EECS 570
Lecture 21
Interconnects: Flow Control

Winter 2020
Prof. Satish Narayanasamy

http://www.eecs.umich.edu/courses/eecs570/

Topics to be covered

- Interfaces
- Topology
- Routing
- Flow Control
- Router Microarchitecture
Switching/Flow Control Overview

• Topology: determines **connectivity** of network

• Routing: determines **paths** through network

• Flow Control: determine **allocation** of resources to messages as they traverse network
  - Buffers and links
  - Significant impact on throughput and latency of network
Flow Control

- Control state records:
  - allocation of channels and buffers to packets
  - current state of packet traversing node

- Channel bandwidth advances flits from this node to next

- Buffers hold flits waiting for channel bandwidth
Packets

- Messages: composed of one or more packets
  - If message size is $\leq$ maximum packet size only one packet created

- Packets: composed of one or more flits

- Flit: flow control digit

- Phit: physical digit
  - Subdivides flit into chunks = to link width
Packets (2)

- Off-chip: channel width limited by **pins**
  - Requires phits

- On-chip: **abundant** wiring means phit size == flit size
• Packet contains destination/route information
  - Flits may not all flits of a packet must take same route
Types of Switching
Switching

• Different flow control techniques based on granularity

  □ **Message-based**: allocation made at message granularity (circuit-switching)

  □ **Packet-based**: allocation made to whole packets

  □ **Flit-based**: allocation made on a flit-by-flit basis
Message-Based Flow Control

• Coarsest granularity

• Circuit-switching
  - **Pre-allocates** resources across multiple hops
    - Source to destination
    - Resources = links
    - Buffers are not necessary
  - Probe sent into network to reserve resources
Circuit Switching

• Once probe sets up circuit
  - Message does not need to perform any routing or allocation at each network hop
  - Good for transferring large amounts of data
    - Can amortize circuit setup cost by sending data with very low per-hop overheads

• No other message can use those resources until transfer is complete
  - Throughput can suffer due setup and hold time for circuits
  - Links are idle until setup is complete
Circuit Switching Example

- Significant latency overhead prior to data transfer
  - Data transfer does not pay per-hop overhead for routing and allocation
Circuit Switching Example (2)

- When there is contention
  - Significant wait time
  - Message from 1 → 2 must wait
Time-Space Diagram: Circuit-Switching

Time to setup+ack circuit from 0 to 8

Time setup from 2 to 8 is blocked
Packet-based Flow Control

• Break messages into packets

• Interleave packets on links
  □ Better utilization

• Requires per-node **buffering** to store in-flight packets

• Two types of packet-based techniques
Store and Forward

• Links and buffers are allocated to entire packet

• Head flit waits at router until entire packet is received before being forwarded to the next hop

• Not suitable for on-chip
  - Requires buffering at each router to hold entire packet
    - Packet cannot traverse link until buffering allocated to entire packet
  - Incurs high latencies (pays serialization latency at each hop)
Store and Forward Example

- High per-hop latency
  - Serialization delay paid at each hop
- Larger buffering required

Total delay = 4 cycles per hop x 3 hops = 12 cycles
Time-Space Diagram: Store and Forward

![Time-Space Diagram](image-url)
Packet-based: Virtual Cut Through

- Links and Buffers allocated to entire packets

- Flits can proceed to next hop before tail flit has been received by current router
  - But only if next router has enough buffer space for entire packet

- Reduces the latency significantly compared to SAF

- But still requires large buffers
  - Unsuitable for on-chip
Virtual Cut Through Example

- Lower per-hop latency
- Large buffering required

Allocate 4 flit-sized buffers before head proceeds

Total delay = 1 cycle per hop x 3 hops + serialization = 6 cycles
Time-Space Diagram: VCT

```
 0 1 2 3 4 5 6 7 8
0 H B B B T
1 H B B B T
2 H B B B T
5 H B B B T
8 H B B B T
```

Location

Time
Virtual Cut Through

- Throughput suffers from inefficient buffer allocation
  - Cannot proceed because only 2 flit buffers available
Time-Space Diagram: VCT (2)

