory
Typical use	BIOS memory	USB key
Minimum access size (bytes)	512 bytes	2048 bytes
Read time (microseconds)	0.08	25
Write time (microseconds)	10.00	1500 to erase + 250
Read bandwidth (MBytes/second)	10	40
Write bandwidth (MBytes/second)	0.4	8
Wearout (writes per cell)	100,000	10,000 to 100,000
Best price/GB (2008)	$65	$4

Flash Wearout

- Flash bits wears out after 1000's of accesses
  - Not suitable for direct RAM or disk replacement
- Wear leveling:
  - controller included to remap data to less used blocks
  - used in all consumer products (MP3 players, ...)
  - also improve yield by avoiding damaged memory cells
- Today's laptops
  - Hybrid hard disks
    - ~1 GB of flash to improve boot times, save energy
  - Fully flash instead of hard disks
- Tomorrow: desktops and servers?

Interconnecting Components

- Need interconnections between
  - CPU, memory, I/O controllers
- Bus: shared communication channel
  - Parallel set of wires for data and synchronization of data transfer
  - Can become a bottleneck
- Performance limited by physical factors
  - Wire length, number of connections
- More recent alternative: high-speed serial connections with switches
  - Like networks

Types of Buses

- Processor-memory bus ("Front Side Bus", proprietary)
  - Short and high speed
  - Matched to the memory system to maximize the memory-processor bandwidth
  - Optimized for cache block transfers
- I/O bus (industry standard, e.g., SCSI, USB, Firewire)
  - Usually is lengthy and slower
  - Needs to accommodate a wide range of I/O devices
  - Use either the processor-memory bus or a backplane bus to connect to memory
- Backplane bus (industry standard, e.g., ATA, PCIExpress)
  - Allow processor, memory, I/O devices to coexist on a single bus
  - Used as an intermediary bus connecting I/O buses to the processor-memory bus

Bus Signals and Synchronization

- Data lines
  - Carry address and data
  - Multiplexed or separate
- Control lines
  - Indicate data type, synchronize transactions
- Synchronous
  - Uses a bus clock
- Asynchronous
  - Uses request/acknowledge control lines for handshaking

Synchronous and Asynchronous Buses

- Synchronous bus (e.g., processor-memory buses)
  - Includes a clock in the control lines and has a fixed protocol for communication that is relative to the clock
  - Advantage: involves very little logic and can run very fast
  - Disadvantages:
    - Every device communicating on the bus must use same clock rate
    - To avoid clock skew, they cannot be long if they are fast
- Asynchronous bus (e.g., I/O buses)
  - It is not clocked, so requires a handshaking protocol and additional control lines (ReadReq, Ack, DataRdy)
  - Advantages:
    - Can accommodate a wide range of devices and device speeds
    - Can be lengthened without worrying about clock skew or synchronization problems
  - Disadvantage: slow(er)

ATA Cable Sizes

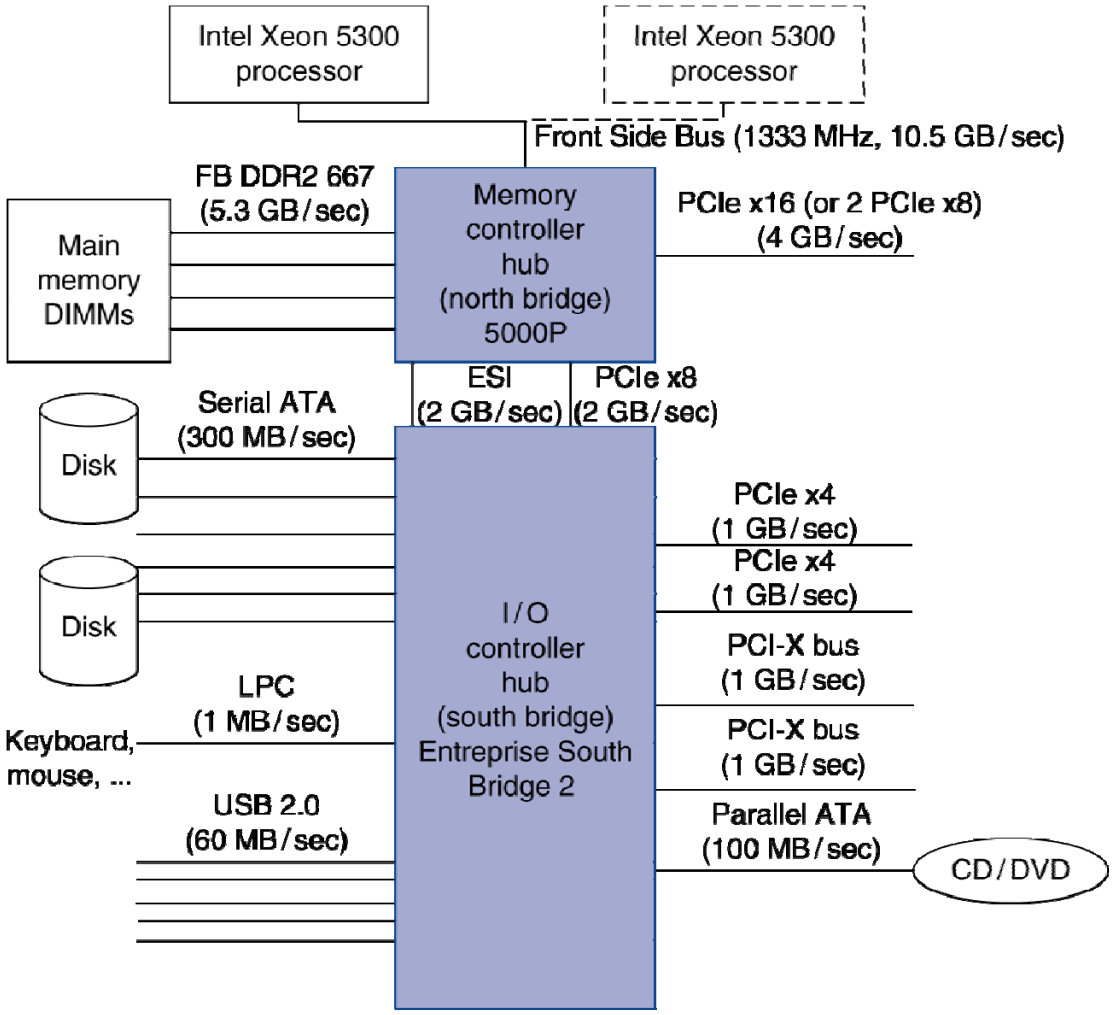
- Companies have transitioned from synchronous, parallel wide buses to asynchronous narrow buses
  - Reflection on wires and clock skew makes it difficult to use 16 to 64 parallel wires running at a high clock rate (e.g., ~400MHz) so companies have moved to buses with a few one-way wires running at a very high “clock” rate (~2GHz)

