The Raw Processor

A Scalable 32 bit Fabric for General Purpose and Embedded Computing

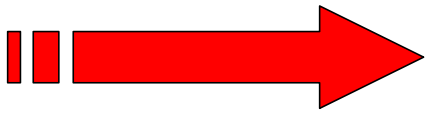
Michael Taylor, Jason Kim, Jason Miller,
Fae Ghodrat, Ben Greenwald, Paul Johnson,Walter Lee,
Albert Ma, Nathan Shnidman, Volker Strumpen, David Wentzlaff,
Matt Frank, Saman Amarasinghe, and Anant Agarwal

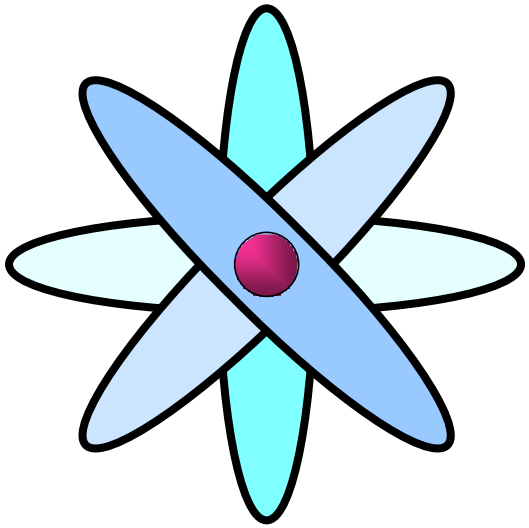
MIT Laboratory for Computer Science

http://cag.lcs.mit.edu/raw

Computer Architecture from 10,000 feet

```
foo(int x)
{ .. }
```





class of computation

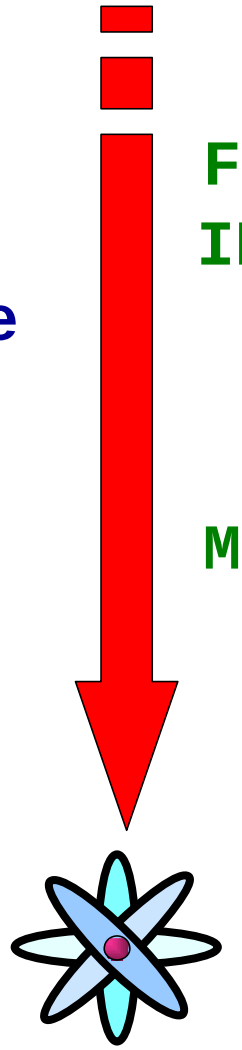
convenient physical phenomenon

… we use abstractions to make this easier

The Abstraction Layers That Make This Easier

```
foo(int x) { .. }
```

Computation
Language / API
Compiler / OS
ISA
Micro Architecture
Layout
Design Style
Design Rules
Process
Materials Science
Physics

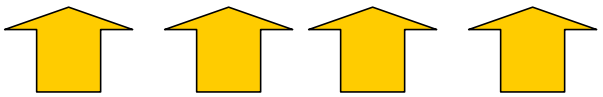


Fortran
IBM 360 /RISC/ Transmeta/x

Mead & Conway

Abstractions protect us from change -- but must also change as the world changes

Language / API
Compiler / OS
ISA
Micro Architecture
Layout
Design Style
Design Rules
Process
Materials Science



Changes in physical constraints

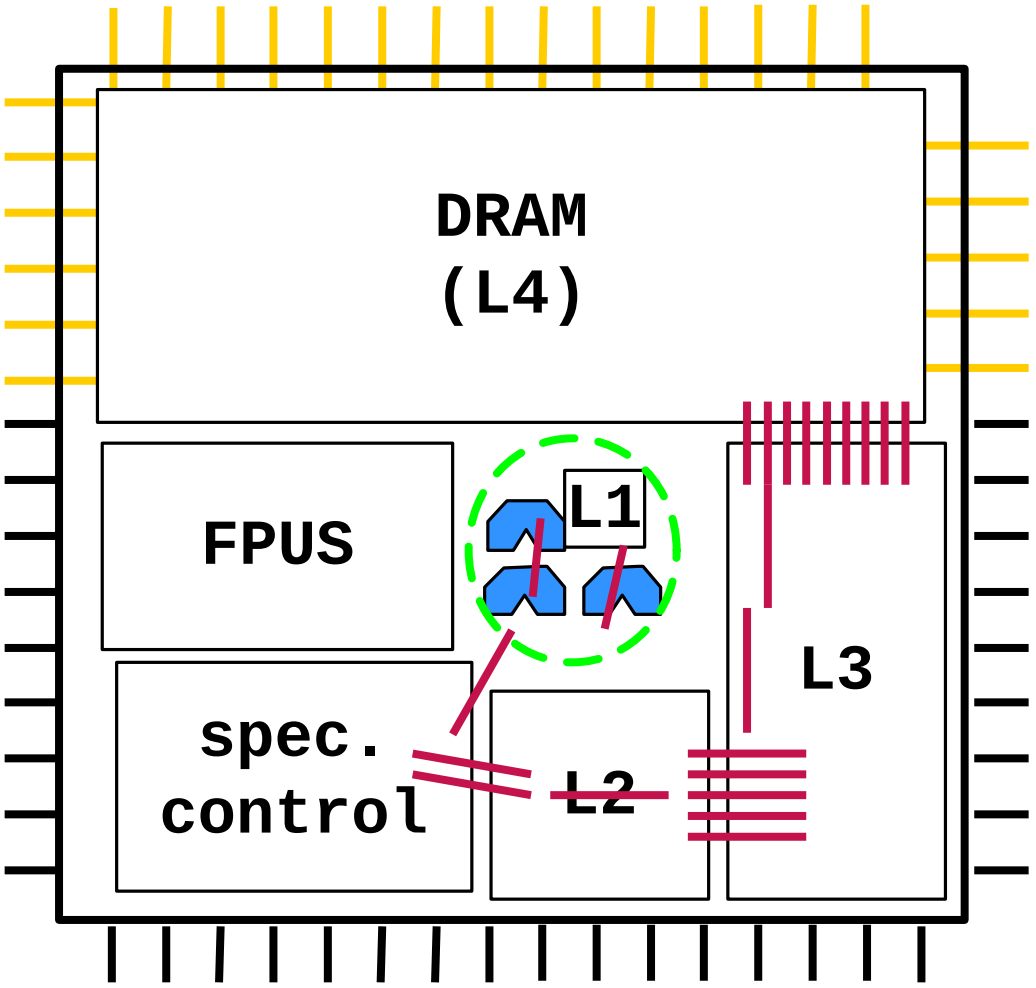
Wire Delay

More Resources:
Gates
Wires
Pins

Wire delay is crashing through the abstraction layers

Language / API	
Compiler / OS	
ISA	
Micro Architecture	**Partitioning(21264)**
	Pipelining (P4)
Layout	**Timing Driving Placement**
Design Rules	**Fatter wires**
Process	**Deeper wires**
Materials Science	**Cu wires**

