A New ECE Curriculum for Carnegie Mellon

Rob A. Rutenbar, Editor
L. Richard Carley
Stephen W. Director
James F. Hoburg
Pradeep K. Khosla
B.V. K. Vijaya Kumar
Ronald A. Rohrer
T. Ehud Schlesinger
Daniel D. Stancil
Jay K. Strosnider
Donald E. Thomas

Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213
CHAPTER 1  Motivations for a 
New ECE Curriculum
at Carnegie Mellon  
Wiping the slate clean  
Background: the existing curriculum, students and faculty 
Why change? 
This document

CHAPTER 2  Curriculum Design:
Problems and Solutions
Problems to address 
General solutions

CHAPTER 3  The New ECE Curriculum
Basic organization 
Key new courses 
Detailed curriculum requirements

CHAPTER 4  Curriculum Templates: Example Paths 
through the New ECE Curriculum
Notational conventions 
A traditional electrical engineering curriculum 
A traditional computer engineering curriculum 
A novel preparatory curriculum:
the ECE core generalist 
A focused preparatory curriculum:
the premedical student 
A curriculum with one year of study abroad 
A curriculum with specific technical focus:
the CAD/VLSI designer 
Another curriculum with technical focus:
the data storage systems designer 
A program for the late starter
<table>
<thead>
<tr>
<th>CHAPTER 5</th>
<th>Implementation Issues 53</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Freshman engineering 53</td>
</tr>
<tr>
<td></td>
<td>Timing ECE course offerings 54</td>
</tr>
<tr>
<td></td>
<td>Advising 54</td>
</tr>
<tr>
<td></td>
<td>ABET accreditation 55</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER 6</th>
<th>Miscellaneous Course Listings 57</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Specific course requirements 57</td>
</tr>
<tr>
<td></td>
<td>ECE breadth courses 58</td>
</tr>
<tr>
<td></td>
<td>ECE capstone design courses 59</td>
</tr>
</tbody>
</table>
FIGURE 1  NEW ECE CURRICULUM: AREAS  18
FIGURE 2  NEW ECE CURRICULUM: ALLOCATION OF COURSES TO AREAS  19
FIGURE 3  PRELIMINARY SYLLABUS FOR INTRODUCTION TO ECE FRESHMAN COURSE  25
FIGURE 4  PRELIMINARY SYLLABUS FOR FUNDAMENTALS OF ELECTRICAL ENGINEERING  27
FIGURE 5  PRELIMINARY SYLLABUS FOR FUNDAMENTALS OF COMPUTER ENGINEERING  29
FIGURE 6  ECE BREADTH AREAS  32
FIGURE 7  TEMPLATE NOTATION  35
FIGURE 8  A TRADITIONAL ELECTRICAL ENGINEER  37
FIGURE 9  A TRADITIONAL COMPUTER ENGINEER  39
FIGURE 10  AN ECE CORE GENERALIST  41
FIGURE 11  AN ECE PREMEDICAL STUDENT  43
FIGURE 12  AN ECE STUDENT WHO STUDIES ABROAD FOR ONE YEAR  45
FIGURE 13  A CAD/VLSI DESIGNER  47
FIGURE 14  A DATA STORAGE SYSTEMS DESIGNER  49
FIGURE 15  A LATE STARTING ECE STUDENT  51
This document describes the results of approximately two years of effort to redefine the undergraduate curriculum of the Department of Electrical and Computer Engineering (ECE) at Carnegie Mellon University (CMU). We begin by reviewing the major steps along the path toward the curriculum proposal presented here. We then summarize the state of the existing ECE curriculum, and most importantly, the motivations for making substantial changes to a curriculum that we and others already regard as being of exemplary quality.

Wiping the slate clean

In October of 1989, the Dean of CIT—the Carnegie Institute of Technology, our engineering college—initiated a college-wide review of all undergraduate engineering curricula. The goal was to evaluate how well the educational mission of the college was being conducted, with an eye toward redefining both college-wide and department-specific curriculum requirements. A welcome attribute of this review was its lack of imposed constraints. Curricula have tremendous inertia, and often resist all but the most incremental and cosmetic of changes. Because this review process was initiated across all engineering departments simultaneously, we were allowed to consider the best possible curriculum changes, not merely those that could be wedged conveniently into our current web of requirements, pre-requisites, constraints and customs.

In January of 1990, we—the ECE department—formed a committee to analyze the strengths and weaknesses of our existing curriculum, and thereafter to develop a proposal for a new curriculum to address these problems. We took the Dean’s charge quite seriously, and immediately became the “Wipe the Slate Clean Committee.” The committee was composed of ECE faculty representing a full spectrum of technical areas and undergraduate teaching experience. The committee had the following members:
Motivations for a New ECE Curriculum at Carnegie Mellon

In May of 1990, a college-wide faculty retreat was convened to discuss the curriculum requirements that apply across departments. The consensus reached at this retreat effectively determined common Freshman requirements across all departments. Following this, several months of intense work by the Wipe the Slate Clean Committee produced a rough outline for a completely revised ECE curriculum. In November of 1990, this was presented to the ECE faculty for review. In January of 1991 a special two-day ECE faculty retreat was convened to consider the committee's curriculum proposal. After vigorous debate, the proposal was further refined, and the faculty unanimously endorsed the overall structure of the proposed curriculum. Immediately thereafter, the ECE Undergraduate Education Committee was charged with resolving unfinished details related to implementation of the proposed curriculum. In May of 1991, a more detailed proposal was again returned to the ECE faculty for review, and again was accepted with minor alterations.

Background: the existing curriculum, students and faculty

The ECE department offers two four-year ABET-accredited Bachelor of Science degrees: the Bachelor of Science in Electrical Engineering (BSEE) and the Bachelor of Science in Computer Engineering (BSCE). Both curricula share a common Freshman year emphasizing mathematics, science, and computer programming. They also share a common core of engineering classes, emphasizing linear circuits, electronics, solid state devices, digital logic design and microprocessors. In addition, these curricula (as do all curricula in the colleges of engineering and science) share common requirements for humanities and social science courses (called H&SS) that amount to roughly one such course per semester.

After this common core, the two curricula diverge. The BSEE emphasizes traditional-electrical engineering topics such as fields, analog circuits and signals and systems. The BSCE emphasizes computer hardware and software topics such as computer architecture, processor design, data structures and concurrency. Both curricula require several technical electives, and a capstone design elective.

Currently, about 40% of our students pursue the BSEE, and about 50% pursue the BSCE. Roughly 10% of our students choose to double major in both electrical engineering and computer engineering. This is accomplished at the sacrifice of most elective classes: students complete the core requirements of one curriculum using the elective slots provided in the other. Also, a few of our students double major in computer engineering and computer science (which is a separate college at Carnegie Mellon). This essentially requires that all elective classes in the BSCE curriculum are chosen to complete computer science core requirements; typically these are mathematics and software classes.
As departments go, our faculty is of medium size (about 35 faculty) and certainly first rate, its members mostly working at the frontiers of their respective research areas. Many of our faculty are Fellows of the IEEE; three are members of the National Academy of Engineering. There is also an unusual commitment to teaching in our department. For example, three of our faculty have won the Frederick Emmons Terman Award for Outstanding Young Electrical Engineering Educator, given annually by the American Society for Engineering Education (ASEE). Two of our faculty have received CIT's college-wide top teaching prize, the Teare Award. And two of our faculty have also received Carnegie Mellon's university-wide teaching prize, the Ryan Award.

Why change?

By any traditional measure, the ECE department is doing well educating its students. The department as a whole is now being consistently ranked alongside the country's top EE departments. Select components of our curricula, e.g., our graduate CE curriculum, which ranked in the top five in the 1991 U.S. News & World Report survey of graduate professional schools, are on a par with the very best in the country. We attract outstanding undergraduate students: ECE is the first choice among engineering departments of most Freshmen entering CIT. Our graduates are recruited heavily by U.S. companies, and ECE is on the list of must-visit departments for many companies that recruit only among a select set of elite schools. Our graduates who choose to pursue an advanced degree go on to the best graduate schools.

So why then are we proposing a substantial and far-reaching restructuring of the entire ECE curriculum? The answers are not simple, nor are they independent. But they can be categorized roughly as follows.

Remove structural impediments to accommodate incremental change

Curricula usually evolve by accretion, with new requirements and constraints often layered incompatibly on top of existing structures. The resulting rigid course sequences connected by spaghetti-like chains of prerequisites are difficult to modify. This is certainly true of our existing EE and CE curricula, and by extension, likely true in many similar Electrical Engineering departments that evolved over the last two decades to become departments of Electrical and Computer Engineering, or departments of Electrical Engineering and Computer Science. In our own case, the end result is that even incremental changes have become difficult to implement.

In our current parallel BSEE and BSCE curricula, even modest changes ripple in undesirable ways throughout the two programs. An example makes this concrete. As a result of a recent ABET accreditation visit, we were asked to add a linear algebra class as a graduation requirement. We responded enthusiastically, on the assumption that we could migrate the course into the early years of the curriculum, and thus make it a prerequisite for our linear circuits class. In this position, it would strengthen the background of all

EE students in our circuits and electronics courses, and broaden the background of our CE students by exposing them to more non-calculus mathematics.

Unfortunately, this ideal proved impossible to implement. There was no small-scale alteration of the BSEE and BSCE course sequences that could permit the linear algebra class to be taken by all students before the courses that would use it as a prerequisite. This problem derives from the slight differences in the first few years of the BSEE and BSCE requirements. The BSCE student begins to take computer science classes fairly early, so that Junior and Senior computer engineering courses are correctly synchronized with their computer science prerequisites. In contrast, the BSEE student has no such requirements. The end result is that we now require our students to take a linear algebra class, but we do essentially nothing to exploit this background in our core classes. The goal of uniform mathematics, science and ECE core preparation for both BSEE and BSCE students is increasingly difficult to achieve.

**Rationalize requirements for topical coverage and workload**

The disciplines of electrical and computer engineering are expanding rapidly as new technical discoveries are made and applied. Likewise, society is placing increasing demands on our graduates to apply their skills in new contexts, and to appreciate and manage intelligently the consequences of their technical decisions. Consequently, the number of “critical” topics to which ECE students could profitably be exposed is also expanding. What is not expanding is the time we have to educate someone to level of the bachelors degree\(^2\). Coming to grips with this accelerating problem is at the heart of our motivation for a significant restructuring of our curriculum.

The current ECE curriculum requires a large number of core classes, designed to ensure familiarity with a substantial subset of traditional EE and CE topics. We believe this approach, which implicitly assumes all students need exposure to (almost) all areas, will grow increasingly credibl as the core of a curriculum for the 1990s. It mandates that we compress ever more material into the same number of classes. Many courses thus fall victim to “units-creep,” i.e., challenging 12-unit classes meant to require 12 hours of work per week inflate to require 15 or 18 hours of work from even the best of students. This is caused by well-meaning faculty working hard to give students the best, most thorough view of as many topical areas they can, often with the assumption that this is the only opportunity students will ever have to see the material.

We are not opposed to demanding classes, but we believe the overall strategy of putting more material into the curriculum has become decreasingly effective. Students are being

\(^2\) An alternative is, of course, to extend the amount of time required to educate students to some minimum level of professional competence. Such an approach is being advanced by the Massachusetts Institute of Technology (see the memorandum: F. J. Corbato, L. A. Gould, J. V. Guttag, F. R. Morgenbesser, P. L. Penfield, M. F. Schlecht, S. D. Semutia, J. H. Shapiro, W. M. Siebert, G. J. Sussman, G. C. Verghese, S. A. Ward, J. L. Wyatt and C. L. Searle, “Revised Proposals for New Degree Structures for EECS Students,” MIT, March 4, 1991) which proposes a five-year accredited Masters program as the principal mechanism for educating entry-level engineers. At CMU, we have chosen instead to continue with a four-year program, but to restructure it dramatically to meet the needs of the next generation of students, and the discipline itself.
Why change?

asked to absorb increasing amounts of material, which leaves less time for reflection, for alternative perspectives on similar technical problems, and for revisiting background material to ensure comprehension. The unpredictable preparation of entering students only exacerbates this problem: we keep discovering that many of our students have never seen material fundamental to the background of our core courses. The end result is that by forcing students to juggle too many topics with too little time to master these topics, many students are learning even less material, less well.

