ARYAN SCHOOL OF ENGINEERING & ECHNOLOGY

BARAKUDA,PANCHAGAON,BHUBANESWAR,KHORDHA-752050

LECTURE NOTE

SUBJECT NAME- COMPUTER SYSTEM ARCHITECTURE

BRANCH-COMPUTER SCIENCE ENGG.

SEMESTER-3RD SEM

ACADEMIC SESSION-2022-23

PREPARED BY- HARAPRIYA ROUT

COMPUTER SYSTEM ARCHITECTURE

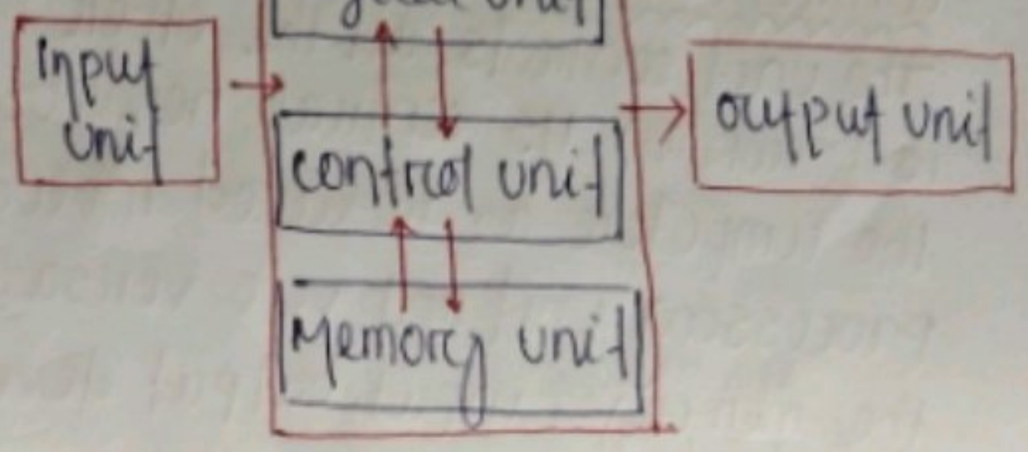
1. Basic structure of computer hardware:

What is computer architecture?

It is a specification describing how hardware and software technologies interact to create a computer platform or system.

Functional units:-

- Input unit
- output unit
- memory unit
- Arithmetic Logic unit (ALU)
- control unit



- Input unit:

* Input unit performs following tasks:
- Accept the data and instructions from the outside environment.
- convert it into machine language.
- supply the converted data to computer system.

- output unit:
- It connects the internal system of a computer to the external environment. It provides the results of any computation, or instructions to the outside world.
- some output devices are printers, monitor etc.

- memory unit:

memory unit or storage unit contains many computer components that are used to store data. It is traditionally divided into primary storage and secondary storage. primary storage is also known as the main memory and is the memory directly accessible by the CPU. The data from secondary storage needs to be brought into the primary storage before the CPU can use it.
secondary storage contains a large amount of data permanently.

- Arcithmetic Logic unit:

All the calculations related to the computer system are perform by the arcithmatic logic unit. It can perform operations like addition, subtraction, multiplication, division etc. The control unit transfers data from storage unit to arcithmatic logic unit when calculations need to be performed. The arcithmatic logic unit and the control unit together form the central processing unit.

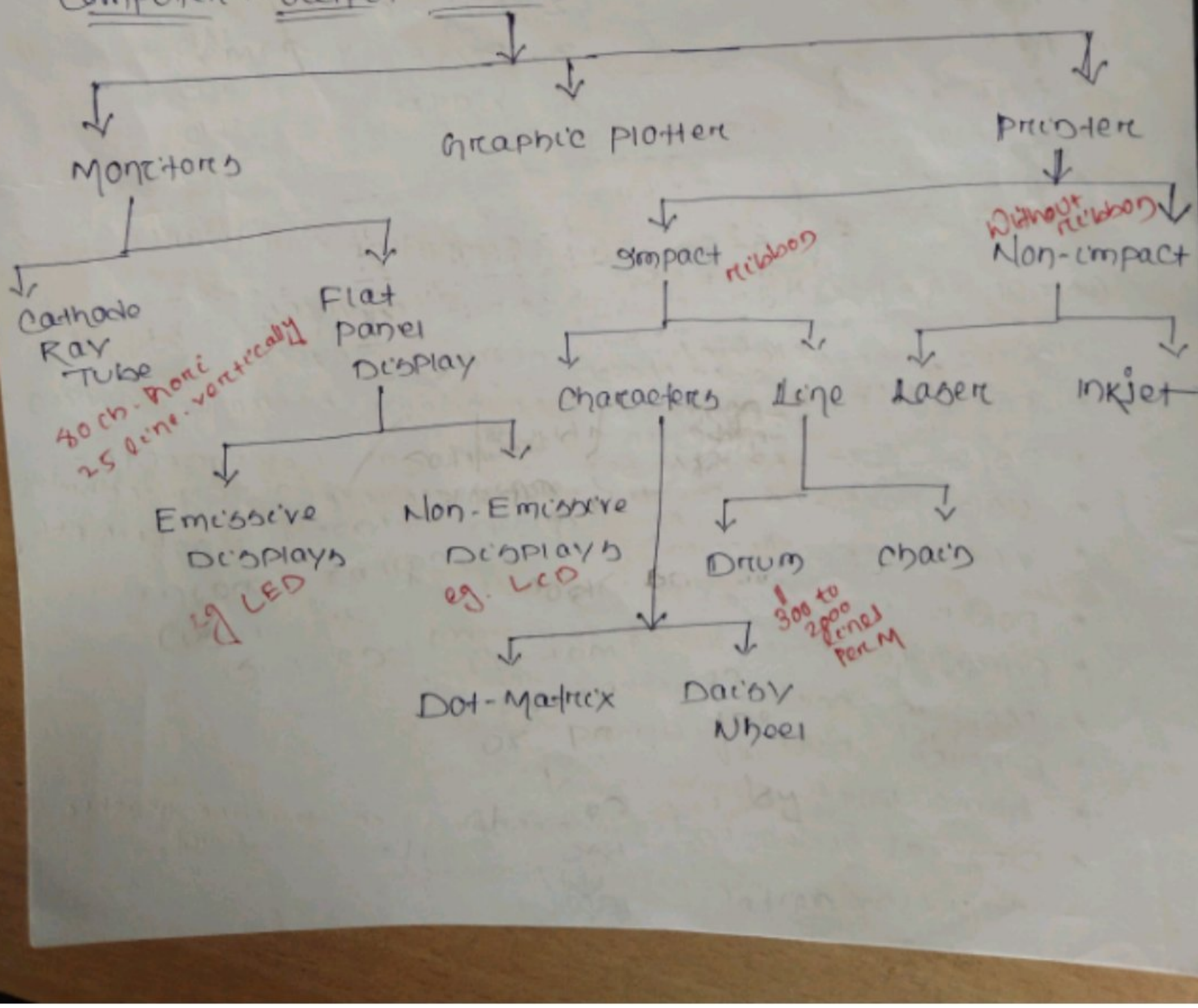
control unit:

The unit controls all the other units of the computer system and so is known as its central nervous system. It transfers data throughout the computer as required including from storage unit to central processing unit and vice versa. The control unit also dictates how the memory, input, output devices, arcithmatic logic unit etc should behave.

Computer - Input devices

* Keyboard
* Mouse
* Joy Stick
* Light pen
* Track Ball
* Scanner
* Graphic Tablet
* Microphone
* Magnetic INK Card Reader(MICR)
* Optical Character Reader (OCR)
* Bar code Reader
* Optical Mark Reader (OMR)

Computer - Output devices



Computer - Memory

Memory is primarily of three types

- Cache Memory
- primary memory / Main memory
- Secondary memory

RAM
- Static RAM
- Dynamic RAM

ROM
- Masked ROM
- PROM
- EPROM
- EEPROM

Computer - Motherboard

* Intel
* ASUS
* Acer
* ABIT
* Biostar
* Gigabyte
* MSI

Computer - Memory units

1. Bit
2. Nibble — KB
3. Byte — MB
4. Word — GB
   TB
   PB

Computer - ports

* Serial port - Modem & older mouse
* Parallel port - Scanner & Printer
* PS/2 port - KB & Mouse
* USB port - hd, Printer, Scanner, Mouse, KB
* VGA port - Monitor
* Power connection - Power cable
* Firewire port - Video equipment
* Modem port - telephone network
* Ethernet port - network cable
* Game port - Joystick
* Digital video interface port - Lcd to video graphic card
* Sockets - Microphone & Speaker

Computer - Hardware

Input devices

Output devices

Secondary storage devices

External components - CPU, motherboard, RAM etc

Computer - Software

* System software
* Application software

Functional units :→

A computer consists of five functionally independent main parts input, memory, ALU, output & control unit.

→ A computer consists of input unit that takes input, a CPU that processes the input and an output unit that produces output. All these devices communicate with each other through a common bus.

Performance measures :→

Aspects of performance. Computer performance metrics include availability, response time, channel capacity, latency, completion time, service time, bandwidth, throughput, relative efficiency, scalability, performance per watt, compression ratio, instruction path length and speed up.

→ CPU performance measured in megahertz or gigahertz and corresponds with how many instruction cycles the CPU can deal with in a second.

→ A 2 GHz CPU performs two Billion cycles a second.

→ Some people increase a CPU clock speed to try to make their computer run faster this is called overclocking.

$$\text{performance} = \frac{1}{\text{Execution time}}$$

Total time taken for execution of a program = CPU Time + I/O time + others

* CPU Time in seconds = $\frac{\text{No of instructions in the program}}{\text{Average number of instructions executed per second by the CPU}}$

* CPU performance for scientific application, vector processing, business app'ic'ations etc
⇒ Instructions per second

* Graphics performance → Rendering - Pixels per second

* I/O performance → Transactions per second

* Internet performance and more ⇒ Bandwidth utilization in Mbps or Gbps

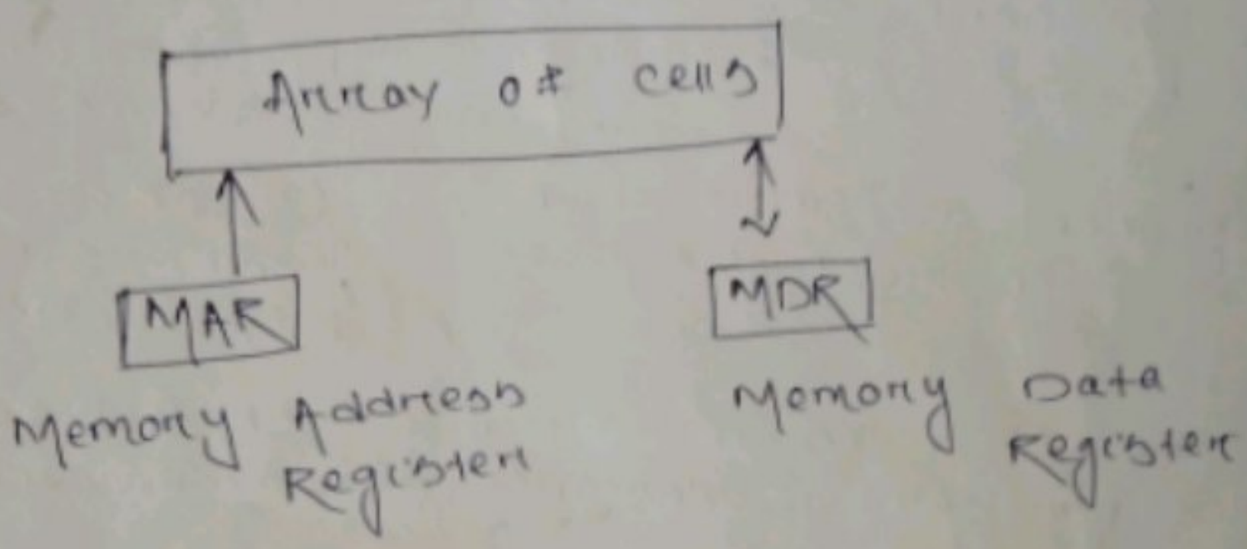
Speedup - Amadhal's Law
performance improvement is achieved by tuning part of Hardware.

$$\text{Speedup (achieved)} = \frac{\text{Execution Time (before improvement)}}{\text{Execution Time (After improvement)}}$$

Memory addressing & operations

- there are two key operations on memory.

