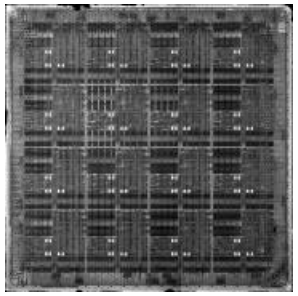
Creating a Scalable Microprocessor:

A 16-issue Multiple-Program-Counter Microprocessor With Point-to-Point Scalar Operand Network

Michael Bedford Taylor

J. Kim, J. Miller, D. Wentzlaff, F. Ghodrat, B. Greenwald, H. Hoffmann, P. Johnson, W. Lee, A. Saraf, N. Shnidman, V. Strumpen, Saman Amarasinghe, Anant Agarwal



Raw Architecture Group
Laboratory for Computer Science
Massachusetts Institute of Technology

Creating a Scalable Microprocessor

Motivation

The Raw Architecture

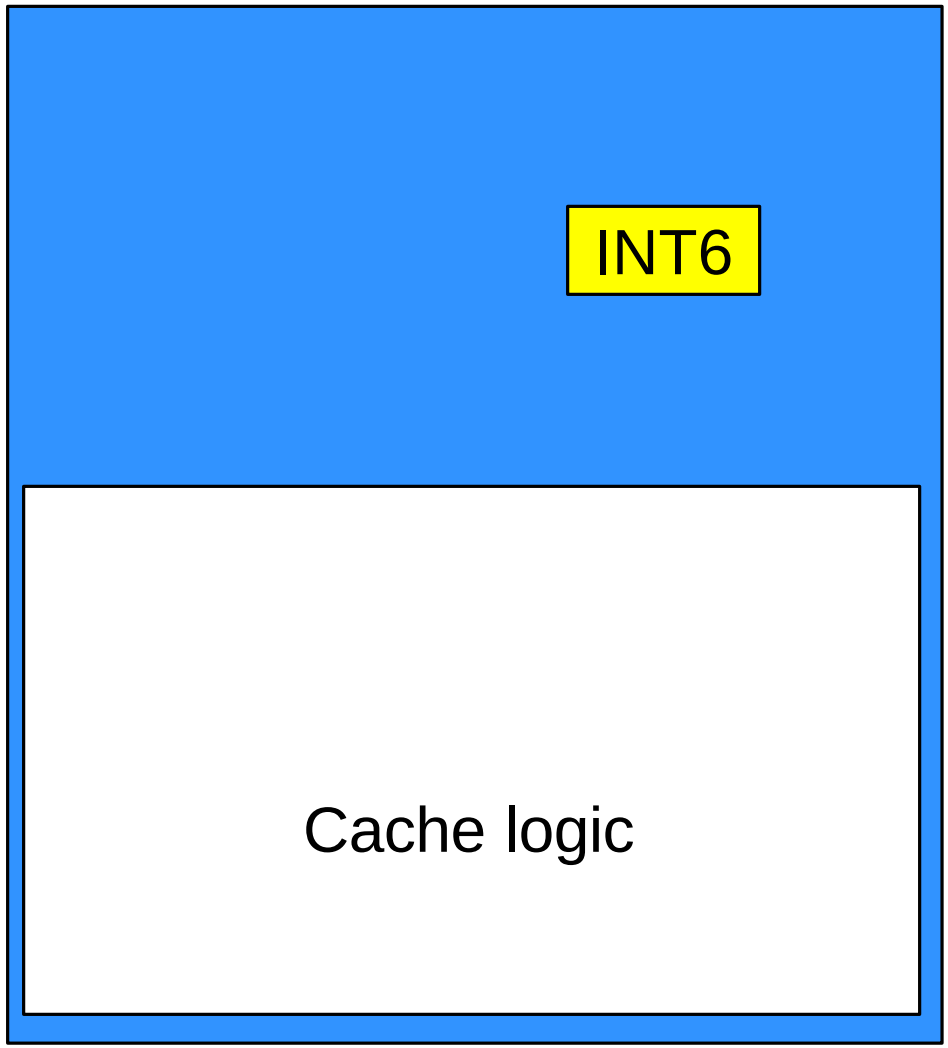
The Raw Prototype

Motivation

As a thought experiment, let's examine the Itanium II, published in last year's ISSCC:

6-way issue Integer Unit
< 2% die area

Cache logic
> 50% die area

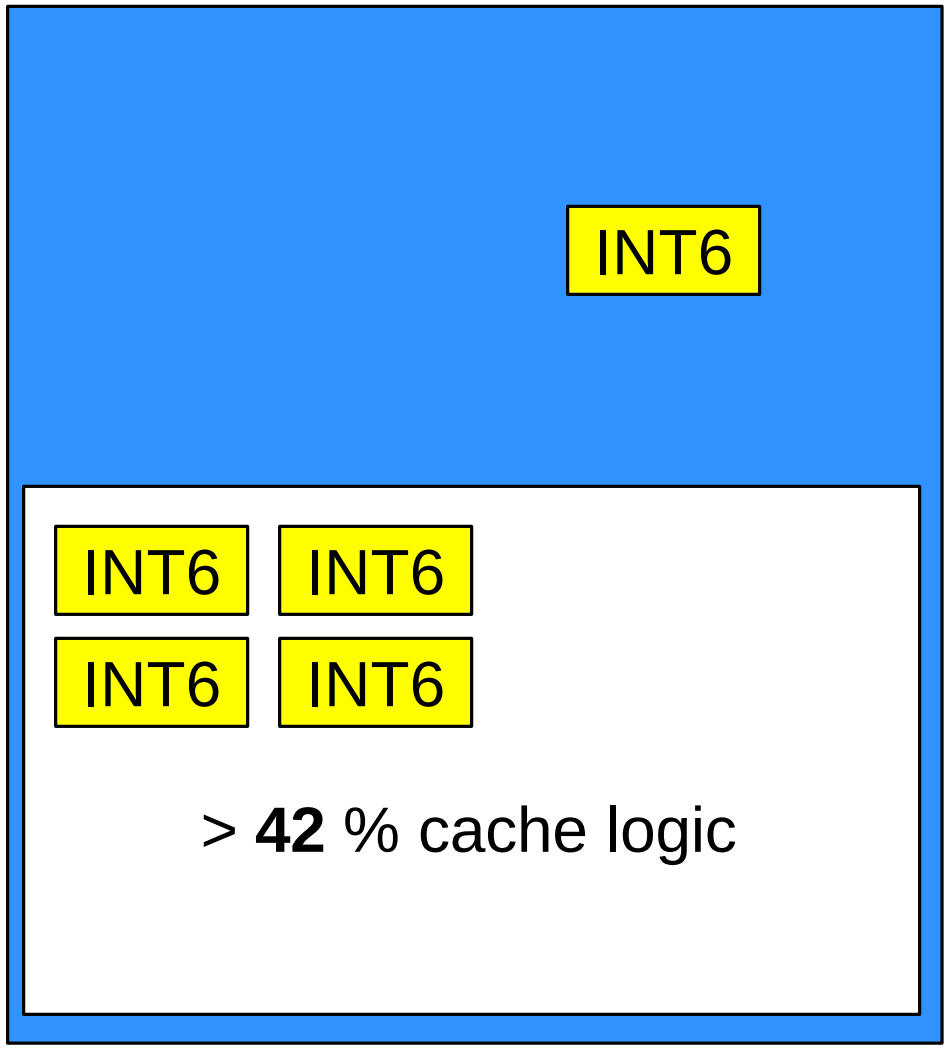


Hypothetical Microprocessor

Why not replace a small portion of the cache with additional issue units?

“30-way” issue micro!

