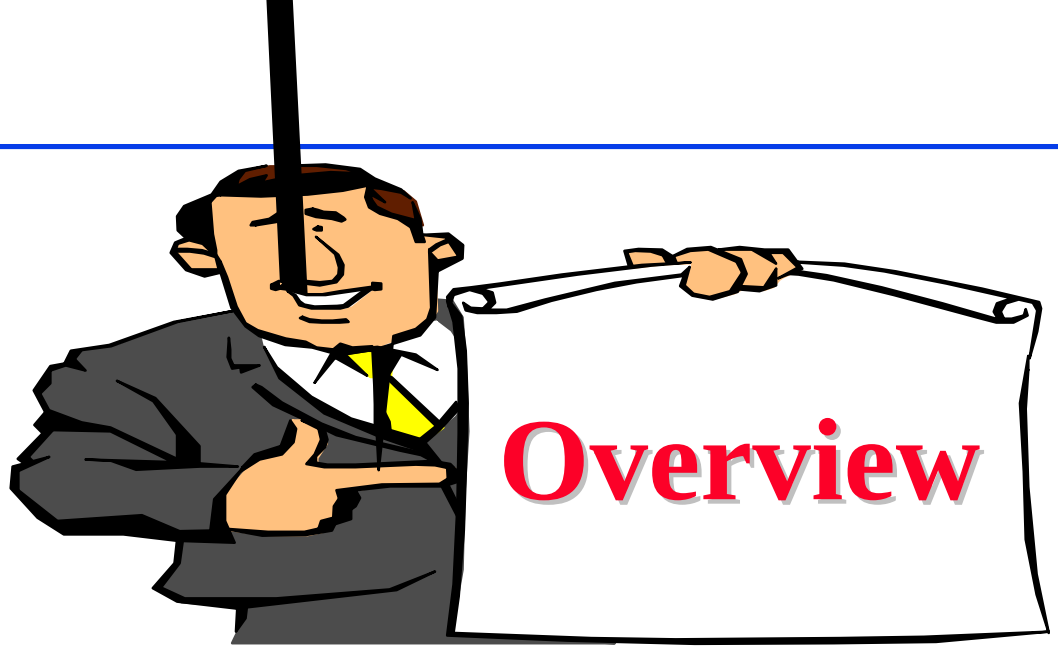
# 97-1089: Modifications to Appendix B and Sections 3.1.7 and 3.2.7 of Testing Baseline Text on Scalable Configurations

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/**



# Overview

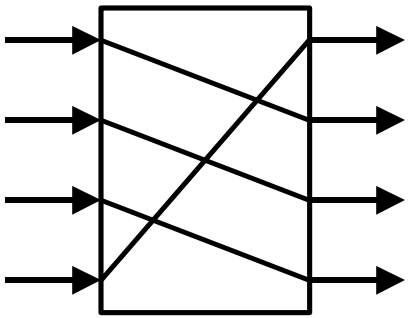
- Why these modifications?
- Methodology
- Examples