The future of wire delay handled in micro-architecture



execution core

1 cycle wire

The bottom line

Language / API
Compiler / OS
ISA

Micro Architecture

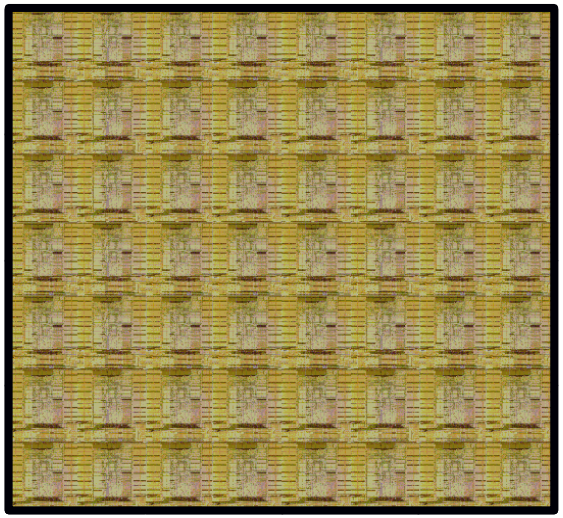
Floorplan / Layout
Design Rules
Process
Materials Science

Raw handles this change by exposing the underlying resources (e.g. wires) with a scalable, parallel ISA.

It orchestrates these resources with spatially-aware compilers.

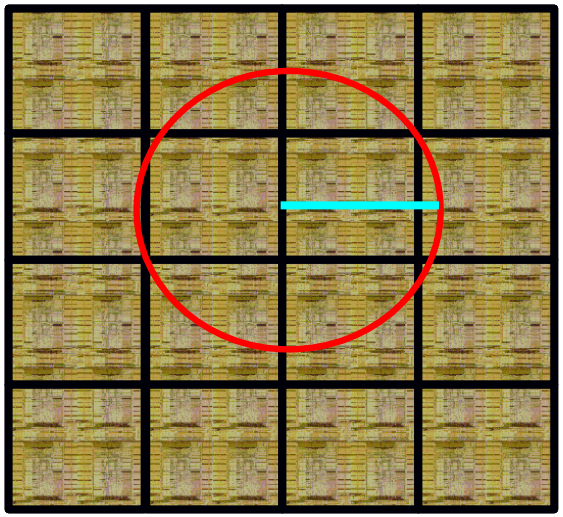
	More Resources:
Wire	Gates
Delay	Wires
	Pins

How does Raw expose the resources?



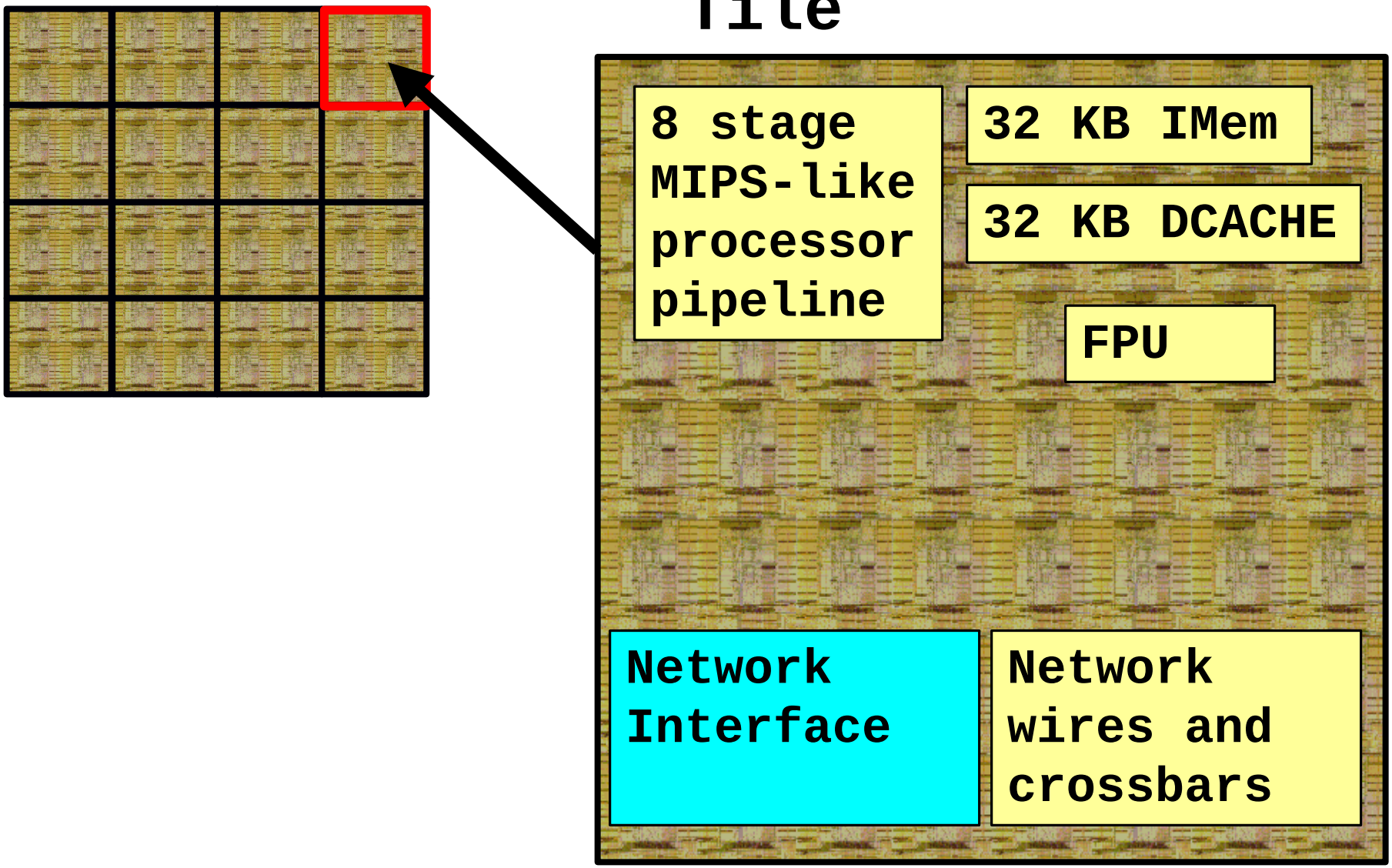
We started with a blank sheet of silicon.

