PC DC work

NVIDIA GTC (GPU Technology Conference) is a global AI conference for developers that brings together developers, engineers, researchers, inventors, and IT professionals.

CONCURRENCY :

Computer science defines concurrency as **a property of systems where several processes are executing at the same time, and may or may not interact with each other**.

CONCURRENT SYSTEM :

A concurrent system is **one where a computation can advance without waiting for all other computations to complete**. Concurrent computing is a form of modular programming. In its paradigm an overall computation is factored into subcomputations that may be executed concurrently.

INSTRUCTION STREAM:

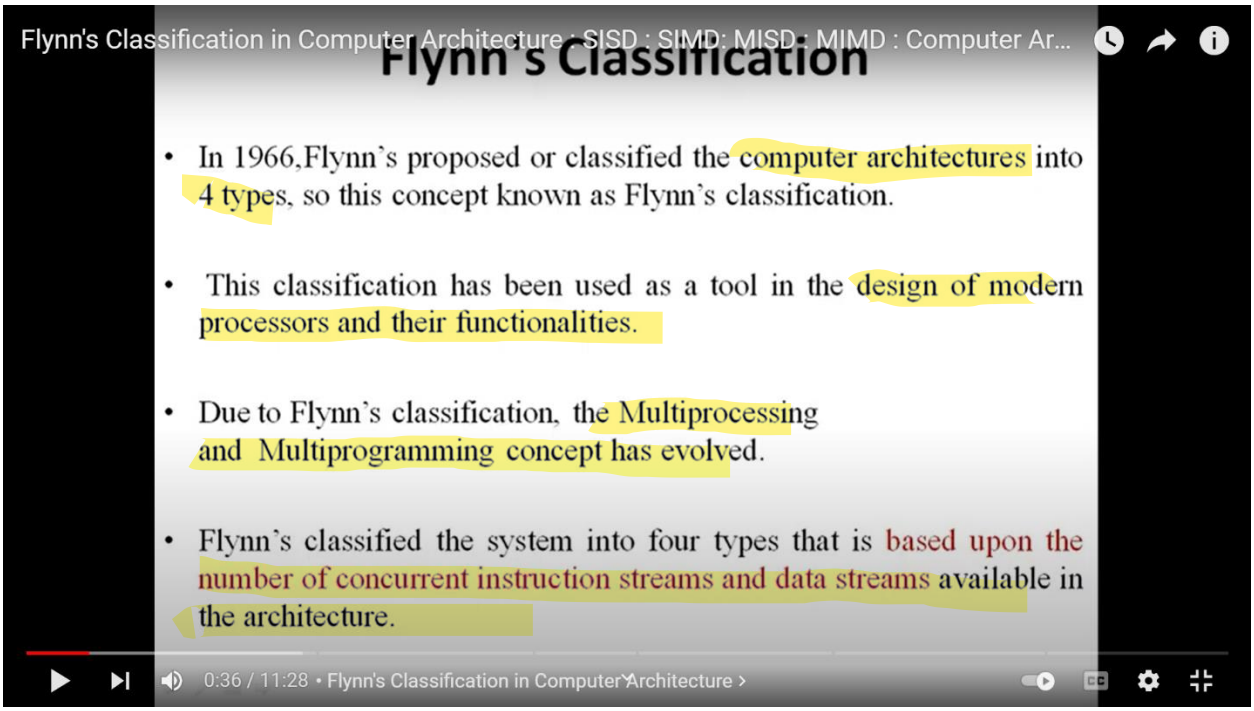
Jb hamare processor ke pass instr ponchty hain to wo process kis type ke instruction ko lega aur kese wo execute krega use .

