ASEE
125 YEARS AT THE HEART OF ENGINEERING EDUCATION
2018 ASEE ANNUAL CONFERENCE & EXPOSITION
JUNE 24-27, 2018
SALT PALACE CONVENTION CENTER
SALT LAKE CITY, UT

Paper ID #23087

Design and Development of an Auto-fetch Dog System Using a System Engineering Approach in an Electrical Engineering Master's Capstone Course

Prof. John M. Santiago Jr., Colorado Technical University

Professor John Santiago has been a technical engineer, manager, and executive with more than 26 years of leadership positions in technical program management, acquisition development and operation research support while in the United States Air Force. He currently has over 16 years of teaching experience at the university level and taught over 40 different graduate and undergraduate courses in electrical engineering, systems engineering, physics and mathematics. He has over 30 published papers and/or technical presentations while spearheading over 40 international scientific and engineering conferences/workshops as a steering committee member while assigned in Europe. Professor Santiago has experience in many engineering disciplines and missions including: control and modeling of large flexible space structures, communications system, electro-optics, high-energy lasers, missile seekers/sensors for precision guided munitions, image processing/recognition, information technologies, space, air and missile warning, missile defense, and homeland defense.

His interests includes: interactive multimedia for e-books, interactive video learning, and 3D/2D animation. Professor Santiago recently published a book entitled, "Circuit Analysis for Dummies" in 2013 after being discovered on YouTube. Professor Santiago received several teaching awards from the United States Air Force Academy and CTU. In 2015, he was awarded CTU's Faculty of the Year for Teaching Innovations. Professor Santiago has been a 12-time invited speaker in celebration of Asian-Pacific American Heritage Month giving multi-media presentations on leadership, diversity and opportunity at various military installations in Colorado and Wyoming.

Dr. Jing Guo, Colorado Technical University

Dr. Jing Guo is a Professor in Engineering Department at Colorado Technical University. She is the course director in circuits and electronics area. She taught variety of underrated and graduate courses including capstone design in Electrical and Computer Engineering area.

Design and Development of an Auto-Fetch Dog System Using a Systems Engineering Approach in an Electrical Engineering Master's Capstone Course

John Santiago, Jr., Ph.D. and Jing Guo, D.Eng.
Colorado Technical University (CTU), College of Engineering, Colorado Springs, CO

Introduction

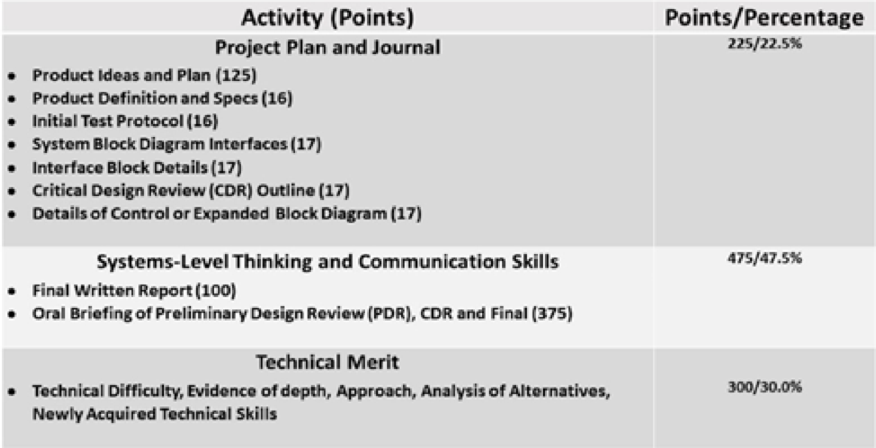
Recently, the College of Engineering (COE) Master of Science in Electrical Engineering program shifted emphasis in the capstone course used for assessment of the program. The shift stresses going from extensive technical detail to the design and development process using a systems engineering approach [1]. The program provides a mix of in-depth technical electrical engineering courses with holistic systems engineering approach to bring their systems into being. Students will have taken two systems engineering courses before registering for the capstone course.

Using the systems engineering approach led to the Auto-Fetch Dog System as proposed by one student. Description of weekly deliverables are presented in another paper to help students practice the systems engineering process [1]. The structure of the capstone course revolved around the Vee Model found in systems engineering. A detailed description of weekly deliverables and rubric for the Critical Design Review are described elsewhere [1]. In addition, the top-down and bottom-up design perspectives are briefly described followed by two perspectives of the Vee-Model. The Vee-Model also provides a holistic perspective of system-level thinking [2] [3] [4]. The proposed Auto-Fetch Dog System provides an illustration of the student's efforts using the systems engineering process [5]. For those who owns dogs, the primary benefit of having an auto-fetch system is that the proposed system attempts to remedy poor dog behavior.

Capstone Course Description

The capstone course offers the student the opportunity to integrate skills developed throughout the graduate program and practice systems thinking by completing a project that focuses on a current issue or need requiring an engineering solution.

