Contents

Abstract

Team Upper Echelon Engineering has designed, tested, and built a multi-link robotic manipulator with six degrees of freedom to enhance the laboratory for Fundamentals of Robotics (ME 4773) at the University of Texas at San Antonio. The laboratory currently does not have equipment to fully apply key concepts and current robotic manipulators used in the market. Further, these manipulators are expensive and funds are lacking for laboratory support. The newly developed robotic manipulator applies forward kinematics, inverse kinematics, and trajectory generation via three uniquely designed laboratory experiments while being relatively inexpensive compared to manipulators in the market. Six revolute joints are required to maneuver the end-effector (gripper) in a three-dimensional Cartesian space. Six servo motors were implemented to act as revolute joints. A predefined manipulator configuration was selected due to predetermined solutions found in literature. The trajectory generation applies inverse kinematics to maneuver the gripper position specified by time-dependent equations. The manipulator accomplishes three experiments with a gripper position uncertainty no greater than 5 in., in x-y-z. The cycle time of trajectory generation is 10 seconds reaching at most 10°/sec and $2°/sec^2$. The manipulator is an enhancement to the prior laboratory because it applies key concepts via three experiments.

Table of Acronyms

The following is a list of acronyms used throughout the text:

1. DOF - Degrees of freedom
2. UTSA - The University of Texas at San Antonio
3. ME - Mechanical Engineering
4. BOM - Bill of Materials

1.0 Introduction and Background

A robot is a machine that can do the work of a person and that works automatically or is controlled by a computer. Robotics is a field of engineering that is concerned with the design, building, and operations of robots. A manipulator is a mechanism that performs an operation, it functions as if it was a hand and is actuated in a skillful manner. Robotics is a field that is growing in popularity across the world especially in industries where automation processes are favored. In the year 2014 robot sales increased by 29% which is one of the highest levels ever recorded for one year. The automotive parts suppliers and the electrical/electronics industry were the main drivers of the growth. From 2010 and 2014 robot sales increased to about 48% which is a clear indication for the rise in demand for industrial robots worldwide. China, Japan, the United States, the Republic of Korea, and Germany contributed to 70% of the global robot sales. The United States was the third largest robot market which increased by 11% to the peak of 26,200 units. The trend to automate production in order to strengthen American industries on the global market was the likely reason for the growth.

Fundamentals of Robotics is a technical elective course offered every fall semester at UTSA. The course has gained popularity amongst undergraduate ME students since it was first offered in 2014 due to a 138% enrollment increase relative to 2016 according to course instructor Dr. Bhounsule.

The course focuses on the fundamental analysis and control methods of robot manipulators by studying their kinematic and dynamic mathematical descriptions. The nature of the theory is fully appreciated when it is actually applied, hence the implementation of MATLAB for visualization of the abstract mathematical formulations. Upcoming students who enroll in the course could benefit from a more effective laboratory. Currently, the course instructor Dr. Bhounsule began incorporating a laboratory section for the first time during the fall semester of 2015. The laboratory consisted of 4 experiments to be conducted using the LEGO Mindstorms EV3 kit. The first three experiments had no relation to the theory taught in the course and the EV3 was an insufficient instrument for accomplishing the last experiment of drawing various shapes using a two-link manipulator configuration. The EV3 has two main limitations when it comes to relating the theory established in the course to the manipulation of the robot. First, the EV3 is a low quality medium for a practical college level laboratory due to its simplistic programming interface. Lastly, the resolution of the motors could be significantly improved from its current rated value of 1^o by the use of higher quality servomotors. Therefore, upcoming students could benefit from a robotic manipulator that effectively applied the key concepts taught throughout the semester within an enhanced laboratory section.

2.0 Purpose

The purpose of this project is to analyze, test, manufacture and build a multi-link robotic manipulator for laboratory experiments in the Fundamentals of Robotics (ME 4773).

3.0 Objectives

The Upper Echelon Engineering team has been tasked with developing a multi-link robotic manipulator with the capability of executing 3 potential experiments in ME 4773. The first experiment focuses on forward kinematics by comparing the end-effectors theoretical and experimental position in x-y-z through the implementation of joint angles 1 through 6. The second experiment focuses on inverse kinematics by specifying an end-effector position in x-y-z and calculating, theoretically, the joint angles 1 through 6 necessary to achieve that position and comparing the theoretical and experimental position of the end-effector. Lastly, the third experiment focuses on trajectory generation which requires the user to specify a time-dependent x-y-z function for the end-effectors position and the manipulator to replicate the movement; the user will visually describe the behavior of the trajectory generated and is required to demonstrate his or her results in a video to the professor.

4.0 Specifications

4.1 Functional Requirements

Construct a robotic manipulator for use in three experiments in ME 4773

Experiment 1: Forward Kinematics

- Determine theoretical and experimental end-effector position.

Experiment 2: Inverse kinematics

- Determine theoretical and experimental joint angular positions of joints by specifying a desired end-effector position (x,y,z)

Experiment 3: trajectory Generation

- Apply Inverse Kinematics incrementally to determine joint angles required to set end-effector position at each step of the way (discretize path)

To also fit into these functional requirements, the team will produce:

1. Develop operation manual consistent with the selected concept
2. Develop maintenance manual consistent with the selected concept
3. Develop laboratory manuals consistent with the selected design.

4.2 Performance Specifications

1. Manipulator must work with 6 degrees of freedom
   - End-effector must navigate through at least ½ of Cartesian coordinate
2. Fulfill 3 laboratory experiments for ME 4773
   - Forward Kinematics
   - Inverse Kinematics
   - Trajectory generation
3. Servo motors
   - Resolution $\leq 5.0°$ per servo
   - Supply voltage ≤ 12V
   - Mass ≤ 60 grams per servo
4. Easy to use programming interface
   - MATLAB and SimuLink to control motors
5. Manipulator Performance
   - End-Effector position uncertainty = $w_{x,y,z} \leq 3$ in.
   - Maximum joint angular position
     - Base servo: $0° \leq \theta_{base} \leq 270°$
     - Rest of Servos: $0° \leq \theta \leq 180°$
   - Maximum joint angular velocity: $0 \leq \omega \leq 20°/sec$
   - Maximum joint angular acceleration: $0 \leq \alpha < 10°/sec^2$
   - Cycle time ≤ 10 sec
6. Expected load
   - Cube geometry
   - Maximum volume of 1 $in.^3$
   - Weight ≤ 0.25 lbf
7. Gripper for robotic manipulator
   - Can be purchased off the shelf

- Maximum grip area when fully open = 1 in.^{3}
- Clamp force ≥ 0.35 lbf

8. Power Supply
   - Tethered connection
   - AC-DC converter to supply ≤ 12V DC
9. Length
   - Robotic manipulator arm length 1.5 ft. ≤ l ≤ 2 ft.

5.0 Concept Designs

In order to fulfill the course requirement for ME 4812 (Senior Design I) regarding conceptual designing, Upper Echelon Engineering sought out to produce 3 unique and functional concepts. The team researched pre-existing small-scale designs currently on the market and tailored each design with the improvements that were lacking in found designs.

5.1 Design Methodology

5.1.1 Concept Design 1: Bender

Bender was designed in order to implement 3 experiments that demonstrate concepts learned in the Fundamentals of Robotics course for the laboratory portion. Functional features of Concept 1 include its 6 degrees of freedom and their ability to navigate through ½ of the Cartesian space. This functional specification specifically allows it to execute the 3 concepts mentioned which are Forward Kinematics, Inverse Kinematics, and Trajectory Generation. The servo motor requirements that are needed to actuate all joints of the robotic manipulator were also implemented in the design. For this task, the Dynamixel AX-12 servo motors were chosen in order to meet the requirements of a resolution less than or equal to 5° , a voltage supply less than or equal to 12V, and a mass less than or equal to 60 grams. These features allow for the 3 links of Bender to be actuated smoothly and accurately. The AX-12's can be programmed and interfaced with MATLAB, as well as Simulink for control system implementation and joint actuation. The design concept also takes into account the maximum angular joint displacements specified. The base moves between 0° , and 270° , while the rest of the servos move between 0° , and 180° without any impedance. From these angles, the end-effectors position uncertainty is inherently equal to or less than 3 in. This can be done due to the servomotors capability of position feedback and its ability to be continuously rotated or be fixed between two limits. The maximum joint angular velocity, acceleration, and stalling torque at servo 2 were addressed with Bender. The weight reduction implemented through the geometry as well as material selection allows for the links to comfortably operate between 0 to 20° /s for angular speeds, and 0 to 10° /s^2 for angular accelerations. The mass of the links also takes into consideration cycle time, generally the less mass to actuate the faster the response. Overall inertial force, and its reduction of it, was the main focus of the link design and in order to make this achievable PLA was utilized for its density and tunable printing parameters. The design of the links however is questionable when it comes to picking up a cubic geometric load to be exerted at the end-effector position of not greater than ¼ lbf. They run a high risk of significant deflection that can potentially cause failure. Overall arm span of Bender is 2 ft. in total length starting from the base of servo 2. The link design of Bender is effective in mitigating weight of the links, however its non-modular design causes complexity in manufacturing and has a much longer lead-time. It should be noted that it is inexpensive to print due to its minimal use in material though. The links also allow for enough space to properly manage electrical cables that supply power and communication. This lightweight structure overall reduces loading on the servos located towards the base but due to this it has low rigidity which may potentially cause poor support and experience diminishing of structural integrity much faster than desired.

5.1.2 Concept Design 2: Johnny 5

Johnny 5 much like Bender was developed in mind for use in the Fundamentals of Robotics laboratory portion as well. The concept is able to perform the three experiments specified for the lab: Forward Kinematics, Inverse Kinematics, and Trajectory Generation. As mentioned previously, these experiments require but are not limited to the arm moving in 6 degrees of freedom. To actuate the joints of Johnny 5, the Dynamixel AX-12 servo motors were also implemented in this design. The motors operate at 12V and when combined, have a resolution less than or equal to 5° . The link and motor combination for this design does not exceed the weight of 75 grams. This makes it easy for them to be actuated smoothly and effectively. The servos are interfaced with MATLAB and Simulink to implement desired motions. The motion of the motors can be either continuous rotation or within 0-270° for the base or 0-180° for the other joints which the design addresses with its geometry. The concept is fashioned out of PLA plastic that is light and easily modified through printing parameters. Because the links are modular in design, the manufacturing and assembly of the arm is simplified and thus lead-time is reduced. The links also provide housing for electrical wires because they are hollow. The base provides suitable housing for a power hub, and USB2Dynamixel. This is achieved by separating the base section into two parts: a base plate piece for hardware and electronics and a hollow base lid. Wiring would be fed through the inside of the base attachment and then through the links. The mass of the links for Johnny 5 is a concern, however this can be mitigated through iterations of the design without loss of structure. Overall, the design focuses on simple modular linkages to reduce manufacturing lead-time and a large base for stability and storage.

5.1.3 Concept Design 3: Mr. Roboto

Mr. Roboto was developed in order to be incorporated into the laboratory portion of the Fundamentals of Robotics, also known as ME 4773. It was designed taking into consideration its ability to perform 3 experimental procedures. The first being Forward Kinematics, the second being Inverse Kinematics, and the final being Trajectory Generation. In order for the arm to properly do these tasks it was given 6 degrees of total freedom, and an arm span ranging between 1.5 ft and 2 ft., which grants it the ability to navigate through ½ of the Cartesian space. To do this, the joints of the arm must be actuated. Dynamixel AX-12 servomotors were chosen for this task to be achieved. They operate at the desired specified voltage, which is 12V, they meet the resolution specification of less than or equal to 5° , and weigh no greater than 60 grams. The servomotors are programmable and interface-able with MATLAB and Simulink in order to implement the control system for actuation. This is done with the use of the USB2Dynamixel and the servos position feedback capability. The servomotors also possess the ability to be set to continuous rotation or be fixed between two limits that allows for the design to incorporate operating angles. These angles range from 0° to 270° for the base, and 0° to 180° for the rest of the servos. These displacements allow for Mr. Roboto to be actuated smoothly while avoiding stalling in motors 1 through 3 while traveling through angular speeds of up to 20° /s, and angular accelerations of up to 10° /s^2. The use of PLA for the links helps the design process due to its low density and tunable parameters through 3D printing. Geometry of the design allows for a load of no greater than ¼ lbf. of cubic type geometry at the end effector position without any loss in stability. The link design allows for the arm to be compact and lightweight while maintaining high rigidity. The modular design allows for quick and easy repeatability. However, longer manufacturing lead times may be experienced due to the complexity of the links, and causes manufacturing to be more expensive or almost impossible due to how complex the geometry is. The links also happen to have more material, thus increasing load on servos located around the base as well as inertia to overcome. Finally, the concept lacks proper cable management housing.

5.2 Selection Process

Although all concepts satisfied the specifications in one form or another, a pugh chart was constructed based on the key specifications involved in the project. To select a design concept, key specifications were weighted and scored for each concept 1 through 3. Concept 2 was selected largely because of its geometry of both the base and the links. The links provide concealed wire housing as well as simplicity of production as all the links are modular by design. This in turn allows printing time to have a relatively short lead-time ranging between 4 hours to 12 hours depending on the piece being printed. The base provides stability and suitable housing for hardware for the power hub and the USB2Dynamixel. The base was kept small and compact to optimize workspace.

Table 1: Pugh Chart for Selection Process

Weight			Bender	Johnny 5	Mr. Roboto
Portability	8	Value	8	8	5
		Calculated	64	64	40
Mathematical Challenge	8	Value	4	7	7
		Calculated	32	56	56
Design Complexity	6	Value	2	7	7
		Calculated	12	42	42
Accuracy	9	Value	2	7	5
		Calculated	18	63	45
Rigidity	8	Value	2	8	6
		Calculated	16	64	48
Size	8	Value	6	7	4
		Calculated	48	56	32
Design Printability	5	Value	10	7	7
		Calculated	50	35	35
Total			240	***380***	298

5.3 Selected Design Concept Strength and Weakness

Johnny 5 is a 6-degree of freedom multi-link robotic manipulator that spans out about 1.7ft of length. It is comprised of 3 modular rectangular PLA link to house servos, a PLA bracket for the gripper, a PLA base for the electronics, and 7 Dynamixel AX-12 servos for actuation. Strengths of this design include simple and lightweight printable geometry with short lead times, link modularity, sufficient use of manipulator workspace, and proper housing for electrical cables. The main weakness of the design is focused on stalling torques for servo 2 and 3. Although link weight has been significantly reduced the total inherent weight of the servos, links, and gripper exceeds the torque value required by servo 2 and 3 to properly lift the entire arm. For Johnny 5 the greatest torque experienced by servo 2 was approximated to be about 1.62 Nm. The least torque experienced was in servo 1 with a value equal to 0.00098 Nm. The results for Johnny 5 differ from Bender and Mr. Roboto for it has the greatest torque value required by servo 2 to keep the structure static. The reason for the increase of the servo 2 torque was because of the increase in mass for the links. For the torque value about servo 1 there were no significant differences because of the insignificant change in the total inertia of the manipulator. The torque value experience by servo 2 by Johnny 5 was the highest value compared to all of the design concepts. Next, the dynamic torque for servo 2 through simulation was approximately equal to 1.65 Nm. The subtle increase in torque about servo 2 was likely due to the low cycle time to change its angular position. The torque value experienced at servo 2 is the highest that the value will ever be because of the fully extended arm exerting the highest moment of inertia. Thus, the torque rating for the servomotor at this joint location should be greater than or equal to 1.65 Nm to achieve the desired trajectory in all experiments. The team is looking into incorporating 3 Dynamixel MX-64T servos at servo locations 1 through 3 in order to mitigate these problems. Engineering change orders will be submitted in order to accomplish this in later iterations of the design.

6.0 Prototype Design

6.1 Prototype Key Features

The main features that allow the prototype to stand out from the current alternative is how much cheaper it is to manufacture, and also how much quicker the manufacturing process takes as the Manipulator relies heavily on Rapid Prototyping (3D Printing). The other key feature that is important and makes the Manipulator outstanding is that it comes with its own experiments that suits the senior technical elective course Fundamentals of Robotics, allowing the instructor to visually teach the important principles of Robotics which is then revolved around the experiments designed (Inverse Kinematics, Forward Kinematics, and Trajectory Planning). The Manipulators extremely light-weight design allows it to be extremely mobile allowing versatility and since it is 3D printed, the replication of any damaged parts of the Manipulator can be easily fixed without a long lead time or an outrageous cost making it extremely efficient.

6.2 Prototype Performance

Refer to Test Plan in the Appendix.

6.3 Prototype Safety and Failure Modes

The following section overviews the safety rules, procedures and recommendations if any problems with the manipulator arise. Please ensure to follow thoroughly to avoid any incidents. Information appearing in a DANGER concerns the protection of personnel from the immediate and imminent hazards that, if not avoided, will result in immediate, serious personal injury or loss of life in addition to equipment damage.

- Keep away from the robot while it's moving
- Do not touch with the robot with wet hands
- Disconnect the robotic manipulator whenever robot is problematic

Information appearing in WARNING concerns the protection of personnel and equipment from potential hazards that can result in personal injury or loss of life in addition to equipment damage. Please, consider the following:

- Setup the robot in an environment with low on dust and humidity.

- The robot must be fully assembled prior to powering on. Do not plug in the robot if the base cover is removed.
- The robot wiring must be checked prior to powering on.
- The robot connection to power supply must be checked prior to powering on.
- Do not change wiring on the robotic while powered on.

Information appearing in a CAUTION concerns the protection of personnel and equipment, software from hazards that can result in minor personal injury or equipment damage. Please, consider the following:

- Keep the robot’s workspace clear of object.
- Ensure wiring is not tangled up on every joint.
- Ensure the wiring from the robot to the computer does not interfere with the robot’s moving

The prerequisites listed below are to ensure the smooth operation of the manipulator and avoid injury to the user as well as adherently prevent damaging of Robotic Manipulator in the process.

i) The Robotic Manipulator is based on Window 10 OS and Simulink via MATLAB.

ii) Allow sufficient workspace prior to set-up by clearing objects in the arm’s vicinity.

iii) Always ensure the manipulator is properly fixed to the base plate prior to operations; otherwise arm movements can cause damage and physical injury.

iv) Supply power to the manipulator after making sure all cables are properly connected. While powered on do not touch the cables as it may cause erroneous operations and/or damage.

vi) When handling the manipulator do so carefully as not to have your fingers stuck in the links.

vii) If the manipulator operates erroneously quickly cut off power by turning the power supply off.

viii) While the manipulator is in operation keep out of its workspace; ensure no objects enter the workspace during operations.

6.4 Prototype Diagrams

The figure 3 shows the summation of forces around the Gripper which utilizes the Newton's 2nd Law of Motion where the forces in the x- direction are zero for static condition and the forces in the y direction are zero for static conditions. The coefficient of static friction is based on the material interface which is plastic on plastic, it is assumed that this value would be constant and would vary based on manufacturer but should be equal to but no greater than 0.35. The factor of safety of the gripper is calculated to be roughly 2.5 and a clamp force applied of around 0.36 pound force. In figure 2, the servo 2 would require the largest torque value with a maximum torque of about 13.9 kg cm or 1.3 Nm and a maximum stall of around 1.5 Nm as the servos used in Joint 2 and 3 have a stall torque of about 6 Nm. Servo 1 from Figure 1 requires the least torque as the maximum torque required is calculated at 0.1 kg cm or 0.01 Nm; this is mainly because the servo does not have to counteract any force of gravity and only actuates the arm about the y-axis.

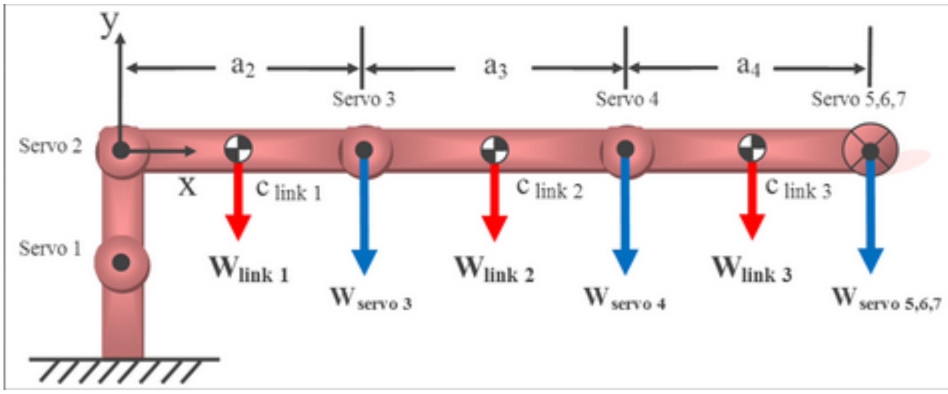


Figure 1: Free Body Diagram of Weight on Links

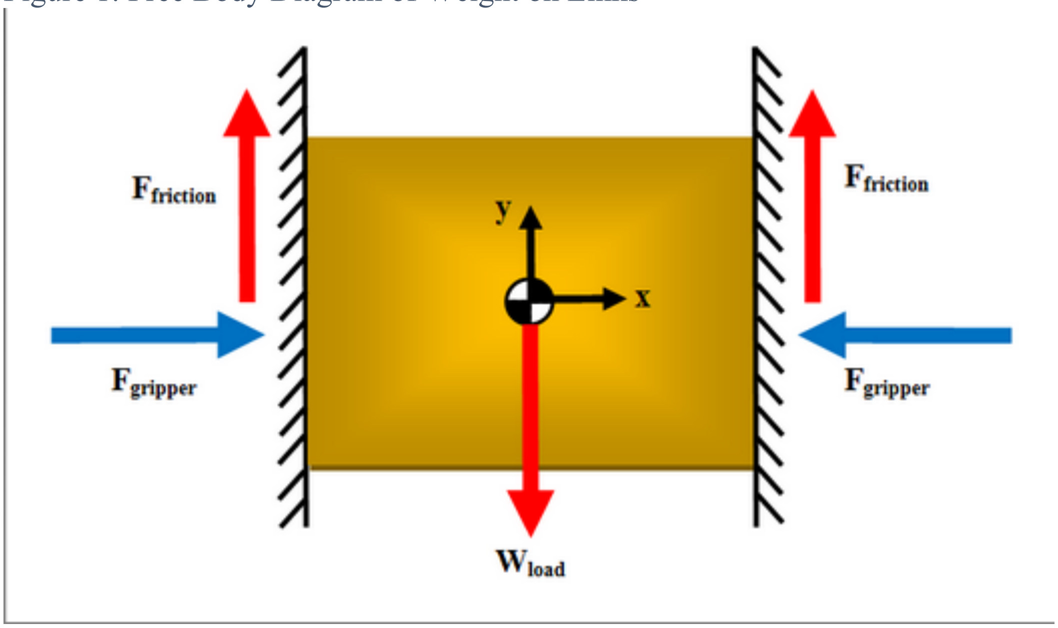


Figure 2: Gripper Free Body Diagram

7.0 Prototype Fabrication

7.1 Prototype Fabrication Methods

The main focus on the fabrication method for the Robotic Manipulator revolved around Rapid Prototyping or 3D Printing with the use of Polylactic Acid (PLA), which allowed the prototype to be easily duplicated if necessary, and any parts easily replaced as long as the drawings are available. The specifications and main purpose of the project revolved around the versatility of the Manipulator, cost reduction in regards to the alternative used in the Fundamentals of Robotics course as of today which is the Lego Mindstorm. The idea of replicating the entire manipulator in a timely yet low budget manner suits the fabrication methods that the teams decided to go with. The 3D printer used in the Rapid Prototyping process was the Ultimaker 2 that was provided to the team by our sponsor Dr. Pranav Bhounsule, located in his laboratory. Other parts of the Robotic Manipulator were store bought by the sponsor and incorporated into the selected design the team agreed on. The average print time for the parts in the Robotic Manipulator is about 14 hours with a print density of 20% in order to meet the weight specifications required and still maintain its rigidity and strength.

7.2 Prototype Fabrication Drawings

See Design Drawings in tab H.

