HERO GLOVE

Haptic Exoskeletal Robot Operator

WPI

A Major Qualifying Project
Submitted to the Faculty of Worcester Polytechnic Institute
in partial fulfillment of the requirements for
the Degree of Bachelors of Science in
Robotics Engineering
Computer Science
Mechanical Engineering
Electrical & Computer Engineering
By:

_____   _____   _____   _____
     Alexander   Tri Khuu   Peerapat   Saraj
     Caracappa   Luxsuwong   Pirasmepulkul

Project Advisors:
Professor Cagdas Onal
Professor William Michalson

Date: April 28th, 2016

Abstract

Non-repetitive manipulation tasks that are easy for humans to perform are difficult for autonomous robots to execute. The Haptic Exoskeletal Robot Operator (HERO) Glove is a system designed for users to remotely control robot manipulators whilst providing sensory feedback to the user. This realistic haptic feedback is achieved through the use of toroidal air-filled actuators that stiffen up around the user's fingers. Tactile sensor data is sent from the robot to the HERO Glove, where it is used to vary the pressure in the toroidal actuators to simulate the sense of touch. Curvature sensors and inertial measurement units are used to capture the glove's pose to control the robot.

Acknowledgements

Table of Contents

Table of Figure

Executive Summary

  Increased development in robotics technology has allowed for more capable robotic systems to perform more dexterous tasks in remote conditions that may be too dangerous for humans. Such tasks often include object grasping and manipulation, as can be seen in industrial assembly lines, space exploration, search and rescue systems, and military applications where robot manipulators can be used for hazardous material handling. An example of such application can be seen in the DARPA robotics challenge, where humanoid robots such as Boston Dynamic's Atlas robot were programmed to perform unstructured tasks which require the dexterous use of their manipulator such as driving a vehicle, opening a door, and closing a valve near a leaking pipe. However, if these non-repetitive manipulation tasks are imposed with a time constraint and require high-level decision making due to unpredictability, then the process would still be arguably faster if completed with humans.

  To allow for the control of remote systems to perform unstructured and non-repetitive tasks, robot teleoperation is still more suitable over fully autonomous robots as it allows for the human user to be in direct control over the system. Examples of such teleoperation devices include radio control systems and joystick-style gimble. Teleoperation allows human operators to properly interact with the slave robot and manipulate objects located in remote environments. However, this control strategy may not be accurate enough for certain applications as it relies on open loop control, where the only feedback that the human operator receives is visual feedback. Humans are able to perform highly dexterous tasks in minimal time due to the combination of many sensory feedbacks, including force feedback. Therefore, the absence of force perception in teleoperation systems results in an increase of task completion time due to reduced feedback which the human users can use as clues to gauge their input to the environment that they are interacting with.

  Similar to teleoperation, telepresence allows for increased information being sensed in the environment to be communicated back to the user in a way sufficient enough for the user to feel as if they are physically present at the remote site. This allows users to make input adjustments through closed-loop control from artificial sensory feedback. Haptic feedback is an example of such feedback, which recreates the sense of touch by applying mechanical forces, vibration, or motions to the user based on sensory readings from the remote robotic system. One current application of haptic feedback is in surgical robotic applications because this mechanical feedback allows surgeons to perform high risk tasks with better precision. However, these ultra-precise medical systems are application specific and therefore are very expensive and, thus making them not applicable to commercially available robotic manipulators.

  Realizing the need to fill this gap between commercially available open-loop robot teleoperation devices and precise, purpose-built haptic telepresence systems, we then sought to design this system in which we named the haptic exoskeletal robot operator, or HERO Glove. The goal for the HERO Glove project is to create a low-cost robot teleoperation system that provides realistic haptic feedback to deliver an intuitive user experience for the control of commercially available robotic manipulators. To accomplish this goal, we laid out three main objectives.

1. The HERO Glove system must provide an intuitive user interface for the control of a remotely located robot, or telerobot.
2. The HERO Glove system must provide realistic haptic feedback to the user for enhanced telepresence experience.
3. The HERO Glove system must be modular enough for use with robotic manipulators available on the market.

In order to accomplish these systems objectives, we used a systems engineering approach to follow a methodological process for designing, integration, and testing of the HERO Glove. We started by defining the needs description of what the system should do after we have identified our stakeholders. Using the needs, we then established a set of functional system requirements that the system must do for validation and verification. After so were we then able to lay out the high level systems diagram and then delve into greater engineering design.

The HERO Glove design consists of two main functionalities. The first functionality is to capture the user's hand gesture and track their arm position to use in controlling the robot manipulator. To capture the user's hand gesture, finger curvature sensors were developed to accurately measure finger bending in one direction and map that as commands sent to the robot. Our design utilized deflection of light as the method of detecting bend as it provided fast response time and greater precision over resistive flex sensors of comparable price range. To track the user's arm position in three-dimensional space, we used three 9 degree-of-freedom inertial measurement units to produce roll, pitch, and yaw readings of the user's joint. This reading was then used to calculate the robot's end-effector's final position using vector addition, which was then sent as commands to the robot as well.

The second functionality is to provide haptic feedback to the user using the force exerted on the robot's end-effector as sensory data. To provide accurate, realistic feedback, we used soft robotics technology to create toroidal shaped pneumatic actuators that inflate up and thereby providing resistance on the user's finger joints. In creating this resistance in the user's joints using air filled toroid, we were able to artificially simulate the sensation of grabbing a non-deformable object in the user's hand. In order to replicate the sense of touch on the robot's finger tips and palm, air-filled silicone domes were developed to inflate up when the robot touches an object. Using seven miniature solenoid valves, we were able to control them independently using pulse width modulation to achieve seven points of haptic feedback for both touch and force feedback.

To establish a versatile yet reliable communication protocol to transmit commands to control the telerobot as well as receive force readings from its end-effector, the open-source robot operating system was used as the software framework. A MinnowBoard MAX computer running an Intel Atom processor served as the main hub of communication by being run as a robot operating system core. The telerobots that we used to test our system on included a Jaco Arm in VRep simulaton and a Robotiq three-finger robotic end effector, which was the same gripper used on WPI's Warner Atlas Robot from Boston Dynamics. To provide force and tactile feedback, we developed a Robot Side Module that relays force reading from an array of tactile sensors to the MinnowBoard MAX, which is then transmitted wirelessly to the HERO Glove.

To test the functionality of our system, we analyzed each aspect to assess whether or not it has achieved the metrics as was described in the functional system requirements for validation and verification purposes. We then developed a set of user performance evaluations in which we invited students external to our project to test our system's operability requirements and allow

for us to evaluate qualitatively our system performance. Based on our quantitative results and user testing evaluations, we were able to conclude that our system was able to achieve the objectives set forth at the beginning of the project. First of all, the HERO Glove was able to accurately and precisely capture the user's hand gesture and arm pose enough to allow untrained users to successfully grab the objects scattered within the Jaco Arm's workspace in V-Rep simulation software. Next, the user was able to feel the haptic feedback provided by the glove well enough that they felt were able to control and maintain a grasp on the ball. Lastly, by using the robot operating system to run our system, the HERO Glove was able to communicate with and control various types of robotic manipulators, including the Jaco Arm in simulation and an actual Robotiq 3-Finger robotic end effector.

1 Introduction

Increased development in robotics hardware and software technology has allowed for more capable robotic systems to perform tasks in remote conditions that may be too dangerous for humans. Such tasks include object grasping and manipulation, as can be seen in industrial assembly lines, space exploration, search and rescue systems, and military applications where robot manipulators can be used for hazardous material handling. Traditionally, robot manipulators were found in industrial settings, where they perform repetitive, pre-programmed tasks to replace mundane human tasks with more accuracy and precision. Recent improvements in machine learning, object recognition, and motion planning algorithms now allows for robotic platforms to perform more human-like tasks with more autonomy, allowing robots to work in unknown environments without human supervision, assistance, and intervention.

Currently, there has been an increase in research for more intelligent autonomous robots to perform human-like tasks, as can be seen in the DARPA robotics challenge. In the DARPA robotics challenge, for example, humanoid robots such as Boston Dynamic's Atlas have to perform various tasks that require the dexterous use of their manipulator. Such tasks include driving a vehicle, removing debris blocking an entryway, opening a door, climbing up an industrial ladder, using a tool to break through a concrete panel, locate and close a valve near a leaking pipe, and connect a fire hose to a standpipe and turn on a valve (Dzieza, 2015). Such tasks may seem trivial for humans, but can take the first place winner, the Hubo robot from Korean Advanced Institute of Science and Technology, over 44 minutes to complete due to the robot having to process and learn about the information it receives. This significantly increased response and operational time required by an autonomous robot these non-repetitive tasks can affect the outcome of a mission significantly, for example, if for instance the robot must rescue humans in a burning house. Furthermore, if tasks require high dexterity and decision making, such as when disposing hazardous material or disarming a bomb, then the process would still be much faster if completed with humans.

To allow for the control of robotic systems in unstructured and non-repetitive tasks, robot teleoperation is still more suitable over fully autonomous robots as it allows for the human user to be in direct control over the system. Humans are using their sensory feedback and cognition to make decisions and perform dexterous manipulating tasks through a master robot, which sends the command to a slave robot that outputs the commands it receives onto its environment. This allows human operators to properly interact with the slave robot and telemanipulate objects located in remote environments. Examples of teleoperation devices include the radio transmitter or joystick gimble. However, this control strategy may not be precise enough for certain applications as it relies on open loop control, where the only feedback that the human operator receives is visual feedback. Humans are able to perform highly dexterous tasks in minimal time due to the combination of many sensory feedbacks. Therefore, the absence of force perception in teleoperated systems results in an increase of task completion time.

To increase the realism of the control system for the operator, the telepresence allows the information being sensed in the environment to be communicated back to the human user in a way sufficient enough to make the user feel them to be physically present at the remote site. This

information sent back to the operator allows them to make adjustments to their input to the system based on how well they are able to monitor their output, thus closing the control loop through this artificial sensory feedback. Haptic feedback is an example of mechanical feedback that recreates the sense of touch by applying forces, vibration, or motions to the operator based on the tactile or force sensor readings from the telerobotic system. This feedback can be simulated as vibration or resistance of the master device, among many other technologies. Haptic feedback is used extensively in surgical robotics applications because it allows surgeons to perform high risk tasks with better precision. However, these ultra-precise medical systems can cost upwards of 1.5 million dollars, and are not applicable to other commercially available robot manipulators. Additionally, commercially available teleoperators such as the joystick gimbal setup require training and expertise to use, as the motion required to operate the teleoperator may not intuitively reflect the actual movement of the telerobot.

The goal for this project is to create a low-cost robot teleoperation system that provides an intuitive user experience for the control of commercially available robotic manipulators. To accomplish this goal, three objectives were laid out. The first objective is that the master controller must provide an intuitive user interface for the control of the telerobot. To accomplish this, the range of motion of the master system must as closely as possible replicate the range of motion of the slave telerobot that is being controlled. Since the targeted telerobot systems are commercial robot manipulators, the design of the master controller will be in the form of a glove, drawing the analogy between the control of a human hand and that of a robotic manipulator. The second objective is that the system must provide haptic feedback to the user. To do so, non-invasive force and tactile sensors would be developed for the slave telerobot, and pneumatic force feedback actuator systems would be developed for the master glove device. The final objective is to make the system modular for easy installation and use with any commercially available robotic manipulators on the market. This is to help increase the uptake of haptic telepresence for operating more advanced robotic systems to perform dangerous tasks in settings that are too dangerous for humans.

2 Background

This section describes the engineering topics and literature review that is relevant to the understanding of the various design aspects of the HERO Glove.

2.1 Teleoperation

Teleoperation is the process of operating a vehicle or system over a distance using human intelligence, where the human operator is the person who monitors the operated machine and makes the needed control actions. This distance can vary from micromanipulation, which works with a scale of several centimeters, to space applications where the range spans millions of kilometers (Teleoperation and applications). The main function of teleoperation system is to assist the operator in performing and accomplishing complex and uncertain tasks in hazardous and less structured environments, such as space, nuclear plants, battlefield, surveillance, and underwater operations. A typical teleoperation system usually composes of two robot manipulators that are connected either mechanically or wirelessly in a way that allows the human operator to control one of the manipulators, called the master, to generate commands that map to the remote manipulator, called the slave. To increase understanding of teleoperation systems, some of the basic terms and concepts pertaining to such systems would be first explained, as defined by Cui et al.

The teleoperator is a machine that enables the human user or operator to sense, grasp, and mechanically manipulate objects from a distance. In general, any tool that extends a person's mechanical action beyond her reach is considered to be a teleoperator. A telerobot is a subclass of a teleoperator, which is defined as a "robot that accepts instructions from a distance, generally from a human operator and performs live actions at a distant environment through the use of sensors or other control mechanisms" (Cui et al., 2003). A telerobot usually has sensors and effectors for manipulation and mobility, including a mean for the human operator to communicate with both. Telemanipulation refers to when a slave robot arm or system, usually in a remote and/or dangerous environment, tracks the motion of the master manipulator or follows its command. Telemanipulation is divided into two strongly coupled processes; the interaction between the operator and the master device, and the interaction between the remote slave device and its environment (Cui et al., 2003).

A technology of teleoperation is mechanical manipulation, where the control commands can be transmitted hydraulically or mechanically to the teleoperator and the visual feedback can either be direct or through a monitor. Mechanical manipulation allows for good feedback. Another type of teleoperation is radio control, where commands are sent electrically by wire or through radio. An example of such system includes radio controlled cars and planes. Radio control is a type of closed loop, supervisory control system, which means that the operator mainly monitors and gives high-level commands to the teleoperator which can perform part of the tasks autonomously. Therefore, teleoperation can be further classified into closed loop control or direct teleoperation, and coordinated teleoperation. In closed loop control, the actuators of the teleoperator are controlled by the operator through direct analog signals who gets real time feedback. This system, however, is only possible when there is minimal delay in the control loop.

Telepresence, also referred to as tele-existence, is when sufficient amount of sensor information such as vision, sound, and force, is transferred from the operator site to the operator, so that he or she feels as physically present in the teleoperator site (Teleoperation and applications). Both teleoperation and telepresence involves a master and slave control system. In the radio controlled car example, the master is the radio module transmitting the control commands, and the slave is the car receiving the command and performing the actions as specified by the master. However, in order for a system to be called telepresence, a more sophisticated system is necessary. In perfect telepresence, for example, all human senses should be transmitted from the teleoperator site to the operator site. Most common senses that are transmitted are vision, hearing, and sense due to the wide availability of sensors and actuators on the market, as opposed to the more complicated sense of smell and taste. Cui et al., 2003 states that telepresence in robotics generally refers to a remotely controlled system that combines the use of computer vision, computer graphics, and virtual reality. Telepresence systems are usually viewed as composed of three parts; "a capture system to record and represent the information from the remote site, a network transmission system, and a display system to make the local user feel as if they were present in the remote scene".

2.1.1 Applications of Teleoperation

Applications where teleoperation is best suited for includes unstructured and non-repetitive tasks, where key portions of the task require dexterous manipulation, such as hand-eye coordination. It is also best suited for tasks where key portions of the task require object recognition or situational awareness. However, the needs of the display technology must not exceed the limitations of the communication link such as bandwidth and time delays. Teleoperation system tasks are distinguished by the continuous interaction between the human operators, teleoperator system, and the environment (Cui et al., 2003).

Space telerobots is a great application for teleoperation due to greater safety and lower costs when compared to using astronauts. Teleoperation allows for the supervision of human operators in situations where the technology for building intelligence space robots does not exist yet. A possible downfall of space teleoperation is the time delay required to transmit data over such great of a distance. An example of a space telerobot system is the Lunokhod 2, which operated for four months in 1973. Another example is the Space Shuttle manipulator on the Mars Pathfinder shuttle, which was controlled by two three degree of freedom (DOF) joysticks from within the shuttle itself (Cui et al).

Remotely operated vehicles (ROV) are one main application of teleoperation, where they are widely used to work in water conditions that are too deep or dangerous for divers. Examples of tasks performed include inspection and repairing of offshore oil platforms and pipelines, attaching cables to sunken ships to hoist them, and classified navy tasks (Cui et al). ROVs are tethered through an umbilical to the master control system. Advancements in manipulator technologies, for example, has allowed for more complicated teleoperated actuation tasks such as repairing underwater pipeline ball valves.

Toxic waste cleanup is an application of robot teleoperation that has a non- substitutable function (Cui et al). This is because the presence of radioactive materials and leakage, especially

in nuclear plants, make the environment too dangerous for humans that the task can only be accomplished through teleoperation of robots.

Another application of teleoperation is in military, where the unmanned ground vehicles (UGV) with manipulators would operate in foreign environments such as underwater, ground, air, and are controlled semi autonomously with the human controller closing the control loop. In such long distance tele-operation technology, the UGVs would be operated through the internet using cellular data signal (Quick, 2011).

The last example of robotic teleoperation application is in medical surgery, which allows the surgeon or specialists to perform tasks over distances. In robot assisted surgical systems, the ability of surgeons to perform minimally invasive procedures are enhanced in several ways. For example, by filtering high-frequency signals, unwanted surgical tremor can be eliminated Furthermore, the interface can allow for motion scaling in which easy-to- perform macroscopic movements at the surgeon console can be scaled down inside the patient, and thus enhancing accuracy (Okamura, 2005). A term commonly associated with feedback in medical robots is haptic feedback, which is a broad term used to describe both cutaneous (tactile) as well as kinesthetic (force) information (Okamura, 2005). Both types of feedback information are necessary to replicate the standard sensations that are felt with the human hand.

2.1.2 Available Teleoperation Systems

Joystick is a dominant method that has been widely used to teleoperate a variety of robotic platforms. A traditional joystick is an input device that consists of a stick pivoting on a base, additional buttons could be added to provide more input signals. Joystick is ideal for many applications because it is reliable, ergonomic and inexpensive.

Despite its many advantages, joystick is limited in certain applications. Specifically, joystick fails to provide intuitive operation in controlling multiple degrees of freedom system such as humanoid manipulators and therefore require a better method.

Robot manipulator is an assembly of joints and links, attached at the end of the assembly is an end-effector, which is a mechanism that can interact with the environment and perform tasks. Many robotic manipulators to an extent resemble the human hand with more or less degrees of freedom. Although most robotic manipulators are inspired by the human hand, the differences in configuration create an infinite workspace and thus require a variety of control implementations. Observing the limitations that existing controllers can provide and inspired by the concepts of haptic feedback, the HERO Glove was designed to act as a universal controller.

One robotic manipulator in particular was investigated to establish a proof of concept for the HERO Glove: The Jaco arm[1]. The Jaco arm was a prime choice for a proof of concept for this project due to the availability of two Jaco arms on WPI campus. Noteworthy features of the Jaco arm include its 6 DOF arm and its end effector: a robotic gripper featuring 2 under-articulated fingers and identical opposing thumb. After getting the HERO Glove to work

---

[1] Jaco Arm robotic manipulator http://www.kinovarobotics.com/

successfully with the JACO arm in a virtual environment, the HERO Glove was additionally made to work with the Robotiq Gripper[2] to demonstrate the ability of the HERO Glove to be used as a general controller interface with a variety of manipulators and grippers.

2.1.3 Problems with Teleoperation

Control in teleoperation system experiences time delay, which occurs in every electro-mechanical system (Cui et al., 2003). Even though in most cases, this delay is not noticeable, in some cases, it can result in the system being unstable. An ideal teleoperator would cause a telerobot to move in space, which would exactly match the motions of an operator. Then, all the forces imposed on the telerobot would be accurately reflected back to the operator. However, in reality, not only will the master have a much lower impedance than the slave, but the operator's muscle stiffness is low and the human hand has little mass compared to a large industrial manipulator. Furthermore, the physical separation of master and slave that can cover great distances is another factor that contributes to a delay (Cui et al., 2003).

In some cases of wireless teleoperation, such as in the long-distance teleoperated UGV presented by Quick, 2011, the data is sent over the internet. This may result in an even more pronounced time delay effect on the remote control, and is not negligible. If this issue is untreated, even small delays may lead to instability due to unwanted power generation in the communications (Cui et al., 2003). Even though it is known that there are always delays in a teleoperation loop as every part of the system has some delays, digital systems increase this delay. In systems with long delay teleoperation, there are no possibilities for closed loop control and due to the unpredictable nature of the mission, it is difficult to increase the autonomy. The problems that result in this latency, especially from wireless communication systems, include cognitive fatigue, communication dropouts, lag, and this increased difficulty in user experience may require too many people to run one robot, which increases cost (Cui et al., 2003).

2.2 Haptic Feedback

An effective way of providing feedback to a user interacting with a remote environment is through a haptic interface, which is defined as an interface through which a human operator may convey a sense of kinesthetic presence (Adams & Hannaford, 1999) and receive tactile feedback. Of the stated definition, there are two elements of interest: kinesthetic feedback and tactile feedback.

Kinesthetic feedback is defined, in its most general sense, to be the feedback of the physical state of a teleoperated slave module to the user of a master module. Whereas kinesthetic feedback is defined to be the flow of information from the slave module to the master, the HERO Glove is also concerned with the exact opposite: exerting kinesthetic presence, which is the flow of information from the master module to the slave. Examples of desired kinesthetic feedback physical quantities are the physical position, shape, and dimensions of an object that would be remotely interacted with via a robotic manipulator slave module. Examples of desired physical

---

[2] Robotiq 3-Finger robotic end effector http://robotiq.com/products/industrial-robot-hand/

quantities that need to be established are a kinesthetic presence of are the position, orientation, and pose of the robotic manipulator.

Tactile feedback is defined to be the feedback of physical characteristics of a teleoperated slave module's ability to interact directly with the environment to the user of a master module. Examples of such physical characteristics are the texture of a surface, ambient surrounding temperature, and pressure exerted on the surface of the slave module's surface. The primary characteristic of concern in regards to receiving tactile feedback is the pressure exerted on the slave module's pressure sensors.

The type of feedback that a user would receive from the HERO Glove, however, is force feedback. Force feedback is an emergent property of incorporating both kinesthetic and tactile elements into the system. Through the establishment of bilateral communications between the master and slave modules, the user of the master module may provide feedback from the slave module, in the form of physical resistance to motion and pressure where the slave is in contact with the environment, that accurately simulates what the user would feel in the slave module's place.

2.3 Soft Robotics

Most robotic systems are hard, in the sense that they are composed of hard structures, with joints based on conventional bearings. Though such systems are often well suited to the tasks and functions that they were designed for, there are also times when a "softer" approach would be more apt. Reasons for why such an approach would more suitable may range from those of safety to more functional rationales.

Soft robotics is the subfield of robotics that focuses on using soft, physically compliant materials to construct robotic systems. Applications of soft robotics often involve bioinspired designs and functions, but the application of soft robotics to the HERO Glove will be for more functional reasons: safety and intuitiveness of use. The techniques and tools used were adopted from those developed and used in WPI's Soft Robotics Lab.

2.3.1 Soft Actuation

Soft actuation is an area of research in soft robotics that encompasses the design, fabrication, and control of compliant actuators. An example of a soft actuator are the soft pneumatic actuators or SPA, which are fabricated with silicone using molding techniques to form chambers that could be filled with either compressible fluids such as air or incompressible fluids such as hydraulics. Depending on its design, the SPAs exhibit different shape deformations as the pressure inside the air chamber varies. The pressure in the actuators can be controlled via PWM-driven solenoid valve.

A technique to control the motion of SPAs include fiber reinforcements, as by reinforcing the walls of a tubular body, it is possible to mechanically program the actuator to perform a range of motions under fluidic pressurization (Polygerinos et al). Such motions include bending, extending, extend-twist, and bend-twist motions.

2.3.2 Soft Sensing

In a system with a prevalence of soft components, it is difficult to draw from the same set of sensors often used in hard robotic systems. In this case, less conventional means have to be used when it comes to matters of the measurement of state variables, such as joint angles. In soft robotic systems, such as the HERO Glove's toroidal pneumatic actuators, for instance, the curvature of a soft link would be analogous to a joint angle between two hard links. An example of such curvature sensor includes the magnetic curvature sensor which was developed by WPI's Soft Robotics Laboratory (Luo 2015). The magnetic curvature sensor is comprised of two components: a magnet and a Hall- Effect sensor (HES). The HES detects changes in magnetic flux to deduce the change in curvature in the soft link the sensor system is embedded in. Other alternatives to the magnetic curvature sensor includes the resistive flex sensor, which was used in a previous WPI MQP (Erad, Hedlund, Wells, & Wiegman, 2015). Though both the magnetic curvature sensor and piezoresistive flex sensor had proofs of concept, the independence of curvature readings from the silicone the sensor was embedded in made it the more desirable option. Soft sensing is required in soft robotic systems to use as feedback to more accurately control the state as the system may not behave as expected due to compliance.

Another form of sensing in soft robotics can also refer to the development and addition of compliant tactile sensors to increase feedback from the robot's interaction with its environment. This is to mimic how the human body has sensors all over the body to measure force and pressure exerted upon the skin. These soft sensors can be added onto the robot's structure, such as the end effector, that would most likely come in contact with the environment. As Dr. Eduardo Torres-Jara explains, "current industrial machine arms that are very precise are dangerous when they come in contact with objects in their path because they are likely to break themselves or the obstacle because robots are not sensitive to contact" (Fagella 2013). This integration of sensor technology can create more powerful, or more versatile robots.

2.3.3 Why Soft Robotics is a Viable Solution

Though alternative solutions exist, they were deemed inappropriate for the task assigned to the HERO Glove either due to safety reasons, or because they were deemed to be less effective in practice. For instance, the soft glove fingers of the HERO Glove were favored over hard systems such as those of the Rutgers Master II-ND glove (Bouzit, Burdea, Popescu, & Boian, 2002) and the RML Glove (MA & Ben-Tzvi, 2015), due to the application of soft robotics being considered safer. The compliance of the materials used in soft robotics and compressibility of the air used by the pneumatic system were deemed safer due to the system's innate higher tolerance for user variability of input. Whereas a hard system could easily injure a user resisting the system's feedback mechanism, a soft system such as the HERO Glove would pose a much lower risk to the user in the same scenario.

2.4 Gap Analysis

The purpose of the gap analysis section was to research into what current state of technology is available that we can work with and assess what still needs to be invented to help us accomplish our systems objective.

2.4.1 Finger Joint Measurement

      The first objective, or future state that the project must be once it is completed, is that the glove must be able to measure the finger joint angle as controlled by the user wearing the glove, and map that to a command for controlling the slave robot to the same position. The current state of technology is that there exists a curvature sensor developed by WPI's Soft Robotics Lab that can be used to measure the joint angle of the finger as curvature. One curvature sensor constitutes a flexible printed circuit board that contains both a Hall Effect sensor chip and a small one-eighth-inch-squared neodymium magnet mounted in close proximity to each other. The curvature is measured as the change in voltage from the change in magnetic flux as measured by the Hall Effect sensor (HES). The advantage of using this technology over resistive or strain-based flex sensors is that flex sensors often produce hysteresis and therefore less accuracy.
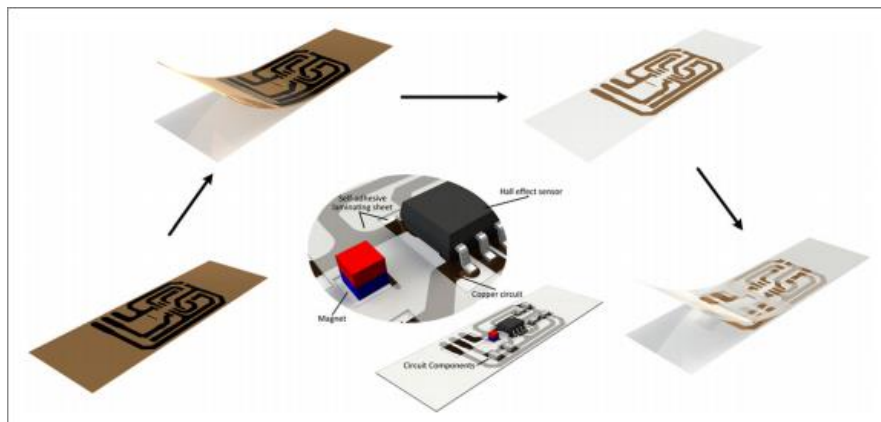
Figure 1: Design and fabrication of the curvature sensor (Luo et al, 2015).

      To bridge this gap between objective and current state of technology, the curvature sensor unit has to be designed and tested to have it fit on the back of the finger and mounted on the glove. Additionally, the placement of the magnet and HES relative to each other must be tested to find the optimal distance to most accurately measure the change in curvature. This accuracy could be tested by creating a test fixture with a potentiometer mounted on its joint or axis of rotation, and mounting the curvature sensor over the joint, to see how accurately the two measurements are to each other. This curvature sensor design has a much lower components cost compared to other resistive sensors; the HES and magnet costs less than two dollars in components while the cheapest resistive sensors that are available through Digikey[3] cost approximately ten dollars.

2.4.2 Hand Position Tracking

The second objective is to track the motion of the user's hand and use that trajectory to control the position of the slave end effector in real time. Through literature review, several journal articles proposed using an integrated sensor packet (ISP) that uses a 9 axis inertial measurement unit (IMU) that includes an accelerometer, a gyroscope, and a magnetometer, also known as a compass. The accelerometers from the IMU are used to track the gravity direction of the Earth,

---

[3] Digikey online electronics store http://www.digikey.com/

the magnetometers are used to track the North direction of the Earth's magnetic field and the gyroscopes are used to track the rotational velocity. Using sensor fusion algorithms, a quaternion output that represents the absolute orientation can be obtained. Several sources from Zhu et al, 2004 and Huyghe et al, 2009 have used methods such as Kalman filtering to accurately derive displacement from all 9 axes.
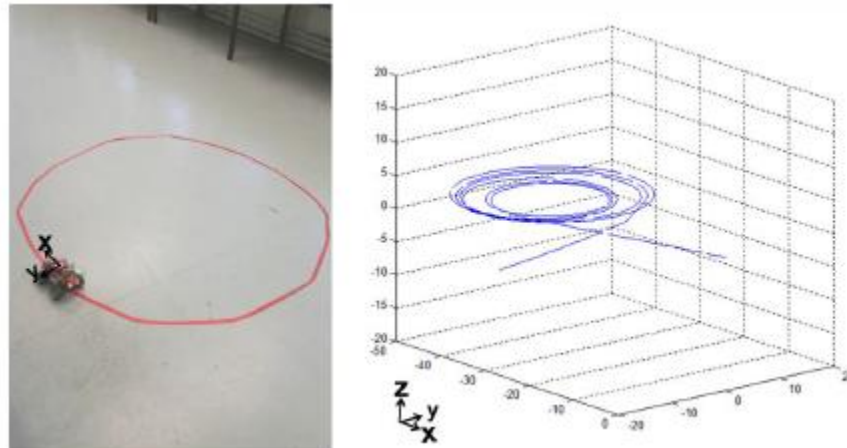
Figure 2: 9 axis IMU sensor readings were processed and displayed on MATLAB to track R/C car movement in 3D space (Reuveny et al).

The gap that needs to be bridged between this current state of technology and the objective lays in the algorithm that digital signal processing technique which would need to be designed to most accurately measure the 3D position. Even though the journal articles have shown through modeling that a 9 axis IMU and Kalman filtering techniques can provide enough sensor readings for use to measure 3D position, they do not explicitly show the actual code implementation, so that is one area that needs further research on. Secondly, 3 axis of the magneto sensor may become useless as the neodymium magnetic field may affect its readings. Therefore, greater testing has to be done to investigate the extent to which the neodymium affects the magneto sensor, and if it does, whether 6 axis reading from the accelerometer and gyroscope provide enough accuracy and precision.

2.4.3 Haptic Feedback

The third objective for the project is to provide realistic tactile and force feedback. The glove must be able to provide a sensation realistic enough for the user to feel the object that the robot is touching in the fingertips, finger joints, and palm. While some haptic feedback systems use vibration motors for feedback, the sensation provided by a vibration motor may not be realistic enough to replicate the robot grasping an object. A system that uses inflating pneumatic pockets for more realistic tactile feedback has been created by Rice University, called the Hands Omni glove.

Figure 3: Rice University's Hands-Omni Glove

http://www.coolwearable.com/hands-omni-glove/

Using the same principle of inflating and deflating pneumatic pockets for tactile feedback, a large pocket could also be created for the palm. Therefore, tactile feedback on the fingertips and palm could be achieved by creating an inflatable pneumatic pocket. The limitation of this type of tactile feedback is that it may not be able to provide accurate *force* feedback. An example of such force feedback is in replicating when a robotic gripper is grasping a non-deformable object and there is force exerting back on the joint, thus preventing the finger from closing any further. To do so, a system has to be created for the glove that can vary its stiffness or tension as the joint force on the slave robot increases. To do so, a pneumatic toroid can be created that fits around the finger that does not expand axially or radially, but instead straightens out as internal pressure increases. Several design, fabrication, and modeling tutorials have been published on the Soft Robotics Toolkit by Harvard University Wyss Institute that uses fiber-reinforced actuators. The expansion profile of these actuators change depending on the pattern of reinforced fiber (McAlpine).

Figure 4: Soft robotic glove using soft pneumatic actuators for rehabilitation

http://biodesign.seas.harvard.edu/soft-robotics

Similar to the system in Figure 4, a force feedback system can be created. Using soft pneumatic actuators provides an advantage over rigid exoskeletal gloves that may use spring-mass damper and motor for actuators because the latter system requires high level of precision and alignment with the user's joint knuckles to function properly. Furthermore, the cost of silicone is relatively inexpensive, with the main cost being the miniature solenoid valves, which cost 40 dollars apiece. The gap between the proposed force feedback design and the system illustrated in Figure 4 is to modify the design so that it most accurately mimics the hindrance to movement found physical robotic grippers. Furthermore, the final glove design must be able to incorporate the tactile sensors on the fingertip and palm with the force feedback on the finger joints, whilst minimizing bulkiness and maximizing portability for movement in free space.

2.4.4 HERO Glove to Robot Communication

The fourth objective or desired future state is to establish a low-error, safe, and low latency communication system between the master and slave system, which is the HERO Glove and the robot in this case. This bilateral control system must be non-invasive; that is, the system involving the HERO Glove and the robot must not require hacking or modifying of both the hardware and software of the robot itself. Ultimately, the master and slave system needs to be wirelessly communicating as that increases the flexibility of the system applications in remote environments. Furthermore, not only does the HERO Glove need to send joint angle measurements to the robot, but it also needs to receive fingertip and finger joint forces back as well.

The current state of the art is that robot communication could be established via the robot operating system or ROS[4]. ROS was developed specifically for Ubuntu[5] Linux distribution. Therefore, a microprocessor such as the Raspberry Pi[6] microcomputer, BeagleBone Black[7] platform, or the MinnowBoard MAX[8] computer is desirable over a microcontroller such as an Arduino due to its ability to run ROS. Running ROS on these microprocessor boards makes it more portable and streamlined than using a laptop computer, making it more suitable for wearable devices. ROS node on the BeagleBone Black can communicate to the ROS node on the robot by sending wireless packets through Wi-Fi. Further development requires running ROS on a microprocessor compatible enough with the peripherals required for the HERO Glove and how to effectively transmit and receive ROS packets to the robot with as little latency and as much safety as possible.

---

[4] Robot Operating System http://www.ros.org/
[5] Ubuntu open source hardware platform http://www.ubuntu.com/
[6] Raspberry Pi microcomputer https://www.raspberrypi.org/products/raspberry-pi-2-model-b/
[7] BeagleBone Black microprocessor https://beagleboard.org/black
[8] MinnowBoard MAX computer http://wiki.minnowboard.org/MinnowBoard_Wiki_Home

3 Methodology

This section details the entire design process of the HERO Glove following a systems engineering approach beginning with defining the systems objective, then requirements, high level design overview, and then the details of each product design aspect.

3.1 Objective

The Haptic Exoskeletal Robot Operator (HERO) Glove is a system that is used to teleoperate robot manipulators whilst providing realistic, accurate haptic feedback. Our system consists of two parts: the master glove and the slave module. The master glove achieves intuitive tactile and force feedback through use of soft pneumatic actuators at the fingertips and joints, that stiffen correspondingly to the interfaced manipulator's interactions with its environment. Accurate finger joint angle measurements are obtained through light-based curvature sensors. Positional control is obtained through the use of 3 9-DOF IMUs. A MinnowBoard MAX is used as the robot-side control module to allow for an easy add-on via USB to existing manipulators. The module receives the glove's 3D position, and passes it to the robot software for end effector pose control. Tactile sensors are attached to the manipulator. The module attached to the manipulator will then transmit sensor data from the tactile sensors back to glove. A concept of operation is having a user control a manipulator to open a door remotely.

3.2 System Requirements

This section describes the requirements of the systems that were created following the systems engineering approach. We begin by describing the needs, or what the system should be able to do. Next, we assess the "ilities" or characteristics that are most relevant to the system we are designing to help set the priorities of the system. Finally, we layout a list of functional system requirements, which is a list of requirements used to test the system, defined in a way that is measurable to help with the validation and verification of the system.

3.2.1 Needs Description

1. N01 – The system should provide accurate and real time teleoperational control of the robot manipulator.

2. N02- The system should provide realistic and real time visual, tactile, and force feedback from the robot to the glove.

3. N03- The system should be intuitive and easy to use for untrained personnel

4. N04-The system should be maintainable and expandable to increase wider uptake for greater commercialization of our system in the robotics industry.

5. N05- The system should be safe and durable, so that it could be used in a wide range of applications.

3.2.2 Acronym Description

·   F means functional requirements, which define specific behavior or functions.

·   NF means non-functional requirements, which is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors.

·   OR means operational requirements, which define the operational conditions or properties that are required for the system to operate or exist. Operational requirements include: human factors, ergonomics, availability, maintainability, reliability, and security.

·   C means cost and schedule constraints, which are the limits we put to the cost or expected delivery date of the system.

·   M means maintainability, which define how the system can potentially extend, grow, or scale during the lifetime of the system.

·   U means usability, which define the quality of the system, for example measurable effectiveness, efficiency, and satisfaction criteria.

3.2.3 Functional System Requirements Table

The following sets of requirements are stated in a concise way while being complete in conveying out the necessary. Functional Systems Requirements are stated to be cohesive and singular; each requirement does not try to include multiple requirements into one by using conjunctions such as "and". The requirements are technology independent and specify what the system must do, not how the system should do it. For example, the requirements state that the processing unit of the system needs 12 digital output pins, and not whether the system would be controlled via a BeagleBone Black computer or an Arduino Uno microcontroller. The requirement uses the terms "must" and "shall", and not "should", as the key word "should" belongs to system needs and not requirements. The requirements are also traceable to a stakeholder need for validation, and testable for verification to see whether the system was built right and whether the right system was built for the problem that has to be solved.

Ref. #	Title	Description	Validation (Traceability)	Priority	Verification (Testing)	Explanation
F01	Control Accuracy	The robot manipulator joint must be controlled within ±2 degrees error of the input glove finger joint angle.	N01	1	Testing, with reference system	These three requirements specify the accuracy that the control system from master to slave has to be, otherwise the system would not be as intuitive for the user to use if the user has to map a certain input to disproportionate output and have to keep compensating for error.
F02	Control Accuracy	The curvature sensor must measure the joint angle of each finger in the glove with less than ±2 degrees error of actual angle, up to 1000 times of use.	N01	1	Testing, with reference system	
F03	Control Accuracy	The slave robot end effector wrist rotation must be within ±2 degrees error of the glove rotation.	N01	1	Testing, with reference system	

F04	Velocity Limit	The system must have a velocity bandwidth between V_1 m/s and V_2 m/s in which the glove will use to control the robot manipulator.	N01	1	Testing	This velocity bandwidth is set so that the system can filter out unwanted fast jerks or drifts.
F05	Proportionality of Velocity	The robot's end effector velocity must be controlled such that it is proportional to that of the glove velocity.	N01	1	Testing	Proportionality is important to realistic control of glove, and not exceeds the speed limit of the robot arm.
F06	Force Sensing	The force sensors mounted on the robot manipulator shall measure the force exerted with at most 5% error.	N01	1	Testing, with reference system	We want to ensure enough sensitivity for accurate feedback
F07	Glove Position Tracking	The system must track the glove's position in free space within ±2 cm accuracy of the actual position.	N01	1	Testing, with reference system	This requirement is for controlling the end effector accurately to desired position
F08	SPA Control	Each SPA must incorporate a pressure sensor to monitor the internal pressure	N02	1	Test, to see if precise enough	Required to regulate pressure of SPA to turn on/off valve.
F10	SPA Control	The maximum pressure that the solenoid valve can supply must be at least 5 psi.	N02	1	Testing, using pressure sensor	The greater the pressure range, the higher the sensitivity from change in SPA size/stiffness
F11	SPA	The SPA must withstand at least 7 psi	N02	1	Testing, using	Human threshold is 7 psi, so no
	Design	pressure without rupturing.			pressure sensor	need to go higher than that
F12	SPA Design	The force feedback SPA on the finger joint must not expand axially or radially when inflated.	N02	1	Visual inspection	The SPA has to stiffen up, and not lose energy to changing its unnecessarily shape.
F13	Wireless Communication	The system must communicate wirelessly for a minimum range of 100 meters in open space.	N01	1	Field testing	Any closer and it is not really realistic for remote teleoperation.
F14	Wired Communication	The system shall support wired communication.	N01	2	Test to see if it works	Wired communication is faster and more accurate, therefore, it should always be an option.
F15	Software Design	The program shall adhere to Coding Conventions *Found Here* (to be further	N04	2	Inspection	System will be open source; therefore, it must be maintainable

		established) to improve readability				
F16	Software Design	The program shall follow objectoriented design.	N04	2	Agreement, Professor and team members	Object-oriented design is better for software maintenance and easier to program
F17	Processor Hardware Capability	The glove processing unit must be able to control a minimum of 11 solenoid valves	N02	1	MCU, MPU specs	Up to 11 SPA on the glove (5 fingertips, 5 finger joints, 1 palm)
F18	Processor Hardware Capability	The glove processing unit must be able to control the hold button		1	Field testing	Needs one more digital output pin
F19	Visual Feedback	The system must provide visual feedback of the manipulator from the slave robot to the user.	N01, N02, N03	1	Field testing, lab testing, real user experience	Visual feedback is effective in acquiring information of unknown environments

3.2.4 Ilities Assessment

Using a systems engineering approach, the top five "ilities" or characteristics for our system that are most relevant to our system are descibed and assessed. This is to further increase the transparency of our goal. The first top "ility" or characteristic for this project is **operability.** The reason for operability is because one of the needs of this system is to provide intuitive user experience for teleoperational control of the robot as an improvement over traditional joystick controllers or systems without haptic feedback. This is why the system was designed as glove that can move in free space in the first place. Therefore, the user of the system must be able to operate the system and control the robotic manipulator without as much training involved as other systems on the market. The importance of operability is reflected in the systems requirements for accurate controls of the robot manipulator as a mirror of the user's hand and arm gesture. Furthermore, the feedback mechanisms that constitute a major role of this system must enhance the user experience, making the process of operating a remotely located robotic manipulator to accomplish grasping and manipulation tasks more realistic. In designing this system for operability, one must constantly ask and verify "how intuitive to use is the system for new users?" and "how easy is it to operate the system?" If the system is not intuitive to use and the slave robot manipulator is not easy to operate, then the system does not meet its goal and must be redesigned.

The next top characteristic for this system is **compatibility.** The goal for this system is to be a universal robot controller that can work with multiple commercial robot manipulators on the market, as designing and creating a new robot manipulator is beyond the scope defined in this project. The system should work like a wireless mouse, where the dongle is an easy to use, non-invasive add-on to the computer USB ports and does not require any modification to the computer itself. Therefore, compatibility of this system with existing robot manipulators is an important pillar in the design. To make this system compatible with commercial robot

manipulators, the software program and electrical hardware design has to be well considered. A possible approach on the software aspect is to use ROS, as the communication protocol between the glove and existing robot manipulators that run ROS. ROS allows the system to send packets of information to each other. ROS is a feasible approach because many existing robotic platforms run it. Such systems include ABB, Adept, Motoman, Baxter, BeagleBoard, PR2, SummitXL by Robotnik, and Shadow Robot Hand. Tactile and force sensors would be developed to be easily attached to existing robot grippers. The tactile and force sensor that would be developed would be like a sticker, and will have servo connector-style 3-pin wires that can be easily attached to the microcontroller "dongle".

The third top characteristic for this system is **predictability.** For the system to be intuitive and user friendly, the user must be able to predict the outcome of their action. For example, the user must be able to trust that once they move the glove to the right at a certain velocity they can see that the robot manipulator that is synced to the system is moving in the same direction at a proportional velocity. Furthermore, the user must be able to trust that once the robot gripper comes in contact with an object, they can feel the feedback proportional to the force that the robot is actually exerting on the object. In order to make the system predictable, a wide range of feedback mechanisms are necessary for the user to be able to constantly monitor and adjust their output to the environment through the system. However, predictability must be verified through the users, therefore, the system must be tested with a wide demographic of user and listen to their feedback about whether the system output is predictable enough for their use.

The fourth top characteristic for this system is **demonstrability**, because the designer for the system must be able to convince the robotics industry that this system offers notable advantages over existing robot teleoperation system. During the MQP project presentations and hopefully Cornell Cup competition, the system has to be demonstrated in an effective way that fully conveys its purpose and advantages to the spectators. Otherwise, the spectators may not able to visualize the system performance as this system is very interactive to the user unlike autonomous robotic systems. Therefore, to incorporate demonstrability into this system for presentations, both physical and virtual robot manipulators should be present, and controlled right in front of the audience. Furthermore, the presentation should be interactive by allowing the audience to wear the glove and control a robot by themselves. Only then can the system demonstrate the feedback mechanisms which constitute a large portion of the engineering to the public.

The fifth top characteristic for this system is **reliability**. The application for this system involves interfacing with commercial robots to accomplish rigorous tasks, such as quickly opening a door for rescue. Therefore, the system must be reliable to deliver peace-of-mind to the users that the system would work until the task is successful. There are two main aspects of this system's reliability that the design engineers must consider and address. The first is mechanical reliability. This system uses a lot of soft, flexible silicone structures that would be inflated with air to increase adaptability and compliance to the user's fingers. These SPAs offer an advantage over rigid exoskeletal linkages as those linkages must perfectly line up with the user's joints and knuckles, otherwise they would not work effectively. However, SPAs are going to be developed

in the Soft Robotics Lab and not commercially available yet, as compared to rigid components such as bearings, springs, and ball joints. Therefore, SPAs can be unreliable as the soft material can rupture and leak if not designed well, and can thus deliver unreliable performance for user feedback. To verify the reliability of these SPAs, the system requirements have laid out the testing procedure that the system must pass before it can be delivered to the customer. The second aspect of the system's reliability is the communication reliability. Ultimately, this system must be able to communicate between the glove and the robot manipulator wireless. However, this communication protocol has to be reliable and account for errors that may occur when transmitting and receiving data because if this possible error is not accounted for, the user may lose control of robot manipulator. The hazard associated with losing connection is that the robot manipulator may damage its surrounding, or move in a way that may damage itself. Therefore, the wireless protocol must be designed both from the hardware and software aspect to transmit the data fast, and reliably. These aspects of reliability are addressed in the system requirements.

