THE MICROPROCESSOR AS A MICROCOSM: A HANDS-ON APPROACH TO VLSI DESIGN EDUCATION

David Harris[1]

Abstract *¾ Introductory Very Large Scale Integration (VLSI) design courses face a tension between teaching good design practices by example and giving students the freedom to learn for themselves through open-ended team design projects. On one hand, guiding students through implementation of a well-planned chip is an efficient way to teach design and verification methods, CAD flow, and proper use of datapaths, arrays, and synthesized logic. On the other hand, permitting students to select and carry out a design project of their own choice through tapeout is very motivational and provides a deeper mastery. The author has developed a new undergraduate VLSI course at Harvey Mudd College that reconciles this conflict by packing a complete set of microprocessor design labs into the first five weeks, leaving the remainder of the semester for a major team-based design project.*

Index Terms ¾ CAD, project-based design education, VLSI

INTRODUCTION

Integrated circuit design has advanced from a highly specialized skill practiced by a tiny elite to an essential ability for a vast number of practicing electrical engineers. Very Large Scale Integration (VLSI) design, commonly taught at the graduate level in the early 1990's, is now available to juniors and seniors in many engineering schools. However, there is still debate over the best way to structure VLSI courses. Some instructors teach the course primarily through lecture; some use projects; and some argue that VLSI is so fundamental that it should be part of the introductory digital electronics course. This author believes that the only way to really understand how to build a chip is to actually build a chip and that this is most feasible in a project-based VLSI course.

Among those who teach a project-based course, some assign a project that is carefully architected by the instructor to expose students to key concepts while others allow students to choose their own project and take ownership of a design. The former approach provides systematic exposure to a variety of CAD tools and methods and demonstrates good design practices that might not be obvious to a beginner. The later approach is particularly valuable for undergraduates, for whom this may be the first major design project taken from proposal through preliminary design and detailed design to verification. Building one's own chip also marks a transistion for some students from bright kids with no special knowledge to professional engineers designing microscopic circuits. This paper describes an attempt to combine the advantages of both approaches while keeping student workloads to a reasonable level. It combines five weeks of laboratory exercises in the beginning of the semester with a major student-defined project later in the semester. The laboratories involve the design of an 8-bit microprocessor, which serves as a microcosm to illustrate key design issues. The skills acquired in these laboratories become the foundation for the project. This is an unusual approach in that for many classes a microprocessor design consumes the entire semester. By providing students with much of the microprocessor design and directing them to complete the missing components, the laboratories provide most of the learning experience without as much repetitive and tedious labor.

The laboratories and project are part of E158, Introduction to CMOS VLSI Design, taught by the author as a 3-unit course at Harvey Mudd College in the spring of 2001 and 2002. The course is taken by juniors and seniors, primarily from the Eingeering Department, but also from Computer Science, Physics, and Chemistry. The initial enrollment was 42 in 2001 when the course was first offered and 15 in 2002 when the time conflicted with two required classes. To put these numbers in perspective, approximately 60 engineering majors graduate each year from the nonspecialized engineering program. Nine students dropped the class in 2001; none have dropped in 2002. Prerequisites are E84, Electronic and Magnetic Circuits and Devices, and E85, Digital Electronics and Computer Engineering. All course materials are on the class web page [1].

This paper describes E158 with emphasis on the microprocessor labs and final projects. It defines the goals of the course and seeks to provide enough information to facilitate similar projects elsewhere.

SYLLABUS

Table 1 shows a week-by-week list of the lecture topics and assignment deadlines as the course was offered in the Spring of 2002. The initial two weeks provide enough background in transistors and layout for students to begin the laboratories. The next lectures focus on designing for peformance. The later lectures address specific techniques for designing common VLSI structures. The final weeks include design reviews and advanced topics.

[1] David Harris, Department of Engineering, Harvey Mudd College, 301 E. Twelfth St., Claremont, CA 91711 david_harris@hmc.edu

0-7803-7444-4/02/$17.00 © 2002 IEEE **November 6 - 9, 2002, Boston, MA**
32nd ASEE/IEEE Frontiers in Education Conference

The first five weeks are dedicated to the series of laboratory projects in which students are guided through completing a partially designed MIPS microprocessor. Most of the remainder of the semester is available for the team chip design projects. Most students are simultaneously enrolled in Clinic, a year-long team project sponsored by industry. VLSI projects complete two weeks before the end of the semester to avoid conflicting with the Clinic crunch period. Six problem sets are scattered across the semester to reinforce material that is not directly addressed by the labs or project.

