"Supercomputing" is the best description of the future of HPC

James Reinders
(Intel Corporation, retired 2016),
Parallel Programming and HPC Enthusiast (and Expert)

Tuesday, Sept. 26, 2017
IXPUG Annual Fall Conference
Disclaimer

Opinions expressed today are purely my own, and do not necessarily represent those of any employers past, present, or future.
Making computer systems fast is my passion 😊

My greatest joy in life – is helping create super fast and reliable computers, for others to use to improve the world through science and engineering.

A passion that I continue to pursue every day!

Enjoy the Journey Together.
When performance matters, we need supercomputers.
Machines warp algorithms, Algorithms warp machines, rinse and repeat.
Machines warp algorithms, Algorithms warp machines, rinse and repeat.

Pick the BEST solution available today.
Machines warp algorithms, Algorithms warp machines, rinse and repeat.

Pick the BEST solution available today.

But, remember to reexamine when conditions change.

<< EASILY FORGOTTEN>>
What’s changed?

Clock rates stalled

Moore’s Law continues (but slowing – and the end is coming nearer)

Power concerns

New algorithms (Deep Learning)
What’s changed?

Clock rates stalled

Moore’s Law continues (but slowing – and the end is coming nearer)

Power concerns

New algorithms (Deep Learning)

What to reexamine?

Accelerators / New architectures

MCMs

New chip technologies

Application design / Performance portability
Performance: Embree vs. NVIDIA* OptiX*

Frames Per Second (Higher is Better), 1024x1024 image resolution

- Intel® Xeon® Processor E5-2699 v4
  2 x 22 cores, 2.2 GHz
- Intel® Xeon Phi® Processor 7250
  68 cores, 1.4 GHz
- NVIDIA TITAN X (Pascal)
  Coprocessor
  12 GB RAM

Embre 2.12.0, ICC 2016 Update 1, Intel® SPMD Program Compiler (ISPC) 1.9.1
NVIDIA OptiX 4.0.1, CUDA® 8.0.44

Source: Intel
Intel Xeon Phi

How much can you use flexibility + parallelism?
Hardware Wish:
Performance Possible

Software Wish:
Performance Easy
Hardware Wish: Performance Possible

Performance Portability: Performance Probable

Software Wish: Performance Easy
Performance and portability are important but often conflicting objectives.
Performance Portability
1. DEFINE the problem
2. SOLVE the problem
When performance matters, we need supercomputers.
When performance matters, we need supercomputers (not HPC).
I reject the notion that HPC and HPDA are fundamentally different.
We called them “supercomputers” far before they were labelled “high performance computing” (HPC).
“Accelerators” have been around for decades.
“Accelerators” have been around for decades

they came, and
then they we assimilated,
or they disappeared
“Accelerators” have been around for decades they came, and then they we assimilated, or they disappeared, over and over again.
Growth of processor clock rates over time.

1973 1MHz, 2003 1 GHz
2004 3 GHz, still today...

power wall + ILP wall + memory wall
solve: parallel hardware + explicit parallel software + software memory optimization

© 2017, James Reinders, used with permission. http://lotsofcores.com
Moore's Law

© 2017, James Reinders, used with permission. http://lotsofcores.com
NVidia popularized their GPUs as accelerators as clock rates climbs ended
NVidia popularized their GPUs as accelerators as clock rates climbs ended.

right time, right place, right product accelerators are here to stay.
Intel challenged NVidia’s claim that only GPUs owned the future with their first accelerator: a high-core-count CPU *Intel Xeon Phi*
Intel effectively joined the “accelerator” game and reiterated the value of CPUs at the same time.
Accelerators are here to stay

Forever complicating what a “computer” means
Accelerators are here to stay

BTW – Xeon Phi and GPUs are not the final answer
Not all AI is Deep Learning

• Most important today

• Amazingly capable, and yet seriously flawed in many ways

• What discoveries remain?
Machines warp algorithms, Algorithms warp machines, rinse and repeat.

Pick the BEST solution available today.

But, remember to reexamine when conditions change.

<< COMMONLY OVERLOOKED>>
Accelerators are here to stay

Oh yeah... what about us poor people working on software?
It’s the “system,” stupid.

