NAROUTO: A Tool Framework for architecture-level exploration of 2D and 3D heterogeneous FPGA architectures

Harry Sidiropoulos, Kostas Siozios and Dimitrios Soudris
School of ECE, National Technical University of Athens, Greece
{harry, ksiop, dsoudris}@microlab.ntua.gr

SUMMARY

This work introduces a tool framework, named NAROUTO, that targets the lack of a complete academic framework for architecture level exploration of heterogeneous 2D and 3D FPGA architectures. NAROUTO supports evaluation of these architectures in terms of area, power and delay.

One of the upmost important tasks for designing an efficient FPGA device is the architecture-level exploration in order to find its blocks/components, as well as their optimal organization. This problem becomes even more important nowadays, due to increased complexity posed by additional (heterogeneous) IP blocks.

Another critical factor in FPGA architectures are interconnection structures which increasingly contribute more to the delay and power consumption. Three-dimensional (3-D) chip stacking is touted as the silver bullet technology that can keep Moore's momentum and will fuel the next wave of consumer electronics products. 3D-NAROUTO supports application mapping and architecture exploration of a novel 3-D FPGA architecture, consisted of layers dedicated to logic, memory and I/O resources.

Fig. 1 gives an abstract view of the 2-D and 3-D NAROUTO frameworks. The first step at NAROUTO framework deals with application's synthesis and technology mapping.

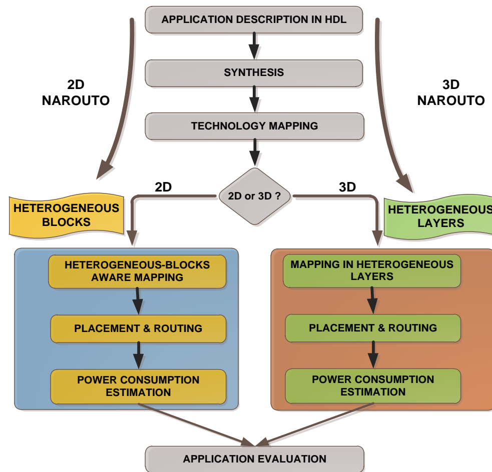

Figure 1: The NAROUTO (2D and 3D) Framework

For the next step of NAROUTO Framework an architecture description of the underlying FPGA is needed. If we choose a 2D FPGA, the application will be mapped into regular FPGA fabric and heterogeneous blocks (DSPs, RAM blocks, etc.). If a 3D FPGA is selected the application will be mapped onto layers dedicated to Logic, I/O, Hard blocks, etc.

Experimentation with different memory floor-plans, shown in Figure 2, where the memories are assigned to the borders of the device, to the center, and a scenario where memories are uniformly distributed over the FPGA lead to EDP reduction up to 33%, shown in Table 1.

Benchmark	Energy×Delay Product ($\times 10^{-6}$)		
	Border	Center	Uniform
oc_aes_core_inv	6.574	6.045	6.850
oc_ata_ocidec3	1.108	1.525	1.921
oc_hdlc	4.152	2.351	5.435
oc_minirisc	0.677	0.742	0.782
oc_oc8051	6.105	3.330	6.230
os_blowfish	4.885	4.210	5.849
Average:	**3.917**	**3.034**	**4.511**
Ratio:	**0.87**	**0.67**	**1.00**

Table 1: Exploration results for topology selection of memory blocks

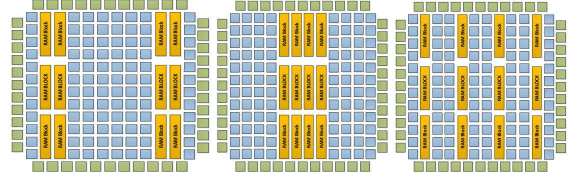

Figure 2: Different floor-plans for memory blocks: (a) placed in borders, (b) placed in center, and (c) uniformly distributed.

Furthermore we evaluated the efficiency of a 3-D FPGA, where logic and I/O blocks are assigned to different layers, and a 3-D FPGA, where logic, memory and I/O blocks are assigned to different layers, as depicted from Fig 3. With the first architecture (2 Layers) we achieved 22% gain at average, and with the second (3 Layers) 30% gain at average, in terms of delay under the same power budget as shown in Fig. 4.

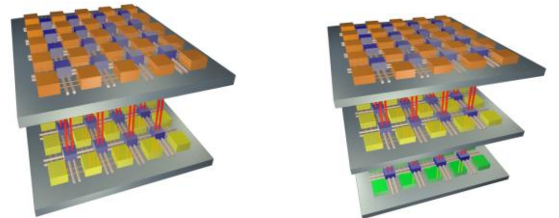

Figure 3: A 3D FPGA with: a layer of CLBs and one with I/Os (left), an extra layer of RAMs (right).

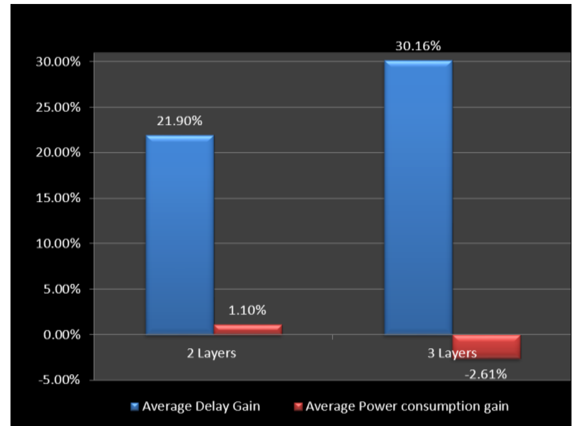

Figure 4. Experimental results in terms of Delay and Power consumption for a 3D FPGA with: a layer of CLBs and one with I/Os (left), an extra layer of RAMs (right).