3.3 System Design Overview

The system has three main components: the HERO Glove, the MinnowBoard MAX and the robot being controlled. In order for the glove to be as compatible as possible, the entire software was built based on ROS due to its modularity and popularity in robotic applications.

At the high level view, the MinnowBoard MAX is the central of the communication network. The HERO Glove publishes the user arm and hand motion to the MinnowBoard MAX through WI-FI. The MinnowBoard MAX uses the data received to control the robot. Finger force sensing on the robot gripper can be obtained from the tactile sensors or the robot built-in sensors. If tactile sensors are used, the MinnowBoard MAX would get their values by communicating to an Arduino which is used to read the sensor through Bluetooth. If the robot provides built-in finger force sensing, the MinnowBoard MAX would subscribe to the ROS topic to get their values. The MinnowBoard MAX then publish them to the HERO Glove to provide feedback through the actuators.
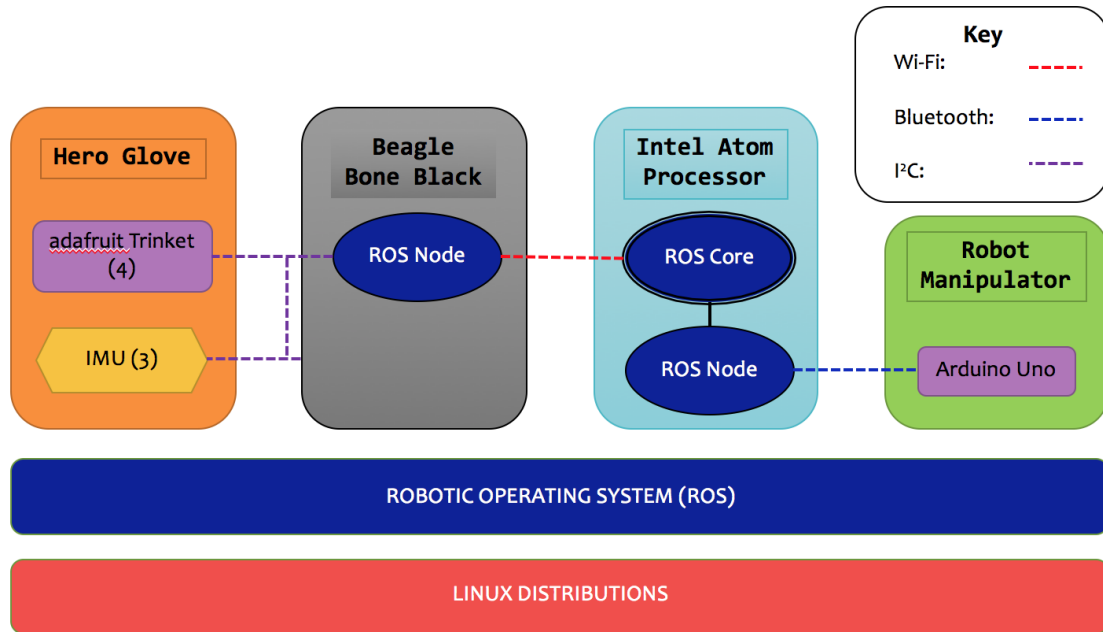
Figure 5: The HERO Glove Architecture

3.4 HERO Glove

The HERO Glove design consists of two main functionalities. The first functionality is to capture the user's hand gesture and track their arm position to use in controlling the robot manipulator. The second functionality is to provide haptic feedback to the user using the force exerted on the robot's end-effector as sensory data. To capture the user's hand gesture, finger curvature sensors were developed to accurately measure finger bending in one direction and map that as commands sent to the robot. Our design utilized deflection of light as the method of detecting bend as it provided fast response time and greater precision over resistive flex sensors of comparable price range. To track the user's arm position in three-dimensional space, we used three 9 degree-of-freedom inertial measurement units to produce roll, pitch, and yaw readings of the user's joint. This reading was then used to calculate the robot's end-effector's final position using vector addition, which was then sent as commands to the robot as well.

3.4.1 High Level Glove Assembly

The final top level design of the HERO Glove consisted of 5 main modules and an arm sleeve. Several views of the final top level assembly can be seen in Figure 6. Three fingers were used, as opposed to five fingers, because the team determined that most robotic grippers on the market were three fingered grippers. All modules were designed to be 3D printed, excluding the actuators themselves. By creating these modules, the HERO Glove was able to be slid on and off the user's hand easily and efficiently. Detailed descriptions of the HERO Glove modules will be outlined in the following sections. For exploded views of the top level assembly refer to Appendix J.
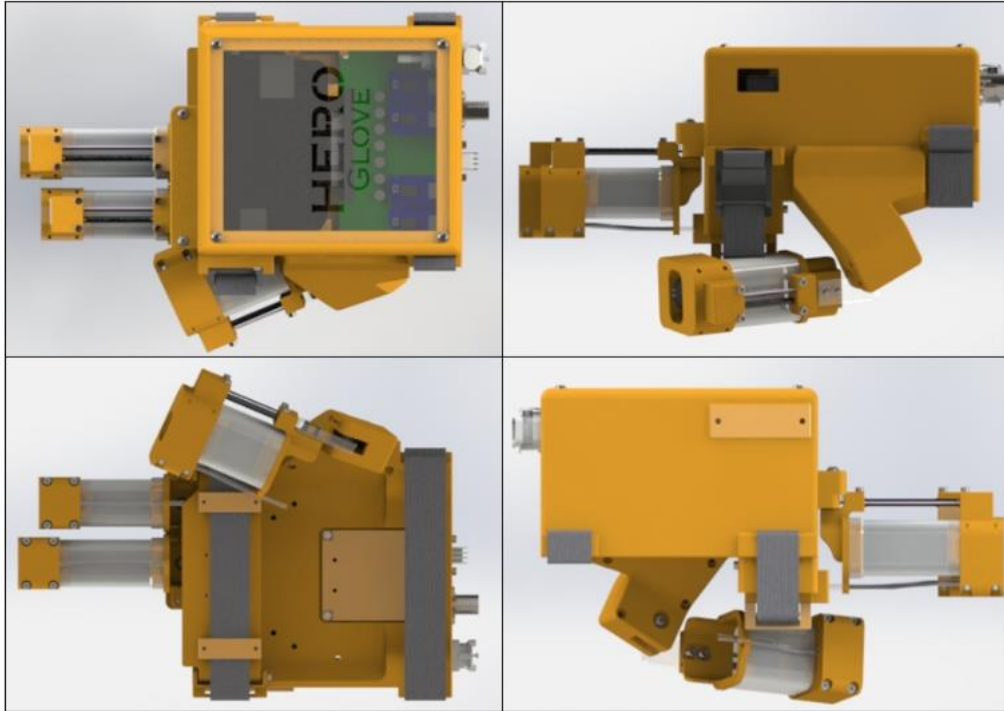
Figure 6: HERO Glove final CAD renderings
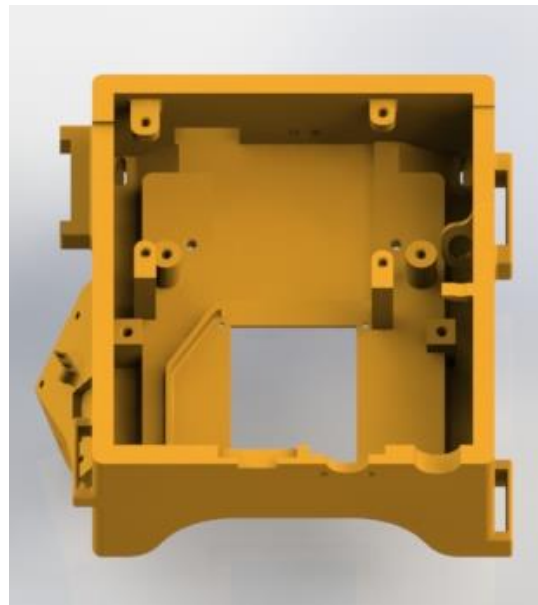
3.4.1.1 Glove Shell Module

The main module was a 3D printed shell. This shell, seen in Figure 7 housed the gloves electronics, solenoid valve manifold, and power and air connections to the glove. Because this component was designed for 3D printing, the design of the glove shell included built in mounting hardware for the gloves electronics. This included standoffs and mounting holes for the PCB motherboard, the BeagleBone Black Microprocessor, and the solenoid valve manifold. It also included a mounting flange for the pointer and middle finger modules and the thumb module. In addition, the glove shell was flanges for the palm module. These flanges allowed the palm module to slide in and out of the glove shell with the use of the strap. This adjustment made it possible to accommodate different user hand sizes and allow the palm touch actuators to make contact with the user's palm.
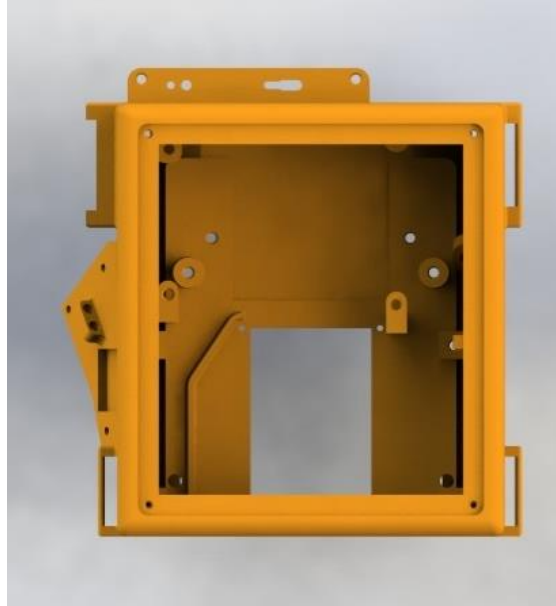
Figure 7: HERO Glove shell

In order to route the various cables and tubing needed for the HERO Glove's sensor and pneumatic actuators, various features allowed for a neat and organized glove design. As seen in Figure 8, two identical trenches span the inside of the glove on either side. The left trench allows for tubing and wires from the thumb flange to be neatly routed to the solenoid valves and the PCB motherboard. Similarly, on the right side, the trench allowed for the air input tube to be routed from the back wall of the shell.

Figure 8: Tubing trenches of glove shell

In order to mount the pointer and middle finger modules to the shell module a flange was designed on the front of the glove. Seen in Figure 9, mounting holes were incorporated into the flange to mount the finger modules. Through holes were also designed to allow for the tubing to

the finger actuators to route up and into the glove. The tubes and wires from the finger modules were able to enter the inside of the glove shell through several holes in the front wall of the shell. To allow for the easy removal of the finger modules barbed connectors were installed in to the two smaller holes on the front wall of the glove shell. On the inside of the glove shell the tubes were routed from the connector to the valve manifold. The outside exposed part of the connector was used to connect the finger joint actuators. Similarly, a connector was placed inside the glove shell just past the larger rectangular opening seen in the figure. This opening was made for the connector for the finger curvature sensors. This again allowed for the wires to be disconnected easily without having to re-route them through the glove shell. Finally, to cover the wires and tube connectors, a cover was designed to sit on the top of the finger flange.
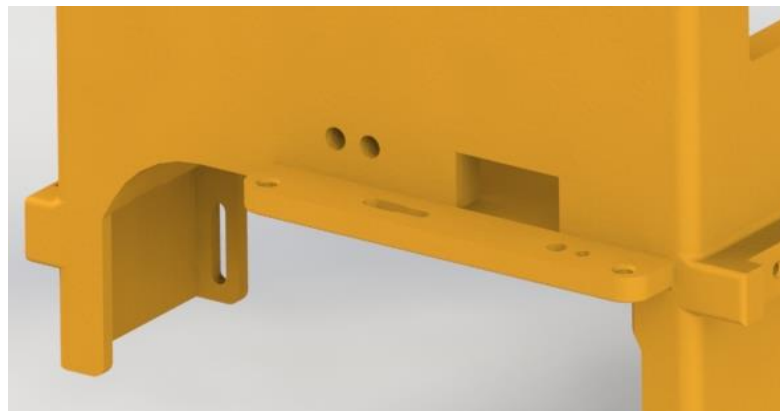
*Figure 9: Glove shell finger module mounting flange*

In order to mount the thumb module to the glove shell flange was designed to extend out of the side of the glove. The glove was designed for the user's right hand so the thumb flange extended out of the left wall of the glove shell. Seen in Figure 10, the flange was angled to match the anatomy of a human hand. To determine the proper dimension and angles of the flange, an estimated angle and length was determined using a ruler and protractor. The angle and length of the flange was then tested in an iterative process by printing the glove shell until the dimensions of the flange matched the team member's hands.

In order to route the wires for the thumb curvature sensor and the tubes for the thumb module's actuators holes were designed to allow access to the inside of the glove shell. To allow for the easy removal of the thumb actuators, a bracket was designed into the center of the thumb flange. This bracket was designed to hole two barbed connectors. The air input tubes to the actuators would then be connected to the barbed connectors and another tube would route from the connector into the glove shell. This made it possible to disconnect the thumb module without having to re-route the tube into the glove shell. Similarly, a cutout was designed to contain a connector for the four curvature sensor wires. This again allowed the thumb module to be removed without rerouting the wires into the glove shell. Finally, to cover the wire and tube connections, a cover was designed to match the angles of the flange. This cover ensured that the wires and tubes would not get pullout of their connectors when the HERO Glove was in use.
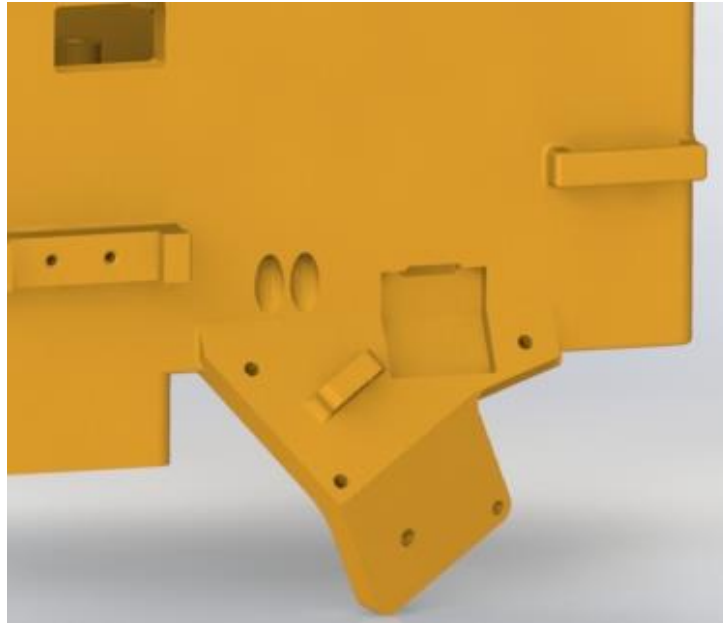
Figure 10: Glove shell thumb module mounting flange

To easily access one of the HERO Glove's inertial measurement units (IMU) an opening was created in the bottom of the glove shell. As seen in Figure 11, the opening allowed for the IMU to be mounted in the glove by using another 3D printed mounting block. This block was then screwed into the shell bottom.
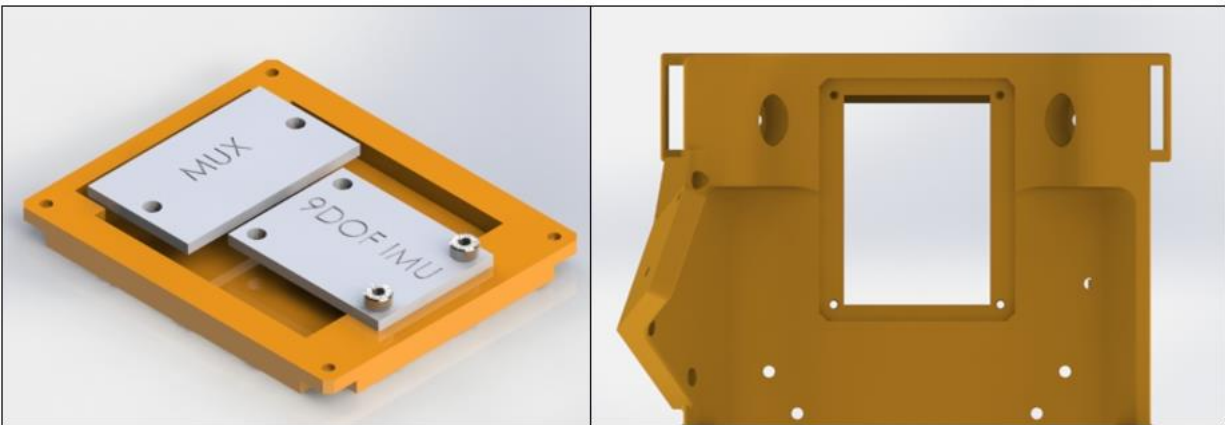
Figure 11: HERO Glove IMU mounting plate (left) IMU mounting hole (right)

The other various features of the glove shell included mounting surfaces for the glove's straps, openings in the shell walls to access the BeagleBone Black's Ethernet port and the Wi-Fi dongle, and opening in the back of the shell for the quick connect air connector, BNC connector for the glove's power, and a connector to plug the HERO Glove's arm sleeve into the main glove. Finally, a mounting surface was designed to allow for an acrylic cover to be installed to protect the glove's electronics.

3.4.1.2 Palm Module

A palm module was created in order to house the palm touch actuator. The final design of this module can be seen in Figure 12. This module was designed to have two a top and bottom component that could be fixed together using screws. The purpose of having a top and bottom piece was to allow for the mounting of the palm touch actuator. It also allowed for a cavity inside the module to route the tubing not only for the palm touch actuators, but for the incoming tubes from the point and middle finger tip actuators. These tubes would be connected to barbed connectors on the front of the module. Those barbed connectors were inserted into another tube that routed up and into the HERO Glove to the valve manifold. This allowed for the easy connection and disconnection of the finger modules. The palm touch actuator was designed to have three air input points. These three points were all connected into one input tube using barbed connectors.
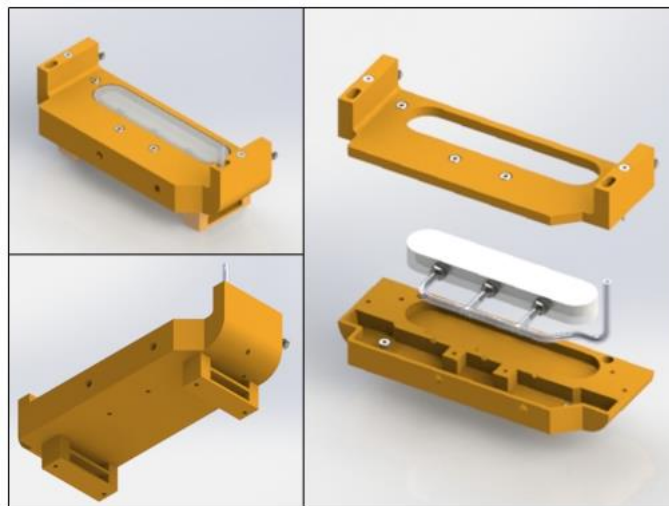
Figure 12: Palm Module: Isometric view (top left) Bottom view (bottom left) Exploded view (right)

The palm actuator input tube along with the two input tubes from the fingertip actuators were then routed through the side walls of the palm module into the glove shell. This can be seen in Figure 13. Routing these tubes in this way allowed for the palm module to slide up and down freely without getting caught. It also hid the tubes for a sleeker design.


Figure 13: Tube input from pal module and use of HERO Glove's strap

The side walls of the palm module also gave the ability for the module to slide up and down in the glove shell allowing for size adjustment. In order to control the size adjustment, a nylon strap was used. This strap was fixed to the glove shell and the strap was fed through the bottom set of holes on the strap guides, seen in Figure 13. The strap then wrapped around a feature on the glove shell and was fed back through the top holes of the strap guides. This design allowed for the module to be tightened evenly on both sides by simply pulling on the strap and locking it with the strap's locking mechanism.

3.4.1.3 Pointer and Middle Finger Modules

To house the finger joint and fingertip actuators a finger module was design. The design of the pointer and middle finger modules were identical, however, the length of the finger joint actuators differed for both fingers to match the human finger anatomy. These modules can be seen in Figure 14. The module comprised of three components, two rigid components and the finger joint actuator that will be outlined
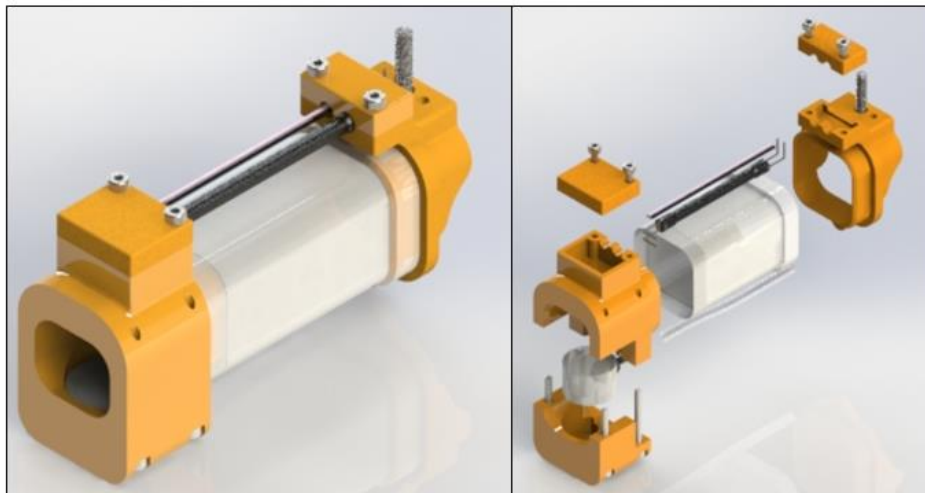
Figure 14: Finger Module: Isometric view (left) Exploded view (right)

The first component was the finger base part seen on the right hand side of the module in Figure 14. This component was designed to be 3D printed allowing for a rigid opening at the front of the finger modules. This allowed for the user to easily slide their finger in and out without having to push through any material. It also allowed for the finger joint actuator to be held open. The finger base part was designed to have a channel for the input air tube to the finger joint actuator. The channel was able to hide the air tube from the outside of the glove and protect it from being punctured. The channel can be seen in a cross-section depicted in Figure 15. Finally, a lip was designed on the inside edge of the part to allow the finger joint actuator to be stretched over.

Figure 15: Finger joint actuator input tube channel

The second component was the fingertip assembly. This assembly, seen in the left side of the module in Figure 16 was designed to house the fingertip actuator. It was necessary to have a rigid part to house the fingertip actuator to ensure that when inflated the actuator expanded upwards towards the user's finger. To fix the actuator in place the fingertip component was designed as two parts. The top component, seen in Figure 16, allowed for the user's finger to enter the assembly. The bottom component had an opening for the actuator to be mounted in. To keep the actuator fixed two half holes were designed into the top and bottom pieces. These holes were dimensioned slightly smaller than the vented screw entering the fingertip actuator. This caused a friction fit between the screw and the two half holes. Four M2 screws were used to fix the top and bottom parts of the fingertip assembly.
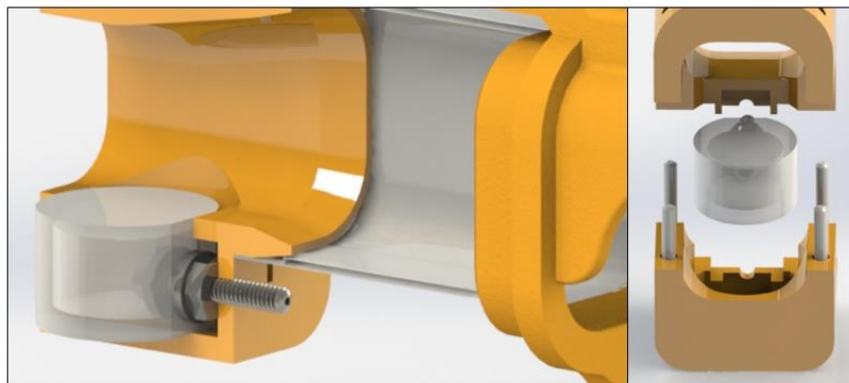
Figure 16: Fingertip actuator mount: Cutout view (left) Exploded view (right)

The final functionality of the finger module is the mounting hardware for the finger curvature sensors. In order to allow for easy removal of the sensor clamp plates were designed to hold the tubing of the sensors in place. On the finger base end the tubing and IR LED were held in place by the clamp plate. The sensor then was suspended over the finger joint actuator to the fingertip. On the fingertip end the IR photo resistor and tubing were clamped in place using the adjacent clamp plate. An opening was created to allow for the contacts of the IR photo resistor of

the finger curvature sensor to wrap around. The wires were then fed through another opening and suspended over the finger joint actuator back into the finger base component. This allowed for the four wires of the sensor to be easily routed into the glove shell.

To complete the assembly of the module the fingertip actuator was inserted and the two parts of the fingertip assembly were fixed together. Next, the finger joint actuator was fixed to the finger base by inserting the air input tube through the channel described above. The actuator was stretched around the lip of the finger base and secured to the 3D printed part using super glue. The same process was repeated to fix the finger joint actuator to the fingertip assembly.

3.4.1.4 Thumb Module

Once the design and manufacturing was completed for the pointer and middle finger modules a thumb module was then designed. The design of the thumb module was identical to the finger modules, however the dimensions of the opening of the 3D printed components was made larger to accommodate the size of the human thumb. The length of the thumb joint actuator was also altered to be shorter in length. The final design of the thumb module can be seen in Figure 17.
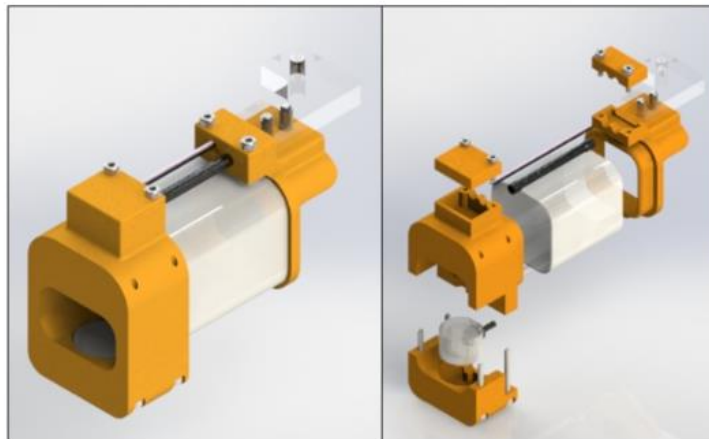
Figure 17: Thumb Module: Isometric View (left) Exploded view (right)

3.4.1.5 Arm Sleeve Design

In order to control the position of the robot manipulator with the HERO Glove three IMU sensors were used. The use of these sensors will be explained in a later section, however, the mounting of these sensors will be discussed in this section.

One of the three IMUs was located in the glove shell, described above, and the other two sensors were located on the user's forearm on a sleeve. In order for the program to successfully control the manipulator one IMU was positioned above the user's elbow and the other below. To package these sensors efficiently so that the user could easily attach the sensors to their arm a commercial elbow brace was used. This elbow brace, seen in the right of Figure 18, was then slid on the user's arm. The user would then be instructed to slide the brace up the arm until the center of the brace, located by the tightening strap, was around the user's elbow.
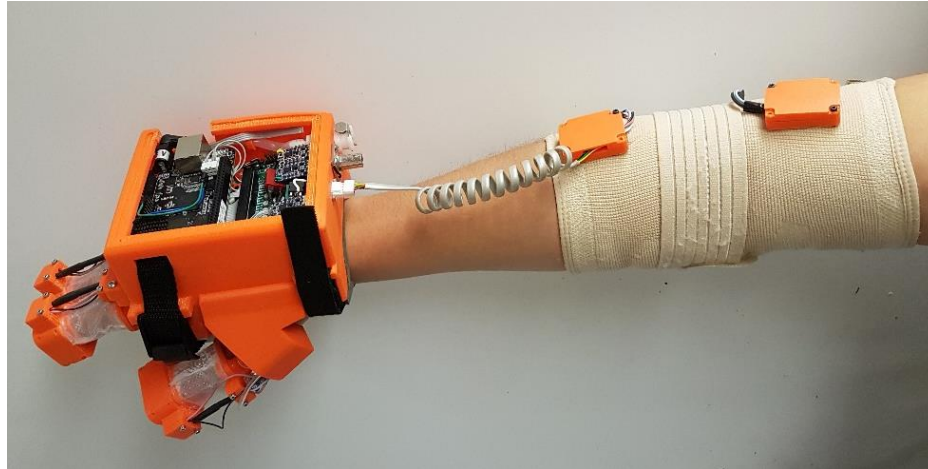
Figure 18: HERO Glove with IMU sleeve on user

To ensure that the sensors were always in the correct orientation in reference to themselves and the glove the sensors were fixed to the sleeve. To protect the sensors from damage and allow for the sensors to be fixed to the sleeve a 3D printed case was designed for the two IMUs. Seen in Figure 19, the IMU was fixed to the bottom of the case using M2 screws. The top of the case was then fixed to the bottom using M2 screws. The bottom of the case was designed to have an opening to allow for the wires, connecting the sensors to the glove, to enter the case. The bottom of the case also included through holes to allow for the case to be stitched to the arm sleeve. This along with super glue fixed the cases firmly to the sleeve.
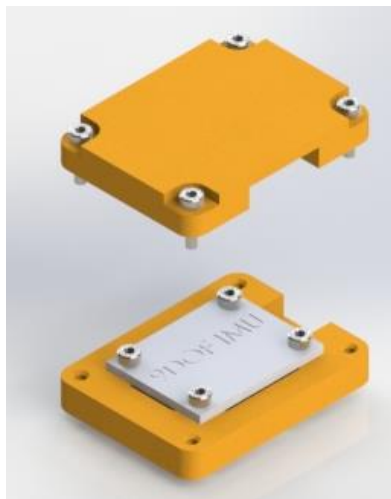
Figure 19: IMU case

To connect the sensors to the glove shell a stretchable coiled wire was used with a four pin connector. This allowed for easy removal of the sleeve when putting the sleeve on or taking the sleeve off. To connect the lower IMU to the higher IMU four wires were used.

3.4.2 User Hand and Arm Gesture Measurement

The configuration of the user's hand was captured using the HERO Glove's built-in sensors: the inertial measurement units, and the finger curvature sensors.

3.4.2.1 Arm Position Tracking

Three BNO055 9-DOF Inertial Measurement Units (IMU) were used to calculate the glove's position and orientation relative to the user's shoulder. The BeagleBone Black communicated with the IMUs using I²C protocol and a TCA9548A I²C multiplexer, due to the BNO055 having hard-coded physical addresses. The BNO055 contained its own processor, which performed sensor fusion on its sensor inputs to derive orientation.

3.4.2.2 Inertial Measurement Unit Program

The user's arm was subcategorized into three main parts, with a single IMU being attached to each part: the upper arm, the forearm, and the hand. The sleeve allowed for the orientation of the upper arm and forearm to be measured, while a single IMU on the actual HERO Glove's body allowed for the hand's orientation to be measured.

The orientation of each IMU was with reference to the user's shoulder joint, so solving the forward kinematics problem was not necessary. Instead, three-dimensional vector addition was performed using three vectors, where each vector was comprised of its respective IMU's orientation and respective arm sublink length, which was set to extend along the positive x-axis, correlating to how the IMUs were oriented. By summing the three vectors, the position of the HERO Glove relative to the user's shoulder was able to be derived. The orientation of the HERO Glove was a much simpler matter to calculate, with it merely being the orientation of the IMU on the HERO Glove's body.

The mathematics involved in calculating the end-effector pose in the global coordinate frame were as follows:

$$R_x(\theta_x)_a^b = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta_x) & -\sin(\theta_x) & 0 \\ 0 & \sin(\theta_x) & \cos(\theta_x) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_y(\theta_y)_a^b = \begin{bmatrix} \cos(\theta_y) & 0 & \sin(\theta_y) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta_y) & 0 & \cos(\theta_y) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_z(\theta_z)_a^b = \begin{bmatrix} \cos(\theta_z) & -\sin(\theta_z) & 0 & 0 \\ \sin(\theta_z) & \cos(\theta_z) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T(L)_a^b = \begin{bmatrix} 1 & 0 & 0 & L \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C\left(\theta_z, \theta_y, \theta_x, L\right)_a^b = R_z(\theta_z)_a^b \, R_y\left(\theta_y\right)_a^b \, R_x(\theta_x)_a^b \, T(L)_a^b$$

$$C(\vec{\theta}_0, L_0, \vec{\theta}_1, L_1, \vec{\theta}_2, L_2)_B^2 = C\left(\vec{\theta}_0, L_0\right)_B^0 + C\left(\vec{\theta}_1, L_1\right)_B^1 + C(\vec{\theta}_2, L_2)_B^2$$

Each theta vector represents the roll, pitch, yaw reading corresponding to each of the three IMUs. The supplied lengths followed an 8:8:1 ratio, and summed up to the robotic manipulator's maximum reach.

3.4.2.3 Implementation of IMU Sleeve

The HERO Glove control system was comprised of the HERO Glove's main body, and an additional sleeve, to which two BNO055 IMUs are attached. Each IMU was attached in a way that allowed for independent control of its orientation by a single arm sublink. The two IMUs were able to be wired directly to each other to support an I²C bus as the BNO055 IMU was capable of being configured to have one of two optional physical hardware addresses. This allowed for two BNO055 IMUs to be daisy-chained, as was the case with the sleeve. The forearm IMU then interfaced with the I²C multiplexer on the HERO Glove's main body, which allowed for the use of three BNO055 IMUs.

3.4.2.4 Finger Curvature Sensing

To control the grasping of the robotic gripper, we must first accurately measure our finger joint angles to map and send as commands for the robot.

3.4.2.4.1 Curvature Sensor Design.

We have experimented with numerous flexible curvature sensors, both commercially available components and the ones designed by the WPI Soft Robotics Lab, however, each of them do not produce the response that we needed for our system. For example, when using the curvature sensor test fixture that we developed, we were able to see a lot of hysteresis among resistive-based curvature sensors. This means that when our fingers bend and then return to its initial straight pose, the readings are not consistent. Examples of such sensors we tested were the Flex Sensor 10K and TSP Series ThinPot, which are depicted below respectively in order.

Figure 20: Resistive flex sensors that we have tested

Another curvature sensor we have developed was a modification of the WPI Soft Robotics Lab's curvature sensor that was used in the pneumatic air snake that utilized a HES and magnet (Luo et al, 2015). However, after much experimentation, we did not use those sensors due to us needing to exceed the optimal distance between magnet and HES, which produced a non-linear response. Furthermore, these aforementioned sensors were flat but not stretchable, which reduced accuracy when it had to be placed either above or below the fingers as it required sliding to compensate for its inelasticity.

With these limitations in mind, we designed our own curvature sensor with the specific purpose of being incorporated with the soft pneumatic actuators on the finger modules. Using a light-based design, the sensor is as responsive as the material it is made of. As depicted in Figure 21, the sensor consists of an infrared light emitting diode (IR LED) on one end of a flexible, elastic opaque rubber tubing, and a phototransistor on the other end of the tube. By emitting infrared light through one end of a fully sealed opaque tube, the voltage collected at the phototransistor on the opposite end would be proportional to the amount of bend on the tube due to reduced light passing from the IR LED to the phototransistor. This entire system, consisting of the flexible tube, IR LED and its drive circuitry, and phototransistor and its amplifying circuitry, constitutes the sensor. Thus, when there is zero bending, the signal amplitude is theoretically the greatest as the signal does not get reflected along the tubing. As the sensor bends, less of the signal reaches the phototransistor, producing a smaller voltage reading.
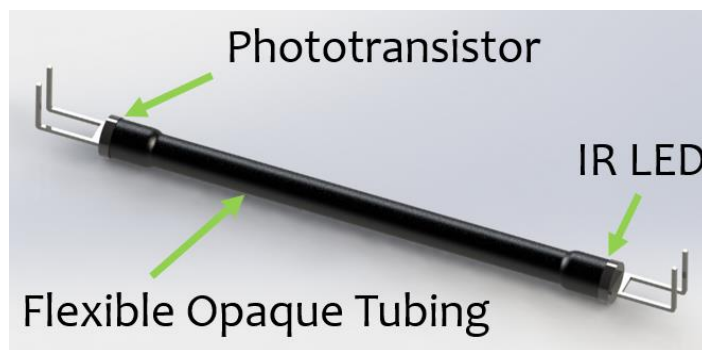
Figure 21: Design of light-based curvature sensor

The light-based curvature sensors were mounted above the actuators on the top of the finger modules due to user ergonomics. When the sensors were mounted on the side, the sensor mounts would extend to the side of the finger, resulting in chaffing of the ring finger from the increased bulk of the middle finger module. This can reduce user comfort. Change in voltage reading is due to the amount of stretch over the finger joint that results in bend. This is illustrated in Figure 22.

Figure 22: Light-based curvature sensor mounted on finger module

The final sensor design consists of a black elastic rubber tubing that slides over the 3mm IR LED and phototransistors, heat shrunk with two lengths of shrink tubing at each end of the tube. The heat shrinks were to secure the components in place whilst allowing for the sliding of heat shrink to accommodate the elasticity of the rubber tube.

3.4.2.4.2 Signal Conditioning Circuit

The circuit required to condition the sensor readings before entering in the analog-to-digital converter (ADC) of the microcontroller consists of an amplifier to increase the voltage change to a more workable range, low pass filter to reduce noise, and unity gain buffer to prevent loading. The phototransistor's readings were amplified by -3030 V/V gain to produce a consistent 0 to 2V workable range when powered through a 3.3V rail. This gain is determined using the inverting op-amp equation, displayed below.

$$Av = -\frac{Rf}{Ri} = -\frac{1,000,000}{330} = -3030\frac{V}{V}$$

Where A_v is the gain of the amplifier, R_f is the feedback resistance, and R_i is the input resistance to the op amp, established by the current limiting resistor on the IR LED. The gain was determined through experimentation as the voltage change was very difficult to read through an oscilloscope. We first started by creating a gain of 1000, using Rf as 1MΩ and Ri as 1kΩ.

However, this gain produced a signal of approximately 1.9V when the sensors are straight, which is not realistic for a finger actuator as users tend to bend their fingers a little bit even when not trying to close their grasp, reducing the max voltage to 1.7V under normal use. By reducing the R_i to 330Ω, the gain was able to increase to -3030V/V, which now allows for a max voltage of 2.1V when installed on the glove with slight inevitable bending on the finger actuators.

The low pass filter was designed to have a cutoff frequency of 132 Hz to attenuate undesirable high frequency noise from affecting the curvature sensor readings. The cutoff frequency was determined through the following equation.

$$f_{3dB} = \frac{1}{2\pi RC} = \frac{1}{2\pi * 1200 * 0.000001} = 132 \ Hz$$

An LM358N single supply op-amp was used due to availability and two op-amps in one chip, allowing amplification and unity gain buffering to be done on one chip. Furthermore, its single-supply capability makes it more suitable for portable systems that only have access to single voltage sources when compared to dual-supply op-amps such as the LM741.

Below is the circuit schematic of the curvature sensor system. The cathode of the phototransistor is connected to the V_{in}^- of the op-amp, while the anode is connected to ground.
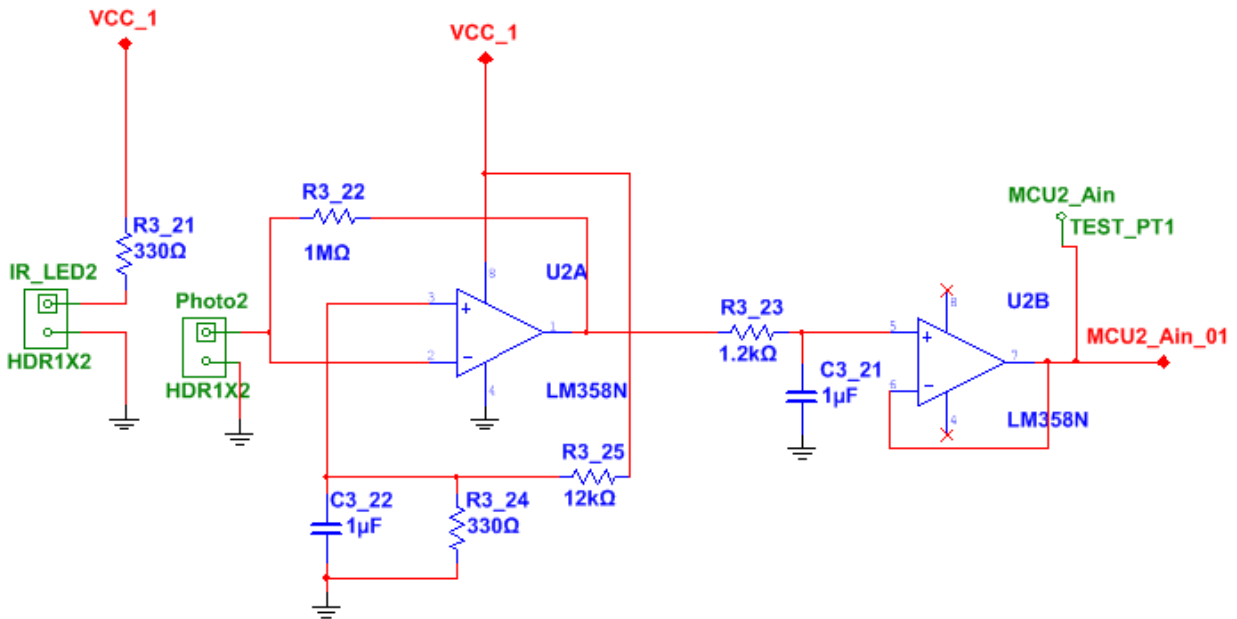
Figure 23: Light-based curvature sensor circuit schematic

One important design consideration of the phototransistor signal amplifier circuit is the use of a single-supply op-amp and the necessary offset that is needed to allow for the full range of sensor reading. Unlike conventional op-amp laboratory exercises, the single-supply op-amp such as the LM358N that is used does not connect its rails to 12V V_{cc+} and -12$V_{cc-.}$ Instead, its V_{cc} rail is connected to the 5V source, and its V_{ee} is connected to ground. Furthermore, the LM358N configured as a photodiode transimpedance amplifier for amplifying the light-

36

dependent current of a phototransistor requires a small bias voltage derived from the positive supply and applied to the op-amp's non-inverting input. This is to prevent the output from saturating at the negative supply rail in the absence of input current. The DC bias that we ended up using from the resistor network is 130 mV.

A transimpedance amplifier is a current to voltage converter where the gain, or transimpedance, is equal to the feedback resistance on the op amp. This is used with photodiodes and phototransistors because the diode's current response is more linear than the voltage response. The transimpedance amplifier presents a low impedance to the photodiode and isolates it from the output voltage of the op amp. Even though there are various different configurations of trnsimpedance amplifiers, each suited to particular application, the one common factor between them is the requirement to convert low-level current of a sensor, such as in our case the phototransistor, to a voltage.

3.4.3 Haptic Feedback

This section overviews the haptic feedback system of the HERO Glove.

3.4.3.1 Force Feedback

The HERO Glove was designed to incorporate force feedback in order to achieve more intuitive control of a robotic manipulator. From the previous research it was determined that soft actuators would provide more accurate and realistic feedback. This section outlines the design and manufacturing of these soft actuators.

3.4.3.1.1 Finger Joint Actuators

3.4.3.1.1.1 _Design_

To provide feedback to the user for the sensation of grasping an object the concept of a toroidal actuator around the user's finger was decided upon. The toroidal approach had not been previously attempted by other haptic feedback gloves. Many approaches were attempted to create these toroidal actuators. Initially, multi-step silicone molds were designed and 3D printed. These molds provided poor results and the small wall thickness required by the actuator made it difficult to mold properly. It was also difficult to keep the silicone completely sealed. The next approach was to use flexible tubing. The tubing was folded inside itself and sealed using glue to form a toroidal shape. After several attempts the tubing was unable to remain air tight and appropriate sized tubing was not available within the budget of the project. More detail in the attempted toroidal actuators can be found in Appendix C and Appendix D.

It was then decided to attempt heat sealable plastic sheets for the actuator design. This approach resulted in the best made actuators and are incorporated in our final design. The final design of the actuator can be seen in Figure 24.

Figure 24: Finger joint actuator

The thickness of the plastic sheeting was approximately 0.15 mm in thickness. The shape of the actuators was a square opening with rounded corners. The dimensions of the square differed for the pointer and middle finger actuator and the thumb actuator. The pointer and middle finger actuators had an opening of 22mm by 22mm. The thumb joint actuator had an opening of 25mm by 25mm. These dimensions were chosen to fit the finger size of the team members to allow for comfortability when testing. The length of the actuators differed between all three actuators, however, the design of the actuators made it easy to change the opening dimensions and the length of the actuator.

3.4.3.1.1.2 *Manufacturing*

To manufacture the actuators, seen in Figure 25, vacuum sealable food storage plastic was cut and heat sealed to create the toroid. Two layers of the plastic were initially cut to size using a laser cutter. The plastic roll was made to have a pre sealed edge, so the position of the laser cut allowed for a small strip of the pre sealed portion of the plastic to remain. This resulted in two layers of plastic with a seal along the longer edge. The opposite edge was then fed into the heat sealer so create a cylindrical shell of plastic. The heat sealer had an offset from the alignment wall to the heat sealing strip so the size of the rectangle cut by the laser cut took into account this offset. Once the cylinder was made, one end of the cylinder was folded over and fit into itself. This created an open toroid of plastic.

A thin piece of acrylic was then slid into the center opening. This acrylic piece prevented the center opening from being sealed closed. The first side of the opening was sealed. Once again, the laser cut plastic rectangle took into account the needed material to fold over the plastic and the offset of the heat sealer strip to ensure the length of the final actuator was the desired length. Once the first side of the opening was sealed, a 1mm diameter tube was fed through the folded over end of the toroid. A small amount of hot glue was applied to the tube about 10mm from the end. The tube was then pulled through the wall of the toroid until the hot glue sealed to hole made from the tube. The hot glue produced a tight seal.

 Finally, again using the acrylic, the other end of the toroid was sealed. This resulted in a completely sealed actuator. The actuator was then tested with the air supply to ensure no leaks. If leaks were suspected, the actuator was dipped into a cup of water and air was fed into the chamber to see if any air bubbles escaped. Due to the low cost of the actuator and the quick

assembly time, if the actuator was not sealed properly, the actuator would be scrapped and the process would start over. Again, the process of creating the thumb actuators was exactly the same. The only difference was the size of the final actuator. The final manufacturing process included three files for the three actuator sizes. The total time of assembly was approximately 10 to 15 minutes.

