Computer Architecture
Interconnect

Ting-Jung Chang

NYCU CS

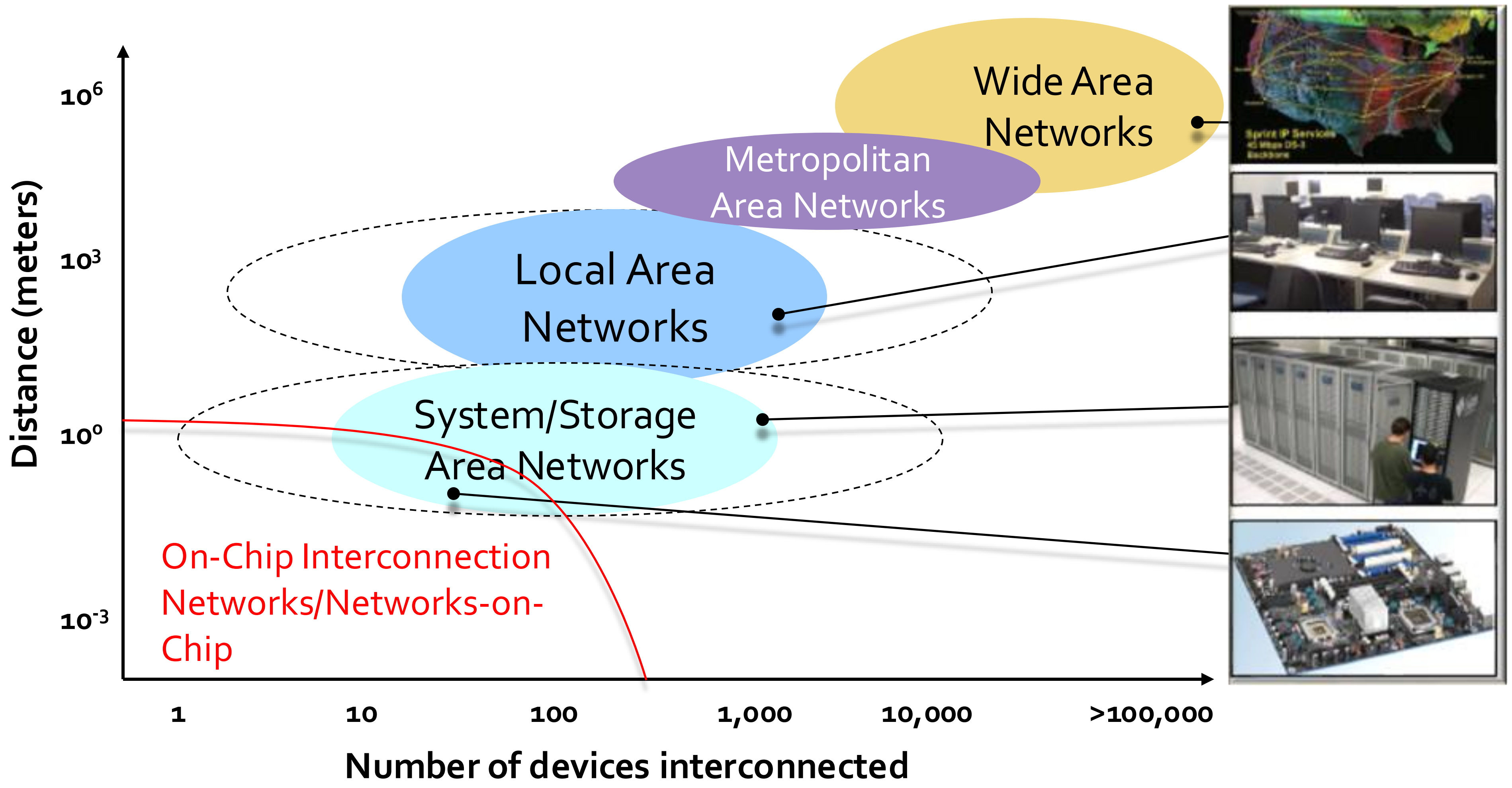
Course Administration

- PS6 solution released
- Final 12/16
  - Lecture 1-14 (Mostly lecture 7-14)
  - Ps1 – Ps6
  - 110 minutes closed-book exam (13:20~15:10)

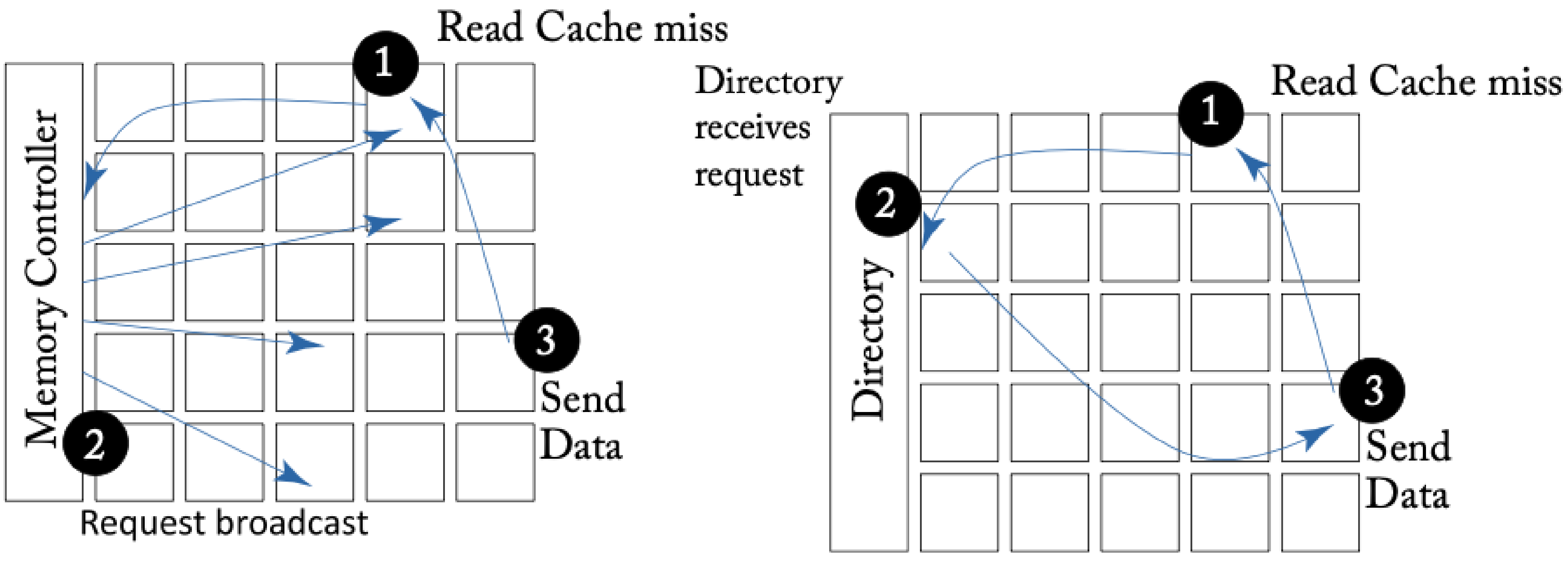
Agenda

- Introduction to networks
- Interconnect design
  - Topology
  - Routing & Router Microarchitecture
  - Switching & Flow control

Interconnection Network Domains



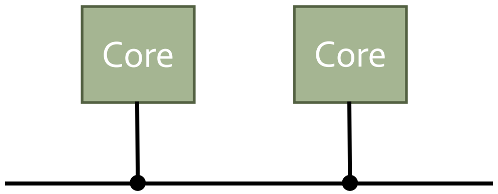
What's an on-chip network?



