Systems Engineering Framework to Design a Laboratory Course: A Case Study

Prashanth Chetlur Adithya[1], Shraddha Pandey[1], Jose. A. Caballero[1], Özgür Yürür[2] and Wilfrido. A. Moreno[3]

[1]University of South Florida, Tampa, FL, [2]Qurvo, Greensborough, NC, [3]University of South Florida; Corresponding Author; 4202 East Fowler Avenue, ENB 118, Tampa, FL 33620-5350, USA; email: wmoreno@usf.edu.

Abstract

This manuscript introduces a Systems Engineering (SE) framework to design a state-of-the-art engineering laboratory course. In order to illustrate the developed framework, results from an implemented undergraduate embedded systems laboratory course are being presented as a case study. The Concept of Operations was developed for the lab design to bridge the gap between skillset required by the industry and the learning objectives defined by the academic program. This was accomplished by preparing the students with a skillset that would facilitate their smooth transition from the academic program to the industry. Following the SE framework, industry ready skills were identified based on the survey carried out from the TRUE (Taking Responsibility to Understand Engineering) Partners of the Electrical Engineering (EE) Department at the University of South Florida (USF). The course design requirements were then identified based on the needs of the stakeholders: the industry; the students; and the EE department of USF. Further, the course was implemented through CANVAS, a learning management system, by incorporating innovative instructional interventions. The case study is being verified and validated by presenting the results of the course exit survey and metrics such as grade point averages, enrollment, retention and completion rates; the trends of the results show that the students were able to grasp the presented technical material with ease. In addition, the results also indicated that maximum learning experience was accomplished due to the incorporation of instructional interventions like spacing and interleaving, pair programming, online lectures, active and social learning, into the course design. Finally, it can be seen that a SE approach is very effective not only in maximizing the student learning experience but also in meeting the ever-changing skillset requirement of the industry.

Keywords

Experiential learning, engineering education, innovative curriculum, systems engineering, course design, embedded systems.

1. Introduction

Embedded Systems Engineering has evolved rapidly in the last two decades and has become a significant part of human lives. It is the heart of the digital systems market ranging from automotive, telecommunications, wearable devices, medical appliances, industrial machines to aerospace and military applications. By looking at the first prototypes of cell phones and comparing them with today's smartphones, it can be seen how hardware designs and software implementations of these devices have evolved over the time. Embedded Systems Engineering has gained an increasing interest, especially after the advent of the ARM architecture and the corresponding software development tools. The industry has been adapting these technologies virtually into every electronic product to sprinkle a taste of smartness.

In the context of EE, innovation and design of digital electronic devices play a crucial role to fulfill consumer expectations. For this reason, electronic device manufacturers have to maintain the quality of their engineering recruits such that the needs of a consumer are fulfilled while simultaneously being able to create new developments and innovations. The considerable advances in embedded systems engineering requires the industry to maintain a workforce that is well-trained in microprocessors based systems. Moreover, the workforce should not limit their expertise to a specific topic, but also should have a general conceptual idea of the total system architecture and development process from an interdisciplinary perspective. To satisfy this skillset requirement, the industry predominantly relies on universities. Hence, it is necessary that universities design their courses that are in sync with the needs of the industry to provide engineering students with relevant practical knowledge.

Typically, for many academicians, designing a course involves selecting a body of content and then deciding how to distribute it within a set time period. However, the content-first approach to design a course tends to focus on the delivery by the teacher and not the student learning[1]. In the literature, different course design frameworks have been used to improve the learning process. The integrated course design is an example design framework that incorporates processes such as analyzing the situational factors, formulating the learning goals, designing the feedback and assessment procedures, and selecting the teaching/learning activities. However, these frameworks assemble the fundamental components into a relational, integrated model rather than a linear one[2]. Another set of course design frameworks were implemented specifically to satisfy the ABET engineering criteria and have primarily focused on how to assess certain outcomes defined to meet the accreditation requirements by equipping the students with the skills and attitudes specified by the outcomes[3].

To date, most electrical engineering programs continue to teach the general concepts of embedded systems, such as the fundamentals of the computer architecture and the organizational structure of hardware components of earlier microchips[1,4-6]. Other programs focus on using prototyping platforms like Arduino which offer advantages such as low cost, user friendly programmability features and form factor; however it is better suited for accomplishing a task rather than learning the microcontroller's architecture[7,8]. All these approaches are valid in the learning process, however instead of keeping up with the latest developments in the field, they create a gap between the skillset required by industry and the learning objectives defined in academia. As a consequence, the graduating students experience difficulties during the transition from academia to the industry. Therefore, education of embedded systems engineering is a great example of meeting workforce development among the universities and the industry under the presence of mentioned emergent necessities. In this sense, a strong desire has arisen to create a microprocessor based embedded systems laboratory course in electrical engineering programs in order to transform the relevant curriculum by providing a strong practical experience.

Over the past decade, open online courses that have been adopting an SE process for the course design have been very successful. Development of a course is a complex process as it involves interactions of different stakeholders having individual goals. SE processes for online course design have been able to provide a holistic view of these interactions. SE process outlines a systematic philosophy for the course design in a way that the ultimate goals of the individual stakeholders are fulfilled. In this process, the course design is treated as a system and the entire process of design is divided into different phases. Each phase receives inputs from the preceding phase and the process is refined to develop the curriculum[9,10]. This methodology provides the course designer a clear understanding of the constraints, requirements and available resources to create a curriculum which can dynamically track the ever changing requirements of the industry. The design process when using this approach becomes modular and portable, thereby making it efficient to modify as well as reuse for the development of new courses.

