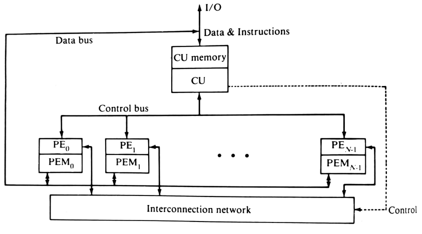
Data Level Parallelism: SIMD Vector Architecture, Graphics Processing Units, Systolic Arrays. Interconnection Networks

Prasanta K. Jana, IEEE Senior Member

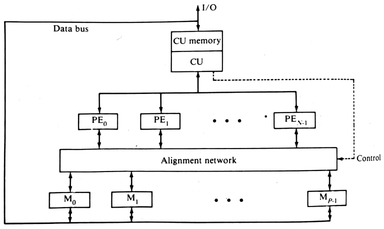
Department of Computer Science and Engineering
Indian Institute of Technology (ISM), Dhanbad
E-mail: prasantajana@yahoo.com

SIMD Computer Organisation

Each PE_i is arithmetic logic unit (ALU).
Each PME_i is local storage.



(a) Configuration I (Illiac IV)



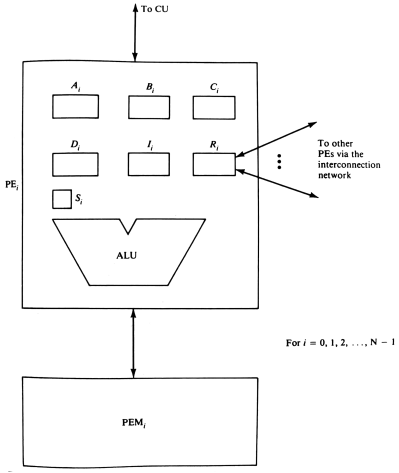
(b) Configuration II (BSP)

SIMD Computer Organisation

- Formally, an SIMD computer C is characterize by the following set of parameters:

$$C = < N, F, I, M >$$

where N = the no. of PEs in the system.
F = a set of data-routing functions.
I = the set of machine instructions.
M = the set of masking schemes.


To CU
A_i
B_i
C_i
D_i
I_i
R_i
S_i
ALU
PE_i
To other PEs via the interconnection network
For $i = 0, 1, 2, \ldots, N-1$
PEM_i

Components in a Processing Element(PE_i)

An Illustration of Data routing and masking schemes

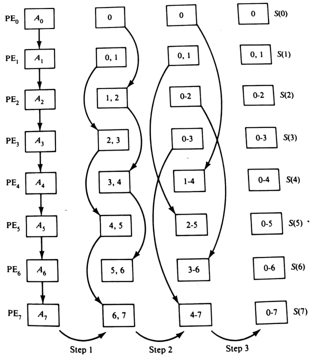
- Let $A = (A_0, A_1, \ldots, A_{n-1})$, we need to compute the following n summations:

$$s(k) = \sum_{i=0}^{k} A_i, \text{ for } k = 0, 1, \ldots, n-1$$

- These n vector summations can be computed recursively by :

$s(0) = A_0$,
$s(k) = s(k-1) + A_k$, for $k = 0, 1, \ldots, n-1$

SIMD Computer Organisation



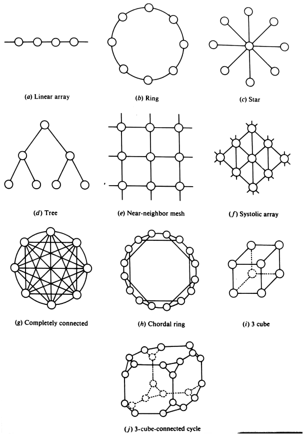
The calculation of the summation $s(k) = \sum_{i=0}^{k} A_i$, for $k = 0, 1, \ldots, 7$ in an SIMD Machine

- Routing function: $R_i(j) = j + 2^i, 0 \leq i \leq log(n) - 1,\ 0 \leq j \leq n - 1$

SIMD Interconnection Networks

- Static versus Dynamic networks.
  The SIMD interconnection networks are classified into the following two categories based on network topologies: Static networks versus dynamic networks.

