Multiprocessors

A multiprocessor is a single computer that has multiple processors. It is possible that the processors in the multiprocessor system can communicate and cooperate at various levels of solving a given problem. The communications between the processors take place by sending messages from one processor to another, or by sharing a common memory.

Types of Multiprocessors

The following are the types of multiprocessors.

1. Symmetric Multiprocessors

Each processor in these systems runs a similar version of the operating system and communicates with the others. There is no master-slave connection between the processors because they are **all peer-to-peer.**

The Encore version of Unix for the Multimax Computer is a symmetric multiprocessing system.

2. Asymmetric Multiprocessors (Master-slave)

In an asymmetric system, each CPU is allocated a certain task. A master processor is in charge of giving all of the other processors' instructions. **An asymmetric multiprocessor system has a master-slave relationship.**

Asymmetric multiprocessors were the only type of multiprocessor available before the advent of symmetric multiprocessors. This is also the more affordable alternative right now.

Advantages of Multiprocessor Systems

Here is the list of the potential advantages of multiprocessor systems.

1. More reliable Systems

Even if one processor fails in a multiprocessor system, the system will not come to a halt. The ability to work seamlessly even in the case of hardware failure can be defined as graceful degradation. If one of the five processors in a multiprocessor system fails, the remaining four processors continue to work. As a result, rather than coming to a complete stop, the machine slows down.

2. Increasing Throughout

The system's throughput increases as several processors work together, indicating the number of processes done per unit of time increases. The throughput increases by a factor of N when there are N processors.

3. More Economic Systems

Since multiprocessor systems share data storage, peripheral devices, power supply, and other resources, they are less expensive in the long run than single-processor systems. If several processes share data, it is preferable to schedule them on multiprocessor systems with shared data rather than separate computer systems with different copies of the data.

Characteristics of Multiprocessor

The following are the important characteristics of multiprocessors.

1. Parallel Processing: This requires the use of many processors at the same time. These processors are designed to do a particular task using a single architecture. Processors are generally identical, and they operate together to create the effect that the users are the only individuals who are using the system. In reality, several others are trying to use the system in the first place.

2. Distributed Computing: In addition to parallel computing, this distributed processing requires the use of a processor network. Each processor in this network can be thought of as a standalone computer with the ability to solve problems. These processors are diverse, and each one is typically assigned to a separate job.

3. Supercomputing: This entails using the quickest machines to address large, computationally difficult issues. Supercomputers used to be vector computers, but nowadays, most people accept vector or parallel computing.

4. Pipelining: Besides supercomputing, this is a method that divides a task into multiple subtasks that must be completed in a specified order. Each subtask is aided by the functional units. The devices are connected serially, and they all work at the same time.

5. Vector Computing: This is a method that divides a task into multiple subtasks that must be completed in a specified order. Each subtask is aided by the functional units. The devices are connected serially, and they all work at the same time.

6. Systolic: **Pipelining is similar, but the units are not organized linearly.** Systolic steps are often tiny and numerous, and they are conducted in lockstep. This is more commonly used in specialized hardware like image or signal processors.

Interconnection structures

The processors must be able to share a set of main memory modules & I/O devices in a multiprocessor system. This sharing capability can be provided through interconnection structures. The interconnection structure that are commonly used can be given as follows –

- Time-shared / Common Bus
- Cross bar Switch
- Multiport Memory
- Hypercube System

1. Time-shared / Common Bus (Interconnection structure in Multiprocessor System): In a multiprocessor system, the time shared bus interconnection provides a common communication path connecting all the functional units like processor, I/O processor, memory unit etc. The figure below shows the multiple processors with common communication path (single bus).