The EE Department at the USF has recognized the need for designing courses through SE methodologies. In this manuscript, the developed framework is illustrated by presenting the results of an implemented course as a case study. The Concept of Operations was developed for the lab design to bridge the gap between skillset required by the industry and the learning objectives defined in the academic program. Following the SE framework, industry ready skills were identified based on the survey carried out from the TRUE Partners of the EE Department. The course design requirements were then identified based on the needs of the stakeholders; the industry, students and the EE department. Further, the course was implemented through CANVAS, a learning management system, by incorporating innovative instructional interventions. Finally, the case study is verified and validated by presenting the responses of the course exit survey and metrics such as grade point averages, enrollment, retention and completion rates. The rest of the paper provides a detailed description of all the aspects of course design framework such as: pedagogical survey; Concept of Operations; course requirements; design; implementation; testing; verification; validation; upgrades and maintenance.

2. Course Design Framework

Designing a course curriculum is a complex process as it involves identification of stakeholders, the stakeholders' needs and mapping the identified needs to the course requirements. In this work, a SE framework has been used to design a curriculum for the undergraduate embedded systems laboratory. Each aspect of the laboratory design is divided into individual subsystems. Fragmenting the process into individual subsystems enabled the designers to clearly understand the sequence of actions for each subsystem and their interactions. Figure 1 shows the SE framework that was used to design the course; the course design is initiated by accomplishing a pedagogical survey. This survey enabled the course designer to identify the stakeholders within the course ecosystem, the existing course design methodologies and the stakeholders' needs. As a next step, a feasibility study was performed, followed by the development of the Concept of Operations, (ConOps), for the laboratory; then, the course requirements were defined from the ConOps. During this phase, a one to one relation between the ConOps and the course validation was formed based on the exit surveys. Once the requirements were defined and laid out, the course design and implementation became streamlined. During this phase, the verification criteria for course requirements were defined. The course verification and validation criteria served as a feedback for the designers to tune the course for any further upgrades and maintenance.

2.1 Pedagogical Survey and Concept of Operations

Pedagogical survey analyzes needs pertaining to different stakeholders in relation to the course design. In this subsystem, the stakeholders who play an important role within the course ecosystem are identified. Once the concerned stakeholders involved in the process are identified, their needs are assessed. In the current case study, the industry, the students and the department of Electrical Engineering were the major stakeholder in the ecosystem. With respect to the stakeholders pertaining to the industry, TRUE Partners such as: General Electric; Honeywell; Welbilt; Occam MD; RCA Solutions; Florida Power and Light; DeliverLogic; and Withlacoochee River Electric Cooperative Inc. were contacted and their needs were assessed. The needs assessment from industry showed that a skilled work force in the form of embedded C programmers with the basic knowledge of programming in Real Time Operating Systems, ARM and PIC architecture based processors are required. In addition, a survey conducted by IEEE Spectrum found that the need for embedded C programmers has reached an all-time high[11]. Another survey showed that 71 % of the industry selected a specific prototyping board based on the availability of software development tools. In the industry, most widely used software development tool was an eclipse based Integrated Development Environment (IDE)[12-14]. With respect to the students, major needs were to learn the technical material with ease and to be able to smoothly transition from academia to industry. In engineering education, designing the course based on trends and developments of the industry is a dynamic process. This dynamic environment has created a need for a new course design framework within the department that adapts to the trends and needs in industry. In addition, achieving high retention and completion rates were other departmental needs. Taking the identified needs into account, the ConOps for the lab was developed. The course vision was to enable a smooth transition for the students from the academia to the industry by means of a dynamic course design framework and innovative instructional strategies.

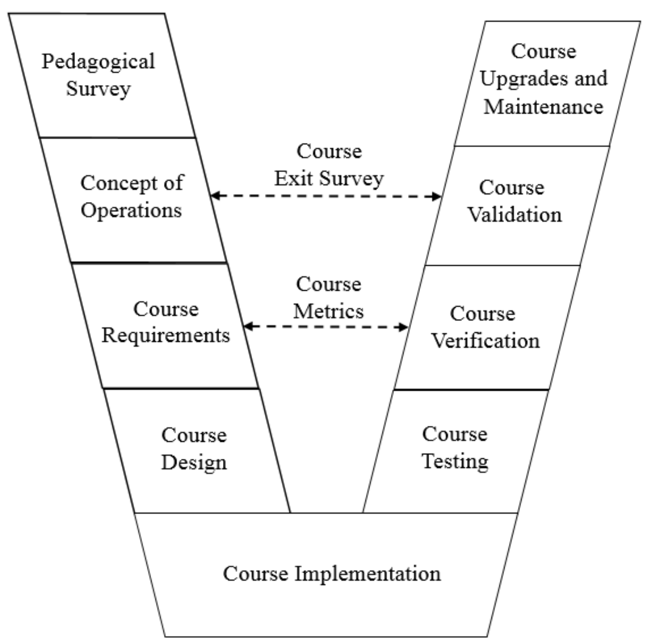

Figure 1. High level Overview of the Course Design Framework.

2.2 Course requirements

As a next step, the course requirements were formulated based on the input from the pedagogical survey and the ConOps. When it comes to needs of the industry, it is highly fluctuating as technology is constantly evolving and short-lived. As a result, preparing a student to adapt to this transitory environment is a challenging task. From the industry perspective, it is important to keep track of the evolution of the software and hardware tools. This will help in predicting the future platforms and tools that industry may adopt. Therefore, the course requirements were formed in such a way that by learning the fundamental concepts presented in the course, the students can smoothly migrate to any future technology. Based on these needs, a set of eleven requirements were defined. Since, the laboratory course is student centric, it is crucial to transform their needs into requirements and map them into effective teaching methodologies. Also, it is critical to determine the skill set which are in line with the interest of industry. Rather than assuming that students already possess the prerequisite knowledge required for the course, providing a review of the basics tend to help students navigate through the available resources. This will also create a conducive environment for learning and consequently can lead to an, efficient course completion. Hands-on experience in the laboratory course work is critical to get a better understanding of the concepts. This will simulate an industry like environment in the classroom where they will be constantly challenged to solve unforeseen problems. As a result, determining the right hands-on experiments in order to enable smooth transition from theory to practice is necessary. In the current case study, the students' needs were transformed into a set of four requirements. The next stakeholder is the department, which plays an active but invisible role in determining other educational requirements of the laboratory course work. The department has to properly place the laboratory course within the curriculum map. Since the scope of the instructional material for this course is very broad, it is necessary to carefully determine the right contents and a delivery time frame. Finally, the resources in the form of classroom availability, instructor availability, and software/hardware tools need to be streamlined. To transform all the departmental needs, a set of six requirements were identified in the context of the current case study. Table 1 summarizes the transformation of needs into course requirements specific to each stakeholder.

