A Forward Looking Digital Curriculum In Electrical Engineering

Joerg Mossbrucker
Department of Electrical Engineering and Computer Science
Milwaukee School of Engineering
Milwaukee, WI

Abstract

This paper describes the new digital track in the Electrical Engineering program at the Milwaukee School of Engineering (MSOE). It uses a combined top-down bottom-up approach. Students are exposed to a number of programming languages on embedded systems in three courses starting in the Freshmen year. Digital logic design ranging from simple gate logic to complex programmable logic devices is covered in two courses. In addition, a sixth course focuses on systems interfacing and mobile robots. These changes assure breadth and depth of knowledge in the digital field of Electrical Engineering.
This paper focuses on both, the goals and objectives of the entire digital curriculum and the objectives and contents of the individual courses in the digital track.

Technological changes in the Electrical Engineering field

In the last ten years the field of Electrical Engineering has undergone a tremendous shift towards digitalization of almost everything. This is readily apparent in most consumer products and is evident for example in the shift from traditional control to digital control[1]. It does not mean that the analog side of Electrical Engineering is suddenly being replaced but rather that electrical engineers use digital systems as the controlling mechanism. This trend goes hand in hand with the increased usage of microcontrollers for systems control. Recent advances in 8-bit microcontroller technologies along with dramatic cost reductions increased the usage of these low-end controlling devices. It is estimated that by the year 2005 a total number of 5 billion 8-bit microcontroller units are shipped annually[2]. MSOE has realized that the microcontroller has become one of the core elements in an Electrical Engineering design and has, therefore, shifted the focus of its digital track.

Objectives of the digital track in the EE program

Objectives of the digital track in Electrical Engineering can be grouped into the following areas:

1. Programming languages

   a. Knowledge of designing and implementing computer programs using various techniques, such as algorithm development, flowcharts etc.
   b. Knowledge of at least one procedural programming language.
   c. Knowledge of at least one objective-oriented programming language (possibly an extension of the procedural language).
   d. Knowledge of at least one assembly language.
2. Embedded Systems
   a. Knowledge of programming embedded systems in HLL and assembly.
   b. Knowledge of designing embedded systems.
   c. Knowledge of interfacing embedded systems to real-life sensors and actuators.
3. Digital Signal Processing
   a. Knowledge of the traditional fields of DSP, such as discrete time signals and systems, difference equations, z-transform, FIR and IIR filters, etc.
   b. Knowledge of discrete and fast Fourier transform.
4. Digital Logic and Circuits
   a. Knowledge of number systems, codes, Boolean algebra.
   b. Design of combinational and sequential logic circuits.
   c. Design of digital circuits using SSI, MSI, and PLDs.
   d. Design of digital systems and subsystems using basic digital building blocks, such as multiplexers, decoders, full adders, ROMs etc.
   e. Design of sequential circuits using various representations such as state diagrams, ASM charts, and VHDL.
   f. Design of the Data Path and Control Path of a Control Unit.
   g. Design of a Control Unit as a FSM and microprogramming.

