Date:

# Application Of Information and Communication Technologies

1) Computer
2) Advantages — Disadvantages

| Advantages | Disadvantages |
|---|---|
| Speed | Privacy |
| Consistency | Public Safety |
| Reliability | Health risk |
| Storage | Impact on Labor |
| Communication | Impact on Environment |

3) Hardware / Software
4) Input Devices
5) CPU
6) Primary Storage
7) Types of Memory
- MROM (Mask ROM)
- PROM (Programmable ROM)
- EPROM (Erasable Programmable ROM)
- EEPROM (Electrically Erasable Programmable ROM)

8) RAM

| • DRAM (Static RAM) | • SRAM (Dynamic) |
|---|---|
| 1) Constructed of tiny capacitors that leak electricity. | • Constructed of circuits similar to D flip-flops. |
| 2) Requires a recharge after few $10^{-3}$s to maintain data. | • Holds data as long as power is available |
| 3) Inexpensive | • Expensive |
| 4) Slower than SRAM | • Faster than DRAM. |

Date:

| DRAM | SRAM |
|---|---|
| • can store many bits per chip | • Can't store many bits per chip. |
| • Uses less power | • Uses more power |
| • Generates less heat | • Generates more heat |
| • Used for main memory | • Used for cache. |

9) Secondary Storage
- Hard Disk Drive (HDD)
- Solid State Drive (SSD)

10) Interfaces

The connecting part between the computer and the hard drive.

* Types of interfaces:
- Parallel Advanced Technology Attachment (PATA)
- Serial ATA (SATA)
- Small Computer System Interface (SCSI)
- Non-voltalile Memory Express (NVMe)

11) Output Devices

12) System Unit

The main part of a micro computer. Sometimes called the chassis.

⇒ It includes following parts:
- Motherboard
- Microprocessor
- Busses
- Ports
- Expansion Slots
- Cards

13) Computer

Analog Computer
Hybrid Computer
Digital Computer
- Super Computer
- Mainframe Computer
- Mini Computer
- Micro Computer

14) Algorithm

An algorithm is an ordered step set of unambiguous, executable steps that define a terminating process.

15) Euclidean Algorithm

This algorithm assumes that its input consists of two +ve integers and proceeds to compute the greatest common divisors of these two values.

S1: Assign M and N the value of the larger and smaller of the two input values, respectively.

S2: Divide M by N and call the Remainder R.

S3: If R is not 0, then assign M the value of N, assign N the value of R, and return to S2; otherwise, the GCD is the value currently assigned to N.

Date:

## 16) Polya's Problem Solving Steps

1) Understand the problem.
2) Devise a plan for solving the problem.
3) Carry out the plan.
4) Evaluate the solution for accuracy and its potential as a tool for solving other problems.

## 17) Polya's Steps in context of Program Development

- Understand the problem.
- Get an idea of how an algorithmic function might solve the problem.
- Formulate the algorithm and represent it as a program.
- Evaluate the solution for accuracy and its potential as a tool for solving other problems.

## 18) Characteristics of Algorithm

- Finiteness
- Input
- Output
- Definiteness

  Each step must be clear, well-defined and precise. No ambiguity
- Effectiveness

  Each step must be simple and should take finite time.

## 19) Abstraction

Abstraction is a simplification technique.

- extract the external properties for components.
- ignore the internal details.

Use abstract modules to construct more complex function.

⇒ The process of filtering out - ignoring the characteristics of patterns that we don't need in order to concentrate on those that we do.

Date:

# Number System

| System | Base | Range |
|---|---|---|
| Decimal | 10 | 0, 1, ...., 9 |
| Binary | 2 | 0, 1 |
| Octal | 8 | 0, 1, ...., 7 |
| Hexa decimal | 16 | 0, 1, .... 9, A, B, ... F |

## Binary ⟶ Decimal

- Multiply each bit by $2^n$, where 'n' is the weight of the bit which is the position of bit starting from 0 on the right.

