Insight, Analysis, and Advice on Signal Processing Technology

Evaluating the DSP Processor Options

(DSP-522)

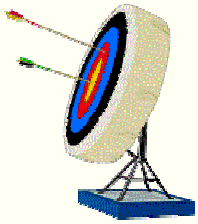
Jeff Bier
Berkeley Design Technology, Inc.
Berkeley, California USA
+1 (510) 665-1600

info@BDTI.com
http://www.BDTI.com

Presentation Goals

By the end of this workshop, you should know:

- What to consider when choosing a DSP processor
- Key characteristics of modern DSP processors
- Strengths and weaknesses of the latest processors
- Why DSPs are not always the best solution



Outline

Signal processing application needs

Architectural approaches

Benchmarking methodology

Strengths and weaknesses of new processors

Alternatives

Conclusions

Key Application Needs

Performance

Energy efficiency

System cost

Integration

Miniaturization

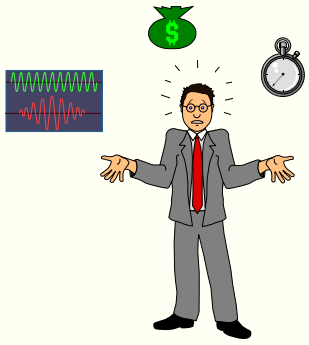
Software complexity and optimization

In varying combinations, with diverse algorithms

Time to market

Reduce risk

Installed base



DSP Algorithms Shape DSPs

How Signal Processing is Different From Other Tasks

- Very computationally demanding
- Requires attention to numeric fidelity
- High memory bandwidth requirements
- Streaming data—and lots of it
- Predictable data access patterns
- Execution-time locality
- Math-centric
- Real-time constraints
- Standards: algorithms, interfaces

 5

Outline

Signal processing application needs

Architectural approaches

Benchmarking methodology

Strengths and weaknesses of new processors

Alternatives

Conclusions

 6

Compound Instructions

- Powerful instructions
  - Each instruction performs multiple operations
- "Tuned" instruction sets
  - Specialized instructions common
  - Instructions have complex, non-uniform structure
  - Instruction widths may vary
- Non-orthogonal instruction sets
  - Very non-uniform treatment of operations, data types, and addressing modes
  - Irregularity in combinations of operations supported
- Multiple small register sets dedicated to specific functions
- Example: TMS320C5xxx

 7

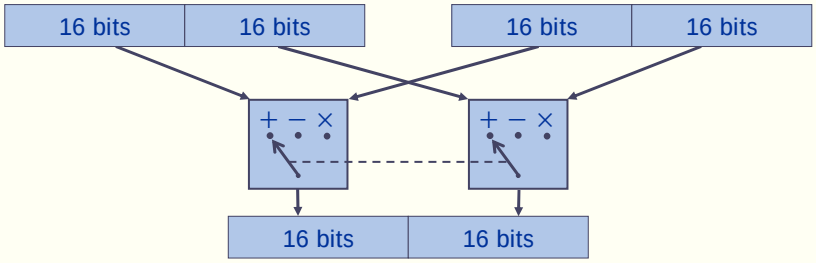
Reduced Instruction Set Computing

- Simple instructions
  - Each instruction typically describes one operation
- Regular, orthogonal instruction sets
  - Instructions have simple, uniform structure & width
    - Newer processors offer instruction subsets with narrower widths
  - Uniform treatment of operations, data types, and addressing modes
  - Few specialized instructions
- Typically feature large, uniform register sets
- Example: ARM7

 8

SIMD

Single Instruction, Multiple Data



Performs the same operation simultaneously on multiple sets of operands

- Under the control of a single instruction

Some SIMD processors support multiple data widths (for example, 32-bit, 16-bit, and 8-bit)

Multi-Issue Techniques

VLIW: Multi-issue architectures with compile-time instruction scheduling

- Typically four or more instructions per cycle with flexible instruction grouping
  - Examples: TI 'C6xxx, StarCore SC1000
- Sometimes 2-3 instructions per cycle with restricted instruction grouping
  - Examples: ADI Blackfin, TI 'C55x