1. fetch (address)
   returns value without changing the value stored at that address.
2. store (address, value)
   writes new value into the cell at the given address.

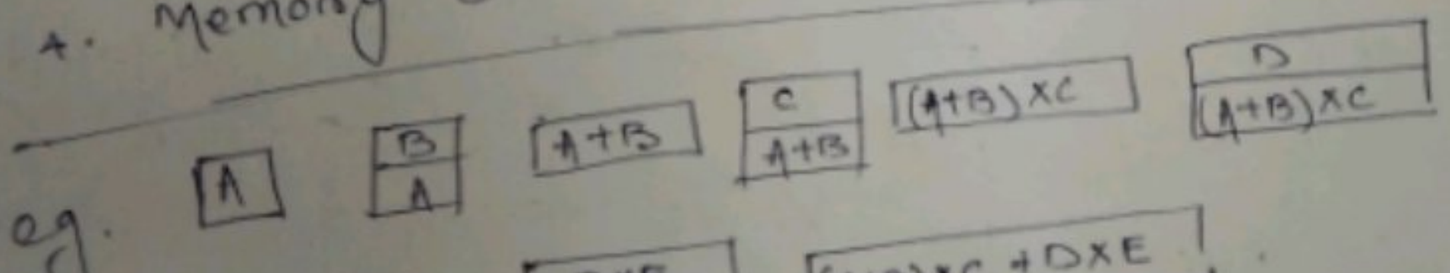


fetch (addr):
1. put addr into MAR
2. Tell memory unit to "load"
3. Memory copies data into MDR

store (addr, new-value):
1. put addr into MAR
2. put new-value into MDR
3. Tell Memory unit to "store"
4. Memory stores data from MDR into memory cell.

---

eg.



PUSH A
PUSH B
ADD
PUSH C
MUL
PUSH D
PUSH E
MUL
ADD

Module-2 18CS54

Instructions & instruction sequencing :-

The tasks carried out by a computer program consist of a sequence of small steps, such as adding two numbers, testing for a particular condition, reading a character from the keyboard, or sending a character to be displayed on a display screen.

⇒ A computer must have instructions capable of performing 4 types of operations.

1. Data transfers between the memory & the registers (MOV, PUSH, POP, XCHG)
2. Arithmatic and logic operations on data (ADD, SUB, MUL, DIV, AND, OR, NOT)
3. program sequencing & control (CALL, RET, LOOP, INT)
4. I/O transfers (IN, OUT)

REGISTER TRANSFER NOTATION (RTN)

* The possible locations in which transfer of information occurs are

1. Memory location
2. Processor register &
3. Registers in I/O device.

Location	Binary Address	Example	Description
Memory	LOC, PLACE, NUM	R1 ← [LOC]	contents of Memory location LOC are transferred to Reg R1
Processor	R0, R1, R2	[R3] ← [R1]+[R2]	Add the register R1 & R2 and places their sum to R3
I/O Register	DATAIN, DATAOUT	R1 ← DATAIN	contents of I/O Register DATAIN are transferred to register R1

2 ASSEMBLY LANGUAGE NOTATION

→ To represent machine instructions and programs, assembly language format is used.

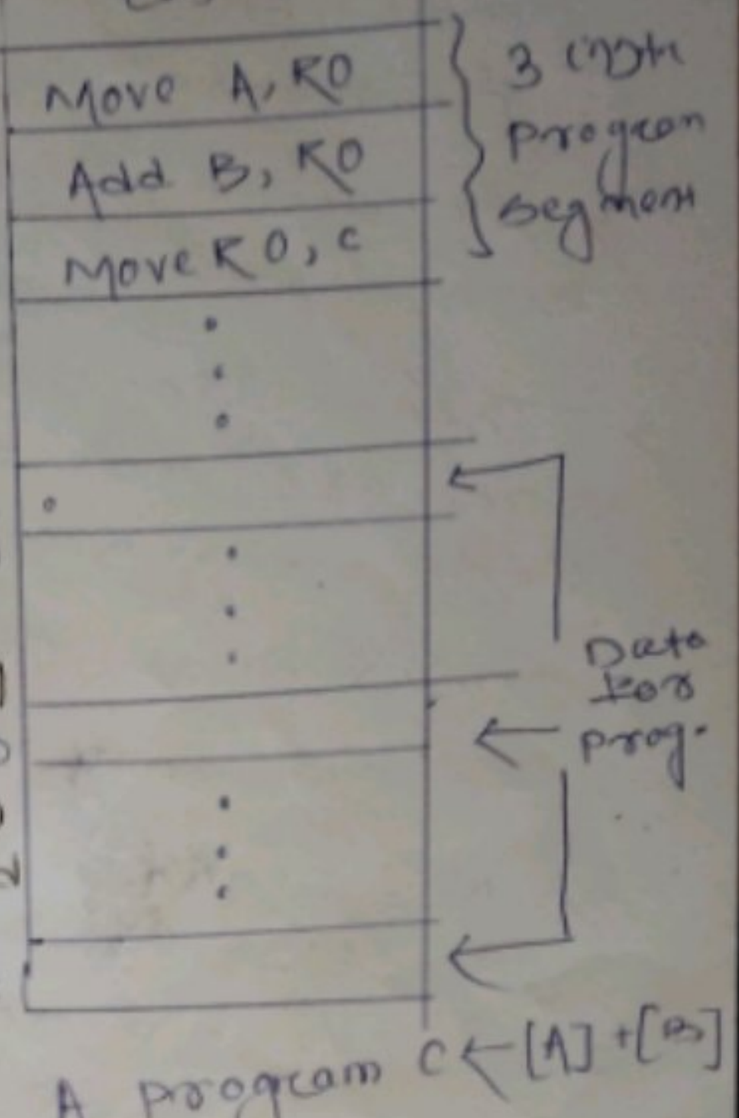
Assembly language format	Description
Move LOC, R1	Transfer data from memory location LOC to register R1. The contents of LOC are unchanged by the execution of this instruction, but the old contents of register R1 are overwritten
Add R1, R2, R3	Add the contents of registers R1 and R2 and places their sum into register R3.

3 BASIC INSTRUCTION TYPES

Instruction type	Syntax	Example	Description	Instruction for operation C ← [A] + [B]
Three Address	Opcode Source1, Source2, Destination	Add A,B,C	Add the contents of memory locations A&B then place the result into location C	
TWO Address	opcode Source, Destination	Add A,B	Add the contents of Memory locations A&B then Place the result into location B, replacing the original contents of this locate	Move B,C Add A,C
One Address	opcode Source/ Destination	Load A Add B	copy contents of memory location A into accumulator	Load A Add B Store C
Zero address	OPCode (no source/ Destination	Store C Push	copy the contents of the accumulator into location C Locations of all operands are defined implicitly. The operands are stored in a pushdown stack.	Not possible

Instruction execution & straight line sequencing :-

1. Initially, the address of the first instruction is loaded into PC
2. Then, the processor control circuits use the information in the PC to fetch & execute instructions, one at a time, in the order of increasing addresses.
3. During the execution of each instruction, PC is incremented by 4 to point to next instruction.



→ There are 2 phases for A program instruction.

Execution

1. Fetch phase :- The instruction is fetched from the memory location and placed in the IR
2. Execute phase :- The contents of IR is examined to determine which operation is to be performed. The specified operation is then performed by the processor

5 Branching :-

- Branch type of instruction loads a new value into program counter.
- So processor fetches & executes instruction at this new address called "branch target".
- Conditional Branch - causes a branch if a specified condition is satisfied.

eg. Branch > 0 LOOP - conditional Branch instruction, it executes only if it satisfies condition.

6. Condition codes :-

→ Recording required information in individual bits called "condition code flags".

→ These flags are grouped together in a Special processor register called "condition Code register" or "status register".

⇒ Individual condition code flags - 1 or 0

⇒ 4 commonly used flags.

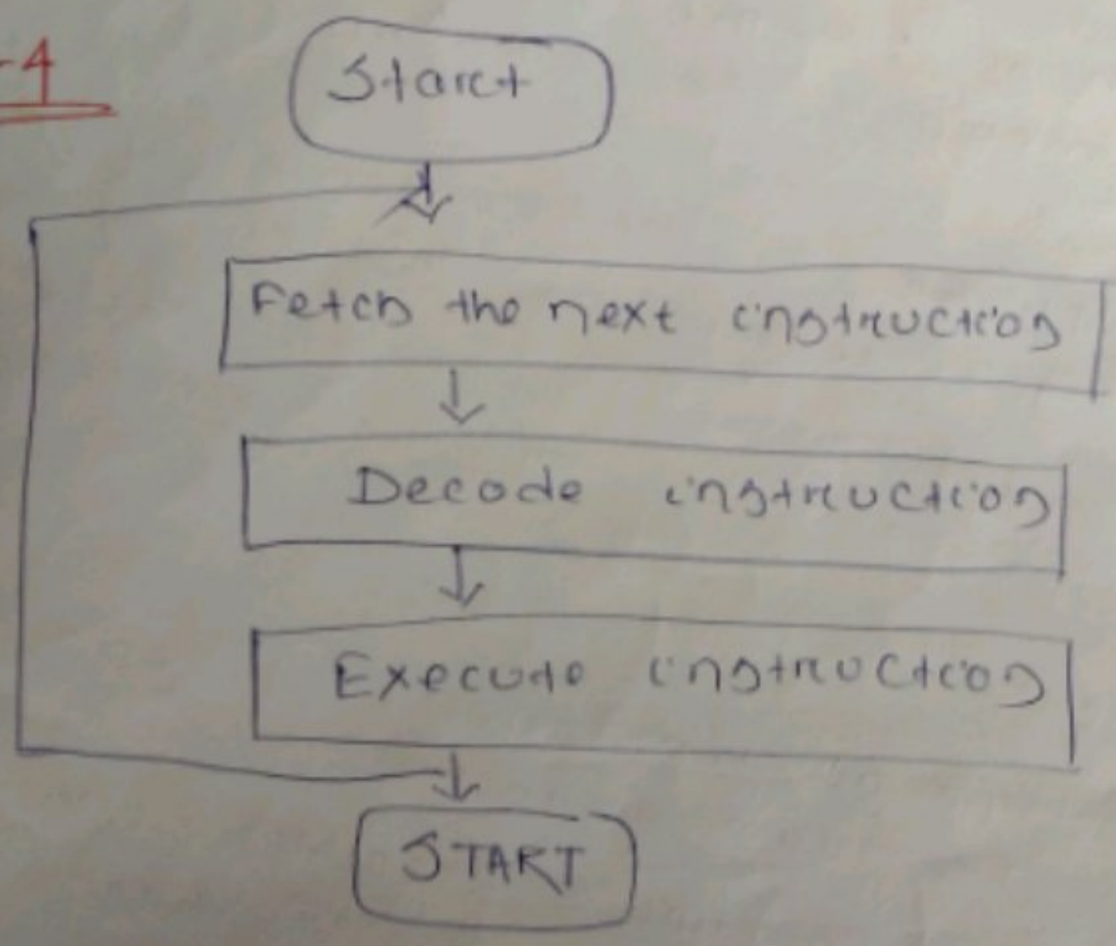
1) N (negative) - set to 1 if result is -ve or else 0.

