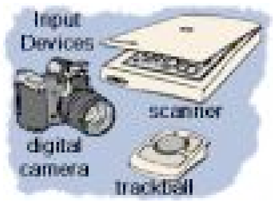
Notes - Computer Hardware Basics

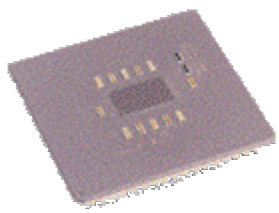
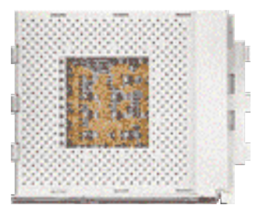
The Computer

A computer is made up of many parts:

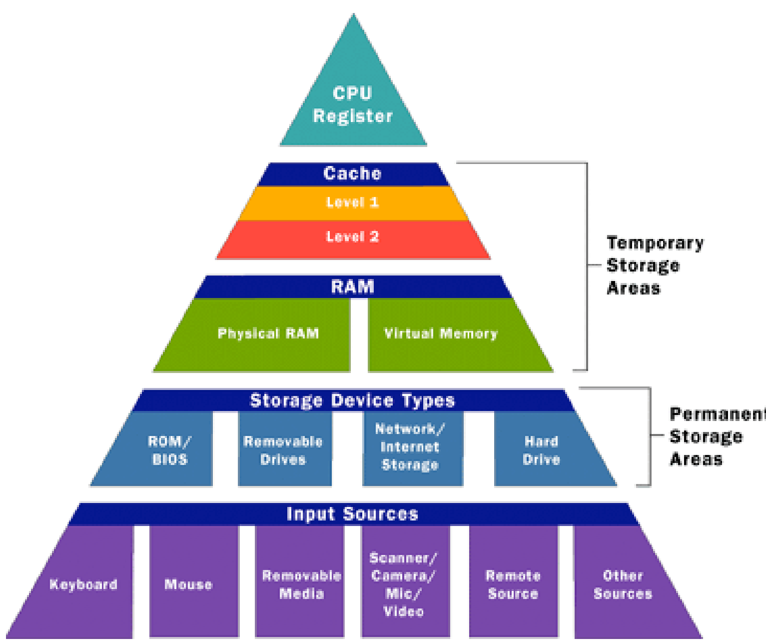
1. **Input/Output (I/O) devices** – These allow you to send information to the computer or get information from the computer.

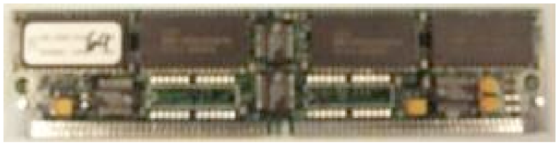

2. **Central Processing Unit** – CPU or Processor for short. The brain of a computer. Approximately 1.5 in X 1.5 in. Does all the computation/work for the computer.

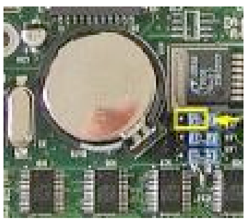
3. **Memory** – Although memory is technically any form of electronic storage, it is used most often to identify fast, temporary forms of storage. Accessing the hard drive for information takes time. When the information is kept in memory, the CPU can access it much more quickly.



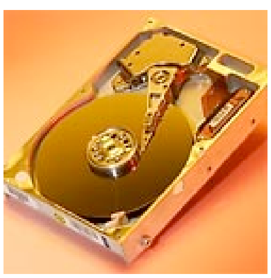
a. **Random Access Memory** – RAM. Where information is stored temporarily when a program is run. Information is automatically pulled into memory, we cannot control this. RAM is cleared automatically when the computer is shutdown or rebooted. RAM is volatile (non-permanent).



b. **Read Only Memory** – ROM. More permanent than RAM. Data stored in these chips is nonvolatile -- it is not lost when power is removed. Data stored in these chips is either unchangeable or requires a special operation to change. The BIOS is stored in the CMOS, read-only memory.