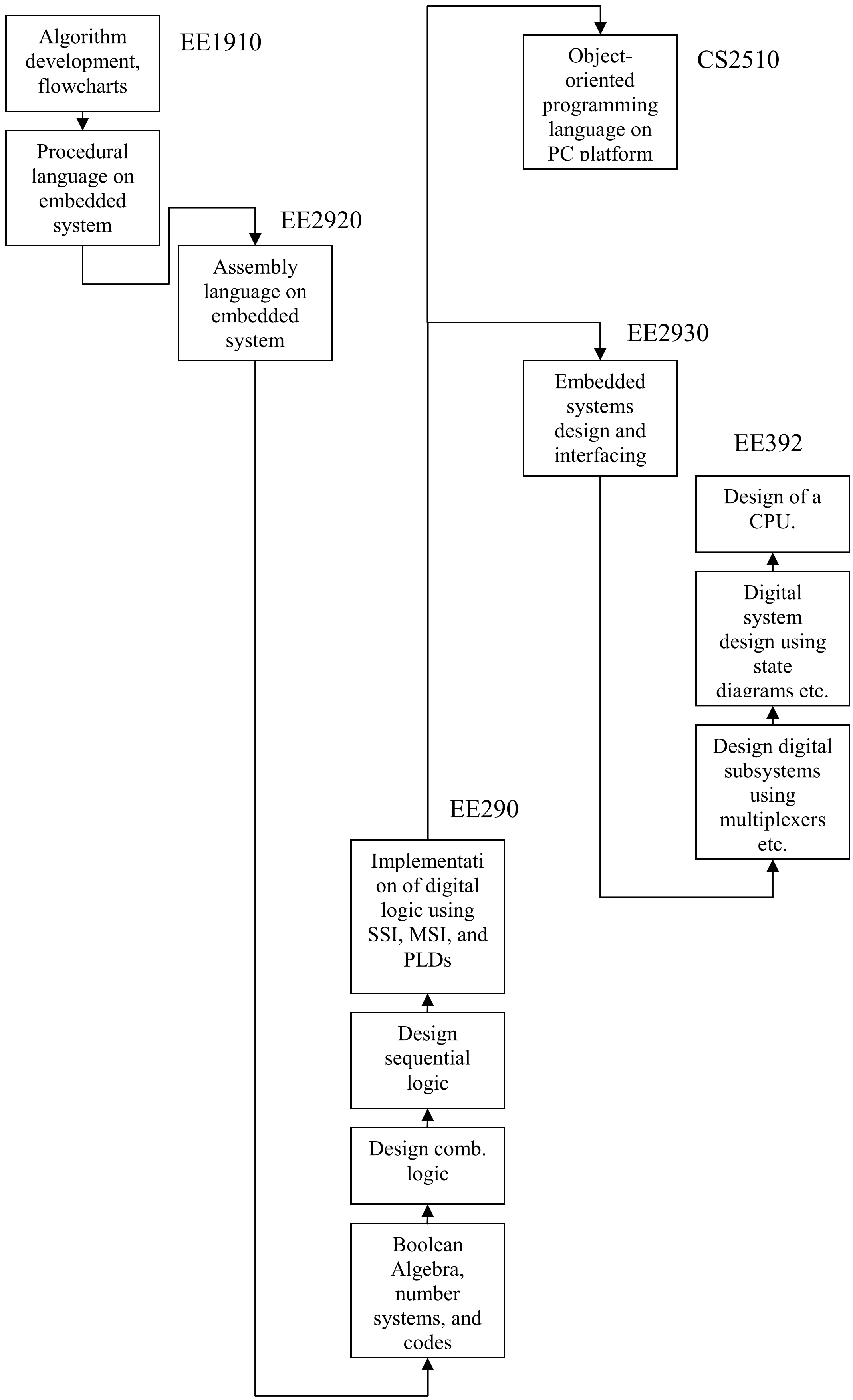
It has been decided that a combined top-down bottom-up would be the best approach to conquer the above list.

The objectives concerned with programming languages and embedded systems are covered with a top-down approach, while DSP and Digital Logic and Circuits are covered with a bottom-up approach. This exhibits several consequences:

- Since no hardware experience is required in programming microcontrollers in a high level language, objectives 1.a, 1.b, and 1.d can be achieved by using an embedded system as the development platform rather than a personal computer. This has the added advantage that students in Electrical Engineering learn programming in an environment they are most likely to continue using. In addition, these programming classes can be taught relatively early in the curriculum, as it is done at MSOE in the Winter quarter of the Freshman year for objectives 1.a and 1.b (EE1910), and in the Fall quarter of the Sophomore year for objective 1.d (EE2920).

- Since objectives 1.a, 1.b, and 1.d are performed on an embedded system, there is no need for additional coverage of objective 2.a.
- Even though implementations of compiler for object-oriented languages do exist on 8-bit microcontroller platforms, it has been deemed necessary that students in Electrical Engineering also learn programming on a personal computer platform. Hence, objective 1.c is covered by an additional course in the Sophomore year using standard PC software as the development environment (CS2510).
- Objectives 3.a and 3.b are given their own course in the junior year, since these objectives are only loosely dependent on the other objectives (EE3220).
- Objectives 4.a, 4.b, 4.c are covered by a course in the fall quarter of the sophomore year immediately after courses covering circuit theory. This course is structured in a bottom-up fashion, starting with Boolean algebra, number systems and codes, and ends in design of combinational and sequential logic circuit design using SSI, MSI, and PLDs (EE290).
- Objectives 2.b and 2.c are covered by a single course in the sophomore year, therefore enabling students to design microcontroller systems and interfacing circuitry (EE2930).
- Objectives 4.d – 4.g are covered in a digital system design course in the senior year in a top-down approach, reaching the topics of the bottom-up approach of combinational and sequential logic circuit design (EE392).

The combined coverage of all topics can be seen in the following figure, clearly indicating the combined bottom-up top-down approach. The horizontal axis shows the different courses in the order left-to-right while the vertical axis shows the order of coverage of the objectives by the individual course. The arrows show order of coverage in the individual course as well as order of courses within the digital track. It has been deemed vital that the bottom-up and the top-down approach reach the same level of abstraction, as it can be clearly seen in the figure.



