Highlights

• Recent trends in extreme-scale HPC paint an ambiguous future
  – Contemporary systems provide evidence that power constraints are driving architectures to change rapidly
  – Multiple architectural dimensions are being (dramatically) redesigned: Processors, node design, memory systems, I/O
  – Complexity is our main challenge

• Applications and software systems are all reaching a state of crisis
  – Applications will not be functionally or performance portable across architectures
  – Programming and operating systems need major redesign to address these architectural changes
  – Procurements, acceptance testing, and operations of today’s new platforms depend on performance prediction and benchmarking.

• We need portable programming models and performance prediction now more than ever!
  – Heterogeneous processing
    • OpenACC->FPGAs
    • Intelligent runtime system (IRIS) (not covered today)
    • Clacc – OpenACC support in LLVM (not covered today)
  – Emerging memory hierarchies (NVM)
    • DRAGON – transparent NVM access from GPUs (not covered today)
    • NVL-C – user management of nonvolatile memory in C (not covered today)
    • Papyrus – parallel aggregate persistent storage (not covered today)

• Performance prediction is critical for design and optimization (not covered today)
Time for a short poll...
Q: Think back 10 years. How many of you would have predicted that many of our top HPC systems would be GPU-based architectures?
Q: Think forward 10 years. How many of you predict that most of our top HPC systems will have the following architectural features?

- X86 multicore CPU
- GPU
- FPGA/Reconfigurable processor
- Neuromorphic processor
- Deep learning processor
- Quantum processor
- RISC-V processor
- Some new unknown processor
- All/some of the above in one SoC
Q: Now imagine you are building a new application with ~3M LOC and 20 team members over the next 10 years. What on-node programming model/system do you use?

- C, C++, Fortran
- C++ templates, policies, etc (e.g., AMP, Kokkos, RAJA, )
- CUDA, cu***, HIP
- OpenCL, SYCL
- OpenMP or OpenACC
- R, Python, Matlab, etc
- A Domain Specific Language (e.g., Claw, PySL)
- A Domain Specific Framework (e.g., PetSc)
- Some new unknown programming approach
- All/some of the above
Motivating Trends
Contemporary devices are approaching fundamental limits

Dennard scaling has already ended. Dennard observed that voltage and current should be proportional to the linear dimensions of a transistor: 2x transistor count implies 40% faster and 50% more efficient.


Foundries' Sales Show Hard Times Continuing

Peter Clarke 5/23/2016 09:33 PM EDT 2 comments

Uncertainty Grows For 5nm, 3nm

Semiconductor Engineering

GlobalFoundries Forfeit 7nm Manufacturing - EE Times Asia

Samsung to Invest $115 Billion in Foundry & Chip Businesses by 2030

Intel's 10nm Is Broken, Delayed Until 2019 by Paul Alcorn April 26, 2018 at 6:30 PM

GlobalFoundries Selling ASIC Business to Marvell

Another Step Toward the End of Moore's Law

Samsung and TSMC move to 5-nanometer manufacturing
Business climate reflects this uncertainty, cost, complexity, consolidation.

---

**NVIDIA Buys Mellanox To Bring HPC Scaling To Data Centers**

The 2019 semiconductor merger and acquisition season has officially been kicked off, technology and Xilinx said in a statement on the creation of an.

**Hewlett Packard Enterprise to Acquire Supercomputer Pioneer Cray**

Hewlett Packard Enterprise will pay about $1.4 billion to acquire Cray, which has designed some of the most powerful computer systems in use. Credit: Paco Freire/SOPA Images, via LightRocket and Getty Images.

---

**Tech giant ARM Holdings sold to Japanese firm for £24bn**

Son puts stake worth $8bn in UK's largest tech company into $100bn Vision Fund.

**Amazon Is Becoming an AI Chip Maker, Speeding Alexa Responses**

Amazon is developing a chip designed for artificial intelligence to work on the Echo and other hardware powered by Amazon's Alexa virtual assistant, says a person familiar with Amazon's plans. The chip should allow Alexa-powered devices to respond more quickly to commands, allowing more data processing to be handled in the cloud.