Expose the gates

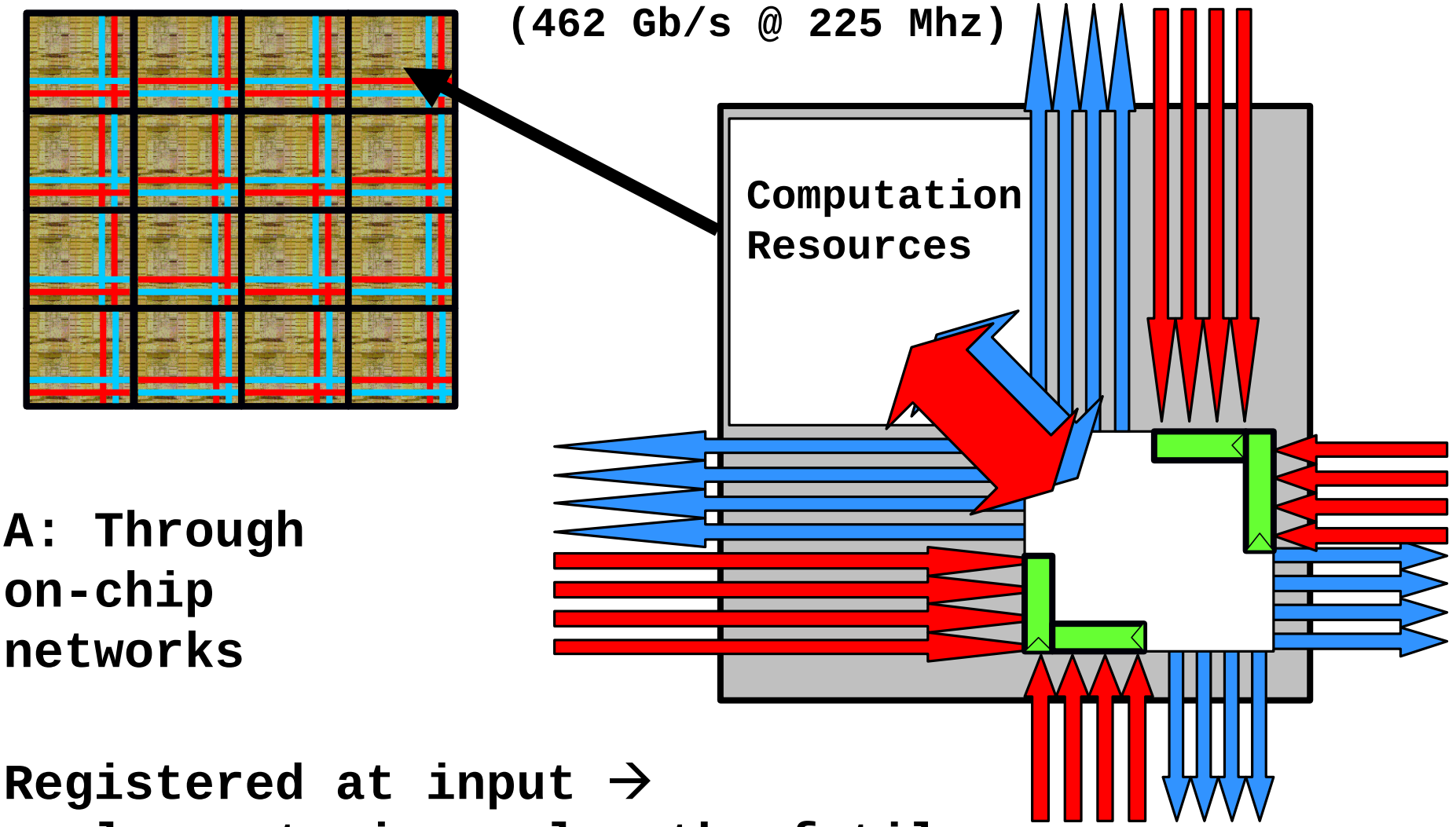


Cut the silicon up into an array of 16 identical, programmable tiles.

What's inside a tile?



How do we expose the wires?



A: Through on-chip networks

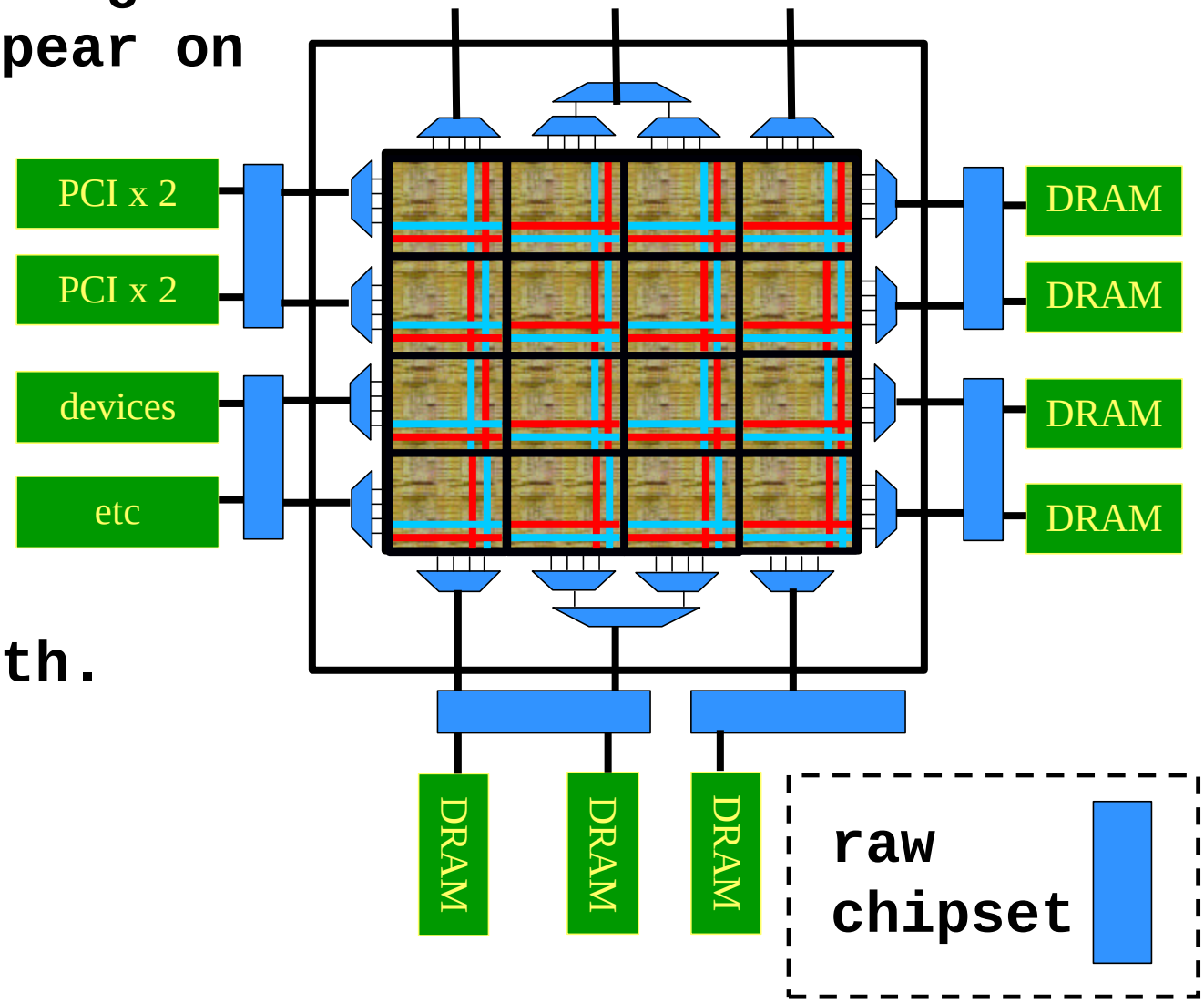
Registered at input →
longest wire = length of tile

How do we expose the pins?