Example

$(101011)_2 \Rightarrow$

$1 \times 2^0 = 1$

$1 \times 2^1 = 2$

$0 \times 2^2 = 0$

$1 \times 2^3 = 8$

$0 \times 2^4 = 0$

$1 \times 2^5 = +\ 32$

$(43)_{10}$

## Octal ⟶ Decimal

- Multiply each bit by $8^n$, where n starts from 0 on the right position.

Example

$(724)_8 \Rightarrow$

$4 \times 8^0 = 4$

$2 \times 8^1 = 16$

$7 \times 8^2 = +\ 448$

$(468)_{10}$

## Hexadecimal ⟶ Decimal

- Multiply each bit by $16^n$

Example

$(ABC)_{16}$

$C \times 16^0 = 12 \times 1 = 12$

$B \times 16^1 = 11 \times 16 = 176$

$A \times 16^2 = 10 \times 256 = +\ 2560$

$(2748)_{10}$

## Decimal ⟶ Binary

- Divide by 2, keep track of the remainder

Example

$(125)_{10} = (1111101)_2$

| | | |
|---|---|---|
| 2 | 125 | |
| 2 | 62 | 1 |
| 2 | 31 | 0 |
| 2 | 15 | 1 |
| 2 | 7 | 1 |
| 2 | 3 | 1 |
| | 1 | 1 |

Date:

## Octal ⟶ Binary

- Convert each octal bit to a 3-bit equivalent binary representation.

Example

$(705)_8 = (111000101)_2$

| 7 | 0 | 5 |
|---|---|---|
| ↓ | ↓ | ↓ |
| 111 | 000 | 101 |

$(111\ 000\ 101)_2$

## Hexadecimal ⟶ Binary

- Convert each hexadecimal digit to a 4-bit equivalent binary representation.

Example

$(10AF)_{16} = (0001000010101111)_2$

| 1 | 0 | A | F |
|---|---|---|---|
| ↓ | ↓ | ↓ | ↓ |
| 0001 | 0000 | 1010 | 1111 |

$(0001\ 0000\ 1010\ 1111)_2$

## Decimal ⟶ Octal

- Divide by 8, keep track of the remainder

Example

$(1234)_{10} = (2322)_8$

| | |
|---|---|
| 8 | 1234 |
| 8 | 154 – 2 |
| 8 | 19 – 2 |
| | 2 – 3 |

## Decimal ⟶ Hexadecimal

- Divide by 16, keep track of the remainder.

Example

$(1234)_{10} = (4D2)_{16}$

| | |
|---|---|
| 16 | 1234 |
| 16 | 77 — 2 |
| | 4 — 13 = D |

4 D 2

Date:

## Binary ⟶ Octal

- Group bits in three, starting on right and then convert to octal digits.

Example

$(1011010111)_2 = (1327)_8$

1 011 010 111
↓ ↓ ↓ ↓
1 3 2 7

## Binary ⟶ Hexadecimal

- Group bits in fours, starting on right and then convert to hexadecimal digits.

Example

$(1010111011)_2 = (2BB)_{16}$

10 1011 1011
↓ ↓ ↓
2 B 11 = B

## Octal ⟶ Hexadecimal

- First convert each digit of the octal number into 3-bit binary number then pair up the binary digits in the groups of 4-bits then simply convert the 4-bits groups into the hexadecimal digits.

Example

$(213)_8$

2 1 3
↓ ↓ ↓
010 001 011

~~0~~ 1000 1011
↓ ↓
8 11 = B

Thus,

$(213)_8 = (8B)_{16}$

## Hexadecimal ⟶ Octal

- First convert each digit of octal in 4-bit binary then make pairs of 3-bit binary digits and then convert them in decimal.

Example

$(8B)_{16}$

8 B (11)
↓ ↓
1000 1011

Date:

010 001 011

↓ ↓ ↓

2 1 3

Thus,

$(8B)_{16} = (213)_8$

## Negative ⟶ Binary

First convert the positive decimal number into binary and place zero at left side then copy the binary number's digits from right side as it is else you find 1. As soon 1 encounters, change the rest of bits.

