6.3 Buffer Space versus Number of Flows

- What to do when the number of flows is so large that it becomes impractical to allocate a separate flow-control “window” for each one of them
  - Dynamically sharing the buffer space among the flows: the ATLAS I and the QFC flow control protocol, and its evaluation
  - Regional Explicit Congestion Notification (RECN)
  - Request-Grant protocols: N. Chrysos, 2005
  - End-to-end congestion control via request-grant: N. Chrysos, 2006
  - Other buffer sharing protocols of the mid-90’s
  - Buffer memory cost versus transmission throughput cost in 1995
Buffered Switching Fabrics with Internal Backpressure

- Performance of OQ at the cost of IQ,
- Requires per-flow backpressure.
Cell Distribution Methods

• Aggregate traffic distribution:
  – Randomized routing (no backpressure)
  – Adaptive routing (indiscriminate backpressure)
  ⇒ load balancing on the long-term only

• Per-flow traffic distribution:
  – Per-flow round-robin (PerFlowRR)
  – Per-flow imbalance up to 1 cell (PerFlowIC)
  ⇒ accurate load balancing, on a shorter-term basis
Too many Flows

- $N^2$ per chip in the middle stage

Per-output Flow Merging

- Retains the benefits of per-flow backpressure
- $N$ flows per link, everywhere

- Re-sequencing needs to consider flows as they were before merging
- Freedom from deadlock
The “ATLAS I” Credit Flow-Control Protocol


• Features:
  – identically destined traffic is confined to a single “lane”
  – each “lane” can be shared by cells belonging to multiple packets

• As opposed to Wormhole Routing, where:
  – a “virtual circuit” (VC) is allocated to a packet and dedicated to it for its entire duration, i.e. until all “flits” (cells) of that packet go through
  – identically destined packets are allowed to occupy distinct VC’s
QFC-like Credit Protocol

- Quantum Flow Control (QFC) Alliance: proposed standard for credit-based flow control over WAN ATM links

- ATLAS I: similar protocol, adapted to
  - short links
  - hardware implem.

Number of Lanes \( L = \frac{B}{b} \)

Both kinds of credit are needed for a cell to depart (in ATLAS I: \( b=1 \))

Buffer
Saturation Throughput

64x64 fabric: 6-stage banyan using 2x2 elements
20-cell or 20-flit bursts, uniformly destined

Buffer Space (=Lanes) per Link
(cells or flits)

(B=L, with b=1)
Non-Hot-Spot Delay, in the Presence of Hot-Spot Destinations

non-hot-spot load = 0.2; 20-cell/flit bursts; 64x64 fabric: 6-stg banyan w. 2x2 el.

Number of Lanes \((L)\)

Delay (cell times)

(w with buffer space \(B=16\) cells or flits per link)
ATLAS I

- Single-chip ATM Switch with Multilane Backpressure
- 10 Gbit/s = 16×16 @ 622 Mb/s/port
- Shared Buffer
- 0.35 µm CMOS
- 1996-98, FORTH-ICS, Crete, GR
**Core:**

- Cell Buffer & Switching: 15%
- Header Pr., Rt'ng, Transl.: 20%
- Credit-based Flow Control: 15%
- Queue Pointer Manag'mnt: 17%
- Scheduling, Pop. Counts: 13%
- Ctrl/Mgt, Load Mon, Misc.: 20%
- Elastic buf., I/O Link Intf.: 13%

**GigaBaud Transceivers**

- Design Effort: 15.0 p-yrs
- Gates: 150 K
- FF: 44 K
- SRAM: 570 Kbits
- Area: 25% wiring
- Power: 25%

**Periphery:**

- Pads & Drivers: 10% 10%

**Area Power**

- SRAM: 570 Kbits
- Gates: 150 K
- FF: 44 K
- Cells: 15.0 p-yrs

**Credit-based Flow Control**

- 4% wiring

**Queue Pointer Manag'mnt**

- 4% wiring

**Ctrl/Mgt, Load Mon, Misc.**

- 4% wiring

**Elastic buf., I/O Link Intf.**

- 4% wiring

**Header Pr., Rt'ng, Transl.**

- 4% wiring

**Cell Buffer & Switching**

- 4% wiring

**Scheduling, Pop. Counts**

- 4% wiring

**Design Effort**

- 15.0 p-yrs

**225 mm² 9. W**

---

CS-534 - Copyright University of Crete
Backpressure Cost Evaluation, versus Alternatives

