AC 2012-4299: INTRODUCING SYSTEMS ENGINEERING CONCEPTS IN A SENIOR CAPSTONE DESIGN COURSE

Dr. Michael W. Prairie, Norwich University

Michael Prairie is an Electrical Engineer who spent 15 years in the U.S. Air Force, managing R&D programs and developing various optical sensor technologies, and then ten years in industry developing infrared system concepts for DoD clients. In 2008, he returned to Norwich University, his alma mater, to teach electrical and computer engineering. His current interests include integrating sensors into embedded systems for controlling processes in various applications that support the courses he teaches.

Prof. Ronald Lessard, Norwich University

Introducing Systems Engineering Concepts in a Senior Capstone Design Course

Abstract

Senior capstone design projects can often expand in complexity to include systems of systems, particularly in projects involving embedded systems to control a larger system. Principles of Systems Engineering (SE) can be integrated into the capstone course to help students—who may not have been exposed previously—manage this increased complexity.

This paper presents an evolving framework of essential SE fundamental elements, including the top-level processes of Requirements Analysis, Functional Analysis and Allocation, Design or Synthesis, and Analysis & Control. A case study of a mechatronic system (a robotic tin whistle player) was used in a two-semester Electrical and Computer Engineering capstone design course in which the students had no prior exposure to formal SE processes. The students still had to design, build and test their projects' systems, so a balance had to be struck between teaching basic SE principles with its various reporting opportunities, and leaving enough time to do the hands-on engineering required to produce a prototype at the end of the year. The result is a capstone design course that exposes students to the language of SE so that they will be able to recognize its processes when encountered after graduation.

Introduction

Until recently, the capstone course in the Electrical and Computer Engineering (ECE) program at the authors' institution had been based on a traditional Engineering Design Process (EDP) that is outlined below in Table 1. This classical approach is appropriate for developing components and small systems, but a need to prepare students to manage the development of larger systems with more complexity was recognized, so a Systems Engineering (SE) framework was sought for the capstone course. Furthermore, the diversity of projects undertaken by the students often precluded a common process for the entire senior class to follow, leaving each advisor to manage their teams independently. The SE framework appeared to have a broad enough applicability to bring the projects back into a more common program, to be reintegrated by a series of lectures to introduce concepts and explain how to produce SE-appropriate deliverables.

The framework for presenting SE content in the capstone course for non-SE majors is to present modules as required[1] beginning in a single weekly lecture that decreased in frequency as the projects transitioned from intensive up-front planning to longer execution tasks. The SE process outlined in the Systems Engineering Fundamentals (SEF) guide[2] was initially adopted, and is outlined in Table 1 with an approximate correspondence to the steps in the EDP. Supplemental material was also drawn from Systems Engineering Handbook,[3] as well as other sources.

Table 1: The Engineering Design Process (EDP) compared to the Systems Engineering (SE) model.

Engineering Design	**Systems Engineering**	
• Identify the problem • Define goals and criteria	Requirements Analysis	• Analyze missions and environments • Identify functional requirements • Define constraints
• Research and gather data	Functional Analysis	• Decompose to lower-level functions • Allocate requirements to levels • Define functional interfaces • Define functional architecture
• Brainstorm • Analyze potential solutions • Model and test candidates • Down-select to one solution • Communicate and specify • Implement	Design Synthesis	• Define physical architecture • Define alternatives • Select preferred solutions • Define physical interfaces
• Review and assess (refine)	System Analysis and Control	• Trade-off studies • Effectiveness analyses • Risk management • Interface management • Data management • Performance measurement

.

The students in the capstone course have not been exposed to formal SE processes, so the challenge was to introduce the students to SE concepts and terminology without overwhelming the capstone experience and turning the "senior project" into a program management course. Furthermore, the small size of the institution's ECE program precludes large project teams, so the scope and complexity of the projects must be maintained at an appropriate level. A typical project will have approximately three or four subsystems or components that must be integrated into the overall system, but applying rigorous SE processes to even a few subsystems at all levels can overwhelm the design process. A more practical approach for such a non-SE degree program is to expose students to the concepts, language, and a minimum number of essential tools.[4] Other concerns include the diversity of capstone project experiences[5] and the question of uniform applicability of specific SE methods to all projects. To strike a balance, the emerging approach is to select a subset of key SE concepts, and apply them with some rigor to only one or two threads that reach the lowest decomposition levels. This approach offers the potential to preserve the capstone design experience while giving the students exposure to SE management tools as they use them to manage segments of their projects.