The course deliverables listed in Table 1 includes: Project Plan and Journal (22.5%), Communication Skills (47.5%) and Technical Merit (30%). Students must take an ill-defined problem and use a systems engineering approach to implement a proof-of-concept solution. A detailed description of the weekly deliverables is given elsewhere and will not be described here due to space limitations [1]. The Critical Design Review (CDR) rubric was also developed to balance the course weighting between system-level thinking fostered by weekly deliverables and acquired technical skillsets from the MSEE program. The weekly deliverables are guided by the Vee Model [1] [2] [3] [4].



Activity (Points)	Points/Percentage
Project Plan and Journal • Product Ideas and Plan (125) • Product Definition and Specs (16) • Initial Test Protocol (16) • System Block Diagram Interfaces (17) • Interface Block Details (17) • Critical Design Review (CDR) Outline (17) • Details of Control or Expanded Block Diagram (17)	225/22.5%
Systems-Level Thinking and Communication Skills • Final Written Report (100) • Oral Briefing of Preliminary Design Review (PDR), CDR and Final (375)	475/47.5%
Technical Merit • Technical Difficulty, Evidence of depth, Approach, Analysis of Alternatives, Newly Acquired Technical Skills	300/30.0%

Table 1. Measuring Learner Achievement

The Systems Engineering (SE) Approach

In order to promote the systems engineering approach, the COE began by changing the required courses from project management courses to the first two systems engineering courses. Then, the capstone was revised to follow the Vee Model from systems engineering. The perceived benefits of implementing systems engineering concepts include:

- Shift students' focus from technology solution to what system must do (e.g. requirements)
- SE helps with planning, organizing, executing and evaluating a project
- SE provides structure through weekly deliverables to guide students throughout a project from start to finish
- The heavy military presence within the area values systems engineering to meet their defense requirements

Many times, students are engrossed on a final solution or technology when first starting a project. In this case, the students are focused on the '***how***', preferring to apply a technology of interest without consideration of a market or customer need. This initial approach by a student may be viewed as bottom-up thinking.

The shift requires students to focus first on the '***what***' during the initial stage of the project: that is, what functions must the system perform to meet customer needs and requirements. This view of market or demand push is known as top-down thinking. These two rival views are commonly known to industry as: technology push versus requirements pull (or market/demand pull). A systems engineering approach expands student learning by emphasizing a holistic perspective while building upon specialized skills from the other courses.

Electrical and computer engineering students, especially those who are international students, may be unfamiliar with the systems engineering processes. To assure that graduates are truly prepared, the weighting and weekly deliverables allow students to practice systems engineering. A step-by-step and iterative plan helps students expand their perspectives in engineering as systematic processes that deliver the system solution to the marketplace. The structure drives students to shift from a technology-focused solution to one that addresses customer needs. The CoE values both bottom-up and top-down approaches used in combination.

The Systems Engineering Process and the Vee Model [2] [3] [4]

Figure 1 depicts the systems engineering Vee-Model from a testing perspective shown in Figure 1a and from an architecture perspective as shown in Figure 1b [2].

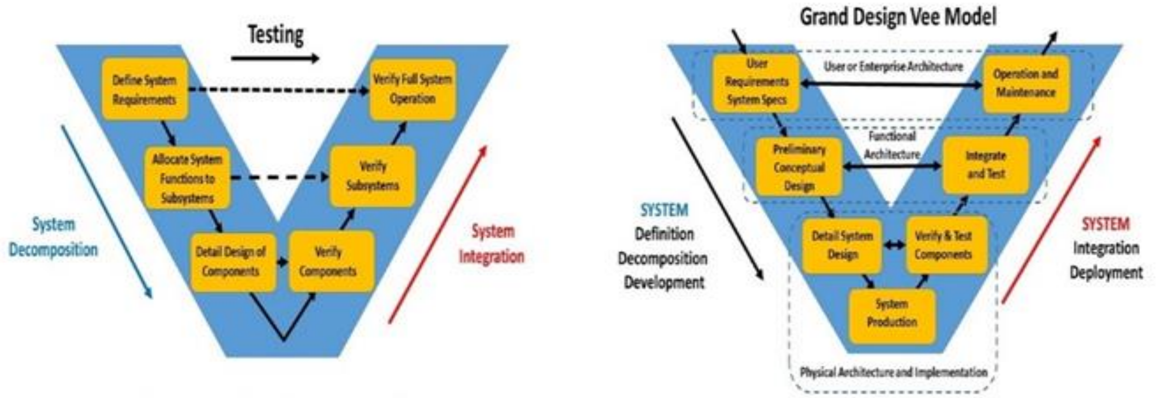


Figure 1. Systems Engineering Process "Vee" Model: Testing [2] & Architecture [4] Perspective

In Figure 1a, the Vee-Model looks at the system, subsystem and component level of testing. The model starts with an identification of user needs on the upper left and ends with a fully system-level acceptance testing and evaluation on the upper right. Advancing down the left side of the Vee-Model is the decomposition of the system into subsystems, and then into components. The technical activities involve defining and resolving the system architecture to mature the design (or definition) of the system with increased fidelity.