2) Z (zero) - set to 1 if result is 0, or else 0.

3) V (overflow) - set to 1 if arithmatic overflow occurs or else 0.

4) C (carry) - set to 1 if carry out results from operation or else 0.

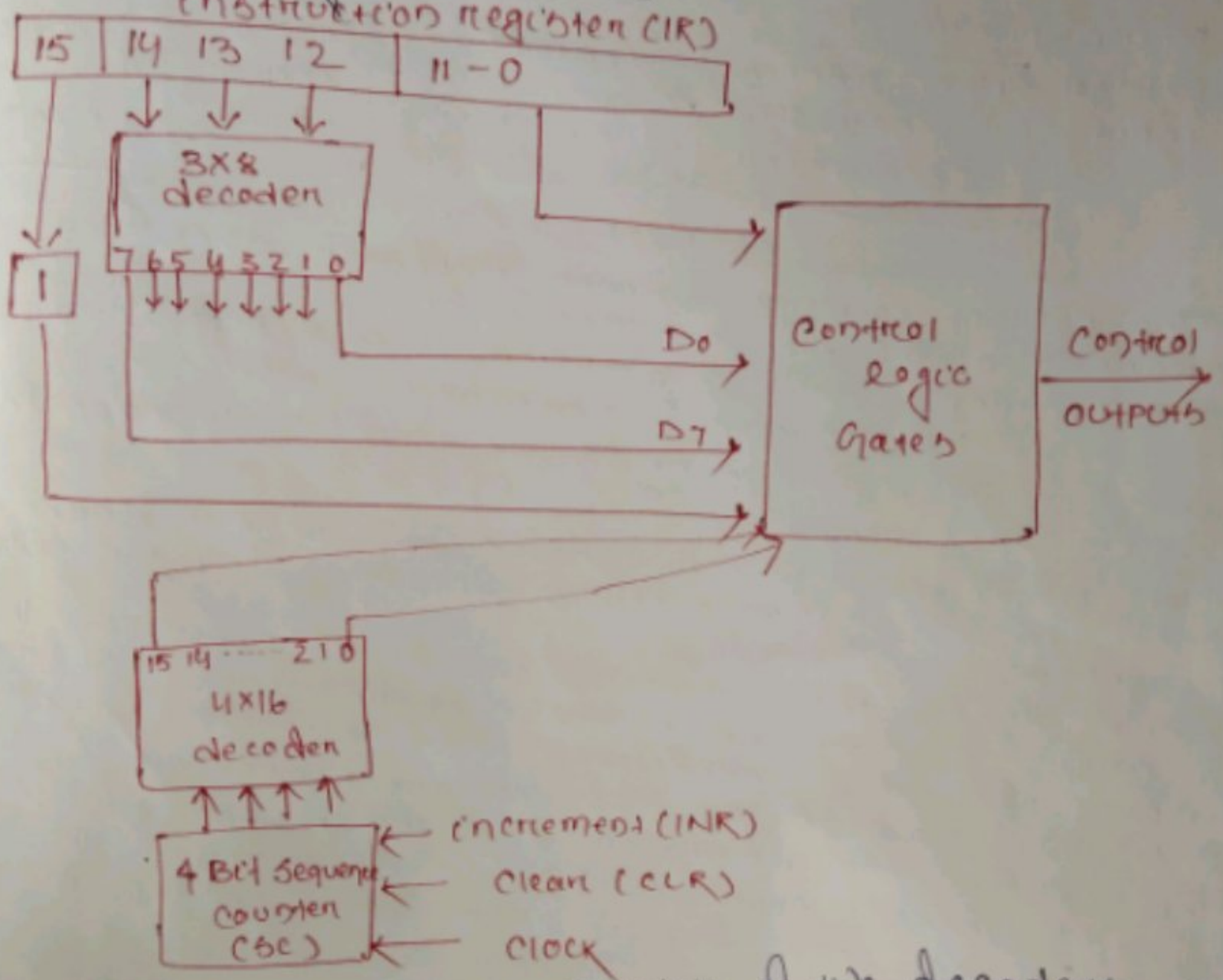
Fig-4



The control unit is classified into two major categories
1. Hardwired control
2. Microprogrammed control

Hardwired control

The Hardwired control organization involves the control logic to be implemented with gates, flip-flops, decoders and other digital circuits.



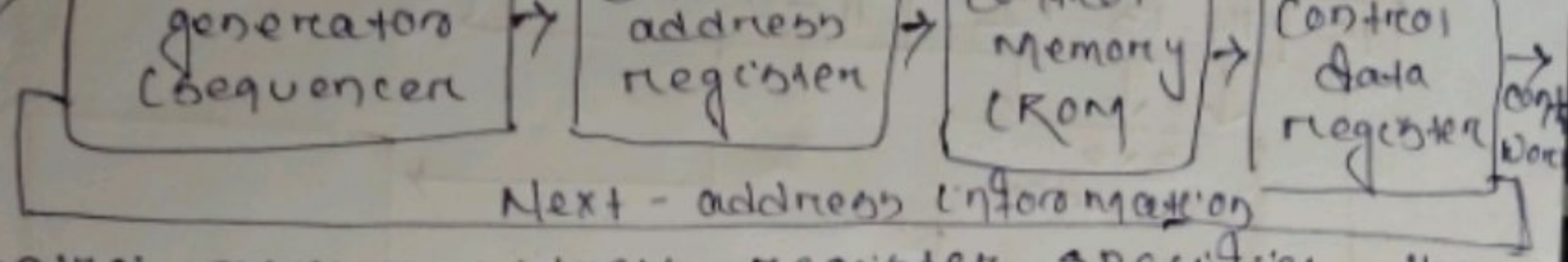
- A Hard-Wired control consists of two decoders, a sequence counter, and a number of logic gates.
- An instruction fetched from the memory unit is placed in the instruction register (IR).
- The component of an instruction register includes: 1 bit, the operation code, and bits 0 through 11.
- The operation code in bits 12 through 14 are coded with a 3x8 decoder.
- The outputs of the decoder are designated by the symbols D0 through D7.
- The operation code at Bit 15 is transferred to a Flip-flop designated by the symbol I.
- The operation codes from Bits 0 through 11 are applied to the control logic gates.

→ The sequence counter (SC) can count in Binary from 0 through 15.

Micro-programmed control :→

The Microprogrammed control organization is implemented by using the programming approach.

→ In Microprogrammed control, the micro-operations are performed by executing a program consisting of micro-instructions.



→ The control memory address register specifies the address of the micro-instruction.

→ The control memory is assumed to be a ROM, within which all control information is permanently stored.

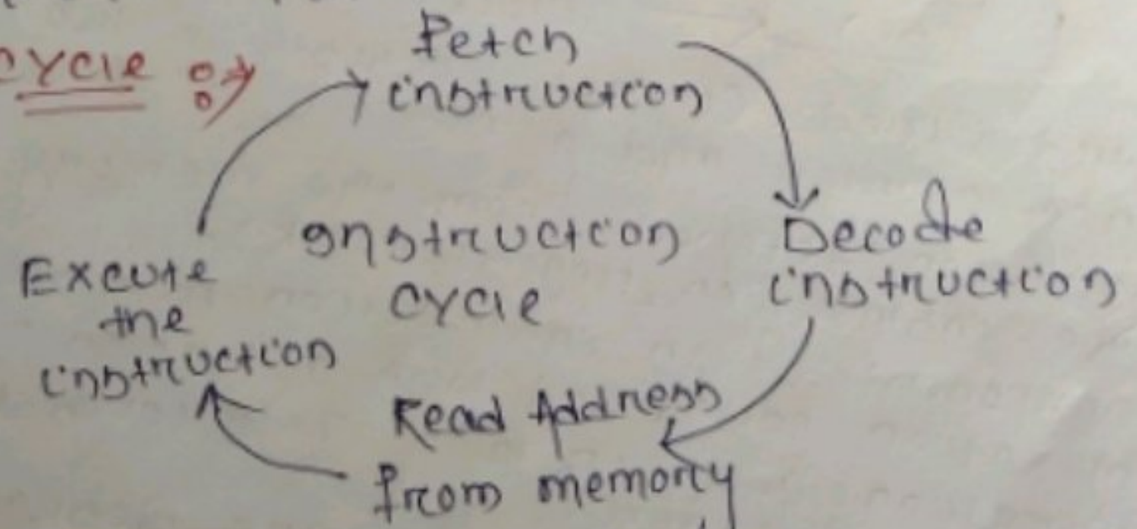
→ The control register holds the microinstruction fetched from the memory.

→ The micro-instruction contains a control word that specifies one or more micro-operations for the data processor.

→ While the micro-operations are being executed, the next address is computed in the next address generator circuit and then transferred into the control address register to read the next microinstruction.

→ The next address generator is often referred to as a micro-program sequencer, as it determines the address sequence that is read from control memory.

Instruction cycle :→



1. Fetch instruction from memory
2. Decode the instruction
3. Read the effective address from memory
4. Execute the instruction

UNIT-4 Memory System

Memory characteristics :⇒

- Primary (Main Memory)
  - RAM
    - Static
    - Dynamic
  - ROM
    - PROM
    - EPROM
    - EEPROM
- Secondary
  - Hard Disk
  - Magnetic tape
  - CD, DVD, zip Disk etc
- Cache
- Registers

→ The two most important characteristics of memory are capacity and performance.

1. location
2. capacity
3. unit of transfer
4. Access Method
5. performance
6. physical type
7. physical characteristics
8. organization

1. **location** :⇒ CPU :-

→ Internal or Main :⇒ This is the main memory like RAM or ROM. The CPU can directly access the main memory.

→ External or secondary :⇒ The CPU doesn't access these devices directly.

2. **capacity** :⇒ Word size, Number of words

eg. If a memory device is given as 4K×16 This means the device has a word size of 16 bits and a total of 4096 (4K) words in memory.

3. **units of transfer** :⇒ It is the maximum number of bits that can be read or written into the memory at a time. In Main memory, it is mostly equal to word size. In external memory, unit of transfer is not limited to the word size.

4. **Access Methods** :⇒ These are three types.
   - ⇒ Random Access
   - ⇒ Serial Access
   - ⇒ Semi random Access

5. performance :→ Access time, Memory cycle time, Transfer rate.

6. physical type :→ Memory devices can be either semiconductor memory (like RAM) or magnetic memory.

