Evaluating FPGAs For Communication Infrastructure Applications

Presentation Goals

By the end of this workshop, you should know:
- Key processor-selection criteria and trends for communication infrastructure
- Key strengths and weaknesses of high-end DSPs
- Key strengths and weaknesses of high-end FPGAs
- How typical DSPs and FPGAs stack up in terms of performance and cost/perf.
Evaluating FPGAs For Communication Infrastructure Applications

Generalized Comm System

- Signal In
- Source Coding
- Channel Coding
- Modulation
- Multi. Access
- Receiver
- Inverse Channel Coding
- Source Decode
- Signal Out
- Encryption, Decryption
- Transmitter
- Multi. Access
- Detection, Demodulation
- Parameter Estimation

Systems: Two Types

Infrastructure
- Examples: base stations, central office equipment, cable “head-end”

Terminals
- Portable
  - Battery-powered, size-constrained
  - Examples: cellular phone, mobile media player, PDA
- Non-portable (e.g., “CPE”)
  - Examples: set-top box, home media server
Terminal Requirements

Key criteria
• Sufficient performance
• Cost
• Energy efficiency
• Memory use
• Small-system integration support
• Packaging
• Tools
• Application-development infrastructure
• Chip-product roadmap

Infrastructure Requirements

Key criteria
• Board area per channel
• Power per channel
• Cost per channel
• Large-system integration support
• Tools
• Application-development infrastructure
• Architecture roadmap
  • Compatibility, multi-vendor support
### Key Processing Technologies

<table>
<thead>
<tr>
<th>DSPs</th>
<th>Massively parallel processors</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPPs/DSP-enhanced GPPs</td>
<td>ASSPs</td>
</tr>
<tr>
<td>Reconfigurable architectures</td>
<td>ASICs</td>
</tr>
<tr>
<td>- <strong>FPGAs</strong></td>
<td>- Licensable cores</td>
</tr>
<tr>
<td></td>
<td>- Customizable cores</td>
</tr>
<tr>
<td></td>
<td>- Platform-based design</td>
</tr>
<tr>
<td>Reconfigurable processors</td>
<td></td>
</tr>
</tbody>
</table>

**Massively parallel processors**

### DSPs: The Incumbents

Modern conventional DSPs introduced ~1986
- One instruction, one MAC per cycle
- Developed primarily for telecom applications

High-performance VLIW DSPs introduced ~1997
- Developed primarily for wireless infrastructure
- Speed focused:
  - Independent execution units support many instructions, MACs per cycle
  - Deeper pipelines and simpler instruction sets support higher clock rates
  - Emphasis on compilability
Example: StarCore SC140

- 6-issue 16-bit fixed-point architecture
  - Up to four 16-bit MACs per cycle
- Motorola MSC8101 (one SC140 core) shipping at 300 MHz, $116 (1 ku)

Motorola MSC8101

- PowerPC Bus (100 MHz)
- DMA Controller
- Memory Controller
- SC140 Core
- Filter Coprocessor
- 512 KB SRAM
- CPM
  - ATM
  - Ethernet
  - UTOPIA
  - E1/T1 E3/T3
- HDLC
- UART
- I²C
- SPI
- Addr. (32-bit)
- Data (64-bit)
Other Infrastructure DSPs

Texas Instruments TMS320C64x
- 8-issue 16-bit fixed-point architecture
  - Up to four 16-bit MACs per cycle
  - Special instructions and co-processors for communications
  - Compatible with 'C62x, 'C67x
- Sampling at 720 MHz, $216 (1 ku)
  - Shipping at 600 MHz, $108 (1 ku)

Analog Devices TigerSHARC (ADSP-TS20x)
- 4-issue fixed- and floating-point
  - Up to eight 16-bit fixed-point MACs per cycle
  - Special instructions for 3G base stations
  - High memory bandwidth (18 GB/s)
- Sampling at 600 MHz, $334 (1 ku)
  - TS101 shipping at 300 MHz, $234 (1 ku)

DSP Processors

Strengths and Weaknesses

- DSP performance, efficiency strong compared with other types of off-the-shelf processors
- But may not be adequate for demanding tasks
  - Fixed architectures limit efficiency, design flexibility
  - Centralized computation and extensive indirection reduce efficiency
- Relatively limited selection of chips per family
- But products offer strong, relevant integration
**DSP Processors**

*Strengths and Weaknesses*

- Relatively low development cost, risk
  - Mature technology
  - Large, experienced developer base
  - Fast time-to-market
  - Some architectures available from multiple vendors
    But some vendors' roadmaps are unclear or uncertain

