Carnegie Mellon University

16-682

Project Course II

Final Report
COBORG

Team C:
Gerald D'Ascoli | Jonathan Lord-Fonda | Yuqing Qin | Husam Wadi
Feng Xiang

Sponsor:
Biorobotics Lab

December 8, 2021

Abstract

COBORG is a backpack-mounted robotic arm that helps people accomplish strenuous overhead manufacturing tasks. This project aims to stabilize a part through human-robot interaction, computer vision, and intelligent motion planning. It accomplishes this goal by receiving voice commands from the user and identifying the target location with a vision system that tracks the user's hands. The vision system also provides localization for the robot. The actuated manipulation system uses this localization to move the end-effector between the user's hands, regardless of the user's movement during the COBORG's operation. This report details all of the progress made on the COBORG project in terms of system architecture, technical development, and project management.

Table of Contents

1. Project Description

Overhead manufacturing in the automotive and aerospace industries can often cause strain to an operator's arms after working for long periods, and the operator usually requires a second person to assist them as they work [1] [2]. Specifically, the attachment of under-wing panels requires two individuals, one to hold the panel and the other to attach it. A robotic backpack system could hold the part for its wearer, allowing the operator to use their hands to secure the object. This Collaborative Cyborg Backpack Platform (COBORG) would allow a single individual to accomplish the entire task by themselves and alleviate the strain induced throughout the workday. To achieve this objective, the COBORG would have to be simple, accurate, and hands-free. The system described within this report accomplishes these objectives by taking all of its direction from the user through intuitive voice commands and automating the processes so that the user only has to say when a task has begun and finished. The result is that the user functions like they usually would, holding the panel and then screwing it in, while the COBORG automatically finds its target location (by identifying the hands) and stabilizes the part, all without further input from the user.

2. Use Case

2.1. Narrative

Jason works on the assembly of airplane wings at Boeing. When he arrives at work, he checks his to-do list and sees his list is topped with a series of tricky assemblies. Knowing that he will require aid, he walks over to the COBORG backpack arm station and signs out one of the units. Picking the unit up from its charging station, he straps the backpack on, adjusting the straps for comfort, and heads over to the plane he will be working on today. After completing some remedial tasks, he is ready to move on to the trickier cases where he will require the COBORG's help. First, he powers on the backpack arm, which has been in a compact/home position and has not been using energy up until this point. Next, Jason grabs the part he requires assistance with and holds it up over his head, fitting it into place. When the piece is stabilized, he says, "Hey COBORG, go here," and the COBORG backpack detects his hands in 3D space and moves the robot arm to a position where it can push onto the part and stabilize it (see Figure 1). Now that the COBORG backpack arm is holding the part, Jason lowers his arms and uses a drill with his free hands to screw the piece into place. While Jason's body shifts its position, the COBORG backpack arm adjusts to maintain the position of the end effector in 3D space, supporting the part regardless of Jason's position within the limits of the four degrees-of-freedom (DoF) arm. Now that the part is fastened, Jason says, "Hey COBORG, come back." The COBORG backpack arm returns to its compact/home position and goes into sleep mode, awaiting further instruction with minimal power usage. After completing all of his tricky tasks for the day, Jason returns the COBORG backpack arm to its charging station and signs it back in. While Jason completes the rest of his work for the day, the COBORG backpack arm charges, awaiting its next user.

2.2. Graphical Representation

Figure 1 - Graphical Representations of Use Case: Front Angle (Left), Rear Angle (Right)

3. System-level Requirements

Mandatory (*.M.#) and desired (*.D.#) performance (P.*.#) and non-functional (N.*.#) requirements are organized into their respective subsystems and shown in the subsections below. System-level requirements that pertain to more than one or all subsystems can be found in Table 1 below, vision subsystem requirements can be found in Table 2, voice subsystem requirements can be found in Table 3, and the actuated manipulation subsystem requirements can be found in Table 4 below.

3.1. System-level Requirements

Table 1 - System-level Requirements

ID	Requirement
N.M.1	Will be ergonomic for spinal comfort. Will be comfortable to wear for 30 consecutive minutes.
N.M.2	Will weigh less than 40 pounds.
N.M.3	Will be aesthetically pleasing.
N.M.4	Will operate safely.
N.M.5	Will be simple to operate.
N.M.6	Will be able to perform untethered for 20 minutes.
N.M.7	Will require minimal part modification to assist with assigned tasks.
N.M.8	Will be operable on a portable computer.
N.D.1	Will be able to operate standalone (no WiFi).
P.M.7.1	Will reach targets within 1.5 feet and 3 feet of the user's chest.
P.M.7.2	Will reach targets within 10 degrees (in the horizontal plane) on either side of perpendicular to the user's chest.
P.M.7.3	Will reach targets between horizontal and up to 20 degrees before vertical.