  - Serial ATA cables (red) are much thinner than parallel ATA cables (green)

Dominant I/O Standards

	Firewire	USB 2.0	PCI Express	Serial ATA	Serial Attached SCSI
Intended use	External	External	Internal	Internal	External
Devices per channel	63	127	1	1	4
Data width	4	2	2/lane	4	4
Peak bandwidth	50MB/s or 100MB/s	0.2MB/s, 1.5MB/s, or 60MB/s	250MB/s/lane 1×, 2×, 4×, 8×, 16×, 32×	300MB/s	300MB/s
Hot pluggable	Yes	Yes	Depends	Yes	Yes
Max length	4.5m	5m	0.5m	1m	8m
Standard	IEEE 1394	USB Implementers Forum	PCI-SIG	SATA-IO	INCITS TC T10

Typical x86 PC I/O System



I/O Chip Sets: Intel, AMD

	Intel 5000P chip set	Intel 975X chip set	AMD 580X CrossFiret
Target segment	Server	Performance PC	Server/Performance PC
Front Side Bus (64 bit)	1066/1333 MHz	800/1066 MHz	—
Memory controller hub ("north bridge")			
Product name	Blackbird 5000P MCH	975X MCH	
Pins	1432	1202	
Memory type, speed	DDR2 FBDIMM 667/533	DDR2 800/667/533	
Memory buses, widths	4 × 72	1 × 72	
Number of DIMMs, DRAM/DIMM	16, 1 GB/2 GB/4 GB	4, 1 GB/2 GB	
Maximum memory capacity	64 GB	8 GB	
Memory error correction available?	Yes	No	
PCIe/External Graphics Interface	1 PCIe x16 or 2 PCIe x	1 PCIe x16 or 2 PCIe x8	
South bridge interface	PCIe x8, ESI	PCIe x8	
I/O controller hub ("south bridge")			
Product name	6321 ESB	ICH7	580X CrossFire
Package size, pins	1284	652	549
PCI-bus: width, speed	Two 64-bit, 133 MHz	32-bit, 33 MHz, 6 masters	—
PCI Express ports	Three PCIe x4		Two PCIe x16, Four PCI x1
Ethernet MAC controller, interface	—	1000/100/10 Mbit	—
USB 2.0 ports, controllers	6	8	10
ATA ports, speed	One 100	Two 100	One 133
Serial ATA ports	6	2	4
AC-97 audio controller, interface	—	Yes	Yes
I/O management	SMbus 2.0, GPIO	SMbus 2.0, GPIO	ASF 2.0, GPIO

Similar to Intel Nehalem

I/O Management

- I/O is mediated by the OS
  - Multiple programs share I/O resources
    - Need protection and scheduling
  - I/O causes asynchronous interrupts
    - Same mechanism as exceptions – transfer to kernel or supervisor mode
  - I/O programming is complex
    - OS provides abstractions to programs

I/O Commands

- I/O devices are managed by I/O controller hardware
  - Transfers data to/from device
  - Synchronizes operations with software
- Command registers
  - Cause device to do something
- Status registers
  - Indicate what the device is doing and occurrence of errors
- Data registers
  - Write: transfer data to a device
  - Read: transfer data from a device

I/O Register Mapping

- Memory mapped I/O
  - Portions of address space assigned to I/O devices; reads/writes to those addresses interpreted as commands to I/O device
  - OS uses address translation mechanism to make them only accessible to kernel
- I/O instructions
  - Separate instructions to access I/O registers
  - Can only be executed in kernel mode
  - Example: Intel x86, IBM 370

Polling

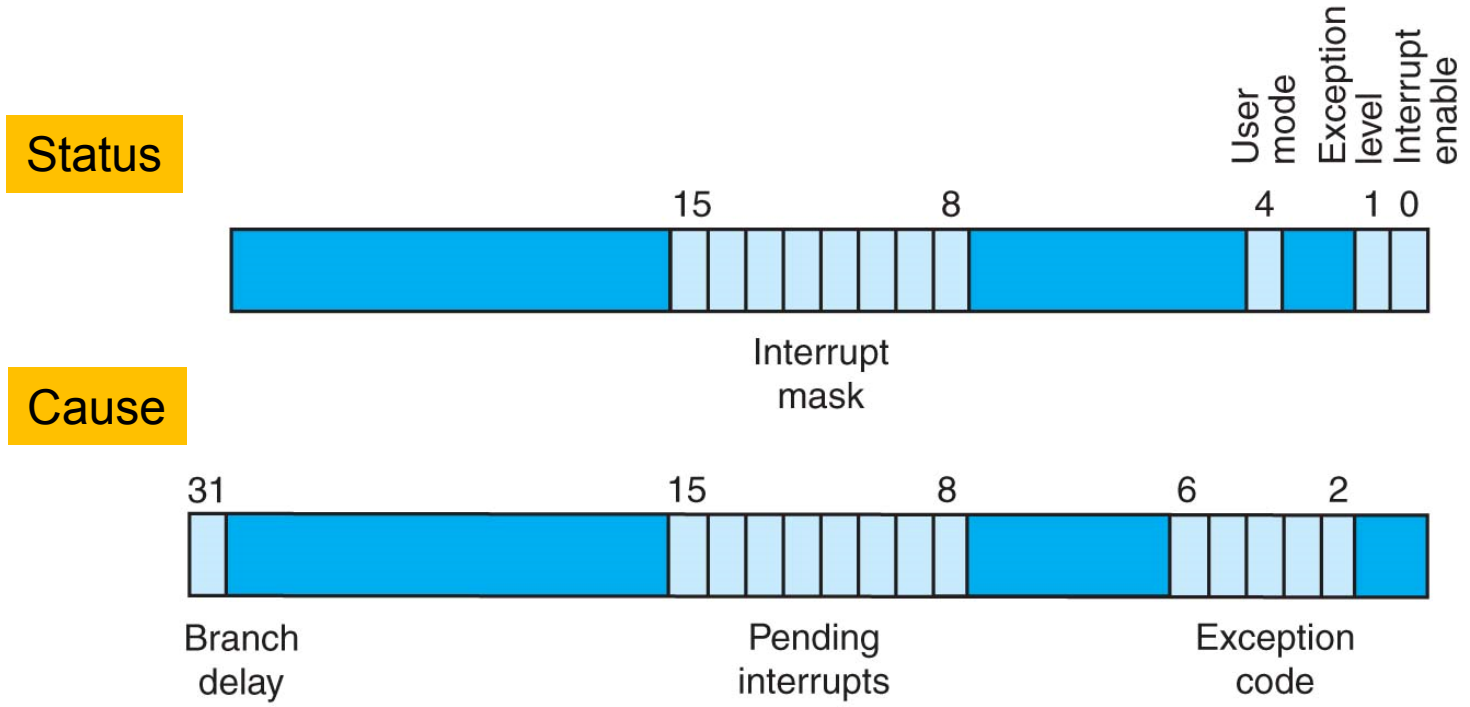
- Periodically check I/O status register
  - If device ready, do operation
  - If error, take action
- Common in small or low-performance real-time embedded systems
  - Predictable timing
  - Low hardware cost
- In other systems, wastes CPU time
  - CPU speed >> I/O device speed