The right side of the Vee-model in Figure 1a involves the integration of system components with testing, moving upward to the subsystem and finally the system level. [2] The testing process flows up and to the right as higher levels of subsystems are verified. Finally, the system level is validated through user acceptance testing and evaluation. The user acceptance test plan insures overall specifications are met while testing is performed at all-levels: component, subsystem and system level. The test plan verifies and validates the entire the system in preparation for user acceptance.

The 'Grand Design' and architecture perspective [4] is shown in Figure 1b. The definition of the system begins with '***what'*** functions the solution must perform to define an initial functional architecture. These functions are collected and allocated to a subsystem which further defines the functional architecture. Defining the functional architecture increases the system fidelity and definition during the design stage.

The subsystem functions are further decomposed to identify (or design) physical components (hardware/software) which defines the lowest physical architecture. At this point, the physical architecture describes ***'how'*** the system solution is implemented. Moving up to the upper right of the Vee Model, components are tested and integrated, then subsystems are tested and integrated and up to the full system level. The result is an enterprise architecture, deployable for intended stakeholders.

The following section describes the activities or deliverables using systems engineering for each week. Students are encouraged and advised several weeks before the capstone quarter to begin thinking about their project so they are well-prepared to start quickly. The student must design a new product, and either demonstrate how it behaves or model its performance. Several self-motivated and talented students followed this advice in the past and successfully completed technically challenging projects while following the systems engineering process.

Detailed Description of the Auto-Fetch System Project [5]

One of the authors, an MSEE graduate, recently bought a dog and discovered that he spent several hours trying to keep the dog entertained. This experience with the dog ownership led him to figure out how to keep a dog entertained while the student is away.

The intent of the project is to focus more on practicing the system engineering process than producing the actual product since this is an 11-week course. The student deliverables for this project provided documentation with sufficient engineering details to produce system by another student. The student also successfully tested for the electronic functionally of the system.

Problem Statement and Need. One of the first tasks students need to do after coming up with an idea is to do some market research and market interest about his/her idea. The student referenced the American Society for the Prevention of Cruelty to Animals (ASPCA). ASPCA said that approximately 70-80 million dogs in the U.S., and between 37 and 47 percent of U.S. households have a dog [6]. About 29% of the cases, describe that some of the most common reasons people give away their dogs are: (1) their place of residence does not allow pets; (2) they don't have time to take care of the dogs, or (3) there are issues with how the dog behaves.

Behavioral issues often arise from dogs when they are left alone for extended periods of time. One particularly frustrating behavioral issue for owners is destructive chewing. Chewing is a normal thing which dogs do to help relieve anxiety, frustration, and boredom. However, this behavior can be very frustrating to owners when dogs decide to chew up the furniture, or one's favorite pair of shoes.

Product Description. One of the most popular games owners play with their dogs is 'fetch'. This is an excellent case in which a machine can easily take over the job of the owner while the owner is not home, or otherwise occupied. Such a machine would simply need to be able to accept an object from the dog, and subsequently 'throw' the object. Optimally, the machine could accept objects of various sizes, shapes, and weights, as not everything which can be fetched is the same. This type of machine would provide both mental and physical stimulation for a dog, preventing the boredom which could lead to problem behaviors such as destructive chewing.

Users. Users of this system would consist primarily of dog owners and their dogs. Since the primary user of the system would be a dog, basic interactions with the machine would necessarily have to be very simple, but there could be numerous different modes of operation set by the human user. The machine could have multiple settings for how far to throw the object, as well as a setting where it could throw the object various distances and directions at random, to keep intelligent dogs guessing.

Environment. The system should be able to operate both indoors and outdoors, each environment having unique requirements. Outdoor operations would likely necessitate battery operations, while AC could be used indoors. While the machine would not be used during severe weather, it still would need some level of water resistance due to the possibility of it being left out in the rain. Since the machine's basic function launches objects, safety features must be implemented to limit the possibility of harm coming to persons or property, especially for indoor operations, where breakables are more likely to be encountered.

Subsystem Overview. The Auto-Fetch System will consist of five major subsystem blocks, shown in Figure 2. The first of these blocks is the physical launching mechanism, which will be what actually does the throwing. This will consist of a modified slingshot which can be entirely automated. The second of these blocks is the electro-mechanical system. This will consist of a set of three electric motors, and an electro-mechanical latch (or coupler). The third block is the sensor suite. This will consist of a load sensor, an infrared proximity sensor, and a radio frequency identification (RFID) reader. The fourth block is the user interface. This block will consist of a liquid crystal display (LCD), light emitting diodes (LEDs), pushbuttons, and physical switches. The final subsystem block is the logic circuit. This will consist of a microcontroller such as the Arduino MEGA 2560 (or equivalent).

