Effects of I/O Routing Through Column Interfaces in Embedded FPGA Fabrics

Christophe Huriaux ❖, Olivier Sentieys ❖, Russell Tessier ★

Inria, Rennes, FR ❖
University of Massachusetts, Amherst, USA ★
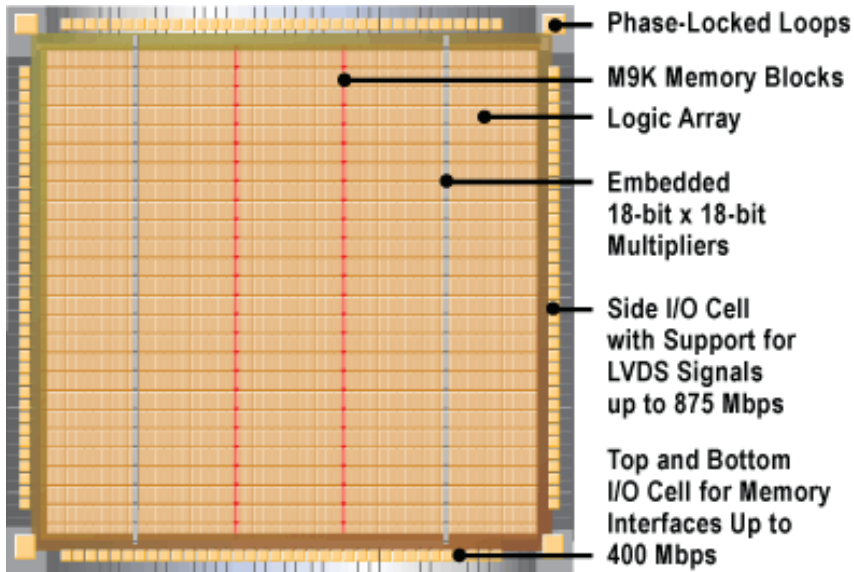
UMASS AMHERST

Inria informatics mathematics

UNIVERSITÉ DE RENNES 1
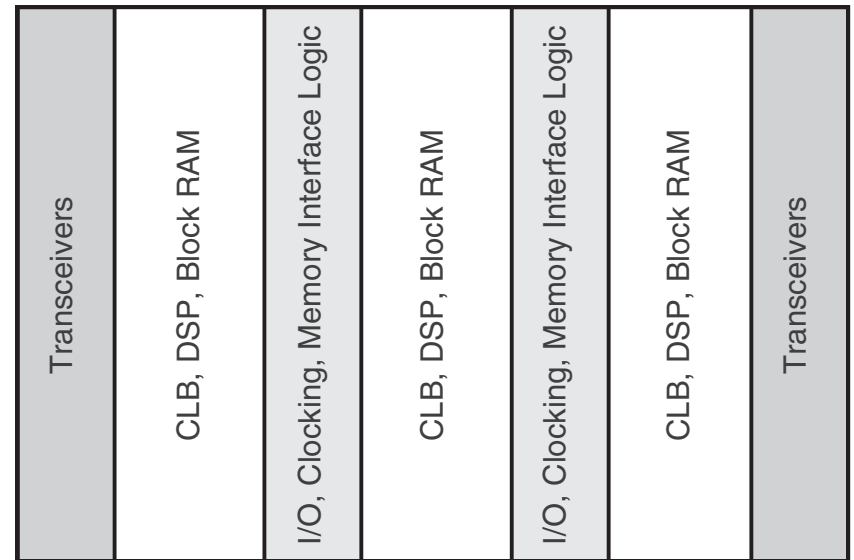
Inria

Overview

- Introduction

  - Motivational example: the FlexTiles platform

- Approach

  - Interface models

  - Implementation methodology

- Experimental results

  - Placement and routing quality of results (QoR)

  - Performance evaluation

- Conclusion

Introduction

- Field-Programmable Gate Arrays (FPGAs) are ubiquitous in the reconfigurable hardware market
- Many applications have high bandwidth requirements
- Input and output (I/O) signals are usually handled through simple I/O blocks or transceiver interfaces
  - I/Os arranged in an outer ring or in columns

