

# **AVATAR XPRIZE ARM**

**MEIE 4702**

## **Technical Design Report**

**AVATAR XPRIZE ARM**

**Final Report**

**Design Advisor: Prof. John Whitney**

**Design Team**

**Nathaniel Berman, Aayush Parekh, Brian Thomas,  
Orion Wilmerding, Dominic Yamarone**

**December 3, 2019**

**Department of Mechanical and Industrial Engineering**

**College of Engineering, Northeastern University**

**Boston, MA 02115**

# **AVATAR XPRIZE ARM**

## **Design Team**

**Nathaniel Berman, Aayush Parekh, Brian Thomas, Orion Wilmerding, Dominic Yamarone**

## **Design Advisor and Sponsor**

**Prof. John Whitney**

## **Abstract**

Transporting a human's presence anywhere in the world presents an opportunity to solve problems and connect people faster than ever. 'Avatar' systems use existing and developing technology to allow users to remotely operate robots using a wearable exoskeleton. This exoskeleton allows the user to transmit his or her movements to the robot, which in turn senses its environment and sends haptic (touch) feedback to the user. The All Nippon Airlines (ANA) Avatar XPRIZE (AAXP) competition provides guidelines for what an Avatar system should be able to accomplish, such as lifting strength, dexterity, and haptic feedback. To meet the competition requirements, the team designed and built a robotic arm and wearable arm exoskeleton, designed a custom motor controller, integrated a haptic interface into the exoskeleton, and operated the arm with the exoskeleton. Team members worked in pairs on these tasks and combined their subsystems into the whole system. A final solution was realized through prototyping, analytical and experimental testing, and multiple redesigns.

## Table of Contents

1.	Acknowledgments	7
2.	Copyright	7
3.	Introduction	8
3.1.	The ANA Avatar XPRIZE Competition	8
3.2.	Capstone Project Scope	8
3.3.	Goals	8
4.	Background	9
4.1.	Key Theories	9
4.1.1.	Human Joint Motion	9
4.1.2.	Robotic Haptic Systems	11
4.1.3.	Brushless Motor Control	14
4.1.4.	Exoskeletons	15
4.1.5.	Kinematics	15
4.2.	Previously Published Work	16
4.2.1.	MIT Mini-Cheetah	16
4.2.2.	Anticogging for Hobby Brushless Motors	17
4.3.	Existing Technologies	17
4.3.1.	Collaborative Robots	17
4.3.2.	da Vinci Surgical Robotics System	18
4.3.3.	FARO Quantum 5 Arm	19
4.3.4.	MIT Hermes Project	20
4.4.	Previous Phases	21
5.	Initial Designs	22
5.1.	Arm	22
5.1.1.	Differential Joint	22
5.1.2.	Arm Spanner	23
5.1.3.	Arm Assembly	24
5.2.	Exoskeleton	25
5.2.1.	Rotation About the Arm	25
5.2.2.	Shoulder	25
5.2.3.	Exoskeleton Assembly	26
5.3.	Haptics	26

5.4.	Motor Controller	29
5.5.	Communication	29
5.6.	Criteria for Final Designs	29
6.	Updated Design	32
6.1.	Arm	32
6.1.1.	Arm Assembly	32
6.1.2.	Differential Joint	33
6.1.3.	Upper Arm Spanner	34
6.1.4.	Shoulder Rocker Assembly	35
6.1.5.	Mounting Frame	36
6.2.	Exoskeleton	37
6.2.1.	Shoulder	37
6.2.2.	Exoskeleton Assembly	38
6.2.3.	Mounting Frame	39
6.3.	Haptics	40
6.4.	Motor Controller	40
6.5.	Communications	41
7.	Analysis and Testing	42
7.1.	Motor Dynamics Modeling	42
7.2.	Arm Assembly Stress Calculations	44
7.3.	Exoskeleton Finite Element Analysis	45
8.	Project Management	46
8.1.	Ownership	46
8.2.	Gantt Chart	47
8.3.	SWOT Analysis	48
8.4.	Budget	48
9.	Future Work	49
9.1.	Mechanical	49
9.2.	Electrical	50
9.3.	Software	50
10.	Summary	51
11.	Intellectual Property	52
11.1.	Description of Problem	52

11.2.	Proof of Concept	52
11.3.	Progress to Date	52
11.4.	Individual Contributions	53
11.5.	Future Work	53
12.	References	54
13.	Appendix A: MATLAB Script	57
14.	Appendix B: Gantt Chart	64

## List of Figures

Figure 1: Shoulder DOF Diagram.....	9
Figure 2: Elbow Range of Motion .....	10
Figure 3: Wrist Range of Motion.....	10
Figure 4: Gimbal Lock Singularity .....	11
Figure 5: Physiological Sense Mapping .....	12
Figure 6: Graphical Representation of the Clarke and Park Transform .....	14
Figure 7: Seated rehabilitation exoskeleton (left) Exoskeleton with rigid linkages (right).....	15
Figure 8: Homogeneous Transform matrix from frame n-1 to frame n.....	16
Figure 9: Form for the Jacobian Matrix for a 6DOF system .....	16
Figure 10: MIT Mini-Cheetah Actuator .....	17
Figure 11: PF3400 arm (left) and UR5 arm (right).....	18
Figure 12: da Vinci Surgical System .....	19
Figure 13: FARO Quantum 5 Arm .....	20
Figure 14: MIT Hermes and Operator .....	21
Figure 15: Preliminary Differential Design .....	22
Figure 16: Differential Assembly .....	23
Figure 17: Differential Assembly Cross Section .....	23
Figure 18: Arm Spanner Assembly.....	24
Figure 19: Arm Assembly.....	24
Figure 20: Exoskeleton Wrist with Cable Drive.....	25
Figure 21: Exoskeleton Shoulder.....	26
Figure 22: Exoskeleton Assembly .....	26
Figure 23: a) Forearm Skin Framework Concept. b) Strain Member Concept Cross Section .....	27
Figure 24: Simply supported beam, shear and bending diagrams .....	28
Figure 25: Top-Level Arm Assembly .....	32
Figure 26: Differential Assembly .....	33
Figure 27: Differential Assembly Cross Section .....	34
Figure 28: Arm Spanner Assembly.....	35
Figure 29: Shoulder Rocker Assembly .....	35
Figure 30: Mounting Frame .....	36
Figure 31: Assembled Arm and Mount .....	37
Figure 32: Shoulder Actuators as Ball and Socket .....	38
Figure 33: Exoskeleton Assembly .....	38
Figure 34: Rendered Exoskeleton Assembly on Mounting Frame .....	39
Figure 35: Exoskeleton Assembly on Mounting Frame .....	40
Figure 36: Custom Motor Controller Architecture .....	41
Figure 37: Step Input Response and Motor Torque Output.....	43
Figure 38: Motor Driver Current Output per Phase.....	43
Figure 39: Shoulder (left), upper arm (middle), and wrist (right) linkages in the exoskeleton .....	46
Figure 40: SWOT Analysis Diagram.....	48

## List of Tables

Table 1 Comparison of workspace and torque limits of human arm and exoskeleton joints [4] .....	11
Table 2 Performance Metrics of Avatar Arm and Exoskeleton Haptic System .....	27
Table 3 Arm Design Requirements.....	30
Table 4 Exoskeleton Design Requirements .....	30
Table 5 Haptics Design Requirements.....	31
Table 6 Communication Interface Design Requirements .....	31
Table 7 Motor Controller Design Requirements .....	31
Table 8 Message Types, ID's and sizes for custom protocol .....	42
Table 9 Command message data breakdown.....	42
Table 10 Force and Moments of Inertia for Extension/Flexion or Abduction/Adduction at Shoulder .....	44
Table 11 Bending and Contact Stresses for Shoulder Bevel Gears .....	45
Table 12 FEA Analysis on Exoskeleton Members .....	46
Table 13 Division of Ownership.....	47
Table 14 Budget.....	49
Table 15 Division of Ownership and Work Accomplished to Date .....	53

## 1. Acknowledgments

The group would like to recognize Professor John Whitney for advising us throughout the project. We would also like to thank the Robotics Collaborative for offering use of their resources. Finally, we thank Markforged for their time and technical resources.

## 2. Copyright

“We the team members, Nathaniel Berman, Aayush Parekh, Brian Thomas, Orion Wilmerding, Dominic Yamarone, and the faculty advisor Professor John Whitney,

Hereby assign our copyright of this report and of the corresponding Executive Summary to the Mechanical, and Industrial Engineering (MIE) Department of Northeastern University.” We also hereby agree that the video of our Oral Presentations is the full property of the MIE Department.

Publication of this report does not constitute approval by Northeastern University, the MIE Department or its faculty members of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

### **3. Introduction**

#### **3.1. *The ANA Avatar XPRIZE Competition***

XPRIZE competitions are meant to spur innovation in technologically challenging industries, such as private spaceflight, renewable energy, and personal healthcare. The All Nippon Airlines (ANA) Avatar XPRIZE (AAXP) competition challenges groups to be “focused on the development of an Avatar System that will transport a human’s sense, actions, and presence to a remote location in real time, leading to a more connected world” [1]. This capstone project will contribute to Northeastern University’s effort to participate in the competition, led by Professors Peter Whitney and Taskin Padir.

The robotic avatar must be controlled by a person; fully autonomous capabilities are discouraged for this competition. The Avatar is expected to involve computer vision to process video and identify relevant task relevant information such as locating an entryway or identifying a handle. It will use haptic feedback technology enabling the user to feel the environment the robot is touching and augment what the user sees through the robot’s cameras using virtual reality. Potential applications for the Avatar system include emergency response for disaster relief, tele-surgery, remote education facilities, or extreme-environment exploration.

#### **3.2. *Capstone Project Scope***

The scope has been focused down to the design and construction of a humanoid arm, from the shoulder to the forearm, emphasizing precise control over power. This project consists of five main parts:

1. A humanoid, robotic arm with five (5) degrees of freedom (DOF) enabling motion in three (3) DOF for the shoulder, one (1) DOF for the elbow, and one (1) DOF for wrist rotation.
2. A custom-designed motor controller for managing each motor’s actions.
3. A haptic feedback system that relays force from the robotic arm to the user.
4. A tele-operation communication method for communicating between the robotic arm and the user.
5. An exoskeleton suit, worn by the user, that acts as a controller for the robotic arm and a carrier of the haptic feedback system.

#### **3.3. *Goals***

The Avatar is expected to operate in three modes, as described by the official competition guidelines: Operator Control Mode, Enhanced Avatar Mode, and Semi-Autonomous Mode. For the scope of the project, the team will focus on Operator Control Mode only. A number of suggested tests are outlined in the competition guidelines; however, for the purpose of this capstone project, success will be defined from accomplishing the following tasks [1]:

- The user, through the Avatar Arm, physically feels a handshake.
- The user, through the Avatar Arm, physically feels a hug.
- The user, through the Avatar Arm, can feel and guess the weight of a cube that weighs between 0.5 and 3kg.

In addition to the quantitative design requirements, the group has set various functional goals to test the capabilities of the system against. In the initial discussions of the scope of the project, the group identified the design divergence between a low power, high precision arm and exoskeleton and a less precise but more physically capable duo. After discussion on this subject, the group decided to focus on precision motion over powerful motion in the design of both the arm and exoskeleton. The group is also focused on creating a thoughtful user experience. The experience of using the arm should be intuitive and as smooth as possible, minimizing exposed wires, drives, and other peripheral components. The Avatar Arm will also be designed with low cost in mind, with a focus on usage of 3D printing parts where possible.

## 4. Background

### 4.1. Key Theories

#### 4.1.1. Human Joint Motion

The Avatar system is meant to precisely transfer human motion and dexterity. To achieve this in a robotic medium, the robot must be designed to be highly biomimetic. In this case, the Avatar Arm design must include a mirrored set of degrees of freedom to a human arm. Beginning with the shoulder, three (3) distinct degrees of freedom can be identified: flexion and extension, abduction and adduction, and inward and outward rotation. A diagram of this motion can be seen in Figure 1.

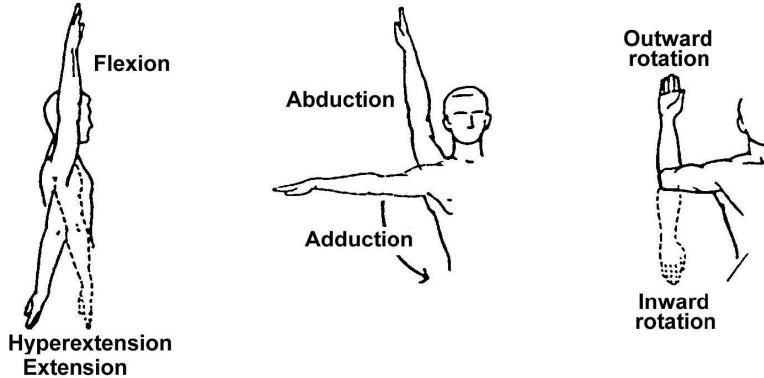
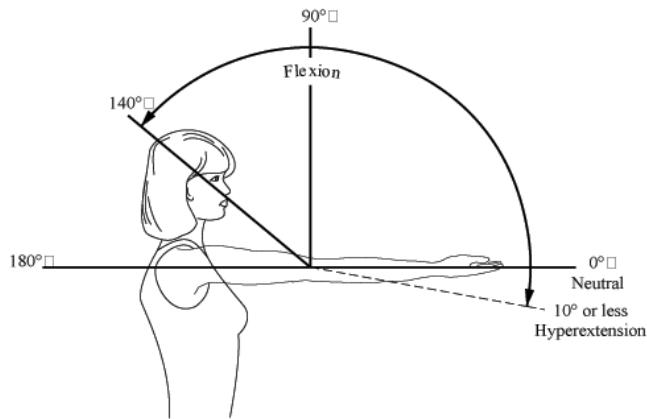


Figure 1: Shoulder DOF Diagram [2]

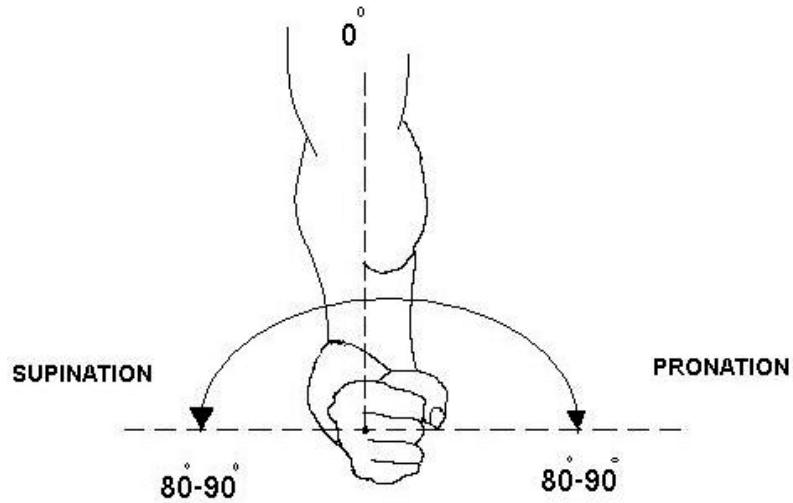
Flexion and extension cover a workable range of about  $220^\circ$ , abduction and adduction require an angle range of  $150^\circ$ , and inward and outward rotation cover a  $180^\circ$  range [3]. These movement profiles and ranges must be accurately mapped to the arm in order to create a realistic experience.

The elbow is a much simpler joint, only containing one direct degree of freedom. The elbow provides flexion and extension. Range of motion is shown in Figure 2.



**Figure 2: Elbow Range of Motion [2]**

As shown, the designed elbow must carry a  $140^\circ$ - $150^\circ$  adjustable range to completely encompass the human workable range [3]. The rotation at the wrist is driven by muscles within the forearm and thus must be incorporated into the scope of the arm. Figure 3 below shows supination and pronation range of motion of the wrist.



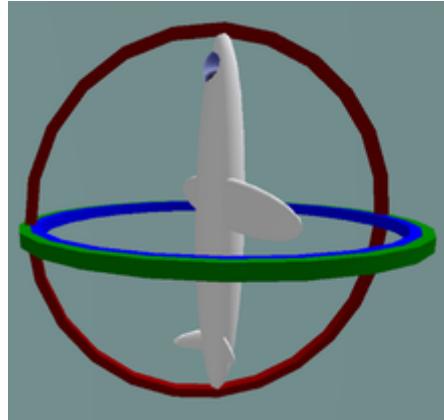
**Figure 3: Wrist Range of Motion [2]**

Based on these motion profiles and workable areas, the arm and exoskeleton can be designed to incorporate the entire workable area of a human arm to create the most intuitive experience possible. A full summary of a human arm's 'workspace' is described in Table 1.

**Table 1 Comparison of workspace and torque limits of human arm and exoskeleton joints [4]**

Joint	Human Isometric Strength <sup>1</sup>	Human Joint Workspace Limits	Torque Specification	Workspace Specification	Peak Torque Output Capability	Workspace Capability
Elbow Flexion/Extension	72.5 Nm	Flexion: 146° Extension: 0°	6 Nm	Flexion: 120° Extension: 0°	5.46 Nm	Flexion: 90° Extension: 0°
Forearm Supination/Pronation	9.1 Nm	Supination: 86° Pronation: 71°	5 Nm	Supination: 90° Pronation: 90°	5.08 Nm	Supination: 90° Pronation: 90°
Wrist Palmar/Dorsal Flexion	19.8 Nm	Palmar Flexion: 73° Dorsiflexion: 71°	4 Nm	Palmar Flexion: 60° Dorsiflexion: 60°	≈ 0.4 Nm @ $\alpha = 30^\circ; \beta = 9^\circ$	Palmar Flexion: 60° Dorsiflexion: > 60°
Wrist Abduction/Adduction	20.8 Nm	Adduction: 33° Abduction: 19°	4 Nm	Adduction: 30° Abduction: 30°	≈ 0.4 Nm @ $\alpha = 21^\circ; \beta = -11^\circ$	Adduction: 30° Abduction: > 30°

A final consideration in biomimicry is avoiding mechanical singularity. In robotics, singularity, sometimes referred to as kinematic singularity, describes a state in which a robot cannot move its end effector in a specific direction or point in space within its workable area due to actuator and joint arrangement. An example is gimbal lock, shown in Figure 4 where two gimbals are in the same plane and one (1) DOF is lost.