EE1910
Algorithm development, flowcharts
Procedural language on embedded system
EE2920
Assembly language on embedded system
CS2510
Object-oriented programming language on PC platform
EE2930
Embedded systems design and interfacing
EE392
Design of a CPU.
Digital system design using state diagrams etc.
Design digital subsystems using multiplexers etc.
EE290
Implementation of digital logic using SSI, MSI, and PLDs
Design sequential logic
Design comb. logic
Boolean Algebra, number systems, and codes

Individual course outcomes

The following gives an overview over the individual course outcomes. More information regarding official course descriptions, course topics, laboratory topics, and ABET content can be found on-line at http://www.msoe.edu/eecs/ee/.

EE1910 – Introduction to Computer Programming
Upon successful completion of this course, the student will:

- Design computer software to solve engineering problems
- Develop algorithms, flowcharts, and pseudo code
- Implement algorithms in C
- Identify and implement C elementary data types
- Implement digital I/O statements and I/O formats on an embedded system
- Implement C standard operations
- Implement C arithmetic and assignment statements
- Implement C logical expressions and control constructs
- Implement C looping techniques
- Implement one and two-dimensional subscripted variables
- Implement library functions and user-defined functions

EE2920 – Embedded Systems
Upon successful completion of this course, the student will:

- Explain how a microprocessor, and a microcontroller work.
- Write programs in assembly language.
- Interpret bus timing waveforms.
- Identify different types of input/output devices and their uses.
- Interface different types of I/O devices to a microcontroller.
- Demonstrate the use of interrupt systems, and write programs to use them.
- Demonstrate knowledge of the various I/O features of the microcontroller.
- Design simple microcontroller- based systems.
- Diagnose hardware and software problems.
- Use a Personal Computer for software development and debugging.
- Demonstrate the ability to write a concise, professional report.

EE2901 – Digital Logic Circuits
Upon successful completion of this course, the student will:

- Convert number from one number system to another.
- Perform basic binary arithmetic.
- Evaluate and simplify a Boolean and logic functions.
- Analyze a combinational logic circuit to obtain its switching functions.
- Design a combinational logic circuit, simplify the design, implement the design using SSI, MSI, and programmable logic devices/EVB.
- Analyze synchronous sequential logic circuits.
- Design synchronous logic circuits using state diagram and ASM chart, simplify the design circuits, and implement the design circuits using SSI, MSI, and programmable logic devices/EVB.

- Simulate the design circuits using CAD software and VHDL editor.

CS2510 – Intro to Object-Oriented Programming
Upon successful completion of this course, the student will:

- Design computer software to solve engineering problems using object-oriented programming method.
- Create and use classes.
- Create and use objects.
- Apply encapsulation and information hiding in software design.
- Create and apply derived classes (inheritance).
- Create and apply overloading operator (polymorphism).
- Apply pointers in software design.
- Apply data structure vectors, queue, lists and stacks provided in Standard Template Library in designing software for engineering application.

EE2930 – Systems Interfacing
Upon successful completion of this course, the student will:
- Design and implement hardware and software for complex microcontroller based embedded system.
- Design interfacing circuitry for analog and digital sensors
- Design interfacing circuitry for small and medium power actuators.
- Design microcontroller based networks using various interfaces.
- Interface microcontroller based systems to remote controls.

EE392 – Digital System Design
Upon successful completion of this course, the student will:
- Design a combinational digital subsystem using the basic digital building blocks such as multiplexers, decoders, full adders and ROMs and verify the correct operation of the design through simulation and/or implementation
- Design, simulate and/or implement sequential circuits using various representations such as state diagrams, ASM charts, and hardware description language, specifically VHDL
- Design, simulate and/or implement a digital system as a circuit consisting of a Data Path and Control Unit
- Design the Control Unit as a finite state machine and using micro-programming
- Be able to describe the design and verification process through written communication in the form of laboratory reports

Summary

A combined bottom-up top-down approach has been implemented for the digital track in the Electrical Engineering program of Milwaukee School of Engineering. This combined approach assures not only breadth but also depth of knowledge in the digital field of Electrical Engineering.

Bibliography

1. Too many references to list, an excellent example is Lee, Edward and Messerschmitt, David, "Engineering and Education for the Future", *IEEE Computer Magazine*, Vol. 31, No. 1, January, 1998
2. Peatman, John, "Embedded Design", *Prentice Hall 2003*

Biographical Information

JOERG MOSSBRUCKER is an Assistant Professor at Milwaukee School of Engineering. He received his M.S. and Ph.D from the University of Kaiserslautern / Germany. He has extensive industrial experience and teaches courses in analog and digital circuits, microprocessors, and computer programming.