Performance Analysis of Kernel Benchmarks for Tiled Architectures

James Lebak
Ryan Haney, Matt Alexander,
Hector Chan, Edmund Wong
Massachusetts Institute of Technology
Lincoln Laboratory

High Performance Embedded Computing Workshop
(HPEC 2005)
22 September 2005

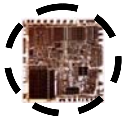
This work is sponsored by the Defense Advanced Research Projects Agency under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

Outline

- Introduction to Tiled Architectures
- Measuring Performance
- Results and Analysis

Microprocessor Design Evolution

- Number of gates that can communicate in one cycle has remained roughly constant
  - Not a consideration for early designs

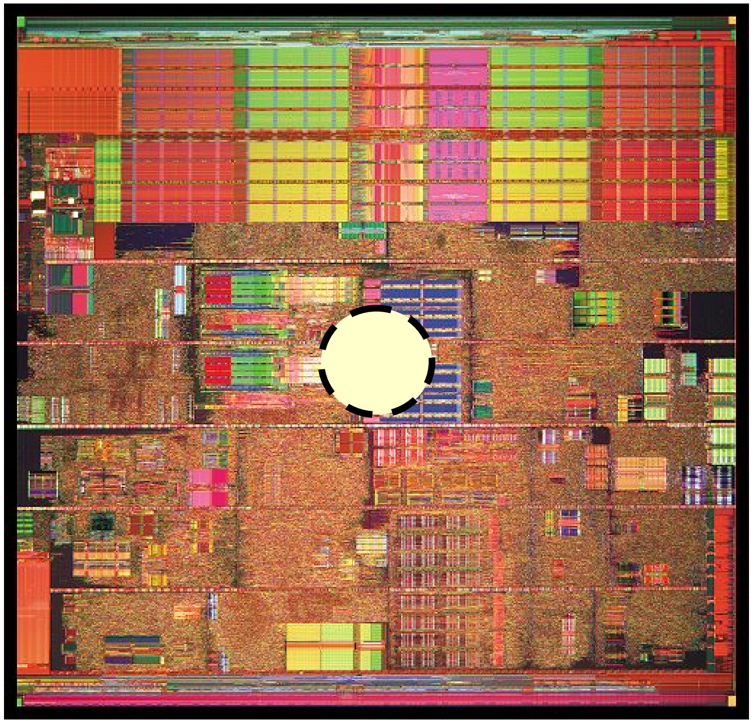


- Intel 8088
  - 29,000 transistors
  - 3-micron technology
  - 5 MHz clock rate

See Ho, Mai, Horowitz, “The Future of Wires,” *Proc. IEEE* 89(4) Apr 2001.

Microprocessor Design Evolution

- Number of gates that can communicate in one cycle has remained roughly constant
  - Not a consideration for early designs
  - Much more important now!
- Preserving a uniprocessor programming model requires
  - Complex control hardware
  - Deep pipelines to hide delays



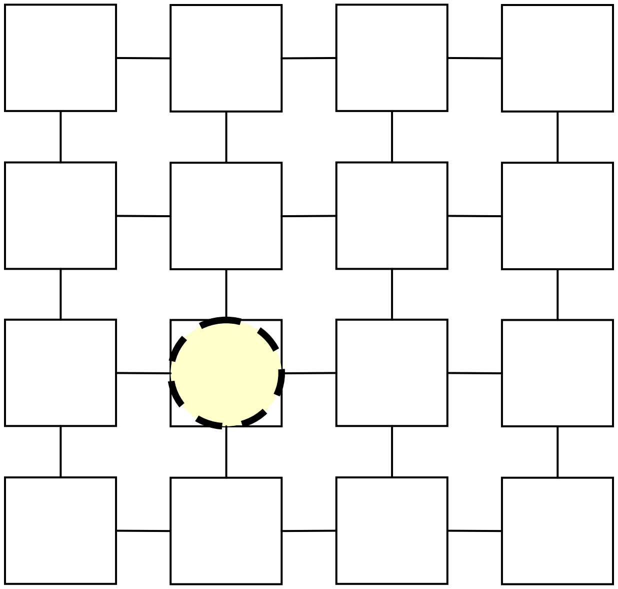
Not to scale...

- Intel Pentium 4
  - 125 Million transistors
  - 90 nm technology
  - 3.2 GHz clock rate
  - 103 W

See Ho, Mai, Horowitz, "The Future of Wires," *Proc. IEEE* 89(4) Apr 2001.

Microprocessor Design Evolution

- Number of gates that can communicate in one cycle has remained roughly constant
  - Not a consideration for early designs
  - Much more important now!
- Preserving a uniprocessor programming model requires
  - Complex control hardware
  - Deep pipelines to hide delays
- Tiled architectures expose the delays and the parallelism to the software
  - Simpler hardware
  - More complex software

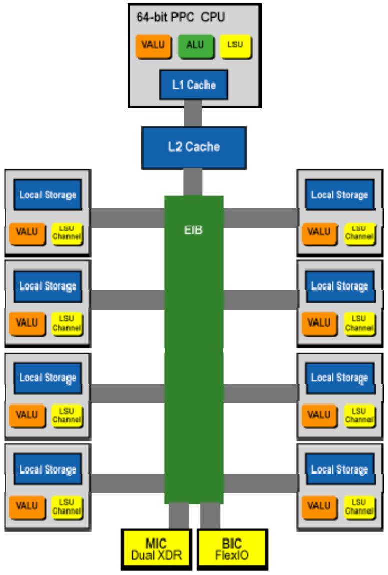


Not to scale...

See Ho, Mai, Horowitz, “The Future of Wires,” *Proc. IEEE* 89(4) Apr 2001.

