An Area-efficient Partially Reconfigurable Crossbar Switch with Low Reconfiguration Delay

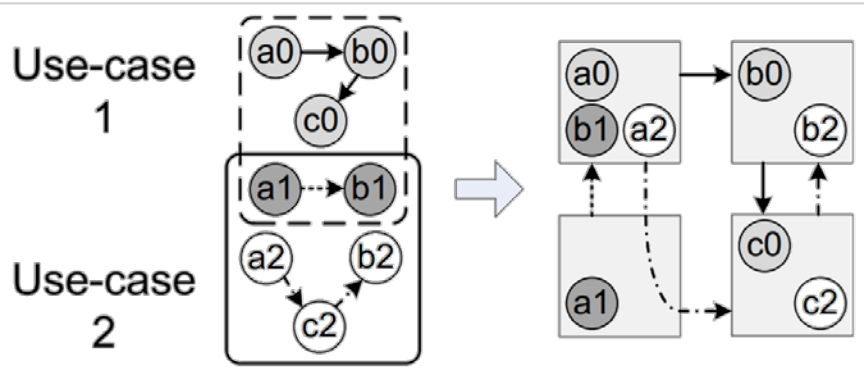
Hoo Chin Hau and Akash Kumar

Motivation

- Multiprocessors on the rise (MPSoCs)
  - Single processor systems unavailable to meet application performance requirements
  - Multiple processors + Memory + IO devices
  - Bottleneck shifts from computation to communication
- Networks-on-Chip (NoCs) proposed
  - Bus is not scalable

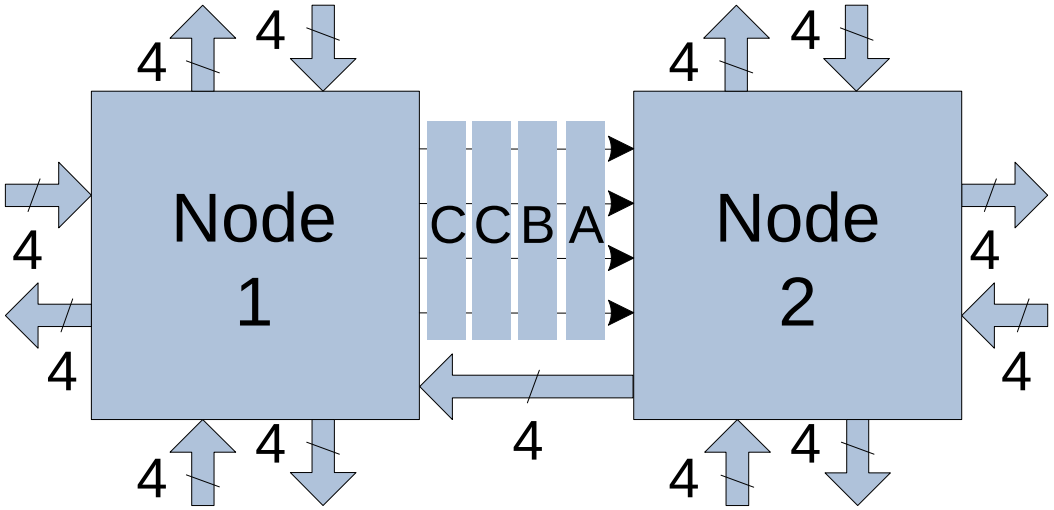
Motivation

- MPSoCs need to handle multiple use cases – combinations of multiple applications
- Use cases have different connectivity requirements
- NoCs have to be dynamically reconfigurable and provide guaranteed throughput: TDM vs SDM