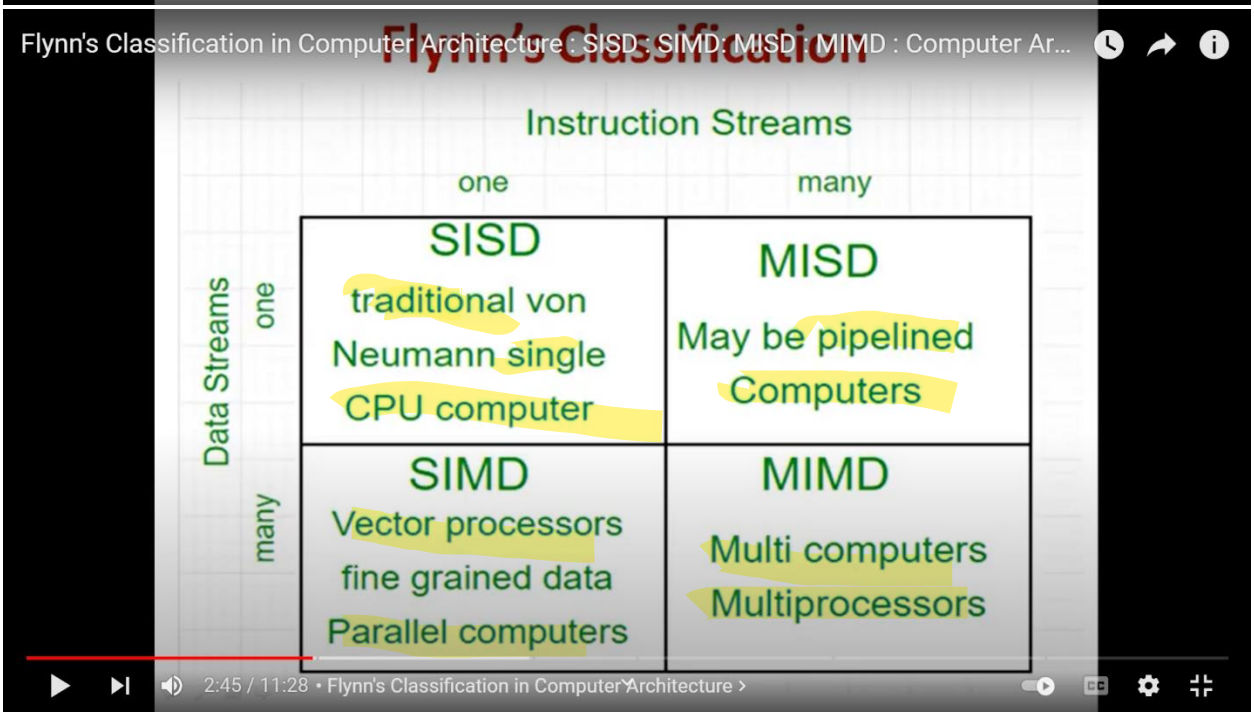
DATA STREAM :I

Kese memory se data ata hai aur kese processor use executer krega .

SISD:

One instr and one data pr kam krta hai


Flynn's Classification in Computer Architecture : SISD : SIMD: MISD : MIMD : Computer Ar...
Flynn's Classification
• In 1966,Flynn's proposed or classified the computer architectures into 4 types, so this concept known as Flynn's classification.
• This classification has been used as a tool in the design of modern processors and their functionalities.
• Due to Flynn's classification, the Multiprocessing and Multiprogramming concept has evolved.
• Flynn's classified the system into four types that is based upon the number of concurrent instruction streams and data streams available in the architecture.
0:36 / 11:28 • Flynn's Classification in Computer Architecture ›


Flynn's Classification in Computer Architecture : SISD : SIMD: MISD : MIMD : Computer Ar...
Flynn's Classification
Instruction Streams
one
many
Data Streams
one
many
SISD
traditional von Neumann single CPU computer
MISD
May be pipelined Computers
SIMD
Vector processors
fine grained data
Parallel computers
MIMD
Multi computers
Multiprocessors
2:45 / 11:28 • Flynn's Classification in Computer Architecture ›