Integer Units still occupy less than 10% area



Scalability Problems Compound

- Fetch unit fetches 30 instructions every cycle
- Decode 30 instructions every cycle

- Check dependencies on 30 instructions every cycle

- Register file needs 60 read ports and 30 write ports
  -- and 100's of physical registers just to keep all of the live values around

- Bypassing: a 30-input 60-output, 32(64)-bit crossbar

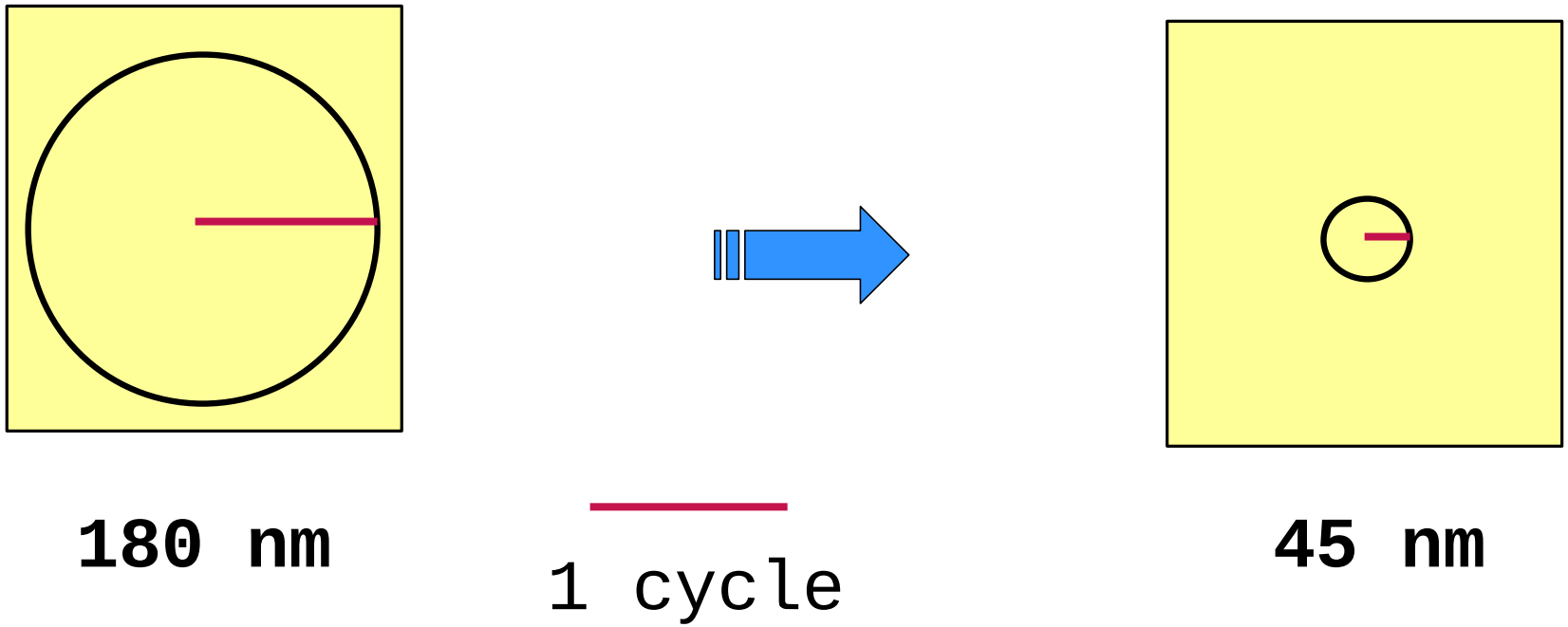
- L1 cache will need many read and write ports
  -- and like the register file, be larger

Can monolithic structures like this be attained at high frequency?

The 6-way integer unit in Itanium II already spends 50% of its critical path in bypassing.
[ISSCC 2002 – 25.6]

Even if dynamic logic or logarithmic circuits could be used to flatten the number of logic levels of these huge structures –

...wire delay is inescapable



Ultimately, wire delay limits the scalability of un-pipelined, high-frequency, centralized structures.

Raw addresses scalability by...

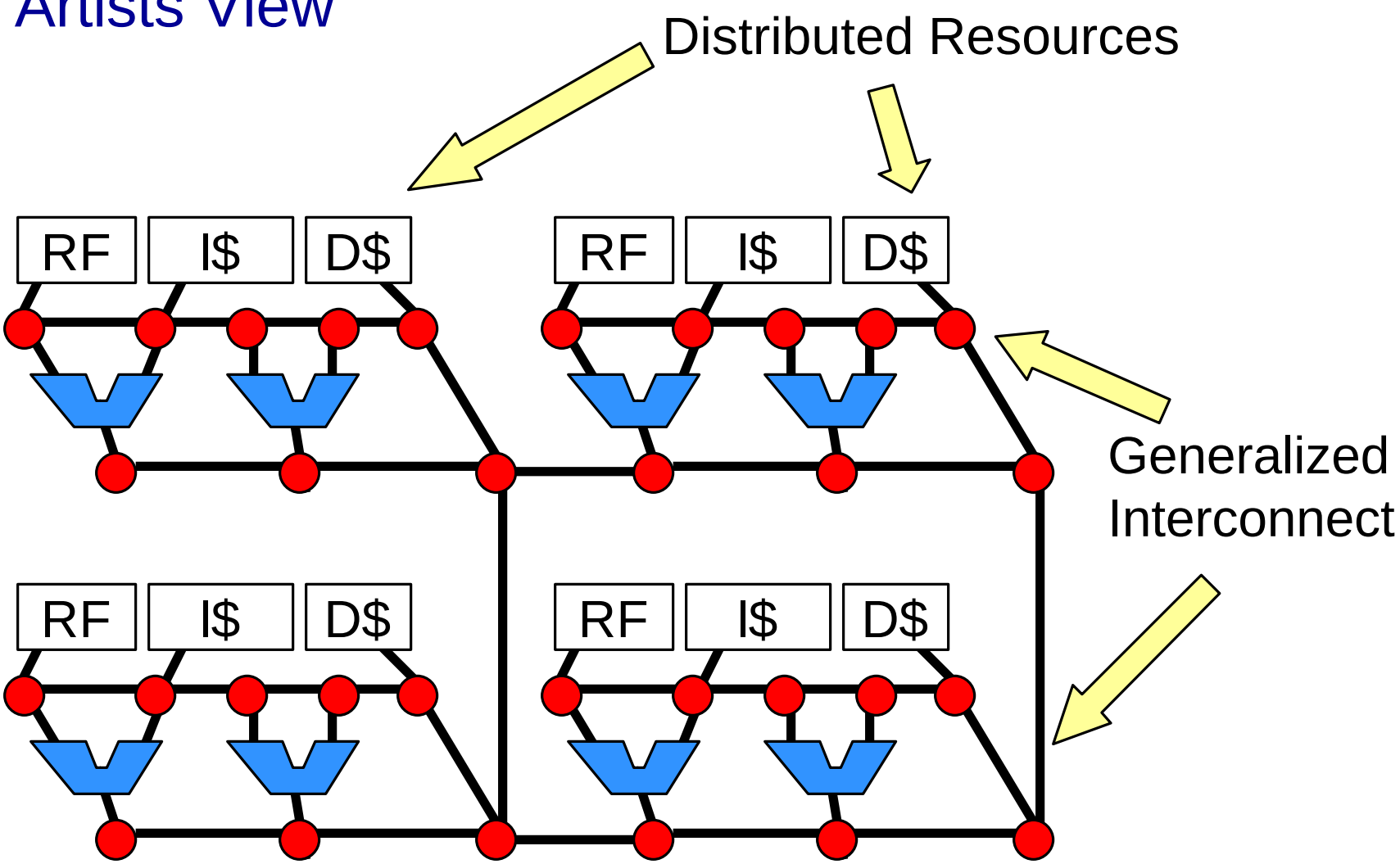
→ Distributing everything over an interconnection network ←

- Fetch Unit
- Decode
- Register File
- L1 Data Cache and Instruction Caches
- Control
- Stall signals
- I/O, memory system, interrupts
- Operand Bypassing

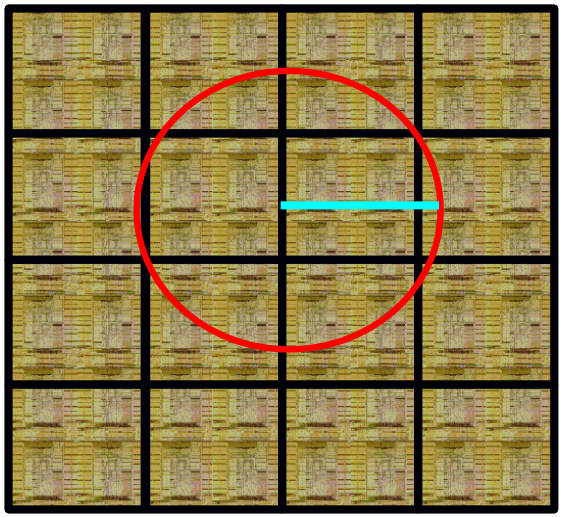
The only centralized resource is a global clock.
(we could get rid of that too)

Idea: distribute everything over a generalized on-chip network

"Artists View"



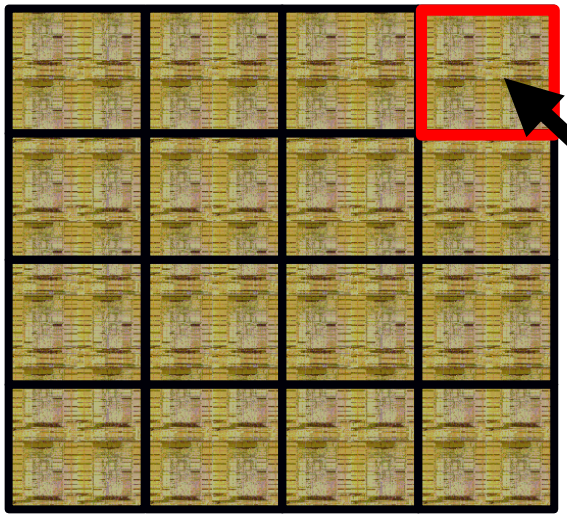
The Raw Architecture



Divide the silicon into an array of 16 identical, programmable tiles.

