

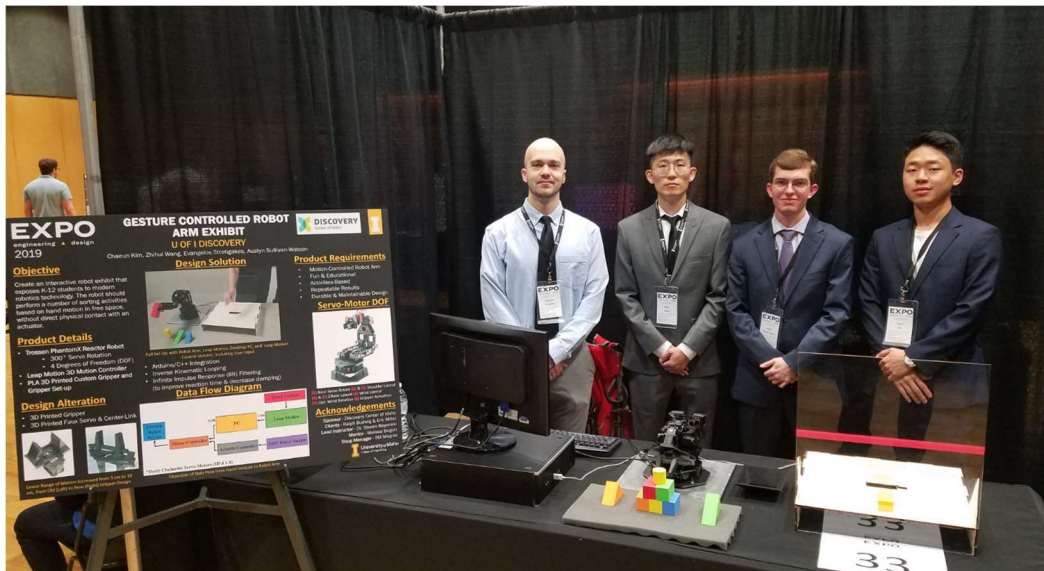
Team U of I Discovery

With The



DISCOVERY

Center of Idaho



Gesture Controlled Robotics Exhibit

FINAL DESIGN REPORT

Zhihui Wang: wang6162@vandals.uidaho.edu

Chaehun Kim: kim7885@vandals.uidaho.edu

Angelo Stratigakes: strat8803@vandals.uidaho.edu

Austyn Sullivan-Watson: sull1271@vandals.uidaho.edu

Table of Contents

Executive Summary	3
Background	3
Problem Definition	5
Project Plan	6
Concepts Development	9
System Architecture	12
Design Evaluation	15
Conclusions	22
References	23
Appendix	24

Executive Summary

The U of I Discovery Senior Capstone Design team, in conjunction with the Discovery Center of Idaho, present the project documentation for the DCI Gesture Controlled Robotics Exhibit. This project, involves the use of a 3D motion activated controller in the Leap Motion Device, and a Trossen PhantomX AX-12 Reactor Robot Arm. The Discovery Center provided a list of product requirements, that include the need for a compact, transportable and maintainable design, that is fun, engaging, and activity based. By including these features, the Discovery Center will be able to bring motion activated control to the museums constant stream of visitors, for many years to come.

The robot arm contains four degrees of freedom, with rotational joints acting at three different points, base rotation around a fixed point, and gripper functionality. The hope is that the robot captivates users and onlooker's attention, in the hopes of sparking interest with the possibilities that lie ahead with the future of robotics technology.

The solution used to complete the Discovery Center's need was to connect and configure the leap motion device, to the robot arm, so that the transfer of information was immediate, and responsive. To do this, many steps were taken, including designs of designated system diagrams, system set ups, and code data flow diagrams.

The need for such a device at the Discovery Center is high, as robotics technology and advancements within the niche, have enticed interest by many to think towards what the future of robotics holds. Robots are already being used for surgery, like the da Vinci Robot, though instead of being motion actuated by a user, the Discovery Center's device thrives off of physical user input. With the introduction of the motion controlled device, the hope is to provide individuals at the exhibit an entertaining exhibit, that they will find both educational, and fun.

Based off of the introduction of the device, the overall response has been positive, while usability has also been mostly positive. With a few minor tweaks to be made prior to final submission of the project to the Discovery Center, the robot ran continuously, with small breaks between users.

Background

The Discovery Center of Idaho's main motivation as the sponsor of this project was to complete a prior University of Idaho Capstone Project from the 2017-2018 school year. The project was inspired through the thought of incorporating an interactive activities based exhibit for K-12 students to enjoy at the Discovery Center. With this motivation, the plan has been to implement such feature within an exhibit set up by the summer or fall of 2019.

Motion activated robot technology is needed within the southern Idaho area, as it isn't readily available for individuals of the public. By bringing such technology to the forefront, more individuals will be impacted by the future of robotic interfaces, at the tip of their fingers.

The benefit of this device is entirely positive, from the user's standpoint, and the hope is to not only incorporate a motion controlled device, but provide an individual learning experience, for

each user. By allowing individuals to do what they want, with an extension of one's self, in the robot arm, the experience will be repeatable for everyone, whether you are 4 years old, or 50 years old. The device is user friendly, and responsive to any motion within the 2 cubic foot volume. The only downside to the device, is the need for users to follow simple directions to activate and work the device, which will need to be followed to an extent, to provide positive results to the user. The hope is that in the future, the device can be expanded to a larger format, with the possibility of adding another degree of freedom, into the already existing structure.



Figure 1: General Layout of Exhibits at the Discovery Center of Idaho in Boise.



Figure 2: Young Students captivated by the Robot Arm Exhibit at the University of Idaho's XXVI Annual Engineering & Design EXPO.

Problem Definition

The Discovery Center of Idaho is looking to incorporate interactive exhibits that expose K-12 students to modern robotics technology. This project features a four degree of freedom robot arm that can perform a number of sorting activities based on hand motion in free space, without direct physical contact with an actuator. The goal is to deliver a functioning robot arm that takes user input, and turns it into an extension of one's self.

Specifications & Constraints

The robot arm will be limited in some aspects, as far as range of motion, extension, contraction, strength and overall movement. The robot arm is limited, due to its size, though what it lacks in size, it overrides with ease of usability. The robot arm has a specified Vertical extension of approximately 55.5 cm, and Horizontal Extension of 43 cm.

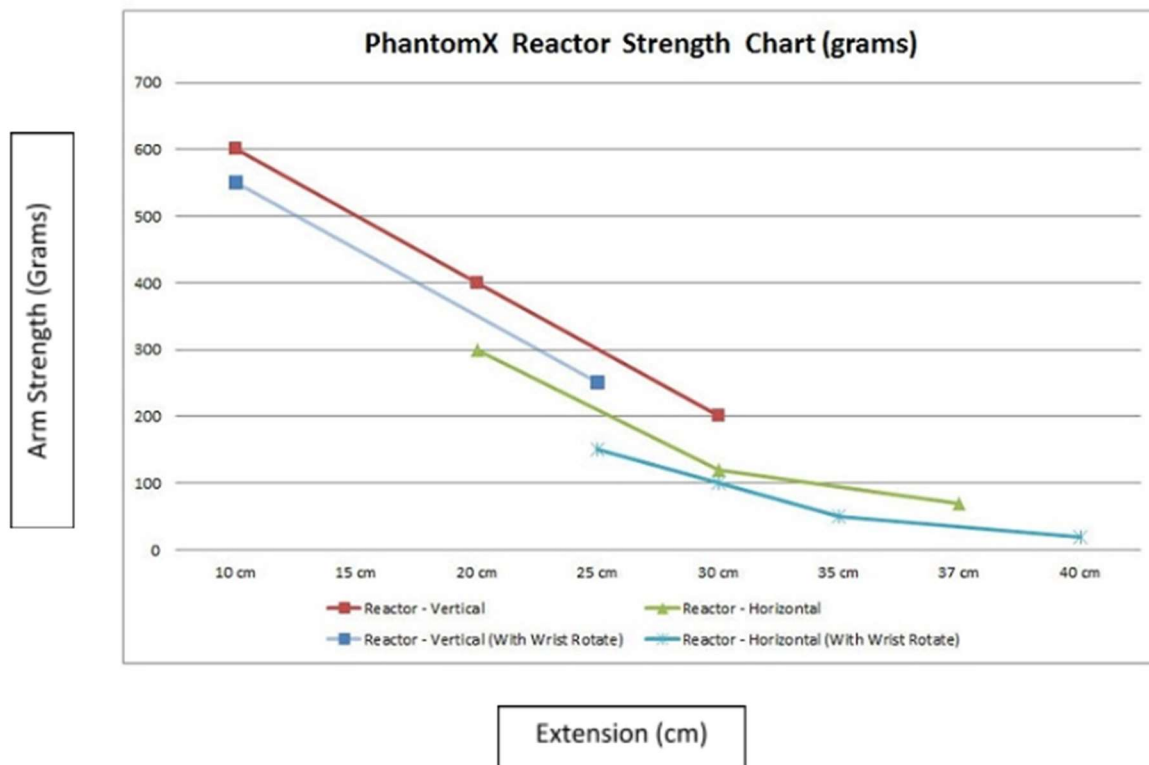


Figure 3: Trossen PhantomX Reactor Strength vs. Extension Chart

In Figure -- the Arm Strength vs. Arm Extension is shown for the PhantomX Reactor Robot Arm. The Vertical direction is defined as the longitudinal motion in the y-direction, while Horizontal is defined as the latitudinal motion in the x-direction for the robotic arm. If the wrist rotation is ignored, and we look at both the red and green lines with square and triangular points respectively, the arm performs better without the wrist rotation addition. Figure - and - depict the actual approximate values for the performance of the robot arm based on its overall extension displayed on the graphic in Figure -.

