Computer Organization

B.Tech.II CSE (Sem-3)

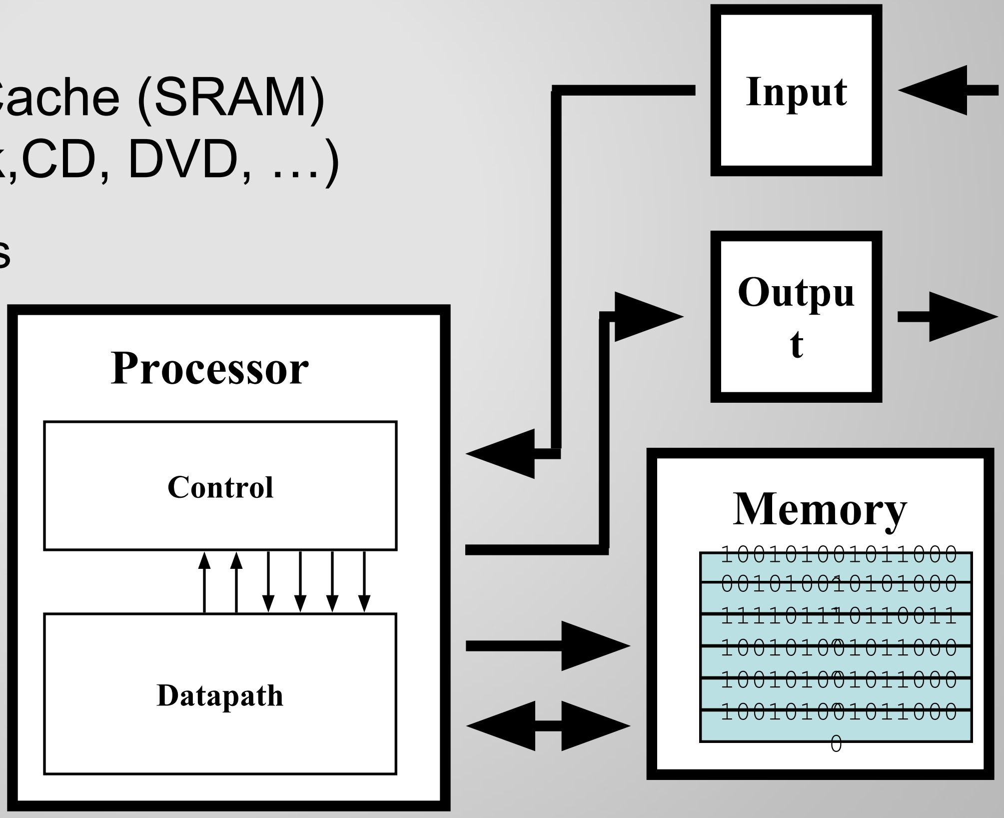
Topics

- Components of Computer
- Structure & Function of Computer
- The Computer Level Hierarchy
- The Von Neumann Model
- Moore’s Law
- Terminology

Components of a Computer

1. Input (mouse, keyboard, …)
2. Output (display, printer, …)
3. Memory
   - Main (DRAM), Cache (SRAM)
   - Secondary (Disk,CD, DVD, …)
4. Datapath
5. Control

Processor or (CPU)



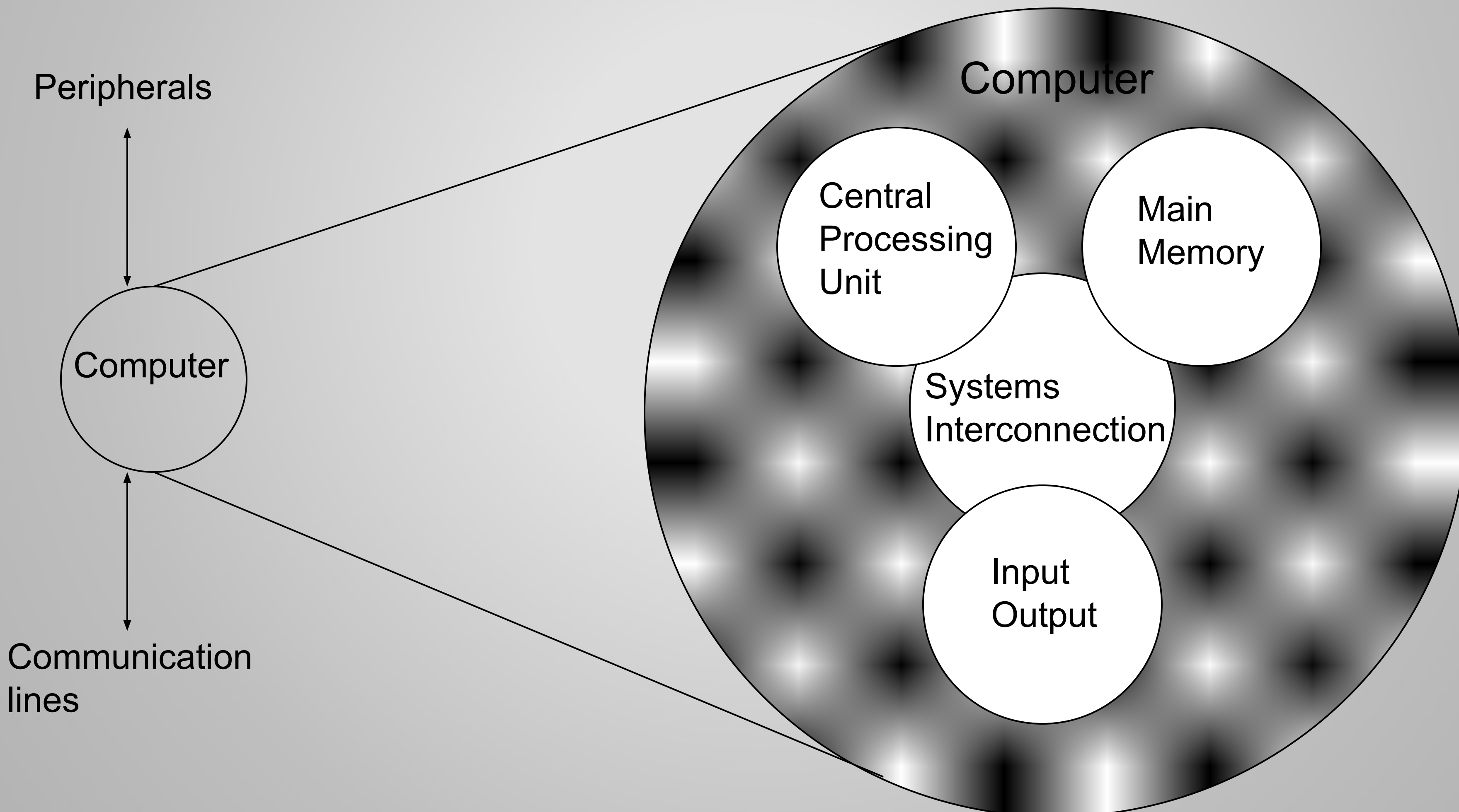
Structure & Function of Computer

- At the most basic level, a computer is a device consisting of :
  - A processor to interpret and execute programs
  - A memory to store both data and programs
  - A mechanism for transferring data to and from the outside world (Input/Output)

