AC 2007-814: DESIGN AND IMPLEMENTATION OF A PROGRAM OUTCOME ASSESSMENT PROCESS FOR AN ABET-ACCREDITED COMPUTER ENGINEERING PROGRAM

Fadi Kurdahi, University of California, Irvine

Fadi Kurdahi has been a professor in the Electrical Engineering and Computer Science Department at the University of California, Irvine since 1987. He holds a PhD degree from the University of Southern California in Computer Engineering. From 2004 till 2006 he was the ABET lead person for the Computer Engineering degree program in the EECS Department.

Judy Shoemaker, University of California, Irvine

JUDY SHOEMAKER is Director of Research and Evaluation for the Division of Undergraduate Education at UC Irvine. Dr. Shoemaker holds a Ph.D. from the University of Kansas in educational psychology and research methods. She is currently serving as a consultant to the Henry Samueli School of Engineering, assisting with assessment of student learning outcomes. She has conducted numerous evaluations of educational programs, including programs funded by NSF, FIPSE and the Department of Defense. Before coming to UCI she was a program officer with the U.S. Department of Education and also worked for three years with the National Assessment of Educational Progress (NAEP).

John LaRue, University of California-Irvine

Design and Implementation of a Program Outcome Assessment Process for an ABET-accredited Computer Engineering Program

Abstract

This paper describes the design and implementation of a program outcomes assessment process for the Computer Engineering Program at the Henry Samueli School of Engineering, the University of California, Irvine. The purpose of the assessment process is to collect and analyze information on student performance in order to improve student learning and the effectiveness of the curriculum, and to meet the ABET accreditation requirements. In the last two years we have adopted two new direct measures of program outcomes which are described in this paper: (1) course-embedded assessment which makes use of assessment results already being collected as part of regular coursework, and (2) a scoring rubric for assessing program outcomes related to the required senior design project. Assessment results from 2004-05 indicated that a relatively small percentage of students achieve some of the program outcomes. After making adjustments to the curriculum, assessment results from 2005-06 indicated that the vast majority of students achieved all the program outcomes.

Background

The Computer Engineering program resides in the department of Electrical Engineering and Computer Science (EECS) in The Henry Samueli School of Engineering at the University of California, Irvine. As of Fall Quarter 2005, the Computer Engineering program enrolled 273 undergraduates (including students with multiple majors). The curriculum for computer engineering majors includes a core of required courses in mathematics, physics and chemistry; engineering science courses, including a required senior design project; and campuswide general education requirements. One feature of our campus is that Engineering majors have the same comprehensive general education requirements (including foreign language) as every other major on campus, which ensures that our students obtain an excellent grounding in the liberal arts and sciences. The University operates on the quarter system (Fall, Winter, and Spring).

Assessment Plan

In response to ABET requirements regarding Criterion 3 (Program Outcomes and Assessment), we have developed an overall assessment plan to measure program outcomes. The assessment plan is shown in Table 1 and the schedule for assessment activities is shown in Table 2. The assessment plan includes a mix of direct and indirect measures of program outcomes. The direct measures are (1) course-embedded assessment and (2) a scoring rubric for the senior design project. These two direct measures are described in more detail below. The two indirect measures are end-of-course student surveys and a graduating senior survey.

Our assessment plan has several characteristics worth noting. First, we listed all of our program outcomes which were adopted from ABET's (a) to (k) outcomes, plus those specific to our program, which we labeled (l), (m) and (n). Second, we have included multiple assessment measures for each outcome, instead of relying on a single assessment measure per outcome.

Using more than one method of assessment helps improve the reliability of our conclusions.[1,2,3] According to Gloria Rogers, "the use of multiple assessment methods provides converging evidence of student learning."[3] Furthermore, multiple measures provide better means of diagnosing and addressing problems in the curriculum as well as in the assessment methods themselves. For example, if the results generated by individual measures for an outcome vary profoundly, further analysis on the cause would be very useful.

And, third, as noted above, our assessment plan contains at least one direct measure for each program outcome. Direct measures involve "direct examination or observation of student knowledge" in comparison to indirect measures which "ascertain the perceived extent or value of learning experiences." [3] Course exams, quizzes, tests, reports and presentations are examples of direct assessment; surveys, interviews and focus groups are examples of indirect assessment.[4]

Another feature of our assessment plan is that we have elected to conduct assessments in upper division courses only. This decision was made for two reasons: (1) program outcomes are, by definition, what students are expected to know and be able to do at the time of graduation, so it makes sense to conduct the assessment at times close to graduation, and (2) to focus on only a few assessment activities each year, thus reducing the assessment burden on our students and our faculty. The courses chosen were EECS 111, System Software; EECS 115, Introduction to VLSI; EECS 129, Senior Design Project; and EECS 140, Engineering Probability.

Table 1. Assessment of Program Outcomes – Computer Engineering

Program Outcomes	**Course-Embedded Assessment (FCARs)** ***(Direct)***	**Senior Design Project Rubric** ***(Direct)***	**Student Course Survey** ***(Indirect)***	**Graduating Student Survey** ***(Indirect)***
(a) An ability to apply knowledge of mathematics, science, and engineering.	EECS 111		X	X
(b) An ability to design and conduct experiments as well as analyze and interpret data		X	X	X
(c) An ability to design a system to meet desired needs.		X	X	X
(d) An ability to function on multidisciplinary teams		X	X	X
(e) An ability to identify, formulate, and solve engineering problems.	EECS 115		X	X
(f) An understanding of professional and ethical responsibility.	EECS 129		X	X
(g) An ability to communicate effectively.		X	X	X
(h) A broad education necessary to understand impact of engineering solutions in a global/societal context	General education courses	X	X	X
(i) Recognition of the need for and ability to engage in lifelong learning.	EECS 115		X	X
(j) Knowledge of contemporary issues.	EECS 115		X	X

