Archi 2013

Réseaux-sur-puce, systèmes GALS et DVFS, circuiterie asynchrone, et intégration 3D, quatre domaines qui se croisent dans les futures architectures des systèmes multiprocesseur embarqués

Abbas Sheibanyrad

TIMA, SLS

Grenoble, France

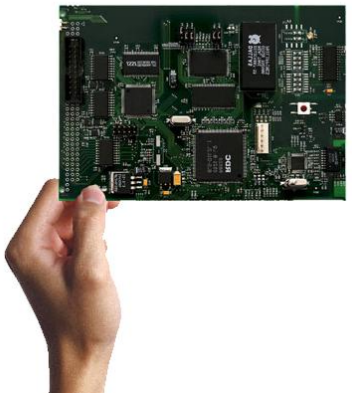
Outline

- GALS/DVFS and Asynchronous NoCs
- ASPIN an example of Async NoCs
- Async-Sync Interface
- Synchronous vs. Asynchronous
- 3D-Integration and Asynchronous NoCs
- Vertical Link Serialization
- Vertically-Partially-Connected 3D-NoC
- Conclusion

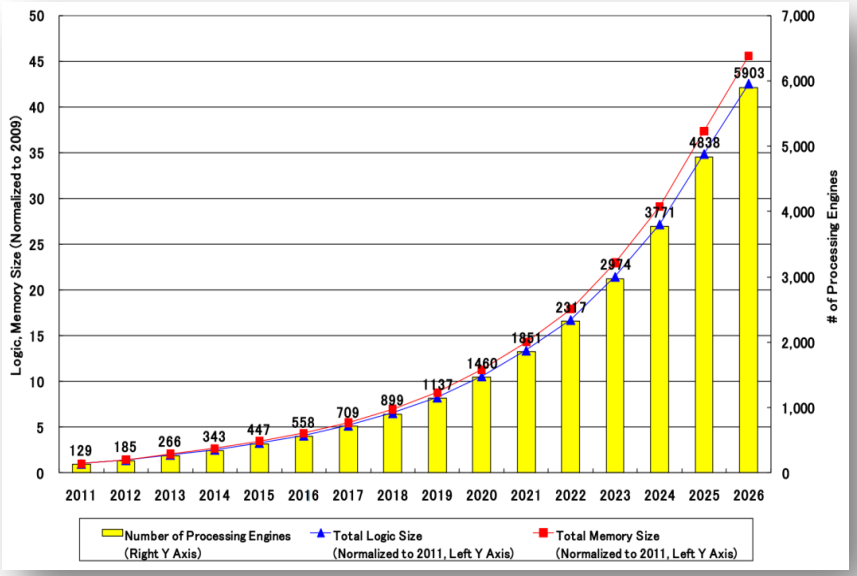
Technology Evolution !

- Evolution of the fabrication technology
  - Integration of systems with billions of transistors in only one chip
  - Hundreds and even thousands of components

System-on-Board



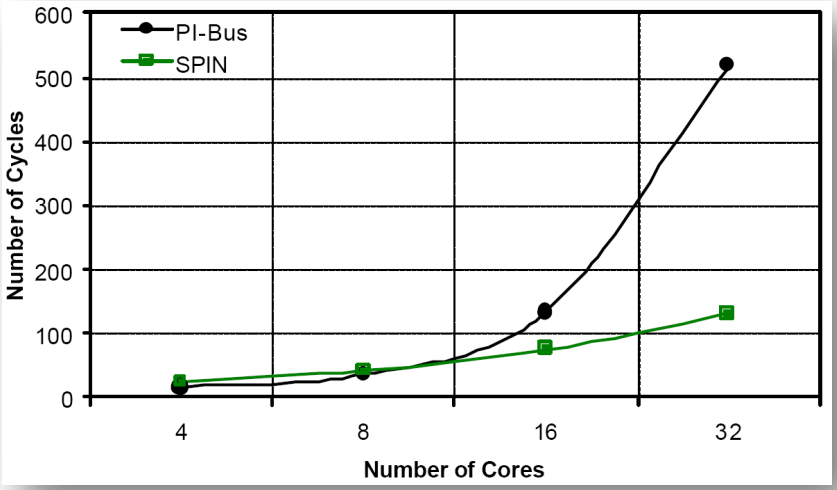
System-on-Chip

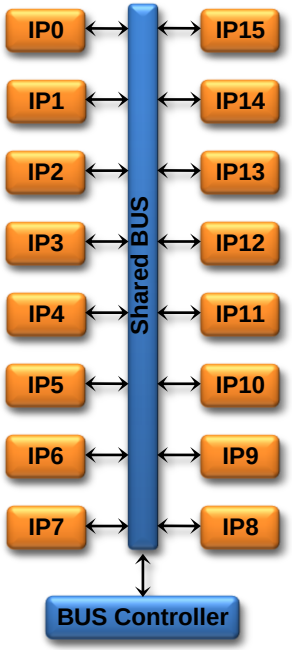


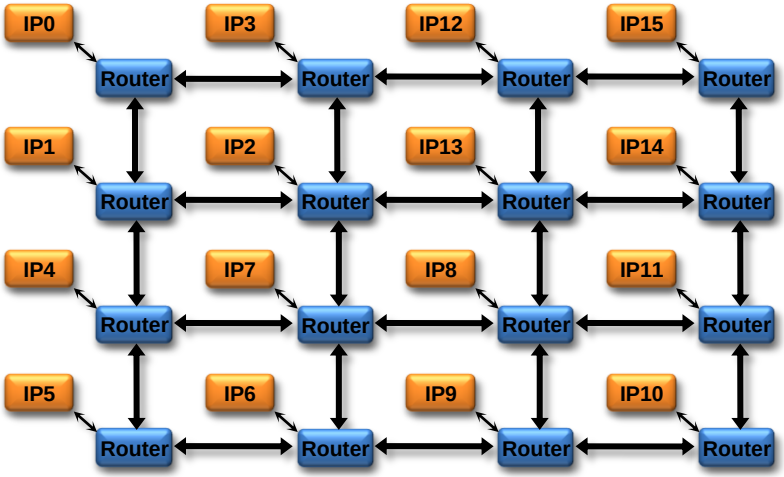
- Multi-Core Systems
  - Performance
    - Thermal issues and clock skews limit the increase of clock frequency
    - Parallelism
  - Power Consumption
    - Simple Cores
    - Power down the idle cores
  - Rapid Design
    - Reuse of repeated tiles integrated into a common infrastructure

Scalability !

- Key role of the communication infrastructur
  - Networks-on-Chip (NoCs) attempt to solve the bandwidth bottleneck of traditional interconnects in large Multi-Processor Systems-on-Chip (MP-SoCs)

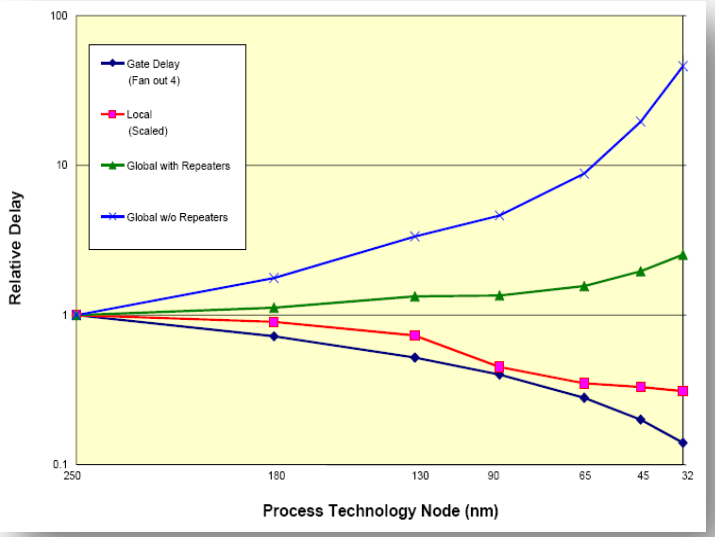


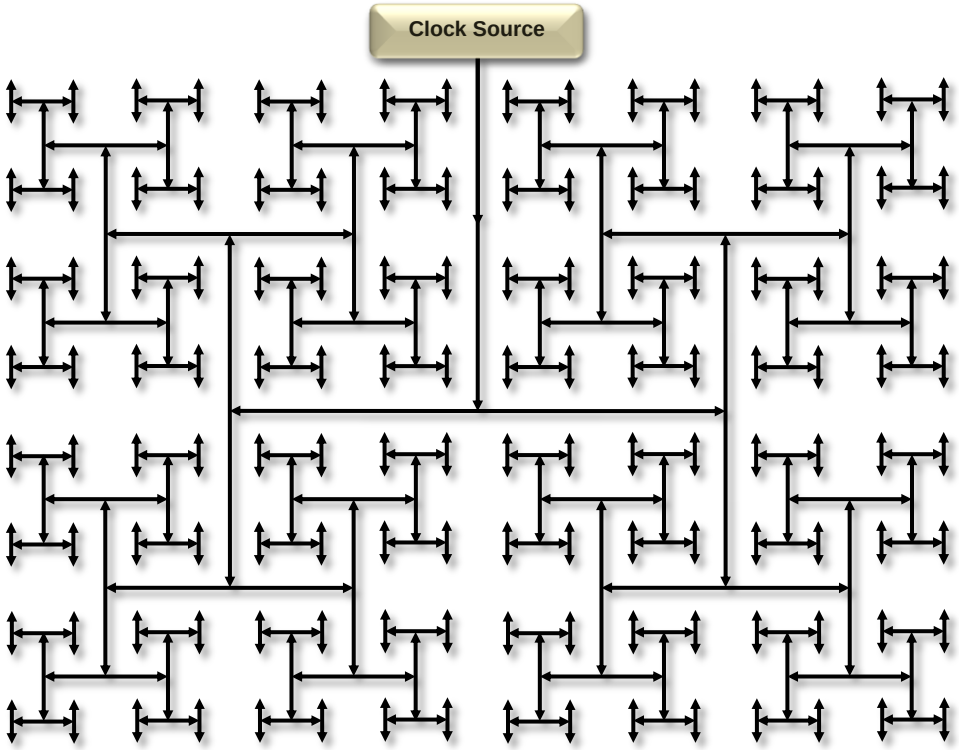




Clock Distribution

- Deep Submicron Technologies
  - Aggravation of physical problems
  - Predominant effect of long wires on delay and consumption

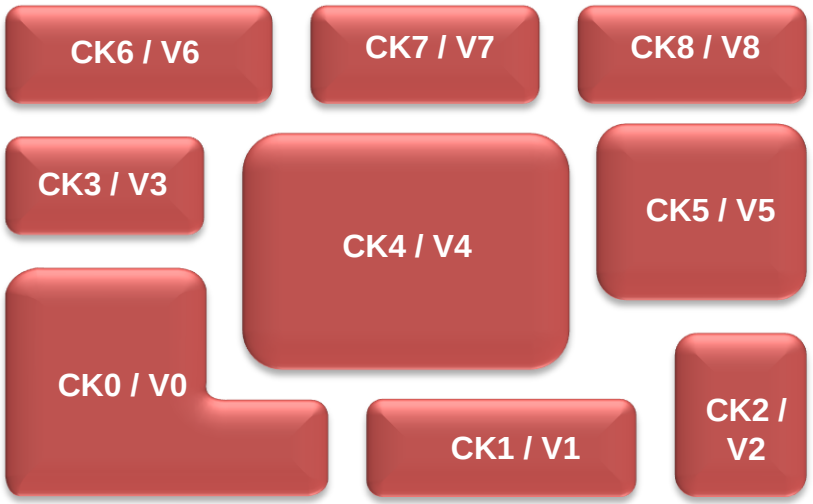




- Nightmare of Global Synchronization
  - Impossible Global Distribution of a single clock signal over a chip
  - Clock skew claiming a larger relative part of the total cycle time
  - The clock distribution network demanding increasing portions of the power and area budget
  - Fabrication Process Variation
  - Temperature Variation

GALS/DVFS Paradigm, a prominent solution

- Reducing the Problem to a number of smaller Sub-Problems
  - Several Independent Clock/Voltage Clusters
- Networks-on-Chip as a Structured Approach
  - The network: Globally Asynchronous part
  - Subsystems: Locally Synchronous parts

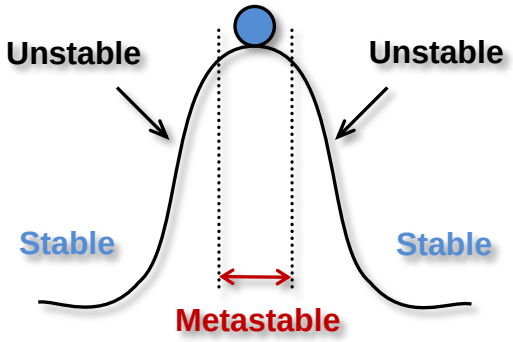


Timing Dependency Methods

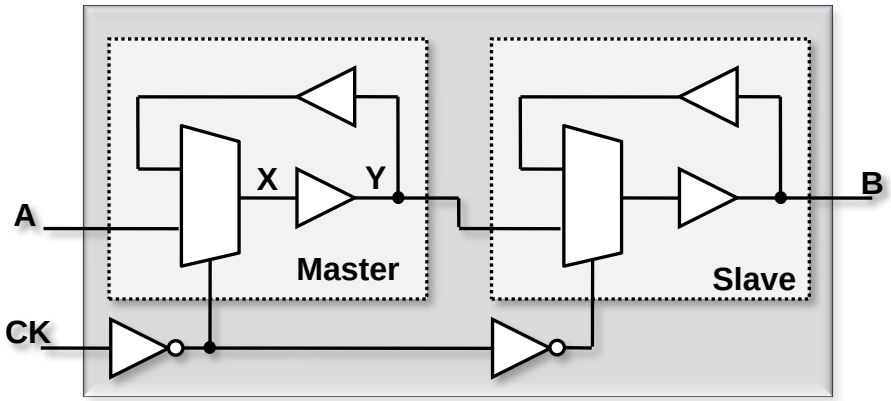
Type	Δ Frequency	Δ Phase
Synchronous	0	0
Pseudochronous	0	Constant
Mesochronous	0	Undefined
Plesiochronous	ε	ε
Heterochronous	Rational	Undefined
MultiSynchronous	Undefined	Undefined
Asynchronous	-	-

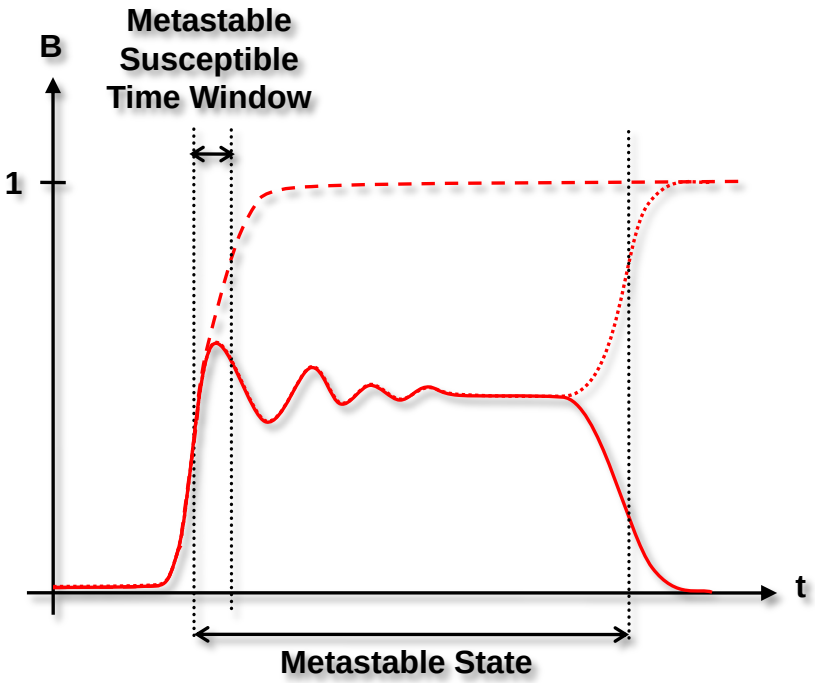
BUT …

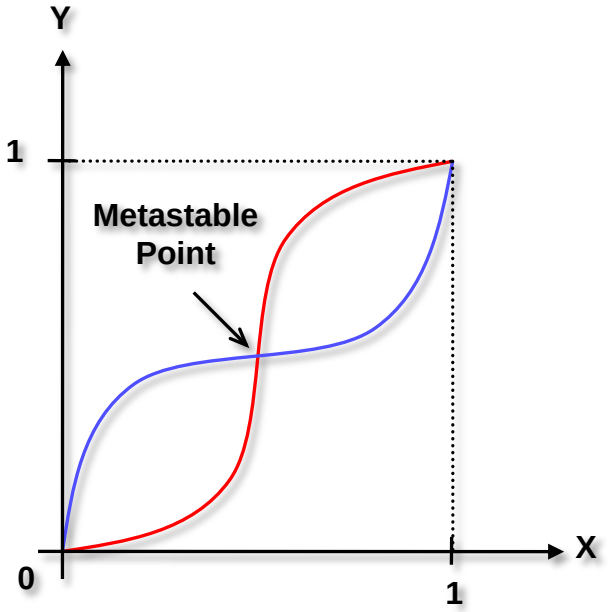
- How separated synchronous domains can robustly communicate together ?
  - Transferring data between different timing domains requires safe synchronization
  - Metastability, an unavoidable state of a bistable device is a major concern



Flip-Flop susceptible to Metastability

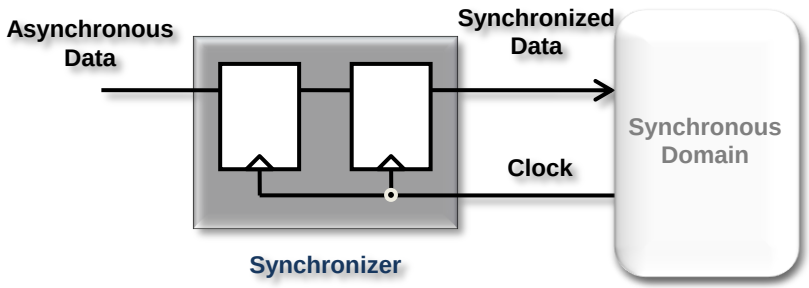


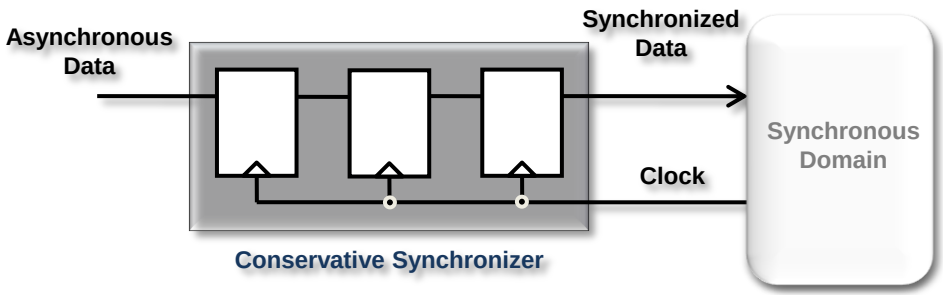




Synchronizer

- Clock Stretching or Pausible/Stoppable Clocking
  - Non applicable for high speed design with large clock buffer delays
- Prevention of undesirable value propagation
  - Asynchronous behavior for the next stage
- Several Cascaded Flip-Flops
  - Acceptable value (many years) for Mean Time Between Failure
  - $MTBF = \frac{e^{T/\tau}}{T_W f_A f_D}$
  - Trade-Off between Latency and Robustness