Table 1: Stakeholder Requirements for Embedded Systems Laboratory Design.

#1. Industry Requirements:	
1.	The course shall provide students with deep understanding of ARM architecture.
2.	The course shall teach students how to integrate software development kits into an eclipse based Integrated Development Environment (IDE).
3.	The course shall instruct students on how to design experiments corresponding to General Purpose Input Output (GPIO) peripherals.
4.	The course shall provide students with a knowledge base that enables creation of Interrupt Service Routines (ISR).
5.	The course shall provide an experimental understanding of Analog to Digital Conversion (ADC).
6.	The course shall provide a practical knowledge on how to implement Universal Asynchronous Receive and Transmit Protocol (UART).
7.	The course shall provide practical knowledge about time management using Periodic Interrupt Timer (PIT).
8.	The course shall provide a deep understanding of how to use smart sensors and manage data using Inter Integrated Circuit (I2C).
9.	The course shall provide a deep understanding of implementing Controller Area Network (CAN) protocol through a simulation of real time automotive application.
10.	The course shall provide experimental knowledge of data management using TCP/IP protocol through an Internet of Things (IOT) application.
11.	The course shall enable students with an implementation knowledge of Universal Serial Bus (USB).
#2. Student Requirements:	
1.	The course shall provide hands-on experience of latest embedded hardware and software tools.
2.	The course shall provide an active learning experience.
3.	The course shall provide a review of required fundamental knowledge.

4.	The course shall be facilitated using a human in the-Loop learning model.
#3. Department Requirements:	
1.	The course shall be designed using a dynamic framework to reflect latest developments in the industry to enhance employability.
2.	The course shall simulate an experiential learning environment for the students.
3.	The course shall provide a maximum learning experience for the students.
4.	The course shall be delivered within a 12 week timeline.
5.	The course shall be counted as one credit hour towards the program.
6.	The course shall maintain high retention and completion rates.

2.3 Course Design

With ever-changing information available for the course design, judicious selection and presentation of the right information within a stipulated time period are important decisions for a curriculum designer. The content needs to be conveyed to the students such that it captures their attention and is easily understood. For laboratory design, practical demonstrations have to be intertwined with theoretical perspectives. Challenges in real life scenarios have to also be encapsulated within practical demonstration so that the industry environment is simulated in the laboratory setting. In the current case study, instructional interventions like spacing, interleaving, pair programming, social learning, online learning and gamification have been incorporated to make the learning interactive, challenging and engaging.

2.3.1 Spacing and Interleaving

As per the cognitive research findings, greater performance in terms of memorizing and learning was observed when the same concepts are repeated at a specific time interval (spacing), and when different concepts are intertwined together (interleaving)[15]. The curriculum of the lab was designed to include these interventions to help students be successful. A basic introduction to a new concept needs to be discussed and the module has to be designed to reuse these concepts in addition to the introducing the new concepts over subsequent modules. For example: GPIO concept needs to be taught in the beginning of the course and be incorporated into the subsequent modules like timer, interrupt and I2C. This repetitive and interleaved usage of the concepts would thereby help students to understand the course material clearly and maintain the knowledge for longer period of time.

2.3.2 Pair Programming and Social Learning

In addition to theoretical learning of embedded hardware and software architectures, the lab curriculum also includes a challenging practical team activity. Working in a team fosters a conducive learning environment for the students. Better dissemination of the information, learning with imitation and easy transfer of skills are the primary advantages of the pair programming and social learning[16]. As a result, having incorporated these methodologies into the curriculum will greatly benefit the students. This enables the students to understand the technical material with ease and capture detailed information through peer interaction.

2.3.3 Online Lectures

It is important that the tools and instructional materials are readily available for students to access the content at their own convenience and personal needs. Familiarizing the students with the basics prior to the class will enhance their understanding of the concepts and will easily get reflected during the practical implementation. Hence, online delivery is an important tool for self-paced learning and engagement. It also helps students to revisit the concepts again and again thereby help in obtaining the advantages offered by spacing and interleaving.

2.3.4 Gamification

Programming is a challenging task. As a result, for laboratory courses like embedded systems design it is important to keep students motivated to learn different programming methodologies. Gamification when incorporated with teaching, enhances the student engagement. This methodology constantly challenges the individual and keeps them motivated to solve problems. It also fosters active learning by providing them a bonus for every hindrance they overcome. As a result, it is proposed to interleave the concepts of gamification while designing the modules and rewarding the students for every module they complete.

Course design when implemented with above mentioned interventions will enable the instructor to convey the course content in an efficient manner. This will also provide students with maximum and experiential learning experience and will help them to retain the material for a longer time. The next section of the paper outlines the key processes for course implementation.

2.4 Course Implementation

The earlier subsystems of the course design are the basic building blocks for the efficient course implementation. The defined requirements of the stakeholders when combined with the innovative instructional interventions of the course design yields a crisp definition of the course implementation. The first step to course implementation was to combine the requirements of all the stakeholders and implement them into the course within a stipulated timeline of 12 weeks. For efficient availability of the course contents to the students, CANVAS, a learning management system, was selected. CANVAS enabled the designers to effectively implement interactively rich course content pages and enable modularity in course as shown in Figure 2.