Location

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>H</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>H</td>
<td>B</td>
<td>B</td>
<td>B</td>
<td>T</td>
<td></td>
</tr>
</tbody>
</table>

Insufficient Buffers
Flit-Level Flow Control

• Help routers meet tight area/power constraints

• Flit can proceed to next router when there is buffer space available for that flit
  - Improves over SAF and VCT by allocating buffers on a flit-by-flit basis
Wormhole Flow Control

• Pros
  □ More efficient buffer utilization (good for on-chip)
  □ Low latency

• Cons
  □ Poor link utilization: if head flit becomes blocked, all links spanning length of packet are idle
    ○ Cannot be re-allocated to different packet
    ○ Suffers from head of line (HOL) blocking
Wormhole Example

- 6 flit buffers/input port

Red holds this channel: channel remains idle until red proceeds

Channel idle but red packet blocked behind blue

Buffer full: blue cannot proceed

Blocked by other packets
Time-Space Diagram: Wormhole

Location

0
1
2
5
8

H B B B T
H B B B T
Contention
H B B B B T
H B B B B T
H B B B B T

Time

0 1 2 3 4 5 6 7 8 9 10 11
Virtual Channels

• First proposed for deadlock avoidance
  □ We’ll come back to this

• Can be applied to any flow control
  □ First proposed with wormhole
Virtual Channel Flow Control

- Virtual channels used to combat HOL blocking in wormhole

- Virtual channels: **multiple** flit queues per input port
  - Share **same** physical link (channel)

- Link utilization improved
  - Flits on different VC can pass blocked packet
Virtual Channel Flow Control (2)
Virtual Channel Flow Control (3)

In1

\[
\begin{array}{ccccccc}
AH & A1 & A2 & A3 & A4 & A5 & AT \\
1 & 1 & 2 & 2 & 3 & 3 & 3 \\
\end{array}
\]

In2

\[
\begin{array}{ccccccc}
BH & B1 & B2 & B3 & B4 & B5 & BT \\
1 & 2 & 2 & 3 & 3 & 3 & 3 \\
\end{array}
\]

Out

\[
\begin{array}{ccccccccccc}
\end{array}
\]

A Downstream

\[
\begin{array}{ccccccc}
AH & A1 & A2 & A3 & A4 & A5 & AT \\
\end{array}
\]

B Downstream

\[
\begin{array}{ccccccc}
BH & B1 & B2 & B3 & B4 & B5 & BT \\
\end{array}
\]
Virtual Channel Flow Control (3)

- Packets compete for VC on flit by flit basis

- In example: on downstream links, flits of each packet are available every other cycle

- Upstream links throttle because of limited buffers

- Does not mean links are idle
  - May be used by packet allocated to other VCs
Virtual Channel Example

- 6 flit buffers/input port
- 3 flit buffers/VC
## Summary of techniques

<table>
<thead>
<tr>
<th></th>
<th>Links</th>
<th>Buffers</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit-Switching</td>
<td>Messages</td>
<td>N/A (buffer-less)</td>
<td>Setup &amp; Ack</td>
</tr>
<tr>
<td>Store and Forward</td>
<td>Packet</td>
<td>Packet</td>
<td>Head flit waits for tail</td>
</tr>
<tr>
<td>Virtual Cut Through</td>
<td>Packet</td>
<td>Packet</td>
<td>Head can proceed</td>
</tr>
<tr>
<td>Wormhole</td>
<td>Packet</td>
<td>Flit</td>
<td>HOL</td>
</tr>
<tr>
<td>Virtual Channel</td>
<td>Flit</td>
<td>Flit</td>
<td>Interleave flits of different packets</td>
</tr>
</tbody>
</table>
Deadlock
Deadlock

- Using flow control to guarantee deadlock freedom gives more flexible routing
  - Recall: routing restrictions needed for deadlock freedom

- If routing algorithm is not deadlock free
  - VCs can break resource cycle

- Each VC is time-multiplexed onto physical link
  - Holding VC implies holding associated buffer queue
  - Not tying up physical link resource

