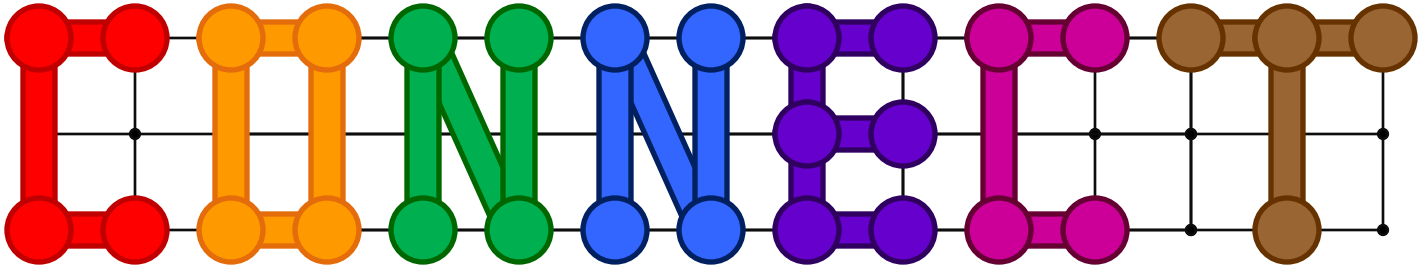


Re-Examining Conventional Wisdom for Networks-on-Chip in the Context of FPGAs

Michael K. Papamichael, James C. Hoe
papamix@cs.cmu.edu, jhoe@ece.cmu.edu

Computer Architecture Lab at
Carnegie Mellon

Monterey, CA, February 2012

This work was funded by NSF. We thank Xilinx for their FPGA and tool donations. We thank Bluespec for their tool donations.

FPGAs and Networks-on-Chip (NoCs)

- **Rapid growth of FPGA capacity and features**
  - Extended SoC and full-system prototyping
  - FPGA-based high-performance computing

Need for flexible NoCs to support communication

Map existing ASIC-oriented NoC designs on FPGAs?

FPGAs and Networks-on-Chip (NoCs)

- **Rapid growth of FPGA capacity and features**
  - Extended SoC and full-system prototyping
  - FPGA-based high-performance computing

Need for flexible NoCs to support communication

Map existing ASIC-oriented NoC designs on FPGAs?

- Inefficient use of FPGA resources
- ASIC-driven NoC architecture not optimal for FPGA

FPGAs and Networks-on-Chip (NoCs)

- **Rapid growth of FPGA capacity and features**
  - Extended SoC and full-system prototyping
  - FPGA-based high-performance computing

Need for flexible NoCs to support communication

Map existing ASIC-oriented NoC designs on FPGAs?

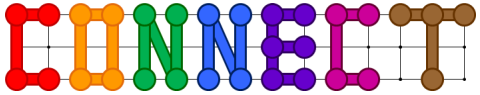
- Inefficient use of FPGA resources
- ASIC-driven NoC architecture not optimal for FPGA

CONNECT **FPGA-tuned NoC Architecture**

- Embodies FPGA-motivated design principles
- Very lightweight, minimizes resource usage
  - **~50% resource reduction** vs. ASIC-oriented NoC
- Flexible NoC generator – **public release today!**

