Adapting and Automating XILINX's Partial Reconfiguration Flow for Multiple Module Implementations

Rainer Scholz

Universität der Bundeswehr München, Germany
rainer.scholz@unibw.de

Abstract. In this paper, we present a modification of XILINX's Partial Reconfiguration Design Flow. Starting with either HDL-Design files or synthesised netlists, the presented flow generates all partial as well as the complete configuration bitstreams. In contrast to the established XILINX design flows, our flow is completely automated by a generator. By checking partial reconfiguration constraints it assists the user to avoid typical errors in module and bus macro placement. Compared with the PlanAhead partial reconfiguration flow, it is a single flow for generating multiple implementation for each reconfigurable area.

1 Introduction

Dynamic Partial Reconfiguration (DPR) has been widely proposed as revolutionary computing paradigm over at least a decade. But until now, it was mainly an academic research topic and didn't find it's way to many real-life products. Some quite promising prototypes for productive systems have been created, but none of these approaches has lead to a commercial break-through.

Some of the reasons for this, is the complicated building process for systems using DPR and the restrictive requisitions for designing them.

The following section describes the solutions XILINX makes with a special focus on the partial reconfiguration flow generated by the tool PlanAhead. In Section 3 we introduce our adaptation and extension of the baseline XILINX flow. Section 4 presents a tool for automating the modified flow. The paper concludes with a summary including plans for future work and code release.

2 XILINX and Partial Reconfiguration

In [5] XILINX presents the flow for building reconfigurable designs based on their Modular Design [3, Chapter 4]. This flow allows the reconfiguration of entire columns of Configurable Logic Blocks (CLB) and does not support static routes through reconfigurable areas.

In a new version of the baseline flow target [6] reconfigurable modules may span any rectangular area of an FPGA and routing static logic through reconfigurable modules is possible. However users are still limited, because tests on

P.C. Diniz et al. (Eds.): ARC 2007, LNCS 4419, pp. 122–129, 2007.