

Advanced Computer Application



**Bachelor Of Computer Application
(B.C.A)**
M.J.P.R.U (Bareilly)

Prepared By:
Priyank Chaudhary
(Asst.Professor,C.S Dept.)

Parallel Processing

Parallel Processing is an efficient form of information processing, which emphasize the exploitation of concurrent events in the computing process.

(Concurrency emphasizes simultaneity)

Parallel events may occur in multiple resources during the same time interval, simultaneous event may occur at the same time instant.

These concurrent events are available in a computer system at various processing levels.

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

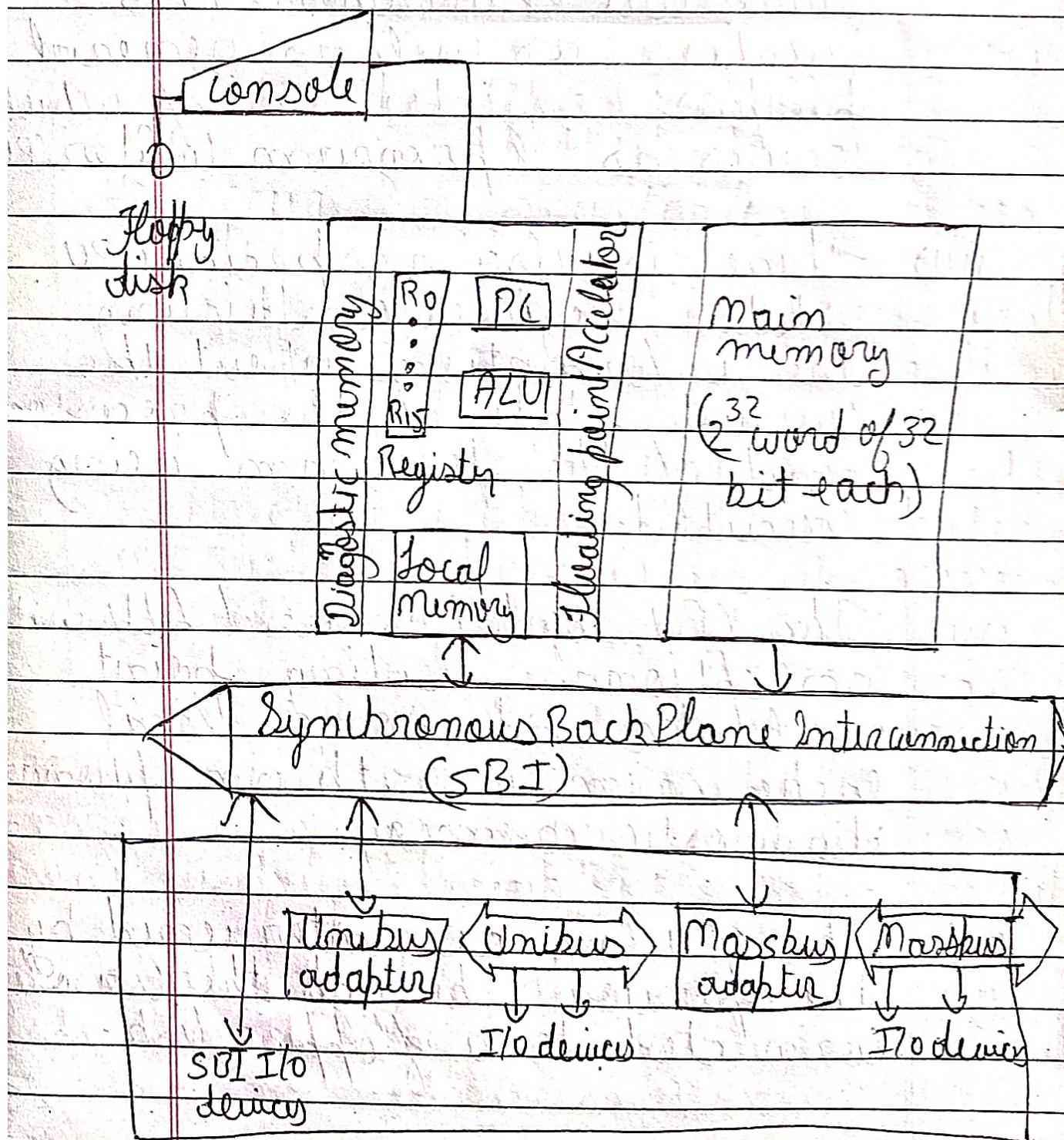
* Parallelism in uniprocessor system:-

- (i) Most general purpose uniprocessor system have the same basic structure.
- (ii) The development of parallelism in uniprocessor will then be introduced categorically.
- (iii) It will provide only concise specifications of the arithmetic facilities of the popular commercial computers.
- (iv) Parallel processing mechanism and methods to balance subsystem band-width will then be describe for very typical uniprocessor system.

* Basic uniprocessor architecture

- (i) A typical uniprocessor computer consists of three major components are the main memory, the central processing unit & and I/O subsystems.

(ii) The architecture of two commercially available uniprocessor computers are given to show the possible interconnection of structures among the three subsystems.



- (i) The architecture components of the super mini-computer VAX-11/780 manufactured by digital equipment corporation.
- (ii) The CPU contains the master controller of the VAX-11/780 system. There are sixteen 32 bits registers as well as general purpose register, one of which serves as program counter (PC).
- (iii) There is also a special CPU status register containing the information about the current state of the processor and of the program being executed.
- (iv) The CPU contains and ALU with an optional floating point accelerator and some local cache memory with an optional diagnostic memory.
- (v) The CPU can be interfaced by the operator through the console connected to a floppy disk.

- (vi) The CPU main memory (2^{32} words of 32 bit each) and the I/O sub systems are all connected to a common bus, the synchronous backplane interconnect (SBI).
- (vii) Through this bus all I/O devices communicate with each other with the CPU or with the memory.
- (viii) I/O devices can be connected directly to the SBI through the Unibus and its controller or through a mass bus and its controller.

~~QUESTION~~ Basic UniProcessor Architecture IBM-370.

A typical uniprocessor architecture consists of three major components:-

- (i) Main memory.
- (ii) Central Processing unit.
- (iii) Input output subsystem.

Main Memory

Logical Storage Unit

[LSU0] [LSU1] [LSU2] [LSU3]

storage controller

Central Processing Unit
(CPU)

I/O channels

I/O system

System architecture of IBM
system 370 / model 168.

- (i) The main frame computer IBM system 370 model 168 uniprocessor is shown in above figure.
- (ii) The CPU contains structure decoding and execution Units as well as cache.

- (iii) Main memory is divided into four units referred to as local storage unit.
- (iv) The storage controller provides multi mode connection b/w the CPU & the four LSUs. The peripheral devices are connected to the system via high speed input output channels which operates asynchronously with the CPU.

A. Parallel Processing mechanism

- (i) Multiplicity of functional units
The early computers had only one ALU in this CPU. Furthermore the ALU could only perform one function at a time. Many of the functions of the ALU can be distributed to multiple and specialized function units which can operate in parallel.
- (ii) Parallelism & Pipelining within the CPU: →

(i) High speed Multiplier recording and convergence division are techniques for exploring Parallelism and the sharing of hardware resources for the function of multiple 8 division.

(b) Various phases of instruction execution are now pipeline including instruction fetch, decode, operand fetch, arithmetic logic execution and store result.

(iii) Overlapped CPU & I/O operation :-
I/O operation can be performed simultaneously with the CPU computation by using separate I/O controllers, channels or I/O processors.

The DMA channels can be used to provide direct information transfer b/w I/O devices and main memory.

(iv) Use of hierarchical memory system :-

Hierarchical memory system can be used to eliminate the speed gap b/w CPU & memory.

Cache memory can be used to serve as a buffer b/w the CPU & main memory.

Virtual memory space can be established with the use of disks & tape unit at the outer levels.

(v) Balancing of sub system bandwidth:

The CPU is the fastest unit in a computer with the processor cycle time of tens of nano second.

The main memory has a cycle time t_m of 100ns a nanosecond.

