Interfacing the I/O Subsystem - Issues

- **What talks to the devices?**
  - Programmed I/O: the CPU
  - Delegation of responsibility:
    - DMA (Direct Memory Access): block transfers
    - I/O Processor: more general coprocessor
    - RDMA (DMA to/from a remote processor)

- **Where does the I/O data path attach?**
  - cache
  - main memory

- **How do devices draw the CPU’s attention?**
  - Polling vs. interrupts
  - Identifying the requester
  - Priority arbitration

- **Communication channels**
  - shared (bus) vs. point-to-point
  - synchronous vs. asynchronous

- **Capacity, Access Time, Bandwidth**
I/O Communication - Trends

- From shared bus to switched
- **Semantics:**
  - Send/Receive
  - DMA / RDMA
- **InfiniBand**
  - **Semantics:**
    » Send/Receive
    » RDMA
  - Transfer rate: 20Gb/s, full duplex end-to-end
  - Latency: several microseconds
  - Support for quality of service
  - Included in a growing number of the top 500 computers
  - Israel leads the world (Mellanox; Voltaire)
On-Chip Communication Trends

• From shared bus to switched

• Network on chip with QoS
Operating System: The Brain

• Mission: provide an environment for running user programs conveniently and efficiently.
  – management of shared resources.
  – separation and protection among different users and between users and the system.
  – provide a “standard” representation of the machine to user programs.

• Main tasks:
  – Management of processes (create, stop, resume, communication, deadlock-avoidance)
  – Main-memory management: decide which processors to load in to memory that has been freed; memory allocation
  – Secondary-mem management: allocate space on disk and manage schedule.
  – Manage I/O system: buffer caching; interface to device drivers; I/O queues (printing etc.).
  – File system
  – Protection
  – Communication
  – Command interpretation
  – accounting
Hardware Support for the Operating System

• **Timers**
  – periodic interrupts
  – prevent processor-stealing by user process
  – prevent hang-up on infinite loops or errors
  – support time-dependent actions

• **Support interrupts and masks**

• **Base and bound registers for protection**

• **Dual-mode operation (user/system mode)**

• **Mode bit**
Hardware + OS performance

- Polling vs. interrupts
- Frequent process switch ==> cache flush ==> thrashing
- Data placement on disk determines performance.
  - In Unix, placing data on one disk drive and i-nodes on another dramatically increased performance
  - In video servers, accommodate dependence of transfer rate on track location
- Hardware support for interrupt-source determination improves performance.
Multi-Processor/Computer Architectures

- **Symmetric multiprocessor:**
  - Shared bus and memory
  - A processor that becomes available grabs the next task in the queue

- **Single-instruction – multiple data (SIMD)**

- **Multiple instruction – multiple data (MIMD)**
  - shared memory
    - UMA
    - NUMA
    - or:
  - message passing
Multi-Computer Architectures

• Parallel computers

• Networks of workstations

• Grid computing
  Looser and looser coupling among the machines (e.g., each runs its own OS, interconnection through standard networks)

• Challenges include:
  – scalability
  – latency hiding
  – security
  – fault tolerance
Course Summary

• Goals:
  – learn and understand the components of a modern computer system and the interaction among them.
  – understand the computer as a system, not merely a collection of modules.
  – learn to “think systems”:
    » interplay among components
    » trade-offs

• Main topics:
  – Performance measures
  – Memory system
  – Processor (control)
  – I/O

• Solving a systems problem:
  – Understand it: build a mental picture of the system
  – Use common sense
  – Employ specific techniques in the solution
  – Evaluate the solution against alternatives
Related Courses

- Advanced computer and VLSI architecture
- Compilers (particularly back end)
- VLSI
- Operating systems
- Tools for analysis of computer systems
Exam

• Take the time to read the problems
• Every line serves to restrict the problem, so build an entire picture in your mind before beginning to answer
• Be concise and precise:
  – answer the question being asked and only it
  – use accepted terminology
• Manage your time smartly!
• Don’t look for close matches with previous problems. Keep your head on your shoulders and think!
• Do your best and don’t worry; the grade is our responsibility!

Good luck!