Electrical and Computer Engineering at Carnegie Mellon– a New Curriculum

Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Preface

Although the American university system is the envy of the world, over the past few years, with rising tuition, the quality of teaching has begun to receive more attention. As a result, universities are re-examining their educational process. Student evaluations are taken seriously; some universities have instituted peer review of faculty teaching; and feedback from companies that hire our graduates is being sought.

At the same time, the nature of engineering is changing. Engineers are now not only expected to be able to solve problems, but to understand the implications of their solutions, to be able to work as a team with others from all aspects of product development, and to take a "systems" approach. This amounts to the need for a new culture for engineering in academia.

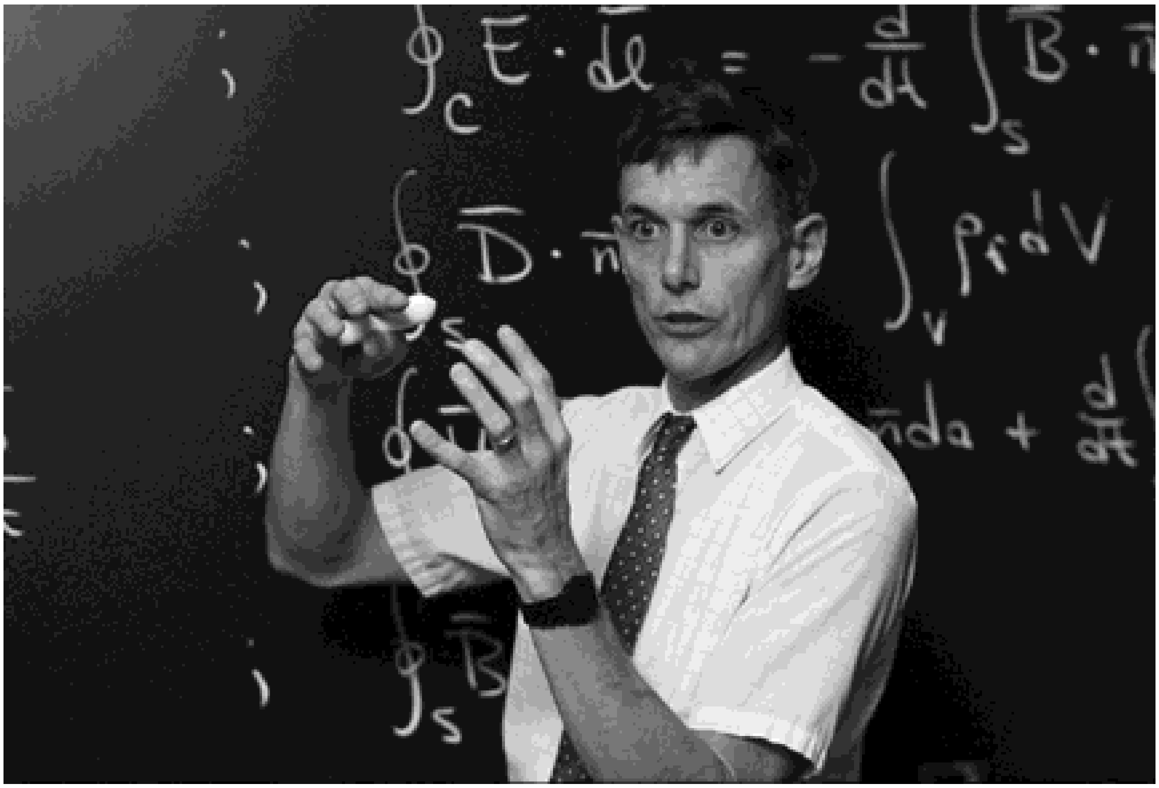
The Electrical and Computer Engineering Department at Carnegie Mellon has had a strong tradition for teaching, with each course being taught by a faculty member and each faculty member teaching at least one course each semester.

Thus, when the opportunity came for this department to "re-engineer" itself, the faculty was very enthusiastic. This booklet describes the result of a five-year effort to redesign and implement an entirely new undergraduate curriculum in electrical and computer engineering that takes into account this changed nature of engineering today. Not only has expanded knowledge led to more depth, but engineers are also being called upon to exercise more breadth as their careers develop. To address these challenges, the faculty has created a curriculum characterized by enormous flexibility. This has been, and continues to be, an exciting adventure in educational innovation.

Such a dramatic curriculum change as described here would not have been possible without a very enlightened and dedicated faculty. New courses had to be developed and interfaced with each other, which took a great deal of time and cooperation. The Department is indebted to Steve Director for his encouragement and the Wipe the Slate Clean Committee for its enormous contribution. I would also like to thank Professors Rutenbar, Kumar, Neuman, and Stancil and Sandra Salmonsen for their efforts in preparing this document.

Robert M. White
Department Head

Hamerschlag Hall, home of ECE.



Professor James Hoburg describing electromagnetic fields.

CHAPTER 1 Introduction

In the fall of 1991, after approximately two years of development, the Department of Electrical and Computer Engineering (ECE) at Carnegie Mellon University implemented a new curriculum that differed radically from its predecessor. Somewhat to our surprise, several hundred copies of our contemporaneous 1991 curriculum design document [1] were requested from colleagues around the world. Our proposed curriculum and its design philosophy came to be widely discussed outside of Carnegie Mellon, *e.g.*, [2] [3] [4] [5]. In this new document, we revisit the issues raised in 1991, but now with four years of experience and hindsight. As in [1], we again review the design of this curriculum, including a taxonomy of problems we needed to address, and a set of general principles we evolved to address those problems. We also now describe both the 1991 proposed curriculum and its current 1995 incarnation in detail, including implementation experiences. Finally, we offer new data from an ongoing analysis of its impact on students' curricula choices.

Wiping the Slate Clean

Current engineering practice has, by necessity, evolved to keep pace with technology: witness the rate at which fundamentally new ideas are introduced into new products. One might suppose, then, that current engineering education has also evolved to track such new developments. However, we argue that engineering education has really evolved only to the extent that individual engineering courses have been updated–usually with increased density of content–to reflect new developments.

The prevailing philosophy of engineering education–teach first the basics in mathematics and science, follow with exposition of engineering applications–has remained unchanged and unchallenged for more than four decades. While contributing to the creation of engineers who are current in specific technologies, we believe that the teaching of unmotivated math and science followed by incrementally updated technical courses, is fundamentally flawed. It contributes little to the education of engineers who can acquire new knowledge as necessary, cope with dynamically changing work environments or excel in nontraditional jobs. We believe that real impact in engineering education will be made only by looking at the curriculum *as a whole*, in the context of present technological and societal needs, and not just by constant repolishing of aging courses. There are advantages to be found in taking a fresh, unfettered look at the undergraduate curriculum.

Unfortunately, curricula have tremendous inertia and often resist all but the most incremental and cosmetic of changes. In October of 1989, the dean of the college of engineering at Carnegie Mellon (called the Carnegie Institute of Technology, or CIT) instituted a review process across all engineering departments. 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. Because of the breadth of this undertaking, each engineering department was allowed to consider the best possible curriculum changes, not merely those that could be wedged conveniently into its current web of requirements, prerequisites, constraints and customs. This document describes the design and implementation of the new Electrical and Computer Engineering bachelor's degree program that emerged from this process. This curriculum, which took approximately two years to fully design, was implemented within the Department of Electrical and Computer Engineering in the fall of 1991, and will produce its first four-year graduates in the spring of 1995.

Within ECE, the curriculum was designed by a committee whose quickly adopted name reflected the spirit of process: the *Wipe the Slate Clean Committee*. Composed of 11 faculty from across the breadth of the department's research and teaching areas, the committee interviewed both students and faculty and worked aggressively for roughly one year to dissect, analyze, disassemble and finally to redefine the ECE undergraduate curriculum. The committee had the following members:

L. Richard Carley	Rob A. Rutenbar (Chairman)
Stephen Director	T. Ehud Schlesinger
James Hoburg	Daniel Stancil
Pradeep Khosla	Jay Strosnider
B. V. K. Vijaya Kumar	Donald Thomas
Ronald Rohrer	

In May of 1990 a college-wide faculty retreat was convened to discuss the curriculum requirements that apply across engineering 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 was accepted with minor alterations. In the fall of 1991 the first class of new freshmen was admitted into this new curriculum.

Our Starting Point: The “Old” ECE Curriculum

At the beginning of 1991, the ECE Department offered two four-year Accreditation Board for Engineering and Technology (ABET) accredited bachelor of science degrees: the Bachelor of Science in Electrical Engineering (B.S.E.E.) and the Bachelor of Science in Computer Engineering (B.S.C.E.). Both curricula shared a common freshman year emphasizing mathematics, science and computer programming. They also shared a common core of engineering classes, emphasizing linear circuits, electronics, solid state devices, digital logic design and microprocessors. In addition, these curricula (as did all curricula in the colleges of engineering and science) shared common requirements for humanities and social sciences courses (called H&SS) that amounted to roughly one course per semester. An overview of these curricula appears in FIGURE 1.

FIGURE 1. Old (Pre-1991) Carnegie Mellon EE and CE Curricula

Electrical Engineering	Courses	Computer Engineering	Courses
Mathematics & Sciences		Mathematics & Sciences	
Calculus	2	Calculus	2
Differential Equations	1	Differential Equations	1
Linear Algebra	1	Linear Algebra	1
Probability	1	Probability	1
Physics	3	Modern Math	1
Chemistry	1	Physics	3
Computer Programming	1	Chemistry	1
		Computer Programming	1
Electrical & Computer Engineering		Electrical & Computer Engineering	
Intro Digital Systems	1	Intro Digital Systems	1
Linear Circuits	1	Linear Circuits	1
Intro Electronic Devices	1	Intro Electronic Devices	1
Electromagnetics	2	Computer Architecture	1
Signals & Systems	2	Concurrency & Real Time Systems	1
Analog Circuits	1	Digital Integrated Circuits	1
Digital Integrated Circuits	1	Logic & Processor Design	2
EE Elective	1	Computer Science	
Senior Design Elective	1	Fundamentals of CS	2
		CS Elective	1
Electives		Electives	
Freshman	2	Freshman	2
Engineering Science	2	Engineering Science	2
Technical	5	Technical	4
Free	1	Free	1
Humanities & Social Sciences	8	Humanities & Social Sciences	8

After this common core, the two curricula diverged. The B.S.E.E. emphasized traditional electrical engineering topics such as electromagnetics, analog circuits, and signals and systems. The B.S.C.E. emphasized computer hardware and software topics such as computer architecture, processor design, data structures and concurrency. Both curricula required several technical electives and a capstone design elective.