- Phase-Locked Loops
- M9K Memory Blocks
- Logic Array
- Embedded 18-bit x 18-bit Multipliers
- Side I/O Cell with Support for LVDS Signals up to 875 Mbps
- Top and Bottom I/O Cell for Memory Interfaces Up to 400 Mbps

Altera Cyclone III floorplan [Alt16]

| Transceivers | CLB, DSP, Block RAM | I/O, Clocking, Memory Interface Logic | CLB, DSP, Block RAM | I/O, Clocking, Memory Interface Logic | CLB, DSP, Block RAM | Transceivers |

Xilinx Ultrascale logic resources organization [Xil16]

2.5D and 3D technologies

- 2.5D and 3D packaging technologies are increasingly used in large circuits
  - Higher yield (smaller ICs on an interposer)
  - Complex heterogeneous 3D-stacked systems with an FPGA

layer, processor cores

- Communication between components in these FPGA-based systems often take place through dedicated bus or Network-on-Chip (NoC) interfaces

Motivational example: FlexTiles platform

- FlexTiles architecture : 3D-stacked heterogeneous manycore [Lem12]

  - Manycore layer with General Purpose and Digital Signal Processors (GPP, DSP)

  - Hardware accelerators mapped on a reconfigurable FPGA layer

  - Network-on-Chip to interconnect the computing resources
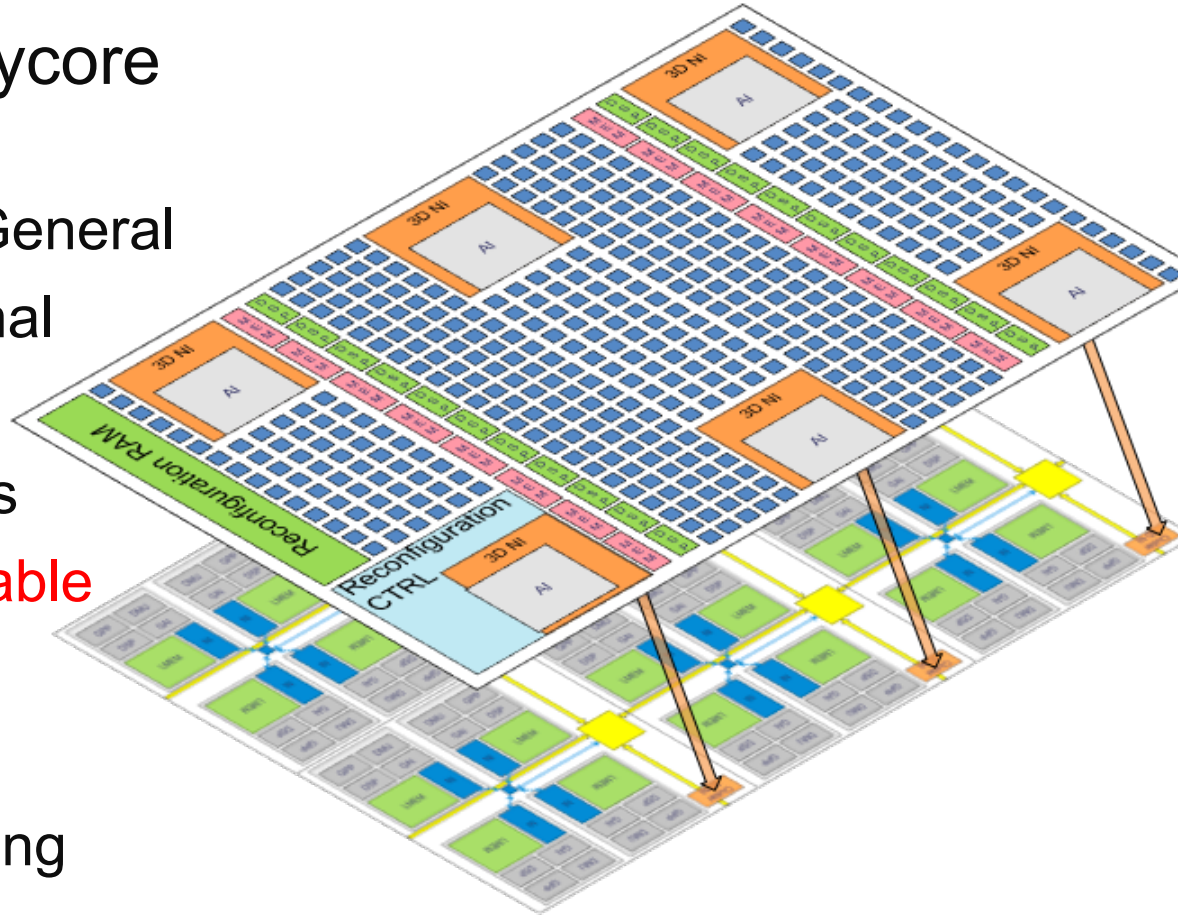
Target applications