(k) An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.	EECS 115		X	X
(l) Knowledge of probability and statistics, including applications to computer engineering.	EECS 140		X	X
(m) Knowledge of mathematics, and basic and engineering sciences, necessary to carry out analysis and design appropriate to computer engineering.	EECS 115		X	X
(n) Knowledge of discrete mathematics.	EECS 140		X	X

Table 2. Assessment Schedule -- Computer Engineering

Assessment Tool	Respondents	Schedule	Analysis	Review process
Course-embedded assessment	Students enrolled in each course	Every quarter	Every quarter	Instructor review
	Faculty (FCARs)		Once a year	Undergraduate committee review
Senior design project	Seniors	Once a year (end of senior year)	Once a year	Undergraduate committee review
Course surveys	Students enrolled in each course	Every quarter	Every quarter	Instructor review
Graduating student survey	Graduating Seniors	Once a year (end of senior year)	Once a year	Undergraduate committee review

1. *Course-embedded assessment using Faculty Course Assessment Reports (FCAR's)*

Course-embedded assessment is an efficient method of collecting information on student learning.[1] It uses the assessment processes that are currently in place in every university course and has been recommended by assessment specialists. One method designed to keep track of the results of course-embedded assessment is a Faculty Course Assessment Report (FCAR), which we saw demonstrated at a recent ABET conference.[5] An example of a completed FCAR is included in the Appendix.

We started the course-embedded assessment process by reviewing and then revising our curriculum map, which shows how each of our required courses is related to the program outcomes. Curriculum maps are often recommended as a means of determining how well the curriculum supports the expected learning outcomes (i.e., program outcomes).[4] Our revised curriculum map is shown in Table 3. In this table, each cell entry describes the relationship between the course and the program outcomes, where I = Introduced, R = Reinforced, and A = Assessed.

The curriculum map was developed from our individual course outlines, recently updated by the faculty coordinators of each required course. An example of a course outline is included in the Appendix. Course outlines include both the course outcomes (CO's) and the program outcomes

(POs) associated with that course. Each faculty coordinator rated the degree of relationship between CO's and POs using the following rating scale: S = Strong, M = Medium, or L = Low. The collected information matrix was then presented to the departmental curriculum committee, discussed and approved. Based on those discussions the curriculum map seen here (Table 3) was created.

As noted above, we have chosen to use data from the upper division core courses as the primary vehicle for assessing all of the POs, except (h) (a broad education necessary to understand impact of engineering solutions in a global/societal context). The rationale here is that since POs should measure what the students are supposed to achieve by the time they graduate, the best place to measure accomplishment of those outcomes would be as close to the time of graduation as possible. The four courses we selected were: EECS 111, Systems Software; EECS 115, Introduction to VLSI; EECS 129, Senior Design Project; and, EECS 140, Engineering Probability.

To decide which of the four senior courses should be used for assessing a particular PO, we used a coverage matrix developed as follows: for each program outcome, we chose the course which has the highest rating or the most "S" ratings (where S = Strong) in its CO coverage of that PO. The unit for assessment is the percent of students passing the specific criterion (either course or program outcome, as appropriate), as this is easy to normalize across course or program outcomes.

Faculty from the relevant courses have been asked to identify course exams, homework assignments, reports, or other student work which illustrates achievement of the relevant outcomes. The results of these assessments are captured in the Faculty Course Assessment Reports. This form is modeled after other FCARs seen at ABET conferences, and includes spaces for information on enrollment and course grades, plus the specific student work which is related to the outcomes being measured in that course.[5] Then faculty add information on students' performance compared to a pre-set criterion of achievement. A completed FCAR is shown in the Appendix.

Table 3. Curriculum Map -- Computer Engineering

Core courses	a. An ability to apply knowledge of mathematics, science, and engineering.	b. An ability to design and conduct experiments as well as analyze and interpret data.	c. An ability to design a system to meet desired needs.	d. An ability to function on multidisciplinary teams.	e. An ability to identify, formulate, and solve engineering problems.	f. An understanding of professional and ethical responsibility.	g. An ability to communicate effectively.	h. A broad education necessary to understand impact of engineering solutions in a global/societal context.	i. Recognition of the need for and ability to engage in lifelong learning.	j. Knowledge of contemporary issues.	k. An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.	l. Knowledge of probability and statistics, including applications to computer engineering.	m. Knowledge of mathematics, and basic and engineering sciences, necessary to carry out analysis and design appropriate to computer engineering.	n. Knowledge of discrete mathematics.
Math 6A														I
EECS 12 Introduction to Programming		I	I											
EECS 20 Computer Systems and Programming in C			I		I									
EECS 31 Introduction to Digital Systems	I	I	I		I									R
EECS 31L Introduction to Digital Logic Laboratory		R	R		R	I	I		I	I	I			R
EECS 40 Object-Oriented Systems and Programming		R	R											
EECS 70A Network Analysis I	R		R											
EECS 70B Network Analysis II	R		R											
EECS 70LB Net. Analysis II Lab		R	R	R	R									
EECS 111 System Software	A	R	R		R			R			R	R	R	R
EECS 112 Organization of Digital Computers		R		R	R	R		R						
EECS 112L Architecture Lab		R	R	R	R	R			R					
EECS 114 Engineering Data Structures and Algorithms	R	R	R		R							R		R
EECS 115 Intro to VLSI	R	R	R		A			R	A	A	A	R	A	R
EECS 129/129A Senior Design Project		A	A	A	R	A	A	A	R	R				
EECS 140 Engineering Probability	R	R	R									A	R	A
EECS 150A Continuous-Time Signals and Systems	R		R											
EECS 150B Discrete-Time Signals and Systems I	R	R	R											
EECS 170A Electronics I	R		R						R					
EECS 170LA Elect I Lab		R	R	R					R	R				
EECS 170B Electronics II		R	R	R										

EECS 170LB Elect II Lab	R	R	R	R		R	R		R					
General education courses								I,R, A						

Note: I = introduced, R = reinforced, and A = assessed.