In 1991 about 40 percent of our students pursued the B.S.E.E., and about 50 percent pursued the B.S.C.E. Roughly 10 percent of our students chose to double major in both electrical engineering (EE) and computer engineering (CE). This was accomplished at the sacrifice of most elective classes: students completed the core requirements of one curriculum using the elective slots provided in the other. Also, a few of our students double majored in computer engineering and computer science (which is in a separate college at Carnegie Mellon). This essentially required that all elective classes in the B.S.C.E. curriculum were chosen to complete computer science core requirements.

Why Change?

By any traditional measure in 1991, the ECE Department was doing well educating its students. The department as a whole was consistently ranked among the country's top electrical engineering departments and components of the curricula were singled out in popular rankings of engineering specialties [6]. The department attracted outstanding undergraduate students: ECE was the first choice among engineering departments of most entering freshmen. Our graduates were recruited heavily by U.S. companies, and the ECE Department was on the list of must-visit departments for many companies that recruited only among a select set of elite schools. Our graduates who chose to pursue an advanced degree went on to elite graduate schools.

So, why did we undertake a substantial reorganization of our curriculum? The answers are not simple, nor are they independent. We categorize our broad concerns as follows. These can be regarded as the beginnings of a set of "specifications" for our new curriculum.

Remove Structural Impediments to 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 was certainly true of our 1991 electrical engineering and computer engineering 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 was that even incremental changes became difficult to implement.

In the 1991 B.S.E.E. and B.S.C.E. parallel curricula, even modest changes rippled in undesirable ways throughout the two programs. An example makes this concrete. As a result of an 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 electrical engineering students in our circuits and electronics courses, and broaden the background of our computer engineering students by exposing them to more noncalculus mathematics.

Unfortunately, this ideal proved impossible to implement. There was no small-scale alteration of the B.S.E.E. and B.S.C.E. 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 derived from the slight differences in the first few years of the B.S.E.E. and B.S.C.E. requirements. The B.S.C.E. student began to take computer science classes fairly early, so that junior and senior computer engineering courses were correctly synchronized with their computer science prerequisites. In contrast, the B.S.E.E. student had no such requirements. The end result was that we required our students to take a linear algebra class, but we did essentially nothing to exploit this background in other ECE core classes. This simple example makes clear how difficult it can become to achieve the goal of *uniform* mathematics, science and engineering core preparation for both B.S.E.E. and B.S.C.E. students.

Rationalize Requirements for Topical Coverage and Workload

As has become amply clear over the last decade, 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 the level of the bachelor’s degree[1]. Coming to grips with this accelerating problem was at the heart of our motivation for a significant restructuring of our curriculum.

The 1991 ECE curriculum required a large number of core classes, designed to ensure familiarity with a substantial subset of traditional EE and CE topics. After a great deal of argument and discussion, we came to believe that this approach, which implicitly assumes all students need exposure to (almost) all areas, was no longer credible as the core of a curriculum for the 21st century. Such a strategy mandates that we compress ever more material into the same number of classes. At Carnegie Mellon, a typical course is 9 to 12 units, where each unit represents one hour of work each week, *i.e.*, spent in a lecture or recitation, or a laboratory, or doing homework or project work. A 12-unit class is nominally intended to require 12 hours of effort per week. In the 1991 curriculum, many of our courses had already fallen victim to “units-creep,” *i.e.*, challenging classes meant to require 12 hours of work per week had inflated to require 15 or 18 hours of work from even the best of students. This was caused by well-meaning faculty working hard to give students the best, most thorough view of as many topical areas as they could–usually with the assumption that this was the only opportunity students would have to see the material.

While certainly not opposed to demanding classes, we concluded that the overall strategy of putting more material into the curriculum had become decreasingly effective. Students were being asked to absorb increasing amounts of material, which left 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 exacerbated this problem: we kept discovering that many of our students had never seen material fundamental to the background of our core courses. The end result was that by forcing students to juggle too many topics with too little time to master these topics, many students were learning even less material, less well.

---

1. An alternative is to extend the amount of time required to educate students to some minimum level of professional competence. Such an approach was advanced by the Massachusetts Institute of Technology in [7], which proposed a five-year accredited master’s program as the principal mechanism for educating entry-level engineers. We return to this idea in Chapter 7.

Emphasize Engineering Ideas Over Techniques

A related consequence of the explosion of material was that many students came to view their courses as a set of unrelated hurdles to be overcome. As a result, many students were acquiring 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 suggested that after first teaching a vast body of fundamental mathematics and science–which students absorbed like sponges–we were free to teach engineering principles, drawing as necessary on the deep well of basic knowledge internalized by the student in these early studies. This was (and is) a lovely idea, but depressingly unrealistic. Students often had 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. When a flood of engineering ideas was introduced on top of this precarious foundation, the outcome was often less than satisfactory. Too often, students only had 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–very large scale integrated circuits (VLSI), microelectromechanical systems (MEMS), electronic materials, computer-aided manufacturing, telecommunications networks, supercomputing–have strong and established interdisciplinary linkages. However, our 1991 curriculum did little to encourage the creation of engineers who could work comfortably across the boundaries of several disciplines.

The 1991 curriculum implicitly assumed that there were only two types of engineers: electrical engineers and computer engineers. These were produced by completion of a large, rigid core of electrical engineering or computer engineering classes. Although industry specifically, and society generally might have valued highly a student who had completed, say, 60 percent of the electrical engineering core classes and 40 percent of the computer engineering core classes, we had no mechanism for giving this broad individual a degree. Nor did we have any mechanism for coping with an even broader individual who might have wished to complete, say, 30 percent of the electrical engineering core, 30 percent of the computer engineering core, then a dozen classes in mechanical engineering, operations research and Japanese language, in preparation for a career in computer-aided manufacturing. Indeed, a key conclusion of the early discussion of the Wipe the Slate Clean Committee was that we would like not only to tolerate such individuals, but to encourage them.

Our Final Result: A Preview of the New Curriculum

The remainder of this document discusses in detail the design and implementation of the curriculum put in place at the end of 1991. However, it is appropriate to preview at least the core features of this revised curriculum. The curriculum that resulted from our design process hinges on a few key ideas:

- Engineering courses begin in the freshman year, concurrent with mathematics and science.
- The core of required "essential" engineering classes is extremely small.
- Area requirements across a spectrum of electrical and computer engineering topical areas replace most specific course requirements.
- Breadth, depth and coverage are mandated across this spectrum of technical areas, but individual courses are not prescribed; students flexibly choose from among available topical areas.
- Nearly a full year of the curriculum is completely unconstrained.
- At completion, the curriculum offers a single, unified bachelor's degree in electrical and computer engineering.

The end result of our exercise is a curriculum that has been recently reviewed by ABET for accreditation under the innovative curriculum provisions of Section II.A.7 of the ABET Engineering criteria. Comments made by the visiting team were favorable. Informal feedback from our ABET visitors indicates that they did not find any "deficiencies" (ABET's language for curriculum problems) in the ECE curriculum we put forth for accreditation. A written report from ABET will most likely reach us in the late summer of 1995.

Initial analysis of the three groups of freshmen entering ECE in 1991, 1992 and 1993 (about 150 students in each group) indicates that the students are enthusiastic about starting engineering classes in the freshman year and that these are helpful to the student when selecting their major. There is also evidence to show that the flexibility in the choice of electives has not resulted in a mass exodus to “easier” courses. In general, students continue to elect challenging courses to suit their interests.

This Document

In this document we share some of the details of the design process for this new curriculum, and an analysis (ongoing) of its implementation and impact. (See [1] for a more detailed, contemporaneous account of this process.) Chapter 2 offers a taxonomy of the basic problems faced by any electrical or computer engineering department as it struggles to keep pace with the technology, students and society, and then elaborates the design principles for the new Carnegie Mellon ECE curriculum that we evolved in response to these problems. Chapter 3 describes the architecture of the new curriculum, some of its novel characteristics and our experiences with its evolution from 1991 to 1995. Chapter 4 describes the key new classes developed for the new curriculum. Chapter 5 describes the detailed requirements for the new unified bachelor’s degree in ECE in the new curriculum, and then describes several possible paths through the program that are matched to different student’s skills and aspirations. Chapter 6 describes recent efforts to analyze its impact on students. Chapter 7 describes the recently introduced five-year integrated B.S.E.C.E. / M.S.E.C.E. degree we constructed on top of this new curriculum. Finally, Chapter 8 offers some concluding remarks.

CHAPTER 2

Curriculum Design: Problems and Solutions

We begin by enumerating the problems that we believe must be confronted in any curriculum re-engineering process, and the general solutions we evolved to attack these problems.

Problems We Addressed

A central tenet of any engineering education is that no elegant solution is likely to be found for a problem that lacks a crisp definition. Unfortunately, curricula are complex, often unwieldy creations subject to conflicting demands from the university, from faculty, from students and their parents, and from the industries that employ graduates. Nevertheless, over the course of its deliberations, our committee kept returning to several specific problems that crystallized as the basic issues to address. We summarize these here.

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. Any curriculum redesign 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 electrical engineering 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 ball park or has gone badly awry; *etc.*
- A related point: students have virtually no hardware tinkering skills. Previous generations of electrical engineers 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.

Any solution here must reconsider how and where in the curriculum to teach these fundamentals, and to what level of detail.

Student Perspective on EE, CE and Subdisciplines is Lacking

By the time they are seniors, faculty usually expect students to make intelligent choices when they have the opportunity to choose an engineering elective course. Students are expected to ask their faculty advisers for guidance here, and to listen to whatever advice is offered. Our experience as educators suggests that it is already questionable whether this works for seniors, who have a fairly extensive technical background. However, it is clear that students taking their very first course in a core ECE area like solid state devices or computer architecture are usually not clear about how this area connects to the rest of ECE as a whole.

This was a particular problem in our 1991 curriculum. At this time, ECE offered two parallel curricula: the B.S.E.E. and B.S.C.E. tracks, one of which students had to choose sometime during the sophomore year. The problem was how to educate students to make an informed choice. Certainly, some students arrived absolutely decided on one track or the other. But many relied on our introductory courses to paint a sufficiently broad picture of the discipline for them to make a choice. Unfortunately, these introductory courses concentrated almost entirely on packing in as much engineering material as possible. As faculty, we were often surprised when, after a few weeks in class, in the middle of some intricate technical discussion, a brave sophomore would 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?

The emphasis on maximizing technical content in those few hours per week left little time to address all these questions satisfactorily. And as the breadth of the discipline continues to expand, we must confront this problem 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 National Science Foundation article on the teaching of college calculus relates this story [8]:

> 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.”