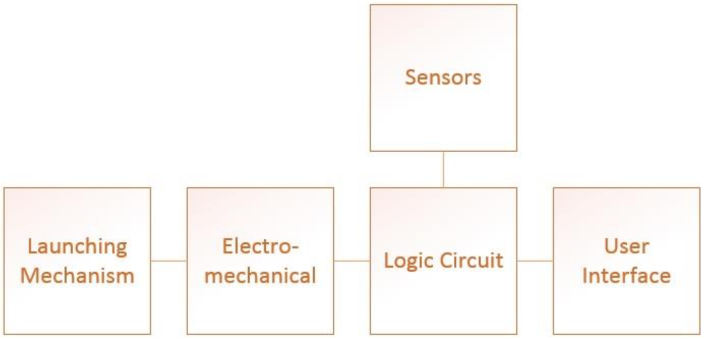


Figure 2. Subsystems Block Diagram [5]

Launch Mechanism. The first question to be answered regarding the launching mechanism of the system, is what method will be used for launching objects? Many methods are available, but some will be better suited for this application than others. The use of rapidly expanding gasses to propel objects (as with firearms) could be used but would be limiting on the type of object which could be launched. Additionally, such a system would require either a separate fuel source such as compressed gas cartridges, or a method of compressing gasses on its own, such as an air compressor. The former would require that the system be routinely reloaded, while the latter would tend to be very noisy.

A purely mechanical method of launch (such as a catapult or slingshot) is a better option, as it would be less limiting on object type, and can easily be run with a simple electric motor. A slingshot-type launch method would allow for variations in projectile travel by varying launch angle as well as draw length. The main question for this type of launch method would be how to set up the mechanism to be used for automatically resetting the system after projectile launch.

Linking the launching mechanism to the other blocks of the system would occur through electric motors which are controlled by the logic circuit. These electric motors would be used to set the launch angle and direction, as well as the draw and reset of the launcher.

The physical launching mechanism shall be an adjustable platform from which varying objects can be thrown. The platform shall be capable of inclining to an angle of 45 degrees from horizontal. It shall turn up to 45 degrees to either side of center. The energy for launching projectiles shall be provided by the rubber bands of a slingshot. These rubber bands shall be attached to a sort of 'launching basket' which will hold objects for launch.

Electro-mechanical System. The electro-mechanical system is what directly controls the physical launching mechanism. The first of three electric motors shall control the platform's angle for launch. It shall raise the platform to 45 degrees when the system is preparing for launch and shall lower the platform to 0 degrees when the system is waiting for an object to be loaded into the launching basket. The second electric motor shall control the platform's direction of launch.

During normal operation, the system will launch objects straight forward (that is, 0 degrees from center). When the system is set to random direction operation, the second electric motor shall turn the platform anywhere between -45 degrees and +45 degrees from center, prior to each launch.

The third electric motor shall control the slingshot draw length. It shall pull the launching basket back by the appropriate amount, so that the object will travel the desired distance when released. The electro-mechanical latch shall control release of the launching basket. When electric motor three pulls the launching basket back to the proper draw length, the latch shall release the launching basket, allowing the stored elastic potential energy in the rubber bands to be rapidly transferred to the projectile as kinetic energy. After launch, the latch shall re-engage, and once again connect electric motor three to the launching basket.

Sensor Suite. The sensor suite is what allows the system to determine when an object is ready to be thrown, whether the object is safe to be thrown, and how the object should be thrown. The load sensor shall determine a loaded object's mass, so that this information can be used in the calculations to determine the proper slingshot draw length. The infrared proximity sensor shall notify the system when an object has been loaded into the launcher basket, thus initiating the launch sequence. The RFID reader shall notify the system when an approved RFID tag has or has not been detected inside an object which has been loaded. This allows for a 'safe mode' which may be engaged by the user, so that only objects containing approved RFID tags will be launched. Approved RFID tags may contain additional information about the object in which they reside (such as object mass), allowing the RFID reader to act as a backup load sensor as well.

User Interface. The user interface is what allows the user to change system settings and observe the current system settings. The LCD shall inform the user of current distance and direction settings, as well as the presence or absence of an object which has been loaded into the launching basket. The LCD shall additionally display information about an object which is loaded that has an approved RFID tag imbedded. The LEDs shall indicate when 'safe mode' is on or off (green for on, red for off). One physical switch shall turn the system power on and off, while the other switch shall turn 'safe mode' on and off. One pushbutton shall cycle through distance settings, while the other pushbutton shall cycle through direction settings.

Numerous options for a user interface could be implemented, but as with the launching mechanism, some are more suitable than others. A touch screen interface is one option but might be a bit much considering the relatively few settings which the system can make use of. Settings such as distance could consist of labeled LED's, with a single button used to cycle through options. The RFID safe mode setting could work as a simple toggle switch which could flip the mode on and off.

As with the sensors, the user interface would be linked to the rest of the system as input to the logic circuit, though it would also show output from the logic circuit in the form of indicators, such as LED's.

Logic Circuit. The logic circuit is the heart and brain of the system. It shall take sensor inputs to determine when an object is loaded and whether the object is safe to launch. Sensor inputs shall also be used to calculate the proper slingshot draw length to fire an object a prescribed distance. The logic circuit shall additionally take user inputs to set the desired launch distance and direction. These user inputs to the logic circuit shall also include the ability to turn the RFID 'safe mode' on and off. The logic circuit shall send output signals to run the LCD and LEDs, giving information to the user regarding current settings and state of the system. The logic circuit shall also send control signals to each of the individual electro-mechanical system components to run the physical launching mechanism.