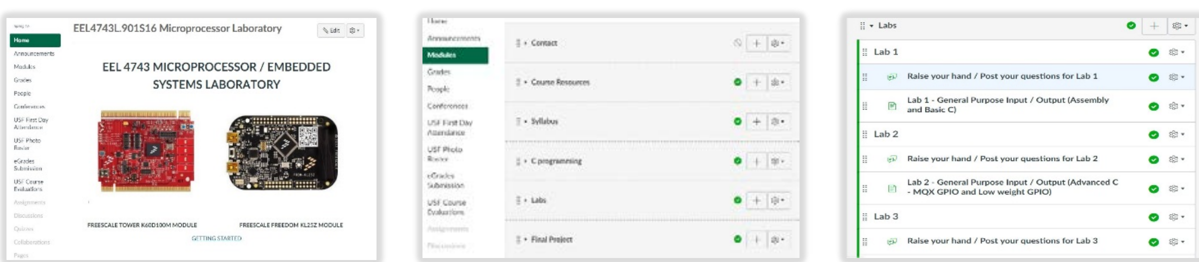

Figure 2: Overview of Course Implementation in CANVAS.

The learning material for this course was divided into various modules; each module included a pre-lab, post-lab, video lectures, application notes and code snippets as shown in Figure 3. The students were to complete each module within a time period of one week. By completing the components of the module, the students were constantly in the loop, thereby, their constant participation was enhanced. Students were put into teams and were challenged with in class exercises, as well as individual activities within pre-labs and post-labs. The CANVAS platform comprised of discussion forums to encourage peer interaction.

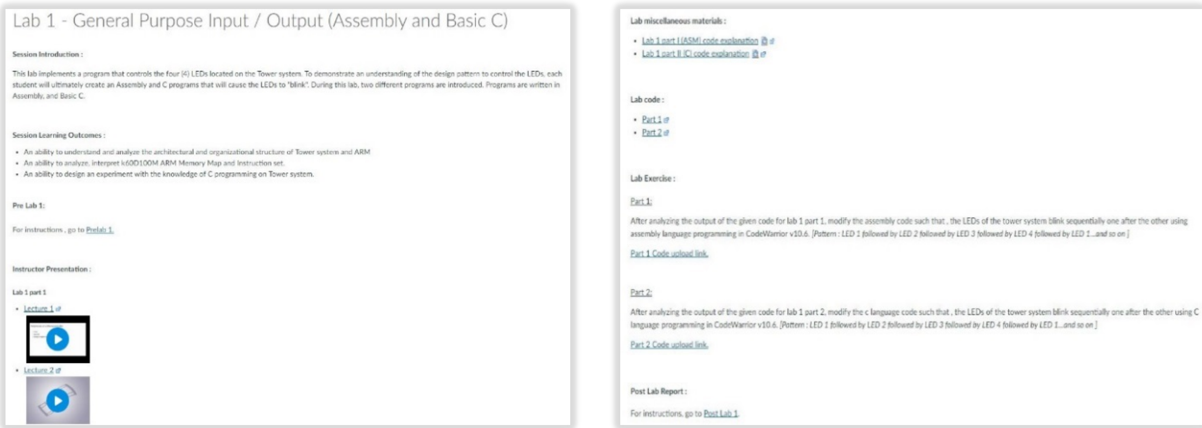

Figure 3: Material Flow in the Each Canvas Lab Content Page.

In the current course, students go through a designed sequence of activities each week. This enables the students to participate in active learning process. Pre-lab activities have to be performed in the form of viewing the online lectures and answering questions based on the concepts to be implemented. For a hands-on experience with respect to hardware implementation, a prototyping board needs to be programmed using embedded C by the student each week before coming to the lab. These activities have been enhanced to reflect the instructional strategies like gamification, spacing and interleaving. The actual lab session that meets once per week begins with a presentation wherein the instructor motivates the student with a "real-time application", followed by a detailed presentation of the concepts and then navigates the students through a flow chart of the particular activity that needs to be performed during that lab session; this type of delivery fosters interaction among peers and the instructor. These activities provide the advantages rendered by the instructional strategies shown in Table 2. Finally, the post-lab exercises help students to understand the theoretical aspects of the lab demonstrations. Discussion forums are actively answered by the instructor, giving students a direct communication channel with the instructor. In addition, the students also demonstrate industry readiness by implementing a real time application through final projects. Table 2 summarizes the advantages of the instructional strategies that are associated with each of the course components.

Table 2: Canvas Implementation Overview of Innovative Course Design Interventions.

Course Content	Instructional Strategy
Pre-lab	Gamification, Spacing
Post-lab	Gamification, Spacing
Discussion Forums	Human in the-Loop Learning, Social Learning
Online lectures	Online Learning
Application Note	Online Learning
Class Instructions	Human in the-Loop Learning , Social Learning
Class Exercises	Paired Programming, Social Learning, Spacing and Interleaving
Pre class Activities	Spacing, Online Learning, Interleaving

Based on the requirements of the industry, it was decided to use a NXP's TWR K60D100M Tower Module, a microcontroller based on the ARM cortex M4 architecture as the development platform for the current course[17]. In terms of software, the Kinetis Software Development Kit (KSDK) by NXP with a collection of peripheral drivers, high level stacks including USB and LWIP with the RTOS support was selected as the desired architecture. Kinetis Design Studio (KDS) IDE is built on top of Eclipse environment and provides an editor that recognizes the syntax of C language files, uses color highlighting for syntactic elements, provides a simple means for indenting code, integrates project management capabilities for determining dependencies between source code files, locating libraries, and maintaining up-to-date executable files[7,18,19]. Figure 4 depicts the TWR K60D100M development platform with labeled peripherals and the corresponding KDS IDE.