7. physical characteristics :→

→ Volatile / Non-volatile :→

8. organization :→ Erasable / Non-erasable.

eg. RAM (erasable), ROM (non-erasable)

* RAM and ROM organization :→

RAM	ROM
Random Access Memory	Read only memory
→ RAM is volatile	→ ROM is non-volatile
→ Allows Reading & Writing	→ Allows Reading only
→ Temporary storage	→ Permanent storage
→ RAM is expensive	→ ROM is cheap
→ performs R&W functions	→ performs R function
→ DRAM & SRAM	→ PROM & EPROM, EEPROM
→ The data in RAM can be modified easily	→ ROM can be hardly or never be modified
→ used in CPU cache, primary memory	→ used in firmware, Microcontrollers.
→ very fast but uses a lot of power	→ Fast and uses very little power.

Cache Memory

⇒ The performance of the cache memory is frequently measured in terms of a quantity called Hit ratio.

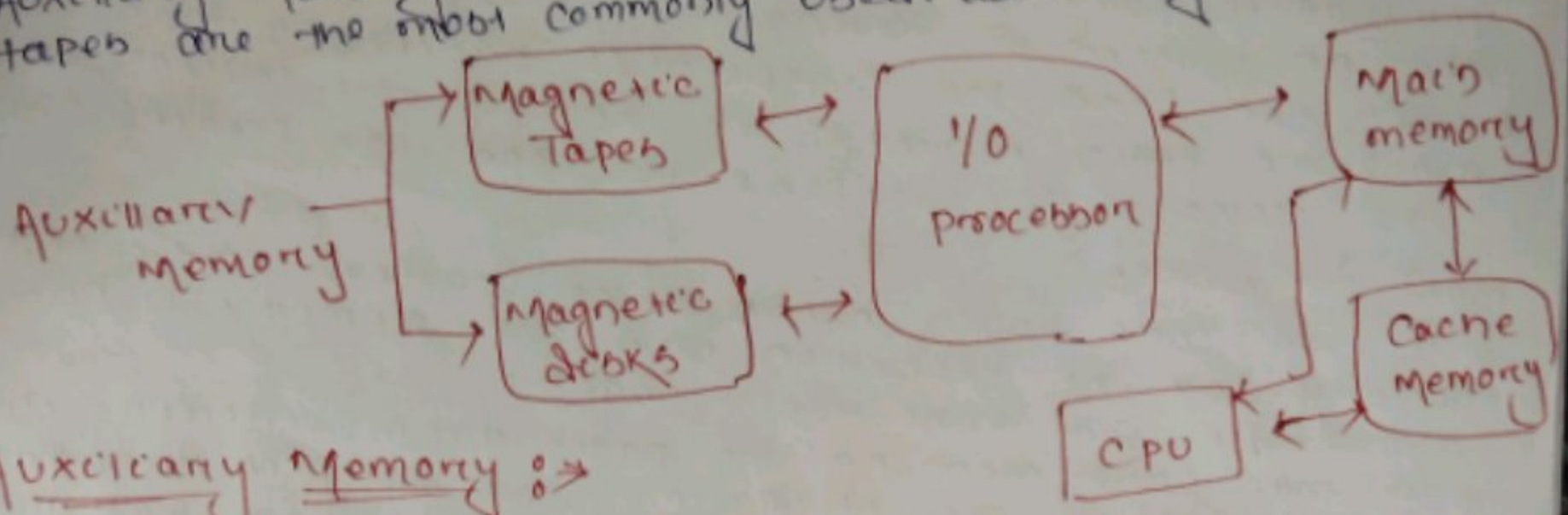
⇒ When the CPU refers to memory and finds the word in cache, it is said to produce a hit.

⇒ If the word is not found in the cache, it is in main memory and it counts as a Miss.

Memory Hierarchy :-

memory unit is an essential component in any digital computer since it is needed for storing programs and data.

- The memory unit that establishes direct communication with the CPU is called main memory. The main memory is often referred to as RAM
- The memory units that provide Backup storage are called Auxiliary memory. For instance, magnetic disks and magnetic tapes are the most commonly used auxiliary memories.



Auxiliary Memory :-

- Auxiliary memory is known as the lowest-cost, highest capacity and slowest-access storage in a computer system.
- Auxiliary memory provides storage for programs & data that are kept for long term storage or when not in immediate use.
  eg. magnetic tapes & magnetic disks
- A magnetic disk is a digital computer memory that uses a magnetization process to write, rewrite & access data, eg. hard drives, zip disks & floppy disks.
- Magnetic tape is a storage medium that allows for data archiving, collection, and Backup for different kinds of data.

Main Memory

The memory unit communicates directly with the CPU and with auxiliary memory devices through an I/O processor.

I/O Processor :-

The primary function of an I/O processor is to manage the data transfers between auxiliary memories and the main memory.

Cache memory :-

The data or contents of the main memory that are used frequently by CPU are stored in the cache memory so that the processors can easily access that data in a

shorten time.

→ Whenever the CPU requires accessing memory, it checks the required data into the cache memory

→ If the data is found in the cache memory, it is read from the fast memory. Otherwise, the CPU moves onto the main memory for the required data.

CPU ⟷ main memory

CPU ↕ cache memory

Virtual memory :→

Virtual memory is a storage scheme that provides user an illusion of having a very Big main memory.

→ User can load the Bigger size processes than the available main memory By having the illusion that the memory is available to load the process.

How it works?

Whenever some pages needs to be loaded in the main memory for the execution and the memory is not available for those many pages,

→ In that case, stopping the pages from entering in the main memory, the OS search for the RAM area that are least used in the recent times or that are not referenced and copy that into the secondary memory to make the space for the new pages in the main memory.

Demand paging

Demand paging is a popular method of virtual memory management. In demand paging, the pages of a process which are least used, get stored in the secondary memory.

→ A page is copied to the main memory when its demand is made or page fault occurs. There are various page replacement algorithms which are used to determine the pages which will be replaced

Advantages of Virtual Memory

1. The degree of Multiprogramming will be increased.
2. User can run large application with less real RAM.
3. There is no need to buy more memory RAMs.

Disadvantages of virtual memory :⇒

→ The system becomes slower since swapping takes time

→ It takes more time in switching between applications

→ The user will have the lesser Hard disk space for its use.

Interleaved Memory :→ Why do we use Memory interleaving?

→ When the processor requests data from the main memory a block (chunk) of data is transferred to the cache and then to processor.
→ So whenever a cache miss occurres, the data is to be fetched from the main memory.
→ But main memory is relatively slower than the cache. So to improve the access time of the main memory, interleaving is used.

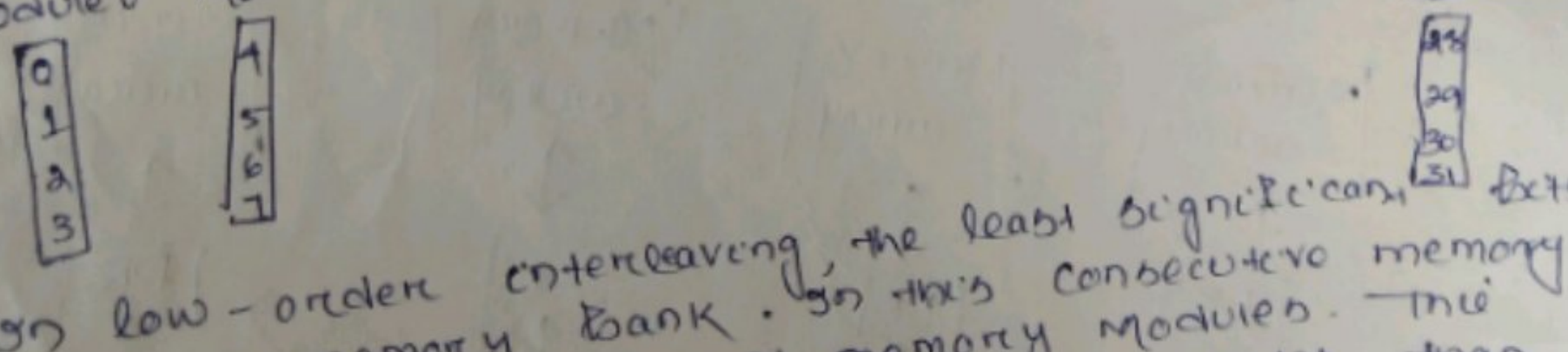
Types of interleaved Memory → High order interleaving
→ Low order interleaving

→ Memory interleaving is an abstraction technique which devides memory into a number of Modules. such that successive words in the address space are placed in the different Module.

1. High order Interleaving :→
→ In high-order interleaving, the most significant bits of the address select the memory chip.
→ The least significant Bits are sent as addresses to each chip.
→ one problem is that consecutive addresses tend to be in the same chip.
→ The Maximum rate of data transfer is limited by the memory cycle time.
It is also known as memory Banking.

Address bits	25-22	21-0
Use	Bank Select	Address to the chip

Module 0 Module 1 Module 7



2. In low-order interleaving, the least significant bits select the memory bank. In this consecutive memory addresses are in different memory Modules. This allows memory access at much faster rates than allowed by the cycle time.

Address bits	25-4	3-0
USO	Address to the CDLP	Bank select

Module 0 | Module-1 | Module-2 | Module-3 | Module-4 | Module-5 | Mod-6 | Mod-7

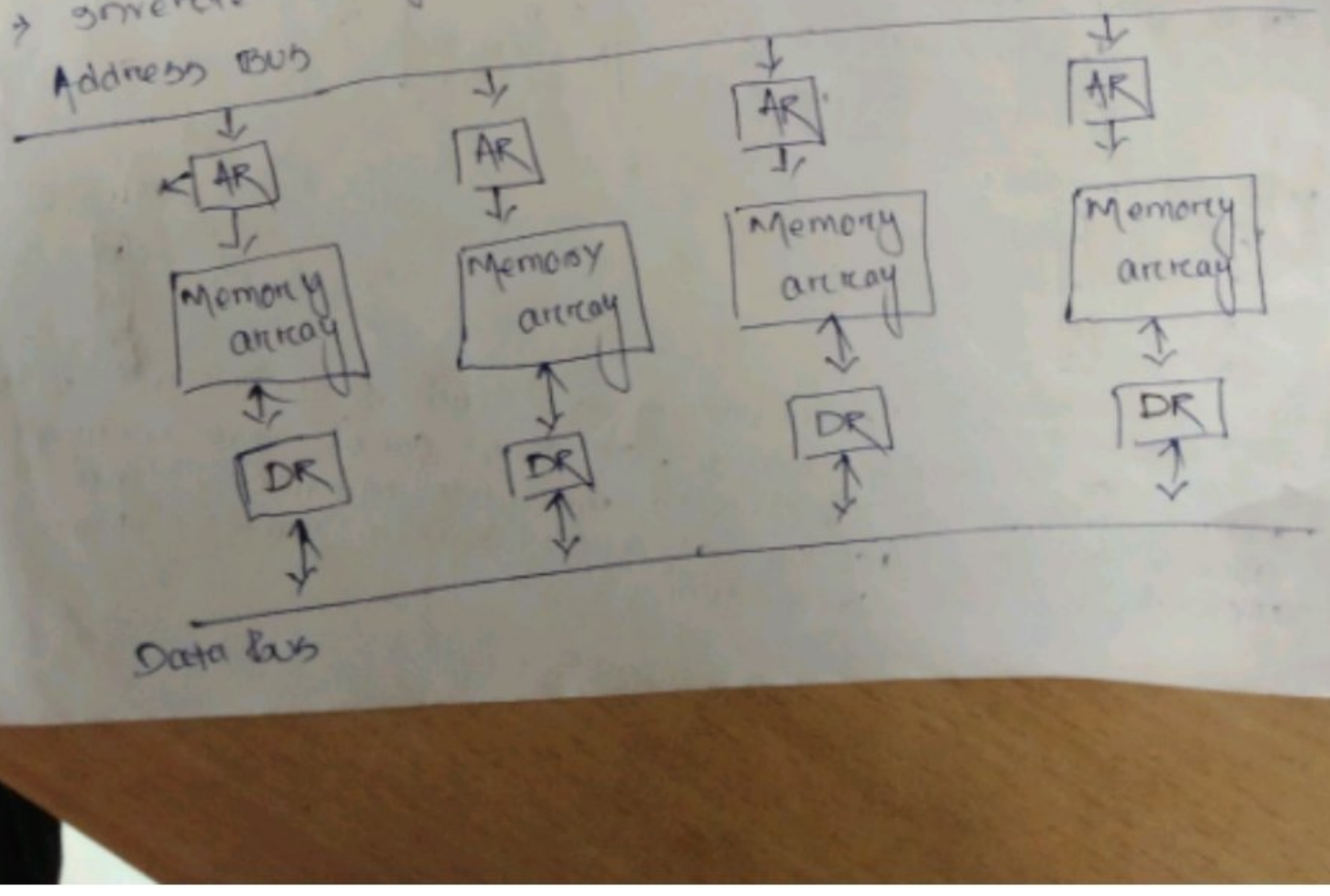
Module 0	Module-1	Module-2	Module-3	Module-4	Module-5	Mod-6	Mod-7
0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