**Figure 4: Gimbal Lock Singularity [5]**

#### 4.1.2. Robotic Haptic Systems

Haptics simply means touch. The haptic component of an Avatar system encompasses sensors to measure user inputs, sensors to measure Avatar stimuli, and transducers to feed sensory data back to the user. An ideal haptic system would simulate the Avatar's environment along the user's skin as if the Avatar's senses were their own. This is not possible with current technologies, and a subset of senses must be selected to simulate touch.

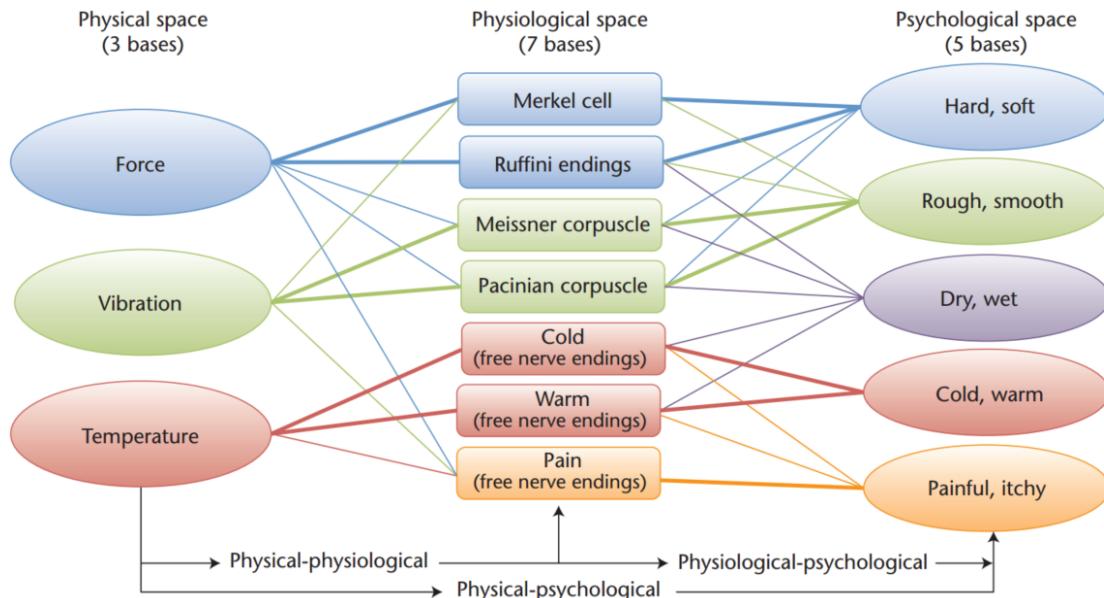
Because haptic interfaces vary significantly in use case, they also vary significantly in what kind of information is transmitted to the user and how. There is, however, one consistent gauge of a haptic interface's quality: system transparency. Transparency, also known as system stiffness, is a measure of how quickly and accurately an Avatar tracks a user's movements. Friction, inertia, backlash,

communication latency, sensor accuracy and a dozen more variables conspire to reduce transparency between user and Avatar [4]. An ideal Avatar Exoskeleton would demonstrate infinite stiffness, where the forces exerted on the arm would be mirrored by the forces exerted by the exoskeleton on the user without limitation or lag.

The core of a robot's functionality is its actuators. Each degree of freedom in the Avatar Arm is driven by an actuator equipped with an encoder that provides precise position information to the control system. What enables feedback are mirrored joint pairs. Each degree of freedom in the arm is mirrored by an actuated joint mechanically or functionally identical to the exoskeleton, meaning it can exert a force on the exoskeleton and user. For example, when the Avatar interacts with a wall, the exoskeleton will exert an inverse and proportional force against the user attempts to change the exoskeleton's joint positions beyond the Avatar's measured joint position. The result is a virtual wall. Similarly, a small weight held by the Avatar Arm will be translated into a force at the joint as if the operators are holding the weight themselves. Care will be taken when selecting the connection method between exoskeleton and operator to minimize latency that would reduce the systems overall transparency.

Actuator-encoder pairs provide a large amount of sensory data, however they do not mimic a person's "skin," providing forces only at the joints/points of attachment between exoskeleton and operator. A 1N force at the end of the Avatar's lever arm produces the same feedback as a 2N force at half the length. Furthermore, joint actuators cannot transmit hot or cold, wet or dry, or smooth or textured sensations, and they are blind to the area of skin affected.

Human skin encompasses seven cell types that detect three distinct senses, force, vibration and temperature, mapping to five psychological spaces. This mapping is shown in Figure 5.



**Figure 5: Physiological Sense Mapping [6]**

The complexity of human skin, and the wide array of senses that can be transmitted necessitate a narrowing of focus depending on the application of the haptic sensor. For example, gaming, one of the

larger markets for haptic feedback systems, has sensory input from the physics engine to vibrators within a controller while some robotic surgery systems can use strain gauges to provide force feedback to a haptic glove [7].

Most modern haptic systems that do not rely solely on motorized joints focus on creating a grid or strategic alignment of transducers along the operator's hand and body. Predictably, complexity and expense increase exponentially with transducers per area density. Transducer selection also plays a key role in the complexity of the design. Of the three (3) physical spaces listed in Figure 5, force applied at nodes along the skin is the most intuitive for users to understand, but the most complex to design.

Vibrotactile sensation is one of the most common implementation of haptic feedback. Most common due to its simplicity and cost efficacy is vibrotactile sensation. Vibrotactile transducers rely on human's ability to interpret or code one sensation as another. By varying the frequency and intensity of vibration, a human can interpret magnitude and location of force. The downside of vibrotactile sensation is that vibration is hard to localize and cannot achieve the same pinpoint accuracy as direct force transducers [8].

[9] describes a glove embedded with a plurality of vibrators to enable haptic feedback via vibrotactile sensation for surgery. The vibrator motors are positioned longitudinally along the finger to wrist to provide localized force information. The exoskeleton could implement a similar vibrotactile grid as a cost effective pseudo-haptic system, however, the controls for this grid of motors would be too complex for the scope of this capstone project. This is similar to [10] but varies in method of implementation. [10] describes a glove that measures finger position and transmits haptic feedback to the operator's fingertips through piezoelectric actuators. Piezoelectric materials generate a charge when placed under mechanical stress and vice versa. Multi-layer piezoelectric actuators dot the surface of the fingertip pads and provide calibrated vibration that can be interpreted as force.

An alternate method of haptic feedback without motors, is outlined by [11]. Small sacs filled with magnetic fluids are elongated by the application of a magnetic force from a variable intensity electromagnet. The strength of the magnetic field is calibrated against input forces on the sensor side of the haptic system. The elongation of the pliable vesicle presses against a human's fingers providing a scaled sense of touch [11]. A different patent from Worcester Polytechnic Institute outlines the use of air-filled silicone domes that inflate and deflate depending on sensor input, which creates the sense of gripping an object [12]. These implementations, however, are fairly bulky, adding considerable weight to the exoskeleton. As such, its current use is found in gloves, for use in conjunction with a virtual reality environment.

Lastly, there is a fourth sensation occasionally used to code as force: electrotactile. A grid of electrodes can deliver highly localized sensation through a flexible transducer as thin as 0.1mm. On the downside, to overcome the impedance of the skin, voltages are typically painfully high. This relegates electrotactile grids to highly specialized applications [13].

#### 4.1.3. Brushless Motor Control

The Avatar Arm makes use of electric motors to create a highly dynamic robotic arm. For a thorough and detailed summary of the principles of electric motors from their construction to control methods, see [14]. The following section will summarize the information applicable to the Avatar project.

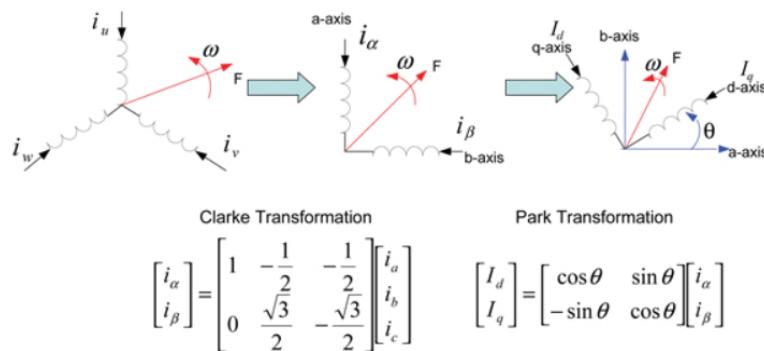
Electric motors come in a wide variety of electrical and physical configurations. This project is making use of brushless permanent magnet synchronous motors. These motors are driven by three (3) sources of alternating current; each is  $120^\circ$  out of phase. These sources drive three (3) sets of electromagnets, called windings, that turn on and off in sequence, attracting permanent magnets on the rotor. This synchronous switching drives the rotation of the motor. The strength of this magnetic field is what produces torque, and is linearly correlated to the magnitude of the current in the windings. The speed of the motor is controlled by the frequency that the windings switch.

The equations that define both the torque output and motor speed are highly dependent on the sinusoidal signal driving the windings. The mathematical representation of these phases can be simplified using phasor notation, separating the signal into a scalar magnitude and a complex exponential representing the sinusoid.

Because these three (3) phases are always  $120^\circ$  out of phase, their net instantaneous current, and therefore field strength can be mapped to a vector on a two dimensional plane, dependent only on the rotational position of one of the phases. This state vector is calculated using the Clarke Transform [14].

A further transform, called the Park Transform, remaps this vector to a reference frame that is independent of the rotational position of the phases. It produces an orthogonal component and a parallel component of the magnetic field. The orthogonal component directly produces torque, while the parallel component produces an axial load that is absorbed by the bearings in the motor.

Performance of the motor is optimized by maximizing the orthogonal component in this reference frame. Inverse Park and Clarke transforms are then used to calculate the necessary inputs to each of the windings. This theory of control is called Field Oriented control. Both transforms are graphically represented in Figure 6.



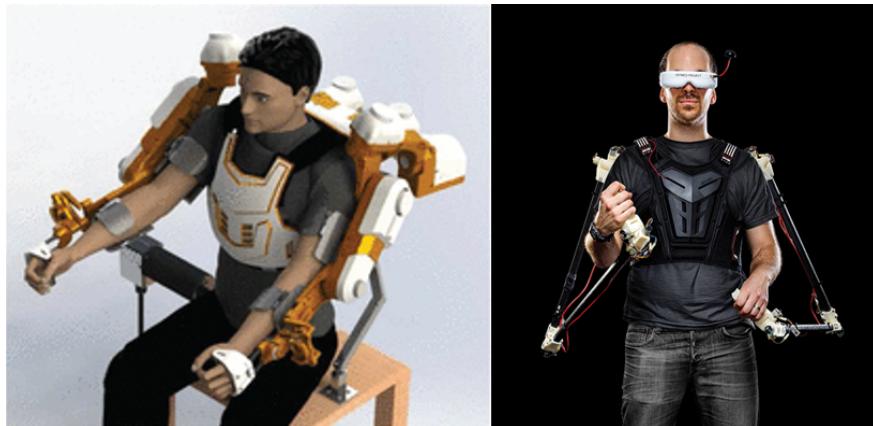
**Figure 6: Graphical Representation of the Clarke and Park Transform [15]**

#### 4.1.4. Exoskeletons

An exoskeleton is a rigid structure outside of a human's body that moves along with their body as they perform tasks. Exoskeletons typically fall into one of two categories. The first is a suit that enhances a user's abilities like speed, strength, and endurance. Exoskeletons of this type are used in applications such as reducing fatigue in the workplace, rehabilitation, and military operations.

The second category is an apparatus that captures a user's motion but does not enhance the user's abilities. Instead, this exoskeleton sends movement information that can be used to control a robot or teach a program how to move more like a human [17]. Exoskeletons for motion capture rely on encoders at joints to detect movement and determine positions. In the cases where the exoskeleton is used to control a robot, this information lets the robot's actuators mirror the user's movement. This project will use a motion-capture exoskeleton as there is no need to augment strength.

As a system that is worn by a user, whether suspended from a fixed base or free to move, an exoskeleton must fit the contours of a body and allow for full range of motion. Some exoskeletons require the user to enter into a structure that allows movement of limbs but not the entire structure. Other exoskeletons use braces around a user's body that carry the weight and straps that attach rigid, skeletal linkages to the body. Figure 7 shows an example of an exoskeleton fixed to a chair and a different exoskeleton that uses long linkages to capture motion.



**Figure 7: Seated rehabilitation exoskeleton (left) Exoskeleton with rigid linkages (right) [16, 18]**

#### 4.1.5. Kinematics

An Avatar system is forward drivable, meaning that the position and rotation of each joint can be used to calculate the end effector location and rotation. It is also backdrivable, meaning that a desired end effector location and rotation can be achieved by driving motors to precise locations. These are calculated through forward and inverse kinematics respectively.

While simple kinematics can be described using a system of equations, more complicated robots with many DOF lend themselves toward linear algebra. For this project, the Denavit Hartenberg method was used to calculate the homogeneous transformation matrices as shown in Figure 8:

$$H_n^{n-1} = \begin{bmatrix} C(\theta_n) & -S(\theta_n)C(\alpha_n) & S(\theta_n)S(\alpha_n) & r_n C(\theta_n) \\ S(\theta_n) & C(\theta_n)C(\alpha_n) & -C(\theta_n)S(\alpha_n) & r_n S(\theta_n) \\ 0 & S(\alpha_n) & C(\alpha_n) & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

**Figure 8: Homogeneous Transform matrix from frame n-1 to frame n**

The homogeneous transformation matrix is used to calculate the Jacobian and inverse Jacobian matrices. Where the homogenous transform matrix is useful for calculating position and rotation, the Jacobian is used to calculate velocity and angular velocity vectors. It is a set of first order differential equations used in many control systems as part of the motion planning control.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} J_{11}\dot{q}_1 + J_{12}\dot{q}_2 + \dots + J_{16}\dot{q}_6 \\ J_{21}\dot{q}_1 + J_{22}\dot{q}_2 + \dots + J_{26}\dot{q}_6 \\ J_{31}\dot{q}_1 + J_{32}\dot{q}_2 + \dots + J_{36}\dot{q}_6 \\ J_{41}\dot{q}_1 + J_{42}\dot{q}_2 + \dots + J_{46}\dot{q}_6 \\ J_{51}\dot{q}_1 + J_{52}\dot{q}_2 + \dots + J_{56}\dot{q}_6 \\ J_{61}\dot{q}_1 + J_{62}\dot{q}_2 + \dots + J_{66}\dot{q}_6 \end{bmatrix}$$

**Figure 9: Form for the Jacobian Matrix for a 6DOF system**

#### 4.2. *Previously Published Work*

##### 4.2.1. MIT Mini-Cheetah

In his Master's thesis, Ben Katz of MIT details the process of developing low cost actuators from hobbyist brushless motors [19]. His work is specifically applied to the development of a quadruped walking robot that is a smaller and much lower cost version of MIT's Cheetah robot. The motor control theory and controller development detailed in [19] is highly relevant to the Avatar Arm project. Both projects have similar goals of being able to develop compliant and highly dynamic actuators for use in a biomimetic system. Both projects are also taking advantage of readily available hobby brushless motors. Projects that similarly use brushless motors and have been well documented in research include the Stanford Doggo [20] and the Berkeley Blue Arm [21].

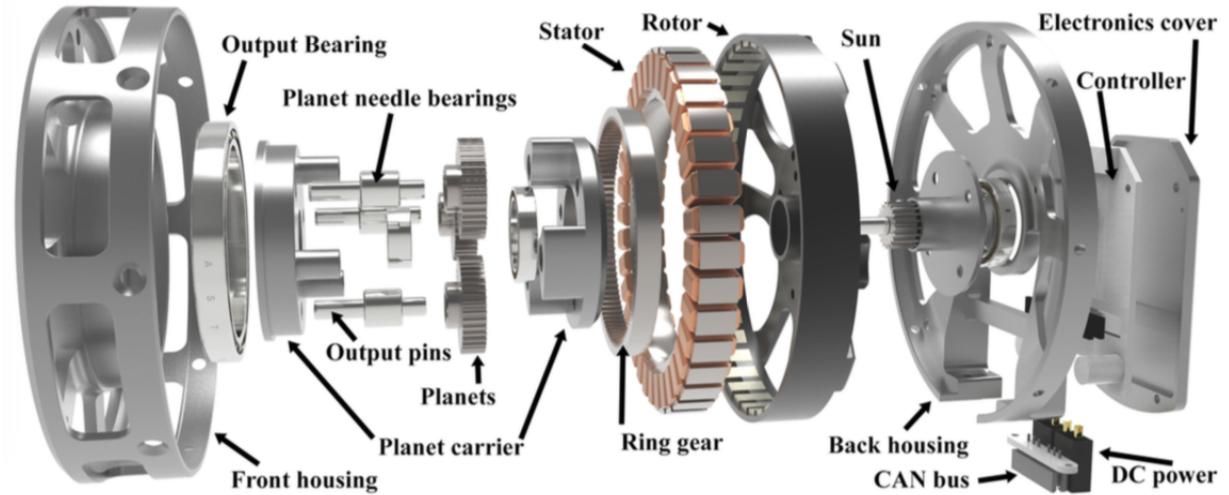


Figure 10: MIT Mini-Cheetah Actuator [19]

The actuator developed for the Mini-Cheetah, shown in Figure 8, integrates a motor controller and planetary gear transmission into a compact housing. The Avatar Arm project aims to similarly integrate a custom motor controller to a commercial brushless motor. Unlike the Mini-Cheetah actuators, Avatar Arm will not be rehousing the motor electronics in an entirely new casing, but instead mounting a housing to the rear of the motor.