- Platform aimed at streaming applications
 - Kernels are partitioned to fit FPGA hardware modules and software GPP / DSP tasks
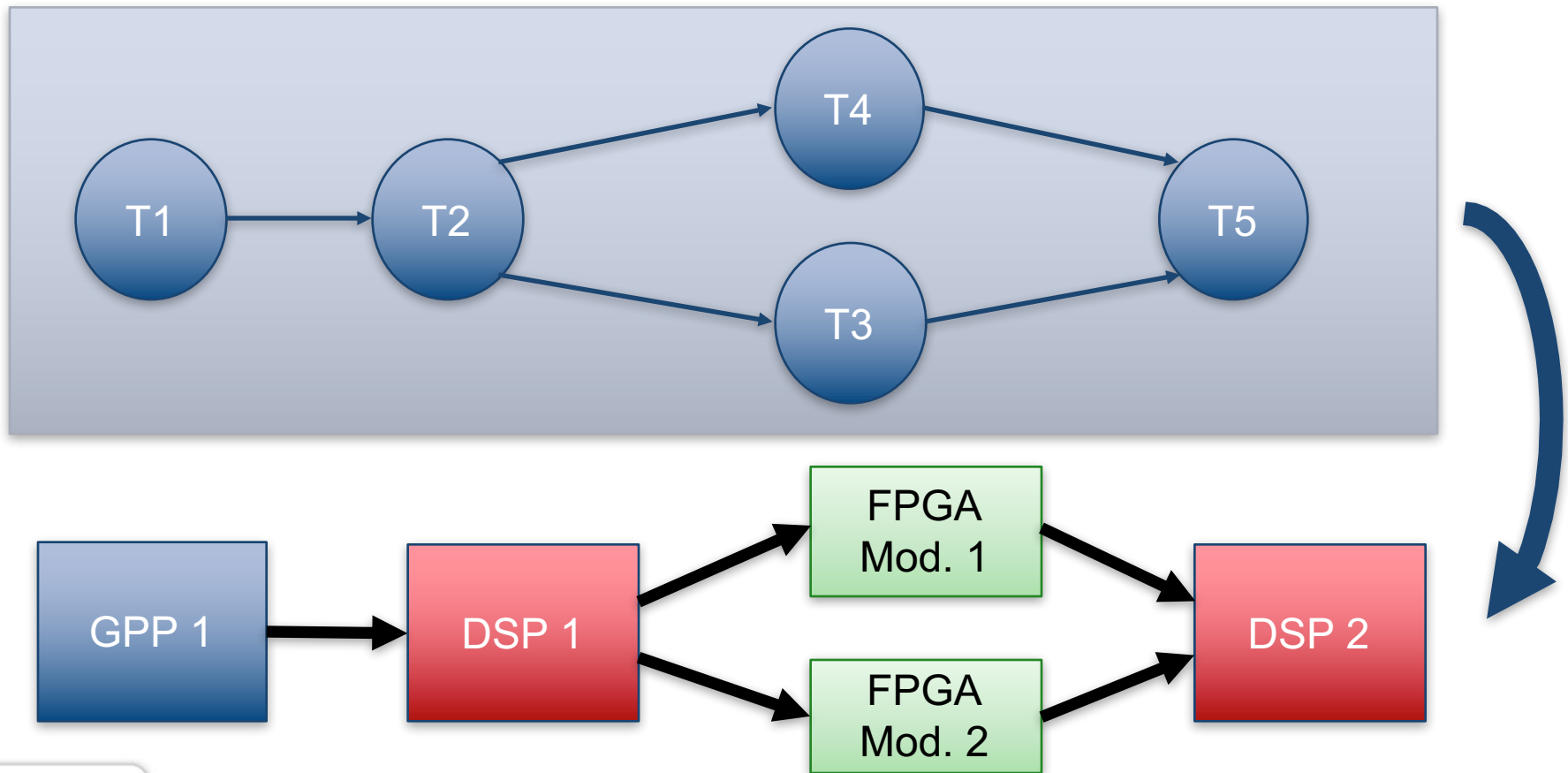
Impact of dedicated interfaces