TABLE I
SYLLABUS

23-Jan	Introduction and overview	
28-Jan	Circuits, fabrication, and layout	
30-Jan	Microprocessor example	Lab 1 due
4-Feb	-- ISSCC: No Class --	PS 1 due
6-Feb	CMOS transistor theory	Lab 2 due
11-Feb	DC gate characteristics	
13-Feb	CMOS processing technology	Lab 3 due
18-Feb	Logical effort	PS 2 due
20-Feb	Interconnect engineering	Lab 4 due
25-Feb	Simulation	Preliminary prop due
27-Feb	Combinational circuit design	Lab 5
4-Mar	Circuit families	Project proposal due
6-Mar	Sequential circuit design	
11-Mar	Adders	PS 3 due
13-Mar	Datapath functional units	Floorplan due
18-Mar	-- Spring Break: No Class --	
20-Mar	-- Spring Break: No Class --	
25-Mar	Memories	
27-Mar	Memories	Schematics complete
1-Apr	Control system design	PS 4 due
3-Apr	Design for testability	
8-Apr	In-class design reviews	Leaf cells complete
10-Apr	In-class design reviews	
15-Apr	Power and clock distribution	
17-Apr	Skew-tolerant circuit design	Final Project due
22-Apr	Asynchronous design	
24-Apr	Low power design	PS 5 due
29-Apr	Yield, scaling, and economics	
3-May	Tour of Intel Microprocessors	PS 6 due
6-May	Presentation Day	Project presentations

The course loosely follows material from Weste & Eshraghian [2] and Mark Horowitz's Stanford EE271 notes. The instructor's books on Logical Effort and Skew-Tolerant Circuit Design are recommended readings [3], [4].

Grading is based 40% on the laboratories, 45% on the final project, 10% on problem sets, and 5% on weekly in-class activites.

CAD TOOLS

While teaching specific CAD tools is not a primary goal of the class, skill with tools is nevertheless essential to completing a design project. Four sets of tools were considered: Cadence, Tanner, Magic, and Electric. Harvey Mudd's Engineering Computational Facility is primarily Windows-based, with limited Solaris support. Students also have ready access to Windows machines. The Cadence tools are powerful industry standards, but are too time consuming to maintain for a small department. The Tanner tools are easy to learn and sufficiently powerful for class projects, but cost universities even more than Cadence. Magic is free and has a long history of academic use, but has a clunky interface and is primarily available on Unix. Electric [5] is a free, open-source CAD package supporting schematic and layout entry, simulation, and verification. It runs on Windows, Unix, and Macintosh and is easy to learn and use. Electric is the least mature of the tools considered, but Harvey Mudd is working closely with the developer, Dr. Steve Rubin, to refine the tool. At the present, is is still prone to crashing and has a number of idiosyncracies, but is powerful enough to handle class projects quite well. Electric is supplemented with Synopsys Design Analyzer for logic synthesis.

LABORATORIES

The laboratory component of the course is intended to teach good design practices. On on hand, the laboratories must guide students through a design large enough to demonstrate the need for such practices. On the other hand, they must be possible to complete in a reasonable amount of time. To balance these competing objectives, the laboratories guide students through assembling an 8-bit microprocessor implementing a subset of the MIPS instruction set. The microprocessor is based on the Patterson & Hennessy multicycle MIPS processor [6] that students designed and simulated at the schematic level in the prerequisite E85 [7]. Much of the processor is provided, but pieces are missing. The microprocessor serves as a microcosm illustrating many of the steps involved in a larger chip design. By designing each of the missing components, students gain all of the key skills required for a design of this scale while spending only a small fraction of the time.

