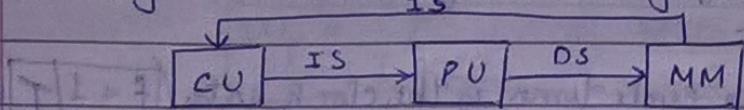


① Flynn's Classification -

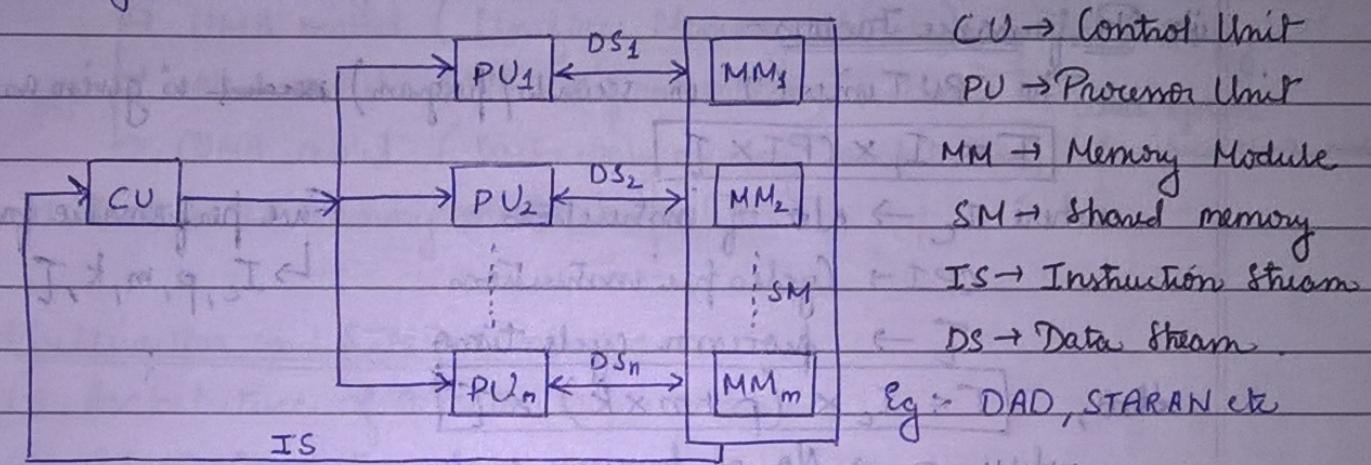
It is based on the multiplicity of instruction stream and data stream. Flynn's four machine organization are -

