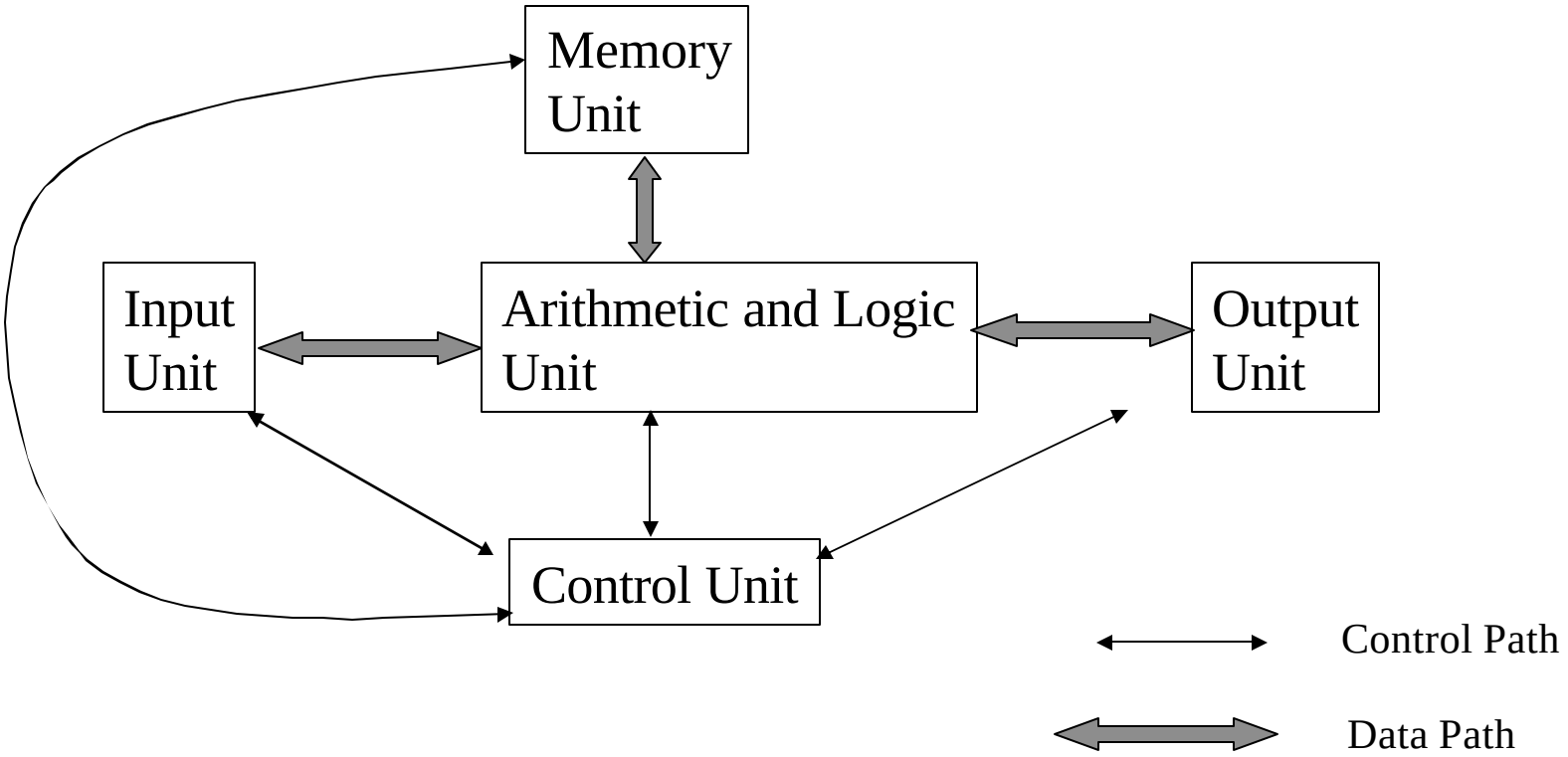
Programmer's Model

Basic Computer Organization

- The **Von Neumann** Model –The mother of conventional digital computers



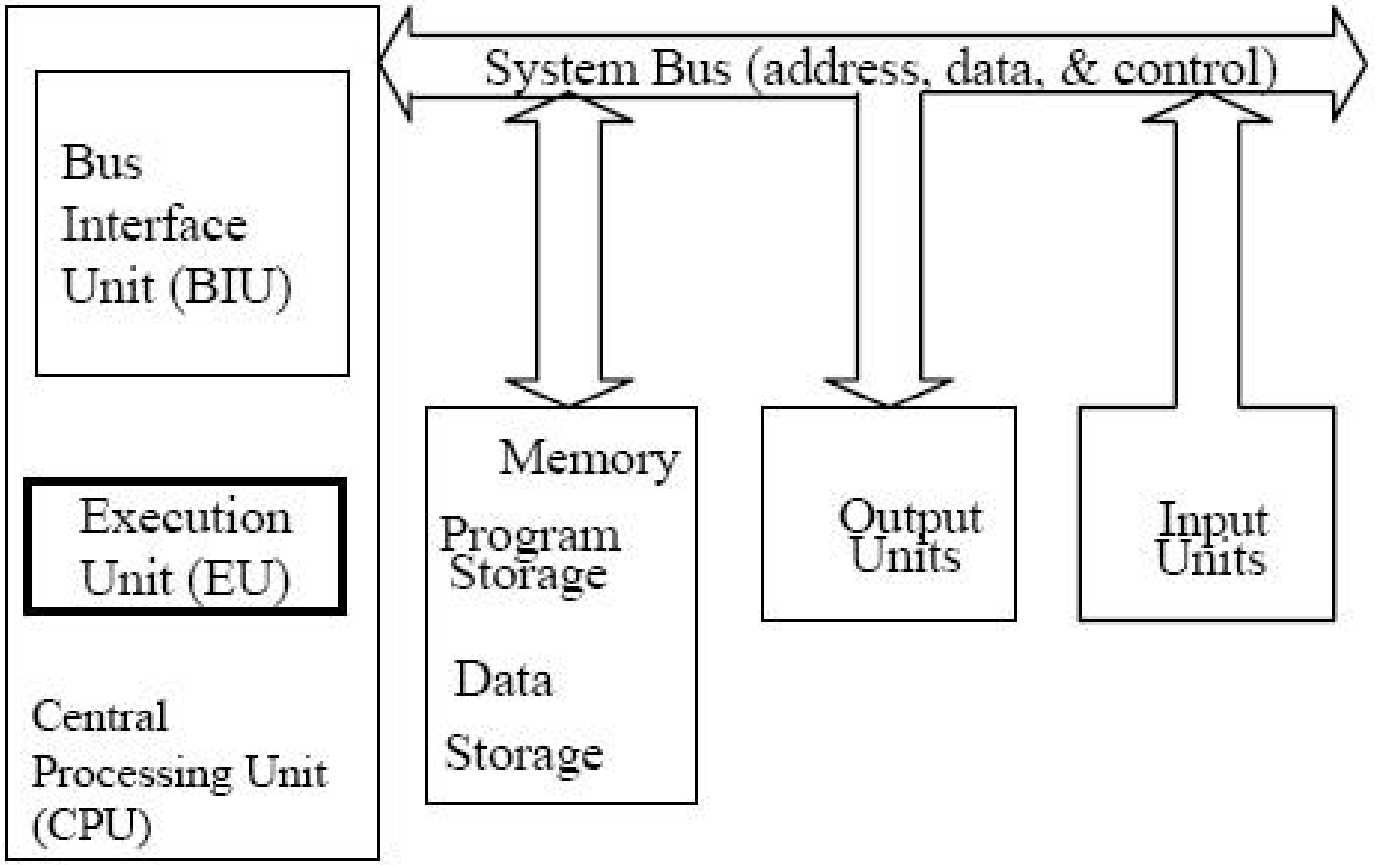
Basic Computer Organization

- Von Neumann defined the **Stored Program Concept**
  - (Before: program was stored externally)
  - Memory contains both **instructions and data**
    - Instructions ARE data: can be manipulated
  - Instructions must be read and then executed (**instruction execution cycle**)

