Using open-ended design projects in an open lab to teach lifelong learning skills in an IC design course

D. W. Parent

EE Department, San Jose State University, San Jose CA 95192-0084, email
dparent@email.sjsu.edu, PH 408.924.3863, FX 408.924.3925

The ABET criterion, recognition for and the ability to engage in life-long learning, is one of the most important, given that we as engineering educators can not teach every concept that students will encounter over the span of a 20-40 year career as an engineer. We believe that one method to teach and verify this critical skill in the area of Integrated Circuit (IC) design, is to have the students learn IC CAD tools by following well crafted tutorials and then complete open-ended IC design projects in an open lab. The project is supposed to be a full custom design of a digital circuit, with logical, timing and layout specifications using standard CMOS circuits in a .5µm CMOS process (AMI06). The frequency of operation for the circuit is supposed to be 200MHz, and the power density below 23C/cm^2 to avoid the use of active cooling.

Our lab environment consists of 40 UNIX workstations with Cadence Design Systems Virtuoso IC CAD software using the NCSU MOSIS[1] design kit. Students can come to lab or remotely log in at their convenience. No formal lab hours or TA/professors are on hand during a formal lab section. There are three full custom IC design courses that follow this model.

An additional advantage of requiring students to learning the CAD tools and completing open-ended IC design projects in an open lab environment is that this kind of environment mimics an industrial setting. Students who have taken our course, who are now working in industry, all state that the course environment closely matches that of their work environment.

Learning IC CAD tools:

The key items required for students to teach themselves the CAD tools are well crafted tutorials that in a very exacting manner detail the use of the tool. In addition, these tutorials must be as "bug free" as possible. Even though writing, verifying, updating these tutorials is a time consuming process (expensive), it is still more cost effective than staffing a lab section with a professor or TA. The cost of developing these tutorials was paid for through a generous gift from Cadence Design Systems.

In EE-166, Introduction to CMOS Digital Circuit Design, students learn the Bottom-Up design flow by completing a tutorial Bottom up IC Design-flow Using CDS tools[2] as a homework assignment. Students submit plots of the schematic and extracted views of a circuit along with an LVS (Layout vs. Schematic) report and the post extracted simulation to prove that they have done the homework. (Since this assignment is part of the homework grade it is worth only 2% of the total grade.) This tutorial is very exacting in that it shows the student every step to learn the basics of the tools. Other tutorials have been developed such as using NC-Verilog[3] to verify the logic of a schematic before starting on the timing analysis. These other tutorials and modules had been left to the students to do on their own, but due to the fact that only a few students actually do them on their own, we will require students to complete all the tutorials as part of a homework grade. There will also be quizzes based on select case studies to make sure students read the case studies.

EE-166 is an elective three-credit course that has an electronics design, introduction to digital design, and a solid-state physics courses as prerequisites. It has been taught for six semesters and currently 50% of the students are seniors and the other 50% are master's students. (EE-166 is a prerequisite to our graduate level digital circuits course.) A review homework consisting of questions from the electronics design, digital design, and solid-state physics course is given the first day of class and is due the next class period (two days later). It has been found that our undergraduate students do not have a problem with the digital design content, but do have trouble (on average) with the solid state and electronics design questions. It was found however that having the electronics design course was not a predictor of success in EE-166. (The on-line registration system did not check prerequisites prior to the Fall 2003 semester.) Since then it has been found that the electronics design course content varied from instructor to instructor, and efforts are under way to teach the electronics design course more consistently. The students in general do not know any spice before taking EE-166.

After four semesters the main tutorial was bug free and as a result 90% of students did not need help during our office hours to complete it. The majority of help given in office hours (sometimes office hours are conducted in the lab) was to what to do with the tools, rather than how to use them.

Projects:

All projects in our Introduction to IC Design course involve the design of flip-flop circuits, which are not covered until late in the semester. In order for a design team to finish the project on time, the students have to learn flip-flop design on their own. Furthermore, the projects contain topics that are not covered in class at all. One example was an ALU designed with transmission gates. Since in this course, transmission gate logic and ALUs are not covered, the students were forced to teach these concepts to themselves in order for the project to succeed. The project is worth 20% of the grade. Other sample projects are multipliers, adders, and IIR filters. The students work in teams of 2 to 4 students. The students pick the groups they are to work with.