7.3 Bill of Materials

1	Number	Name	Rev	Description	Cross References	Quantity
8	MD.007	AC-DC Power Supply	-	-	Tear Sheet # 9	1
9	MD.008	Dynamixel AX-12A Servo	-	-	Tear Sheet # 1	7
10	MD.009	Robotis F2 Frame	-	-	Tear Sheet # 3	5
11	MD.010	Robotis F3 Frame	-	-	Tear Sheet # 4	7
12	MD.011	S1 M2 x 6mm Screw	-	-	Tear Sheet # 5	36
13	MD.012	S2 M2 x 8mm Screw	-	-	Tear Sheet # 5	-
14	MD.013	S3 M2 x 10mm Screw	-	-	Tear Sheet # 5	-
15	MD.014	S4 M2 x 12mm Screw	-	-	Tear Sheet # 5	8
16	MD.015	S5 M2 x 16mm Screw	-	-	Tear Sheet # 5	22
17	MD.016	S6 M2 x 20mm Screw	-	-	Tear Sheet # 5	-
18	MD.017	S7 M2 x 25mm Screw	-	-	Tear Sheet # 5	-
19	MD.018	S8 M2 x 30mm Screw	-	-	Tear Sheet # 5	-
20	MD.019	SB M3 x 10mm Screw	-	-	Tear Sheet # 5	19
21	MD.020	N1 M2 x 1.8mm Nut	-	-	Tear Sheet # 5	54
22	MD.021	N2 M3 x 2.6mm Nut	-	-	Tear Sheet # 5	-
23	MD.022	SP1 M2 x 4mm Spacer	-	-	Tear Sheet # 5	-
24	MD.023	SP2 M2 x 6mm Spacer	-	-	Tear Sheet # 5	-
25	MD.024	BU M3 Bushing	-	-	Tear Sheet # 5	4
26	MD.025	WA M8 Washer	-	-	Tear Sheet # 5	4
27	MD.026	250 mm 3 Pin Cables	-	-	Tear Sheet # 6	8
28	MD.027	PhantomX Parallel Gripper Rail	-	-	Tear Sheet # 8	1
29	MD.028	PhantomX Parallel Gripper Arm	-	-	Tear Sheet # 8	2
30	MD.029	PhantomX Parallel Gripper Hand	-	-	Tear Sheet # 8	2
31	MD.030	HSH M2 x 8mm	-	-	Tear Sheet # 8	4
32	MD.031	BU M2 Bushing	-	-	Tear Sheet # 8	4
33	MD.032	IHS M8 x 15mm Screw	-	For power supply	Tear Sheet # 10	4
34	MA.001	Base Plate Sub Assembly 1	A	Glued together	MD.001, MD.002	1
35	MA.002	Base Plate Sub Assembly 2	A	Fastened components	MA.001, MD.005, MD.006, MD.007, 7 x MD.019, 4 x MD.032	1
36	MA.003	Base Cover & Servo Sub Assembly	A	Servo fastening	MD.003, MA.005, 10 x MD.013, 10 x MD.020	1
37	MA.004	Base Plate & Base Cover Sub Assembly	A	Base w/ servo	MA.002, MA.003, 8 x MD.019	1
38	MA.005	Servo Sub Assembly 1	A	Base w/ Servo 1 & 2	2 x MD.008, MD.010, 8 x MD.011, 4 x MD.020	1
39	MA.006	Servo Sub Assembly 2	A	Attaches gripper to servo 5	2 x MD.008, 2 x MD.009, MA.008, 4 x MD.011, 4 x MD.014	1
40	MA.007	Link Sub Assembly	A		MD.004, MD.008, MD.009, 2 x MD.010, 4 x MD.015, 12 x MD.020, 4 x MD.011, 4 x MD.014	3
41	MA.008	Gripper Assembly	A		MD.008, MD.027, 2 x MD.028, 2 x MD.029, 4 x MD.030, 4 x MD.031, 4 x MD.011, 4 x MD.020	1
42	MA.009	Full Assembly	A		MA.004, MA.006, 3 x MA.007, 4 x MD.024, 4 x MD.025, 16 x MD.011, 4 x MD.019	1
43	ED.001	Servo Connections	A			
44	ED.002	Power Connections	A			

Figure 3: Bill of Materials with Cross References to Assembly Parts

8.0 Prototype Tests

8.1 Test Plan Summary

All tests will be conducted either in the Robotics and Motion Laboratory located in the Biotechnology, Sciences and Engineering Building (BSE) 2.216 or at team member Jonathan Sackett's apartment. In order to maintain project time-management during the execution of the test plan a Test Plan Schedule was created. The test plan has been completed as 25/26 (96%) of the tests passed; the static and dynamic loading tests initially failed for servos 2 and 3 due to excessive motor torque but these issue was mitigated by purchasing more powerful motors with a higher stall torque value. The team completed that the test plan schedule between the dates of 11/08/2016 to 11/11/2016 giving us 4 days to redesign and retest the Manipulator with the new motors.

8.2 Test Setup

See Appendix.

8.3 Test Results

See Appendix.

9.0 Project Management

9.1 Overall Schedule

The project was divided into two semesters, SD1 and SD2. SD1 assignments consisted of work packages, presentations and progress reports. SD2 consisted of status reports, presentations and final binder submission. To meet these deadlines a Gantt chart for both semesters was devised and is illustrated below.

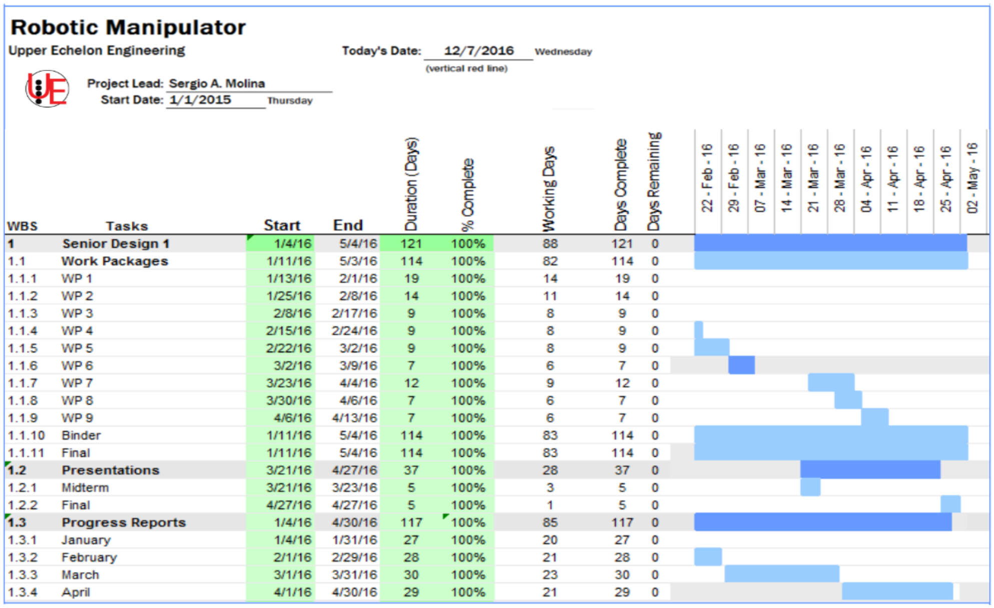


Figure 4: Gantt Chart for Senior Design 1

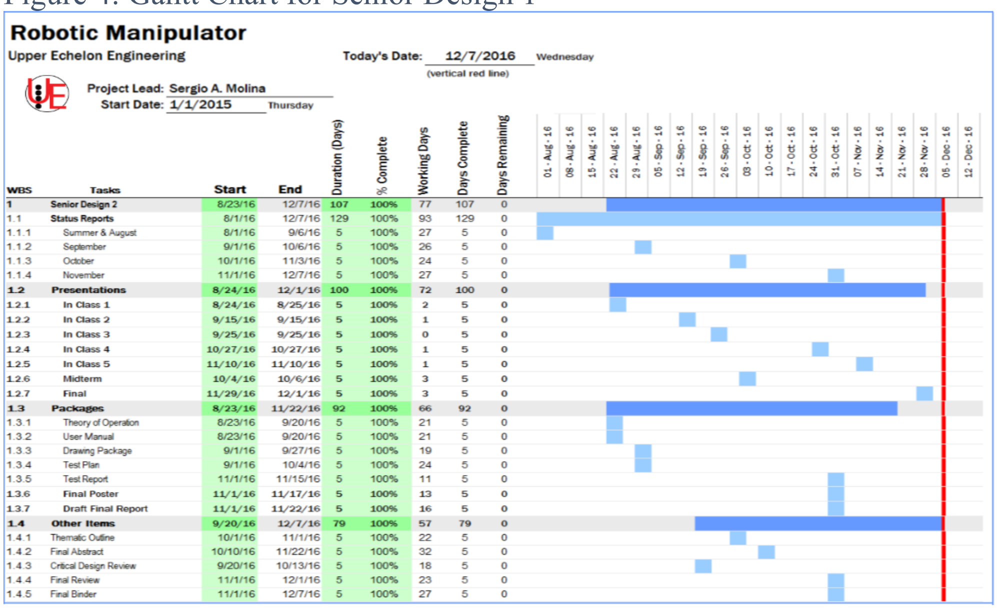


Figure 5: Gantt Chart for Senior Design 2

9.2 Team Member Roles

To efficiently manage time, individual member roles were assigned based on strengths. The following is a list of member roles within the project:

- Sergio A. Molina - Project Manager, Analyst and Code Developer
- Omar E. Castro - SolidWorks Design Specialist, Electrical Designer and Analyst
- Jonathan Sackett - Lead SolidWorks Design Specialist
- Paul Alabi - Manufacturing and Lead Technical Writer

9.3 Team Member Major Accomplishments

The completion of the project couldn't of been achieved without the individual efforts of the members. Sergio A. Molina created a custom library that allows users to control the manipulator via block diagrams in Simulink. He created a custom GUI within MATLAB that culminates the three laboratory experiments into one interface to control the manipulator. This would enable Dr. Bhounsule to demonstrate the power of robotics during presentations or use for research purposes. Omar E. Castro created two of three concept designs and assisted in static and dynamic analysis of the manipulator. He researched power supplies, communication devices and motors required to meet the project specifications. Jonathan Sackett created the selected design. He, with the help of Omar, devised a complete mechanical drawing package for the design of the manipulator. He created the SD1 and SD2 posters. Jonathan lead the team in manufacturing with assistance from Paul Alabi. Paul Alabi researched and selected the material for the manipulator. He managed and led the execution of the test plan according to the test schedule. He initiated all assignments and presentations.

9.4 Financial Performance

The project budget was $1000. The team finished the project ahead of schedule with a CPI value of 1.01. The team was under the scheduled budget finishing with a SPI of 1.0. During the months of October the project was falling behind schedule due to excess in employee hours relative to scheduled hours. The project manager mitigated the excess of employee hours by cutting scheduled hours for all employees for mid November, which resulted in budget stabilization. For example, CV values of 5,808.57, 5,408.57 and 2,408.57 during 10/04/16, 11/15/16 and 11/17/2016, as referenced in Table 1 Appendix 1E. Overall, on average, the team stayed within budget and schedule as interpreted by the parameters CV= -$2,792.10, SPI= 1.00 and CPI = 1.03.

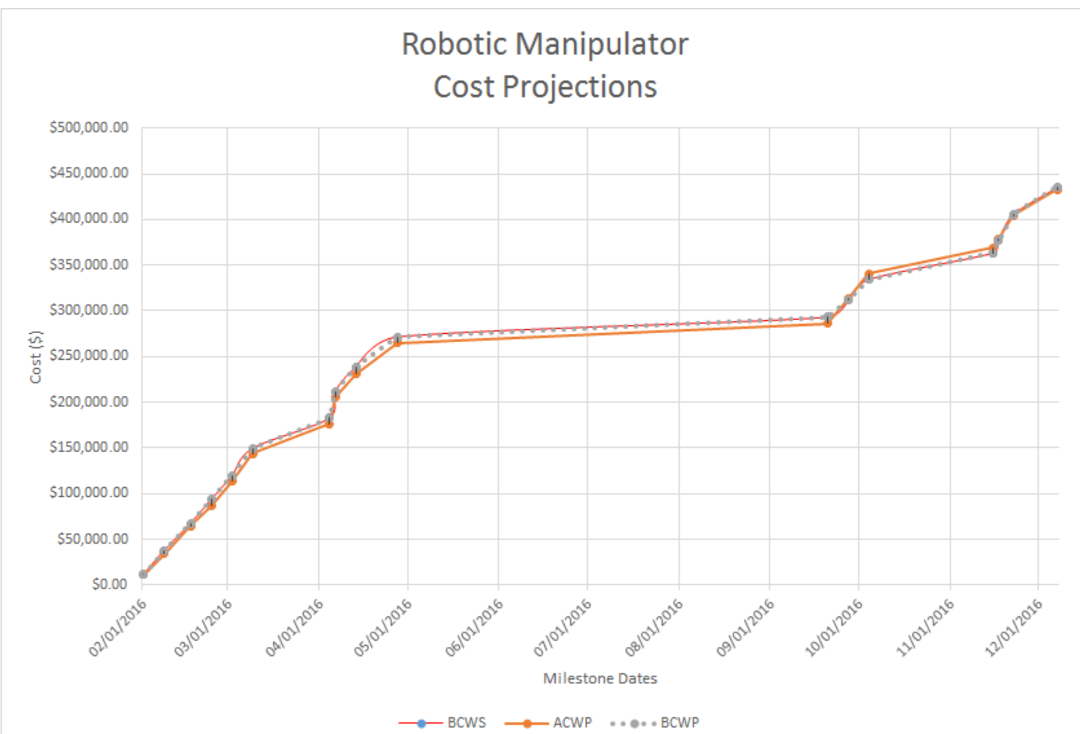


Figure 6: Robotic Manipulator Cost Projections

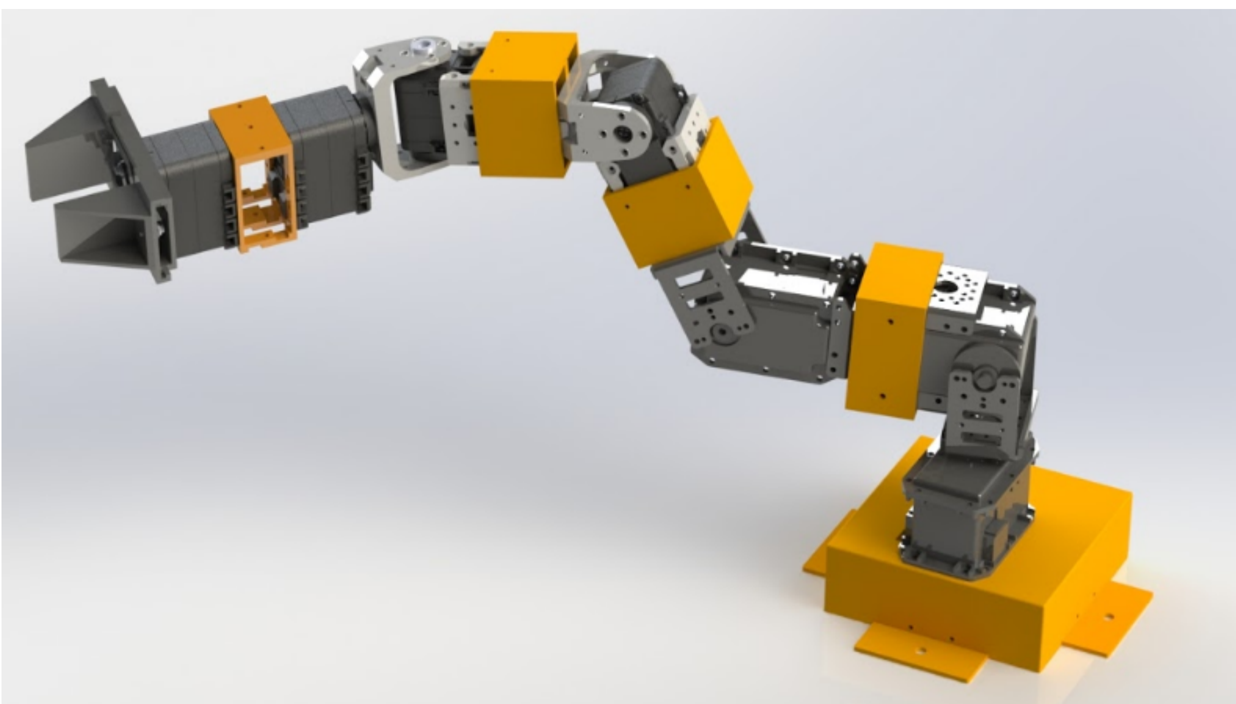
10. Conclusion

Over all the Upper Echelon Engineering completed the multi-link robotic manipulator for Dr. Bhounsule on time and under budget. The team learned the importance of scheduling and project managing in order to deliver on time and produce a functioning product for the client. The manipulator will be implemented in Dr. Bhounsules robotics laboratory in the months to come. The team overcame multiple points of adversity through the semester but quickly fashioned solutions for each problem encountered. This allowed the team to realize engineering projects often time need extra planning in order to compensate for errors of unprecedented nature.

Appendices A – Theory of Operations Manual

Robotic Manipulator

User Manual



Upper Echelon Engineering

Glossary, Terms, and Definitions/Acronyms

Robotic Manipulator	Robot, manipulator
Dynamixel	Servo Motor
Denavit-Hartenberg Table	D-H Table

Table of Contents

<u>Table of Contents</u>

[1.0] Product Description

The purpose of this section is to provide an overview of the product functionality, benefits and summary of key performance specifications.

The robotic manipulator is a 6-degree of freedom, multi-link arm that is built and designed for laboratory use in the Fundamentals of Robotics or any undergraduate mechanical engineering robotics class. The manipulator is designed to function with three fully devised laboratory experiments, specifically created to apply key concepts taught in robotics theory, that are embedded as part of this package.

The robotic manipulator is a great tool that will reinforce key concepts such as Forward Kinematics, Inverse Kinematics, and Trajectory Generation. As a result, the laboratory experience could be enhanced if there is an absence of such a device. Robotics can be a difficult subject for people to comprehend and part of the reason is that it is difficult to visualize. For example, when finding the end-effector (manipulator gripper) position in three-dimensional Cartesian space, it is difficult to visualize how the robotic manipulator's configuration is oriented in this space from using a four-by-four matrix. Thus, the goal of this product is to make the application of robotic theory much easier while establishing a fruitful experience for the user(s). In order for the robotic manipulator to be successful in interfacing with the three laboratory experiments the following key specifications were established:

Performance Specifications:

- **Load**
  - Cubic geometry
  - Maximum volume of 1 $in.^3$
  - Weight ≤ 0.25 lbf
- **Gripper**
  - Purchase off the shelf
  - Maximum grip area when fully open = 1 $in.^2$
  - Clamp force ≥ 0.35 lbf
- **Power Supply**
  - Tethered connection
  - AC-DC converter to supply ≤ 12 V DC
- **Length**
  - Arm length between 1.5 and 2 feet

[1.1] About This Document

I. This manual applies to the Robotic Manipulator produced by Upper Echelon Engineering.

II. All parameters in this manual are based on default values used in its respective experiments.

III. The manipulator applies to the Fundamentals of Robotics course.

IV. For understanding and operation, a proficiency in MATLAB and Simulink is strongly recommended.

V. Modifying the wiring, components, and/or procedures stated in this guide may result in adverse operations.

[1.2] Safety

The following section overviews the safety rules, procedures and recommendations if any problems with the manipulator arise. Please ensure to follow thoroughly to avoid any incidents.

- **DANGER!** 

Information appearing in a DANGER concerns the protection of personnel from the immediate and imminent hazards that, if not avoided, will result in immediate, serious personal injury or loss of life in addition to equipment damage.

- Keep away from the robot while it's moving
- Do not touch with the robot with wet hands
- Disconnect the robotic manipulator whenever robot is problematic

- **WARNING!**

Information appearing in WARNING concerns the protection of personnel and equipment from potential hazards that can result in personal injury or loss of life in addition to equipment damage. Please, consider the following:

- Setup the robot in an environment with low on dust and humidity.
- The robot must be fully assembled prior to powering on. Do not plug in the robot if the base cover is removed.
- The robot wiring must be checked prior to powering on.
- The robot connection to power supply must be checked prior to powering on.
- Do not change wiring on the robotic while powered on.

- **CAUTION!** 

Information appearing in a CAUTION concerns the protection of personnel and equipment, software from hazards that can result in minor personal injury or equipment damage. Please, consider the following:

- Keep the robot's workspace clear of object.
- Ensure wiring is not tangled up on every joint.
- Ensure the wiring from the robot to the computer does not interfere with the robot's moving

[1.3] Package Contents

The following section provides a table of the contents provided in the product package.

Table 1: List of Product Package Contents

Product Package Contents		
No	**Name**	**Quantity**
1	USB2Dynamixel	1
2	Power Hub	1
3	Dynamixel Servos	7
4	3 Pin Connectors	8
5	Fasteners (S1 - SB)	70
6	AC to DC Power Supply Converter	1
7	Link	3
8	Sticker of Gridline	1

9	Gripper	1
10	Wooden Board	1
11	Frames (F2 & F3)	10
12	Loctite Ultra Gel Control Super Glue	1
13	DVD with Relevant Content	1

[1.4] Layout

In order for the robotic manipulator to function the electrical components must be properly connected. The figure below illustrates the schematic of the electrical component layout.

First, the computer is connected via USB to the USB2Dynamixel where serial communication is received and transmitted via Half Duplex. Next, the Power Hub receives power from the Power Supply and provides the servos with the necessary current and voltage via the 3-pin for operation. In addition, depending upon the instruction packet ID number established in MATLAB, the computer may transmit and receive data (TTL) through the 3-pin connector.

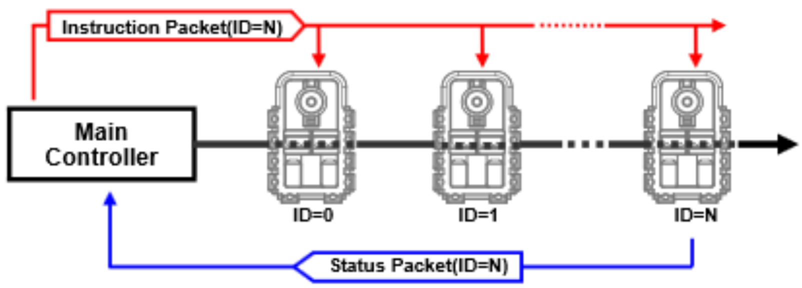


Figure 1: Communication Process

The servos have been configured to a unique ID address from 1 - 7 that correspond to the joint angles 1 - 7 for the robotic manipulator. Finally, the servo with ID 1 corresponds to the first joint angle of the robotic manipulator, which is then connected in a serial fashion (daisy-chain link) to the rest of the motors. The daisy-chain line provides all of the motors with the data, voltage, and ground lines necessary for operation.

[2.0] Theory of Operation

The purpose of this section is to provide an overview of how our product works as a system

[2.1] Prerequisites

i) The Robotic Manipulator is based on Window 10 OS and Simulink via MATLAB.

ii) ⚠ Allow sufficient workspace prior to set-up by clearing objects in the arm's vicinity.

iii) ⚠ Always ensure the manipulator is properly fixed to the base plate prior to operations; otherwise arm movements can cause damage and physical injury.

iv) ⚠ Supply power to the manipulator after making sure all cables are properly connected. While powered on do not touch the cables as it may cause erroneous operations and/or damage.

vi) ⚠ When handling the manipulator do so carefully as not to have your fingers stuck in the links.

vii) ⚠ If the manipulator operates erroneously quickly cut off power by turning the power supply off.

viii) ⚠ While the manipulator is in operation keep out of its workspace; ensure no objects enter the workspace during operations.

[2.2] Denavit-Hartenberg Configuration

The following section defines the parameters included in the mathematical analysis for the robotic manipulator. In addition, this section provides the Denavit-Hartenberg table necessary to find the end-effector position in Cartesian space.