#### 4.2.2. Anticogging for Hobby Brushless Motors

Matthew Piccoli and Mark Yim of the University of Pennsylvania describe a method for significantly reducing the torque ripple caused by magnetic cogging in brushless motors [22]. Torque ripple is a variation in the output torque that causes a bumpiness, or roughness on the output. Most noticeably, torque ripple is caused by cogging in the motor. This cogging is caused by the magnetic fields of the magnets on the rotor aligning with the winding fields.

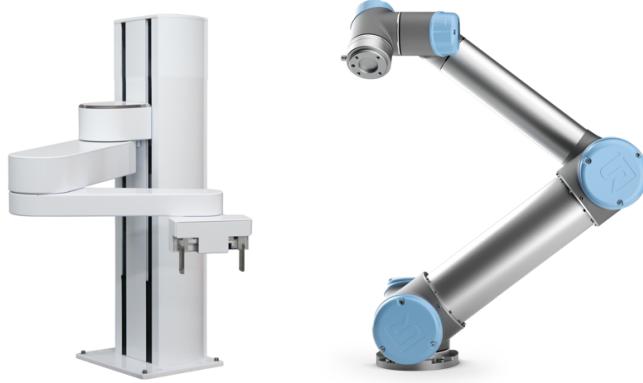
This torque ripple is detrimental to a haptic system, where it causes inaccuracies in force and position transmission. The method proposed in this paper produces a cogging torque map, relating cogging torque to rotational position, that is then fed into the software controller and used to compensate for the torque loss from cogging. The Avatar Arm team plans to use a similar method to develop a cogging map for the actuator. This will reduce inaccuracies in the motion of both the arm and the exoskeleton to provide a higher quality haptic experience.

### 4.3. ***Existing Technologies***

#### 4.3.1. Collaborative Robots

As the Avatar Arm is designed to mimic human operation, it is necessary to look at existing robotic arms that achieve a similar functional levels. Collaborative robots (cobots) are designed to work alongside people on assembly lines or in production facilities. As opposed to industrial robots that operate in a

safeguarded environment or at a distance from people, cobots can interact with people in a safe manner. They are easily programmable from a simple environment, and can be equipped with a variety of end effectors to accomplish specialized tasks such as moving product from one conveyor line to another. As such, each cobot is designed for a particular application, such as pick-and-place or manipulating objects in various orientations. Figure 9 shows examples of collaborative robots, such as UR5 robot from Universal Robotics, and the PF3400 Arm from Precise Automation [23, 24].



**Figure 11: PF3400 arm (left) and UR5 arm (right) [23, 24]**

Most relevant to the Avatar Arm project is their collision detection behavior, which simply recognizes when a joint has bumped into an object or human during the job. A method for collision detection without specialized sensors is to collect position, velocity, and acceleration data from an encoder on a motor, and calculate a ‘crash threshold value’ to determine how large of an error in those parameters are correlated to a collision. This is used in combination with a low-pass filter to reduce noise the system sees. It is easily implementable through a few lines of code [25].

#### 4.3.2. da Vinci Surgical Robotics System

Surgical robotic systems feature extremely precise movement of any number of arms; they are typically teleoperated from a computer or control panel. Figure 10 shows an industry-standard surgical robotic arm system, the da Vinci Arm from Intuitive Surgical [26].



**Figure 12: da Vinci Surgical System [26]**

Each arm can be equipped with a variety of surgical instruments, such as needle drivers, graspers, scissors, staplers, etc. The system is modular, so parts can be swapped out as needed without damaging motors or cables of the system. The downside of surgical arms like da Vinci is that they do not incorporate haptic feedback and do not mirror a human arm's functionality.

The capstone project is aiming to see how natural human movement can be translated into a similar kind of modular and precise robotic arm. An application for this would be a surgeon performing surgery without having to be in the operating room and learning the functions of a consumer control panel. It is difficult to manipulate arms from only dials and switches. The typical learning curve for the non-haptic da Vinci system is 55 procedures at 95 minutes per operation [27]. With a human using it, there is more flexibility and intuition afforded. Shaky surgical instruments may be an issue on a human-controlled, teleoperated robotic arm, however, there is a method for ensuring a stable end effector despite robotic motion. Two sets of filters are used in a control system: one for the input to reduce the effects of incoming commands that cause tip vibrations, and an inverse filter to compensate for any delay that was caused by the first [28].

#### 4.3.3. FARO Quantum 5 Arm

The FARO Quantum 5 arm, shown in Figure 11, is a high accuracy encoded arm designed and manufactured by FARO technologies. This device incorporates five (5) degrees of freedom in a five (5) joint design, allowing for the tool head to operate in a 1.5 meter radius of the base.

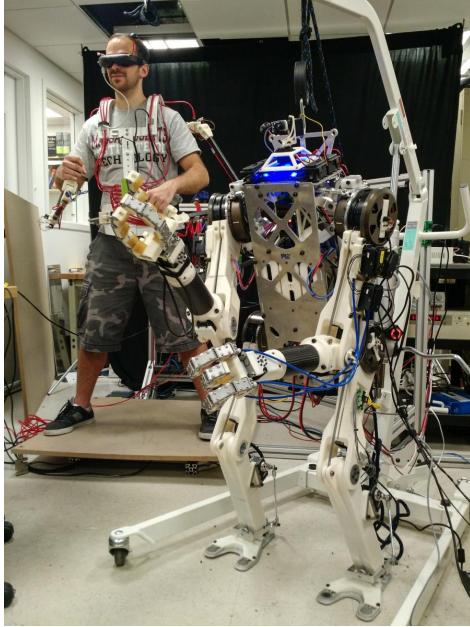


**Figure 13: FARO Quantum 5 Arm [29]**

Using high resolution encoders, the FARO Arm is able to kinematically map the position and orientation of the tool head at any given point in space, allowing for unlimited operational flexibility for the user. Data is gathered using either the laser or touch off probe at the tip of the tool head. The low friction and highly encoded joints are a great guide for the design of the Avatar Exoskeleton with the addition of drivable motors.

#### 4.3.4. MIT Hermes Project

The MIT Hermes project is a full humanoid robot controlled by a user wearing an exoskeleton [17]. The purpose of this project is to be able to teleoperate a robot in hazardous environments that humans cannot withstand. The impetus for this project was the incident at the Fukushima nuclear power plant where treaded robots were insufficient to traverse the affected environment and thus provided only limited help.



**Figure 14: MIT Hermes and Operator [18]**

Hermes, shown above in Figure 12, is designed for high force tasks like turning a shutoff valve or clearing heavy debris. To accomplish these kinds of tasks, the robot has six (6) DOF on each limb and uses the same motors as the MIT Mini-Cheetah. The robot has balance feedback that detects when a user is about to make the robot tip over and applies a force on the user's hips to avoid having the robot fall over [30]. This type of feedback will be useful in the Avatar Arm when it detects an obstacle and drives the motors on the exoskeleton to oppose the user's motion.

The design of the Hermes exoskeleton provides a good example of how to control a robotic arm. In the Hermes project, the user operates the arm by holding a joystick attached to linkages with encoders at the joints. The team will use a similar setup with the addition of driven motors and more direct attachment with the limb for haptic feedback.

#### 4.4. *Previous Phases*

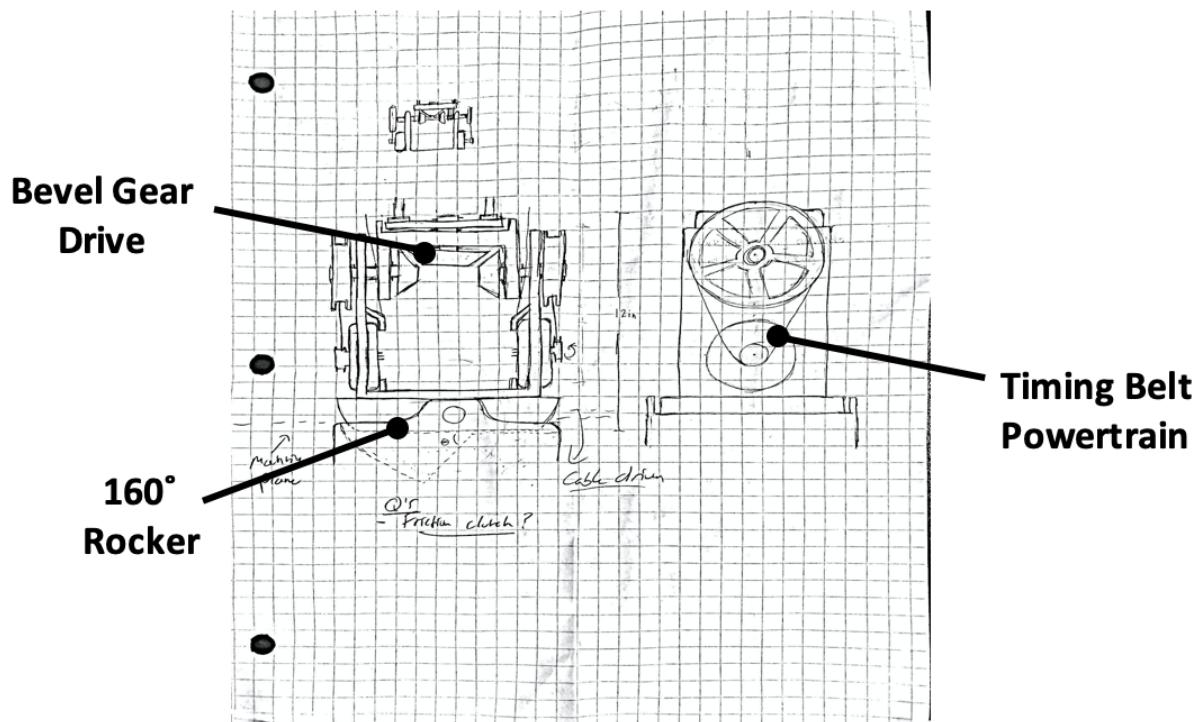
There is a Summer 2019/Spring 2020 capstone group working on designing an Avatar Haptic End Effector. The project plans for creating at least two fingers and one thumb as part of a biomimetic human hand, able to grasp and sense simple objects through actuated finger joints. Communication with the Avatar Haptic End Effector team was established early in the Summer II cycle. It was decided for the Avatar Arm team to provide a mounting interface on the robotic arm for the Haptic End Effector team to mount to. In addition, scope was narrowed down for the robotic arm to have five (5) DOF, with the intention for two (2) of the three (3) DOF in the wrist to be implemented by the Haptic End Effector team.

## 5. Initial Designs

### 5.1. Arm

#### 5.1.1. Differential Joint

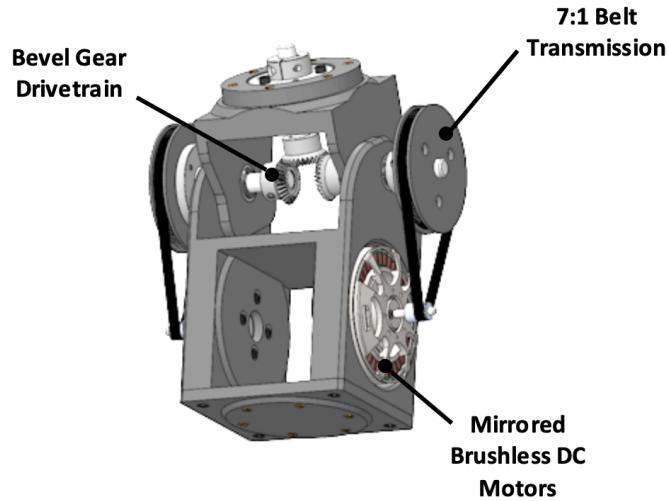
The first part of the arm design taken on was the shoulder joint, as this incorporates the greatest number of degrees of freedom (3). Based on initial research, the group began experimenting with differential joints to combine two (2) degrees of freedom into a relatively compact package. A preliminary design for such a joint is shown below in Figure 13.



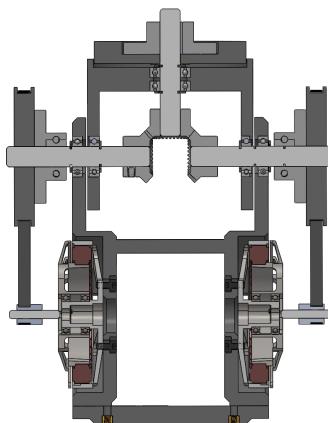
**Figure 15: Preliminary Differential Design**

The two (2) motor configuration allows for a differential mitre gear train to translate both rotational and axial motion. A third degree of freedom is located below the differential in the form of a rocker. These three (3) degrees of freedom translate directly to the three modes of human shoulder movement, abduction/adduction, flexion/extension and humeral rotation. In this design, the brushless motors drive large-toothed pulleys through a timing belt with a gear ratio of 7:1 to achieve a peak output torque of 15N·m per motor. Another advantage of the differential design is that it can be mirrored on the elbow to provide flexion and extension as well as pronation and supination at the wrist. This arrangement greatly reduces the arm complexity and cost compared to two distinct designs for the shoulder and elbow.

After a review of the design, a model was created to more completely flush out details. A rendering of the design and cross section view taken across the gear plane are shown below in Figures 14 and 15 respectively.



**Figure 16: Differential Assembly**



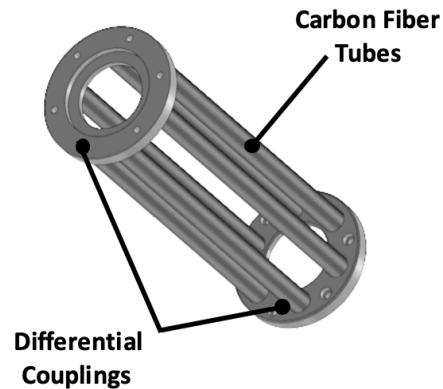
**Figure 17: Differential Assembly Cross Section**

This iteration includes more precise bearing and shaft constraints as well as a variable belt tensioning system through slots in the motor mounting positions.

#### 5.1.2. Arm Spanner

To connect the joints, the group targeted a lightweight spanner with minimal complexity that allows for easy wire routing, high bending and torsional stiffness, and modular components. With this criteria, the

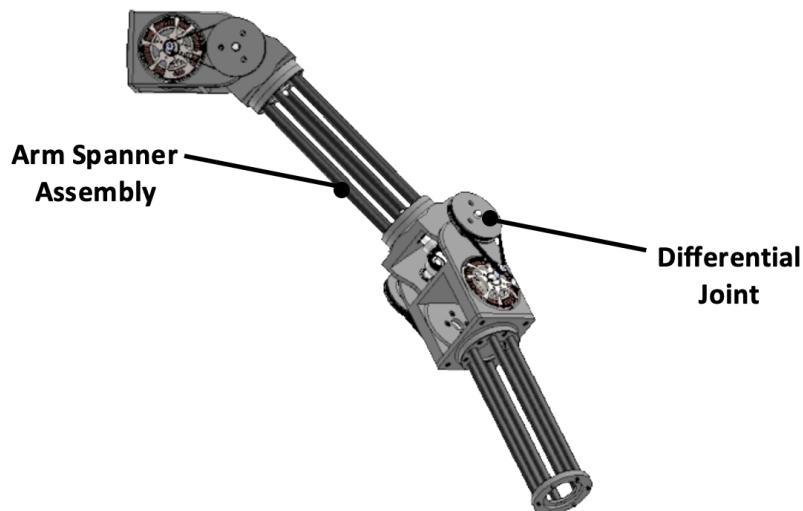
design was quickly narrowed down to carbon fiber tube and 3D printed couplers for each end. The carbon fiber tubes offer exceptional torsional and bending stiffness in an extremely lightweight package. The 3D printed end caps will be bonded to each end of the tubing to create a strong connection without risk of damage to the tube through over clamping. With the design settled, a CAD model was created and is shown in Figure 16 below.



**Figure 18: Arm Spanner Assembly**

#### 5.1.3. Arm Assembly

As mentioned, with the modularity of the differential joint and spanner, the components are able to be easily patterned across the arm assembly. The full arm assembly can be seen in Figure 17 below.



**Figure 19: Arm Assembly**

## 5.2. *Exoskeleton*

The main goals in designing the exoskeleton are control of the robotic arm, minimizing weight and friction, and ensuring the user has a full range of motion. Each motorized degree of freedom in the arm will be mirrored by motorized joints in the exoskeleton so it can resist a user's motion when encountering a virtual wall. The motor package will include the encoders that send position information to the robotic arm. For the first iteration of the exoskeleton, the same motors and encoders from the arm were used to ensure uniformity across the two systems. To accommodate the irregular shape of a human arm and minimize weight, the exoskeleton was made of 3D printed components.

