Introducing

A methodology and tool framework for supporting rapid exploration of memory hierarchies in FPGAs

H. Sidiropoulos, K. Siozios and D. Soudris
School of Electrical and Computer Engineering
National Technical University of Athens, Greece
Presentation outline

- Current state in academic tools
- Limitations of existing approaches
- NAROUTO framework
- Experimental results
- Conclusions
Overview of Presentation

Research

Motivation
Motivation

- FPGAs are composed by numerous macro-blocks
  - e.g. memories, DSP cores, embedded CPUs
- Architecture-level exploration is an important task
Motivation

- FPGAs are composed by numerous macro-blocks
  - e.g. memories, DSP cores, embedded CPUs
- Architecture-level exploration is an important task
- Academic frameworks provided limited support to heterogeneous blocks
  - They support only devices consisted of glue logic (CLBs) and routing resources
  - There is no support for evaluating designs in terms of power/energy dissipation
- Commercial frameworks support heterogeneity and power estimation
  - They allow only a small degree of freedom
Proposed Solution: NAROUTO Framework

- NAROUTO supports architecture-level exploration, as well as power estimation for FPGAs
  - Different memory organizations can be explored
  - It is the only public available framework that allows exploration of heterogeneous FPGAs in terms of power/energy consumption
    http://proteas.microlab.ntua.gr/narouto

- NAROUTO is an extension of 2D MEANDER toolflow
NAROUTO Framework

...on the road of technical domain
NAROUTO framework consists of a number of new open-source CAD tools.
NAROUTO framework consists of a number of new open-source CAD tools.
Experimental Results

. . . on supporting the proposed methodology
# Qualitative comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>NAROUTO</th>
<th>[4], [11]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Different types of macro-blocks</td>
<td>Unlimited</td>
<td>1</td>
</tr>
<tr>
<td>Realistic number of macro-blocks</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Realistic number of I/Os per macro-blocks</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Power estimation</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Part of complete framework</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Open source</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Qualitative comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>NAROUTO</th>
<th>[4], [11]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Different types of macro-blocks</td>
<td>Unlimited</td>
<td>1</td>
</tr>
<tr>
<td>Realistic number of macro-blocks</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Realistic number of I/Os per macro-blocks</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Power estimation</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Part of complete framework</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Open source</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

- NAROUTO handles designs with multiple macro blocks
- The power/energy estimation is incremental to existing solutions
### Experimental setup

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Existing [4], [11]</th>
<th>SP</th>
<th>FP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td># of BBs</td>
<td>Size of BBs</td>
<td># of BBs</td>
</tr>
<tr>
<td>oc_aes_core_inv</td>
<td>128</td>
<td>1×34,176</td>
<td>1</td>
</tr>
<tr>
<td>oc_ata_ocidec3</td>
<td>32</td>
<td>1×224</td>
<td>1</td>
</tr>
<tr>
<td>oc_hdlc</td>
<td>16</td>
<td>2×1,024</td>
<td>1</td>
</tr>
<tr>
<td>oc_minirisc</td>
<td>8</td>
<td>2×1,024</td>
<td>1</td>
</tr>
<tr>
<td>oc_oc8051</td>
<td>67</td>
<td>3×486</td>
<td>1×678</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1×972</td>
<td>1×678</td>
</tr>
<tr>
<td>os_blowfish</td>
<td>160</td>
<td>5×13,434</td>
<td>1</td>
</tr>
<tr>
<td><strong>Average:</strong></td>
<td>68.5</td>
<td>5.17</td>
<td>18,282</td>
</tr>
</tbody>
</table>

Both commercial and academic existing frameworks cannot handle efficiently designs with macro-blocks.
Exploration of memory blocks

- Different organizations and floorplans could be evaluated
- Apart from these solutions, any other designer-specific architecture could also be evaluated
Results about topology selection

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Max. Frequency (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(B)</td>
</tr>
<tr>
<td>oc_aes_core_inv</td>
<td>17.5</td>
</tr>
<tr>
<td>oc_ata_ocidec3</td>
<td>22.4</td>
</tr>
<tr>
<td>oc_hdlc</td>
<td>41.7</td>
</tr>
<tr>
<td>oc_minirisc</td>
<td>19.8</td>
</tr>
<tr>
<td>oc_oc8051</td>
<td>9.7</td>
</tr>
<tr>
<td>os_blowfish</td>
<td>14.9</td>
</tr>
<tr>
<td><strong>Average:</strong></td>
<td><strong>21.0</strong></td>
</tr>
</tbody>
</table>

- Uniform distribution of memories leads to:
  - higher operation frequencies (smaller delay)

---

FPL 2011  5 SEPT. 2011, Greece
### Results about topology selection

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Max. Frequency (MHz)</th>
<th>Power Consump. (mWatt)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(B)</td>
<td>(C)</td>
</tr>
<tr>
<td>oc_aes_core_inv</td>
<td>17.5</td>
<td>17.6</td>
</tr>
<tr>
<td>oc_ata_ocidec3</td>
<td>22.4</td>
<td>28.6</td>
</tr>
<tr>
<td>oc_hdlc</td>
<td>41.7</td>
<td>30.6</td>
</tr>
<tr>
<td>oc_minirisc</td>
<td>19.8</td>
<td>21.2</td>
</tr>
<tr>
<td>oc_oc8051</td>
<td>9.7</td>
<td>5.9</td>
</tr>
<tr>
<td>os_blowfish</td>
<td>14.9</td>
<td>14.8</td>
</tr>
<tr>
<td><strong>Average:</strong></td>
<td>21.0</td>
<td>19.8</td>
</tr>
</tbody>
</table>