---

**SANDISK COMPLETES ACQUISITION OF FUSION I/O**

Western Digital, based in San Jose, Calif., has acquired Fusion, the San Diego company that makes storage solutions for servers, and has a go-to-market presence in the enterprise flash solutions in the data center.

---

**TOSHIBA**

Toshiba to sell ‘minority stake’ in chip business to Western Digital.

**Q1 Chip Sales Drop Among Largest on Record**

SAN FRANCISCO -- Global chip sales sank by 15.5% sequentially in the first quarter, among the largest quarter-to-quarter declines for the industry in the last 35 years.

Chip sales totaled $96.8 billion in the first quarter, down from $114.7 billion last year, according to the World Semiconductor Trade Statistics (WSTS) organization, which tracks sales data from chipmaker member companies. On a year-over-year basis, first-quarter revenue for chip manufacturers tumbled 11.4%.
Sixth Wave of Computing

http://www.kurzweilai.net/exponential-growth-of-computing
Predictions for Transition Period

Optimize Software and Expose New Hierarchical Parallelism

• Redesign software to boost performance on upcoming architectures
• Exploit new levels of parallelism and efficient data movement

Architectural Specialization and Integration

• Use CMOS more effectively for specific workloads
• Integrate components to boost performance and eliminate inefficiencies
• Workload specific memory+storage system design

Emerging Technologies

• Investigate new computational paradigms
  • Quantum
  • Neuromorphic
  • Advanced Digital
  • Emerging Memory Devices
Predictions for Transition Period

Optimize Software and Expose New Hierarchical Parallelism

• Redesign software to boost performance on upcoming architectures
• Exploit new levels of parallelism and efficient data movement

Architectural Specialization and Integration

• Use CMOS more effectively for specific workloads
• Integrate components to boost performance and eliminate inefficiencies
• Workload specific memory+storage system design

Emerging Technologies

• Investigate new computational paradigms
  • Quantum
  • Neuromorphic
  • Advanced Digital
  • Emerging Memory Devices
Predictions for Transition Period

Optimize Software and Expose New Hierarchical Parallelism

- Redesign software to boost performance on upcoming architectures
- Exploit new levels of parallelism and efficient data movement

Architectural Specialization and Integration

- Use CMOS more effectively for specific workloads
- Integrate components to boost performance and eliminate inefficiencies
- Workload specific memory+storage system design

Emerging Technologies

- Investigate new computational paradigms
  - Quantum
  - Neuromorphic
  - Advanced Digital
  - Emerging Memory Devices
Quantum computing: Qubit design and fabrication have made recent progress but still face challenges.
Neuromorphic (Brain Inspired) Computing

- SpiNNaker
- IBM True North
- BrainScaleS
- DANNA
- Others...


New devices: Carbon Nanotube Transistors and Circuits


Emerging Memory Devices

Image Source: IMEC
Predictions for Transition Period

Optimize Software and Expose New Hierarchical Parallelism

- Redesign software to boost performance on upcoming architectures
- Exploit new levels of parallelism and efficient data movement

Architectural Specialization and Integration

- Use CMOS more effectively for specific workloads
- Integrate components to boost performance and eliminate inefficiencies
- Workload specific memory+storage system design

Emerging Technologies

- Investigate new computational paradigms
  - Quantum
  - Neuromorphic
  - Advanced Digital
  - Emerging Memory Devices
Pace of Architectural Specialization is Quickening

