



Lecture 2

A General Discussion on Parallelism

John Cavazos

Dept of Computer & Information Sciences

University of Delaware

www.cis.udel.edu/~cavazos/cisc879

CISC 879 : Advanced Parallel Programming



Lecture 2: Overview

- Flynn's Taxonomy of Architectures
- Types of Parallelism
- Parallel Programming Models
- Commercial Multicore Architectures

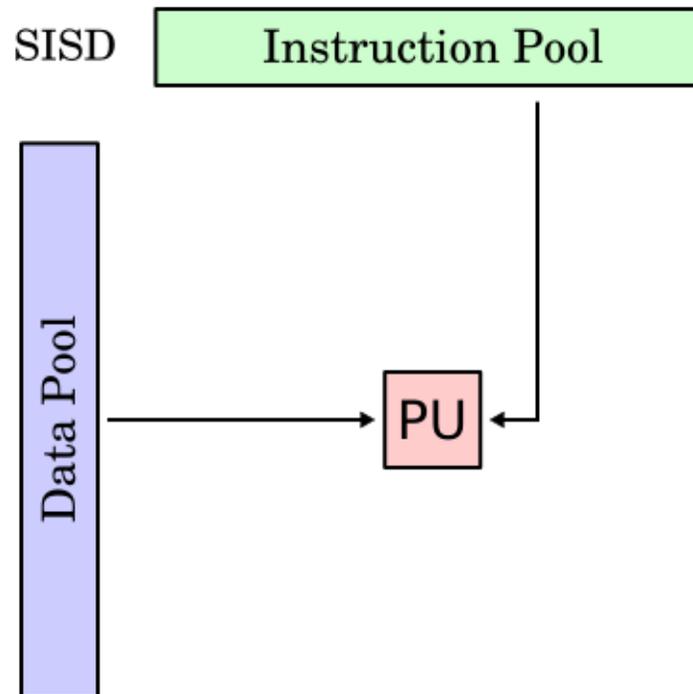


Flynn's Taxonomy of Arch.

- SISD - Single Instruction/Single Data
- SIMD - Single Instruction/Multiple Data
- MISD - Multiple Instruction/Single Data
- MIMD - Multiple Instruction/Multiple Data



Single Instruction/Single Data

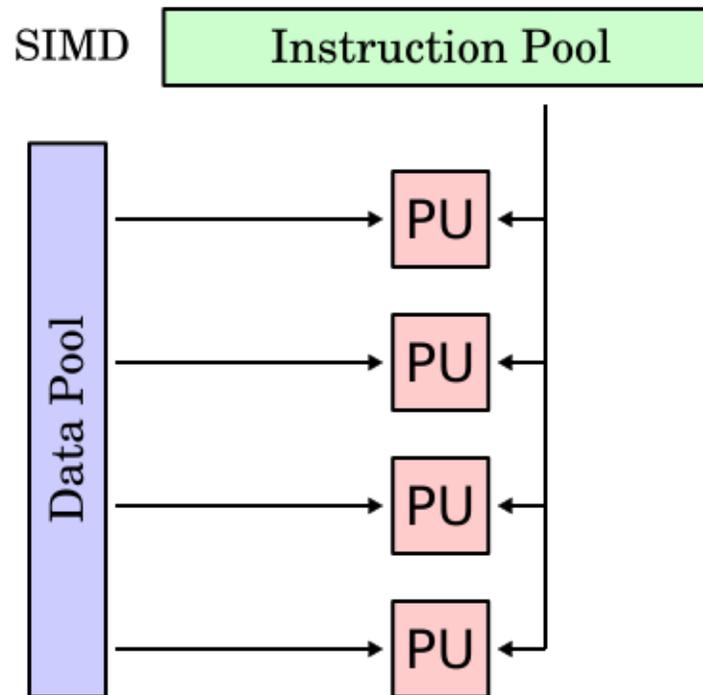


The typical machine you're used to (before multicores).

Slide Source: Wikipedia, Flynn's Taxonomy



Single Instruction/Multiple Data



Processors that execute same instruction on multiple pieces of data.