This paper presents a case study of a mechatronic system designed to play a tin whistle. The top-level processes of Requirements Analysis, Functional Analysis and Allocation, Design or Synthesis, and Analysis & Control are presented in sequence with associated SE concepts highlighted for the mechatronic "piper." A brief discussion of the students' responses and lessons learned will be presented as appropriate.

Requirements Analysis

The Requirements Analysis phase of the EDP and SE approaches are essentially the same, except the SE approach expands the problem identification to specifically analyze the mission to be performed and the environment in which it will be performed, as well as to identify the primary functions the system must perform to accomplish the mission. The outcome of this phase is a top-level Functional Flow Block Diagram (FFBD) and a set of Measures of Effectiveness (MOE) derived from Critical Operational Issues (COI) agreed upon by the stakeholders. Figure 1 depicts this process. In addition to the MOE, a Measure of Suitability (MOS) is often used, however the subtly of the difference between the two led to distraction among the students, so the MOS is only mentioned, while only the MOE is used in the course.

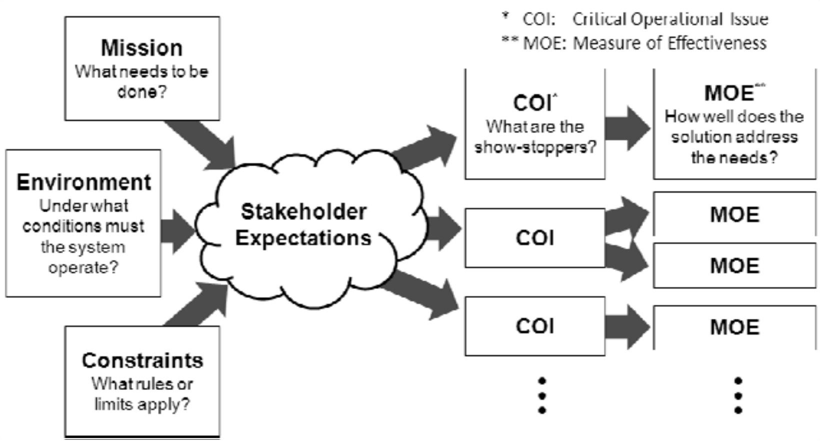

Figure 1: Stakeholder expectations converted to Measures of Effectiveness (MOE).

After considering the mission, environment, and constraints (cost, size/weight/power, IEEE Standards, available expertise, etc.), a concept of operations is developed, which is captured in a top-level FFBD. The top-level FFBD for the piper is shown in Figure 2. The labels on the function blocks are actions, not nouns—this is an important distinction that most students miss in their first attempt at a FFBD. Here, the mission is to play typical Irish whistle tunes with appropriate ornamentation, and a constraint is that the notes be derived using a MIDI (Musical Instrument Data Interface) –based protocol. The environment was not specified, so a benign laboratory environment is assumed. The COI for this example is simply to provide the system a sequence of notes to play, and then the system will produce an appropriate flow of air into the mouthpiece while opening and closing finger holes at the appropriate time.

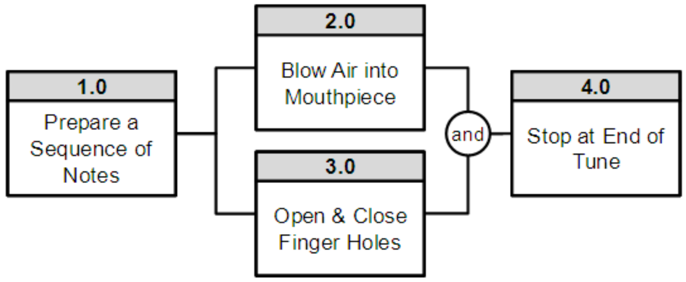

Figure 2: Top-level Functional Flow Block Diagram (FFBD) for the mechatronic piper.

The functions that must be performed were considered in the context of the mission & constraints, and "show stoppers" emerged as properties the system must have to adequately perform the functions. These "show stoppers" are the COIs. In order to know that the COIs are adequately performed, something has to be measured to prove it, so at least one MOE is usually defined each COI. The resulting COIs and MOEs for the piper are given in Figure 3.

