CONGESTION AND STARVATION DETECTION FOR RIPPLE FIFOS

William S. Coates *bill.coates@sun.com*
Robert J. Drost *robert.drost@sun.com*
Sun Labs

Sun Microsystems, Inc.
Mailstop UMTV29-01
2600 Casey Avenue
Mountain View, CA 94043

Presentation Outline

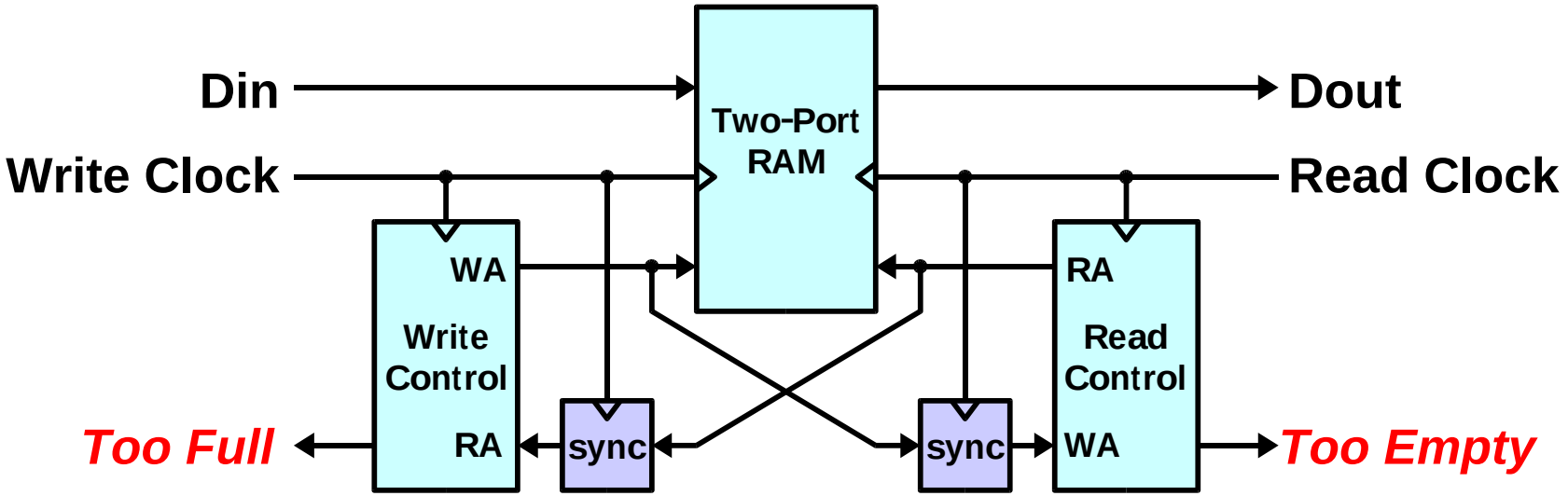
- The problem - "How full is the FIFO?"
- Congestion and starvation detection
- Simulation results
- Conclusions

The Problem

- Real-world FIFO applications:
  - Elastic data buffering
  - Crossing clock domains
  - Absorbing clock skew
- Most applications require measure of FIFO fullness
  - Must avoid *Overflow* or *Underflow*

Typical FIFO Buffer Implementation

- **Independently clocked I/O interfaces**
- **Synchronous "pointer" FIFO**
- **Fullness flags for Flow Control**

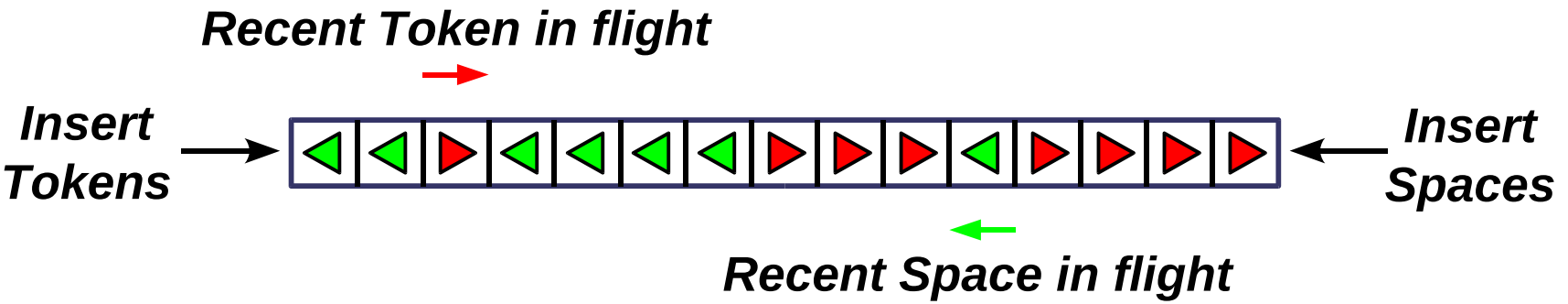


Why Not Ripple FIFOs?