- 1) Single instruction stream - single data stream (SISD) -

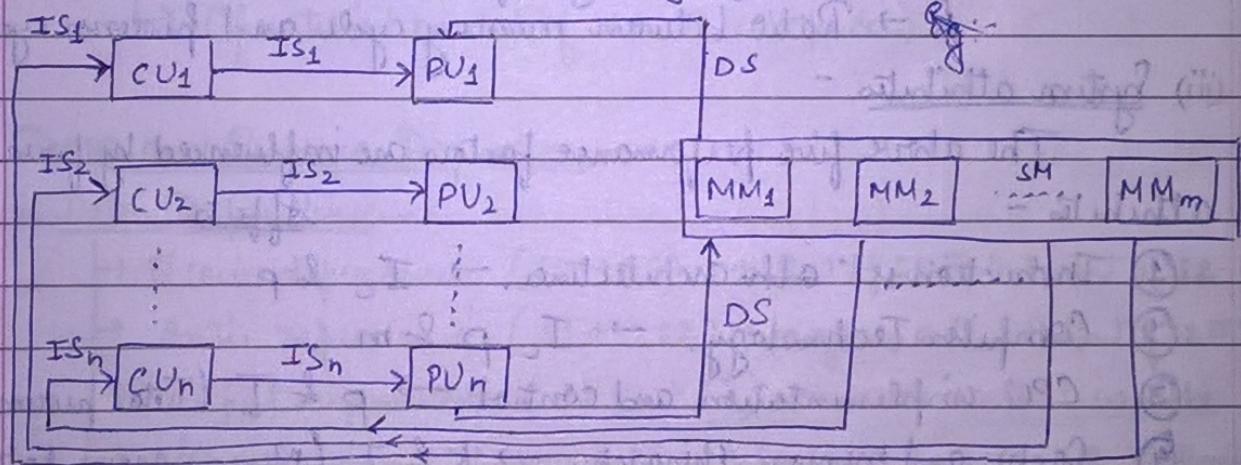


Eg - IBM 701, CDC 6600 etc

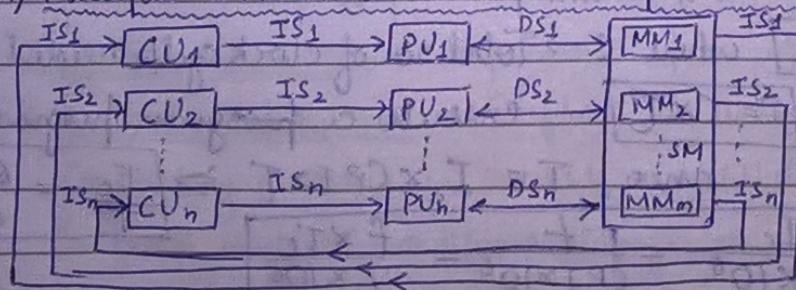
- 2) Single instruction stream - multiple data stream (SIMD) -



- 3) Multiple instruction stream - single data stream (MISD) -



- 4) Multiple instruction stream - multiple data stream (MIMD) -



Eg - Cray - 2, Univac 1100/80

② System Attributes to Performance -

~~It is impossible to achieve perfect match between hardware and software by merely improving only a few factors without touching other factors. Some of the factors are as follows -~~

(i) Clock Rate and CPI -

The inverse of the cycle-time is the clock rate ($f = 1/T$ in MHz)

CPI (cycles per instruction) is for measuring the time needed to execute each instruction.

(ii) Performance Factors -

The CPU Time (T in seconds/program) ~~needed~~ is given as -

$$T = I_c \times CPI \times T_c$$

where $I_c \rightarrow$ No. of instructions

$CPI \rightarrow$ Cycles per instruction

$T_c \rightarrow$ processor cycle time.

Also, $T = I_c \times (p + m \times k) \times T_c$

where $p \rightarrow$ No. of processor cycles.

$m \rightarrow$ No. of memory reference needed

$k \rightarrow$ Ratio between memory cycle and processor cycle.

Five performance factors

$$\hookrightarrow I_c, p, m, k, T_c$$

(iii) System attributes -

The above five performance factors are influenced by four system attributes -

Affects

① Instruction set & architecture. $\rightarrow I_c \& p$

② Compiler Technology. $\rightarrow I_c, p \& m$

③ CPU implementation and control. $\rightarrow p \& T$ (Total processor Time)

④ Cache and memory Hierarchy. $\rightarrow k \& T$ (Memory access latency)

(iv) MIPS Rate (Million Instructions per second) -

Let $[C = I_c \times CPI]$ where $C \rightarrow$ Total no. of clock cycles needed to execute a program

~~$T = C \times T_c$~~ $T = C \times T_c$ since, $T = I_c \times CPI \times T_c \Rightarrow \frac{I_c}{T} = \frac{C}{CPI}$

Therefore, $MIPS = \frac{I_c}{T \times 10^6} = \frac{f \times I_c}{CPI \times 10^6} = \frac{f \times I_c}{C \times 10^6}$

(V) Throughput Rate -

How many programs a system can execute per unit time, called the system throughput W_s

In multiprogrammed system, $W_s < W_p$ (CPU throughput)

$$W_p = \frac{f}{I_c \times CPI} = \frac{\text{MIPS} \times 10^6}{I_c}$$

due to I/O, compiler & OS overhead

(3) Parallel Computer models -

(i) Shared memory multiprocessor

→ UMA model (Uniform Memory Access) (shared local memory)
EBN Butterfly

→ NUMA model (Non-uniform memory access) → Cedar system

→ COMA model (Cache-only memory architecture) (Hierarchical cluster model)

(ii) Distributed memory multiprocessors

(4) Multivectors and SIMD computers -

(i) Architecture of a vector supercomputer

(ii) SIMD Computers

→ SIMD machine model → 5 tuples → $M = \{N, L, I, M, R\}$

→ Operational model of SIMD computers.