- Industry, lacking Moore’s Law, will need to continue to differentiate products (to stay in business)
  - Use the same transistors differently to enhance performance
- Architectural design will become extremely important, critical
  - Dark Silicon
  - Address new parameters for benefits/curse of Moore’s Law
- 50+ new companies focusing on hardware for Machine Learning


https://www.thebroadcastbridge.com/content/entry/1094/altera-announces-ariia-10-2666mbps-ddr4-memory-fpga-interface

http://www.wired.com/2016/05/google-tpu-custom-chips/


http://www.nvidia.com/content/topics/developer/technology/nvidia-tpu.html

Analysis of Apple A-* SoCs

http://vlsiarch.eecs.harvard.edu/accelerators/die-photo-analysis
Growing Open Source Hardware Movement Enables Rapid Chip Design

RISC-V Ecosystem

Software

Open-source software:
Gcc, binutils, glibc, Linux, BSD, LLVM, QEMU, FreeRTOS, ZephyrOS, LiteOS, SylixOS, ...

Commercial software:
Lauterbach, Segger, Micrium, ExpressLogic, ...

Hardware

Open-source cores:
Rocket, BOOM, RISCY, Ariane, PicoRV32, Piccolo, SCR1, Hummingbird, ...

Commercial core providers:
Andes, Bluespec, Cloudbear, Codasip, Cortus, C-Sky, Nuclei, SiFive, Syntacore, ...

Inhouse cores:
Nvidia, +others

RISC-V Summit, 2018
DARPA ERI Programs Aiming for Agile (and Frequent) Chip Creation

IDEA/POSH End State – A Universal Hardware Compiler

$ git clone https://github.com/darpa/idea
$ git clone https://github.com/darpa/posh
$ cd posh
$ make soc42

A. Olofsson, 2018
Summary: Transition Period will be Disruptive – Opportunities and Pitfalls Abound

• New devices and architectures may not be hidden in traditional levels of abstraction

• Examples
  – A new type of CNT transistor may be completely hidden from higher levels
  – A new paradigm like quantum may require new architectures, programming models, and algorithmic approaches

<table>
<thead>
<tr>
<th>Layer</th>
<th>Switch, 3D</th>
<th>NVM</th>
<th>Approximate</th>
<th>Neuro</th>
<th>Quantum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Algorithm</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Language</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>API</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Arch</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>ISA</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Microarch</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>FU</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Logic</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Device</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

Adapted from IEEE Rebooting Computing Chart
Department of Energy (DOE) Roadmap to Exascale Systems
An impressive, productive lineup of accelerated node systems supporting DOE’s mission

<table>
<thead>
<tr>
<th>Pre-Exascale Systems [Aggregate Linpack (Rmax) = 323 PF]</th>
<th>First U.S. Exascale Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>2016</td>
</tr>
<tr>
<td>Titan (9)</td>
<td>Summit (1)</td>
</tr>
<tr>
<td>ORNL</td>
<td>ORNL</td>
</tr>
<tr>
<td>Cray/AMD/NVIDIA</td>
<td>AMD/Cray</td>
</tr>
<tr>
<td>Mira (21)</td>
<td>Theta (24)</td>
</tr>
<tr>
<td>ANL</td>
<td>ANL</td>
</tr>
<tr>
<td>IBM BG/Q</td>
<td>IBM BG/Q</td>
</tr>
<tr>
<td>Sequoia (10)</td>
<td>Trinity (6)</td>
</tr>
<tr>
<td>LANL</td>
<td>LANL/SNL</td>
</tr>
<tr>
<td>LLNL</td>
<td>Cray/Intel Xeon/KNL</td>
</tr>
<tr>
<td>LLNL</td>
<td>LLNL</td>
</tr>
<tr>
<td>SSD</td>
<td>SSD</td>
</tr>
<tr>
<td>LBNL</td>
<td>LBNL</td>
</tr>
<tr>
<td>Cray/AMD/NVIDIA</td>
<td>Cray/AMD/NVIDIA</td>
</tr>
<tr>
<td>LLNL</td>
<td>LLNL</td>
</tr>
<tr>
<td>TBD</td>
<td>TBD</td>
</tr>
</tbody>
</table>

