

Evaluation of Static and Dynamic Scheduling for Media Processors

Jason Fritts

Assistant Professor

Department of Computer Science

Co-Author: Wayne Wolf



Washington
University in St. Louis

Overview

- **Media Processing – Present and Future**
- **Evaluation Environment**
- **Dynamic vs. Static Architectures**
- **Effects of High Frequency**
- **Conclusions**
- **Future Research**

2

Multimedia Applications

- **Wide range of applications**

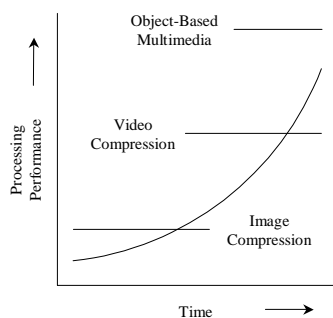
- *Communication*
 - video conferencing
 - World Wide Web
 - digital/video libraries
 - videophones
- *Entertainment*
 - video/computer games
 - movies
 - animation
- *Computer Vision*
 - image understanding
 - surveillance
 - tracking
- *Education*
 - interactive learning
 - virtual classrooms
- *Art and Architecture*

Multimedia is primarily a communication media

3

Future of Multimedia

Multimedia industry evolves with processor performance.



Multimedia is moving towards advanced representations

4

Current Media Processing Solutions

- **Application-specific processors**
 - high performance at low cost
 - very limited flexibility

- **Multimedia extensions to general-purpose processors**
 - good programmability at little added cost
 - some speedup with subword parallelism
 - optimized for general-purpose processing

- **Current “programmable” media processors**
 - good performance
 - specialized hardware
 - subword parallelism
 - ILP
 - good programmability (w/ special programming libraries)
 - moderate frequency

5

Future Media Processors

- **Increasing Performance**
 - high frequency
 - improved ILP

- **Cost is Major Barrier**
 - high resource costs are primary barrier to using such mechanisms
 - smaller market for media processing prohibits high resource costs
 - media processors currently much more expensive per MIPS

- **Diminishing Costs**
 - increasing market for media processing
 - decreasing power per MIPS
 - demonstrated by recently announced TI C64x => frequencies up to 1.1 GHz

<i>Processor</i>	<i>Frequency</i>	<i>Power</i>	<i>Execution Units</i>	<i>L2 Cache</i>	<i>VLSI Technology</i>
TI C62x	up to 300 MHz	up to 2 W	8	varies, w/ up to 7 Mb mem	.15 - .18 μ m
Intel Pentium III	500 MHz - 1 GHz	13 - 16 W	5	32 KB L1, 256 KB L2	.18 - .25 μ m

6

Evaluation Environment

7

MediaBench Benchmark Suite

- **Developed at UCLA**

[CLee97] "MediaBench: A Tool for Evaluating and Synthesizing Multimedia Communication Systems," MICRO-30, 1997.

- **Excellent combination of applications**

- video: MPEG-2
- audio: ADPCM coder
- graphics: Mesa
- image: JPEG, EPIC, Ghostscript
- security: PGP, Pegwit
- speech: GSM, G.721, Rasta

- **Augmented for greater representation of future multimedia**

- MPEG-4 object-oriented video
- H.263 very-low bitrate video

8

IMPACT Environment

- **Aggressive ILP research compiler**
 - Three levels of optimizations
 - Classical - classical optimizations only
 - Superscalar - adds loop unrolling and superblock formation
 - Hyperblock - adds hyperblock optimization
- **Architecture-independent evaluation**
 - large, generic instruction set
 - retargetable back-end
- **Performance analysis tools**
 - parameterizable simulator
 - statistical and cycle-accurate simulation
 - models VLIW and in-order superscalar architectures
 - expanded tools to include out-of-order superscalar architectures

9

Dynamic vs. Static Architectures

10

Related Research

- **Media processors currently statically-scheduled**

- TI C6x
- TriMedia TM-1000, TM-2000
- Equator/Hitachi MAP1000

- **Research-based media processors**

- [CLee97] "MediaBench: A Tool for Evaluating and Synthesizing Multimedia Communications Systems," MICRO-30, 1997.
- [CLee98] "Media Architecture: General Purpose vs. Multiple Application-Specific Programmable Processors," DAC-35, 1998.
- [PPirsch97] "On Implementation of Media Processors," IEEE Signal Processing Magazine, vol. 14, no. 4, July 1997.
- [SRixner99] "Media Processors Using Streams," SPIE Photonics West – Media Processors '99, 1999.

