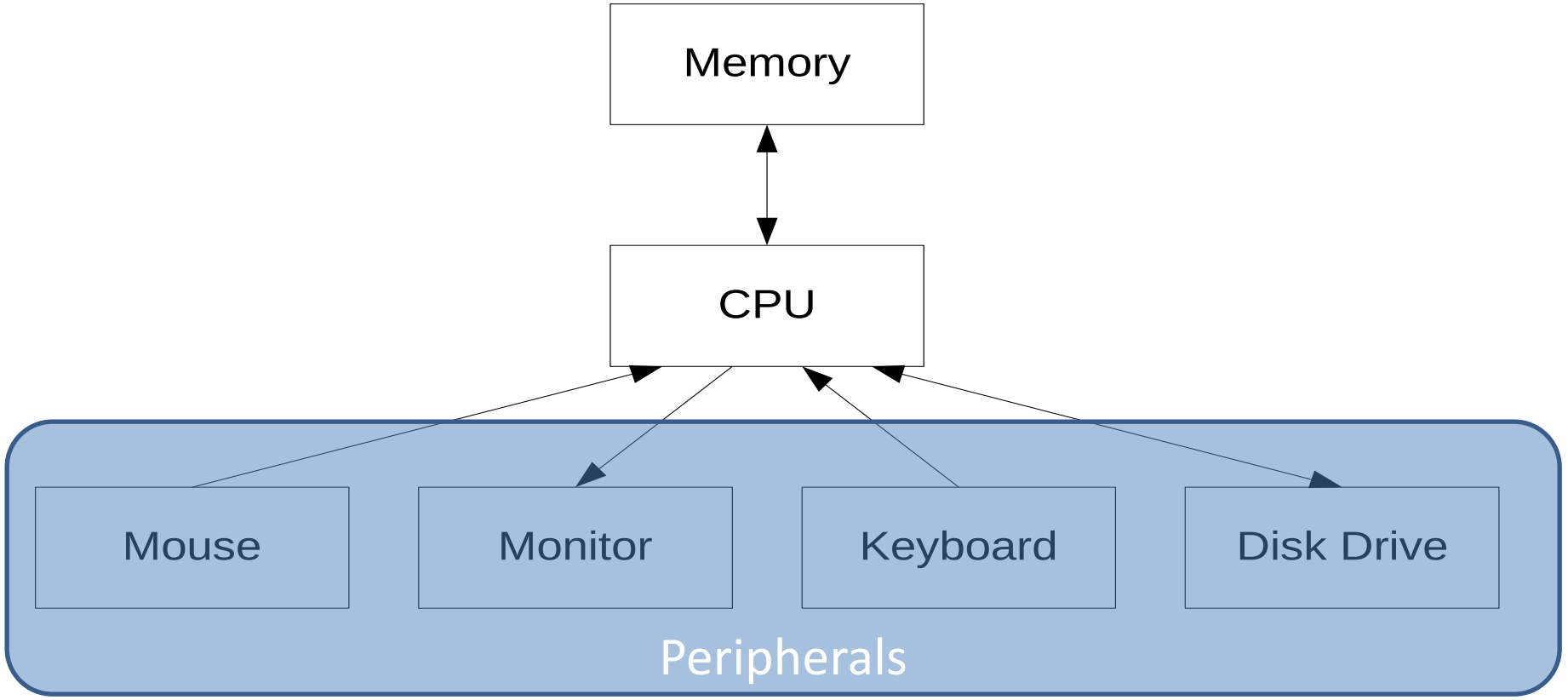
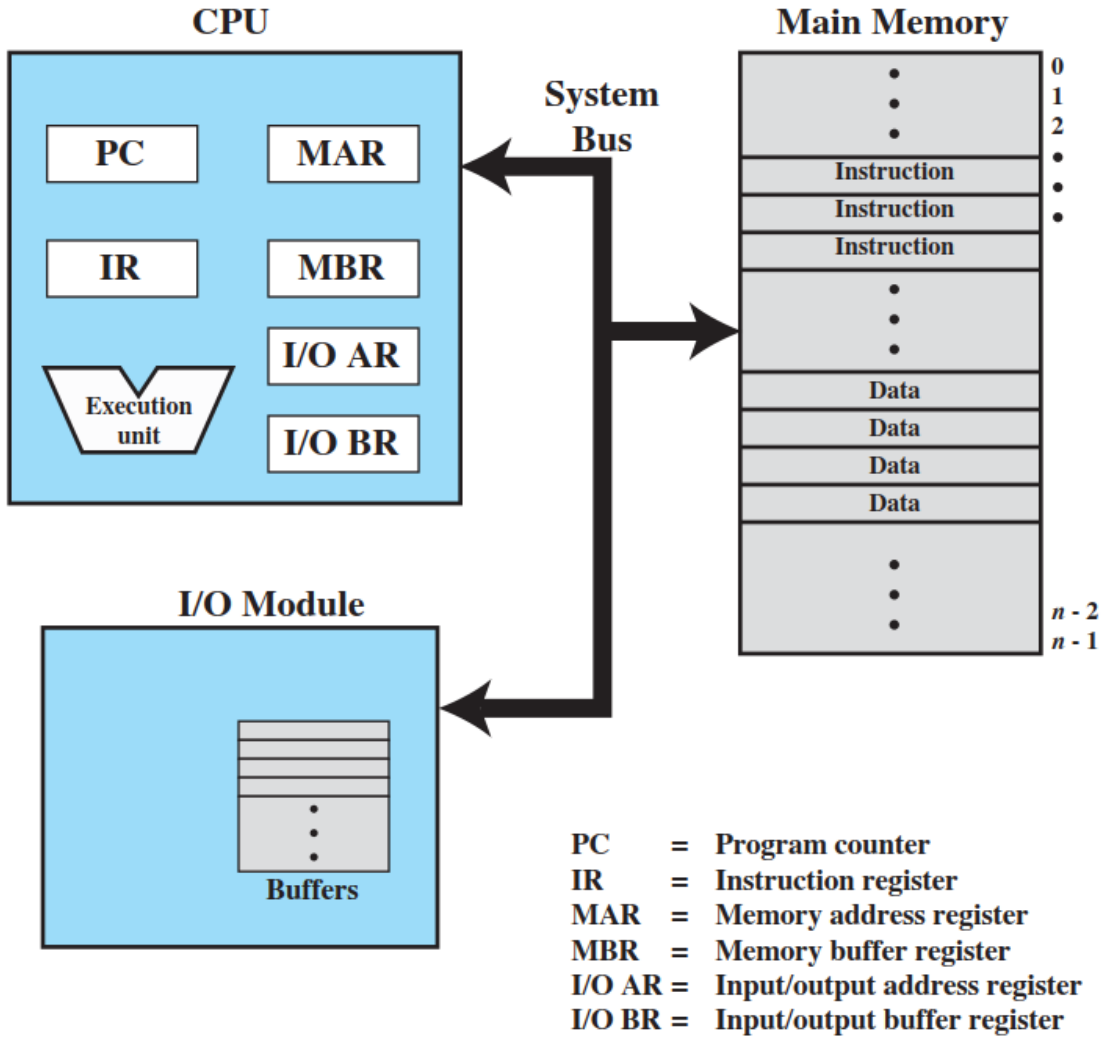
CS 3841 Operating Systems
An Introduction to Operating Systems