- *Hardware tasks* are logic modules placed on FPGA logic fabric

- Communications between e.g. processors and hard tasks take place through dedicated, coarse-grained interfaces

- *What is the impact of such interfaces on the placement and routing QoR of FPGA modules ?*

Model of the interfaces
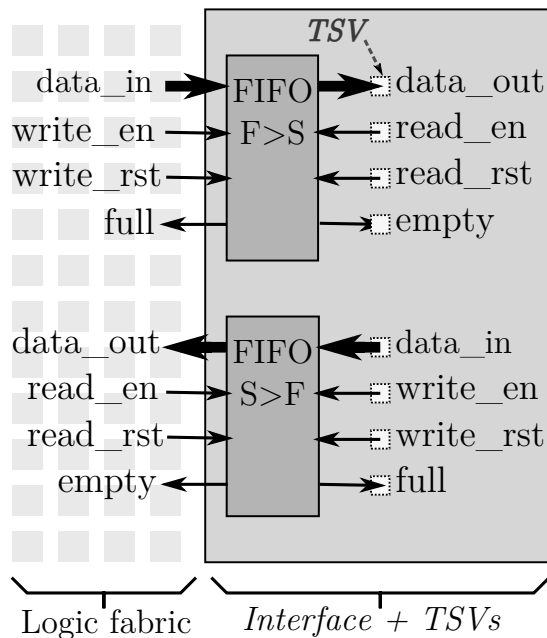
- Generic interface model
  - Read and write FIFOs
  - Separate clock domains