### 5.2.1. Rotation About the Arm

Multiple axes of rotation lie within the human arm. These include the rotation of the wrist and outward/inward rotation in the upper arm. Unlike the AVATAR Arm, which can have gears and spanners along these axes, the Exoskeleton cannot because the user's arm occupies that space. Therefore, the team devised a mechanism to allow for haptic feedback along these axes. The component with the motor has a C-shaped opening for the user's arm, and the motor is mounted so that its axis is parallel to that of the C. This motor connects to a cable attached to another component that uses the C like a rail and rotates about it concentrically. A low-friction material, such as PTFE (polytetrafluoroethylene), should be used everywhere the two components interface. Figure 18 shows this mechanism in the wrist. The Exoskeleton linkage coming from the elbow joint holds the motor and also has the C-shape. The motor links to a cable attached to the wrist/forearm linkage, so that when the user rotates his or her wrist, the cable turns the motor and the encoder senses the rotation.

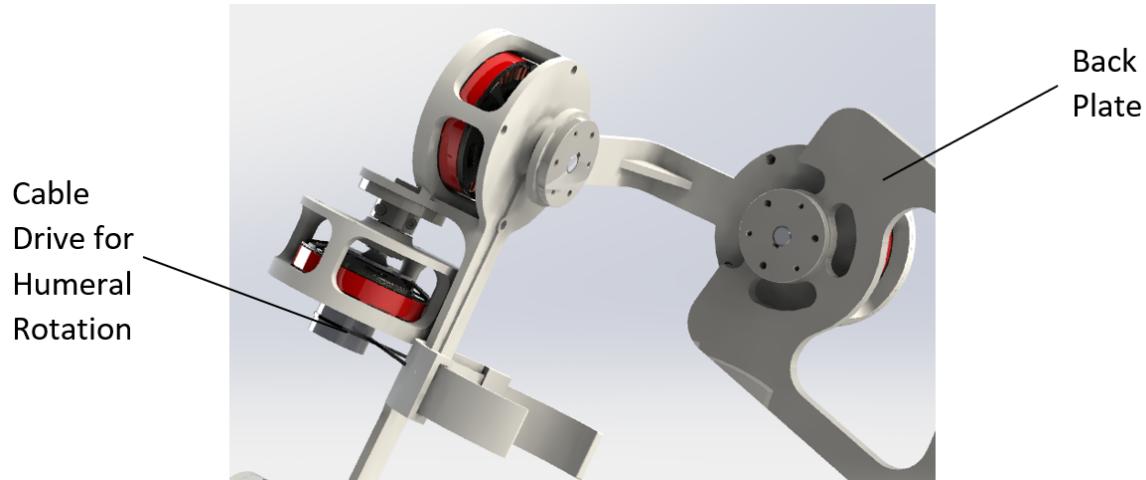


**Figure 20: Exoskeleton Wrist with Cable Drive**

### 5.2.2. Shoulder

Human shoulders are ball and socket joints. This configuration cannot be built into the Exoskeleton as the user occupies the center of rotation. To mirror this movement, the Exoskeleton incorporates three (3) degrees of rotation. Outward/inward rotation is covered by the mechanism detailed in the previous section. The remaining two (2) degrees of freedom are represented with a motor mounted at the user's

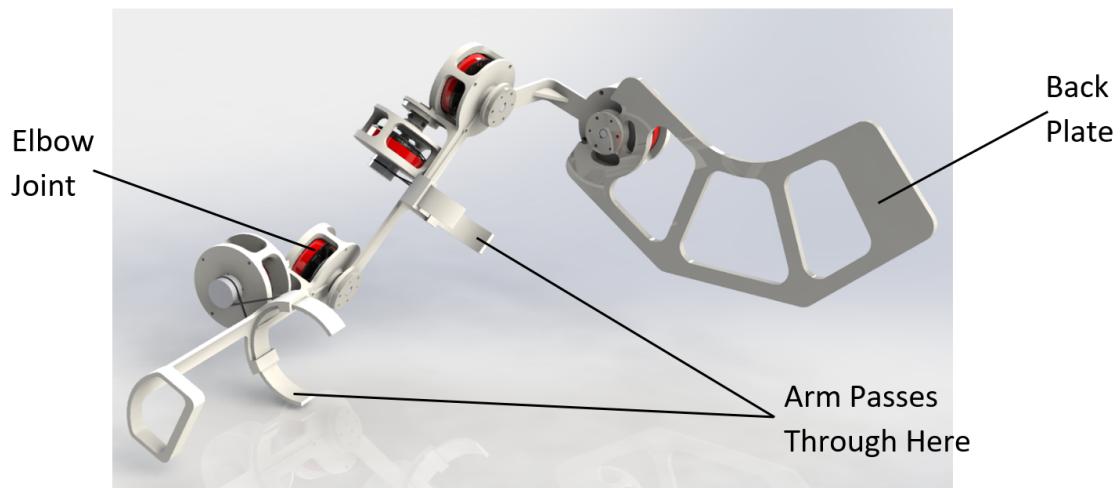
back for shoulder abduction/adduction and a motor whose axis of rotation is perpendicular to that for shoulder flexion and extension. The point at which these two (2) axes meet is the center of the ball of the ball and socket in the user's arm so that the motion of a human can still be accurately transmitted. Figure 19 shows the three (3) motors used in the shoulder of the Exoskeleton.



**Figure 21: Exoskeleton Shoulder**

### 5.2.3. Exoskeleton Assembly

The last joint is flexion and extension in the elbow. This joint is straightforward as it can mirror a human arm more easily than the ball and socket of the shoulder or rotation in the wrist. Figure 20 shows this within the entire Exoskeleton assembly. This minimalist design connects joints with small, stiff linkages and secures motors at each joint. Mechanical stops prevent users from over extending joints.



**Figure 22: Exoskeleton Assembly**

### 5.3. *Haptics*

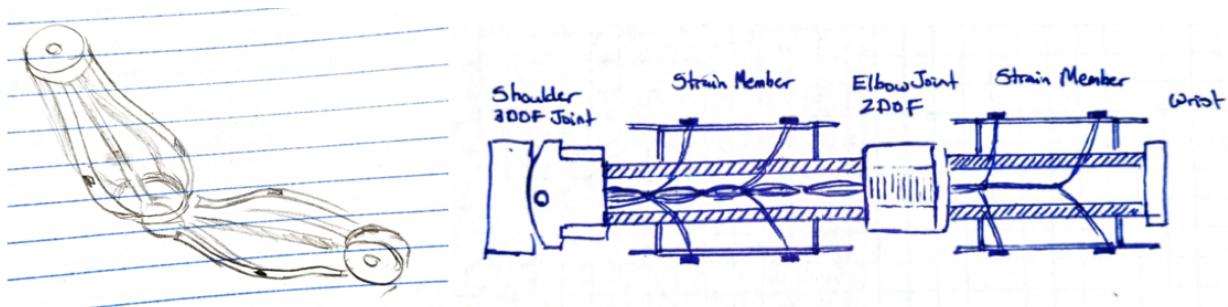
The basis of the Avatar Arm's haptic system will be motor-encoder pairs mirrored between the arm and exoskeleton's joints with a control system maximizing virtual stiffness between them. This will provide

macro-force feedback to the operator's joints and exoskeleton attachments and enable simulation of a virtual resistance up to 30N of force (~3kg lifting capacity). All of the factors that affect transparency, friction, low stiffness, high inertia, communication latency, backlash and encoder alignment, will undergo iterative designs to maximize system stiffness. The ultimate measure of the haptic system will be the transparency of force feedback. Success in creating a high transparency system is measured on a sliding scale, not a binary pass/fail. Based on the requirements from the AAXP, the team has developed performance-based metrics of increasing difficulty, listed in Table 2.

**Table 2 Performance Metrics of Avatar Arm and Exoskeleton Haptic System**

Differentiating Mass (kg)	Writing Legibility (font size)	Threading A Loop (Eye width, mm)	Reaction Speed (Type of ball)	Letter Grade
Failed at any mass	Failed at any font size	Failed at any loop size	Failed at any ball size	F
3	100	10	Beach Ball, Large	D
1	60	5	Beach Ball, Small	C
0.5	30	3	Wiffle Ball	B
0.25	20	1	Ping Pong	A

If time and resources allow, the Arm may be fitted with a “skin” equipped with strain gauges to sense a force’s magnitude and location. Rather than a complete shell, the “skin” will be thin, lightweight beams running longitudinally to the arm and spaced radially around the arm above and below the elbow as described in Figure 21.



**Figure 23: a) Forearm Skin Framework Concept. b) Strain Member Concept Cross Section**

Using two strain gauges on a simply supported beam, it is possible to calculate the location,  $x$ , of a force

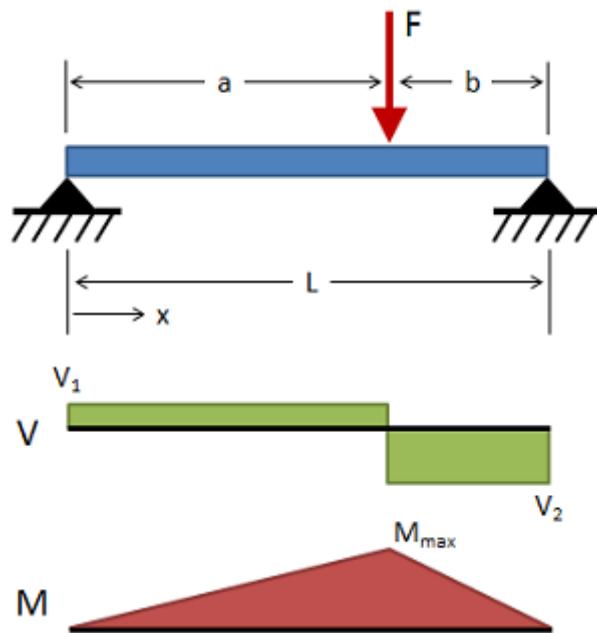
$$\sigma = \frac{Mc}{I} = \varepsilon E \quad \text{Eq. 1}$$

Where  $\sigma$  is the stress,  $M$  is the moment at the point of stress,  $c$  is the characteristic length,  $I$  is the moment of inertia,  $\varepsilon$  is the strain and  $E$  is the Young's modulus. For a simply supported beam as described in Figure 22. The moment is described as

$$M = \frac{Fxb}{L}, 0 \leq x < a \quad \text{Eq. 2}$$

$$M = \frac{Fab}{L}, x = a \quad \text{Eq. 3}$$

$$M = \frac{Fax}{L}, a < x \leq b \quad \text{Eq. 4}$$



**Figure 24: Simply supported beam, shear and bending diagrams [31]**

Using the moment of inertia for a beam and solving equations 1 and 3 for  $x$ , the location of force can be determined from strain measured at points 1 and 2.

$$a = \frac{\varepsilon_2 x_1 L}{\varepsilon_1 x_2 + \varepsilon_2 x_1} \quad \text{Eq. 5}$$

Preliminary research demonstrated that force actuators, such as described in [10] would be prohibitively time consuming to develop within the scope of the project. Testing will prioritize vibration motors in a coarse grid pattern across the haptic interface to provide general localization of forces along the operator's skin. The final design will be selected based on minimizing the complexity or wiring,

minimizing weight of sensors and skin as well as interfacing with the communication protocol between arm and exoskeleton.

#### **5.4. Motor Controller**

The custom motor controller design is heavily based on open source brushless motor servo drives such as the ODrive [32] and MIT Mini-Cheetah Controllers [19]. These controllers make use of a high performance microcontroller, three phase driver integrated circuit; typically from Texas Instruments (TI); and a high resolution absolute encoder. These components allow all control logic calculations to be done in near real time, independent of the communications network. Communications with other components or a main computer is typically done over high speed protocols; up to 5Mbps.

The custom controller will use a TI Tiva Series Microcontroller with integrated encoder interface, multiple serial communication busses, and high speed pulse width modulation. The features of this microcontroller will help reduce adverse effects such as torque ripple and communications latency. The group will use a similar, but newer version of the three phase driver that both the ODrive and Mini-Cheetah use, as well as an updated model of the absolute encoder from the Mini Cheetah. The controller will implement Field Oriented Control to provide both torque and position control of the motor. As a result of the newer three phase driver and updated absolute encoder, the group expects similar, if not better, performance than equivalent systems like the Stanford Doggo [20] and Mini-Cheetah [19].

#### **5.5. Communication**

Smooth communication between arm and exoskeleton is required for a high performance haptic system. Typical requirements are a latency below 30ms and a sampling rate greater than 750Hz [33]. As discussed in the motor controller design, the communication protocol has been narrowed down to CAN Bus, an automotive industry standard, RS-485 and SPI. In addition to the communication speed constraints, minimizing wiring and using well supported systems such as I2C and CAN is a priority.

#### **5.6. Criteria for Final Designs**

To qualify a final design, the group identified key metrics to achieve a successful final product. These metrics were developed both through research on human capabilities as well as the XPRIZE competition requirements. These metrics encompass capacity, workable area, speed and accuracy. A complete tabulated list of the design requirements for each project subsystem can be found below in Table 3.

**Table 3 Arm Design Requirements**

Requirement	Value
Degrees of Freedom	5
Workable Area	36" Radius
Lifting Capacity	3kg
Max End Effector Speed	0.2 m/s
End Effector Positional Accuracy	<5mm

The arm design requirement table is an amalgamation of specifications taken from several sources in order to holistically evaluate and compare the Avatar Arm's integration with, and performance against a human arm. As seen in Figures 14 and 17, the human shoulder and elbow have a combined five (5) degrees of freedom. In order to accurately mirror human motion, the Avatar Arm must also include five (5) degrees of freedom. Requirements for workable area and max end effector speed were derived from human-performance metrics. In normal use, humans reach a maximum instantaneous wrist velocity of 0.2m/s [34], a speed that the Avatar Arm should be capable of matching. Similarly, the workable area of a 36" radius from the shoulder again mirrors typical human capabilities based on the arm span and workable area of a 6ft tall human. The end effector positional accuracy and lifting capacity are based on the competition requirements outlined in Section 1.3. These two parameters are set to encompass minimum requirements to perform specific tasks with the necessary strength and dexterity.

**Table 4 Exoskeleton Design Requirements**

Requirement	Value
Degrees of Freedom	4
Workable Area	36" Radius
Weight	5 kg

Unlike the Avatar Arm, the exoskeleton does not have a fifth degree of freedom. The mechanism for wrist rotation will be developed by a future capstone team. This exoskeleton instead has four (4) degrees of freedom, as detailed in Table 4. These mirror the degrees of freedom in the shoulder and elbow of the Avatar Arm. The workable area is the same as the Avatar Arm.

**Table 5 Haptics Design Requirements**

Requirement	Value
Range of Mass Detection	0.5-3.0 kg
Skin Localization Area	400 mm <sup>2</sup>

The requirements listed in Table 5 are defined by both internal requirements and the Avatar XPRIZE Challenge guidelines. The competition requires a user to be able to distinguish weights between 0.5 and 3 kg. Metrics for testing this are defined in Table 2. Internally, the team has defined a metric for assessing the performance of a haptic skin. However, haptic skin is a reach goal for this project.

**Table 6 Communication Interface Design Requirements**

Requirement	Value
Communication Rate	> 750Hz
Latency	< 30ms

The requirements listed in Table 6 largely focus on minimizing latency with high speed. Latency is the time between the sending of a message and its receipt on another part of the system. The communication rate is the number of times that a node can be updated per second. This is largely dependent on the amount of data that the communication protocol can transfer per second and the number of nodes in the system. While these requirements are agnostic of protocol, they will be deciding factors for the final implementation.

**Table 7 Motor Controller Design Requirements**

Requirement	Value
Maximum Current Output	60 A
Operating Voltage	24 V
Encoder Resolution	2K ppr or 14 bit
Minimum Driver PWM Rate	16 kHz

The requirements for the motor controller hardware are detailed in Table 7. The intention of these requirements is to reduce inaccuracies in the movement of the motors in order to have a higher quality haptic experience. The maximum current output and operating voltage requirements define the amount of power the motor controller will be able to supply to a motor. This will affect the amount of torque the arm

can exert at each joint, and therefore its “strength.” The encoder resolution, in units of pulses per revolution (ppr) for analog signals, or bits for digital signals, will determine the sensitivity of the Avatar Arm. Higher resolution encoders will allow for smaller changes in position to be sensed and fed back to the user. The driver pulse width modulation (PWM) rate is the speed at which the controller will switch on and off to regulate power output. A low PWM rate induces jitter into the motion of the arm and causes the motion to feel rough. The requirements set for the Avatar Arm controller are comparable with the specifications of controllers used in similar projects such as the Stanford Doggo [20] and MIT Mini-Cheetah [19]. The Avatar Arm controller is expected to exceed these requirements.

## 6. Updated Design

### 6.1. Arm

#### 6.1.1. Arm Assembly

The arm layout is relatively unchanged beyond the significant design updates to the differential, arm spanner, and shoulder rocker sub-assemblies. The current assembly is shown in Figure 23.