Example

$(15)_{10} = (1111)_2$

$+15 = 01111$

$-15 = (10001)_2$

## 1's complement

1's complement is simply the inverse of bits of the given binary number.

Example

$(15)_{10} = (1111)_2$

1's complement is given as

$\sim 15 = (0000)_2$

## Binary Addition

The two binary numbers can be added and also other operations can be carried out on them.

Example

| | | |
|---|---|---|
| 1 0 1 0 | = | 10 |
| + 1 0 0 1 | = | 9 |
| 1 0 0 1 1 | = | 19 |

* In case of 1 + 1, the answer is zero with 1 transfered to the next bit.

## Binary Subtraction

The subtraction of two binary numbers is also possible and is given as

| | | | |
|---|---|---|---|
| 1 | 1 | 0 | $\cancel{1}^{0}$ $0^{2}$ |
| - 1 | - 0 | - 0 | - 1 |
| 0 | 1 | 0 | 01 |

Date:

**Example**

$$\begin{array}{ccc} 1 & 0 & 0 \\ - & & 1 \\ \hline 0 & 1 & 1 \end{array} \quad \begin{array}{c} = 4 \\ = -1 \\ = 3 \end{array}$$

## 2's complement ⟶ Binary

Check the most significant (left) bit, if it is 1 then the term is (−) if 0 then the number is (+). To convert to binary just take 2's complement again of the given 2's complement.

**Example**

~ 10110000

Binary 01010000

**OR**

10110000

Take 1's complement

01001111

Add 1

01001111

\+ 1

$\boxed{01010000}$

## Decimal ⟶ IEEE 754 Floating.

- First convert the decimal part simply in binary.
- Then multiply the decimal point number by 2 till you encounter 1.00 or 0.000 situation.
- Take the decimal point to the left most significant number's right side and raise the no. of terms jumped to the power of 2.
- Bias the exponent of 2 by adding 127 then convert the resulting number to binary.
- Now represent the resulting number in binary as

x xxxxxxxx xxxxxxxxxxxxxxxxxxxxxxx

(sign) (exponent) (mantissa)

**Example**

15.375

I)

| | | |
|---|---|---|
| 2 | 15 | |
| 2 | 7 | − 1 |
| 2 | 3 | − 1 |
| | 1 | − 1 |

$(15)_{10} = (1111)_2$

Date:

II)

| | | |
|---|---|---|
| 0.375 × 2 | 0.750 | 0 |
| 0.750 × 2 | 1.500 | 1 |
| 1.500 × 2 | 1.000 | 1 |

Thus,

0.375 = 011

III)

01111 . 011

$01.111011 \times 2^{3}$

* Bias the exponent by adding 127 to it

$\underbrace{0}_{sign}1.\underbrace{111011}_{mantissa} \times 2^{130}$

* Convert the exponent to binary

| | | |
|---|---|---|
| 2 | 130 | |
| 2 | 65 | 0 |
| 2 | 32 | 1 |
| 2 | 16 | 0 |
| 2 | 8 | 0 |
| 2 | 4 | 0 |
| 2 | 2 | 0 |
| | 1 | 0 |

$(130)_{10} = \underbrace{(10000010)_2}_{exponent}$

IV) Final representation is given as :

x xxxxxxxx xxxxxxxxxxxxxxxxxxxxxxx

0 10000010 11101100000000000000000

IEEE 754 Floating ⟶ Decimal

- Check the sign bit if its 0 (+) or 1 (-).
- Convert the next eight bits to decimal the subtract 127 from the resulting decimal number.
- Convert the next 23 bits to decimal by multiply each with 2 to the negative power starting from 1 to the right side of the mantissa bits.
- Use the given formula to find the resulting IEE 754 floating point number's binary representation.

