I/O

formerly an orphan in the architecture domain

❑ Before

- I/O meant the non-processor and memory stuff
  - disk, tape, LAN, WAN, etc.
  - performance was not a major concern
- devices characterized as:
  - extraneous, non-priority, infrequently used, slow
- exception is swap area of disk
  - part of the memory hierarchy
  - hence part of system performance but you're hosed if you use it often
  - a big deal for business applications - importance varies in scientific app's

I/O Now

❑ Trends

- market is communications bonkers
  - voice response & transaction systems, real-time video
  - multimedia expectations
  - creates many modes - continuous/fragmented, common/rare, dedicated/interoperable
  - diversity is even more chaotic than before
  - single bucket I/O didn't work before - worse now
- even standard networks come in gigabit/sec flavors
- for multicomputers
- I/O is the bottleneck

❑ Result

- significant focus on system bus performance
  - common bridge to the memory system and the I/O systems
  - critical performance component for the SMP server platforms

I/O Comments

❑ I/O differentiates today's machines

- workstation - CPU, memory, CRT, keyboard, mouse, disk, LAN
- server - CPU, more memory, more disk, LAN
- fast machine - lots of PE's
  - CPU + memory + tight coupled interconnect & a disk array

❑ PC's are a bit different

- like a workstation but with cheaper/slower everything
- low power is a concern for the portable machine

❑ But not for long

- the workstation CPU of today is often just one family member
  - low power and low cost versions show up subsequently
- Intel is moving into the high end
- key is common software base - i.e. more profit

System vs. CPU Performance

❑ Care about speed at which user jobs get done

- throughput - how many jobs/time
- latency - how quick for a single job

❑ CPU performance main factor when:

- job mix is fits in the memory
  - namely there are very few page faults

❑ I/O performance main factor when:

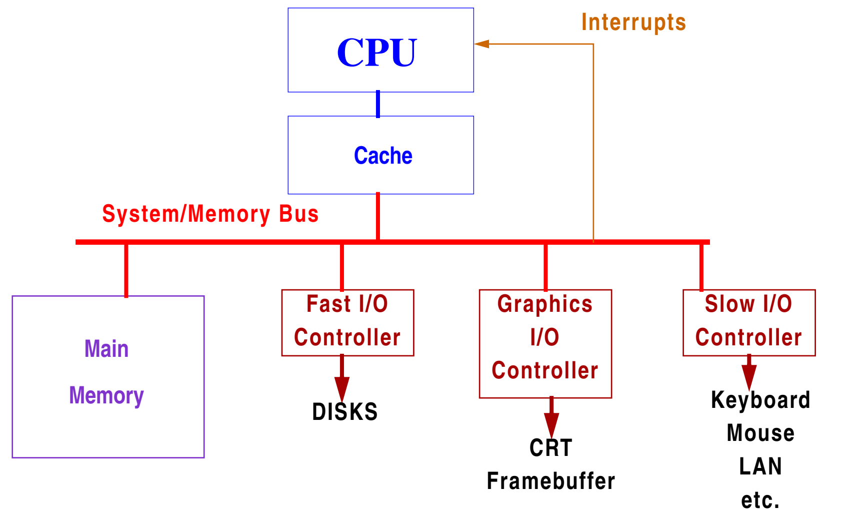
- the job is too big for the memory - paging is dominant
- when the job rips into lots of unexpected files
  - OLTP
  - decision support
  - database
  - almost every commercial application you can imagine
- and then there is graphics & specialty I/O devices