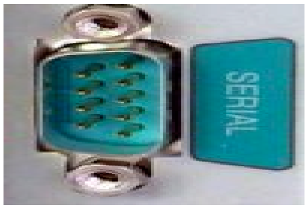


c. **Hard Drive** – Where you store information permanently most frequently. This is also nonvolatile.



4. **Motherboard** – A circuit board that allows the CPU to interact with other parts of the computer.

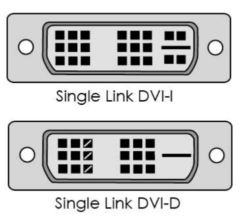
5. **Ports** – Means of connecting peripheral devices to your computer.
    a. **Serial Port** – Often used to connect a older mice, older external modems, older digital cameras, etc to the computer. The serial port has been replaced by USB in most cases. 9-pin connector. Small and short, often gray in color. Transmits data at 19 Kb/s.



    b. **Monitor Ports –** Used to connect a monitor to the computer.

PCs usually use a VGA (Video Graphics Array) analog connector (also known as a D-Sub connector) that has 15 pins in three rows. Typically blue in color.

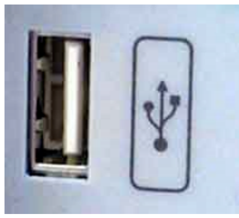
Because a VGA (analog) connector does not support the use of digital monitors, the Digital Video Interface (DVI) standard was developed.



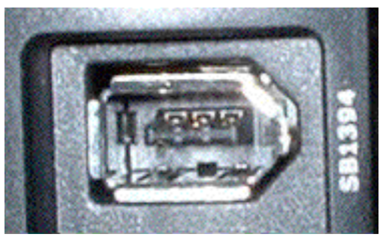
LCD monitors work in a digital mode and support the DVI format. At one time, a digital signal offered better image quality compared to analog technology. However, analog signal processing technology has improved over the years and the difference in quality is now minimal.

    c. **Parallel Port** – Most often used to connect a printer to the computer. 25-pin connector. Long and skinny, often pink in color. Transmits data at 50-100 Kb/s.

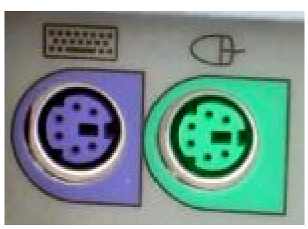

d. **USB Port** – Universal Serial Bus. Now used to connect almost all peripheral devices to the computer. USB 1.1 transmits data at 1.5 Mb/s at low speed, 12 Mb/s at full speed. USB 2.0 transmits data at 480 Mb/s.



e. **Firewire/ IEEE 1394 Port** – Often found on Apple Computers. Often used with digital camcorders. Firewire transmits data at 400 Mb/s. Firewire 1394B (the new firewire) transmits data at 3.2 Gb/s.



f. **PS/2 Port** - sometimes called a mouse port, was developed by IBM. It is used to connect a computer mouse or keyboard. Most computers come with two PS/2 ports.



g. **Ethernet Port** – This port is used for networking and fast internet connections. Data moves through them at speeds of either 10 megabits or 100 megabits or 1 gigabit (1,000 megabits) depending on what speed the network card in the computer supports. Little monitor lights on these devices flicker when in use.

6. **Power Supply** – Gives your computer power by converting alternating current (AC) supplied by the wall connection to direct current (DC).

7. **Expansion Cards –** Used to add/improve functionality to the computer.
   a. **Sound Card –** Used to input and output sound under program control. Sound cards provide better sound quality than the built in sound control provided with most computers.

   b. **Graphics Card –** Used to convert the logical representation of an image to a signal that can be used as input for a monitor.

   c. **Network Card** – Used to provide a computer connection over a network. Transmit data at 10/100/1000 Mb/s.

8. **CD ROM –** A device used to read CD-ROMs. If capable of writing to the CD-ROM, then they are usually referred to as a 'burner' or CD-RW.

9. **DVD ROM –** A device that is used to read DVDs/CDs. If capable of writing to the DVD, then it is often referred to as a DVD-burner or a DVD-RW.

10. **Floppy Drive –** A device that is used to read/write to floppy diskettes.