$(-1)^s \times (1+m) \times 2^e$

s ⟶ sign bit

e ⟶ exponent

Example

0100 0011 0101 0100 0000 0000 0000 0000

I] Sign bit

0 ⟶ number is +ve

II] Exponent bits

$= (10000110)_2$

$= 2^7 \times 1 + \ldots\ldots + 2^2 \times 1 + 2^1 \times 1 + 2^0 \times 0$

$= 128 + 4 + 2$

$= 134$

CamScanner

Date:

$e = 134 - 127 = 7$

III] Mantissa

$(1010100\ 0000\ 0000\ 0000\ 0000)_2$

$m = 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 0 \times 2^{-4} + 1 \times 2^{-5} + \ldots\ldots$

$m = 2^{-1} + 0 + 2^{-3} + 0 + 2^{-5}$

$m = 2^{-1} + 2^{-3} + 2^{-5}$

$m = 0.65625$

IV] Formula

$= (-1)^{s} \times (1 + m) \times 2^{e}$

$= (-1)^{0} \times (1 + 0.65625) \times 2^{7}$

$= 1.65625 \times 2^{7}$

$= 212$

CamScanner

Date:

# Computer Networking

## 1) Uses

Text messaging
Picture messaging
Video messaging

## 2) GPS

Global Positioning System (GPS) is a navigation system that consists of one or more earth-based receivers that accept and analyze signals sent by satellites in order to determine the GPS receiver's geographic location.

⇒ GPS receivers are:
- Built into mobile devices
- Available as handheld device
- Available with new vehicles.

## 3) Network

⇒ A network is a collection of computers and devices connected together via communication devices and transmission media.

⇒ Advantages:
- Facilitating communications
- Sharing Hardware
- Sharing Data & Information
- Sharing software

## 4) PAN

⇒ Personal Area Network (PAN) connects electronic devices within a users immediate area.

⇒ Connection b/w a bluetooth earpiece and a smartphone.

## 5) LAN

⇒ Local Area Network (LAN) is a network that connects computers and devices in a limited geographical area.

⇒ Two or more computers connected in a computer lab.

## 6) WLAN

⇒ Wireless LAN is a LAN that uses no physical wires.

⇒ A home or office Wi-Fi network

## 7) MAN

⇒ Metropolitan Area Network (MAN) connects LANS in a metropolitan area.

e.g. A local telephone company or local TV cable company.

Date:

8) **WAN**

Wide Area Network (WAN) is a network that covers a large geographical area

e.g. The world's largest WAN is the internet.

9) **Open Network**

- The term open networking describes a network that uses open standards and commodity hardware. Open network internal operations is based on designs that are in Public domain.

e.g. Internet

10) **Close Network**

- A closed network can refer to a private network that has no external connectivity.
- Close Network innovations owned and controlled by a particular entity such as an individual or a corporation.

e.g. Microsoft Cooporation.

11) **Point Server Network**

- A point server is a dedicated appliance or central point of software. They help clients connect to shared printers and they process or pass through spool files from a client to the printer.
- When you hit print, the point Server accepts the job and pushes it to the appropriate printer.

12) **Client Server Network**

- Client/Server network is designed for the end users called clients, to access the resources such as songs, videos, etc. from a central computer called Server.
- Clients communicate with each other through a server.

13) **Peer-to-Peer Network**

- Peer-to-Peer is a Network in which all the computers are linked with equal privilege and responsibilities for processing the data.
- P2P describes an Internet network on which users

Date:

access each other's hard disks and exchange files directly over the internet.

- P2P network has no dedicated server.

14) **Network Topology**

- A network topology refers to the layout of the computers and devices in a communication network.

15) **Bus Topology**

- In Bus topology, the machines are all connected to a common communication line called a bus.
- The bus cable is considered as a "single lane" through which the message is broadcast to all the stations.

16) **Ring Topology**

- Ring topology is like a bus topology, but with connected ends.
- The node that receives the message from the previous computer will retransmit it to the next code.