All POs except (h) are assessed through the course-embedded assessment process. Program Outcome (h) is assessed two ways. The first way is through the students' general education courses. In order to partially satisfy their general education requirements for graduation, each student must pass 11 courses (or 44 quarter units) in the following areas: composition, humanities, social sciences, foreign language, and multicultural and international/global issues. The second way is through the scoring rubric used in the senior design course (explained in more detail, below). The scoring rubric includes an assessment the extent to which students considered realistic constraints in the design of their senior projects, such as economic, manufacturability, health and safety, sustainability, and social and political constraints. A combination of those two metrics are being used to assess the overall achievement of this outcome.

The results of course-embedded assessment are collected at the end of each quarter using the FCARs . These forms were first used during Fall Quarter 2005 and for all core courses in Winter Quarter 2006. Results were reviewed in Spring Quarter 2006 for use in our required ABET interim report. Results included faculty comments and recommendations for courses, in accordance with our assessment plan.

For one course, we were able to create a full cycle of the *assessment-improvement-assessment* process by recreating a FCAR for Spring 2005. The course was EECS 140, Engineering Probability which is normally taken by majors at the end of their junior year. Since EECS 140 had not yet been taught by the time that the interim report was due, we asked the previous year's instructor to fill out the FCAR for Spring 2005. We then used that data for assessing program outcomes (l) and (n). The argument for such an action is that statistics of courses should be self-similar from year to year and thus data from last year would be representative for this year's courses should no changes have occurred. The other reason for employing the FACR from last year is to show a full cycle of the *assessment-improvement-assessment* process.

The results of the EECS 140 FCAR were discussed between last year's instructor, the current year's instructor, and the ABET lead faculty member for Computer Engineering. Based on the findings from the assessment and those discussions, changes were introduced to EECS 140 being offered Spring 2006. Assessment of the course will take place at the end of Spring 2006 and results will be analyzed later this year to see if changes made have actually improved the direct measures assessment scores.

2. Scoring rubric for senior design projects

The second direct assessment measure we have added for measuring program outcomes is a detailed scoring rubric for the senior design project. Senior design courses are an important part of our curriculum; they provide an opportunity for students to integrate what they've learned about engineering science and to apply that knowledge to a realistic problem and design challenge. It also allows them an opportunity to practice the profession of engineering and to build teamwork skills. Since students complete their projects near the end of their senior year, this also is an excellent time to assess a broad range of program outcomes.

Scoring rubrics are scoring guidelines or a scoring matrix that is used to describe, in words, the characteristics of performance at different levels. The rows of the matrix describe what is to be assessed (i.e., the program outcomes), and the columns describe, in words, performance at different levels. For example, a simple scoring rubric for an essay might include descriptions at 4 levels of performance, from 4 = excellent, no grammatical errors, clear organization, good transitions and conclusions, to 1 = very poor, numerous grammatical errors, unclear organization, poor or missing transitions or conclusions.

Scoring rubrics have several advantages. They can be a very useful pedagogical and learning tool when used during instruction. They provide detailed and timely feedback to students, are extremely efficient for grading, and promote consistency in grading (especially important when multiple graders are used, or projects are graded over time). When used during instruction, students have a clear vision of what constitutes acceptable performance at different levels, thus influencing study habits.

Examples of scoring rubrics and more information on how to develop scoring rubrics can be found from a number of sources.[6,7,8]

Our first scoring rubric for senior design projects was developed by the EECS 129 instructor and modified by the ABET lead faculty in order to emphasize the consideration of realistic constraints, as recommended by ABET requirements. The rubric was used to assess the Winter 2005 projects by an ad-hoc faculty committee and with the help of additional computer engineering faculty.

The faculty who used the Winter 2005 scoring rubric commented that it was too complex and tried to measure too many program outcomes. With the assistance of Dr. Judy Shoemaker, who has been advising the School of Engineering on assessment, a new, much simpler scoring rubric was developed, largely based on one developed at the University of Pittsburgh,[10] and used in Winter 2006 for EECS 129 senior design projects. A copy of the revised scoring rubric is located in the Appendix.

The revised scoring rubric uses 14 scales to assess four POs: (b) design and conduct experiments in computer engineering, (c) design a system to meet desired needs, (d) function effectively on multidisciplinary teams, and (g) communicate effectively. For each PO, there was a detailed description of what constituted performance at each of four levels: Exemplary, Proficient, Apprentice and Novice. The standard of performance was selected as scores of 3 or 4; that is, students had to demonstrate achievement at the Exemplary or Proficient levels to be acceptable.

In order to showcase the senior design projects and to facilitate the assessment process, a Project Expo was held for the first time in Winter 2006 during which each student group showcased their senior design projects. A total of 36 team projects were presented. Projects included a robotic arm, a “sun tracker,” a light sensor for window blinds, a safe mousetrap, a voice-activated alarm, a remote door lock, an Ethernet cable tester, a “plant buddy,” and a method for remote habitat maintenance.

The Project Expo was attended by the department faculty. Each faculty member was assigned a subset of the projects to evaluate using the new scoring rubric. Informal feedback collected from faculty as they assessed each project indicated that faculty understood how to apply the scoring rubric and found it to be an extremely efficient method of grading projects. They also liked the detail of the scoring rubric which would provide useful feedback for the students.