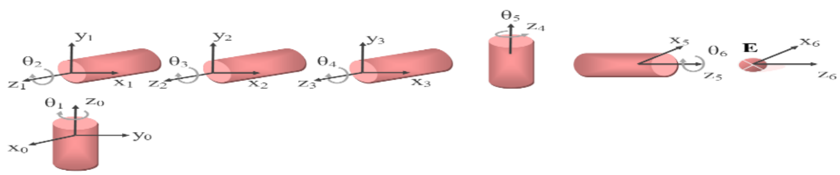


Figure 2: Robotic Manipulator Initial Configuration

<u>Defining Denavit-Hartenberg Table Parameters:</u>

- Link # - Number assigned to the link
- Theta - Joint angle associated with the servo motor
- d - Distance between adjacent x-axis
- a - Distance between adjacent z-axis which correspond with link length
- Alpha - Angle between adjacent z-axis along the x-axis

Table 2: Denavit-Hartenberg Table

Denavit-Hartenberg Table for Robotic Manipulator				
Link #	**θ**	**d**	**a**	**α**
0 - 1	θ_1	0	0	90^0
1 – 2	θ_2	0	a_2	0
2 – 3	θ_3	0	a_3	0
3 – 4	θ_4	0	a_4	-90^0
4 – 5	θ_5	0	0	90^0
5 - 6	θ_6	0	0	0

***Please Note: The above table provides the initial geometric orientation of the manipulator**

[2.2.1] Initial Position of Robotic Manipulator
The initial position of the robotic manipulator will be configured according to the D-H table and figure provided in the previous section. A MATLAB script file, titled "robot_initialization", has been provided in the DVD in order to initialize the robotic manipulator. Please ensure to always initialize the robotic manipulator prior to and after every experiment procedure by executing the script file previously mentioned.

[2.3] Manipulator System
The following section describes an overview of the electrical system, component specifications, and communication methods established for the robotic manipulator.

In order for the robotic manipulator to function, the servo components require a maximum of 12 V and 900 mA. We strongly suggest adhering to the following specifications for all electrical equipment if any component needs replacement as all parts are fit for the manipulator functionality.

[2.3.1] Overview of Robotic Manipulator System
The robotic manipulator joints are actuated by the Dynamixel AX-12A servomotors that allow position, speed and torque control. A block diagram of the system is provided below this paragraph. The computer will be transmitting and receiving instruction and status packets respectively via USB port through a half-duplex communication protocol.

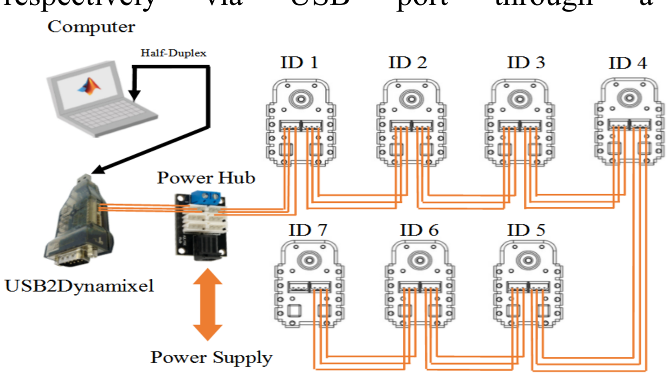


Figure 3: Robotic Manipulator Electrical System Configuration

The servomotors have an embedded microcontroller that will be accessed in order to control the motors using the computer interface. The USB2Dynamixel will convert the USB port to Serial port in order to interface with the microcontroller that operates via UART protocol.

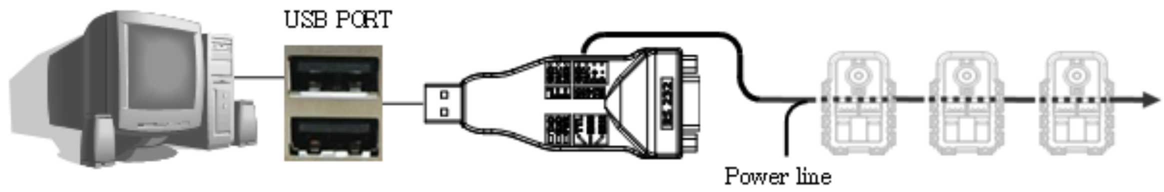


Figure 4: Robotic Manipulator Communication System

The power hub serves to provide the necessary power to the motors via the power supply, which is transferred via the 3-pin cable. Finally, the servos are connected via a daisy-chain link configuration so that all of the power and data is available for the seven servomotors. The servos have a preconfigured ID that matches with the joint of the manipulator from the theory devised in section 1.6. Finally, the ID is specified in the instruction packet that is constructed via the MATLAB script files and libraries provided with this package.

For a complete reference to the AX-12A User Manual please visit the following website: http://www.trossenrobotics.com/images/productdownloads/AX-12%28English%29.pdf

[2.3.2] 3-Pin Connector Specifications

The 3 Pin cable connects the servomotors and USB2Dynamixel and can be seen in the following table. Note that both the USB2Dynamixel and the AX-12A servos have the same connector slots for mating.

Table 3: 3-Pin Configuration Specifications/Image

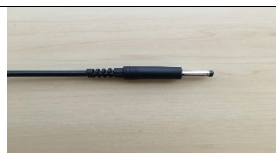
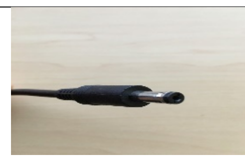
<table>
<tr><td colspan="3">3-Pin Connector</td></tr>
<tr><td>Connector</td><td colspan="2">
</td></tr>
<tr><td rowspan="4">Pin Array</td><td>No</td><td>Pin</td></tr>
<tr><td>1</td><td>Ground (GND)</td></tr>
<tr><td>2</td><td>Voltage (VDD)</td></tr>
<tr><td>3</td><td>DATA</td></tr>
</table>

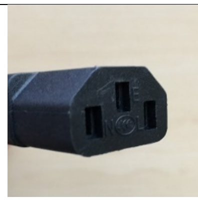
Cable	

[2.3.3] Power Supply Specifications

The following table details the power supply selected for the Robotic Manipulator, deviation from this power source is not advised.

Table 4: Power Supply Components/Specifications

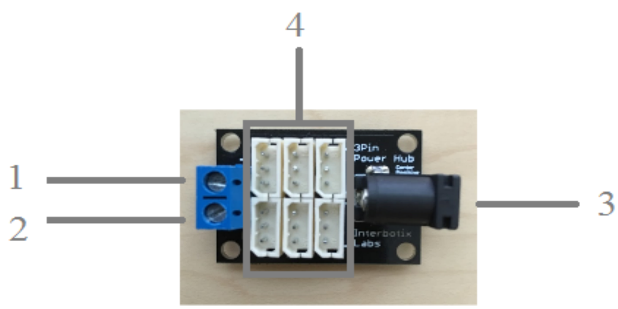
SUPERNIGHT 12V 10A Power Supply Components	
Title	Image/Specification
Jack Connector	 
	Specifications: 1. Output Voltage = 12V (DC) 2. Output Current = 10 A 3. Center Positive Polarity 4. Diameter of Connector = (5.5 X 2.1) mm barrel jack
Power Adaptor	
	Specifications: 1. Input Voltage = 100 ~ 240 V (AC) 2. Frequency = 50/60 Hz 3. Power = 120 W
Power Cable (Male End)	

	Specifications 1. Material = Plastic 2. Cable Length = AC-137CM/53.9 in.; DC-111CM/43.7 in.
Power Cable (Female End)	 Specifications: 1. Plug Type = U.S. Plug

[2.3.4] Power Hub Specifications

The following table provides the power hub's components.

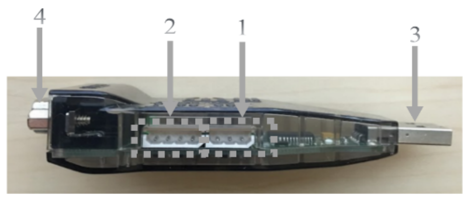
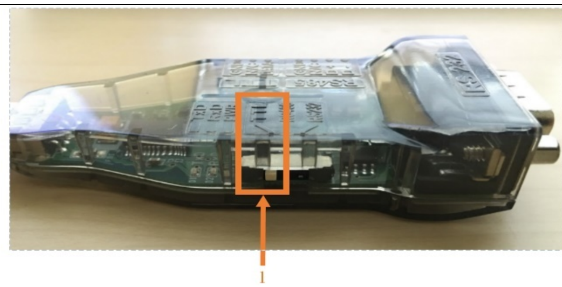
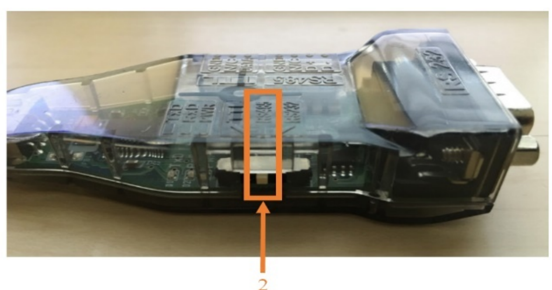
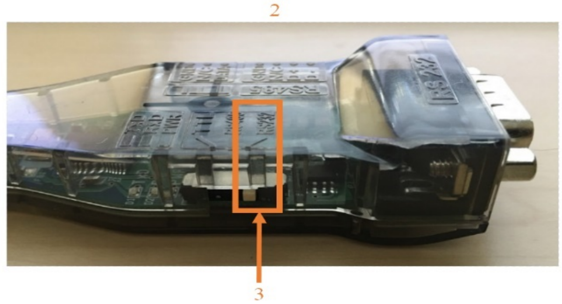
Table 5: Power Hub Component List

6 Port AX/MX Power Hub	
Title	Image
Power Hub	 Specifications: 1. Voltage (+) 2. Ground (-) 3. 3 Pin Input Ports 4. Jack Input Source

[2.3.5] USB2Dynamixel Specifications

The following table describes the different settings for the USB2Dynamixel communications.

Table 6: USB2 Dynamixel Specifications and Configurations

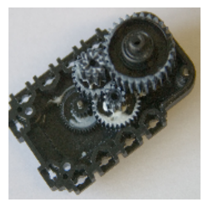
USB2Dynamixel Adapter	
Title	Image
USB2Dynamixel	
	Specifications: 1. 3-Pin Connection 2. 4-Pin Connection 3. USB Connection 4. Serial Port
Communication Modes	  
	Communication Type: 1. TTL-Level (RS232) - Recommended for AX Series

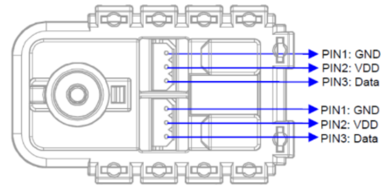
	2. RS485 - Recommended for RX, EX, and MX-R series 3. RS232 DB9

[2.3.6] Dynamixel AX-12A Servo Specifications

The following table describes the functions of each pin located in the ports of the servo.

Table 7: Dynamixel AX-12A Specifications

Dynamixel AX-12A Servo Characteristics	
Title	Image/Specification
General Specifications	 
	Specifications: 1. Weight = 53.5 g 2. Resolution = 0.29º 3. Gear Reduction Ratio = 254:1 4. Stall Torque = 1.5 Nm @ 12.0 V
Atmega8 Microcontroller	
	Specifications: 1. Command Signal = Digital Packet 2. Protocol Type = Half-Duplex Asynchronous Serial Communication (8 bit, 1 stop, No Parity) 3. ID Range = 0 - 253 4. Communication Speed = 7343 bps - 1 Mbps

Pin Configuration		
	Specifications: 1. Ground (GND) 2. Voltage (VDD) 3. Data (TTL)	

[2.3.7] Electrical Connections

The following schematics depict how to connect all electrical systems previously specified in order to ensure proper functionality.

The first schematic depicts how all motors should be wired in a daisy chain link. Then the following schematic after that depicts where the connections from the motors are connected to the power hub. It also shows how the other components are interfaced with the power hub.

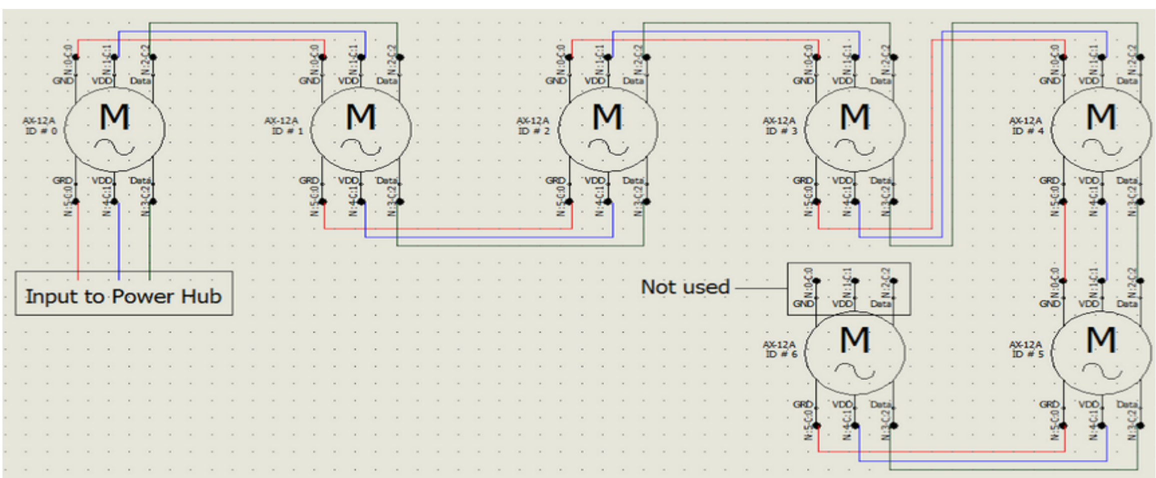


Figure 5: Electrical Schematic of Servo Motors

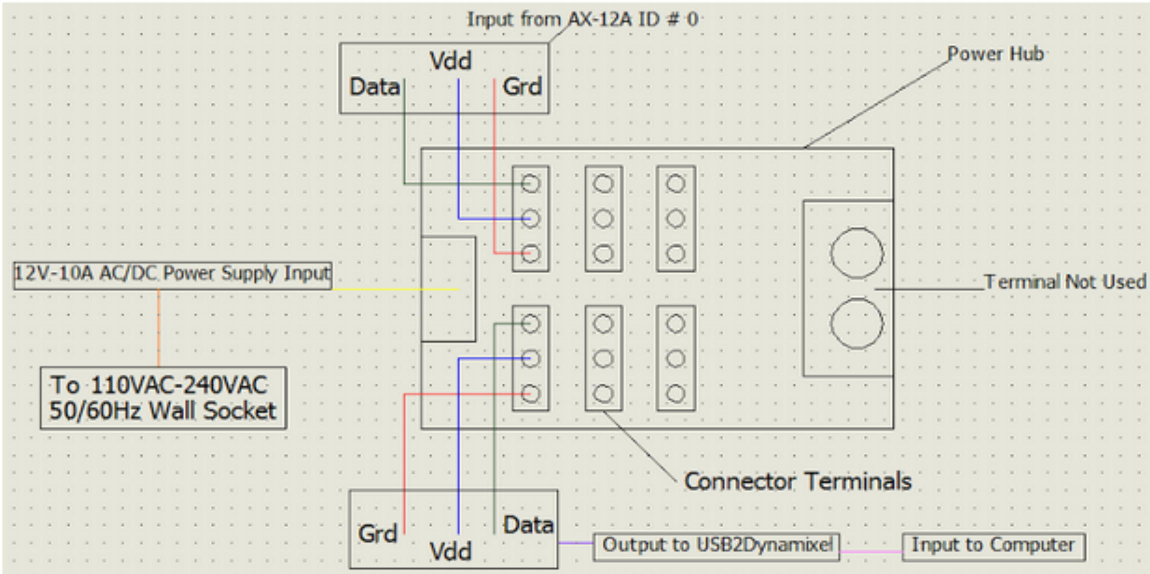


Figure 6: Electrical Schematic of Power Hub

[3.0] Setup and Operating Instructions

The purpose of the following section is to describe the details involved in setup and operation of robotic manipulator. The section describes the hardware setup, computer and software requirements, library installation, and calibration code required to initiate the robotic manipulator. A process for validating the installation of the Dynamixel SDK library is also provided.

[3.1] Hardware Setup

The purpose of this section is to provide hardware configuration and computer requirements for interfacing with the robot. In addition, procedures and tables are provided in order to ensure proper mechanical, electrical, & software setup.

[3.1.1] Installation of Robotic Manipulator

These steps listed are subject to change as the design is finalized. Changes made would be immediately noted and updated to keep the manual as up to date as possible.

Steps:

1. Apply the Loctite to the intersecting faces of both pieces of the base plate. Then connect the pieces by sliding them together. (Optional: for better performance, apply clamps to hold the pieces together until the glue dries.)

2. Remove the original screws from the power supply box but do not open the box. Screw the power supply to the base from the bottom using the 1”screws.

3. Screw the power hub into place using 4 Robotis SB screws. The screws are inserted into the hole while simultaneously threading the hole in the process. Once completed, connect power supply to power hub.

4. Attach the USB2Dynamixel using the bracket provided. The bracket is secured using the S2 screws. The screws are inserted into the hole while simultaneously threading the hole in the process. Once completed, connect it to the power hub.

5. Attach the Robotis F2 frame (flat side with 12 holes) to one of the three links (side with crosshair section and 4 holes) using 4 Robotis S5 screws and 4 Robotis N1 nuts.

6. Attach the F2 frame to a servo using a washer, a bushing, and a Robotis S-B screw. This is done by inserting the S-B screw through the bushing. These are then put through the large hole on the side of the F2 frame. The washer is slipped on the end of the bushing that extrudes from the inside of the F2 frame. Now carefully slip the Dynamixel inside of the F2 frame so that the S-B screw is on the side of the Dynamixel that says "ID". Then screw in the S-B screw into the Dynamixel to hold the parts together.

7. Attach a F3 framer to the Dynamixel and F2 frame from the previous step. These are connected by putting the flat face of the F3 frame against the side of the F2 frame that is opposite of the bushing and S-B screw. Spin the Dynamixel until the holes of the Dynamixel are aligned with the holes of the F2 and F3 frame. Once the holes are aligned, use a S3 screw to secure the pieces together.

8. Attach one of the servo/link assemblies to the slot in the cover of the base using 8 S5 screws and 8 N1 nuts. Orient the link side so that it sticks up out of the cover.
9. Make sure the cover servo is connected to the power hub first and then attach the base cover to the base plate using 12 S-B screws. The screws are inserted into the hole while simultaneously threading the hole in the process.

10. Repeat steps 5-6 for the other two links.

11. Attach 2 F3 frames to the sides of each of the Dynamixel from step 10. Use 4 S1 screws and 4 N1 nuts to secure each F3 frame.

12. Once all frames are attached, attach connectors to the 3 pin ports located on the servos and slide cables within the link.

13. Attach the three link servo/assemblies to each other. The servo fits inside the socket of the links. The servo is attached by 5 S4 screws and 5 N1 screws on two sides.

14. Attach the gripper assembly to a F2 frame, 2 spacers, a washer, a S-B screw, and 2 S4 screws. The F2 is connected to the back end of the Dynamixel, along the axis of rotation. The two spacers are used to raise the F2 frame off the Dynamixel and are held in place by the S4 screws. Likewise, the washer is used to raise the F2 frame and is held in place by a S-B screw. The S-B screw and S4 screws are screwed through the F2 frame into the Dynamixel keeping the F2 frame centered along the axis of rotation of the gripper assembly's Dynamixel. The S-B screw is screwed in directly along the axis of rotation. The S4 screws are screwed into the body of the Dynamixel.

15. The last Dynamixel is attached to the F2 frame from step 12 using 4 S1 screws and 4 N1 nuts.

16. Finally the servo/gripper assembly is attached to the end of the manipulator assembly. This is done by locking in the end Dynamixel (of the servo gripper assembly) into the end link (of the manipulator assembly) using 5 S1 screws and 5 N1 nuts on the two sides of the link.

[3.1.2] Computer Requirements

The MATLAB software used to control the robot is designed for use with MATLAB R2016a. Prior to connecting the manipulator to the computer it is recommended to verify that the minimum computer specifications are met. The USB2Dynamixel is not intended for Macintosh operating system computers. In order to properly interface with the manipulator the computer must meet at least the following specifications:

- Operating System: Windows XP/Vista/7/10 with 32/64 bit architecture
- Microsoft Visual C++ 2005 Redistribution Package
- Minimum requirements to run MATLAB R2016a

[3.1.3] Setting Up USB2Dynamixel

The USB2Dynamixel has a variety of communication modes so it MUST be configured to TTL mode in order to communicate with the Dynamixel AX-12A servo motors and actuate the manipulator. If the USB2Dynamixel is not configured to the specified settings no communication between the computer and the motors will occur. Therefore, it is recommended to always verify that the configuration on the USB2Dynamixel is always set to TTL mode as illustrated in table. Next, connect the USB2Dynamixel to any USB port. After connecting USB2Dynamixel the computer will automatically install the necessary drivers. If the drivers are not installed please visit the link provided for further assistance:

http://support.robotis.com/en/software/dynamixel_sdk/usb2dynamixel/usb2dxl_windows.htm

[3.1.4] Setting Up Power Supply and Power Hub

Before proceeding with this section please ensure that section 2.2 is completed. Please ensure that all cables are covered before connecting the power supply components.

First, the female end of the power supply cord is connected to the power supply housing by mating the two ends. Next, the power supply jack connector (component 1) shown in table 2 is connected to the power hub's jack input source (component 3). Next, connect the 3 pin connector from the manipulator to any of the six, 3-pin connector inputs of the power hub (4). All of the 3-pin connector slots function the same. The 3-pin connector is specifically designed to prevent any incorrect connections. Do not forcefully connect any component. All should connect with ease. Next, gather a 3-pin cable and connect it to any of the remaining five 3-pin connection slots then connect the opposite end of that same cable to the 3-pin connector of the USB2Dynamixel, illustrated in table (specification 1). Last, connect the male end of the power cable to a power outlet with a voltage rating between 120 - 240 V and a frequency of 50/60 Hz. Please, verify that the outlets are receiving the proper voltage and current ratings specified in the previous sentence as deviation from these values will result in component failure and may lead to potential hazards.

[3.2] Programming Methods

[3.2.1] Overview

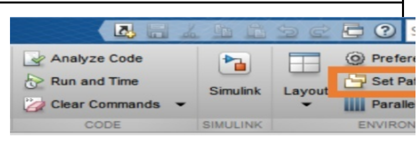
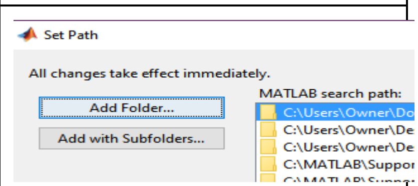
The actuation of the robotic manipulator is executed through the use of MATLAB and Simulink. In order to actuate the manipulator it is necessary to install the required MATLAB libraries which could be downloaded via the link provided after this paragraph. An installation guide of relevant libraries, along with step-by-step procedures are provided in this section in order to ease the software installation process. Furthermore, a validation process is described to ensure the Dynamixel SDK library was properly referenced by MATLAB. Next, the execution of a MATLAB script file, provided in the package via DVD or CD, will ensure that all motors are properly configured to the motors initial position. As a part of this package, all motor identifications (ID) are already configured to their corresponding joint number (1-7). Finally, the use of custom Simulink block diagrams, also included in the package, will enable the coding process to be more efficient and user-friendly.