Advantages

- OS can take first one out of list it keeps
- Eliminates external fragmentation
- Data can be scattered all over
- Allows demand paging & prepaging
- More efficient swapping
- No need for considerations about fragmentation
- Just swap out page least likely to be used.

Disadvantages :-

- Longer memory access times
- can be improved using TLB
- Wasted page tables
- Inverted page tables



UNIT-5

ER, Snehalata Raut

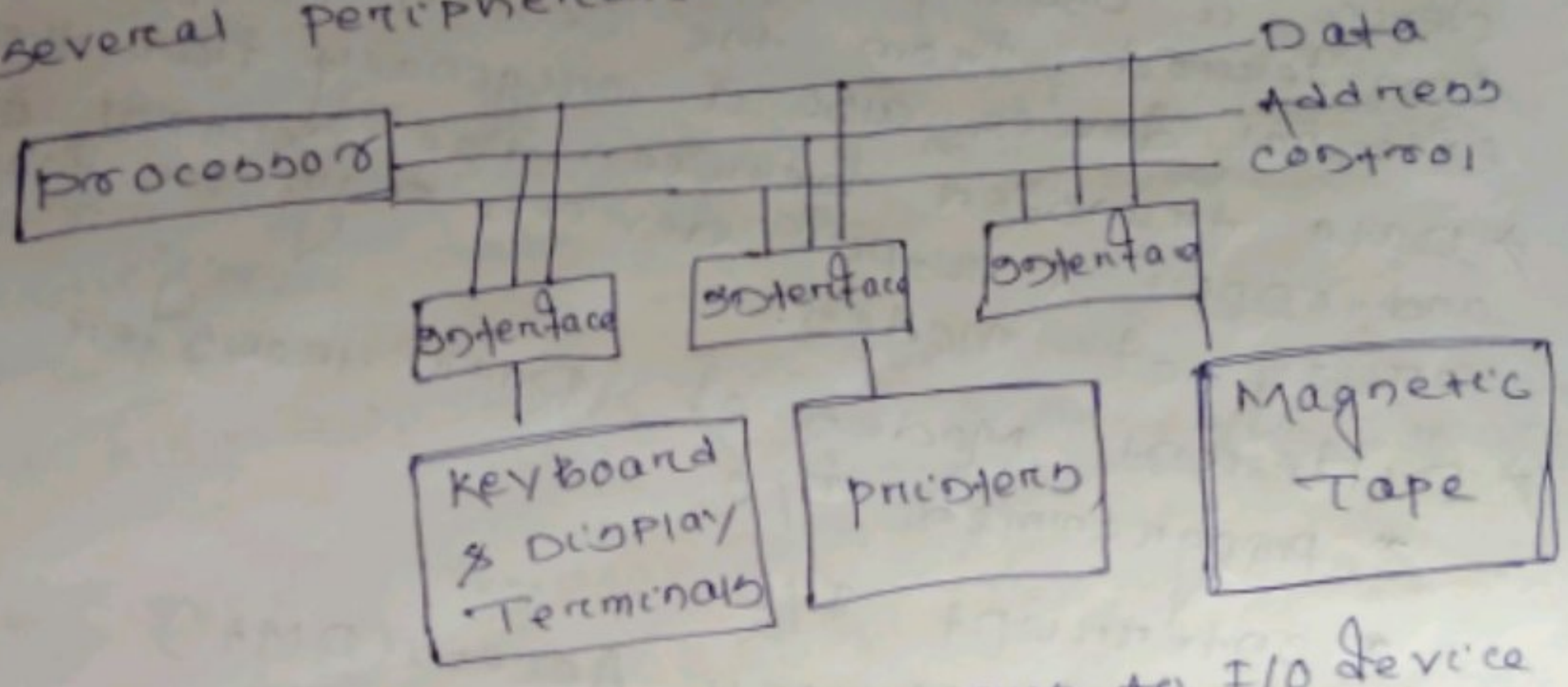
INPUT-OUTPUT SYSTEM :

Input-output interface provides a method for transferring information between processor and external input/output devices.
i.e peripherals.

→ These input/output interface in computer system are exists in the form of special hardware component between the system Bus and peripherals.

→ This component is called "interface unit".

→ input-output interface between the processor and several peripherals



Connection of I/O BUS to I/O device

Why input/output interface is required for transferring information between processors and peripherals?

The input/output interface is required because there are exists many differences between the central computer & each peripheral while transferring information.

→ peripherals are electromechanical & electromagnetic devices and their manner of operation is different from the operation of CPU and memory, which are electronic device. Therefore, a conversion of signal values may be required.

→ The data transfer rate of peripherals is usually slower than the transfer rate of CPU, and

consequently a synchronisation mechanism is needed.

3. Data codes and formats in peripherals differ from the word format in the CPU and memory.

4. The operating modes of peripherals are different from each other and each must be controlled so as not to disturb the operation of other peripherals connected to CPU.

Modes of data transfer :→

1. Binary information received from an external device is usually stored in memory. Information transferred from the central computer into an external device also is originally from the memory.

→ Data transfer between the central computer and input & output devices may be handle in a variety of modes.

→ Different Modes of Data transfer
  * Programmed I/O
  * Interrupt I/O
  * Direct Memory Access (DMA)
  * Programmed I/O

→ Programmed I/O :→ These operations are a result of I/O instructions written in the computer program.

→ Data transfer is initiated by an instruction in the program.

→ Usually the data transfer data between CPU register and peripheral has to be constantly monitored.

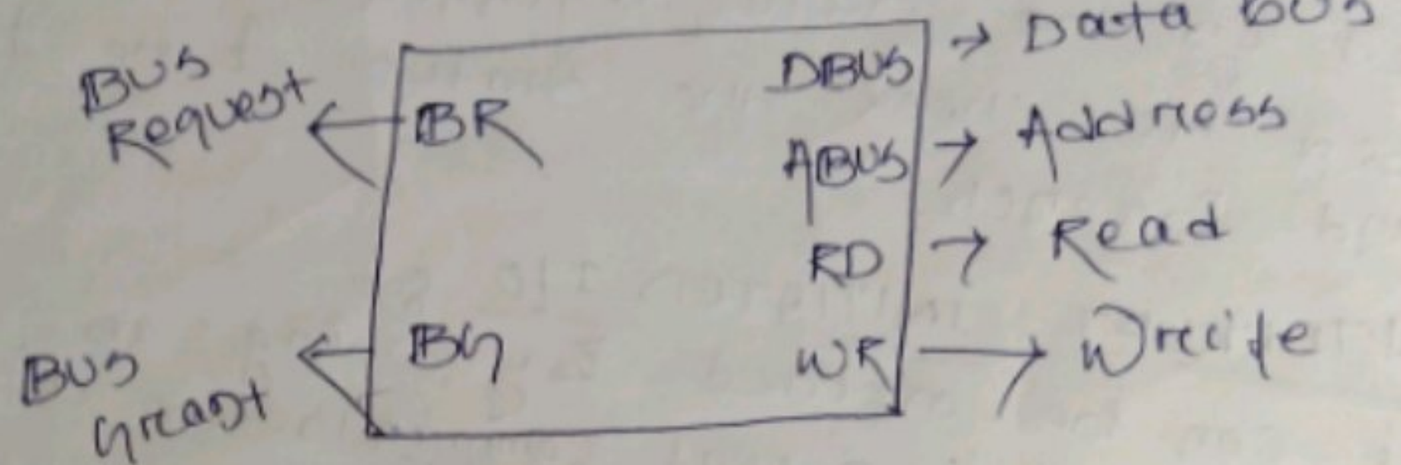
→ Once a data transfer is initiated the CPU is required to monitor the interface to see when a transfer can again be made.

* Application of Programmed I/O Method :-
→ useful in small low speed computers
→ used in systems that are dedicated to monitor a device continuously.
→ used in the data register
→ used to check the status of the flag bit and Branch.

* INTERRUPT INITIATED I/O :-
This can be avoided by using an interrupt facility and special commands to inform the interface to issue an interrupt request signal when the data are available from the device.
→ The interfaces keeps monitoring the device, when the interface determines that the device is ready for data transfer, it generated an interrupt request to the computer.
* It is also two types ⟨ vectored interrupt / Non vectored interrupt

* DMA (DIRECT MEMORY ACCESS)
During DMA transfer, the CPU is idle & has no control of the memory Buses.
→ The Buses can be disabled by using two Special control signals
→ BUS Request or BUS Grant (1BG) data Bus Address Bus Read Write DBUS ABUS RD WR BR BG BUS request BUS grant.
→ The interface transfer data into and out of the memory unit through the memory Bus.
The CPU initiates the transfer at supplying the interface with the starting address and the number of words needed to be transferred and then proceed to execute the other tasks

→ When the request is granted by the memory controller, the DMA transfer the data directly into memory.



BUS Request :→ It is used by the DMA controller to request the CPU to relinquish the control of the buses

BUS Grant :→ It is activated by the CPU to inform the external DMA controller that the buses are in high impedance state & the requesting DMA can take control of the buses.

→ Once the DMA has taken the control of the buses it transfers the data.

Types of DMA transfer using DMA controller :→

Burst Transfer :→

DMA returns the bus after complete data transfer. A register is used as a byte count, being decremented for each byte transfer, and upon the byte count reaching zero, the DMAC will release the bus.

→ When the DMAC operates in burst mode, the CPU is halted for the duration of the data transfer.

Steps involved are

1. Bus grant request time
2. Transfer the entire block of data at transfer rate of device because the device is usually slow than the speed at which the data can be transferred to CPU

3. Release the control of the bus back to CPU
So, total time taken to transfer the N bytes.

= BUS grant request time + (N)* (memory transfer time) + BUS release control time.

cycle stealing

An alternative method in which DMA controller transfers one word at a time after which it must return the control of the buses to the CPU.