Dr. Shoemaker also attended the Project Expo and subsequently made the following recommendations[9] for increasing the value of the scoring rubric when it is used again in 2006-07:

- Faculty should review the scoring rubric again, making sure that program outcomes are adequately assessed and appropriately described by the performance levels.
- The scoring rubric should be shared with students early in the course, so they know what the expectations are for good performance.
- Provide a review session for faculty who will grade the projects, using the scoring rubric with sample projects from last year.
- Invite industry affiliates to attend the Project Expo and to assess the projects using the scoring rubric.
- Check on the consistency of grading between raters (that is, check inter-rater reliability).

Results and Discussion

Table 4 provides a summary of the direct assessment results from 2005-06 using both measures. We first provide some details on how the scores were obtained:

- For EECS 111, the instructor assessed the CO's in the FCAR. EECS 111 was used to assess Outcome (a) as follows: we averaged out the individual assessment scores for EECS 111 CO's 1, 2, 3, 4 since they were rated as strongly related to POs 1.
- For EECS 115, we directly assessed the POs (e), (i), (j), (k), and (m) using the FCAR.
- For EECS 129, we used the scoring rubric described above to assess POs (b), (c), (d), and (g). We also used an in-class quiz and/or homework to assess PO (f).
- EECS 140 was used to assess POs (l) and (n). We used an approach similar to that used for EECS 111. PO (l) was assessed using all three CO's in EECS 140. However, not all three CO's were rated strongly affecting PO(n): CO1 was strongly related to PO (n) but CO2 and CO3 has a medium level relationship. Thus we applied a weighted sum where CO2 and CO3 each had 2/3 the weight of CO1.

Table 4. Direct Assessment Results for 2005-2006

Program Outcomes	Course-embedded Assessment (FCARs) *(Direct)*	Senior Design Project Rubric *(Direct)*
(a) An ability to apply knowledge of mathematics, science, and engineering.	79.8%	
(b) An ability to design and conduct experiments as well as analyze and interpret data		86.0%
(c) An ability to design a system to meet desired needs.		86.0%
(d) An ability to function on multidisciplinary teams		100%
(e) An ability to identify, formulate, and solve engineering problems.	82.0%	
(f) An understanding of professional and ethical responsibility.	87.5%	
(g) An ability to communicate effectively.		93.0%
(h) A broad education necessary to understand impact of engineering solutions in a global/societal context	93.0%	
(i) Recognition of the need for and ability to engage in lifelong learning.	91.0%	
(j) Knowledge of contemporary issues.	65.0%	
(k) An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.	86.0%	
(l) Knowledge of probability and statistics, including applications to computer engineering.	70.3%	
(m) Knowledge of mathematics, and basic and engineering sciences, necessary to carry out analysis and design appropriate to computer engineering.	74.0%	
(n) Knowledge of discrete mathematics.	72.8%	

Using a performance standard of 80%, as determined by the faculty, the Direct Assessment results shown in Table 4 indicate that for 10 of the 14 outcomes 80% or more of students passed the performance standards (rounding up for 79.8% for Outcome (a)). Only POs (j), (l), (m) and (n) had less than 80% passing rate. Those Program Outcomes are discussed next.

For PO (l) and PO (n) the scores were discussed between the 2005 instructor and the 2006 instructor of EECS 140 and a set of course modifications was established for the Spring 2006 offering of the course. As a result, the following course changes were made:

1. Added an outcome "summarizing data using simple statistics" to the list of course outcomes. This will cover the statistics component mentioned in PO (l).

2. Use demos (in Java and Matlab) in the lectures and the discussion sessions to improve students' understanding of random variables via computer simulation.

3. Put more emphasis in teaching and practicing through exams the materials related to CO2 (random variables).

At the end of Spring quarter 2006, the instructor will use direct measures to assess PO (l) and (n) and compare the results to Spring 2005.

POs (j) and (m) were assessed using EECS 115 (VLSI Design) and had scores lower than 80%. As a result, the instructor has been asked by the ABET lead faculty to propose modifications to the course in Fall 2006 in order to improve the scores. The recommended modifications include:

1. Adding a lecture on Design Methodologies and its evolution over the past decade as well as discussing future directions (PO (j)).

2. Adding lecture material on the International Technology roadmap for Silicon (ITRS) that forecasts the IC scaling into the future. (PO (j)).

3. Adding more design examples in the lecture material as well as the discussion section. Also adding more design-related questions in the midterm and final exam (PO (m)).

At the conclusion of the Fall quarter, 2006 the instructor will use direct measures to assess POs (j) and (m) and compare the results to Fall 2005.

Updating Program Outcomes

Our list of program outcomes continues to be developed and updated with input from key constituents. Our key constituents are:

- Practitioners in industry (including alumni),
- Practitioners in industry who supervise our alumni, and
- Faculty from engineering graduate programs who advise our alumni.

One survey was conducted by the Student Affairs Office during March 2006. This survey was distributed to industry representatives and asked about the appropriateness of the program outcomes. Fourteen out of 48 questionnaires were completed and returned for a response rate of 29%. Overall, respondents rated 13 of 14 program outcomes as appropriate for the Computer Engineering program.

The item that received the lowest rating was “Students will have knowledge of contemporary issues” (PO (j)). Past experience has shown that due to the lack of context for that survey item, respondents can be confused about which “contemporary issues” computer engineering students are expected to know; that is, whether the outcome pertains to contemporary issues in the discipline or simply what is being reported in the news. This survey item should be revisited to provide some context to survey respondents and is actually stated more specifically in the program objectives. Indeed, outcome (j) is too broad in its current form. It will be reviewed by the curriculum committee which will proposed a new wording that is more specific to Computer Engineering and more appropriate in light of the new ABET criteria..