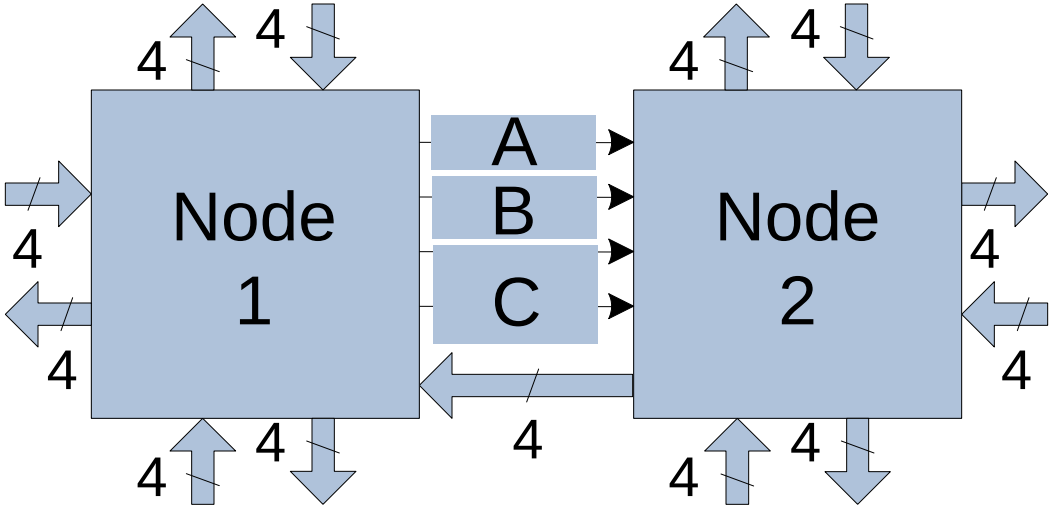


TDM vs SDM

TDM



SDM



Motivation

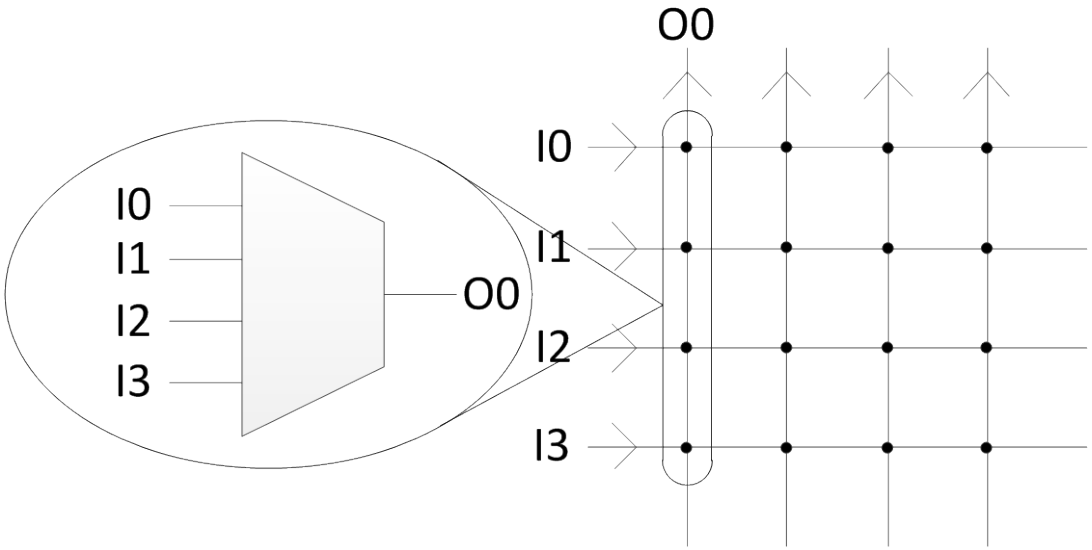
- Two approaches to dynamic reconfiguration of NoC
  - Adding reconfiguration logic
    - Incurs area overhead
  - Partial reconfiguration (PR)
    - High reconfiguration delay
    - Requires large storage space
    - Caters to predefined use cases only