- Objectives
  - Identify the four main structural elements of a computer.
  - Draw a diagram showing the instruction execution cycle.
  - Draw the memory hierarchy for a computer.
  - Explain the concept of direct memory access.
  - Explain how an operating system serves as a user interface, resource manager, and supports evolution.
  - Explain how operating systems evolved from serial processing through batch scheduling to modern structures.
  - Explain the difference between user mode and kernel mode.

What is a computer?



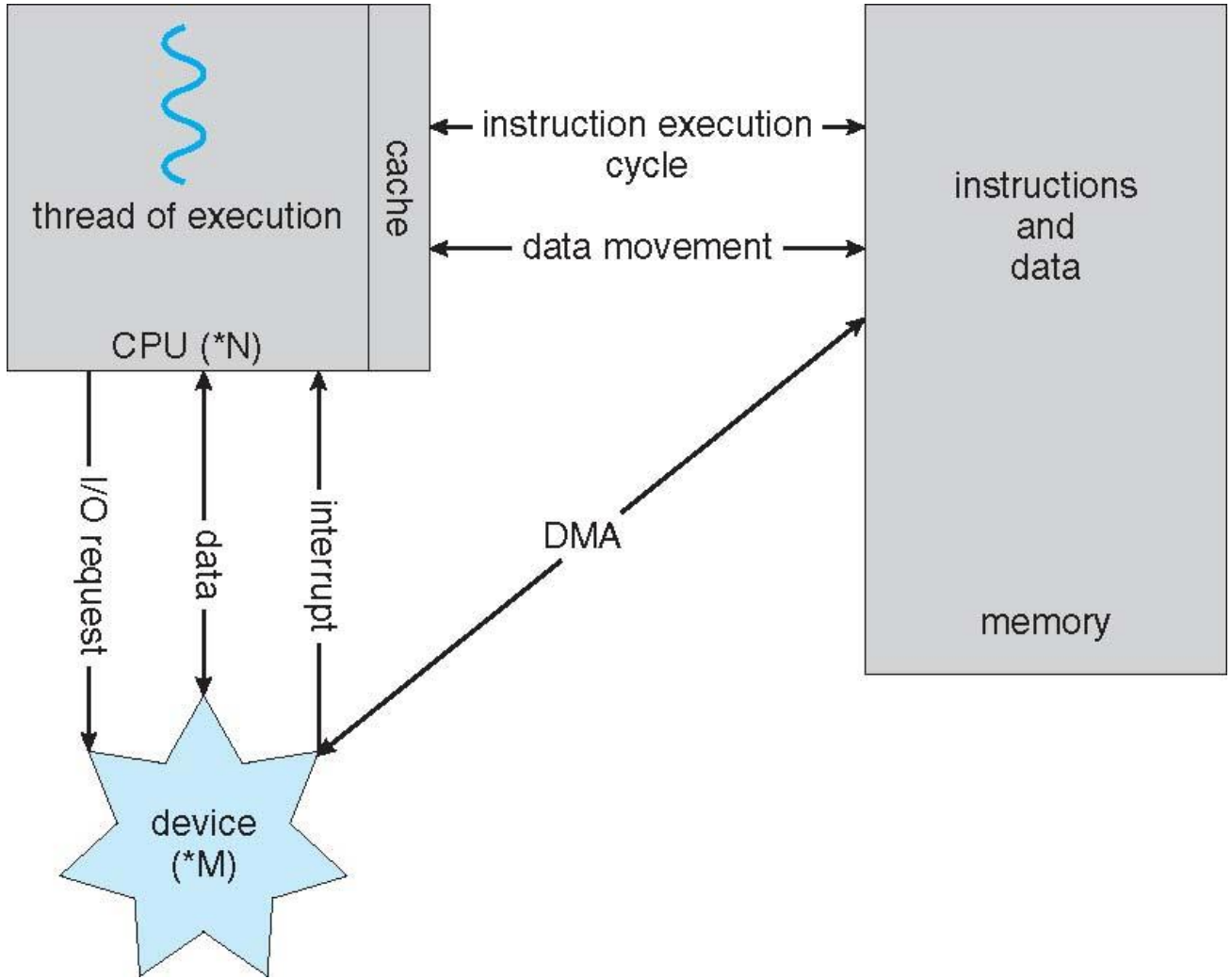
What is a computer?