- Uniform distribution of memories leads to:
  - higher operation frequencies (smaller delay)
  - higher power consumption
## Results about topology selection

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Max. Frequency (MHz)</th>
<th>Power Consump. (mWatt)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(B)</td>
<td>(C)</td>
</tr>
<tr>
<td>oc_aes_core_inv</td>
<td>17.5</td>
<td>17.6</td>
</tr>
<tr>
<td>oc_ata_ocidec3</td>
<td>22.4</td>
<td>28.6</td>
</tr>
<tr>
<td>oc_hdlc</td>
<td>41.7</td>
<td>30.6</td>
</tr>
<tr>
<td>oc_minirisc</td>
<td>19.8</td>
<td>21.2</td>
</tr>
<tr>
<td>oc_oc8051</td>
<td>9.7</td>
<td>5.9</td>
</tr>
<tr>
<td>os_blowfish</td>
<td>14.9</td>
<td>14.8</td>
</tr>
<tr>
<td><strong>Average:</strong></td>
<td><strong>21.0</strong></td>
<td><strong>19.8</strong></td>
</tr>
</tbody>
</table>

- Uniform distribution of memories leads to:
  - higher operation frequencies (smaller delay)
  - higher power consumption
- Whenever memories are floorplanned at the center of the FPGA, then it results to a power-aware architecture
## Results about topology selection

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Energy×Delay Product (×10^{-6})</th>
<th>(Border)</th>
<th>(Center)</th>
<th>(Uniform)</th>
</tr>
</thead>
<tbody>
<tr>
<td>oc_aes_core_inv</td>
<td></td>
<td>6.574</td>
<td>6.045</td>
<td>6.850</td>
</tr>
<tr>
<td>oc_ata_ocidec3</td>
<td></td>
<td>1.108</td>
<td>1.525</td>
<td>1.921</td>
</tr>
<tr>
<td>oc_hdlc</td>
<td></td>
<td>4.152</td>
<td>2.351</td>
<td>5.435</td>
</tr>
<tr>
<td>oc_minirisc</td>
<td></td>
<td>0.677</td>
<td>0.742</td>
<td>0.782</td>
</tr>
<tr>
<td>os_blowfish</td>
<td></td>
<td>4.885</td>
<td>4.210</td>
<td>5.849</td>
</tr>
<tr>
<td><strong>Average:</strong></td>
<td></td>
<td><strong>3.917</strong></td>
<td><strong>3.034</strong></td>
<td><strong>4.511</strong></td>
</tr>
<tr>
<td><strong>Ratio:</strong></td>
<td></td>
<td><strong>0.87</strong></td>
<td><strong>0.67</strong></td>
<td><strong>1.00</strong></td>
</tr>
</tbody>
</table>

- Whenever memories are assigned in the center of the device, it leads to smaller EDP value
- up to 33% EDP improvement
## Comparison between SP vs. FP

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Existing [4], [11]</th>
<th># of BBs</th>
<th>SP</th>
<th>FP</th>
</tr>
</thead>
<tbody>
<tr>
<td>oc_aes_core_inv</td>
<td>128</td>
<td>1</td>
<td>1×34,176</td>
<td>1</td>
</tr>
<tr>
<td>oc_ata_ocidec3</td>
<td>32</td>
<td>1</td>
<td>1×224</td>
<td>1</td>
</tr>
<tr>
<td>oc_hdlc</td>
<td>16</td>
<td>2</td>
<td>2×1,024</td>
<td>1</td>
</tr>
<tr>
<td>oc_minirisc</td>
<td>8</td>
<td>1</td>
<td>2×1,024</td>
<td>1</td>
</tr>
<tr>
<td>oc_oc8051</td>
<td>67</td>
<td>21</td>
<td>3×486</td>
<td>16×60</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1×972</td>
<td>1×678</td>
</tr>
<tr>
<td>os_blowfish</td>
<td>160</td>
<td>5</td>
<td>5×13,434</td>
<td>1</td>
</tr>
</tbody>
</table>

**Average:** 68.5 5.17 18,282 1
Evaluation in term of delay

SP and FP lead to average delay reduction 46%, as compared to existing frameworks [4, 11]
Evaluation in term of power consumption

Whenever memory blocks are assigned to the center, both SP and FP achieve average power savings 82%
Conclusions

- NAROUTO framework was introduced
- It supports architecture-level exploration
- The framework is public available [http://proteas.microlab.ntua.gr/naruto](http://proteas.microlab.ntua.gr/naruto)
- The introduced solution enables:
  - Designs with IP cores
  - Evaluation in terms of power/energy consumption
- Experimental results shown average gains in terms of delay and power consumption about 46% and 82%, respectively

Future directions:

Extend this framework to 3-D architecture
Thank you

Additional info at ksiop@microlab.ntua.gr