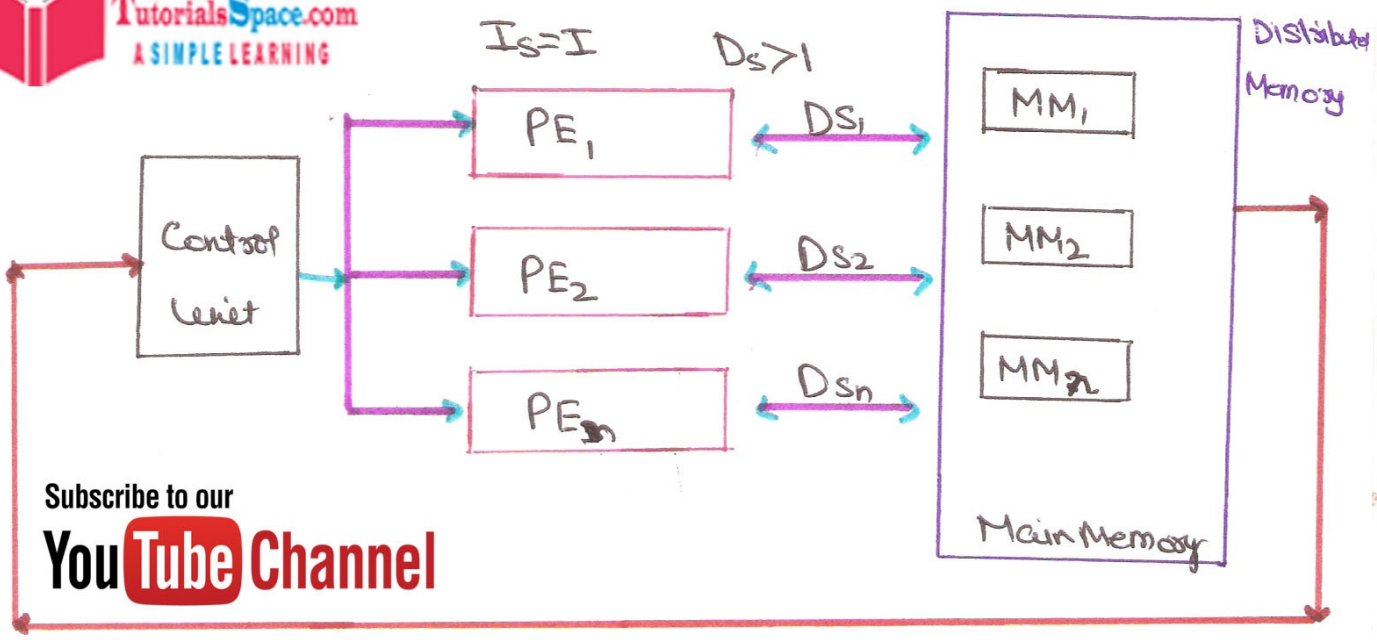
2) Single Instruction and Multiple Data Stream (SIMD)

In this Organisation, Multiple processing elements work under the control of a single control unit.

'It has One Instruction & Multiple Data stream'

→ All the processing elements of this Organization recieve the same instruction broad cast from the CU.

→ Main Memory can also divided into modules for generating multiple Data Stream acting as a Distributed Memory.



SIMD Organization

Therefore, all the processing elements simultaneously execute the same instruction and are said to be 'lock-stepped' together.

→ Each processor takes the data from its own memory and hence it has on distinct data stream.

Some Systems also provide a shared global memory for Communication.

Every processor must be allowed to complete its instruction before the next instruction is taken for execution.

Thus the execution of instructions is Synchronous

Example

ILLIAC-IV

PEPE, MPP

BSP, STARAN &

Connection Machine (CM-1)

Example of instruction & Data:-

Consider the addition of two No. A & B, to create a Sum Vector C. That is

$$C[i] = A[i] + B[i] \quad 1 \leq i \leq N$$

This Computation requires N add times plus the loop Control overhead on an SISD.

Also SISD processor has to fetch the instructions corresponding to this program from the memory each time through loop.

→ The elements of arrays A and B are distributed over N memory blocks and hence each PE has access to one pair

of operands to be added.

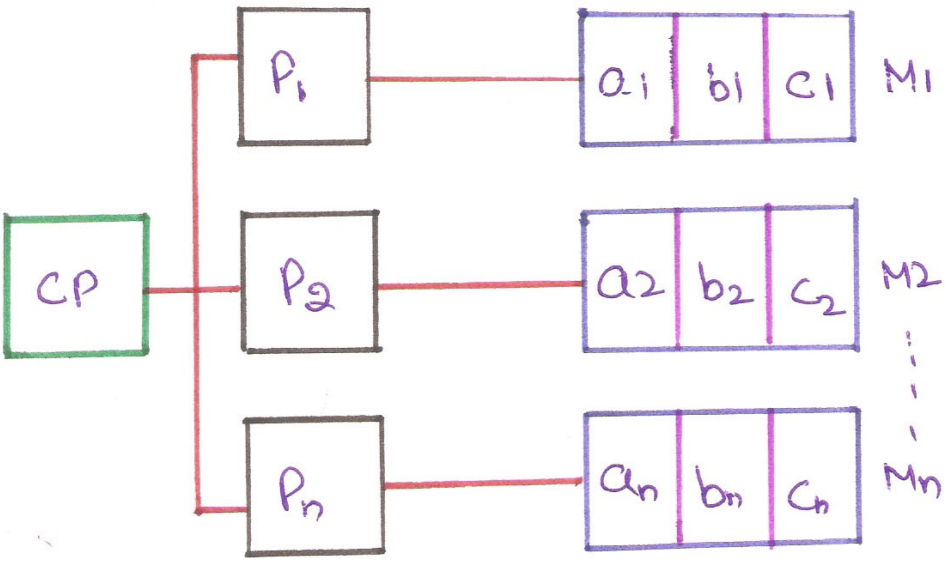
Thus, the program for the SIMD consists of one instruction:-

$$C = A + B$$

This instruction is equivalent to

$$C[i] = A[i] + B[i] \quad (1 \leq i \leq N)$$

where, i represents the PE that is performing the addition of the ith elements and the expression in Parantheses implies that all N



PEs are active Simultaneously.

Total Execution time for this Computation is the time to fetch one instruction plus time for one addition

The Data need to be structured in N memory blocks to provide for the Simultaneous access of the N data elements.

" SIMD more Suitable for ~~Array~~ or ~~Vector~~ Processing. "