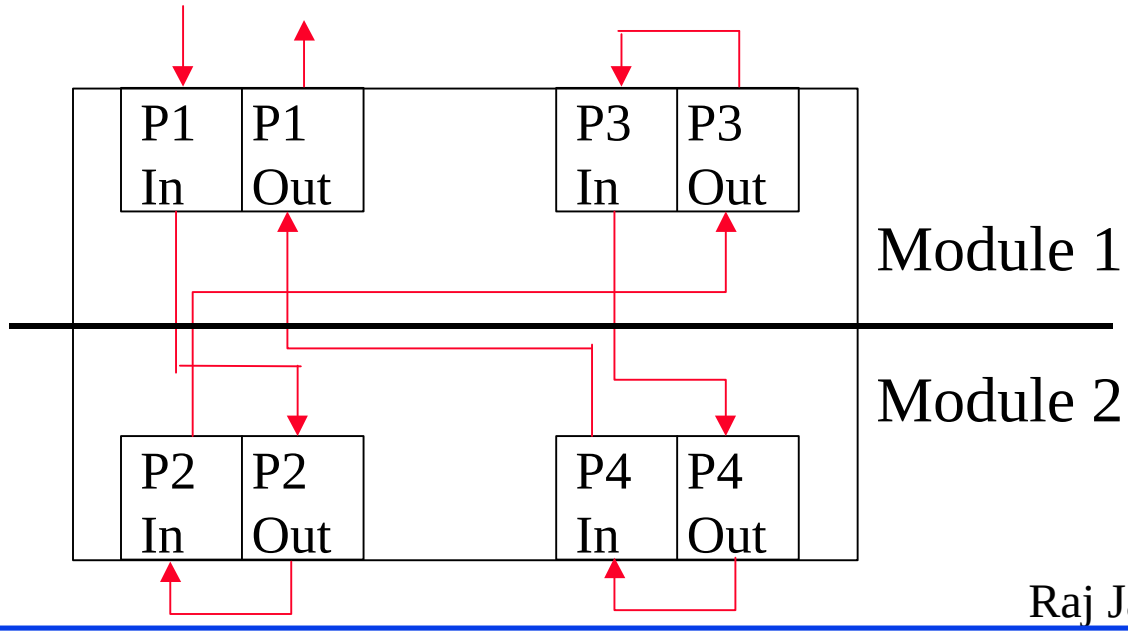
# Scalable Configurations

- 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

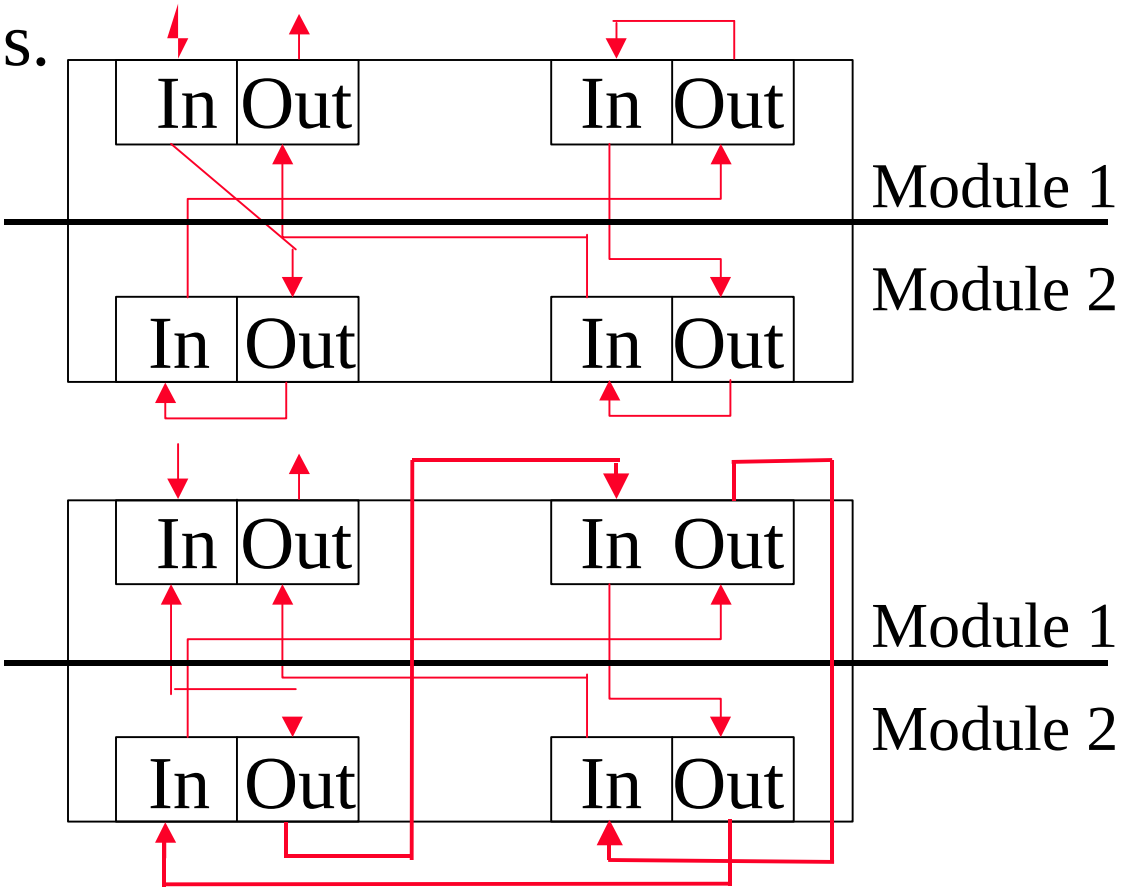
- Different implementations could provide different results.
- P1-P2-P3-P4-P1 Four module crossings
- P1-P3-P4-P2-P1 Two module crossings