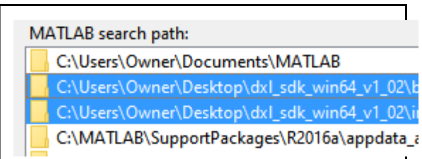
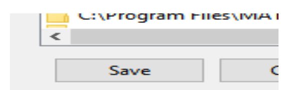
http://support.robotis.com/en/software/dynamixel_sdk/usb2dynamixel/usb2dxl_windows.htm

[3.2.2] Installation Guide of Relevant Libraries

It is necessary to install the Dynamixel SDK library from the link provided in order to make use of the open source code. The installation of the library is exactly the same as that found within the link and will be repeated here for convenience. In addition, the CD or DVD will contain the "DXL_SDK_Win32_ v1_02.zip" and "DXL_SDK_Win64_ v1_02.zip" files. First, locate the "DXL_SDK_Win32_ v1_02.zip" and "DXL_SDK_Win64_ v1_02.zip" files from the CD, DVD, or website link and download to computer hard-drive. The process is same for either version depending on your computer architecture, either 32 or 64 bit. Save the file in a convenient location within your hard drive for easy access. Next, extract the zip folder to a desired location on the hard drive. Next, open up MATLAB and follow the following the table of summarized steps:

Table 8: Instructions for Installing Dynamixel Library to MATLAB

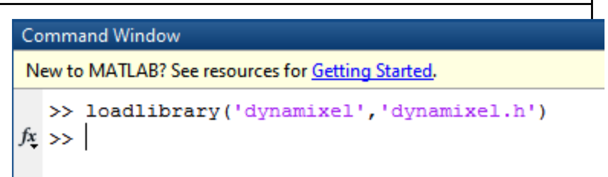
Instructions for Installing Dynamixel SDK Library to MATLAB		
No	**Instruction**	**Image**
1	Click "**File**" > "**Set Path**" in the menu option.	
2	Select "**Add Folder…**"	

3	Select the location of "**dynamixel.dll**" and "**dynamixel.h**" in the "**bin**" and "**import**" folder respectively, which are located where the "**Dynamixel SDK**" was extracted to. For example, the **Dynamixel SDK** was extracted to desktop in the image shown to the right with the following folder location: 1. "**dynamixel.dll**" location: C:\Users\Owner\Desktop\dxl_sdk_win64_v1_02\bin\dynamixel.dll 2. "**dynamixel.h**" location: C:\Users\Owner\Desktop\dxl_sdk_win64_v1_02\bin\dynamixel.h	
4	Select "**Save**" and close window after path is confirmed	

[3.2.3] Verification of Dynamixel SDK Library

It is highly recommended that before any calibration or experimental procedure is performed to verify that the Dynamixel SDK library was properly installed. The following table was established in order to verify that the library was successfully referenced by MATLAB:

Table 9: Instructions for Installation Verification

Instructions for Verifying Dynamixel SDK Library Installation		
Step	**Instruction**	**Image**
1	Load library into the MATLAB workspace by the use of the following command: **"loadlibrary('dynamixel','dynamixel.h')"**	

2	Check the list of library functions that are included with the following command:
	"libfunctions('dynamixel')"
		`>> libfunctions('dynamixel')`
		
		`Functions in library dynamixel:`
		
		`dxl_get_highbyte`
		`dxl_get_lowbyte`
		`dxl_get_result`
		`dxl_get_rxpacket_error`
		`dxl_get_rxpacket_length`
		`dxl_get_rxpacket_parameter`
		`dxl_initialize`
		`dxl_makeword`
		`dxl_ping`
		`dxl_read_byte`
		`dxl_read_word`
		`dxl_rx_packet`
		`dxl_set_txpacket_id`
		`dxl_set_txpacket_instruction`
		`dxl_set_txpacket_length`
		`dxl_set_txpacket_parameter`
		`dxl_terminate`
		`dxl_tx_packet`
		`dxl_txrx_packet`
		`dxl_write_byte`
		`dxl_write_word`

***Please Note: If the aforementioned steps were successfully achieved then the installation of the Dynamixel Library is complete.**

[3.2.4] Calibration of Robotic Manipulator

The following subsection provides an overview of the calibration process in order to initialize the robotic manipulator to the home state. This calibration process is required prior to conducting any of the three experimental exercises.

Upon completion of installing the Dynamixel SDK library and verification of its installation it is required to calibrate the servos to the initial state. Begin by downloading the "manipulator_initialize.m" file contained in the DVD provided in the package. Once the file appears in the "Editor" select "Run". Finally, servos should move to the initial state configuration as depicted figure 2. Upon executing the initialize script file the experimental laboratory may begin. Please repeat this process at the start and end of any experimental laboratory.

[3.3] Experiment Summary

This section provides an overview of the laboratory experiments that will be provided with this package.

[3.3.1] Experiment 1: Forward Kinematics

The focus of the first experiment is to calculate the Denavit-Hartenberg table and a transformation matrix relative to base frame. The matrix would substitute angles in the matrix in order to calculate the position vector. Using MATLAB the angles would be input into the servo motors and the student, using a ruler would be able to measure the x-y-z location of the gripper which is located at the end of the robotic manipulator and then compare the experimental values measured at the end of the gripper to the theoretical value originally derived.

$$A_i^{i-1} = \begin{bmatrix} cos\theta & -sin\theta cos\alpha & -sin\alpha sin\alpha & acos\theta \\ sin\theta & cos\theta cos\alpha & -cos\theta sin\alpha & asin\theta \\ 0 & sin\alpha & cos\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Equation 1: 4X4 Denavit-Hartenberg Matrix

$$A_6{}^0 = A_1{}^0 A_2{}^1 A_3{}^2 A_4{}^3 A_5{}^4 A_6{}^5$$

Equation 2: Matrix Product for End-Effect Orientation and Position

[3.3.2] Experiment 2: Inverse Kinematics

The focus of the second experiment revolves around a matrix form which is used to determine its closed-form inverse solution. The user would substitute the required end-effector position in relation to the x-y-z plane, and then use the positions to calculate the corresponding robotic manipulator angles. The calculated angles are then inputted into the servo motors using MATLAB and then compared to the theoretical values.

$$A_{Desired}{}^0 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Equation 3: End-Effector Orientation and Position Matrix

[3.3.3] Experiment 3: Trajectory Generation

The Trajectory Generation experiment utilizes the techniques established in the Inverse Kinematics experiment but iterates the solution to the end-effector position and orientation by a specified time-interval. First, the trajectory is a time function that is described in a three-dimensional Cartesian space. To add, the end-effector is only allowed to actuate in the upper, ¼ of the space due to the limiting angles provided by the servo motors. The specified trajectory is substituted into the inverse kinematic solutions for the joint angles provided below.

$$\theta_1 = tan^{-1}\left(\frac{p_y}{p_x}\right) + 180°$$

$$cos\theta_3 = \frac{[p_x cos\theta_1 + p_y sin\theta_1 - a_4 cos(\theta_2+\theta_3+\theta_4)]^2 + [p_z - a_4 sin(\theta_2+\theta_3+\theta_4)]^2 - a_2{}^2 - a_3{}^2}{2a_2a_3}$$

$$sin\theta_3 = \pm\sqrt{1 - cos\theta_3{}^2}$$

$$\theta_3 = tan^{-1}\left[\frac{sin\theta_3}{cos\theta_3}\right]$$

$$sin\theta_2 = \frac{(a_3 cos\theta_3 + a_2)(p_z - a_4 \sin(\theta_2+\theta_3+\theta_4)) - a_3 sin\theta_3 (p_x cos\theta_1 + p_y sin\theta_1 - a_4 cos(\theta_2+\theta_3+\theta_4))}{(a_3 cos\theta_3 + a_2)^2 + a_3{}^2 sin^2\theta_3}$$

$$cos\theta_2 = \frac{(a_3 cos\theta_3 + a_2)(p_x cos\theta_1 + p_y sin\theta_1 - a_4 cos(\theta_2+\theta_3+\theta_4)) + a_3 sin\theta_3 (p_z - a_4 sin(\theta_2+\theta_3+\theta_4))}{(a_3 cos\theta_3 + a_2)^2 + a_3{}^2 sin^2\theta_3}$$

$$\theta_2 = tan^{-1}\left[\frac{(a_3 cos\theta_3 + a_2)(p_z - a_4 \sin(\theta_2+\theta_3+\theta_4)) - a_3 sin\theta_3 (p_x cos\theta_1 + p_y sin\theta_1 - a_4 cos(\theta_2+\theta_3+\theta_4))}{(a_3 cos\theta_3 + a_2)^2 + a_3{}^2 sin^2\theta_3}\right]$$

$$\theta_4 = (\theta_2 + \theta_3 + \theta_4) - \theta_2 - \theta_3$$

$$sin\theta_5 = \cos(\theta_2+\theta_3+\theta_4)\,(a_x cos\theta_1 + a_y sin\theta_1) + a_z \sin(\theta_2+\theta_3+\theta_4)$$

$$cos\theta_5 = -a_y cos\theta_1 + a_x sin\theta_1$$

$$\theta_5 = tan^{-1}\left[\frac{\cos(\theta_2+\theta_3+\theta_4)(a_x cos\theta_1 + a_y sin\theta_1) + a_z \sin(\theta_2+\theta_3+\theta_4)}{-a_y cos\theta_1 + a_x sin\theta_1}\right]$$

$$\theta_6 = tan^{-1}\left[\frac{-\sin(\theta_2+\theta_3+\theta_4)\,(n_x cos\theta_1 + n_y sin\theta_1) + n_z \cos(\theta_2+\theta_3+\theta_4)}{-\sin(\theta_2+\theta_3+\theta_4)\,(o_x cos\theta_1 + o_y sin\theta_1) + o_z \cos(\theta_2+\theta_3+\theta_4)}\right]$$

Equation 4: Solution Set for Joint Angles 1 - 6

Next, the solutions are executed iteratively as the resulting joint angles are inputted into the corresponding servomotors and the manipulator's end-effector moves in that specified trajectory.

$$\theta_i = \theta_0 + \theta_1 t + \theta_2 t^2 + \theta_3 t^3$$

Equation 5: General Joint Equation

Initial Conditions:

$\theta(t_o = 0) = \theta_o = 0°$ $\quad\dot{\theta}(t_o = 0) = \dot{\theta}_o = 0°/s$

$\theta(t_f = 10s) = \theta_f = 90°$ $\quad\dot{\theta}(t_f = 10s) = \dot{\theta}_f = 0°/s$

Equation 6: Joint Initial Conditions

[4.0] Troubleshooting Instructions

The following section specifies troubleshooting conditions and solutions for the robotic manipulator components.

[4.1] Robotic Manipulator Joint

Table 10: Troubleshooting Scenarios for Manipulator Joints

Symptom	Possible Cause	Remedy
Multiple joints are unresponsive	No electrical power supplied to manipulator.	Make sure power supply is connected and operational.

	Damaged wiring or loose connection in cables between joints.	Replace any faulty or damaged wires.
	Incorrect code inputted into MATLAB.	Refer to created MATLAB library for appropriate code.
	Fried USB2DYNAMIXEL.	Replace USB2DYNAMIXEL.
	Load exceeds gripper capacity.	Ensure the appropriate weight as listed in specifications is used.
	Servo failure.	Replace faulty servo motor.
	Incorrect servo ID in code	Make sure the right servo ID is inputted in their appropriate order.
Joints are erratic.	Faulty Servo motors.	Inspect servo motor and replace if necessary.
	Incorrect code inputted into MATLAB.	Refer to corresponding experiment to make sure correct code is inputted
	Faulty or loose connections (bolts and fasteners/brackets).	Ensure all brackets are fastened tightly.
	Pinched wireway/ short wireway	Replace damaged wires.
	Incorrect communication between manipulator and MATLAB	Make sure the manipulator is connected correctly to MATLAB via USB2DYNAMIXEL.
Joints are sluggish	Power Supply.	Check and make sure power supply is plugged in.
	Incorrect MATLAB code.	Ensure correct code is inputted into MATLAB.
	Broken brackets and fasteners.	Replace all broken brackets.
	Faulty Servo Motors.	Replace all faulty servos in the manipulator.
	Load at or near joint's max capacity.	Reduce load and refer to specifications for maximum load.

[4.2] Robotic Manipulator Gripper

Table 11: Troubleshooting Scenarios for Manipulator Gripper

Symptom	Possible Cause	Remedy

Gripper not holding writing utensil	Loose fasteners/Loose gripper/	Tighten any loose fasteners or replace in order to secure gripper in place.
	Power Supply.	Make sure the manipulator is plugged into a working power source.
	Writing utensil does not meet specifications for manipulator.	Read specifications in order to use appropriate writing utensil.
	Too much friction between the surface and the tip of the writing utensil due to moment.	Restart MATLAB code and re-attach gripper to writing utensil.
Gripper unresponsive	Power Supply.	Make sure the manipulator is plugged into a working power source.
	Miscommunication between gripper and input code.	Ensure the correct MATLAB code is correspondent to the gripper location.
	Damaged wiring between Servo motor and gripper.	Replace any damaged wires or motors before running code.

References

[1] "API Reference." Robotis. 2010. Accessed September 20, 2016. http://support.robotis.com/en/software/dynamixel_sdk/api_reference.htm.

[2] "AX-12/ AX-12 / AX-12A." Robotis. Accessed September 20, 2016. http://support.robotis.com/en/product/dynamixel/ax_series/dxl_ax_actuator.htm.

[3] "Dynamixel SDK for Windows." Robotis. 2010. Accessed September 20, 2016. http://support.robotis.com/en/software/dynamixel_sdk/usb2dynamixel/usb2dxl_windows.htm.

[4] Mensink, Arno. "Characterization and Modeling of a Dynamixel Servo." November 2008. Accessed September 20, 2016. https://www.ram.ewi.utwente.nl/aigaion/attachments/single/1015.

[5] Niku, Saeed B. *Introduction to Robotics: Analysis, Control, Applications*. Hoboken, N.J: Wiley, 2011.

[6] "MATLAB." Robotis. 2010. Accessed September 20, 2016. http://support.robotis.com/en/software/dynamixel_sdk/usb2dynamixel/window_communication_1/matlab.htm.

[7] Rouse, Margaret. "UART (Universal Asynchronous Receiver/Transmitter)." WhatIs. February 2011. Accessed September 20, 2016. http://whatis.techtarget.com/definition/UART-Universal-Asynchronous-Receiver-Transmitter.

Contact Information

For more information and any questions related to the manual, please feel free to contact the Upper Echelon team members;

Table 12: Upper Echelon Team Contact Information

Name	Position	Email	Phone Number
Paul Alabi	Research/Manufacturing	paul.alabi@rocketmail.com	832.283.2063
Omar E. Castro	Electrical Design Analyst	oecastrom@gmail.com	832.523.4645
Sergio A. Molina	Project Leader/ Analyst	sergioamolina92@gmail.com	956.744.1133
Jonathan Sackett	Mechanical Design/ Test Planning	jonnysackett@hotmail.com	281.743.5187

List of Tables

List of Figures

List of Equations

Appendix AA

Table 26: Dynamixel SDK Library Function Reference

Dynamixel SDK Function Reference				
No	Function	Description	Input	Output
Device Control Method				
1	dxl_initialize(a,b)	Initializes the communication devices	a - COM Port Number b - Baud Rate (mbps)	0 - Unsuccessful 1 - Successful
2	dxl_terminate()	Terminates Serial Communication	None	None
Set/Get Packet Method				
3	dxl_set_txpacket_id(a)	ID is inputted into instruction packet	a - ID Number (between 0 - 254)	None
4	dxl_set_txpacket_instruction(a)	Command codes inputted	Numeric Values of a Value | Description	None

		into instruction packet	1	No Execution	
			2	Reads Data	
			3	Writes Data	
			4	Writes Data (Standby)	
			5	Initializes Motions	
			6	Restores Factory Setting	
			131	Control multiple motors simultaneously	
5	dxl_set_txpacket_length(a)	The length is inputted into Instruction Packet	a - Length		None
6	dxl_get_lowbyte(a)	Lower byte is extracted from WORD-type data	a - Word		Lower byte
7	dxl_get_highbyte(a)	High byte is extracted from WORD-type data	a - Word		Higher byte
8	dxl_get_rxpacket_length()	Checks the length value of status packet	None		Length value of Status Packet
9	dxl_get_rxpacket_parameter(a)	Checks the parameter value specified by a	a - index value		Parameter Value
10	dxl_makeword(a,b)	Changes 2-byte into WORD type	a - low byte b - high byte		WORD type data
Packet Communication Method					

11	dxl_tx_packet()	Transmits Instruction Packet	None	None
12	dxl_rx_packet()	Receives Status Packet	None	None
13	dxl_txrx_packet()	Transmits Instruction Packet and Receives Status Packet	None	None
14	dxl_get_results()	Packet communication results	None	None
High Communication Method				
15	dxl_ping(a)	Verification of selected ID with Dynamixel is checked	a - Dynamixel ID	None
16	dxl_write_byte(a,b,c)	Information of 1 byte is written on Dynamixel	a - Dynamixel ID b - Address c - Value	None
17	dxl_read_byte(a,b)	Information of 1 byte is read on Dynamixel	a - Dynamixel ID b - Address	None
18	dxl_write_word(a,b,c)	Information of 2 byte is written on Dynamixel	a - Dynamixel ID b - Address c - Value	None
19	dxl_read_word(a,b)	Information of 2 byte is read on Dynamixel	a - Dynamixel ID b - Address	None

% MATLAB Script File is Intellectual Property of Upper Echelon Engineering %

% Written by: Sergio A. Molina%%%%%%%%%%%%%%%%%%%%%%

%%%

```
% This Code Initializes Robotic Manipulator to Initial State
% Please run this code before and after experimental procedures

% Clear Workspace Variables
clear all
clc
%%
% Load the Dynamixel SDK Library and Initialize Serial Communication
loadlibrary('dynamixel','dynamixel.h');
res = calllib('dynamixel','dxl_initialize',3,1);
%%
numberOfDynamixels = 7;
init_Pos = 0;

%Broadcast id 0xFE
calllib('dynamixel','dxl_set_txpacket_id',254);

%Length is 49
%That handles position and speed for two dynamixels
calllib('dynamixel','dxl_set_txpacket_length',7*numberOfDynamixels);

%SyncWrite instruction 0x83
calllib('dynamixel','dxl_set_txpacket_instruction',131);

%Starting address
calllib('dynamixel','dxl_set_txpacket_parameter',0, 30);

%length of data to write to each dynamixel
calllib('dynamixel','dxl_set_txpacket_parameter',1, 4);
%%
%Parameters for syncwrite dynamixel id = 1
% id | position | speed
```

```
%ID = 1
calllib('dynamixel','dxl_set_txpacket_parameter',2, 1);

%Position = 0
lowByte = calllib('dynamixel','dxl_get_lowbyte',init_Pos);
highByte = calllib('dynamixel','dxl_get_highbyte', init_Pos);
calllib('dynamixel','dxl_set_txpacket_parameter',3, lowByte);
calllib('dynamixel','dxl_set_txpacket_parameter',4, highByte);

%%
%Parameters for syncwrite dynamixel id = 2
% id | position | speed
%ID = 2
calllib('dynamixel','dxl_set_txpacket_parameter',7, 2);

%Position = 0
lowByte = calllib('dynamixel','dxl_get_lowbyte',init_Pos);
highByte = calllib('dynamixel','dxl_get_highbyte', init_Pos);
calllib('dynamixel','dxl_set_txpacket_parameter',8, lowByte);
calllib('dynamixel','dxl_set_txpacket_parameter',9, highByte);
%%
%Parameters for syncwrite dynamixel id = 3
% id | position | speed
%ID =
calllib('dynamixel','dxl_set_txpacket_parameter',12, 3);

%Position = 0
lowByte = calllib('dynamixel','dxl_get_lowbyte',init_Pos);
highByte = calllib('dynamixel','dxl_get_highbyte', init_Pos);
calllib('dynamixel','dxl_set_txpacket_parameter',13, lowByte);
calllib('dynamixel','dxl_set_txpacket_parameter',14, highByte);
%%
%Parameters for syncwrite dynamixel id = 4
```

```
% id | position | speed
%ID = 4
calllib('dynamixel','dxl_set_txpacket_parameter',17, 4);

%Position = 0
lowByte = calllib('dynamixel','dxl_get_lowbyte',init_Pos);
highByte = calllib('dynamixel','dxl_get_highbyte', init_Pos);
calllib('dynamixel','dxl_set_txpacket_parameter',18, lowByte);
calllib('dynamixel','dxl_set_txpacket_parameter',19, highByte);

%%
%Parameters for syncwrite dynamixel id = 5
% id | position | speed
%ID = 5
calllib('dynamixel','dxl_set_txpacket_parameter',22, 5);

%Position = 0
lowByte = calllib('dynamixel','dxl_get_lowbyte',init_Pos);
highByte = calllib('dynamixel','dxl_get_highbyte', init_Pos);
calllib('dynamixel','dxl_set_txpacket_parameter',23, lowByte);
calllib('dynamixel','dxl_set_txpacket_parameter',24, highByte);
%%
%Parameters for syncwrite dynamixel id = 6
% id | position | speed
%ID = 6
calllib('dynamixel','dxl_set_txpacket_parameter',24, 6);

%Position = 0
lowByte = calllib('dynamixel','dxl_get_lowbyte',init_Pos);
highByte = calllib('dynamixel','dxl_get_highbyte', init_Pos);
calllib('dynamixel','dxl_set_txpacket_parameter',25, lowByte);
calllib('dynamixel','dxl_set_txpacket_parameter',26, highByte);
%%
```

```
%Parameters for syncwrite dynamixel id = 7 (A.K.A. Gripper)
% id | position | speed
%ID = 7
calllib('dynamixel','dxl_set_txpacket_parameter',27, 7);

%Position = 0
lowByte = calllib('dynamixel','dxl_get_lowbyte',init_Pos);
highByte = calllib('dynamixel','dxl_get_highbyte', init_Pos);
calllib('dynamixel','dxl_set_txpacket_parameter',28, lowByte);
calllib('dynamixel','dxl_set_txpacket_parameter',29, highByte);
%%
% Transmit Instruction Packet to Joints
calllib('dynamixel','dxl_tx_packet');

% MATLAB Script File is Intellectual Property of Upper Echelon Engineering %
% All rights reserved to Dynamixel for the SDK library used in this script %%%
% Written by: Sergio A.  Molina%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% This script resets Servos to Factory Settings
clear all
clc

% Load SDK Library
loadlibrary('dynamixel','dynamixel.h')

% Initialization of Serial Communication
calllib('dynamixel','dxl_terminate')
calllib('dynamixel','dxl_initialize',3,1);

% Construction of Instruction Packet

    % Writing Position for Joint 1
```

```
        calllib('dynamixel','dxl_set_txpacket_id', 254)
        calllib('dynamixel','dxl_set_txpacket_length', 2)
        calllib('dynamixel','dxl_set_txpacket_instruction', 6)

% Transmit Packet
calllib('dynamixel','dxl_tx_packet')

% MATLAB Script File is Intellectual Property of Upper Echelon Engineering %
% All rights reserved to Dynamixel for the SDK library used in this script %%%
% Written by: Sergio A.  Molina%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Configures Dynamixel ID to user defined value
% ID value must be between 0 - 254

clear all
clc

ID = 1;

loadlibrary('dynamixel','dynamixel.h')

% Initialization of Serial Communication
calllib('dynamixel','dxl_initialize',3,1);

% Construction of Instruction Packet

% Writing ID for Servo
calllib('dynamixel','dxl_set_txpacket_id', 254)
calllib('dynamixel','dxl_set_txpacket_length', 4)
calllib('dynamixel','dxl_set_txpacket_instruction', 3)

calllib('dynamixel','dxl_set_txpacket_parameter', 0, 3)
```

```
calllib('dynamixel','dxl_set_txpacket_parameter', 1, ID)

% Transmitting Packet
calllib('dynamixel','dxl_tx_packet'
```

Appendices B – Test Plan

List of Tables

1.0 Introduction

The purpose of this test plan is to analyze the performance of the Robotic Manipulator designed by Upper Echelon Engineering and ensure its adherence to the physical and functional specifications shown in the Appendix. Testing of the Robotic Manipulator will include evaluating the manipulator as a whole and some of its components. By ensuring that all the specifications are met, the Robotic Manipulator will be proven to be portable, reliable, accurate, capable of six degrees of freedom, as well as adhering to the requirements for the three experiments in the Fundamentals of Robotics Course at the University of Texas at San Antonio.

2.0 Scope

Instruments used for the testing of the Robotic Manipulator will be provided by Upper Echelon Engineering. Final Testing will be completed in accordance with this test plan. The testing will be conducted either in the Robotics Laboratory BSE 2.216 or team member Jonathan Sackett's house.

3.0 Features to be Tested

The features of the Robotic Manipulator to be tested are listed in Table 1 below. For reference purposes, the test numbers are described in Table 2.

