No: Date: __/__/____

01. A) Generations of the Computers

① 1st Gen (1944 - 1955)
② 2nd Gen (1955 - 1964)
③ 3rd Gen (1964 - 1971)
④ 4th Gen (1971 - Present)
⑤ 5th Gen (Present & Beyond)

1st Gen

- Thermion valves
- Large, Heavy
- Large Power Consumption
- Machine Language (1, 0) - switches
- Programing - Hard & Slow
- Expensive
  ↳ Large Electricity ✓
  ↓
  Lot of Heat ✓
  ↓
  Malfunctions ✓

ENIAC - Electronic Numerical Integrator & Calculator
↳ First electronic general purpose calculator (1946)
⇒ 5000 operations per sec
& Used Hard wired (switching programing) progoamming

2nd Gen

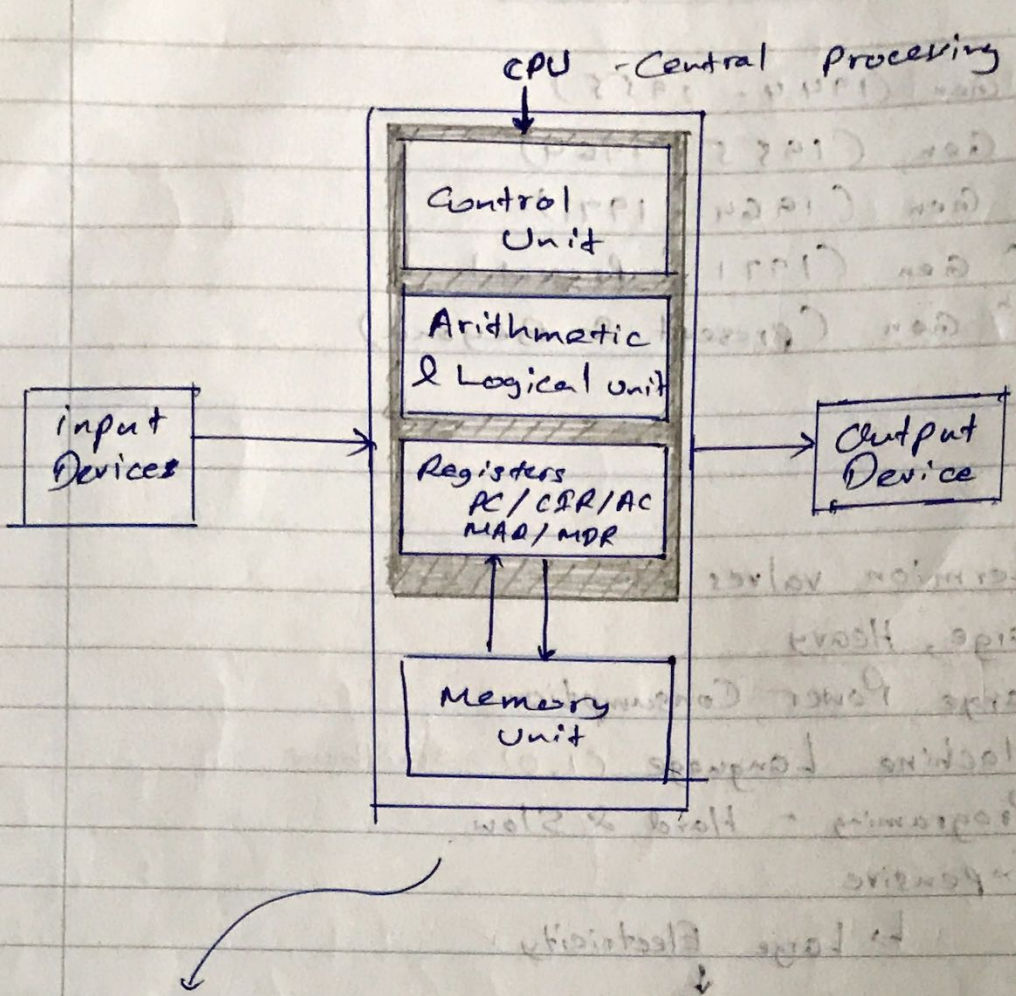
1st Gen → Von Neumann Architecture → 2nd Gen

memory was introduced
↓
and it avoided Re-wiring (@ Hard wired progaming)

· And the invention of Transistor

No: Date: / /

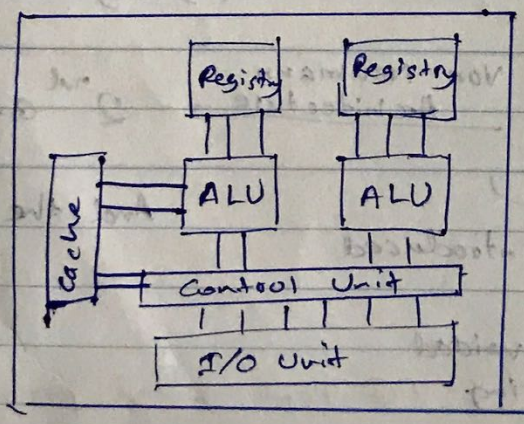
Von Neumann Archi



This same architecture follows every computer from mainframe to plamtop (even smart phones)

- memory
- I/O System (Input / output)
- arithmatic & logical unit.
- Control Unit

CPU:



No: Date: / /

1st Gen to 2nd Gen

Von Neumann Architecture (memory)

- The use of binary Number System
- A single sequentially addressed memory (SAM - Before RAM)
  - Reading a page from the first page (SAM)
  - Reading a page by selecting a random page. (RAM)
- Seperate ALU for computation
- Stored programs (Program & Its data was stored in memory)
- A controller to fetch instructions from memory to execute.

The Invention of the Transistor (Transistor)

- Used instead Thermion values
- ∴ Operating speed increased
- Size / Weight decreased
- Manufacturing cost reduced
- Concepts of - CPU
  - memory
  - Programming language
  - I/O Systems

Developed

- High-level programing languages introduced
- Developed new softwares
- Rapid growth of computer industry.

eg: IBM 1620
IBM 1401

No: Date: __/__/__

3rd Gen (IC)

IC

(*) many → transistors
→ Resistors
→ Capasitors
} inside IC

~~IC~~

Highlights

Size ↓
Speed ↑ (Processing Speed)
memory ↑
Cost ↓

- High level languages developed (further)
- large IC companies introduced (Intel, AMD)

~~Substantial OS~~

- OS Developed ✓
- Bussiness & Administration begun using
- Many different applications @once

ex: IBM System/360 & 370

No: Date: / /

4th Gen (Microprocessor)

> personal computers - IBM - Power PC
- Pentium 8088 / 8086 (Intel) (Microprocessor)

> LSIC - Large Scale ICs - Single Silicon chip - Microprocessor
> Memory Chips were megabit range

> Unix introduced
> 4GLs (4th Gen languages)
> Apps (Application softwares became cheap & easy)
> Software development techniques developed.

5th Gen

> AI
> Voice recognition
> Parallel processing
> NLP

01. Components of the Computer

(*) Input devices
(*) Output devices
(+) Processing devices
(*) Storage devices
(+) Other devices (Motherboard, Expansion Card, Power Supply)

I/O

Input	Common	Output
- keyboard	- Digital Cam	- Headphones
- Optical pen	- Pendrive	- Screen
- Joystick	- CD/DVD	- Printer (inkjet, laser)
- Scanner	- Fax	- Plotter (Banner printer)
- Barcode reader	- Touch screen	- Speakers
	- Modem	
	- Webcam	

Storage devices

Features

Volatility
> Volatile
> non-volatile

Accessibility
> Random access (RAM)
> Sequential access (SAM)

Mutability
> Mutable (Read/write)
> Read only
> Slow write, fast read storage.

No: Date: __/__/____

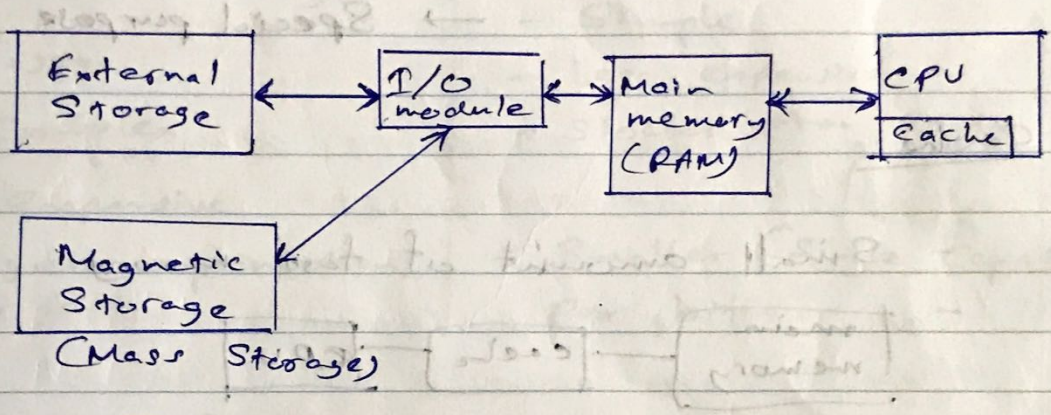
Addressability

- > Location addressable
- > File addressable
- > Content addressable

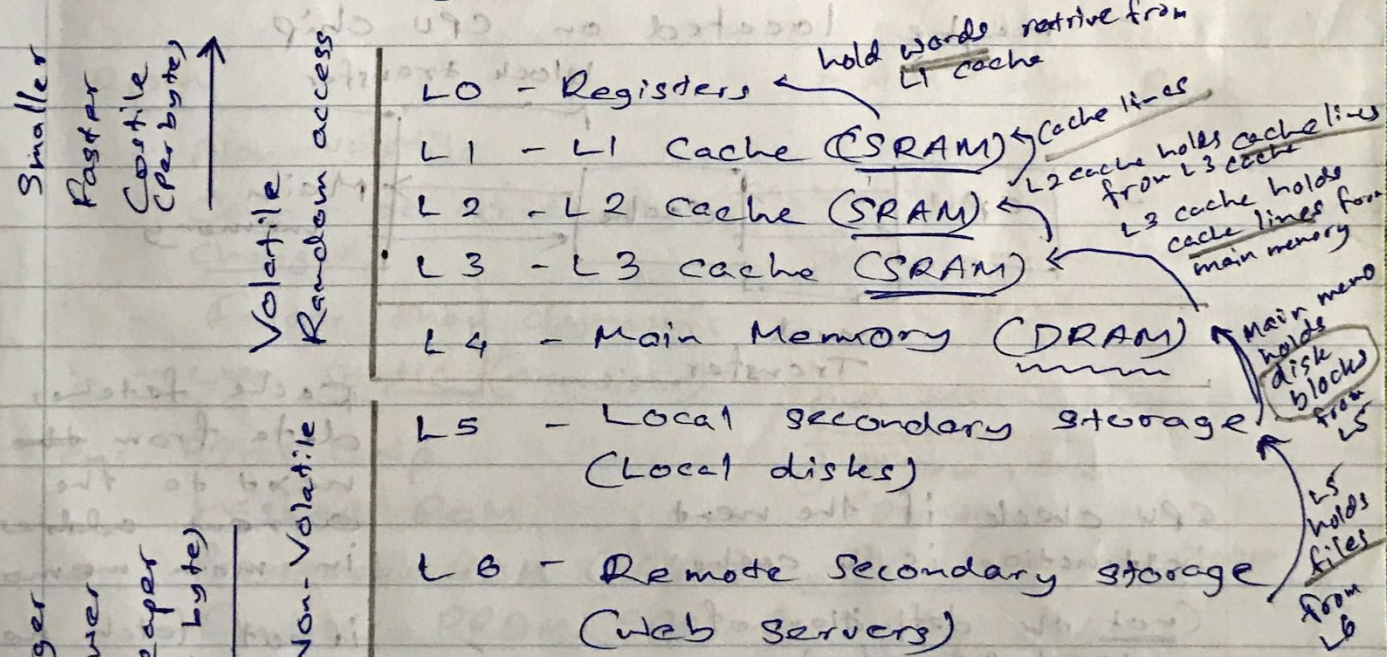
Types

Four types of storage

- > Primary storage - Main Memory (RAM), CPU
- > Secondary storage - Mass Storage (Hard disk)
- > Tertiary Storage - ~~CD/DVD, Pendrive~~ Removable media / Robotic access system
- > Off-line Storage - CD/DVD, Pendrive