- **Pro:**
  - **Fast**: *6 gate cycle (GasP), 8 gate cycle (asP*)*
  - **Simple**: *Low control overhead*
  - **Modular**: *Local communication; extensible*
  - **Flexible**: *Easy to embed in clocked systems*
- **Con:**
  - **Latency**: *Not a problem for short FIFOs*
  - **Area / Power**: *Ways exist to get around this*
  - **Fullness Flags**: *Have been a problem...*

How Full is the FIFO?

- **In general, exact occupancy is indeterminate**
- **Dynamic global state complicates things...**



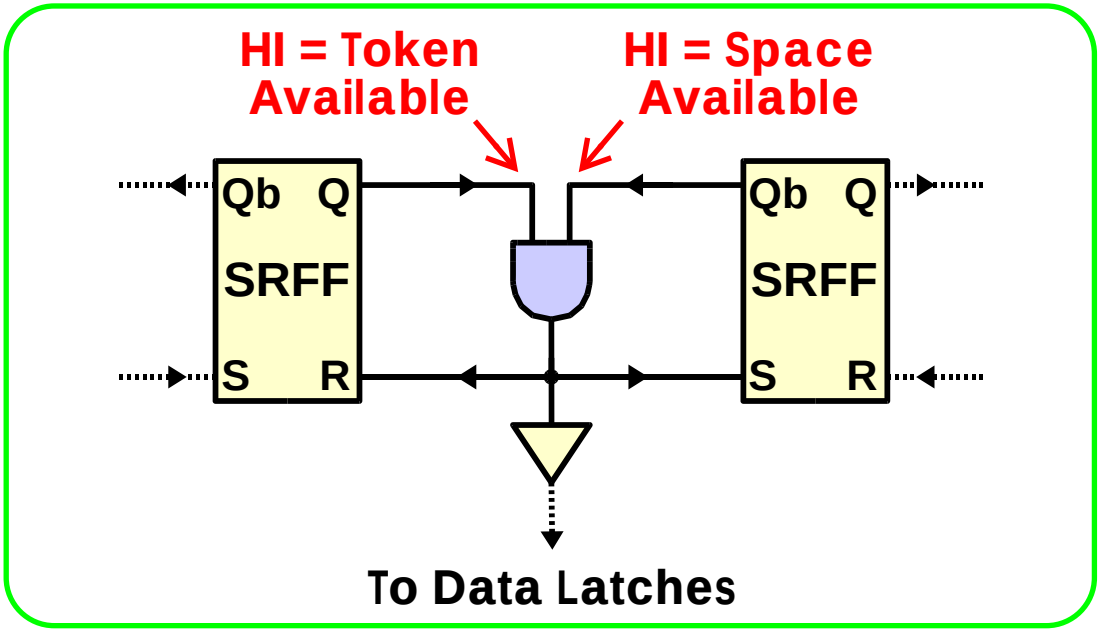
- **So, use local info: At any stage boundary, define**

Local Starvation : Throughput limited by *tokens*

Local Congestion : Throughput limited by *spaces*

Example FIFO Control: asP Protocol*

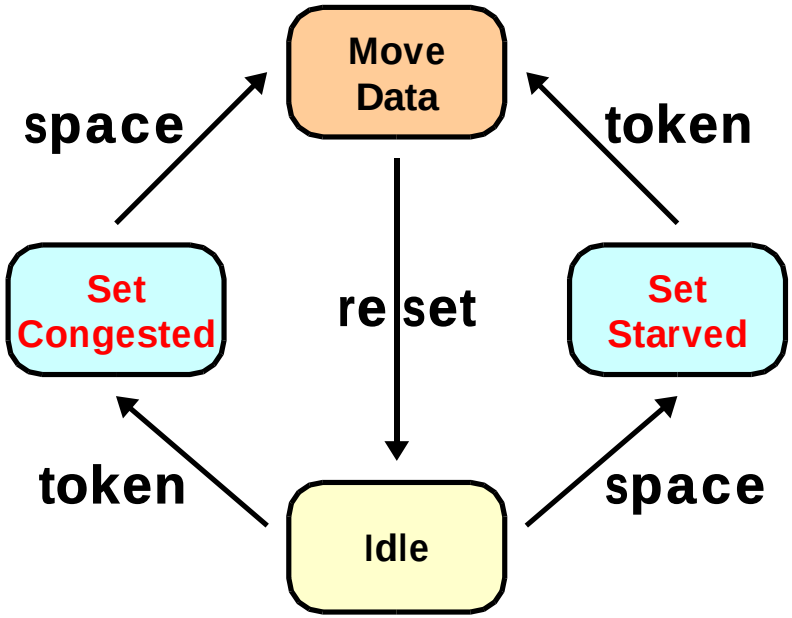
- **Core function is synchronization of 2 events:**
  - **When Token and Space are available, transfer data**