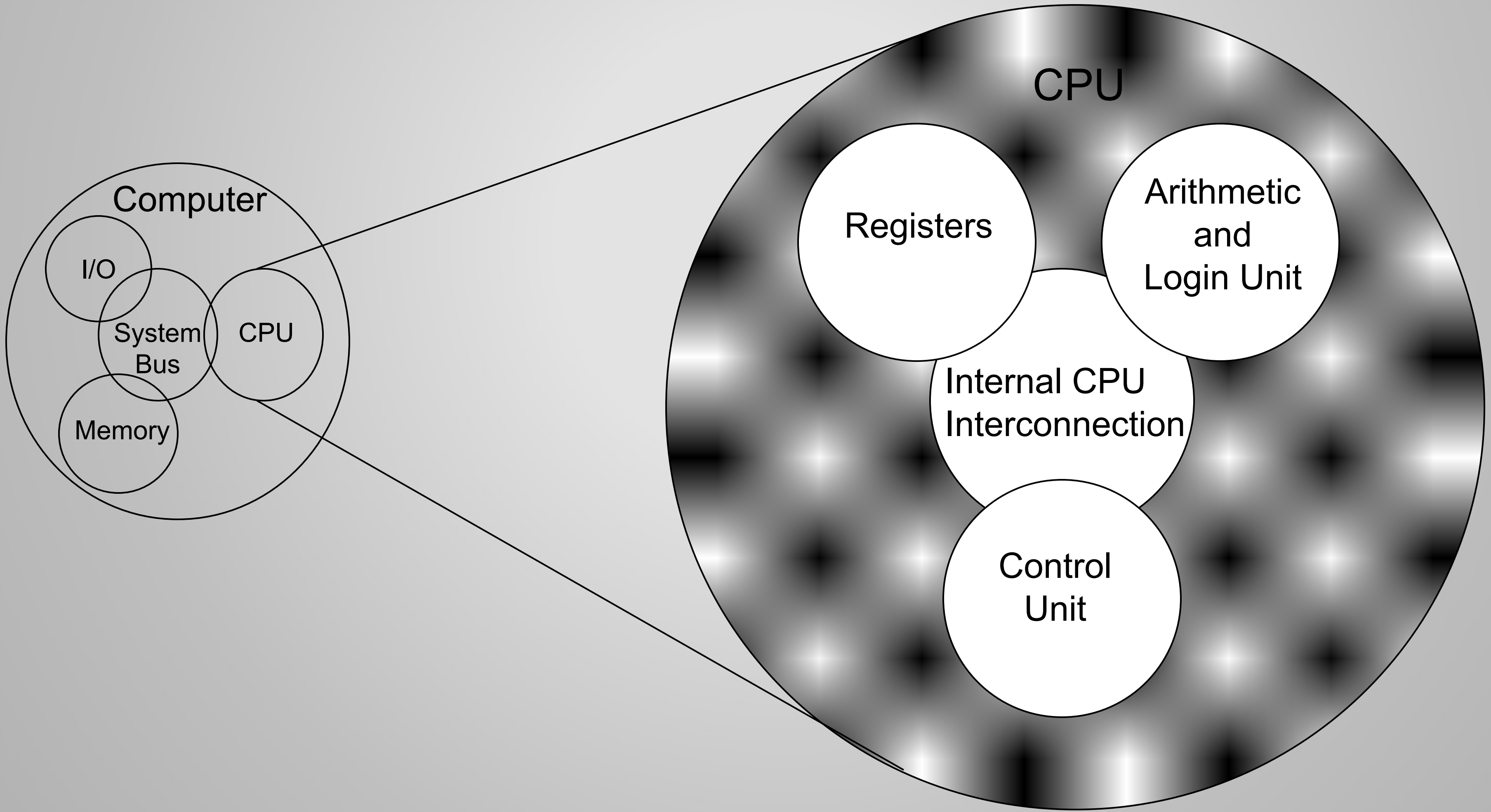
- Structure is the way in which components relate to each other