The I/O devices are the slowest within average access time t_d of a few milliseconds.

$$t_d > t_m > t_p$$

(vi) Multiprogramming & time sharing:

We can mix the execution of various types of programs

in the computer to balance bandwidth memory the various functional units.

Interleaving of CPU & I/O operations among several programs is called as multiprogramming.

* Structure of parallel computing

Parallel Computers are those systems that emphasize parallel processing. Parallel computers can be divided into three architecture configuration.

(i) Pipeline Computers: -> A pipeline computer performs overlaid computations to exploit temporal parallelism. Normally the processor of executing an instruction of digital computer involves four major steps:-

- (a) Instruction fetch (IF) from the main memory.
- (b) Instruction decoding (ID).
- (c) Operand Fetch (OF).
- (d) Execution (EX).

In a non-pipeline computer,

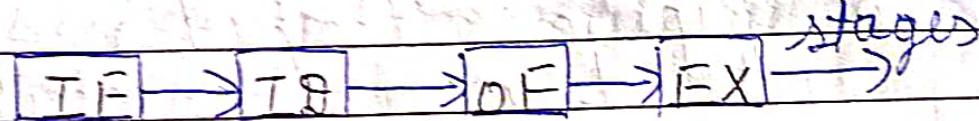
These four steps must be completed before the next instruction can be issued in a pipeline computer, successive instruction are executed in an overlapped fashion. Four pipeline stages, IF, ID, OF, EX are arranged into a linear cascade. Some main issues in designing a pipeline computer include job sequencing, collision prevention, contention control, branch handling.

* Second array computers

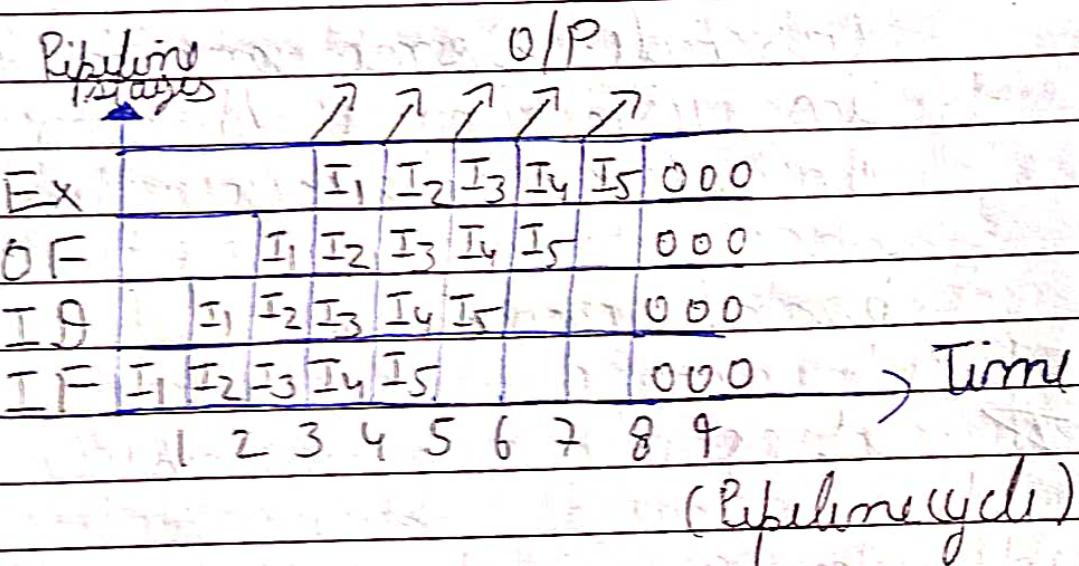
An array processor is a synchronous parallel computer with multiple arithmetic logic unit called processing elements (PE) that can operate in a parallel in a lock step fashion.

The PE's are synchronized to perform the same function at a same time. Each PE consist of an ALU with registers and a local memory. The PE

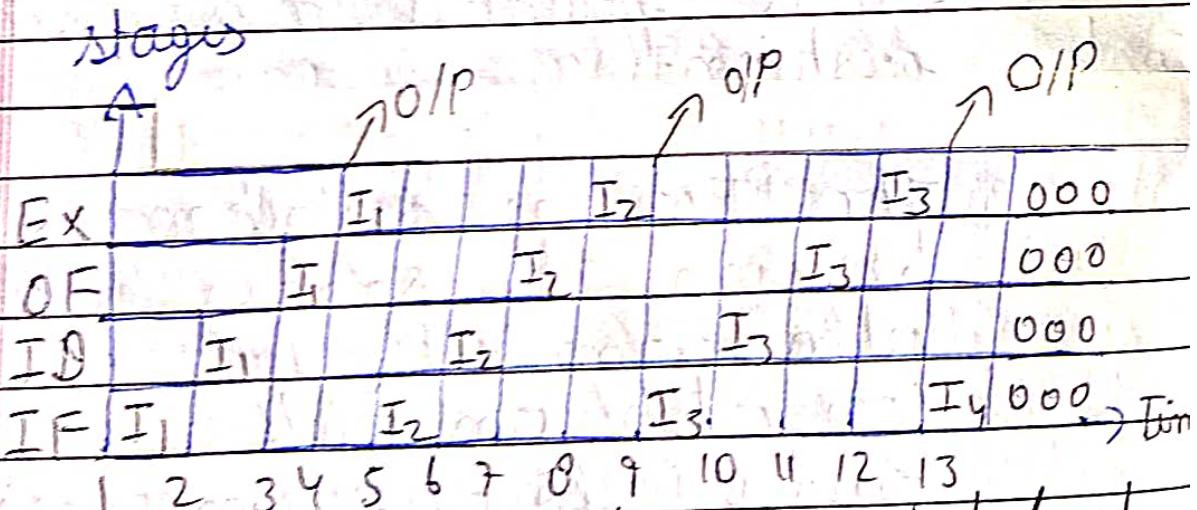
are interconnected by a data routing network.



Pipelined Processing



(Space-time diagram for pipelined processor)



space-time diagram for a non-pipelined processor.

* Multi Processor System

In this, the system contains two or more processor of approximately comparable compatibility. All processors share access to common set of memory modules I/O channel and peripheral devices.

Multiprocessor hardware system orgⁿ is determine primarily by the inter-connection structure to be used b/w the memory and processors. Three different inter-connection have been practiced.

- (a) Time-share common bus
- (b) Cross bar switch network.
- (c) Multipoint memories.

* Define & differentiate b/w Control flow & data flow?

Control Flow	Data Flow
(i) Control Flow is where we define operations & order of execution of those operations. For ex- we put two	(i) Data Flow is where we define data stream & where data comes from data source, how data should be transformed & where data should be

operations execute ~~transformed~~ loaded t-sql commands on a database & send email then we define order of them with precedence constraint & if t-sql statement should be executed & then if it's successful mail will be sent.

- (ii) Dominant questions is new focus of central moves through the programmatic computation.
- (iii) Data may accompany dominant.
- (iv) Reasoning is about order of computation.
- (v) A program is a series of addressable instructions, each availability of operand of which either.
- (vi) As data moves, control is activated.
- (vii) Reasoning is about data availability, transformation etc.
- (viii) The execution is driven only by the availability of operand.

→ Specifies an operation along with memory location of the operands.

→ Specifies unconditional transfer of control to some other instruction.

(vi) Essentially, the next instruction to be executed depends on what happened during the execution of the current instruction.

(vi) No (Program control) & global updatable store.

A Data Driven Computing and languages:-

(i) Data driven is about building tools, ability, and most crucial, a culture that acts on data.

(ii) It means that progress in an activity or personal expansion.

- (iii) A data driven strategy details understanding the characteristics of the digital environment and adopting a rigorous method of interpreting data to drive a suitable web strategy.
- (iv) It is an on-going iterative approach that enables two output the digital resources constituents to reflect changes in the online environment rights of view social network, constitutive action, new user habit etc.