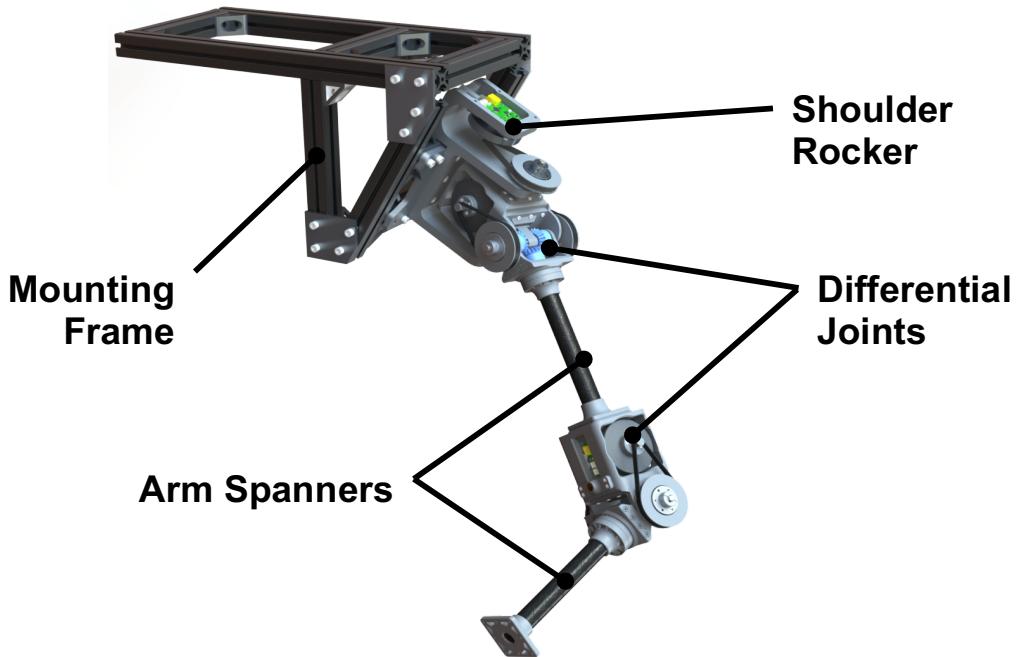


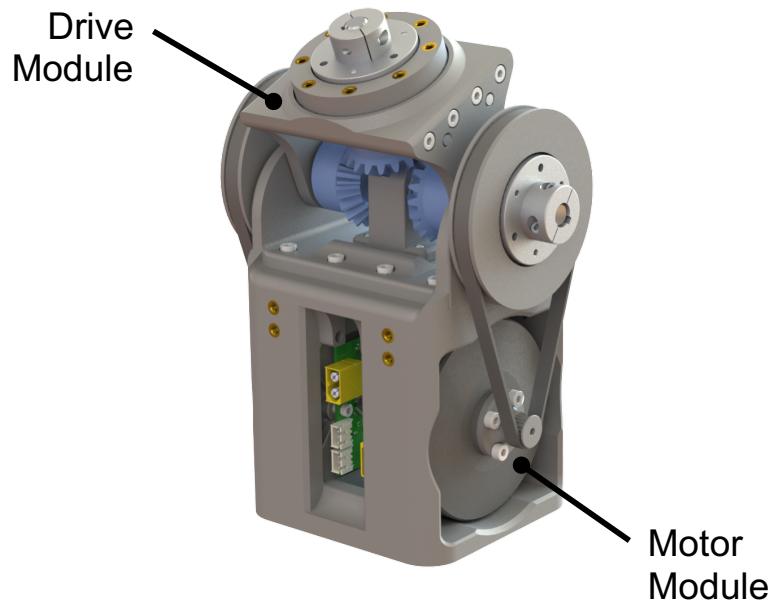
Figure 25: Top-Level Arm Assembly

Making up the arm are two identical differential joints at the shoulder and elbow, both providing two degrees of freedom. The elbow differential provides flexion and extension at the elbow joint as well as pronation and supination at the wrist. Two arm spanners connect the joints. At the shoulder, the shoulder rocker assembly provides the last degree of freedom required in the joint. The shoulder rocker assembly is rigidly attached to the mounting frame assembly. This assembly is designed to sit and clamp to a table to

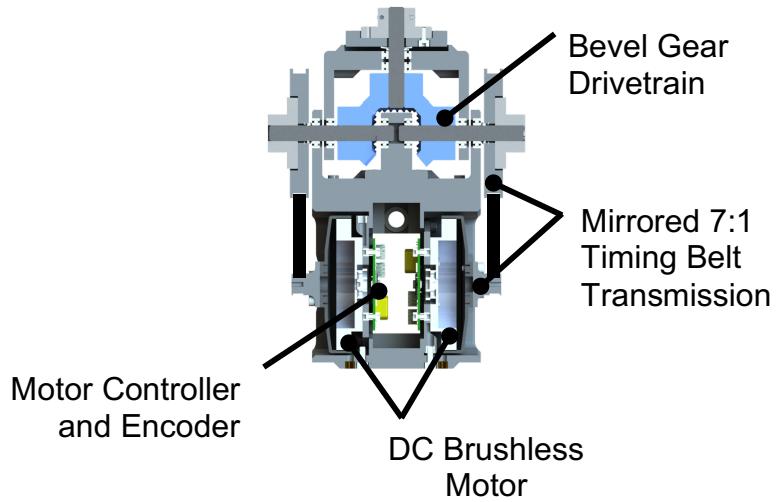
provide a rigid base for the arm. The arm is mounted at a 30 degree angle from the vertical in order to minimize the occurrence of a vertical singularity at the arms resting position.

### 6.1.2. Differential Joint

Upon formal review of the initial model, there were several points of improvement to incorporate into an updated and more refined design. The initial rendering seen in Figures 13 and 14 show considerable unused spaced within the design envelope. This unused space represents non-optimized weight and packaging utilization. Figure 24 below shows the updated design which was designed to be much more space efficient and with a focus on ease-of-assembly. This attention to packaging is also well-illustrated when comparing the cross sections of Figure 24 and Figure 25. As a product of the industry design review, a central bearing support block was added to reduce the cantilever load on the differential bearings and shafts.



**Figure 26: Differential Assembly**

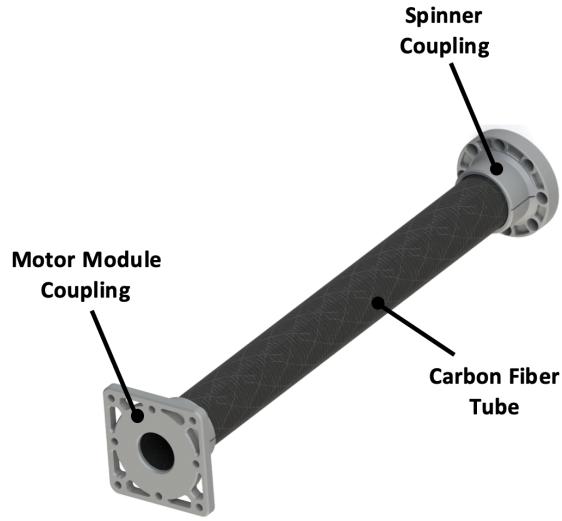


**Figure 27: Differential Assembly Cross Section**

Other updates in the differential design include a stepped down shaft size from 12mm to 10mm and a transition to machined plastic gears for further weight optimization. Lastly, to decrease the slop in the system and increase the overall accuracy, the bearings are now axially preloaded using a combination of shaft clips, stacked wave disk springs, and belleville washers.

#### 6.1.3. Upper Arm Spanner

The upper arm spanner changed considerably from the group's initial design, shown in Figure 26. The group opted to continue with a single large carbon fiber tube rather than six (6) smaller tubes. With this change, the torsional stiffness of the spanner increases drastically. This change also improved the manufacturability of the assembly as the relative lengths and tolerances of the six (6) tubes are no longer a concern in the bonding process.

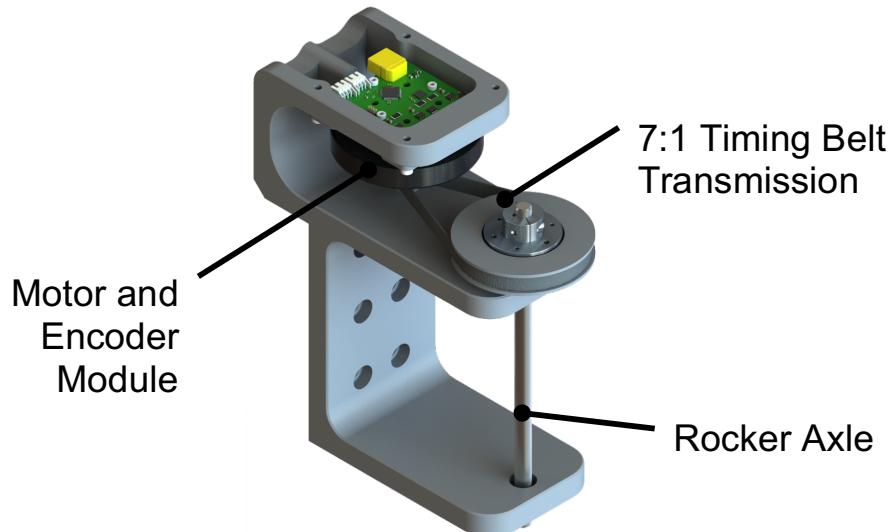


**Figure 28: Arm Spanner Assembly**

The updated design no longer incorporates identical mirrored couplers. The updated design has unique couplers for each side of the differential to better optimize for the local mounting geometry.

#### 6.1.4. Shoulder Rocker Assembly

The shoulder rocker assembly provides the third degree of freedom in the shoulder joint. This assembly uses the same motor and pulley assembly to rotate the shoulder differential around its central axis. A diagram of the joint can be found below in Figure 27.

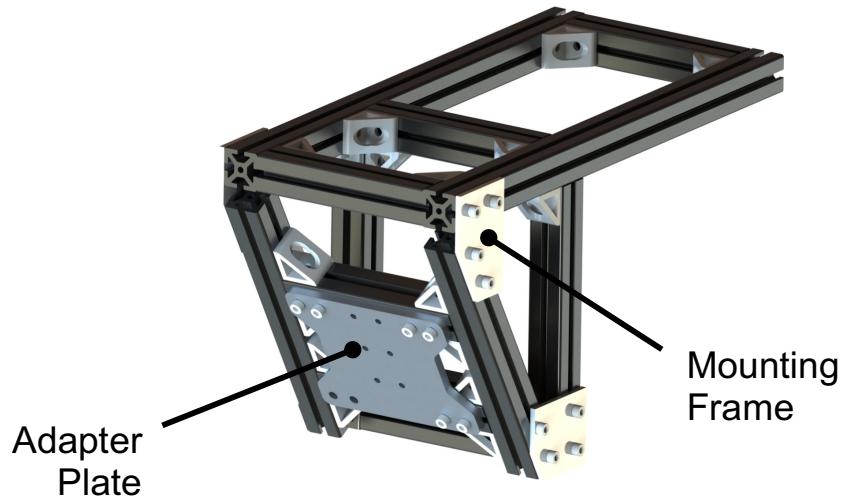


**Figure 29: Shoulder Rocker Assembly**

The encoder and motor module drive a 7:1 timing belt drive train attached to the central rocker axle. The shoulder differential is attached and rotationally coupled to this axle using two clamping assemblies mounted on the differential joint. The body of the rocker is attached to the mounting frame using screws and an adapter plate.

#### 6.1.5. Mounting Frame

The last piece of the arm assembly is the mounting frame. The mounting frame is designed to be clamped to an existing table or countertop such that the arm hangs over the edge. The frame is comprised of modular aluminum extrusion and connecting brackets. The adapter plate is mounted on the angled face and was manufactured from  $\frac{3}{8}$ " aluminum sheet. Figure 28 below shows a rendering of the mounting frame. Figure 29 shows the assembled design.



**Figure 30: Mounting Frame**

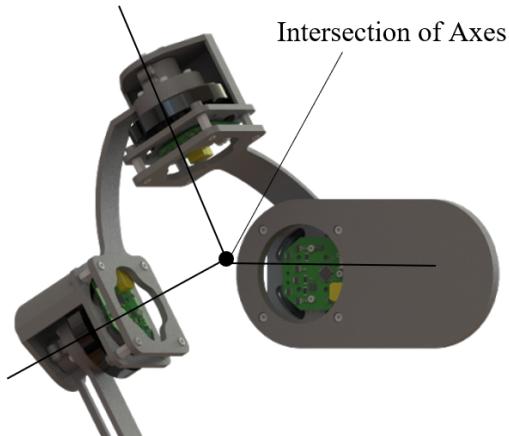


**Figure 31: Assembled Arm and Mount**

## 6.2. *Exoskeleton*

### 6.2.1. Shoulder

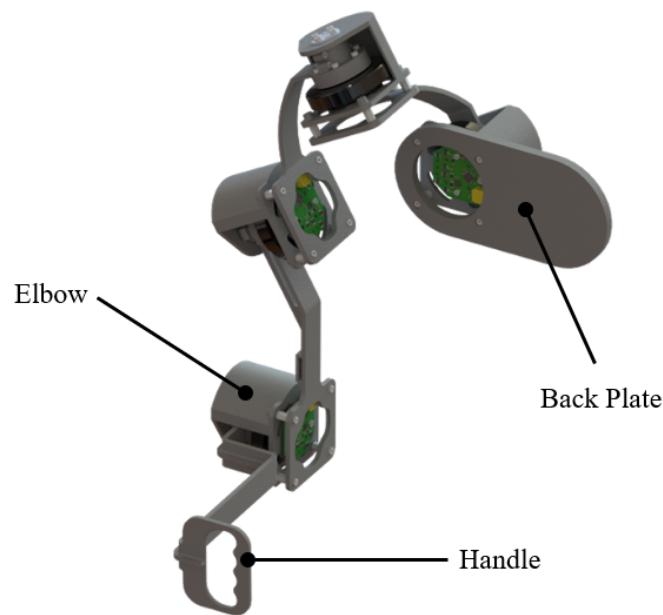
Upon review, the cable driven method of achieving humeral and wrist rotation was found to be inadequate. Pre-tensioning of the cable and surface contact would likely prevent the sliding components from moving due to increased friction, so another solution was required. The three (3) motors in the shoulder were reoriented to better match a ball and socket joint. Figure 30 shows the axes of rotation of the motors intersecting at one point, which represents the center of the ball. This solution eliminates the friction of the initial design and better approximates a human arm. This solution takes inspiration from Harmony, a rehabilitation exoskeleton that assists patients' arm motion [16]. Harmony has five (5) degrees of freedom in the shoulder but only three (3) have axes that intersect at one point. Their design effectively approximates shoulder movement and avoids hitting the user with the exoskeleton.



**Figure 32: Shoulder Actuators as Ball and Socket**

#### 6.2.2. Exoskeleton Assembly

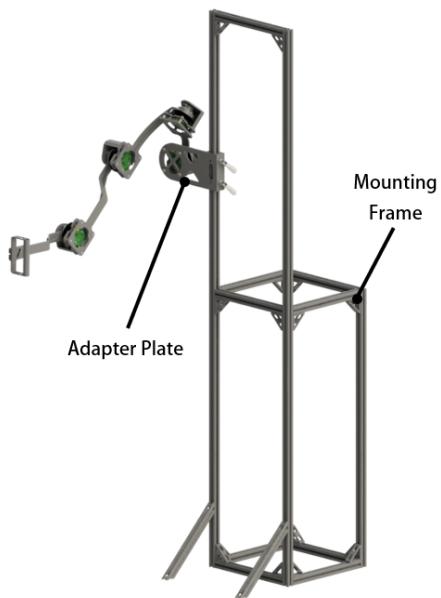
The exoskeleton has been updated to encompass only four (4) degrees of freedom. After the design review, the team decided to eliminate the motor in the wrist for wrist rotation. A future goal is to include an encoder at the wrist so the exoskeleton can still transmit the motion to the arm, but currently the exoskeleton does not allow wrist motion. The motors in the initial design were mounted on shafts, but in the current design they act as structural components. Although this puts more stress on the motors, the exoskeleton becomes much slimmer and lighter. In addition, this makes mounting the motor controller boards considerably easier. Slots in the forearm and upper arm linkages hold straps that the user will wear to control the exoskeleton as closely as possible. Figure 31 shows the entire exoskeleton mounted onto a backplate. This backplate was mounted onto an 8020 frame with adjustable height. This also helps to support the weight of the exoskeleton as well as assist with wire management.



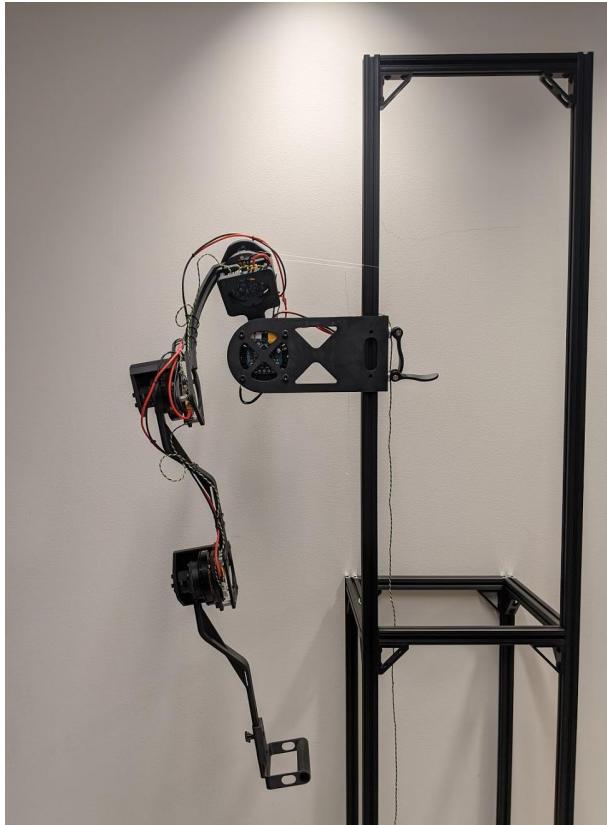
**Figure 33: Exoskeleton Assembly**

### 6.2.3. Mounting Frame

The exoskeleton mounting frame is designed to stand on its own while holding the exoskeleton. Users stand in front of the frame and use the straps on the exoskeleton to attach themselves to the system. The adapter plate can slide vertically to account for users of different heights. The frame is comprised of modular aluminum extrusion and connecting brackets. The adapter plate was 3D printed. Figure 32 below shows a rendering of the mounting frame. Figure 33 shows the assembled exoskeleton mounted on the frame.



