DSP Benchmark Results for the Latest Processors

Outline

• Why are benchmarks important?
• Comparing benchmarking approaches
• Benchmark results for the latest processors
  • ‘C64xx, SC140, ‘C55xx, ‘BF53x, OMAP, PXA2xx
• The effect of architecture on benchmark results
  • Architectural trends and trade-offs
• Emerging challenges in benchmarking
• Conclusions
Why Benchmark?

- Assess and compare key processor metrics accurately
  - DSP speed
  - Memory efficiency
  - Energy efficiency
  - Cost-performance

- Compare performance across a wide range of architectures (conventional, VLIW, SIMD, DSP-enhanced GPP, etc.)—difficult without benchmarks
  - Simple metrics (MIPS, MACS) don't cut it

How to Benchmark?

A few candidate approaches:

- Simplified metrics
  - E.g., MIPS (Millions of Instructions Per Second), MOPS, MMACS

- Full DSP applications
  - E.g., v.90 modem

- DSP algorithm “kernel” benchmarks
  - E.g., FIR filter, FFT
What’s Wrong with MIPS?

MIPS and MFLOPS (Millions of Floating-Point Operations per Second) are frequently used as shorthand for processor speed. But are they really meaningful?

Two instructions from different processors:

DSP16410
\[ A0 = A0 + P0 + P1 \]
\[ P0 = Xh * Yh \]
\[ P1 = Xl * Yl \]
\[ Y = R0 ++ \]
\[ X = PTO ++ \]

TMS320C6414
ADD A0, A3, A0

Full-Application Benchmarks

This approach has pros and cons

- Applications tend to be ill-defined
- Hand-optimization needed
  - Costly, time-consuming to implement
  - Measures programmer as much as processor
- Measures system, not just processor
  - Sometimes this is an advantage
- Results useful only for specific app (or similar apps)
  - But if results are available for your app, this not a disadvantage
- For processors, similar results via simpler approach
  - But this is not true for all DSP implementation technologies
Algorithm Kernel Benchmarks

The BDTI Benchmarks are based on DSP algorithm kernels
- The most computationally intensive portions of DSP applications
- Examples include FFTs, IIR filters, and Viterbi decoders

Benchmark results are used with application profiling to predict overall performance

Advantages:
- Relevant; chosen by analysis of real DSP apps
- Kernels are short, allowing
  - Functionality to be precisely specified
  - Benchmarks to be implemented, optimized in a reasonable amount of time

Disadvantages:
- Not practical to implement all possible algorithms
- Don’t reflect application-level optimizations and trade-offs
  - For some implementation technologies, this is a problem
- Ignores system-level considerations
  - This, too, is a problem for some implementation technologies
Vendor Benchmarks

Many processor vendors provide benchmark results, but
- Benchmarks not standardized across vendors
- Results not independently verified
- Clock speeds often projected

Results are often misused, for example,
- Comparing results for functionally different benchmarks
- Comparing fastest chip to slowest from another vendor
- Comparing vaporware to real silicon
- Presenting cycle counts as a proxy for performance
- Cherry-picking benchmark results

Benchmark Results for the Latest Processors

High-performance processors
- Texas Instruments TMS320C64xx
- StarCore SC140

Low-power processors
- Texas Instruments TMS320C55xx
- Analog Devices Blackfin (ADSP-BF53x)

General-purpose/DSP processors
- Intel PXA2xx
- Texas Instruments OMAP5910
DSP Benchmark Results for the Latest Processors

**Overall DSP Speed: BDTI mark2000™**

Higher is faster

- **TI 'C5502 (300 MHz)**: 1460
- **ADI 'BF53x (600 MHz)**: 3360
- **TI 'C6414 (720 MHz)**: 6480
- **StarCore SC140 (300 MHz)**: 3430
- **Intel PXA2xx (400 MHz)**: 930

(See www.BDTI.com for more scores)

**Relative Results on Benchmarks**

- The BDTI mark2000 shows overall speed results across 12 DSP kernel benchmarks
- Relative results on specific benchmarks can vary widely

**BDTImark2000 Higher is Faster**

- 'C6414 (720 MHz): 6480
- SC140 (300 MHz): 3430

**LMS Adaptive FIR Filter (execution time, in µs) Lower is Faster**

- 'C6414 (720 MHz): 0.05
- SC140 (300 MHz): 0.06
What Factors Affect DSP Speed?