→ The CPU delays its operation only for one memory cycle to allow the direct memory I/O transfer to "steal" one memory cycle.

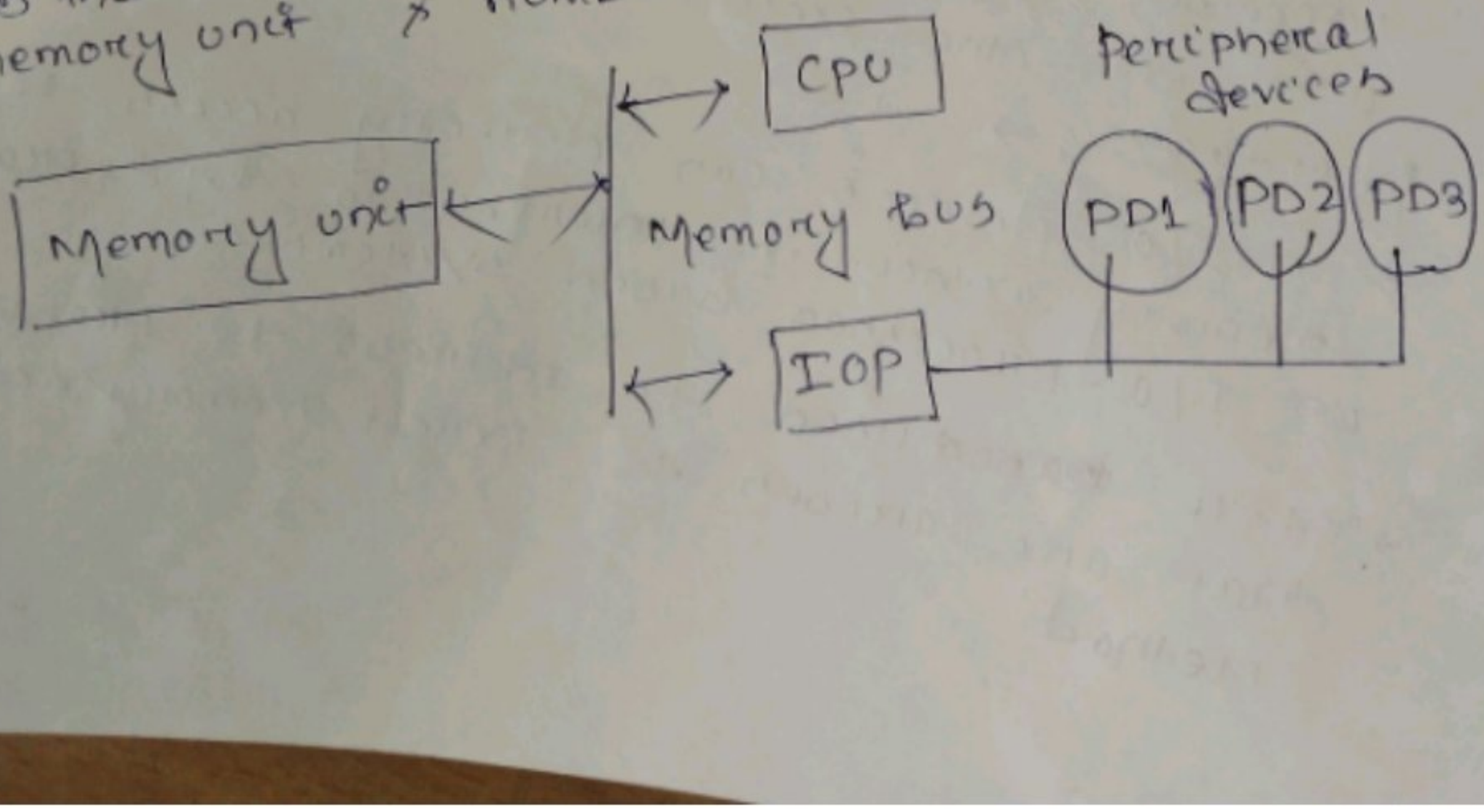
→ In cycle stealing mode we always follow pipelining concept that when one byte is getting transferred then Device is parallel preparing the next Byte.

Interleaved mode :-

In this technique, the DMA controller takes over the system bus when the microprocessor is not using it.

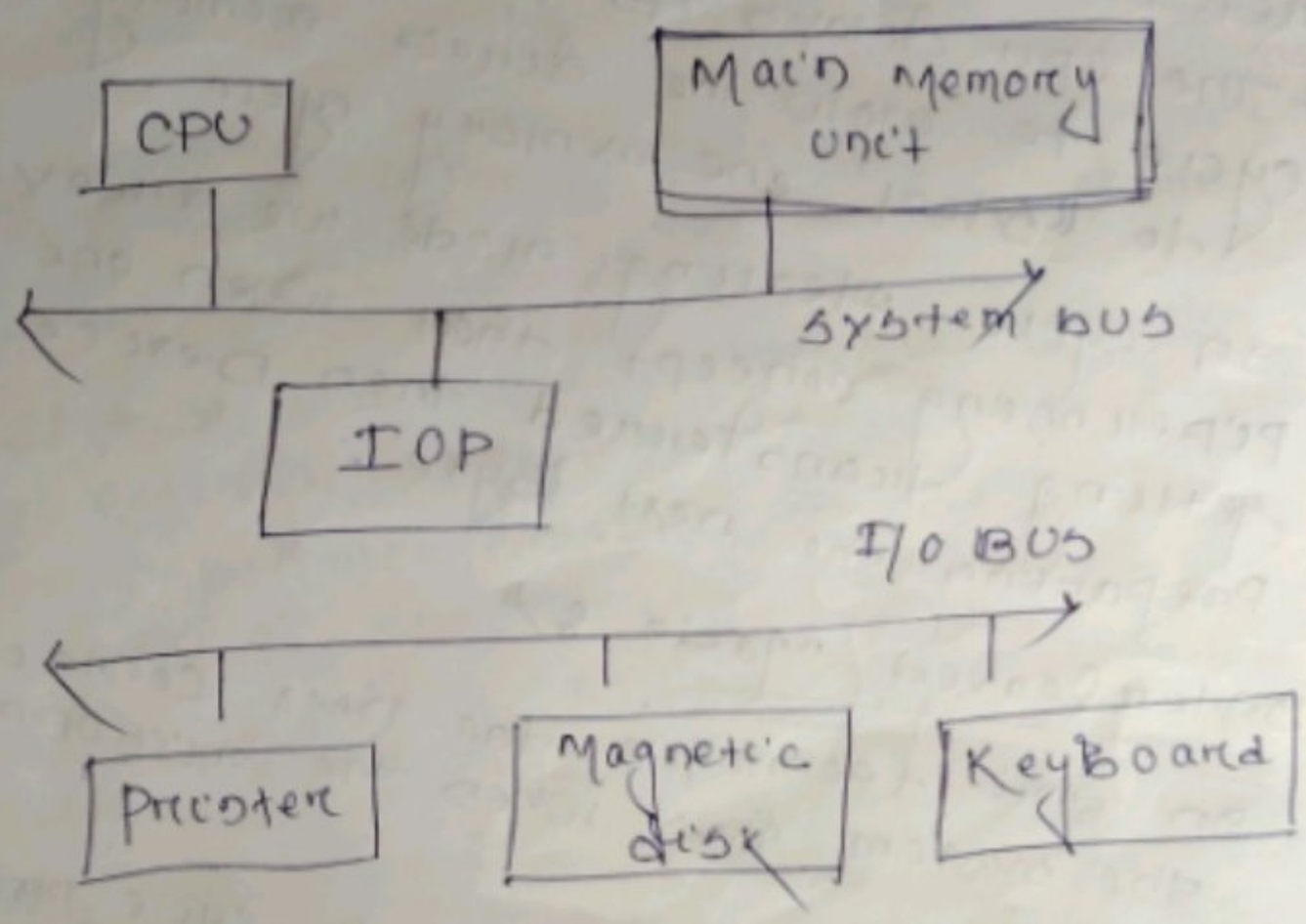
Ex half cycle DMA + half cycle processor

→ IOP is a processor with DMA capability. In this the Computer system is divided into a memory unit & number of processors.



I/O Processor :→ Many computer combines the interface logic with the requirements for the direct memory access into one unit and call it an I/O processor.

→ The I/O can handle many peripherals through a DMA and interrupt facility. The computer is divided into three separate modules on such a system :- 0 Memory unit, 0 CPU, 0 IOP.



Input-Output processor

→ The input output processor is a specialized processor which loads & stores data into memory along with the execution of I/O instructions.

Advantages :→

→ By I/O devices can directly access the Main memory without the intervention by the processor in I/O processor based systems.

→ It is used to address the problems that are arises the Direct Memory access Method.

UNIT-7

Parallel Processing

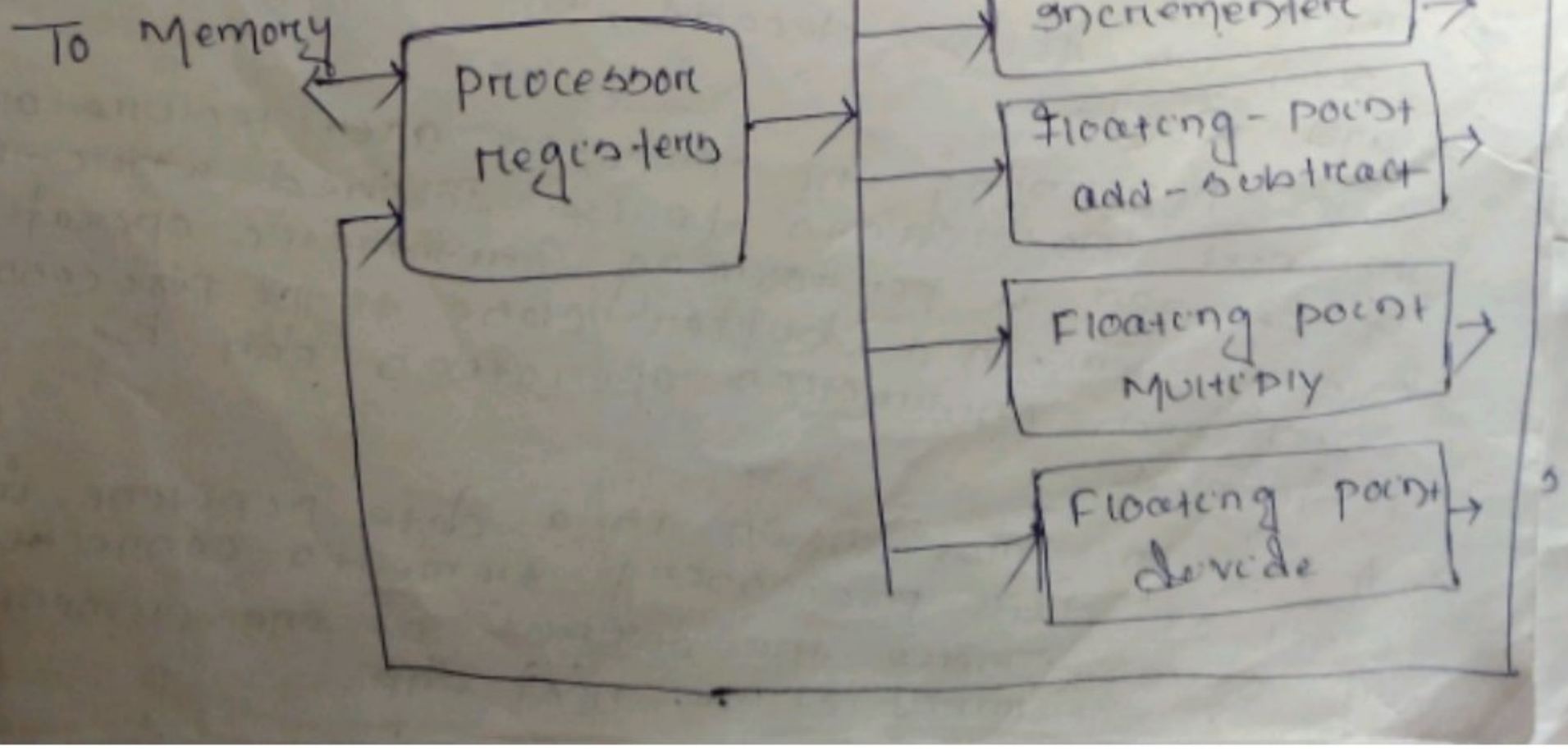
→ Parallel processing can be described as a class of techniques which enables the system to achieve simultaneous data-processing tasks to increase the computational speed of a computer system.