3.2. Vision Subsystem Requirements

Table 2 - Vision Subsystem Requirements

ID	Requirement
P.M.1.1	(Detect indicated parts) Will have 60% accuracy of detecting indicated location within 6" in 3D space, and always within 12".
P.M.1.2	(Calculate object) Will detect the intended object within 5 seconds of when the move command is issued.
P.M.1.3	(Pose Detection) Will detect the surface normal of the part with an error no greater than 45°.
P.M.8.2	Will detect the user's hands, even if one or more hands lie outside of the task space by 17 degrees in the horizontal plane and/or 10 degrees in the vertical plane.
P.M.8.3	Will detect the user's hands if they are in an approved location with the back of the hand's facing the user (no under-handed grips).
P.M.9.1	Will detect obstacles within 0.6m of depth camera mount.
P.M.9.2	Will generate voxels on obstacle surfaces with <1.25" between the voxels.
P.D.1.2	(Texture Invariant) Will be invariant to part texture, specifically matte finish and gloss finish.

3.3. Voice Subsystem Requirements

Table 3 - Voice Subsystem Requirements

ID	Requirement
P.M.4.1	(Voice command) Will be able to understand the voice command 60% of the time.
P.M.4.2	(Voice command) Will be able to understand at least 2 unique voice commands, up to 8.
P.M.4.3	(Voice command) Will be able to understand commands of at least 2 words in length, up to 8.
P.D.2.1	Speaker will alert the user to state changes with an 80% success rate.
N.M.9	Audio feedback will be clearly audible in a representative work environment.

3.4. Actuated Manipulation Subsystem Requirements

Table 4 - Actuated Manipulation Subsystem Requirements

ID	Requirement
P.M.2	(Move to object) Will reach within 6 in of the planned target position 60% of the time, and always within 12 in.
P.M.3.1	(Hold object) Will maintain the target's spatial position with less than 6 in of error margin.
P.M.3.2	(Hold object) Will be able to hold a representative part overhead.
P.M.5	(Release object) Will release object within 5 seconds of when the release command is issued.
P.M.6	(Compact arms) Will bring the full robot arm to within 20" of the point of attachment to the backpack.
P.M.8.1	Will stabilize the part, even if the user moves 6" in any cartesian direction and/or rotates their body up to 90 degrees to the left or 20 degrees to the right.
P.M.10.1	Will not collide with the stationary user's arm during actuation 90% of the time.

4. Functional Architecture

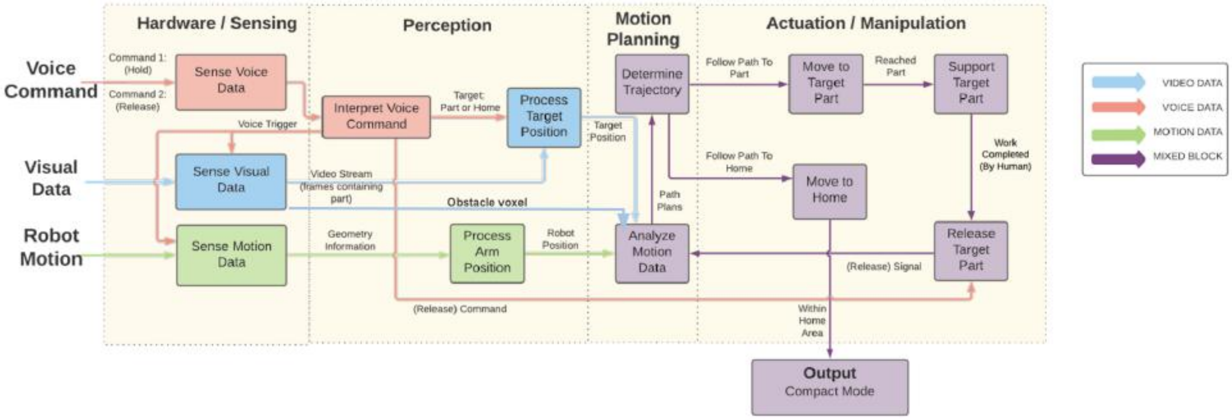


Figure 2 - Functional Architecture of COBORG System

The Functional Architecture diagram shown in Figure 2 outlines the significant functions of our system and data flow between subsystems. It contains three aspects: the input data, the output, and the four major subsystems. Input data comes from the hardware sensors on the COBORG system. Our system will interpret the data and generate a manipulation output through the robot arm. The details for each subsystem are introduced in the following paragraphs.