- Function is the operation of individual components as part of the structure

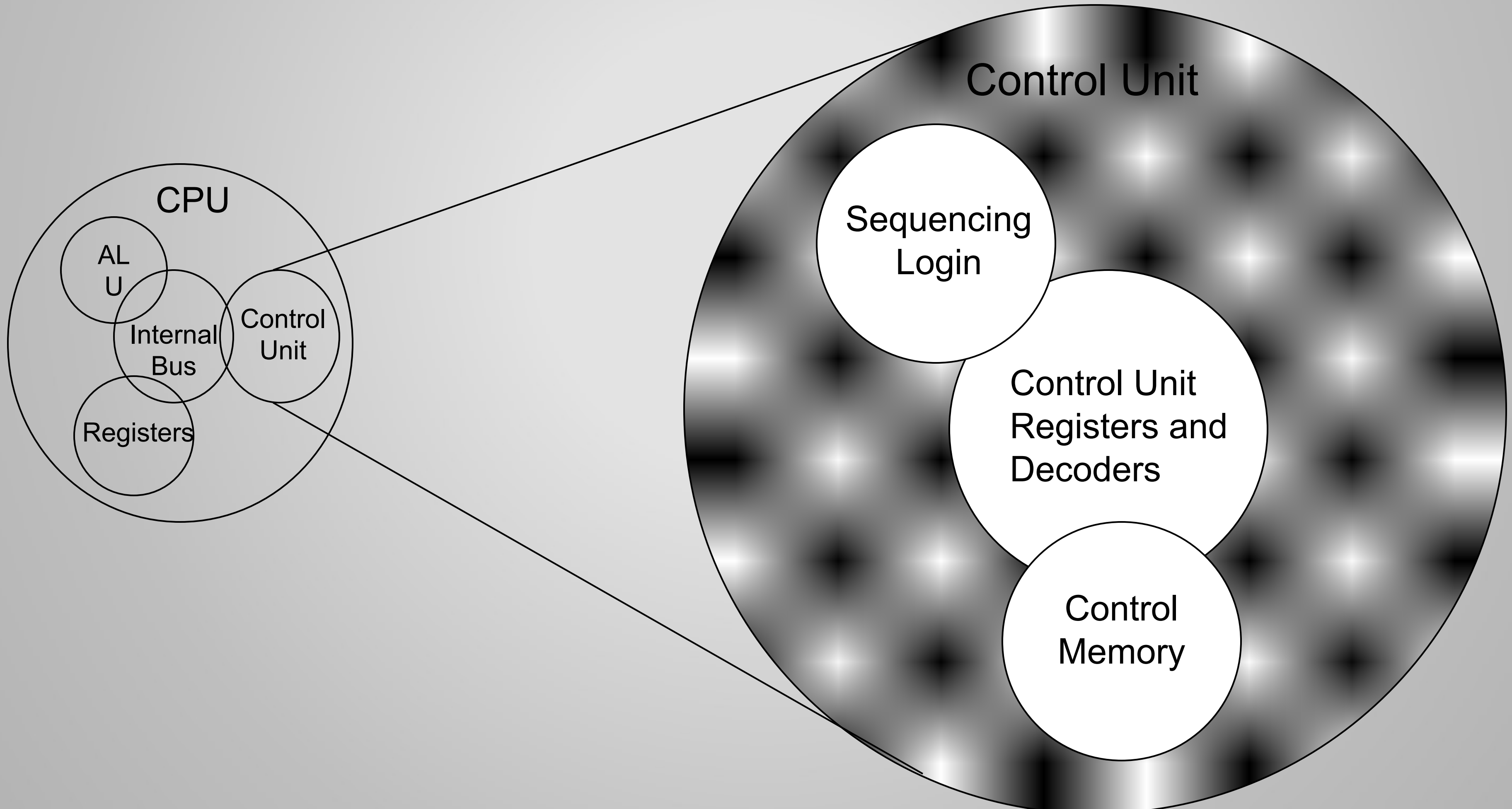
Structure - Top Level



Structure - The CPU



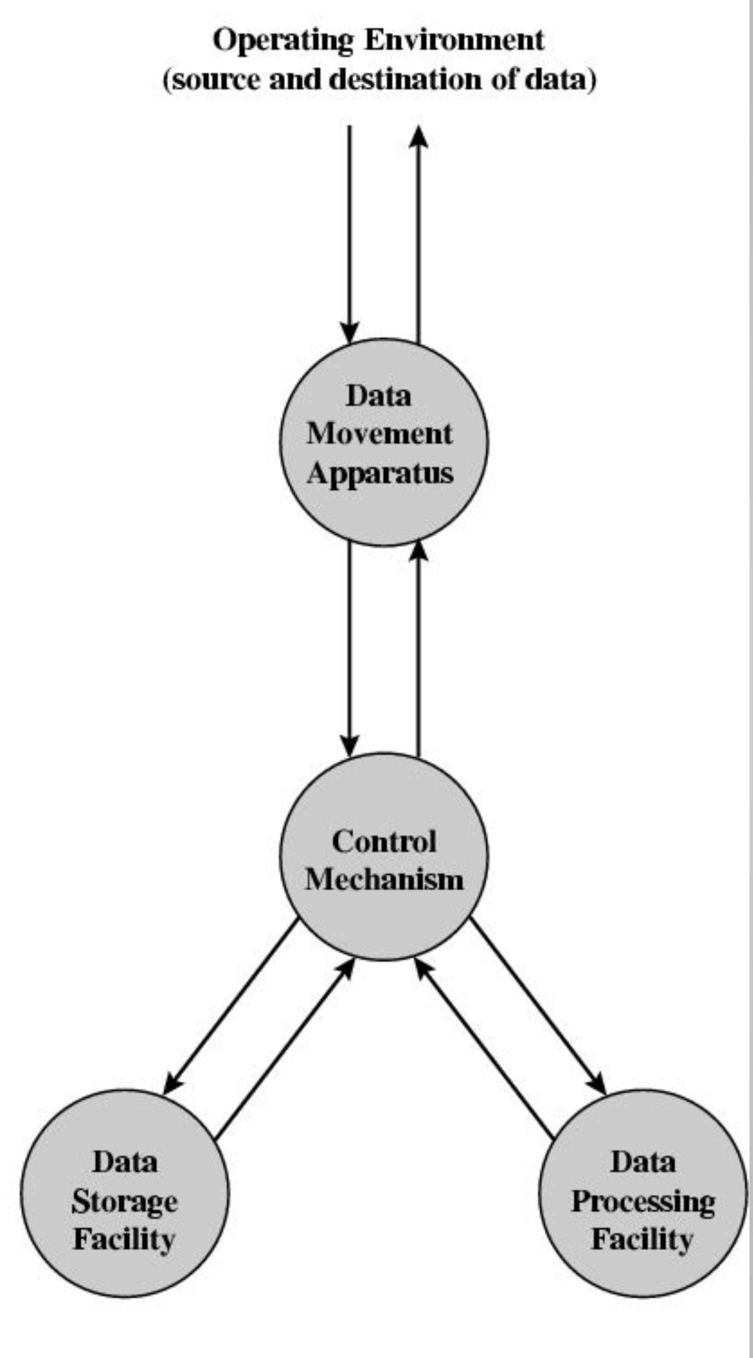
Structure - The Control Unit



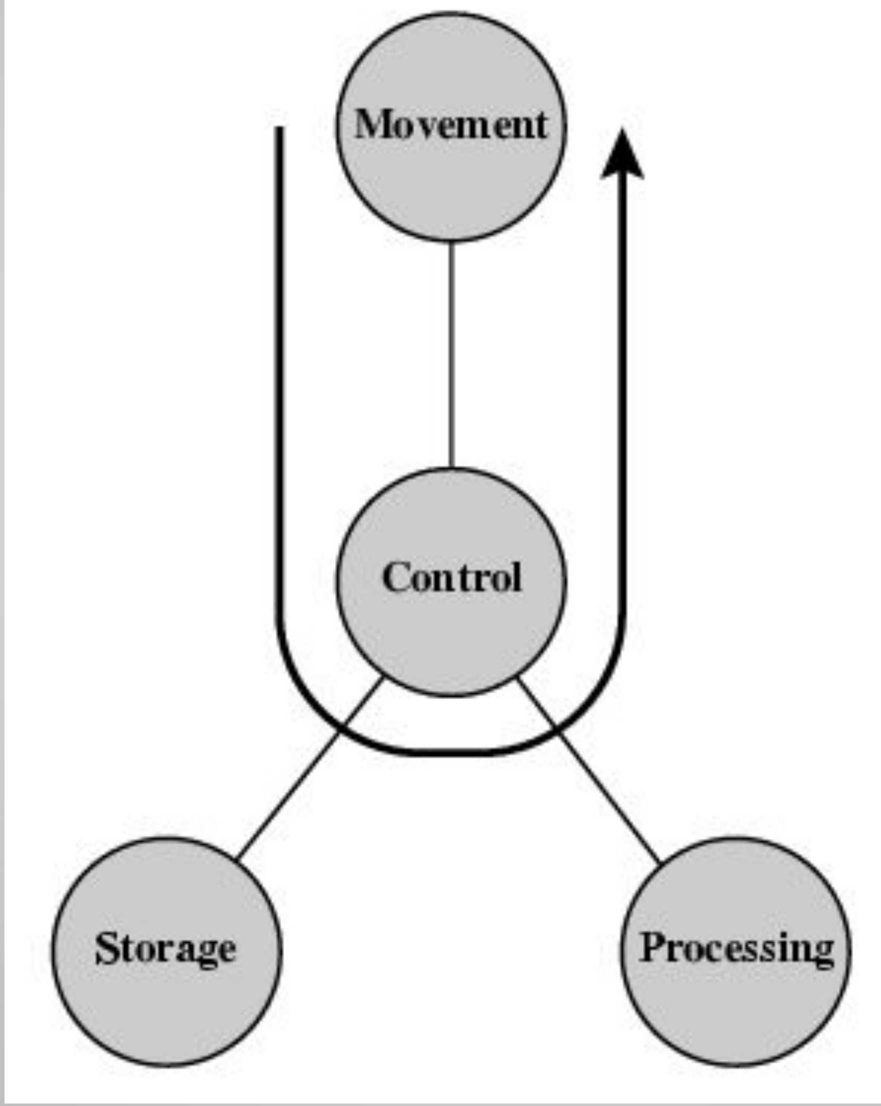
Computer Functions

- Data processing
- Data storage
- Data movement
- Control

Functional view