(5) Data and Resource Dependences -

(i) Data Dependence -

→ Flow dependence ($s_1 \rightarrow s_2$) → output of $s_1 \equiv$ input of s_2

→ Anti dependence ($s_1 \rightarrow s_2$) → program order, output of $s_2 \equiv$ input of s_1

→ Output dependence ($s_1 \circ \rightarrow s_2$) → same output variable

→ I/O dependence → same file is referenced by both I/O statements

→ Unknown dependence (4 conditions) → subscript.

(ii) Control dependence - dependent upon conditional, loop etc statements.

(iii) Resource dependence - Conflicts in using shared resources.

Conflicting resources in A&V → A&V dependence.

Conflicts involve workplace storage → storage dependence.

(6) Hardware and Software Parallelism

⑦ Program Partitioning and Scheduling -

(i) Grain size and latency -

Grain size or granularity is a measure of the amount of computation mixed in a software process. Eg- No. of instructions in a grain. Latency is a time measure of the communication overhead incurred between machine and subsystems Eg- Memory latency.

(ii) levels of parallelism -

- Instruction level
- loop level
- Procedure level
- Subprogram level
- Job (program) level

(iii) Communication latency - Inter processor communications

⑧ Program Flow Mechanisms -

statements are executed when

Control Flow (Control-driven) → depends on token of control

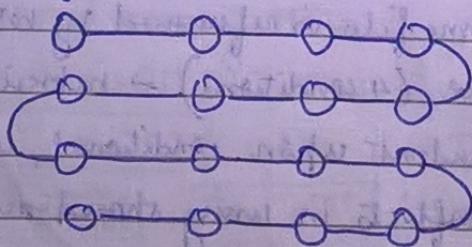
Data Flow (Data-driven) → all of their operands are available

Reduction (Demand-driven) → their result is required for another computation

⑨ Static Interconnection Networks -

Point-to-point direct connections which will not change during program execution. Static Networks use direct links which are fixed once built.

