CSCI 460—Operating Systems

Lecture 01

Textbook: Operating Systems — Internals and Design Principles (9th edition) by William Stallings

- Course Homepage:
  - https://github.com/adiesha/CSCI460Fall2024
- Basic operating systems (2/3) | Advanced operating systems (1/3)
- Basic operating systems
  - Memory management.
    - Paging
    - Memory hierarchy
  - Processor management.
    - Basic concepts: Processor, process, job, etc.
    - Process scheduler.
    - Scheduling policy.
    - Deadlock concepts, handling deadlocks, avoidance, prevention, etc.
    - Starvation.
  - Device management.
    - Device handler strategies, etc.
  - File management.
    - Basic definitions.
    - File organization.
    - Access control.
- Advanced operating systems
  - Threads
    - How to create threads in Linux.
  - Symmetric multiprocessing
  - Multiprocessor scheduling
    - Mutual exclusion.
    - Algorithms for mutual exclusion (centralized, distributed etc.).
    - Real-time scheduling.
  - Networking
    - Distributed Mutual Exclusion Concepts.
  - Security

- Diffie-Hellman algo
- RSA

- Focus more on concepts and algorithms of OS and how OS works.
- We will use c programs under Linux operating system to understand concepts.
- You are required to read the sections of the book that is assigned.
- GitHub is the main repository of lecture notes, assignments, and solutions, but I will try to update them in D2L as well. I expect you to understand how git and GitHub works by now.
- Discord will be used for announcements. Please check the discord channel for announcements.
- Evaluation
  - Random Pop-up tests (in class) 10% (I will allow 1 or 2 to be dropped: I haven't decided on how many yet.)
  - 5 homework (D2L quizzes) 20%
  - 3 in class tests (30%)
  - 3 programming assignments (24%)
  - 1 final project (16%)

1. Name some operating systems (OS) that you have heard of

    a. ____

    b. ____

    c. ____

    d. ____

    e. ____

    f. ____

    g. ____

    h. ____

i. ____

j. ____

Computer system overview

- You already have an idea about the overview of the computer system from the computer systems class (CSCI 366).
- Here I would touch briefly about the computer system overview.

Basic Elements

- Processor: Controls the operation of the computer and performs its data processing functions.
- Main memory: Controls the operation of the computer and performs its data processing. functions.
- I/O modules: Move data between the computer and its external environment.
- System Bus: Provides for communication among processors, main memory, and I/O modules.

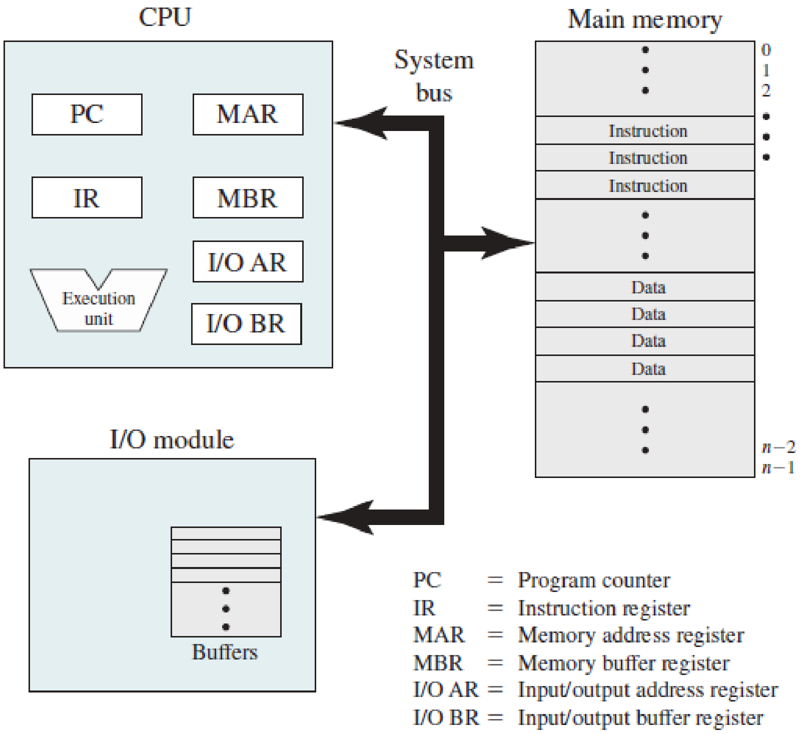

CPU
PC
MAR
IR
MBR
I/O AR
Execution unit
I/O BR
System bus
Main memory
0
1
2
Instruction
Instruction
Instruction
Data
Data
Data
Data
n−2
n−1
I/O module
Buffers
PC = Program counter
IR = Instruction register
MAR = Memory address register
MBR = Memory buffer register
I/O AR = Input/output address register
I/O BR = Input/output buffer register