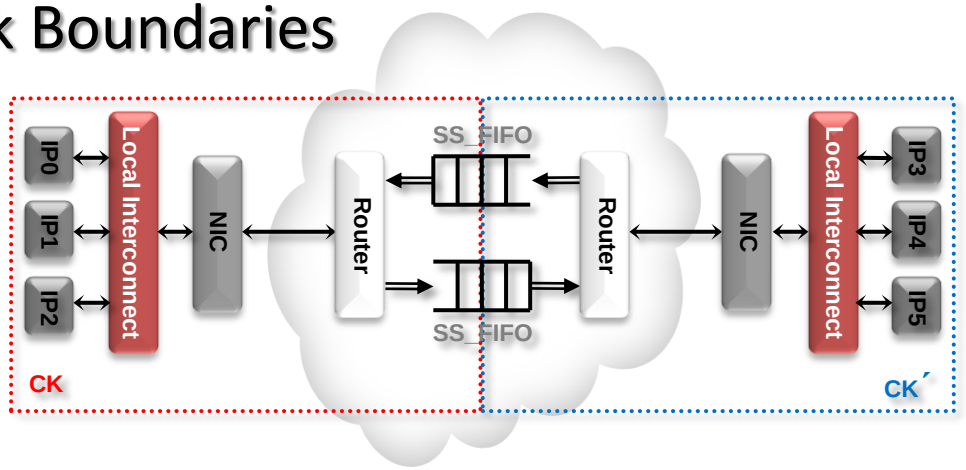


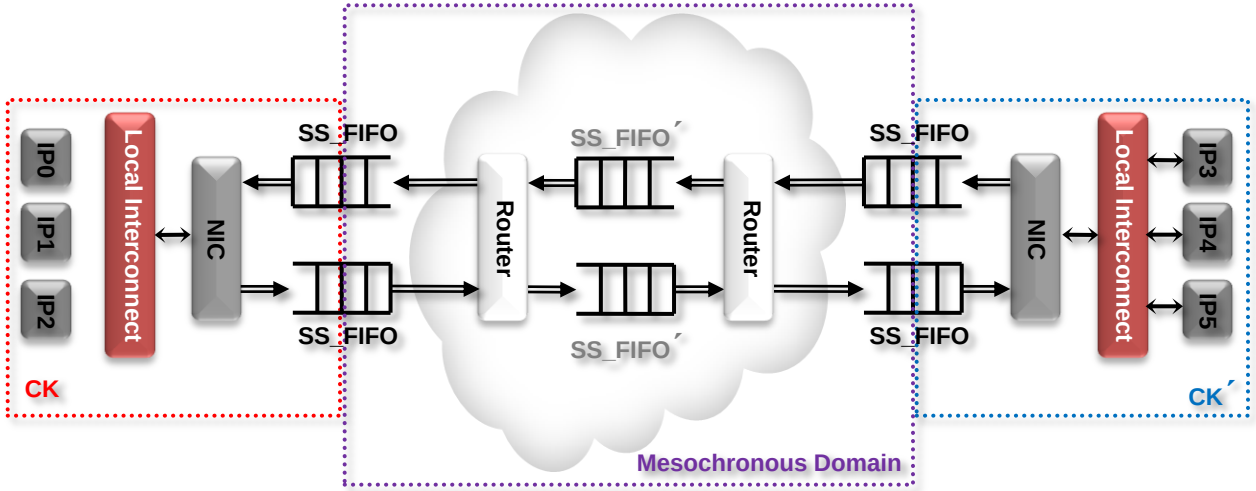
Special FIFOs as Robust Interfaces

- Coupling two Fundamental Problems
  - High-Level issue of Flow Control
  - Low-Level issue of Synchronization
- Minimizing the Synchronization Cost
  - Only the handshake signals must be synchronized
  - Adjustable Trade-off between Latency and Robustness

Multi-Synchronous NoC

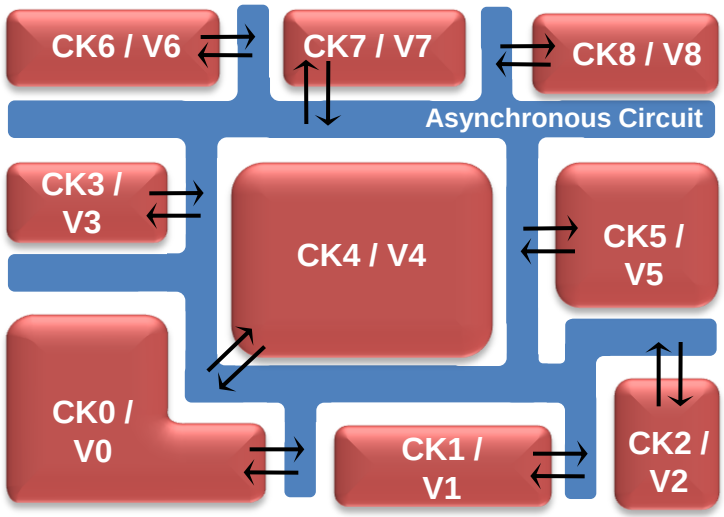
- Synchronous Circuits for Routers
- Bisynchronous FIFOs in Clock Boundaries
- Mesochronous Clocking
  - Fast Frequency
  - Fixed Frequency
  - But, extra Clock Frequency…





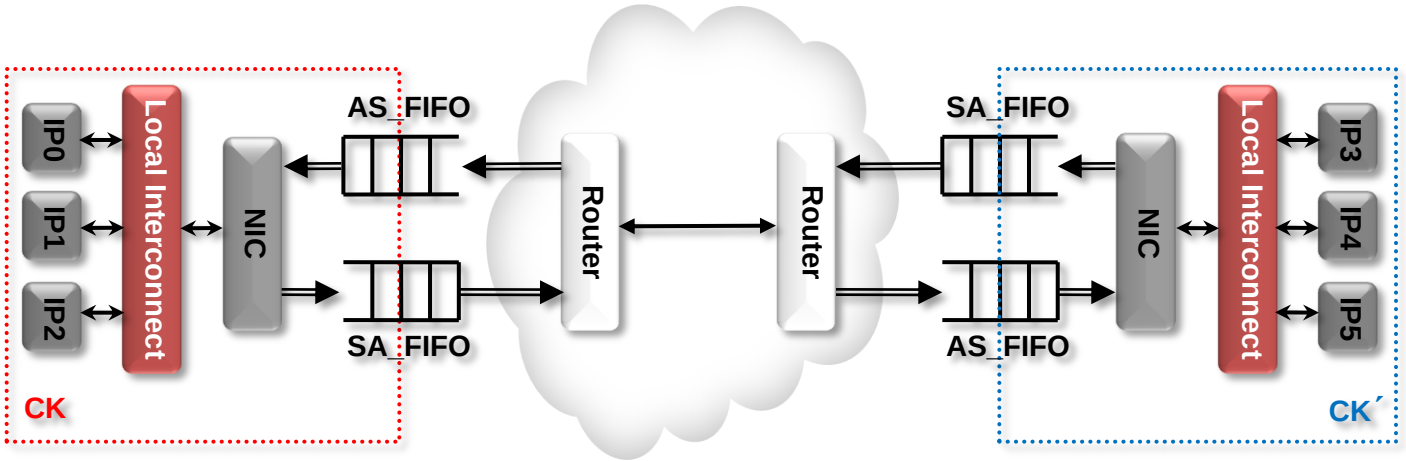
Asynchronous NoC, an Alternative Solution

- Reduce the need of Synchronization
  - Just in the Network Interface Controller
- Reusability in a Plug-and-Play Fashion
- Almost Zero Idle Power Dissipation
- As Fast as Possible
  - Global Timing Independence
- Scalability
  - Cluster Size Independence



Asynchronous NoC

- Fully Asynchronous Architecture for the Network
- Synchronous Compliant Interfaces
  - Synchronous-to-Asynchronous FIFO
  - Asynchronous-to-Synchronous FIFO

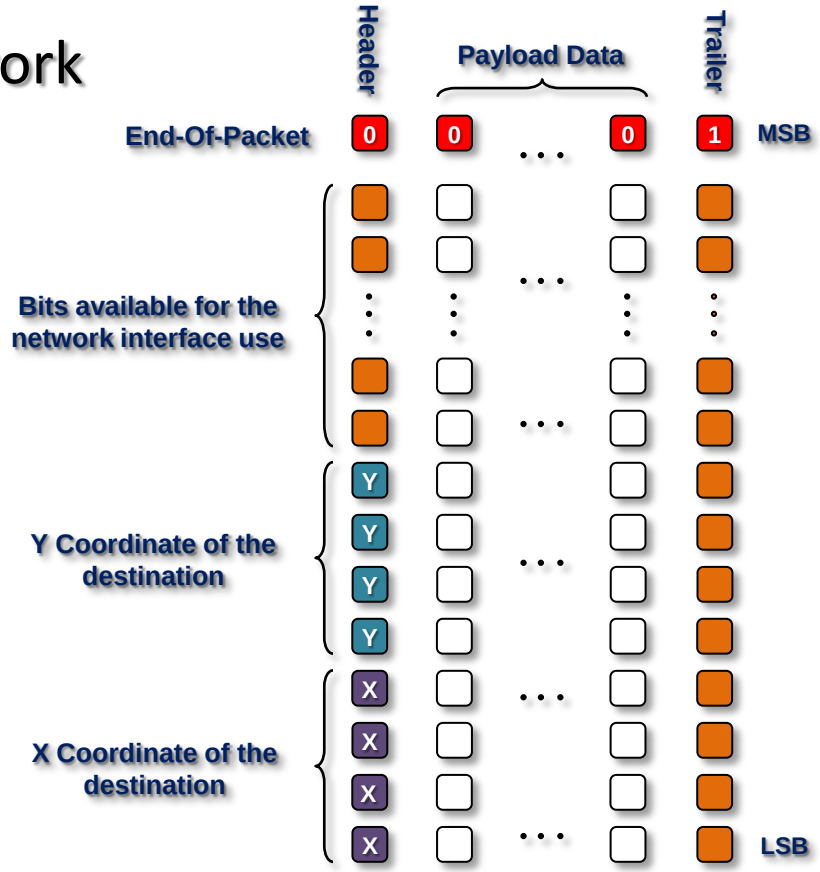


Outline

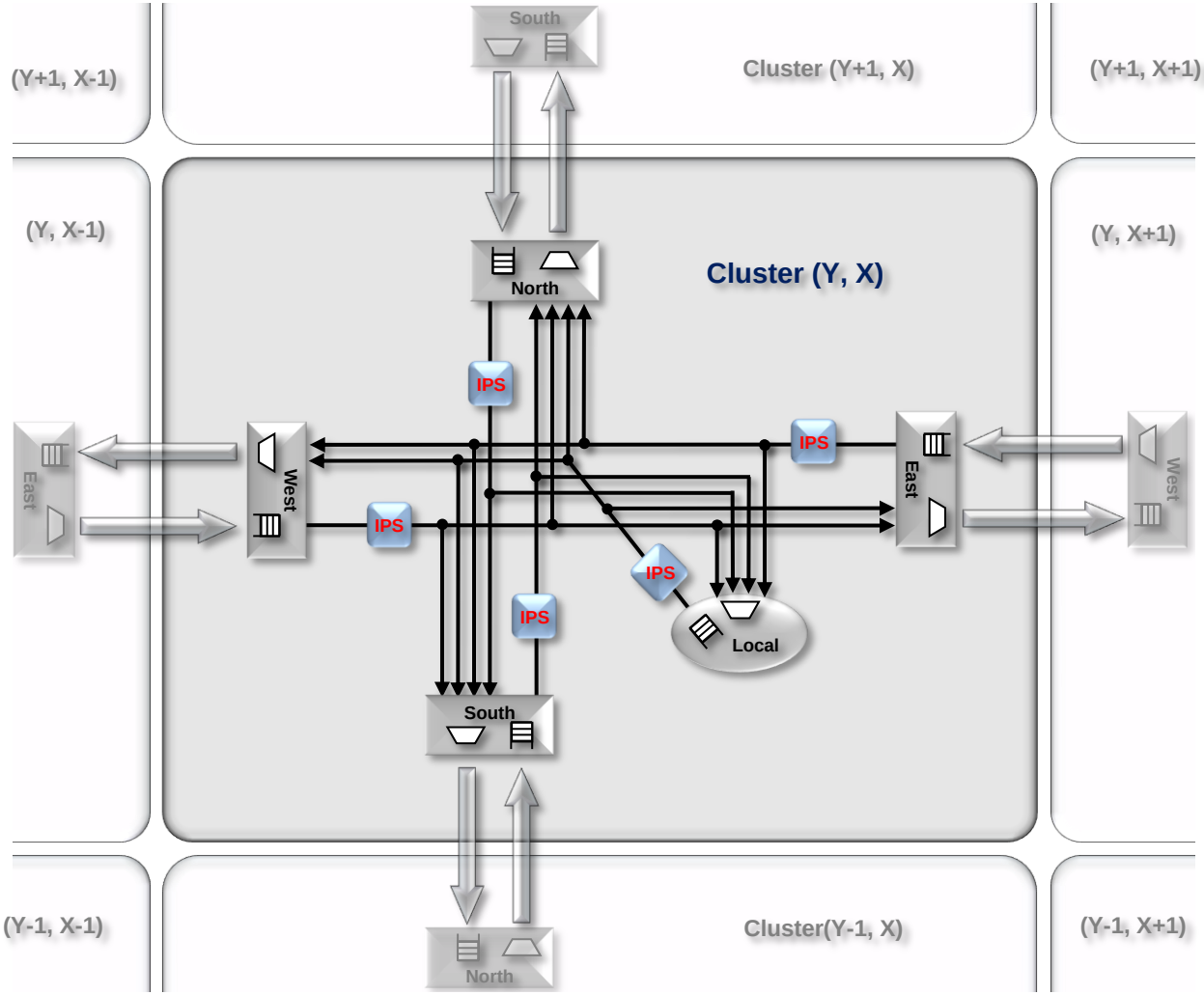
- GALS/DVFS and Asynchronous NoCs
- ASPIN an example of Async NoCs
- Async-Sync Interface
- Synchronous vs. Asynchronous
- 3D-Integration and Asynchronous NoCs
- Vertical Link Serialization
- Vertically-Partially-Connected 3D-NoC
- Conclusion

ASPIN General Features

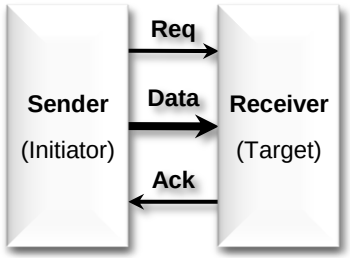
- Two-Dimensional Mesh topology
  - Destination Address as an absolute coordinate of (Y, X)
- X-First Routing Algorithm
- Wormhole Packet-Switching Network
  - 33-bit Word as a Flit
    - 1 bit for Flag of EOP
    - 32 bit for Data
- Input Buffering
- Round-Robin Scheduling

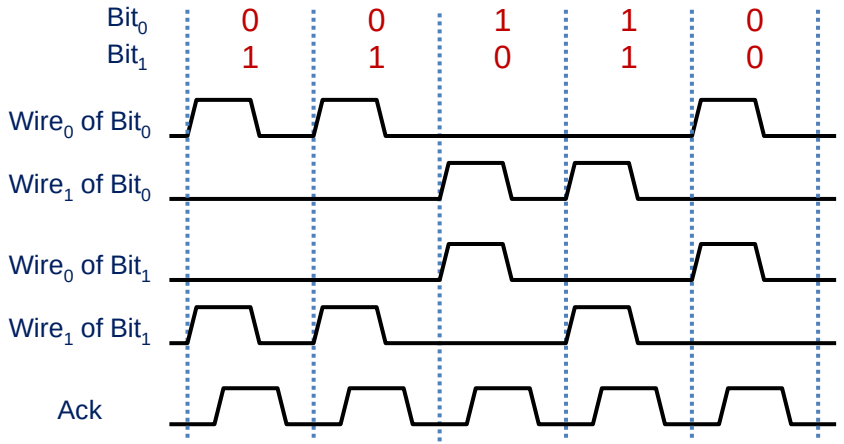


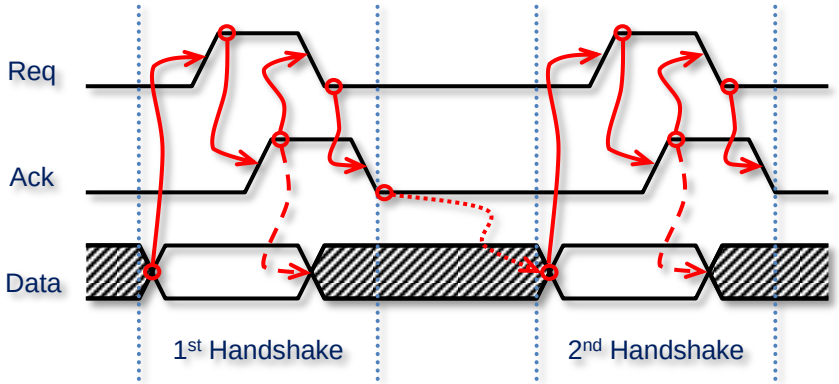
Distributed Router



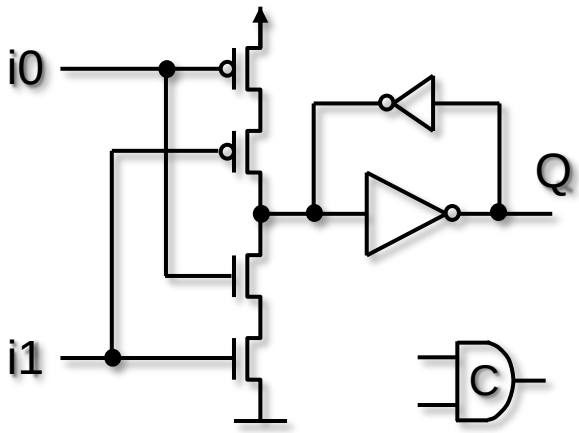
Asynchronous Handshake Protocol



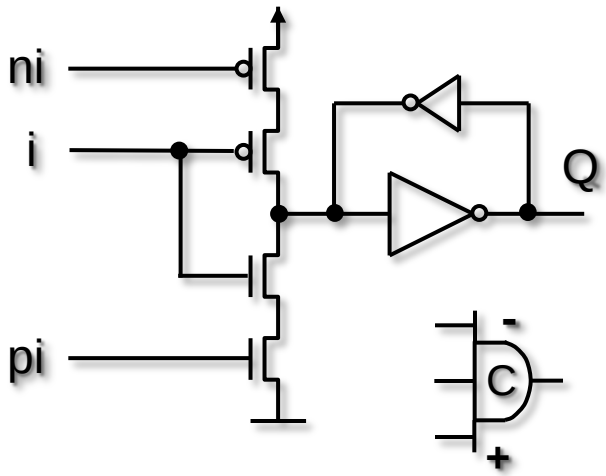




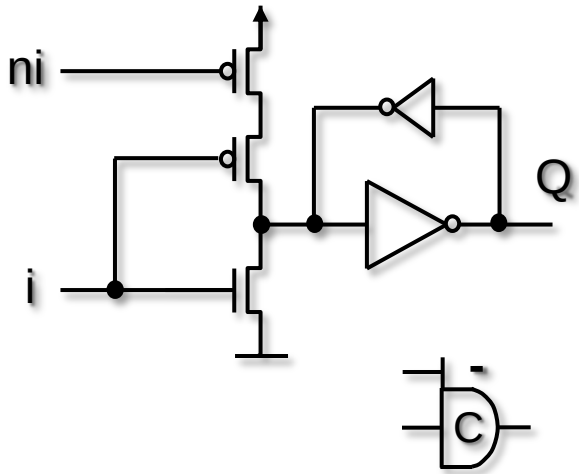
Asynchronous Design Cells



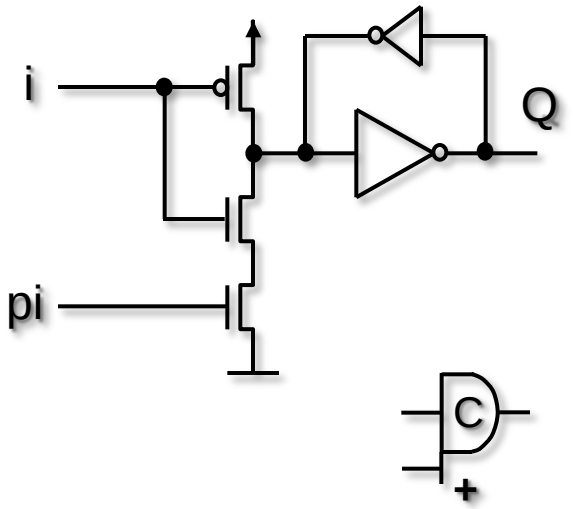
i1	i0	Q
0	0	0
1	1	1
1	0	Q’
0	1	Q’



i	ni	pi	Q
0	0	x	0
1	x	1	1
1	x	0	Q’
0	1	x	Q’