Four main pieces to a computer

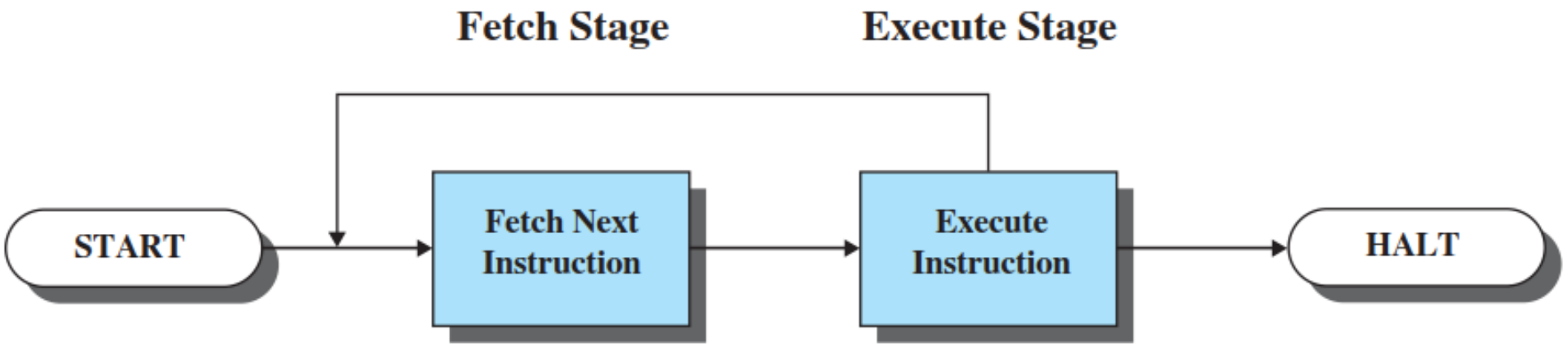
- Processor
  - Controls the operation of the computer
- Main Memory
  - Stores the data used by a computer and the programs that execute.
  - Typically losses all values when the computer is shut down
- I/O Modules
  - Moves data between the computer and the external environment
- System Bus
  - Allows for communication between processors, main memory, and IO Modules

How a Modern Computer Works



Abraham Silberschatz, Peter B. Galvin, and Greg Gagne. 2013. Operating System Concepts Essentials (2nd ed.). Wiley Publishing.