Superscalar: Multi-issue architectures with run-time instruction scheduling

- Examples: Intel Pentium, LSI Logic ZSP

Traditional vs. Modern DSPs

	Traditional DSP	Modern DSP
Instructions	Highly compound	RISC or combination of RISC and compound
Multi-issue	N/A	Typically VLIW; a few superscalar
Issue width	1	2-8
SIMD	Limited, e.g., only a dual-16-bit add	Extensive, e.g., 1x32, 2x16, or 4x8 for most arithmetic operations
Coprocessors / accelerators	Rare	Communications and multimedia hardware common

 11

Multi-Core DSPs

Another level of parallelism

A growing trend?

- Examples: ADI Blackfin, Freescale MSC81xx

Good for applications with multiple channels

Good for applications with easily separable components, e.g., video decoding

Potentially difficult programming model

CAUTION: comparing single-core and multi-core processors is tricky!

 12

General-Purpose Features

Several DSPs now include features traditionally associated with general-purpose processors
- User modes
- Memory management units
- Compiler-oriented features such as specialized addressing modes

Enable more sophisticated OSs

Ease implementation of non-signal-processing tasks

Often make processors better compiler targets

 13

Compatibility

Compatibility is often a weak point for DSPs
- Some new DSP architectures sacrifice compatibility for leaps in performance, compilability, efficiency
  - Examples: ADI Blackfin, CEVA CEVA-X
- Assembly-level and even binary compatibility increasingly common
  - CAUTION: “compatible” may mean “partially compatible”

Increasing attention to compatibility complicates and constrains new architectures
- E.g., ‘C5xxx very complex instruction set

Compatibility does not ensure code reusability!
- Legacy code may need to be re-optimized—or largely rewritten—to achieve good performance

 14

Proprietary vs. Shared

Most DSP architectures are not shared, not licensable
- One silicon vendor per architecture

In contrast, many GPPs are shared and licensable
- ARM, MIPS, PowerPC, SH-x, x86

There are some shared or licensable DSP architectures
- CEVA, StarCore, ZSP

Shared architectures can encourage stability, price competition, and wide third-party support

But proprietary architectures
- May achieve faster technological advances
- May have market momentum, integration advantages

 15

Development Infrastructure

Application development infrastructure is a critical consideration

System developers increasingly depend on chip vendors for software, reference platforms, etc.

DSPs are increasingly specialized for communications or media processing
- When does a DSP become an application-specific standard product (ASSP)?

 16

Outline

Signal processing application needs
Architectural approaches
Benchmarking methodology
Strengths and weaknesses of new processors
Alternatives
Conclusions

What's Wrong With MMACS?

MMACS approximate performance on some signal processing algorithms like FIR filters, but:

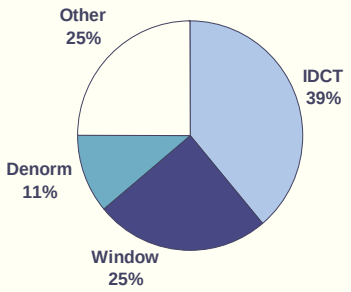
- It ignores other operations required to sustain repeated MACs
- It ignores memory bandwidth bottlenecks
- Many important signal processing algorithms don't use MACs!

Example: TI 'C55x vs. Intel PXA26x

- 200 MHz 'C55x: 400 MMACS and 1,200 million bytes/sec
- 400 MHz PXA26x: 800 MMACS and 1,600 million bytes/sec
- These two processors have comparable signal processing speed!