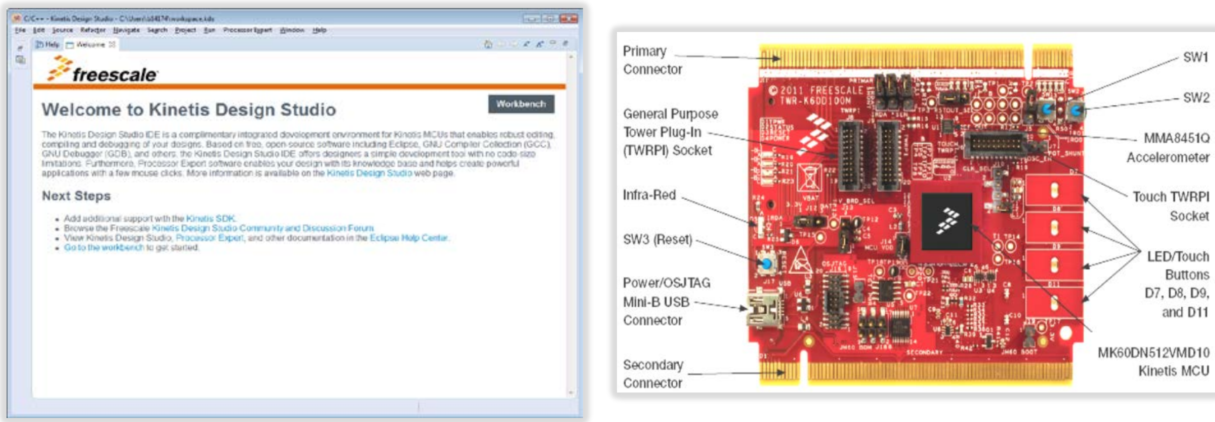

Figure 4: Implemented KDS IDE and TWR-K60D100M Development Board.

The next step was to implement the course content in a chronological manner for the efficient assimilation of fundamental knowledge and skills to the advanced competency levels required by industry. In order to provide a smooth learning environment, the curriculum was designed to follow a sigmoid function as shown in Figure 5. The course began with the introduction to embedded C, KDS, KSDK and ARM architectures followed by GPIO, ISR, ADC, UART, Timers, I2C, CAN and IOT based applications. This design effectively acted as a bridge between the fundamental knowledge and skills to the latest advances of the industry.

2.4.1 GPIO

A GPIO is an interface available on most modern microcontrollers to provide an ease of access to the devices internal functionality. Generally there are multiple GPIO pins on a single microcontroller for the use of multiple interactions to simultaneous application. The pins can be programmed as input, where data from some external source is being fed into the system. The output can also be performed using GPIOs, where internal information can be transmitted to external peripherals. In this lab, GPIO programming was implemented to demonstrate input and output operations. This is accomplished by setting-up the appropriate pins of the GPIO and reading the state of the Push-Buttons that are connected on top of the Tower system. Another aspect of this lab is to demonstrate the I/O design patterns for both input and output with multiple pins interacting with the software.

2.4.2 Interrupts

Interrupts in microcontrollers are the most important functionalities that enable multitasking. The interrupts allow the microcontroller to attend different peripheral requests based on priorities. During the interruption, the microcontroller is asked to execute a piece of code written specifically to service each request from an interrupt source. In this lab, students learn how to configure interruptions and learn how the KSDK interrupt manager handles hardware interrupts and exceptions using ISRs.

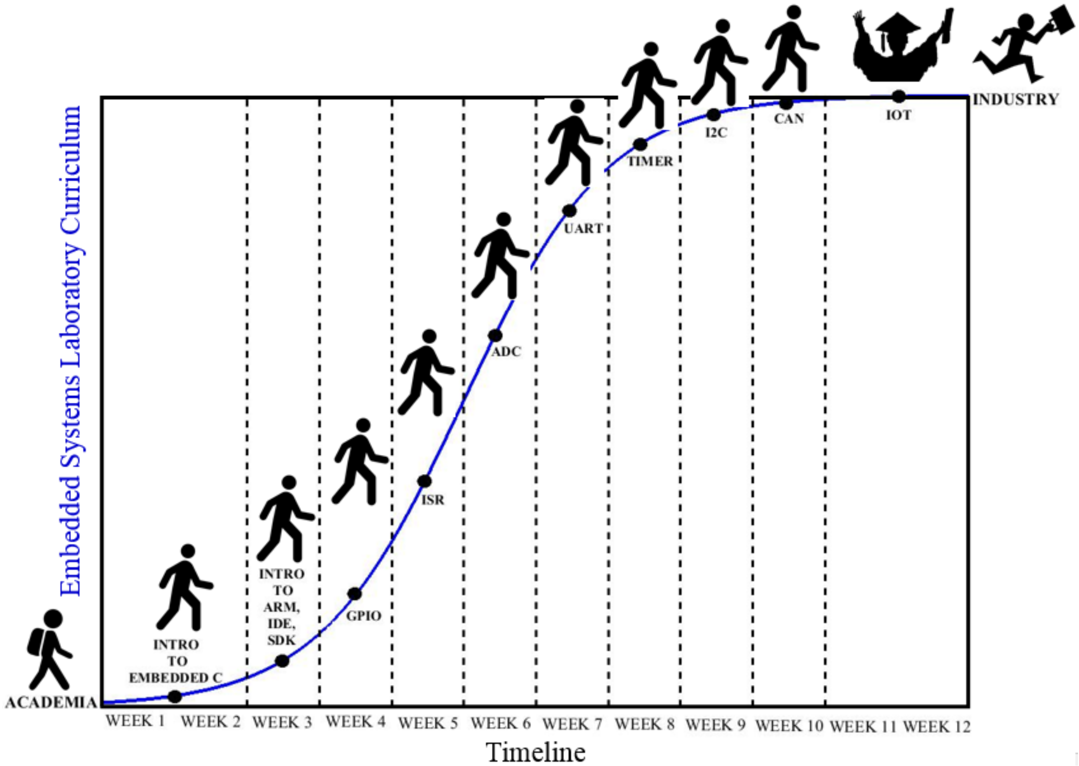

Figure 3: Embedded Systems Laboratory Learning Curve.