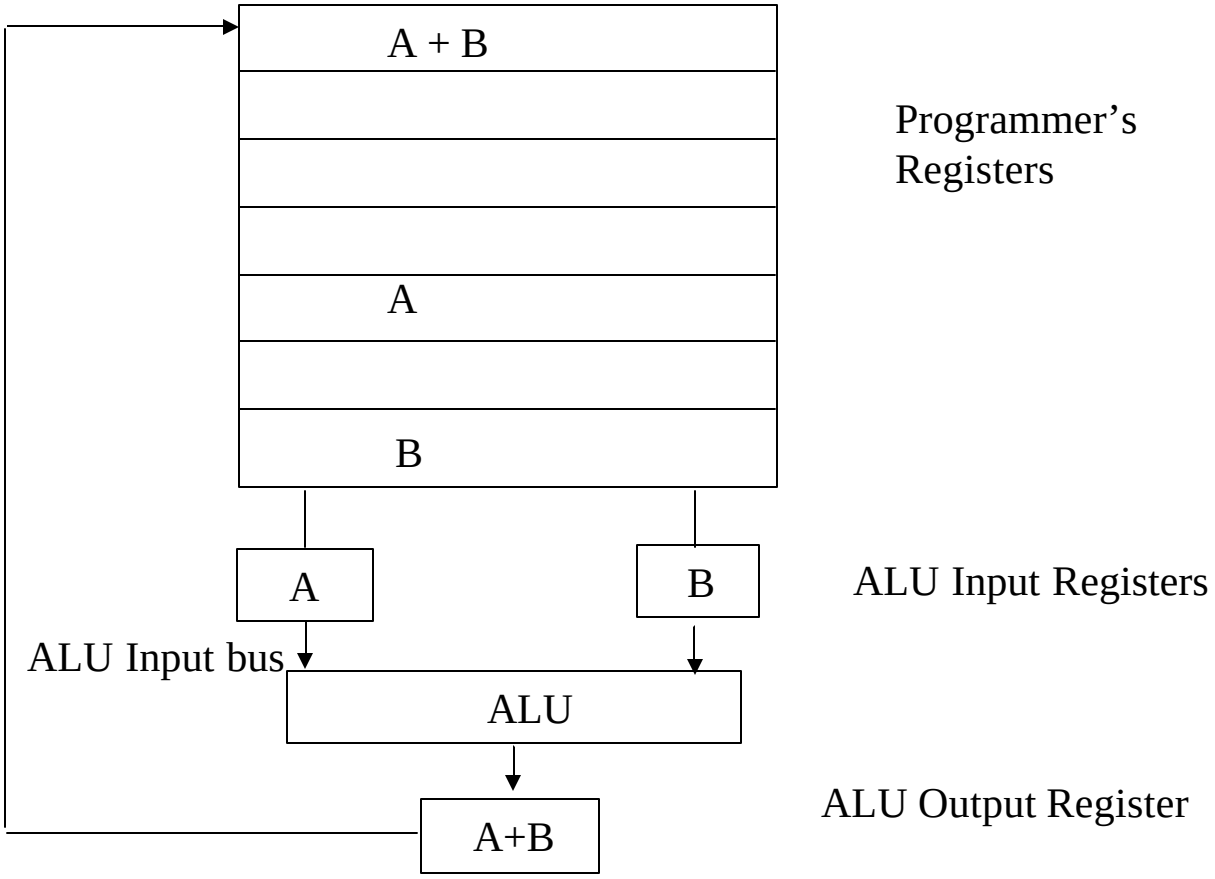
- Instruction execution cycle: sequence of operations required to execute a single instruction
  - *Control unit* **fetches** next instruction from memory, **decode** it and then **execute** it
    - Execution of instruction may require further fetches of data from memory and/or storage of data to memory
  - How do we start the instruction execution cycle ?

Simple Computer System Organization (expanded)

Central Processing Unit (CPU)



Data Path (Tanenbaum Figure 2-2)



Data Path (Tanenbaum Figure 2-2)

