Improving the Throughput of Asynchronous On-chip Networks with SDM

Wei Song and Doug Edwards
The Advanced Processor Technologies Group (APT)
School of Computer Science
The University of Manchester
{songw, doug}@cs.man.ac.uk
Outline

• Introduction
  – Network-on-Chips (NoCs)
  – Flow control: wormhole, virtual channel (VC) and spatial division multiplexing (SDM)

• SDM router
  – Implementation
  – Area and speed model

• Speculation of a VC router
  – Area and speed model

• Performance analysis
  – Latency accurate SystemC models
Network-on-Chips

Switch
Allocator
IP0
PxP
W
W
OP0
IP(P-1) OP(P-1)

header
data
data
tail

wormhole
Virtual Channel (VC)

Blocked flits are buffered in virtual channels; therefore, links are available to other flits.
Spatial Division Multiplexing (SDM)

Switch Allocator

IP₀  IP₀(P-1)  →  MPxMP  →  OP₀  OP₀(P-1)
W/M  W/M

Saturation Throughput

\[ th_{\text{wormhole}} = 0.67 \]
\[ th_{SDM} (M=4) = 0.83 \]
VC vs. SDM

- VC
  - Extra virtual channels (buffer)
  - An extra VC allocator
  - Increased crossbar
  - ANoC, QoS NoC, MANGO, QNoC

- SDM
  - Increased crossbar plus extra control logic
  - No asynchronous implementation
Input/Output Buffer

\[ A = 2.5WAC_L \]
Input/Output Buffer

\[ A_{IB} = M \left[ L \left( 2.5 \frac{W}{M} A_C + A_{EOF} \right) + A_{RC} + A_{CTL} \right] \]

\[ A_{OB} = 2.5WA_C + MA_{EOF} \]
Crossbar

\[ A_{CB} = \left( \frac{2W}{M} + 2 \right) (2M^2P^2 - MP)A_g \]
Switch Allocator

Switch Allocator

\[ A_A = M^2 P^2 A_{arb} \]
Area Consumption

- **Wormhole**
  \[
  A_{IB,WH} = L(2.5W A_C + A_{EOF}) + A_{RC} + A_{CTL}
  \]
  \[
  A_{OB,WH} = 2.5W A_C + A_{EOF}
  \]
  \[
  A_{CB,WH} = (2W + 2)(2P^2 - P)A_g
  \]
  \[
  A_{A,WH} = P^2 A_{arb}
  \]

- **SDM**
  \[
  A_{IB,SDM} = M[L(2.5 \frac{W}{M} A_C + A_{EOF}) + A_{RC} + A_{CTL}]
  \]
  \[
  A_{OB,SDM} = 2.5W A_C + MA_{EOF}
  \]
  \[
  A_{CB,SDM} = \left(\frac{2W}{M} + 2\right)(2M^2 P^2 - MP)A_g
  \]
  \[
  A_{A,SDM} = M^2 P^2 A_{arb}
  \]
Area Consumption

<table>
<thead>
<tr>
<th>Component</th>
<th>WH</th>
<th>err(%)</th>
<th>SDM</th>
<th>err(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Buffers</td>
<td>14,303</td>
<td>0.0</td>
<td>21,995</td>
<td>-0.4</td>
</tr>
<tr>
<td>Output Buffers</td>
<td>5,935</td>
<td>0.0</td>
<td>6,000</td>
<td>1.7</td>
</tr>
<tr>
<td>Crossbar</td>
<td>4,356</td>
<td>0.0</td>
<td>21,744</td>
<td>-0.2</td>
</tr>
<tr>
<td>Switch Allocator</td>
<td>772</td>
<td>78.2</td>
<td>22,208</td>
<td>-0.9</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>25,366</td>
<td>2.4</td>
<td>71,956</td>
<td>-0.3</td>
</tr>
</tbody>
</table>

P=5, L=2, W=32, M=4
Critical Cycle

Switch Allocator

IP₀ OP₀

IP₀(W/M) OP₀(W/M)

IP₀(IP₀(P-1)) OP₀(OP₀(P-1))