Often goes against ASIC-driven NoC conventional wisdom

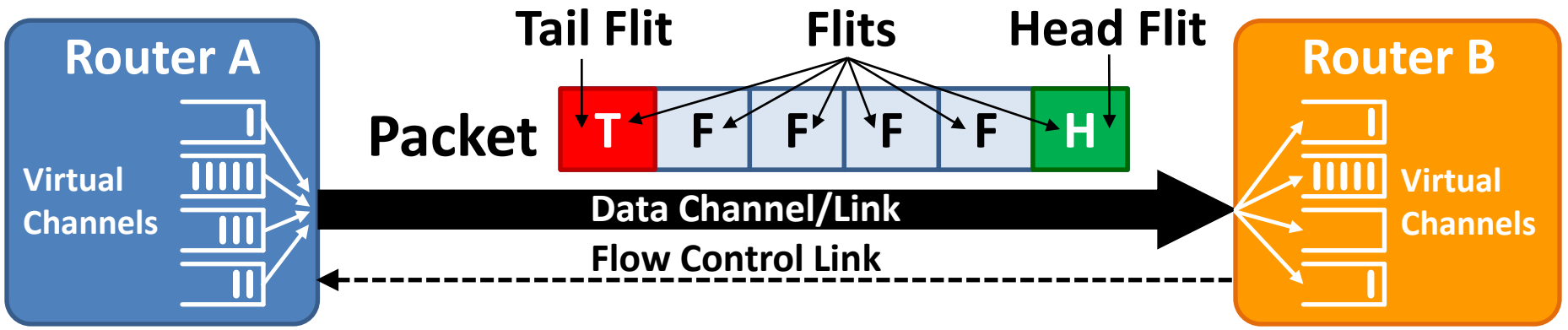
Outline

- **NoC Terminology** (single-slide review)
-  **Approach**
  - Tailoring NoCs to FPGAs
- **Results & Demo**
- **Related Work & Conclusion**
- **Public Release**

Outline

- **NoC Terminology** (single-slide review)
- CONNECT Approach
  - Tailoring NoCs to FPGAs
- Results & Demo
- Related Work & Conclusion
- Public Release

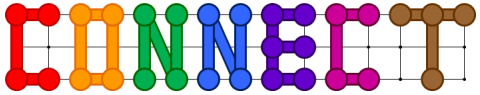
NoC Terminology Overview



- **Packets**
  - Basic logical unit of transmission
- **Flits**
  - Packets broken into into multiple **flits** – unit of flow control
- **Virtual Channels**
  - Multiple logical channels over single physical link
- **Flow Control**
  - Management of buffer space in the network

Single-Slide Review

Outline

- NoC Terminology (single-slide review)
-  **Approach**
  - Tailoring NoCs to FPGAs
- Results & Demo
- Related Work & Conclusion
- Public Release

How FPGAs are Different from ASICs

- **FPGAs peculiar HW realization substrate in terms of**
  - Relative cost of **speed** vs. **logic** vs. **wires** vs. **memory**
  - Unique mapping and operating characteristics
- **CONNECT focuses on 4 FPGA characteristics:**

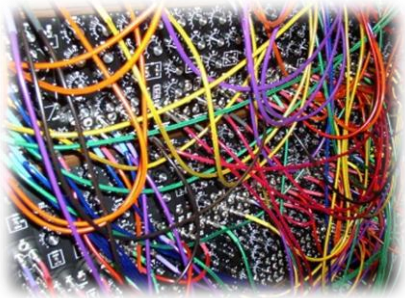
1. **Abundance of Wires**
2. **Storage Shortage & Peculiarities**
3. **Frequency Challenged**
4. **Reconfigurable Nature**

FPGA characteristics uniquely influence key NoC design decisions

Tailoring NoCs to FPGAs

1 Abundance of Wires

- **Densely connected wiring substrate**
  - (Over)provisioned to handle worst case
  - Wires are “free” compared to other resources



CONNECT NoC Implications

- **Make datapaths and channels as wide as possible**
- **Adapt packet format**
  - E.g. carry control info on the side through dedicated links
- **Adapt traditional credit-based flow control**

Tailoring NoCs to FPGAs

2 Storage Shortage & Peculiarities

- **Modern FPGAs offer storage in two forms**
  - Block RAMs and LUT RAMs (use logic resources)
  - Only come in specific aspect ratios and sizes
- **Typically in high demand, especially Block RAMs**

CONNECT NoC Implications

- **Minimize usage and optimize for aspect ratios and sizes**
  - Implement multiple logical flit buffers in each physical buffer
- **Use LUT RAM for flit buffers**
  - Block RAM much larger than typically NoC flit buffer sizes
  - Allow rest of design to use scarce Block RAM resources

Tailoring NoCs to FPGAs

3 Frequency "Challenged"

- **Much lower frequencies compared to ASICs**
  - LUTs inherently slower that ASIC standard cells
  - Large wire delays when chaining LUTs
- **Rapidly diminishing returns when pipelining**
  - Deep pipelining hard due to quantization effects

