

Processor Architectures At A Glance: M.I.T. Raw vs. UC Davis AsAP

Presenter: Jeremy W. Webb

Course: EEC 289Q: Reconfigurable Computing

Course Instructor: Professor Soheil Ghiasi

Outline

- Overview of M.I.T. Raw processor
- Overview of UC Davis AsAP processor
- Raw vs. AsAP
- Conclusion
- References

M.I.T. Raw Processor

- Raw Project Goals
- M.I.T. raw Architecture Workstation (Raw) Architecture
- Raw Processor Tile Array
- What's in a Raw Tile?
- Raw Processor Tile
 - Inside the Compute Processor
 - Raw's Networking Routing Resources
 - Raw Inter-processor Communication
- M.I.T. Raw Contributions
- M.I.T. Raw Novel Features

Raw Project Goals

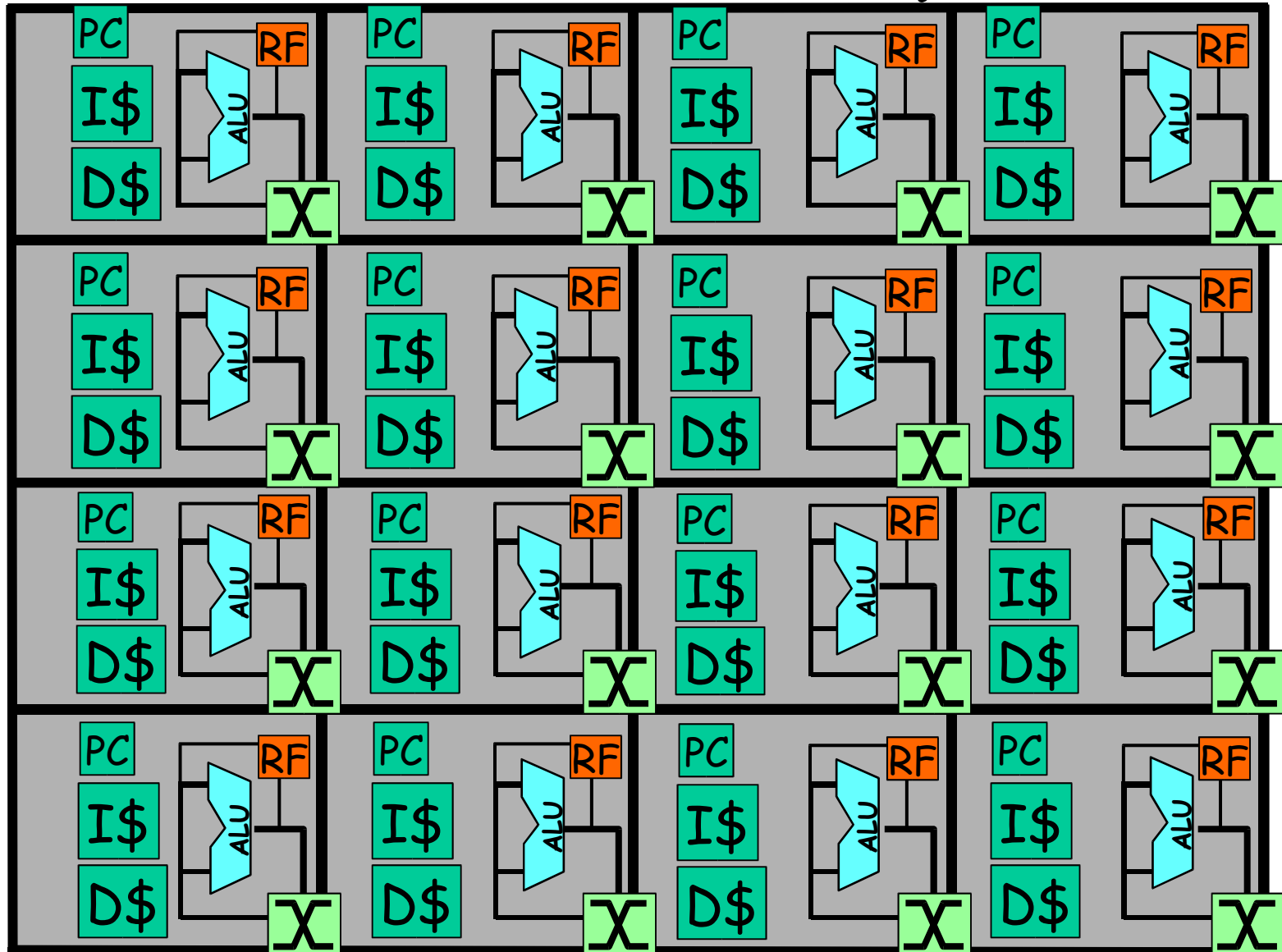
1. Create an architecture that scales to 100's-1000's of functional units, by exploiting custom-chip like features while being “general purpose”. [8,9]
2. Support standard general purpose abstractions like context switching, caching, and instruction virtualization.