System P erformance

depends on lots of things in the worst case

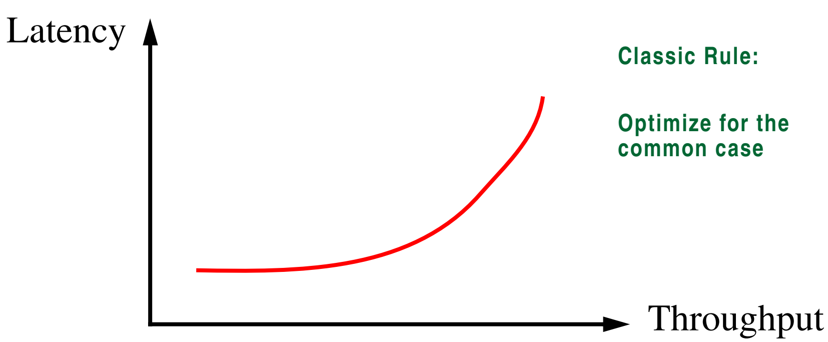
- CPU
- compiler
- operating System
- cac he
- main Memory
- memory-IO bus
- I/O controller or c hannel
- I/O drivers and interrupt handlers
- I/O devices: there are many types
  - level of autonomous beha vior
  - amount of internal buffer capacity
  - + device specific parameters for latency and throughput

A Typical System



Keys to a Balanced System

- It's all about overlap - I/O vs CPU
  - $\text{Timework}_{\text{load}} = \text{Time}_{\text{cpu}} + \text{TimeI/O} - \text{Time}_{\text{overlap}}$
- Consider the benefit of just speeding up one
- Latency vs . Bandwidth



I/O System Design Considerations

- Depends on type of I/O device
  - size , bandwidth, and type of transaction
  - frequency of transaction
  - defer vs . do now
- Appropriate memory bus utilization
- What should the controller do
  - programmed I/O
  - interrupt vs . polled
  - priority or not
  - DMA
  - buffering issues - what happens on over -run
  - protection
  - validation

Types of I/O Devices

Consider

- Beha vior
  - read, Write , Both vs . once , multiple
  - size of a verage transaction
  - bandwidth
  - latency
- Partner - the speed of the slowest link theory
  - human operated (interactive or not)
  - mac hine operated (local or remote)

Some I/O Device Characteristics

Device	Behavior	Partner	Data Rate KB/sec
Keyboard	Input	Human	0.01
Mouse	Input	Human	0.02
Voice Input	Input	Human	0.20
Scanner	Input	Human	200.00
Voice Output	Output	Human	0.60
Impact Printer	Output	Human	1.00
Laser Printer	Output	Human	100.00
Graphics Display	Output	Human usually	30,000.00
CPU to frame buffer	Output	Machine/Human	200.00
Network-terminal	I/O	Machine	0.05
Network-LAN	I/O	Machine	200 - ??
Optical disk	Storage	Machine	500.00
Disk	Storage	Machine	2,000.00 - 8,000.00
Tape	Storage	Machine	2,000.00

Important I/O?

- Depends on your application
  - business - disks for file system I/O
  - graphics - graphics cards or specialty co-processors
  - parallelism - the communications fabric
- Our focus = mainline uniprocessing
  - storage subsystems
  - networks
- Noteworthy P oint
  - the traditional orphan
  - but now often viewed more as a front line topic

Magnetic Disks

- 2 important Roles
  - non-volatile storage
    - the file system and the operating system
    - inc luding most of what happens when you boot up
  - lowest level of the memory hierarc hy
    - most of the virtual memory is physically resident on the disk
- Long viewed as a villain
  - mec hanical system ==> slow
  - hence they seem to be an easy target for improved tec hnology
  - disk improvement w .r.t. density ha ve done better than Moore' s la w
  - interesting that the revenue from disks > than from silicon

Physical Organization Options

- Platters - 1 or more
- Density - fixed or variable
  - bits/inch
  - tracks/inch
- Organization - sectors, cylinders, and tracks
  - actuators - 1 or more
  - heads - 1 per track or 1 per actuator
  - access - seek time vs. rotational latency
    - seek related to distance but not linearly
    - typical rotation: 3600 RPM or 5400 RPM
    - higher rpm rate as time progresses
- Diameter - 8” to 1.25” now common, .5” coming
- Addressing - index (physical and logical)