**Emphasize engineering ideas over techniques**

A related consequence of the explosion of material is that many students come to view their courses as a set of unrelated hurdles to be overcome. As a result, many students acquire only a bag of seemingly unrelated problems and solution techniques, without ever really understanding the big ideas that bind and inform these techniques.

Conventional wisdom assumes that after first teaching a vast body of fundamental mathematics and science—which students absorb like sponges—we are free to teach engineering principles, drawing as necessary on the deep well of basic knowledge internalized by the student in these early studies. This is a lovely idea, but depressingly unrealistic. Students often have weak or wildly varying preparation in K-12 mathematics and science, and hence uncertain motivation to master the rigorous college-level versions of these fundamentals. This is also troublesome as we seek to encourage the entry of more women and minorities into engineering, many of whom have had particularly unsatisfying experiences with high school mathematics and science. When a flood of engineering ideas is introduced on top of such a precarious foundation, we should not be surprised at the imperfect results. Too often, students only have time to focus on the mechanical problem-formula-solution aspects of the topics, without developing a deeper sense of the fundamentals, the interconnections and the real ideas.

This is especially unfortunate in a fast-moving discipline, where the half-life of a Bachelor's degree is probably less than a decade, and a solid understanding of the “big picture” is the most successful base from which to acquire new skills. As educators, we do our students a disservice if we fail to impart a coherent, connected view of the ideas that define our discipline.

**Support interdisciplinary studies**

The most creative and far-reaching contributions are often made by individuals at the boundaries of several disciplines. Likewise, society is placing increasing value on engineers who can apply their skills across disciplines, and can evaluate intelligently the broader consequences of their actions. ECE is an extremely wide field, and many of its most exciting frontiers—VLSI, micro electromechanical systems, computer-aided manufacturing, telecommunications networks, supercomputing—have strong and established interdisciplinary linkages. However, in hindsight, our curriculum does little to encourage the creation of engineers who can work comfortably across the boundaries of several disciplines.

Our current curriculum implicitly assumes that there are only two sorts of engineers: EEs and CEs. These are produced by completion of a large, rigid core of EE or CE engineering classes. Although industry specifically, and society generally might value highly
Motivations for a New ECE Curriculum at Carnegie Mellon

a student who has completed, say, 60% of the EE core classes and 40% of the CE core classes, we have no mechanism for giving this broad individual a degree. Nor do we have any mechanism for coping with an even broader individual who might wish to complete, say, 30% of the EE core, 30% of the CE core, then a dozen classes in mechanical engineering, operations research and Japanese language, in preparation for a career in computer-aided manufacturing. Indeed, we would like not only to tolerate such individuals, but to encourage them.

This document

This document is the final report of the Wipe the Slate Clean Committee. It discusses our motives and concerns during the long curriculum design process, and describes thoroughly our new ECE curriculum. The following section, CHAPTER 2, revisits the problems we worried over as we dissected and reassembled our curriculum, and offers the outline of some solutions. CHAPTER 3 then describes our new ECE curriculum in detail. An important attribute of our reorganization is its flexibility. Hence, CHAPTER 4 illustrates several different paths through the new curriculum, each intended to illustrate how students with differing backgrounds, technical interests and career aspirations can be accommodated. Finally, CHAPTER 5 offers some initial thoughts on issues related to implementing the new curriculum, and CHAPTER 6 summarizes our tentative new course offerings to support the new curriculum.
We will first describe here the issues we intend our proposed new curriculum to address, and then offer the outline of some general solutions to these problems.

Problems to address

We have already outlined some of the motivations for restructuring the existing ECE curriculum. We now dissect these issues a bit more carefully, and point out which are peculiar to CMU ECE, and which are endemic problems faced by engineering curricula generally.

Student preparation is incomplete

American K–12 education can be blamed for the incomplete mathematics and science preparation of many of our students. Nevertheless, allocating blame does nothing to improve the preparation of our students after they arrive. Moreover, entering students are simply different than they were in past decades: less homogeneous, more diverse in their personal goals and career aspirations.

We must deal with the following facts:

• Students have less facility and depth in the technical areas we expect all students to have seen, for example, algebra and geometry. Some unremarkable mathematical manipulations that appear frequently in introductory science and engineering classes severely tax many students.

• Students—even the best students—have seemingly random gaps in their backgrounds. In the course of our meetings, the Wipe the Slate Clean Committee talked to a superb Senior EE student, a straight-A student who was being aggressively pursued by elite
graduate schools. Yet she mentioned to us that she was very uncomfortable in her first circuits class; having never seen a complex variable before.

- Most students have almost no basic laboratory skills when they enter the department, e.g., how to keep a lab notebook; how to observe an experiment; how to deal with significant digits and experimental error; how to use orders of magnitude and quick-and-dirty calculations to estimate whether a measured result is in the right ballpark or has gone badly-awry; etc.

- A related point: students have virtually no hardware tinkering skills. Previous generations of EEs were notorious tinkerers, with radios and motors and the like. Upon entering college, they knew what a wire was, and a battery. They knew how to solder and read the resistor color code. This is no longer true, and the most elementary of hardware skills—what a wire is, what it does, how it can and cannot connect to a battery—must now be taught explicitly. (This is not exactly surprising, given the inaccessibility of the insides of most electronic products these days.) Our students are now much more likely to have software tinkering experience. However, many students, especially those from less well-off high schools, arrive without any exposure to programming ideas or hardware concepts.

- Student expectations and faculty expectations often differ. Roughly speaking, we tend to assume students have the background, energy and motivation to go acquire whatever mathematics, science, lab skills, etc., that they lack, if we send them off in the right direction. (This has always been true of the best students.) In contrast, many students tend to assume that we will teach them every topic—the big ideas as well as the basic mechanical skills, the central topics as well as the peripheral background material—without independent initiative on their part.

Clearly, these problems are not peculiar to CMU. Any solution here must reconsider how and where in the curriculum we teach these fundamentals, and to what level of detail. We must also do a better job of synchronizing our expectations and our students’ expectations on ECE and its sub-disciplines is lacking

By the time they are Seniors, we expect our students to make intelligent choices when they have the opportunity to choose an engineering elective course. We rely on students to ask their faculty advisers for guidance here, and to listen to whatever advice is offered. It is already debatable whether this works for Seniors, who have a fairly extensive technical background. Students taking their very first course in a core area like solid state or software engineering are usually not clear about how this area connects to the rest of ECE as a whole.

The problem is much worse for our Freshmen and Sophomores. ECE offers two parallel curricula: the BSEE and BSCE tracks, one of which our students must choose sometime during the Sophomore year. In the existing curricula, the problem is how to educate students to make an informed choice. Certainly, some students arrive absolutely decided on one track or the other. But many rely on our introductory courses to paint a sufficient-

---

3. It is worth noting that the same problem exists at the-college level: many students arrive in CIT knowing that they like technical subject matter, and prefer some topics over others, but unsure of which engineering department will best suit their needs and goals.
Problems to address

ly broad picture of the discipline for them to make a choice. Unfortunately, these introductory courses concentrate almost entirely on packing in as much engineering material as possible. As faculty we are often surprised when, after a few weeks in class, in the middle of some intricate technical discussion a brave Sophomore will ask something like this:

Exactly what does a computer engineer do? And how does this material help me to be a computer engineer? Is this different from computer science? Is the difference that we do hardware and they do software? When I graduate will I only be able to design big computers, or do computer engineers do something else as well? And why am I taking all these circuits classes—isn’t that for the electrical engineers?

Our emphasis on maximizing technical content in these few hours per week leaves little time to address all these questions satisfactorily. And as the breadth of the discipline continues to expand, we must confront this problem more directly if our students are to make informed curriculum choices.

Appreciation of underlying ideas is weak

We are not alone in observing that students often acquire only the mechanical aspects of the topics that we teach, without understanding the underlying ideas. The problem is pervasive in the teaching of technical material. For example, a recent National Science Foundation article on the teaching of college calculus relates this story:

A mechanical engineering professor mentioned in passing to a class of sophomores that an integral is a sum. He simply assumed that the students had learned this basic idea from first-year calculus. But the students stared uncomprehendingly back at the professor. “Students seem to have a facility for doing things,” [the professor] concludes, “but they lack a sense of ideas.”

In our own department, similar stories abound, even from our best teachers. For example, in a recent issue of CMU Focus magazine, Jim Hoburg, ECE’s Associate Department Head and a past winner of the university-wide Ryan teaching award, observed:

What we currently do in many of our engineering “core” courses doesn’t work very well for a majority of students in the courses. The core courses in the Electrical Engineering curriculum with which I have direct experience are the Sophomore required course in Linear Circuits and the two-semester Junior required sequence in Engineering Electromagnetics. In each case, I’ve worked hard to help students achieve a rich and insightful understanding of fundamental material. Most of them seem to think I do a good job; they say on their FCEs [Faculty Course Evaluations, a survey of each student’s evaluation of and reactions to the course, conducted by CMU itself] that I make even difficult and abstract concepts seem clear.

Yet, when I look at the reality of their understanding, as gaged through exams and discussions in and out of class, it’s grossly disappointing. The majority simply don’t get it. Their survival skills allow many to get through with C’s and D’s, based mostly

---

upon regurgitation of techniques I've shown them repetitively, as both they and I have forced ourselves through a distasteful process of pounding in material which they find mysterious and useless and which I find beautiful and important.\(^5\)

There is no simple, isolated fix to this problem. Students' varying technical preparation, the increasing diversity of their backgrounds, the divergence of student and faculty expectations and the current practice of packing ever more material into the same number of classes all compound the problem. Yet we must consider the problem seriously. The mere mechanical skills that a student acquires in "survival" mode have a disturbingly short half-life in our rapidly moving discipline. The question is how to motivate students to appreciate the connections among abstract ideas, concrete applications, their classes and their careers.

**Breadth in all relevant topical areas is impossible**

The biggest problem inherent in "classical" engineering curricula is the notion that every engineer must know something about every area of the discipline. There was certainly an era in which this was a reasonable assumption for electrical engineers. But this is no longer a viable assumption, especially for a department whose faculty engage in a broad program of research ranging from basic physics to advanced computer science. Distancing our curriculum from this notion is difficult, since it tramples on nearly every faculty member's most cherished subjects.

One approach that we have already tried is to partition our undergraduates into separate degree tracks leading to different BS degrees. As Carnegie Mellon's EE department evolved into an ECE department, its degree offerings evolved into the current parallel BSEE and BSCE tracks. The motivation for this was three-fold:

- Computer engineering faculty argued that many required electrical engineering classes were inessential to the education of a computer engineer, and should be replaced by more relevant course requirements.
- Electrical engineering faculty argued that if students didn't take the full complement of required electrical engineering classes, they should not be graduated as "electrical engineers" and so a separate computer engineering degree was an acceptable solution.
- Separate degree programs offered a way for ECE to distinguish itself from its competitors for the best undergraduates, and made our commitment to computer engineering as a discipline more explicit.

Unfortunately, this is not a workable growth strategy as the discipline continues to broaden: we cannot just keep slicing off portions of the curriculum and award them separate degrees. Worse, there is a sense that the separate BSEE and BSCE tracks erect a few artificial barriers between the two "ends" of the department at a time when we should strive to enhance the links between EE and CE concerns.

---

On the other hand, there are components of our curricula that are quite strong, successful and visible nationally, e.g., aspects of the CE curriculum, and the emphasis on design across the department. Hence, any reorganization of the department and its curricula must be careful not to submerge acknowledged strengths under some overly simplified concept of departmental homogeneity.

**Interdisciplinary studies are difficult**

Our BSEE and BSCE curricula are based on a large number of required engineering classes and restricted technical electives. Of course, a BSEE student can take some computer engineering courses, just as a BSCE student can take some electrical engineering courses. But we have no mechanism to give a degree to someone who chooses to be broad, who chooses to take, say, 50% of the required electrical engineering core courses, and 50% of the required computer engineering courses. Each student must satisfy 100% of the requirements for either the BSEE or BSCE degree options. Some of our best students solve this problem by double majoring in both EE and CE. This requires completing 100% of both curricula, using elective slots in one track to take required core courses from the other track. This challenging but rigid program ends up being a four-year degree with essentially no elective classes.