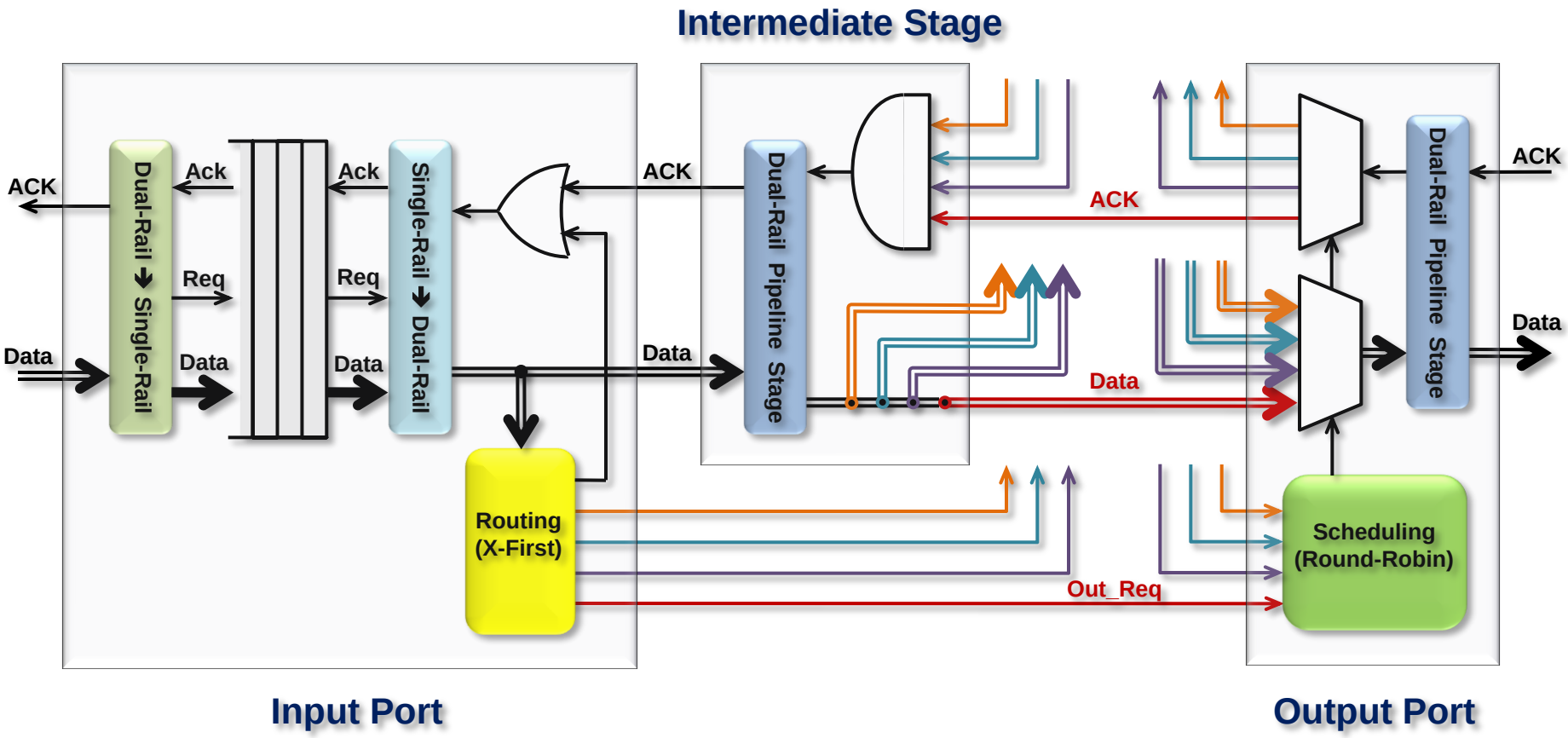


i	ni	Q
0	0	0
1	x	1
0	1	Q’



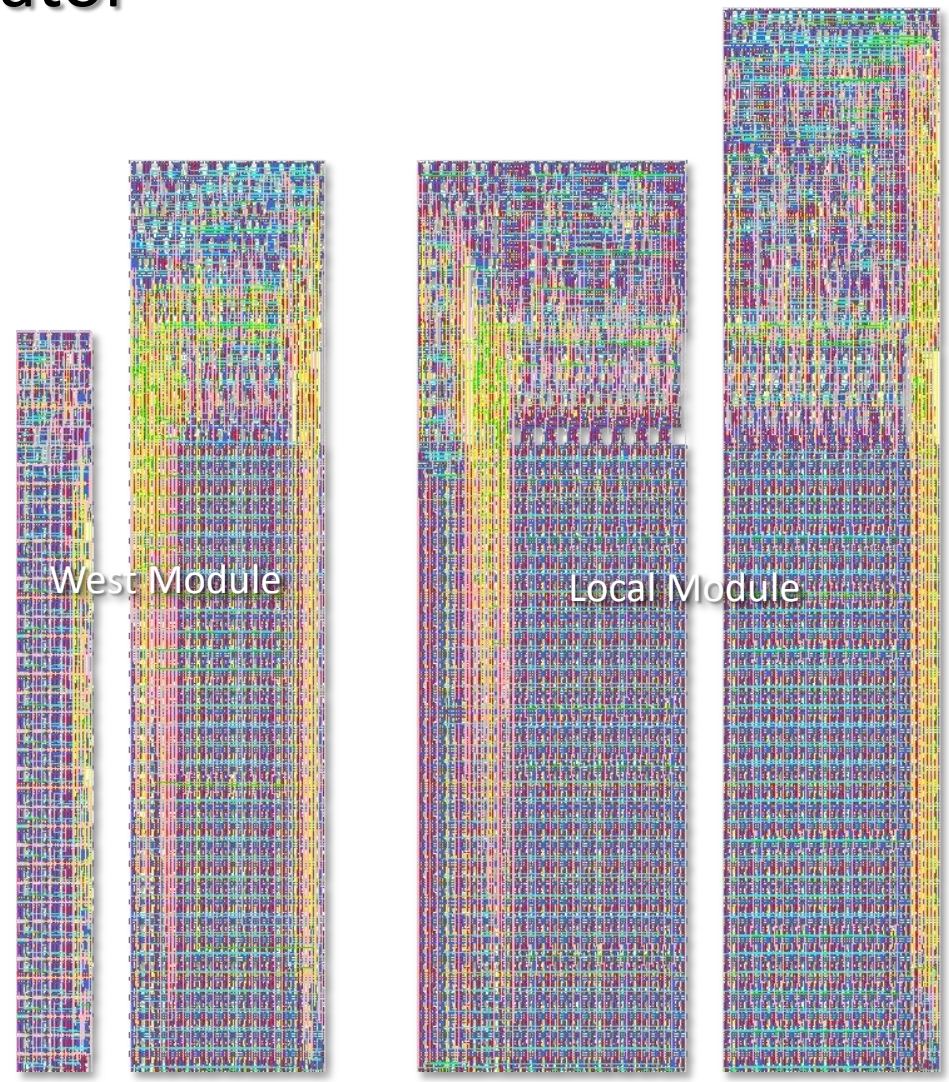
i	pi	Q
0	x	0
1	1	1
1	0	Q’

Router General Architecture



ASPIN Generator

- a Generic ASPIN Generator
  - Two Parameters
    - FIFO Depth
    - Word Length
  - Two Products
    - Gate-level Net-List
    - Physical Layout
- 47672 Transistors for a Router with:
  - 8-Word FIFO Depth
  - 32-Bit Word Length

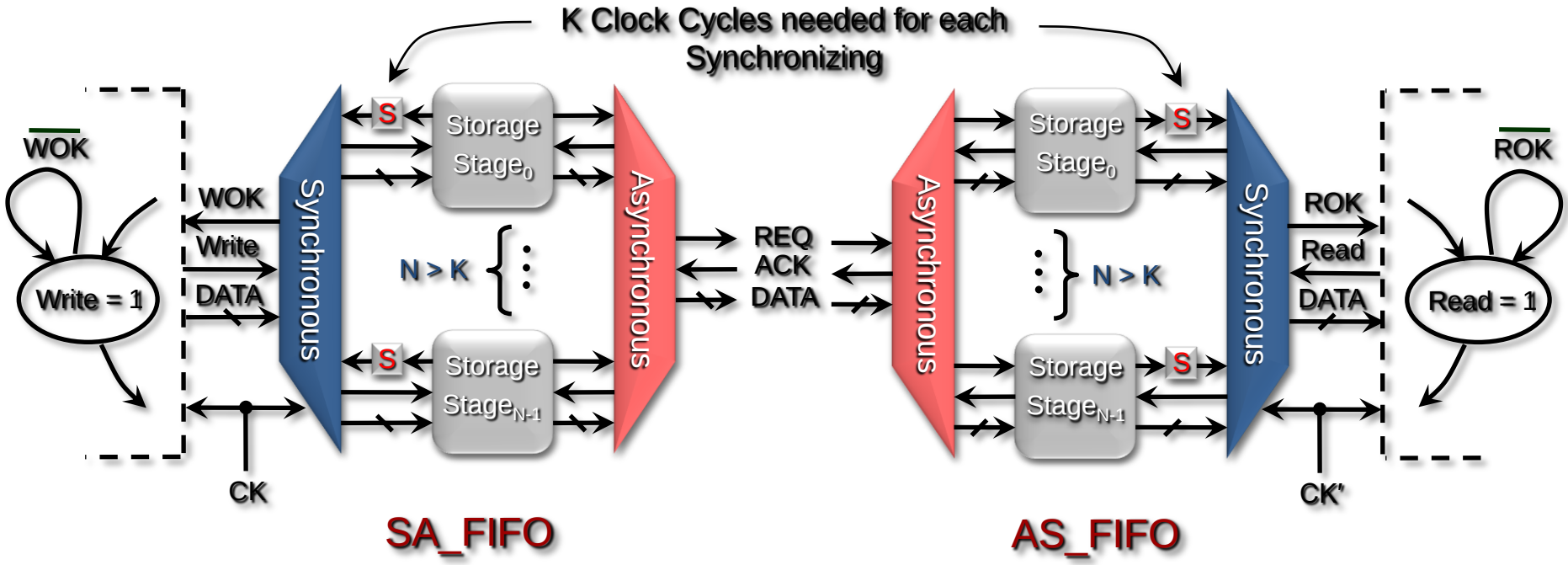


Outline

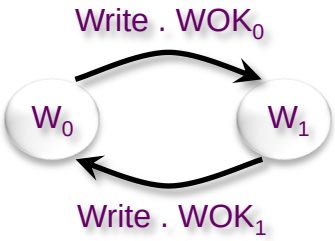
- GALS/DVFS and Asynchronous NoCs
- ASPIN an example of Async NoCs
- Async-Sync Interface
- Synchronous vs. Asynchronous
- 3D-Integration and Asynchronous NoCs
- Vertical Link Serialization
- Vertically-Partially-Connected 3D-NoC
- Conclusion

General Architecture

- Synchronizers on Handshake Signals toward Synchronous Side
- K Storage Stages (at least) to obtain Maximum Throughput
- K Clock Cycles as an Initial Latency



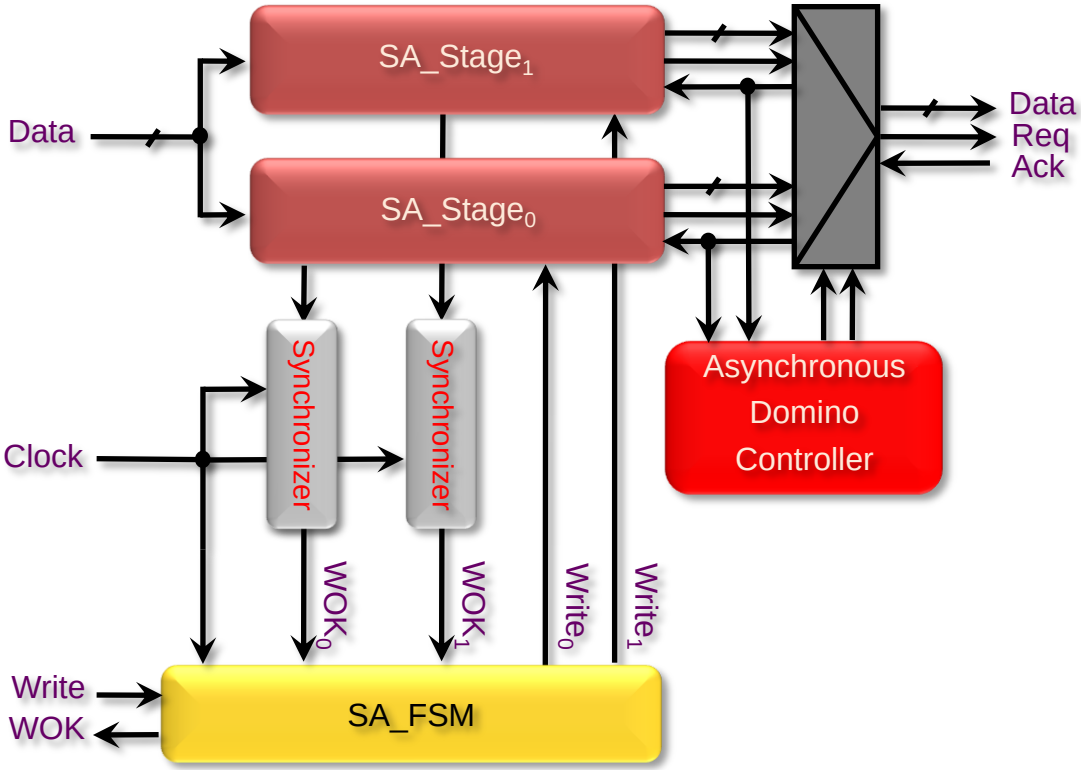
Synchronous-to-Asynchronous FIFO



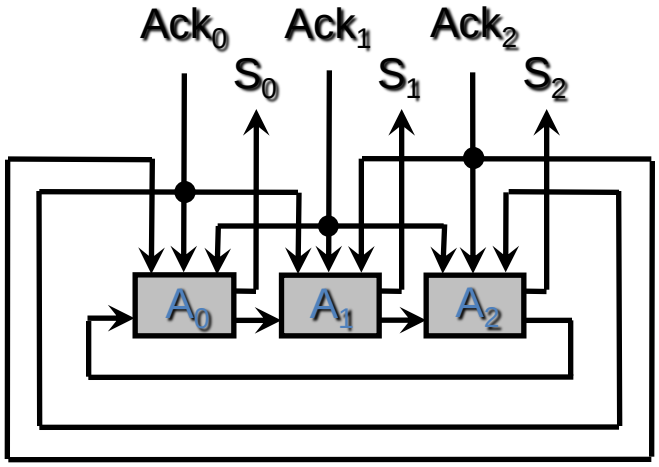
$WOK = WOK_0 . W_0 + WOK_1 . W_1$

$Write_0 = WOK_0 . W_0 . Write$

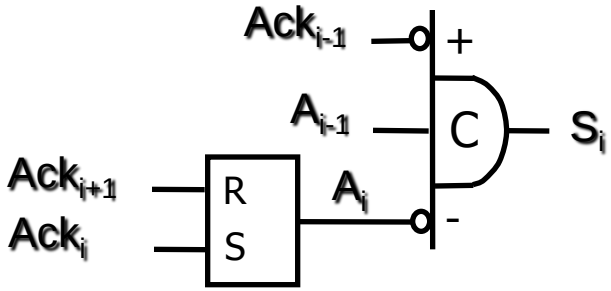
$Write_1 = WOK_1 . W_1 . Write$



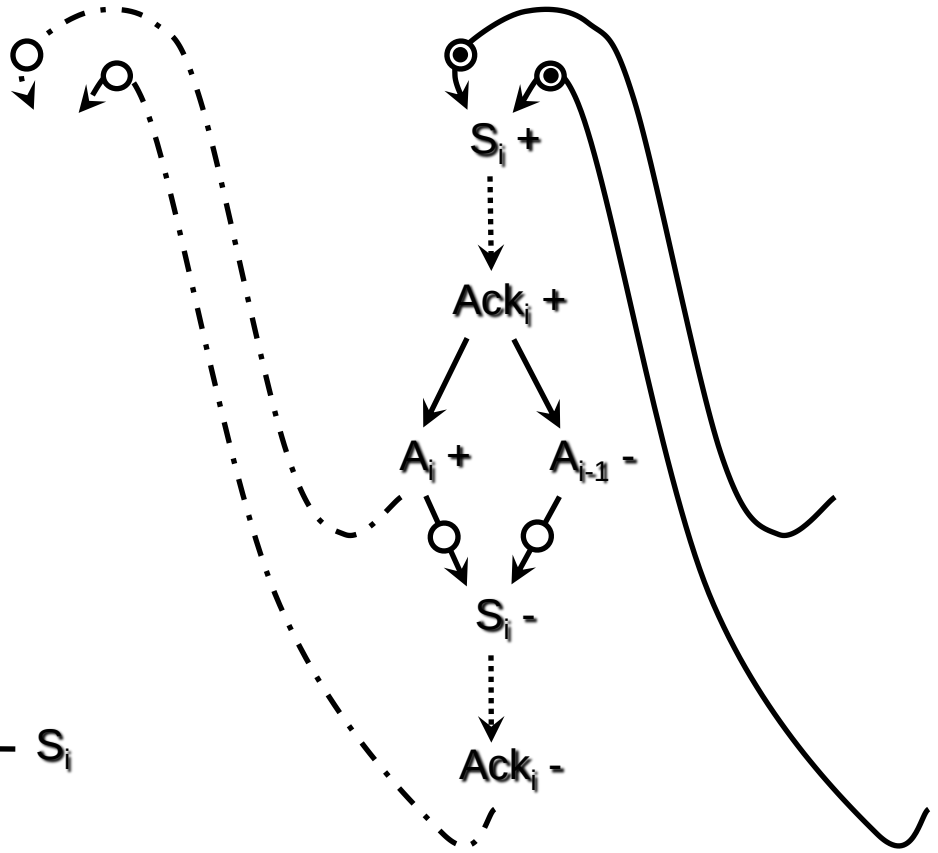
Asynchronous Domino Controller



3-Bit Domino Controller

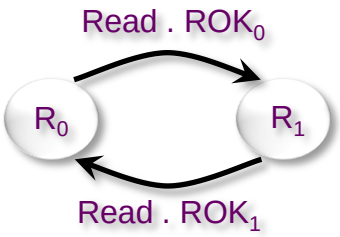


Synthesized Circuit (Cell)



Signal Transition Graph (Cell)

