AC 2008-363: REAL-TIME, EMBEDDED-SYSTEMS NETWORKING: A NOVEL WAY TO DEVELOP AN INTERACTIVE UNDERGRADUATE COURSE

Ece Yaprak, Wayne State University

Dr. Ece Yaprak is a Professor of Engineering Technology in the College of Engineering at Wayne State University. Her areas of interest include computer networks and communications where she has published extensively. She has held engineering positions at General Electric and Ford Motor Company, and research fellowships at NASA (John Glenn, Jet Propulsion Laboratory, Ames Research Center, and the Johnson Space Center) and the U.S. Navy (SPAWAR). She has received teaching excellence awards from her Division and the College of Engineering. She has received funding for her research from the NSF, the US Navy, NASA, and the business community. She is currently Vice-Chair of the Committee on Technology Accreditation Activities (CTAA) at IEEE, and is an active program evaluator for ABET.

Karen Tonso, Wayne State University

Dr. Karen L. Tonso, an Associate Professor of Social Foundations in the College of Education at Wayne State University, uses approaches common to cultural anthropology to study the structure of schooling, especially everyday life in learning communities. She brings to bear 15 years of experiences as a reservoir engineer in the petroleum industry to her research in engineering education. Her research in engineering education received the 1999 Mary Catherine Ellwein Outstanding Dissertation Award (Qualitative Research Methodology) and the 1998 Selma Greenberg Distinguished Dissertation Award (Research on Women and Education) from the American Educational Research Association. She received the 2006 Distinguished Paper Award for her article "Engineer Identity" from Cultural Studies of Education. She is currently the PI of ESCALATE: Engineering & Science Careers in Academia, Learning from ADVANCE and Translating Effectively, an NSF ADVANCE-PAID Project. Address: #319 Education, 5425 Gullen Mall, Wayne State University, Detroit, MI 48202, 313-577-1764 (v), 313-577-5235 (f), ag7246@wayne.edu.

Real-Time, Embedded-Systems Networking:
A Novel Way to Develop an Interactive Undergraduate Course

INTRODUCTION

During the last century, discoveries in the sciences and engineering aided the creation of increasingly wider bases for new scientific breakthroughs, facilitated particularly during the last few decades by advances in information technologies. These developments impact higher education and policy-making in at least two ways: globalization of knowledge and rapid change in understandings. Globalization of knowledge resulted in a *flat world* where knowledge is now available everywhere, at any time, and at lower cost. And, to stay competitive in such a flat world, nations are recognizing the importance of continuously creating knowledge to ensure industries are more robust, more agile, and much more responsive to people's needs. Shifting toward the future requires joining the transformation of the world economy from computer-based to internet-based platforms; more quickly understanding the significance of all-world, around-the-clock supply chains in manufacturing; and adapting to modes of business involvement of this decade, such as outsourcing, open sourcing, off-shoring, and in-sourcing[1].

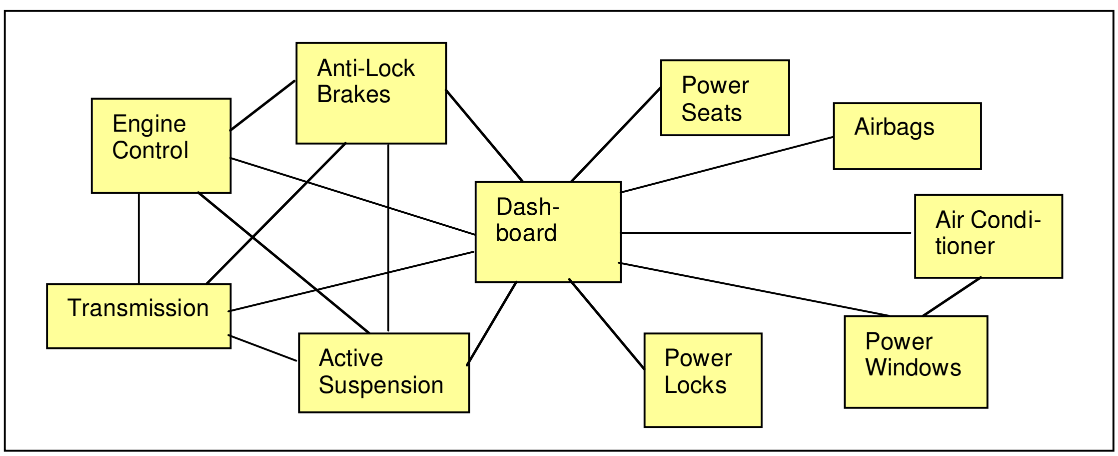
Globalization places a great burden on STEM higher education programs to keep up-to-date in scientific knowledge and to invest in making new discoveries to prepare students as they become more informed individuals and learned professionals[2]. Upgrading science and engineering education to prepare students for today's *flat world* is particularly urgent in U.S. manufacturing regions, where America is faced, at least in manufacturing belt states like Michigan, Pennsylvania, and Ohio, with rapid implosion of manufacturing industries, dramatic increases in unemployment, and deteriorating state economies. Here, manufacturing capabilities are giving way to facile and nimble competitors from abroad and technological products for citizens must now balance social and environmental needs, suggesting that static or unresponsive curricula risk producing students ill-prepared for the work-lives of future decades. But, the demands of changing a course can be great for individual engineering faculty and this likely prevents, or slows, needed innovation in engineering courses. We describe a way to build a new problem-based, laboratory-centered course, among the more difficult to produce, but first begin with a description of the kind of course we have in mind.