Computer Storage - Hierarchy



No: Date: / /

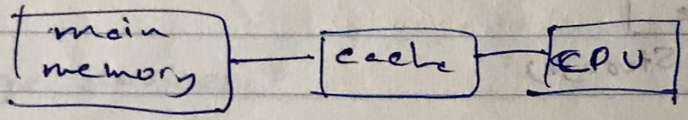
Primary Memory

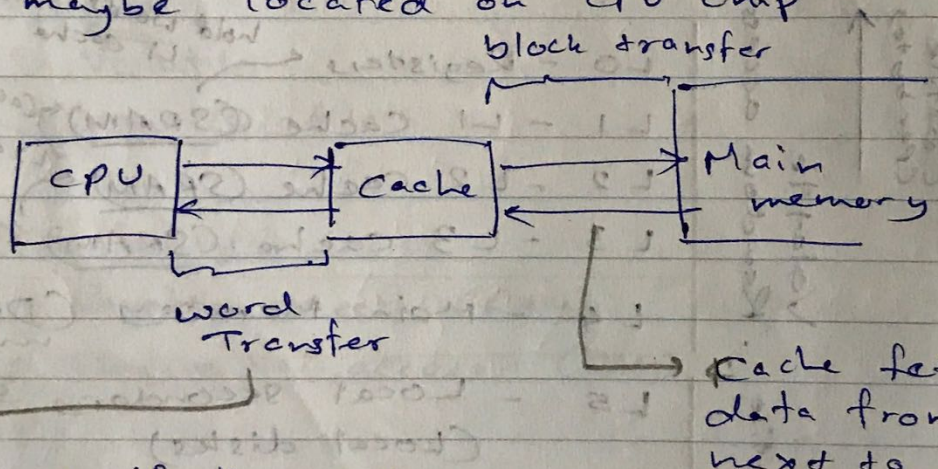
③ Main memory
① Registers
② Cache

① Registers

- very small
- very fast (Fastest in memory)
- measured by bits (32 bit, 8 bit)
- Classified into → General purpose
  → Special purpose

② Cache

- Small amount of fast memory
- 

- maybe located on CPU chip
- 

CPU check if the next instruction is in cache
(note the definition of word transfer)

Cache fetches data from the next to the current address in main memory.
if not, fetch next.
-(much slower)
(Note the definition of block transfer)

3-pre

Memory - Technologies

Static Memory

Static memory is a boolean method designed by transistors.

It holds a static data (constant)

If it contain 1 or 0 it remains that way until it is changed.

∴ Faster

∴ Complex

∴ Expensive

- Larger implementation per bit

Dynamic Memory

Dynamic memory is a capacitor contains a charge (semi conductor)

capacitor charge decrease without continues power supply.

- Charge leak

∴ need refreshing even when powered

- Simple
- Less expensive
- Slower than Static
- Smaller (Single Capacitor) per bit

3-pre

Memory - Types

ROM

- Non-volatile
- Cannot be accidently changed
- Faster than dynamic
- Use Static (transistor)

ex: BIOS chip
Masked ROM
Programable ROM (PROM)
Erasable PROM (EPROM)
EEPROM

RAM aka Read/write memory

Volatile

~~Semi~~ dynamic Memory (Capacitor/Semi Conductor)

ex: Main Memory

* All semiconductor memory (Dynamic memory) are Random Access

why use DRAM to Main Memory? (NOT SRAM)

Self study large

No: Date: / /

Main Memory

(1) Main memory consists of a number of Storage locations, with unique address.

(2) The ability of the CPU to identify each location is known as "Addressability".

(3) Each location stores a word. i.e.
i.e. The number of bits that can be processed by the CPU in a single operation.
word length may be typically 16, 32 or as many as 64 bits.

(4) Program & Data stored in memory prior to execution. (Von Neumann)

(5) Memory is a Semi-Conductor device (Modern) (Magnetic core memories used earlier)

(6) Main Memory, Primary Storage, RAM are Same

Main Memory - Highlights
- Byte addressable
- Each byte has unique address
- Addresses start from zero and increment sequentially.
- Memory Refresh
  - Memory Refresh is the process of periodically read data from an area of computer memory & immediately writing the read information to the same area without no modification.

No: Date: __/__/____

Secondary Storage

Hard Disk

Performance Parameters.

- Access Time - Seek + Rotational + Transfer
- Seek Time - Track Selection Time
- Rotational Delay - Head to reach the beginning of the sector
- Transfer Time → Time required to transfer Data

Tertiary Storage

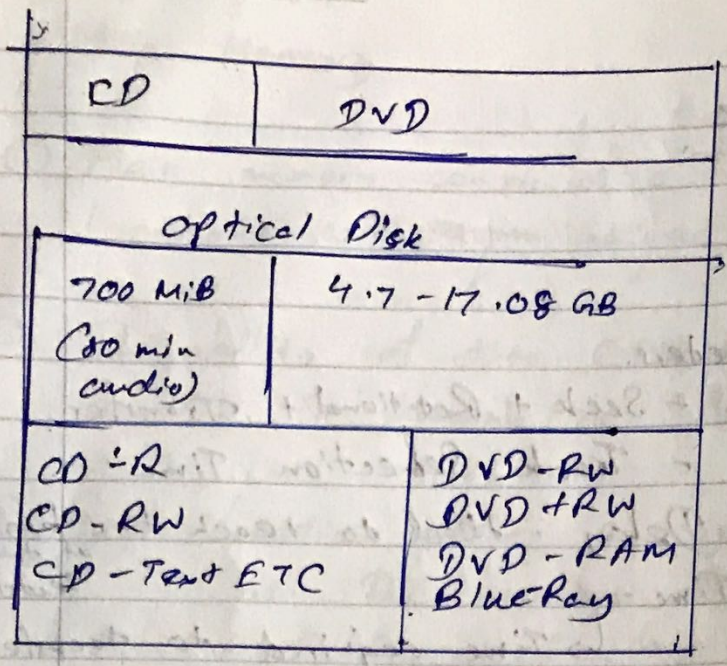
- Typically, if it involves a robotic mechanism which will mount (insert) and dismount removable mass storage media into storage device.
- Usuall very slow, usuall used to archive data (Library)
- Primarily useful for extraordinarily large data stores, access without human operators.

Offline Storage - aka Disconnected storage

> Is a computer data storage on a medium or a device that is not under the control of a Processing unit

> It must be connected by a human operator before a computer can access it again.
ex: Floppy disk, CD/DVD, ~~Pendrive~~, Memory Cards, USB Flash Drive

No: Date: / /



Other Storage Techniques

ex: Google Drive
Flickr
SkyDrive (MS)

Cloud storage means "the storage of data Online in the cloud."

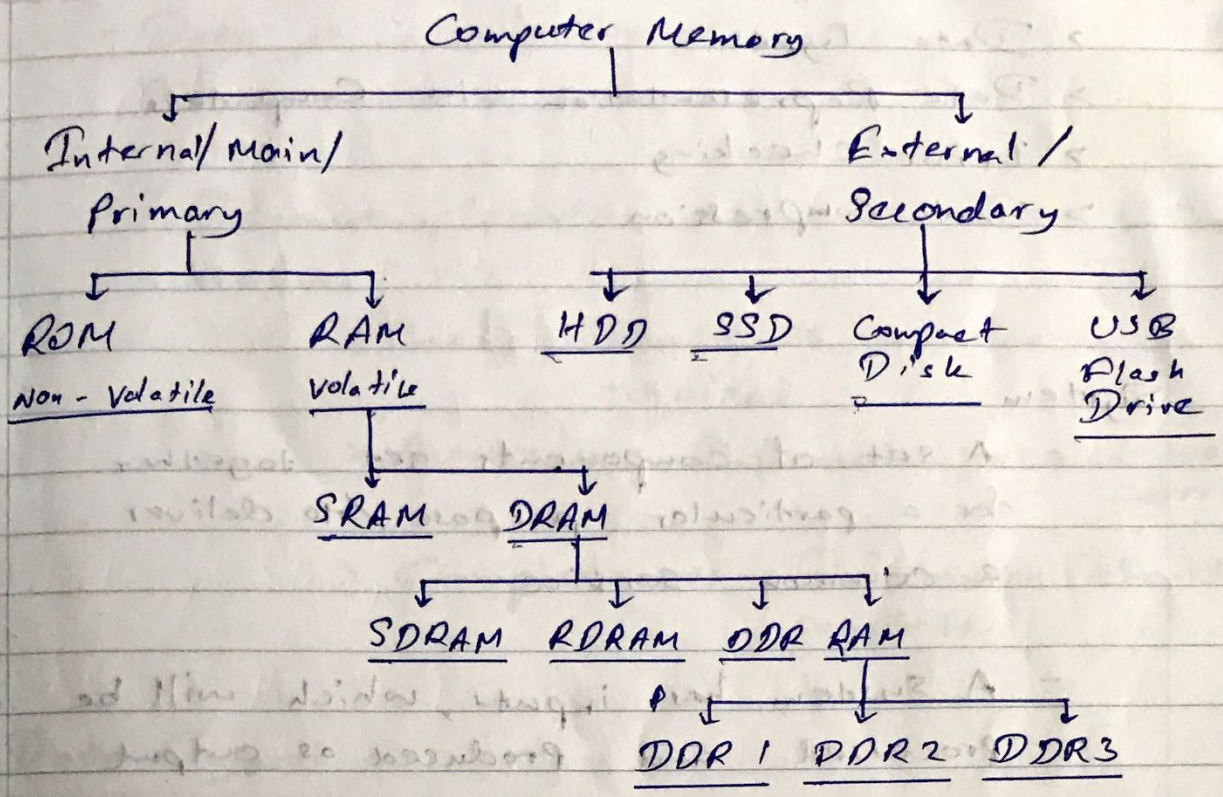
Wherein a data is stored in and accesible from multiple distributed and connected resources that comprise a cloud.

- Data Backup (Strong protection for data backup)
- Archival
- Disaster recovery
- Greater accessibility & Reliability.

No:

Date: / /

Evolution of storage



No: Week - 2

Date: / /

02. Data Representation in Computer Systems

> Data & Information
> Data Types
> Data Representation in Computers
> Error Checking
> Data Compression

System

- A set of components get together for a particular purpose / to deliver a common service
- A system has inputs, which will be processed and produced as output

Information System

- IS is a computer-based system to deliver information.
- Input :- Data
- Output :- Information

Software Engineering

- The engineering discipline of constructing information systems.

Information & Communication Technology (ICT)

- The technology used to engineer ISs.