Example Tiled Architectures

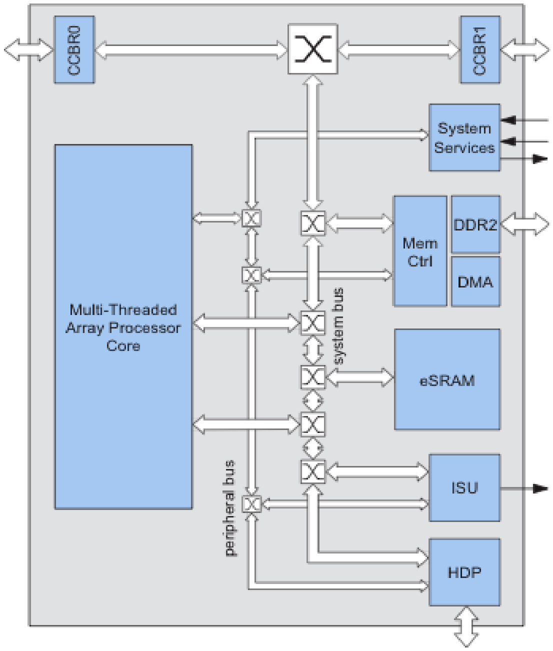
Cell Processor (IBM/Sony/Toshiba)



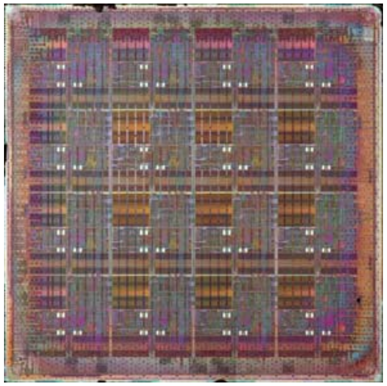
The CELL Architecture

8 “synergistic processing elements” with vector processing capabilities

Clearspeed



96 ALUs in Array Processor Core

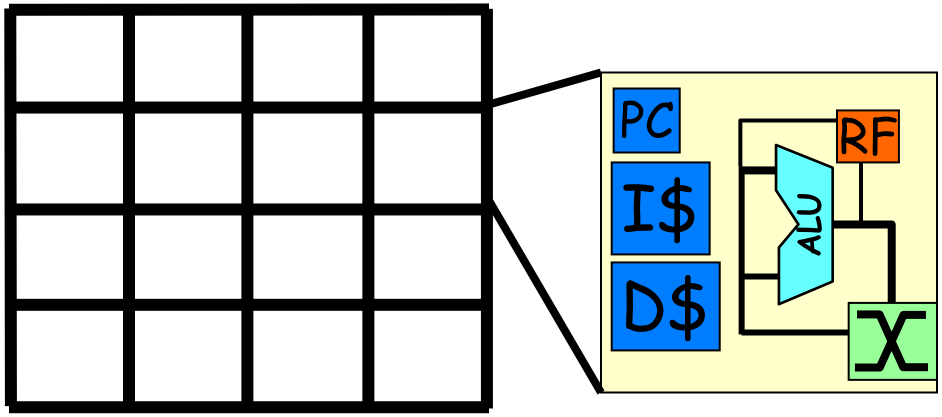


RAW (MIT)

- RAW is a 4x4 array of tiles
  - Small amount of memory per tile
  - *Scalar operand network* allows delivery of operands between functional units
  - MIT and USC/ISI are building a 1024-tile RAW fabric

Tiled Architectures

- Two views of a tiled architecture
  - Exposed instruction-level parallel machine
    - Compiler exploits parallelism
  - Multiprocessor on a chip
    - Programmer and library exploit parallelism



Opportunity: Co-optimize program (software) and use of tiles (hardware)

- *Scalable.* If our problem is big enough, performance should improve as the number of tiles increases.
- *Flexible.* Meet different application requirements with the same resources.
- *High performance per cycle.* Utilize parallelism to achieve performance as good as conventional superscalar processors but with a lower clock rate.

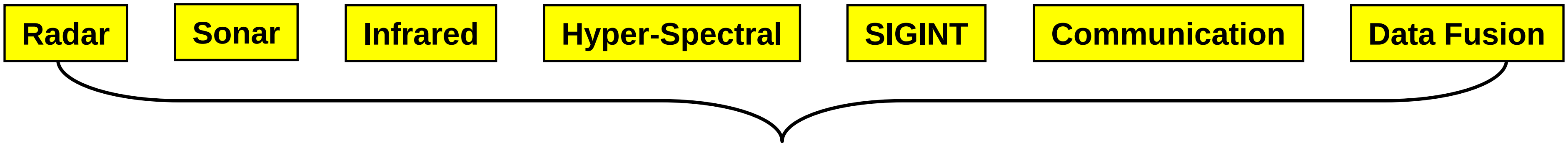
Outline

- Introduction to Tiled Architectures
- Measuring Performance
- Results and Analysis

Measuring Performance

- **Identify kernel benchmarks from DoD application survey**
- Measure performance on conventional architectures
- Map kernels to Raw
- Measure performance on Raw board

Specific Application Areas



Signal/Image Processing	Communication	Information/Knowledge Processing
• FIR Filter • QR/SVD • CFAR Detection	• Corner Turn	• Graph Optimization • Pattern Recognition • Real-time Database Operations

<u>MIT-LL Surveyed DoD Applications to Provide:</u>

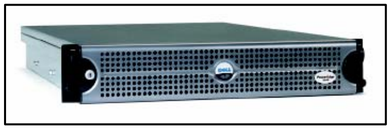
- **Kernel Benchmark Definitions**
- **Example Requirements and Data Sets**