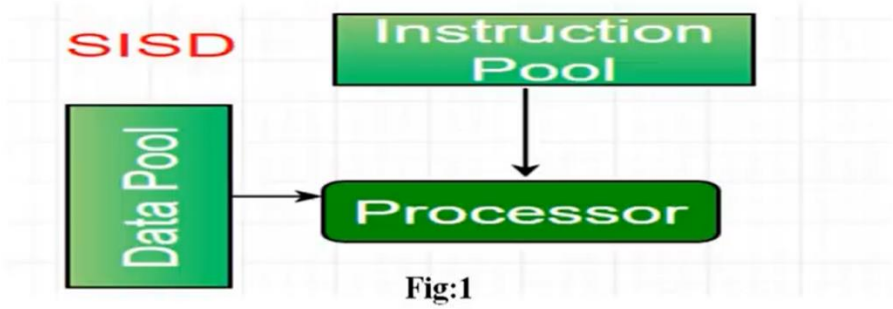
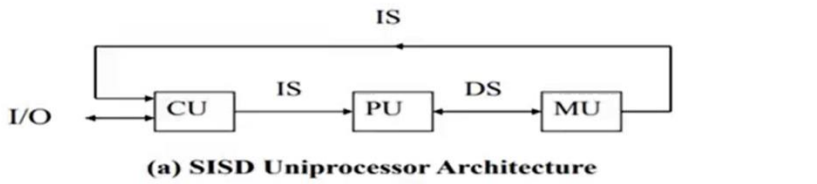


Fig:1



(a) SISD Uniprocessor Architecture

Captions:

CU - Control Unit ; PU – Processing Unit

MU – Memory Unit ; IS – Instruction Stream

DS – Date Stream

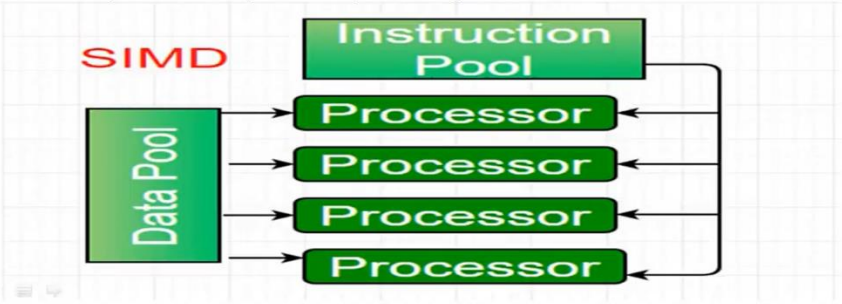
Fig:2

Single Instruction Single Data (SISD) Systems

- It is a uni-processor machine
- It executes a single instruction which operate on a single data stream.
- In SISD, machine instructions are processed in a sequential manner, so it is known as sequential computers.
- In this, the speed of the processing element in the SISD model is limited or dependent on the rate at which the information is transferred.

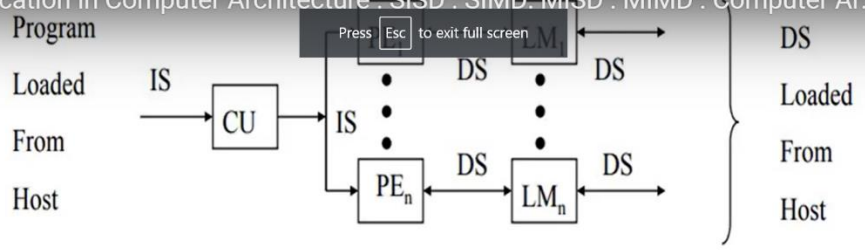
Single Instruction Multiple Data (SIMD) systems

- SIMD is a multiprocessor system.
- It execute the **same instruction** on all the CPUs but operate on different data streams.
- **SIMD model is well suited to scientific computing because it involve lots of vector and matrix operations.** So that the information can be passed to all the processing elements (PEs) organized data elements of vectors can be divided into multiple sets(N-sets for N PE systems) and each PE can process one data set.
- SIMD systems is Cray's vector processing machine.