It is important that laboratories stress timeless principles of engineering complex systems as well as the specific practices for the CAD tools and implementation technology [8]. Specific goals for the laboratory component include:

- design techniques: regularity, hierarchy, modularity
- CMOS layout styles: arrays, datapaths, random logic
- CMOS design with the Electric CAD tool including:

- schematics and layout editors
- leaf cell design for datapaths and control logic
- datapath, control, and full-chip interconnect
- synthesis (Synopsys), place & route
- pad frame generation and routing
- simulation: IRSIM, test vector generation
- verificaton: Design rule checking (DRC), Electrical rule checking (ERC), layout vs. schematic network consistency checking (NCC)
- pretapeout checks

- exposure to a complete, well-designed chip
- understanding the interdependence of components in a complex design

The students are issued an Electric library at the beginning of the semester with all of the components except those listed in the laboratories below. In each week, building blocks from the previous lab are combined with other components provided to compose larger blocks. Figure 1 shows the completed chip layout. Verification is essential; carelessness in early labs leads not only to a poor grade but also to more time debugging the later labs. A few students were issued electronic versions of lab solutions in situations where their libraries were corrupted by Electric.

FIGURE. 1
8-BIT MIPS PROCESSOR LAYOUT

Lab 1: Gate Design

Schematics, icons, layout. Simulation, design rule checking, electrical rule checking, network compare. Guided through NAND2 and AND2 cells in a detailed lab writeup. Complete NOR2 and OR2 without assistance.

Lab 2: Full adder Design

Research and design a full adder cell schematic and layout under constraints so adder will snap together with the remainder of the datapath. Verification, including use of command files to automate testing.

Lab 3: Datapath and Zipper Assembly

Combine AND2, OR2, and fulladder cells with provided multiplexer to build ALU. Attach ALU to remainder of datapath bitslice and route datapath interconnects over the cells. Add drivers to the zipper to provide control signals to the ALU. This was the most time consuming lab as students learned to use NCC on complex cells.

Lab 4: Controller Design

NOR3 gate design and layout on standard cell pitch. Manual design and layout of alucontrol decocder using standard cells. Modify Verilog model of controller FSM to support ADDI instruction; synthesis and place & route of controller. Simulation and verification.

Lab 5: Microprocessor Assembly

Layout assembly of microprocessor core from datapath, alucontrol, and controller (see Figure 1). Test vector development from assembly language. MOSIS TinyChip pad frame generation and routing. Tapeout checks.

PROJECT

Once students are exposed to VLSI design in the microcosm through the microprocessor labs, they are turned loose to propose and implement a chip of their own. Projects are generally carried out in teams of two. The projects must fit on a MOSIS TinyChip in a 0.6 μm process (3400 x 3400 λ of core area excluding I/O pads) and should be of reasonable difficulty, as judged by the instructor. Projects have included:

- FIR filter for guitar acoustic effects
- Simple digital neural netork
- GPS searcher
- Wallace-tree multipler
- DES encryption
- Hangman game
- Java-based PLA generator and associated test chip

The project is not only the capstone of the course, but also, for many students, the largest design project undertaken in any course excluding Clinic. This leads to further educational objectives not necessarily applicable to traditional graduate-level VLSI courses. These objectives include:

- Teamwork and leadership
- Formal design and communcation: proposals, preliminary design, resource budgeting, detailed design, design reviews, documentation, and presentations
- Experience moving from concept to a digital design

Work expands to fill the time available, so project milestones are strictly enforced. Most milestones involve a conference or demonstration between teams and the instructor. Milestones include:

Preliminary Proposal

Teammate and general project topic identified. Feedback on suitability of project in size and dificulty.

Proposal

Inputs, outputs, and functional description.

Floorplan

Identify all cells required for project and budget area for each cell. This is a shock for most students who are accustomed to toy projects; such students do not realize that sketching out the entire design is only the first phase and that most of the work lies ahead. It is also often difficult for undergraduates who have never applied formal technqiues such as finite state machines and logic design to large and ambiguously stated specifications.

Schematics Complete

Schematics drawn, test vectors written and simulating.

Leaf Cells Complete

Roughly 50% of layout complete; this milestone depends on the specific nature of each project. Most students underestimate the time required for layout, so this checkoff emphasiszes how much work lies ahead.

Design Review

In-class presentation of projects with focus on interesting parts of the architecture and on difficult areas. Feedback from instructor and classmates.

