K.Nreddy

UNIT-I

Introduction to Computer Organization and Architecture Basic Computer Organization - CPU organization - Memory Subsystem organization and Interfacing - I/O Subsystem Organization and Interfacing - A Simple Computer - Levels of Programming Languages, Assembly Language Instructions, Instruction Set Architecture Design, A simple instruction set Architecture.

Introduction:

The basic functional units of computer are made of electronics circuit and it works with electrical signal. We provide input to the computer in form of electrical signal and get the output in form of electrical signal.

There are two basic types of electrical signals, namely, **analog** and **digital**. The analog signals are continuous in nature and digital signals are discrete in nature.

The electronic device that works with continuous signals is known as **analog device** and the electronic device that works with discrete signals is known as **digital device**.

Computer is a digital device, which works on two levels of signal. We say these two levels of signal as **High** and **Low**. The High-level signal basically corresponds to some high-level signal (say 5 Volt or 12 Volt) and Low-level signal basically corresponds to Low-level signal (say 0 Volt). This is one convention, which is known as positive logic.

Computer is used to solve mainly numerical problems. Again it is not convenient to work with symbolic representation. For that purpose we move to numeric representation. In this convention, we use 0 to represent **LOW** and 1 to represent **HIGH**.

0 means LOW

1 means HIGH

With the symbol 0 and 1, we have a mathematical system, which is knows as **binary number system**. Basically binary number system is used to represent the information and manipulation of information in computer. This information is basically strings of 0s and 1s.

The smallest unit of information that is represented in computer is known as **Bit** (Binary Digit), which is either 0 or 1. Four bits together is known as **Nibble**, and Eight bits together is known as **Byte**.

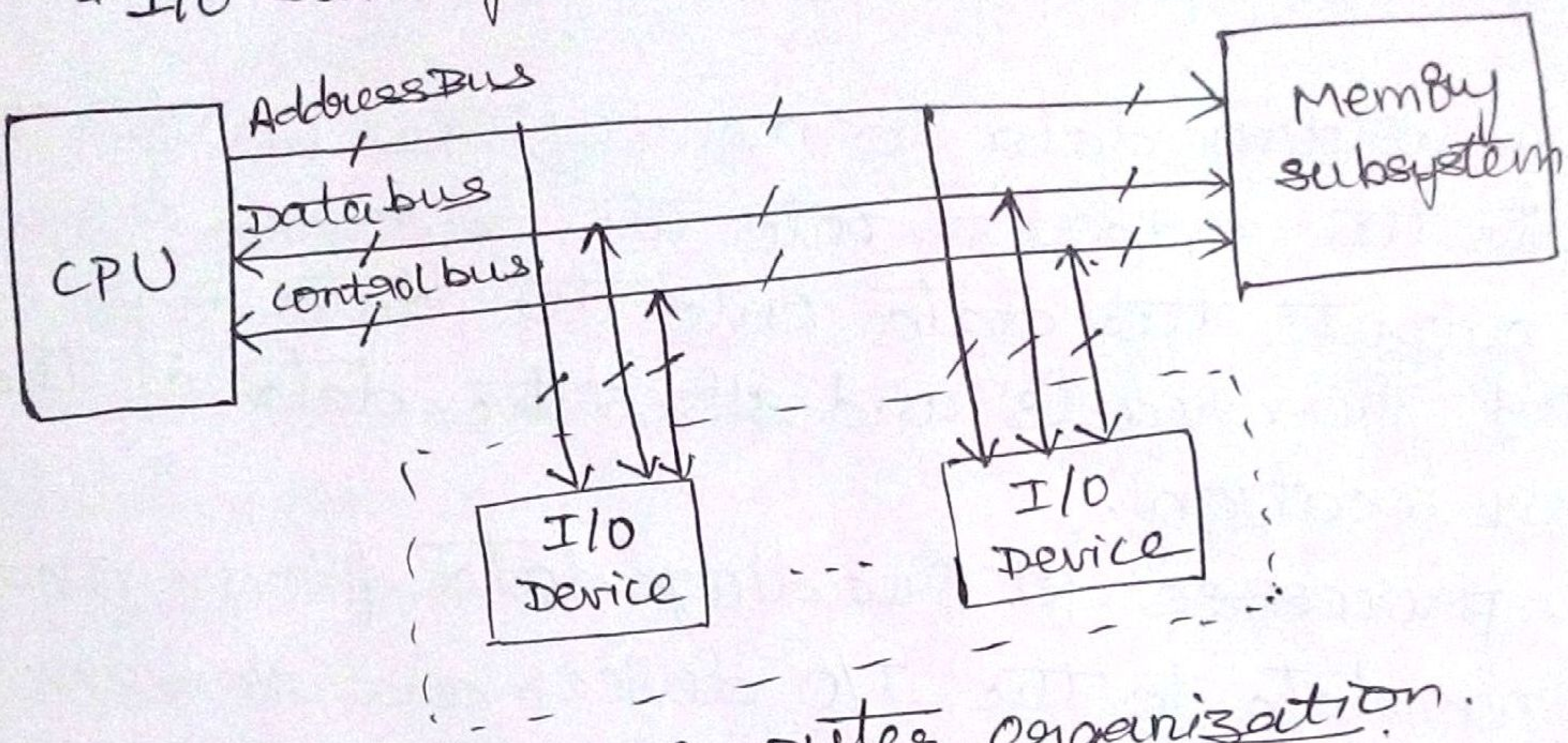
Computer architecture refers to those parameters of a computer system that are visible to a programmer or those parameters that have a direct impact on the logical execution of a program. Examples of architectural attributes include the instruction set, the number of bits used to represent different data types, I/O mechanisms, and techniques for addressing memory.