Similar stories were easy to come by in our own department. For example, in [9], one of own faculty observed:

> [In several ECE courses] 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 Carnegie Mellon itself] that I make even difficult and abstract concepts seem clear.
>
> Yet, when I look at the reality of their understanding, as gauged 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.

Students’ varying technical preparation, the increasing diversity of their backgrounds, the divergence of student and faculty expectations and the widespread practice of packing ever more material into the same number of classes all compound the problem. We argue that curriculum designers must now address this problem directly. 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 connectedness among abstract ideas, concrete applications, their classes and their careers.

Breadth in All Relevant Topical Areas is Impossible

A foundation of many “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. We argue that this is no longer a viable assumption–especially for an ECE department whose faculty engage in a broad program of research ranging from basic physics to advanced computer science. Distancing a curriculum from this notion is difficult, since it tramples on nearly every faculty member’s most cherished subjects. Any attempt to reach consensus on a minimum set of advanced topics to mandate in a curriculum rapidly yields a huge and unwieldy set of essential classes.

One approach that Carnegie Mellon ECE had already tried in 1991 was to partition the undergraduates into separate degree tracks leading to different bachelor of science degrees. As Carnegie Mellon’s Electrical Engineering Department evolved into an electrical and computer engineering department, its degree offerings evolved into parallel B.S.E.E. and B.S.C.E. tracks. 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 countered that if students did not take the full complement of required electrical engineering classes, they should not be graduated as “electrical engineers.” So, a separate computer engineering degree was an agreeable solution.

In hindsight, this debate nicely crystallizes a key problem for curricula as they try to evolve: what is *essential* to earn a degree with the words “electrical engineer” (or, in our case “computer engineer”) in the title? Slicing off portions of the curriculum to award them separate degrees whenever they attain some sort of “critical mass” is not a viable long-term strategy: new technologies and ideas become candidates for core topics in the curriculum faster than old topics expire. Any serious attempt at curriculum redesign must address the necessarily contentious issue of which topics are truly essential.

Interdisciplinary Studies are Difficult

Many engineering curricula are based on a large number of required engineering classes and restricted technical electives. This was certainly true of our early 1991 ECE curriculum. The problem was how to deal with a student who wished to trade some technical depth for breadth. In 1991 one of our B.S.E.E. students could take some computer engineering courses, just as a B.S.C.E. student could take some electrical engineering courses. But we had no mechanism to give a degree to someone who chose to be broad, who chose to take, say, 50 percent of the required electrical engineering core courses and 50 percent of the required computer engineering courses. Some of our best students solved this problem by double majoring in both electrical engineering and computer engineering. But this required completing 100 percent of both curricula, using elective slots in one track to take required core courses from the other track. This challenging but rigid program ended up being a four-year degree with essentially no elective classes.

More generally, we argue that there is a need for mechanisms to allow for students to trade engineering depth for breadth–either breadth within a single department (*e.g.*, electrical engineering *and* computer engineering), or breadth among other disciplines (*e.g.*, electrical engineering *and* mechanical engineering).

Demographics Have Changed

In 1991 it was already clear that, on the entering side of the curriculum, our students were less homogeneous than in the past. Today, any department serious about attracting and retaining talented but under-represented minorities to engineering, must expect further diversity in their backgrounds. The problem of students not having the basic skills and motivations we would prefer may yet worsen. Thus, we argue that it is appropriate simply 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 a few of our best and brightest do not choose an engineering degree with the intent to become practicing engineers, 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 committed technologist, the mainstream engineer who may be in management in less than a decade and the interdisciplinary student using ECE as a launching point for a career in another professional discipline. In our own curriculum redesign, we concluded that 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

As mentioned in Chapter 1, many curricula evolve by accretion, and the resulting web of constraints can render even modest changes difficult. Hence, we suggest that another problem to address directly is *planned growth:* how to structure the curriculum to add flexibility to its basic organization so that necessary incremental changes are more easily effected.

Solutions We Developed

Before describing the exact organization of the new ECE curriculum that we created in response to the problems raised in the previous section, we summarize our attempt at general solutions to these problems. The ideas described can be regarded as the "design principles" for our revised curriculum.

Teach Engineering Early, Concurrent with Fundamentals

In 1991 the freshman year was common to all departments in Carnegie Mellon's college of engineering, and emphasized mathematics, science and humanities. A few "freshman elective" slots were available, as well as a few so-called "engineering science" slots (comprising elementary statics, dynamics, thermodynamics, materials science and so forth; see again FIGURE 1), but the choices comprised an *ad hoc* selection of peripheral engineering courses largely unrelated to the core curriculum of any engineering department. Students were largely disappointed by these courses.

In the new curriculum, every department in Carnegie Mellon's college of engineering offers a freshman engineering course that introduces students to the ideas, problems, modes of thought, tools and techniques of its discipline. All students take at least two such courses in their freshman year, concurrent with their mathematics, science and humanities classes. ECE offers a single course, called *Introduction to Electrical and Computer Engineering*, that strives specifically to provide an integrated view of connectedness of electrical engineering and computer engineering problems.

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 mostly lack. The goal is to generate real enthusiasm–the "Aha!" that accompanies insight as students grasp that they can, for example, model interesting physical phenomena with a little mathematics, science and judgment–and let them get their hands dirty on real problems. This experience demonstrates to new students the practical need for more preparation in subsequent mathematics and science classes and also provides students with the elementary hands-on laboratory skills that they often lack. At the college level, it allows undecided students to sample various engineering departments to be sure they are choosing the right one.

Base Curriculum Requirements More on Areas, Less on Specific Courses

The 1991 electrical engineering and computer engineering curricula were each based on a rigid core of required classes. In the new ECE curriculum, we first drastically reduced this required core, from about a dozen courses down to a select few. Next, we replaced the bulk of the remaining specific course requirements with *area* requirements: ECE was “partitioned” into a spectrum of topical areas, and all upper-level courses assigned to one of these areas. Students are required to demonstrate breadth, depth and coverage across some chosen subset of these areas. However, we no longer require students to take one or two courses in *every* core electrical or computer engineering area. Instead, we 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 electrical and computer engineer–and a capstone design requirement complete the basic curriculum.

Increase Flexibility, Elective Courses

The 1991 curriculum featured a hodgepodge of curriculum-specific elective slots (the B.S.E.E. and B.S.C.E. manifested different constraints on allowable electives) restricted in a variety of *ad hoc* ways. The actual number of completely unconstrained elective slots was amazingly small: one course slot.

The new curriculum substantially increased the number of unconstrained elective slots: we allowed slightly less than one full year of free electives. Aside from the fact that these courses must be ones for which students receive credit and a grade, they were not further constrained. The intent was not to reduce the rigor, complexity or depth of understanding associated with the ECE degree. Rather, the intent was to create new opportunities for more broadly based curricula that integrate ECE courses or courses from other disciplines in innovative ways. We saw no reason to prevent an ECE student with an interest in, for example, integrated silicon sensors, from taking a half dozen courses chosen sensibly from mechanical engineering, physiology, biology, chemistry, mathematics or physics. Nor did 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 only stifles the creation of innovative programs of study and innovative engineers themselves.

Manage the Workload

In the 1991 curriculum, students typically took five courses per semester. Usually, these comprised four technical classes and one humanities class. Especially in the last two years of their studies, engineering classes tended to inflate in content past their specified units as our faculty members struggled 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) was at best an optimistic lower bound on the time students needed to invest to survive. Independent of the merits of demanding course schedules, our students were 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 attacked this by reducing the number of courses from five per semester to four. ECE courses remained challenging and work-intensive, but the switch to one fewer class per semester made it possible for ordinary students to concentrate fully on their studies and master their material. We also tightened the requirements for "overloading," *i.e.*, taking a course load beyond a reasonable number of units, in our case roughly 4.5 courses per semester. In the 1991 curriculum, any student with a sufficiently high aggregate grade point average could overload. The problem this occasionally created was students who, having achieved good grade point averages on nonengineering courses in their early years, would later elect an insupportable engineering course load and perform weakly in each course. In the new curriculum, an overload requires a high grade point average for the courses in the preceding semester; now, a student must demonstrate continuously the ability to excel while taking extra courses, or those courses cannot be elected. Again, the strategy here is to encourage mastery, rather than mere survival, in a core set of carefully selected courses.

In addition, we rendered the workload a bit more uniform across all ECE courses by reallocating topics more carefully across course sequences. (In the 1991 curriculum, topics tended to creep downwards through a course sequence, to make room for more topics at the high-end of the sequence.) Finally, we made all ECE courses the same number of units, in some cases adding laboratories to relatively abstract and mathematical courses like electromagnetics and signals and systems to balance out the per-course workload.

Offer One B.S. Degree, Not Two

In 1991 we offered two degrees: the B.S.E.E. and B.S.C.E. In the new curriculum, we offer only one: the Bachelor of Science in Electrical and Computer Engineering, or B.S.E.C.E. There is an appealing symmetry here. We originally offered a single B.S.E.E. degree when we were the Department of Electrical Engineering. As the computer engineering discipline gained stature, we offered a B.S.E.E. "with a computer engineering option," which eventually split off to become the separately accredited B.S.C.E. degree. Now, we have merged all our degrees back into a single integrated B.S.E.C.E. This explicitly recognizes evolutionary trends in the discipline and in industry to emphasize the commonality across electrical and computer engineering, and not the differences.

Structure Curriculum to Accommodate Change

In 1991 the rigid set of interlocking course requirements made even small changes difficult. 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 freed it to adapt more easily to change. It is 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.

An interesting (and unexpected) example of this was the creation in 1994 of an integrated five-year bachelor's/master's degree built seamlessly on top of our new undergraduate curriculum. We shall return to this five-year program in Chapter 7.

CHAPTER 3

The New ECE Curriculum: From 1991 to 1995

We can now describe more fully the redesigned ECE curriculum. We begin by surveying the basic components of the new curriculum as it was proposed in 1991, their organization and relationships. We then describe some implementation realities that modified this ideal form of the curriculum, as it has evolved into its present form in 1995. We close with a discussion of our progress toward accreditation.