The BDTI Benchmarks™

- Algorithm kernels are the most computationally intensive portions of signal processing applications
- Example algorithm kernels include FFTs, IIR filters, and Viterbi decoders

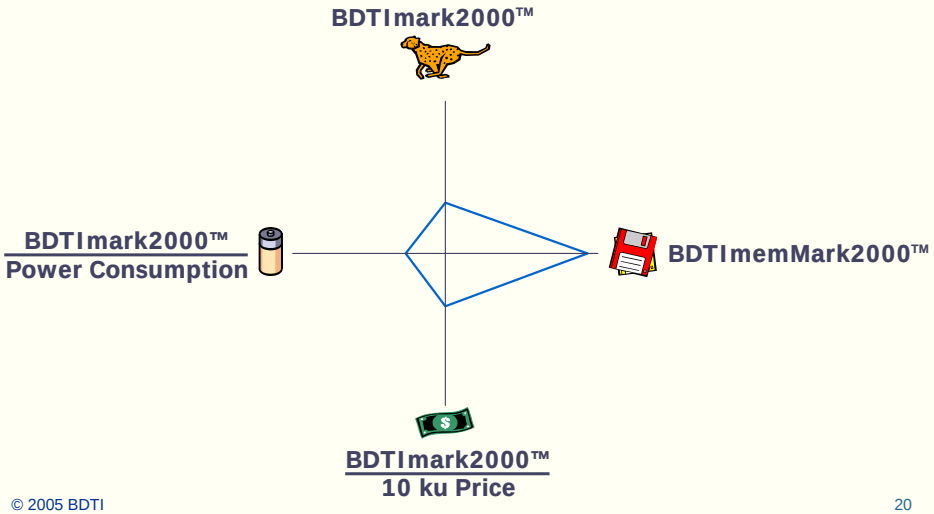


- Application-relevant algorithm kernels are strong predictors of overall performance
- About 70 architectures already benchmarked

 19

Benchmark Results

Median Result



 20

Outline

Signal processing application needs

Architectural approaches

Benchmarking methodology

Strengths and weaknesses of new processors

Alternatives

Conclusions

Texas Instruments TMS320C64x

The 'C62x Gets Serious Enhancements

8-issue 16-bit fixed-point architecture
- Up to four 16-bit MACs per cycle
- Special instructions and co-processors for communications and multimedia
- Compatible with 'C62x, 'C67x

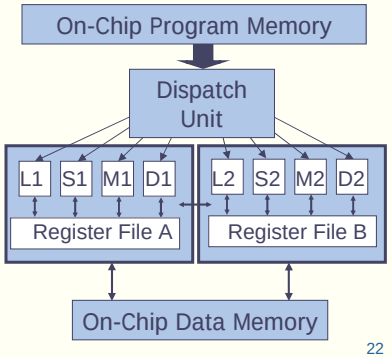
11-stage pipeline with multi-cycle latencies

Two-level cache memory system

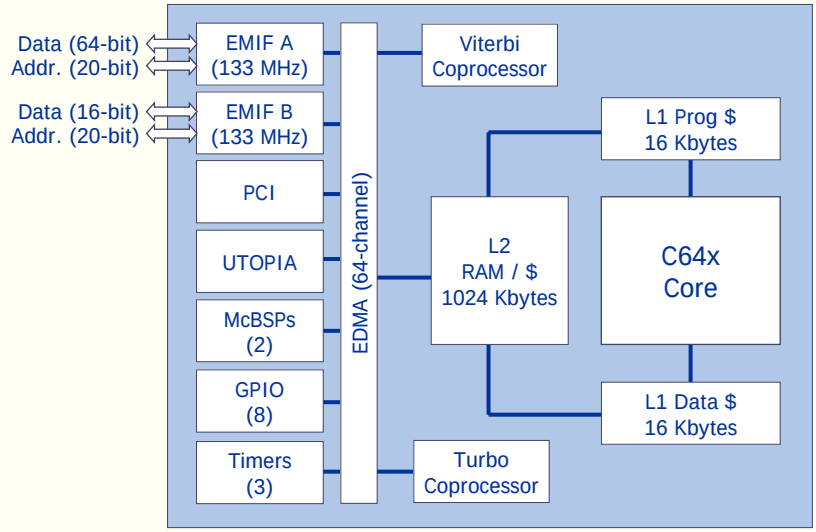
32-bit instruction set

Shipping at 1 GHz, $189 (10 ku)