Computer organization refers to the operational units and their interconnections that realize the architectural specifications. Examples of organizational attributes include those hardware details transparent to the programmer, such as control signals, interfaces between the computer and peripherals, and the memory technology used.

⇒ Basic Computer Organization :-

Basic computer organization has three main components:

- CPU
- Memory sub system
- I/O sub system.



1.1 Generic computer organization.

→ System Buses

- Physically, a bus is a set of wires. The components of the computer are connected to the buses.
- To send information from one component to another, the source component outputs data onto the bus. The destination component then inputs this data from the bus.

Address bus :-

- When CPU reads data or instructions from or writes data to memory, it must specify the address of memory location. It outputs this address to the address bus; memory inputs this address from the address bus and uses it to access the proper memory location.
- Each I/O device has unique address.
- The Address bus always receives data from the CPU; the CPU never reads the address bus.

Databus

- Data is transferred via the data bus.
- When the cpu fetches data from memory, it first outputs the memory address on its address bus. Then memory outputs the data onto the data bus. The cpu can then read the data from the data bus.
- When writing data to memory, the CPU first outputs the address onto the address bus, then outputs the data onto the data bus. Memory then reads and stores the data at the proper location.
- The processes for reading data from and writing data to the I/O devices are similar.

Control bus:-

- Control bus is different from the other two buses.
- Control bus is collection of individual control signals. These signals indicate whether data is to be read into or written out of the CPU, whether CPU is accessing memory or an I/O device, or whether the I/O device or memory is ready to transfer data.
- A system may have hierarchy of buses.
- Local bus - eg: PCI (peripheral component Interconnect).

→ Instruction cycles

- The instruction cycle is the procedure a microprocessor goes through to process an instruction.
- First the microprocessor fetches, or reads, the instruction from memory. Then it decodes the instruction, determining which instruction it has fetched. Finally, it performs the operation necessary to execute the instruction.

Fetch

- First the microprocessor places the address of the instruction on to the address bus. The memory subsystem inputs this address and decodes it to access the desired memory location.
- After microprocessor allows sufficient time for memory to decode the address and access the requested memory location, the microprocessor asserts a READ control signal. The read signal is a signal on the control bus which the microprocessor asserts when it is ready to read data from memory or an I/O device.
- when read signal is asserted, the memory subsystem places the instruction code to be fetched onto the computer systems data bus.
- The microprocessor then inputs this data from the bus and stores and stores it in one of its internal registers. At this point, the microprocessor has fetched the instruction.