---

**Why Consider Alternatives?**

**Convergence**
- DSP-intensive products increasingly include complex non-DSP functionality

**Processing throughput, density**
- E.g., 3G wireless computation demands outstripping DSP processor advances

**Development**
- DSP processor software development tools (e.g., compilers) have significant limitations

**Cost**
- Desire for integration drives SoC approach

**Energy efficiency**

**Flexibility**
Wireless Bandwidth Growth

2G
- GSM
- DSC1800
- PCS1900
- IS-95B
- IS-54B
- IS-136
- PDC

2.5G
- GPRS
- HCSD
- IS-95C
- IS-136+
- IS-136 HS
- Compact EDGE

3G
- 3GPP-DS-FDD
- 3GPP-DS-TDD
- 3GPP-MC
- ARIB W-CDMA
- IS-2000 CDMA
- IS-95-HDR

Wireless
Bandwidth
Growth

NARROWBAND
CIRCUIT
VOICE

8-13 Kbps

64-384 Kbps

384-2000+ Kbps

WIDEBAND
PACKET
DATA

~100 MIPS

~10,000 MIPS

~100,000 MIPS

Are Processors Efficient?

The Monarchial Model of Computing

Steps for performing one basic operation:
- Fetch instruction from memory
- Decode instruction
- Compute address
- Fetch data
- (Off-chip memory \(\rightarrow\) L2, update cache state)
- (L2 \(\rightarrow\) L1, update cache state)
- L1 \(\rightarrow\) registers
- Registers \(\rightarrow\) arithmetic unit
- Perform desired operation
- Write result
- Compute address, access hierarchy
- Update data pointers
- Update program counter

© 2003 Berkeley Design Technology, Inc.
FPGAs

Field-Programmable Gate Arrays

An amorphous “sea” of reconfigurable logic with reconfigurable interconnect
- Possibly interspersed with fixed-logic resources, e.g., processors, multipliers
Potential for very high parallelism
Historically used for prototyping and “glue logic,” but becoming more sophisticated
- DSP-oriented architecture features
- DSP-oriented tools and design libraries
  - Viterbi, Turbo, and Reed-Solomon coders and decoders, FIR filters, FFTs,...
Key DSP players: Altera and Xilinx

Altera Stratix

Up to 28 hard-wired “DSP blocks”
- 8×9-bit, 4×18-bit, 1×36-bit multiply operations
- Optional pipelining, accumulation, etc.
Three sizes of hard-wired memory blocks

© 2003 Berkeley Design Technology, Inc.
Altera Stratix

High-end, DSP-enhanced FPGAs

IP blocks
- Filters, FFTs, Viterbi decoders,...
- Nios processor
- Third-party IP, e.g., DMA controllers

DSP tools
- Parameterized IP block generators
- Simulink to FPGA link
- C+Simulink to FPGA design flow

Most family members available now
Prices begin at $170 (1 ku)
Evaluating FPGAs For Communication Infrastructure Applications

**Xilinx**

"Virtex" line of FPGAs

Virtex-II
- Includes array of hard-wired $18 \times 18$ multipliers plus distributed memory
- Up to 168 multipliers in biggest chip
- Most versions shipping now

Virtex-II Pro: joint effort with IBM
- Adds up to four hard-wired PowerPC 405 cores
- Up to 216 multipliers in biggest chip
- Most versions shipping now

Prices begin at $169 (1 ku)

---

**Xilinx**

Soft IP blocks; e.g.,
- Reed-Solomon encoder, Viterbi decoder, turbo decoder
- ARC processor, MicroBlaze CPU

Sophisticated "Core Generator" tool for generating parameterized IP blocks

Simulink to FPGA link via "System Generator"
Performance Analysis

- Comparing performance of off-the-shelf DSPs to that of FPGAs is tricky
- The common MMACS metric is oversimplified to the point of absurdity
  - FPGA vendors use distributed-arithmetic benchmarks that require fixed coefficients
  - MMACS metric overlooks need to dedicate resources to non-MAC tasks
  - MMACS metric ignores memory bandwidth needed to feed MACs
  - Many important DSP algorithms don’t use MACs at all!
Alternative Approach: Application Benchmarks

Use a full application, e.g., N channels of an OFDM receiver

Hazards:
- Applications tend to be ill-defined
- Hand-optimization usually required in real-world applications
  - Costly, time-consuming to implement
  - Evaluates programmer as much as processor
  - What is a “reasonable” benchmark implementation?

