Excellence in Computer Engineering Education (EXCEED): Integrating PDC Topics in the ECE Courses at UIUC

Dong San Choi, Yuting Chen, Zuofu Cheng, and Ujjal Bhowmik
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Urbana, IL 61801

I. Motivation and Background

The widespread presence of multicore and general-purpose graphics processing units (GPUs) in PCs, laptops, and now even handhelds has changed the computing landscape and empowered users to make valued, innovative contributions to the technology of computing. However, the current undergraduate Electrical and Computer Engineering (ECE) curriculum of the University of Illinois at Urbana-Champaign (UIUC) at the introductory level has not yet adequately addressed the needs for this rapid change in computing hardware platforms, devices, languages, and supporting programming environments. Recognizing this urgency, the principle goal of our EXCEED project is to expose our freshmen and sophomore students to the concepts of parallel and distributed computing (PDC) by integrating modules of PDC topics in the existing curriculum. The project is currently supported by the Strategic Instruction Innovation (SIIP) grants from UIUC. Our main focus is to allow our students to develop parallel computational thinking while they learn the core topics on von-Neumann-model-based sequential computational techniques. To achieve this goal, we will introduce module-based PDC topics into a series of existing courses (ECE 120, ECE 220 and ECE 385) following the NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing - Core Topics for Undergraduates" guideline [1]. Emphasizing "early and often" as the key to success, we intend to integrate PDC concepts in the ECE 120 (Introduction to Computing) and ECE 220 (Computer Systems and Programming) curriculum in a level-appropriate manner, while not harming the coverage of the traditional course content. As a test case of our implementation, we have started integrating byte sized PDC lecture modules on selected digital logic design topics in the Fall 2019, ECE120 course. For ECE 220, we will initially target our honors freshmen/sophomore student cohort in the Spring 2020. We also intend to introduce PDC modules as honors projects in ECE 385 (Digital Systems Laboratory), an advanced level Computer Engineering (CompE) course. Since ECE 120, ECE 220 and ECE 385 are taught in the freshmen, sophomore and junior levels, respectively, it will give us the opportunity to observe how students perform with/without having prior knowledge in PDC concepts. Based on the outcome of our results, we will take necessary steps to introduce proposed PDC topics in the regular course syllabus. Our work will be an important step in continuing to prepare our students for the impending ordinaries of PDC. As our students become experienced in many aspects of PDC and develop parallel computational thinking, we expect to see more integration of advanced PDC topics in our upper level courses and most importantly in the senior capstone course. To evaluate the successes and failures of our proposed PDC topics adoption, we will consider the following two approaches. First of all, we will introduce Focus Questions related to proposed PDC topics in Quizzes. Student performance in the Focus Questions and its impact on the overall course outcomes will be studied and documented carefully. Furthermore, we will measure pre-course student performance in ECE 408 (applied parallel programming) to evaluate the impact of early introduction of PDC topics in ECE 120, 220 and 385. These findings will be brought up in the weekly course staff meeting for open discussion and suggestion for continuous improvement process. Finally, we intend to contribute to the ongoing conversation about what a core PDC curriculum should be, along with the best practices in its adoption to achieve the intended outcomes.

II. Proposed EXCEED Implementation

The proposed project will span two years, along with two preparatory summers. Figure 1 shows the implementation phases. During Summer 2019, we prepared PDC modules for the first month of the Fall 2019 semester in ECE 120. Currently, a graduate TA (GTA) is developing the remaining PDC modules for ECE120 under our guidance. All developed modules will be posted on the ECE 120 course website. A couple of modules have already been posted on the ECE120 Fall 2019 course website, https://wiki.illinois.edu/wiki/display/ece120/Home. Modules are selected following the guideline in [1]. We intend to replace the supplemental advanced topics in our existing ECE120 lecture slides with the PDC modules. During the Fall 2019 semester, the course coordinator of ECE 120 shared the developed PDC modules among all course instructors for comments and necessary revision before introducing the modules in the lectures. Concurrently, the course coordinator of ECE 220 is working with a GTA to develop PDC modules for the honors track, which will be introduced in Spring 2020. In Spring 2020, we will analyze students' performance in ECE 120 in the previous semester and share the findings with the other course instructors and course director for necessary revision. A pre-course survey will be conducted in ECE 220 to check the retention of PDC learning in ECE 120 from the previous semester. During the same semester, a GTA will implement and lead the PDC honors track in ECE 220. In Summer 2020, we will analyze and report student performance in ECE 120 and ECE 220 and take necessary measures to update the PDC modules for future semesters. The outcome in ECE 120 and ECE 220 will provide necessary information to develop upper level PDC design projects to be introduced in ECE 385. In Summer 2020, the course director of ECE 385 along with the other project members and a GTA will design and implement the PDC honors track projects, which will be introduced in Fall 2020.

