Chapter one: Microcomputer Architecture

Microcomputer Architecture

A computer system has three main components: **a Central Processing Unit (CPU) or processor**, **a Memory Unit** and **Input Output Units (devices)**. In any microcomputer system, the component which actually processes data is entirely contained on a single chip called **Microprocessor** (MPU). This MPU can be programmed using assembly language. Writing a program in assembly language requires a knowledge of the computer hardware (or Architecture) and the details of its instruction set.

The main **internal hardware** features of a computer are the processor, memory and **registers** (registers are special processor components for holding address and data).

The **external hardware** features are the computer Input/output devices such as keyboard, monitor…

Software consists of the operating system (O.S) and various programs and data files stored on disk. Inside any computer based on a member of the 8086 family, the basic arrangement of the main components is shown in Figure 1.

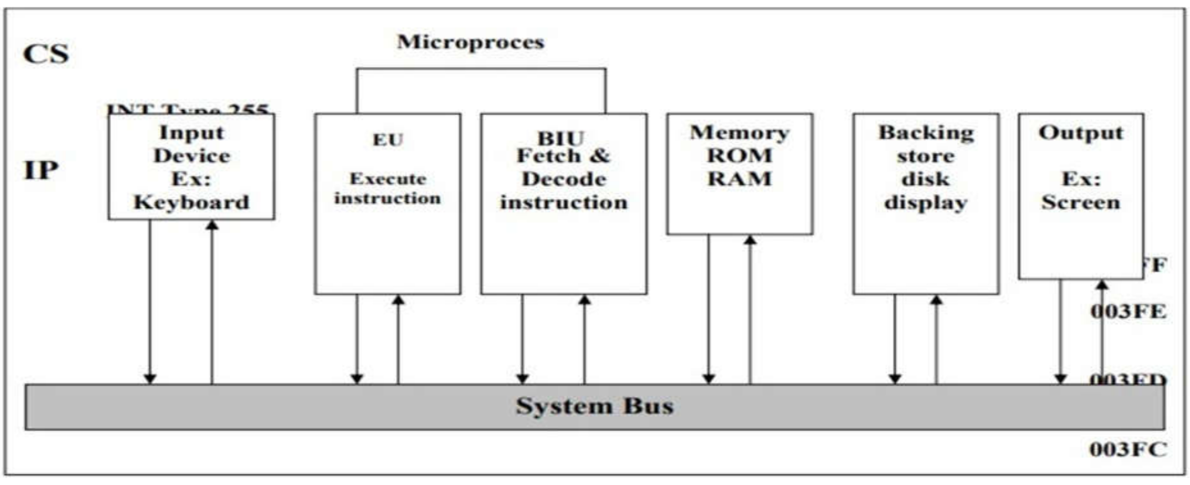

Figure 1: Data flow between the main components of an 8086 family computer.

Information is sent from one main component to another along the communication channel, which is often called the **System Bus**. Both programs and data are stored in the memory.

The **Bus Interface Unit** (BIU) within the MPU fetches new instruction or data as necessary. It is also the BIU jobs to interpret or decode instruction and to route results to their proper destination.

The MPU **Execution Unit** carries out any arithmetic which is required, including memory calculation. Microcomputer memories consist of a collection of chips of two kinds **Read Only Memory** (ROM) and **Random Access Memories** (RAM).

System Bus

The components of the computer system must communicate with each other and with the outside world. Although it may be possible to connect each component to the CPU separately as a practical matter this would require too many physical connects. To keep the number of connections manageable, the processor is connected to memory and all peripherals using a bus.

A **Bus** is a bunch of wires, and electrical path on the printed IC to which everything in the system is connected.

There are three types of Bus:

- **Address Bus (AB)**: the width of AB determines the amount of physical memory addressable by the processor.
- **Data Bus (DB)**: the width of DB indicates the size of the data transferred between the processor and memory or I/O device.

- **Control Bus (CB)**: consists of a set of control signals, typical control signals includes memory read, memory write, I/O read, I/O write, interrupt acknowledge, bus request. These control signals indicates the type of action taking place on the system bus.

Personal Computer (PC) Components

The main component of the PC is its **System Board** (or mother board). It contains the processor, co-processor, main memory, connectors, and expansion slots for optional cards. The slots and connectors provide access to such components as ROM, RAM, hard disk, CD-ROM drive, additional memory, video unit, keyboard, mouse, parallel and serial device, sound adapter and cache memory (the processor use high speed cache memory to decrease its need to access the slower main memory). A bus with wires attached to the system board connect the components. It transfers data between the processor, memory and external devices.

