Rapid prototyping methodology in ASIC Design Education

Vassiliy Tchoumatchenko, Tania Vassileva
Department of Electronics, Technical University –Sofia
vpt@vmei.acad.bg

Abstract. The paper describes our experience in integrating commercial quality CAD tools from Xilinx, Synopsys and Aldec in an undergraduate teaching environment where the students go through design process from high-level specification until a real hardware is implemented and tested. The educational benefits of using rapid prototyping methodology as "learning by doing" approach in ASICs education are considered.

Introduction

One of the main goals of the engineering education is to help the students to understand the most up-to-date technologies together with the methods and tools they will use in their future industrial positions.

Changing needs of industry, advances in technology and design methodology has required a significant reorganisation of education in Application Specific Integrated Circuit (ASIC) design with the emphasis on obtaining complete hands-on experience from design specification through to hardware verification.

With the emergence of Field Programmable Gate Array (FPGA) technologies, the use of gate array and standard cells approaches in education somehow loose interest: the same logic synthesis and place and route concepts are used in both approaches, and the FPGA avoids the high fabrication cost.

Competitive pressures are focusing industry to reduce product development cycles and bring out new products more and more frequently. During the last years, rapid prototyping on FPGA become one of the key technologies to help revolutionise the design process. This approach makes use of FPGAs as an ASIC emulator [1] that usually runs at a much lower speed.

This paper describes the educational benefits of using rapid prototyping methodology as "learning by doing" approach in ASICs education. This methodology places the emphasis on high-level design, which reduces time to market by relying on synthesis software and programmable logic to produce working prototypes rapidly.

Hardware Description Languages (HDLs) are becoming increasingly popular in design large-scale integrated circuits. They support the design at higher, more abstract levels. The use of such languages makes it possible to utilise top-down methods known from software engineering. Designing at a more abstract level frees the designer from implementation details required for designs at the logic level.

The papers examines the use of HDL, Programmable Logic Devices (PLDs) and FPGA in an undergraduate teaching environment where the students go through the logic design process from high-level behavioural specification toward logic synthesis and simulation until a real world circuit is programmed and tested. Training exercises, which fulfil the course objectives and a follow-on project, are also outlined.

2. Course Overview

The objective of the course is to teach students an ASIC design methodology based on Hardware Description Languages (HDL) and automatic synthesis. The principal goal of our ASIC Design Automation course is to convey system level electronics prototyping concepts, with an emphasis on using FPGAs to implement digital logic and processing functions. Design, manufacturing, and economics issues are addressed in a case study tradeoff analysis between semicustom ICs, Application Specific ICs (ASICs) and Application Specific Standard product (ASSPs) alternatives. Different design methodologies (top down, bottom up) are discussed.

The lecture component of the class introduces students to high-level system specification and simulation techniques as well as to synthesis and schematic capture alternatives for hardware design. The course concentrates on VHDL for synthesis so all the VHDL code examples have been written to be both simulable and synthesisable. With this approach the subset of VHDL that is synthesisable is covered first.

The course also introduces the concept of test benches to show that VHDL can be used to test by simulation a previously written synthesisable VHDL design. At this point additional VHDL statements and features are covered.

The introduction of behavioural modelling with VHDL has significantly affected the design process. The main pedagogical advantages of using VHDL as a teaching tool in logic design are:

- a methodological framework driving the students toward top down thinking and algorithmic description of a design;
- using a unified description language for designs targeted towards circuit families extending from simple PLDs to large FPGAs and ASICs.

Synthesis and test methodology part covers issues relating to design space exploration, taking different constraints into account in the design, and links to physical design environments (floorplanning, place and route). A vital advantage of VHDL is its device independent nature. The design's source code can be targeted to any technology without changes. Technology mapping and design migration to new technologies are also covered.

Laboratory exercises and projects reinforce the concepts introduced in the lectures and help students to get a working hands-on confidence. The laboratory component provides practical hands-on experience using VHDL, simulation, and synthesis tools, as well as digital hardware prototyping on PLDs, CPLDs and FPGAs.