Typical Physical Organization

- Multiple platters
  - coated top and bottom of each platter
- Single actuator (since they are expensive)
  - single R/W head per arm
  - one arm per surface
  - all heads therefore over same cylinder
- Fixed sector size
  - IBM permits soft sectored disk configuration
- Variable density encoding
  - although fixed density seems to be making a comeback
- Disk controller
  - usually built in processor + buffering

Anatomy of a Read Access

- Steps
  - memory mapped I/O over bus to controller
  - controller starts access
  - seek + rotational latency wait
  - sector is read and buffered (validity checked)
  - controller says ready or DMA’s to main memory and then says ready
- Calculating Access Time

$$\text{Access Time} = \text{Controller Delay} + \frac{\text{Block Size}}{\text{Bandwidth}} + \frac{0.5}{\text{RPM}} + \text{Seek Time} + \text{Pick Time} + \text{Check Delay}$$

seek time is very non-linear: accelerate and decelerate times complicate actual value

time for head to pass over sector

Seagate ST31401N (1993)

- 5.25” diameter
- 2.8 GB formatted capacity
- 2627 cylinders
- 21 tracks/cylinder (why is this an odd number?)
- 99 sectors per track - variable density encoded
- 512 bytes per sector
- rotational speed 5400 RPM
- average seek - random cylinder to cylinder = 11 ms
- minimum seek = 1.7 ms
- maximum seek = 22.5 ms
- transfer rate = 4.6 MB/sec

Disk Trends

❑ bits/unit area is common improvement metric

$$\text{Areal Density} = \frac{\text{Tracks}}{\text{Platter Surface Inch}} \times \frac{\text{Bits}}{\text{Track Inch}}$$

- assumption is fixed density encoding
- may not get this since variable density control is cheaper

❑ Trends

- Until 1988 - 29% improvement per year
  - doubles density in three years
- After 1988 (and thin film oxide deposition) - 60% per year
  - 4x density in three years

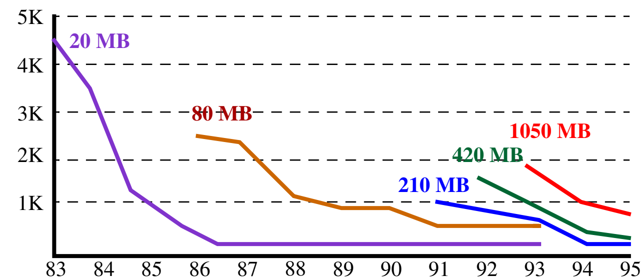
❑ Today

- 644 Mbits/in ^{2}
- 3 Gb/in ^{2} demonstrated in labs

Disk Price Trends

❑ Personal Computer (as advertised)

❑ In 1995 Dollars



Cost per Megabyte

❑ Improved 100x in 12 years

- 1983: $300
- 1990: $9
- 1995: $0.90

❑ Comparison 1995

- SRAM chips: 200 $/MB with access times around 10 ns
- DRAM chip: 10 $/MB with access times around 800 ns
  - there are chips with access times around 300 ns with 40 $/MB
- DRAM boards: $20 $/MB and access times around 1 us
- Disk: .9 $/MB access times around 100 msec

❑ Disk futures look bright

- all doomsday predictions have failed - e.g. bubble memories

Disk Alternatives

❑ DRAM and a battery

- big reduction in seek time and lower latency
- more reliable as long as the battery holds up
- cost is not attractive
- prediction: they'll fail just like bubbles
  - only some reasons will differ

❑ Optical Disks

- CD's are fine for archival storage due to their write once nature
  - they're also cheap and high density
- unlikely to replace disks until many writes are possible
  - numerous efforts at creating this technology
  - none likely to mature soon however
- role is likely to be for archival store and SW distribution