2.4.3 ADC

Analog to Digital Conversion is a key feature in microcontrollers, while all the processing inside the microcontroller is done digitally, the real world is analog, and therefore many of the applications require first converting the information from analog to digital in order to be processed further. This lab introduces Analog to Digital Conversion in embedded systems, and also leverages on the knowledge gained in previous labs. This lab will focus on how to use the ADC by reading an external analog sensor signal.

2.4.4 UART

Serial communication is simple in terms of resources. It involves digital transmission through a simple wire and it is cost effective in comparison to parallel transmission. . Specifically UART communication module comes integrated in most of the microcontrollers. This lab is intended to give students an opportunity to become familiar with installing an UART device for receiving and transmitting characters.

2.4.5 Timers

The timer module plays a crucial role for proper functionality and synchronization of different components of the board. To effectively introduce the timer module to the students, an example related to working principle of digital watches is demonstrated in the lab. Then, the students will use the timers through interruption, and other time management applications to gain implementation knowledge.

2.4.6 I2C

The I2C bus is a standard bidirectional interface that uses a controller, known as the master, to communicate with a device, known as a slave. A slave may not transmit data unless it has been addressed by the master. Each device on the I2C bus has a specific device address to differentiate between other devices that are on the same I2C bus. Many slave devices will require configuration upon startup to set the behavior of the device. This is typically done when the master accesses the slave's internal register maps, which have unique register addresses. A device can have one or multiple registers where data is stored, written, or read[20]. In this lab, the students will use the I2C module to communicate with an on board accelerometer and will dynamically display the acceleration values via the UART.

2.4.7 CAN

The CAN bus was developed by BOSCH[21] as a multi-master, message broadcast system that specifies a maximum signaling rate of 1 megabit per second (bps). In a CAN network, many short messages like temperature or RPM are broadcasted to the entire network. The CAN communication protocol is a carrier-sense, multiple-access protocol with collision detection and arbitration on message priority where the higher priority identifier always wins bus access. Signaling is differential which is where CAN derives its robust noise immunity and fault tolerance. In this lab, students will gain a deep understanding of data management through CAN bus by simulating a real time CAN network.

2.4.8 USB

The Universal Serial Bus (USB) was designed from the ground up to be an interface for communicating with many types of peripherals under the same communication standard. It is easy to use, so there is no need to fiddle with configuration and setup details, and it is fast. USB is a likely solution any time that a computer is required to communicate with other peripheral devices. Due to all these features, the USB protocol is constantly used in the microcontroller industry and almost a mandatory peripheral used during product development. In this lab, students will be trained to acquire the implementation knowledge of the USB.

2.4.9 Ethernet

Ethernet is the most widely deployed network in offices and industrial buildings. Based on its ease of use, low cost, high bandwidth, stability, security, and compatibility across devices, Ethernet has become the de facto standard of network access for 32, 16 and 8 bit microcontrollers. It enables to monitor, to control and to access devices over the Internet. Today's embedded systems designers and developers are increasingly asked to incorporate Ethernet connectivity into their systems. Ethernet's ubiquity and longevity in connecting to a network makes it an attractive communication choice for embedded systems. Through this lab, students will gain practical knowledge of Internet of Things by managing data coming from a HTTP server to the board using TCP/IP stack.

2.5 Course Testing and Verification

The implemented course is initially tested by the graduate students and industry personnel before the course is facilitated to the students. The ground work for the course verification has been performed in the course requirements phase. This phase is crucial as it provides a concrete feedback to the designer about the flaws and improvements required in the design process. Based on the output of this phase, course validation, upgrades and maintenance will be accomplished. Grade point averages, enrollment, retention, and completion rates were determined to be the key indicators of the course verification process. These parameters were calculated for each semester starting from spring 2014 when the developed framework of course design was first implemented. Further, the metrics indicate that all the defined requirements were addressed.

2.5.1 Retention rate

Retention rate is the amount of students actively registered in the course at the end of each semester. Educational institutions use this information to evaluate overall student engagement and coursework effectiveness. Figure 6 depicts the retention rate per semester. It is calculated as shown in (1).

$$\text{Retention Rate} = \frac{\text{No. of students that finish the course}}{\text{Total number of enrolled students}} \text{ X } 100 \tag{1}$$

2.5.2 Course Completion Rates

Completion rate is the amount of students that passed the course and is calculated as a ratio of total number of students that finish the course with A, B and C grades to total number of students enrolled in the course. Figure 6 shows the trend of course completion rates per semester. It is calculated as shown in (2).

$$\text{Course Completion Rate} = \frac{\text{No. of students that pass the course with A, B and C}}{\text{Total number of enrolled students}} \text{ X } 100 \tag{2}$$

2.5.3 Enrollment rate

The enrollment rate is used to know interest of the students over the semesters. In the results presented, it can be seen that the term spring 2017 has achieved highest enrollment rate due to upgrades in the curriculum. Figure 7 demonstrates trends of the course enrollment rate.

2.5.4 Grade Point Averages

Figure 7 shows the grade point averages for 7 semesters since the introduction of the new course design framework in spring 2014. The calculation of the grade point averages is done by using (3).

$$\text{GPA} = \frac{\text{A} * \text{no. of As} + \text{B} * \text{no. of Bs} + \text{C} * \text{no. of Cs} + \text{F} * \text{no. of Fs}}{\text{Total number of students}} \tag{3}$$

In (3), grades A, B, C, and F are considered to be 4, 3, 2 and 1 points respectively. The GPA trends for microprocessor laboratory are found to be stable above 3.3.