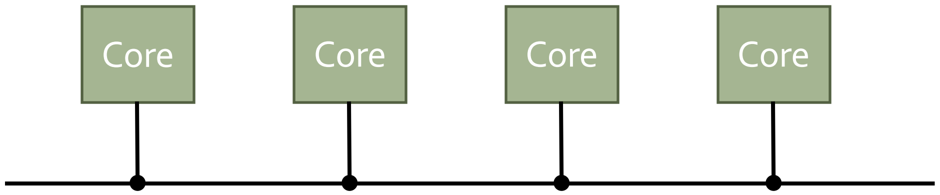
(a) Broadcast Protocol (b) Directory Protocol

Network transports cache coherence messages and cache lines between processor core

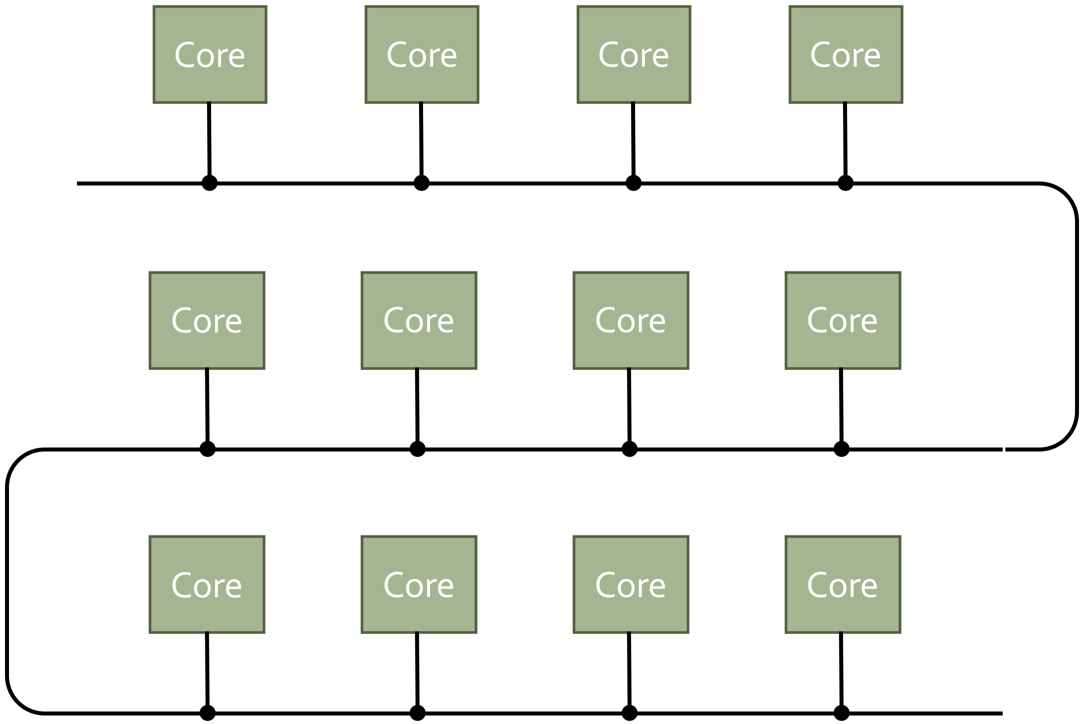
Overview of Interconnection Networks: Buses



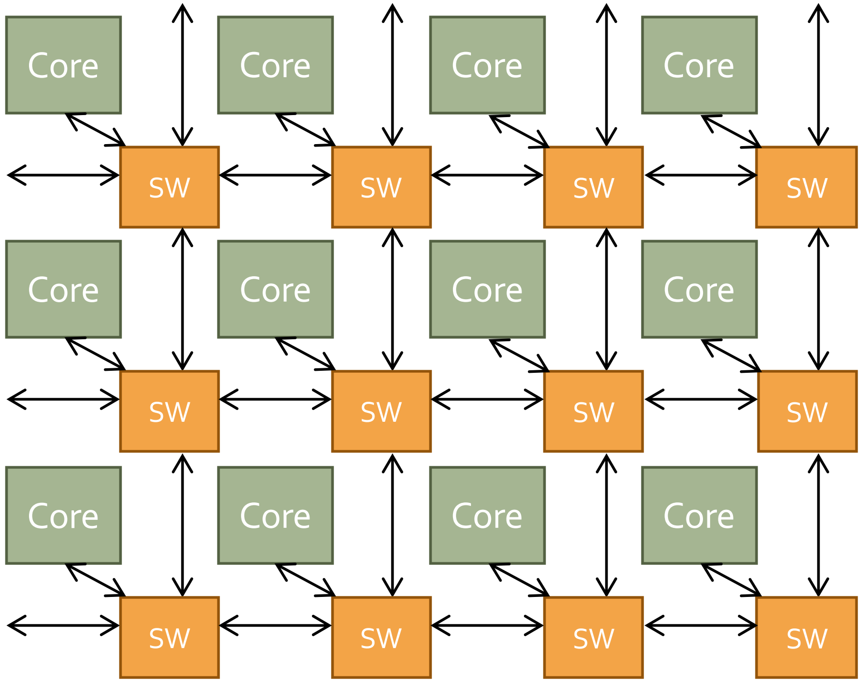
Overview of Interconnection Networks: Buses



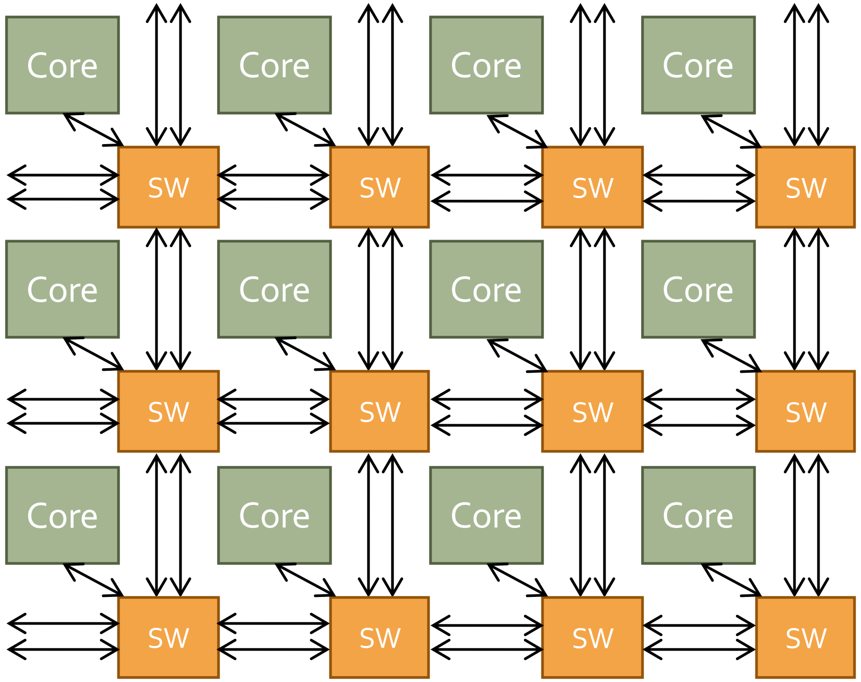
Overview of Interconnection Networks: Buses