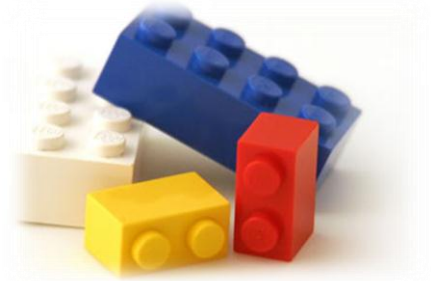
CONNECT NoC Implications

- **Design router as single-stage pipeline**
  - Also dramatically reduces network latency
- **Make up for lower frequency by adjusting network**
  - E.g. increase width of datapath and links or change topology

Tailoring NoCs to FPGAs

4 Reconfigurable Nature

- **Reconfigurable nature of FPGAs**
  - Sets them apart from ASICs
  - Support diverse range of applications



CONNECT NoC Implications

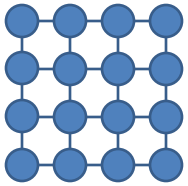
- **Support extensive application-specific customization**
  - Flexible parameterized NoC architecture
  - Automated NoC design generator (demo!)
- **Adhere to standard common interface**
  - NoC appears as plug-and-play black box from user-perspective

Outline

- NoC Terminology (single-slide review)
- CONNECT Approach
  - Tailoring NoCs to FPGAs
- **Results & Demo**
- Related Work & Conclusion
- Public Release

CONNECT vs. ASIC-Oriented RTL

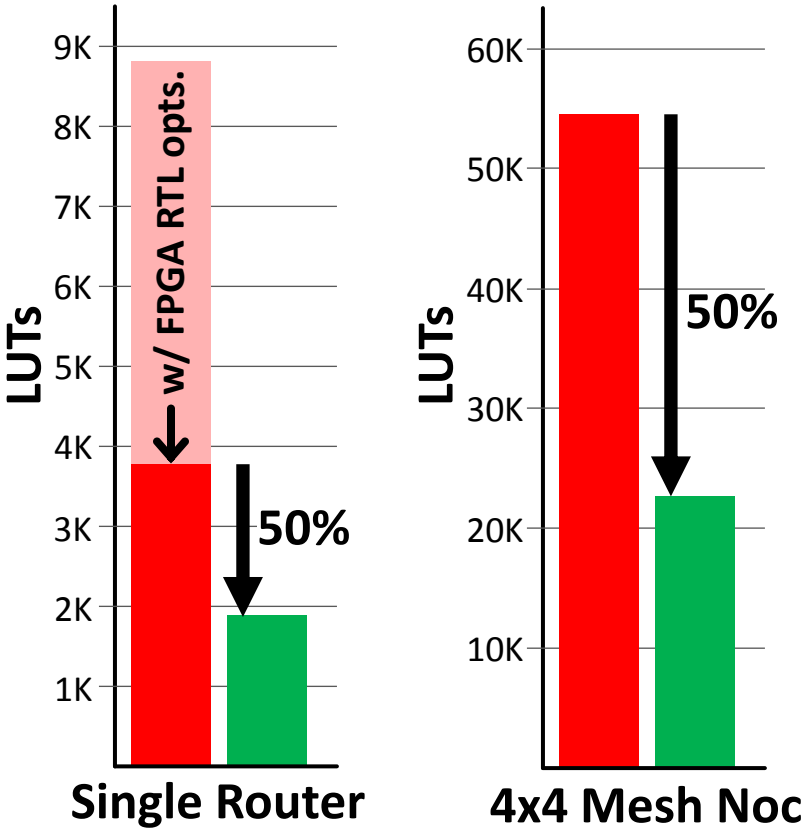
16-node 4x4 Mesh Network-on-Chip (NoC)



- **SOTA:** state-of-the-art high-quality ASIC-oriented RTL*
- **CONNECT:** identically configured CONNECT-generated RTL

FPGA Resource Usage

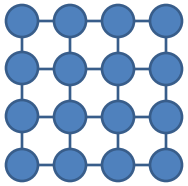
(same router/NoC configuration)