- **Static vs. dynamic scheduling**

- [PChang91] "Comparing Static and Dynamic Code Scheduling for Multiple-Instruction Issue Processors," MICRO-24, 1991.

11

Base Architecture Model

- **Architecture model**

- 8-issue media processor
- operation latencies targeting 500 MHz to 1 GHz processor frequency
- 64 integer and floating-point registers
- pipeline: 1 fetch, 2 decode, 1 write back, variable execute stages
- 1024-entry 2-bit branch predictor

- **L1 Cache**

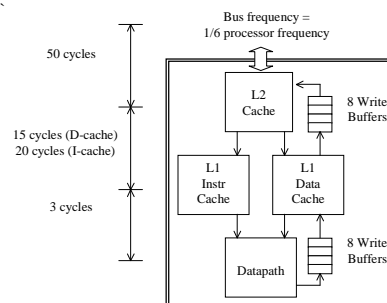
- 16 KB direct-mapped L1 instruction cache w/ 256 byte lines
- 32 KB direct-mapped L1 data cache w/ 64 byte lines

- **On-Chip L2 Cache**

- 256 KB 4-way set associate w/ 64 byte lines

- **External Memory**

- 6:1 Processor to bus frequency ratio

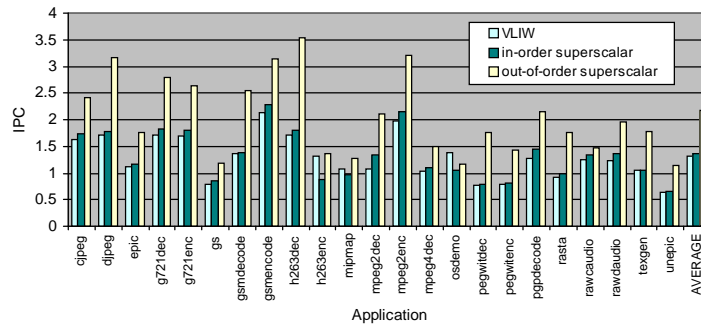


12

Static vs. Dynamic Scheduling

- Architectures for static and dynamic scheduling

- VLIW and in-order superscalar perform comparably (5% difference)
- out-of-order superscalar has 64% better performance on average
 - out-of-order issue with 32-entry issue-reorder buffer
 - early branch evaluation
 - large degree of dynamic control speculation

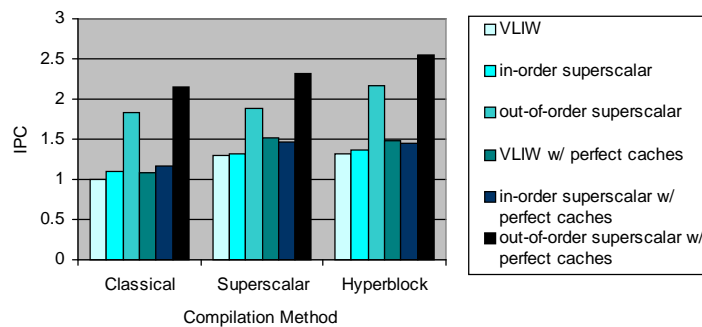


13

Scheduling Variations across Compiler Methods

- Compared compilations models across architectures

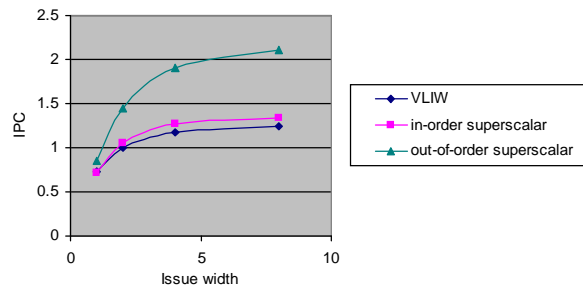
- hyperblock demonstrates best performance
 - 12% increase over superblock on out-of-order superscalar
 - only 2% increase over superblock otherwise
 - gain likely does not warrant resources for predication



14

Scheduling Variations across Processor Widths

- **Compared processor widths across architectures**
 - performance gain minimizes after 4 issue slots
 - 3-4 issue slots sufficient for these compiler methods
 - 2-issue out-of-order superscalar outperforms 8-issue VLIW and 8-issue in-order superscalar