Overview of Interconnection Networks: Point-to-point / Switched



Overview of Interconnection Networks: Point-to-point / Switched



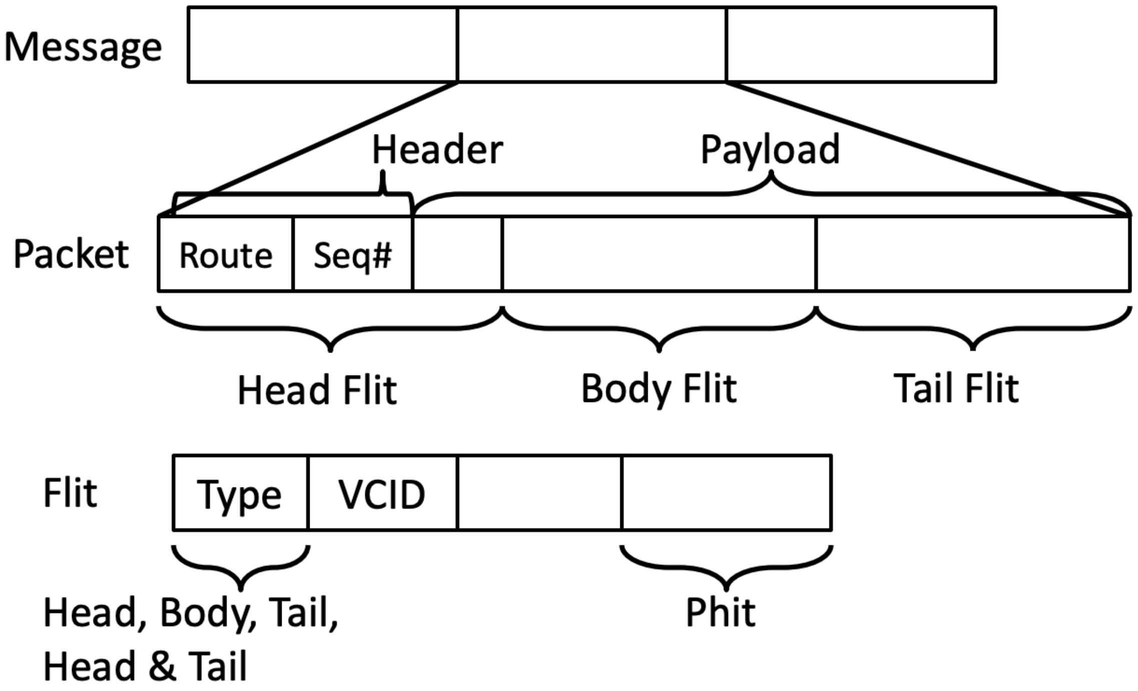
Agenda

- Introduction to networks
- Interconnect design
  - Topology
  - Routing & Router Microarchitecture
  - Switching & Flow control

Anatomy of a Message

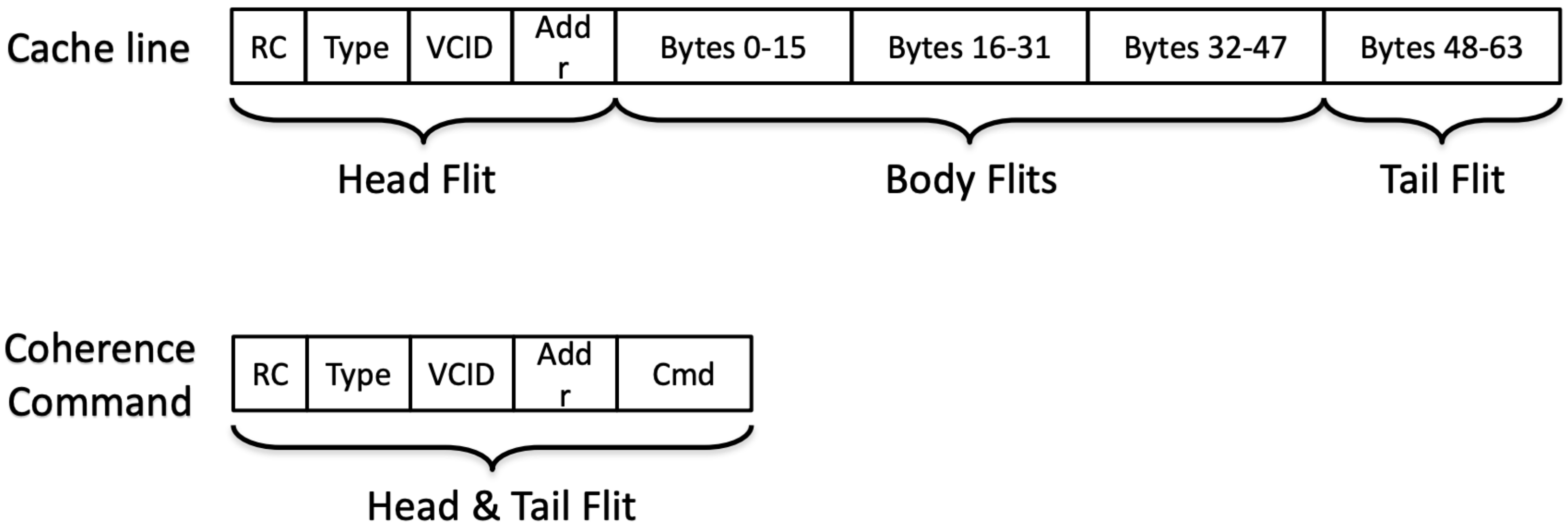
- Messages: composed of one or more packets
  - If message size is <= maximum packet size only one packet created
- Packets: composed of one or more flits
- Flit: flow control digit (Basic unit of flow control)
  - All flits in packet follow the same path
- Phit: physical transfer digit (Basic unit of data transferred in one clock)

Packets



- Off-chip: channel width limited by pins → requires phits
- On-chip: abundant wiring means phit size == flit size

Packets



- Packet contains destination/route information
  - Flits may not → all flits of a packet must take same route