• Measure the cost of credit flow control in ATLAS & compare to:

• Alternatives, without internal backpressure in the fabric:
  – large buffers (off-chip DRAM) in all switches throughout the fabric, or
  – internal speedup in the fabric and output buffers

Making Large ATM Switches:
Switching Fabrics with Internal Backpressure

Switching Fabrics with Internal Backpressure

input queues (per-flow)

input interface

ATLAS

ATLAS

ATLAS

small, on-chip buffers

large, off-chip buffers (DRAM); total throughput = 2N

backpressure

backpressure

backpressure

small, on-chip buffers
Switching Fabrics without Backpressure 1: Large Buffers

large, off-chip buffers (DRAM); total throughput = 2 N logN
Switching Fabrics without Backpressure 2: Internal Speedup

speedup $s > 1$; under bursty or non-uniform traffic: $s \gg 1$...
Backpressure Cost/Benefit 1:
No Backpressure, Large Off-Chip Buffers

Core:

<table>
<thead>
<tr>
<th>Component</th>
<th>Gates</th>
<th>FF</th>
<th>SRAM</th>
<th>Area</th>
<th>Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell Buf. &amp; Switching</td>
<td>20%</td>
<td>20%</td>
<td>20%×0</td>
<td>10%×0.6</td>
<td>25%×0.5</td>
</tr>
<tr>
<td>Hdr, Rt'ng, Transl.</td>
<td></td>
<td>20%</td>
<td></td>
<td>10%×0</td>
<td>4%×0</td>
</tr>
<tr>
<td>Credit-b. Flow Ctrl</td>
<td>6%+</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q.Ptr, Sch, Ctrl, etc.</td>
<td>42%+</td>
<td>45%+</td>
<td>25%×0</td>
<td>6%+</td>
<td>25%+</td>
</tr>
<tr>
<td>Elastic buf., I/O Intf.</td>
<td></td>
<td></td>
<td></td>
<td>35%×2</td>
<td>55%×2</td>
</tr>
</tbody>
</table>

Periphery:

Off-Chip Communication Cost

CS-534 - Copyright University of Crete
Backpressure Cost/Benefit 2:
No Backpressure, Internal Speedup, Output Queues

**Core:**

<table>
<thead>
<tr>
<th></th>
<th>Gates</th>
<th>FF</th>
<th>SRAM</th>
<th>Area</th>
<th>Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell Buf. &amp; Switching</td>
<td>20% xS</td>
<td>20% xS</td>
<td>20%</td>
<td>10% ++</td>
<td></td>
</tr>
<tr>
<td>Hdr, Rt'ng, Transl.</td>
<td>20%</td>
<td>20%</td>
<td>50%</td>
<td>10%</td>
<td>25%</td>
</tr>
<tr>
<td>Credit-b. Flow Ctrl</td>
<td>20%</td>
<td>20%</td>
<td>50%</td>
<td>10% x0</td>
<td>25% xS</td>
</tr>
<tr>
<td>Q.Ptr, Sch, Ctrl, etc.</td>
<td>42%xS</td>
<td>45%xS</td>
<td>25%x0</td>
<td>10% x0</td>
<td>4%x0</td>
</tr>
<tr>
<td>Elastic buf., I/O Intf.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>25% wiring</td>
</tr>
</tbody>
</table>

**Periphery:**

<table>
<thead>
<tr>
<th></th>
<th>Off-Chip Communication Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>35%xS</td>
</tr>
<tr>
<td></td>
<td>55%xS</td>
</tr>
</tbody>
</table>
Regional Explicit Congestion Notification (RECN)

- Generalization & evolution of the ATLAS/QFC protocol
- Source-routing header describes path through fabric
- Intermediate or final link congestion sends back-notification
- All packets to congested link confined to a single lane
  - intermediate links identified via path component in header
  - entire trees of destinations in single lane (improvement over QFC)
  - equivalent of lane here called “Set-Aside Queue” (SAQ)
- VOQ’s replaced by Single (!) Input Queue + SAQ’s
  - dynamically create/delete SAQ’s
  - CAM assumed to match incoming pck header versus current SAQ’s
Request-Grant Protocols