Asynchronous-to-Synchronous FIFO



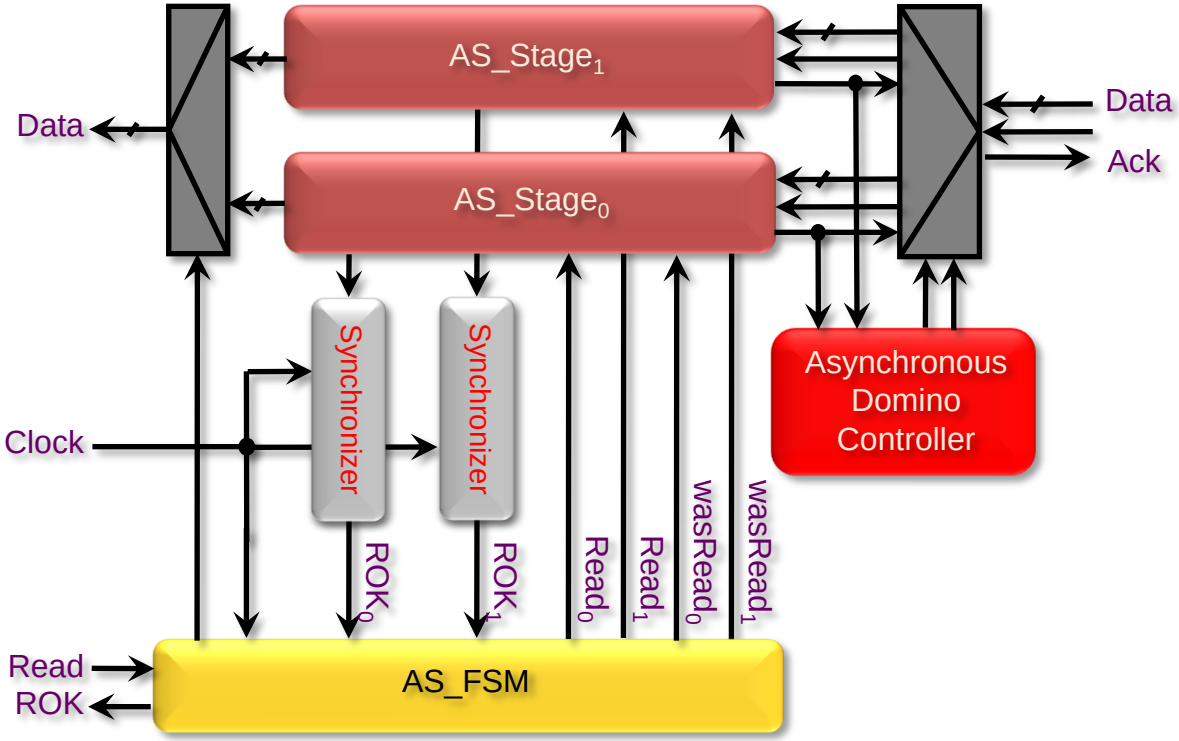
$ROK = ROK_0 . R_0 + ROK_1 . R_1$

$Read_0 = ROK_0 . R_0 . Read$

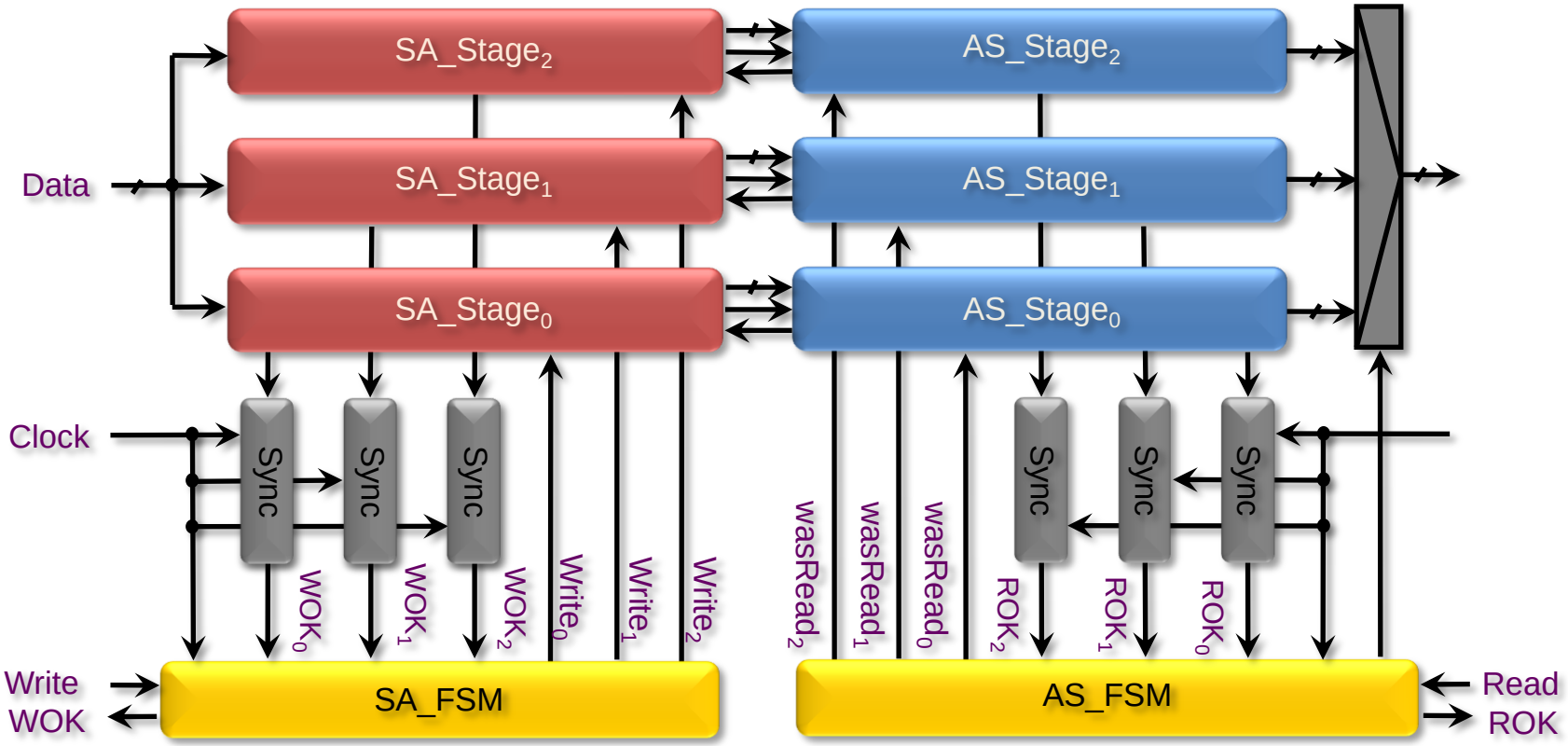
$Read_1 = ROK_1 . R_1 . Read$

$wasRead_0 = R_1$

$wasRead_1 = R_0$



Bi-Synchronous FIFO



Experimental Results

- STMicroelectronics CMOS 90nm GPLVT
- A boundless throughput of connected block as a Hypothesis for throughput measurement

Type	Transistors	Surface	Min Latency	Max Latency	Max Throughput
2-Place SA_FIFO	1338	1422 µm²	177 pS		2.39 GEvents/S
3-Place SA_FIFO	1969	2054 µm²	207 pS		2.36 GEvents/S
8-Place SA_FIFO	5126	5215 µm²	219 pS		2.22 GEvents/S
2-Place AS_FIFO	1388	1452 µm²	271 pS + T	271 pS + 2T	1.50 GEvents/S
3-Place AS_FIFO	1942	2011 µm²	247 pS + T	247 pS + 2T	2.61 GEvents/S
8-Place AS_FIFO	5054	5107 µm²	263 pS + T	263 pS + 2T	2.89 GEvents/S
6-Place SS_FIFO	2985	2940 µm²	362 pS + T	362 pS + 2T	2.61 GEvents/S
8-Place SS_FIFO	3956	3869 µm²	366 pS + T	366 pS + 2T	4.60 GEvents/S

Outline

- GALS/DVFS and Asynchronous NoCs
- ASPIN an example of Async NoCs
- Async-Sync Interface
- Synchronous vs. Asynchronous
- 3D-Integration and Asynchronous NoCs
- Vertical Link Serialization
- Vertically-Partially-Connected 3D-NoC
- Conclusion

Will future NoCs be Synchronous or Asynchronous ?

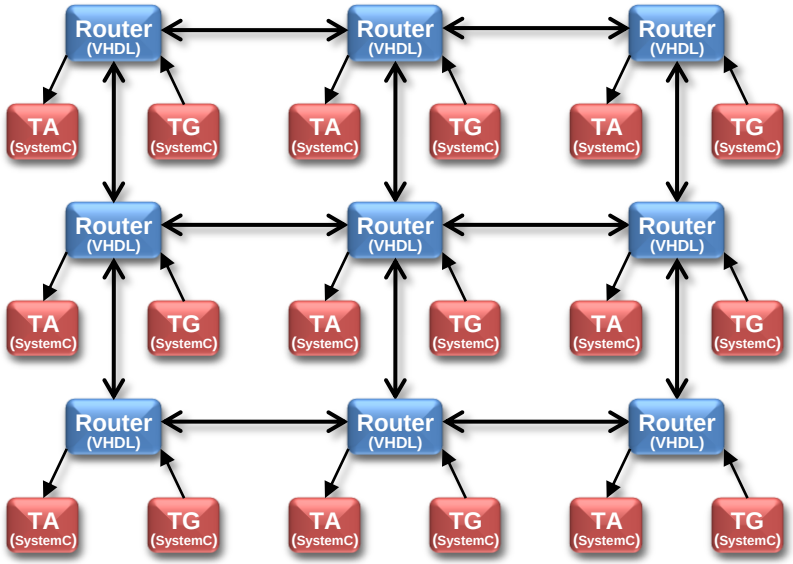
- Saturation Threshold ?
- Silicon Area ?
- Communication Throughput ?
- Packet Latency ?
- Power Consumption ?

Saturation Threshold

- Any Interconnect saturates when the average Offered Load reaches a point called Saturation Threshold
- Saturation Threshold Improvement is the main motivation supporting the NoC Paradigm
- The exact value depends on:
  - Traffic Load
  - Average Packet Length
  - Destination Distribution
  - Distributed Storage Capacitance

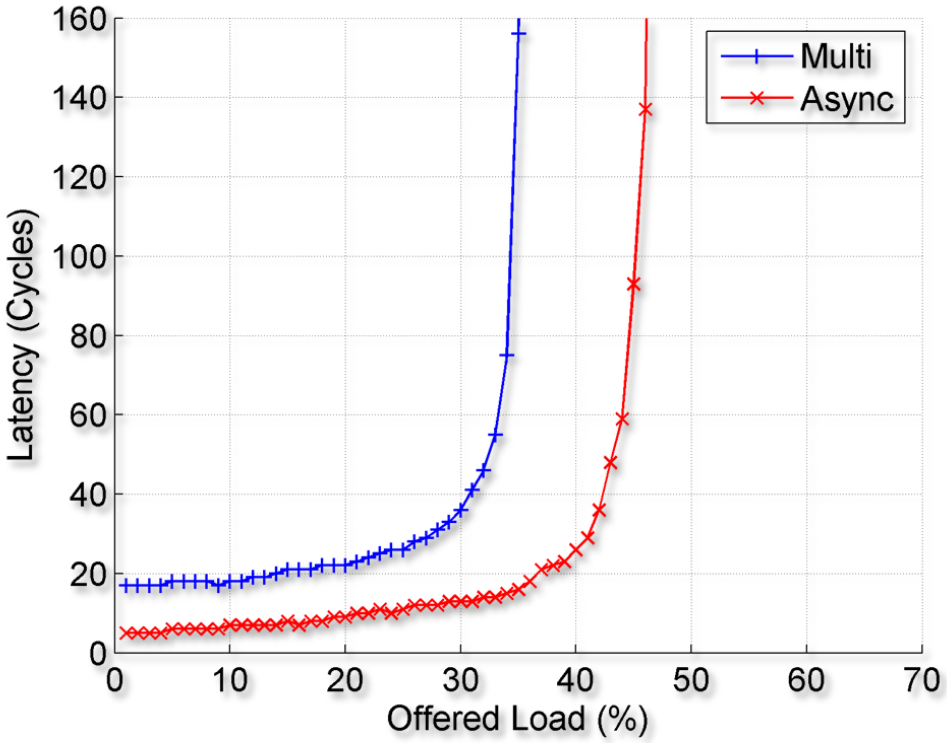
Simulation Platform …

- VHDL Model for ASPIN Router
  - Structural Net-List of Standard Cells
  - Behavior Model for Asynchronous Cells developed Manually
- Simulation Platform developed in SystemC
  - Traffic Generator-Analyzer
- SystemC-VHDL Co-Simulation under *ModelSim*



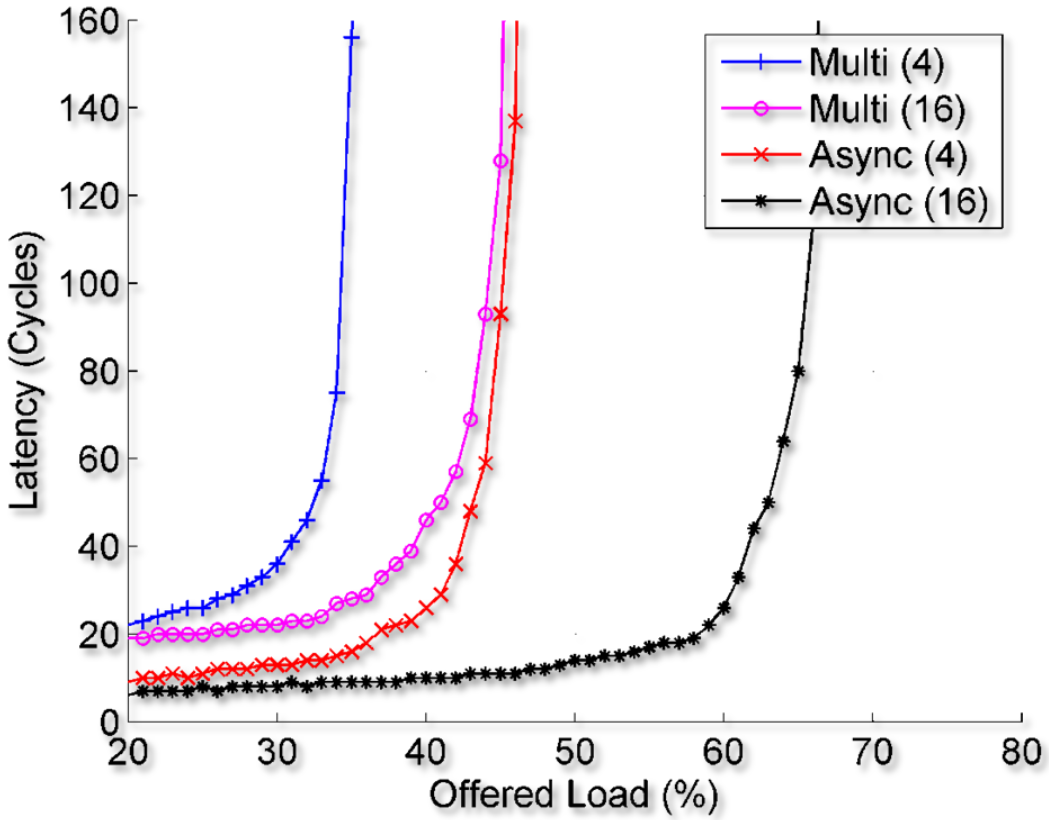
Synchronous vs. Asynchronous

- Uniform Random Distribution for Destinations
- Network of 5×5 Clusters
- Packet Length of 16 Flits
- FIFO Depth of 4 Words
- Speed Ratio of 5



