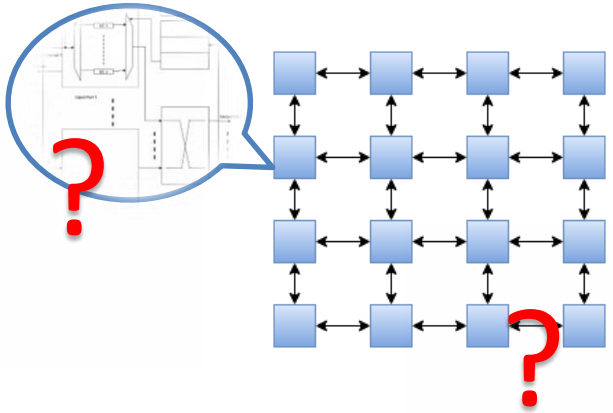
Exploring Fault-Tolerant Network-on-Chip Architectures

D. Park, C. Nicopoulos, J. Kim, N. Vijaykrishnan, Chita R. Das

Tan Bie **Yang Jiao**

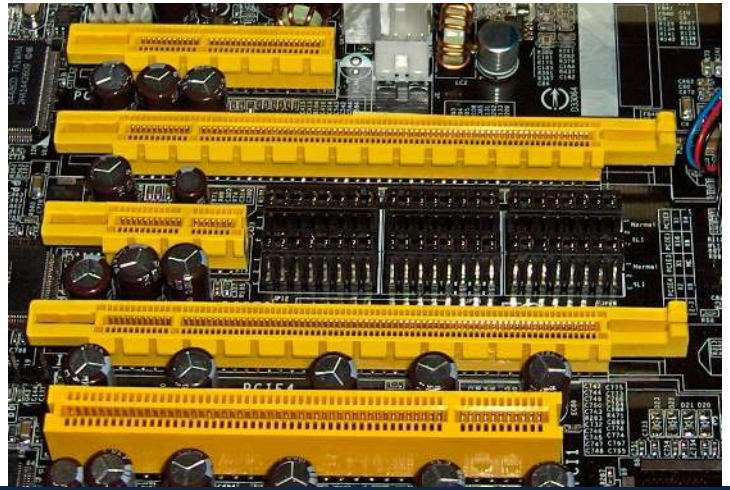
Outline



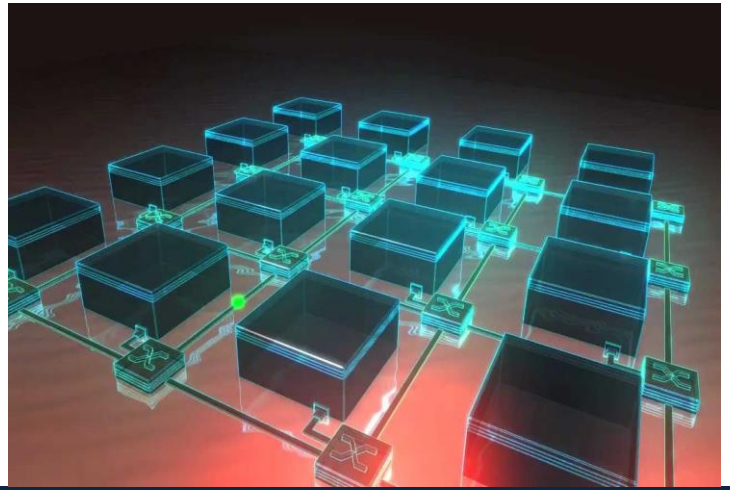
- A brief review of NoC (Network-on-Chip)
- Why the soft errors of NoC matters?
- Two major failure sources of on-chip interconnection
  - NoC Link Soft errors & Handling
  - Soft errors in Intra-Router Logic & Handling
- Evaluation of proposed schemes
- Retransmission Buffer reuse for deadlock recovery

Review: On-chip interconnections

- **Bus**
- Low complexity (+)
- Limited bandwidth/scalability(-)

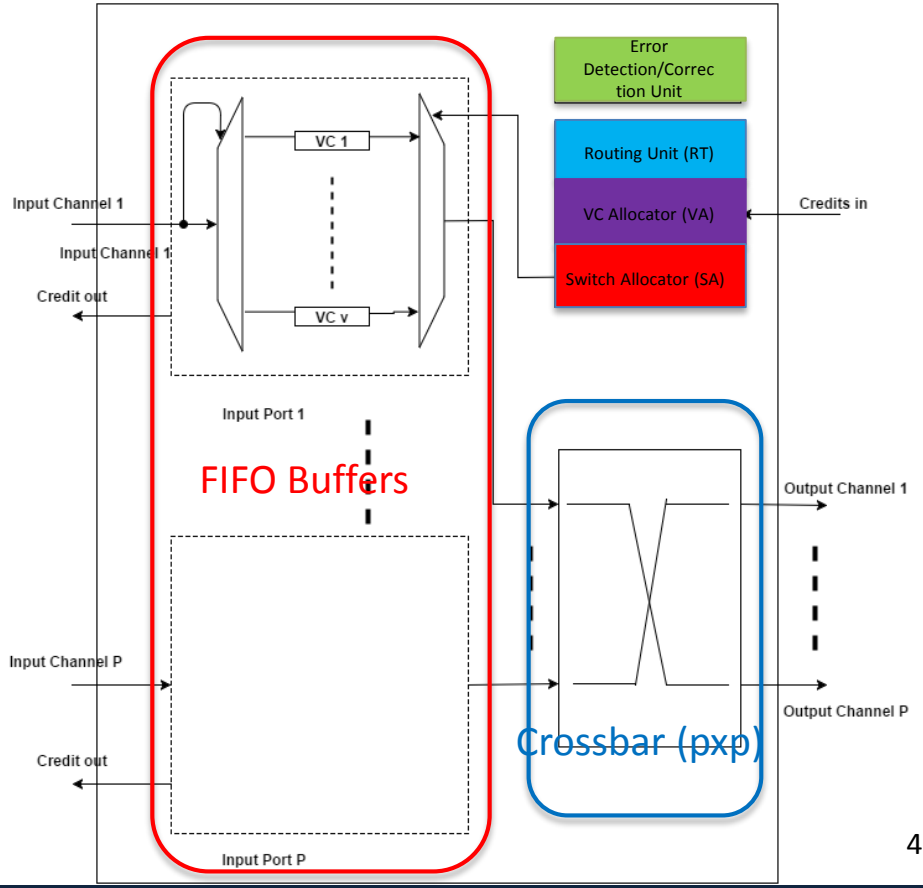


- **NoC**
- Higher bandwidth/scalability (+)
- Reliability issue (-)
- Stringent resource constraints (-)



Review: NoC router architecture

- FIFO Buffers
- Error Detection/Correction Unit
- Routing Unit (RT)
- Virtual Channel Allocator (VA)
- Switch Allocator (SA)
- Crossbar



Soft errors in NoC

Two major sources of soft errors in NoC:
Link & Intra-router errors

feature sizes ↓ + operating frequency ↑

SER(soft error rate) per chip in Link & routers ↑

↓

Reliability issues regarding NoC becomes a **concern** for us

↓

A Comprehensive set of techniques to protect against link errors and single-event upsets within the router is needed!