M.I.T. raw Architecture Workstation (Raw) Architecture

- Composed of a replicated processor tile. [8]
- 8 stage Pipelined MIPS-like 32-bit processor [7]
- Static and Dynamic Routers
- Any tile output can be routed off the edge of the chip to the I/O pins.
- Chip Bandwidth (16-tile version).
 - Single channel (32-bit) bandwidth of 7.2 Gb/s @ 225 MHz.
 - 14 channels for a total chip bandwidth of 201 Gb/s @ 225 MHz.

Raw Processor Tile Array

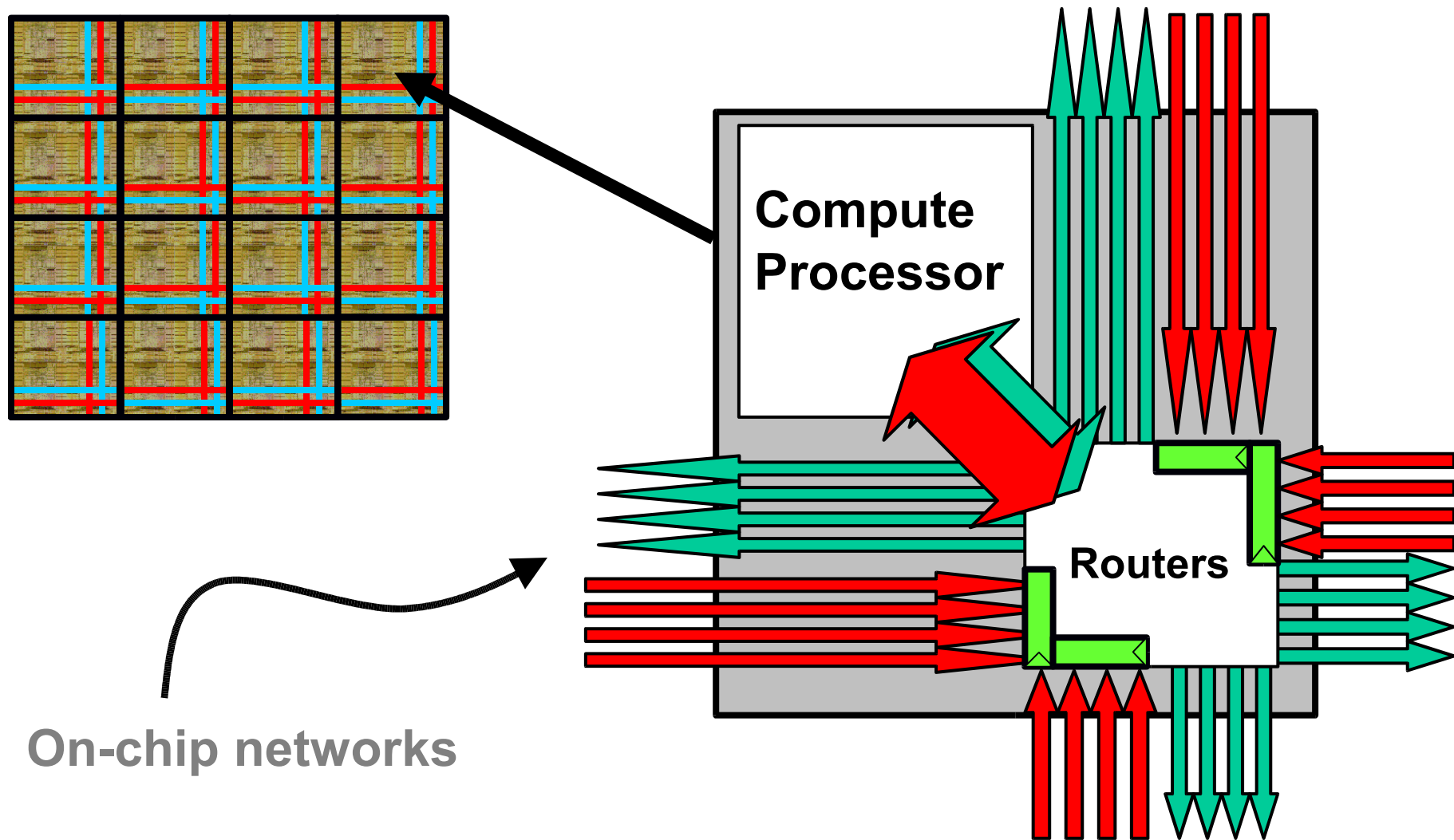
[8]



What's in a Raw tile?

