97-1089: Modifications to Appendix B and Sections 3.1.7 and 3.2.7 of Testing Baseline Text on Scalable Configurations

Arjan Durresi, Raj Jain, Gojko Babic, Justin Dolske

Raj Jain is now at
Washington University in Saint Louis
Jain@cse.wustl.edu
http://www.cse.wustl.edu/~jain/
- Why these modifications?
- Methodology
- Examples
Scalable Configurations

- ATM testing equipment are expensive.
- Scalable Configurations permit to simulate the desired basic configuration using a limited number of generators.
- But there are many ways to set up the scalable connections configurations and the results could vary with the set up.
A Sample 4-to-4 Configuration

- Different implementations could provide different results.
- P1-P2-P3-P4-P1  Four module crossings
- P1-P3-P4-P2-P1  Two module crossings
Problem w/ Current Text

- Some switches set only bidirectional VCs
  Can’t have the same VCI on the same port for two VCs.
VCC Chain

- Performance testing requires setting up connections between ports of the switch.
- Some connections are internal through the switch fabric and others are external through wires or fibers.
- An external connection between two switch ports is referred in this appendix as a wire W.
- The sequence of concatenated connections (internal and external) is called a VCC Chain.
- The proposed algorithm permits to create standard VCC Chains for any number of generators and any number of ports ⇒ Scalable and basic (both)
Example

- 4-to-4 configuration with one generator

One of six possible VCC chains:

P1 W1 W2 W3 P1
VCC Chain Implementation

- Implementation of External Connections
  1. Numbering the ports
  2. Identifying the ports connected to generators and analyzers
  3. Numbering the wires
- Implementation of Internal Connections
### 1. Port Numbering

<table>
<thead>
<tr>
<th>Module #</th>
<th>Group 1</th>
<th>Group 2</th>
<th>Group 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>P1 P3 P5 P7 P9 P11 P13 P14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2)</td>
<td>P2 P4 P6 P8 P10 P12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3)</td>
<td>P15 P16 P17 P18 P19 P20 P21 P22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4)</td>
<td>P23 P24 P25 P26 P27 P28</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Need to group modules by technology and speed
- This port numbering helps creating VCC chains that cross modules using a simple algorithm
2. Generator/Analyzer Ports

- Identifying the ports connected to the generator and/or analyzers
  - Avoid having only one port left over in a group. (That port cannot be connected externally to any other port)
  - This condition does not apply if the switch allows loopbacks.
- Note: The algorithm works with loopbacks also.
3. Numbering the Wires

- In each group start with the first output port available. Connect it to the next port whose input is available. (Note: With loopbacks, the output of a port can be connected to the input of the same port. The rest of the methodology is same.)

- Continue until all ports have been connected.

- Numbered the wires sequentially as W1, W2, …with the restriction that the end of wire Wi and the beginning of W(i+1) must be different ports.

- May need to skip some wires and include them in the next round.
Example

- **Straight 7-to-7, 1 Gen.**

  Background Traffic
  Generator 2

  Foreground Traffic
  Generator 1
Algorithm

\[
\begin{align*}
f &= 1 \\
&\text{for } (k = 1 \text{ to } r, \text{ step } 1) \\
&\quad \{ \text{if}(k>1) \quad f = 1 + \sum_{d=1}^{k-1} NW(d) \}
\end{align*}
\]

\[
\begin{align*}
&\text{for } (j = 1 \text{ to } m, \text{ step } 1) \\
&\quad \{ \text{if}(j>1) \quad f = \text{mod}*(CH(1,j-1,k)+1, TNW)\}; \\
&\quad \text{for } (i = 1 \text{ to } NW(k), \text{ step } 1) \\
&\quad \quad \{ \\
&\quad \quad \quad CH(i,j,k)=W(f); \\
&\quad \quad \quad f = \text{mod}*(f+1, TNW); \\
&\quad \quad \quad } \text{end for } i \\
&\quad \} \text{end for } j \\
&\} \text{end for } k.
\]
Algorithm Rules

- Each chain generally goes sequentially from wire i to wire i+1 unless the wire has already been fully used by other chains. Use modulo N arithmetic.
- Multiple Chains/Generator: Each new VCC chain is obtained from the previous one shifting by one its wire number.
- Multiple Generators: **Divide** the wires between the generators. Each generator will start its traffic from its wires.
Performance testing requires two kinds of virtual channel connections (VCCs): foreground VCCs (traffic that is measured) and background VCCs (traffic that simply interferes with the foreground traffic).

We need in throughput measurements foreground traffic, and both foreground and background traffic in latency measurements.
Throughput and Latency Measurements (Cont)

- Foreground traffic in Latency measurements uses only two ports, one source and one destination.
- For scalable configurations in Latency Measurements, foreground and background traffic share ports in opposite directions.
8-to-8 Straight
8-to-8 Straight, 1 Gen.

VCC chain is: P1-W1-W2-W3-W4-W5-W6-W7-P1.

Throughput
Foreground Traffic

The Ohio State University
8-to-2 Partial Cross Throughput Foreground
8-to-2 Partial Cross, 2 Gen.

Throughput

Foreground Traffic

Gen. 1
1) P1-W1-W2-W3-P1
2) P1-W2-W3-W4-P1

Gen. 2
1) P2-W4-W5-W6-P2
2) P2-W5-W6-W1-P2
7-to-2 Partial Cross, 2Gen.

<table>
<thead>
<tr>
<th>IN</th>
<th>OUT</th>
<th>Foreground Traffic</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>P1</td>
<td></td>
</tr>
<tr>
<td>P2</td>
<td>P2</td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td>P3</td>
<td></td>
</tr>
<tr>
<td>P4</td>
<td>P4</td>
<td></td>
</tr>
<tr>
<td>P5</td>
<td>W1</td>
<td></td>
</tr>
<tr>
<td>P6</td>
<td>W4</td>
<td></td>
</tr>
<tr>
<td>P7</td>
<td>W2</td>
<td></td>
</tr>
<tr>
<td>P8</td>
<td>W5</td>
<td></td>
</tr>
</tbody>
</table>

Gen.1
1) P1-W1-W2-W3-P2
2) P1-W2-W3-W4-P2

Gen.2
1) P3-W4-W5-P3
2) P3-W5-W1-P3
Modifications to Sections 3.1.7 and 3.2.7

- Existing Figure 3.3: 8-to-8 straight configuration, 1 Gen.
New Figure 3.3
Modifications to Sections 3.1.7 and 3.2.7 (Cont.)

- Similarly, change Figure 3.4 and 3.5
- Replace “loopbacks” by “wires”
- Exchange “w” and “n” for consistency
  Throughout the document: $n = \# \text{ of ports.}$
Summary

- New Methodology:
  - Allows both loopback and non-loopback external connections.
  - Allows any number of generators.
  - Can be used for both scalable and basic configurations.
  - Algorithm can be implemented as a computer program.
Motion

- Adopt the text of 97-1089 as Appendix B of Performance Testing Baseline Text.
- Adopt the appropriate modifications to Section 3.1. and 3.2.7 of the Baseline text.