Figure 1.1 **Computer Components: Top-Level View**

Instruction cycle

- Instruction processing consists of two steps:
  - Fetches the instruction from the memory one at a time.
  - Executes the instruction.
- Program execution consists of repeating the process of instruction fetch and instruction execution.
- Processing required for single instruction is called an instruction cycle.
- Program execution halts only if the processor is turned off, some sort of unrecoverable error occurs, or a program instruction that halts the processor is encountered.

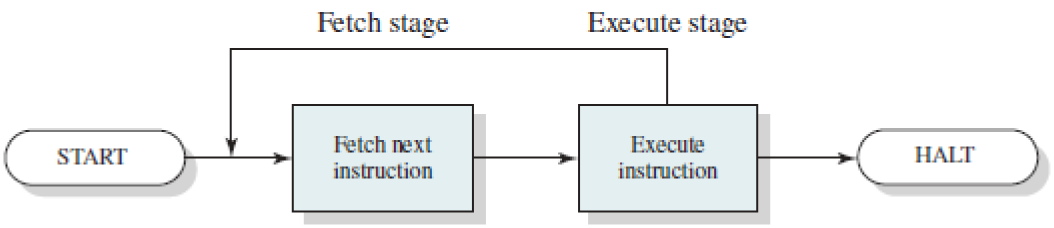


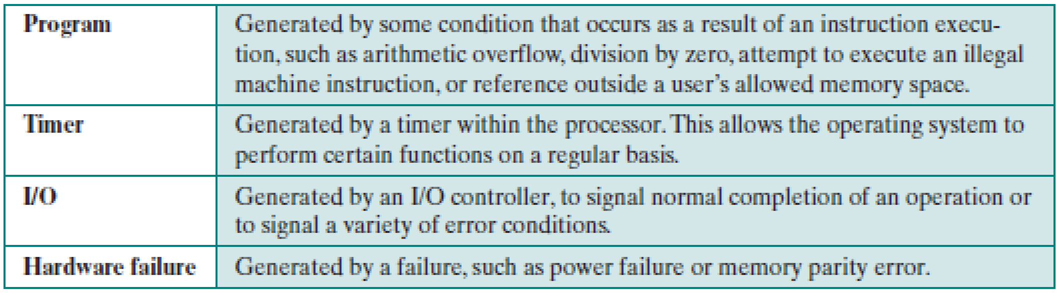
Figure 1.2 **Basic Instruction Cycle**

- Typically, the <u>program counter (PC)</u> holds the address of the next instruction to be fetched.
- Processor always increments the PC after each instruction fetch cycle unless instructed otherwise.
- The fetched instruction is loaded into the <u>instruction register (IR)</u>
  - The instruction contains bits that specify the action the processor is to take.
- Typically, these actions fall into 4 categories:
  - Processor ←→ memory: Data transfer between processor and memory (both ways)
  - Processor ←→ I/O: Data transfer between to or from a peripheral device by transferring between processor and I/O module.
  - Data Processing: Arithmetic and logical operations on Data.
  - Control: An instruction may specify that sequence of execution be altered.

Interrupts

- Virtually all computers provide a mechanism by which other modules (I/O, memory) may interrupt the normal sequencing of the processor.
- There are several types of Interrupts:

Table 1.1 Classes of Interrupts



Program	Generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, or reference outside a user's allowed memory space.
Timer	Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.
I/O	Generated by an I/O controller, to signal normal completion of an operation or to signal a variety of error conditions.
Hardware failure	Generated by a failure, such as power failure or memory parity error.

- Improves processor utilization.
    - I/O devices are much slower than the processor.
- When an interrupt occurs user program is suspended, and the interrupt handler code is executed, once the interrupt handler code is finished then the execution of the suspended process resumes.
- The **user program does not have to contain any special code to accommodate interrupts**; **the processor and the OS are responsible** for suspending the user program, then resuming it at the same point.
- The interrupt-handler routine is generally part of the OS.
- To accommodate interrupts, an interrupt stage is added to the instruction cycle.