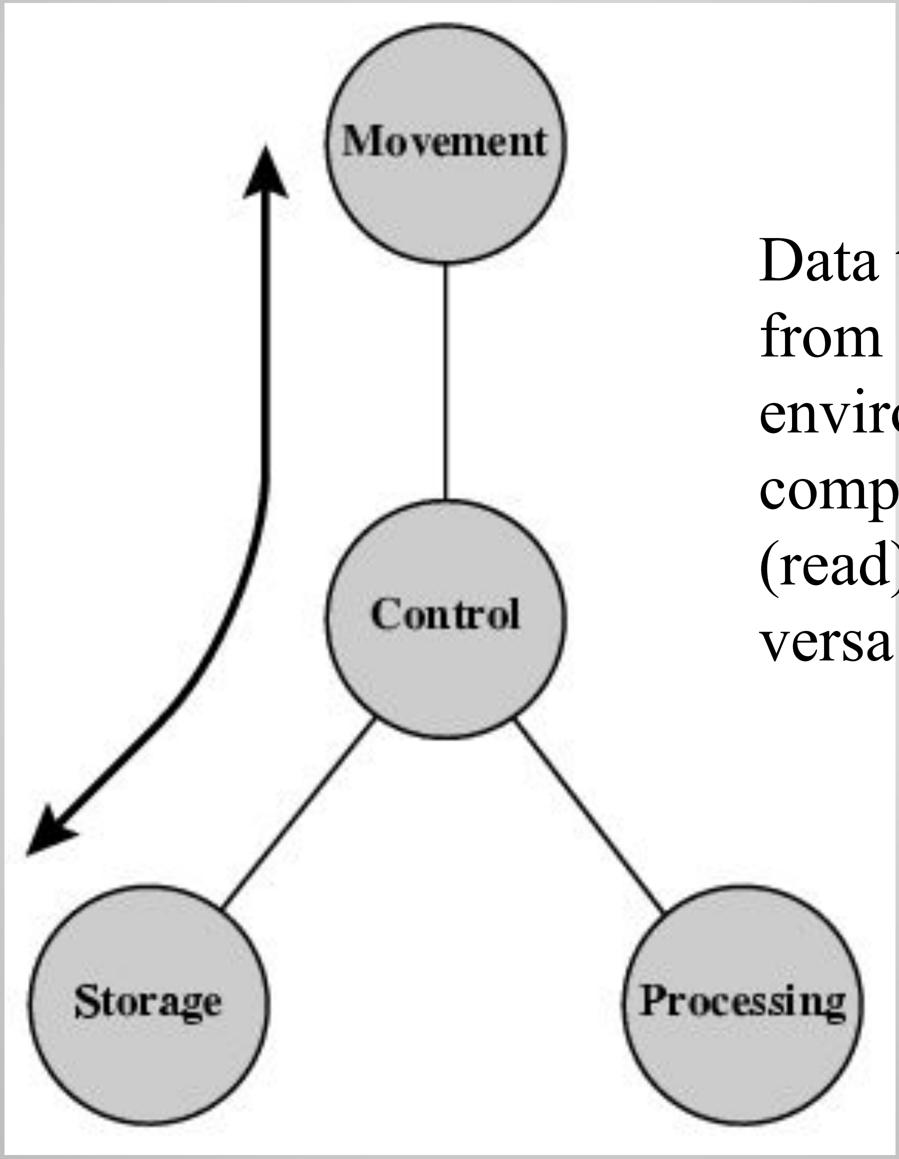


Operations (1) Data movement



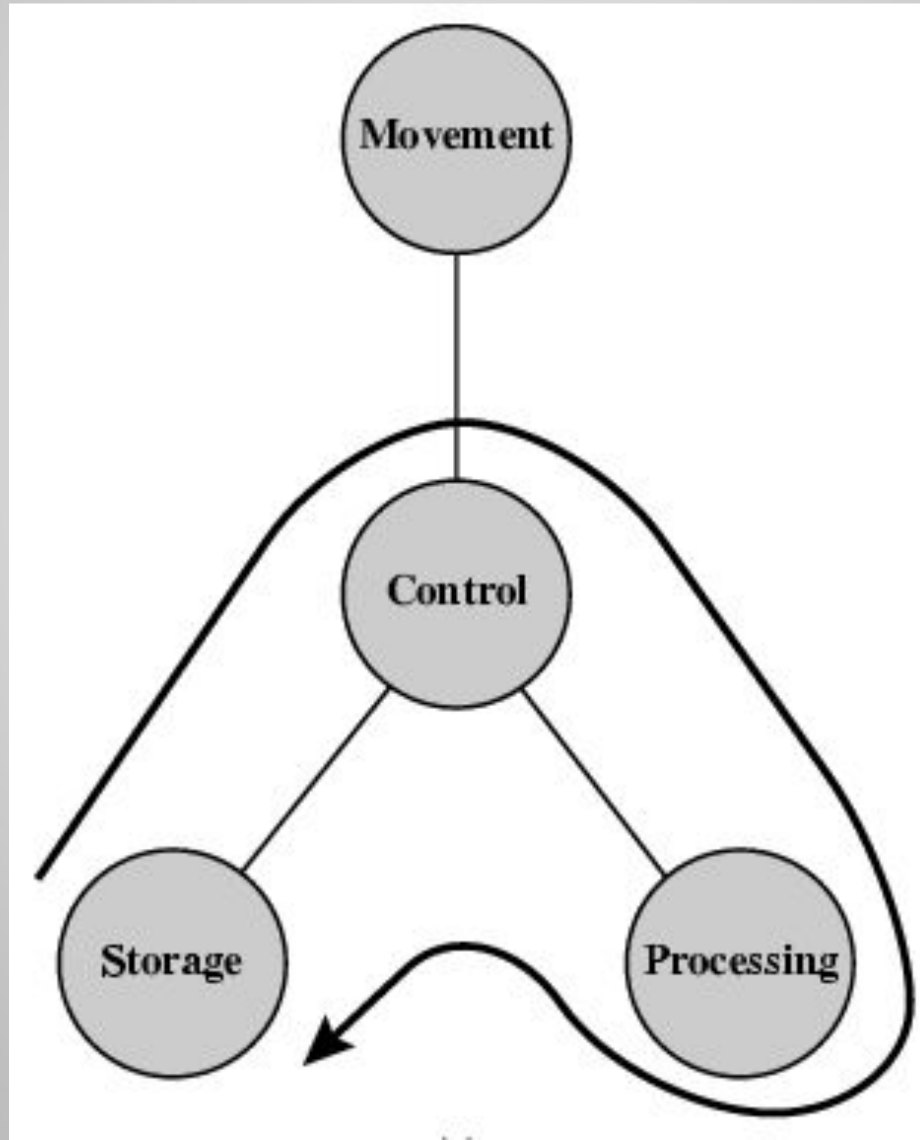
Simply transferring data from one peripheral or communications line to another.

Operations (2) Storage

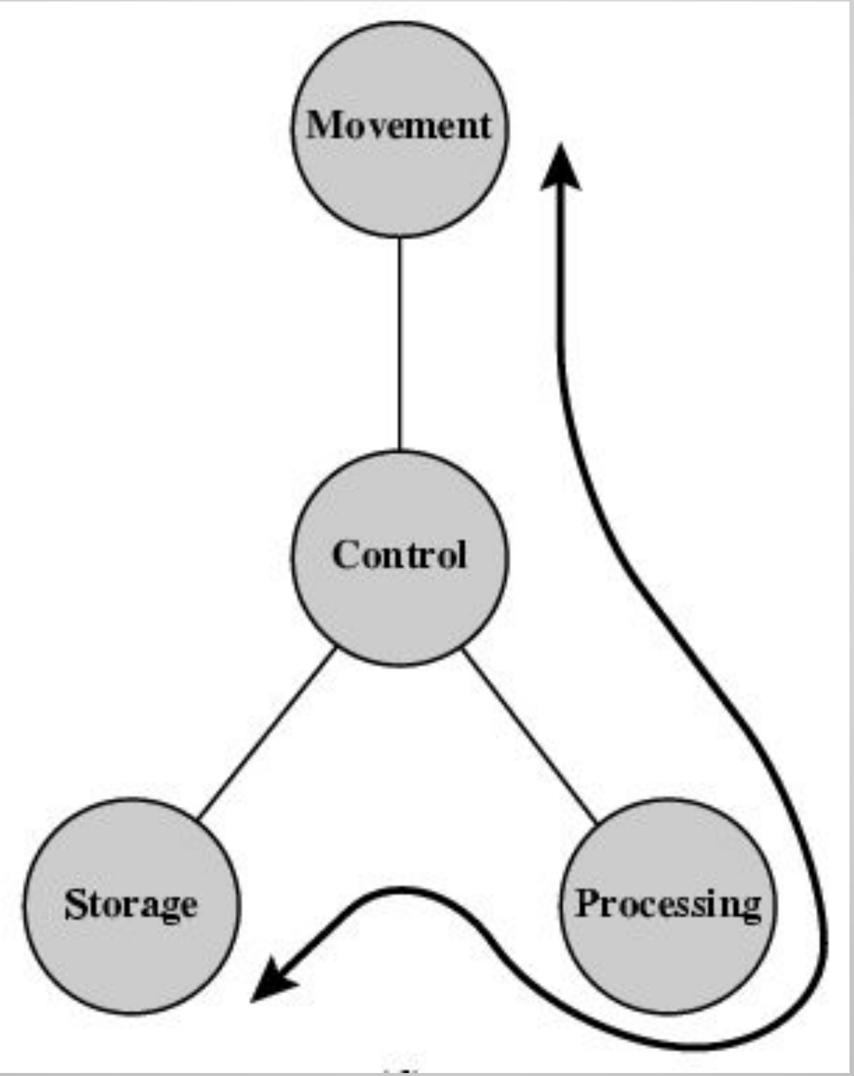


Data transferred from the external environment to computer storage (read) and vice versa (write).