11. **Fan** – Keeps your computer cool. If the inside of your computer becomes too hot, then the computer can overheat and damage parts.

12. **Heatsink –** Used to disperse the heat that is produced inside the computer by the CPU and other parts by increasing surface area.

13. **The little parts** – Capacitors – store energy, Resistors – allows a current through, Transistors – a valve which allows currents to be turned on or off.

14. **Case** – (Tower if standing upright.) What your motherboard, CPU, etc is contained in.

The three main components of a computer:

1. **CPU** – Central Processing Unit, coordinates all actions that occur in the system, executes program instructions.
2. **Memory** – Used to store information.
3. **I/O Devices** – Input/Output devices, which allow you to obtain or display data.

Comprehension Questions:

1. **What are the 3 main components of a computer?**
2. **Name 3 input devices. Name 3 output devices.**
3. **What is the brain of the computer?**
4. **Explain the difference between memory and your hard drive.**
5. **What are the similarities and differences between RAM, ROM, and hard drives?**
6. **What allows the brain of the computer to interact with the other parts of the computer?**
7. **Describe each of the different ports and explain what they are used for.**
8. **What gives your computer power?**

The central processing unit(CPU).

The brain of any computer system is the CPU. It controls the functioning of the other units and process the data. The CPU is sometimes called the **processo**r, or in the personal computer field called “**microprocessor**”. It is a single integrated circuit that contains all the electronics needed to execute a program. The processor **calculates** (add, multiplies and so on), **performs logical operations** (compares numbers and make decisions), and **controls the transfer of data among devices**.

The processor acts as the controller of all actions or services provided by the system. Processor actions are synchronized to its clock input. A **clock signal** consists of *clock cycles*. The time to complete a clock cycle is called the *clock period*. Normally, we use the clock frequency, which is the inverse of the clock period, to specify the clock. The clock frequency is measured in Hertz, which represents one cycle/second. Hertz is abbreviated as Hz.

Usually, we use mega Hertz (MHz) and giga Hertz (GHz) as in 1.8 GHz Pentium. The processor can be thought of as executing the following cycle forever:

1. Fetch an instruction from the memory,
2. Decode the instruction (i.e., determine the instruction type),
3. Execute the instruction (i.e., perform the action specified by the instruction).

Execution of an instruction involves fetching any required operands, performing the specified operation, and writing the results back. This process is often referred to as the *fetch-execute* cycle, or simply the *execution* cycle. The execution cycle is repeated as long as there are more instructions to execute. This raises several questions. Who provides the instructions to the processor? Who places these instructions in the main memory? How does the processor know where in memory these instructions are located?

When we write programs—whether in a high-level language or in an assembly language— we provide a sequence of instructions to perform a particular task (i.e., solve a problem). A compiler or assembler will eventually translate these instructions to an equivalent sequence of machine language instructions that the processor understands. The operating system, which provides instructions to the processor whenever a user program is not executing, loads the user program into the main memory. The operating system then indicates the location of the user program to the processor and instructs it to execute the program.

The actions of the CPU during an execution cycle are defined by micro-orders issued by the control unit. These micro-orders are individual control signals sent over dedicated control lines. For example, let us assume that we want to execute an instruction that moves the contents of register **X** to register **Y**. Let us also assume that both registers are connected to the data bus, **D**. The control unit will issue a control signal to tell register **X** to place its contents on the data bus **D**. After some delay, another control signal will be sent to tell register **Y** to read from data bus **D**.

The components of CPU.

A typical CPU has three major components: **(1) register set**, **(2) arithmetic logic unit (ALU)**, and **(3) control unit (CU)**. The register set differs from one computer architecture to another. It is usually a combination of ***general-purpose and special purpose registers***. General-purpose registers are used for any purpose, hence the name general purpose. Special-purpose registers have specific functions within the CPU. For example, <u>the program counter (PC)</u> is a special-purpose register that is used to hold the address of the instruction to be executed next. Another example of special-purpose registers is the <u>instruction register (IR)</u>, which is used to hold the instruction that is currently executed. Figure 12 shows the main components of the CPU and its interactions with the memory system and the input/output devices.