Flynn's Classification in Computer Architecture : SISD : SIMD: MISD : MIMD : Computer Ar...

Press Esc to exit full screen



(b) SIMD Architecture (with Distributed Memory)

Captions:

CU - Control Unit ; PU - Processing Unit

MU - Memory Unit ; IS - Instruction Stream

DS - Date Stream ; PE – Processing Element

LM – Local Memory

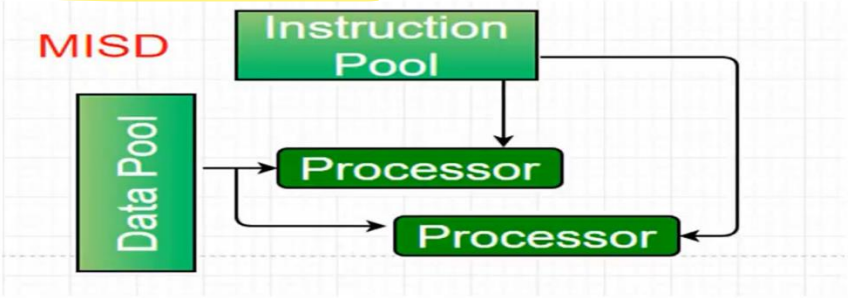
7:17 / 11:28 • Single Instruction Multiple Data (SIMD) systems >

Multiple Instruction Single Data (MISD) Systems

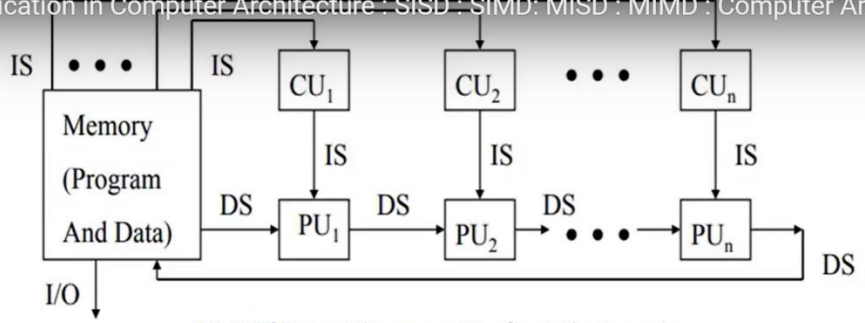
- It is a multiprocessor machine.
- It execute **different instructions** on different PE(Processing Elements) but all of them **operates on the same dataset** .

Example : = sin(x) + cos(x) + tan(x)

- **It performs different operations on the same data set.**
- The computer system built using the MISD model are not useful in most of the application.



Flynn's Classification in Computer Architecture : SISD : SIMD: MISD : MIMD : Computer Ar...

