Abstract— Following a successful introduction of the first 100% hands-on course “ECE 3730 Principles of Embedded Systems Design” offered in the Department of Electrical Engineering in the winter term of the 2012-2013 year [1], a similar 100% hands-on course was implemented in the fall term of the 2012/2013 year in the Department of Computer Engineering at the University of Manitoba. Similar with the previous course, this course “ECE 3740 Systems Engineering Principles” was designed specifically to directly assess student performance particularly in the CEAB attributes of Design, Investigation, Problem Analysis, and Tools. Differently than the previous course, this course used the Digilent MX7cK microcontroller board [2] to design, develop, and implement a self-configuring clustered wireless sensor network (WSN). The assessment included in-depth direct evaluation of student performance in 5 sub-projects, consisting of three evaluative components and hands-on midterm test and final exam, which were performed in the laboratory in real-time.

Keywords—design, UML modeling; multitasking; and C, HTML, and AJAX code development, design thinking, project based learning, CEAB attributes.

I. INTRODUCTION

Since 2010 the Canadian Engineering Accreditation Board (CEAB) [1] has instituted a transitional and development period for new accreditation criteria, part of which requires undergraduate programs in Canadian Engineering educational institutions to develop methodologies that demonstrate that their graduates possess the specified 12 CEAB attributes:

1. A knowledge base for engineering
2. Problem analysis
3. Investigation (Validation)
4. Design
5. Use of engineering tools
6. Individual and teamwork
7. Communication skills
8. Professionalism
9. Impact of engineering on society and the environment
10. Ethics and equity
11. Economics and project management
12. Life-long learning

The transitional period ends in 2015, at which time the CEAB will require full compliance with the new accreditation criteria.

In response to this new CEAB attribute based assessment mandate, and taking advantage of the opportunity, the author re-designed the ECE 3740 Systems Engineering Principles course in the fall of 2012. The course was re-focused to provide students with hands-on design and development experience and to provide a method that directly trains and assesses student performance, particularly in the CEAB attributes of Design, Investigation, Problem Analysis, and Tools.

II. MODEL OF COURSE DESIGN

Previously, the course was taught using the lecture style approach, which consisted of 32 50-minute lectures and five 3-hour laboratory sessions. Among the course objectives was to teach students the following computer systems engineering principles: divide and conquer; incremental design; minimizing complexity; maximizing cohesion; abstraction; designing for reuse and reusing existing designs; and unit and system test plans and procedures. The course met with limited success because students were not given sufficient opportunity to experience these concepts due to the style and method by which the material was presented to students. The laboratories were hands-on, but they were limited to 3-hours and no complex project could be done in that amount of time.

To overcome this limitation, in the fall semester of 2012, a project-based learning (PBL) and 100% hands-on learning approach was implemented in the course “ECE 3740 Systems Engineering Principles” in the Department of
Electrical and Computer Engineering, University of Manitoba.

**Systems Design**

A large and complex project was chosen as the single project which students would work on throughout the entire course. The instructor broke down the project into 5 sub-projects, each of which consisted of three parts. Splitting the course project into smaller parts demonstrated to students how to practise and implement the principles of divide and conquer, incremental development, unit testing, minimizing complexity, and maximizing cohesion. The instructor guided students in how to think about solving a complex design problem, and how to implement a design process. These principles were not taught to students through the lecture style of teaching; rather, students experienced these concepts on their own during the design process. A small company design process was simulated, where the instructor played the role of the systems engineer and team leader, and the students played the role of engineers and team members. However, lectures were given to guide, strengthen, and reinforce the concepts, at the same time students were experiencing these principles. Since students designed, developed, and implemented the sub-projects, they were intimately involved in the procedure; thus, students more deeply understood how and why the project was broken down into smaller pieces, and the rational for doing so. For instance, the sub-projects demonstrated how a complex project may be split into cohesive parts, where each part contained only parts which belonged together, and the rest were kept out. Furthermore, the sub-projects were selected to minimize the associations with other parts, thus minimizing coupling between other parts. As such, the splitting of the project in this way demonstrated to students how each part may be reused in other completely different scenarios and projects. Anecdotal observations showed a higher level of learning, participation, and retention in students. It was noted that during the reinforcing lectures, students were more involved in class discussions; the lectures were transformed into peer-to-peer style discussions and debates about design alternatives and other design principles.