P Differentiate b/w Parallelism & Pipeline

Parallelism

Pipeline

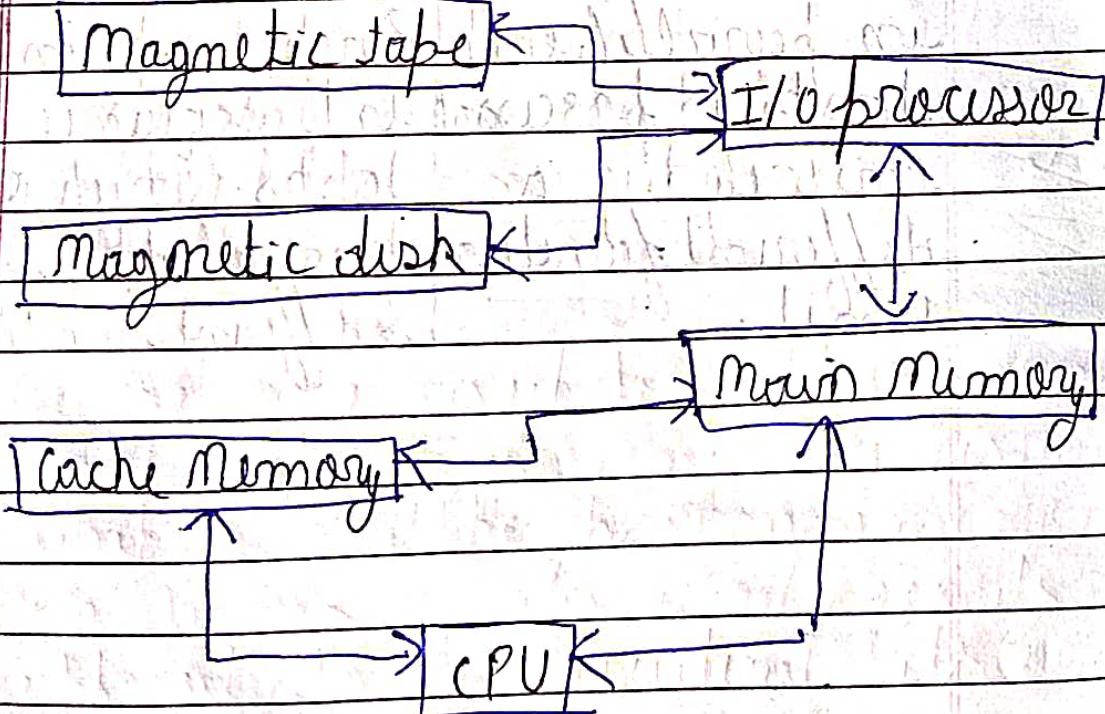
- | | |
|--|--|
| i) Parallel processing is a technique for duplicating function units to operate different tasks simultaneously. | Pipeline is a set of elements connected in a series, where the output of one element is the input of the next one. |
| ii) In Parallelism mechanism, duplicated function units working in parallel, each task is processed entirely by a different function unit. | In pipelining mechanism, different function units working in parallel, each task is split into sequence of sub tasks which are handled by specialized units. |

Memory hierarchy :-

The memory unit is an essential component in any digital computer since it is needed for storing programs and data. There is a trade off among the three key characteristics of memory i.e. cost, capacity, and access time.

The relation b/w these characteristics are:-

- (i) Small access time, greater cost per bit
- (ii) Greater capacity, smaller cost
- (iii) Greater capacity, greater access time



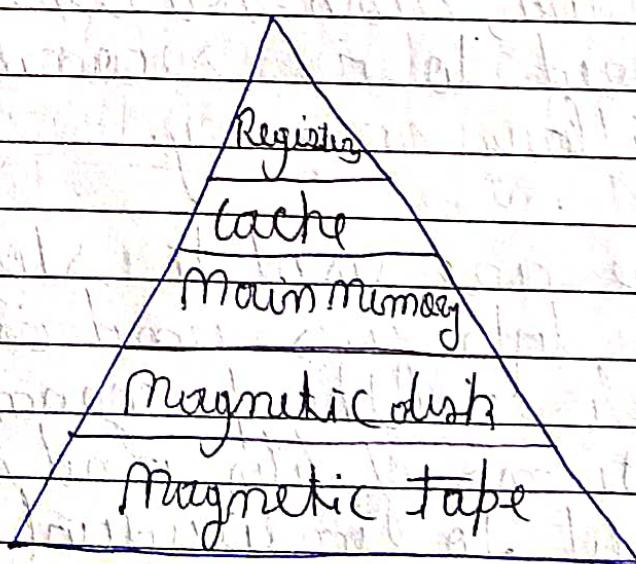
Memory hierarchy in computer system

The designer would like to use memory technologies that provide for large capacity memory, both because the capacity is needed, and because the cost per bit is low.

As one goes down the hierarchy the following occur :-

- Decrease in cost per bit.
- Increased capacity
- Increased access time
- Decrease in the frequency of access of the memory by the CPU.

Traditional Memory Hierarchy :-



- (i) The fastest, smallest and most expensive type of memory consists of the registers internal to the processor.
- (ii) The main memory is usually extended with a higher speed, smaller cache.
- (iii) The cache is not usually visible to the programmer or indeed to the processor.
- (iv) Data are stored more permanently on external mass storage devices which the best most common are magnetic disk and tape.
- (v) External, non-volatile memory is referred to as secondary or auxiliary memory.
- (vi) These are used to store programs and data files and are usually visible to the programmer only in terms of files and records as opposed to individual bits or words.

ii) Disk are also used to provide an extension to main memory known as virtual storage or virtual memory.

Importance of memory hierarchy

The memory hierarchy in most computers are :-

i) Processor registers :> The fastest possible access (usually one (PU cycle) only hundreds of bits in size).

ii) Level 1 (L1) Cache :> Often accessed in just a few cycles, usually tens of KB.

iii) Level 2 (L2) Cache :> Higher latency than L1 by usually 512 kb or more.

iv) Level 3 (L3) Cache :> Higher latency than L2, usually has 2048 kb and more.

v) Main Memory :> May takes 100th of cycles what can be multiple gigabytes. Access time may not be uniform.

(vi) Disk Storage: \rightarrow Millions of cycles
latency of most cache but with
multiple of terabytes

(vii) Tertiary Storage: \rightarrow Several seconds
latency but can be huge.

* Addressing Schema for Main Memory :-

- In a parallel processing environment main memory is a prime system resources which is normally shared by all processor or independent units of a pipeline Processor.
- Care must be taken in the orgⁿ of the memory system to avoid severe performance degradation because of memory interference caused by two or more processors simultaneously attempting to access the same module of the memory system. The scheme called interleaving, resolve some of the interference by allowing concurrent access to more than one module.