Survey respondents' recommendations were mostly directed toward the computer engineering curriculum itself. One specific recommendation on the outcomes themselves questioned whether it was realistic to expect that graduates "be able to design a system to meet desired needs." This respondent thought it would be more realistic to expect an individual with a B.S. in Computer Engineering to be able to *understand* a system due to a lack of interdisciplinary knowledge, suggesting that the focus should instead be on *designing a component or process* and only expect an *understanding of the system* in which these are incorporated (italics added by authors). The results of the direct assessment of this outcome should provide some insight as to whether students are able to perform design at that level. If so, this comment can be put aside.

Using the Senior Design Course for Partial Program Outcome Assessment.

In the past, our senior design projects/capstone courses have lacked sufficient attention to the wide range of considerations specified by ABET. While we have always focused on the considerations of health and safety as well as manufacturability, we have not emphasized the other considerations.

To study the problems identified, we formed a faculty committee to review the senior design projects from 2004-05. We looked specifically at the constraints and standards addressed in those reports and checked the overall quality of the reports, including the analytical analyses. We found that the scoring rubric used in 2005 was not a good fit with the revised program outcomes and we found that a better assessment of some of these outcomes would be the courses (using course-embedded assessment) rather than the senior design project. Based on that, we developed a new scoring rubric (2006) to evaluate program outcomes (b), (c), (d), (e), (g), (h) and (i), as well as the extent to which students considered a majority of relevant design constraints (section 3.4 in the rubric). Outcomes not listed here are being assessed as part of course-embedded assessment process.

Summarizing the Winter 2005 results, only program outcomes (b) and (d) were satisfied by over 70% of the students, while the other outcomes were deemed not satisfied. In particular, the committee noted that less than 30% of the projects considered realistic constraints from the set of constraints required by ABET.

The findings above were communicated to the instructors in charge of the course for Winter 2006. The instructors were asked to do the following:

- Require that students consider realistic constraints in their projects, specifically to address a majority of the following considerations: economic, environmental, sustainability, manufacturability, health and safety, as well as social and political constraints. Students were asked to include in their design process as well as in their design documentation how they addressed the relevant constraints.
- Include discussions and assessment methods for ethical and professional issues as they relate to engineering in their course syllabi.

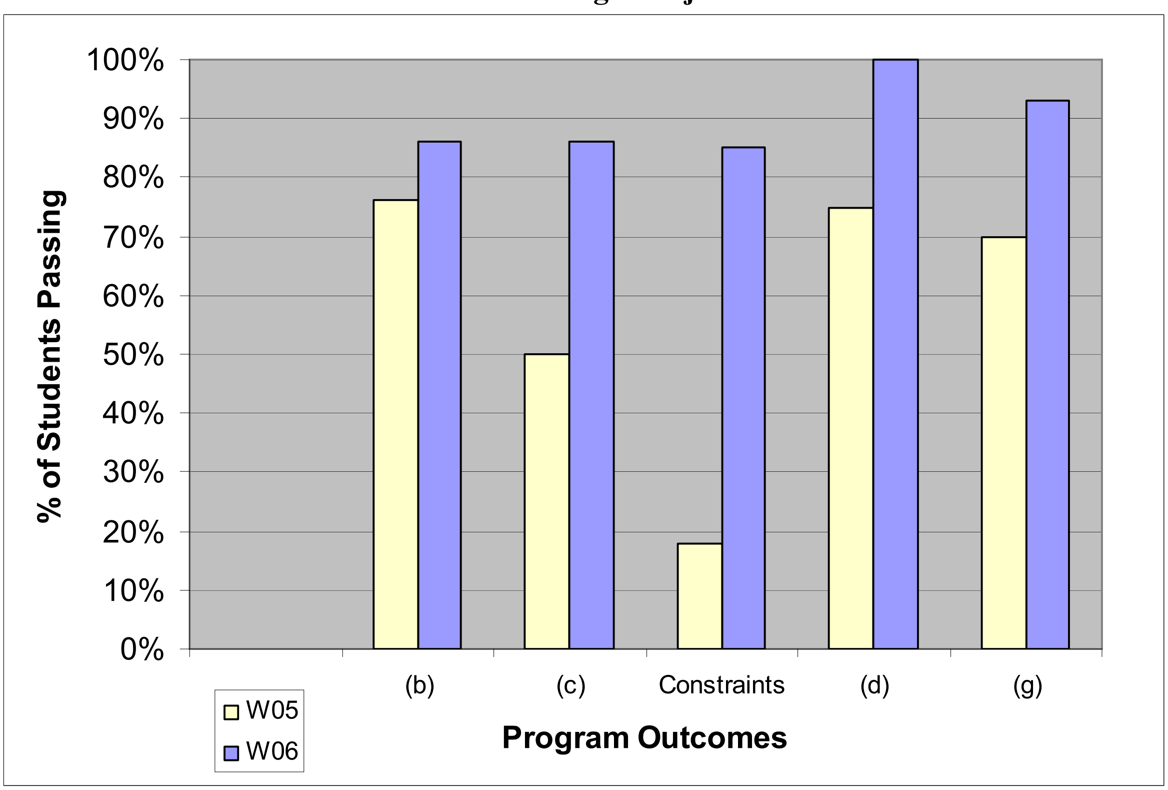
- Prepare students to participate in a project expo at the end of the quarter during which their final projects will be assessed by the departmental faculty.

