Use of CAD Tools in the Integrated Computer Engineering Design (ICED) Curriculum

Augustus K. Uht
Department of Electrical and Computer Engineering
University of Rhode Island
Kelley Hall
4 East Alumni Ave.
Kingston, RI 02881-0805, USA

(401) 874-5431
uht@ele.uri.edu
www.ele.uri.edu/∼uht

Abstract

A new undergraduate computer engineering curriculum, ICED, is being introduced at the University of Rhode Island. The main feature of the curriculum is a design project spanning the last three years of the major. This gives continuity to a student's studies: they will always know why they are learning a particular topic, and how it fits into the big picture. It also introduces them to long-term projects, and the requisite good documentation and communication habits necessary for its completion.

The project to be undertaken is the design, simulation and construction of a computer and its compiler, including the design of its instruction set. Further, the various students' computers will be networked together during the final integration phase of the project. Students will learn to make hardware/software design tradeoffs, as well as get hands-on experience with hardware.

A key element of the design experience is the use of modern CAD tools. The Mentor Graphics CAD tool suite will be used throughout the curriculum. By standardizing on one set of tools, the time for the students to learn the tools is amortized over the entire curriculum.

The curriculum has just received funding from the National Science Foundation, and has been formally approved by the University. Students will start the curriculum in Fall, 1997.

The paper gives the rationale of ICED in depth, and describes the core courses, their activities and use of the Mentor tools, and how they interrelate to achieve the goal.

1 Introduction

The University of Rhode Island has developed a novel undergraduate Computer Engineering (CE) major curriculum. Traditional CE curricula are composed of disparate courses which poorly unify key themes and do not allow students to develop critical abilities. The Electrical and Computer Engineering (ELE) department at URI will remedy these ills with a new CE curriculum. The centerpiece of the curriculum will be the design of a richly complex digital

This work was supported in part by a software donation from the Mentor Graphics Corp., and in part by the National Science Foundation through grant number DUE-9751215.

system: a network of computers including, particularly, a computer with hardware and important system software built entirely by each student. This project will span the courses taken for the last 3 years of each undergraduate's studies.

This curriculum will allow a novel emphasis on fundamental engineering understanding, which can not be developed in projects of shorter duration. Students will develop an understanding of how design decisions play out over time, they will employ sophisticated design techniques using standard simulation and synthesis tools and they will make complex software/hardware tradeoffs.

This unified, longitudinal Integrated Computer Engineering Design (ICED) curriculum has been planned over several years. It has been accepted by the university.

This paper introduces the ICED curriculum and provides a broad view of the key use of Mentor tools in ICED. Details of the tools' use will be published later.

Curricular Need - The old version of the computer engineering curriculum lacked cohesion, as well as large-systems design, construction and evaluation. Also, there was not a large software component, and hardware and software were not linked in a meaningful way, especially to understand the tradeoffs between the two. Further, the emphasis on networks was not as great as it should be, especially with the explosive growth of the Internet.

The students need experience in complex digital system design, simulation, construction and evaluation. Although design these days is predominantly done with simulation, physical experience with complex hardware is essential for a computer engineer. "...there is something about late night debugging of that 'last' hardware problem that gives student engineers respect for the real difficulties of hardware, that simulations alone do not provide."[1]

Related Curricula - Several schools design and build computers as part of their CE curricula, e.g., Georgia Tech[3, 4], UCSC[9] and others[5]. However, such projects are usually done all at once, as capstone Senior design projects. A key element of ICED is a three-year long design project as part of most of the CE courses.

Some schools also involve compilers in the capstone projects. At Georgia Tech, this includes the modification of a retargetable C compiler to accommodate the student-designed computer. However, compiler construction in general is not incorporated in the experience; to our knowledge no school integrates a complete compiler design course with the computer hardware component of a CE curriculum. In fact, no school seems to require a compiler construction course as part of its CE curriculum[5]. Therefore, in such traditional curricula, the opportunity to make hardware/software design tradeoffs is relatively restricted, especially considering the large role such issues have in current computer (and most digital system) design. ICED requires a compiler construction course; it is integrated with the major project.