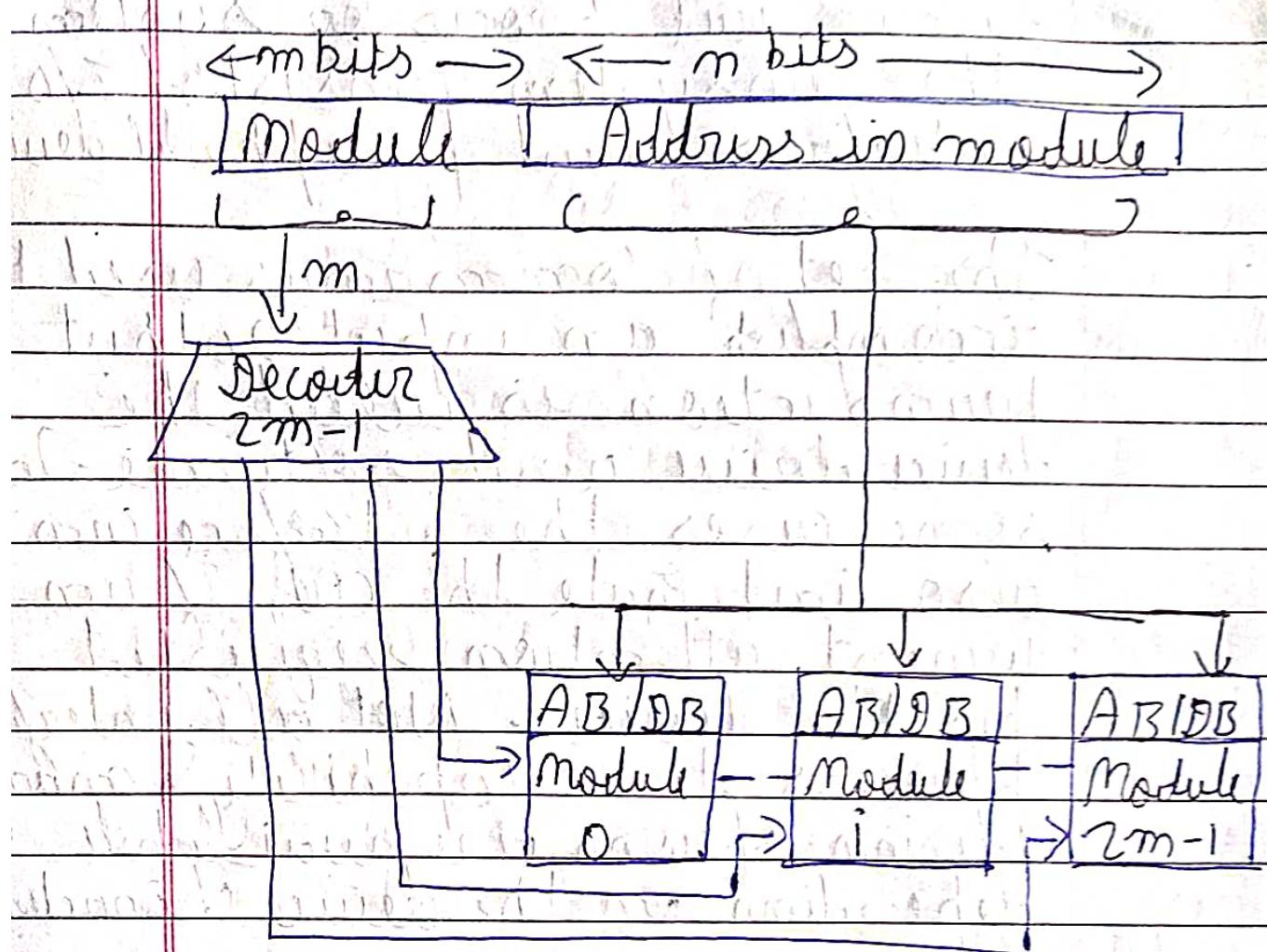
The interleaving of addresses among the modules is called m way interleaving. Assume that there are total of capacity equal to 2^n words in main memory then the physical

address for a words in the memory consist of 'n' bits $a_{m-1}, a_{m-2}, \dots, a_1, a_0$.

One method, high order interleaving distribute the addresses $m = 2^m$ modules shows that each modulus (i) for $0 \leq i \leq m-1$ contains consecutive addresses $[i2^{m-m}]$ to $[i+1]^{2^{m-m}-1}$ inclusive.

The high order m bits are used to select the module while the remaining $m-m$ bits select the address within the module.

The second method, the low order interleaving, distribute the addresses so that consecutive addresses are located within consecutive modules. The low order M bits of the address select the module while the remaining $m-M$ bits select the address within module.

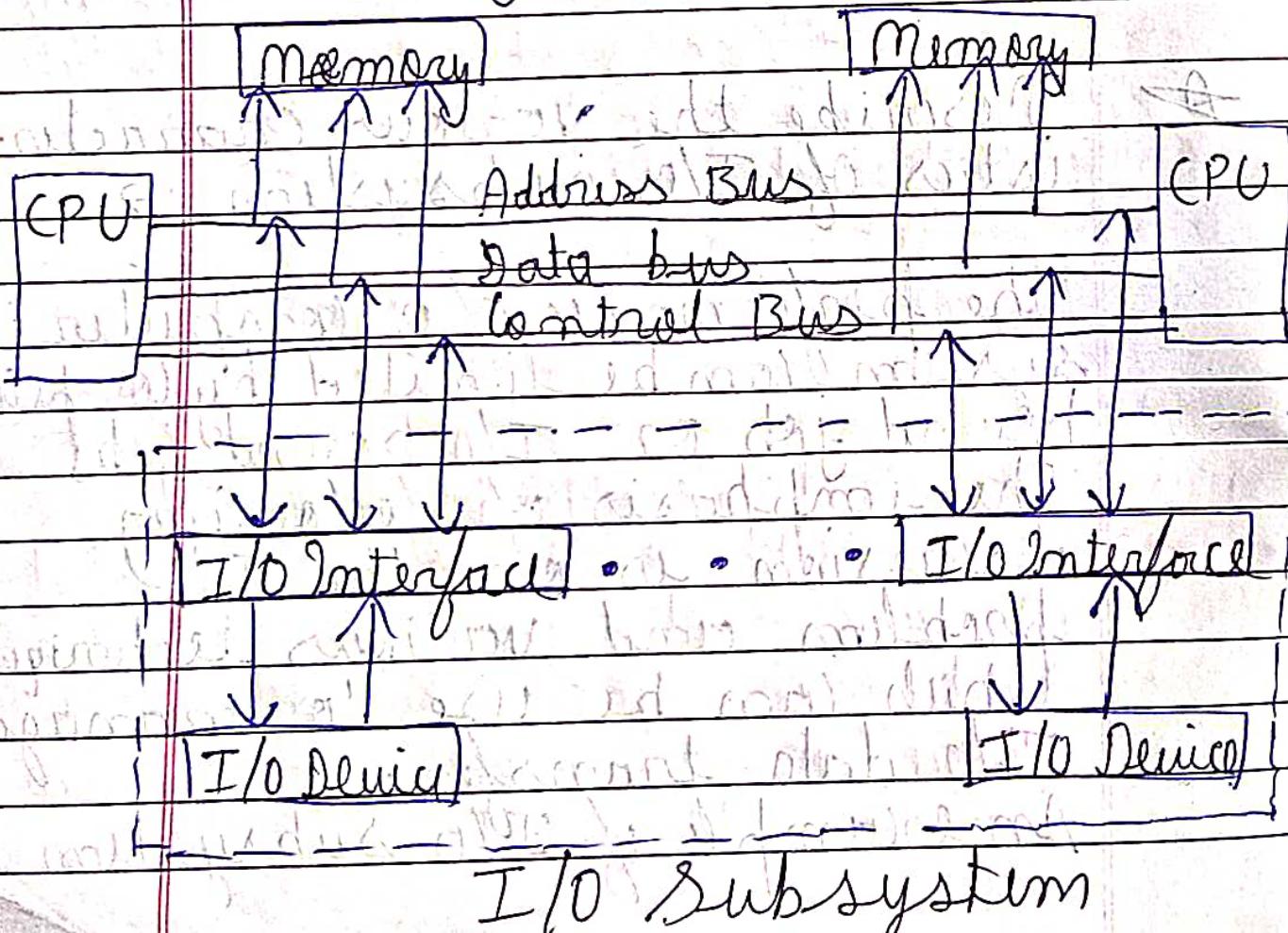


~~A~~ Describe the various characteristics of I/O subsystem :-

- The performance of a computer system can be limited by computation bound job or I/O bound job.
- The emphasis of following discussion is on the I/O problem and various techniques which can be used to manage I/O data transfer.
- An example of I/O subsystem

for a dual processor system
the subsystem consist of I/O
interfaces and peripheral devices

The set of commands used to
accomplish an input output
transaction is called the
device driver and software. In
some cases the interface can
also integrate the CPU, if an
urgent attention requested
by the device. Not all interfaces
process these capability. Many
design options are available
depending on the device characteristic.



E Devise types of I/O subsystem

Virtual Memory

Virtual memory is a memory mgmt capability of a OS that uses hardware & software to allow a computer to compensate for physical memory shortage by temporarily transferring data from RAM to disk storage.

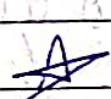
In earlier computers when the entire program wouldn't fit into memory space at one time, a technique called overlay was used. Pages of the program were brought into memory when needed overlay those that were no longer needed.