Interrupts

- When a device is ready or error occurs
  - Controller interrupts CPU
- Interrupt is like an exception
  - But not synchronized to instruction execution
  - Can invoke handler between instructions
  - Cause register information often identifies the interrupting device
- Priority interrupts
  - Devices needing more urgent attention get higher priority
  - Can interrupt handler for a lower priority interrupt

Cause and Status Registers

- MIPS-32 architecture
  - To enable a Pending interrupt, the correspond bit in the Interrupt mask must be 1
  - Once an interrupt occurs, the OS can find the reason in the Exception codes field



I/O Data Transfer

- Polling and interrupt-driven I/O
  - CPU transfers data between memory and I/O data registers
  - Time consuming for high-speed devices
- Direct memory access (DMA)
  - Offload I/O management from CPU
  - OS provides starting address in memory and transfer size
  - DMA controller transfers to/from memory autonomously
  - Controller interrupts on completion or error
- I/O Processors
  - GP CPUs similar to DMAs for offloading I/O transfer management from CPU

DMA/Cache Interaction

- With DMA, path to memory changes
  - Bypass of address translation, cache hierarchy
  - Complicates systems with caches
- If DMA writes to a memory block that is cached
  - Cached copy becomes stale
- If write-back cache has dirty block, and DMA reads memory block
  - Gets stale data
- Need to ensure cache coherence
  - Routing I/O through cache is expensive
    - causes conflict misses as I/O data is rarely used immediately
  - OS can flush blocks from cache if they will be used for I/O
  - *Selectively* invalidate cache entries with snooping controller

DMA/VM Interaction

- OS uses virtual addresses for memory
  - DMA blocks may not be contiguous in physical memory
- Should DMA use virtual addresses?
  - Would require controller to do translation
- If DMA uses physical addresses?
  - May need to break transfers into page-sized chunks
  - Or chain multiple transfers
  - Or allocate contiguous physical pages for DMA

Measuring I/O Performance

- I/O performance depends on
  - Hardware: CPU, memory, controllers, buses
  - Software: operating system, database management system, application
  - Workload: request rates and patterns
- I/O system design can trade-off between response time and throughput
  - e.g. handling a request as early as possible minimizes response time, but greater throughput is achieved if related requests are handled together

Transaction Processing Benchmarks

- Transactions
  - Small data accesses to a DBMS
  - Interested in I/O rate (accesses/sec), not data rate (bytes/sec)
- Measure throughput
  - Subject to response time limits and failure handling
  - Overall cost per transaction should be minimized
- Transaction Processing Council (TPC) benchmarks (www.tpc.org)
  - TPC-C: complex query environment (1992)
  - TPC-W: Web based transactional server
  - TPC-APP: B2B application server and web services
  - TPC-E: on-line transaction processing for brokerage firm
  - TPC-H: decision support — business oriented ad-hoc queries

File System & Web Benchmarks

- SPEC Server File System (SFS)
  - Synthetic workload for measuring NFS (network file system) performance, based on monitoring real systems
  - Results
    - Throughput (operations/sec)
    - Response time (average ms/operation)
- SPEC Web Server benchmark
  - Simulates multiple clients requesting static and dynamic pages from a server, as well as clients posting data to servers
  - Three workloads: Banking, Ecommerce, and Support
- SPECPower
  - Measures power, performance of small servers

I/O vs. CPU Performance

- **Amdahl's Law**
  - Don't neglect I/O performance as parallelism increases compute performance
- Example
  - Benchmark takes 90s CPU time, 10s I/O time
  - Double the number of CPUs/2 years
    - CPU speed, I/O time unchanged
  - How much faster will our benchmark run after 6 years?

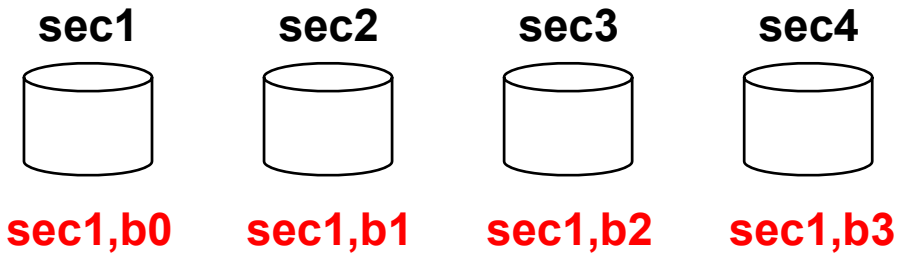
Year	CPU time	I/O time	Elapsed time	% I/O time
now	90s	10s	100s	10%
+2	45s	10s	55s	18%
+4	23s	10s	33s	31%
+6	11s	10s	21s	47%

  - CPU performance improvement = 90/11 = 8
  - Improvement in elapsed time = 100/21 = 4.7