There are several problems with this approach. Many of our BSEE students would prefer a stronger software component, similar to the BSCE track. However, most balk at the course load required to fit the natural software sequence early in their studies—when it would be most useful—in parallel with a demanding load of EE courses. Similarly, many of our CE faculty would prefer to see their BSCE students acquire some more electrical engineering breadth, especially for example those students interested in VLSI or CAD. Again, typical CE students are reluctant to increase their course load and take something they (erroneously) perceive to be exotic and useless—like electromagnetics.

More generally, we have no mechanisms to allow students to trade ECE depth for breadth—either breadth within ECE, or breadth among other disciplines. We would like to encourage more interdisciplinary students, both across ECE areas, and across other departments at CMU.

**Demographics are changing**

On the entering side of the curriculum, our students are less homogeneous than in the past. We are serious about attracting and retaining talented women and minorities, and so we must expect further diversity in the backgrounds of our entering students. The problem of students not having the basic skills and motivations we would prefer of them will likely worsen. Thus, we have to construct the first few years of courses around this fact.

On the graduating side, there is also increasing diversity. As faculty members, it is common for us to treat our students as though they are replicas of ourselves, i.e., to assume they all wish to become first-class researchers and stay on a technical path for the rest of their lives. But such students are clearly in a minority. Many of our students who graduate to become engineers will not stay in technical positions for their whole lives. Moreover, it has become increasingly apparent in recent years that some of our best and brightest do not choose an engineering degree with the intent to become practicing engi-
neers, but rather with the intent to enter other postgraduate professional schools, such as law, business and medicine. To encourage a population of more broadly educated engineers, we must refuse any urge to relegate these particular students to second-class status, or deride them as defectors from the fold. Indeed, we can see few negatives associated with the idea of a future generation of technically literate legislators, judges, physicians and business leaders.

The central question is how to structure a curriculum to handle the educational needs of all these constituencies: the hard-core committed technologist, the mainstream engineer who may be in management in less than decade, and the interdisciplinary student using ECE as a launching point for a career in another professional discipline. These facts argue for a curriculum in which a strong core of ECE topics can be augmented with advanced ECE classes, or preparatory courses for other disciplines.

Rigid curricula impede necessary changes
Constraints have accreted on and between our existing BSEE and BSCE curricula to the point where even modest changes are difficult. A top to bottom rethinking of the overall structure of our curricula must consider how we can add flexibility to its basic organization so that necessary incremental changes are more easily effected.

General solutions
The new ECE curriculum we will describe in the next chapter departs substantially from our current curriculum. As well as changes in our own department, there are changes at the college level, specifically the Freshman year common to all CIT engineering students. Here, we briefly summarize our attempt at general solutions to the problems raised above.

Teach engineering early, concurrent with fundamentals
In our existing curricula, the Freshman year is common to all departments in CIT, and emphasizes mathematics, science and H&SS. A few “Freshman elective” slots must be filled, but the choices comprise an ad hoc selection of peripheral engineering courses that are largely unrelated to the core curriculum of any engineering department in CIT. Students are largely disappointed by these courses.

In the new curriculum, every department in CIT will offer a Freshman engineering course that introduces students to the ideas, problems, modes of thought, tools and techniques of their discipline. Every student will take at least two such courses in their Freshman year, concurrent with their mathematics, science and H&SS classes. The Freshman elective slots will be eliminated in favor of these departmental introduction courses. ECE will offer a single course, called Introduction to Electrical and Computer Engineering, that strives specifically to provide an integrated view of connections between EE and CE problems.

CIT and its constituent departments want the new Freshman engineering courses to show students what engineering is all about. We hope to generate real enthusiasm—the “Ahah!” that accompanies insight as students grasp that they can, for example, model in-
teresting physical phenomena with a little mathematics, science and judgment—and let them get their hands dirty on real problems. This experience should demonstrate to new students the practical need for more preparation in subsequent mathematics and science classes, and should also provide students with the elementary hands-on laboratory skills that they often lack. At the college level, it will allow undecided students to sample various engineering departments to be sure they are choosing the right one.

In addition, we will migrate some of our ECE core classes downward in the curriculum. Currently, these engineering classes are like rewards: after completing a rigorous (for many, mysterious) set of mathematics and science courses, the student is finally allowed to take one. In the past, we always tended to move core classes upward in the curriculum: the idea was to shove as many prerequisites as possible underneath them. In the new curriculum the first set of core engineering classes will be structured to have mathematics and science as corequisites instead of prerequisites. These will be available to students somewhat earlier in their studies, typically the first semester of the Sophomore year. These ECE courses will also be substantially reorganized to account for what students really need to know at this point in their studies, to teach some essential mathematics with an applied flavor, and to motivate the links to other related courses in ECE, mathematics and science.

The unifying idea here is to expose students to real engineering as early as possible, to motivate their studies in necessary mathematics and science courses while they are taking them, and to teach explicitly some manual skills they universally lack.

Base curriculum requirements more on areas, less on specific courses

Our existing EE and CE curricula are each based on a rigid core of required classes. In the proposed curriculum, we first drastically reduce this required core, from about a dozen courses down to a select few. Next, we replace the bulk of the remaining specific course requirements with area requirements: ECE is now "partitioned" into a spectrum of topical areas, and all upper-level courses are assigned to one of these areas. Students will be required to demonstrate breadth, depth and coverage across some chosen subset these areas. We will no longer require students to take one or two courses in every core EE or CE area. Instead, we will let students demonstrate that they are broad enough to take courses in several—but not all—different areas, and that they are deep enough to take more advanced courses in some—but not all—areas. Coverage requirements—to ensure that the student takes enough ECE courses to be called an engineer—and a capstone design requirement complete the basic curriculum.

Increase flexibility, elective courses

The existing curricula feature a hodgepodge of curriculum-specific elective slots (the BSEE and BSCE manifest different constraints on allowable electives) restricted in variety of ad hoc ways. The actual number of completely unconstrained elective slots is amazingly small: one slot.

The new curriculum substantially increases the number of unconstrained elective slots: we will allow slightly less than one full year of free electives. These courses must be graded courses offered by a Carnegie Mellon academic unit, but aside from this they are not further constrained. The intent here is not to reduce the rigor, complexity, or depth of
understanding associated with the ECE degree. Rather, the intent is to create new opportunities for more broadly-based curricula that integrate ECE courses or courses from other disciplines in innovative ways. We see no reason to prevent an ECE student with an interest in, say, integrated silicon sensors, from taking a half dozen courses chosen sensibly from mechanical engineering, physiology, biology, chemistry, mathematics or physics. Nor do we see any compelling reason to prevent an ECE student with an interest in computer speech recognition from taking a year of linguistics, a foreign language, or even music theory and cello mastery. Adhering to some unnecessarily rigid concept of an ECE degree can only serve to stifle the creation of innovative programs of study, and innovative engineers themselves.

Manage the workload

In the existing curriculum, students typically take five courses per semester. Usually, these comprise four technical classes and one H&SS class. Especially in the last two years of their studies, engineering classes tend to inflate in content past their specified units as our faculty members struggle to compress every relevant topic, technique, nuance and anecdote into their classes. During such semesters, the nominal 40 to 50 hours of work per week (computed by tallying the units on each student’s classes) is only an optimistic lower bound on the time students must put in to survive. We are not opposed to demanding course schedules, but our students are juggling too many topics, often at unsustainable levels of stress and well beyond an optimal level for real understanding.

In the new curriculum, we first attack this by reducing the number of courses from five per semester to four. ECE courses will remain challenging and work-intensive, but the switch to one fewer class per semester should make it possible for ordinary students to concentrate fully on their studies, master their material, and still have a life.

In addition, by revisiting all the courses we currently offer to see how they fit in the spectrum-of-areas structure of the new curriculum, we will render the workload a bit more uniform across all courses. In particular, we will reallocate topics more carefully across course sequences. (In the existing curriculum, topics tend to creep downwards through a course sequence, to make room for more topics at the high-end of the sequence.) Finally, we will try to make all ECE courses the same number of units. Currently, some ECE courses are 9 units (typically those without labs) and some are 12. In the new curriculum, each course will be 12 units, and we will try to make the work demands of each ECE course more roughly constant. The new 12-unit versions of relatively abstract and mathematical courses like electromagnetics and signals and systems will incorporate new simulation laboratories and, thus, a more experiment-based learning component.

Offer one BS degree, not two

We currently offer two degrees: the BSEE and BSCE. In the new curriculum, we will offer only one: the Bachelor of Science in Electrical and Computer Engineering, or BSECE. There is an appealing symmetry here. We originally offered a single BSEE degree when we were the department of Electrical Engineering. As the computer engineering discipline gained stature, we offered a BSEE “with Computer Engineering Option” which eventually split off to become the separately accredited BSCE degree. Now we propose to take the next step and merge all our degrees back into a single integrated
BSECE. This explicitly recognizes evolutionary trends in the discipline and in industry to emphasize the commonality across EE and CE, and not the differences.

Nevertheless, there has been considerable discussion among our faculty about the merits of recognizing on each student’s degree the completion of courses in areas of common concentration. As a concrete example, we might think of awarding a BSECE degree “with emphasis in Computer Hardware” or “with emphasis in Signals and Systems” to allow students to highlight their curriculum choices. It has been argued that this is a neat mechanism to preserve the visibility of the strong, existing components of our department and the strong courses in our curriculum. Whether and how to implement this are still under debate, however.

**Structure curriculum to accommodate change**

A key feature of the new curriculum is that it is based less on specific courses, and more on requirements to take courses in general topical areas. By organizing the curriculum to be more loosely independent of the content of specific courses, we have freed it to adapt more easily to change. It now much easier to see how to add a breadth/depth course, a new topical area, or even a core requirement. As the discipline itself evolves, the new curriculum should be able to absorb necessary incremental changes without violating the basic spirit of its design.
We can now describe fully our new ECE curriculum. We begin by surveying the basic
components of the curriculum, their organization and relationships, and then enumerate
the specific requirements for the new BSECE degree.

Basic organization

FIGURE 1 illustrates the basic components of the curriculum by topic; FIGURE 2 adds de-
tail to show how courses are allocated among these areas.

Curriculum components by area

The architecture of the proposed curriculum is essentially simple, and comprises the fol-
lowing:

- A typical humanities and social sciences component. (8 courses.)
- A typical mathematics, science and computer programming component. (7 courses.)
- Freshman engineering courses—very much atypical—which use these mathematics,
  science and programming classes as corequisites, introduce students to their depart-
  ment (and other departments) and provide needed motivation and background skills.
  (At least 2 courses.)
- ECE core requirements, a select set of fundamentals classes required of all ECE stu-
dents. These courses are the gateway to all elective upper-level ECE courses. (2
courses.)
- ECE breadth requirements, selected from across the set of specified topical areas in
ECE, to ensure exposure to different styles of thinking, modeling and problem-solv-
ing. (3 courses)
FIGURE 1  NEW ECE CURRICULUM: AREAS

8 semesters

4 courses per semester

- Humanities & Social Sciences
- Freshman Engineering
- Mathematics, Science, Computer Programming
- ECE Required Core
- ECE Breadth
- ECE Coverage, Depth, Design
- Free Electives
- Free Elective
### Basic Organization

**FIGURE 2**

**NEW ECE CURRICULUM: ALLOCATION OF COURSES TO AREAS**

<table>
<thead>
<tr>
<th></th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>FRESHMAN</td>
<td><strong>H&amp;SS</strong></td>
<td>Introduction to ECE</td>
</tr>
<tr>
<td></td>
<td><strong>H&amp;SS</strong></td>
<td>Introduction to Engin</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOPHOMORE</td>
<td><strong>H&amp;SS</strong></td>
<td>Fundamentals of Electrical Eng</td>
</tr>
<tr>
<td></td>
<td><strong>H&amp;SS</strong></td>
<td>Fundamentals of Computer Eng</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>JUNIOR</td>
<td><strong>H&amp;SS</strong></td>
<td>ECE Breadth 1</td>
</tr>
<tr>
<td></td>
<td><strong>H&amp;SS</strong></td>
<td>ECE Coverage 1 (Ex: Depth)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>SENIOR</td>
<td><strong>H&amp;SS</strong></td>
<td>ECE Coverage 3 (Ex: Design)</td>
</tr>
<tr>
<td></td>
<td><strong>H&amp;SS</strong></td>
<td>Free Elective</td>
</tr>
</tbody>
</table>
• ECE coverage requirements, to ensure enough exposure to ECE courses to earn a degree called Bachelor of Science in Electrical and Computer Engineering. (At least 3 courses.)

