CS 340
Lecturer: Dr. Simina Fluture

Lecture #3

Read: Web lecture and class notes

Topics: Computer-System Structures
Starting a Computer
Interrupts
Dual-mode operations
I/O Structure (only what it is covered in class)

Starting a Computer: In order to start running (whether it is started up or rebooted), a computer needs to have an initial program to run; called a **bootstrap program**. When the CPU is started, it branches to a fixed memory location. This hardwired address points to the ROM, which contains a small program called the bootstrap loader (program). This program initializes all aspects of the system, which include locating the operating-system kernel and loading it into memory. The OS then starts executing the first process, and waits for some **event** to occur.

Interrupts:
The occurrence of an event is signaled by an interrupt from the hardware (sending a signal to the CPU) or the software (by executing a special operation called the **system call**).
An interrupt can be *synchronous* if it is the DIRECT result of the current instruction being executed by the CPU. Otherwise it is known as *asynchronous*.

Dual-Mode Operation (processor modes):
A mode bit, is added to the hardware of the computer to indicate the current mode.
User-mode (1) the operation is done on behalf of the user.
System (monitor) -mode (0) the operation is done on behalf of the OS.

Privileged - instructions are instructions that can be executed only in monitor mode.

CS 340
Lecturer: Dr. Simina Fluture

I/O Structure

Programmed I/O

Data is exchanged between the CPU and the I/O module whenever the CPU issues a command to the I/O module. Most of the time the CPU will wait until the I/O operation is complete.

Interrupt Driven I/O

The CPU issues a command requesting I/O. Then the CPU will continue to execute other instructions, until the I/O module completes its work. At that point, the I/O module will issue an interrupt.

Both Programmed I/O and Interrupt Driven I/O require the CPU to manage the extraction of data from the main memory for output and the storing of data into the main memory for input.

Direct Memory Access (DMA)

The I/O module and main memory exchange data directly without CPU involvement.

Storage Structure

Main (primary) memory and the CPU registers are the only storage areas that the CPU can access directly.
1. Main memory is usually too small to store all needed programs and data.
2. Main memory is volatile (especially RAM).

If the data are not in memory, they must be brought into the memory before the CPU can operate on them.

Secondary memory (storage) is an extension of main memory. The secondary memory is slower but is able to hold larger quantities of data permanently.

As noted above, there is a tradeoff between the speed and cost of memory, with faster memory being more expensive and volatile.

Memory Protection

Note: This topic will be considered in more detail at the end of the semester.
Memory used by the OS programs has to be protected from access by user programs. Memory occupied by a user program also has to be protected from being accessed by other user programs.

There are 2 registers used to accomplish this:
The **Base register** holds the smallest address of the memory that can be accessed by a given process.
The **Limit register** holds the size of the range of memory allocated to a given process.

Memory protection is handled by the CPU hardware every time a request for memory access takes place in user mode. It compares the address of the memory request to the content of these registers, and decides whether or not the address is of a legal memory location. Only the OS, using privileged mode instructions, can change the content of the base and limit registers.