Another issue involves the design tools used in a curriculum. Although it is possible to construct one's own in-house CAD (Computer-Aided-Design) tools[1], keeping them simple (as compared to industrial tools) and thus easier to learn, the student then does not experience the full power of industrial tools, nor does she/he become acclimated to the complexities of real-world design. Another issue is that at some schools different design tools are used in

[1] Robert Colwell, Chief Architect of the Intel Pentium Pro, at the panel session at the International Symposium on Computer Architecture, May, 1996 (paraphrased).

different courses, adding to the overhead of tool-learning time to the student. In ICED, the industrial CAD tool suite from Mentor Graphics will be used throughout the curriculum.

In current curricula, students do not participate in projects lasting more than one or two semesters. In practice, projects last much longer, often years. With the major project occurring over three years, ICED necessitates that the student create clear documentation, and deal with outcomes beyond any one course.

There are related projects. In [6] and others, networks and distributed systems are investigated, but only from a programming point of view, and the studies are not integrated with other topics. Integrated research across the curriculum is proposed in [10], but this is in the field of Psychology. In a closer but still different area, [8] proposes an integrated curriculum in Digital Signal Processing and Communications. Many hardware simulation, design, and emulation or construction curriculum projects have been pursued, including the use of modern Field Programmable Gate Array chips, such as in [2, 7], but the study of compilers was not included, networks were not addressed, and the work was not integrated across a curriculum.

2 The New CE Curriculum

A key ingredient to successful digital system design is making appropriate design trade-offs amongst the hardware and software components of the system to achieve a suitably-performing, cost-effective design. The ICED curriculum ties together what is traditionally unrelated content from different courses. This allows students a key, novel experience – the design, over years of studies, of an actual, complete, working network, particularly including a computer with both the processor and the compiler built by the student.

The design tasks to achieve these aims are distributed amongst the required core courses as follows. The courses are taken roughly in the order shown. The catalog descriptions of the courses are given first, then the design tasks are described in *[italics]*, followed by the names of tools used (all of them Mentor, unless noted otherwise).

1. **ELE 201/202 - Digital Circuit Design:** Logic gates, Boolean algebra, combinational and sequential circuits, analysis and design of sequential systems, multi-input system controllers, asynchronous finite state machines. Laboratory experience in digital electronics; logic design projects using standard integrated circuits.

   [Familiarization with industry-standard CAD tools. Design, simulation, construction and test of components for the computer, e.g., an adder, as well as simple control circuits. Introduction to the VHDL hardware modeling language.] Design Architect, Quicksim II, QuickHDL.

2. **ELE 305 - Introduction to Computer Architecture:** Architecture of digital computers. CPU microarchitecture. Instruction execution cycle. Instruction sets. The memory hierarchy. Pipelining, instruction level parallelism, introduction to parallel computing. Register-level design and simulation of a simple computer.

   [Design and simulation of a simple machine instruction set for the student's computer, using VHDL and other CAD tools. High-level design is performed.] Design Architect, QuickHDL, Renoir.

3. **ELE 405 - Digital Computer Design:** Hardware implementation of digital computers. Arithmetic circuits, memory interface, data path, control path, input/output. Synthesis tools. Gate level design, simulation, construction and verification of a simple digital computer.

   [Design, simulation, construction/emulation and test of the digital hardware realizing the instruction set devised in ELE 305. Gate-level design is performed, along with VHDL aids (e.g., for logic synthesis). The same suite of CAD tools is used. The computer will be emulated, debugged and evaluated, using rapid prototyping emulation hardware, and logic analyzers.] Design Architect, QuickHDL, Quicksim II, Leonardo.