RAID

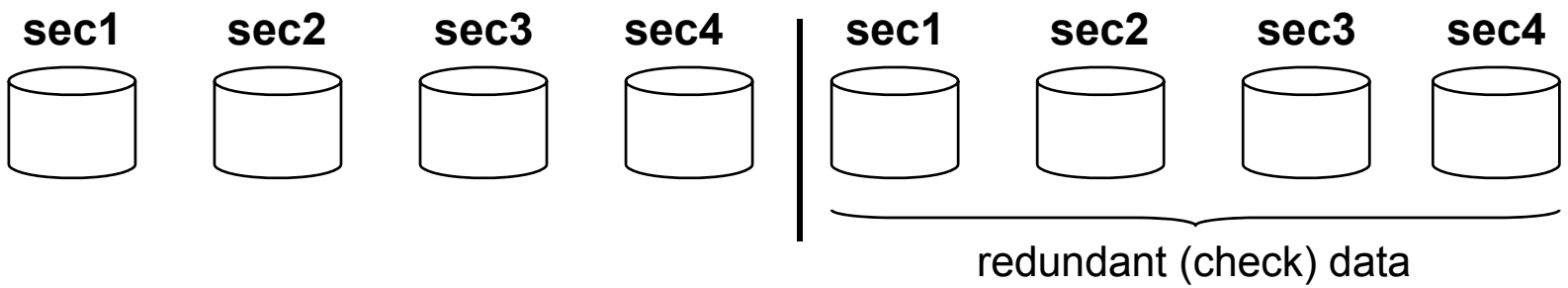
- Redundant Array of Inexpensive (Independent) Disks (~late 1980s)
  - Motivation: improve I/O performance
  - Use multiple smaller disks (c.f. one large disk)
  - Parallelism improves performance
  - Problem: smaller inexpensive drives have lower MTTF
    - e.g. 50 small drives have a 50x greater failure rate
  - Solution: use extra disk(s) for redundant data storage
- Provides fault tolerant storage system
  - Especially if failed disks can be “hot swapped”
  - Key reason for RAID popularity, even though RAID designed for improving performance

RAID 0 (No Redundancy; Striping)



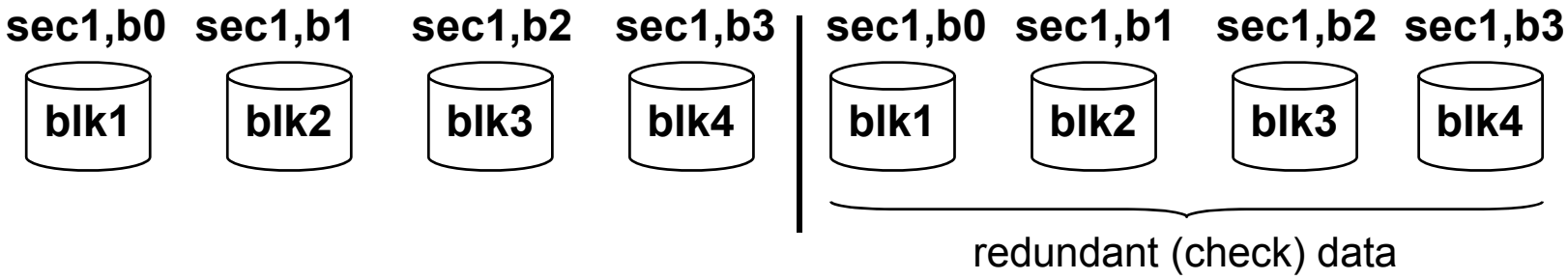
- Multiple smaller disks as opposed to one big disk
  - Spreading the sector over multiple disks – striping – means that multiple blocks can be accessed in parallel increasing performance
    - a 4 disk system gives four times the throughput of a 1 disk system
  - Same cost as one *big* disk
    - assuming 4 small disks cost the same as one big disk
- No redundancy (“AID”), so what if one disk fails?
  - Failure of one or more disks is more likely as the number of disks in the system increases

RAID 1 (Redundancy via Mirroring)



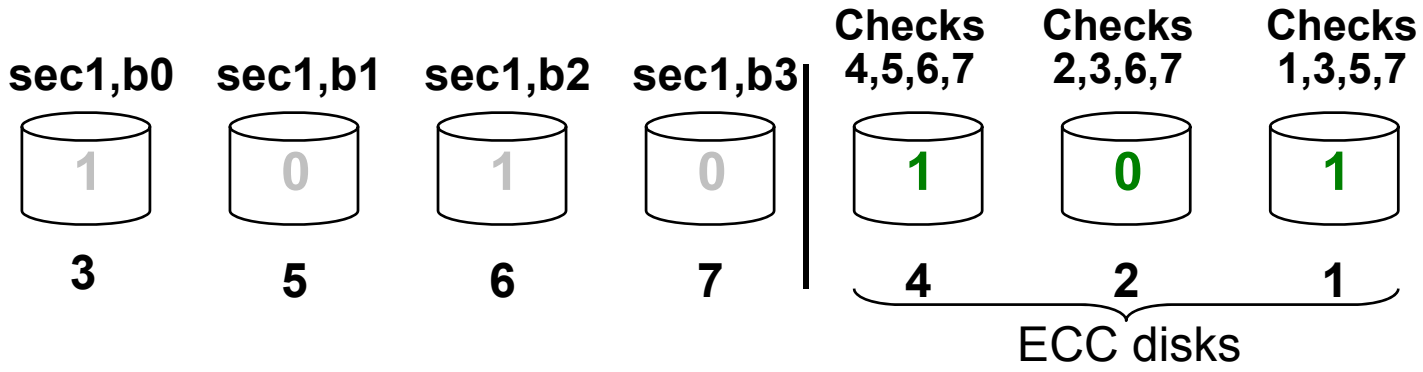
- Uses twice as many disks as RAID 0 (e.g., 8 smaller disks with the second set of 4 duplicating the first set) so there are always two copies of the data
  - # redundant disks = # of data disks so twice the cost of one big disk
    - writes have to be made to both sets of disks, so writes would be only 1/2 the performance of a RAID 0
- What if one disk fails?
  - If a disk fails, the system just goes to the "mirror" for the data

RAID: Level 0+1 (Striping with Mirroring)



