Deflection Routing in 3D Network-on-Chip with TSV Serialization

Jinho Lee, Dongwoo Lee, Sunwook Kim and Kiyoung Choi

2013. 01. 23
Design Automation Lab, SNU
Deflection Routing in 3D NoC with TSV Serialization

1. 3D NoC
2. Deflection Routing
3. Together
Outline

1. Motivation
   - 3D NoC

2. Background
   - Deflection routing

3. Deflection Routing for 3D NoC
   - Input/output Imbalance Problem
   - 3D Deflection Overhead Problem
   - Deadlock Problem

4. Experimental Result

5. Conclusion
Motivation, 3D NoC
• Example 4x4 Mesh
Introduction

• 3D integration
  – Stacking multiple dies over others
  – New trend for high performance and low energy
3D Mesh

- Apply 3D technology to NoC...
  - Looks like this

TSV
(Through-silicon-via)
Assumption

- Inter-layer (TSV) links are narrower than intra-layer (wires) in their width.

- Because TSVs are limited in its number
  - Area
  - Reliability, expensive process, etc..

- A flit needs 4 cycles to traverse thorough a TSV-link (1 cycle for a planar wire-link)
**Motivation**

- If XYZ routing is used, congestions on TSV links will last long.

- Taking non-minimal path may be beneficial
Solutions

- Adaptive routing will solve the problem
  - Hard to design
  - Large area
  - Needs extra VCs

- Deflection Routing can be a cheaper solution
Background, Deflection Routing
Deflection Routing

- Typical router architecture
Deflection Routing

- More buffers
  - Pros: high performance
  - Cons: large area, power consumption
- Bufferless deflection routing tries to get rid of buffers with minimal performance loss
  - Also known as 'hot potato routing’
**Deflection Router (CHIPPER\(^1\))**

- Flits destined to local node is ejected in stage 1
- A new flit is injected only when there is a free slot (either by ejection or no input)
- Permute stage replaces 4x4 crossbar
  - Cheaper, but partial permutability only

---

Summary of deflection routing

• Eliminates input buffer of traditional router

• Advantage
  – Power reduction (50%)
  – Area reduction (40%)

• Disadvantage
  – Reduced bandwidth
  – Increased power consumption on high load
  – Additional information on each flit

• Deflection routing is naturally adaptive in a very cheap way
Deflection Routing for 3D NoC
Basic Idea

• When deflection routing is applied to target 3D NoC, it is expected to balance the utilization of inter-layer traffic
Problem 1 (Excess Input)

• Problem 1: There may not be enough outputs
  – Deflection routing works because there are same number of inputs and outputs
  – However, a TSV link may not be done transmitting because it takes 4 cycles
Problem 2 (3D Deflection)

- Problem 2: Deflection through TSV link is not desirable
  - When a flit is deflected to a TSV link, it takes 4 cycles.
  - During 4 cycle, the TSV link cannot be used by any other flit.
  - The deflected flit eventually has to come back (4 more cycles).
  - TSV consumes more energy.
Solution of Problem 1, 2

- Treating TSV ports like ports to end-node (inject / eject) can solve the problems
  - Flits directed to TSV ports are taken in first stage
  - Flits coming from TSV ports are injected only when there is a free slot
Deadlock Problem

- Using TSV ejection scheme, a deadlock can occur

Layer 1

(Full of flits destined to layer 0)

Cannot be injected because there is never a free slot

Cannot be transferred to layer 0
Because TSV link is never free

Layer 0

(Full of flits destined to layer 1)

Flits destined to a node in layer 1
Flits destined to a node in layer 0
Deadlock avoidance

1. When Rx buffer is filled, Tx buffer should not accept a new flit. (a TSV link can hold at most one flit at a time)
Solution of Deadlock Problem

• Escaping from deadlock
Deadlock avoidance

2. Entrance to TSV Tx buffer is allowed on following condition even though its Rx buffer is filled
   – There is no free slot in the 2D input ports.
   – The TSV Rx buffer on same layer is ready to inject (full).
   – The TSV Tx buffer is empty (receive buffer on the other side may be full).
Solution of Deadlock Problem

• Escaping from deadlock

• What if it was not a dead-lock?
  – Does not cause a functional problem
Experimental Result
Experimental Result

- 4x4x4 configuration
- Compared with 4 other routers
  - Simple buffered – 8 buffers
  - VC buffered – 4 VC, 8 buffers per VC
  - Naïve deflection – allowing inter-layer deflection
  - adaptiveXYZ – minimal adaptive routing for 3D NoC
- Four traffic patterns
  - Uniform random, Hotspot random, Tornado, Bit-complementary
Experimental Result - latency

- Metric “saturation load” – input load when latency exceeds 500 cycles
  - 25.3 % better than simple buffered
  - 9.2% better than adaptive XYZ
**Experimental Result - thruput**

- **Saturated Throughput**
  - 1.9% better than simple buffered
  - 1.2% worse than adaptiveXYZ
  - Reaches maximum point quickly
Experimental Result – power efficiency

- Energy per packet
  - Minimum: 33.3% lower than simple buffered
  - 72.5% lower than adaptiveXYZ
  - Saturated: 13.3% higher than simple buffered
  - 54.3% lower than adaptiveXYZ
Conclusion

• Use of bufferless deflection routing is suggested on 3D NoC with TSV serialization.

• Some problems are solved.
  – Excess Input
  – 3D deflection
  – Deadlock & livelock

• Higher performance in terms of throughput and power efficiency is obtained.
- Thank you for your attention
- Feel free to ask any questions
Introducing Sidebuffer (MinBD$^2$) (Opt)

- Side buffer stores at most one deflected flit per cycle
  - Reduces deflection rate

Livelock Avoidance (opt)

• Livelock is avoided using sidebuffer
  – “golden packet” always wins.
  – If golden packet’s way is TSV and it is blocked, put it into sidebuffer instead.
  – Duration for flit to stay in sidebuffer is limited and advance of golden packet is guaranteed.