Virtual memory gives programmer the illusion that there is a very large memory at their disposal whereas the actual physical memory available may be small.

In a multiple processor system with virtual memory, this mechanism must be provide for each processor. Assume that memory space V_j generated by the j^{th} program running on a processor

consist of a set of unique identifiers.

$$v_j = \{0, 1, \dots, n-1\}$$



Flynn's Classification

The 4 classifications defined by Flynn are based upon the no. of concurrent instructions (or control) and data streams available in the architecture.

- (i) Single instruction, single data stream \rightarrow (SISD) \rightarrow A sequential computer which exploits no parallelism in either the instruction or data streams.
Single control unit (CU) fetches single instruction stream from memory.

- (ii) Single instruction, multiple data stream (SIMD) \rightarrow A sequential computer which exploits multiple data stream against a single instruction stream to perform operations which may be naturally parallelized - An array processor or CPU.

iii) Multiple Instruction, single data stream (MISD) \rightarrow Multiple Instructions operates on a single data streams. Uncommon architecture which is generally used for fault tolerance. Heterogeneous systems operate on a same data stream & must agree on the result.
 \rightarrow Space shuttle flight control computer.

iv) Multiple Instruction, Multiple data stream (MIMD) \rightarrow Multiple autonomous processor simultaneously executing different data. Distributed system are generally recognized to be MIMD. Architecture either exploiting or single shared memory space or distributed memory space.

\rightarrow A multicore superscalar processor is an MIMD processor.

Concept of Pipelining

- (i) Pipelining is a method to realize overlapped parallelism in the proposed solution of a problem on a digital computer in an economical way.
- (ii) Pipelining is widely used in modern processors.
- (iii) Pipelining improves system performance in terms of throughput.

Example of Pipelining

- (i) Concept of assembly lines in an automated production plants where items are assembled from separate parts (stage).
- (ii) Out of one stage becomes to input to another stage.
- (iii) Assembly line is a pipelining and the separate parts of the assembly line are different stages through which operands of an operation are passed.

Steps to introduce pipelining

- i) Sub divide the input process into a sequence of sub tasks. These sub task will make stages of pipeline which are also known as segments.
- ii) Each stage s^i of the pipeline acc. to the sub task will perform some operation on a distinct set of operands.
- iii) When stage s^i has completed its operation, results are passed to the next stage s^{i+1} for the next operation.
- iv) The stage s^i receives a new set of inputs from previous stage s^{i-1} .

Principles of Linear Pipelining

- i) In pipelining we divide a task into set of subtasks.
- ii) The precedence relation of a set of sub tasks ($T_1, T_2 \dots T_k$) for a

Classification of Pipelining processing

Acc. to the labels of processing, handles proposed thru pipeline classification schemes \rightarrow

(i) Arithmetic pipeline

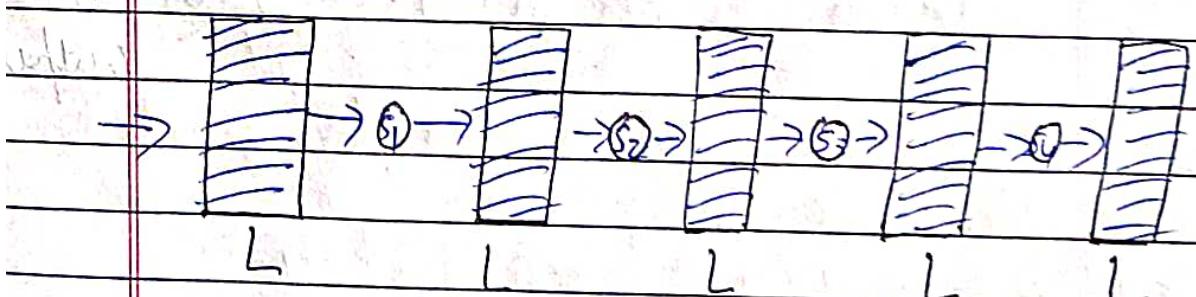
(ii) Processor

(iii) Instruction

(i) Arithmetic pipeline: \rightarrow An arithmetic pipeline generally breaks an arithmetic operation into multiple arithmetic steps that can be executed one by one - in segments in ALU: \rightarrow

(b) In arithmetic pipeline the value of a computer is segmented for pipeline operations in various data formats.

\rightarrow Jatch

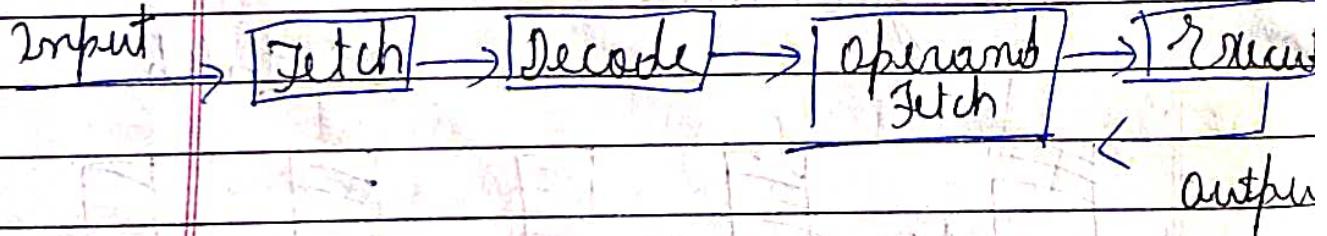


→ Four stage pipeline in
start one hundred (100) SIAR-100
eight stage pipeline in
TI-ASC

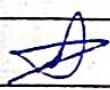
Instruction Pipeline

In Instruction Pipeline, the execution of a stream of instruction can be pipeline by overlapping the execution of current instruction with the fetch, decode and operand fetch of subsequent instruction.

This technique is also known as instruction look ahead
Example Almost all high performance computers now a days are equipped with instruction pipeline processor



Step	1	2	3	4	5	6	7	8	9
1	F _I	DA	F _O	Ex					
2		F _I	DA	F _O	Ex				
3			F _I	DA	F _O	Ex			
4				F _I	DA	F _O	Ex		
5					F _I	DA	F _O	Ex	
6						F _I	DA	F _O	Ex



Processor Pipelining

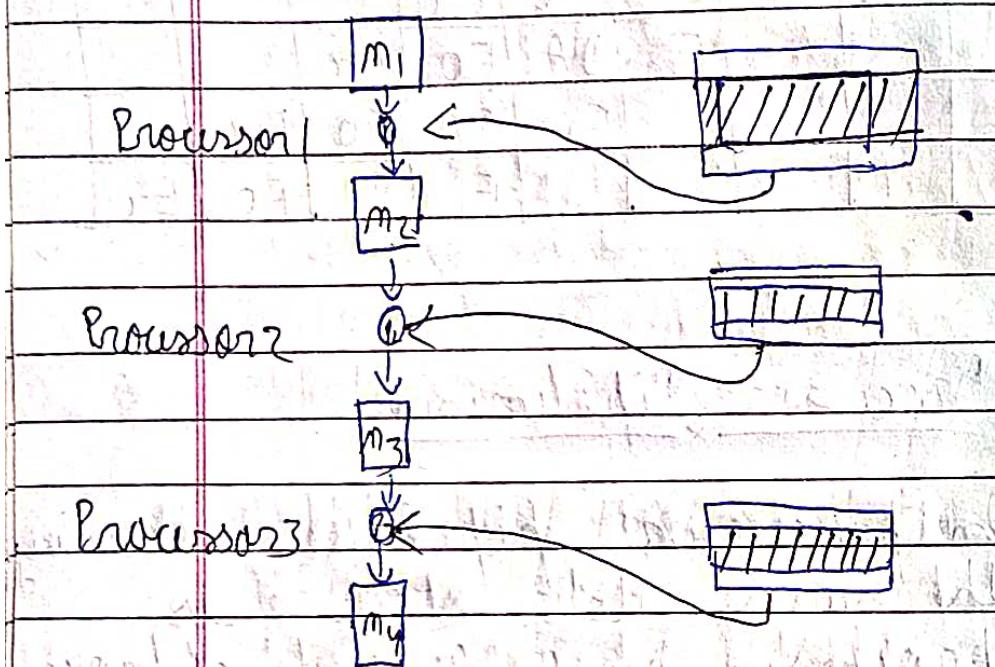
In processor pipeline processing the same data stream is processed by a cascade of processor. Each processor performs a specific tasks.