As listed on the left side in Figure 2, the System Inputs are the data streams coming from the hardware (sensors) on the COBORG. Specifically, our system uses three input data types: voice command, visual data, and robot motion. The microphone captures voice commands from the user. Visual input localizes the target object from the depth camera mounted on the COBORG frame. Finally, motion data input allows us to analyze the robot arm motion and plan trajectory paths.

The Hardware and Sensing Subsystem is responsible for capturing the inputs. This subsystem will keep track of the sensor data during system operation. Depending on the voice command (i.e. “Go here”, “Come back”), it will trigger the following subsystem processes. Video stream and point cloud information will be fed into the Perception Subsystem and motion data will be used in the Motion Planning sub-function.

The Perception Subsystem receives sensing data from the Sensing Subsystem, specifically visual data. After the system interprets the voice command from the Sensing Subsystem based on the voice command content, the system will detect the desired target (part) position using visual data and retrieve robot arm motion data to execute the Motion Planning sub-function. Moreover, obstacle detection is triggered during the robot arm execution.

The Motion Planning sub-function will use motion data, obstacle data, and target position information to determine a trajectory path for the robot arm. Data will come into the “Analyze Motion Data” block to generate possible path plans. The determined trajectory from the multiple path plans will be forwarded to the Actuation and Manipulation Subsystem.

The Actuation and Manipulation Subsystem receives a trajectory plan as the input. By controlling the robot arm, COBORG will follow the trajectory, move to the desired position, and stabilize the object overhead using a resolved rate controller. Once the voice command trigger is received again (i.e. "Come Back"), the robot arm shall release control of the object and move back to the compact position, which is the system's final output.

5. System-level Trade Studies

5.1. Vision Hardware Upgrade

We performed a trade study to determine the ideal hardware upgrade plan for the vision system to meet our task space requirement. Table 5 below summarizes all the options we have considered so far. As defined in our system requirements (P.M.7.2 and P.M.7.3), we expected the upgraded system to extend the current camera's FOV for the system to operate in front and overhead use cases without changing any camera settings. The options we considered using are: two cameras, one camera with a motor, and one camera attached to a hat worn by the user. The fourth and fifth options could give more vertical FOV, but additional installation and integration effort are required to make it work properly. The "two cameras stacked" option was chosen because of the higher priority in extending the vertical FOV over horizontal FOV. It was also easier to integrate with the current hardware and software system than the single camera options.

Table 5 - Vision Hardware Trade Study

Attributes	**1 Camera**	**2 Cameras Stacked**	**2 Cameras Each Side**	**1 Camera with Dynamixel**	**1 Camera with a Hat**
Horizontal FOV	66	66	120	66	120
Vertical FOV	42	75	42	90+	120
Cost	No	1 more camera	1 more camera	1 more dynamixel	Hat cost
Estimated Time to Set Up	Low	Medium	Medium	Medium	Large
Software Modification	Low	Medium	Medium	Large	Low
Run Time (YOLOv3)	10FPS	5FPS	5FPS	10FPS	10FPS

5.2. Actuated Manipulation

We performed a trade study to determine the ideal robot arm design for the COBORG to meet our task and stabilization space requirements. Table 6 below summarizes all the designs we considered and the scoring criteria for each design. As we defined in our system requirements (P.M. 7.1, PM 7.3, and PM 8.1), we expected the upgraded robot arm to actuate and stabilize within our designated task space to sufficiently meet our overall use case requirements. Overall, the options we considered were power, weight, and accuracies determined from our software simulations (see Section 7.3.2). The 4 DoF robot arm design was chosen for the significant increase in simulation accuracy compared to the baseline 3 DoF arm. Also, any increased DoF configuration would significantly increase the weight and power specifications for minimal performance gain.