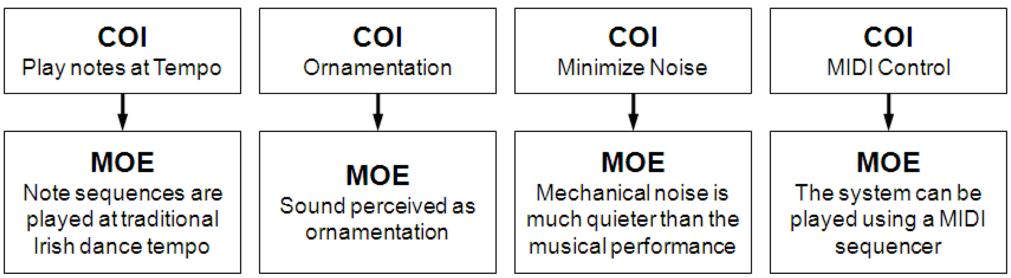

Figure 3: COIs and MOEs for the mechatronic piper.

Students can easily confuse MOEs with Measures of Performance (MOP) which are discussed in the next section, so a few properties of MOEs are worth noting:[6]

- MOEs are owned by the stakeholders who impose the requirements.
- MOEs are independent of any particular solution.
- MOEs provide a standard against which any solution can be measured
- MOEs provide a baseline from which future tests can be derived

MOEs represent an external view of any solution from the perspective of the customer, whereas MOPs are concerned with the performance of a particular solution from the perspective of the engineers that create the solution.

Functional Analysis

The Functional Analysis phase of the SE process is a disciplined way to break a problem down to its lowest levels without considering a particular solution. Refraining from thinking about solutions is difficult for students to grasp at first since they tend to want to start designing a widget to solve the problem right away, but they must wait for the Design Synthesis phase to do that. The discipline of the Functional Analysis phase, done early in the project, is why the SE approach is well suited for managing projects with a high level of complexity—it provides a procedure to break the problem down to a comprehensive set of low-level problems, and provides a method for keeping track of them.

In this phase the goal is to decompose the functions of the FFBD into lower-level functions until the next logical step is to transform the function to a physical component that can perform the function. The students are asked to decompose their FFBDs to three levels (although not all threads will reach three levels). Each function can be decomposed by asking "how do I know the function has been performed?" or "what needs to happen for this function to be performed?" and generating a flow of lower-level functions that will enable the higher function. An example of one thread in the piper FFBD decomposed to three levels is shown in Figure 4. For instance, asking "what needs to happen to close a finger hole?" under 3.2 generates functions 3.2.1, 3.2.2 and 3.2.3.