3. Rapid prototyping methodology

FPGA devices are increasingly being used as prototypes for ASICs. This allows a design to be verified in a system environment before changing to an ASIC process. Prototype also permits drivers and software related activities to be initiated at earlier stages thus reducing significantly the design-to-market time. Potential bugs and system incompatibilities can be detected at a very early stage.

Our course emphasises design methodology, techniques and practical prototyping trade-offs (design time/cost-speed-power-area) as applied to the entire electronic system design hierarchy.

The rapid design prototyping methodology used in the design process is shown in Figure 1.

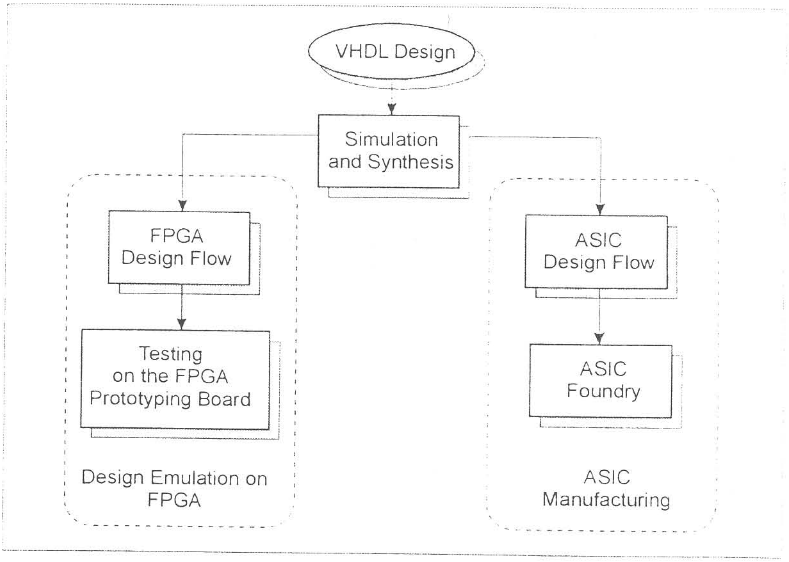

Figure 1. Rapid prototyping methodology

After creating a synthesisable VHDL model of a design, logic synthesis with automatic partitioning is used to develop a working prototype on Xilinx FPGA. Synopsys FPGA Express software capabilities is used for this reason.

The resulting schematic or netlist after synthesis are then implemented on the XC4003 device using commercial Xilinx's FPGA software tools Foundation. The automatic interconnection using FPGAs eliminates the need for any manual wiring on the prototype. This automatic interconnection feature makes this approach ideal for classroom. Automatic generation of the data file to produce the prototype requires different amount of workstation time for each design depending of its complexity. Design can be downloaded to the hardware in few minutes.

FPGAs are very attractive solution for designs where specifications are evolving and the time to market must be very small. But, when the specification has been finalised and high volume production is needed, then cost is a significant issue. The transition to an ASIC technology is essential to meet these requirements. These prototypes may then be altered as requirements change or convert into other application-specific integrated circuits (ASICs). This type of methodology is widely used in industry to prototype ASICs and recent microprocessor designs as Intel's Pentium and AMD's K5 processor chips.

In our course design migration is used to port already prototyped and functionally tested design to new technology thus reducing design risks and errors. Project is re-synthesised targeting new technology. Place, route and timing verification are obligatory. For standard cell designs we have used Cadence and Taner Ledit software tools. Problems, concerning design migration and mapping to new technology are considered in [2,3].

4. Experience

A major goal of the class is to provide students with a design experience using a range of techniques and methodologies for rapid prototyping of electronic systems.

The design flow is based on the Xilinx EDA toolset (Figure 2).