→ A parallel processing system can carry out simultaneous data-processing to achieved faster execution time.

→ While an instruction is being processed in the ALU component of the CPU, the next instruction can be read from memory.

→ The primary purpose of parallel processing is to enhance the computer processing capability and increase its throughput, i.e the amount of processing that can be accomplished during a given interval of time.

→ The following diagram shows one possible way of separating the execution unit into eight functional units operating in parallel.



→ The adder and integer multiplier performs the arithmetic operation with integer numbers.

→ The floating-point operations are separated into three circuits operating in parallel.

→ The logic, shift, and increment operations can be performed concurrently on different data.

→ All units are independent of each other, so one number can be shifted while another number is being incremented.

Linear pipeline :→

Linear pipeline is a pipeline in which a series of processors are connected together in a serial manner.

→ In linear pipeline the data flows from the first block to ~~the~~ final block of processor.

→ Linear pipeline are static pipeline because they are used to perform fixed functions.

→ Non-linear pipeline are dynamic pipeline because they can be reconfigured to perform variable functions at different times.

→ In pipelining, we divide a task into set of subtasks.

→ There are five stages of pipelining such as fetch, decode, execute, memory, and write.

→ With pipelining the computer architecture allows the next instructions to be fetched while the processor is performing arithmetic operations, holding them in a buffer close to the processor until each instruction operation can be performed.

→ A pipeline, also known as a data pipeline, is a set of data processing elements connected in series, where the output of one element is the input of the next one.

Flynn's Classification

M.J. Flynn proposed a classification for the organization of a computer system by the number of instructions and data items that are manipulated simultaneously.

→ The sequence of instructions read from memory constitutes an instruction stream.

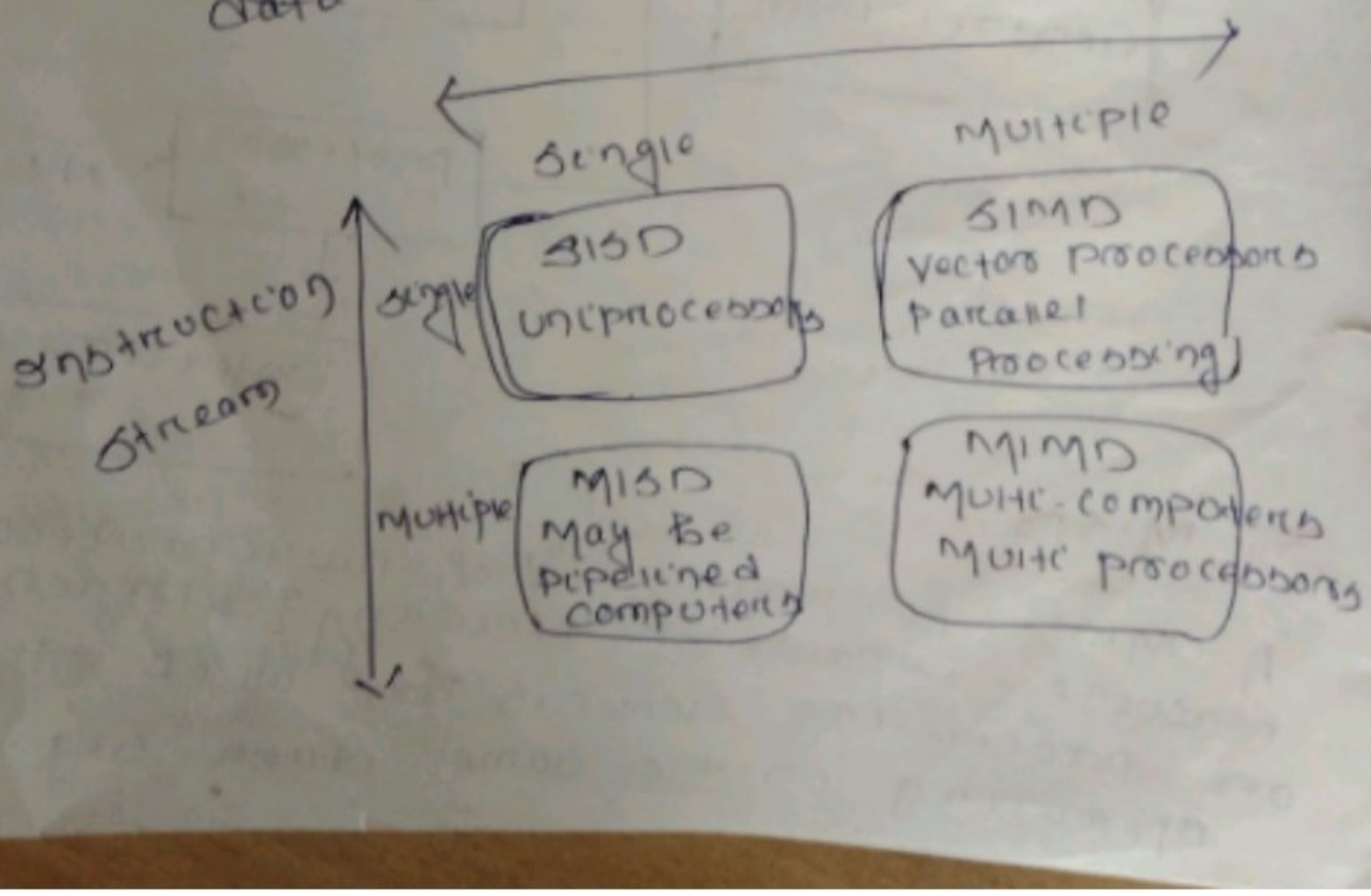
→ The operations performed on the data in the processor constitute a data stream.

(Stream refers to the flow of instructions or data)

Parallel processing may occur in the instruction stream, in the data stream, or both.

Flynn's classification divides computers into four major groups that are:

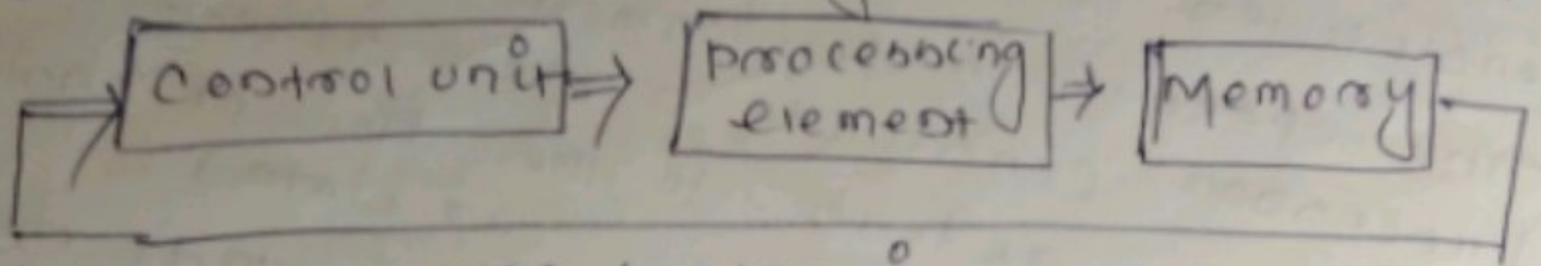
1. Single instruction stream, single data stream (SISD)
2. Single instruction stream, multiple data stream (SIMD)
3. Multiple instruction stream, single data stream (MISD)
4. Multiple instruction stream, multiple data stream (MIMD)



1. SISD :→

→ Single instruction :→ only one instruction stre
is being acted on executed by CPU during
one clock cycle.

→ Single data stream :→ only one data stream is
used as input during one clock cycle.

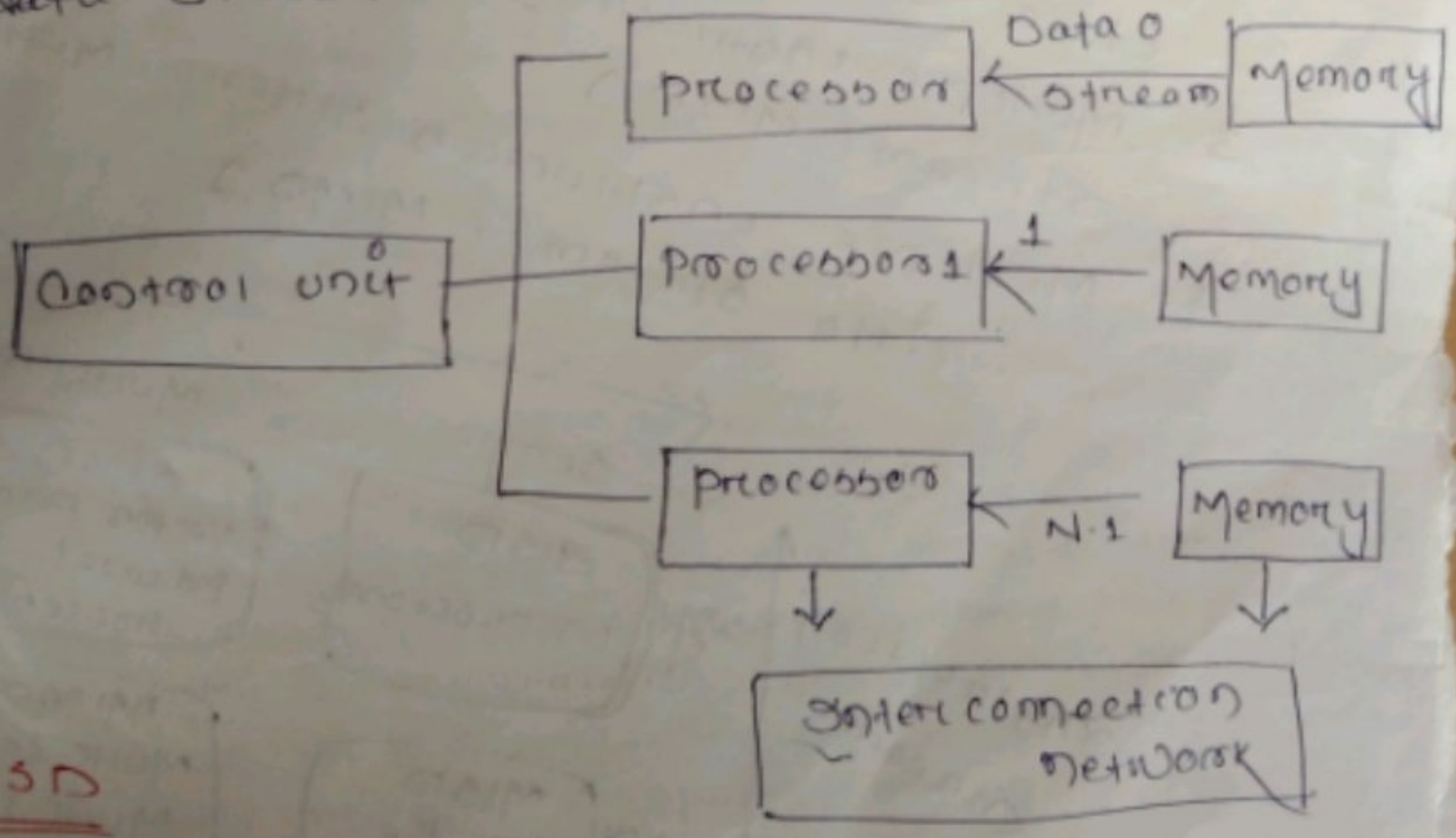


A SISD computing system is a uniprocessor machine that is capable of executing a single instruction operating on a single data stream.