Figure 25: Completed finger joint actuator

3.4.3.1.2   Fingertip Actuators

3.4.3.1.2.1.1 <u>*Design*</u>

To provide tactile feedback to the user's finger tips, silicone actuators were designed and molded. For simplicity, the fingertip actuator was designed to be the same for the finger and thumb modules. The geometry of the actuators was a simple hollowed cylinder with one flat edge, seen in Figure 26. The flat edge was incorporated for a tight seal. The hollowed cylinder allowed for the actuator to be inflated, resulting in the bulging out of the walls. The actuator was designed to fit snuggly in the fingertip assembly of the finger and thumb modules. When the actuator was in the 3D printed component it only allowed for the top wall of the actuator to bulge. This produced the tactile feedback to the user's finger tips.

Figure 26: Fingertip Actuator

In order to allow for air to enter the actuator and ensure an air tight seal, vented screws were used. As seen in Figure 27, a vented screw (a screw with a through hole down the center of the body of the screw) head was inserted through the wall of the silicone from the inside of the cylinder. Using two washers and a nut, the opening in the silicone wall was sealed. The air supply could then be inputted into the actuator by sliding a tube over the threads of the vented screw.
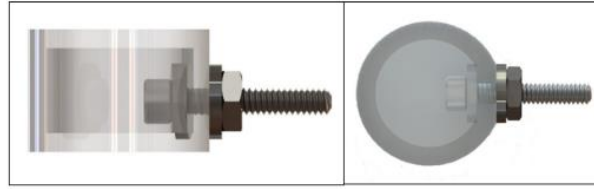
Figure 27: Air input to fingertip actuator

3.4.3.1.2.1.2 _Manufacturing_

To create these silicone actuators a mold was designed. This mold was a two-step molding process. The first step, seen in Figure 28, consisted of a mold base and an insert that created the bottom and body of the cylinder. The insert created the air pocket in the center of the cylinder and a screw was inserted into a hole in the front of the mold to keep an opening for accurate positioning of the vented screw. The insert also had cutouts to allow for air to escape the mold, preventing air bubbles in the dried silicone. Seen in the right of the figure is a mold for the top face of the actuator.
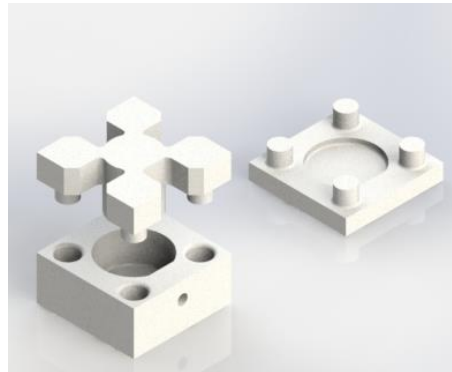
Figure 28: Fingertip actuator base mold

To create the actuators Dragon skin silicone was mixed and left in a vacuum chamber for about 10 minutes. The vacuum chamber extracted many of the air bubbles left in the silicone. Once the silicone was de-gassed, it was then poured in the mold base and the top face mold. The mold insert was then inserted into the mold base and centered by the four posts seen in the figure above. A straight edge was then used to scrape the excess silicone from the opening of the insert. The straight edge was then used to scrape away excess from the actuator top mold. The two molds were then left to dry for approximately four hours.

After the molds were dry, the actuator body was removed and the vented screw, washers, and nut were installed into the hole left from the mold. Once it was ensures that the nut was tight the screw was pulled to ensure that the washers kept the opening in the silicone closed. The body with the installed screw was then inserted into another mold base seen in Figure 29. The top face of the actuator was left in its original mold.

To bond the actuator body with the top face, a small amount of silicone was mixed. It was then spread along the top edge of the actuator body. Using the four posts, the top face mold was inserted in to the mold base and clamped together. The mold was left to dry for four hours.

Once the actuator was completed, the actuator was tested for leaks. If leaks were suspected the actuator was dipped in water to see if any air bubbles escaped. It was also tested to see if it could withstand at least 5 psi of pressure.
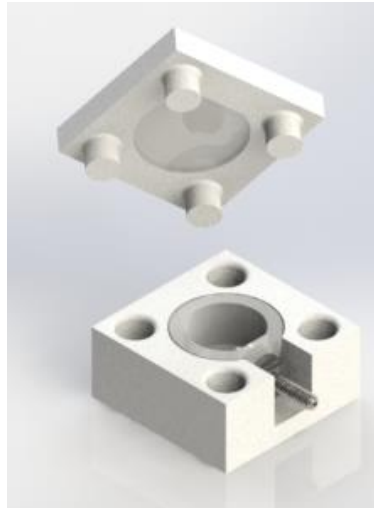
Figure 29: Fingertip actuator assembly mold

Due to the low cost of the silicone, if the actuator was not sealed it would be scraped and the process would start again.

3.4.3.1.3 Palm Touch Actuator

3.4.3.1.3.1 <u>Design</u>

Similarly to the fingertip actuators, the palm touch actuator was designed to be molded from silicone. The geometry of the actuator was based off the geometry of the fingertip actuator. The width was the same, however, to accommodate the size of the user's palm the actuator was designed to be 73mm long. The way in which air was inputted into the actuator was also identical to the fingertip actuator, however, due to the length of the actuator three sets of vented screws, washers, and nuts were used. The final design of the actuator can be seen in Figure 30.
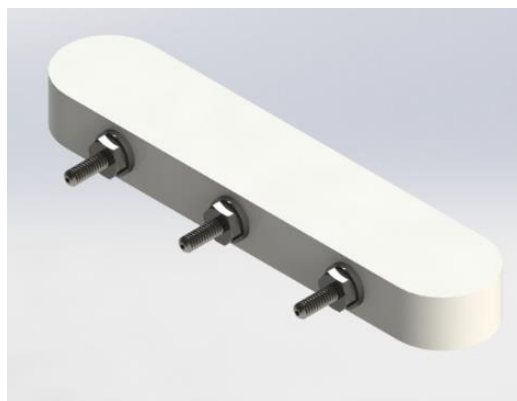
Figure 30: Palm touch actuator

The manufacturing of the palm touch actuators was also identical to the fingertip actuators. A two-step molding process was performed. The first step resulted in the body of the actuator and a separately molded top. The vented screws, washers, and nuts were then installed into the actuator body. The top of the actuator was fixed to the body using a small amount of silicone and the actuator was left to dry. When it was completed it was tested to ensure that the actuator was air tight and could withstand at least 5 psi of pressure.

3.4.4 Solenoid Valve Manifold Specifications

In order to provide pressure control of the seven feedback actuators, a solenoid valve manifold was purchased. The manifold seen in Figure 31, was purchased from SMC Pneumatics. The model of the valve was S070C-SCG- 32. This manifold allowed for one input and one output to all seven valves. It also allowed for easy connection to the actuators due to the manifolds barbed connectors. The manifold was then fixed in the glove with two screws. Technical data for the valves can be seen in Appendix J.
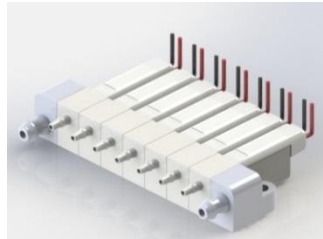
Figure 31: SMC S070C-SCG-32 Solenoid Valve Manifold

3.4.5 Solenoid Valve Drive Circuitry

A dedicated drive circuitry is required to drive the solenoid valves with sufficient current, in which the microcontroller pins are unable to supply. In order to do this, we designed a transistor driving circuit, in which we drive the base of the transistor with the microcontroller output signal to allow for the solenoid to be driven with the larger voltage and current from the V_{CC}. The transistor that we used is the 2N2222A BJT instead of a prior IRFD110 MOSFET due to the BJT's 3.3V turn-on voltage capability, which is the digital HIGH output of the microcontroller. By turning on the BJT, this allows for the valve to be powered by the 5V from the V_{CC}.

2N2222
BC107, BC108, BC109

Tag

c   b   e

e - emitter
b - base
c - collector
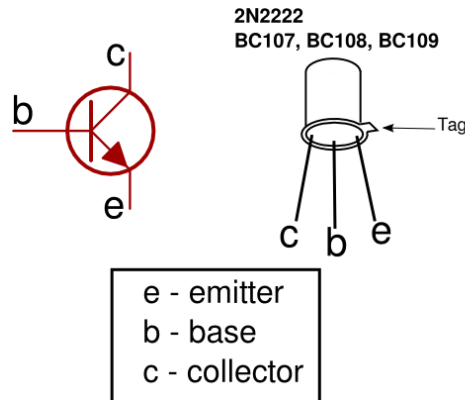
Figure 32: 2N2222A used in solenoid circuit

The following diagram details the drive circuitry for one finger module, which takes in the PWM signal from the AdafruitTrinkets[9] microcontrollers to control the transistors. A 1 kΩ resistor was used to limit the current at the transistor's base, I_b. This resistance is acceptable for limiting the current as calculated from the following equation, knowing that the maximum current at the collector, I_c, is 150 mA as indicated from the valve's datasheet.

$$I_b = \frac{I_c}{\beta} = \frac{150 \ mA}{100} = 15 \ mA$$

As the voltage drop across base-emitter is 0.7V, the total voltage is 2.6V, as 0.7V was subtracted from the HIGH signal voltage coming out of the trinkets at 3.3V. The base resistance was derived from the following equation.

$$R = \frac{V}{I} = \frac{2.6V}{15mA} = 173.3\Omega$$

The resistance that was required was 173.3Ω, therefore, a 1kΩ resistance is more than enough to limit the current to avoid damaging the BJTs.

A 1N4001G diode was used to freewheel the back electromotive force (EMF) created by the switching of the valves to prevent backflow of current that can damage it. The valves are connected across the header pin with the positive lead connected to V_{CC} node and negative lead connected to the transistor's collector, thereby the valves become a resistive load to pull the current through.

---

[9] AdafruitTrinket mini microcontroller https://www.adafruit.com/products/1500
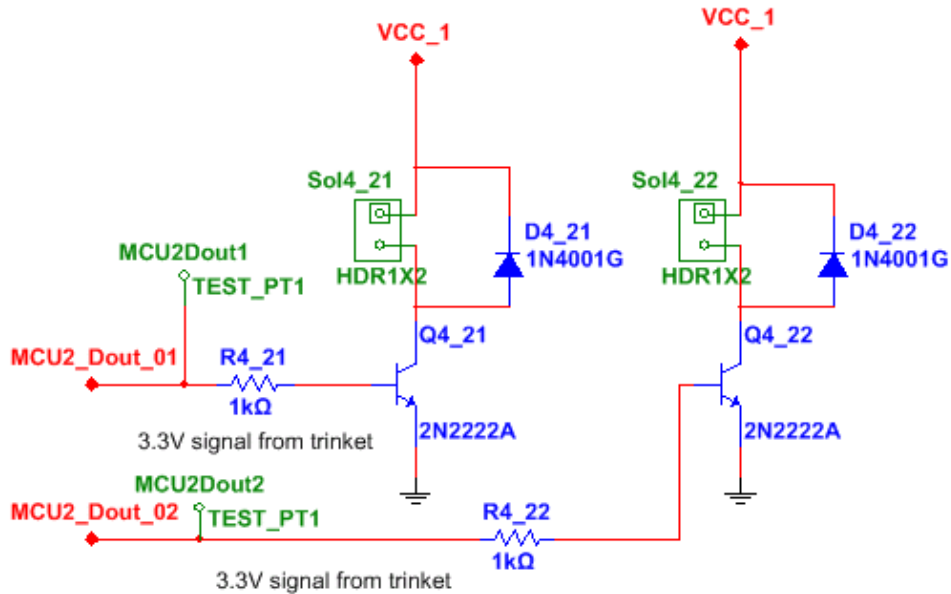
Figure 33: Solenoid valve drive circuitry for one finger module

Test points were created before the base resistor to allow for easy testing and debugging of circuit to measure the PWM signal coming directly out of the microcontroller.

3.4.6 Solenoid Valve Control Program

Control over the pneumatic valves was achieved through pulse-width modulation (PWM). The period of each cycle was set to be 14 ms, to achieve a signal frequency of 70 Hz. The ROS node running on the HERO Glove's processor, the BeagleBone Black, would send two bytes of data over the I²C bus to the designated AdafruitTrinket, indicating whether to set the output voltage of the AdafruitTrinket two output pins to high or low. The ROS node would be responsible for monitoring and setting the duty cycle of each of the pneumatic valves.

3.5 Glove - Robot Communication

The flow of data between the HERO Glove and the robot being manipulated was designed to be bidirectional. Data in the form of ROS messages containing the HERO Glove's pose was sent to the robot, and sensor data regarding tactile sensor feedback was sent from the robot to the HERO Glove.

3.5.1 Glove to Robot

One of the key components of establishing a comprehensive flow of data from the HERO Glove to the robot manipulator was the MinnowBoard MAX. Since there was no guarantee that the robot being controlled would have wireless or sensing capabilities, a central hub of communications was deemed necessary. Fortunately, the MinnowBoard MAX was able to be set up to run Ubuntu, and so the MinnowBoard MAX was able to be configured to run ROS nodes.

Four ROS nodes formed the crux of the system's communications. The HERO Glove was configured to run a single ROS node, which published the user's finger curvature and hand pose

data to their respective ROS topics. The MinnowBoard MAX was configured to run three ROS nodes, to perform multiple responsibilities in parallel, as well as the ROS core, due to it having higher performance capabilities than the BeagleBone Black.\

3.5.1.1 Jaco Arm

The Jaco Arm was chosen as the robotic manipulator of choice to achieve a proof of concept in simulation. Controlling the Jaco Arm in simulation required that instead of running all three ROS nodes, a single ROS node be used instead, and on the machine running the simulation environment. The physical setup also differed slightly, as the Robotiq gripper was replaced by a laptop computer running the simulation software. The single ROS node acted as a bridge, subscribing to the topics being published to by the HERO Glove, and translating the data into commands to the simulated Jaco Arm. The Jaco Arm's end-effector pose and end-effector joint angles were able to be controlled through such means.

3.5.1.2 Robotiq Gripper

In addition to controlling the simulated Jaco Arm, we also implemented the HERO Glove to control the Robotiq 3-finger gripper. In the setup to accomplish this task, the Robotiq gripper was connected to the MinnowBoard MAX via an Ethernet cable, the BeagleBone Black then communicated wirelessly with the MinnowBoard MAX.

The Robotiq ROS driver was installed on the MinnowBoard MAX in order for it to send commands to the Robotiq gripper. A ROS node was created to run on the MinnowBoard MAX to subscribe to the curvature sensor data published by the HERO Glove. These values are scaled accordingly and sent to control the Robotiq gripper fingers.

3.5.2 Robot Force Sensing

This section details the system that we designed to detect force exerted on the controlled robot's gripper and relay that information back to the HERO Glove. This was then used to create the haptic feedback functionality by regulating the valve's PWM.

3.5.2.1 Tactile sensor design

In order to detect the force that is exerted on the robot's finger tips, joints, and palm, tactile sensors called the Robot Grip Force sensors, must be developed to measure, as the name suggests, the robot's grip force. This measurement of single-axis normal force is what is used to determine the haptic feedback on the HERO Glove when the system is controlling a real robot without built in force or tactile sensing information. The goal of these force sensors are that they must be small enough to fit on the robot's finger tips, and must be easily attached on without requiring any hardware modification.

We collaborated with the Frankenhand MQP team to develop these sensors which consists of an AH49e HES mounted on an etched PCB and a K&J Magnetics D101-N52 neodymium magnet mounted on a 1/32th-inch thick acrylic magnet holder. These two components are embedded in a silicone dome which is created using a mold to produce an elastic medium separating them. As force exerted on the Robot Grip Force sensor increases, the distance between the magnet and HES reduces, changing the magnetic flux and either increasing or

decreasing the HES voltage reading. The voltage reading would be obtained as an absolute value because either the north or the south of the magnet can be facing the HES during manufacturing, which we would not know when placing due to such small magnet size.
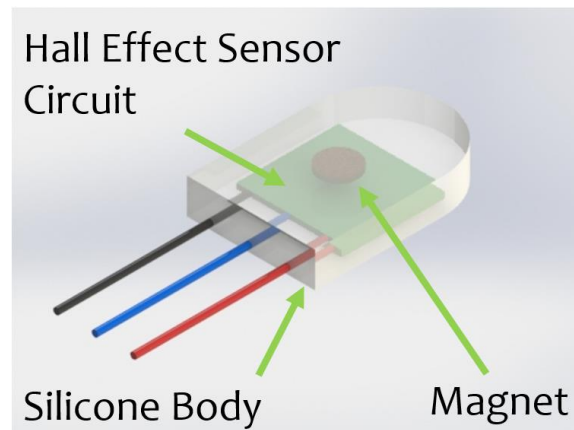
Figure 34: Robot Grip Force sensor design diagram

Figure 35 details the components that gets molded into the silicone. The AH49e HES was soldered to a etched PCB. The magnet was inserted onto the three-legged acrylic magnet holder that fits onto the mold, where the thin legs were broken off once the silicone has cured, to ensure that the magnets stay at a fixed height from the HES.
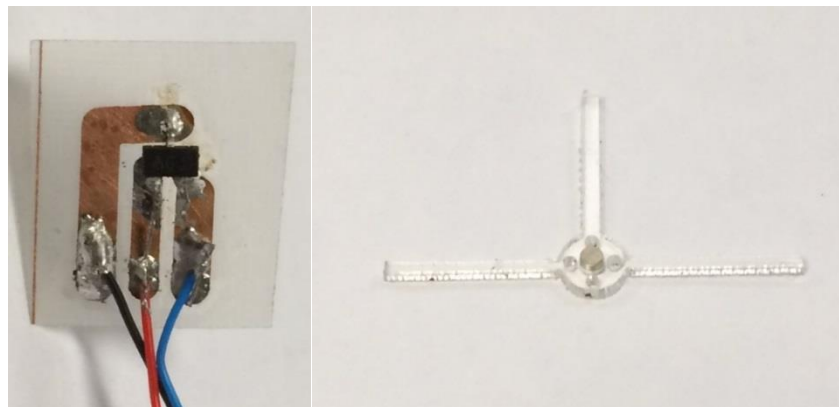
Figure 35: Robot Grip Force sensor components

The silicone mold for the Robot Grip Force sensor consists of two 3D printed halves with slots on the three sides for the magnet holder's legs to come out of. We used Dragonskin 10 silicone for the final Robot Grip Force sensor due to its greater stiffness than the Ecoflex 30. When using the softer Ecoflex 30 silicone, the circular magnet holder would deform more unpredictably upon contact, producing unreliable results. The stiffer Dragonskin 10, while producing overall less sensitive sensors, provided more consistent readings, which could be easily mapped or amplified to produce a workable range. After the silicone was mixed, we left it to vacuum in the vacuum chamber for five minutes, and then poured it into the mold, which we then left to cure for at least four hours.

Figure 36: Robot Grip Force sensor silicone mold

A finished Robot Grip Force sensor is depicted in Figure 37 with a size reference compared to a penny, as well as installed on the Robotiq gripper. The sensor was mounted on the gripper using a thick rubber band due to lack of double-sided tape that would adhere to the silicone. The sensor readings were scaled using a mapping function in Arduino. These sensors were then plugged into the Robot-Side Module, in which its reading can then be sent to the HERO Glove to vary PWM on the valves.
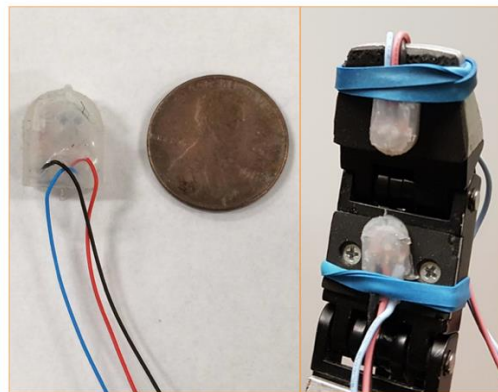
Figure 37: Complete Robot Grip Force sensor used on Robotiq Gripper

3.5.2.2 Robot-side module design

In order to feel touch and force on the robot's finger as it interacts with its environment, we created a system that relays force sensory data from the tactile sensors we developed back to the glove. This system, called the Robot-Side Module, consists of the tactile sensors, an analog multiplexer (MUX), a microcontroller, and a Bluetooth module to transmit the sensor readings to the MinnowBoard Max and then sent, through ROS, to the HERO Glove. The Robot-Side Module was designed with the intention of being mounted on the robotic gripper itself.
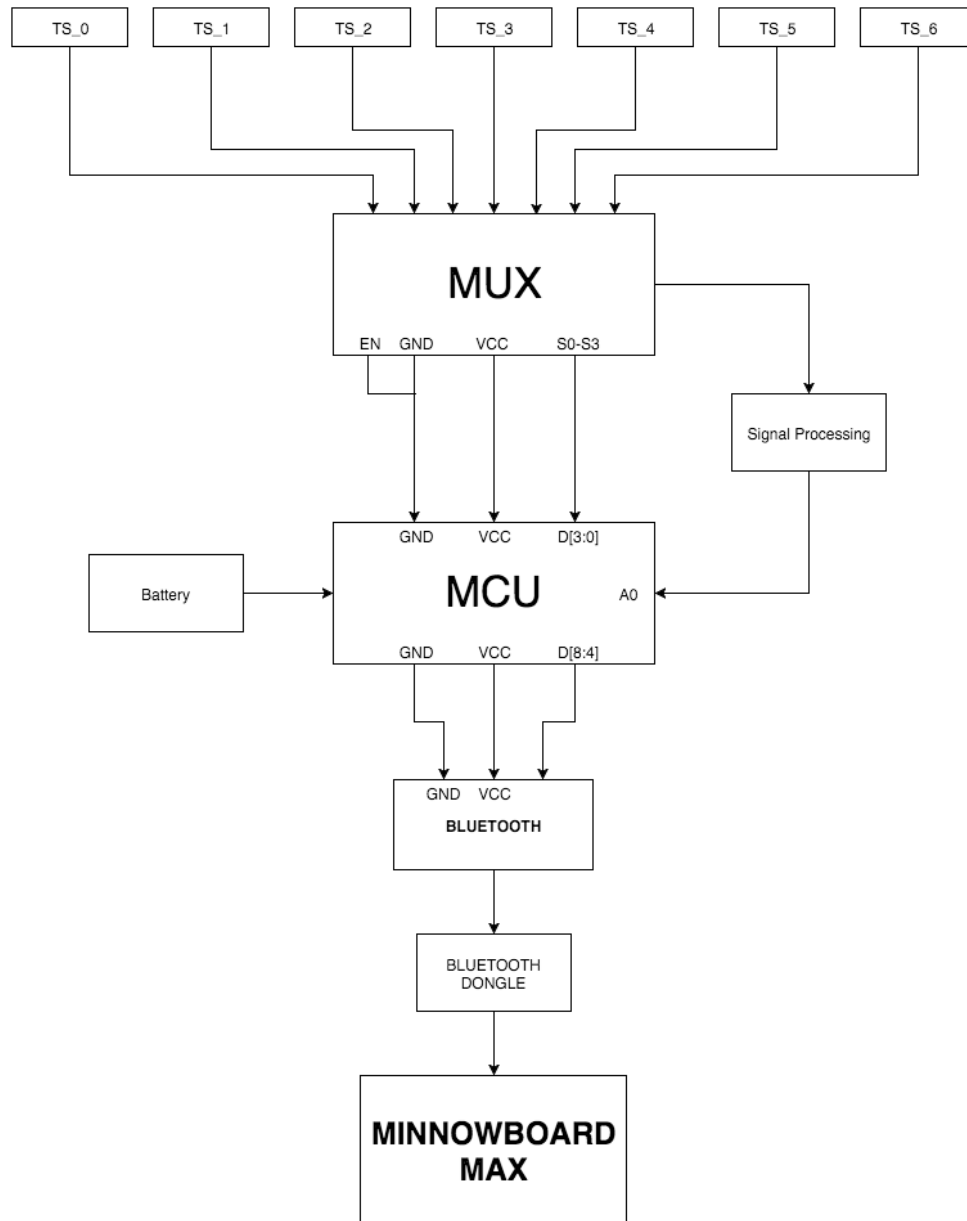
Figure 38: Low level diagram of Robot Side module.

The tactile sensor readings were multiplexed through a CD74HC4067 analog MUX [10] that takes up to 16 inputs. The MUX acts as an interface between the sensors and the microcontroller the MUX will take the sensor reading as inputs and transfer these data over to the Uno over a single signal pin. This also benefits us in the signal processing step, instead of having seven signal processing block for each sensor, we only need one. Beside the single analog pin, the MUX requires four other digital pins from the Arduino in addition to Vcc and Ground. Even though we only use seven sensors in the current system, having 16 inputs allow for easy scalability for future projects to create a more sensitive system.

---

[10] CD74HC4067 analog multiplexer https://www.sparkfun.com/products/9056

In order to process the input signal from the MUX, we used an Arduino Uno as the microcontroller, which converts the data into formatted string and send the reading from seven sensors as one package of one sample to the MinnowBoard Max. The Arduino Uno was chosen due to its easily accessible documentation and user community online.

Even though the Arduino Uno can communicate with the MinnowBoard Max through a serial cable, in practice, the entire Robot Side Module will be attached on the robot's manipulator and the use of a cable can mean the risk of it becoming tangled on the robot. This can potentially lead to both a performance and safety concern. As a result, we decided to use Bluetooth for wireless communication between the Robot Side Module and the MinnowBoard Max at the base of the robot being controlled. The Arduino Pro Mini itself does not have Bluetooth built in; therefore, we added the AdafruitBluetooth Low Energy UART Friend to send packages over Bluetooth.
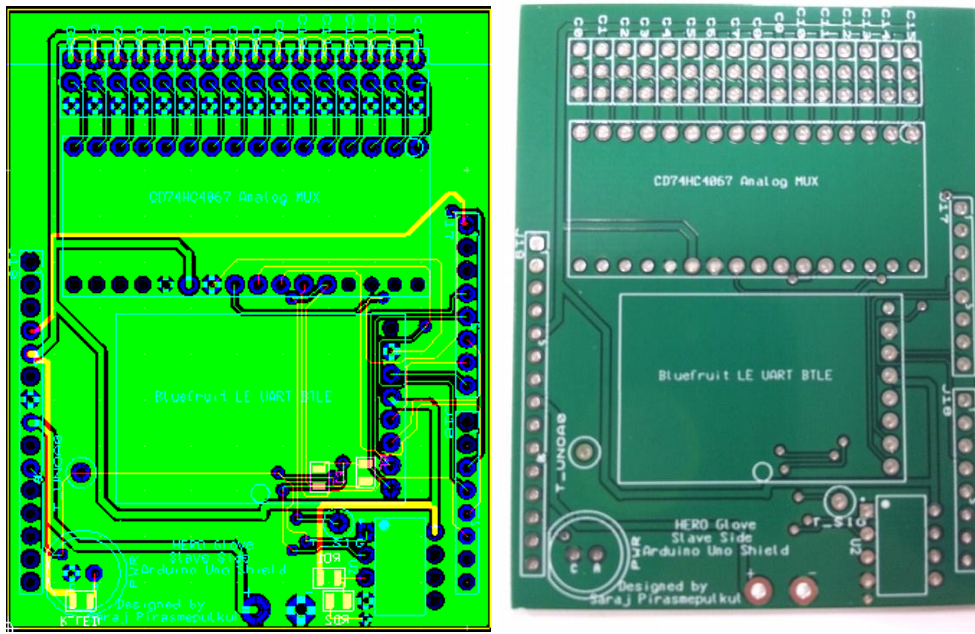
Figure 39: Printed circuit board design and actual Robot Side Module Arduino Uno shield.
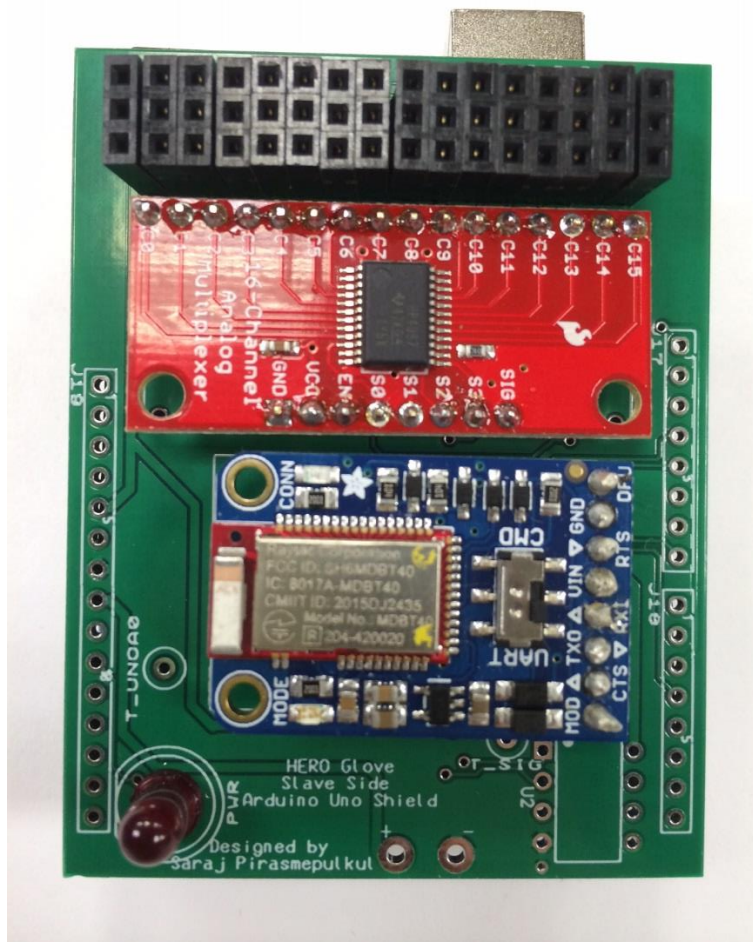
Figure 40: Populated Robot Side Module

3.5.3 Robot to Glove

As previously mentioned, the communication system used was designed to be bidirectional, to allow for the user to receive feedback from the robot manipulator's interactions with its environment.

3.5.3.1 Jaco Simulation

Achieving data-flow from the simulated Jaco Arm to the HERO Glove did not require the setup of any additional ROS nodes. The single existing ROS node running on the machine running the simulation was responsible for parsing data regarding the virtual tactile sensors' data and publishing it to ROS topics that the HERO Glove could subscribe to. The virtual environment's implementation of the Jaco Arm's end-effector tactile sensors was provided by the simulation software, but fortunately coincided with that of our own design. The reading from each tactile sensor was published as a 3-D vector, indicating the magnitude and orientation of the force being exerted on the specific sensor. An assumption that was made was that only forces

parallel to the Z-axis were relevant. This assumption allowed for simple parsing of the data received by the HERO Glove's ROS node.
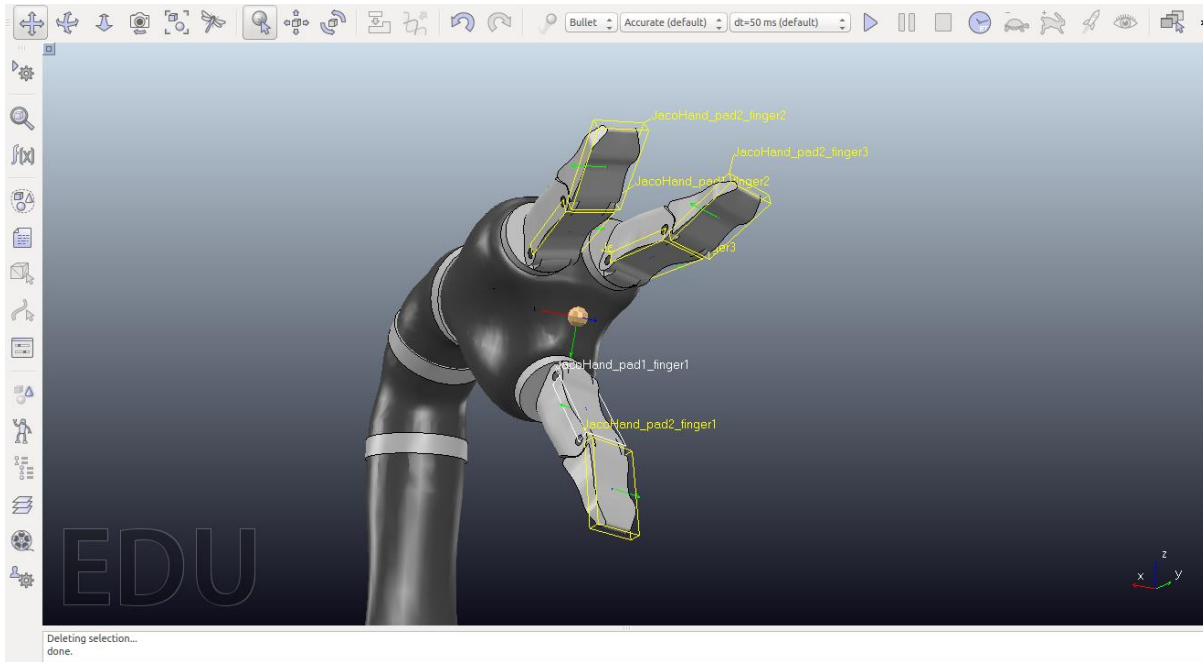
Figure 41: Virtual Tactile Sensors

3.5.3.2 Robotiq Hand

Communication between the Robot-Side Module was achieved via Bluetooth to prevent unnecessary wires from becoming tangled in the robotic manipulator. Similar to the Arduino Uno used on the Robot-Side Module, the MinnowBoard Max does not have Bluetooth built in, which was why we used the CSR8510 Bluetooth 4.0 USB dongle to the board.

The program that receive and send tactile sensor data is ran as a ROS node on the MinnowBoard MAX. After receiving the package sent from the Arduino Uno, the node will construct a ROS message containing the data of the seven tactile sensors and then publishing them as a ROS topic. The BeagleBone Black on the HERO Glove subscribes to this topic to receive these values.

3.6 User Performance Evaluation

To evaluate the performance of the system as a whole, several metrics and benchmarks were established to allow for an objective assessment to be made, based on empirical data. The assessment of the HERO Glove was divided into several subcategories, with each providing an assessment of each of the key physical and design-based components of the system. Additionally, several students from WPI who had no affiliation with the HERO Glove project were asked to put on the HERO Glove and perform a series of tasks to evaluate the performance of the HERO Glove. They were then asked to score the HERO Glove, according to predetermined metrics on a discretized scale. By weighting the evaluated scores from each of the

subcategories according to their importance, and calculating the total of the scores across all the subcategories, a score gauging the performance of the overall system was able to be derived.

Virtual Robot Experimentation Platform (V-Rep), was used to simulate the Jaco Arm robotic manipulator. A ROS node was set up to serve as a bridge between the ROS network and the simulated environment. Two scenarios were set up, to allow for all the key aspects of the HERO Glove system to be better evaluated.

3.6.1 Scenario A

Scenario A was used to allow the user to more easily assess the HERO Glove's haptic feedback and curvature sensor performance. Scenario A consisted of a single floating ball within the Jaco Arm end-effector workspace. The Jaco Arm joints were fixed to disable position tracking of the end-effector. The ball was positioned in such a way that it could be easily grasped by the user. The user was allowed to run through Scenario A as many times as they liked, as the ball being knocked away would require the simulation environment be restarted, should the user desire to experience the HERO Glove's haptic feedback again.
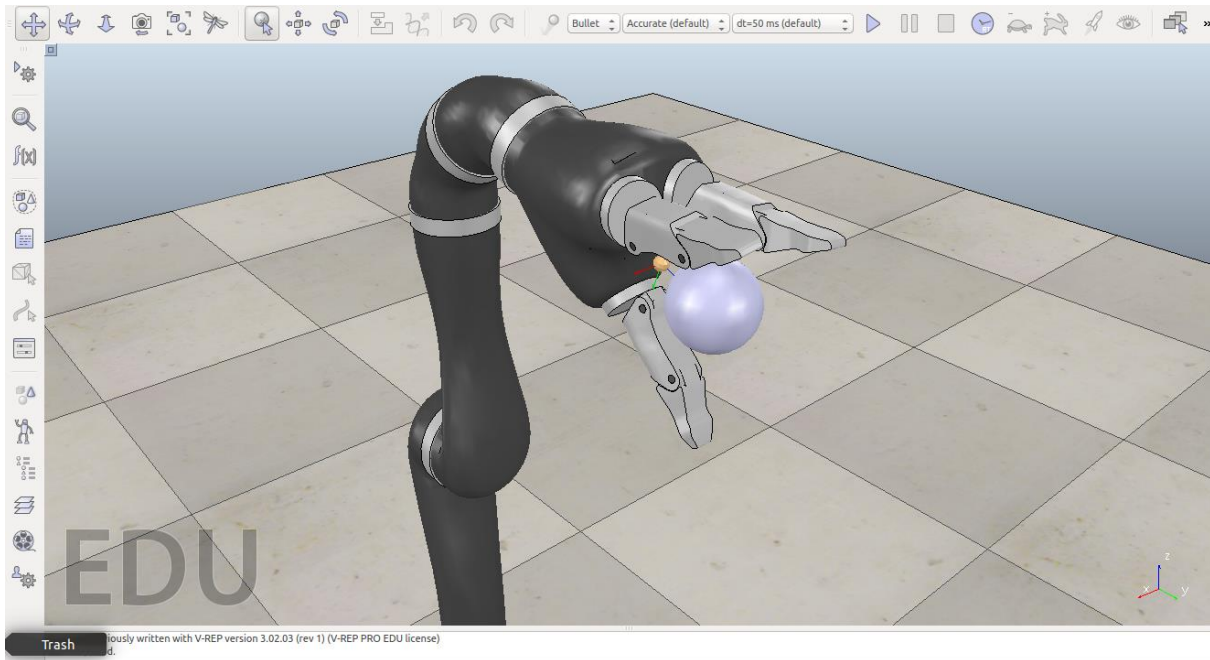
Figure 42: Scenario A in V-Rep

3.6.2 Scenario B

Scenario B was used to allow the user to experience the HERO Glove's capabilities as a cohesive system. Scenario B consisted of five floating balls, randomly positioned with the Jaco Arm workspace. In this scenario, the Jaco Arm joints were allowed their full range of motion, to allow for position control of the Jaco Arm end-effector by the user. The user was allowed to run

through Scenario B a maximum of three times, to properly gauge how easy to use, and intuitive the system was to a user with little to no premeditated training.
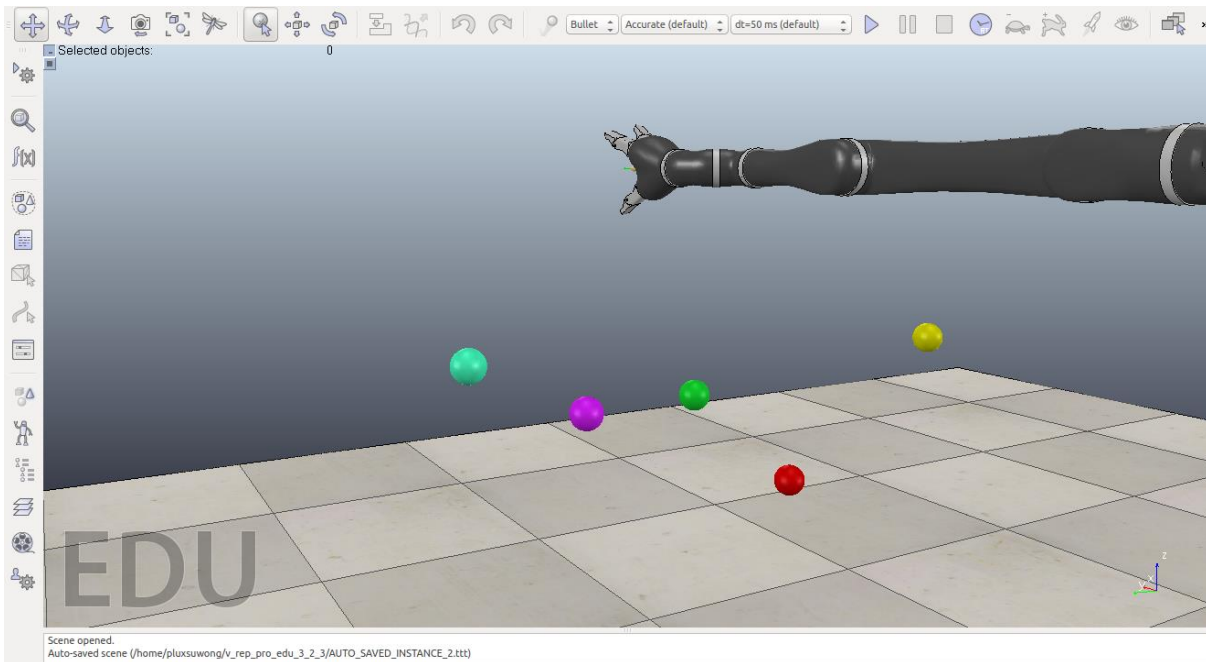
Figure 43: Scenario B in V-Rep

4   Results

The following section details the testing and results of the HERO Glove.

4.1   Curvature Sensor Performance

This section details the performance of the HERO Glove's light curvature sensor as a means to capture the user's hand gesture. As illustrated in the following two figures, the opening and closing of the user's hand gets mapped directly to the Robotiq Gripper's fingers, in the physical manipulator scenario, or the Jaco Arm end-effector, in the virtual manipulator scenario.
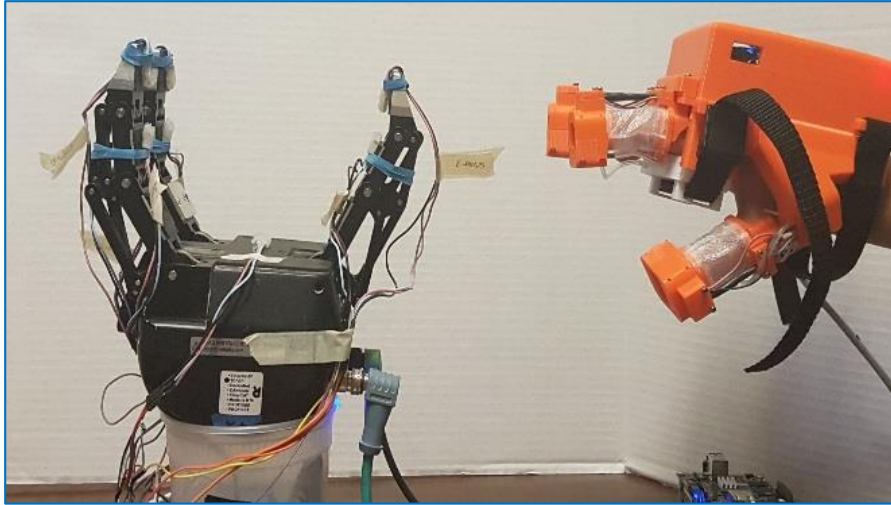
Figure 44: HERO Glove opening Robotiq Gripper
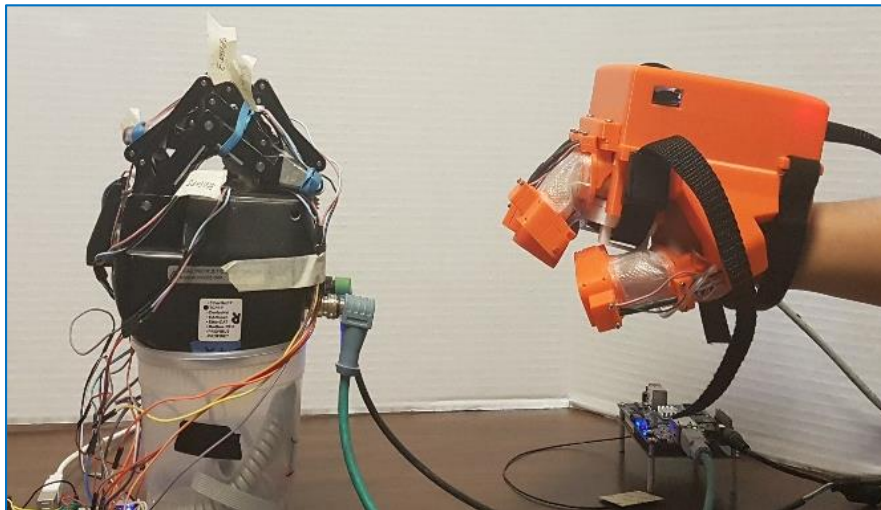
Figure 45: HERO Glove closing Robotiq Gripper

The accuracy and precision of the sensors were tested in greater detail using an oscilloscope to capture the voltage change. As the sensors produce an analog signal, its resolution is inherently infinite and is only limited by the ADC of the microcontroller. As shown in the following three figures, the workable range of the voltage is between approximately 0.307V to 2.07V at 3.3V operating voltage.
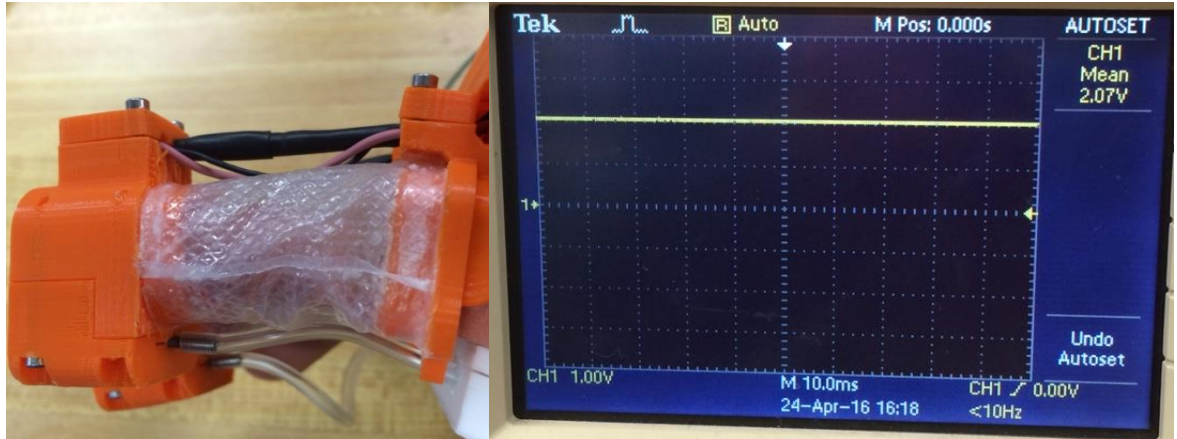
Figure 46: Curvature sensor reading at no bending
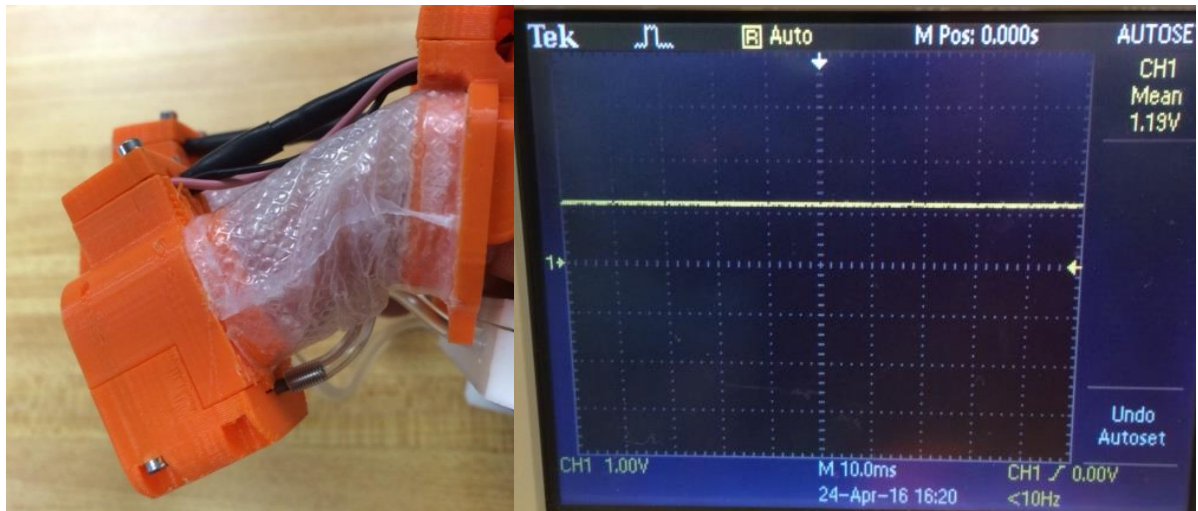
Figure 47: Curvature sensor reading at half bend
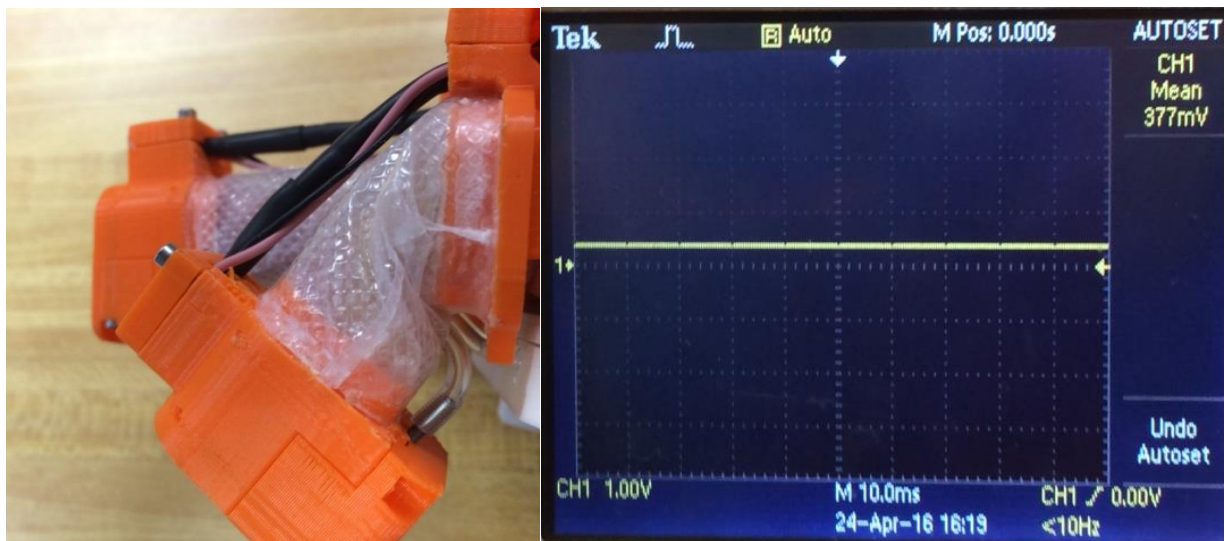
Figure 48: Curvature sensor reading at maximum bend

Additionally, the HERO Glove's curvature sensors were also tested in a virtual environment, as was specified in both Scenario A and Scenario B. Based on user feedback, 1 out of 14 users rated the curvature sensors' performance as being 3 out of 5, 4 users scored the system as 4 out of 5, and 9 users scored the system as 5 out of 5. This results in an average score of 4.57 for this category, indicating that the performance of the curvature sensors was satisfactory.

