Computer Architecture
Technical Committee

NEWSLETTER

A Technical Committee of the IEEE Computer Society

LANGUAGES

NETWORKS

ARCHITECTURE

SYSTEMS

IEEE Computer Society Technical Committee on Computer Architecture Newsletter

Table of Contents

February 1999

IEEE
COMPUTER
SOCIETY
http://computer.org

The Institute of Electrical &
Electronics Engineers Inc.

Integration of CAD Tools and Structured Design Principles in an Undergraduate CE Curriculum

D.L. Andrews,M.A. Thornton, Department of Computer Engineering, University of Arkansas
{dla,mat1}@engr.uark.edu

Abstract--**This paper describes the integration and usage of modern CAD tools and structured design principles into a computer engineering curriculum. This approach emphasizes design methodology using the tools rather than focusing solely on the tool usage.**

Key Terms
***Index Terms*--CAD/CAE, Structured Design, Computer Architecture, Digital Design**

I. Introduction

The Computer Engineering undergraduate curriculum at the University of Arkansas is incorporating computer aided engineering and design (CAE/CAD) packages into undergraduate courses to support the teaching of structured design principles. The intent of augmenting the curriculum with these packages is to enhance the students theoretical understanding of the material with hands on design and analysis experience in a structured design environment. Integrating CAD and CAE packages into courses allows the students to focus their efforts on developing a clean, efficient design instead of spending a large portion of their time engaged in drafting. Although design is emphasized in all digital design and computer architecture classes, we have integrated a junior level class that combines subsystem design with the teaching of structured design approaches. The structured design approach is explicitly stated to the students, and represents an intentional attempt to teach top down structured design.

II. Design Entry Using CAD Systems

Many universities now utilize HDLs in course-work. A common approach to HDL instruction is to introduce the language syntax with supporting design examples. Our approach differs from other universities in the following two aspects:

1. A simple text based design entry language for PLDs is presented before more complex HDLs such as VHDL and Verilog are discussed.

2. Instead of discussing HDL syntax first with supporting examples, the concepts of discrete simulation algorithms and techniques are presented with examples written in HDL.

In our sophomore introduction to digital design course, students are exposed to the architecture of simple programmable PLA, PAL, and PROM devices as a first introduction to programmable logic devices (PLDs). After the discussion of the internal architecture, problems are assigned requiring the students to generate fuse map information manually for simple designs as motivation for the use of a basic PLD based design system such as PALASM.

The second aspect of our approach involves describing simulation algorithms and techniques instead of presenting HDL syntax an examples. This approach seems to better retain the interest of the students since going over language syntax can be a very dry experience and require more mental memory effort than conceptual understanding. By surveying the techniques of discrete event circuit simulation, concepts regarding the analysis and physical characteristics are reinforced with a secondary benefit of introducing the use of a particular HDL.

This sequence culminates in the use of HDL's to specify and simulate moderately complex designs. The HDL is then used as input to a FPGA based synthesis tool enabling students to gain experience with modern design flows including concepts such as synthesis, back annotation, and verification.

III. Integration of Design Methodology

The principles of a structured design methodology are formalized in the junior level computer subsystem design course. This is the first level where the students are exposed to the design and analysis of moderately complex systems. The course is presented analogous to a structured top down complete design cycle. The course uses a series

Department of Computer Engineering, University of Arkansas, Fayetteville, AR

of laboratories, first starting with a requirements assessment, a top level design, and than the detailed design of the various subsystems of a fictitious single board computer flight control system. Each laboratory combines subsystems designed and tested in previous laboratories, in a structured top down design fashion.

Models of the chips used for this course are obtained from standard libraries associated with our CAD package (Mentor Graphics). Models are used for the Intel 8086, 8284A clock generator, RAM, ROM, 8255 Programmable Interface Chip, 8251 UART, and other I/O devices. These models include a complete behavioral specification for each chip. Simulation of these behavioral models allow complete system simulations to be run. The behavioral models for the memory chips include the capability of specifying small machine code programs for execution during the simulation. The behavioral model for the 8086 actually executes these programs during simulation, generating the correct handshake signals for data transfers, interrupt signals for external interrupts, etc.

The first laboratory focuses on a simple clocking and reset circuit for the 8086 microprocessor. This circuit is simple enough to allow the students to gain the confidence of entering a completely new design, taking the design from schematic entry through simulation. In the next laboratory, the students build up a memory system comprised of both RAM and ROM. Mentor allows the students to develop and implement the decode circuitry for the memory system, based on the actual timing requirements of the 8086. The simulations performed on their decode circuits and memories allows the students to blend the theoretical aspects of memory organization with the practical aspects of designing a fairly complex realizable system in accordance with actual system timing requirements. Subsequent laboratories continue to build on these laboratories, adding complexity to the design in a structured Socratic fashion.

The use of Mentor in this course reinforces the students understanding of the complexity and timing effects of propagation delays through decode circuitry. The schematic entry portion of Mentor using predefined library models allows the rapid development of a complete system comprised of fairly complex circuitry. Students are able to focus their efforts on developing a clean, efficient design instead of spending a large portion of their time engaged in drafting. The simulation portion of Mentor allows the students to quickly see the real effects of their design. This environment provides the students the ability to iterate their designs, much as is done in industry.

IV. Conclusions

An approach for the integration of modern CAD tools and structured design approaches into a computer engineering curriculum was described. By closely coordinating the introduction of CAD tool usage throughout the course sequence, unnecessary repetition of instruction regarding the mechanics of tool usage was described. Furthermore, we feel that emphasizing design methodology using the tools rather than focusing solely on their use allows students to learn how to use these tools without losing valuable engineering design instruction time. Finally, by introducing the use of HDLs by teaching simulation, specification generation, and automated synthesis techniques allows the instruction to retain more design content versus the more common approach of simply describing the language syntax and analyzing a number of design examples.

V. Bibliography

[1] Haggard, Roger L., *"Classroom Experiences and Student Attitudes toward Electronic Design Automation"*, Proceedings of the 25th Southeastern Symposium on System Theory, Los Alamitos Ca., IEEE Computer Society Press, 1993, pp. 411-415

[2] Andrews, D., A. Azemi, S. Charlton, and E. Yaz, *" Computer Simulation in Electrical Engineering Education"* Proceedings of the American Society of Engineering Education Conference , Baton Rouge, LA. March 24-25, 1994

[3] Nixon, M.S., *"On a Programmable Approach to Introducing Digital Design"*, IEEE Trans. Educ., vol. 40, pp. 195-206, 1997