Solution: Simplified Application Benchmark

BDTI’s benchmark is based on a simplified OFDM receiver
- Closely resembles a real-world application
- Simplified to enable optimized implementations
- Constrained to ensure consistent, reasonable implementation practices

Benchmark implementer goals:
- Maximize number of channels
- Minimize cost per channel
**Benchmark Overview**

Flexibility is an asset:

- Algorithms range from table look-ups to MAC-intensive transform
- Data sizes range from 4 to 16 bits
- Data rates range from 40 to 320 MB/s
- Data includes real and complex values

```
IQ Demodulator -> FIR -> FFT -> Slicer -> Viterbi Decoder
```

---

**Benchmark Requirements**

“Pins to pins”

- Real-time throughput
- Bit-exact output data

Resource sharing is permitted

```
Channel 1
Channel 2
Channel 3
Channel 4
Channel 5
Channel 6
Channel 7
Channel 8
```

```
FIR 8 ch. -> FFT 4 ch. -> Slicer 4 ch. -> Viterbi 2 ch.
FIR 8 ch. -> FFT 4 ch. -> Slicer 4 ch. -> Viterbi 2 ch.
```
Evaluating FPGAs For Communication Infrastructure Applications

<table>
<thead>
<tr>
<th></th>
<th>Motorola MSC8101 (300 MHz)</th>
<th>Altera Stratix 1S20-6 (Preliminary)</th>
<th>Altera Stratix 1S80-6 (Preliminary)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channels</td>
<td>&lt;&lt;1</td>
<td>~10</td>
<td>~50</td>
</tr>
<tr>
<td>Cost (1 ku)</td>
<td>$116</td>
<td>$325</td>
<td>$3,480</td>
</tr>
</tbody>
</table>

From BDTI’s report, *FPGAs for DSP*.

Density Comparison

© 2003 Berkeley Design Technology, Inc.
FPGAs

**Strengths and Weaknesses**

† Massive performance gains on some algorithms
† Architectural flexibility can yield efficiency
   † Adjust data widths throughout algorithm
   † Parallelism where you need it
   † Massive on-chip memory bandwidth

Efficiency compromised by generality
• Embedded MAC units and memory blocks improve efficiency but reduce generality
† Potentially good cost and energy efficiency
   But absolute prices and power consumption are much higher than DSPs’

---

FPGAs

**Strengths and Weaknesses**

Development is long and complicated
Higher complexity inherent due to flexibility
Design flow is unfamiliar to most DSP engineers
† But cost and complexity is much lower than ASICs’
Development infrastructure badly lags DSPs’
DSP-oriented tools are immature
† Field reconfigurability (for some products)
† Reconfigure hardware for diverse tasks
• Xilinx has mature products, but others are playing catch-up
Evaluating FPGAs For Communication Infrastructure Applications

Why Use a DSP?

- Some applications are not amenable to FPGA implementations
  - Parallelism is sometimes inherently limited
  - Ultimate speed is not always the first priority
- FPGAs are still too expensive for terminal applications
- FPGA energy efficiency is still an unknown
- Implementing a complex algorithm is much more difficult on an FPGA than on a DSP

Grading the Alternatives

<table>
<thead>
<tr>
<th></th>
<th>DSPs</th>
<th>GPPs</th>
<th>FPGAs</th>
<th>Custom Cores</th>
<th>ASICs</th>
<th>ASSPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design Effort</td>
<td>B</td>
<td>A</td>
<td>D</td>
<td>C</td>
<td>E</td>
<td>A+</td>
</tr>
<tr>
<td>Design Flexibility</td>
<td>E</td>
<td>E</td>
<td>B</td>
<td>C</td>
<td>A</td>
<td>E</td>
</tr>
<tr>
<td>Run-time Flexibility</td>
<td>C</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>Top Speed</td>
<td>D</td>
<td>E</td>
<td>B</td>
<td>C</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>Energy Efficiency</td>
<td>C</td>
<td>D</td>
<td>C</td>
<td>B</td>
<td>A</td>
<td>A</td>
</tr>
</tbody>
</table>

A = Best, E = Worst

© 2003 Berkeley Design Technology, Inc.
Evaluating FPGAs For Communication Infrastructure Applications

Future Communications Applications
Dealing with Non-ideal Channels

Multi-antenna approach exploits multi-path fading by sending data along good channels
Results in large theoretical improvements in bandwidth efficiency for fading channels
But … computationally hungry

Conclusions

High-end FPGAs can outstrip DSPs on certain DSP tasks
- Computation-intensive, highly parallelizable tasks
High-end FPGAs are expensive, but they can beat DSPs in terms of performance per dollar
DSP have the advantage in development infrastructure, time-to-market, developer familiarity.
In many applications, a heterogeneous combination of computing engines is desirable
- Expect to see more heterogeneous processor chips
The “best” architecture depends on the details of the application