Table 1: Vertical Extension of Trossen PhantomX Reactor Robot Arm w/ specified lengths and load carrying capacities.

Vertical Extension (cm [in])	Max Specified Mass Carrying Capacity (g [lbm])	Load Carrying Capacity (N [lbf])
10 [3.9370]	600 [1.2125]	5.3955 [1.2125]
20 [7.8740]	400 [0.7716]	3.4335 [0.7716]
25 [9.8425]	300 [0.5112]	2.4525 [0.5112]
30 [11.811]	200 [0.4409]	1.9620 [0.4409]

The performance for the vertical extension of the robot arm decreases as extension increases, which is depicted within *Table 1*. The same performance issues found with vertical extension can also be seen within horizontal extension. Though these values can be exceeded, for best performance, the load at any time within the gripper should not exceed 6.67 N (1.5 lbf). However, this will most likely not be an issue due to the plans for the actual exhibit, involving foam blocks and other games/activities that will keep the user interested.

Table 2: Horizontal Extension of Trossen PhantomX Reactor Robot Arm w/ specified lengths and load carrying capacities.

Horizontal Extension (cm [in])	Max Specified Mass Carrying Cap. (g [lbm])	Load Carrying Capacity (N [lbf])
20 [7.8740]	300 [0.6614]	2.9430 [0.6614]
25 [9.8425]	200 [0.4409]	1.9620 [0.4409]
30 [11.811]	160 [0.3527]	1.5700 [0.3527]
35 [13.780]	150 [0.3307]	1.4720 [0.3307]
40 [15.748]	25 [0.0551]	0.2453 [0.0551]

Further specifications on the product can be found within the User/Owners Guide, which is a supporting document located on the flash drive.

Project Plan

During the duration of the project, the main concern was accomplishing a final goal of a hands free, motion activated robot. By using the strengths of both the mechanical engineering and computer science students, the project was consistently improving, daily, throughout the semester.

Discovery Center of Idaho Project Planner

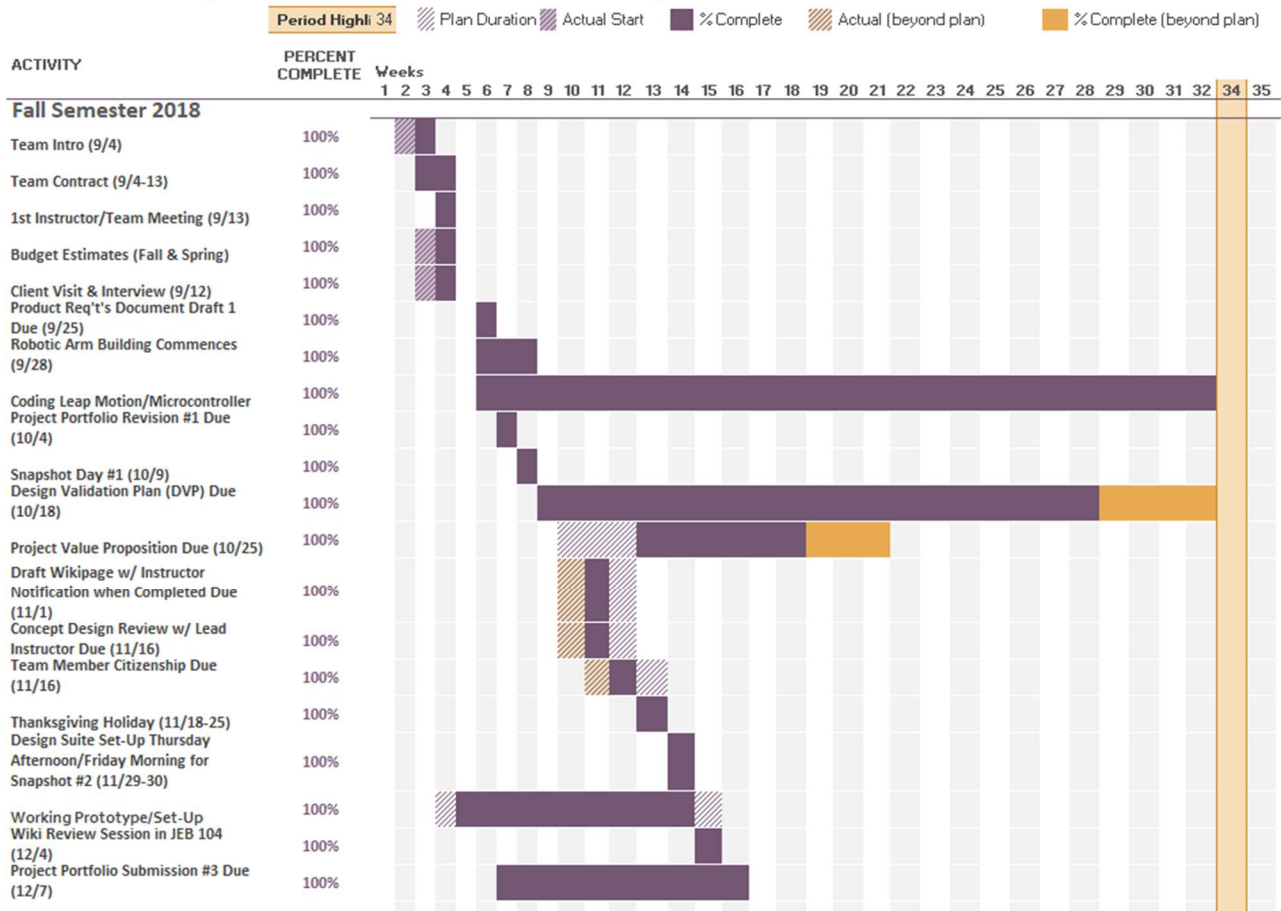


Figure 4: Fall 2018 Semester Schedule Overview

Throughout the first semester of the project, the group had several tasks to take care of. Whether it be team/instructor/client meetings, project documentation or hands on project work, several milestones were achieved throughout the semester. Including robot arm selection, research and development, the first semester of work was more or less a time for the group to build the robot arm, and test different system setups, in the hopes of formulating a solution to motion activated control. With the help of the ArbotiX X-Bee v2.0 Commander Controller, code development was sparked into action. Though immediate results weren't prominent, the overall structure provided through the basic Arduino code, supplied enough for the Computer Scientists within the group to write up an accompanying program that would decipher the data provided from Leap Motion, to the Robot Arm. However, this didn't come into fruition until the Spring Semester, where data analysis and testing was a larger focus.