Operation (3) Processing from/to storage



Operation (4)
Processing from storage to I/O



The Computer Level Hierarchy

- Writing complex programs requires
  - A “divide and conquer” approach, where each program module solves a smaller problem

- Complex computer systems employ a similar technique through a series of virtual machine layers

Six Level Computer

LEVEL 5 — **Problem Oriented Language Level (High Language Level)**

↓ Translation(Compiler)

LEVEL 4 — **Assembly Language Level**

↓ Translation(Assembler)

LEVEL 3 — **Operating System Machine Level**

↓ Partial Interpretation (Operating System)

LEVEL 2 — **Instruction Set Architecture Level**

↓ Interpretation (Microprogram) / Direct Execution (Hardwired)

LEVEL 1 — **Micro Architecture Level**

↓ Hardware

LEVEL 0 — **Digital Logic Level**

The Computer Level Hierarchy

- **Level 5: Problem Oriented (High-Level) Language Level**
  - Program execution and user interface level
  - The level with which we are most familiar
  - The level with which we interact when we write programs in languages such as C, Pascal, Lisp, and Java
- **Level 4: Assembly Language Level**
  - Acts upon assembly language produced from Level 5, as well as instructions programmed directly at this level
- **Level 3: Operating System Machine (System Software) Level**
  - Controls executing processes on the system
  - Protects system resources
  - Assembly language instructions often pass through Level 3 without modification

The Computer Level Hierarchy

- **Level 2: Instruction Set Architecture (ISA) Level**
  - Also known as the Machine Level
  - Consists of instructions that are particular to the architecture of the machine
  - Programs written in machine language need no compilers, interpreters, or assemblers

The Computer Level Hierarchy

- **Level 1: Control Level**
  - A *control unit* decodes and executes instructions and moves data through the system
  - A collection of 8 to 32 registers form a local memory and a circuit called an ALU capable of performing simple arithmetic operations are present
  - The registers are connected to the ALU to form a data path over which data flow
  - On some machines the operation of the data path is controlled by a program called a microprogram – *microprogrammed CU*
    - A microprogram is a program written in a low-level language that is implemented by the hardware
  - On other machines the data path is controlled directly by hardware – *hardwired CU*
    - Hardwired control units consist of hardware that directly executes machine instructions

The Computer Level Hierarchy

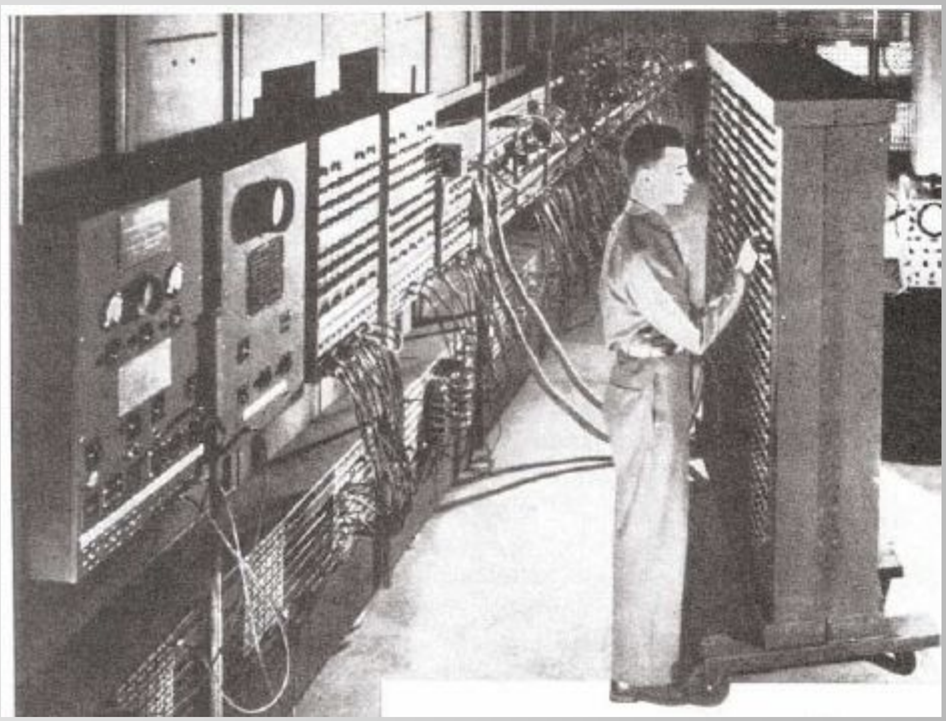
- **Level 0: Digital Logic Level**
  - Where find digital circuits (the chips)
  - Digital circuits consist of gates and wires
    - Each gate has one or more digital inputs and computes some simple function of the inputs such as AND or OR
  - These components implement the mathematical logic of all other levels
  - A small number of gates can be combined to form a 1-bit memory, 1-bit memories can be combined to form 16, 32, or 64 bit registers which can hold a single binary number

Basic Computer Model

- The von Neumann Model

Why The Von Neumann Model ?

- On the ENIAC, all programming was done at the digital logic level
- Programming the computer involved moving plugs and wires
- Configuring the ENIAC to solve a “simple” problem required many days labor by skilled technicians
- A different hardware configuration was needed to solve every unique problem type



The von Neumann Model

- The invention of stored program computers has been ascribed by a mathematician, John von Neumann

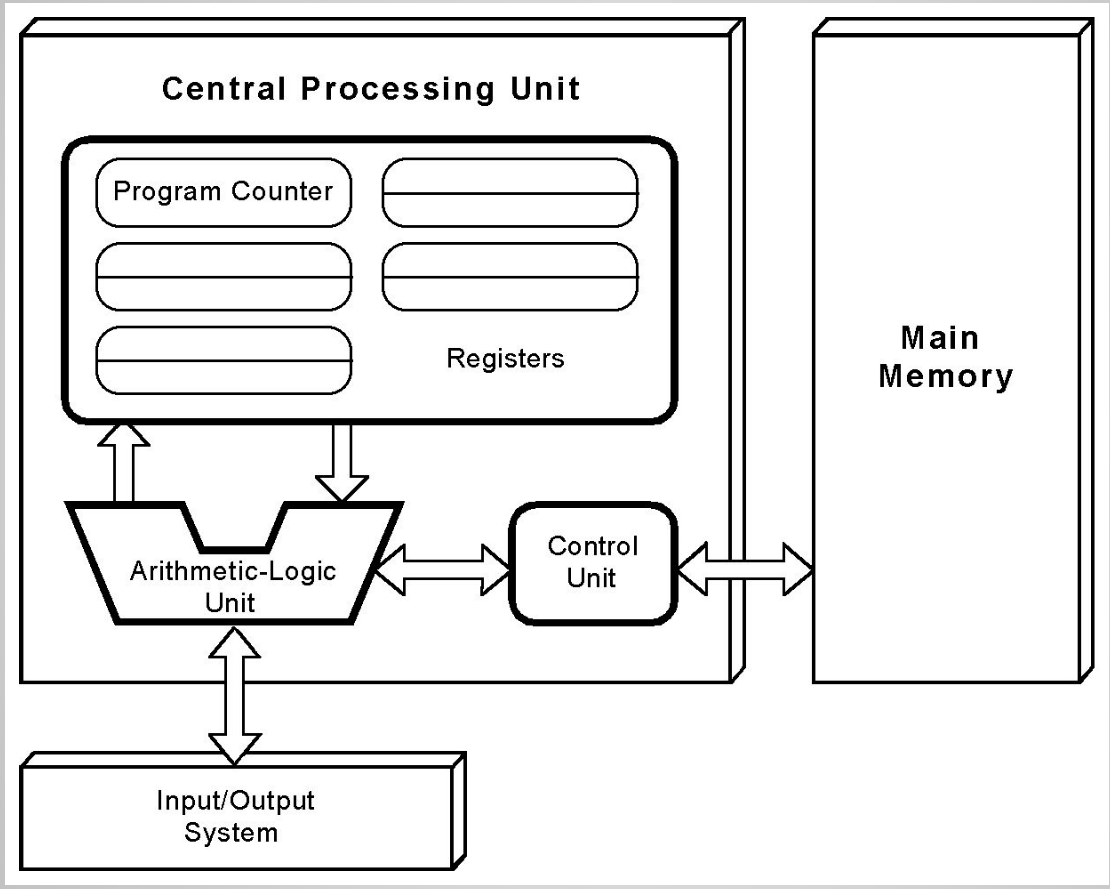
- Stored-program computers have known as von Neumann Architecture systems