# Problem w/ Current Text

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

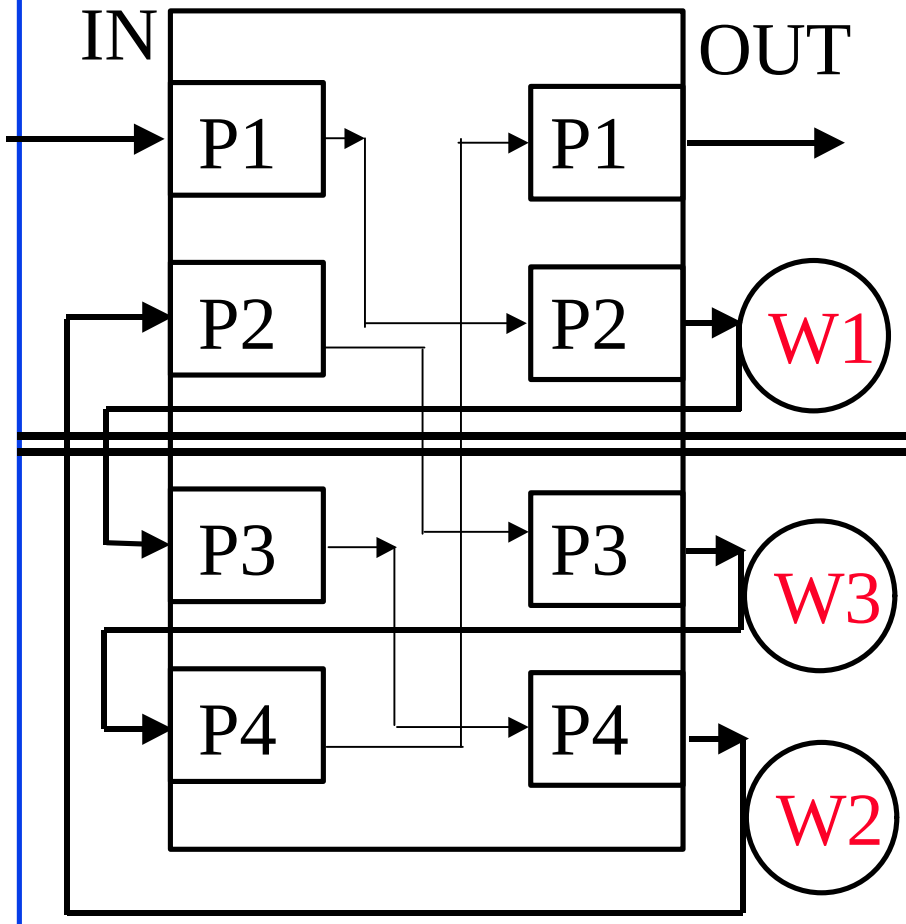


# VCC Chain

- 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



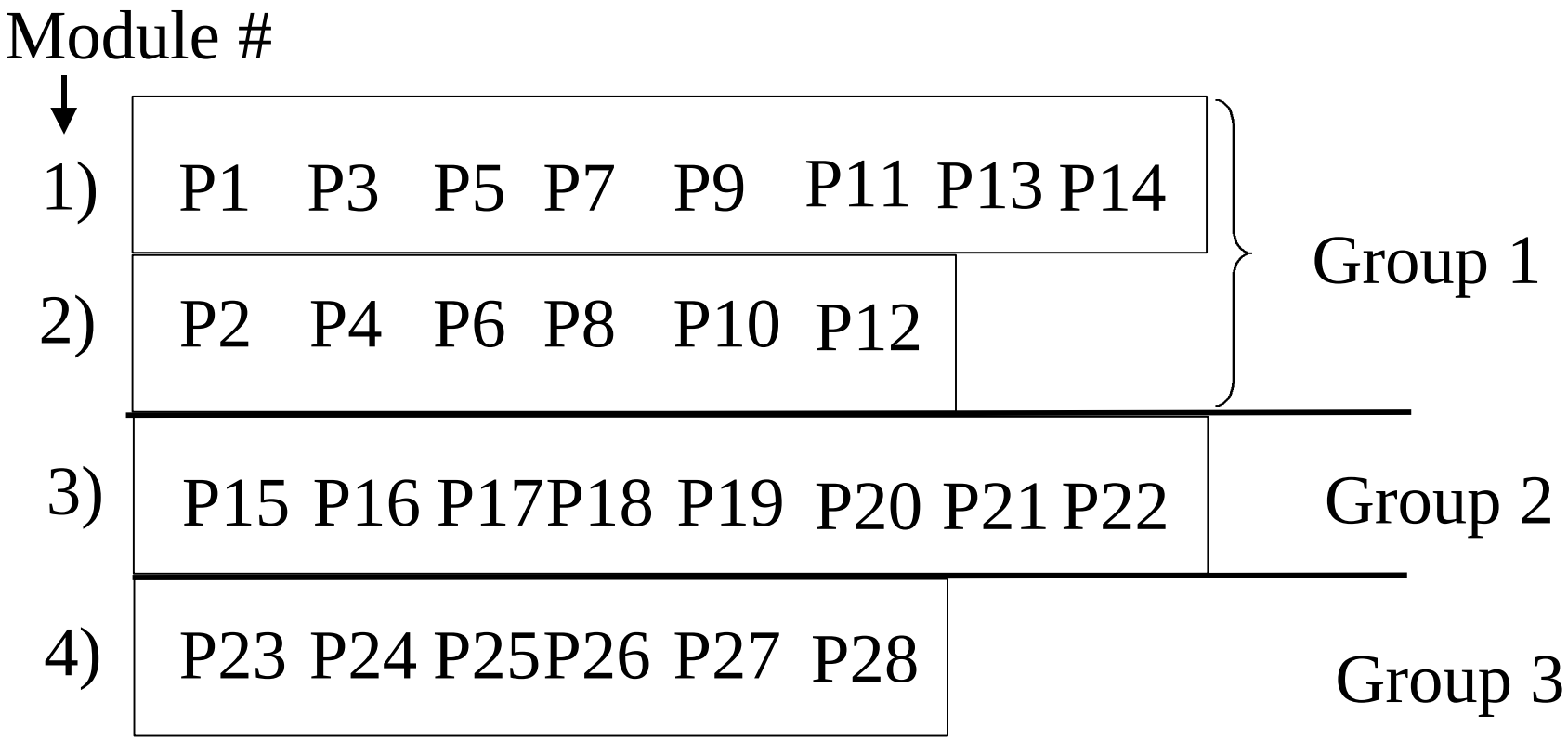

One of six possible VCC chains:

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




- 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

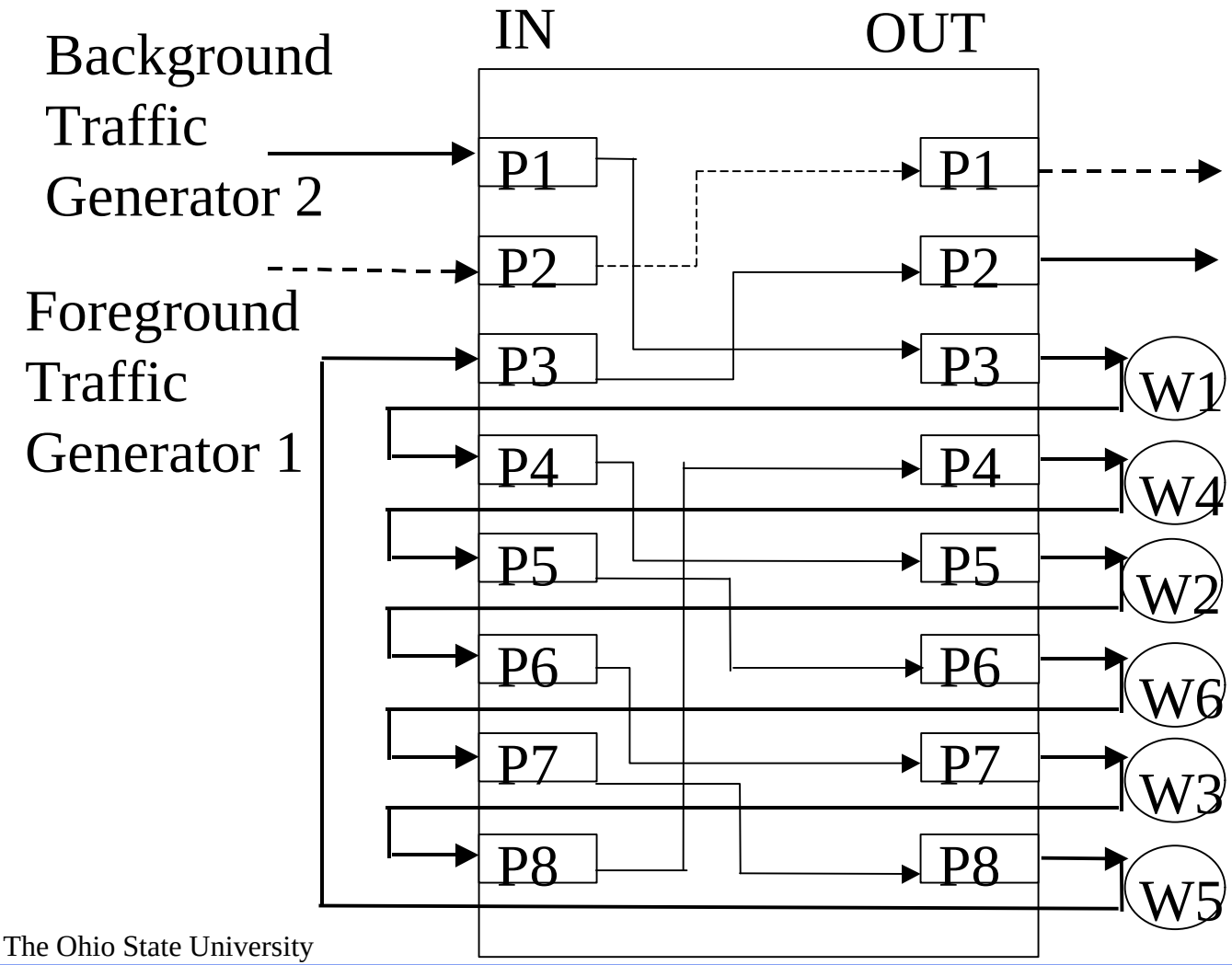
- Identifying the ports connected to the generator and/or analyzers
  - Avoid having only one port left over in a group. (That port cannot be connected externally to any other port)
  - This condition does not apply if the switch allows loopbacks.
- Note: The algorithm works with loopbacks also.