Every senior design project is now expected to cover all of the following considerations: environmental, sustainability, manufacturability, ethical, health and safety. In fact, this is now a part of the scoring rubric that we have adopted for our senior design projects.

During the Project Expo which took place in Winter 2006, faculty were specifically asked to pay special attention to how well student projects considered realistic constraints in their design process. Indeed a special section in the rubric was broken down to the individual constraints and assessed separately by the faculty evaluators (sub-outcome c.4 or section 3.4 in the 2006 rubric). The faculty assessment was based on: (1) the final project reports, (2) the presentation, (3) the demonstration and (4) interview with the project teams.

The results of the Winter 2006 assessment were compared to those of Winter 2005 and shown in Figure 1. Thanks primarily to the curricular changes introduced in the course in Winter 2006, a vast improvement in the percentage of students passing the performance standards as observed in Figure 1 which compares the assessment results between 2005 and 2006. We note that over 85% of student project considered realistic design constraints. Furthermore, the results show that Program Outcomes (b) (c) (d) and (g) were satisfied by over 85% of the students. ***In particular, we note that while less than 20% of the 2005 projects assessed properly considered realistic design constraints, over 85% of the projects assessed in 2006 considered those same constraints properly.***

Figure 1. Comparison of Assessment Results between 2005 and 2006 for the Senior Design Project Course



Future Improvements for EECS 129, Senior Design Project

Planned for 2006-2007 academic year are several modifications to EECS 129, the senior design experience. Specifically, we have made the following major improvements:

1. **Two quarter sequence:** EECS 129 is now split up into two quarters: EECS 129A is offered in the Fall quarter 2006. During that quarter, students will identify, formulate and plan the design project as well as initiate any parts procurement needed. They will also be lectured on design aspects, especially with realistic constraints, ethics and professionalism and project management. In Winter 2006, students taking 129B will execute the plan developed in 129A, write a final report, and participate in the Project Expo.

2. **Individual faculty involvement:** While an instructor is still involved in the overall "timekeeping" of the project course, each student project will be assigned to a faculty member who will serve as consultant and advisor to the project.

3. **Industry involvement in EXPO:** Given the success of the 2006 project EXPO, we are planning on making this event a permanent component in the senior's design experience. In addition to faculty, we plan on inviting colleagues from industry to attend, critique and participate in the assessment process that takes place at the EXPO. In the future, we will solicit industry participation earlier in the process by asking for input on projects of interest to industry as well as regular monitoring of those project's progress during the two quarters. Their participation will also help us enforce a more prevalent awareness of industry needs as well as improve the usage of standards in the projects.

Conclusion

In this paper we presented a program outcome assessment process that was developed in order to fulfill the ABET Criterion 3 requirements. We have implemented the process in our Computer Engineering Degree Program curriculum and have shown that it is an effective method of ensuring that students going through the program have indeed achieved the stated program outcomes. The process in continual and will allow us to keep improving our course offerings, overall curriculum and the program outcome assessment process itself.

Bibliography

1. Suskie, L. 2004. Assessing student learning: A common sense guide. Bolton, MA: Anker Publishing.

2. Yamayee, Z. A., & Albright, R. J. 2006. Multiple direct and indirect assessment methods are "a must" for a successful ABET visit and continuous quality improvement. Paper presented at the 2006 Best Assessment Processes VIII Symposium, Rose-Hulman Institute of Technology, Terre Haute, Indiana.

3. Rogers, G. 2006. Direct and indirect assessments: What are they good for? Assessment 101 Column, ABET CommunityMatters Newsletter, August 2006, p. 3. Retrieved January 5, 2007 from http://www.abet.org/Linked%20Documents-UPDATE/Newsletters/06-08-CM.pdf.
4. Allen, M. J. 2004. Assessing academic programs in higher education. Bolton, MA: Anker Publishing.

5. Electrical & Computer Engineering and Computer Science Department, Ohio Northern University. Faculty course assessment report (FCAR) instructions. Paper presented at the 2005 Best Assessment Processes VII Symposium, Rose-Hulman Institute of Technology, Terre Haute, Indiana.

6. Moskal, B. M. 2000. Scoring rubrics: what, when and how? *Practical Assessment, Research & Evaluation*, 7 (3). Retrieved October 10, 2006 from http://PAREonline.net/getvn.asp?v=7&n=3.

7. Moskal, B. M. 2003. Recommends for developing classroom performance assessments and scoring rubrics. *Practical Assessment, Research & Evaluation*, 8(14). Retrieved June 11, 2006 from http://PAREonline.net/getvn.asp?v=8&n=14.

8. Estell, J. K., & Hurtig, J. 2006. Using rubrics for the assessment of senior design projects. Paper presented at the 2006 Best Assessment Processes VIII Symposium, Rose-Hulman Institute of Technology, Terre Haute, Indiana.

9. Shoemaker, J. 2006. Computer engineering senior design expo. Presentation for Computer Engineering Faculty Treat, University of California, Irvine.

10. Electrical Engineering Program. 2005. Scoring Rubric for Senior Design Projects. Self-Study Report, June 29, 2005. Pittsburgh, PA: University of Pittsburgh. Retrieved January 10, 2006 from http://www.engr.pitt.edu/electrical/faculty-staff/boston/EE_2005_Self_Study.pdf.