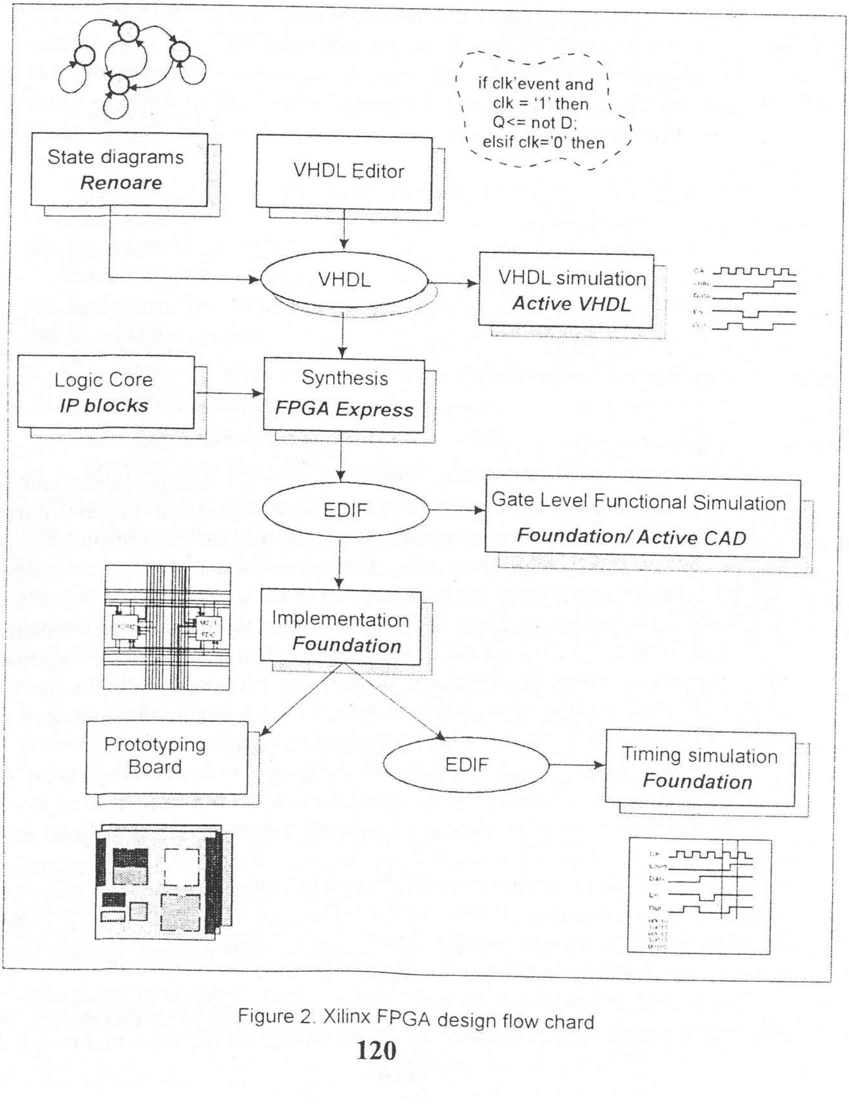

Figure 2. Xilinx FPGA design flow chard

Xilinx tools allow for designing digital logic circuits using either schematic capture or logic synthesis. Capabilities for targeting both FPGAs and PLDs are included as required to implement various portion of the project. Both functional simulations without timing and complete back annotated timing simulations are supported. Finally, the possibility to actually program the chosen FPGAs and CPLDs is provided.

Each of the individual laboratory exercises was designed to teach the students a specific portion of the design flow. The labs begin with a tutorial that takes the students through the required design flow with a very simple example. It gives the students detailed instructions for each step in the process and shows them how to use the Synopsys, ActiveCAD, Vsistem-V and Xilinx tools as well as what tool output they should expect.

The design of a simple traffic light controller is used to show the complete design process from text entry of the VHDL description to the place and route of the synthesised design into an FPGA. We use Vsistem-V tool to carry out a functional simulation on the design and then use FPGA Express to synthesise the design targeting Xilinx FPGA. After placing and routing using Foundation tool students download configuration file into Xilinx's development board with XC4003 chip so they can see the traffic light controller working.