Processors’ DSP speeds are affected by:
- Parallelism
  - How many parallel operations can be executed per cycle
- Instruction set
  - Suitability for the task at hand
- Clock speed
- Data types
- Data bandwidth
- Pipeline depth
  - Instruction latencies
- Support for DSP-oriented features, e.g.,
  - DSP addressing modes
  - Zero-overhead looping
  - Saturation, scaling, rounding

Memory Use:

BDTI Control Benchmark

<table>
<thead>
<tr>
<th>Memory Use (Bytes)</th>
<th>TI 'C55xx (8/16/32/48)</th>
<th>ADI 'BF53x (16/32/64)</th>
<th>TI 'C64xx (32)</th>
<th>StarCore SC140 (16/32)</th>
<th>Intel PXA2xx (16/32)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lower is better</td>
<td>146</td>
<td>140</td>
<td>256</td>
<td>144</td>
<td>140</td>
</tr>
</tbody>
</table>
What Factors Affect Memory Use?

Processors' memory usage affected by:

- **Instruction set**
  - Wider instructions take more memory
  - Mixed-width instruction sets becoming popular
    - Use short, simple instructions for simple tasks
    - Use longer instructions for more complex tasks
  - Suitability of instruction set for task at hand

- **Architecture**
  - VLIW, SIMD, and deep pipelines all may encourage (or require) optimizations that increase memory use to obtain speed-optimized code

- **Compiler quality (for compiled code)**

Energy Efficiency:

<table>
<thead>
<tr>
<th>Processor</th>
<th>Frequency</th>
<th>Voltage</th>
<th>BDTI mark2000/mW</th>
</tr>
</thead>
<tbody>
<tr>
<td>TI 'C55xx</td>
<td>300 MHz,</td>
<td>1.26V</td>
<td>11.8</td>
</tr>
<tr>
<td>ADI 'BF53x</td>
<td>600 MHz,</td>
<td>1.2V</td>
<td>16.9</td>
</tr>
<tr>
<td>TI 'C64xx</td>
<td>300 MHz,</td>
<td>1.0V</td>
<td>16.1</td>
</tr>
<tr>
<td>Motorola MSC8101 (SC140)</td>
<td>300 MHz,</td>
<td>1.5V</td>
<td>13.7</td>
</tr>
<tr>
<td>Intel PXA2xx</td>
<td>200 MHz,</td>
<td>1.0V</td>
<td>2.6 (estimated)</td>
</tr>
</tbody>
</table>

Higher is better
What Factors Affect Energy Efficiency?

Processors’ energy consumption affected by:

- Hardware implementation
  - Fabrication process, voltage, circuit design, logic design
- Memory usage
- Match between instruction set and task at hand
- Compiler quality (for compiled code)

Cost-Perf: BDTI mark2000/ $

<table>
<thead>
<tr>
<th>Processor</th>
<th>Clock Speed</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>TI 'C55xx</td>
<td>300 MHz</td>
<td>$9.95</td>
</tr>
<tr>
<td>TI 'C64xx</td>
<td>500 MHz</td>
<td>$44.95</td>
</tr>
<tr>
<td>ADI 'BF53x</td>
<td>400 MHz</td>
<td>$5.95</td>
</tr>
<tr>
<td>Motorola MSC8101 (SC140)</td>
<td>300 MHz, $118.10</td>
<td></td>
</tr>
<tr>
<td>Intel PXA2xx</td>
<td>300 MHz</td>
<td>$27.20</td>
</tr>
</tbody>
</table>
What Factors Affect Cost-Perf?