REAL-TIME, EMBEDDED-SYSTEMS NETWORKING

Real-time, embedded-systems (RT-ES) networking suggests an area where many undergraduate students would benefit from a systematic educational experience. The number of electronic systems embedded into automobiles, industrial systems, factory automation, machine control and medical systems, among others, has increased dramatically in the last few decades[3-5]. As these systems increasingly interconnect collections of distributed processors via *real-time* networks, better understanding the nature and the functioning of these networks will become critical. Also, rapid advances in computing hardware now make embedding these capabilities in manufactured devices common, especially where rapid communication is needed. Because of the ubiquitous nature of real-time, embedded-systems networked into a wide range of devices, any recent engineering graduate working in a wide range of technological development, production, and research activities could reasonably expect to work with, design, or manufacture devices that employ real-time, embedded-systems networking. But, even as the demand for a workforce skilled in networking areas such as in-vehicle networking, factory automation, and real-time distributed control systems in industrial, medical, and other applications is increasing, many schools teach courses only in embedded systems, with only a few explicitly teaching about embedded networking, especially at the undergraduate level. Thus, the U.S. is falling behind countries like Germany where embedded-system curricula with emphasis in embedded networking are widely implemented (personal communication, Holger Zeltwanger - CEO, CAN in Automation, 2006).

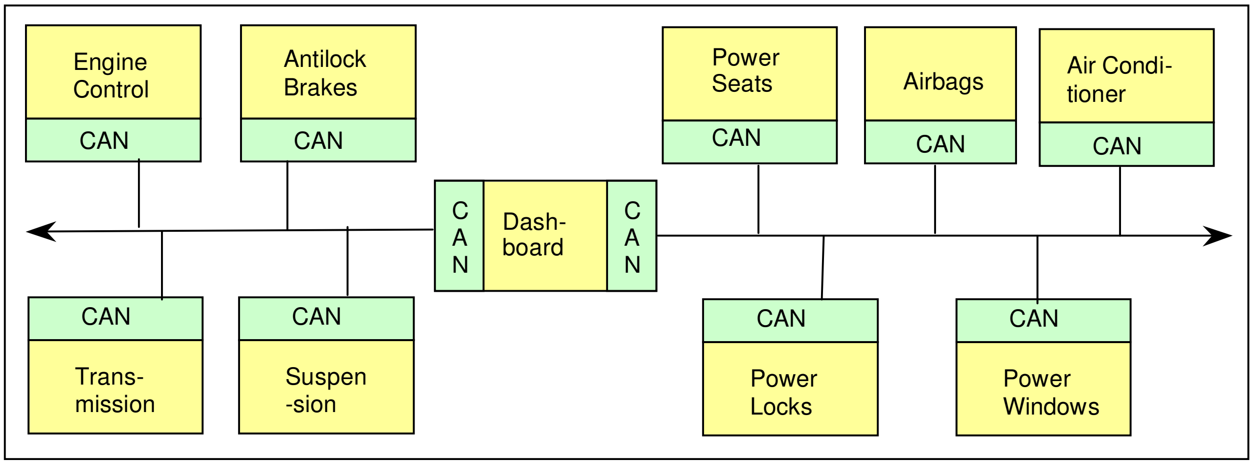
Embedded-systems networking involves bonding various systems in a meaningful configuration allowing effective communicate among them. While systems and networking are not new, embedded systems have grown in significance with increasing complexity built into products and services today. Until recently, these devices communicated with one another via point-to-point wiring, an inefficient means of communication, with too many connections. (Figure 1 shows an example of an outdated point-to-point embedded system from the automobile industry where devices like anti-lock brakes, power seats, and power windows communicate with one another.)