(1) Linear Array - N nodes $N-1$ links

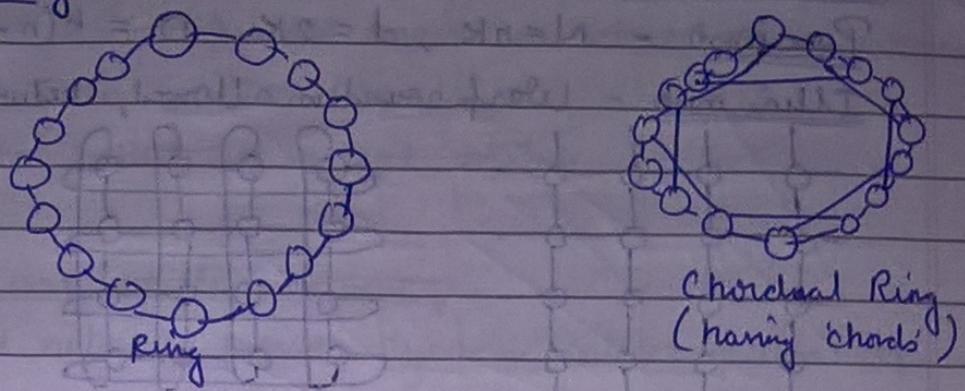


Network complexity = $O(N)$

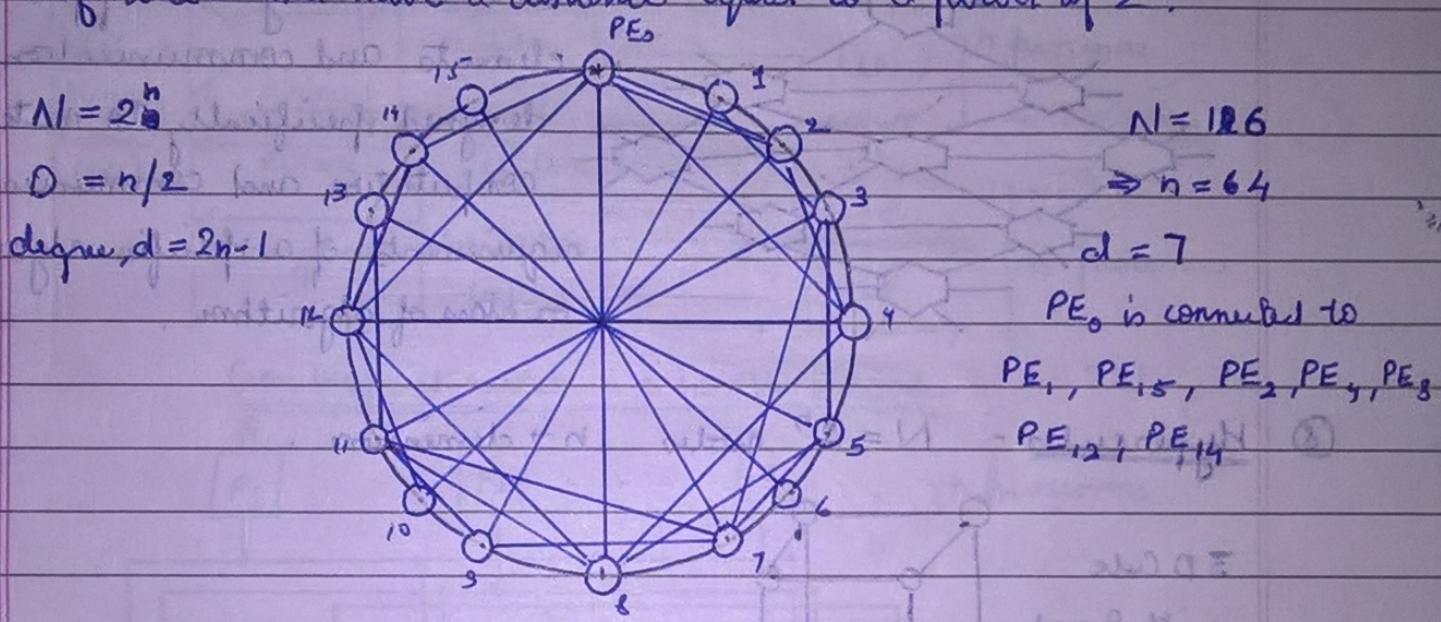
Time complexity = $O(N)$

Diameter, $D = N-1$

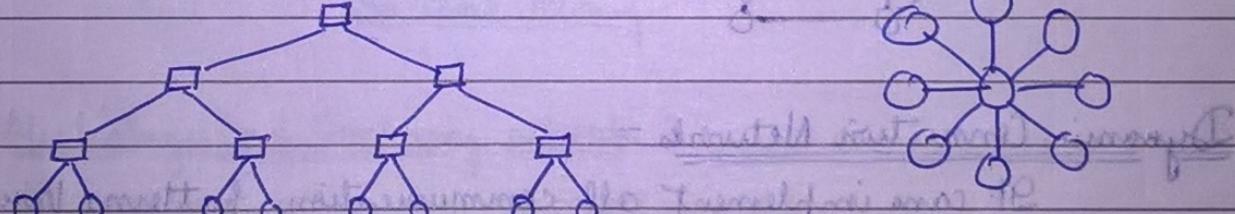
(2) Ring, Chordal Ring - N nodes N links. $D = N/2$