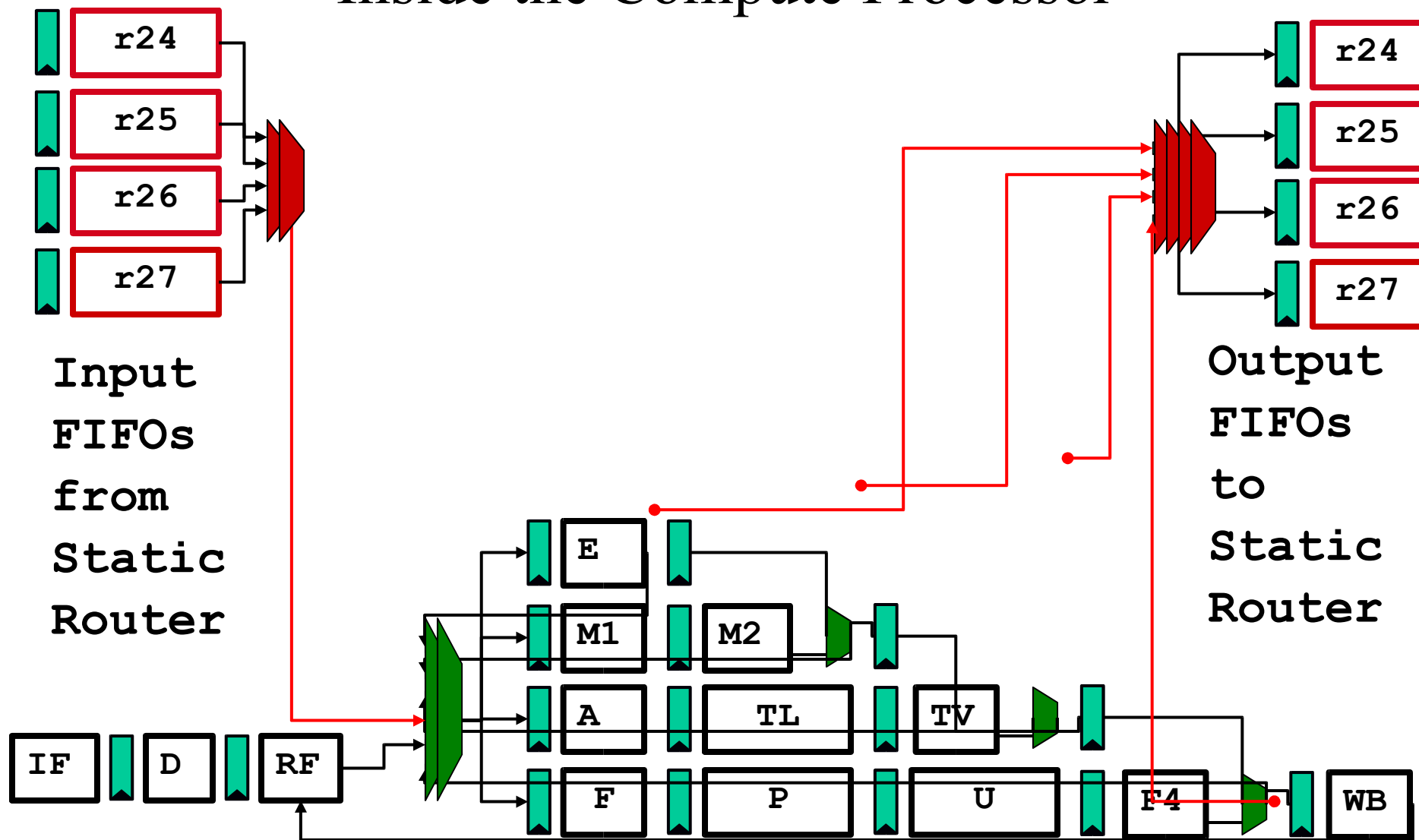
- 8 stage Pipelined MIPS-like 32-bit processor ^[7]
- Pipelined Floating Point Unit
- 32KB Data Cache
- 32KB Instruction Memory
- Interconnect Routers