Handling link soft faults

Two Central Themes:

•Error correction: **Forward Error Correlation (FEC)**

•Retransmission:

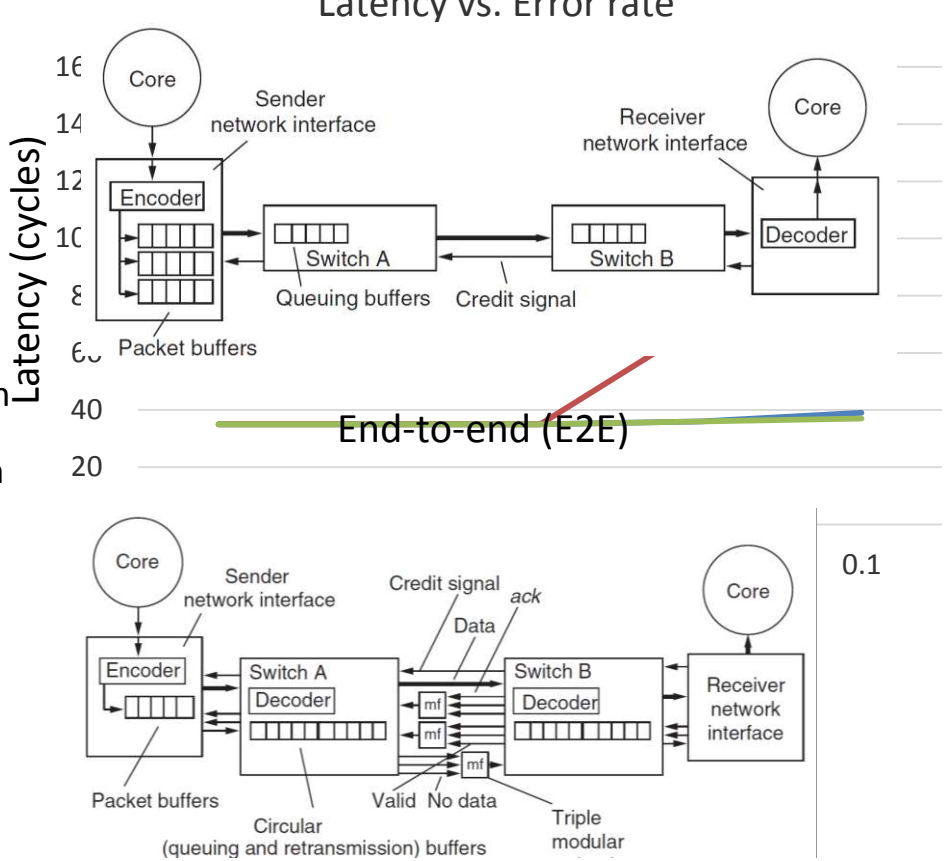
-**End-to-End (E2E)**: data checked only at the destination

-**Hop-by-Hop (HBH)**: data checked along the path from source to destination

Our choice:

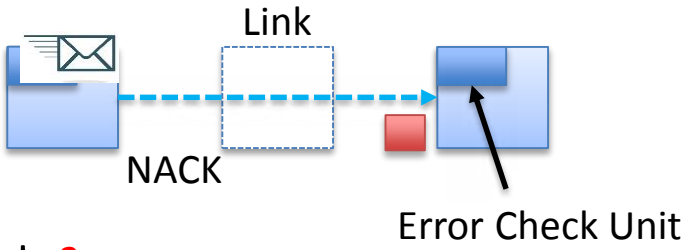
Hybrid techniques!

Combination of **HBH** and **FEC**

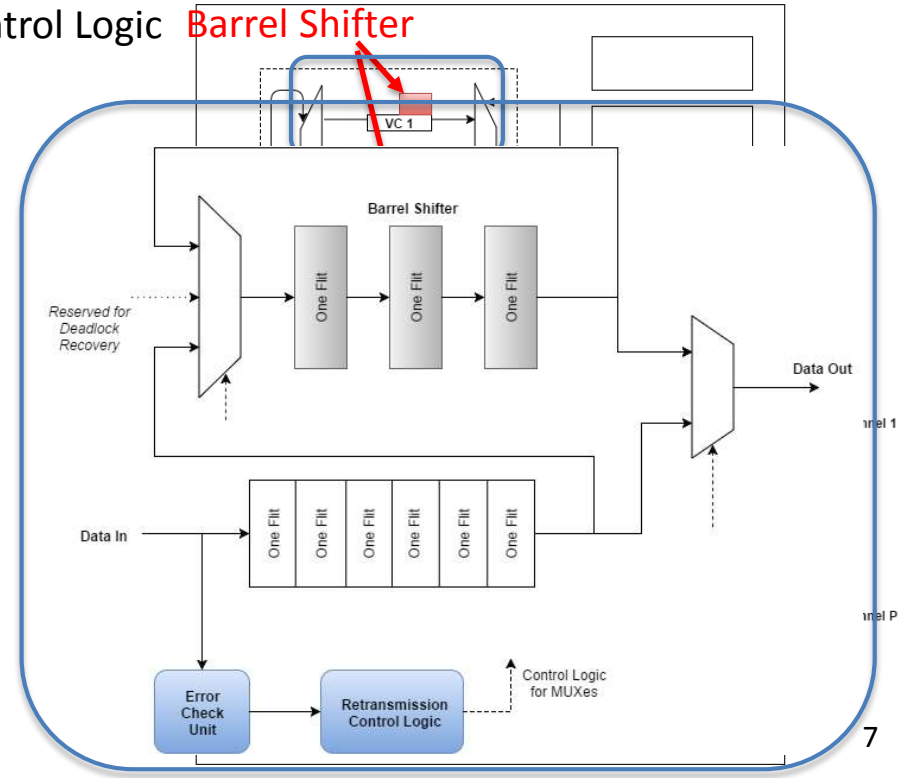


Flit-based HBH (Hop-by-Hop) retransmission scheme

- Error Check Unit & Retransmission Control Logic
- Barrel Shifter:
  - 3-flit-deep retransmission buffer
  - FIFO (First-in, First-out)