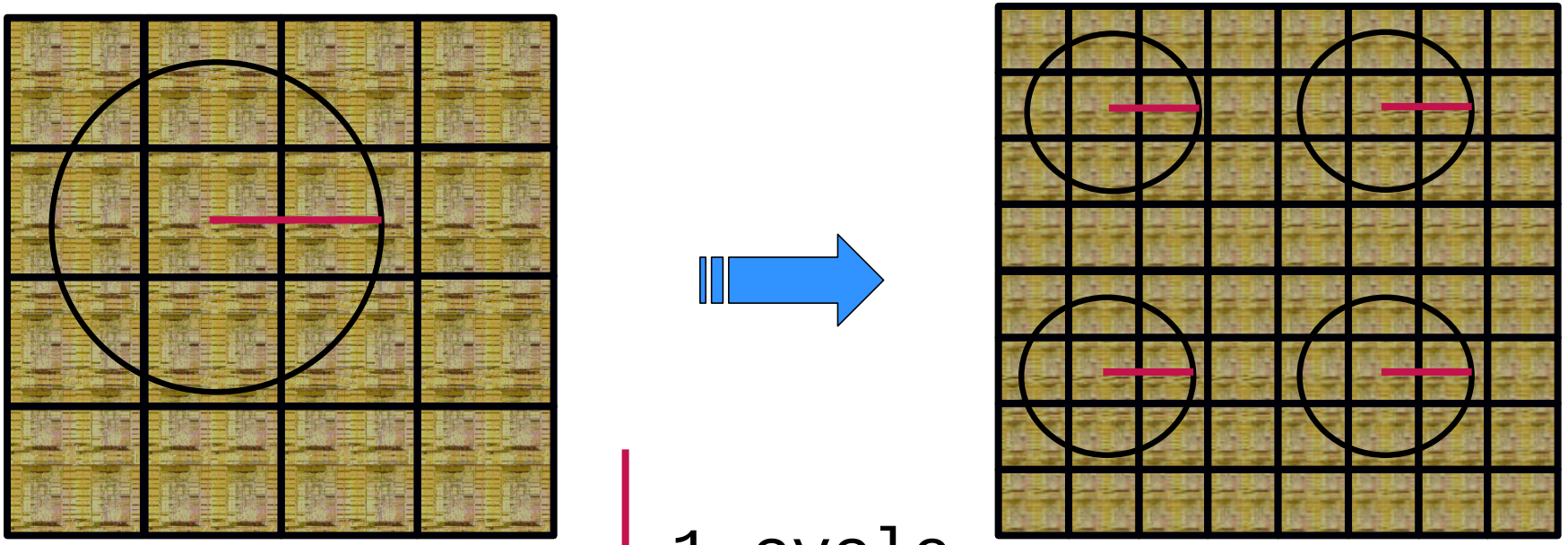
14 7.2 Gb/s channels
(201 Gb/s @ 225 Mhz)

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

Gives user direct access to pin bandwidth.



The Raw ISA scales



.15u

.07u

1. longest wire
2. Design complexity
3. Verification complexity

… are all independent of transistor count.

tiles, network bandwidth and I/O bandwidth scale

Raw is also backwards-compatible.

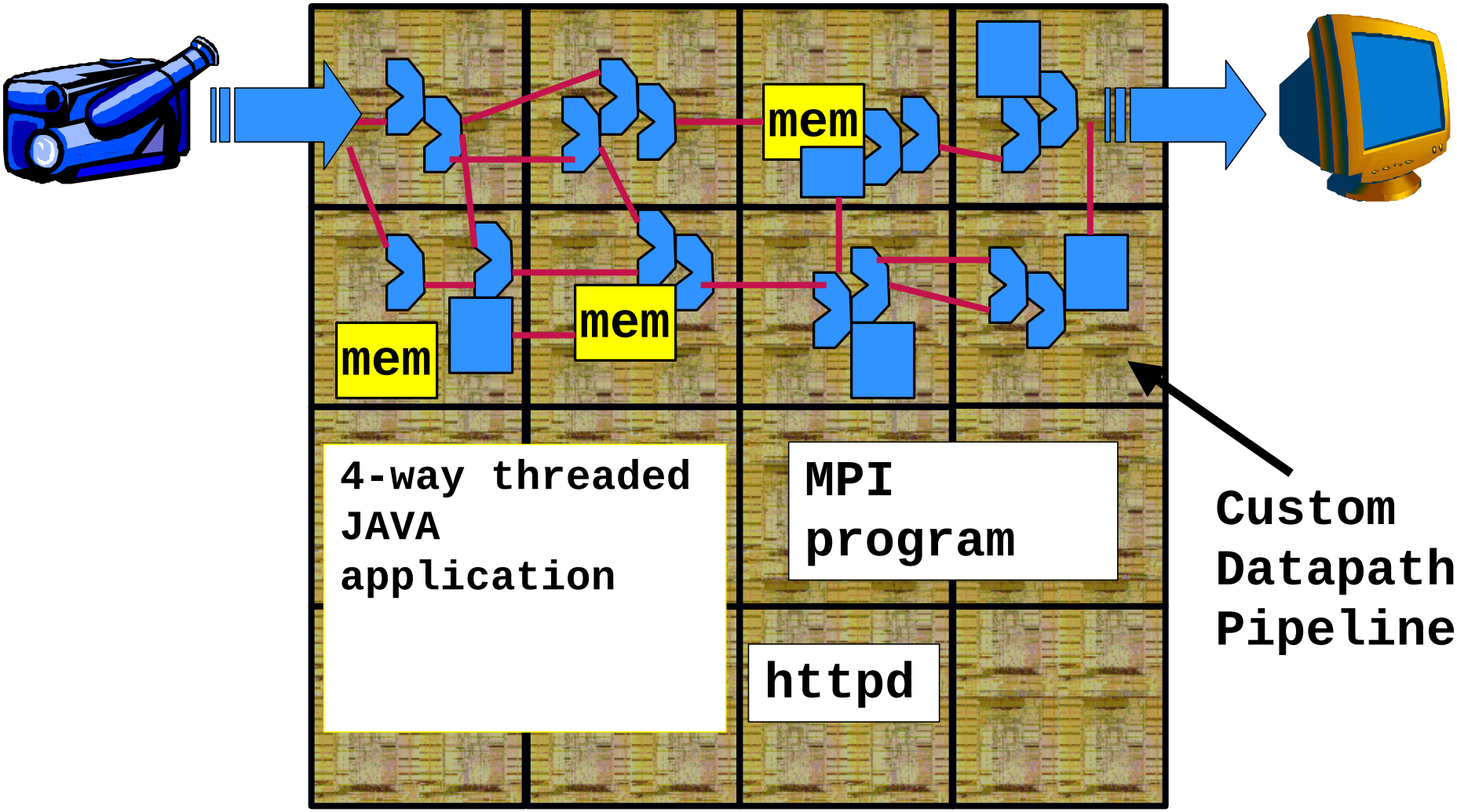
How well does Raw expose the resources?

Raw Chip (ASIC @225 MHz)

- 16 OPS/FLOPS per cycle
- 462 Gb/s of on-chip “bisection bandwidth”
- 201 Gb/s I/O bandwidth
- 57 GB/s of on-chip memory bandwidth

... but how are the resources going to be coordinated?