The main conceptual problem in completing the project that students have is using cell-based design, and that cells can be nested inside each other. Sometimes the students think that the project is too difficult because they believe there are a large number of logic gates to design. What they do not realize is that you do the cells you need once and then re-use them. For example a student thought a pseudo random bit stream generator was going to be too difficult to accomplish in one semester because the circuit used 21 D-Flip Flops and an XOR gate. What the student did not realize is that there are really only three logic blocks to design (the Flip Flop, XOR and the PRBS generator).

The second conceptual problem was how to break up a design to solve for the widths of the transistors to meet timing. Solving for the widths is gone over in class for one gate to drive another, but there is no example given on how to do this for gates sandwiched between D-Flip Flops in tutorial form at this time. This question was asked so often in office hours it was included on the final exam. On the final at least 90% of the students were able to correctly describe the approach you would follow to solve for the widths to meet the timing specification.

The third problem that at least 50% of the students had was project management. This has been historically a problem in this class over the semesters so we developed a set of notes describing the design of a FIFO[4] ready for fabrication through MOSIS. In the notes there are charts showing how long it took to design, layout, DRC, Extract, LVS, and simulate each part. The charts show that as the number of gates increase the time it takes to do a design is increasing almost exponentially. As a result only one group did not have their circuit working in time. There were still too many students working at the last minute, needing extra help outside of office hours.

In the senior level course EE-166 there are usually about 10 graduate students. They tend to have less trouble with project management and have much less trouble finding information on their own than undergraduate students. The graduate students tended to try and complete more complex designs, while undergraduate students tended to try and do projects that they had seen before. For example many undergraduate students like to do serial to parallel data converters that they had seen in a previous logic course. These projects were so common that they are no longer allowed.

The most common request from students was to assign TA's to be in lab to help students. At first this was done, but it had a negative affect. There was a large population of students that would not even try to find information on their own, if a TA was in the lab. An other problem was that when the TAs were off duty (but had to use the lab for their own work) students would not let them work because they would very aggressively ask questions.

Even though having TA's in lab did not work out, the students are encouraged to help each other learn the material. Since in EE166 the grades are not curved there is no disincentive to help your fellow classmates, and one could observe students helping each other out. Sometimes this was not ideal, because sometimes a bad idea would propagate through the lab.

Prevention of Cheating:

There is one disadvantage to this kind of learning environment; since there is no TA or professor in lab watching the students work; it is easier for students to cheat. In order to prevent cheating the accounts need to be handed out in a manner that it is easy to tell who owns the account. In small classes one could use the first letter of the first name and the last name together as the user id. Since the number of users of our system approaches 400 unique users we use the first 4 letters of the student's last name and the last 4 numbers of a university issued student ID (not SSN). This makes it easy to grade assignments because the user ID appears in some of the plots. Another effective technique to prevent cheating is to monitor the license log file that keeps a track of each user's license activity. We tell the students several times that we have the ability to monitor how much they work because the idea is to prevent cheating, not "catch cheaters". Another method is to look at the disk usage of the users. Another method to monitor student work is to just see how much data are in there home directories. In this type of class, each user should have about 10-20MB of data if they have been really doing the assignments. These very quick methods will prevent most cheating. Since adopting these methods there was no observed cheating in the homework assignment where students learned the cad tool. It is still difficult to monitor if some one in the group is not helping with the project.

Summary:

Although some students will still become stuck and need a professor's help when completing an IC design (near the end of the semester we do have to hold extended office hours) we still feel that having students complete open-ended IC design projects in an open lab environment for advanced level courses is an effective method for teaching life long learning skills. The most important required item to teach how to use a particular CAD tool are well written, detailed bug-free tutorials. Extra time is needed in the form of extended professor office hours to help students complete their projects, but it is much more cost efficient to have some extra office hours as needed to meet this need, rather than commit to a three hour lab period each week.

[1] T. Schaffer, A Stanaski, A. Glaser, P. Franzon, "The NCSU Design Kit for IC Fabrication Though MOSIS", http://www.cadence.ncsu.edu/CUG1998/icug98.pdf, 1998 Cadence Users Group Conference.

[2] D. W. Parent, http://www.engr.sjsu.edu/~dparent/ICGROUP/CDS_1.pdf

[3] D. W. Parent, http://www.engr.sjsu.edu/~dparent/ICGROUP/MOD2.htm

[4] D. W. Parent, http://www.engr.sjsu.edu/~dparent/ee166/FIFO_CHIP.pdf