*NoC RTL from http://nocs.stanford.edu/cgi-bin/trac.cgi/wiki/Resources/Router

CONNECT vs. ASIC-Oriented RTL

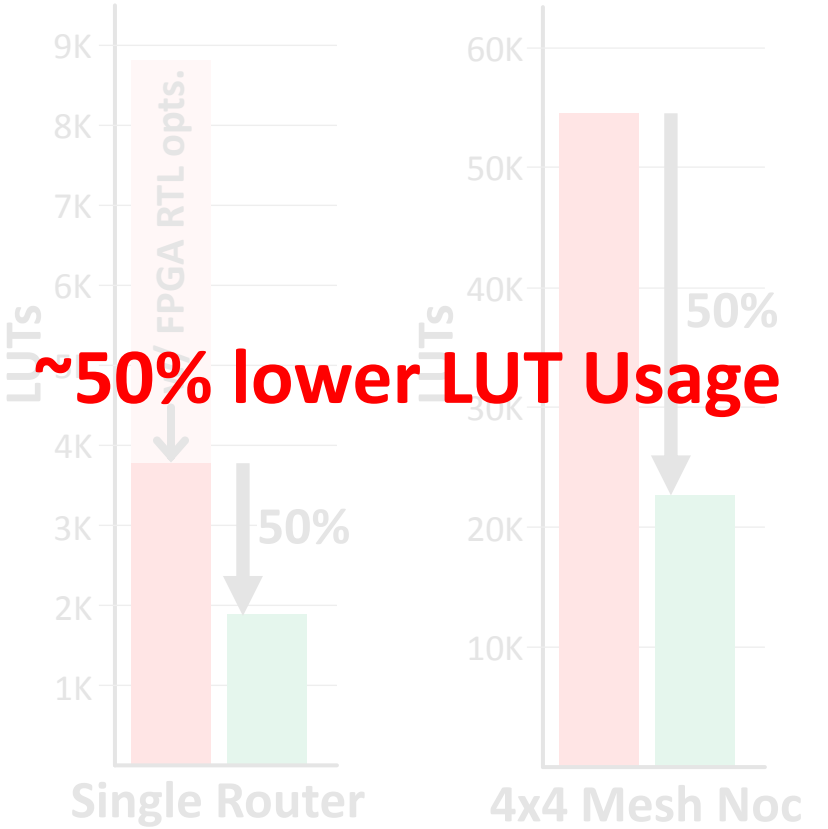
16-node 4x4 Mesh Network-on-Chip (NoC)



- **SOTA:** state-of-the-art high-quality ASIC-oriented RTL*
- **CONNECT:** identically configured CONNECT-generated RTL

FPGA Resource Usage

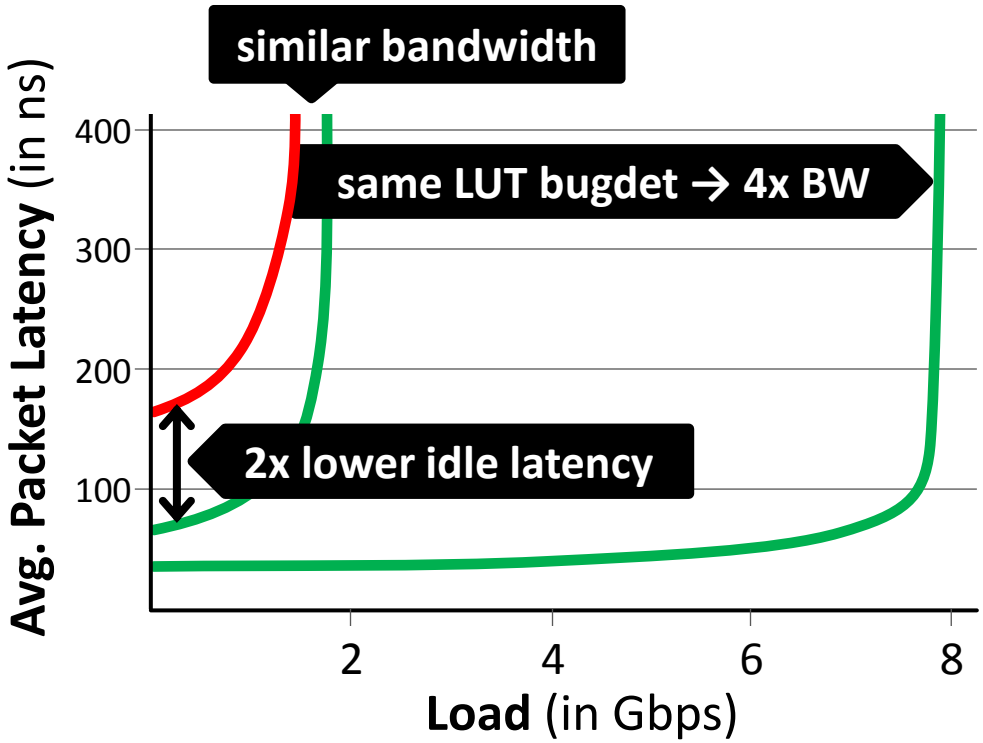
(same router/NoC configuration)



