Getting More Learning From Labs - Six Principles to Build Understanding and Skill

Dr. Clark Hochgraf, Rochester Institute of Technology (CAST)
Prof. Richard Cliver, Rochester Institute of Technology (CAST)

Richard C. Cliver is an Associate Professor in the department of Electrical, Computer and Telecommunications Engineering Technology at RIT where he teaches a wide variety of courses both analog and digital, from the freshman to senior level. He was the recipient of the 1998 Adjunct Excellence in Teaching Award, the recipient of the 2002 Provost's Excellence in Teaching Award and a finalist in the 2009 Eisenhart Excellence in Teaching Award. In addition, he works part-time for Eastman Kodak as a Senior Design Engineer and is a TAC of ABET commissioner.

Dr. David S Martins, Rochester Institute of Technology

David S. Martins is Associate Professor and director of the University Writing Program at Rochester Institute of Technology. His article on the use of scoring rubrics won the Best Article of the Year 2008 in Teaching English in the Two Year College, and his articles have appeared in Communication Studies, the Journal of Medical Humanities, and in edited collections. He works with faculty across the curriculum to integrate writing into their design of high quality learning environments.

Getting More Learning From Labs – Six Principles to Build Understanding and Skill

Abstract
This paper describes an approach to increasing student learning in engineering labs based on recent education research [1-4] and industry needs [5]. A set of six guiding principles for designing lab exercises are presented. The principles are illustrated in the context of an undergraduate laboratory course in Digital Signal Processing where the pedagogy has evolved over four years. Changes to the labs were driven by the desire to motivate students' engagement, improve their understanding of fundamental concepts, and help them gain competence and confidence in implementing a working Digital Signal Processing system in hardware.

Introduction
In engineering education, and engineering technology in particular, laboratory courses play a central role in helping students learn challenging concepts and develop practical skills. Labs are often one of the college learning experiences that students enjoy the most. As lab course instructors, we strive to create learning environments in which students experience maximum benefit from their labs. We recognize the complexity of creating rich environments for learning, and know the enormous amount of work it takes to create and run high quality laboratory experiments. There are ample opportunities for students to stray off-course, for experiments to go wrong, and for students to misunderstand which aspects of the lab are most relevant and require extra attention, and which aspects are not critical.

In this paper, we present a set of guiding principles for the design of lab experiments to promote more effective learning. The principles are based on education research [1-4, 8, 9] and are illustrated in the context of a redesigned senior-level digital signal processing lab course in an undergraduate engineering technology program. The principles are given in figure 1 and discussed in separate sections below. We begin with a description of the Digital Signal Processing (DSP) lab course to which the principles were applied, followed by discussion of each principle, and conclude with an assessment of the outcomes achievement in the redesigned lab.

Overview of DSP lab course
The guiding principles were applied to an existing DSP course where a traditional set of lab exercises explored analog to digital conversion, sampling, DSP hardware, FIR and IIR filters, and FFTs. To a large extent, before we redesigned the course, the labs were isolated illustrations of concepts from the lecture and of implementation methods, without an overt progressive

Guiding principles for redesign of labs:

1) activate students' self motivation
2) scaffold labs to develop component skills first and then integration skills
3) provide a framework for students to organize new knowledge
4) manage cognitive overload
5) develop practical universal implementation skills
6) use "writing to learn" to promote deeper understanding

Figure 1. Principles for redesigning labs to improve learning.

connection between labs. To the extent a connection between labs existed, it was not clearly communicated to the students how the laboratory exercises were a path to build skills they would need in their career. As a result, many students did not engage deeply in trying to understand the concepts, implementation methods, or the limitations of the implementation employed in the labs.

The redesigned course begins with foundational labs that build component skills such as measuring execution time, memory usage, and the effects of roundoff error in DSP algorithms. The later labs in the redesigned course focus on using component skills to design, implement and debug FIR and IIR filters. The labs culminate in a final project that promotes the development of integration skills, as students must determine when, where, and how to apply component skills appropriately.

In addition to changes in the lab activities, the new format for each lab report now highlights its relation to the students' final project. Traditional reports are replaced by incremental reports that ask students to produce high quality figures to be used in the final report. Each week the students write a few paragraphs in response to "write to learn" questions about their figures and about the code they have written or data they have collected. This practice requires students to think critically about the validity of their results, an essential engineering skill.

Principle 1: Activate students' self motivation

Student learning is strongly influenced by motivation. Motivation often results from how much value students perceive they will get from the class, their expectation of how likely they are to succeed in the class, and whether or not the environment is supportive [1].