ASCI Red:
Sandia National Laboratories

Number 1 system from June 1997 to June 2000

<table>
<thead>
<tr>
<th>Date</th>
<th>Cores</th>
<th>Linpack Peak</th>
<th>Theoretical Peak</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/97</td>
<td>7,264</td>
<td>1.068 Tflop/s</td>
<td>1.453 Tflop/s</td>
</tr>
<tr>
<td>11/97</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>6/98</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>11/98</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>6/99</td>
<td>9,472</td>
<td>2.1 Tflop/s</td>
<td>3.1 Tflop/s</td>
</tr>
<tr>
<td>11/99</td>
<td>9,632</td>
<td>2.4 Tflop/s</td>
<td>3.2 Tflop/s</td>
</tr>
<tr>
<td>06/00</td>
<td>9,632</td>
<td>2.4 Tflop/s</td>
<td>3.2 Tflop/s</td>
</tr>
</tbody>
</table>

Interconnect: Proprietary
Operating System: Paragon OS

Last appearance on list: No. 276 in November 2005
"caches do not belong on-die or on-package"

Solving Problem Case Study

Architecture of PA-8000

L1 cache on-die
L2 cache on-package
"caches do not belong on-die or on-package"

4" space on 2 boards for 2 processors vs. 1.25" rack space on 1 board for 2 processors

L1 cache on-die
L2 cache on-package

Architecture of PA-8000

System Bus
L2 3MB L313K Cache
Bus Interface Unit
L1 Cache

Instruction Pool

L1 Instruction Cache
Next IP
Instruction Encode
RTS
X86 Instructions
Executions to Machine-Code
Microcode Execution Sequences
Register File

Reservation Station
PART 1

Shipping Point (UP)
Interrupt
Jump EU
Address Generation
PART 2

Address Generation
Part 2

Part 3

Reclaim Buffer
Replacement Registers
ASCI Red:
Sandia National Laboratories

Number 1 system from June 1997 to June 2000

<table>
<thead>
<tr>
<th>Date</th>
<th>Cores</th>
<th>Unpack Peak</th>
<th>Theoretical Peak</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/97</td>
<td>7,264</td>
<td>1.068 Tflop/s</td>
<td>1.453 Tflop/s</td>
</tr>
<tr>
<td>11/97</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>6/98</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>11/98</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>6/99</td>
<td>9,472</td>
<td>2.1 Tflop/s</td>
<td>3.1 Tflop/s</td>
</tr>
<tr>
<td>11/99</td>
<td>9,632</td>
<td>2.4 Tflop/s</td>
<td>3.2 Tflop/s</td>
</tr>
<tr>
<td>06/00</td>
<td>9,632</td>
<td>2.4 Tflop/s</td>
<td>3.2 Tflop/s</td>
</tr>
</tbody>
</table>

Interconnect: Proprietary
Operating System: Paragon OS

Last appearance on list: No. 276 in November 2005
ASCI Red:
Sandia National Laboratories

Number 1 system from June 1997 to June 2000

Intel ASCI Red
Sandia National Laboratories, USA

<table>
<thead>
<tr>
<th>Date</th>
<th>Cores</th>
<th>Linpack Peak</th>
<th>Theoretical Peak</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/97</td>
<td>7,264</td>
<td>1.068 Tflop/s</td>
<td>1.453 Tflop/s</td>
</tr>
<tr>
<td>11/97</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>6/98</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>11/98</td>
<td>9,152</td>
<td>1.34 Tflop/s</td>
<td>1.83 Tflop/s</td>
</tr>
<tr>
<td>6/99</td>
<td>9,472</td>
<td>2.1 Tflop/s</td>
<td>3.1 Tflop/s</td>
</tr>
<tr>
<td>11/99</td>
<td>9,632</td>
<td>2.4 Tflop/s</td>
<td>3.2 Tflop/s</td>
</tr>
<tr>
<td>06/00</td>
<td>9,632</td>
<td>2.4 Tflop/s</td>
<td>3.2 Tflop/s</td>
</tr>
</tbody>
</table>

Interconnect: Proprietary
Operating System: Paragon OS

Last appearance on list: No. 276 in November 2005
Random quiz

CAFFEINE LOADING...
BUFFERING
PLEASE WAIT...
Coffee Rule of Thumb

1 shot
Slept
8-10 hrs
12 oz.

2 shots
Slept
5-7 hrs
16 oz.