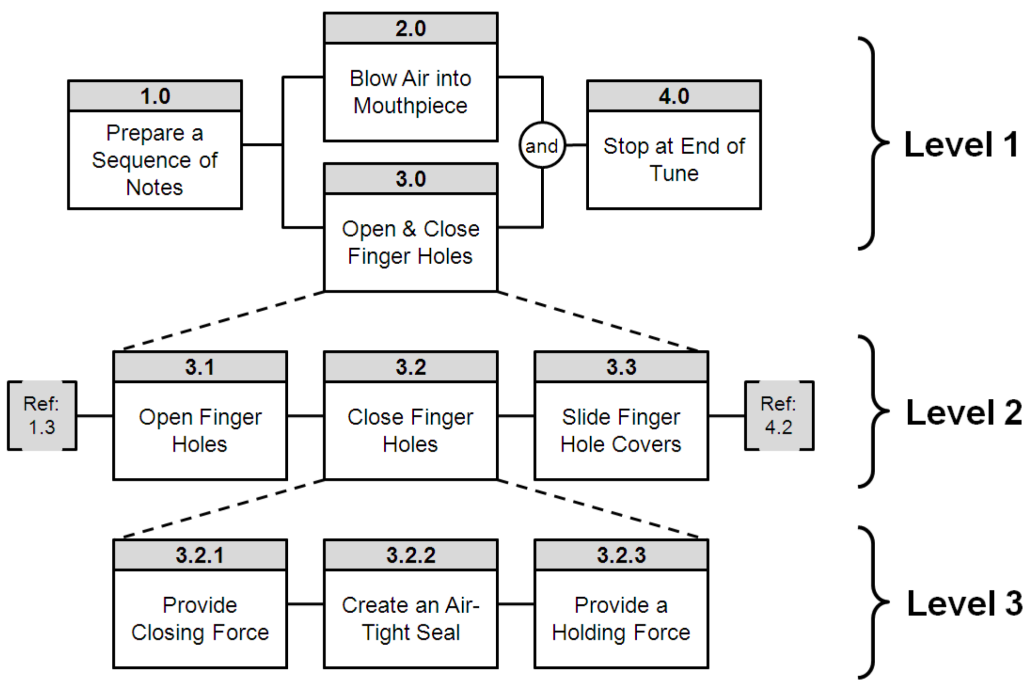

Figure 4: Piper FFBD decomposition to the third level showing the basic functions involved in closing a finger hole

Next, the MOEs are considered and the function blocks that will influence the MOEs are identified. For instance, the MOE "sound perceived as ornamentation" refers to "cuts" and "strikes" that are quick open-close or close-open sequences, respectively, that are too short to be perceived as separate notes. Clearly this MOE will be influenced by 3.1 and 3.2 as well as the level-three functions under them. In order to satisfy the MOE, a Measure of Performance (MOP) is created that will provide a basis for measuring the ability to open and close the holes fast enough for ornamentation. So the MOP for a “strike,” for instance, will be the duration that the hole is closed, or “the duration of a strike.” Note that the “measure” itself does not have a number—it is simply the thing that is being measured. There will, however, be a threshold value associated with the maximum duration of the cut or strike below which the performance is acceptable, and usually a baseline value as well that indicates a desired performance level that exceeds the threshold level of performance. The distinction is another concept with which the students have trouble at first. The determination of the threshold or baseline values may take some analysis or estimation; for instance the threshold value of a cut was found after analyzing cut durations on an instructional recording for learning to play the whistle.

The output of the Functional Analysis phase is a Systems Requirement Document (SRD) that provides

- statements of what the engineers understand to be the stakeholder’s requirements,
- the mission, environment and constraints,
- definitions and references to supporting documentation (i.e., MIDI Specification, etc.)
- the COIs and MOEs
- the three-level FFBD with a separate, numbered paragraph describing each functional block (with the same corresponding number as the paragraph)
- the MOPs with threshold and baseline values.

One type of metric not listed above is the Technical Performance Metric (TPM) that is similar to the MOP, which may be used as well. The TPM and MOP can easily be confused, but if the TPM is used, it should be reserved for measures of critical performance attributes that are associated with risk items that need to be tracked and their progress reported.

The SRD is used as a communication tool to make sure both the stakeholders and the engineers have the same understanding of what needs to be done, but it is also meant to be a living document that can be updated as refinements are made.

Design Synthesis

The Design Synthesis phase is when the students can finally focus on the solution. Since the functions have been decomposed to their lowest level, the basic approach to the Synthesis phase is to convert the functional architecture to a physical architecture, or to convert the actions of the functional blocks to physical components and subsystems capable of performing the functions. A two-step process is used that results in two presentations in the first semester of the course: a Preliminary Design Review (PDR) and a Critical Design Review (CDR). The PDR is presented mid-semester and includes a discussion of the initial design options and how they arrived at them, initial plans for developing the designs, and a summary of the Requirements and Functional Analyses. The CDR builds on the PDR, and includes tests and analyses performed leading up to the final design. The CDR is produced as a written final report for the first semester as well as the final presentation. The goal is to have a design that is sufficiently complete so that parts can be ordered over the holiday break between semesters, leaving the second semester to focus on building, testing and refining their designs, as well as preparing final reports and presentations.

The phase begins with the usual tasks (also in the EDP) of reviewing the literature to find out how similar problems have been solved elsewhere, and brainstorming ideas for performing the functions in the FFBD. As the brainstormed solutions emerge, they may be completely different, or they may be similar with different components or subsystems. The number of candidate solutions is then reduced by applying the constraints, leaving two to three viable candidates.

Each candidate solution is then rendered with a concept drawing that is used as a guide to create a block diagram. The example for the finger mechanism concept drawing for two of the piper finger actuators is shown in Figure 5, and the block diagram representing the system's physical architecture is shown in Figure 6 with the finger actuators highlighted.