Following is the course description and proposed PDC topics to be included in the existing ECE 120, ECE 220 and ECE 385 curriculum. The topics are selected following the “NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing - Core Topics for Undergraduates” guideline**.**

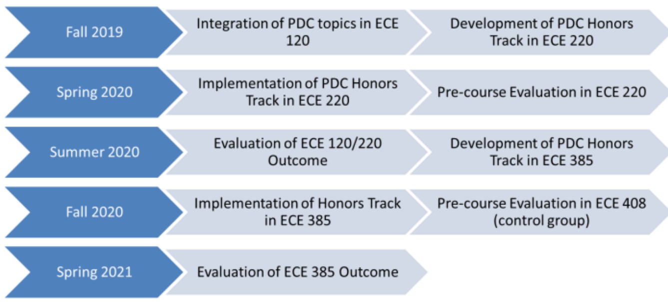

Figure 1. Flow-chart of the proposed EXCEED Project implementation phases.

III. Course Description with intended PDC Topics

A. ECE 120 – Introduction to Computing

This course introduces students the computer systems organized as a systematic set of transformations. Starting with binary representation, arithmetic operations, CMOS and Logic gates, Boolean expression, combinational logic structures, Bit-sliced design, decoder, multiplexers, latches flip-flops, finite state machines models (FSM), and from FSM to computer datapath, the course teaches the basics of a simple LC-3 computer design, instruction set architecture (ISA) of LC-3 computers, and programming techniques using LC-3 assembly language. Thus, this course provides an excellent context in which to consider big picture ideas in PDC. Coverage may include, parallel and distributed computing concepts in digital logic [2] (such as, Number representations, Fan-in and Fan-out of gates, Tristate Gates and Buses, Timing Analysis, Karnaugh Map, Adder, Multiplexers, Counters, Shift registers and Flip-flops). Vaidyanathan, R. et.al. provided an excellent framework for introducing parallel and distributed computing (PDC) concepts as they arise naturally in digital logic [1].

B. ECE 220 – Computer Systems and Programming

This course adopted a bottom-up learning approach using LC-3 ISA and assembly language and C/C++ programming. The primary objective is to give students the basic understanding of how the high-level language instruction are executed in the lowest level (ISA) of computer architecture. This course covers the basics of the Assembly Language, C/C++ language and their interrelationship. Topics include variables, types, expressions, control structures, method definition, parameters, arrays, strings, and data abstraction. Topics covered include language syntax, data types, the concepts of variable scope and storage, arrays, structures, functions and function call structure, parameter passing, and the sequence, selection (if-then-else), and repetition control structures (for, while, repeat-until loops). Some of the PDC programming topics that will be covered in the ECE220 course are concurrency and parallelism (A*), decomposition (A*), data and task parallelism (C*), divide-and-conquer (A*), speedup (A*), scalability (C*).

C* = Comprehend so as to paraphrase/illustrate

A* = Apply it in some way.

C. ECE 385 –Digital Systems Laboratory

The course presents the analysis and design of digital circuits, number systems, combinational and sequential circuits. Basic computer arithmetic, applications and implementation of logic design. Because a digital logic course lays the foundation for other courses such as computer organization and architecture, an early exposure to PDC concepts would benefit in the subsequent courses that address PDC topics more directly. Some of the PDC topics that can be implemented as design projects in our ECE 385 curriculum are: Fan-in and Fan-out of gates, Tristate Gates and Buses, Timing Analysis, parallel Adder, Multiplexers, Counters, Shift registers, Latches and Flip-Flops, Finite State Machine, pipelining, Verilog, PLD to FPGA and practical consideration.

Upon successful integration of PDC topics in the courses listed above, the next step is to integrate the Crosscutting and Advanced PDC topics in the CompE senior capstone course. Our ultimate objective is to prepare our students for their future careers in light of the technological shifts towards parallelism through multicores, GPUs, and corresponding software environments by gradual incorporation of all the PDC topics in our undergraduate CompE curriculum.

IV. Evaluation Plan

Evaluation will be an important part of our project. The goals of our evaluation will be to assess how the integrated PDC modules impact student ability to think effectively using parallel computing concepts and gained knowledge in PDC topics. We intend to evaluate our work in several ways. At the end of each phase, we will analyze student surveys of interest and understanding of PDC concepts. Particularly in introductory courses, these surveys can also gauge interest in learning additional PDC concepts in upper-level courses. We will also analyze grades on PDC related focus questions (such as, assignments and test questions) to assess students’ understanding. Finally, we will evaluate our work from our own perspectives, considering the successes and failures of our efforts, obtaining feedback from colleagues, and considering the degree to which our work is adopted and refined further. This will be done both within our department and externally through our contributions to the CDER courseware website [3]. We will also assess the efforts required by the faculty to implement the redesigned courses with PDC topics. The pre and post surveys will also be designed to gather feedback from faculty to aid in the redesign effort.

References

[1] Prasad, Gupta, Rosenberg, Sussman, and Weems. *Topics in Parallel and \Distributed Computing: Enhancing the Undergraduate Curriculum: Performance, Concurrency, and Programming on Modern Platforms, Springer International Publishing, 2018, ISBN: 978-3-319-93108-1*

[2] http://grid.cs.gsu.edu/~tcpp/curriculum/?q=cedr_ book.

[3] https://tcpp.cs.gsu.edu/curriculum/?q=courseware_management

ACKNOWLEDGEMENT: The work presented here is supported by the NSF/TCPP Curriculum Early Adopter Award and SIIP, UIUC