https://inst.eecs.berkeley.edu/~eecs251b/sp24/

EECS251B
Advanced Digital Circuits and Systems

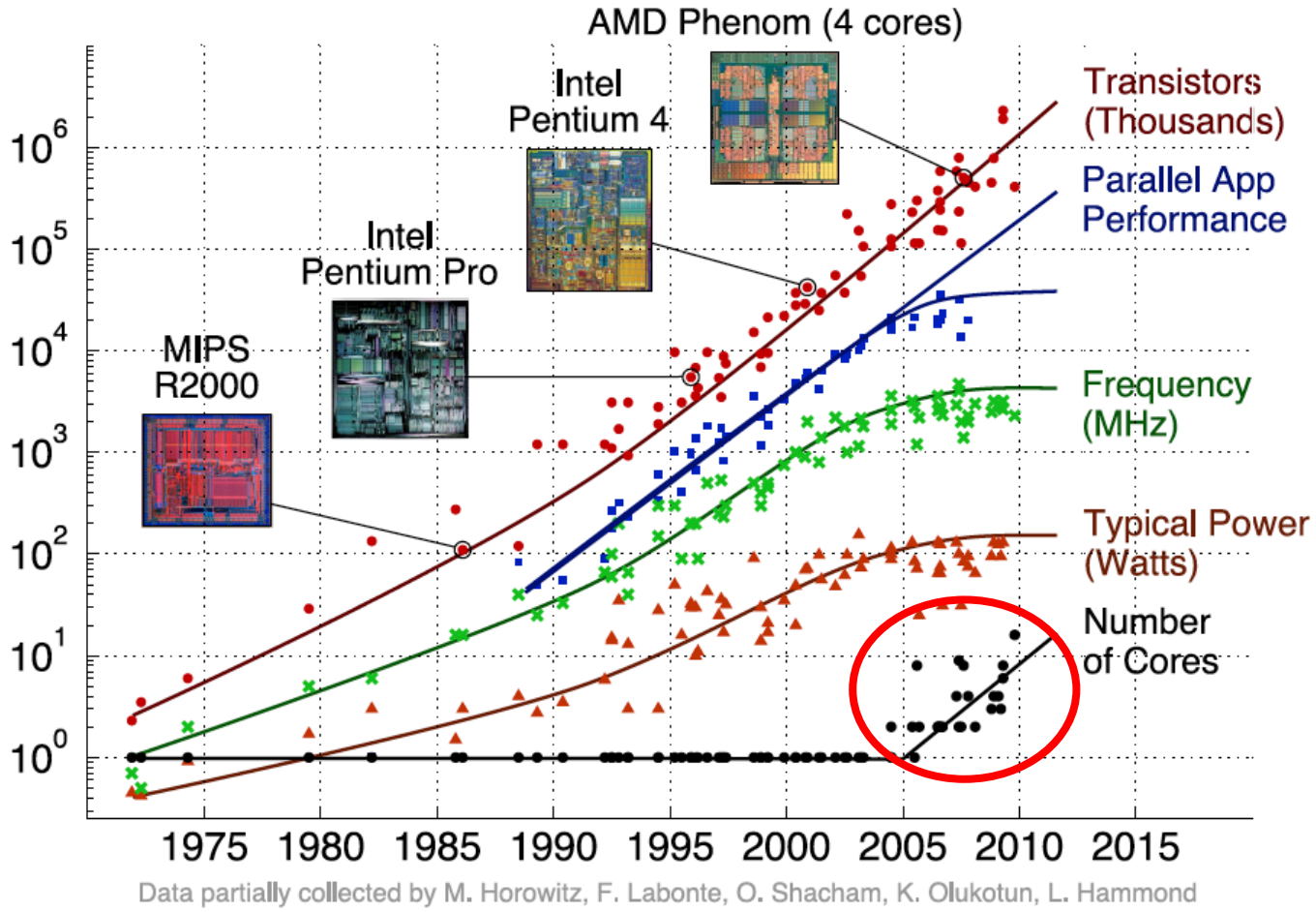
Lecture 5&6 – System Interconnect

Borivoje Nikolic

Tuesdays and Thursdays 9:30-11am

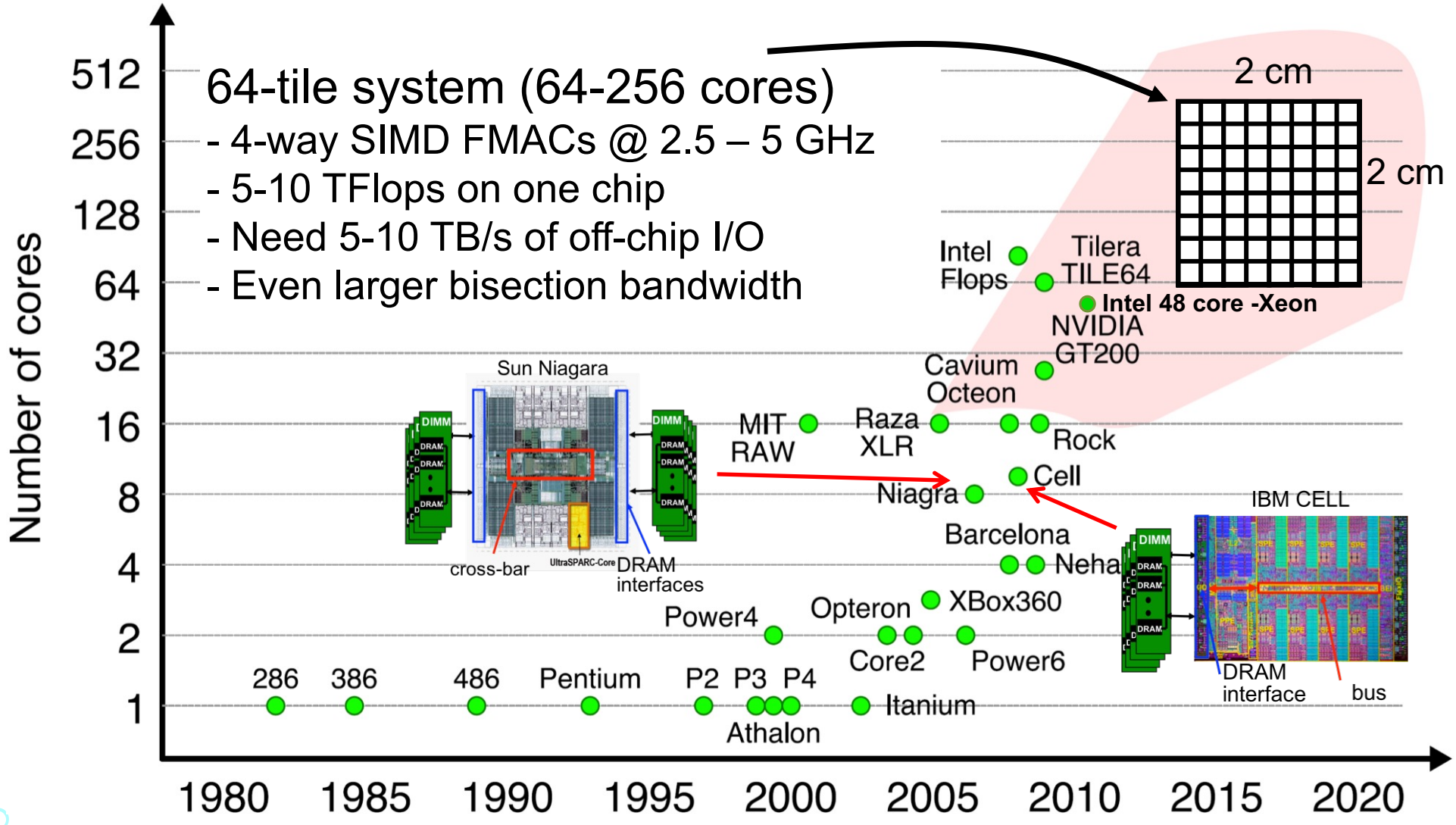
Cory 521

Power and Performance Trends



- With clock frequencies saturating CPUs started using many cores to leverage parallelism and deal with fabrication yields

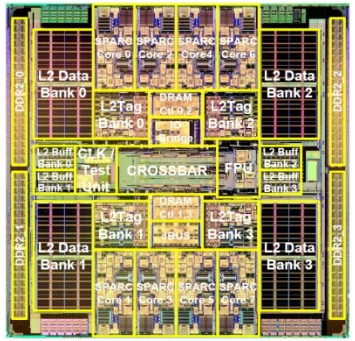
Manycore System Roadmap



The rise of manycore machines

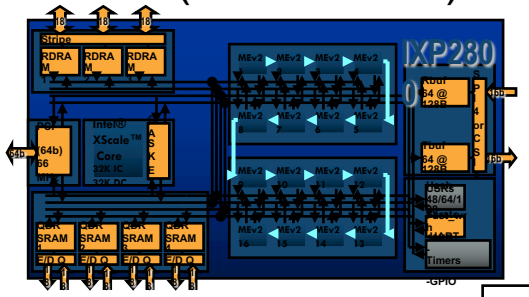
Only way to meet future system feature set, design cost, power, and performance requirements is by programming a processor array

- Multiple parallel general-purpose processors (GPPs)
- Multiple application-specific processors (ASPs)

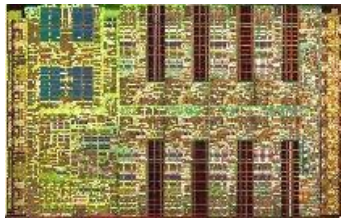


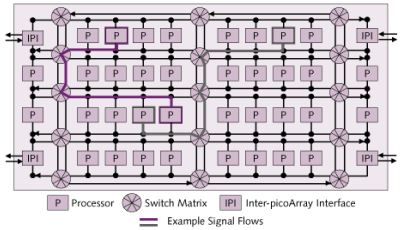
Sun Niagara
8 GPP cores (32 threads)

Intel Network Processor
1 GPP Core
16 ASPs (128 threads)

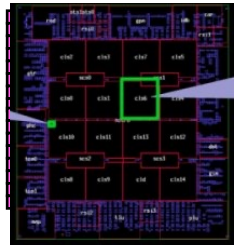


IBM Cell
1 GPP (2 threads)
8 ASPs

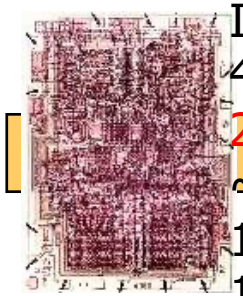




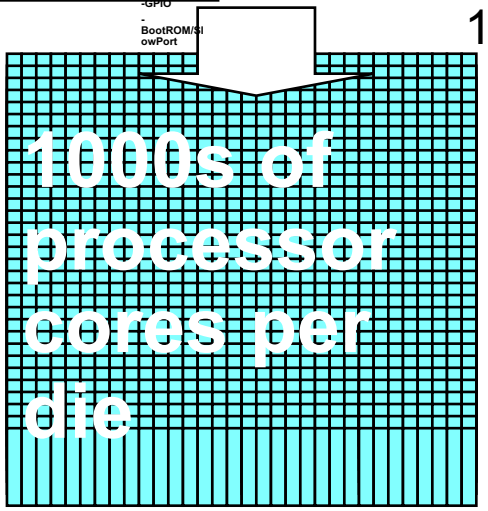
Picochip DSP
1 GPP core
248 ASPs



Cisco CRS-1
192 Tensilica GPPs

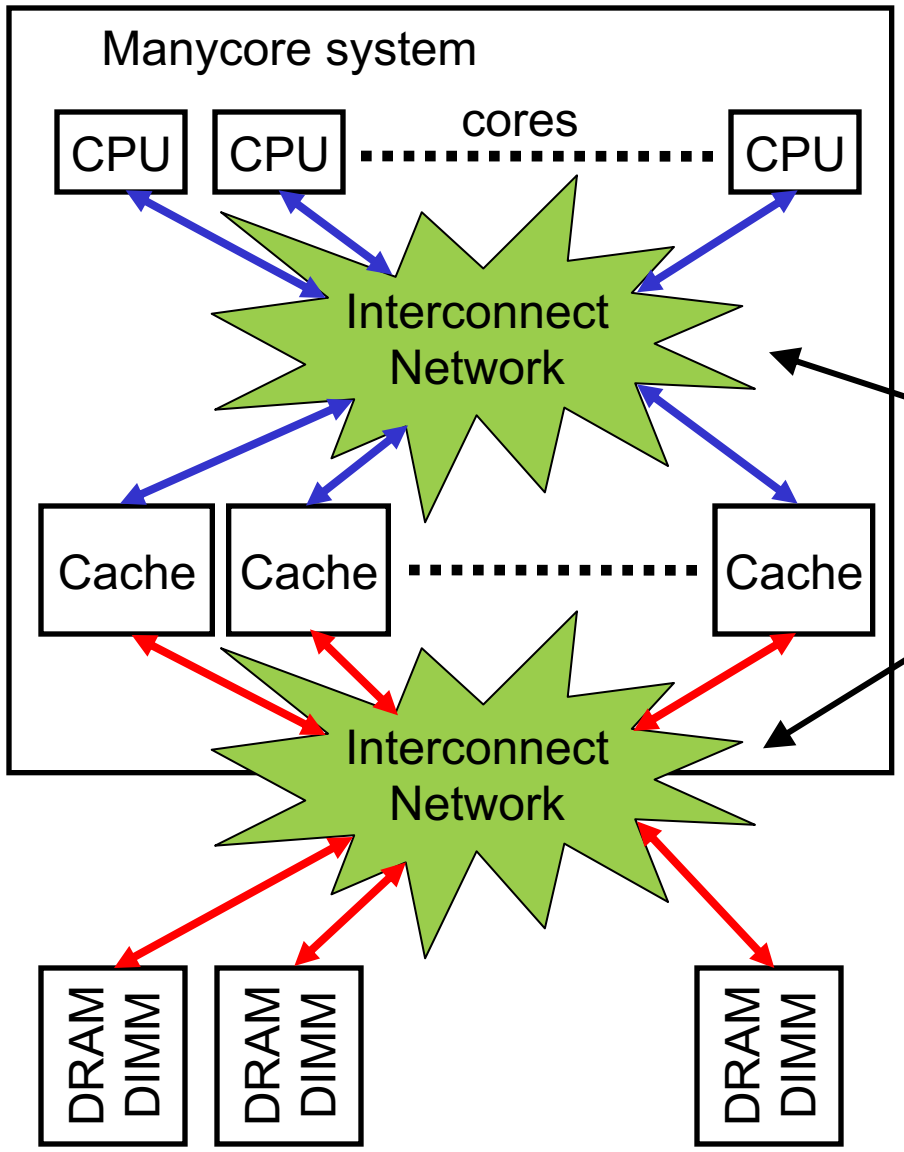


Intel 4004 (1971):
4-bit processor,
2312 transistors,
~100 KIPS,
10 micron PMOS,
11 mm^2 chip



"The Processor is the new Transistor" [Rowen]

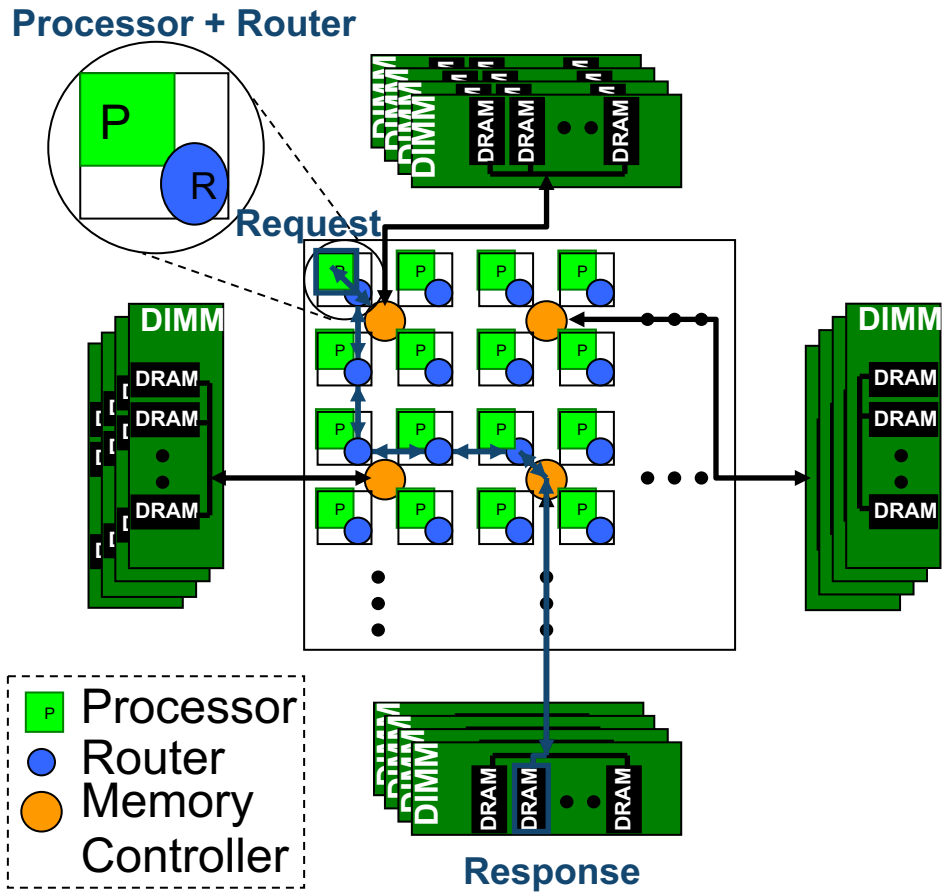
Interconnect bottlenecks

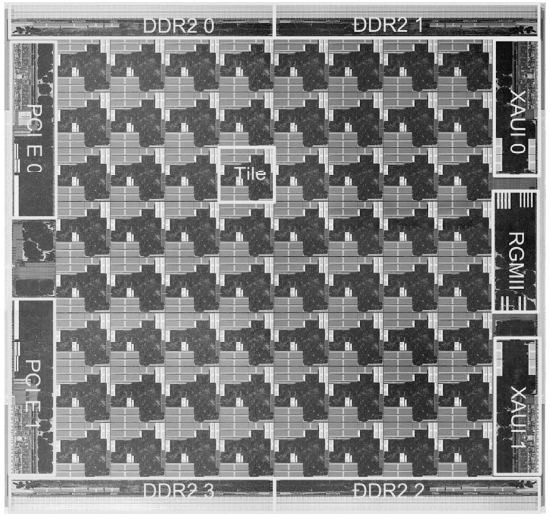


Bottlenecks due to energy and bandwidth density limitations

Need to jointly optimize on-chip and off-chip interconnect network

Scaling to many cores

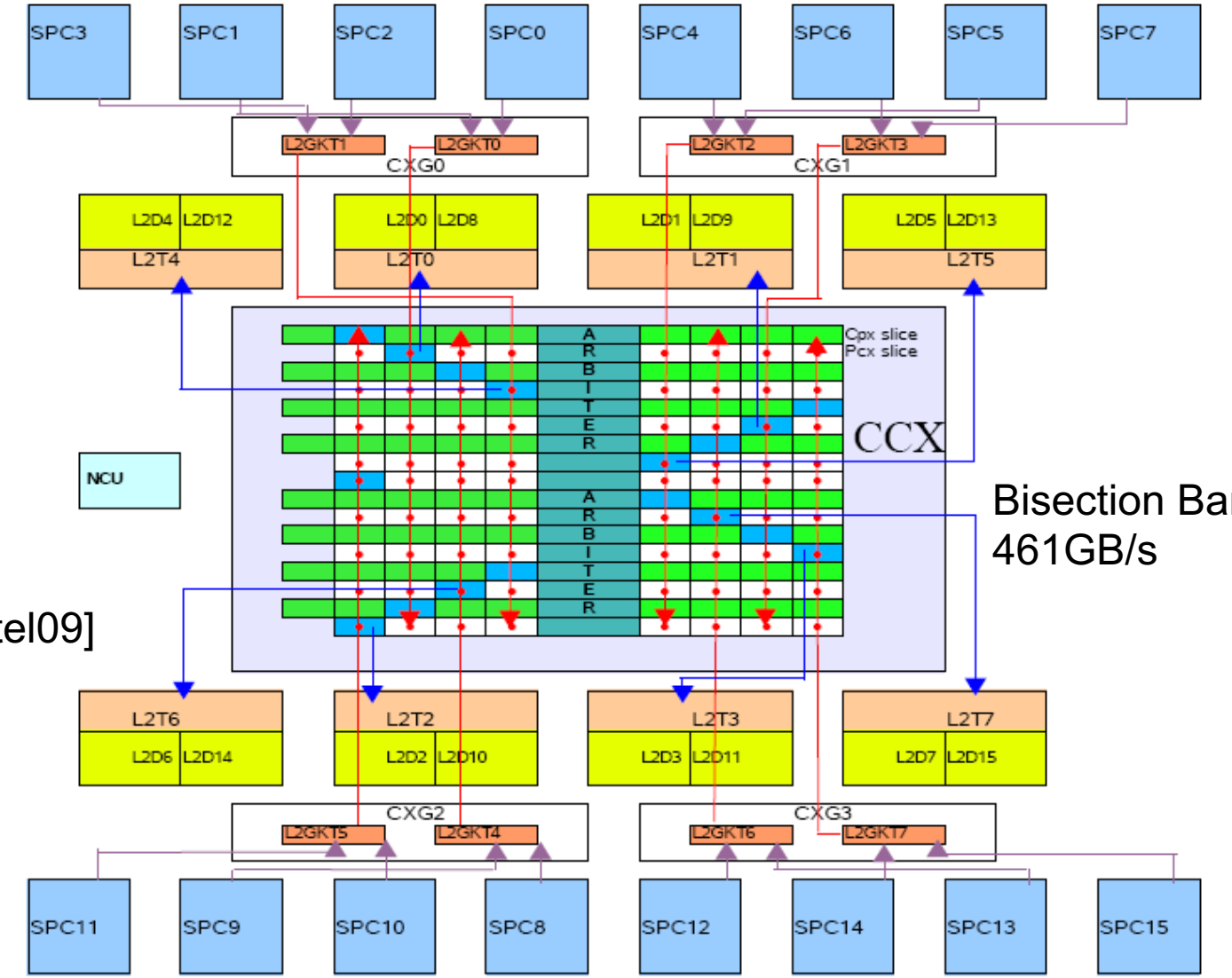




TILE64
[Bell08]

- Networks-on-chip
  - Many meshes
    - Slow, latency varies greatly
    - Easy to implement
  - Large crossbars
    - Fast, predictable latency
    - Hard to build and scale
  - Rings

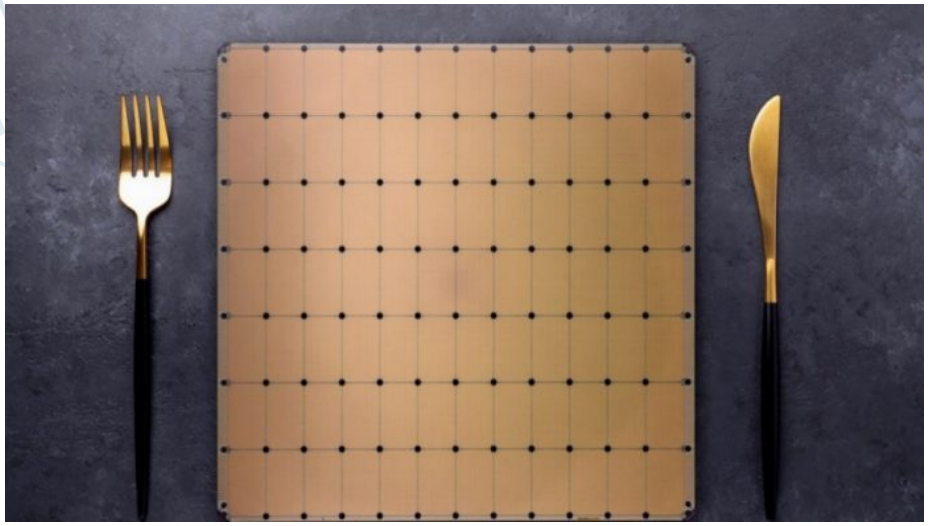
Rainbow-Falls 2-stage Crossbar

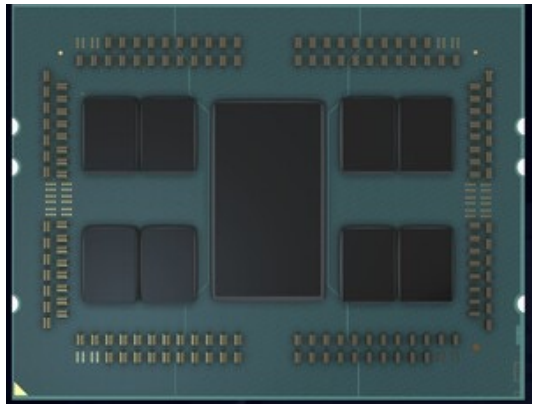


[Patel09]

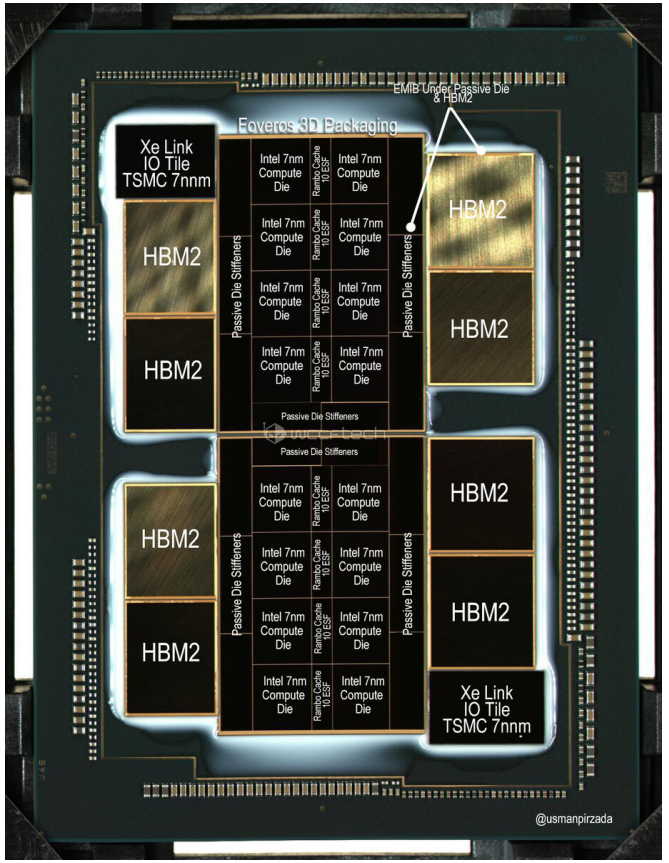
Bisection Bandwidth
461GB/s

Recent trends

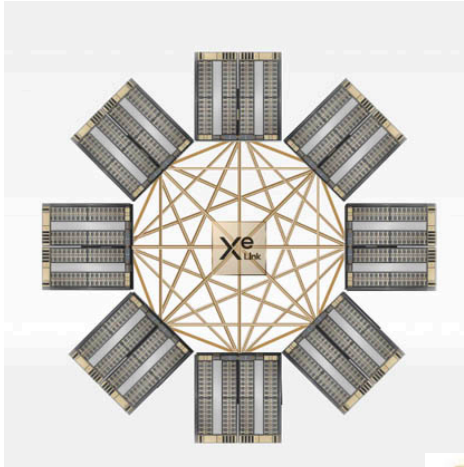




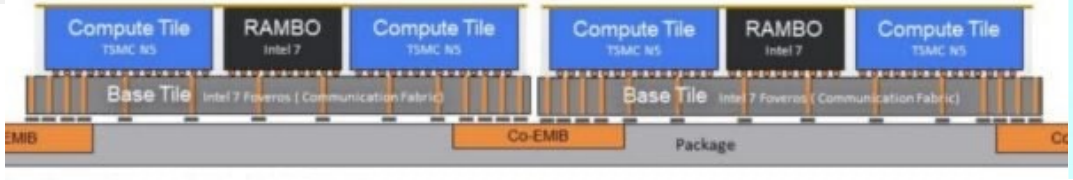
[AMD] Milan/Rome CPUs
>100B Transistors
8 CPU die 1 I/O die
64 cores/128 Threads
280W



[Cerebras Systems] WSE-2
2.6T Transistors
850,000 AI optimized cores
15kW
40GB on-chip SRAM
Mem BW 20PB/s (on-chip)
On-chip Fabric BW 220Pb/s



[Intel] Ponte Vecchio GPU
>100B Transistors
47 Active Tiles
120GB on-package HBM
Multi-package interconnect

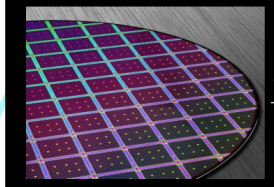


Rack-scale systems

Dojo Training Tile

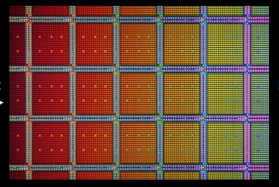
Tile (25 Chips) vs PCIe Card (1 or 2 Chips)

Dojo Unique Innovation: Flattened Hierarchies



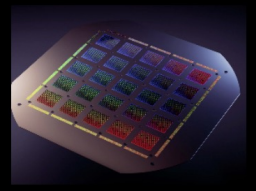
Silicon Wafer

Test & Sort



Known Good Dies

New Integration



Reconstructed Fanout Wafer

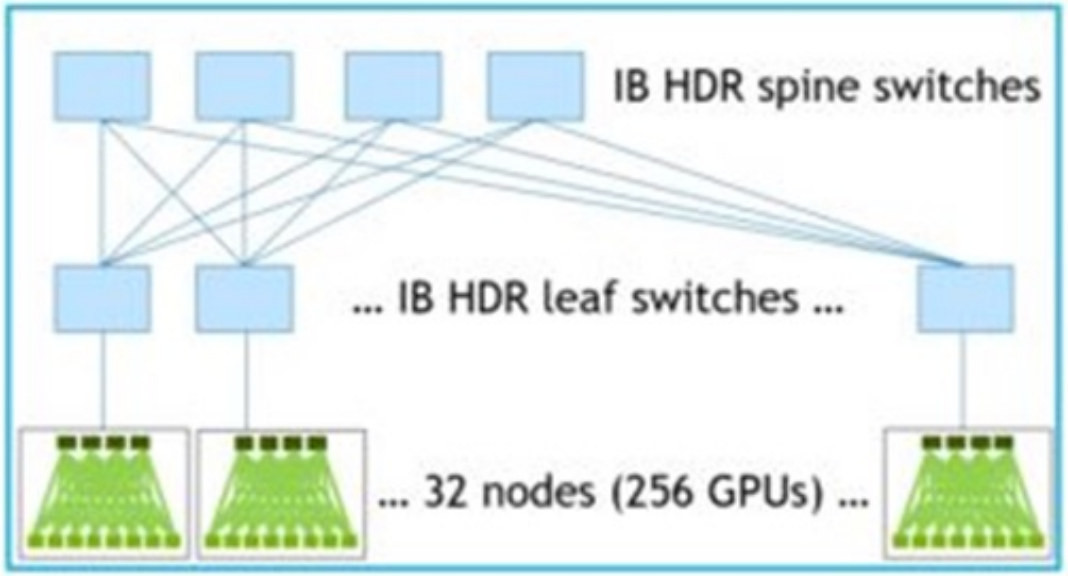
9 PFLOPs (BFP16/CFP8)
40 TB/s Bisec BW (X+Y)
36TB/s I/O BW