In the revised DSP course, three different methods were used to activate students' motivation for learning: bringing the end forward, choosing an authentic problem, and relating in-class work to career goals.

First, at the start of the digital signal processing course, students were told they would complete a final project where they must both design a digital signal processing system and demonstrate that

From the cover page of syllabus:

At the end of this course:

You will have demonstrated that you can design a digital signal processing filter that reduces noise and demonstrated that you can make that filter run in real time on digital signal processor / microprocessor hardware.

You will have written a paper for your portfolio in the IEEE journal format, which you can use in job interviews over the next 10 years to show your knowledge of DSP techniques.

Figure 2. Taken from the cover page of the DSP course syllabus, this illustrates the overt communication to students about the desired learning outcomes and purpose of assignments.

system running in real time on digital signal processor hardware. It was explained at the onset that understanding each of the early labs would be instrumental to their success on the final project and to getting a good grade in the course.

Second, the final project topic was chosen to be an authentic application of digital signal processing techniques with benefits to society. . To motivate a larger percentage of students, a project topic in the biomedical field was chosen [6,7]. In the revised digital signal processing course, students build a biomedical monitoring device to detect pneumonia in infants and young children. The real world need for the device is presented by showing students several videos and papers describing infant respiratory distress along with World Health Organization statistics on the number of pneumonia cases and corresponding mortality rates worldwide. Additional papers given to the students describe the inadequacy of existing methods of detecting pneumonia, such as X-rays, due to patient risk and high cost.

As a third motivational factor, students document their final project in journal paper format so that it can become part of their permanent portfolio; such a portfolio can be used not just at graduation, but also for future job interviews. Students are told that while they may already have a job lined up, it is likely they will change jobs and that their paper may be very valuable in a future interview. As a result of this future value, students are encouraged to put extra effort into making their final report polished, professional, and effective at demonstrating their skills. The importance placed on the journal paper is conveyed by placing a note about it on the first page of the syllabus as shown in figure 2.

Principle 2: Scaffold labs to develop component skills first and then integration skills

Scaffolding is an instructional approach that promotes learning by enabling novices to engage in challenging learning activities by compensating for gaps in their knowledge. Examples of

Sequence of DSP Labs:

Number / Topic		Duration	Pages in Handout
Lab 1	Intro to DSP hardware, IDE, Statistics	(1 week)	13
Lab 2	Analog to Digital Conversion, SNR, Dithering	(1 week)	18
Lab 3	Datatypes, Memory Usage, Execution Speed	(2 weeks)	17
Lab 4	Convolution, FIR Filter Design	(1 week)	8
Lab 5	Frequency Response Analysis of FIR, IIR Filters	(2 weeks)	6
Lab 6	IIR Filter Design	(1 week)	4
Project	Respiratory Rate Monitor	(2 weeks)	4

Figure 3. Sequence of labs in DSP course, duration of each lab, and number of pages in lab handout.

scaffolding techniques include providing procedural guidelines or a checklist for the students to follow, giving out partially solved problems, and having an expert (e.g. the professor) describe out loud their thought process as they solve a problem [7]. The theory of scaffolding is that students who are initially assisted by an expert can develop more quickly than they would on their own [8]. As the student acquires knowledge, the scaffolding is progressively removed, giving more responsibility and ownership to the learner. In the end, the student is able to perform independently at a high level.

In the revised digital signal processing course, as presented in figure 3, the lab assignments are deliberately scaffolded: each subsequent lab is designed for students to apply knowledge gained in the previous week's lab as a means to provide repetition and reinforcement. In the early labs, students receive generous amounts of guidance on how to succeed. The lab handouts are longer and provide a step-by-step procedure, with detailed photos of the equipment setups and sample results from the experiments as shown in figure 4. Students are also given expert tips on how to approach reading the lab handout as shown in figure 5. The early labs are designed so that students can follow the instructions, feel a sense of self-efficacy, and complete the lab successfully.

As time progresses, the lab handouts become shorter, as seen figure 3. The later labs provide less procedural guidance and rely on the student to play a larger role in making choices relevant to the lab exercise. Students begin having to write their own code, based on code and algorithms they have learned in earlier labs.