These Kernels are part of the “HPEC Challenge” Benchmark Suite

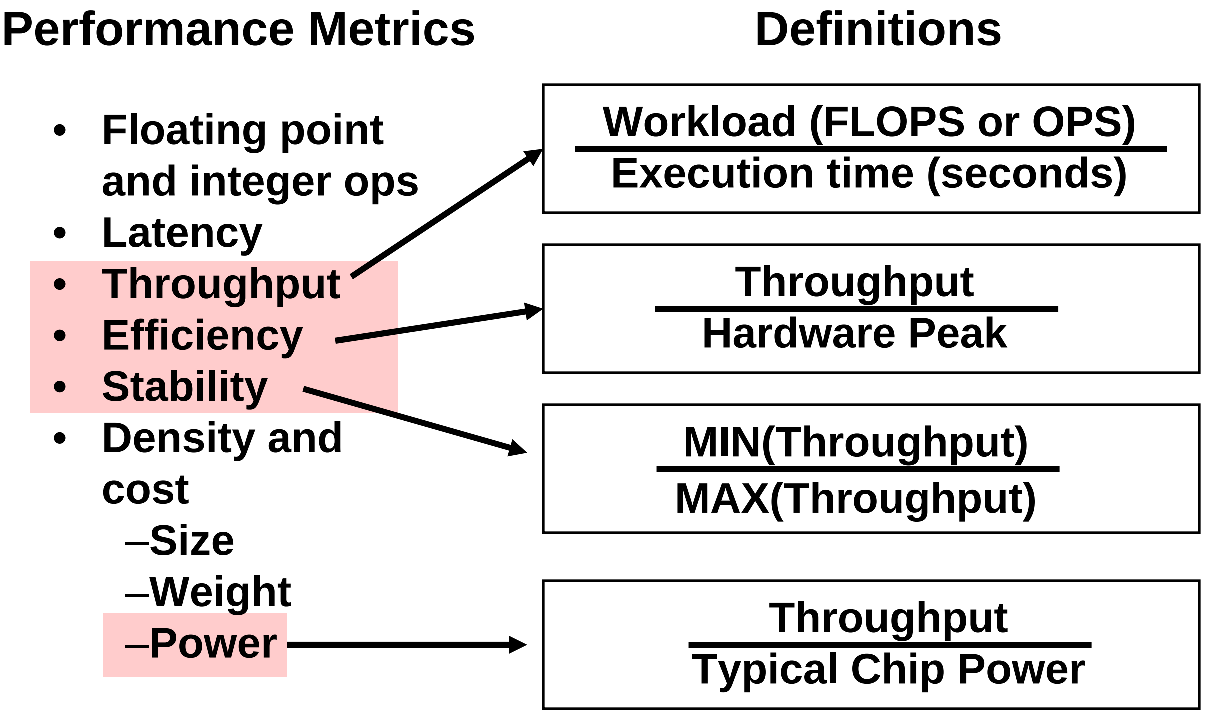
Measuring Performance

- Identify kernel benchmarks from DoD application survey
- **Measure performance on conventional architectures**
- Map kernels to Raw
- Measure performance on Raw board

Power PC G4
500 MHz Mercury



Xeon
2.8 GHz Dell PowerEdge



Performance Metrics

- Floating point and integer ops
- Latency
- Throughput
- Efficiency
- Stability
- Density and cost
  - –Size
  - –Weight
  - –Power

Definitions

$$\frac{\text{Workload (FLOPS or OPS)}}{\text{Execution time (seconds)}}$$

$$\frac{\text{Throughput}}{\text{Hardware Peak}}$$

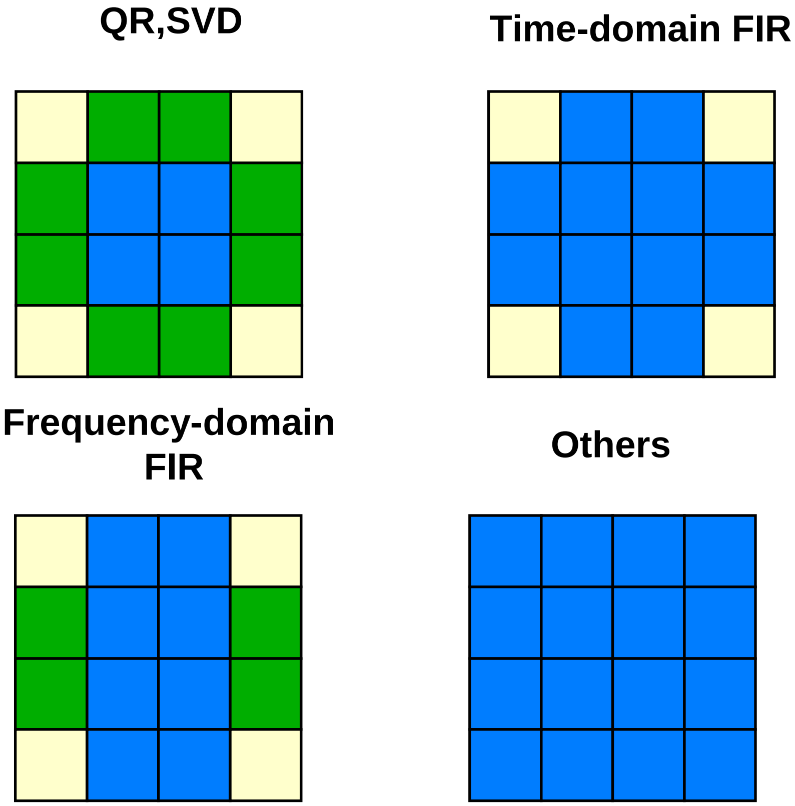
$$\frac{\text{MIN(Throughput)}}{\text{MAX(Throughput)}}$$

$$\frac{\text{Throughput}}{\text{Typical Chip Power}}$$

Measuring Performance

- Identify kernel benchmarks from DoD application survey
- Measure performance on conventional architectures
- **Map kernels to Raw**
- Measure performance on Raw board

- **Co-optimization of hardware, software**
- **Signal processing kernels use scalable “stream algorithm” approach [Hoffmann]**
  - **QR, SVD kernels use 2x2 area in chip center for computation**
  - **Time-domain convolution uses 12 of 16 tiles for computation**
  - **Frequency-domain convolution uses 8 of 16 tiles for computation**
- **Other kernels use a data-parallel approach**


QR,SVD

Time-domain FIR

Frequency-domain FIR

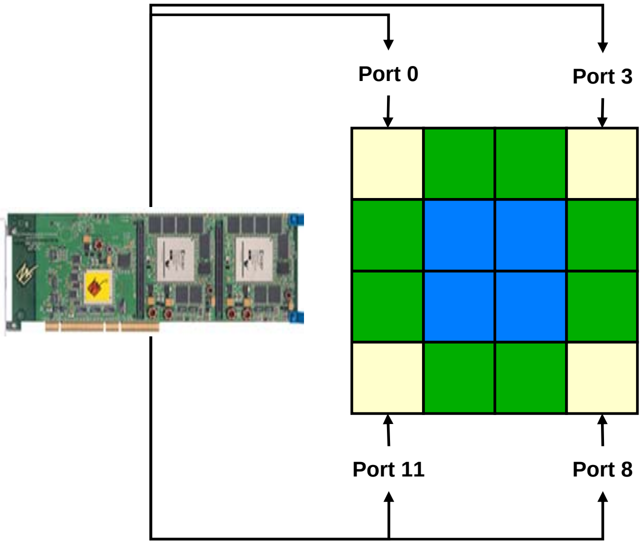
Others

🟦 Compute Tiles 🟩 Memory Tiles 🟨 I/O Tiles

Measuring Performance

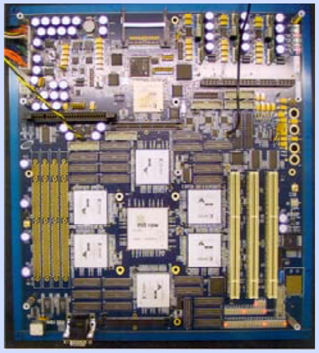
- Identify kernel benchmarks from DoD application survey
- Measure performance on conventional architectures
- Map kernels to Raw
- **Measure performance on Raw board**

- **Raw clocked at 100 MHz with current board firmware**
  - **Chip could run at 425 MHz**
- **Streaming interface built by MIT/LL**
  - **Allows direct access to on-chip networks**