(A signal can get through a small amount of logic and to the next tile in one cycle.)

The Raw Tile



Tile

8 stage 32b MIPS-style single-issue in-order compute processor

32 KB ICache

32 KB DCache

4-stage 32b pipelined FPU

Routers and wires for three on-chip mesh networks

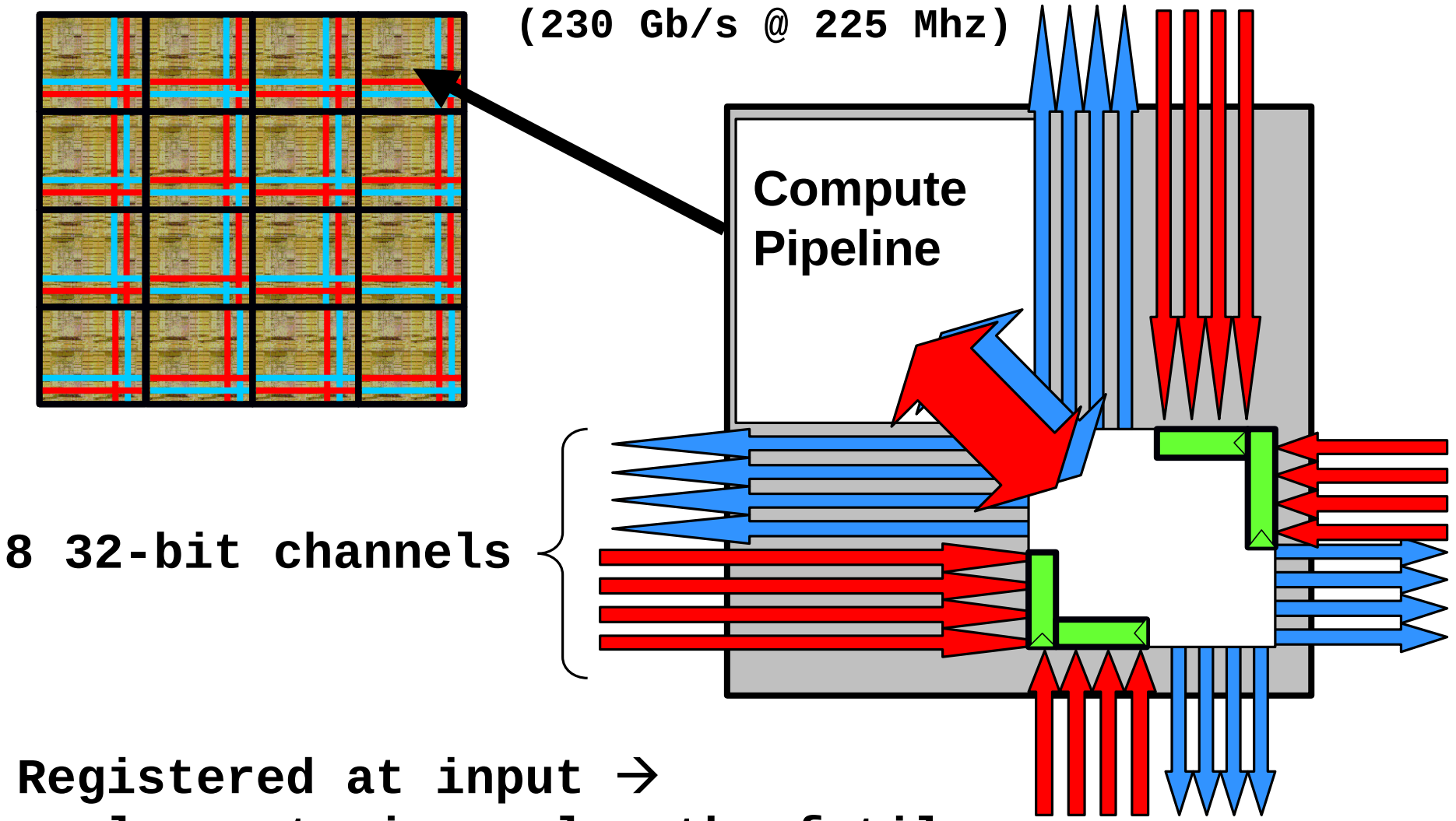
Tiling has gotten us this far...

✓ Fetch Units	16 instructions / cycle	aggregate totals for a 16 tile processor
✓ Decoders	16 instructions / cycle	
✓ Register Files	32 read / 16 write ports 512 registers	
✓ L1 I/D Caches	32 read / write ports 1 MB	

- Inter-tile Bypassing } Why Raw is not just a CMP.
- Control
- Stall signals
- I/O, memory system, interrupts

} Using the on-chip networks.

Raw's three on-chip mesh networks



Registered at input →
longest wire = length of tile

Raw's three on-chip networks

"Scalar Operand Network"
For communication of operands among local and remote ALUs

"Memory Network"
dynamic, dim. order wormhole routed
cache misses, I/O, DMA, OS

"General Network"
dynamic, dim. order wormhole routed
user-level, message passing programs

Raw's Scalar Operand Network

Consists of two tightly-coupled sub-networks:

- Tile interconnection network

  For communication of operands between tiles

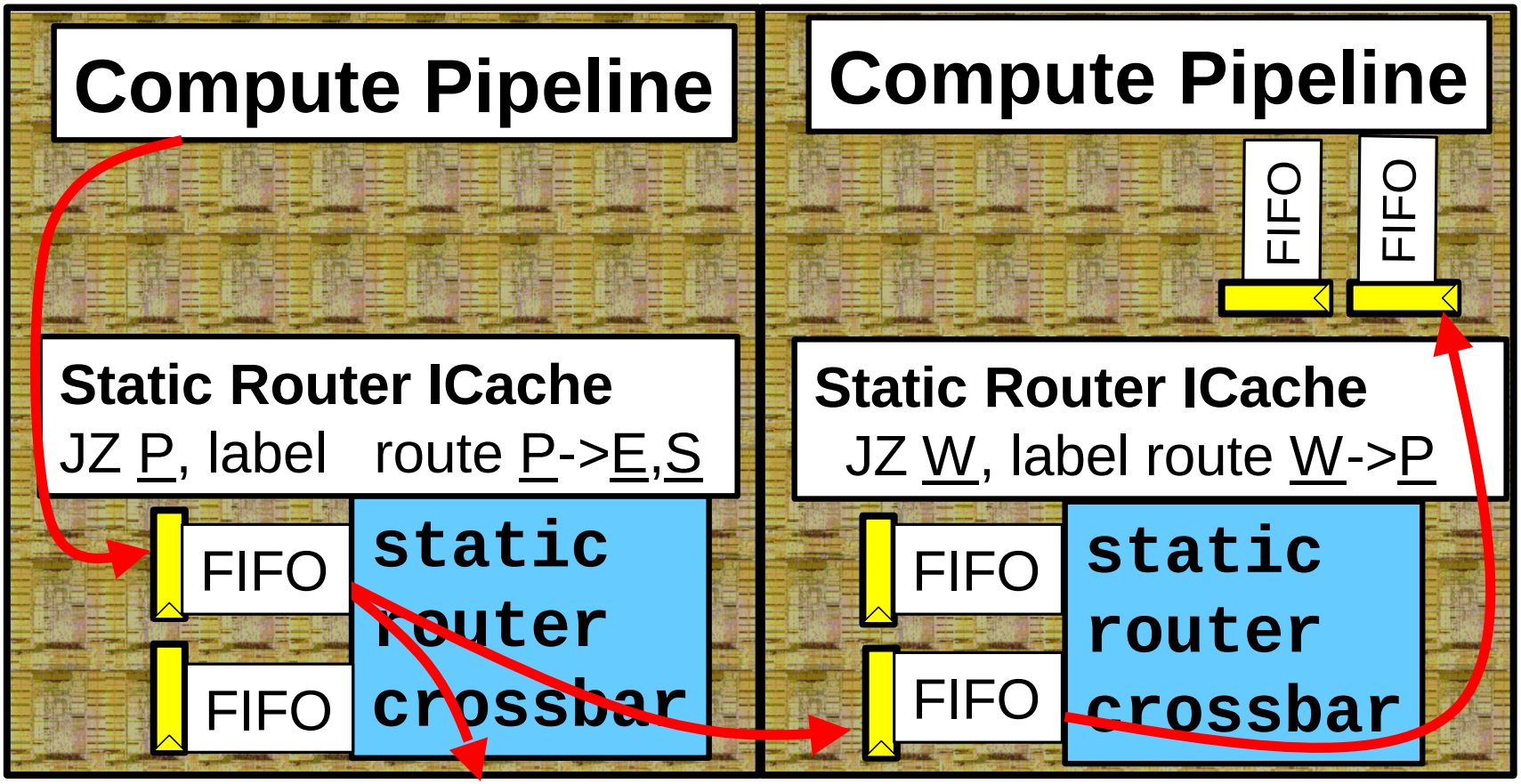
  Controlled by the 16 tiles' static router processors