Raw: How we want to use the tiles



The Raw Tile network support

Tile processor

Computation Resources

4 32-bit mesh networks
2 static, 2 dynamic

5 stage
static
router
Pipeline

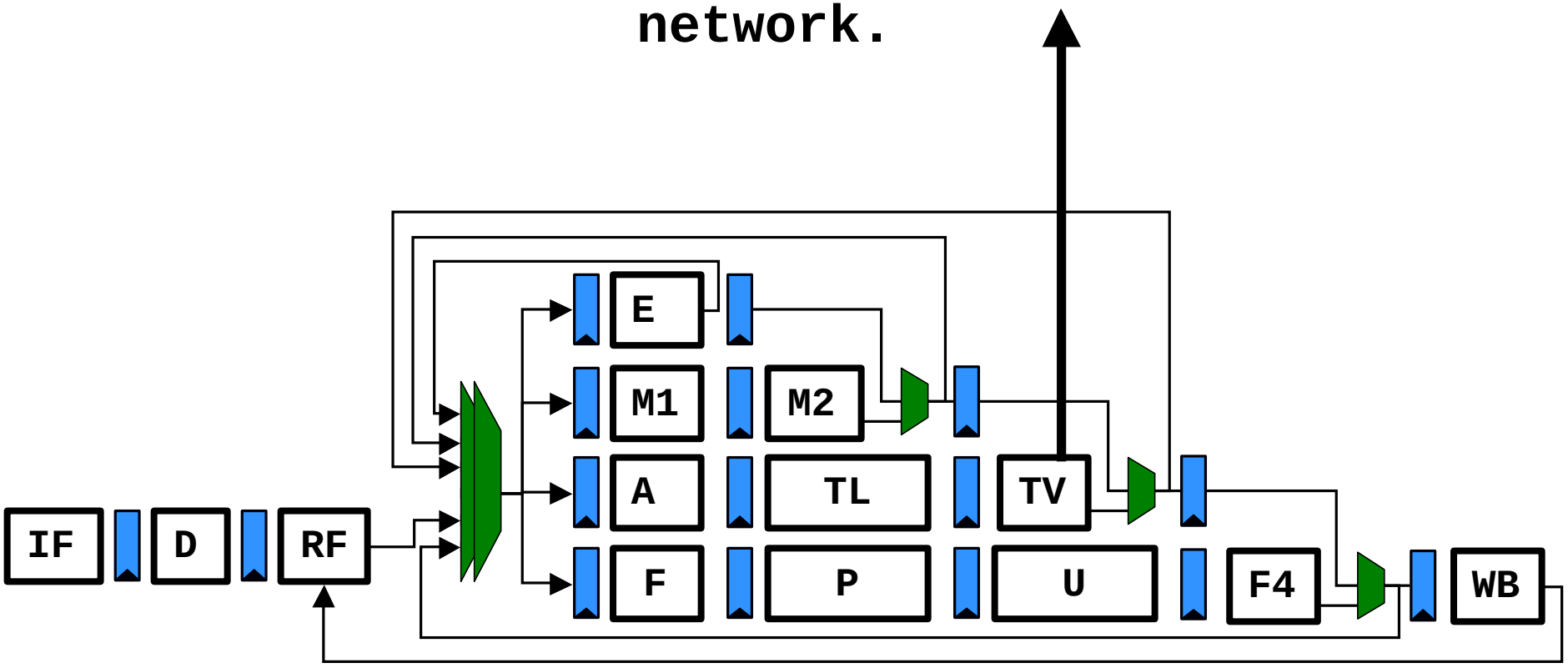
2 stage
dynamic
router
pipeline

64 KB SMem

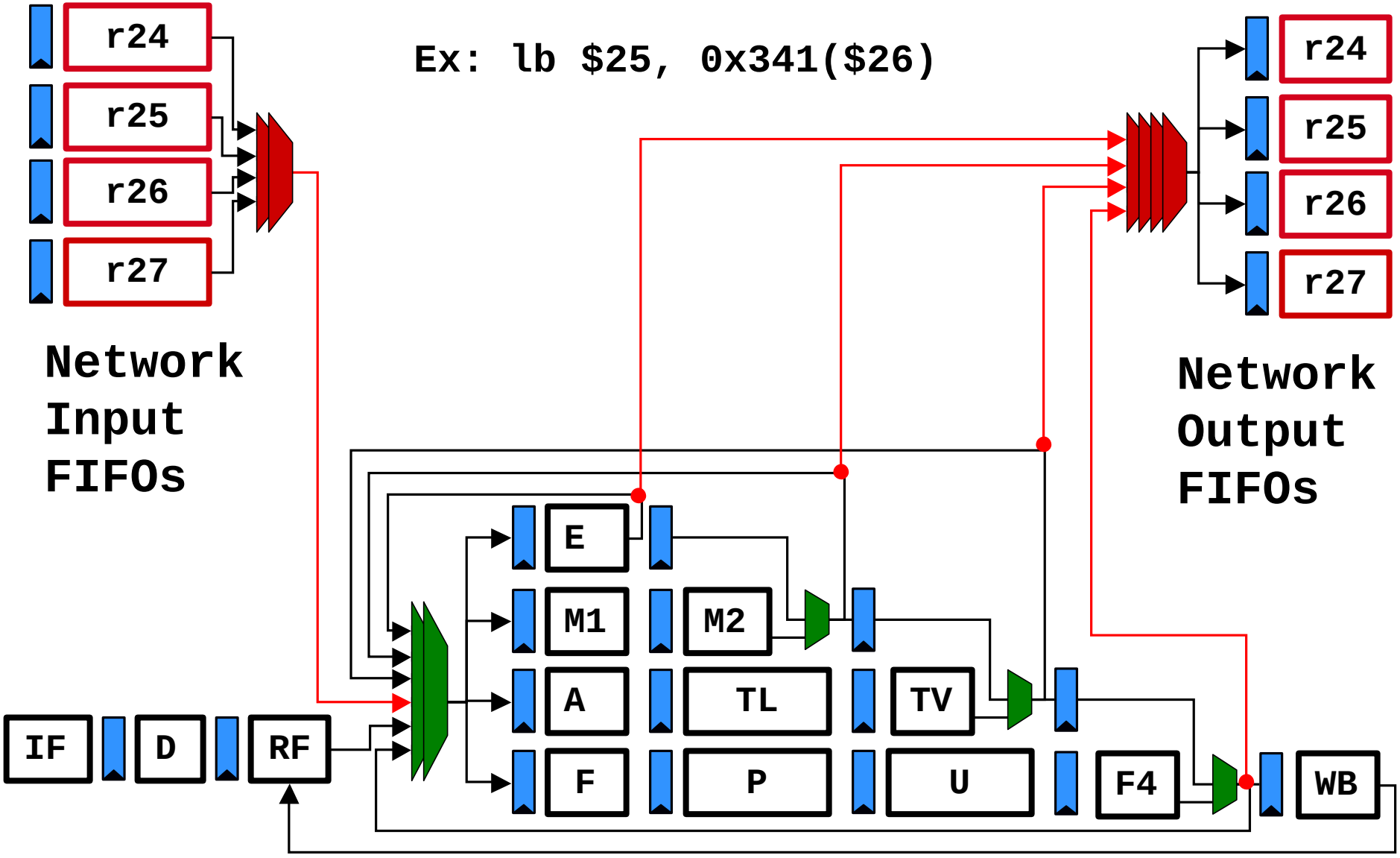
How does the main pipeline interface to the networks?