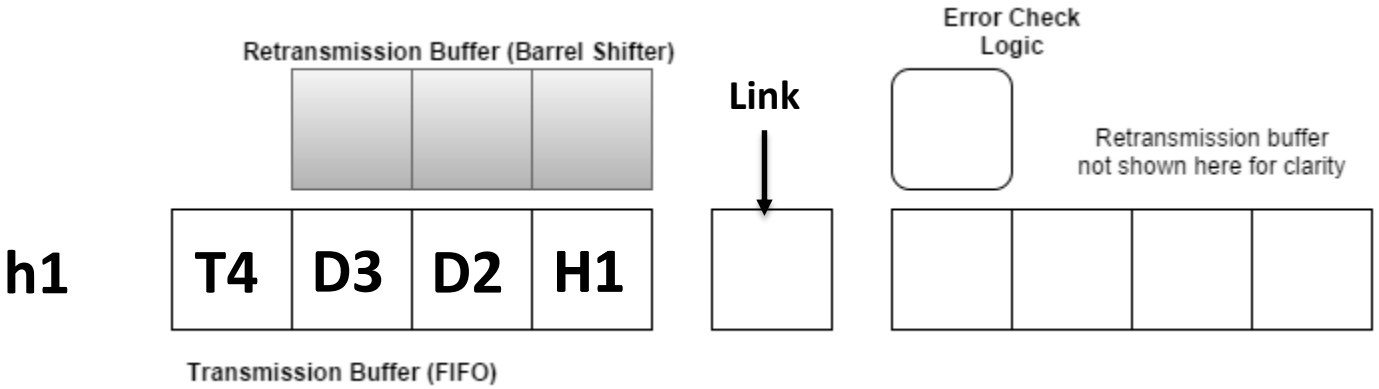


Cycle 3



NACK – Negative Acknowledge

An example of HBH (Hop-by-Hop) retransmission scheme

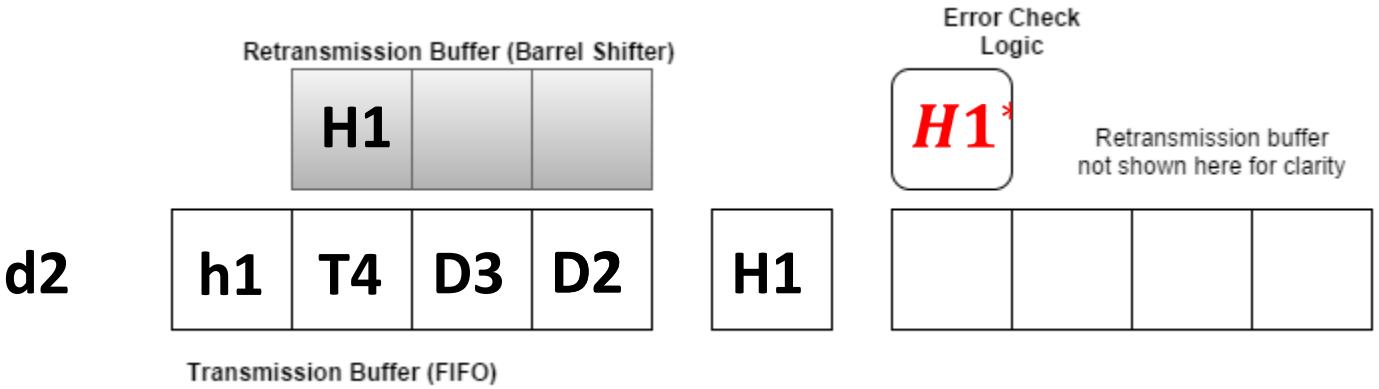


Clock 0

H1 - Header flit
D2 – Data flit
D3 – Data flit
T4 – Tail flit

h1 - Header flit of next packet
d2 – Data flit of next packet
d3 – Data flit of next packet
t4 – Tail flit of next packet

An example of HBH (Hop-by-Hop) retransmission scheme

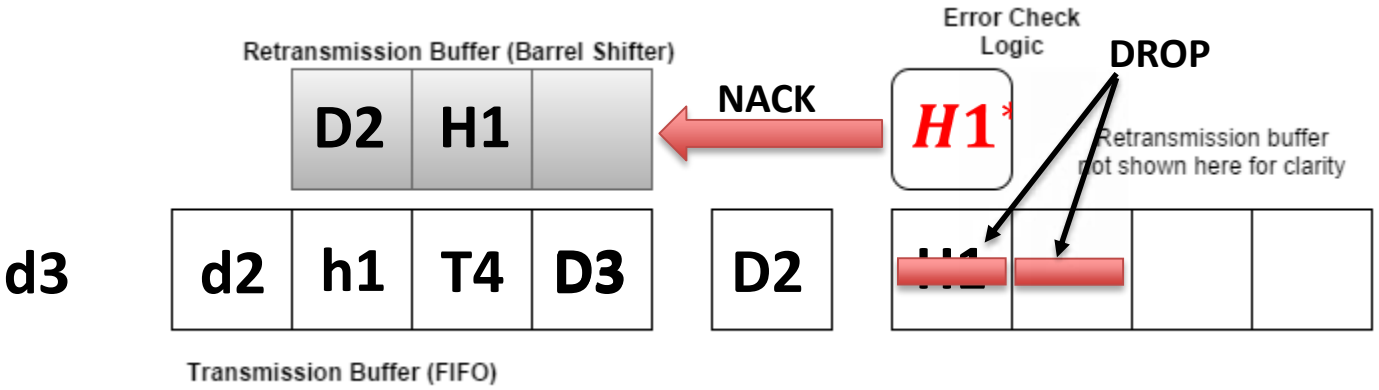


Clock 12

H1 - Header flit
D2 – Data flit
D3 – Data flit
T4 – Tail flit

h1 - Header flit of next packet
d2 – Data flit of next packet
d3 – Data flit of next packet
t4 – Tail flit of next packet

An example of HBH (Hop-by-Hop) retransmission scheme



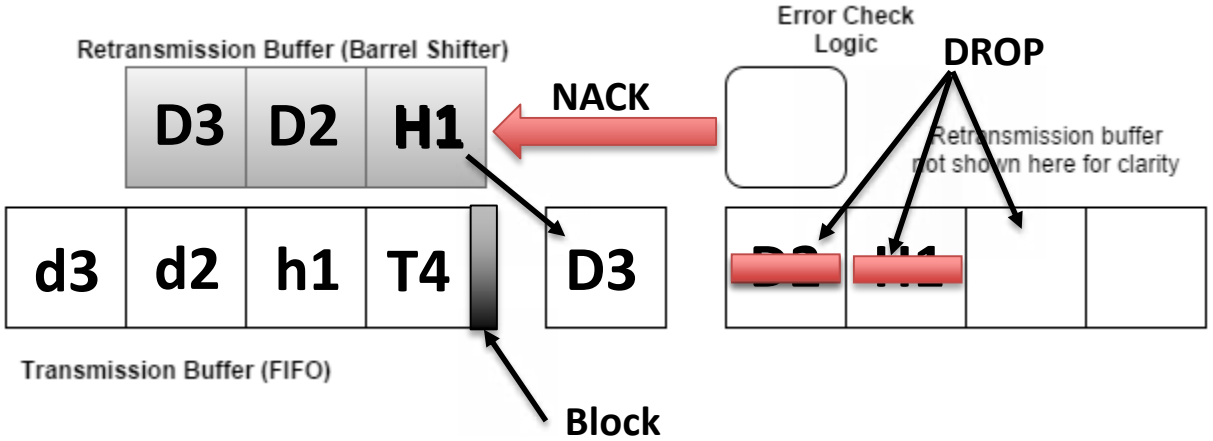
Clock 3

H1 - Header flit
D2 – Data flit
D3 – Data flit
T4 – Tail flit

h1 - Header flit of next packet
d2 – Data flit of next packet
d3 – Data flit of next packet
t4 – Tail flit of next packet

