Visual Exploration of Changing FPGA Architectures in the VTR Project

Konstantin Nasartschuk / Kenneth B. Kent / Rainer Herpers
University of New Brunswick
Faculty of Computer Science
Bonn-Rhine-Sieg University of Applied Sciences
kons.na@unb.ca ken@unb.ca rainer.herpers@h-brs.de

Outline

- Visualization framework for the workflow used in the VTR project
- Exploration support for FPGA architecture research
- Exploration of changes made to the netlist in the CAD flow
- Visual simulation of resulting circuits

Motivation

The VTR project is a collaboration of three tools which provide a CAD flow for FPGA architecture research. Each of the tools optimizes the hardware description according to the resources available on the target FPGA device. Sub results of the tool are stored using the Berkeley Logic Interchange Format (BLIF). The format is not created to be human readable but to contain all information necessary to reconstruct the netlist. Odin II, the first stage of the CAD flow, provides visualization software which presents the circuits created as a visual graph. The BLIF explorer is able to visualize a circuit which is described by a BLIF file. Comfort functions such as search, highlight, remove, rearrange are included.
The goal of this project was to extend the visualization software to support the visualization and simulation of output files produced by later tools in the workflow. In addition, the usability functions of the visualization were extended and multiple clock simulation support established.

Background

The VTR project consists of three different tools. Odin II compiles a Verilog source file and creates a hardware circuit. This circuit is optimized using an architecture description file which represents the specific abilities of a FPGA device. ABC performs logic optimization on the circuit. The VPR tool is then used to pack the logic blocks, place them on the FPGA device and connect them according to the structure in an optimized way.

Problem

The project was divided into three parts:

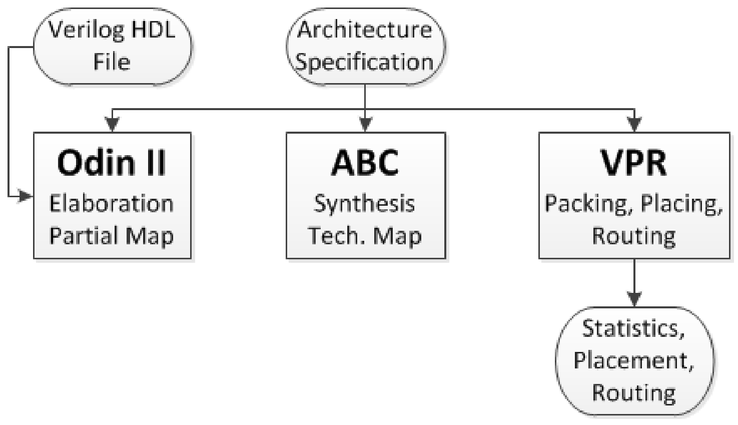
1. The output BLIF files created by the tools are following the BLIF specification but still contain differences. An example is a specific naming convention used only in Odin II while producing or reading a BLIF file.
2. As netlists grow larger seeing an overview of the structure and to examine specific sub parts of a graph becomes almost impossible.
3. The simulation component of Odin II only supports netlists with one clock. Being able to simulate circuits with multiple clocks expands the set of possible circuits for experiments.

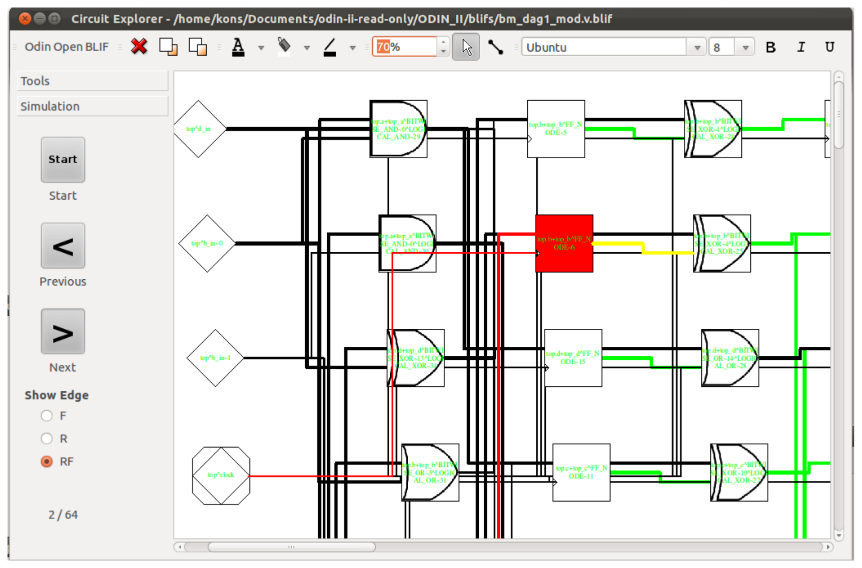
Challenges

- Odin II is written in C. The explorer is written in C++. New functional features developed in the visualization have to be developed on the C and the C++ side implementing a communication interface.
- The reading process of Odin II had to be changed, revalidated and tested.
- The simulation process was based on one single clock which determined the cycles. Having multiple clocks required to make changes to this structure in order to compute each possible clock edge.

Solution

- A new read in BLIF flow which ignores naming and creates nodes based on their truth table and template matching algorithms.
- The ability to show only sub parts of visual graphs while the complete structure is still considered during visual simulation was implemented in order to provide a powerful debugging and functional evaluation tool
- A new communication protocol between the visualization and Odin II was created in order to control the simulation of netlists with multiple clocks. The user is able to define clock ratios and the simulation computes all possible clock edges which need to be computed.


Verilog HDL File
Architecture Specification
Odin II
Elaboration
Partial Map
ABC
Synthesis
Tech. Map
VPR
Packing, Placing, Routing
Statistics, Placement, Routing



Result

The current version of Odin II provides exploration functionality which is used by developers in the CAD flow. The ability to visualize, simulate and inspect areas of interest allows developers to verify their research of new FPGA architectures. New structures can be examined and simulated in the visualization. The ability to examine the circuit after ABC and VPR allows one to track back changes and to verify the correctness of the specific tools. Multiple clock simulation expands the set of possible circuits which can be used for experiments in the VTR flow.

UNIVERSITY OF NEW BRUNSWICK
Reconfigurable Computing Research Group
FACULTY OF COMPUTER SCIENCE