Available in ASSPs for communications and multimedia



TMS320C6416 Integration

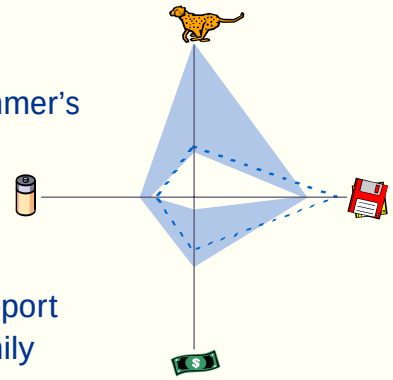


 23

Texas Instruments TMS320C64x

Strengths and Weaknesses

- ↑ Large family spans wide range of cost, performance and on-chip integration
- ↑ Most parts are very fast
- ↓ Complex programming model
  - ↓ 'C6xxx is assembly programmer's worst nightmare
  - ↓ Caches reduce execution-time predictability
- ↑ Good integration
- ↑ Good tools and third-party support
  - ↑ Compatible with 'C6xxx family



 24

Analog Devices ADSP-BF53x

3-issue VLIW, 16-bit fixed-point architecture
- Up to two 16-bit MACs per cycle
- MMU and mode-dependent instructions

10-stage pipeline with single-cycle latencies

16/32-bit instruction set

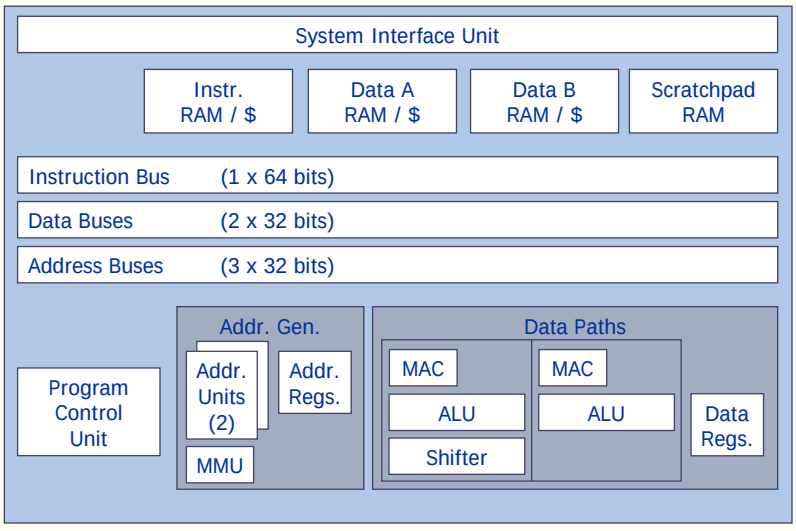
Memory system configurable as cache

Speed/voltage scaling: 100 MHz/0.7 V – 750 MHz/1.4 V

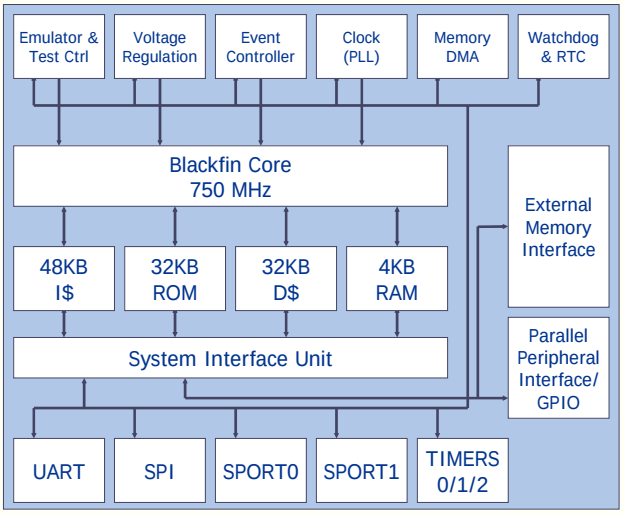
Shipping at 750 MHz, $32 (10 ku)
- Dual-core 'BF561 shipping at 750 MHz, $40 (10 ku)