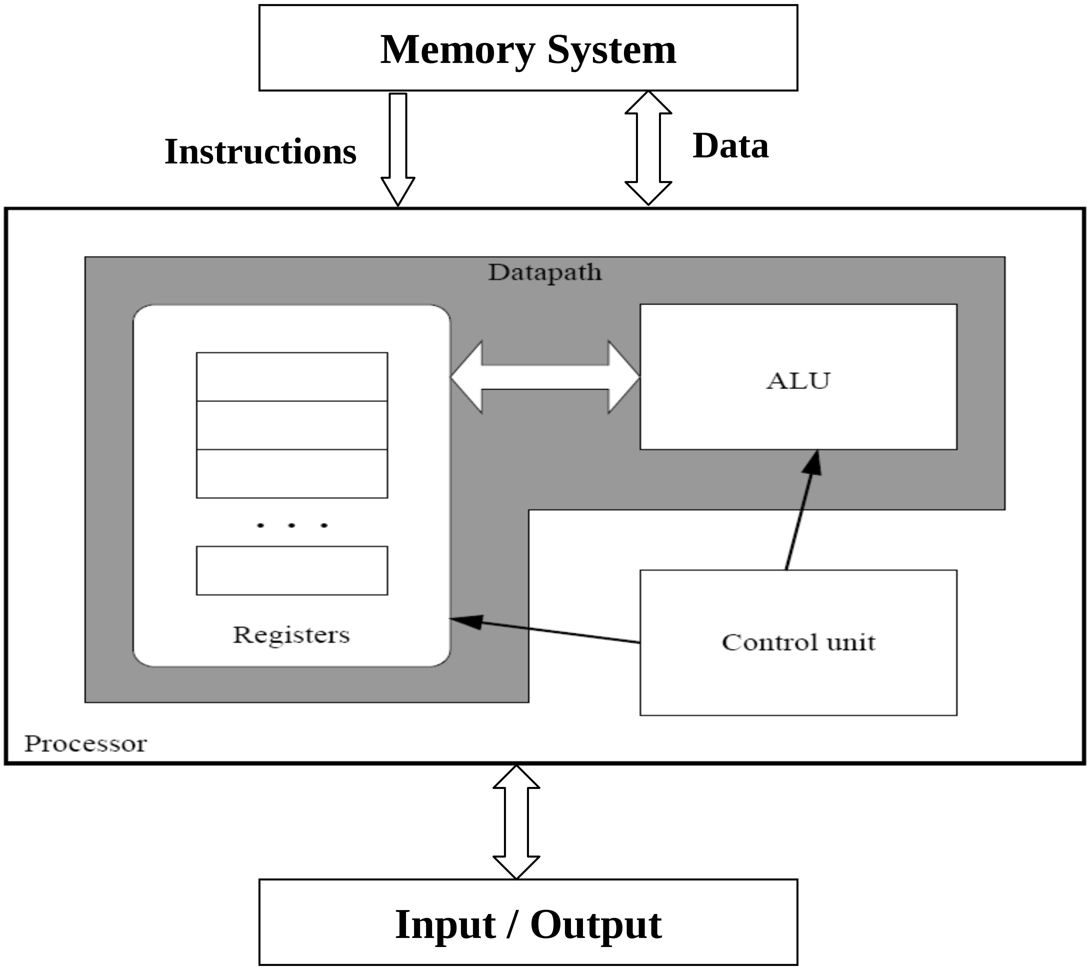


Figure 12: Central processing unit main components and interactions with the memory and I/O.

The ALU provides the circuitry needed to perform the arithmetic, logical and shift operations demanded of the instruction set. The control unit is the entity responsible for fetching the instruction to be executed from the main memory and decoding and then executing it.

The CPU can be divided into a data section and a control section. The data section, which is also called the datapath, contains the registers (known as the **register file)** and the ALU. The datapath is capable of performing certain operations on data items. The register file can be thought of as a small, fast memory, separate from the system memory, which is used for temporary storage during computation.

The control section is basically the control unit, which issues control signals to the datapath. The control unit of a computer is responsible for executing the program instructions, which are stored in the main memory. It can be thought of as a form of a “computer within a computer” in the sense that it makes decisions as to how the rest of the machine behaves.