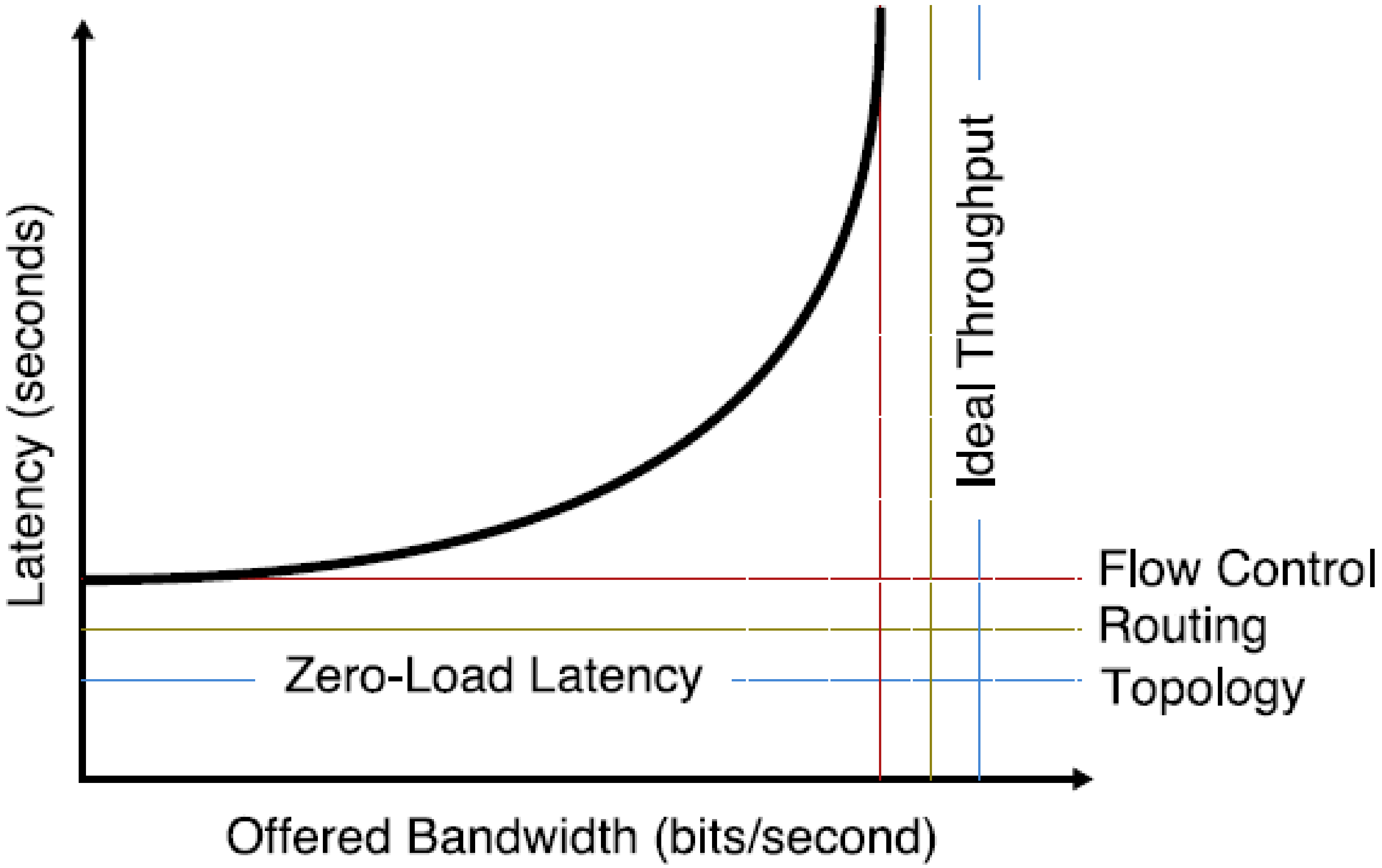
Network Architecture

- Topology
  - How to connect the nodes
  - ~ Road Network
- Routing
  - Which path should a message take
  - ~ Series of road segments from source to destination
- Flow Control
  - When does the message have to stop/proceed
  - ~ Traffic signals at end of each road segment
- Router Microarchitecture
  - How to build the routers
  - ~ Design of traffic intersection (number of lanes, algorithm for turning red/green)

Network Performance

- Bandwidth: The rate of data that can be transmitted over the network (network link) in a given time
- Latency: The time taken for a message to be sent from sender to receiver
- Bandwidth can affect latency
    - Reduce congestion
    - Messages take fewer Flits and Phits
- Latency can affect Bandwidth
    - Round trip communication can be limited by latency
    - Round trip flow-control can be limited by latency

Network Performance



Design-Time Metrics

- Degree – number of ports at a node
  - Proxy for area/energy cost
- Bisection Bandwidth – bandwidth crossing a minimal cut that divides the network in half
  - (Min # channels crossing two halves) * (BW of each channel)
  - Proxy for peak bandwidth
    - Can be misleading as it does not account for routing and flow control efficiency
    - At this stage, we assume ideal routing (perfect load balancing) and ideal flow control (no idle cycles on any channel)
- Diameter – maximum routing distance (number of links in shortest route)
  - Proxy for latency

Runtime Metrics

- Hop count (or routing distance)
  - Number of hops between a communicating pair
  - Depends on application and mapping
  - Average hop count or Average distance: average hops across all valid routes
- Channel load
  - Number of flows passing through a particular link
  - Depends on application and mapping
  - Maximum channel load determines throughput
- Path diversity
  - Number of shortest paths between a communicating pair
  - Can be exploited by routing algorithm
  - Provides fault tolerance

Agenda

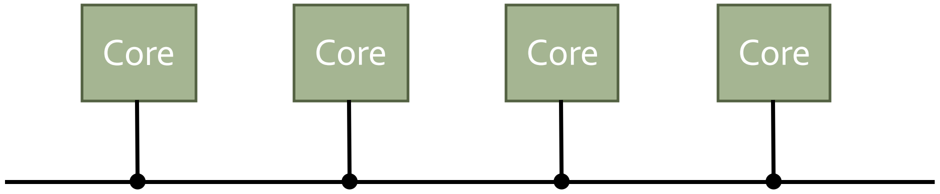
- Introduction to networks
- Interconnect design
  - Topology
  - Routing & Router Microarchitecture
  - Switching & Flow control

Topology Parameters

- Routing Distance: Number of links between two points
- Diameter: Maximum routing distance between any two points
- Average Distance
- Minimum Bisection Bandwidth: The bandwidth of a minimal cut though the network such that the network is divided into two sets of nodes
- Degree of a Router

Bus

- Pros
  - Cost-effective for small number of nodes
  - Easy to implement snoopy coherence
  - Most multicores with 4-6 cores use Buses
- Cons
  - Bandwidth! → Not scalable



- Diameter = ?
- Degree = ?
- Bisection BW = ?

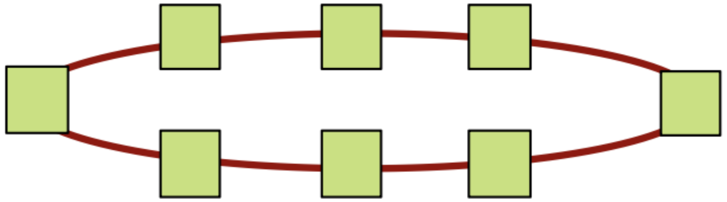
Topology Classification

- Direct
  - Each router is associated with a terminal node
  - All routers are sources and destinations of traffic
  - Example: **Ring, Mesh, Torus**
    - Most on-chip networks use direct topologies

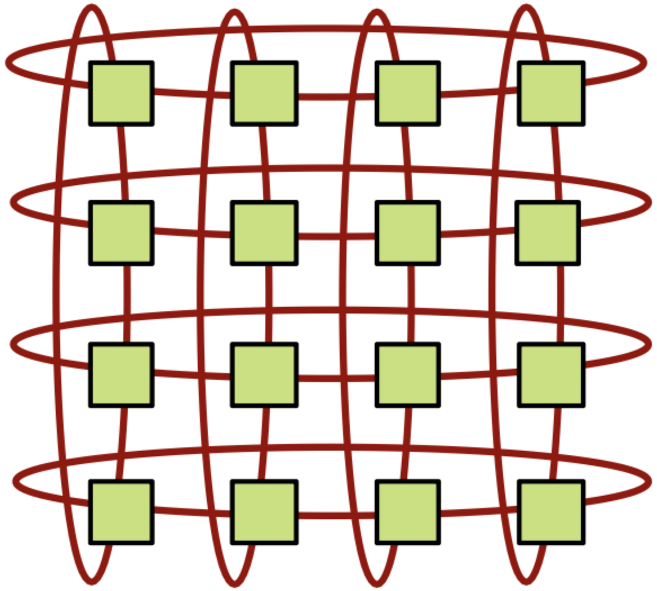
- Indirect
  - Routers are distinct from terminal nodes
  - Terminal nodes can source/sink traffic
  - Intermediate nodes switch traffic between terminal nodes
  - Examples: **Crossbar**, Butterfly, Clos, **Omega**, Benes, …

Rings and Torus

- Formally: k-ary n-cube
  - k^n network nodes
  - n-Dimensional grid with k nodes in each dimension



