System-on-chip Design Methodology in Engineering Education

William D. Mensch, Jr.[1] and Dennis A. Silage[2]

[1]*Chairman and Chief Executive Officer, The Western Design Center, Inc., Mesa, AZ, USA, http://www.westerndesigncenter.com*
Tel:(+1)480-962-4545, Fax:(+1)480-835-6442, mensch@westerndesigncenter.com
[2]*Professor and Director, System Chip Design Center, Electrical and Computer Engineering, Temple University, Philadelphia, PA, USA, http://www.temple.edu/scdc*
Tel:(+1)215-204-6761, Fax:(+1)215-204-5960, silage@vm.temple.edu

Abstract: The system-on-chip design methodology is a new paradigm for electrical and computer engineering education in digital logic and microelectronics. The development of a close relationship between the undergraduate course sequence in digital logic and behavioral synthesis and that in electronics and microelectronics has been problematical until now. The system design issues considered by these two historically distinct course sequences usually did not have very much in common. The Western Design Center, Inc., Mesa, AZ, USA and the Department of Electrical and Computer Engineering, Temple University, Philadelphia, PA, USA have now established the System Chip Design Center to address these issues in engineering education and the resulting system-on-chip design methodology. The traditional undergraduate sequence in digital logic consists of three courses: combinational and sequential logic with schematic capture and a register transfer language; a processor architecture and assembly language programming course; and finally a hardware description language course. There would also be a three course undergraduate sequence in electronics: physical devices and electronic circuits; microelectronics; and finally integrated circuit design and fabrication. In this new paradigm a coupling and cross-over occurs between these two undergraduate course sequences with lecture materials and the laboratory assignments that emphasize the target architecture and the reconfigurable environment of system-on-chip design. Utilizing the microprocessor hardware developer boards and a software environment provided by The Western Design Center, Inc. for embedded systems development, undergraduate students investigate system-level design issues that then translate to specific microelectronic cells and integrated circuit design. The computer-aided design tools used in the laboratory are exactly those employed by industry. The system-on-chip design methodology can be successful with three salient traits: faculty in digital logic that recognize the new paradigm; faculty in microelectronics that are attuned to the new system design issues; and the support of industry for the application.

Keywords: system-on-chip, digital logic, microelectronics, engineering education.

1. Two Disparate Threads

Undergraduate electrical and computer engineering education in digital logic and microelectronics has developed in a somewhat disparate manner, until now. Physical integrated circuit (IC) devices are specified electrically and functionally from the manufacturer. The task of the undergraduate student, as an incipient digital logic designer, has been to assemble these packaged ICs into a system by noting their electrical device characteristics, such as the device input and output voltage and current levels, clock and setup timing requirements and propagation delay, and the intended functionality of the system. The undergraduate student who studies microelectronics considers device physics and the complexity of IC layout to meet such electrical characteristics, without much concern for logic system design. However, the new paradigm of the system-on-chip (SOC) design methodology, prevalent now in commercial design efforts, has forced engineering educators to modify and conjoin these two threads.

The traditional three-course sequence in digital logic begins with Boolean algebra and combinational and sequential logic. Initially this first course devoted substantial time and resources to manual methods for combinational logic minimization, such as the Karnaugh map and the Quine-McCluskey technique, using basic logic gates. Today this course introduces computer-aided-design (CAD) tools with either instructional software or, what is more prevalent today, student editions of the same CAD tools as used in industry. These CAD tools feature both

schematic capture and register transfer language (RTL) design techniques in logic design now for programmable array logic (PAL and PLA) and programmable logic devices (PLD).

The second course considers a processor architecture described functionally and by inference from its available operation codes (op-codes) and addressing modes. Assembly language programming of bit, logic, and arithmetic functions and the control of flow of operations of the processor is considered in this course. Hardware interfacing of the processor to its standard peripherals continues to substantiate by repetition the logic design methodology employed for any IC device with fixed electrical characteristics. External parallel bus architectures are presented as the means to integrate these components. Finally, asynchronous interrupts and interrupt programming provide a simple introduction to task management in these processors without an operating system. Processor simulation and emulation, usually on a personal computer, and a cross-assembler, linker, and loader provide the CAD tools for this course.

