

CS422

Computer Architecture

Spring 2004

Lecture 23, 26 Mar 2004

Bhaskaran Raman
Department of CSE
IIT Kanpur

<http://web.cse.iitk.ac.in/~cs422/index.html>

Topics Remaining

- *HW2 handed out today/tomorrow*
- Multiprocessors: 4 lectures
- Inter-connection networks: 1 lecture
- I/O: 2 lectures
- *Review: 1 or 2 lectures*
- *Take-home part of end-sem handed out*
- Special topics:
 - Vector processors
 - Power optimization issues

Topic for Today's Lecture

- Multiprocessing
-
- *Scribe for today?*

Why Multiprocessors?

Motivation:

Go beyond the performance offered by a single processor

Without requiring specialized processors

Without the complexity of too much multiple issue

Opportunity:

Software available

Parallel programs

Multi-programmed machines

Multiprocessors: The SIMD Model

- SISD: Single Instruction stream, Single Data stream
 - Uniprocessor
 - This is the view at the ISA level
 - Tomasulo uncovers data stream parallelism
- SIMD: Single Instruction stream, Multiple Data streams
 - ISA makes data parallelism explicit
 - Special SIMD instructions
 - Same instruction goes to multiple functional units, but acts on different data

SIMD Drawbacks

SIMD useful for loop-level parallelism

Model is too inflexible to accommodate parallel programs as well as multi-programmed environments

Cannot take advantage of uniprocessor performance growth

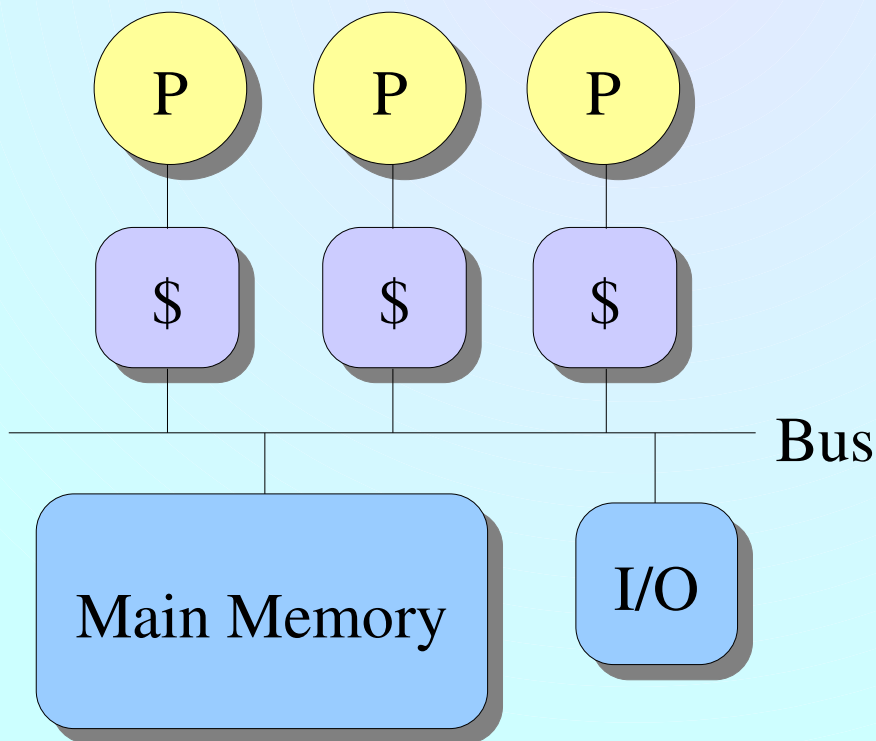
SIMD architecture usually used in special purpose designs

Signal or image processing

Multiprocessors: The MIMD Model

- MIMD: Multiple Instruction streams, Multiple Data streams
 - Each processor fetches its own instruction and data
- Advantages:
 - Flexibility: parallel programs, or multi-programmed OS, or both
 - Built using off-the-shelf uniprocessors

MIMD: The Centralized Shared-Memory Model

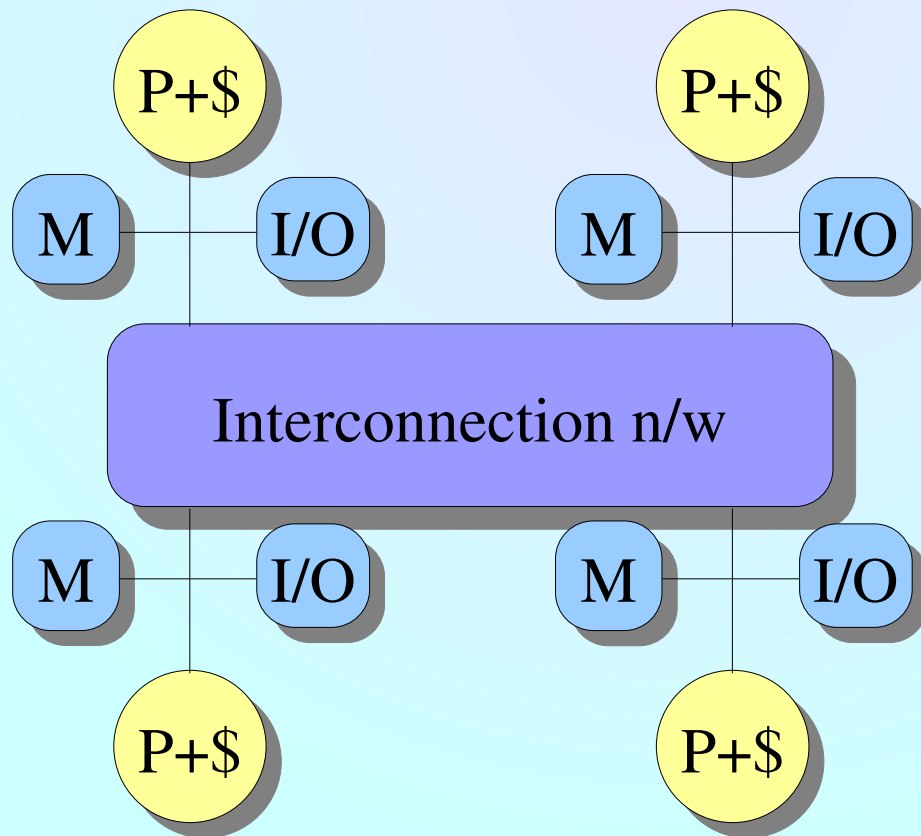


Single bus connects a shared memory to all processors

Also called *Uniform Memory Access (UMA)* machine

Disadvantage: cannot scale very well, especially with fast processors (more memory bandwidth required)

MIMD: Physically Distributed Memory



Independent memory for each processor

High-bandwidth interconnection

Adv: cost-effective memory bandwidth scaling

Adv: lesser latency for local access

Disadv: communication of data between nodes

Communication Models with Physically Distributed Memory

- Distributed Shared Memory (DSM)
 - Memory *address space* is the same across nodes
 - Also called scalable shared memory
 - Also called *NUMA: non-uniform memory access*
 - Communication is implicit via load/store
- Multicomputer, or Message Passing Machine
 - Separate private address spaces for each node
 - Communication is explicit, through messages
 - Synchronous, or asynchronous
 - Std. Message Passing Interface (MPI) possible

Multiprocessing: Classification