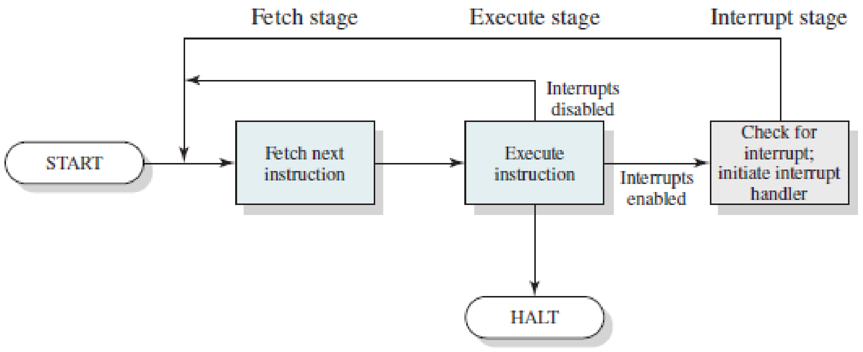


Figure 1.7 Instruction Cycle with Interrupts

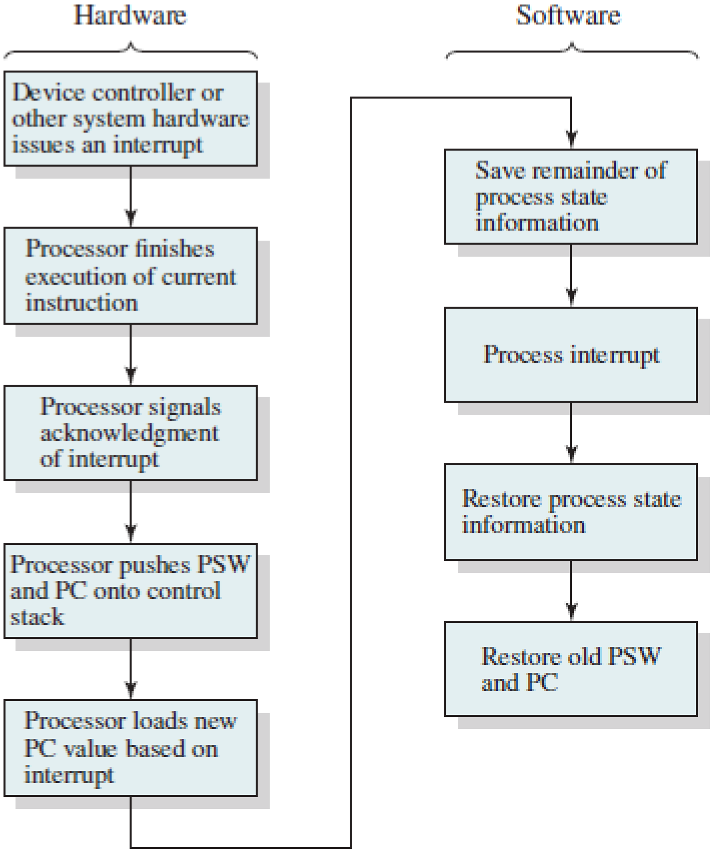

Hardware
Software
Device controller or other system hardware issues an interrupt
Processor finishes execution of current instruction
Processor signals acknowledgment of interrupt
Processor pushes PSW and PC onto control stack
Processor loads new PC value based on interrupt
Save remainder of process state information
Process interrupt
Restore process state information
Restore old PSW and PC

Figure 1.10 Simple Interrupt Processing

- The PSW contains status information about the currently running process, including memory usage information, condition codes, and other status information such as an interrupt enable/disable bit and a kernel/user-mode bit.

What is an OS?

- OS is the computing system that manages all the hardware and software.
- In the first part of this course, we will mainly focus on how OS works, the related concepts as well as algorithms.
- Some concepts related to distributed computing and network operating systems will be covered later in the course.
- Some concepts on computer architecture and hardware will be covered as well.

What is OS composed of?

1. Memory Manager
    a. Controls and manages the main memory.
2. Processor Manager
    a. Decides how to allocate processing power to processes.
3. Device Manager
    a. Manages every device, channel, and control unit.
4. File Manager
    a. Manges file system.
5. Network Manager
    a. Manages network connections.