Finger Sensor Accuracy (14 responses)
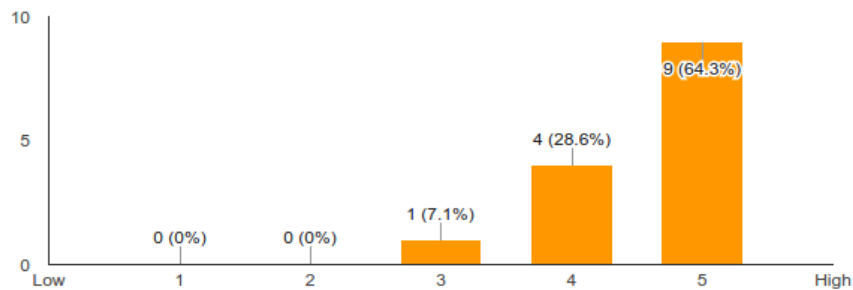
Figure 49: Curvature Sensor Performance Google Forms Results

4.2 Position Tracking Performance

Evaluating the HERO Glove's position tracking performance was primarily done through Scenario B. The users were tasked with controlling the robotic manipulator to grasp each of the floating balls, and throwing them in a desired direction. Predetermined metrics for measuring the HERO Glove's position tracking capabilities included the HERO Glove's end-effector pose stability, the HERO Glove's position accuracy and drift over time, and the HERO Glove's orientation accuracy and drift over time.

The HERO Glove ROS node published the HERO Glove's pose to the relevant ROS topics at an average rate of 12 Hz, as was measured by the ROS command-line tool, rostopic. The HERO Glove's end-effector pose was extremely stable, with any real-time fluctuations in the pose being objectively negligible, on the scale of tenths of a millimeter.

To evaluate the HERO Glove's positioning and orientation drift over time, the HERO Glove ROS node ran for over 3 hours while user testing was being performed. A comparison was made of the position and orientation of the HERO Glove prior to the user testing, and after the user testing. There was little to no effect on either the position or orientation due to drift over time.

Evaluation of the HERO Glove's positioning and orientation accuracy was done in a boolean manner, by deeming the position tracking system to either be accurate enough to allow the user to grasp an object in V-Rep, or not. With little to no training, all users were able to grasp at least 1 floating ball on their third runs of Scenario B. Out of the 14 users, 5 were able to grab 2 balls, 2 were able to grab 3 balls, and 1 user was able to grab 4 balls. Due to each user being able to control the Jaco Arm to grasp at least 1 ball, the HERO Glove's positioning was deemed to be

satisfactory. The HERO Glove's orientation was evaluated in a similar boolean manner. The orientation of the Jaco Arm end-effector was gauged to be either similar to that of the HERO Glove, or not. As was the case, the HERO Glove's orientation tracking was also deemed to be satisfactory.

After each user had completed three runs of Scenario B, he/she was required to fill out a survey scoring how accurate the position tracking was. 1 user out of 14 scored the HERO Glove's position tracking performance as being 3 out of 5, 11 users scored 4 out of 5, and 2 scored 5 out of 5. This result indicates the HERO Glove's average position tracking performance score to be 4.0.

Figure 50: Position Tracking Performance Google Forms Results

4.3 Glove-Robot Communication

The latency in the system was evaluated based on both empirical data collected through the use of computer networking tools and user based feedback. ICMP packets were used to measure the average round-trip time (RTT) in system. This measurement indicates the RTT for a packet sent out from a machine connected to the MinnowBoard MAX via ethernet, to the BeagleBone Black. The average RTT was measured to be about 190 ms.

User feedback was also provided regarding system latency. 2 users out of the 14 scored the latency in the system as 3 out of 5, with 9 users scoring 4 out of 5, and 3 users scoring 5 out of 5. This resulted in an average score of 4.1, indicating that the amount of latency in the system had little to no impact on the user's experience when handling the HERO Glove.

Latency (14 responses)
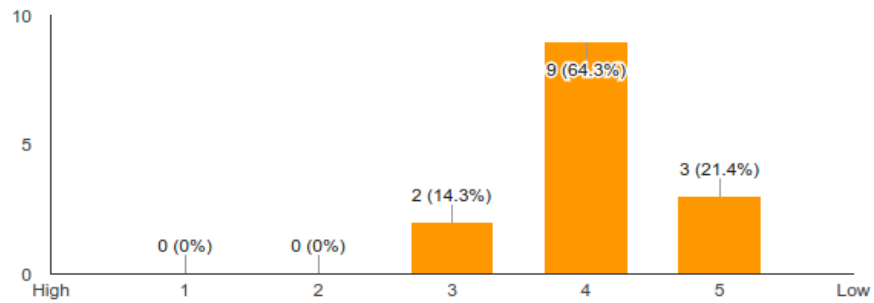
Figure 51: Glove-Robot Communication Google Forms Results

4.4 Haptic Feedback

The result of the haptic feedback of the finger modules can be seen in Figure 52. This shows the inflation of the finger joint actuators. The fingertip actuator inflated to occupy the space that the user's fingertip would be in. This pressed into the user's fingertip simulating the sensation of touch. The finger joint actuator also inflated occupying the space that the user's finger would be in. This squeezed the user's finger simulating the sensation of resistance to their finger joint. This demonstrated the functionality of the finger and thumb module actuators.
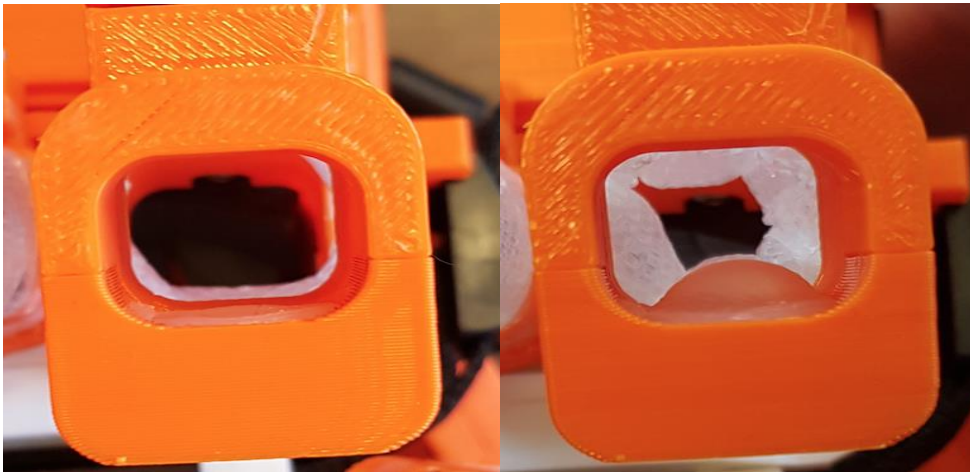
Figure 52: Inflated fingertip and finger joint actuators before (left) and after (right) inflation

Results from the palm touch actuator can be seen in Figure 53. As shown in the picture, the actuator inflated occupy the space where the user's palm would be in. This pressed into the user's hand giving the sensation of touch to the palm.
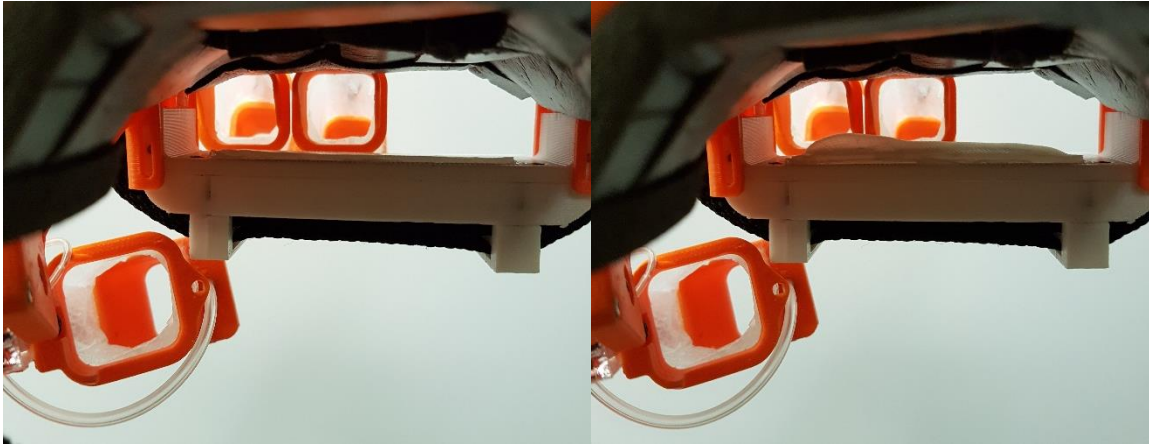
Figure 53: Palm touch actuator before (left) and after (right) inflation

After basic functionality testing of the HERO Glove, a large amount of testing occurred to determine the intuitiveness of the feedback it provided. This was because determining the results of haptic feedback was hard to quantify. Tests showed that response time of the actuators was fast and that the variance of pressure coincided with the amount of contact made with the robot. However, there was no straight forward way to pass or fail the haptic system. This led to human testing, similar to how the curvature sensors were evaluated.

The majority of the evaluation of the curvature sensors was performed in Scenario A. The actuators were all able to be inflated to a maximum of 5 psi when at 100% duty cycle, which provided enough of a physical response to the user's hand to simulate the sensation of both the fingertips and palm touching an object and the sensation of resistance to the bending of the user's finger when an object was in the grasp of the robot. The response was quick enough in the system to feel short bursts of impact when the robot's fingers hit into an object for an instant.

System tests were also performed using Scenario B. When compared to tests performed without force feedback it was very easy to determine whether an object was in contact with the robot's gripper.

Overall the performance of the haptic feedback system was excellent. The response was quick and the feeling was intuitive. Although the user could not determine the exact shape of the object in the gripper, it was very clear that there was an object in its grasp. There was also a clear distinction between contact made with the robot's fingertips and the rest of the robot's fingers as well as the robot's palm.

Once the users were done testing the HERO Glove, their feedback was obtained to allow for quantifiable data on the system's performance to be collected. Two metrics used to evaluate the system's haptic feedback: Finger Feedback Usefulness, and Finger Feedback Intuitiveness.

The Finger Feedback Usefulness metric was used to evaluate how much of a positive impact the haptic feedback had on the user's experience, when handling an object in V-Rep through the HERO Glove. 3 users out of 14 scored the HERO Glove's Finger Feedback

Usefulness as 4 out of 5, and 11 users scored 5 out of 5. This resulted in an average score of 4.79 for this metric, indicating that the haptic feedback was highly useful to the user.

Finger Feedback Usefulness (14 responses)
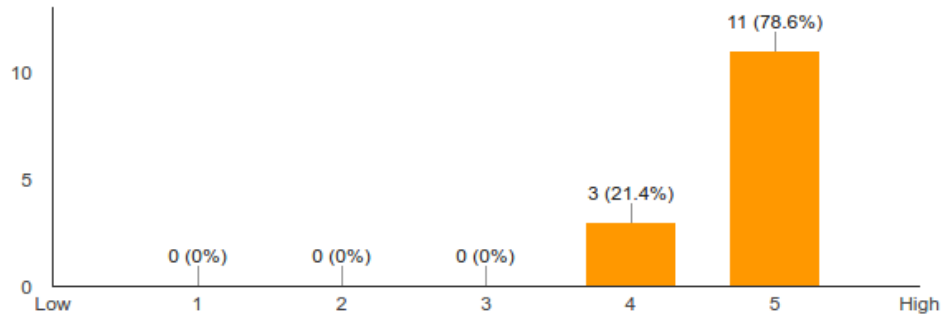
Figure 54: Haptic Feedback: Finger Feedback Usefulness Google Forms Results

The Finger Feedback Intuitiveness metric was used to evaluate how realistic the haptic feedback was in simulating that an object was currently being grasped. 4 users out of 14 scored this metric as 4 out of 5, while 10 users scored this metric as 5 out of 5. This resulted in an average score of 4.71, indicating that the haptic feedback provided by the HERO Glove was a good representation of how much force was being exerted by the controlled robotic end-effector on an object in its grasp.

Finger Feedback Intuitiveness (14 responses)
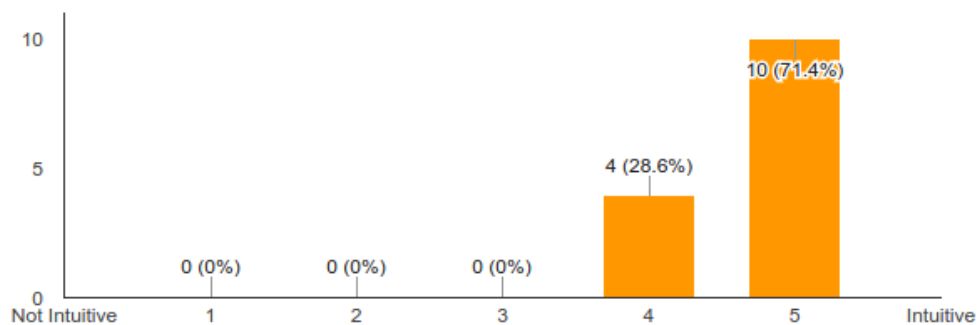
Figure 55: Haptic Feedback: Finger Feedback Usefulness Google Forms Results

There were few problems and issues with the haptic feedback actuators. In terms of durability the actuators never leaked throughout many hours of testing. The finger joint actuators were able to be inflated and the user could try as hard as they could to bend their fingers without any damage to the actuator. The same was true for the fingertip and palm actuators.

Although there were no significant issues with the actuators, the placement of the actuators was an issue for some users. In the final prototype, the location of the fingertip actuators was not in the correct spot for all users. Users with smaller or shorter fingers were often

unable to feel the feedback from the finger tips. The same went for the palm actuator depending on how tight the strap was on the HERO Glove. Although these did not serve any functional problems, the overall intuitiveness of the system was decreased for those who had smaller hands and fingers.

Another problem with the design of the actuators was the size of the thumb actuator. Due to the anatomy of the human thumb, the thumb module had to have a large opening to accommodate the widest part of the thumb. This meant that the actuator had to be much larger in width and height than the portion of the thumb that the actually felt the feedback. It also meant that the fingertip actuator was almost never in contact with most user's thumb tip. Another problem with the thumb actuators was the length of the thumb joint actuator. Due to the dimensions of the 3D printed components of the thumb module, and the short length of the thumb in comparison to the other fingers, the thumb joint actuator needed to be long enough for the thumb tip assembly to be past the center joint of the thumb. This caused the finger module to be too long for most user's thumb causing a bad fit to the user's thumb. In turn this provided less feedback than the other finger modules.

4.5 User Experience

The final of the metrics the HERO Glove was evaluated on, was the quality of the user's experience. This metric was evaluated purely on feedback obtained from the users who participated in the user performance evaluation of the HERO Glove. Three sub-metrics used in the evaluation of this category: Ease of Use, Comfortability, and Overall Feeling.

The Ease of Use sub-metric rated the HERO Glove with regards to the learning curve associated with using the HERO Glove in Scenario B. 1 of the 14 users scored 2 out of 5, 4 users scored 3 out of 5, 8 users scored 4 out of 5, and 1 user scored 5 out of 5. This resulted in an average score of 3.64 for this sub-metric. The relatively low score received for this metric indicates that controlling a robotic manipulator with the HERO Glove may entail a medium learning curve. However, the high scores in some of the other metrics, such as Position Tracking Performance imply that the low scores in User Experience may be due to depth perception being an inherent issue when using a 2-dimensional virtual environment to perform a 3-dimensional task, such as picking up an object.
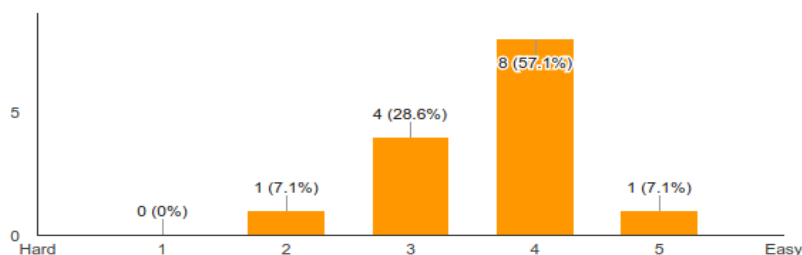
Figure 56: User Experience: Ease of Use Google Forms Results

The Comfortability sub-metric rated the HERO Glove with regards to how comfortable the HERO Glove was to wear and use over an extended period of time. 4 users scored 3 out of 5, 7 users scored 4 out of 5, and 3 users scored 5 out of 5. This resulted in an average score of 3.93 for this sub-metric. The mid-range score received for this metric is an indication that though the HERO Glove is not out-right uncomfortable to wear, future work can still be done to improve the comfortability of the HERO Glove.

Comfortability (14 responses)
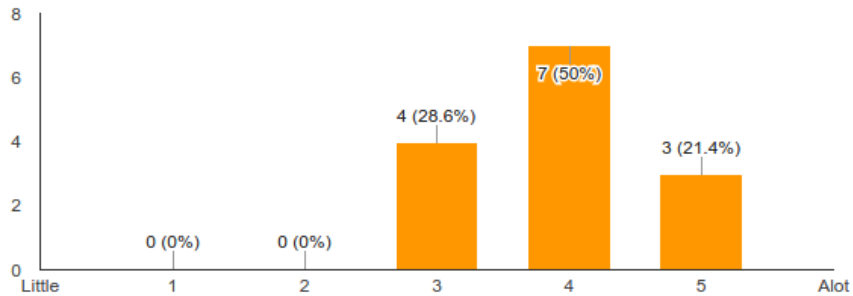
Figure 57: User Experience: Comfortability Google Forms Results

The Overall Feeling sub-metric was an attempt to gauge the user's overall opinion of the HERO Glove, after explaining the possible applications of the HERO Glove, and having had them use the HERO Glove. This metric rated the HERO Glove based off how close the users though the HERO Glove was to achieving its outlined potential applications. 7 users scored 4 out of 5, and 7 users scored 5 out of 5. This resulted in an average score of 4.5 for this sub-metric.

Overall Feeling (14 responses)
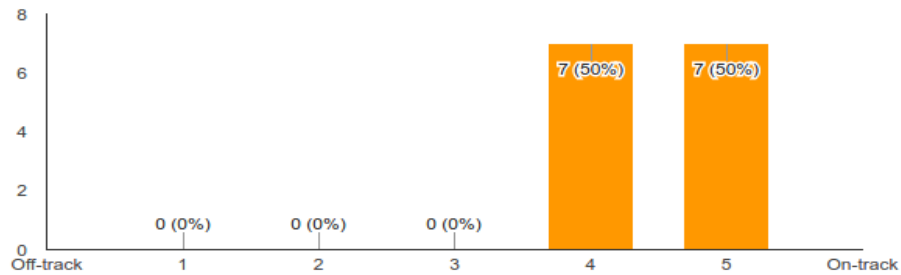
Figure 58: User Experience: Overall Feeling Google Forms Results

The Overall Feeling metric was somewhat subjective, and so was not formally included when calculating a single score to represent the overall score of the HERO Glove as a whole. However, it is included in the results for completeness, as it does provide some insight to the attitude of the users towards the HERO Glove as a project, at a higher level.

5 Recommendations

In this section, we describe recommended future works that should be done to improve the system. These are steps that we would have followed to improve our system, if time had permitted. We hope that this section would be useful for future researchers that see the potential in our system and want to refine it further.

5.1 Hardware

5.1.1 Mechanical Hardware

The mechanical design of the HERO Glove's shell had two main changes that we would recommend further design work. The first recommendations refer to the overall high level assembly of the glove. Due to the amount of electronic components needed for the glove's electrical system, the glove shell dimensions became fairly large and box like on top of the user's hand. The box like design was convenient for development and made it easy to install and remove components, however, the overall aesthetics of the glove were affected. We would recommend further design work into the packaging of the HERO Glove's components to allow for a sleeker and smaller glove shell design.

The final recommendation to the design of glove shell would be the complication of its geometry. The team had access to a 3D printed throughout the project so many components of the glove, especially the glove shell, were designed for 3D printing. This meant that there were features in the shell that would not be possible to be machined. Our recommendation would be to simplify the geometry of the glove shell to allow for the option of having the shell made without 3D printing. This would give further teams the option to produce a more accurately toleranced part.

5.1.2 Electrical Hardware

5.1.2.1 HERO Glove Motherboard Redesign

The current motherboard was designed for parallelism of processing, and modularity for ease of maintenance. However, this results in a large size which has the potential to be reduced. The first improvement that should be done to the system is redesigning the motherboard to reduce the amount of components. To do so, one can first start by removing the four Trinkets and transforming the current PCB into a break-out shield that fits on top of the BeagleBone Black. This is to use the BeagleBone's GPIOs to output PWM and input curvature sensor readings directly, making the BeagleBone the only processor on the HERO Glove itself. Eliminating the Trinkets can also remove the pull-up resistance issue reduces voltage on the lower range of the curvature sensors.

Even though eliminating the trinkets would already reduce the size of the PCB significantly, one can also switch current through-hole components to their surface mount equivalents. Examples of current through-hole components on the existing motherboard that can be switched to surface mount includes the op-amps, voltage regulator, and to a certain extent, the transistors.

5.1.2.2 Curvature Sensors

The latest iteration of the light-based curvature sensors that we developed and its signal conditioning circuitry has worked reliably and no longer pinches, which was a problem found in the first sensor iteration. However, improvements could be made to further reduce its size. From the mechanical perspective, the sensors could be improved to prevent it from bending in the opposite direction when the user extends their finger too much.

5.1.2.3 Robot Grip Force Sensors

The Robot Grip Force sensors that we manufactured and used in this system was designed in collaboration with the Frankenhand MQP, thus its shape was more suitable for the Frankenhand's prosthetic fingers. However, using the same concept of HES and magnet embedded in a silicone body can be further refined to better fit the larger finger surface of the Robotiq Gripper. One can also create an array of HES and magnets on one sensor with the possibly larger size, allowing for the sensor to detect the direction of force and not just measuring the normal force. Lastly, the sensors could be redesigned using a barometric pressure sensor embedded in silicone, where the force would be a result of the depression on the silicone body which changes the pressure reading as described in Jentoft et al 2013

5.2 Software

5.2.1 Oculus Rift Virtual Reality

In order to improve the user experience, an extended goal is to integrate the HERO Glove with the Oculus Rift Virtual Reality. In a real life scenario, if the robot is in a distance and the user can only see it through a camera, it could be difficult to perceive the depth perspective and thus the HERO Glove performance may be reduced. Therefore, it is necessary for the Oculus Rift to improve the user's perception.

5.2.2 Physical Robotic Manipulator

One of the HERO Glove's original goals was to use it to control a real-life serial robotic manipulator. The potential applications of the HERO Glove could be greatly expanded, should a physical robotic manipulator be integrated into the system. Integrating the Jaco Arm into the system could prove to be an excellent proof of concept, especially as it would provide a point of comparison against which to contrast the performance of the HERO Glove with the virtual Jaco Arm. However, to provide further proof for another of the HERO Glove's requirements, which was to be a general controller for robotic manipulators, integrating a different robotic manipulator into the system could prove to be just a great a boon.

5.2.3 Start-up Automation

Another possible improvement to the system that could be made in future iterations would be the automation of starting up and initializing the HERO Glove. Currently, different aspects of the HERO Glove must be manually started up manually by the user, through the use of a myriad of scripts and command-line tools. Having a single startup script or executable would benefit the user by greatly simplifying the startup process.

6 Conclusions

Realizing the need to fill the gap between commercially available open-loop robot teleoperation devices and precise, purpose-built haptic telepresence systems, we designed this system in which we named the haptic exoskeletal robot operator, or HERO Glove. The goal for the HERO Glove is to design a low-cost robot teleoperation system that provides realistic haptic feedback to deliver an intuitive user experience for the control of commercially available robotic manipulators. To accomplish this goal, we laid out three main objectives. First, the HERO Glove system must provide an intuitive user interface for the control of a remotely located robot, or telerobot. Second, the HERO Glove system must provide realistic haptic feedback to the user for enhanced telepresence experience. Third, the HERO Glove system must be modular enough for use with robotic manipulators available on the market.

In order to accomplish these systems objectives, we used a systems engineering approach to follow a methodological process for designing, integration, and testing of the HERO Glove. We started by defining the needs description of what the system should do after we have identified our stakeholders. Using the needs, we established a set of functional system requirements that the system must do for validation and verification. After so were we then able to lay out the high level systems diagram and then delve into greater engineering design.

The final HERO Glove design consists of two main functionalities; capturing the user's hand and arm position to control a robot manipulator, and provide haptic feedback to the user to replicate the sensation of grasp and touch the robot exerts on a remotely-located object. ROS was used as the main software framework to provide communication between HERO Glove and commercially available industrial robot manipulators such as the Jaco Arm and the Robotiq 3-Finger gripper.

To test the functionality of our system, we analyzed each engineering aspect of the HERO Glove to assess whether it has achieved the requirements we state in the functional system requirements for validation and verification purposes. We developed a set of user performance evaluations in which we invited students external to our project to test our system's operability requirements and allow for us to evaluate qualitatively our system performance. Based on our quantitative results and user testing evaluations, we were able to conclude that our system was able to achieve the objectives set forth at the beginning of the project. First of all, the HERO Glove was able to accurately and precisely capture the user's hand gesture and arm pose enough to allow untrained users to successfully grab objects within the Jaco Arm's workspace in V-Rep simulation. Secondly, the user was able to feel the haptic feedback provided by the HERO Glove well enough that they felt were able to control and maintain a grasp on the object and knowing when the object has been released. Lastly, by using ROS, the HERO Glove was able to communicate with and control various types of robotic manipulators.

Works Cited

Adams, Richard J., and Blake Hannaford. "Stable haptic interaction with virtual environments."

Robotics and Automation, IEEE Transactions on 15.3 (1999): 465-474.

"All About Robotic Surgery." *Force Feedback, Haptics, Tactile Feedback*. N.p., 1 Sept. 2014.

Web. 10 Sept. 2015.

Bouzit, Mourad, et al. "The Rutgers Master II-new design force-feedback glove."*Mechatronics,*

IEEE/ASME Transactions on 7.2 (2002): 256-263.

Cui, Jiahong, Sabri Tosunoglu, Rodney Roberts, Carl Moore, and Daniel W. Repperger. *A*

REVIEW OF TELEOPERATION SYSTEM CONTROL. N.p.: FCRAR, 8 May 2003. PDF.

Dzieza, John. "Behind the Scenes at the Final DARPA Robotics Challenge." *The Verge*. N.p., 12

June 2015. Web. 10 Sept. 2015.

Fagella, Daniel. ""Sensitive" Robotics Isn't What You Think – with WPI's Eduardo Torres-Jara,

CEO of Robot Rebuilt." *TechEmergencecom*. N.p., 05 Dec. 2013. Web. 24 Apr. 2016.

"Fiber-Reinforced Actuators." *Fiber-Reinforced Actuators*. N.p., n.d. Web. 19 Sept. 2015.

Hammond, Frank L., Yigit Menguc, and Robert J. Wood. "Toward a modular soft sensor

embedded glove for human hand motion and tactile pressure measurement." Intelligent

Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on. IEEE,

2014.

Huyghe, Benoit, Jan Doutreloigne, and Jan Vanfleteren. "3D Orientation Tracking Based on

Unscented Kalman Filtering of Accelerometer and Magnetometer Data." *2009 IEEE*

Sensors Applications Symposium. Print.

Luo, Ming, Yixiao Pan, Erik H. Skorina, Weijia Tao, Fuchen Chen, Selim Ozel, and Cagdas D.

Onal. "Slithering towards Autonomy: A Self-contained Soft Robotic Snake Platform with

Integrated Curvature Sensing." Bioinspir. Biomim. Bioinspiration & Biomimetics 10.5

(2015): 055001. Web.

Madgwick, S. (2013, November 3). Oscillatory Motion Tracking with x-IMU | x-io

    Technologies. Retrieved December 18, 2015, from http://www.x-io.co.uk/oscillatory-

    motion-tracking-with-x-imu/

Madgwick, Sebastian OH. "An efficient orientation filter for inertial and inertial/magnetic sensor

    arrays." *Report x-io and University of Bristol (UK)*(2010).

Mavridis, Nikolaos, et al. "On the subjective difficulty of Joystick-based robot arm teleoperation

    with auditory feedback." *GCC Conference and Exhibition (GCCCE), 2015 IEEE 8th*.

    IEEE, 2015.

McAlpine, Kat J. "A New Grasp on Robotic Glove." *Harvard Gazette*. Wyss Institute

    Communications, 5 June 2015. Web. 23 Apr. 2016.

Okamura, A. M. *Methods for Haptic Feedback in Teleoperated Robot-assisted Surgery*. N.p.:

    Emerald Insight, 20 Dec. 2005. PDF.

Perkins, M. (2013, November 13). A C class to implement low-pass, high-pass, and band-pass

    filters | Cardinal Peak. Retrieved December 18, 2015, from

    https://cardinalpeak.com/blog/a-c-class-to-implement-low-pass-high-pass-and-band-pass-

    filters/

Quick, Darren. "Long Distance Tele-Operation System for Remote Control of Unmanned

    Ground Vehicles." Long Distance Tele-Operation System for Remote Control of

    Unmanned Ground Vehicles. Gizmag, 3 Aug. 2011. Web. 10 Sept. 2015.

Zhu, R., and Z. Zhou. "A Real-Time Articulated Human Motion Tracking Using Tri-Axis

    Inertial/Magnetic Sensors Package." IEEE Trans. Neural Syst. Rehabil. Eng. IEEE

Transactions on Neural Systems and Rehabilitation Engineering: 295-302. Print.

Appendix A Setup Instructions

MinnowBoard MAX

1. Controlling Jaco Arm in VREP

 A. Connect the MinnowBoard MAX to your computer with an Ethernet cable

 B. Connect to the MinnowBoard MAX
- From your computer, open a new terminal and connect to the MinnowBoard MAX using the command:

  `ssh hero@192.168.1.3`

- Type in the password: password

```
tkkhuu@TKhuu-Ubuntu: ~
tkkhuu@TKhuu-Ubuntu:~$ ssh hero@192.168.1.3
hero@192.168.1.3's password:
```
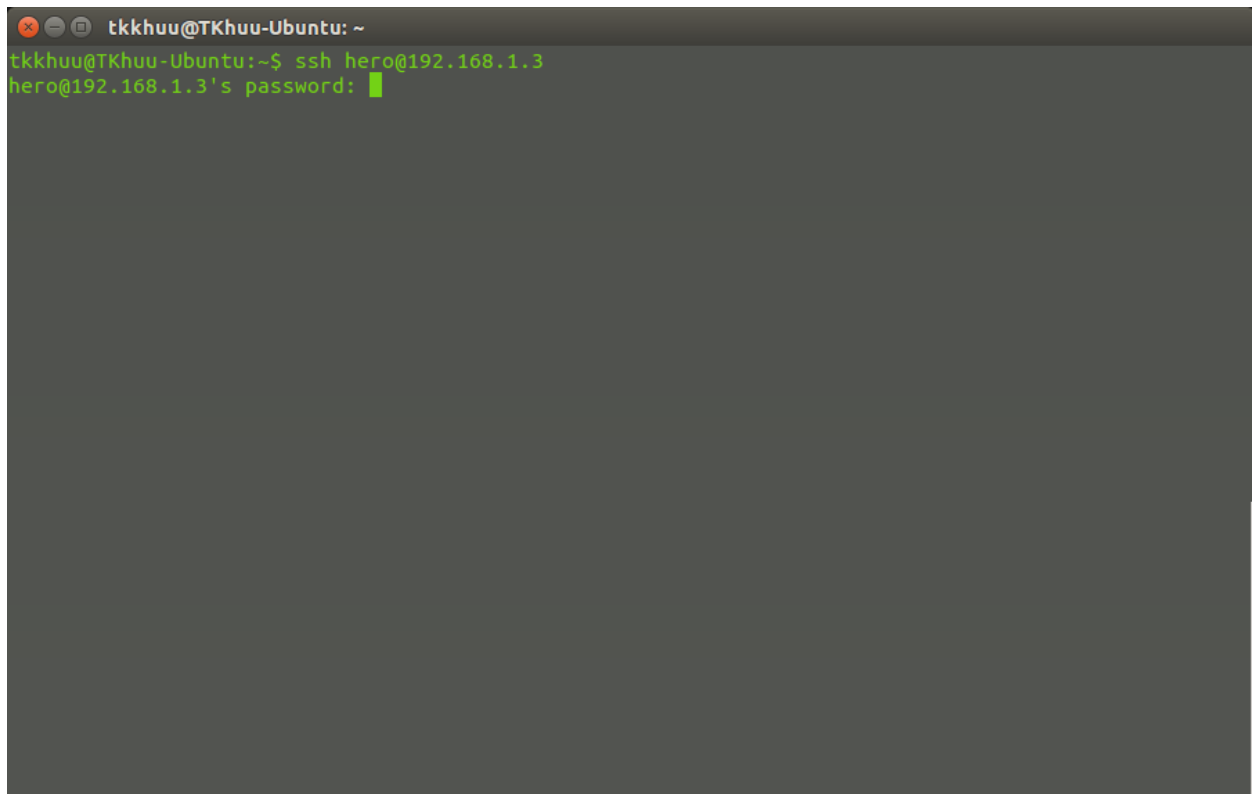
 C. Running `roscore`
- In the .bashrc file, make sure the ROS_MASTER_URI and ROS_IP are set to 192.168.134.101. The .bashrc file can be opened by:
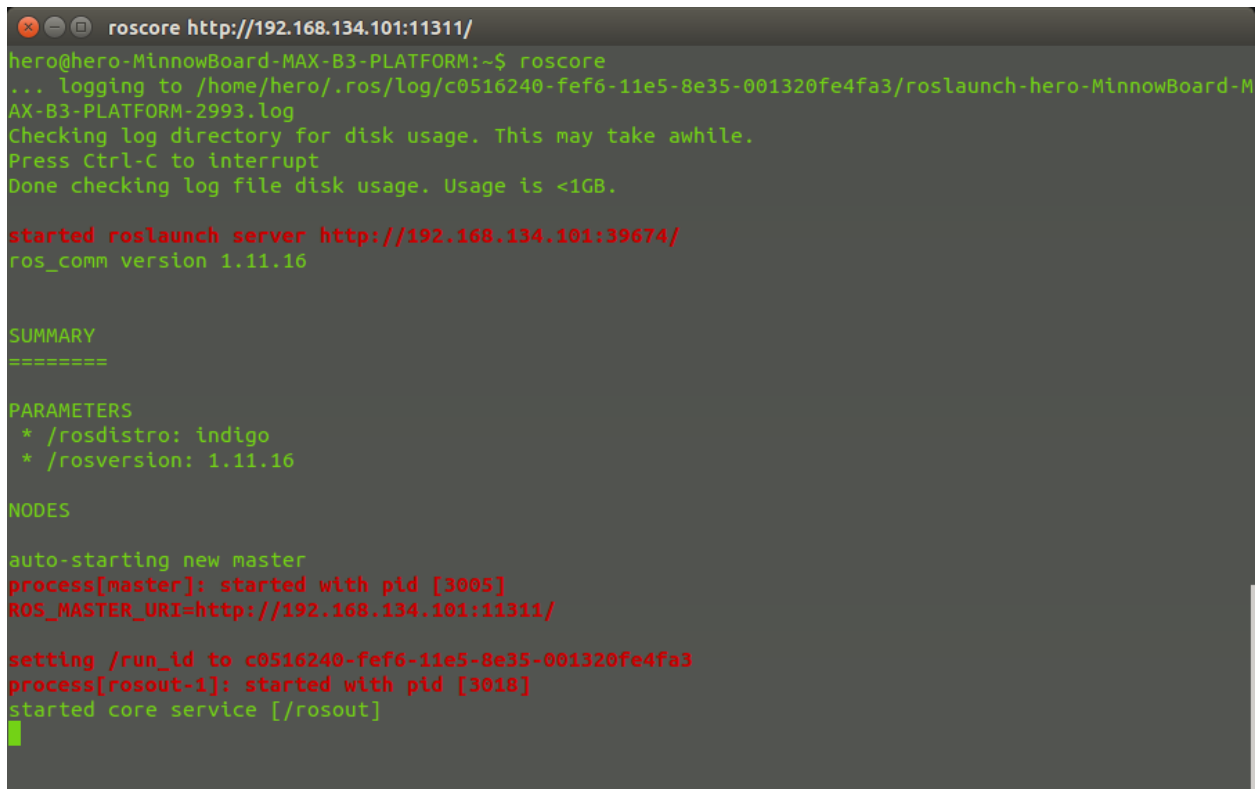
  `vim ~/.bashrc`
- Make sure the following lines are at the end of the file:

```
export ROS_MASTER_URI=http://192.168.134.101:11311/
export ROS_IP=192.168.134.103
```

- Start roscore with the following command:

```
roscore
```


2. Controlling Robotiq Gripper

A. Connect the Robotiq gripper to the Ethernet port of the MinnowBoard Max. You should be able to see the blue LED is on, this indicates that Robotiq is powered. The yellow LED should be flashing, indicating that there are connections found but it has not been connected.

B. Connect to MinnowBoard MAX
- From your computer, open a new terminal and connect to the MinnowBoard MAX using the command:

```
ssh hero@192.168.1.3
```

- Type in password: password
- Then run the script in the home directory:

```
./enable_robotiq.sh
```

C. Running `roscore`
- In the .bashrc file, make sure the ROS_MASTER_URI and ROS_IP are set to 192.168.134.101. In the same terminal, run:

```
vim ~/.bashrc
```

- Make sure the the following lines are at the end of the file:

```
export ROS_IP=192.168.134.101
export ROS_MASTER_URI=http://192.168.134.101:11311/
```

- Start roscore by the following command:

```
roscore
```


D. Running the Robotiq Driver

- First open a new terminal on the MinnowBoard MAX. Open a new terminal on your computer and then type in the command:

  ```
  ssh hero@192.168.1.3
  ```
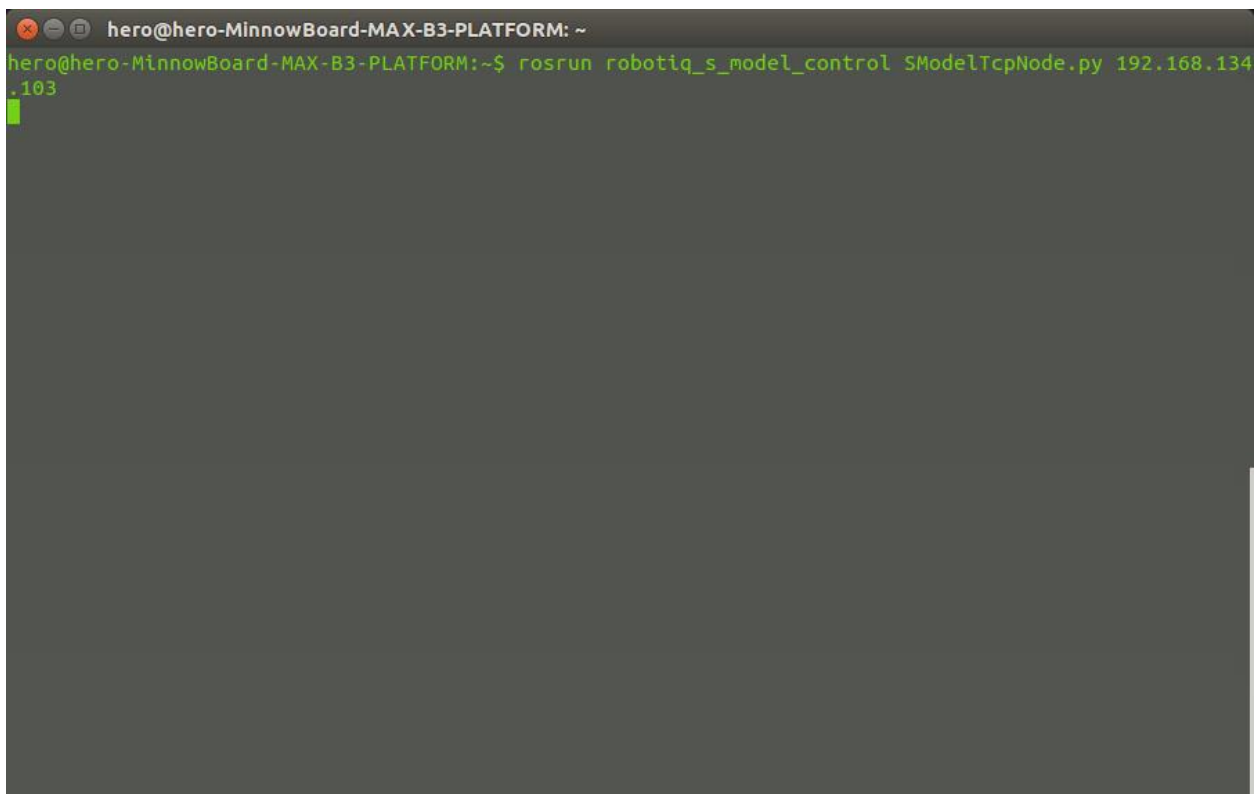
- Type in the password: password
- Run the Robotiq driver with the following command:

  ```
  rosrun robotiq_s_model_control SModelTcpNode.py        <Robotiq
  Gripper's IP>
  ```

- In our case, the Robotiq static IP address is 192.168.134.103 so we would run:

  ```
  rosrun robotiq_s_model_control SModelTcpNode.py 192.168.134.103
  ```


E. Run the Robotiq Hero Controller node

- In a new terminal of the MinnowBoard MAX, type the command:

  ```
  rosrun robotiq_hero_controller robotiq_hero_controller.py
  ```

- If successful, the Robotiq will start its initialization and finish with a close position. It is now subscribing to the topics:
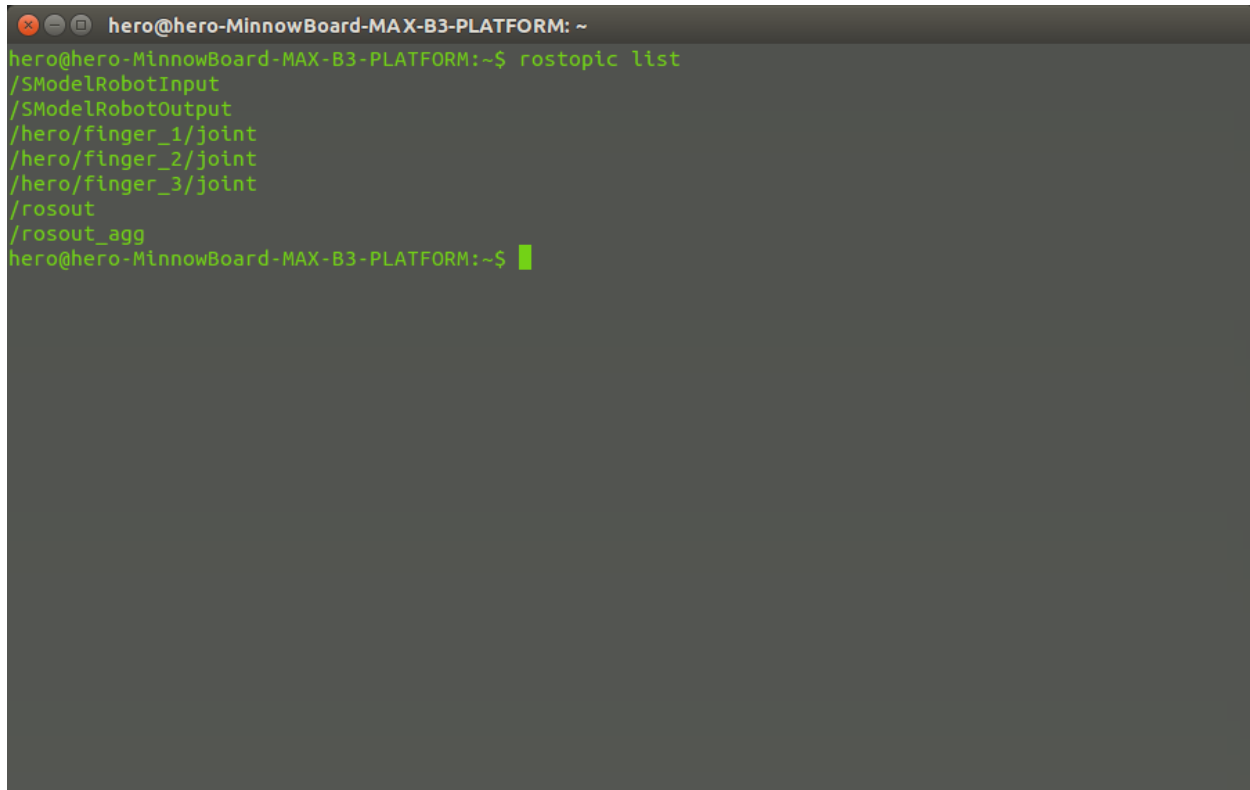
  /hero/finger_1/joint
  /hero/finger_2/joint
  /hero/finger_3/joint
- This can be verified by running:

  `rostopic list`

```
hero@hero-MinnowBoard-MAX-B3-PLATFORM: ~
hero@hero-MinnowBoard-MAX-B3-PLATFORM:~$ rostopic list
/SModelRobotInput
/SModelRobotOutput
/hero/finger_1/joint
/hero/finger_2/joint
/hero/finger_3/joint
/rosout
/rosout_agg
hero@hero-MinnowBoard-MAX-B3-PLATFORM:~$
```

VREP Simulation

1) Make sure to install VREP simulation on your computer as well as the VREP ROS package. Connect the computer to the MinnowBoard MAX with an Ethernet cable.