NACK – Negative Acknowledge

An example of HBH (Hop-by-Hop) retransmission scheme



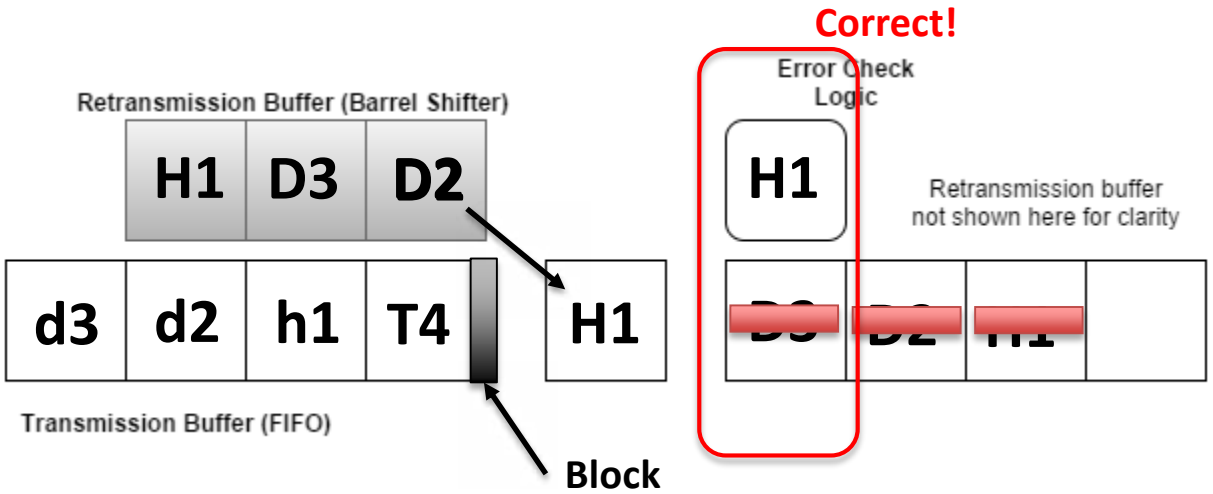
Clock 4

H1 - Header flit
D2 – Data flit
D3 – Data flit
T4 – Tail flit

h1 - Header flit of next packet
d2 – Data flit of next packet
d3 – Data flit of next packet
t4 – Tail flit of next packet

NACK – Negative Acknowledge

An example of HBH (Hop-by-Hop) retransmission scheme

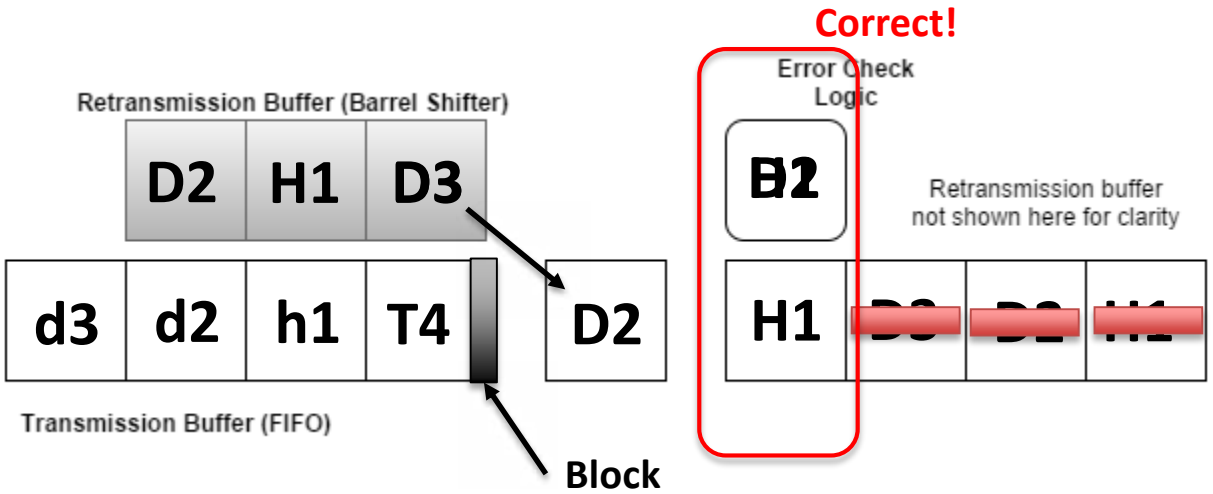


Clock 4

H1 - Header flit
D2 – Data flit
D3 – Data flit
T4 – Tail flit

h1 - Header flit of next packet
d2 – Data flit of next packet
d3 – Data flit of next packet
t4 – Tail flit of next packet

An example of HBH (Hop-by-Hop) retransmission scheme

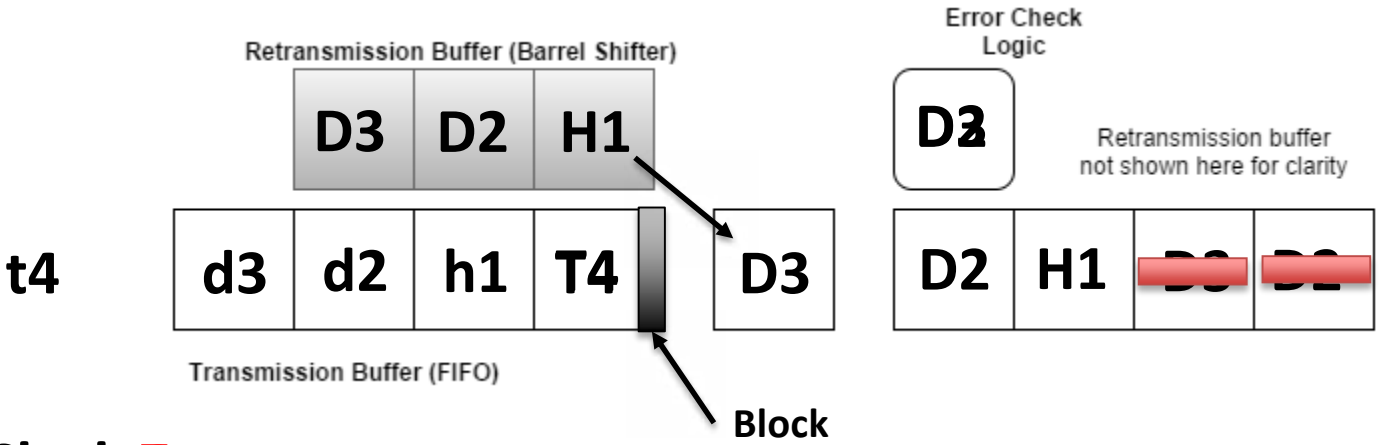


Clock 6

H1 - Header flit
D2 – Data flit
D3 – Data flit
T4 – Tail flit

h1 - Header flit of next packet
d2 – Data flit of next packet
d3 – Data flit of next packet
t4 – Tail flit of next packet