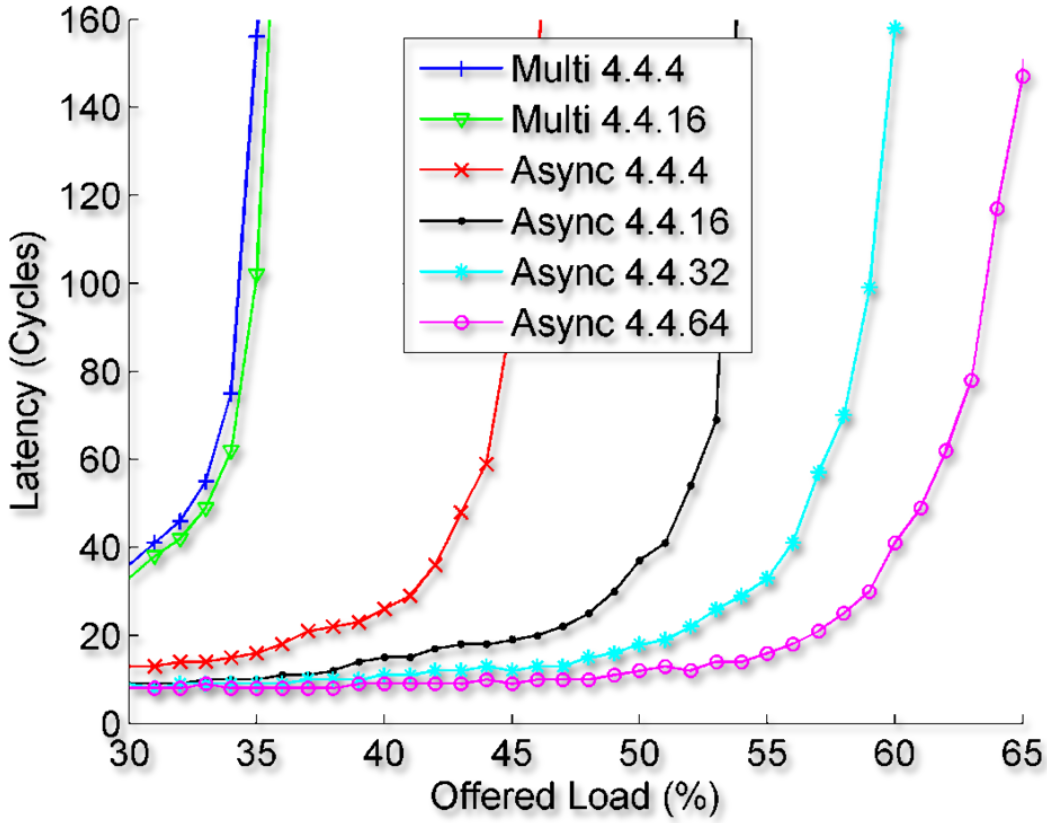
FIFO Depth Impact …

- FIFO Depth of 4 and 16 Words



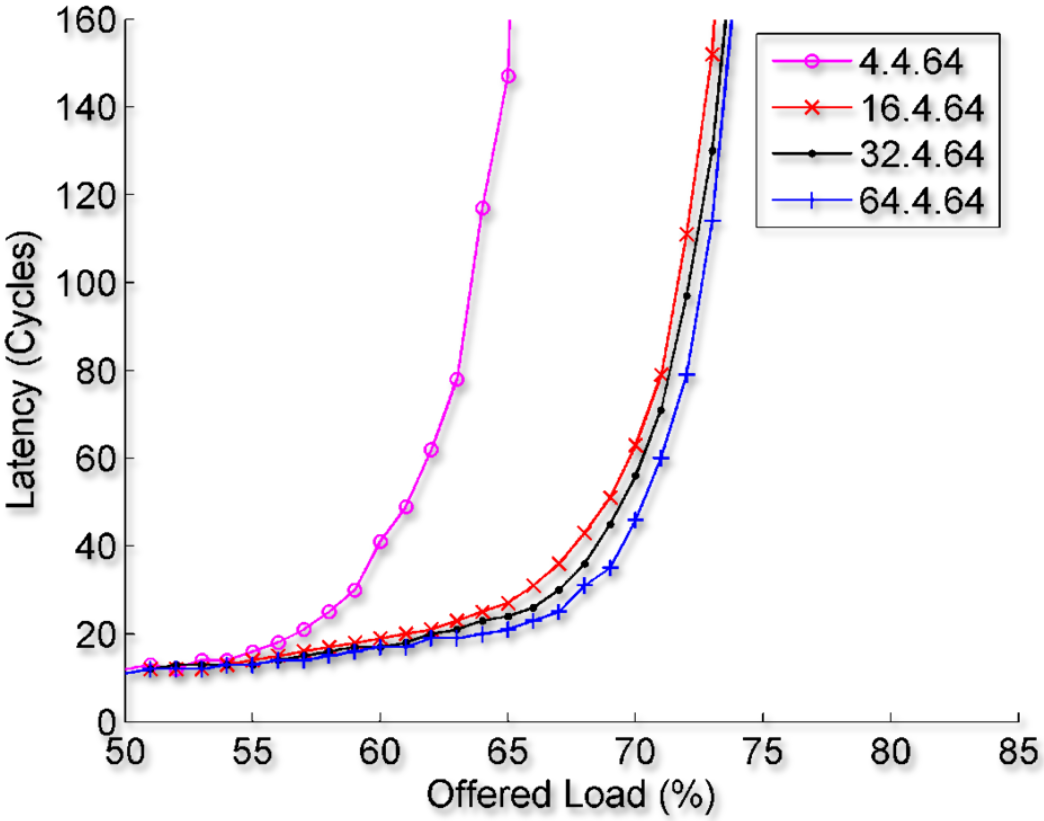
The Last FIFO Depth Impact …

- The Last FIFO (AS_FIFO) Depth of 4, 16, 32 and 64 Words



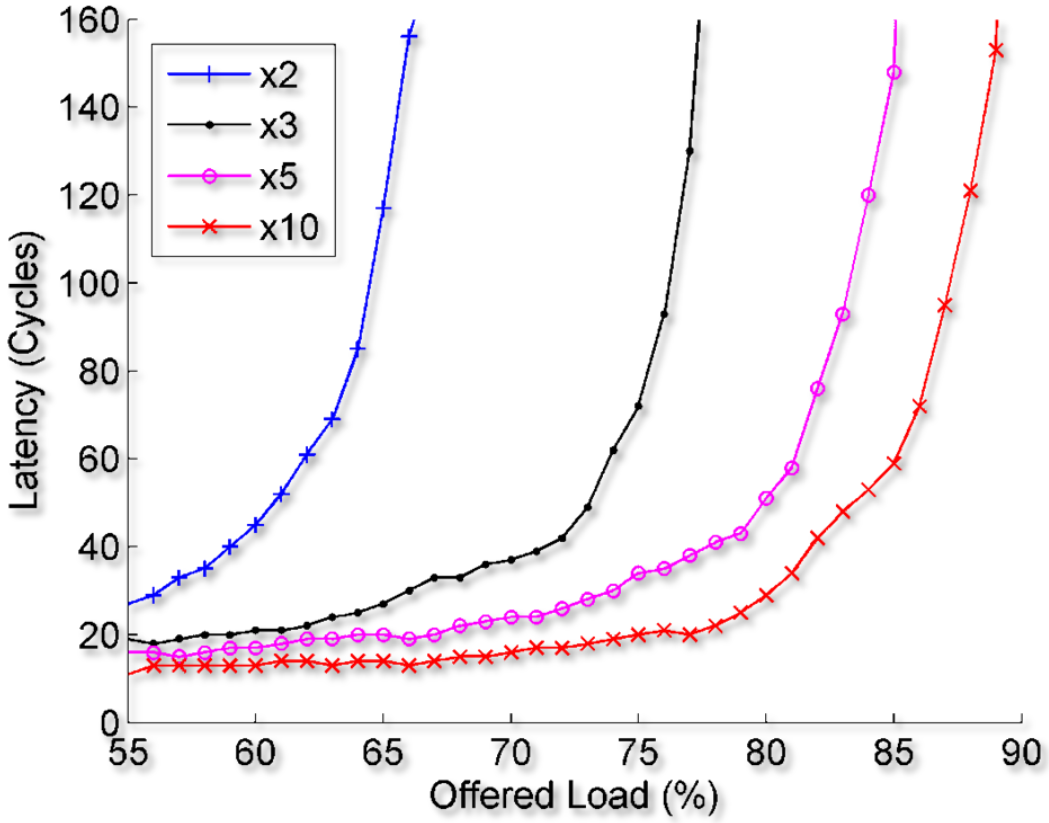
The First FIFO Depth Impact …

- The First FIFO (SA_FIFO) Depth of 4, 16, 32 and 64 Words



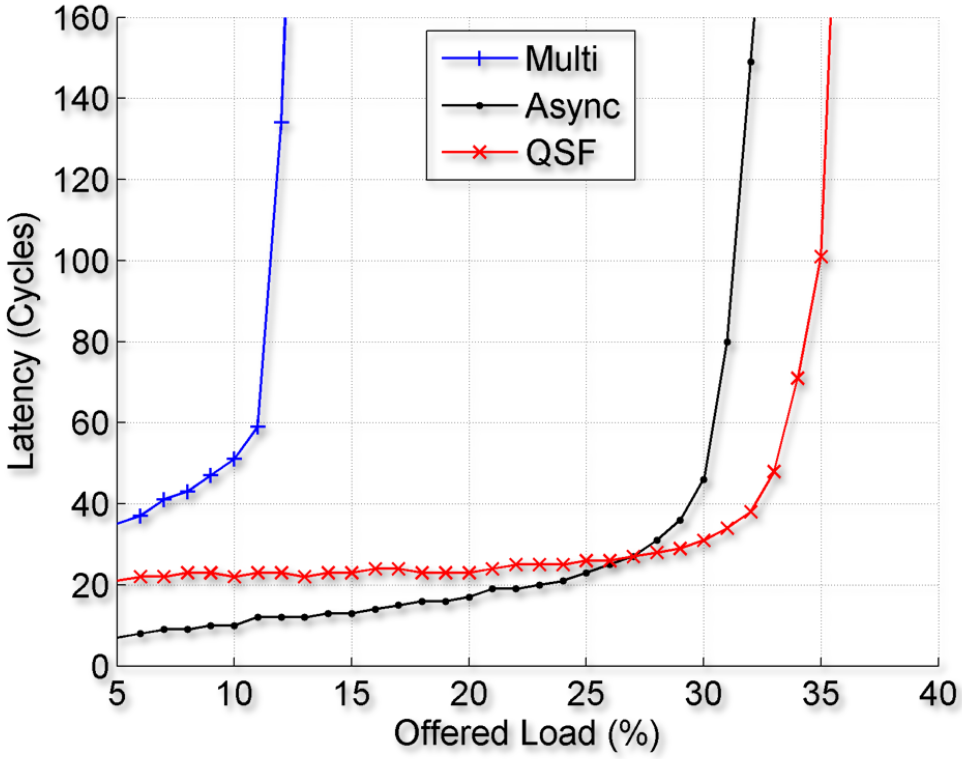
Network Throughput Impact …

- Speed Ratio of 2, 3, 5, and 10



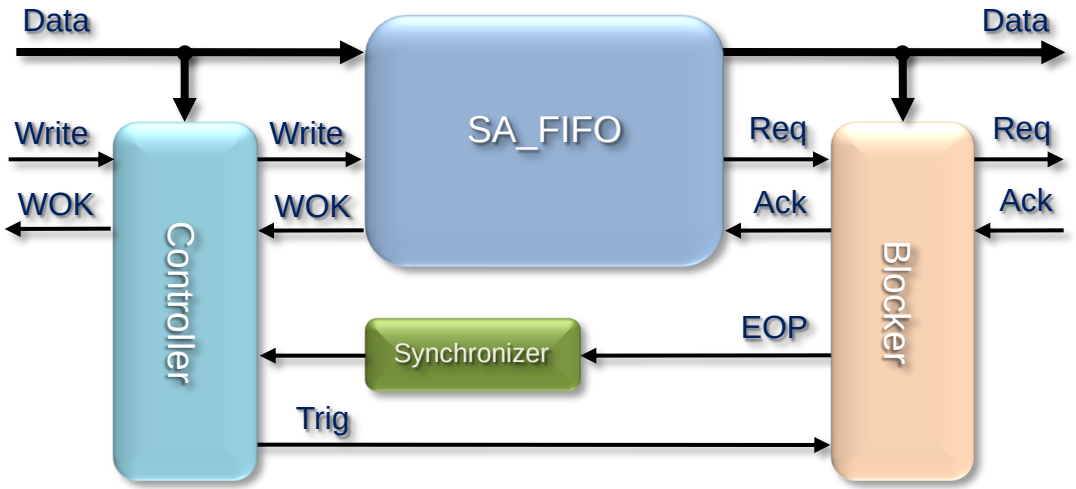
Network Size Impact …

- Network of 14×14 Clusters
- Quasi-Store-And-Forward
  - An Attempt to realize Offered Load Reduction by a coefficient equal to the speed ratio



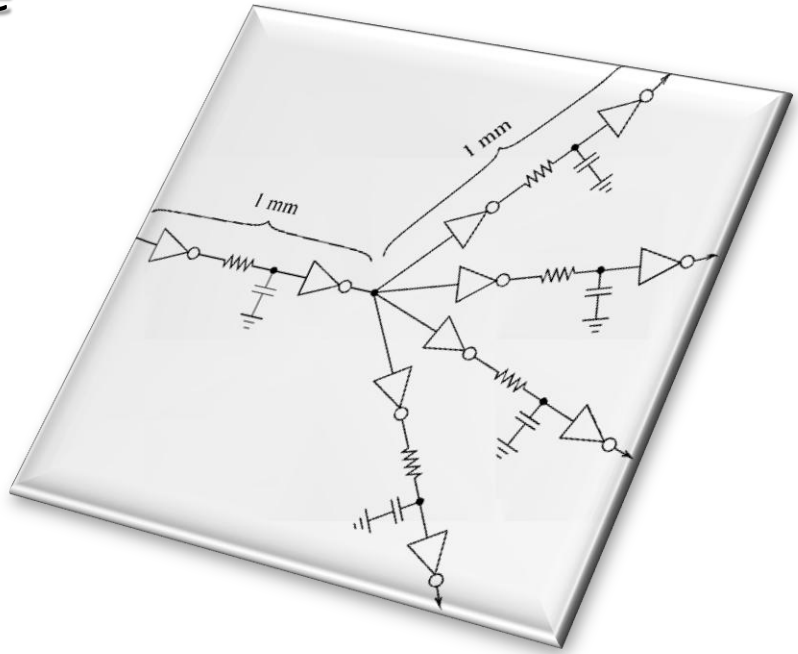
QSF Implementation

- Packets does not occupy any network resources unless it is sure it can traverse the network at full speed
- All flits of a given packet accumulate in the asynchronous form before entering the network



Electrical Characterization

- Transistor-Level Spice Model extracted for STMicroelectronics 90 nm GPLVT Technology
- Long Wires are considered in Electrical Simulations
  - An RC Model described in Spice
  - The length of 2 mm as a rough Estimation for Large Clusters



Silicon Area

- ASPIN 10% smaller than DSPIN, but …

  … Long Wire Buffers of ASPIN twice of DSPIN

- ASPIN and DSPIN have nearly similar Foot-Print if Long Wires are taken into account

	DSPIN	ASPIN
Router	40200 μm^2	36199 μm^2
Long Wire Buffers	4276 μm^2	7815 μm^2
Total	**44476 μm^2**	**44014 μm^2**

Communication Throughput

- Higher Communication Throughput in ASPIN
- Lower Long Wire Delay in DSPIN

- ASPIN and DSPIN have approximately equal Throughput applicable in Large Clusters

	DSPIN	ASPIN
Maximum Throughput	787 MFlits/S	1131 MFlits/S
Long Wire Effect	135 ps	515 ps
Applicable Throughput	**711 MFlits/S**	**714 MFlits/S**

Packet Latency

- DSPIN : $L = 17 + (N-2) \times 5$ ns (at 500 MHz)
- ASPIN : $L = 6.6 + (N-2) \times 1.92$ ns (at 500 MHz)

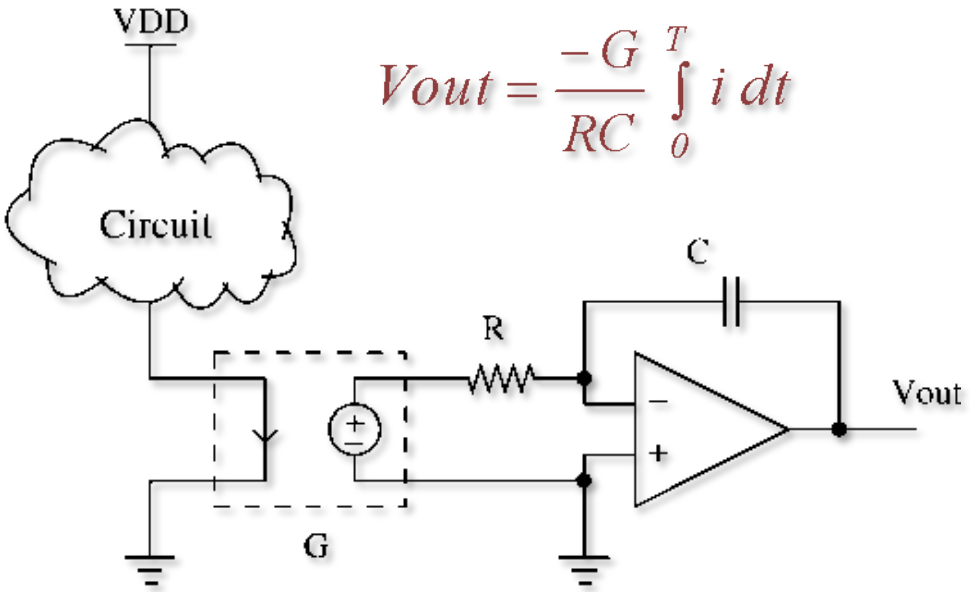
- ASPIN has a much Lower Packet Latency compared with DSPIN Latency caused by Shynchronizers

	DSPIN	ASPIN
First Router	3 ~ 4 T	1.06 ns
Intermediate Router	2.5 T	1.53 ns
Last Router	4.5 ~ 5.5 T	1.76 ns + 1~2 T
Long Wire Effect	0 ns	0.39 ns

Power Consumption

- Instantaneous Power Consumption during a Defined Period of Time
- Current Integrator Model to measure the Electrical Energy Consumption

```
.param One = 1v
vSupply VDD 0 One
*************************
*   Current Integrator   *
* VOUT : Consumed Energy *
*************************
hh1 p1 0 vSupply 1
rr1 p1 p2 1meg
cc1 p2 VOUT 1u ic=0
ee1 VOUT 0 0 p2 999k
*************************
```



$$Vout = \frac{-G}{RC} \int_0^T i\,dt$$

Power Consumption

- ASPIN and DSPIN have approximately the same Power Consumption if Long Wires are taken into account and Activation Rate is about 20%

	DSPIN	ASPIN
Idle Router	2060 µWatt	640 µWatt

With Long Wire Effect (pJ/Flit)				
Transmission Path	With Constant Content		With Alternate Content	
	DSPIN	ASPIN	DSPIN	ASPIN
	9×(N-2)+19	25.8×(N-2)+54.2	16.2×(N-2)+32.8	27.4×(N-2)+58.4

So what?

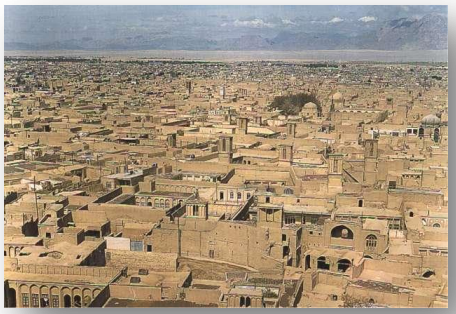
- For both of DSPIN (multisynchronous) and ASPIN (asynchronous) the Silicon Area, Power Consumption, and Throughput have similar values
- The end-to-end Latency of the fully asynchronous approach is at least two times smaller, and the saturation threshold is much higher

Outline

- GALS/DVFS and Asynchronous NoCs
- ASPIN an example of Async NoCs
- Async-Sync Interface
- Synchronous vs. Asynchronous
- 3D-Integration and Asynchronous NoCs
- Vertical Link Serialization
- Vertically-Partially-Connected 3D-NoC
- Conclusion

… when the population grows!

- As the population grows, there is a tendency to build vertically rather than horizontally
  - Increase the density
    - The land becomes more and more expensive
  - Decrease the length and the number of long paths
    - The average time and energy of moving from one point to another becomes unaffordable



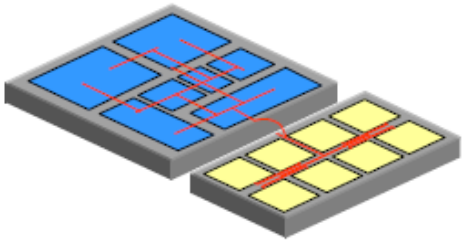
Three-Dimensional Integration

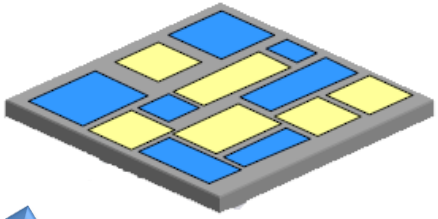
System-on-Board

System-on-Chip

no External Connections

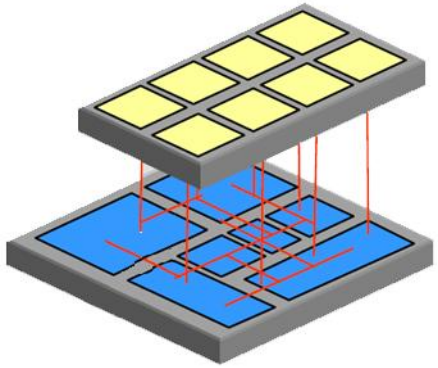
Large Die



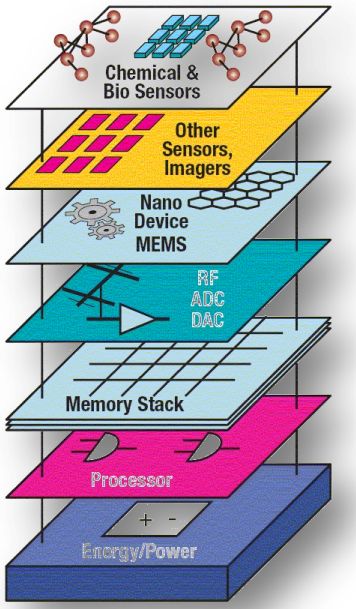


Short and Direct Connections

Small Dies

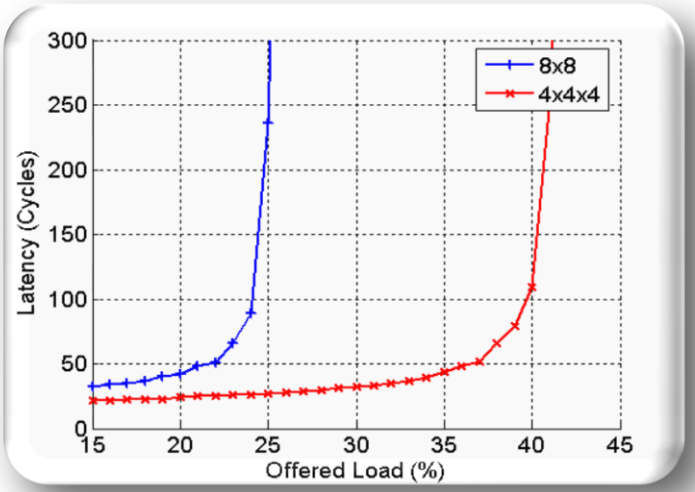


* Possibility of Integrating Different Dies with Different Technologies in a Same Package



*James Lu

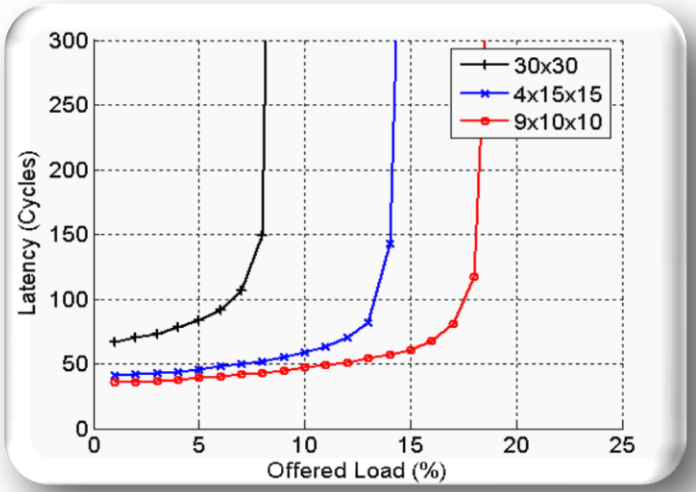
The Third Dimension

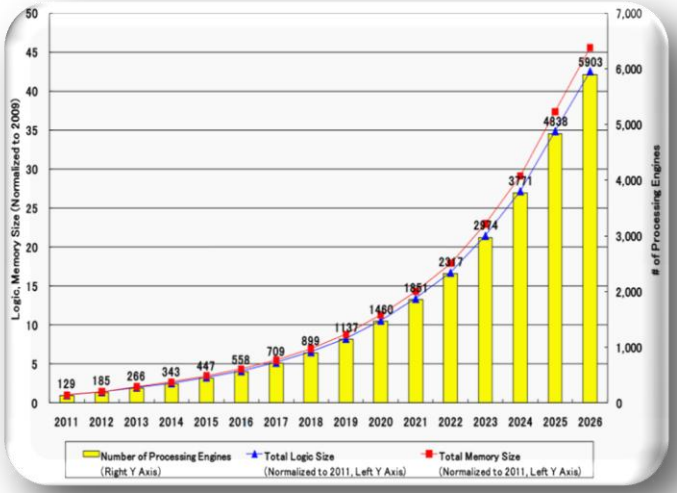


	Number of Nodes	Switch Degree	Network Diameter	Number of Channels	Number of Vertical Channels	Number of Bisection Channels	Load of the Busiest Channels [1]
2D-Mesh	$N = n^2$	5	$2\sqrt{N}$	$6N - 4\sqrt{N}$	0	$2\sqrt{N}$	$C \times \frac{1}{4}\sqrt{N}$
3D-Cube	$N = m^3$	7	$3\sqrt[3]{N}$	$8N - 6\sqrt[3]{N^2}$	$2N - 2\sqrt[3]{N^2}$	$2\sqrt[3]{N^2}$	$C \times \frac{1}{4}\sqrt[3]{N}$

[1] Assuming uniform destination distribution and dimension-ordered routing, C is the average load injected to the network by each node

How Many Layers?



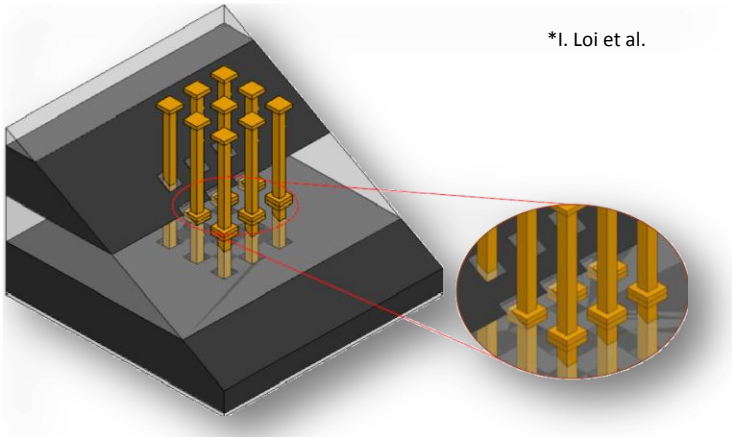


	Number of Nodes	Switch Degree	Network Diameter	Number of Channels	Number of Vertical Channels	Number of Bisection Channels	Load of the Busiest Channels (1)
30x30	900	5	60	5280	0	60	C × ¼ × 30
4x15x15	900	7	34	6510	1350	120	C × ¼ × 15
9x10x10	900	7	29	6640	1600	180	C × ¼ × 10

(1) Assuming uniform destination distribution and dimension-ordered routing, C is the average load injected to the network by each node

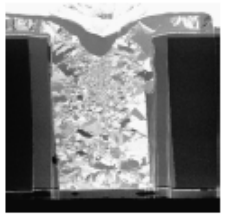
Through-Silicon-Via

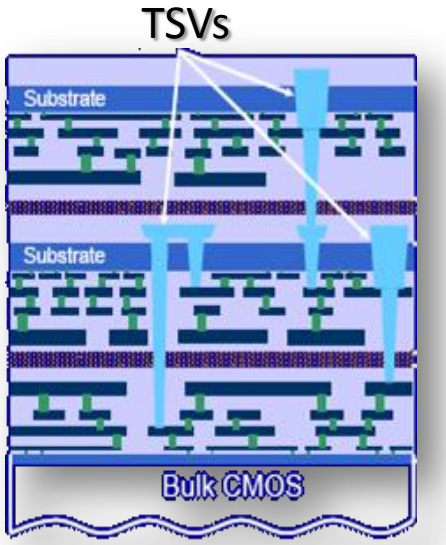
- The most promising Technology of Vertical Interconnection
  - Low Resistance and Capacitance
    - High Bandwidth
    - Low Power Consumption



- Via-First (higher density of TSVs)
  - Diameter ≈ 5 µm
  - Pitch ≈ 10 µm
  - Depth ≈ 20-50 µm

- Via-Last (lower cost of the process)
  - Diameter ≈ 35 µm
  - Pitch ≈ 50 µm
  - Depth ≈ 40-150 µm



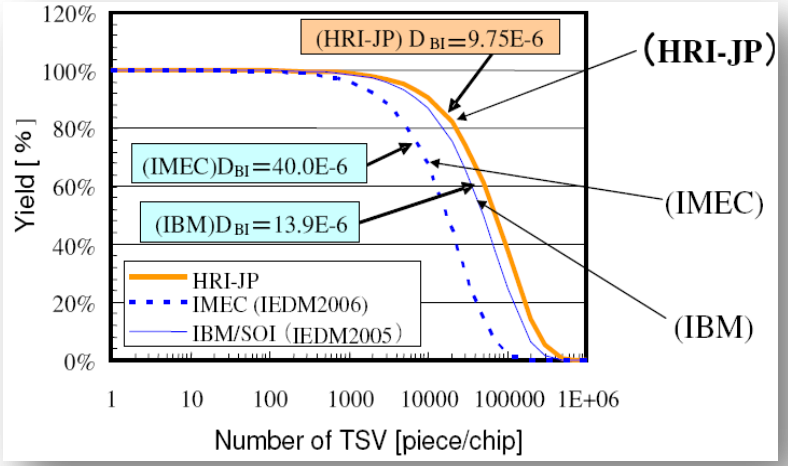


…but, is there any problem ?