Used in ASSPs for communications and multimedia

Blackfin Architecture



ADSP-BF533 On-Chip Integration

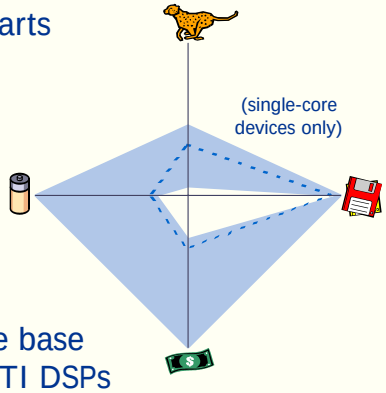


 27

Analog Devices ADSP-BF53x

Strengths and Weaknesses

- ↑ Excellent memory-, cost-, and energy-efficiency
  - ↑ Unusual speed/energy flexibility
- • Modest speed for single-core parts
- ↑ Easy to program; good compiler target
- ↑ OS-friendly hardware features
- • Sophisticated but complex memory system
- ↑ Good integration
- ↑ Good tools
- ↓ Not compatible; no legacy code base
  - ↓ Less 3rd-party support than TI DSPs



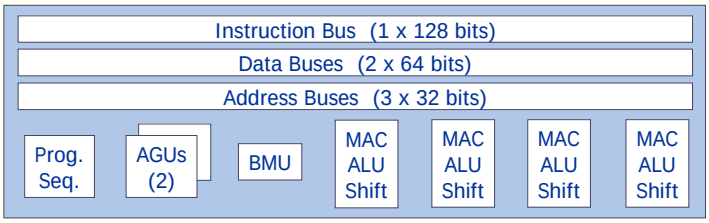
 28

Freescale MSC711x and MSC81xx

StarCore-based DSPs

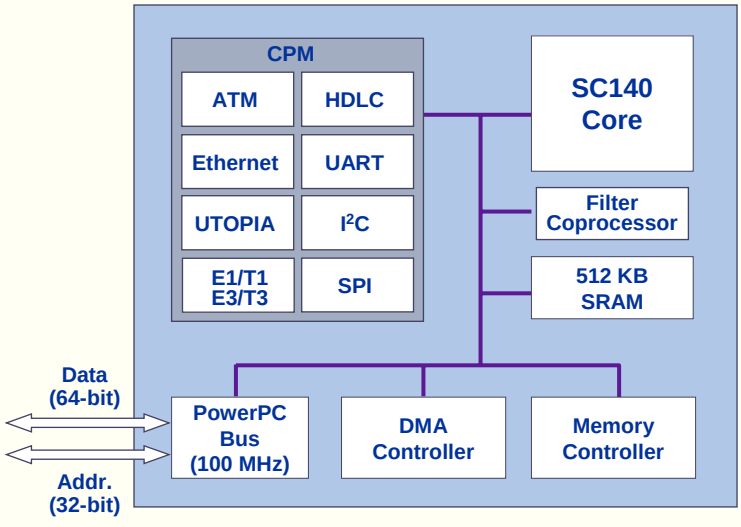
6-issue 16-bit fixed-point VLIW architecture
- Up to four 16-bit MACs per cycle

5-stage pipeline with single-cycle latencies
Mixed-width 16- and 32-bit instruction set
Mainly targeting telecom applications
Single-core parts shipping at 300 MHz, $80 (10 ku)
Quad-core parts shipping at 500 MHz, $191 (10 ku)



 29

MSC8101 On-Chip Integration

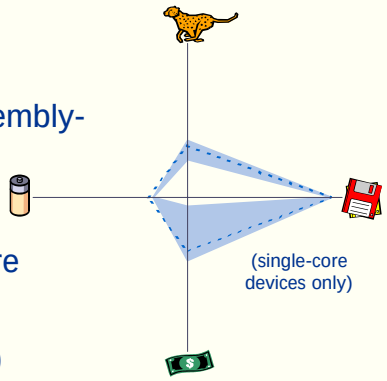


 30

MSC711x and MSC81xx

Strengths and Weaknesses

- ↓ Modest performance on most metrics
  - ↑ Good memory-efficiency
  - ↑ Quad-core parts offer stronger performance
- ↑ Good target for compilers, assembly-language programmers
- ↑ Strong, relevant integration
- ↑ Architecture available as StarCore SC1400 licensable core
  - ↑ Compatible with lower-performance SC1200, higher-performance SC2400
- ↓ Modest 3rd-party support