A. The processor

The CPU or processor acts as the controller of all actions or services provided by the system. The operations of a CPU can be reduced to three basic steps: **fetch**, **decode**, and **execute**. Each step includes intermediate steps, some of which are:

- **Fetch the next instruction**:
  - Place it in a holding area called a queue.
  - Decode the instruction.
- **Decode the instruction** :
  - Perform address translation.
  - Fetch operand from memory.

➢ **Execute the instruction**:

- Perform the required calculation.
- Store results in memory or register.
- Set status flag attached to the CPU.

Figure 2 shows a block diagram of a simple imaginary CPU. The CPU is divided into two general parts. **Arithmetic Logic Unit** (ALU) and **Control Unit** (CU).

✓ The **ALU** carry Arithmetic, logical, and shifting operations.

✓ The **CU** fetches data and instruction, and decodes addresses for the ALU

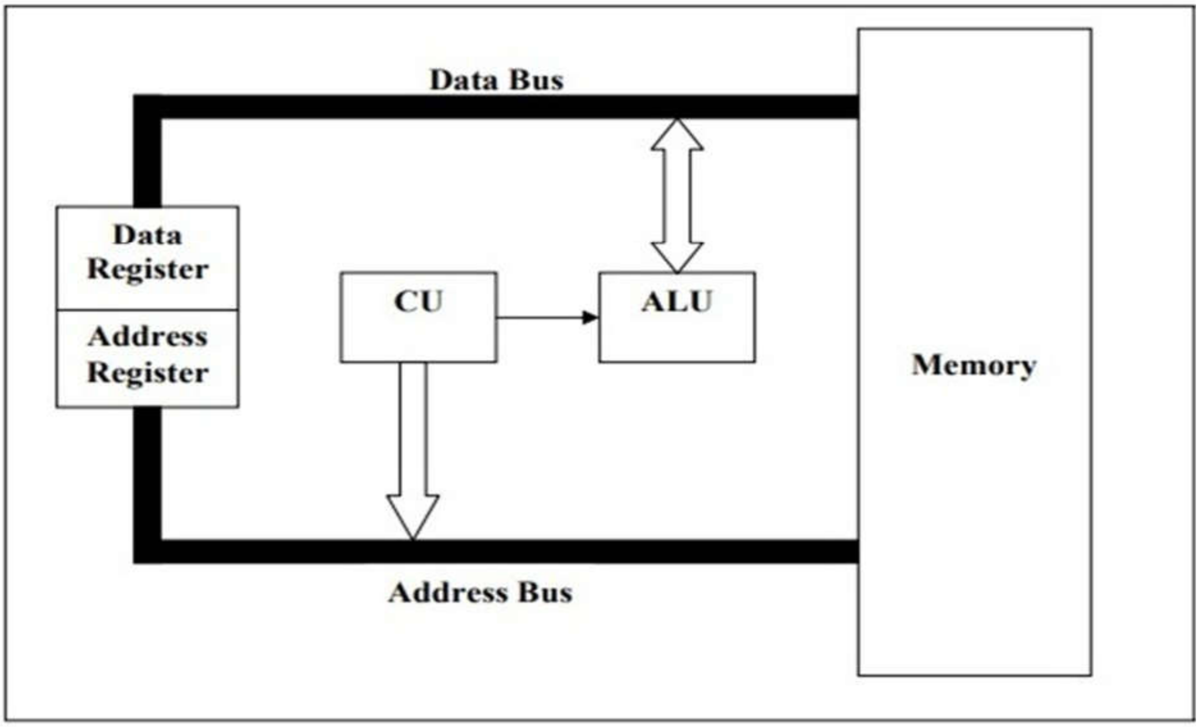

Figure 2: A block diagram of a simple CPU.

B. Memory

The memory of a computer system consist of tiny electronic switches, with each switch set in one of two states: open or close. It is however more convenient to think of these states as **0** and **1**.

Thus each switch can represent a binary digit or bit, as it is known, the memory unit consists of millions of such bits, bits are organized into groups of eight bits called **byte**.

Memory can be viewed as consisting of an ordered sequence of bytes. Each byte in this memory can be identified by its sequence number starting with 0, as shown in Figure 3. This is referred to as memory address of the byte. Such memory is called **byte addressable memory**.

8086 can address up to 1 MB (220 bytes) of main memory this magic number comes from the fact that the address bud of the 8086 has 20 address lines. This number is referred to as the **Memory Address Space** (MAS).

The memory address space of a system is determined by the address bus width of the CPU used in the system. The actual memory in a system is always less than or equal to the MAS.