Raw Test Board

- **2 GB DRAM**
- **Expansion FPGAs**
- **USB Interface**
- **High Speed A/D**



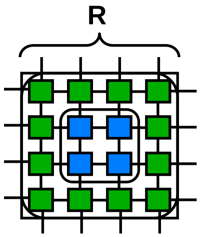
Outline

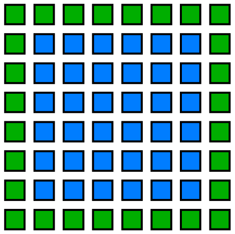
- Introduction to Tiled Architectures
- Measuring Performance
- Results and Analysis
  - Scalability
  - Flexibility
  - Overall Performance

Scaling

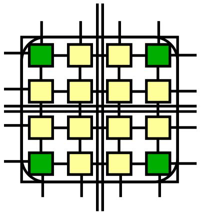
- A key feature of tiled architectures is that they are scalable
  - The Raw simulator includes the ability to increase the number of tiles
- We modified two kernels to run on the Raw simulator at 8x8

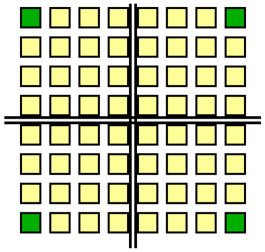
- Fast Givens QR factorization
  - Stream algorithm, from Hoffmann
  - Matrix streamed in columnwise
  - Factorization computed in a systolic fashion
  - Inner tiles compute 🟦
  - Outer tiles manage memory 🟩
  - Requires matrix size N > R



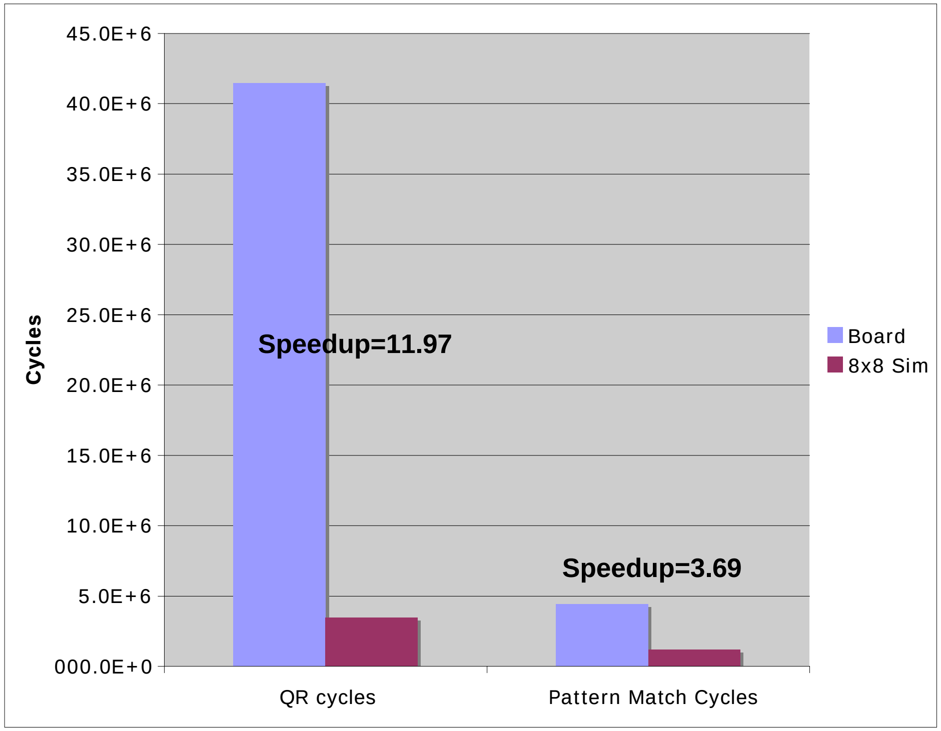


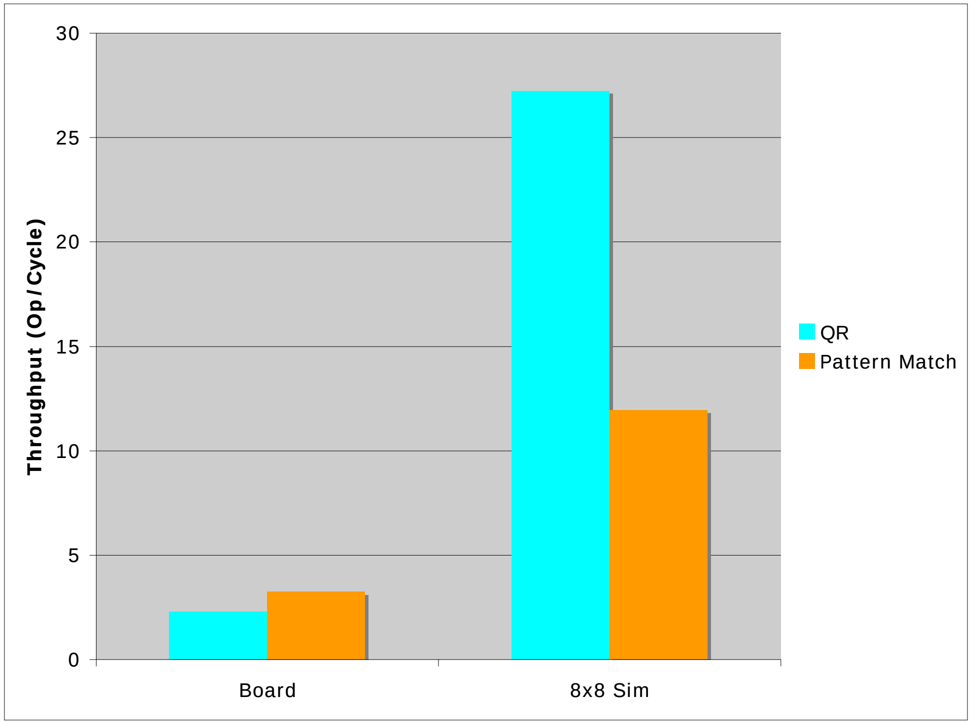
- Pattern Match
  - Matches a test pattern against a library of patterns
  - Library patterns streamed in to corner tiles 🟩
  - Corner tiles distribute library patterns to worker tiles 🟨
  - Each worker tile compares the test pattern to a number of library patterns
  - Requires library size K > R^2