Project Complete

Layout complete and wired to a padframe. IRSIM test vectors completed successfully. DRC, ERC, and NCC. Final report including chip documentation, comparison of actual and floorplanned areas, and records of time spent on each cell. Project library and test vectors archived.

Formal Presentation

Presentation open to entire campus community during Presentation Days. Chip plot generated (see Figure 2).

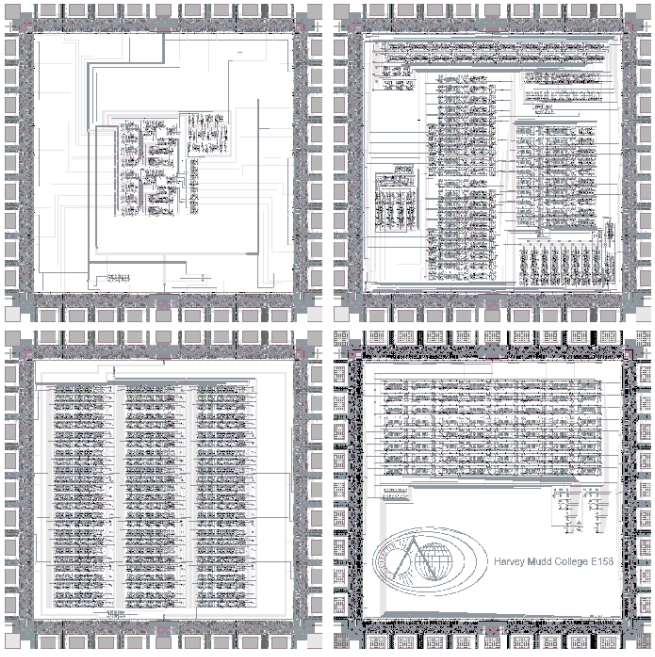

FIGURE. 2
SPRING 2001 FABRICATED PROJECTS

Up to four projects are selected for fabrication through MOSIS [9] with support from the MOSIS Educational Program. To qualify, at least one student on the team must be on campus in the fall and willing to commit to testing the chip after fabrication.

RESULTS

The results of the teaching approach were assessed based on student surveys, formal course evalutions, grades, silicon test results, and subjective observations of the instructor.

Students were surveyed about the time spent on each assignment. The average times reported in 2002 are given in Table II. These times are slightly lower than in 2001 because of numerous bug fixes in Electric. Lab 3 could be split into two labs, one building the ALU and the other finishing the datapath.

TABLE II
HOURS PER ASSIGNMENT

Assignment	Hours
Lab 1	5.4
Lab 2	8.9
Lab 3	17.0
Lab 4	7.8
Lab 5	7.5

Project data is only available from 2001 at the time of writing. Projects varied widely in difficulty and teams varied in ability, but 100 hours of work per team was typical. Thirteen of the seventeen projects simulated correctly and passed all layout verification steps. The remaining four projects demonstrated a good deal of learning even though they were not completely successful.

Chips are tested using a low-speed functional tester developed at Harvey Mudd [10]. The same IRSIM test vectors used for simulation are directly applied to the device under test and outputs are automatically checked against expectation. Of the four chips fabricated, one was operational on first silicon, one was operational except for an output shorted to ground (which was not caught under the old DRC but would now be flagged by Electric), and two have not yet been tested. Students misplace their IRSIM command files and have difficulty finding time to test chips given their other coursework.

In Spring 2001, the course was taught using computer-based slides based on Mark Horowitz's Stanford EE271 course. The slides were praised by students for being well designed and covering a large amount of material. However, students tended to nap in the darkened room, lecture attendence sagged, and comprehension seemed to be low on material not reinforced by labs or the project. In Spring 2002, the course is being taught at the chalkboard. This requires cutting about 20% of the material and students do not have hard copies of slides to refer to. However, attendence has been nearly perfect through the first half of

the course, students ask far more questions, and comprehension appears to be much higher. This is consistent with the author's experience that students are more easily engaged at the chalkboard than with computer presentations. Also, more short problem sets are being assigned to reinforce the lecture material that is not practiced elsewhere.