As a side note, while the student handouts are getting shorter in the later labs, the instructor's manual for the labs needs to get longer and more detailed. One of the challenges in a lab course is to make it possible for other instructors to be able to run the course in a similar fashion. It takes a great deal of specific know-how to guide the students through the later stages of the project. For anyone other than the person who developed the project, this can be a significant

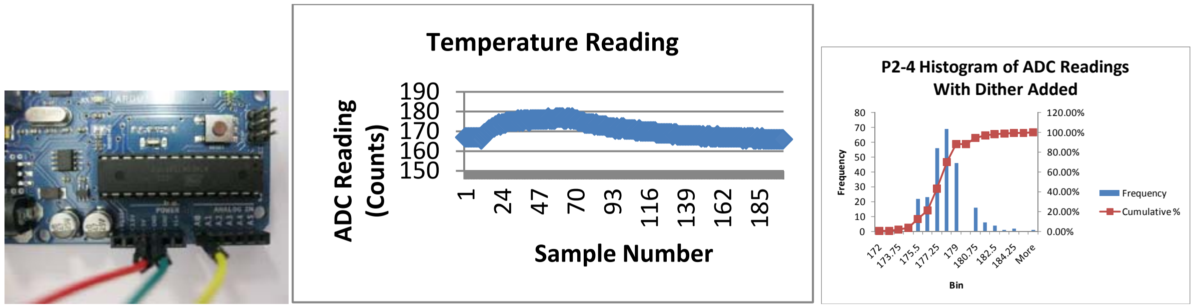

Figure 4. Example of guidance provided in early lab handouts: photos of equipment setup and sample results are provided.

challenge. Keeping good notes and purposely creating an instructor's manual is highly recommended.

During the course, students work in groups to gain peer-to-peer support and avoid getting stuck. Within a lab group, some students may have a better grasp of the material and be able apply their knowledge as well as teach their peers about the subject. Conversation between groups is encouraged. In practice, peer-to-peer and group-to-group communication behavior was frequently observed in the course and benefited both novice and expert students.

How to Succeed With This Lab:

Before executing the procedure, skim-read through the whole lab handout. Skim-read through the ***Write Up*** *section so you know what is expected for the lab report. Work together as a team with your lab partners to solve problems and overcome any snags (e.g. syntax issues with C, fabrication of the sensor leads, writing up an explanation of results).*

The number one problem students face doing this lab:

Mis-wiring the sensor/ not correctly reading the datasheet pinout. Not sure? Ask.

Figure 5. Excerpt from lab handout giving students guidance on how to approach reading the lab handout and avoid common mistakes.

In addition to the knowledge linkages from one lab to the next, each week's lab write-up is structured to provide content that will be used in the final project report. For example, the analysis of quantization noise in lab 2 is a crucial topic in the final project report. The specific graphs produced for the lab 2 write-up can be used directly for the final report. There is both an efficiency of work effort built into this sequence as well as a motivational factor. The students are told throughout the course that every lab report they do will contribute to the final project and that they should take care to make high quality, fully labeled graphs so that they don't have to redo their work at the end. The same line of thinking applies to the questions asked in the lab write-ups. Each question relates to some aspect of the final project and should be considered

How could knowing the execution time for floating point multiplies affect which filter design you choose in this lab?

Why would modeling ADC quantization error as noise be useful, given what you know about how noise sources add?

How could what you know about the memory space requirement for integer versus floating point datatypes be applied to debugging the following non-working code?

Figure 6. Example of "how" and "why" questions used in lab handouts.

relevant and important to understand. For the early lab write-ups, the contribution might be just a specific figure, while for the later labs, the responses to conceptual questions may apply.

As part of the scaffolding and to encourage transfer of content skills, the lab handouts also ask students "how" and "why" questions that help students look for and find connections between what they learned during the last lab and what they are working on for the current lab. If content skills are going to transfer to other labs and be integrated into future problems, it is not enough to scaffold the labs, students must also be given overt cues so that they recognize the linkages between each experiment. Figure 6 provides examples of "how and why questions" used in the lab handouts.

Principle 3: Provide a framework for students to organize new knowledge

Arranging the labs to support the final project goals is not enough. The students must know about this linkage and it must be clearly, consistently, and repeatedly communicated throughout the course. As students begin to realize how each previous lab is an important part of the final project puzzle, their quality of work and attention to task improves significantly.

In the third lab, students are provided a concept map (figure 7) that shows relationships between concepts they have been exposed to and concepts that they will soon learn. Concept maps provide a means for students to organize new knowledge and begin to identify meaningful rather than superficial relationships.