• As part of the coverage requirement, an ECE depth requirement to ensure that students can handle advanced as well as introductory material. (1 course of the 3 coverage courses.)

• Also as part of the coverage requirement, a capstone design requirement, to ensure exposure to the unique problems of building concrete engineering artifacts under tight time, resource and cost constraints. (1 course of the 3 coverage courses.)

• Free electives, nearly one year in all, to be chosen by individual students based on their interests and goals. (7 courses.)

Flexibility = core + breadth + depth + design + electives

The key attribute of this new curriculum architecture is its flexibility. Requirements to "take one course in every important area" or "attain basic mastery in every area," which in the past led to cumbersome intertwined course sequences are avoided entirely. Instead, we mandate only select core knowledge, along with breadth, depth and design. Within this framework, many sensible plans of study can be formulated, some favoring depth, some favoring breadth, others somewhere in between. Another important consequence of this flexibility is how well the proposed curriculum will accommodate new students with different backgrounds and skills:

• The best students are free to begin engineering classes earlier, and pursue several ECE areas in great depth.

• Students inclined to be generalists can, within clear limits, trade depth for breadth, and explore a wider range of ECE topical areas. At the very least, we can now actually accommodate the student who wants to take, say, half of her courses in traditional electrical engineering topics, and half in traditional computer engineering topics.

• Those with more potential than actual preparation can use electives to fill in any gaps, and defer some engineering until later; these students in particular may benefit by choosing to trade-off some depth for breadth. (Specific courses do not have to be taken in the semesters indicated in FIGURE 2, as long as all pre- and corequisite requirements are satisfied.)

• Interdisciplinary students now have the time to dive into another discipline, its background topics, its core classes, and even a few advanced courses. This will be helpful for students pursuing a parallel technical discipline, as well as students using ECE as preparation for graduate studies in another profession such as medicine, law or business.

The large number of free electives deserves special comment. Sadly, it seems that the presence of any free electives in an engineering curriculum is rare, let alone nearly a full year as our proposed ECE curriculum allows. The desire to pound increasing amounts of technical, discipline-specific material into our students in the same number of hours per week is at least partially to blame here. M. Granger Morgan, Professor of ECE and also head of the Department of Engineering and Public Policy at Carnegie Mellon, in a recent front-page editorial in Science magazine also laid part of the blame on the engineering accreditation process:
For more than a century and a half, engineering schools in the United States have pursued a variety of educational philosophies, offering programs built around their local comparative advantages. The resulting diversity has been an important source of national technological strength. Today, faced with challenges induced by rapid global political, economic, and environmental change, we need diversity and innovation in our new engineering graduates more than ever before. But, in contrast to the flexibility of undergraduate science education, heterogeneity and innovation in U.S. engineering education are threatened by the creeping demands of our system for accrediting undergraduate engineering curricula. When a school's most basic educational objectives and the ABET [Accreditation Board for Engineering and Technology] constraints have both been met, there is often little or no flexibility left. It is not unusual for an engineering undergraduate to have no opportunity to select a completely free elective course.  

The pressures for limiting electives to a negligible few may be numerous and well-intentioned. Nevertheless, we remain attracted to the goal of increasing the diversity among the population of Electrical and Computer Engineers that we educate. Augmenting our required ECE core, coverage, depth, breadth and design courses with a substantial number of free elective courses is a solid step in this direction.

Key new courses

The new curriculum introduces three critical new courses. One of these is the department's Freshman engineering introductory course, called Introduction to Electrical and Computer Engineering. It is worth noting that the Wipe the Slate Clean Committee considered for several months the idea of two such courses, Introduction to Electrical Engineering and Introduction to Computer Engineering, in keeping with the existing parallel BSEE and BSCE tracks. In the end, however, we rejected this in favor of a single course that emphasizes the links between the component areas in ECE.

Typically in the Sophomore year, the student will complete the small ECE core, comprising two courses and their corequisites:

- **Fundamentals of Electrical Engineering**, which has Linear Algebra as a corequisite, and can be thought of as a course in linear circuits and circuit theory.
- **Fundamentals of Computer Engineering**, which has Introduction to Modern Mathematics (i.e., Discrete Math) as a corequisite, and is a course in digital design, microprocessors and elementary computer organization.

In the following sections we describe the contents of these courses in more thorough detail.

---

Freshman engineering: Introduction to Electrical and Computer Engineering

- **Prerequisites:** None
- **Corequisites:** Calculus I, Introduction to Computer Programming

Although our incoming Freshmen are highly motivated, many have no real understanding of what undergraduate studies in engineering are all about, and few have the hands-on laboratory experience that was common a decade ago. In the existing curriculum, our Freshmen take preparatory courses such as physics and calculus, but must wait until the Sophomore year to take any real core ECE engineering courses. The result is often a significant reduction in motivation as students work to understand a wealth of relatively abstract fundamental mathematics and science presented without any supporting ECE-specific engineering context.

Our new Freshman introductory course is intended to rectify this situation. The course will motivate and introduce basic concepts in Electrical and Computer Engineering in an integrated manner, provide hands-on laboratory experience early, and strive to imbue students with some ability to look at the “big picture” and ask questions that will lead to a solution to the problem at hand. The course emphasizes a top-down approach aimed at providing students with a high-level, systems-oriented view of the material. We introduce fundamental theoretical ideas in both EE and CE. A simple mobile robot system serves as the experimental vehicle to motivate the teaching of basic concepts like Kirchhoff’s laws, DC models of circuit elements, logic gates, flip-flops, counters, and so forth. The subsystems that comprise the robot provide the basis for a sequence of interesting laboratory exercises.

Since ECE is a blend of physics, mathematics, computer science and engineering practice, the course covers both theory and applications. Lectures emphasize theoretical aspects, laboratories emphasize experimental techniques, and recitations focus on problem-solving skills. The robot is used to illustrate how complex systems can be decomposed into subsystems and to motivate the need for the theory behind each of these subsystems. In the laboratory, students analyze, construct and test these subsystems to make sure they really do work as predicted by theory. The laboratory experience also demonstrates the thought processes behind the development of complex systems with many component parts. The virtues of focusing on a small robot are that it is simple enough for Freshmen to understand, complex enough to illustrate the larger view of how EE and CE ideas mesh, and provocative enough to keep students interested when the going gets tough.

The course focuses on black box models of systems, such as power supplies, sensor circuits and digital controllers, and the behavioral description of primitive elements, such as resistors, capacitors, transistors, gates and flip-flops. We begin by considering a black box model of a system and then move to consideration of black box descriptions of each subsystem. This process is repeated until we ultimately consider primitive elements like circuit components. For example, after examining the mobile robot as a whole, we break it down into electrical and mechanical subsystems. Later, the electrical system is further refined into power supplies, sensors, control logic, etc. While considering the power supply, we introduce notions such as voltage regulation and load capability. At this time, concepts such as an ideal source and a real source are introduced, along with
a behavioral model of resistive elements—which of course turns out to be Ohm’s law. All these more basic concepts are required to understand the “larger” concept of voltage regulation. In this class, students do not start by learning how a circuit works; rather, initial emphasis is on understanding the black box behavior of the system at hand. Fundamentals and theory—ideal versus real sources, Ohm’s law—are introduced only after the need to explain some physical behavior has been established. This approach has been referred to by some as just-in-time learning7, since we avoid piling up a potentially confusing inventory of unmotivated and mystifying theory and mathematics taught with the promise that “it’ll be good for you—we’ll tell you why later.”

To follow the power supply example somewhat further, note that we end up decomposing this subsystem into resistors, capacitors, transistors, batteries (or voltage sources) and Zener diodes. We develop simple models of each of these elements and introduce the relevant physical quantities (e.g., charge, current, voltage) and relationships. For example, for a capacitor, we tell students how the current flowing through the device is proportional to the slope of the curve defining the time-varying behavior of the voltage across its terminals; this can be illustrated quite clearly in graphical form. Similarly, we illustrate the behavior of an ideal diode graphically, using current versus voltage plots. This serves as a foundation for discussion of forward and reverse biasing, which leads subsequently to Zener diodes and elementary transistor behavior. An understanding of each of these circuit elements allows students to grasp the overall structure of the power supply itself, and provides us with the context for introducing fundamentals like Ohm’s law, Kirchoff’s voltage and current laws, and implicit techniques for circuit analysis. Note also that our behavioral models of the transistor and diode are introduced without ever introducing the concept of electrons, holes, doping, etc. These notions are really not required to understand the basic workings of a transistor or diode, and the engineering context for developing these ideas has not yet been established.

A set of coordinated laboratory exercises track the lectures and give many students their first hands-on laboratory experience. Although the ultimate aim is to assemble a working mobile robot, students attain this goal through a series of smaller projects that allow them to test the ideas being developed in lecture. For example, while a power supply is being dissected in lecture, students are building, testing and debugging a simple power supply (on a proto-board separate from the robot itself) to ensure that they understand what will actually be happening after they wire up their robot’s supply. These exercises are also designed to show how systems are built and tested in a methodical manner.

Similarly, we enumerate, decompose and describe the other subsystems of our robot, such as the motor driver, the sensors, and the digital control and programming components. The bridge between analog and digital ideas is made after ideal operational amplifiers—op-amps—are described, and it is shown how they can be used in an analog circuit as a linear amplifier, and also as part of a digital logic element. Digital topics are then similarly introduced as black box behavioral models, and students acquire the fundamentals of Boolean algebra, combinational circuits, simple sequential circuits like counters, memories, and so forth. One of the important points we stress is the connec-

7. We attribute the use of the term in this context to Mac Van Valkenberg of the University of Illinois.
tion between analog and digital ideas. In our current curriculum, entering students often have no clear vision of the relationships between the EE and CE “ends” of the department. Traditionally, students are exposed to analog systems in the guise of circuit analysis in an introductory circuits class, and then exposed to digital systems in another introductory course on combinational and sequential logic. Accordingly, some students develop an “us” versus “them” parochialism as they come to identify themselves as primarily electrical engineers or primarily computer engineers. We hope to remedy this by exposing our students immediately to a more unified view of the different disciplines comprising ECE.

Overall, the course consists of two lectures, one laboratory session, and one recitation session per week. Over the course of a semester, students will assemble a small mobile robot capable of making right and left turns, moving in a straight line, flashing a small light, and beeping, all under simple program control. Toward the end of the semester, when a completely working system has been realized, students will be asked to add some hardware to the robot, or program it in creative ways to make it do interesting things. This will be the final project for the course and students will have a large degree of latitude in defining it. By the time they complete the course students should be able to analyze basic circuits that include resistors, capacitors, ideal diodes, transistors, and opamps. They will be able to manipulate simple logic circuits composed of gates and flip flops. They will have a basic understanding of some simple sensors and transducers. The laboratory experience will also promote understanding some of the practicalities of making things work, and the reasons real systems sometimes behave differently than what is predicted by theory. Students will also have had a measured dose of ECE “engineering reality” designed to help develop logical thinking and problem solving skills.