Other Major Fixed-point DSPs

Vendor	Family	Data Width	Core Clock Speed	BDTImark2000 *BDTIsimMark2000*	On-Chip Memory, Bytes	Unit Price	Notes
Analog Devices	ADSP-219x	16 bits	160 MHz	410	20 K–160 K	$10–24	Enhanced version of the ADSP-218x
	ADSP-TS20x (TigerSHARC)	8/16/32/40 bits	600 MHz	6400	512 K–3 M	$47–197	4-way VLIW with SIMD capabilities; uses eDRAM
Freescale	DSP563xx	24 bits	275 MHz	820	24 K–649 K	$4–47	Many audio-oriented parts; binary-compatible with '560xx
	DSP5685x/ 56F8xxx	16 bits	120 MHz	*340*	22 K–300 K	$4–17	Contains many microcontroller-like features
LSI Logic	LSI40x (ZSP400)	16/32 bits	200 MHz	940	96 K–252 K	$4–13	4-way superscalar DSP; available as licensable core
NEC	µPD77050 (SPXK5)	16 bits	250 MHz	*1770*	400 K	$15	Dual-MAC DSP with variable speed and voltage
Renesas	SH772x (SH3-DSP)	16 bits	200 MHz	490	32 K	$17–24	Hybrid DSP/microprocessor based on SH3-DSP
	SH775x (SH-4)	16/32 bits	240 MHz	750	32 K	$21–31	Superscalar microprocessor with 3D geometry instructions
TI	TMS320C28x/ TMS320F28x	32 bits	150 MHz	n/a	40 K–294 K	$6–17	Hybrid microcontroller/DSP; assembly-compatible w/ 'C24x
	TMS320C54x	16 bits	160 MHz	500	24 K–1280 K	$4–104	Many specialized instructions
	TMS320C55x	16 bits	300 MHz	1460	80 K–376 K	$5–19	Dual-issue, dual-MAC DSP; assembly-compatible w/ 'C54x

All data as of the fourth quarter of 2004

Outline

Signal processing application needs
Architectural approaches
Benchmarking methodology
Alternatives
Conclusions

Alternatives to DSP Processors

ASICs
- Licensable cores
- Customizable cores

ASSPs

General-purpose processors (GPPs)
- PC CPUs
- Embedded GPPs

Reconfigurable architectures
- FPGAs
- Reconfigurable processors

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

 35

BDTI Communications Benchmark™

	DSP A	DSP B	Altera Stratix 1S20-6	Altera Stratix 1S80-6
Channels	<0.2	~0.7	~20	~60
Cost (1 ku)	~$15	~$210	~$210	~$3,200
Cost per channel	~$90	~$300	~$10	~$50

From BDTI's report *FPGAs for DSP* and unpublished benchmarks.

 36

Grading the Alternatives

	DSPs	GPPs	FPGAs	Custom Cores	ASICs	ASSPs
Design Effort	B	A	D	C	E	A+
Design Flexibility	E	E	B	C	A	E
Run-time Flexibility	C	B	A	C	E	E
Top Speed	D	E	B	C	A	A
Energy Efficiency	C	D	C	B	A	A

A = Best, E = Worst

Outline

Signal processing application needs
Architectural approaches
Benchmarking methodology
Strengths and weaknesses of new processors
Alternatives
Conclusions

Conclusions

Consider all relevant performance metrics
- Speed, power, price, ...
- Today's complex DSPs require a thoughtful benchmarking approach

Factors other than performance are always important

DSPs are increasingly specialized for specific applications
- Many target communications and multimedia

DSPs are not always the best solution
- Consider all of the options

 39

For More Information... www.BDTI.com

Inside [DSP] newsletter and quarterly reports

Benchmark scores for dozens of processors

Pocket Guide to Processors for DSP
- Basic stats on over 40 processors

Articles, white papers, and presentation slides
- Processor architectures and performance
- Signal processing applications
- Signal processing software optimization

comp.dsp FAQ

2004 Edition

 40