- Variable data size
  - W input/output data bits

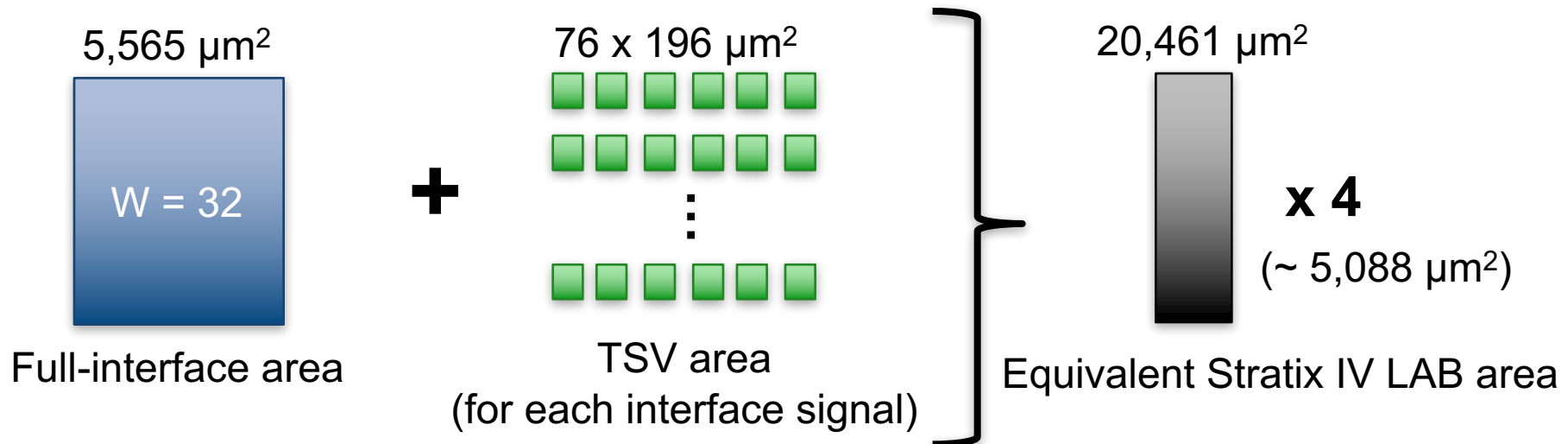- Two FIFOs for bi-directional communications
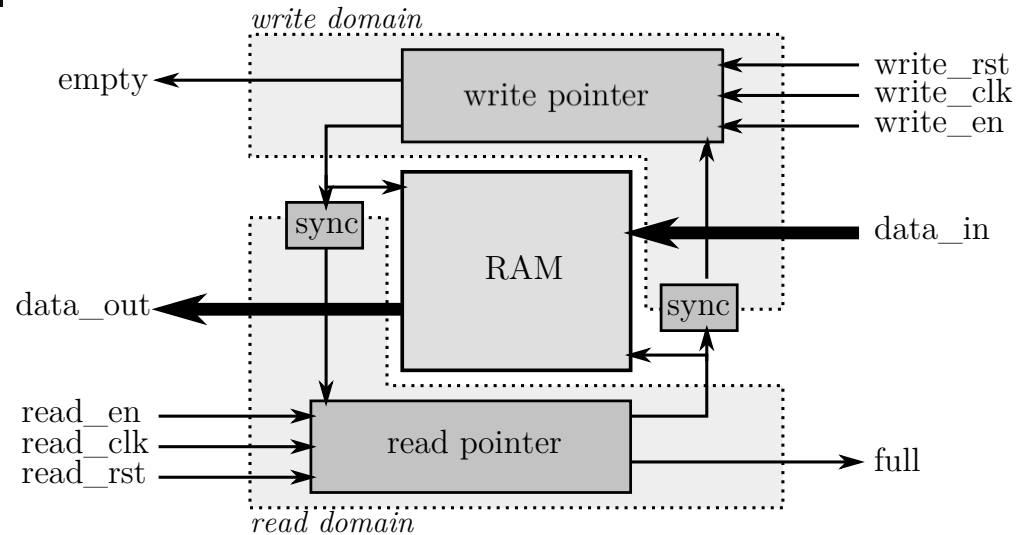
Full and I/O-only models

- Two interface implementations
  - *Full* interface: only control and data signals exposed to the fabric
  - *I/O-only* interface: FIFO and control logic implemented with FPGA logic
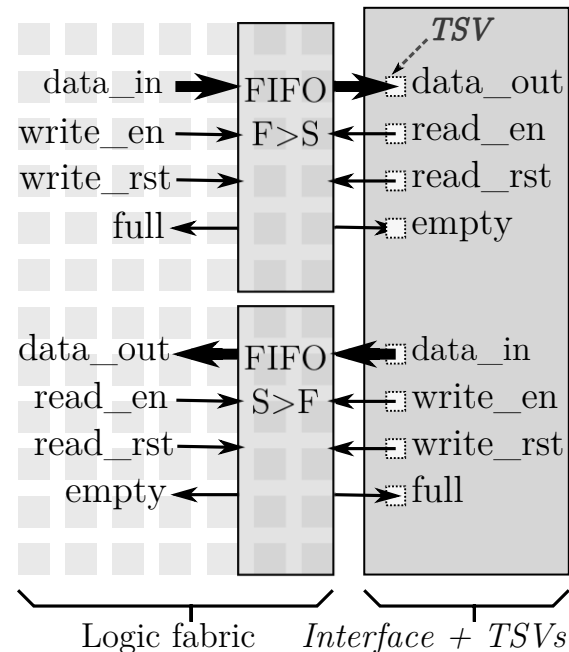
Interface modeling in Quartus