Slide Source: Wikipedia, Flynn's Taxonomy

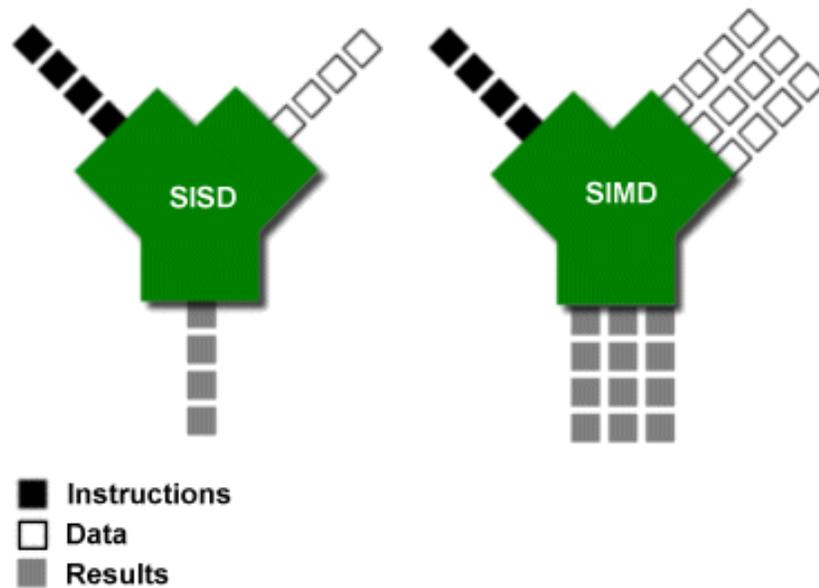


Single Instruction/Multiple Data

- Each core executes same instruction simultaneously
- Vector-style of programming
- Natural for graphics and scientific computing
- Good choice for massively multicore



SISD versus SIMD

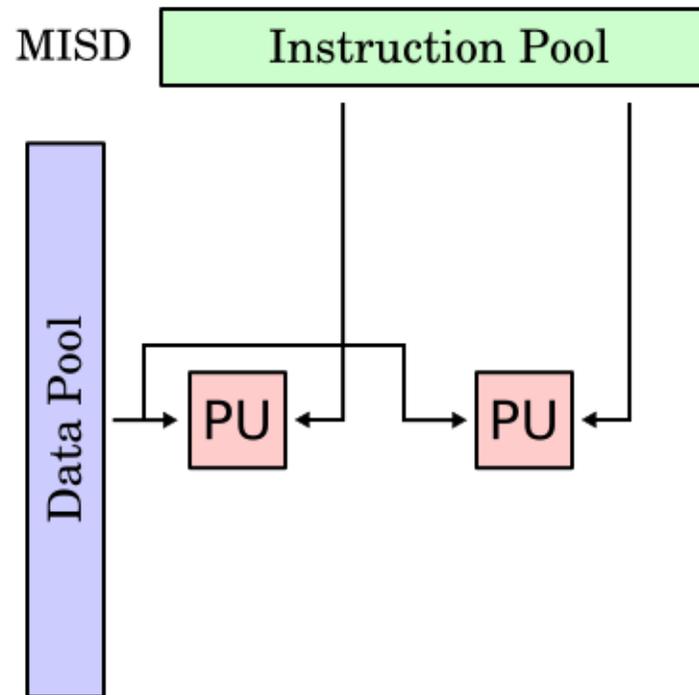


SIMD very often requires compiler intervention.

Slide Source: ars technica, Peakstream article



Multiple Instruction/Single Data

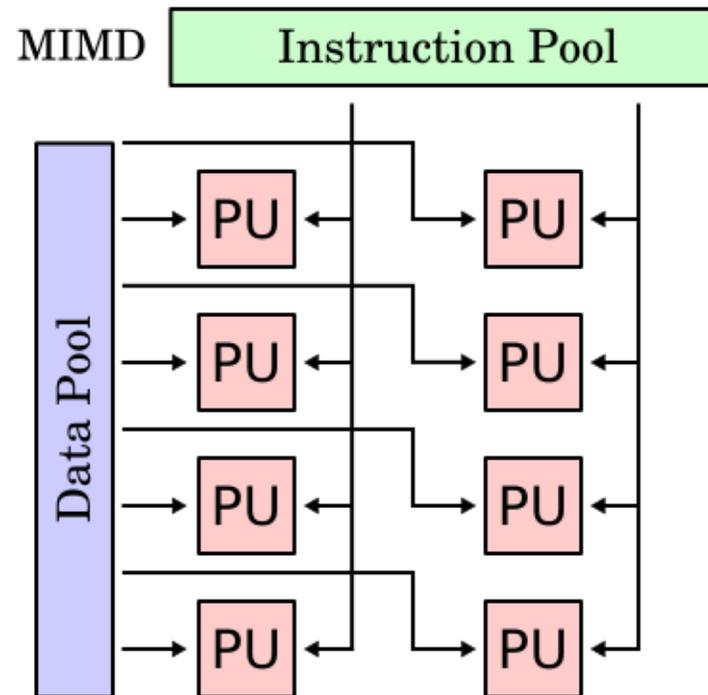


Only Theoretical Machine. None ever implemented.

Slide Source: Wikipedia, Flynn's Taxonomy



Multiple Instruction/Multiple Data



Many mainstream multicores fall into this category.

Slide Source: Wikipedia, Flynn's Taxonomy



Multiple Instruction/Multiple Data

- Each core works independently, simultaneously executing different instructions on different data
- Unique upper levels of cache and may have lower level of shared cache
- Cores can have SIMD-extensions
- Programmed with a variety of models (OpenMP, MPI, pthreads, etc.)

