

## Why Multiprocessors?

Moore's Law predicted a doubling of processor performance every couple of years

- true until about 2000

Limits on the performance of a single processor: what are they?

## Why Multiprocessors

Utilizes coarser granularities than ILP

Lots of workload opportunity

- Scientific computing/supercomputing
  - Examples: weather simulation, aerodynamics, protein folding
  - Each processor computes for a part of the grid
- Server workloads
  - Example: airline reservation database
  - Many concurrent updates, searches, lookups, queries
  - Processors handle different requests
- Media workloads
  - Processors compress/decompress different parts of image/frames
- Desktop workloads ...
- Gaming workloads ...

What would you do with a billion transistors on a chip? Or more?

## Multiprocessors

### Low-end

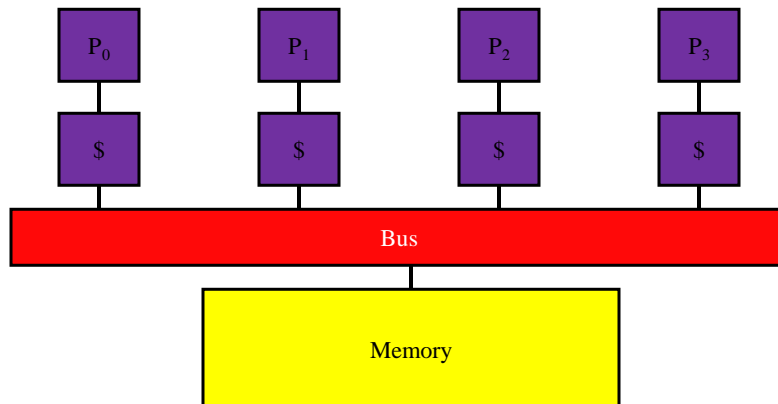
- bus-based
  - simple, but a bottleneck
  - broadcast-based cache coherency protocol
- physically centralized memory
- uniform memory access (UMA machine)
- today's small CMPs:  
Intel Core i3, i5, i7 (2-6 cores), AMD Opteron "Bulldozer" (4-16 cores), Sun SPARC T4 (8 cores per processor, 4 processors per system), ARM Cortex A5 (2 cores), Nvidia Tegra 3 (4 cores)

Spring 2012

CSE 471 - Multiprocessors

3

## Low-end MP



4

## Multiprocessors

### High-end

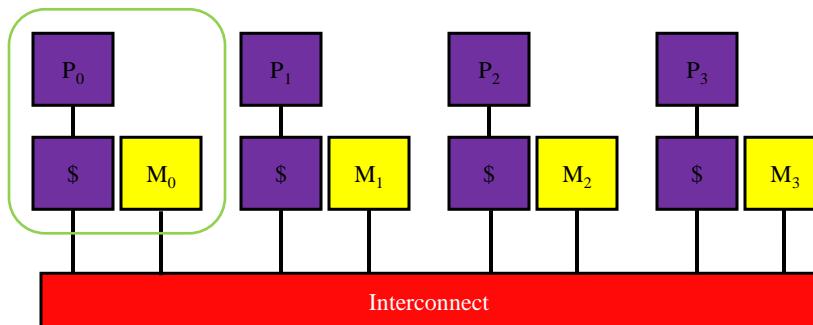
- multiple-path interconnect
  - higher bandwidth
  - longer memory latencies
  - more scalable
  - point-to-point cache coherency protocol
- physically distributed memory
- non-uniform memory access (NUMA machine)
- could have processor clusters
- today's large MPs:  
SGI UV (256 10-core Xeon processors, 2D torus), Cray XE6 (1M Opteron 6200 cores), IBM BlueGene/Q (100K 16-core PowerPCs, 5D torus), Fujitsu K Computer (44K 16-core SPARC64s)

Spring 2012

CSE 471 - Multiprocessors

5

## High-end MP



6

## Comparison of Issue Capabilities