Launcher Calculations. The rubber bands of a slingshot store elastic potential energy like springs, but unlike normal springs, the spring stiffness of a rubber band is not constant. This is because unlike normal springs, in which there is a linear relationship between force and displacement, rubber bands have a nonlinear relationship between force and displacement. For a nonlinear spring, a function for force versus displacement must be determined by measuring the force required to stretch the spring at several different displacement values. Curve fitting can then be used to estimate the force function.

For this project, the force required for a given displacement was measured at several values for a common, handheld slingshot. The measured values are shown in Table 2.

Displacement (in. / m)	Force (N)
1.0 / 0.0254	8.624
2.5 / 0.0635	18.13
5.0 / 0.1270	30.576
7.5 / 0.1905	39.886
10 / 0.2540	47.824
12.5 / 0.3175	55.468

Table 2. Measured results of Handheld Slingshots [5]

These values were then used to fit a power curve function. The results of the curve fitting are shown in Figure 3.

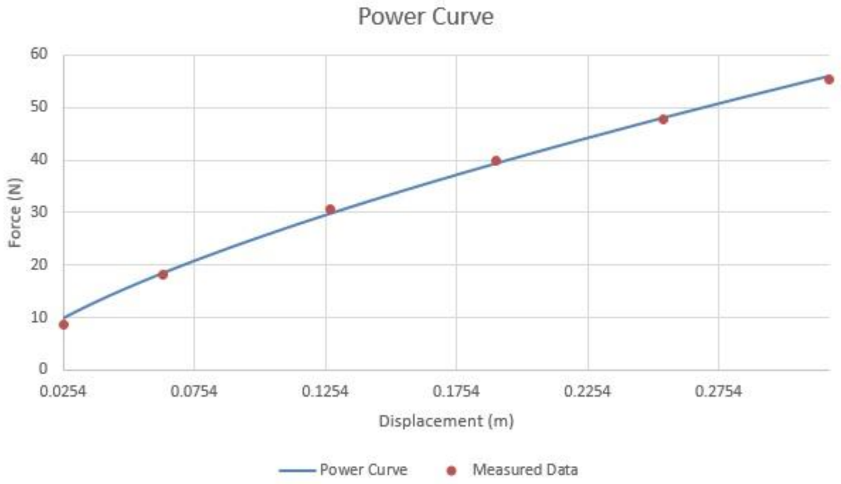


Figure 3. Curve Fitting of Experimental Data with Force Displacement Function

The force versus displacement function F(x) derived from curve fitting the measured data is defined as:

$$F(x) = 123.82x^{0.692}$$

Since the force versus displacement is not linear, as with an ideal spring, the spring stiffness does not follow Hooke’s law. Instead, the spring stiffness k, of a nonlinear spring can be found by taking the derivative of the force function [7]. The elastic potential energy PE, then, is the integral of the force function evaluated from zero to x.

$$k = \frac{dF}{dx} = 85.6834x^{-0.308}$$

$$PE = \int_0^x F(u)\,du = 73.18x^{1.692}$$

The remainder of the launcher calculations are just a matter of simple projectile physics. The student developed the Matlab code to predict the required slingshot draw length, given a desired projectile travel distance, projectile mass, and a given launch angle.

The intent of the project is to focus more on practicing the system engineering process than producing the actual product since this is an 11-week course. The student deliverables for this project provided documentation with sufficient engineering details to produce system by another student. The student also successfully tested for the electronic functionally of the system.

Design Tradeoffs [5]

The student was asked to seek alternatives when designing their engineering solution at the system, subsystem and component level. Here are sample views from the student into this tasking.

Launch Method. There are numerous methods for launching objects of various types. One of the first that comes to mind is that of rapid gas expansion, like that employed by firearms. While this method can certainly be effective, it does not fit well with the requirements for this system. A rapid gas expansion launch typically requires some type of tube which a tightly fitting object is fired through. This would not work for firing objects of varying sizes and shapes, such as dog toys.

Mechanical methods of launch are more conducive to the requirements for an auto-fetch system than rapid gas expansion. One mechanical method which could be used is that of a catapult. While this method can accommodate objects of varying sizes and shapes, it cannot be easily adjusted for different object mass, or changes in launch distance.

Another mechanical method is that of a slingshot. Slingshots can accommodate differing sizes and shapes of objects, much like catapults. Unlike a catapult, however, a slingshot can easily be adjusted for distance or object mass simply by changing the draw length of the slingshot. The auto-fetch system will thus use the slingshot method for object launch.

Electric Motors. The power for the system is electrical and mechanical, with the electrical power being supplied either by AC grid power, or DC battery power. This electrical power provides the energy to run the logic circuit, sensors, user interface, and electric motors. The mechanical power is supplied by the electric motors which run the launching mechanism. Three motors will be required. The first will control the direction the system is facing to determine the direction of launch. The second will control the angle of launch, determining the height and distance that projectiles will reach. The third motor will control the draw length, helping to determine height and distance of projectile travel, and will also reset the launch mechanism.

The electric motors serve to connect the physical launch mechanism to the logic circuit, and thus the rest of the system.