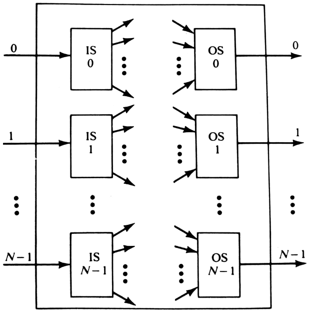
SIMD Interconnection Networks



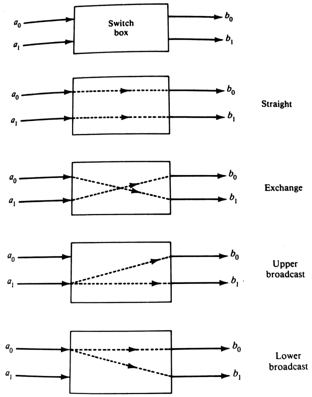
Static interconnection network topologies

Dynamic networks

- We consider two classes of dynamic networks:
  - Single-stage
  - Multi-stage



Conceptual view of a single-stage interconnection network



A two-by-two switching box and its four interconnection states.

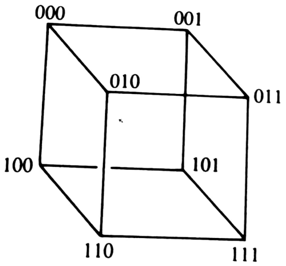
Evaluation criteria of Interconnection network

- **Diameter**: Maximum of shortest distance between any two arbitrary nodes.
- **Bisection width**: The minimum number of edges to be removed to divide the network into two halves.
- **Number of edges per node.**
- **Maximum edge length.**

- Dynamic networks can be divided into three classes:
    - Blocking
    - Rearrangeable
    - Nonblocking

- **Blocking networks**: Simultaneous connections of more than one terminal pair may result in conflicts. Examples of a blocking network are data manipulator, omega, flip, n cube, and baseline.
- **Rearrangeable networks**: It can perform all possible connections between inputs and outputs by rearranging its existing connections so that a connection path for a new input-output pair can always be established. Example is Benes network.

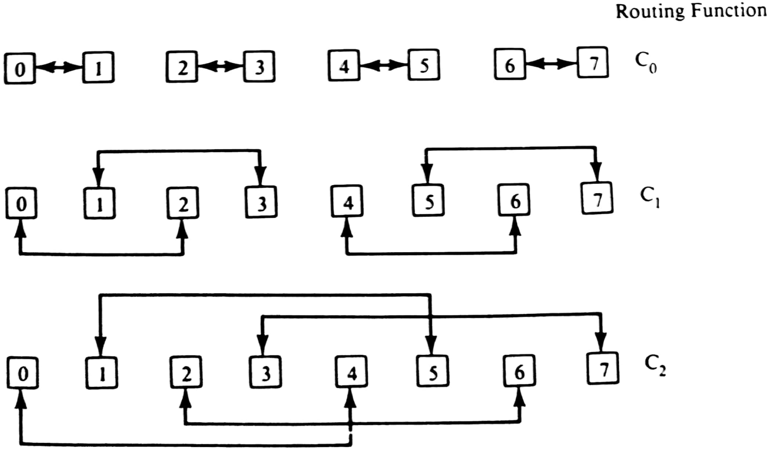
Cube Interconnection network



A 3 cube of 8 nodes.

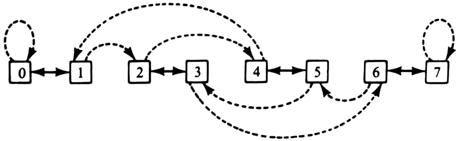
- Cube routing function: $C_i(a_{n-1}\dots a_1a_0) = a_{n-1}\dots a_{i+1}\overline{a_i}a_{i-1}\dots a_0$, for $i = 0, 1, \dots, n-1$

Implementation as a single stage network

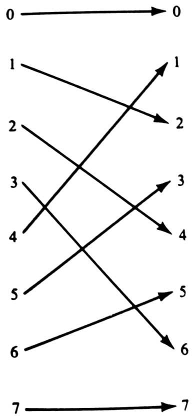


(*b*) The recirculating cube network

- Shuffle function is defined as follows:
  $S(a_{n-1} \dots a_1 a_0) = a_{n-2} \dots a_1 a_0 a_{n-1}$
- Inverse shuffle function is defined as follows:
  $IS(a_{n-1} \dots a_1 a_0) = a_0 a_{n-1} \dots a_1$
- Exchange routing function is defined as follows:
  $E(a_{n-1} \dots a_1 a_0) = a_{n-1} \dots a_1 \overline{a_0}$
- Note that $E(A) = C_0(A)$
- Butterfly function: $\beta_k(b_{n-1} b_{n-2} \dots b_{k+1} b_k b_{k-1} \dots b_1 b_0) =$
  $b_{n-1} b_{n-2} \dots b_{k+1} b_0 b_{k-1} \dots b_1 b_k$