Memory mapped networks are not first class citizens.

To other tiles, through memory system that happens to go over a network.

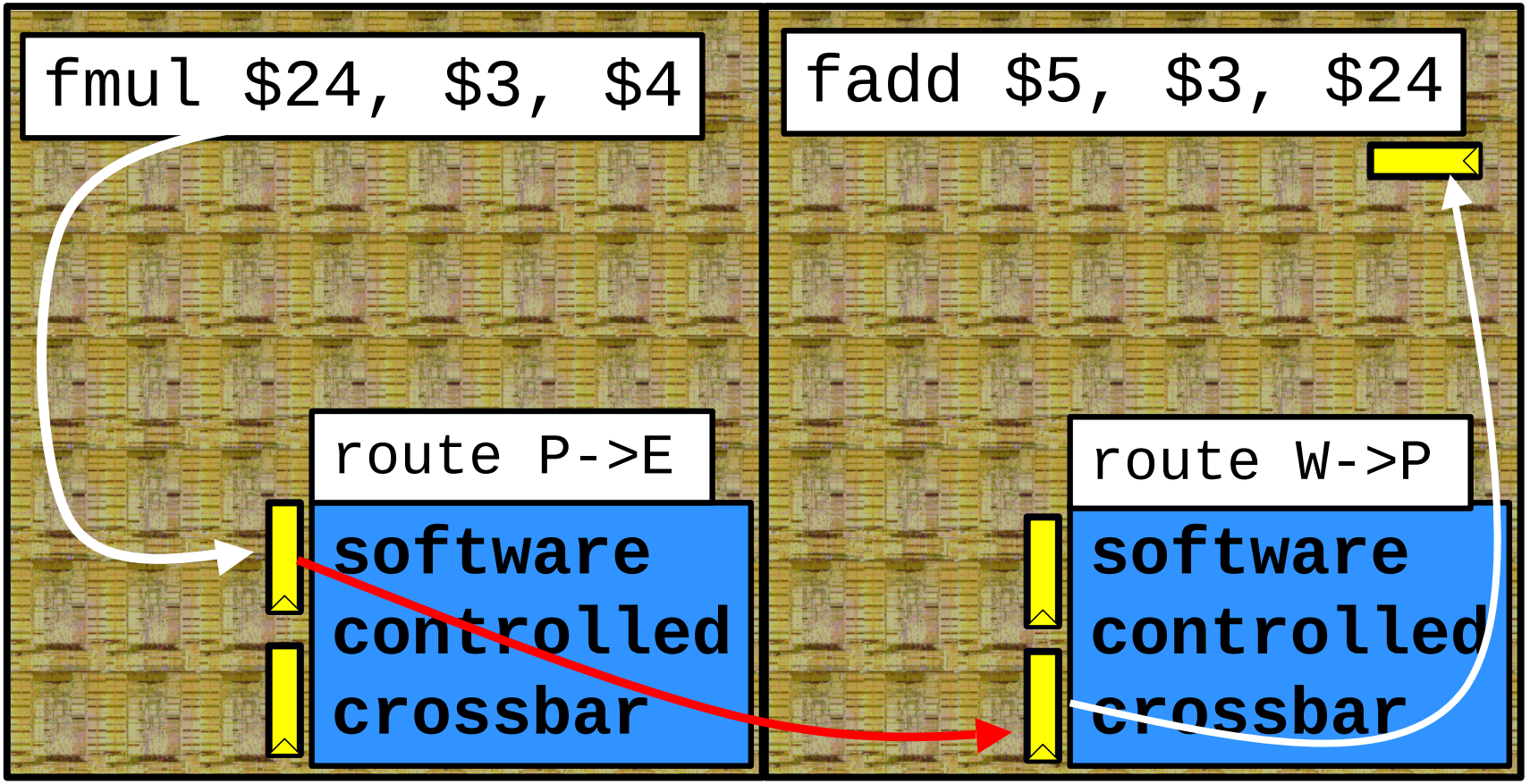


Instead, Raw's networks are tightly coupled into the bypass paths



How the static router works.