V1 Dojo Training Matrix

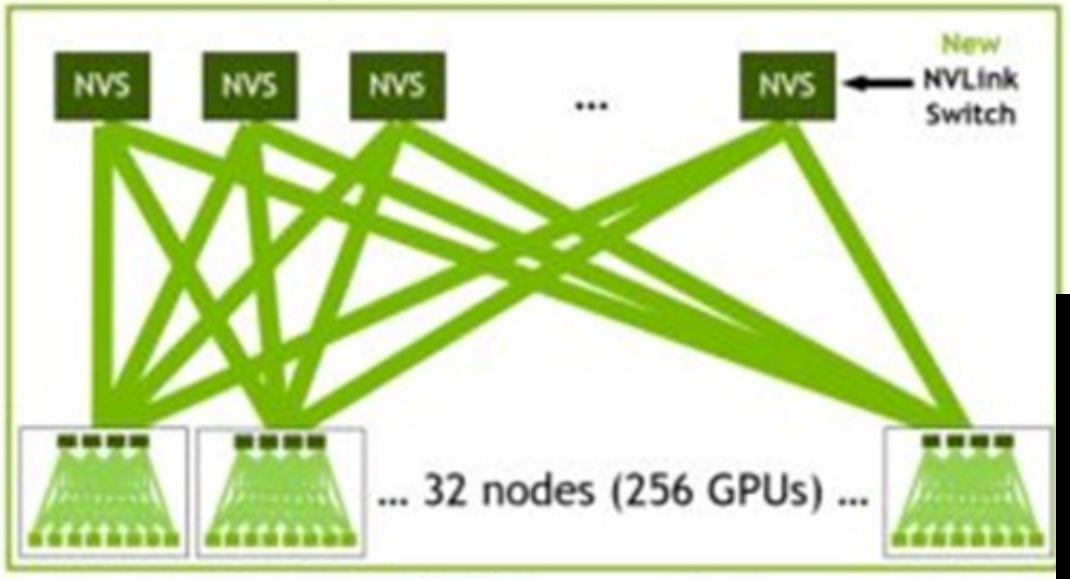
HotChips'22

Expansion of memory-semantic fabrics

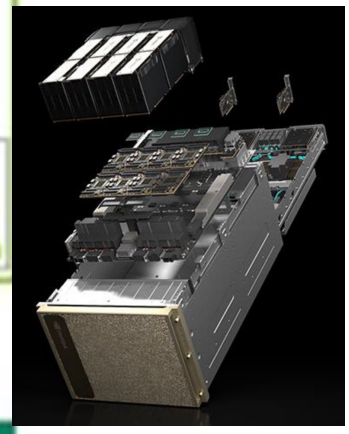
DGX A100 256 SuperPOD



DGX H100 256 SuperPOD



Fully NVLink-connected
Massive bisection bandwidth



	A100 SuperPod			H100 SuperPod			Speedup	
	Dense PFLOP/s	Bisection [GB/s]	Reduce [GB/s]	Dense PFLOP/s	Bisection [GB/s]	Reduce [GB/s]	Bisection	Reduce
1 DGX / 8 GPUs	2.5	2,400	150	16	3,600	450	**1.5x**	**3x**
32 DGXs / 256 GPUs	80	6,400	100	512	57,600	450	**9x**	**4.5x**

Lecture Roadmap

- Networking Basics
- Building Blocks
- Evaluation

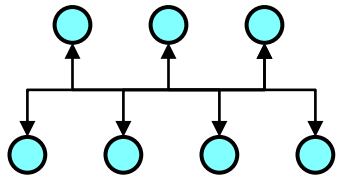
Lecture Roadmap

- Networking Basics
  - Topologies
  - Routing
  - Flow-Control
- Building Blocks
- Evaluation

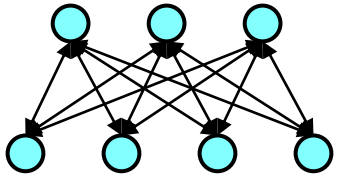
Lecture Roadmap

- Networking Basics
  - Topologies
  - Routing
  - Flow-Control
- Building Blocks
- Evaluation

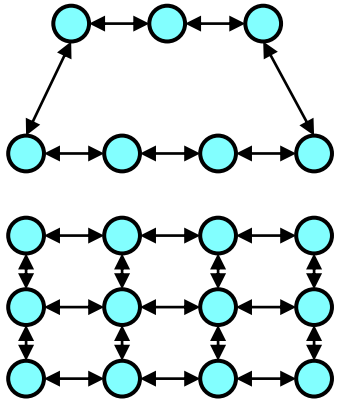
Topologies



- Shared-bus
  - Advantages: cheap & easy to implement, broadcast, serialized messages
  - Disadvantages: low bandwidth, tri-state logic



- Crossbar – all-to-all connection
  - Advantages: high bandwidth due to all-to-all routing, no contention, serialized messages, predictable latency
  - Disadvantages: $O(n^2)$ scaling, scales poorly past 4x4 networks

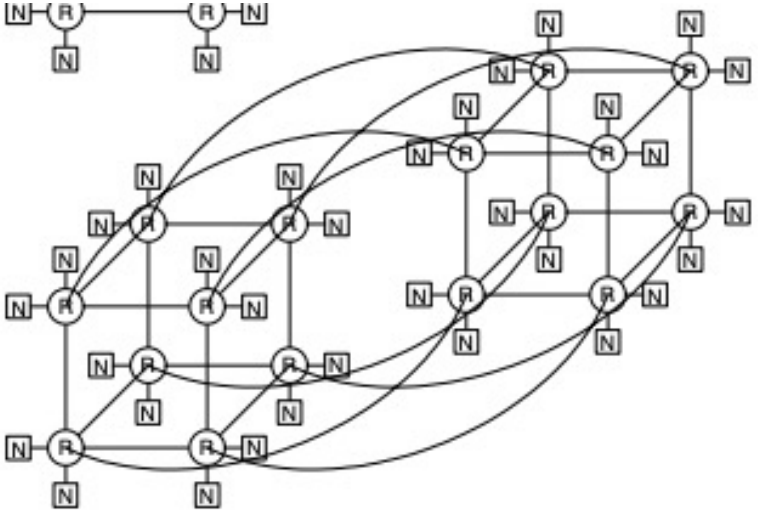


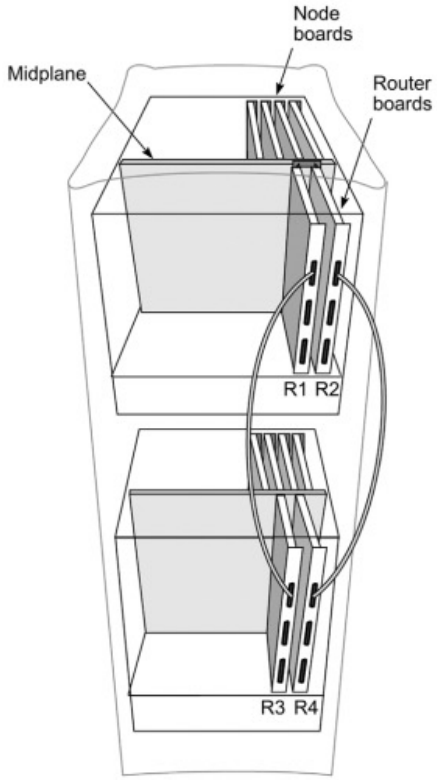
- **1D torus/ring (unidirectional/bidirectional)**
  - Advantages: simple to implement, well-behaved
  - Disadvantages: low bisection bandwidth, high-hop-count
- **2D mesh**
  - Advantages: scalable with good bandwidth/low-latency
  - Disadvantages: complex routing for deadlock-freedom

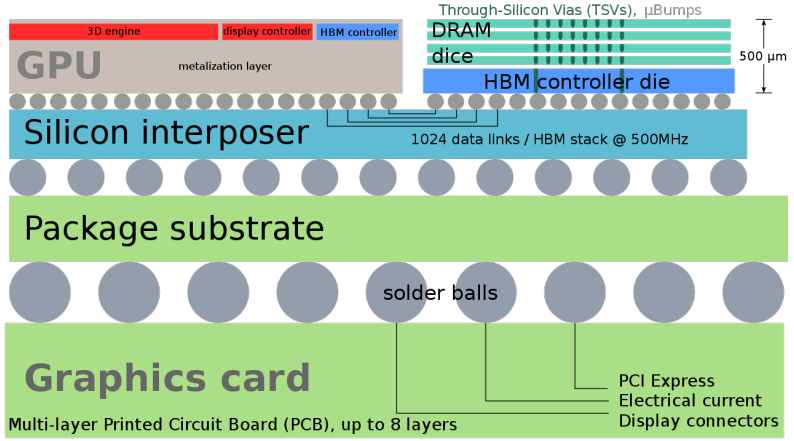
These are **network-on-chips (NoCs),**
with proper routers and channels

3D Topologies

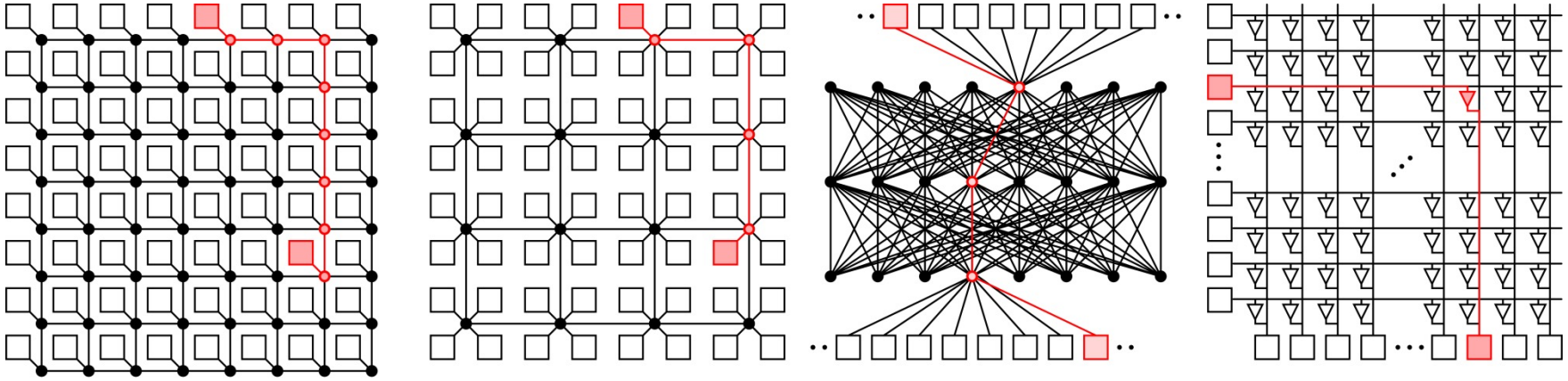
- Before single-chip multi-cores, 3D topologies were used for rack-scale computers
  - SGI Origin 2000
  - Hyper-cube topology to maximize bandwidth
- 3D topologies don't match 2D silicon architectures
- New opportunities with 3D stacking







Network topology spectrum



Increasing diameter

Easy to design
Hard to program

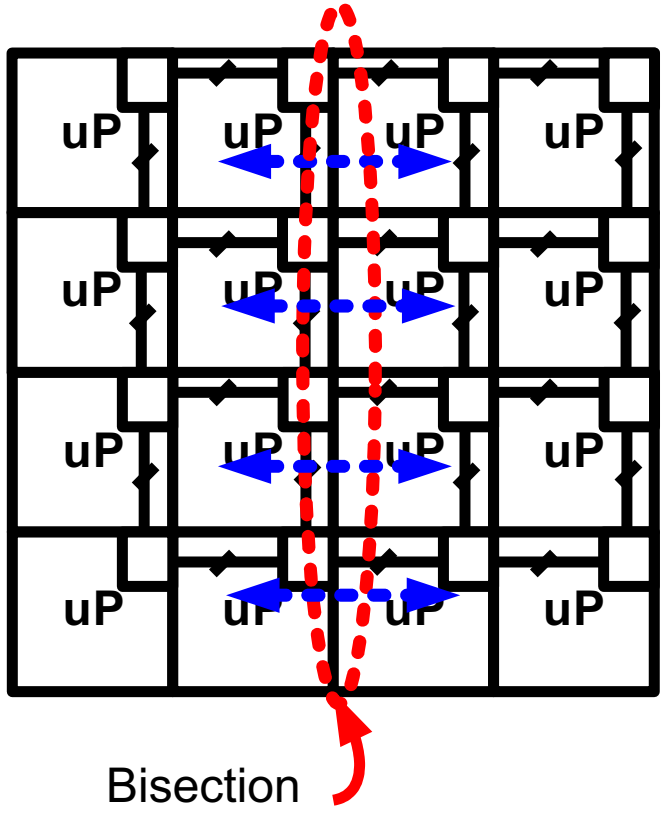
Hard to design
Easy to program

Increasing radix

Radix – Number of inputs and outputs of each switching node
Diameter – largest minimal hop count over all node pairs

In **power constrained** systems – Need to look at networks in a **cross-cut** approach
Connect physical implementation (channels, routers, power) with **network topology, routing and flow-control**

Ideal network throughput (capacity)



- Maximum traffic that can be sustained by all cores
- Mesh throughput
  - 50% of data crosses the bisection assuming uniform random traffic
- Bisection bandwidth $= 2\sqrt{N}b$
- Data crossing the bisection $= \frac{1}{2}Nb_{core}$
- Maximum throughput

$$\Theta_{ideal} = Nb_{core} = 4\sqrt{N}b$$

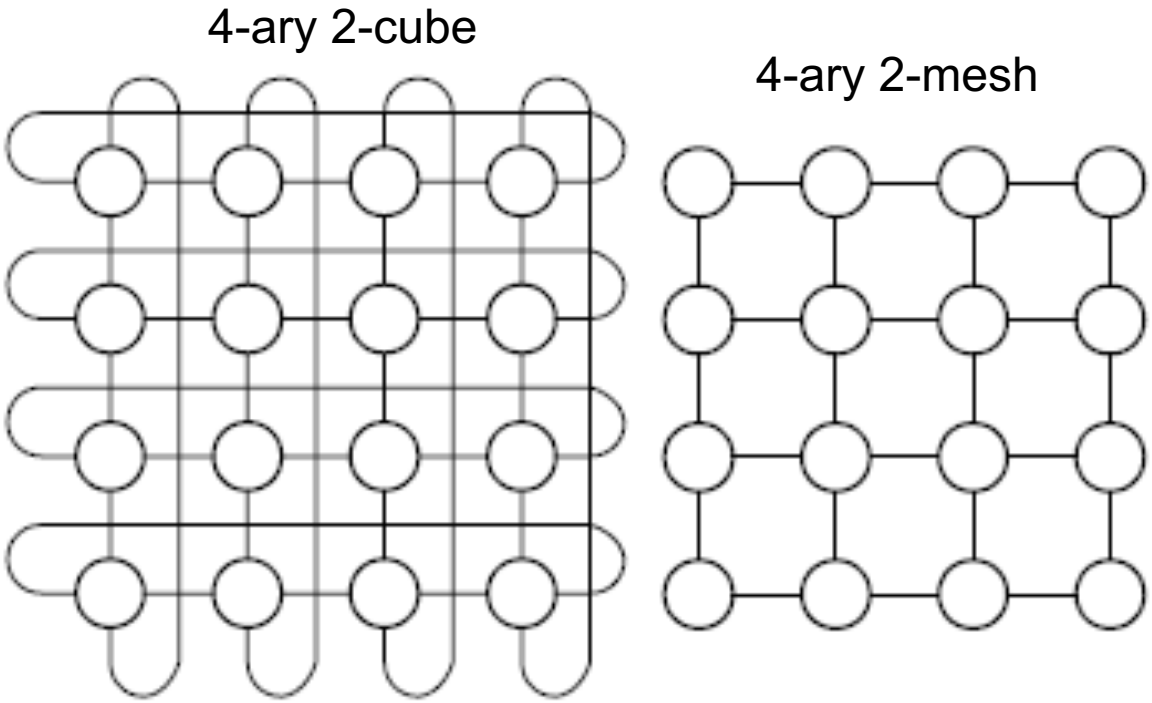
N = number of cores
b = router-to-router link bandwidth
b_{core} = rate at which each core generates traffic

To maximize bandwidth, a topology should saturate the bisection bandwidth

Tori

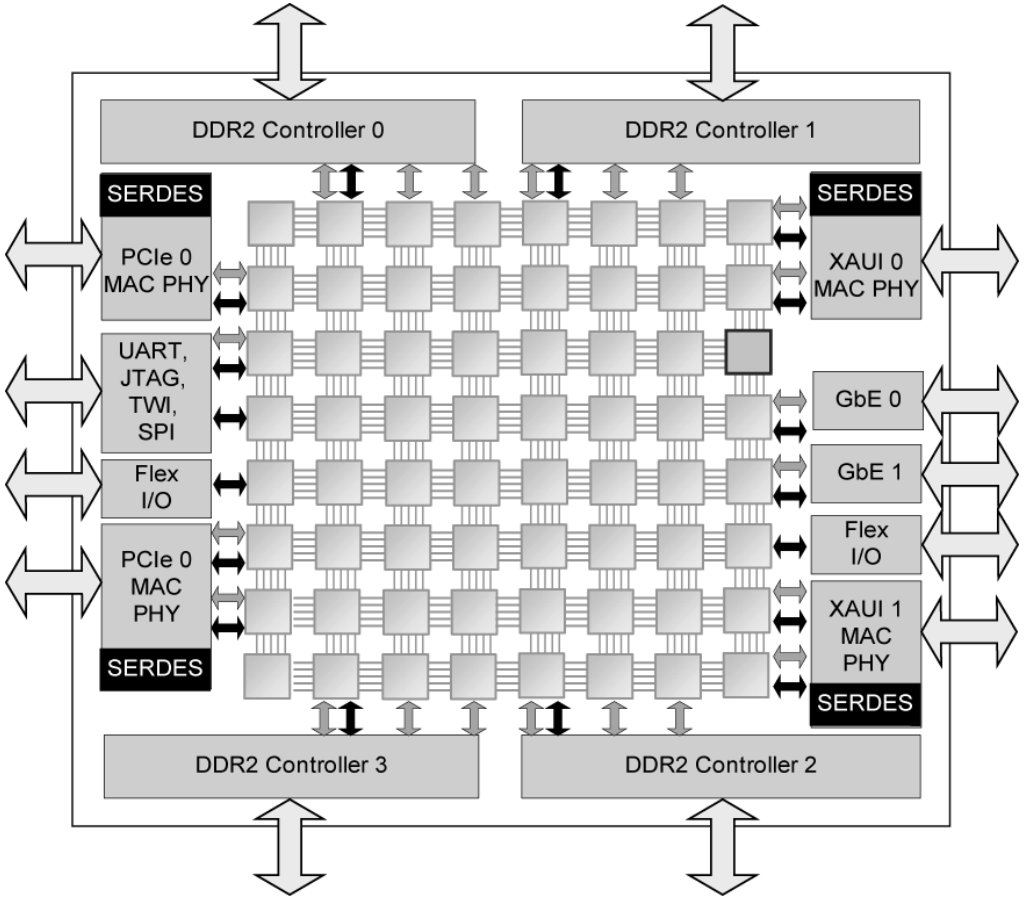
[Dally04]

- Low-radix, large diameter networks
- N-ary, K-cube (mesh)
  - N nodes per dimension
  - K dimensions

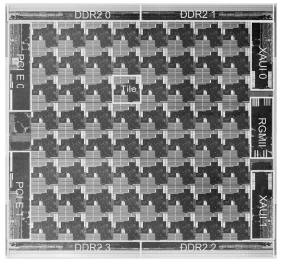


- Cubes have 2x larger bisection bandwidth

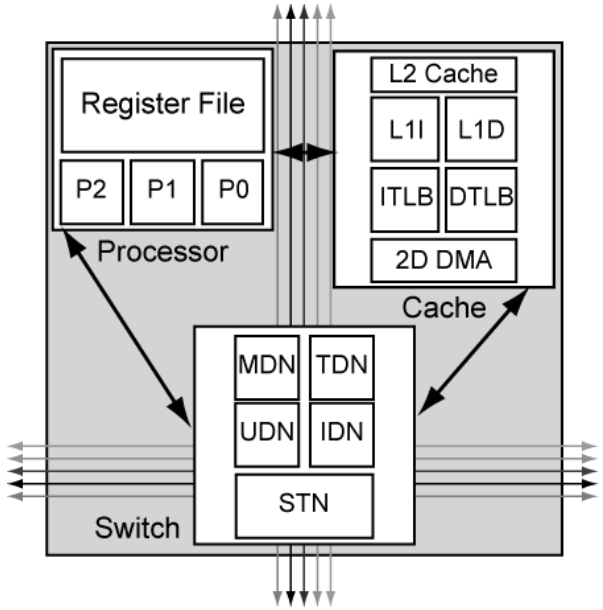
TILE64



[Bell08]

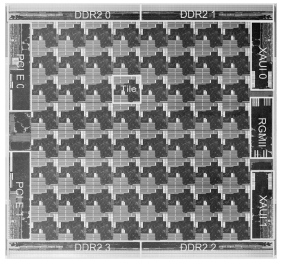


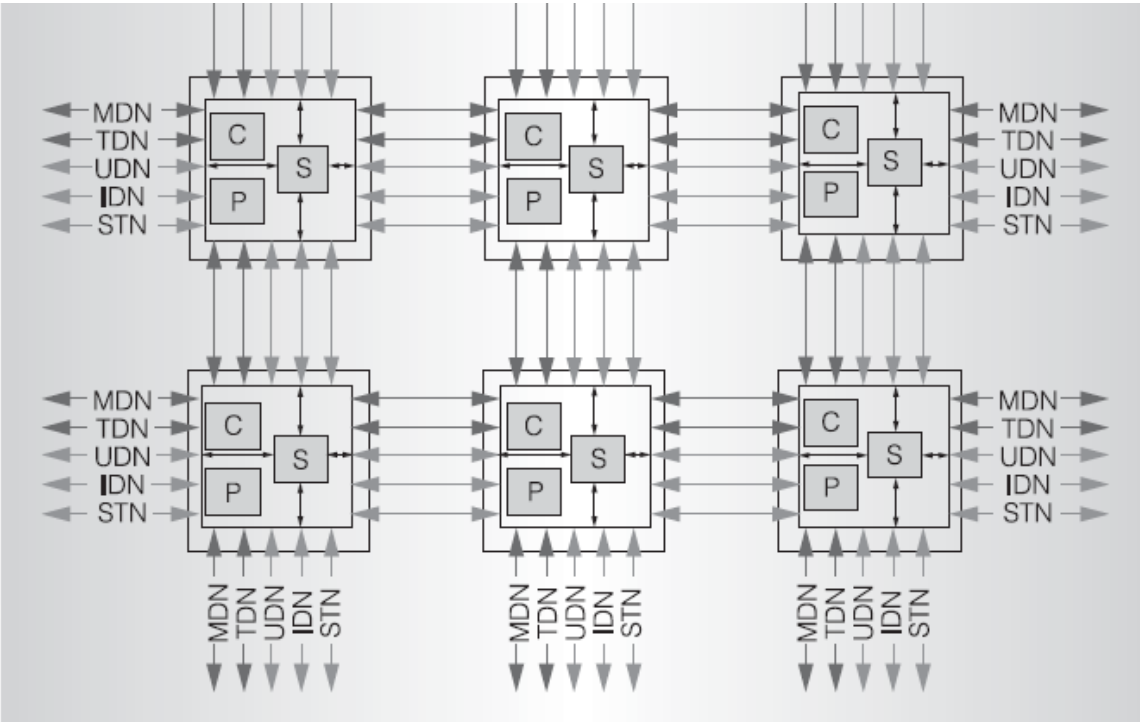
- 64 cores at 750 MHz
- Memory BW 25 GB/s
- 240 GB/s bis. Bw



TILE64 Networks

[Wentzlaff07]





STN – Static network
TDN – Tile Dynamic network
UDN – User Dynamic network
MDN – Memory Dynamic network
IDN – I/O Dynamic network

32 bit channels on all networks

Wormhole, dimension-order routed

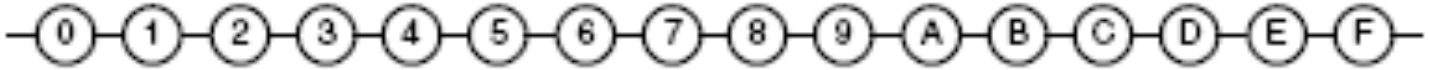
5-port routers with credit-based flow-control

STN – Scalar operand network
TDN and MDN implement the memory sub-system
UDN/IDN – Directly accessible by processor ALU (message-based, variable length)

Improving Tori - Express cubes

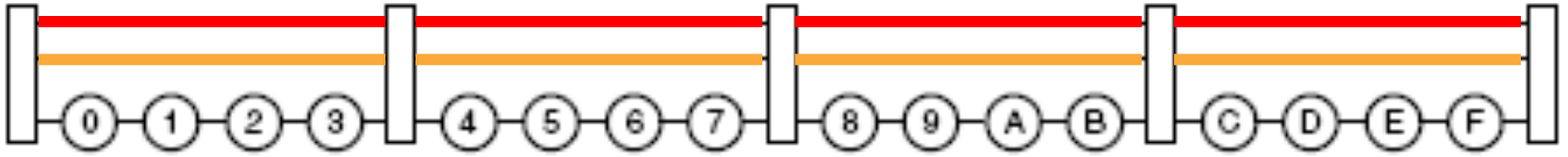
- Increase bisection bandwidth, reduce latency
  - Add expressways - long “express” channels

One dimension of 16-ary express cube with 4-hop express channels



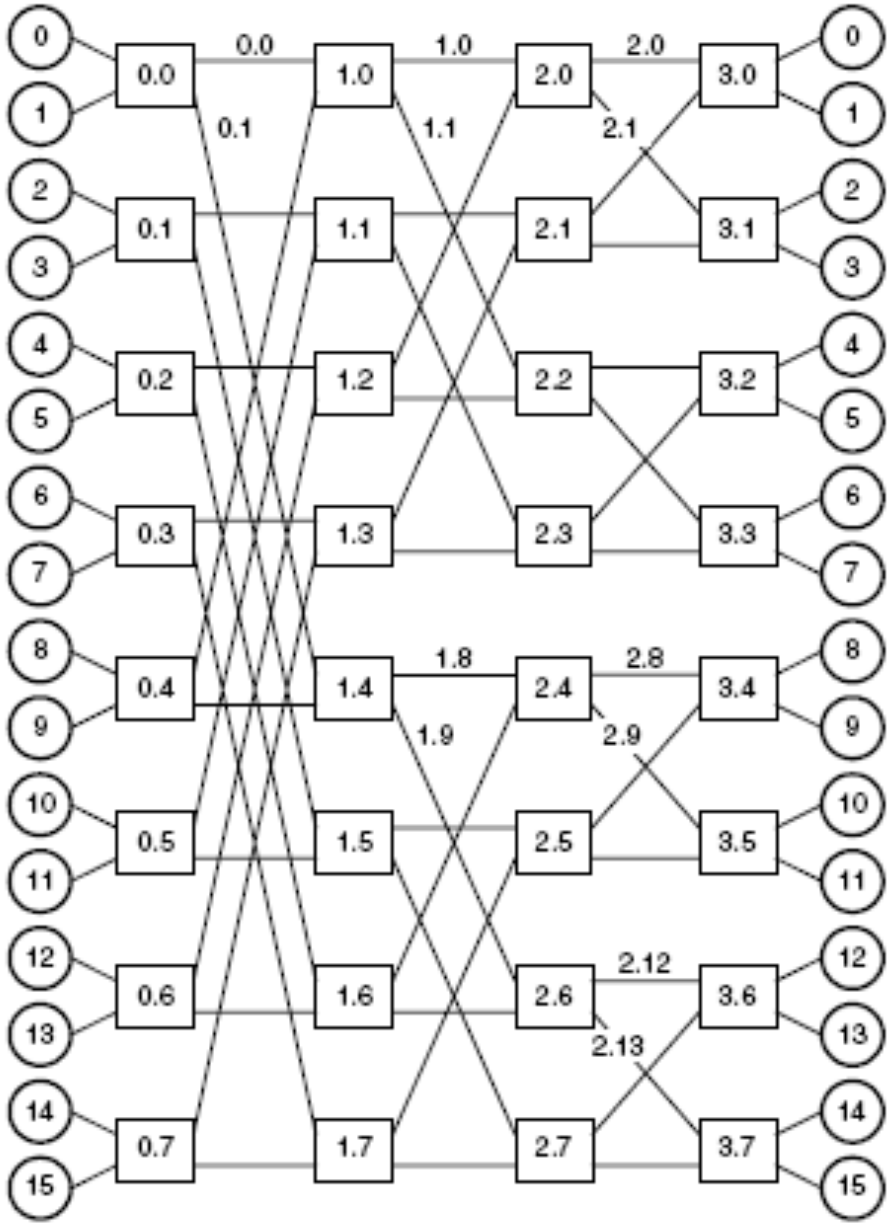
One dimension of 16-ary express cube with 4-hop express channels

Add extra channels to diversify and/or increase bisection



Buterflies

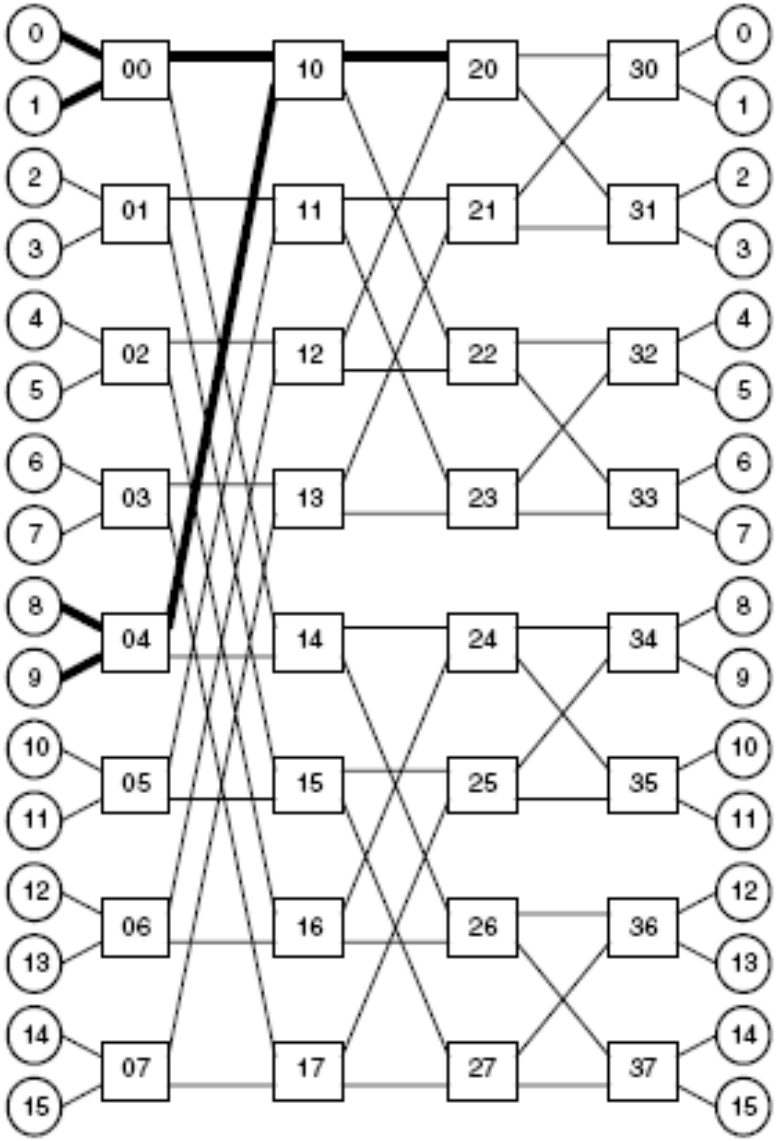
- N-ary, K-fly
  - N nodes per switch
  - K stages
- Example
  - 2-ary 4 fly