The data stream passes the first processor with results stored in a memory block which is also accessible by second processor.

The second processor processes this result and passes it to third and so on.

This pipeline processor is not

much popular. There is no practical for example for processor pipeline.



Acc. to pipeline configuration and control strategies Billes and Roma Murthy has proposed following pipeline classification

(i) Unifunction vs multifunction pipeline

(ii) Static vs Dynamic pipeline

~~vs~~ Unifunction vs multifunction pipeline →

(a) Unifunction pipelines →

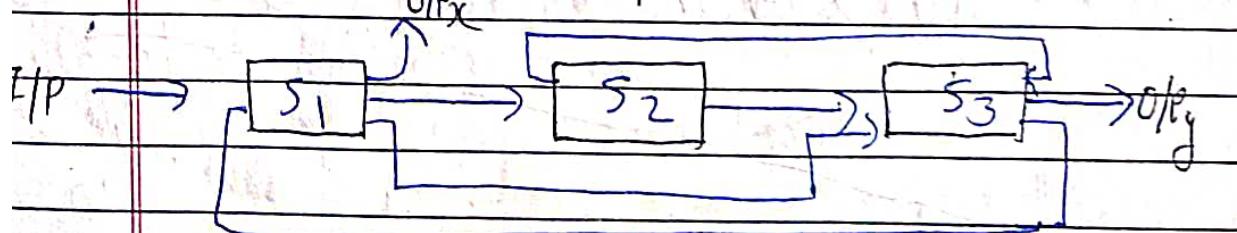
A pipeline with fixed and dedicated functions is called a unifunction pipeline.

	1	2	3	4
S_1	X			
S_2		X		
S_3			X	
S_4				X

100. Floating point Adder

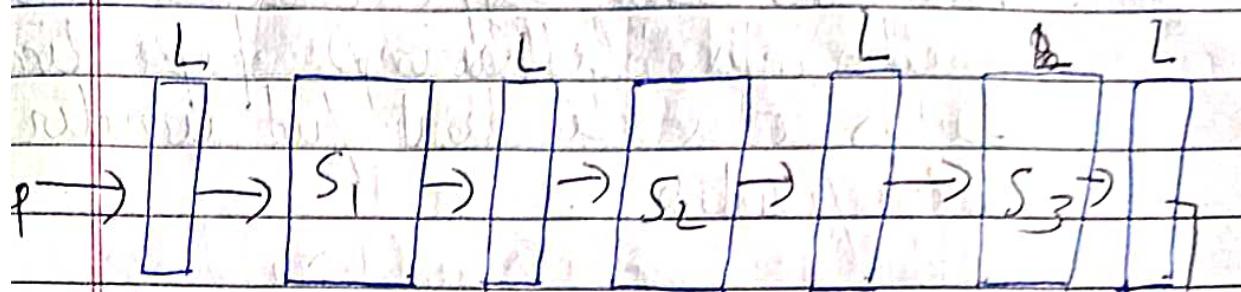
(3) Multifunction pipelines

A pipeline that performs different functions either at different times or at the same time by interconnecting different subset of stages in the pipeline is called a multifunction pipeline. IX-TI-ASC has multifunction pipeline processing.

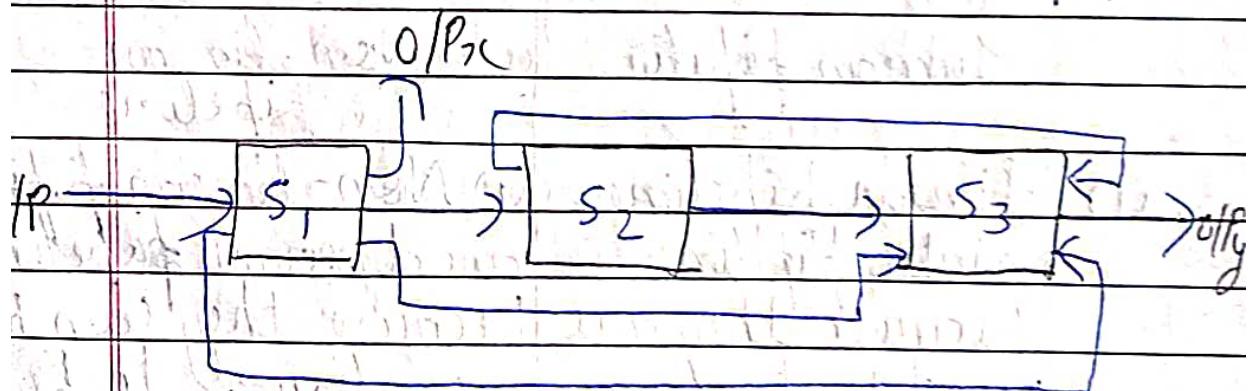


* Static vs Dynamic pipeline

- (i) Static pipeline \rightarrow Static pipeline
- (ii) assume only one functional configuration at a time
- (iii) It can be either functional or multi-functional
- (iv) static pipelines are preferred when instruction of some type are to be executed continuously.
- (v) A uniformization pipeline must be static



- (b) Dynamic pipeline: → Dynamic pipeline permits several functional configuration to exist simultaneously.
- (c) A dynamic pipeline must be multifunctional.
- (iii) The dynamic configuration requires more elaborate control and sequencing mechanism than static pipeline.



Acc. to the type of instruction and data following pipeline types are identified under this classification.

(i) Scholar pipeline: \Rightarrow This type of pipeline process and a set of operand of repeated scholar instruction process scholar operand under the control do loop.
 \rightarrow IBM-360

(ii) Vector pipeline: \Rightarrow This type of pipeline process vector instruction over vector operands.
 \rightarrow STAR-100, CRAY-1

~~Distinguish b/w linear pipeline & non-linear pipelining~~

Linear pipeline

Non-linear pipeline

(i) Linear pipeline and (ii) Non-linear pipeline
 static pipeline are dynamic pipeline because they can be used to perform fixed functions. variable functions at different times.

- | | |
|---|--|
| (i) It allows only streamline connections. | (iii) It allows feed-forward and feedback connections in addition to the streamline connections. |
| i) It is relatively easy to partition owing to reality, a given function difficult because the into a sequence pipeline stages are of linearly ordered interconnected with sub functions. | (iii) Function partitioning is non-trivial in the sense that there is that data flows in linear form data flows. |
| iv) Reservation table is trivial in the sense that there is no linear streamline. | iv) Reservation table is non-trivial in the sense that there is no linear streamline. |
| v) Static pipeline is specified by single reservation table. | (v) Dynamic pipeline is specified by more than one reservation table. |
| vi) All initiations to a static pipeline use the same reservation table. | (vi) A dynamic pipeline may allow different initiation to allow a mix of reservation table. |

* Different cache memory addressing techniques : →

Cache can be addressed using either a physical address or a virtual address.

* Physical address cache : →

- (a) When a cache is accessed with a physical memory address it is called a physical address cache.
- (b) In this cache is indexed and tagged with the physical address.
- (c) A cache hit occurs when the address data / instructions is found in the cache; otherwise a cache miss occurs. After a miss the cache is loaded with the data from the memory. Since a whole cache block is loaded at one time, unloaded data may also be loaded.
- (d) The major advantage of physical cache is include no need to

perform cache flushing no aliasing problems and thus fewer cache bugs in the OS kernel.

- e) The disadvantage is slow down in accessing the cache until the MMU finishes translating the address.

Example when physical address cache is not used in a Unix environment no flushing of data cache is needed if bus watching is provided to monitor the system bus for DMA request from input out device or from other CPU.