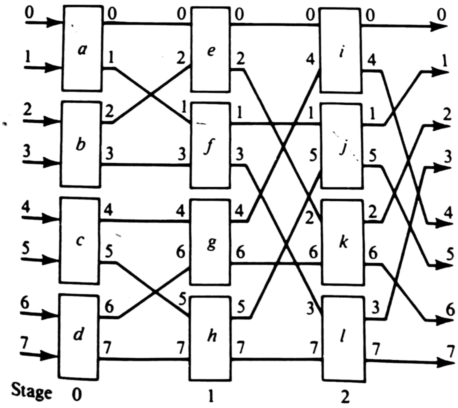


Single stage shuffle exchange network (solid lines are exchange and dashed lines are shuffle).



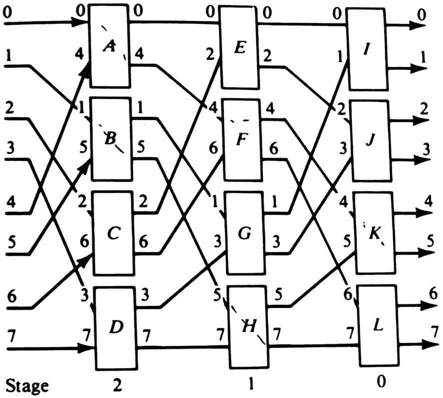
Shuffle connection

Multi-stage implementation

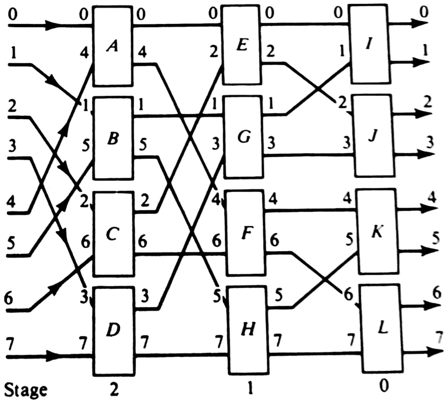


A Multistage cube network for $N = 8$

Multi-stage implementation



Three stage 8×8 Omega switching network

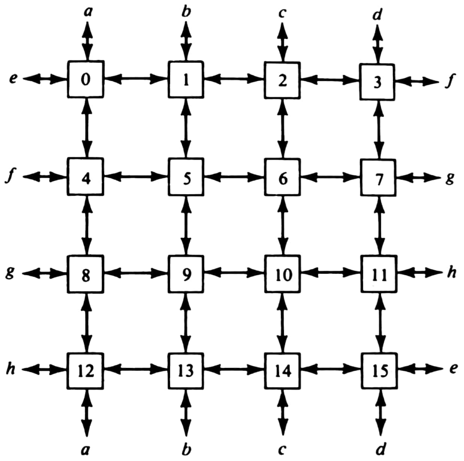


The Omega network with switch box repositioned

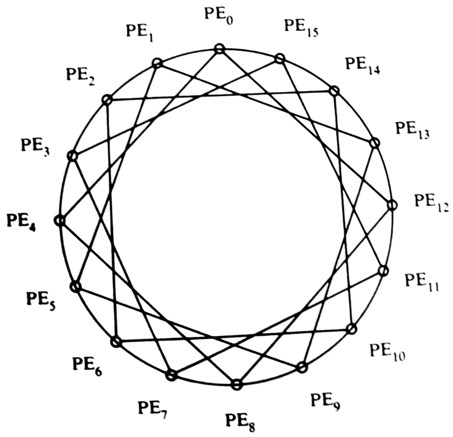
- The cube network uses two function switch boxes, whereas the omega network uses four function switch boxes.
- The data flow directions in the two networks are opposite to each other. In other words, the roles of input-output lines are exchanged in the two networks.