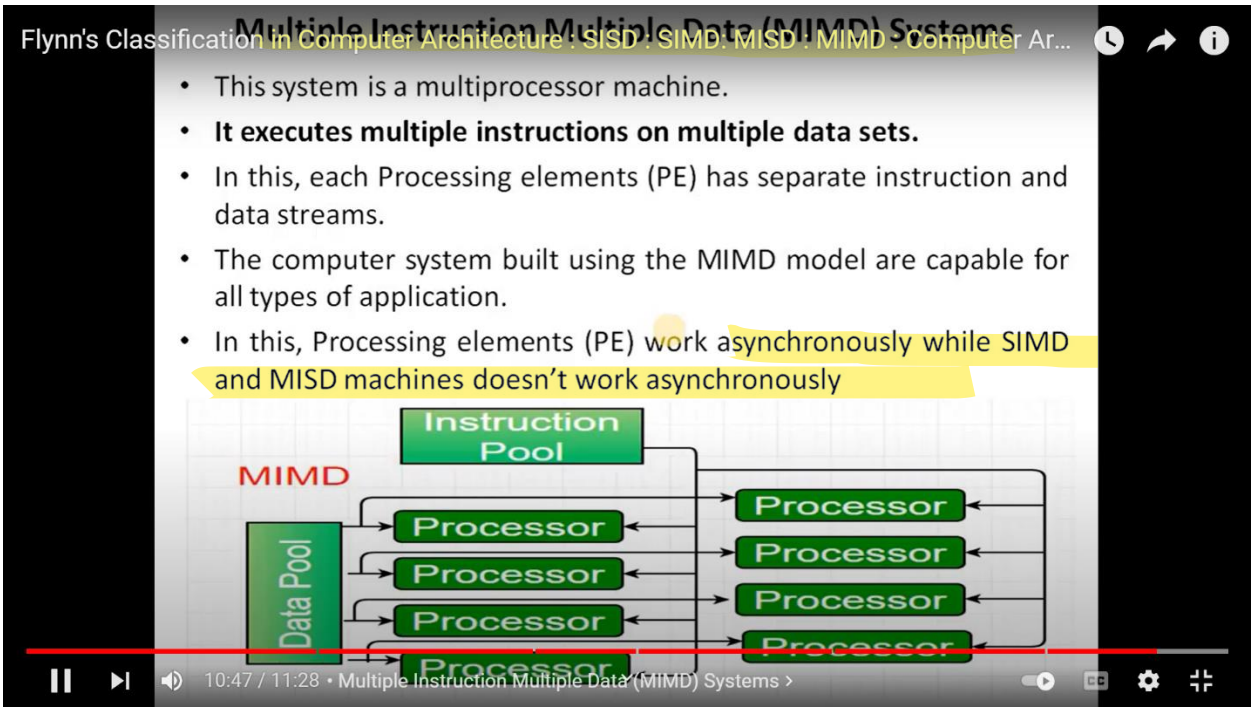


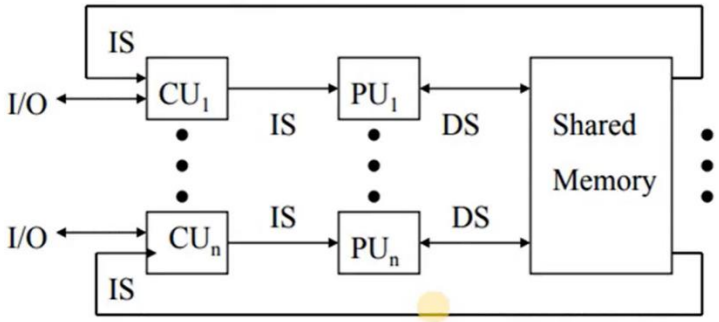
(d) MISD Architecture (the Systolic Array)

Captions:

CU - Control Unit	;	PU - Processing Unit
MU - Memory Unit	;	IS - Instruction Stream
DS - Date Stream	;	PE – Processing Element
LM – Local Memory		

9:24 / 11:28 • Multiple Instruction Single Data (MISD) Systems >





(c) MIMD Architecture (with Shared Memory)

Captions:

CU - Control Unit ; PU - Processing Unit

MU - Memory Unit ; IS - Instruction Stream

DS - Date Stream ; PE – Processing Element

LM – Local Memory

MULTI TASKING AND MULTI THREADING :

Multitasking lets the CPU perform various tasks simultaneously (threads, process, program, task), while multithreading helps in the execution of various threads in a single process simultaneously . The processes in multi-tasking, unlike multi-threading, share separate resources and memories.

THREAD :

Definition: A thread is **a single sequential flow of control within a program**. The real excitement surrounding threads is not about a single sequential thread. Rather, it's about the use of multiple threads running at the same time and performing different tasks in a single program