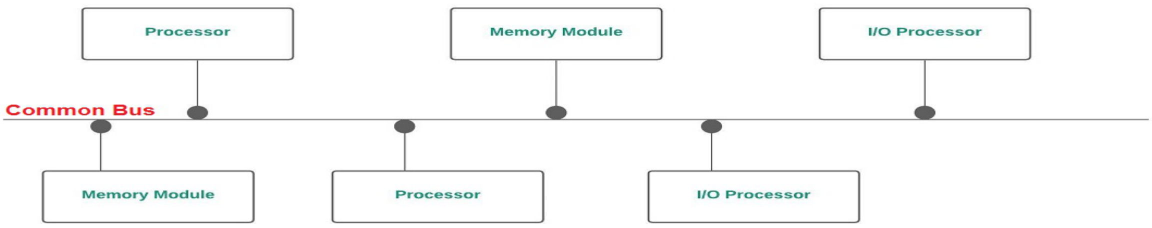


Fig: Single-Bus Multiprocessor Organization

To communicate with any functional unit, processor needs the bus to transfer the data. To do so, the processor first need to see that whether the bus is available / not by checking the status of the bus. If the bus is used by some other functional unit, the status is busy, else free.

A processor can use bus only when the bus is free. The sender processor puts the address of the destination on the bus & the destination unit identifies it. In order to communicate with any functional unit, a command is issued to tell that unit, what work is to be done. The other processors at that time will be either busy in internal operations or will sit free, waiting to get bus.
We can use a bus controller to resolve conflicts, if any. (Bus controller can set priority of different functional units)
This Single-Bus Multiprocessor Organization is easiest to reconfigure & is simple. This interconnection structure contains only passive elements. The bus interfaces of sender & receiver units control the transfer operation here.
To decide the access to common bus without conflicts, methods such as static & fixed priorities, First-In-Out (FIFO) queues & daisy chains can be used.

Advantages –

- Inexpensive as no extra hardware is required such as switch.
- Simple & easy to configure as the functional units are directly connected to the bus.

Disadvantages –

Major fight with this kind of configuration is that if malfunctioning occurs in any of the bus interface circuits, complete system will fail.

Decreased throughput —
At a time, only one processor can communicate with any other functional unit.

Increased arbitration logic —
As the number of processors & memory unit increases, the bus contention problem increases.

To solve the above disadvantages, we can use two uni-directional buses as :

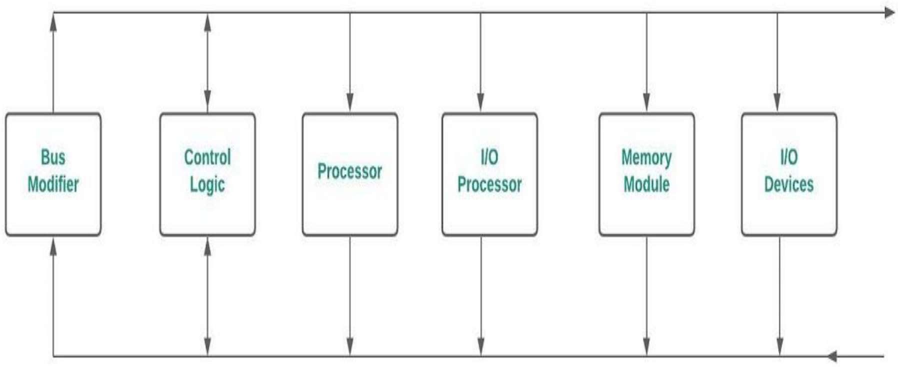


Fig: Multiprocessor System with unidirectional buses

Both the buses are required in a single transfer operation. Here, the system complexity is increased & the reliability is decreased, The solution is to use multiple bi-directional buses.

Multiple bi-directional buses:

The multiple bi-directional buses means that in the system there are multiple buses that are bi-directional. It permits simultaneous transfers as many as buses are available. But here also the complexity of the system is increased.