**Figure 34: Rendered Exoskeleton Assembly on Mounting Frame**



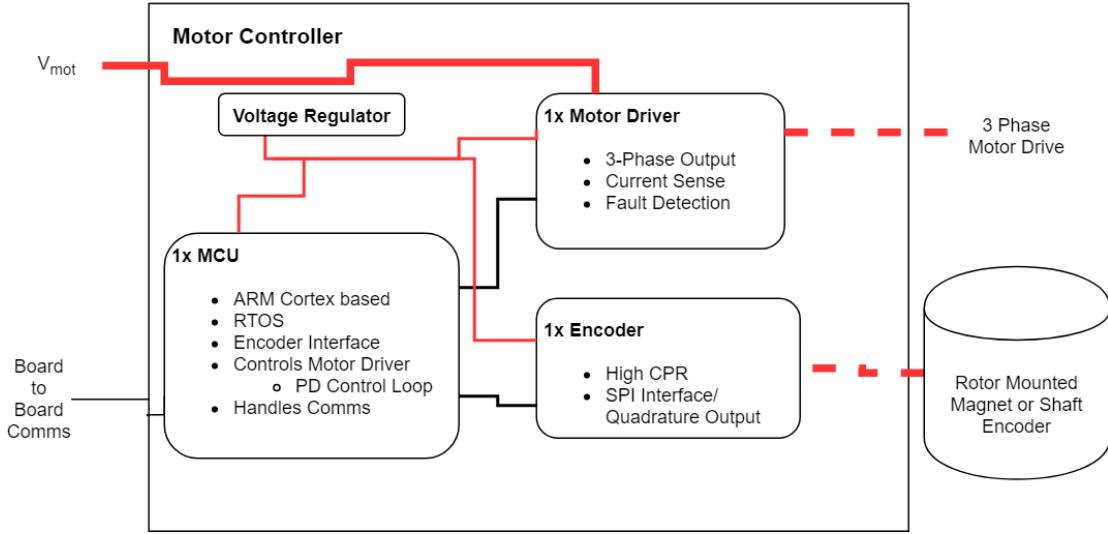
**Figure 35: Exoskeleton Assembly on Mounting Frame**

### **6.3. *Haptics***

The basis of the Avatar Arm's haptic system will be motor-encoder pairs mirrored between the arm and exoskeleton's joints with a control system maximizing virtual stiffness between them. This will provide macro-force feedback to the operator's joints and exoskeleton attachments and enable simulation of a virtual resistance up to 30N of force (~3kg lifting capacity). All of the factors that affect transparency, friction, low stiffness, high inertia, communication latency, backlash and encoder alignment, will undergo iterative designs to maximize system stiffness. The ultimate measure of this component of the haptic system will be the transparency of force feedback.

### **6.4. *Motor Controller***

The design of the motor controller largely remains the same as the initial design. The block diagram of the custom motor controller is shown below in Figure 34. The custom controller will use a TI Tiva Series Microcontroller with integrated encoder interface, multiple serial communication busses, and high speed pulse width modulation. The features of this microcontroller will help reduce adverse effects such as torque ripple and communications latency. The group will use a similar, but newer version of the three phase driver that both the ODrive and Mini-Cheetah use, as well as an updated model of the absolute encoder from the Mini Cheetah.



**Figure 36: Custom Motor Controller Architecture**

The controller will implement Field Oriented Control to provide both torque and position control of the motor. To do this, the controller runs a chain of control loops in order to produce precise position tracking. An outer loop controls the position of the motor using a software PD controller. This loop creates a virtual spring by mapping a difference in position to a desired torque. The inner loop then controls the current to the motor based on the desired torque. The group expects similar, if not better, performance than equivalent systems like the Stanford Doggo [20] and Mini-Cheetah [19].

## 6.5. *Communications*

In accordance to the requirements laid out in the Initial Design Section, the communication protocol has been narrowed down to RS-485. This is a well-supported hardware protocol that will allow high speed serial communication. RS-485 only defines the hardware requirements for data transmission. The software protocol used with RS-485 will be a custom serial protocol. This allows the team to define shorter, efficient messages. Currently, there are three message types defined. A message to request position data from the motor, one to send a position command to the motor, and one for the motor to respond with. In addition to the communication speed constraints, minimizing wiring is a priority.

**Table 8 Message Types, ID's and sizes for custom protocol**

Message Type	Message ID	Length in Bytes
Request	0x00	2
Response	0x01	6
Command	0x02	5

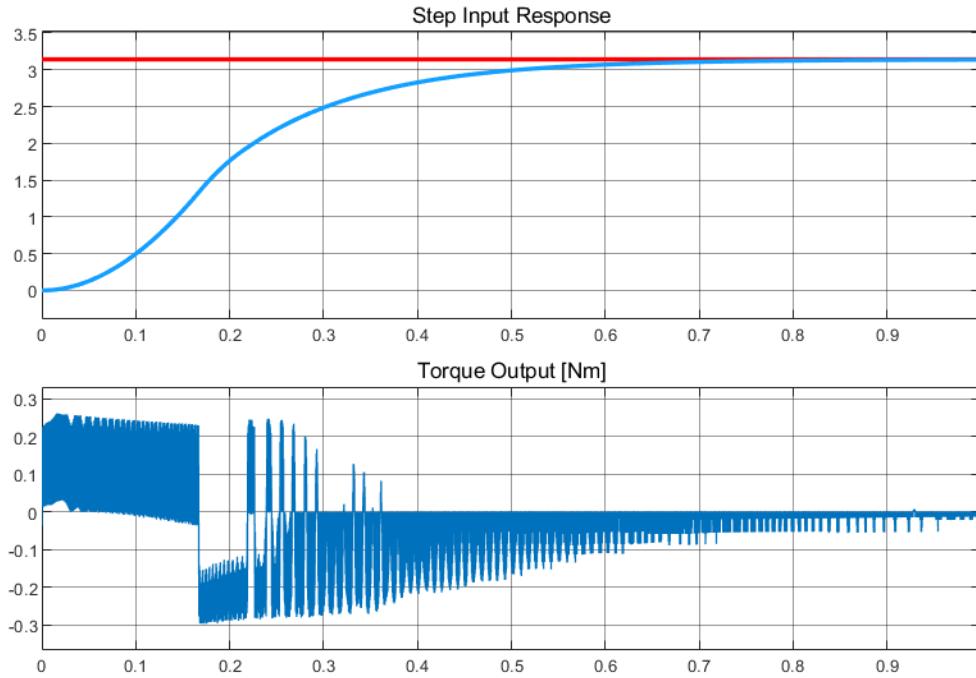
**Table 9 Command message data breakdown**

Byte	Name	Description
0	Message ID	Identify that this is a <b>Command</b> message
1	Controller ID	Unique ID for each motor 32 max devices in RS-485
2	Position	Rotations: Bits [7-4] <ul style="list-style-type: none"><li>● Bit 7 - Sign Bit</li><li>● Bits [6-4] - Rotations, up to 7</li></ul> Angle MSB: [3-0]
3	Position	Angle LSB [7-0]
4	Torque	Range +14 N-m to 14 N-m with ~0.1 N-m per bit increment.

## 7. Analysis and Testing

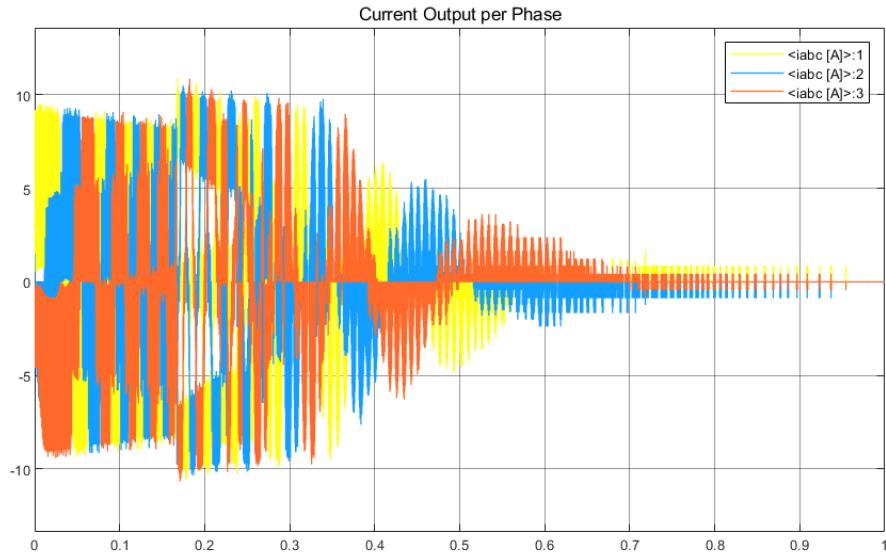
### 7.1. Motor Dynamics Modeling

A model of the actuator and motor controller was developed using MathWorks Simulink and modules from Simscape Electrical. This model allows for analysis of the mechanical dynamics of the systems as well as the electrical response of the motor controller. This model implements field oriented control as described in Section 4.1.3. This allows for very accurate simulations of motor response time and power requirements. The model also allows external loads and input commands to be varied.



**Figure 37: Step Input Response and Motor Torque Output**

Figure 35 shows the step response with the modeled inertial load at the elbow. The control loop gains are then tuned to minimize settling time and eliminate overshoot. The figure also shows the torque output of the motor. These graphs show the behavior of the outer control loop that reduces the commanded torque as the motor approaches its target, implementing a virtual spring.



**Figure 38: Motor Driver Current Output per Phase**

Figure 36 shows the current in each phase of the motor as supplied by the motor driver. The curves of the three phases are sinusoidal, a characteristic of field-oriented control. As the motor approaches its target

position, the overall magnitude of the current decreases. This magnitude is directly proportional to the torque produced by the motor. These currents are controlled by the inner control loop, which maximizes torque producing current.

## 7.2. *Arm Assembly Stress Calculations*

With design specifications solidifying, more complex calculations were performed to predict maximum loading and optimize component selection. These calculations were done in MATLAB, and source code can be found in Appendix A. The code's architecture is variable based, allowing for greater flexibility to experiment with changing materials, masses, distances etc.

Calculations for the arm's effective moment of inertia,  $I$ , were done using MATLAB, the raw code of which is shown in Appendix A. The moments of inertia tensor for extension/flexion, abduction/adduction and rotation of the shoulder is shown in Table 10.

**Table 10 Force and Moments of Inertia for Extension/Flexion or Abduction/Adduction at Shoulder**

Humeral Spanner Connection Point	Force (y-axis) (N)	Torque, (x-axis) (N-m)	Geared Motor Output (N-m)	Safety Factor
<b>Hand Mass 3.0 kg</b>	-51.0	28.9	42.0	1.5
<b>Hand Mass, 0.05 kg</b>	-22.1	7.6	42.0	5.5

The maximum supported weight of 3 kg at the end of the arm, assuming full extension for maximum bending moment throughout the arm, more than triples the functional moment of inertia for the system. This means that the arm itself is less significant than an object being manipulated.

Another component under high stress is the bevel gear. The bending and contact stresses for the bevel gear at the shoulder were analyzed as shown in Table 11.

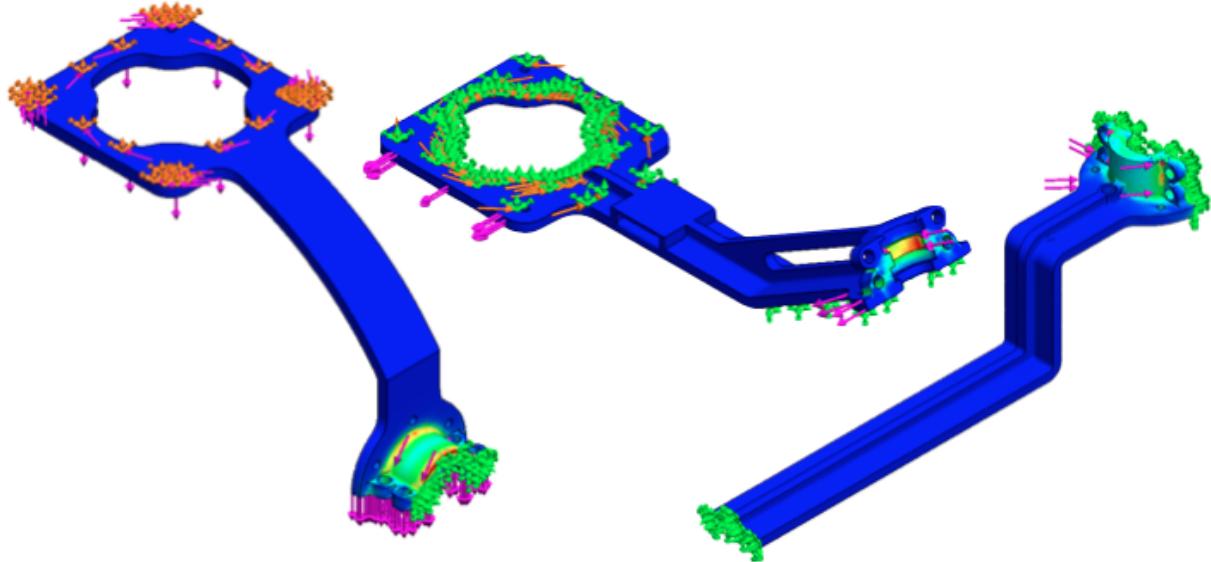
**Table 11 Bending and Contact Stresses for Shoulder Bevel Gears**

Shoulder Differential Housing	Gear, Contact Stress (MPa)	Allowable Gear, Contact Stress (MPa)	Safety Factor	Gear, Bending Stress (MPa)	Allowable Gear, Bending Stress (MPa)	Safety Factor
<b>Hand Mass 3.0 kg</b>	1.2	1000.0	854.7	60.3	150.0	2.5
<b>Hand Mass, 0.05 kg</b>	0.6	1000.0	1666.7	15.8	150.0	9.5

Between the safety factor for gear bending and contact stress at a million cycles, bending was the more significant. It was determined that steel rather than plastic gears were required in the shoulder.

### 7.3. *Exoskeleton Finite Element Analysis*

Finite element analysis (FEA) through Solidworks was done on the linkages in the exoskeleton, to validate the design and safety, because a person is meant to wear it. The goal was to ensure high factors of safety, and low maximum stresses and displacements, to ensure high-accuracy coupling through the haptic system and robotic arm. As shown in Figure 37 and Table 12, the goals are achieved; this is in part due to the fact that the bulk of the exoskeleton is supported by the fixed mounting point in the chair.



**Figure 39: Shoulder (left), upper arm (middle), and wrist (right) linkages in the exoskeleton**

**Table 12 FEA Analysis on Exoskeleton Members**

Linkage	Factor of Safety	Maximum Displacement (mm)	Maximum Stress (MPa)
Shoulder	23	0.006	1.5
Upper Arm	8	0.03	4.5
Wrist	16	0.01	2.2

## 8. Project Management

### 8.1. Ownership

The team identified six (6) technical challenges that will be encountered during the project. For each technical challenge, a primary and secondary owner were assigned. Table 13 shows the division of ownership for each challenge.

**Table 13 Division of Ownership**

Technical Challenge	Primary Owner	Secondary Owner
Selecting and Customizing Actuators and Transmission	Dominic	Nat
Designing and Building a Robotic Arm	Nat	Dominic
Designing and Building a Haptic Feedback System	Orion	Aayush
Designing and Building an Exoskeleton	Brian	Aayush
Designing a Motor Controller	Dominic	Orion
Tele-operating the Arm with the Exoskeleton	Orion	Aayush

The team has various experiences in the aerospace, robotics, medical devices, rapid prototyping, and consumer products industries. As such, there is a high level of confidence in the assigned subsystems and tasks, as well as the schedule.

## **8.2. Gantt Chart**

Appendix B displays the Gantt chart for each subsystem. In summary, preliminary design and initial round of purchasing was completed by mid-August, multiple design reviews occurred during September, a preliminary build was completed late October. Revisions, testing, and further purchasing occurred throughout the remainder of October, and the final test, build, and integration phase were completed in late November. The team is considering implementing a force sensitive haptic feedback system. However, this was a stretch goal due to the complexity of the other tasks associated with the system. There were multiple parallel paths built into the schedule, such that progress was not wholly dependent on only one task. This is done to reduce risk, and to allow time to experiment with other solutions for a given problem.

### 8.3. SWOT Analysis

SWOT Analysis	
Strengths	Weaknesses
<ul style="list-style-type: none"><li>• Diverse experience between group members.</li><li>• Good chemistry.</li><li>• Driven and motivated team members.</li><li>• Communication skills.<ul style="list-style-type: none"><li>◦ Prepared to ask for help.</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Balance of expertise.<ul style="list-style-type: none"><li>◦ We all share similar engineering strengths.</li></ul></li><li>• Scope tunneling.</li><li>• Scope creep.</li><li>• External commitments.</li><li>• Story problems.</li><li>• Lack of knowledge for FEA.</li></ul>
Opportunities	Threats
<ul style="list-style-type: none"><li>• Resources from our advisor Professor Whitney.</li><li>• Previous well documented research.</li><li>• Connections to other researchers through our advisor.</li><li>• Personal professional connections.<ul style="list-style-type: none"><li>◦ Markforged.</li><li>◦ Haddington Dynamics.</li></ul></li><li>• External makerspaces.</li><li>• Other professors at Northeastern.</li></ul>	<ul style="list-style-type: none"><li>• Group scheduling conflicts.</li><li>• Ordering timelines.</li><li>• Potential safety restrictions.<ul style="list-style-type: none"><li>◦ Limits on work time in lab spaces.</li></ul></li><li>• Hand group coordination.</li></ul>

**Figure 40: SWOT Analysis Diagram**

Figure 38 is a graphical representation of the team's strengths and weaknesses. The team was cognizant of external threats such as part lead times and scheduling conflicts; these issues were tackled early in the development cycle. The team has a variety of strengths and opportunities such as a diverse skill set and good chemistry. Resources provided by connections with the group's advisor as well as professional connections within the group became valuable as the project matured in the build and review phases.