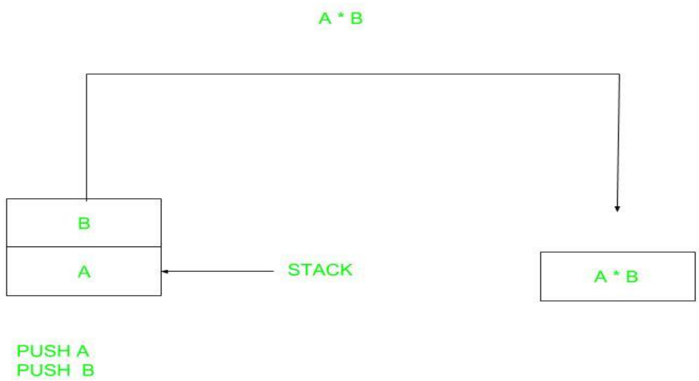
Like the system memory, each register in the register file is assigned an address in sequence starting from zero. These register “addresses” are much smaller than main memory addresses: a register file containing 32 registers would have only a 5-bit address, for example. The major differences between the register file and the system memory is that the register file is contained within the CPU, and is therefore much faster. An instruction that operates on data from the register file can often run ten times faster than the same instruction that operates on data in memory. For this reason, register-intensive programs are faster than the equivalent memory intensive programs, even if it takes more register operations to do the same tasks that would require fewer operations with the operands located in memory.

Instruction format:

On the basis of number of address instruction are classified as:

Note that we will use X = (A+B)*(C+D) expression to showcase the procedure.

1. **Zero Address Instructions –**



A stack based computer do not use address field in instruction.To evaluate a expression first it is converted to revere Polish Notation i.e. Post fix Notation.

Expression: X = (A+B)*(C+D)

Postfixed : X = AB+CD+*

TOP means top of stack

M[X] is any memory location

PUSH	A	TOP = A
PUSH	B	TOP = B
ADD		TOP = A+B
PUSH	C	TOP = C
PUSH	D	TOP = D
ADD		TOP = C+D
MUL		TOP = (C+D)*(A+B)

POP	X	M[X] = TOP

2. **One Address Instructions –**
This use a implied ACCUMULATOR register for data manipulation.One operand is in accumulator and other is in register or memory location.Implied means that the CPU already know that one operand is in accumulator so there is no need to specify it.

opcode	operand/address of operand	mode

Expression: X = (A+B)*(C+D)

AC is accumulator

M[] is any memory location

M[T] is temporary location

LOAD	A	AC = M[A]
ADD	B	AC = AC + M[B]
STORE	T	M[T] = AC
LOAD	C	AC = M[C]
ADD	D	AC = AC + M[D]
MUL	T	AC = AC * M[T]
STORE	X	M[X] = AC

3. **Two Address Instructions –**
This is common in commercial computers.Here two address can be specified in the instruction.Unlike earlier in one address instruction the result was stored in accumulator here result cab be stored at different location rather than just accumulator, but require more number of bit to represent address.

opcode	Destination address	Source address	mode

Here destination address can also contain operand.

Expression: X = (A+B)*(C+D)

R1, R2 are registers

M[] is any memory location

MOV	R1, A	R1 = M[A]
ADD	R1, B	R1 = R1 + M[B]
MOV	R2, C	R2 = C
ADD	R2, D	R2 = R2 + D
MUL	R1, R2	R1 = R1 * R2
MOV	X, R1	M[X] = R1

4. **Three Address Instructions –**
   This has three address field to specify a register or a memory location. Program created are much short in size but number of bits per instruction increase. These instructions make creation of program much easier but it does not mean that program will run much faster because now instruction only contain more information but each micro operation (changing content of register, loading address in address bus etc.) will be performed in one cycle only.

opcode	Destination address	Source address	Source address	mode

Expression: X = (A+B)*(C+D)

R1, R2 are registers

M[] is any memory location

ADD	R1, A, B	R1 = M[A] + M[B]

ADD	R2, C, D	R2 = M[C] + M[D]
MUL	X, R1, R2	M[X] = R1 * R2