[Dally04]

Path diversity problem

- Butterflies have no path diversity
- Bad performance for some traffic patterns
  - e.g. shuffle permutation
- Wide spread in BW
- Inherently blocking
- Fixed in Clos topologies

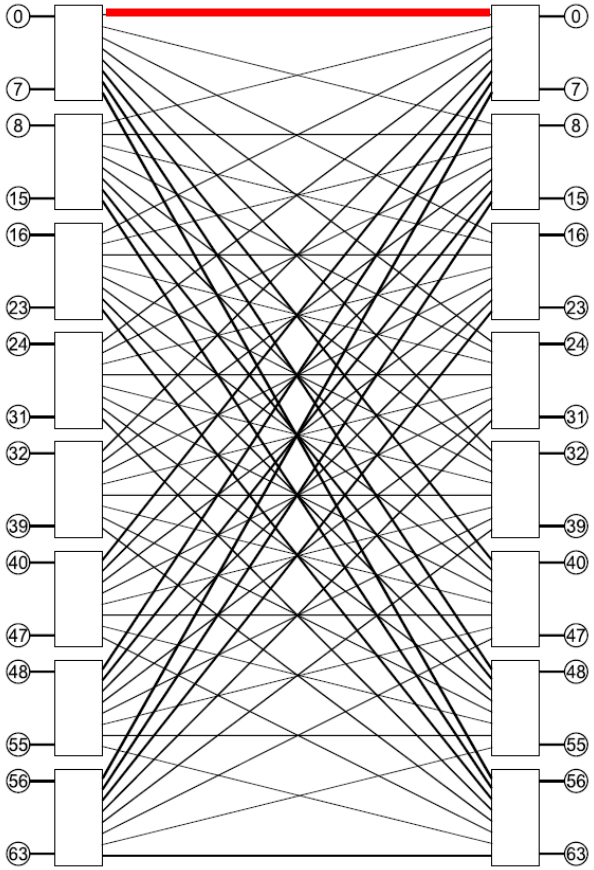


[Dally04]

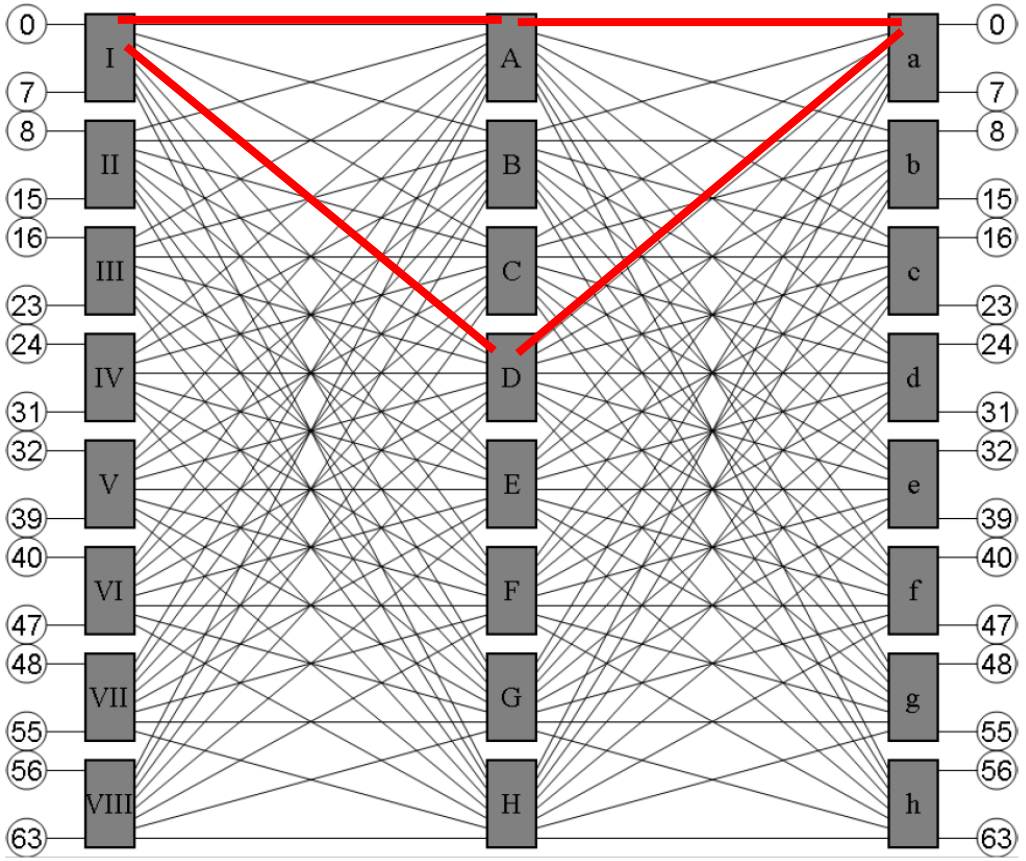
Clos networks

[Clos53]

8-ary 2-fly Butterfly

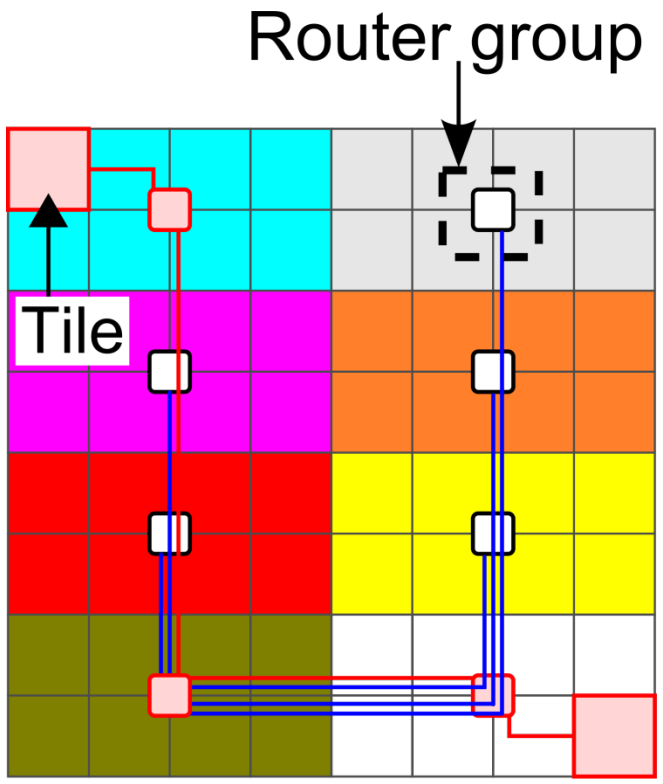


8-ary 3-fly Clos



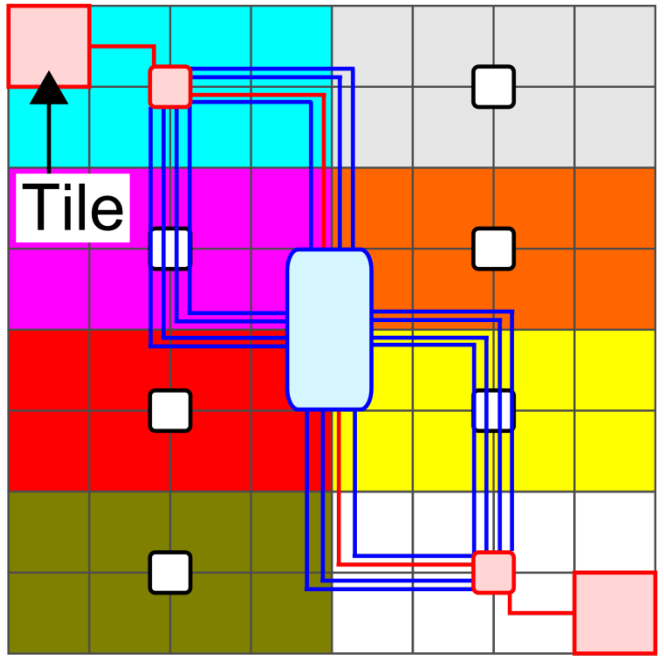
- Redundant paths – more uniform throughput

Logical to Physical Mapping



Three 8 x 8 Routers
(I-VIII, a-h, A-H)

8-ary 3-stage Clos

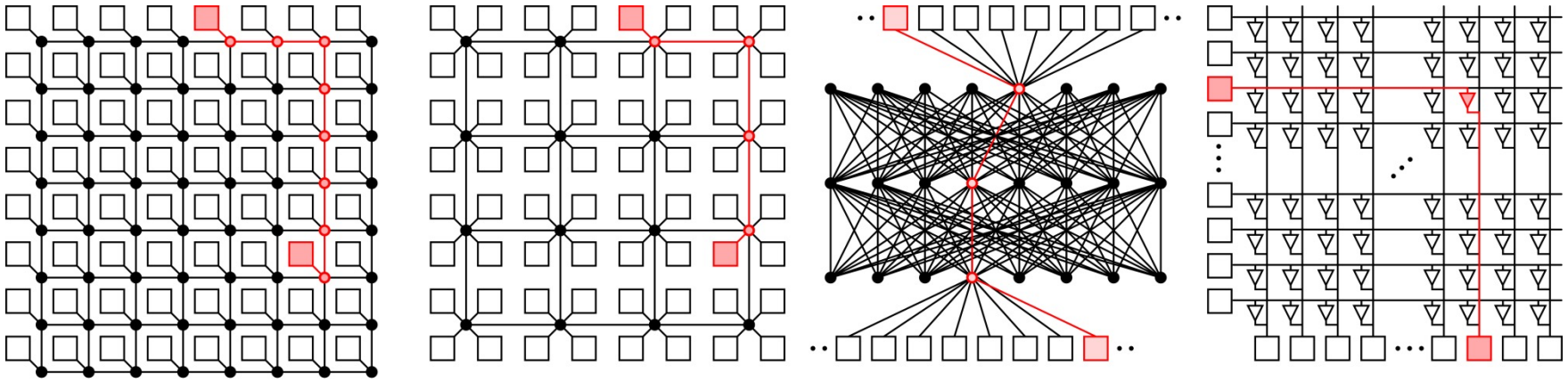


Two 8 x 8 Routers (I-VIII,a-h)
Eight 8 x 8 Routers
(middle stage A-H)

- Same topology – different physical mapping

Topology comparison

[Joshi10]



Mesh CMesh Clos Crossbar

	Channels				Routers		Latency					
Topology	N_C	b_C	N_{BC}	$N_{BC} \cdot b_C$	N_R	radix	H	T_R	T_C	T_{TC}	T_S	T_0
Mesh	224	256	16	4,096	64	5x5	2-15	2	1	0	2	7-46
CMesh	48	512	8	4,096	16	8x8	1-7	2	2	0	1	3-25
Clos	128	128	64	8,192	24	8x8	3	2	2-10	0-1	4	14-32
Crossbar	*64	*128	*64	8,192	1	64x64	1	10	n/a	0	4	14

Table 1: Comparison of network parameters – Networks sized to support 128 bits/cycle per tile under uniform random traffic. N_c = number of channels, b_C = bits/channel, N_{BC} = number of bisection channels, N_R = number of routers, H = number of routers along data paths, T_R = router latency, T_C = channel latency, T_{TC} = latency from tile to first router, T_S = serialization latency, T_0 = zero load latency. *Crossbar "channels" are the shared crossbar buses.

Lecture Roadmap

- Networking Basics
  - Topologies
  - Routing
  - Flow-Control
- Building Blocks
- Evaluation

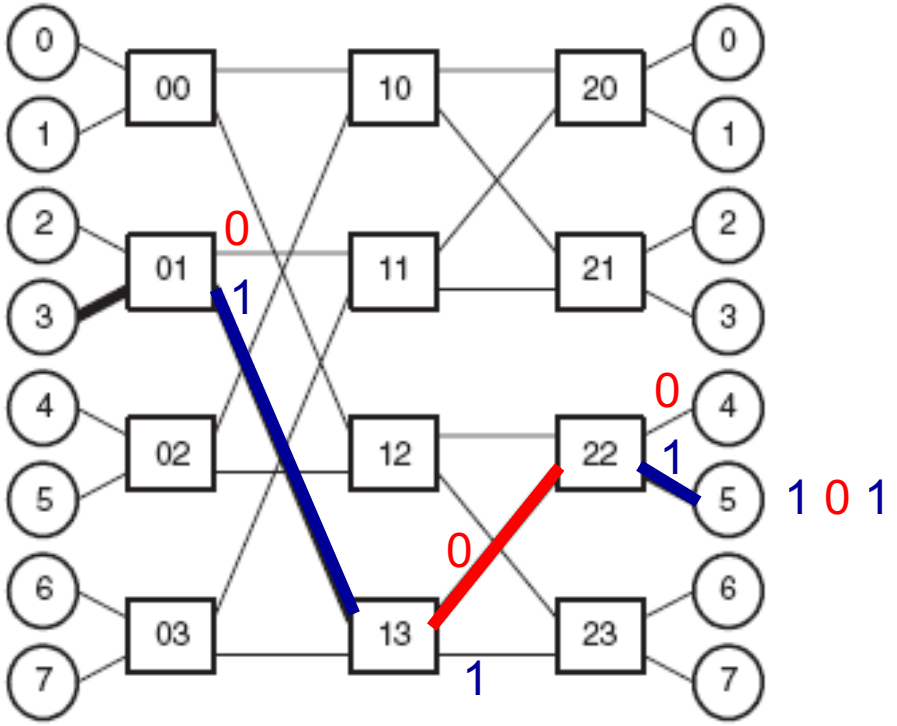
Routing Algorithms

- **Deterministic routing algorithms**
  - Always same path between x and y
    - Poor load balancing (ignore inherent path diversity)
    - Quite common in practice
      - Easy to implement and make deadlock-free.
- **Oblivious algorithms**
  - Choose a route w/o network's present state
    - E.g. random middle-node in Clos
- **Adaptive algorithms**
  - Use network's state information in routing
    - Length of queues, historical channel load, etc

Deterministic Routing

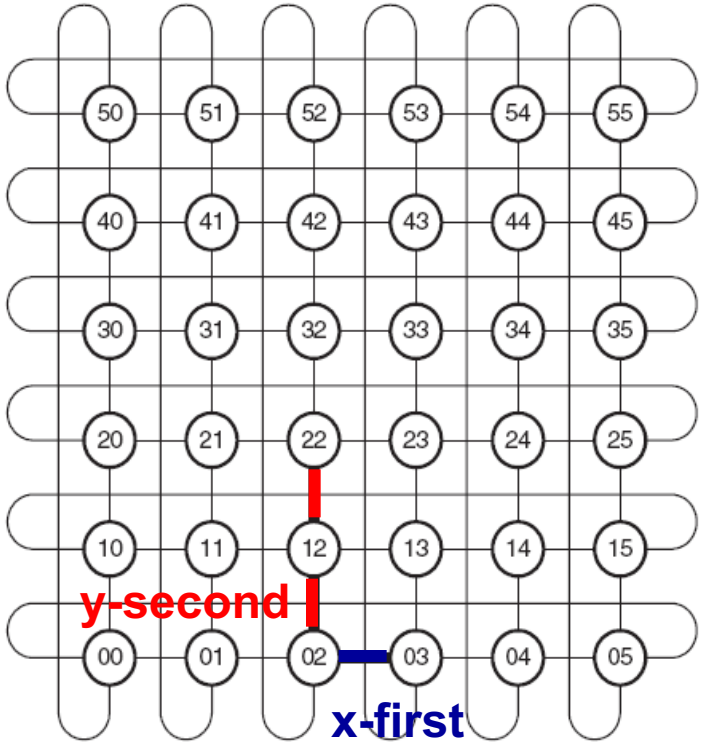
[Dally04]

2-ary 3-fly



Destination-tag Butterflies

6-ary 2-cube

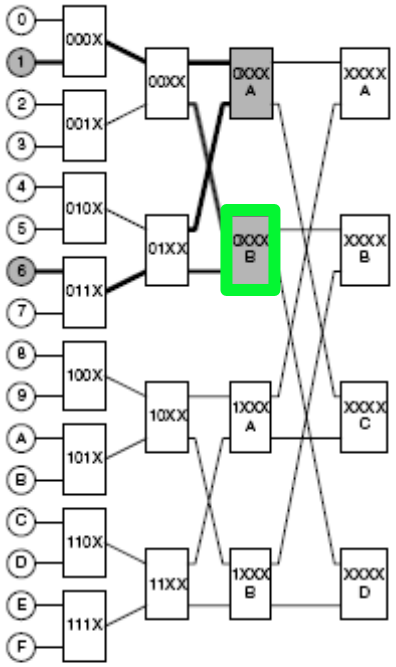


Dimension-order Tori

Oblivious Routing

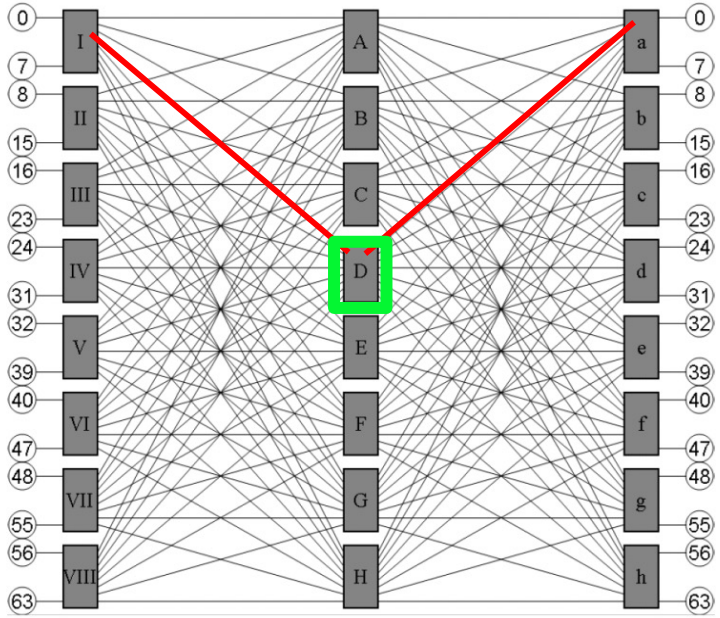
- Valiant's algorithm (Randomized Routing) [Dally04]

Folded Clos (Fat Tree)



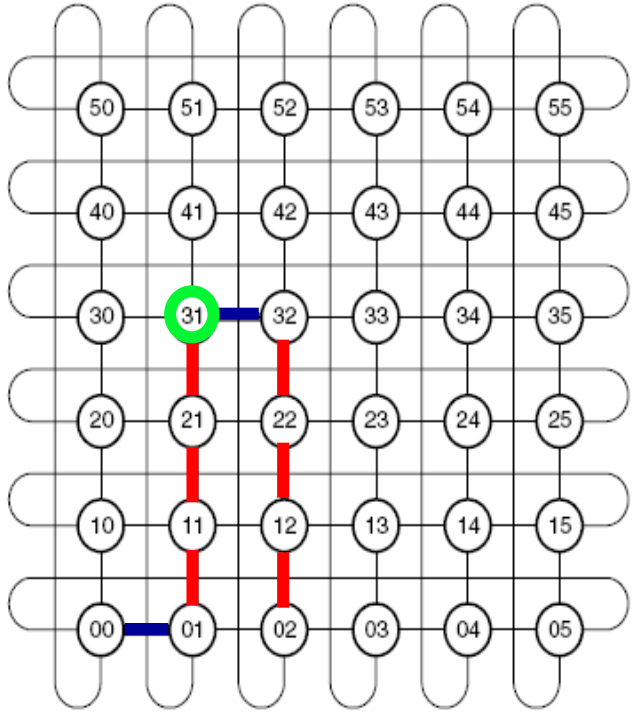
Randomly select
nearest common ancestor switch

8-ary 3-fly Clos



Randomly select middle switch

6-ary 2-cube

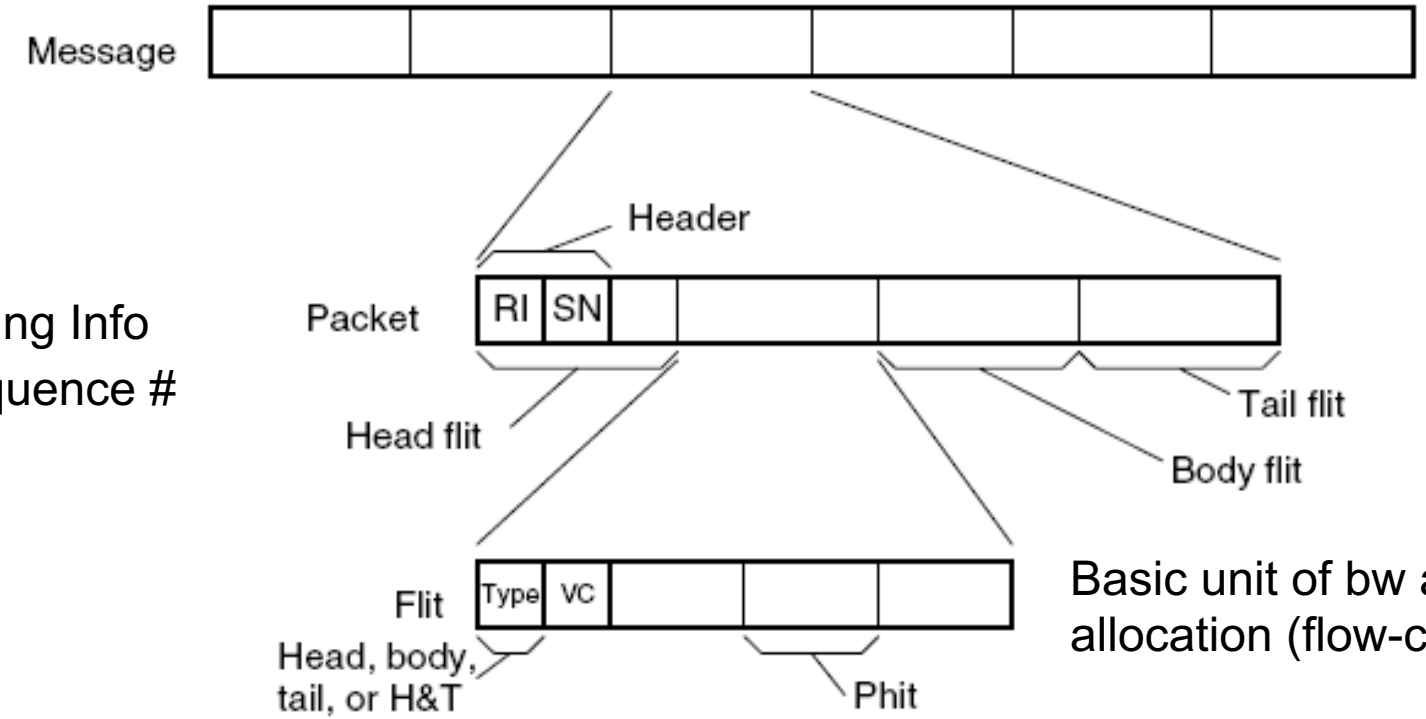


Randomly select middle node
Dimension-order to/from node

Lecture Roadmap

- Networking Basics
  - Topologies
  - Routing
  - Flow-Control
- Building Blocks
- Evaluation

Message definitions



RI - Routing Info
SN – Sequence #

1kb (typ)

Basic unit of bw and storage allocation (flow-control)

64b (typ)

Sent across channel in a clock cycle

8b (typ)

- Basic trade-off
  - Minimize overheads (large size)
  - Efficient use of resources (small size)

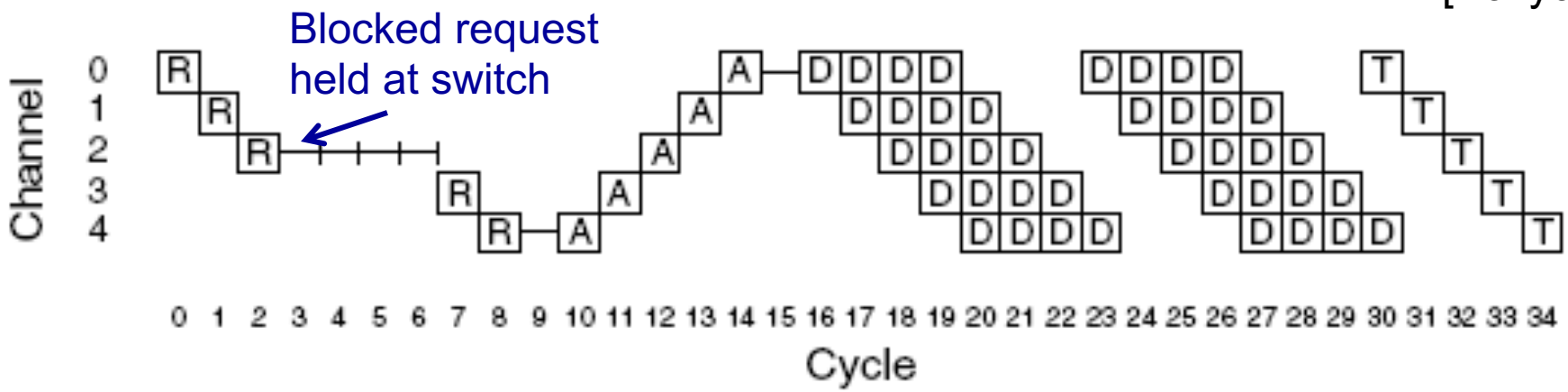
[Dally&Towles: Principles and Practices of Interconnection Networks - Dally04]

Flow Control

- Bufferless flow-control (Circuit Switching)
- Buffered flow-control (Packet Switching)
  - Packet-based (store&forward, cut-through)
  - Flit-based (wormhole, virtual channels)
- Buffer Management
  - Credit-based, on-off, flit-reservation

Circuit switching

[Dally04]



R - Request
Acquires channel state at each hop

A - Acknowledgment

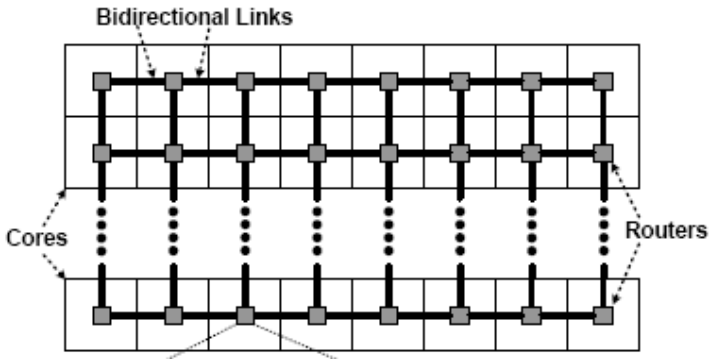
D – Data packets
e.g. Two, four-flit packets

T – Tail flit
Deallocate channels

- Pros
  - Simple to implement (simple routers, small buffers)
- Cons
  - High latency (R+A) and low throughput

Example - Pipelined Circuit Switching

[Anderson08]



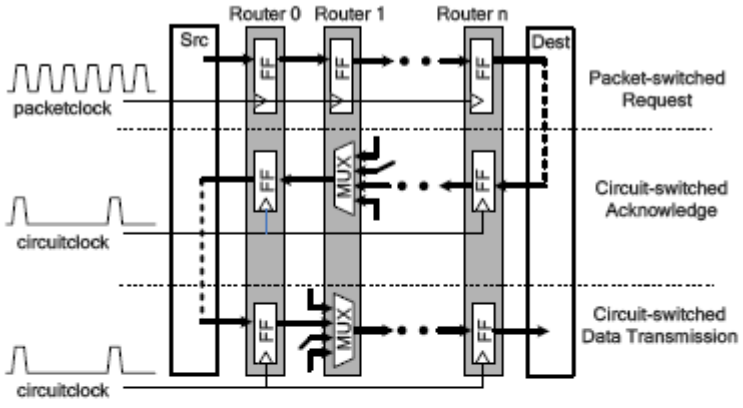
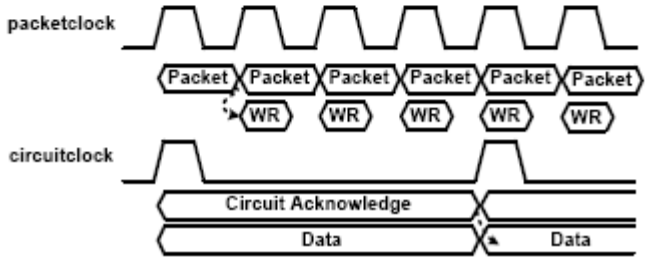
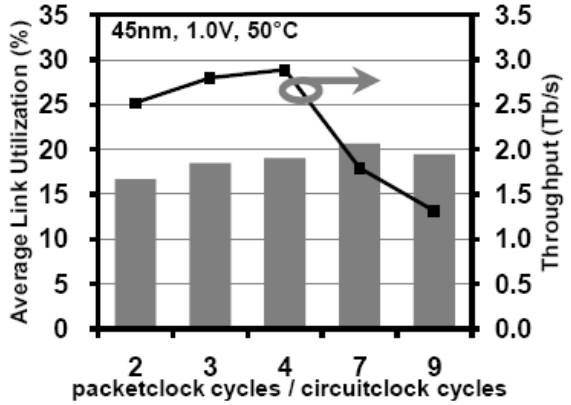
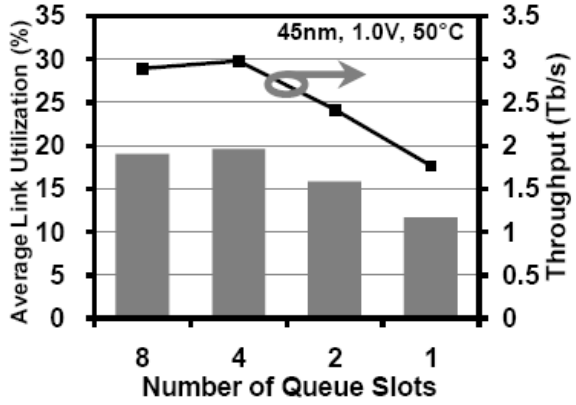
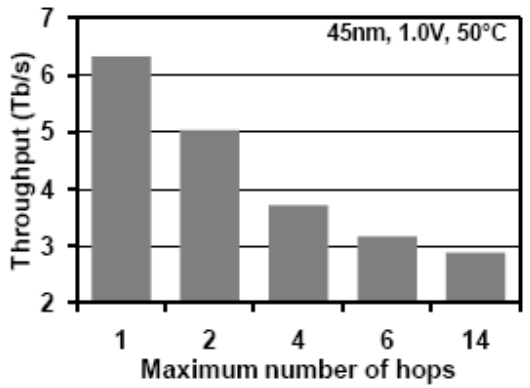


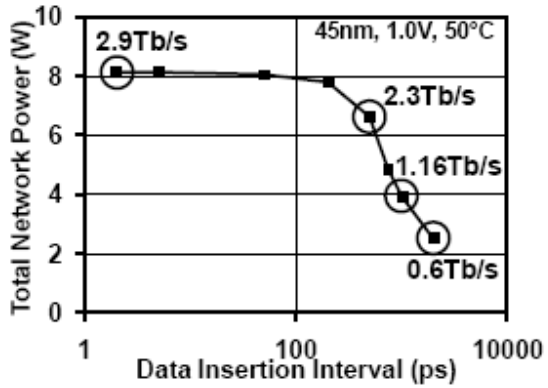
Figure 2. Circuit-switched pipeline and clocking.