Speed
Chip cost, which is affected by:
- Die size
  - Fabrication process
  - Size of on-chip memory
    - Influenced by processor’s memory usage
  - On-chip peripherals

Good cost-performance results don’t necessarily mean chip is suitable for apps with severe cost constraints
- OEMs don’t want to pay for more performance than is needed

Architectural Trends

- VLIW (multi-issue) to increase performance
- SIMD to increase performance
- Simplified instruction sets, architectures to increase clock speeds, compilability
- Mixed-width instruction sets to reduce memory usage
- Deeper pipelines to enable higher clock speeds
- DSP-enhanced general-purpose processors (GPPs)
Architectural Trends: The Down Side

- VLIW (multi-issue), SIMD, and deep pipelines can increase
  - Memory use
  - Energy consumption
  - Code-generation complexity

- Simple instruction sets often increase memory usage
  - More instructions are needed to accomplish a given task

- Sometimes a processor’s legacy constraints are overriding

Each processor makes different tradeoffs, depending on its target application—top speed is often not the goal!

Texas Instruments TMS320C64xx

Targets high-performance DSP applications.

Goals:
- Fast
- Compilable
- Compatible with earlier ‘C62xx

Sacrifices:
- High memory consumption
- High chip price for fastest ($199, qty 10K)
- Difficult to program in assembly language
DSP Benchmark Results for the Latest Processors

‘C64xx is Fast Because...

Highly parallel architecture
- Based on ‘C62xx: VLIW, up to 8 instructions/cycle
- Adds SIMD to ‘C62xx
  - Faster, but still compatible
  - Four 16-bit MACs/cycle
  - More powerful instructions

Very high clock speed (720 MHz)
- Deep pipe (11 stages)
- Most instructions are simple
  - 32-bit, mostly RISC-like
  - Also increases compilability
- Caching

But ‘C64xx Makes Sacrifices

- High memory use
  - Wide, uniform-width (32-bit) instructions
  - Mostly simple, RISC-like instructions
  - VLIW, SIMD, deep pipeline

- Memory use increases chip cost

- Deep pipe also complicates code generation
  - Multi-cycle latencies
DSP Benchmark Results for the Latest Processors

StarCore SC140

Targets high-performance DSP applications.

Goals:
- Fast
- Low memory usage
- Easy to program in assembly
- Compilable
- Low energy consumption

Sacrifices:
- Not as fast as ‘C64xx
- High chip price (300 MHz MSC8101 is $118, qty 10K)
- No compatibility with previous architectures

SC140 Fast, But Not Fastest

Like ‘C64xx, highly parallel architecture
- VLIW, up to 6 instructions/cycle
  - Like ‘C64xx, four 16-bit MACs/cycle
  - Limited SIMD

Mid-range clock speed (300 MHz)
- About half as high as ‘C64xx
- Shallow pipe (5 stages)
  - Not deep enough for ultrahigh clock
  - But uniform single-cycle latencies ease code generation, decrease memory use
- Simple instruction set
SC140 Has Surprisingly Low Memory Use

- Mixed-width instruction set
  - 16-bit instructions with optional 16-bit prefixes
- Surprising, since it’s VLIW and uses RISC-like instructions

Memory Use on Control Benchmark
Lower is Better

<table>
<thead>
<tr>
<th>Processor</th>
<th>Memory Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>C55xx</td>
<td>146</td>
</tr>
<tr>
<td>BF53x</td>
<td>140</td>
</tr>
<tr>
<td>C64xx</td>
<td>256</td>
</tr>
<tr>
<td>SC140</td>
<td>144</td>
</tr>
<tr>
<td>PXA2xx</td>
<td>140</td>
</tr>
</tbody>
</table>

Texas Instruments TMS320C55xx

Targets low-power, cost-sensitive DSP applications.