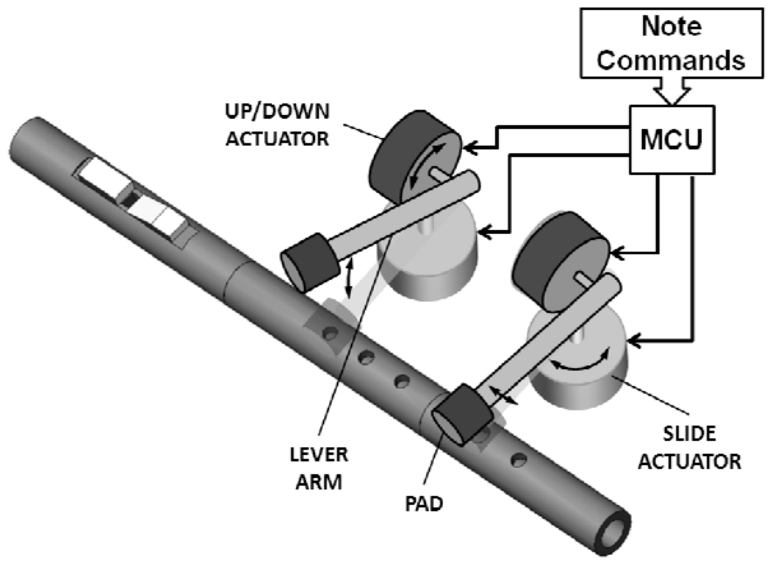

Figure 5: Finger actuator concept for two of six actuators. The lower-right actuator highlights the motion required to open and close the finger holes, and the upper-left actuator highlights the motion required for a slide ornamentation.

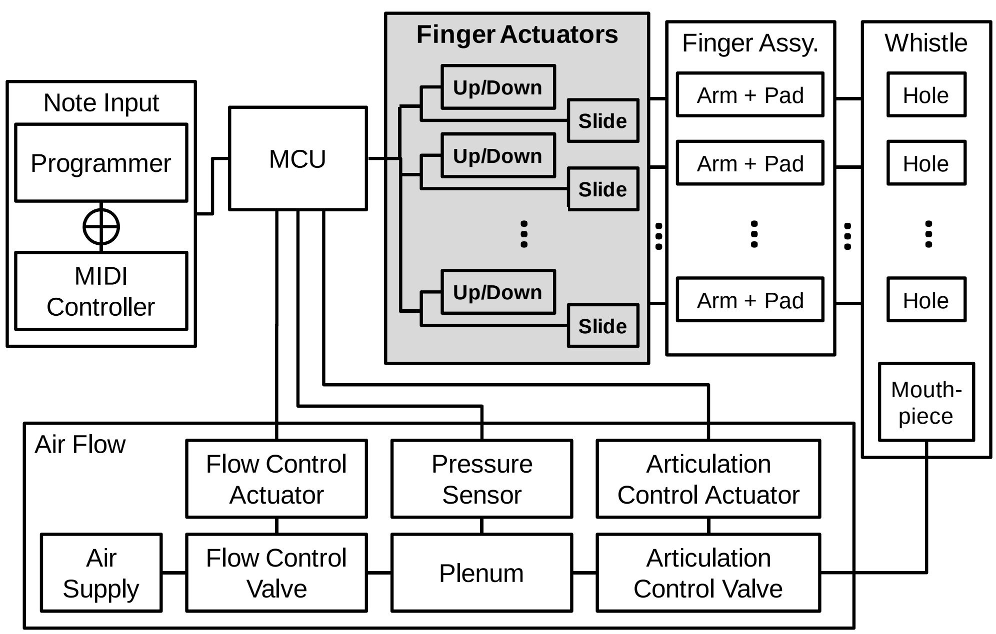

Figure 6: System block diagram for the piper physical architecture. The finger actuator block illustrated in the text is highlighted.