An example of HBH (Hop-by-Hop) retransmission scheme



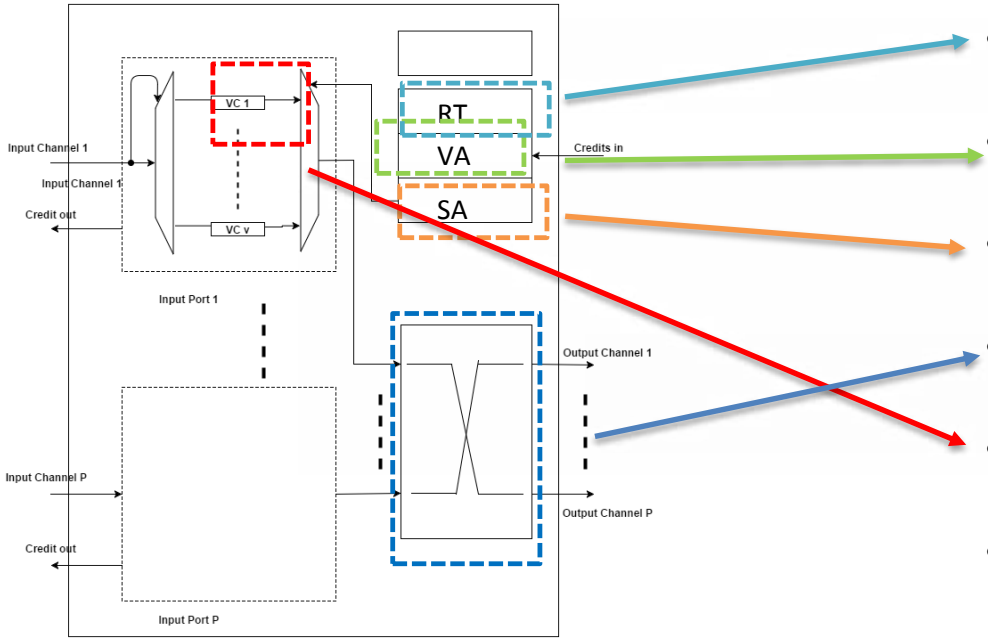
Clock 6

H1 - Header flit
D2 – Data flit
D3 – Data flit
T4 – Tail flit

h1 - Header flit of next packet
d2 – Data flit of next packet
d3 – Data flit of next packet
t4 – Tail flit of next packet

Classes of Soft errors in Intra-Router Logic

NoC Router Architecture



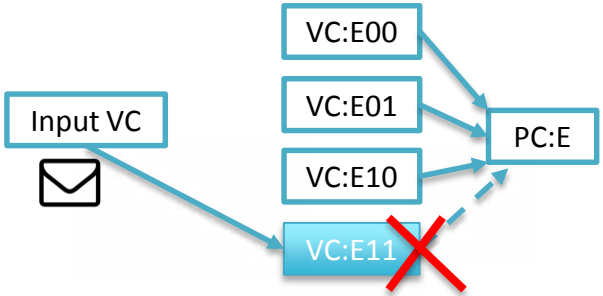
Errors Types

- Virtual Channel Allocator Error
- Routing Unit Errors
- Switch Allocator Errors
- Crossbar Errors
- Retransmission Buffer Errors
- Handshaking signals Errors

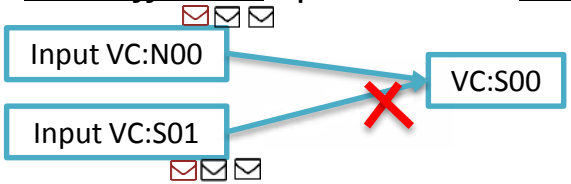
Virtual Channel Allocator Errors

VA operates on header flits, works as an arbitrator to assign the allocations of packets to the required output VCs

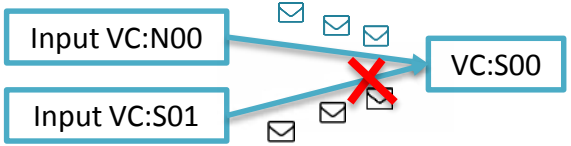
1.One input VC to an *invalid* output VC



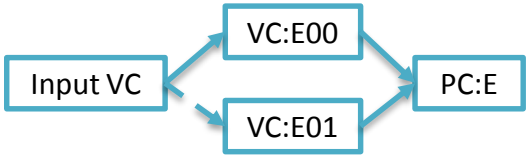
2.*Two different* input VC to an *unreserved* output VC



3.One input VC to an *reserved* output VC

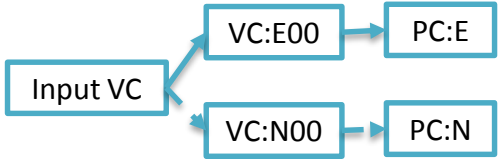


4.One input VC to an *erroneous* output VC



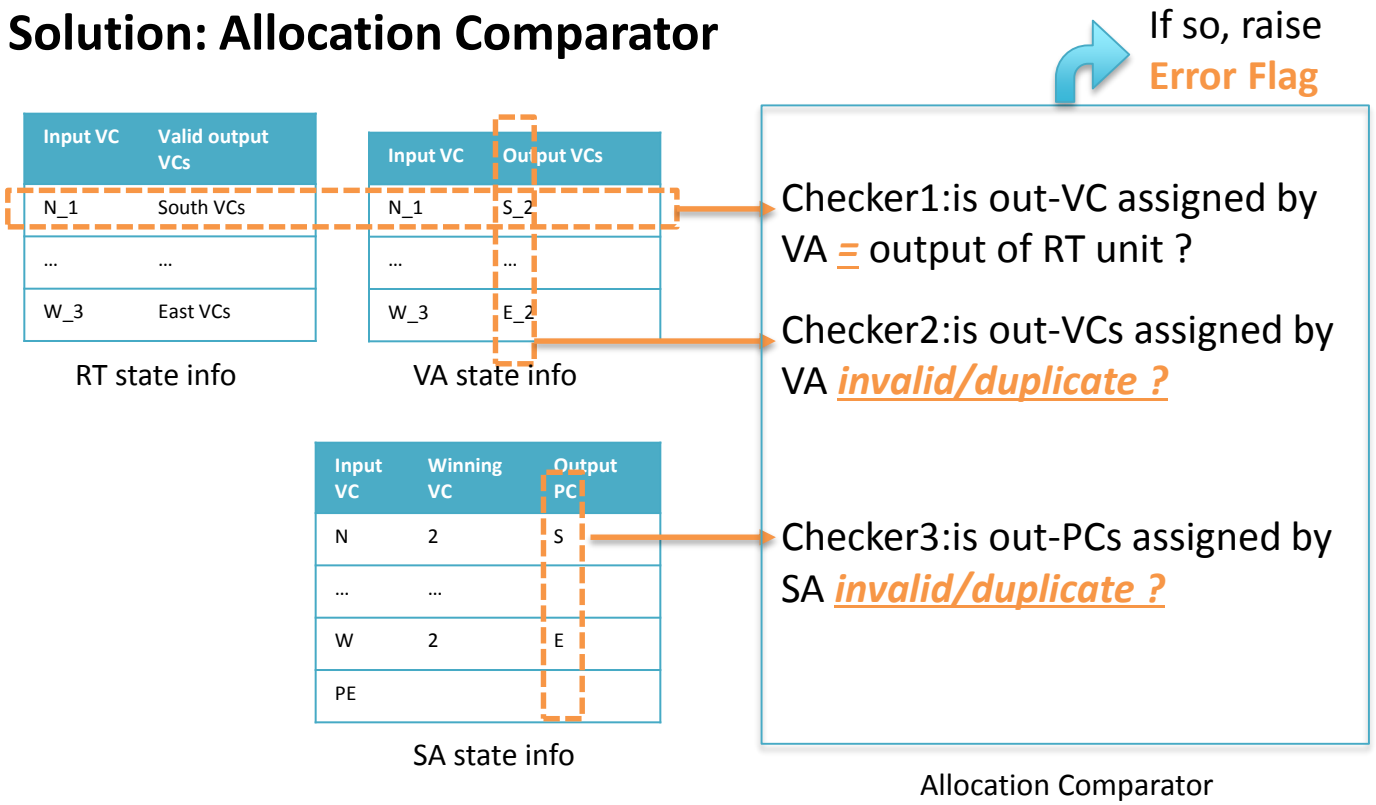
(a)