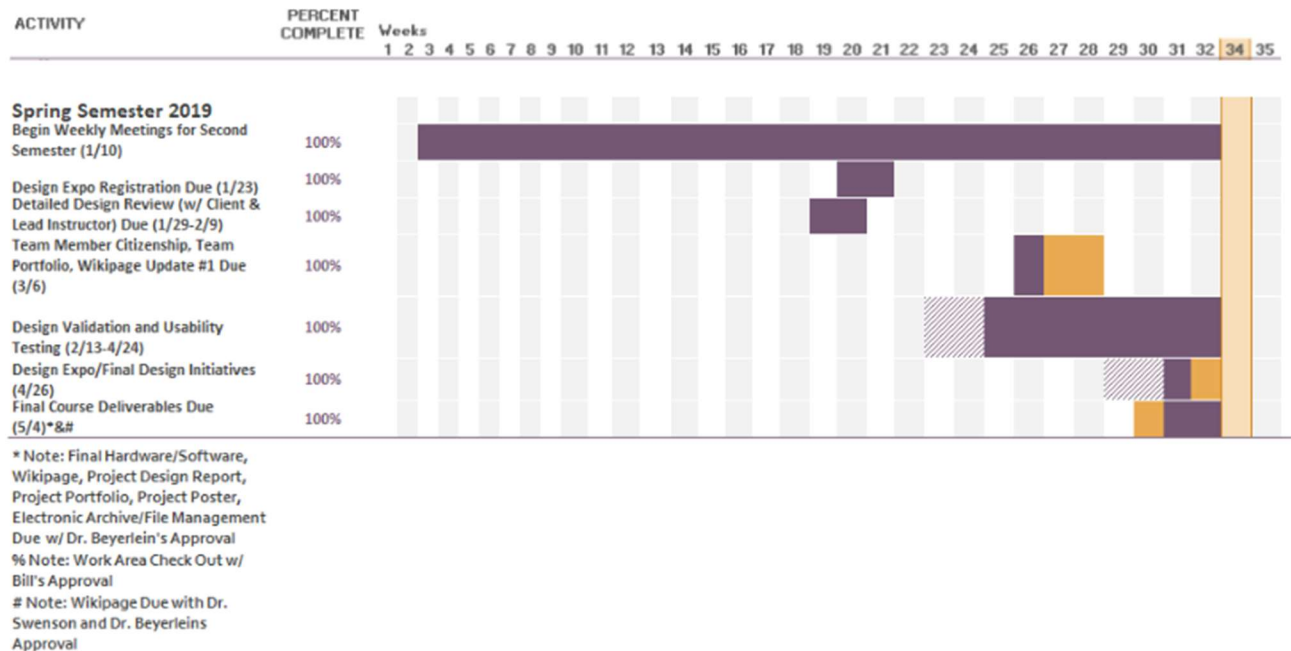


Figure 5: Spring 2019 Semester Schedule Overview

Like I previously stated for the Fall Semester, the Spring Semester involved more code development and usability testing for the robot arm. In regards to the schedule, the overall project isn't totally represented well from the milestones completed by the U of I Discovery team. Design Validation was prominent throughout both Fall and Spring Semesters, though it was more applicable within the Spring Semester.

Throughout both semesters, the design stayed relatively the same, with some upgrades, and restructuring. However, the overall functionality of the robot arm increased over time, due to the advancement in knowledge from different resources, and testing. By far the most challenging aspects of the project were finding code to manipulate, and repurpose for our use within the project. Once the code for the Arduino was discovered, the overall introduction of C++ code was almost formatted directly to fit the same basic structure of the Arduino, though it was run primarily for the Leap Motion Control.

The mechanical engineering students within the group Austyn Sullivan-Watson and Zihui Wang, made it their responsibility to take care of any maintenance, part modeling, or feature driven aspects of the project. With the initial build beginning in early September, part alteration and modification, via mainly 3D printing, occurred throughout the remaining months, culminating in a final design that will promote minimal maintenance, and ease of part replacement.

The computer science students within the group, Chaeun Kim and Evangelos Stratigakes, took on the responsibility of coding, and incorporating any software improvement to the project. In so doing, the pair was responsible for finding solutions to communication between both the Leap Motion controller and Robot arm, which were the main priorities being strived for completion over the course of the projects entirety.

Concepts Development

Given the advice of purchasing a robotic arm kit, rather than take away from the time that we could be using to get the robot working via Leap Motion, we wanted to make sure that our configuration would work best for the given situation. Given the advice of the lead instructors, Dr. Steven Beyerlein, Dr. Joel Perry, and Dr. Ralph Budwig, the group decided to purchase a base model robot arm kit that would be suitable to complete the objectives laid out by the Discovery Center. The decision making process was completed using the thought process of longevity, reliability, and cost as the primary factors. Several candidates were in the running for purchase, though two truly stood out for the features that they both offered. The choice came down between the Trossen PhantomX Reactor Robot Arm, and Trossen WidowX Robot Arm Comprehensive kit.

Trossen Robotics PhantomX Reactor Robot Arm Kit - Price: \$649.95 w/ Wrist Rotation

This configuration allows for 5 Degrees of Freedom, with the wrist rotation addition, and runs with seven AX-12A Servo-Motors, which are inexpensive compared to that of the MX-Series Servo-Motors, that contain heavier duty gearing. The importance of using all of the same servos, is that they are interchangeable at any position within the system. Regarding the free motion, the arm's base has a maximum rotation of 300-degrees. This option was also provided in the project document, that was used to promote the project.

Trossen Robotics WidowX Robot Arm Comprehensive Kit - Price: \$1699.95

This configuration allows for 5 Degrees of Freedom, with an extra pivot point within the arm setup. The WidowX is comparable to the PhantomX, though it does have higher-torque driven Servo-Motors in the MX-64T and the MX-28T Series. The WidowX uses 6 total servos, two MX_64T's, two MX28T's, and two AX-12A's, which operate at different locations within the arm, and supply better resolution regarding arm movement, than that of the PhantomX. Also, the WidowX's base arm rotation is not limited like the PhantomX, as it has the ability to rotate 360-degrees.

After several comparisons, both robotic arms are quite similar. They both have the maximum Degrees of Freedom without spending an atrocious amount on a medically driven robotic arm. Both arms have similar size, can both be adjusted to use 3D-printed parts, come with firmware, and use the same Arbotix Robocontroller. Also, both robot frames come with 14 cm ball bearing bases.

The biggest difference, is price and can be explained by the cost for the four MX-Series servo-motors, which cost between \$220-\$300 each, while the AX-Series costs approximately \$45 each. This is the explanation for the \$1000 difference. Another difference is that the total rotation for both, being that the PhantomX can rotate 300-degrees, while the WidowX rotates 360-degrees, freely. Another difference is found in the shoulder and elbow designs, the PhantomX uses a dual servo design, while the WidowX uses heavier duty servos. The great thing about the dual servo design, in the shoulder and elbow of the PhantomX, is that the amount of use that this robotic

arm will receive, will be quite extensive, by supplying two separate motors, this may actually help with reducing wear on a single motor.

The best option, for the U of I Discovery teams budget, and for the Discovery Center, from a price, and parts replacement standpoint, was to purchase the Trossen Robotics PhantomX Model Robot Kit. This configuration, in the end will achieve the results that our client Dr. Ralph Budwig, and the Discovery Center were looking for, which allowed the team to save more money, that would benefit future purchases of hardware, and other componentry.

System Diagram Concepts

The main concepts that were in need of development, began with basic system design schematics. Due to this fact, our group was mainly entranced with the issue of actually getting the robot arm to function after initial assembly was completed near the end of September, prior to the first Snapshot day. Below are the three concepts that our group brainstormed as possible solutions to operate the robot arm.

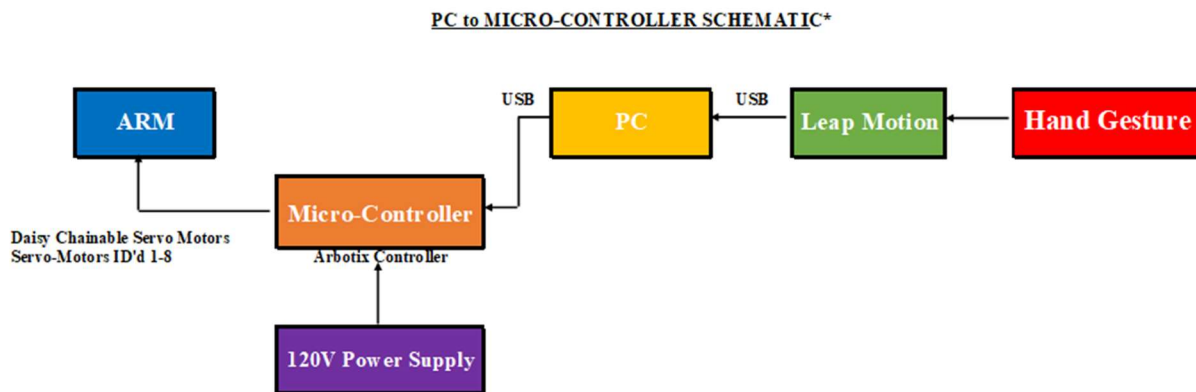


Figure 6: System Diagram Concept 1 - PC to Microcontroller Schematic, that uses USB connections and the ArbotiX microcontroller to power the 8 servo-motors working with the robot arm.

The first schematic that we brainstormed up, was to use Arduino, and C++ coding, and process it through the ArbotiX microcontroller, which would supply the code to the robotic arm. With values simultaneously being transferred, our thoughts on this schematic were on again, off again throughout the first semester of work. The main concern was overriding the system with too many values, to the point that latency would become an issue.

