Interconnection Networks

Alvin R. Lebeck
CPS 220

Admin

- **Homework #5 Due November 20**
- **Work on your projects…do a good job**

© Alvin R. Lebeck 1999

Interconnection Networks

- **Goal: Communication between computers**
- **Warning: Terminology-rich environment**
- **Focus on Networks for Parallel Computing**
  - **today's System Area Networks exhibit many of the same properties**

Terms

Network characterized by

- **Topology**
  - **physical structure of the graph**
- **Routing Algorithm**
  - **which paths through network can message fllow**
- **Switching Strategy**
  - **How data in message traverses its route**
  - **Circuit Switched vs Packet Switched**
- **Flow Control**
  - **When does a packet (or portions of it) move along its route**

Organization

- Given topology constructed by linking switches and network interfaces, must deliver packet from node A to node B
- Link: cable with connectors on each end
  - connect switches to other switches or network interfaces
- Switch: N inputs N outputs (degree N)
- Phit: Minimum # of bits physically moved across link in one cycle (Can pipeline on single wire)
- Flit: Minimum # of bits move across link as a single unit
- Packet: unit that requires routing information, some number of flits

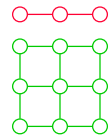
© Alvin R. Lebeck 1999

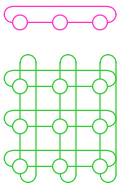
Topology

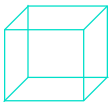
- Structure of the interconnect
- Determines
  - Switch Degree: number of links from a node
  - Diameter: number of links crossed between nodes on maximum shortest path
  - Average distance: number of hops to random destination
  - Bisection: minimum number of links that separate the network into two halves

© Alvin R. Lebeck 1999

Important Topologies



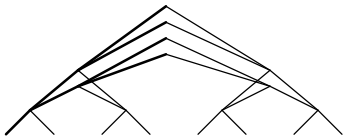


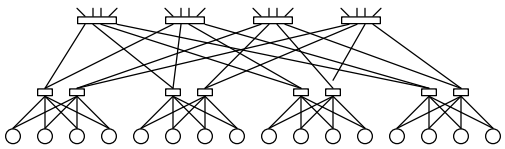


Type	Degree	Diameter	Ave Dist	Bisection	N = 1024 Diam	N = 1024 Ave D
1D mesh	2	N-1	2N/3	1		
2D mesh	4	$2(N^{1/2} - 1)$	$2N^{1/2} / 3$	$N^{1/2}$	63	21
3D mesh	6	$3(N^{1/3} - 1)$	$3N^{1/3} / 3$	$N^{2/3}$	~30	~10
nD mesh (N = k^n)	2n	$n(N^{1/n} - 1)$	$nN^{1/n} / 3$	$N^{(n-1) / n}$		
Ring	2	N / 2	N/4	2		
2D torus	4	$N^{1/2}$	$N^{1/2} / 2$	$2N^{1/2}$	32	16
k-ary n-cube (N = k^n)	2n	$n(N^{1/n})$ nk/2	$nN^{1/n}/2$ nk/4	$2k^{n-1}$	15	8 (3D)
Hypercube	n	n = LogN	n/2	N/2	10	5

Topologies (cont)

Type	Degree	Diameter	Ave Dist	Bisection	N = 1024 Diam	N = 1024 Ave D
2D Tree	3	$2Log_2 N$	$\sim 2Log_2 N$	1	20	~20
4D Tree	5	$2Log_4 N$	$2Log_4 N - 2/3$	1	10	9.33
kD	k+1	$Log_k N$				
2D fat tree	4	$Log_2 N$		N		
2D butterfly	4	$Log_2 N$	$Log_2 N$	N/2	20	20

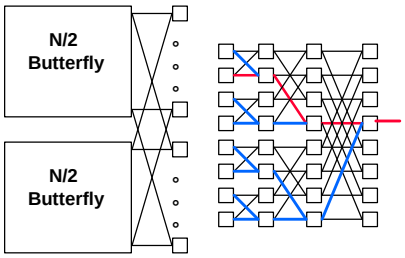




CM-5 Thinned Fat Tree

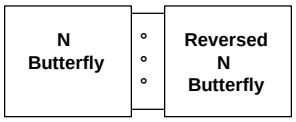
Butterfly

Multistage: nodes at ends, switches in middle



- All paths equal length
- Unique path from any input to any output
- Conflicts cause tree saturation