The third course presents a hardware description language (HDL) for digital logic design and task management in complex field programmable gate arrays (FPGA). Either VHDL or Verilog is the featured HDL, using either the student or professional edition of an appropriate CAD tool. In essence a reconfigurable SOC device, the FPGA provides the undergraduate student with the first modern concept in this series of traditional logic design courses.

At the same time as this logic design sequence is progressing, undergraduate students interested in physical devices and microelectronics are introduced to the IC process technology that would meet the target electrical specification of the device family and the internal organization of the combinational and sequential logic of the device. The traditional three course sequence in microelectronics begins with device physics and electronics circuits. The circuits are analyzed as ideal passive components and lumped parameter device models. However, the complexity of electronic circuit calculations, predicating various manual methods as the Karnaugh map in logic design, produced early CAD tools such as ECAP and then SPICE. Student editions of SPICE remain the primary CAD tool in both this course and the sequel.

The second course considers microelectronics, where complex distributed parameter models are used for both passive components and active devices. Undergraduate students produce simple logic designs from standard microelectronic cells in a given process technology (usually CMOS) by literally drawing the IC using a CAD editor. Assigned design projects begin with combinational logic and progress to sequential logic and finite state machine design. Performance of the design is verified in CAD simulation.

The third course presents very large scale integrated circuit (VLSI) design as an extension of the second course now to more complex devices. For undergraduate students, existing VLSI designs are often modified in project assignments to meet new device specifications. Ancillary design issues, such as on-chip busing, power consumption and propagation delay, may also be presented late in this last course of the sequence. CAD simulation again verifies the performance of the design, although some institutions provide experience in fabrication and testing of the IC.

2. A Single Conjoined Thread

An undergraduate student, who would consider an emphasis in both of these threads, quickly notes the disparity in the treatment of the material. Logic system design issues, prevalent especially in the second and third courses in that sequence, are left unresolved in the traditional microelectronics and VLSI course sequence. Logic designs are now parsed in behavioral synthesis as arbitrary tasks in HDL or in a real-time operating system (RTOS), which, although crucial to the development of complex systems, are not the manner in which traditional undergraduate microelectronic and VLSI designs are presented. Microelectronic design issues are not presented in any detail in the traditional undergraduate logic design course sequence. Yet the SOC design methodology now requires that even the undergraduate microelectronics courses consider such topics as on-chip busing, clock dispersion and signal propagation in a VLSI device now assembled from preconfigured or hard-core IC components.

A new course sequence in a conjoined thread presents the SOC design methodology in these undergraduate courses and provides the curriculum synergy that make this new paradigm possible. The focus of this educational effort is a grant and laboratory provided by The Western Design Center (WDC), Inc., Mesa, AZ, USA: the System Chip Design Center (SCDC) at the Department of Electrical and Computer Engineering (ECE), Temple University, Philadelphia, PA, USA.

In our undergraduate ECE curricula, the first course in digital logic precedes the first course in electronics, as shown in Table 1. The displacement of the courses in their threads is intentional and an important concept in education for the system-on-chip design methodology. In this first course in digital logic only the simplest of electronic issues that are supported by circuit theory courses are presented, such as input and output current and voltage thresholds, and the course concentrates on modern logic design issues. The first course in electronics is offered in the following semester and at the same time as the second course in digital logic. This course introduces

traditional materials on device physics, junction and channel devices, but also presents digital and non-linear active circuits to complement and extend the electronic logic circuit synthesis presented earlier.

Table 1. Digital logic and microelectronics curricula

Fifth Semester	Digital Logic Design	
Sixth Semester	Microprocessor Systems	Electronics
Seventh Semester	Advanced Microprocessor Systems	Microelectronics
Eighth Semester		VLSI Design

In the second course in digital logic undergraduate students use the WDC Software Development System (SDS) CAD tool and hardware developer boards for the W65C02 8-bit and W65C816 16-bit microprocessor. Logic design issues are emphasized initially with laboratory projects that utilize assembly language tasks and standard peripherals, such as read-only (ROM) and static memory devices in specific design projects suitable for undergraduates in this course. Laboratory projects of this type traditionally have used assembly language routines presented as flow-charts; fixed point arithmetic and data manipulation; data smoothing, integration and differentiation; input/output peripherals; and asynchronous interrupt processing. Also presented, though, are the system design issues of integrating disparate but modern components such as relatively slow electrically erasable ROM devices, serial bus peripheral architectures and an introduction to task and memory management in a small nucleus RTOS. The W65C02 and W65C816 microprocessor architectures are described in functional detail as a prelude to the second course in microelectronics.