Contributions

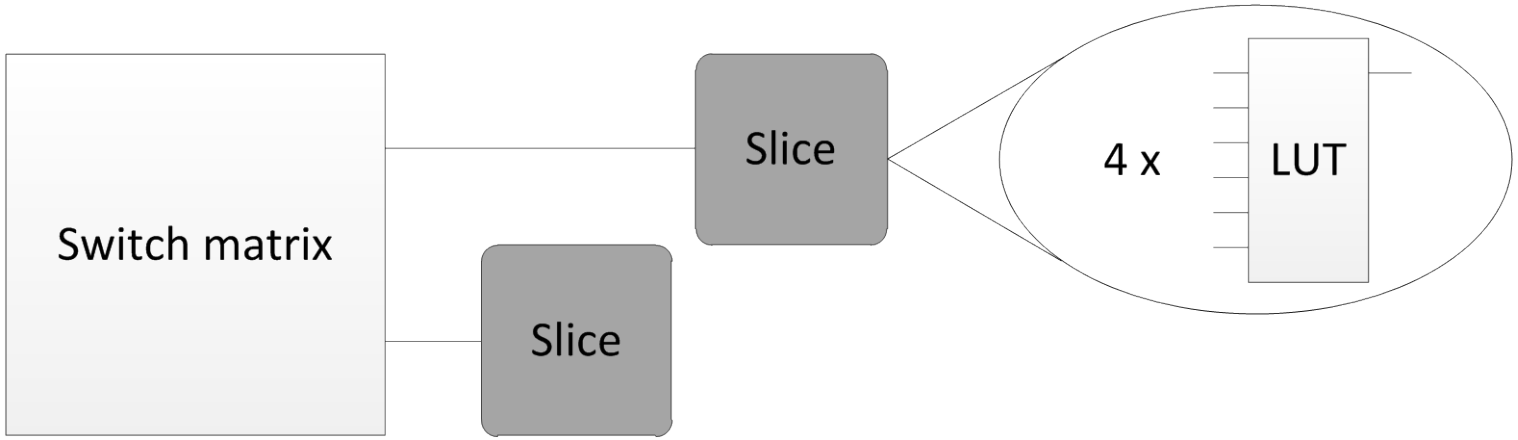
- Problem lies with the design of crossbar switch
- A novel partially reconfigurable crossbar switch design is proposed
  - 84% area saving!
  - 78% reconfiguration delay reduction!
  - Runtime bit-stream generation
  - Glitch-free reconfiguration

Crossbar Switch Architecture

Multiplexer as building block of crossbar switch



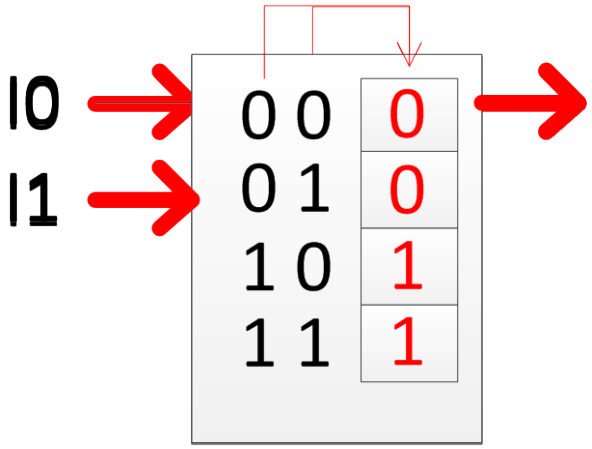
A Configurable Logic Block (CLB)



LUT AS MULTIPLEXER

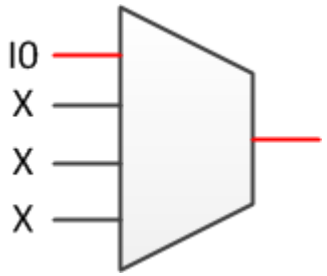
- Conventional LUT multiplexer
  - Requires log2N selector pins for N data inputs
- PR based LUT multiplexer
  - Requires no dedicated selector pins
  - Selection is done by changing LUT content through PR
  - Allows larger multiplexer to be built with the same number of LUTs