Kernel Scaling

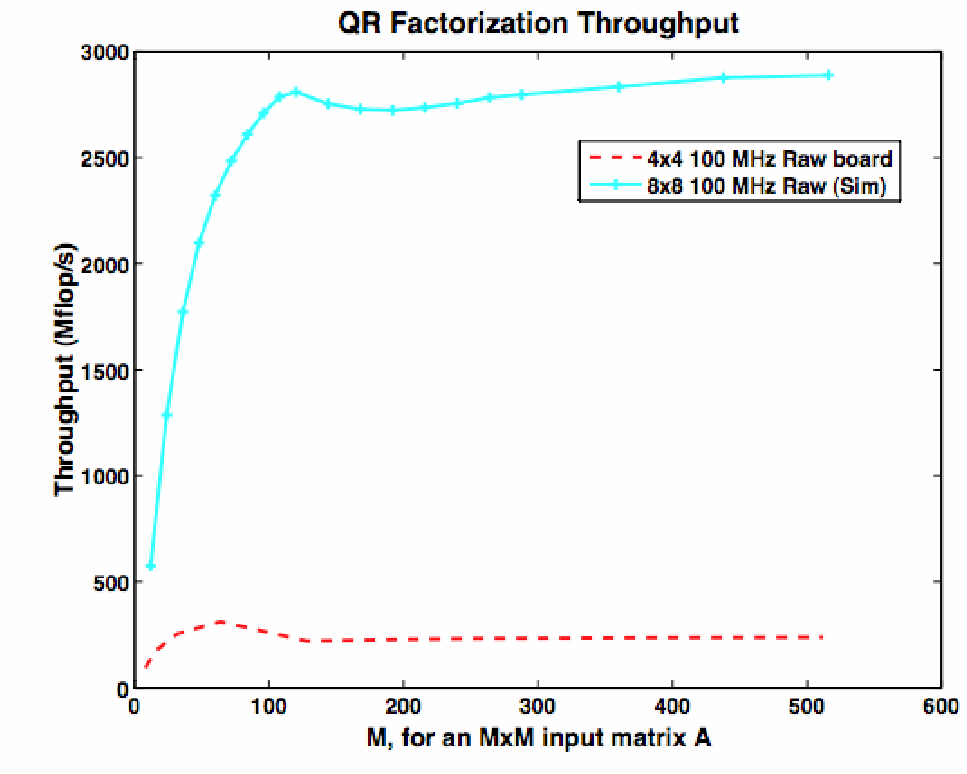




- **Compare 8x8 simulator and 4x4 Raw board results**
- **QR factorization of 192x192 matrix**
  - **36 compute tiles vs 4 (9X)**
  - **Simulator predicts 33% higher efficiency on compute tiles in 8x8 case**
- **Pattern match with library of 256 patterns, length 128**
  - **64 compute tiles vs. 16 (4X)**
  - **Simulator predicts 10% higher efficiency in 4x4 case**

Throughput of Scaled QR Factorization

Simulator and Board Results



Compare QR factorization throughput on

- 4x4 Raw Board @ 100 MHz
- 8x8 Raw Simulator @ 100 MHz

Increased performance on simulator due to

- More compute tiles
- More memory bandwidth

System	Throughput for 128x128
4x4 Raw@100 MHz	220 Mflop/s
8x8 Raw@100 MHz	2810 Mflop/s

Tiled architectures can exhibit scalable performance for a range of data sizes

Outline

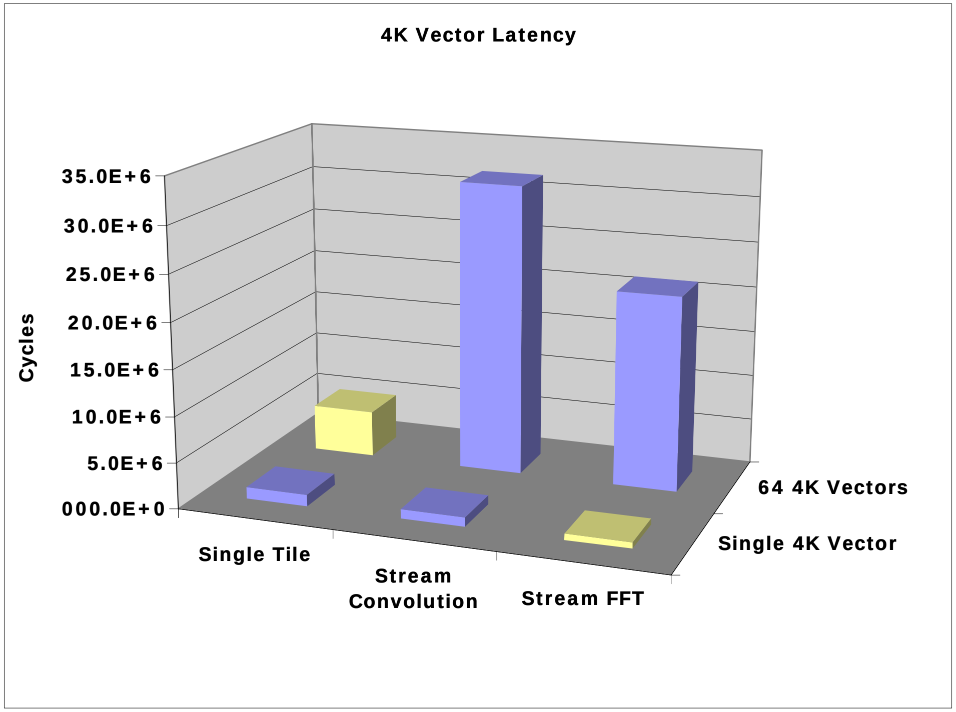
- Introduction to Tiled Architectures
- Measuring Performance
- Results and Analysis
  - Scalability
  - Flexibility
  - Overall Performance