8-ary 1-cube

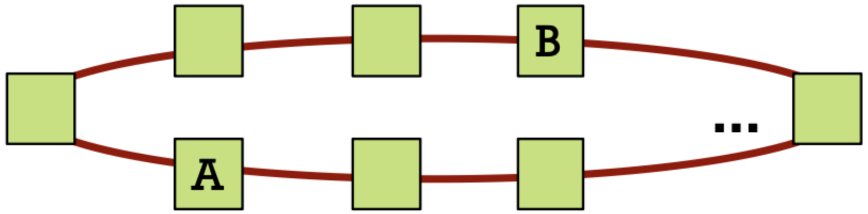


4-ary 2-cube

Ring

- Pros
  - Cheap: O(N) cost
  - Used in most multicores today
- Cons
  - High latency
  - Difficult to scale – bisection bandwidth remains constant
  - No path diversity
    - 1 shortest path from A to B

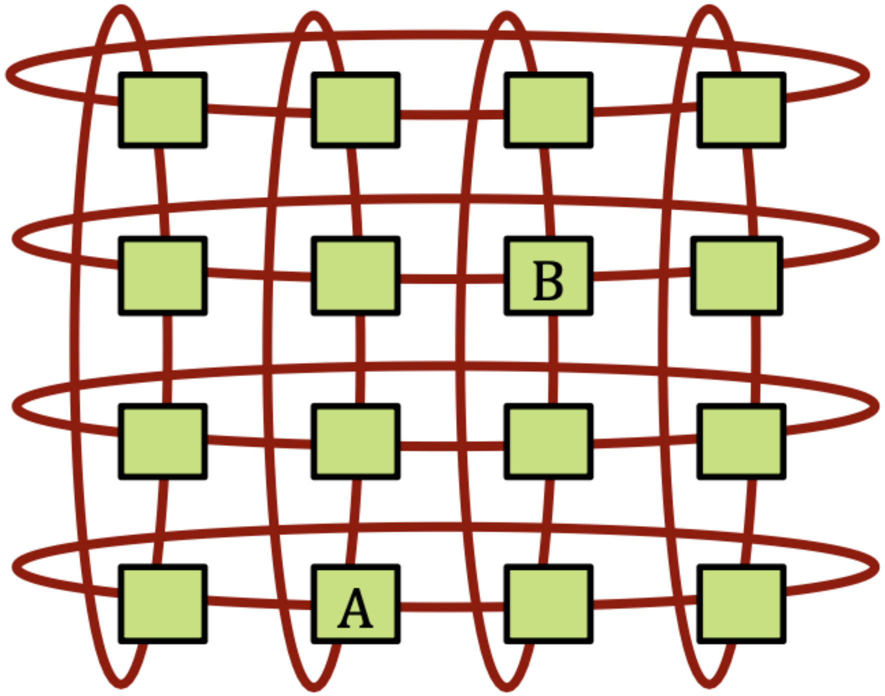
Diameter? N/2 (if even N)
Bisection BW? 2
Degree? 2



Torus

Diameter? $\sqrt{N}$
Bisection BW? $2\sqrt{N}$
Degree? 4

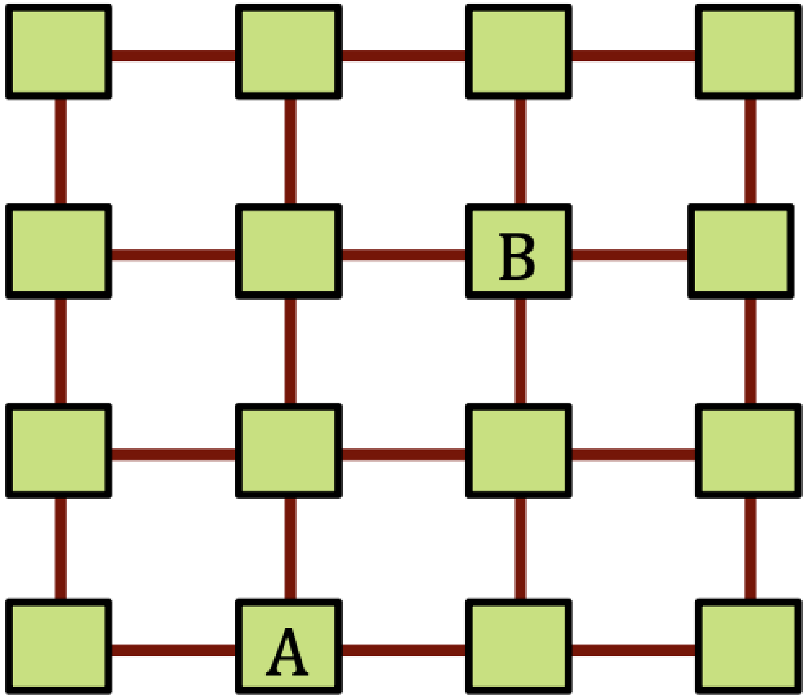
- Pros
  - O(N) cost
  - Exploit locality for near-neighbor traffic
  - High path diversity
    - 6 shortest paths from A to B
  - Edge symmetric
    - Good for load balancing
    - Same router degree
- Cons
  - Unequal link lengths
  - Harder to layout



Mesh

Diameter? $2(\sqrt{N}-1)$
Bisection BW? $\sqrt{N}$
Degree? 4

- Pros
  - O(N) cost
  - Easy to layout on-chip: regular and equal-length links
  - Path diversity
    - 3 shortest paths from A to B
- Cons
  - Not symmetric on edges
    - Performance sensitive to placement on edge vs. middle
    - Different degrees for edge vs. middle routers
  - Blocking, i.e., certain paths can block others (unlike crossbar)



Topology Classification

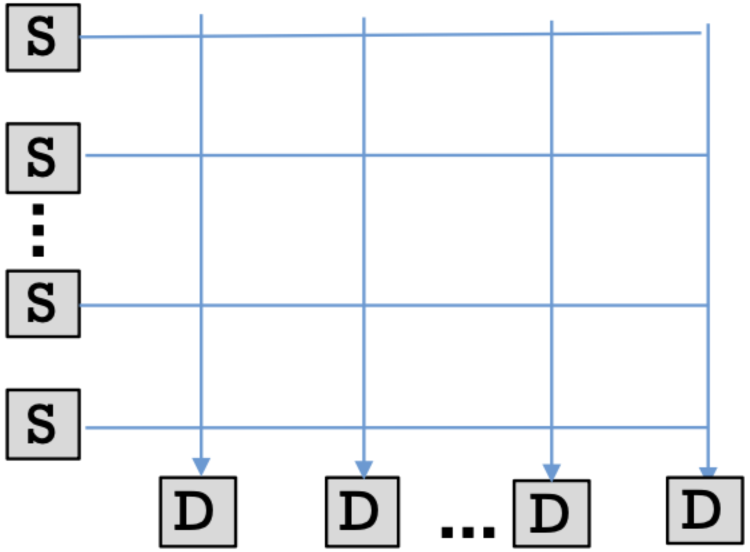
- Direct
  - Each router is associated with a terminal node
  - All routers are sources and destinations of traffic
  - Example: **Ring, Mesh, Torus**
    - Most on-chip networks use direct topologies

- Indirect
  - Routers are distinct from terminal nodes
  - Terminal nodes can source/sink traffic
  - Intermediate nodes switch traffic between terminal nodes
  - Examples: Crossbar, Butterfly, Clos, Omega, Benes, …

Crossbar

Diameter? 1
Bisection BW? N
Degree? 1

- Pros
  - Every node connected to all others (non-blocking)
- Low latency and high bandwidth
- Cons
  - Area and Power goes up
  - quadratically ($O(N^2)$ cost)
  - Expensive to layout
  - Difficult to arbitrate



Agenda

- Introduction to networks
- Interconnect design
  - Topology
  - Routing & Router Microarchitecture
  - Switching & Flow control