No: Date: / /

Data & Information

- Information are derived from data
- Data is a collection of raw facts (Numbers, words, measurements)

Information provide meaningful values to the receiver

> Timely - Should be available when required

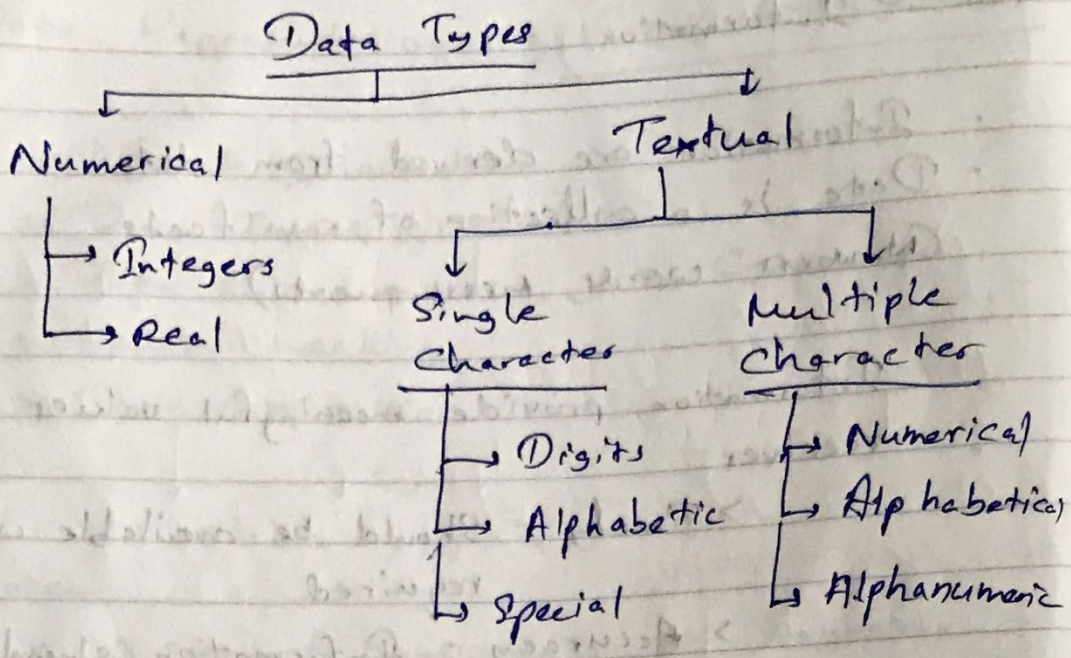
> Accuracy - Information should be correct

> Completeness - Information should be complete.

Data Types

- Quantitative Data
  - Discrete data (within a range) data, certain values only
  - Continuous Data (Can take any value within a range)
- Qualititative Data

No: Date: / /



Numeric Data

Integer numbers
> whole number (+ or -)
ex: > 451 or -352

Real (Decimal) numbers
> All numbers including everything between integers.
ex: 23.245, 5½, -2.3

Character data

Single character
0, 1, 2, 3... (numeric) /digit
A, B, a, b, ... (alphabetic)
#, @, %, &.. (Special)

Multiple characters (String)
349, 254.781, .. (Numeric)
Cat, Hey (Alphabetic)
(#, %, $), .. (Special)
DIT4521..
(Alpha-numeric)

No: Date: / /

Data Representation

> Data representation refers to the form in which data is stored, processed, and transmitted

> Store data in digital formats that can be handled by electronic circuitry

> Digitalization is the process of converting information, such as text, numbers, photo or music into digital data that can be manipulated by electronic devices

> The Digital Revolution has evolved through four phases, beginning with big, expensive, standalone computers and progressing to today's digital world in which small, inexpensive digital devices everywhere.

How do computers represent data.

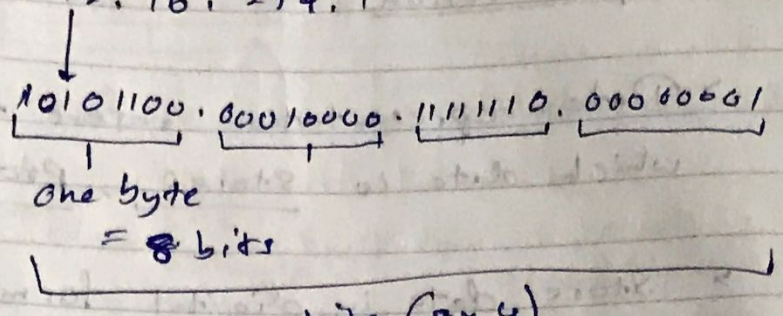
> computers are digital (on or off)
- recognize only two discrete states
- computers are electronic devices powered by electricity, which has only two states, on or off.
- Binary number system (0, 1) is used for processing

? what is the meaning of digital
? what are the advantages of using binary number system

No:

Date: / /

IPV4 : 172.16.254.1



Binary number system

(*) The memory is made up of BITS and Bytes

* > 8bits = 1 Byte
> 1024 Bytes = 1KB
> 1024 KB = 1MB
> 1024 MB = 1GB
> 1024 GB = 1TB

Permonology related to bits and bytes is extensively used to describe
- Storage capacity
- network access speed

(*) Each BYTE can be addressed uniquely.
(*) When the address is expressed in Binary, the number of maximum BITS used to write the address specifies the total number of locations available.
(*) If n number of BITS are available then the total number of locations available is 2^n
(*) If we have 32 BITS then we can have 4GB of memory (2^{32} = 4GB)

1 bit can hold 2^1 (2) possible values (0, 1)
2 bit can hold 2^2 (4) possible values (00, 01, 10, 11)
3 " " " 2^3 (8) " " (000, 001, ..., 111)

⋮

n bits can hold 2^n possible values.

∴ we can have 2^n Bytes as memory (not the actual size)

64 bit ⟶ maximum amount of memory is 2^{64} = 16 exabytes

(4 bit, 16 but 10)

BCD - Binary Coded Decimal

4 bit code (for numeric values only)

0 - 0000
1 - 0001
⋮
9 - 1001
10 - 0001 0000
11 - 0001 0001
12 - 0001 0010

(7 bit, 128)

ASCII - American Standard Code for Information Interchange

7 bit code for all 128 characters

A: 1000001 (65)

Letters & other characters are stored by assigning a number for each one.

Extended ASCII (8 bit, 256)

> 8-bit System & allows the system to store up to 256 different characters

Unicode (16 bit, 65 536 characters)

Unicode provides a unique number for every character
> no matter - platform
- program
- language

It uses the number to identify a character
(Likewise ASCII uses decimal value for each characters)

No:

Date: __/__/__

Error Checking : Parity

> The parity is ~~not~~ used in error checking to find errors that may occur during data storage/ transmission.

> A parity bit is a single bit added to a binary data transmission used to indicate if whether the 0's and 1's within that data transmission is an even or odd number.

There two modes of parity

① odd parity.

The number of 1-bits (including the parity bit) must be an odd number

② Even parity.

The number of 1-bits (including the parity bit) must be an even number

A single bit is append to each data chunk (either as the least or most significant bit)
- makes the number of 1 bits even/odd

Example:

Even parity	Odd parity
1000000(1)	1000000(0)
~~0000~~	
1111101(0)	1111101(1)
1001001(1)	1001001(0)

No: Date: / /

Example:

① 11000101

- Odd parity is used ✗ - wrong (Even)
- MSB is used for parity (if even) - can't say

② Even parity of ASCII "V" (0110101)

= 01101010 or (0) 0110101

Data Compression

> To reduce file size and ~~the~~ transmission times, digital data can be compressed.

> Data compression refers to any technique that records the data in a file so that it contains fewer bits.

> Compression is commonly referred to as "zipping"

~~→ the process is commonly referred to as "zi~~

> The process of reconstituting zipped files is called extracting / unzipping.

> Compressed files may end with a .zip, .gz, .pkg or .tar.gz formats.

Date: / /

No: Week 9

03. Computer Architecture

- Fundamentals of computer Organization
- The heart of a computer system, CPU, its structure and functionality.
- How CPU is seen by programmers and CPU design engineers.
- Instruction set architecture of a processor.

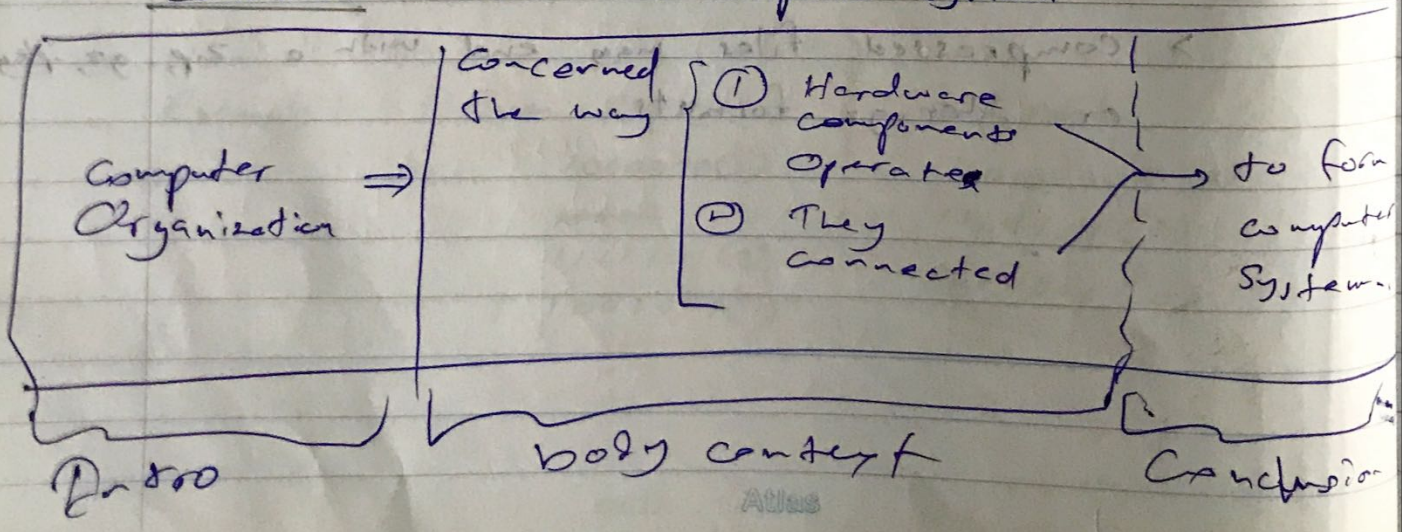
Computer Organization

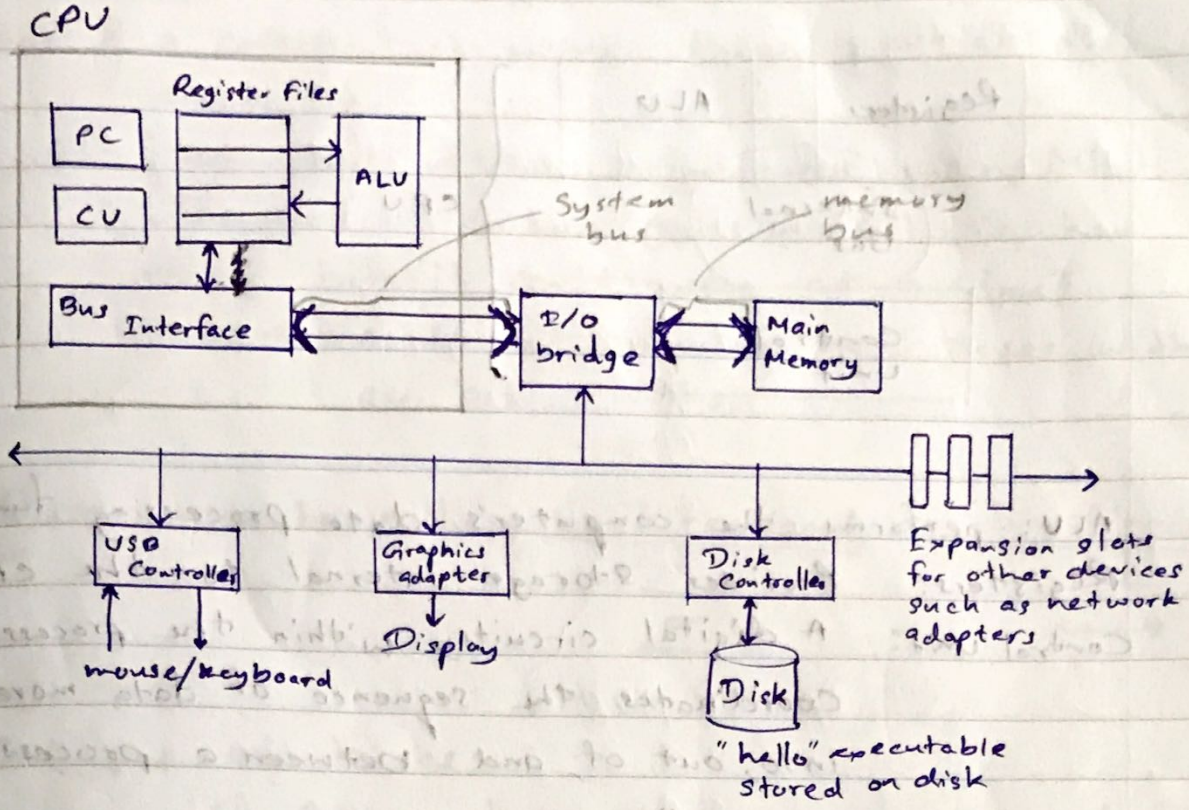
(M) Different groups of engineers see computer different manner

(A) Hardware Engineers.. See How different components work, their configuration, how they are organized - (Computer Organization)

(A) Software Engineers see how computer can be used for different applications, how their high-level language code can be executed in machine, How easily it can be used for programs - (Instruction Set architecture)

Computer Organization is concerned with the way the hardware components operate and the way they are connected to form the computer system.



