Updating the contents and structure of Computer Engineering

Larry Hughes
Electrical and Computer Engineering
Dalhousie University
18 November 2016

1 Introduction

The Department of Electrical and Computer Engineering at Dalhousie University offers a Computer Engineering option as part of its Electrical Engineering degree. The option was originally developed in the early 1990s and consisted of five courses:[1] *Systems Analysis*, *Computer Architecture*, *Operating Systems*,[2] *Real Time Systems*, and *Computer Networks*. At its peak in the late 1990s, the number of Computer Engineering students often equalled or exceeded the number of Electrical Engineering students.

In the early 2000s, the option was turned into a distinct degree (i.e., a Computer Engineering as opposed to Electrical Engineering with a Computer Engineering option). However, by 2009, falling enrollment in Computer Engineering and a push to consolidate programs within the Faculty of Engineering resulted in the Computer Engineering degree being returned to its designation as an Electrical Engineering option.

At this time, a long-standing difference between the Associate Universities and the Department over the teaching of *Systems Analysis* in second year (which many of the Associate Universities were unable to offer) was resolved by moving the course into third year. Other program design changes were made without thought for the impact on the program.

As a result, students pursuing the Computer Engineering option now take an introduction to programming course in their first year and no other computer-related courses until third year. In third year, students take three courses in the first term (*Systems Analysis*, *Data Structures*, and *Microprocessors*) and two in the second term (*Computer Architecture* and *Operating Systems*). Not only is this proving to be an overly demanding workload for many students, especially in the second term, some of them learn, halfway through their penultimate year that they have no interest whatsoever in Computer Engineering.

It is generally agreed by students and those faculty involved with the Computer Engineering option that changes are needed to order the courses in such a way as to maximize the educational benefits to the students.

This is only one of the changes needed to improve the quality of the Computer Engineering option. Over the past decade there has been an evolution in the recommended body of knowledge required by Computer Engineers, reflecting some of the changes taking place in computers, computer systems, and their applications. These changes are detailed in the ACM-IEEE report, *Computer Engineering Curricula 2016* (CCCE2016, 2016).

[1] Three other computer-related courses are taught to Electrical Engineering students as well as those in the Computer Engineering option: *Introduction to Programming*, *Data Structures*, and *Microprocessors*.
[2] *Operating Systems* is taught by the Faculty of Computer Science.

If Dalhousie's Computer Engineering program is to remain relevant, it will be necessary to make changes, both to the contents and its structure. The following report discusses ways in which this can be achieved, given the constraints placed on it by the faculty (limiting the number of changes in the first two years of the Engineering degree) and the Department (the cap on the number of Computer Engineering courses).

2 An overview of the IEEE-ACM recommendations for Computer Engineering

Over the past 20 years, the ACM and the IEEE have collaborated in the development of a set of recommendations for undergraduate Computer Engineering. The current recommendations were released in 2008. Given the changes that have taken place since then, ACM and IEEE have published a new set of recommendations for 2016 (referred to as CE2016).

The remainder of this section examines some of the issues raised in CE2016.

2.1 How Computer Engineering differs from Electrical Engineering and Computer Science

CE2016 makes a distinction between computer engineers, electrical engineers, other computer professionals, and engineering technologists. Although there are clear overlaps between these different groups, CE2016 expects that, in general, computer engineers should satisfy the following three characteristics:

1. *Possess the ability to design computers, computer-based systems and networks that include both hardware and software and their integration to solve novel engineering problems, subject to trade-offs involving a set of competing goals and constraints. In this context, "design" refers to a level of ability beyond "assembling" or "configuring" systems.*
2. *Have a breadth of knowledge in mathematics and engineering sciences, associated with the broader scope of engineering and beyond that narrowly required for the field.*
3. *Acquire and maintain a preparation for professional practice in engineering.*

2.2 Breadth of knowledge

CE2016 expects all computer engineering graduates to exhibit the following characteristics:

- *System Level Perspective – Graduates should appreciate the concept of a computer system, the design of the hardware and software for that system, and the processes involved in constructing, analyzing, and maintaining it over the lifetime of the system. They should have an understanding of its operation that goes to a greater depth than a mere external appreciation of what the system does or the way(s) in which one uses it.*
- *Depth and Breadth – Graduates should have familiarity with topics across the breadth of the discipline, with advanced knowledge in one or more areas.*
- *Design Experiences – Graduates should have completed a sequence of design experiences, encompassing hardware and software elements and their integration, building on prior work, and including at least one major project.*

- *Use of Tools – Graduates should be capable of utilizing a variety of computer-based and laboratory tools for the analysis and design of computer systems, including both hardware and software elements.*
- *Professional Practice – Graduates should understand the societal context in which engineering is practiced, as well as the effects of engineering projects on society.*
- *Communication Skills – Graduates should be able to communicate their work in appropriate formats (written, oral, graphical) and to critically evaluate materials presented by others in those formats.*

2.3 The future

CE2016 expects that the need for computer engineering will increase as computing and electronic technologies become more complex. In response, undergraduate programs can be expected to evolve in different ways; for example:

- *An expanded content from and tighter integration with computer science,*
- *Collaboration with the software engineering discipline on application-focused projects and embedded systems with a greater emphasis on design and analysis tools to manage complexity, or*
- *A re-integration with electrical engineering, as computer-based systems becomes dominant in areas such as control systems and telecommunications.*

2.4 ACM-IEEE CE2016 Recommended Knowledge Areas

CE2016 has 13 recommended Knowledge Areas, which are listed in Table 1, along with the total number of recommended core hours (420 in total).

Table 1: CE2016 Recommended Knowledge Areas

Designation	Knowledge area	Core hours
CE-CAO	Computer Architecture and Organization	60
CE-CAE	Circuits and Electronics	50
CE-DIG	Digital Design	50
CE-ESY	Embedded Systems	40
CE-SWD	Software Design	40
CE-CAL	Computing Algorithms	30
CE-SGP	Signal Processing	30
CE-SPE	Systems and Project Engineering	30
CE-NWK	Computer Networks	20
CE-PFP	Professional Practice	20
CE-SEC	Information Security	20
CE-SRM	Systems Resource Management	20
CE-SET	Strategies for Emerging Technologies	10

The remaining courses are then taken in the student's specialized area, such as Electrical Engineering or Computer Science.

3 The existing Computer Engineering program

The Computer Engineering courses (both required and shared with Electrical Engineering students) at Dalhousie and the terms in which they are taught are listed in Table 2.

Table 2: Course offerings and academic term in which they are taught

Year	Term	Course
1	1	Introduction to programming
	2	
2	3	
	4	
3	5	Systems Analysis Microprocessors Data Structures
	6	Computer Architecture Operating Systems
4	7	Computer Networks
	8	Real Time Systems

In the first three terms of study, potential Computer Engineering students have one programming course in first year, yet students are expected to select Computer Engineering as their program of study based on this one course.

In year 3, the students are exposed to their first Computer Engineering course (*Systems Analysis*) and two computer-related common courses (*Microprocessors* and *Data Structures*).

Table 3 is an estimate of the number of hours the Computer Engineering students are exposed to CE2016's Knowledge Areas. While these numbers are simply estimates, they give an indication of what is needed to improve the program.

Table 3: CE2016's recommended Knowledge Areas compared with existing course offerings (some courses exceed CE2016's core hours)

CE2016 Knowledge area	Core hours	Service courses	Courses shared with EE	Systems Analysis	Microprocessors	Computer Architecture	Real Time Systems	Operating Systems	Computer Networks	Total	Meets or exceeds core-hours
Computer Architecture and Organization	60				10	15	5	5		35	No
Circuits and Electronics	50		50							50	Yes
Digital Design	50		50							50	Yes
Embedded Systems	40				5		15	5		25	No
Software Design	40	5		5	5	15	5	5		40	Yes
Computing Algorithms	30	10		5	5		5	5		30	Yes
Signal Processing	30		30							30	Yes
Systems and Project Engineering	30		15	15						30	Yes
Computer Networks	20								30	20	Yes
Professional Practice	20	20								20	Yes
Information Security	20							5		10	No
Systems Resource Management	20			5				5		10	No
Strategies for Emerging Technologies	10									0	No
Total	420									360	No

The table shows that:

- The core-hour requirements are not met for the Knowledge Areas: Computer Architecture and Organization, Embedded Systems, Information Security, Systems Resource Management, and Strategies for Emerging Technologies.
- The core-hours in the Software Design knowledge area are met in large part because *Computer Architecture* devotes time to covering software design principles and concepts (at the expense of the Computer Architecture and Organization knowledge area).
- Although not apparent from the table, the existing courses give Computer Engineering students limited exposure to different computer architectures.