Benes Network

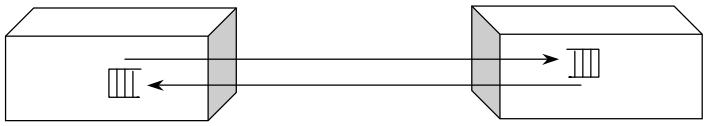


- Routes all permutations w/o conflict
- Notice similarity to Fat Tree (Fold in half)
- Randomization is major breakthrough

© Alvin R. Lebeck 1999

ABCs of Networks

- Starting Point: Send bits between 2 computers



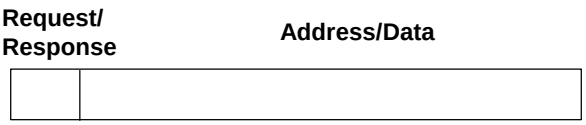
Queue on each end

- Can send both ways ("Bi-directional, Full Duplex")
- Rules for communication? "protocol"
  - Synchronous send
    - » Need Request & Response signaling
  - Name for standard group of bits sent: Packet

© Alvin R. Lebeck 1999

A Simple Example

- What is the packet format?
  - Fixed? (for HW Interpretation)
  - Number bytes?



0: Please send data from Address
1: Packet contains data corresponding to request

© Alvin R. Lebeck 1999

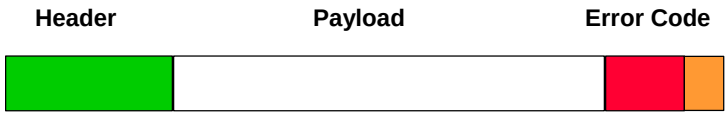
Questions About Simple Example

- What if more than 2 computers want to communicate?
  - Need node identifier field (destination) in packet
  - Routing and topology
- What if packet is garbled in transit?
  - Add error detection field in packet (e.g., CRC)
- What if packet is lost?
  - More elaborate protocols to detect loss (e.g., NAK, time outs)
- What if multiple processes/machine?
  - Dispatch
  - Queue per process
- Questions such as these lead to more complex protocols and packet formats

© Alvin R. Lebeck 1999

General Packet Format

- **Header**
  - routing and control information
- **Payload**
  - carries data (non HW specific information)
  - can be further divided (framing, protocol stacks…)
- **Error Code**
  - generally at tail of packet so it can be generated on the way out



© Alvin R. Lebeck 1999

Message v.s. Packet

- A Message may be composed of several packets
- Applications reason about messages
- Network transfers packets
- Small fixed size packets. Problems?
  Fragmentation and reassembly (SW overhead)
- Variable Size packets. Problems?
  Congestion

© Alvin R. Lebeck 1999

Packet Switched v.s. Circuit Switched

Circuit Switched

- Establish Route then Send Data
- Telephone system

Packet Switched

- Route each packet individually
- Delivery Guarantees
  - Reliable
  - In order, what if not?

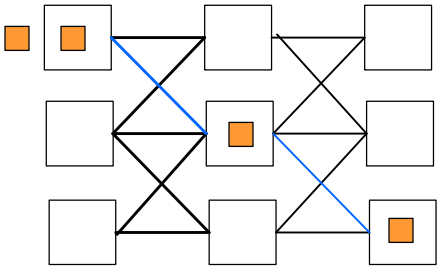
© Alvin R. Lebeck 1999

Routing

- Store-and-forward
- Cut-through
- Virtual cut-through
- Wormhole

© Alvin R. Lebeck 1999

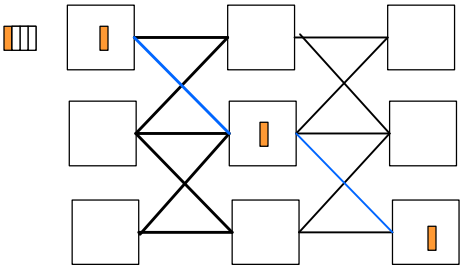
Store and Forward



- **Store-and-forward policy: each switch waits for the full packet to arrive in the switch before it is sent on to the next switch**

© Alvin R. Lebeck 1999

Cut Through



- **Cut-through routing: switch examines the header, decides where to send the message, and then starts forwarding it immediately**