• Consider a buffer feeding an output link, and receiving traffic from multiple sources:
• If credits are pre-allocated to each source, the buffer needs to be as large as one RTT-window per source;
• If credits are “held” at buffer and only allocated to requesting source(s) when these have something to transmit, then a single RTT-window suffices for all sources!
  ⇒ economize on buffer space at the cost of longer latency

• N. Chrysos, M. Katevenis: “Scheduling in Switches with Small Internal Buffers”, IEEE Globecom 2005, St. Louis, USA, Nov. 2005;
Buffer Space for Bounded Peak-to-Average Rate Ratio

- Assume $R_{peak}(i) / R_{average}(i) \leq PAR$ for all flows $i$ on a link
  - $R(i)$ is the rate (throughput) of flow $i$
  - $PAR$ is a constant: peak-to-average ratio bound
  - Interpretation: rate fluctuation is bounded by $PAR$
- Each flow $i$ needs a credit window of $RTT \cdot R_{peak}(i)$
- Buffer space for all flows is $\sum (RTT \cdot R_{peak}(i)) = RTT \cdot \sum (R_{peak}(i)) \leq RTT \cdot \sum (PAR \cdot R_{average}(i)) = PAR \cdot RTT \cdot \sum (R_{average}(i)) \leq PAR \cdot (RTT \cdot R_{link})$
  $\Rightarrow$ Allocate buffer space = $PAR$ number of “windows”
  When individual flow rates change, rearrange the allocation of buffer space between flows – but must wait for the buffer of one flow to drain before reallocating it (not obvious how to)
Dynamically Sharing the Buffer Space among Flows

• In order to depart, a packet must acquire both:
  – a per-flow credit (to guard against “buffer hogging”), and
  – a per-link credit (to ensure that the shared buffer does not overflow)

• Properly manage (increase or decrease) the per-flow window allocation based on traffic circumstances:
  – ATLAS and QFC protocols never change the per-flow window
  – H.T.Kung protocol moves allocations between flows (unclear how)
  – other idea: use two window sizes – a “full” one and a “small” one; use full-size windows when total buffer occupancy is below a threshold, use small-size windows (for all flows) above that point (flows that had already filled more than a small window will lose their allocation on packet departure) – C. Ozveren, R. Simcoe, G. Varghese: “Reliable and Efficient Hop-by-Hop Flow Control”, IEEE JSAC, May 1995.

Communication Cost versus Buffer/Logic Cost

- **On-Chip:** millions of transistors - hundreds of pins

- **Off-Chip:** data from Hot Interconnects '95 keynote speech, by A. Fraser, VP, AT&T Bell Labs:

  - speed of transmission line: 45 Mb/s
  - cost of long distance transmission: 45 $/mile/month
  - speed of signal propagation: 7 microsec/mile
  - round-trip window size: 79 bytes/mile
  - cost of 16 MByte DRAM: 1000 $
  - cost of window size memory: 0.5 cents/mile
  - investment write-down period: 36 months
  - cost of queue mem. per month: 0.014 cents/mile/month
  - ratio: transmission/memory cost: 330,000 to 1
Per-Connection Queueing & FC:

How many ``Windows'' of Buffer Space?

- \( \text{windowSize}(VC_i) = \text{RTT} \times \text{peakThroughput}(VC_i) \)
  \(\Rightarrow\) \( \text{windowSize}(VC_i) < \) or \(<<\) \( \text{windowL} := \text{RTT} \times \text{throughput}(\text{Link}) \)
- \( \text{cost}(\text{Link}) \approx 330,000 \times \text{cost}(\text{windowL}) \)
- Lossy flow control usually operates the network with goodput reaching up to 70 - 80 \% of link throughput
- Lossless flow control operates up to 98 - 100 \% link utilization
- The 20-30 \% extra utilization with lossless FC is worth approx. 10 to 100 thousand \( \text{windowL} \)'s worth of extra buffer memory
  \(\Rightarrow\) if lossless flow control can yield its link utilization advantage with less than a few tens of thousands of \( \text{windowL} \)'s of extra buffer memory, then lossless flow control is a clear win
- Indeed, lossless FC can do that, even with quite less buffer space...