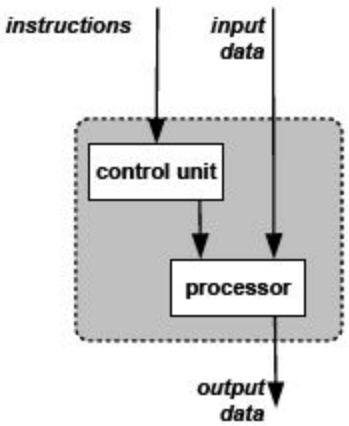
Software Engineering for Parallel and Distributed Systems
Exercise Sheet 1

1. Exercise 1.3, Flynn's Taxonomy (5 Points)
In the lecture, Flynn's taxonomy was introduced.Describe the characteristics of and an example
for each of the four systems.

ANSWER:
In general a stream is a sequence of instructions or data on which
a computer operates.
In Flynn's taxonomy, there are four possibilities:
a. SISD
b. SIMD
c. MISD
d. MIMD

a. SISD
>In a SISD system, one stream of instructions processes a single stream of data.
>This is the common von Neumann model used in virtually all
single-processor computers.

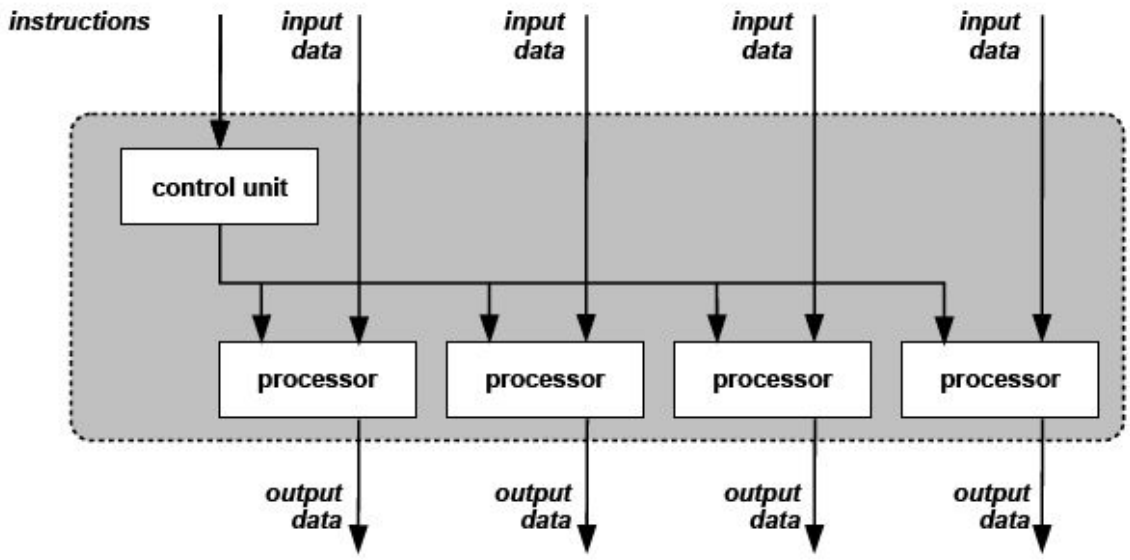
SISD architecture



b. SIMD
>In a SIMD system, a single instruction stream is concurrently

broadcast to multiple processors, each with its own data stream.
>The original systems from Thinking Machines and MasPar can be classified as SIMD.
>Vector processors, which operate on vector data in a pipelined fashion, can also be categorized as SIMD.
Exploiting this parallelism is usually done by the compiler.
Example: The NEC vector computing systems at CAU's computing center.

