End User Update: High-Performance Reconfigurable Computing

Tarek El-Ghazawi

Director, GW Institute for Massively Parallel Applications and Computing Technologies (IMPACT)
Co-Director, NSF Center for High-Performance Reconfigurable Computing (CHREC)

The George Washington University
hpcl.gwu.edu
Paul Muzio’s Outline!

◆ Performance
  – What hardware accelerators are you using/evaluating?
  – Describe the applications that you are porting to accelerators?
  – What kinds of speed-ups are you seeing (provide the basis for the comparison)?
  – How does it compare to scaling out (i.e., just using more X86 processors)?
  – What are the bottlenecks to further performance improvements?

◆ Economics
  – Describe the programming effort required to make use of the accelerator.
  – Amortization
  – Compare accelerator cost to scaling out cost
  – Ease of use issues

◆ Futures
  – What is the future direction of hardware based accelerators?
  – Software futures?
  – What are your thoughts on what the vendors need to do to ensure wider acceptance of accelerators?
Why Accelerators: A Historical Perspective

Performance

Vector Machines

Massively Parallel Processors

MPPs with Multicores and Heterogeneous Accelerators

Time

1993- HPCC

2006- End of Moore’s Law in Clocking!

Hopes are in Architecture!

Tarek El-Ghazawi, GWU

HPC User Forum, Roanoke

April 21, 2008
Which Accelerators?

- We considered HPRCs more than anything else
  - To be addressed today
- We are increasingly using GPUs
- Some Cell
High-Performance Reconfigurable Computing (HPRC)

- IEEE Computer, March 2007

High-Performance Reconfigurable Computers are parallel computing systems that contain multiple microprocessors and multiple FPGAs. In current settings, the design uses FPGAs as coprocessors that are deployed to execute the small portion of the application that takes most of the time—under the 10-90 rule, the 10 percent of code that takes 90 percent of the execution time.
An Architectural Classification for Hardware Accelerated High-Performance Computers

Uniform Nodes Non-Uniform System (UNNS)

HPRC Examples: SRC 6/7, SGI Altix/RC100 Systems
Non-Uniform Nodes
Uniform System (NNUS)

IN and/or GSM

HPRC Example: Cray XD1, Cray XT5h
Applications and Performance

Cryptography, Remote Sensing and Bioinformatics
Hyperspectral Dimension Reduction

◆ Multi-Spectral Imagery →
10’s of bands (MODIS ≡ 36 bands, SeaWiFS ≡ 8 bands, IKONOS ≡ 5 bands)

◆ Hyperspectral Imagery →
100’s-1000’s of bands (AVIRIS ≡ 224 bands, AIRS ≡ 2378 bands)
  – Challenges (Curse of Dimensionality)
  – Solution
    ✷ Dimension Reduction
Hyperspectral Dimension Reduction (Techniques)

- **Principal Component Analysis (PCA):**
  - Most Common Method Dimension Reduction
  - Complex and Global computations: difficult for parallel processing and hardware implementations

- **Wavelet-Based Dimension Reduction***:
  - Simple and Local Operations
  - High-Performance Implementation

Wavelet-Based Dimension Reduction
(Execution Profiles on SRC)

Total Execution Time = 20.21 sec (Pentium4, 1.8GHz)
Speedup = 24.06 x (without-streaming)
Speedup = 32.04 x (with-streaming)

Total Execution Time = 1.67 sec (SRC-6E, P3)
Speedup = 12.08 x (without-streaming)
Speedup = 13.21 x (with-streaming)

Total Execution Time = 0.84 sec (SRC-6)
Speedup = 24.06 x (without-streaming)
Speedup = 32.04 x (with-streaming)
Cloud Detection

Band 2 (Green Band)  Band 3 (Red Band)  Band 4 (Near-IR Band)  Band 5 (Mid-IR Band)

