General architecture of computer: -

A computer:- is a multipurpose programmable machine that reads binary instructions from its memory, accepts binary data as input and processes data according to those instructions, and provides result as output.

- The physical components of the computer are called hardware
- A set of instructions written from the computer to perform a task is called a program and a group of programs is called software

the computer includes: CPU , memory, input and output as shown:

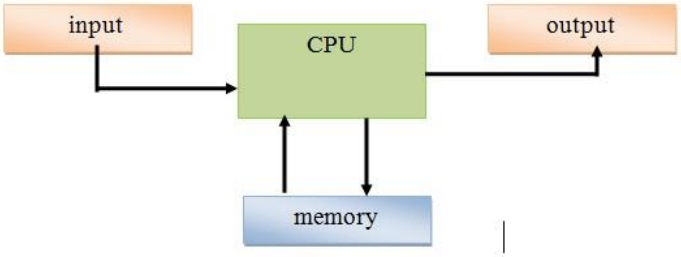


Fig –a- block diagram of computer

General architecture of A microcomputer:

The microcomputer is a complete computer similar to any other computer, except that the CPU functions of the microcomputer are performed by the microprocessor.

It includes four components: microprocessor, input, output and memory (read/write memory (RAM) and read only memory (ROM)). These components are organized around a common communication path called a bus.

Fig –b- shows a simplified but formal structure of a microcomputer.

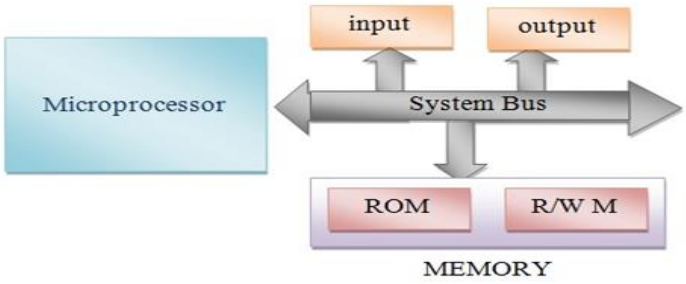


Fig-b- block diagram of a microcomputer

- A microprocessor is a general purpose processing unit built into one chip (a single integrated circuit (IC)). is the heart of a microcomputer that perform:-
  - executes instructions of the program and processes data.
  - It is responsible for performing all arithmetic and logic operations
  - controls overall system operation.

The mp can be divided into three segments:

1- Arithmetic / logic unit (ALU).
2- Register unit.
3- Control unit.

1- Arithmetic / logic unit (ALU): -

The ALU performs arithmetic operation such as addition and subtraction, and logic operations such as AND, OR and exclusive OR, the results are stored either in registers or in memory or set to output devices.

2- Register unit: -

The registers are used primarily to store data temporarily during the execution of program.

3- Control unit: -

This unit provides the necessary control signal to all the operations in the microcomputer.

- Input device : -

  Use to transfer data binary from the outside world to the mp. It includes devices such as keyboards, analog-to-digital converters.

- Output device : -

  Use to transfers data from the mp to output devices such as light emitting diodes (LEDs), printers.

- Memory: -

Use to stores binary information such as instructions and data, and provides information to the mp whenever necessary.

The memory has two types :

1- Read Only Memory (ROM)
2- Read /Write Memory (R/W M) or known as Random Access Memory (RAM)

1- ROM: - this type can store program single time and read it many times.
Programs stored in the ROM can only be read they cannot be altered.

Note: - data stored in ROM will not be lost when the power-off.

2- R/W M or RAM: - it is used to store user programs and data. The information stored in this memory can be read and altered easily.

Note: All data stored in RAM will be lost when power-off.

- System bus: -
  The system bus is a communication path between the mp and peripherals.
  It is a group of wire that carries bits.

Types of Microprocessors:-

Microprocessors generally is categorized in terms of the maximum number of binary bits in the data they process " their word length". Over time, five standard data widths have evolved for microprocessors: 4-bit, 8-bit, 16-bit, 32-bit, 64-bit. There are so many manufacturers of Microprocessors, but only two companies have been produces popular microprocessors: *Intel* and *Motorola*.

How does the microcomputer work?

Assume that a program and data are already entered in the R/WM. the instruction are stored sequentially in the memory. The mp fetches the first instruction from its memory, decodes it, and executes that instruction. The sequence of fetch, decode and execute is continued until the mp comes the instruction stop.
During the entire process, the mp uses the system bus to fetch the binary instruction and data from the memory. It uses registers from the register section to store data temporarily and it performs the computing function in the ALU section. Finally, it sends out the result in binary using the same bus lines, to output device.

Data types: -

Any data in computer system can be represented in many kinds depending on its size. These kinds are:-

1- Bit (binary digit) and this can take only two state either zero (0) or one (1)
2- Nibble: this type consists of four bits and can take 16 state (2^4).
3- Byte: this consists of 8-bits and can taken 256 state (2^8).
4- Word: consists of 16-bits the computer recognizes and processes at a time.
5- Double word: consists of 32-bit .

Number Systems

For Microprocessors, information such as instruction, data and addresses are described with numbers. The types of numbers are not normally the decimal numbers we are familiar with; instead, binary and hexadecimal numbers are used. Table below shows Binary and Hexadecimal representations for some decimal numbers.

Decimal	Binary	Hexadecimal
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Example: Evaluate the 16-bit binary representation of decimal number107_{10}, then evaluate its hexadecimal representation

Solution:

$$\mathbf{107_{10} = 01101011_2 = 6B\ H}$$

Computer languages: -

- low- level language:-

Machine language: -

Is the set of instructions designed into the machine makes up its machine language, a binary language, composed of 0's and 1's, that is specific to each computer.