- Local bypass network

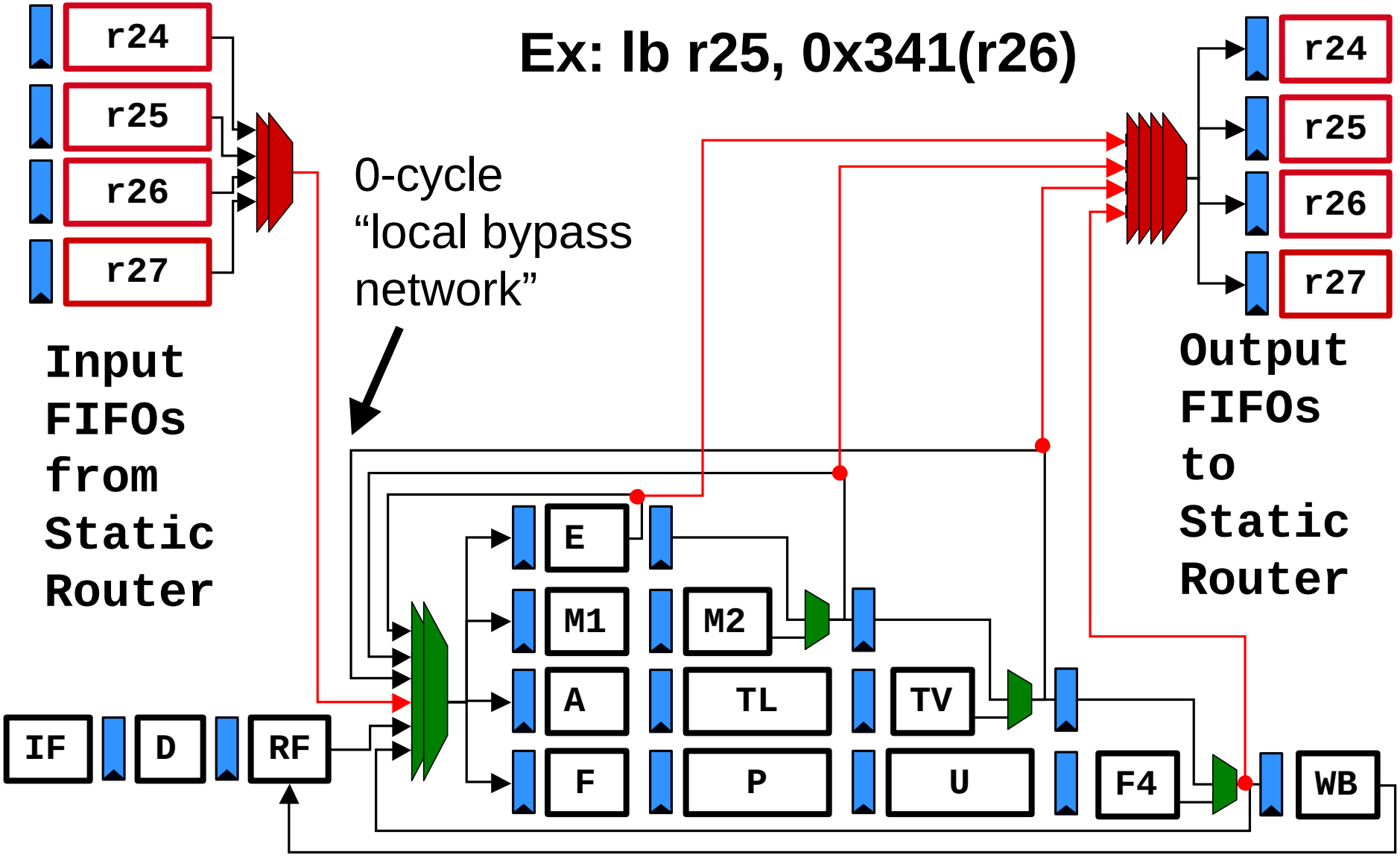
  For communication of operands within a tile

Between tile operand transport

The routes programmed into the static router ICache guarantee in-order delivery of operands between tiles at a rate of 2 words/direction/cycle.



Operand Transport among functional units and the static router



Raw also distributes:

- ✓ Inter-tile Bypassing

  3 cycles nearest neighbor
  1 cycle per hop

- ✓ Control

  Branch conditions are transmitted over scalar operand network. Each affected tile and static router executes a branch instruction. Tiles can have vastly different control flow.

- ✓ Stall signals

  Inter-tile stall conditions are conveyed only by stalls on input and output FIFOs, usually reflecting an operand dependence between the two tiles.

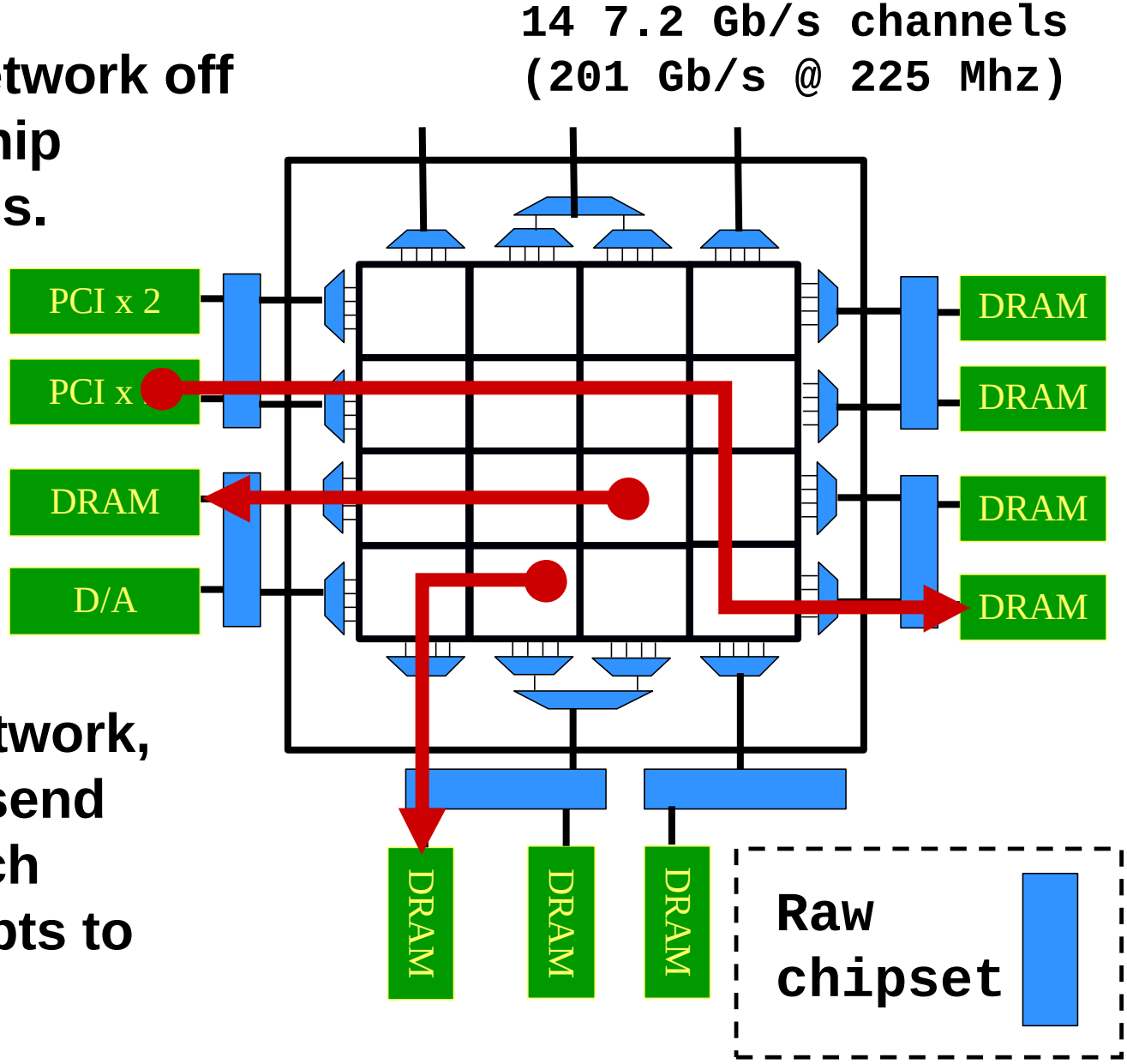
- I/O, memory system, interrupts

Raw's I/O and Memory System

Routes on any network off the edge of the chip appear on the pins.