© Alvin R. Lebeck 1999

Virtual Cut-Through

- What to do if output port is blocked?
- Lets the tail continue when the head is blocked, absorbing the whole message into a single switch.
  - Requires a buffer large enough to hold the largest packet.
- Degenerates to store-and-forward with high contention
- Compaq EV7 network

© Alvin R. Lebeck 1999

Wormhole

- When the head of the message is blocked the message stays strung out over the network
  - Potentially blocks other messages (needs only buffer the piece of the packet that is sent between switches).
  - CM-5 used it, with each switch buffer being 4 bits per port.
  - Myrinet uses it
- Interaction with Packet Size
- Can cause tree saturation…

© Alvin R. Lebeck 1999

Store and Forward vs. Cut-Through

- **Advantage**
  - **Latency reduces from function of:**

    Store and Forward

    number of intermediate switches times the size of the packet

    to

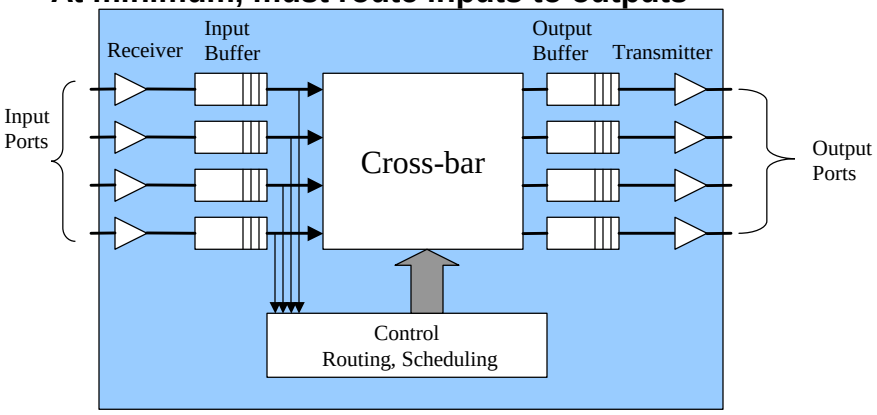
    Cut-Through

    time for 1st part of the packet to negotiate the switches + the packet size ÷ interconnect BW

© Alvin R. Lebeck 1999

Switches

- **At minimum, must route inputs to outputs**



VLSI makes it easier to create larger fully connected switches

© Alvin R. Lebeck 1999

Routing Algorithm

- How do I know where a packet should go?
- Arithmetic
- Source-Based
- Table Lookup
- Adaptive—route based on network state (e.g., contention)

© Alvin R. Lebeck 1999

Arithmetic Routing

- For regular topology, simple arithmetic to determine route
- 2D Mesh (Also called NEWS network)
  - packet header contains signed offset to destination
  - switch ++ or -- one field of header (x or y dimension)
  - when x == 0 and y == 0, then at correct processor
- Requires ALU in switch
- Must recompute CRC

© Alvin R. Lebeck 1999

Source Based and Table Lookup Routing

Source Based

- Source specifies output port for each switch in route
- Very Simple Switches
  - no control state
  - strip output port off header
- Myrinet uses this

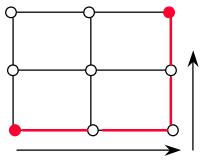
Table Lookup

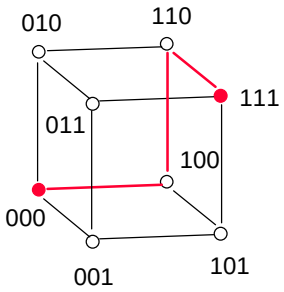
- Very Small Header, index into table for output port
- Big tables, must be kept up to date...