~50% lower LUT Usage

Network Performance

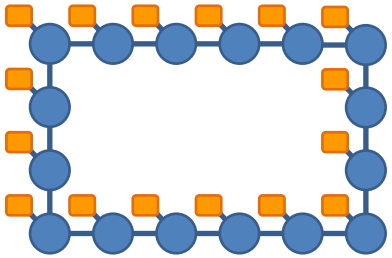
(uniform random traffic @ 100MHz)



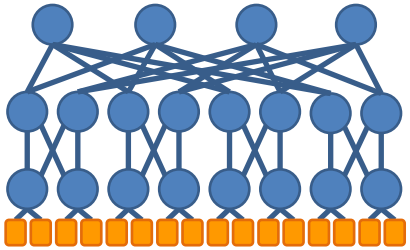
*NoC RTL from http://nocs.stanford.edu/cgi-bin/trac.cgi/wiki/Resources/Router

CONNECT Sample Networks

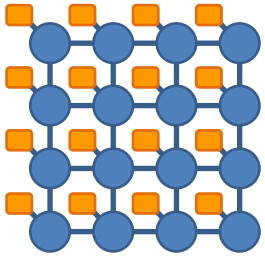
- **Four sample CONNECT Networks** (router, endpoint)
  - 16 endpoints, 2/4 virtual channels, 128-bit datapath