Table 1. Test Compliance Matrix

Item No.	**Feature to be Tested**	**Specification Reference**	**Testing Procedure**	**Pass/Fail**
1	Load Geometry	1.1.1	T-1.1	
2	Load Volume	1.1.2	T-1.2	
3	Load Weight	1.1.3	T-1.3	
4	Gripping Area	2.2.2	T-2.1	
5	Gripping Force	2.2.3	T-2.2	
6	Power Supply Voltage	3.3.2	T-3.1	
7	Power Supply Current	3.3.2	T-3.2	
8	Manipulator Length	4.4.1	T-4.1	
9	Degrees of Freedom	5.5.0	T-5.1	
10	End Effector Navigation	5.5.1	T-5.2	
11	End Effector Position Uncertainty	9.9.1	T-5.3	
12	Max Joint Angular Positions	9.9.2	T-5.4	
13	Max Joint Angular Velocity	9.9.3	T-5.5	

14	Max Joint Angular Acceleration	9.9.4	T-5.6	
15	Torque Values	-	T-5.7	
16	Cycle Time	9.9.5	T-5.8	
17	Laboratory Experiments	6.1.0	T-6.1	
18	Servo Motor Angle Resolution	7.7.1	T-7.1	
19	Servo Motor Current	-	T-7.2	
20	Servo Motor Voltage	7.7.2	T-7.3	
21	Servo Motor Temperature	-	T-7.4	
22	Servo Motor Mass	7.7.3	T-7.5	
23	GUI	8.8.1	T-8.1	
24	Material	-	T-9.1	
25	Part Dimensions	-	T-9.2	
26	Assemblies	-	T-9.3	

4.0 Test Description

4.1 - T-1.1 Load Geometry Test
The Load Geometry Test is conducted in order to ensure the load being applied at the gripper is cubic in nature.

4.1.1 - Equipment
The following is a list of required equipment to conduct the test:

1. Load cell

4.1.2 - Procedure
Upon visual inspection, determine if the load is cubic in geometry.

4.1.3 - Data analysis
The data collected from this test will be recorded as a pass/test as it is part of the specifications listed in the Appendix. Pass is when the load is cubic is geometry. Fail is if the load is not cubic.

4.2 - T-1.2 Load Volume Test
The Load Volume Test is conducted to determine the volume of the load.

4.2.1 - Equipment
The following is a list of required equipment to conduct the test:

1. Venire Calipers

4.2.2 - Procedure
Measure the length, width, and height of the load and multiply them to obtain the volume. Record the data in the appropriate section within Table 2.

4.2.3 - Data Analysis
The data collected from this test will be recorded as a pass/test as it is part of the specifications listed in the Appendix. Pass is when the load is less than or equal to one cubic inch. Fail is if the load is greater than one cubic inch.

4.3 - T-1.3 Load Weight Test
The Load Weight Test is conducted in order to determine how much load the Robotic Manipulator can handle without failure.

4.3.1 - Equipment
The following is a list of required equipment to conduct the test:

1. Weight set (0-250g set)
2. MATLAB script file for initialization

4.3.2 - Procedure
The manipulator is placed in the initial state by the provided GUI. Weights from the weight set are added to the end effector in increments until the servos either stall, or are incapable of movement, or the manipulator experiences structural failure. Record the last weight before failure is experienced within the appropriate entries within Table 2.

4.3.3 - Data Analysis
The data is tabulated under the respective loads allowing the user to determine at which quantity of load the end-effector would experience failure. "Pass" is tabulated if the load value recorded is greater than or equal to 0.25 lbf. "Fail" is tabulated if the recorded load is less than 0.25 lbf. Even if the test is failed, the specification might still be met.

Table 2. Load Testing Table

Load Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**

T-1.1 Load Geometry	Cubic		
T-1.2 Load Volume	≤ 1 in^3		
T-1.3 Load Weight	≥ 0.25 lbf		
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

4.4 - T-2.1 Gripping Area Test
The Gripping Area Test is conducted in order to determine if the specification for the Gripper is met as listed in the Appendix.

4.4.1 - Equipment
The following is a list of required equipment to conduct the test:

1. Venire Caliper
2. Metric Ruler.

4.4.2 - Procedure
The procedure for this test details using Vernier calipers and a metric ruler to measure the gripping area of end-effector. The length, width, and height of the area would be measured in order to determine if the gripping area matches the defined specifications.

4.4.3 - Data Analysis
The data collected from the Gripping Area Test would be compared to the gripping area listed in the Specifications located in the Appendix.

4.5 - T-2.2 Gripping Force Test
The Gripping Force Test is conducted in order to measure if the clamp force generated by the Gripper meets the Specifications listed in the Appendix.

4.5.1 - Equipment
The following is a list of required equipment to conduct the test:

1. Digital Hand Dynamometer

4.5.2 - Procedure
The gripping force of the robotic manipulator is measured by placing the Dynamometer between the hands of the gripper. Next, the servo actuates the gripper to “closed” position. The value measured by the Dynamometer is recorded into the appropriate section of Table 3.

4.5.3 - Data Analysis
The data collected for this test is analyzed according to the maximum gripping force obtained, which is produced by the robotic manipulator’s end-effector. If the force recorded in Table 3 matches the required force specified in the specification listed in the Appendix then a “Pass” status is tabulated within the appropriate section.

Table 3. Gripper Testing Table

Gripper Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-2.1 Gripping Area	1 $in.^2$		
T-2.2 Gripping Force	$\geq$ 0.35 lbf		
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

4.6 - T-3.1 Power Supply Voltage Test
The Power Supply Voltage Test is conducted to ensure that voltage produced by the power supply is within the specifications.

4.6.1 - Equipment
The following is a list of required equipment to conduct the test:

1. Digital Multi-Meter

4.6.2 - Procedure
Using a digital multi-meter, the voltage produced by the power supply would be tested with load (when the power supply is connected to the robotic manipulator, and is in full function); the other

test would be conducted when the power supply is without load (not connected to the robotic manipulator).

4.6.3 - Data Analysis

The data collected from the power supply voltage test would include data from when the power supply is under load and not under load. The specifications for the required voltage are listed in the attached Appendix.

4.7 - T-3.2 Power Supply Current Test

The Power Supply Current Test is conducted to ensure that current produced by the power supply is within the specifications.

4.7.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Digital Multi-Meter

4.7.2 - Procedure

Using a digital multi-meter, the current produced by the power supply would be tested with a load (when the power supply is connected to the robotic manipulator, and is in full function) and without a load (not connected to the robotic manipulator).

4.7.3 - Data Analysis

The data collected from the power supply current test would include data from when the power supply is under load and not under load. The specifications for the required voltage are listed in the attached Appendix.

Table 4. Power Supply Testing Table

Power Supply Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-3.1 Power Supply Voltage	≤ 12V-DC		
T-3.2 Power Supply Current	10A		
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

4.8 - T-4.1 Manipulator Length Test

The Manipulator Length Test is conducted in order to see that the specifications are met for the required total length of the Robotic Manipulator.

4.8.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Venire Calipers
2. Metric Ruler

4.8.2 - Procedure

Using Venire Calipers and a metric ruler, measure and record the total length of the manipulator after assembly. The measurement should include the length of each link, and the total length of the manipulator's "arm".

4.8.3 - Data Analysis

The data collected from the Manipulator Length Test would be directly compared to the theoretical values that are listed in the Specifications that are located in the Appendix.

Table 5. Manipulator Length Test Table

Manipulator Length Test			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-4.1 Manipulator Length	1.5 ft $\leq$ L $\leq$ 2 ft		
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

4.9 - T-5.1 Degrees of Freedom Test

The Degrees of Freedom Test is conducted in order to determine if the Robotic Manipulator meets the specification listed in the Appendix.

4.9.1 - Equipment

The following is a list of required equipment to conduct the test:

1. MATLAB
2. Simulink

4.9.2 - Procedure

Using the MATLAB library, run a code via Simulink and MATLAB to ensure the specified degrees of freedom are met.

4.9.3 - Data Analysis

There is no data collected as this is a pass or fail test. Pass means the specifications were met (6-DOF), and fail means that the specification was not met.

Table 6. Degree of Freedom Test Table

Degree of Freedom Test			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.1 Degrees of Freedom	6		
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

4.10 - T-5.2 End Effector Navigation Test

The End Effector Navigation Test is conducted in order to determine how efficient the Robotic Manipulator's End-Effector operates during load (running an experiment).

4.10.1 - Equipment

The following is a list of required equipment to conduct the test:

1. MATLAB
2. Simulink

4.10.2 - Procedure

The end-effector navigation test will be conducted using the codes inputted into MATLAB during the experiments and see how the end-effector moves efficiently to the inputted code and if the results are in conjunction with the specified requirements that need to be met.

4.10.3 - Data Analysis

There would be no data analysis for this test as it is a pass/fail test.

Table 7. End Effector Navigation Test Table

End Effector Navigation Test			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.2 End Effector Navigation	1/2 Cartesian Space		
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

4.11 - T-5.3 End Effector Position Uncertainty Test

The end effector position uncertainty test is concerned with validating that the robotic manipulator's end effector is within the specified level of uncertainty for position.

4.11.1 - Equipment

The following is a list of the equipment required to successfully execute this test:

1. Metric or Imperial style of unit Ruler

2. Marker or writing utensil
3. Weighted Set 10 to 1000g for loading condition testing
4. Graphical User Interface

4.11.2 - Procedure

To verify that the end effector position is within the specified tolerance range its position must be measured under no-load and load conditions. A ruler and marker or writing utensil is needed in order to determine the x, y, and z position of the end effector relative to the origin of joint 1. The steps for testing the end effector position will be described respectively.

Steps for No load End Effector Position Uncertainty Test:

1. Ensure robotic manipulator is powered on
2. Verify the manipulator is in its initial state by selecting the "Reset" button on GUI
3. Open GUI via MATLAB
4. Using the GUI maneuver joint 1 to 45º
5. Take ruler and measure the x, y, and z values respectively
6. Tabulate results in appropriate rows and columns of Table 7
7. Calculate the error

a. Subtract the desired value from the obtained value

b. Take the absolute value of the obtained result

c. Populate the relevant sections of Table 7

8. Repeat steps 4 - 7 for joints 2 - 7

Steps for Load End Effector Position Uncertainty Test:

1. Ensure robotic manipulator is powered on
2. Verify the manipulator is in its initial state by selecting the "Reset" button on GUI
3. Open GUI via MATLAB
4. Begin with the lowest specified value of the load weight provided in table 7
5. Securely apply the required weight(s) to the end effector to match the value from table 7
6. Using the GUI maneuver joint 1 to 45º
7. Take ruler and measure the x, y, and z values respectively
8. Tabulate results in appropriate rows and columns of table 7
9. Repeat steps 5 - 8 for joints 2 - 6

4.11.3 - Data Analysis

By comparing the true percent error value of the end effector obtained for each joint to the tolerance level the test will either fail or pass. If the error is greater than 3 in, the test is a fail. If the error is less than or equal to 3 in then it is a pass.

4.12 - T-5.4 Max Joint Angular Positions Test

The Max Joint Angular Position Test is intended is meant to measure the range of angles the motors could output and to ensure the corresponding specification is met.

4.12.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Graphical User Interface
2. Protractor
3. Writing utensil
4. Ten sheets of 11 in. by 17 in. white paper
5. Clear Tape

4.12.2 - Procedure

Each joint will be tested one at a time. In order to measure the range of angles, a sheet of white paper will be taped to the manipulator such that the joint being evaluated is roughly at the center of that sheet. First, the maximum range of joint 1 will be measured. The start position will be traced onto the paper and annotated then the manipulator will be actuated via the GUI to the max angle and the final position will be traced and annotated. Next, the person(s) conducting the test will calculate the difference in angles and make a determination of the outcome of the experiment. This process will be repeated for joints 2 – 7.

The following is a list of the steps required for completion of the test on joint 1:

1. Open GUI
2. Verify the manipulator is in its initial state by selecting the "Reset" button on GUI
3. Tape a 11 in. by 17 in. sheet of paper underneath the base of the manipulator
4. Mark a line (parallel) directly underneath the manipulator on the paper
5. Identify the line by writing "Min" within the proximity of the line

6. Increment the angle values for joint 1 from 0 - 270º
7. Repeat step 4
8. Identify the line by writing "Max" within the proximity of the line
9. Measure the change in angle using the protractor

The following is a list of the steps required for completion of the test on joint 2 - 6:

1. Open GUI
2. Verify the manipulator is in its initial state by selecting the "Reset" button on GUI
3. Tape a 11 in. by 17 in. sheet of paper adjacent to the link nearest the manipulator
4. Mark a line parallel to the manipulator on the sheet of paper
5. Identify the line by writing "Min" within the proximity of the line
6. Increment the angle values for joint 1 from 0 - 180º
7. Repeat step 4
8. Identify the line by writing "Max" within the proximity of the line
9. Measure the change in angle using the protractor

4.12.3 - Data Analysis

Comparing the measured values to those provided will determine a "Pass" for the maximum angular position of the joints if the error is less than 20%. For each test, the technician will record with a "Pass" for confirmation that the joint passed or "Fail" if the joint failed to reach the criteria.

4.13 - T-5.5 Max Joint Angular Velocity

The Max Joint Angular Velocity Test is intended is to measure the range of angular velocities that the motors could output and to ensure that the corresponding specification for joint angular velocity for each joint is successfully met.

4.13.1 - Equipment

The following is a list of required software and physical hardware necessary for executing the test:

1. MATLAB Script File for Angular Velocity Sweep

4.13.2 - Procedure

In order to test for the maximum angular velocity the robotic manipulator joints would have to be subjected to a sweep in angular velocity one at a time starting from the initial state. A fifth-order polynomial, time-dependent function that accounts for the maximum joint angular velocity within its unique solution will be used to control the motors to the desired angular velocity. The script file is created in MATLAB and will automatically execute the sweep upon execution. In addition, the file will be outputting the measured joint velocities from each individual motor into the “Command Window” of MATLAB.

4.13.3 - Data Analysis

The data will be displayed in the “Command Window” where the technician will be able to tabulate the results into the following table 14. The angular velocity values will be tabulated for each joint and will either “Pass” or “Fail” based on whether the values of the angular velocity (ω) are within the bounds, $0 \leq \omega \leq 20°/sec$. If a joint doesn’t reach at least 50% of the maximum bound of 20°/sec then the technician will specify “Fail” on the table 14 and if it does surpass that measure then the technician will specify “Pass”.

4.14 - T-5.6 Max Joint Angular Acceleration Test

The Max Joint Angular Acceleration Test is intended is to measure the range of angular acceleration that the motors could output and to ensure that the corresponding specification for joint angular velocity for each joint is successfully met.

4.14.1 - Equipment

The following is a list of required software and physical hardware necessary for executing the test:

1. MATLAB Script File for Angular Acceleration Sweep

4.14.2 - Procedure

In order to test for the maximum angular acceleration the robotic manipulator joints would have to be subjected to a sweep in angular acceleration one at a time starting from the initial state. A polynomial, time-dependent function, which accounts for the maximum joint angular acceleration within its unique solution, will be used to control the motors to the desired angular acceleration. The script file is created in MATLAB and will automatically execute the sweep upon execution. In addition, the file will be outputting the measured joint acceleration from each individual motor into the “Command Window” of MATLAB.

4.14.3 - Data Analysis

The data will be displayed in the “Command Window” where the technician will be able to tabulate the results into the following table 14. The angular velocity values will be tabulated for each joint and will either “Pass” or “Fail” based on whether the values of the angular acceleration (α) are within the bounds, $0 \leq \alpha < 10°/sec^2$ .If a joint doesn’t reach at least 50% of the maximum bound of 20°/sec then the technician will specify “Fail” on the table 14 and if it does surpass that measure then the technician will specify “Pass”.

4.15 - T-5.7 Torque Values Test

The purpose of this test is to ensure the output torque of the servo motors do not exceed the motors rated stall torque value provided by the Dynamixel AX-12A datasheet.

4.15.1 - Equipment

The necessary equipment and software for conducting the test are listed below:

1. Fully Assembled Robotic Manipulator
2. MATLAB Script file for Measuring Torque Values

4.15.2 - Procedure

The following is a step-by-step list necessary to execute this experiment

1. Ensure robotic manipulator is powered on
2. Open MATLAB Script file for Measuring Torque Values
3. Begin with the lowest specified value of the weight of load from table 14
4. Securely apply the required weight(s) to the end effector to match the value from table 14
5. “Run” the MATLAB Script file for Measuring Torque Values
6. Tabulate the results obtained in the “Command Window” pertaining to the relevant servo
7. Increment the load and repeat process until the 0.25 lbf is tested
8. Repeat steps 3 - 6 for joints 2 - 6

4.15.3 - Data Analysis

In order for the manipulator to be operating safely during actuation the motors must not exceed ⅘ of the rated stall torque for the Dynamixel AX-12A. Furthermore, if any of the data values obtained from the various trails exceed ⅘ of the rated stall torque for the Dynamixel AX-12A then a “Fail” will be reported, otherwise “Pass” will be reported.

4.16 - T-5.8 Cycle Time Test

The Cycle Time Test will test the manipulator performance for whether it meets the cycle time specification. The following four conditions will be imposed on the manipulator in order to evaluate all of the working conditions that it will be exposed to:

1. Static Condition with No Load
2. Static Condition with Loads at End Effector
3. Dynamic Condition with No Load
4. Dynamic Condition with Loads at End Effector

4.16.1 - Equipment

1. MATLAB Script file for Executing Trajectory Generation in Desired Cycle Time
2. Fully Assembled Robotic Manipulator
3. Stopwatch

4.16.2 - Procedure

The Dynamixel AX-12A is not capable of measuring Torque values directly so the method of measurement will be indirect. Two methods will be discussed in order to cross-reference the two experimental outcomes in order to best evaluate the criteria. The following conditions will be imposed on the full assembly of the robotic manipulator:

1. Dynamic Condition with No Load
2. Dynamic Condition with Loads at End Effector

The list of steps for the "Dynamic Condition with No Load":

1. Ensure robotic manipulator is powered on
2. Load the "Trajectory Generation in Desired Cycle Time" script file into MATLAB
3. Visually inspect to ensure the manipulator is in initial state configuration
4. Run the script file
5. Track the time it takes for the manipulator to complete the trajectory
6. Tabulate data into table 15
7. Calculate the difference in time values from the measured and theoretical
8. Tabulate results
9. Visually inspect the trajectory generated to ensure it is following the specified path

The list of steps for the "Dynamic Condition with Loads at End Effector":

1. Ensure robotic manipulator is powered on
2. Load the "Trajectory Generation in Desired Cycle Time" script file into MATLAB
3. Visually inspect to ensure the manipulator is in initial state configuration
4. Mount the necessary weight at the end effector according to table 14
5. Ensure the weight is properly secured
6. Run the script file
7. Track the time it takes for the manipulator to complete the trajectory
8. Tabulate data into table 14

9. Calculate the difference in time values from the measure to the theoretical
10. Tabulate results
11. Visually inspect the trajectory to ensure it is following the specified path
12. Tabulate visually obtained results of the trajectory path

4.16.3 - Data Analysis

In order to ensure the manipulator performs to the specified cycle time the maximum amount of error resulting from the Cycle Time Test is within ±1% of 10 seconds. If the error value resulting from the data is not within the specified range of acceptable deviation then the technician is to specify "Fail" in the relevant columns and "Pass" if it does fall within the range of acceptable values.

Table 8. Static Loading Test Table for Servo 1

Static Loading Tests									
Servo 1									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									

Team Member Conducting Test:
Team Member Signature:

Table 9. Static Loading Test Table for Servo 2

Static Loading Tests									
Servo 2									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 10. Static Loading Test Table for Servo 3

Static Loading Tests									
Servo 3									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 11. Static Loading Test Table for Servo 4

Static Loading Tests

Servo 4									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 12. Static Loading Test Table for Servo 5

Static Loading Tests									
Servo 5									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**

T-5.7 Torque Value									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 13. Static Loading Test Table for Servo 6

Static Loading Tests									
Servo 6									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value									
T-7.2 Servo									

Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 14. Dynamic Loading Test Table for Servo 1

Dynamic Loading Tests									
Servo 1									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty									
T-5.4 Max Joint Angular Position									
T-5.5 Max Joint									

Angular Velocity									
T-5.6 Max Joint Angular Acceleration									
T-5.7 Torque Value									
T-5.8 Cycle Time									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 15. Dynamic Loading Test Table for Servo 2

Dynamic Loading Tests
Servo 2

	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Requir ed Value**	**Measur ed Value**	**Pass/F ail**	**Requir ed Value**	**Measur ed Value**	**Pass/F ail**	**Requir ed Value**	**Measur ed Value**	**Pass/F ail**
T-5.3 End Effector Position Uncertaint y									
T-5.4 Max Joint Angular Position									
T-5.5 Max Joint Angular Velocity									
T-5.6 Max Joint Angular Accelerati on									
T-5.7 Torque Value									
T-5.8 Cycle Time									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperat ure									

Date Conducted:
Team Member Conducting Test:
Team Member Signature:

Table 16. Dynamic Loading Test Table for Servo 3

Dynamic Loading Tests									
Servo 3									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty									
T-5.4 Max Joint Angular Position									
T-5.5 Max Joint Angular Velocity									
T-5.6 Max Joint Angular Acceleration									
T-5.7 Torque Value									

T-5.8 Cycle Time									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 17. Dynamic Loading Test Table for Servo 4

Dynamic Loading Tests									
Servo 4									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty									

T-5.4 Max Joint Angular Position									
T-5.5 Max Joint Angular Velocity									
T-5.6 Max Joint Angular Acceleration									
T-5.7 Torque Value									
T-5.8 Cycle Time									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 18. Dynamic Loading Test Table for Servo 5

Dynamic Loading Tests									
Servo 5									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty									
T-5.4 Max Joint Angular Position									
T-5.5 Max Joint Angular Velocity									
T-5.6 Max Joint Angular Acceleration									
T-5.7 Torque Value									
T-5.8 Cycle Time									
T-7.2 Servo Motor Current									
T-7.3 Servo									

Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

Table 19. Dynamic Loading Test Table for Servo 6

Dynamic Loading Tests									
Servo 6									
	0 lbf			0.125 lbf			0.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty									
T-5.4 Max Joint Angular Position									
T-5.5 Max Joint Angular Velocity									
T-5.6 Max Joint									

Angular Acceleration									
T-5.7 Torque Value									
T-5.8 Cycle Time									
T-7.2 Servo Motor Current									
T-7.3 Servo Motor Voltage									
T-7.5 Servo Motor Temperature									
Date Conducted:									
Team Member Conducting Test:									
Team Member Signature:									

4.17 - T-6.1 Laboratory Experiment Tests

This test ensures that the manipulator successfully and easily interfaces with the three laboratory experimental manuals included the manipulator package.

4.17.1 - Equipment

The following software and hardware is necessary to execute the Laboratory Experiment Test:

1. Full assembly of robotic manipulator
2. Three complete laboratory manuals

3. Complete Simulink library for robotic manipulator

4.17.2 - Procedure

The Laboratory Experiment Test requires the technician to complete the three experimental laboratories sequentially, one after the other. In addition, the technician must follow the exact instructions provided by the laboratory manuals.

4.17.3 - Data Analysis

The data obtained from the Laboratory Experiment Test will either be a "Pass" or "Fail" status depending on the outcome of each experiment. To add, if the technician was able to successfully complete every instruction within the laboratory manual with no error or evident problem, a "Pass" will be tabulated in the respective section of the test outcome table 20 and "Fail" otherwise.

Table 20. Lab Experiments Testing Table

Lab Experiments Testing	
Featured Test	Pass/Fail
T-6.1 Lab Experiments	
Experiment 1 - Forward Kinematics	
Experiment 2 - Inverse Kinematics	
Experiment 3 - Trajectory Planning	
Date Conducted:	
Team Member Conducting Test:	
Team Member Signature:	

4.19 - T-7.2 Servo Motor Current Test

The Servo Motor Current Test is a test to ensure that the current being supplied to each motor during actuation is not exceeding the maximum current as stated in the Dynamixel AX-12A datasheet.