Other Devices

- Robotic Tape Storage
  - STC Powderhorn - 60 TB at 50$/GB
    - could hold the library of Congress in ASCII
  - problem is tapes tend to wear out so rare access role is key
- Optical Juke Boxes
  - now latency depends on even more mechanical gizmo's
- Tapes - DAT vs. the old stuff
  - amazing how dense you can be if you don't have to be right
- The new storage frontiers - terabyte per cm^3
  - plasma
  - biological
  - holographic spread spectrum
  - none are anywhere close to deployment

I/O Connection Issues

connecting the CPU to the I/O device world

- Typical choice is a bus - advantages
  - shares a common set of wires and protocols ==> cheap
  - often based on some standard - PCI, SCSI, etc. ==> portable
- But significant disadvantages for performance
  - generality ==> poor performance
    - lack of precise loading model
    - length may also be expandable
  - multiple devices imply arbitration and therefore contention
  - can be a bottleneck
- Common choice: multiple busses in the system
  - fast ones for memory and CPU to CPU interaction (maybe I/O)
  - slow for more general I/O via some converter or bridge IOA

Bus Design Options

Option	High Performance	Low Cost
Bus Width	Separate Address and Data Lines	Multiplex Address and Data Lines
Data Width	Wider is faster	Narrower is cheaper
Transfer Size	Multiple words	Single word or even byte
Mastership	Multiple masters	Single Master (usually the CPU)
Split Transaction	Yes - separate request and replies	No - just keep the sucker till you're done
Timing	Synchronous - clock skew limits to short	Asynchronous = handshake protocols - easy to expand as well as make long

- Other things to standardize as well
  - connectors
  - voltage and current levels
  - physical encoding of control signals
  - protocols for good citizenship

Standards

- The Good
  - provides a path for second party (e.g. cheaper) competition
  - you get useful gizmo's the main vendor can't be bothered with
- The Bad
  - become major performance anchors
  - inhibit change
- The Ugly (The Path)
  - bottom-up
    - company tries to get standards committee to approve it's latest philosophy in hopes that they'll get the jump on the others (e.g. S bus, PC-AT bus, ...)
  - top-down
    - design by committee and hence doesn't work well for anybody (PCI, SCSI, ...)

Some Sample I/O Bus Designs

note S-bus and MicroChannel started as memory busses

Feature	S Bus	MicroChannel	PCI	IPI	SCSI 2
Data Width	32 bits	32 bits	32 to 64 bits	16 bits	8 to 16 bits
Clock Rate	16-25 MHz	Asynchronous	33 MHz	Asynchronous	10 MHz or Asynchronous
Number of Masters	Multiple	Multiple	Multiple	Single	Multiple
Bandwidth (32 bit reads)	33 MB/sec	20 MB/sec	33 MB/sec	25 MB/sec	20 or 6 MB/sec
Peak Bandwidth	89 MB/sec	75 MB/sec	111 MB/sec	25 MB/sec	20 or 6 MB/sec
Standard	None (SUN)			ANSI X3.129	ANSI X3.131

Compare with Memory Busses

- 1994 Servers - all use split transactions

Feature	HP Summit	SGI Challenge	Sun XDBus
Data Width	128 bits	256 bits	144 bits
Clock Rate	60 MHz	48 MHz	66 MHz
Number of Masters	Multiple	Multiple	Multiple
Bandwidth peak	960 MB/sec	1200 MB/sec	1056 MB/sec
Slots	16	9	10
Standard	None	None	None

- Note:
  - custom ASIC required to interface device to limit loading
  - slots are all in the same cabinet for length control
  - all provide some coherency transactions
    - for multiple CPU SMP configurations

Connecting the I/O Bus

- To main memory
  - since cac he misses are rare doesn't tend to stall the CPU
  - problem is lac k of coherency
- To cac he (means I/O bus = memory bus)
- Access
  - memory mapped or distinct instruction
  - control register format will determine the protocol
- Interrupt vs . Polling
- DMA or not
  - autonomous control allows overlap and latency hiding
  - however there is a cost impact