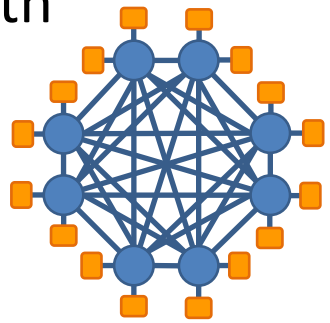
Ring



Fat Tree

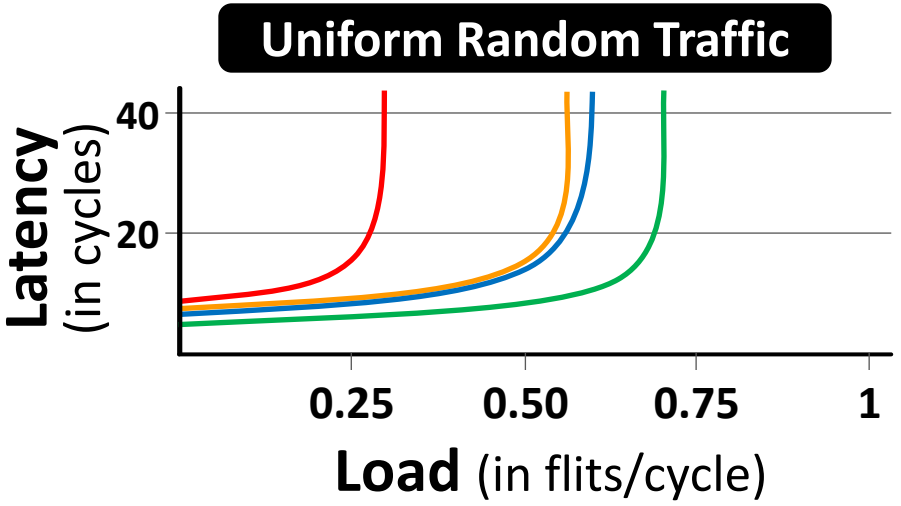


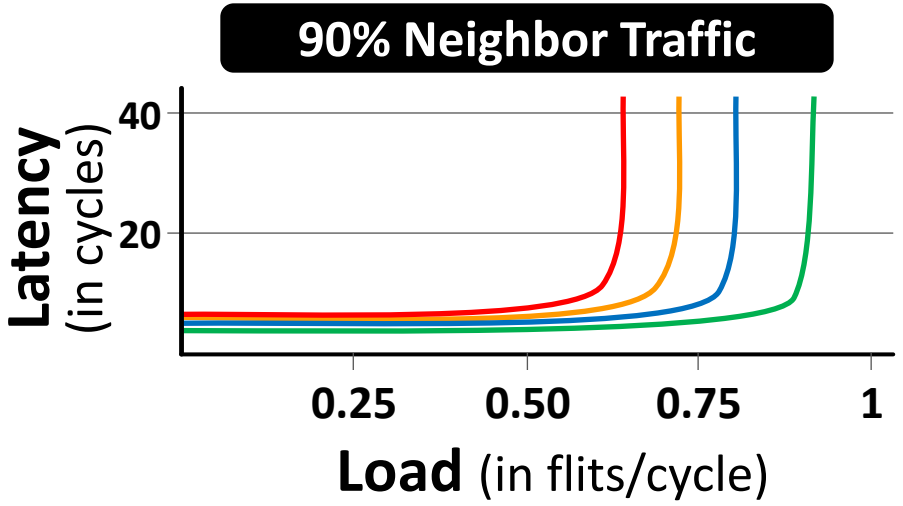
Mesh



High Radix

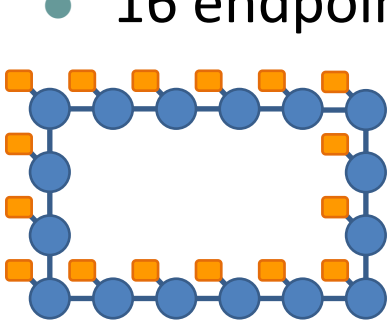
All above networks are interchangeable from user perspective



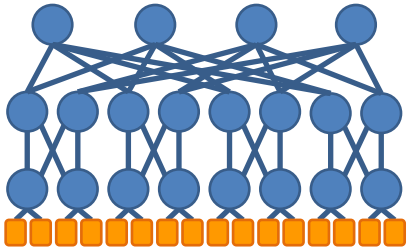


CONNECT Sample Networks

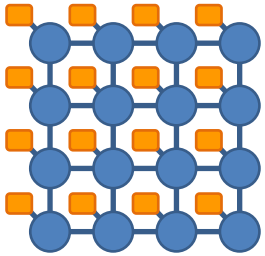
- **Four sample CONNECT Networks** (router, endpoint)
  - 16 endpoints, 2/4 virtual channels, 128-bit datapath



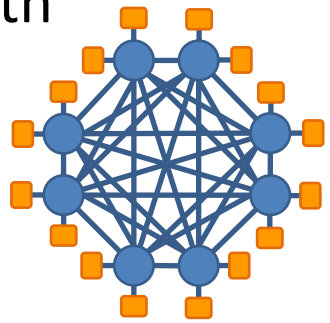
Ring



Fat Tree

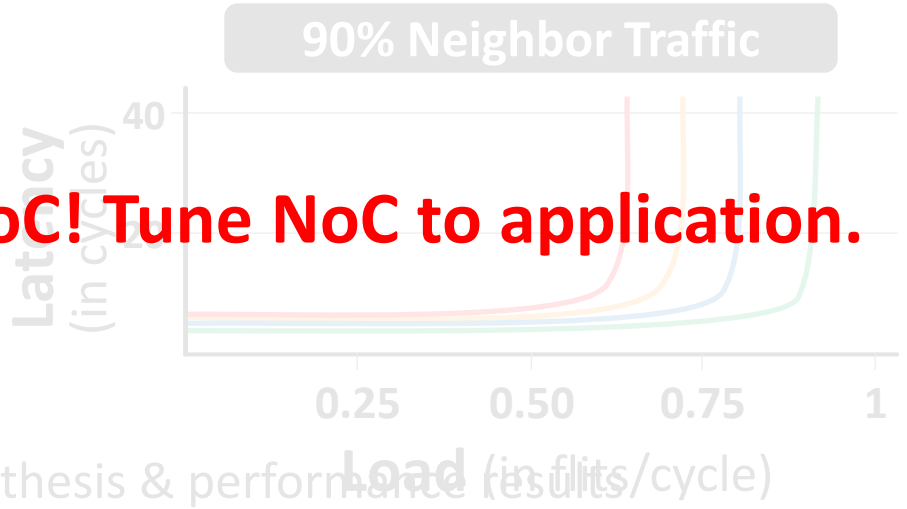


Mesh



High Radix

All above networks are interchangeable from user perspective



There is no one-size-fits-all NoC! Tune NoC to application.