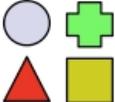


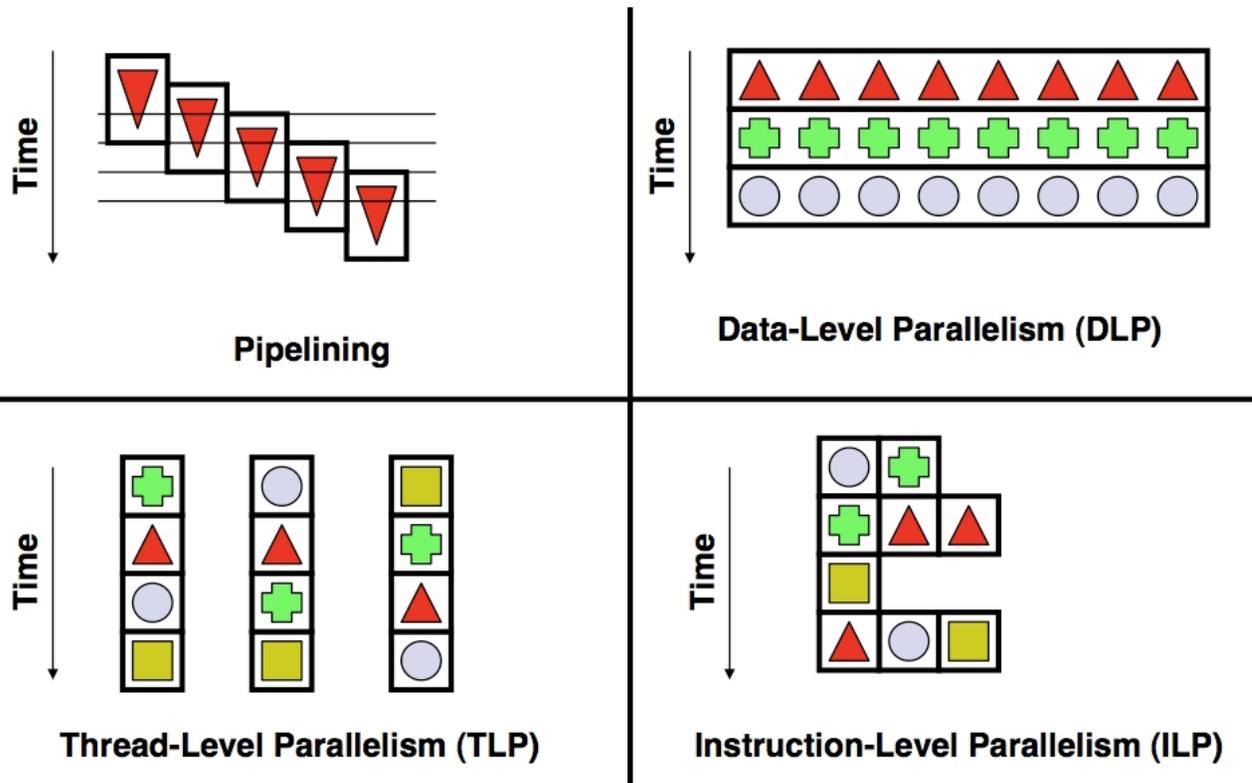
Lecture 2: Overview

- Flynn's Taxonomy of Architecture
- **Types of Parallelism**
- Parallel Programming Models
- Commercial Multicore Architectures



Types of Parallelism

Instructions: 



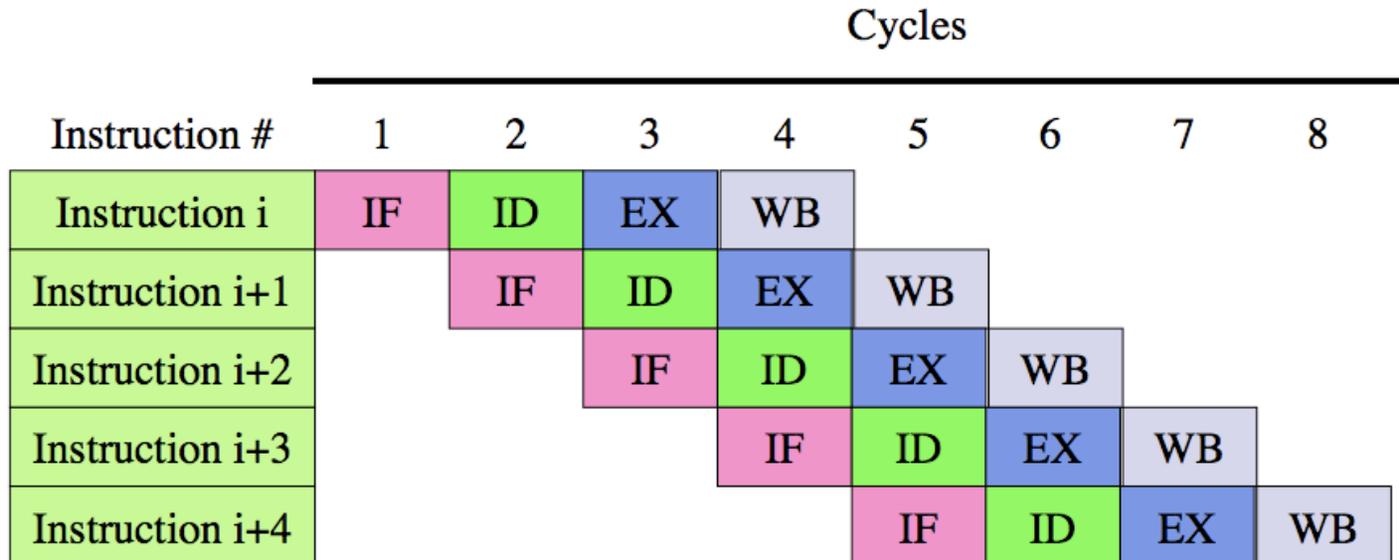
Slide Source: S. Amarasinghe, MIT 6189 IAP 2007