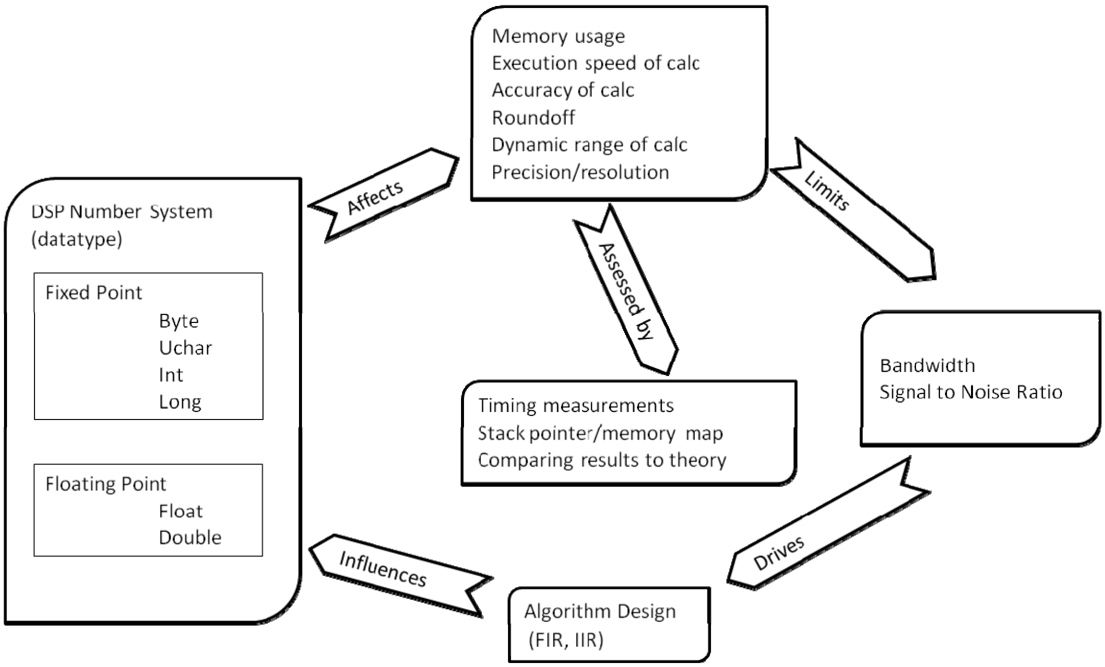

Figure 7. Concept map framework provided to students to help them organize new knowledge and develop relationships between concepts they learn in the labs.

In the present DSP course, students do not create their own concept maps, however, adding a few small concept map exercises is under consideration. Feedback from student course evaluations shows students continue to struggle relating the large number of concepts and procedures they learn to the problems of their final project.

The medical and engineering journal papers that students read to understand the performance specifications of the respiratory rate monitor also play a secondary role. They serve as genre examples of scholarly publications on which students can model their final project report. This provides students a road map for what information must be gathered and how to present it in their final project report.

Principle 4: Manage cognitive overload

When learners are faced with rapidly acquiring a large quantity of new material, they can become overwhelmed and experience cognitive overload. Scaffolded learning, as described above, is one way to prevent cognitive overload. Another way is for instructors to recognize which tasks are relevant to learning and which are distracting, and then systematically remove or reduce distractions.

In the DSP lab course, cognitive overload is a severe challenge. The course is offered on the quarter system as a single required 10 week class for electrical engineering technology students. Prior to the class, students typically have taken one programming course and one microprocessor course. In general, the students are not confident in their coding skills and are not fluent with an integrated development environment or hardware platform.

The original DSP labs in the course used an industry standard, high speed 32-bit floating point DSP development board with a high function integrated development environment and associated tool chain. Several of the early labs were used to bring the students up-to-speed on the hardware, the IDE, and the build process (creation of make files, folder structures and setting paths). Although these activities have proven essential for students to be able to create, compile, and test their own algorithms, such an overview in a 10 week class cuts significantly into the time for learning the fundamentals of signal processing, and diverts students' attention from the learning objectives of the course.

To free up more time for DSP specific topics, the hardware platform was changed in the redesigned course to one with a simpler tool chain, setup and IDE that would still support the learning objectives of the class. The new open-source software/hardware platform was much easier to use, install, and maintain, and at a low enough in cost (<$30) that every student purchased their own system and could take it home. This allowed students extended access to spend more time learning.

By reducing the distractions of setting up paths and licenses, creating make files, and learning a high-function IDE, more time was available for learning the fundamentals of DSP and universal implementation issues.

Principle 5: Develop practical, universal implementation skills

Implementation skills are essential when students are expected to create a working system as part of the lab. Development of implementation skills requires guidance, time, and support from the instructor. One concern frequently raised by faculty members is that students won't learn implementation skills using a simple open-source software/hardware platform. A simple system, however, provides students opportunities to learn practical, universally-applicable implementation skills. Using a simple system in the DSP lab allowed students to experience the same limitations seen in industry where much more advanced processors are used. A real-world FIR filter, while meeting the performance specifications, can easily press up against memory limits and begin to operate in unexpected ways. This behavior was observed when a high performance floating point DSP development system was used in an early version of the course. After converting to the simpler open-source software/hardware platform, students saw the same memory issues and became sensitized to what can go wrong if attention is not paid to memory usage. To some degree, the simpler system made it easier to show that a memory space limitation was the cause of the errors.