Heterogeneous Cores
Deep Memory incl NVM
Plateaing I/O Performance

First U.S. Exascale Systems

First U.S. Exascale Systems

Pre-Exascale Systems [Aggregate Linpack (Rmax) = 323 PF]

2012: Titan (9)
ORNL Cray/AMD/NVIDIA

2016: Mira (21)
ANL IBM BG/Q

2018: Theta (24)
ANL Cray/Intel KNL

2020: Summit (1)
ORNL IBM/NVIDIA

First U.S. Exascale Systems

2021-2023: Aurora

ANL Intel/Cray

ORNL AMD/Cray

LANL/SNL Cray/Intel Xeon/KNL

LANL/SNL Cray/AMD/NVIDIA

LANL/SNL TBD

LLNL TBD

Jan 2018
**Summit (IBM POWER9+NVIDIA Volta) Node installed**

Experimental Computing Lab (ExCCL) managed by the ORNL Future Technologies Group

IBM Summit Node with 6 Nvidia Tesla V100 GPUs (B335-GTX)

- Same CPU/GPU/Memory as nodes in OLCF Summit
  - 2 Power9 CPUs (IBM E2120)
  - 22 Cores each, 4 threads/core
  - 60GB main memory
  - 6 Tesla V100 SXM2 32GB GPUs

- Provides a development and evaluation environment for Power9/V100 GPUs

- Tracks (as closely as possible) the software stack in use on Summit

- Shared / Queued / Single User availability modes will be available

---

**NVIDIA DGX Workstation Available**

Experimental Computing Lab (ExCCL) managed by the ORNL Future Technologies Group

- 4X Tesla V100 GPUs
- TFLOPS (Mixed precision) 500
- GPU Memory 128 GB total system
- NVIDIA Tensor Cores 2,560
- NVIDIA CUDA® Cores 20,480
- CPU Intel Xeon E5-2698 v4 2.2 GHz (20-Core)
- System Memory 256 GB RDIMM DDR4
- Full NVIDIA stack
- Other compilers/tools installable on request

---

**AMD Radeon VII Available**

Experimental Computing Lab (ExCCL) managed by the ORNL Future Technologies Group

- AMD Radeon VII, Vega 20 Architecture
  - 540 S in TSMC 7FF process, 13.2B transistors
  - 50 Compute Units with 3.4 GF peak TF
  - 16 GB HBM2 with 4096-bit width (x4 ~1TBps bandwidth)
  - TRP 500W
  - PCIe 3.0 x16

- Intel Xeon Skylake Host
  - HP Z4 64 Workstation w/ PCIe 3.0 x16
  - X-3224 6/820 host
  - 2 CPU / 4 cores / 2 threads/core
  - 512 GB SSD uncommitted/available

- Software
  - AMD ROCm development tools
  - HIP (Heterogeneous Compute Interface for Portability) available
  - OpenCL 2.1

- Additional Details
  - [https://www.amdtech.com/blog/13032/amd-radeon-vii-high-end/](https://www.amdtech.com/blog/13032/amd-radeon-vii-high-end/)

---

**ARM ThunderX2 Node Available**

Experimental Computing Lab (ExCCL) managed by the ORNL Future Technologies Group

ThunderX2 Workstation

- Cavium (Marvell) ThunderX2 with ARMv8.1 instruction set.
- 2 Cpus, each with 28 Cores with 4 threads/core
- 128 GiB Main Memory
- Gigabyte M701-F81-00 motherboard
- Multiple access levels available to researchers investigating ARM8v1 performance
- Traditional ARM/Linux software stack available
NVIDIA Jetson AGX Xavier SoC available

Experimental Computing Lab (ExCL) managed by the ORNL Future Technologies Group

NVIDIA Jetson AGX Xavier:

• High-performance system on a chip for autonomous machines

• Heterogeneous SoC contains:
  – Eight-core 64-bit ARMv8.2 CPU cluster (Carmel)
  – 1.4 CUDA TFLOPS (FP32) GPU with additional inference optimizations (Volta)
  – 11.4 DL TOPS (INT8) Deep learning accelerator (NVDA)
  – 1.7 CV TOPS (INT8) 7-slot VLIW dual-processor Vision accelerator (PVA)
  – A set of multimedia accelerators (stereo, LDC, optical flow)

