FPGAs for Accelerated Computing

- What drives FPGA use in embedded and HPC systems?
  - Performance/power relative to standard processors and DSPs
  - Time-to-market and development costs relative to ASICs
  - System integration – complete systems on programmable logic

- What limits FPGA use?
  - PRODUCTIVITY – difficulty of programming FPGA platforms

The solution? Software-to-hardware via C-to-FPGA tools!
Why C-to-FPGA Tools?

Moving C applications to FPGAs

- Desktop applications
- Embedded processing
- Signal processing
- Image processing
- Etc

Software Coprocessing

Systems on FPGA

Complex Control

Control Logic

Glue Logic

Hardware accelerated coprocessing from C-language

Increasing FPGA Complexity...

What is Impulse C?

- Not a new language!
  - Based on standard ANSI C
  - Provides a software level of abstraction

- ANSI C for FPGA programming
  - Supports standard C development tools
  - Supports multi-process partitioning
  - Compatible with a wide range of FPGA-based computing platforms

- A software-to-hardware compiler
  - Optimizes C code for parallelism
  - Generates HDL, ready for FPGA synthesis
  - Includes interactive tools for optimization
  - Also generates hardware/software interfaces
Uses for Impulse C

1. Create a hardware module

2. Accelerate an embedded CPU

3. Accelerate an external/host CPU
Who Uses Impulse C?

Panasonic
Fujitsu
Sanyo
Vigilant
DRA
Hitachi
Aisin
General Dynamics
Toshiba
Sharp
NEC
Canon
Schut
Toyota
Intel
Pioneer
FUJI XEROX
Advantest
Draper
Lockheed Martin
ARXAN
Fukuda Denshi
EADS
MNB Technologies
Polycom
National Security Technologies

And many more!

www.ImpulseC.com
Types of Applications

- Image Processing and Embedded Systems
  - Data compression and secure communications
  - Medical diagnostics and forensic image analysis
  - Robotics and machine vision
  - DSP and control applications

- High Performance Computing
  - Financial modeling and arbitrage
  - Bioinformatics and scientific computing
  - Cryptography and secure communications
  - Network traffic analysis and filtering
  - National security
Supporting Iterative Methods

*Bioinformatics example: Multiple Sequence Alignment*

<table>
<thead>
<tr>
<th>Processor/FPGA</th>
<th>Clock Rate</th>
<th>Execution time (seconds)</th>
<th>FPGA slices (Virtex-4 LX25)</th>
<th>Acceleration vs. MicroBlaze</th>
<th>Acceleration vs. 2GHz PC</th>
</tr>
</thead>
<tbody>
<tr>
<td>MicroBlaze with on-chip BRAM</td>
<td>100MHz</td>
<td>41.3</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Pentium PC</td>
<td>2GHz</td>
<td>.454</td>
<td></td>
<td>91X</td>
<td>1</td>
</tr>
<tr>
<td>FPGA, 1 PE</td>
<td>100MHz</td>
<td>.209</td>
<td>1076 (10%)</td>
<td>198X</td>
<td>2.2X</td>
</tr>
<tr>
<td>FPGA, 2 Pes</td>
<td>100MHz</td>
<td>.145</td>
<td>1509 (14%)</td>
<td>285X</td>
<td>3.1X</td>
</tr>
<tr>
<td>FPGA, 8 Pes</td>
<td>100MHz</td>
<td>.102</td>
<td>4229 (39%)</td>
<td>405X</td>
<td>4.5X</td>
</tr>
<tr>
<td>FPGA, 2 optimized processes, each with 2 PEs</td>
<td>100MHz</td>
<td>.074</td>
<td>5118 (47%)</td>
<td>558X</td>
<td>6.1X</td>
</tr>
</tbody>
</table>
See a Demonstration at RSSI

- C-based design
- Desktop simulation
- C-to-FPGA compilation
- Interactive optimization

www.ImpulseC.com