3 shots
Slept
0-4 hrs
20 oz.
compiler writer's nightmare:

pipelined mode vs.
scalar mode
underpowered PPE; very hard to optimally program
on-chip accelerators are old news...

- floating-point
- memory management
- clock management/production/distribution
- serial ports, and other I/O ports
- DSP
- A/D conversion
and it continues today...
Oracle SPARC S7 and M7 processors

• on-chip Data Analytics Accelerator (DAX) engines
• silicon secured memory
• cryptographic instruction accelerators
• in-memory query acceleration
• in-line decompression
Enjoy the Journey Together.
module lightsequencer( enable, in3, out3 );
    input enable;
    input [2:0] in3;
    output [2:0] out3;
    assign out3[0] = (in3[2:0] > 4) && !enable;
endmodule
Microsoft Outlines Hardware Architecture for Deep Learning on Intel FPGAs

At Build, Microsoft's annual developers conference, taking place this week, Microsoft Azure CTO Mark Russinovich disclosed major advances in Microsoft's hyperscale deployment of Intel® field programmable gate arrays (FPGAs). These advances have resulted in the industry's fastest public cloud network and new technology for acceleration of Deep Neural Networks (DNNs) that replicate "thinking" in a manner that's conceptually similar to that of the human brain.

Intel Launches Software Tools to Ease FPGA Programming

September 5, 2017

Field Programmable Gate Arrays (FPGAs) have a reputation for being difficult to program, requiring expertise in specialty languages like Verilog or VHDL. The programming burden is key to unlocking broader adoption for FPGAs and is a prime goal of FPGA vendors, like Intel.

Yesterday Intel, which purchased FPGA company Altera in 2015, announced a new set of software tools aimed at making FPGA programming accessible to mainstream developers. It's part of Intel's strategy to boost FPGA use in mainstream applications. The datacenter, where large workloads include high-performance computing, artificial intelligence, data and video analytics, and 5G network processing, is a prime target.

The three tools launched by Intel are:

1. Altera FPGA Compiler: A new software tool that simplifies FPGA programming by automatically generating optimized code from high-level descriptions.
2. Intel FPGA deployment kit: A set of tools that simplify the deployment of FPGA-based applications into production environments.
3. Intel FPGA development kit: A software development kit that provides access to a wide range of FPGA-based applications and tools for developing FPGA-based applications.

These tools are designed to make FPGA programming more accessible to mainstream developers and to accelerate the adoption of FPGAs in a variety of applications.
IBM TrueNorth Board
(16 chips)
BREAKING CONSTRAINTS OF MOORE’S LAW

- Revolutionary Infinity Fabric
- High-performance, scalable links
- Enables architectural innovations that increase real-world performance
- Improves product yields
- Reduces product costs

AMD dubbed this a “Purpose-built MCM architecture”
Intel’s “Broadwell Proof of Concept” (an MCM)

**INTEL® STRATIX® 10 FPGA CUSTOM HARDWARE**

A New Level of Cloud Performance for Real-Time AI Computation

Disrupting AI with Record Low Latency, Performance and Batch-Free Executions

**Energy Efficient Inference with Infrastructure Flexibility**
- Using DLA Library provides reconfigurable accelerator for variety of workloads and topologies
- Enable custom solutions with inline analytics for lower latency solutions
MCM-GPU: Multi-Chip-Module GPUs for Continued Performance Scalability

Historically, improvements in GPU-based high performance computing have been tightly coupled to transistor scaling. As Moore’s law slows down, and the number of transistors per die no longer grows at historical rates, the performance curve of single monolithic GPUs will ultimately plateau.

Publication Date: Monday, June 26, 2017
Published in: IEEE/ACM International Symposium on Computer Architecture (ISCA)
Accelerators work best when code is structured to take advantages in specific focused parts of an application.