Mesh Connected Illiac Networks

$$
\begin{aligned}
R_{+1}(i) &= (i+1)\ mod\ N \\
R_{-1}(i) &= (i-1)\ mod\ N \\
R_{+r}(i) &= (i+r)\ mod\ N \\
R_{-r}(i) &= (i-r)\ mod\ N
\end{aligned}
$$



An Illiac network with $N = 16$ and $r = 4$



An Illiac network with $N = 16$ PEs

- The permutation cycles $(a, b, c), (d, e)$ stands for the permutation $a \to b$, $b \to c$, $c \to a$, and $d \to e$, $e \to d$ in a circular fashion within a pair of parenthesis:

$$\begin{array}{l} R_{+1} = (0\ 1\ 2\ \ldots\ N-1) \\ R_{-1} = (N-1\ \ldots\ 2\ 1\ 0) \end{array}$$

$$R_{+r} = \prod_{i=0}^{r-1} (i\ \ i+r\ \ i+2r\ \ \ldots\ \ i+N-r)$$

$$R_{-r} = \prod_{i=0}^{r-1} (i+N-r\ \ \ldots\ \ i+2r\ \ i+r\ \ i)$$

Barrel shifter

- Routing functions:

$$B_{+j}(i) = (i + 2^j)(mod\ N)$$
$$B_{-j}(i) = (i - 2^j)(mod\ N)$$

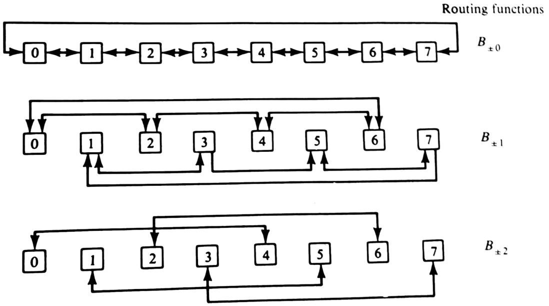
where $0 \leq i \leq N-1,\ 0 \leq j \leq N-1$ and $n = log_2 N$

$$B_{+0} = R_{+1}$$
$$B_{-0} = R_{-1}$$
$$B_{+n/2} = R_{+r}$$
$$B_{-n/2} = R_{-r}$$

$B \leq (log_2 N)/2$

$$B_{\pm 0}(j) = (j \pm 1)\ mod\ 8$$
$$B_{\pm 1}(j) = (j \pm 2)\ mod\ 8$$
$$B_{\pm 2}(j) = (j \pm 4)\ mod\ 8$$

where $N = 8$ and $0 \leq j \leq 7$



A recirculating barrel shifter for $N = 8$.

$$S_N = \frac{\sqrt{N}-1}{(\log_2 N)/2} = \frac{2^k - 1}{k}$$

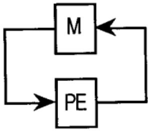
Systolic System

Systolic System

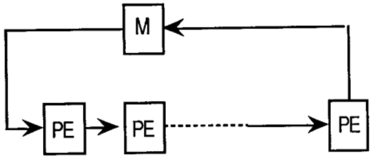
- Consist of a set of inter-connected cells
- Each capable of performing some simple operation

Basic Principle

- By replacing a simple PE with an array of PEs, a higher computation throughput can be achieved (as shown in figure) without increasing memory.



Conventional Processor

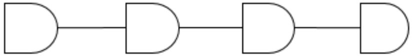


A Systolic Processor Array

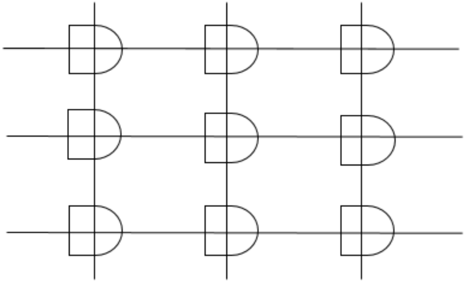
Basic Principle

- The function is an analogous to that or heart ,that is
  - It 'pulses' data through the array of PEs.Once data item is brought out from the memory.
  - It can be used effectively at each cell,it passes.
  - This is possible for a wide class of computed bound computation.