Raw Processor Tile



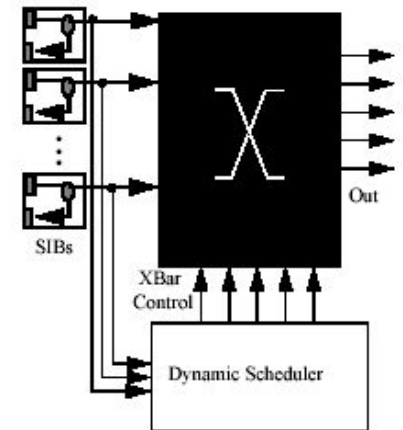
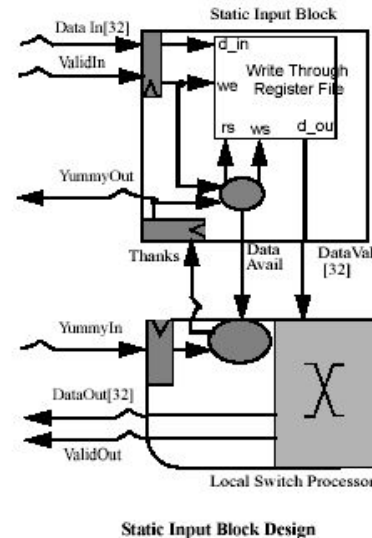
On-chip networks

Inside the Compute Processor



Raw's Networking Routing Resources

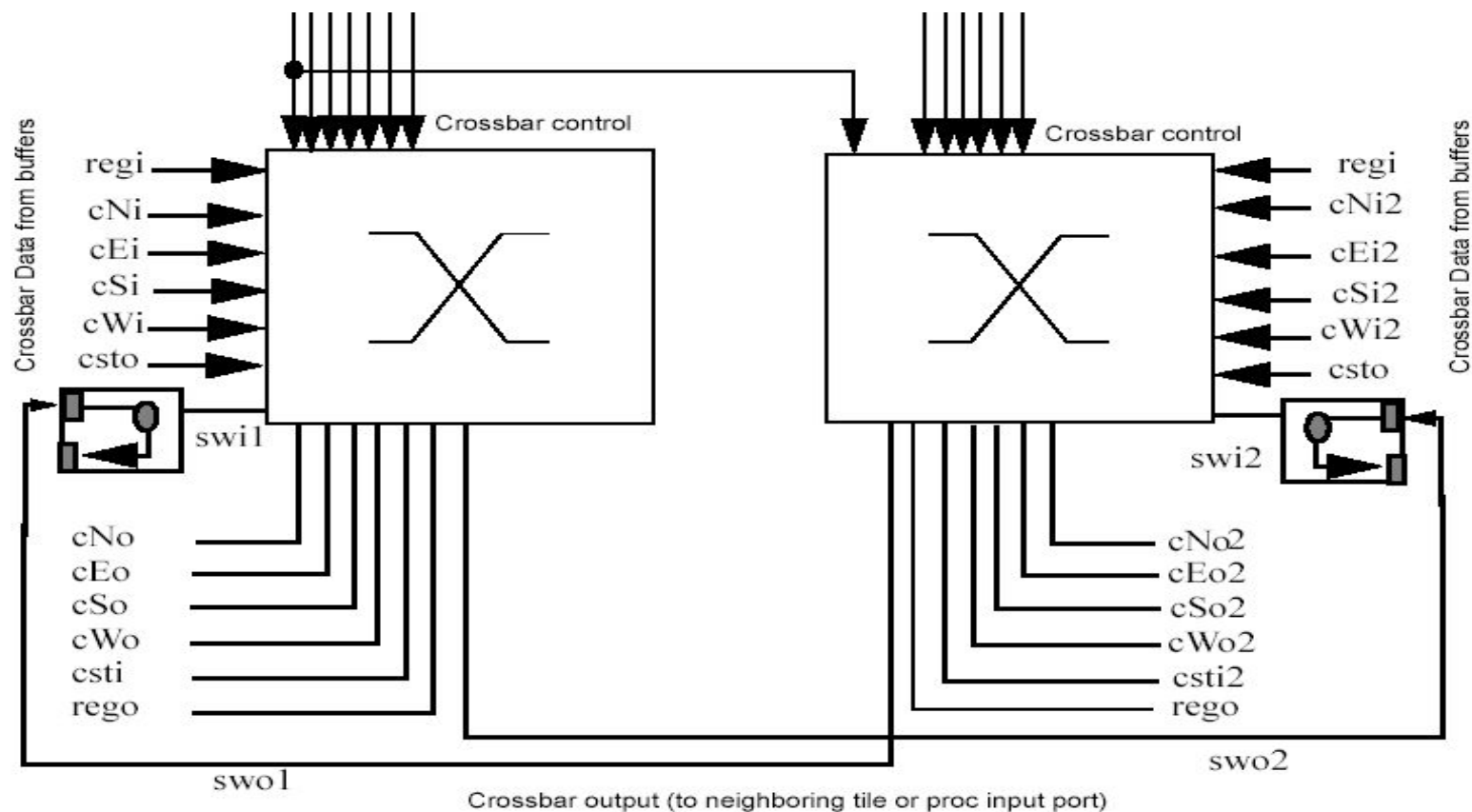
- 2 Dynamic Networks^[7]
 - Fire and Forget
 - Header encodes destination
 - 2 Stage router pipeline
- 2 Static Networks
 - Software configurable crossbar
 - Interlocked and Flow Controlled
 - 5 Stage static router pipeline
 - 3 cycle nearest-neighbor ALU to ALU communication latency
 - No header overhead, but requires knowledge of communication patterns at compile time