• Provides researchers access to advanced high-performance SOC environment
Qualcomm 855 SoC (SM8510P)
Experimental Computing Lab (ExCL) managed by the ORNL Future Technologies Group

- Adreno 640
- Snapdragon X24 modem
- Snapdragon X50 5G (external) modem for 5G devices
- Qualcomm Wi-Fi 6-ready mobile platform: (802.11ax-ready)
- Qualcomm 60 GHz Wi-Fi mobile platform: (802.11ay, 802.11ad)
- Bluetooth Version: 5.0
- Bluetooth Speed: 2 Mbps
- High accuracy location with dual-frequency GNSS.

- Hexagon 690
  - Quad threaded Scalar Core
  - DSP + 4 Hexagon Vector Xccelerators
  - New Tensor Xccelerator for AI
  - Apps: AI, Voice Assistance, AV codecs

- Kyro 485
  - Snapdragon X24 LTE (855 built-in) modem LTE Category 20
  - Snapdragon X50 5G (external) modem for 5G devices
  - Qualcomm Wi-Fi 6-ready mobile platform: (802.11ax-ready, 802.11ac Wave 2, 802.11ay, 802.11ad)
  - Spectra 360 ISP
  - New dedicated Image Signal Processor (ISP)
  - Dual 14-bit CV-ISPs; 48MP @ 30fps single camera
  - Hardware CV for object detection, tracking, stereo depth process
  - 6DoF XR Body tracking, H265, 4K60 HDR video capture, etc.

For more information or to apply for an account, visit https://excl.ornl.gov/
Intel Stratix 10 FPGA available

Experimental Computing Lab (ExCL) managed by the ORNL Future Technologies Group

- Intel Stratix 10 FPGA and four banks of DDR4 external memory
  - Board configuration: Nallatech 520 Network Acceleration Card
- Up to 10 TFLOPS of peak single precision performance
- 25MBytes of L1 cache @ up to 94 TBytes/s peak bandwidth
- 2X Core performance gains over Arria® 10
- Quartus and OpenCL software (Intel SDK v18.1) for using FPGA
- Provide researcher access to advanced FPGA/SOC environment

For more information or to apply for an account, visit https://excl.ornl.gov/
Directive-based Solutions for FPGA Computing


Challenges in FPGA Computing

• Programmability and Portability Issues
  – Best performance for FPGAs requires writing Hardware Description Languages (HDLs) such as VHDL and Verilog; too complex and low-level
    • HDL requires substantial knowledge on hardware (digital circuits).
    • Programmers must think in terms of a state machine.
    • HDL programming is a kind of digital circuit design.
  – High-Level Synthesis (HLS) to provide better FPGA programmability
    • SRC platforms, Handel-C, Impulse C-to-FPGA compiler, Xilinx Vivado (AutoPilot), FCUDA, etc.
    • None of these use a portable, open standard.
Standard, Portable Programming Models for Heterogeneous Computing

• OpenCL
  – Open standard portable across diverse heterogeneous platforms (e.g., CPUs, GPUs, DSPs, Xeon Phis, FPGAs, etc.)
  – Much higher than HDL, but still complex for typical programmers.

• Directive-based accelerator programming models
  – OpenACC, OpenMP4, etc.
  – Provide higher abstraction than OpenCL.
  – Most of existing OpenACC/OpenMP4 compilers target only specific architectures; none supports FPGAs.
FPGAs | Approach

• Design and implement an OpenACC-to-FPGA translation framework, which is the first work to use a standard and portable directive-based, high-level programming system for FPGAs.

• Propose FPGA-specific optimizations and novel pragma extensions to improve performance.