Misdirection, may lead to deadlock



(b)

Solution: Allocation Comparator

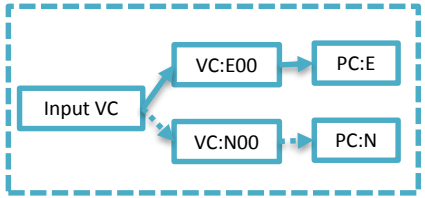


- Purely combination logic
- Consist of XOR gates
- Collect info from RT, VA, VC entries
- Three checker run in parallel

Allocation Comparator: Detect

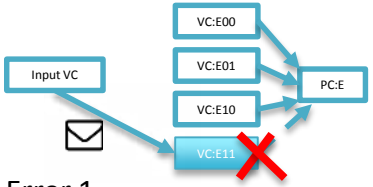
Checker1:is out-VC assigned by VA = output of RT unit ?

Error 4



- NACK Message
- Error Flag

Checker2:is out-VCs assigned by VA *invalid/duplicate ?*



Error 1

Error 2/3

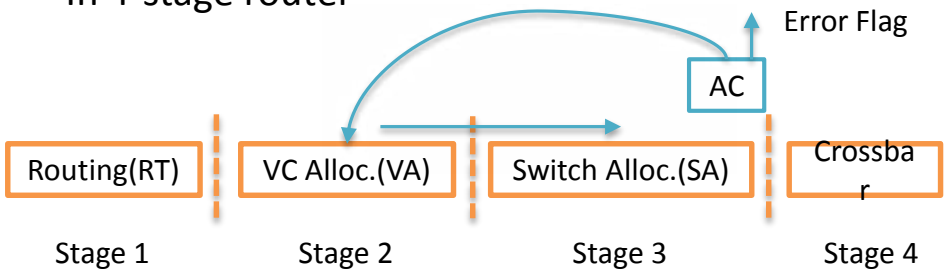
- NACK Message
- Error Flag

Checker3:is out-PCs assigned by SA *invalid/duplicate ?*

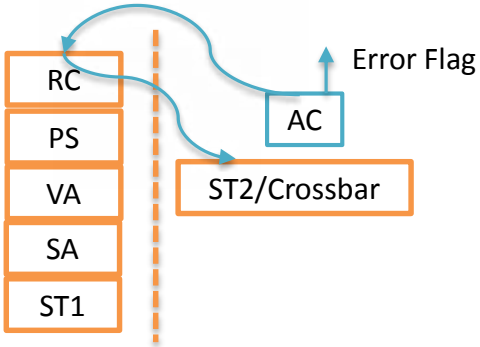
Further used in *Switch allocator Errors*

Allocation Comparator: Recovery

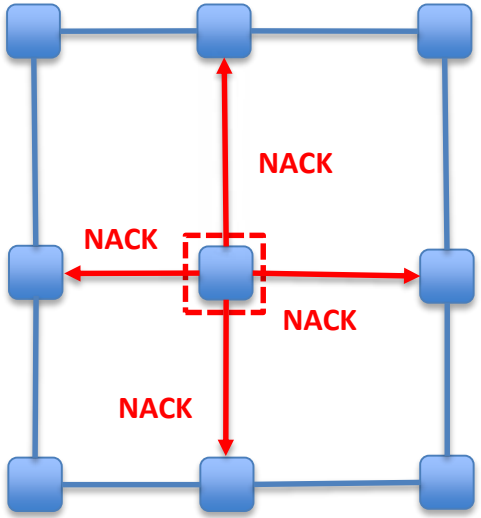
In 4-stage router



In other cases, such as 2-stage router



&



Switch Allocator Errors

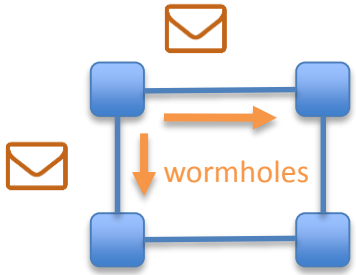
Soft errors in SA may incur:

a)Prevent flits from assessing the crossbar - Easy to deal with

b)Data flits may sent to a wrong direction, without header flits

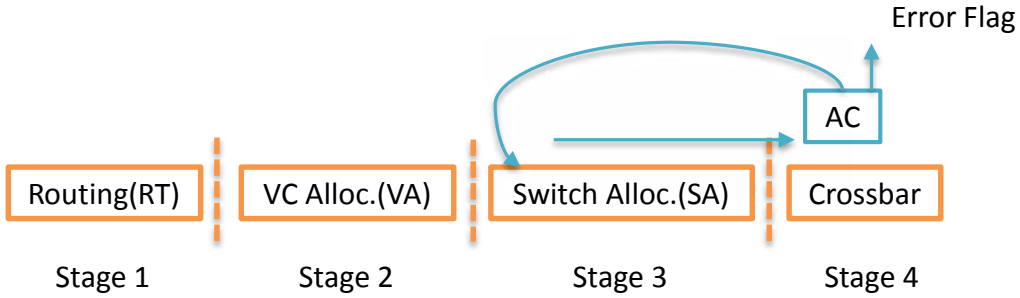
c)Two flits direct to the same output - Easy to deal with

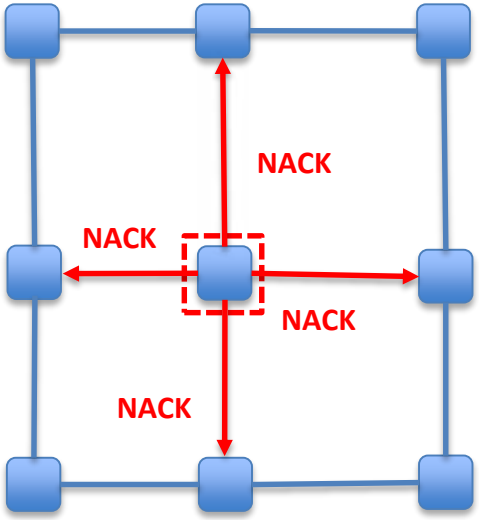
d)One flit may be mistakenly sent to multiple outputs



Switch Allocator Errors

AC: Checker3 is out-PCs assigned by SA ***invalid/duplicate ?***
to deal with b) d)





Recovery process need 1 cycle

Routing Unit Errors

Wrong

Packet misdirection

If the erroneous direction is ***blocked***

❖By using looking-ahead routing

- Do we need NACK?

  Yes

❖ By using current-node routing

- Do we need NACK?

  No

- How long is the recovery process?