The Dynamic Network Router

Raw Inter-processor Communication

[5,6]



M.I.T. Raw Contributions

- Raw's communication facilitates exploitation of new forms of parallelism in Signal Processing applications ^[7]

M.I.T. Raw Novel Features

- Dynamic and Static Network Routers.
- Scalability of Raw chips.
 - Fabricated Raw chips can be placed in an array to further increase the system computing performance.
- Exposes the complete details of the underlying HW architecture to the SW system.

UC Davis AsAP Processor

- AsAP Project Goals
- UC Davis Asynchronous Array of simple Processors (AsAP) Architecture
- Asynchronous Array of simple Processors
- What's in an AsAP Tile?
- AsAP Single Processor Tile
- AsAP Contributions
- AsAP Novel Features

AsAP Project Goals

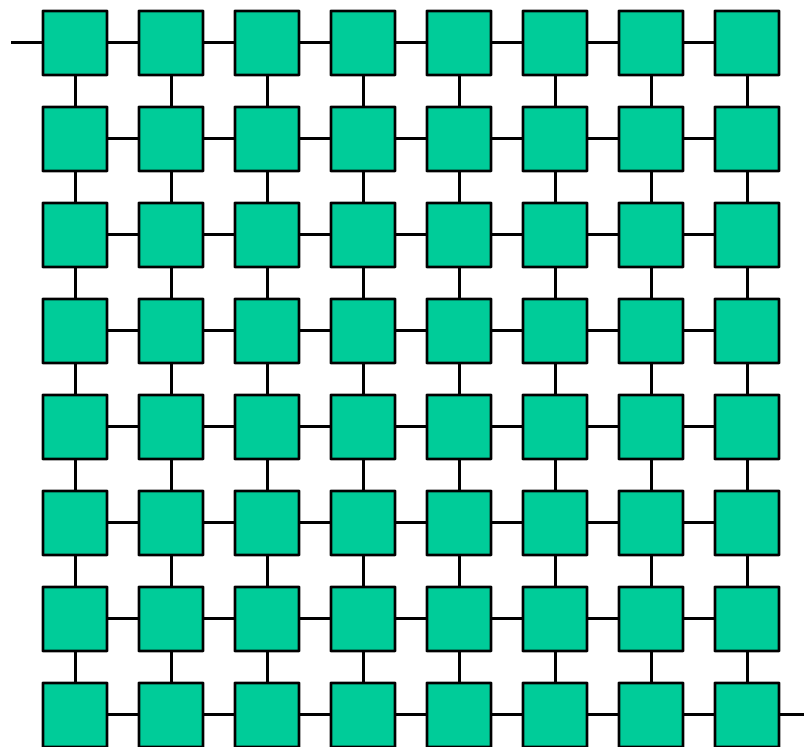
AsAP's proposed architecture targets four key goals: [3]

1. Well matched with DSP system workloads.
2. High-throughput.
3. Energy-efficient.
4. Address the opportunities and challenges of future VLSI fabrication technologies.

UC Davis Asynchronous Array of simple Processors (AsAP) Architecture

- Composed of a replicated processor tile.
- 9-stage pipelined reduced complexity DSP processor ^[2]
- Four nearest neighbor inter-processor communication.
- Individual processor tile can operate at different frequencies than its neighbors. ^[2]
- Off chip access to the I/O pins must be reached by routing to boundary processors.
- Chip Bandwidth
 - Single channel (16-bit) bandwidth of 16 Gb/s @ 800 MHz.
- The array topology of AsAP is well-suited for applications that are composed of a series of independent tasks. ^[2]
 - Each of these tasks can be assigned to one or more processors.