4.18.1 - Equipment

1. Fully assembled robotic manipulator
2. MATLAB Script file for Executing Trajectory Generation in Desired Cycle Time
3. Digital Multi-Meter (capable of measuring 120V, 10A)

4.18.2 - Procedure

The following is a list of steps required to execute the test.

1. Ensure that the robotic manipulator is turned “On”
2. Load the “robot_initialize” MATLAB script file
3. Initialize the robotic manipulator to the “initial state” by executing the script file
4. Load the “robot_trajectory_gen” script file
5. Connect, in series, the positive terminal of the DMM to the “Voltage” pin of the robotic manipulator for the first joint
6. Connect, in series, the negative terminal of the DMM to the “Ground” pin of the robotic manipulator for the first joint
7. Visually ensure that all wiring is properly connected
8. “Run” the “robot_trajectory_gen” script file
9. Tabulate the maximum current value throughout the execution time of the script file
10. Tabulate results in table 14
11. Repeat steps 1 - 11 for joints 2 - 6

4.18.3 - Data Analysis

The maximum current values for each joint will be analyzed and compared to the Dynamixel AX-12A maximum current rating. If any of the joint current values exceed 900mA then the “Fail” criteria is established in the appropriate entry box. If any of the joint current values are less than or equal to 900mA then the “Pass” criteria is established in the appropriate entry box.

4.19 - T-7.2 Servo Motor Voltage Test

The Servo Motor Voltage Test is a test to ensure that the voltage being supplied to each motor during actuation is not exceeding the maximum voltage as stated in the Dynamixel AX-12A datasheet.

4.19.1 - Equipment

1. Fully assembled robotic manipulator
2. MATLAB Script file for Executing Trajectory Generation in Desired Cycle Time
3. Digital Multi-Meter (capable of measuring 120V, 10A)

4.19.2 - Procedure

The following is a list of steps required to execute the test.

1. Ensure that the robotic manipulator is turned “On”
2. Load the “robot_initialize” MATLAB script file
3. Initialize the robotic manipulator to the “initial state” by executing the script file
4. Load the “robot_trajectory_gen” script file
5. Connect, in parallel, the positive terminal of the DMM to the “Voltage” pin of the robotic manipulator for the first joint
6. Connect, in parallel, the negative terminal of the DMM to the “Ground” pin of the robotic manipulator for the first joint
7. Visually ensure that all wiring is properly connected
8. “Run” the “robot_trajectory_gen” script file
9. Tabulate the maximum voltage value throughout the execution time of the script file
10. Tabulate results in table 14
11. Repeat steps 1 - 11 for joints 2 - 6

4.19.3 - Data Analysis

The maximum voltage values for each joint will be analyzed and compared to the Dynamixel AX-12A maximum voltage rating. If any of the joint voltage values exceed 12V then the “Fail” criteria is established in the appropriate entry box. If any of the joint current values are less than or equal to 12V then the “Pass” criteria is established in the appropriate entry box.

4.20 - T-7.3 Servo Motor Mass Test

The Servo Motor Mass Test is a test to determine whether the mass of the servo’s adhered to the Dynamixel AX-12A datasheet.

4.20.1 - Equipment

The following equipment is necessary in order to execute the Servo Motor Mass Test:

1. 7 of the Dynamixel AX-12A Servos
2. Digital Scale

4.20.2 - Procedure

The following list is a step-by-step outline of the Servo Motor Mass Test:

1. Gather all 7 Dynamixel AX-12A servos that belong to the robotic manipulator
2. Please ensure that no cell-phones or powered electronic devices are within a 1 foot radius of the digital scale as this may interfere with the results
3. Turn the digital scale to "On" mode
4. Set the units of measurement on the digital scale as specified in the column entry for "Mass of Servo 1"
5. Calibrate the scale to a value of zero reading
6. Place servo 1, as identified by the back of the servo housing under "ID", on top of the platform
7. Record the value of the mass in appropriate entry of the table 22
8. Repeat steps 2-7 for servo motors 2-7

4.20.3 - Data Analysis

The experimental values of the servomotors as obtained from the "Servo Motor Mass Test" must be no greater than the value of 60g. If any of the motor mass' experimental values exceed 60g then "Fail" must be tabulated in the appropriate section of that particular motor, otherwise the "Pass" status is tabulated.

4.21 - T-7.5 Servo Motor Temperature Test

The purpose of "Servo Motor Temperature Test" is to determine whether the Temperature values of the servomotors during operation exceed the maximum allowable temperature as specified in the Dynamixel AX-12A datasheet.

4.18.1 - Equipment

The following is a list necessary to execute the "Servo Motor Temperature Test":

1. Full Assembly of robotic manipulator
2. MATLAB Script file for Temperature measurements

4.18.2 - Procedure

The following list is a step-by-step outline of the Servo Motor Temperature Test:

1. Ensure robotic manipulator is properly assembled
2. Ensure that the robotic manipulator is powered "On"
3. Load the "servo_initialization" script file in MATLAB
4. "Run" the "servo_initialization" script file in MATLAB
5. Load the "temp_read_servo" script file in MATLAB

6. Tabulate the temperature values for each of the seven joints provided in the "Command Window" onto table 14
7. Repeat steps 3 and 4
8. Turn off the robotic manipulator

4.18.3 - Data Analysis

The servomotors for the robotic manipulator must be within a strict temperature range of -5℃≤ T ≤ 70℃ as specified by the Dynamixel AX-12A datasheet during operation in order to not fail. If any of the measured values from the servomotors after the execution of the "Servo Motor Temperature Test" falls outside of the temperature range previously mentioned then the "Fail" status is to be reported, otherwise "Pass" is reported.

Table 22. Servo Motor Testing Table

Servo Motor Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-7.1 Servo Motor Angle Resolution			
T-7.3 Servo Motor Mass			
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

4.21 - T-8.1 GUI Test

The purpose of the GUI Test is to ensure that the robotic manipulator properly interfaces with the graphical user interface through the following two modes of actuation:

1. Forward kinematics Section

a. Actuation of each joint using the "slider" feature

2. Inverse Kinematics Section

. Actuation via end-effector position specification (x,y,z)

4.21.1 - Equipment

The following is a list necessary to execute the "GUI Test":

1. Full assembly of robotic manipulator
2. Graphical User Interface

4.21.2 - Procedure

The following list is a step-by-step outline of the "GUI Test":

1. Ensure robotic manipulator is properly assembled
2. Ensure that the robotic manipulator is powered "On"
3. Load the GUI via MATLAB
4. Initialize "COMPort" and "Baud Rate" by inputting their respective value

a. Note: Follow the Operations Manual to determine "COMPort" number from computer

b. Select a "Baud Rate" value of 1

5. Vary joint angle 1 from 0 to 30º within the "Forward Kinematics" panel
6. Input a value of x = -6, y = 6, and z =8 within the "Inverse Kinematics" panel
7. Select "Enter"
8. Repeat step 5 for joint angles 2 - 7

4.21.3 - Data Analysis

The criteria established for this test is based on visual inspection. If the manipulator does not respond (actuate) to either of the joint angle or position inputs then the "Fail" status is reported, otherwise the "Pass" status is reported.

Table 21. GUI Test Table for Servo 1

GUI Test						
Servo 1						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI						
Date Conducted:						
Team Member Conducting Test:						
Team Member Signature:						

Table 22. GUI Test Table for Servo 2

GUI Test						
Servo 2						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI						
Date Conducted:						
Team Member Conducting Test:						
Team Member Signature:						

Table 23. GUI Test Table for Servo 3

GUI Test						
Servo 3						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI						
Date Conducted:						
Team Member Conducting Test:						
Team Member Signature:						

Table 24. GUI Test Table for Servo 4

GUI Test						
Servo 4						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI						
Date Conducted:						
Team Member Conducting Test:						
Team Member Signature:						

Table 25. GUI Test Table for Servo 5

GUI Test						
Servo 5						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI						
Date Conducted:						
Team Member Conducting Test:						
Team Member Signature:						

Table 26. GUI Test Table for Servo 6

GUI Test						
Servo 6						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI						
Date Conducted:						
Team Member Conducting Test:						
Team Member Signature:						

Table 27. GUI Test Table for Servo 7

GUI Test						
Servo 7						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI						
Date Conducted:						
Team Member Conducting Test:						
Team Member Signature:						

4.24 - T-9.1 Material Test

The material test is conducted in order to verify that the specified material is being used in the fabrication process.

4.22.1 - Equipment

In order to complete the material test a technical data sheet for PLA generated by Ultimaker will be used to confirm the material meets specification.

4.22.2 - Procedure

The team member who is to conduct the test should refer to table 28 while simultaneously referring to the technical data sheet spoken of in section 4.22.1. The team member is required to visually confirm with said technical data sheet that the material on hand that was purchased is PLA.

4.22.3 - Data Analysis

No data analysis will be generated from this test.

Table 28. Material Test Table

Material Test			
Featured Test	**Required**	**Confirmation**	**Pass/Fail**
T-9.1 Material	PLA		
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

4.25 - T-9.2 Part Dimensions Test

The part dimensions test is conducted in order to ensure all values adhere to the drawing package and manipulator length specification is met.

4.23.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Venire Caliper
2. Required Parts

4.23.2 - Procedure

Begin with the first printed link and proceed to measuring both the total length and the thickness of the link and cross check these values specified in the drawing package. If values match up, the team member conducting the test should indicate the test has been passed, otherwise the test has been failed. The same procedures should be taken for the second and third link printed. The base plate and base plate cover follow the same routine except the measure values being considered are the width and height. These values too need to be cross referenced with the drawing package and checked off on.

4.23.3 - Data Analysis

Five measurements for each dimension is expected to be recorder and the average of all 5 values will deem the measure value. Standard deviation will be taken to make sure tolerances are met in order to properly check off on tests. All data analysis will be recorder with Excel.

Table 29. Part Dimensions Testing Table

Part Dimensions Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-9.2 Part Dimensions			
Link 1 Length			
Link 1 Thickness			
Link 2 Length			
Link 2 Thickness			
Link 3 Length			
Link 3 Thickness			
Base Plate Width			
Base Plate Height			
Base Plate Cover Height			
Base Plate Cover Width			
Date Conducted:			
Team Member Conducting Test:			

Team Member Signature:

4.26 - T-9.3 Assemblies Test

The assemblies test will be administered to each assembly in order to ensure proper installation of each piece.

4.24.1 - Equipment

No equipment is required in order to conduct the assemblies test. All that is required is the assemblies themselves.

4.24.2 - Procedure

Begin with the first assembly by visually inspecting all pieces have been assembled correctly based off the assembly instructions provided in the drawing package. The team member conducting the experiment should verify all steps have been followed properly in order to ensure the pieces have properly been assembled.

4.24.3 - Data Analysis

No data collection or analysis will be acquired from this test.

Table 30. Assembly Testing Table

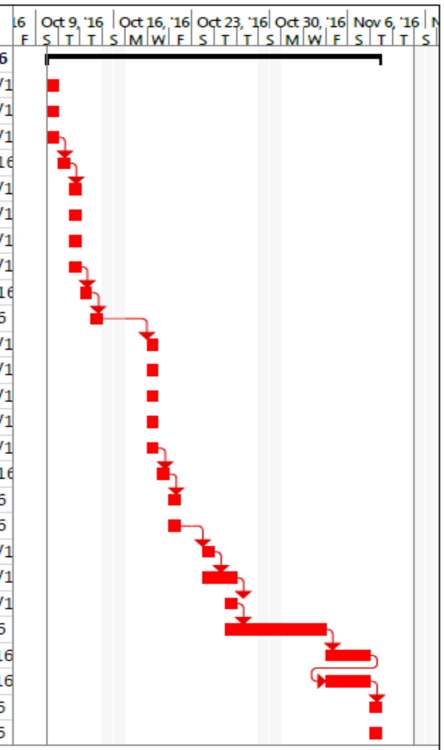
Assembly Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-9.3 Assemblies			
Assembly 1 Mechanical Connections			
Assembly 2 Mechanical Connections			
Assembly 2 Electrical Connections			
Assembly 3 Mechanical Connections			
Assembly 4 Mechanical Connections			
Assembly 5 Mechanical Connections			
Assembly 6 Mechanical Connections			

Assembly 7 Mechanical Connections			
Assembly 8 Mechanical Connections			
Assembly 9 Mechanical Connections			
Assembly 10 Mechanical Connections			
Date Conducted:			
Team Member Conducting Test:			
Team Member Signature:			

5.0 Schedule

All tests will be conducted either in the Robotics and Motion Laboratory located in the Biotechnology, Sciences and Engineering Building (BSE) 2.216 or at team member Jonathan Sackett's apartment. In order to maintain project time-management during the execution of the test plan a Test Plan Schedule was created. The test plan will begin 10/10/2016 (Monday) and will last for a total duration of 25 days, ending on 11/11/2016. The Gantt chart for the test plan reflects an earlier deadline for the test which is on 11/08/2016. The team has established that the test plan schedule won't surpass 11/11/2016 giving us 4 days to complete any unfinished testing or data collecting. The deadline gives the team 2 weeks to complete the writing of the Final report which is due on 11/22/2016. The test plan schedule is displayed below in figure 1.

ID		Task Mode	Task Name	Duration	Start	Finish
1			**Project Duration**	**22 days**	**Mon 10/10/1**	**Tue 11/8/16**
2			T-1.1: Load Geometry	1 day	Mon 10/10/1	Mon 10/10/1
3			T-1.2: Load Volume	1 day	Mon 10/10/1	Mon 10/10/1
4			T-1.3: Load Weight	1 day	Mon 10/10/1	Mon 10/10/1
5			T-2.1: Gripping Area	1 day	Tue 10/11/16	Tue 10/11/16
6			T-2.2: Gripping Force	1 day	Wed 10/12/1	Wed 10/12/1
7			T-3.1: Power Supply Voltage	1 day	Wed 10/12/1	Wed 10/12/1
8			T-7.1: Servo Motor Angle Resolution	1 day	Wed 10/12/1	Wed 10/12/1
9			T-3.2: Power Supply Current	1 day	Wed 10/12/1	Wed 10/12/1
10			T-9.1: Material	1 day	Thu 10/13/16	Thu 10/13/16
11			T-7.4: Servo Motor Mass	1 day	Fri 10/14/16	Fri 10/14/16
12			T-4.1: Manipulator Length	1 day	Wed 10/19/1	Wed 10/19/1
13			T-5.1: Degrees of Freedom	1 day	Wed 10/19/1	Wed 10/19/1
14			T-5.2: End Effector Navigation	1 day	Wed 10/19/1	Wed 10/19/1
15			T-9.2: Part Dimensions	1 day	Wed 10/19/1	Wed 10/19/1
16			T-9.3: Assemblies	1 day	Wed 10/19/1	Wed 10/19/1
17			T-5.3: End Effector Position Uncertainty	1 day	Thu 10/20/16	Thu 10/20/16
18			T-5.4: Max Joint Angular Positions	1 day	Fri 10/21/16	Fri 10/21/16
19			T-5.5: Max Joint Angular Velocity	1 day	Fri 10/21/16	Fri 10/21/16
20			T-5.6: Max Joint Angular Acceleration	1 day	Mon 10/24/1	Mon 10/24/1
21			T-5.7: Torque Values	3 days	Mon 10/24/1	Wed 10/26/1
22			T-5.8: Cycle Time	1 day	Wed 10/26/1	Wed 10/26/1
23			T-6.1: Laboratory Experiments	7 days	Wed 10/26/1	Thu 11/3/16
24			T-7.2: Servo Motor Current	2 days	Fri 11/4/16	Mon 11/7/16
25			T-7.3: Servo Motor Voltage	2 days	Fri 11/4/16	Mon 11/7/16
26			T-7.5: Servo Motor Temperature	1 day	Tue 11/8/16	Tue 11/8/16
27			T-8.1: GUI	1 day	Tue 11/8/16	Tue 11/8/16



Project: Test Plan Schedule
Date: Tue 10/4/16

Task — Summary

Figure 1: Test Plan Schedule Overview

Appendix BB

Performance Specifications

1. **Load:**
    1. Cubic geometry
    2. Maximum volume of 1 in.3
    3. Weight ≤ 0.25 lbf
2. **Gripper:**
    1. Purchase off the shelf
    2. Maximum grip area when fully open = 1 in.^{2}
    3. Clamp force ≥ 0.35 lbf
3. **Power Supply:**
    1. Tethered connection
    2. AC-DC converter to supply ≤ 12 V DC

4. **Length:**
    1. Arm length between 1.5 and 2 feet

Functional Specifications

5. Manipulator must work with 6 degrees of freedom

5.1. End-effector must navigate through at least ½ of Cartesian coordinate

6. Fulfill 3 laboratory experiments for Fundamentals of Robotics (ME 4773)

6.1 Forward Kinematics

6.2 Inverse Kinematics

6.3 Trajectory generation

7. Servo motors

7.1 Resolution ≤ 5.0° per servo

7.2 Supply voltage ≤ 12V

7.3 Mass ≤ 60 grams per servo

8. Easy to use programming interface

8.1 MATLAB and Simulink to control motors

9. Manipulator Performance

9.1 End-Effector position uncertainty = $w_{x,y,z} \leq 3$ in.

9.2 Maximum joint angular position

- Base servo: $0° \leq \theta_{base} \leq 270°$
- Rest of Servos: $0° \leq \theta \leq 180°$

9.3 Maximum joint angular velocity: $0 \leq \omega \leq 20°/sec$

9.4 Maximum joint angular acceleration: $0 \leq \alpha < 10°/sec^2$

9.5 Cycle time ≤ 10 sec

Appendices C – Test Report

<u>Table of Contents</u>

List of Tables

List of Figures

1.0 Introduction

The purpose of this test plan is to analyze the performance of the Robotic Manipulator designed by Upper Echelon Engineering and ensure its adherence to the physical and functional specifications shown in the Appendix. Testing of the Robotic Manipulator will include evaluating the manipulator and some of its components. By ensuring that all the specifications are met, the Robotic Manipulator will be proven to be portable, reliable, accurate, capable of six degrees of freedom, as well as adhering to the requirements for the three experiments in the Fundamentals of Robotics Course at the University of Texas at San Antonio.

2.0 Scope

Instruments used for the testing of the Robotic Manipulator will be provided by Upper Echelon Engineering. Final Testing will be completed in accordance with this test plan. The testing will be conducted either in the Robotics Laboratory BSE 2.216 or team member Jonathan Sackett's house.

3.0 Features to be Tested

The features of the Robotic Manipulator to be tested are listed in Table 1 below. For reference purposes, the test numbers are described in Table 2.

Table 1. Test Compliance Matrix

Item No.	Feature to be Tested	Specification Reference	Testing Procedure	Pass/Fail
1	Load Geometry	1.1.1	T-1.1	PASS
2	Load Volume	1.1.2	T-1.2	PASS
3	Load Weight	1.1.3	T-1.3	PASS
4	Gripping Area	2.2.2	T-2.1	PASS
5	Gripping Force	2.2.3	T-2.2	PASS
`6	Power Supply Voltage	3.3.2	T-3.1	PASS
7	Power Supply Current	3.3.2	T-3.2	PASS
8	Manipulator Length	4.4.1	T-4.1	PASS
9	Degrees of Freedom	5.5.0	T-5.1	PASS
10	End Effector Navigation	5.5.1	T-5.2	PASS
11	End Effector Position Uncertainty	9.9.1	T-5.3	FAIL
12	Max Joint Angular Positions	9.9.2	T-5.4	PASS
13	Max Joint Angular Velocity	9.9.3	T-5.5	PASS
14	Max Joint Angular Acceleration	9.9.4	T-5.6	PASS
15	Torque Values	-	T-5.7	PASS
16	Cycle Time	9.9.5	T-5.8	PASS
17	Laboratory Experiments	6.1.0	T-6.1	PASS
18	Servo Motor Angle Resolution	7.7.1	T-7.1	PASS
19	Servo Motor Current	-	T-7.2	PASS
20	Servo Motor Voltage	7.7.2	T-7.3	PASS
21	Servo Motor Temperature	-	T-7.4	PASS
22	Servo Motor Mass	7.7.3	T-7.5	PASS
23	GUI	8.8.1	T-8.1	PASS
24	Material	-	T-9.1	PASS
25	Part Dimensions	-	T-9.2	PASS
26	Assemblies	-	T-9.3	PASS

4.0 Test Description

4.1 - T-1.1 Load Geometry Test

The Load Geometry Test is conducted to ensure the load being applied at the gripper is cubic in nature.

4.1.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Load cell

4.1.2 - Procedure

Upon visual inspection, determine if the load is cubic in geometry.

4.1.3 - Data analysis

The data collected from this test will be recorded as a pass/fail as it is part of the specifications listed in the Appendix. Pass is when the load is cubic is geometry. Fail is if the load is not cubic.

4.2 - T-1.2 Load Volume Test

The Load Volume Test is conducted to determine the volume of the load.

4.2.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Venire Calipers

4.2.2 - Procedure

Measure the length, width, and height of the load and multiply them to obtain the volume. Record the data in the appropriate section within Table 2.

4.2.3 - Data Analysis

The data collected from this test will be recorded as a pass/test as it is part of the specifications listed in the Appendix. Pass is when the load is less than or equal to one cubic inch. Fail is if the load is greater than one cubic inch.

4.3 - T-1.3 Load Weight Test

The Load Weight Test is conducted to determine how much load the Robotic Manipulator can handle without failure.

4.3.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Weight set (0-250g set)

2. MATLAB script file for initialization

4.3.2 - Procedure

The manipulator is placed in the initial state by the provided GUI. Weights from the weight set are added to the end effector in increments until the servos either stall, or are incapable of movement, or the manipulator experiences structural failure. Record the last weight before failure is experienced within the appropriate entries within Table 2.

4.3.3 - Data Analysis

The data is tabulated under the respective loads allowing the user to determine at which quantity of load the end-effector would experience failure. "Pass" is tabulated if the load value recorded is greater than or equal to 0.25 lbf. "Fail" is tabulated if the recorded load is less than 0.25 lbf. Even if the test is failed, the specification might still be met.

Table 2. Load Testing Table

Load Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-1.1 Load Geometry	Cubic	Cubic	PASS
T-1.2 Load Volume	≤ 1 in^3	.9 in^3	PASS
T-1.3 Load Weight	≥ 0.25 lbf	.253 lbf	PASS
Date Conducted: 10/10/2016			
Team Member Conducting Test: Omar E. Castro			
Team Member Signature:			

4.4 - T-2.1 Gripping Area Test

The Gripping Area Test is conducted to determine if the specification for the Gripper is met as listed in the Appendix.

4.4.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Venire Caliper

2. Metric Ruler.

4.4.2 - Procedure

The procedure for this test details using Vernier calipers and a metric ruler to measure the gripping area of end-effector. The length, width, and height of the area would be measured to determine if the gripping area matches the defined specifications.

4.4.3 - Data Analysis

The data collected from the Gripping Area Test would be compared to the gripping area listed in the Specifications located in the Appendix.

4.5 - T-2.2 Gripping Force Test

The Gripping Force Test is conducted to measure if the clamp force generated by the Gripper meets the Specifications listed in the Appendix.

4.5.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Digital Hand Dynamometer

4.5.2 - Procedure

The gripping force of the robotic manipulator is measured by placing the Dynamometer between the hands of the gripper. Next, the servo actuates the gripper to "closed" position. The value measured by the Dynamometer is recorded into the appropriate section of Table 3.

4.5.3 - Data Analysis

The data collected for this test is analyzed per the maximum gripping force obtained, which is produced by the robotic manipulator's end-effector. If the force recorded in Table 3 matches the required force specified in the specification listed in the Appendix, then a "Pass" status is tabulated within the appropriate section.

Table 3. Gripper Testing Table

Gripper Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-2.1 Gripping Area	1 in.^{2}	.93 in^2	PASS

T-2.2 Gripping Force	≥ 0.35 lbf	.364 lbf	PASS
Date Conducted: 10/11/2016			
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro			
Team Member Signature:			

4.6 - T-3.1 Power Supply Voltage Test

The Power Supply Voltage Test is conducted to ensure that voltage produced by the power supply is within the specifications.