LUT AS MULTIPLEXER

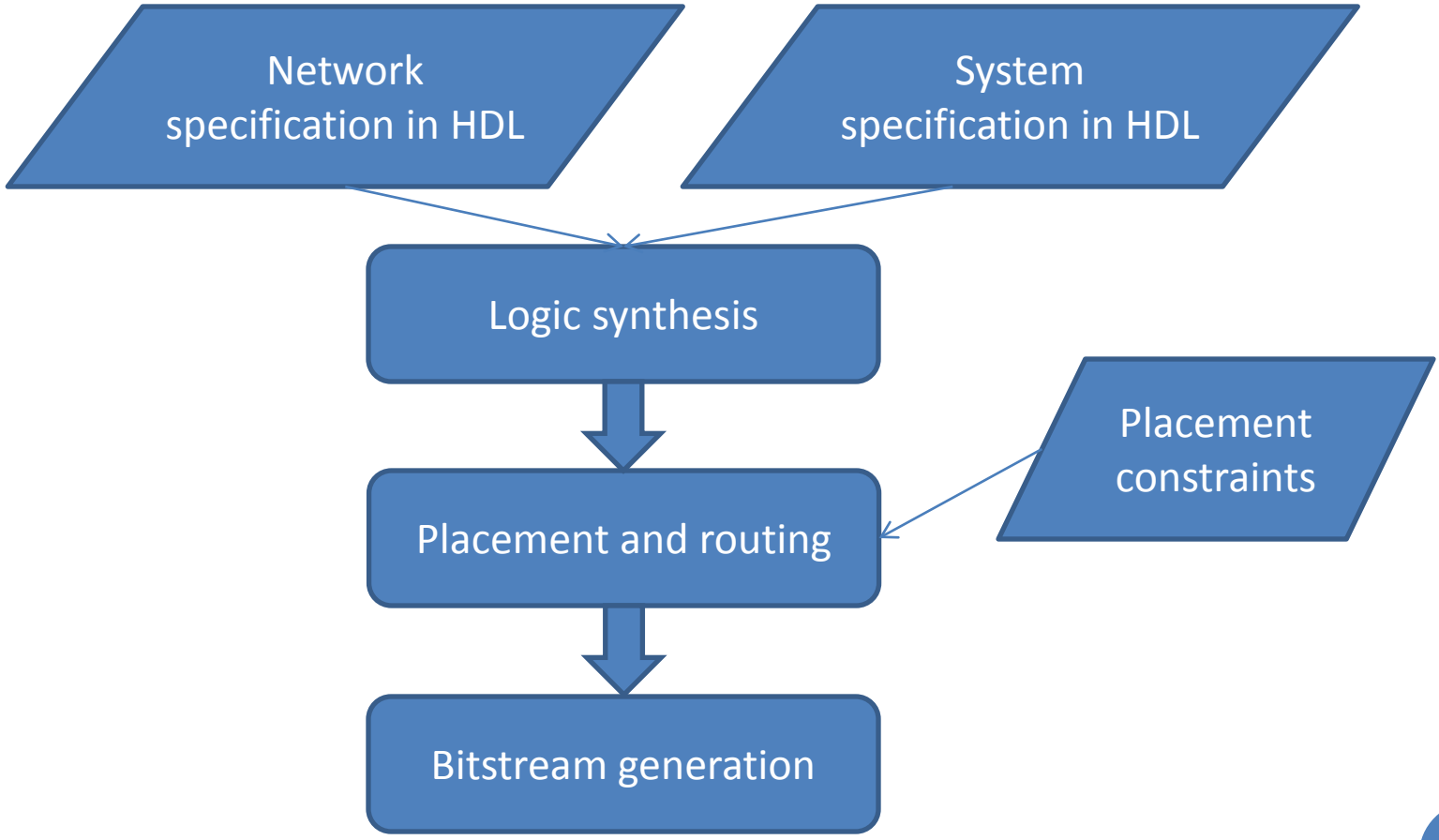


Scalable Approach

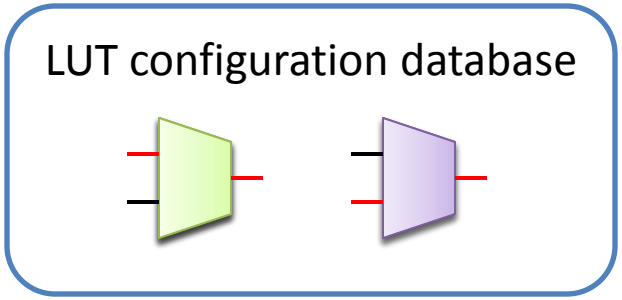
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1



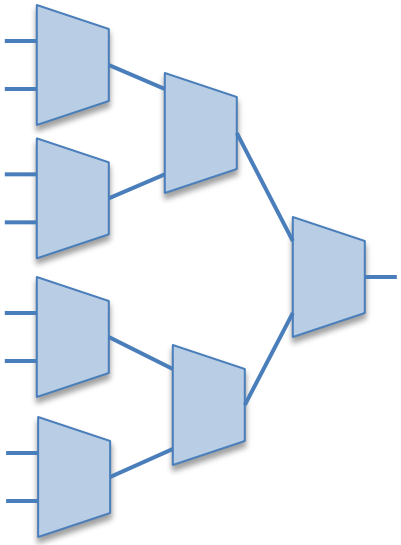
Design Flow


Network specification in HDL
System specification in HDL
Logic synthesis
Placement constraints
Placement and routing
Bitstream generation