Table 6 - Actuated Manipulation Degrees of Freedom Trade Study

Attributes	3 DoF	4 DoF	5 DoF
Number of Motors	3	4	5
Max Power Draw [W]	124.8	232.8	274.4
Total Motor Weight [lbs]	2.34	3.43	4.22
Task Space Simulation Accuracy [%]	59	86	87
Stabilization Space Simulation Accuracy [%]	40	75	77

6. Cyberphysical Architecture

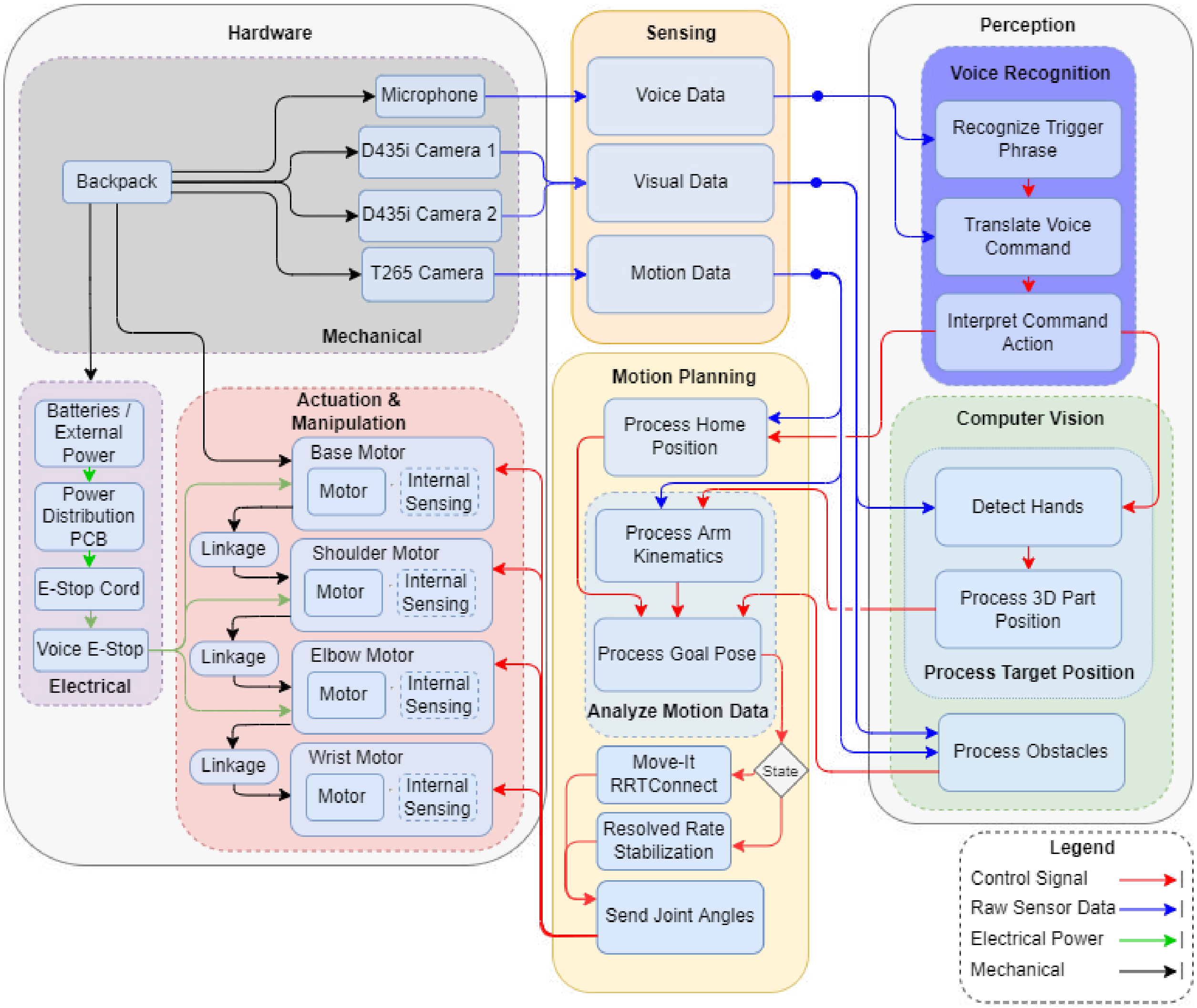


Figure 3 - Cyberphysical Architecture

The Cyberphysical Architecture shown in Figure 3 illustrates the interaction between the hardware actuation and the perception layer of the COBORG system. This elaborates on the general functionality of the COBORG described in the Functional Architecture in Figure 2 and details how this functionality is processed between the subsystems and physical components.