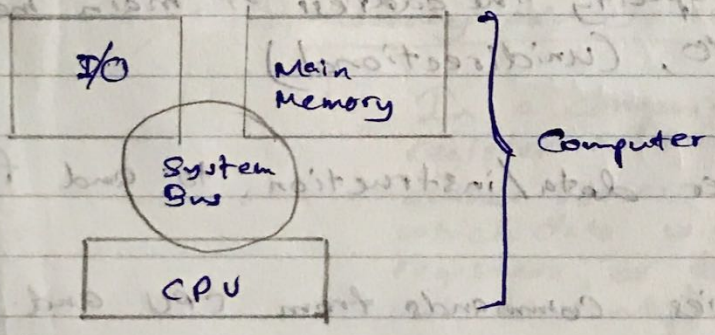


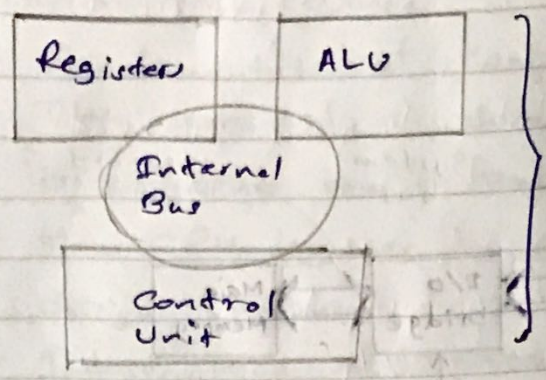
CPU: Controls the operation of the computer and performs its data processing function

Main memory: Stores data and instructions (Programs)

I/O: Moves the data between the computer and its external environment

System bus (System Interconnection):
Mechanism that provides for communication among CPU, MM and I/O



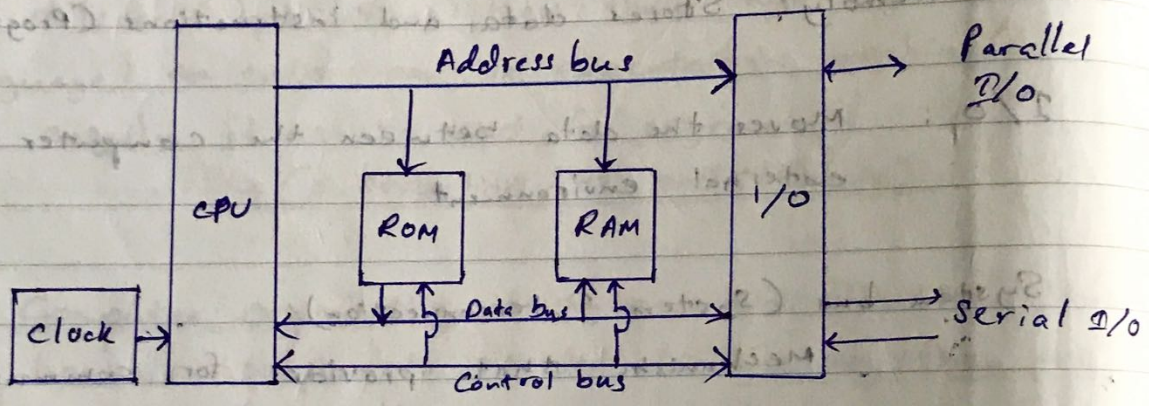


ALU: performs the computer's data processing functions

Registers: Provides storage internal to the CPU

Control unit: A digital circuitry within the processor that coordinates the sequence of data movements into, out of, and between a processor's sub-units.

Modern Computer



Address Bus: CPU Specify the address of main memory or I/O, (unidirectional)

Data Bus: Transfer data/instruction to and from CPU

Control Bus: Carries commands from CPU and status signals to CPU

My explanation (not in the ppt)

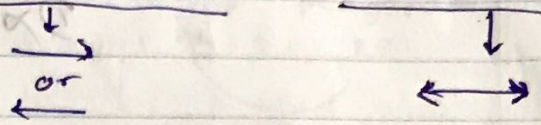
Bus = a collection of wires through which data is transmitted

> (The speed of bus is measured ~~by~~ in MHz)

> (The size of a bus (its width) is how many bits it can transfer at a time)

ex: a 64 bit computer has buses with 64 bit widths.

(*) Busses can be either unidirectional or bi-directional



3 types of buses:

(1) Address bus: (Unidirectional) Send a memory address along the bus from the CPU to the memory. To fetch/write data, the CPU needs to tell the RAM the address.

(2) Data bus: (Bi-directional) → sends the actual data to and from the memory

(3) Control bus: (Bi-directional) Carries commands from the CPU and status messages from other hardware.

ex: Status: clock Read write
1 0 1
Interrupt
1

Additional Component:

(4) Memory address register

In a computer, the memory address register is the CPU register that either stores the memory address from which data will be fetched to the CPU registers, or the address to which data will be sent and stored via system bus.

Back to the PPT

8086 Registers

They are grouped into several categories as follows:

① General purpose registers
- AX
- BX
- CX
- DX

② Special purpose
- SP
- BP
- SI
- DI

③ Segment registers
- CS
- DS
- ES
- SS

④ The program counter / instruction pointer

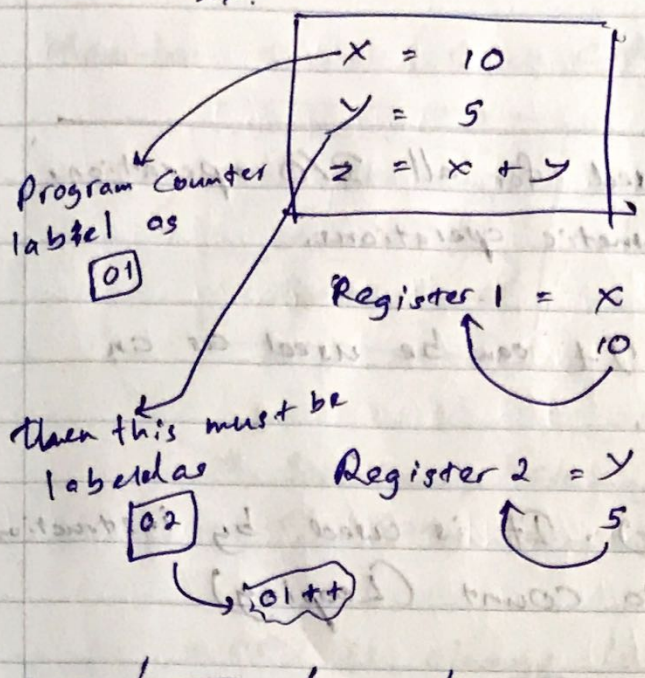
⑤ The status flag register, FLAGS

(PC) / (IP)

No:

Additional Note By Dr. Saruvitha

ex:



01 / 02 / .../...

variable 1

Variable 2

Increment by 1 - saquently increase for the next value.

By PC/IP
Program counter / Instruction pointer

8086 Registers

General Purpose Registers

AX : (Accumulator) Used for all I/O operations and some arithmetic operations.

BX : (Base register). It can be used as an address register.

CX : (Counter register). It is used by instructions which require to count (Looping)

DX : (Data register). It is used for some input / output and also when multiplying and dividing.

Special Purpose (Addressing registers)

> "Addressing registers" are used in memory addressing operations, such as holding the source address of the memory and the destination address.

> SI : (Source index) and is used with extended addressing commands.

> DI : (Destination index) and is used in some addressing modes.

> BP : (Base pointer)

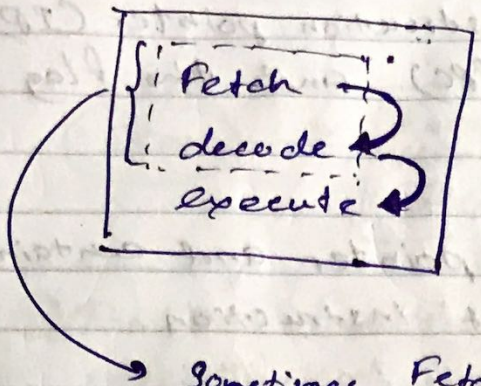
> SP : (Stack pointer)

No:

Dr. Savitha

Date: / /

Machine cycle / CPU cycle



Sometimes, Fetching & decoding is done by the same (component / Process)

> CPU is always looking for interruptions
> Interrupt handler

↓

always looks:
what? / what's next?

If there are any interruption it called "HALT"

(#) Assembly languages have kinda close words to natural languages. And those parts are called,

mnemonics

PPT

(*) The two status registers have 16 bits and are called the instruction pointer (IP) or program Counter (PC) and the flag register (F)

(*) IP is the instruction pointer and contains the address of the next instruction

(*) Flag register holds a collection of 16 different conditions. (Ex: result is zero or not, there is a "carry" etc.)

> Segment registers: There are four areas of memory called segments, each of which are 16 bits and can thus address upto 64 KB (from 0000h to FFFFh)

(1) Code segment (cs register)
where the program code is stored.

(2) Data segment (ds register)
where data from the program is stored

(3) Stack segment (ss register)
where the stack is stored

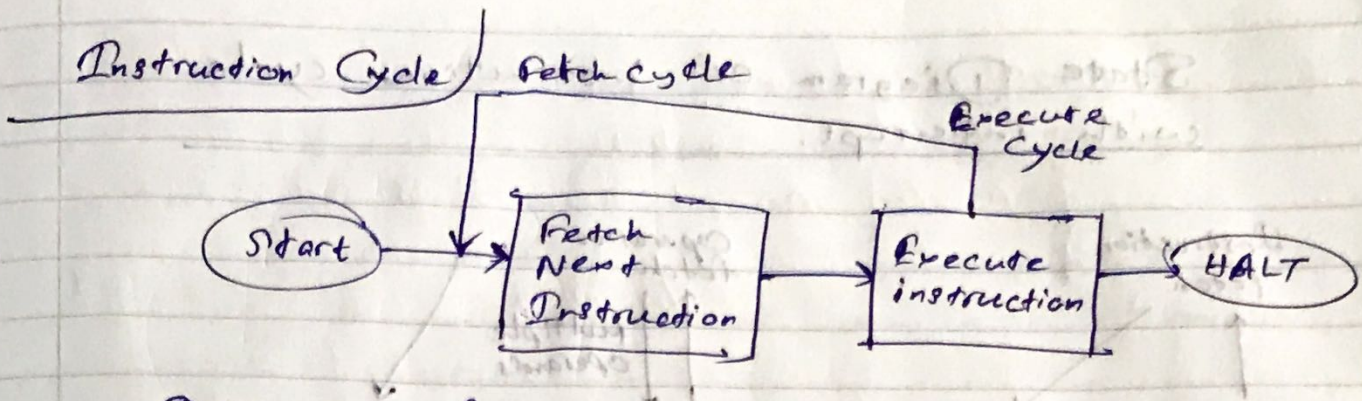
(4) Extra segment (es register)
a spare segment

All address are with reference to the segment registers.