Basic Instruction Cycle



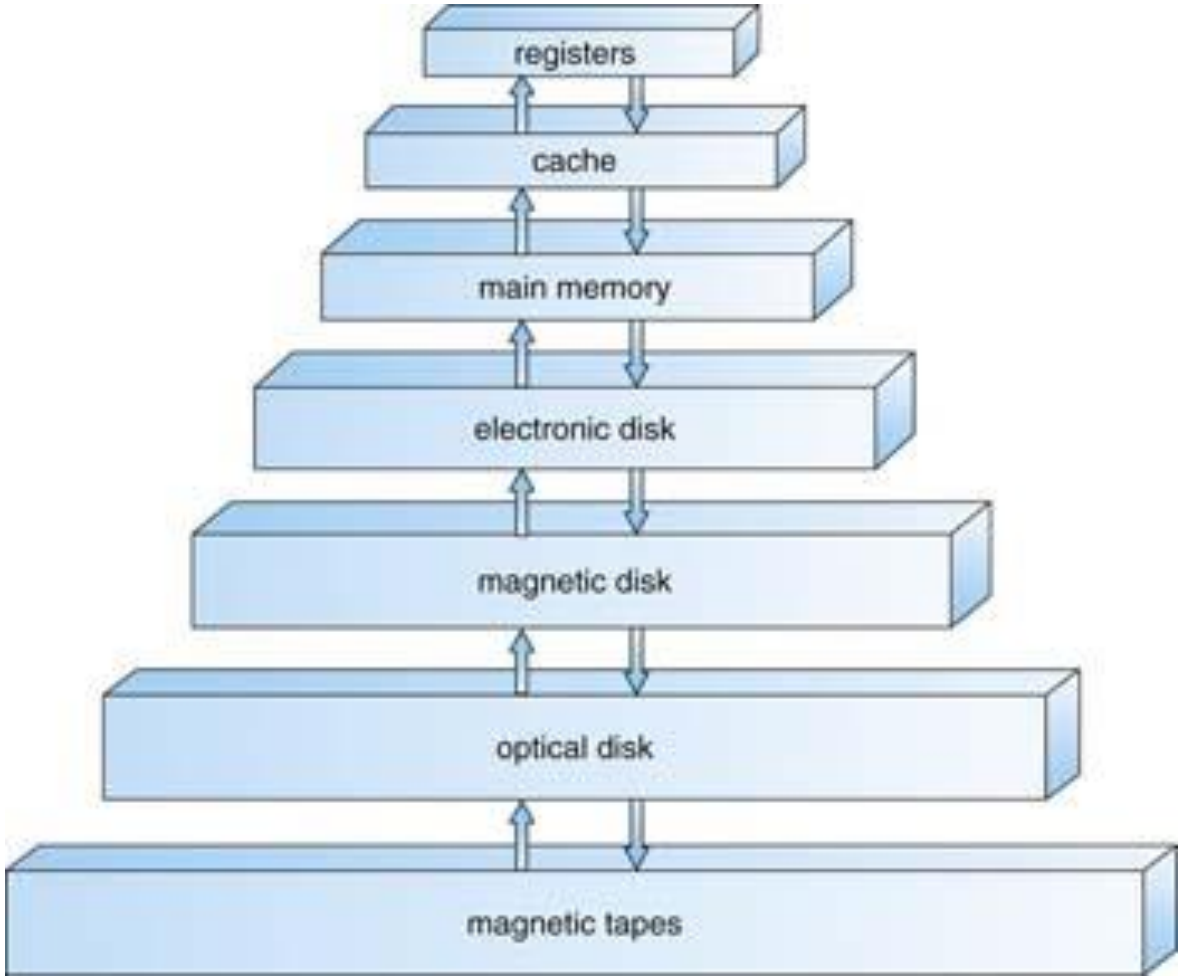
Interrupt

- An unscheduled procedure call or method invocation caused by a hardware signal or software flag being set

Direct Memory Access

- DMA is the hardware mechanism that allows peripheral components to transfer their I/O data directly to and from main memory without the need for the system processor to be involved in the transfer. Use of this mechanism can greatly increase throughput to and from a device, because a great deal of computational overhead is eliminated.

What is the memory hierarchy for a computer?



Abraham Silberschatz, Peter B. Galvin, and Greg Gagne. 2013. Operating System Concepts Essentials (2nd ed.). Wiley Publishing.

Operating System

- “OS is simply the software that controls your computer and tells it what to do. Your computer must have an OS installed before it can do anything useful or fun.”
  - Support.apple.com

Operating System

- “An operating system is a collection of system programs that control computer and any other peripherals connected to it. The program that hides the truth about the hardware from the programmer and present and a nice simple view a named file that can be read & written as “operating system”. Operating system shields the programmer from the interface, the abstraction offers by the operating system is slower & easier to use than the underlying hardware.”