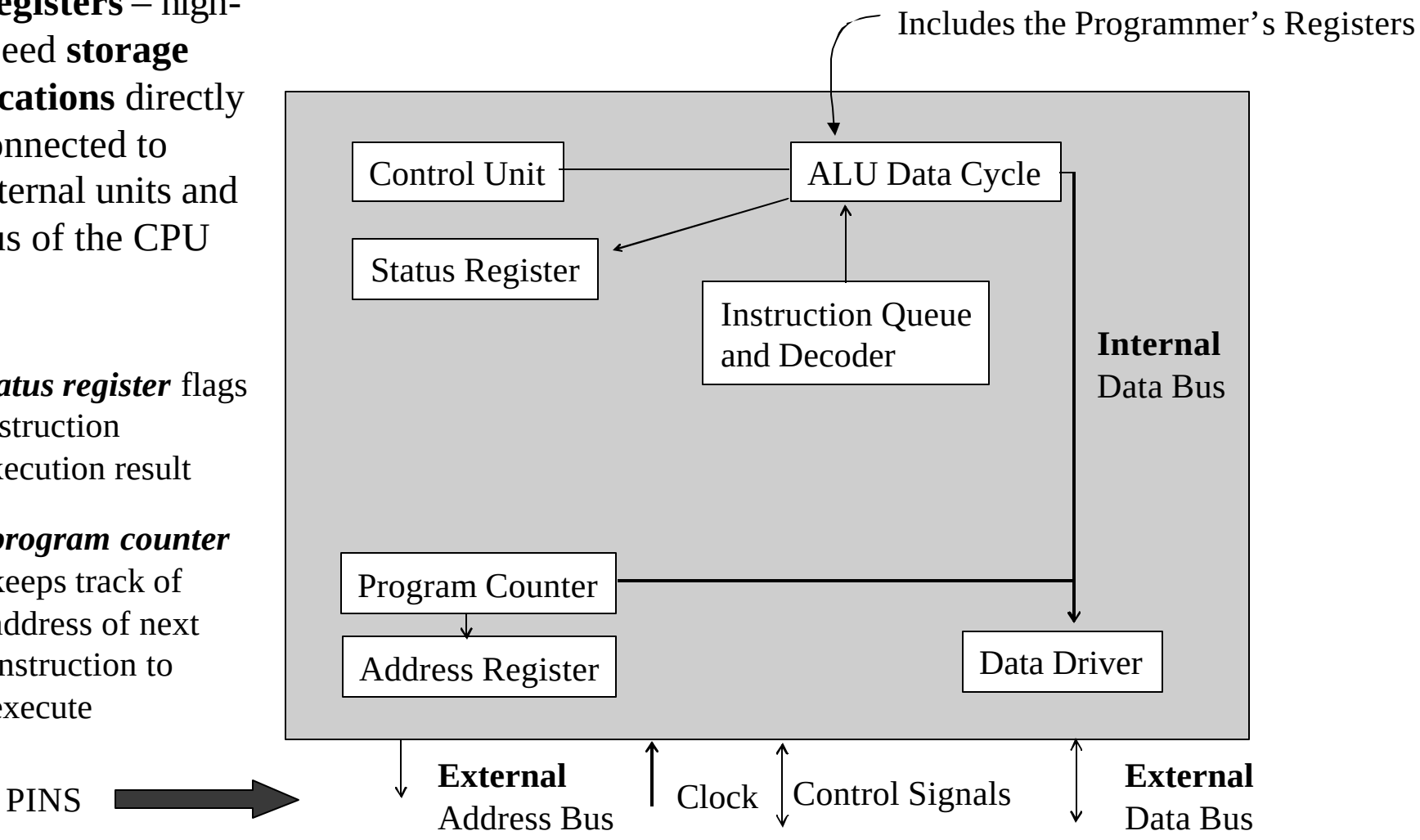
- Two major categories of instructions
  - **Register-memory**: transfer information between registers and memory
    - Read: data can be used as **inputs** to the ALU in subsequent instruction(s)
    - Write: results of ALU can be **stored persistently**
  - **Register-to-Register**
    - Typically: fetch two operands from programmer's registers, used as two ALU inputs and storing ALU output back in one of the registers
      - Eg. addition, boolean
    - Called the data path cycle
      - Defines what the computer can do
      - Faster the data cycle, the faster the machine

Central Processing Unit Organization

Registers – high-speed **storage locations** directly connected to internal units and bus of the CPU

status register flags instruction execution result

program counter keeps track of address of next instruction to execute



Instruction Execution Cycle

- For each machine instruction: control unit uses program counter (PC) and instruction queue to run the **Fetch-Execute Cycle**

1. **Fetch Phase**: read memory at the address given in the PC. Copy instruction into internal instruction queue.
2. **Decode Phase**: determine type of instruction and number/location of operands.
3. **Execute Phase** :
    1. Fetch **Operands**: [if needed] read memory, copy data into ALU input register
    2. Instruction Execution: **signal ALU** to perform operation. Send data to **output register** and **set status flags**
    3. **Store** output operand: [if needed] write to memory to store data from ALU output register.

Instruction Execution Cycle

- The instruction execution cycle is the “heartbeat” of the computer.
  - It is all synchronized on the CPU clock.