In short, the Computer Engineering students are weak in Knowledge Areas relating to hardware and software.

4 Updating and restructuring the program

At present, the Computer Engineering program has fallen behind what CE2016 recommends for both a number of Knowledge Areas and the number of core-hours. Meeting these

recommendations is constrained by the limit on the number of Computer Engineering courses permitted by the Department.

The following section presents a possible Computer Engineering program that meets the CE2016 recommendations for Knowledge Areas and core-hours by:

- Adding a new, introductory Computer Engineering course, tentatively titled, *Design I for Computer Engineers*, which gives Computer Engineering students in term 4 an introduction to a number of in Computer Engineering Knowledge Areas dealing with Computer Architecture and Organization and Embedded Systems.
- Splitting the existing *Microprocessor* course shared with Electrical Engineering systems into a distinct Computer Engineering course, *Microsystems*, with a heavy focus on Computer Architecture and Organization (as well as Embedded Systems and Software Design).
- The Faculty of Engineering creating an additional programming course, *Programming II* (equivalent to the existing *Data Structures* course), for students in terms 2 or 3.

Table 4 presents a possible Computer Engineering program that meets CE2016's core-hour requirements using seven Computer Engineering courses. These courses meet half (210 hours) of CE2016's total of 420 core-hours. The program is far from ideal; however, this is an example of what could be done given Faculty and Departmental constraints.

Table 4: A sample Computer Engineering program that meets CE2016 requirements

CE2016 Knowledge area	Core hours	Service courses	Courses shared with EE	Design I for Computer Engineers	Systems Analysis	Microsystems	Computer Architecture	Operating Systems	Real Time Systems	Computer Networks	Total	Meets or exceeds core-hours
Computer Architecture and Organization	60			10		15	20	5	10		60	Yes
Circuits and Electronics	50		50								50	Yes
Digital Design	50		50								50	Yes
Embedded Systems	40			10		10		10	10		40	Yes
Software Design	40	20			5	5	5	5			40	Yes
Computing Algorithms	30	20		5	5						30	Yes
Signal Processing	30		30								30	Yes
Systems and Project Engineering	30		20		10						30	Yes
Computer Networks	20									20	20	Yes
Professional Practice	20	20									20	Yes
Information Security	20				5			5	5	5	20	Yes
Systems Resource Management	20				5			5	5	5	20	Yes
Strategies for Emerging Technologies	10			5			5				10	Yes
Total	420	60	150	30	30	30	30	30	30	30	420	Yes

4.1 Four year (eight-term) program

As discussed in section 3, Computer Engineering students would benefit from changes to both the contents and structure of the existing Computer Engineering program. An example of a four year (eight-term) structure teaching the proposed courses (see above) is shown in Table 5.

Table 5: A possible four year (eight-term) structure

Year	Term	Course
1	1	Programming I
	2	Programming II
2	3	Design I for Computer Engineers
	4	Systems Analysis
3	5	Microsystems
	6	Computer Architecture
4	7	Operating Systems Computer Networks
	8	Real Time Systems

Advantages: The program fits into eight terms. Students are exposed to Computer Engineering concepts earlier than in the existing program and in a more logical order. Perhaps more importantly, five Computer Engineering courses aren't taught in year 3.

Disadvantages: Significant changes are required in year 2, with the introduction of two Computer Engineering courses. Two Computer Engineering courses are taught in term 7, increasing the student workload. Moreover, whether the program could be completed in eight terms is another issue entirely.