U2D2 CONVERTER SCHEMATIC*

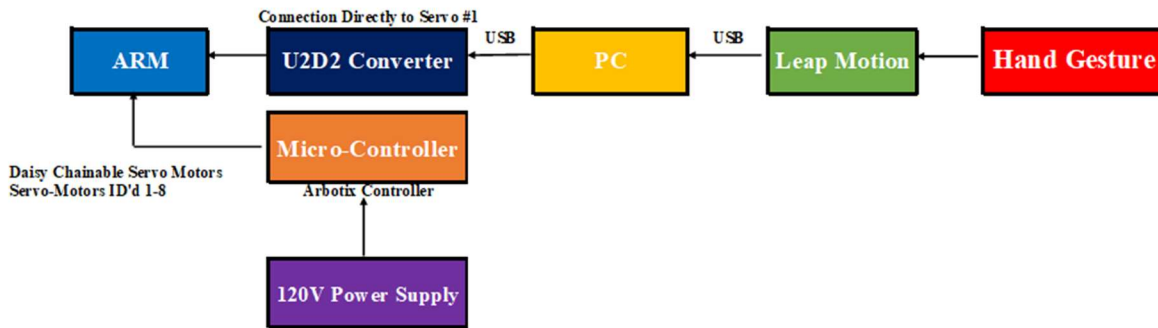


Figure 7: System Diagram Concept 2 - U2D2 Converter implementation, with though that this configuration will bypass the need for the Arbotix-M Controller

From the discovery of the U2D2 converter, the group had high hopes of its ever certain implementation, as it seemed to be a wonderful solution to any problems that we may encounter. The main reason for such confidence in this particular software setup came from the viewing of a YouTube video in early October, that showed a similar robot mirroring user input via Leap Motion control.

Arbotix Commander to Micro-Controller Schematic*

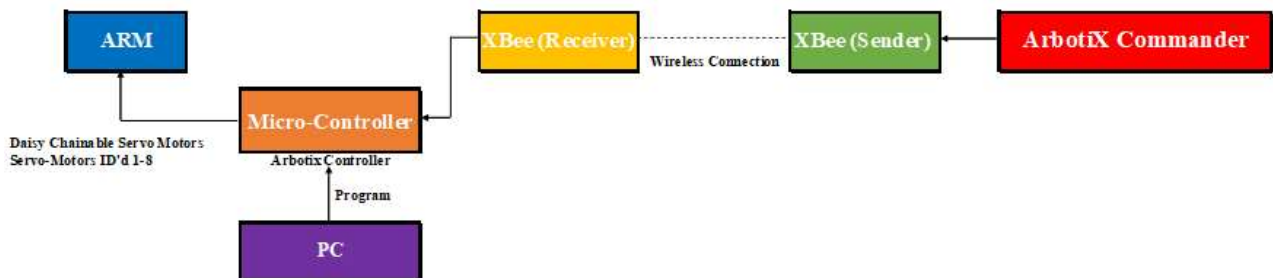
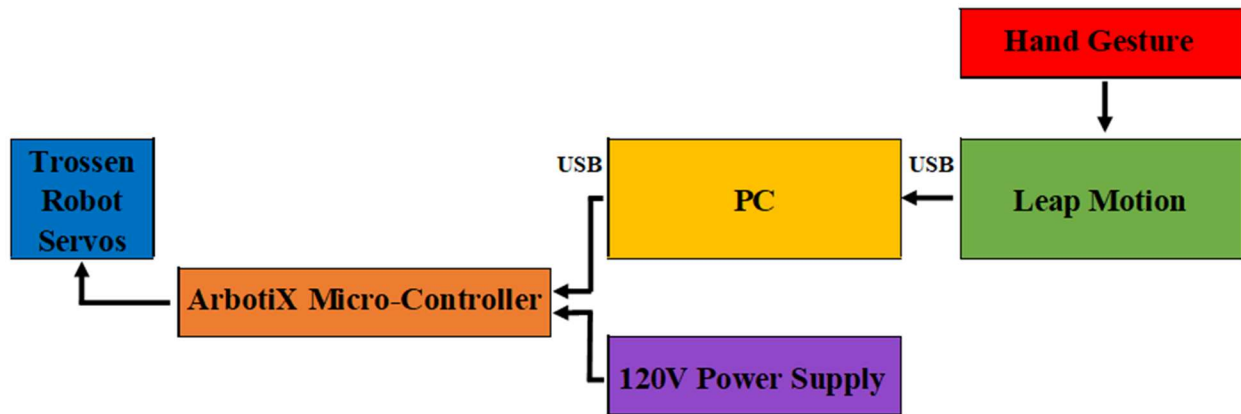


Figure 8: System Diagram Concept 3 - Arbotix Commander Controller with the XBee wireless microchips, controls the robotic arm with implementation of code. This was a primary solution that would help us brainstorm overall code, from provided firmware given for the hand held controller.

The main reason for this concept was to see if the robot arm would work with given input, as we were struggling at the time to implement the first concept involving PC to microcontroller interface, shown in Figure -. The importance of this aspect of the conceptual period was also to help show off at Snapshot Day 2, that we were developing strategies to come up with an actual set up that was feasible. While using this feature, we found that after ten minutes or less, of consistent use, the robot arms servo motors actually overheated, which lead to system shut down. This was mainly occurring in the base servos, 1,2,3 and 4, which support the lower half of the robot. This was not ideal, though it did bring up into question what the actual working temperatures of the servo motors is.

Throughout the process of actually considering certain system set ups, our group toyed around with the idea of using three similar, but slightly different designs as explained earlier within this

paper. The final decision came down to which design was actually feasible, and what would work best for all intents and purposes at the Discovery Center of Idaho. The final design that the group selected is the system diagram shown in *Figure* - below.



***Daisy Chainable Servo Motors (ID'd 1-8)**

Figure 9: Final Design Schematic, which shows the general flow from hand gesture to actual robot arm motion.

System Architecture

The general design of the robot arm involves the use of three devices, and two different types of software. The three devices, are the robot arm, the Leap Motion device, and desktop PC, with Windows 10 operating system. The software that is primarily used to operate the robot arm is Arduino IDE spliced with C++ based code, which works with the robot arm and Leap Motion device respectively.

Trossen PhantomX AX-12 Reactor Robot Arm



Trossen Robotic states that, “The PhantomX Reactor Robot Arm is the first in InterbotiX Labs' offering of ArbotiX based research grade robotic arms. The Reactor Arm was designed with

reach and agility in mind, but it still boasts considerable strength for an arm of its size. The PhantomX Reactor Robot Arm was designed with entry-level research and university use in mind, providing one of the highest featured arms on the market today, while not breaking one's budget."Included is a four to five degree of freedom robot kit, that uses AX-12 Plastic Geared 300-degree rotation, Servo-Motors. Delrin, Acrylic, and metal is used to hold together the basic robot structure. Dual Joint motion acting at servo-motors 2 and 3, as well as servo-motors 4 and 5. Daisy chaining is used, which interconnects every servo-motor within the system. Daisy Chaining allows for simultaneous data transfer throughout the system, and more effective communication.

ArbotiX Commander v2.0 Handheld Controller



Handheld controller that uses a video game controller layout similar to that of the PlayStation Dual Shock controller configuration. Connected to the robot arm using wireless/Bluetooth connectivity X-Bee controller chips mounted on both the handheld controller and the ArbotiX Micro-Controller, control is at the user's fingertips. The main reason this device was used was to help with early prototyping opportunities, which helped guide the team towards a discovery, on how they could solve the coding and communication issues between Leap Motion, and the robot arm.

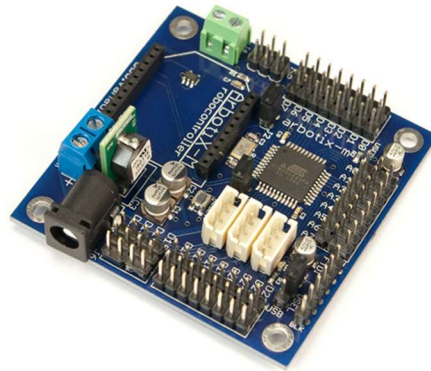
Leap Motion 3D Motion Controller



The main device within the set up that allows the robot arm to actuate, and imitate the user's direct physical inputs. Using two infrared camera sensors that track palm, finger and overall hand position in free space, the device is quite intricate for its use within the system. With a general

working space of one cubic foot, the dexterity of the robot arm is heightened with the integration of the Leap Motion controller.

ArbotiX-M Controller



The ArbotiX-M Controller is the top of the line microcontroller found within the Trossen catalog. As an advanced control solution for BIOLOID and DYNAMIXEL based actuators, the ArbotiX-M Controller is versatile enough for any hobbyist or project robot arm. Using Arduino based code, the benefit is that open source communities of libraries and examples are widely spread throughout the internet, and can be adapted for personal use. With 10 more general purpose I/O pins, a Barrel Jack for DC Power, Separate Power Bus for 4 Pulse Width Modulation (PWM) Digital outputs for the AX-12A servo-motors used with the PhantomX robot set up.