2) Set the Ethernet static IP of your computer to "192.168.134.103".

3) Running VREP
    1) Then make sure in the ".bashrc" file ROS_MASTER_URI is set to 192.168.134.101. The .bashrc file can be opened by the command:

        ```
        vim ~/.bashrc
        ```

    - At the end of the file, make sure the following lines are added:

        ```
        export ROS_MASTER_URI=http://192.168.134.101:11311/
        export ROS_IP=192.168.134.103
        ```

    - Navigate to the folder where VREP is installed and launch VREP:

        ```
        ./vrep.sh
        ```

    - Download the Jaco Arm scene from the HERO Glove repository.

    - Bring up the Jaco Arm by opening the scene Jaco_User_Test_S_Balls.ttt.
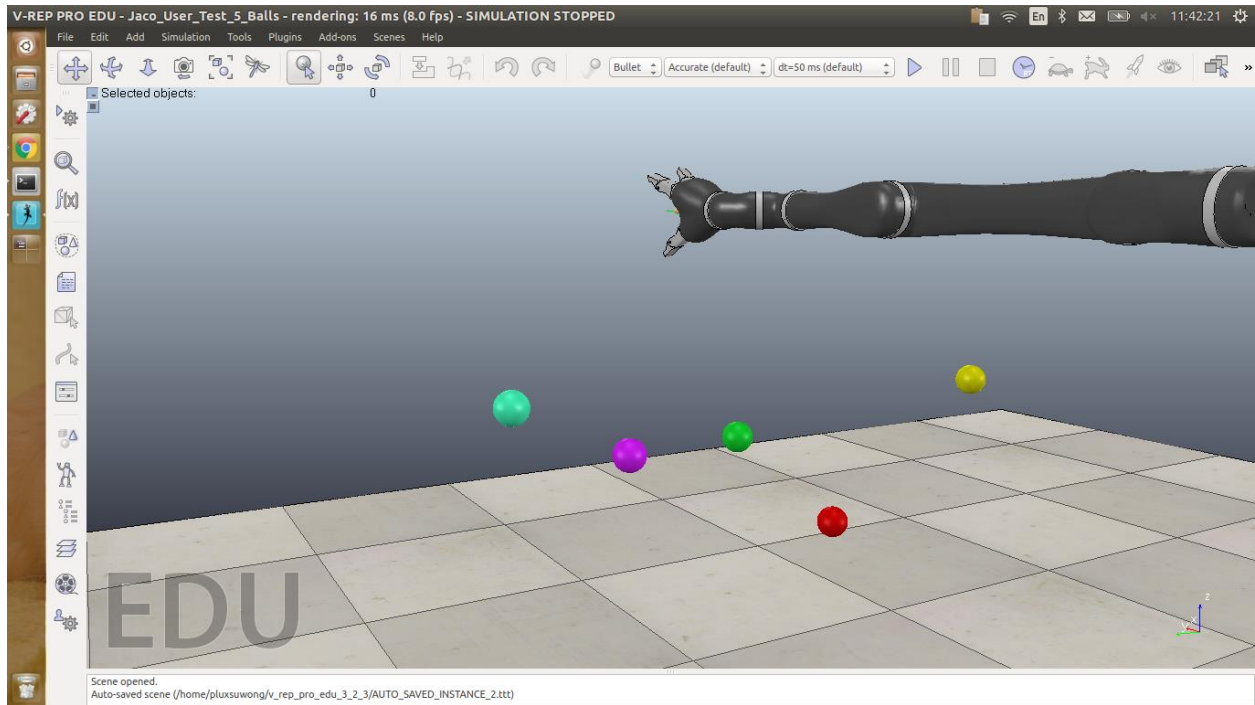
Robotiq Gripper

1) In order for the MinnowBoard MAX to send commands to the Robotiq gripper, the Robotiq ROS Driver has to be installed. Make sure to run enable_wifi.sh on your computer and pkt_fwd.sh on the MinnowBoard MAX access to grant Internet access to the MinnowBoard MAX through your computer.

2) Connect to MinnowBoard MAX
   - Plugin the WI-FI module to the MinnowBoard MAX. Open a new terminal on your computer and connect to it by entering the command:

   `ssh hero@192.168.1.3`
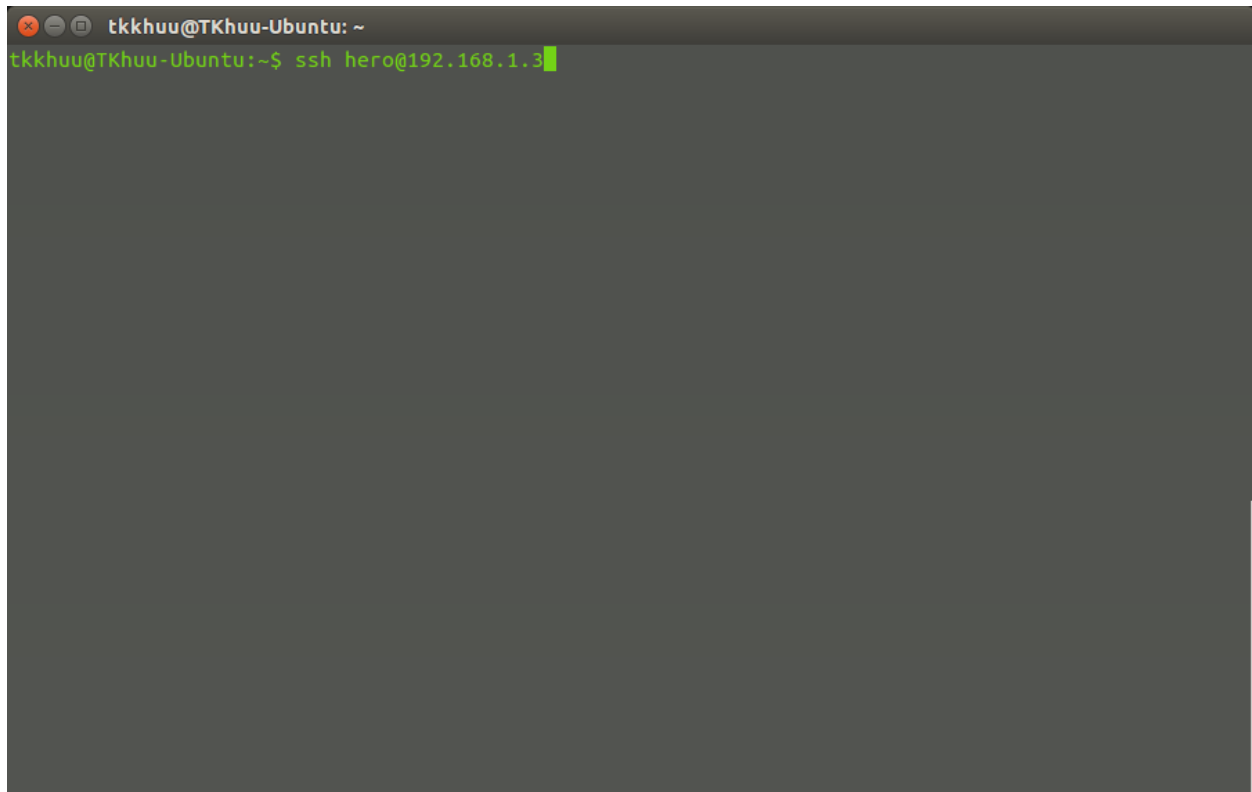
   - Type in password: password

```
tkkhuu@TKhuu-Ubuntu: ~
tkkhuu@TKhuu-Ubuntu:~$ ssh hero@192.168.1.3
hero@192.168.1.3's password:
```

3) Download the Robotiq ROS Driver

- Navigate to the catkin workspace on the MinnowBoard MAX:

    ```
    cd ~/catkin_ws/src
    ```

- Download the Robotiq ROS repository:
    ```
    git clone https://github.com/ros-industrial/robotiq.git
    ```

- Make sure to check out the correct ROS distribution branch, in this case is indigo-devel:
    ```
    cd ~/catkin_ws/src/robotiq
    git checkout indigo_devel
    ```

- The Robotiq packages depends on "Modbus TCP" and "EtherCAT soem", install them by running:

    ```
    rosdep install robotiq_modbus_tcp
    sudo apt-get install ros-indigo-some
    ```

4) Compile the package

- Navigate to the ROS catkin workspace and compile the Robotiq package:
  ```
  cd ~/catkin_ws
  catkin_make
  ```

BeagleBone Black

A. Controlling Jaco Arm in VREP

1) First connect to the BeagleBone Black through an Ethernet. From a new terminal on your computer, connect to the BeagleBone Black by running:

   ssh root@192.168.10.4
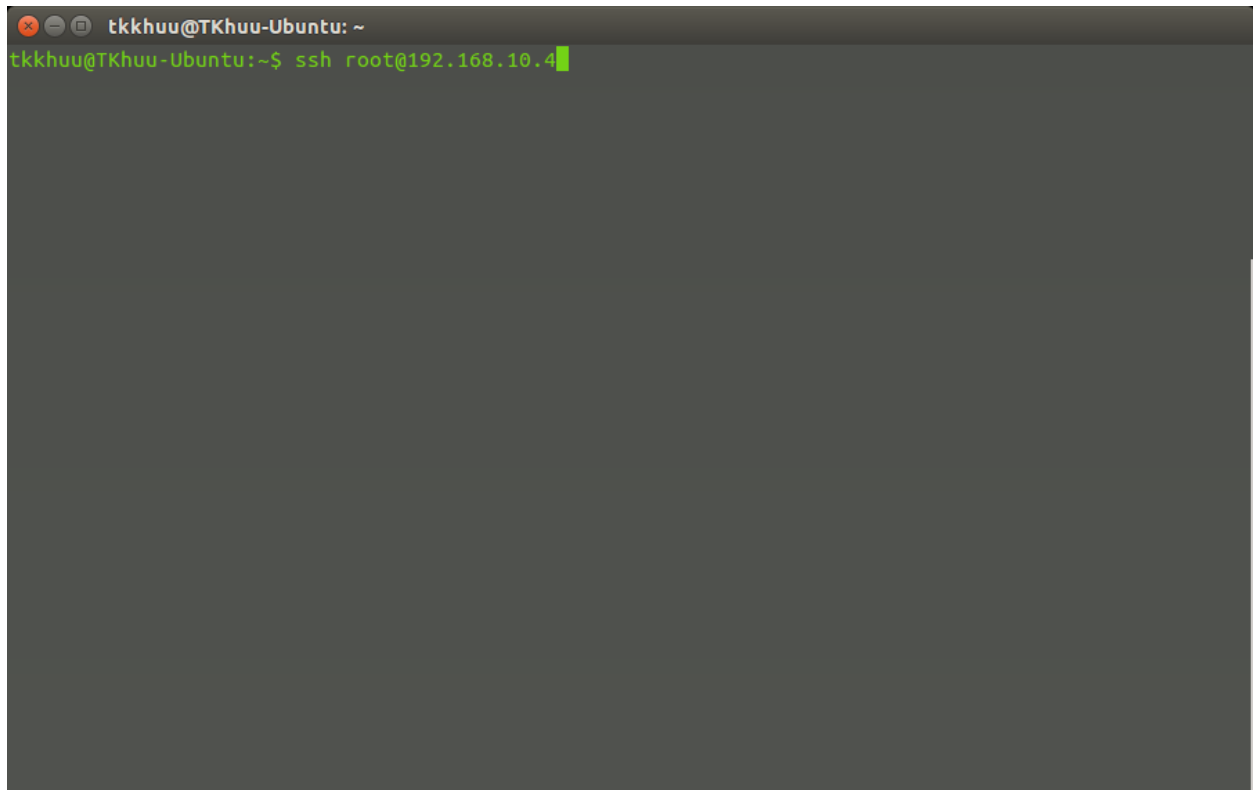
2) Run the following script in the home directory to enable the WI-FI dongle on the BeagleBone Black:

   ./wifi_startup.sh

3) Then make sure the following lines in the ~/.bashrc file are at the end of the file:

   export ROS_MASTER_URI=http://192.168.134.101:11311/
   export ROS_IP=192.168.1.4

```
😣⊖ ▢   tkkhuu@TKhuu-Ubuntu: ~
# Some more alias to avoid making mistakes:
# alias rm='rm -i'
# alias cp='cp -i'
# alias mv='mv -i'

export TERM='xterm-256color'

source /opt/ros/indigo/setup.bash
source ~/ros_catkin_ws/devel/setup.bash
█

# Adhoc WLAN with Robotiq
export ROS_MASTER_URI=http://192.168.134.101:11311/
export ROS_IP=192.168.1.4


export ROS_PACKAGE_PATH=/root/ros_catkin_ws:$ROS_PACKAGE_PATH
~
~
~
~
~
~
~
~
~
~
~
                                               24,0-1          Bot
```

4) Run the hero_glove program:

```
rosrun hero_glove hero_glove
```

B. Controlling Robotiq Gripper

1) First connect to the BeagleBone Black. From a new terminal on your computer, connect to the BeagleBone Black by running:

    `ssh root@192.168.10.4`

2) Run the following script in the home directory to enable the WI-FI dongle on the BeagleBone Black:

    `./wifi_startup.sh`

- Wait a few seconds then run the script:

    `./enable_robotiq.sh`

- Then make sure the following lines in the ~/.bashrc file are at the end of the file:

    `export ROS_MASTER_URI=http://192.168.134.101:11311/`
    `export ROS_IP=192.168.1.4`

```
# Some more alias to avoid making mistakes:
# alias rm='rm -i'
# alias cp='cp -i'
# alias mv='mv -i'

export TERM='xterm-256color'

source /opt/ros/indigo/setup.bash
source ~/ros_catkin_ws/devel/setup.bash


# Adhoc WLAN with Robotiq
export ROS_MASTER_URI=http://192.168.134.101:11311/
export ROS_IP=192.168.1.4


export ROS_PACKAGE_PATH=/root/ros_catkin_ws:$ROS_PACKAGE_PATH
~
~
~
~
~
~
~
~
~
~
~
                                               24,0-1         Bot
```

- Run the hero_glove program:

```
rosrun hero_glove hero_glove
```

Appendix B Use Case: Robot Opening a Door

Use Case Identifier: SS_01

Use Case Name: Using glove to control robot to open a door

Primary Actor: Haptic Feedback Glove user/operator

Participating Actors: Co-operator who monitors the primary actor

Initial Conditions: There is currently a fire in a building and there are people stuck inside it that need to be rescued. The robot is sent to the control site and traverses until it reaches the door that needs to be opened. The operator can see the robot and its field of vision through a monitor.

UC Description:

I.   User puts on glove and run through the calibration process

II.   User can see the robotic manipulator he or she is controlling in real time through a camera vision system for visual feedback

III. User moves the glove around freely while observing the robot matching the user's motions. The user could also test the functionality of the end effector by bending fingers and rotating wrist.

IV.   User controls the end effector towards the door knob by moving the glove in the desired direction.

V.   Once the robot makes contact with the door knob, the pneumatic actuators in the glove's fingertips will inflate giving the user the sensation that they are touching the door knob.

VI.   Knowing that they have made contact, the user would then control the robot manipulator to grab the door knob by making a grasping motion with their hands.

VII. User could then feel resistance in their finger joints simulating the sensation of grasping the solid door knob

VIII. User then rotates their wrist which controls the manipulator's end effector twisting the door knob.

IX.   User then control the robot to push or pull the door to open while maintaining the grasping of the door knob from force and tactile feedback.

X.   If the range of the manipulator was larger than that of the user, the user could hold down the RESET button, move their hand into a comfortable position, release the RESET button, and continue their desired operation.

XI.   If user powers the glove off, or if the glove loses communication with the manipulator, the manipulator would remain in its last position.

Alternatives:

- (V. alternative) glove provides vibrational feedback instead of pneumatic actuator inflating to allow for faster response time.
- (IX. Alternative) glove provides vibrational feedback once the manipulator has extended to its maximum range of motion, resulting in singularity.
- (XI. Alternative) robot manipulator continues the action autonomously using intelligence algorithm until mission is complete.

Exit Conditions:

- The door is successfully opened under one minute
- The robot is able to continue the mission without damaging its environment.

Needs/Requirements Discovered:

1. There should be a hold function that allows for the user to re-orientate the glove to a comfortable position without affecting the movement of the slave robot.
2. The glove should only be powered on/transmit data to the slave robot when it detects that there is a hand in the glove to avoid unintended operation.
3. Need position tracking on the glove to move robot manipulator closer to the target object. The glove needs to move in free space. Kinesthetic (tactile and force) feedback is not enough for user to accurately control the robot to execute manipulation tasks.

Models/Studies Needed:

1. Model/study worst case latency allowable in the system. How much latency could the system allow that does not contribute to user frustration and user error.

Model/study what pressure range must we need to replicate the sensation without overwhelming (too sensitive) and too little (not sensitive enough).

Appendix C Prototype 1 Mechanical Design and Results

Design:

Top Level Assembly Design

The overall design of the glove consisted of 4 main sub-assemblies. The first sub-assembly was a 3D printed shell that would be strapped onto the user's hand. The shell would include components on the palm of the user to house the palm touch actuators. The top of the hand would house pneumatic valves and various electrical components. There would also be a wrist component that would be attached to the top of the hand shell. This would house the battery and the main micro controller of the glove. The shell would be strapped around the user's hand underneath the knuckles and again around the user's wrist. The top of the shell would also have some way to connect the finger sub-assemblies to the glove. These connections would allow the fingers to move freely in all directions.

The three remaining sub-assemblies would be for the fingers. One subassembly was designed for the thumb and two identical sub-assemblies would be designed for the middle and index finger. The thumb and middle/index finger sub-assemblies would be similar, however, the thumb sub-assembly would be larger in diameter and shorter in length to accommodate the anatomy of the thumb. These subassemblies would include a solid ring that would slide all the way down the finger and connect to the hand shell. They would also include a hard shell to slide onto the last digit of the finger. This would house the fingertip touch actuator. Between the two rings would be a soft actuator to provide resistance to the middle joint of the finger. Finally, a curvature sensor would be imbedded into the actuator so that the bending of the finger could be measured. An initial concept of the full assembly of the glove can be seen in Figure 59.
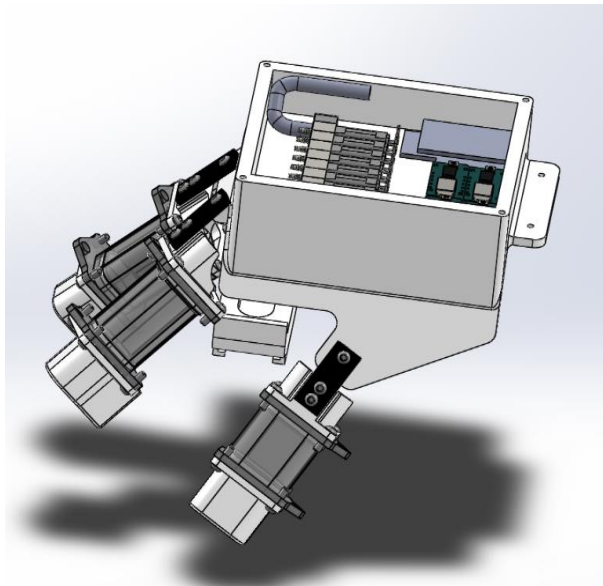
Figure 59: Top Level Assembly

Component CAD

A full model of the glove in Solidworks was created. It included components to be strapped onto the hand, as well as two finger actuator sub-assemblies for the pointer and middle finger, and a finger sub-assembly for the thumb. The top level assembly of the glove will be updated as the project goes on.

Actuator Concepts

A crucial functionality of the HERO Glove is for the user to experience intuitive feedback. This feedback would interact with the user's hand on the middle joint of the finger, the fingertip, and the palm of the hand. In order to achieve this feedback, we prototyped soft pneumatic actuators using several different methods.

Finger Joint Feedback Actuators

Four approaches were prototyped and tested to create a pneumatic actuator to provide feedback to the middle joint of the user's finger. The first two attempts were similar in the approach because they both were molded silicone toroids. The third attempt used a different approach by using non-permeable materials. The final attempt that proved successful was the use of vacuum sealable plastic. The following sections describe our process for designing, manufacturing, and testing of the actuators.

Silicone Actuators

Cylindrical Toroid Actuator:

Our first method of providing feedback to the middle joint of the finger used a silicone rubber toroid that would be slid around the finger. To begin the process, we first designed the geometry of the toroid in Solidworks. Once the geometry was finalized, we designed a mold, also in Solidworks. The mold was designed by taking the negative geometry of the toroid. The first toroid was a cylindrical shape with an inner air pocket and a hole through the middle for the user's finger to slide through, seen in Figure 60. To mold this shape, we designed an outer base to create the outer dimensions of the toroid. Next, an insert was made to fit in the mold base to create the air pocket and center through hole. The mold was then designed in a way so that the dried silicone could easily be removed from the mold. This was done by cutting the mold base in half and allowing the two halves to be screwed together. That way, when the silicone was dried the insert could be removed from the top and the two base pieces could be separated. Figure 61 represents the first toroid mold. The mold was then 3D printed. To mold the silicone
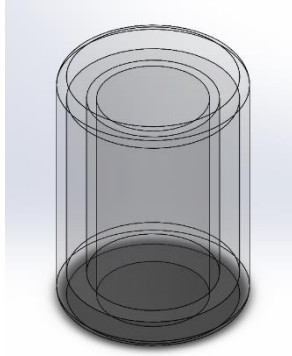
Figure 60: Cylindrical Toroid Model
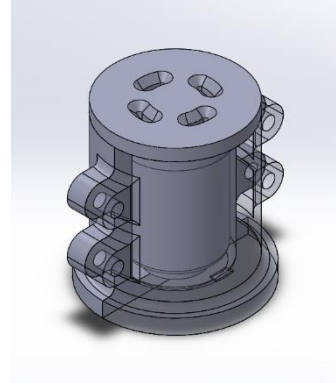
Figure 61: Cylindrical Toroid Mold

toroid, we mixed Silicone Rubber ECOFLEX 0030 and allowed to sit in a vacuum chamber for several minutes to allow air bubbles to exit the silicone. The silicone was then poured into the mold and the mold insert was inserted. The mold was left to dry for 4 hours. From this process we were left with an open-ended toroid.

In order to cap the end, silicone was mixed, as described above, and poured up to an indicated height in the mold base. The height to which the silicone was to be poured was calculated by taking the volume of end cap. The volume was provided from Solidworks. We then used the value of the volume to determine the height up the mold base needed based off the radius of the mold base. The open-ended toroid molded in the previous molding step was then inserted into the base of the mold. It was suspended so that the end of the dried silicone piece made contact with the viscous silicone. It was then left to dry for 4 hours. The result was a completed toroid, seen in Figure 60. To allow for air to be inputted into the toroid, a pneumatic tube was inserted and hot glued in place. We then tested the actuator by connecting the pneumatic tube to a syringe. By compressing the syringe we were able to pressurize the toroid.

Rectangular Toroid Actuator:

Further attempts were made following the results of the first cylindrical toroid. From the first attempts we found the size of the actuator to take up too much room around the finger. The diameter of the toroid was 33 mm. Due to this size, if the user was wearing a finger actuator on both their pointer and middle finger, the actuators would make contact with each other. This would take away the ability of the two fingers to move freely. To solve this in further attempts of creating toroid, we decided to make the toroid rectangular in shape. However, the width of the toroid would be less than the height of the toroid. This allowed for more room in between the fingers while still allowing a substantial amount of area for the air pocket. We designed the toroid to be 28mm in width and 34 mm in height, see Figure 64.

The team made a mold for the rectangular shaped silicone toroid. All attempts repeated the same process mentioned above, however, the design of the mold was altered from attempt to attempt. The first attempt of the mold comprised of two flat rectangular molds. The silicone

would be poured into the mold and allowed to dry into a solid sheet. The sheet would then be wrapped around a guide. The two ends of the sheet would be bonded with silicone and left to bond together. The same process would be repeated for the outer ring of the toroid. Finally, two end caps would be molded separately and bonded to the two rings using guides. This method proved unsuccessful because the silicone would not retain the square shape after it was released from the clamp.

The next attempt at the mold was similar to the mold seen in Figure 63, however, the insert was made as one piece. This caused issues when the insert was removed from the mold. To fix this problem the insert was split into two pieces. The center insert, to keep an opening through the middle of the toroid for the finger, was also made to be a separate piece. In order to cap the other end of the silicone after the first step of the molding process, another mold was made to produce the end of the toroid. Once the open-ended toroid and the end cap were left to dry for 4 hours, the end cap was bonded to the toroid using more silicone rubber. Figure 62 represents the end cap mold.
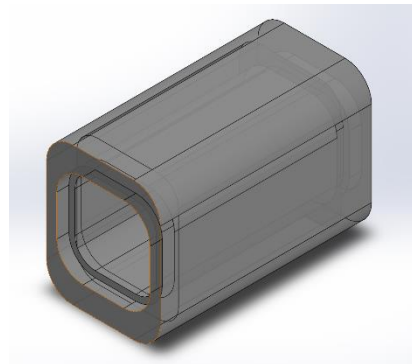
Figure 64: Rectangular Toroid Model

Figure 62: Rectangular Toroid End Cap Mold

Figure 63: Rectangular Toroid Mold

From previous attempts at pressurizing the toroid, we found the method of feeding a pneumatic tube through the silicone to be unsuccessful. To create a proper seal, we sandwiched the silicone wall between two acrylic plates with small machine screws. The acrylic plate were then tightened using a nut. This can be seen in Figure 65. To allow for air to input air into the chamber we used a vented screw, a screw with a through hole the center, and slid a pneumatic tube on the exposed threads of the screw outside of the silicone toroid.
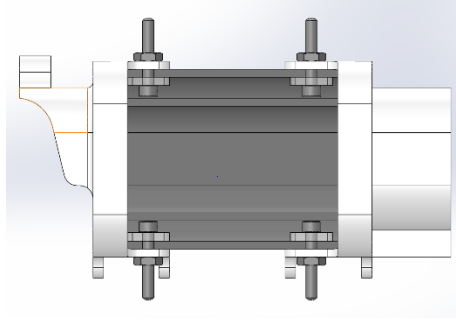
Figure 65: Representation of Method of Sealing Toroid

An issue that arose during the molding process was the outer wall of the toroid bloating when pressure was introduced into the air cavity. This caused the toroid to expand radially, instead of giving the actuation desired. To prevent the bloating of the toroid we inserted fabric into the mold. The fabric width was cut to be close to the height of the toroid. The fabric was then wrapped around the mold insert and lightly glued. Silicone was then poured in the mold and allowed to dry for 4 hours. This resulted in a silicone toroid that could not expand radially, but could still be flexible.

The mold seen in Figure 63 proved unsuccessful because the acrylic piece that sandwiched the outer wall of the toroid tore the silicone. In order to move forward with the project, a new method of creating silicone toroids was devised. Instead of creating a mold of the completed toroid, a hollow cylinder would be formed by wrapping meshed fabric around a 3D printed cylindrical guide. Then silicone would be brushed onto the fabric and left to dry. This would create a solid hollow cylinder of silicone with fabric embedded into it. When the silicone was dried one end of the silicone cylinder would be fed into the other. This could create the air gap needed to be inflated for actuation. From there the open ended toroid would be slid onto a 3D printed guide. An acrylic ring would be laser cut and a vented screw would be secured to the ring using a nut. This acrylic ring would be sandwiched between the outer and inner wall of the turoid. More silicone would then be applied to the top of the toroid. This extra silicone would seal the toroid in an attempt to create a sealable toroid. Seen in Figure 66 is the two 3D printed
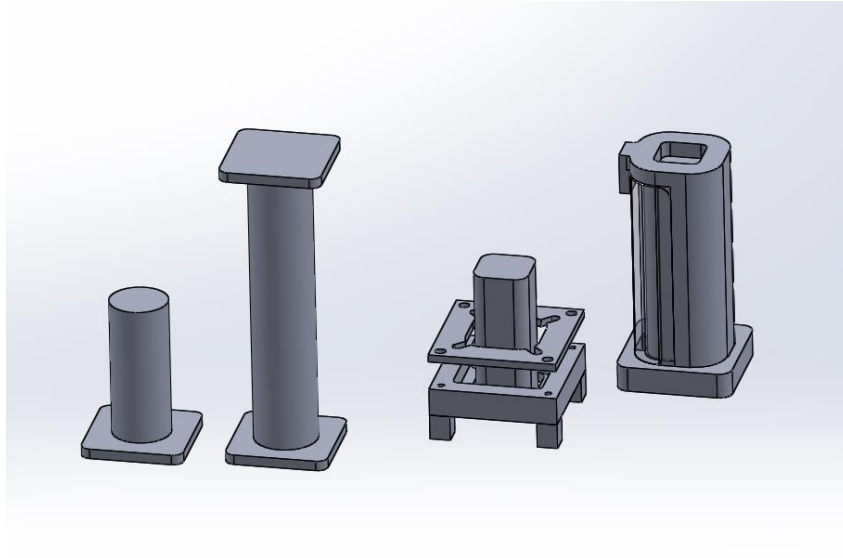
Figure 66: Fabric Coated Silicone Molds

guides that were created. The first attempt, on the left side of the figure, was a simpler version. The fabric would be wrapped around the larger cylinder. The length of the cylinder was calculated by the Solidworks model of the desired toroid. Once the first step of the molding process was completed, the open-ended toroid would be slid onto the smaller cylinder open-ended side down. Then the outer and inner wall of the toroid would then be stretched outward and sandwiched together by two acrylic rings. The rings would be fixed together using screws and nuts.

The 3D printed mold guides in the right hand side of Figure 66 was a more refined prototype of the mold guide. The larger assembly was used similar to the previous attempt, however, this guide was able to create a rectangular toroid with a consistent wall thickness. Like the other guide, fabric was wrapped around the center rectangular guide. Silicone would be brushed on the fabric and the outer walls would be slide around the silicone. This ensured the constant thickness. Once the open-ended toroid was formed it would be slid onto the mold piece to the right. In order to seal this toroid, a new concept of sandwiching the outer and inner wall of the silicone between the 3D printed finger piece and an acrylic ring. Seen in Figure 67 is a Solidworks model of this process. The 3D printed finger base would be slid into the base of the mold. Then the toroid was slid onto the center guide piece, open-ended side first. Then the open ended walls would be folded outward and held together. Next, an acrylic ring would be forced
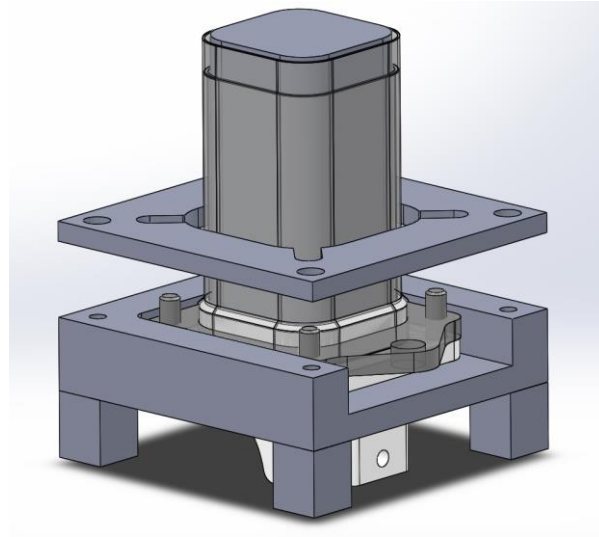
Figure 67: Mold Guide for Closing Open-Ended Toroid

on to sandwich the silicone between the 3D printed base and itself. The larger square plate would then be used to hold the sandwiched components together by screwing it into the mold base into the four holes seen in the figure. The mold guide also had holes in the bottom so screws could fix the 3D printed finger base and the acrylic ring together by threading the holes in the acrylic ring. The clamp plate would then be removed from the mold guide base and the sealed toroid could be removed.

Non-Permeable Material Actuators

With all of the methods that were attempted with silicone actuators, none provided the team with successful results. To move forward we decided to explore other options for create a toroid actuator that were not silicone. The three materials we tested were latex gloves, surgical tubing, and vacuum sealable plastic. Unfortunately, it was difficult to model these concepts in CAD so no screenshots can be provided for these concepts. See the Results section for the completed prototypes of these concepts.

Latex Glove

The second method of actuation was done using fabrics instead of silicone. An initial prototype was made using latex gloves. First an acrylic ring was cut, using the lab's laser cutter. This ring was similar in shape to the cross section of the rectangular shaped silicone toroid seen in Figure 65. The ring also had a hole within the ring so that a pneumatic tube could be fed through it. Next, a finger from a latex glove was cut from the glove and fed through the opening of the ring. Using hot glue, the finger of the glove was bonded to the acrylic ring. A 3mm diameter pneumatic tube was fed through the hole in the ring and glued in place. Once we ensured that there was no openings between the glove and the acrylic ring a second finger was cut off the glove and stretched over the outside edge of the ring. Using hot glue, the outer layer of latex glove was bonded to the acrylic ring. This created an air chamber between the two layers

of glove that could be pressurized by the pneumatic tube. To test this actuator, the pneumatic tube was connected to a syringe.

Surgical Tubing

The latex glove concept proved unsuccessful because the latex expanded too much when it was inflated. A second attempt of this concept was done by using large diameter surgical tubing. To prototype this 7/8" inner diameter surgical tubing was acquired. Similar to the fabric coated silicone actuators described in Section 0, the surgical tubing was fed inside itself. An acrylic ring was then slid into the far end to keep the far end of the actuator open. Finally, it was attempted to make a seal for the open end of the toroid. Unfortunately, we had trouble with sealing the surgical tubing to itself (see Results section for details).

Finger Tip Touch and Palm Touch Actuators

Fingertip and palm actuators were manufactured similar to the silicone toroid. They were designed to be made out of silicone. The geometry of these actuators was a solid disk of silicone with an air cavity left in the middle of the disk. This can be seen in Figure 68. A hole was then left in the bottom to insert a vented screw. The head of the screw would be left inside the air cavity so that air could be supplied to the actuator to inflate it. The vented screw would then be fixed to the actuator with a nut on the bottom of the actuator. The force applied by the nut would seal the hole that the screw was fed through.
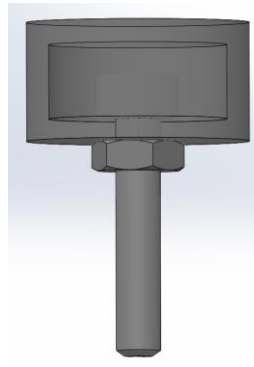
Figure 68: Finger Tip Actuator

Similar to the toroid, a two-step molding process was designed. The first step of the mold consisted of a mold base with a flat bottomed hole in the center. An insert was made to fit in the top of the mold base to create the air pocket in the center of the disk. The mold was then left to dry for 4 hours. The first step of the molding process left us with a bowl shaped silicone piece with a hole through the bottom of the bowl. The vented screw would then be inserted into the mold. To create the top of the disk, the same mold base will be used with a different insert. This will create a thin silicone disk that will then be bonded to the hollowed out disk using more silicone. Figure 69 represents the fingertip actuator mold design.
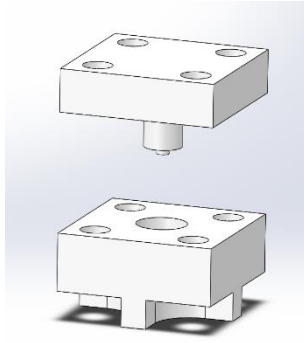
Figure 69: Finger Tip Actuator Mold

Similar to the fingertip actuator, the palm touch actuator had similar geometry. The actuator can be seen in Figure 70. The molding process and assembly process was the same as the fingertip actuator. The only difference was there were three screws instead of one. This was because the actuator needed to be securely fixed to the bottom of the hand component of the glove so that when it was actuated it would not pull away from the glove component. In this assembly, the middle screw would be vented and the left and right screws would not be vented. The mold to this actuator would be very similar to the fingertip actuator, however, the geometry would be changed to fit the design of the palm touch actuator.
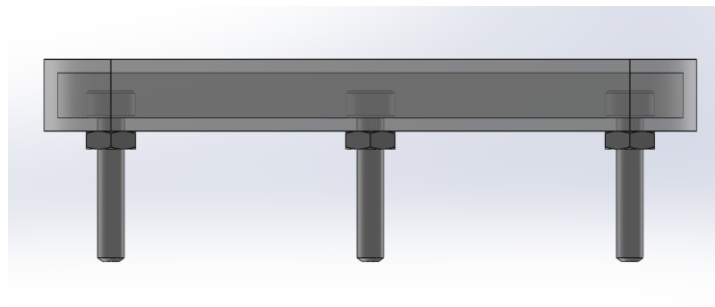
Figure 70: Palm Touch Actuator

The mold and creation of these actuators have not been completed yet, however, further prototyping and design will occur as the project moves forward.

Glove Component Concepts

There were 6 main component to the glove. The first component is the top of the hand glove component that would house the valve manifold, the IMU, various electrical components, and have ways to mount the bottom of the hand component, the wrist component, and the 3 finger modules. The second component is the bottom of the hand component that would house the palm touch actuator. The third component was the wrist component that would house the glove's battery and main microcontroller. Components 4 through 6 would be the two finger and one thumb modules. The following describes the modules.

Top of the Hand Glove Component Concepts

Several attempts were made to create a 3D printed glove that fit comfortable and snuggly on the user's hand. The first concept can be seen in Figure 71. This concept consisted of several components that could rotate in respect to the base plate. The idea was that these components could rotate to better form to the user's hand. A 3D print of this concept was made (see Results) but the concept was too simplistic to fit the needs of the glove.
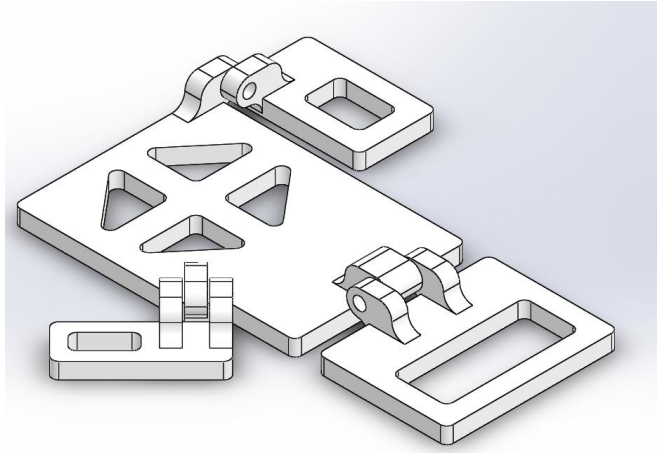
Figure 71: Concept 1 of Top of the Hand Glove Component

A second concept was attempted for the top of the glove piece. The concept can be seen in Figure 72. This concept was made to better form to the user's hand without needing several components. Foam strips would then be glued onto the bottom of this component so that the glove would be more comfortable for the user. This concept also included front mounting holes to attach the fingers glove. It also included mounting holes on the back of the glove to attach a wrist piece that would hold the glove's microcontroller. Finally a cavity was created on the top of the glove so the valves and various other electrical components could be mounted. This concept was again 3D printed (see Results), however, the component was not large enough to fit on the user's hand.
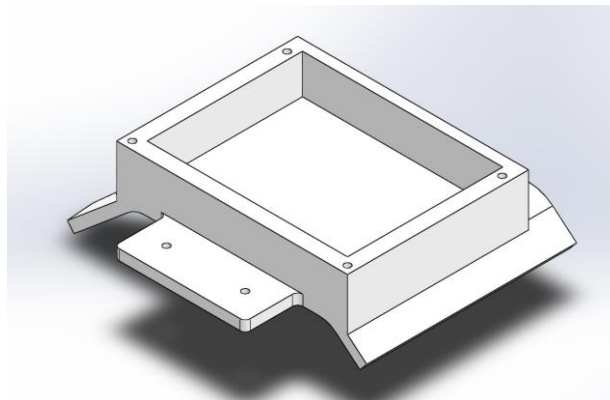
Figure 72: Concept 2 of Top of Hand Glove Component

A third and fourth concept was designed to be similar to concept 2, however, the geometry of the glove was altered. These two concepts were very similar, however, the fourth concept (seen in Figure 73) was altered slightly from the third concept after a 3D print was made of the third concept. These concepts included mounting holes for the fingers and wrist piece as well as a cavity for the valves and electrical components. It also included a flange for the thumb module. Between the 3rd and 4th concept, a few dimensions and angles changed due to analyzing the 3D print of the third concept. Similar to concept 2, this component would also have strips of foam glues to the bottom of the glove. There will also be mounting holes for the valve manifold and PCB circuit board. This glove component would include the valve manifold, the IMU circuit board, and a PCB motherboard that would include 4 AdafruitTrinket microcontrollers.
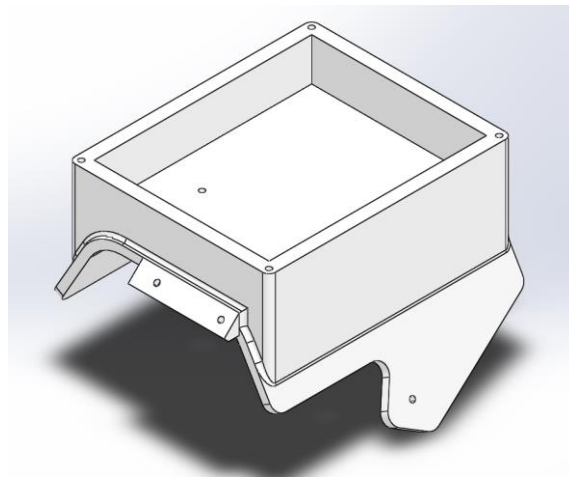
Figure 73: Concept 4 of the Top of Hand Glove Component

Bottom of the Hand Glove Component Concept

In order to house the palm touch actuator, a 3D printed casing needed to be designed. The design of this component was made to match the largest dimension of the flanges on the final concept of the top of the hand glove component. Like the top of the hand component, foam strips would be glued onto this component. That way the top and bottom of the hand components would fit together and snuggly sandwich the user's hand. This component also had a slot bored out of the center with three mounting holes. This slot and mounting holes were for the palm touch actuator. In addition, there is also a slotted out geometry on the bottom of the component. This was to contain the strap, that would connect the bottom and top of the hand glove components together to, to the bottom of the hand glove component.
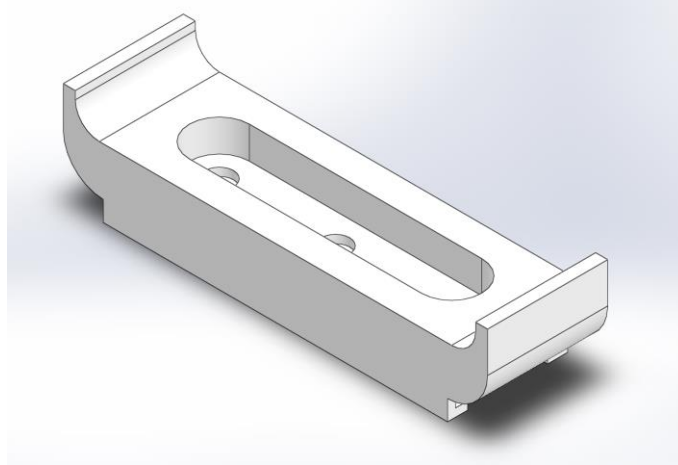
Figure 74: Bottom of the Hand Glove Component

Finger and Thumb Modules

The finger and thumb models were both identical, however, the thumb was made larger in width and height and shorter in length to accommodate the size of the thumb. Figure 75 shows the two assemblies of the finger and thumb modules. The method of actuation was the vacuum sealable plastic actuators described above in Section **Error! Reference source not found.**. The finger modules also included the fingertip touch silicone actuators. These actuators were identical for the thumb
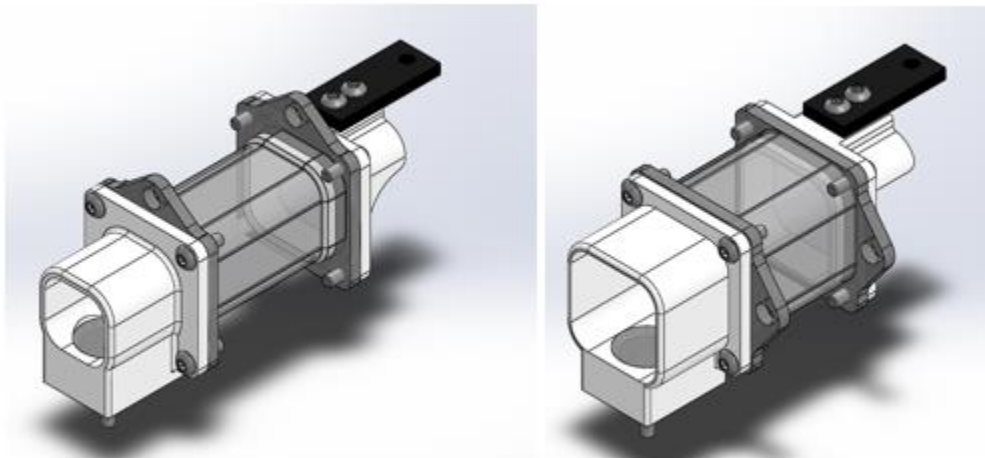
Figure 75: Finger and Thumb Modules (Finger to the left and Thumb to the right)

and finger modules. The actuators were mounted in a bore in the fingertip 3D printed component, seen in Figure 76. In the center of the bore was a through holes that the vented screw of the fingertip actuator would be fed through. It would then be fixed to the module using a nut on the bottom of the 3D printed component. The 3D printed components were very similar

between the thumb and finger module, however, as described above, the width and height of the thumb components were larger to fit the user's thumb better.
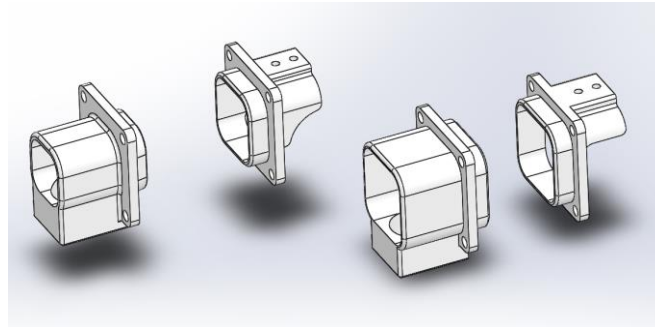
Figure 76: 3D Printed Finger and Thumb Components (Finger components on the left and thumb components on the right)

The way in which the vacuum sealable plastic actuators were fixed to the 3D printed components was done using a combination of friction fit and glue. As seen in Figure 76, the 3D printed components had a flange on the actuator side of the components. This was designed so the actuator could be stretched over the flange. Next, an acrylic plate (seen in Figure 75) was laser cut so the middle opening of the acrylic plate had 0.15 mm clearance to the flange in the 3D printed components. This created a friction fit by sandwiching the actuator between the acrylic and the 3D printed component. To ensure that the hold never let the actuator lose, small amount of hot glue were also used to fix the actuator to the 3D printed component. The acrylic plate also included an extrusion and through hole that was designed to hold the optical curvature sensor.