Proposed New Curriculum: 1991

FIGURE 2 illustrates the basic components of the curriculum by topic; FIGURE 3 adds detail to show how courses are allocated among these areas. The architecture of the curriculum is essentially simple and comprises the following:

- A typical humanities, social sciences and fine arts 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. (at least 2 courses)

FIGURE 2. **New ECE Curriculum: Basic Organization in 1991**

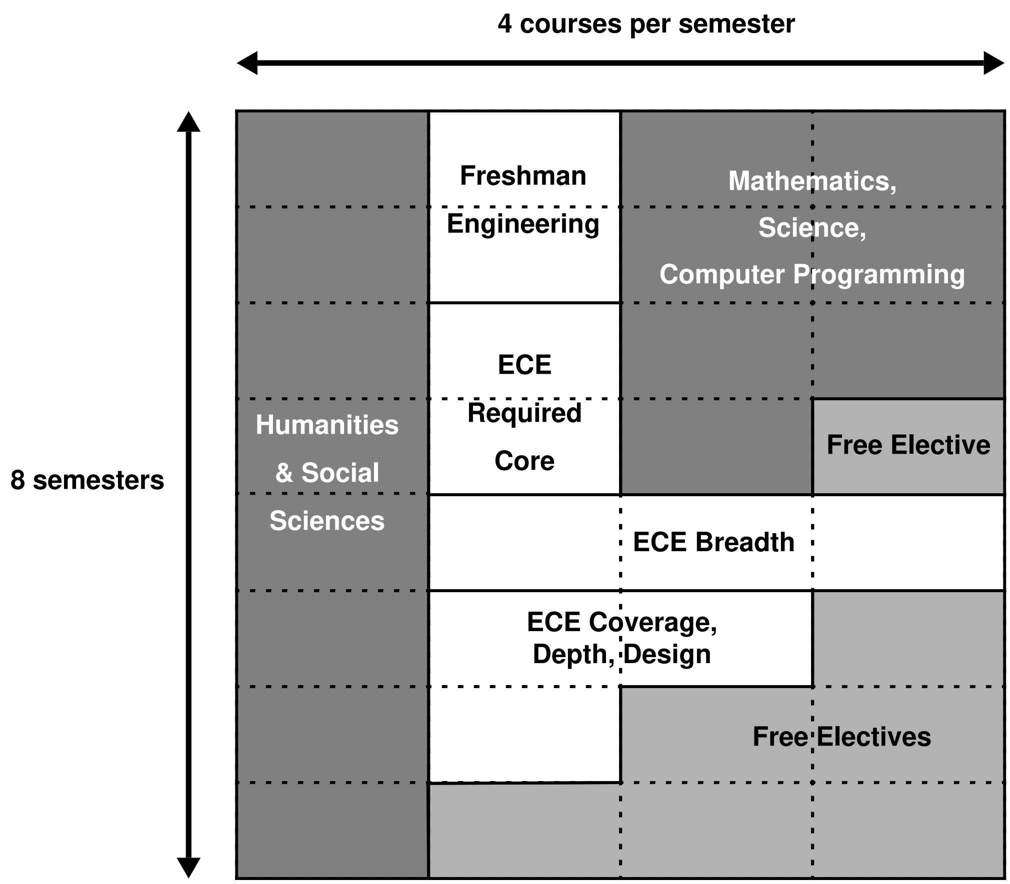


FIGURE 3. New ECE Curriculum: Allocation of Courses to Areas in 1991

FRESHMAN	**FALL**	Humanities & Social Sciences	Introduction to ECE	Computer Science Intro Programming	Calculus I
	SPRING	Humanities & Social Sciences	Introduction to Engineering	Physics I	Calculus II
SOPHOMORE	**FALL**	Humanities & Social Sciences	Fundamentals of Electrical Eng	Physics II	Linear Algebra
	SPRING	Humanities & Social Sciences	Fundamentals of Computer Eng	Modern Math	Free Elective
JUNIOR	**FALL**	Humanities & Social Sciences	ECE Breadth 1	ECE Breadth 2	ECE Breadth 3
	SPRING	Humanities & Social Sciences	ECE Coverage (Ex: Depth)	ECE Coverage	Free Elective
SENIOR	**FALL**	Humanities & Social Sciences	ECE Coverage (Ex: Design)	Free Elective	Free Elective
	SPRING	Humanities & Social Sciences	Free Elective	Free Elective	Free Elective

- ECE core requirements, a set of two fundamentals classes (in addition to a required ECE freshman engineering course) required of all ECE students. 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-solving. (3 courses)
- ECE coverage requirements, to ensure enough exposure to ECE courses to earn a degree called the 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)

Our previous curriculum required 388 units in all for a bachelor's degree. The new curriculum reduces this slightly–by roughly 2.5 classes–to 360 units to satisfy our goal of four classes per semester in a typical semester. At Carnegie Mellon, a typical course is 9 to 12 units, where each unit represents one hour of expected work each week. A 12-unit class is nominally intended to require 12 hours of effort per week.

Freshman Engineering Requirements

Two requirements for obtaining a B.S.E.C.E. 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 the freshman year, and one of them will be our ECE course. However, the student unsure of a major may end up taking three of these courses, without penalty, and defer deciding on ECE until the fall of the 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.[2] This poses no additional constraints on the new ECE curriculum.

For completeness, the list of all current (1995) CIT freshman introductory engineering courses follows. Each course is permitted to have one science and one mathematics corequisite:

- *Introduction to Chemical Engineering* (corequisites *Modern Chemistry I* and *Calculus I*)
- *Innovation and Design in Civil Engineering* (corequisites *Physics for Engineering Students 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 for Engineering Students I* and *Calculus I*)
- *Fundamentals of Mechanical Engineering* (corequisites *Physics for Engineering Students I* and *Calculus I*)
- *Materials in Engineering* (corequisites *Physics for Engineering Students I* and *Calculus I*)

Flexibility = Core + Breadth + Depth + Design + Electives

The key attribute of this 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, coverage 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 that it accommodates new students with different backgrounds and skills:

- The best students can begin engineering classes earlier and pursue several ECE areas in great depth.

2. Eight semesters total minus three semesters of introductory engineering courses leaves five semesters for department-specific specialization.

- 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 students who want to take, say, half of their 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 any gaps, and defer some engineering until later; these students in particular may benefit by choosing to trade off some depth for breadth.
- Interdisciplinary students now have the time to dive into another discipline, its background topics, its core classes and even a few advanced courses. This is 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. Interestingly, it seems that the presence of *any* free electives in an engineering curriculum is rare, let alone nearly a full year as the new 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. However, blame has also been attributed to the engineering accreditation process [10]:

> 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.

Whatever the reason, the pressures for limiting electives to a negligible few have been numerous and well intentioned. Nevertheless, if one of our goals is to increase the diversity among the population of electrical and computer engineers that we educate–both the diversity of their background upon entry and their portfolio of skills upon graduation–a substantial number of free elective courses is a sensible solution.

In the following sections, we discuss each of the component areas of this curriculum. We shall return to the topic of accreditation at the end of this chapter.

ECE Core

The new curriculum introduces three critical new courses. By the end of the sophomore year, the average student will complete the small ECE core, comprising these courses:

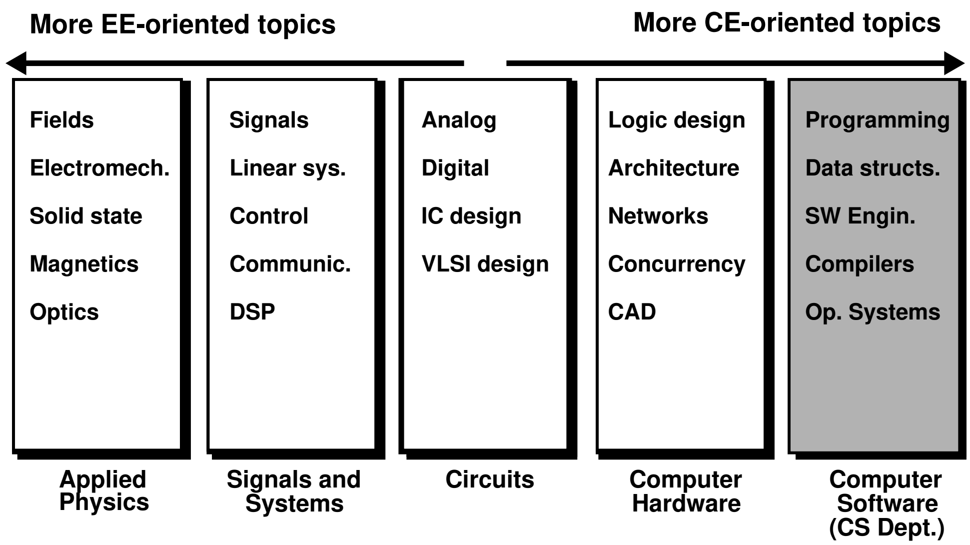
- *Introduction to Electrical and Computer Engineering*, which has introductory calculus and computer programming classes as corequisites, introduces basic engineering ideas related to electricity and computers. It is typically taken during the first year.
- *Fundamentals of Electrical Engineering*, which has a linear algebra class as a corequisite, is a course in linear circuits. It is typically taken in the second year.
- *Fundamentals of Computer Engineering*, which has a discrete math class as a corequisite, is a course in digital design, microprocessors and elementary computer organization. It is typically taken in the second year.

These courses are a critical part of the proposed curriculum; we shall return to them in greater detail in Chapter 4.

ECE Breadth

Before describing these requirements, it is best to illustrate what “breadth” means in ECE by referring to the illustration in FIGURE 4. Rather than simply partitioning the department into electrical engineering and computer engineering halves, we chose instead to restructure it as a spectrum of five areas. Going to the left in the figure takes us more toward traditional electrical engineering topics; going to the right takes us toward traditional computer engineering 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 consistent notion of breadth, without having to resort to requiring specific courses.

FIGURE 4. **ECE Breadth Areas in New Curriculum**



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:

- **Applied Physics** includes courses in electromagnetics, solid state devices, magnetics, data storage, optics, *etc.*[3]
- **Signals and Systems** includes courses in signals and systems fundamentals, as well as control, communication, signal processing, robotics, *etc.*
- **Circuits** includes courses in both analog and digital electronics, as well as IC and VLSI design, *etc.*
- **Computer Hardware** includes courses in digital design and verification, computer architecture, processor design, networks, concurrency, CAD, *etc.*
- **Computer Software** includes programming, data structures, formal methods, software engineering, compilers, operating systems, *etc.* (These courses are all

3. In the original 1991 curriculum, this area also included some courses from the Physics Department. As of this writing in early 1995, there are plans to remove these courses.

offered by the School of Computer Science, with whom we maintain a reciprocal relationship: computer science students take several hardware-oriented ECE classes, and ECE students can elect software-oriented computer science classes.)

Courses in these areas are referred to as *breadth* courses. Revisiting FIGURE 4, we see this spectrum of five areas, with several representative types of courses listed in each area. Each area offers at least one, and possibly several introductory courses that can be taken starting with the three ECE fundamentals classes as prerequisites, along with perhaps some additional mathematics or science courses. Other courses in each area are more advanced, and require some earlier area-specific ECE breadth courses as prerequisites.