Tiles cache-miss independently to the DRAM that owns a given cache line.

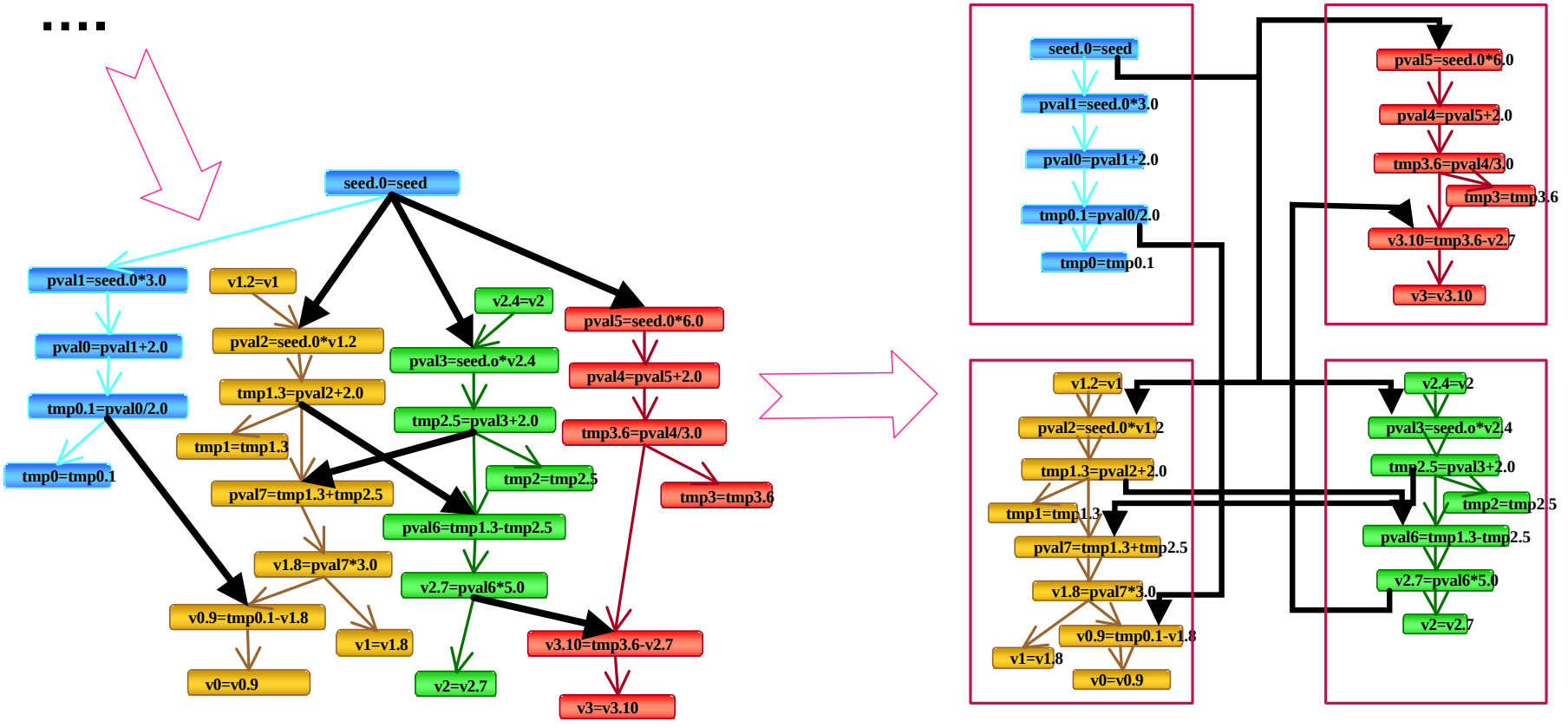
Using on-chip network, off-chip devices send data (DMA) to each other and interrupts to one or more tiles.