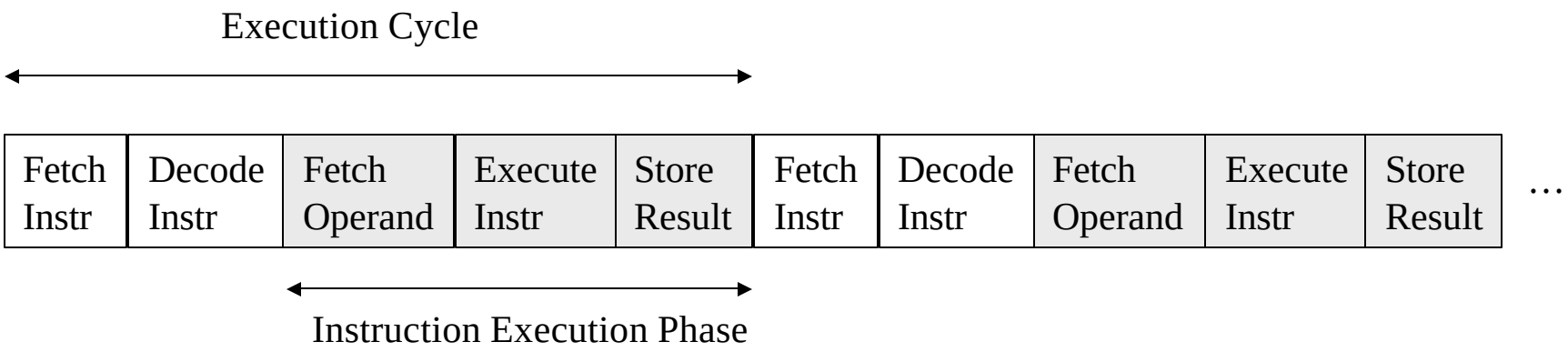
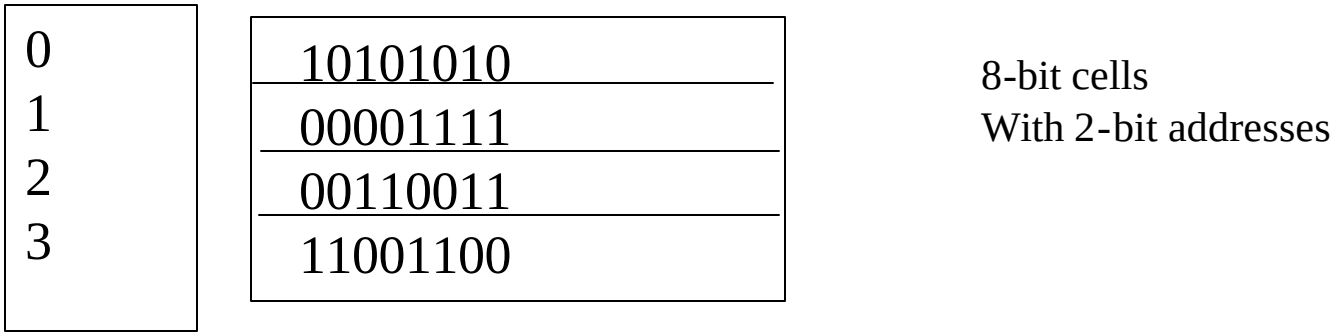


Figure 1.9 Dandamudi

- What is the execution time of an instruction ?

Memory

- Central storage for (program) instructions and data
  - Memory **persistent**; changes only as result of a write
  - What are the contents before the first write (during power-on) ?
- Memory: sequence of directly **addressable locations**.
  - Organized as 8-bit cells (bytes). Modern processors: “wider” addressable locations (16 bit, 32 bit, 64 bit!), each byte can be accessed.



  - Wider memories: **faster access to the processor**.

Memory

- Memory location: **address** and its **contents**.

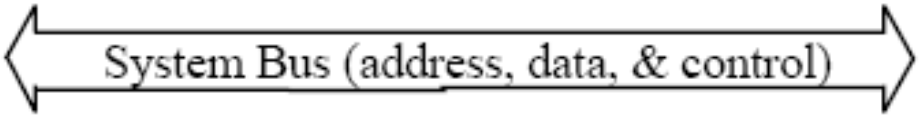
- Notation: m[addr] represents the contents of a memory location,
  - Example: m[20h] refers to contents of memory location at 20h.