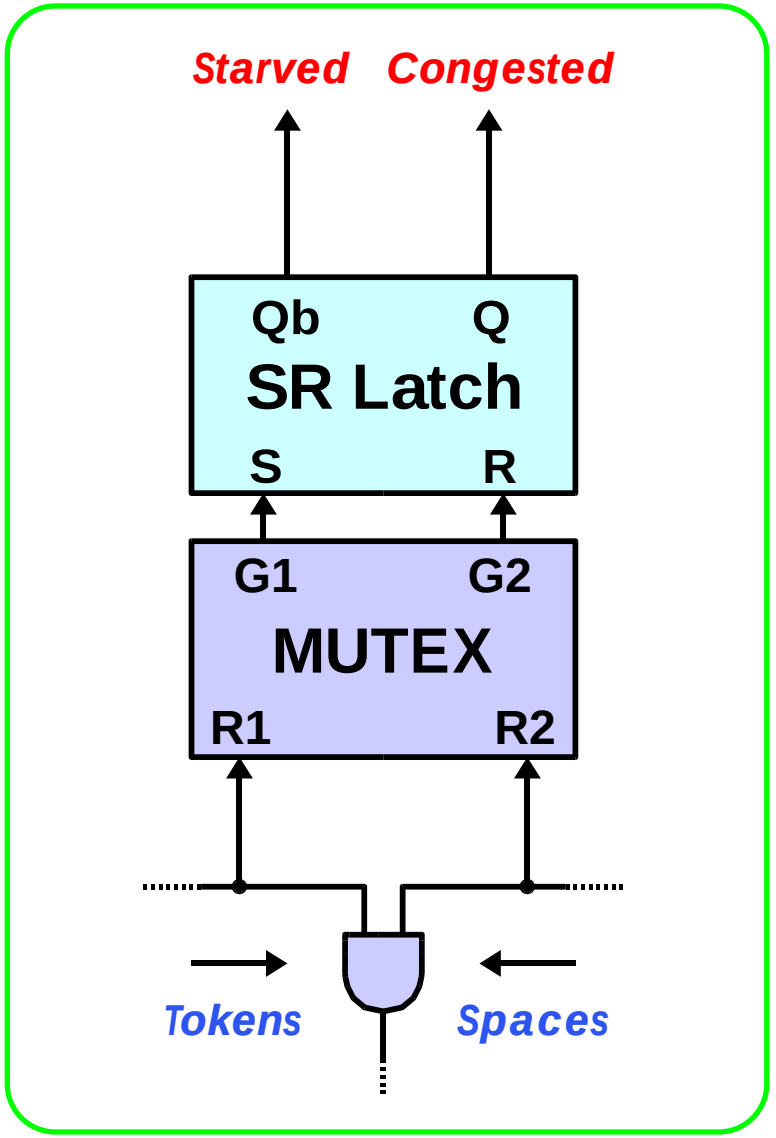


- **Monitor relative arrival time of 2 events:**
  - **Token arrives first** → ***locally congested***
  - **Space arrives first** → ***locally starved***