**Tools Usage**

By implementing a 100% hands-on approach to teaching, the CEAB attribute of Tools was more easily assessed. Each student was given the required hardware and software to implement the course project to take with them for the duration of the course. The hardware was of limited size so that it could fit in a lunchbox type of container, and students carried their lunchboxes like they would textbooks to class, labs, and wherever. In this way, students were able to work on their projects whenever they had spare time, either commuting to the University, during lunch, during class, in the library or study halls, or at home. Throughout the entire term, students were required to demonstrate, in person to the team leader (i.e., instructor) - in a hands-on fashion - their design, analysis, investigation, and tools usage abilities, using the supplied computer hardware and software. The instructor noted that there can be no better way of measuring students’ ability to use computer systems tools, than by requiring students to directly demonstrate this ability not only once, but during the entire term. In this way, their developmental improvement was witnessed first-hand as well.

**Problem Analysis**

For each of the sub-projects students practised an iterative development procedure, which consisted of domain analysis; requirements gathering, review and analysis; modeling, translation and coding; unit and system testing; deployment, and management. Short, high-level descriptions of the sub-projects to be solved were created and intended to be incomplete, ill defined, and sometimes ambiguous to emulate real-life initial project descriptions. From these descriptions, students were required to uncover what was actually being requested by following the given systems engineering process. First, students needed to study the domain of the problem in order to become more familiar with the topic. Once a reasonable understanding of the domain was achieved, students then gathered additional problem definition statements from the stakeholders, which were sourced by the instructor and any other pertinent information source. Students reviewed and analyzed the statements to form them into proper statements of requirements. By experience (i.e., learning via the hard way) students realized the statements should be formatted as unambiguous, verifiable, identifiable, safe, consistent, and realistic. Throughout the domain analysis and requirements phases of the iterative procedure, students used appropriate knowledge and skills to identify, formulate, analyze, and solve the complex engineering problems of the sub-projects and the complete project. Students practiced this iterative systems engineering process throughout the course and project development - in the labs assignments, tests, and final exam.

**Investigation**

Students practiced investigation through the five projects, one midterm, and the final exam. The test and final exam were completely hands-on, and students needed to determine some sort of debugging method to get their code working properly. Many students discovered the use of breakpoints in the software. The instructor demonstrated several times in person to each student how to logically approach a problem. Many times students would consult with the instructor regarding a problem with their project by saying, “well, it doesn’t work.” The instructor guided the students by trying to get them to start thinking about locating a strategic point in the software to place a breakpoint.
“Carefully analyze the symptoms of the problem, determine what the program is supposed to be doing at each point in your program, and then determined a strategic point to place a breakpoint,” the instructor would say to the students. “When the program stops at your breakpoint, analyze the contents of the registers and memory locations to determine if they contain the expected values,” the instructor went on to say. “By stepping through each instruction following the breakpoint, you will then determine at which instruction the problem begins to manifest,” the instructor finalized. By guiding students in person in this way, and going through the debugging process with them in person and in real-time, students really understood the procedure, and were able to solve future problems by themselves in this manner.

III. IMPLEMENTATION OF COURSE DESIGN

A wireless sensor network (WSN) for gathering sensor readings and reporting the data to a central base station on the Internet was chosen as the course project. This project was a more elaborate version of the project suggested by Microchip’s TCP/IP Stack [2], a real-time vending machine monitoring system (people can monitor the inventory of a vending machine using a web browser anywhere on the Internet). However, this project may also be used in general sensing, monitoring, and control applications, such as real-time smart building monitoring and control, lake water quality monitoring, water content for flood prediction in the Red River Basin, shopping for dummies, and forest fire monitoring and control.

Digilent’s MX7cK board was chosen to implement each node in the WSN. Features of the MX7cK include an 80 MHz 32-bit microcontroller, 10/100 Ethernet controller, USB controller, CAN controllers, SPI interface, I2C interface. Pmod expansion connectors allowed additional components, such as microphone, gyroscope, accelerometer, temperature, WiFi transceiver, and real-time clock (Fig. 1). The board may be powered and debugged through a single USB cable and USB connector on the board. This proved to be very convenient for students as they could carry the entire hardware in the given Mx7cK package.

