A TOOL FOR EXPLORING HYBRID FPGAS

Chi Wai Yu

Department of Computing
Imperial College London, London, England
cyu@doc.ic.ac.uk

1. INTRODUCTION

Fine-grained field programmable gate arrays (FPGAs) have been modelled extensively using the versatile place and route (VPR) tool [1], which has been used to explore many architectural choices in FPGAs including switch boxes, interconnect, logic elements and routing. VPR is an open source place and route tool which uses an obsolete FPGA architecture based on the Xilinx 4000X series. However, modern devices include coarse grained elements such as memories, multipliers and processors, which serve to improve performance, power and density [2].

Since the original VPR framework does not model many modern features, researchers have modified VPR to include features present in modern FPGA. Wilton's HHVPR supports carry chains and embedded blocks such as BRAMs and multipliers. VPR for Hybrid FPGAs (VPH), the proposed tool, is based on HHVPR with the added features that are explained in the next section. Another approach is done by Beauchamp, who modified VPR to support carry chains, BRAMs, multipliers and floating point units [3].

Ho et. al. [4] developed the virtual embedded block (VEB) methodology to model arbitrary embedded blocks in existing commercial FPGA devices. This approach allows accurate evaluation of domain-specific hybrid FPGA. However that work is limited to existing devices only. We extend that work to a design exploration tool which uses the VEB approach within VPR to enable more flexible modelling of hybrid-FPGAs than in previous work.

To summarise, the major contributions of our work are:

- calibrating our tool with existing commercial tools to ensure consistent results,
- supporting coarse-grained block modelling by following the VEB method,
- adapting VPR to support hybrid FPGA architectures such as the Xilinx Virtex II.

2. VPH - VPR FOR HYBRID FPGA ARCHITECTURES

2.1. Requirements

In order to evaluate architectural tradeoffs, a modelling tool must capture the salient parameters. An optimal setting of parameters should be found to minimize a cost function that normally involves delay, area and power consumption.

VPH is based on HHVPR and is designed with the following requirements to capture the important parameters in domain-specific hybrid FPGA design:

- supporting the VEB methodology for hybrid FPGA architecture performance evaluation,
- modelling modern Xilinx and Altera devices. The work presented is based on the Xilinx Virtex II but can be generalised to other architectures,
- calibrating with existing vendor's tools. By changing the model parameters such as CLB, switch box and routing delay, we can tune our model to match commercial place and route tools such as Xilinx ISE,
- allowing exploration of different hybrid FPGA architectures.

2.2. Overview of Approach

A fine-grained element in an FPGA is flexible and small in size. However, implementation of complex logic functions with multiple fine-grained units are inefficient in terms of area, delay and power. Instead, less flexible coarse-grained units are usually more efficient for implementing complex functions such as floating point units, but waste area if unused. Therefore, embedding coarse-grained units in fine-grained fabric can result in improvements in speed, area and power consumption over a normal fine-grained FPGA for specific applications.

The virtual embedded block (VEB) methodology allows the evaluation of embedded coarse-grained elements on existing FPGA devices [4]. In this approach, the speed and area of each coarse-grained element are obtained from ASIC design tools and modelled by a group of logic cells in the

1-4244-1060-6/07/$25.00 ©2007 IEEE.

fine-grained fabric. A performance analysis of the hybrid FPGA can then be performed by following the standard synthesis, technology mapping and place and route design flow provided by the target FPGA vendor.

The vendor's performance analysis tools are based on existing FPGA devices. This is restrictive in the sense that it is not possible to target a non-existent FPGA device. For example, we are not able to model an FPGA with more logic resources than the largest existing FPGA. We use VPH to model a hybrid FPGA to combine the benefits of flexible architecture modelling via VPR with the VEB method to include embedded blocks.

3. IMPLEMENTATION OF VPH

VPH models the configurable logic block (CLB) in Virtex II FPGA. There are two slices in each CLB and each slice contains LUTs, flip flops, carry chains, shift registers, internal multiplexers and XOR gates. In addition, there are column based block RAMs and multipliers.