FIGURE 3 shows a preliminary syllabus of topics for the course.
Functional decomposition of a system
- Block diagrams

Basic circuit concepts
- Ideal voltage and current sources
- Real voltage and current sources
- Behavioral model of R, C elements
- KVL, KCL, Ohm’s law
- Behavioral model of active devices: transistor, diode, Zener diode
- Operation of a DC motor
- Transducers: LED, touch switch, speaker, ultrasonic range sensor
- Ideal op amp model
- Inverting amplifier
- Buffer and non-inverting amplifier
- RC time constants

Digital logic concepts
- Digital signals
- Binary numbers
- Logic functions
- Karnaugh maps
- Flip flops and shift registers
- Asynchronous logic
- Synchronous logic
- Coding (BCD)

Building complex systems from basic building blocks
- Interconnecting digital elements
- Interconnecting circuit elements

Laboratory project: Building a working robot
- Electrical safety
- Dealing with the power supply and motor
- Dealing with transducers: LEDs, beeper, touch switch, clock, buffers
- Dealing with digital subsystems: gates, flip flops, counters, reset
- Dealing with the system: memory, programmed and hardwired control
- Integrating all the pieces
ECE core requirement: Fundamentals of Electrical Engineering

- Prerequisites: *Introduction to Electrical and Computer Engineering*
- Corequisites: *Linear Algebra*

This is the first real “circuits” course. It differs from traditional courses primarily in two ways. First, we can exploit the fact that students have had some exposure, both practical and theoretical, to linear circuit ideas and issues in the robot project from the Freshman *Introduction to ECE* course. Second, we choose a non-traditional focus for the class (like the robot in the Freshman course) to motivate students and provide a vehicle for developing the fundamentals. This focus is transient analysis of linear interconnect circuitry, or, said another way: *how fast can a computer be?*

The course introduces the idea that computer system speed is measured in Millions of Instructions per Second (MIPS)—the more MIPS the better, the more MIPS the more expensive the system. Students are expected to have a rudimentary idea of what a computer instruction is from their assembly and test of the stored program control portion (memory plus state machine) of the robot in *Introduction to ECE*. We will explain how, a decade or so ago, this speed was determined more by transistor size, *i.e.*, how small a device can be made, how many can be packed inside one chip. But now and for the foreseeable future, computer system speed is determined more by the interconnect wiring itself, inside the chips, among the chips, and among the increasingly exotic system-level packages that carry the chips. This focus provides a suitably interesting context to introduce the basics of lumped linear circuits: resistance, capacitance and now, inductance.

As expected, the course focuses on studying in detail many RLC circuits. However, we always motivate these circuits by asking the question: *what will be the effect on switching signal behavior?* The goal is to show that simple circuits provide the right insights to understand even the most complex of interconnections, and that complex circuits can be understood by mastering the basic concepts of circuit theory: Kirchoff’s current and voltage laws; superposition and convolution; series, parallel and ladder circuit analysis; Thevenin’s and Norton’s theorems; natural frequencies and *jω*; circuit partitioning; and nodal analysis.\(^8\)

The class will have lectures, a laboratory session, and a recitation session. Lectures will motivate and develop theoretical material, laboratories will provide more hands-on exposure to circuits to test how well the theory really works, and recitations will provide opportunities for problem solving and review. An interesting component of the class will be computational programming exercises, in which students will build upon a sequence of very small linear circuit analysis programs (*e.g.*, for simple ladder circuits) to demonstrate that the analysis of complex circuits is a straightforward extension. This will exploit the *Linear Algebra* corequisite course, and we hope better engage students who want to think of themselves primarily as CEs, not EEs.

**FIGURE 4** shows a tentative syllabus of topics for this course.

---

\(^8\) For additional perspective here, see: Ronald A. Rohrer, “Taking circuits seriously,” *IEEE Circuits and Devices*, vol. 6, no. 4, pp. 27-31, July 1990.
- **Introduction**: Lecture demonstration and laboratory verification of the delay that can be introduced between a driving and a driven digital gate. Show waveforms that result from one inch, one foot, one hundred feet of wire, etc., between the two CMOS gates. Determine delay thresholds. Discuss delay as a function of fanout.


- **Approximate IO modeling of (CMOS) driving and driven gates**: Laboratory experiments with one gate driving one/many gates and with various lengths of interconnect to determine a functional form for the driving source, its equivalent resistance and load capacitance.

- **IC interconnect resistance and capacitance modeling**: Resistance as a function of height, width and length; capacitance (over an SiO₂ insulated ground plane) as a function of length and width; studies of various materials and feature sizes in terms of single time constant RC model; examples of 3-D interconnect structures. Analysis of CMOS inverter: behavior with/without capacitive load; simplified models of inverter and more complex gates.

- **Lumped approx. of linear interconnect with distributed parameter elements**: Two capacitances and general solution of related state equations; several lumps, the related state equations and the form of their solution; numerical solution of ladder circuits via forward Euler, backward Euler and trapezoidal integration and nodal and ladder analysis.

- **Fanout interconnect**: Signal distribution and clock distribution; tree structured circuit solution via Thevenin’s and Norton’s theorems; port description; brief discussion of sparse matrices.

- **Speedup connections**: Meshes of resistance and nodal analysis in general; circuit partitioning and sparse matrices revisited.

- **Capacitance coupling**: Noise immunity; analysis of RC ladders with line-to-line capacitance coupling.

- **Inductance effects**: self-inductance on wires and coils; transient analysis of an RLC circuit; over-, under-, and critical-damping; natural frequencies; general solutions; numerical analysis of general circuits; numerical solution of RLGC-tree structured circuits; mutual inductance.

- **Transmission lines**: telegrapher’s equations, travelling wave solutions; lumped LC approximation.

- **Clock distribution**: transient analysis for periodic excitations; direct solution of periodic steady state; Fourier series; sinusoidal steady state; Bode plots; filtering.

- **General RLC circuit analysis**: (Review and generalization of what has gone before.) Nodal and state equations; zero-state and zero-input responses; natural frequencies; linearity and superposition; poles, zeros and residues.
ECE core requirement: Fundamentals of Computer Engineering

- **Prerequisites:** Introduction to Electrical and Computer Engineering
- **Corequisites:** Introduction to Modern Mathematics (Discrete Math)

This course builds upon the rudimentary digital design and computer engineering concepts presented in the Freshman Introduction to ECE class. The new course is actually patterned more closely on an existing course, 18-133 Introduction to Digital Systems, than either of the other two fundamentals classes. This is simply because the existing course already had the right overall emphasis: a "vertical slice" through the layers of abstraction that comprise computer design: a gradual, bottom-up evolution from 0s and 1s up to basic processor architecture; an integrated hardware and software laboratory that closely tracks the lectures; and a successful, motivating central focus on building up the component pieces of a simple but real CPU. In the past, this course was our students' first exposure to ECE, and their first hands-on hardware laboratory. We believe the new course will benefit from the exposure our Freshmen will already have to basic digital elements and real design problems. Hence, the new course is a slightly more aggressive version of this existing course that relies on students' recently acquired background to revisit digital design ideas in more depth and with a greater emphasis on systematic analysis and synthesis.

The centerpiece of the course is a simple CPU called the P-18140\(^9\) which is dissected in lecture. Students reach this point after they have seen classical combinational and sequential logic design methods, and have completed a sequence of basic combinational and sequential hardware design projects in laboratory sessions. We use the processor to explain basic computer organization, data path and control path design techniques, and assembly language programming. A version of the processor exists as an interactive software simulator on which students design, debug and run assembly language programs. (We have also recently begun to consider the possibility of implementing the CPU, which tends to evolve semester by semester, as a field programmable gate array, and making this hardware available to students along with the simulator. This would allow some interesting speed comparisons of simulation versus actual execution.) The processor is itself microcoded. To complete their assembly language programming projects, students are required to design and implement their own microcode for a few of the P-18140's native instructions whose microcode we omit from the simulator. Students then demonstrate these instructions as part of a small assembly language program.

The course will have lectures, a laboratory session, and a recitation section. Overall, the course is designed to demystify computers for our students. By building up the concepts that define and inform each layer of the conventional design hierarchy—combinational circuits, sequential circuits, the register transfer level, stored program computing, control path / data path partition and implementation, rudimentary instruction set architecture and assembly language programming—we provide a solid foundation for students who wish to pursue more advanced computer engineering courses, and for those whose interests lie elsewhere in ECE.

\(^9\) Formerly the P-18133, the processor is always named after the current course number.
PRELIMINARY SYLLABUS FOR FUNDAMENTALS OF COMPUTER ENGINEERING

Binary number systems
- Positional number systems; useful radices: unsigned binary, octal and hexadecimal
- Signed numbers: sign magnitude, ones complement and twos complement

Combinational circuit design
- SSI logic gates
- Boolean algebra and canonical forms
- Minimization via Karnaugh maps
- Nontraditional logic implementation forms
- MSI building blocks: multiplexers, decoders, ROMs and PLAs
- Arithmetic circuits

Sequential circuit design
- Introduction to behavior of sequential circuits
- Basic latches and flip flops, latch/FF timing and triggering
- Sequential (state machine) design methods
- MSI sequential parts: registers, shift registers, counters

Processor design
- Register-transfer level ideas, abstractions, design style
- Stored program computers, data path and control path partitions
- Example processor design: P-18140 instruction set
- General data path design techniques, the P-18140 data path design
- General control path design techniques: hardwired control, microprogrammed control; the P-18140 microprogrammed control path design

Assembly language programming
- Basics of storing, manipulating, moving data on P-18140 processor
- Basics of control flow: straight-line code, conditional branches and loops
- Breaking programs into manageable pieces: subroutines, stack management; links to higher-level languages
Detailed curriculum requirements

In the following, we enumerate completely the requirements and constraints specified in the new ECE curriculum.

Humanities & Social Science (H&SS) requirements

These requirements remain unchanged. Eight courses must be taken in all. One course must be chosen in each of four areas, from a set of allowed classes. These H&SS breadth areas are:

- Writing and literature
- History
- Social sciences
- Psychology and philosophy

In addition, a set of three related H&SS (or fine arts) classes must also be selected to constitute a so-called depth sequence.

Finally, one free H&SS elective slot must be filled.

Mathematics, science and computer programming requirements

In all, students are required to take seven classes:

- Introduction to Programming (co-requisite for Introduction to ECE)
- Calculus I (co-requisite for Introduction to ECE)
- Calculus II
- Physics I (co-requisite to most other CIT engineering department’s Freshman Introduction to Engineering course)
- Physics II
- Linear Algebra (co-requisite for Fundamentals of Electrical Engineering)
- Introduction to Modern (Discrete) Mathematics (co-requisite for Fundamentals of Computer Engineering)

In contrast to our existing curriculum, many of these classes are corequisites to real engineering classes, starting from the first semester of the Freshman year.

Freshman engineering requirements

Two requirements for obtaining a BSECE are:

- At least two CIT departmental introductory engineering courses must be taken during the Freshman year.
- The ECE Freshman course, Introduction to Electrical and Computer Engineering, must be taken by ECE majors, although it may be taken as late as the Fall semester of the Sophomore year.
In other words, a typical student will take two of these introductory engineering courses in her Freshman year, and one of them will be our ECE course. However, the student unsure of her major may end up taking three of these courses, without penalty, and defer deciding on ECE until the Fall of her Sophomore year. This places an implicit upper bound on the length of required course sequences: each CIT department must not require a sequence deeper than five courses. This poses no additional constraints on the new ECE curriculum.

For completeness, the list of all CIT Freshman introductory engineering courses follows. Each course is permitted to have one science and one mathematics corequisite:

- **Introduction to Chemical Engineering**, corequisites **Chemistry I and Calculus I**
- **Introduction to Civil Engineering**, corequisites **Physics I and Calculus I**
- **Introduction to Electrical and Computer Engineering**, corequisites **Introduction to Computer Programming and Calculus I**
- **Introduction to Engineering and Public Policy**, corequisites **Physics I and Calculus I**
- **Introduction to Mechanical Engineering**, corequisites **Physics I and Calculus I**
- **Introduction to Metallurgical Engineering and Material Science**, corequisites **Physics I and Calculus I**

**ECE core requirements**

There are only two required core courses:

- **Fundamentals of Electrical Engineering**, corequisite **Linear Algebra**
- **Fundamentals of Computer Engineering**, corequisite **Introduction to Modern (Discrete) Math**

**ECE breadth requirements**

Before describing the breadth requirement in detail, it is best to illustrate what “breadth” now means in ECE by referring to the illustration in FIGURE 6. Rather than simply partitioning the department into EE and CE halves, we have chosen to restructure it as a spectrum of five areas. Going to the left in the figure takes us more toward EE topics; going to the right takes us toward CE topics. Courses from other departments appear at the far left and far right of the diagram, where our department has obvious interdisciplinary links to other departments at Carnegie Mellon, notably Physics and Computer Science. The idea is that these five areas have unique problems, methods, mathematics and modes of thinking. Courses allocated to an area share some of these attributes. By requiring students to take courses in three of these five areas, we enforce a solid and consistent notion of breadth, without having to resort to requiring specific courses.

