Programming for Cell Processors
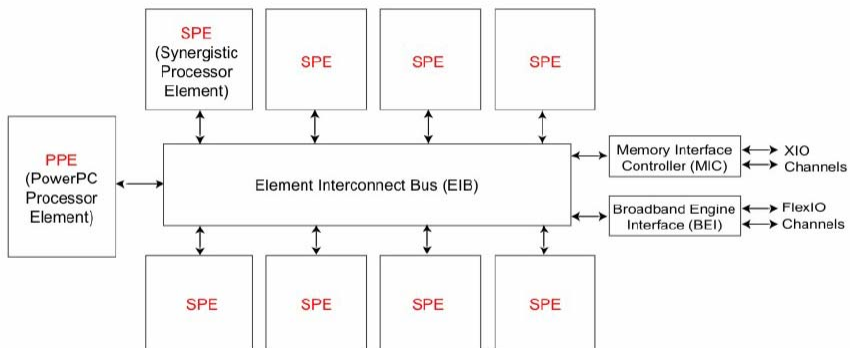
Huiyang Zhou

School of Electrical Engineering and Computer Science
University of Central Florida

Review of Cell Processor Architecture

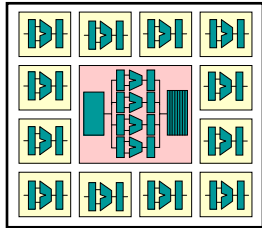- Heterogeneous Chip Multiprocessors (CMP)

Future *High Performance CMPs*: A shared view of comp. arch. community

- Heterogeneous cores on a single chip
- *M* large complex OOO cores and *N* small in-order cores (*N>>M*)
  - Large OOO cores for control-intensive, hard-to-parallelize code (Instruction-level parallelism, Memory-level parallelism, aggressive speculation)
  - Many small in-order processors for data-level parallelism, task/thread-level parallelism
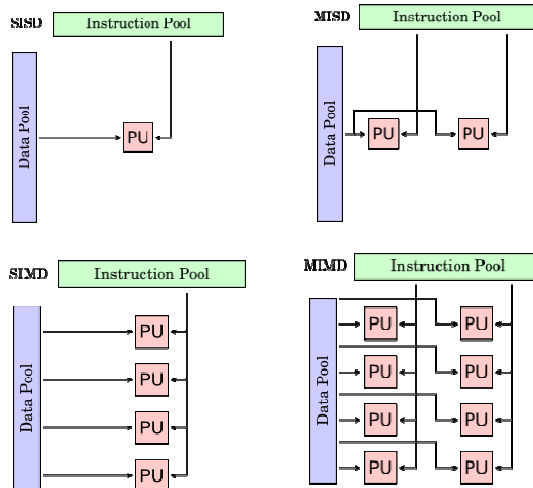
- E.g.

- Cell is heterogeneous but does not really fit the description
  - PPE is far from powerful enough
  - Next generation of Cell may address PPE performance

University of Central Florida

Flynn's Taxonomy

Figure from wikipedia

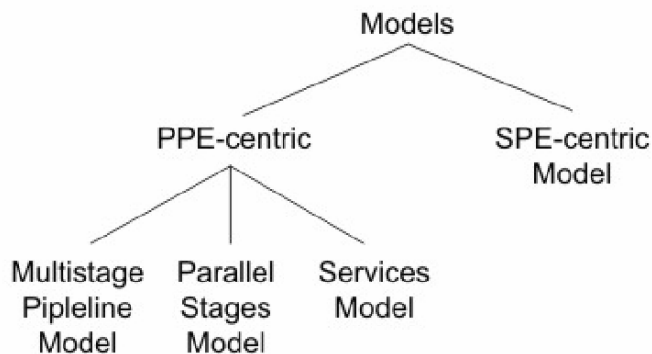University of Central Florida

MIMD Architectures

- Further Division of MIMD
  - Single Program, Multiple Data Stream (SPMD)
    - Exploit Data-Level Parallelism
    - Difference between SIMD: no lockstep
    - In GPU, SIMD in a warp/cluster; SPMD among multiple warps/clusters
  - Multiple Program, Multiple Data Stream (MPMD)
    - Exploit Function/Task-Level Parallelism
    - E.g., Master/worker
- Cell processors
  - Supports MPMD
  - In each SPU, SIMD execution exploits data-level parallelism
  - Multiple SPUs can execute *different* codes.

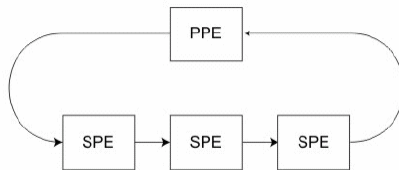University of Central Florida

Workload Partition

- PPE-centric vs. SPE-centric

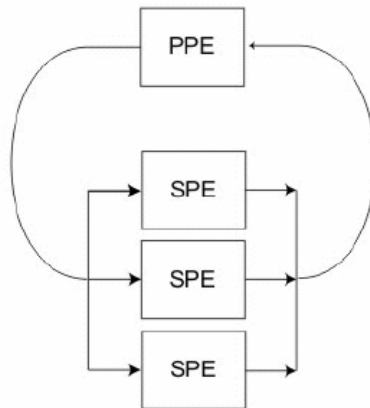University of Central Florida

Multistage Pipeline Model

- Main issues:
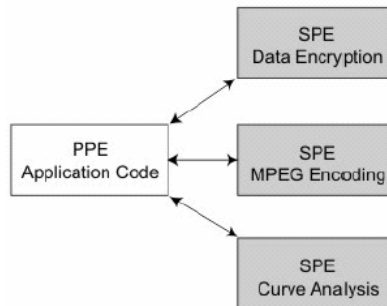 - Load balance
  - Data transmission

Parallel Stage Model

- SPMD – Similar to CUDA or Brooks+ model

Service Model

- MPMD

Outline

- Introduction of programming models
- "Hello world"
  - Three versions
- Run-time support
- Programming using vectors: SIMDization
- DMA
- Program Optimization