15

Effects of High Frequency

16

Impact of Higher Frequencies

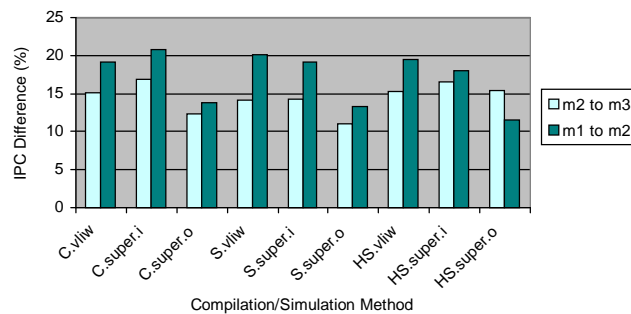
- **Increasing frequency**
 - Causes greater wire delays and fewer levels of logic per cycle
 - Leads to:
 - deeper pipelines
 - longer operation latencies
 - increased communication costs
- **Compared three different processor frequency models**
- **Compared immediate vs. delayed bypassing**

Instruction	Model 1	Model 2 (Base)	Model 3
Frequency Range	250-500 MHz	500 MHz – 1 GHz	1-2 GHz
Processor-Bus Freq. Ratio	4:1	6:1	8:1
ALU	1	1	1
Branches	1	1	1
Store	1	2	3
Load	2	3	4
Floating-Point	3	4	5
Multiply	3	5	7
Divide	10	20	30

17

Comparison of Frequency Models

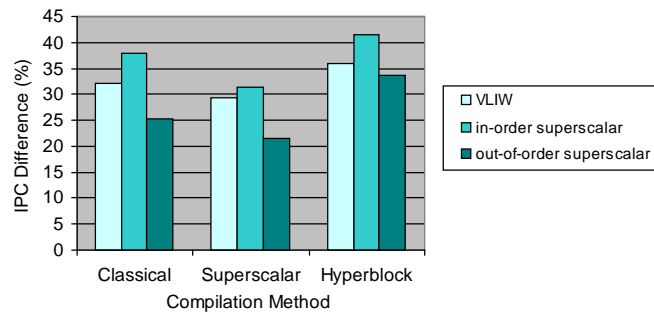
- **Results from doubling processor frequency**
 - average IPC degradation of 15%
 - 2/3 of degradation from longer operation latencies
 - 1/3 of degradation from longer memory latencies
 - performance increase of 70% from doubling frequency
 - out-of-order superscalar and superscalar compilation least susceptible to IPC degradation at higher frequencies



18

Impact of Delayed Bypassing

- **Results from delaying bypassing one cycle**
 - average IPC degradation of 32%
 - out-of-order superscalar and superscalar compilation least susceptible to IPC degradation



19

Conclusions

- **VLIW and in-order superscalar perform comparably**
 - Only 5% average difference in performance
- **Out-of-order superscalar has significantly higher performance**
 - 64% better average performance than VLIW
 - 2-issue out-of-order superscalar outperforms both 8-issue VLIW and 8-issue in-order superscalar
- **Compilation and Processor Width**
 - Hyperblock compilation is best, but likely not worth overhead
 - Processor widths of 3-4 issue slots sufficient for these compilation methods
- **Effects of High Frequency**
 - Doubling processor frequency decreases IPC by 16%
 - Delayed bypassing decreases IPC by 32%
 - Out-of-order scheduling and superscalar compilation up to 30% less susceptible to high frequency effects

20

Areas for Future Work

- **Advanced Compilation Methods**
 - Software pipelining
- **Impact of Subword Parallelism**
 - Current work only evaluates scheduling mechanisms on ILP-based code
 - How does inclusion of subword parallelism affect performance?
 - Anticipate greater impact from dynamic aspects:
 - Subword parallelism primarily used across loop iterations with regular control flow
 - Subword parallelism reduces regularity, giving dynamic aspects greater weight
- **Evaluating DSP Features**
 - DSP operations: multiply-accumulate, saturation arithmetic, etc.
 - Low-overhead looping
- **Evaluate Performance with Specialized Functional Units**
 - Motion estimation, DCT, variable-bit rate coding, etc.
 - Support specialized media functions with reconfigurable co-processor?

21