64 core 2D mesh, 125 mW/router

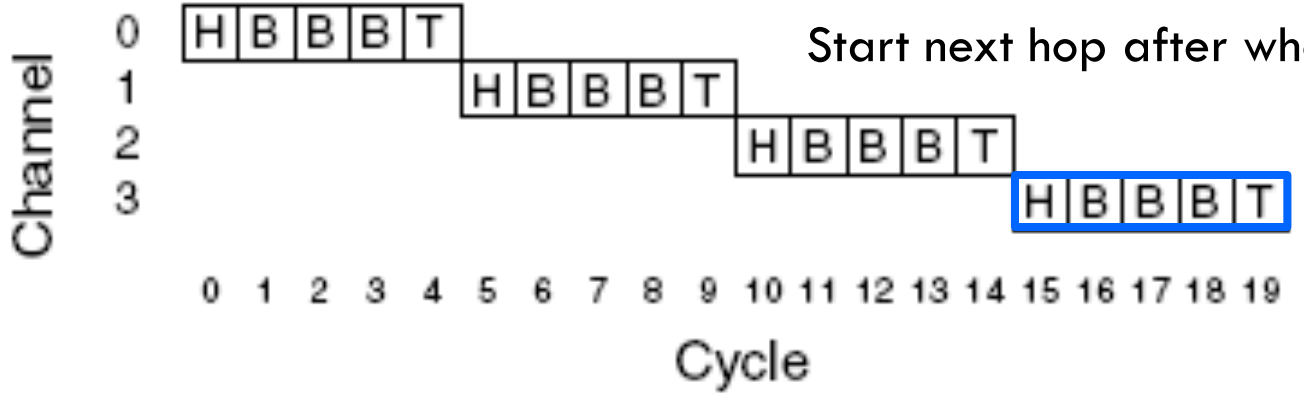
Network efficiency 3 pJ/bit

Packet-buffered Flow Control

Buffer and channel allocated to the whole packet

[Dally04]

- Store-and-forward



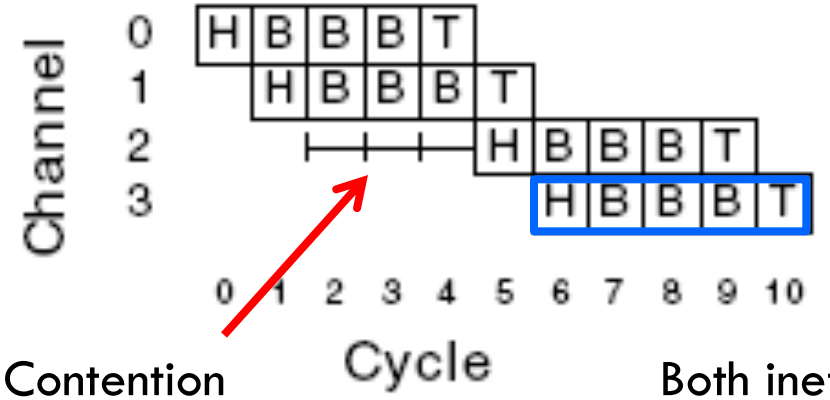
Start next hop after whole packet received

5-flit packet

$$T_0 = H\left(t_r + \frac{L}{b}\right)$$

- Cut-through

Start next hop after head flit received



5-flit packet

$$T_0 = Ht_r + \frac{L}{b}$$

Contention for channel 2

Both ineffective in use of buffer storage

Contention latency increased in channels

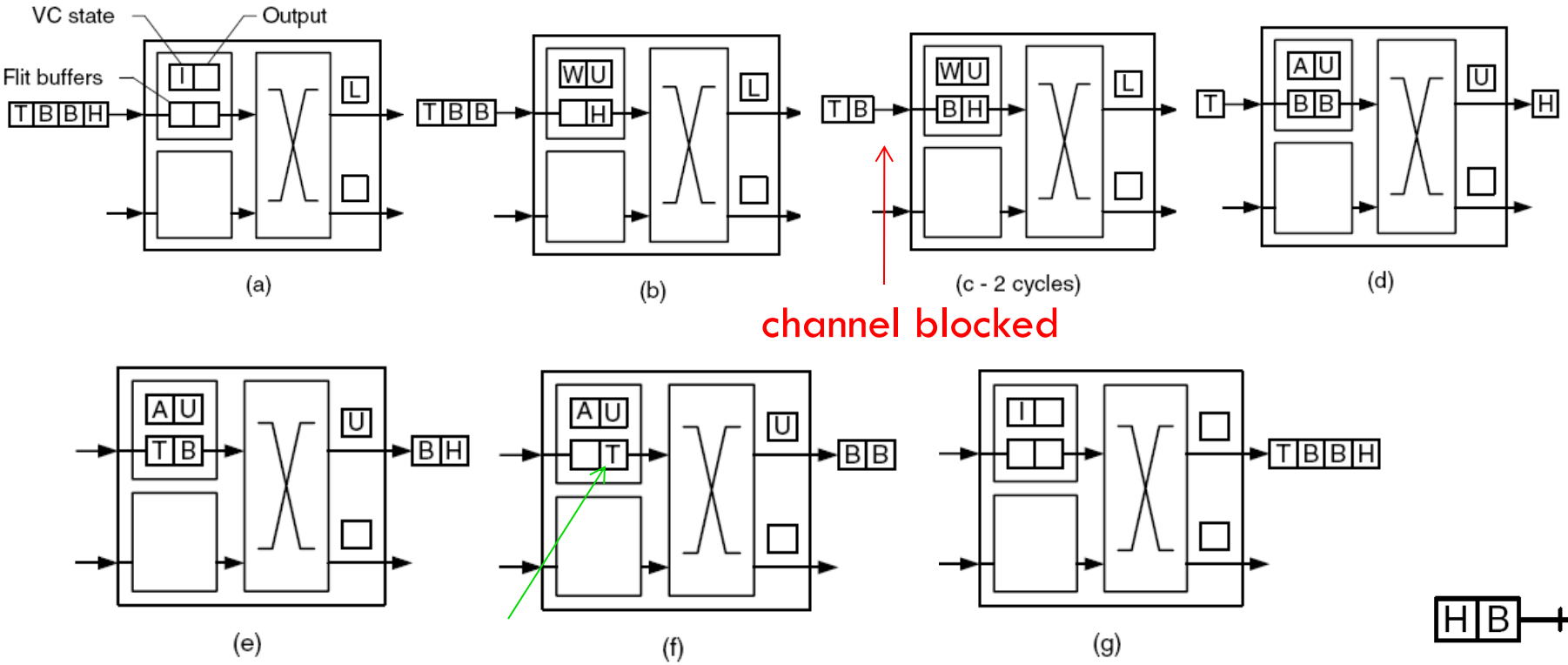
Flit-buffered Flow Control

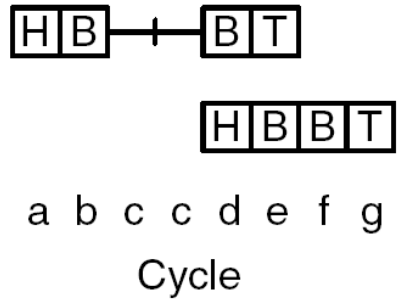
Buffer and channel allocated to flits

[Dally04]

- Wormhole

I – idle, W – waiting, A - allocated





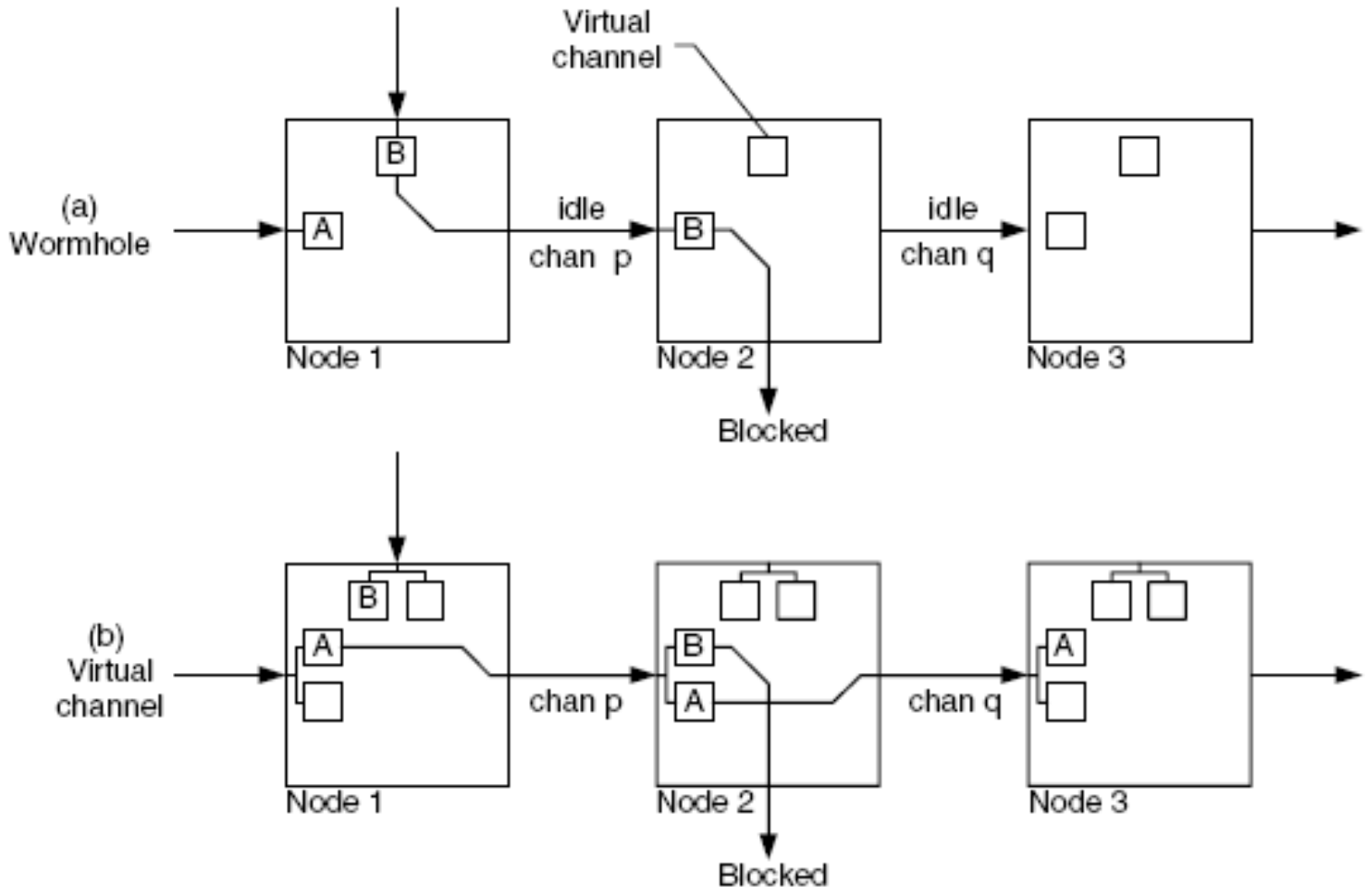
More efficient buffer usage than cut-through

But, may block a channel mid-packet

Flit-buffered Flow Control

- Wormhole vs. Virtual-Channel

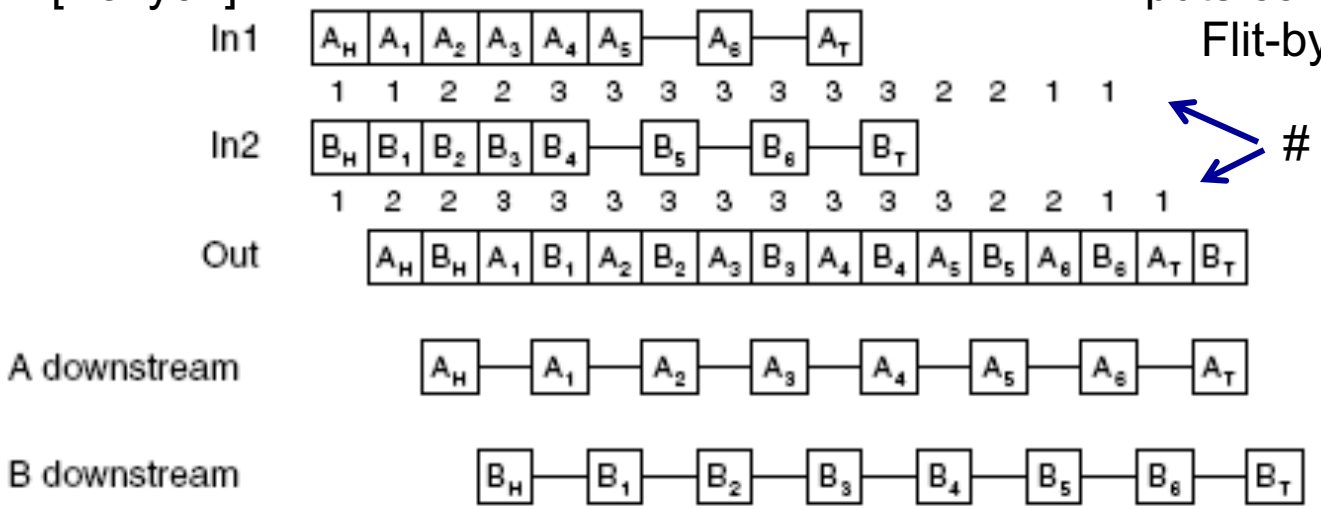
[Dally92]



[Dally04]

Virtual-channels – Bandwidth Allocation

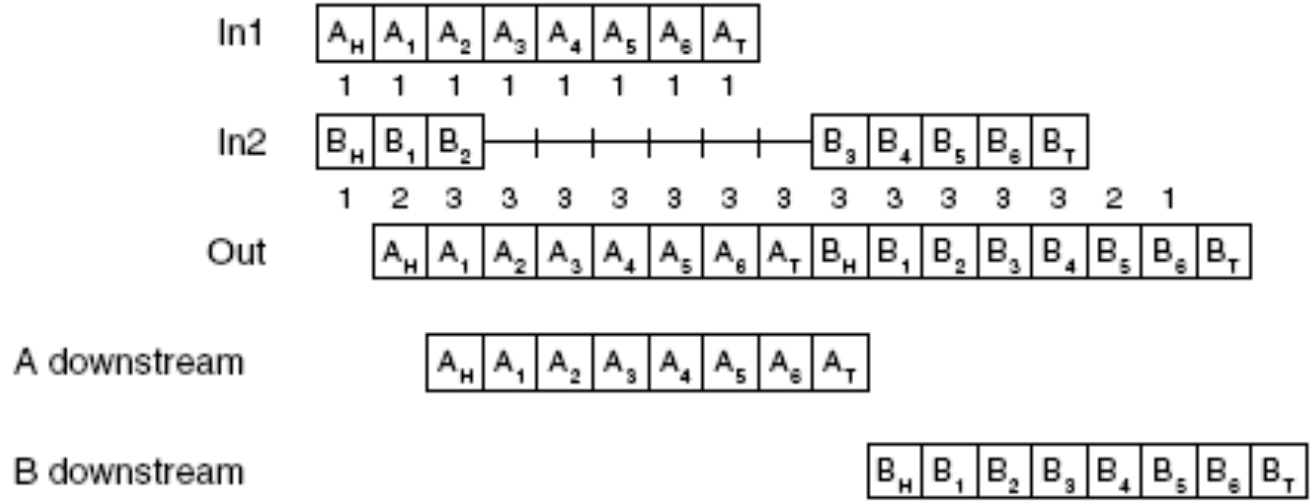
[Dally04]



Inputs compete for bandwidth Flit-by-Flit

flits in VC buffer (cap 3)

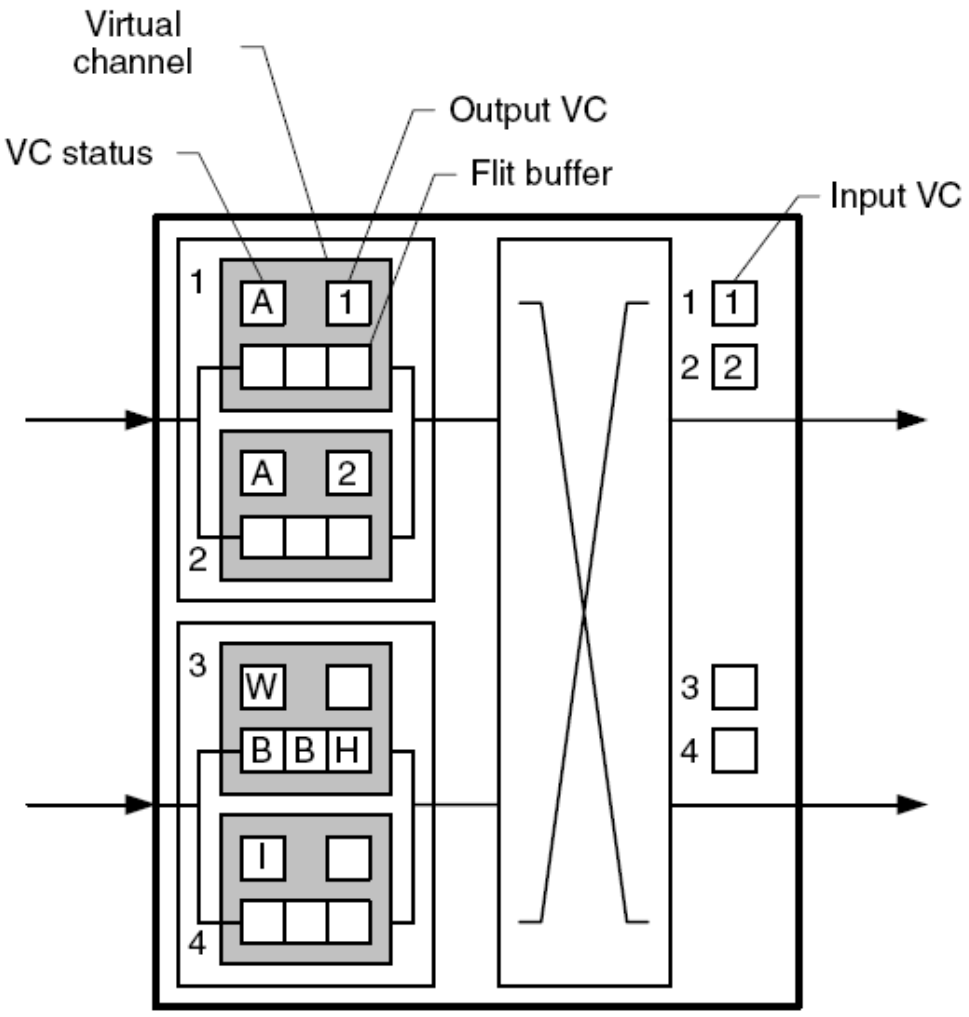
Fair Arbitration



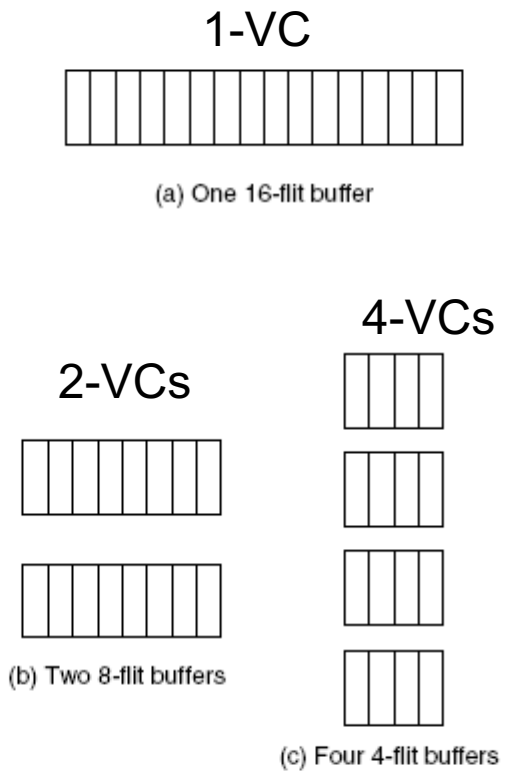
Winner-take-all Arbitration

Reduced latency
No throughput penalty

Virtual-channel Router



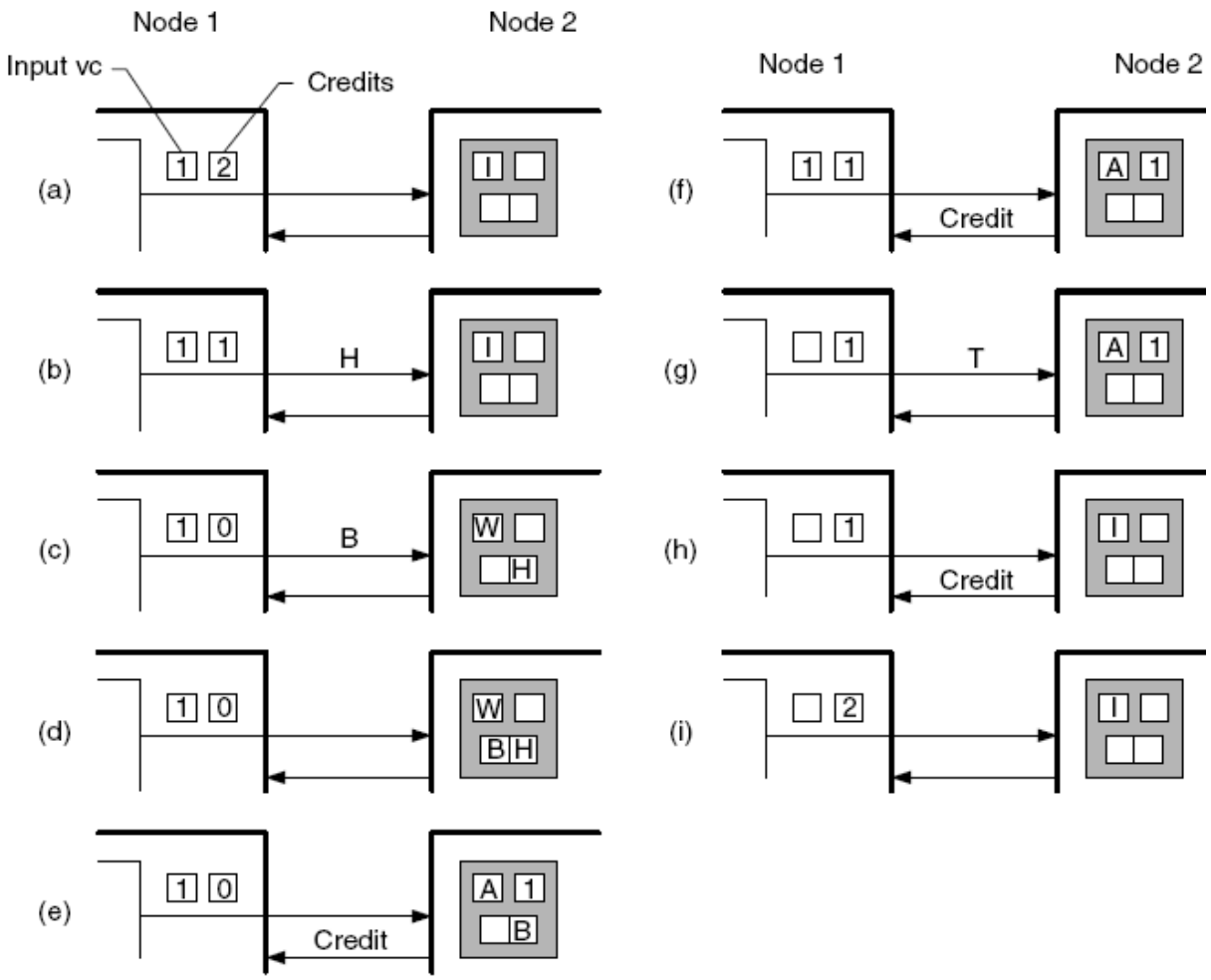
[Dally04]



Each channel only as deep as round-trip credit latency

More buffering, more virtual channels

Credit-based buffer management



[Dally04]

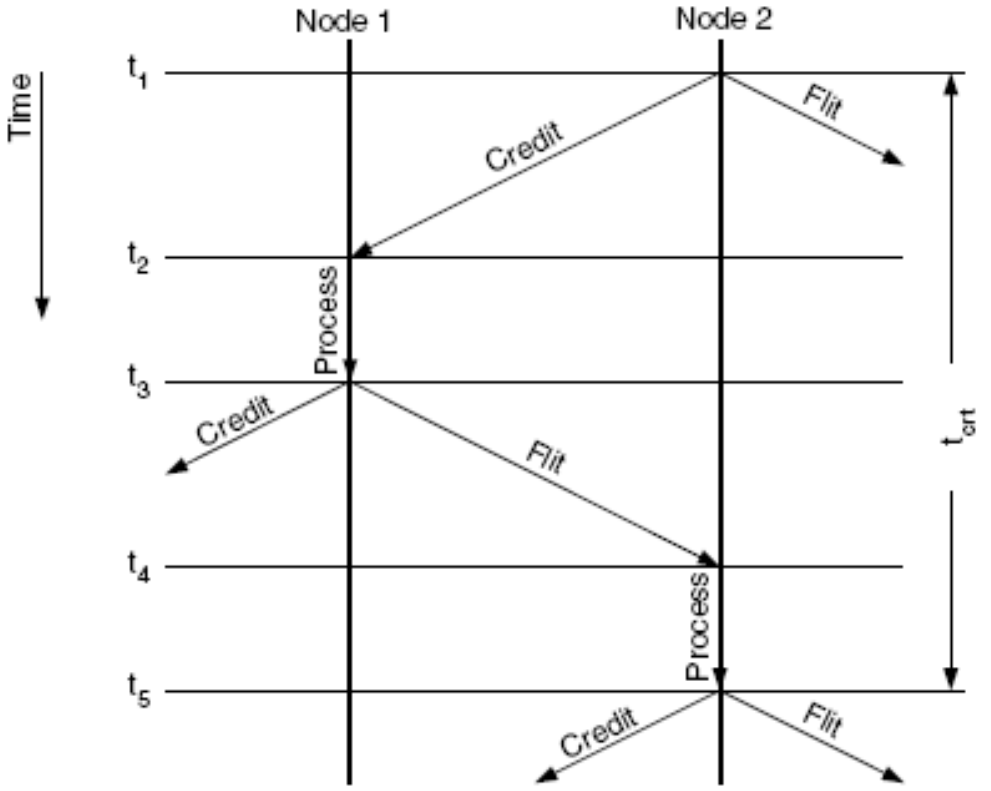
$$F \geq \frac{t_{crt} b}{L_f}$$

F - Flit buffer depth

L_f – Flit length

b – channel bandwidth

t_{crt} – credit round-trip delay

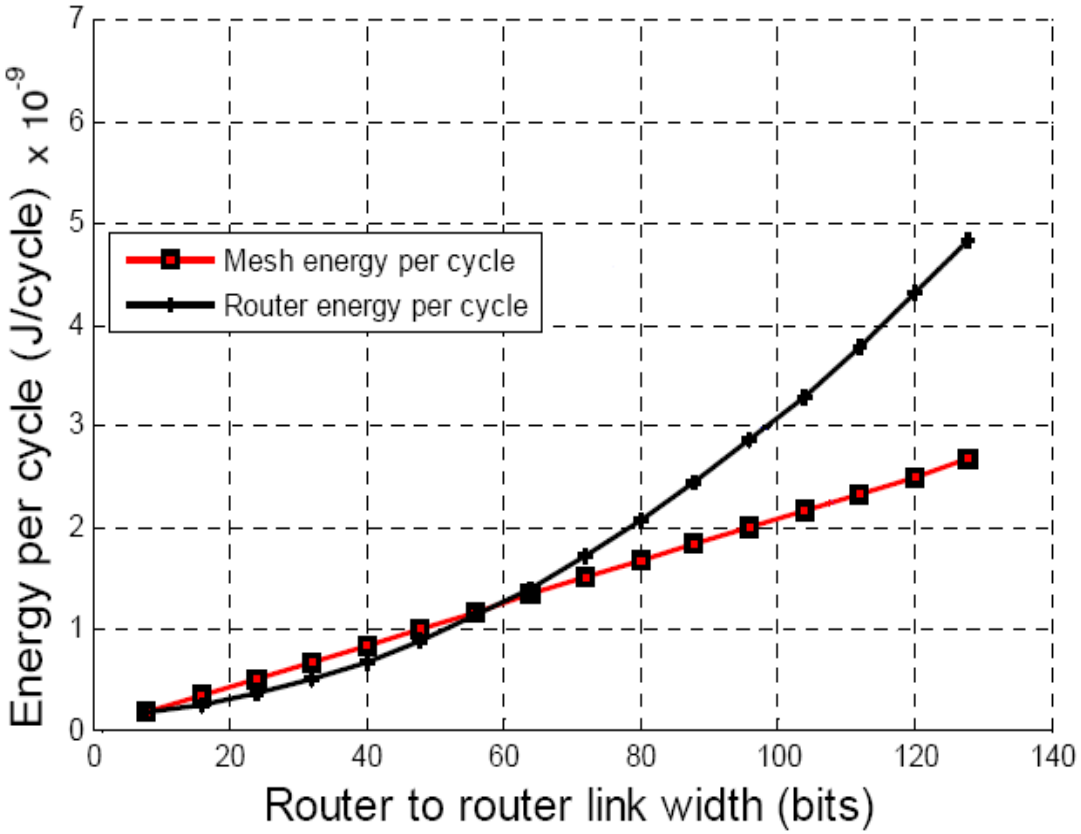


Lecture Roadmap

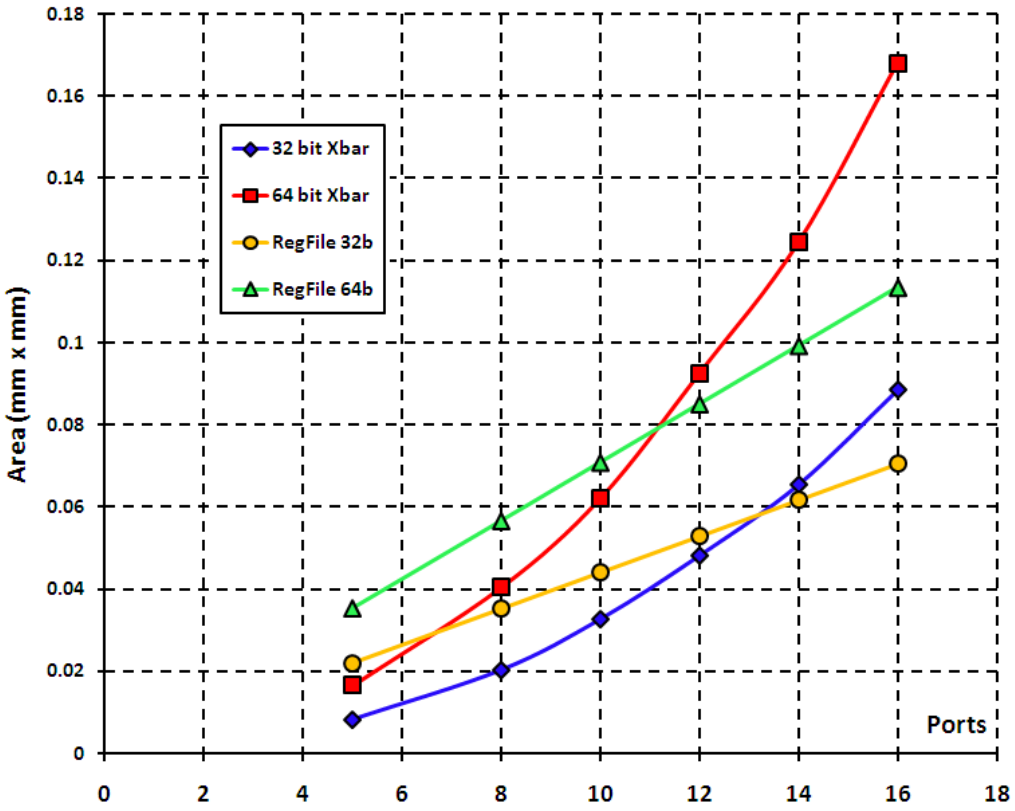
- Networking Basics
- Building Blocks
  - Channels
  - Routers
- Evaluation