Dynamixel U2D2 Device



Initially thought to be a replacement or possible solution to communication issues between the Leap Motion Device and Robot, this was actually found to be nothing of that nature. Instead, the device was found to be a USB communication converter that enables direct servo-motor control through a PC. Mainly used to ID individual servo-motors, which should be done during initial servo-motor setup after replacement within the given robot arm structure.

System Requirements

The system needs to be Windows® 7+ or Mac® OS X 10.7+, with AMD Phenom™ II or Intel® Core™ i3/i5/i7 processor, 2 GB RAM, and three USB 2.0 ports, for robot arm and Leap Motion Connectivity. Arduino IDE version 1.0.6 is the main software used to run the robot arms code, for minimum and maximum servo-motor values, interpreted through the ArbotiX M-Controller. Arduino should not be updated from its current version, if it is, the system will no longer be runnable, due to the internal components of the ArbotiX M-Controller. GNU g++ compiler should also be usable, to support various programming languages, including Visual Studio, dynaManager software, and Arduino IDE.

To properly run and/or troubleshoot the basic system features, refer to the **Owner/User Manual** located on this flash drive.

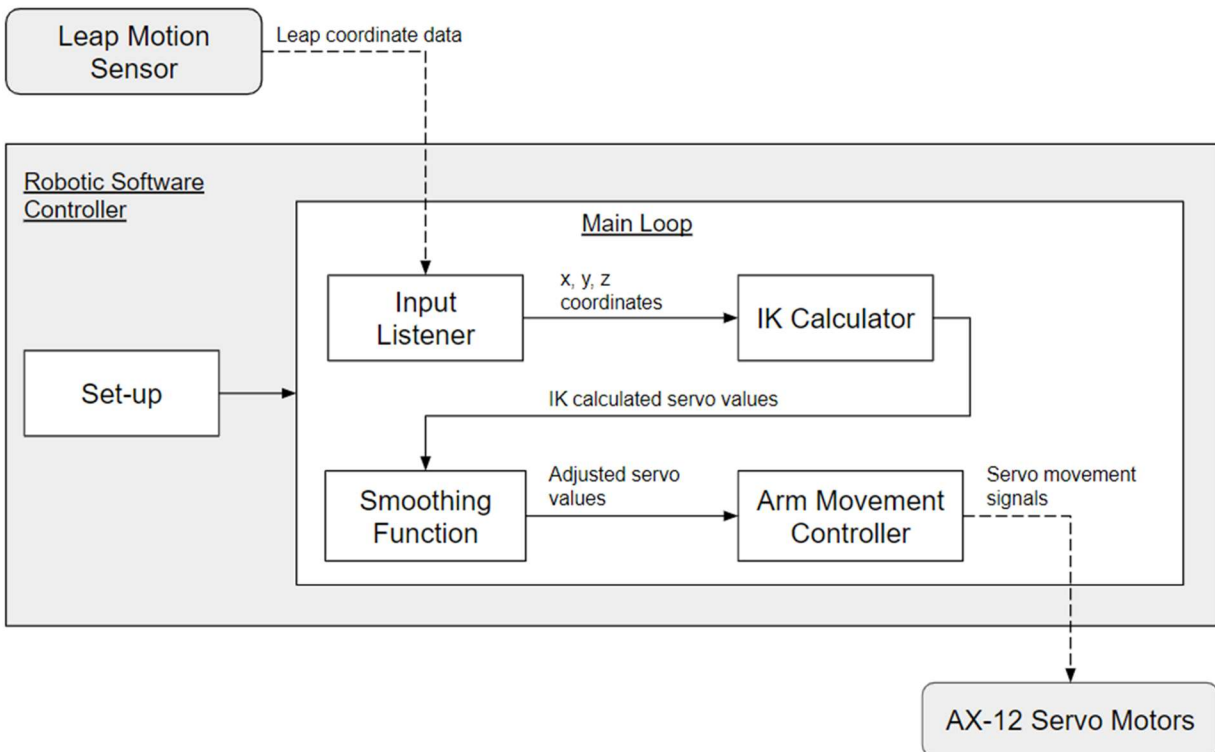


Figure 10: Data Flow Diagram, which depicts the basics of how information is processed within the working system, from Leap Motion input, to eventual robot arm function.

Design Evaluation

To test our design, our group took the basic structure of the Design Validation Plan (DVP) created, and distributed by Dr. Swenson, and tweaked the document to satisfy our needs within the testing environment. In doing so, our group came up with the set up shown primarily throughout *Figures 6 and below*. Throughout the entirety of the document, there are seven individualized columns, that state what is being tested, and a description of the test, target and completion dates for each test, and performance, result and actions needed to be taken, which

help compartmentalize each individual function. By using this, our group was able to complete tasks, and know what worked, and what didn't. Though this document is all "green" in the result column, for many different interactions, the result column was "red" or "yellow" for several weeks, while the group made corrections to create better working conditions and functionality of the robot arm.

Senior Capstone Design

Project: **Discovery Center - Gesture Control Robotics E** Primary Author: **Austyn Sullivan-Watson**

Team: **U of I Discovery**

Date: **April 24th 2019**

Design Validation Plan & Results (DVP&R)

Call Out	Test	Target Date	Completed	Performance	Result	Actions
Leap Motion						
Base Servo-Motor Actuation (Servo-Motor ID #1)	Free Rotation, with the 300 Degrees of Motion, Using leap motion to rotate the arm 90 deg. w/ hand motion	11/1/2018	10/24/2018	Completed, with refinements on the way. Actuation occurs, though it could be improved.		Purchase the U2D2 Converter, in hopes of reducing the amount of conversions needed. Look into reducing latency.
	Rotation using Servo-Motor ID #1 on the base via Arduino Software.	11/1/2018	11/1/2018	Completed, with refinements on the way. The Arm actuates by itself to the upright position, through Arduino Coding		Continue progress, and look into actuating the arm through Inverse Kinematics through Leap Motion. Look into reducing latency.
	Rotation using Servo-Motor ID #1 on the base via Arduino/C++ Code Mod.	2/13/2019	1/30/2019	Completed, base servo actuates, and moves side-to-side, mirroring user input. Latency isn't detrimental		See if reaction speed can be improved, but more than anything, functionality is of highest priority

Figure 11: Leap Motion Portion of the Design Validation Plan, which analyzes the performance, and actions that should be taken to improve overall usability of the robot arm.

In Figure 11 above, a portion of the Design Validation Plan, is shown for the Leap Motion device. Within this portion of the DVP, the base servomotor #1, is under analysis, which was the main point of emphasis for movement during the first few months of the project. By completing several smaller tasks within the Leap Motion programming, the overall big picture of the project was achieved in a more systematic, but smooth process. If I could use the analogy of baseball, if the team is down a few runs, it's better to string together a few hits, and take the small ball approach, rather than try to hit a home run.

Shoulder Servo-Motor Actuation (Servo-Motor ID #2 & #3)	Use Arduino Coding to activate the robotic arm into the upright position	11/1/2018	10/24/18	Completed, with refinements on the way. The Arm actuates by itself to the upright position, through Arduino Coding		Continue progress, and look into actuating the arm through Inverse Kinematics through Leap Motion
	Lateral Motion upwards and downwards, following user input via C++ Coding to Leap Motion Controller	2/13/2019	1/30/19	Completed, shoulder mirrors what the individual inputs to the Leap Motion Controller		See if reaction speed can be improved, but more than anything, functionality is of highest priority
Elbow Servo-Motor Actuation (Servo-Motor ID #4 & #5)	Using Arduino Coding to activate the robotic arm to the upright position.	11/1/18	11/1/18	Completed, with refinements on the way. The Arm actuates itself to horizontal/vertical position, through Arduino Coding		Continue progress, begin looking into how to make the arm actuate smoother, transition wise under Leap Motion Control
	Using C++ Coding w/ Leap Motion Sensory device, we use upward and downward hand motions to test functionality	2/13/19	1/30/19	Complete, the elbow mirrors user input, just like the base and shoulder.		Refinements in the overall movement, via restrictions on certain movements and overcorrections, so that we can reduce risks of broken parts, and any undeeded wear and tear on the robotic arm
Wrist (Lateral Motion) Actuation (Servo-Motor ID #6)	Using Arduino Coding to activate the robotic arm to	11/1/2018	11/3/18	Actuation occurs and it is seamless		Continue progress, now see if we can attempt to do the same thing through inverse-kinematics (IK)

Figure 12: : Leap Motion Portion of the Design Validation Plan, which analyzes the performance, and actions that should be taken to improve overall usability of the robot arm.