The von Neumann Model

Stored-program computers have the following characteristics:

- Three hardware systems:
  - A central processing unit (CPU)
  - A main memory system
  - An I/O system
- Provides the capacity to carry out sequential instruction processing

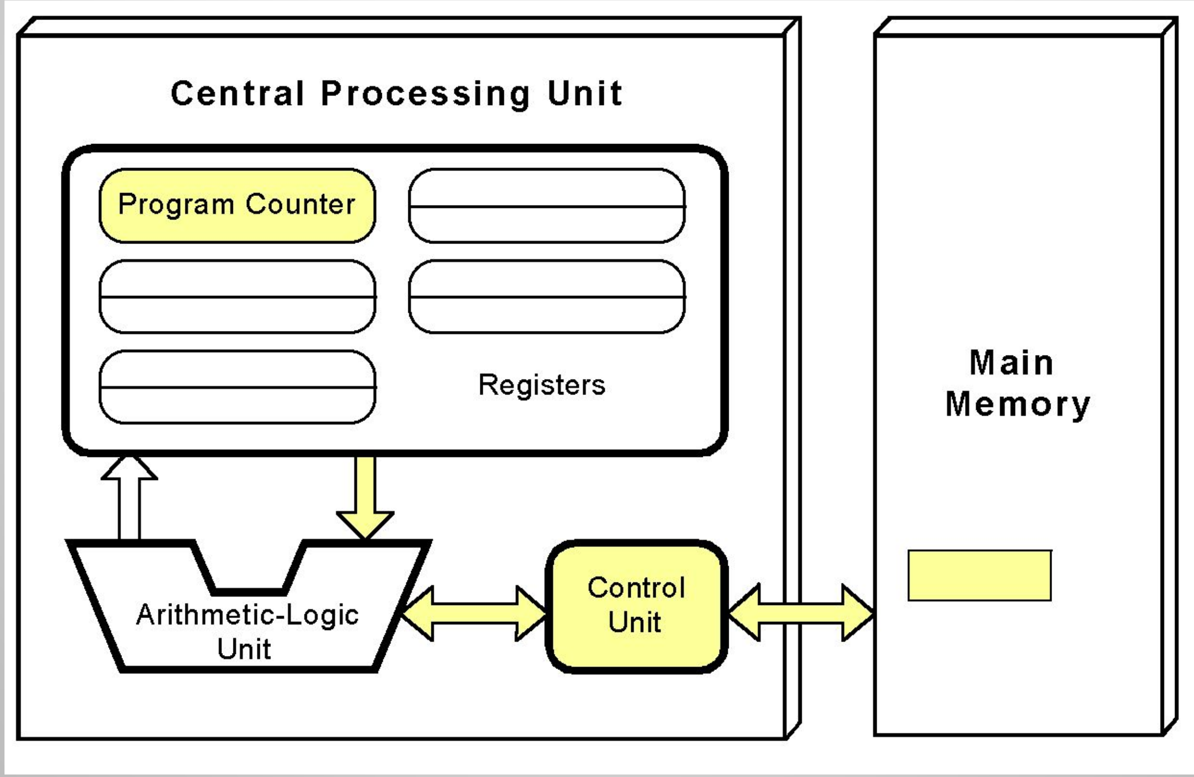
The von Neumann Model



Computers employ a <u>fetch-decode-execute</u> cycle to run programs

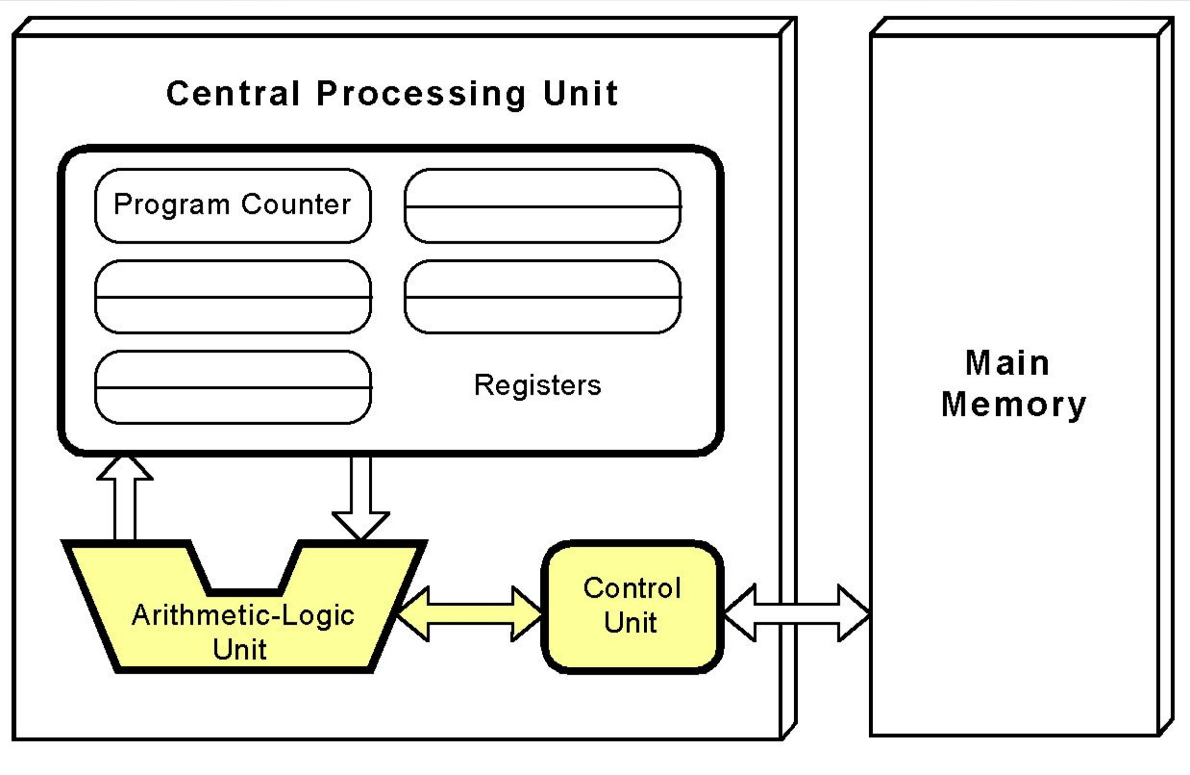
The von Neumann Model

- The control unit fetches the next instruction from memory using the program counter to determine where the instruction is located