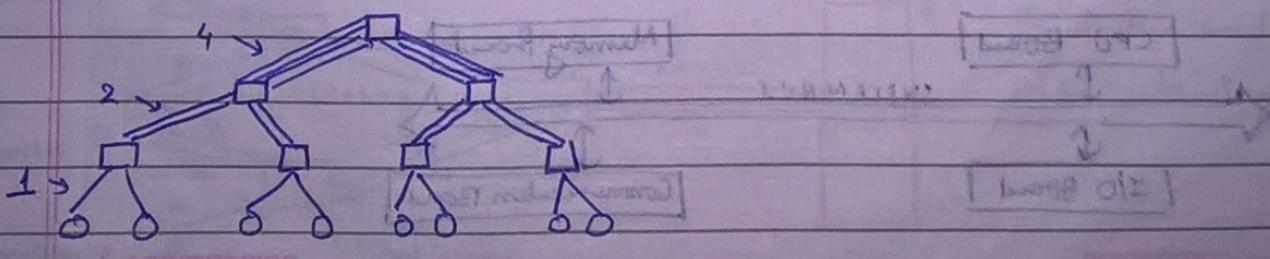
(3) Bucket Shifter - Ring with additional ~~Aug~~ links between all pairs of nodes that have a distance equal to a power of 2.



(4) Tree and Star - $N = 2k-1$ (Binary Tree), $N = k$, $d = N-1$ (Star)



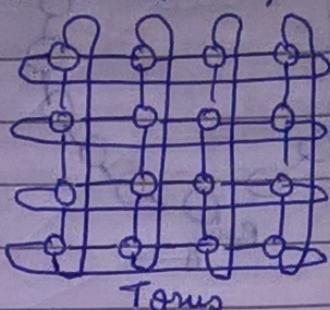
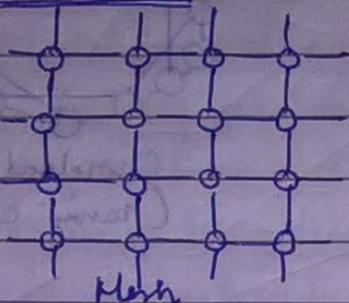
(5) Fat Tree - Number of nodes increases close to the root.



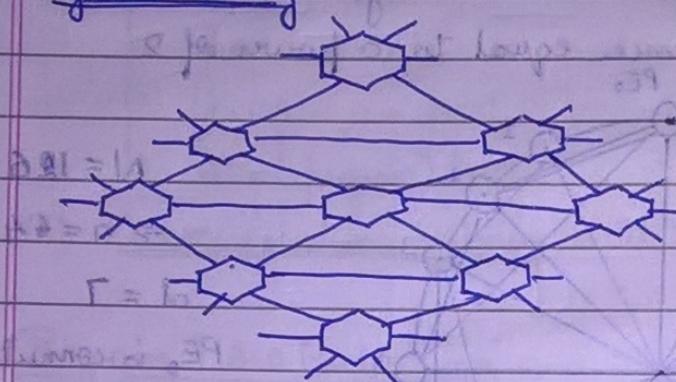
⑥ Mesh and Torus -

Pine mesh - $N = nk$, $d = 2k$, $D = k(n-1)$

Illinois mesh - Wrap round is allowed, network diameter becomes half.

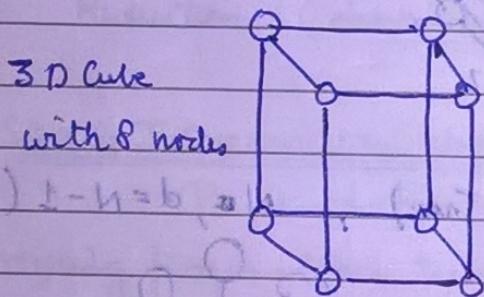


⑦ Systems array -



It is an arrangement of processing elements and communication links designed specifically to match the computation and communication requirements of a specific algorithm or class of algorithm.