Pipelining

IF: Instruction fetch
EX : Execution

ID : Instruction decode
WB : Write back



Corresponds to SISD architecture.

Slide Source: S. Amarasinghe, MIT 6189 IAP 2007



Instruction-Level Parallelism

Cycles

| Instruction type | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|------------------|----|----|----|----|----|----|----|
| Integer | IF | ID | EX | WB | | | |
| Floating point | IF | ID | EX | WB | | | |
| Integer | | IF | ID | EX | WB | | |
| Floating point | | IF | ID | EX | WB | | |
| Integer | | | IF | ID | EX | WB | |
| Floating point | | | IF | ID | EX | WB | |
| Integer | | | | IF | ID | EX | WB |
| Floating point | | | | IF | ID | EX | WB |

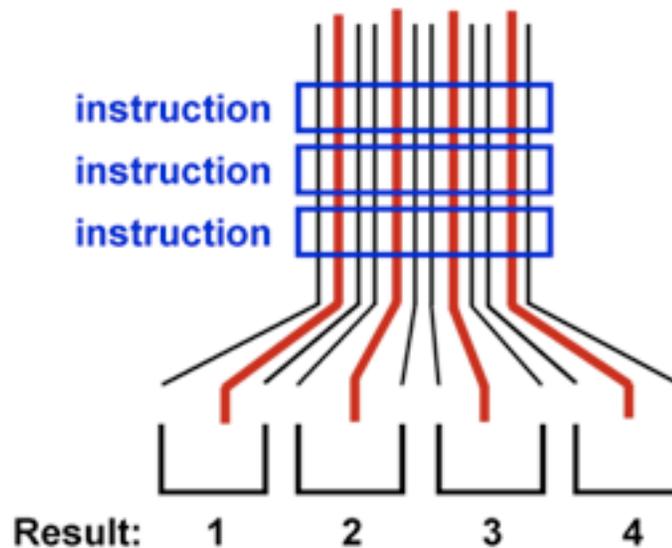
**Dual instruction issue superscalar model.
Again, corresponds to SISD architecture.**

Slide Source: S. Amarasinghe, MIT 6189 IAP 2007



Data-Level Parallelism

Data Stream or Array Elements



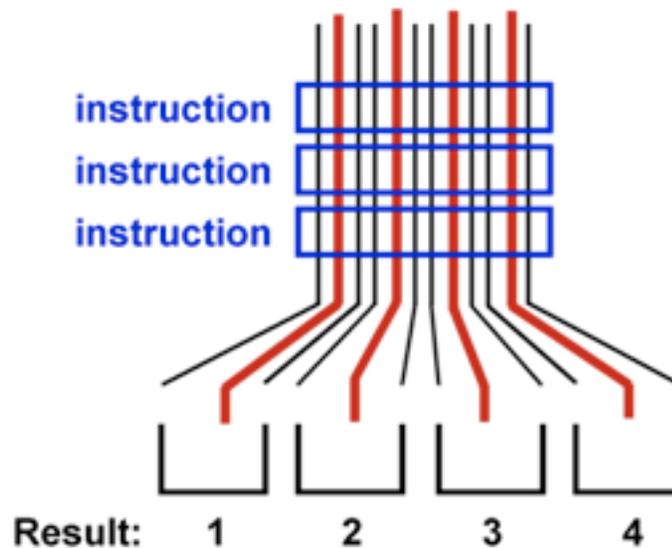
What architecture model from Flynn's Taxonomy does this correspond to?