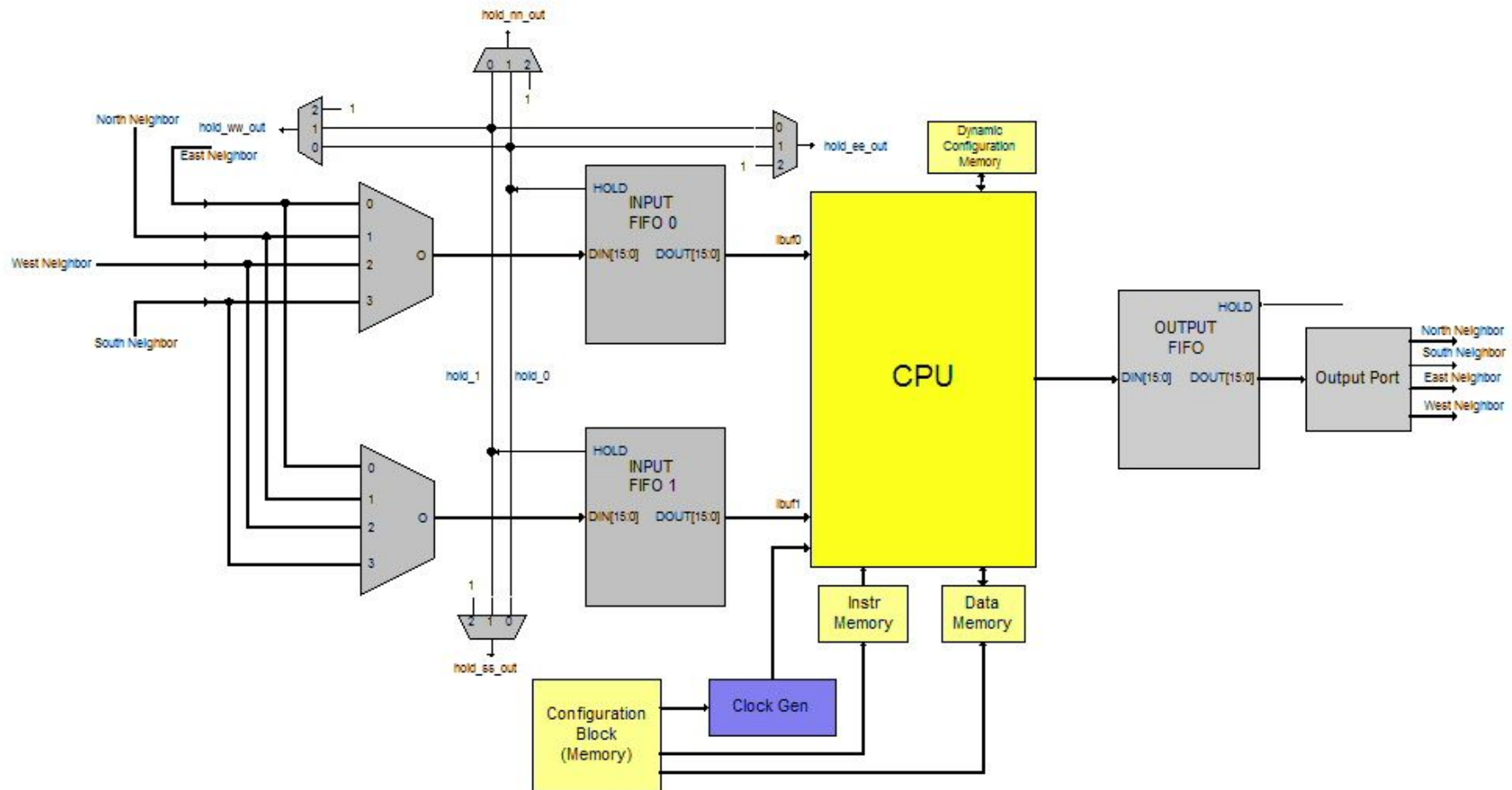
Asynchronous Array of simple Processors



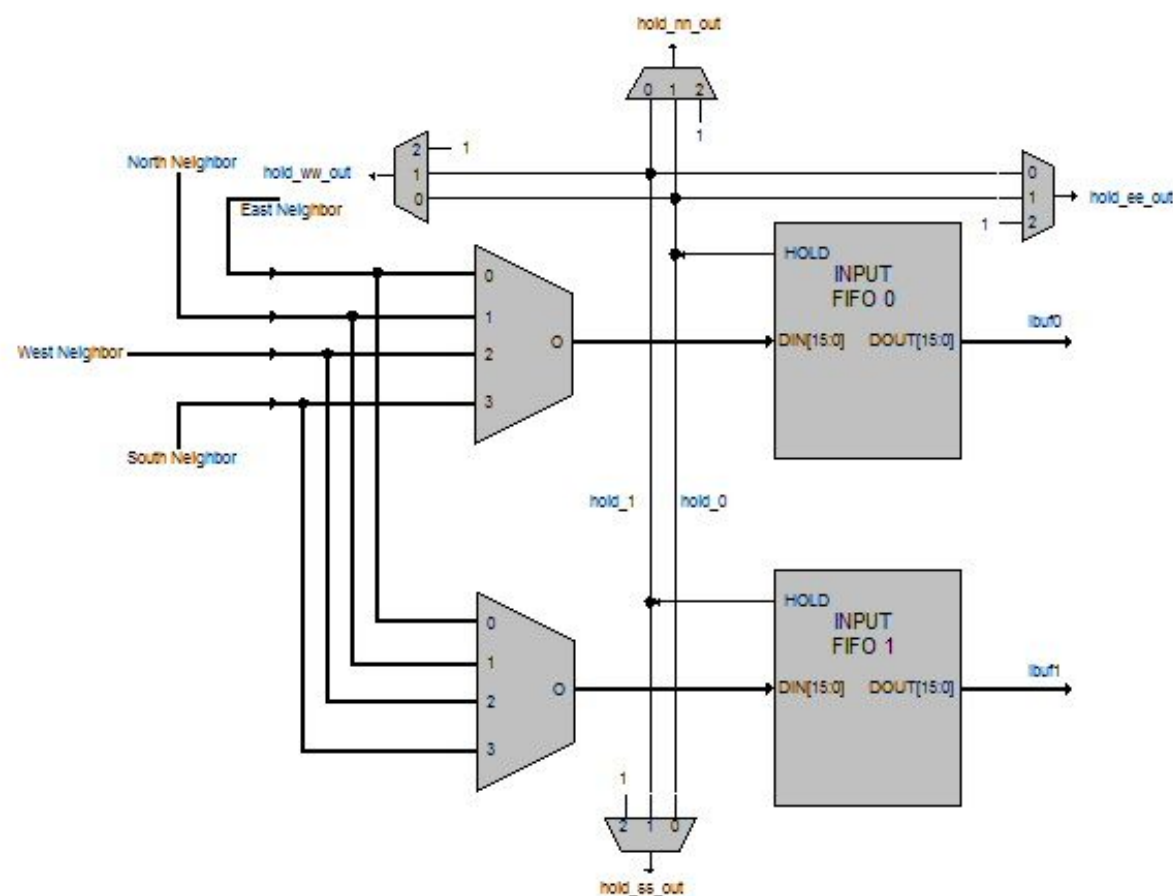
What's in an AsAP tile?

- 16-bit fixed point datapath single issue CPU ^[1]
 - Instructions for AsAP processors are 32-bits wide. ^[2]
- ALU, MAC
- Small Instruction/Data Memories
 - 64-entry instruction memory and a 128-word data memory. ^[2]
- Hardware address generation
 - Each processor has 4 address generators that calculate addresses for data memory. ^[2]
- Local programmable clock oscillator
- 2 Input and 1 Output 16-bits wide and 32-words deep dual-clock FIFOs. ^[2]
- $\sim 1.1 \text{ mm}^2/\text{processor}$ in $0.18 \mu\text{m}$ CMOS
- 800 MHz targeted operation