A) Virtual address cache :-

- (a) When a cache is index or tagged with a virtual address it is called virtual address cache. In this both cache MMU translation or validation are done in parallel.
- (b) The physical address generated by the MMU can be stored

in tags for later write back but isn't used during the cache lookup operations.

- (c) The virtual address cache is motivated with its enhanced efficiency to assist the cache faster, overlapping with the mmu translation.
- (d) The major problem associated with the virtual address cache is aliasing, when different logically address data have the same index/tag in the cache. Multiple process will generally use the same range of virtual addresses.

Ans → SUN 3/200 series has used a virtual address, write back cache with the capability of being not cachable.

Date: ___ / ___
Page: ___

Architecture of STAR-100

The STAR-100 is a vector oriented processor with 10 non-homogeneous arithmetic pipelines. Control Data Corporation started a design of STAR in 1965 and delivered it two user in 1973.

It is structured around a 4 million transistor high bandwidth core memory with stand alone operation.

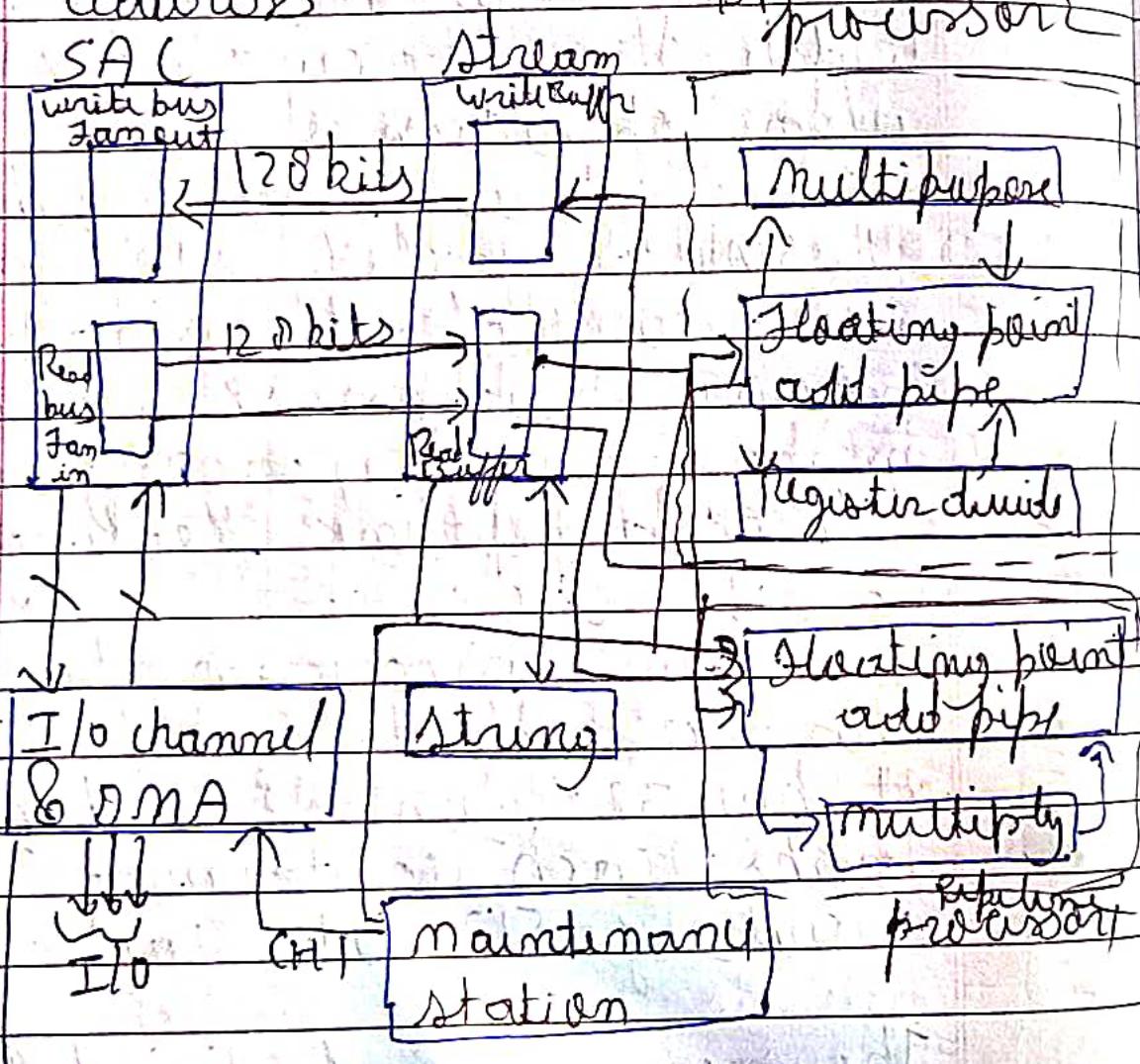
Some features designed in the STAR-100 include stream processing, virtual addressing hardware, microinstruction semiconductor memory register, among registers file and pipeline floating point arithmetic.

The core memory in the STAR has a cycle time of 1.28 ms. It has 32 interleaved memory banks each containing 2048 words of 512 bits each.

The memory cycle divided into 32 minor cycles with the rate of 40 ms each.

The pipeline unit are specially designed for sequential parallel operations on single bit 8 bits/bytes and 32 to 64 bits floating point operands having

Virtual addressing employs a high speed mapping technique to convert a logical address to an absolute memory address.



Direct

access areas System architecture of STAR-100

Central channel

Working of STAR-100 architecture

- i) The storage access control (SAC) unit controls the transmission of all data to and from the memory. It is responsible for memory sharing among the various busses shared by the stream & I/O units. Its principle function is to perform virtual memory address comparison and translation.
- ii) The Stream unit provides basic control for the system. All memory references and many control signals originate from this unit. It has the facilities for instruction buffering and decoding.
- iii) The read Buffer and write Buffer ^{are used to} synchronous the four victim buses to maintain a smooth data transfer.
- iv) Other functional unit in the

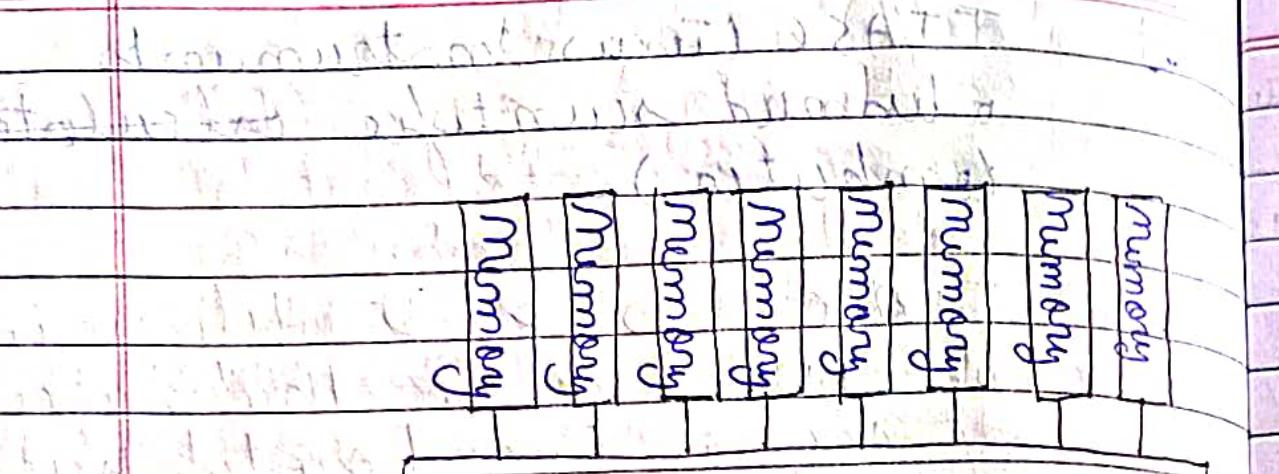
Stream unit include the register file and the micro code memory. The register file supplies necessary addressing from for all source operands and results. It also has a capability of performing simple logical and arithmetic operations.

The semiconductor microcode memory is used as part of the stream control.

