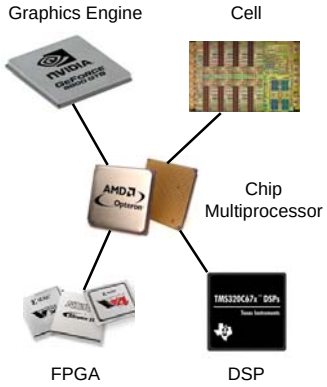
Visions for Application Development on Hybrid Computing Systems

Roger D. Chamberlain, Joseph M. Lancaster, Ron K. Cytron
Department of Computer Science and Engineering
Washington University in St. Louis

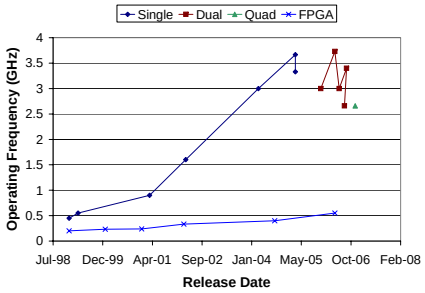
Reconfigurable Systems Summer Institute
July 17-20, 2007
National Center for Supercomputing Applications
Urbana, Illinois

Hybrid Systems



- Hybrid systems have diverse architectures, usually with distributed memory subsystems

- Individual processor core performance isn't increasing much any more
- Clock rate gap is decreasing



- Application acceleration has been demonstrated many times [1,2]
- Power savings can be substantial

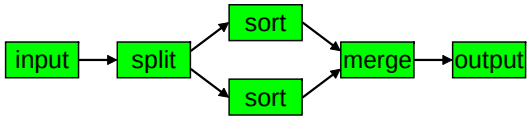
Authoring Applications

- Memory is a critical resource

Chip	Process	Memory
Intel Xeon (Tulsa)	65 nm	16 MB
Xilinx Virtex-5	65 nm	1.4 MB
TI TMS320C DSP	90 nm	2 MB

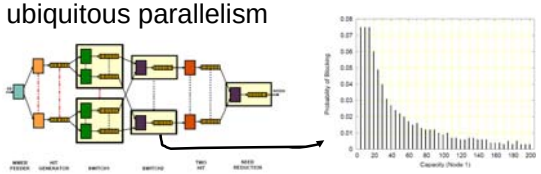
- Applications often have large memory footprint
- Dataflow programming models nudge programmers in the right direction

- Language choices are important
- Use familiar languages to express kernel computations, e.g.,
  - C/C++/Java/etc. for processors
  - Verilog/VHDL for FPGAs
  - Brook/CUDA for graphics engines
- Use coordination language to express communications between kernels [3,4]



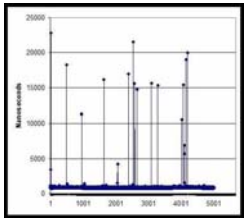
Debugging and Understanding the Performance of Applications

- Significant challenges
  - Diversity leads to incoherent system
  - Components have limited observability and ubiquitous parallelism



- Need hardware resources throughout system enabling monitoring
  - Dedicate resources for observing and processing system data
  - Reconfigurable monitoring elements allow data reduction and flexibility [5]

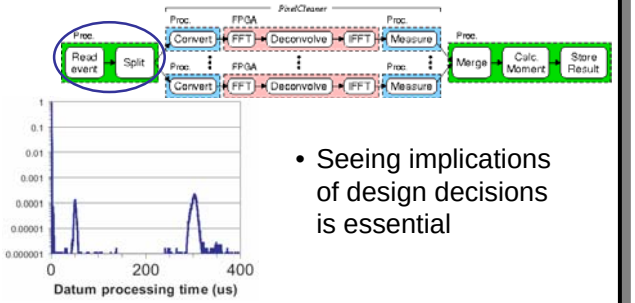
- Deploy monitors at the communication arcs
  - Use results to develop and validate system performance models
  - Unobtrusive monitoring allows for detection of rare events



- Provide "gdb-like" debugging environment with performance visualization programmers need

Putting it all Together

- Easing the deployment task is key



- Seeing implications of design decisions is essential

- A development environment can do all this [6]

[1] Z. Guo et al. A quantitative analysis of the speedup factors of FPGAs over processors. In *Proc. of Symp. on Field Programmable Gate Arrays*, Feb. 2004.
[2] I. Buck et al. Brook for GPUs: Stream computing on graphics hardware. *ACM Trans. on Graphics*, 23(3):777-786, Aug. 2004.
[3] E. Lee. The problem with threads. *IEEE Computer*, 39(5):33-42, 2006.
[4] M. Franklin et al. Auto-pipe and the X language: A pipeline design tool and description language. In *Proc. of IPDPS*, April 2006.
[5] M. Schulz et al. Owl: Next Generation System Monitoring. In *Proc. of Computing Frontiers*, May 2005.
[6] R. Chamberlain et al. Application Development on Hybrid Systems. To appear in *Proc. of Supercomputing*, Nov. 2007.