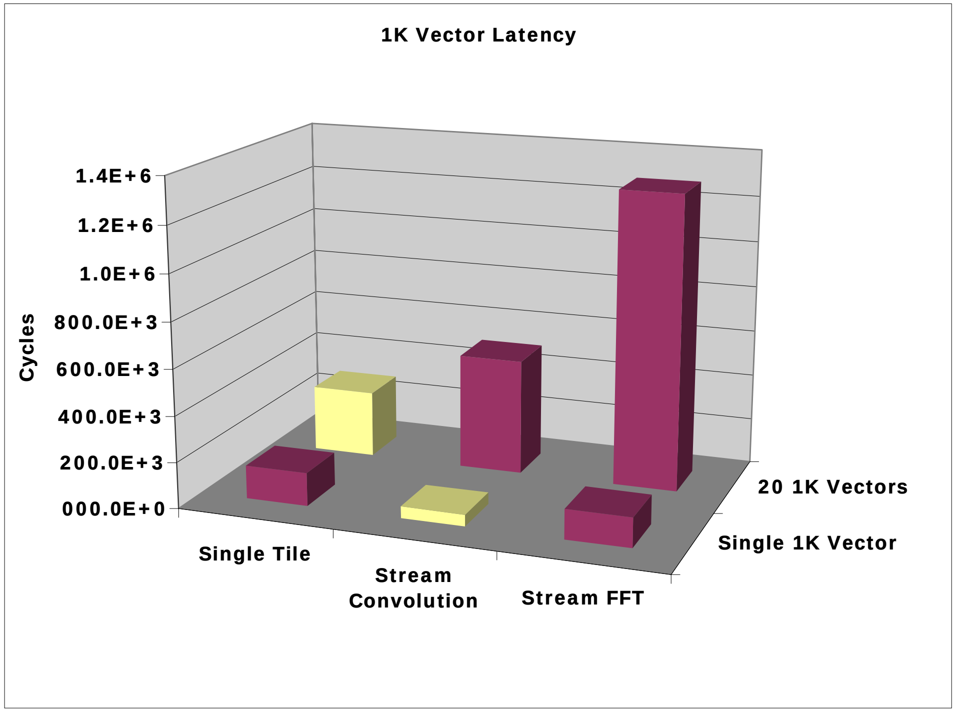
FIR Filter Implementations

Implementation	Tiles per filter	Frequency-domain?	Performs bit-reverse?	Implementation
Single Tile*	1	Yes	No	C+Assembly
Stream Convolution	6	No	N/A	C+Assembly
Stream FFT	4	Yes	Yes	C+Assembly

- Compare three implementations of FIR filter
- *Single Tile implementation and results provided by Jinwoo Suh, USC/ISI-East
  - Uses Overlap-and-Save convolution to reduce operation count
- Stream FFT and Convolution by MIT/LL
  - Multi-tile implementations based on work by Hank Hoffmann

FIR Filter Latency Comparison





🟨 = best implementation for a given data set

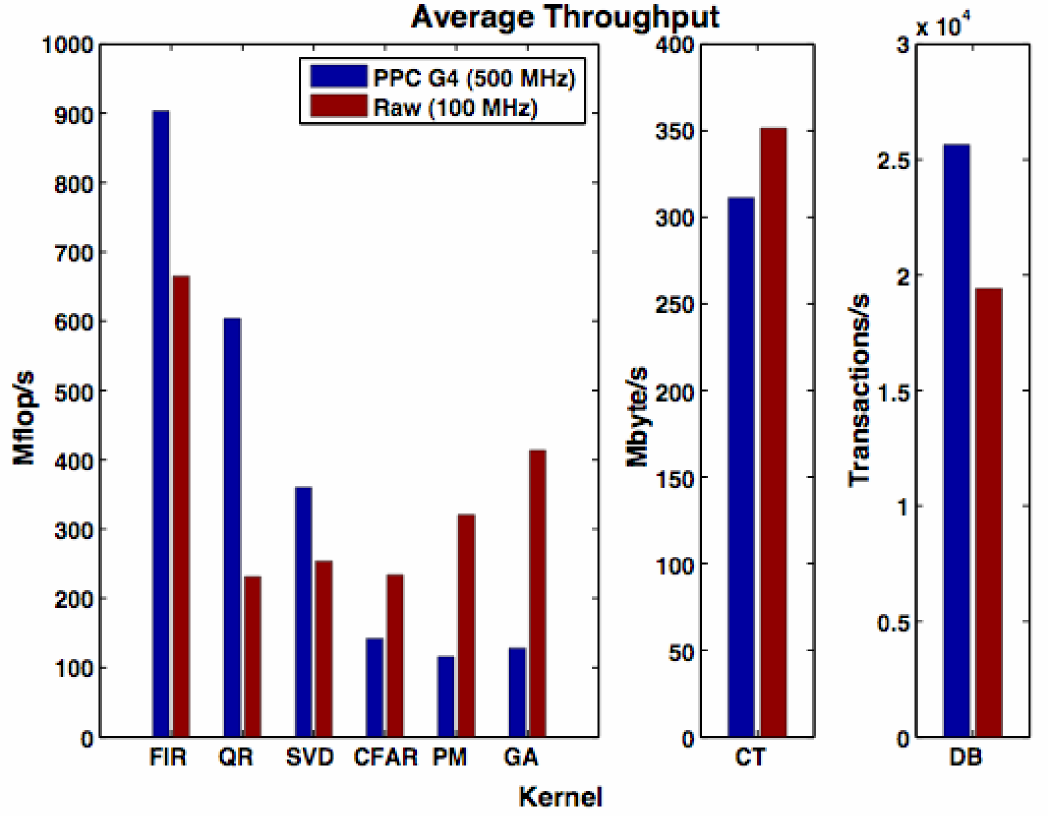
- Compare FIR on four different data sets
- Lowest-latency implementation depends on data set
- Raw is flexible
  - Supports many choices of implementation
  - Application requirements determine the “best” use of the architecture

Outline

- Introduction to Tiled Architectures
- Measuring Performance
- Results and Analysis
  - Scalability
  - Flexibility
  - **Overall Performance**

Raw Kernel Performance (1)

- 100 MHz results obtained on the Raw board
  - FIR results courtesy of USC/ISI
- G4 7410 results on Mercury hardware
  - FIR uses MSTI VSIPL
  - QR, SVD, Corner turn use AltiVec instructions