**Token Passing:** is a network access method in which token is passed from one node to another node.

**Token:** is a frame that circulates around the network.

17) **Protocols**

- The set of rules which is used for a network to function reliably is known as Protocol.

e.g. FTP, HTTP

18) **CSMA/CD**

- Carrier Sense Multiple Access with Collision Detection (CSMA/CD) is a medium access control method. It uses carrier sensing to defer transmissions until no other stations are transmitting.
- This used with collision detection in which a transmitting station detects collisions by sensing transmissions from other stations while it is transmitting a frame.
- If a collision is detected, the station stops transmitting the frame and waits for a random time interval before trying to resend the frame.

Date:

- CSMA/CD is not suitable for Star networks. As star network is wireless and CSMA/CD is primarily designed for wired ethernet networks and thus it's not used in wireless communication protocols.
- Wireless networks face challenges such as Hidden node problems, & signal interference which make collision detection less effective.

## 19) CSMA/CA

- Carrier Sense Multiple Access with Collision Avoidence (CSMA/CA) is a network multiple access method in which carrier sensing is used, but nodes attempt to avoid collisions by beginning transmission only after the channel is sensed to be "idle". When they do transmit, nodes transmit their packet data in its entirety.
- The nodes which has to send a packet to the other channel first checks the channel. If channel is not clear, the node waits for a randomly chosen period of time, and then checks again to see if the channel is clear. This period of time is called the backoff factor.
- If the channel is not clear when the backoff counter reaches zero, the backoff factor is set again, and the process is repeated.

## 20) Bridge

- It connect two buses, but it does not necessarily pass all messages accross the connection. Instead, it looks at destination address that accompanies each msg and forwards a message across the connection only when that msg is destinated for a computer on other side.

## 21) Repeater

- It connects two buses and passes all messages across the connection. It sends the destinated message also to the other computers to which that message is not destinated. In this way a message is sent to all

Date:

computers in the network.

## Switch

Switch is essentially a bridge with multiple connections, allowing it to connect several buses rather than just two. Produces a network consisting consisting of several buses. A switch considers the destination addresses of all messages and forwards only those messages destined for other spokes.

## Limitations of Bridge, Repeater and Switch

- Cannot be used with different types of networks.
- Cannot be used to combine a bus network with star network

## Routers

- Handle connection b/w networks to form an internet.
- Special purpose computers used for forwarding messages.
- Provide links b/w networks while allowing each network to maintain its internal characteristics.
- Decides routes
- Forwarding table.

## Distributed Softwares

- Global information retrieval systems
- Company wide accounting & inventory
- Computer Games
- Software controlling network infrastructure.

## Cluster Computing

- Cluster computing is a collection of tightly or loosely connected computers that work together so that they can act as a single entity. The connected computers execute operations all together thus creating the idea of a single system. The clusters are generally connected through fast LANs.
- Advantages

* Low maintenance cost
* High Availability * Load Balancing

## Grid Computing

- Grid computing is a computing infrastructure that combines computer resources spread

Date:

over different geographical locations to achieve a common goal. All unused resources on multiple computers are pooled together and made available for a single task.

- Services of these computers can be used when they are free.

e.g. Condor System, Berkeley's Open Infrastructure for Network Computing (BOINC).

## Cloud Computing

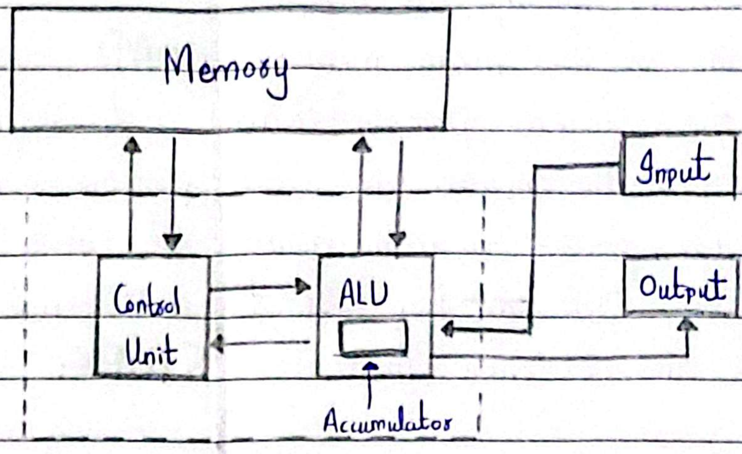
Cloud computing is the on-demand delievery of computer resources over Internet with pay-as-you-go pricing. Insted of buying, owning and maintaining physical data centres and servers, you can access technology services, such as computing power, storage and databases.