Goals:
- Low energy consumption
- Low memory use
- Low chip cost
- Midrange speed
- Partly compatible with earlier ‘C54xx architecture

Sacrifices:
- Not nearly as fast as high-end DSPs
- Not very compilable
- Difficult to program in assembly
DSP Benchmark Results for the Latest Processors

‘C55xx Focuses on Power, Cost, Compatibility... Not Speed

- Moderate parallelism
  - Adds limited (2-issue) VLIW capabilities to boost speed while maintaining partial compatibility with ‘C54xx
    - Two MACs/cycle
    - Convoluted architecture (like ‘C54xx)
- Medium clock speed (300 MHz)
  - 7-stage pipeline
  - Single-cycle latencies
- Moderately priced ($8-20 qty 10K)

Analog Devices ADSP-BF53x

Targets low-power, cost-sensitive DSP applications.

Goals:
- Low energy consumption
- Low memory use
- Low chip cost
- Midrange speed
- Compilable

Sacrifices:
- Not nearly as fast as high-end DSPs
- No compatibility with previous architectures
ADSP-BF53x Balances Power, Cost, Speed

- Moderate parallelism
  - 3-issue VLIW
    - Two MACs per cycle
  - Somewhat more parallelism than ‘C55xx; not nearly as much as SC140 or ‘C64xx
    - Not constrained by legacy architecture
- High clock speed (600 MHz)
  - 10-stage pipeline
  - Single-cycle latencies
- Good energy efficiency
- Moderately priced
  ($6-35 qty 10K)

© 2003 Berkeley Design Technology, Inc.

‘BF53x and ‘C55xx Both Have Low Memory Usage

- Both use mixed-width instruction sets
  - ‘BF53x uses 16/32/64-bit instructions
  - ‘C55xx uses instructions ranging from 8-48 bits
- ‘BF53x instructions (and architecture) are fairly simple
  - ‘BF53x is easy to program in assembly, good compiler target
- ‘C55xx inherits ‘C54xx instruction set
  - Not as easy to program in assembly as ‘BF53x, but familiar to ‘C54xx programmers
  - Not a good compiler target
  - Results for compiled code would likely favor ‘BF53x

© 2003 Berkeley Design Technology, Inc.
DSP Benchmark Results for the Latest Processors

‘BF53x vs. ‘C55xx Energy Efficiency

- ‘BF53x supports multiple voltages
  - Good energy efficiency at top speed of 600 MHz/1.2V
  - Energy results at other speed/voltage combos will vary
- ‘C55xx supports one voltage

<table>
<thead>
<tr>
<th></th>
<th>BDTrimark2000/mW</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘BF53x</td>
<td>11.8</td>
</tr>
<tr>
<td>‘C55xx</td>
<td>16.9</td>
</tr>
</tbody>
</table>

© 2003 Berkeley Design Technology, Inc.

Intel PXA2xx (XScale)

Targets low-power, cost-sensitive DSP applications where general-purpose processing features or software are needed.

Goals:
- Low memory use
- Low chip cost
- Midrange speed
- Compatible with earlier ARM architectures
- Support for operating systems, compilers

Sacrifices:
- Not very efficient for DSP
- Poor energy and cost efficiency

© 2003 Berkeley Design Technology, Inc.
### PXA2xx Speed Comes from Clock

- Relatively low level of parallelism
  - Single-issue, mostly general-purpose 32-bit architecture
    - Good compiler target
  - ARM architecture augmented with limited SIMD
    - Two MACs/cycle
    - Few additional DSP-specific features
- Moderately high clock speed (400 MHz)
  - 7-stage pipeline
  - Multi-cycle latencies
- Moderately priced
  ($27-42 \text{ qty } 10K$)

### PXA2xx is Efficient in Memory

- Not surprising; general-purpose architecture is a good match for control code
- Mixed-width (16/32) instruction set
- Probably would look even better if benchmark showed compiled code results
**PXA2xx Can’t Compete With DSPs on Energy, Cost**