In the revised DSP course, students also face a number of universal DSP implementation issues. The list includes the impact of roundoff error on algorithm stability, execution time limitations, memory requirements, and the impact of quantization noise. When students face these issues in the context of their final project, they must apply component skills learned in the earlier labs, e.g. measuring execution time, and thereby taking their understanding to a new deeper level.

When students design an IIR filter that appears to work in Matlab but fails work on the DSP hardware, they encounter a common implementation problem. In the process of debugging the system, students become keenly aware of coefficient roundoff error using single precision floating point numbers. Students who are sensitive to the problem of roundoff error will have an important insight into what drives certain filter-structure design choices.

Instructional design that utilized scaffolding can also be very helpful in teaching students implementation skills. Students can learn a great deal by seeing when the algorithm stops working due to implementation issues. In one lab, students are given a recipe to follow that illustrates how the algorithm is supposed to work and they make measurements. Then the work is extended slightly to show where the algorithm ceases to work as expected. By having students exposed to both working and non-working situations, the students are able to compare and contrast the situations and gain deeper understanding. For example, in analog to digital conversion, quantization noise is reduced by averaging numerous samples. The first part of the lab shows how this works. However, in the next part of the lab, the analog noise in the system is reduced to the point where there is insufficient dithering of the ADC values. Students have to figure out why the averaging system has stopped working. This leads to deeper understanding and appreciation for the limitations of averaging and for having sufficient dithering noise. Later in the final project, students again apply their knowledge to decide whether or not additional dithering noise must be added for their particular system. The answer is not the same for all groups. Additional dithering noise is needed by some groups but not by other groups, depending upon the level of sensor noise.

Principle 6: Use “writing to learn” to promote deeper understanding

Throughout the previous sections, examples of using writing to learn have appeared, for example using “how” and “why” questions to activate students awareness of linkages between labs, concept mapping, analysis of unsuccessful systems, and peer-to-peer and group-to-group communication. Two more examples of using writing to learn are provided in figure 8. As part of the lab write-up, students are asked questions to get them to reflect on how data they have collected, e.g. the data table shown in figure 9, relates to concepts in the course. The questions are structured to have the students responding not to the instructor, who in the student’s eyes probably already knows the answer and doesn’t need a good explanation, but rather to another student, their lab partner who actually needs a good clear explanation. These types of questions are good for helping the instructor determine how well the students understand the material and can be very helpful in identifying misconceptions which need to be addressed.

Q1-7 Your lab partner believes when a person breaths on the temperature sensor (causing the temperature signal to go up and down) that the mean statistic will show this better than the standard deviation statistic. Write a one paragraph response to your lab partner than explains why you think the standard deviation statistic will be more responsive to temperature variations than the mean statistic. Support your argument with either data from your experiments or a rationale based on the formulas for mean and standard deviation.

Q3-4 Your lab partner collected the data in the table 3-4 but doesn't understand what it means. Write a paragraph to your lab partner describing what the data means and what observations you have made about how datatype affects execution speed, memory usage and accuracy. If any of the output values were not accurate, be sure to address that.

Figure 8. Examples of "Write to Learn" questions used in the lab handouts. Q1-7 comes from an earlier lab and represents a leading question. Q3-4 comes from a later lab and is a more challenging open-ended question.

Table 3-4 Convolution memory usage and execution time versus datatype

Convolution Performance when impulse response length=30, xdatalength=100				
	byte	int	long	float
FreeMemory				
microseconds for each new datapoint				
Max update rate (Hz)				
final value of output				

Figure 9. Data table corresponding to "Write to Learn" question.

Assessment

To assess the impact of applying the six principles to the redesign of an existing DSP lab course, several assessment methods were used. First, student ratings of the course and instructor are presented for pre- and post- treatment time periods. Then the results of a student self-assessment survey are presented. Instructor observations of student behaviors and final project performance are also discussed. Finally, the students' written comments from the end of the course evaluations are presented.