e.g. Amazon's Elastic Compute Cloud.
Google Cloud
Microsoft Azure
Amazon Web Services (AWS)

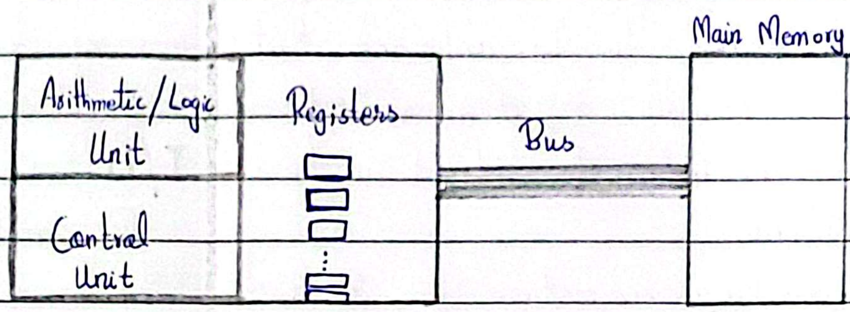
Date:

# Data Manipulation

## Von Neuman Basic Structure




## CPU & Main Memory connected via bus




## Machine Language

CPUs are designed to recognize instructions encoded as bit patterns. This collection of instructions along with the encoding system is called Machine Language.

## Machine Instruction

An instruction encoded as a bit pattern recognizable by the CPU is machine instruction.

## General Purpose Register

General Purpose Registers serve as temporary holding places for data being manipulated by CPU. These registers hold inputs to the ALU's circuitry and provide storage space for results produced by that unit.

Date:

## Stored Program concept

The idea of storing a computer's program in its main memory is called the Stored program concept. A program can be encoded as bit patterns and stored in main memory. From there, CPU can then extract instructions and execute them.

**RISC**

- Reduced Instruction Set Computer (RISC) leads to the approach that a CPU should be designed to execute a minimal set of machine instructions.
- A machine according to RISC architecture is efficient, fast, and less expensive to manufacture.

e.g. POWERPC
Apple / IBM / Motorola

**CISC**

- Complex Instruction Set Computer (CISC) leads to the approach that CPUs should have ability to execute a large number of complex instruction.
- A machine according to CISC approach has more complex CPU which can better cope with complexities.

e.g. Intel

**LOAD**

A request to fill a general-purpose register with the contents of a memory cell is called LOAD instruction.

**STORE**

A request to transfer the contents of a register to a memory cell is called a STORE instruction.

**I/O**

The instructions handling input/output activities of machine are called I/O instructions.

⇒ The encoded version of a machine instruction consists of two parts:

**Op-code**

The bit pattern appearing in the op-code field indicates which of the elementary operations, such as STORE, SHIFT, XOR,

Date:

and JUMP is requested by the instruction.

## Operand

The bit patterns found in the operand field provide more detailed information about the operation specified by the op-code.

e.g. In case of a STORE operation, the information in the operand field indicates which register contains the data to be stored and which memory cell is to receive the data.

→ Each of the machine instruction containing op-code and operand is encoded using 16 bits, represented by 4 hexadecimal digits.

→ The op-code for each instruction consists of first 4 bits or equivalently first hexadecimal digit.