Building block costs

Router vs. channel energy



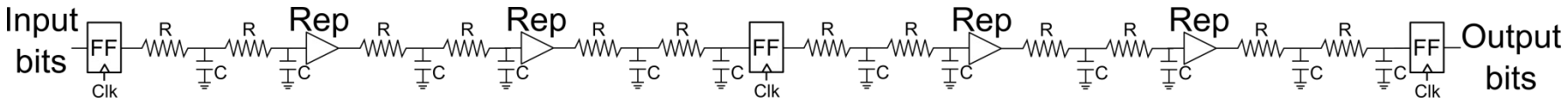
Router Area Breakdowns



- Simple routers and channels roughly balanced
- Narrower networks scale better

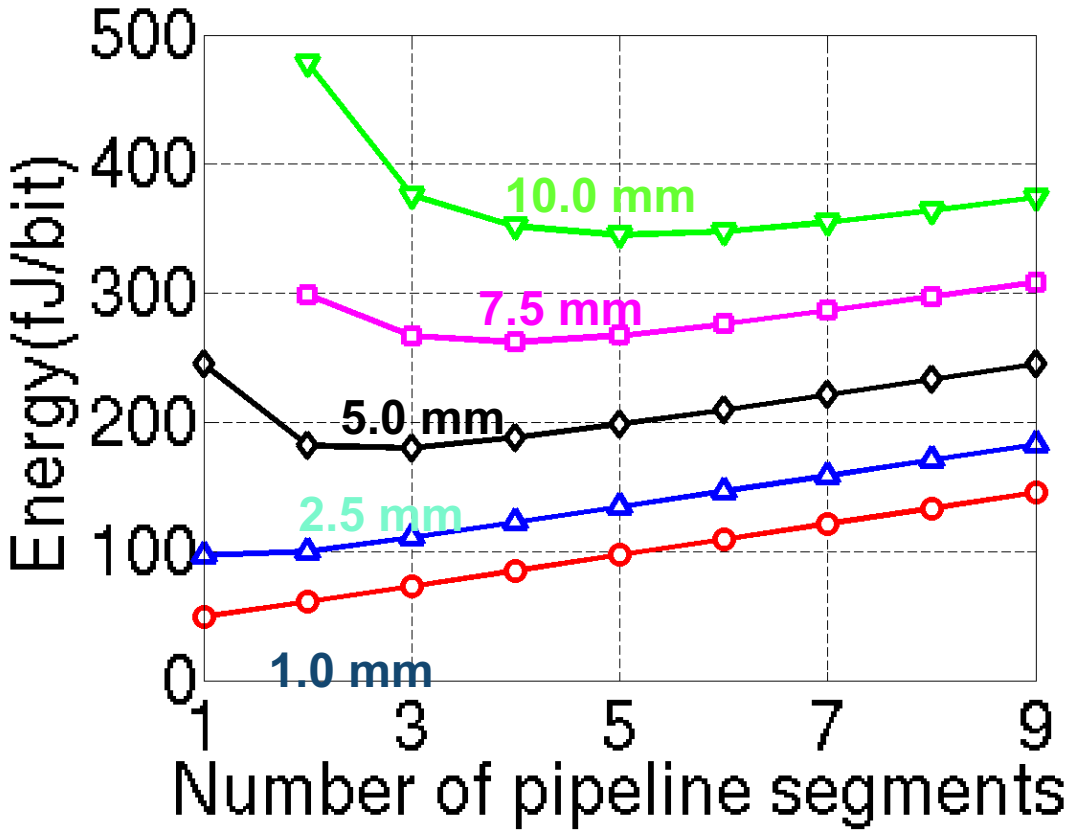
90nm technology

Channels: Electrical technology

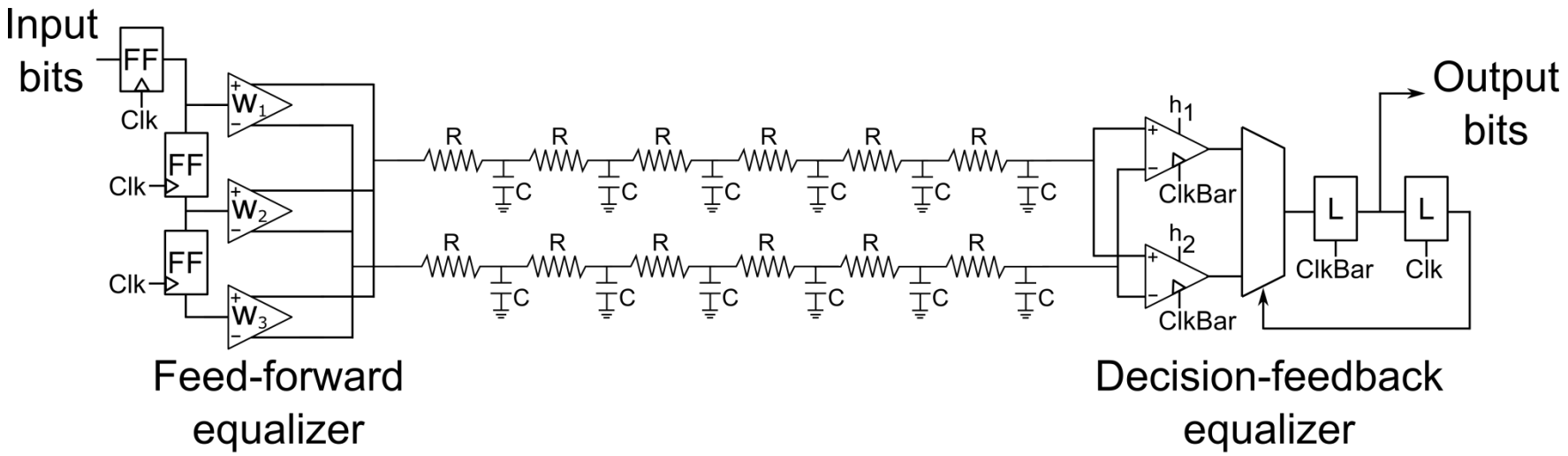


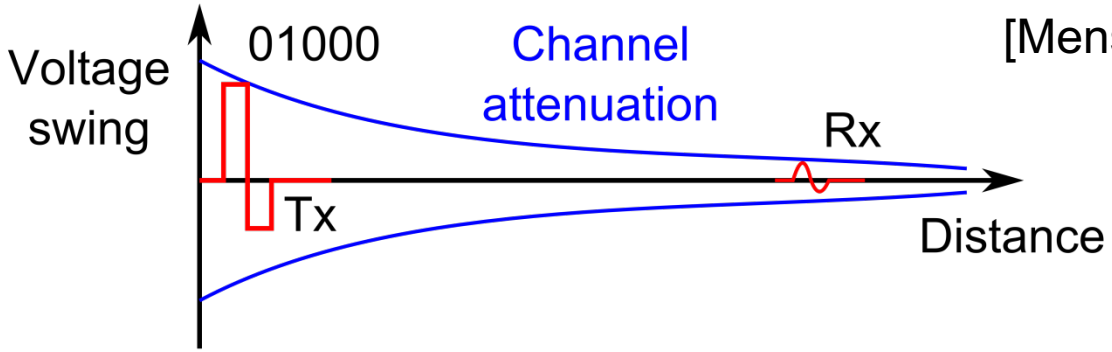
Repeater inserted pipelined wires

- Design constraints
  - 22 nm technology
  - 500 nm pitch
  - 5 GHz clock
- Design parameters
  - Wire width
  - Repeater size
  - Repeater spacing



Channels: Equalized interconnects

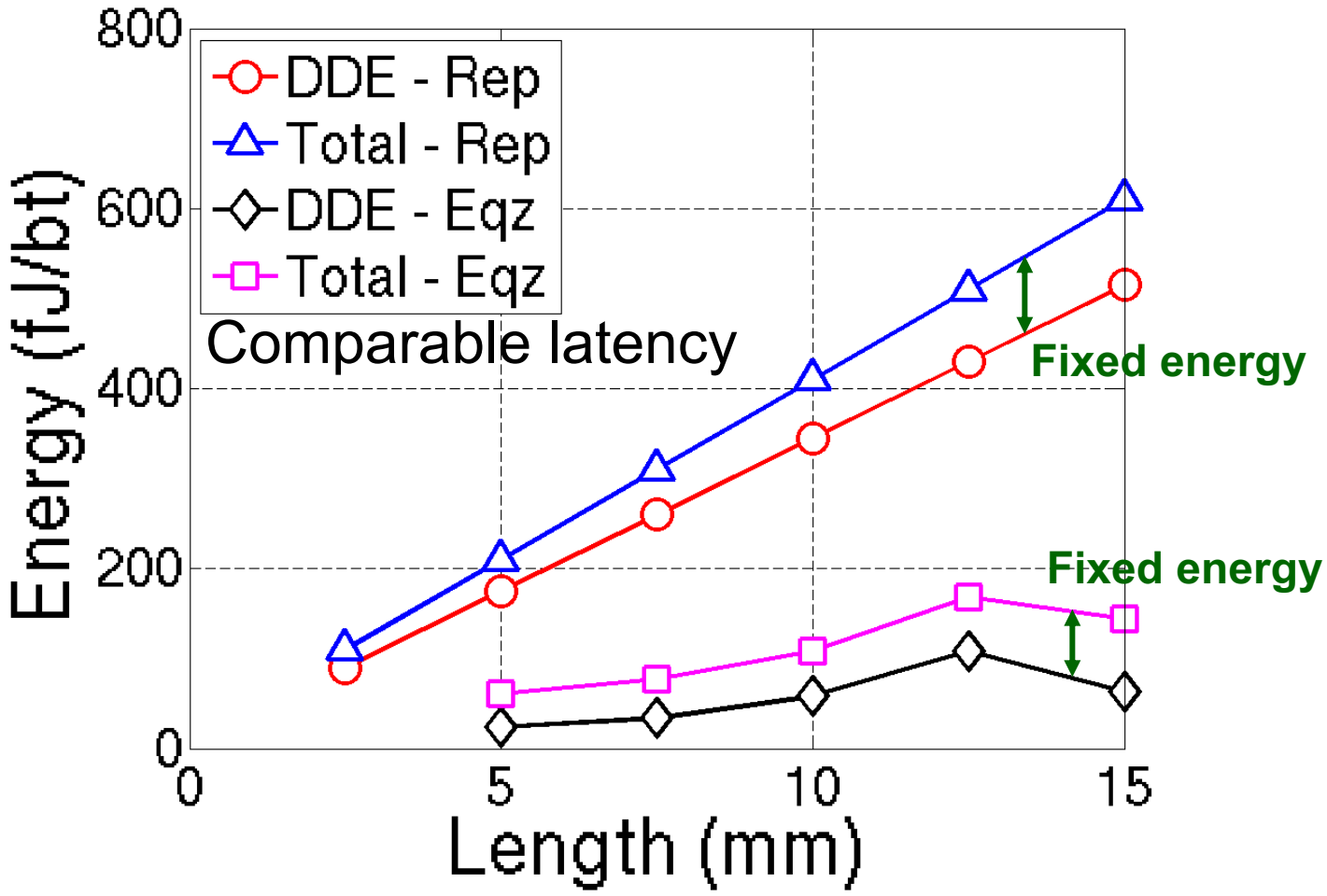




[Mensik07,Kim08,Kim09]

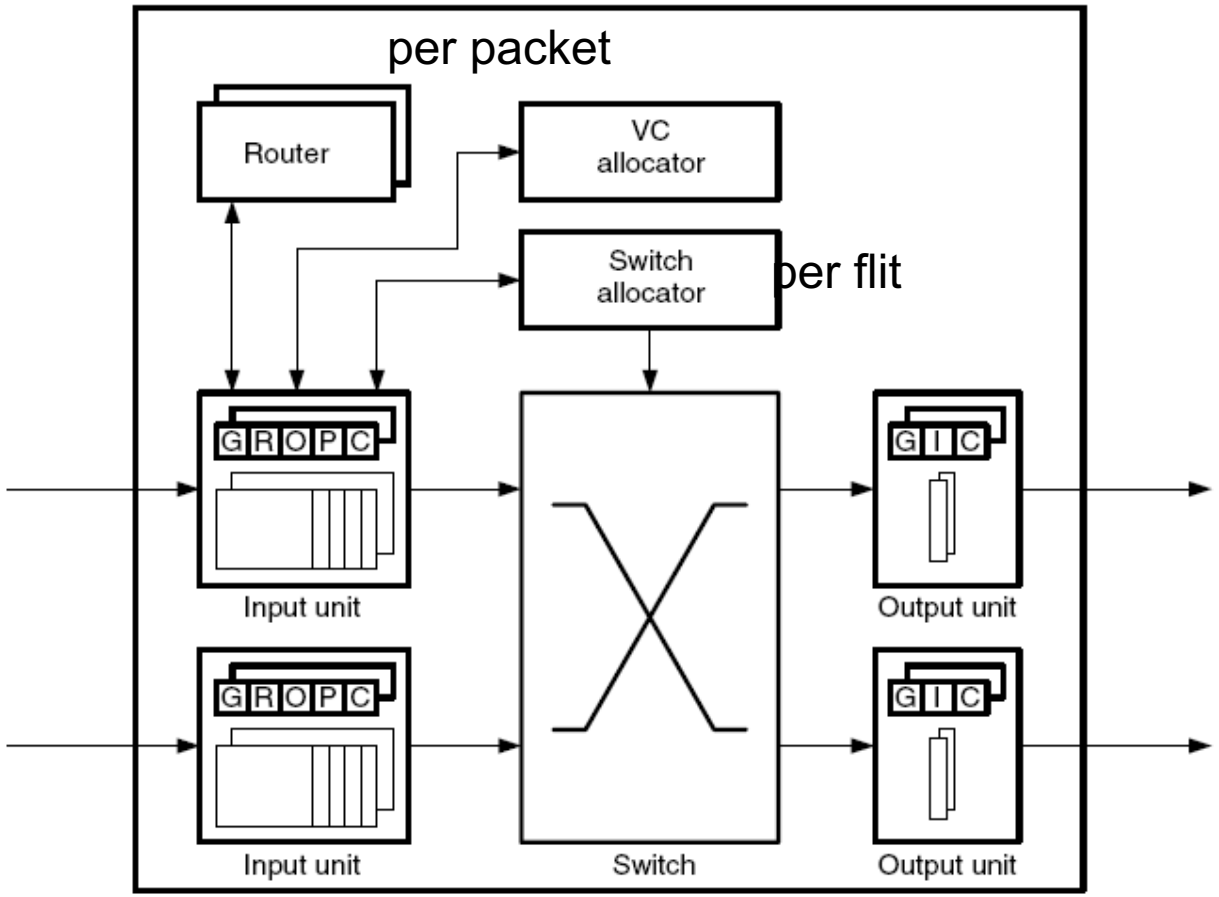
- FFE shapes transmitted pulse
- DFE cancels first trailing ISI tap
- Lower energy cost due to output voltage swing attenuation

Repeated interconnects vs Equalized interconnects



Data-dependent energy (DDE) is 4-10x lower for equalized interconnects, while fixed energy (FE) is comparable

Routers



Input VC state

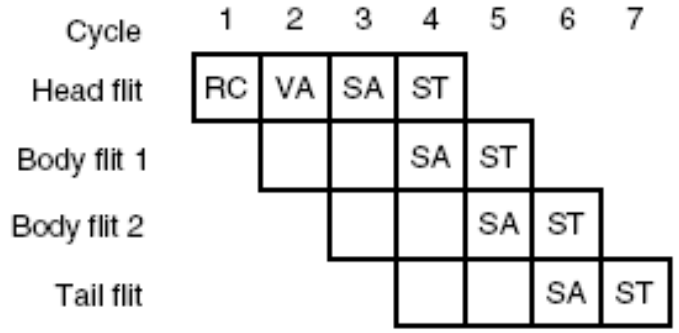
Field	*Name*
G	Global state
R	Route
O	Output VC
P	Pointers
C	Credit count

Output VC state

Field	*Name*
G	Global state
I	Input VC
C	Credit count

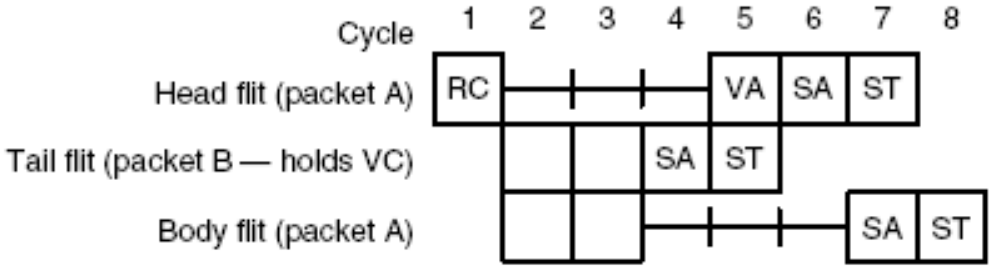
Router pipeline

- Pipelined routing of a packet



RC – route computation
VA – virtual channel allocation
SA – switch allocation
ST – switch traversal

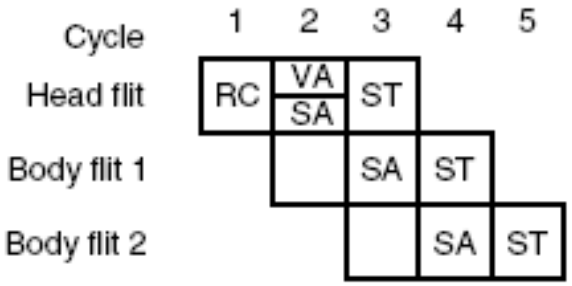
Pipeline stalls (virtual-channel allocation stall – output VC)

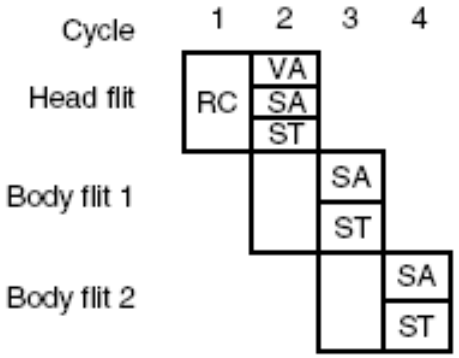


VC stall need not slow transmission over the input channel as long as there is sufficient buffer space (in this case, six flits) to hold the arriving head and body flits until they are able to begin switch traversal.

Speculation and Lookahead

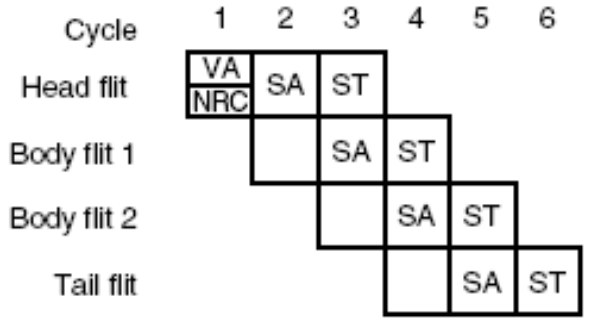
Speculative allocation

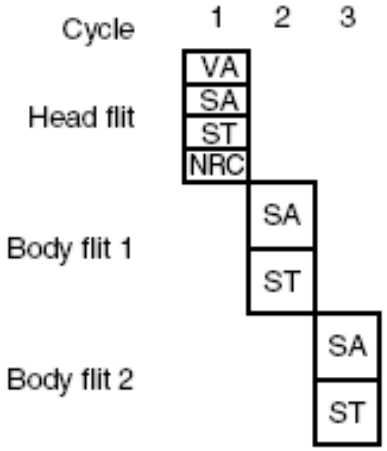




Lookahead routing

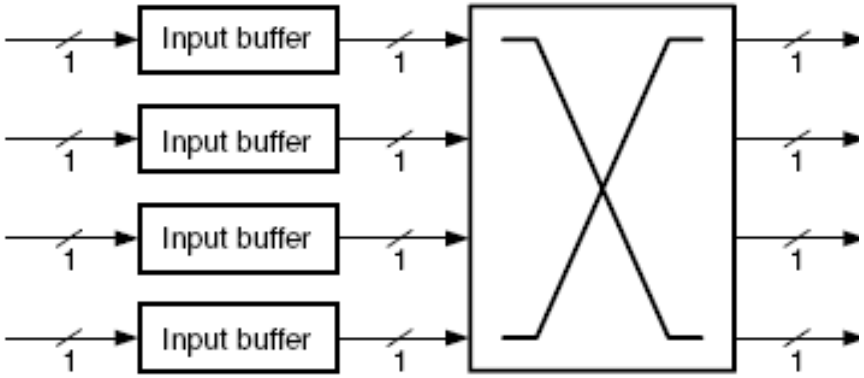
(pass routing for next hop in head flit)



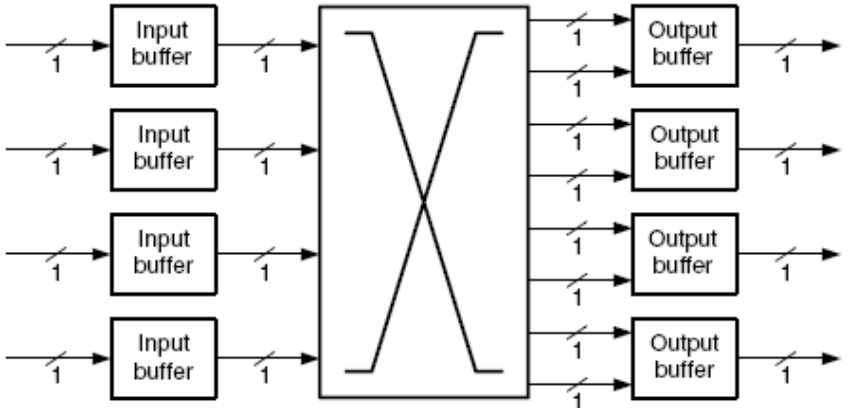


Crossbar switches

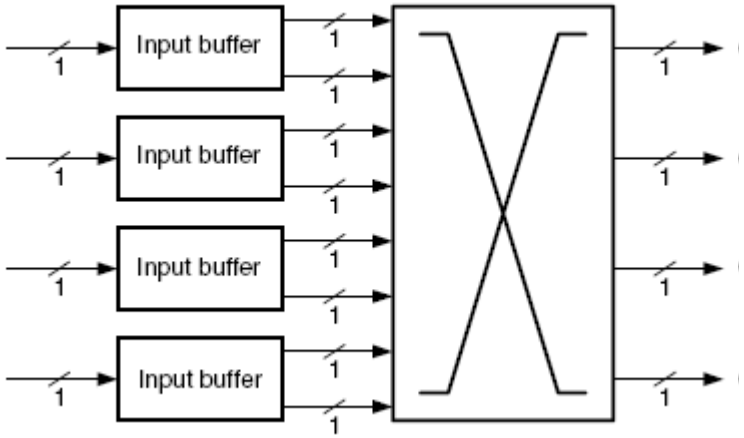
No Speedup – 68% capacity



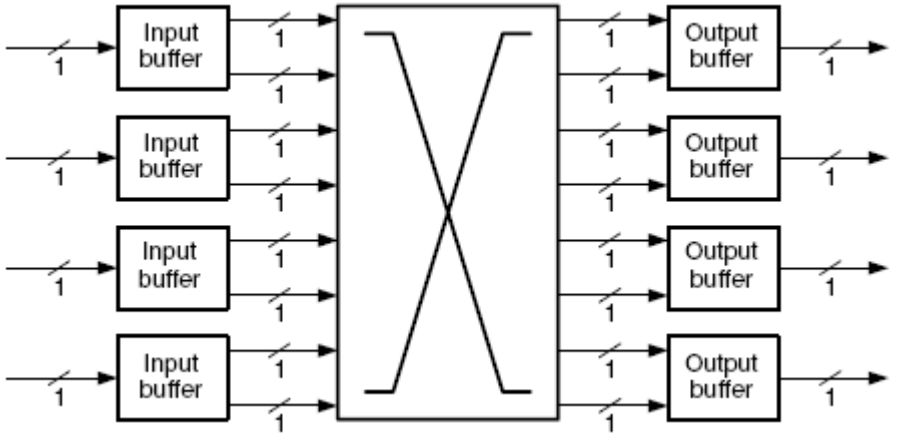
2x Output Speedup – 87% capacity



2x Input Speedup – 90% capacity



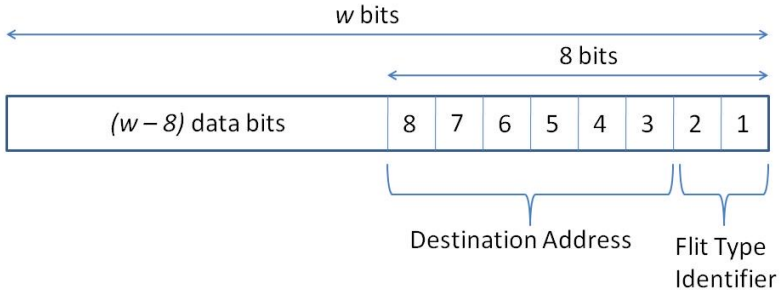
2x Input & Output Speedup – 137% capacity



$$\Theta = s_o \left(1 - \left(\frac{k-1}{k}\right)^{\frac{s_i k}{s_o}}\right)$$

Router design space exploration - Setup

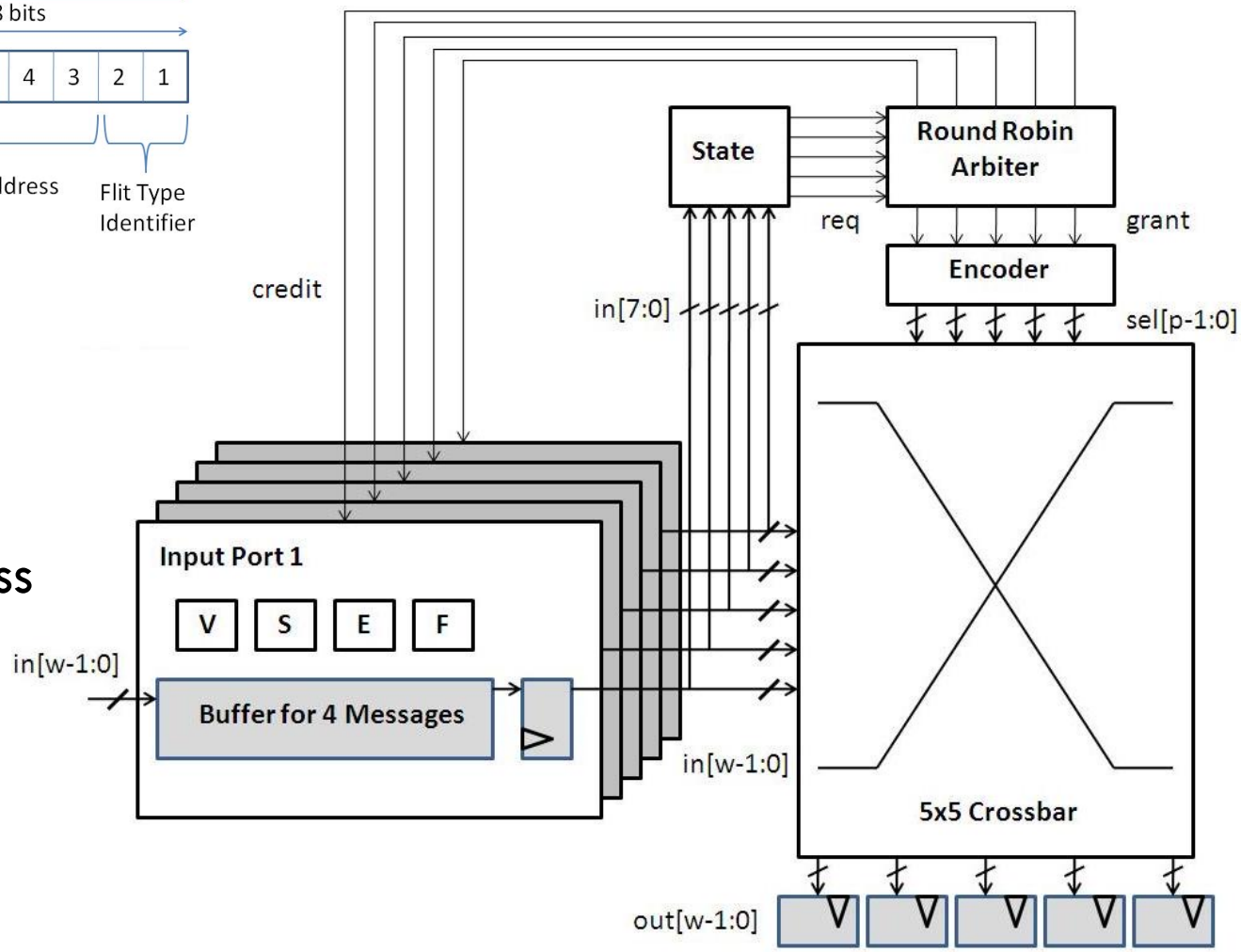
[Shamim09]