Slide Source: Arch. of a Real-time Ray-Tracer, Intel



Data-Level Parallelism

Data Stream or Array Elements

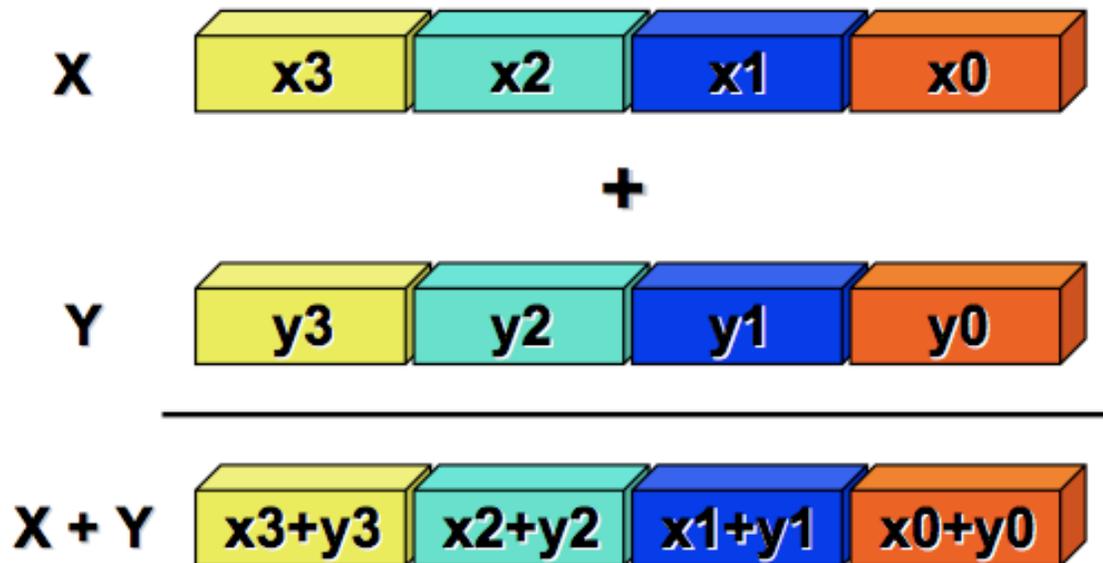


Corresponds to SIMD architecture.

Slide Source: Arch. of a Real-time Ray-Tracer, Intel



Data-Level Parallelism

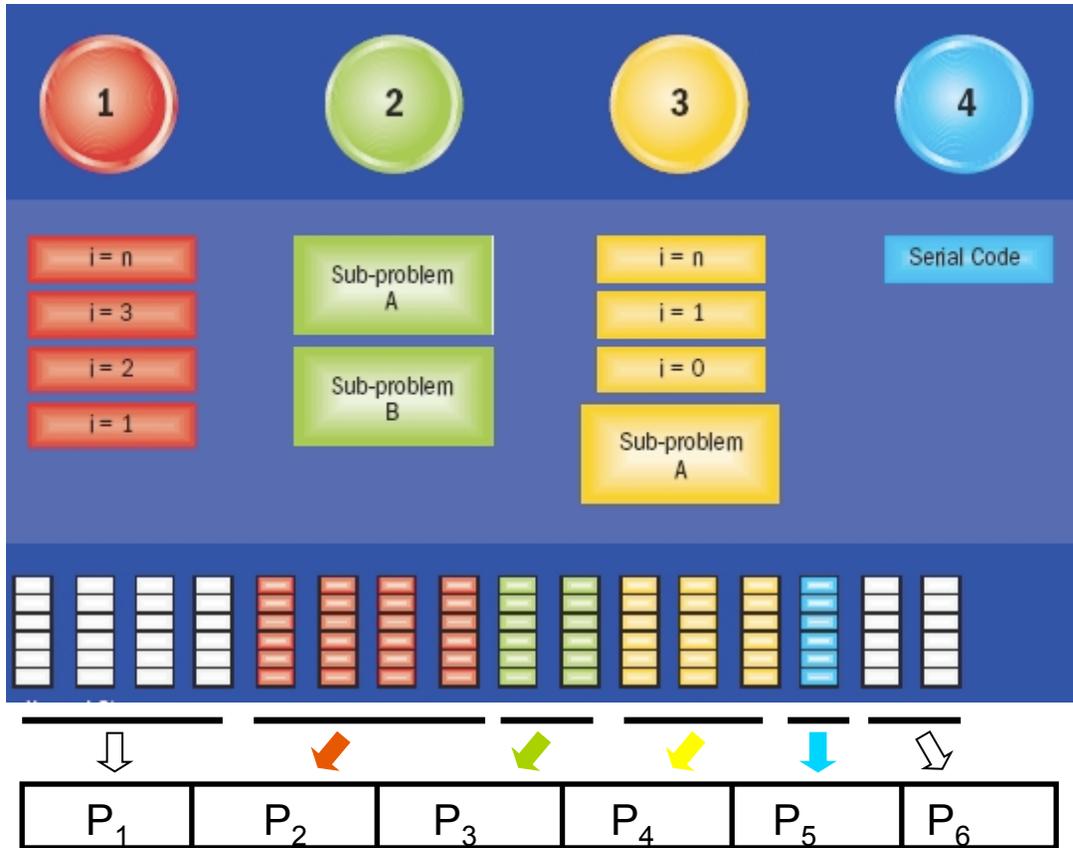


One operation (e.g., +) produces multiple results. X, Y, and result are arrays.

Slide Source: Klimovitski & Macri, Intel



Thread-Level Parallelism



Program partitioned into four threads.

Four threads each executed on separate cores.