  One cycle

If the erroneous direction is ***non-blocked***

❖By using deterministic routing

- like the situation left
- Recovery process takes (1+n) cycles

❖ By using adaptive routing

- Such error doesn't matter

Evaluation Platform

- A cycle-accurate network simulator
- A 64-node (8*8) MESH network
- 3-stage pipelined virtual-channel-based wormhole router
  - 5 physical channels (PCs) per router and 3 virtual channels (VCs) per PC
- Three traffic permutation: Normal random(NR), Bit-complement(BC), Tornado(TN) are used to select destination node
- Hardware Implemented in RTL and synthesized at TSMC 90nm Technology
- Supply voltage = 1v, clock speed = 500MHz

Some assumptions

- Single link traversal complete in one clock cycle
- Single event upsets, i.e., only one fault happen at any given time

Evaluation of Allocation Comparator

Power & area overhead of the Allocation Unit

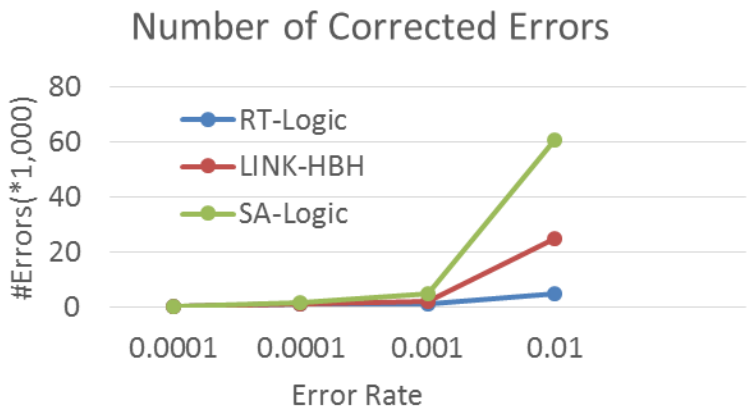
Low power/area overhead

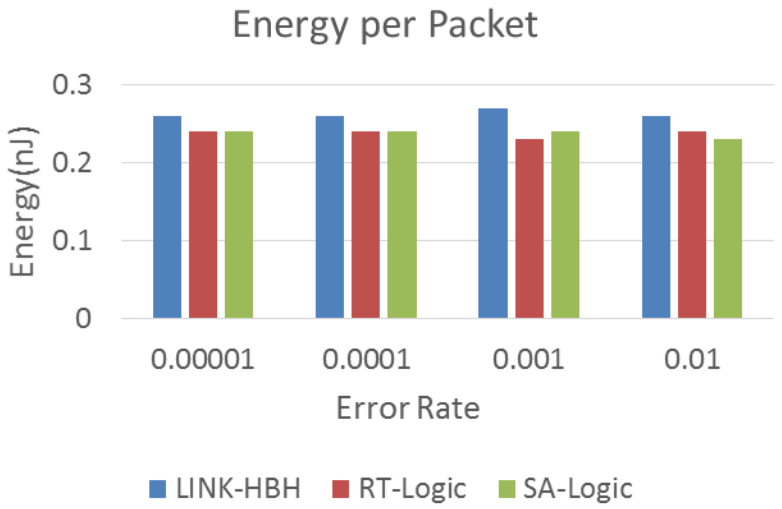
Component	Power	Area
Generic NoC Router (5 PCs, 4VCs per PC)	119.55mw	0.374862mm^2
Allocation Comparator	2.02mw (+1.67% overhead)	0.004474mm^2 (+1.19% overhead)

Evaluation of proposed soft errors handling solutions

Simulated Three types of error situations separately:

Routing logic errors(RT-Logic); Switch allocator logic errors(SA-Logic); Link errors(LINK-HBH)



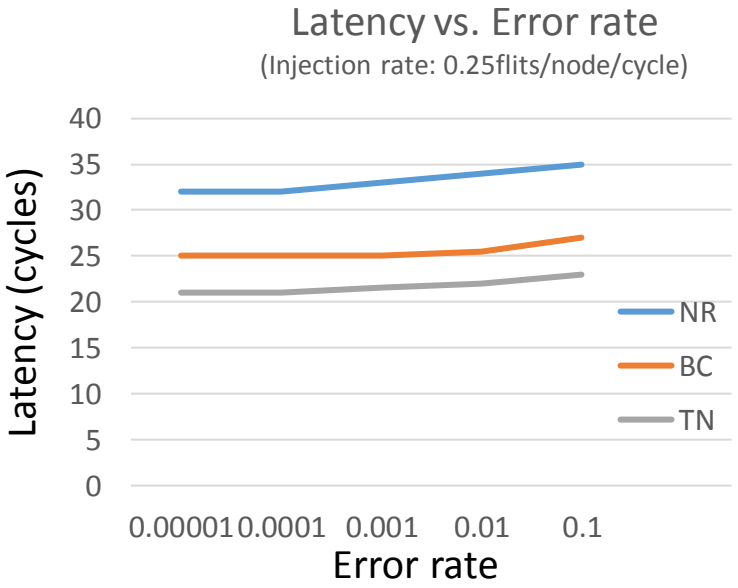


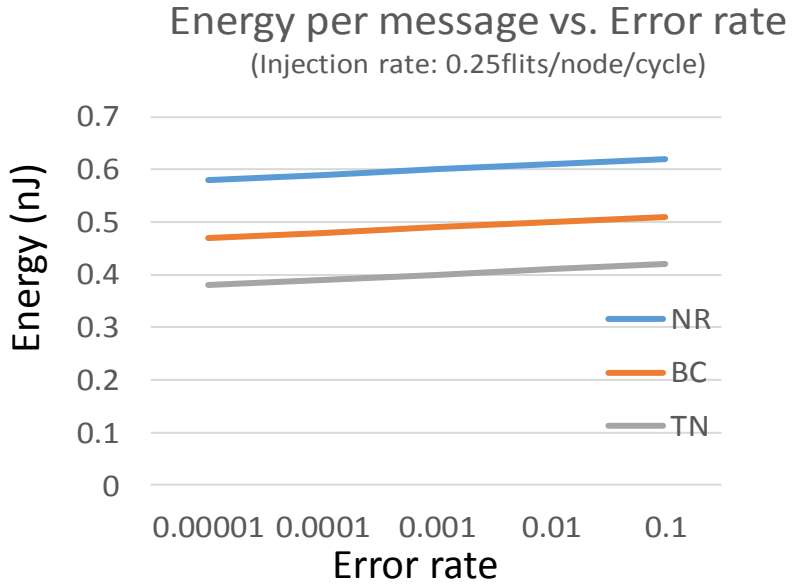
Evaluation of HBH retransmission scheme

Cycle-accurate simulation with three traffic patterns

[Number Random (NR), Bit-Compliment (BC) and Tornado(TN)]