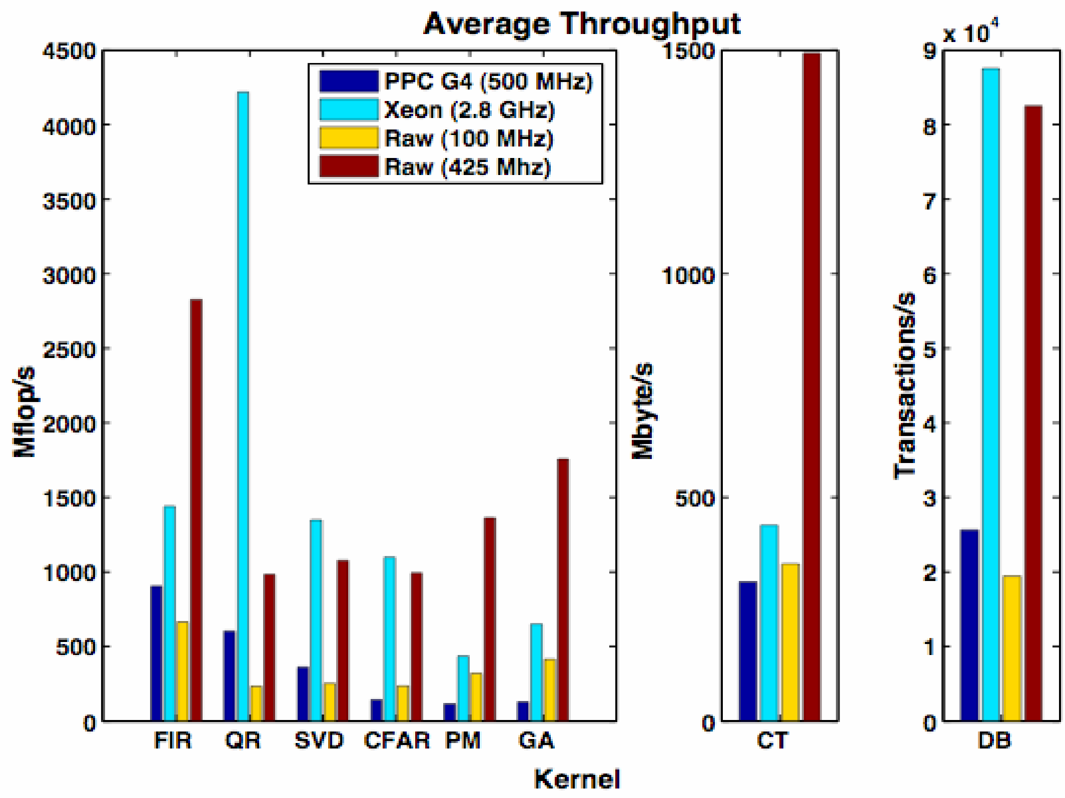


	G4	Raw
Clock (MHz)	500	100
Peak (Gflop/s)	4	1.6

- Raw shows consistent high performance across different kernels

Raw Kernel Performance (2)

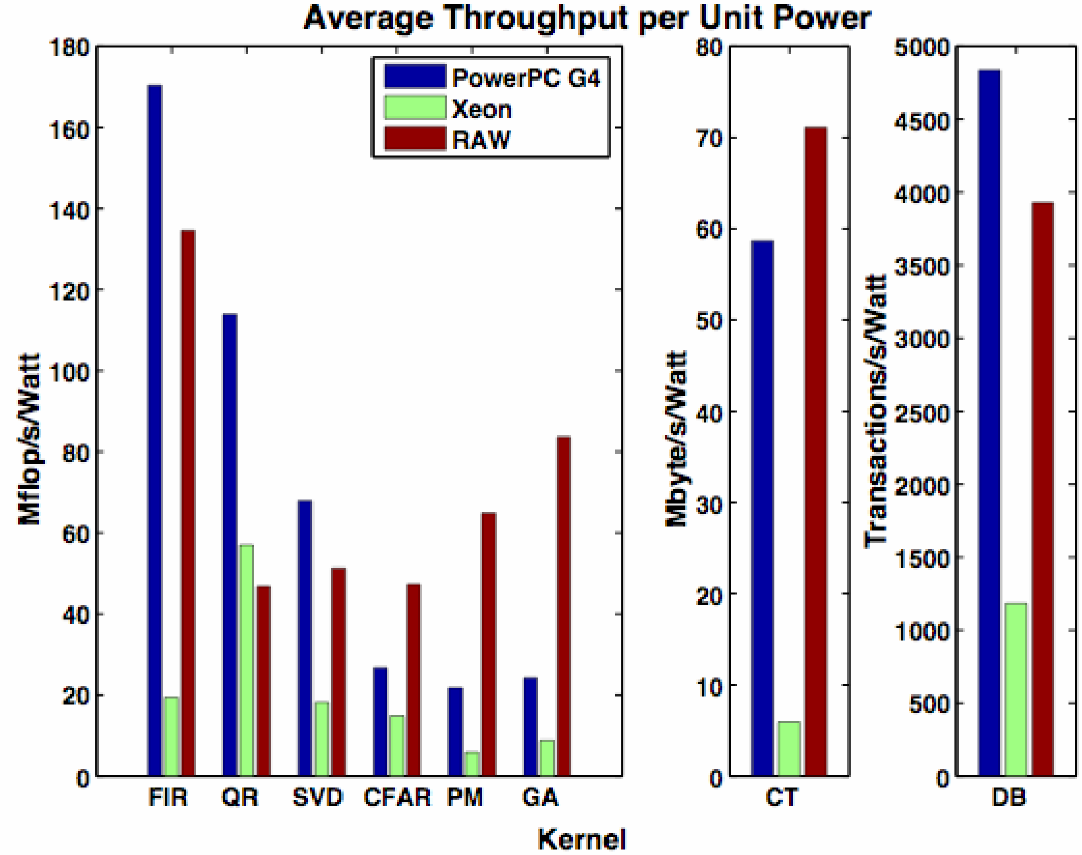
- 100 MHz results obtained on the Raw board
  - FIR results courtesy of USC/ISI
- 425 MHz results based on scaling Raw board results
  - Assumes FPGAs, memory can all keep up with Raw
- Xeon kernels use SSE



	Xeon	G4	Raw
Clock (MHz)	2800	500	100
Peak (Gflop/s)	11.2	4	1.6
Tech (µm)	0.13	0.18	0.18

Raw Kernel Performance (3)

- G4 is designed for embedded systems
- Xeon is a designed for servers
- Raw is not a power-optimized design