AsAP Single Processor Tile



AsAP Inter-processor Communication



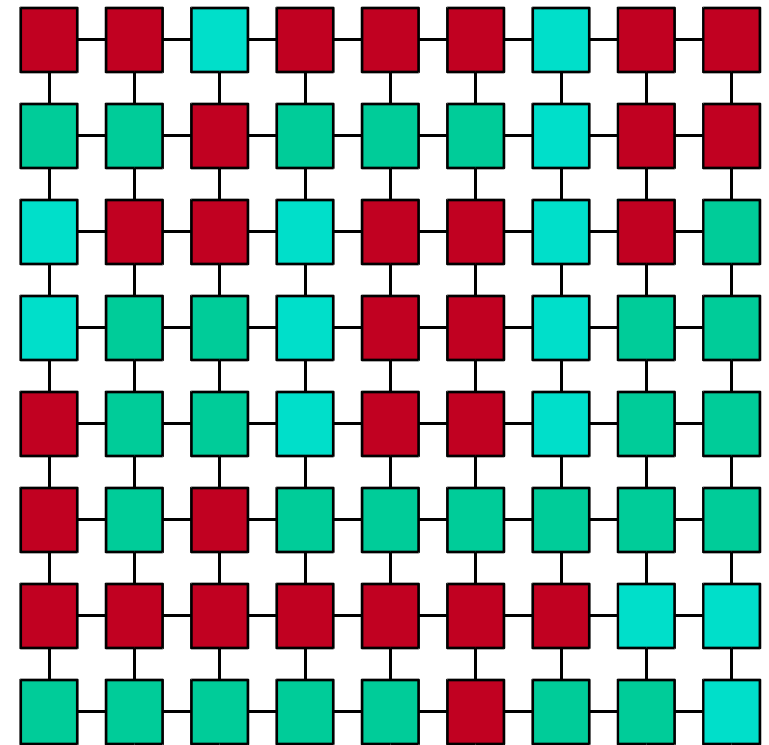
- Each processor output is hard-wired to its four nearest neighbors input multiplexers.
- At power-up the input multiplexers are configured.
- As input FIFOs fill up the sourcing neighbor can be halted by asserting corresponding hold signal.

AsAP Contributions

- Provides parallel execution of independent tasks by providing many, parallel, independent processing engines ^[3]
- AsAP specifies a homogenous 2-D array of very simple processors
 - Single-issue pipelined CPUs
- Independent tasks are mapped across processors and executed in parallel
- Allows efficient exploitation of Application-level parallelism.

AsAP Novel Features

- AsAP
 - Many processing elements
 - High clock rates
 - Possibly many processors inactive
 - Activity localized to increase energy efficiency and performance



Raw vs. AsAP

Parameter	IBM SA-27E (Raw) [4,5,6]	UC Davis AsAP (estimated) [1]
Litho	180 nm	180 nm
Design Style	Std Cell ASIC	Full Custom
Clk Freq (MHz)	425	800
BW per I/O Bus	13.6 Gb/s	12.8 Gb/s
# tiles/chip	64	405
CPU type	8-stage MIPS (32-bit floating point)	9-stage reduced complexity DSP (16-bit fixed point)
Die Area	331 mm ²	~445 mm ²
Tile Area	~5 mm ²	1.1 mm ²

Conclusion

The M.I.T. Raw and UC Davis AsAP processors set out to accomplish similar goals, and to some extent have accomplished them.

While Raw has a smaller number of processors per chip and more memory, AsAP has a larger number of processors per chip with the ability to distribute the memory hogging tasks over multiple processors. This will certainly allow these processors to compete in many of the same markets.

References

- [1] Michael J. Meeuwsen, Omar Sattari, Bevan M. Baas, "A Full-rate Software Implementation of an IEEE 802.11a Compliant Digital Baseband Transmitter," In Proceedings of the *IEEE Workshop on Signal Processing Systems (SIPS '04)*, October 2004.
- [2] Omar Sattari, "Fast Fourier Transforms on a Distributed Digital Signal Processor," *Masters Thesis, Technical Report ECE-CE-2004-7, Computer Engineering Research Laboratory, ECE Department, University of California, Davis, Davis, CA, 2004.*
- [3] Bevan M. Baas, "A Parallel Programmable Energy-Efficient Architecture For Computationally-Intensive DSP Systems," In *Signals, Systems and Computers, 2003. Conference Record of the Thirty-Seventh Asilomar Conference*, November 2003.
- [4] Michael Taylor, "Evaluating the Raw Microprocessor," presented at the *Boston Architecture Research Conference*, January 30, 2004
- [5] Michael Bedford Taylor, "Design Decisions in the Implementation of a Raw Architecture Workstation," *MS Thesis, Massachusetts Institute of Technology, Cambridge, MA, September, 1999.*
- [6] Michael Bedford Taylor, "The Raw Processor Specification (LATEST)," *Comprehensive specification for the Raw processor*, Cambridge, MA, Continuously Updated 2003.
- [7] David Wentzlaff, Michael Bedford Taylor, et al., "The Raw Architecture: Signal Processing on a Scalable Composable Computation Fabric," *High Performance Embedded Computing Workshop*, 2001
- [8] Michael Taylor, "Evaluating The Raw Microprocessor: Scalability and Versatility," *Presented at the International Symposium on Computer Architecture*, June 21, 2004
- [9] M.I.T. raw Architecture Workstation website: <http://cag-www.lcs.mit.edu/raw/purpose/>