Phase Detector Operation

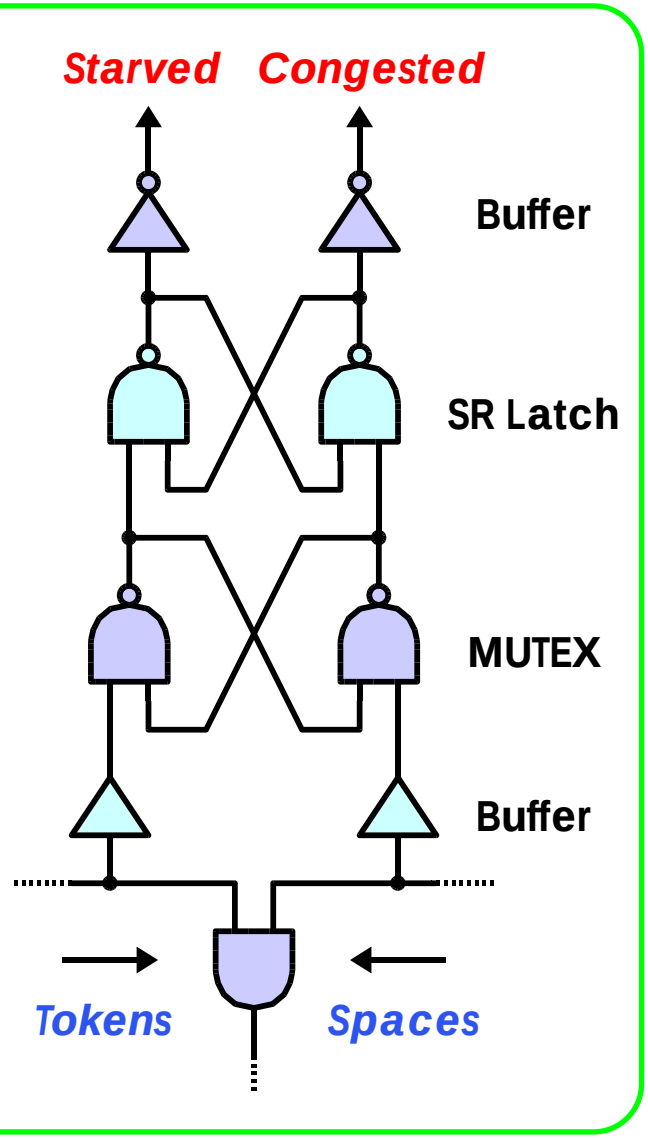
- Uses a *Latching Arbiter*



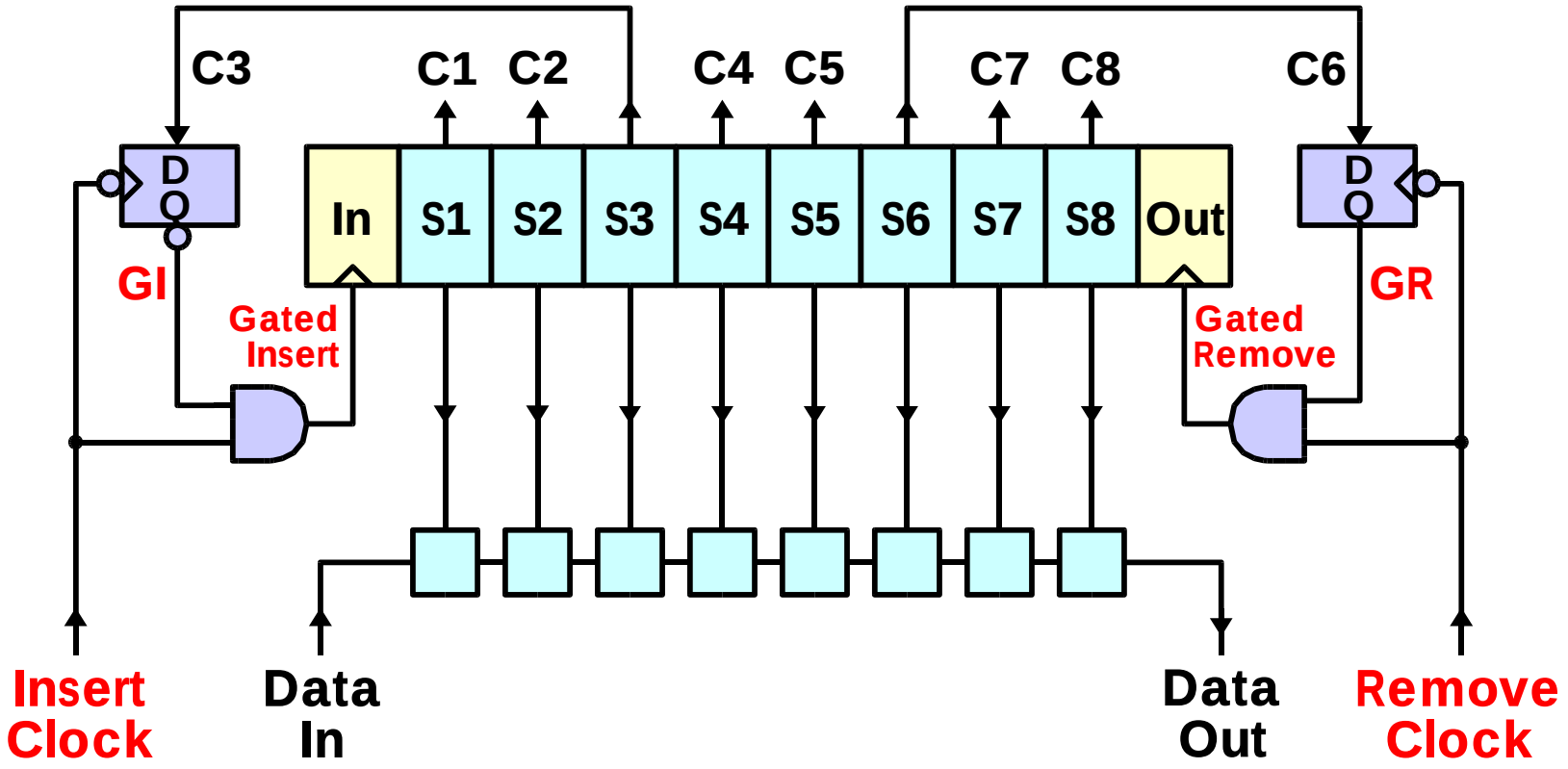


Phase Detector Implementation

- Output glitch acceptable (still need to synchronize)
- Requires near-simultaneous reset of inputs
- Also works for GasP