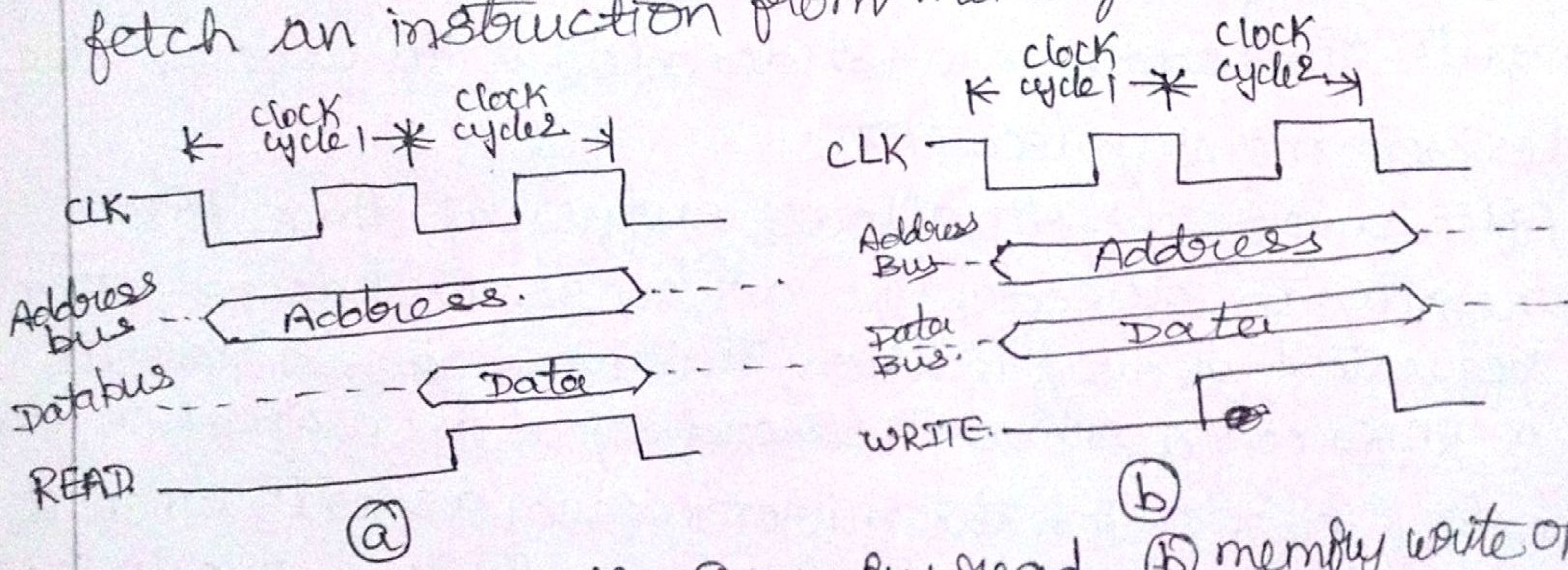
Decode

- When the microprocessor decodes the instruction, it determines which instruction it is in order to select the correct sequence of operations to perform.
- This is done entirely within the microprocessor; it doesnot use the system bus.

Execute

- Finally the microprocessor executes the instruction. The sequence of operations to execute the instruction varies from instruction to instruction.
- The execute routine may read data from memory, write data to memory, read data from or write data to an I/O device, perform only within the CPU, or perform some combination of these operations.

- To read data from memory, the microprocessor performs the same sequence of operations it uses to fetch an instruction from memory. [1.2]