Routing Overview

- Discussion of topologies assumed ideal routing

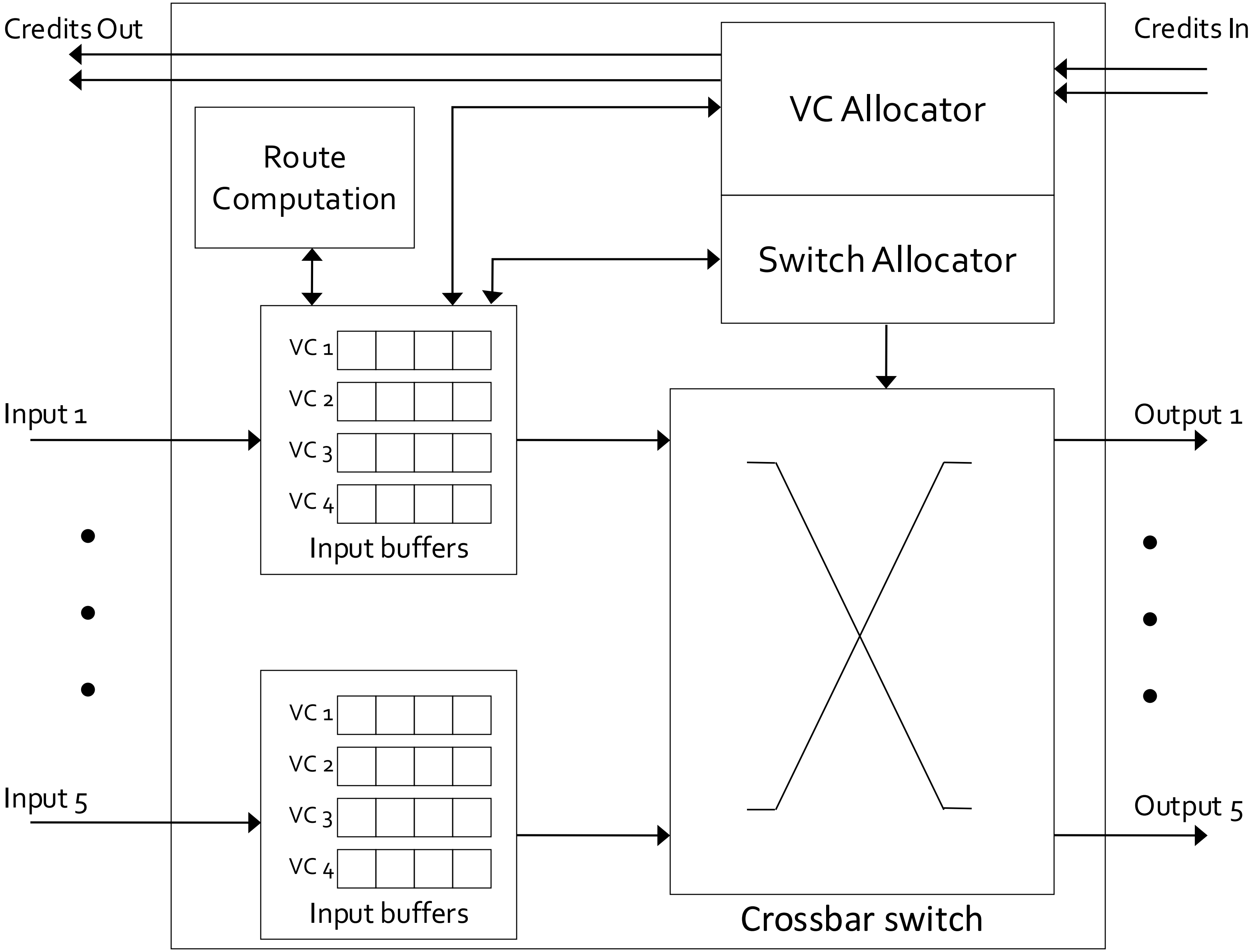
- In practice…
  - Routing algorithms are not ideal

- Goal: distribute traffic **evenly** among paths
  - Avoid hot spots, contention
  - More balanced → closer throughput is to ideal

- Keep complexity in mind

Routing

- Oblivious (routing path independent of state of network)
  - Deterministic
  - Non-Deterministic
- Adaptive (routing path depends on state of network)



Credits Out
Credits In
VC Allocator
Switch Allocator
Route Computation
VC 1
VC 2
VC 3
VC 4
Input buffers
Input 1
Input 5
Output 1
Output 5
VC 1
VC 2
VC 3
VC 4
Input buffers
Crossbar switch

Routing vs Flow Control

- Routing algorithm chooses path that packets should follow to get from source to destination
- Flow control schemes allocate resources (buffers, links, control state) to packets traversing the network

Agenda

- Introduction to networks
- Interconnect design
    - Topology
    - Routing & Router Microarchitecture
    - Switching & Flow control

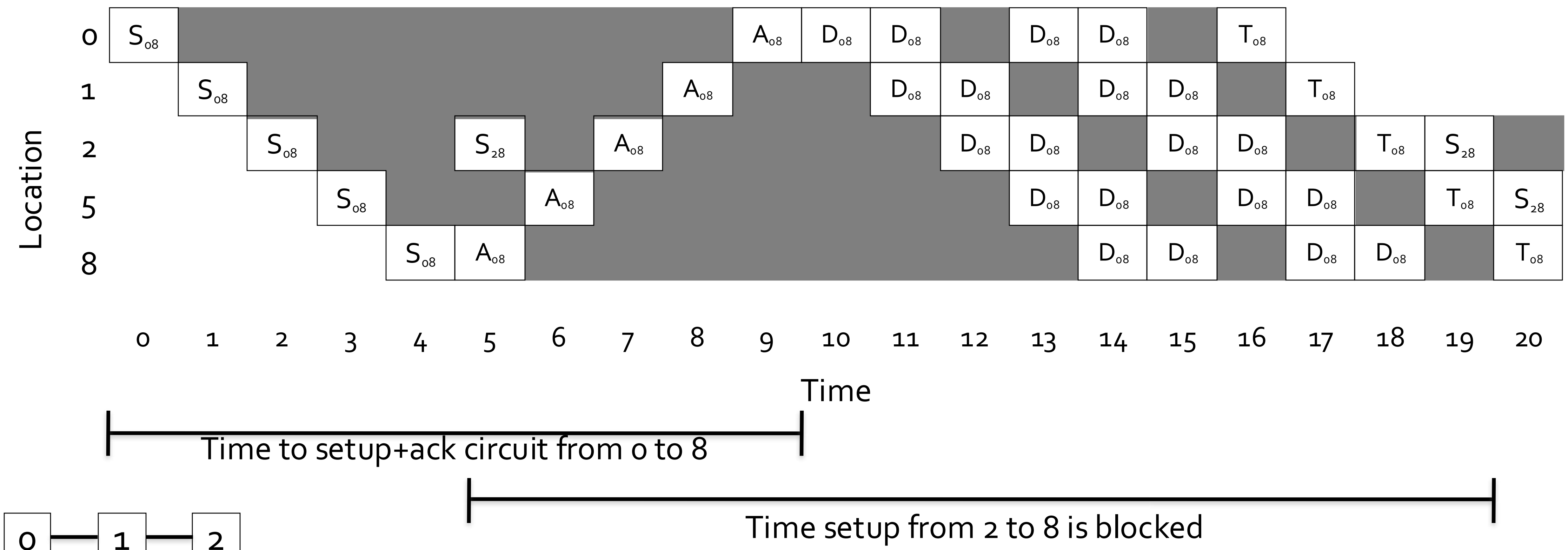
Switching Overview

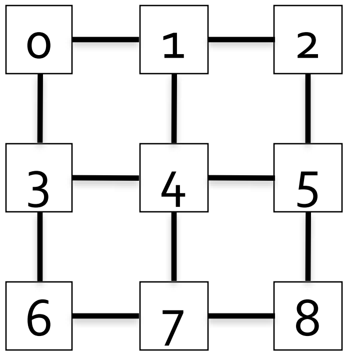
	Links	Buffers	Comments
Circuit-Switching	Messages	N/A (buffer-less)	Setup & Ack
Store and Forward	Packet	Packet	Head flit waits for tail
Virtual Cut Through	Packet	Packet	Head can proceed
Wormhole	Packet	Flit	HOL
Virtual Channel	Flit	Flit	Interleave flits of different packets