- Not very efficient on DSP algorithms
- High clock rate helps boost speed, but doesn’t help with energy or chip cost
- This is often a drawback of GPPs for DSP

```
   BDTImark2000/mW
   Higher is Better

<table>
<thead>
<tr>
<th>Processor</th>
<th>BDTImark2000/mW</th>
</tr>
</thead>
<tbody>
<tr>
<td>'C55xx</td>
<td>11.8</td>
</tr>
<tr>
<td>'BF53x</td>
<td>16.9</td>
</tr>
<tr>
<td>'C64xx</td>
<td>16.1</td>
</tr>
<tr>
<td>SC140</td>
<td>13.7</td>
</tr>
<tr>
<td>PXA2xx</td>
<td>2.6 (est.)</td>
</tr>
</tbody>
</table>
```

```
   BDTImark2000/$
   Higher is Better

<table>
<thead>
<tr>
<th>Processor</th>
<th>BDTImark2000/$</th>
</tr>
</thead>
<tbody>
<tr>
<td>'C55xx</td>
<td>146</td>
</tr>
<tr>
<td>'BF53x</td>
<td>376</td>
</tr>
<tr>
<td>'C64xx</td>
<td>98</td>
</tr>
<tr>
<td>SC140</td>
<td>29</td>
</tr>
<tr>
<td>PXA2xx</td>
<td>26</td>
</tr>
</tbody>
</table>
```

**TI OMAP (‘C55xx plus ARM9)**

Targets low-power, cost-sensitive DSP applications where general-purpose processing features or software are needed.

Goals:

- Low energy consumption
- Low memory use
- Low chip cost
- Midrange speed
- Support for operating systems, compilers
- Compatibility with ‘C54xx, ARM

Sacrifices:

- ‘C55xx not very compilable
- Dual-core approach complicates system development
DSP Benchmark Results for the Latest Processors

OMAP BDTI mark2000 Score??

OMAP5910 BDTI mark2000 score could range from 186 (ARM only) to 916 (ARM plus C55xx) ... depending on app partitioning.

Emerging Benchmarking Challenges

New technologies create benchmarking challenges
- Multi-core devices
- DSP-enhanced FPGAs
- Application-specific processors
- Customizable processors
- Reconfigurable processors

Evolving applications and tools also lead to new challenges
- Increasing reliance on C compilers
Application Benchmarking

- For technologies not well served by kernel benchmarks
  - DSP-enhanced FPGAs
  - Application-specific processors
- Limited applicability
- Practicality concerns can be partly addressed by
  - Using off-the-shelf implementations where available, or
  - Using simplified applications
    - E.g., BDTI's OFDM Benchmark—simplified telecom receiver

Compiler Benchmarking

- Better compilers, more compilable architectures encourage migration to C for DSP
- Processor selection may hinge on compiler quality
- But it is difficult to assess how efficient a compiler is for DSP...
  - Or to compare two compilers for the same processor
- Compiler benchmarking sounds simple, but raises complex questions...
  - Benchmark the compiler or the compiler+processor?
  - Allow intrinsics?
  - Allow C source code optimizations?
  - Allow limited assembly tweaking?
  - Where to draw the line?
Conclusions

• Today’s DSP-oriented processors cannot be meaningfully compared using simplified metrics
  • Relevant, meaningful benchmark results are essential to processor evaluation
• There is no ideal processor
  • Fastest doesn’t mean best
    • The “best” processor depends on the details of the application
  • Different architectural approaches make different performance trade-offs
    • Understanding these is key to selecting a processor

Conclusions

• Consider all the options
  • Increasing performance overlap between dissimilar architectures
  • Alternatives increasingly viable
• Application requirements and processor performance are both moving targets
• Emerging architectures and technologies require benchmarking evolution
• Factors other than performance are often important
  • Compatibility, tools, off-the-shelf software, ...
For More Information…

www.BDTI.com

Free Information
- BDTI mark2000™ scores
- DSP Insider newsletter
- Pocket Guide to Processors for DSP

White papers on processor architectures and benchmarking

Article reprints on DSP-oriented processors and applications
- EE Times
- IEEE Spectrum
- IEEE Computer and others

comp.dsp FAQ

© 2003 Berkeley Design Technology, Inc.