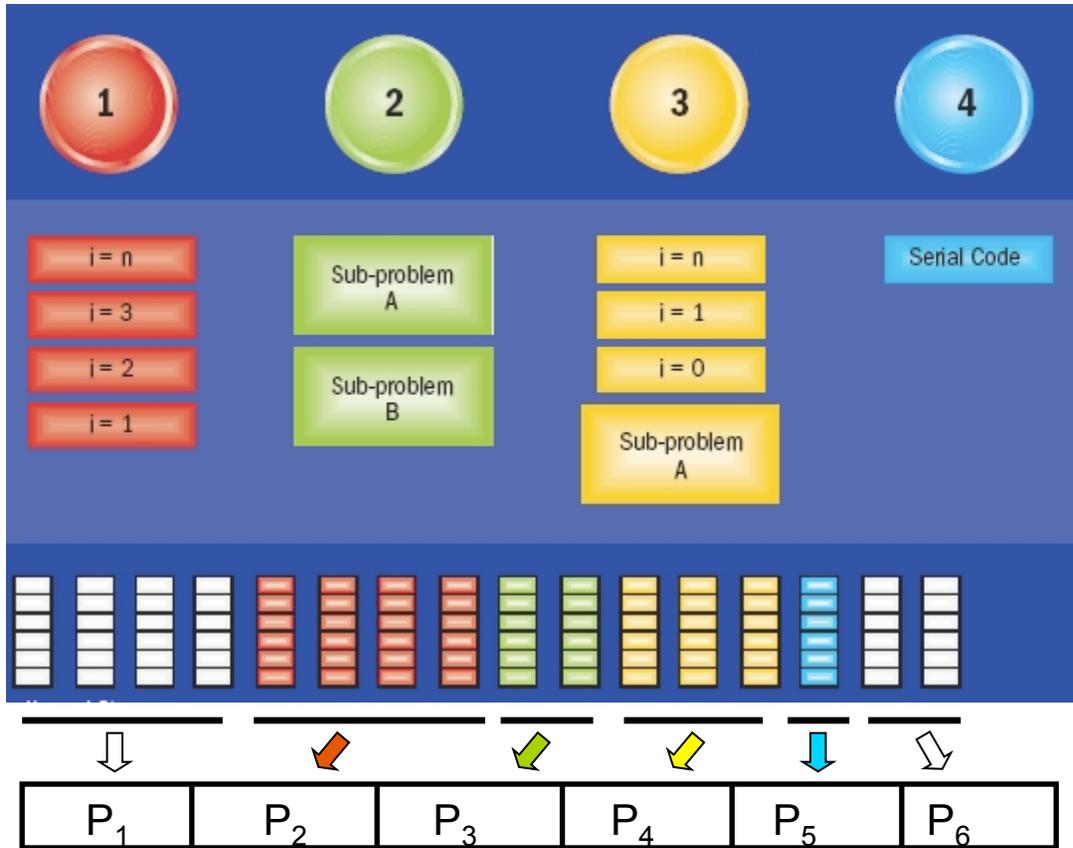
Multicore with 6 cores.

What architecture from Flynn's Taxonomy does this correspond to?

Slide Source: SciDAC Review, Threadstorm pic.



Thread-Level Parallelism



Program partitioned into four threads.

Four threads each executed on separate cores.

Multicore with 6 cores.

Corresponds to MIMD architecture.

Slide Source: SciDAC Review, Threadstorm pic.



Lecture 2: Overview

- Flynn's Taxonomy of Architecture
- Types of Parallelism
- **Parallel Programming Models**
- Commercial Multicore Architectures



Multicore Programming Models

- Message Passing Interface (MPI)
- OpenMP
- Threads
 - Pthreads
 - Cell threads
- Parallel Libraries
 - Intel's Thread Building Blocks (TBB)
 - Microsoft's Task Parallel Library
 - SWARM (GTech)
 - Charm++ (UIUC)
 - STAPL (Texas A&M)



GPU Programming Models

- CUDA (Nvidia)
 - C/C++ extensions
- Brook+ (AMD/ATI)
 - AMD-enhanced implementation of Brook
- Brook (Stanford)
 - Language extensions
- RapidMind platform
 - Library and language extensions
 - Works on multicores
 - Commercialization of Sh (Waterloo)