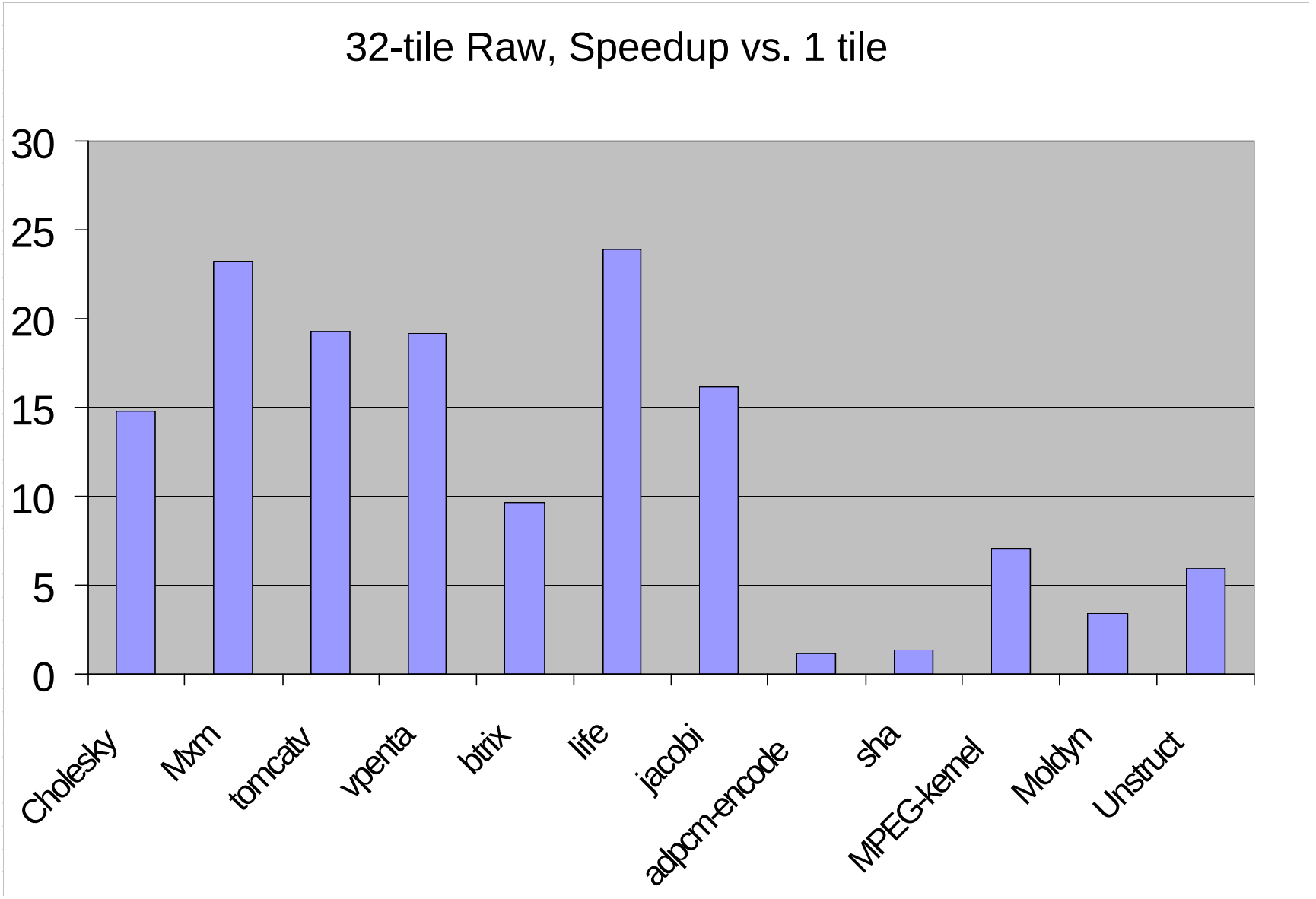


Compilation

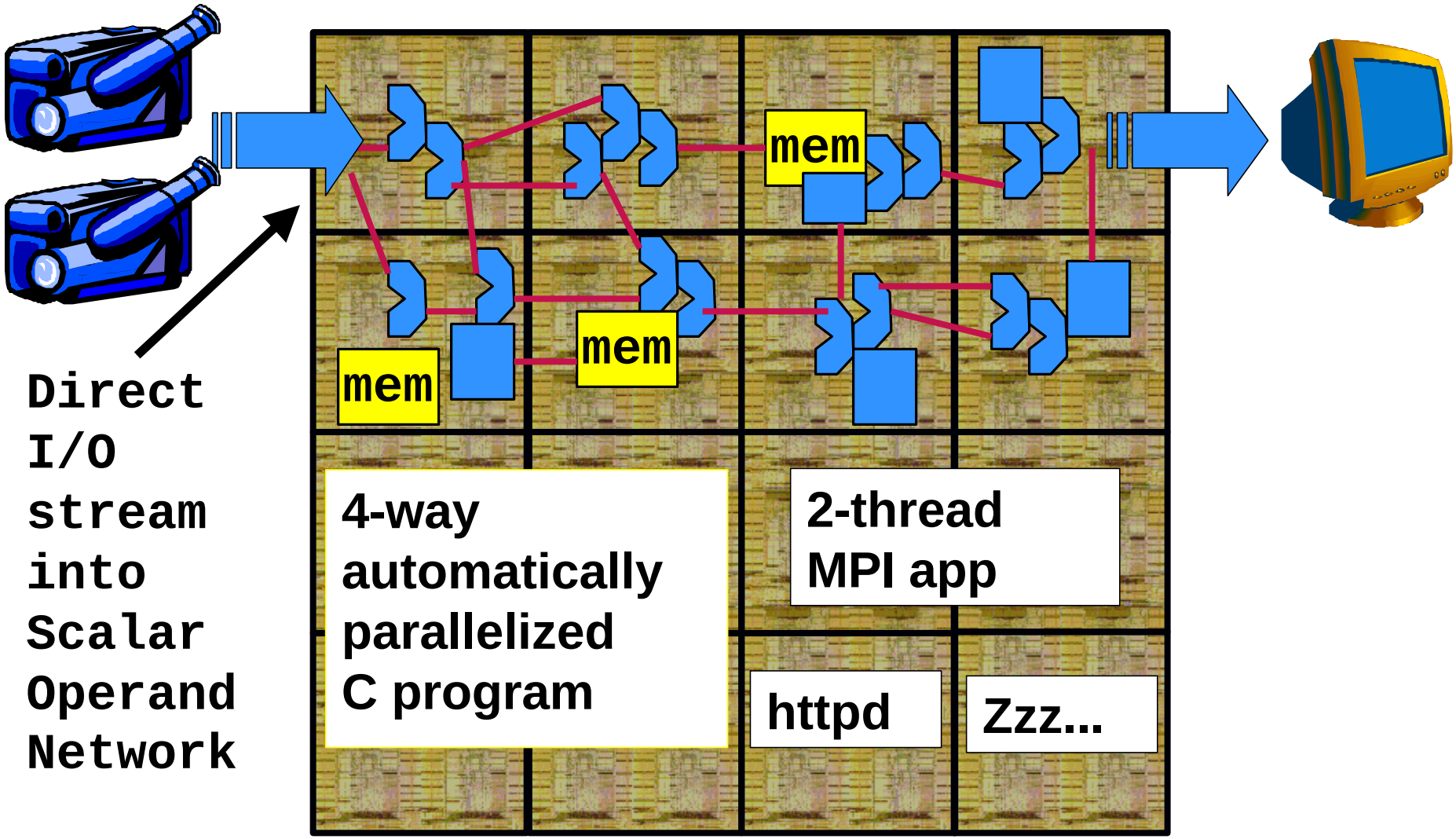
tmp3 = (seed*6+2)/3
v2 = (tmp1 - tmp3)*5
v1 = (tmp1 + tmp2)*3
v0 = tmp0 - v1
….

The Raw compiler assigns instructions to the tiles, maximizing locality. It also generates the static router instructions that transfer operands between tiles.



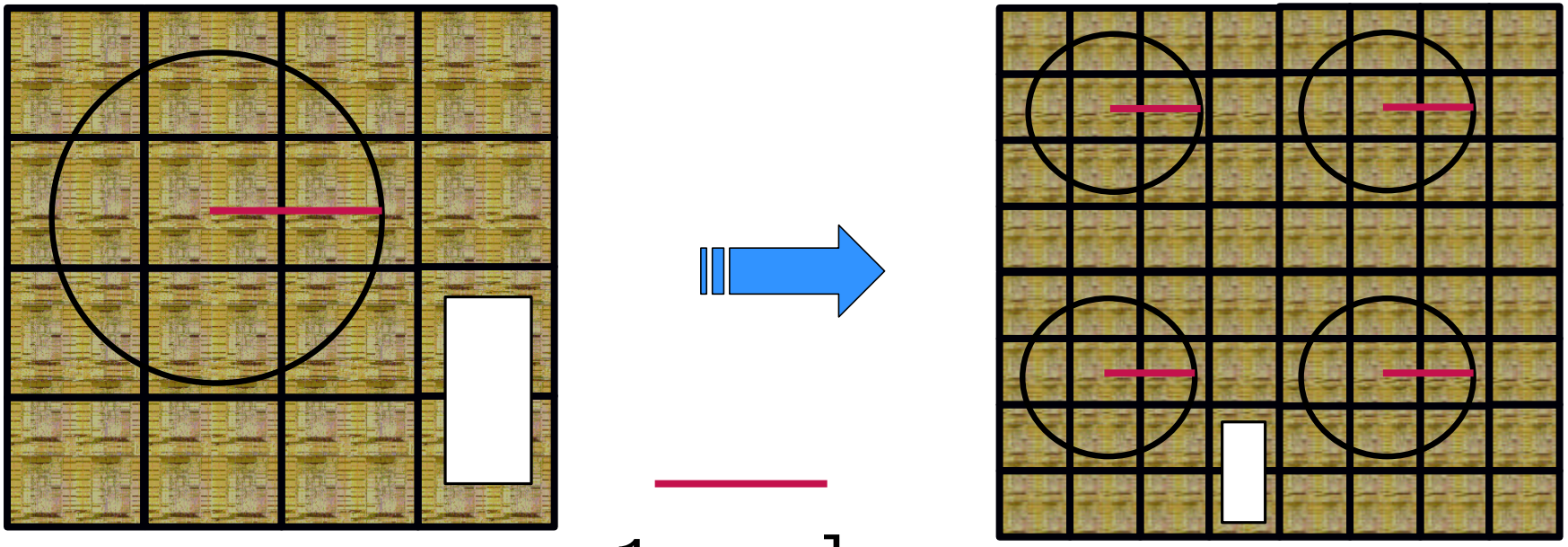

32-tile Raw, Speedup vs. 1 tile
30
25
20
15
10
5
0
Cholesky
Mxm
tomcatv
vpenta
btrix
life
jacobi
adpcm-encode
sha
MPEG-kernel
Moldyn
Unstruct

A Raw System in Action



Note that an application uses only as many tiles as needed to exploit the parallelism intrinsic to that application.

Scalability to Larger Issue Width



180 nm
16-issue

90 nm
64-issue

Just stamp out more tiles!

Longest wire, frequency, design and verification complexity all independent of issue width.

Architecture is backwards compatible.

The Raw Prototype

16 tile processor in an ASIC process

- Intended as an experimental prototype for future processor designs

- Imagine a version implemented by a full-custom design team

The architecture is targeted for systems from 16-issue to 1024-issue – this prototype is at the beginning of this range. We're just coming to VLSI processes where this architecture starts to make sense.