For a relatively simple system like the mechatronic piper, the number of candidate subsystems and components can grow quite rapidly, especially when, for example, technologies considered for the actuators alone could include linear or rotational actuators driven by solenoids, servo motors or pneumatics. Performing a rigorous decision analysis on all possibilities could easily overwhelm a small team of students. The students are, however, expected to do a decision analysis on at least one or two key components proposed as candidate solutions and summarizing the results in a decision matrix table. The decision criteria should be tied to the appropriate MOPs developed earlier, using weights commensurate with the performance threshold and baseline values. This can reduce the tendency of some students to adjust arbitrary weight factors until the emotionally-preferred choice emerges with the highest score. Most of the other choices can be made by invoking the "available expertise" constraint and going with low-risk, proven approaches in order to control the scope of the decision analysis.

Once the physical architecture is chosen, the students are asked to produce a table that maps the functional architecture to the physical architecture to make sure all functions are covered. This is also a useful tool to use to map MOPs to physical components when allocating requirements for future testing. The piper example is shown in Table 2.

Table 2: Functional-to-Physical Architecture map. Each block in the system block diagram is mapped with a "check" to show which function the physical component performs. The question marks indicate potential correspondence, depending on the final configuration.

Functional Architecture		Physical Architecture																
		Note Input		MCU				Finger Actuators		Finger Assy		Air Flow						
		Programmer	MIDI Controller	Digital I/O	A/D	D/A	Control Program	Up/Down	Slide	Arm	Pad	Air Supply	Flow Control Valve	Flow Control Actuator	Pressure Sensor	Plenum	Articulation Control Valve	Articulation Control Actuator
Prepare Note Sequence	Receive MIDI Sequence		✓	✓														
	Pre-Program Note Sequence	✓		✓														
	Convert to Control Signals			✓			✓											
Blow Air into Mouthpiece	Supply Air											✓						
	Control Air Flow Rate			?	✓	?	✓						✓		✓	✓		
	Articulate Air Flow			?		?	✓							✓	?	?	✓	✓
Open & Close Finger Holes	Open Finger Holes			✓			✓	✓		✓								
	Close Finger Holes			✓			✓	✓		✓	✓							
	Slide Finger Hoe Covers			?		?	✓		✓	✓	✓							
Stop at End of Tune	Stop Air Flow			✓			✓						✓	✓			?	?
	Return Fingers to Default Pos.			✓			✓	✓		✓								
	Wait for Next Tune						✓											

By the end of the first semester, the majority of the design process is complete, and the project is at the base of the iconic "V" of the systems engineering process. While some prototype items are usually built in the first semester—primarily for testing concepts—the main build-test phase occurs in the second semester. During that time, the students are encouraged to revisit and revise their design using the principles employed in the Functional Analysis and Design phases.

System Analysis and Control

System Analysis and Control is not a separate phase of the process; it is applied to all steps of the systems engineering process. The activities include technical management required to measure progress, evaluate and select alternatives, and document process. For the capstone course, the focus beyond what has already been discussed is on scheduling, test planning to ensure the MOP thresholds and baselines are tracked, and risk management.

One of the mainstays of managing the senior project, whether using the EDP or SE approach, is the Gantt chart. The students are introduced to the Gantt chart using Microsoft Project in their freshman engineering course, and that is revisited early in the capstone course when the initial Gantt chart is produced to show course milestones in a Management task and the high-level Requirement Analysis and Functional Analysis activities leading up to them. Producing a detailed Gantt chart that early in the process was found to be an inefficient use of time because the students had not yet worked out how they expected to solve the problem. The students later are asked to expand their Gantt charts to include the details of the physical architecture derived from a Work Breakdown Structure (WBS). The Functional-to-Physical Architecture map in Table 2 provides the bulk of the technical tasks in the WBS, as shown in Figure 7. These tasks are in addition to the management tasks that were included in the original Gantt chart.