Appendix 1 – Scoring Rubric used in Winter 2006

EECS 129: Scoring Rubric for Senior Design Projects

Group members: ____ **Qtr/Year:** ______ **Project:** ______________________________________

Outcomes	Exemplary 4	Proficient 3	Apprentice 2	Novice 1	Score *
Program Outcome 3: An ability to design a system to meet desired needs.					
3.1 Identify specific project objectives based on project and client requirements.	All important project objectives are identified.	All important objectives are identify but 1 or 2 minor ones are missing.	Most objectives are identified but at least 1 or 2 important ones are missing.	Most or all important objectives not identified	
3.2 Gather and use relevant background data	All relevant information is obtained and used to support design recommendations.	Sufficient information is obtained and used to support design recommendations.	Some information is obtained but more is needed to support design recommendations.	No significant background information is gathered.	
3.3 Generate and analyze alternative solutions by synthesizing and applying appropriate engineering knowledge	Three or more alternative solutions are considered; each is correctly analyzed for technical feasibility.	At least 3 alternative solutions are considered; analysis is complete but contains minor procedural errors.	At least 2 alternative solutions are considered; analysis contains minor conceptual and/or procedural errors.	Only one solution considered; no optimization included; better solutions were available.	
3.4 Consider all relevant constraints if applicable					
Economic	All economic issues included; computations are correct.	Important economic issues correctly included; minor ones may have been ignored.	One or more important considerations ignored; but computations correct.	Most if not all economic considerations ignored; and/or computations flawed.	
Environmental, sustainability	Environmental factors, sustainability adequately considered.	Important issues considered; certain minor ones may have been ignored.	One or more important issues ignored.	Most if not all issues ignored.	
Manufacturability	If applicable, thoroughly considered.	Important issues considered if applicable.	Certain important issues ignored.	Most if not all important issues ignored.	
Ethical, health and safety	Ethical issues including safety of public and work health considered.	Primary issues considered; one or two secondary issues may have	Most but not all important issues considered.	Most if not all important issues ignored.	

		been ignored.			
Social/political	Problem placed in appropriate social/political context; all issues considered.	Primary issues considered; some secondary issues may have been neglected.	Most but not all primary issues considered.	Most if not all primary issues ignored.	
3.5 Choose the best solution based on technical and economic criteria and considering other relevant constraints	Best solution is recommended based on stated criteria and constraints.	Reasonable solution is recommended; other alternatives should have been developed and analyzed.	Satisfactory solution is recommended; better solutions were available and should have been considered.	Only one solution considered; better solutions were available; most constraints ignored.	
Program Outcome 2: An ability to design and conduct experiments as well as analyze and interpret data, if applicable					
2.1 Quality of experimental design	Experiment is well-designed and conducted and documented in a professional manner.	Well-designed experiment with minor exceptions; conducted and documented professionally	Design adequate, but not outstanding; lacked some controls; information reliable, but not definitive.	Poor design; information of little value.	
2.2 Appropriate statistical analyses used and results correctly interpreted.	Appropriate statistical analyses used; proper assumptions made; results correctly interpreted.	Appropriate analysis and interpretation with a few minor exceptions.	Analysis and/or interpretation contain a few serious flaws.	Analysis and resultant interpretation seriously flawed or non-existent.	
4. Function on multidisciplinary teams					
4.1 Team demonstrates cooperation; all team members participate.	Team members cooperate with each other; no one member dominates.	Team members cooperate with each other, but 1 or 2 members dominate.	Team members cooperate somewhat with each other; a few members dominate.	Team members do not cooperate with each other; 1 or 2 members dominates.	
7. Communicate effectively					
Goals of project, methods and solutions clearly articulated and technical terms used appropriately.	Goals, methods and solutions presented clearly; all technical terms used appropriately.	Goals, methods and solutions were presented clearly; most technical terms used appropriately.	Presentation of goals, methods and solutions was not very clear, difficult to follow; some technical terms not used appropriately.	Presentation of goals, methods and solutions was lacking clarity, very difficult to follow; most technical terms not used appropriately.	
Presentation meets professional standards , with balance of text	The presentation had a good balance of text and visual graphics. A variety of graphics were	The presentation had a good balance of text and visual graphics. Some variety of graphics	The presentation was dominated by either text or graphics. Some variety of graphics	The presentation was dominated by either text or graphics; little	

and graphic materials.	used appropriately.	were used.	or graphics misused.	if any variation in graphics used or graphics misused.	

* Indicate if category is not applicable (N/A).

Note: Adopted from scoring rubric developed by University of Pittsburgh, Electrical Engineering Program.[10]

Comments for the Project Team:

Comments regarding the Scoring Rubric:

Appendix 2 – Example of a Completed FCAR

FACULTY COURSE ASSESSMENT REPORT
COMPUTER ENGINEERING PROGRAM

EECS 115 Intro VLSI

The school's outcome assessment process requires direct measures of achievement of student performance. Departments have various opportunities to directly measure student performance, but demonstrated achievement of outcomes in courses provides detailed and localized evidence of student performance as they progress through their academic program. Provide information on the academic term (quarter) that the course was taught, the instructor's name and the grade distribution for the class. For each course outcome below, indicate how it was assessed. For example, if problems on exams were used, identify which problems on which exam were used. Identify any homework assignment numbers, lab reports, and so on that are used for assessment. For each assessment method, report the average score and the percent of enrolled students who met the performance standards for that outcome. If the outcome was not taught or not assessed, please include that information.