**Example**

3 5 A7

3 → Store, 5 → in register 5, A7 → from memory cell A7

$(0011\ 0101\ 1010\ 0111)_2$

| | |
|---|---|
| 1 5 6C | LOAD Register 5 with data found in memory cell at address 6C |
| 5056 | ADD contentents of register 5 and 6 and leave result in register 0. |
| 3 0 6E | STORE contents of register 0 in memory cell at 6E. |
| C000 | Halt. |

## Instruction Register

The instruction register is used to hold the instruction being executed.

## Program Counter

The program counter contains the address of the next instruction to be executed.

## Fetch

- During the Fetch step, the CPU requests the main memory to provide it with the instruction that is stored at the address indicated by the program counter.
- The CPU places the instruction received from main memory in

Date:

its instruction register and increments the program counter by 2 so that the counter contains the address of next instruction stored in memory.

## Decode

With the instruction, now in instruction register, the CPU decodes the instruction, which involves breaking the operand field into its proper components based on the instruction's op-code.

## Execute

The CPU then executes the instruction by activating the appropriate circuitry to perform the requested task.

## JUMP Instruction

- The op-code 'B' represents the JUMP instruction. Thus, the instruction B258 means " JUMP to the instruction at address 58 if the contents of register 2 is the same as that of register 0"

B 2 58

B — change the value of the program counter if the contents of indicated register is sames as that of register 0

2 — Register to be compared with register 0

58 → Address to be placed in program counter if the contents of two registers is same

## Mask

- Masking is a process of applying a mask to a value and it can be achieved by using bitwise operators AND, OR, XOR.

## AND

- ANDing means to extract a subset of the bits from the given value.

## OR

- ORing means to set a subset of the bits in the given value.

### Example

* To make sure a bit is on we will OR it with 1
* To make a bit unchanged we will OR it with 0.

10011010

⇒ Turn ON the higher nibble of the given byte

To do this we will OR it with mask 1 to all bits of higher nibble and will OR it with mask 0 to all bits of lower nibble.

Date:

| | | |
|---|---|---|
| 1001 1010 | value | |
| 1111 0000 | mask | |
| 1111 1010 | result | |

$\underbrace{1111}_{\text{higher nibble turned ON}}\ \underbrace{1010}_{\text{lower nibble unchanged}}$

## XOR

- XORing means to toggle a subset of the bits in the given value.
- XOR is used to toggle a subset of bits in the given set of bits.
- The bits we want to toggle are masked with 1 and the bits we want to remain unchanged are masked with 0.

Example

10011010

⇒ Toggle the lower nibble of the given byte

| | |
|---|---|
| 10011010 | value |
| 0000 1111 | mask |
| 1001 0101 | result |

higher nibble unchanged — low nibble toggled

Example — AND

- ANDing means to extract a subset of bits from the given set of bits.
- A bit ANDed with 0 will be 0 and a bit ANDed with 1 will be unchanged.
- To extract the bits from the set of bits we have to AND/mask that bits with 0.

10011010

⇒ Extract the lower nibble from the given byte

| | |
|---|---|
| 1001 1010 | value |
| 0000 1111 | mask |
| 0000 1010 | |

higher nibble masked off — lower nibble extracted/unchanged

Date:

# Circular Shift / Rotation

- It circulates the bits of the register around the two ends without the loss of information.
- Consider shifting the bits of a register to the right by one bit. Then we imagine the right-most bit will fall off or will be discarded ~~then~~ and a hole will be generated at the left most side. In this case the rightmost discarded bit will be filled in the hole at leftmost side.

Example — Right Shift

1 0 0 1 1 0 0 0

0 1 0 0 1 1 0 0

* If we perform a right circular shift on a byte-size bit pattern eight times, we obtain the same bit pattern.

Example — Circular Left Shift

1 0 0 1 1 0 0 0

0 0 1 1 0 0 0 1

## Arithmetic Shift

### Arithmetic left Shift

- Shifting bits to the left means multiplying a signed binary number by 2.
- In Arithmetic left shift, we will shift all the bits to the left side with the bit at most significant place or we can say that the Most Significant Bit (MSB) will be discarded and LSB will be filled with 0.