Circuit Switching

- Form a circuit from source to dest
- Probe to set up path through network
- Reserve all links Data sent through links
- Bufferless

Time-Space Diagram: Circuit-Switching





Buffered Routing

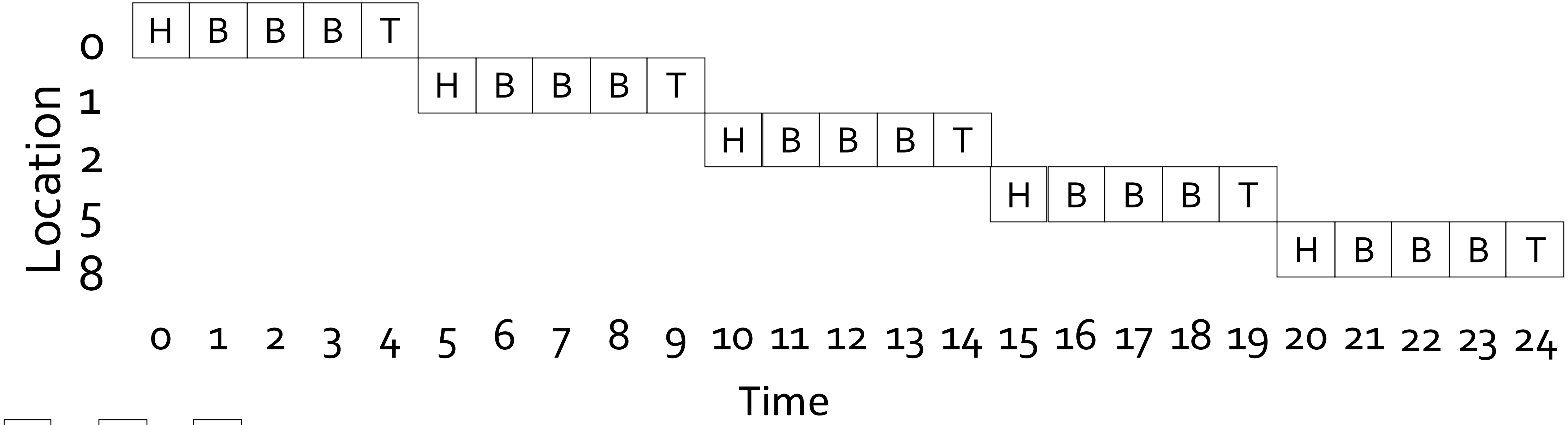
- Link-level flow control:
  - Given that you can't drop packets, how to manage the buffers?
  - When can you send stuff forward, when not?

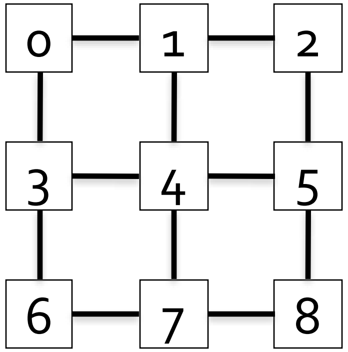
- Metrics of interest:
  - Throughput/Latency
  - Buffer utilization (turnaround time)

Store-and-Forward (packet-based, no flits)

- Strategy:
  - Make intermediate stops and wait until the entire packet has arrived before you move on
- Advantage:
  - Other packets can use intermediate links