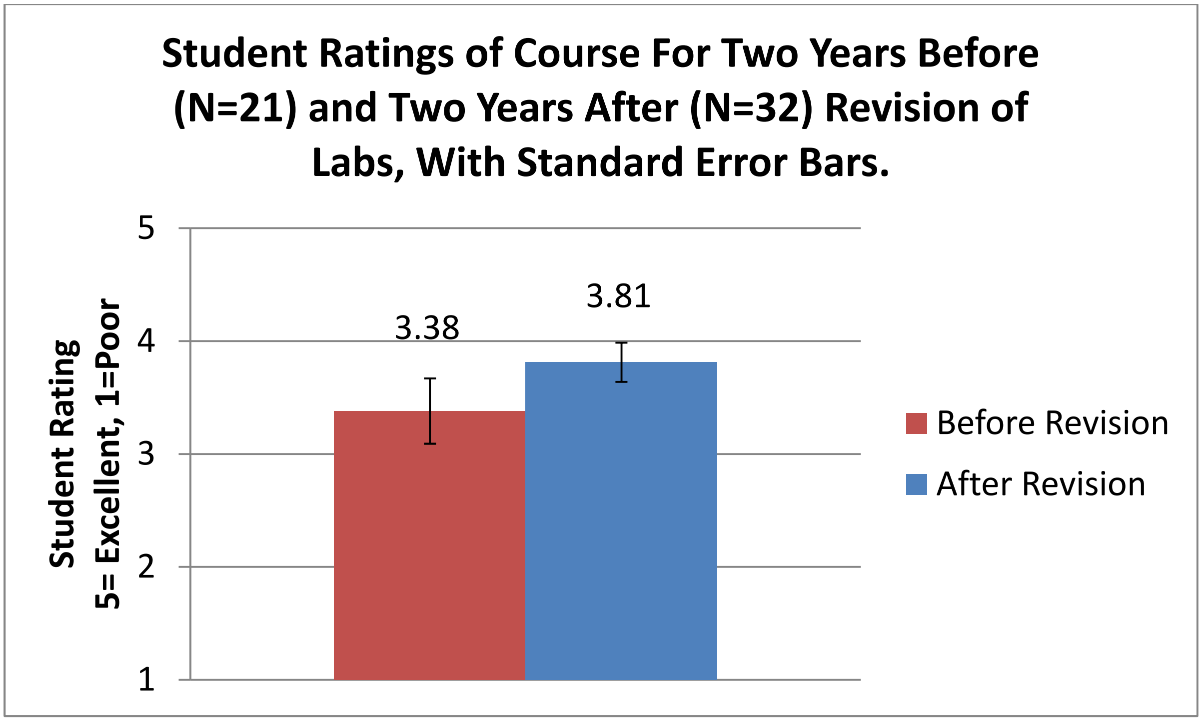

Figure 10. Student ratings of the course before and after revision of the labs.

Four years of student course evaluation data were collected consisting of two years of data before the change and two years after the change. During the entire four years, the course was taught by the same instructor, using the same textbook, in the same quarter (fall) and covering the same lecture topics. The differences in instructional design between the two course have been described above.

The mean values of the students' rating of the course before and after the lab revision are shown in figure 10. The mean rating of the course increased 0.43 points on a 5 point scale. The error bars indicate the standard error of the mean values. The mean values of the students' rating of the instructor before and after the lab revision are shown in figure 11. The mean rating of the instructor increased 0.63 points on a 5 point scale. The error bars indicate the standard error of the mean values.

Students in the redesigned course were asked to complete a self-assessment survey of their learning at the end of the course. The survey questions and results are shown in figure 12. Students felt strongly (93%) that that the project improved their implementation skills. The students were motivated (87%) to pay attention to quality of the work because of the structured formal project writeup that they could use in their portfolio. The students were also proud (87%) of the quality of work they produced.