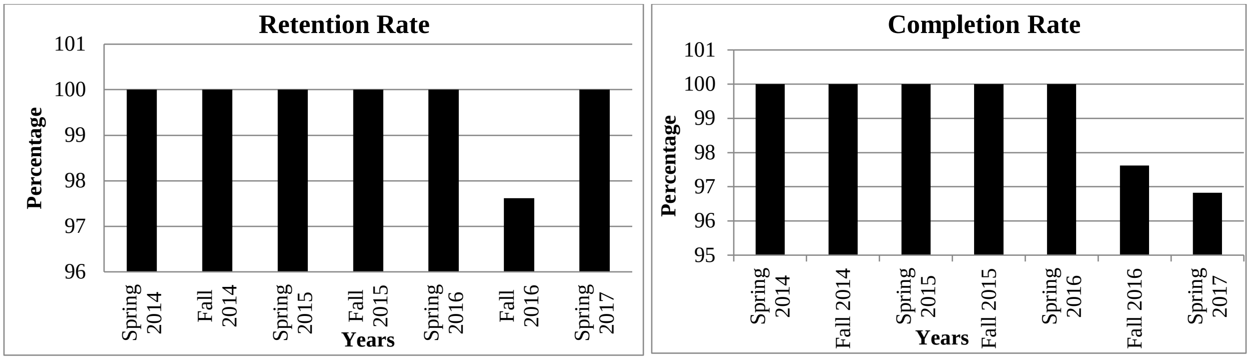

Figure 6: Course Retention and Completion Rate Per Semester.

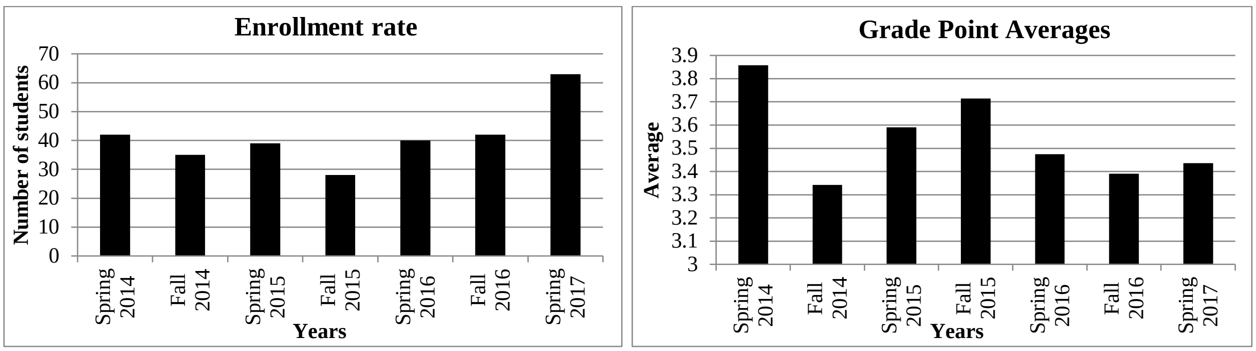

Figure 7: Enrollment Rates and Grade Points Averages Per Semester.

2.6 Course Validation, Upgrades and Maintenance

The results of the course verification criteria provided a clear picture of the improvements required for the course design. A course exit survey was conducted as a means of course validation which focused on specific areas of the course design such as ease of accessing the information, stimulation and facilitation of learning and important suggestions for lab improvement. Table 3 provides a summary of course exit survey responses that was used to upgrade and maintain the course constantly. The summary of course exits surveys show that most of the students had maximum learning experience from the current framework. The next important feedback was from stakeholders of the industry. After every couple of semesters changing trends in the requirement patterns of the skill set, and the hardware and software tools, used were analyzed. Based on all these inputs, modifications in the form of addition of lab activities were designed and incorporated. In the context of the current case study, a series of hardware and software tools were used upgraded based on the feedback. During initial semesters of the lab, Cold fire V1 with Code warrior and MQX operating was used. Then the lab was upgraded to ARM cortex M4, Code warrior and MQX RTOS. Currently, the lab is being implemented using ARM cortex M4, KDS and KSDK. Upgrades in the form of change in the instructional techniques and methodologies have also been incorporated in the microprocessor lab course design throughout the implementation period.

Table 3: Summary of Course Exit Surveys from Spring 2014 to 2017.

Questions	E(%)	G(%)	F(%)	P(%)
Description of Course Objectives and Assignments	60.6	32.8	3.3	3.3
Communication of Ideas and Information	65.5	21.3	6.6	6.6

Expression of Expectations for Performance	73.7	22.9	3.3	0
Availability to Assist Students In or Out of Class	95.1	6.5	0	0
Respect and Concern for Students	91.8	3.3	3.3	0
Stimulation of Interest in the Course	67.7	16.4	9.8	6.6
Facilitation of Learning	70.4	18	4.9	6.6
Overall Rating of Instructor	85.1	11.5	1.6	1.6

E=Excellent, G=Good, F=Fair, P=Poor

Conclusion

In this manuscript, a dynamic course design framework using systems engineering is being introduced. The developed SE framework is illustrated by presenting the results of an under graduate microprocessor laboratory course as a case study. Following the SE framework, initially a pedagogical review was accomplished from the stakeholders within course ecosystem. From the review, the stakeholders were identified and their needs were assessed. Then, the ConOps was developed for the lab design. Later, the requirements were defined specific to the stakeholders i.e. the industry, the students and the department. Later, critical components of the course design like identifying the instructional objectives of the course, demand metrics of the market, identifying the innovative methodologies for correct delivery of course content were determined. Finally, based on these requirements the course design was finalized and implemented. The course was then implemented through CANVAS, a learning management system, such that the contents were delivered to the students by incorporating the innovative instructional interventions. The SE approach makes the course design portable and reconfigurable thereby easy to maintain and upgrade. From the results of verification and validation criteria it can be concluded that a Systems Engineering approach is very effective in maximizing the student learning experience and in closing the academia and industry skill gap.

References

1 D. Spiller, "An Introduction to Course Design," The University of Waikato, 2011.

2 L. Fink, "A Self-Directed Guide to Designing Courses for Significant Learning," Jossey Bass, San Francisco, 2003.

3 R. Felder, "Designing and Teaching Courses to Satisfy the ABET Engineering Criteria," Journal of Engineering Education, vol. 92, no. 1, pp. 7-25, 2003.