Gripper Actuation (Servo-Motor ID #7)	Use U2D2 converter to run Dynamixel Software, to see maximum and minimum gripper opening	2/27/2019	2/27/19	Minimum and Maximum Gripper Operating width, found and specified within the code.		Begin implementing the Gripper feature into the robot arms overall functionality
	Set limits in code, just below maximum and minimum for servo 7, then test the code via Leap Motion	3/1/2019	3/3/19	Smoother Operation from Minimum to Maximum opening of the gripper		Smooth function, though it differs for different block size. Make alterations to minimum gripper opening value, so that the servo isn't overworked.
Implementation of U2D2 Converter Directly into Base Servo Motor	Using Leap Motion and Inverse Kinematics Equations (w/ help of Dr. Perry) to actuate robotic arm					No ability to actually implement such software, due to the fact that it wouldn't necessarily provide better results than that of the Arduino/C++ Configuration
Implementation of the ArbotiX Commander v2.0 Controller	Movement/Resolution and overall testing of the Servo-Motor's Actuation, before continual progress with Leap Motion	11/29/2018	11/28/18	Completed, motion is somewhat lagging, might be able to increase reaction time		Continue progress, and see what can be done before Snapshot Day #2, and introduction to Ralph. Gain insight from Arduino code and see how it can be implemented to make Leap Motion work

Figure 13: Leap Motion Portion of the Design Validation Plan, which analyzes the performance, and actions that should be taken to improve overall usability of the robot arm.

Figure 12 & 13 like the previous figures shows the last few tests used to get the Gripper actuating with Leap Motion, along with the implementation of two other devices, previously depicted in the *Concepts Development* portion of this document. The U2D2 device is the third listed item, and was unfortunately found to not be useful in terms of running the device, as some of our research had stated. Instead, we found that the device would be useful with servo-motor identification, which is needed when the servo-motors are being initially set up for use, or replaced. In regards to the Implementation of the ArbotiX Commander v2.0 Controller, the device was incredibly helpful with eventual code development and proved that during Snapshot Day 2, which showed the response of the robot arm, when hand held control was implemented. The controller provided a basic foundation for the group to start developing code, which in return gave the group more confidence in the first system diagram

Unit Test for checking servo-motor actuation through automated program	Submit the code through either a C++ or Arduino format to test/debug any actuation errors within servo-motors	2/22/2019	2/27/19	Servo-Motors work, and will provide any feedback by individually moving to specified test location		Keep this software for any troubleshooting that may be needed in the future
Arduino						
Base Servo-Motor Actuation (Servo-Motor ID #1)	Actuates the Arm into the Upright Position, before rotation ensues	11/1/2018	10/24/2018	Completed, with refinements on the way. Actuation occurs, though it could be improved.		Look into implementing the Leap Motion Device through the U2D2 Converter or ArbotiX Micro-Controller
Shoulder Servo-Motor Actuation (Servo-Motor ID #2 & #3)	Actuates the Arm into the Upright Position, before rotation ensues	11/1/2018	10/24/18	Completed, with refinements on the way. The Arm actuates by itself to the upright position, through Arduino Coding		Look into implementing the Leap Motion Device through the U2D2 Converter or ArbotiX Micro-Controller
Elbow Servo-Motor Actuation (Servo-Motor ID #4 & #5)	Actuates the Arm into the Upright Position, before rotation ensues	11/1/2018	10/24/18	Actuation occurs and is seamless with the hard coding		Look into implementing the Leap Motion Device through the U2D2 Converter or ArbotiX Micro-Controller
Wrist (Lateral Motion) Actuation (Servo-Motor ID #6)	Actuates the Arm into the Upright Position, before rotation ensues	11/1/2018	10/24/18	Actuation occurs and is seamless with the hard coding		Look into implementing the Leap Motion Device through the U2D2 Converter or ArbotiX Micro-Controller
Wrist Rotation Actuation (Servo-Motor #-)	See Explanation Above within Leap Motion Section	N/A	N/A	N/A		N/A
Gripper Actuation (Servo-Motor ID #7)	Actuates the Gripper laterally with Servo-Motor rotational motion	2/27/2019	3/3/19	Actuation Occurs and is seamless with the hard coding		Look into implementing specific movement of the Gripper, by limiting the closing of the Gripper, thus reducing Servo-Motor freezing
System Operation	Implement IIR (Infinite Impulse Response) Filter Code, to help facilitate more seamless and smooth response w/o adding pauses within the system	4/2/2019	4/5/19	Actuation Occurs, is smoother, and is definitely refined. Works much better than the pauses that were within the code previously.		Make actuation even more seamless by tweaking the IIR Filter's equation. Change it slightly in one aspect of the robot arm, and see how the robot are actuates.

Figure 14: Arduino Portion of the Design Validation Plan, which analyzes the performance, and actions that should be taken to improve overall usability of the robot arm.

For the Arduino portion of the Design Validation testing, shown in Figure 14, the true control system implemented was the Arduino hard coded software, which was used for checking servo-motor actuation. By implementing the software separately, the group was able to focus on individual sections of the arm, which in the end helped validate how Leap Motion would interact with the Arduino code.



Figure 15: Visitors at the DCI Engineering EXPO booth, testing their hand precision.

For its first official test run with other users, then that of students and faculty of the University of Idaho, the robot held up considerably well to the use from well over two-hundred users, throughout a seven-hour period of time. The usability data collected mainly was operational, and how individuals reacted to the use of the arms gripper, and overall robot maneuverability. The majority of responses were positive, though at the end of the day, there were two main concerns include, gripper usability and robot arm usability.

Actuation of the gripper wasn't necessarily natural, as individuals were forced to adapt to the instructions from Team U of I Discovery's members. Smaller hands, mainly children less than the age of 6 or 7, struggled to use the robot arm, as the Leap Motion device wouldn't recognize the individual's hands on the infrared sensor.

In regards to robot arm usability, the arm's range of motion was seen to not be as great as the group had anticipated. By altering values within the codes structure for both C++ and Arduino, the overall maneuverability of the robot has been increased, which will hopefully supply the Discovery Center with a usable, and ever-changing exhibit. With limited time remaining in the semester, the team tackled these issues head on, and assure the Discovery Center of Idaho has a product that will be usable, for any age.



Figure 16: Servo-Motor Diagram depicting the name, and motion of each Servo-Motor.

Code to Servo-Motor Verification

Using basic Arduino coding and functions, a basic verification code was structured for the individual testing of servo-motors within the robot arm setup. The main purpose for the code is to isolate individual servo performance. For example, refer to *Figure 16* for visualization, but each servo-motor would be tested in sequence of their ID. However, this differs for both the shoulder lateral and elbow lateral servos identified 2, 3 and 4,5 respectively, run in conjunction with one another, being that they are run in line.

Hand Tracking Accuracy & Smoothness of Actuation

To test the main accuracy of the robot arm, and smoothness of actuation, filtering was implemented into the system, by the means of Infinite Impulse Response (IIR) filtering. By implementing such a “smoothing” function into the C++ code in Visual Studio, the data being inputted, could thoroughly be filtered in a way that would allow more values to be interpreted over the range.

Hand Tracking Delay

With immediate response a main priority, latency is not acceptable within the system, which is why the data input into the Leap Motion must be processed and turned into working robot motion, almost simultaneously.

Range of Motion Studies

The Range of Motion of the robot arm was tested using the full system set up, while the arm was connected to the Leap Motion device. Mainly testing was done with the approach of the user moving their arm in a circular motion, to see how the robot arm would respond. By doing so, initial tests resulted in the robot arm making a more oval motion, rather than circular motion, which lead to the group focusing in on the Inverse Kinematic (IK) properties, and the angles being calculated, to help facilitate a more accurate representation of what the users arm was really truly doing. By implementing different minima and maxima values, the robot arm began to imitate user input more so, thus providing more accurate feedback to what the user was actually trying to portray.

Conclusions

In regards to the adoption and implementation of the project, there are tips that are laid out within the Owner/User Manual, which documents the hardware, software, and different features that are included within the set-up.

In the future, the Discovery Center maintains the ability to add a servo-motor within the design, though it is highly unlikely that it will be implemented, due to the fact that the motion hinders the use of the gripper. By eliminating the wrist rotation, the Discovery Center will instead have an extra servo motor to use in the event of an old, or malfunctioning servomotor, which saves approximately \$50, when servo motors are in need of replacement.

Maintenance for the robot arm and leap motion device, are independent of one another. The robot arm will require maintenance at some point, after long durations of daily eight hour run time. The basic structural components of the robot arm should remain intact, with only servo-motor replacement applied when necessary.

Smoothness of function is the most important feature of the device, which has been stated directly from the director of the Discovery Center, Eric Miller. With this in mind, the group worked diligently on implementing software that increases the robotic arms smoothness, though, it could always be improved.



Figure 17: Team U of I Discovery's Booth Display at the XXVI Annual Engineering Design EXPO, held in the International Ballroom in the University of Idaho's Bruce M. Pitman Center.