- Architectural exploration using Verilog-To-Routing (VTR) [Luu14]
- Quartus yields more accurate performance results
  - Not feasible to define custom hardware blocks
  - Interfaces were modeled with dummy logic
  - Dummy logic resource count depends on the interface size

5,565 μm^2

W = 32

Full-interface area

+

76 x 196 μm^2

TSV area
(for each interface signal)

20,461 μm^2

x 4

(~ 5,088 μm^2)

Equivalent Stratix IV LAB area

Interface modeling in Quartus (2)

I/O pads   I/O interface columns   DSP column   RAM column
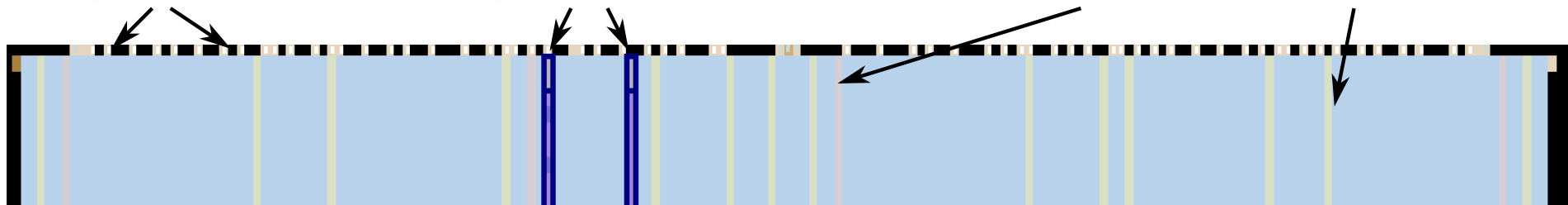
- Dummy LABs arranged contiguously in columns

- Interface columns reserved *every R columns* in Stratix IV

Logic region reserved for the interface

LAB

Interconnect

Interface

Next interface in column
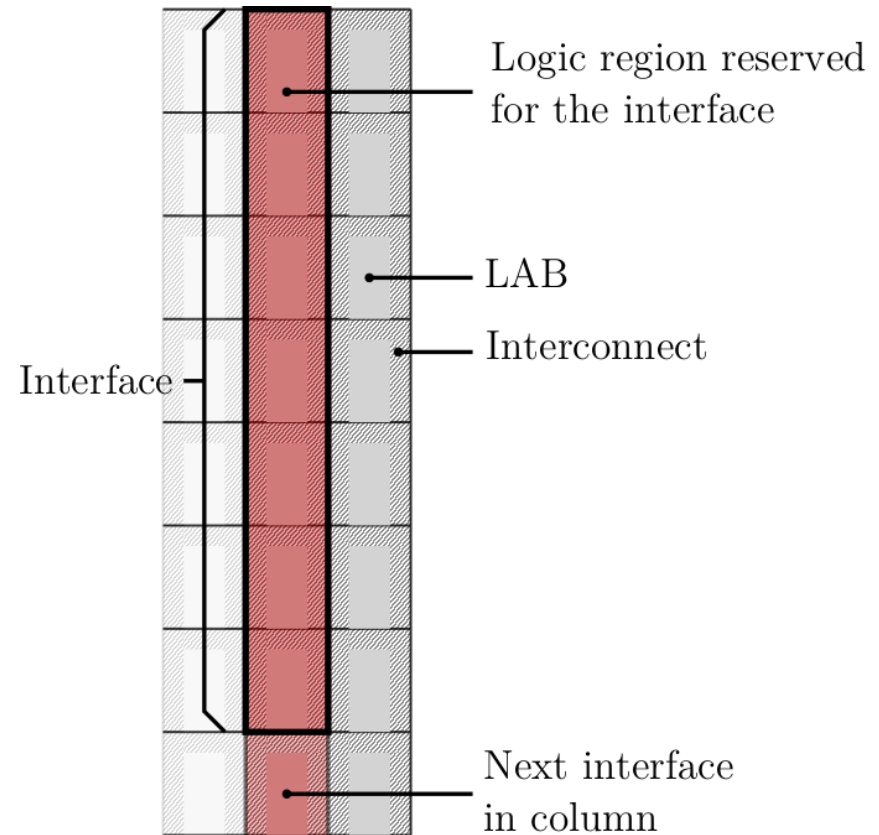
Experimental methodology