There in lies the rub: they don’t need the same things.
The future will have lots of diversity in what we can accelerate in hardware.
Hardware Diversity

- **IC**
  - fully custom
  - Integrated Circuit

- **ASIC**
  - Application Specific
  - Integrated Circuit

- **FPGA**
  - Field Programmable
  - Gate Arrays

More flexible, greater density/complexity

Faster turnaround
Acceleration Diversity
Acceleration Diversity

**VPU**
Vector Processing Unit (IC)

**CPU**
Central Processing Unit (IC)

**GPU**
Graphic Processing Unit (IC)

**TPU**
Tensor Processing Unit (ASIC)
Acceleration Diversity

**VPU**
Vector Processing Unit (IC)

**CPU**
Central Processing Unit (IC)

**GPU**
Graphic Processing Unit (IC)

**TPU**
Tensor Processing Unit (ASIC)

**NPU**
Neuromorphic Processing Unit (IC)
Acceleration Diversity

**VPU**
Vector Processing Unit (IC)

**CPU**
Central Processing Unit (IC)

**GPU**
Graphic Processing Unit (IC)

**TPU**
Tensor Processing Unit (ASIC)

**NPU**
Neuromorphic Processing Unit (IC)

**Proprietary**
Customer Specific Algorithms (FPGA)
Acceleration Diversity

**VPU**
Vector Processing Unit (IC)

**CPU**
Central Processing Unit (IC)

**GPU**
Graphic Processing Unit (IC)

**TPU**
Tensor Processing Unit (ASIC)

**NPU**
Neuromorphic Processing Unit (IC)

**Proprietary**
Customer Specific Algorithms (FPGA)

**Optimization**
Quantum Computing (IC)
dreaming...

Standards to be interchangeable?
DARPA Common Heterogeneous Integration and IP Reuse Strategies (CHIPS)
Integration: Enabling IP and Chiplet Re-use

- Develop a pre-defined “common” interposer (SiC/Si/Glass) platform
- Populate common platform with library of chiplets of IP/circuit blocks
- Different complex configurations can be formed rapidly with reuseable IP blocks/chiplets

DAHI-enabled integration technology plus IP re-use ecosystem to speed the design cycle and reduce the access cost

The DARPA CHIPS “program seeks to establish a new paradigm in IP reuse.”

DARPA-BAA-16-62
what about software?
Performance Portability is demonstrated when a similar percent of achievable peak performance is obtained on a range of machines with the same code.
Accelerators: Increasing diversity and availability – how can you use them?
Memory Hierarchy: locality is good data movement is bad
Parallelism: coarser grained parallelism is not embarrassing
I hold these truths to be self-evident

- Do not move data... unless it will pay off to copy/move
- Do in parallel... unless it will pay off to synchronize
I hold these truths to be self-evident

- Do not move data... unless it will pay off to copy/move
- Do in parallel... unless it will pay off to synchronize

Of course, power consumption is one way things can “pay off” (performance, perf/watt, etc.)
This future is both, not “either”
Highly Parallel CPUs + Accelerators

**Intel Xeon Phi**
(highly parallel CPU)

How much can you use flexibility + parallelism?

<table>
<thead>
<tr>
<th>VPU</th>
<th>GPU</th>
<th>NPU</th>
<th>Proprietary</th>
<th>Optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vector</td>
<td>Graphic</td>
<td>Neuromorphic</td>
<td>Customer Specific</td>
<td>Quantum Computing</td>
</tr>
<tr>
<td>Processing</td>
<td>Processing</td>
<td>Processing</td>
<td>Algorithms (FPGA)</td>
<td>(IC)</td>
</tr>
<tr>
<td>Unit (IC)</td>
<td>Unit (IC)</td>
<td>Unit (IC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(IC)</td>
<td>(IC)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Skills needed in BOTH – for the best software solutions
This future is both, not “either”
Highly Parallel CPUs + Accelerators

**Intel Xeon Phi**
(highly parallel CPU)

How much can you use flexibility + parallelism?

**VPU**
Vector Processing Unit (IC)

**GPU**
Graphic Processing Unit (IC)

**NPU**
Neuromorphic Processing Unit (IC)

**Proprietary**
Customer Specific Algorithms (FPGA)

**Optimization**
Quantum Computing (IC)

Skills needed in BOTH – for the best software solutions
This future is both, not “either”
Highly Parallel CPUs + Accelerators

It always has been BOTH.
Any appearance otherwise is fleeting.

Skills needed in BOTH – for the best software solutions
Accelerators will be assimilated. Resistance is futile.
Accelerators will be assimilated. Resistance is futile.

Supercomputers are the assimilators.
Accelerators will be assimilated. Resistance is futile.

Supercomputers are the assimilators.

Performance portability will be most possible with the best system+software architectures.
Thank you.

Enjoy the Journey Together.