### 8.4. Budget

A preliminary budget for the fall semester was put together based on the subsystems identified previously. Table 14 shows the top-level expected costs. This preliminary cost reflects prototyping, testing, and revision phases. The quantity and price of the T-Motor U8 Lite motors was the most expensive part of the project. Gearboxes and machined housings also contributed significantly to cost. The communications cost is accounted for as it is almost exclusively software and equipment already owned, such as computers and the motor controllers. Most structural parts of the Arm and Exoskeleton were 3D printed at no cost.

**Table 14 Budget**

Subsystem	Cost
Actuators	\$3,600.00
Robotic Arm	\$2,929.07
Haptic Feedback System	\$520.94
Exoskeleton	\$743.98
Motor Controls	\$1,212.00
<b>Total Expected Cost</b>	<b>\$8,485.05</b>
<b>Total Spent to Date</b>	<b>\$7,636.28</b>

## 9. Future Work

The work done on the Avatar Arm and Exoskeleton serves to provide a foundation for Northeastern University's student-run XPRIZE team. This team intends to compete in the international Avatar XPRIZE challenge. With this continuation of this project, a smooth handoff of design files, electrical schematics, and software repositories to the student group will be coordinated.

### 9.1. Mechanical

All mechanical designs were made in SolidWorks, a software available to all students at Northeastern University for free. Design files are stored in GrabCAD, a cloud management system. With GrabCAD, ownership of the design vault can be transferred to the appropriate parties in the student-run organization and can be managed accordingly going forward. In addition to the design files, this written report in conjunction with weekly memos will serve to convey the incremental learnings made by the group and justification behind design decisions.

The Exoskeleton could be improved to incorporate more degrees of freedom to match those of a human arm. One recommended addition would be a degree of freedom to allow wrist rotation. In addition, for full haptic feedback (including sensations on the wearer's skin, for example), the Exoskeleton would need to be fitted with a "haptic skin," or device which can provide localized feedback at various points on the user's skin with variable intensity vibrating nodes.

The manufacturability of the Arm could be improved. Bonding the gears to the shafts in the differential, to ensure proper torque transmission, was difficult. An alternative method for future assembly would be to use keyed shafts and gears. In addition, if power consumption and weight become constraints in a future system, the design will need to be made less modular.

## **9.2.      *Electrical***

Electrical designs were done in KiCAD, a free and open-source schematic capture and board layout software. This software is used by the team's advisor Prof. Whitney in his research. It is expected for this to remain the software of choice for the Avatar XPRIZE student design team. All of the capstone team schematics and layout files will be provided to the student group for further development as well as reference. The capstone team will also be providing helpful references.

The motor driver boards developed within this project meet the goals for communications, sensing, and driving motors. However, within the timeframe of capstone, only three of the nine boards were built and tested. There are also features of the board that could be improved, specifically the layout of the motor driver IC and filtering on the power regulator input. These aspects of design can be worked around to achieve operation, but reduce reliability. As this project transitions to a student group design team, there will need to be a review of the electrical design. A third revision of this design will ensure the best functionality in the full scale system.

In their current state, the Avatar Arm and Exoskeleton are ready to be cabled. The Exoskeleton has communications cabling done and a preliminary power cabling. The power cabling could be re-done to better distribute the current loading per wire. The Arm needs both communications and power cabling.

With these last steps, the Avatar Arm and Exoskeleton will be electrically operational. The team also expects the motor driver boards to be usable in other areas around the Avatar system. These boards can be used with any size brushless motor, allowing for customization based on torque, speed, and power requirements. This team encourages the Avatar XPRIZE student design team to build off the current prototype system. A majority of the necessary work has been done, but there are always improvements to be made.

## **9.3.      *Software***

Source code for both embedded firmware and high-level software is stored in GitHub. This allows for easy distribution and collaboration of the code.

Board level firmware was developed in Texas Instrument's free Code Composer software. This software makes programming for the Tiva microcontrollers straightforward and streamlined. Documentation on how to develop in this environment and upload code for testing will be provided to the Avatar XPRIZE student group.

Firmware was developed in C and C++, using Texas Instruments device drivers and Free-RTOS, an open-source task scheduling framework. Documentation for both of these dependencies is very clear with active forum communities that offer help and guidance. Software developed by the Capstone team includes message formats, encoder data processing, and motor control algorithms. This code has been commented clearly and written so that it will be readable to a new developer. This should reduce the time required to learn what the software is doing and allow for new developers to build off the code.

The control loops for each motor will need to be tuned accordingly to maximize response speed while remaining stable. This is dependent on the load at the joint from the structure of the arm. The code will

also need to optimized for loop rate and communication speed. Additional communications functionality could also be added and defined in the controller firmware.

Communication from the master device was written in Python for easier transfer to Robot Operating System (ROS). Every loop, the master device pings each slave by ID to get position and updates the denavit hartenberg rotational and displacement vectors as well as the Jacobian for inverse and forward kinematics. It filters these responses to only those with the proper format as a significant percentage of messages return broken bit strings. It was observed that slave devices farther from the master device suffered higher losses however at the 1Mbps baud rate, loss did not create significant latency issues.

The homogenous transform matrices were used to create a live simulation of the exoskeleton's locations using the vpython library. This live simulation is one step away from a fully functional Avatar system as the communication and kinematics are fully functional needing only finished motor controllers to complete the haptic circuit between exoskeleton and avatar arm.

## 10. Summary

The goal set out by the AAXP of transporting a human's presence and sense anywhere in the world was pared down to a five (5) DOF humanoid arm and haptic exoskeleton system. The scope was narrowed to target precision motion for applications like surgery and delicate work in hazardous or inaccessible areas. The designs for the Arm were based on the actuator technologies used in assembly and autonomous robots including the Stanford Doggo and Mini-Cheetah. The Arm is then fleshed out with a haptic interface based on previous interfaces developed for research and rehabilitation. Finally, the system is integrated with a user through the exoskeleton drawing from designs like the MIT Hermes and University of Texas' Harmony [35].

The team has successfully designed and built a robotic arm and exoskeleton which can operate with the range similar to a human arm. The Exoskeleton was comfortable to wear, and the encoders on each joint were able to accurately record angular displacements of each joint. These angular displacements were shown in a simulation of the exoskeleton, to prove viability. The Arm demonstrated precise movement in all 5DOF. Custom electronics were assembled, integrated, and tested to specifications. Forward and inverse kinematics of the arm and exoskeleton were calculated to map positions across systems, and to simulate the movement of each system in 3D-space.

The system will eventually be measured on strength, precision, transparency, and workable area. In the future, the motor controller will be refined to provide mirrored movements and feedback across both the Arm and Exoskeleton. In addition, the Exoskeleton will be fitted with a localized haptic feedback system. These tasks, and integration into an entire avatar system, will occur through the Robotics Collaborative at Northeastern University.

## **11. Intellectual Property**

### ***11.1. Description of Problem***

The All Nippon Airlines (ANA) Avatar XPRIZE competition challenges groups to be “focused on the development of an Avatar System that will transport a human’s sense, actions, and presence to a remote location in real time, leading to a more connected world.” Potential applications for the Avatar system include emergency response for disaster relief and surgery, remote education facilities, or extreme-environment exploration.

As a capstone project, due to limited time and money, the scope was reduced from an entire Avatar system to a humanoid arm, excluding the hand. This project consists of five main parts:

- A humanoid, robotic arm with five (5) degrees of freedom (DOF), encompassing three (3) DOF of a shoulder, one (1) DOF of an elbow, and one (1) DOF for wrist rotation.
- A custom-designed motor controller for managing each motor’s actions.
- A haptic feedback system that relays tactile information from the robotic arm to the user.
- A tele-operation communication method for communicating between the robotic arm and the user.
- An exoskeleton suit, worn by the user, that acts as a controller for the robotic arm and a carrier of the haptic feedback system.

### ***11.2. Proof of Concept***

Initial designs for the Avatar Arm, exoskeleton, and haptic feedback systems address some of the challenges posed by the project. The Arm has been designed with a transmission scheme and actuator layout, with easy of manufacturability in mind. The exoskeleton has been designed with a similar direct-drive transmission scheme, as well as low-weight and friction; there are few heavy, moving parts overall in the exoskeleton. The CAD for both the Arm and Exoskeleton generally shows the designs will have the range of motion desired. The physical builds of these systems meets the internal requirements set by the team.

### ***11.3. Progress to Date***

Several iterations of CAD models of the Arm and Exoskeleton have been completed. Several design reviews have taken place, including one with Markforged employees (who previously worked at Rethink Robotics). In addition, there have been 2 revisions for the custom motor controller, and a design review focused on the controller.

The Exoskeleton and Arm have been completely built and mounted on their frames. The software for the motor controller boards is currently being developed; a motor has been demonstrated to turn and record angular displacements to a computer.

#### **11.4. Individual Contributions**

The team identified six (6) technical challenges that will be encountered during the project. For each technical challenge, a primary and secondary owner were assigned. Table 15 shows the division of ownership for each challenge. In addition to what is noted, during the summer, each team member spent time researching theories and technologies related to their subsystem.

**Table 15 Division of Ownership and Work Accomplished to Date**

Technical Challenge	Primary Owner	Secondary Owner	Work Done
Selecting and Customizing Actuators and Transmission	Dominic	Nat	Actuators and transmission have been designed and implemented into the Exoskeleton and Arm.
Designing and Building a Robotic Arm	Nat	Dominic	Final design complete: features 2DOF differential design and 1DOF rocker assembly
Designing and Building a Haptic Feedback System	Orion	Aayush	Sketches for viable implementations; sensors have been ordered
Designing and Building an Exoskeleton	Brian	Aayush	Final design complete: features ball-and-socket shoulder design and adjustable forearm handle
Designing a Motor Controller	Dominic	Orion	Second revision of the board has been tested
Tele-operating the Arm with the Exoskeleton	Orion	Aayush	Motors are functioning and encoders are recording angular displacements

#### **11.5. Future Work**

The work done on the Avatar Arm and Exoskeleton serves to provide a foundation for Northeastern University's student-run XPRIZE team. This team intends to compete in the international Avatar XPRIZE challenge. With this continuation of this project, a smooth handoff of design files, electrical schematics, and software repositories to the student group will be coordinated.

## 12. References

- [1] XPRIZE, "ANA Avatar XPRIZE Competition Guidelines 2.0"  
<https://www.xprize.org/prizes/Avatar/guidelines>, Last Accessed July 15, 2019
- [2] N. Hamilton, W. Weimar, and K. Luttgens, "Kinesiology: Scientific Basis of Human Motion. New York, NY: McGraw-Hill, 2012.
- [3] Morrison, William, MD, "Understanding the Normal Shoulder Range of Motion," Healthline.com [Online]. Available: <https://www.healthline.com/health/shoulder-range-of-motion#3> Last Accessed August 6, 2019
- [4] Gupta, Abhishek and O'Malley, Marcia, 2006, "Design of a haptic Arm exoskeleton for Training and Rehabilitation citation," IEEE/ASME Transactions on Mechatronics, Vol. 11 No. 3 pp 280-289
- [5] "Gimbal Lock", [https://en.wikipedia.org/wiki/Gimbal\\_lock](https://en.wikipedia.org/wiki/Gimbal_lock), Last Accessed August 7, 2019
- [6] Tachi, Susumu, February 2016, "Telexistence Enabling Humans to Be Virtually Ubiquitous," IEEE Computer Graphics and Applications, Vol. 36, Issue 1, pp 8-14.
- [7] Marchese, Andrew, Hoyt, Hubbard, April 29th, 2010, "Force Sensing and Haptic Feedback For Robotic Telesurgery," B.S. Thesis, Worcester Polytechnic Institute
- [8] Mori, Giulio, Paterno, Fabio and Santoro, Carmen, June 1st, 2018, "Towards Understanding the Usability of Vibrotactile Support for Indoor Orientation," Castiglione della Pescaia, Italy.
- [9] Park, K., "Haptic gloves and surgical robot systems," U. S. Patent 10,292,780, May 21, 2019
- [10] Rubin, J., et al., "Haptic Feedback Glove," U. S. Patent 0,335,842, November 22, 2018
- [11] Black, G., Osman, S., "Gloves that include haptic feedback for use with HMD systems," U. S. Patent 10,275,025, April 30, 2019
- [12] Pirasmeppulkul, S., et al., "Haptic Glove as a Wearable Force Feedback User Interface," U. S. Patent 0,322,629, November 9, 2017
- [13] Tezuka, Mayuko et al, February 2016, "Presentation of Various Tactile Sensations Using Micro Needle Electrotactile Display," PLOS ONE DOI 10.1371
- [14] Mevey, J., 2009, "Sensorless Field Oriented Control Of Brushless Permanent Magnet Synchronous Motors", M.S. Thesis, Kansas State University, Manhattan, KS.
- [15] Jani, Yashvant, 2011, "3-Phase Motor Control: Renesas' Floating Point RX MCU Series Advantage Over Fixed Point", <https://www.digikey.com.au/en/articles/techzone/2011/oct/3phase-motor-control-renesas-floating-point-rx-mcu-series-advantage-over-fixed-point>, Last Accessed, August 11, 2019.
- [16] Kim, B., Deshpande, A., 2017, "An upper-body rehabilitation exoskeleton Harmony with an anatomical shoulder mechanism: Design, modeling, control, and performance evaluation", The International Journal of Robotics Research, Vol. 36, Issue 4.

- [17] Wang, A., Ramos, J., Mayo, J., Ubellacker, W., Cheung, J., and Kim, S., 2015, "The HERMES humanoid system: A platform for full-body teleoperation with balance feedback," RAS 15th International Conference on Humanoid Robots (Humanoids) IEEE 2015, pp. 1-2.
- [18] MIT Biomimetics Robotics Lab, "MIT HERMES Project," <https://biomimetics.mit.edu/research/mit-hermes-project>, Last Accessed August 6, 2019
- [19] Katz, B., 2018, "A Low Cost Modular Actuator for Dynamic Robots", S.M. Thesis, Massachusetts Institute of Technology, Cambridge, MA.
- [20] Kau, N., Schultz, A., Ferrante N., Slade, P., 2019, "Stanford Doggo: An Open-Source, Quasi-Direct-Drive Quadruped", Proceedings, IEEE International Conference on Robotics and Automation 2019, Montreal, Canada.
- [21] Vincent, James., "Blue the robot could be the AI-powered workhorse of the future"  
<https://www.theverge.com/2019/4/9/18295029/robot-ai-blue-uc-berkeley-cheap-safe-humans>, Last Accessed August 10, 2019
- [22] Piccoli, M. and Lim, M., 2014 "Cogging Torque Ripple Minimization via Position Based Characterization", Proceedings, Robotics: Science and Systems X, Berkeley, CA.
- [23] Universal Robots, "UR5 Robot Arm", <https://www.universal-robots.com/products/ur5-robot/>, Last Accessed August 3, 2019
- [24] Precise Automation, "PF3400 SCARA Robot", <http://preciseautomation.com/PF3400.html>, Last Accessed August 2, 2019
- [25] 유승열전인액조영화, "Apparatus and Method for Collision Detection for Collaborative Robot," South Korea Patent KR101844542B1, March 4, 2016
- [26] Intuitive Surgical, "About da Vinci Systems" <https://www.davincisurgery.com/da-vinci-systems/about-da-vinci-systems>, Last Accessed August 2, 2019
- [27] Lenihan, J., et al., 2008 "What is the Learning Curve for Robotic Assisted Gynecologic Surgery?" Journal of Minimally Invasive Gynecology, Vol. 15, pp. 589-594
- [28] Prisco, G., Rosa, D., "Robotic surgical system with joint motion controller adapted to reduce instrument tip vibrations," U. S. Patent 7,865,269, January 4, 2011
- [29] FARO Technologies , "FARO Case Studies" <https://www.faro.com/products/3d-manufacturing/farogage/case-studies/>, Last Accessed August 2, 2019
- [30] Ramos, J., Wang, A., and Kim, S., 2015, "A Balance Feedback Human Machine Interface for humanoid teleoperation in dynamic tasks," Intelligent Robots and Systems (IROS) IEEE 2015, pp. 1-2.
- [31] Budynas, R. G., Nisbett, K. J., 2011, "Shigley's Mechanical Engineering Design", 9th Ed., McGraw-Hill, New York, NY, pp. 1015-1020

[32] Weigl, O., “ODrive - High performance motor control”, <https://hackaday.io/project/11583-odrive-high-performance-motor-control>, Last Accessed August 9, 2019

[33] Gokhale, V., Chaudhuri, S., Dabeer, O., 2015, “HoIP: A point-to-point haptic data communication protocol and its evaluation,” 2015 Twenty First National Conference on Communications (NCC)

[34] Kurt M. DeGoede, James A. Ashton-Miller, Jimmy M. Liao, Neil B. Alexander, How Quickly Can Healthy Adults Move Their Hands to Intercept an Approaching Object? Age and Gender Effects, The Journals of Gerontology: Series A, Volume 56, Issue 9, 1 September 2001, Pages M584–M588,