© Alvin R. Lebeck 1999

Deterministic v.s. Adaptive Routing

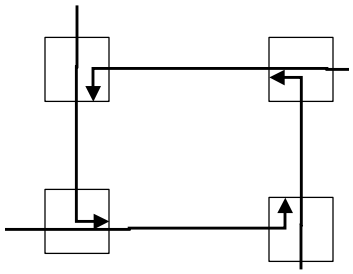
- Deterministic—follows a pre-specified route
  - mesh: dimension-order routing
    - » (x1, y1) -> (x2, y2)
    - » first Dx = x2 - x1,
    - » then Dy = y2 - y1,
  - hypercube: edge-cube routing
    - » X = x0x1x2 . . .xn -> Y = y0y1y2 . . .yn
    - » R = X xor Y
    - » Traverse dimensions of differing address in order
  - tree: common ancestor
- Adaptive—route determined by contention for output port





© Alvin R. Lebeck 1999

Deadlock



Deadlock Free Routing

- **Virtual Channels**
  - Not virtual cut-through
  - Add buffers so, flits of wormhole packets can be interleaved
- **Up*-Down***
  - Number switches: higher = farther away from processors
  - route up, make one turn, route down
- **Turn Model Routing**
  - Restrict order of turns
    - » West First
    - » North Last
    - » Negative First
  - Can increase number of hops

Congestion Control

- Packet switched networks do not reserve bandwidth; this leads to *contention*
- Solution: prevent packets from entering until contention is reduced (e.g., metering lights)
- Options:
  - End-to-end Flow Control
  - Link-level Flow Control

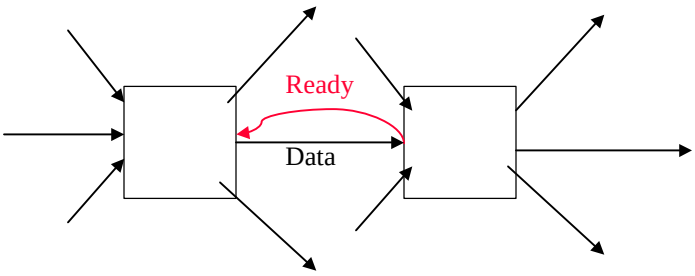
© Alvin R. Lebeck 1999

Link-Level Flow Control

- Packet discarding: If a packet arrives at a switch and there is no room in the buffer, the packet is discarded
  - no communication between switches, requires higher level protocol
- Flow control: between pairs of receivers and senders; use feedback to tell the sender when it is allowed to send the next packet
  - Choke packets: aka "rate-based"; Each packet received by busy switch in warning state sent back to the source via choke packet. Source reduces traffic to that destination by a fixed % (ATM Forum) Back-pressure: separate wires to tell to stop
  - high water, low water (stop & go back to source switch) (Myrinet)
  - Window: give the original sender the right to send N packets before getting permission to send more (overlap the latency of the interconnection with the overhead to send and receive a packet)

© Alvin R. Lebeck 1999

Link-Level Flow Control

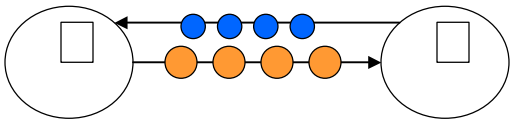


- Transfer single flit when receiver is ready
- Could have long links with many flits in flight

© Alvin R. Lebeck 1999

Credit-based (Window) Flow Control

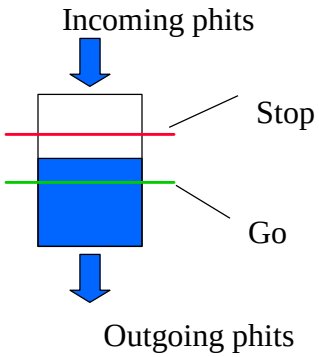
- **Receiver gives N credits to sender**
  - **sender decrements count**
  - **stops sending if zero**
  - **receiver sends back credit as it drains its buffer**
  - **bundle credits to reduce overhead**
- **Must account for link latency**



© Alvin R. Lebeck 1999

Water Level

- high water, low water
- stop & go back to source switch (Myrinet)
- can send redundant stop/go



© Alvin R. Lebeck 1999

Case Study Cray T3D

- 1024 switch nodes each connected to 2 processors
- 3D Torus, bidirectional, 300 MB/s
- Link: 16 bits, 8 control bits
- Variable size packet (multiple of 16 bits)
- Logical request & response networks
  - 2 virtual channels each for deadlock
- Stacked dimension routing
- Wormhole for large packets, virtual cut-through for small packets

© Alvin R. Lebeck 1999

IBM SP-2 (Vulcan)

- Switch has eight bidirectional 40 MB/s links
- Link: 8 data bits, 1 tag, 1 reverse flow-control
- Flit is 16 bits, phit is 8
- input FIFO + output FIFO + central buffer 128 8-byte segments

© Alvin R. Lebeck 1999