Figure 1. Automobile communication using point-to-point wiring (microcontroller.com)



As embedded networks became much more complex during the last couple of decades, another mode of communication in systems emerged, the bus system. As seen in Figure 2, this mode allowed replacing point-to-point systems with a serial bus (the connecting lines) linked to all control systems. This was accomplished typically by adding controller area network (CAN) hardware to each control unit to provide the protocal (or rules) for transmitting and receiving information via the bus. As an *event-triggered* protocol, CAN was ideal for situations where microcontrollers communicate either with each other or with remote peripherals. The CAN bus, then, established a network among microcontrollers. It was well suited for high-speed applications using short messages. Its robustness and reliability made it suitable for the semiconductor industry. Changing the number of nodes dynamically without disturbing communication within, and among, other nodes became a great advantage of CAN technology.

Figure 2. Automobile communication using Controller Area Network (microcontroller.com)



Today, such devices are everywhere. Medical-device manufacturers have implemented CAN-based networks for a number of devices, such as injection-scanner machines, radiation dosimeters, and collimators, and they also exist in patient beds, X-ray machines, operating-room function, and hospital control systems with voltage control, indication and control units, digital I/O, and visualization software. FDA embraces open network standards and works cooperatively with medical device manufacturers[4]. Some universities, such as Johns Hopkins, are researching uses of CAN-controlled surgical robots[4]. CAN is also used as an embedded network in other applications. For instance, CAN is used in motion-control oriented industrial machine control to link single devices (e.g. I/O modules) and sub-systems—such as in textile, printing, injection molding, and packaging machines. CAN-based networks are also used in non-industrial machine control, such as CANopen, which is used in money-changing and vending machines.

Clearly, RT-ES networking is a very fluid set of technological ideas that differs dramatically from the bounded formal knowledge common to many engineering courses; there is, however, no practical way to teach about everything. This suggests that, among central learning goals, students need not only to learn about the capabilities of different technologies, but also to *learn how to learn*, that is they must develop skills for troubleshooting, for researching new ideas using on-line and library resources, and for learning about software/hardware capabilities, as well as learn to find and use source materials explaining emerging technologies. Ultimately, these learning needs drove the shape of the course, while logistics of developing laboratory exercises influenced what material was included.

Though the research project began with a focus on what, and how, students learned, it quickly became apparent that studying how the course came into being would offer strategies to other faculty who develop new courses, and for some that these might be innovative approaches that would ease course development. Thus, the purpose of the research reported here is to chronicle the development of a course in real-time, embedded-systems networking. As such, this paper not only offers a strategy for developing courses, but also suggests what such a course might incorporate.

RESEARCH METHODS

The research process followed ethnographic traditions commonly used by cultural anthropologists: enter the field being studied and become an observer of everyday activity there, capture the activity, who participates in it, and where it takes place through field notes taken over the course of the activity studied; converse with insiders to the activity (via informal interviews) about how they interpret what is happening and why things are done in the way observed; and then deepen understandings of the events observed, people participating, and setting through formal interviews[6-8].

As part of the larger data set about the course itself, the ethnographer (Tonso) observed Yaprak during the course development activities, as well as conversed with her on a regular basis about her efforts, and interviewed her on two occasions about the course development. Findings here are drawn from this subset of the data.

This methodology was well suited to the research purpose, because the course itself was not a fait accompli that must simply be put into action, but emerged through conversations between the faculty and graduate students, and in thinking about how to fit the course to the context. By early May of 2007 (only four months before the course would be offered for the first time), it became evident that developing laboratory experiments would be a more complicated undertaking than had earlier been anticipated. Thus, just as the spring-summer session began, Yaprak realized that writing all of the lab modules would be impossible. Solving this dilemma, then, became a central innovation of developing this course.