No:

Date: / /

Instruction Cycle

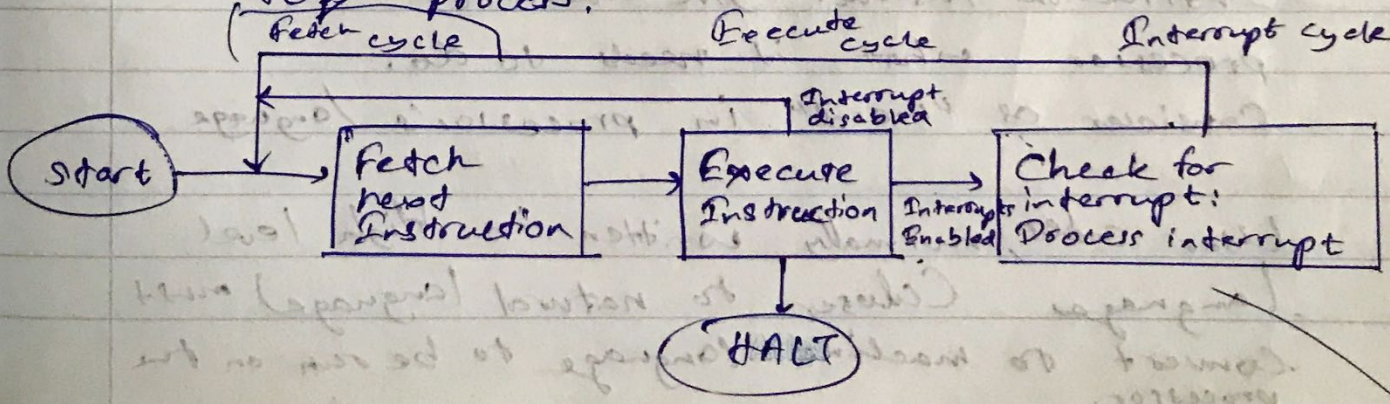


- The basic function performed by a computer is execution of a program.
- Program is a well-defined set of instructions.
- Processing a single instruction is called instruction cycle.
- It has two steps referred as fetch cycle and execute cycle.

Instruction Cycle with interruption

> All computers provide a mechanism by which other modules (I/O, Memory) may interrupt the normal processing of the processor.

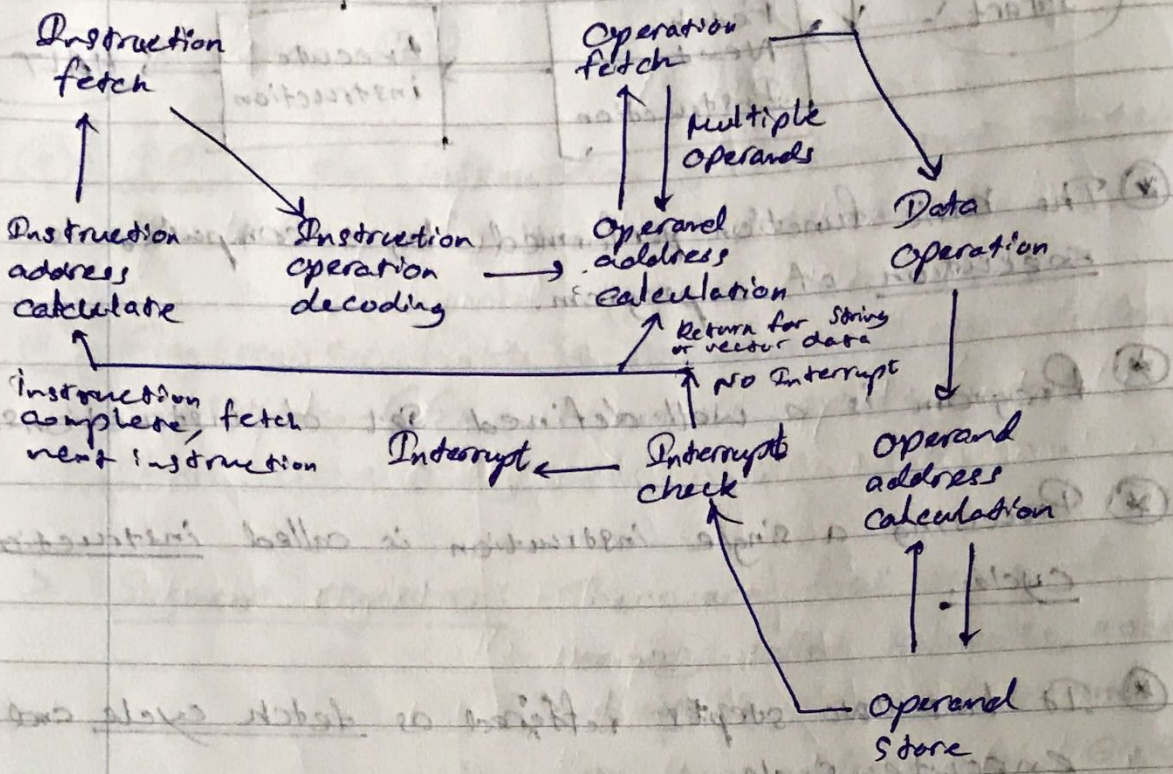
> Processor engaged in executing other instructions while slow external devices (ex: I/O module) become ready and send interrupt request to the processor.



Date: / /

No:

State Diagram of instruction cycle with interrupt.



ISA - Instruction set architecture.

Instruction set is a app part of the computer that pretains to internal (or low level) programing, which is basically machine language

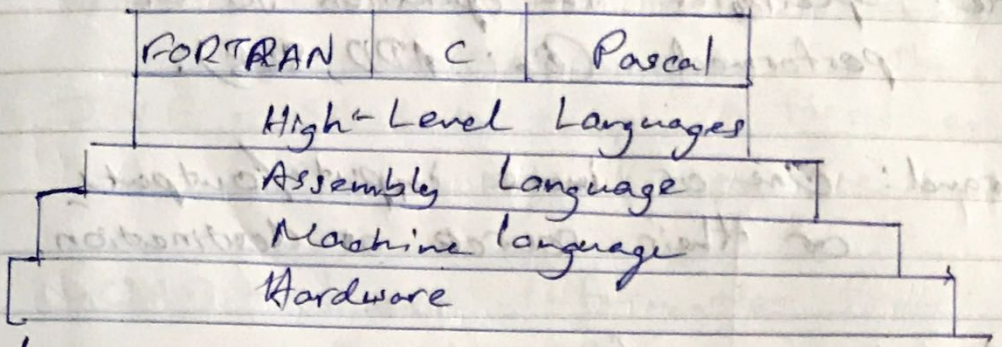
The instruction set provides commands to process.

Instructions instruct (give orders to) the processor what it needs to do.
Consider as "word" in processor's language

User codes, normally written in High level languages (closer to natural language) must convert to machine language to be run on the processor.

Assembly language has rich set of mnemonics to represent machine language instructions.

Assembly language commands are in a human readable format.



Assembly lang

```
MOV AX, DATA
MOV DS, AX

MOV AL, NUM1
MOV AL, NUM2
```

Typical ISA defines

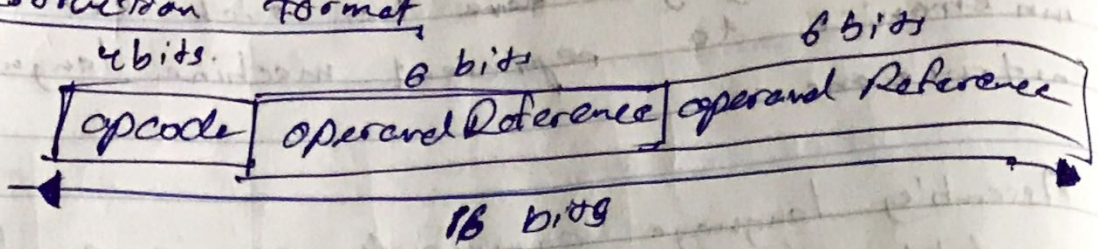
- How to access data in registers, memory and other I/O device
- Mechanism to transfer data and instruction to and from processor.
- Operations such as additions, subtractions which processor can execute
- Control mechanisms such as branch, jump

> To be effective as a programmer or processor designer, one should know how ISA works.

Date: __/__/__

No:

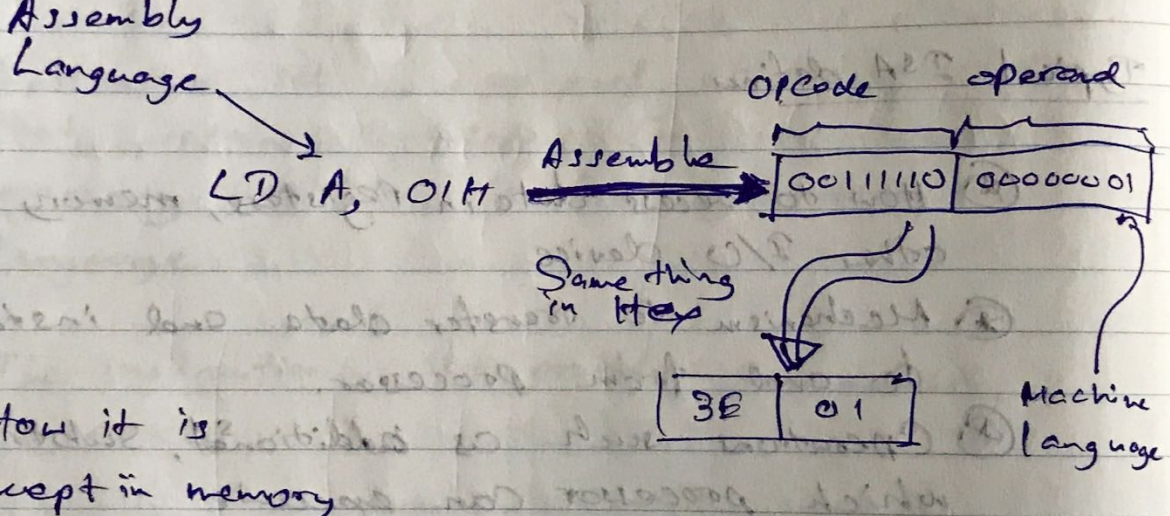
Instruction format



opcode: Specifies the operation to be performed (Ex: ADD, SUB).

Operand: One or more input/outputs or their source or destination.

Next instruction reference:
Tells the processor where to fetch the next instruction once the execution of this instruction is completed.