The main thing to be determined about the electric motors in the system is the best type of motors for the job. Standard DC motors have the benefit of high power and low cost, but lack precise position control. Typical DC motors are not a good choice for this application, because precise position control is a must.

Stepper motors provide precise position and speed control, as well as high torque at low speed, but they are very energy inefficient [8]. A stepper motor holding position uses more current than at any other time, and the motors for this system will spend a great deal of their time holding position with no load. This issue can be at least somewhat overcome, however, by integrating a controller which will limit the current draw of the steppers during periods of inactivity. Stepper motors can simply be commanded to go a specific number of steps, allowing for precise positioning, but this control is considered 'open loop,' which can cause some issues. If for some reason the stepper motor misses one or more steps, it has no way of knowing this. To overcome this, an encoder can be integrated with the stepper to check and see if it reached its commanded position after it finishes, allowing the loop to be somewhat closed.

Servo motors combine precise position control, good power, and high energy efficiency [9]. Unlike steppers, which require maximum current to hold position, servo control only provides the motor with just enough current to either move or hold position. Additionally, servo motors use proportional control, so that the speed of the motor is proportional to the difference between the motor's current position, and the commanded position. This means that a servo motor only runs as fast as it needs to, making it very efficient. They tend to be somewhat more expensive than stepper motors, but the benefits may outweigh the cost in this case. Servo motor control is 'closed loop,' as the position of the shaft is constantly monitored, so that if the load on the motor changes, extra current can be applied to compensate in real time.

Due to the differing requirements of the motors, the auto-fetch system will use both servo and stepper motors. The direction and launch angle controlling motors can both be stepper motors, as both will have relatively light, constant loads, and are thus unlikely to miss steps. For the motor controlling the slingshot draw length, a servo motor is the better choice. When the slingshot is being drawn back, the load is constantly changing. The servo motor control will recognize this, and compensate for the change, ensuring that the slingshot is quickly and accurately drawn to the commanded position.

Sensors. The system will make use of sensors to determine when an object to be launched is present and determine whether the object is safe enough to be launched. The main question in this regard is how will a sensor determine if an object is safe for launch? One way to determine the relative safety of an object to be launched is by checking the object's weight. This is, of course, because heavier objects tend to make for more destructive projectiles than lighter objects. For more certainty, a set of "approved" objects could be used. Such objects would include an imbedded passive RFID tag. The system would include an RFID reader, which would allow any object with an "approved" RFID tag to be launched, while rejecting any object without such a tag. This RFID system could potentially be implemented as an optional "safe" mode which could be enabled or disabled at the discretion of the user.

The sensors would be linked to the rest of the system as input to the logic circuit, which in turn provides control signals to the motors controlling the launching mechanism. For object detection, the two obvious choices are ultrasonic sensors, and infrared sensors. Ultrasonic sensors can provide very accurate ranging information, and since they use sound instead of light, they are not adversely affected by direct sunlight [10]. Accurate ultrasonic sensors do tend to be more expensive than infrared sensors, however.

Infrared sensors do have the capability of providing accurate ranging information, but they run into problems when used in direct sunlight [10]. Additionally, since light reflects differently off different surfaces and different colors, the range reading can differ between two objects which are the same distance away from the sensor. Some very simple infrared sensors will simply give a binary output, rather than a range, for any object in view. These sensors are very cheap, and this system only needs a binary reading, not an accurate range, so a simple infrared sensor is the best choice for the auto-fetch system.

Cost Estimate [5]

The cost of the system will be an important factor to determine the marketability of the auto-fetch system. Table 3 shows estimated costs of the components for the system. Note that these are rough estimates of retail prices, not wholesale prices.

Description	**Price Estimate**
Servo motor (x3)	$30
Electromechanical latch	$10
2560 MEGA	$15
RFID reader	$30
IR sensor	$5
Load sensor	$10
LCD screen	$10
Assorted small electronic components	$5
Assorted physical components	$50
Grand Total	$165

Table 3. Cost Estimate of Proposed Auto-Fetch System [5].

Existing automatic fetch machines on the market retail between $115 and $180 on Amazon. The auto-fetch system retail estimate is between these and offers features that existing machines do not. Given these facts, it appears likely that the auto-fetch system can be competitive in the market.

System Operation [5]

The student learned how to use auto-cad software to illustrate a prototype model of the auto-fetch system. The auto-fetch system prototype in Figure 4 illustrates launching a tennis ball at 30 degree angle.