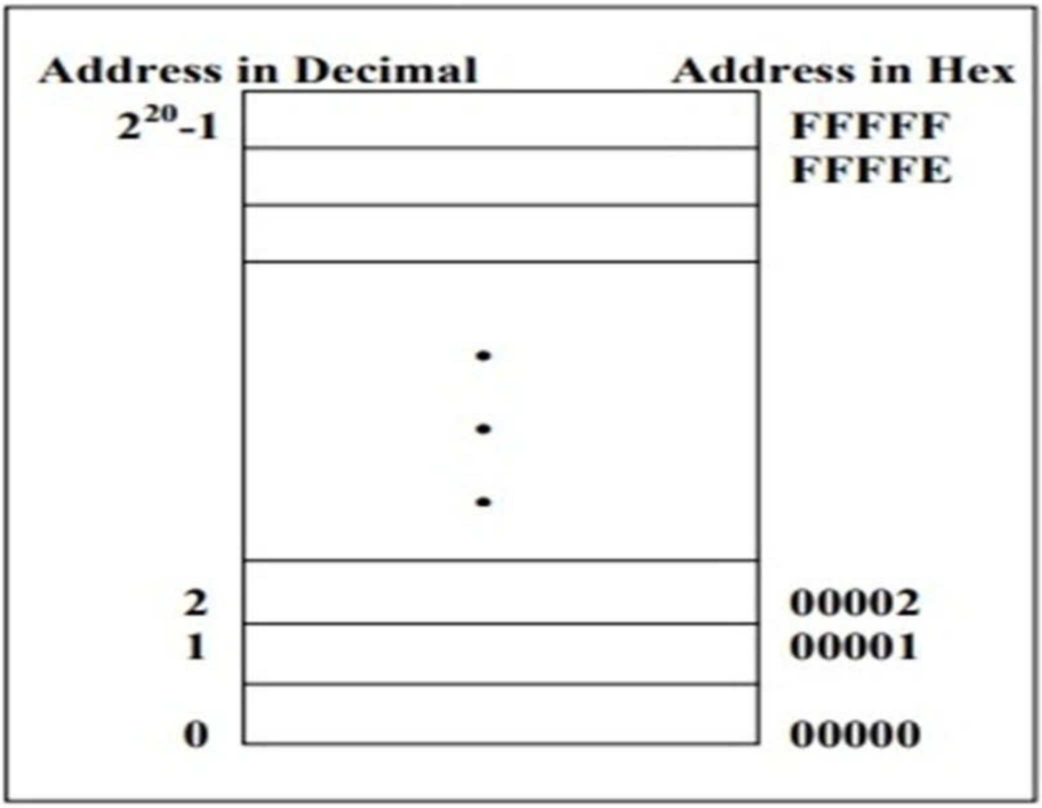

Figure 3: Logical view of the system memory

Two basic memory operations

The memory unit supports two fundamental operations: Read and Write. The read operation read a previously stored data and the write operation stores a value in memory. See Figure 4

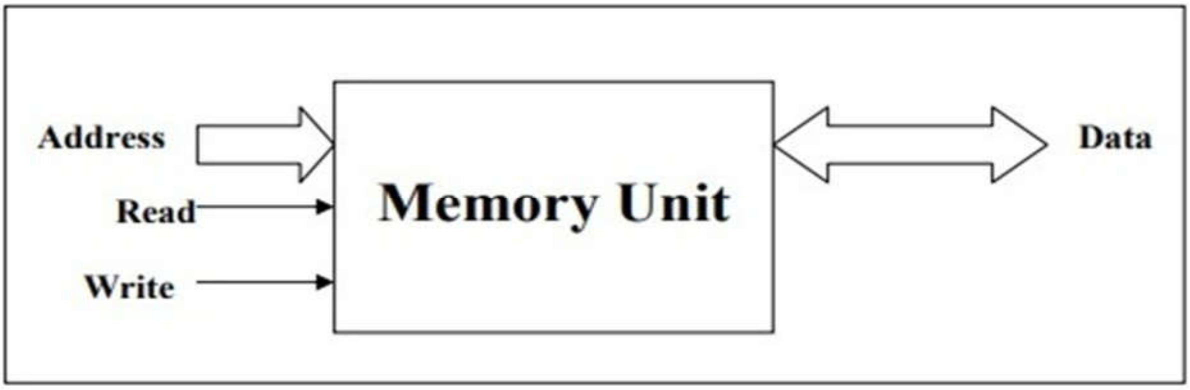

Figure 4: Block diagram of system memory

Steps in a typical read cycle

- ❑ Place the address of the location to be read on the address bus.
- ❑ Activate the memory read control signal on the control bus.
- ❑ Wait for the memory to retrieve the data from the address memory location.
- ❑ Read the data from the data bus.

Drop the memory read control signal to terminate the read cycle.

Steps in a typical write cycle

- o Place the address of the location to be written on the address bus.
- o Place the data to be written on the data bus.
- o Activate the memory write control signal on the control bus.
- o Wait for the memory to store the data at the address location.
- o Drop the memory write control signal to terminate the write cycle.