Glove Prototype 1 CAD Model

The first prototype of the glove consisted of top and bottom components of the glove as well as the two finger modules and the thumb module. The final design can be seen in Figure 77. As seen in the figure, the valve manifold would be mounted inside the glove as well as the PCB mother board and the IMU. In order to mount the finger and thumb modules to the glove, strips of mouse pad foam. They would then be screwed into the two holes on the 3D printed component of the finger and thumb modules. The other end of the foam strip would be screwed into the top of the hand glove component.
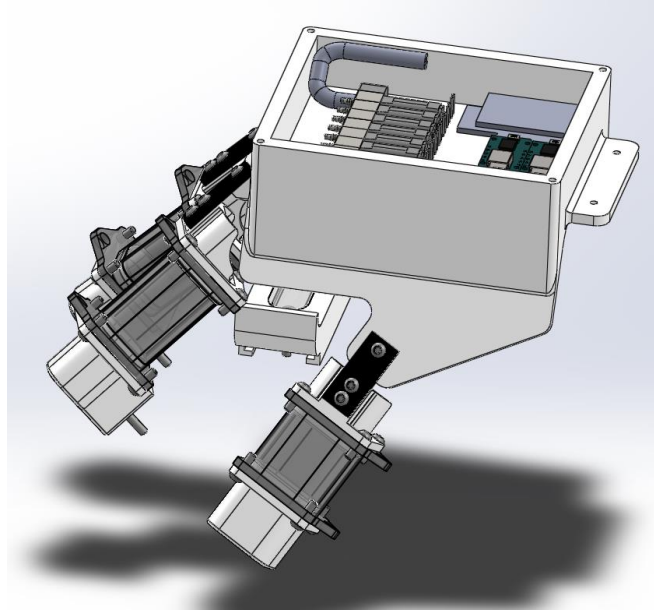
Figure 77: Prototype 1 CAD Design

Curvature Sensor Test Platform

In order to test the curvature sensors used in the glove, we designed and manufactured a test platform. It consisted of three 3D printed simulated fingers, see Figure 78. The fingers were able to pivot in the middle of the finger, simulating the middle joint of the human finger. In addition, 10 Ohm rotational potentiometers were imbedded into the finger to provide a control for determining rotation of the finger. This allowed the data from the curvature sensor to be compared to the actual value provided by the potentiometer. The fingers also had indents built in to house the curvature sensors. Finally, the three fingers were mounted on a solid base. Further testing and calibration of the curvature sensors will be completed using this test platform.
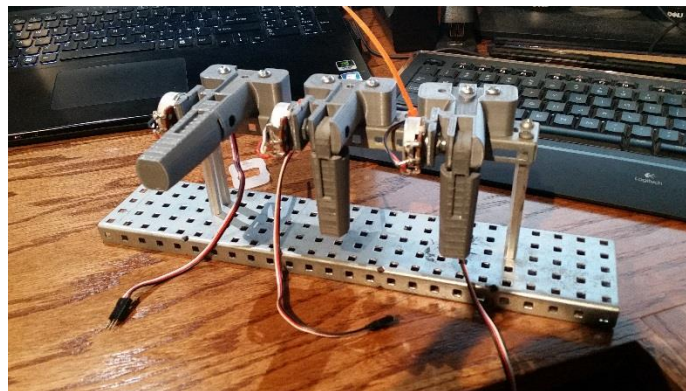
Figure 78: Curvature Sensor Test Platform

Results:

Finger Joint Actuators

After many iteration of testing and prototyping the team was able to choose a final method of creating the finger joint actuator. From our results we found that the vacuum sealable actuator was the best and most reliable method of creating the actuator. To see the progression of concepts in chronological order see Figure 79 below.

Figure 79: Evolution of Finger Joint Actuators

Cylindrical Toroid Actuator

The first attempt at molding a silicone toroid was in a cylindrical shape, see Figure 60. We were able to create a complete toroid from our first mold. This can be seen in Figure 80. However, we found some issues with our process. The first major issue was the way in which we capped the other end of the toroid after the first step of the molding process. As described in Section 0, the open-ended toroid was suspended over an approximated amount of silicone. The toroid was then suspended in the

Figure 80: Successful Attempt of Molded Silicone Cylindrical Toroid

mold base so that it made contact with the silicone. This caused an issue of uncertainty because the mold base was opaque and it was impossible to see how far the silicone seeped into the open-ended toroid. From this uncertainty, the end cap of our toroid in Figure 80 seeped too far into the toroid so that half of the height was solid silicone. To prevent this from happening in future mold attempts, we decided to mold the end cap separately from the open-ended toroid and then bond the two pieces together to form the complete toroid.

Another issue with our first toroid was the way in which we secured the pneumatic input tube. For this first attempt we simply fed a tube through the wall of silicone and glue around the

edges of the tube to try to seal. The problem was that silicone does not bond to glue or the pneumatic tube. This caused the toroid to not properly seal so that it could not hold any significant pressure. From the small amounts of pressure we were able to input into the toroid we also found that the outer walls of the silicone toroid expanded radially. This caused the toroid to bloat instead of providing the amount resistance to the finger we desired. However, despite the bloating and un-proper seal of the toroid, from the short amount of testing we were able to complete we found that the toroid did provide a sensation of resistance to the finger. The results were small and not easily measured, but by human testing with some of the team members we determined that the toroid was a viable method to provide feedback to the middle joint of the finger.

Rectangular Toroid Actuator

Several attempts were made to mold the rectangular shaped silicone toroid. Initial attempts proved unsuccessful due to the design of the mold. This was because the mold did not break up into enough components. The walls of the toroid were only 1.5 mm thick, and the air gap between the outer and inner wall of the toroid was 1 mm on the sides and 4 mm on the top and bottom. This caused the inner wall to either not be molded completely or get stuck in the mold after being removed. As described in Section 0, the mold was altered so that all inserted mold pieces could be split in half once the silicone dried. With a proper mold printed we were able to mold several toroids with fabric inserted into the. However, the first results were not satisfactory. Figure 81 depicts the first two attempts of the rectangular toroid with fabric embedded in the silicone. As seen in Figure 81, the walls

Figure 81: Initial Attempts of Molded Silicone Rectangular Toroids

of the toroid did not materialize well. This was because the fabric we chose to test initially had a small band on each end of the fabric that housed a thin metal wire. This fabric was chosen so that the wire could help retain the shape of the fabric before the silicone dried. However, because silicone could not penetrate the band that housed the wire, it left many small holes in the outer wall of the silicone. This led us to the conclusion that only fabric that had sufficient spacing between threads would be viable to embed in the silicone.

Another mold was made using the same fabric, but the bands with the small wires were cut from the fabric so that only the mesh of the fabric remained. Figure 82 shows the results of the toroid.

Figure 82: Successful Attempt of Molded Silicone Rectangular Toroid

As seen in Figure 82, the outer wall of the toroid was completely intact and the fabric was successfully embedded in the outer wall.

The next attempts to create silicone actuators can be seen in Figure 83. The actuator on the right was the first attempt of the fabric coated silicone toroid. This method proved to be very difficult to stretch the open-ended side of the toroid outward so the acrylic could sandwich the silicone and seal it. After the prototype was made in the figure, we found that the toroid was not sealed. There were many leaks throughout the toroid. From these factors we decided to attempt another mold for the actuator.
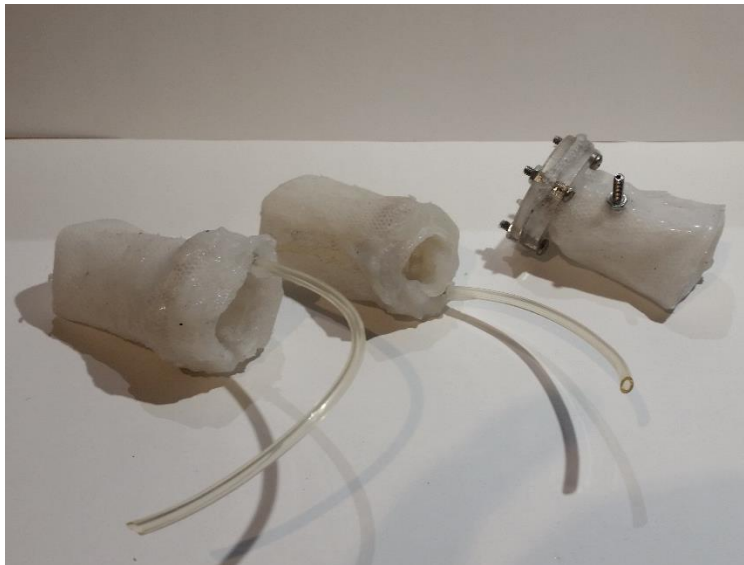
Figure 83: Fabric Coated Silicone Actuators

The toroid on the left in the figure were two attempts with the updated mold guide. These actuators were the closes the team came to creating a working silicone toroid. However, like the others, there were many small leaks and pressure could not be held. Based on the difficulty and un-reliability of assembly of all of the silicone actuators, the team decided to move onto to different concepts of creating a toroidal actuator.

Non-Permeable Material Actuators

The following sections describe the results and observations of the non-permeable material actuators.

Latex Glove

The team attempted to create a prototype of the concept of using two layers of latex glove fingers to create toroidal actuator. Several attempts were made to create a prototype, but many problems arose when trying to manufacture the actuators in this manner. The biggest problem was creating a seal between the two layers of glove fingers. Hot glue was used, but the glue did not completely bond to the latex. The same issue arose when trying to seal the pneumatic tube. Despite failed attempts, the prototype shown in Figure 84 was able to hold a small amount of pressure. From the testing done, using a syringe,

Figure 84: Latex Glove Actuator

we found that the latex expanded radially as long as axially. This caused little to no resistance to the finger. However, the purpose of the prototype was to understand the method in which non-permeable fabric could be used to create a pneumatic actuator. From this prototype the team was convinced that the method could be a viable solution to replace the use of silicone toroids. The fabric solution provided an easier and more reliable way to create an actuator.

Surgical Tubing

The team attempted to create toroidal actuator using large diameter surgical tubing. The results can be seen in Figure 85. As seen in the figure, the end of the toroid was never sealed. This was because we had difficulties finding ways to seal the tubing to itself. In addition to this, the wall thickness of the tubing was too thick so that the user would have a very difficult time bending their finger once in the actuator. We searched for tubing with a thinner wall thickness, however, not suppliers had thinner walled tubing. From this the idea was than scrapped and further concepts were explored.

Figure 85: Surgical Tubing Actuator

Vacuum Sealable Plastic Actuators

The final actuator that provided the only completely successful result was the vacuum sealable plastic actuators. The actuators seen in Figure 86. These actuators not only were completely sealed, and could hold over 5 PSI of pressure, but they were very simple to manufacture. The total time of assembly of each actuator was less than 30 mins. The plastic used for the actuators was also durable and not susceptible to tearing. In addition, the actuator was very slim and the thickness of the plastic was small enough that it provided little to no extra resistance to the user's finger, unlike the silicone and surgical tubing methods. In the end or our testing, this was the clear choice for our finger actuators.

Figure 86: Vacuum Sealable Plastic Actuators

Glove Components

Initial Concept and Prototype

In order to understand the sizing needed for the components of the glove, various components of the top level assembly were 3D printed. As seen in Figure 87 and Figure 88, finger rings were 3D

Figure 87: 3D Printed Finger Rings
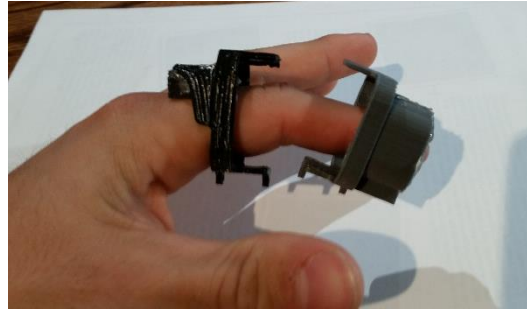
Figure 88: 3D Printed Finger Rings on Finger

printed. In order to allow for different sized fingers to fit in the ring, we hot glued foam on the inside of the ring. From these prototypes we found that the geometry of the rings was sized well to allow for a tight fit around the finger. Insertion into the ring was also done with ease.

Top of the Hand Component

Several designs and prototypes were created to provide us with a successful top of the hand component of the glove. The first one can be seen in Figure 89. As you can see in the figure, the fit on the user's hand was not acceptable. Also the flatness to all of the components did not allow for a good fit to the user's hand. From this concept, it was discovered that the top of the glove component could be made to be one part.
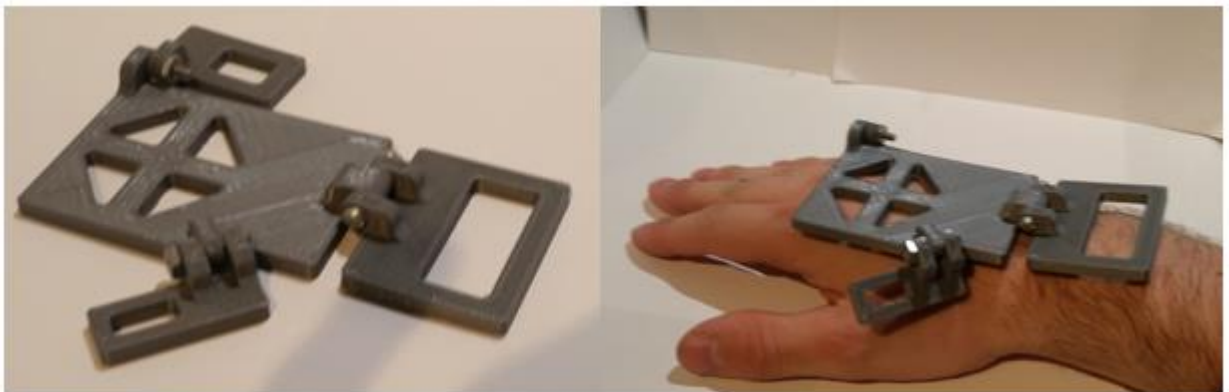
Figure 89: Concept 1 of Top of Hand Glove Component

The second concept be seen in Figure 90. This concept was a much better fit to user's hand however, as seen in the right side of the figure, the component was too small for the user's hand. Also the positioning of the thumb mounting flange was off.
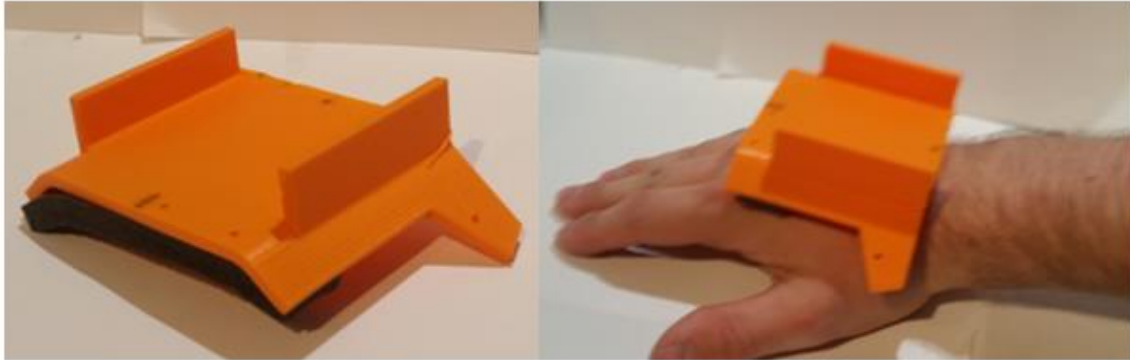
Figure 90: Concept 2 of Top of the Hand Glove Component

From the second concept a third concept was prototyped. The design was based of concept 2 and the geometry was altered to fit the user's hand better. The results can be seen in Figure 91. This prototype had a fairly good fit to the hand, however, the thumb mounting flange was off from the user's hand. Also the mounting flange for the finger modules needed to be slopped down towards the user's fingers.

Figure 91: Concept 3 of Top of the Hand Glove Component

The final concept and prototype of the top of the hand component can be seen in Figure 92. As seen in the figure, the fit on the hand is much closer than previous concepts. There were some minor geometrical issues with the concept, however, it fits the user's hand fairly well.
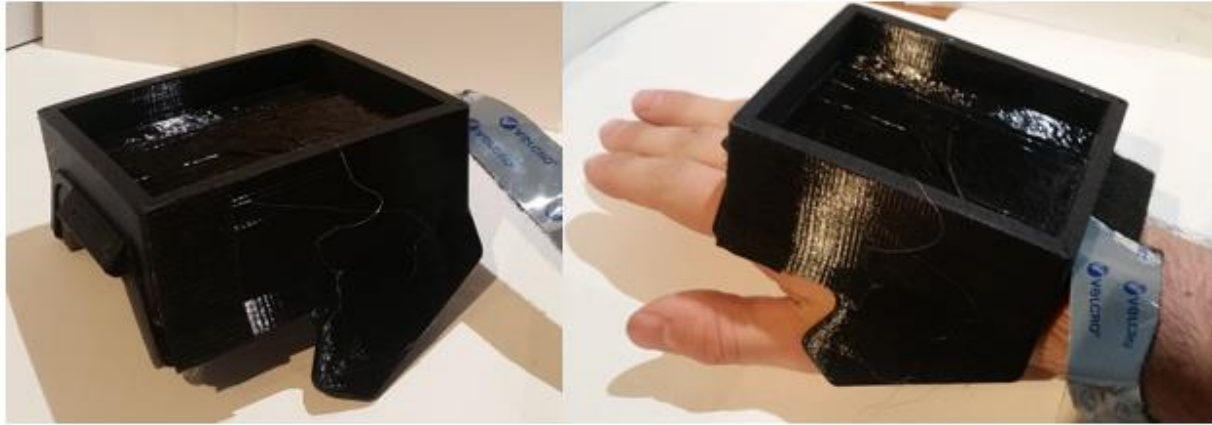
Figure 92: Final Concept of Top of the Hand Glove Component

Bottom of the Hand Component

        The bottom of the hand component was only prototyped once, however, it fit fairly well to the user's hand. The prototype can be seen in Figure 93. In addition, the component also fit well with the top of the hand glove component. Further concepts and prototypes will continue as the project continues.

Figure 93: Prototype of Bottom of the Hand Glove Component

Finger Modules

        Two prototypes were made of the finger modules. These included the actuator attached to the two 3D printed components as well as the mounting foam piece and the optical curvature sensor. These prototypes can be seen in Figure 94. From these prototypes it was found that the actuators could successfully provide the user with accurate haptic feedback. This feedback can simulate the sensation of grasping an object. We also found the modules to be fairly comfortable

106

on the finger. The only major problems we have encountered though was the un-commutability of the head of the vented screw. When the finger is bent, the head of the screw can dig into the user's finger. Further work to fix this problem will be explored as the project continues.

Figure 94: Finger Module Prototypes

Prototype 1

From the completed prototypes of the top and bottom of the hand components as well as the two finger module prototypes we were able to complete our first semi-complete prototype. This prototype can be seen in Figure 95.
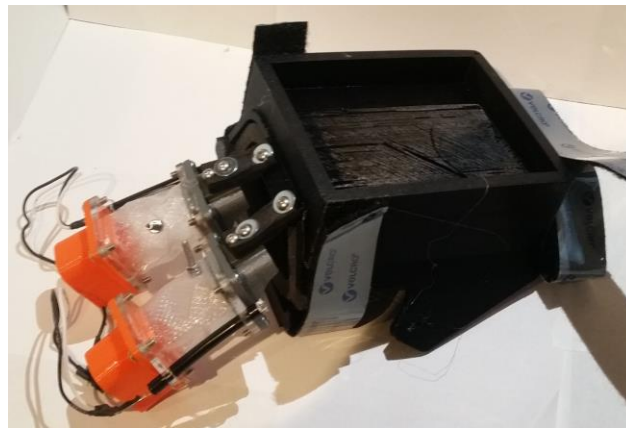
Figure 95: Prototype 1

From this prototype we determined that the overall design of the glove is very feasible and can be made into a comfortable device. This can be seen in Figure 96. The current prototype has some issues in comfortability. For example, the finger modules are too wide so they rub into each other. This problem will be fixed by decreasing the size of the finger modules. Further work on this will be done as the project continues.
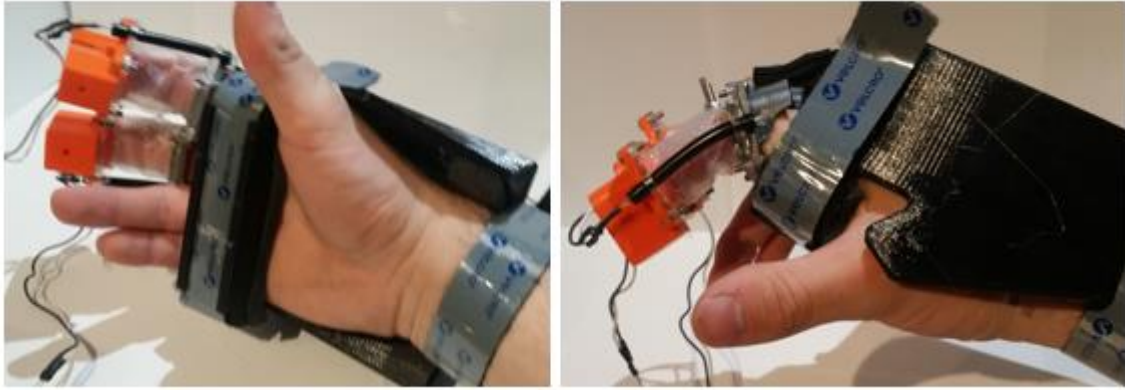
Figure 96: Prototype 1 on User's Hand

Appendix D Prototype 2 Mechanical Design and Results
Design:

Top Level Assembly

The final design for the top level assembly of the HERO Glove can be seen in Figure 97. This assembly consists of main glove base, two finger modules, a thumb module, and a palm module. The glove base would house the system's solenoid valves, microcontroller, vibration motors, and PCB motherboard. It would also allow for the finger and thumb modules to connect to the glove base forming a wearable "glove" for the user. Within the thumb modules and palm modules soft pneumatic actuators are used to give the user haptic feedback based on the contact the robot makes with the environment.
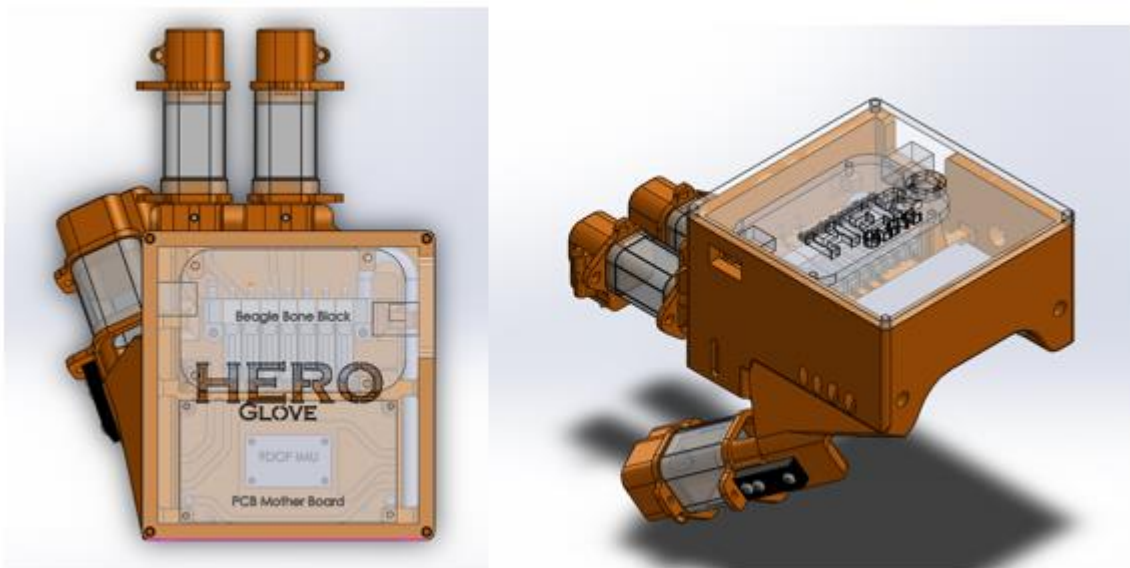
Figure 97: Top Level Assembly

Finger and Thumb Modules

The final design of the finger module and thumb module can be seen in Figure 98. The finger module consists of two 3D printed exoskeletal components and two soft pneumatic actuators.
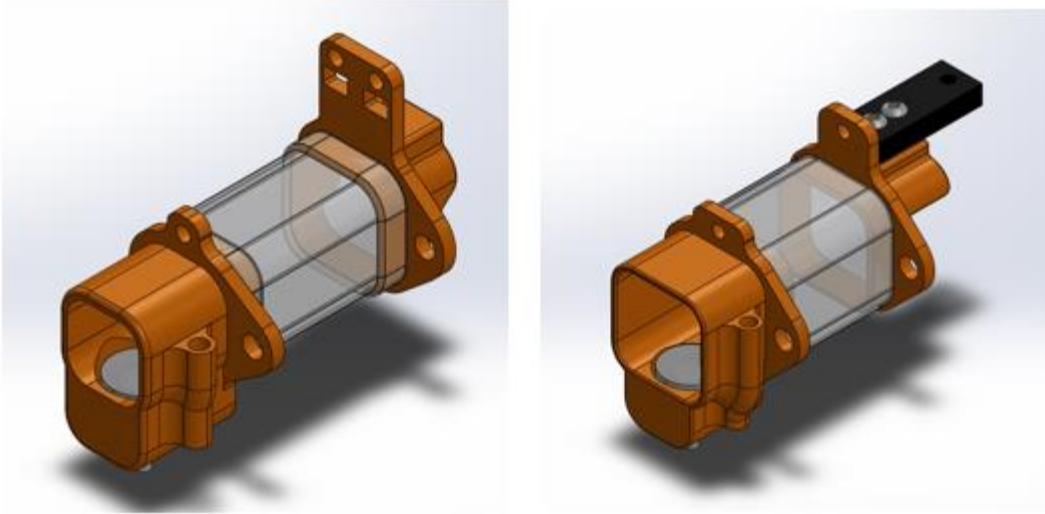
Figure 98: Fingure Module (left) Thumb Module (right)

Finger and Thumb Module Exoskeletal Components

As part of the figure module, there were two 3D printed exoskeletal components. These components can be seen in Figure 99. The purpose of the first component (right in the figure) was to provide a rigid opening for the user to insert their finger into the module. This component also allowed for the figure module to be mounted to the base component of the glove. This was done by inserting a M3 screw through the hole on the top platform of the component and screwing it into the base component. The second exoskeletal component (left in the figure) was designed to house the fingertip actuator of the finger module. In addition to providing a rigid structure to the finger module, these components also allowed for the curvature sensor to be mounted on the finger. The sensors was slid though the flanges on the side of the module. In addition, both components were designed to have a step on the opposite side of the opening. This step up allowed the finger joint actuator to be slid onto the components, completing the finger assembly. Finally, features on both of the components were designed to keep the tubes and wires of the actuators and sensors contained to the finger. On the fingertip component a tunnel was designed to contain the tube for the fingertip actuator. On the finger base component, a tab was created with two holes for the actuator tubes and two square opening for the connectors and wires of the curvature sensor. These features prevented wires and tubes to dangle from the module.
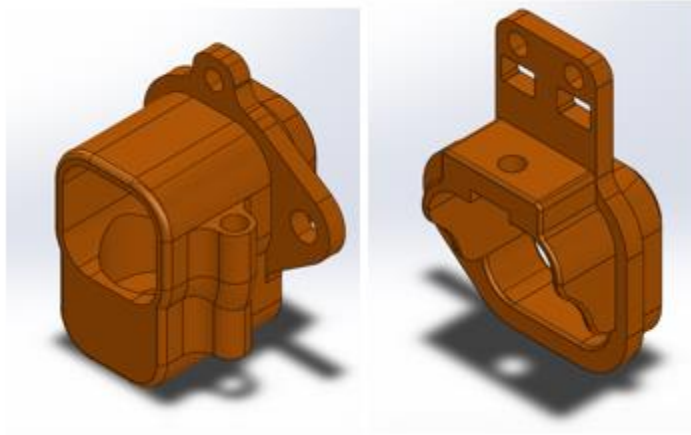
Figure 99: Finger Module 3D Printed Exoskeletal Components

The design of the thumb module was almost identical to the figure module design, however, the dimensions of the openings and the length of the actuator was altered to better match the anatomy of the human thumb. Also the tab that was designed on the figure module was removed for the thumb because of the way the thumb mounted on the base component.

Finger Joint Actuator Design and Fabrication

The general concept of the actuator was to create a toroid out of vacuum sealable plastic. The user would then slide their finger through the toroid. The toroid would be fixed to the two exoskeletal 3D printed components of the finger module. The finger joint actuators were designed to give the user the sensation of grasping an object. This was done by pressurizing the air chamber within the toroid. This pressurization would squeeze the user's finger, effectively resisting the motion of the center joint of their finger. The shape of the actuators for both the thumb and finger modules were square with rounded corners.

To manufacture the actuators, vacuum sealable food storage plastic was cut and heat sealed to create the toroid. Two layers of the plastic were initially cut to size using a laser cutter. The plastic roll was made to have a pre sealed edge, so the position of the laser cut allowed for a small strip of the pre sealed portion of the plastic to remain. This resulted in two layers of plastic with a seal along the longer edge. The opposite edge was then fed into the heat sealer so create a cylindrical shell of plastic. The heat sealer had an offset from the alignment wall to the heat sealing strip so the size of the rectangle cut by the laser cut took into account this offset. Once the cylinder was made, one end of the cylinder was folded over and fit into itself. This created an open toroid of plastic. A thin piece of acrylic was then slid into the center opening. This acrylic piece prevented the center opening from being sealed closed. The first side of the opening was sealed. Once again, the laser cut plastic rectangle took into account the needed material to fold over the plastic and the offset of the heat sealer strip to ensure the length of the final actuator was the desired length. Once the first side of the opening was sealed, a vented screw was then pushed through the outer layer of the toroid. Using a nut, the screw was then fixed to the outer layer of the toroid with the head of the screw inside the toroid's air chamber. Finally, again using the acrylic, the other end of the toroid was sealed. This resulted in a completely sealed actuator. The

actuator was then tested with the air supply to ensure no leaks. If leaks were suspected, the actuator was dipped into a cup of water and air was fed into the chamber to see if any air bubbles escaped. Due to the low cost of the actuator and the quick assembly time, if the actuator was not sealed properly, the actuator would be scrapped and the process would start over. Again, the process of creating the thumb actuators was exactly the same. The only difference was the size of the final actuator.

Fingertip Actuator Design and Fabrication

The general concept of the fingertip actuators, seen in Figure 100, was to create a cylinder of silicone rubber with an air chamber in the center. When the air chamber was pressurized, the actuator would expand up, physically depressing the user's finger. IT would expand upwards because the actuator was designed to be housed in a cavity in the 3D printed fingertip component of the finger or thumb module.
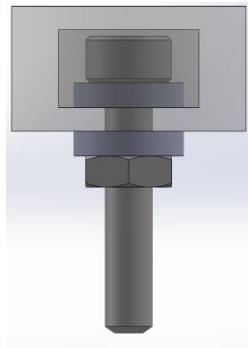
Figure 100: Fingertip Actuator

The design of the actuator was fairly simple. The actuator was a simple cylinder with an opening in the center. To provide air to the actuator, a vented screw was fed through the bottom wall of the cylinder. The head of the screw and the nut then sandwiched the bottom wall of the cylinder between two custom acrylic washers. This ensure that there was no leaks in the actuator.

To fabricate the actuator, a two-step mold process was created. The first step of the mold was to create an open shell of the cylinder. This mold can be seen in Figure 101. The four posts of the mold insert (right in the figure) were inserted into the holes in the mold base (left in the figure). This perfectly positioned the insert to be in the center of the mold cavity. The hole in the center of the insert was to allow for the vented screw to be inserted during the molding process. This not only allowed the air to escape from the mold, but it also kept the hole open for the vent screw in the final actuator. To create the actuator, silicone rubber was then mixed and poured into the mold. When the mold insert was mated to the mold base, the excess silicone rubber was squeezed out. To ensure the two piece of the mold did not separate during the drying process a wire was wrapped around the mold to keep it shut. The silicone was then left to dry for 4 hours. While the silicone was drying, a silicone disc was also left to dry. To create the disc, the mold insert, Figure 101, was used. Silicone rubber was poured into the cavity and a straight edge was used to push the excess away from the mold.

When both of the components of the actuator were dry, the silicone was left inside the mold cavities. The mold insert was removed, and excess silicone was cut away from both molds. Next, the vented screw with the acrylic washer was inserted into the mold base through the bottom of the cylinder. The mold base also had a hole though the bottom to allow for the screw to be pushed through during the final molding process. Silicone rubber was again mixed and a small amount of silicone was spread around the edge of the open cylinder. The top of the cylinder was then mated to the base by inserting the four posts of the mold into the base. Once again, the mold was wrapped with a wire and the mold was left to dry for 4 hours. When the mold was done it was removed from the mold. To tighten the vented screw and sandwich the bottom of the cylinder between two acrylic washers, the vented screw was fixed from rotating by tightening the end of the screw in a vice. Then, pliers were used to tighten the nut on the screw completing the seal of the actuator. The actuator was then tested using the air supply, once again, being dipped in a cup of water to check for air bubbles escaping the actuator. The fingertip actuator for the thumb module was identical to the actuator for the finger modules.
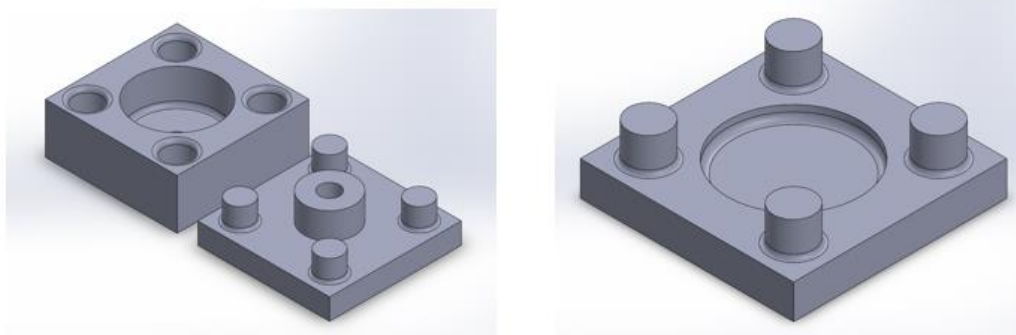
Figure 101: Fingertip Actuator: Base mold (Left) Top Mold (Right)

Finger and Thumb Module Assembly

Once the both the fingertip and finger joint actuators and the 3D printed components were completed the finger and thumb modules were then assembled. The process in which the modules were assembled was identical for both the finger and thumb modules. First the finger joint actuator was stretched to fit over the set up on the 3D printed finger base component. To ensure a tight fit the actuator was designed to fit snuggly over the 3D printed component, so forceful stretching of the opening of the plastic was sometimes necessary. Once the actuator was full stretched onto the 3D printed component, super glue adhesive was used to fix the actuator to the 3D printed component. The super glue was applied on the outside perimeter of the actuator as well on the inside edge. This same process was repeated to fix the actuator to the fingertip 3D printed component.

Once the actuator was fixed to the two 3D printed components, the fingertip actuator was then attached. The completed actuator was inserted into the cavity on the fingertip 3D printed

component. A screw was then used to tighten the actuator to the 3D printed component. This ensure that the actuator would not try to push itself out of the cavity when pressurized. To complete the assembly, the curvature sensor was inserted into the flanges on the 3D printed components. The wire for the far end of the curvature sensor was fed though the containing features of the 3D printed components. The tube for the fingertip actuator was also fed through the tunnel and attached to the vented screw that was protruding from the bottom of the 3D printed component. The final finger and thumb modules were then fixed to the finger base.

Palm Module

The palm module of the HERO Glove was used to contain the glove to the user's hand as well as house the palm touch actuator. The final design can be seen in Figure 102. This assembly consisted of a 3D printed component and a silicone actuator. The 3D printed component had a center cavity that would house the palm actuator. It also included flanges so that it could slide in and out of the glove base. The hole on the flanges was used to keep the palm module from escaping the base component.
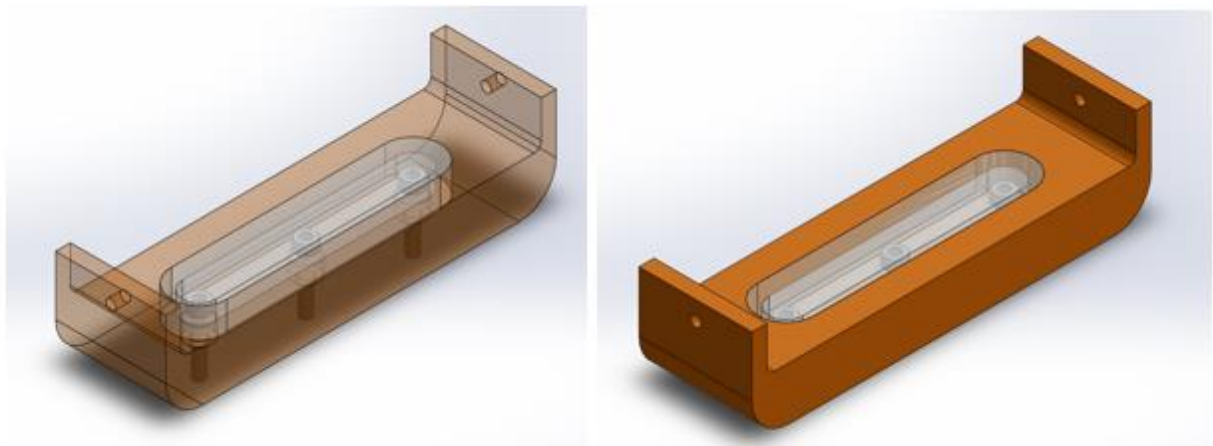
Figure 102: Palm Module

Palm Touch Actuator Design and Fabrication

The design of the palm touch actuator was identical in diameter and height to the finger touch actuator, however, the actuator was extended to stretch across the user's palm. The actuator also included 3 sets of vented screws and washers. This prevented the actuator from bulging in the center when pressurized resulting in an even amount of pressure across the actuator. In addition, the three sets of screws prevented the actuator from pushing itself out of the cavity.

To mold these actuators the same process was used as the fingertip actuators. The molds can be seen in Figure 103. One the mold was completed, the three vented screws were fed into the holes of the 3D printed component and fixed with nuts.
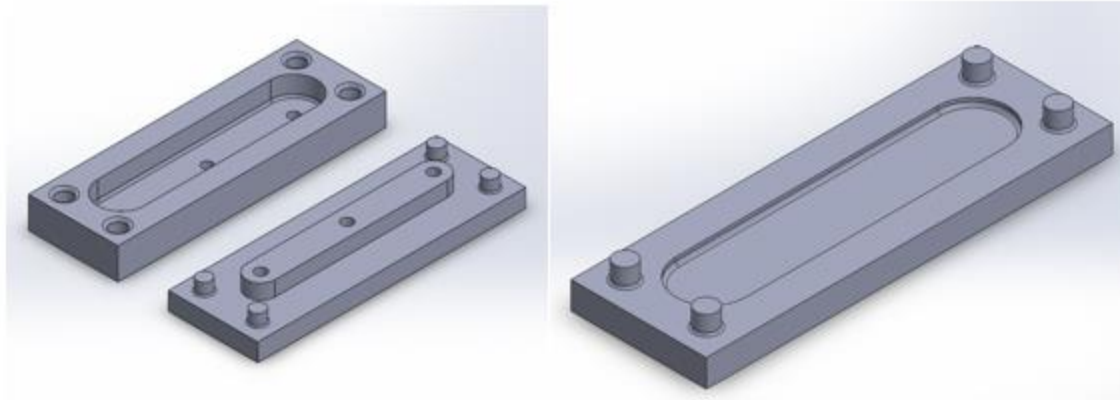
Figure 103: Palm Actuator Mold: Base Mold (Left) Top Mold (Right)

Glove Base

The glove base is the hub of the whole HERO Glove. This allows the system to resemble a wearable glove. This component was designed to be 3D printable. It was also designed so that all of the features needed to mount the various electrical components could be built into the base. This would allow the system to require less hardware to mount the components. The final design of the glove base can be seen in Figure 104. In the figure, it various extrusions can be seen within the open cavity in the center of the base. These were effective standoffs to mount the microcontroller and the PCB motherboard. The holes were tapped, using standard M2 and M3 metric taps, so that screws could fix the electronic components to the base. Similarly the solenoid valve manifold was fixed to the base using the same method.

To route the many wires and tubes for the figure module sensors and actuators, channels were built into the base component. These channels can be seen again in the center cavity of the base. On the top-left portion of the base 6 channels can be seen. These channels each held the power and ground wires to the curvature sensor for each figure or thumb module. These channels were then connected so that the 12 wires could easily be bundled up to be connected to the PCB motherboard. The purpose of these channels was to allow the wires to the finger modules to be contained neatly underneath the valve manifold. This way a stationary wire could be glued in place inside the channels. One end of the wire could be easily detached and attached to the PCB motherboard, and the other end could be fixed to the glove to allow the finger curvature sensors to be connected. In addition, on the bottom center section of the base a rectangular channel can be seen with 2 channels exiting the rectangle. This channel was designed for the system's IMU. The IMU could be mounted directly on the base component and the wires could route through the two channels to connect to the PCB motherboard above. The IMU was designed to be inset to the bottom of the cavity of the base component so that there was enough clearance for the PCB motherboard that was mounted above it. To route the input air tube to the manifold and tubes from the thumb modules, two identical channels were designed on the left and right side of the base component. For the input tube, a valve was fit on the outside of the base component so that

115

an external air supply could easily be connected to the manifold without removing the cover of the glove.

To mount the finger modules to the base component a mounting platform was designed for the middle and pointer finger modules. The finger module could be fixed to the bottom of the mounting platform using a M3 screw. To allow for the wires and tubes from the module to enter the base component, four holes were created for each finger. The two top holes, seen in the figure, were designed to fit the tubes for the finger joint and fingertip actuators. The two larger rectangular holes were designed to fit the female connectors of the wires. Similarly for the thumb, a flange was designed off the left side of the glove. The angle and dimensions of the flange were determined by error checking and approximation of the anatomy of the human thumb. To allow the wires and tubes to enter the base component from the thumb module, holes were designed to allow access to the inside of the base component from the outside.

Vibration motors were used in the system to alert the user when the robot was reaching its workspace limits. To mount these vibration motors, cavities were designed into the base component with a through hole extending into the main cavity of the part. The cavity for the motors was dimensioned so that the motors would have a loose press fit into the part. The wires for the motors were then inserted through the through hole so that they could be connected to the PCB mother board.