Example System: Automatic Flow Control

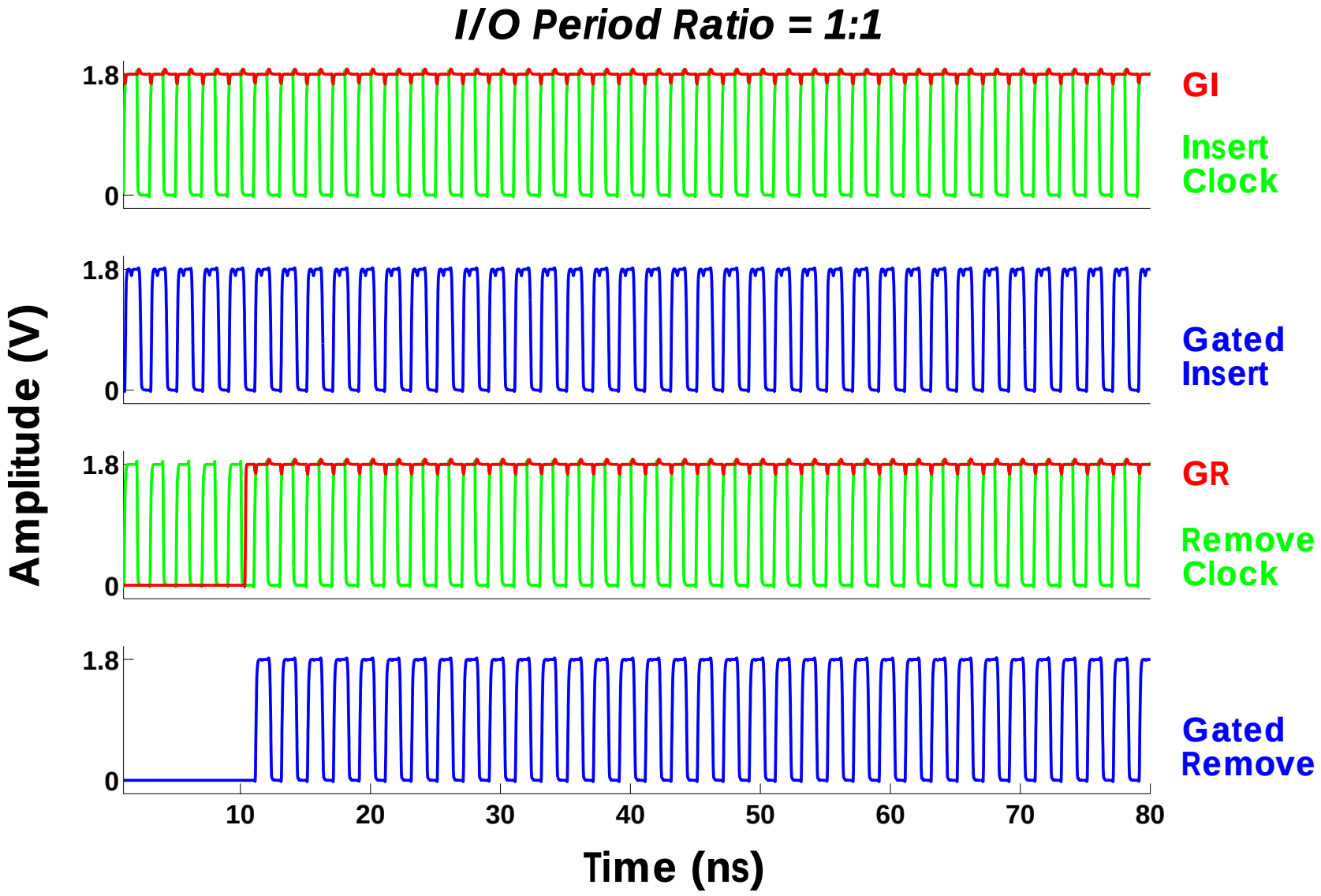


Simulation Results

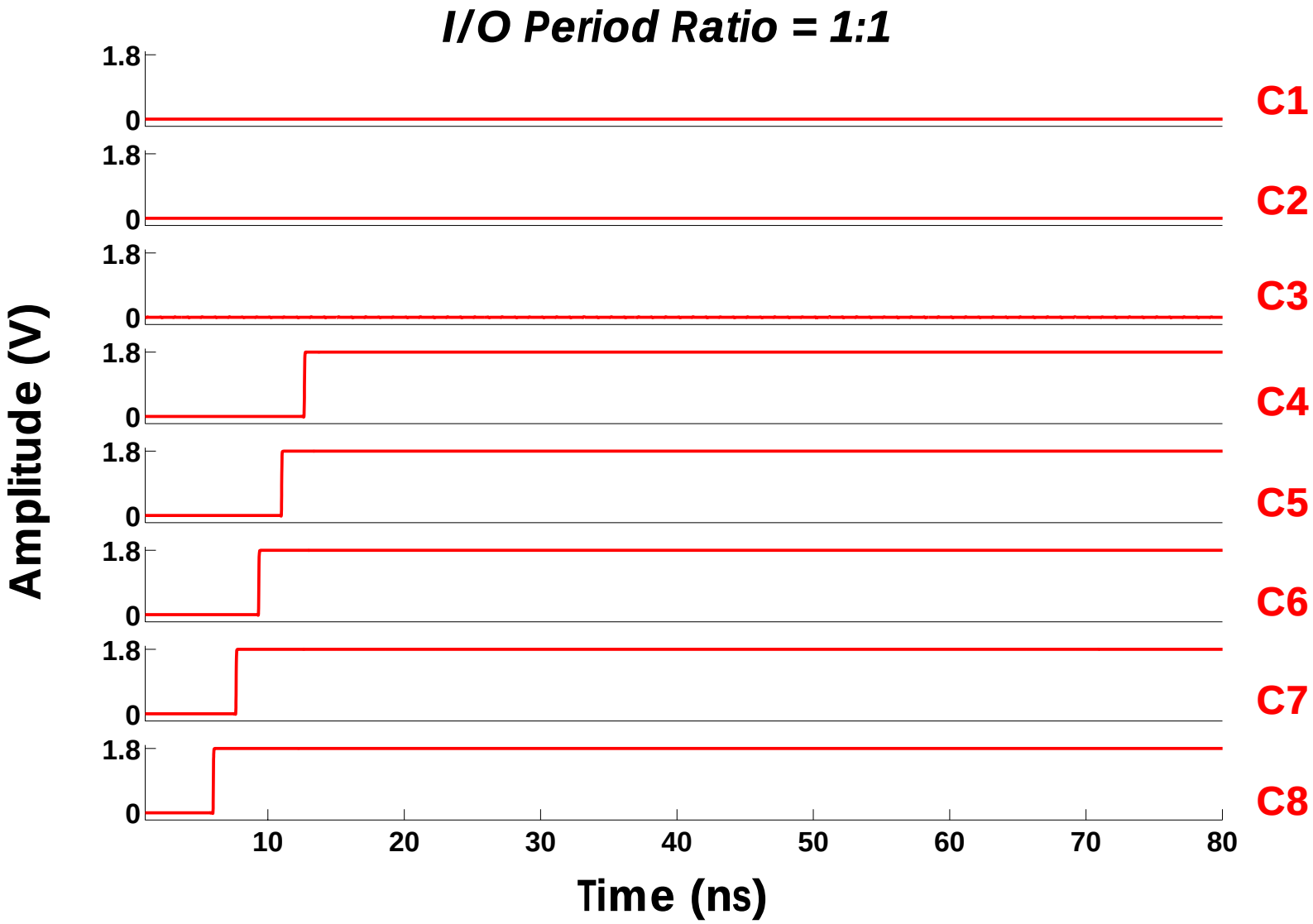
- Hspice circuit simulation
  - TSMC 180 nm process parameters
  - 32-bit datapath load; estimated wire parasitics
  - FIFO cycle time: 680 ps
- Simulation Test Environment
  - Nominal I/O cycle time set to 2 ns (~ 3X FIFO cycle)