4.6.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Digital Multi-Meter

4.6.2 - Procedure

Using a digital multi-meter, the voltage produced by the power supply would be tested with load (when the power supply is connected to the robotic manipulator, and is in full function); the other test would be conducted when the power supply is without load (not connected to the robotic manipulator).

4.6.3 - Data Analysis

The data collected from the power supply voltage test would include data from when the power supply is under load and not under load. The specifications for the required voltage are listed in the attached Appendix.

4.7 - T-3.2 Power Supply Current Test

The Power Supply Current Test is conducted to ensure that current produced by the power supply is within the specifications.

4.7.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Digital Multi-Meter

4.7.2 - Procedure

Using a digital multi-meter, the current produced by the power supply would be tested with a load (when the power supply is connected to the robotic manipulator, and is in full function) and without a load (not connected to the robotic manipulator).

4.7.3 - Data Analysis

The data collected from the power supply current test would include data from when the power supply is under load and not under load. The specifications for the required voltage are listed in the attached Appendix.

Table 4. Power Supply Testing Table

Power Supply Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-3.1 Power Supply Voltage	≤ 12 V-DC	12.64 V-DC	PASS
T-3.2 Power Supply Current	10 A	9.96 A	PASS
Date Conducted: 10/12/2016			
Team Member Conducting Test: Omar E. Castro			
Team Member Signature:			

4.8 - T-4.1 Manipulator Length Test

The Manipulator Length Test is conducted to see that the specifications are met for the required total length of the Robotic Manipulator.

4.8.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Venire Calipers
2. Metric Ruler

4.8.2 - Procedure

Using Venire Calipers and a metric ruler, measure and record the total length of the manipulator after assembly. The measurement should include the length of each link, and the total length of the manipulator's "arm".

4.8.3 - Data Analysis

The data collected from the Manipulator Length Test would be directly compared to the theoretical values that are listed in the Specifications that are located in the Appendix.

Table 5. Manipulator Length Test Table

Manipulator Length Test			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-4.1 Manipulator Length	1.5 ft. $\leq L \leq$ 2 ft.	1.7 ft.	PASS
Date Conducted: 11/10/2016			
Team Member Conducting Test: Omar E. Castro			
Team Member Signature:			

4.9 - T-5.1 Degrees of Freedom Test

The Degrees of Freedom Test is conducted in order to determine if the Robotic Manipulator meets the specification listed in the Appendix.

4.9.1 - Equipment

The following is a list of required equipment to conduct the test:

1. MATLAB
2. Simulink

4.9.2 - Procedure

Using the MATLAB library, run a code via Simulink and MATLAB to ensure the specified degrees of freedom are met.

4.9.3 - Data Analysis

There is no data collected as this is a pass or fail test. Pass means the specifications were met (6-DOF), and fail means that the specification was not met.

Table 6. Degree of Freedom Test Table

Degree of Freedom Test			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.1 Degrees of Freedom	6	6	PASS
Date Conducted: 10/19/2016			
Team Member Conducting Test: Omar E. Castro			
Team Member Signature:			

4.10 - T-5.2 End Effector Navigation Test

The End Effector Navigation Test is conducted to determine how efficient the Robotic Manipulator's End-Effector operates during load (running an experiment).

4.10.1 - Equipment

The following is a list of required equipment to conduct the test:

1. MATLAB
2. Simulink

4.10.2 - Procedure

The end-effector navigation test will be conducted using the codes inputted into MATLAB during the experiments and see how the end-effector moves efficiently to the inputted code and if the results are in conjunction with the specified requirements that need to be met.

4.10.3 - Data Analysis

There would be no data analysis for this test as it is a pass/fail test.

Table 7. End Effector Navigation Test Table

End Effector Navigation Test			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.2 End Effector Navigation	1/2 Cartesian Space	½ Cartesian Space	PASS
Date Conducted: 10/19/2016			
Team Member Conducting Test: Sergio A. Molina			
Team Member Signature:			

4.11 - T-5.3 End Effector Position Uncertainty Test

The end effector position uncertainty test is concerned with validating that the robotic manipulator's end effector is within the specified level of uncertainty for position.

4.11.1 - Equipment

The following is a list of the equipment required to successfully execute this test:

1. Metric or Imperial style of unit Ruler
2. Marker or writing utensil
3. Weighted Set 10 to 1000g for loading condition testing
4. Graphical User Interface

4.11.2 - Procedure

To verify that the end effector position is within the specified tolerance range its position must be measured under no-load and load conditions. A ruler and marker or writing utensil is needed to determine the x, y, and z position of the end effector relative to the origin of joint 1. The steps for testing the end effector position will be described respectively.

Steps for No-load End Effector Position Uncertainty Test:

1. Ensure robotic manipulator is powered on
2. Verify the manipulator is in its initial state by selecting the "Reset" button on GUI
3. Open GUI via MATLAB

4. Using the GUI maneuver joint 1 to 45º
5. Take ruler and measure the x, y, and z values respectively
6. Tabulate results in appropriate rows and columns of Table 7
7. Calculate the error

a. Subtract the desired value from the obtained value

b. Take the absolute value of the obtained result

c. Populate the relevant sections of Table 7

8. Repeat steps 4 - 7 for joints 2 - 7

Steps for Load End Effector Position Uncertainty Test:

1. Ensure robotic manipulator is powered on
2. Verify the manipulator is in its initial state by selecting the "Reset" button on GUI
3. Open GUI via MATLAB
4. Begin with the lowest specified value of the load weight provided in table 7
5. Securely apply the required weight(s) to the end effector to match the value from table 7
6. Using the GUI maneuver joint 1 to 45º
7. Take ruler and measure the x, y, and z values respectively
8. Tabulate results in appropriate rows and columns of table 7
9. Repeat steps 5 - 8 for joints 2 - 6

4.11.3 - Data Analysis

By comparing the true percent error value of the end effector obtained for each joint to the tolerance level the test will either fail or pass. If the error is greater than 3 in, the test is a fail. If the error is less than or equal to 3 in, then it is a pass.

4.12 - T-5.4 Max Joint Angular Positions Test

The Max Joint Angular Position Test is intended is meant to measure the range of angles the motors could output and to ensure the corresponding specification is met.

4.12.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Graphical User Interface
2. Protractor

3. Writing utensil
4. Ten sheets of 11 in. by 17 in. white paper
5. Clear Tape

4.12.2 - Procedure

Each joint will be tested one at a time. To measure the range of angles, a sheet of white paper will be taped to the manipulator such that the joint being evaluated is roughly at the center of that sheet. First, the maximum range of joint 1 will be measured. The start position will be traced onto the paper and annotated then the manipulator will be actuated via the GUI to the max angle and the final position will be traced and annotated. Next, the person(s) conducting the test will calculate the difference in angles and make a determination of the outcome of the experiment. This process will be repeated for joints 2 – 7.

The following is a list of the steps required for completion of the test on joint 1:

1. Open GUI
2. Verify the manipulator is in its initial state by selecting the "Reset" button on GUI
3. Tape a 11 in. by 17 in. sheet of paper underneath the base of the manipulator
4. Mark a line (parallel) directly underneath the manipulator on the paper
5. Identify the line by writing "Min" within the proximity of the line
6. Increment the angle values for joint 1 from 0 - 270º
7. Repeat step 4
8. Identify the line by writing "Max" within the proximity of the line
9. Measure the change in angle using the protractor

The following is a list of the steps required for completion of the test on joint 2 - 6:

1. Open GUI
2. Verify the manipulator is in its initial state by selecting the "Reset" button on GUI
3. Tape a 11 in. by 17 in. sheet of paper adjacent to the link nearest the manipulator
4. Mark a line parallel to the manipulator on the sheet of paper
5. Identify the line by writing "Min" within the proximity of the line
6. Increment the angle values for joint 1 from 0 - 180º
7. Repeat step 4
8. Identify the line by writing "Max" within the proximity of the line
9. Measure the change in angle using the protractor

4.12.3 - Data Analysis

Comparing the measured values to those provided will determine a "Pass" for the maximum angular position of the joints if the error is less than 20%. For each test, the technician will record with a "Pass" for confirmation that the joint passed or "Fail" if the joint failed to reach the criteria.

4.13 - T-5.5 Max Joint Angular Velocity

The Max Joint Angular Velocity Test is intended is to measure the range of angular velocities that the motors could output and to ensure that the corresponding specification for joint angular velocity for each joint is successfully met.

4.13.1 - Equipment

The following is a list of required software and physical hardware necessary for executing the test:

1. MATLAB Script File for Angular Velocity Sweep

4.13.2 - Procedure

To test for the maximum angular velocity, the robotic manipulator joints would have to be subjected to a sweep in angular velocity one at a time starting from the initial state. A fifth-order polynomial, time-dependent function that accounts for the maximum joint angular velocity within its unique solution will be used to control the motors to the desired angular velocity. The script file is created in MATLAB and will automatically execute the sweep upon execution. In addition, the file will be outputting the measured joint velocities from each individual motor into the "Command Window" of MATLAB.

4.13.3 - Data Analysis

The data will be displayed in the "Command Window" where the technician will be able to tabulate the results into the following table 14. The angular velocity values will be tabulated for each joint and will either "Pass" or "Fail" based on whether the values of the angular velocity (ω) are within the bounds, $0 \leq \omega \leq 20°/\text{sec}$. If a joint doesn't reach at least 50% of the maximum bound of 20°/sec, then the technician will specify "Fail" on the table 14 and if it does surpass that measure then the technician will specify "Pass".

4.14 - T-5.6 Max Joint Angular Acceleration Test

The Max Joint Angular Acceleration Test is intended is to measure the range of angular acceleration that the motors could output and to ensure that the corresponding specification for joint angular velocity for each joint is successfully met.

4.14.1 - Equipment

The following is a list of required software and physical hardware necessary for executing the test:

1. MATLAB Script File for Angular Acceleration Sweep

4.14.2 - Procedure

To test for the maximum angular acceleration, the robotic manipulator joints would have to be subjected to a sweep in angular acceleration one at a time starting from the initial state. A polynomial, time-dependent function, which accounts for the maximum joint angular acceleration within its unique solution, will be used to control the motors to the desired angular acceleration. The script file is created in MATLAB and will automatically execute the sweep upon execution. In addition, the file will be outputting the measured joint acceleration from each individual motor into the "Command Window" of MATLAB.

4.14.3 - Data Analysis

The data will be displayed in the "Command Window" where the technician will be able to tabulate the results into the following table 14. The angular velocity values will be tabulated for each joint and will either "Pass" or "Fail" based on whether the values of the angular acceleration (α) are within the bounds, $0 \leq \alpha < 10°/sec^2$ .If a joint doesn't reach at least 50% of the maximum bound of 20°/sec then the technician will specify "Fail" on the table 14 and if it does surpass that measure then the technician will specify "Pass".

4.15 - T-5.7 Torque Values Test

The purpose of this test is to ensure the output torque of the servo motors do not exceed the motors rated stall torque value provided by the Dynamixel AX-12A datasheet.

4.15.1 - Equipment

The necessary equipment and software for conducting the test are listed below:

1. Fully Assembled Robotic Manipulator
2. MATLAB Script file for Measuring Torque Values

4.15.2 - Procedure

The following is a step-by-step list necessary to execute this experiment

1. Ensure robotic manipulator is powered on
2. Open MATLAB Script file for Measuring Torque Values
3. Begin with the lowest specified value of the weight of load from table 14
4. Securely apply the required weight(s) to the end effector to match the value from table 14
5. "Run" the MATLAB Script file for Measuring Torque Values

6. Tabulate the results obtained in the “Command Window” pertaining to the relevant servo
7. Increment the load and repeat process until the 0.25 lbf is tested
8. Repeat steps 3 - 6 for joints 2 - 6

4.15.3 - Data Analysis

For the manipulator to be operating safely during actuation the motors must not exceed ⅘ of the rated stall torque for the Dynamixel AX-12A. Furthermore, if any of the data values obtained from the various trails exceed ⅘ of the rated stall torque for the Dynamixel AX-12A then a “Fail” will be reported, otherwise “Pass” will be reported.

4.16 - T-5.8 Cycle Time Test

The Cycle Time Test will test the manipulator performance for whether it meets the cycle time specification. The following four conditions will be imposed on the manipulator to evaluate all the working conditions that it will be exposed to:

1. Static Condition with No Load
2. Static Condition with Loads at End Effector
3. Dynamic Condition with No Load
4. Dynamic Condition with Loads at End Effector

4.16.1 - Equipment

1. MATLAB Script file for Executing Trajectory Generation in Desired Cycle Time
2. Fully Assembled Robotic Manipulator
3. Stopwatch

4.16.2 - Procedure

The Dynamixel AX-12A is not capable of measuring Torque values directly so the method of measurement will be indirect. Two methods will be discussed to cross-reference the two experimental outcomes to best evaluate the criteria. The following conditions will be imposed on the full assembly of the robotic manipulator:

1. Dynamic Condition with No Load
2. Dynamic Condition with Loads at End Effector

The list of steps for the “Dynamic Condition with No Load”:

1. Ensure robotic manipulator is powered on
2. Load the “Trajectory Generation in Desired Cycle Time” script file into MATLAB

3. Visually inspect to ensure the manipulator is in initial state configuration
4. Run the script file
5. Track the time it takes for the manipulator to complete the trajectory
6. Tabulate data into table 15
7. Calculate the difference in time values from the measured and theoretical
8. Tabulate results
9. Visually inspect the trajectory generated to ensure it is following the specified path

The list of steps for the "Dynamic Condition with Loads at End Effector":

1. Ensure robotic manipulator is powered on
2. Load the "Trajectory Generation in Desired Cycle Time" script file into MATLAB
3. Visually inspect to ensure the manipulator is in initial state configuration
4. Mount the necessary weight at the end effector according to table 14
5. Ensure the weight is properly secured
6. Run the script file
7. Track the time it takes for the manipulator to complete the trajectory
8. Tabulate data into table 14
9. Calculate the difference in time values from the measure to the theoretical
10. Tabulate results
11. Visually inspect the trajectory to ensure it is following the specified path
12. Tabulate visually obtained results of the trajectory path

4.16.3 - Data Analysis

To ensure the manipulator performs to the specified cycle time the maximum amount of error resulting from the Cycle Time Test is within ±1% of 10 seconds. If the error value resulting from the data is not within the specified range of acceptable deviation then the technician is to specify "Fail" in the relevant columns and "Pass" if it does fall within the range of acceptable values.

Table 8. Static Loading Test Table for Servo 1

Static Loading Tests
Servo 1

	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value	< 1.5 Nm	.33 Nm	PASS	< 1.5 Nm	.354 Nm	PASS	< 1.5 Nm	.41 Nm	PASS
T-7.2 Servo Motor Current	< 1.5 A	.79 A	PASS	< 1.5 A	1 A	PASS	< 1.5 A	1.15 A	PASS
T-7.3 Servo Motor Voltage	≤ 12 V	11 V	PASS	≤ 12 V	10.7 V	PASS	≤ 12 V	10.6 V	PASS
T-7.5 Servo Motor Temperature	< 70 °C	39 °C	PASS	< 70 °C	39.4 °C	PASS	< 70 °C	40.2 °C	PASS
Date Conducted: 11/12/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 9. Static Loading Test Table for Servo 2

Static Loading Tests									
Servo 2									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**

T-5.7 Torque Value	< 1.5 Nm	.705 Nm	PASS	< 1.5 Nm	.893 Nm	PASS	< 1.5 Nm	1.5 Nm	FAIL
T-7.2 Servo Motor Current	< 1.5 A	.56 A	PASS	< 1.5 A	.76 A	PASS	< 1.5 A	1.36 A	PASS
T-7.3 Servo Motor Voltage	≤ 12 V	12 V	PASS	≤ 12 V	10.9 V	PASS	≤ 12 V	10.4 V	PASS
T-7.5 Servo Motor Temperature	< 70 °C	57 °C	PASS	< 70 °C	60 °C	PASS	< 70 °C	61 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 10. Static Loading Test Table for Servo 3

Static Loading Tests									
Servo 3									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value	< 1.5 Nm	.32 Nm	PASS	< 1.5 Nm	.42 Nm	PASS	< 1.5 Nm	.47 Nm	PASS
T-7.2 Servo	< 1.5 A	.33 A	PASS	< 1.5 A	.39 A	PASS	< 1.5 A	.44 A	PASS

Motor Current									
T-7.3 Servo Motor Voltage	≤ 12 V	11.8 V	PASS	≤ 12 V	11.7 V	PASS	≤ 12 V	11.7 V	PASS
T-7.5 Servo Motor Temperature	< 70 °C	50 °C	PASS	< 70 °C	53 C	PASS	< 70 °C	56 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 11. Static Loading Test Table for Servo 4

Static Loading Tests									
Servo 4									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value	< 1.5 Nm	.19 Nm	PASS	< 1.5 Nm	.19 Nm	PASS	< 1.5 Nm	.28 Nm	PASS
T-7.2 Servo Motor Current	< 1.5 A	.38 A	PASS	< 1.5 A	.41 A	PASS	< 1.5 A	.44 A	PASS
T-7.3 Servo	≤ 12 V	11.8 V	PASS	≤ 12 V	11.7 V	PASS	≤ 12 V	11.5 V	PASS

Motor Voltage									
T-7.5 Servo Motor Temperature	< 70 °C	49 °C	PASS	< 70 °C	51 °C	PASS	< 70 °C	52 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 12. Static Loading Test Table for Servo 5

Static Loading Tests									
Servo 5									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value	< 1.5 Nm	0 Nm	PASS	< 1.5 Nm	0 Nm	PASS	< 1.5 Nm	0 Nm	PASS
T-7.2 Servo Motor Current	< 1.5 A	.10 A	PASS	< 1.5 A	.09 A	PASS	< 1.5 A	.10 A	PASS
T-7.3 Servo Motor Voltage	≤ 12 V	12 V	PASS	≤ 12 V	12 V	PASS	≤ 12 V	12 V	PASS
T-7.5 Servo Motor	< 70 °C	35 °C	PASS	< 70 °C	35 °C	PASS	< 70 °C	35 °C	PASS

Temperature									
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 13. Static Loading Test Table for Servo 6

Static Loading Tests									
Servo 6									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.7 Torque Value	< 1.5 Nm	0 Nm	PASS	< 1.5 Nm	0 Nm	PASS	< 1.5 Nm	0 Nm	PASS
T-7.2 Servo Motor Current	< 1.5 A	.06 A	PASS	< 1.5 A	.07 A	PASS	< 1.5 A	.07 A	PASS
T-7.3 Servo Motor Voltage	≤ 12 V	12 V	PASS	≤ 12 V	12 V	PASS	≤ 12 V	12 V	PASS
T-7.5 Servo Motor Temperature	< 70 °C	34 °C	PASS	< 70 °C	34 °C	PASS	< 70 °C	32 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									

Team Member Signature:

Table 14. Dynamic Loading Test Table for Servo 1

Dynamic Loading Tests									
Servo 1									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL
T-5.4 Max Joint Angular Position	≤ 270°	101°	PASS	≤ 270°	86°	PASS	≤ 270°	85.2°	PASS
T-5.5 Max Joint Angular Velocity	≤ 20°/s	10.44°/s	PASS	≤ 20°/s	10.4°/s	PASS	≤ 20°/s	10.32°/s	PASS
T-5.6 Max Joint Angular Acceleration	≤ 10°/s^2	1.04°/s^2	PASS	≤ 10°/s^2	1.04°/s^2	PASS	≤ 10°/s^2	1.03°/s^2	PASS
T-5.7 Torque Value	< 1.5 Nm	.33 Nm	PASS	< 1.5 Nm	.377 Nm	PASS	< 1.5 Nm	.45 Nm	PASS
T-5.8 Cycle Time	≤ 10 s	10 s	PASS	≤ 10 s	10 s	PASS	≤ 10 s	10 s	PASS

T-7.2 Servo Motor Current	< 1.5 A	.79 A	PASS	< 1.5 A	1.03 A	PASS	< 1.5 A	1.22 A	PASS
T-7.3 Servo Motor Voltage	≤ 12 V	11 V	PASS	≤ 12 V	10.7 V	PASS	≤ 12 V	10.6 V	PASS
T-7.5 Servo Motor Temperature	< 70 °C	39 °C	PASS	< 70 °C	39.4 °C	PASS	< 70 °C	45.2 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 15. Dynamic Loading Test Table for Servo 2

Dynamic Loading Tests									
Servo 2									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL
T-5.4 Max Joint	≤ 180°	N/A	FAIL	≤ 180°	N/A	FAIL	≤ 180°	N/A	FAIL

Angular Position									
T-5.5 Max Joint Angular Velocity	≤ 20°/s	N/A	FAIL	≤ 20°/s	N/A	FAIL	≤ 20°/s	N/A	FAIL
T-5.6 Max Joint Angular Acceleration	≤ 10°/s^2	N/A	FAIL	≤ 10°/s^2	N/A	FAIL	≤ 10°/s^2	N/A	FAIL
T-5.7 Torque Value	< 1.5 Nm	>1.5 Nm	FAIL	< 1.5 Nm	N/A	FAIL	< 1.5 Nm	N/A	FAIL
T-5.8 Cycle Time	≤ 10 s	N/A	FAIL	≤ 10 s	N/A	FAIL	≤ 10 s	N/A	FAIL
T-7.2 Servo Motor Current	< 1.5 A	1.5 A	FAIL	< 1.5 A	N/A	FAIL	< 1.5 A	N/A	FAIL
T-7.3 Servo Motor Voltage	≤ 12 V	11.7 V	PASS	≤ 12 V	N/A	FAIL	≤ 12 V	N/A	FAIL
T-7.5 Servo Motor Temperature	< 70 °C	49 °C	PASS	< 70 °C	N/A	FAIL	< 70 °C	N/A	FAIL
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 16. Dynamic Loading Test Table for Servo 3

Dynamic Loading Tests									
Servo 3									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL
T-5.4 Max Joint Angular Position	≤ 180°	88.9°	PASS	≤ 180°	N/A	FAIL	≤ 180°	N/A	FAIL
T-5.5 Max Joint Angular Velocity	≤ 20°/s	2.3°/s	PASS	≤ 20°/s	N/A	FAIL	≤ 20°/s	N/A	FAIL
T-5.6 Max Joint Angular Acceleration	≤ 10°/s^2	.06°/s^2	PASS	≤ 10°/s^2	N/A	FAIL	≤ 10°/s^2	N/A	FAIL
T-5.7 Torque Value	< 1.5 Nm	1.47 Nm	PASS	< 1.5 Nm	>1.5 Nm	FAIL	< 1.5 Nm	>1.5 Nm	FAIL
T-5.8 Cycle Time	≤ 10 s	10 s	PASS	≤ 10 s	N/A	FAIL	≤ 10 s	N/A	FAIL
T-7.2 Servo Motor Current	< 1.5 A	1 A	PASS	< 1.5 A	1.22 A	PASS	< 1.5 A	1.40 A	PASS
T-7.3 Servo Motor Voltage	≤ 12 V	11.8 V	PASS	≤ 12 V	10.5 V	PASS	≤ 12 V	10.5 V	PASS

T-7.5 Servo Motor Temperature	< 70 °C	51 °C	PASS	< 70 °C	57 °C	PASS	< 70 °C	58 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 17. Dynamic Loading Test Table for Servo 4

Dynamic Loading Tests									
Servo 4									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL
T-5.4 Max Joint Angular Position	≤ 180°	86.02°	PASS	≤ 180°	86°	PASS	≤ 180°	89.8°	PASS
T-5.5 Max Joint Angular Velocity	≤ 20°/s	19.8°/s	PASS	≤ 20°/s	20°/s	PASS	≤ 20°/s	18.8°/s	PASS
T-5.6 Max Joint Angular	≤ 10°/s^2	1.98°/s^2	PASS	≤ 10°/s^2	2°/s^2	PASS	≤ 10°/s^2	1.88°/s^2	PASS