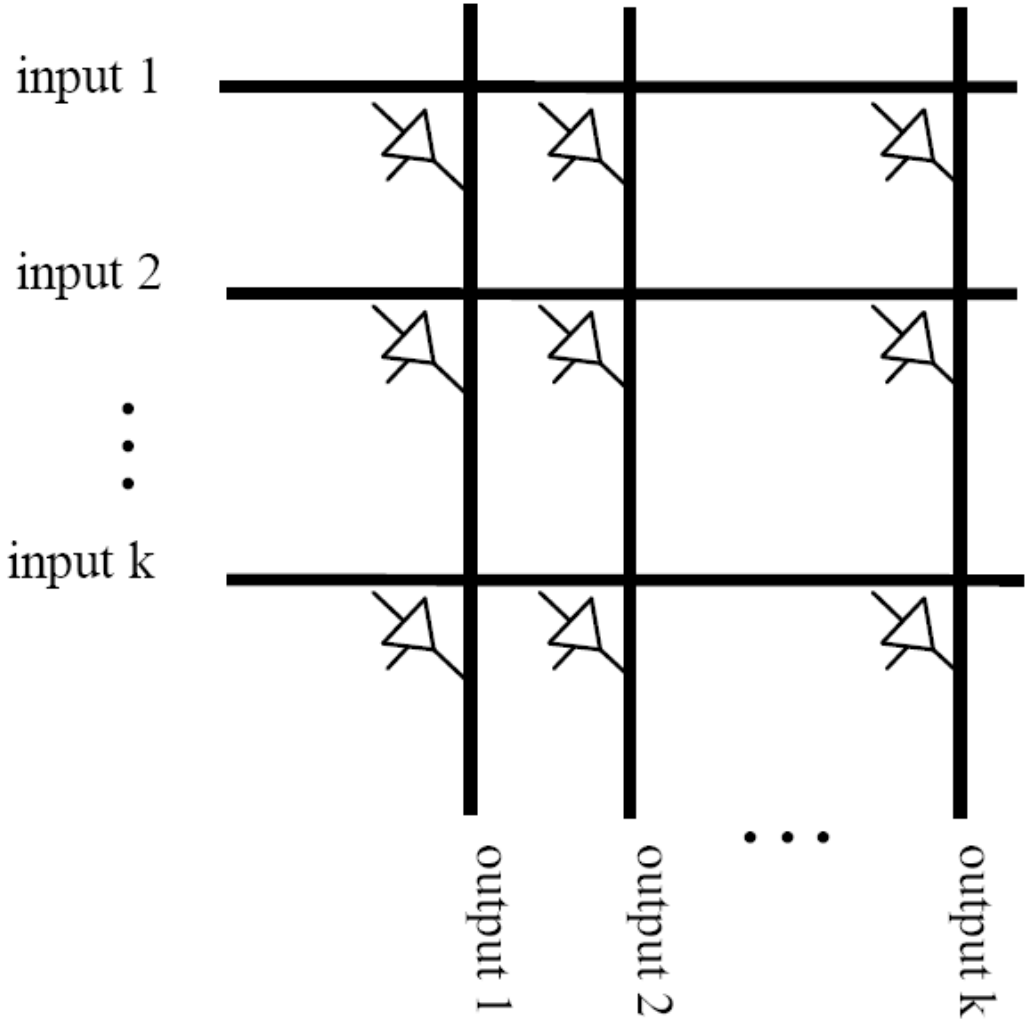
w = Flit size (bits)

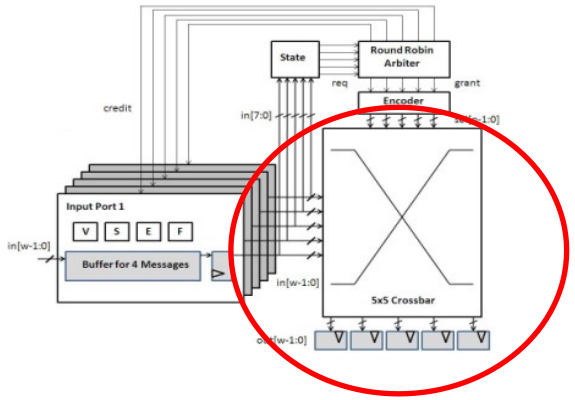
p = Ports = 5

6-bit Destination Address for 64-core system

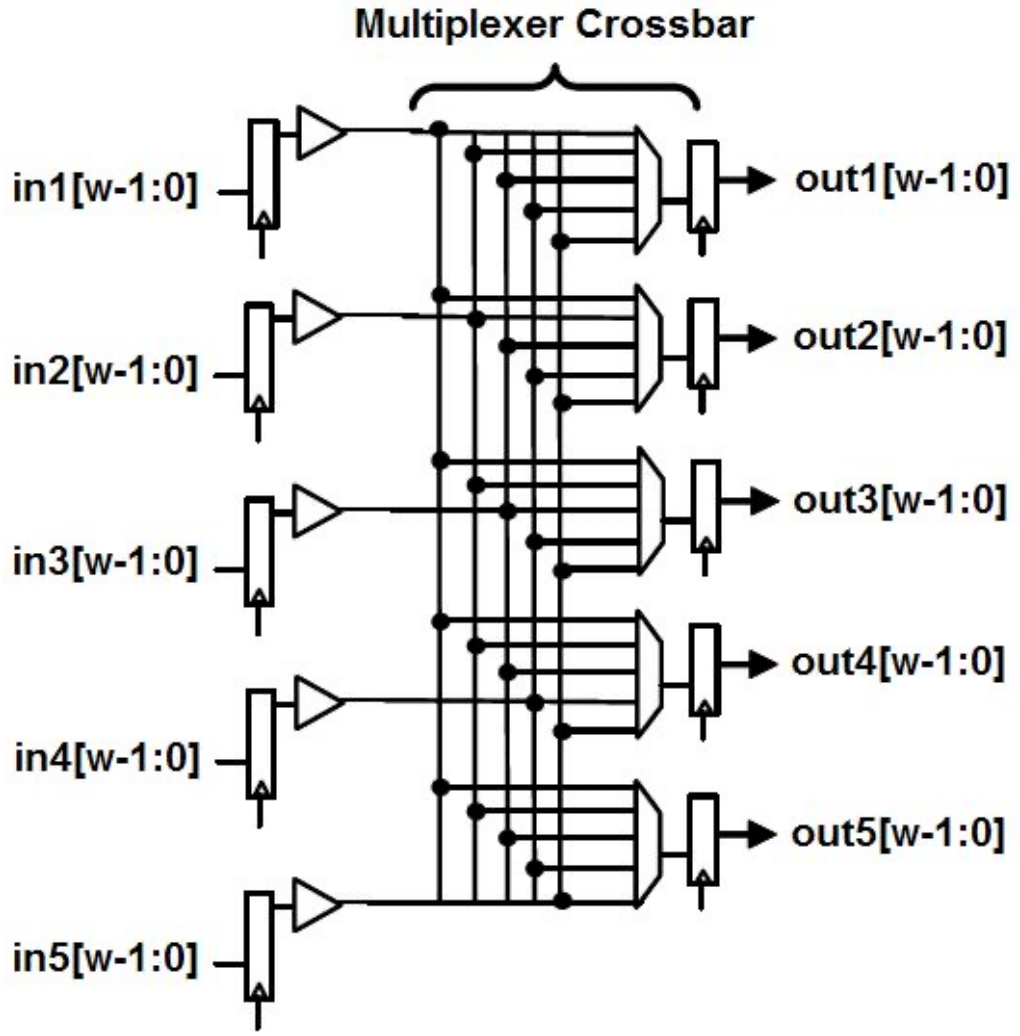


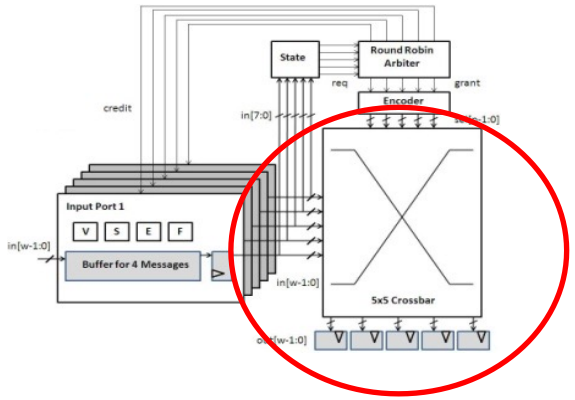
Matrix Crossbar



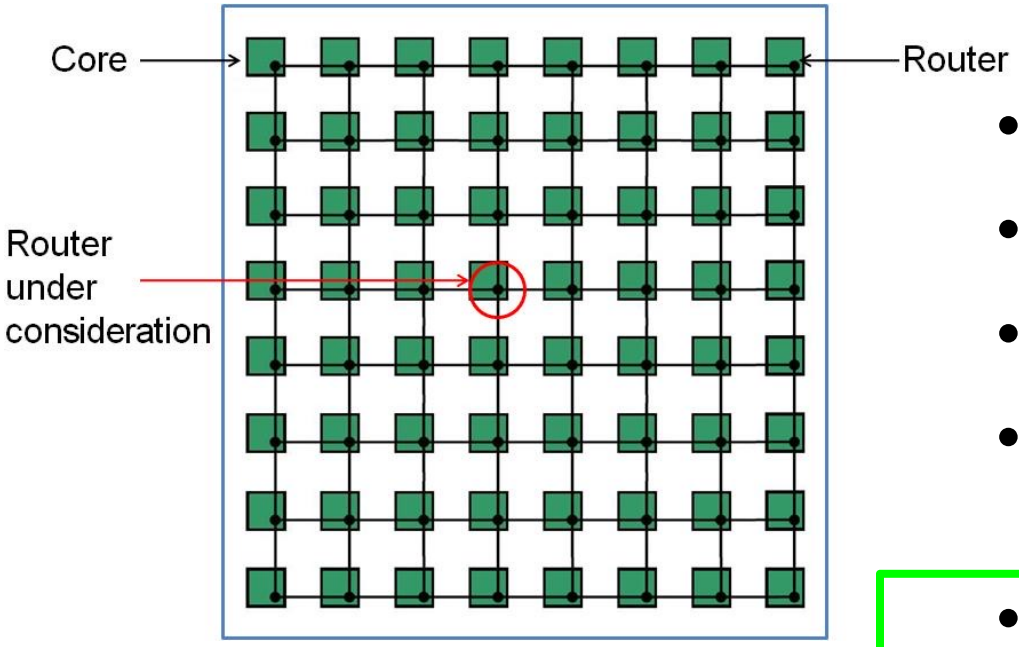


Mux Crossbar





Example System



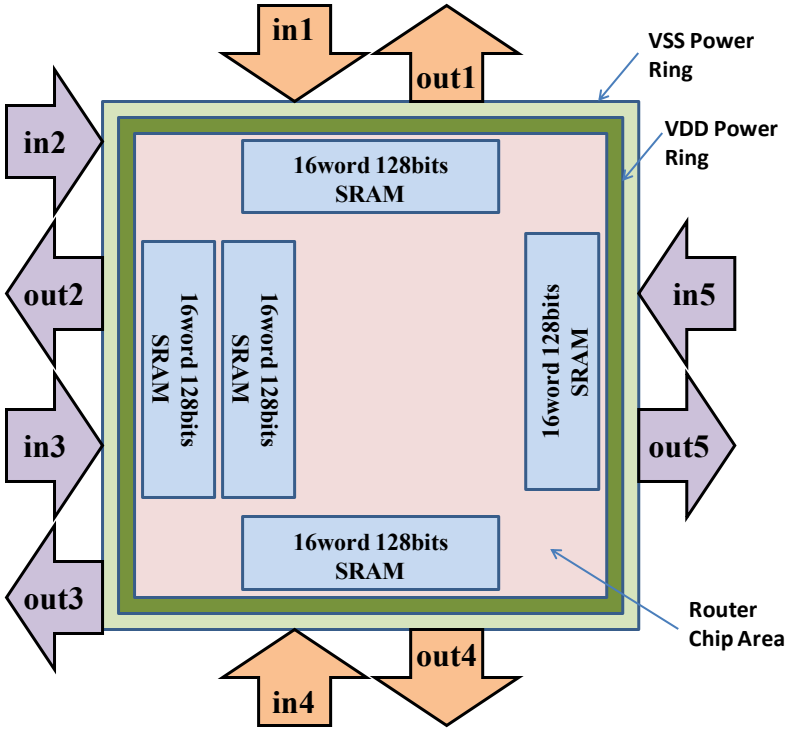
- 64 tiles.
- 1GHz frequency
- 1 Message = 512-bits
- 4 Messages per input port (2048-bits)

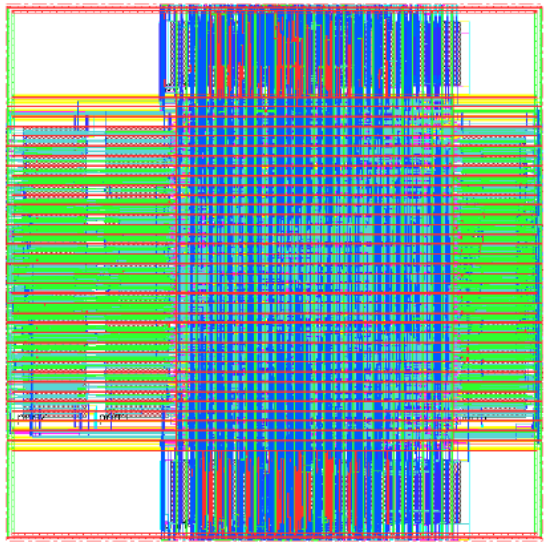
Design space
18 Routers

- Router Aspect Ratio 1
- p = 5, 8, 12
- w = 32, 64, 128 (bits)
- Matrix xbar
- Mux xbar

5x5 Router Floorplan (128bit)

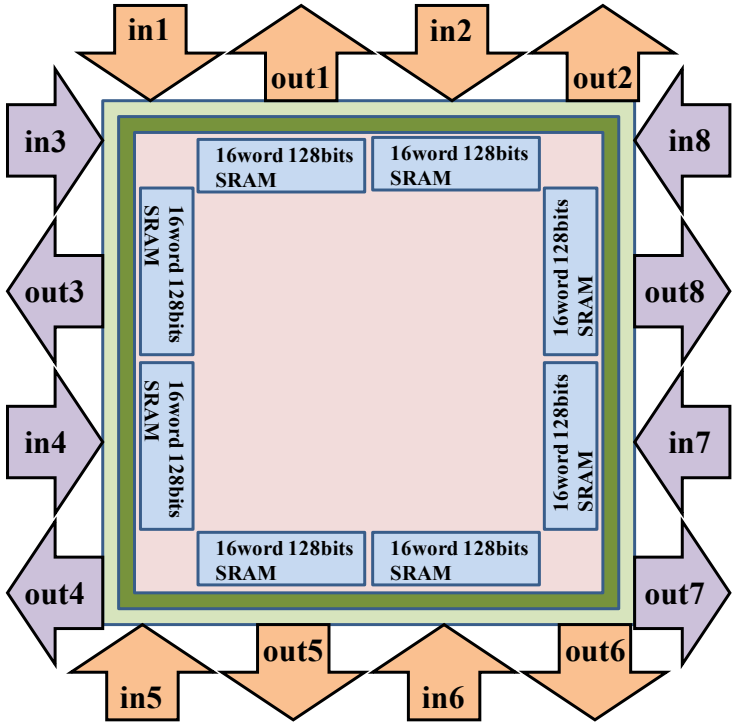
0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63





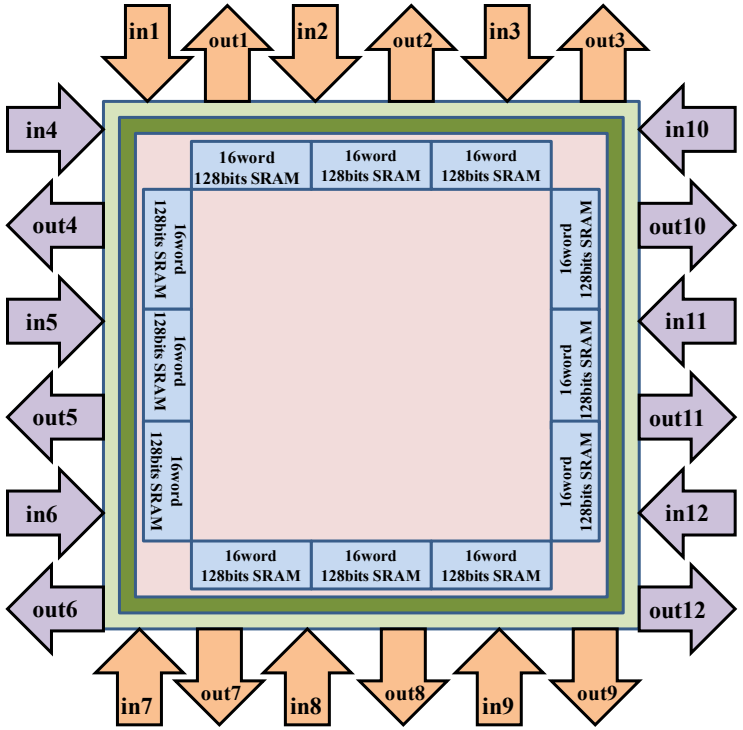
8x8 Routers Floorplan (128bit)

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

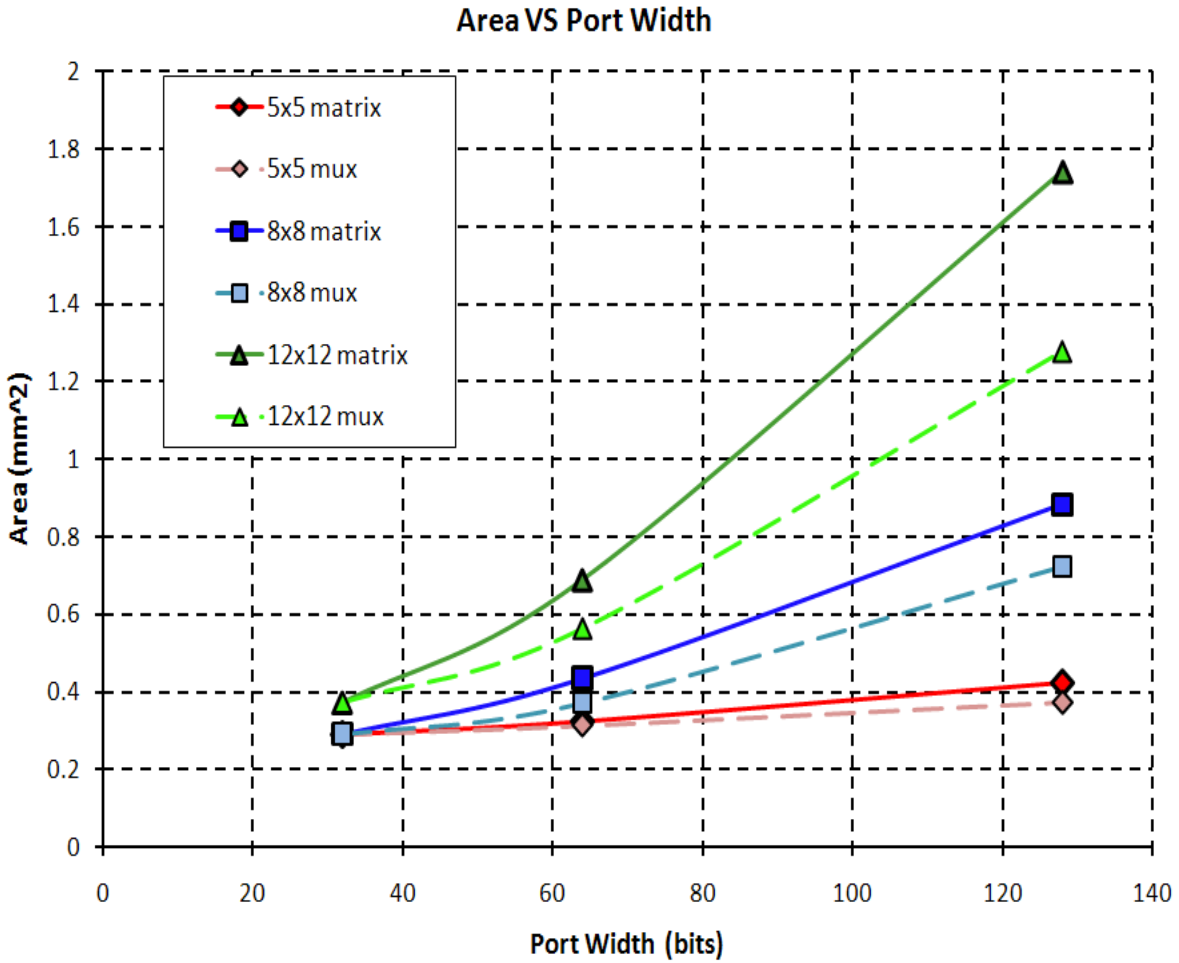


12x12 Routers Floorplan (128bit)

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

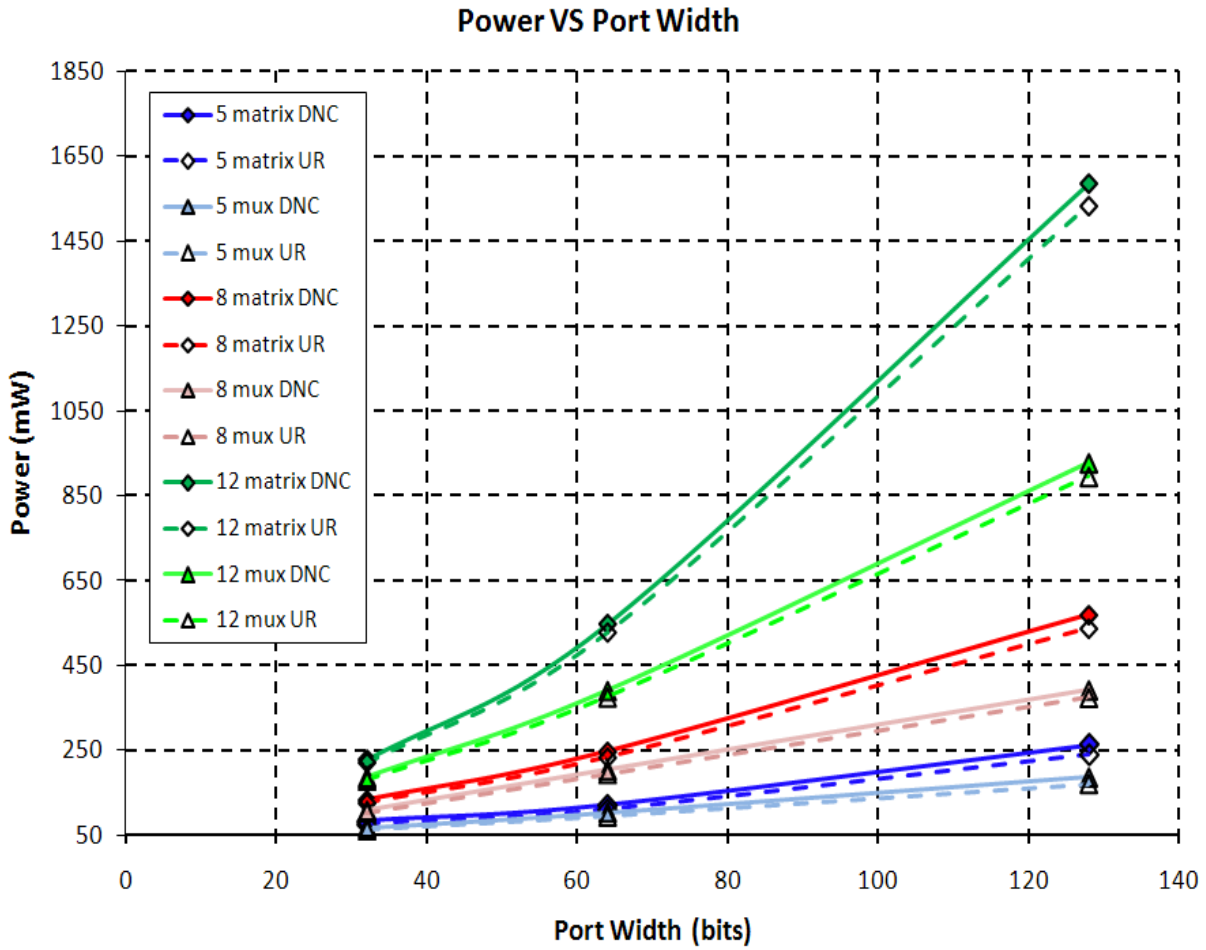


Area vs Port Width and Radix



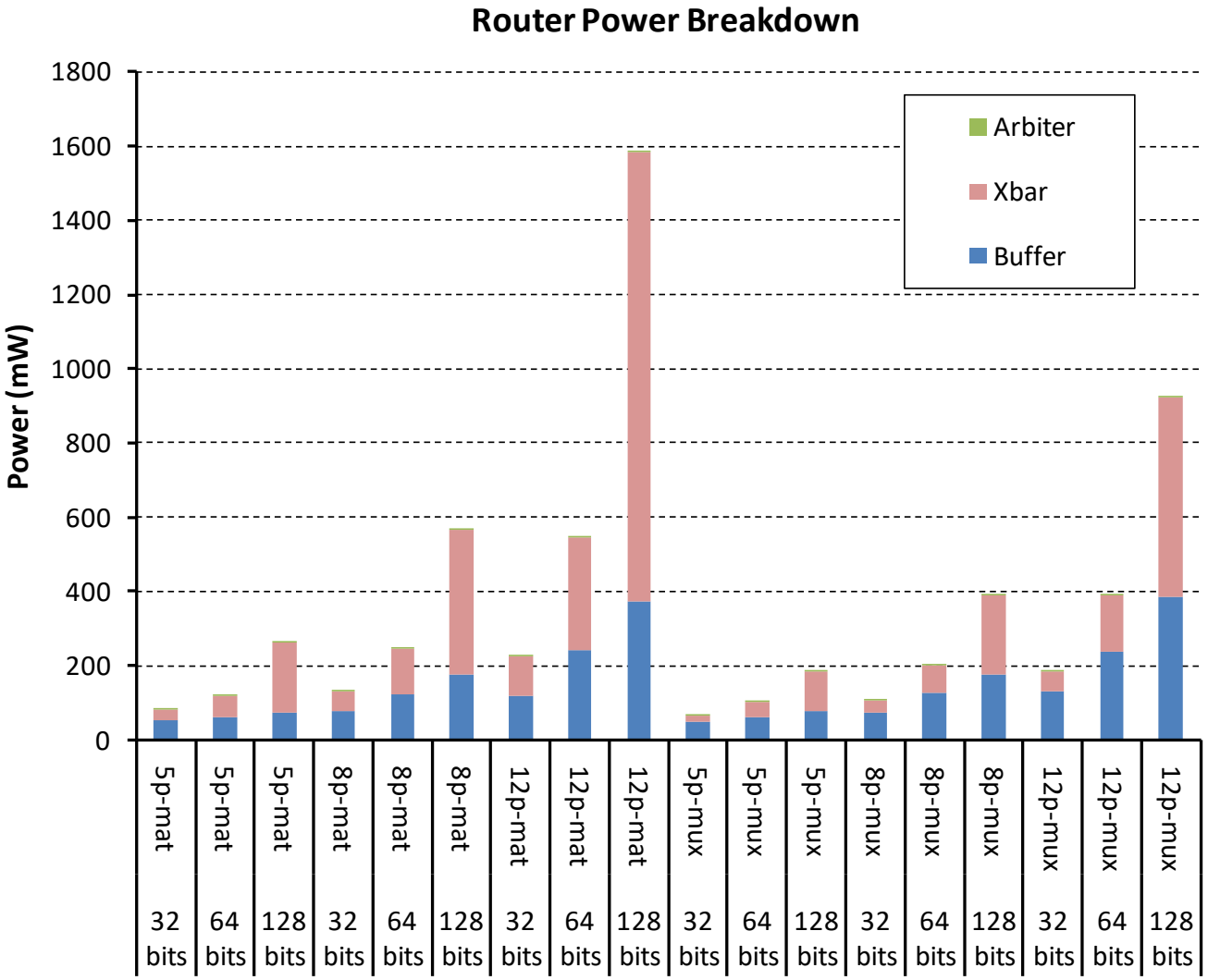
- Mux crossbar always better
- 5-12 port routers scale well (sub p^2, b^2)

Power vs Port Width and Radix



- Mux crossbar always better
- 5-12 port routers scale well (sub p^2, b^2)

Router Power Breakdown

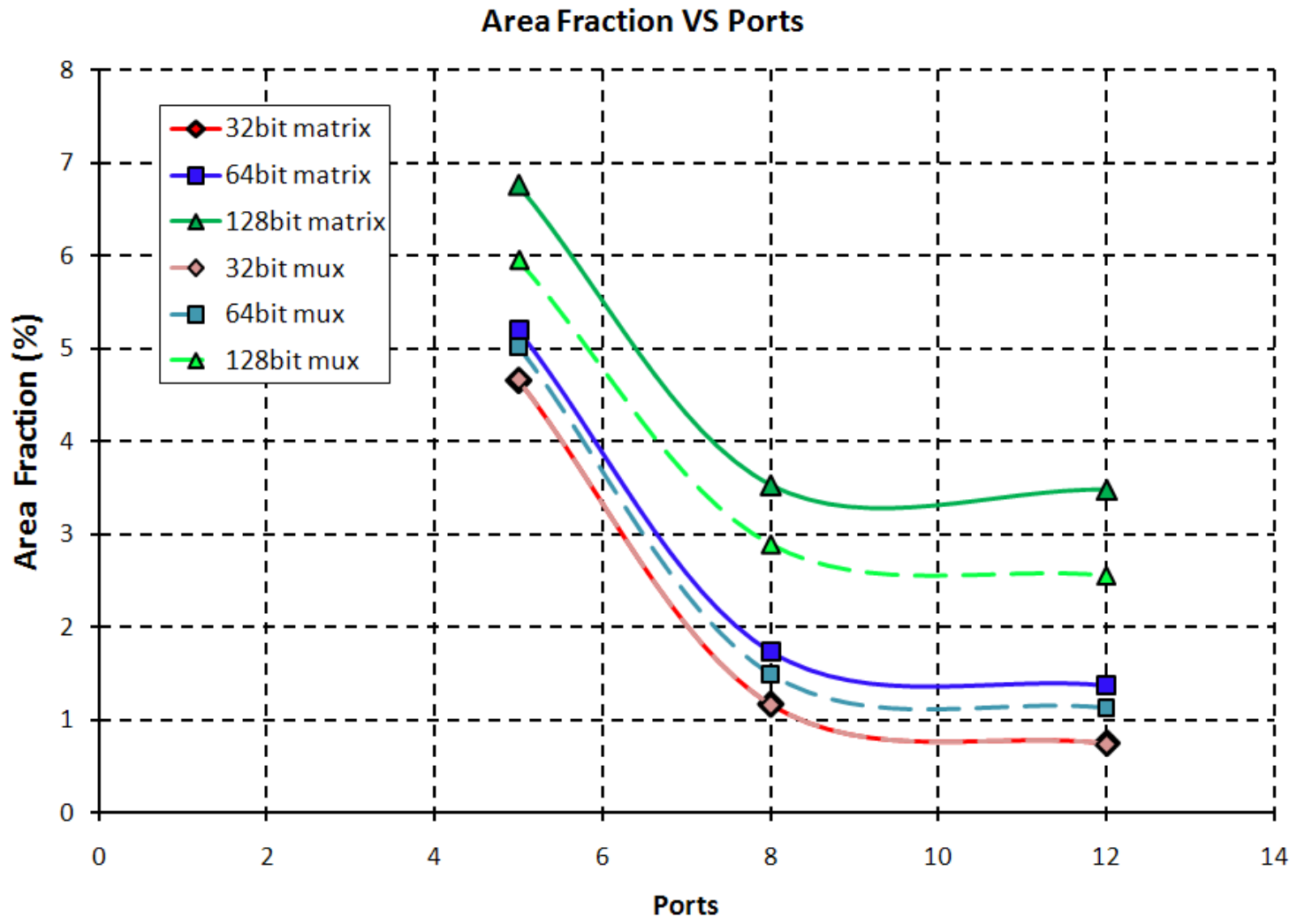


Xbar and Buffer power roughly even

Improve Xbar with Ckt/channel design (equalized, low-swing)