Time-Space Diagram: Store and Forward

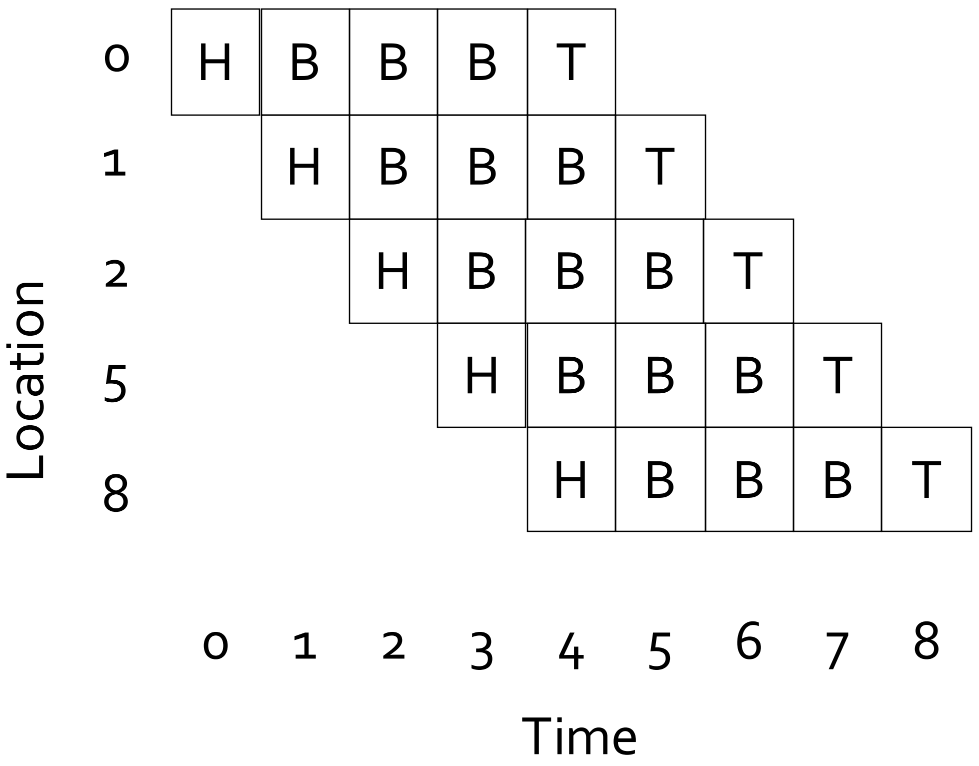


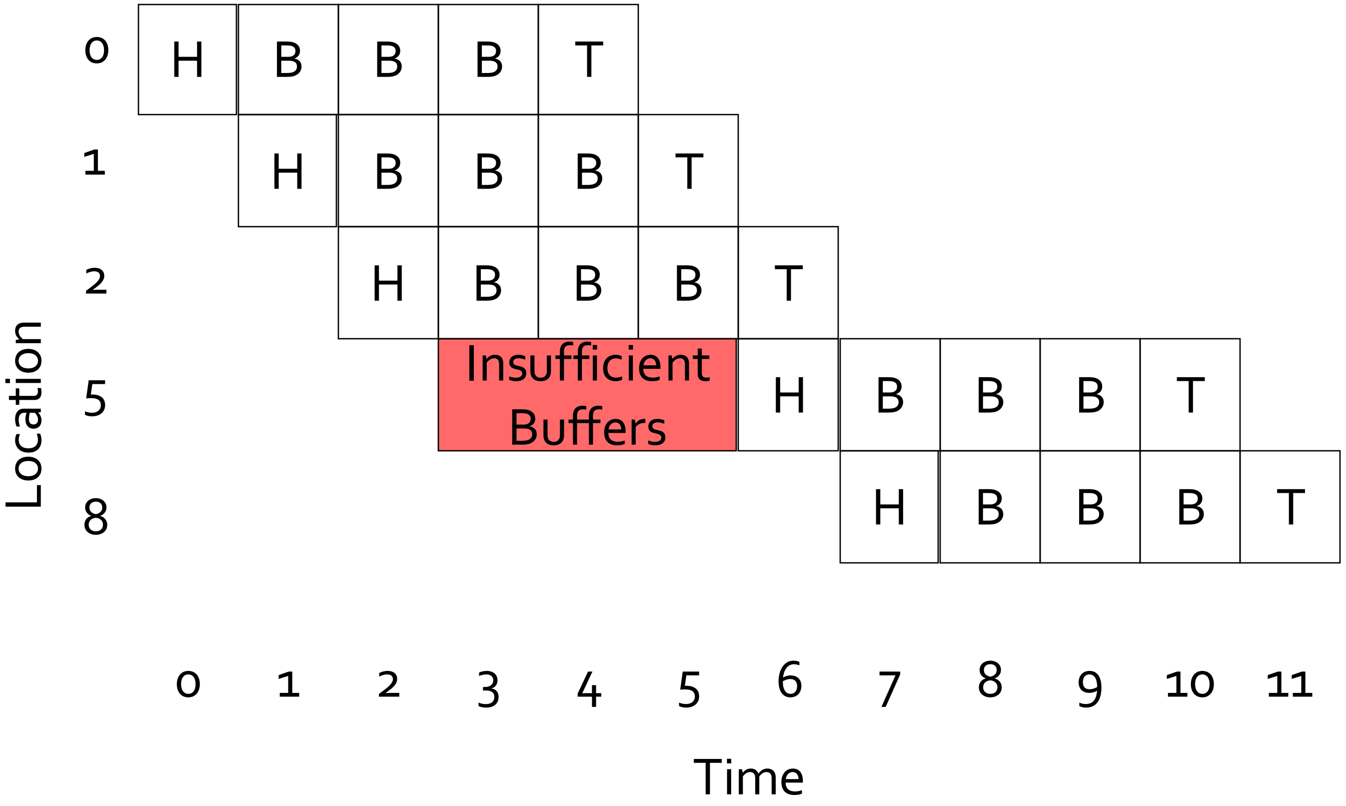


Virtual Cut-through (packet-based)

- Why wait till entire message has arrived at each intermediate stop?
- The head flit of the packet can dash off first
- When the head gets blocked, whole packet gets blocked at one intermediate node
- Used in Alpha 21364

Time-Space Diagram: VCT

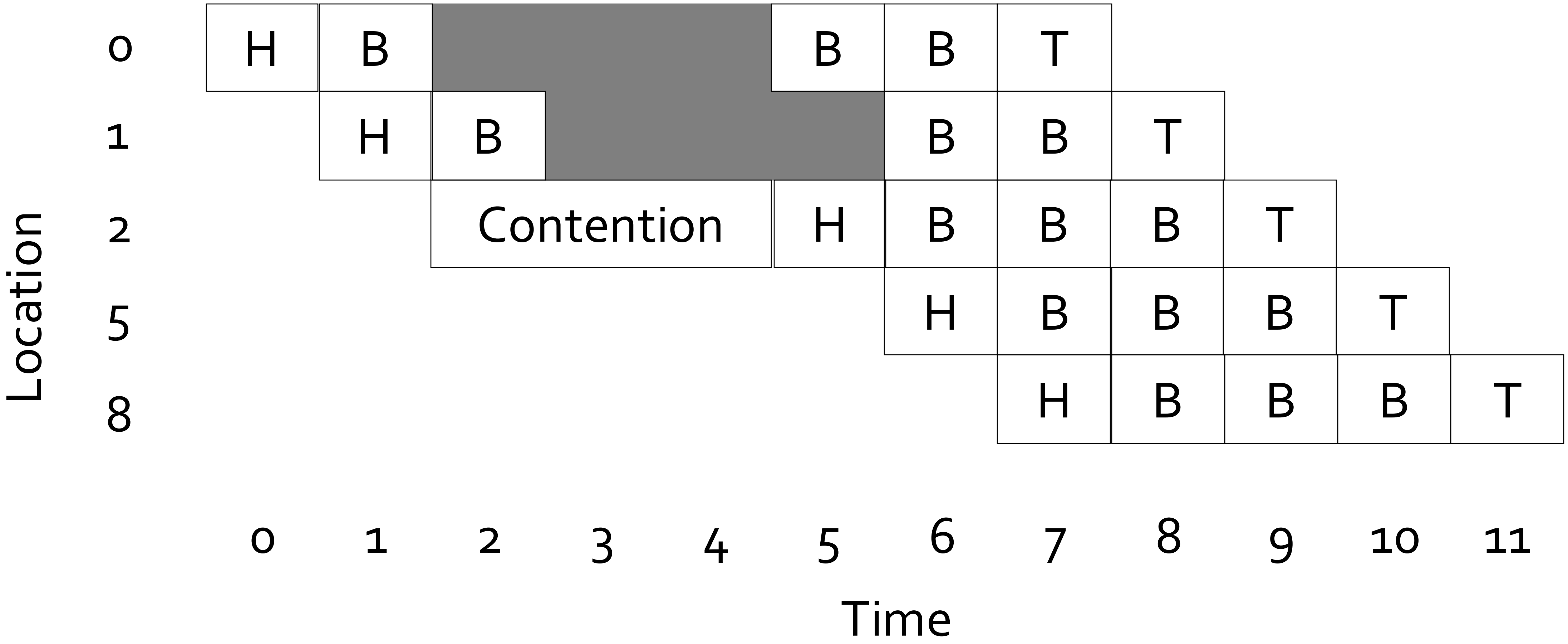




Wormhole (Flit-Buffer Flow Control)

- When a packet blocks, just block wherever the pieces (flits) of the message are at that time.
- Operates like cut-through but with channel and buffers allocated to flits rather than packets
  - Channel state (virtual channel) allocated to packet so body flits can follow head flit

Time-Space Diagram: Wormhole



Virtual-Channel (VC) Flow Control

- When a message blocks, instead of holding on to links so others can’t use them, hold on to virtual links
- Multiple queues in buffer storage
  - Like lanes on the highway
- Virtual channel can be thought of as channel state and flit buffers

Time-Space Diagram: Virtual-Channel

Switching Overview

	Links	Buffers	Comments
Circuit-Switching	Messages	N/A (buffer-less)	Setup & Ack
Store and Forward	Packet	Packet	Head flit waits for tail
Virtual Cut Through	Packet	Packet	Head can proceed
Wormhole	Packet	Flit	HOL
Virtual Channel	Flit	Flit	Interleave flits of different packets

Recap: Interconnect

- Introduction to networks
- Interconnect design
  - Topology
  - Routing
  - Switching
  - Flow control

Acknowledgements

- These slides contain material developed and copyright by:
  - Arvind (MIT)
  - Krste Asanovic (MIT/UCB)
  - Joel Emer (Intel/MIT)
  - James Hoe (CMU)
  - John Kubiatowicz (UCB)
  - David Patterson (UCB)
  - Christopher Batten (Cornell)
  - David Wentzlaff (Princeton)
- MIT material derived from course 6.823
- UCB material derived from course CS252
- Cornell material derived from course ECE 4750
- Princeton material derived from course ECE 475