---

10. Eight semesters total minus three semesters of introductory engineering courses leaves five semesters for department-specific specialization.
Specifically, to satisfy the ECE breadth requirement, students must take at least one first-level course in each of three of the five basic areas of ECE:

- **Physics**: which includes courses in electromagnetics, solid state devices, magnetics, data storage, optics, etc. (This area also includes some courses from the Physics department itself, for example, courses in topics such as quantum mechanics and solid state physics.)
- **Signals and systems**: which includes courses in signals and systems fundamentals, as well as control, communication, signal processing, robotics, etc.
- **Circuits**: which includes courses in both analog and digital electronics, as well as IC and VLSI design, some circuits-related CAD classes, etc.
- **Computer hardware**: which includes courses in digital design and verification, computer architecture, processor design, concurrency, some systems-oriented VLSI design, etc.
- **Computer software**: which includes programming, data structures, formal methods, software engineering, compilers, operating systems, etc. (These courses are all offered by the Computer Science department, with whom we maintain a reciprocal relationship: CS students take several hardware-oriented ECE classes, and ECE students take several software-oriented CS classes.)

Courses in these areas are referred to as *breadth* courses. Revisiting FIGURE 6, we see this spectrum of five areas, with several representative types of courses listed in each
Detailed curriculum requirements

area. Courses listed near the top of each area are more introductory; each area offers at least one, and possibly several such courses that can be taken starting with the three ECE fundamentals classes as prerequisites, along with perhaps some additional mathematics or science courses. Courses near the bottom of each area are more advanced, and require some earlier area-specific ECE breadth courses as prerequisites. A complete list of breadth courses is given in CHAPTER 6.

ECE coverage, depth and design requirements

At least three more courses must be taken from the areas defined in the ECE breadth requirement. (Recall that these include all ECE courses, plus most CS courses and some courses from Physics.) The idea here is to ensure students see enough engineering courses to be called "engineers" when they graduate. Two additional requirements, the depth and design requirements, may be satisfied by dedicating two coverage courses to depth and design, respectively, as long as these depth and design courses\textsuperscript{11} are selected from courses listed among the five ECE breadth areas.

To satisfy the depth requirement, students must take at least one course that has as a prerequisite one of the courses used to meet the breadth requirement. Since many fundamental topics in our curriculum are covered in two-semester sequences, practically speaking, the depth requirement will cause students to complete at least once such sequence. Of course there are other ways to satisfy this requirement, but this is likely to be the most common scenario.

Carnegie Mellon has a long tradition of offering aggressive and challenging design courses. Our design requirement is satisfied when a student completes one course from an approved list of design courses across ECE. (Again, this list may include courses from outside of ECE, notably Computer Science classes.) In addition a college-level CIT Honors Project, available to students with strong academic records, may be substituted for this requirement. A complete list of approved-design courses is given in CHAPTER 6. These courses tend to be on the high end of our curriculum, and are generally quite popular. We thus expect that students will start to take aim at these courses sometime during their Junior year, carefully picking up the required mathematics, science and ECE prerequisites. As a practical matter, attendance in some of our design courses will probably require more preparatory ECE classes than the minimum specified by the breadth and coverage requirements of the BSECE.

Workload, overload policy, QPA policy

The curriculum is designed around the assumption that students will take four courses per semester. Typically, but not necessarily, this will involve one H&SS course, and three technical or free elective courses per semester, for eight semesters. Our assumed model is that the one H&SS class is 9 units, and the remaining three classes are each 12

\textsuperscript{11} A minor point: a few courses, such as the college-level Senior honors project and design courses in some other departments; will count for the capstone design requirement, but will not then be able to be counted toward the ECE coverage requirement, which mandates courses from "inside" ECE, as defined in the broad sense of our five breadth areas. Most design courses, however, will count for both coverage and capstone design requirements.
units. Eight semesters at this workload is thus a 360 unit degree. In fact, during some semesters students may not always be able to take three 12 unit courses because, for example, many science and mathematics courses are only 9 units. In these cases the student may still end up taking five classes, for example, three 9 unit classes and two 12 unit classes. Our intent in establishing a workload norm and overload policy is to make four courses per semester the norm rather than the exception, especially for students taking several challenging ECE courses in the same semester.

The term “overload” refers to the maximum number of units per semester a typical student is allowed to carry, without getting special permission. In the existing curriculum, only students with a high quality point average (QPA) are allowed to overload. In the new curriculum, we modify this somewhat. First, an overload is defined as any schedule with more than 54 units in one semester. This amounts to anything more than three 12 unit classes and two 9 unit classes in the same semester. Second, we will now only permit students to overload if they achieved a QPA of 3.5 out of 4.0 in the previous semester. We want the privilege of taking an extra class to be regarded as a reward for exemplary performance, specifically, a high QPA in the previous semester. More importantly, we want to encourage students to focus more deeply on fewer courses, and so we are consciously erecting some barriers to prevent ordinary students from getting in over their heads. We have had some experiences with very bright students who achieve a high overall QPA early in their studies, then start overloading every semester, only to watch their QPA crash as they refuse to acknowledge that they are overwhelmed. The new overload policy should resolve such problems.

In order to graduate, students must maintain a 2.0 QPA in the set of courses used to satisfy the ECE Freshman, ECE core, breadth, depth, content and design requirements.
CHAPTER 4

Curriculum Templates: Example Paths through the New ECE Curriculum

To illustrate further some of the important features of the new curriculum, we present here several sample curriculum templates. Each template shows one possible path to a four year Bachelors degree in Electrical and Computer Engineering comprising eight semesters with four courses per semester. Because our exact course offerings are still evolving, the specific higher-level courses, and actual timing of when these courses are taken in the templates should not be considered precise. Nevertheless, the templates do serve to suggest ways that the flexibility inherent in the new curriculum can be used to satisfy a wide variety of goals.

Notational conventions

For clarity, courses in our templates are labeled to indicate their origins; see FIGURE 7. Humanities and Social Science courses are all labeled as H&SS in a box marked with a light cross. ECE courses, and courses in other departments such as Computer Science that are considered to be in core ECE areas, appear in shaded boxes with the appropriate title. All other courses appear without shading or highlights.

FIGURE 7

<table>
<thead>
<tr>
<th>TEMPLATE NOTATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>H&amp;SS</td>
</tr>
<tr>
<td>Fields I</td>
</tr>
<tr>
<td>Physics I</td>
</tr>
</tbody>
</table>

Humanities and Social Science
ECE
Other
A traditional electrical engineering curriculum

Perhaps the first observation to make is that a flexible curriculum in no way prevents a student from choosing a “traditional” sequence of courses. Hence, the course template in FIGURE 8 shows how a conventional Electrical Engineering program—conventional except, of course, for the existence of real engineering classes in the Freshman year—can be constructed within the constraints of the proposed ECE curriculum. This template shows the student pursuing the usual component of mathematics, physics, computer programming and ECE Fundamentals classes, as well as substantial breadth and depth in electromagnetics, circuits, signals and systems, and solid state. This curriculum culminates in a Senior year with a capstone design project in the area of controls, for example, as well as additional breadth classes in communications and magnetics. By the standards of any traditional Electrical Engineering curriculum, this is an exceptionally broad, solid plan of study. This program, or something similar, would be the “default” program suggested by an adviser to a student who is interested in traditional Electrical Engineering without more specific interest areas.
FIGURE 8

A TRADITIONAL ELECTRICAL ENGINEER

FRESHMAN

Fall
- H&SS
- Intro ECE
- CS Intro Prog
- Calc I

Spring
- H&SS
- Intro Met Eng & Mat Science
- Physics I
- Calc II

SOPHOMORE

Fall
- H&SS
- Fund of EE
- Physics II
- Linear Algebra

Spring
- H&SS
- Fund of CE
- Discrete Math
- Calc 3D

JUNIOR

Fall
- H&SS
- Fields I
- Analog Circuits I
- Signals & Systems I

Spring
- H&SS
- Fields II
- Digital Circuits I
- Signals & Systems II

SENIOR

Fall
- H&SS
- Solid State I
- Probability & Statistics
- Control Sys Lab

Spring
- H&SS
- Solid State II
- Digital Communic Sys
- Magnetics I
A traditional computer engineering curriculum

A "traditional" Computer Engineering plan of study can also be formulated within the framework of the proposed ECE curriculum. Following the usual mathematics, physics, introductory programming and ECE Fundamentals classes, the computer engineering student pursues a variety of computer hardware and computer software topics. Note that the ECE breadth requirement (which mandates that students elect one course in three of the five different ECE breadth areas) will not permit a course of study so narrow that only hardware-related topics, or only software-related topics are selected to the exclusion of other areas. Here, the breadth requirement means that even after taking courses in the computer hardware and computer software areas, a third area must be selected for study. In the template shown in FIGURE 9, the student attains this breadth in the circuits area, after having acquired the necessary preparation in the Fundamentals of EE class required of all ECE students. The Senior year culminates in a capstone design project in digital systems design, along with yet more depth and breadth in circuits, software and hardware. Again, by the standards of any traditional Computer Engineering program, this is a solid and well-balanced course of study.
A novel preparatory curriculum: 
the ECE core generalist

We have seen that the flexibility inherent in the proposed ECE curriculum does not preclude traditional avenues of study. Now let us consider instead new avenues of study that it creates. One possibility is illustrated by the course template shown in FIGURE 10 which we call a “Preparatory Curriculum.” Such a curriculum has exactly the same connotation that a traditional liberal arts curriculum usually carries: a broad, solid course of study emphasizing a variety of different areas that is worthy of a degree in itself, but undertaken specifically as preparation for a subsequent professional degree. It has become increasingly obvious in recent years that not all students who pursue a Bachelors degree in ECE actually enroll in Electrical or Computer Engineering for the rest of their lives. Moreover, we are seeing significantly more students who consciously choose an engineering degree as preparation for post-graduate study in other professional disciplines such as law, business, or medicine. In the past, such preparatory curricula were, it seems, exclusively undertaken under the rather broad umbrella of liberal arts. We are specifically interested now in supporting an ECE degree as a general preprofessional degree—but without compromising the integrity of the ECE Bachelors degree itself.