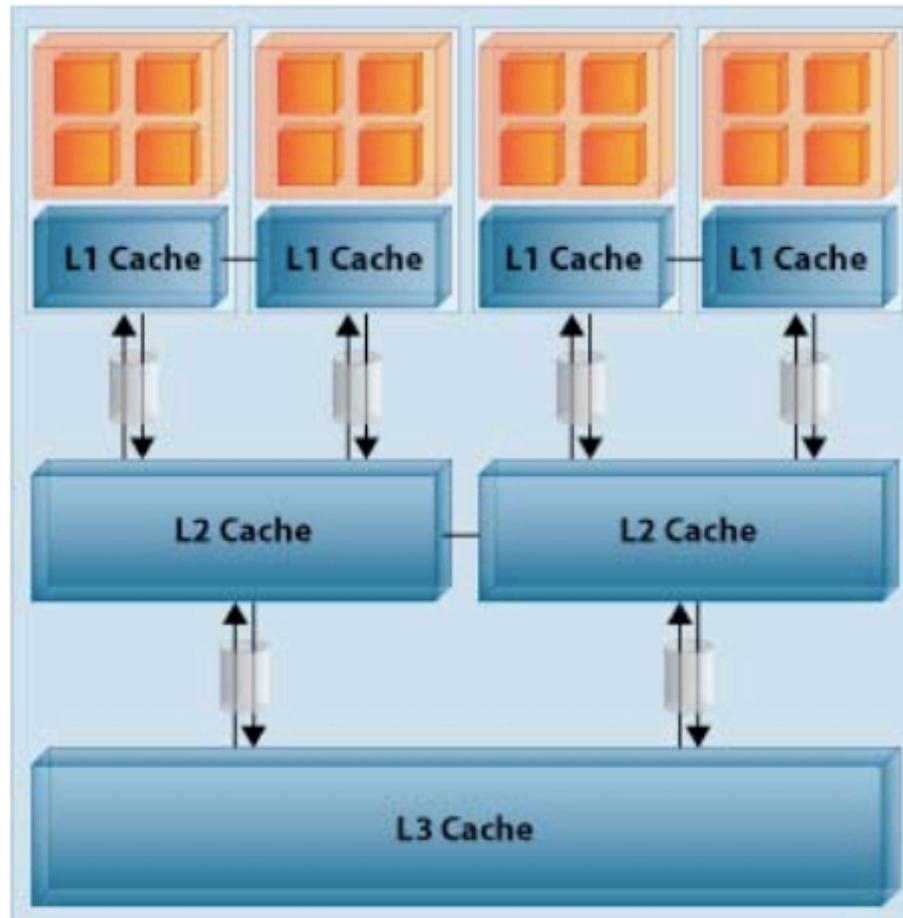


Lecture 2: Overview

- Flynn's Taxonomy of Architecture
- Types of Parallelism
- Parallel Programming Models
- **Commercial Multicore Architectures**