Finally, to connect the palm module of the glove, two flanges were designed to allow for a tight fit with the palm module. The palm module would then slide into the flanges and be contained in two dimensions. The third free dimension would allow for the palm module to raise and lower based on the thickness of the user's hand. A slot was incorporated on each flange so a M3 screw could be screwed through the slot into the palm module. This contained the palm module so that it would not pop out of the flanges when the user was putting the glove on.
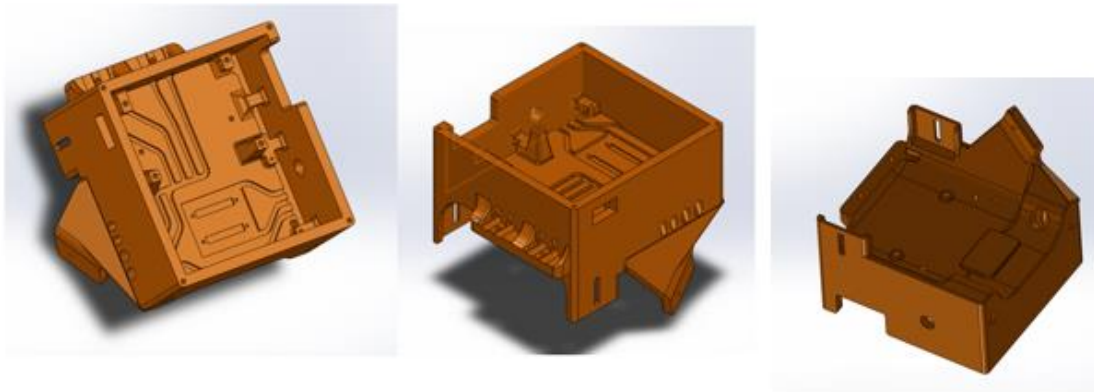
Figure 104: Base Component

Results:

Prototype 2

The final prototype of the HERO Glove can be seen in Figure 105. As seen in the figure, all the components were able to be incorporated into the prototype. The two finger modules as well as the thumb module had fully functionality. In addition, the palm module also had full functionality. Also seen in the figure is the apparent disorganization of wires and an extra breadboard on the back side of the glove. The reason for this was because there were a few errors in the first iteration of the PCB motherboard. This resulted in the addition of a breadboard circuit to bypass the PCB for all of the curvature sensors. The next prototype of the HERO Glove will be much more organized and neatly packaged.
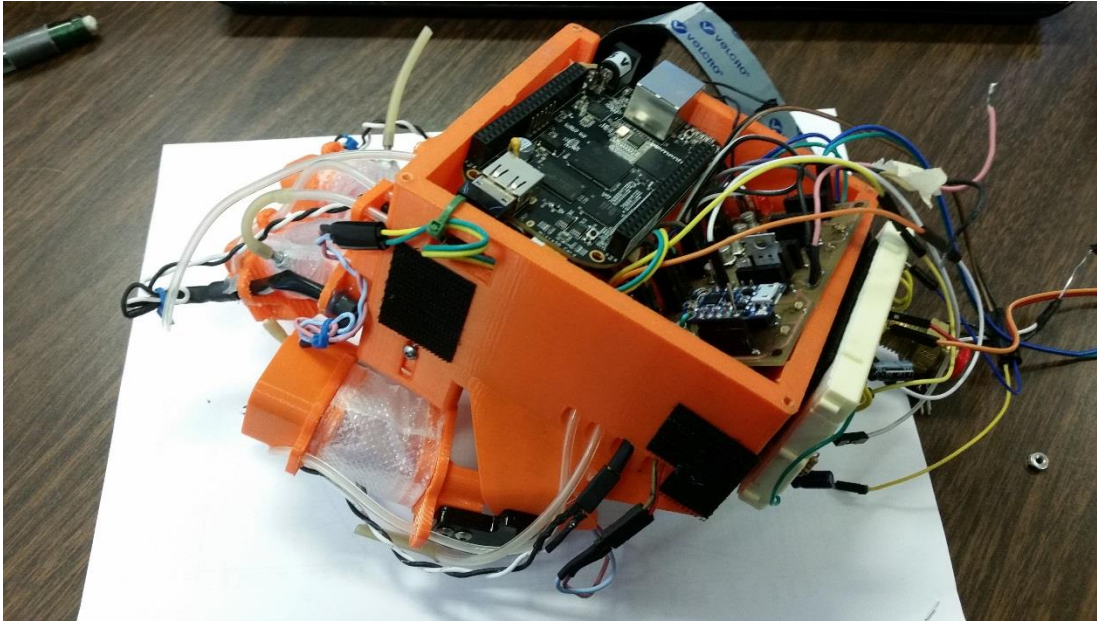
Figure 105: Complete Prototype 1

Vacuum Sealable Plastic Actuators

The final actuator that provided the only completely successful result was the vacuum sealable plastic actuators. The actuators seen in Figure 86. These actuators not only were completely sealed, and could hold over 5 PSI of pressure, but they were very simple to manufacture. The total time of assembly of each actuator was less than 30 mins. The plastic used for the actuators was also durable and not susceptible to tearing. In addition, the actuator was very slim and the thickness of the plastic was small enough that it provided little to no extra resistance to the user's finger, unlike the silicone and surgical tubing methods. In the end or our testing, this was the clear choice for our finger actuators.

Figure 106: Vacuum Sealable Plastic Actuators

Fingertip Actuators

The final product of the fingertip actuator can be seen in Figure 107. As seen in the figure the actuator was successfully molded. There were few to no impurities in the silicone rubber. In addition, the actuator was able to be fully sealed and was able to hold pressure of over 5 psi.

Figure 107: Fingertip Actuator

Finger and Thumb Module

The final figure and thumb modules can be seen in Figure 108. As seen in the figure, the curvature sensor was able to be successfully mounted on the finger and thumb module. However, the tube chosen for the sensor fatigued over time and created a crease in the center of the tube. This caused occasional issues with the functionality of the sensor. The functionality of the finger joint actuators and fingertip actuators were exceptional. The desired sensation of both finger joint resistance and the fingertip depression was very accurate and it was clear to the user that the robot had made contact with an object.
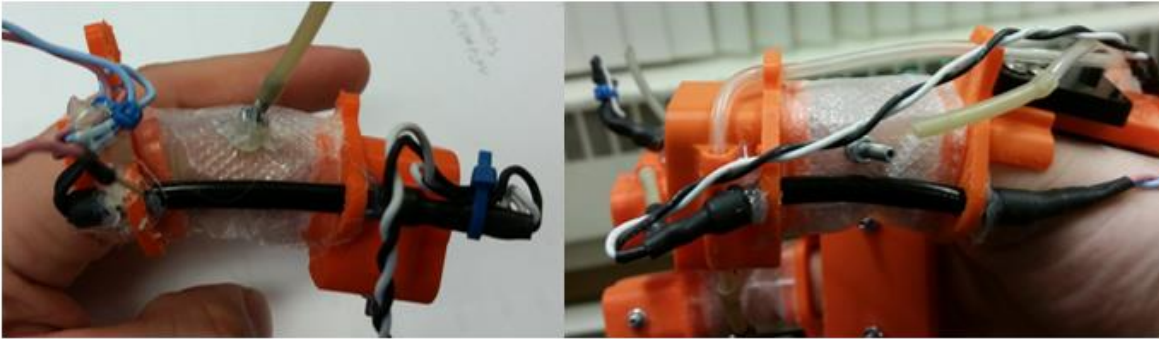
Figure 108: Finger Module (Left) Thumb Module (Right)

Palm Module

The final prototype of the palm module can be seen in Figure 109. The palm touch actuator was able to be mounted within the cavity of the palm module. Also seen on the right of the picture was the finished silicone palm touch actuator. This actuator was able to be successfully molded. The final actuator was fully sealed and was able to hold over 5 psi of pressure. The final result of the feedback provided by the palm module was exceptional. The feeling provided to the user's palm from the actuator almost exactly mimicked a human hand gasping the edge of a table or any object.
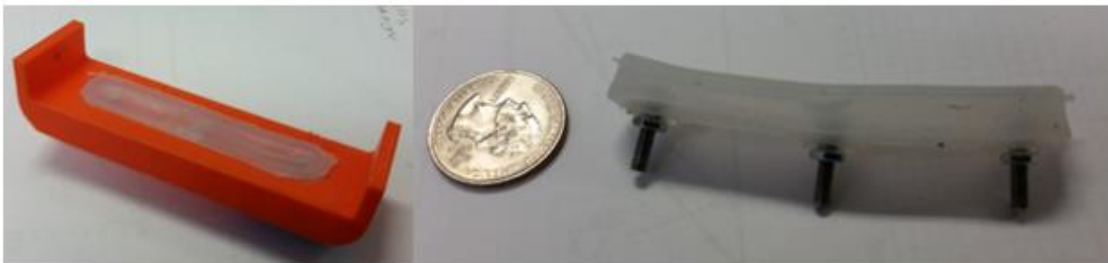
Figure 109: Palm Module (Left) Palm Touch Actuator (Right)

Appendix E HERO Glove Circuit Schematic
Schematic Overview

Power Regulating Circuit Power Outlets	**Palm Finger Microcontroller**	**Thumb Microcontroller**	**Thumb Curvature Sensing**	**Thumb Solenoid Driver**		
SC1	SC5	SC2	SC7	SC8		
SC14						
	Palm Solenoid Driver	Index Finger Microcontroller	Index Curvature Sensing	Index Solenoid Driver		
	SC13	SC3	SC9	SC10		
	Extra GPIO	Alert System Drive Circuitry	Middle Finger Microcontroller	Middle Curvature Sensing	Middle Solenoid Driver	
	SC15	SC6	SC4	SC11	SC12	
BeagleBone Connections	IMU Connection					
	SC16					
	SC17					

Power Regulating Circuitry

SC2 Thumb Microcontroller
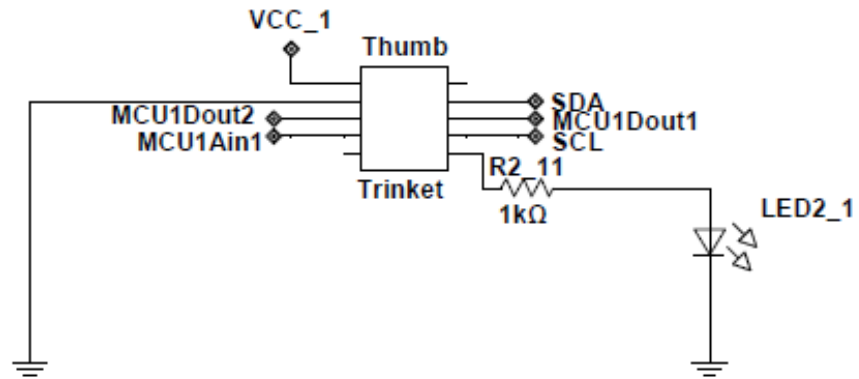
SC3 Index Finger Microcontroller

SC4 Middle Finger Microcontroller

VCC_1

Middle

MCU3Dout2
MCU3Ain1

SDA
MCU3Dout1
SCL

Trinket

R2_31

1kΩ

LED2_3

SC5 Palm Finger Microcontroller

VCC_1

Palm

MCU4Dout02
MCU4Ain01

SDA
MCU4Dout01
SCL

Trinket

R2_41

1kΩ

LED2_4
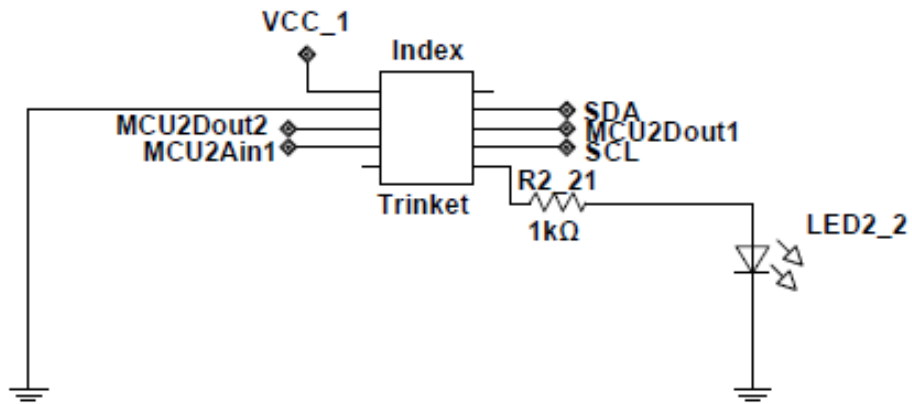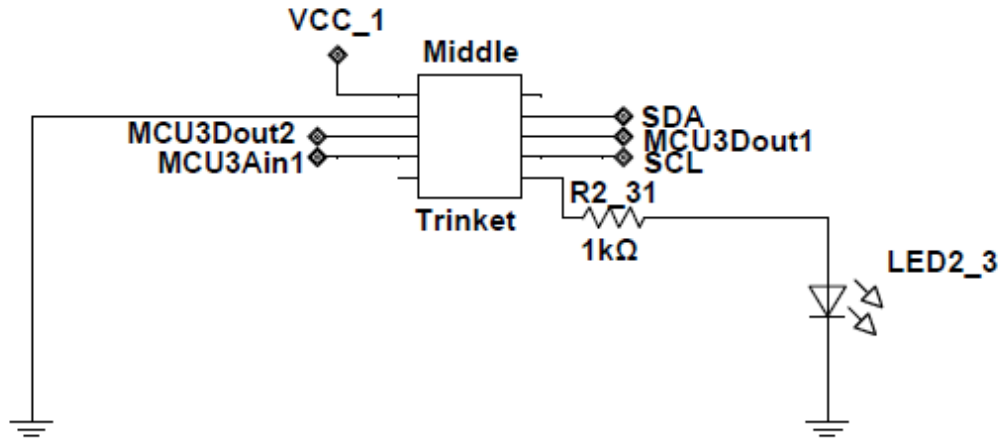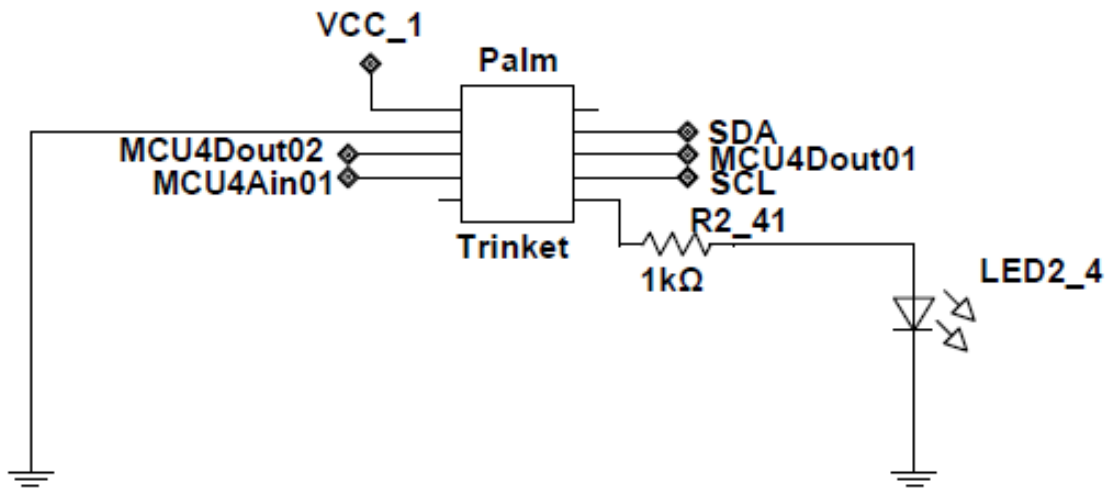
SC6 Alert System Drive Circuitry
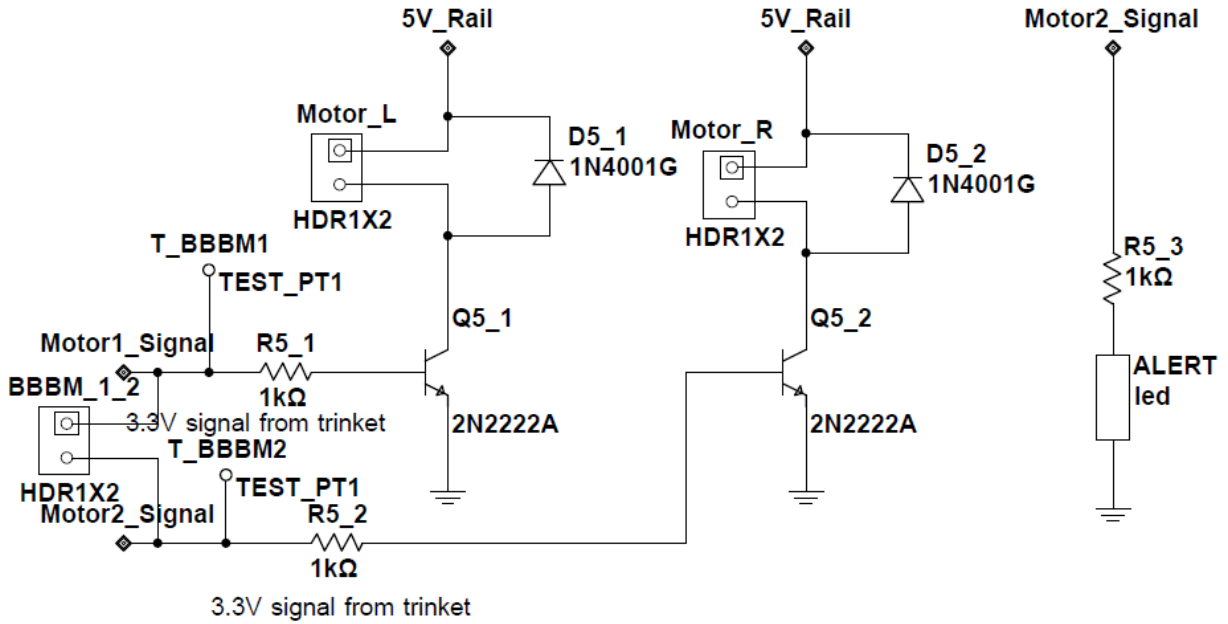
SC7 Thumb Curvature Sensing
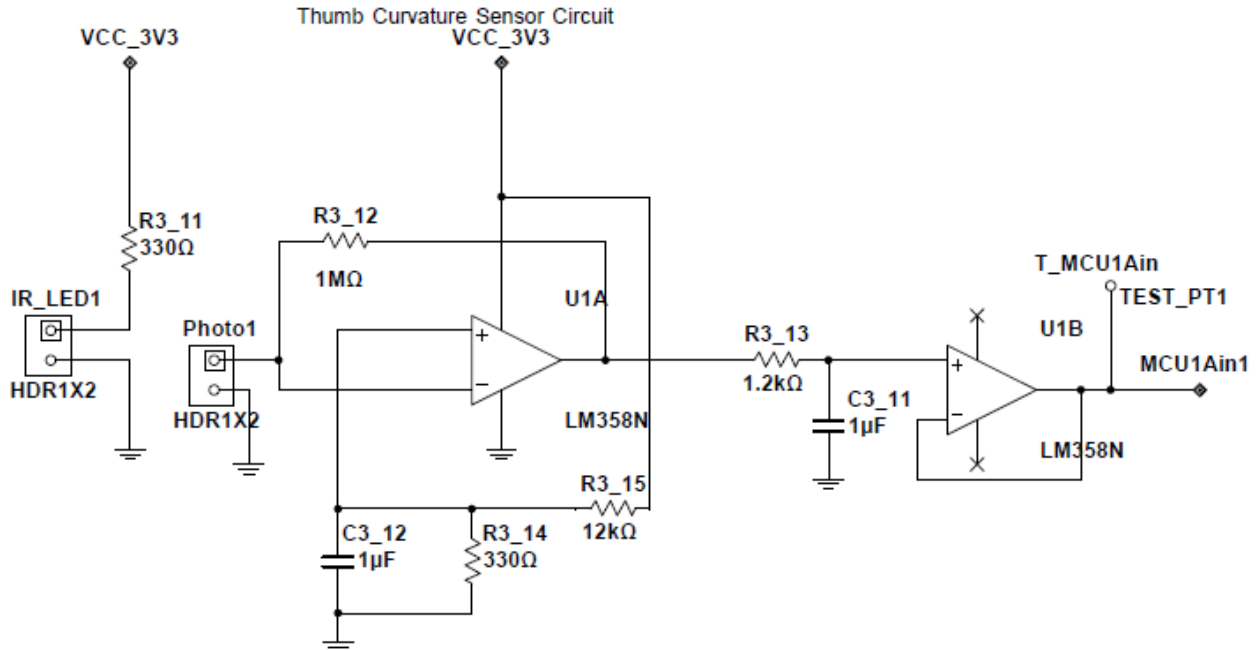
SC8 Thumb Solenoid Driver
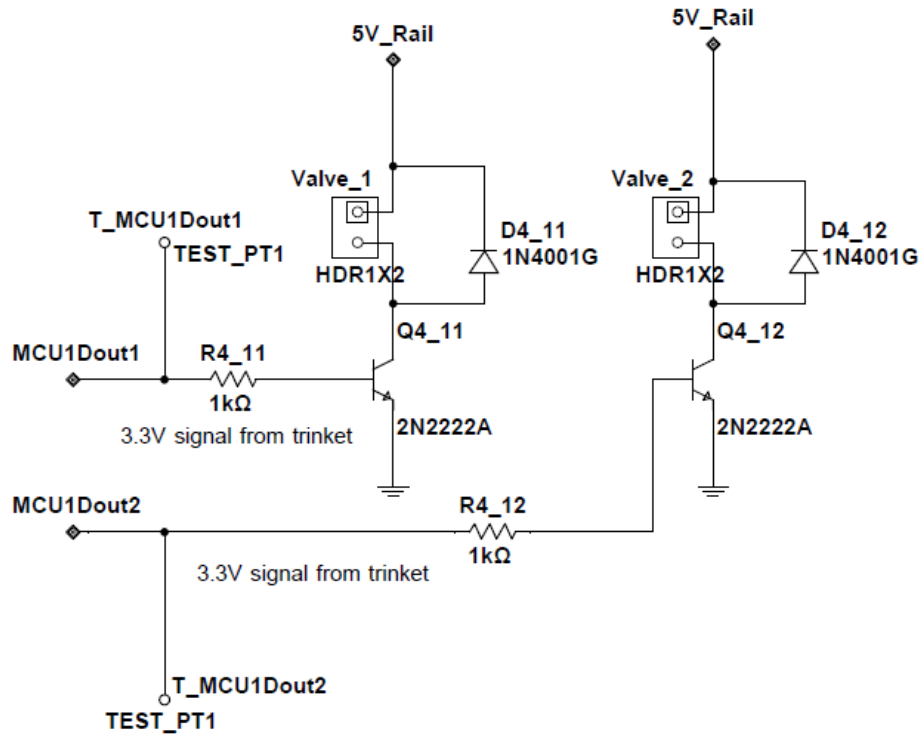
SC9 Index Curvature Sensing

SC10 Index Solenoid Driver

5V_Rail

5V_Rail

Valve_3

Valve_4

D4_21
1N4001G

D4_22
1N4001G

HDR1X2

HDR1X2

T_MCU2Dout1

TEST_PT1

Q4_21

Q4_22

MCU2Dout1

R4_21

2N2222A

2N2222A

1kΩ

3.3V signal from trinket

T_MCU2Dout2

MCU2Dout2

TEST_PT1

R4_22

1kΩ

3.3V signal from trinket

SC11 Middle Curvature Sensing

VCC_3V3

VCC_3V3

R3_32

T_MCU3Ain

TEST_PT1

R3_31
330Ω

1MΩ

U3A

U3B

IR_LED3

Photo3

R3_33

MCU3Ain1

HDR1X2

HDR1X2

1.2kΩ

C3_31
1µF

LM358N

LM358N

R3_35

12kΩ

C3_32
1µF

R3_34
330Ω

SC12 Middle Solenoid Driver

5V_Rail

5V_Rail

T_MCU3Dout1

TEST_PT1

Valve_5

HDR1X2

D4_31
1N4001G

Valve_6

HDR1X2

D4_32
1N4001G

Q4_31

Q4_32

MCU3Dout1

R4_31

1kΩ

2N2222A

2N2222A

3.3V signal from trinket

T_MCU3Dout2

MCU3Dout2

TEST_PT1

R4_32

1kΩ

3.3V signal from trinket

SC13 Palm Solenoid Driver

5V_Rail

T_MCU4Dout1

TEST_PT1

Valve_7

HDR1X2

D4_41
1N4001G

Q4_41

MCU4Dout01

R4_41

1kΩ

2N2222A

3.3V signal from trinket

SC14 Power Outlets

VCC_1

PWR3_1   PWR3_2   PWR3_3

HDR1X2   HDR1X2   HDR1X2

5V_Rail

PWR5_1   PWR5_2   PWR5_3

HDR1X2   HDR1X2   HDR1X2

SC15 Extra GPIO

MCU4Dout02

T_MCU4_Dout2

GPIO_0   TEST_PT1

HDR1X2

MCU4Ain01

T_MCU4_Ain1

GPIO_1   TEST_PT1

HDR1X2

SC16 BeagleBone Black Connections
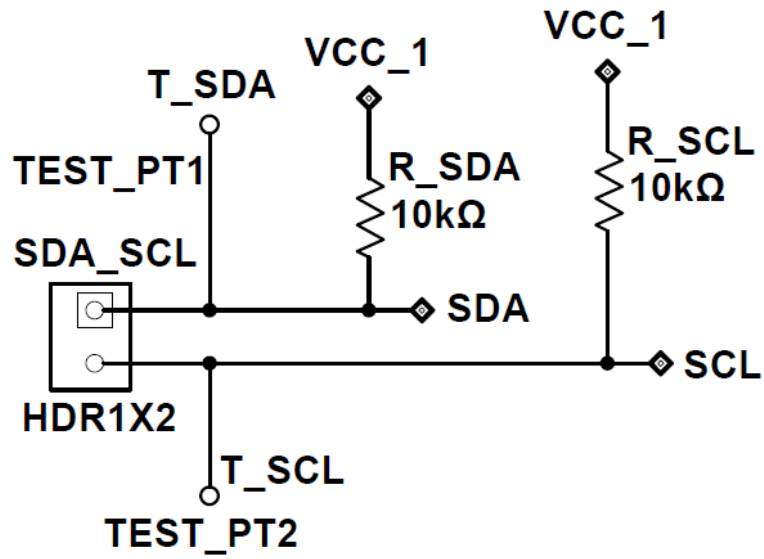
SC17 IMU Connections
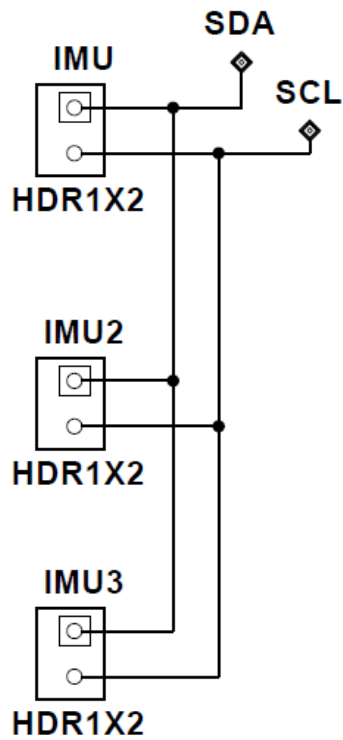
Appendix F HERO Glove Printed Circuit Board Layout
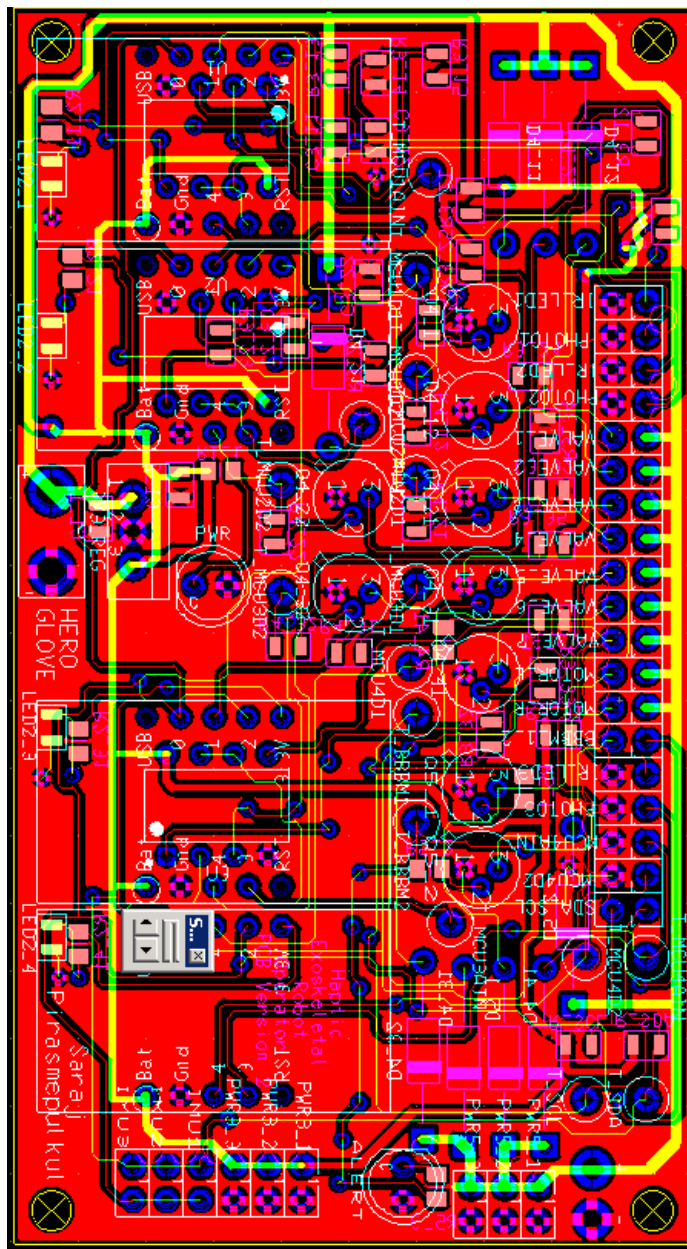
Appendix G HERO Glove PCB List of Components

Capacitor	Package	Quantity Per Board	Digikey Part Number
1u	SMT (0805)	6	311-1365-1-ND
10u	SMT (0805)	2	311-1459-1-ND
Resistor			
1k	SMT (0805)	18	311-1.0KARCT-ND
1M	SMT (0805)	3	311-750KCRCT-ND
1.2k	SMT (0805)	3	RMCF0805JT1K20CT-ND
330	SMT (0805)	6	RMCF0805JT330RCT-ND
10k	SMT (0805)	2	311-10KARCT-ND
12K	SMT (0805)	3	311-12KARCT-ND
LED			
5 mm Green	TH	1	C503B-GCN-CY0C0791-ND
5 mm Red	TH	1	C5SMF-RJS-CT0W0BB2CT-ND
Green	SMT (0805)	4	160-1423-1-ND
IR LED	TH	3	751-1201-ND
Photo transistor	TH	3	160-1030-ND
Diodes			
1N4001	TH	9	641-1310-1-ND
Transistor			
2N2222	TH	9	2N2222ACS-ND
Op-Amp			
LM358N	DIP18	3	LM358NFS-ND
Power			
LD1117V33	TH	1	497-1491-5-ND
Connectors			
Header	0.1"	50	929400E-01-36-ND
DIP18	TH	3	AE9986-ND
Female Socket for Trinket	TH	4	S7038-ND
Microcontrollers			
AdafruitTrinket	TH	4	1528-1020-ND

Appendix H Initial Position Control with Single IMU

In the original design, we used a single IMU to capture the hand pose and position.

Orientation

The accelerometer measures the magnitude and direction of gravity. The gyroscope measures angular velocity. The magnetometer measures the magnitude and direction of the Earth's magnetic field. These three sensors combine creates the quaternion that accurately represents the orientation of the user's hand.

Displacement

Displacement of the body can be obtained by integrating the linear acceleration of the body twice. However, the output of the accelerometer from the IMU measures the magnitude and direction of the gravity instead of linear acceleration, therefore the output from the accelerometer always contains the gravity constant. Therefore, to get the linear acceleration, we have to eliminate the gravity constant from the readings. Given the gravity values measured by the accelerometer:

However, since there are always bias errors in the measurement from the accelerometer. Therefore, these error terms accumulate exponentially over time after each double integration which results in a lot of drifts in displacement.

Solutions to eliminate drifting

Although determining position from an accelerometer is not accurate and is a well known problem, there are several reasons that it is appropriate in our application. The user will have user feedback from the real robot and therefore would learn to adapt to the accelerometer drift by correcting their movements based on the camera feedback. The user is able to reset the reading of the accelerometer over time to get rid of the bias errors. It is a low-cost solution and can be simply attached on the human hand so we made a few attempts to get rid of the errors.

Kalman filter

One common approach to correct the error is to use a Kalman filter to estimate the position over time. We attempted to use this method and design a Kalman filter for the IMU. But we later found that Kalman filter requires a state transition model of the state variables in order to estimate the current position based on the previous state, which is commonly used in mobile robots or aircrafts. It is not appropriate in our application, however, because we are tracking the complex motion of a human hand, which is very difficult to constrain and determine a transition model.

High Pass Filter

A simpler solution developed by Sebastian Madgwick to reduce the drift is to use two high pass filters. The first one is applied to the velocity to remove any drifts after the integration from acceleration. The filtered velocity is used to calculate the displacement. After the integration to get displacement, another high pass filter is applied. One neat feature using a high

pass filter is that if the IMU is stationary, its position will slowly pull back to zero. However, the IMU that was used to demonstrated by Madgwick is the x-IMU which is a very high quality and expensive device. It did not give the best results in our case.

Human in the loop control

In our system, the glove and the robot will be constantly communicating with each other, therefore we have the property of human-in-the-loop control. Taking advantage of this property, the user can be trained with the glove to correct himself from the drift.

Real Model Correction

The drift can also be eliminated using the position of the real robot. Using the real position of the end effector of the robot, it can be fed back to correct the drifting position model on our glove.

Resetting Velocity Model

Drifts occur when error accumulates over time from the result of the double integration. Therefore, it is necessary to reset the velocity after a certain amount of time and then continues sampling. This way, we will have the velocity constantly zeroed out and prevent the displacement to accumulate too fast.

Pause Button

Another feature that we implemented in combination with the solutions above to prevent drifting is to add a hold button. When the button is released, the program will stop sampling from the IMU, the velocity will be set to 0, which indicates that there is no displacement, but orientation is still active. When the button is push, the program will start sampling as normal. For this to be effective, the user will release the button when the glove needs to be stationary and only push the button when making movements.

Although these solutions had some significance in getting rid of the drift, we realized that it would be difficult to control the robot with such low precision in position control. Therefore, instead of trying to fix the problem, we used an alternative solution that does not provide this problem.

3 Port Solenoid Valve
Series *S070*/Body Ported Manifold
Stacking Type Specifications

How to Order Manifold

Body ported manifold stacking type

SS07 **3** **M01**–**08** **C**

Ports

3	3 port

Port size

Symbol	SUP/EXH port (Applicable tubing)	OUT port	
			Applicable tubing
M01	Barb fittings (ø6/ø4)	Barb fittings	ø3.18/ø2
M02			ø4/ø2.5

Note) The outside and inside diameters of the "applicable tubing" are indicated for the barb fitting.

Stations

02	2 stations
03	3 stations
:	:
20	20 stations

Note) Maximum of 20 stations

Electrical entry

C	Grommet/Plug lead

How to Order Manifold Assembly

Enter the part numbers of the valves and options to be mounted below the manifold base part number.

<Example>
SS073M01-04C __ 1 set ——— Manifold Base no.
∗S070M-5BG-32 __ 4 set ——— Valve no.

→ Prefix the symbol "∗" to the solenoid valve part number.

Write sequentially from the 1st station on the D side.
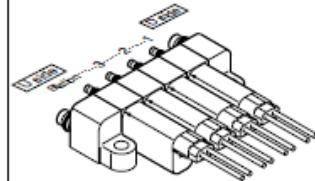
How to Order Valves

S070 **M**–**5** **B** **G**–**32**

Body type

Symbol	Body type
M	Body ported stacking manifold type

Coil voltage

5	24 VDC
6	12 VDC
V	6 VDC
S	5 VDC
R	3 VDC

Port size

Symbol	Connection	Applicable tubing
32	Barb fitting	ø3.18/ø2
40		ø4/ø2.5

Electrical entry

G	Grommet
C	Plug lead with light/surge voltage suppressor

Power consumption – Pressure specification – Flow rate

Symbol	Power consumption (W)	Maximum operating pressure (MPa)	Cv factor
A	0.35	0.1	0.021
B		0.3	0.011
C	0.5	0.3	0.021
D		0.5	0.011
E [Note]	0.1 (With power saving circuit)	0.1	0.011
F [Note]		0.3	0.006

Note) An option only applicable to 24 VDC plug lead type.

4-11-12

SMC

134

Dimensions

Body ported stacking type manifold

SS073M$^{01}_{02}$- Stations C

ø6 barb fitting
1 (SUP) port

n-ø3.2, ø4 barb fitting
2 (OUT) port

ø6 barb fitting
3 (EXH) port

2-ø3.4
Mounting
hole

0.1 W with power
saving circuit

≦150 (Lead wire AWG#26)

≦150 (Lead wire AWG#26)
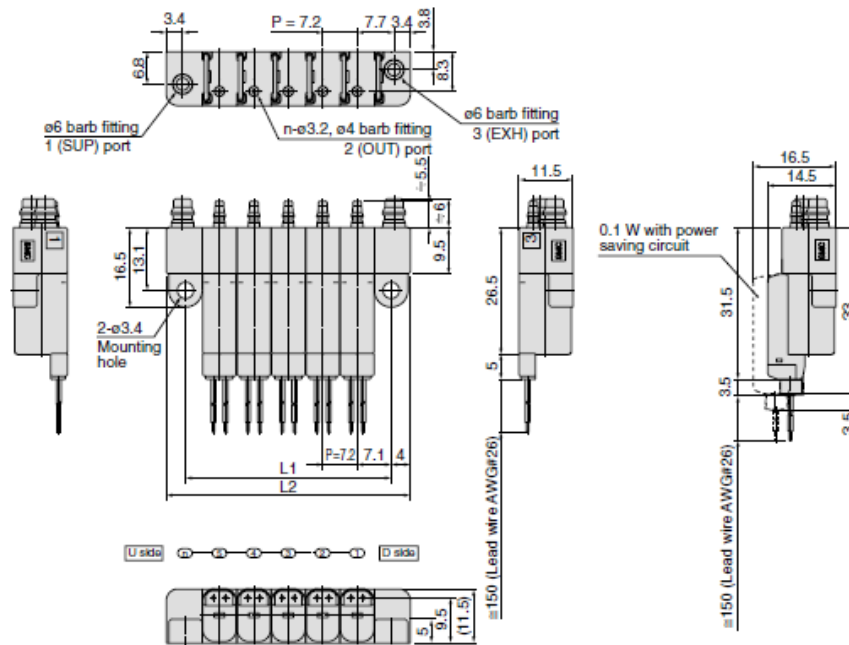
| V100 |
| SY |
| SYJ |
| VK |
| VZ |
| VT |
| VP |
| VG |
| VP |
| **S070** |
| VQ |
| VKF |
| VQZ |
| VZ |
| VS |
| VFN |

Dimensions

Formulas: L1 = n x 7.2 + 7, L2 = n x 7.2 + 15, n: Stations (maximum 20 stations)

n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
L1	21.4	28.6	35.8	43	50.2	57.4	64.6	71.8	79	86.2	93.4	100.6	107.8	115	122.2	129.4	136.6	143.8	151
L2	29.4	36.6	43.8	51	58.2	65.4	72.6	79.8	87	94.2	101.4	108.6	115.8	123	130.2	137.4	144.6	151.8	159

