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 ★
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

Altera Cyclone III floorplan [Alt16]
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

![Diagram of interface implementations](image)
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

\[ W = 32 \]

Full-interface area

\[ 5,565 \text{ µm}^2 \]

TSV area (for each interface signal)

\[ 76 \times 196 \text{ µm}^2 \]

Equivalent Stratix IV LAB area

\[ 20,461 \text{ µm}^2 \times 4 \approx 5,088 \text{ µm}^2 \]
Interface modeling in Quartus (2)

- Dummy LABs arranged contiguously in columns
- Interface columns reserved every $R$ columns in Stratix IV
Experimental methodology

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

- 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**

<table>
<thead>
<tr>
<th>RW</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>0.923</td>
<td>0.911</td>
<td>0.908</td>
<td>0.911</td>
</tr>
<tr>
<td>64</td>
<td>0.954</td>
<td>0.939</td>
<td>0.940</td>
<td>0.940</td>
</tr>
<tr>
<td>128</td>
<td>1.065</td>
<td>1.100</td>
<td>1.104</td>
<td>1.093</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>RW</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>1.002</td>
<td>1.008</td>
<td>1.003</td>
<td>1.000</td>
</tr>
<tr>
<td>64</td>
<td>1.002</td>
<td>0.991</td>
<td>0.987</td>
<td>0.997</td>
</tr>
<tr>
<td>128</td>
<td>0.999</td>
<td>0.992</td>
<td>0.982</td>
<td>0.995</td>
</tr>
</tbody>
</table>

**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

<table>
<thead>
<tr>
<th>R W</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>0.979</td>
<td>1.003</td>
<td>0.986</td>
<td>0.983</td>
</tr>
<tr>
<td>64</td>
<td>1.019</td>
<td>1.005</td>
<td>1.025</td>
<td>1.021</td>
</tr>
<tr>
<td>128</td>
<td>1.004</td>
<td>0.998</td>
<td>1.025</td>
<td>1.034</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R W</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>1.019</td>
<td>1.011</td>
<td>0.995</td>
<td>0.994</td>
</tr>
<tr>
<td>64</td>
<td>1.010</td>
<td>1.013</td>
<td>0.998</td>
<td>1.012</td>
</tr>
<tr>
<td>128</td>
<td>1.014</td>
<td>1.024</td>
<td>1.010</td>
<td>1.010</td>
</tr>
</tbody>
</table>

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

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

<table>
<thead>
<tr>
<th>W</th>
<th>Memories</th>
<th>LABs</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>11.87</td>
<td>33.33</td>
</tr>
<tr>
<td>64</td>
<td>12.80</td>
<td>25.67</td>
</tr>
<tr>
<td>128</td>
<td>15.47</td>
<td>26.07</td>
</tr>
</tbody>
</table>

Average amount of additional resources required for the I/O-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

<table>
<thead>
<tr>
<th>Circuit</th>
<th>Std. arch. $F_{\text{max}}$ (MHz)</th>
<th>Full interface arch. $F_{\text{max}}$ (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgm</td>
<td>81.17</td>
<td>76.48</td>
</tr>
<tr>
<td>blob_merge</td>
<td>103.75</td>
<td>108.71</td>
</tr>
<tr>
<td>mcml</td>
<td>35.73</td>
<td>35.78</td>
</tr>
<tr>
<td>stereovision1</td>
<td>136.93</td>
<td>130.36</td>
</tr>
<tr>
<td>stereovision2</td>
<td>113.95</td>
<td>125.08</td>
</tr>
</tbody>
</table>

*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 $F_{\text{max}}$
- **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


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