Timing diagram for @ memory read ⓑ memory write operations

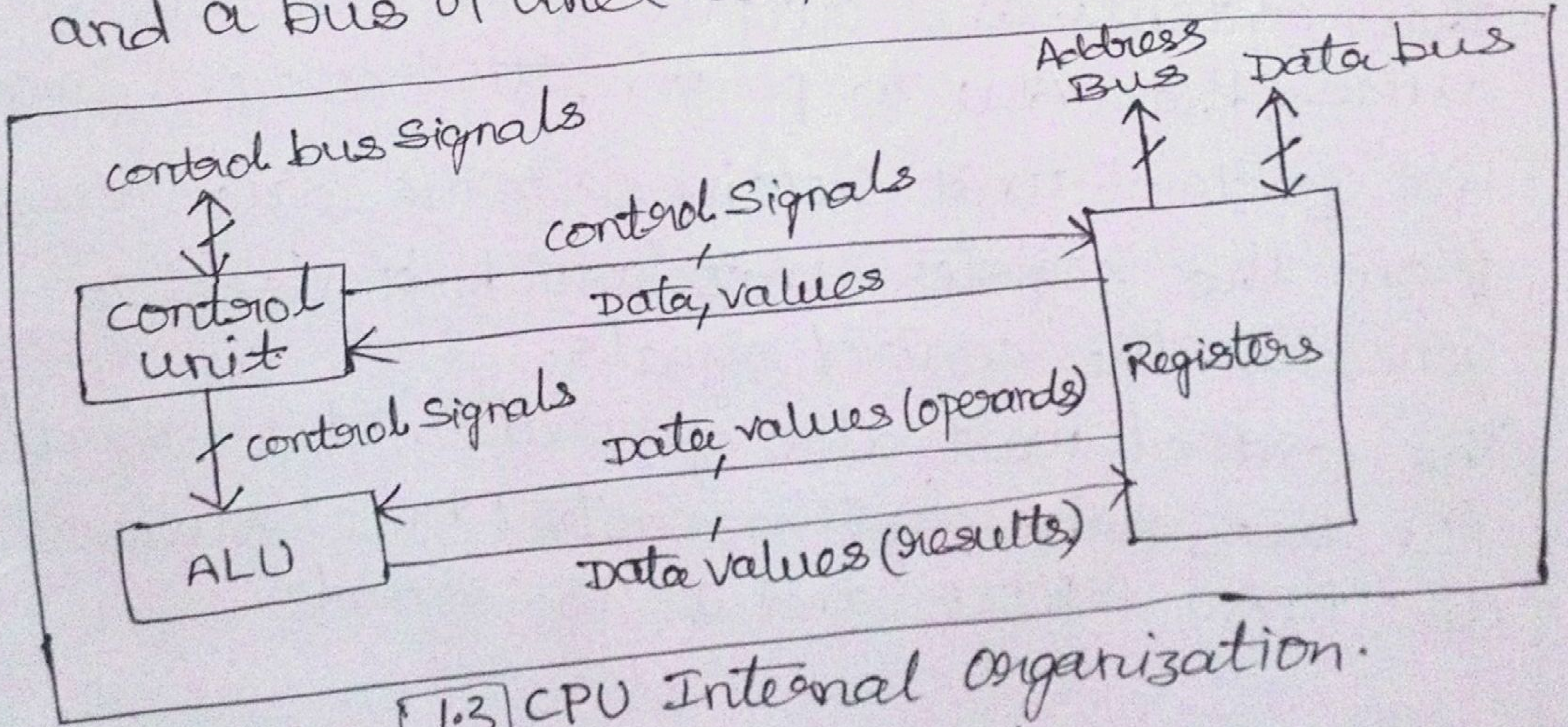
CLK - system clock; microprocessor uses the system clock to synchronize its operations.

- The microprocessor places the address onto the bus at the beginning of a clock cycle, a 0/1 sequence of the system clock.
- One clock cycle later, the memory decode the address and access its data, the microprocessor asserts the READ signal. This cause memory to place its data onto the system data bus. During this clock cycle, the microprocessor reads the data of the system bus and stores it in one of its registers.

- At the end of the clock cycle it removes the address from the address bus and deasserts the READ signal.
- memory then removes the data from the data bus, completing the memory read operation.

⇒ CPU ORGANIZATION

- The CPU controls the computer.
- It performs some operations internally, and supplies the address, data, and control signals needed by memory and I/O devices to execute the instruction.
- CPU has three sections: Registers, ALU, control Unit.
- The register section includes a set of registers and a bus or other communication mechanism.



1.3 CPU Internal organization.

- CPU contains following registers.
  - Address register (AR), which supplies an address to memory.
  - Program counter (PC), which contains the address of next instruction to be executed.
  - Data register (DR) which receives instructions and data from memory.

- Instruction register (IR) which stores the opcode portion of the instruction code fetched from memory.

→ The Arithmetic/logic unit or ALU, performs most arithmetic and logical operations, such as adding or ANDing.
- It receives its operands from the register section of the CPU and stores its results back in the register section.
- Since the ALU must complete its operations within a single clock cycle, it is constructed using only combinatorial logic.

→ Control Unit controls the CPU. This unit generates the internal control signals that cause registers to load data, increment or clear their contents, and output their contents as well as cause the ALU to perform the correct function.
- The control unit receives some data values from the register unit, which it uses to generate the control signals.
- The control unit also generates the signals for the ~~stbs~~ system control bus such as the READ, WRITE and IO/$\overline{M}$ signals.