RESULTS

The literature on developing and revising engineering courses often depends on faculty's anecdotal descriptions[9-15] or on faculty suggesting how to match curricula to emerging policy[16]. These articles generally describe the course content with little attention paid to the process of developing such courses. While these are no doubt helpful for others trying to develop similar courses, what we report here comes from systematically collected data that included information about how a faculty member incorporated advice about developing courses, selected labs for the course, and got those labs ready for student use, a research approach we hope to see more often.

Course development funding arrived in early 2007. Because this left too little time to work through bureaucratic approvals for an undergraduate course before fall, a special-topics, graduate-level course could be offered in the spring-summer session (lasting seven weeks from roughly early May through late June), with the first undergraduate course held in fall 2007. This four-semester-hour undergraduate course would meet from three to five in the afternoon on Mondays and Wednesdays. Yaprak planned to hold lecture sessions on some Mondays, with lab time on Wednesdays and Mondays when no lecture was scheduled. Thus, the spring/summer course would allow graduate students to pursue individualized study and the fall course would teach about a wider range of topics to all students.

During the winter term, faculty began to meet on a biweekly basis and one of Mahmud's graduate student agreed to help with lab development as an independent study course. He expressed interest in working with controller area networks (CAN), but for a variety of reasons (software/hardware mismatches, other coursework, a need to write code for some portions) his progress was slow. Faculty debated to what extent CAN labs would be appropriate for an undergraduate course, while the graduate student assistant plugged away to develop a set of integrated labs demonstrating key aspects of CAN's. Ultimately, these labs, completed just as the fall 2007 semester began, would not play a part in the undergraduate course.

Both the special-topics and the undergraduate courses were taught in a learning lab equipped with nine internet-connected computer workstations, though there were only four sets of equipment for RT-ES networking. Figure 3 illustrates that (for the undergraduate course) board, debugger, and power supply sets were affixed to a wooden board in the department shop, which eased handling, and improved equipment storage and security. In addition, the screen of the PC illustrates that the step-by-step procedure provided for students incorporated photos of devices that could be clearly labeled and referenced.

Figure 3. Laboratory configuration

The special-topics master's course allowed students to engage in independent research into a personally meaningful topic drawn from RT-ES networking, but it also attracted two undergraduate students. Yaprak knew that understanding the selected microprocessor's capabilities would be central to making headway in such a short time period. Thus, she began the spring session with all students learning about the microprocessor. A "demo kit" was used to develop the special-topics graduate session and only enough copies ordered to cover course development. But after developing a lab guide using this demo kit, when attempting to order enough for the undergraduate lab, the rapid advance of such technologies became apparent. The original demo kit was not available and required ordering an updated, but different kit. This meant that the learning slides had to be re-done for the new software and hardware. This event reinforced that students must not only develop deep understandings of the software-hardware interactions, but also learn ways to find out about new ones, not just come away with point-and-click cookbook techniques.

Students in the special-topics course then worked in pairs to complete a project about a selected RT-ES-network topic: ethernet connectivity, Zigbee (a wireless protocol), or controller area networks (CAN). Here, graduate student teams existed in the lab side-by-side with an undergraduate team. Not surprisingly, this produced a fertile ground for the exchange of ideas. Each of the teams not only had to research their particular topic using library and on-line resources, but also had to use supplied user manuals to develop understandings of the capabilities of the particular devices purchased for the lab. As part of their project, students developed step-by-step lab exercises, wrote detailed explanations of the technology and its applications, and provided a technical report drawn from the on-line and library literature on such technologies. Thus, the special-topics course provided a rich field of activity where students could explore the technologies and become something of an expert in one area. Some Master's students from this course continued their work by expanding it into a Master's thesis.

From the special-topics students' step-by-step lab exercises, Yaprak developed a set of lab exercises for the first offering of the undergraduate course. Here, students who had taken advanced digital design and an introductory microprocessor course would develop expertise in four central technological arenas: microprocessors, network connectivity and protocols, Zigbee, and controller-area networks (YaprakInterview, 5-20-07). This required not only situating each technology in a general course plan, but also revising students' step-by-step labs to teach specific concepts upon which students could build in subsequent courses and work experiences. The two undergraduate students who took the special-topics course agreed to serve as undergraduate teaching assistants for the fall course. Thus, for each lab project, Yaprak provided step-by-step exercise guides for conducting the lab, and required students to explain what they had found. For instance, as she explained during an interview:

> [This will be like what they have done in another class where they] start capturing anything [that] goes through their network. This package captures everything, and they send each other messages and they stop capturing, but even that within this one minute, they have maybe 500 packages captured. The rest they do themselves. They go through whatever they have captured or filtered and find a package that they either sent or received. After that, they can actually see the package in hexadecimal format and they print that for me. Then I ask them, why don't you look at the actual package, decode that for me. Tell me this is my address, this is the sender's address, this is the data I sent, this is the frame-check sequence, and so on, They actually capture the package they sent and decode it. Things that they learn in class [lecture] they can actually see it.