The perception layer reads information about the system state and other environmental data through the sensing block in the form of voice, vision, and motion data read from the microphone, D435i depth camera, and the T265 tracking camera, respectively. The voice data is used in the voice recognition system to identify the key phrase "Hey COBORG" and interprets the words following the key phrase to identify command actions. For example, suppose the recognized command is to move to the target. In that case, the vision system activates and detects the hands, processes the 3D position of the part based on the hand positions, then feeds that target position to the motion planning system. On the other hand, suppose the recognized command moves from the target to the home position. In that case, the motion planning system directly activates and compacts the arm. The motion planning system is also fed obstacle information from the vision system for obstacle avoidance processing.

The hardware components are controlled by the perception layer through the motion planning system. The COBORG arm has four HEBI motors in the form of a base, a shoulder, an elbow, and a wrist joint stemming from outside the user's right arm. This arm, along with the rest of the hardware frame and components, is represented in the COBORG's Unified Robot Description Format (URDF). These motors are actuated using a URDF informed trajectory planned by the motion planning system. This trajectory is generated by planning from the arm's current position to the desired goal position, whether it be the home position or some other target pose. This planned path will also avoid obstacles detected by the vision system.

The remainder of the hardware section describes the linkages between the motors, the physical connectivity of the sensors used, and the electrical system. The electrical system is described later as the COBORG custom power distribution PCB.

7. System Description & Evaluation

7.1. Overall System Depiction

COBORG is a collaborative robot arm that can help people hold objects within its task space. The inputs to the system come from the user's voice, which triggers the system to start moving to the target then stabilizes it in place. Our system consists of several subsystems to achieve this goal: the hardware framework, electrical framework, sensing, perception, motion planning, and actuated manipulation. The current status of the overall system is depicted in Figure 4 below. In the Fall semester, we performed a hardware redesign and complete system integration for full use case operation. By the end of the FVD encore, we demonstrated in front and overhead use case demonstrations.

Figure 4 - Overall Depiction of COBORG

7.2. Subsystem Descriptions/Depictions

7.2.1. Hardware Framework

The Hardware Framework of the COBORG is broken down into three subcomponents: the frame, the case, and the manipulator arm.

The frame consists of the structural foundation of the robot that is mounted to the user. This frame includes the backpack that the user is wearing and the structural metal frame attached to the backpack straps. In addition, the manipulator arm and the case are structurally secured to the extruded aluminum. See Figure 5 (top left) for pictures of the as-built frame.

The manipulator arm is attached to the frame and serves as the mobile unit of the robot. This arm includes the aluminum linkages, motors, and end effector. See Figure 5 (top right) for pictures of the as-built manipulator arm.

The case is the container box that houses all head-end equipment to the robot. A majority of the electrical framework components are housed in the case. In addition, access holes are installed on the case's exterior to allow wiring to travel into and out of the housing safely. See Figure 5 (bottom middle) for pictures of the as-built case.