Subsequent laboratory exercises require students to write their own VHDL descriptions on different designs starting with combinational logic type circuits to state machine designs. In this way the students learn the flow as applied to their own design. The labs are done individually by the students to ensure that each knows how to use the tools before they are organised into groups for their projects.

Projects are performed by two-person teams. A variety of projects have been undertaken. Some of them are reported in [4,5]. Generally, each contains a mixture of computation and control. Emphasis is placed on finishing all of the project design steps (synthesis, simulation, testing, prototyping) and not on the complexity of the project itself.

5. Laboratory Support

In keeping with the goal of providing students with exposure and experience with state-of-the-art CAD tools, we have joined the university programs of several CAD vendors. We integrate commercial quality CAE tools from Xilinx, Synopsys and Aldec in an undergraduate teaching environment where the students go through the logic design process from high-level specification until a real hardware is programmed and tested.

For prototyping of FPGAs we use custom-made PLD and FPGA boards [6], and demonstration boards supplied by Xilinx. All of these units contain reconfigurable logic so the equipment is available for reuse. We have two custom boards and several Xilinx boards but they are easily shared by all of the students in the class since they are accessible from any workplace.

With the proper CAD tools and equipment it is now possible for groups of undergraduate students to design, simulate, and develop working prototypes of complex systems as a part of their laboratory coursework. After testing properly working prototypes are migrated to ASICs targeting ES2 CMOS technology. Place and route for standard cells are performed with Cadence software, obtained through Euro Practice initiative of the European Community.

Conclusions

The paper reports a successful attempt that has been made to offer an undergraduate training in ASIC design principles at the Technical University of Sofia with the emphasis on obtaining complete hands-on experience from design specification through to hardware verification.

A VHDL based rapid prototyping approach to simulate, synthesise, and implement a design prototype using commercial CAD tools is described. The benefit of a rapid design prototyping and of being able to test out parts of the design as it proceeds is discussed.

The Xilinx, Synopsys and Aldec tools offer students the opportunity to work with commercial CAD software to reduce the "concept-to-system prototype" time, and to increase the probability of first pass design success. Students are strongly motivated creating *real circuit* not a computer running program.

As a result we believe that our students are able to graduate with not only a good theoretical background but they also are familiars with the types of tools being used by industry today.

References

1. I.Stamoulis, N.Ford, G.J.Dunnett, M.White, P.F.Lister, VHDL Methodologies for Effective Implementation on FPGA Devices and Subsequent Transition to ASIC Technology, Proc. of the *Design, Automation and Test in Europe DATE'98*, pp.165-170, February 23-26, 1998, Paris, France
2. T.Vassileva, V.Tchoumatchenko, Practical Experiance in Teaching Computer Aided Design of ASIC, *The European Conference on Design Automation with The European Event in ASIC Design - EUROASIC 1993*, pp. 206-209, 22 - 25 February, Paris, France
3. V. Tchoumatchenko, V.Zahariev, T. Vassileva, FPGA Implementation of a bit-serial Multiplier, Proc. of the *European Design and Test Conference, ED&TC95*, pp. 49-52, 6 - 9 March 1995, Paris, France
4. V. Tchoumatchenko, T.Vassileva, FPGA Design Portability, Proc. of *Workshop on Design Methodologies for Microelektronics*, pp. 326 – 330, September 11-13, 1995, Smolinice, Slovakia
5. T. Vassileva, V.Tchoumatchenko, R. P. Ribas, A. Guyot, FPGA Design Migration: Some Remarks FPL'96, *Sixth International Workshop on Field Programmable Logic and Application*, September 23-25, pp. 405-409, 1996, Darmstadt, Germany
6. T.Vassileva, V.Tchoumatchenko, FPGA as Educational ASIC, Proc. of the *1st European Workshop on Microelectronics Education*, pp. 225-228, 5-6 February 1996, Villard de Lans, Grenoble, World Scientific, France