# 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.



# Algorithm

```
f = 1
 for  (k = 1 to r, step 1)
        { if(k>1)  f = 1 + Σ_{d=1}^{k-1} NW(d)
    for  (j = 1 to m, step 1)
        {  if(j>1) f = mod*(CH(1,j-1,k)+1, TNW);
         for  (i =1 to NW(k), step 1)
                {
                CH(i,j,k)=W(f);
                f = mod*(f+1, TNW);
                } end for i
          } end for j
} 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.

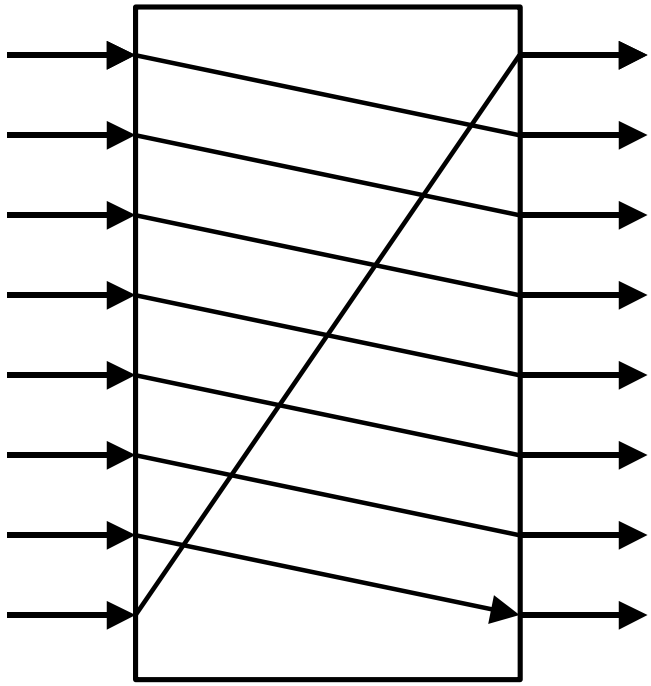
# Throughput and Latency Measurements

- 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.

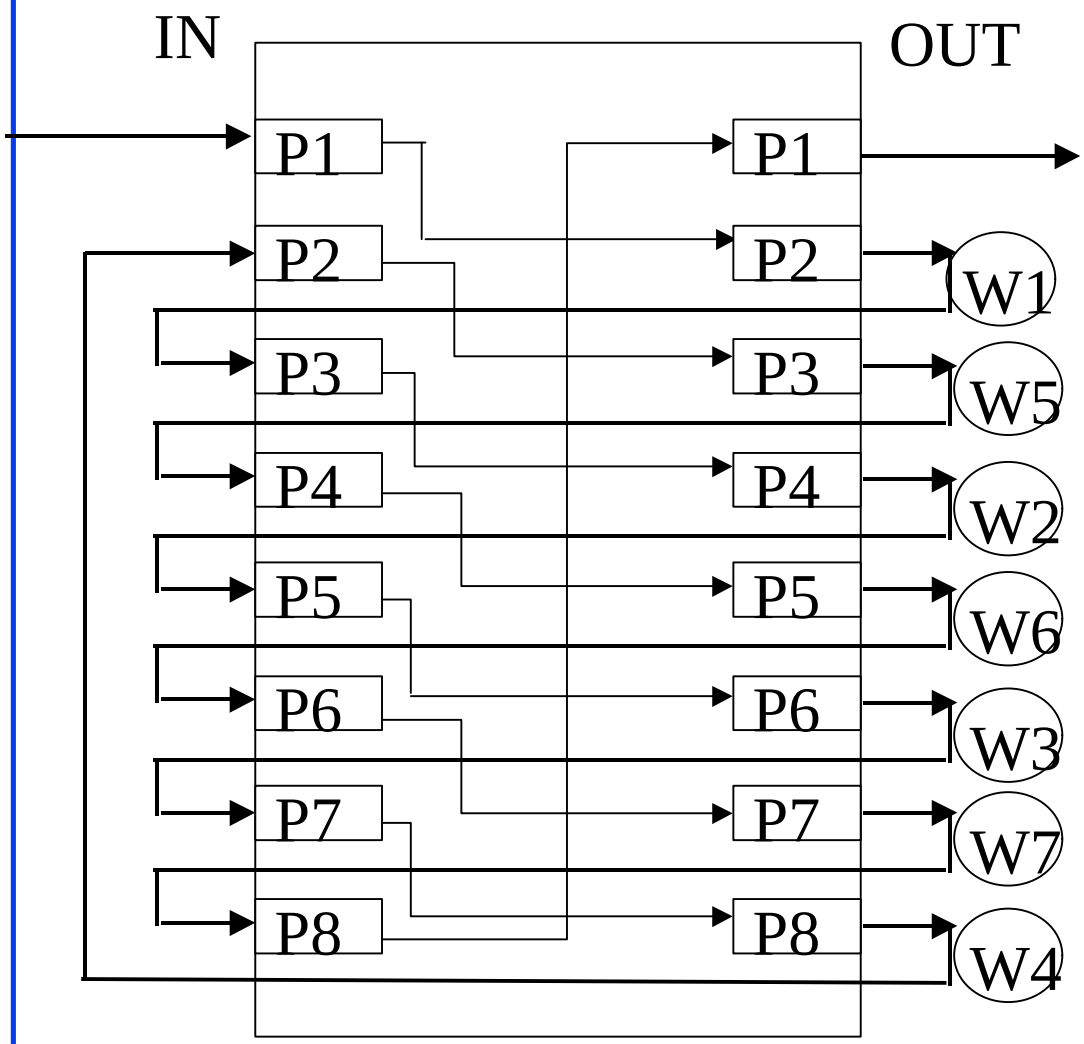
# Throughput and Latency Measurements (Cont)

- 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



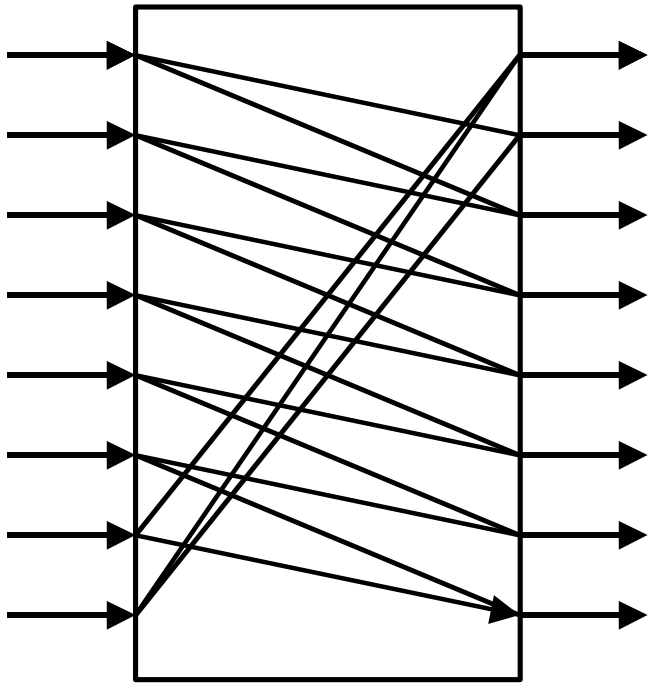
# 8-to-8 Straight, 1 Gen.



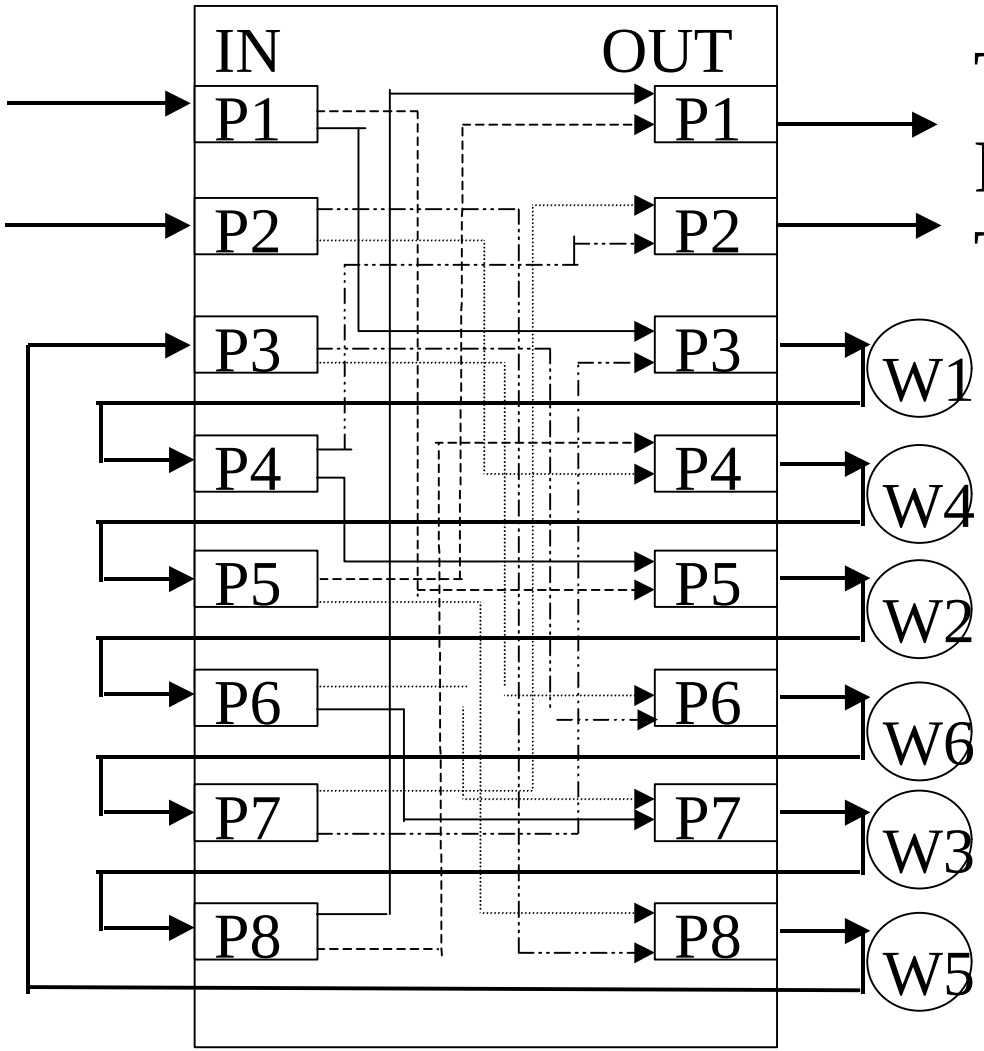

Throughput
Foreground
Traffic