The plan of study shown here is one middle-ground approach. Observe that this student still takes the usual mathematics, physics, computer programming and ECE Fundamentals classes. In addition, the breadth, depth, ECE content and capstone design requirements are met in this case with a combination of courses in signals and systems, circuits, computer hardware and computer software. Note that the emphasis in such a curriculum is breadth of experience, and exposure to many different technical topics. Nevertheless, the result is not a weak or simplified program, but rather, a program for what might be called the “ECE Core Generalist.” Significant here is that the curriculum still leaves room for seven elective courses, in addition to Carnegie Mellon University’s already stronger-than-average requirement of eight H&SS classes. With appropriate choices among these fifteen non-technical classes, a very strong preparatory program can be created. In the example shown, we have populated the seven elective courses with economics, political science and a foreign language. Of course, some of these seven elective choices could also be accomplished within the eight H&SS courses themselves; the point worth noting is that taken together, the H&SS and elective slots provide considerable flexibility. With appropriate course choices, this course of study will produce students with a solid background in ECE core material, along with good preparation for further study in, for example, business or law.
## FIGURE 10

AN ECE CORE GENERALIST

<table>
<thead>
<tr>
<th>FRESHMAN Fall</th>
<th>H&amp;SS</th>
<th>Intro ECE</th>
<th>CS Intro Prog</th>
<th>Calc I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>H&amp;SS</td>
<td>Intro Engin &amp; Public Policy</td>
<td>Physics I</td>
<td>Calc II</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SOPHOMORE Fall</th>
<th>H&amp;SS</th>
<th>Fund of EE</th>
<th>Physics II</th>
<th>Linear Algebra</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>H&amp;SS</td>
<td>Fund of CE</td>
<td>Discrete Math</td>
<td>Fund. of CS I</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>JUNIOR Fall</th>
<th>H&amp;SS</th>
<th>Signals &amp; Systems I</th>
<th>Digital Circuits I</th>
<th>Foreign Language I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>H&amp;SS</td>
<td>Computer Arch I</td>
<td>Analog Circuits I</td>
<td>Foreign Language II</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SENIOR Fall</th>
<th>H&amp;SS</th>
<th>Control &amp; Instrument Lab</th>
<th>Economics I</th>
<th>Foreign Language III</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>H&amp;SS</td>
<td>Political Science</td>
<td>Economics II</td>
<td>Foreign Language IV</td>
</tr>
</tbody>
</table>
A focused preparatory curriculum: the premedical student

This template shown in FIGURE 11 is a more specific example of the Core-Generalist template presented in FIGURE 10 in the previous section. The student in this program is specifically tailoring an ECE program toward a career in medicine. Notice, for example, the choice of the *Introduction to Chemical Engineering* class as a Freshman engineering elective (in addition to the *Introduction to ECE* Freshman course). This requires a chemistry class as a corequisite, which displaces an H&SS class to the Spring of the Sophomore year. In addition to the usual mathematics, physics, programming and ECE Fundamentals classes, this student pursues breadth in signals and systems, circuits, and computer software. Depth in signals and systems ultimately leads to a capstone design course in controls and instrumentation. Observe that it is possible to pursue a fairly traditional ECE core that also meshes nicely with the overall interest in medicine. In particular, using the elective slots, this student manages to take three biology classes and five chemistry or chemical engineering classes, to complete preparation for medical school.
A focused preparatory curriculum: the premedical student

**FIGURE 11** AN ECE PREMEDICAL STUDENT

<table>
<thead>
<tr>
<th>FRESHMAN</th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Intro ECE</td>
</tr>
<tr>
<td></td>
<td>Chemistry I</td>
<td>Intro Chemical Engineering</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SOPHOMORE</th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Fund of EE</td>
</tr>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Fund of CE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>JUNIOR</th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Signals &amp; Systems I</td>
</tr>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Signals &amp; Systems II</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SENIOR</th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Control Systems</td>
</tr>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Control &amp; Instrument Lab</td>
</tr>
</tbody>
</table>
A curriculum with one year of study abroad

Carnegie Mellon University has recently strengthened its commitment to the concept of spending one year abroad during undergraduate studies. Foreign study can be extremely rewarding, but structural impediments in an overly rigid curriculum can render such plans unattractive to even the most committed student. The template shown in FIGURE 12 is merely one example of how this might be accommodated in the new ECE curriculum. Using the flexibility in the proposed curriculum, we have "stacked" the Junior and Senior years as shown here. During the Junior year spent abroad, this student pursues humanities and foreign language studies. Upon returning, the student plunges back into a technical program, in this example, a plan of study emphasizing computer hardware, with breadth in computer software, circuits, and signals and systems. Of course, other strategies are possible, but the point to note is that the proposed curriculum can adapt itself to students intent on pursuing foreign studies.
A curriculum with one year of study abroad

FIGURE 12  AN ECE STUDENT WHO STUDIES ABROAD FOR ONE YEAR

FRESHMAN

<table>
<thead>
<tr>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>H&amp;SS</td>
<td>Intro ECE</td>
</tr>
<tr>
<td></td>
<td>CS Intro Prog</td>
</tr>
<tr>
<td></td>
<td>Calc I</td>
</tr>
<tr>
<td>H&amp;SS</td>
<td>Intro Engin &amp; Public Policy</td>
</tr>
<tr>
<td></td>
<td>Physics I</td>
</tr>
<tr>
<td></td>
<td>Calc II</td>
</tr>
</tbody>
</table>

SOPHOMORE

<table>
<thead>
<tr>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>H&amp;SS</td>
<td>Fund of EE</td>
</tr>
<tr>
<td></td>
<td>Physics II</td>
</tr>
<tr>
<td></td>
<td>Linear Algebra</td>
</tr>
<tr>
<td>H&amp;SS</td>
<td>Fund of CE</td>
</tr>
<tr>
<td></td>
<td>Discrete Math</td>
</tr>
<tr>
<td></td>
<td>Fund. of CS I</td>
</tr>
</tbody>
</table>

JUNIOR YEAR ABROAD

<table>
<thead>
<tr>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>H&amp;SS</td>
<td>Foreign Language I</td>
</tr>
<tr>
<td></td>
<td>Foreign Language III</td>
</tr>
<tr>
<td></td>
<td>H&amp;SS</td>
</tr>
<tr>
<td>H&amp;SS</td>
<td>Foreign Language II</td>
</tr>
<tr>
<td></td>
<td>Foreign Language IV</td>
</tr>
</tbody>
</table>

SENIOR

<table>
<thead>
<tr>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signals &amp; Systems I</td>
<td>Computer Arch I</td>
</tr>
<tr>
<td>Logic Design &amp; Verific</td>
<td>Fund. of CS II</td>
</tr>
<tr>
<td>Signals &amp; Systems II</td>
<td>Computer Arch II</td>
</tr>
<tr>
<td>Digital System Design Proj</td>
<td>Digital Circuits I</td>
</tr>
</tbody>
</table>

A NEW ECE CURRICULUM FOR CARNEGIE MELLON 45
A curriculum with specific technical focus: the CAD / VLSI designer

The flexibility in the new ECE curriculum allows an ECE Bachelors program to be used as preparation for other post-graduate professional studies. However, the skeptical reader should not now jump to the conclusion that the new ECE curriculum is thus specifically tailored to those students who will not pursue engineering as their livelihood, to the detriment of those who will actually become practicing Electrical and Computer Engineers. The critical attribute of a flexible curriculum is that it can adapt to a variety of students. Accordingly, we now consider yet another kind of curriculum: the "focused" program of study that emphasizes particular technical strengths at Carnegie Mellon.

The template shown in FIGURE 13 focuses on Computer-Aided Design (CAD) for Very Large Scale Integrated Circuits (VLSI). Carnegie Mellon's ECE department is an internationally recognized leader in this area, and a number of our faculty devote their research to its problems. The plan of study illustrated here has substantial breadth and depth in the areas of circuits, computer software and computer hardware. The Senior year culminates in a capstone design course in silicon system design that integrates all these skills. In addition, the Senior year is also used to attain CAD-specific depth, in this case in circuit simulation, as well as general circuits-related breadth in signals and systems, electromagnetics and numerical methods. Note that this highly focussed plan of study sacrifices none of the general breadth or exposure one would expect of a strong undergraduate ECE degree.
### A CAD/VLSI Designer

#### Freshman

<table>
<thead>
<tr>
<th>Fall</th>
<th>H&amp;SS</th>
<th>Intro ECE</th>
<th>CS Intro Prog</th>
<th>Calc I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>H&amp;SS</td>
<td>Intro Mech Engineering</td>
<td>Physics I</td>
<td>Calc II</td>
</tr>
</tbody>
</table>

#### Sophomore

<table>
<thead>
<tr>
<th>Fall</th>
<th>H&amp;SS</th>
<th>Fund of EE</th>
<th>Physics II</th>
<th>Linear Algebra</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>H&amp;SS</td>
<td>Fund of CE</td>
<td>Discrete Math</td>
<td>Fund. of CS I</td>
</tr>
</tbody>
</table>

#### Junior

<table>
<thead>
<tr>
<th>Fall</th>
<th>H&amp;SS</th>
<th>Digital Circuits I</th>
<th>Logic Design &amp; Verification</th>
<th>Fund. of CS II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>H&amp;SS</td>
<td>Digital Circuits II</td>
<td>Computer Arch I</td>
<td>Calc 3D</td>
</tr>
</tbody>
</table>

#### Senior

<table>
<thead>
<tr>
<th>Fall</th>
<th>H&amp;SS</th>
<th>Silicon Sys Design Proj</th>
<th>Circuit Sim Theory &amp; Prac</th>
<th>Fields I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>H&amp;SS</td>
<td>Numerical Methods</td>
<td>Signals &amp; Systems I</td>
<td>CS Algorithms</td>
</tr>
</tbody>
</table>
Another curriculum with technical focus: 
the data storage systems designer

FIGURE 14 shows a template for a plan of study that focuses on data storage systems, with particular emphasis on magnetic recording media and the computer applications that use these media. This program is another example of a focused technical curriculum that exploits particular strengths at Carnegie Mellon. The ECE department at Carnegie Mellon houses a National Science Foundation-supported Engineering Research Center called the Data Storage Systems Center. This center is the focus of interdisciplinary work on a variety of topics in data storage, from the material properties of magnetic media to the architecture of distributed file systems for computer networks. The template shown here is an example of how the new ECE curriculum can support interdisciplinary studies, surmounting some of the barriers between disciplines. The interesting feature of this course of study is its two seemingly disparate areas of concentration: magnetic media and computer systems. Normally, one just does not expect a student to take courses in, say, device physics and computer operating systems. And yet, engineers with precisely such an interdisciplinary background would be ideally positioned to understand and attack the interesting problems in data storage system design. This student achieves breadth in physics, circuits, computer hardware and computer software, as well as depth in relevant physics- and computer-related topics. The Senior year culminates in a capstone design project in data storage systems.
Another curriculum with technical focus: the data storage systems designer

**FIGURE 14** A DATA STORAGE SYSTEMS DESIGNER

<table>
<thead>
<tr>
<th></th>
<th>FRESHMAN Fall</th>
<th>Spring</th>
<th>SOPHOMORE Fall</th>
<th>Spring</th>
<th>JUNIOR Fall</th>
<th>Spring</th>
<th>SENIOR Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Intro ECE</td>
<td>CS Intro Prog</td>
<td>Calc I</td>
<td>H&amp;SS</td>
<td>Fund of EE</td>
<td>Physics II</td>
<td>Linear Algebra</td>
</tr>
<tr>
<td></td>
<td>Intro Met Eng &amp; Mat Science</td>
<td>Physics I</td>
<td>Calc II</td>
<td>H&amp;SS</td>
<td>Fund of CE</td>
<td>Discrete Math</td>
<td>Calc 3D</td>
<td></td>
</tr>
<tr>
<td></td>
<td>H&amp;SS</td>
<td>Fund of EE</td>
<td>Physics II</td>
<td>Linear Algebra</td>
<td>H&amp;SS</td>
<td>Fields I</td>
<td>Analog Circuits I</td>
<td>CS Fund I</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fund of EE</td>
<td>Physics II</td>
<td>Linear Algebra</td>
<td>H&amp;SS</td>
<td>Fields II</td>
<td>Solid State I</td>
<td>Computer Arch I</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Analog Circuits I</td>
<td>CS Fund I</td>
<td>CS Fund II</td>
<td>H&amp;SS</td>
<td>Magnetics I</td>
<td>Adv. Physics or Materials Sci</td>
<td>CS Operating Systems</td>
</tr>
</tbody>
</table>

A NEW ECE CURRICULUM FOR CARNEGIE MELLON
A program for the late starter

All incoming Freshmen in the CIT must take two introductory engineering courses during their Freshman year. It is possible that a student may not take the introductory ECE course during the Freshman year but still decide to pursue an ECE degree. The template shown in FIGURE 15 shows a program that accommodates such a student.
## A Late Starting ECE Student

### Freshman
- **Fall**
  - H&SS
  - Intro Mech Engineering
  - Physics I
  - Calc I
- **Spring**
  - H&SS
  - Intro Mat Eng & Mat Science
  - Physics II
  - Calc II

### Sophomore
- **Fall**
  - H&SS
  - Intro ECE
  - Calc 3D
  - CS Intro Prog
- **Spring**
  - H&SS
  - Fund of EE
  - Linear Algebra
  - Discrete Math

### Junior
- **Fall**
  - H&SS
  - Fund of CE
  - Analog Circuits I
  - Signals & Systems I
- **Spring**
  - H&SS
  - Fund of CS I
  - Digital Circuits I
  - Signals & Systems II

### Senior
- **Fall**
  - H&SS
  - Computer Arch I
  - Fund of CS II
  - Fields I
- **Spring**
  - H&SS
  - Computer Arch II
  - Probability & Statistics
  - Control & Instrument Lab
CHAPTER 5

Implementation Issues

Implementation of the new ECE curriculum began in the Fall semester of 1991. The new curriculum will be phased in one year at a time; in the 1991-1992 academic year our entering Freshmen will be pursuing the new curriculum, while our Sophomores, Juniors and Seniors still pursue our old curriculum. We summarize here some of the issues we have confronted and will continue to confront as this implementation process unfolds.