Runtime bitstream generation & Reconfiguration



ICAP

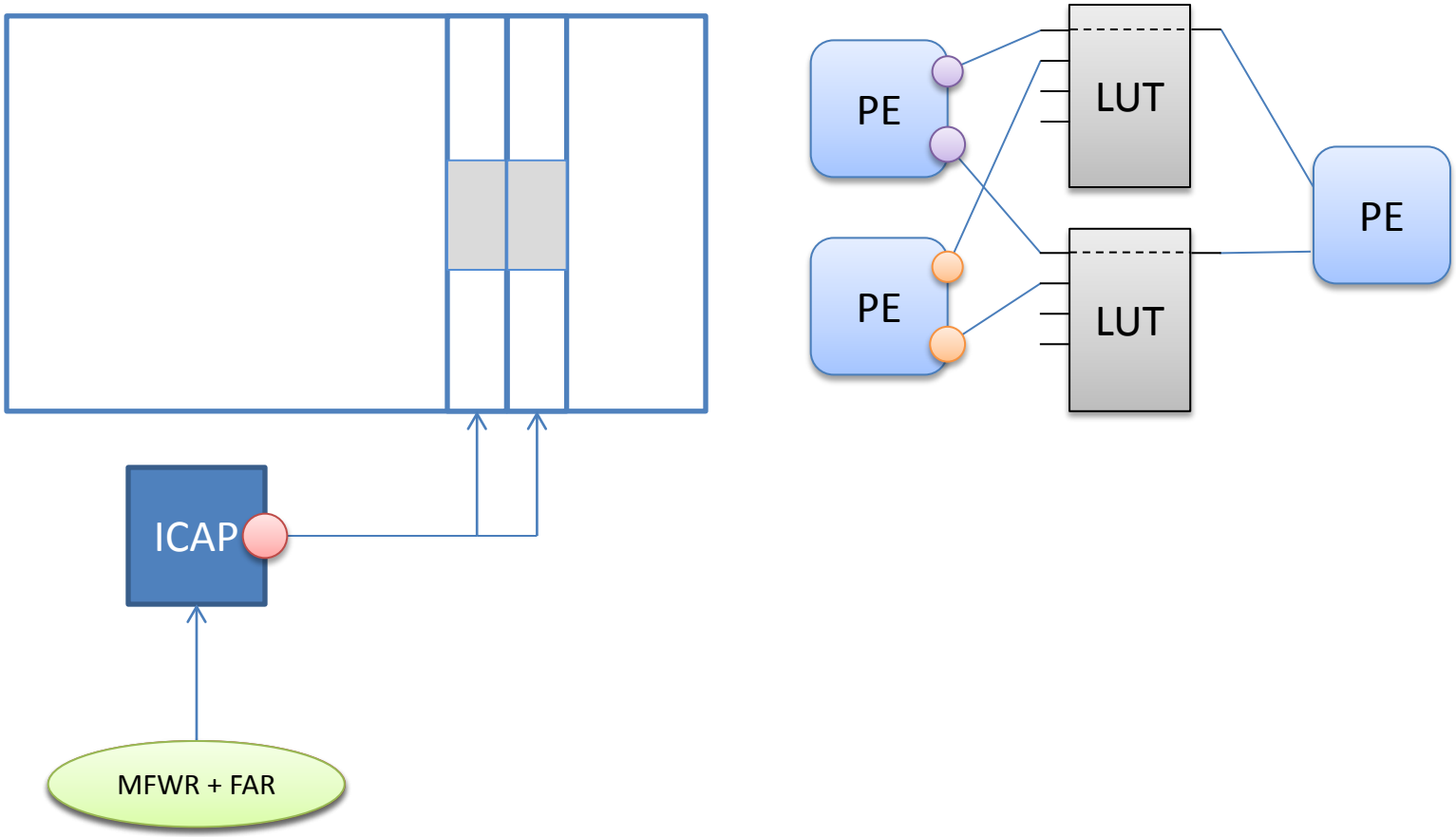


Minimizing reconfiguration delay

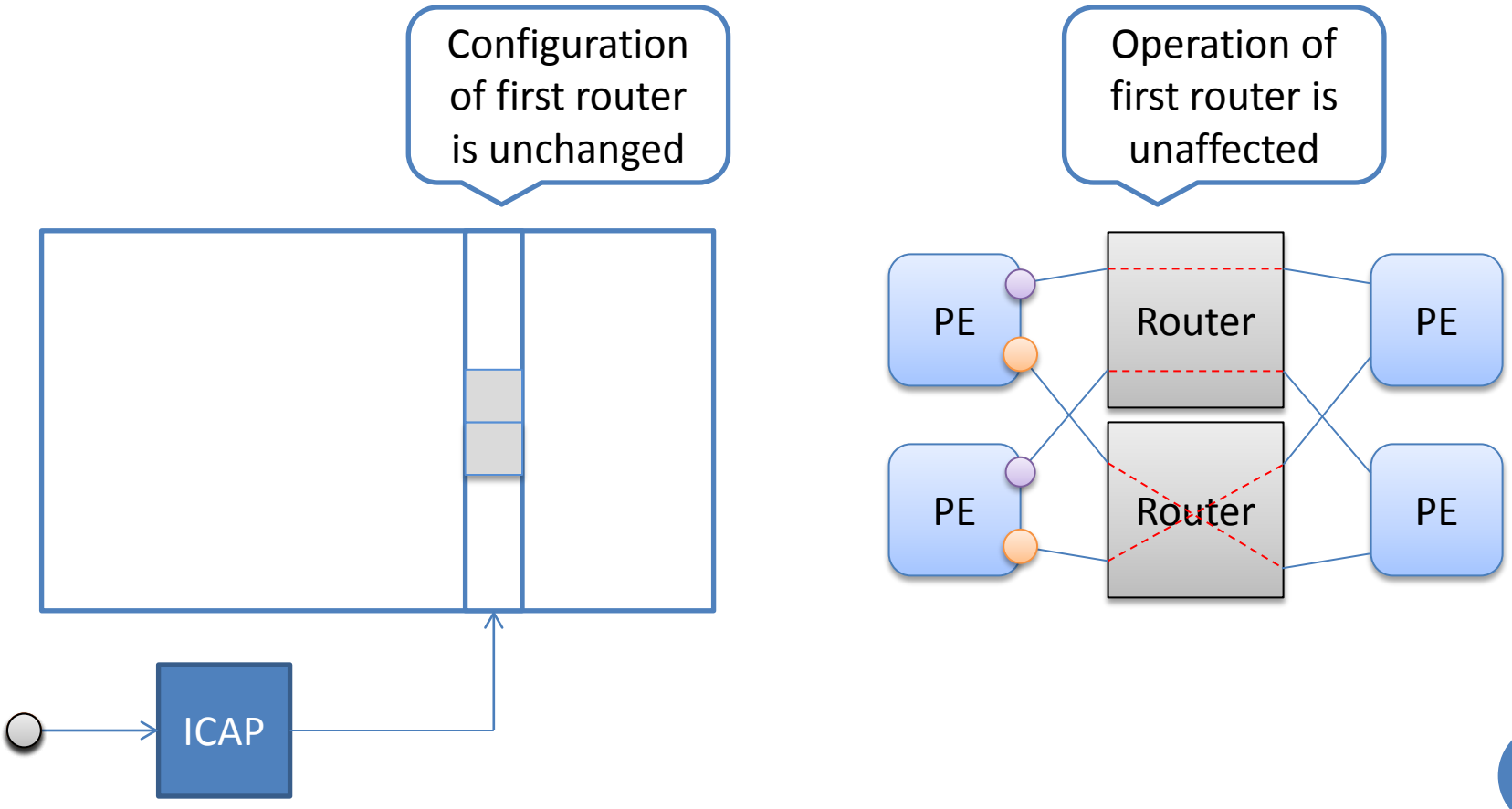
- Apply AREA_GROUP constraint to limit the placement of LUT to the minimum number of CLB columns required
- Identify the frames that are responsible for LUT content
  - 8 out of 36 frames are required
  - 78% speedup!
- Use the Multiple Frame Write (MFW) command of ICAP

Multiple Frame Write (MFW)