Band 6 (Thermal IR Band)  Software/Reference Mask  Hardware Floating-Point Mask (Approximate Normalization)  Hardware Fixed-Point Mask (Approximate Normalization)
Protein and DNA Matching-The Scoring Matrix

\[
F(i, j) = \max \begin{cases} 
F(i-1, j-1) + s(x_i, y_j) \\
F(i-1, j) + \text{gap\_penalty} \\
F(i, j-1) + \text{gap\_penalty} \\
0 \end{cases}
\]

<table>
<thead>
<tr>
<th></th>
<th>-</th>
<th>T</th>
<th>T</th>
<th>T</th>
<th>C</th>
<th>A</th>
<th>T</th>
<th>A</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td>16</td>
</tr>
<tr>
<td>G</td>
<td>2</td>
<td>-1</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>13</td>
</tr>
<tr>
<td>G</td>
<td>4</td>
<td>-1</td>
<td>-2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>T</td>
<td>6</td>
<td>-3</td>
<td>0</td>
<td>-2</td>
<td>-1</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>T</td>
<td>8</td>
<td>-5</td>
<td>2</td>
<td>0</td>
<td>-3</td>
<td>-2</td>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>A</td>
<td>10</td>
<td>-7</td>
<td>4</td>
<td>1</td>
<td>-3</td>
<td>-3</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>12</td>
<td>-9</td>
<td>6</td>
<td>3</td>
<td>0</td>
<td>-2</td>
<td>-3</td>
<td>-4</td>
<td>-2</td>
</tr>
<tr>
<td>G</td>
<td>14</td>
<td>11</td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>-1</td>
<td>-3</td>
<td>-4</td>
<td>-4</td>
</tr>
<tr>
<td>C</td>
<td>16</td>
<td>13</td>
<td>10</td>
<td>7</td>
<td>4</td>
<td>1</td>
<td>-2</td>
<td>-4</td>
<td>-5</td>
</tr>
</tbody>
</table>
# Savings of HPRC
(Based on one Altix 4700 10U rack)

<table>
<thead>
<tr>
<th>Application</th>
<th>Speedup</th>
<th>SAVINGS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Cost Savings</td>
</tr>
<tr>
<td>Smith-Waterman (DNA Sequencing)</td>
<td>8723</td>
<td>22x</td>
</tr>
<tr>
<td>DES Breaker</td>
<td>38514</td>
<td>96x</td>
</tr>
<tr>
<td>IDEA Breaker</td>
<td>961</td>
<td>2x</td>
</tr>
<tr>
<td>RC5(32/12/16) Breaker</td>
<td>6838</td>
<td>17x</td>
</tr>
</tbody>
</table>

**Assumptions**

- 100% cluster efficiency
- Cost Factor $\mu$P : RP $\rightarrow$ 1 : 400
- Power Factor $\mu$P : RP $\rightarrow$ 1 : 11.2
  - 1 10U Rack: 1230 W
  - $\mu$P board (with two $\mu$Ps): 220 W
- Size Factor $\mu$P : RP $\rightarrow$ 1 : 34.5
  - Cluster of 100 $\mu$Ps = four 19-inch racks
    - footprint = 6 square feet
  - Reconfigurable computer (10U)
    - footprint = 2.07 square feet
## Savings of HPRC
*(Based on one Cray-XD1 chassis)*

<table>
<thead>
<tr>
<th>Application</th>
<th>Speedup</th>
<th>SAVINGS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Cost Savings</td>
</tr>
<tr>
<td>Smith-Waterman (DNA Sequencing)</td>
<td>2794</td>
<td>28x</td>
</tr>
<tr>
<td>DES Breaker</td>
<td>12162</td>
<td>122x</td>
</tr>
<tr>
<td>IDEA Breaker</td>
<td>2402</td>
<td>24x</td>
</tr>
<tr>
<td>RC5(32/8/8) Breaker</td>
<td>2321</td>
<td>23x</td>
</tr>
<tr>
<td>Hyperspectral Dimension Reduction</td>
<td>125</td>
<td>1x</td>
</tr>
<tr>
<td>Cloud Detection</td>
<td>110</td>
<td>1x</td>
</tr>
</tbody>
</table>