ECE Coverage and Depth

At least three more courses must also be taken from the areas defined in the ECE breadth requirement; we refer to this as a “coverage” requirement. The idea here is to ensure that students see enough engineering courses to be called “engineers” when they graduate. Two additional requirements, the depth and design requirements, can be satisfied by dedicating two coverage courses to depth and design, respectively.

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 most fundamental topics in our curriculum are covered in two-semester sequences, practically speaking, the depth requirement will cause students to complete at least one such sequence.

ECE Design

Carnegie Mellon has a long tradition of offering aggressive and challenging design courses. The design requirement is satisfied when a student completes one course from an approved list of design courses across ECE. These courses tend to be on the high end of our curriculum and are generally quite popular. The intent is for students to take aim at these courses sometime during their junior year, carefully picking up the required mathematics, science and ECE prerequisites.

Workload, Overload Policy, QPA Policy

The curriculum was designed around the assumption that students should 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 was that the one H&SS class was 9 units, and the remaining three classes each 12 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. (See page 34 for more discussion of this problem.) Our intent in establishing a workload norm and overload policy was 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 old curriculum, only students with a high quality point average (QPA) were allowed to overload. In the new curriculum, we modified this somewhat. First, an overload was 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 now only permit students to overload if they have achieved a QPA of 3.5 out of 4.0 in the *previous* semester. We wanted 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 wanted to encourage students to focus more deeply on fewer courses, and so we consciously erected some barriers to prevent ordinary students from getting in over their heads. We had ample experience with very bright students who achieved a high overall QPA early in their studies, then started overloading every semester, only to watch their QPA crash as they refused to acknowledge that they were overwhelmed. The new overload policy was designed to 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, coverage and design requirements.

Curriculum Evolution: 1991 to 1995

Several modifications have occurred to the “original” organization, which was illustrated in FIGURE 2 and FIGURE 3. Nevertheless, the overall organization of the curriculum as it is now implemented has remained surprisingly faithful to our original vision. In this section we discuss the critical changes and their motivations.

Allocation of Units

Most of our changes are simply the result of local idiosyncracies. Most notable among these is that not all courses at Carnegie Mellon have an identical number of credit units (notably mathematics, science and humanities). For example, almost all ECE technical courses carry 12 units of credit; many mathematics and science classes carry between 9 and 12 units. Most nontechnical humanities classes carry 9 units. In addition, a few so-called skills classes (*e.g.*, a university-required computing skills class taken by all freshmen) carry only 3 units. Hence, many non-ECE courses do not completely "fill" one of the planned slots in FIGURE 2, making it difficult in a few places to obey the four-courses-per-semester guideline. One result of this was the inclusion of some extra mathematics requirements to partially fill this units gap. In addition, examination of ABET accreditation guidelines also contributed to these minor changes, resulting in a decision to increase somewhat the overall fraction of the program dedicated to technical subjects. As a result, roughly two and one half of the original free electives were transformed into:

- **Math/science electives:** Two courses must be selected in mathematics or the sciences–biology, chemistry, physics. These two courses occupy slots equivalent to 1.5 ECE engineering courses.
- **Probability and statistics:** We added this as a required mathematics course.

In addition, we added constraints to another free elective, requiring it to be technical, resulting in:

- **Engineering elective:** One course must be selected from across all engineering departments. This occupies one ECE slot.

As a practical matter, the resulting impact on flexibility was offset by changes in Carnegie Mellon's humanities requirements (described on page 34). Two of the eight humanities and social sciences courses (equivalent to 1.5 ECE engineering slots) can now be chosen freely from among nontechnical topics across Carnegie Mellon. Hence, constraining a few of our own free electives to be technical is somewhat offset by these nontechnical free electives; this brings the curriculum back fairly close to its original form.

Updating the original form of the curriculum illustrated in FIGURE 2 and FIGURE 3, FIGURE 5 and FIGURE 6 illustrate more accurately the current, 1995 form of the ECE curriculum.

FIGURE 5. New ECE Curriculum: Basic Organization in 1995

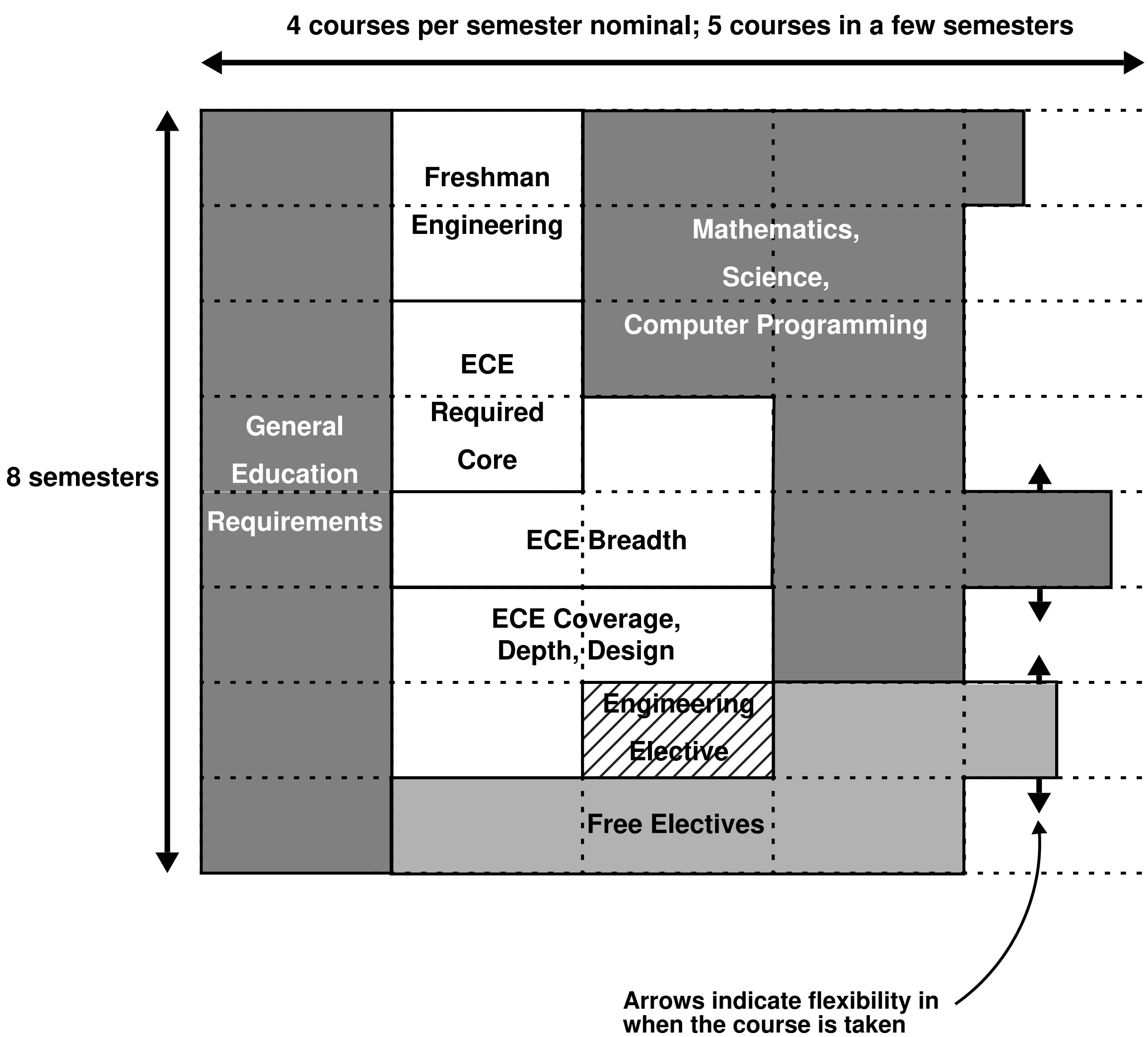


FIGURE 6. **New ECE Curriculum: Allocation of Courses to Areas in 1995**

FRESHMAN	FALL	General Education	Introduction to ECE	Computer Science Intro Programming	Calculus I	Computing Skills
	SPRING	General Education	Introduction to Eng	Physics I	Calculus II	
SOPHOMORE	FALL	General Education	Fundamentals of Electrical Eng	Physics II	Linear Algebra	
	SPRING	General Education	Fundamentals of Computer Eng	ECE Breadth 1	Discrete Math	
JUNIOR	FALL	General Education	ECE Breadth 2	ECE Breadth 3	Math/Science Elective	Probability / Statistics
	SPRING	General Education	ECE Coverage (Ex: Depth)	ECE Coverage	Math/Science Elective	
SENIOR	FALL	General Education	ECE Coverage (Ex: Design)	Engineering Elective	Free Elective	Free Elective
	SPRING	General Education	Free Elective	Free Elective	Free Elective	

General Education Requirements

In 1991 Carnegie Mellon required all undergraduates to complete a sequence of humanities and social sciences–called H&SS–courses. Those requirements, totaling eight courses, included:

- One course chosen from a list of designated *writing and literature* courses.
- One course chosen from a list of designated *history* courses.
- One course chosen from a list of designated *social science* courses.
- One course chosen from a list of designated *psychology and philosophy* courses.
- Three more advanced courses, constituting a so-called *depth* sequence, chosen from among nontechnical courses in humanities, social sciences, foreign language or fine arts.
- One freely chosen nontechnical course from humanities, social sciences, foreign language or fine arts.

In 1994, the H&SS requirements were reorganized university-wide and renamed the *General Education* requirements. These again mandate eight classes, but now comprise:

- One course chosen from a list of designated *writing and expression* courses.
- One course chosen from a list of designated *humanistic studies* courses.
- One course chosen from a list of designated *cognition and institutions* courses.
- Three more advanced courses, constituting again a *depth* sequence, chosen from among nontechnical courses in humanities, social sciences, foreign language or fine arts.
- Two freely chosen nontechnical courses from humanities, social sciences, foreign language or fine arts.

We note that there is slightly more elective freedom in the General Education requirements now–two versus one free course–which somewhat offsets our own changes to the free ECE elective restricting them to be technical.

Timing ECE Course Offerings

Constraints on the size of the department mean that we cannot offer every course every semester. Nevertheless, we 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.

In addition, some noncore classes have begun to appear every semester, just to satisfy the demand from the students. The introductory breadth course in the signals and systems area is now offered nearly every semester.