- overall, the retransmission scheme is efficient



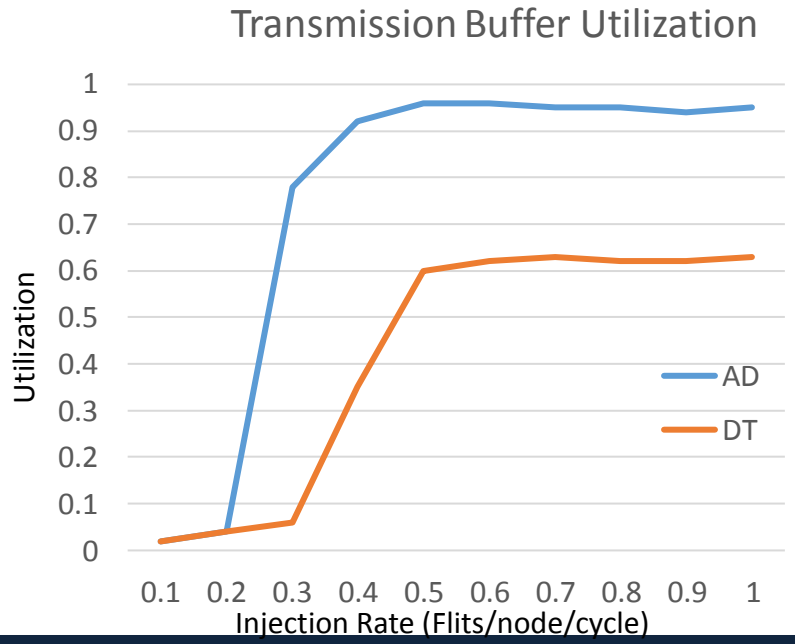


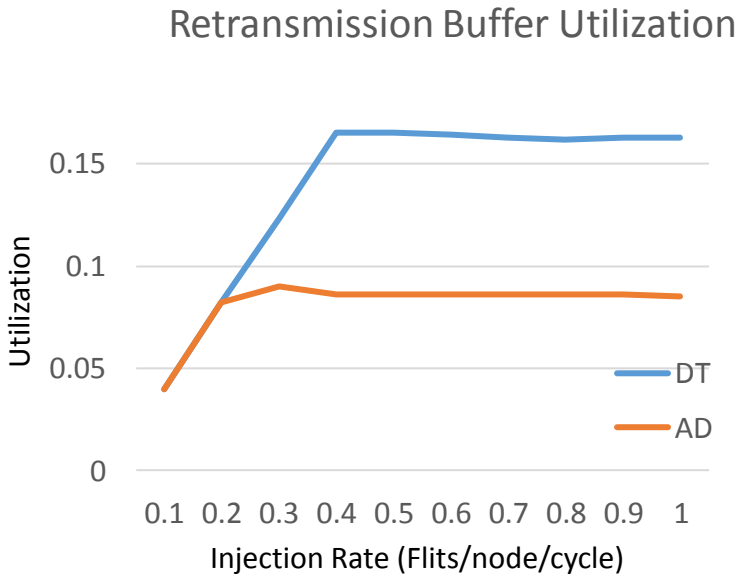
Utilization of transmission & retransmission buffers

Utilization of retransmission buffer relatively small with packet blocking increases

- **One bonus**: Exploit idle buffers for **deadlock recovery**!

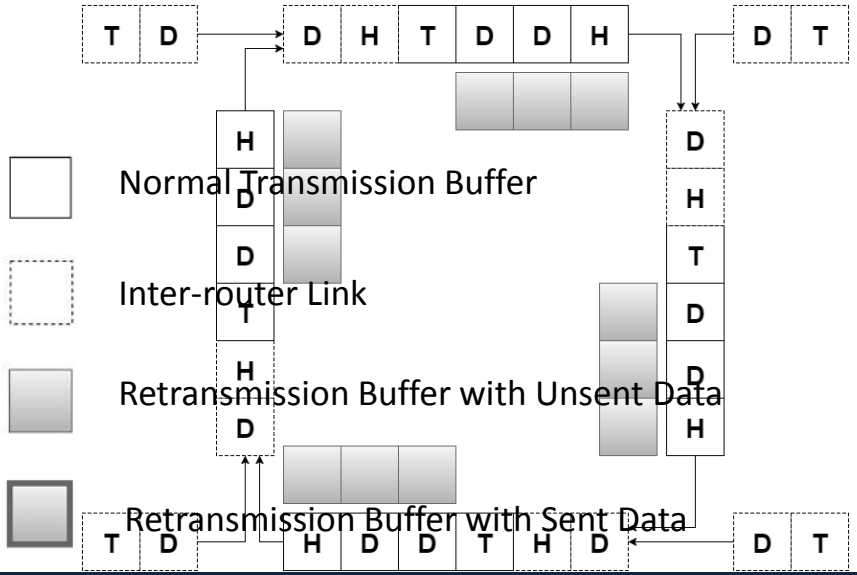
AD – Adaptive routing
DT – Deterministic routing

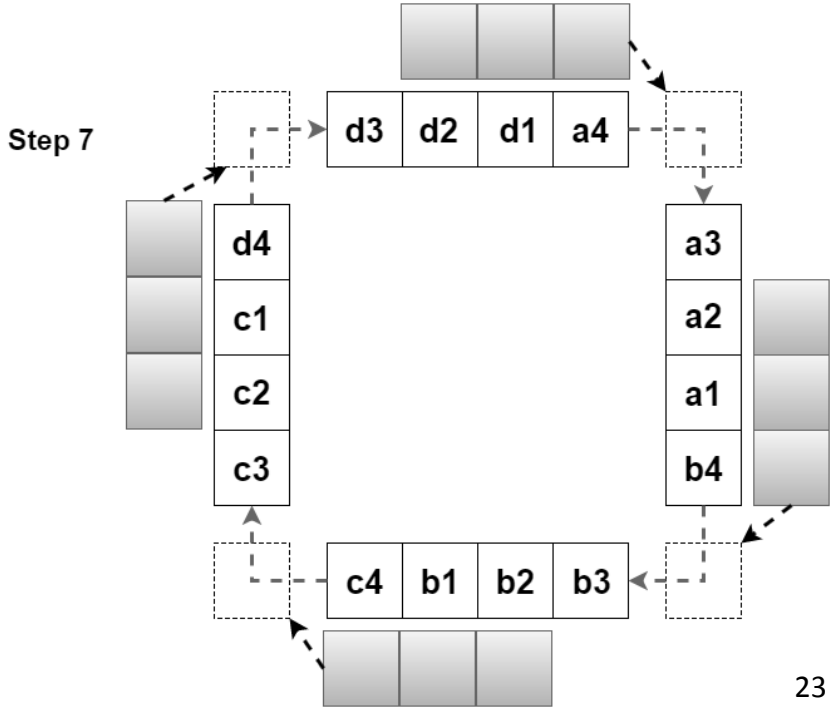




Retransmission buffer for deadlock recovery

- The procedure repeated until at least one of the packets breaks the deadlock
- Worst-case Scenario





Conclusion

- The paper presents a comprehensive plan of attack on various type of reliability hindrances in on-chip networks, including link & intra-router errors, with small power & performance overhead

- A hybrid retransmission buffer is designed into dealing with link soft faults

- Intra-router errors are effectively handled by proposed schemes

- Taking advantage of the proposed retransmission buffer structure, deadlock recovery is obtained

Questions?

Debates

1. Is it worthy to use retransmission buffer?

2. If soft error and deadlock happen at the same time, will our retransmission buffer still handle such situation?