FPGAs in HPC
Special session

Andrew Putnam – Microsoft Azure
June 21, 2021
Most Promising Features

• Advanced Network Integration
  • Compute directly on network packets
  • Lowest latency, highest bandwidth (RDMA)
  • Filtering – avoiding work entirely
  • Selective and adaptive Multicast

• Advancing DSP block functionality (e.g. FP, Stratix 10 NX)

• Deep pipelines & MISD Parallelism

• Cloud FPGAs allow for unprecedented access and scaling
  • But Cloud FPGAs are not a given – must be commercially viable
  • Shifting baseline – GPUs are also gaining direct network access
Missing Technology / Future Research

- Ability to span generations/vendors beyond source level
  - Cohesive development framework (Shell)
- Ability to scale up/down easily
  - Within one FPGA, and across multiple FPGAs
  - Enable incremental development & iterative debug (at home)
- IP / library integration
- Tools for composing & understanding deep multi-machine pipelines

Make things easier for developers – way beyond HLS
How Can Academics Help Industry?

• Continue to research irregular parallelism, sparsity, and control-flow divergence
• Need tools for anticipating different levels of integration/hierarchy
  • On-chip, T0, T1, T2...
  • Make use of general-purpose CPU instances where possible
• Memory tools for scratchpad memories
  • Rethink Coherence, Consistency for distributed scratchpads (Must automate)
• Integrity & Security
  • Integrated data integrity checking & checkpoints
• Next-generation DSP blocks and Tiles
• Cloud-scale debug tools
Miriam Leeser 08:38 AM

The catapult model does not allow users to write their own applications to run on their cloud FPGA system. Other speakers mention challenges with writing applications and lack of tools. Will programming FPGAs always be a specialized skill or can we make FPGAs in HPC more usable?

Karl Friebel 08:38 AM

What are your visions for how we can avoid the massive penalty in porting all the current, homogeneous HPC applications and their algorithms, such as weather simulations, without re-inventing them? Do you have any ideas for enabling a gradual transition?

Fernando Martin del Campo 08:41 AM

How important will HLS be in the future of FPGAs and HPC. And, are the legacy of C, C++ hurting the potential of HLS?
2. What technology is missing in infrastructure and operation of HPC systems using FPGAs? What should be researched and developed for hardware and software of FPGA-based HPC?

How are you Programming FPGAs?
- Abstraction, HLS, OpenACC/OpenMP

What do you want to have a tool for better (easier/more productive) programming

**Abstraction**: Tool for conversion from abstract code to FPGA implement.

**Open/common IP, lib modules** really available for various FPGA products over vendors.

Common **Open shell, open BSP, and common drivers**?

- What to offload, or how to do,
  - **More runtime support**,
  - **good debugging tool**, structured way to annotate (but difficult due to long compilation time),
  - Autotuning

**Common API for communication on FPGA**

**Common HLS (for vendors)**

Common something