References

- 10 Tips for Museum Exhibit Design Success. (2014, January 23). Retrieved October 1, 2018, from <http://www.teamdesignshop.com/blog/2014-01-23-10-tips-museum-exhibit-design-success>
- Blasius, A., Heiser, J., Puttre, K., & Stackable, J. (2013, April 29). Robotic Arm Exhibit [Scholarly project]. In Worcester Community Project Center. Retrieved October 1, 2018, from https://web.wpi.edu/Pubs/E-project/Available/E-project-042913-120855/unrestricted/EcoTarium_Final_Proposal.pdf
- Documentation - Leap Motion. (2017). Retrieved October 1, 2018, from <https://developer.leapmotion.com/documentation/>
- IP Rating Chart. (n.d.). Retrieved September 26, 2018, from <http://www.dsmt.com/resources/ip-rating-chart/>
- PhantomX Robot Arm Kit. (n.d.). Retrieved September 20, 2018, from <https://www.trossenrobotics.com/widowxrobotarm>
- PhantomX Reactor Robot Assembly Guide. (2017). Retrieved October 1, 2018, from <https://www.trossenrobotics.com/productdocs/assemblyguides/phantomx-deluxe-robot-arm.html>
- UL - 1004-6 Standard for Servo and Stepper Motors | Standards Catalog. (n.d.). Retrieved September 25, 2018, from https://standardscatalog.ul.com/standars/en/standard_1004-6_2

Appendix

Product Requirements Document

Discovery Center Gesture Control
Robotics Exhibit

Product Requirements



U of I Discovery

Austyn Sullivan-Watson

Angelo Stratigakes

Zhihui Wang

Chaeun Kim

Document History

Rev Number	Date	Modified By	Reason
0	26Sept2018	A.S. Watson	Initial Release
0	26Sept2018	C. Kim	Initial Release
0	26Sept2018	A.Stratigakes	Initial Release
0	26Sept2018	Z.Wang	Initial Release
1	1Oct2018	A.S. Watson	Revision 1
2	1Oct2018	A.S. Watson	Revision 2

Table of Contents

1	Objective	27
2	Scope	27
3	References	27
3.1	Cited Documents	27
3.2	Acronyms	27
4	Functional Requirements	28
4.1	User Interface Requirements	28
4.2	What it should do	28
5	Mechanical Requirements	28
5.1	Strength Requirements	28
5.2	Spatial Requirements	29
5.3	Weight/Mass Requirements	30
5.4	Mounting / Interface Requirements	30
5.5	Appearance Requirements	30
5.6	Durability Requirements	31
5.7	Reliability Requirements	31
5.8	Maintenance Requirements	31
6	Electrical Requirements	31
6.1	Operational Voltage	31
6.2	Operational Power Capability	31
6.3	Energy Storage Capacity	31
6.4	Functionality	31
6.5	User Interfae	31
7	Software Requirements	32
7.1	Functionality	32
7.2	User Interface	32
8	Environmental Requirements	32
8.1	Temperature	32
8.2	Environmental Sealing	32
9	Regulatory Requirements	32
9.1	UL Requirements	33
9.2	Shipping Requirements	33
10	Schedule Requirements	33

1 Objective

The objective of this document is to document the requirements regarding the Discovery Centers Gesture Control Robotics Exhibit, Robotic Arm configuration.

2 Scope

The scope of this document is to define the requirements that are needed to satisfy the client, and designer's vision for the project. By using details derived from the Trossen Robotics website, and designer insight into the project, this document should be able to explain most, if not all of the requirements needed to complete this project, to satisfy the client's requirements.

3 References

3.1 Cited Documents

10 Tips for Museum Exhibit Design Success. (2014, January 23). Retrieved October 1, 2018, from

<http://www.teamdesignshop.com/blog/2014-01-23-10-tips-museum-exhibit-design-success>

Blasius, A., Heiser, J., Puttre, K., & Stackable, J. (2013, April 29). Robotic Arm Exhibit [Scholarly project]. In Worcester Community Project Center. Retrieved October 1, 2018, from

https://web.wpi.edu/Pubs/E-project/Available/E-project-042913-120855/unrestricted/EcoTarium_Final_Proposal.pdf

Documentation - Leap Motion. (2017). Retrieved October 1, 2018, from

<https://developer.leapmotion.com/documentation/>

IP Rating Chart. (n.d.). Retrieved September 26, 2018, from <http://www.dsmt.com/resources/ip-rating-chart/>

PhantomX Robot Arm Kit. (n.d.). Retrieved September 20, 2018, from

<https://www.trossenrobotics.com/widowxrobotarm>

PhantomX Reactor Robot Assembly Guide. (2017). Retrieved October 1, 2018, from

<https://www.trossenrobotics.com/productdocs/assemblyguides/phantomx-deluxe-robot-arm.html>

UL - 1004-6 Standard for Servo and Stepper Motors | Standards Catalog. (n.d.). Retrieved September 25, 2018, from

https://standardscatalog.ul.com/standards/en/standard_1004-6_2

3.2 Acronyms

DOF	Degrees of Freedom
EPO	Engineering Purchase Order
POC	Proof of Concept

4 Functional Requirements

4.1 User Interface Requirements

The operator will be using hand/arm movements to operate the Leap Motion Sensor. The Sensor should be calibrated to react to movements taking place within the “specified dimensions,” of the overall exhibit from the Discovery Center of Idaho. The user, shall be accompanied by an adult if below the age of 10, for accountability, and overall usability of the system. Only a single operator should be allowed at the exhibit, at a time, for general safety, due to the fact that the robotic arm is a piece of machinery, and should be treated with respect, as it is available for use to several individuals visiting the Discovery Center of Idaho, daily.

4.2 What it should do

The Trossen Robotics PhantomX Arm shall be able to transport foam blocks, from a laser cut, see through container, within the exhibit, to specific target layouts, on a given table schematic, that corresponds to the foam blocks specific shape(s). This action shall be done through the act of the user’s arm placement over the “Leap Motion” Controller, which will transfer the information, at a rapid pace, which has yet to be specified for the ATmega644p Microprocessor, to the robotic arm, to complete the action within the exhibit. The Robotic Arm, should be able to withstand extended use for a 5-7-hour period, 7 Days a week. This use, will cause a durability issue over time, which will lead to required maintenance, which will be covered later within this document.

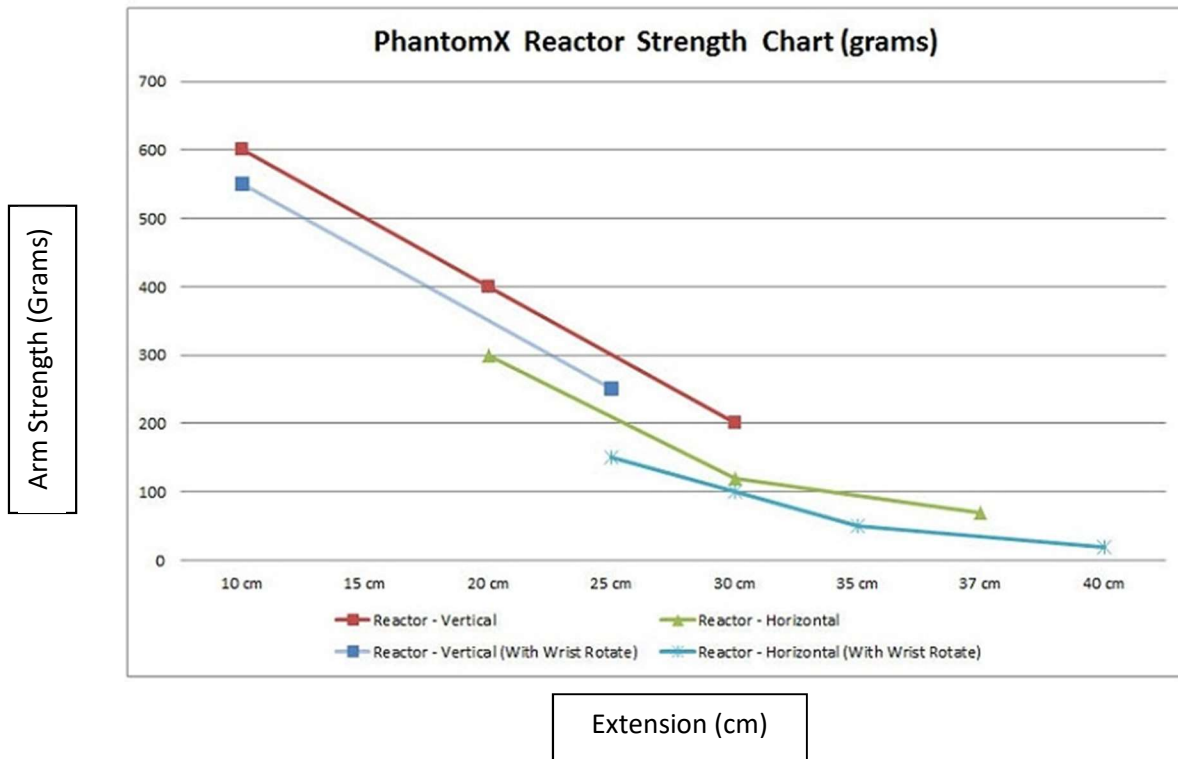
5 Mechanical Requirements

5.1 Strength Requirements

The design shall be able to carry a static load less than 20 grams, and should be able to take the static load, and place it accurately, and precisely within a given target, that has notable tolerances for fitment. By adding the wrist rotation to the device, the overall strength decreases about 50 grams to the same robotic arm configuration of the non-wrist rotation device.