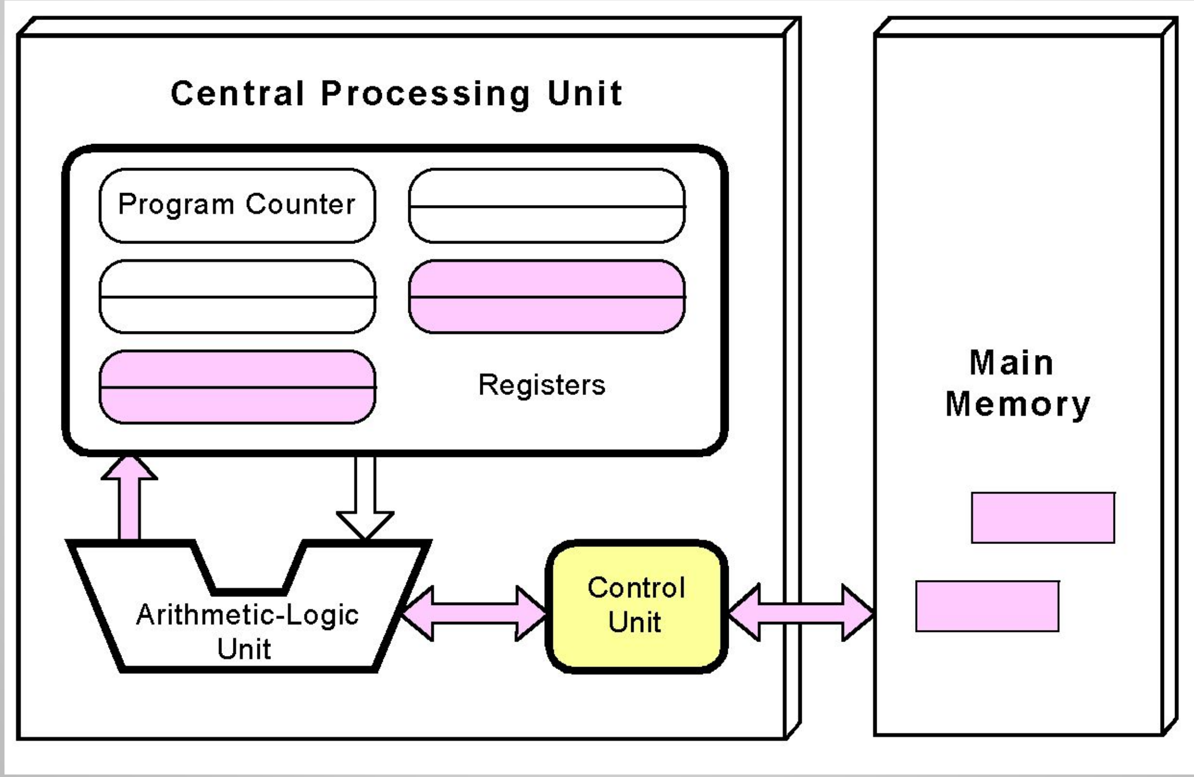
The von Neumann Model

- The instruction is decoded into a language that the ALU can understand



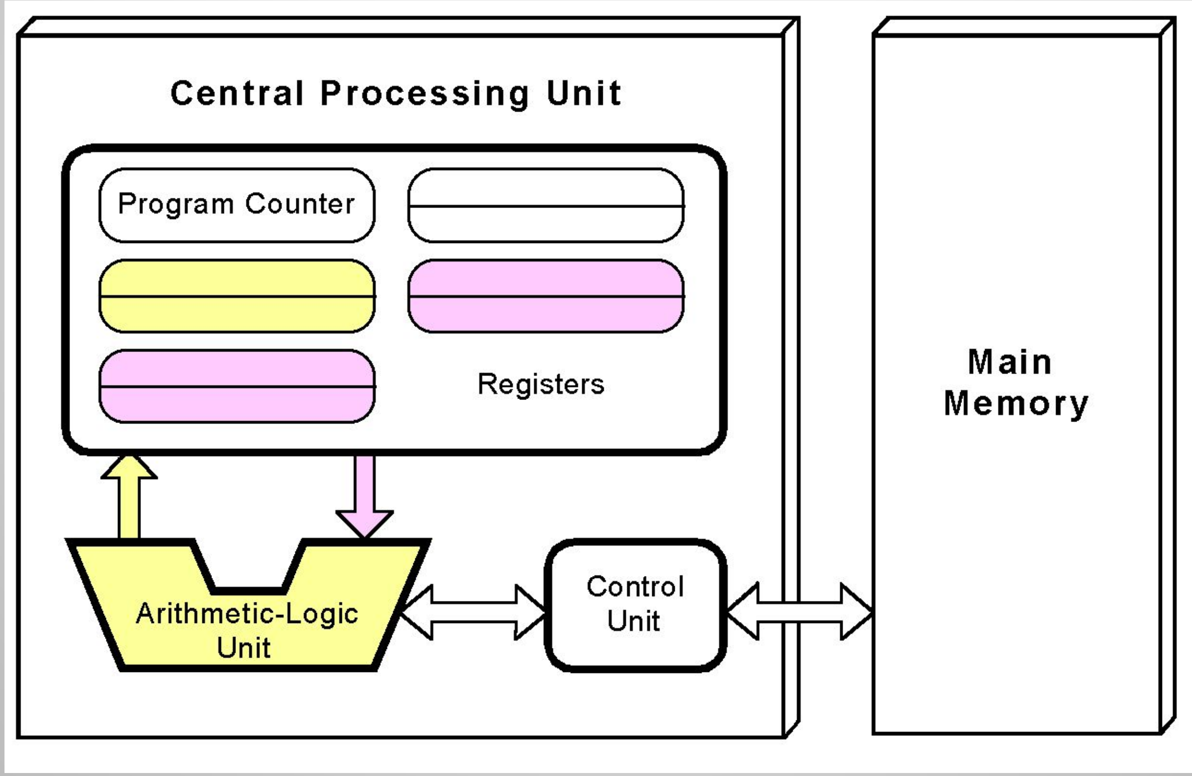
The von Neumann Model

- Any data operands required to execute the instruction are fetched from memory and placed into registers within the CPU