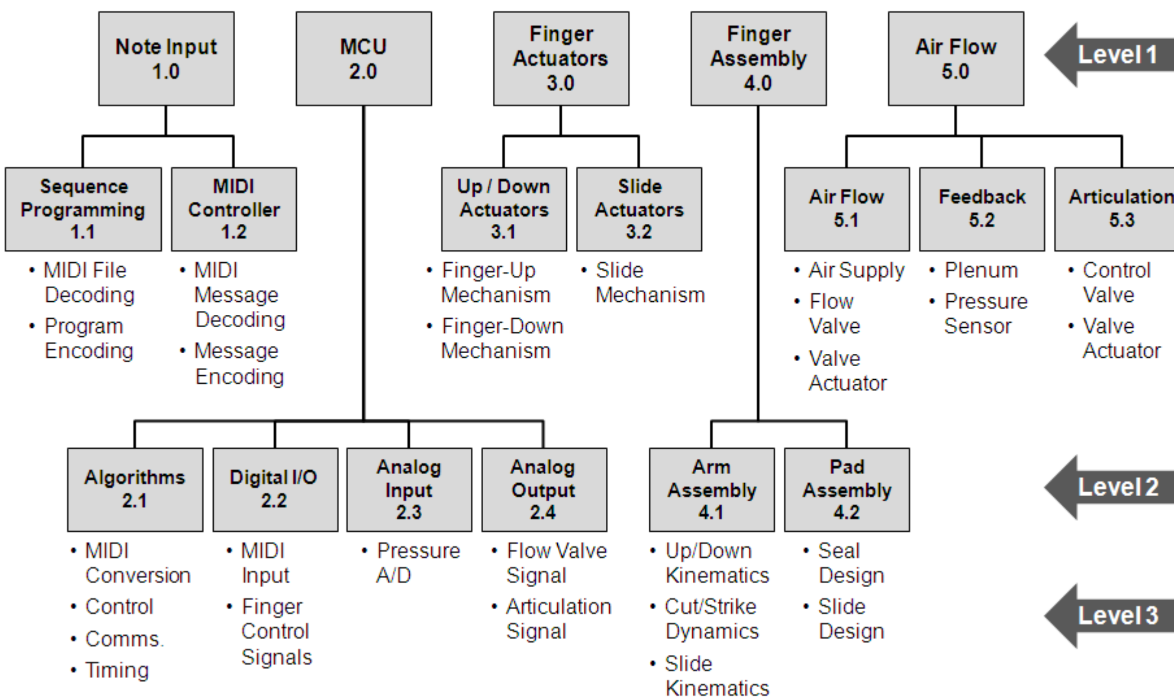

Figure 7: Work Breakdown Structure (WBS) for technical tasks in the piper project.

The students do a mid-semester presentation that is modeled as a Test Readiness Review (TRR). By this time, the students should be nearing completion of the components and subsystems, and be at least beginning to integrate them into the overall system. The scope of the TRR presented by the students is limited to the final test plan. The elements that the students are expected to include in their test plan are summarized in Table 3. The emphasis on the outcome continues to be on the MOPs as the traceable metrics that show satisfaction of the MOEs that embody the stakeholder or customer expectations.

Table 3: Test Plan elements and the questions they are to answer

Test Plan Elements	**Questions to be Answered**
Scope & Objectives	• What data do you mean to collect? • What system configuration will be used? • MOEs, MOPs
Procedures	• How will you measure for specific MOPs? • What is the required test setup?
Resources	• What equipment is needed? • Where will the test be conducted? Is it available? • What fixtures are needed? Are they fabricated?
Testable Features & Functions	• What specific functions will be tested? • Which blocks on the Functional Block Diagram?
Risks & Contingencies	• What if the test fails to yield the desired data? • What other ways of collecting data may be available?
Test Schedule	

Formal risk analysis is a concept that is currently being introduced into the capstone course, and the outcome is unclear as of the writing of this paper. This is being done at the risk of encroaching on the time available for the students to engage in the design-build-test process that embodies the "capstone" nature of the course. Risk analysis, however, is viewed as a valuable tool that can save time if things go wrong.

The main components of risk analysis are identification, analysis, and mitigation. Students are asked to identify several risk items that are critical to achieving MOP thresholds or baselines based on uncertainty (lack of technical knowledge, parts availability, scheduling pressures, etc.). for each item, they are to ask "What is the likelihood the situation or circumstance will happen?" and assign a probability value between 1 (not likely) and 5 (nearly certain). Then they are to ask "Given the event occurs, what is the magnitude of the impact to the project?" and assign a consequence value between 1 (little or no impact) and 5 (show stopper). The results are plotted in a risk matrix shown in Figure 8. The students then prioritize the risk items, and are encouraged to seek alternative approaches for high-risk items, and consider having an alternate plan for medium-risk items in case progress is too slow toward achieving the MOP thresholds.