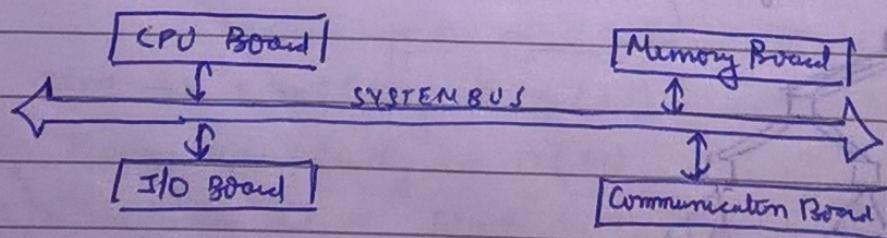
⑧ Hypercubes - $N = 2^n$ nodes $n \rightarrow$ dimension.



⑨ Dynamic Connection Networks -

It can implement all communication patterns based on program demands.

→ Bus System -



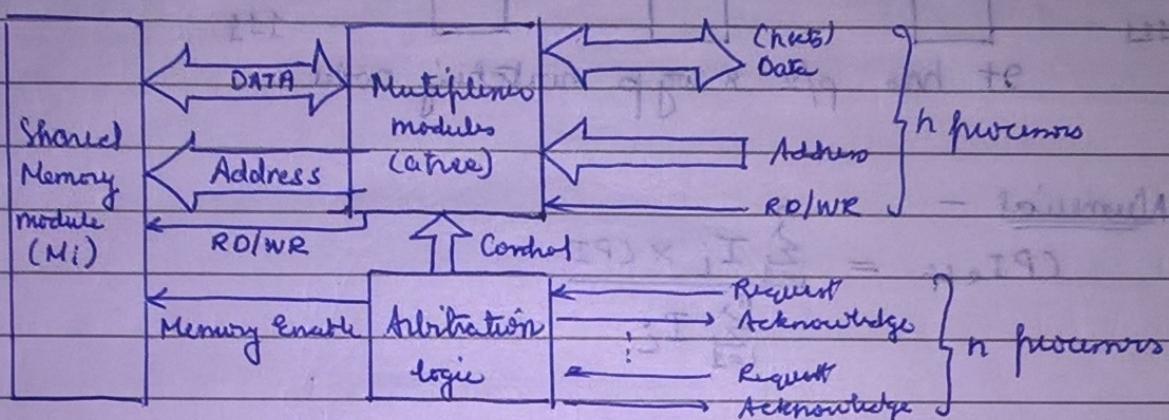
local Bus - Buses implemented as printed circuit boards

Backplane Bus - It is a printed circuit on which many connectors are used to plug-in functional boards

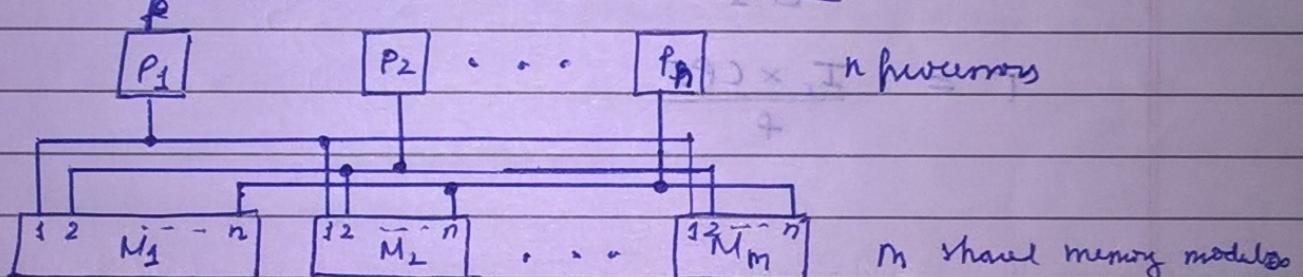
I/O Bus - For input and output operations.

→ Crossbar switch and multiplex memory-

Switched networks provide dynamic interconnection between the inputs and outputs.