Appendix J   HERO Glove Exploded Views
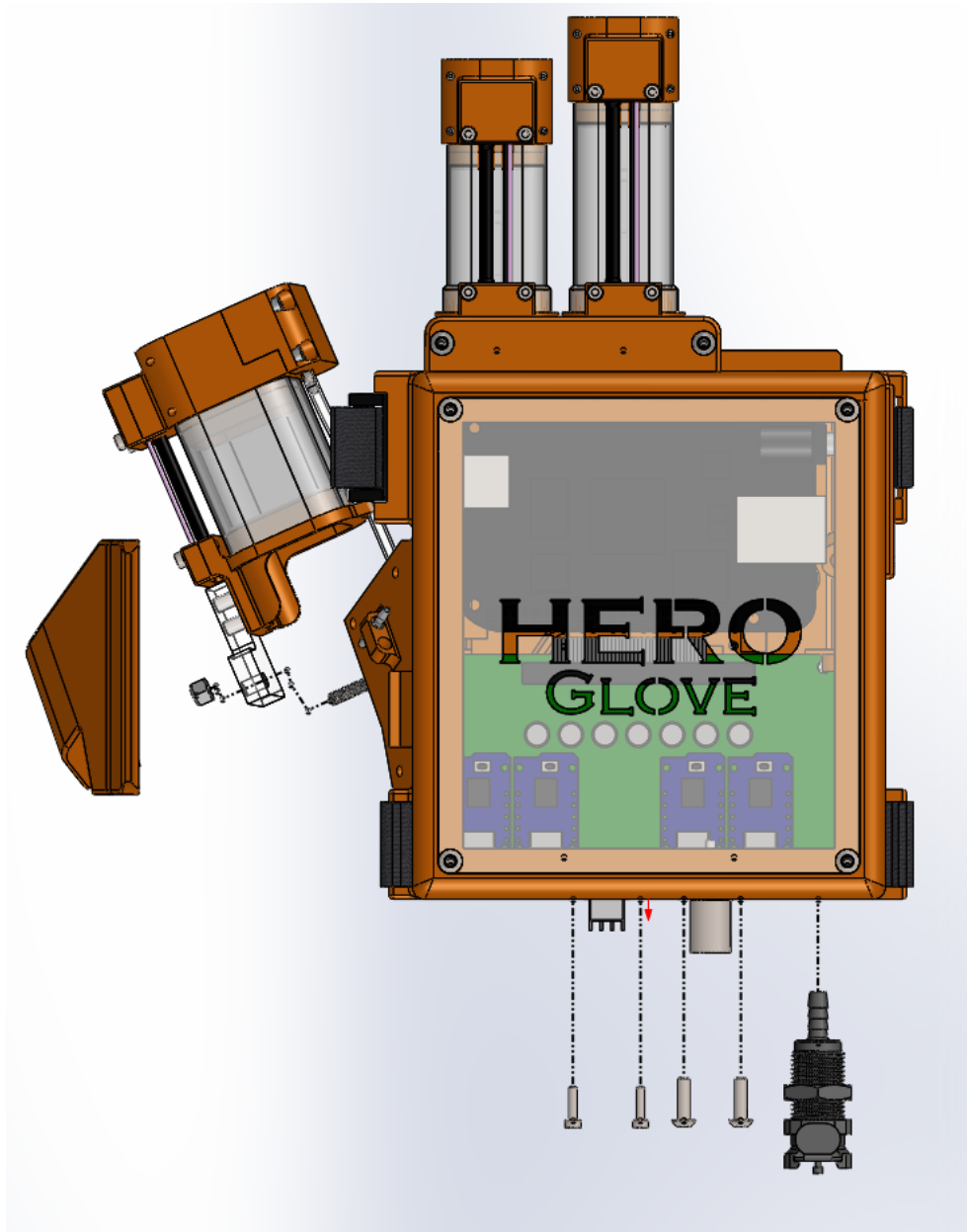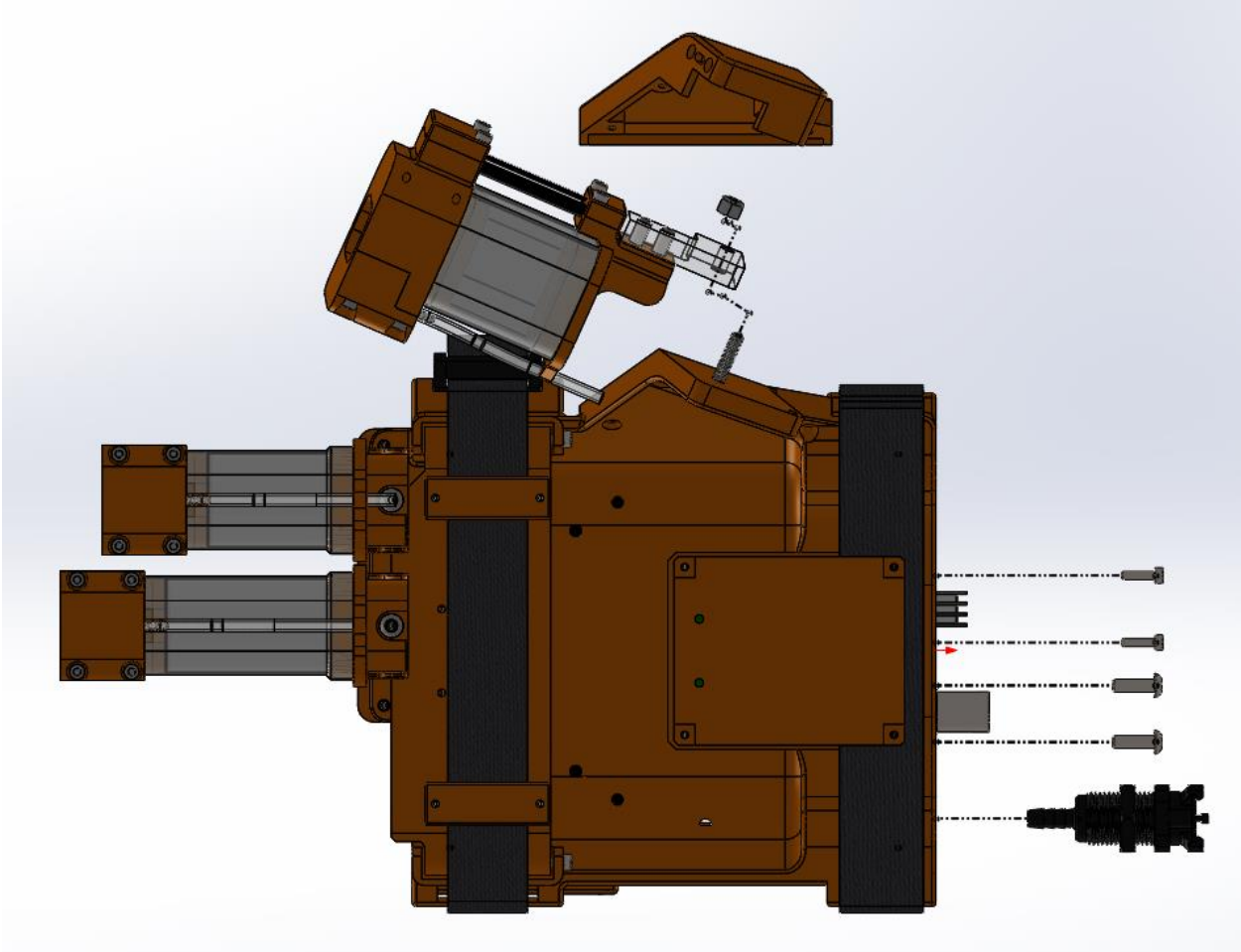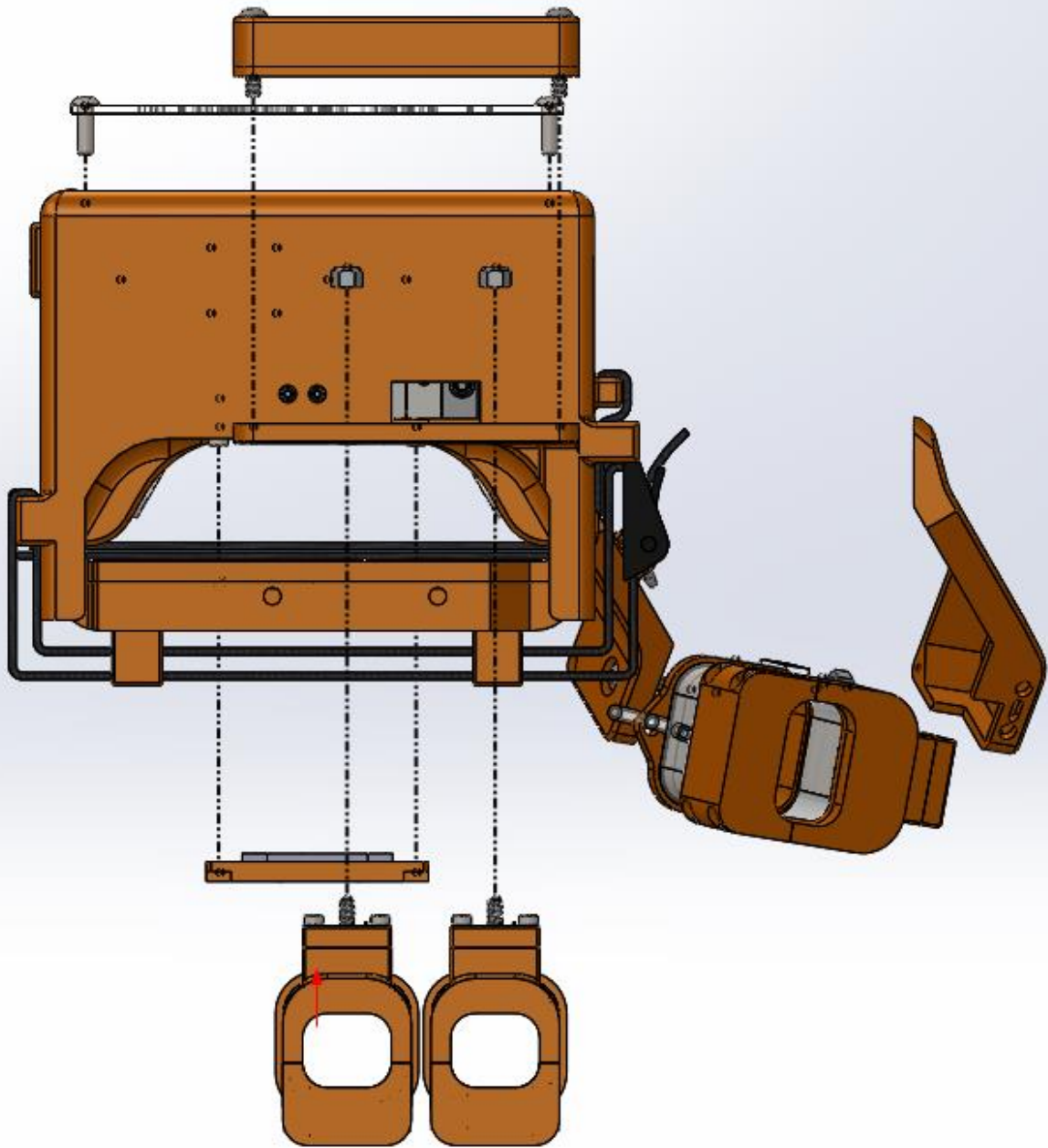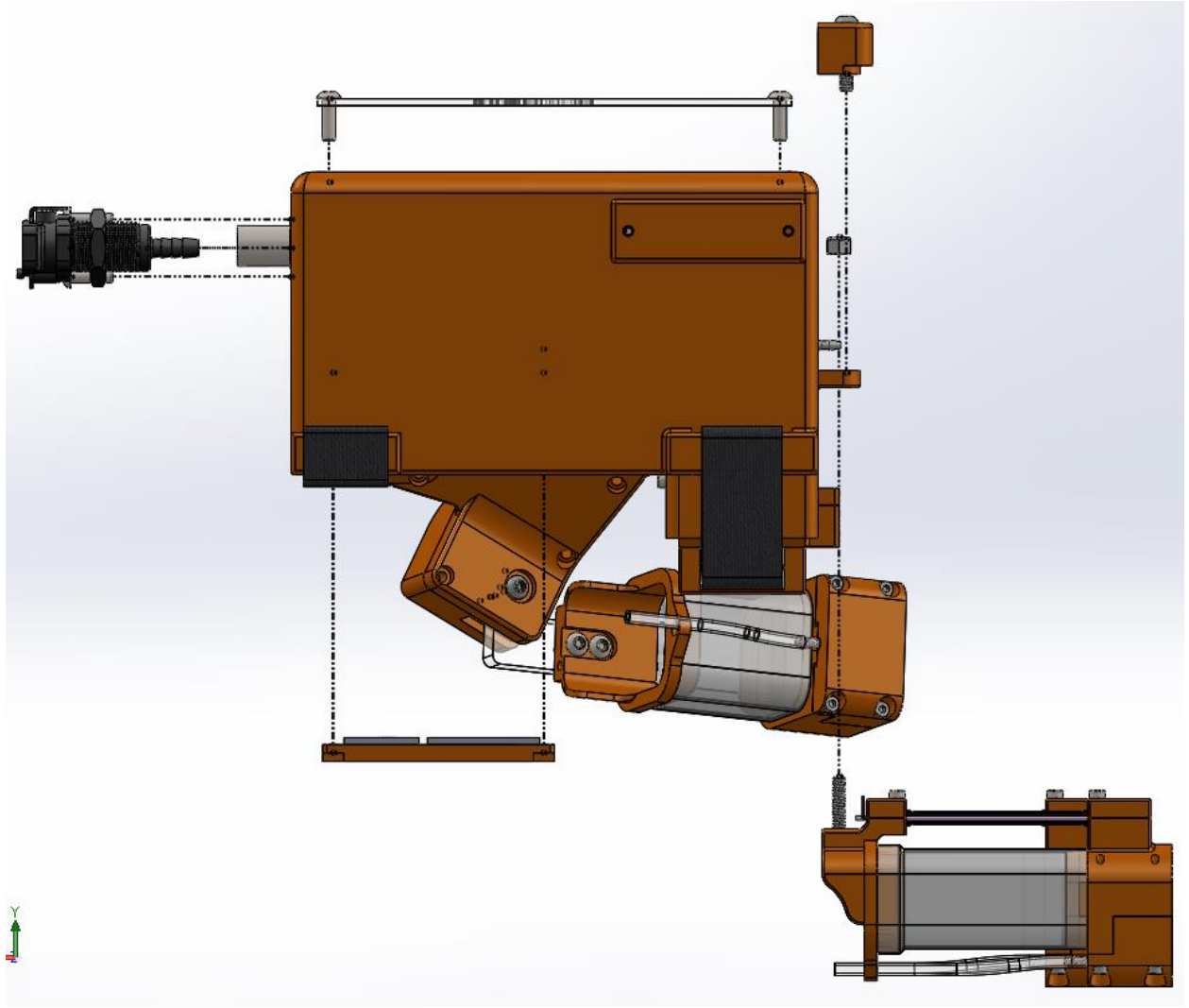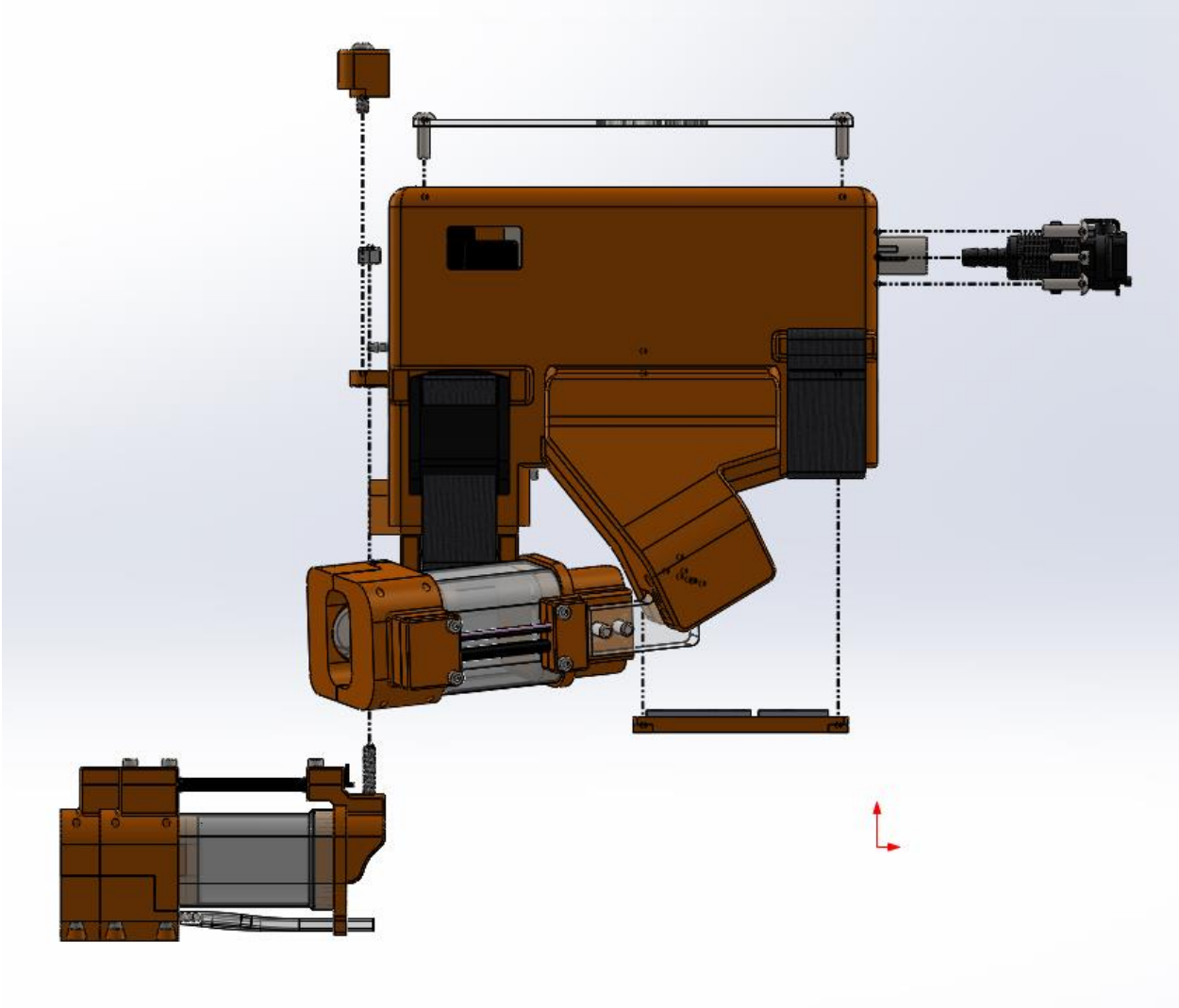
Appendix K HERO Glove Bill of Materials

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.	MANUFACTURER
1	Glove Shell		1	
2	SS073M01-07C_SS073M d sIDE	Solenoid Valve Manifold	1	SMC
3	94500A223	M3X16 BHCS	5	MCMASTER CARR
4	Top Cover		1	
5	Vibration Motors		2	
6	Finger Module Base		2	
7	Finger Joint Actuator		1	
8	Finger Module Tip Top		2	
9	Finger Module Tip Bottom		2	
10	Fingertip Actuator Silicone		3	
11	94610A420	M2 FLAT WASHER	12	MCMASTER CARR
12	93235A081	2-56 X 0.5 VENTED SHCS	6	MCMASTER CARR
13	90480A003	2-56 HEX NUT	6	MCMASTER CARR
14	91290A017	M2X20 SHCS	17	MCMASTER CARR
15	Fingertip Sensor Cover		3	
16	Finger Tip Actuator Tube		3	
17	Base Sensor Cover		3	
18	Curvature Sensor Tube Finger		1	
19	Photoresistor		3	
20	LED		3	
21	92290A015	M2X8 SHCS	32	MCMASTER CARR
22	Finger Joint Actuator		1	
23	Curvature Sensor Tube Finger		2	
24	Palm Module Bottom		1	
25	Palm Module Top		1	
26	Palm Touch Actuator Silicone		1	
27	2974K431	BARBED TEE CONNECTOR	2	MCMASTER CARR
28	Strap Guide		2	
29	2808K101	BARBED STRAIGHT CONNECTOR	5	MCMASTER CARR
30	92290A015	M2X6 SHCS	2	MCMASTER CARR
31	Quick-Connect Coupler Female	Quick-Connector Coupler Female	1	MCMASTER CARR
32	Quick-Connect Coupler Male	Quick-Connector Coupler Male	1	MCMASTER CARR

33	BBB Power Supply Cable		1	
34	A19431-ND	HEAD 4 PIN CONNECTOR	1	DIGIKEY
35	IMU Arm Connector Plate		1	
36	Finger Mount Cover		1	
37	90576A102	M3 LOCK NUT	3	MCMASTER CARR
38	Thumb Joint Actuator		1	
39	Thumb Module Tip Bottom		1	
40	Thumb Module Tip Top		1	
41	Thumb Module Base		1	
42	Thumb Connector Plate		1	
43	94500A223	M3X10 BHCS	20	MCMASTER CARR
44	Thumb Flange Cover		1	
45	92497A200	M3 NUT	4	MCMASTER CARR
46	Ethernet Cover		1	

Appendix M Software High Level System Flowchart
High Level HERO Glove System Flowchart
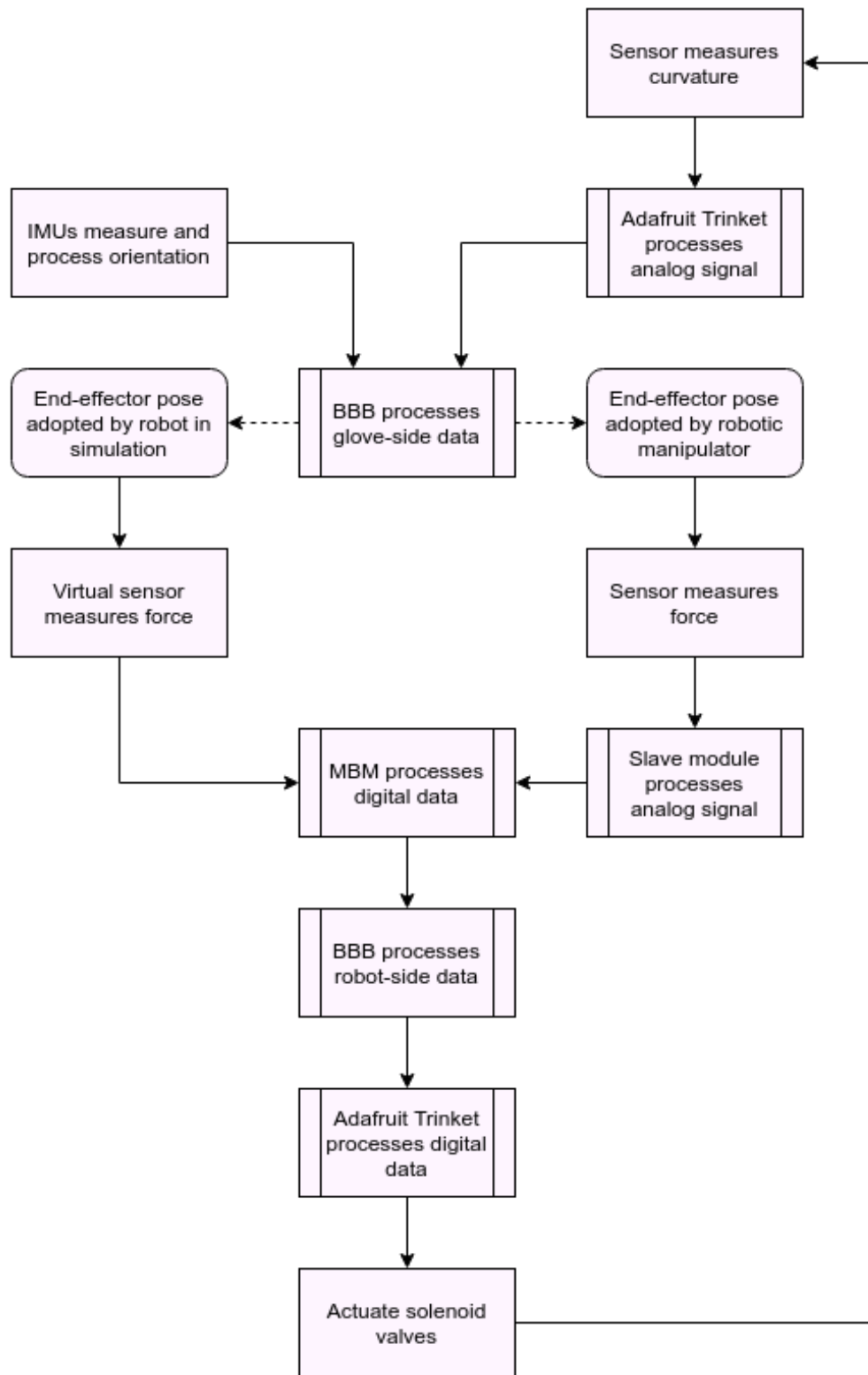
Adafruit Trinket Processes Analog Signal

Receive analog signal

Use ADC to convert analog signal to digital signal

Send digital signal over I2C bus
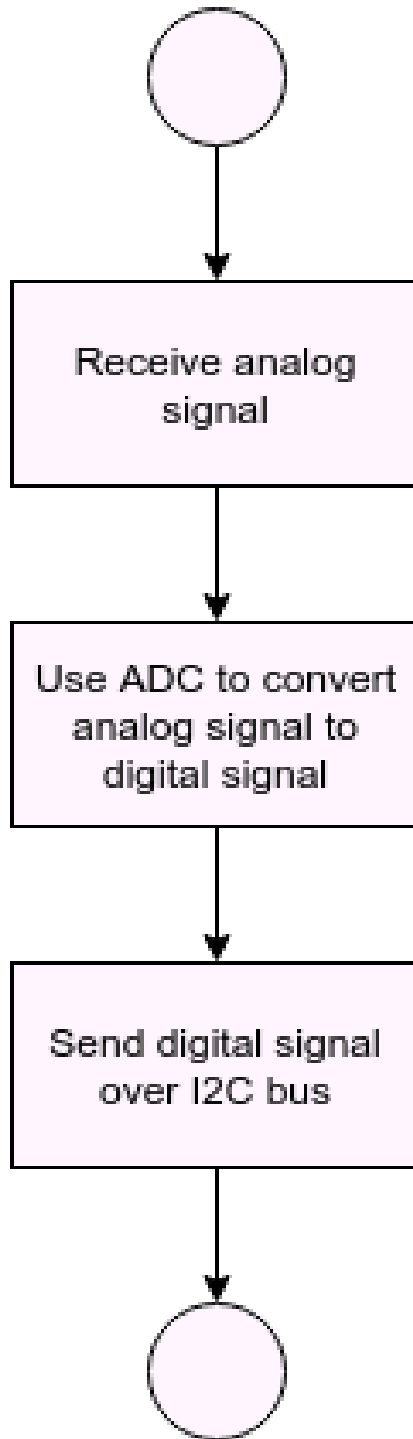
BBB Processes Glove Side Data
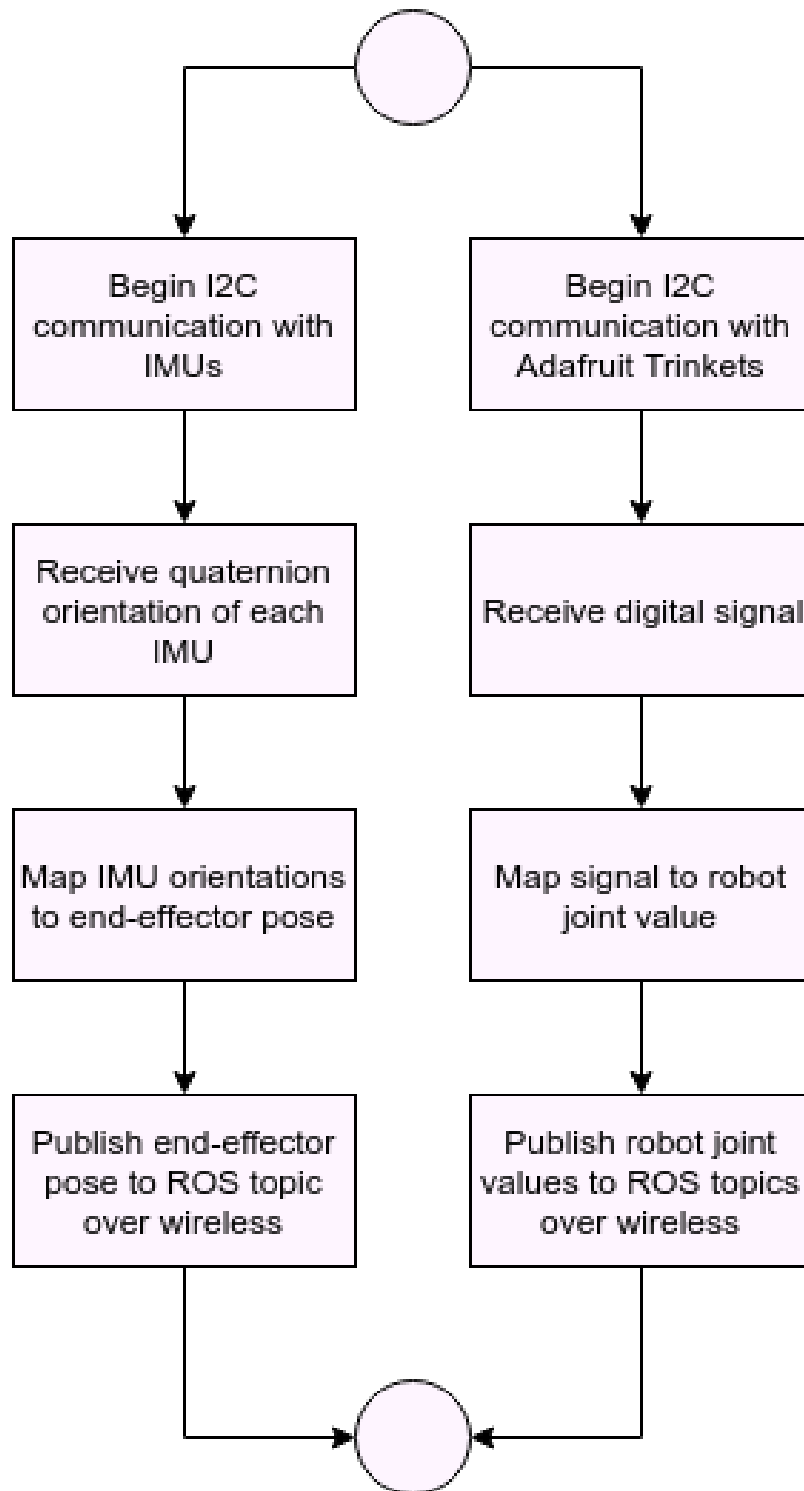
Slave Module Processes Analog Signal

```
                    ( )
                     |
                     v
          +---------------------+
          |  Initialize Bluetooth |
          +---------------------+
                     |
                     v
          +---------------------+
          |  Wait for connection |
          |     request from     |
          |  MinnowBoard MAX     |
          +---------------------+
                     |
                     v
          +---------------------+
          |   Receive analog     |<----+
          |  signal from tactile |     |
          |      sensors         |     |
          +---------------------+     |
                     |                 |
                     v                 |
          +---------------------+     |
          |    Amplify signal    |     |
          +---------------------+     |
                     |                 |
                     v                 |
          +---------------------+     |
          |  Map signal to valve |     |
          |     duty cycle       |     |
          +---------------------+     |
                     |                 |
                     v                 |
          +---------------------+     |
          |   Format data into   |     |
          |     string format    |     |
          +---------------------+     |
                     |                 |
                     v                 |
          +---------------------+     |
          |   Format data into   |     |
          |     string format    |     |
          +---------------------+     |
                     |                 |
                     v                 |
          +---------------------+     |
          |    Send data to      |-----+
          |  MinnowBoard MAX     |
          |  through Bluetooth   |
          +---------------------+
                     |
                     v
                    ( )
```

146
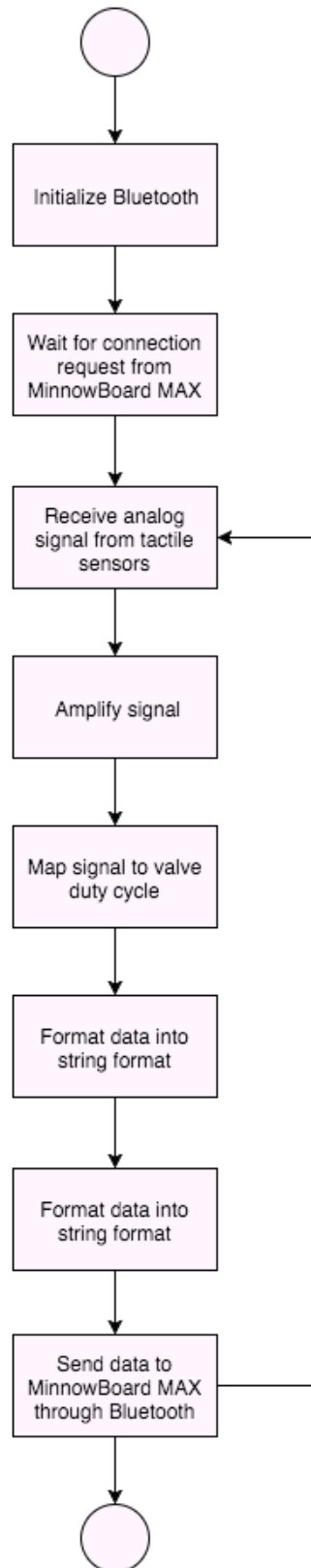
MBM Processes Digital Data
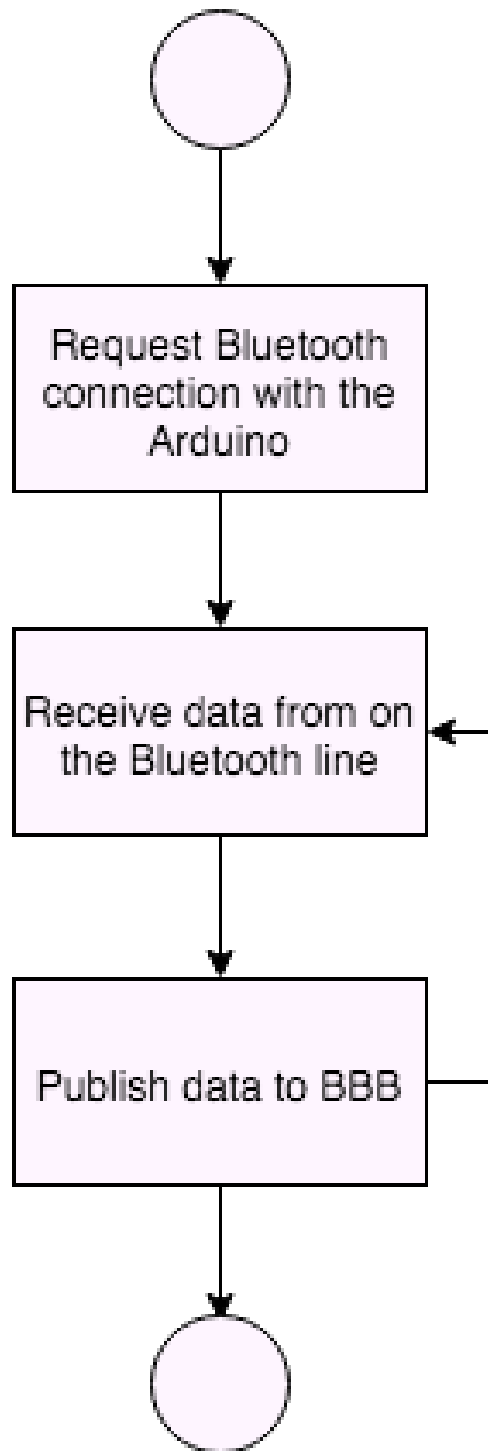
BBB Processes Robot Side Data

Receive digital
signals from MBM
over wireless

Map digital signals to
valve duty cycles

Update valve duty
cycles phase

Send digital signals
to Adafruit Trinkets
over I2C bus
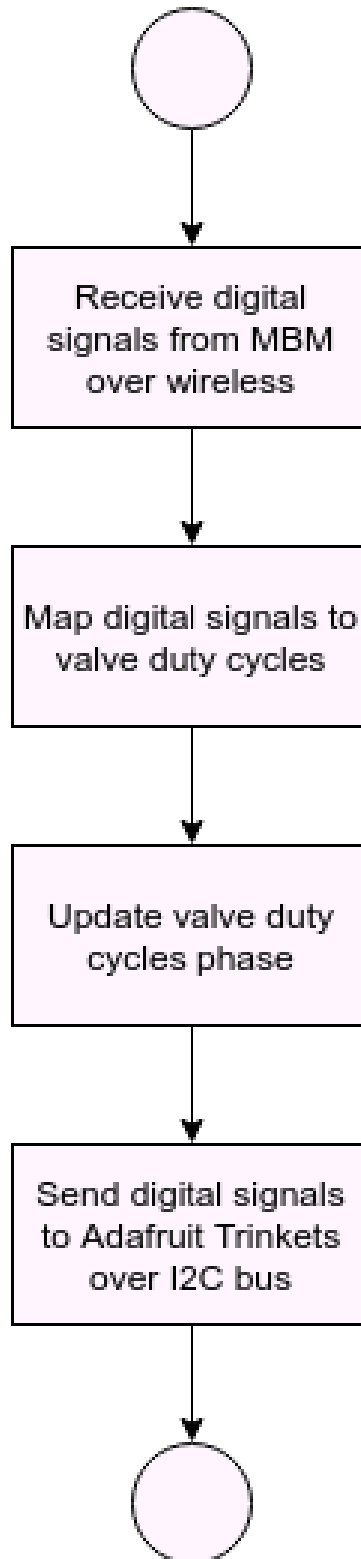
Adafruit Trinket Processes Digital Data
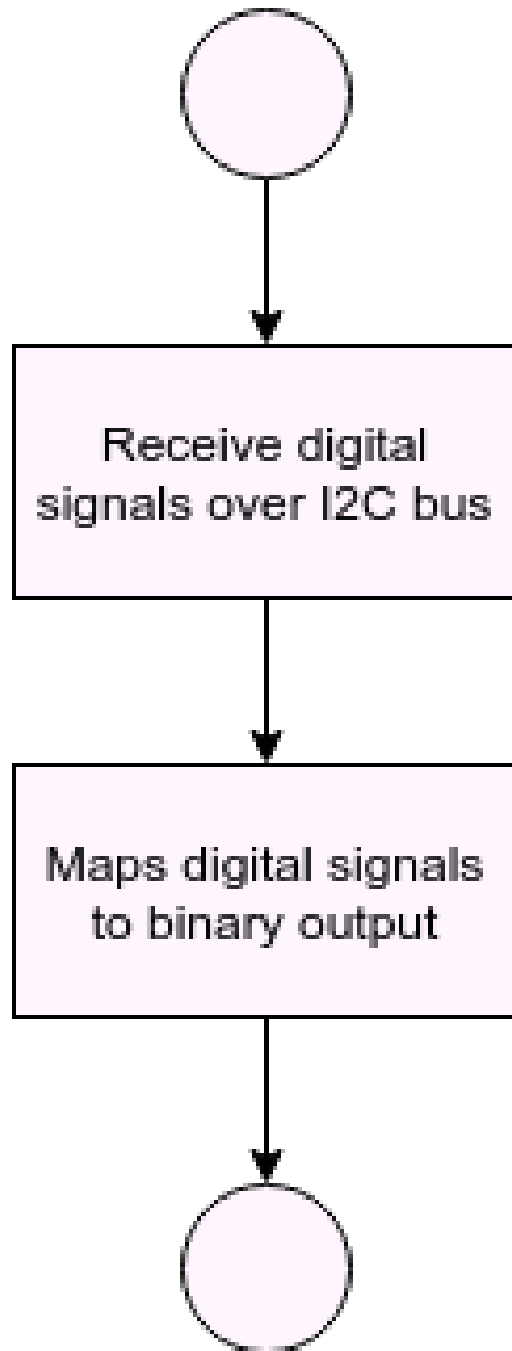
Appendix N HERO Glove Component Drawings
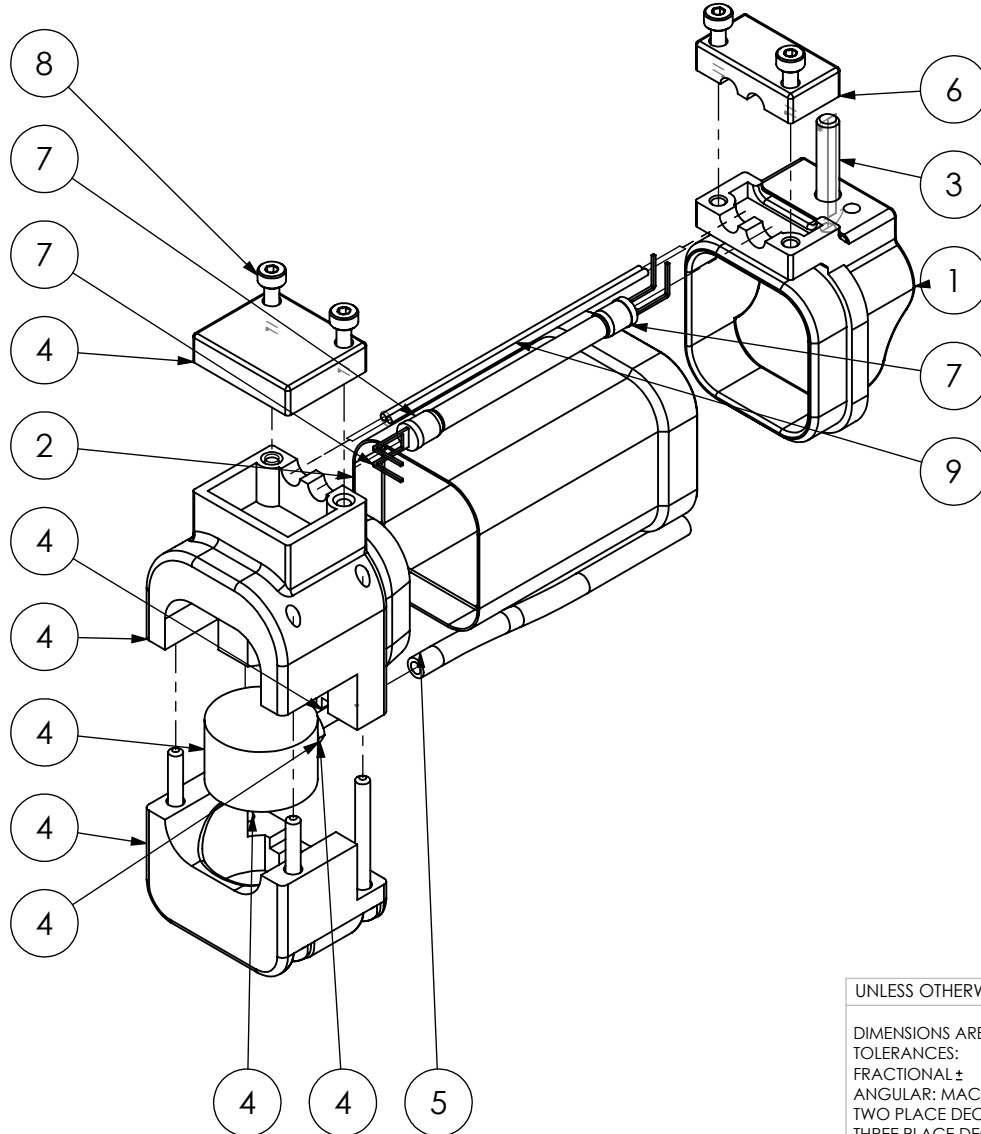
ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Finger Module Base	FINGER MODULE BASE	1
2	Finger Joint Actuator		1
3	94500A223	M3X16 BHCS	1
4	Finger Module Tip Assy		1
5	Finger Tip Actuator Tube		1
6	Base Sensor Cover		1
7	Finger Curvature Snesor		1
8	92290A015	M2X8 SHCS	4
9	Finger Wires		1

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A**

DWG. NO.
Finger Module

REV

SCALE: 1:1   WEIGHT: SHEET 1 OF 1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Thumb Joint Actuator		1
2	Thumb Module Tip Assy		1
3	Thumb Module Base		1
4	Fingertip Sensor Cover		1
5	Base Sensor Cover		1
6	Thumb Connector Plate		1
7	94500A223	M3X10 BHCS	2
8	Finger Tip Actuator Tube		1
9	Curvature Sensor Tube Finger		1
10	Finger Wires		1
11	92290A015	M2X8 SHCS	4
12	91290A017	M2X20 SHCS	4

UNLESS OTHERWISE SPECIFIED:

DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL±
ANGULAR: MACH±   BEND ±
TWO PLACE DECIMAL   ±
THREE PLACE DECIMAL ±

INTERPRET GEOMETRIC
TOLERANCING PER:

MATERIAL

DO NOT SCALE DRAWING

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A**

DWG. NO.
Thumb Module

REV

SCALE: 1:1   WEIGHT: SHEET 1 OF 1

5   4   3   2   1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Palm Module Bottom		1
2	Palm Module Top		1
3	Palm Touch Actuator		1
4	2974K431	BARBED TEE CONNECTOR	2
5	Bottom of Hand Tube 1		1
6	92290A015	M2X8 SHCS	10
7	Bottom of Hand Tube 3		1
8	Bottom of Hand Tube 2		1
9	Bottom of Hand Tube 4		2
10	Strap Guide		2
11	91290A017	M2X20 SHCS	2
12	2808K101	BARBED STRAIGHT CONNECTOR	2

112
49
6
6.74
Ø12 THRU
4X Ø 2.75 THRU
14.50
6.74
13.49
71.71
21.71
2XR 22
13.49
2X R 10
31.50
40.25
60.00°

6
41
10
2X 7
7
8
6
18
15

40
2.50
35
M2X0.4 THRU
A
A
4
58
50
2.50

SECTION A-A

6
2.25
1.50
1.50
31.50
9
6
9
2.25
47
109

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE: **A**

DWG. NO.
Glove Shell

REV

SCALE: 1:2   WEIGHT: SHEET 1 OF 5

5   4   3   2   1

SECTION B-B

B

B

5   4   3   2   1

M2X0.4 THRU

107
3
4

C
C

Ø7 THRU
3.50

9

4X R 4.50

3
12

SECTION C-C

30.00°

20.49

8.50

SECTION E-E

3

94 100

D
D

5

8.50

23.50   4.50
4.50

Ø3

E

12.50   Ø2.20

2.20

3

45

113

6

6

8.18

16   14

2X R 4

R3.50

1

8

SECTION D-D

50

6

44

6

6

10

2X Ø 3.50 THRU

SECTION F-F

F

6   42.14

8

72.93°

128.96°

60.14

F

50.68°

SIZE   DWG. NO.   REV
A   Glove Shell
SCALE: 1:2   WEIGHT: SHEET 3 OF 5

5   4   3   2   1

Top-left view dimensions: 66.04, 4.18, 3.18, 41.91, 48.26, 11.13

Top-right view dimensions: 21.26, 21.75, 3.50, 6, 7, 3.50, 6, 7.75, 33, 2X M3X0.5 THRU, 57.40, 15.30

Bottom-left view dimensions: 8, 4, 7, 3.50, 8.18, 11.35, 12, 11.13, 4X M3X0.5 THRU, 12, 3, 3, 6, 6

Bottom-right view dimensions: 4X M3X0.5 THRU, 45.13, 3.54, 87.92, 3.89

SIZE	DWG. NO.		REV
A	Glove Shell		

SCALE: 1:2 WEIGHT: SHEET 4 OF 5

5 4 3 2 1

G

G

SECTION H-H

32

SECTION G-G

3

40

8

14.80

8.50

10

2X 2.50

4

H

H

2X 9

3.50

3.50

H

H

SIZE	DWG. NO.		REV
A	Glove Shell		
SCALE: 1:2	WEIGHT:		SHEET 5 OF 5

5   4   3   2   1

11

1.50

3.50

18

14

2

5

3

2X ∅ 2.50 THRU

2.25

2.25

13.50

6.50

5

A

A

2X R 1.63

9.50

4.50

3

SECTION A-A

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE
A

DWG. NO.
Base Sensor Cover

REV

SCALE: 4:1   WEIGHT: SHEET 1 OF 1

5

4

3

2

1

46

5.50

20.50

6.50

3.25

⌀ 2.50 THRU

R1.25

3X R 2.75

2.50

5.50

3.75

2

2

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A**

DWG. NO.
Ethernet Cover

REV

SCALE: 2:1   WEIGHT: SHEET 1 OF 1

5   4   3   2   1

22

22

4X R 5

TYP 1.80

49.70

25.60

1.15

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A**

DWG. NO.
Finger Joint Actuator

REV

SCALE: 1:1   WEIGHT: SHEET 1 OF 1

5　　　4　　　3　　　2　　　1

18
14
11
2X 2
1.50
5
9.50
2.25
13.50
3
2.25

∅ 3.50 THRU
10
4
B   B
8
18

SECTION A-A

6.50
5
TYP 1.50
2X R 1.75
4X R 7.50
4X R 5
4X R 6.50
22 27.25
4.50
22
27

9.50
7
4
A   A
2.75
5.25
12
6
3
(10)
R11
2X R 4

2.91
6.21
2.50
30.00°
TYP 2.18
R3.99

SECTION B-B

SECTION C-C

∅ 3.50 THRU
3.50
R3.50
C   C
2
4
7
9

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A**   DWG. NO.   Finger Module Base   REV

SCALE: 1:1   WEIGHT: SHEET 1 OF 1

13

Ø 17.21

5.87

10

7   9.50

2.50

1.63

13

4

7.75

5

R1.05

TYP R 5

R5

1.70

8

10.50

18

9

16.70

5

6

20

4X Ø 4.75

4X Ø 2.50 THRU

26

3

13

2   22

TITLE:

SIZE   DWG. NO.   REV
A   Finger Module Tip Bottom

SCALE: 2:1   WEIGHT: SHEET 1 OF 1

5   4   3   2   1

1.50

1.50

2X M2x0.4 THRU

3

15

2.25

2.25

13.50

13.50

5

6.50

2X R 1.75

18

18.64

R1.05

1.70

8.50

4.50

9

26

A

2X M2x0.4 THRU

A

13

A

22

2

4

10.40°

3.41

16.20

10

12.60

SECTION A-A

15

6.25

23

26

9

1.50

6

6

20

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A**

DWG. NO.
Finger Module Tip Top

REV

SCALE: 1:1   WEIGHT: SHEET 1 OF 1

5   4   3   2   1

70

A

12

A

15

9

66

1

2X 6

2X 2

3

4

4

46

9

4

13

4

62

2X ⌀ 3.50 THRU

6.75

SECTION A-A

5   4   3   2   1

18

15

1.50

2X ⌀ 2.50 THRU

10.50

15

2.25

3

2.25

13.50

4.50

5

6.50

A

A

2X R 1.75

3

SECTION A-A

5

4

3

2

1

A

A

9.70

R7.50

6

4

3

∅2 THRU

2X X 2

SECTION A-A

2X 1.50

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE DWG. NO.   REV
A   Fingertip Actuator Silicone

SCALE: 2:1 WEIGHT: SHEET 1 OF 1

5   4   3   2   1

20

16

2

2

1

1.18

10

5

4

2X ⌀ 1.75 THRU

5

4

3

2

1

5.49 — 37.51 — 9

2

31

35

5

4X 1 X 45.00°

B | B

45

3 | 3

TYP 2

SECTION B-B

55

A | A

4

2.50

6X M2x0.4 THRU

6

TYP 5

14.88

TYP 5

40

15.24

50

2.50

SECTION A-A

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A** | DWG. NO. IMU Plate | REV

SCALE: 1:1 | WEIGHT: | SHEET 1 OF 1

5 | 4 | 3 | 2 | 1

2X Ø2.30 THRU
⌴ 4.2⍌6

73

17

2.50

15

25

3

6

2X Ø2.30 THRU

3

6

101

3

4

6

SECTION B-B

A

7

29.12

29.12

2X R 10

A

38

10

83

2

20

31.50

18

26

59

2X 10

3X 10

4.50

17

17

B

10.75

15.50

2

8

2.50

11.12

B

SECTION A-A

TYP 1.50

8

68.76

4X M2x0.4 THRU

TITLE:

SIZE
A

DWG. NO.
Palm Module Bottom

REV

SCALE: 3:4   WEIGHT:

SHEET 1 OF 1

5

4

3

2

1

Top view dimensions:
3.25 | 59 | 18 | 3.25
3.50
2X 3.50 X 6.50
3.50
15.50
10.75

95 | 6
2X M2x0.4 THRU | 2X R 2
A ... A
101
107
6
3
4

70
3 | 3
17 | 17
B
2
2.50
13.50
11.12
8
B
TYP 45.00°
2X 6
TYP 1.50
3.50
68.76
50.50 | 28.25

SECTION C-C
3X R 1.25
20
40
5.25

SECTION B-B
2.50
30
16

SCALE 1:2

SCALE 1:2

4x ⌀ 2.30 THRU
⌴ 4.5 ▽ 2
2x ⌀ 2.30 THRU
⌴ ⌀ 4.20 ▽ 2
20 | 37.50
15
25.50
8
83 | 6
101 | 3
C
C

SECTION A-A

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A** | DWG. NO. Palm Module Top | REV

SCALE: 3:4 | WEIGHT: | SHEET 1 OF 1

SECTION A-A

2X 1.50

2X 3

A — A

58

15

8

40

20

9

∅ 1.50 THRU

4

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE | DWG. NO. | REV

A Palm Touch Actuator Silicone

SCALE: 1:1 WEIGHT: SHEET 1 OF 1

5   4   3   2   1

2X ⌀ 1.75 THRU

25

2.50

5

30

4

4

1.50

20

5

11

10

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A** DWG. NO. Strap Guide REV

SCALE: 2:1 WEIGHT: SHEET 1 OF 1

5

4

3

2

1

30

5.58

10

3X ⌀ 3.50 THRU

2X R 2

12

22

12

6

4

6

4

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A** | DWG. NO. Thumb Connector Plate | REV

SCALE: 2:1 | WEIGHT: | SHEET 1 OF 1

5

4

3

2

1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Sleeve IMU Case Bottom		1
2	IMU		1
3	Sleeve IMU Case Top		1
4	92290A015	M2X8 SHCS	8

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A**

DWG. NO. Sleeve IMU Case

REV

SCALE: 2:1   WEIGHT: SHEET 1 OF 1

4X ∅ 1.50 THRU

4X R 3

4X R 2.88

40

6.27 — 14.81 — 12.60

20.99 10.50

10.18

9.75

10 — 15

5.50

30

4

A

8X M2X0.4 THRU

9.21

7.38

3

24

15.24

3

21.59

34

SECTION A-A

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A** DWG. NO. Sleeve IMU Case Bottom   REV

SCALE: 2:1 WEIGHT: SHEET 1 OF 1

5   4   3   2   1

40

B

4

12.50   15   TYP 4 X 45.00°

4   4X R 3

2

A

22

30

32   4

6

A

TYP 6

4X ⌀ 2.60 THRU   4X R 3

3

TYP 6

24

SECTION B-B

4

3   34

SECTION A-A

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		
COMMENTS:		

TITLE:

SIZE **A**   DWG. NO.   REV
Sleeve IMU Case Top

SCALE: 2:1   WEIGHT: SHEET 1 OF 1

5   4   3   2   1