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

# 8-to-2 Partial Cross Throughput Foreground



# 8-to-2 Partial Cross, 2 Gen.

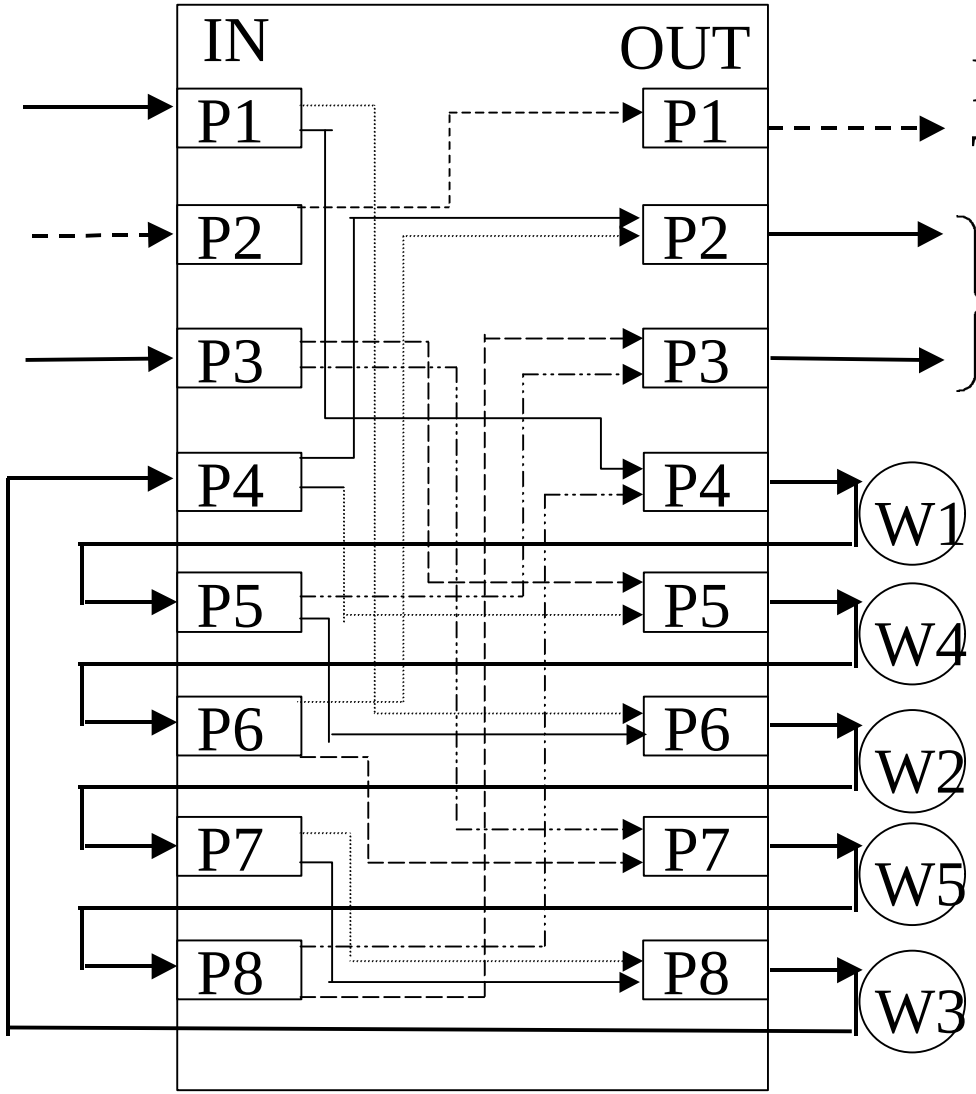



Throughput
Foreground
Traffic

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.




Foreground Traffic

Background Traffic

Gen.1

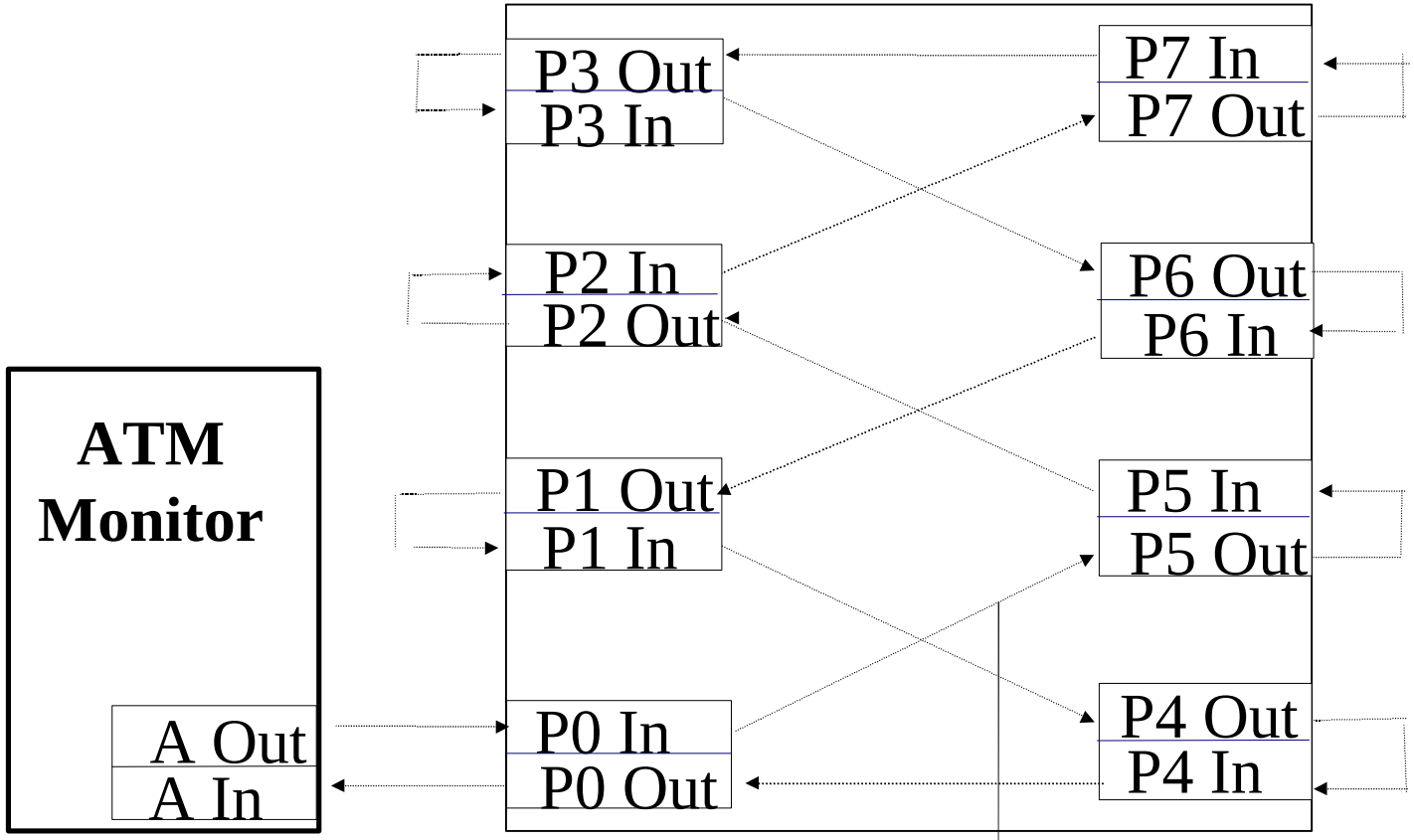
1) P1-W1-W2-W3-P2
2) P1-W2-W3-W4-P2

Gen.2

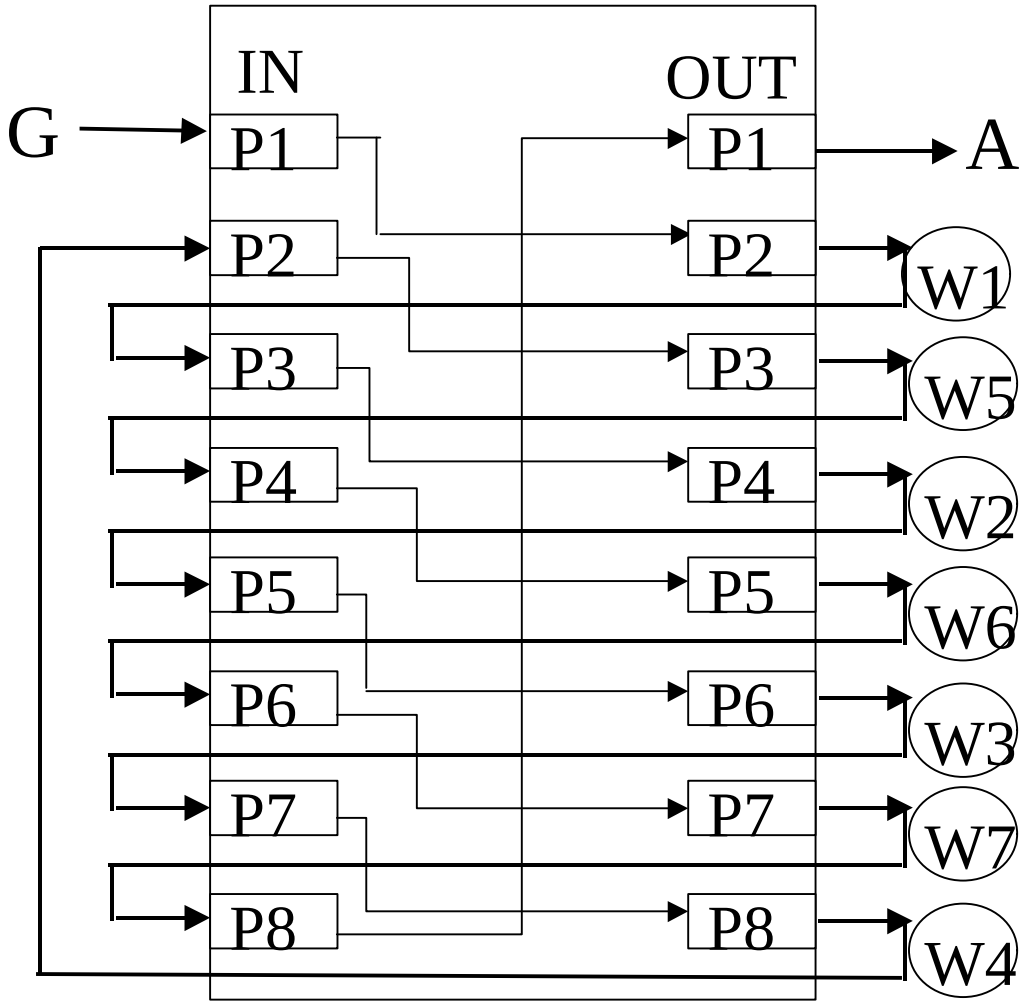
1) P3-W4-W5-P3
2) P3-W5-W1-P3

# Modifications to Sections 3.1.7 and 3.2.7

- Existing Figure 3.3: 8-to-8 straight configuration, 1 Gen.



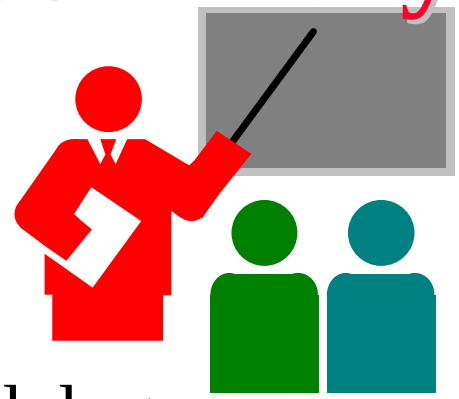
# New Figure 3.3



# 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=# of ports.

# Summary



- New Methodology:
  - Allows both loopback and non-loopback external connections.
  - Allows any number of generators.
    Þ 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.