The second course in microelectronics in the succeeding semester considers CMOS standard IC cell CAD design and layout techniques as an extension to the first course in electronics. Undergraduate students now in this conjoined thread consider the processor and peripheral architecture introduced in the previous second course in digital logic [1]. This microelectronic course also considers the physical integration of hard-core IC devices [2]. Distributed parameter analysis is presented in support of on-chip bus termination and loading issues and power dissipation. At the same time, the third course in digital logic discusses HDL and its natural relationship to task management in complex digital systems. The WDC W65C02 and W65C816 hardware developer boards, which include an FPGA system peripheral, are used for SOC projects with single or multiple boards. Additional FPGA peripherals programmed in the Verilog HDL are used to implement digital data communication encoders and decoders, data protocol converters, or message passing coprocessors.

The conjoined thread finishes after the third course in digital logic with a capstone course in VLSI that integrates the concepts of these undergraduate courses in design projects. The students also investigate and try to adhere to the proposed standards for SOC development [3]. Industry standard CAD tools in the SCDC assist in the conversion of HDL specific components to VLSI layouts. The design project VLSI devices are verified in simulation.

3. What's Different?

What is different in this undergraduate electrical and computer engineering curriculum? It could be said that any group of undergraduate digital logic and microelectronic engineering educators can assembly such a curriculum, but there are some unique pieces that have coalesced in ECE at Temple University. These traits seem to be sufficient, but may be even necessary, to affect this change in engineering education [4].

First, the faculty responsible for the undergraduate digital logic sequence must become aware and convinced that the SOC paradigm is as much of a revolution in digital design in the 2000s as the introduction of the microprocessor was in the 1970s. It would also be highly instructive if the faculty were exposed, first hand, to both of these revolutions! Engineering educators often try to delay the introduction of so-called "new" curricula with the dictum that undergraduates are burdened with too much material. This had delayed, in many instances, the introduction of microprocessor courses in undergraduate ECE education. They must also be reasonably well versed in microelectronics.

Second, the faculty responsible for the undergraduate microelectronics sequence must be attuned to the design issues in SOC. Likewise, they must also be reasonable well-versed in digital logic design. This conjoined thread affinity must be encouraged by a concerted faculty development effort from the individuals, the department, and even external agencies that support engineering education.

Finally, industry must be supportive, not only of the conjoined threads in digital logic and microelectronics, but of the SOC design paradigm which, although new, is organizing as a distinct discipline quite rapidly. This has translated as support for design tools from CAD vendors, but now must be expanded to encourage the end-user

product industry to consider the work-for-hire aspects of engineering education and research not only at the graduate, but also now at the undergraduate level.

4. References

[1]. A.M. Rincon, W.R. Lee and M. Slattery, "The changing landscape of system-on-a-chip design," IEEE Custom Integrated Circuits Conference, pp. 83-90, May 1999.

[2]. E. Wein, "Core integration: overview and challenges," IEEE/ACM International Conference on Computer-Aided Design, 1998, pp. 450-452, November 1998.

[3]. M. Birnbaum and H. Sachs, "How VSIA answers the SOC dilemma," IEEE Computer, vol. 32, pp. 42-50, June 1999.

[4]. A. Smailagic, R. Brodersen, and H. DeMan, "Future systems-on-a-chip: impact on engineering education," Proceedings IEEE Workshop in System Level Design, pp. 78-83, April 1998.

The Western Design Center, Inc., http://westerndesigncenter.com

System Chip Design Center, Department of Electrical and Computer Engineering, Temple University, http://www.temple.edu/scdc