- Impact of migrating FPGA I/Os to interface blocks
  - Routability (minimum channel width)
  - Design delay

Channel width
(# of wires/routing channel)
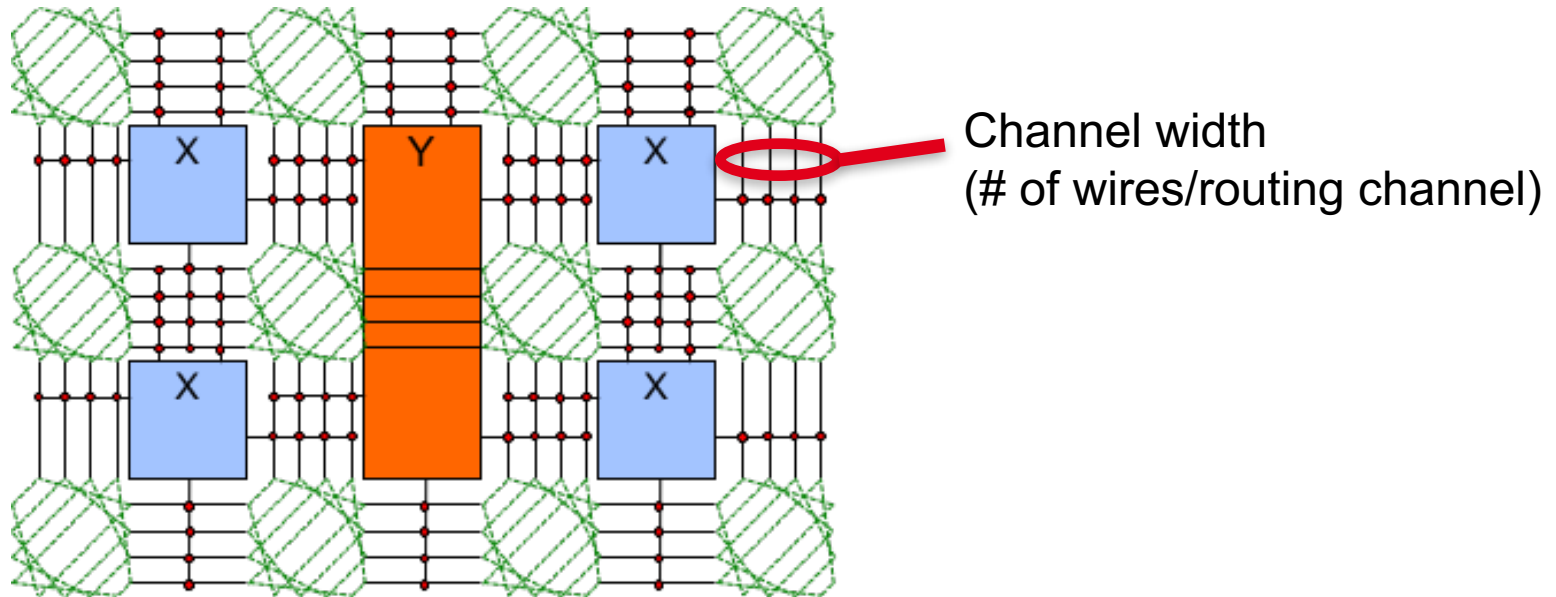
- Placement and routing QoR using VTR
- Performance results using Quartus

Interface-based architecture exploration

- Evolution of an Altera Stratix IV architectural model
  - Clusters of 10 fracturable 6-LUTs
  - 32 Kb single or dual port memories
  - Fracturable 36x36 multipliers

- Custom interface hard block added to the architecture
  - Number of interface columns parameterized by a repeat parameter R
  - Variable interface data width W

- Exploration of varying R, W against a standard, outer I/O-ring Stratix IV architecture

Benchmark set

*   **19 benchmarks** from the VTR benchmark set
    *   I/O count ranging from 40 to 779
    *   Design size up to ~100k 6-LUTs
    *   **Heterogeneous logic resources** including memories,
    multipliers