We have also changed the way that we time the offerings of the core sequences in each of ECE's breadth areas. In the past, we synchronized each multiple-course sequence so that the starting courses, *e.g.*, *Electromagnetics I*, *Solid State 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 have staggered the offerings to ensure that every semester offers a good selection of beginning breadth courses for students.

Note also that FIGURE 5 shows the first breadth course in the spring of the sophomore year rather than in the fall of the junior year. Many students find that this timing gives them earlier access to more advanced courses in their areas of interest. The vertical arrows in FIGURE 5 indicate flexibility in when a course may be taken.

Design Across the Curriculum

Carnegie Mellon in general, and ECE in particular, have a long tradition of infusing aggressive design problems into the classroom. Interestingly, the new ECE curriculum provided yet more opportunities to strengthen our commitment to design.

One less obvious opportunity arose from the need to make all ECE technical courses the same number of units (to occupy one of the so-called "ECE slots" on FIGURE 2 or FIGURE 5). In the prior curriculum, courses with some laboratory or design exposure were typically 12 units, whereas more mathematically oriented classes *(e.g.*, signals, solid state, *etc.*) were typically 9 units, and lacked a lab. The new curriculum allowed us to rethink these decisions, with the result that most breadth-area classes now include some laboratory experience. Sometimes this is a new hands-on lab component, constructed from scratch, as with our revised solid state courses. Sometimes this is centered around engineering design, simulation or verification software, such as the use of electromagnetic field solvers and visualization software in the revised electromagnetics courses. Sometimes both hardware and software are used to advantage in new lab experiences, as with the use of signal processing software, combined with PC-based signal acquisition and processing hardware, in our revised introductory signals class.

An example helps make this concrete. In our current *Engineering Electromagnetics I* class, students create their own Matlab^{tm} code and use existing software (some commercial finite element software and some educational software created by the instructor) to work on assignments that are relatively open-ended, with an orientation toward design as well as analysis. In the fall of 1994 students in this course created their own software to map out the magnetic field structure in one wing of the building in which the Computer Science, Mathematics and Physics departments are housed. This wing of the building is situated directly above the 24kV - 4kV main step-down transformers that supply power to the Carnegie Mellon campus. The magnetic field levels are high enough to cause significant interference with workstation monitors in this area, and some residents have also expressed concern about possible health effects. Starting from basic principles they had learned in the course and using tools they had created in one of the prior projects, students mapped out the field structure and compared their calculated results with measured results. This project worked well: it was interesting, challenging, relevant to a current engineering design problem and reinforced basic ideas from the course. It also demonstrates clearly how even abstract, mathematically oriented engineering topics can benefit from creative laboratory assignments.

Our advanced classes also continue to offer a spectrum of design experiences. In advanced design classes, students are asked to create a device, a component, a process or a system where the goals or performance criteria are specified, and where a multitude of alternative solutions are possible, so trade-offs must be examined and decisions made.

An industrial design project may involve only a single engineer in a single location, or hundreds of people distributed over the globe. Neither the problem nor the test criteria may be well defined in advance. In fact, the typical industrial project involves three simultaneous activities:

- Design the *organization* to be used in the project (the people, their responsibilities and their resources).

- Design the *problem* to be solved (specify the goals, define the "designable" parts of the solution, design the tests to be used in distinguishing good solutions from bad ones and select starting points).
- Design a *solution* to the problem.

In the typical industrial project, both the organization (who is responsible for which parts of the solution) and the problem itself changes over the course of the project. Industrial projects also tend to be open ended–unanticipated behaviors of a product can come back to haunt its designers years after the product has stopped being made. As a result, risk and liability also play an important role. Thus, topics visited in our design-oriented classes can include:

- Problem specification and requirements
- Verification and test
- Alternatives, choices, decisions and trade-offs
- "Language" of design
- Orderly subdivision
- Analysis and synthesis
- Uncertainty and risk
- Use of design tools

Almost every course in the ECE curriculum involves some design. Some courses focus on design problems faced by individual designers, such as building a performance-constrained circuit or software module as part of a larger system. Other courses offer group design projects with more open-ended goals and realistic limits on resources, alternatives and time. Examples here range from design of complete superscalar processors to telecommunication systems to control systems. In ECE these are referred to as "capstone" design classes, since they require a rigorous sequence of prerequisite courses covering the necessary mathematics, theory and engineering practice. It is also worth noting that many ECE faculty are themselves designers–of systems, of circuits, of manufacturing processes, of software, *etc.*–and so students have access to world-class expertise in many of these courses.

Our analog IC design courses, for example, teach students design by having them perform several design projects throughout the course of the semester. Generally there will be two or three small design projects during the semester, and one larger design project at the end of the semester. Students normally carry out these design projects individually, but on occasion group projects of large scope are undertaken.

Both during and after completion of each design project the students meet one on one with the instructor to review their design. The instructor provides feedback on the design process as well as feedback on the design itself. Because of the nature of analog IC design, these projects typically involve circuit design, circuit simulation, IC layout, estimation of layout parasitics and handling of variation in the IC manufacturing process. Further, students are required to take into consideration reliable and economical testing of their circuit during the design process. On several occasions student's designs have been fabricated through the ARPA/NSF supported MOSIS service and the students have tested their designs in a follow-on course on testing of integrated circuits.

Another interesting example of design is the design of CAD tools themselves. For VLSI circuits, CAD tools are large, interlinked suites of communicating software that successively transform a design specification into a final, finished chip and the manufacturing process for that chip. In ECE, students have the opportunity to actually design examples of these software tools. Recent projects have included partitioning of large gate-level designs across multiple physical chips, wire routing for the newest generations of field-programmable gate arrays, wherein not only the function of each logic block but the pattern of interconnections are programmable by the user, and detailed simulation of electronic circuits.

Advising

An interesting new problem we faced is that a highly flexible curriculum is a more difficult curriculum to advise. Especially during the highly regimented sophomore and junior years of our previous ECE curriculum, too many of our students treated 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 could be made. This situation has changed 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 knowledgable faculty and advising staff. We have reorganized our advising so that all of our undergraduates obtain consistent and up-to-date information about classes and degree requirements from a central faculty adviser. Professional and area-specific information such as job and graduate school opportunities is obtained from individual faculty advisers. We are also implementing or considering several new mechanisms to improve the advising process generally. These include:

- **Curriculum documentation:** Although a succinct description of the ECE curriculum can be found in the "Carnegie Mellon Undergraduate Catalog," a more extensive description is contained in a new departmental booklet, "The ECE Curriculum Primer." It describes the structure of the curriculum, which courses can count for what, and general academic and departmental policies. It also gives several example curricula that illustrate typical course choices emphasizing several popular areas within the department. The "Primer" is distributed in both booklet and on-line form.
- **Curriculum planning software:** Once students have begun to understand the basic structure and requirements of the curriculum, they are encouraged to begin to lay out a four-year schedule plan. They may change and update the plan as they learn more and discover the things they enjoy most, but at any given time they are advised to have a workable four-year plan. To assist with this, the department is developing a software application called the CMU Planner. This is a UNIX-based application that is available on workstations in the department's computer lab as well as workstations available to the campus-wide Andrew network. The software allows students to view catalog descriptions, select courses and move them around like tiles on their four-year plan. The program will check to make sure the course prerequisites have been satisfied, and will ultimately check to make sure the degree requirements have been satisfied. The details of this program are described in documents made available to our students in both paper and on-line forms.
- **Template catalog:** As mentioned above, several common course plans, or templates, are given in the "Primer." In addition, a catalog of templates that students and faculty work out is maintained by the Undergraduate Office. Students may find that one of these templates matches their interests well, or they may want to start with a template and make some changes. Students are encouraged to submit new templates to be added to the catalog.
- **Plain language catalog:** This proposal under consideration is to compile descriptions for each ECE course that are designed to be intelligible to someone who has not yet taken the course. Whereas the official catalog descriptions are designed to be concise and precise, the Plain Language Catalog descriptions are intended to give a nonexpert the idea and flavor of the course.
- **Registration seminars:** Another proposal under consideration is to hold a seminar before registration each semester to give students an opportunity to address questions to the faculty about courses that will be offered during the next semester. The seminar would be structured as an informal panel discussion with representative faculty from each area of the department. From the question/answer discussions students should gain a better "feel" for what the courses will be like than they can obtain from simply reading the catalog description.

> The faculty will describe how the courses lead to other courses in the area and, where possible, how the course sequences relate to actual job functions and opportunities. An informal "peer advising" session would be held immediately following the panel discussion. This session would be sponsored by the Honor Society for Electrical and Computer Engineers, Eta Kappa Nu. Although every effort would be made to schedule the seminar at a convenient time, the seminar would also be videotaped for those who are not able to attend. Copies of the videotape could be checked out from the Undergraduate Office.

Our advising strategy is very much a "work in progress." We continue to believe that advising is critical and that our recent advising changes have been timely and helpful, but also that it remains difficult to do advising well.

Accreditation

Despite these changes to the curriculum as it evolved from 1991 to today, many elements of current curriculum remain at odds with existing accreditation criteria: required courses remain few (*e.g.*, there is still no chemistry class), relatively free electives remain numerous, the electrical engineering and computer engineering curricula remain integrated into a single B.S.E.C.E. degree.