- - Oscience.info

Computer Architectures

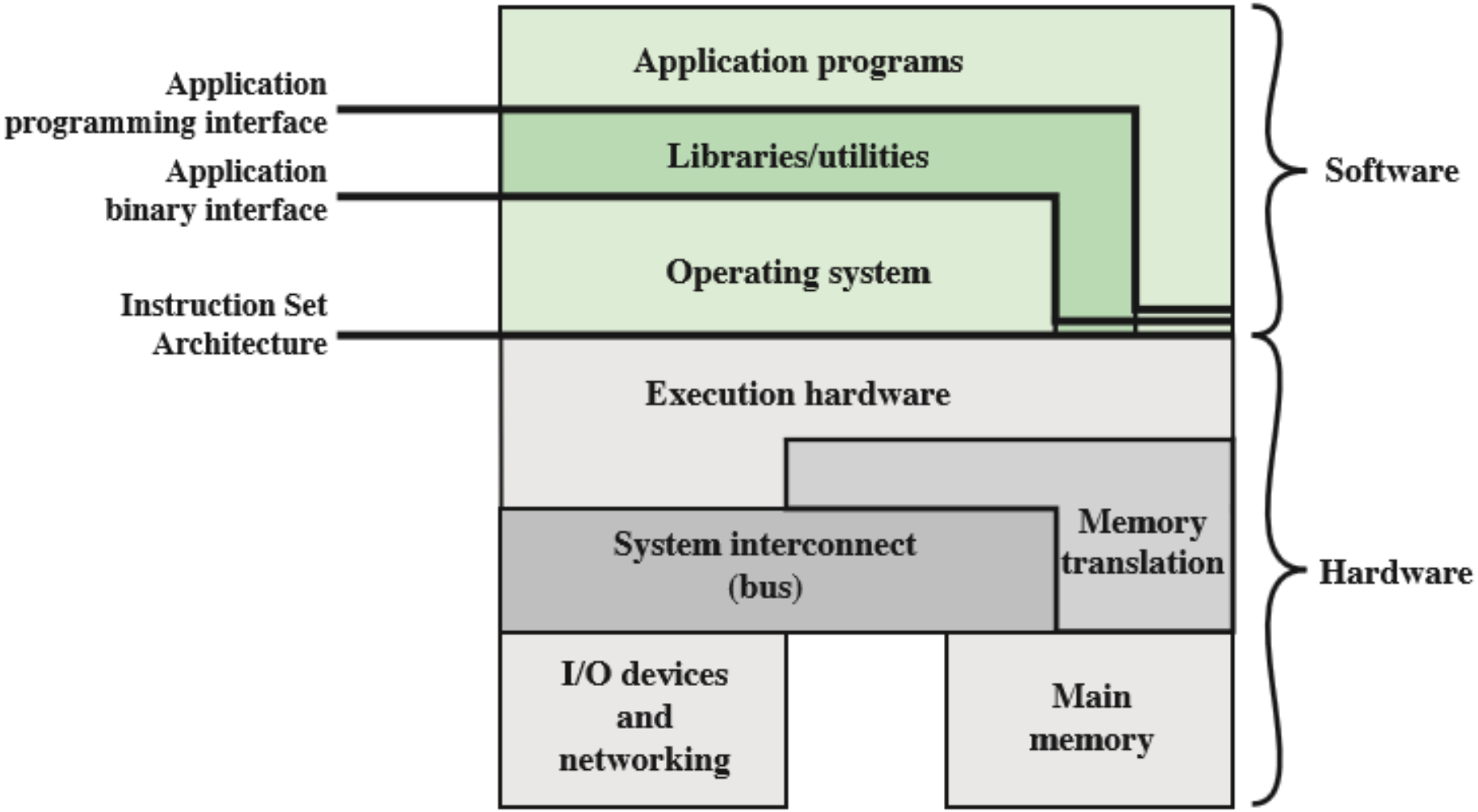
- How is ARM different from Intel?
- ARM and Intel represent different Instruction Set Architectures
  - We can write hello world in both of them
  - The assembly is very different
  - The behavior is the same
- How does printf work?

Why do we use an OS?

A piece of software that provides services to applications

- Advantages
  - Convenience
  - Efficiency
  - Security
  - Flexibility - Ability to evolve
- Disadvantage - Overhead

Hardware and Software Structures



What does the OS Provide

- Program Development
- Program Execution
- IO Device Access
- File Access
- System Access
- Error detection and handling
- Accounting/Bookkeeping
- Instruction Set Architecture Support
- Application Binary Interface
- Application Programming Interface

ENIAC
(Electronic Numerical Integrator and Computer)

- Completed 1945
- Primary job: Calculate artillery firing tables
- Study thermonuclear weapons
  - 357 multiplications/second
- No operating system
- Today 2020
  - Supercomputer Fugaku
  - RIKEN Center for Computational Science
  - 415,530,000,000,000,000 multiplications/second

Serial Processing

- Programs simply ran
  - Operator fed punched cards
  - Lights indicated completion
- Problems:
  - Scheduling
  - Setup time

Batch Systems

- IBM 701
  - Developed 1950's by GM
  - Software monitor used to control job
  - Jobs submitted and batched together and feeds entire job set into the computer

https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4640507

Early Operating System Discoveries

- Isolation and Protection
  - The monitor's memory area was not to be modified by programs
- Timer
  - Timer prevented single job from monopolizing system
- Privileged Instructions
  - Certain instructions can only be executed by the monitor
- Interrupts
  - Allows tasks to be taken over if needed

Multiprogramming Systems

- Problem with basic batch systems
  - Processor is often idle
    - When?