- Enforce order on VCs
Deadlock: Enforce Order

- All messages sent through VC 0 until cross dateline
- After dateline, assigned to VC 1
  - Cannot be allocated to VC 0 again
Deadlock: Escape VCs

• Enforcing order lowers VC utilization
  - Previous example: VC 1 underutilized

• Escape Virtual Channels
  - Have 1 VC that is deadlock free
  - Example: VC 0 uses DOR, other VCs use arbitrary routing function
  - Access to VCs arbitrated fairly: packet always has chance of landing on escape VC

• Assign different message classes to different VCs to prevent protocol level deadlock
  - Prevent req-ack message cycles
Buffer Backpressure
Buffer Backpressure

• Need mechanism to prevent buffer overflow
  - Avoid dropping packets
  - Upstream nodes need to know buffer availability at downstream routers

• Significant impact on throughput achieved by flow control

• Two common mechanisms
  - Credits
  - On-off
Credit-Based Flow Control

- Upstream router stores credit counts for each downstream VC

- Upstream router
  - When flit forwarded
    - Decrement credit count
  - Count == 0, buffer full, stop sending

- Downstream router
  - When flit forwarded and buffer freed
    - Send credit to upstream router
    - Upstream increments credit count
• Round-trip credit delay:
  - Time between when buffer empties and when next flit can be processed from that buffer entry
  - If only single entry buffer, would result in significant throughput degradation
  - Important to size buffers to tolerate credit turn-around
On-Off Flow Control

• Credit: requires upstream signaling for every flit

• On-off: decreases upstream signaling

• Off signal
  □ Sent when number of free buffers falls below threshold $F_{off}$

• On signal
  □ Sent when number of free buffers rises above threshold $F_{on}$
On-Off Timeline

- Less signaling but more buffering
  - On-chip buffers more expensive than wires

F\textsubscript{off} set to prevent flits arriving before t4 from overflowing

F\textsubscript{on} set so that Node 2 does not run out of flits between t5 and t8
## Buffer Utilization

<table>
<thead>
<tr>
<th>Cycle</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit count</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

### Head Flit

<table>
<thead>
<tr>
<th>VA/SA</th>
<th>ST</th>
<th>LT</th>
<th>BW</th>
<th>VA/SA</th>
<th>ST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Body Flit 1

<table>
<thead>
<tr>
<th>SA</th>
<th>ST</th>
<th>LT</th>
<th>BW</th>
<th>SA</th>
<th>ST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Credit (head)

<table>
<thead>
<tr>
<th></th>
<th>C</th>
<th>C-LT</th>
<th>C-Up</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Body Flit 2

<table>
<thead>
<tr>
<th>SA</th>
<th>ST</th>
<th>LT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Credit (body 1)

<table>
<thead>
<tr>
<th></th>
<th>C</th>
<th>C-LT</th>
<th>C-Up</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Tail Flit

<table>
<thead>
<tr>
<th>SA</th>
<th>ST</th>
<th>LT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Buffer Sizing

• Prevent backpressure from limiting throughput
  □ Buffers must hold # of flits ≥ turnaround time

• Assume:
  □ 1 cycle propagation delay for data and credits
  □ 1 cycle credit processing delay
  □ 3 cycle router pipeline

• At least 6 flit buffers
Actual Buffer Usage & Turnaround Delay

Flit arrives at node 1 and uses buffer

1

Credit propagation delay

Flit leaves node 1 and credit is sent to node 0

1

Credit pipeline delay

Node 0 receives credit, freed buffer reallocated to new flit

3

flit pipeline delay

New flit leaves Node 0 for Node 1

1

flit propagation delay

New flit arrives at Node 1 and reuses buffer
Flow Control Summary

• On-chip networks require techniques with lower buffering requirements
  - Wormhole or Virtual Channel flow control

• Avoid dropping packets in on-chip environment
  - Requires buffer backpressure mechanism

• Complexity of flow control impacts router microarchitecture