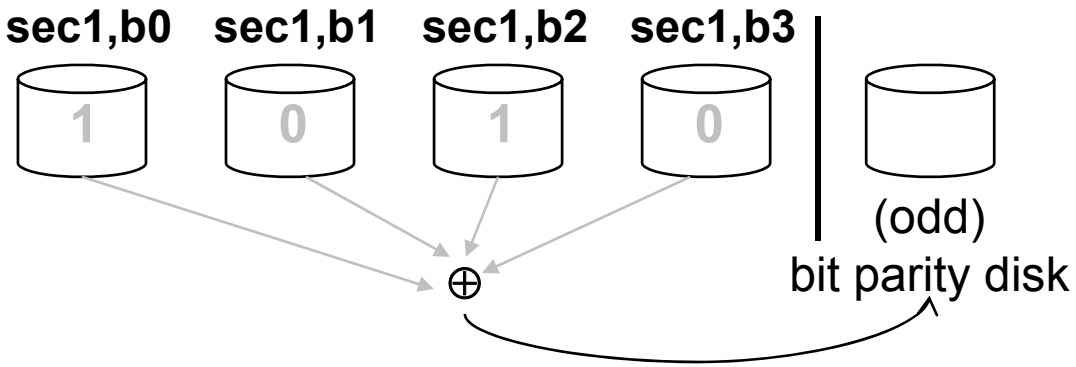
- Combines the best of RAID 0 and RAID 1, data is striped across four disks and mirrored to four disks
  - Four times the throughput (due to striping)
  - # redundant disks = # of data disks, so twice the cost of one big disk
    - writes have to be made to both sets of disks, so writes would be only 1/2 the performance of RAID 0
- What if one disk fails?
  - If a disk fails, the system just goes to the "mirror" for the data

RAID 2 (Redundancy via ECC)



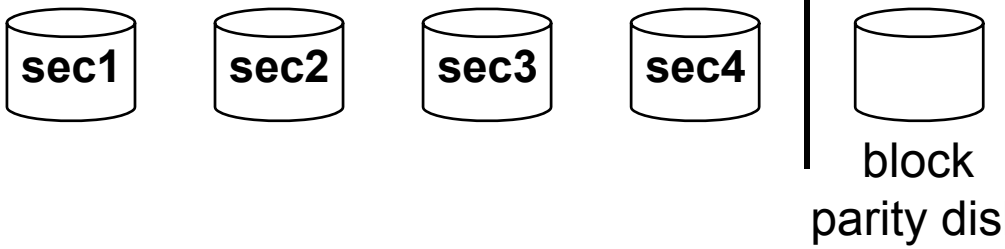
- ECC disks contain the parity of data on a set of distinct overlapping disks
  - # redundant disks = log (total # of data disks) so almost twice the cost of one big disk
    - writes require computing parity to write to the ECC disks
    - reads require reading ECC disk and confirming parity
- Can tolerate *limited* disk failure, since the data can be reconstructed; But too complex, not used in practice

RAID 3 (Bit-Interleaved Parity)



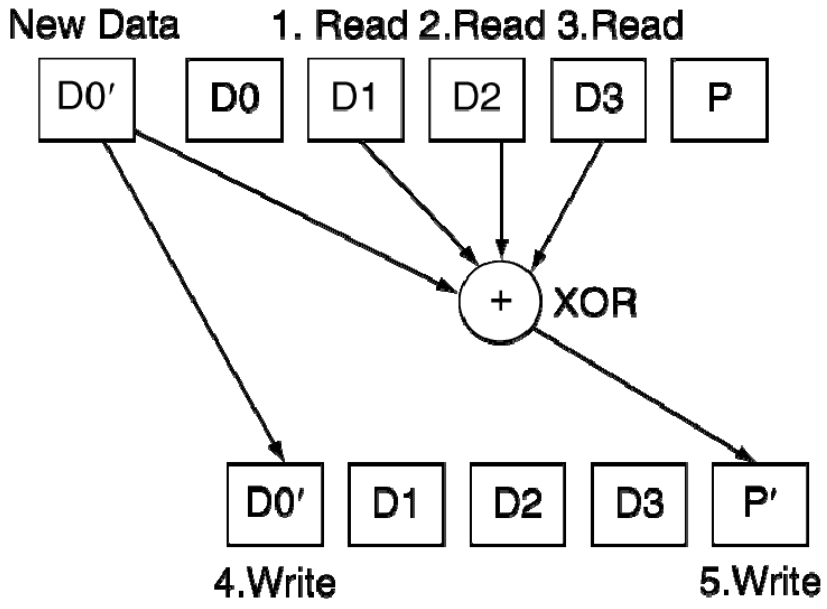
- Cost of higher availability is reduced to 1/N where N is the number of disks in a protection group
  - # redundant disks = 1 × # of protection groups
    - Data striped across N disks at byte level; redundant disk stores parity
    - Read access – Read all disks
    - Write access – Generate new parity and update all disks
    - On failure – Use parity to reconstruct missing data
- Can tolerate *limited* disk failure, since the data can be reconstructed
- RAID 3: takes longer to recover but spends less on redundant storage
- Popular in applications with large data sets (multimedia, scientific code)

RAID 4 (Block-Interleaved Parity)

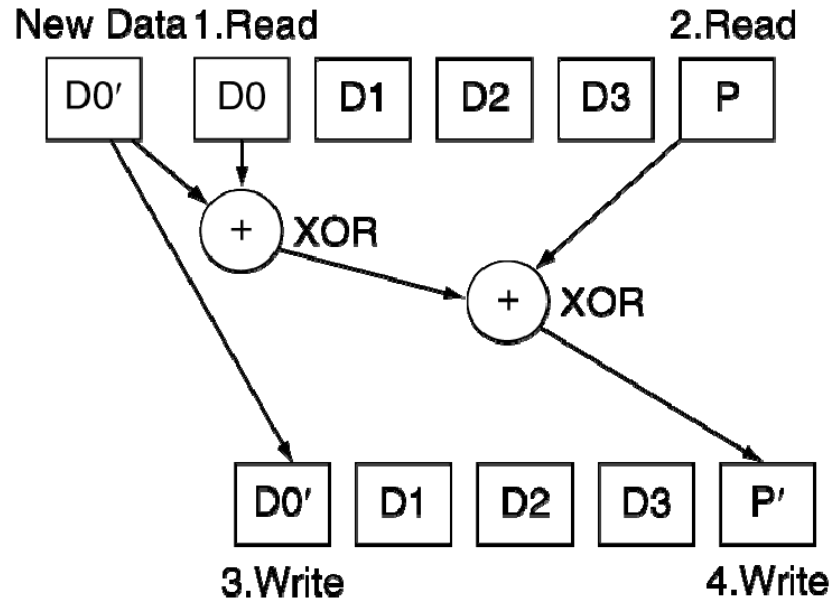


- Cost of higher availability still only 1/N but the parity is stored as blocks associated with sets of data blocks
  - # redundant disks = 1 × # of protection groups
  - Data striped across N disks at block level
  - redundant disk stores parity for group of blocks
  - Supports “small reads” and “small writes” (reads and writes that go to just one (or a few) data disk in a protection group)
  - Read access - read only the disk holding the required block
  - Write access - just read disk containing modified block, and parity disk; Calculate new parity, update data disk and parity disk
  - On failure - use parity to reconstruct missing data
- Can tolerate *limited* disk failure, since the data can be reconstructed