Diference Between Multitasking and Multithreading

Parameters	Multi-tasking	Mu
Basics	The process of multi-tasking lets a CPU execute various tasks at the very same time.	The process of multi-threading lets a CPU generate multiple threads out of a task and process all of them simultaneously.
Working	A user can easily perform various tasks simultaneously with their CPU using multi-tasking.	A CPU gets to divide a single program into various threads so that it can work more efficiently and conveniently. Thus, multi-threading increases computer power.
Resources and Memory	The system needs to allocate separate resources and memory to different programs working simultaneously in multi-tasking.	The system allocates a single memory to any given process in multi-threading. The various threads generated out of it share that very same resource and memory that the CPU allocated to them.
Switching	There is the constant switching between various programs by the CPU.	The CPU constantly switches between the threads and not programs.
Multiprocessing	It involves multiprocessing among the various components.	It does not involve multiprocessing among its various components.
Speed of Execution	Executing multi-tasking is comparatively slower.	Executing multi-threading is comparatively much faster.
Process Termination	The termination of a process takes up comparatively more time in multi-tasking.	The termination of a process takes up comparatively less time in multithreading.

MULTIPROGRAAMING :

Execution of multi process using single processor .

Multiprogramming is **a rudimentary form of parallel processing in which more then one process/programs are run at the same time on a uniprocessor**. Since there is only one processor, there can be no true simultaneous execution of different programs.

Multiprogramming means **interleaved execution of several tasks on the same computer system**. One of the major aims of multiprogramming is to manage the various resources of the entire system. Examples of multiprogramming operating systems are Windows, IOS, Excel, Firefox, etc

MULTIPROCESSING :

Execution of multi process using multiple processors .

MULTITASKING :

Single cpu/processor is shared among multiple processes like multiprogramming but this time cpu would execute one process for a particular time frame known as quantum .

Multithreading :

Execution of multiple taks in same process .

ARRAY PROCESSING AND VECTOR PROCESSING:

A vector processor is in contrast to the simpler scalar processor, which handles only one piece of information at a time.

An array is made up of indexed collections of information called indices, the plural form of the word "index." Though an array can, in rare cases, have only one index collection, a vector is technically indicative of an array with at least two indices. Vectors are sometimes referred to as "blocks" of computer data.

COMPARISON OF VARIOUS TIME COMPLEXITIES :

ORDER : (BIG O NOTATION)

Order means atmost , maximum , upper bound

ORDER OF CONSTANT :

Number jaha pr fix hogya jaha pr apko pta lg gya ke apko yaha se start krke yaha tk pocnhna hai jo bhi no apko dya hai agr wo fixed hogya iska mtln o(1)

DSM:

In computer science, distributed shared memory (DSM) is **a form of memory architecture where physically separated memories can be addressed as a single shared address space**.

What is the difference between shared memory and distributed memory in distributed systems?

Shared memory allows multiple processing elements to share the same location in memory (that is to see each others reads and writes) without any other special directives, while distributed memory requires explicit commands to transfer data from one processing element to another.

SM:
In computer science, shared memory is **memory that may be simultaneously accessed by multiple programs with an intent to provide communication among them or avoid redundant copies**. Shared memory is an efficient means of passing data between programs. For example, **a client process may have data to pass to a server process that the server process is to modify and return to the client**.

DM:
In computer science, distributed memory refers to **a multiprocessor computer system in which each processor has its own private memory**. Computational tasks can only operate on local data, and if remote data are required, the computational task must communicate with one or more remote processors

What is concurrency control explain?

In a database management system (DBMS), concurrency control **manages simultaneous access to a database**. It prevents two users from editing the same record at the same time and also serializes transactions for backup and recovery.

What causes deadlock?

A deadlock occurs **when 2 processes are competing for exclusive access to a resource but is unable to obtain exclusive access to it because the other process is preventing it**. This results in a standoff where neither process can proceed

A deadlock is **a situation in which two computer programs sharing the same resource are effectively preventing each other from accessing the resource, resulting in both programs ceasing to function**. The earliest computer operating systems ran only one program at a time.

CONCURRENCT CONTROL :

Concurrency controlling techniques ensure that multiple transactions are executed simultaneously while maintaining the ACID properties of the transactions and serializability in the schedules.