- Large area overhead because of large TSV pitch, mainly due to the large pads to compensate misalignment of dies

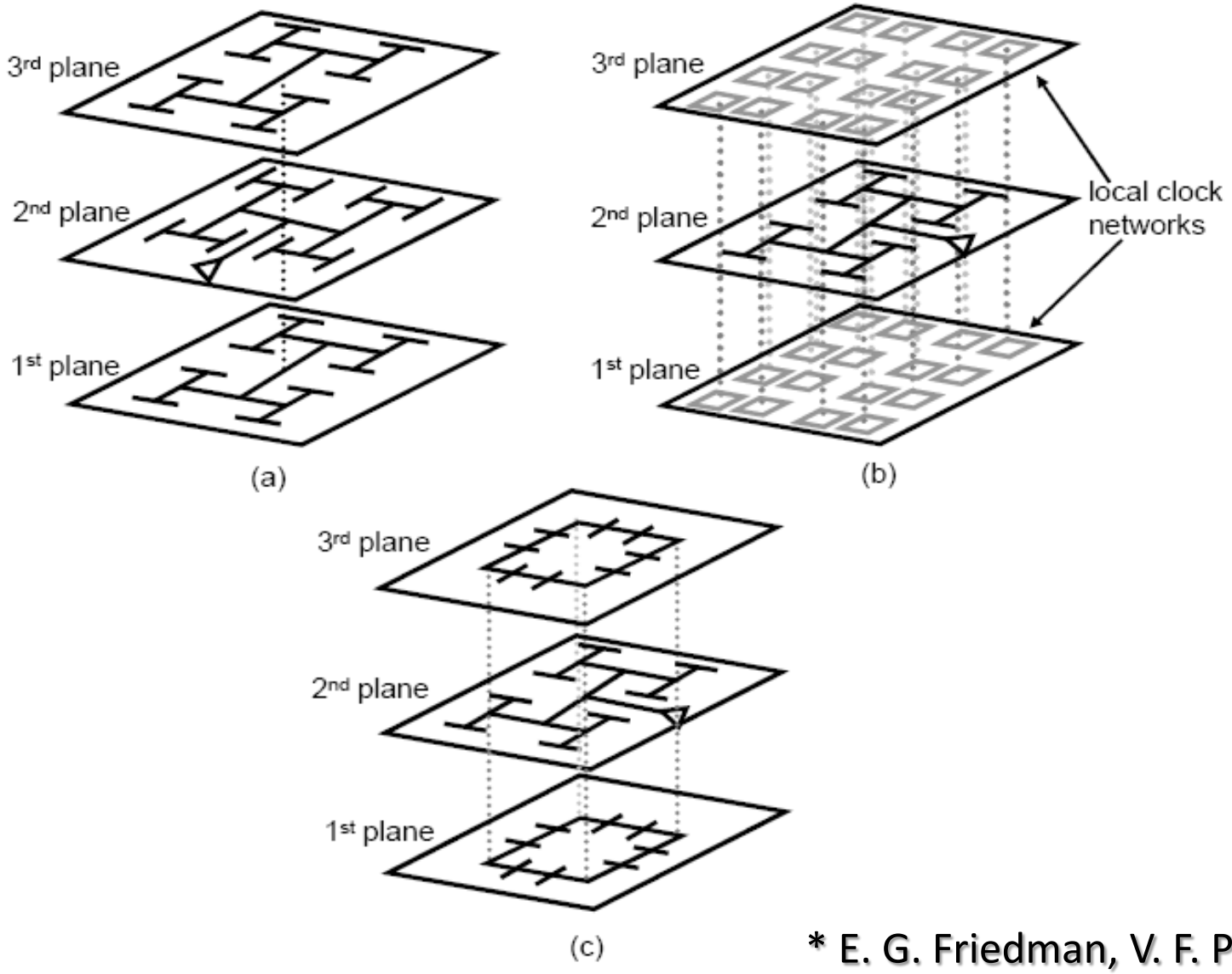
- Important risk of failure due to several additional fabrication steps (a potential reduction on the Yield)
  - Misalignment
  - Dislocation
  - Void formation
  - Oxide film formation over Copper interfaces
  - Pad detaching
  - Defects due to temperature
  - …

The Three-Dimensional Integrated Circuits are limited by the number of TSVs to be exploited



*C. Seiculescu et al.

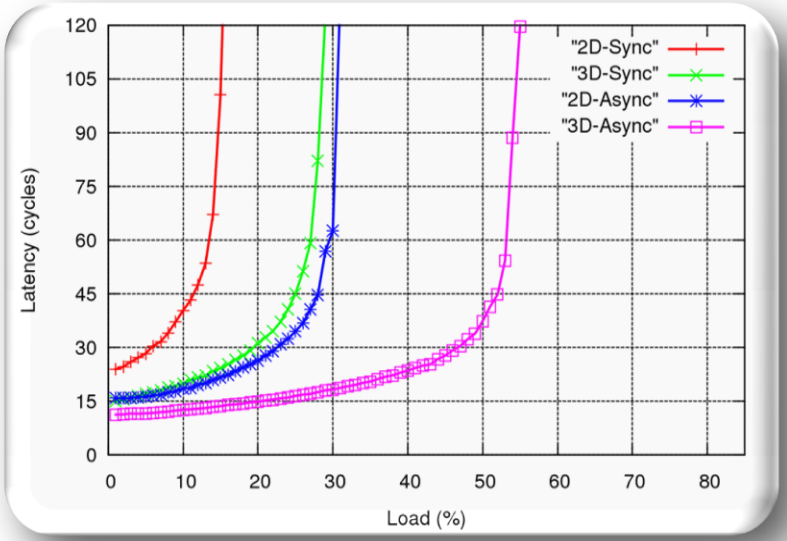
Clock Distribution in 3 Dimensions

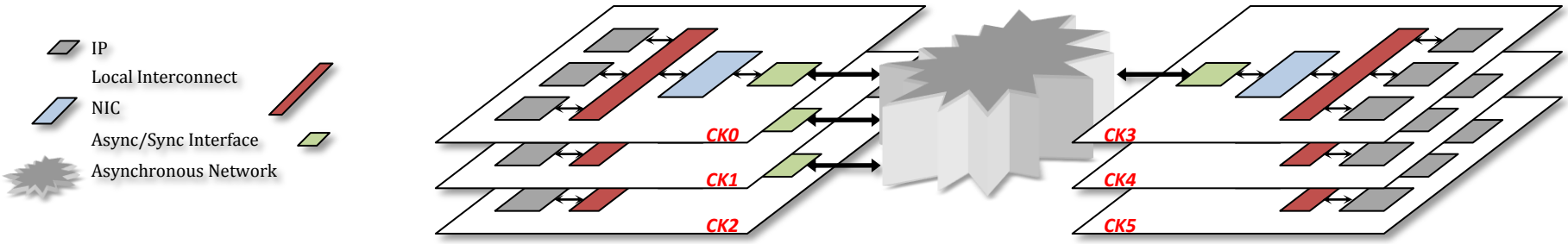


* E. G. Friedman, V. F. Pavlidis

Asynchronous 3D-NoC

- Insensitive to Delay Variation due to Temperature Variation or Process Variation
- Exploitation of the whole high Bandwidth of TSVs
- Speed ratio of 2 as a worst-case assumption
  - Using STMicroelectronics 90nm GPLVT transistors, 400MHz as the maximum frequency of usual SoCs
  - Using the same technology, 1100 Mflits/s as throughput of an asynchronous NoC





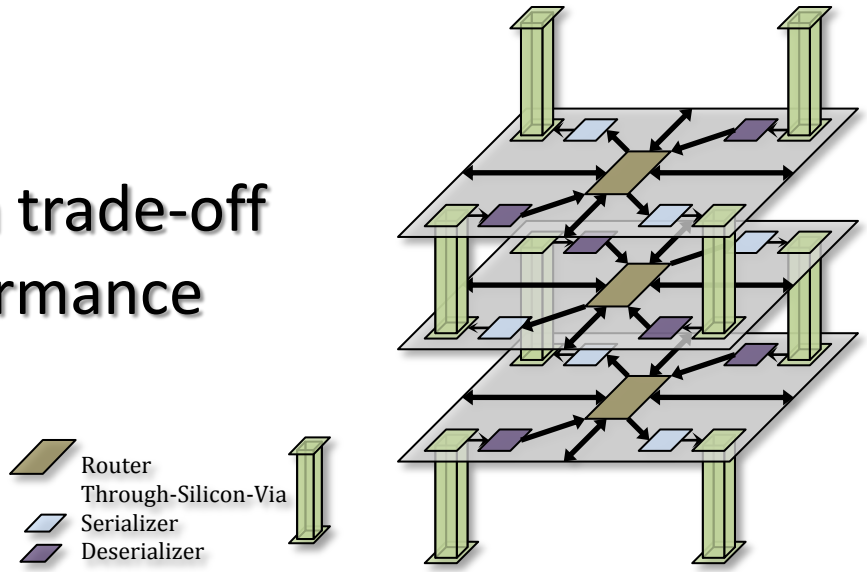
Outline

- GALS/DVFS and Asynchronous NoCs
- ASPIN an example of Async NoCs
- Async-Sync Interface
- Synchronous vs. Asynchronous
- 3D-Integration and Asynchronous NoCs
- Vertical Link Serialization
- Vertically-Partially-Connected 3D-NoC
- Conclusion

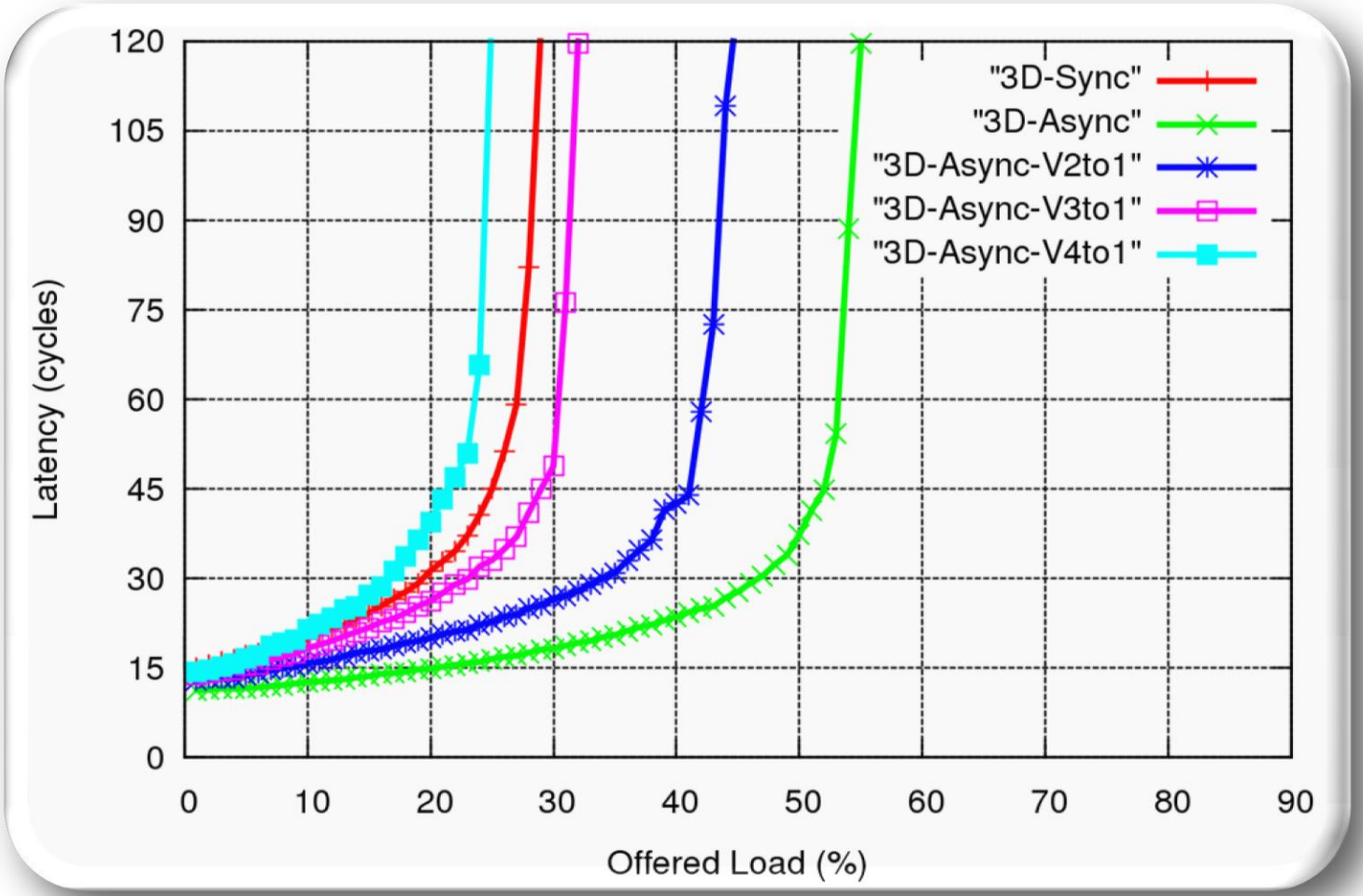
… why not Serialized Vertical Links!

- Remembering
  - Using TSVs guarantees a faster vertical data transfer with lower power consumption than horizontal links in moderate size
  - but, the Pitch of TSVs is large, and, several additional steps of TSV fabrication add a potential reduction of the Yield
  - Only a small fraction of the capacity of vertical link is exploited in a NoC
  - Large number of physical connections for each link of the router

- Serialization of data on TSVs is a trade-off between the cost and the performance



Vertically Serialized Asynchronous 3DNoC



Circuit Implementation

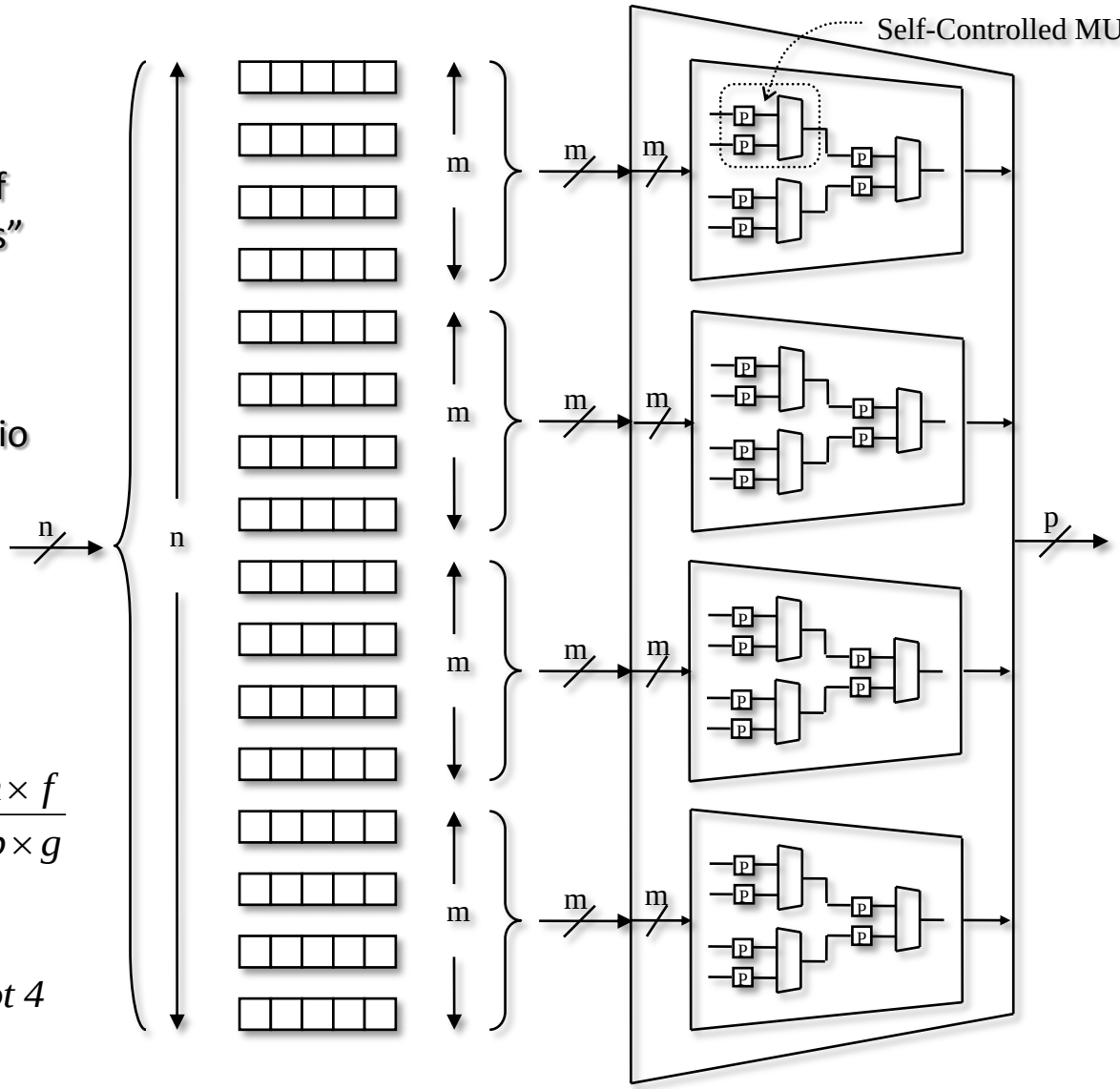
- a Serializer of n:p composed of p Serializer of m:1
  - a Serializer of m:1 is a tree of "Self-Controlled Multiplexors"

$$m = Serialization\ Ratio = \frac{n}{p}$$

- *R*, The Serialization Bandwidth Ratio as the throughput cost factor
  - *f*, the transfer rate of parallel input data
  - g, the transfer rate of serialized output data

$$R = Serialization\ Bandwidth\ Ratio = \frac{n \times f}{p \times g}$$

$$R = \frac{32 \times 750 Mflits/s}{8 \times 2800 Mflits/s} = 1.07, \text{ and not } 4$$



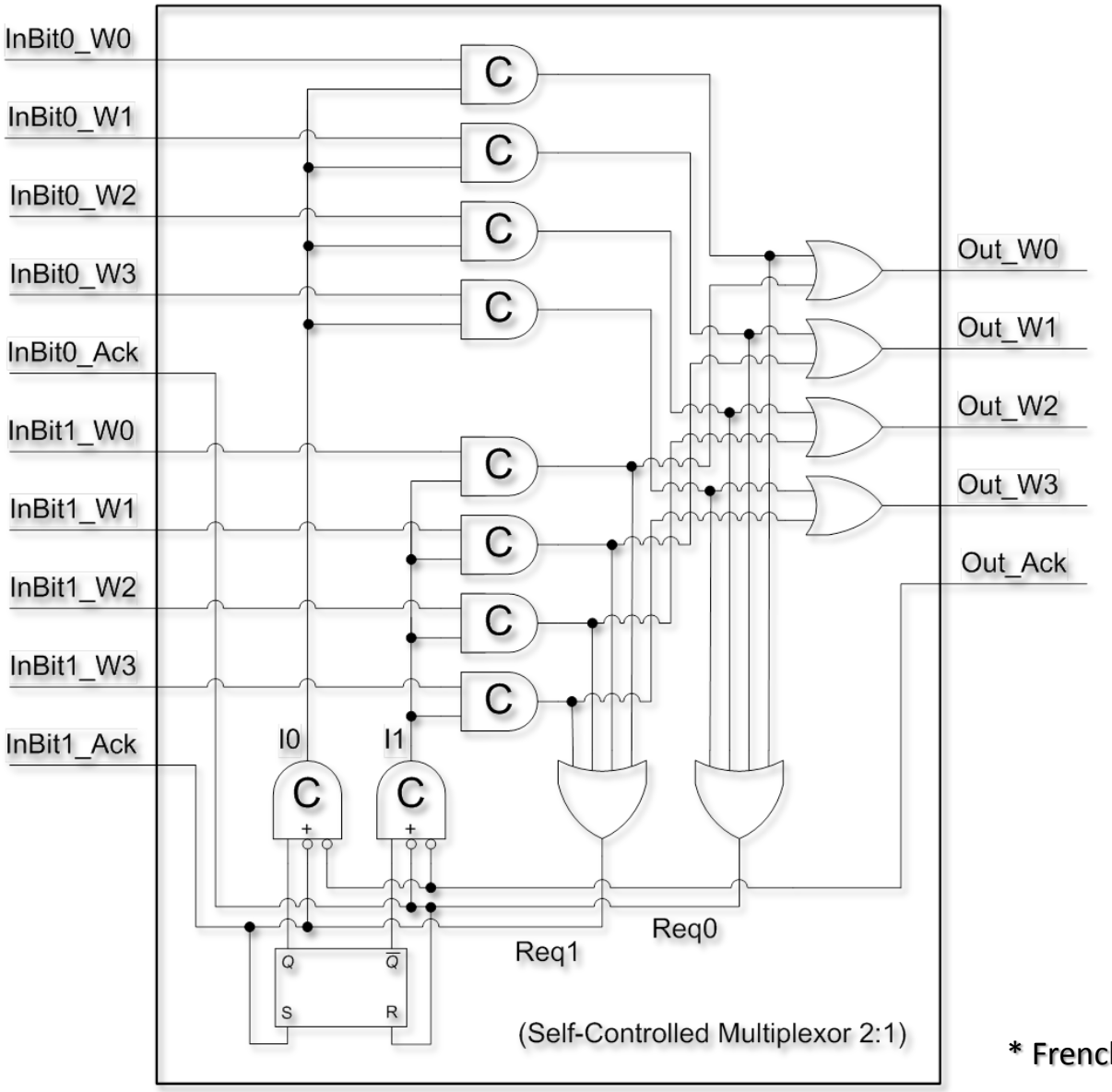
SPICE Simulation Results

- Horizontal Link Throughput: 710 Mflits/sec
  - Router Throughput : 1100 Mflits/sec
  - Inter-Core wire (2mm) delay : 125 ps
- Serialized (8:1) Vertical Link Throughput: 2080 Mflits/sec
  - Serialization Throughput: 2500 Mflits/sec
  - TSV delay: 20 ps

- Speed ratio : (710*32)/(2080*4) = 2.73 (and not 8 !)