As will become clearer in a subsequent paper devoted to student learning activities, having students with mixed expertise proved crucial to developing a learning setting where students could serve as knowledgeable advisors for one another. In fact, some students, who had not taken the special topics course, arrived with substantially enhanced system-maintenance skills learned on the job and this made overcoming complex roadblocks possible.

Developing the laboratory set-up for this particular course came with a certain amount of difficulty. (Table 1 indicates the equipment needed for each laboratory station.) Not only did campus purchasing rules and regulations slow down arrival of needed equipment, but minor changes to the equipment or its software complicated matters. In fact, though boards and debuggers required cords to connect to power sources and to the PC, in some cases these cords did not come with the equipment, nor was this made apparent in the ordering process. Thus, additional time was lost. At its best, it took twice as long to receive equipment as expected, and at its worst months of wrangling could pass before needed materials were in hand. However,

Table 1. Equipment needed for each station

Lab Title	Description	Equipment	Duration
1. INTRODUCTION TO MPLAB ICD2 AND PICDEM 2 PLUS — ***A/D Converter***	This exercise is an implementation of the PIC18F452 analog-to-digital converter using the PICDEM 2 Plus Demo Board. The program configures the A/D module to convert input from A/D channel 0 (connected to the potentiometer on the demo board) and display the result on the four PORTB LEDs.	• **DV164006** MPLAB ICD2 In-Circuit Debugger and Demo Board	2-weeks
2. INTRODUCTION TO MPLAB ICD2 AND PICDEM 2 PLUS — ***LCD Lab***	This laboratory exercise will help students run a sample program which uses the LCD panel on the PICDEM 2 Plus Demo Board. Students will need to make some adjustments on the program so that R4 and R0 buttons can read the menu of the display and choose an option to view the voltmeter, buzzer, temperature and the clock.	• **DV164006** MPLAB ICD2 In-Circuit Debugger and Demo Board	1-week
3. INTRODUCTION TO MPLAB ICD2 AND PICDEM 2 PLUS — ***Scrolling LCD***	This laboratory exercise is an extension of the previous laboratory exercise. Students are asked to modify the program(s) so that they can write the words they wish to scroll on the LCD.	• **DV164006** MPLAB ICD2 In-Circuit Debugger and Demo Board	1-week
4. INTRODUCTION TO THE PICDEM NET2 DEVELOPMENT ENVIRONMENT_1	This laboratory exercise allows students to develop Internet connectivity applications over an Ethernet connection using embedded Microchip controllers over Ethernet and the Internet. Students will get familiar with the Microchip TCP/IP (Transmission Control Protocol/Internet Protocol) Stack architecture. They will learn to work with IP addresses and will access a website that came with the demo program.	• **DV164006** MPLAB ICD2 In-Circuit Debugger and Demo Board • **DM163024** PICDEM 2 Ethernet Development Board	2-weeks
5. INTRODUCTION TO THE PICDEM NET2 DEVELOPMENT ENVIRONMENT_2	This is a continuation of the previous exercise where the students are asked to change a website and/or create their own website.	• **DV164006** MPLAB ICD2 In-Circuit Debugger and Demo Board • **DM163024** PICDEM 2 Ethernet Development Board	1-week
6. INTRODUCTION TO PICDEM Z DEVELOPMENT ENVIRONMENT	This laboratory will demonstrate a Zigbee wireless network setup. A major component involved is the Microchip Zigbee Stack used to carryout software protocol handshaking. The Zena Wireless Network Analyzer will be used as a network traffic monitor.	• **DV164006** MPLAB ICD2 In-Circuit Debugger and Demo Board • **DM163027** PICDEM-Z 2.4 GHz Demo Kit • **DM183023** Zena Network Analyzer	2-weeks
7. CONTROLLER AREA NETWORK (CAN) LABORATORY_1 — ***Understanding CAN Protocol and Frames Using CAN-LIN1 Evaluation Board***	This laboratory exercise allows students to see how the CAN-LIN1 board operates and how the software (CANKingdom) is used to control and process CAN message. Students are able to see and understand CAN messages and locate the identifier, data length, and the value of the data injected onto the bus using the output window of the software.	• **DV164006** MPLAB ICD2 In-Circuit Debugger and Demo Board • **DM163007** CAN-LIN 1 Demo Board	3-weeks