Example

MSB Signed bit LSB

1 1 1 1 0 1 1 0 → -10

1 1 1 0 1 1 0 0 → -20

1 1 0 1 1 0 0 0 → -40

Date:

## Arithmetic Right Shift

- Shifting bits to the right means dividing the decimal number by 2.
- In Arithmetic Right Shift, all the bits will be shifted towards the right side with the **MSB** will be discarded.
- Also the **LSB** will be copied as it is as will also be shifted towards right side.

Example

MSB LSB

0 0 0 0 1 0 1 ~~0~~ 10

0 0 0 0 0 1 0 ~~1~~ 5

0 0 0 0 0 0 1 0 2

## Logical Shift

### Logical Left Shift

- In logical left shift, all the bits are shifted towards the left side.
- In this case, the MSB will be discarded and the LSB will be filled with 0.

Example

MSB LSB

~~1~~ 0 1 0 1 1 0 1

0 1 0 1 1 0 1 0

MSB LSB

~~0~~ 1 0 1

1 0 1 0

### Logical Right Shift

- In logical right shift, all the bits are shifted towards the right side.
- In this case, the LSB will be rejected and the MSB will be filled with 0.

Example

MSB LSB

1 0 1 0 1 1 0 ~~1~~

0 1 0 1 0 1 1 0

1 0 1 ~~1~~

0 1 0 1

Date:

## Controller

- Communication b/w a computer and other devices is normally handled through an intermediary apparatus known as a Controller.

## Port

- The controller connects via cables to peripheral devices within the computer case or perphaps to a connector called a port, on the back of the back of the computer where externel devices can be attached.

⇒ There are two types of Controller.

1. Specialized controllers for each type of device.
2. General purpose controllers (USB and FireWire).

## MMIO

Memory-Mapped I/O is a complementary method of performing I/O b/w the CPU and peripheral devices in a computer. CPU communicates with peripheral devices in a way as they are memory cells.

## DMA

- Since a controller is attached to a computer's bus, it can carry on its own communication with main memory during those nanosecs in which the CPU is not using the bus. This ability of controller to access main memory is called Direct Memory Access (DMA).

## Handshaking

- A constant two way communication or dialogue in which the computer and pheripheral devices exchange information about device's status and coordinate their activities is known as Handshaking.

## Parallel Communication

- In case of Parallel communication, several signals are transfered at the same time, each on a separate "line". Such a technique is capable of transferring data rapidly but requires a complex communication pat

Example

Computer's internal bus.

Date:

## Serial Communication

- Serial Communication is based on transferring signals one after the other over a single line. Thus, it requires a simpler data path.

Example

USB and FireWire

## Bandwidth

- A measurement indicating the maximum capacity of a wired or wireless communication links to transmit data over a network connection in a given amount of time. Normally measured in bits, kilo bits, megabits or gigabits per sec.

## Parallel Processing

- The use of multiple processors to divide the workload and minimize the computation time of a monotonous process.

Date:

# Operating System

## System Software

- System software consists of the programs that control or maintain the operations of the computer and its devices.
  - ⇒ OS
  - ⇒ Utility Programs

## OS

- An OS is a set of programs containing instructions that work together to coordinate all the activities among computer hardware resources.
  - * Manage memory
  - * Administer Security
  - * Start and shut down of comp

## User Interface

- In order to performs the actions requested by the computer's users, an OS must be able to communicate with those users. The portion of OS which handles this communication is called User Interface.
  - ⇒ Text Based (Shell)
  - ⇒ GUI

## Kernel

- In contrast to the operating system's user interface, the internal part of the OS is called Kernel. An OS's kernel contains those software components that perform the very basic functions required by the computer installation.

  such as File Manager, Memory Manager, Scheduler/Dispatcher

## Booting

- The process of starting or restarting the comp. is called Booting.

## Cold Boot

- The process of turning on a comp. that has been powered off completely is called Cold Boot.

## Warm Boot

- Using the OS to restart the comp. is Warm Boot.