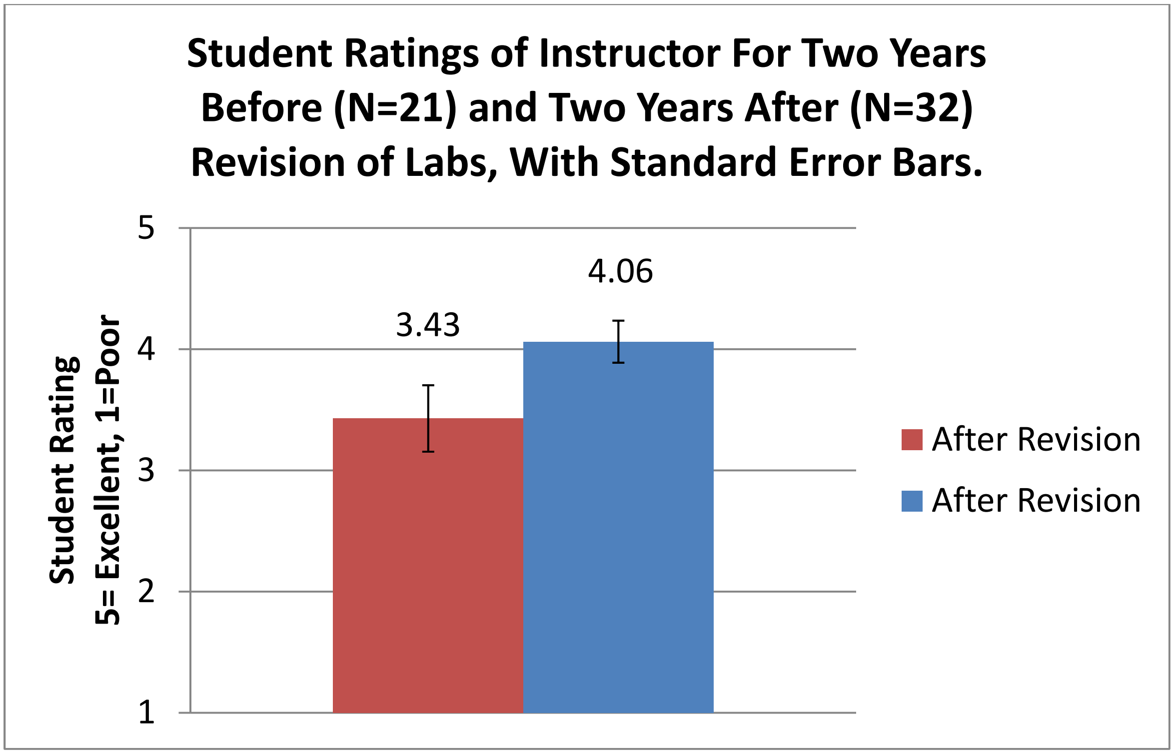

Figure 11. Student ratings of the instructor before and after revision of the labs.

From an instructor's point of view, students demonstrated strong engagement behaviors. Groups of students, for example, were observed having animated discussions about whether the filter needed to be IIR or FIR, whether the quantization noise was causing a problem or not, and other technical aspects of the project. The students were also observed working for several weeks, and through finals week, determined to get the system working.

Student understanding of the material and level of competency was assessed by the instructor during the evaluation of the final project's ability to meet technical specifications. Only a few groups were able to get the system working to an acceptable level. The primary challenge was time. Student underestimated the complexity of making the system work and the extent to which simple filtering tasks would require deep understanding of technical issues. To address this in future class sessions, the project schedule is being pulled ahead to allow an extra week of work. While not all the projects met the technical goals, most of the students could be observed in lab wrestling for many hours with the concepts in the class and making a significant effort to understand how to apply them to the project. An alternative assessment of understanding and competency might be to compare final exam grades before and after treatment however, the final exam format had change significantly over the four years.

I feel my practical skills in implementing DSP methods were greatly improved by working on the DSP final project.

> *responses: 93.33% strongly agree or agree (N=14), 6.6% neutral or disagree(N=1)*

By putting my project report in journal paper format, I had to pay more attention to the quality of writing, quality of explanation, and quality of figures than I would have for a typical term paper project report.

> *responses: 86.6% True (N=13), 13.3% false (N=2)*

I am proud of the quality of work in the journal paper project report I produced.

> *responses: 86.6% True (N=13), 13.3% false (N=2)*

Figure 12. Student survey questions and results (N=15) in the redesigned DSP lab course. The responses indicate students were highly engaged by the final project and the final paper in journal paper format, evidence of the pride they took in their work.

In assessing the students competency through their success or failure with the project, it should be noted that although the project was on its surface straightforward and simple, the technical challenges were significant, especially for a first 10 week course in DSP. However this apparent simplicity helped in motivating students because they believed it was easy and that it should work, but to make it work required deep understanding. To illustrate some of the complexity in just the first step of the project, the left-hand image in figure 13 shows a typical, raw breathing signal the students had to work with as sampled by the analog to digital converter. After appropriate signal processing, the students were able to create the breathing signal shown in the right-hand image of figure 13. Subsequent processing was used to determine breathing rate and avoid false positive warnings.