Small write update on RAID 3 vs. RAID 4



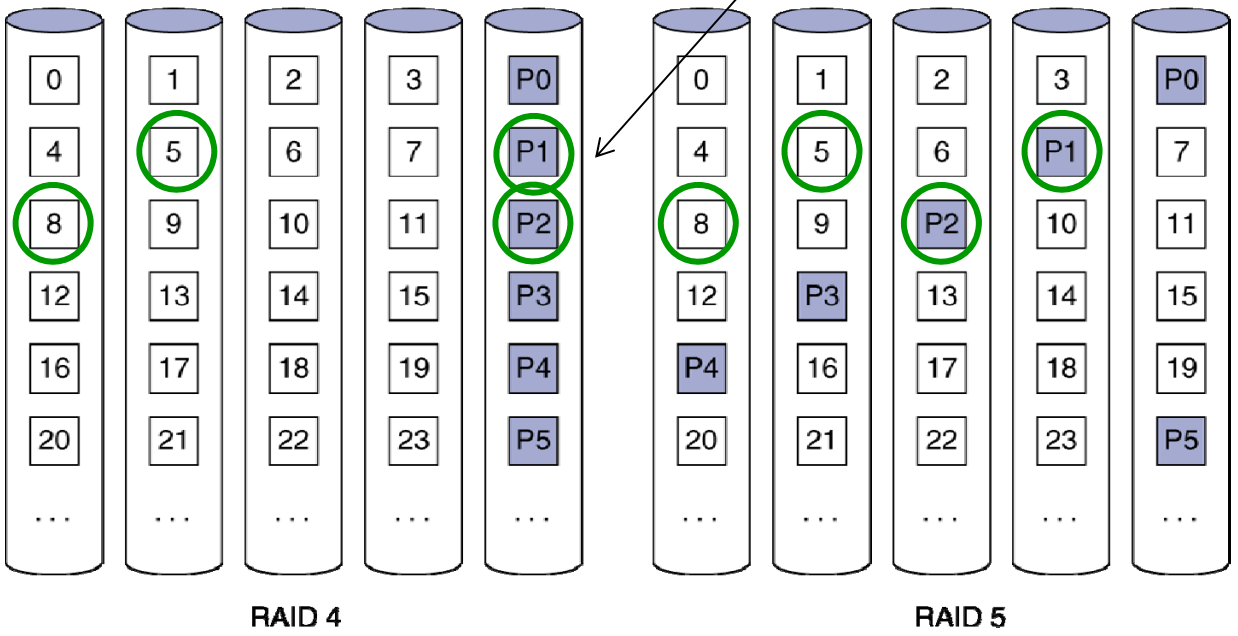
3 reads and
2 writes
involving *all*
the disks



2 reads and
2 writes
involving just
two disks

RAID 5: Distributed Parity

- N + 1 disks
  - Like RAID 4, but parity blocks distributed across disks
    - Avoids parity disk being a bottleneck
- Widely used



RAID 6: P + Q Redundancy

- N + 2 disks
  - Like RAID 5, but two lots of parity
    - Horizontal striped P parity, vertical striped Q parity
    - Second check block allows recovery from second failure
    - Redundant storage overhead twice that of RAID 5
  - Greater fault tolerance through more redundancy
- Multiple RAID
  - More advanced systems give similar fault tolerance with better performance

RAID Summary

- RAID can improve performance and availability
  - High availability requires hot swapping
  - Standby spares to reduce MTTR
- RAID 1 and RAID 5 widely used in servers
  - Growing interest in RAID 6 to protect against multiple failures that are becoming more common
- Assumes independent disk failures
  - Too bad if the building burns down!
- See “Hard Disk Performance, Quality and Reliability”
  - http://www.pcguide.com/ref/hdd/perf/index.htm

I/O System Design

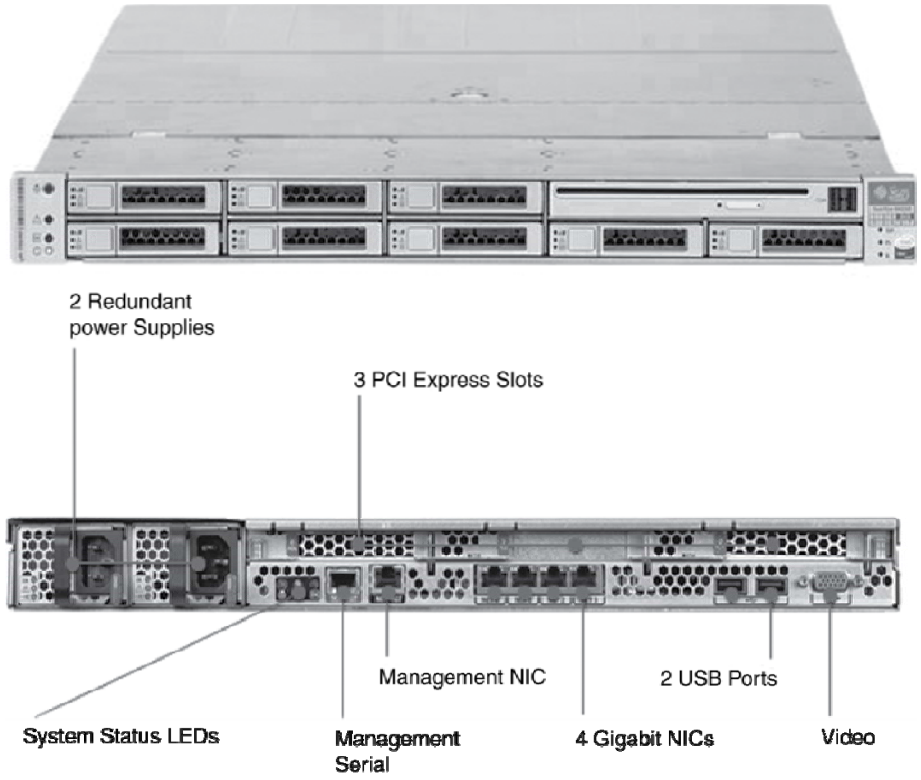
- Satisfying latency requirements
  - For time-critical operations
  - If system is unloaded
    - Add up latency of components
- If system is loaded, simple analysis is insufficient
  - Need to use queuing models or simulation
- Satisfying throughput requirements
  - Find “weakest link” (lowest-bandwidth component)
  - Configure to operate at its maximum bandwidth
  - Balance remaining components in the system