4. **CSC 402 - Compiler Design:** Grammars and languages; lexical analysis, parsing and translation, symbol tables, runtime storage administration, object code generation. Students will construct a compiler for a small programming language.

   [A compiler providing translation from a high-level programming language to the machine instruction set of the student's computer is written and tested.] HP logic analyzer source-code–to–analyzer mapper.

5. **ELE 437 - Computer Communications:** Computer networks, layering standards, communication fundamentals, error detection and recovery, queuing and delay-thruput trade-offs in networks, multiple-access channels, design issues in wide and local area networks.

   [A simple wireless or wired local area network is designed, incorporating standard network elements such as layering, error detection and recovery, and throughput management.] Design Architect, Quicksim II, QuickHDL.

6. **ELE 408 - Computer Systems Laboratory:** Engineering design problems involving modern microprocessor systems, operation of ALUs, data paths, control units, input/output, and memory systems.

   [The hardware and software of the computer are integrated, and the overall system is debugged and evaluated using benchmark programs. The effect of hardware/software design modifications on cost and performance is measured. The different computers of the entire class are hooked together in a simple network; communication amongst them is achieved. The network is evaluated. The emulation hardware and logic analyzers are again used.] All of the above tools.

Other optional or potential components (these are not initially required, to ease the realization of ICED):

- **CSC 412 - Operating Systems:** Presentation of the general concepts underlying operating systems. Topics include process management, concurrency, scheduling, memory management, information management, protection and security, modeling and performance.

  [A simple operating system / monitor is written and tested for the computer.] HP logic analyzer source-code–to–analyzer mapper.

- **ELE 447 - VLSI Design and Simulation:** Design and simulation of digital integrated circuits. Extensive use of software tools such as [Mentor Graphics' IC Station] and simulators. Student designs are fabricated and tested.

  [The computer hardware, or parts thereof, are realized again, this time by designing a custom Integrated Circuit (IC). The IC is fabricated.] IC Station, Lsim, etc.

(The complete curriculum is available via the author's web site: `www.ele.uri.edu/~uht`)

Thus, the students are exposed to many industrial CAD tools in the course of their studies, including schematic capture (Design Architect), hardware programming language use (VHDL, via the QuickHDL tools), high-level design (Renoir), and synthesis (Leonardo). With the guidance of the professor, students estimate the consequences on the compiler design when designing the hardware in ELE 305 and 405. During compiler construction in CSC 402, in their Senior year, students may discover that they need to change the balance between hardware and software, and modify their hardware designs accordingly. The network component is addressed in ELE 437. Lastly, in ELE 408, all of the pieces are put together. Experiments on hardware/software tradeoffs are performed, and the network is tested.

Equipment - The Mentor suite was chosen as the primary CAD tool provider in the department after an exhaustive evaluation and comparison of the major CAD tool vendors. Mentor was the clear favorite, being the most complete and lowest cost of the vendors' offerings. It is also very popular in industry, and may legally be used in research (both undergraduate and graduate), especially that involving industrial participation.

We have recently received an equipment grant from the National Science Foundation for the support of ICED. With the grant, we have purchased 10 Sun Ultra 1's with 128 MB RAM to run the Mentor tools and the emulation hardware. We are also using our old Sun IPC's as X terminals, one IPC connected to each Ultra, to stretch the number of workstation seats further. We also plan to purchase a Sun compute server, for jobs too large for the Ultra's.

Our lab setup will be based on commercially available rapid prototyping emulation hardware units employing Field Programmable Gate Arrays (FPGA's); they will likely be Virtual Computer Corp. units. The programmability of the FPGA's allows for many experiments of different computer designs to be made. Emulation is being used more and more in industry, both for final products as well as prototyping, so this will give the student valuable exposure to a modern technology, lacking in the old curriculum. Three or four lab stations will be set up, each with its own emulation unit, logic analyzer and Sun Ultra (out of the 10 above).