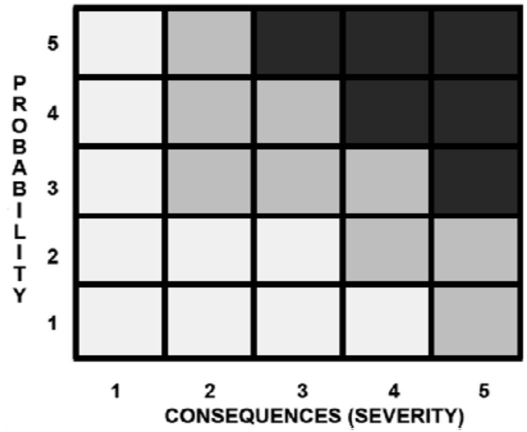

Figure 8: Risk matrix. Plotting the consequence and probability values for each risk item yields a risk level of low (light gray), medium (med. gray) or high (dark gray).

After final testing, the students prepare final presentations and reports. The final presentation is given in a forum that includes the wider School of Engineering and underclassmen are encouraged to attend in addition to the usual internal and external stakeholders. A poster presentation is also required that is tailored to and presented at a University-wide Scholarship Celebration. Finally, a written final report is required of all senior project teams.

Summary and Conclusions

This paper presents an evolving framework of SE fundamentals being used to incorporate systems thinking in a non-SE major capstone course through the application of SE principles. While not a rigorous SE course, it endeavors to expose students to SE tools that can be used in a meaningful way in a capstone design course that preserves the elements of a traditional engineering design project. This is the second year of implementing SE fundamentals in the course, and the scope of the SE component continues to be distilled so that the essential elements remain.

Formal assessment of the improvements have not yet been carried out, but anecdotal evidence suggests that the disciplined structure that SE offers has added focus to the course that was previously absent. This has helped some of the student teams to make and follow more focused plans, and to better manage increasingly complex projects. This was observed in some teams in which the students reluctantly held back from jumping into a design solution until after the SRD was delivered. The resulting solutions presented later in the PDRs took longer to develop than in the traditional approach, but the designs tended to be better decomposed into subsystems or elements that were attainable by the teams, reducing the tendency to naively attempt grander solutions outside their reach. An added benefit that emerged was that the SE structure implied an assessment rubric. The traditional approach had been less defined, and the students appreciated the more specific and objective grading criteria that were drawn from the deliverables.

The end-of-semester course evaluations indicated that the course generated further interest in the subject, and that the students would recommend the course to other students, even though it required significantly more work than their other classes. The evaluation questionnaire, however, is generic to all courses, so the SE component is not specifically addressed. One relevant comment from a student stated “The systems engineering approach seems to make a lot of sense for this application.” A more specific survey tool is needed to assess the improvements made by adding the SE framework, but a more appropriate group to poll than the current students will be the alumni who have entered the work force.

In summary, the course has explored some SE concepts with some successes as well as pitfalls encountered in the implementation process. Concepts that distracted students from the capstone experience were removed, and new concepts are being added where appropriate, such as the risk management module discussed above. The authors hope that the lessons learned will benefit other engineering programs that leverage SE principles as they seek an appropriate level of SE education in their disciplines.

Bibliography

1. McGrath, Elisabeth, Susan Lowes, Chris Jurado, and Alice Squires. "SE Capstone: A Pilot Study of 14 Universities to Explore SE Learning and Career Interest through DoD Problems." Vancouver, BC: ASEE, 2011. 725.
2. Defense Acquisition University. *Systems Engineering Fundamentals.* Fort Belvoir, VA: Defense Acquisition University Press, 2001.
3. NASA. *Systems Engineering Handbook.* Washington, DC: NASA, 2007.
4. Luongo, Cesar, Chiang Shih, James Sturges, and David Bogle. "Senior Design Projects in Mechanical Engineering – Active Involvement of Industry Partners and Advisory Council." *ASEE Annual Conference.* Honolulu, HI: ASEE, 2007. 732.
5. Platts-Mills, Oliver, and Reid Bailey. "A Framework for Assessing the Influence of Variations Between Individual Capstone Experiences." *ASEE Annual Conference.* Austin, TX: ASEE, 2009. 1509.
6. Sproles, Noel. "Comming to Grips with Measures of Effectiveness." *Systems Engineering* 3, no. 1 (2000): 50-58.