Goal: flow controlled,
in order delivery of operands

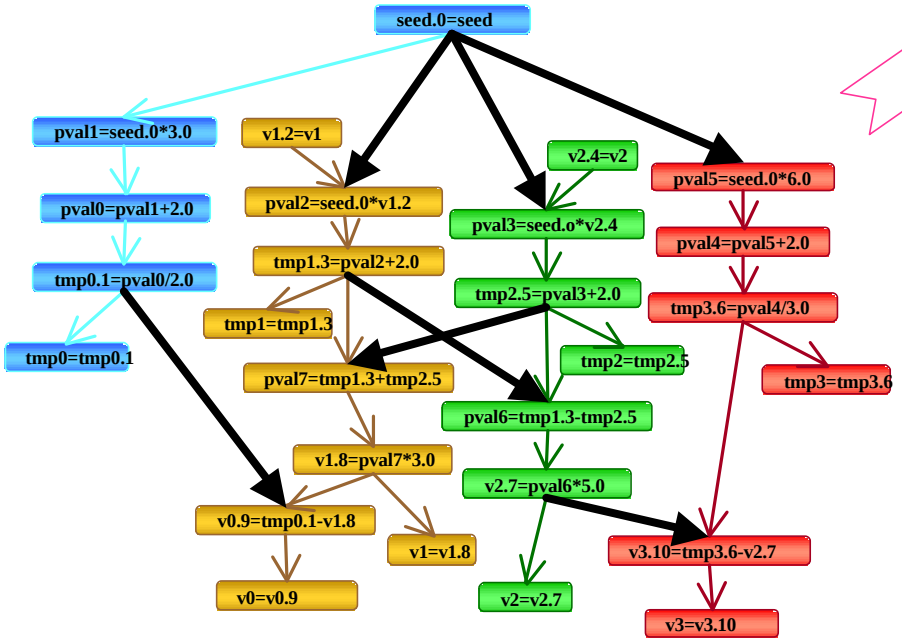


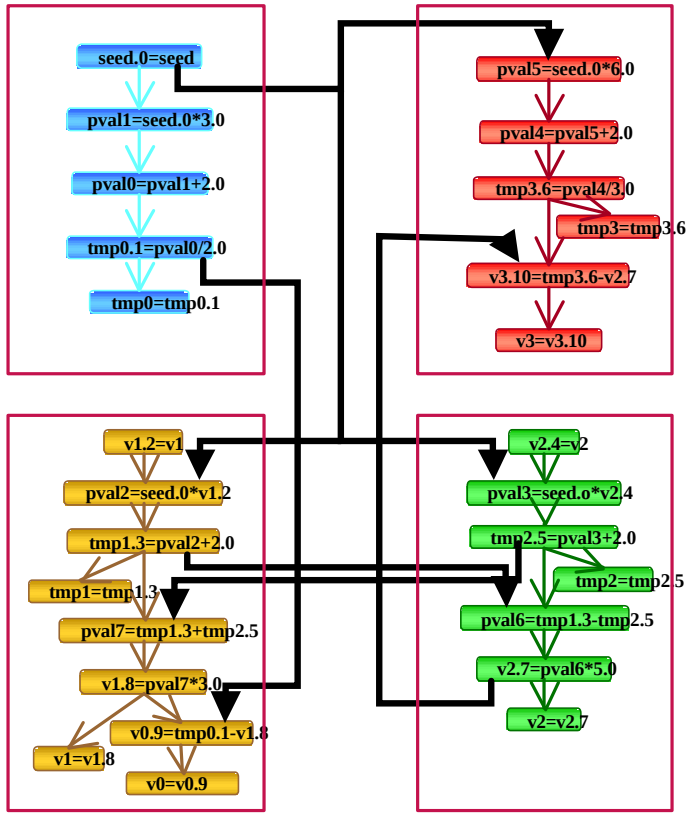
```
tmp0 = (seed*3+2)/2
tmp1 = seed*v1+2
tmp2 = seed*v2 + 2
tmp3 = (seed*6+2)/3
v2 = (tmp1 - tmp3)*5
v1 = (tmp1 + tmp2)*3
v0 = tmp0 - v1
v3 = tmp3 - v2
```

RawCC Operation:

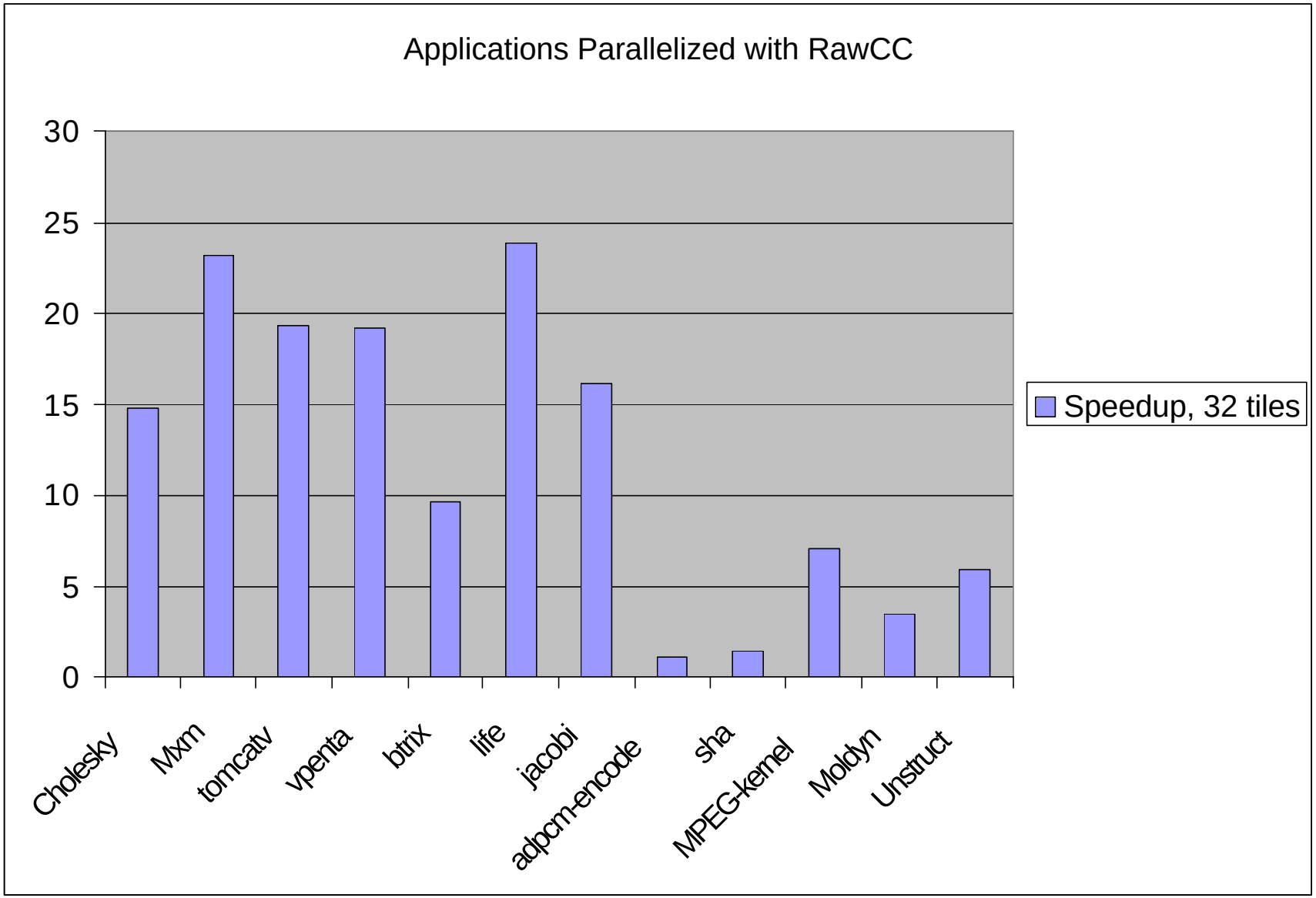
Parallelizes C code onto static network

Black arrows = Static Network





Low Network latency important.


Applications Parallelized with RawCC
30
25
20
15
10
5
0
Cholesky
Mxm
tomcatv
vpenta
btrix
life
jacobi
adpcm-encode
sha
MPEG-kernel
Moldyn
Unstruct
Speedup, 32 tiles

Raw Stats

IBM SA-27E .15u 6L Cu

18.2mm x 18.2mm die.