3 Summary, Status and Conclusions

For those students desiring hardware and software computer engineering design skills, as well as the underlying theoretical knowledge, to create complex, realistic digital systems, the new URI curriculum will offer a unique experience. We believe that our experiences with this ambitious, new approach to undergraduate curriculum should be a valuable model for programs across the nation.

ICED is becoming a reality; it has been approved, and will become official with the Fall 1997 semester. The Mentor donation was granted two years ago (Fall 1995). Since then we have have used many of the CAD tools with great success in our courses. The new workstations were installed in the Summer of 1997, and will be used in ELE 201/202 and 305 in the Fall. The lab stations will be purchased and configured in the Fall, in time to be used by ELE 405 in the Spring. Although only Freshman are required to begin the new curriculum this Fall, Sophomores and Juniors may also easily switch to the new curriculum at that time, and many have elected to do so. Thus, the first students following ICED will graduate in two years.

The ICED curriculum is a novel attempt to bring the problems and opportunities of a long-term project into the undergraduate curriculum, allowing students to engineer complex digital systems from both software and hardware components, and to comprehend the structure of these systems, from wireless networks to gate-level logic in a CPU.

References

[1] G. M. Brown and N. Vrana. A Computer Architecture Laboratory Course Using Programmable Logic. *IEEE Transactions on Education*, 38(2):118–125, May 1995.

[2] PI: Cook. Electronic Design Automation Fabrication. *NSF Undergraduate Instrumentation and Laboratory Improvement Awards*, DUE-9452121, July 1994. Via http://www.fastlane.nsf.gov/a6/A6QueryPgm.htm, select "Undergraduate ILI".

[3] Georgia Institute of Technology. GATech CmpE 4510 Computer Engineering Design II. Internet web site, November 1996. http://www.ece.gatech.edu/academic/courses/cmpe4510.html.

[4] J. O. Hamblen, H. Owen, S. Yalamanchili, and B. Dao. Using Rapid Prototyping in Computer Architecture Design Laboratories. *IEEE Computer Architecture Technical Committee Newsletter*, pages 44–52, June 1996. Part of the 2nd Annual Workshop on Computer Architecture Education, held in conjunction with the Second High Performance Computer Architecture Conference.

[5] Kaeli, D. R., editor. *IEEE Computer Architecture Technical Committee Newsletter - Special Issue on Computer Architecture Education.* IEEE Computer Society, 1996.

[6] PI: McCabe. A Programming Laboratory for Computer Networks. *NSF Undergraduate Instrumentation and Laboratory Improvement Awards*, DUE-9452037, September 1994. Via http://www.fastlane.nsf.gov/a6/A6QueryPgm.htm, select "Undergraduate ILI".

[7] PI: Perskowski. Use of FPGA Hardware Emulator and VHDL to Teach Digital Design. *NSF Undergraduate Instrumentation and Laboratory Improvement Awards*, DUE-9452593, October 1994. Via http://www.fastlane.nsf.gov/a6/A6QueryPgm.htm, select "Undergraduate ILI".

[8] PI: Thompson. Electrical Engineering Systems Laboratory. *NSF Undergraduate Instrumentation and Laboratory Improvement Awards*, DUE-9552060, July 1995. Via http://www.fastlane.nsf.gov/a6/A6QueryPgm.htm, select "Undergraduate ILI".

[9] A. Varma, L. Kalampoukas, D. Stiliadis, and Q. Jacobson. CPU Design Kit: An Instructional Prototyping Platform for Teaching Processor Design. *IEEE Computer Architecture Technical Committee Newsletter*, pages 23–26, June 1996. Part of the 2nd Annual Workshop on Computer Architecture Education, held in conjunction with the Second High Performance Computer Architecture Conference.

[10] PI: Volckmann. Research Across the Curriculum in Psychology. *NSF Undergraduate Instrumentation and Laboratory Improvement Awards*, DUE-9452253, September 1994. Via http://www.fastlane.nsf.gov/a6/A6QueryPgm.htm, select "Undergraduate ILI".