## 13. Appendix A: MATLAB Script

MAIN

---

```
%% Main function, runs all components.
clear
% Global Variables
global g
g = 9.81;
% System Variables
systemMass = 0;
systemCM = [0 0 0];
backlash = 0;
spannerDim = [35 29 120]/1000;
% Functions
hand = hand();
radiiSpanner = spanner(spannerDim, hand.Fout, hand.Tout, hand.I0out);
elbowDifferential = diffHousing(radiiSpanner.Fout, radiiSpanner.Tout, radiiSpanner.I0out);
elbowGear = gear("steel", 10, radiiSpanner.Tout(3)/2);
humeralSpanner = spanner(spannerDim, elbowDifferential.Fout, elbowDifferential.Tout, elbowDifferential.I0out);
shoulderDifferential = diffHousing(humeralSpanner.Fout, humeralSpanner.Tout, humeralSpanner.I0out);
shoulderGear = gear("steel", 10, humeralSpanner.Tout(3)/2);
% Display
disp("Hand Force")
disp("Net Force & Torque [X Y Z]")
disp(hand.Fout)
disp(hand.Tout)
disp("Radii Spanner ")
disp("Net Force & Torque [X Y Z]")
disp(radiiSpanner.Fout)
disp(radiiSpanner.Tout)
disp("Elbow Differential ")
disp("Net Force & Torque [X Y Z]")
disp(elbowDifferential.Fout)
disp(elbowDifferential.Tout)
% disp("Elbow Gear ")
% disp(elbowGear)
disp("Humeral Spanner ")
disp("Net Force & Torque [X Y Z]")
disp(humeralSpanner.Fout)
disp(humeralSpanner.Tout)
disp("Shoulder Differential ")
disp("Net Force & Torque [X Y Z]")
disp(shoulderDifferential.Fout)
disp(shoulderDifferential.Tout)
% disp("Shoulder Gear ")
% disp(shoulderGear)
```

HAND

---

```
function obj = hand()
    global g;
    obj.mat    = matLib("PLA");
    obj.geom   = "sphere";
    obj.dim    = 50/1000; %(m)
    obj.vol    = calcVol(obj.geom, obj.dim);
    obj.mass   = 8000/1000; %(kg)
    obj.ICM    = momentOfInertia(obj.geom, obj.mass, obj.dim);
    obj.CM     = [obj.dim 0 0];
    obj.Fout   = [0 g*obj.mass 0];
    obj.Tout   = [0 0 obj.Fout(2)*obj.dim];
    obj.I0out  = parallelAxis(obj.ICM, obj.mass, obj.CM);
end
```

SPANNER

---

```
function obj = spanner(dim, Fin, Tin, Iin)
    global g;
    % Establish Spanner Characteristics
    obj.mat    = matLib("Carbon Fiber Torsion");
    obj.E      = str2double(obj.mat{3,3});
    obj.Sy     = str2double(obj.mat{3,6});
    obj.geom   = "cylinder shell";
    obj.dim    = dim; %(m)
    obj.vol    = calcVol(obj.geom, obj.dim); % cm^3
    obj.mass   = (str2double(obj.mat{3,2})*obj.vol)/1000; %(kg)
    obj.CM     = [obj.dim(3) 0 0];
    % Calculate Isolated and End Moment of Inertia
    obj.ICM    = momentOfInertia(obj.geom, obj.mass, obj.dim);
    obj.I0     = parallelAxis(obj.ICM, obj.mass, obj.CM);
    obj.I0out  = Iin + obj.I0;
    % Calculate Isolated Component Forces/Torques
    obj.F      = [0 g*obj.mass 0];
    obj.T      = [0 0 obj.F(2)*obj.dim(3)/2];
    % Calculate End Forces
    obj.Fout   = Fin + obj.F;
    obj.Tout   = [0 0 dim(3)*Fin(2)] + obj.T + Tin;
    % Calculate Maximum Deflection, Bending, Torsion and Stresses
    obj.deflection = deflection("cantilever", Fin, obj.dim(1), obj.E, obj.ICM);
    obj.sigBendMax = ( max(obj.Tout)*dim(1)/2 ) / ( pi * dim(1)^4 / 32 );
    obj.tauTorsMax = 0;
    obj.sigX = obj.sigBendMax;
    obj.sigY = 0;
    obj.sigZ = 0;
    obj.sigAvg = (obj.sigX + obj.sigY)/2;
    obj.R = sqrt(((obj.sigX - obj.sigY)/2)^2 + obj.tauTorsMax^2);
    obj.sigP1 = obj.sigAvg - obj.R;
    obj.sigP2 = obj.sigAvg + obj.R;
    obj.sigP3 = 0;
    obj.sigP = [obj.sigP1 obj.sigP2 obj.sigP1];
    obj.n = obj.Sy / (max(obj.sigP) - min(obj.sigP));
end
```

## DIFFERENTIAL HOUSING

---

```

function obj = diffHousing(Fin, Tin, Iin)
    global g;
    disp(Fin)

    %% Left Motor
    motorLObj.geom = "cylinder shell";
    motorLObj.dim = [85.45 25.16 28]/1000; % OD ID Thickness in m
    motorLObj.mass = 245/1000; %kg
    motorLObj.ICM = momentOfInertia(motorLObj.geom, motorLObj.mass, motorLObj.dim);
    motorLObj.CM = [50 0 35] / 1000;
    motorLObj.I0 = parallelAxis(motorLObj.ICM, motorLObj.mass, motorLObj.CM);
    motorLObj.F = [0 motorLObj.mass * g 0];
    motorLObj.T = [motorLObj.CM(3) * motorLObj.F(2) 0 motorLObj.CM(1) * motorLObj.F(2)];
    %% Right Motor
    motorRObj.geom = "cylinder shell";
    motorRObj.dim = [85.45 25.16 28]/1000; % OD ID Thickness in m
    motorRObj.mass = 245/1000; %kg
    motorRObj.ICM = momentOfInertia(motorRObj.geom, motorRObj.mass, motorRObj.dim);
    motorRObj.CM = [50 0 -35] / 1000;
    motorRObj.I0 = parallelAxis(motorRObj.ICM, motorRObj.mass, motorRObj.CM);
    motorRObj.F = [0 motorRObj.mass * g 0];
    motorRObj.T = [motorRObj.CM(3) * motorRObj.F(2) 0 motorRObj.CM(1) * motorRObj.F(2)];
    %% Motor housing
    motorBoxObj.mat = matLib("PLA");
    motorBoxObj.geom = "box shell";
    motorBoxObj.dim = [107 100 201, 6.5] / 1000; %(m)
    motorBoxObj.vol = calcVol(motorBoxObj.geom, motorBoxObj.dim); % cm^3
    motorBoxObj.infill = 0.50; % percent
    motorBoxObj.mass = str2double(motorBoxObj.mat{3,2}) * motorBoxObj.vol * motorBoxObj.infill; % (kg)
    motorBoxObj.ICM = momentOfInertia(motorBoxObj.geom, motorBoxObj.mass, motorBoxObj.dim);
    motorBoxObj.CM = [100.5 0 0]/1000;
    motorBoxObj.I0 = parallelAxis(motorBoxObj.ICM, motorBoxObj.mass, motorBoxObj.CM);
    motorBoxObj.F = [0 motorBoxObj.mass * g 0];
    motorBoxObj.T = [0 0 motorBoxObj.CM(1) * motorBoxObj.F(2)];
    %% Misc Motor Components
    motorCompObj.mass = 1500/1000;
    motorCompObj.ICM = momentOfInertia("sphere", motorCompObj.mass, 50/1000);
    motorCompObj.CM = [100.5 0 0]/1000;
    motorCompObj.I0 = parallelAxis(motorCompObj.ICM, motorCompObj.mass, motorCompObj.CM);
    motorCompObj.F = [0 motorCompObj.mass * g 0];
    motorCompObj.T = [0 0 motorCompObj.CM(1) * motorCompObj.F(2)];
    %% Sum All Components
    obj.totalMass = motorLObj.mass + motorRObj.mass + motorBoxObj.mass + motorCompObj.mass;
    obj.totalI0 = motorLObj.I0 + motorRObj.I0 + motorBoxObj.I0 + motorCompObj.I0;
    obj.totalF = motorLObj.F + motorRObj.F + motorBoxObj.F + motorCompObj.F;
    obj.totalT = motorLObj.T + motorRObj.T + motorBoxObj.T + motorCompObj.T;
    obj.I0out = obj.totalF + Iin;
    obj.Fout = obj.totalF + Fin;
    obj.Tout = obj.totalT + [0 0 Fin(2) * 201 / 1000] + Tin;
end

```

DIFFERENTIAL SHAFT CANTILEVER

GEAR CONTACT AND BENDING STRESS

```

% Input requires material, diameter of the gear (mm), rot per sec, Force in and
% Torque in
function obj = gear(mat, n, Tin)
    %% Establish Variables/Constants
    % Using all S.I. units (ISO Symbols)
    obj.Ze = 1; % Elastic coefficient sqrt((N/mm^2))
    obj.Wt = 1; % INPUT FORCE
    obj.b = 1; % Net face width (mm)
    obj.d = 1; % Outer pitch diameter (mm)
    obj.ZI = 1; % Geometry factor for pitting resistance
    obj.KA = 1; % Overload factor
    obj.KV = 1; % Dynamic factor (calc from pitch line velocity
    obj.KHB = 1; % Load distribution factor
    obj.Zx = 1; % Size factor for pitting resistance
    obj.Zxc = 1; % Crowning factor for pitting resistance
    obj.ZNT = 1; % Stress cycle factor for pitting resistance

```

```

obj.Zw = 1; % Hardness ratio factor for pitting resistance
obj.SH = 1; % Contact safety factor
obj.Kth = 1; % Temperature factor
obj.Zz = 1; % Reliability factor for pitting
obj.met = 1; % Outer transverse module (mm)
obj.Yx = 1; % Size factor for bending strength
obj.Yb = 1; % Lengthwise curvature factor for bending strength
obj.YJ = 1; % Geometry factor for bending strength
obj.YNT = 1; % Stress cycle factor for bending strength
obj.Sf = 1; % Bending factor of safety
obj.Yz = 1; % Reliability factor for bending strength
obj.Qv = 7; % Quality factor
obj.T = Tin; % Torque at input shaft
obj.n = n; % RPS
obj.d = 43; % Maximal diameter of the bevel gear
obj.b = 10; % Length of a single tooth
obj.Wt = 2 * Tin / ( obj.d );
%% Calculate Variables
% Overload factor assumes medium shock, medium shock
obj.KA = 1.75;
% Dynamic factor is calculated from the pitch line velocity.
% Assume Qv = 7
B = 0.25 * ( 12 - obj.Qv )^( 2 / 3 );
A = 50 + 56 * ( 1 - B );
%obj.vt = pi * obj.d / 1000 * n / 60;
obj.vt = 5.236e-5 * obj.d * obj.n;
obj.KV = ( ( A + sqrt( 200 * obj.vt ) ) / A )^B;
% Size factor for Pitting resistance (Zx)
obj.Zx = 0.00492 * obj.b + 0.4375;
% Size factor for bending (assumed met < 1.60 mm
obj.Yx = 0.5;
% Load distribution factor (KHB), both members straddle mounted Kmb = 1
obj.KHB = 1 + 5.6 * ( 10^-6 ) * obj.b^2;
% Determine limiting bending and contact stresses based off materials
if mat == "steel" % Assuming grade II through hardened steel
obj.sigHlim = 1000; %MPa
obj.sigFlim = 150; %MPa
elseif mat == "plastic" % Guessing at plastic har
obj.sigHlim = 50;
obj.sigFlim
end
%% Calculate
% Fundamental Contact Stress Equation
obj.sigH = obj.Ze * sqrt( ( 1000 * obj.Wt ) / ( obj.b * obj.d * obj.ZI ) * ( obj.KA * obj.KV * obj.KHB * obj.Zx * obj.Zxc ) );
% Allowable Contact Stress Number Equation
obj.sigHP = ( obj.sigHlim * obj.ZNT * obj.Zw ) / ( obj.SH * obj.Kth * obj.Zz );
% Bending Stress
obj.sigF = ( 1000 * obj.Wt / obj.b ) * ( obj.KA * obj.KV / obj.met ) * ( obj.Yx * obj.KHB ) / ( obj.Yb * obj.YJ );
% Permissible Bending Stress Equaton
obj.sigFP = ( obj.sigFlim * obj.YNT ) / ( obj.Sf * obj.Kth * obj.Yz );
% Crowning factor for pitting. Assume properly crowned teeth
obj.Zxc = 1.5;
% Pitting resistance Geometry Factor (ZI). Number of teeth, N = 20 for
% both gears.
obj.ZI = 0.0625;
% Bending strength geometry factor (Yj). Number of teeth, N = 20
obj.YJ = 0.20;
% Stress cycle factor for pitting resistance (ZNT). NL = 10^4
obj.ZNT = 3.4822 * ( 10^4 )^-0.0602;
% Stress cycle factor for bending strength YNT. NL = 10^4
obj.YNT = 1.683 * ( 10^4 )^-0.0323;
end

```

## MATERIAL LIBRARY/LOOKUP

---

```
% matLib is a lookup function that returns material properties in a
% specialized format used by the rest of the code. It links to
% "matProperties.csv" where all properties are stored
% ALL QUERRIES MUST USE "", NOT "
function matArray = matLib(matQuery)
    matArray = "Material not Found"; % Default answers;
    %% Material Properties:
    matTable = readtable('matProperties.csv');
    matList = table2cell(matTable);
    if matQuery == "list"          %A user can type in list to see all material names
        matArray = {matList{3:end,1}};
        return
    end

    %% Search Through Material Properties
    matListSize = size(matList); %Finds array size in [Y,X] format
    for i = 3:matListSize(1) %Cycles through each row
        if matList{i,1} == matQuery
            matArray = { }; %Build Array
            matList{1,:}; %Returns titles
            matList{2,:}; %Returns units
            matList{i,:}; %Returns index for specific name at i
        };
        break
    end
end %End for loop
```

## CALCULATE MOMENT OF INERTIA

---

```
function I = momentOfInertia(geom, mass, dim)
    %%
    % Sphere           dim = radius
    % Cylinder         dim = radius, length
    % Hollow Cylinder dim = OD, ID, length
    % Box              dim = w, h, d
    %%
    if geom == "sphere"
        Ix      = 2/5 * mass * dim^2;
        Iy      = 2/5 * mass * dim^2;
        Iz      = 2/5 * mass * dim^2;
    elseif geom == "cylinder"
        Ix      = 1/2 * mass * dim(1)^2;
        Iy      = 1/12 * mass * (3*dim(1)^2 + dim(2)^2);
        Iz      = 1/12 * mass * (3*dim(1)^2 + dim(2)^2);
    elseif geom == "cylinder shell"
        Ix      = 1/2 * mass * (dim(1)^2 + dim(2)^2);
        Iy      = 1/12 * mass * (3*(dim(1)^2+dim(2)^2) + dim(3)^2);
        Iz      = 1/12 * mass * (3*(dim(1)^2+dim(2)^2) + dim(3)^2);
    elseif geom == "box"
        Ix = 1/12 * mass * (dim(2)^2 + dim(3)^2);
        Iy = 1/12 * mass * (dim(1)^2 + dim(3)^2);
        Iz = 1/12 * mass * (dim(1)^2 + dim(2)^2);
    elseif geom == "box shell"
        Ix = 1/12 * mass * (dim(2)^2 + dim(3)^2) - 1/12 * mass * ((dim(2)-dim(4))^2 + (dim(3)-dim(4))^2);
        Iy = 1/12 * mass * (dim(1)^2 + dim(3)^2) - 1/12 * mass * ((dim(1)-dim(4))^2 + (dim(3)-dim(4))^2);
        Iz = 1/12 * mass * (dim(1)^2 + dim(2)^2) - 1/12 * mass * ((dim(1)-dim(4))^2 + (dim(2)-dim(4))^2);
    else
        I = "Geometry Input not found";
    end
    return
end
```

```

%I      = ["Ix" "Iy" "Iz";
%       Ix  Iy  Iz;];
I = [Ix Iy Iz];
end

```

## PARALLEL AXIS THEOREM

---

```

%% Input is Icm[Ix Iy Iz], mass, CM[x y z]
function I0 = parallelAxis(Icm, mass, CM)
    Ix = Icm(1);
    Iy = Icm(2) + mass*CM(1)^2;
    Iz = Icm(3) + mass*CM(1)^2;
    I0 = [Ix Iy Iz];
    % for i = 1:length(Icm)
    % I0(i) = Icm(i) + mass*CM(i)^2;
    % end
end

```

## DEFLECTION

---

```

function deflection = deflection(geom, P, L, E, I)
    % Base equations:
    if geom == "cantilever"
        defX = P(1)*L^3 / (3*E*I(1));
        defY = P(2)*L^3 / (3*E*I(2));
        defZ = P(3)*L^3 / (3*E*I(3));
        end
        deflection = [defX defY defZ];
    end

```

## CALCULATE VOLUME

---

```

function V = calcVol(geom, dim)
    %%
    % Does what it suggests: it calculates the volume
    % sphere           dim = radius
    % cylinder         dim = radius, length
    % cylinder shell  dim = OD, ID, length
    % box              dim = w, h, d
    % box shell        dim = w, h, d, thickness
    %%
    if geom == "sphere"
        V = 4/3 * pi * dim^3;
    elseif geom == "cylinder"
        V = pi * dim(1)^2 * dim(2);
    elseif geom == "cylinder shell"
        V = pi * (dim(1)^2 - dim(2)^2) * dim(3);
    elseif geom == "box"
        V = dim(1) * dim(2) * dim(3);
    elseif geom == "box shell"
        V = dim(1) * dim(2) * dim(3) - ((dim(1)-dim(4)) * (dim(2)-dim(4)) * (dim(3)-dim(4)));
    else
        V = "Geometry Input not found";
    end
end

```

## **14. Appendix B: Gantt Chart**