The Mx7cK board does not come with a TCP/IP stack. Microchip’s TCP/IP stack [2] was adapted to work on the MX7cK board [3]. The adaptation included many software and configuration modifications to make the TCP/IP stack compatible with the Mx7cK board [4]. Furthermore, since the TCP/IP stack only supports an SPI EEPROM, extensive code porting was required to make the use the I2C EEPROM on the Mx7cK board with the TCP/IP stack.

A sequence of five projects incrementally developed a client-server based system, which was autonomously configured in two phases. In “Phase 1 Clustering Process” sensor nodes autonomously and automatically learned the identification of a server node (cluster head), joined the cluster, and became cluster members of the cluster.head’s cluster of nodes. After the clustering process phase, the cluster head broadcasted to the cluster members a TDMA schedule, which informed the cluster members their data communications slot. In “Phase 2 Data Communications” cluster members, periodically, according to their TDMA slot, reported their sensor readings to the cluster head. The cluster heads aggregated their data (i.e., computed averages), and sent the data to the gateway node (Master Node), which was connected to the Internet through a wired connection. The Master Node on demand relayed the aggregated sensor data to sink, which was implemented by a web page on a remote host on the Internet. The web page used AJAX to continuously update the web page according to a user’s demand. In addition to data communications between sensor nodes and the web page, the system also allowed the user to specify and configure sensor node parameters remotely through the web page and AJAX. More details of the course project may be obtained from [4] and [5].

The final exam directly assessed student performance in the CEAB attributes of Design, Problem Analysis, Investigation, and Tools. The exam was 100% “hands-on,” and it was performed entirely in the laboratory. The placement of the workstations in the lab was done in a way to allow the invigilator to easily monitor and observer the performance of each student without being a distraction. Each student was assigned a development workstation, which had supporting software on it. Students brought their MX7cK microcontroller board, with Ethernet cable, to the exam, and used it to solve the given problem. A preparatory

![Fig 1 Digilent Mx7cK microcontroller board, with various sensors: microphone, gyroscope, accelerometer, temperature, WiFi transceiver, and real-time clock.](image)
question on the exam asked students to demonstrate that "ping" was working on their system. This required students to make modifications to the TCP/IP stack software on the MX7cK board, configure the TCP/IP stack on their workstation appropriately, and to make the necessary connections between the MX7cK and the workstation. Students raised their hands when they were ready to demonstrate the ping functionality, and the invigilator examined their solution in real-time.

IV. SUMMARY

This paper reports on the implementation of a project-based-learning and 100% hands-on approach to teaching ECE Systems Engineering Principles at the University of Manitoba. The objective of the course was to implement a wireless sensor network for gathering sensor readings at a remote location, and reporting the data to a central base station on the Internet. The course was designed specifically to provide a direct method of assessing student performance, particularly in the CEAB attributes of Design, Investigation, Problem Analysis, and Tools. Throughout the entire term, students demonstrated, in person - in a hands-on fashion - their design, analysis, investigation, and tools usage abilities, using the supplied computer hardware and software, which was given to each student at the start of the term to carry with them and take home for the duration of the course. Five sub-projects were designed, each of which contained three parts/deliverables. Each sub-project was designed to impart experience to students and to practise embedded systems development through “divide and conquer,” incremental development, software and hardware reuse, maximizing cohesion, and minimizing complexity. The test and final exam were performed by students entirely in the laboratory; they brought their embedded systems hardware, solved the given hands-on problems of the test/exam, and demonstrated their solutions, in real-time.

This novel methodology allowed the examiner to directly assess student performance in the CEAB attributes of Design, Analysis, Investigation, and Tools, because their designs and solutions were actually demonstrated in actual hardware and software, not just on paper, like the conventional approach to student tests/exams.

V. ACKNOWLEDGEMENTS

The author acknowledges the Department of Electrical and Computer Engineering for funding the hardware for ECE 3740, as well as supporting the project-based-learning and hands-on approach to teaching computer engineering courses in the Department of Electrical and Computer Engineering, University of Manitoba.

REFERENCES