During the design of the original curriculum, ABET itself was a national focus of heated debate about the role of accreditation in fostering or stifling curriculum innovation. Indeed, the final report of an ABET task force on this issue [13] wrote in 1991 that:

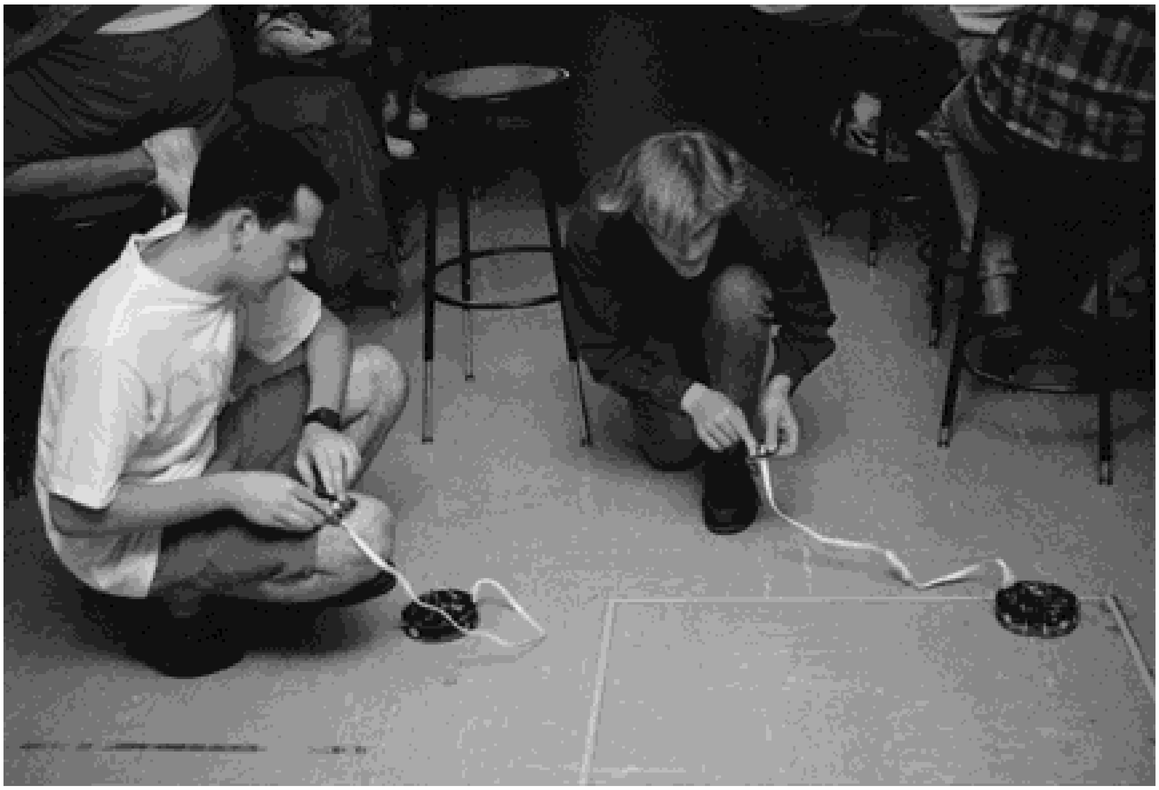
> ...there has been a growing divergence of opinion between the engineering academic community, as represented by the Engineering Deans Council, and ABET, on matters critical to engineering education. This comes at a time when engineering education is facing significant challenges. In the future, engineering 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.
>
> 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.

However, the task force also concluded in [13] 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.” After much discussion among our faculty, and with ABET itself, ECE at Carnegie Mellon chose to pursue accreditation as an innovative program under existing ABET guidelines for curriculum experiments.

In the fall of 1992 we presented our new curriculum to the IEEE Committee on Engineering Accreditation Activities (CEAA). This generated much discussion over the course of the year. The concept of a combined ECE degree was less an issue than whether certain courses should be required, such as one on probability and statistics. But the general reaction of this professional group was positive. In February of 1993 we presented the curriculum to the Engineering Accreditation Commission (EAC) of ABET. As a result, the EAC assigned Philip Lopresti of the EAC Executive Committee as the EAC liaison and adviser to Carnegie Mellon as we prepared for the accreditation process.

In early 1994 it was decided that, despite the small-scale modifications to the original 1991 curriculum proposal we actually implemented (see again FIGURE 5 and FIGURE 6 for the architecture of the curriculum as it is implemented *now*, in the 1994-1995 academic year), ECE would actually apply for accreditation of the “original” curriculum (see FIGURE 2 and FIGURE 3) under the ABET “innovative curriculum” clause that permits thoughtful experimental curricula that diverge from existing ABET standards to be considered on their merits. Our reasoning at the time was that, despite some small-scale changes to deal with local idiosyncracies such as the difference in units awarded in ECE classes versus non-ECE classes, the original, “pure” form of the curriculum was ECE’s unique statement of how a curriculum *ought* to be organized. In September of 1994 ECE was visited to start the accreditation process. Initial results of the visit were quite positive; comments from the review team were favorable. The committee was impressed with the enthusiasm of the students. No deficiencies were found, which should clear the way for final accreditation.



ECE freshmen testing their newly built robots.

CHAPTER 4

Key New Courses in the Current Curriculum

A key part of the current curriculum is the so-called “ECE core” of required classes. This core is extremely small–just three mandated classes–and is aimed at our freshmen and sophomores. The design and “tuning” of these classes has been a critical part of the perceived success of the new curriculum. We describe each of these courses, their structure and motivation, and offer some analysis of how well they are working.

ECE Core: Freshman Introduction to ECE

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 our old curriculum, our freshmen took preparatory courses such as physics and calculus, but waited until the end of the second year to take any real engineering courses. The result was often a significant reduction in motivation as students worked to understand a wealth of relatively abstract fundamental mathematics and science presented without any supporting ECE-specific engineering context.

FIGURE 7. **Syllabus: Freshman Intro to Electrical and Computer Engineering**

FIGURE continued

Our new freshman introductory course rectifies this situation. The course motivates and introduces basic concepts in electrical and computer engineering in an integrated manner, provides hands-on laboratory experience early, and strives 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 has calculus and computer programming courses as corequisites. It introduces fundamental theoretical ideas in both electrical engineering and computer engineering. 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. The basic syllabus appears in FIGURE 7.

Structure

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 electrical engineering and computer engineering 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.

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 noninverting 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

For example, 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 element–which of course turns out to be Ohm's law. 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 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, Kirchhoff'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. This approach has been referred to by some as *just-in-time learning* [14], 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."

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 the robot, such as the motor driver, the sensors, and the digital control and programming components. 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 connection between analog and digital ideas. In our current curriculum, entering students often have no clear vision of the relationships between the electrical engineering and computer engineering “ends” of the department. In the old curriculum, students were 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 developed an “us” versus “them” parochialism as they came to identify themselves as primarily electrical engineers or primarily computer engineers. The new curriculum strives to remedy this by exposing our students *immediately* to a more unified view of the different disciplines comprising ECE.

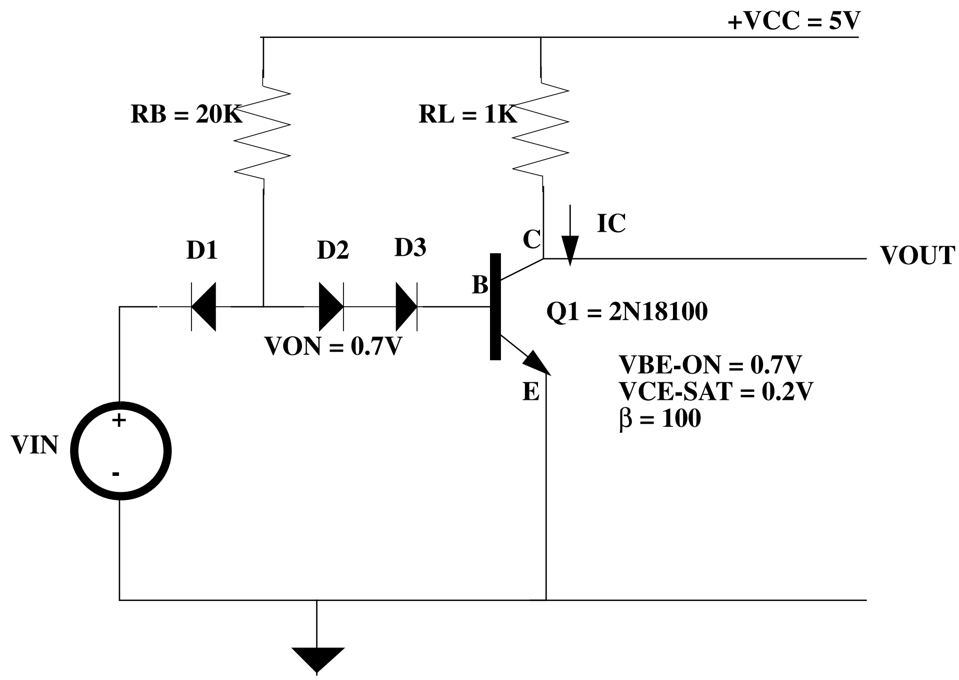
Finally, it is worth noting that, properly motivated, it is possible to treat nontrivial subjects in such a course. FIGURE 8 shows an example of a design problem from a recent final examination in this freshman-level course.

Analysis

Perhaps the most common observation we can share is that entering students are enthusiastic about starting real engineering classes in their first semester. They also simply appreciate the attention paid to them by their chosen departments in their first year, as opposed to the older curriculum in which ECE played no role until late in the second year. There is also evidence that students are using the freshman engineering courses offered by each department to help select their major, or at least, to decide which majors they do not wish to pursue. At least a few students have switched majors from their original choice after sampling the introductory freshman course offered by that department. An unexpected positive side effect of teaching engineering early is that some students have actually been able to land engineering jobs during the summer after their freshman year.

However, there were substantial adjustments necessary in faculty “teaching style” to accommodate very young students making the transition from high school courses with day-to-day, step-by-step guidance to college courses with more demands for individual initiative. One current solution is frequent small exams to test acquisition of the critical concepts. Another is extremely careful synchronization among the lecture material, the recitation problem-solving sessions and the weekly laboratory assignments.

FIGURE 8. **Example Final Exam Problem from Introduction to ECE**



This circuit consists of a general transistor inverter that is used to interface to TTL logic. Suppose the range of logical 1 is defined to be anything between 2.5V and 5V and logical 0 is defined to be between 0V and 1.2V. Anything in between 1.2V and 2.5V is an incorrect logic value.

- Draw VOUT as a function of VIN. Mark the beginning and end of each piece-wise linear region on the VIN axis. In each of these piecewise linear regions, mark the state of D1, D2, D3 and Q1. For example, write D1=off, D2=on, D3=on, Q1=forward active. Hint: It may help you to think of D2 and D3 as a single diode with VON = 1.4V.
- β can vary substantially during manufacturing. Determine the minimum value of β such that the logic gate still operates correctly. Hint: the limit on β will occur when VIN = 2.5V and VOUT = 1.2 V so you should solve for β to make this happen.

ECE Core: Sophomore Fundamentals of EE

This is the first real “circuits” course. It differs from traditional courses primarily in two ways. First, it exploits the fact that students have had some real exposure, both practical and theoretical, to linear circuit ideas and issues in the robot project from the freshman *Introduction to ECE* course. The freshman course is a prerequisite; a linear algebra course or the new ECE course *Mathematical Foundations of Electrical Engineering* is a corequisite. Second, the course uses a nontraditional focus (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. Students are assumed 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 explain how, a decade or so ago, this speed was determined more by transistor size, *i.e.*, how small a device could be made, how many could 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.

Structure

As expected, the course focuses on studying in detail many simple RLC circuits. However, we always motivate these studies 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: Kirchhoff’s current and voltage laws; superposition and convolution; series, parallel and ladder circuit analysis; Thevenin’s and Norton’s theorems; natural frequencies and $j\omega$; circuit partitioning; and nodal analysis. For additional perspective here, see [11].