Raw ASIC

IBM SA-27E .15u 6L Cu

18.2 mm x 18.2 mm

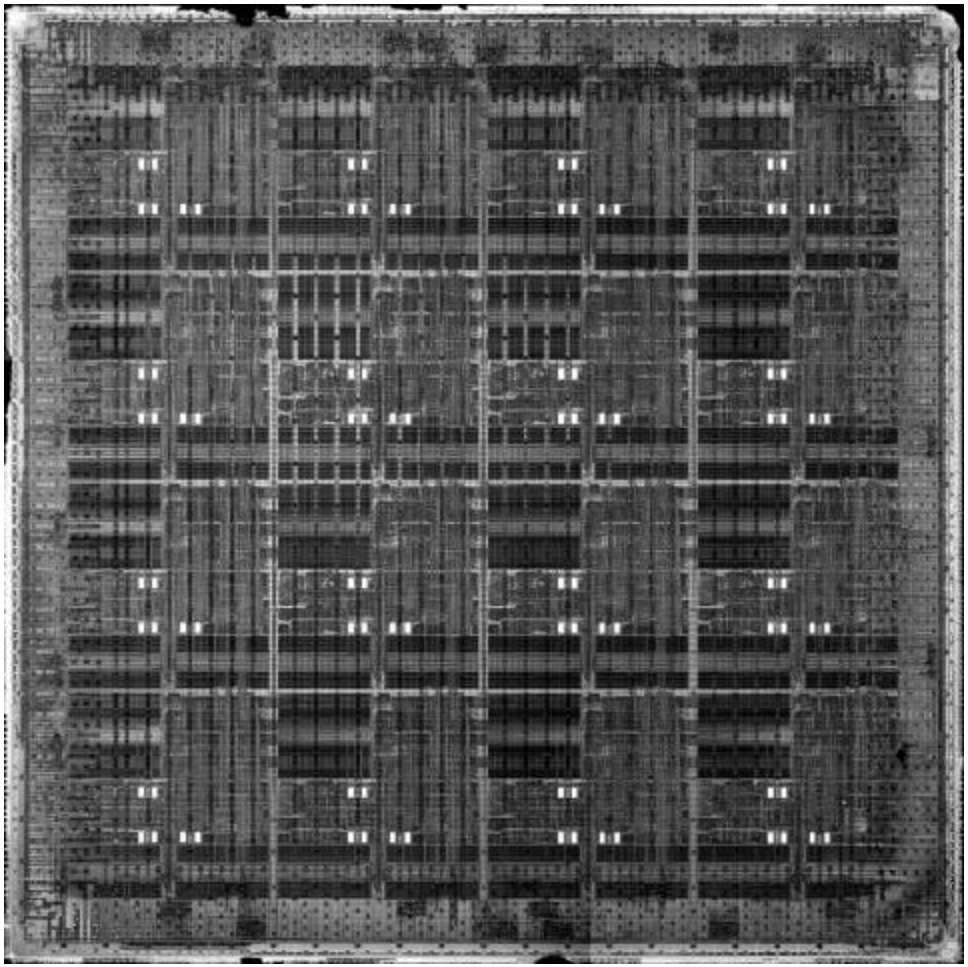
16 Flops/ops per cycle

208 Operand Routes / cycle

2048 KB L1 SRAM

1657 Pin CCGA Package

1080 HSTL core-speed signal I/O

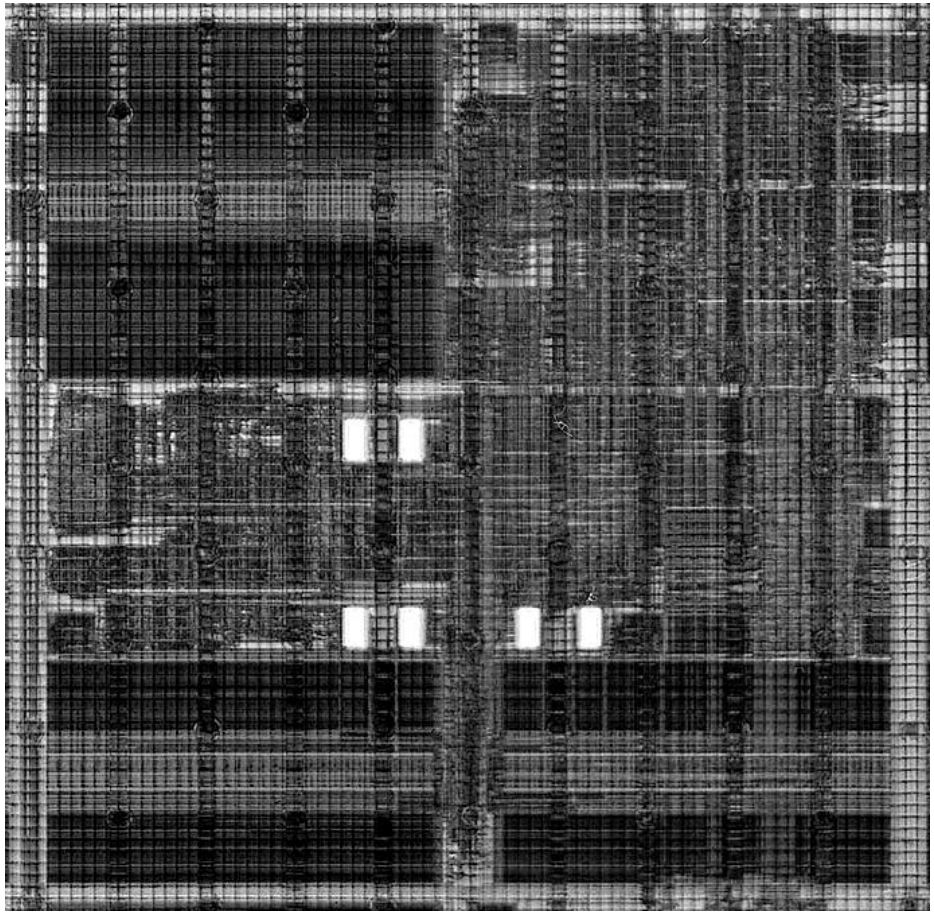


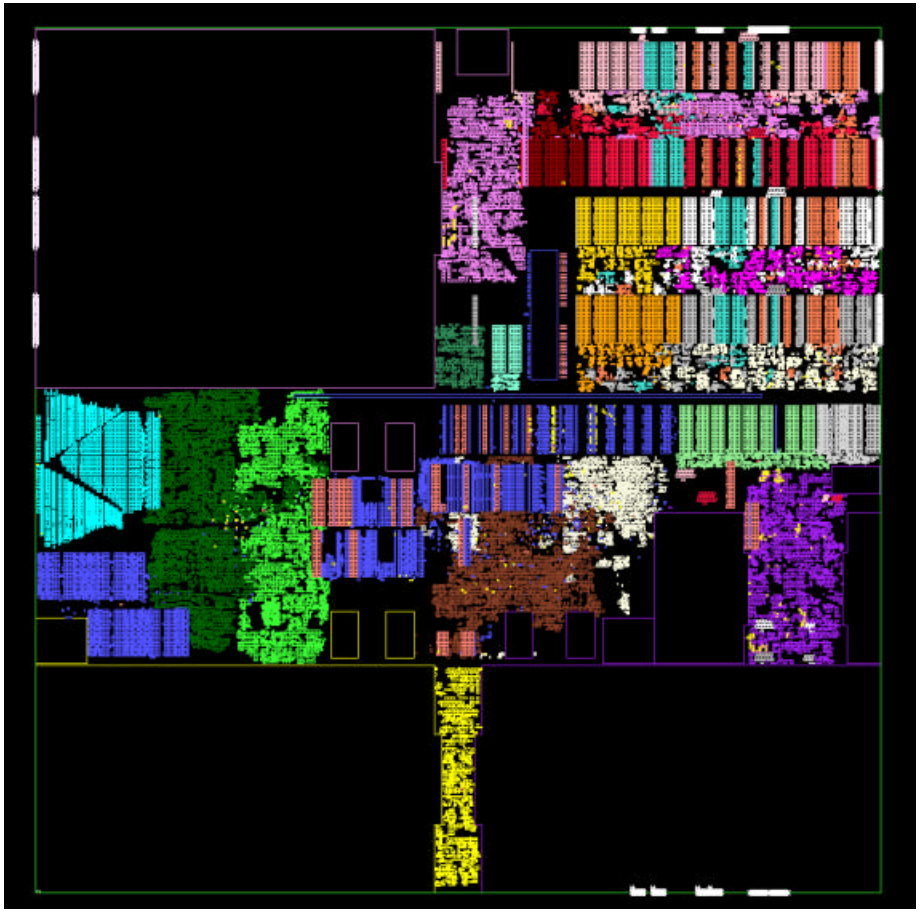
@ 225 MHz Worst Case (Temp, Vdd, process):

3.6 Peak GFLOPS (without FMAC)
230 Gb/s on-chip bisection bandwidth
201 Gb/s off-chip I/O bandwidth

Close-up of a single Raw tile

(design one and the rest will follow!)