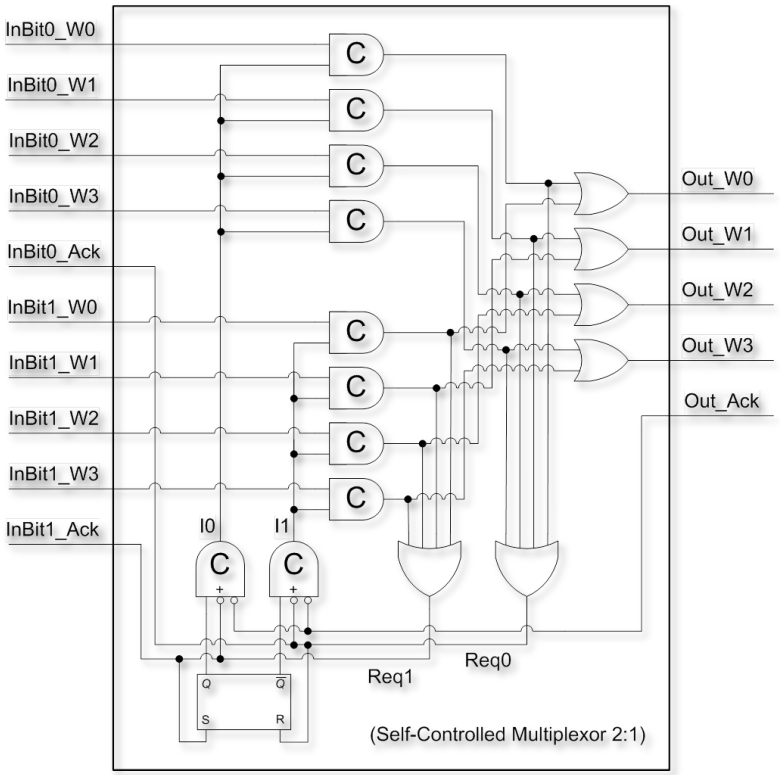
	Self-Controlled Multiplexer 2:1	Self-Controlled Demultiplexor 1:2	Serializer 4:1	Deserializer 1:4	Serializer 8:1	Deserializer 1:8
Transistor count	130	132	390	396	910	924
Latency	80 ps	70 ps	150 ps	130 ps	220 ps	190 ps
Throughput	2.9 Gflits/sec	3.2 Gflits/sec	2.5 Gflits/sec	2.8 Gflits/sec	2.5 Gflits/sec	2.8 Gflits/sec

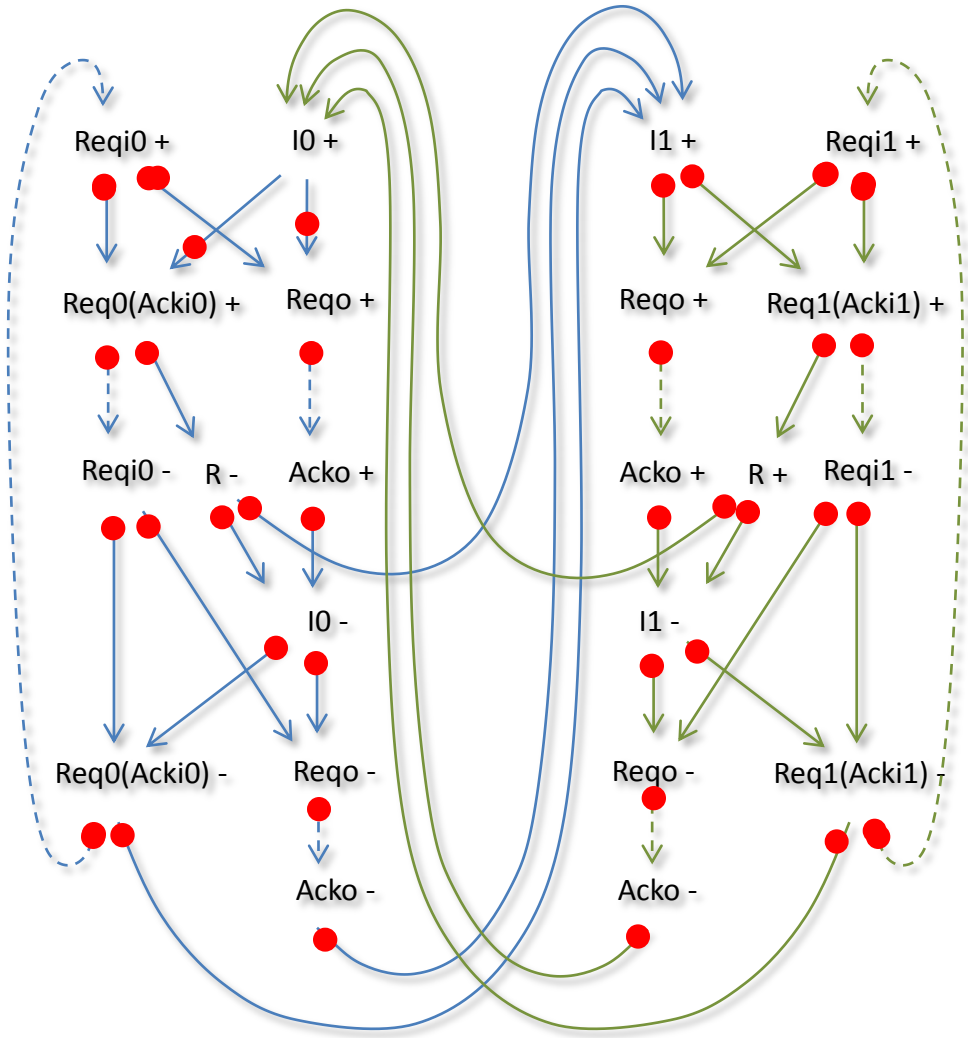
Self-Controlled Multiplexor



* French Patent 09/53637

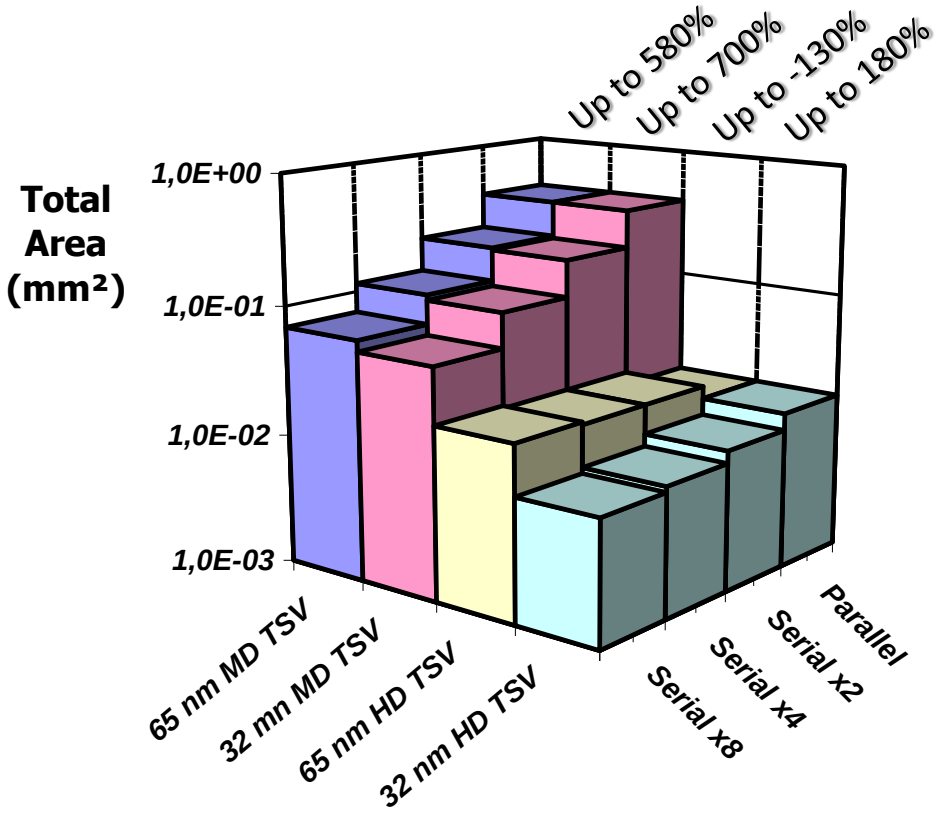
Signal Transitions





Serialization Area Cost Analysis

	MD TSV	HD TSV	65 nm	32 nm
Parallel	0.4 mm²	0.016 mm²	0 mm²	0 mm²
Serial x2	0.2 mm²	0.008 mm²	0.012 mm²	0.0039 mm²
Serial x4	0.1 mm²	0.004 mm²	0.016 mm²	0.0056 mm²
Serial x8	0.05 mm²	0.002 mm²	0.019 mm²	0.0067 mm²

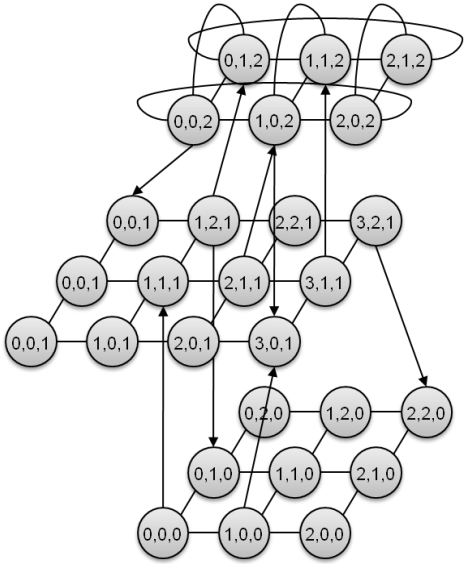


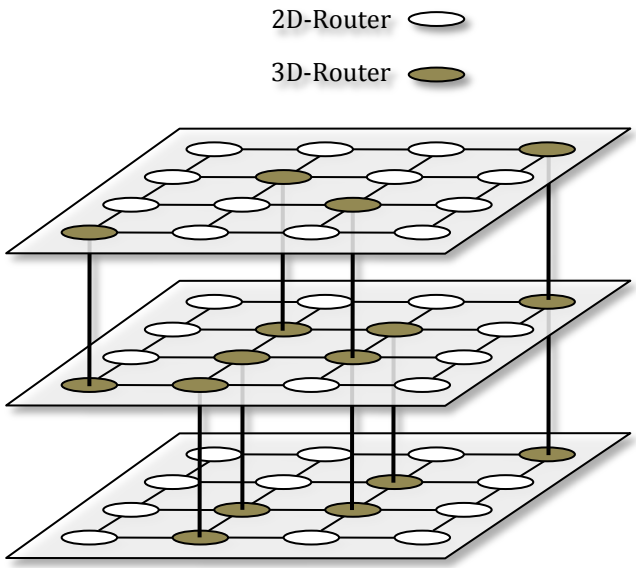
Outline

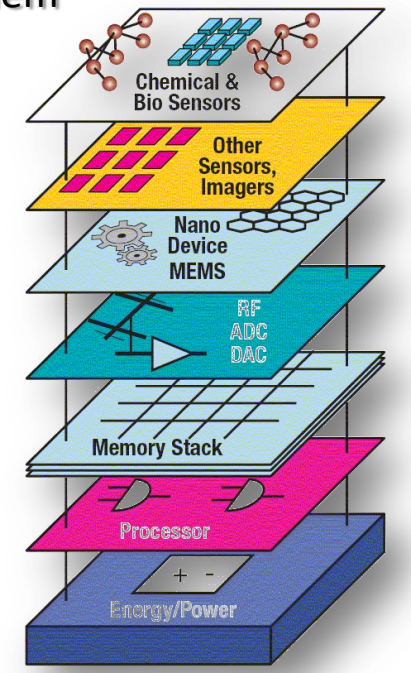
- GALS/DVFS and Asynchronous NoCs
- ASPIN an example of Async NoCs
- Async-Sync Interface
- Synchronous vs. Asynchronous
- 3D-Integration and Asynchronous NoCs
- Vertical Link Serialization
- **Vertically-Partially-Connected 3D-NoC**
- Conclusion

Vertically-Partially-Connected 3D-NoC !

- Limited number of vertical connections (TSVs)
- Network with different dies fabricated with different technoilogies
  - Heterogeneity
  - Irregularity
- Vertically-Partially-Connected Topology as an efficient solution
  - Routing strategy in such an irregular topology as the major problem

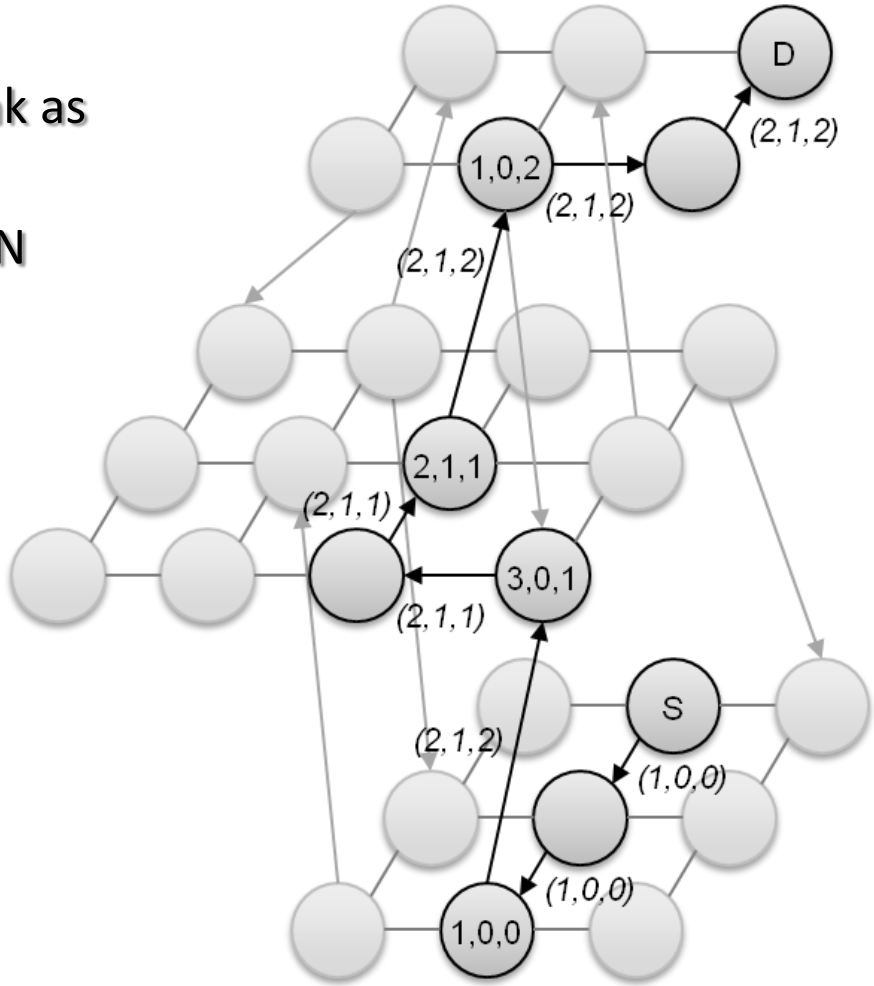






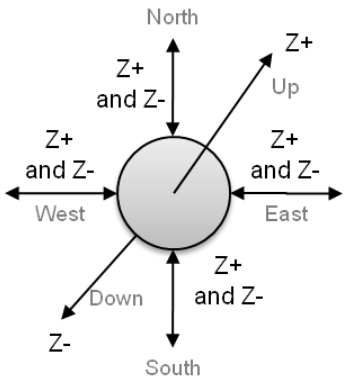
Elevator-First Routing Algorithm

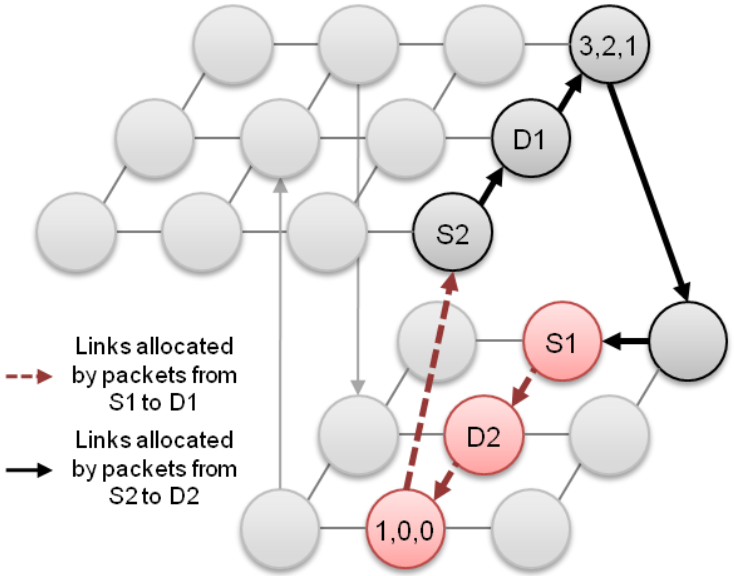
- Each router registers
  - A router in its layer with UP link as ascending elevator
  - A router in its layer with DOWN link as descending elevator



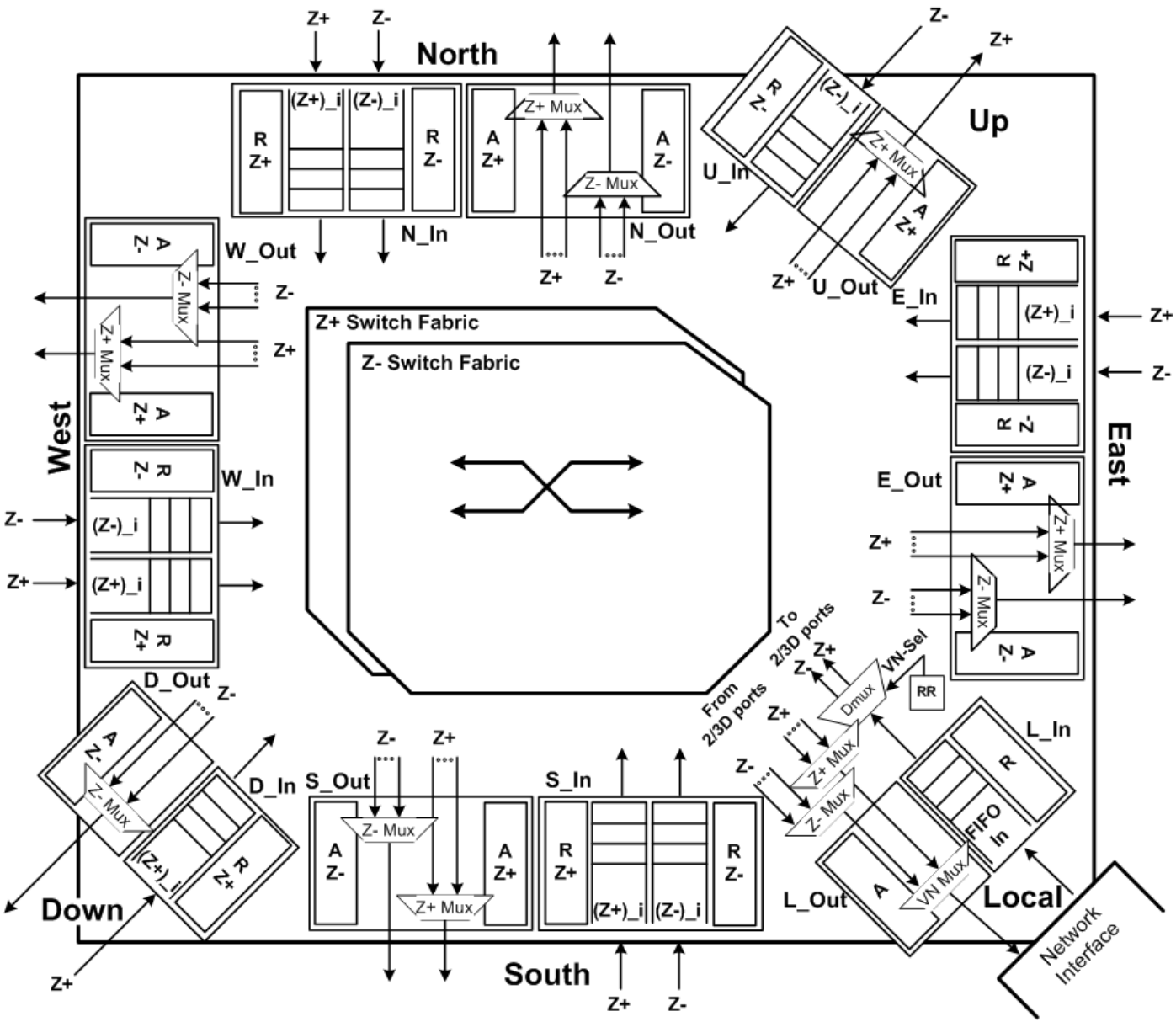
… and … Deadlock !