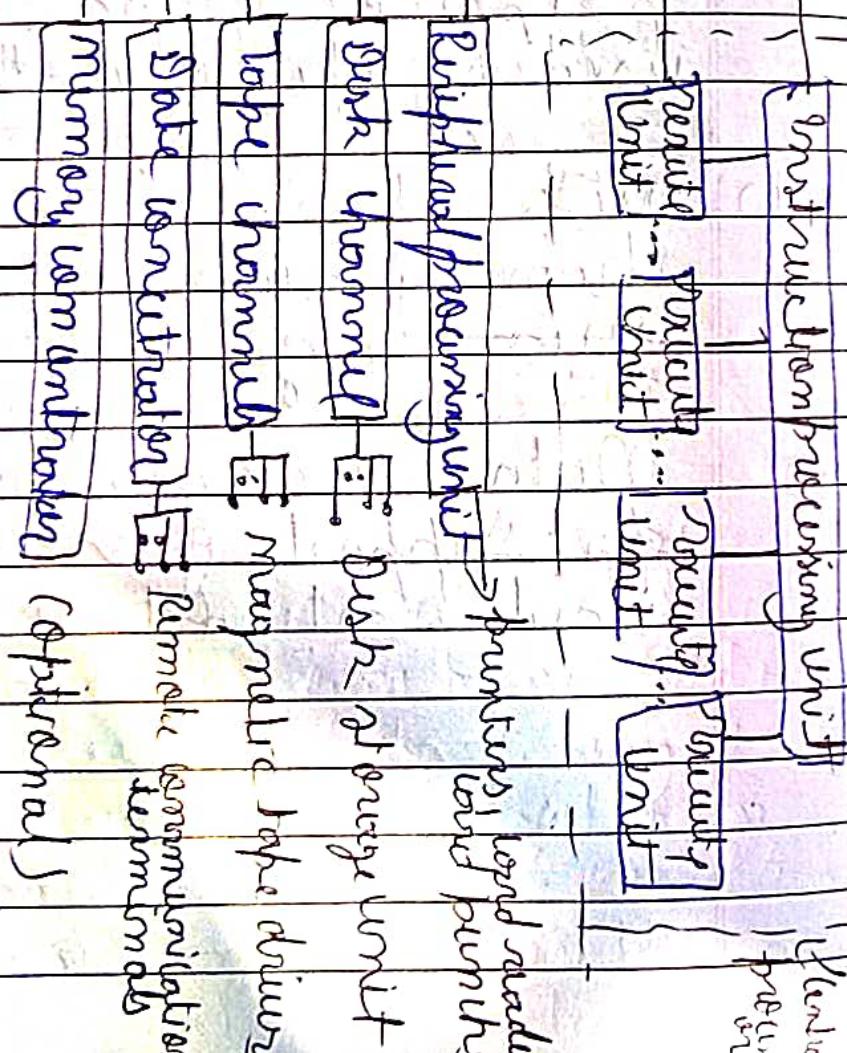
- (v) The stream unit processes string of decimals or binary digits and performs bit logical and character string operations.

TI/ASC (Texas Instruments advanced scientific calculator computer).

- i) TI ASC was first delivered in 1972. The central processor of ASC is incorporated with a high degree of pipelining in both instruction and algorithmic levels.
- ii) The control processor is used for its high speed to process a large array of data. The peripheral processing unit is used by the operating system.
- iii) Disk channel and tape channel supports a large number of storage units.
- iv) Data concentrator are included for support of remote batch and interactive terminals.



Memory
Content
Addressable
Unit



TASC

(optional)

Vector processing

A vector is a set of similar data items, all of the same type stored in memory. Vector processing occurs when arithmetic or logical operations are applied to vector. It is distinguish from scalar processing which operands on one or more pair of data.

Characteristics of vector processing

A vector operand contains an ordered set of ~~all~~ m elements where m is called the length of the vector. Each element in vector is a scalar quantity, which may have floating point number, an integer a logical value or a character.

Vector instruction can be classified into four primitive types where v and s denote a vector operand and a scalar operand respectively. f_1, f_2 & f_3 are unary operations and f_4 & f_5 are binary operations.

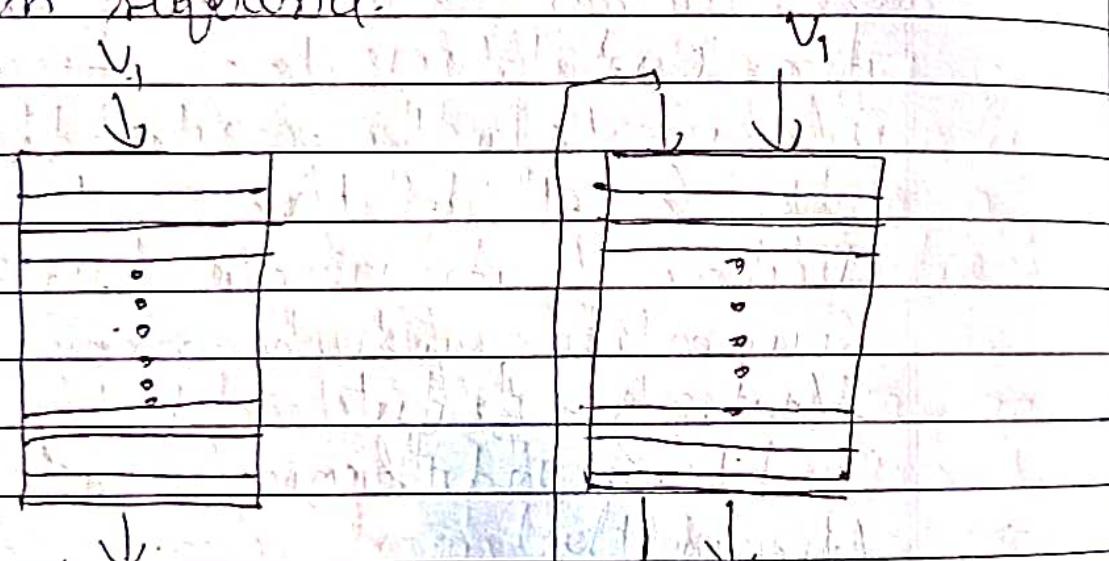
For example VSQR (Vector Square Root) is an f_1 operation, VSUM (Vector summation) is a f_2 operation, SVP (Scalar vector product) is an f_4 operation and VADD (Vector addition) is an f_3 operation.

The dot product of two vectors

$$V_1 \cdot V_2 = \sum_{i=1}^n V_{1i} \cdot V_{2i}$$

is generated

by applying f_3 (vector multiplication) & then f_2 (vector sum) operation in sequence.



A boolean vector can be generated as a result of comparing two vectors and can be used as a masking vector for dis-enabling or disabling component operations in vector instruction.

A compare instruction will be shorter a vector under the control of a masking vector. A merge instruction combines two vectors under a control of a masking vector.

2. Distinguish blur scalar & vector processing

- i) Vector instruction need to perform the same operation on different data sets repeatedly. This is not true for scalar processing vector. On a single pair of strands.
- ii) Advantage of vector processing over scalar processing is the elimination of the overhead by the loop control mechanism.
- iii) A vector processor should perform better with longer vector because of the startup delay in a pipeline.

A Super Computer

- (i) A super computer is a computer at a frontline of contemporary processing capacity particularly speed of calculation.
- (ii) Super computers are used for very complex jobs such as nuclear research or collecting and calculating weather patterns.
- (iii) A super computer is typically used for scientific and engineering applications that must handle very large databases or do a great amount of computation.
- (iv) Mostly super computers are really multiple computers that performs parallel processing.
- (v) In general there are two parallel processing approaches
 - (a) Symmetric multiprocessing
 - (b) Massively parallel

- I) Super computer were introduced in the 1960's design initially and for decades, primarily by Seymour Cray at control data Corporation (CDC). (RAY research and subsequent companies using his name as monogram.
- III) Super computer plays an important role in the field of Computational science and are used for a wide range of computationally intensive tasks in various fields including Quantum mechanics, weather forecasting, climate research, oil and gas exploration and so on