Freshman engineering

All departments in CIT began offering their Freshman introductory engineering courses in Fall 1991. These courses will be offered every semester so that entering Freshmen can pick freely among the alternatives in either semester of their starting year. The ECE course is heavily staffed with faculty and teaching assistants. So far, students are responding well to the class. They seem to appreciate the attention of the department this early in their education, and really like the idea of building something real (the mobile robot) and integrating theory with practice.

We will be using the first few offerings of the course to gauge and refine our syllabus and our expectations, and ensure that we are teaching—and that there is time to cover without rushing—everything the students need to know for the next two ECE courses: Fundamentals of EE and Fundamentals of CE. Interesting low-level problems are already occurring and being recorded, e.g., during the first laboratory session, where students learned to solder and were supposed to solder some IC sockets onto their robot circuit boards, a few students in each lab mistakenly soldered their chips in instead. This simple error points out clearly the vigilance we must continue to exercise about making assumptions, using terminology and jargon, etc., when interacting with our bright but inexperienced Freshmen.
Timing ECE course offerings

We are a sufficiently small department that we cannot offer every course every semester. Nevertheless, we will offer the three fundamentals classes of the new ECE core every semester, to accommodate students who come into ECE via different sequences of courses during their Freshman year and the Fall semester of their Sophomore year. We will also change the way in which we time the offerings of the core sequences in each of ECE’s breadth areas. In the past, we have synchronized each multiple-course sequence so that the starting courses, e.g., Fields I, Signals I, were only offered in the Fall. This rigidity is now highly undesirable since the new curriculum has almost no requirements for students to take particular course sequences in a particular order or in a particular semester. Indeed, the flexibility in the new curriculum means students can arrive at most ECE course sequences from a variety of paths. Hence, we will stagger the offerings of these course sequences so that some start in Fall, others in Spring. The idea is to ensure that every semester there is a good selection of breadth courses for students to begin.

In addition, changes in individual courses are being negotiated among the faculty in each of the breadth areas. For example, the existing two-course Signals and Systems sequence, consisting of two 9-unit courses, may collapse to a single 12-unit course that is offered every semester to serve as a gateway into higher-level courses in this area. Similarly, there is debate in the Computer Hardware area about offering a richer selection of advanced courses by offering some of the most advanced courses on a rotating basis, every other year. Again, the goal is to ensure that every semester there are several interesting, accessible classes. These discussions will continue, and decisions will be made as subsequent years of the new curriculum are phased in.

Advising

An interesting new problem we face is that a highly flexible curriculum is a more difficult curriculum to advise. Especially during the highly regimented Sophomore and Junior years of our existing ECE curriculum, too many of our students treat their faculty advisers as “that person who signs my registration forms,” a situation arising at least in part because few strategic, career-influencing course choices can be made. This situation will change dramatically with the new curriculum. We strongly hope the new curriculum will better engage our students, since they will be required to make more strategic choices earlier in their education, which will require more interaction with their faculty advisers. We are also considering several new mechanisms to improve the advising process generally. These include:

- **Rotating area advisers**: Each student is currently assigned a single ECE faculty member for the duration of her education. We currently try to assign EE students to EE faculty, and CE students to CE faculty. As the technical diversity of students expands, we have considered adding an additional layer of floating advisers for each of the breadth areas, assigned on a rotating basis. The idea is to ensure that students know there is an individual responsible for, say, all the circuits-area classes, who can provide strong technical guidance about individual classes, topics, and career aspirations.
• **Curriculum previews:** Young students often know what topics interest them, but do not know how these topics map to standard courses in ECE. Many students are interested in robotics, for example, but few know that the way to pursue this interest is to take a set of fundamental signals and systems, control, and signal processing courses in our curriculum. We have discussed the idea of using the three ECE fundamentals classes to preview how topics are distributed across the higher-level courses in the curriculum. The robot project in *Introduction to ECE* is a particularly nice starting point for this, since it embraces many of the components of ECE.

• **Curriculum tracking software:** We have discussed providing students with a hypertext system to track their progress through the curriculum, access information about graduation requirements, prerequisites for desired higher-level courses, course descriptions, etc.\(^\text{12}\) We would like students to be able to update a floppy disk every semester with their current progress, and any changes in curriculum requirements. The system should also be able to support "what if" queries along the lines of "what must I take now if I want to take this interesting course when I am a Senior?" We are currently analyzing commercial offerings of such software, and the possibility of building a student-friendly interface that meet ECE’s peculiar needs.

---

**ABET accreditation**

As should be abundantly clear from the previous chapters, the new ECE curriculum we have designed departs in several ways from more traditional EE and CE curricula. These differences are the result of over two years of thoughtful deliberation by our faculty. These ideas have been debated and developed both by focused working groups within ECE, and by the ECE faculty as a whole. As a faculty, we are unanimous in our belief that the changes we are implementing are the right responses to the problems we face as engineering educators. Moreover, we are unanimous in our belief that the curriculum, as it is planned, is accreditable.

We have been particularly heartened by recent comments from ABET itself in support of thoughtful experimental curricula. A recent task force composed of ABET members, and representatives of the academic community chosen from the Engineering Deans Council (EDC) noted succinctly the perceived problems to be addressed, both with curriculum and with current accreditation practices themselves:

Recently, there has been a growing divergence of opinion between the engineering academic community, as represented by the Engineering Deans Council, and ABET, on matters critical to engineering education. This comes at a time when engineering education is facing significant challenges. In the future, engineering schools will have to attract and retain a more diverse student body with widely varying levels of preparation; engineering graduates will need a greater knowledge of foreign cultures and business practices as we compete in world markets; measures should be taken to alleviate the shortage of new, young American faculty in engineering schools; and engineering schools must take a role in reversing the current decreasing interest of high school students in engineering as a career. In order to deal with these problems, additional flexibility and innovation in engineering education are urgently needed.

---

\(^{12}\) In the database world such tools are usually called “academic audit” systems.
During the last years, ABET has sought to improve the quality of engineering education, yet many of the engineering deans have expressed the opinion that ABET stifles innovation. Many in academia criticize the "bean counting" nature of an ABET evaluation and the uniformity of undergraduate engineering curricula, with ABET criteria leaving little room for experimentation or new ideas.\textsuperscript{13}

We are in full accord with these concerns, and note that many of the salient points of our new curriculum—Freshman engineering courses to excite and motivate very young students, mathematics and science course as corequisites to engineering to enhance retention, the ability to defer ECE fundamentals courses until adequate preparation is achieved, enhanced flexibility to allow students actually to take courses in foreign languages and business practices—confront these issues head on.

The 1991 ABET/EDC Task Force finally concluded:

The Task Force firmly believes that engineering schools should be encouraged and rewarded for experimentation and innovation in their programs, rather than be stifled and penalized for trying new ideas. The Task Force believes that new approaches should be fostered and will be necessary to meet the challenges of the next decade. ABET criteria and their implementation must be flexible enough to allow for these new innovations.\textsuperscript{14}

We believe our new ECE curriculum is an excellent candidate for accreditation under these enlightened guidelines.


\textsuperscript{14} Ibid.
We provide here complete, though obviously tentative lists of the courses that satisfy the various requirements of the proposed ECE curriculum. At the time this section is being written, discussions among our faculty members are still actively underway concerning the best partition of topics to classes, especially for new upper-level ECE classes we will offer for the first time in late 1992 or early 1993.

### Specific course requirements

- 99-101 *Computing Skills Workshop*
- 18-100 *Introduction to Electrical and Computer Engineering*
- 18-120 *Fundamentals of Electrical Engineering*
- 18-140 *Fundamentals of Computer Engineering*
- 15-127 *Introduction to Programming and Computer Science*
- 21-121 *Calculus I*
- 21-122, *Calculus II*
- 21-341 *Linear Algebra*
- 21-127 *Introduction to Modern Mathematics* (usually referred to as “Discrete Mathematics” in preceding chapters, for clarity)
- 33-106 *Physics for Engineering Students I*
- 33-107 *Physics for Engineering Students II*
ECE breadth courses

Physics
Note that these are mostly ECE courses, but with a few courses from related departments such as Physics, and Metallurgical Engineering and Material Science (MEMS).

- 18-201 Engineering Electromagnetics I
- 18-201 Engineering Electromagnetics II
- 18-233 Introduction to Solid State Physics
- 18-334 Semiconductor Devices
- 18-341 Electromechanics
- 18-354 Introduction to Data Storage Systems
- 18-355 Data Storage Systems Measurement and Design Laboratory
- 18-362 High Frequency System Design
- 18-381 Computer-Aided Design of Electromagnetic Systems
- 27-432 Electrical, Magnetic, and Optical Properties of Materials
- 33-134 Quantum Physics I
- 33-435 Quantum Physics II
- 33-437 Intermediate Electricity and Magnetism I
- 33-438 Intermediate Electricity and Magnetism II
- 33-446 Advanced Quantum Physics
- 33-448 Introduction to Solid State Physics
- 33-453 Intermediate Optics

Signals and Systems

- 18-2xx Signals and Systems
- 18-301 Fundamentals of Control
- 18-308 Introduction to Digital Communication
- 18-310 Fundamentals of Communication Systems
- 18-313 Computer-Controlled Testing and Measurement System Design
- 18-314 Computer Control Systems Design Laboratory
- 18-3xx Discrete Control Systems
- 24-245 Dynamics of Physical Systems
- 24-246 Feedback Control Systems
- 24-247 Instrumentation and Design of Control Systems

Circuits

- 18-2xx Analog Circuits I
- 18-xxx Analog Circuits II
- 18-2xx Digital Circuits I
ECE capstone design courses

- 18-xxx  Digital Circuits II
- 18-2xx  Circuit Simulation
- 18-425  Integrated Circuit Design Project

**Computer Hardware**
- 18-247  Computer Architecture I
- 18-xxx  Computer Architecture II
- 18-349  Concurrency and Real-Time Systems
- 18-xxx  Logic Design and Verification
- 18-xxx  Digital System Design
- 18-xxx  Silicon System Design

**Computer Software**
Note that these are all courses from the School of Computer Science.
- 15-211  Fundamental Structures of Computer Science I
- 15-212  Fundamental Structures of Computer Science II
- 15-312  Programming Languages Design and Processing
- 15-381  Artificial Intelligence: Representation and Problem Solving
- 15-384  Artificial Intelligence: Robotic Manipulation
- 15-385  Artificial Intelligence: Computer Vision
- 15-411  Compiler Design
- 15-412  Operating Systems
- 15-413  Software Engineering
- 15-414 Structured Programming and Problem Solving
- 15-451  Algorithms
- 15-453  Formal Languages and Automata
- 15-612  Distributed Systems
- 15-682  Engineering of Knowledge Based Systems

**ECE capstone design courses**
Note that these are principally ECE courses, with a few technically-related courses not specifically construed as being in one of the five ECE breadth areas, such as the CIT Honors Project class.
- 18-300  Analysis Synthesis and Evaluation
- 18-313  Computer-Controlled Testing and Measurement System Design
- 18-314  Computer Control Systems Design Laboratory
- 18-355  Data Storage Systems Measurement and Design Laboratory
- 18-362  High Frequency System Design
- 18-371  Design Optimization Techniques
- 18-381  Computer-Aided Design of Electromagnetic Systems
- 18-425  Integrated Circuit Design Project
- 18-xxx  Digital System Design
- 18-xxx  Silicon System Design
- 15-413  Software Engineering
- 39-500  CIT Honors Research Project