- Two memory-related behaviours :
  - **write**(value, addr) and value = **read**(addr)

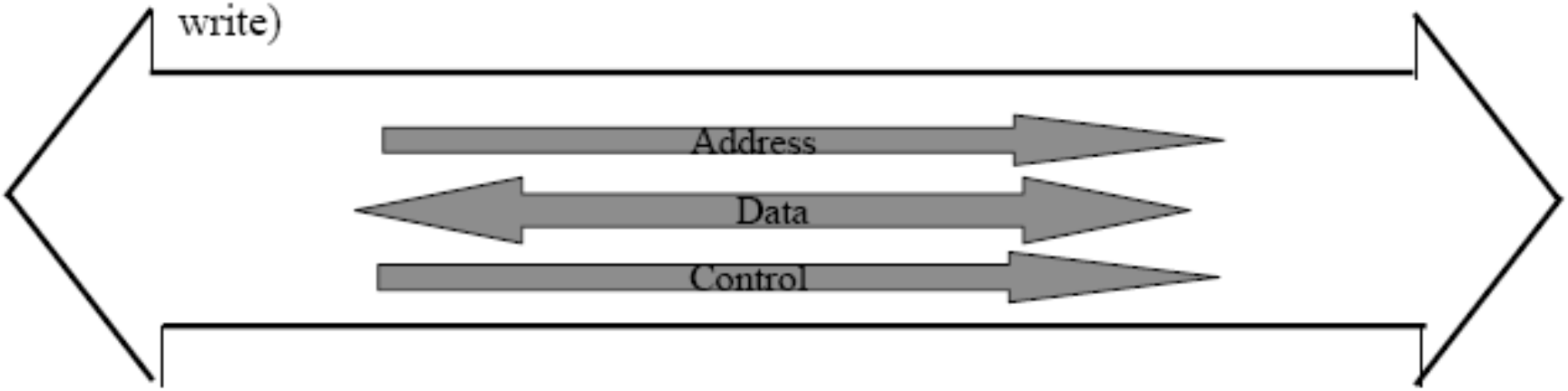
Two parameters
… **Two buses**

Memory – and the System Bus

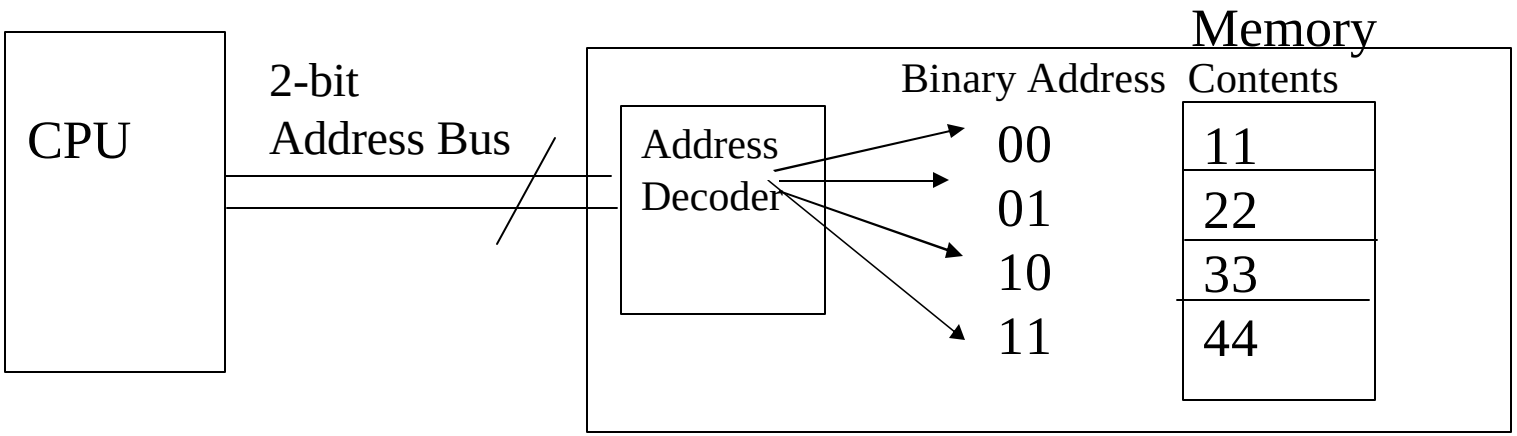
- A memory location has two components: an address and its contents.
- You can read or write a memory location



- Consequently, the system bus has three components ….
  - Address bus : Contains the address of the memory location
  - Data bus : Contains the data being read or written
  - Control bus : Contains signals that control the flow of the data (read vs write)