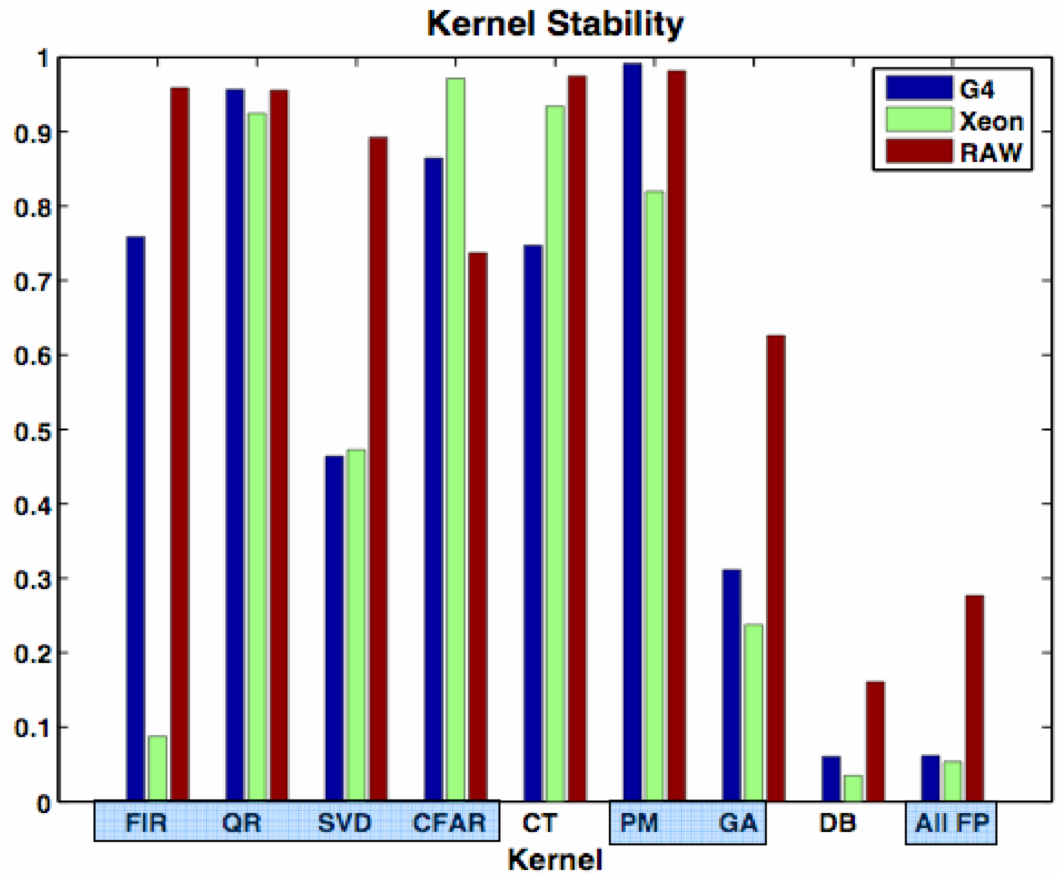


	Xeon	G4	Raw
Clock (MHz)	2800	500	100
Peak (Gflop/s)	11.2	4	1.6
Tech (µm)	0.13	0.18	0.18
Power (W)	74	5	5

- Raw is competitive for all kernels in throughput per watt
  - Despite not being a power-optimized design

Raw Kernel Performance (4)

- Stability: Ratio of minimum throughput to maximum throughput
- Compare Raw's stability to Xeon and G4
  - Same kernels
  - Same data sets



	Xeon	G4	Raw
Clock (MHz)	2800	500	100
Peak (Gflop/s)	11.2	4	1.6
Tech (µm)	0.13	0.18	0.18
Power (W)	74	5	5

- Compared to conventional architectures, Raw shows
  - Similar stability per-kernel
  - Greater stability over all floating-point kernels

Kernel Performance Summary

- Raw’s performance on streaming kernels scales with the number of tiles
  - Requires co-optimization of hardware and software
- The flexibility of Raw enables
  - Multiple implementations to fit application requirements
  - More consistent performance across different kernels

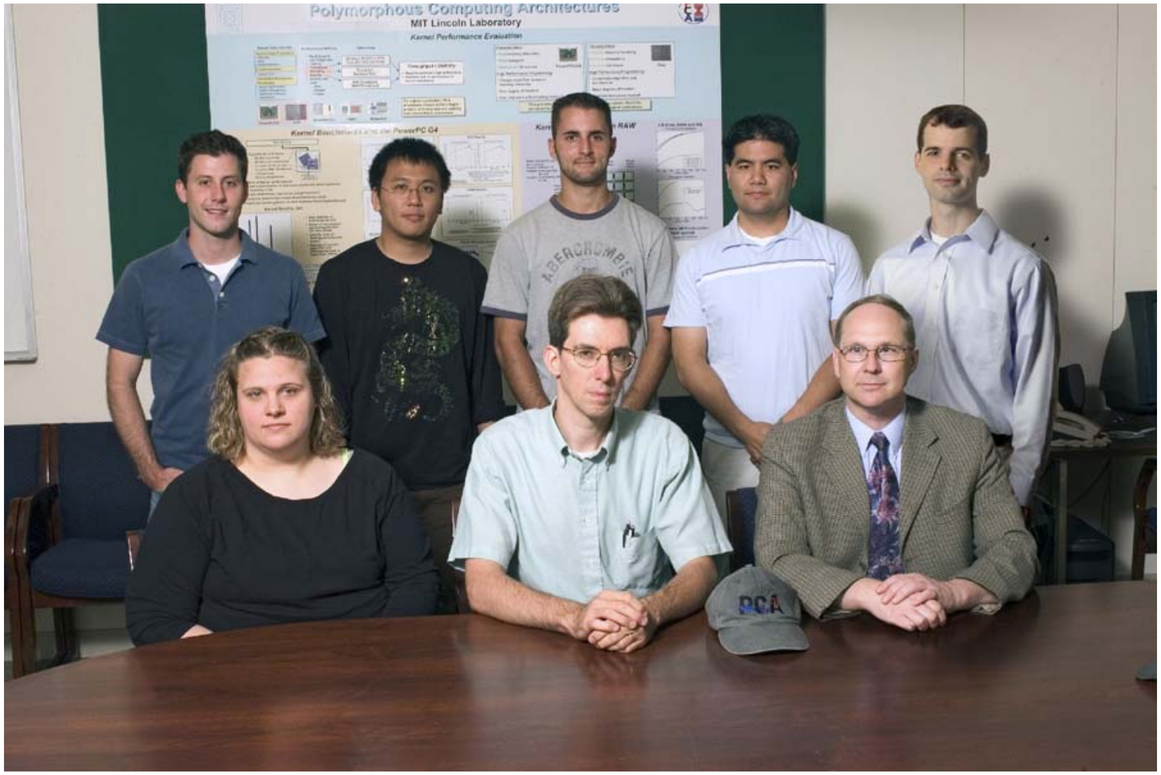
    Raw’s overall stability is 0.28 (G4: 0.062, Xeon: 0.053)
- On floating-point kernels, the 425 MHz Raw and an appropriately scaled board would be expected to deliver:
  - Avg Throughput of 1.50 Gflop/s (G4: 0.37, Xeon:1.53)
  - Avg Power-performance density of 71 Mflop/s/W (G4: 71, Xeon: 21)

425 MHz Raw gives consistent high performance and performance per watt

Summary

- Tiled architectures are an increasingly common design trend
  - Several industrial and academic examples
- MIT Lincoln Lab benchmarked conventional architectures and the Raw tiled architecture
- Raw results demonstrate that tiled architectures are
  - *Scalable* to meet the needs of larger problems
  - *Flexible* to satisfy different application requirements
  - *High-performing* both in throughput and throughput per watt

Credits



Lincoln PCA Team

Back row: Ryan Haney, Hector Chan, Matt Alexander, Edmund Wong, Preston Jackson

Front row: Jeanette Baran-Gale, James Lebak, Robert Bond

Outside Lincoln:

- Raw Processor
  - Anant Agarwal and collaborators, MIT Computer Architecture Group
- Raw Board
  - MIT
  - Steve Crago and collaborators, USC ISI/East
- Raw FIR filter implementation
  - Jinwoo Suh, USC/ISI East
- Stream algorithm development
  - Hank Hoffmann, MIT
- Research Sponsor:
  - Robert Graybill, DARPA PCA Program