Multiprogramming

- Processes do two things:
- Read IO
- Perform Calculations

Multiprogramming

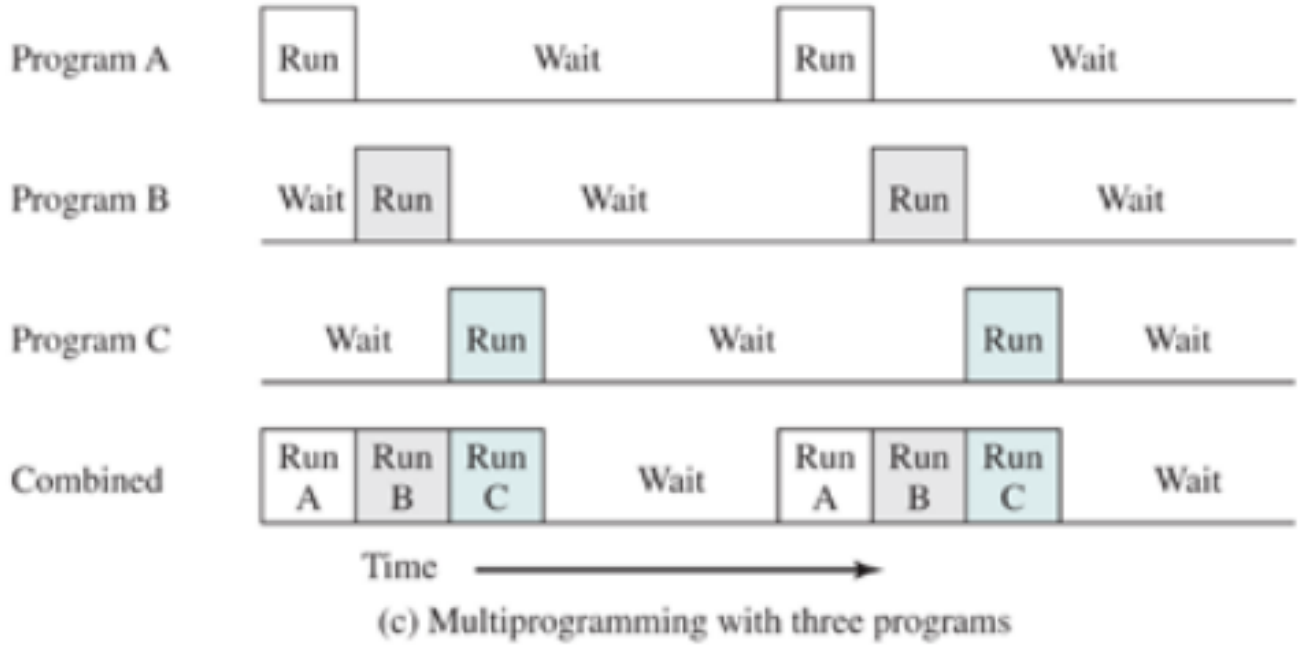


Figure 2.5 Multiprogramming Example

Effects of Multiprogramming

	Uniprogramming	Multiprogramming
Processor use	20%	40%
Memory use	33%	67%
Disk use	33%	67%
Printer use	33%	67%
Elapsed time	30 min	15 min
Throughput	6 jobs/hr	12 jobs/hr
Mean response time	18 min	10 min

Today

- Most OS's are multitasking
- Most OS's support multiple users
  - Exceptions: embedded OS's, mobile OS's, small OS's
- Most OS's support batching / scripting
- Most OS's allow some sort of console operation

Operating System vs Kernel

- Operating System
  - A piece of software that provides services to applications
- Kernel
  - A piece of software that “bridges” hardware and software
  - Figurative sense of "core or central part of anything“ (https://www.etymonline.com/word/kernel)
- Questions
  - Is a kernel an operating system?
  - Is there more to an operating system than just the kernel?
  - Can an operating system have more than one kernel?
  - Does the kernel run on its own?
  - How do we create a kernel?