.122 Billion Transistors

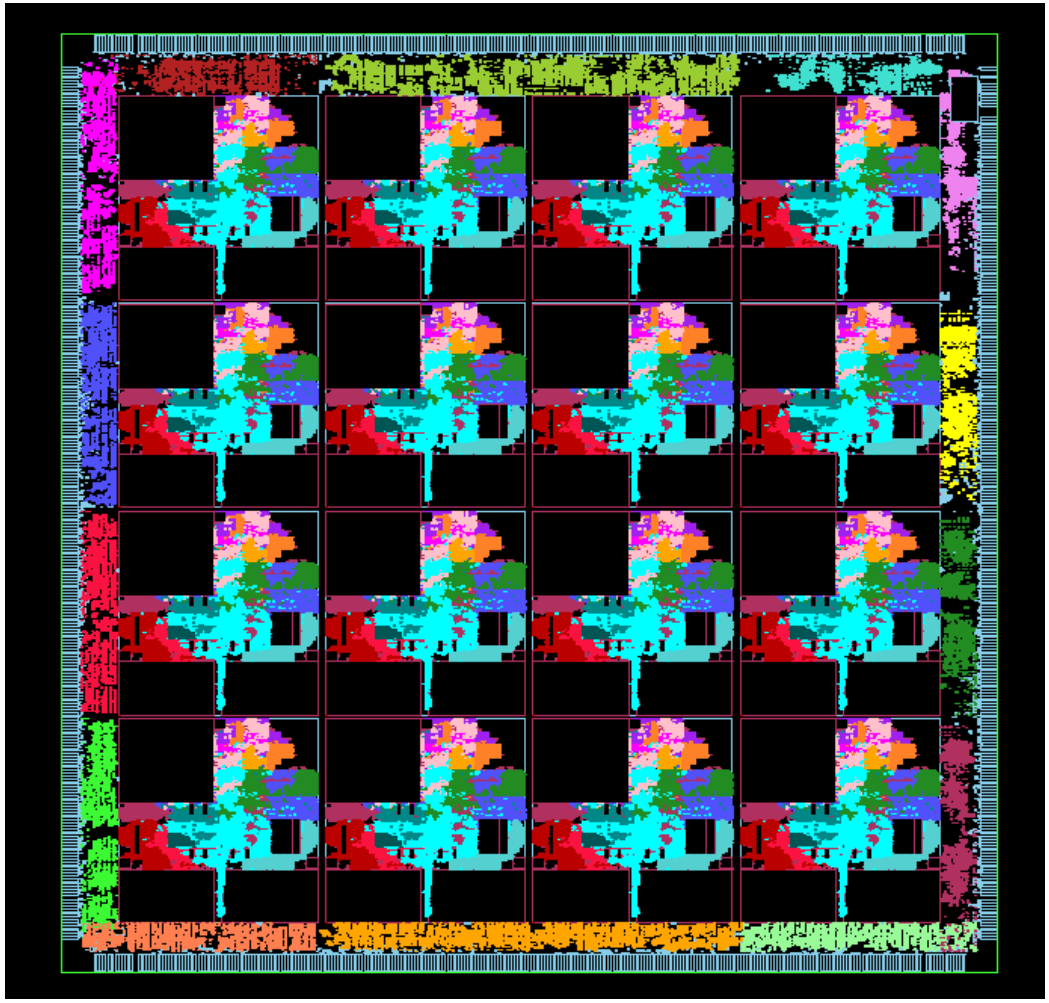
16 Tiles

2048 KB SRAM Onchip

1657 Pin CCGA Package
(1080 HSTL signal IO)

~225 MHz

~25 Watts

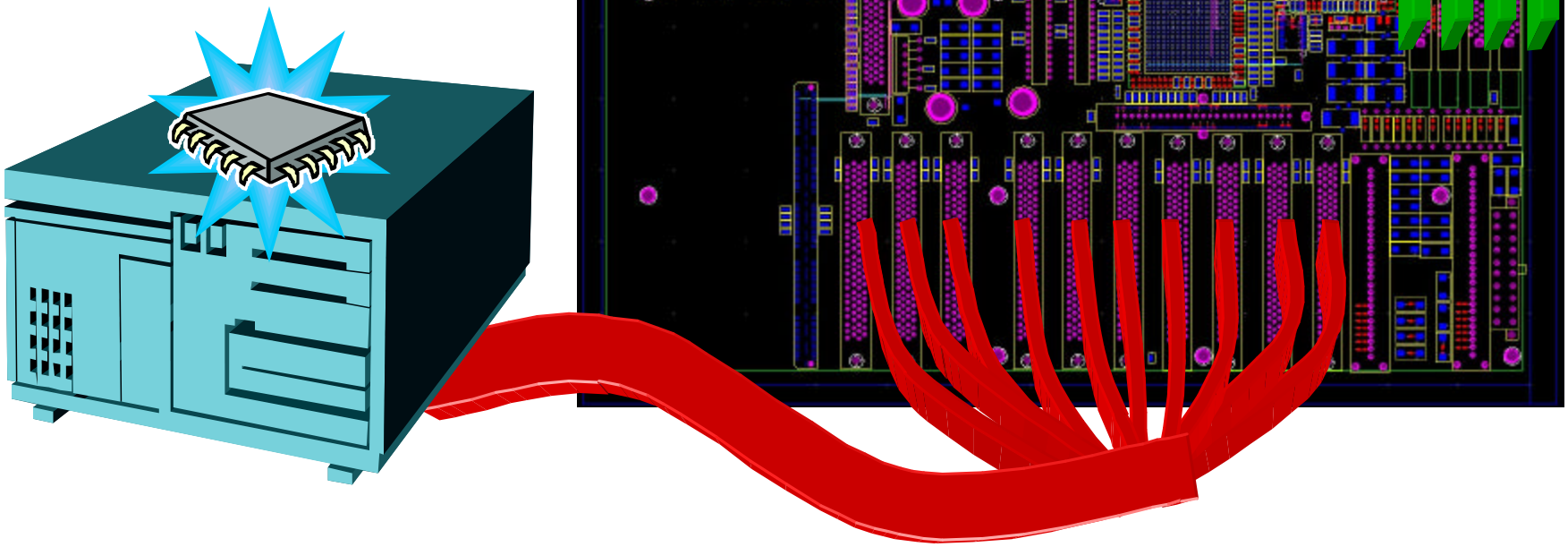


For architectural details, see:
http://cag.lcs.mit.edu/pub/raw/documents/RawSpec99.pdf

Raw Board with IKOS logic emulation

Currently in timing closure and verification.

Tape out: Q4 2001



Enabler: The Raw Networks

The Raw ISA treats the networks as first class citizens, just like registers:

software managed,
bypassed,
encoding space in every instruction

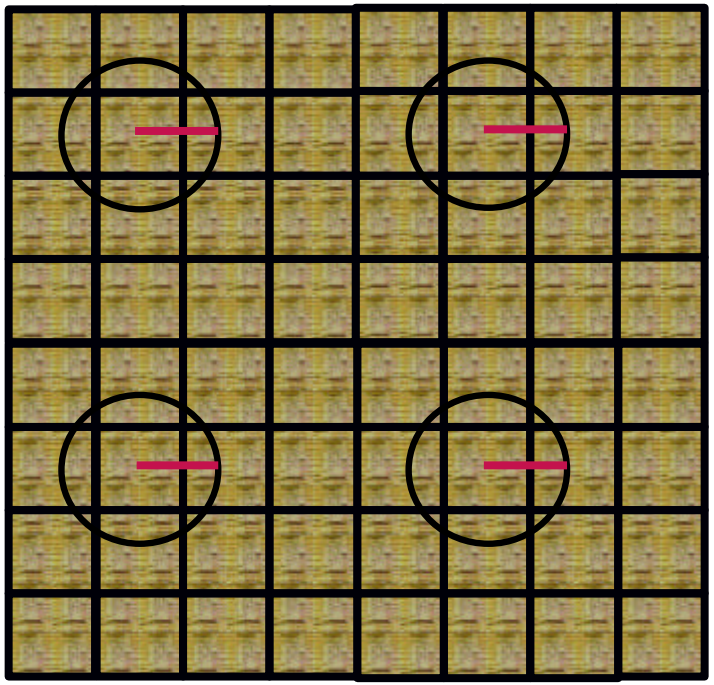
Static Network:

1. routes compiled into static router SMEM
2. Messages arrive in known order

Latency: 2 + # hops
Throughput: 1 word/cycle per dir. per network

Summary

Raw exposes wire delay at the ISA level. This allows the compiler to explicitly manage gates in a scalable fashion.



Raw provides a direct, parallel interface to all of the chip resources: gates, wires, and pins.

Raw enables the use of these gates by providing tightly coupled network communication mechanisms in the ISA.