Academic Term:	Fall 2005	**Instructor:**	Kurdahi

Final course grade distribution:

A	B	C	D	F	P	N/P	I	W	NR	Total

Program Outcomes:

PO(a). An ability to apply knowledge of mathematics, science, and engineering.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard Used***

PO(b). An ability to design and conduct experiments as well as analyze and interpret data.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard Used***

PO(c). An ability to design a system to meet desired needs.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***

PO(d). An ability to function on multidisciplinary teams.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***

PO(e). An ability to identify, formulate, and solve engineering problems.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***
HW1 through HW6	89/100	91%	70%
Midterm P5	21.52/25	82%	18/25
Midterm P6	2.46/5	57%	2/5
Final Q5	6.22/10	78%	6/10
Final Q6	5.4/10	60%	5/10

PO(f). An understanding of professional and ethical responsibility.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard Used***

PO(g). An ability to communicate effectively.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard Used***

PO(h). A broad education necessary to understand impact of engineering solutions in a global and societal context.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***

PO(i). Recognition of the need for and ability to engage in lifelong learning.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***
Project	179.55	91%	150/200

PO(j). Knowledge of contemporary issues.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***
Midterm P1	8.1	81%	7/10
Midterm P2	14.79	70%	14/20
Midterm P3	5.63	20%	10/20
Final Q1	3.71	60%	3/5
Final Q2	1.78	47%	2/5
Final Q3	3.35	74%	3/5
Final Q7	4.78	96%	3/5
Final Q8	3.28	63%	3/5
Final Q10	2	48%	2/5
Final Q12	5.65	71%	5/9
Final Q13	2.83	74%	2/6

PO(k). An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***
HW1-5	89	90%	70/100
Final Q9	4.96	65%	5/10
Project	179.55	91%	150/200

PO(l). Knowledge or probability and statistics, including applications to computer engineering.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***

PO(m). Knowledge of mathematics, and basic and engineering sciences, necessary to carry out analysis and design appropriate to computer engineering.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***
Midterm P4	10.46	86%	7/12
Midterm P5	21.52	82%	18/25
Midterm P6	2.46	57%	2/5
Final Q4	7.99	75%	7/10
Final Q11	7.15	68%	6/10

PO(n). Knowledge of discrete mathematics.			
Assessment Method	***Average Score***	***Percent Meeting Performance Standards***	***Performance Standard***

Summary of this quarter's assessment results:

Assessment showed that >80% of students met POs (e), (g), (k). Less than 80% of students met the standards for Pos (j) and (m).

What changes did you make in this course based on previous assessment results?

More emphasis on direct measures and establishing Midterm and Final questions that attempt to measure POs in a more direct fashion.

What recommendations do you have for improving the course the next time it is taught?

4. Adding a lecture on Design Methodologies and its evolution over the past decade as well as discussing future directions (PO(j)).

5. Adding lecture material on the International Technology roadmap for Silicon (ITRS) that forecasts the IC scaling into the future. (PO(j)).

3. Adding more design examples in the lecture material as well as the discussion section. Also adding more design-related questions in the midterm and final (PO(e) and PO(m))

Do you have any recommendations regarding improving or changing the course objectives and/or course outcomes?

What recommendations do you have, if any, regarding prerequisite courses or other ways to improve student preparation for this course?

EECS COURSE NO.

Any other recommendations and/or comments?

Appendix 3 – Sample Course Outline

EECS 129A COMPUTER ENGINEERING SENIOR DESIGN PROJECT
(Required for CpE)

Catalog Data: EECS 129A Computer Engineering Senior Design Project (Credit Units: 2) Conception, planning, implementation, programming, testing of an approved project. Options include: parallel processing, VLSI design, microprocessor-based design, among others. Prerequisite: senior standing. In-progress grading. Formerly EECS129. (Design units: 2)

Textbook: Varies.

References: http://e3.uci.edu/

Coordinator: _______________

Course Objectives: Introduce the students to emerging computing paradigms as the basis for project platforms [B, C]. Expose students to interdisciplinary application topics and encourage articulation of technology impact [B, C, E].Exercise in project planning ranging from draft project proposal to implementation strategy, practice, oral and visual presentation of project work and research finding [A, B, D]. Foster opportunity for working in small teams, working with other teams, and experts in different application areas [B, D, E]. Application of students' knowledge from previous courses to solving new project problem and to disseminate technical documentation [A, B].

Course Outcomes: Students will: Read API documentation for application programming. Read data sheets for component interfacing. Integrate hardware and software components into a complete plan. Define a project timeline and document progress of the technical work with oral presentation. Consider the economic, environmental, social, political, ethical, health and safety impact of their final product, as well as study its manufacturability, and sustainability.

Prerequisites By Topic: System programming. Computer architecture. Network analysis

Lecture Topics: Project proposal. (2 weeks)Project checkpoint. (2 weeks)Project reports. (2 weeks)

Class Schedule: Each class meets 1 hour for a lecture per week for 10 weeks, 1 hour of discussion per week for 10 weeks and students are assigned to a 3 hour lab session per week.

Computer Usage: Any computer capable of running software development tools for the Palm OS platform, Macintoshes recommended.

Laboratory Projects: Projects can draw from (but are not limited to) one of the topics below: Hand-held logic analyzer with the Palm Hand-held oscilloscope with the Palm Wireless sketchpad involving a Palm and a workstation. Other data acquisition and actuation devices. Goals: Define an application problem in one of the topics with sufficient technical challenges. Demonstrate analytical thinking and apply skills learned from previous courses. Research prior work and build on existing infrastructure.

Professional Component: Contributes toward the Computer Engineering Topics Courses and Major Design experience.

Relationship to Program Outcomes: This course relates to Program Outcomes b, c, d, e, f, g, h, i, and j as stated at: http://undergraduate.eng.uci.edu/degreeprograms/computer/mission

Design Content Description

***Approach**:* Emphasis is placed on planning, research, problem definitions and lectures that set the context for he projects while defining the common platform, and implementation of work.

***Lectures**:* 40%
***Laboratory Portion**:* 60%

Grading Criteria:

Quality of report including weekly progress report, bibliography::	35%
Quality and novelty of work:	45%
Class participation	20%
	100%

Prepared by: ________________ **Date**: July 2006

CEP Approved: Fall 2004