The von Neumann Model

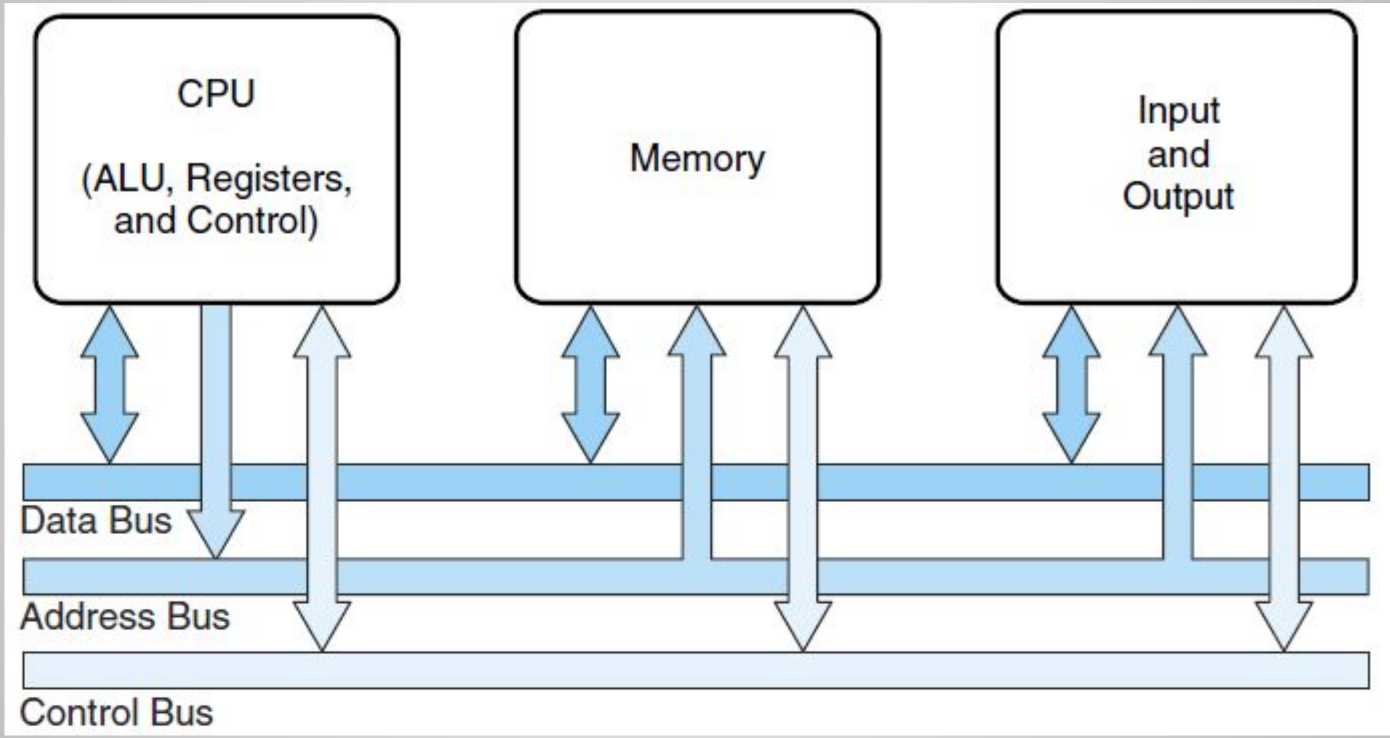
- The ALU executes the instruction and places results in registers or memory



The von Neumann Model

- Used the system bus
  - Data bus moves data from main memory to the CPU registers (and vice versa)
  - Address bus holds the address of the data that the data bus is currently accessing
  - Control bus carries the necessary control signals that specify how the information transfer is to take place

The von Neumann Model



The von Neumann Model

- A single data path between the CPU and main memory
  - This single path is known as the von Neumann bottleneck

Today's-von Neumann Models

- With many incremental improvements over the years:
  - Adding specialized buses
    - Programs and data stored in a slow-to-access storage medium, such as a HD, can be copied to a fast-access, volatile storage medium such as RAM prior to execution.
      - This architecture called the *system bus model*.
        - » The data bus moves data from main memory to the CPU registers (and vice versa).
        - » The address bus holds the address of the data that the data bus is currently accessing.
        - » The control bus carries the necessary control signals that specify how the information transfer is to take place.
  - Use of index registers addressing
  - Floating-point units
  - Cache memories
  - Use of interrupts
  - Asynchronous I/O
  - Concept of Virtual Memory
  - …

von Neumann Models

- Sufficient?
  - No, Computational power require departure from the classic von Neumann architecture

Non-von Neumann Models

- Adding processors is one approach
  - In the late 1960s, high-performance computer systems were equipped with dual processors to increase computational throughput
- In the 1970s supercomputer systems were introduced with 32 processors
- Supercomputers with 1,000 processors were built in the 1980s
- In 1999, IBM announced its Blue Gene system containing over 1 million processors

Non-von Neumann Models

- Parallel processing is only one method of providing increased computational power

- DNA computers, quantum computers and dataflow systems
  - May be the basis for the next generation of computers

Moore’s Law

- Rapidly changing field:
  - Vacuum tube -> Transistor -> IC -> VLSI
  - Memory capacity
  - Processor speed (due to advances in technology and hardware organization)

Moore’s Law

- In 1965 by Gordon Moore, Intel founder

  “The density of transistors in an integrated circuit will double every year.”

- Contemporary version:

  “The density of silicon chips doubles every 18 months.”

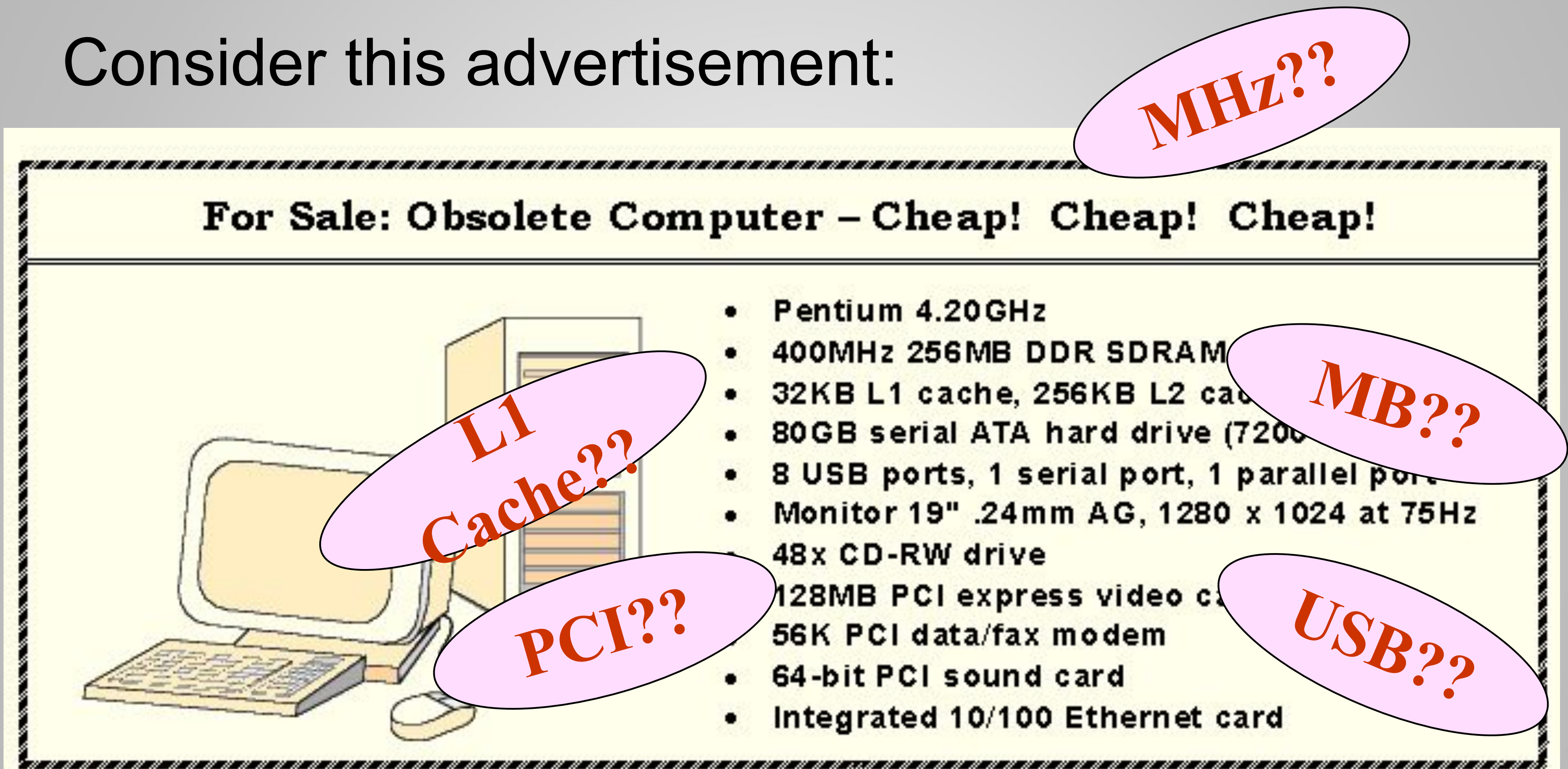
Next

ISA

Terminology

An Example System

Consider this advertisement:



Check the latest one… and understand what does it all mean??