### Assumptions

- 100% cluster efficiency
- Cost Factor $\mu P : RP \rightarrow 1 : 100$
- Power Factor $\mu P : RP \rightarrow 1 : 20$
  - Reconfigurable processor (based on one XD1 Chassis): 2200 W
  - $\mu P$ board (with two $\mu Ps$): 220 W
- Size Factor $\mu P : RP \rightarrow 1 : 95.8$
  - Cluster of 100 $\mu Ps = four 19\text{-inch racks}$
    - footprint = 6 square feet
  - Reconfigurable computer (one XD1 Chassis)
    - footprint = 5.75 square feet
## Savings of HPRC (Based on SRC-6)

<table>
<thead>
<tr>
<th>Application</th>
<th>Speedup</th>
<th>SAVINGS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Cost Savings</td>
</tr>
<tr>
<td>Smith-Waterman (DNA Sequencing)</td>
<td>1138</td>
<td>6x</td>
</tr>
<tr>
<td>DES Breaker</td>
<td>6757</td>
<td>34x</td>
</tr>
<tr>
<td>IDEA Breaker</td>
<td>641</td>
<td>3x</td>
</tr>
<tr>
<td>RC5(32/12/16) Breaker</td>
<td>1140</td>
<td>6x</td>
</tr>
<tr>
<td>Hyperspectral Dimension Reduction</td>
<td>32</td>
<td>0.16x</td>
</tr>
<tr>
<td>Cloud Detection</td>
<td>28</td>
<td>0.14x</td>
</tr>
</tbody>
</table>

### Assumptions
- 100% cluster efficiency
- Cost Factor $\mu P : RP \rightarrow 1 : 200$
- Power Factor $\mu P : RP \rightarrow 1 : 3.64$
  - Reconfigurable processor (based on SRC-6): 200 W
  - $\mu P$ board (with two $\mu P$s): 220 W
- Size Factor $\mu P : RP \rightarrow 1 : 33.3$
  - Cluster of 100 $\mu P$s = four 19-inch racks
    - footprint = 6 square feet
  - Reconfigurable computer (SRC MAPstation™)
    - footprint = 1 square feet
Programming
Historical Perspective

Technology

In-Socket Accelerators
(65 nm)

Embedded Processors
& Transceivers
(90 nm)

DSP Slice & Dual Port
Block RAM
(130 nm)

Logic Fabric
(180 nm)

Users

Domain Scientists

RC-Aware Domain Scientists

Embedded Software Engineers

Embedded System Designers

DSP & Networking Designers

Circuit Designers

Platform Specifications & Parallel SW Languages

Improved-HLLs

HW/SW Codesign

Embedded DSP IDEs

IP Core Generators

HLLs

RTL

Schematics

PCF

HPCs

Custom Comp.

Glue Logic

Glue Logic

Custom Comp.

Embedded Processors
& Transceivers

HPRC

PSOC

IDataBases

Applications
Programming HPReCs

HLLs

Imperative

Data Flow

Functional

Implicit Parallelism

Explicit Parallelism

Text-Based

Graphical-Based

Examples

Graphical

Examples

Streams-C
Impulse-C
Handel-C
Carte-C

HDLs
Mitrion-C
SA-C

Ptolemy II
SysGen
DSPL\text{Logic}
Corefire
Viva
Productivity Analysis of Existing Tools

- **Tools considered**
  - Impulse-C
  - Handel-C
  - Carte-C
  - Mitrion-C
  - SysGen
  - RC-Toolbox
  - HDLs

- **Utility**
  - Frequency
  - Area

- **Cost**
  - Acquisition time
    - Learning time
  - Development time

Results excerpted from GWU papers in SPL’07 and FPT’07 conferences.