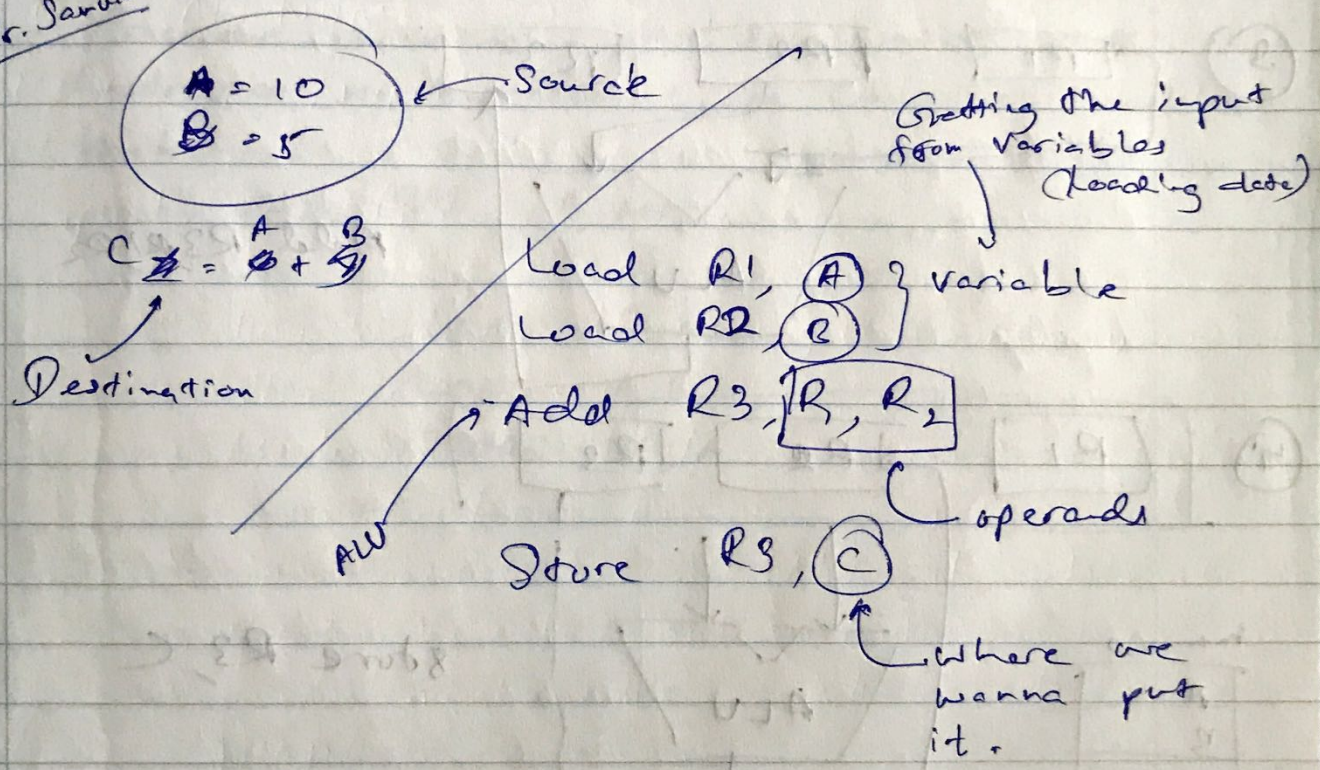
No: Date: / /

Instruction Set - Sample instructions.

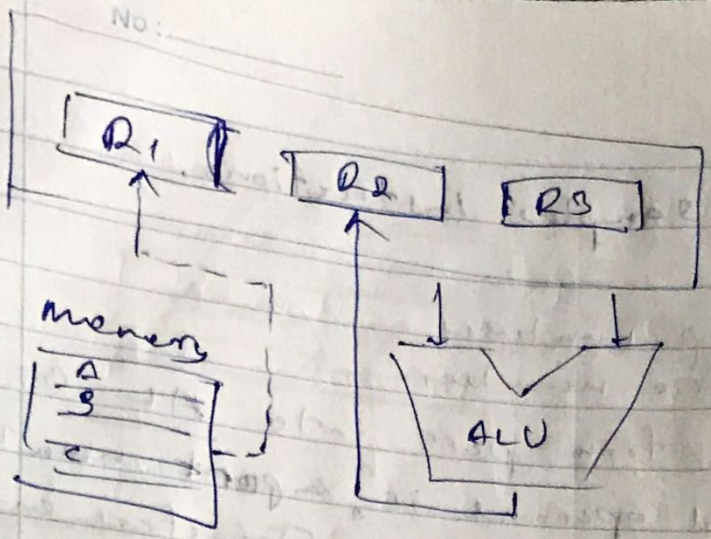
ADD - Add two numbers
CMP - Compare numbers
IN - Input from port into AL or AX. Second operand is a port number.
JMP - Unconditional Jump. Transfers control to another part of the program.
JNE - Short Jump if first operand is NOT equal to second operand.

LOAD - Load information from RAM to the CPU
OUT - Output information to device
STORE - Store information to RAM.

Dr. Samudha

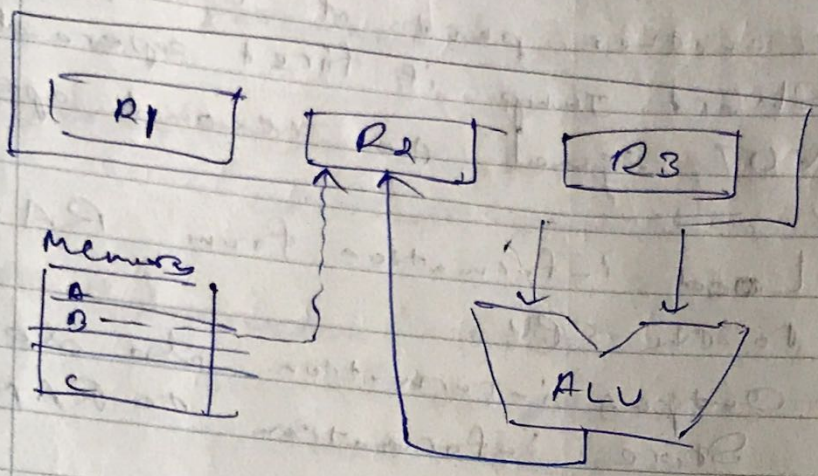


1



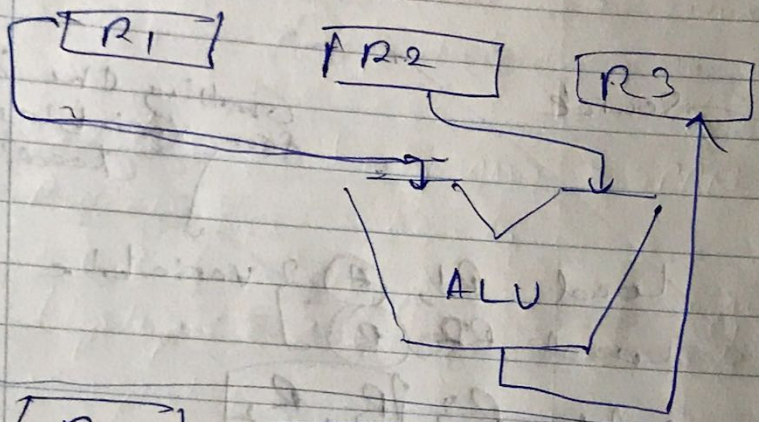
Load R1

2



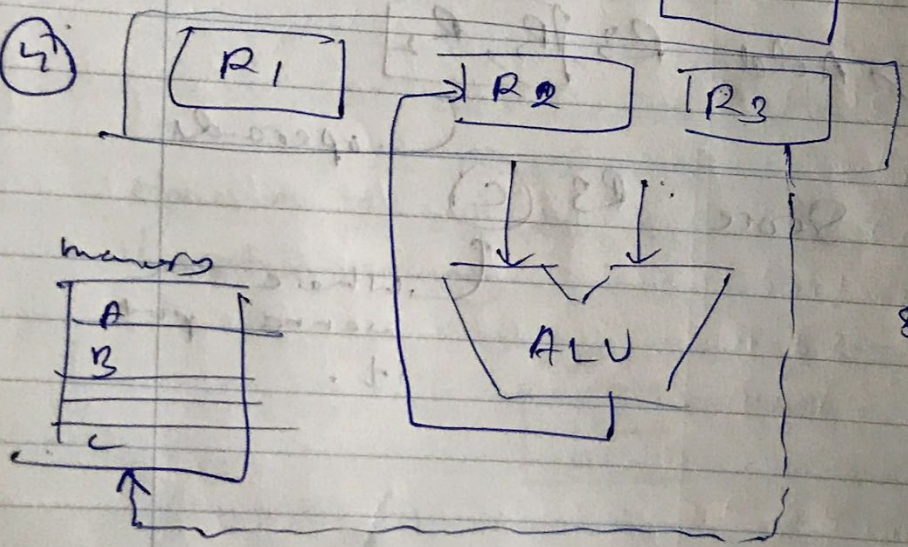
Load R2

3



Add R3, R1, R2

4



Store R3, C

No: Date: / /

CISC & RISC

CISC

> Primary goal is to complete a task in as few lines of assembly as possible.
> Processor hardware complex; needs less RAM to store the code; Instructions set is high-level, hence Compiler workload is low
> Ex: MULT M1, M2

RISC

> Simple instructions that can be executed within one clock cycle
> Processor hardware simple; Need more RAM; Instruction set low-level, hence Compiler workload high

Ex: LOAD A, M1

LOAD B, M2
LOAD A, B
STORE M1, A

Date: / /

No:

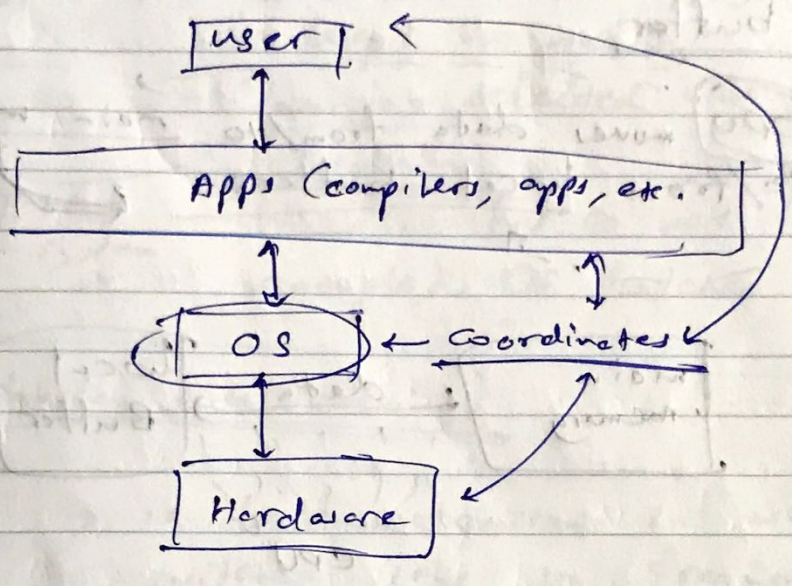
CISC	RISC
> Includes multi-clock complex instructions	> Single clock, reduced instructions only
> Emphasis on hardware	> Emphasis on hardware
> Memory to memory: "LOAD" and "STORE" incorporated in instructions.	> Register to register: "LOAD" and "STORE" are independent instructions.
> Small code sizeds, high cycles per second	> Low cycles per seconds, large code sizes.
> Transistors used for storing complex instruction	> Spends more transistors on memory registers.
> Intel X86 family, AMD Processors are heavily used in desktop, laptop and server computers.	> SPARC and Power PC are used in desktop computers and game consoles.
	> RISC Processors are heavily ~~use~~ used in real-time embedded systems such as mobile phones, washing machines, Routers.
	>
	> Raspberry Pi and Arduino
	> IoT drives by RISC Processors.

OS

OS: The intermediary between user & hardware.

OS Goals:

> Execute user progoams
> make computer convenient to use
> manage hardware in an efficient manner.

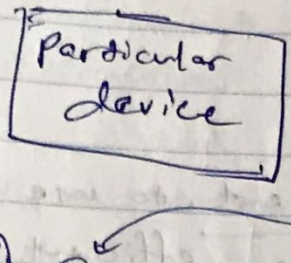


Purpose of OS: (1) Environment for program execution and development.
(2) Manage the resources (CPU/memory/I/O/hard/file etc.)
(3) Provide the access controlling (username & PW)

Booting Process: (*) Bootstrap program is loaded at power-up or reboot
(+) stored in ROM or EPROM generally known as firmware
(*) Initialize all aspects of system (Power on self test)
(*) Loads OS kernal starts execution.