Server Computers

- Applications are increasingly run on servers (software as a service model)
  - Web search, office apps, virtual worlds, …
- Requires large data center servers
  - Multiple processors, networks connections, massive storage, no displays or keyboards
  - Space and power constraints
- Server equipment built for 19” wide racks – standardized for data centers
  - Racks are multiples of 1.75” (1 rack unit or U) high
  - Computers designed for rack called rack mounts or shelf or subrack; 1U servers

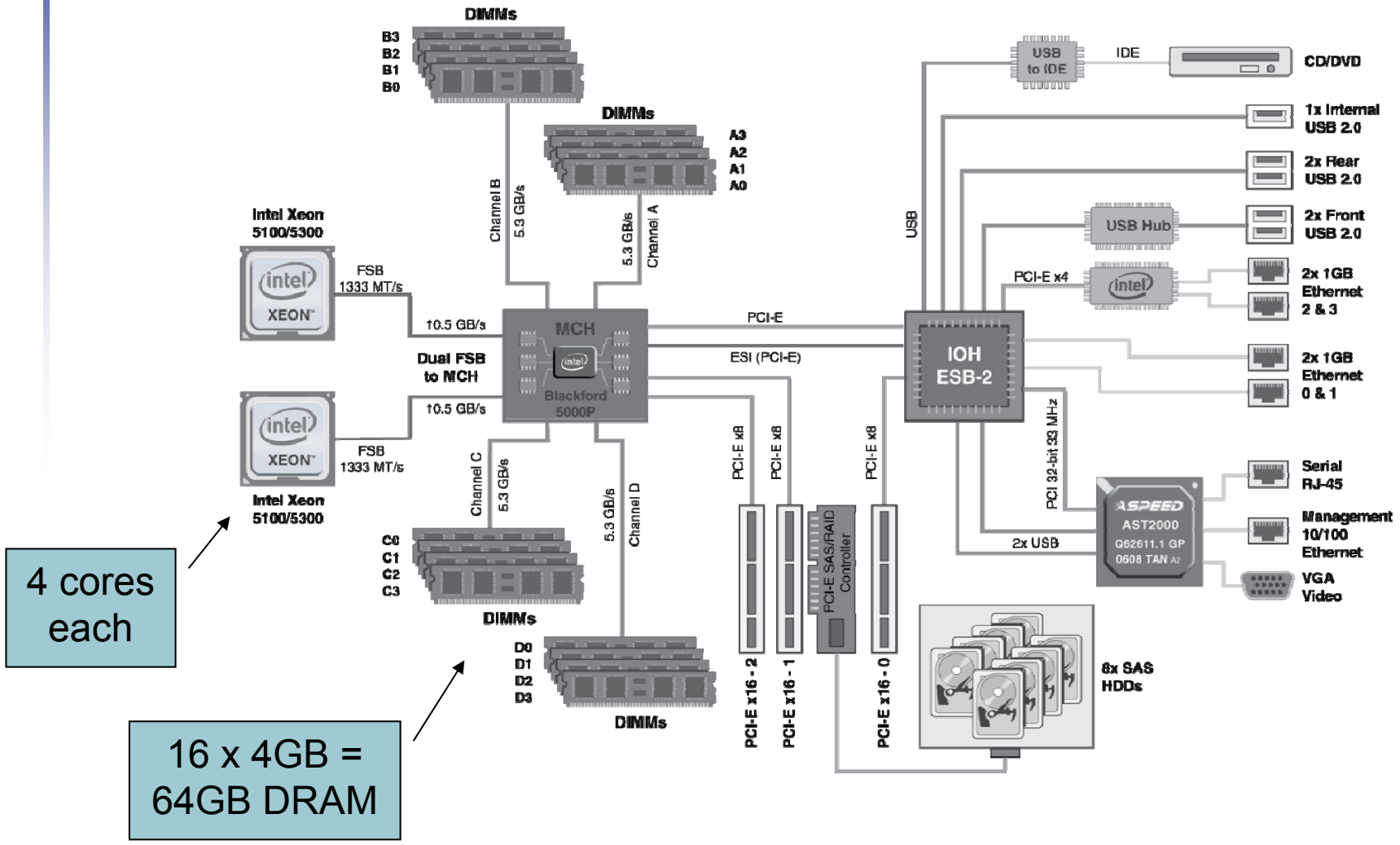
Rack-Mounted Servers

Sun Fire x4150 1U server



Standard 19” rack with 42 1U server

Sun Fire x4150 1U server



4 cores each

16 x 4GB = 64GB DRAM

I/O System Design Example

- Given a Sun Fire x4150 system with
  - Workload: 64KB disk reads
    - Each I/O op requires 200,000 user-code instructions and 100,000 OS instructions
  - Each CPU: 10^9 instructions/sec
  - FSB: 10.6 GB/sec peak
  - DRAM DDR2 667MHz: 5.336 GB/sec
  - PCI-E 8× bus: 8 × 250MB/sec = 2GB/sec
  - Disks: 15,000 rpm, 2.9ms avg. seek time, 4ms rotational latency, 112MB/sec transfer rate
- What I/O rate can be sustained?
  - For random reads, and for sequential reads

Design Example (cont)

- Maximum I/O rate for CPUs
  - Per core: 10^9/(100,000 + 200,000) = 3,333 IOPS (I/Os per sec)
  - 8 cores: 26,667 IOPS
- Random reads, I/O rate for disks
  - Assume actual seek time is average/4
  - Time/op = seek + latency + transfer
    = 2.9ms/4 + 4ms/2 + 64KB/(112MB/s) = 3.3ms
  - 1000ms/3.3ms = 303 ops/sec per disk, 2424 ops/sec for 8 disks
- Sequential reads
  - 112MB/s / 64KB = 1750 I/O ops/sec per disk
  - 14,000 I/O ops/sec for 8 disks

Design Example (cont)

- PCI-E I/O rate
  - 2GB/sec / 64KB = 31,250 I/O ops/sec
- DRAM I/O rate
  - 5.336 GB/sec / 64KB = 83,375 I/O ops/sec per DIMM
- FSB I/O rate
  - Assume we can sustain half the peak rate
  - 5.3 GB/sec / 64KB = 81,540 I/O ops/sec per FSB
  - 163,080 I/O ops/sec for 2 FSBs
- Weakest link: disks
  - 2424 ops/sec random, 14,000 ops/sec sequential
  - Other components have ample headroom to accommodate these rates