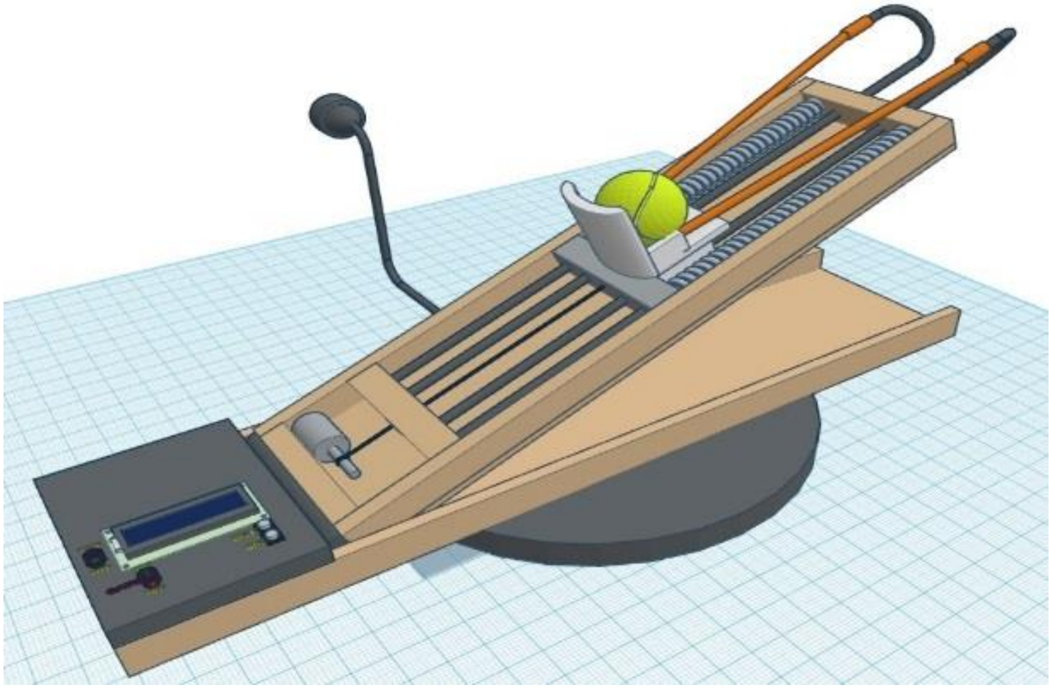


Figure 4. Prototype of Auto-Fetch System Illustration

When idle, the launcher lies flat until an object is detected. When an object determined to be ‘safe’ has been detected, the launcher reaches the proper angle and the launcher basket is pulled back by the servo motor. After the object has been launched, the launcher is then lowered back to the idle position, and the system is ready for the next launch.

Table 4 lists the system requirements for the Auto-Fetch System and Table 5 lists the procedures for user acceptance test. In Table 4, each system requirement has a description that takes into account the launch direction and distance selected by the user, detection of a missing loaded object, safety features and water resistance.

In Table 5, developed a user acceptance plan to verify and validate the system requirements. For each test, the procedure and proper system response is described.

As part of the electronic journal, the student gave several status reports to update Professor xxx on key project activities. Status reports (progress on tasks, schedule, and costs) were used to provide more student accountability as well as providing added realism found in a work environment.

#	Requirement	Description
1	System shall accept user input for launch distance	User interface shall include distance setting
2	System shall accept user input for launch direction	User interface shall include direction setting
3	System shall accept object to be thrown	System must include a place where users can load objects
4	System shall accept varying types of objects for throwing	System must allow for objects of varying shapes, sizes, and weights within specified tolerances
5	System shall detect loaded object	A sensor must notify system of loaded object’s presence
6	System shall launch loaded object a user-specified distance	Distance may be specific, or a range of distances
7	System shall launch loaded object a user-specified direction	Direction may be specific, or a range of directions
8	System shall detect missing object	A sensor must notify system of loaded object’s absence
9	System shall abort launch when object is missing	Object removal must cause system to abort launch
10	System shall have a delay between object detection and launch	Delay must give user time to get clear after loading object
11	System shall detect potentially unsafe loaded object	A sensor must notify system of potentially unsafe object presence
12	System shall abort launch of potentially unsafe loaded object	Unsafe object presence must cause system to abort launch
13	System shall operate on AC power	As described in requirement
14	System shall operate on battery power	As described in requirement
15	System shall be water resistant	Water resistance must prevent damage to system from mild precipitation when left outdoors

Table 4. System Requirements [5]

TEST	PROCEDURE	PROPER SYSTEM RESPONSE
OBJECT ACCEPTANCE	Object is properly placed in the machine	Object remains in place until it is thrown
OBJECT DETECTION	Object is placed in the machine	Machine indicates that an object has been detected
MISSING OBJECT DETECTION	Object is removed from the machine after being detected	Machine indicates that object is no longer detected
THROW DELAY	Object is placed in the machine	Machine waits a specified time before throwing object
THROWING OPERATION	Object is placed in the machine	Machine throws object as expected after specified delay
THROW ABORT (MISSING OBJECT)	Object is removed from the machine after being detected	Machine aborts throwing procedure
UNSAFE OBJECT DETECTION	Object out of acceptable tolerances is placed in the machine	Machine indicates that object is deemed unsafe
THROW ABORT (UNSAFE OBJECT)	Object out of acceptable tolerances is placed in the machine	Machine does not throw object
DISTANCE INPUT	System is set to throw object a specified distance	System indicates proper distance specification
DISTANCE VERIFICATION	Machine is set to a specific distance setting, then object is placed in the machine	Object travels the specified distance before impacting the ground, within tolerances
DIRECTION INPUT	System is set to throw object in a specified direction	System indicates proper direction specification
DIRECTION VERIFICATION	Machine is set to a specific direction setting, then object is placed in the machine	Object is launched in the specified direction, within tolerances
OBJECT DIVERSITY	Objects of varying sizes and weights (within tolerances) are placed in the machine	Machine throws each object as expected
DISTANCE RANDOMIZATION	Machine is set to a range of values distance setting, then object is placed in the machine to be thrown several times	Machine throws object apparently random distances within the specified range of values
DIRECTION RANDOMIZATION	Machine is set to a 'random' direction setting, then object is placed in the machine to be thrown several times	Machine throws object apparently random directions
RANDOM OPERATION	Machine is set to 'random' setting, then object is placed in the machine to be thrown several times	Machine throws object apparently random distances and directions
BATTERY OPERATION	Machine is unplugged from AC power, then object is placed in the machine	Machine throws object as expected
AC OPERATION	Machine is plugged into AC, battery is removed, then object is placed in the machine	Machine throws object as expected
WATER RESISTANCE	Machine is placed in simulated rain conditions for a specified time	Machine is undamaged, and continues to operate normally