	Clock Period (ns) Insert	Remove	I/O Period Ratio
Test 1	2.0	2.0	1 : 1
Test 2	3.0	2.0	3 : 2

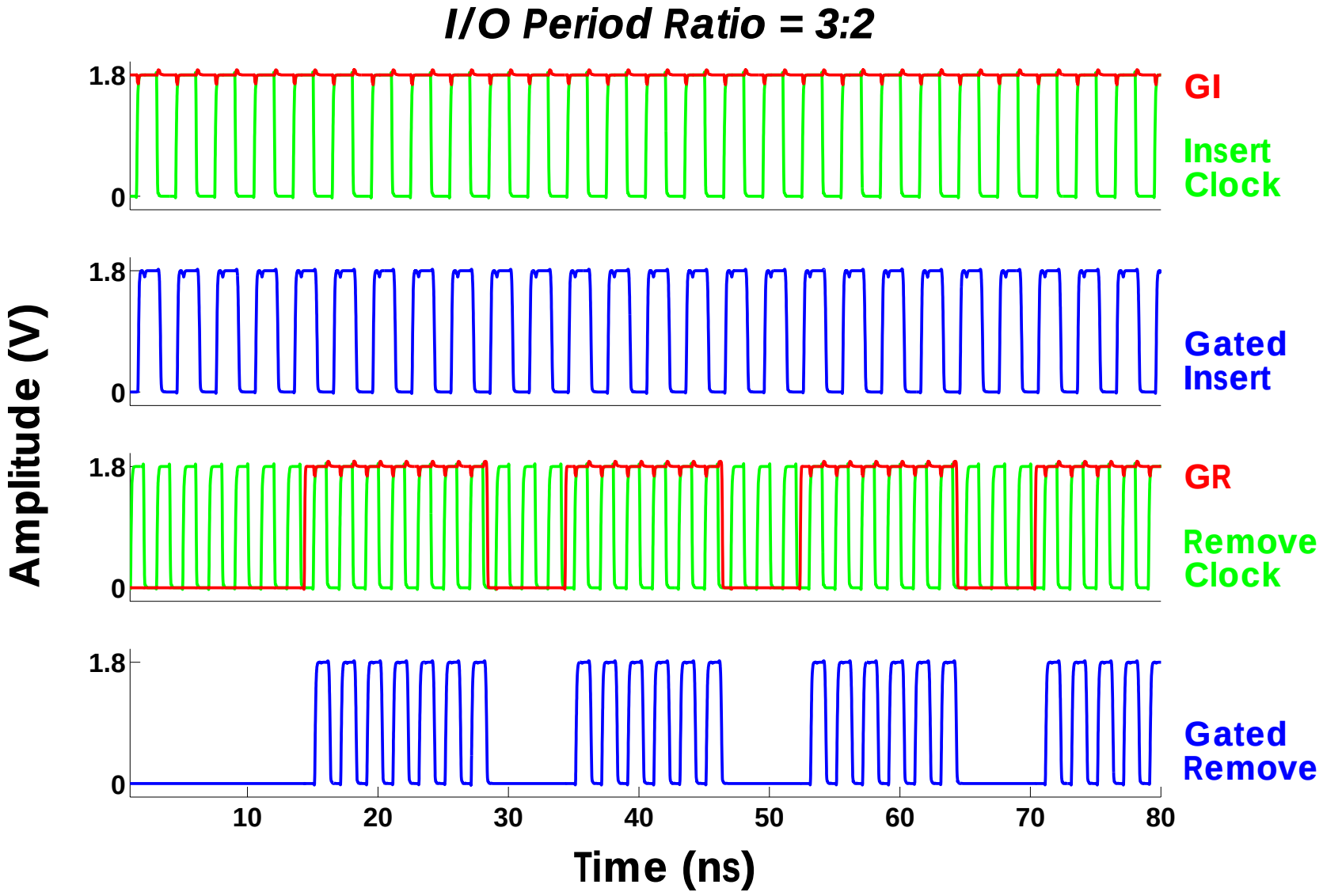
Hspice Simulation 1: Clock Gating



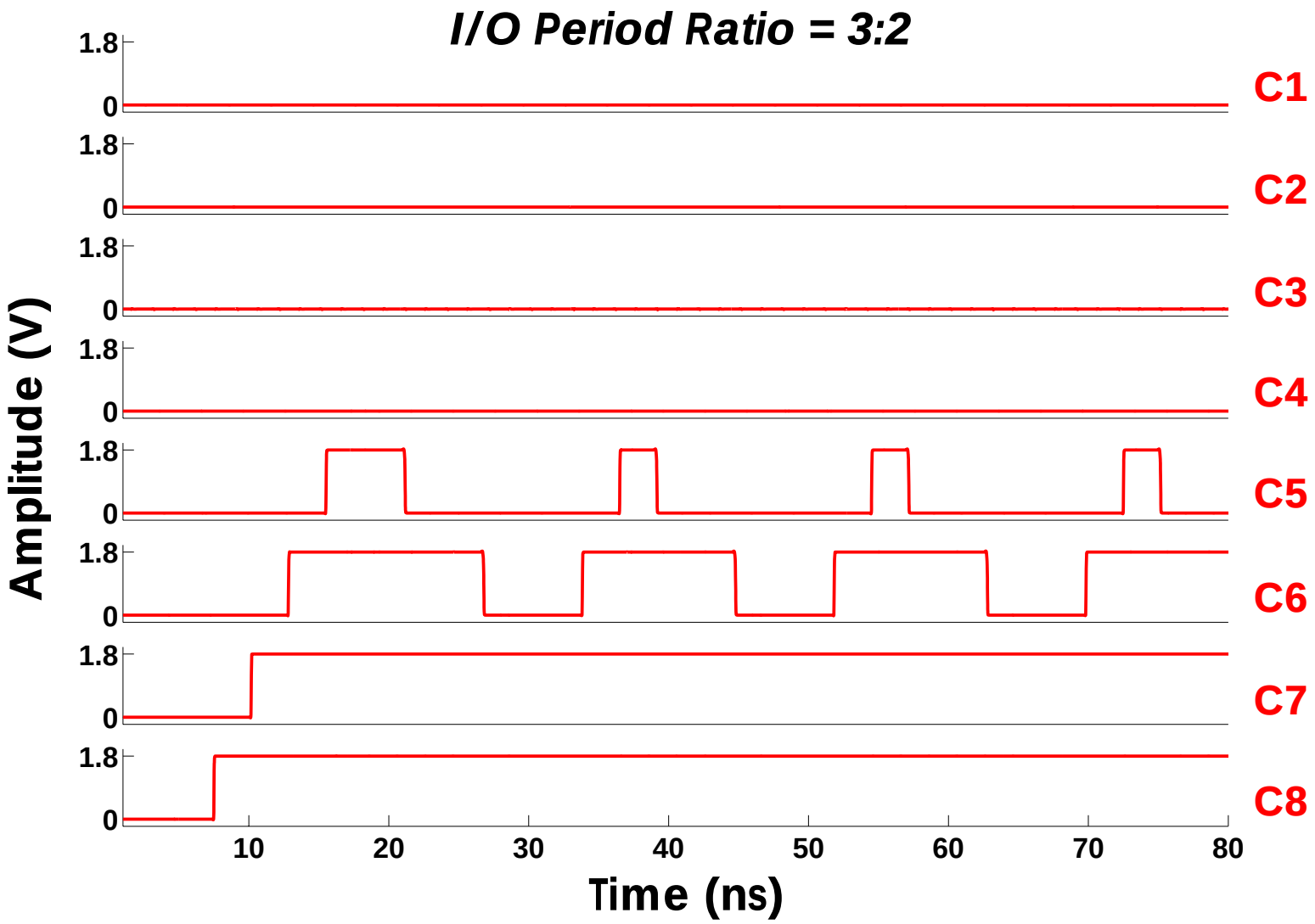
Hspice Simulation 1: Congestion



Hspice Simulation 2: Clock Gating



Hspice Simulation 2: Congestion



Conclusions

- ***Simple***
  - **Monitor local signals only**
  - **Minimal loading on FIFO**
- ***Flexible***
  - **Adaptable to many FIFO control styles**
  - **Easy to monitor multiple congestion conditions**

Future Work

- **Methodology to guide detector placement**
- **Failure analysis**
- **Application to real-world circuits**