please see paper for more synthesis & performance results

CONNECT NoC Generator

- **Generates FPGA-tuned NoCs**
  - Used as part of larger system
  - Coexist with rest of FPGA-resident components
- **Lightweight**
  - Minimize FPGA resource usage
- **Flexible**
  - Support rapid design space exploration
  - Fully parameterized, topology-agnostic architecture
  - Standard interface across different configurations
- **Adjustable**
  - Trade off performance vs. area
- **Public Release – Demo!**

Outline

- NoC Terminology (single-slide review)
- CONNECT Approach
  - Tailoring NoCs to FPGAs
- Results & Demo
- **Related Work & Conclusion**
- Public Release

Related Work

- **FPGA-oriented NoC Architectures**
  - PNoC: lightweight circuit-switched NoC [Hilton '06]
  - NoCem: simple router block, no virtual channels [Schelle '08]
- **FPGA-related NoC Studies**
  - Analytical models for predicting NoC perf. on FPGAs [Lee '10]
  - Effect of FPGA NoC params on multiproccesor system [Lee '09]
- **Modify FPGA configuration circuitry to build NoC**
  - Metawire: use configuration circuitry as NoC [Shelburne '08]
  - Time-division multiplexed wiring to enable new NoC [Francis '08]
- **Commercial Interconnect Approaches**
  - ARM AMBA, STNoC, CoreConnect PLB/OPB, Altera Qsys, etc.

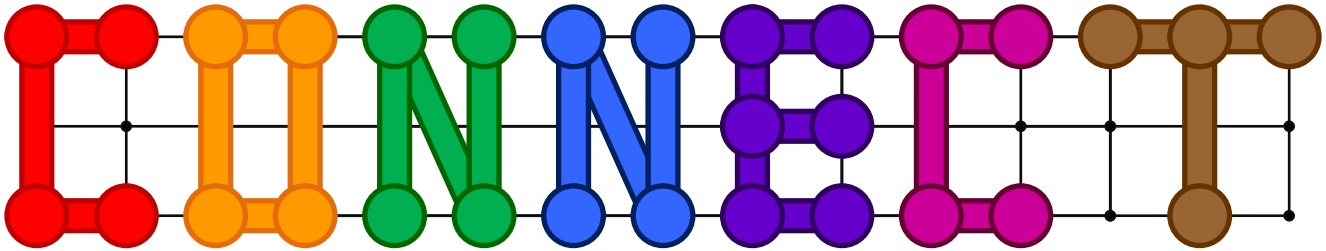
Conclusions

- **Significant gains from tuning for FPGA**
  - FPGAs and ASICs have different design "sweet spot"
- **CONNECT → flexible, efficient, lightweight NoC**
- **Compared to ASIC-driven NoC, CONNECT offers**
  - Significantly lower network latency **and**
  - ~50% lower LUT usage **or** 3-4x higher network performance
- **Take advantage of reconfigurable nature of FPGA**
  - Tailor NoC to specific communication needs of application

Outline

- NoC Terminology (single-slide review)
- CONNECT Approach
  - Tailoring NoCs to FPGAs
- Results & Demo
- Related Work & Conclusion
- **Public Release**

Public Release



http://www.ece.cmu.edu/~mpapamic/connect/

- **NoC Generator with web-based interface**
  - Supports multiple pre-configured topologies
  - Includes graphical editor for custom topologies
  - FreeBSD-like license (limited to non-commercial research use)
- **Acknowledgments**
  - Derek Chiou, Daniel Becker & Stanford CVA group
  - NSF, Xilinx, Bluespec
- **Please come see me for demo!**

Thanks!

Questions?