Figure 1 shows the VPH design flow. Applications are written in a high level hardware description language and synthesized to a netlist in EDIF using commercial tools such as Synplicity's Synplify Premier 8.5. From the EDIF file, Xilinx's ISE 8.1i *map* performs technology mapping. A native circuit description (NCD) file is generated during the mapping process. Xilinx ISE provides a program called *xdl* to convert binary NCD file to plain text xdl file which contains the configuration of each slice and I/O pad. We develop a conversion program *xdl2net* which converts the xdl file to a netlist for VPH.

We introduce user constraints on area and position for the logic blocks which are stored in user constraint file (ucf). With the constrains on VEBs in a specific position with a certain area group, we adopt the VEB approach. A link file is used to connect the ucf file to the associated module netlist. An architecture file specifies the FPGA's architectural parameters such as timing delay. At the end of the design flow, a placement and routing result file is produced and a timing analysis is generated.

4. RESULT

We calibrate VPH using Xilinx Virtex II FPGA. In the calibration, we tune the parameters in the architecture file manually with some simple test circuits, such as buffer chain and adder carry chain. We demonstrate the placement and routing of benchmark circuits [4] using VPH. The result shows that the timing analysis of the benchmark circuits obtained using the VPH tool is within 20% of what can be achieved by commercial tools.

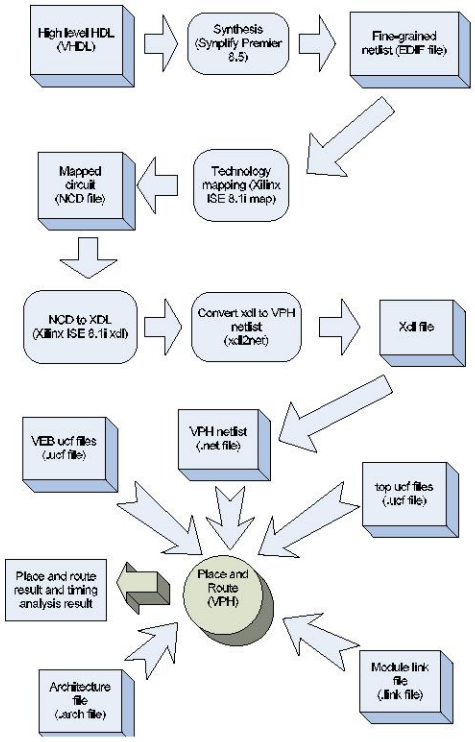

Figure 1: Design flow of VPH

5. CONCLUSION

We show that the VPH tool can accurately model a commercial FPGA on a set of benchmark problems. It is able to model heterogeneous embedded blocks in a hybrid FPGA and facilitates design exploration. This tool combines the benefits of both the VPR and the VEB. VPR allows a larger FPGA architecture design space to be evaluated than commercial tools, and VEB enables analysis of hybrid FPGAs. Current and future work includes extending VPH to cover other devices, using VPH to explore the performance of hybrid FPGAs in various architectures, and developing automated algorithms to optimize the hybrid FPGAs for domain specific applications.

Acknowledgement. The author gratefully acknowledges the support of the UK EPSRC (grant EP/D060567/1).

References

[1] V. Betz and J. Rose, "VPR: A New Packing, Placement and Routing Tool for FPGA Research," in *Proc. FPL*, 1997, pp. 213–222.

[2] J. Rose, "Hard vs. soft: The central question of pre-fabricated silicon," in *34th International Symposium on Multiple-Valued Logic (ISMVL'04)*, May 2004, pp. 2–5.

[3] M. Beauchamp, S. Hauck, K. Underwood, and K. Hemmert, "Embedded floating-point units in FPGAs," in *Proc. FPGA*, 2006, pp. 12–20.

[4] C.H. Ho, P.H.W. Leong, W. Luk, S. Wilton and S. Lopez-Buedo, "Virtual Embedded Blocks: A Methodology for Evaluating Embedded Elements in FPGAs," in *Proc. FCCM*, 2006, pp. 35–44.