• Evaluate the functional and performance portability of the framework across diverse architectures (Altera FPGA, NVIDIA GPU, AMD GPU, and Intel Xeon Phi).

OpenARC System Architecture

OpenARC Runtime

OpenARC Compiler

OpenARC Front-End
- C Parser
- Directive Parser
- Preprocessor
- General Optimizer

OpenARC Back-End
- Kernels & Host Program Generator
- Device Specific Optimizer

OpenARC Auto-Tuner
- Tuning Configuration Generator
- Search Space Pruner

Output Codes
- Kernels for Target Devices

Host Program

CUDA, OpenCL Libraries

HeteroIR Common Runtime with Tuning Engine

Feedback

Input C Program

LLVM Back-End
- Extended LLVM IR Generator
- NVL Passes
- Standard LLVM Passes

OpenACC

OpenMP 4

NVL-C

CUDA, GPU
- GCN
- Xeon Phi
- Altera FPGA

NVM

Run

Executable

pmem.io NVM Library

NVM Runtime

Run
FPGA OpenCL Architecture

- FPGA
  - External DDR Memory
  - External Memory Controller and PHY
  - Global Memory Interconnect
  - Number of Replicated Compute Units
- Host Processor
  - PCIe
  - External DDR Memory
- Pipeline Depth
- Vector Width

- Local Memory Interconnect
  - Kernel Pipel ine
  - Memory
Kernel-Pipelining Transformation Optimization

• Kernel execution model in OpenACC
  – Device kernels can communicate with each other only through the device global memory.
  – Synchronizations between kernels are at the granularity of a kernel execution.

• Altera OpenCL channels
  – Allows passing data between kernels and synchronizing kernels with high efficiency and low latency
Kernel-Pipelining Transformation Optimization (2)

(a) Input OpenACC code

```c
#pragma acc data copyin (a) create (b) copyout (c)
{
#pragma acc kernels loop gang worker present (a, b)
for(i=0; i<N; i++) { b[i] = a[i]*a[i]; }
#pragma acc kernels loop gang worker present (b, c)
for(i=0; i<N; i++) {c[i] = b[i]; }
}
```

(b) Altera OpenCL code with channels

```c
channel float pipe_b;
__kernel void kernel1(__global float* a) {
int i = get_global_id(0);
write_channel_altera(pipe_b, a[i]*a[i]);
}
__kernel void kernel2(__global float* c) {
int i = get_global_id(0);
c[i] = read_channel_altera(pipe_b);
}
```
Kernel-Pipelining Transformation Optimization (3)

(a) Input OpenACC code

```c
#pragma acc data copyin (a) create (b) copyout (c)
{
    #pragma acc kernels loop gang worker present (a, b)
    for(i=0; i<N; i++) { b[i] = a[i]*a[i]; }
    #pragma acc kernels loop gang worker present (b, c)
    for(i=0; i<N; i++) { c[i] = b[i]; }
}
```

(c) Modified OpenACC code for kernel-pipelining

```c
#pragma acc data copyin (a) pipe (b) copyout (c)
{
    #pragma acc kernels loop gang worker pipeout (b) present (a)
    for(i=0; i<N; i++) { b[i] = a[i]*a[i]; }
    #pragma acc kernels loop gang worker pipein (b) present (c)
    for(i=0; i<N; i++) { c[i] = b[i]; }
}
```
FPGA-specific Optimizations

- Single work-item
- Collapse
- Reduction
- Sliding window
- (Branch-variant code motion)
- (Custom unrolling)
Reduction Optimization

- Scalar reduction
- Combine an array of values into a single scalar value
- Associative Operations: Sum, Multiply, Max, Min, AND, OR
- OpenMP and OpenACC support a *reduction* clause to direct the compiler to implement parallel reductions.
Pipeline Parallel Reduction

- We can do multi-threaded parallel tree-based reduction on FPGA
- Can we also do pipeline parallel reduction?
- Scalar reductions have a loop-carried dependency on the reduction variable

```c
#pragma acc parallel loop reduction(+:sum) \
    num_gangs(1) num_workers(1)
for (int i = 0; i < N; ++i) {
    sum += input[i];
}
```
What we want