*   Versatile Place-and-Route (VPR) used to place and route the designs on the **smallest possible** logic fabric
    *   Min. channel width on a **standard architecture** ranges from 34
    wires to 170 wires
    *   Critical path delay ranges from 2.77 ns to 115.5 ns

QoR : full interface

R W	15	20	25	30
32	0.923	0.911	**0.908**	0.911
64	0.954	0.939	0.940	0.940
128	1.065	1.100	**1.104**	1.093

*Average normalized channel width
(w.r.t. standard architecture)*

R W	15	20	25	30
32	1.002	**1.008**	1.003	1.000
64	1.002	0.991	0.987	0.997
128	0.999	0.992	**0.982**	0.995

*Average normalized crit. path delay
(w.r.t. standard architecture)*

- Max ~10% variation of channel width, ~2% of delay
- Larger channel widths with wide interfaces
  - Congestion problems to route signals to/from the interfaces
  - Smaller interfaces min. channel width brought down by small benchmarks with high number of I/Os

QoR : I/O-only interface

R W	15	20	25	30
32	**0.979**	1.003	0.986	0.983
64	1.019	1.005	1.025	1.021
128	1.004	0.998	1.025	**1.034**

Average normalized channel width
(w.r.t. standard architecture)

R W	15	20	25	30
32	1.019	1.011	0.995	**0.994**
64	1.010	1.013	0.998	1.012
128	1.014	**1.024**	1.010	1.010

Average normalized crit. path delay
(w.r.t. standard architecture)

- Max ~3% variation of channel width, ~2% of delay
- More routing stress in comparison to full interfaces
  - Additional logic/memory resources induce overall higher wire-length for the router

Additional resources with I/O-only interfaces

W	Memories	LABs
32	11.87	33.33
64	12.80	25.67
128	15.47	26.07

Average amount of additional resources required for the IO-only architecture

- Higher *W* leads to fewer interfaces
  - Fewer control logic required
  - More memory blocks required to cope with larger data width

Performance evaluation with Quartus

Circuit	Std. arch. F_{max} (MHz)	Full interface arch. F_{max} (MHz)
bgm	81.17	76.48
blob_merge	103.75	108.71
mcml	35.73	35.78
stereovision1	136.93	130.36
stereovision2	113.95	125.08

Performance comparison of the full-interface architecture w.r.t. the standard architecture

- 5 largest circuits used in Quartus with $W = 64$, $R = 25$
- Max. $\pm 10\%$ variation on F_{max}
- Additional LABs required to handle the data to/from the FIFOs

Conclusion

- Traditional outer I/O ring has limited value for fabric embedded in 2.5D and 3D architectures
  - Common FPGA architectures already move towards column I/Os
- Two generic interface models studied
  - Both are implementable with little impact on the placement and routing QoR
  - Up to 10% min. channel width and 3% delay variations on average in comparison to a standard architecture
- More experiments to be performed
  - Comparison with commercial FPGA I/O count
  - TSV design constraints

Thank you for your attention

References

[Alt16] https://www.altera.com/products/fpga/cyclone-series/cyclone-iii/features.html (July 2016)

[Lem12] F. Lemonnier, P. Millet, G. M. Almeida, M. Hubner, J. Becker, S. Pille- ment, O. Sentieys, M. Koedam, S. Sinha, K. Goossens, C. Piguet, M. N. Morgan, and R. Lemaire, "Towards future adaptive multiprocessor systems-on-chip: An innovative approach for flexible architectures," in *International Conference on Embedded Computers*, 2012, pp. 228–235.

[Luu14] J. Luu, J. Goeders, M. Wainberg, A. Somerville, T. Yu, K. Nasartschuk, M. Nasr, S. Wang, T. Liu, N. Ahmed, K. B. Kent, J. Anderson, J. Rose, and V. Betz, "VTR 7.0: Next Generation Architecture and CAD System for FPGAs," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 7, no. 2, pp. 6:1–6:30, June 2014.

[Xil16] Xilinx, DS890, *UltraScale Architecture and Product Overview*, v2.8