A Direct Mapping System for Datapath Module and FSM Implementation into LUT-Based FPGAs

Joerg Abke and Erich Barke
Institute of Microelectronic Circuits and Systems,
University of Hannover, Appelstr. 4, D-30167 Hannover, Germany
abke@ieee.org, barke@ims.uni-hannover.de

Abstract

Today's high capacity Field-Programmable Gate Arrays (FPGAs) and the upcoming trend to System-On-Programmable-Chip (SOPC) require novel implementation strategies. These have to overcome long implementation times of traditional synthesis approaches. In this poster, a unique approach for technology mapping of both datapath modules and controller descriptions into Look-Up Table (LUT)-based FPGAs is presented. The proposed method starts at Register-Transfer-Level (RTL) and follows the Library of Parameterized Modules (LPM) standard. The mapping environment includes an implicit state minimization algorithm for FSMs.

1 Direct Mapping System Overview

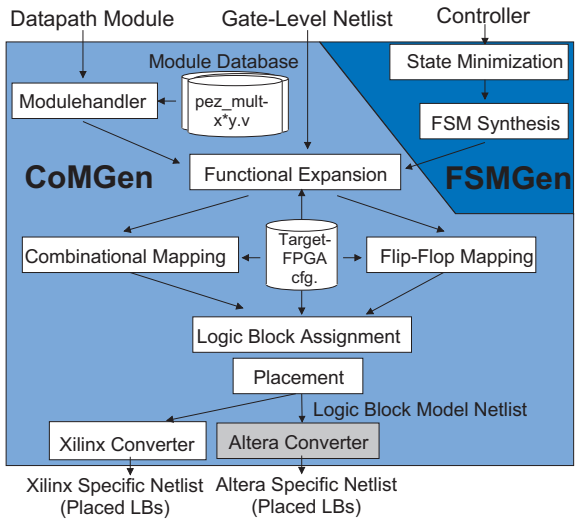

Figure 1. Direct Mapping System Overview

Figure 1 gives an overview of the mapping environment. Datapath modules or gate-level circuit descriptions are mapped by the Configurable Module Generator (CoMGen) [1]. The logic block placement is accomplished by a dedicated placement method for direct mapping [2]. It takes the structure of specified and parameterizable RT-modules into account. The FSM description is given as a state transition graph (STG). Its states are minimized, encoded and transferred to a structural description, which is also mapped by CoMGen.

In contrast to traditional technology mapping methods, CoMGen stores structural information of the module in the module database. This information is used for speeding up the datapath module implementation significantly without sacrificing implementation quality [2]. The component hierarchy or preassigned cuts are used for module decomposition. Furthermore, FPGA specific features (i.e. Fast-Carry chains) are supported by CoMGen.

A dedicated multiplier module generator yields very good results for very regular components. However, CoMGen implements routable and less ressource consuming implementations for irregular module structures like the Booth multiplier. Despite its general approach CoMGen implements regular components as good as a dedicated generation method. Also, arbitrary circuit descriptions are implemented by CoMGen very efficiently as benchmark results show [1].

References

[1] J. Abke and E. Barke. CoMGen: Direct Mapping of Arbitrary Components into LUT-Based FPGAs. *Int'l. Conf. on Field-Programmable Logic and Applications (FPL)*, pages 191–200, 2000.

[2] J. Abke and E. Barke. A New Placement Method for Direct Mapping into LUT-Based FPGAs. *Int'l. Conf. on Field-Programmable Logic and Applications (FPL)*, pages 27–36, 2001.

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 $17.00 © 2002 **IEEE**