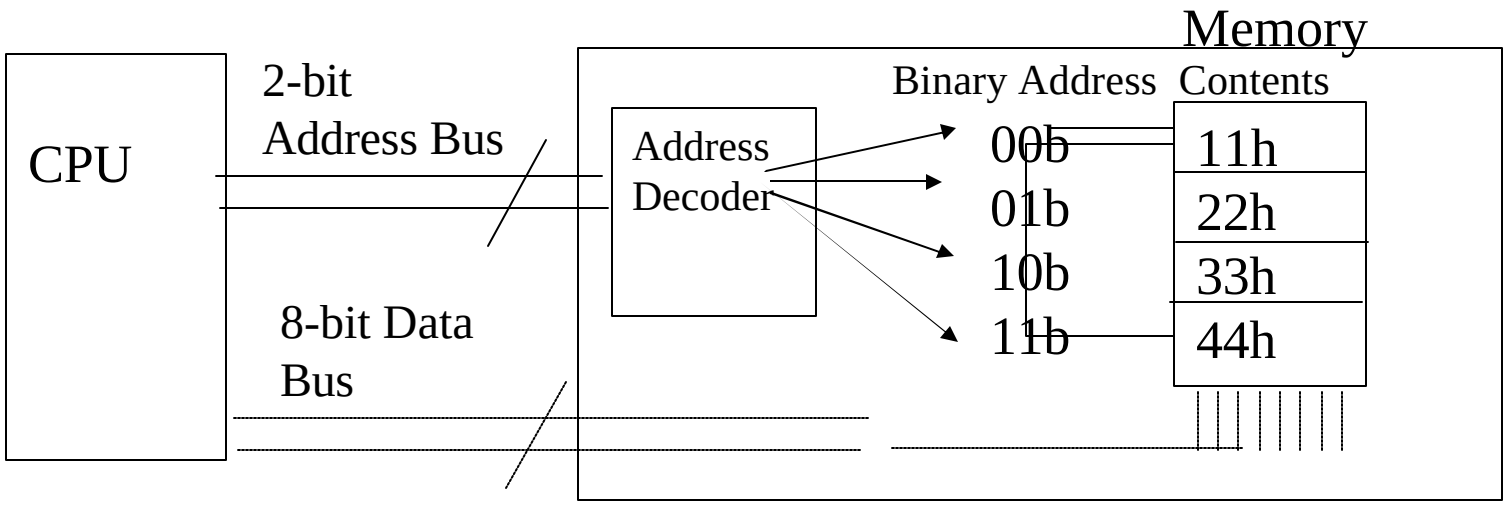


What is a Memory Address ? (A programmer's view)



- Memory cell (byte) is the **smallest addressable memory unit**
  - You cannot "address" individual bits in a cell
- **Address Space**?
  - Address bus: given width. Range or capacity of this memory?
    - 2-bit address : 2^2 different addresses
    - 20-bit addresses: 2^{20} different addresses = 1 Meg address space
      - (Hint : 1K = 2^{10} = 1024)

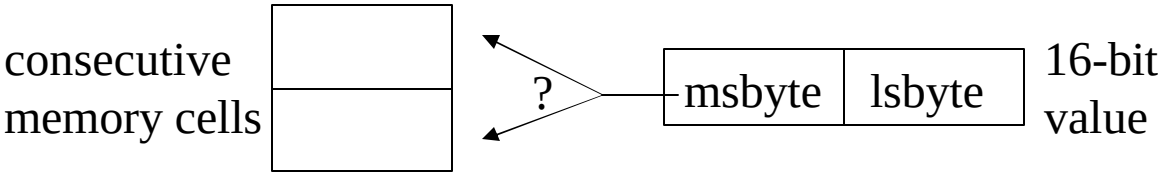
What about the Data bus ?



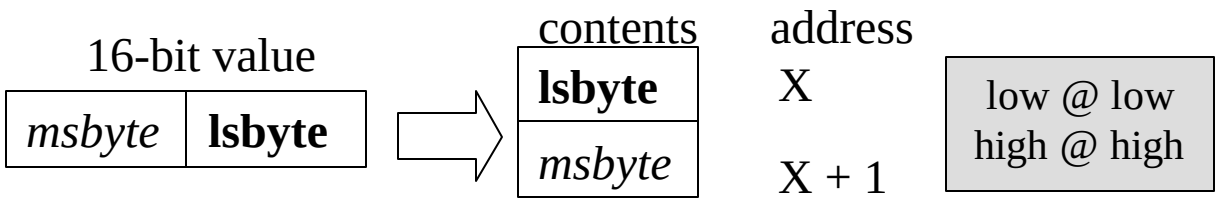
- Memory viewed logically as **array of contiguous bytes**. Many processors 16-bit (and above), meaning
  - They have a **16-bit data bus**
  - They can read/write 16-bit contents all in one read/write cycle.
- How to store binary values wider than 8-bits using 8-bit memory cells ?
  - Use **more than one** consecutive memory locations