- Two Virtual Networks to avoid Deadlock
  - One for ascending packets (Z+)
  - One for descending packets (Z-)





Elevator-First Router



The Algorithm

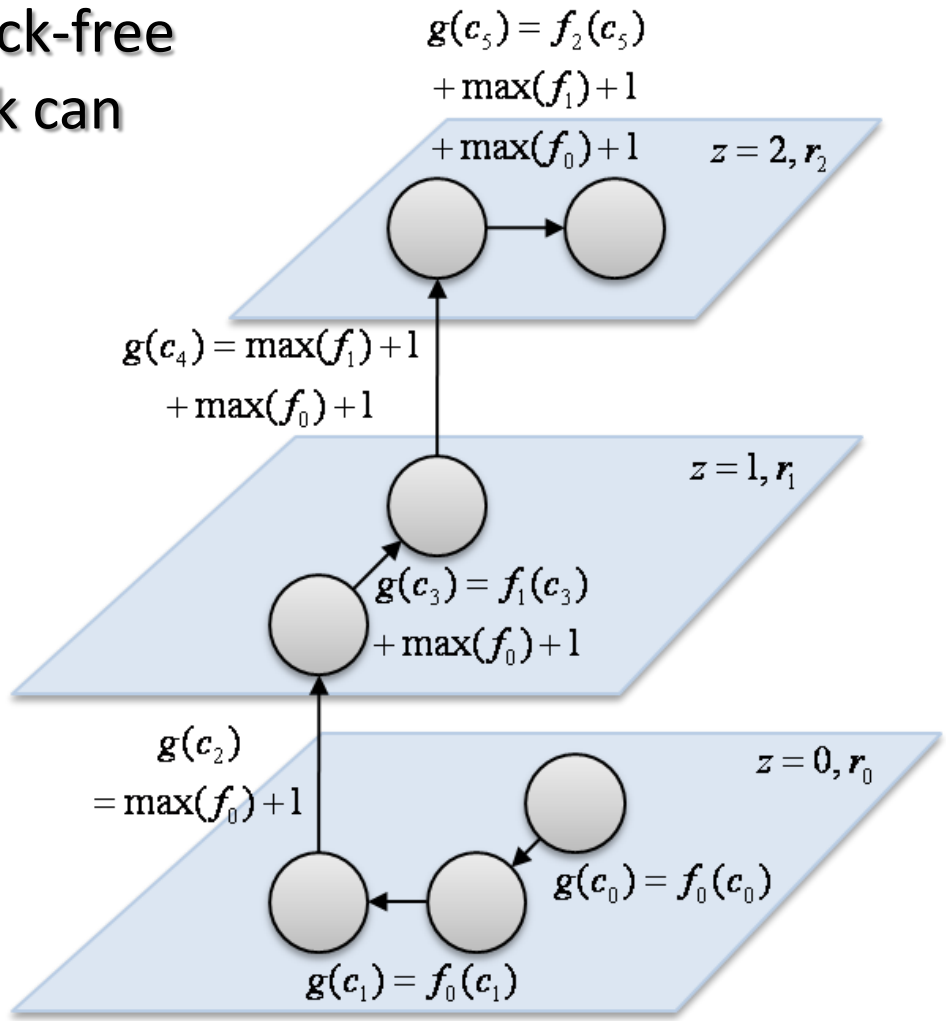
Algorithm 1 - Elevator-First Routing using two Virtual Channels

```
@c : current router address
@s : source router address
@d : destination router address
if ( @s == @c) then
  // The current router is the source
  if ( the destination is on a lower tier ) then
    - Assign the packet to the virtual network (channel) Z-
  else if ( the destination is on an upper tier ) then
    - Assign the packet to the virtual network (channel) Z+
  else
    // The destination is on the current tier
    - Randomly assign the packet to either the virtual network
      (channel) Z- or Z+
  end if
end if
```

```
if ( @d == @c) then
  if ( the elevator flag is set ) then
    // The current router is an elevator node
    - Remove the packet header
    - Get the original header (the next flit)
    - Send the packet to the ascending (if the assigned virtual channel
      is Z+) or descending (if the assigned virtual channel is Z-) vertical
      link
  else
    // The current router is the final destination
    - Consume the packet
  end if
else
  if ( The packet destination is in the current tier ) then
    - Send the packet to the port determined by the given planar
      routing algorithm (e.g. X-First in 2D-meshes)
  else
    // The packet destination is not in the current tier
    - Add a new header with the elevator flag set and an address of a
      vertical link (i.e. elevator) on the current tier (given from local
      registers) as an intermediate destination
    - Send the packet to the port determined by the given planar
      routing algorithm (e.g. X-First in 2D-meshes) toward the
      intermediate destination (i.e. the elevator)
  end if
end if
```

Formal Proof of Deadlock-Freedom

- A routing algorithm is deadlock-free if the channels in the network can be numbered such as every routing path uses strictly increasing (decreasing) channels



… an example …

b = max (n, m)

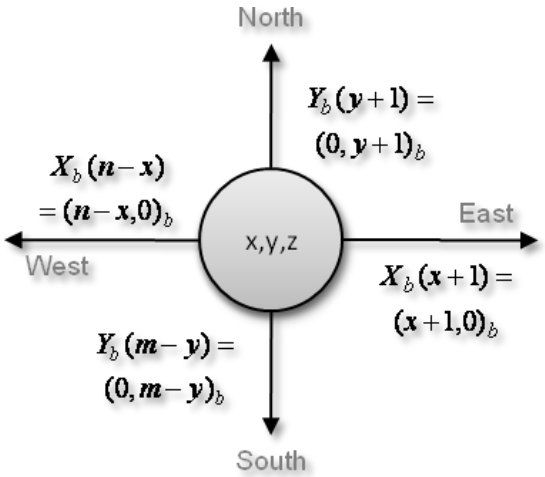
$$X_b : x \rightarrow (x,0)_b = x$$

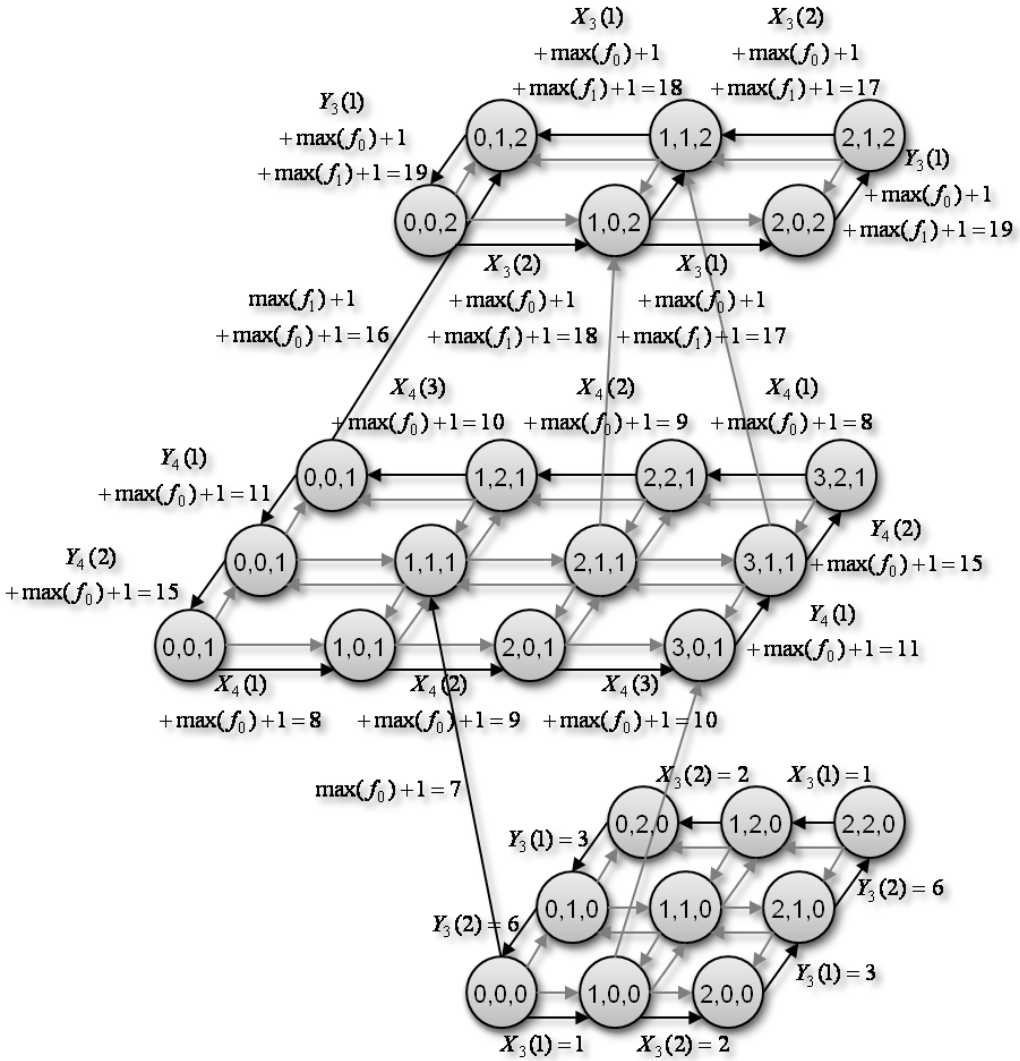
$$Y_b : y \rightarrow (0,y)_b = yb$$

$$f_z(c) \rightarrow \begin{cases} X_b(n-x) & if\ c = West \\ Y_b(y+1) & if\ c = North \\ X_b(x+1) & if\ c = East \\ Y_b(m-y) & if\ c = South \end{cases}$$

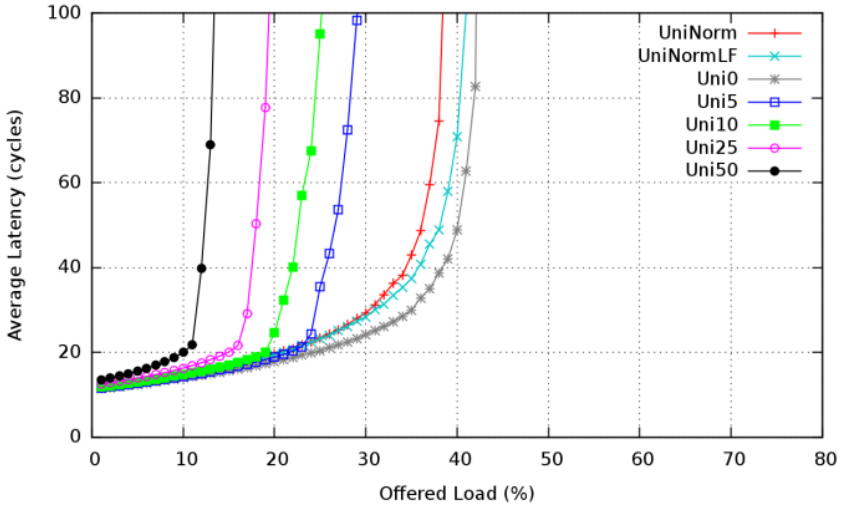
$$\forall (x,y,z),$$

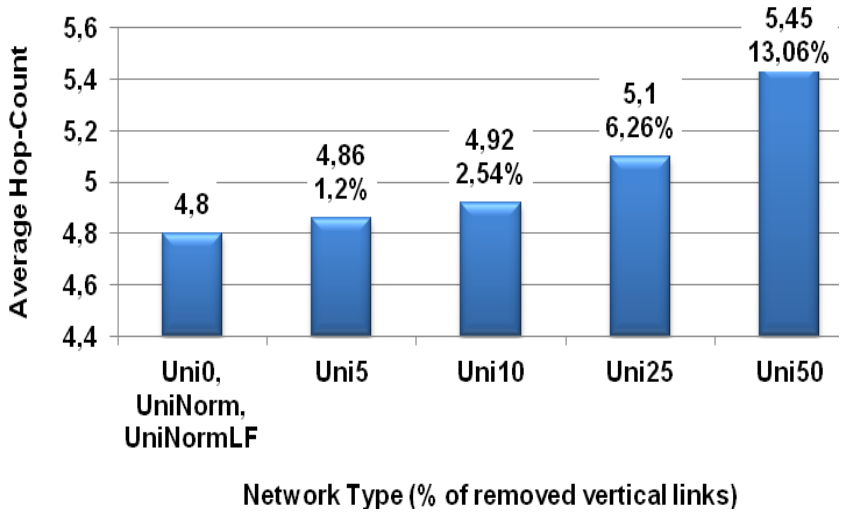
$$X_b(x) < X_b(x+1) < Y_b(y) < Y_b(y+1)$$

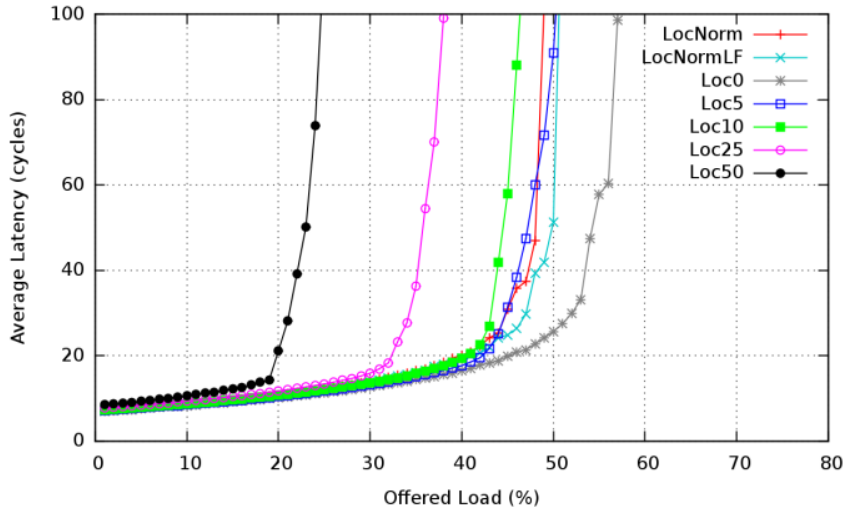


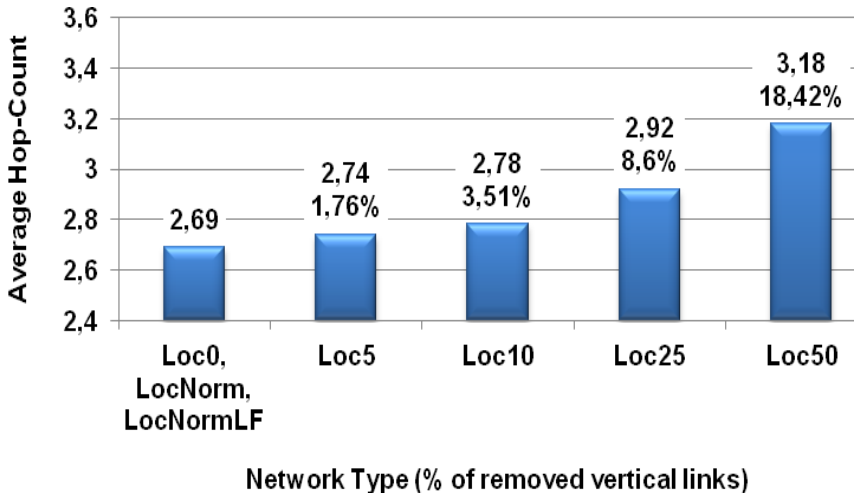


Performance …







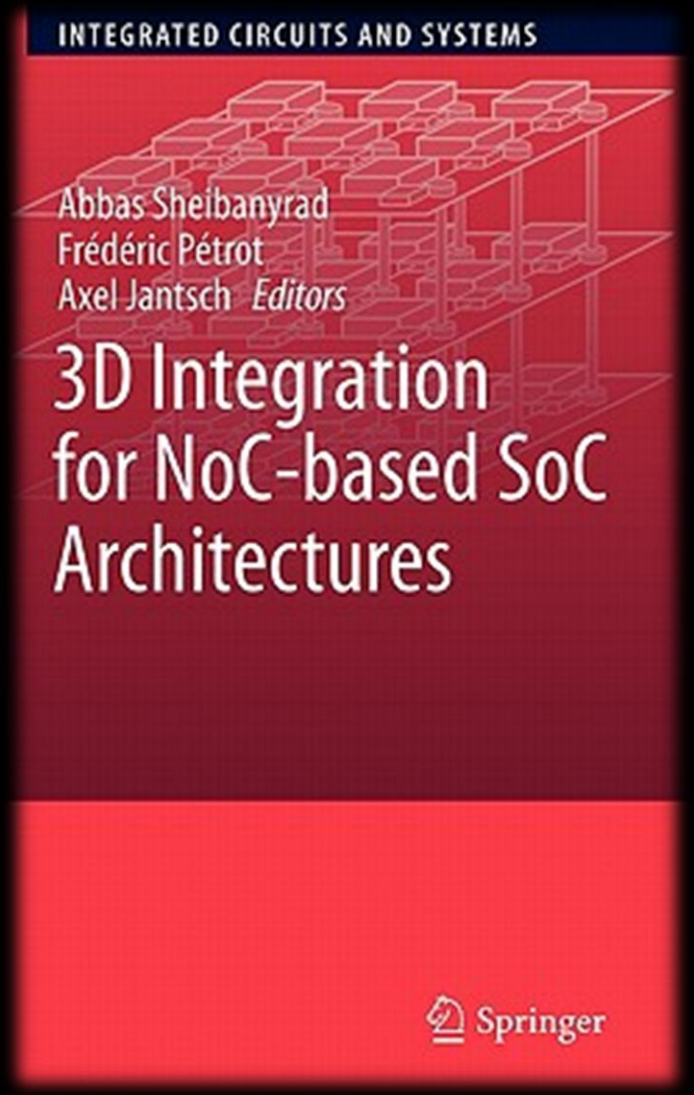


Conclusion

"Vertically-Partially-Connected Asynchronous 3D-NoC making use of Serialized Vertical Links is a viable technology and undeniably will be used as the communication infrastructure of the future Many-Core Systems"

Hamed S.

Merci…


INTEGRATED CIRCUITS AND SYSTEMS
Abbas Sheibanyrad
Frédéric Pétrot
Axel Jantsch Editors
3D Integration for NoC-based SoC Architectures
Springer