The students who were not making a serious effort dropped the class in 2001. The median grade of the remaining students was an A-; this is much better than the norm at Harvey Mudd and reflects an oustanding level of achievement by the class.

The overall course evaluation was 6.6/7.0 in Spring 2001 and 6.5/7.0 in Spring 2002. This compares to the campus-wide average of 5.8. Student comments below support the combination of labs and project and confirm that the early project deadline was a good approach. Feedback on lectures and problem sets lead to the changes for 2002.

"The labs were excellent and really helped me to understand the material. Also, the project at the end helped to bring everything together. It was great that the class ended in early April so we didn't have to worry about it at the end. I had a difficult time understanding the lectures. This was probably because we were never forced to learn the material. Adding homework would have helped, but then I wouldn't have slept all semester."

"Hands-on lab work is great for getting a feel for what's hard and what's easy about design. The lectures that focused on technqiues that most chip designers aren't familar with were really cool. I think the pace was just fine and I was especially glad to have the project done before the end of year crunch. The labs and project didn't really require the use of the advanced design techniques like logical effort and skew-tolerant domino. It may make the course more difficult, but I think students would be glad to learn it. (Don't forget that Mudders won't learn it if they don't have to.)"

"The labs were very informative and the final project was really interesting, as was the course. The timing was beautiful to have the course end before Clinic gets bad. Pacing was a bit fast."

"The labs were cool, especially the final project. I liked the focus on layout and really understanding how things work at a physical level."

"Being able to actually make my own chip really contributed to my learning."

CONCLUSIONS

In conclusion, this paper has described a project-based VLSI course that combines the advantages of demonstrating good design practices with the student-driven learning based on open-ended design projects. The key is a set of labs in which students complete a partially finished microprocessor; the missing components are carefully selected to illustrate many of the essential techniques for chip design. These labs lay the foundation of knowledge so that students can proceed with their own design projects.

Based on experience teaching the course, several changes are planned. Labs should be spread over six weeks to smooth the workload; the final lab may overlap the preliminary project proposal. Students should turn in electronic versions of their simulation test vectors to facilitate testing chips after fabrication. Overall, the workload is slightly high for a three-unit course but consistent with a number of other technical electives.

Students demonstrated mastery of chip design at the logical and layout levels through the labs and successful projects. However, students did not have much opportunity to practice other skills such as delay estimation and design for speed, application of circuit families, or low-power design. Such topics were covered in lecture but not retained as well. While there is a limit to how much material can be exercised in the time available, one might desire closer connection between lectures and assignments.

ACKNOWLEDGMENT

The author acknowledges the MOSIS Educational Program (including Sematech and AMI) for funding fabrication of student projects. Steve Rubin has devoted an enormous effort to improving the Electric tools and Sun Microsystems has supported this effort. David Diaz, Sean Kao, and Daniel Lee assisted with development of the labs. The 2001 and 2002 E158 classes at Harvey Mudd have offered numerous constructive suggestions and have been a joy to work with.

REFERENCES

[1] `www3.hmc.edu/~harris/class/e158`

[2] Weste, N., and Eshraghian, K., *Principles of CMOS VLSI Design*, 2nd Ed., Addison-Wesley, 1993.

[3] Sutherland, I., Sproull, B., and Harris, D., *Logical Effort*, Morgan Kaufmann, 1999.

[4] Harris, D., *Skew-Tolerant Circuit Design*, Morgan Kaufmann, 2001.

[5] `www.staticfreesoft.com`

[6] Patterson, D., and Hennessy, J., *Computer Organization & Design*, 2nd Ed., Morgan Kaufmann, 1998.

[7] Harris, D., "A Case for Project-Based Design Education," *Intl. J. Engineering Education*, vol. 17, no. 4-5, pp. 367-369, 2001.

[8] B. Carlson, "Principles vs. Practices in Undergraduate Microelectronic Systems Education," *Proc. 2001 Intl. Conf. Microelectronic Systems Education*, Las Vegas, NV, June 2001.

[9] Pina, C., "MOSIS: IC Prototyping and Low Volume Production Service," *Proc. 2001 Intl. Conf. Microelectronic Systems Education*, Las Vegas, NV, June 2001.

[10] `www.onehotlogic.com`