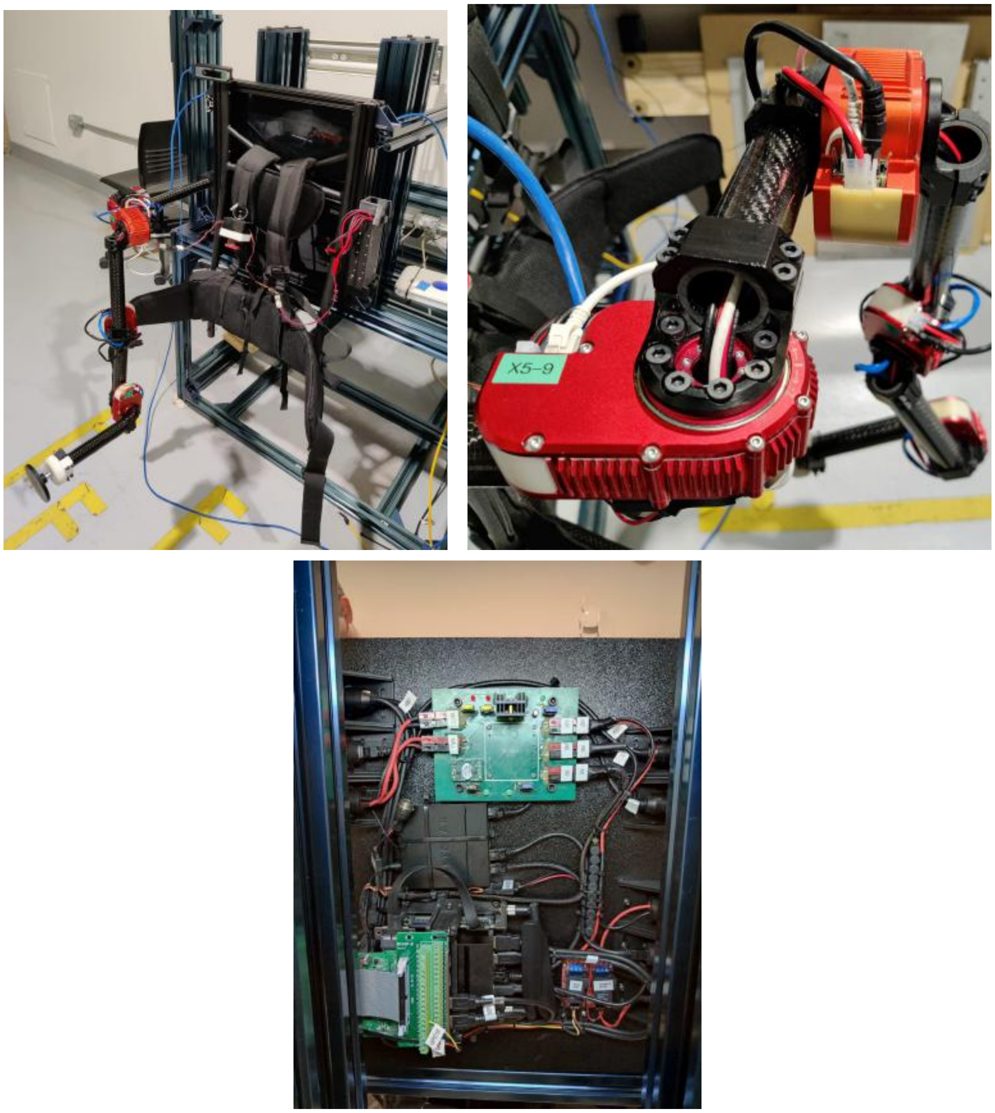


Figure 5 - As-Built COBORG Framework: Front Angle View (Top Left), Arm View (Top Right), Rear View (Bottom Middle)