Sun Fire x4150 1U Server Power

	Components			System			
Item	Idle	Peak	Number	Idle		Peak	
Single Intel 2.66 GHz E5345 socket, Intel 5000 MCB/IOH chip set, Ethernet controllers, power supplies, fans, . . .	154 W	215 W	1	154 W	37%	215 W	39%
Additional Intel 2.66 GHz E5345 socket	22 W	79 W	1	22 W	5%	79 W	14%
4 GB DDR2-667 5300 FBDIMM	10 W	11 W	16	160 W	39%	176 W	32%
73 GB SAS 15K Disk drives	8 W	8 W	8	64 W	15%	64 W	12%
PCIe x8 RAID Disk controller	15 W	15 W	1	15 W	4%	15 W	3%
Total	—	—	—	415 W	100%	549 W	100%

- SPECJBB with 29 different configurations

Power Evaluation

- How can we reconfigure x4150 to minimize power, assuming workload discussed earlier is the only activity on this 1U server?

- To achieve 2424 random 64 KB reads/sec
  - Need all 8 disks, PCI RAID controller
  - A single DIMM supports over 80,000 IOPS
    - We can save power in memory
    - Minimum memory for x4150 is two DIMMS, so we can save the power and cost of 14 4GB DIMMs!
  - Single socket supports 13,333 IOPS
    - Can reduce number of Intel E5345 sockets by one
  - Total System Power:
    - Idle Power (random reads) = 154 + 2x10 + 8x8 + 15 = 253 Watts
    - Peak Power (random reads) = 215 + 2x11 + 8x8 + 15 = 316 Watts

Power Evaluation

- To achieve 14,000 64KB sequential reads/sec
  - Need all disks and disk controller
  - Same number of DIMMS can handle this higher load
  - But workload exceeds processing power of a single Intel E5345 socket (13,333 IOPS), so we need to add another one
  - Total System Power
    - Idle Power (seq reads) = 154 + 22 + 2x10 + 8x8 + 15 = 275 Watts
    - Peak Power (seq reads) = 215 + 79 + 2x11 + 8x8 + 15 = 395 Watts
    - Reduction in power by a factor of 1.4 to 1.5

Fallacy: Disk Dependability

- If a disk manufacturer quotes MTTF as 1,200,000hr (140yr)
  - A disk will work that long
- Wrong: this is the mean time to failure
  - Typical lifetime: 5 years or 43,800 hours
  - What if you have 1000 disks, used 24 hrs/day
    - How many will fail per year?

$$\text{Annual Failure Rate (AFR)} = \frac{1000 \text{ disks} \times 8760 \text{ hrs/disk}}{1200000 \text{ hrs/failure}} = 0.73\%$$

Fallacies

- Disk failure rates are as specified
  - Studies of annual failure rates in the field for 100,000 ATA and SCSI disks
    - Schroeder and Gibson: 2% to 4% vs. 0.6% to 0.8%
    - Pinheiro, *et al.*: 1.7% (first year) to 8.6% (third year) vs. 1.5%
  - Why?
- A 1GB/s interconnect transfers 1GB in one sec
  - Fortunate to get 70%-80% of peak bus bandwidth
  - What's a GB?
  - 1GB = 2^{30} B = 1.075×10^9 B
  - For I/O, storage bandwidth, use 1GB = 10^9 B
  - So 1GB/sec is 0.93GB in one second
  - About 7% error (see Creative lawsuits, circa 2008)

Pitfall: Offloading to I/O Processors

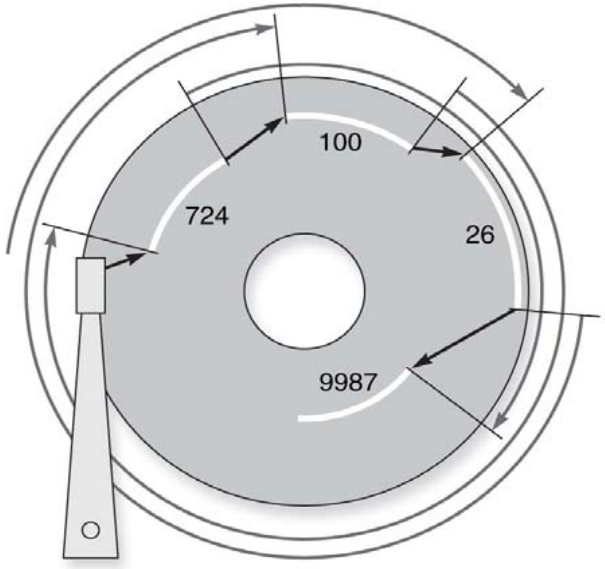
- Latency overhead of setting up I/O request may dominate
  - Quicker to do small operation on the CPU
- I/O processor usually slower than CPU
  - Since it's supposed to be simpler
- Making it faster makes it into a major system component
  - Might need its own coprocessors!

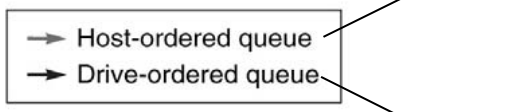
Pitfall: Backing Up to Tape

- Magnetic tape used to have advantages
  - Removable, high capacity
- Advantages eroded by disk technology developments
- Makes better sense to replicate data
  - e.g, RAID, remote mirroring

Fallacy: Disk Scheduling

- Best to let the OS schedule disk accesses
  - But modern drives deal with logical block addresses
    - Map to physical track, cylinder, sector locations
    - Also, blocks are cached by the drive
  - OS is unaware of physical locations
    - Reordering can reduce performance (4x in example below)
    - Depending on placement and caching





Op	Start	Length
Read	26	128
Read	100	16
Read	724	8
Read	9987	1

Read	724	8
Read	100	16
Read	26	128
Read	9987	1

Pitfall: Peak Performance

- Using peak transfer rates of a portion of I/O system to make performance projections/ comparisons
  - Peak I/O rates are nearly impossible to achieve
    - Usually, some other system component limits performance
    - E.g., transfers to memory over a bus
      - Arbitration contention with other bus masters
    - E.g., PCI bus: peak bandwidth ~133 MB/sec
      - In practice, max 80MB/sec sustainable

Concluding Remarks

- I/O performance measures
  - Throughput, response time
  - Dependability and cost also important
- Buses used to connect CPU, memory, I/O controllers
  - Polling, interrupts, DMA
- I/O benchmarks
  - TPC, SPECSFS, SPECWeb
- RAID
  - Improves performance and dependability