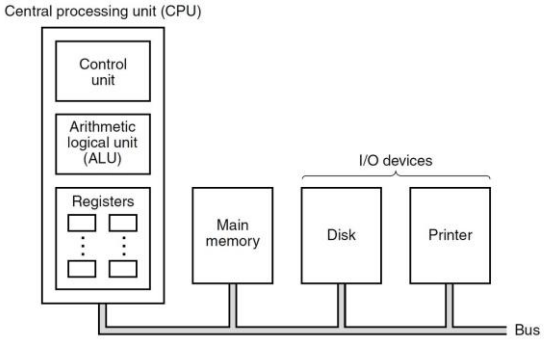
Computer Organization

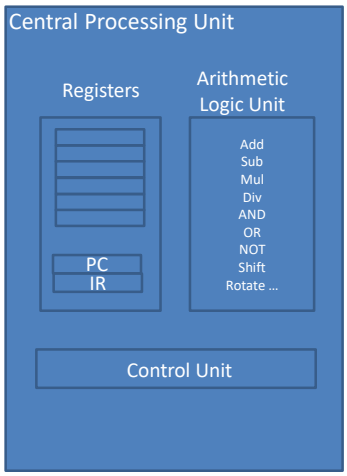
- The figure shows the organization of a simple bus-oriented computer.
- The **CPU** (**Central Processing Unit**) is the engine of the computer. Its function is to execute programs stored in the main memory by
  - Fetching their instructions,
  - Decoding them,
  - Executing them one after another.
- The components of our computer are connected by a **bus**,
  - a collection of parallel wires for carrying address, data, and control signals.
  - Buses can be external to the CPU,
    - connecting it to memory and I/O devices
    - Internal to the CPU
    - Modern computers have multiple buses.



45

Computer Organization

- The CPU is composed of several distinct parts.
  - The **control unit** fetches instructions from main memory and determines their type and what to do with them.
  - The **Arithmetic and Logic Unit** (ALU) performs operations such as
    - addition, subtraction, multiplication and division
    - logical operations such as AND, OR, NOT, Shift, and Rotate



46

Computer Organization

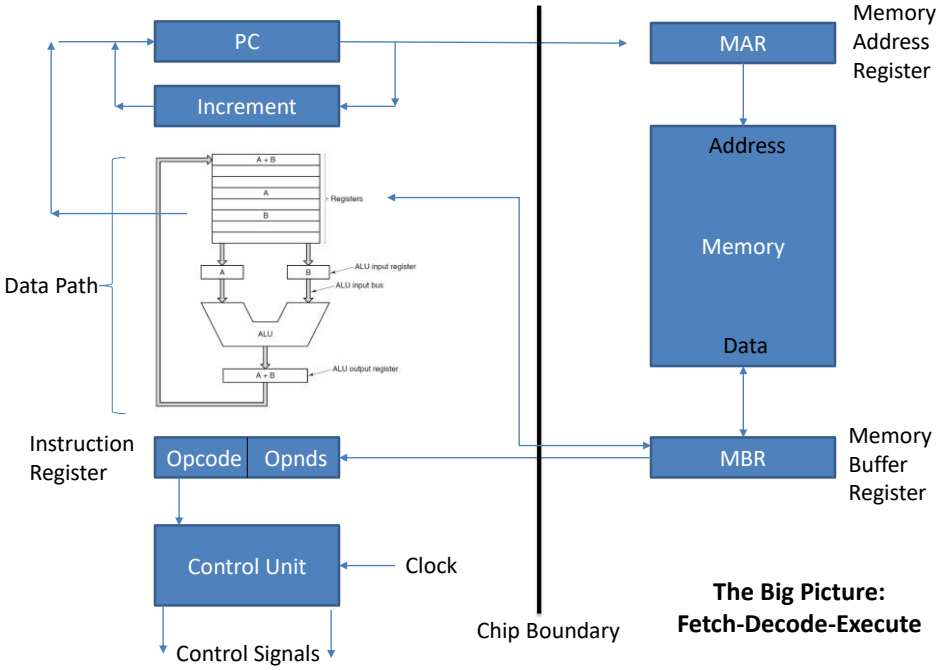
- **Registers**
  - high-speed memory locations used to store temporary results and certain control information.
  - Fast access because they are internal to the CPU
  - Data registers for operand and result data
    - Limited in size and number
    - Number and size of each depends on wider architectural decisions
    - These decisions influence program size, execution time, compiler complexity, etc.

47

Computer Organization

- **Registers**
  - Control registers to hold control information used for supporting the operation of the machine.
    - the **Program Counter** (**PC)** also called the Instruction Counter or the **Instruction Pointer**, holds the address of the next instruction to be fetched for execution
    - the **Instruction Register** (**IR**), holds the instruction currently being executed.
    - The **Status Register** reflects the outcome of the execution of the previous instruction
    - The **Stack Pointer** is used to implement subroutine calls and returns

48



49

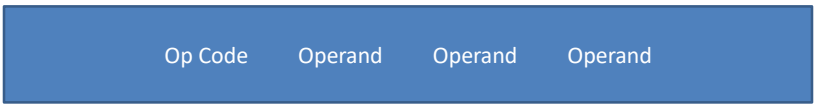
Fetch-Decode-Execute

- The sequencing of the steps described below is done by the control unit issuing control signals to the relevant pieces of hardware at the relevant times.
- Before the CPU can execute an instruction, the instruction must be first brought to the CPU from the computer's memory
- Our description of the way in which a program is executed begins with the CPU's Program Counter (the PC).
- The PC holds the address of the next instruction to be executed .
- To identify an instruction, the contents of the PC is moved into the Memory Address Register.
- The PC is incremented by 1, after which it holds the address of the expected next instruction, while the current instruction is being executed.
- A Memory Read cycle is performed.
- At this point, the Memory Buffer Register will contain the bit pattern of the instruction to be executed.

50

Fetch-Decode-Execute

- The instruction is next moved from the MBR to the Instruction Register (the IR). An instruction is composed of two parts. One part identifies the type of operation to be performed. This part is called the operation code (Op Code) field and the other part refers to any operands that might be needed when carrying out the operation. This part is called the operands field. An operand may be literally present in an operand field or the operand field may contain information on where to find the operand. The operand field may also indicate where any result of the operation is to be placed.



The steps above constitute the Instruction Fetch Phase

51

Fetch-Decode-Execute

- Once in the IR, the Instruction's opcode is used by the Control Unit to generate appropriate control signals to create a pathway through the CPU.
- This pathway is a virtual circuit connecting various components in the CPU, over which data, represented by electrical signals (- signal which are derived from the binary representation of the data), will flow.
- Determining this pathway is called the Instruction Decode Phase

52

Fetch-Decode-Execute

- Once the appropriate pathway through the CPU has been established, the Control Unit enables and disables various parts of the CPU as appropriate.
- As a result, data (bits, electrical signals) flow around the CPU and may be transformed by the components along the pathway into other bits as they proceed.
- After a finite amount of time, these signals settle down to a stable state – they stop changing – and the instruction is said to have executed.
- This is called the Instruction Execution Phase

53