Tarek El-Ghazawi, GWU
Future Hardware Development

- More use of socket-based integration
- Better integration with memory hierarchy
- Better accelerators, in the FPGA side
  - More computationally oriented/Floating-point cores?
  - Coarser grain FPGAs?
- On-chip FPGAs and accelerators?
Early 1970s  
First Vector and SIMD Systems  
• CDC STAR-100  
• TI ASC  
• ILLIAC IV

1971-78  
First MIMD System  
• CMU C.mmp (16 PDP11s)

1985  
FPGA  
Xilinx

1996-1998  
SIMD  
Altivec  
By Apple, IBM, and Motorola

1998  
HPRC SRC

2001  
Vector Processor/SIMD  
CELL BE  
GPGPUs  
NVIDIA and AMD  
Multicore CPUs  
IBM Power 4

Coming soon?  
Hybrid-Reconfigurable/ Chip?  
Accelerators as cores?
Future Software

- More user/application-centric programming
- Unified parallel programming interface?
- More efficient compiling
- Tools for accelerator-GPP application co-design
- Virtualization for ease-of-use and portability

HELP MAY BE COMING?
DARPA Studies

- DARPA is looking at bridging productivity gap for FPGAs
- NSF CHREC Schools (UF, GWU) and (BYU, VT) conducted a DARPA study
- DARPA has at least one more ongoing study
- Are we going to see any BAAs?
Exploration of a Research Roadmap for Application Development & Execution on FPGA-based Systems

Roadmap Exploration Achievement

Main Achievement:
- Project will develop a comprehensive research roadmap for challenges in bridging semantic gap
- Roadmap will lay foundation for solution space enabling significant productivity gains while attaining significant fraction of peak performance
- Recommendations will cite where in this solution space that potential DARPA investments can be expected to have greatest impact and value

How It Works:
1. Survey of DOD use cases (current & future) and state of existing tools
2. Characterization of limitations of existing flows, analysis of key technical challenges, and identification of potential solutions
3. Preliminary analysis and evaluation of solution space and projected impact

Assumptions and Limitations:
Spectrum of application development phases
A. Formulation (algorithm and architecture design exploration, mappings, performance prediction)
B. Design (linguistic and graphical design semantics & syntax, HW/SW codesign)
C. Translation (compilation, libraries & linkage, technology mapping such as synthesis, PAR)
D. Execution (test & verification, performance analysis & optimization, run-time services)

End-of-Phase Goal
- Proposed roadmap will include quantitative estimates of potential impact attainable
- Example: design method X on selected DOD use cases can achieve 90% efficiency of HDL design in 10% of design time

New Insights
- Semantic gap spanning app development & hardware design is #1 challenge of RC for DOD
- Research roadmap is a critical need for addressing challenge
- DOD-oriented NSF Center in RC (CHREC) ideally suited to explore

FPGA-based reconfgurable computing is a promising new paradigm for DOD
- Pro: performance, power, size, weight, etc.
- Con: broad semantic gap for app development

Bridging the app/arch semantic gap will enable RC technologies to revolutionize DOD missions.
Conclusions

◆ Lots of common issues among accelerators
◆ For the applications that they can do well, they do really well!
◆ FPGAs were not built originally for computing
  – Limited applications
  – Less than user friendly interfaces
  – Very long time for compiling
◆ Programming languages expose a restrictive view of the system and are often hardware oriented,
  – Need a single system wide language paradigm
◆ A major bottleneck is data transfer rates between the microprocessor and the FPGA
◆ More work is needed on how to manage heterogeneity
  – Virtualization for portability and ease of use
  – Advanced programming models based on parallel computing
  – New tools for performance tuning and debugging in heterogeneous environments
  – Better integration into memory hierarchy
◆ The above requires fundamental work that will be unlikely supported by vendors alone, it needs a for example DARPA Driven Industry/University effort (like HPCS)