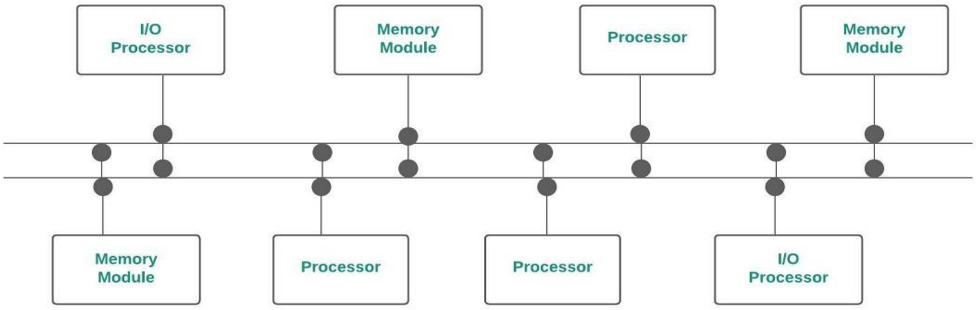


Fig: Multiple Bi-Directional Multiprocessor System

Apart from the organization, there are many factors affecting the performance of bus. They are –

- Number of active devices on the bus.
- Data width
- Error Detection method

- ➢ Synchronization of data transfer etc.

Advantages of Multiple bi-directional buses –

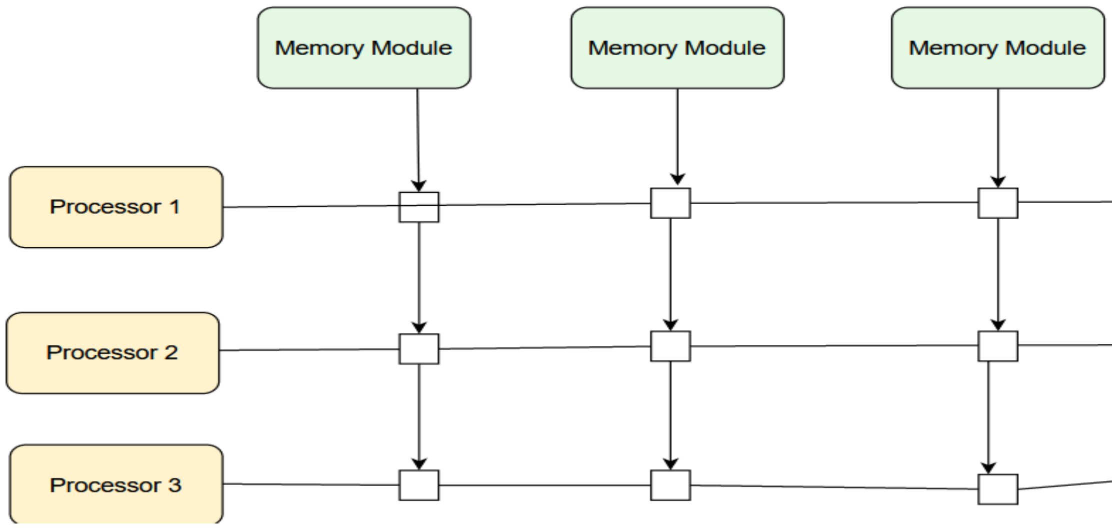
- ➢ Lowest cost for hardware as no extra device is needed such as switch.
- ➢ Modifying the hardware system configuration is easy.
- ➢ Less complex when compared to other interconnection schemes as there are only 2 buses & all the components are connected via that buses.

Disadvantages of Multiple bi-directional buses –

- ➢ System Expansion will degrade the performance because as the number of functional unit increases, more communication is required but at a time only 1 transfer can happen via 1 bus.
- ➢ Overall system capacity limits the transfer rate & If bus fails, whole system will fail.
- ➢ Suitable for small systems only.

2. Crossbar Switch:

A point is reached at which there is a separate path available for each memory module, if the number of buses in common bus system is increased. Crossbar Switch (for multiprocessors) provides separate path for each module.



3. Multiport Memory:

In Multiport Memory system, the control, switching & priority arbitration logic are distributed throughout the crossbar switch matrix which is distributed at the interfaces to the memory modules.

4. Hypercube Interconnection:

This is binary n-cube architecture. Here we can connect 2n processors and each of the processor here forms a node of the cube. A node can be memory module, I/O interface also, not necessarily processor. The processor at a node has communication path that is direct goes to n other nodes (total 2n nodes). There are total 2n distinct n-bit binary addresses.