What we have
Shift Registers

- “A cascade of flip flops sharing the same clock in which the output of each flip-flop is connected to the 'data' input of the next flip-flop in the chain”
- An array where each value can be shifted left or right efficiently (in one cycle).
Shift-register Reduction

- Accumulate partial sums in a shift registers, and perform a final reduction over the partial sums
- By using shift registers we can increase the dependance distance, improving pipeline efficiency

- The size of the shift register needed is directly proportional to the cost of the reduction operation.
OpenACC input

```c
#pragma acc parallel loop reduction(+:sum)
for (int i = 0; i < N; ++i) {
    sum += input[i];
}
```

OpenCL output (Generated by OpenARC)

```c
#pragma acc parallel loop reduction(+:sum) \
    num_gangs(1) num_workers(1)
for (int i = 0; i < N; ++i) {
    sum += input[i];
}
```
# Reduction Optimization (SRAD Benchmark)

<table>
<thead>
<tr>
<th>Reduction Type</th>
<th>Runtime (s)</th>
<th>Resource Usage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-threaded Tree-based</td>
<td>31.053</td>
<td>45</td>
</tr>
<tr>
<td>Single Work-item</td>
<td>78.307</td>
<td>38</td>
</tr>
<tr>
<td>Single Work-item Shift Register</td>
<td>23.239</td>
<td>50</td>
</tr>
</tbody>
</table>

![Diagram showing reduction types](diagram.png)
Overall Performance

FPGAs prefer applications with deep execution pipelines (e.g., FFT-1D and FFT-2D), performing much higher than other accelerators.

For traditional HPC applications with abundant parallel floating-point operations, it seems to be difficult for FPGAs to beat the performance of other accelerators, even though FPGAs can be much more power-efficient.

- Tested FPGA does not contain dedicated, embedded floating-point cores, while others have fully-optimized floating-point computation units.

Current and upcoming high-end FPGAs are equipped with hardened floating-point operators, whose performance will be comparable to other accelerators, while remaining power-efficient.
Final Report on Workshop on Extreme Heterogeneity

1. Maintaining and improving programmer productivity
   – Flexible, expressive, programming models and languages
   – Intelligent, domain-aware compilers and tools
   – Composition of disparate software components

• Managing resources intelligently
   – Automated methods using introspection and machine learning
   – Optimize for performance, energy efficiency, and availability

• Modeling & predicting performance
   – Evaluate impact of potential system designs and application mappings
   – Model-automated optimization of applications

• Enabling reproducible science despite non-determinism & asynchrony
   – Methods for validation on non-deterministic architectures
   – Detection and mitigation of pervasive faults and errors

• Facilitating Data Management, Analytics, and Workflows
   – Mapping of science workflows to heterogeneous hardware and software services
   – Adapting workflows and services to meet facility-level objectives through learning approaches

https://orau.gov/exheterogeneity2018/
https://doi.org/10.2172/1473756
Recap

• Recent trends in computing paint an ambiguous future (for HPC and broader community)
  – Contemporary systems provide evidence that power constraints are driving architectures to change rapidly (e.g., Dennard, Moore)
  – Multiple architectural dimensions are being (dramatically) redesigned: Processors, node design, memory systems, I/O

• Major transition point for computing
  – New devices
  – New architectures
  – New programming systems

• Complexity and uncertainty are ubiquitous

• Programming systems must provide performance portability (in addition to functional portability)!!

• In near term, rate of change will accelerate and grow more diverse

• Visit us
  – We host interns and other visitors year round
    • Faculty, grad, undergrad, high school, industry

• Jobs in FTG
  – Postdoctoral Research Associate in Computer Science
  – Software Engineer
  – Computer Scientist
  – Visit https://jobs.ornl.gov

• Contact me vetter@ornl.gov