Chapter 2: Computer-System Structures

- Computer System Operation
- Storage Structure
- Storage Hierarchy
- Hardware Protection
- General System Architecture

Computer-System Architecture
Computer-System Operation

- I/O devices and the CPU can execute concurrently.
- Each device controller is in charge of a particular device type.
- Each device controller has a local buffer.
- CPU moves data from/to main memory to/from local buffers.
- I/O is done from the device to local buffer of controller.
- Memory controller is used to ensure orderly access to shared memory. So, its function is to synchronize access to the shared memory.
- Device controller informs CPU that it has finished its operation by causing an interrupt.

Interrupt Time Line For a Single Process Doing Output
Storage Structure

• Main memory – only large storage media that the CPU can access directly. The programs must be in main memory to be executed.

• Secondary storage – extension of main memory that provides large nonvolatile storage capacity.

• Magnetic disks – rigid metal or glass platters covered with magnetic recording material
  – Disk surface is logically divided into tracks, which are subdivided into sectors.
  – The disk controller determines the logical interaction between the device and the computer.

Moving-Head Disk Mechanism
Magnetic Disk Mechanism

- The two surfaces of a platter are covered with magnetic material.
- Platter diameters range from 1.8 to 5.25 inches.
- There may be thousands of concentric cylinders in a disk drive, and each track may contain hundreds of sectors.
- The transfer rate is the rate at which data flow between the drive and the computer.
- The positioning time (or random access time), consists of seek time and rotational latency.
- The seek time is the time to move the disk arm to the desired cylinder.
- The rotational latency is the time for the desired sector to rotate to the disk head.

Storage Structure

- Floppy disks consist of one platter and the head sits directly on the surface. Its inexpensive, less storage 1.4MB and slower compare it to the hard disk. Also, it is removable.
- The load instruction moves a word from main memory to an internal register, within the CPU for execution.
- The store instruction moves the content of a register to main memory.
- Can we store the programs and data in main memory permanently? The answer is no for two reasons:
  - Main memory is small to store all programs and data.
  - Main memory (RAM) is volatile storage device that loses its contents when power is turned off or otherwise lost.
- Therefore, to store all programs and data permanently you use secondary storage such as hard disk and diskette.
Storage Hierarchy

- Storage systems organized in hierarchy.
  - Speed
  - Cost
  - Volatility

- Caching – copying information into faster storage system; main memory can be viewed as a last cache for secondary storage.

Storage-Device Hierarchy

Size
- Small
- Large

Fast and Expensive
- Slow and Cheap
Storage-Device Hierarchy

- Registers, cache, and main memory are volatile.
- All storage after main memory are non-volatile.
- Caching: Check first cache memory if data not there go to main memory and copy it into cache under the assumption that there is a high probability that it will be needed again.
- Data must be moved from secondary storage into main memory before use.
- Data transfer from cache to CPU and registers is usually a hardware function with no operating system control.
- Data transfer from disk to memory is usually controlled by the operating system.
- Cache Coherency and Consistency is the state that exists in a multiprocessor system, when any shared data is held by 2 or more caches, and no 2 caches hold different values of such a shared data simultaneously.

Hardware Protection

- Dual-Mode Operation
- I/O Protection
- Memory Protection
- CPU Protection
Dual-Mode Operation

• Protection is needed for any shared resource.

• Sharing system resources requires operating system to ensure that an incorrect program cannot cause other programs to execute incorrectly.

• Provide hardware support to differentiate between at least two modes of operations.
  1. User mode – execution done on behalf of a user.
  2. Monitor mode (also supervisor mode or system mode) – execution done on behalf of operating system.

Dual-Mode Operation (Cont.)

• At system boot time, the hardware starts in monitor mode. The O.S. is then loaded, and starts user processes in user mode.

• Whenever, an interrupt occurs, the hardware switches from user mode to monitor mode.

• Whenever, the O.S. gains control of the computer, it is in monitor mode.

• If you do not have dual mode then you can wipe or write over the O.S. Example: MS-DOS for 8088 architecture does not have a dual mode.

• MS-Windows NT and IBM OS/2 take advantage of dual mode feature and provide greater protection for the O.S.
Dual-Mode Operation (Cont.)

• Mode bit added to computer hardware to indicate the current mode: monitor (0) or user (1).
• When an interrupt or fault occurs hardware switches to monitor mode.

Privileged instructions can be issued only in monitor mode.

I/O Protection

• All I/O instructions are privileged instructions.
• Must ensure that a user program could never gain control of the computer in monitor mode (i.e., a user program that, as part of its execution, stores a new address in the interrupt vector).
Memory Protection

- We want to protect the O.S. from access by user programs, and to protect user programs from one another.
- In order to have memory protection, add two registers that determine the range of legal addresses a program may access:
  - base register – holds the smallest legal physical memory address.
  - Limit register – contains the size of the range
- Memory outside the defined range is protected.

A Base And A limit Register Define A Logical Address Space

Limit register = 300040 - 420940
Protection Hardware

- This protection is accomplished by the CPU hardware comparing every address generated in user mode with the registers.
- The base and limit registers can be loaded by only the O.S.

General System Architecture

User of A System Call to Perform I/O:

- I/O instructions can be executed by only the O.S.
- To do I/O, a user program executes a system call to request that the O.S. perform I/O.