Systolic Arrays

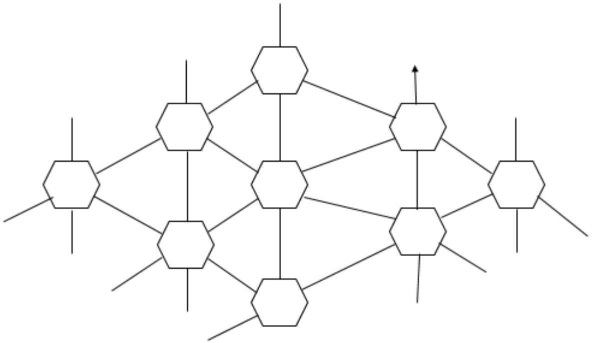


Fir-filter, Convolution,DFT etc.



Mesh for DP, Graph Algorithm Matrix-multi

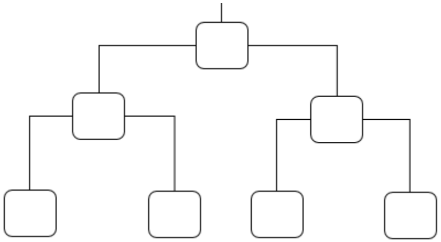
Systolic Arrays



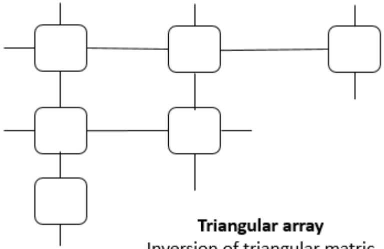
Matrix multi; (banded), LU decomposition

Two dimensional hexagonal array

Systolic Arrays



Tree Searching , Recurrence evaluation etc.

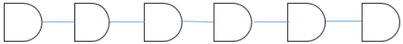


Triangular array

Inversion of triangular matric

Graeffe's Root Squaring

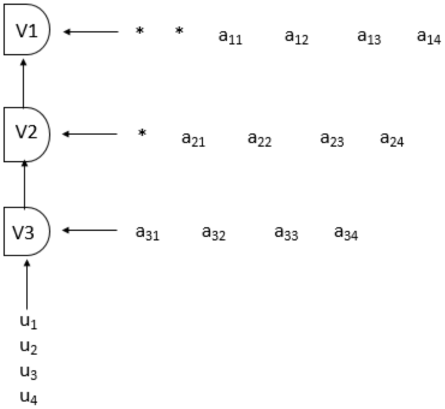
$$C_j = {A_j}^2 - 2A_{j-1}A_{j+1} + 2A_{j-2}A_{j+2} - 2A_{j-3}A_{j+3} + 2A_{j-4}A_{j+4} - ...$$



A_0	A_1	A_2	A_3	A_4	A_5	$\rightarrow$
A_0	A_1	A_2	A_3	A_4	A_5	$\leftarrow$
0	A_0	A_1	A_2	A_3	A_4	
A_1	A_2	A_3	A_4	A_5	0	

Matric Vector Multiplication

$$[A]_{3\times 4} \times [U]_{4\times 1} = [V]_{3\times 1}$$


V1
* * a_{11} a_{12} a_{13} a_{14}
V2
* a_{21} a_{22} a_{23} a_{24}
V3
a_{31} a_{32} a_{33} a_{34}
u_1
u_2
u_3
u_4

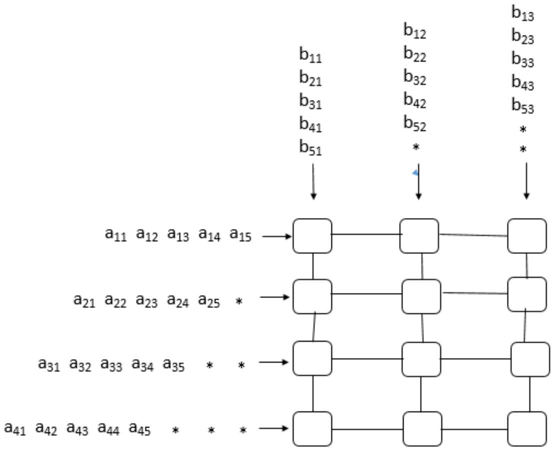
Matrix Vector Multiplication

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{11} & a_{22} & a_{23} & a_{24} \\ a_{11} & a_{11} & a_{11} & a_{11} \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{11} \\ a_{11} \\ a_{11} \end{pmatrix}$$