Table 5. Test Plan for User Acceptance [5]

Figures 5 and 6 shows photo documentation of the hardware setup to test the system components. The Arduino microcontroller is programmed to detect an object and follow user inputs with regards to launch angle (distance) and direction as well as other testing an meeting requirements found in Tables 4 and 5.

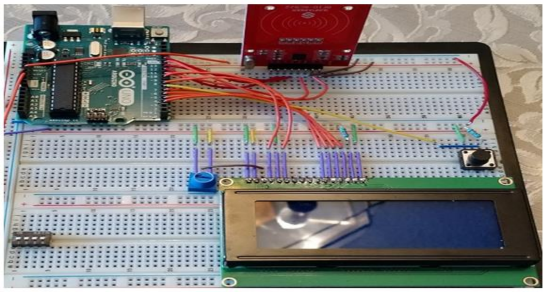


Figure 5. LCD and Arduino testing [5]

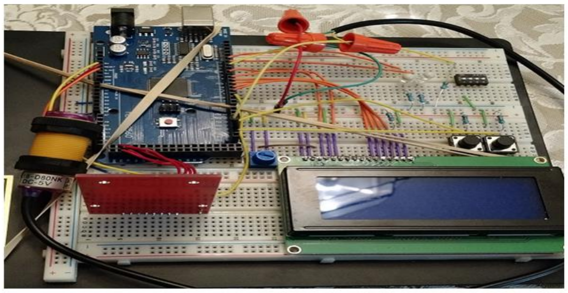


Figure 6. RFID, LCD and Arduino Testing [5]

Conclusion

The CoE successfully incorporated systems engineering concepts into a graduate capstone course in electrical engineering. The approach expands the student experience by providing them with a more holistic view of engineering. When students encounter a complex problem, they can use systems thinking. This was accomplished by having the student think more at the system level and from a variety of perspectives to gain insight into the solutions while investigating market interest. The student presented the solution communicating a message having an economic and societal value. The system-level thinking builds upon the specialized technical skills from other courses found in the CoE's graduate engineering programs as well as learning new skills. The student acquired new skills, such as curve fitting and CAD software while making connections and leveraging technical skills learned from the graduate program found in Table 1. Cost estimates were comparable to existing systems but with more features. The capstone course attempts to provide a right mix of technical merit, systems engineering thinking, and improved communication skills [1].

References

[1] Santiago, John and Jing Guo, "A Systems Engineering Approach for Implementing An Electrical or Computer Engineering Master's Capstone Course," in *ASEE Rocky Mountain Section Conference*, Cedar City, Utah, 2016.

[2] S. B. Blachard and W. Frabrycky, Systems engineering and Analysis" (Fifth Edition), New York: Prentice Hall, 2011.

[3] D. Buede, The Engineering Design of Systems – Models and Methods (Second Edition), Wiley, 2009.

[4] A. P. Sage and J. James E. Armstrong, Introduction to Systems Engineering, Wiley, 2009.

[5] yyy, *Information retrieved from student deliverables for EE 692, Electrical Engineering Capstone,*, 2015.

[6] ASPCA, "Shelter Intake and Surrender," 2015. [Online]. Available: https://www.aspca.org/animal-homelessness/shelter-intake-and-surrender/pet-statistics. [Accessed 23 October 2015].

[7] B. Fineo, "Linear & Nonlinear Springs Tutorial," 2015. [Online]. Available: http://www.sciencebuddies.org/science-fair-projects/project_ideas/Physics_Springs_Tutorial.shtml#introduction. [Accessed November 2015].

[8] Adafruit, "What is a Stepper Motor?," 13 November 2015. [Online]. Available: https://learn.adafruit.com/all-about-stepper-motors/what-is-a-stepper-motor#. [Accessed November 2015].

[9] Advanced Micro Controls Inc., "Stepper vs. Servo," 2015. [Online]. Available: http://www.amci.com/tutorials/tutorials-stepper-vs-servo.asp. [Accessed November 2015].

[10] Society of Robots, "Infrared vs. Ultrasonic - What You Should Know," 27 January 2008. [Online]. Available: http://www.societyofrobots.com/member_tutorials/node/71. [Accessed November 2015].