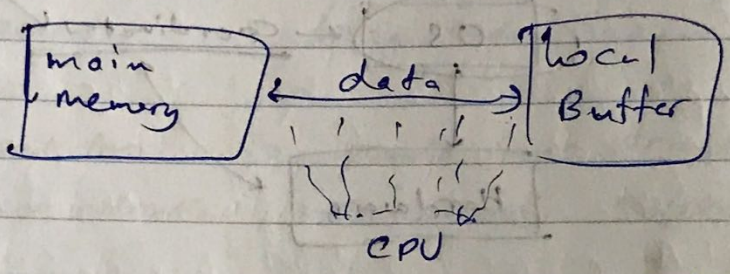
No: Date: / /

* I/O devices & the CPU can execute concurrently

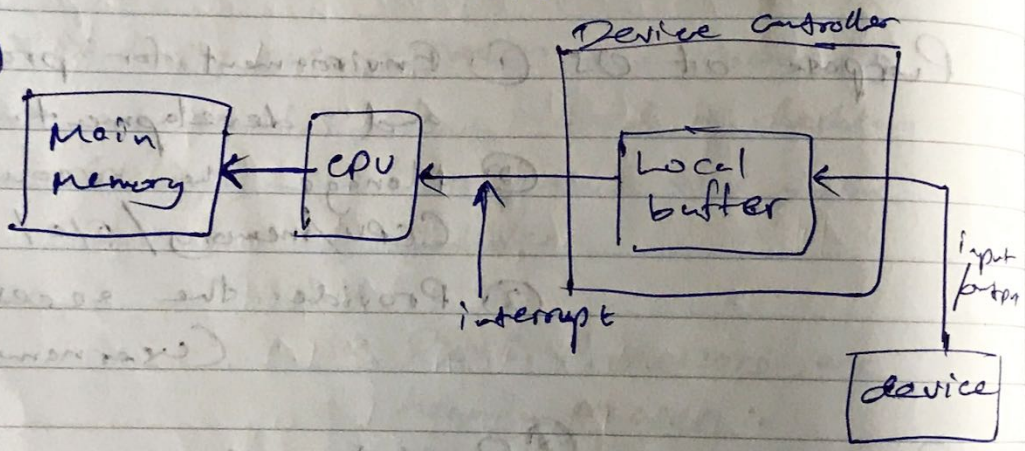
* 

* Each device controller has a local buffer

* CPU moves data from/to main memory to/from local buffers.



*



* Device controller informs CPU that it has finished its operation by causing a signal / an interrupt

No: Date: / /

Multiprogramming (Batch System)

- need for efficiency
- Single user can not keep CPU & I/O devices busy at all times.
- Multiprogramming organizes jobs (code & data) so CPU always has one to execute
- A subset of total jobs in system is kept in memory
- One job selected and run via job scheduling.
- When it has to wait (for I/O for example), OS switches to another job.

The manager of CPU is the time keeper of batch system

Timesharing (Multitasking)

> is logic extension in which CPU switches jobs so frequently that users can interact with each job while it is running, creating interactive computing.

- Response time < 1 second should be
- Each user has atleast one program executing in memory ⟫ process
- If several jobs ready to run at the same time ⟶ CPU scheduling
- If processes don't fit in memory swapping moves them in and out to run,

- Virtual memory allows execution of processes not completely in memory.

Real-time OS

Hard real-time
- Guarantees that critical tasks complete on time.
- Control devices in a dedicated application
  - controlling scientific experiments
  - medical imaging system
  - industrial controlling system
  - some display systems.
- Secondary storage is limited or absent.

Soft real-time.
- priority to critical tasks before others.
- Limited utility :- industrial robotics.
- Useful in multimedia apps required advanced OS features.

No: Date: / /

Types of operating systems

Mobile OS

> These are the operating systems for smart phones, tablets and wearables.

> The system combines the feature of a personal computer with additional feature useful for a handheld device.

> Mobile operating systems start when a device is powered on to provide access to installed applications.

> They manage wireless connectivity

> Apple iOS
Google Android
Microsoft windows

No:

Date: / /

OS Components

1. Process Management
2. Main memory management
3. Secondary management
4. File management
5. I/O System management
6. Protection system
7. Networking (Distribution System)
8. Command - interruption System.

---

(1) Process management

(*) A process is a program in execution

Program ⟶ Passive entity
Process ⟶ Active entity

(*) Process need resources
↳ CPU
Memory
I/O
Files ⟶ to complete execution.

(*) Process execution is sequential and program counter register is specifying the location of next instruction to execute

(*) Process ~~termination~~ requires reclaim of any ~~reusable resources~~
↳ I/O ← Lecture slide this statement is wrong.

Atlas

(*) When the process terminates, the operating System will reclaim any reusable resources.

> Process management activities
  (1) Creating / Deleting both User/System processes.
  (2) Suspending and resuming processes
  (3) Providing mechanisms for process synchronization
  (4) Providing mechanism for process communication
  (5) Providing mechanisms for deadlock handling

(2) Main memory management

(*) To execute a program, it must be in memory
(*) Memory management determines what is in memory and when.
  - optimizing CPU utilization and computer response to users.

(*) Memory management activities
  > Keeping track of which parts of memory are currently being used and by whom.
  > Deciding which processes (or parts thereof) and data to move into and out of memory
  > Allocating and deallocating memory space as needed.

(9) Secondary-Storage Management

> Usually, disks use to store data that does not fit in main memory or data that must be kept for a "long" period of time.

> Proper management is of central importance

> Storage management activities
  (1) Free-space management
  (2) Storage allocation
  (3) Disk scheduling
  (4) Proving a uniform, logical view of data (file)

(9) File management

> Files usually organized into directories
> Access control on most systems to determine who can access what
> File management activities
  (1) Creating and deleting files and directories
  (2) Primitives to manipulate files and directories
  (3) Mapping files onto secondary storage
  (4) Backup files onto stable (non-volatile) storage media

No: Date: / /

Operating System Services

1. Provide user interface (UI)
   - > Command line interface (CLI) - using text command
   - > Batch interface
     - commands and their directives are put in a file
   - > Graphical User Interface (GUI)
     - window system with pointing devices
2. Provide environment for program execution
   - > OS must load program and run it.
3. Provide some means to do I/O
   - > User programs cannot execute I/O operations directly.
4. Provide mechanism to do file-system manipulation
   - > Capability to read, write, create and delete files, directory trees etc.
5. Provide Mechanism for process communication
   - > exchange information between processes executing on the same computer or on different systems through a network.
   - > Implemented via shared memory or message passing.

No: Date: / /

(6) Detect errors and take appropriate action to ensure correct and consistent computing.

> detect error in CPU and memory hardware, in I/O devices or in user programs.

(7) Resource allocation

> Allocating resources to multiple users or multiple jobs running at the same time (CPU scheduling, etc.)

(8) Accounting

> Keep track of and record which users use how much and what kinds of computer resources for account billing or for accumulating usage statistics.

(9) Protection and security

> Ensuring that all access to system resources is controlled (access permissions, etc.)

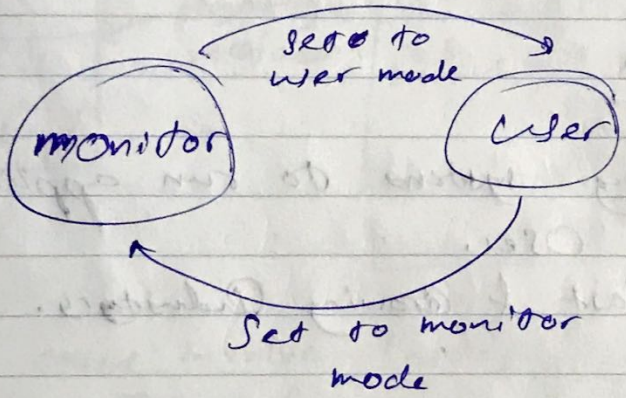
No: Date: / /

Operating Systems - Operations

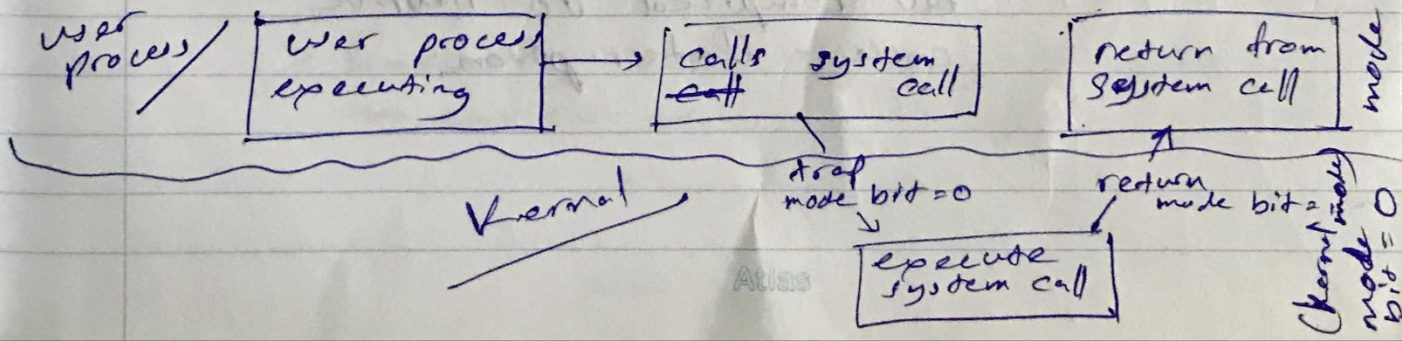
Dual mode operations allows OS to protect itself and other system components.

> User mode and kernal mode
> mode bit provided by hardware
> Provides ability to distinguish when system is running user code or kernal code.

Some machine ~~tool~~ instructions that may cause harm are designated (by hardware) as privileged instructions and they can be executed only in monitor mode



A system call is the way that a computer program requests a service from the kernal. This may include hardware-related services changes mode to monitor/kernal, return from call resets it to user



Interrupts

(+) Operating systems are Interrupt driven.

(+) Three types:

> Hardware interrupts
- generated by hardware devices.

> Software interrupt
- generated by programs when they are executing system calls.

> Traps - generated by CPU itself to indicated an error.

Virtualization

(+) Allows operating systems to run applications within other OSes.
> Vast & Growing Industry.

(+) Emulation - used when source CPU type different from target type
ex: PowerPC to Intel x86

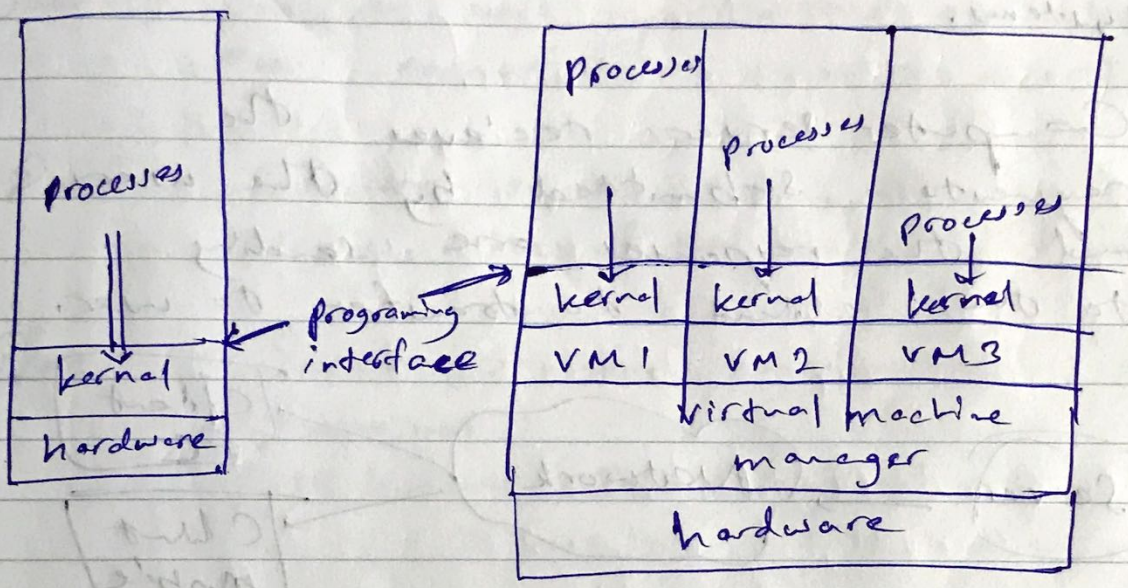
> Generally slowest method
> When computer language not compiled to native code - Interruption.