MPxMP

Control

Input Buffer

Crossbar

Output Buffer

The University of Manchester

Advanced Processor Technologies Group
The School of Computer Science
Critical Cycle

\[ T = 4t_C + 4t_{CB} + 2t_{CD} + 2t_{AD} + t_{CTL} \]

\[ t_C = \begin{cases} l_C + k_C (P + 1) & \text{wormhole}, \\ l_C + k_C (MP + 1) & \text{SDM}. \end{cases} \]

\[ t_{CB} = \begin{cases} l_{CB} + k_{CB} \log_2 (P) & \text{wormhole}, \\ l_{CB} + k_{CB} \log_2 (MP) & \text{SDM}. \end{cases} \]

\[ t_{AD} = \begin{cases} l_{AD} + k_{AD} (2W + 1) & \text{wormhole}, \\ l_{AD} + k_{AD} \left(\frac{2W}{M} + 1\right) & \text{SDM}. \end{cases} \]

\[ t_{CD} = \begin{cases} l_{CD} + l_C \log_2 \left(\frac{W}{2}\right) + k_{CD} P & \text{wormhole}, \\ l_{CD} + l_C \log_2 \left(\frac{W}{2M}\right) + k_{CD} MP & \text{SDM}. \end{cases} \]
# Critical Cycle

<table>
<thead>
<tr>
<th></th>
<th>WH</th>
<th>err</th>
<th>SDM</th>
<th>err(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>cycle period</td>
<td>4.25</td>
<td>2.6</td>
<td>4.15</td>
<td>-3.4</td>
</tr>
<tr>
<td>router latency</td>
<td>2.29</td>
<td></td>
<td>2.49</td>
<td></td>
</tr>
<tr>
<td>routing calculation</td>
<td>0.44</td>
<td></td>
<td>0.51</td>
<td></td>
</tr>
<tr>
<td>switch allocation</td>
<td>0.78</td>
<td></td>
<td>3.21</td>
<td></td>
</tr>
<tr>
<td>$t_C$</td>
<td>0.22</td>
<td>-9.1</td>
<td>0.34</td>
<td>-5.9</td>
</tr>
<tr>
<td>$t_{CB}$</td>
<td>0.16</td>
<td>1.3</td>
<td>0.26</td>
<td>-3.8</td>
</tr>
<tr>
<td>$t_{CD}$</td>
<td>0.79</td>
<td>7.6</td>
<td>0.57</td>
<td>4.2</td>
</tr>
<tr>
<td>$t_{AD}$</td>
<td>0.57</td>
<td>6.1</td>
<td>0.27</td>
<td>-0.4</td>
</tr>
</tbody>
</table>

$P=5, \ L=2, \ W=32, \ M=4$
VC Router

\[
A_{IB,VC} = MA_{IB,WH} \approx MA_{IB,SDM}
\]

\[
A_{OB,VC} = A_{OB,WH} \approx A_{OB,SDM}
\]

\[
A_{CB,VC} = (2MP^2 - P)(2W + 2)A_g \approx A_{CB,SDM}
\]

\[
A_{A,VC} = (M^2P^2 + MP)A_{arb} \approx A_{A,WH} + A_{A,SDM}
\]
VC Router

\[ t_{C,VC} = t_{C,WH} < t_{C,SDM} \]

\[ t_{CD,VC} = l_{CD} + l_c \log_2(W / 2) + k_{CD}MP > t_{CD,WH} > t_{CD,SDM} \]

\[ t_{AD,VC} = t_{AD,WH} > t_{AD,SDM} \]

\[ t_{CB,VC} = t_{CB,WH} < t_{CB,SDM} \]

cycle period = 5.23 ns

routing calculation = 0.44 ns

VC allocation = 3.21 ns

switch allocation = 0.78 ns
SystemC model

- Latency accurate SystemC models
- Wormhole, SDM, VC
- 8x8, 5 ports, XY routing
- 32-bit, 4 VCs/virtual circuits
Average Frame Latency

L=2, W=32, FL=64
VC router with L=2 suffers from credit loop stall.
Both SDM and SDMCS outperform VC.
Wormhole, SDM and SDMCS have constant data transmission latency.
Payload Size and Distance

All routers approach the maximal throughput with longer payload length. FL=64 Byte shows 90% maximal throughput. Throughput decreases with the increasing hop count. SDM shows better through even in the 8-hop case.
Gain = \frac{throughput}{area}
Data Width

\[ \text{Gain} = \frac{\text{throughput}}{\text{area}} \]
Number of VCs

![Graph showing the number of VCs against injected traffic](image)
Conclusions

• Both VC and SDM improve throughput.

• SDM achieves better throughput performance and area to throughput gain than VC.

• SDM has the potential ability to support hard delay guaranteed services.
Thanks!

Question?