Vertical Strength w/ Wrist Rotate: For 10 cm of reach, approximately 550 grams, 20 cm, approximately 350 grams, 25 cm of reach, approximately 250 grams, which for all intents and purposes is plenty, for the given mass of a foam blocks, which will be transported to specific targets within the exhibit.

Horizontal Strength w/ Wrist Rotate: For 25 cm, approximately 150 grams, 30 cm, approximately 100 grams, 35 cm, approximately 50 grams, which is great for the current application of the robotic arm. The gripper strength, according to Trossen’s website, is that it is 500 grams, without any specification, regarding the uncertainty, +/- a specific range of values. The wrist lift strength is 150 grams, with the given wrist rotation set up.



Trossen Robotics PhantomX Reactor Strength vs. Extension graphic

5.2 Spatial Requirements

The Robotic Arm & Corresponding Electronics shall fit within the following:

- Length < 80 cm
- Width < 80 cm
- Height < 80 cm

***Note that this may change, depending upon testing, and overall size that the Discovery Center/Dr. Ralph Budwig would like to have for the exhibit. The group will analyze the documents cited above in Section 3.1, regarding Exhibit Design.**

The Robotic Arm Extension

- Vertical: ~ 55.5 cm
- Horizontal: ~ 43 cm

The Robotic Arm Strength

- 200 grams w/ 30 cm of extension
- 400 grams w/ 20 cm of extension
- 600 grams w/ 10 cm of extension

***Note this will change based upon the extension of any parts of the arm, for viewing purposes. As of now, the encasement for the robotic arm is the last priority.**

5.3 *Weight/Mass Requirements*

The total system shall weigh no more than 6804 grams, or approximately 15 lbm, with the Robotic Arm, Base, Computer Set up, and any targets that may be associated with the exhibit. The mass of the Robotic Arm is 1430 grams. This will be suitable for the given exhibit set-up, as the tables will be able to hold larger weight capacities, than the stated amount above.

5.4 *Mounting / Interface Requirements*

The Robotic Arm shall be attached to the base provided by Trossen Robotics, which holds the Robotic Arm in place. The base is three tiered, for maintenance ease of access. A more substantial base may also be used for mounting, so that any interaction with the robotic arm, doesn't put the arm out of balance. The Robotic Arm will retain the ability to be modified for any reason necessary, in regards to the design, and its feasibility, based off of future testing, and inferences encountered. The robotic arm will also have system settings, that will correspond to specified levels of difficulty, with Novice, Intermediate, and Advanced difficulties for specified age ranges. The exhibit will be equipped with a controller, to specify which option the operator would like to try, and will work hand in hand with the Leap Motion software, to safely operate the arm for the specified difficulty. More will be added on this, with future revisions of the document.



5.5 *Appearance Requirements*

The final product should retain a black/metallic finish, unless otherwise stated by the client. This has yet to be determined, as we haven't discussed in full detail what the arm should look like. The appearance is also dependent upon the replacement of 3D printed parts, and the colors available for customization. The final product should be satisfactory to the Discovery Center and shall represent the University of Idaho, with a great looking, quality product.

5.6 Durability Requirements

The system should be designed to operate for approximately one month, without any scheduled maintenance. This isn't an official number, though it is an estimate, based off of assumption, though further knowledge will be applied once it is somewhat known, through testing. By using dual servos within each moving joint of the robotic arm, the overall life of the servos should be greater than that of single servo-actuated parts. The Robotic Arm should be analyzed, and tested thoroughly enough, to satisfy these requirements. With further testing, this information should become more accurate within the future.

5.7 Reliability Requirements

All components, including servos, and circuit boards shall have 90% reliability. The Robotic Arm should be able to actively be in motion, for 7 to 8 hours daily, besides Sunday's, when the Discovery Center's hours shift to a 5 hour working period.

5.8 Maintenance Requirements

The robotic arm itself has a few different areas, where maintenance will be needed, to take care of the excessive use of the device. The device contains minimal maintenance, with the Daisy-Chain Servo Motor compliance. The daisy-chain set up, is an electrical wiring scheme, in which multiple servo motors, are wired together in sequence, and can be "unchained," or detached. This is a great to have, due to the fact that when a servo motor needs to be replaced, it can be detached, and replaced with ease. By using dual servos within the elbow, and lower arm, where most of the actuation is occurring, this allows for less of a need for servo motor replacement.

6 Electrical Requirements

6.1 Operational Voltage

During operation the voltage should remain at 12 V, with the ability to allow 5 amps of direct current to flow through the system.

6.2 Operational Power Capability

During operation, the system will run off of a 12 V power supply, plugged in directly to a 120 V outlet, to support peak operation of the robotic arm, and other working systems.

6.3 Energy Storage Capacity

The Robotic Arm shall be able to work on a given charge capacity supplied from a 120V outlet, Arduino or Computer Power Supply, depending on the final power supply.

Software Requirements

6.4 Functionality

The software shall be able to withstand constant use over a 5-7-hour period. This will include any misuse that it may encounter during the work day, with the abundant use from mainly children, and adults. The code that is used for the interface should be able to safely allow the robot to move according to given

hand gesture input, without destroying the entire system, if a malfunction is encountered. The arm should be able to grab small foam blocks and move/place them, within a given range, based on the user's movements. The resolution, and reaction times will be found during testing, and will be depicted within this document throughout the ongoing design process.

6.5 *User Interface*

The Leap Motion device, will be securely fastened to the working exhibit, on a horizontal surface, with ease of access by the specified user. Depending on future installments to the overall "system structure," if another Leap Motion device is used, each will be labeled/marked with specific numbering, to coincide with which robotic arm is being controlled by which user. The user should be able to simultaneously move the robotic arm with their hand movement on a Leap Motion device. The movement of the robotic arm should be visible physically to the user as they move their arm over the Leap Motion.

7 **Environmental Requirements**

7.1 *Temperature*

The Product is expected to have full operational capabilities in environments with ambient room temperatures between 16°C and 27°C, or approximately 60.8°F to 80.6°F.

7.2 *Environmental Sealing*

All electrical components shall be at least IP50. This standard means that the components will be protected from limited dust ingress, but will not be protected from liquids. Though, if this is an important feature, needed for the design, this can be increased.

8 **Regulatory Requirements**

8.1 *UL Requirements*

The Servo and Stepper Motors, shall comply with the UL 1004-6 Standard. This standard is an accountable certification that is depicted in the documentation from Underwriters Laboratories. This categorization is used to evaluate the suitability of the servo motor, from the appropriate controller/driver, through a manufacturer declared operating procedure.

8.2 *Shipping Requirements*

The Robotic Arm should be stored, and padded with protection during shipment. The arm should be encapsulated in a box, with foam pellets, or bubble-wrap. Below, is the envisioned storage for the arm, with the current dimensions from Trossen Robotics. The targets, foam blocks, and any other parts needed for the set up will also be accounted for, by placing them within their own separate ports, in the shipping container.



9 Cost Requirements

9.1 *Prototype Cost*

Cost to build a POC prototype shall not exceed \$1500 as of now. Based on design reviews, the overall budget may increase, depending on the needs, or the desires from the client, Dr. Ralph Budwig, and the Discovery Center of Idaho.

9.2 *Production Requirements*

Estimated annual volume of the product will be 1-unit total, with the potential of another, based on the client or Discovery Center of Idaho's decision to incorporate another into the system. Another arm could be incorporated, based on design reviews, or overall working product, to expand the abilities of the exhibit.

Projected cost for a single robotic arm, will approximately be < \$1000

10 Schedule Requirements

The following are the major Project Milestones:

- | | |
|--|------------------------------|
| - Approval of Requirements | Sept. 27, 2018 |
| - Concept Design Review | Oct. 9, 2018 |
| - EPO of Parts | Oct. 9, 2018 |
| - Detailed Design Review | Nov. 30, 2018 |
| - Fully Functional Gesture Controlled System | Nov. 30, 2018 |
| - Upgrades to Overall System/Set-Up | Jan. 9, 2019 - Apr. 26, 2019 |
| - Final Deliverables | April 26, 2019 |
| - UI Design EXPO | April 26, 2019 |
| - Final Report / Drawings | May 4, 2019 |