after about three months, the lab was set up and operable. Thus, unlike the graduate-level course, because of time constraints, the undergraduate course did not require that students wade through equipment user manuals to ascertain functionality, but were guided systematically through using each board.

CONCLUSIONS

Developing this undergraduate course in real-time, embedded-systems networking followed the advice of earlier scholars. First, earlier scholars suggest ways that industry and academics can work together to produce more industry-savvy graduates[17-18]. In our case, using a consistent suite of equipment from Microchip allowed students to gauge how a suite of hardware fit together, but also how even minor manufacturer updates might affect functionality. These experiences suggested to students some of the dilemmas that might occur when vendor products are mixed. Also, the importance of engaging students in working with ideas and issues that are meaningful to them is undisputed[19-20]. This is especially appropriate in a course where only a small subset of information about a given topic can be taught.

Likewise, organizing the RT-ES course to require student collaboration made it possible for the faculty member to become less of an active dispenser of accumulated wisdom, and more of a facilitator of learning. This sort of learning setting inculcates in students a deeper appreciation for their responsibilities for their own learning, which is especially important when working with the kinds of technologies that change rapidly and can be deployed in a wide range of applications. Furthermore, giving students opportunities to perform library and on-line research (and write research papers) about engineering topics, topics which differ dramatically from those usually covered in English composition classes, allows students to appreciate how to meld laboratory research with library/on-line research. In fact, as suggested by earlier scholars[21], the RT-ES networking course provided robust opportunities to perform research more attuned to the kinds of fact-finding and systematic trial-and-error testing associated with engineering work. Ultimately, the organization of the class took full advantage of the possibilities associated with undergraduate research.

Finally, the course took to heart what others have written about incorporating problem-based laboratory activities so central to the “doing” of engineering[22-24]. Rather than the laboratory exercises being simply walking down through a series of steps guaranteed to work, they became a beginning point for understanding what happened and why. In this course, undergraduate research, problem-based laboratory exercises, and group work were inextricably intertwined and taken together would prove central to student learning. (The topic of a subsequent paper.)

This paper suggests that performing research about course development is a worthwhile activity. Even when a faculty member’s time is short and the demands of a new course are great, we found that using the equipment and a set of ideas for a special-topics graduate course is a viable way to develop complex laboratory exercises for a subsequent undergraduate course. In fact, guided study by undergraduates, master’s projects, or portions of doctoral dissertations could also provide sources for laboratory materials, so long as the consent of the students who produce the materials is gained and these students are given credit for their contributions in coursework materials. And, though this proved the case for a course in real-time, embedded-systems networking, there is no reason to believe that this is limited to this kind of course.

ACKNOWLEDGEMENTS

The authors acknowledge the support of the Division of Undergraduate Education of the NSF (NSF Award No. DUE 0632890) and the Microchip Technology Inc. for monetary and in-kind support, and Dr. Syed M. Mahmud, Associate Professor of Electrical and Computer Engineering (our co-PI) and his graduate student for discussions about the undergraduate course. Above all, we are most grateful for the participation of Wayne State University students, whose hard work in the RT-ES networking course, and insightful conversations about their work, motivate us to continue to improve engineering education.

REFERENCES

[1] Friedman, T. L. (2005). *The world is flat*. New York: Farrar, Strauss, and Giroux Publishers.

[2] National Science Foundation (2001). *Greater expectations initiative.* Arlington, VA: National Science Foundation.

[3] Dumont, F.C., Suter, J.J., & Schwartz, P.D. (1999). A controller area network-based telemetry and command system for small space experiments, *Johns Hopkins APL Technical Digest*, *20*(2), 6 pp. Downloaded 3-23-06 from http://techdigest.jhuapl.edu/td2002/dumont.pdf.

[4] Menon, C.J. (2005). Medical device network standards. *Medical Electronics Manufacturing*. Downloaded 3-23-06 from http://www.devicelink.com/mem/archive/05/10/004.html.