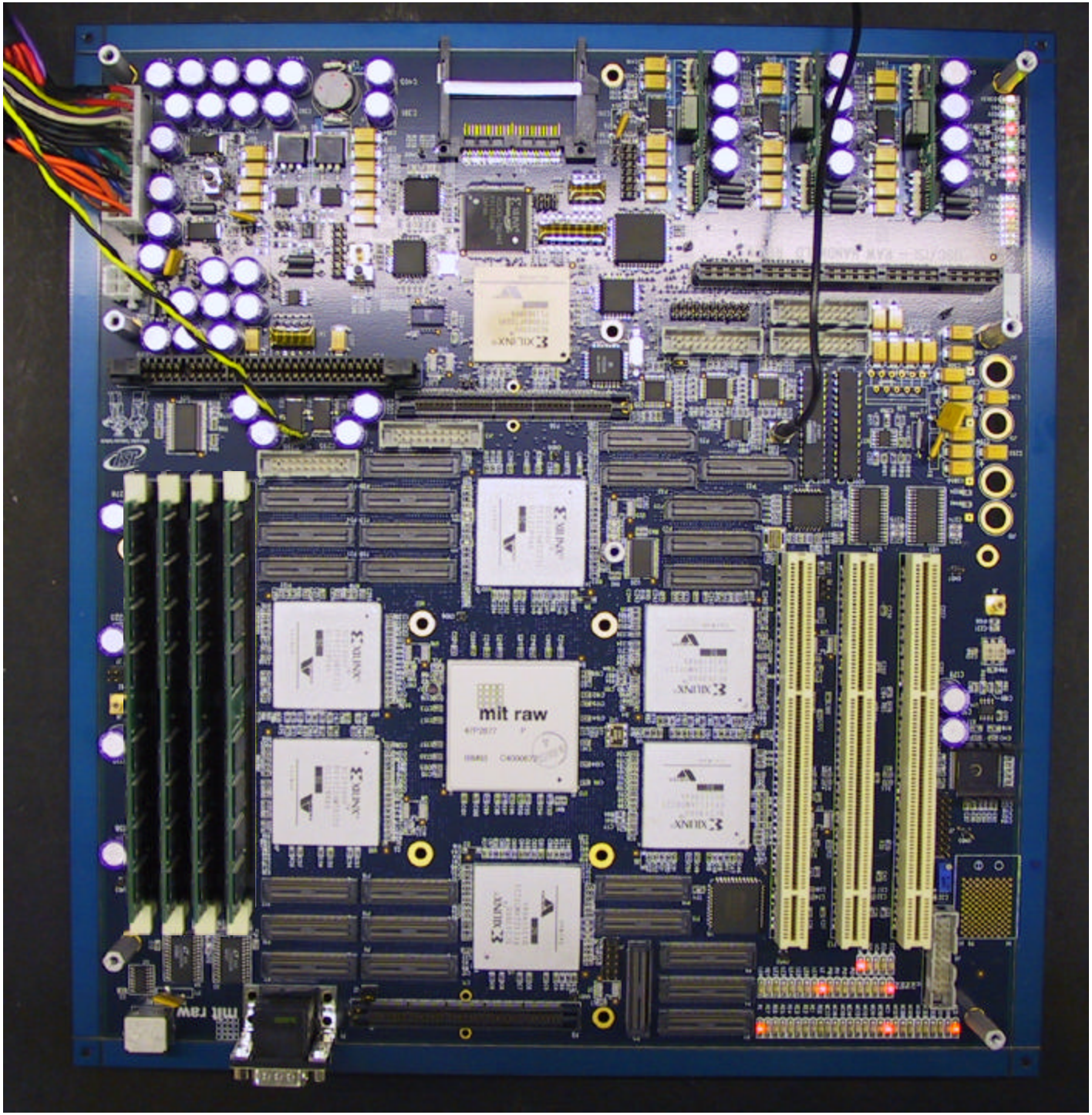


Raw Motherboard

A lot like a PC motherboard.

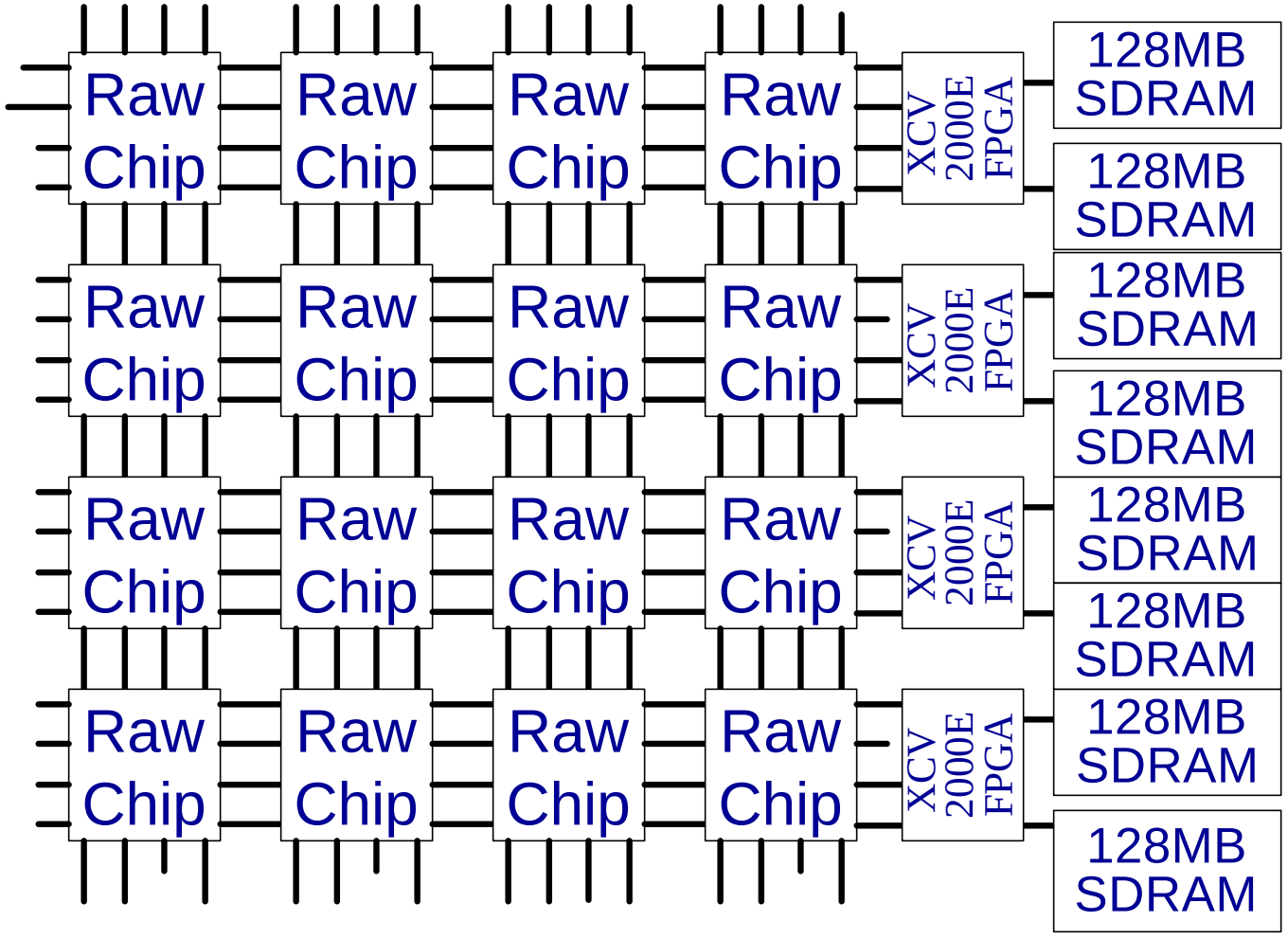
64-bit PCI slots, DIMM slots, PS/2 keyboard, ATX power, and...

.. twenty-eight 32-bit Buses @ 3 225 MHz Connecting I/O Devices and Raw Chip.



Raw chips gluelessly connect to form larger virtual chips up to 32x32 tiles.

This 16 chip array would approximate a 256 tile Raw from a 45 nm process.



Summary

Centralized structures are one of the key impediments to microprocessor scalability.

The Raw architecture scales because it distributes everything over interconnection networks.

- Raw uses a routed, point-to-point scalar operand network to transport operands among functional units with very low latency.

We've designed and built a complete 16-issue prototype system, including a compiler, a chip, and a motherboard.