→ SISD architecture where all the instruction and data to be processed have to be stored in primary memory.

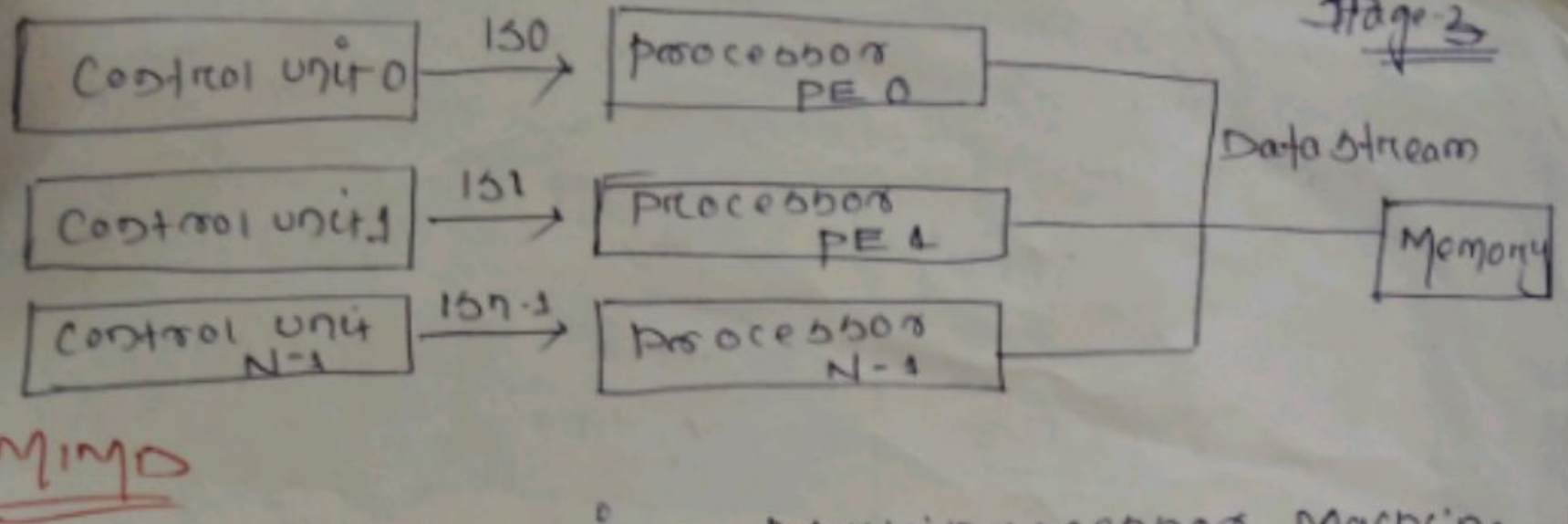
2. SIMD :→

A SIMD system is a multiprocessor machine, capable of executing the same instruction on all the CPUs but operating on the different data stream.



3. MISD

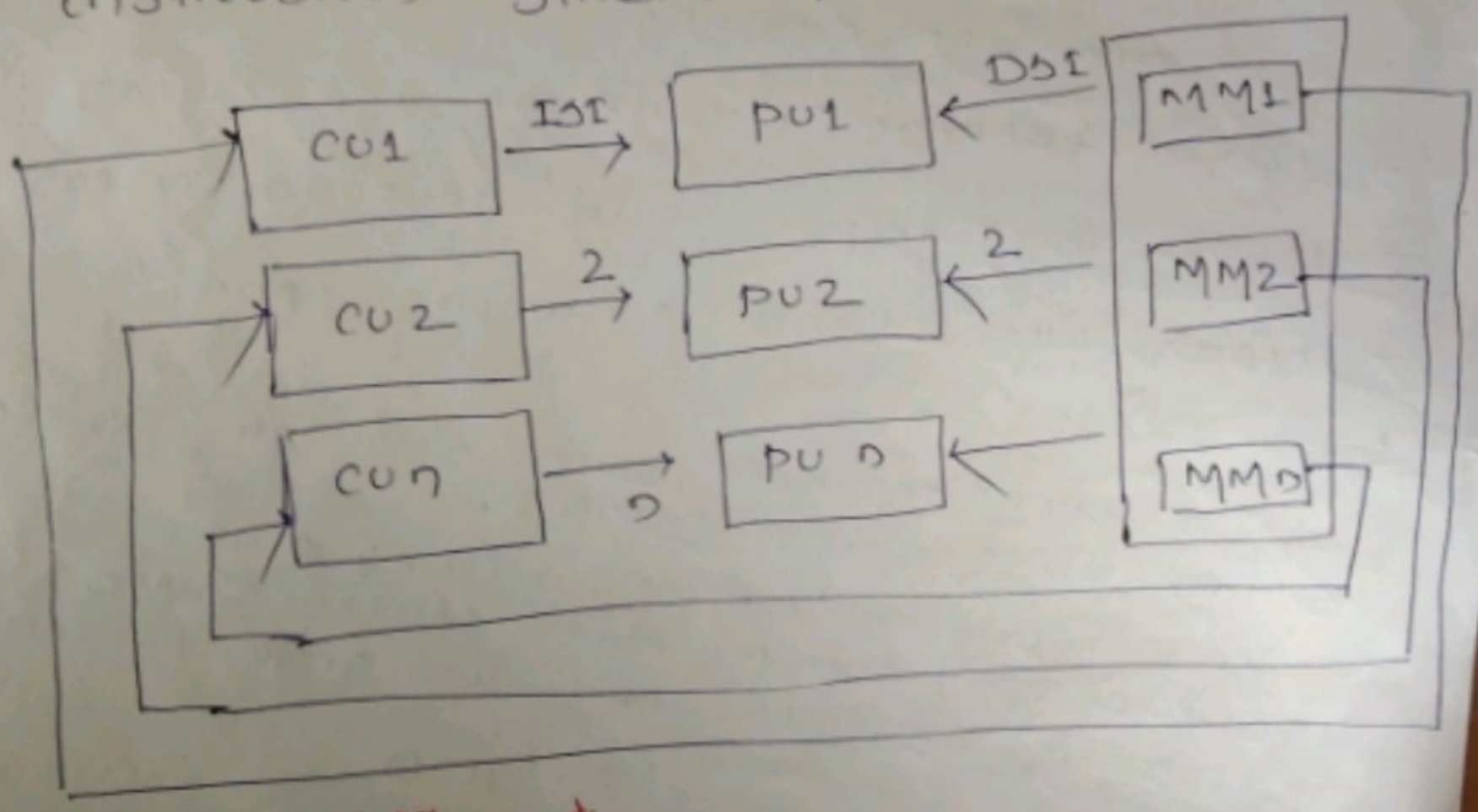
A MISD computing is a multiprocessor machine capable of executing different instruction on processing elements but all of them operating on the same data set.



4. MIMD

A MIMD system is a multiprocessor machine that is capable of executing multiple instructions over multiple data streams.

→ Each processing element has a separate instruction stream and data stream.



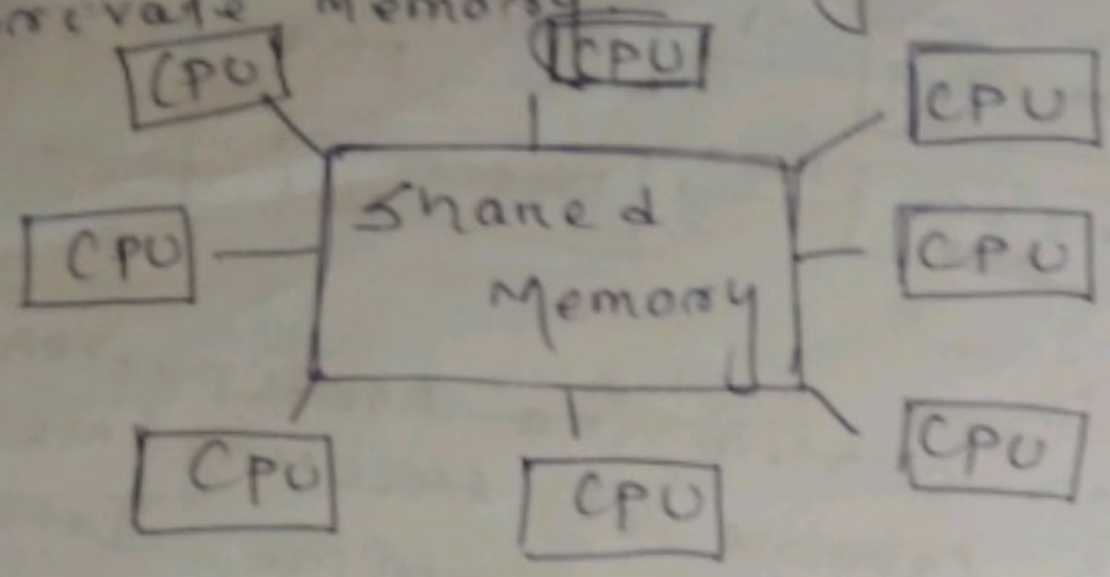
Multiprocessor :→

A multiprocessor is a computer system with two or more central processing units (CPUs) share full access to a common RAM.

→ Multiprocessor is to boost the system's execution speed, with other objectives being fault tolerance & application matching.

→ There are two types of multiprocessors, one is called shared memory multiprocessor and another is distributed memory multiprocessor.

→ In shared memory multiprocessors, all the shared the common memory but in a distributed memory multiprocessors, every CPU has its own private memory



Benefits

→ enhanced performance
→ multiple applications
→ multi-tasking inside an application
→ High throughput and responsiveness.
→ Hardware sharing among CPUs.

Applications

1. As a uniprocessor, such as single instruction single data stream (SISD).
2. As a multiprocessor, such as single instruction, multiple data stream (SIMD), which is usually used for vector processing.
   → MISD
   → MIMD

→ A Mode of operation in which two or more processors in a computer simultaneously process two or more different portions of the same program.

→ It is an interconnection of two or more CPU with memory & input-output equipment

Ex UNIX operating system