[5] Rover, D.T., Cheng, B., Wey, C.-L., & Mutka, M.W. (2000). Incorporating large-scale projects into a multidisciplinary approach to embedded systems. *Proceedings of the 2000 International Conference on Engineering Education (ICEE 2000),* Taipei, Taiwan, pp. 105-108. Downloaded 3-26-06 from http://www.ineer.org/Events/ICEE2000/Proceedings/papers/WC1-3.pdf.

[6] Spradley, J.P. (1979). *The ethnographic interview*. New York: Holt, Rinehart and Winston.

[7] Spradley, J.P. (1980). *Participant observation*. New York: Holt, Rinehart and Winston.

[8] LeCompte, M.D. & Schensul, J.J. (1999). *Designing and conducting ethnographic research*. Altamire Press.

[9] Dong, Y, D., El-Sayed, J. (2007). Mechanics, process, and design simulation of fiber-reinforced composite materials — A new course development. *Proceedings* of the 2007 American Society for Engineering Education Annual Conference and Exposition, Session AC 2007-1005.

[10] Furth, P, M., Johnson, E. E. (1997). Developing and introductory electrical and computer engineering course. *Proceedings* of the 1997 American Society for Engineering Education Gulf-Southwest Annual Conference, Houston, TX. 367-370.

[11] Hsu, T. (1999). Development of an undergraduate curriculum in mechatronics systems engineering. *Journal of Engineering Education,* 173-179.

[12] Miller, R. L., Olds, B. M. (1994). A model curriculum for a capstone course in multidisciplinary engineering

design. *Journal of Engineering Education,* 1-6.

[13] Porter, R. L., Fuller, H. (1998). A new “contact-based” first year engineering course. *Journal of Engineering Education,* 399-404.

[14] Yokomoto, C. F., Rizkalla, M. E., O’Loughlin, C. L., El-Sharkawy, M. A., Lamm, N. P. (1999). Developing a motivational freshman course in using the principle of attached learning. *Journal of Engineering Education,* 99-106.

[15] Zhou, X., Everly, J., Rabiee, M. (2007). A wireless communications and networking course developed for electrical engineering and computer engineering technology programs. *Proceedings* of the 2007 American Society for Engineering Education Annual Conference and Exposition, Session AC 2007-2079.

[16] Felder, R. M., Brent, R. (2003). Designing and teaching courses to satisfy the ABET engineering criteria. *Journal of Engineering Education,* 7-25.

[17] Lakshmi, M., Subach, J., Magrane, J., Popovich, C. (2007). Industry-academia collaboration. *Proceedings* of the 2007 American Society for Engineering Education Annual Conference and Exposition, Session AC 2007-2148.

[18] Williams, D., Butera, R., Uluagac, S., Clark, M., Deck, A., Torba, M., Trahan, S. (2007). Cooperative university/industry development of a freshman ‘introduction to ece design’ course. *Proceedings* of the 2007 American Society for Engineering Education Annual Conference and Exposition, Session AC 2007-1791.

[19] Pickard, J., Lunsford, P., Popoviciu. (2007). IPV6 Course development for information technology curriculums. *Proceedings* of the 2007 American Society for Engineering Education Annual Conference and Exposition, Session AC 2007-1955.

[20] Smith, K. A., Sheppard, S. D., Johnson, D. W., Johnson, R. T. (2005). Pedagogies of engagement: Classroom-based practices. *Journal of Engineering Education,* 87-101.

[21] Thompson, N. S., Alford, E. M., Liao, C., Johnson, R., Matthews, M. A. (2005). Integrating undergraduate research into engineering: A communications approach to holistic education. *Journal of Engineering Education,* 297-307.

[22] Feisel, L. D., Rosa, A. J. (2005). The role of the laboratory in undergraduate engineering education. *Journal of Engineering Education,* 121-130.

[23] Murray, W. R., Garbini, J. L. (1997). Embedded computing in the mechanical engineering curriculum: A course featuring structured laboratory exercises. *Journal of Engineering Education,* 285-290.

[24] Papautsky, I., Bhagat, A. A. (2007). NSF CCLI: A Problem-based microfluidics laboratory course for undergraduates. *Proceedings* of the 2007 American Society for Engineering Education Annual Conference and Exposition, Session AC 2007-2482.