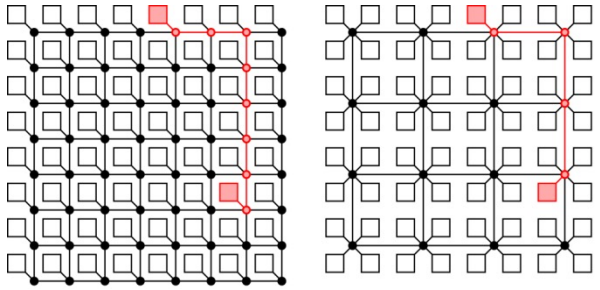
Use less buffers (circuit switching, token flow control) [Anders08, Kumar08]

Router Area per core vs. # Ports



Effects of Concentration

[Balfour06]



- Mesh to Cmesh
  - 5p routers to 8p routers

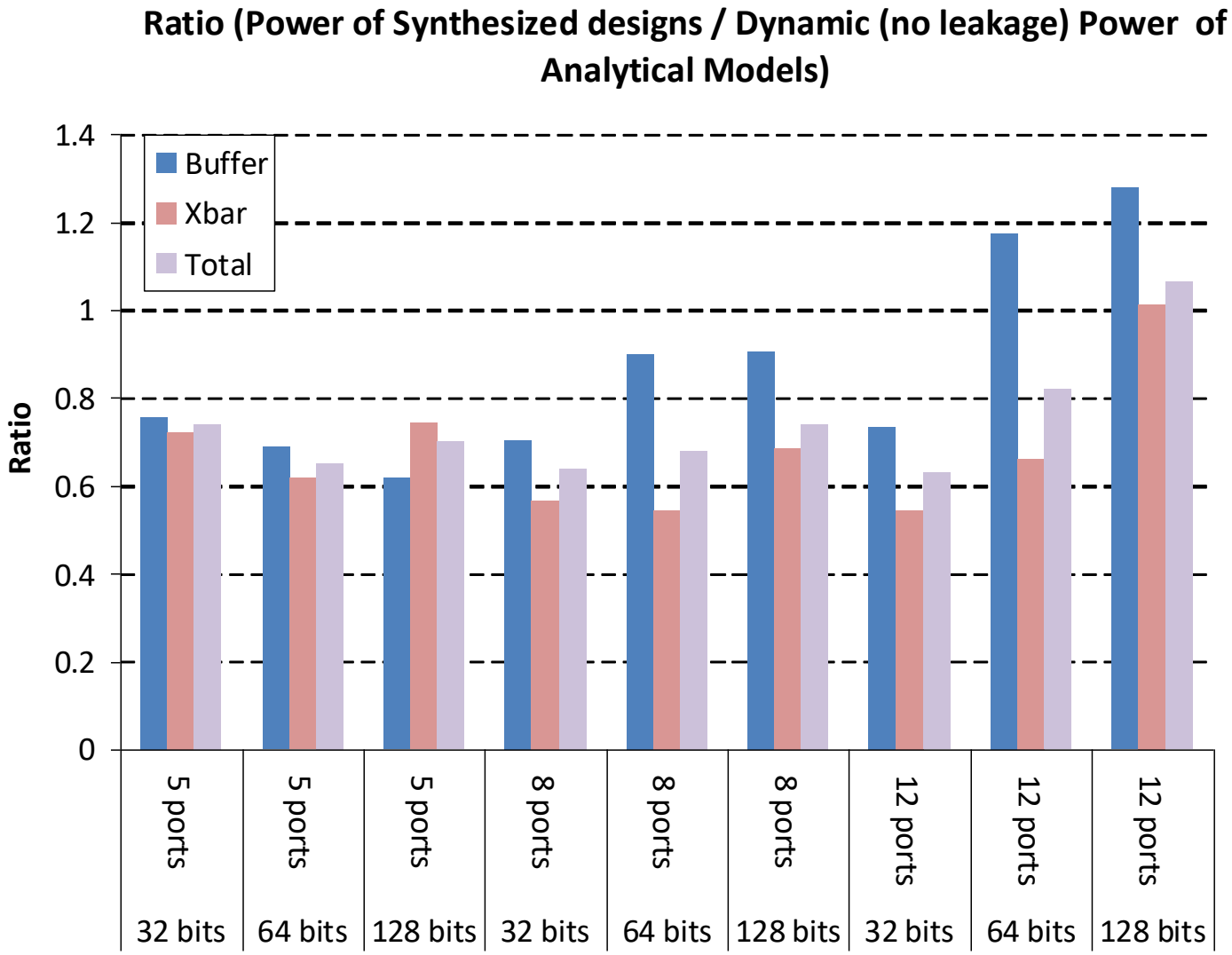
Matrix Design	Area (mm^2)	Power (mW)
4 x 5p32b-mat	1.1664	332.304
1 x 8p64b-mat	0.4356	246.3924
4 x 5p64b-mat	1.2996	484.4544
1 x 8p128b-mat	0.8836	568.2672
2 x 8p32b-mat	0.5832	264.6312
1 x 12p64b-mat	0.6889	546.8928
2 x 8p64b-mat	0.8712	492.7848
1 x 12p128b-mat	1.7424	1584.54
8 x 5p32b-mat	2.3328	664.608
1 x 12p128b-mat	1.7424	1584.54

Mux Design	Area (mm^2)	Power (mW)
4 x 5p32b-mux	1.1664	268.3056
1 x 8p64b-mux	0.3721	203.268
4 x 5p64b-mux	1.2544	410.5872
1 x 8p128b-mux	0.7225	391.0116
2 x 8p32b-mux	0.5832	215.8464
1 x 12p64b-mux	0.5625	389.5896
2 x 8p64b-mux	0.7442	406.536
1 x 12p128b-mux	1.2769	926.2188
8 x 5p32b-mux	2.3328	536.6112
1 x 12p128b-mux	1.2769	926.2188

- Works well for small flits and number of ports

Orion 2.0 vs P & R design

[Kahng09] [Shamim09]

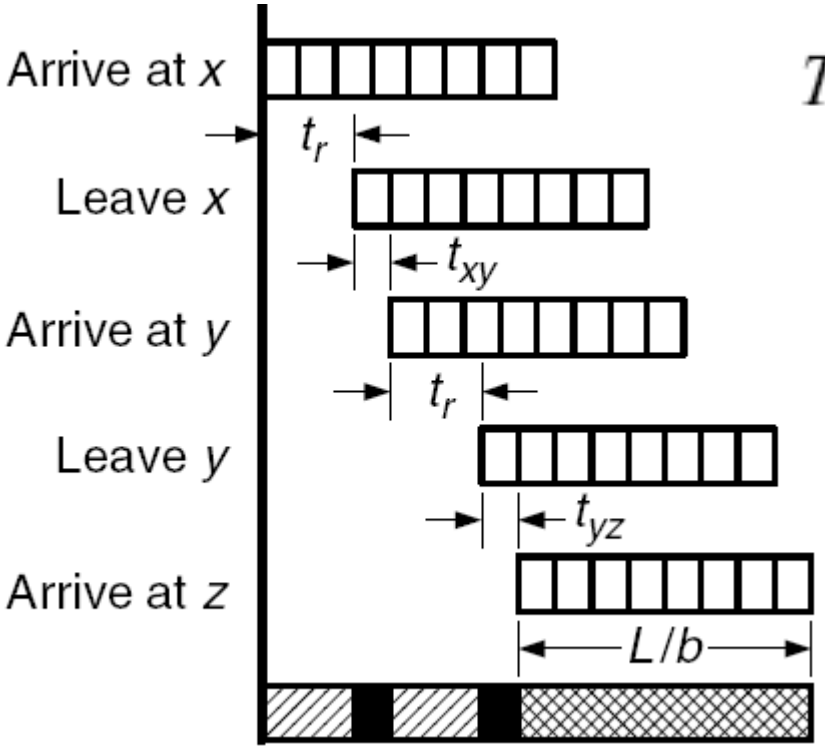


Lecture Roadmap

- Networking Basics
- Building Blocks
- Evaluation

Latency Components

- Zero-load latency
  - Average latency w/o contention



$T_0 = 2t_r + (t_{xy}+t_{yz}) + L/b$

$$T_0 = H_{\min} t_r + \frac{D_{\min}}{v} + \frac{L}{b}$$

Router delays

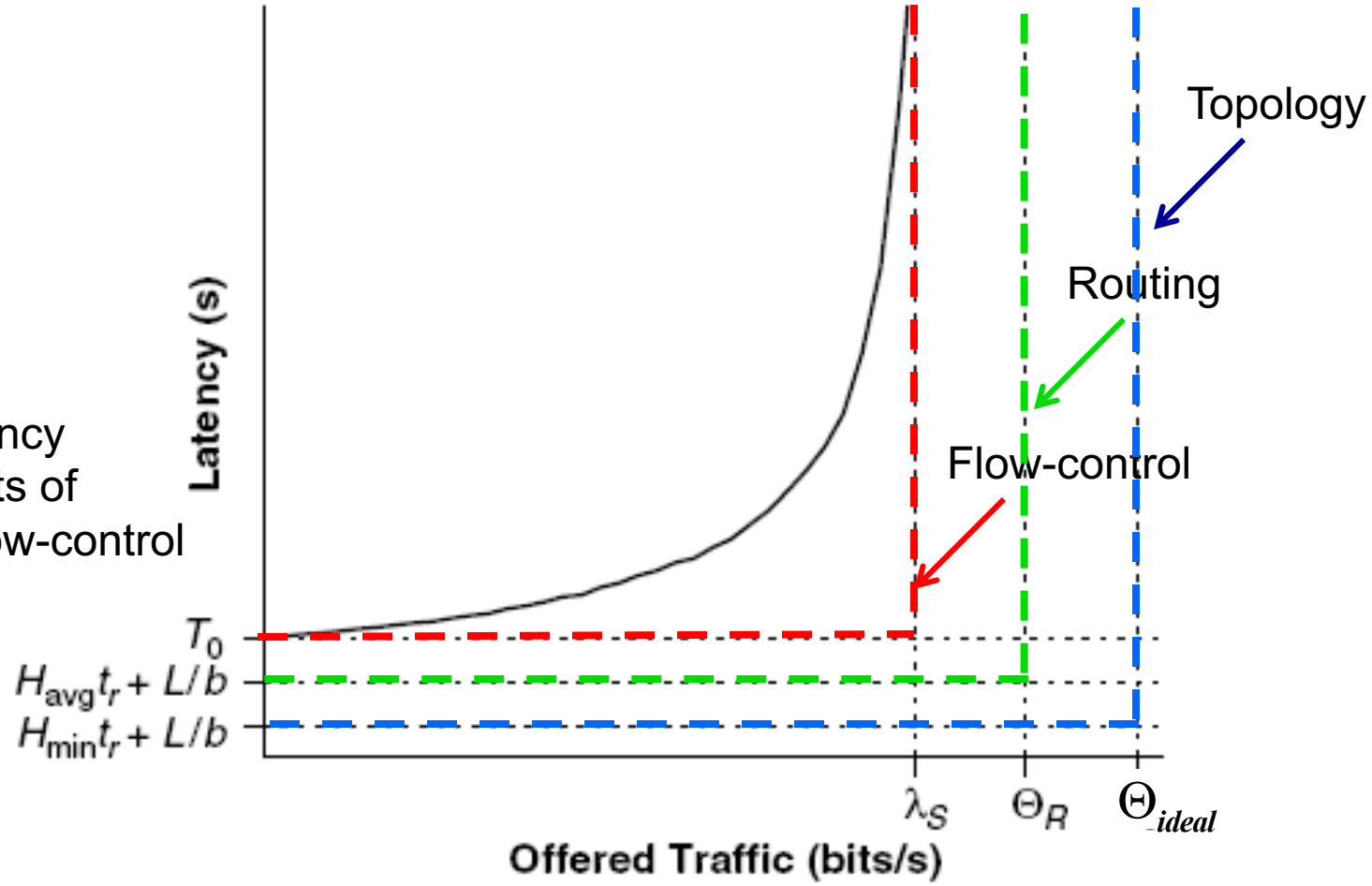
Channel delays

Serialization delay

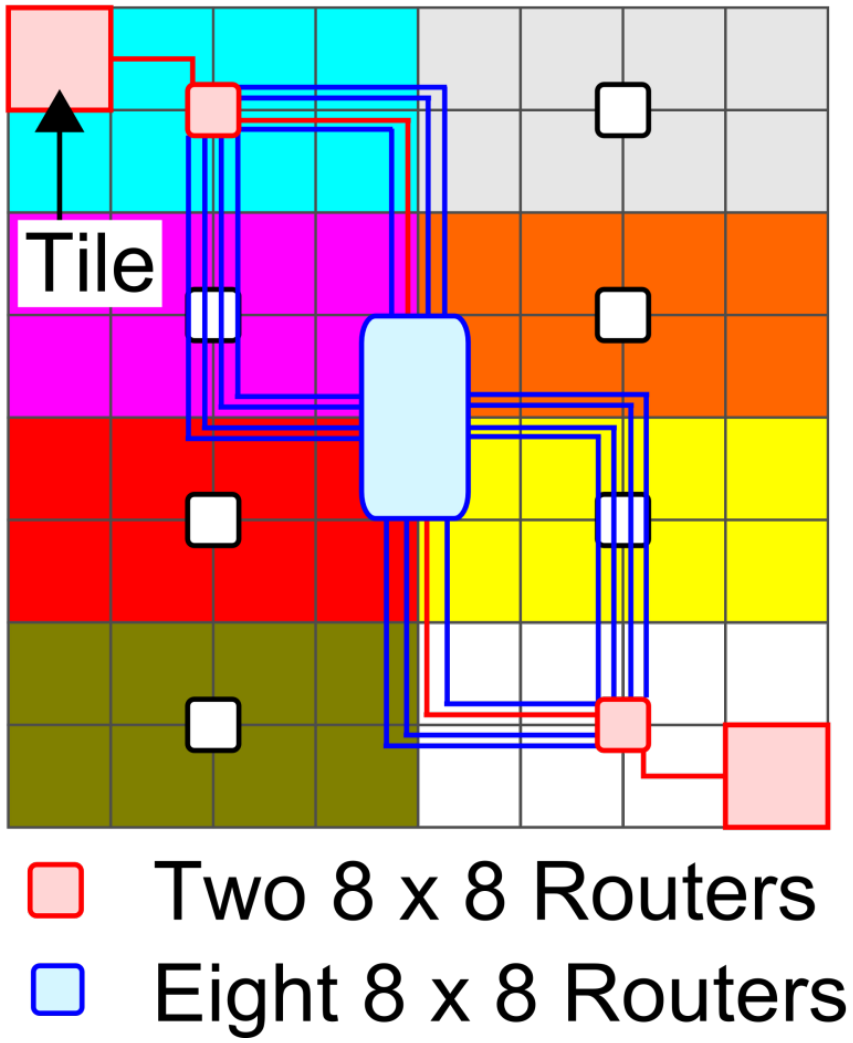
H_{min} – average minimum number of hops
t_r – Router delay
D_{min} – average minimum distance
v – signal velocity
L – packet length in bits
b – router-to-router channel bandwidth

Network performance plots

Zero-load latency includes effects of routing and flow-control



Clos with electrical interconnects

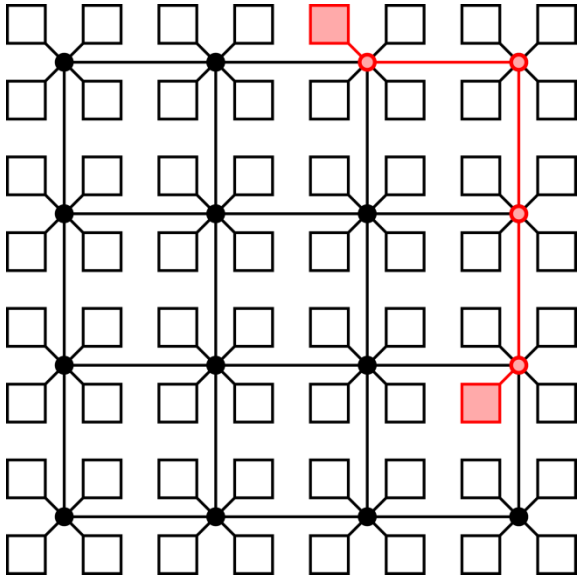


8-ary 3-stage Clos

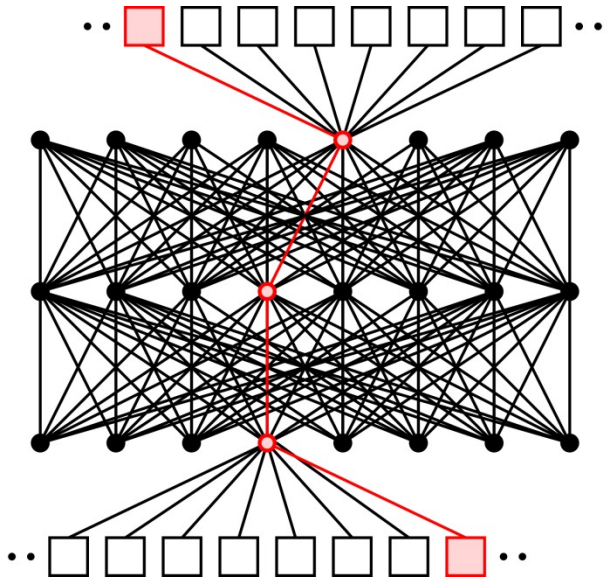
- 10-15 mm channels
- Equalized
- Pipelined Repeaters

Simulation setup

- Cycle-accurate microarchitectural simulator
- Traffic patterns based on partition application model
  - Global traffic – UR, P2D, P8D
  - Local traffic – P8C
- 64-tile system, 512-bit messages
- Events captured during simulations to calculate power



CMesh



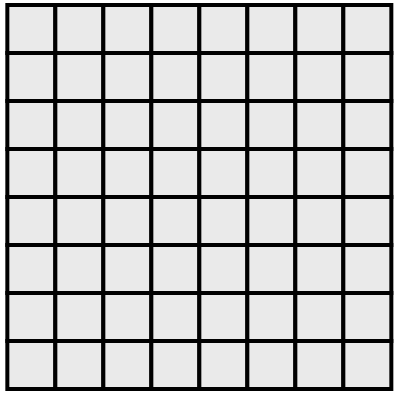
Clos

Partition application model

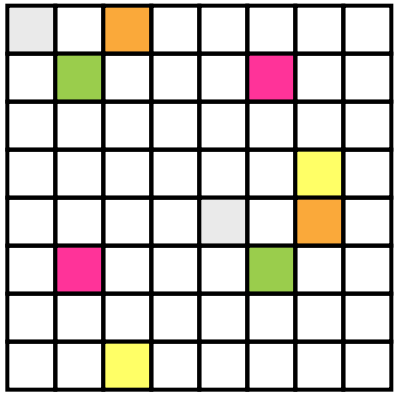
- Tiles divided into logical partitions and communication is within partition

[Joshi'09]

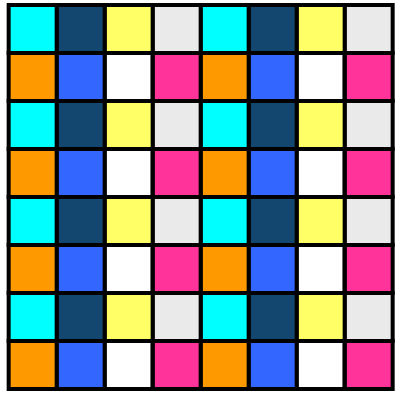
- Logical partitions mapped to physical tiles
  - Co-located tiles → Local traffic
  - Distributed tiles → Global traffic



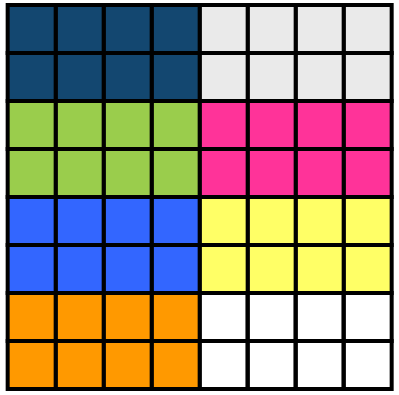
Uniform random (UR)



2 tiles per partition that are distributed across the chip (P2D)



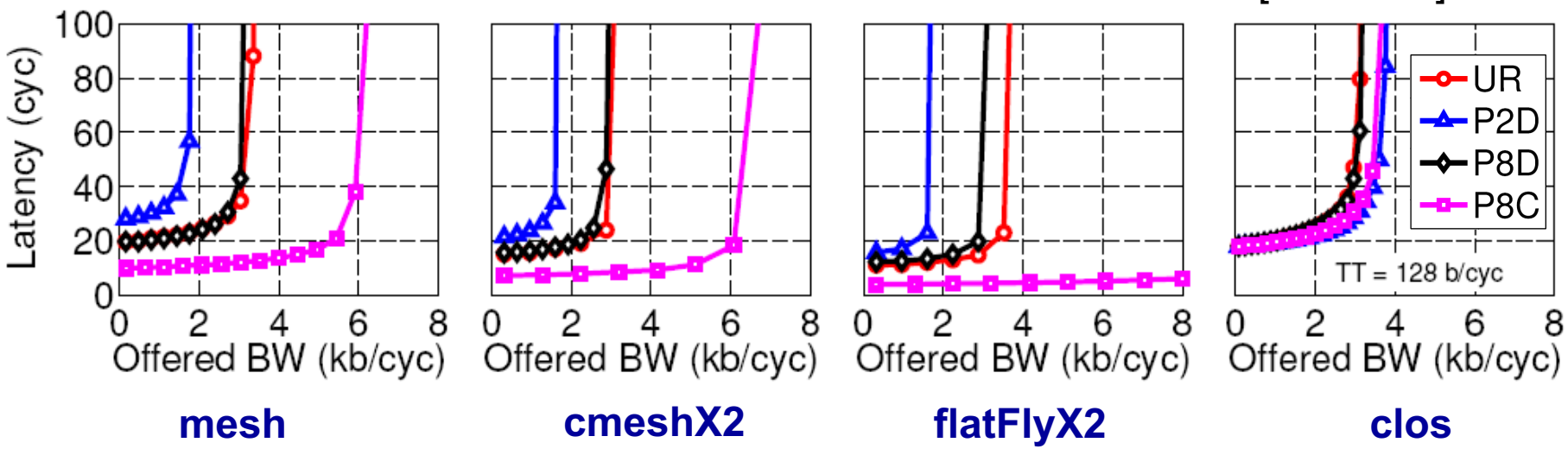
8 tiles per partition that are distributed across the chip (P8D)



8 tiles per partition that are co-located (P8C)

Latency vs BW

[Joshi09b]

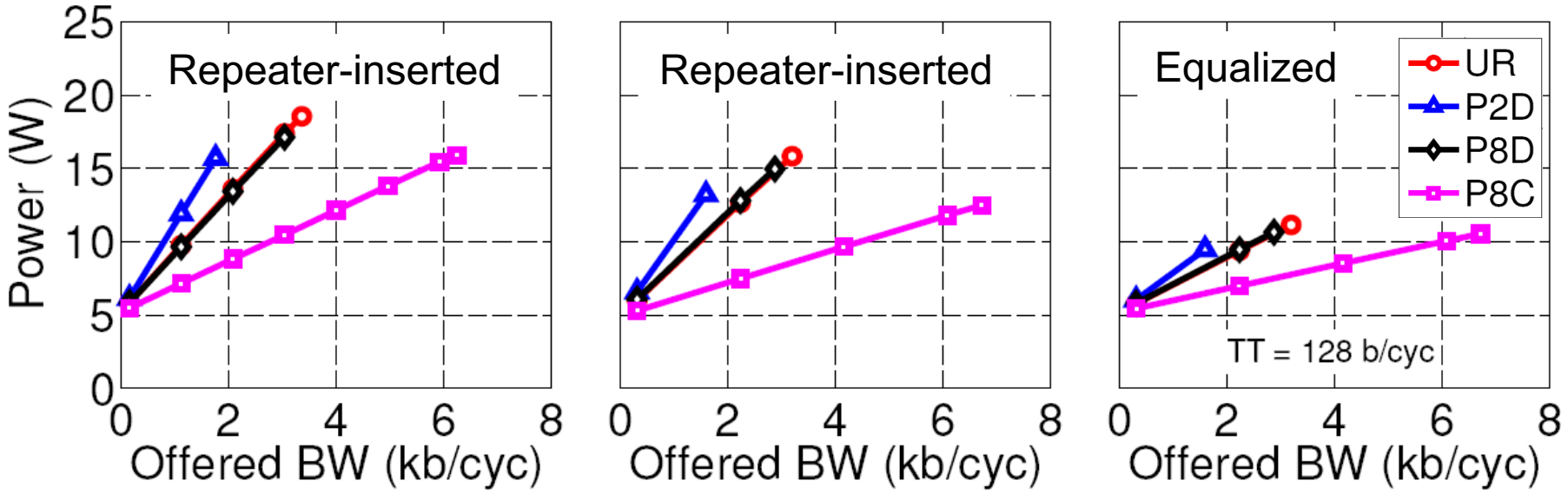


Ideal Throughput θ_T = 8 kb/cyc for UR

- flatFlyX2 vs mesh/cmeshX2
  - Saturation BW → comparable (UR, P8D, P2D)
  - Latency → flatFlyX2 has lower latency

- clos vs mesh/cmeshX2/flatFlyX2
  - Saturation BW → uniform for all traffic, comparable to UR of mesh
  - Latency → uniform for all traffic, comparable to UR of mesh

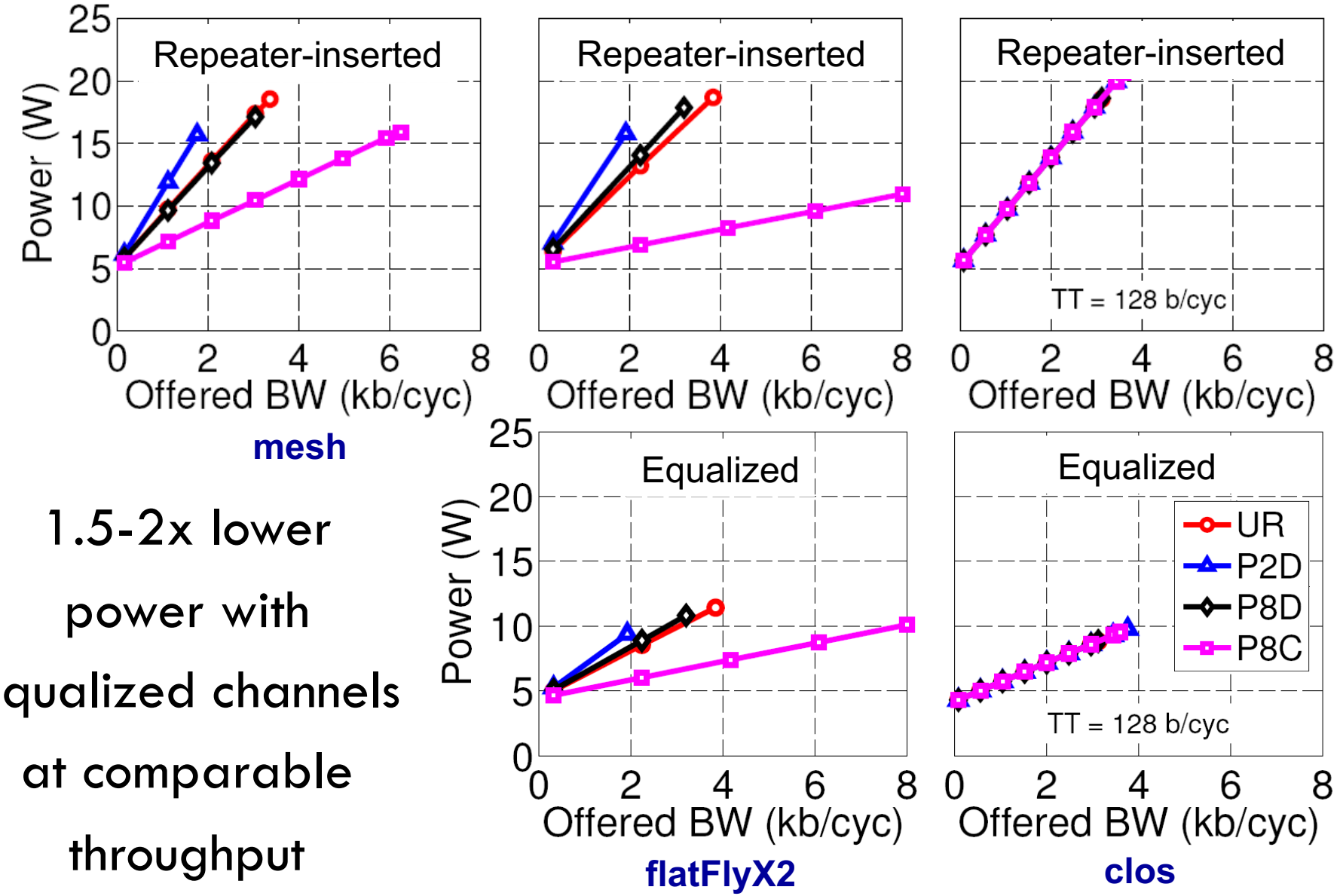
Mesh vs CMeshX2



mesh **cmeshX2** **cmeshX2**

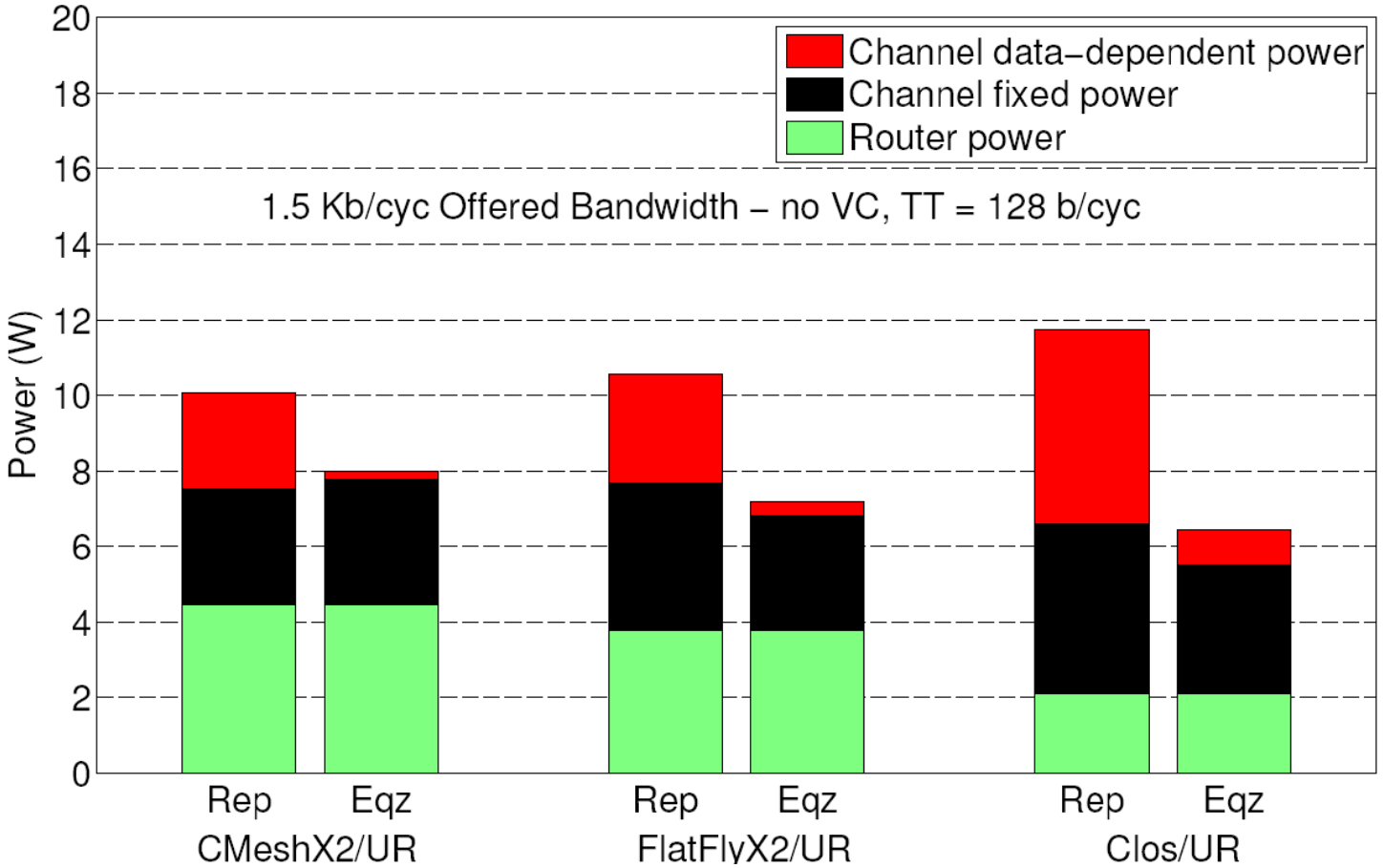
- Repeater-inserted interconnects
  - cmeshX2 lower power than mesh at comparable throughput
- Equalized interconnects
  - cmeshX2 has further 1.5x reduction in power
  - Channel gains masked by router power

