ABSTRACT

This paper presents an educational Electronic Design Automation tool called AYSE (Automated SYstem Synthesis Environment). AYSE is developed to help undergraduate and graduate students better understand and implement fundamental concepts in digital electronic design and synthesis courses. It receives an intermediate format that represents the system behavior and a set of constraints as input; and generates the representation of the actual circuit using the electronic components (functional units, memory and steering logic components) available in its technology library. A user can load the input from an external file or utilize the graphical user interface (GUI). Similarly, the output can be displayed using GUI or written to a file. The user is able to interact with the tool during the synthesis process and select the algorithms to perform three main synthesis tasks; namely, resource allocation, operation scheduling and resource binding/sharing. It is also possible to run the tool for the most efficient solution. The effectiveness of the tool was tested in a graduate level course.

Keywords: Computer Aided Design, Electronic Design Automation, Graphic User Interface (GUI), High Level Synthesis, Optimization

INTRODUCTION

Digital electronic systems combine particular logic and circuit design techniques required to implement Integrated Circuits (ICs). Such circuits consist of miniaturized electronic components built into an electrical network on a semiconductor board. Digital IC design produces components such as microprocessors, Field Programmable Gate Arrays (FPGAs), memories (RAM, ROM, and flash), Application Specific Integrated Circuits (ASICs) (Coussy & Takach, 2009). Digital system design puts emphasis on maximizing performance, reducing power/energy consumption, improving memory space consumption, verifying logical/functional correctness, and maximizing circuit density.

Digital system synthesis is a process in which an abstract form of desired circuit behavior is turned into a design implementation in terms of the components available in a technology library at a given abstraction level (Makris, & Orailoglu, 1999). Circuit behavior is typically given using a Hardware Description Language (HDL) such as VHDL, Verilog or SystemC (Sun, 1994). After analyzing the
behavioral source code, it is translated into an intermediate format. Various Electronic Design Automation (EDA) tools use this intermediate format as input in order to generate the final circuit that meets all design constraints (Martin, & Smith, 2009). Three main tasks in High Level Synthesis (HLS) are resource allocation, operation scheduling, and resource sharing/binding. In resource allocation phase, the number and type of resources that are utilized to implement the final circuit are determined. This information can be also given as a design constraint (i.e., the number and type of resources are specifically given). In scheduling phase, the start times of the operations are determined using various algorithms. Given that the time (or the number of clock cycles) needed to execute each operation is known; the clock cycles in which each operation should execute are calculated. In resource sharing/binding phase, the operations are assigned to available functional units. In addition, register allocation (or memory mapping) is done in this phase. After that, in interface synthesis, the required data and control signals are generated. An EDA tool not only has to meet all design constraints but also aims at improving design metrics as much as possible. Among others, performance, area, power/energy consumption, and memory efficiency are important design metrics.

Raising the abstraction level in the synthesis process brings many advantages including reduced design time, less probability of design errors, and reduced complexity (Sarkar, Dabral, Tiwari, & Mitra, 2009). Due to these advantages, High Level Synthesis (HLS) has become increasingly popular in EDA field. In addition, HLS tools can reduce the verification time (which is a major contributor in overall design cycle) and can optimize the final circuit and creates opportunities for extensive design space exploration (Duranton, Yehia, De Sutter, De Bosschere, Cohen, Falsafi, Gaydadjiev, Katevenis, Maebe, Munk, Navarro, Ramirez, Temam, & Valero, 2009).

Considering the complexity of today’s digital electronic systems, automating the synthesis process is very crucial. Various EDA vendors such as Mentor Graphics, Xilinx, Synopsis, and Cadence provide elegant and complex tools that perform digital system synthesis (Meeus, Beeck, Goedemé, Meel, & Stroobandt, 2012; Tosun, Mansouri, Arvas, Kandemir, Xie, & Hung, 2005). HLS optimization techniques deliver high quality results for digital designers while enabling rapid exploration of performance, throughput, power, reliability, and area tradeoffs. HLS tools aim at helping industry improve quality of results, simplify design flows, establish interoperability, and improve embedded processing flows. Therefore, designers could significantly increase their productivity by using high level synthesis tools during the design process.

On the other hand, many educators in the areas of computer engineering/science and electrical/ electronic engineering fields find it quite challenging to teach how to operate these complex tools, and then, use them in a one-semester course. The instructor sometimes needs to spend a lot of time to teach the details of complex tools instead of focusing on and teaching fundamental concepts in class. Also, as the source codes of those tools are not provided, it is impossible to modify those tools to meet the demands of the course being taught. In addition, these tools may not be very suitable to expose the intermediate steps during the synthesis process. They typically target at the best design practice. Hence, showing the examples of different possible implementations may not be possible. As a result, even though the elegant and complex electronic design tools are good at generating the best design alternatives for the industrial applications, they may not be best for teaching purposes.

Considering the facts given in the previous paragraph, in this work, we present an educational Electronic Design Automation tool to help undergraduate and graduate students better understand and implement fundamental concepts in digital design and synthesis courses. The tool receives an intermediate format that represents the system behavior and a set of constraints as input; and generates the representation of the circuit using the functional units available in a
Related Content

Uncovering Relationships Between Emotional States and Higher-Order Needs: Enhancing Consumer Emotional Experiences in Computer-Mediated Environment
Andrew Pressey, Laura Salciuviene and Stuart Barnes (2013). International Journal of Online Marketing (pp. 31-46).
www.igi-global.com/article/uncovering-relationships-between-emotional-states/77331?camid=4v1a

Electronic Word of Mouth: Definitions and Concepts
www.igi-global.com/chapter/electronic-word-of-mouth/136715?camid=4v1a
E-Memory Choice Architecture: Modeling the Use Diffusion of Twitter Archiving System
www.igi-global.com/article/e-memory-choice-architecture/218179?camid=4v1a

The School Walls Teach: Student Involvement in the Green School
www.igi-global.com/chapter/the-school-walls-teach/115187?camid=4v1a