Endian Memory Schemes

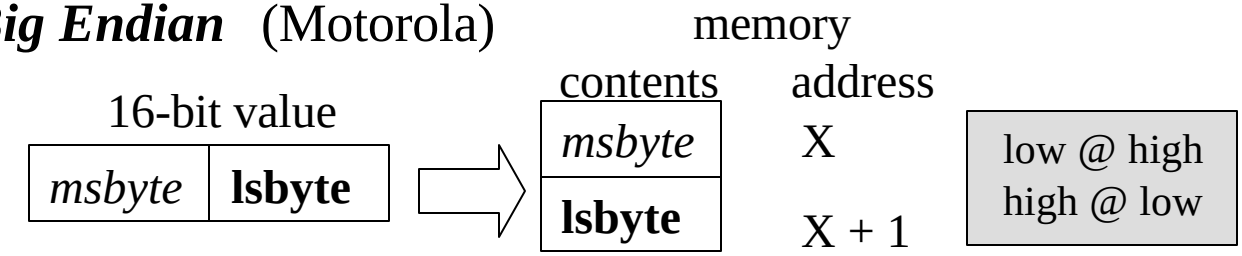
New Problem: 16-bit values, stored in **two memory locations**.
Which location should hold which byte ?



Little Endian (Intel)

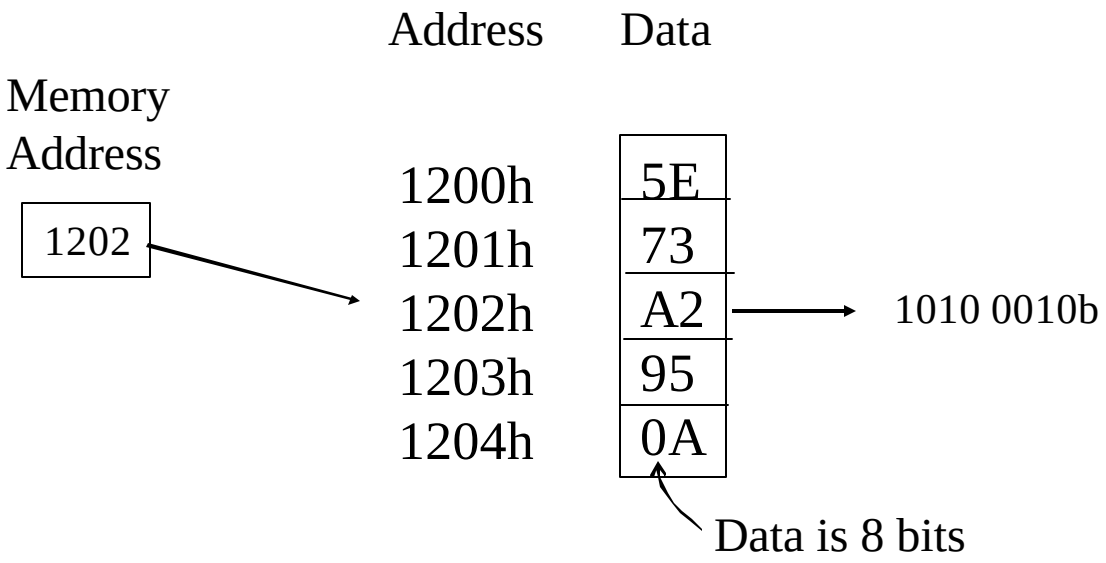


Big Endian (Motorola)



Little Endian Exercise

- What is the byte at address 1201 ?
- What is the word at address 1201 ?
- What is the double-word at address 1201 ?



Semiconductor Memory

- **Read and Modify (Random Access Memory -** RAM): same amount of time required to access any location on the same chip
- **Read-only memory** (ROM): can only be read; not written

Random Access Memory

-**Dynamic random access memory** (DRAM): periodic refresh required to maintain contents of a DRAM chip
-**Static random access memory** (SRAM): no periodic refresh required

Read-Only Memory

-**Mask-programmed read-only memory** (MROM): programmed when manufactured
-**Programmable read-only memory** (PROM): memory chip can be programmed by end user

Erasable Programmable ROM (EPROM)

1. Electrically programmable many times
2. Erased by ultraviolet light (through a window)
3. Erasable in bulk (whole chip in one erasure operation)

Electrically Erasable Programmable ROM (EEPROM)

1. Electrically programmable many times
2. Electrically erasable many times
3. Can be erased one location, one row, or whole chip in one operation

Flash Memory

1. Electrically programmable many times
2. Electrically erasable many times
3. Can only be erased in bulk