Course Overview (Tentative)

- This course vs. AOS, ParAlg, ParProg
- Lectures: mostly me, some by other profs, other profs may sit in sometimes
- Textbooks:
  - Hord: required, we'll cover at least half
  - Foster: optional, online, also cover half
  - Pfister: may not use
- Evaluations:
  - Roughly every 2 weeks
  - Paper assignments, programming, etc.
  - Probably no exams (will decide by 9/8)
- Feedback on course topic, style, content, etc. would be appreciated

Supercomputing

- Supercomputers have higher than average speed and capacity
- US government involvement
  - High Performance Computing and Communications Program (HPCC) — federal agencies, industry, academia
  - DOE Accelerated Strategic Computing Initiative (ASCI)
    - Los Alamos National Labs & clusters of SMPs
    - One program three labs (LANL, ?, ?)
- Grand Challenge problems (1993)
  - Magnetic recording technology, high speed civil transports, catalysts, ocean modeling, digital anatomy, air pollution, Venus imaging

Example Problem: Weather Prediction (in "Parallel Programming", byWilkinson)

- Atmosphere divided into 3D "cells", computations involve temperature, pressure, humidity, wind speed and direction, etc., are computed at time intervals, using info from previous interval
- Some numbers:
  - Suppose cell = 1 mile^3, atmosphere modeled to height of 10 miles, gives 5 x 10^8 cells
  - Suppose each calculation = 200 floating point operations, in one time step need 10^{11} fp ops
  - Suppose forecast weather over 10 days, at 10-minute intervals, would be 10^4 time steps and 10^{15} fp ops
  - Suppose computer runs at 100 Mflops (10^8 floating point operations per second), calcs would take 10^7 seconds = 100 days

Two Taxonomies for Classifying Computer Systems

- Michael Flynn (1966)
  - SISD — single instruction, single data
  - SIMD — single instruction, multiple data
  - MISD — multiple instruction, single data
  - MIMD — multiple instruction, multiple data
- More recent (Stallings, 1993)

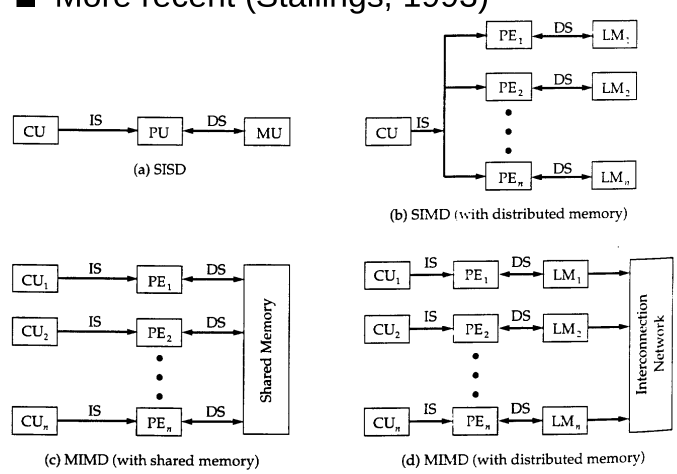
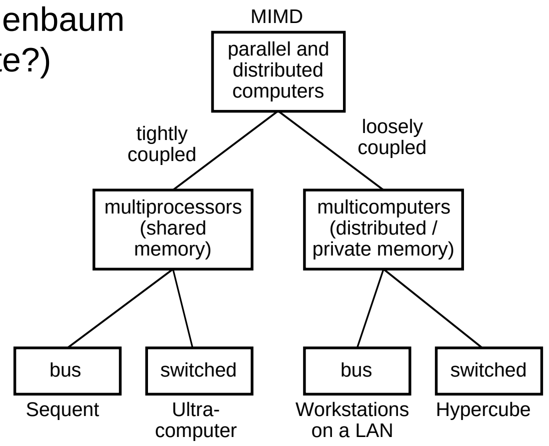


FIGURE 16.16. Alternative Computer Organizations

Classification of MIMD Architectures

- Tanenbaum (date?)



- Tightly coupled ≈ *parallel processing*
  - Processors share clock and memory, run one OS, communicate frequently
- Loosely coupled ≈ *distributed computing*
  - Each processor has its own memory, runs its own OS (?), communicates infrequently

MIMD vs. SIMD (Hord 2.2)

- MIMD:
  - Relatively few powerful processors
  - Control level parallelism that assigns a processor to a unit of code
  - Typically either distributed memory or shared memory; can have memory contention
  - Needs good task scheduling for efficiency
  - Each processor runs its own instruction sequence
  - Each processor works on a different part of the problem
  - Each processor communicates data to other parts
  - Processors may have to wait for other processors or for access to data

MIMD vs. SIMD (Hord 2.2) (cont.)

- SIMD:
  - Many simple processors
  - Data level parallelism that assigns a processor to a unit of data
  - Typically distributed memory; can have data communication problems
  - Needs good processor utilization for efficiency
  - All processors are given the same instruction
  - Each processor operates on different data
  - Processors may “sit out” a sequence of instructions