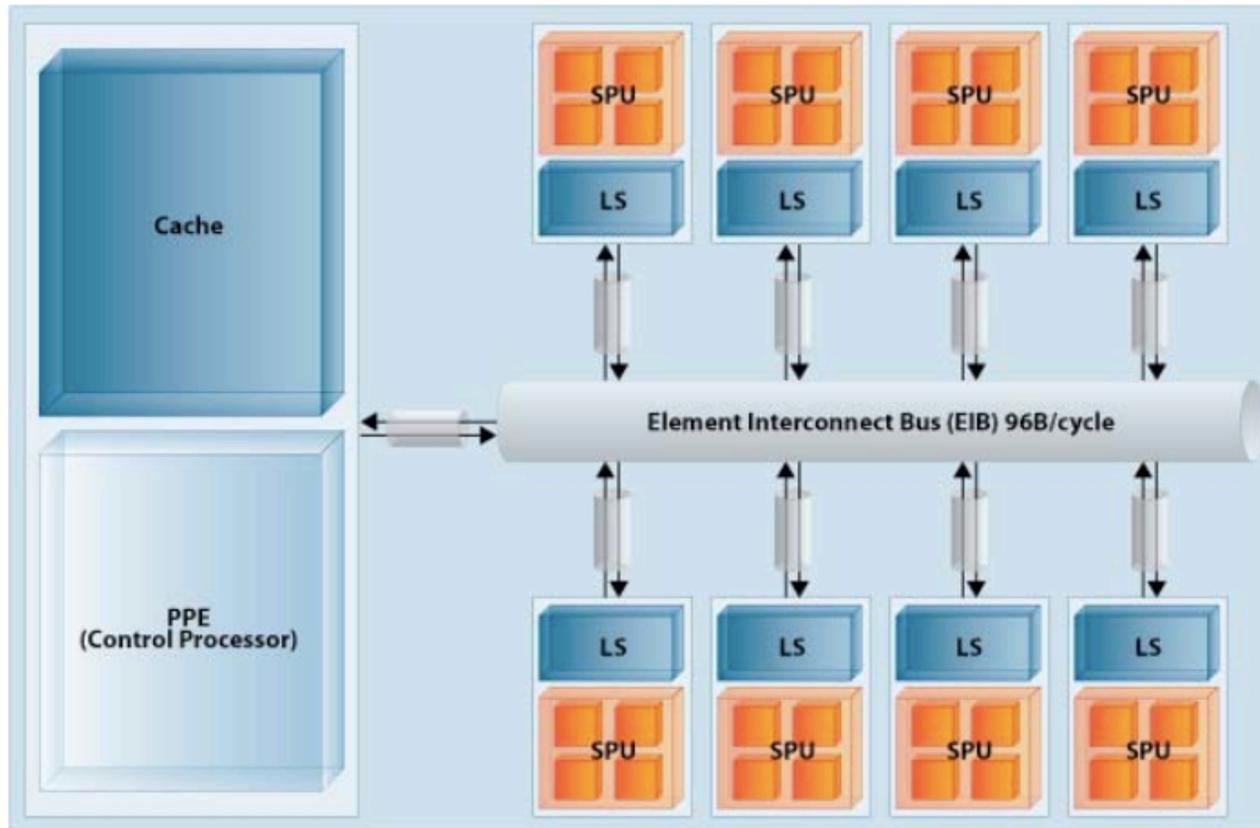
Generalized Multicore



Slide Source: Michael McCool, Rapid Mind, SuperComputing, 2007



Cell B.E. Architecture

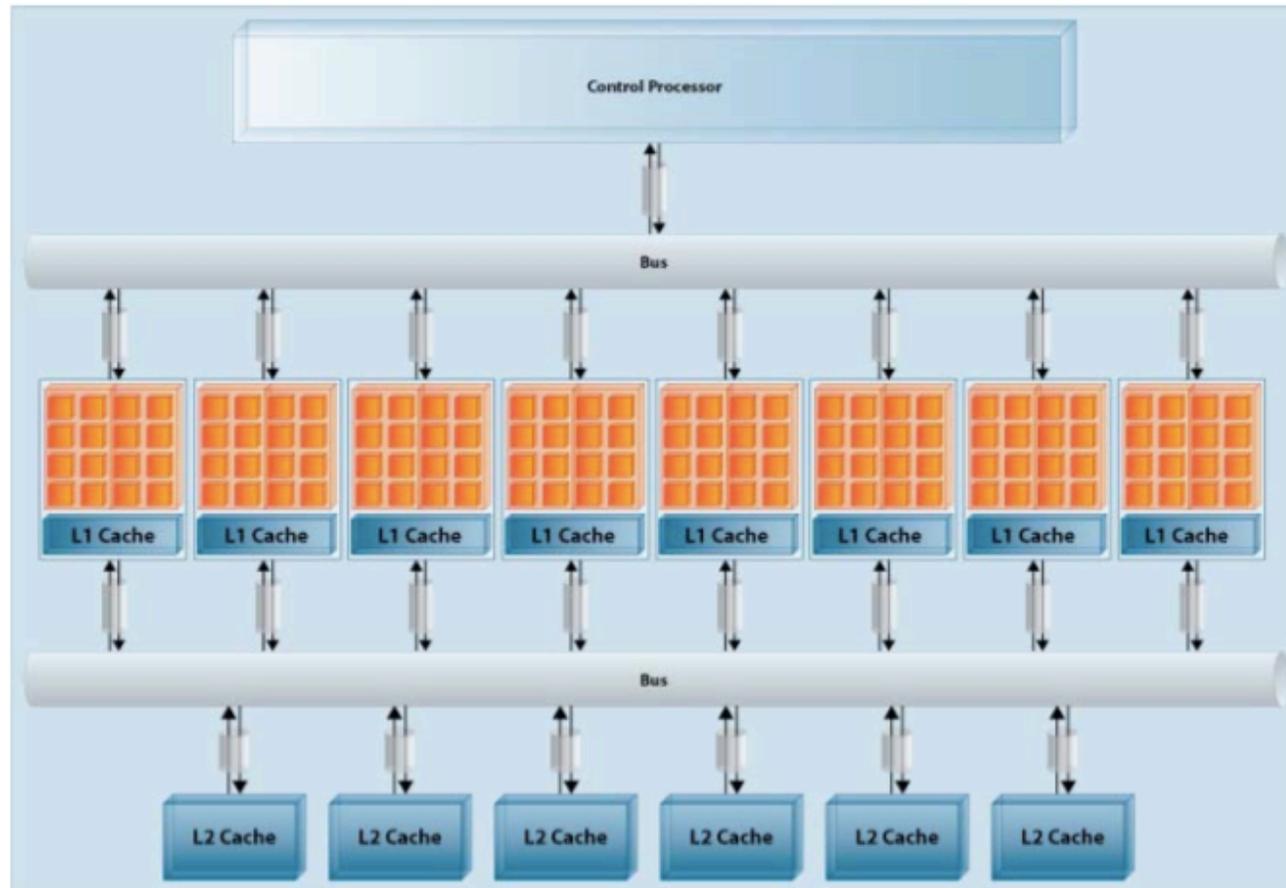


Slide Source: Michael McCool, Rapid Mind, SuperComputing, 2007

CISC 879 : Advanced Parallel Programming



NVIDIA GPU Architecture G80



Slide Source: Michael McCool, Rapid Mind, SuperComputing, 2007

CISC 879 : Advanced Parallel Programming



Commercial Multicores



| Name | Clovertwn | Opteron | Cell | Niagara 2 |
|--------------|-------------------------------|---------|-------------------|--------------------|
| Chips*Cores | 2*4 = 8 | 2*2 = 4 | 1*8 = 8 | 1*8 = 8 |
| Architecture | 4-/3-issue, SSE3, OOO, caches | | 2-VLIW, SIMD, RAM | 1-issue, MT, cache |
| Clock Rate | 2.3 GHz | 2.2 GHz | 3.2 GHz | 1.4 GHz |
| Peak MemBW | 21 GB/s | 21 GB/s | 26 GB/s | 41 GB/s |
| Peak GFLOPS | 74.6 GF | 17.6 GF | 14.6 GF | 11.2 GF |

Slide Source: Dave Patterson, Manycore and Multicore Computing Workshop, 2007