EX:-

0011 1100 :-

is an instruction that increments the number in the register called the accumulator by one.

1000 0000 :-

is an instruction that adds the number in the register called B to the number in the accumulator and keeps the sum in the accumulator.

Since it is difficult and tedious for people to recognize and write instruction in binary language, for convenience, these instructions are written in hexadecimal (or octal) code and entered in a single-board microcomputer by using Hex keys.

EX:-

The instruction	hexadecimal equivalent
0011 1100	3C
1000 0000	80

Assembly language: -

Even though the instructions can be written in hexadecimal codes, it is still difficult to understand a program written in hexadecimal numbers. Therefore, each manufacture of mp has devised a symbolic code for each instruction, called a mnemonic, which consists of letters to suggest the operation to be performed by that instruction.

EX: -

0011 1100 (3C H) is represented by the mnemonic INR A

1000 0000 (80 H) is represented as ADD B

- The machine language and assembly language are specific to each mp, and both are considered Low-level language.
- The machine language is in binary and the assembly language is English-like words, the mp understands only the binary. Then, how do the assembly language mnemonics get translated into binary code?

The translation is performed either manually called hand assembly, or by a program called an assembler.

Assembler: - A computer program that translates an assembly language program from mnemonics to the binary machine code of a computer.

- high- level language:-

Programming languages that are machine independent. Like Fortran, Basic and Pascal. Instructions written in this language are known as statements rather than mnemonics.
The Compiler and interpreter accept the English-like statements as their input called the (source code), they translate the source code into the machine language compatible with the mp being used in the system. This translation is called the (object code)

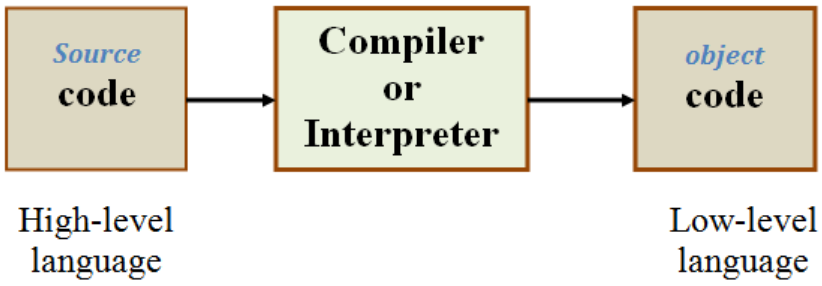


Fig-a- translation of high-level language program into machine code.

The difference between the compiler and the interpreter lies in the process of generating the object code.

a- The compiler reads the entire program and then generates the object code
The compilers are generally used in languages such as Fortran, Pascal.

b- The interpreter reads one instruction at a time, produce its object code, and executes the instruction before reading the next instruction
An interpreter used for the Basic language.

Microprocessor architecture and its operation: -

The mp is a programmable logic device, designed with registers, flip-flops, and timing elements. The mp has a set of instructions designed internally.

The mp performs primarily four operations: -

1- Memory read: read data from memory
2- Memory write: write data into memory
3- I/O read: accepts data from input device
4- I/O write: sends data to output device

All these operation are part of the communication process between the mp and peripheral devices (including memory).

To communicate with a peripheral or memory location the mp needs to perform the following steps: -

1- Identify the peripheral or the memory location (with it address)
2- Transfer data
3- Provide timing signals

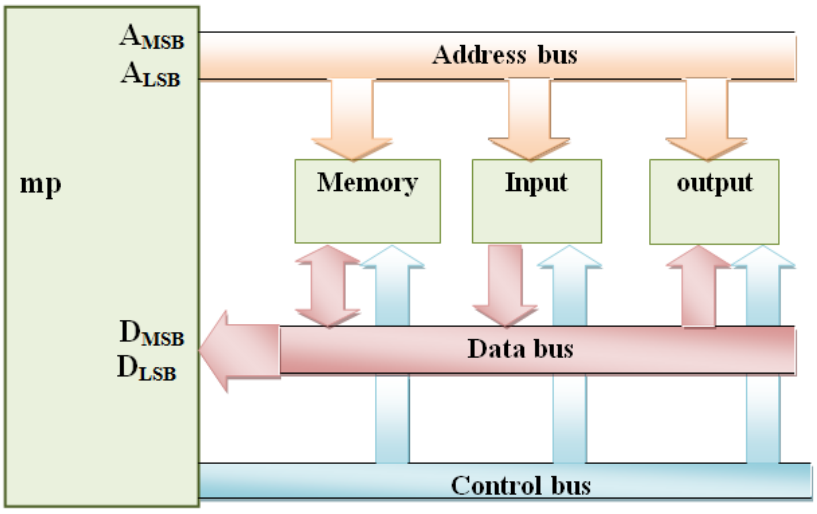
The mp performs these function using three sets of comm. line called buses: -
The address bus, the data bus and control bus.

- Address bus: - a group of lines that are used to send a memory address or device address from the mp to the memory location or the peripheral.

- Data bus: - a group of bi-directional lines which are used to transfer data between the mp and peripherals or memory

The data bus determines the word length and the register size of the mp.

- control bus: - single lines that are generated by the mp to provide controlling of various operations.



bus structure

To read an instruction from a memory location. The mp places the address on the address bus. The address on the bus is decoded and the memory is identified. The mp sends a pulse called memory read as the control signal. The pulse activates the memory chip and the contents of the memory location are placed on the data bus and brought inside the mp.