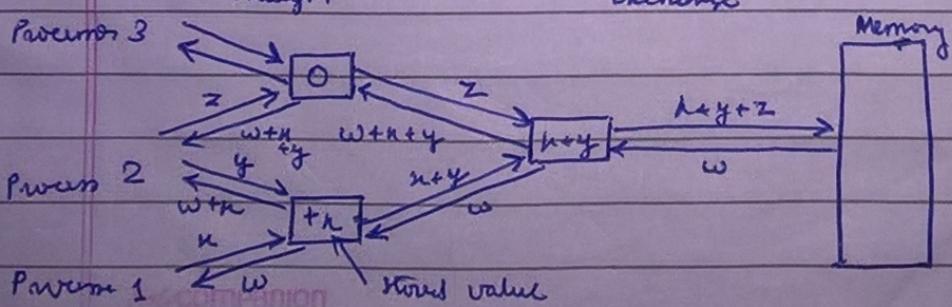
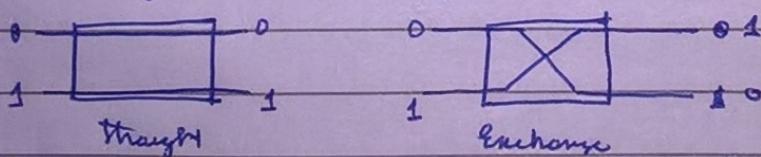


Crossbar switch in a crossbar network



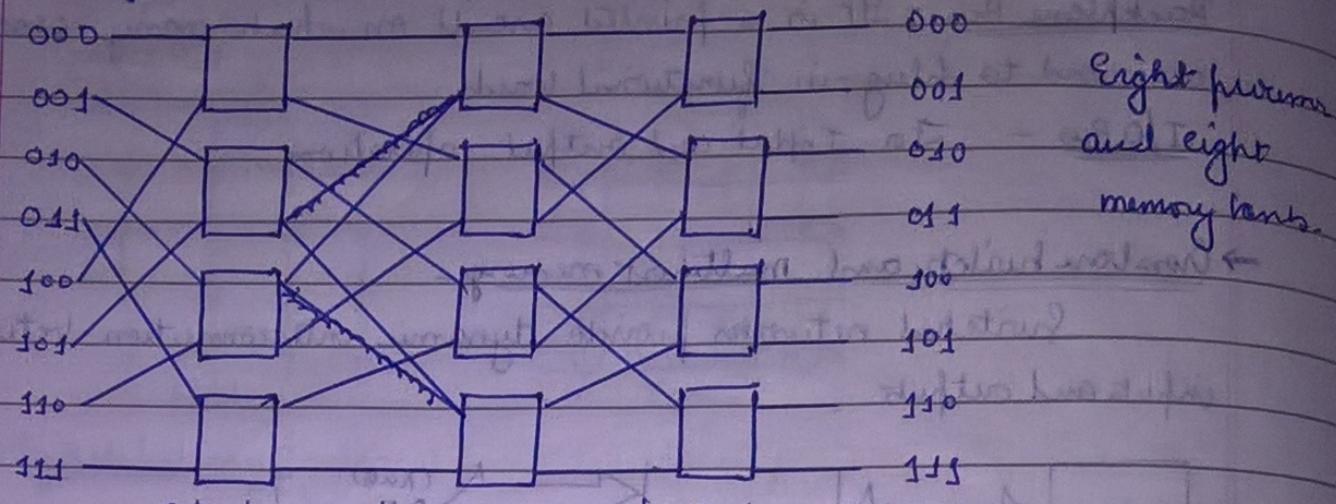
Multiplex memory

→ Multistage and Combining network -



Fetch and add operation in a multistage network

Omega network -



It has $p/2 \times \log p$ switching nodes

(11)

Numerical -

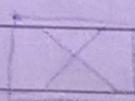
$$CPI_{\text{eff}} = \frac{\sum_{i=1}^n I_i \times CPI_i}{\sum_{i=1}^n I_i}$$

$$\text{MIPS} = \frac{f}{CPI} \times 10^{-6}$$

$$T = \frac{I_c \times CPI}{f}$$

memory contention

solution provided by you will be



Memory