2 LUTs configured identically by transferring the configuration data only once!

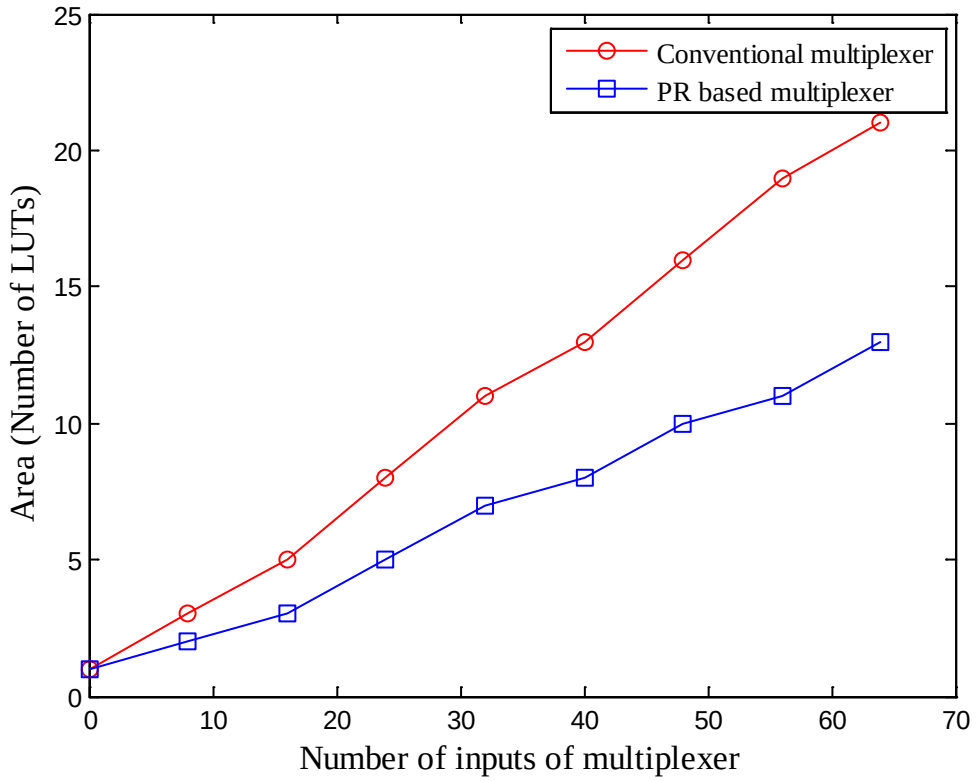


Glitch-Free Connection Establishment

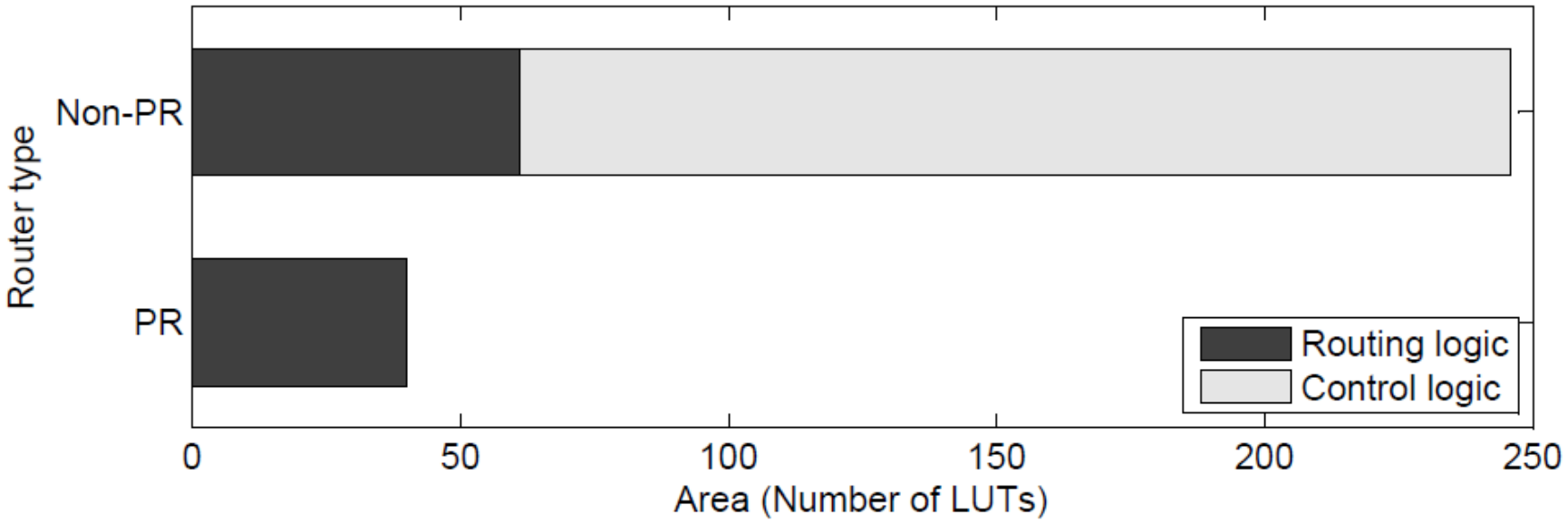


Results and Analysis

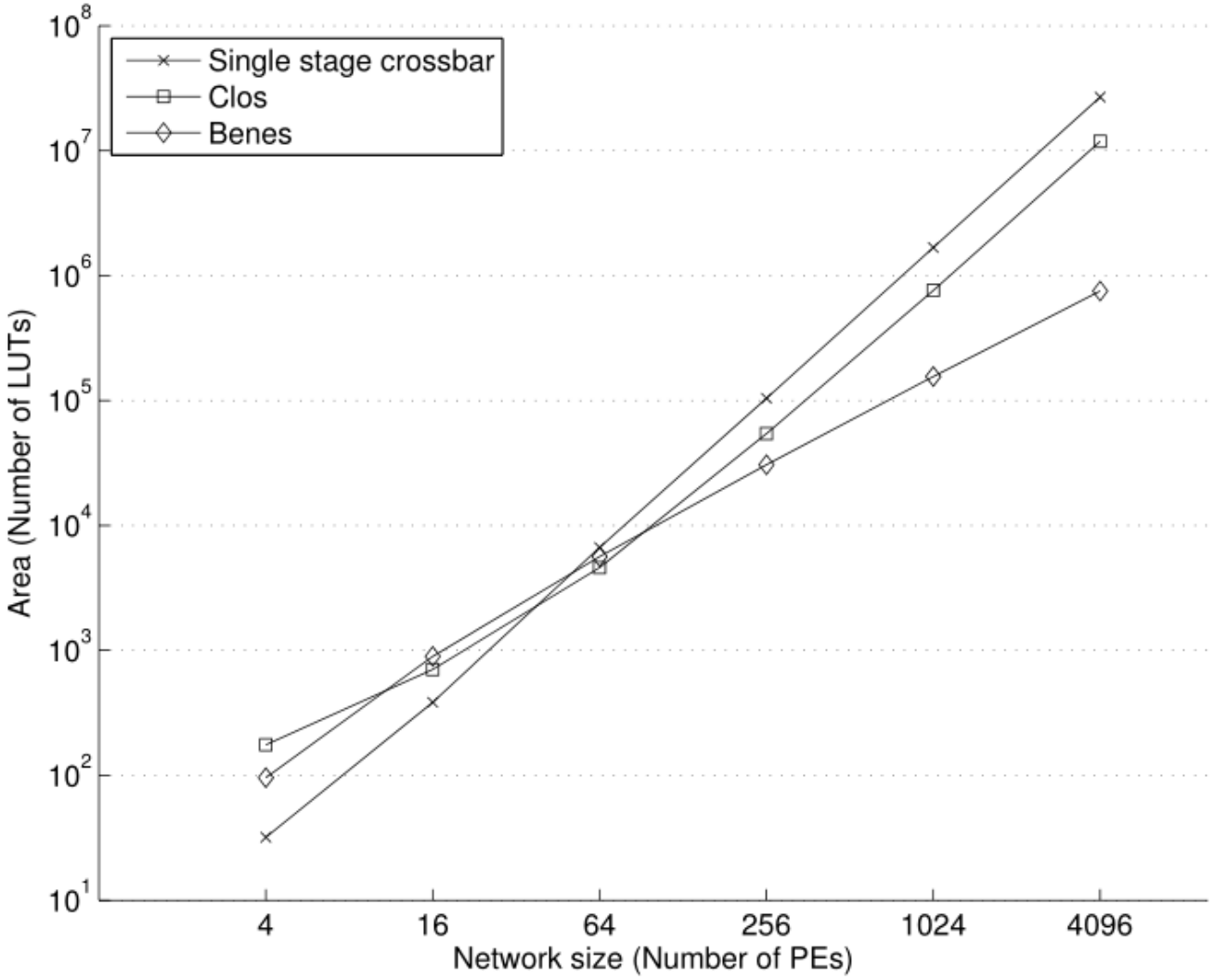
AREA SAVING OF PR ROUTER – SINGLE LINK



Area savings of PR router



Area requirement of various network topologies



Conclusions

- A novel partially reconfigurable crossbar switch design has been presented
  - 84% area saving!
  - 78% reconfiguration delay reduction!
  - Runtime bitstream generation
  - Glitch-free reconfiguration

Questions?