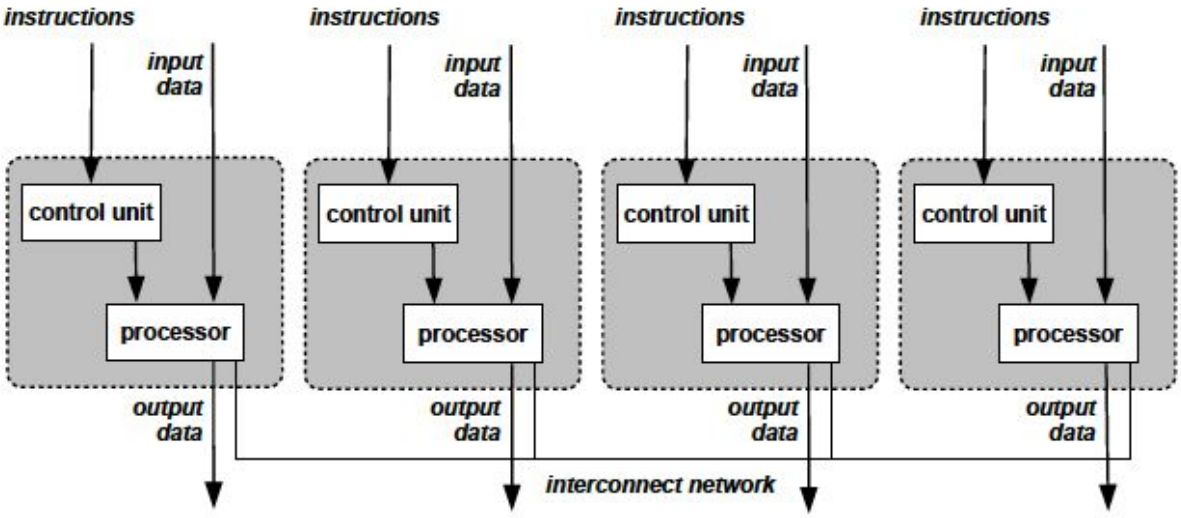
SIMD architecture



c.MISD
>No well-known systems t this designation (mentioned for the sake of completeness).

d.MIMD
>In a MIMD system, each processing unit has its own stream of instructions operating on its own data.
>This architecture is the most general of the architectures in that each of the other cases can be mapped onto the MIMD architecture.
>The vast majority of modern parallel systems fit into this Category.

MIMD architecture



Exercise 1.4, Scalability Limitation of MIMD Systems (5 Points)
Describe the main scalability limitation for each of the following MIMD systems. Consider both
hardware and software aspects. Justify your answer.
• Symmetric multiprocessors
• Clusters

ANSWER:
A.
Limitations :

Software Issues.
1.Scalability beyond thirty-two processors is difficult.
2. Shared memory model is less flexible than the distributed memory model.
3.Deadlock in message passing.
4.Need to physically copy data between processes.
5.Performance limited by bus bandwidth.
6.Each processor should have local cache
- To reduce number of bus accesses.
- Can lead to problems with cache coherence.

7.Synchronized access to share data in memory needed. Synchronising constructs (semaphores, conditional critical regions, monitors) result in non deterministic behaviour which can lead programming errors that are difficult to discover.
8.Lack of scalability due to (memory) contention problem.

Hardware Issues.

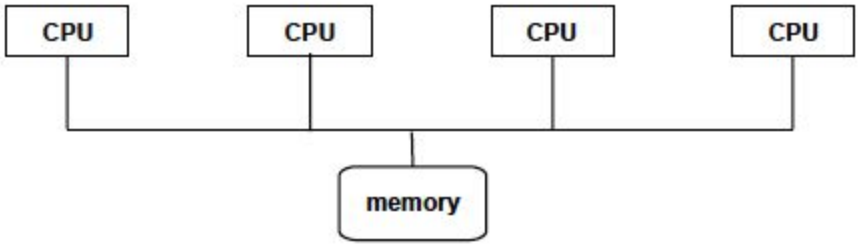
1.An SMP OS manages processor resources so that the user perceives a single system.
2.It should appear as a single-processor multiprogramming system.
3.In both SMP and uniprocessor, multiple processes may be active at one time. OS is responsible for scheduling their execution and allocating resources.

Symmetric multiprocessors

>In symmetric multiprocessors all processors share a connection to a common memory and access all memory locations at equal speeds.
>SMP systems are arguably the easiest parallel systems to program because programmers do not need to distribute data structures among processors.
>Because increasing the number of processors increases contention for the memory, the processor/memory bandwidth is typically a limiting factor.
>Thus, SMP systems do not scale well and are limited to small numbers of processors.

Symmetric multiprocessor systems with shared memory.

>In a shared-memory system, all processes share a single address space and communicate with each other by writing and reading shared variables.
>One class of shared-memory system is called SMPs (symmetric multiprocessors):



Clusters

>A set of computers connected over a high-bandwidth local area network, and used as a parallel computer.
>Clusters are called Loosely Coupled MIMD.
>A group of interconnected stand-alone computers.
>Work together as a unified resource.
>Give illusion of being one machine.
>Each computer called a node A node can also be a multiprocessor itself, such as an SMP.

>Two types for message passing for communication between nodes.

a. NOW - Network of Workstations, homogeneous cluster.
b. GRID -Computers connected over a wide area network.

Advantages of Clusters.
>Absolute scalability - Cluster with a very large number of nodes is possible.
>Incremental scalability - A user can start with a small system and expand it as needed, without having to go through a major upgrade.
>High availability - Fault tolerance by nature.
>Superior price/performance - Can be built with cheap commodity nodes.
>Supercomputing-class commodity components are available.