Locking Based Concurrency Control Protocols

A **lock** is a variable associated with a data item that determines whether read/write operations can be performed on that data item

One-phase Locking Protocol

In this method, each transaction locks an item before use and releases the lock as soon as it has finished using it. This locking method provides for maximum concurrency but does not always enforce serializability.

Two-phase Locking Protocol

In this method, all locking operations precede the first lock-release or unlock operation. The transaction comprise of two phases. In the first phase, a transaction only acquires all the locks it needs and do not release any lock. This is called the expanding or the **growing phase**. In the second phase, the transaction releases the locks and cannot request any new locks. This is called the **shrinking phase**.

Every transaction that follows two-phase locking protocol is guaranteed to be serializable. However, this approach provides low parallelism between two conflicting transactions.

1. LOCK BASED PROTOCOLS read/write operations
2. SHARED LOCK SYSTEMS only on read operations
3. EXCLUSIVE LOCK

What is socket programming?

Socket programming is a way of connecting two nodes on a network to communicate with each other. One socket(node) listens on a particular port at an IP, while other socket reaches out to the other to form a connection. Server forms the listener socket while client reaches out to the server.

https://www.geeksforgeeks.org/socket-programming-cc/

PARALLEL COMPUTING :

Parallel computing refers to the process of breaking down larger problems into smaller, independent, often similar parts that can be executed simultaneously by multiple processors communicating via shared memory, the results of which are combined upon completion as part of an overall algorithm.

Parallel computing is a type of computation in which many calculations or processes are carried out simultaneously. Large problems can often be divided into smaller ones, which can then be solved at the same time

USES:

The primary goal of parallel computing is to increase available computation power for faster application processing and problem solving.

The advantages of parallel computing are that **computers can execute code more efficiently, which can save time and money by sorting through “big data” faster**

than ever. Parallel programming can also solve more complex problems, bringing more resources to the table.

DISTRIBUTED CCOMPUTING :

Distributed computing (or distributed processing) is **the technique of linking together multiple computer servers over a network into a cluster, to share data and to coordinate processing power**.

Examples of distributed systems / applications of distributed computing : **Intranets, Internet, WWW, email**. Telecommunication networks: Telephone networks and Cellular networks.

https://computernetworktopology.com/distributed-computing/

ADVANTAGES:

- **Reliability, high fault tolerance:** A system crash on one server does not affect other servers.
- **Scalability:** In distributed computing systems you can add more machines as needed.
- **Flexibility:** It makes it easy to install, implement and debug new services.
- **Fast calculation speed:** A distributed computer system can have the computing power of multiple computers, making it faster than other systems.
- **Openness:** Since it is an open system, it can be accessed both locally and remotely.
- **High performance:** Compared to centralized computer network clusters, it can provide higher performance and better cost performance.

Disadvantages of Distributed Computing

- **Difficult troubleshooting:** Troubleshooting and diagnostics are more difficult due to distribution across multiple servers.
- **Less software support:** Less software support is a major drawback of distributed computer systems.
- **High network infrastructure costs:** Network basic setup issues, including transmission, high load, and loss of information.
- **Security issues:** The characteristics of open systems make data security and sharing risks in distributed computer systems.