No: Date: / /

(*) Virtualization - OS native compiled for CPU, running guest OSes also natively compiled

> Consider VMware running WinXP guests, each running applications, all on native WinXP host OS

> VMM (virtual machine manager) provides virtualization services.



Use cases involve laptops and desktops running multiple OSes for ~~coop exercise~~ exploration or compatibility.

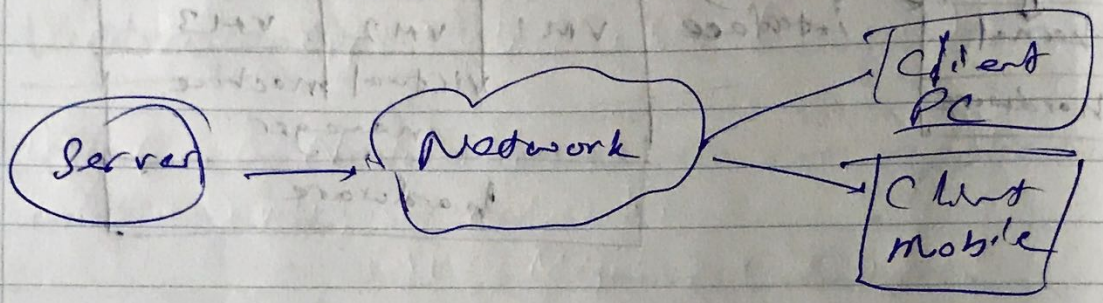
> Developing apps for multiple OSes without having multiple systems.
> QA testing applications without having multiple systems.
> Executing and managing computer environments with data centers.

Computing Environments - Client-Server

(*) The model of a computer network in which many clients (remote computers) request and recieve services from a centralized server (host computer).

(+) User requests are accepting by the interface provided by the client systems.

(*) Computer-server recieves the requests submitted by the users and the responses are sending to the client to transfer to user.



Cloud Computing

(*) Delivers computing, storage, even apps as a service across a network.

(*) Logical extension of virtualization because it uses virtualization as the base for its functionality.

> Amazon EC2 has 1000+ servers VMs & storage across Internet Pay based on usage

LOL

No: Date: / /

Public Cloud - Available via Internet to anyone willing to pay

Private Cloud - run by a company for the company's own use

Hybrid cloud - includes both public and private cloud components.

SaaS - one or more applications available via the internet (ex! word processor)

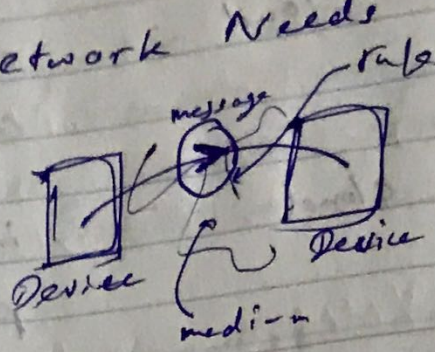
PaaS - software stack ready for application use via the Internet (ex! a db server)

IaaS - Servers or storage available over Internet
(S3 bucket, EC2, Compute engine)

Networking

L8) Network Centric World.

- Data communication Network Needs
  1. Devices
  2. Medium
  3. Messages
  4. Rules



- A computer Network
  1. A collection of computers and other associative devices.
  2. Interconnected by intermediate networking devices (Switches, hubs, bridges, routers)
  3. Transmission media (copper cables, fibre, free air)

Packet tracer steps

- Modern Networks
  Converged networks (SLT Megaline (everything))
  1. Data
  2. Voice
  3. Video
  4. Image over the same network

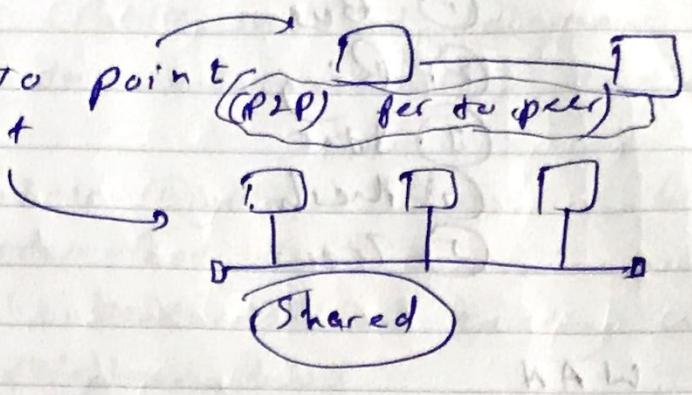
  (DVVI)

- Reliable Network
  1. Fault Tolerance
  2. Scalability
  3. QoS (Quality of Service)
     - Delay
     - Jitter
     - Reliability
     - Bandwidth
  4. Security

FCAPS

- Connecting devices in network

1. Point to point
2. Multipoint



-

- Classification of Computer Networks

1. LAN
2. WAN
3. MAN

- Features of a LAN

1. Smaller Scope
2. Usually owned by same organization
3. Data rates are much higher (speed)
4. Usually, a single broadcast systems
5. Easy to manage

- Network devices used to build a LAN

1. Switches
2. Wifi routers
3. Repeaters } Depriciated
4. Hubs
5. Bridges

- LAN Connectivity methods

1. Ethernet
2. Fast Ethernet
3. Gigabit Ethernet
4. Wireless LAN
5. Token Ring
6. Token Bus

- Topologies
  1. Bus
  2. Ring
  3. Star
  4. Mesh
  5. Tree

- WAN
  1. Large geographical area
  2. Collection of LANs (geographically-dispersed)
  3. Internet is the largest WAN

- Network device for WAN
  1. Routers
  2. WAN Switches (L3 Switches)
  3. ADSL Routers
  4. Servers
  5. Multiplexers

- WAN Technologies
  1. Dial up
  2. Leased Lines
  3. Frame relay
  4. ISDN
  5. ADSL
  6. VPN
  7. More lately: SD-WAN (software defined WAN)

No: Date: / /

- Dial up connections
  > PSTN (Public switched telephone network) used to connect computers to the WAN

- ADSL (Asymmetric Digital Subscriber Line)
  > Facilitates fast data transmission at a high bandwidth on existing copper wire telephone lines.

- MAN > Larger than LAN, Smaller than WAN such as a city
  > Typically owned by a single entity (ex: a government body, large corp.)

Internet

- ~~Def~~ Internet
  → ISOC (Internet Society)
  ↓
  IAB (Internet ~~IAB~~ Architecture Board)
  ↓
  IETF (Internet engineering task force) | IRTF (Internet research task force)

- Internet Services (Email, news, web, & Blah Blah)

-

- Private Networks

① A network which is not directly connected to internet

② It is not mandatory to follow internet standards

③ LAN (and can also have the internet access)

- Public Network

① If a network is directly connected to Internet, It will become a part of internet.

② It is mandatory to follow internet standards.

③ With the internet.

- Intranet

① A private network maintained by a company or particular organization.

② It is used to exchange internal information and it is restricted to the general public

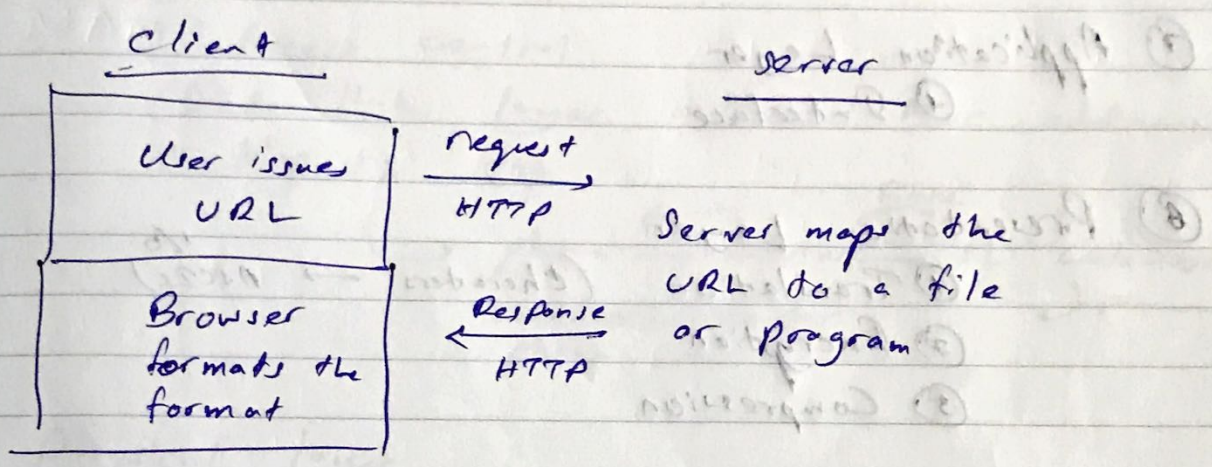
- Extranet.

① A set of interconnected intranets is known as an extranet.

ISO - OSI Reference Model

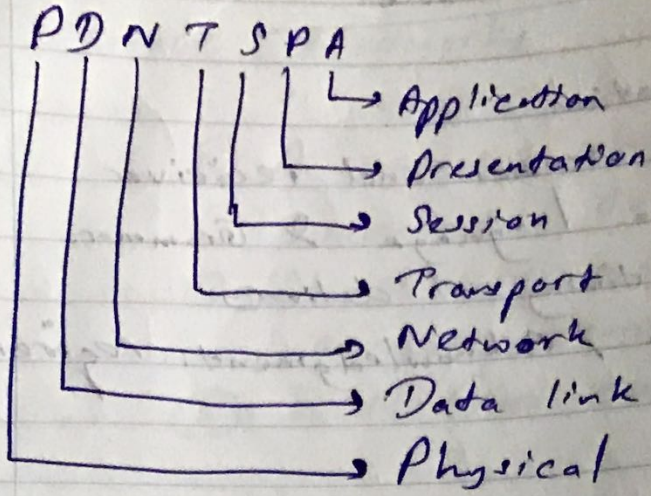
- Rules of communication
  1. Identified sender and receiver
  2. Communication language & Grammar.
  3. Speed & timing of delivery
  4. Confirmation / Acknowledgment. requirements.

- Network protocols



- Reference model for network communication
  1. Defined how applications can communicate over a network.
  2. Guide vendors & developers for interoperate
  3. Layered reference model divides the full process into groups.

- Benefits of using a Layered Model.
  1. Provides a common language
  2. Create a competition between vendors
  3. Changes in one layer do not affect other layers.
  4. Assits in protocol design.



(7) Application Layer
  (1) Interface

(6) Presentation Layer
  (1) Translation (Characters → ASCII) I/O
  (2) Encryption
  (3) Compression

(5) Session Layer
  (1) Dialog control
  (2) Synchronization

(4) Transport Layer
  (1) PORT Number
  (2) Segmentation & Reassembly
  (3) Connection control
  (4) Flow control
  (5) Error control

(Process to process delivery)
(Transport layer address : PORT address)

No:

Date: / /

(3) Network Layer
  (1) Routing
  (2) Logical addressing (IP address)

(2) Data link layer
  (1) Framing - Ethernet framing
  (2) physical addressing (MAC Address)
  (3) Flow control
  (4) Error control
  (5) Access control
  (Data link layer address : MAC address)
  = Hop to Hop
    ↳ phone —Frame→ Router —Fibre→ SLT Colombo
    receiver's Router ↩ Frame
    phone from

(1) Physical layer
  (1) Physical Interfaces transmit and recieve unstructured raw data over the transmission media.
  (2) Converts the digital / Analog bits into electrical / optical signals.

  [To transmit bits over a medium, to provide mechanical and electrical specifications.]