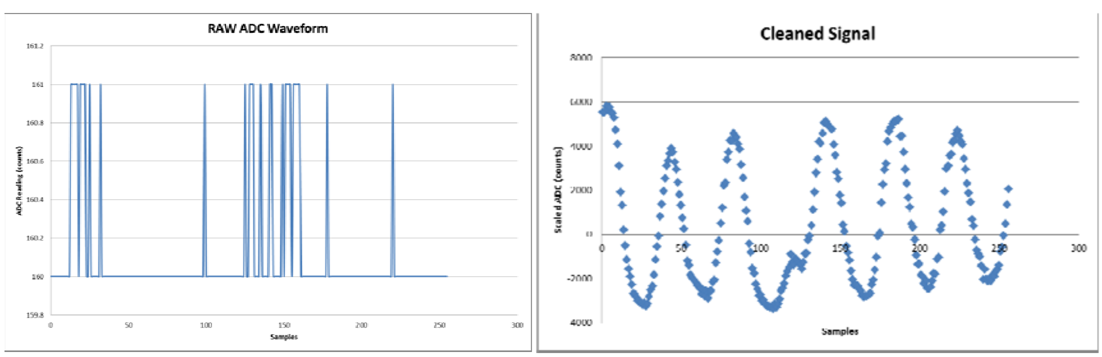

Figure 13. Images from the final report of one student group which provides an example of students' competency in designing and implementing a DSP system. The left image shows raw breathing signal at ADC without signal processing and the right image shows signal after signal processing and filtering.

"Lab project is excellently open-ended – approach, methods, and evaluation of functionality are up to students. There are pitfalls and dilemmas to overcome."

"Arduino and lab layout was very efficient for learning. I like it."

"The end project taught me how to work as a team better."

"I enjoyed the Arduino labs, better application than Matlab."

"I liked how the labs helped towards the final project"

"I thought… the labs we did were impressive and cool"

"programs such as the running statistics were hard to write"

"the amount of time for the final project could be longer"

"the labs relied heavily on the Arduino Uno and intro labs only covered general programming techniques and troubleshooting. Had to learn outside class how to program for the Arduino."

Figure 14. Students' written comments from the course evaluations.

Before the lab course was redesigned, students made no mention of the labs, positive or negative, in their evaluations of the course. In contrast, in the course redesigned following the six principles discussed here, students spoke directly to specific labs and to the project as a whole as shown in figure 14.

Conclusion and Future Work

In this paper, we have presented six guiding principles for designing better labs. Of the six guiding principles, the most important is to activate the student's own motivation for learning. Authentic lab projects can be an effective way to maintain student motivation and creates opportunities for students to experience a certain limited level of mastery as they apply component knowledge and practice integration skills. Scaffolding the labs leading up to the final project improves the students' chances for success.

Throughout this paper, we have attempted to give examples of clear and direct communication with the students about the intent of each task in the lab, seen, for example, in figures 2 and 5. By making the pedagogy overt, students can better identify the purpose of each activity and attend to the tasks that matter most to learning. This helps students avoid skipping over steps or placing insufficient focus on tasks they might otherwise see as irrelevant.

The students' evaluation comments reveal that more work is needed to manage the workload in the course. To address this, a future area of work is exploring scaffolding in the broader context of the total curriculum. This would help address the students confidence gaps in areas such as programming. Scaffolding across multiple courses and instructors creates additional challenges. Some limited exploration of these challenges has begun with an electronics course sequence that feeds into the DSP class and will be the subject of a future paper.

References

1. How Learning Works: Seven Research-Based Principles for Smart Teaching, Susan A. Ambrose, Michael W. Bridges, Michele DiPietro, Marsha C. Lovett, Marie K. Norman, 2010. John Wiley & Sons.

2. Engaging Ideas: The Professor's Guide to Integrating Writing, Critical Thinking, and Active Learning in the Classroom, 2nd Edition. John C. Bean 2011.

3. Learning to Communicate in Science and Engineering.Lerner, Poe and Craig. 2011. MIT Press.

4. How People Learn: Brain, Mind, Experience, and School: Expanded Edition, 2000. National Academies Press.

5. Wicks, C., "Lessons learned: teaching real-time signal processing [DSP Education]," *Signal Processing Magazine, IEEE* , vol.26, no.6, pp.181-185, November 2009 doi: 10.1109/MSP.2009.934191

6. Djordjevic, V.; Lhotska, L.; Gerla, V.; , "Gender ratio in Biomedical Engineering," *EAEEIE Annual Conference (EAEEIE), 2011 Proceedings of the 22nd*, pp.1-6, 13-15 June 2011.

7. AC 2010-788: Women in Biomedical Engineering: Current status and a review of potential strategies for improving diversity, Naomi Chesler, Rebecca Richards-Kortum, Sangeeta Bhatia, Gilda Barabino. ASEE 2010 Annual Conference & Exposition.

8. Beyond technology: questioning, research and the information literate school. Jamie McKenzie, 2000. FNO Press.

9. Mind in Society: Development of Higher Psychological Processes, L. S. Vygotsky, Edited by Michael Cole, Vera John-Steiner, Sylvia Scribner, Ellen Souberman 1980, Harvard University Press.