- Convolution : Given $\{w_1, w_2, ..., w_n\}$ and an input sequence $\{x_1, x_2, ..., x_n\}$
  Compute $\{y_1, y_2, ..., y_{2n-1}\}$
  by $y_i = \sum_{j=1}^{n} x_{i-j+1} \times w_j$ where $1 \leq i \leq 2n-1$
  For $n = 3$, this is equivalent or

$$\begin{bmatrix} x_1 & 0 & 0 \\ x_2 & x_1 & 0 \\ x_3 & x_2 & x_1 \\ 0 & x_3 & x_2 \\ 0 & 0 & x_3 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix}$$

Matrix-Matrix Multiplication



b_{11} b_{21} b_{31} b_{41} b_{51}
b_{12} b_{22} b_{32} b_{42} b_{52} *
b_{13} b_{23} b_{33} b_{43} b_{53} * *
a_{11} a_{12} a_{13} a_{14} a_{15}
a_{21} a_{22} a_{23} a_{24} a_{25} *
a_{31} a_{32} a_{33} a_{34} a_{35} * *
a_{41} a_{42} a_{43} a_{44} a_{45} * * *

Matrix-Matrix Multiplication

$(A)_{m\times n}, (B)_{n\times k}$

```
Procedure MESH-MUL(A,B,C)
 for i=1 to m do in parallel
  for j=1 to k do in parallel
      $(1)\ e_{ij} \leftarrow 0$
      $(2)$ while P $(i,j)$ receives two inputs a & b do
          $(i)\ e_{ij} \leftarrow e_{ij} + a\times b$
          $(ii)$ if $i<m$ then send b to P$(i+1,j)$ endif
          $(iii)$ if $i<k$ then send a to P $(i,j+1)$ endif
      endwhile
  endfor
 endfor
```

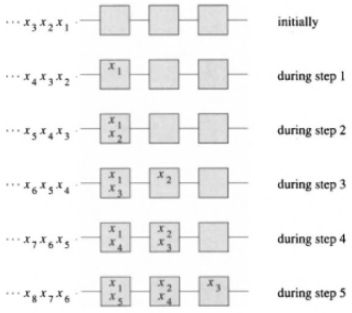
- Time Complexity : Elements a_{m1} and b_{1k} take $m+k+n-2$ steps from the beginning to reach P(m,k), the last processor to terminal. Assuming $n\geq m$ and $n\geq k$ it requires $O(n)$ time.

Palindrome Recognition

Palindrome Recognition

- Palindrome is a string of the form ww^r where w^r is the reverse of w.For example, 110011 is a palindrome, but $110001, 101$ are not.
  Input : $x_1, x_2, ..., x_w$
  Output : Whether or not $x_1, x_2, ..., x_i$ is a palindrome immediately after inputting x_i.

Palindrome Recognition



- Checking : The string $x_1, x_2, ..., x_i$ is a palindrome ,iff, each cell contains identical values at the i^{th} step.

Graphics Processing UNIT

- Definition: A Graphics Processing Unit (GPU) is a single-chip processor primarily used to manage and boost the performance of video and graphics. GPU features include:
- 2-D or 3-D graphics
- Digital output to flat panel display monitors
- Application support for high-intensity graphics software such as AutoCAD
- Used in a PC on a video card or motherboard; it is also used in mobile phones, display adapters, workstations and game consoles.
- These features are designed to lessen the work of the CPU and produce faster video and graphics.

Graphics Processing UNIT

- The first GPU was developed by NVidia in 1999 and called the GeForce 256.
- This GPU model could process 10 million polygons per second and had more than 22 million transistors.
- Generally the GPU is connected to the CPU and is completely separate from the motherboard.
- The random access memory (RAM) is connected through the accelerated graphics port (AGP) or the peripheral component interconnect express (PCI-Express) bus.
- Most GPUs use their transistors for 3-D computer graphics.
- Applications such as computer-aided design (CAD) can process over 200 billion operations per second

Graphics Processing UNIT

- GPU-accelerated computing is the use of a GPU together with a CPU to accelerate deep learning, analytics, and engineering applications.
- GPU-accelerated computing offloads compute-intensive portions of the application to the GPU, while the remainder of the code still runs on the CPU. From a user's perspective, applications simply run much faster.

GPU vs CPU Performance

- A simple way to understand the difference between a GPU and a CPU is to compare how they process tasks.
- A CPU consists of a few cores optimized for sequential serial processing while a GPU has a massively parallel architecture consisting of thousands of smaller, more efficient cores designed for handling multiple tasks simultaneously.
- GPUs have thousands of cores to process parallel workloads efficiently

THANK YOU