Power vs BW plots –repeater inserted pipelined vs equalized



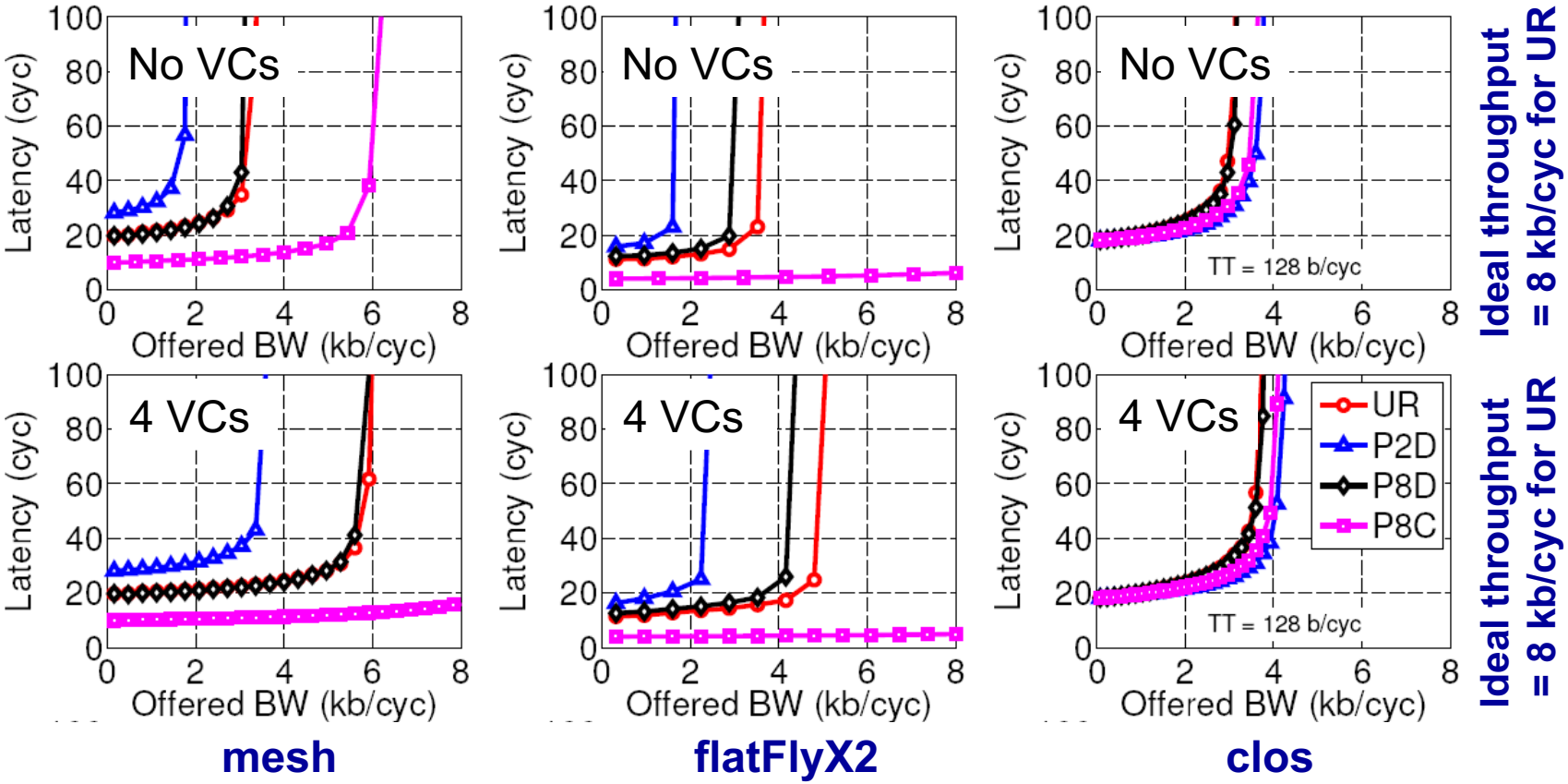
1.5-2x lower power with equalized channels at comparable throughput

Power split



- Channel DDE reduces by 4-10x using equalized links
- Channel fixed power and router power need to be tackled

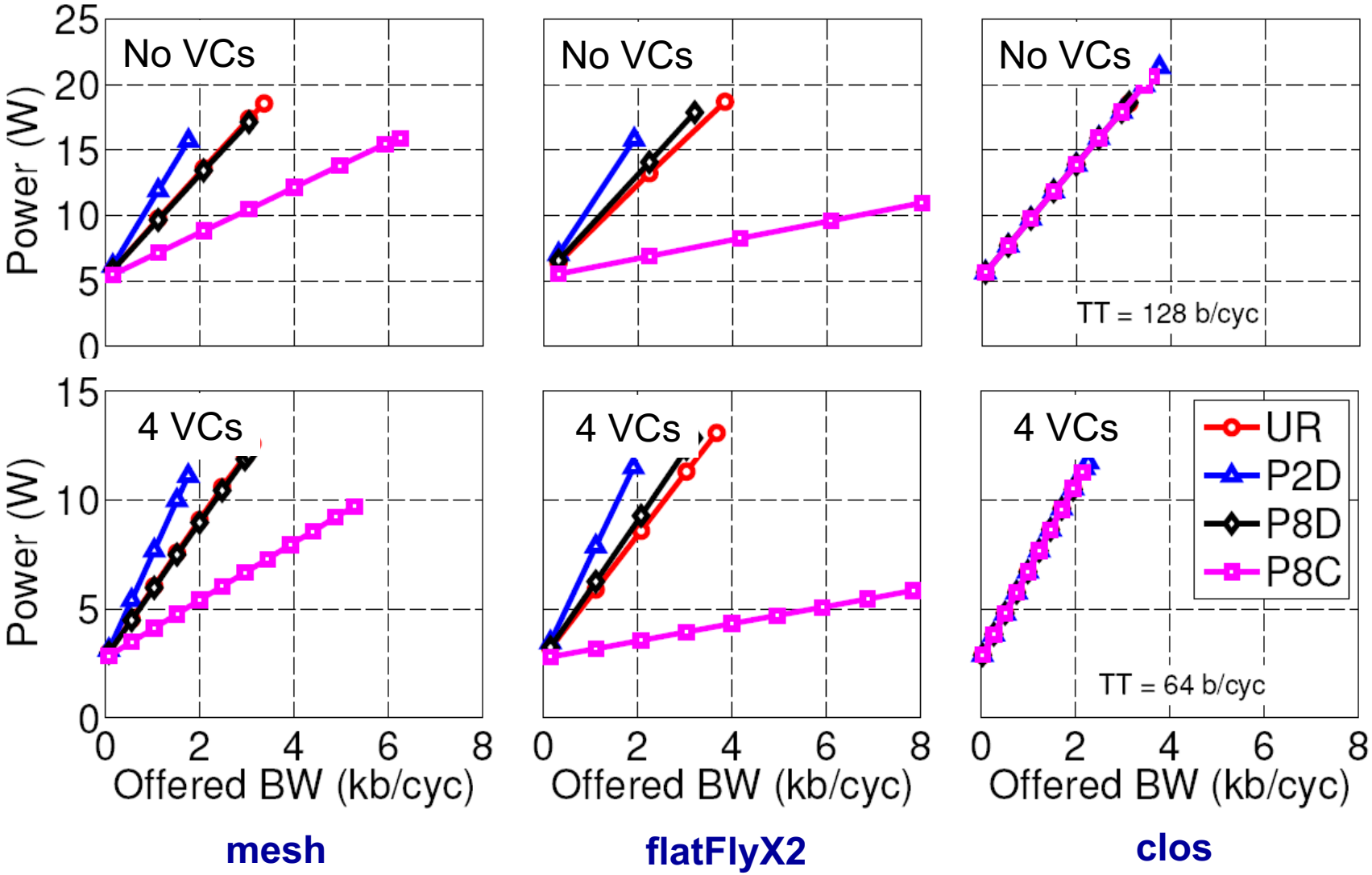
Latency vs BW – no VC vs 4 VCs



Saturation throughput improves using VCs

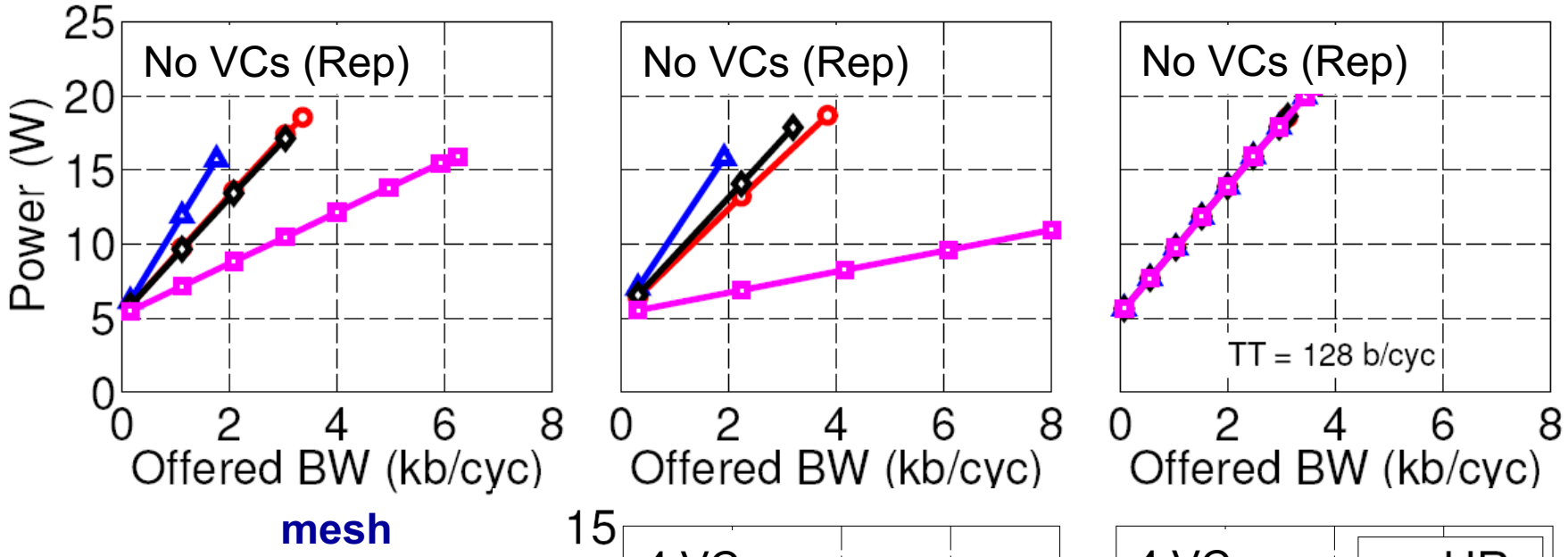
Small change in power at comparable throughput

Power vs BW – no VC vs 4 VCs, repeater inserted pipelined

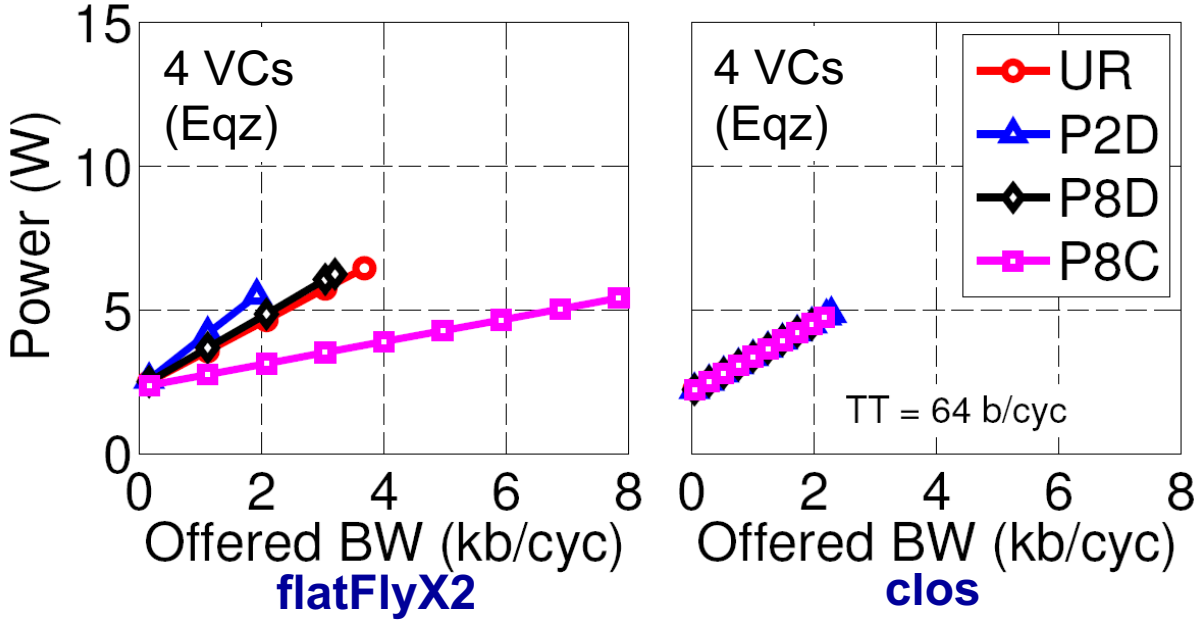


25-50% lower power using VCs at comparable throughput

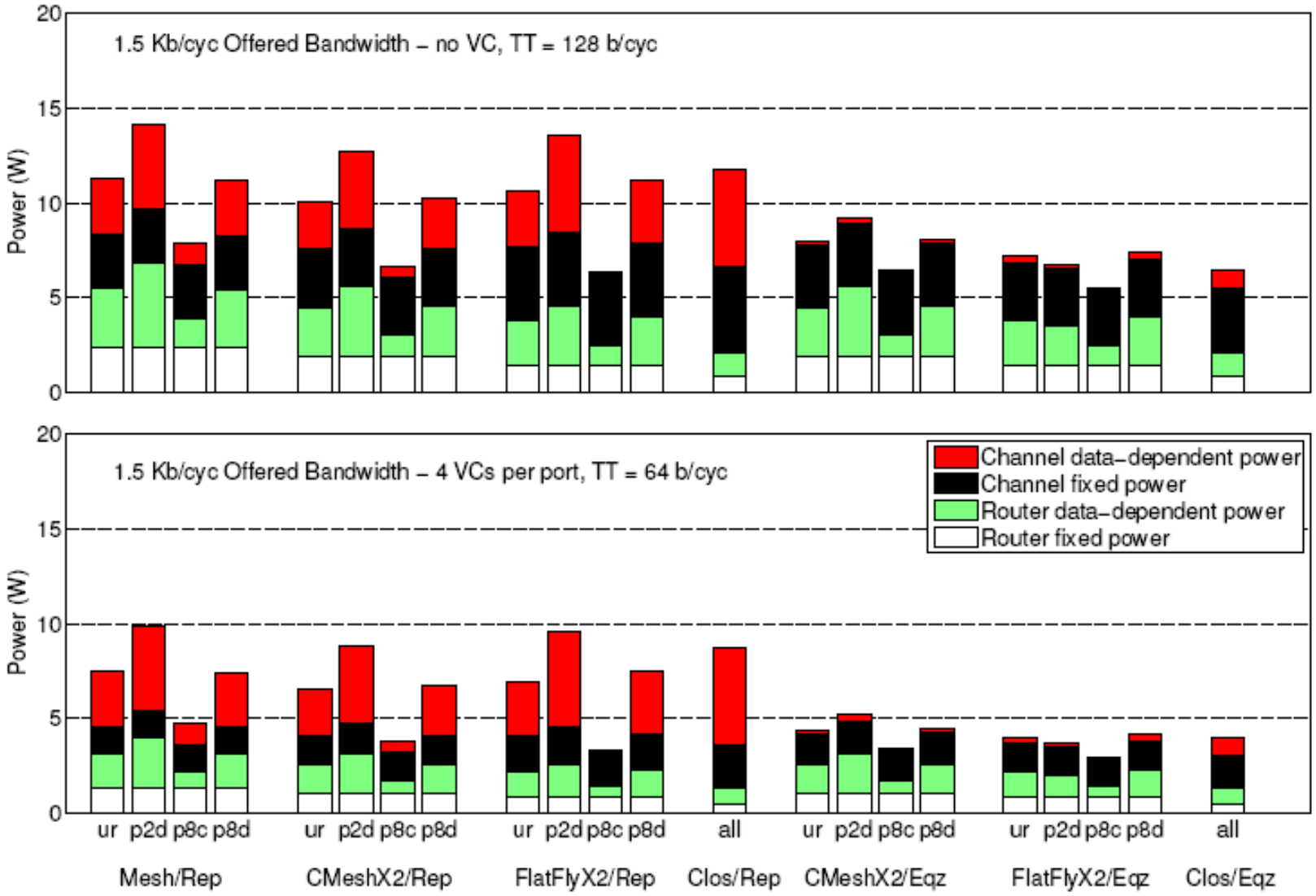
Power vs BW– no VC case, repeater inserted pipelined vs 4 VCs, equalized



2-3x lower power obtained using equalized interconnects and VCs at comparable throughput

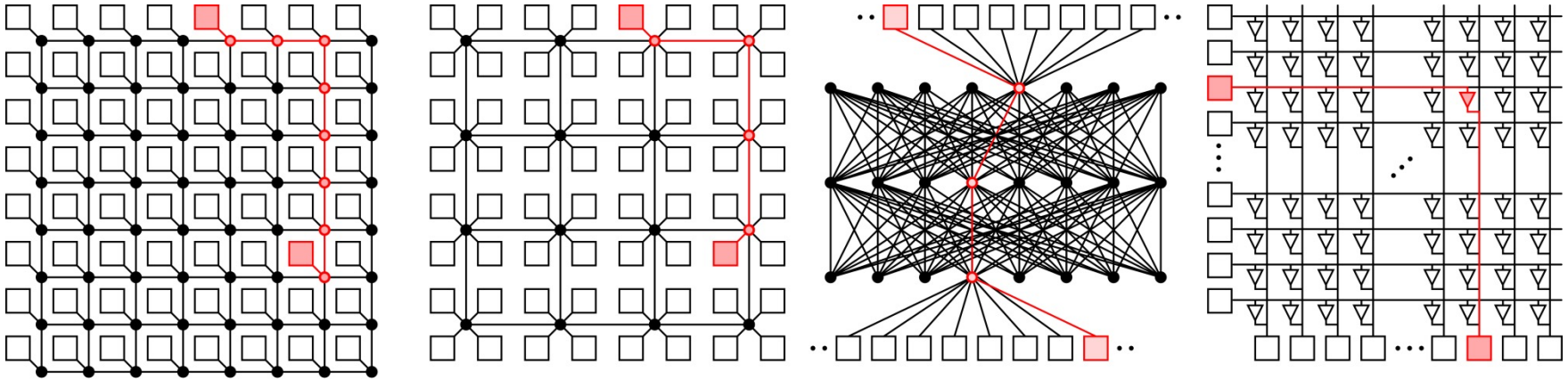


Power split



- VCs an indirect way to increase impact of channel power
  - Narrower networks, lower power for same throughput, keep utilization high

Summary



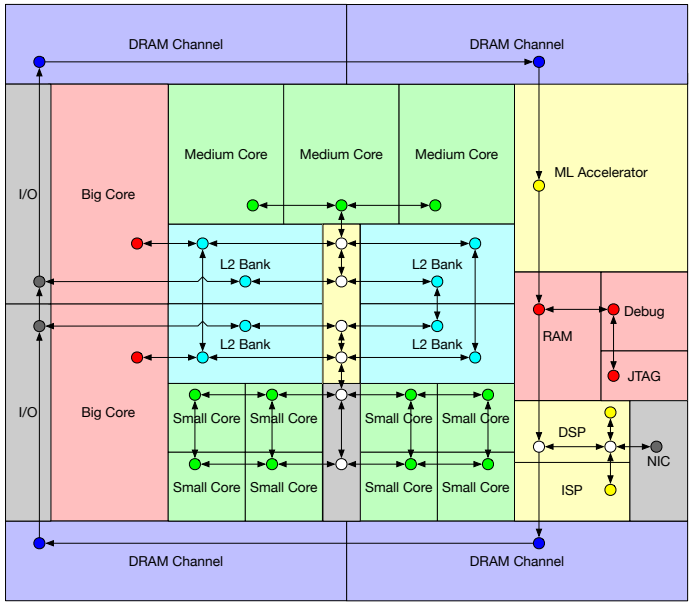
- Cross-cut approach for on-chip system interconnects design needed
  - Application mapping
  - Topology, Routing, Flow-control
  - Improving Routers and Channels equally important
    - New circuit design (low-swing, equalized)
    - System – DVFS, bus-encoding

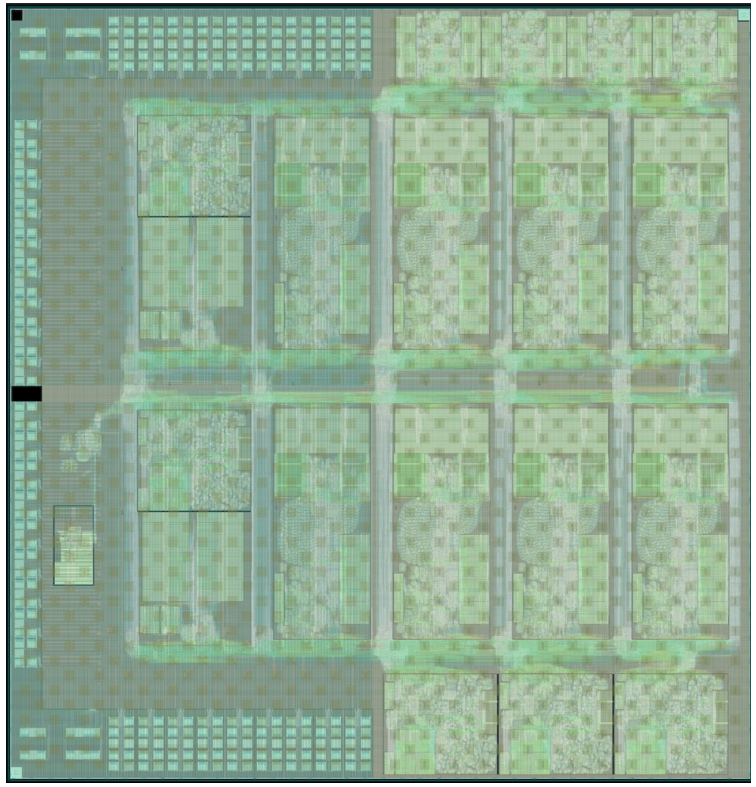
To probe further (tools and sites)

- DSENT - A Tool Connecting Emerging Photonics with Electronics for Opto-Electronic Networks-on-Chip Modeling
  - https://dspace.mit.edu/handle/1721.1/85863
- Orion Router Design Exploration Tool
  - https://github.com/eigenpi/vnoc20
- Router RTLs
  - Bob Mullins' Netmaker (http://www-dyn.cl.cam.ac.uk/~rdm34/wiki)
- Network simulators
  - Garnet (http://www.princeton.edu/~niketa/garnet.html)
  - Booksim (http://nocs.stanford.edu/booksim.html)

Generating Network-on-Chips

- Constellation – a Chisel network-on-chip generator
- Generate realistic interconnects for modern SoCs
- Configurable routing/topology/micro-architecture
- constellation.readthedocs.io
- If interested, email me – jzh@berkeley.edu





Bibliography

[Agarwal09] N. Agarwal, T. Krishna, L.-S. Peh and N. K. Jha, " GARNET: A Detailed On-Chip Network Model inside a Full-System Simulator " In Proceedings of IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), Boston, Massachusetts, April 2009.

[Anders08] M. Anders, H. Kaul, M. Hansson, R. Krishnamurthy, S. Borkar "A 2.9Tb/s 8W 64-Core Circuit-switched Network-on-Chip in 45nm CMOS," *European Solid-State Circuits Conference, 2008* .

[Balfour06] J. Balfour and W. Dally ,"Design tradeoffs for tiled CMP on-chip networks.," Int'l Conf. on Supercomputing, June 2006.

[Bell08] S. Bell et al "TILE64TM Processor: A 64-Core SoC with Mesh Interconnect," ISSCC pp. 88-598, 2008.

[Benini02] L. Benini and G. de Micheli, "Networks on Chips: A New SoC Paradigm," in Computer Magazine, vol. 35 issue 1, pp. 70-78, 2002.

[Clos53] C. Clos. A study of non-blocking switching networks. Bell System Technical Journal, 32:406–424, 1953.

[Dally92] W. J. Dally, "Virtual-channel flow control," IEEE Transactions on Parallel and Distributed Systems, vol. 3, no. 2, pp. 194–205, 1992.

[Dally01] W. J. Dally and B. Towles, "Route Packets, Not Wires: On-chip Interconnection Networks," DAC 2001, pp. 684-689.

[Dally04] W. Dally and B. Towles. *Principles and Practices of Interconnection Networks. Morgan Kaufmann, 2004.*

[Gunn06] C. Gunn, "CMOS photonics for high-speed interconnects,"IEEE Micro, 26(2):58–66, Mar./Apr. 2006.

[Joshi09] Joshi, A., B. Kim, and V. Stojanović,"Designing Energy-efficient Low-diameter On-chip Networks with Equalized Interconnects," *IEEE Symposium on High-Performance Interconnects,* New York, NY, 10 pages, August 2009.

[Kahng09] A. Kahng, B. Li, L-S. Peh and K. Samadi "ORION 2.0: A Fast and Accurate NoC Power and Area Model for Early-Stage Design Space Exploration" in Proceedings of *Design Automation and Test in Europe (DATE), Nice, France, April 2009*

Bibliography

[Kim07] J. Kim, J. Balfour, and W. J. Dally, “Flattened butterfly topology for on-chip networks,” in Proc. 40th Annual IEEE/ACM International Symposium on Microarchitecture MICRO 2007, 1–5 Dec. 2007, pp. 172–182

[Kim08] B. Kim and V. Stojanovic “Characterization of equalized and repeated interconnects for NoC applications,” IEEE Design and Test of Computers, 25(5):430–439, 2008.

[Kim09] B. Kim and V. Stojanovic, “A 4Gb/s/ch 356fJ/b 10mm equalized on-chip interconnect with nonlinear charge injecting transmitter filter and transimpedance receiver in 90nm cmos technology,” in Proc. Digest of Technical Papers. IEEE International Solid-State Circuits Conference ISSCC 2009, pp. 66–67, 8–12 Feb. 2009.

[Krishna08] T.Krishna, A. Kumar, P. Chiang, M. Erez and L-S. Peh, " NoC with Near-Ideal Express Virtual Channels Using Global-Line Communication " In Proceedings of Hot Interconnects (HOTI), Stanford, California, August 2008.

[Kumar08] A. Kumar, L-S. Peh and N. Jha, " Token Flow Control ," in Proceedings of 41st International Symposium on Microarchitecture (MICRO), Lake Como, Italy, November 2008.

[Mensink07] E. Mensink et al., “A 0.28pJ/b 2Gb/s/ch transceiver in 90nm CMOS for 10 mm on-chip interconnects,” in Proc. Digest of Technical Papers. IEEE International Solid-State Circuits Conference ISSCC 2007, 11–15 Feb. 2007, pp. 414–612.

[Nawathe08] U. Nawathe et al., “Implementation of an 8-core, 64-thread, power-efficient SPARC server on a chip,” IEEE Journal of Solid-State Circuits, vol. 43, no. 1, pp. 6–20, Jan. 2008

[Orcutt08] J. Orcutt et al “Demonstration of an electronic photonic integrated circuit in a commercial scaled bulk CMOS process,” Conf. on Lasers and Electro-Optics, May 2008.

Bibliography

[Patel09] S. Patel "Rainbow Falls: Sun's Next Generation CMT Processor", *Hot Chips 2009*.

[Shacham07] A. Shacham et al "Photonic NoC for DMA communications in chip multiprocessors," Symp. on High Performance Interconnects, Aug. 2007.

[Shamim09] I. Shamim, Energy Efficient Links and Routers for Multi-Processor Computer Systems, M.S. Thesis, MIT

[Vangal07] S. Vangal et al., "80-tile 1.28 TFlops network-on chip in 65 nm CMOS," Int'l Solid-State Circuits Conf., Feb. 2007

[Wang03] H. Wang, L. Peh, and S. Malik, "Power-driven design of router microarchitectures in on-chip networks," *IEEE Micro-36, pp.105–116, 2003*

[Wentzlaff07] D. Wentzlaff et al "On-chip Interconnection Architecture of the Tile Processor," *IEEE Micro*, Volume 27, no. 5, pp.15 - 31 , Sept.-Oct. 2007.