Acceleration									
T-5.7 Torque Value	< 1.5 Nm	.71 Nm	PASS	< 1.5 Nm	1.08 Nm	PASS	< 1.5 Nm	1.16 Nm	PASS
T-5.8 Cycle Time	≤ 10 s	10 s	PASS	≤ 10 s	10 s	PASS	≤ 10 s	10 s	PASS
T-7.2 Servo Motor Current	< 1.5 A	.42 A	PASS	< 1.5 A	.52 A	PASS	< 1.5 A	.86 A	PASS
T-7.3 Servo Motor Voltage	≤ 12 V	11.8 V	PASS	≤ 12 V	11.6 V	PASS	≤ 12 V	12 V	PASS
T-7.5 Servo Motor Temperature	< 70 °C	45 °C	PASS	< 70 °C	51 °C	PASS	< 70 °C	59 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

Table 18. Dynamic Loading Test Table for Servo 5

Dynamic Loading Tests									
Servo 5									
	0 lbf			.125 lbf			.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**

T-5.3 End Effector Position Uncertainty	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL
T-5.4 Max Joint Angular Position	≤ 180°	84.3°	PASS	≤ 180°	84.3°	PASS	≤ 180°	84.7°	PASS
T-5.5 Max Joint Angular Velocity	≤ 20°/s	2.64°/s	PASS	≤ 20°/s	2.64°/s	PASS	≤ 20°/s	2.64°/s	PASS
T-5.6 Max Joint Angular Acceleration	≤ 10°/s^2	.264°/s^2	PASS	≤ 10°/s^2	.264°/s^2	PASS	≤ 10°/s^2	.264°/s^2	PASS
T-5.7 Torque Value	< 1.5 Nm	.223 Nm	PASS	< 1.5 Nm	.25 Nm	PASS	< 1.5 Nm	.27 Nm	PASS
T-5.8 Cycle Time	≤ 10 s	10 s	PASS	≤ 10 s	10 s	PASS	≤ 10 s	10 s	PASS
T-7.2 Servo Motor Current	< 1.5 A	.21 A	PASS	< 1.5 A	.21 A	PASS	< 1.5 A	.22 A	PASS
T-7.3 Servo Motor Voltage	≤ 12 V	12 V	PASS	≤ 12 V	12 V	PASS	≤ 12 V	12 V	PASS
T-7.5 Servo Motor Temperature	< 70 °C	33 °C	PASS	< 70 °C	34 °C	PASS	< 70 °C	37 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									

Team Member Signature:

Table 19. Dynamic Loading Test Table for Servo 6

Dynamic Loading Tests									
Servo 6									
	0 lbf			0.125 lbf			0.25 lbf		
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**	**Required Value**	**Measured Value**	**Pass/Fail**
T-5.3 End Effector Position Uncertainty	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL	≤ 3 in	N/A	FAIL
T-5.4 Max Joint Angular Position	≤ 180°	88°	PASS	≤ 180°	87°	PASS	≤ 180°	87.2°	PASS
T-5.5 Max Joint Angular Velocity	≤ 20°/s	18.6°/s	PASS	≤ 20°/s	18.6°/s	PASS	≤ 20°/s	20°/s	PASS
T-5.6 Max Joint Angular Acceleration	≤ 10°/s^2	1.86°/s^2	PASS	≤ 10°/s^2	1.86°/s^2	PASS	≤ 10°/s^2	2°/s^2	PASS
T-5.7 Torque Value	< 1.5 Nm	.20 Nm	PASS	< 1.5 Nm	.20 Nm	PASS	< 1.5 Nm	.20 Nm	PASS
T-5.8 Cycle Time	≤ 10 s	10 s	PASS	≤ 10 s	10 s	PASS	≤ 10 s	10 s	PASS
T-7.2 Servo	< 1.5 A	.13 A		< 1.5 A	.11 A	PASS	< 1.5 A	.11 A	

Motor Current			PASS						PASS
T-7.3 Servo Motor Voltage	≤ 12 V	12 V	PASS	≤ 12 V	12 V	PASS	≤ 12 V	12 V	PASS
T-7.5 Servo Motor Temperature	< 70 °C	37 °C	PASS	< 70 °C	37 °C	PASS	< 70 °C	37 °C	PASS
Date Conducted: 11/14/2016									
Team Member Conducting Test: Sergio A. Molina & Omar E. Castro									
Team Member Signature:									

4.17 - T-6.1 Laboratory Experiment Tests

This test ensures that the manipulator successfully and easily interfaces with the three laboratory experimental manuals included the manipulator package.

4.17.1 - Equipment

The following software and hardware is necessary to execute the Laboratory Experiment Test:

1. Full assembly of robotic manipulator
2. Three complete laboratory manuals
3. Complete Simulink library for robotic manipulator

4.17.2 - Procedure

The Laboratory Experiment Test requires the technician to complete the three experimental laboratories sequentially, one after the other. In addition, the technician must follow the exact instructions provided by the laboratory manuals.

4.17.3 - Data Analysis

The data obtained from the Laboratory Experiment Test will either be a "Pass" or "Fail" status depending on the outcome of each experiment. To add, if the technician was able to successfully complete every instruction within the laboratory manual with no error or evident problem, a "Pass" will be tabulated in the respective section of the test outcome table 20 and "Fail" otherwise.

Table 20. Lab Experiments Testing Table

Lab Experiments Testing	
Featured Test	Pass/Fail
T-6.1 Lab Experiments	
Experiment 1 - Forward Kinematics	PASS
Experiment 2 - Inverse Kinematics	PASS
Experiment 3 - Trajectory Planning	PASS
Date Conducted: 11/13/2016	
Team Member Conducting Test: Sergio. A Molina	
Team Member Signature:	

4.19 - T-7.2 Servo Motor Current Test

The Servo Motor Current Test is a test to ensure that the current being supplied to each motor during actuation is not exceeding the maximum current as stated in the Dynamixel AX-12A datasheet.

4.18.1 - Equipment

1. Fully assembled robotic manipulator
2. MATLAB Script file for Executing Trajectory Generation in Desired Cycle Time
3. Digital Multi-Meter (capable of measuring 120V, 10A)

4.18.2 - Procedure

The following is a list of steps required to execute the test.

1. Ensure that the robotic manipulator is turned "On"
2. Load the "robot_initialize" MATLAB script file
3. Initialize the robotic manipulator to the "initial state" by executing the script file

4. Load the “robot_trajectory_gen” script file
5. Connect, in series, the positive terminal of the DMM to the “Voltage” pin of the robotic manipulator for the first joint
6. Connect, in series, the negative terminal of the DMM to the “Ground” pin of the robotic manipulator for the first joint
7. Visually ensure that all wiring is properly connected
8. “Run” the “robot_trajectory_gen” script file
9. Tabulate the maximum current value throughout the execution time of the script file
10. Tabulate results in table 14
11. Repeat steps 1 - 11 for joints 2 - 6

4.18.3 - Data Analysis

The maximum current values for each joint will be analyzed and compared to the Dynamixel AX-12A maximum current rating. If any of the joint current values exceed 900mA then the “Fail” criteria is established in the appropriate entry box. If any of the joint current values are less than or equal to 900mA then the “Pass” criteria is established in the appropriate entry box.

4.19 - T-7.2 Servo Motor Voltage Test

The Servo Motor Voltage Test is a test to ensure that the voltage being supplied to each motor during actuation is not exceeding the maximum voltage as stated in the Dynamixel AX-12A datasheet.

4.19.1 - Equipment

1. Fully assembled robotic manipulator
2. MATLAB Script file for Executing Trajectory Generation in Desired Cycle Time
3. Digital Multi-Meter (capable of measuring 120V, 10A)

4.19.2 - Procedure

The following is a list of steps required to execute the test.

1. Ensure that the robotic manipulator is turned “On”
2. Load the “robot_initialize” MATLAB script file
3. Initialize the robotic manipulator to the “initial state” by executing the script file
4. Load the “robot_trajectory_gen” script file
5. Connect, in parallel, the positive terminal of the DMM to the “Voltage” pin of the robotic manipulator for the first joint

6. Connect, in parallel, the negative terminal of the DMM to the "Ground" pin of the robotic manipulator for the first joint
7. Visually ensure that all wiring is properly connected
8. "Run" the "robot_trajectory_gen" script file
9. Tabulate the maximum voltage value throughout the execution time of the script file
10. Tabulate results in table 14
11. Repeat steps 1 - 11 for joints 2 - 6

4.19.3 - Data Analysis

The maximum voltage values for each joint will be analyzed and compared to the Dynamixel AX-12A maximum voltage rating. If any of the joint voltage values exceed 12V then the "Fail" criteria is established in the appropriate entry box. If any of the joint current values are less than or equal to 12V then the "Pass" criteria is established in the appropriate entry box.

4.20 - T-7.3 Servo Motor Mass Test

The Servo Motor Mass Test is a test to determine whether the mass of the servo's adhere to the Dynamixel AX-12A datasheet.

4.20.1 - Equipment

The following equipment is necessary in order to execute the Servo Motor Mass Test:

1. 7 of the Dynamixel AX-12A Servos
2. Digital Scale

4.20.2 - Procedure

The following list is a step-by-step outline of the Servo Motor Mass Test:

1. Gather all 7 Dynamixel AX-12A servos that belong to the robotic manipulator
2. Please ensure that no cell-phones or powered electronic devices are within a 1 foot radius of the digital scale as this may interfere with the results
3. Turn the digital scale to "On" mode
4. Set the units of measurement on the digital scale as specified in the column entry for "Mass of Servo 1"
5. Calibrate the scale to a value of zero reading
6. Place servo 1, as identified by the back of the servo housing under "ID", on top of the platform
7. Record the value of the mass in appropriate entry of the table 22

8. Repeat steps 2-7 for servo motors 2-7

4.20.3 - Data Analysis

The experimental values of the servomotors as obtained from the "Servo Motor Mass Test" must be no greater than the value of 60g. If any of the motor mass' experimental values exceed 60g then "Fail" must be tabulated in the appropriate section of that motor, otherwise the "Pass" status is tabulated.

4.21 - T-7.5 Servo Motor Temperature Test

The purpose of "Servo Motor Temperature Test" is to determine whether the Temperature values of the servomotors during operation exceed the maximum allowable temperature as specified in the Dynamixel AX-12A datasheet.

4.18.1 - Equipment

The following is a list necessary to execute the "Servo Motor Temperature Test":

1. Full Assembly of robotic manipulator
2. MATLAB Script file for Temperature measurements

4.18.2 - Procedure

The following list is a step-by-step outline of the Servo Motor Temperature Test:

1. Ensure robotic manipulator is properly assembled
2. Ensure that the robotic manipulator is powered "On"
3. Load the "servo_initialization" script file in MATLAB
4. "Run" the "servo_initialization" script file in MATLAB
5. Load the "temp_read_servo" script file in MATLAB
6. Tabulate the temperature values for each of the seven joints provided in the "Command Window" onto table 14
7. Repeat steps 3 and 4
8. Turn off the robotic manipulator

4.18.3 - Data Analysis

The servomotors for the robotic manipulator must be within a strict temperature range of $-5℃ \leq T \leq 70℃$ as specified by the Dynamixel AX-12A datasheet during operation to not fail. If any of the measured values from the servomotors after the execution of the "Servo Motor Temperature Test" falls outside of the temperature range previously mentioned then the "Fail" status is to be reported, otherwise "Pass" is reported.

Table 22. Servo Motor Testing Table

Servo Motor Testing			
Featured Test	**Required Value**	**Measured Value**	**Pass/Fail**
T-7.1 Servo Motor Angle Resolution	.29 Degrees	.29 Degrees	PASS
T-7.3 Servo Motor Mass	55g	54g	PASS
Date Conducted: 10/14/2016			
Team Member Conducting Test: Paul Alabi			
Team Member Signature:			

4.21 - T-8.1 GUI Test

The purpose of the GUI Test is to ensure that the robotic manipulator properly interfaces with the graphical user interface through the following two modes of actuation:

1. Forward kinematics Section

a. Actuation of each joint using the "slider" feature

2. Inverse Kinematics Section

. Actuation via end-effector position specification (x,y,z)

4.21.1 - Equipment

The following is a list necessary to execute the "GUI Test":

1. Full assembly of robotic manipulator
2. Graphical User Interface

4.21.2 - Procedure

The following list is a step-by-step outline of the "GUI Test":

1. Ensure robotic manipulator is properly assembled
2. Ensure that the robotic manipulator is powered "On"
3. Load the GUI via MATLAB
4. Initialize "COMPort" and "Baud Rate" by inputting their respective value

a. Note: Follow the Operations Manual to determine "COMPort" number from computer

b. Select a "Baud Rate" value of 1

5. Vary joint angle 1 from 0 to 30º within the "Forward Kinematics" panel
6. Input a value of x = -6, y = 6, and z =8 within the "Inverse Kinematics" panel
7. Select "Enter"
8. Repeat step 5 for joint angles 2 - 7

4.21.3 - Data Analysis

The criteria established for this test is based on visual inspection. If the manipulator does not respond (actuate) to either of the joint angle or position inputs then the "Fail" status is reported, otherwise the "Pass" status is reported.

Table 21. GUI Test Table for Servo 1

GUI Test						
Servo 1						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI	90°	90°	PASS	90°	90°	PASS
Date Conducted: 11/14/2016						
Team Member Conducting Test: Sergio A. Molina						
Team Member Signature:						

Table 22. GUI Test Table for Servo 2

GUI Test						
Servo 2						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI	90°	90°	PASS	90°	90°	PASS

Date Conducted: 11/14/2016
Team Member Conducting Test: Sergio A. Molina
Team Member Signature:

Table 23. GUI Test Table for Servo 3

GUI Test						
Servo 3						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI	90°	90°	PASS	90°	90°	PASS
Date Conducted: 11/14/2016						
Team Member Conducting Test: Sergio A. Molina						
Team Member Signature:						

Table 24. GUI Test Table for Servo 4

GUI Test						
Servo 4						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI	90°	90°	PASS	90°	90°	PASS
Date Conducted: 11/14/2016						
Team Member Conducting Test: Sergio A. Molina						

Team Member Signature:

Table 25. GUI Test Table for Servo 5

GUI Test						
Servo 5						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI	90°	90°	PASS	90°	90°	PASS
Date Conducted: 11/14/2016						
Team Member Conducting Test: Sergio A. Molina						
Team Member Signature:						

Table 26. GUI Test Table for Servo 6

GUI Test						
Servo 6						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI	90°	90°	PASS	90°	90°	PASS
Date Conducted: 11/14/2016						
Team Member Conducting Test: Sergio A. Molina						
Team Member Signature:						

Table 27. GUI Test Table for Servo 7

GUI Test						
Servo 7						
Featured Test	**Angle Tolerance Required**	**Position Tolerance Required**	**Pass/Fail**	**Angle Tolerance Measured**	**Position Tolerance Measured**	**Pass/Fail**
T-8.1 GUI	90°	90°	PASS	90°	90°	PASS
Date Conducted: 11/14/2016						
Team Member Conducting Test: Sergio A. Molina						
Team Member Signature:						

4.24 - T-9.1 Material Test

The material test is conducted to verify that the specified material is being used in the fabrication process.

4.22.1 - Equipment

In order to complete the material test a technical data sheet for PLA generated by Ultimaker will be used to confirm the material meets specification.

4.22.2 - Procedure

The team member who is to conduct the test should refer to table 28 while simultaneously referring to the technical data sheet spoken of in section 4.22.1. The team member is required to visually confirm with said technical data sheet that the material on hand that was purchased is PLA.

4.22.3 - Data Analysis

No data analysis will be generated from this test.

Table 28. Material Test Table

Material Test			
Featured Test	**Required**	**Confirmation**	**Pass/Fail**
T-9.1 Material	PLA	PLA	PASS
Date Conducted: 10/13/2016			
Team Member Conducting Test: Paul Alabi			
Team Member Signature:			

4.25 - T-9.2 Part Dimensions Test

The part dimensions' test is conducted to ensure all values adhere to the drawing package and manipulator length specification is met.

4.23.1 - Equipment

The following is a list of required equipment to conduct the test:

1. Venire Caliper
2. Required Parts

4.23.2 - Procedure

Begin with the first printed link and proceed to measuring both the total length and the thickness of the link and cross check these values specified in the drawing package. If values match up, the team member conducting the test should indicate the test has been passed, otherwise the test has been failed. The same procedures should be taken for the second and third link printed. The base plate and base plate cover follow the same routine except the measure values being considered are the width and height. These values too need to be cross-referenced with the drawing package and checked off on.

4.23.3 - Data Analysis

Five measurements for each dimension is expected to be recorder and the average of all 5 values will deem the measure value. Standard deviation will be taken to make sure tolerances are met to properly check off on tests. All data analysis will be recorder with Excel.

Table 29. Part Dimensions Testing Table

Part Dimensions Testing

Featured Test	Required Value	Measured Value	Pass/Fail
T-9.2 Part Dimensions			
Link 1 Length	33 mm	32.90 mm	PASS
Link 1 Thickness	3 mm	3 mm	PASS
Link 2 Length	33 mm	33 mm	PASS
Link 2 Thickness	3 mm	3 mm	PASS
Link 3 Length	33 mm	33.1 mm	PASS
Link 3 Thickness	3 mm	3.05 mm	PASS
Base Plate Width	135 mm	135.1 mm	PASS
Base Plate Height	6.5 mm	6.5 mm	PASS
Base Plate Cover Height	30 mm	31 mm	PASS
Base Plate Cover Width	95 mm	94.9 mm	PASS
Date Conducted: 10/19/2016			
Team Member Conducting Test: Paul Alabi & Jonathan Sackett			
Team Member Signature:			

4.26 - T-9.3 Assemblies Test

The assemblies test will be administered to each assembly to ensure proper installation of each piece.

4.26.1 - Equipment

No equipment is required to conduct the assemblies test. All that is required is the assemblies themselves.

4.26.2 - Procedure

Begin with the first assembly by visually inspecting all pieces have been assembled correctly based off the assembly instructions provided in the drawing package. The team member conducting the experiment should verify all steps have been followed properly to ensure the pieces have properly been assembled.

4.26.3 - Data Analysis

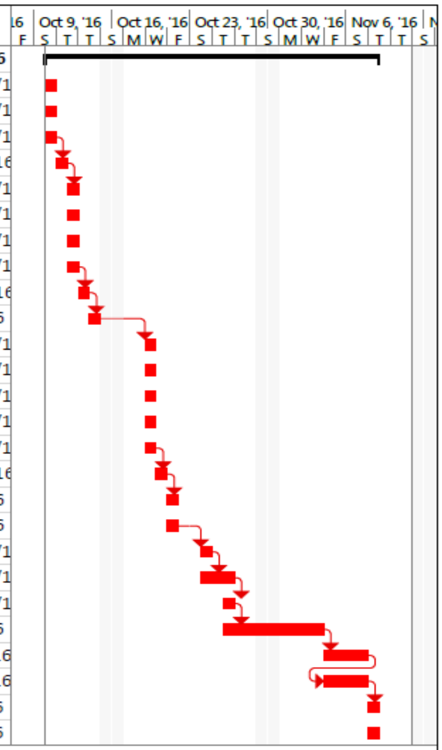
No data collection or analysis will be acquired from this test.

Table 30. Assembly Testing Table

5.0 Schedule

All tests will be conducted either in the Robotics and Motion Laboratory located in the Biotechnology, Sciences and Engineering Building (BSE) 2.216 or at team member Jonathan Sackett's apartment. To maintain project time-management during the execution of the test plan a Test Plan Schedule was created. The test plan will begin 10/10/2016 (Monday) and will last for a total duration of 25 days, ending on 11/11/2016. The Gantt chart for the test plan reflects an earlier deadline for the test which is on 11/08/2016. The team has established that the test plan schedule won't surpass 11/11/2016 giving us 4 days to complete any unfinished testing or data collecting. The deadline gives the team 2 weeks to complete the writing of the Final report which is due on 11/22/2016. The test plan schedule is displayed below in figure 1.

ID		Task Mode	Task Name	Duration	Start	Finish
1			**Project Duration**	**22 days**	**Mon 10/10/1**	**Tue 11/8/16**
2			T-1.1: Load Geometry	1 day	Mon 10/10/1	Mon 10/10/1
3			T-1.2: Load Volume	1 day	Mon 10/10/1	Mon 10/10/1
4			T-1.3: Load Weight	1 day	Mon 10/10/1	Mon 10/10/1
5			T-2.1: Gripping Area	1 day	Tue 10/11/16	Tue 10/11/16
6			T-2.2: Gripping Force	1 day	Wed 10/12/1	Wed 10/12/1
7			T-3.1: Power Supply Voltage	1 day	Wed 10/12/1	Wed 10/12/1
8			T-7.1: Servo Motor Angle Resolution	1 day	Wed 10/12/1	Wed 10/12/1
9			T-3.2: Power Supply Current	1 day	Wed 10/12/1	Wed 10/12/1
10			T-9.1: Material	1 day	Thu 10/13/16	Thu 10/13/16
11			T-7.4: Servo Motor Mass	1 day	Fri 10/14/16	Fri 10/14/16
12			T-4.1: Manipulator Length	1 day	Wed 10/19/1	Wed 10/19/1
13			T-5.1: Degrees of Freedom	1 day	Wed 10/19/1	Wed 10/19/1
14			T-5.2: End Effector Navigation	1 day	Wed 10/19/1	Wed 10/19/1
15			T-9.2: Part Dimensions	1 day	Wed 10/19/1	Wed 10/19/1
16			T-9.3: Assemblies	1 day	Wed 10/19/1	Wed 10/19/1
17			T-5.3: End Effector Position Uncertainty	1 day	Thu 10/20/16	Thu 10/20/16
18			T-5.4: Max Joint Angular Positions	1 day	Fri 10/21/16	Fri 10/21/16
19			T-5.5: Max Joint Angular Velocity	1 day	Fri 10/21/16	Fri 10/21/16
20			T-5.6: Max Joint Angular Acceleration	1 day	Mon 10/24/1	Mon 10/24/1
21			T-5.7: Torque Values	3 days	Mon 10/24/1	Wed 10/26/1
22			T-5.8: Cycle Time	1 day	Wed 10/26/1	Wed 10/26/1
23			T-6.1: Laboratory Experiments	7 days	Wed 10/26/1	Thu 11/3/16
24			T-7.2: Servo Motor Current	2 days	Fri 11/4/16	Mon 11/7/16
25			T-7.3: Servo Motor Voltage	2 days	Fri 11/4/16	Mon 11/7/16
26			T-7.5: Servo Motor Temperature	1 day	Tue 11/8/16	Tue 11/8/16
27			T-8.1: GUI	1 day	Tue 11/8/16	Tue 11/8/16



Project: Test Plan Schedule
Date: Tue 10/4/16

Task | Summary

Figure 1: Test Plan Schedule Overview

Appendix CC

Performance Specifications

1. **Load:**
    1. Cubic geometry
    2. Maximum volume of 1 in.3
    3. Weight ≤ 0.25 lbf
2. **Gripper:**
    1. Purchase off the shelf
    2. Maximum grip area when fully open = 1 in.^{2}
    3. Clamp force ≥ 0.35 lbf
3. **Power Supply:**
    1. Tethered connection
    2. AC-DC converter to supply ≤ 12 V DC
4. **Length:**

1. Arm length between 1.5 and 2 feet

Functional Specifications

5. Manipulator must work with 6 degrees of freedom

5.1. End-effector must navigate through at least ½ of Cartesian coordinate

6. Fulfill 3 laboratory experiments for Fundamentals of Robotics (ME 4773)

6.1 Forward Kinematics

6.2 Inverse Kinematics

6.3 Trajectory generation

7. Servo motors

7.1 Resolution $\leq 5.0°$ per servo

7.2 Supply voltage ≤ 12V

7.3 Mass ≤ 60 grams per servo

8. Easy to use programming interface

8.1 MATLAB and Simulink to control motors

9. Manipulator Performance

9.1 End-Effector position uncertainty = $w_{x,y,z} \leq 3$ in.

9.2 Maximum joint angular position

- Base servo: $0° \leq \theta_{base} \leq 270°$
- Rest of Servos: $0° \leq \theta \leq 180°$

9.3 Maximum joint angular velocity: $0 \leq \omega \leq 20°/sec$

9.4 Maximum joint angular acceleration: $0 \leq \alpha < 10°/sec^2$

9.5 Cycle time ≤ 10 sec