The class has lectures, a laboratory session and a recitation session. Lectures motivate and develop theoretical material, laboratories provide more hands-on exposure to circuits to test how well the theory really works and recitations provide opportunities for problem solving and review. FIGURE 9 shows a syllabus of topics for this course.

Analysis

Aside from being a linear circuits course, the sophomore *Fundamentals of EE* course also introduces and reinforces some crucial mathematics in our curriculum. In our prior curriculum we required three related sets of mathematics courses:

- *Calculus I, Calculus II* to handle functions, basic derivatives, integrals, *etc*.
- *Differential Equations* to handle the functional form of the ordinary differential equations arising in circuits, electronics, signals, *etc*.
- *Linear Algebra* to familiarize students with matrix methods to handle the formulation of circuit solution techniques, control systems, *etc*.

In the 1991 version of the new curriculum, we dropped the differential equations class in favor of an attempt to teach this material in place, “just in time” [14] in the *Fundamentals of EE* class. This was motivated in part by the coverage of topics in the available differential equations class: the Mathematics Department chooses a broad range of functional forms to teach, while we would prefer deeper coverage of a few critical forms, for example, linear, constant coefficient differential equations arising frequently in physical systems. However, we also sincerely believed that circuits formed an excellent problem domain in which we could motivate differential equations to engineers. We kept the linear algebra course (despite similar concerns about coverage) but recast it as a corequisite to the Fundamentals of EE class.

As of early 1995, we appear to be succeeding in conveying basic circuits, but we are less successful with the mathematics. One problem–in hindsight somewhat obvious–is simply the difficulty in covering in a single course a wide range of mathematics in *addition* to an aggressive set of circuit topics. Even with our careful treatment of the fundamentals in this course, we still see random gaps in crucial mathematical preparation (*e.g.*, complex analysis, linear differential equations and vector calculus) coming back to cause problems as our students elect more advanced classes.

This problem is sufficiently pervasive that we have proposed an entirely new ECE class just to rectify it: *Mathematical Foundations of Electrical Engineering*. Starting in the 1995-1996 academic year, this new course can be substituted for the linear algebra class as a corequisite for *Fundamentals of Electrical Engineering*. We will return to this proposed class (see page 58) later in this chapter.

FIGURE 9. Syllabus: Sophomore Fundamentals of Electrical Engineering

A brief history and forecast of microelectronics

- Sizes, speeds and complexities of ICs from 1970 - present, projected to the year 2000.

Fundamental electrical concepts

- Review of voltage, current, Kirchhoff's laws and element relations for ideal independent voltage sources and resistors and capacitors.

Single time constant RC circuits and constraints on gate delay

- DC steady state and transient solutions of circuits with DC sources, switches and a single capacitance: exponential waveforms, their asymptotes and time constants.
- Gate delay and limitations on CMOS digital circuit switching speed derived in terms of switch, resistor, capacitor models of constituent transistors of CMOS inverters.

RC ladder circuits and on-chip IC interconnect characteristics

- Distributed and lumped models of uniform RC line for IC interconnect.
- Detailed analysis of transient response in terms of one and two lump ladder circuit models of such lines. Solutions of second-order equations in coupled first-order matrix form.

Switching circuit performance limitations caused by capacitance

- Fanout effects on switching speed with and without consideration of interconnect.
- Switching time speed-up strategies.
- Line-to-line capacitance coupling and its effects on switching speed and crosstalk.

FIGURE continued

Package inductance and RLC circuit analysis

- Inductance and its differential equation.
- Second order equations that arise from RLC circuits, their natural frequencies and the general forms of their solutions. Overdamped, critically damped, underdamped and undamped cases.

Transmission lines

- Telegrapher's equations; wave propagation down distributed LC lines; signal delay.

Basic AC circuit analysis

- Sinusoidal steady state; superposition of independent frequency components; peak, average, RMS value calculations.

Advanced AC circuit analysis

- Complex representation; phasors; complex impedance; the frequency domain.

Nodal analysis

- Matrix formulation of arbitrary circuit topologies; solution strategies; introduction to simulation methods.

ECE Core: Sophomore Fundamentals of CE

This course builds upon the rudimentary digital design and computer engineering concepts presented in the freshman *Introduction to ECE* class. (The new course was actually patterned more closely on an existing course than either of the other two fundamentals classes.) The emphasis is a “vertical slice” through the layers of abstraction that comprise computer design, including a gradual, bottom-up evolution from 0s and 1s up to basic processor architecture and an integrated hardware and software laboratory that closely tracks the lectures. In the older curriculum, this course was our students’ first exposure to ECE, and their first hands-on hardware laboratory. The new course benefits from the exposure our freshmen have already had to basic digital elements and real design problems. Hence, the new course is a slightly more aggressive version of its predecessor 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.

Structure

Overall, the course is designed to demystify computers for our students. It builds up the concepts that define and inform each layer of the conventional design hierarchy: combinational circuits, sequential circuits, register transfer level, stored program computing, control path / data path partition and implementation, rudimentary instruction set architecture and assembly language programming.

Advances in programmable logic [12] (*e.g.*, PALs, field programmable gate arrays, *etc*.) and associated design software have made it possible for laboratory assignments to target more interesting problems; recent examples include floating point arithmetic hardware, chess move generation and video game controllers. Use of a software-simulated processor for assembly language programming allows campus-wide electronic access and easy alteration of the instruction set (via simulated microcode) for faculty and students. The course provides a foundation for students who wish to pursue more advanced computer engineering courses, and for those whose interests lie elsewhere in ECE.

This course has a lecture section, weekly recitation sections and a weekly lab session. A syllabus appears in FIGURE 10.

We note that this is a particularly large course for ECE, with typically 100 to 150 students per semester. This is so because the course is a core requirement for both ECE undergraduates, and for computer science undergraduates at Carnegie Mellon. One result of this is that we have had to be somewhat cautious in how hard we try to exploit the design expertise that our own ECE undergraduates gain from their freshman *Introduction to ECE* course. Our computer science students, who intermix freely with our ECE students each semester, have more software expertise, but no hardware expertise. Accordingly, we spend the first few weeks of lab gently reintroducing the relevant hardware skills to our ECE students, while rapidly drilling our computer science students in these same basic skills.

Analysis

As mentioned above, this course actually more closely resembles its predecessors in the old ECE curriculum than does *Fundamentals of EE*, the introductory linear circuits course. It maintains the same flow through combinational design, sequential design, and microprocessor organization and programming, but with more aggressive laboratory design assignments. The course is popular, remains highly rated by our students, and prepares our students to succeed in their more advanced computer hardware and software classes.

FIGURE 10. Syllabus: Sophomore Fundamentals of Computer Engineering

Binary number systems

- Positional number systems; useful radices: unsigned binary, octal and hexadecimal
- Signed numbers: sign magnitude, one's complement and two's complement
- Floating point numbers: sign, mantissa, exponent; basic arithmetic operations

Combinational circuit design

- SSI logic gates
- Boolean algebra and canonical forms
- Minimization via Karnaugh maps; via Boolean n-cubes; via classical Quine-McCluskey procedure; via 2-level and multi-level computer synthesis tools
- MSI building blocks: multiplexers, decoders, ROMs and PLAs
- Arithmetic circuits for fast addition, multiplication

Sequential circuit design

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

Rapid prototyping technology

- Field programmable gate array architectures and applications
- Synthesis and optimization tools for FPGAs

FIGURE continued

Processor design

- Register-transfer level ideas, abstractions, design style
- Stored program computers, data path and control path partitions
- Example processor design: a simple instruction set
- General data path design techniques, example data path design
- General control path design techniques: hardwired control, microprogrammed control; example microprogrammed control path design

Assembly language programming

- Basics of storing, manipulating, moving data on a simulated 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

Elective: Sophomore Mathematical Foundations of EE

Mathematical Foundations of Electrical Engineering is, as of this writing, a proposed new course to be offered for the first time in the fall of 1995. It is the result of our ongoing dissatisfaction with the random gaps in crucial mathematical knowledge that even our best students exhibit, despite our best efforts to better integrate mathematics and engineering in the first two years of our new curriculum.

Structure

A syllabus appears in FIGURE 11. It is worth noting an important philosophical point here: *we do not regard this solely as a math class*. Rather, this is a class whose purpose is to build basic mathematical techniques that appear widely in engineering applications (and that are not well integrated from an engineer's perspective when not taught in an engineering context).

This course covers topics from engineering mathematics that serve as foundations for descriptions of electrical engineering devices and systems. It has four major parts:

- Complex analysis, including complex numbers and complex analytic functions
- Ordinary differential equations of first- and second-order
- Linear algebra, including matrices, vectors and determinants
- Vector calculus, including the vector differential operators gradient, divergence and curl, and vector integral calculus, including multiple integrations and integral theorems.

The proposed text is *Advanced Engineering Mathematics*, seventh edition, by E. Kreyszig.

In our current curriculum, the course can replace the required linear algebra corequisite for the required core *Fundamentals of EE* sophomore class (hence the treatment of complex analysis, differential equations and linear algebra). It can also replace the required prerequisite three-dimensional calculus class that serves as a gateway (or, as we now believe, an *obstacle*) to the first electromagnetic fields class, and to related magnetics and data storage system classes.

Analysis

The earlier mention of a prerequisite “obstacle” merits further discussion. In our old curriculum, all math classes were essentially required, so there was never any choice involved in selecting a math (or science) class so that a later engineering class could be elected. This is no longer true in the new curriculum. The beginning breadth classes in each breadth area (recall, these are: applied physics, signals & systems, circuits, computer hardware, computer software) can individually require extra mathematics or science classes beyond those mandated in the ECE math and science core. A somewhat unexpected result here is that now, some introductory breadth courses are “easy” to take because they require only mathematics, science and ECE courses that all students take anyway. But some of courses are now “difficult” to take because they require extra mathematics preparation. The notable example is the electromagnetics sequence, which required an extra three-dimensional calculus class. As well as filling in some critical gaps in our students mathematical knowledge, the new *Mathematical Foundations of EE* class is also intended to lower the barrier to these breadth courses that would prefer a more extensive mathematical background, but which have seen declining enrollments (see Chapter 6 for more on this) in part because of these extra obstacles to enrollment.

FIGURE 11. Syllabus: Proposed Elective, Mathematical Foundations of EE

Complex analysis

- Complex numbers, complex analytic functions

Ordinary differential equations

- First-order differential equations
- Second-order differential equations

Linear algebra

- Matrices, vectors and determinants

Vector calculus

- Vector differential calculus, grad, div, curl
- Vector integral calculus, integral theorems