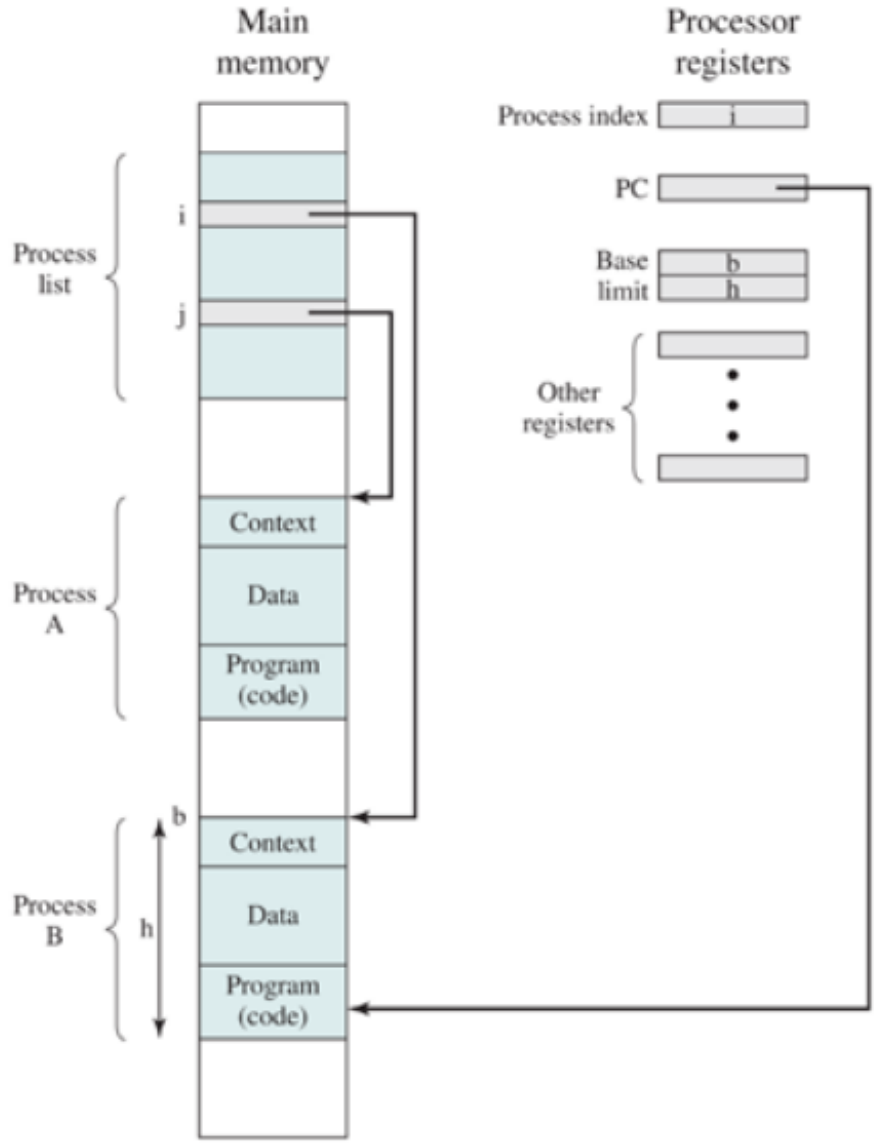


Figure 2.8 Typical Process Implementation

Program vs Process

- Program
  - Static representation of operations and data
  - Compiled code
- Process
  - Instance of active exeuction

Digging Deeper

- OS Structure
- System Calls