4.2 Nine-term program

Experience has shown that an increasing number of students are taking nine or more terms to complete their degree. The principal reason for this is the lack of preparedness the students face when taking *Computer Architecture* and *Operating Systems* in term 6. Those that remain in the program often reduce their course load and repeat the missed courses in term 6 the following year. This effectively turns their eight-term program into nine terms.

Since a significant number of students appear to be taking this approach, Table 6 presents a possible nine-term program.

Table 6: A possible nine-term structure – I

Year	Term	Course
1	1	Programming I
	2	Programming II
2	3	Design I Computer Engineers
	4	Systems Analysis
3	5	Microsystems
	6	Computer Architecture
4	7	Operating Systems
	8	Computer Networks
5	9	Real Time Systems

Advantages: The Computer Engineering courses are taught in separate terms, leveling the workload.

Disadvantages: Significant changes are required in years 1 and 2 which are unlikely to be adopted.

An alternative nine-term program that is less disruptive to years 1 and 2 is presented in Table 7, with only one Computer Engineering course taught in year 2.

Table 7: A possible nine-term structure – II

Year	Term	Course
1	1	Programming I
	2	
2	3	Programming II
	4	Design I for Computer Engineers
3	5	Systems Analysis Microsystems
	6	Computer Architecture
4	7	Operating Systems
	8	Computer Networks
5	9	Real Time Systems

Advantages: Only one Computer Engineering course is taught in year 2.

Disadvantages: Two Computer Engineering courses are taught in term 5.

4.3 Five year (ten-term) program

A five-year program, such as the one shown in Table 8, has the advantage that the Computer Engineering students take one Computer Engineering course per term, starting in term 4. The order of the courses reflects a steady expansion of a student's body of knowledge in Computer Engineering.

Table 8: A possible five year (ten term) structure

Year	Term	Course
1	1	Programming I
	2	
2	3	Programming II
	4	Design I for Computer Engineers
3	5	Systems Analysis
	6	Microsystems
4	7	Computer Architecture
	8	Operating Systems
5	9	Computer Networks
	10	Real Time Systems

With respect to the proposed structure:

Advantages: Only one course specific to Computer Engineering (*Design I for Computer Engineers*) is taught in term 4. The remaining six courses are taught, one per term, during the final three years. A student wanting to complete in nine-terms could do so by taking *Operating Systems* (term 8) and *Computer Networks* (term 10) simultaneously.

Disadvantages: The program is spread over five years, which may discourage students from taking it, although when TUNS was a separate university, its programs were five years in

length. Some of the Associate Universities might not have sufficient instructors to teach *Design I for Computer Engineers*.

5 Recommendations

Dalhousie's existing Computer Engineering program needs to update its contents to reflect the changes taking place in the field and its structure to recognize the increasing student workload.

In addition to a new, faculty-wide programming course in the first two years, it is recommended that:

- A new course, *Design I for Computer Engineers*, be added to the Computer Engineering program in year 2. This would expose term 4 Computer Engineering students to a solid grounding in hardware and software as recommended by CE2016.
- The existing microprocessor course, currently common to both Computer and Electrical Engineering students be split in two, with a distinct Computer Engineering course on microsystems. This course would be the first in-depth examination of computer architecture and organization, showing the relationship between hardware and software.

This report has also shown the difficulties in attempting to meet the Knowledge Areas and core hours recommended in CE2016 with only seven Computer Engineering courses. There are at least two ways in which this can be done:

- Increase the number of Computer Engineering courses, focussing on computer architecture, computer organization, and embedded systems.
- Increase the number of Computer Science courses, focussing on software design, computing algorithms, and information security.
- Redesign the entire Computer Engineering program using one or more of the CE2016 sample programs as a template.

Finally, if the contents and structure of the Computer Engineering program are updated, the university should consider reintroducing the Computer Engineering degree.

References

CCCE2016. (2016). *Computer Engineering Curricula 2016 - Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering (Interim Curriculum Report 2016).* Association for Computering Machinery (ACM) and IEEE Computer Society. Retrieved October 31, 2016, from https://www.computer.org/cms/Computer.org/professional-education/curricula/ComputerEngineeringCurricula2016.pdf