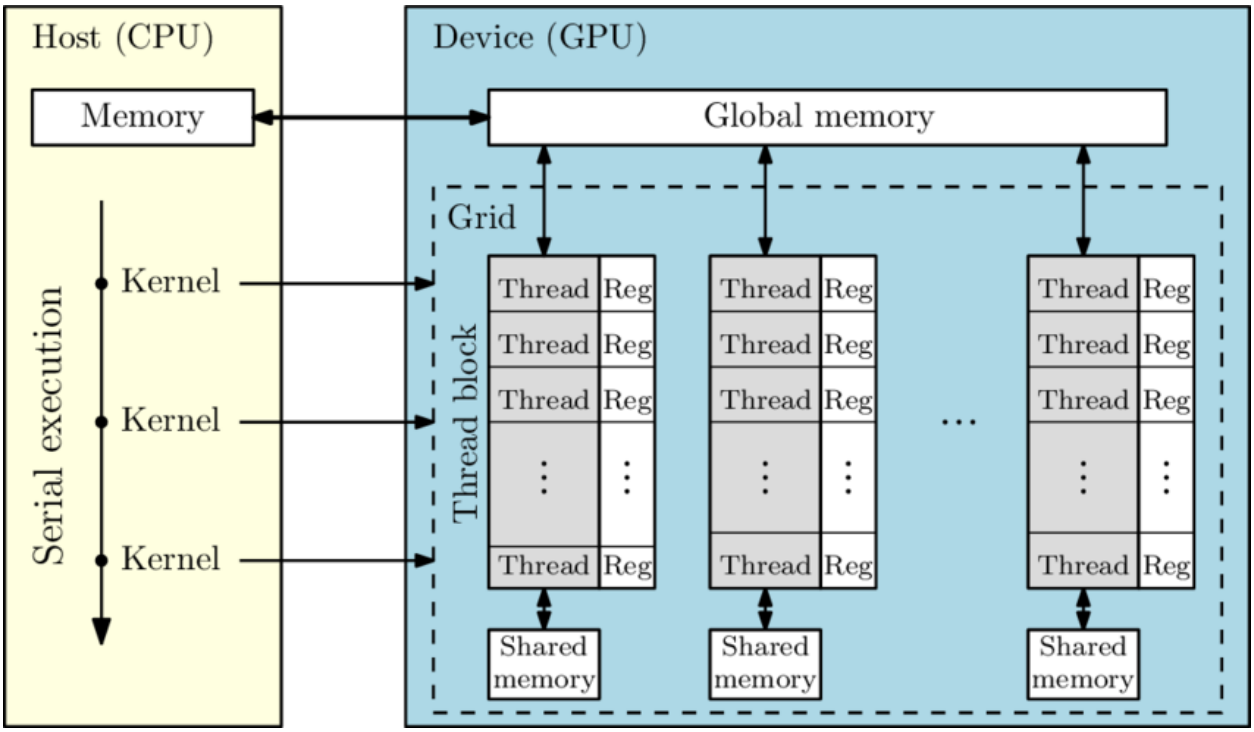
Q : briefly describe the working of array processor with the help of diagram,

https://edurev.in/studytube/Array-Processing-Computer-Organization-and-Archite/76f525eb-f1bb-4a9a-90a2-d4363dcab8d0_t

we learned that vector processing is **the process of using vectors to store a large number of variables for high-intensity data processing**. Weather forecasting, human genome mapping and GIS data are some examples

Array processor: Instruction operates on multiple data. elements at the same time. ❑ Vector processor: **Instruction operates on multiple data**. **elements in consecutive time steps**.

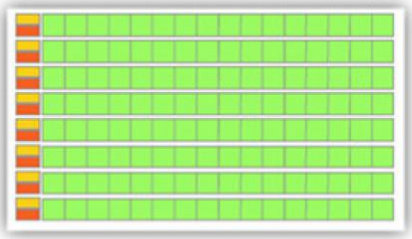
What does GPU stand for? Graphics processing unit, a specialized processor originally designed to **accelerate graphics rendering**. GPUs can process many pieces of data simultaneously, making them useful for machine learning, video editing, and gaming applications



CPU

* Low compute density
* Complex control logic
* Large caches (L1$/L2$, etc.)
* Optimized for serial operations
  - Fewer execution units (ALUs)
  - Higher clock speeds
* Shallow pipelines (<30 stages)
* Low Latency Tolerance
* Newer CPUs have more parallelism

GPU



* High compute density
* High Computations per Memory Access
* Built for parallel operations
  - Many parallel execution units (ALUs)
  - Graphics is the best known case of parallelism
* Deep pipelines (hundreds of stages)
* High Throughput
* High Latency Tolerance
* Newer GPUs:
  - Better flow control logic (becoming more CPU-like)
  - Scatter/Gather Memory Access
  - Don't have one-way pipelines anymore