4 A. Rocio, "Laboratory Enhancements for Improving Embedded Systems Education," in ASEE Electrical Engineering Technology Curriculum, Honolulu, 2007.

5 M. Todd, "New Developments for Courses in Embedded Microcontrollers," in American Society for Engineering Education, Honolulu, 2007.

6 E. Bohorquez, "Teaching Medical Electronics to Biomedical Engineering Students: A Problem Oriented Approach," in ASEE BME Courses and Learning Activities, Vancouver, 2011.

7 K. Joseph and J. Su, "Modernizing The Microcontroller Laboratory With Low-cost and Open-Source Tools," in American Society for Engineering Education, San Antonio, 2012.

8 G. W. Recktenwald and D. Hall, "Using Arduino as a Platform for Programming, Design and Measurement in a Freshman Engineering Course," in American Society for Engineering Education, Vancouver, 2011.

9 M. Bower, "A Learning System Engineering Approach to Developing Online Courses," in Australasian Conference on Computing, Hobart, Australia, 2006.

10 I. Bozkurt and J. Helm, "Development and Application of a Systems Engineering Framework to Support Online Course Design and Delivery," Advances in Engineering Education, vol. 3, no. 3, pp. 1-24, 2013.

11 N. Diakopoulos and S. Cass, "Interactive: The Top Programming Languages 2017," IEEE, 2017.

12 K. Hass and J. Su, "Modernizing the Microcontroller Laboratory with Low-Cost and Open-Source Tools," in American Society for Engineering Education, San Antonio, 2012.

13 R. Quinnell, "2015 Embedded Markets Study Changes in Today's Design, Development & Processing Environments".
14 I. Martinez and D. Torres, "Using General-Purpose Input Output on MCF5213 ColdFire Microcontrollers Freescale Semiconductor," 02 2007. [Online]. Available: https://www.nxp.com/docs/en/application-note/AN3399.pdf.
15 S. Carpenter, "Spacing and Interleaving of Study and Practice," in Applying Science of Learning in Education, Iowa State University, American Psychological Association, 2014, pp. 1-11.
16 Agile Alliance, "Pair Programming," 352 Inc, 15 November 2017. [Online]. Available: https://www.agilealliance.org/glossary/pairing/#q=~(filters~(postType~(~'page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'pair*20programming))~searchTerm~'~sort~false~sortDirection~'asc~page~1).
17 Freescale Semiconductor, "User's Manual," 28 August 2011. [Online]. Available: http://cache.freescale.com/files/32bit/doc/ref_manual/TWRK60D100MUM.pdf.
18 NXP, "Getting Started with Kinetis SDK(KSDK) v.2.0," 01 2016. [Online]. Available: https://www.nxp.com/docs/en/user-guide/KSDK20GSUG.pdf.
19 Freescale Semiconductor, "Kinetis SDK v.2.0 API Reference Manual," [Online]. Available: https://www.nxp.com/docs/en/reference-manual/KSDK20APIRM.pdf.
20 Texas Instruments, "Understanding the I2C Bus," June 2015. [Online]. Available: http://www.ti.com/lit/an/slva704/slva704.pdf.
21 Bosch, "Introduction to the Controller Area Network (CAN)," Robert Bosch GmbH, Stuttgart, 1991.

Prashanth Chetlur Adithya

Prashanth received a B.Tech from Jawaharlal Nehru Technological University, Hyderabad, India and M.S.E.E. from the University of South Florida, Tampa, USA. He is currently a Ph.D. Candidate of Electrical Engineering at the University of South Florida. His research interests include innovation in engineering education, medical device design and development, biomedical signal analysis and modelling, artificial intelligence and embedded systems.

Shraddha Pandey

Shraddha received M.S. degree from Pune University, Pune, India. She is currently a Ph.D. student of electrical engineering at the University of South Florida. Her research interests include course design, delivering embedded systems, instrumentation and controls, data analytics, artificial intelligence and biomedical signal processing.

Jose Caballero

Jose received a B.S.E.E from ISPJAE, Cuba. He served as an instructor at ISPJAE for 5 years and as researcher in the Microelectronics Investigation Center in Havana, Cuba during the same time. He is currently a Ph.D. student at the University of South Florida. His areas of interest include engineering education, embedded systems, digital hardware design and deep learning.

Özgür Yürür

Özgür received a double major from Gebze Institute of Technology, Kocaeli, Turkey and the M.S.E.E. and Ph.D. degrees from the University of South Florida, Tampa, FL, USA. He is currently with RF Micro Devices, Greensboro, NC, USA, responsible for research and design of new test development strategies and also for the implementation of hardware, software, and firmware solutions for 2G, 3G, 4G, and wireless-based company products. His research area covers engineering education, ubiquitous sensing, mobile computing, machine learning, and energy-efficient optimal sensing policies in wireless networks. The main focus of his research is on developing and implementing accurate, energy-efficient, predictive, robust, and optimal context-aware algorithms and framework designs on sensor-enabled mobile devices.

Wilfrido. A. Moreno

Wilfrido Moreno received his M.S.E.E & Ph.D. degrees in Electrical Engineering from the University of South Florida (USF), Tampa – Florida in 1985 and 1993 respectively. He is currently a Professor in the Department of Electrical Engineering at the University of South Florida, Tampa – Florida. From 2003 Dr. Moreno has served as the R&D Initiative Director for the Ibero-American Science & Technology Education Consortium (ISTEC) responsible for fostering Teaching/Learning & Research collaborations throughout the Ibero-american region among ISTEC's members. His research interests are oriented toward system integration by providing "off-the-shelf" hardware/software solutions to industrial applications in areas such as Digital Signal Processing, Communications, Energy, Robotics & Control, Nano/Micro-electronics, Medical Engineering and Multimedia solutions applied to engineering education. Dr. Moreno has supervise over sixty master students and twenty doctoral students.