7.2.2. Electrical Framework

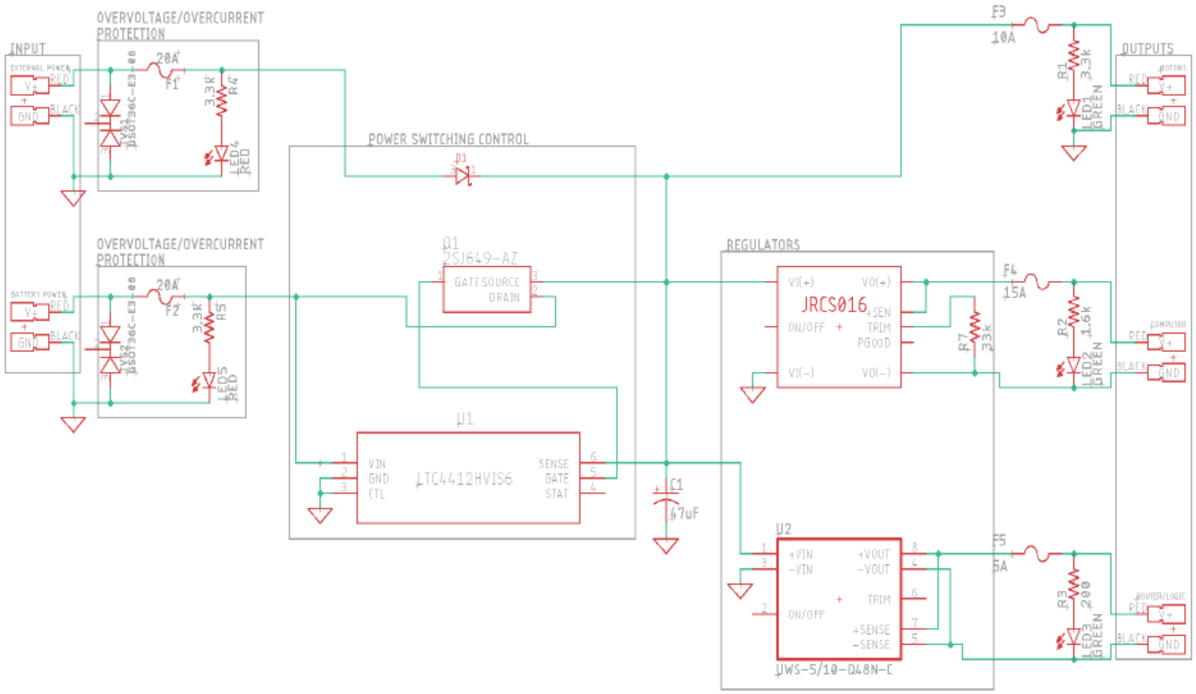


Figure 6 - COBORG Power Distribution PCB Schematic

The COBORG system can be powered by either an external 36VDC power source (i.e. wall outlet) or by the onboard 36VDC battery packs. Power is then distributed to the motors, computer, and router/5V logic via the COBORG Power Distribution Printed Circuit Board (PD PCB), shown in Figure 7 and schematic shown in Figure 6. This PD PCB prioritizes power from the external power source over the battery. The internal batteries are always plugged into the system, but when the external power is connected, the logic on the PCB switches the power sourcing to draw the full 3-10A (~100-400W) from the external supply and only draw ~10mA (~0.4W) of current from the battery. This circuit allows the battery to maintain charge and keeps the batteries in a state of minimal discharge to allow for safer charging; charging which will also source from the external power. This 36VDC input directly powers the COBORG arm motors at the recommended 36VDC. The PCB has two switching voltage regulators to convert the input 36VDC to 19VDC and 5VDC for the computer and router/5V logic respectively. The 5VDC power is achieved by feeding the 36VDC into an “18VDC-75DC to 5VDC” converter (UWS-5/10-Q48P-C). The 19VDC power is achieved by feeding the 36VDC into a variable power supply that accepts 18VDC-85VDC and outputs a voltage related to the trim resistor connected to the PCB. Based on the equation given on the datasheet, Rtrim=700-10*VoutVout-4, with a desired Vout≤ 19VDC, a 37kΩ resistor was selected. Due to tolerances on both the resistor and the regulator, the final output voltage to the computer is ~18.7VDC.

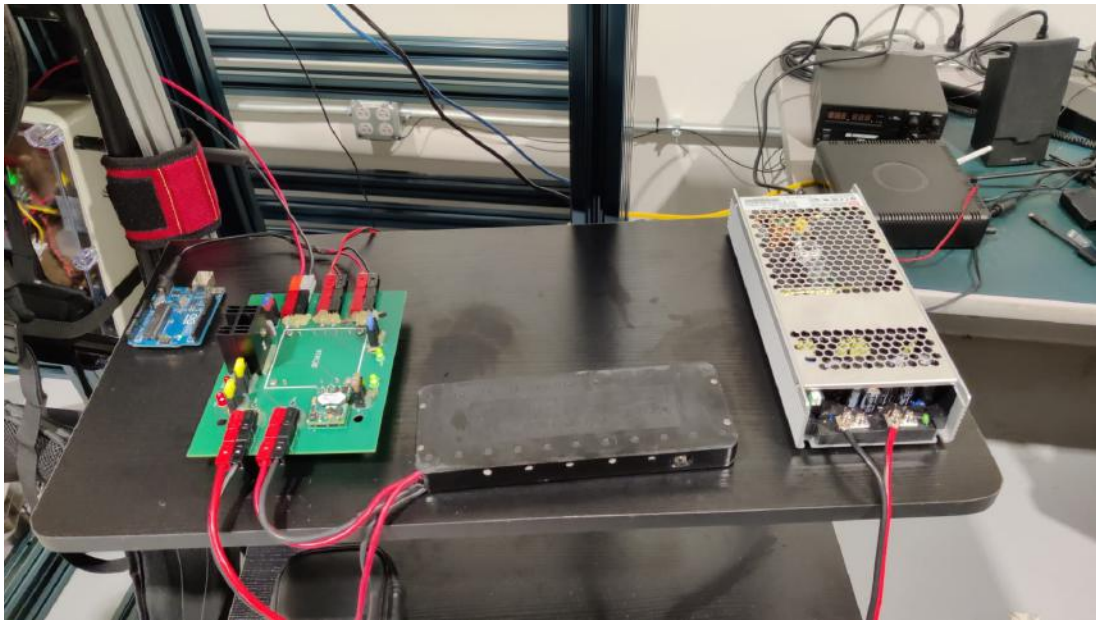


Figure 7 - COBORG Power Distribution PCB w/ Both Power Sources Plugged In

7.2.3. Sensors

The following sensors are used on the COBORG:

- Intel Realsense D435i depth camera (2)
- Intel Realsense T265 tracking camera (1)
- IMU sensor (internal to each HEBI motor) (4)
- Microphone (1)

The Realsense cameras and microphone are currently mounted onto the frame and close to the user's shoulder(s). The two D435i depth cameras are mounted on the user's right shoulder with the configuration shown in Figure 9, which has an extended view of the user's hands and the robot arm. The T265 tracking camera is mounted behind the depth cameras, facing the opposite direction. By defining the translation between these three camera frames, the robot arm follows the hand movement shown in the depth camera. Figure 8 demonstrates the position of these sensors.

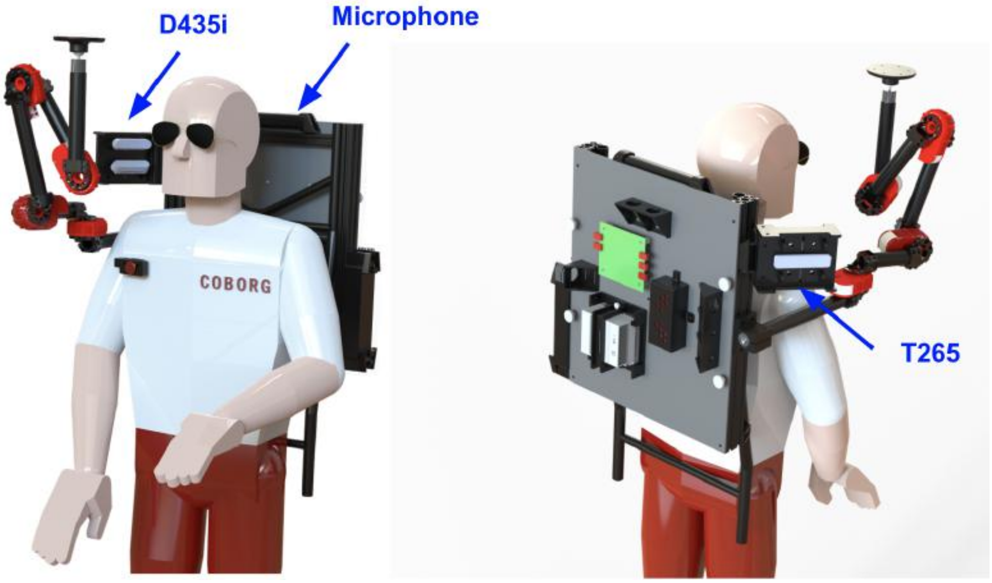


Figure 8 - Sensor Positions

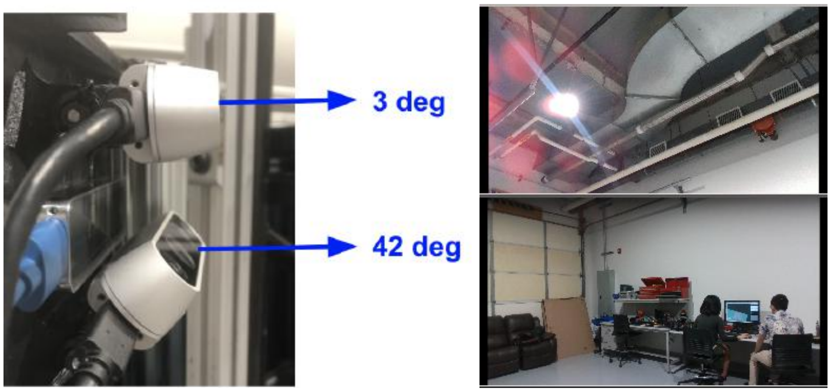


Figure 9 - New Camera Configuration and Extended View

7.2.4. Perception: Vision

Localization:

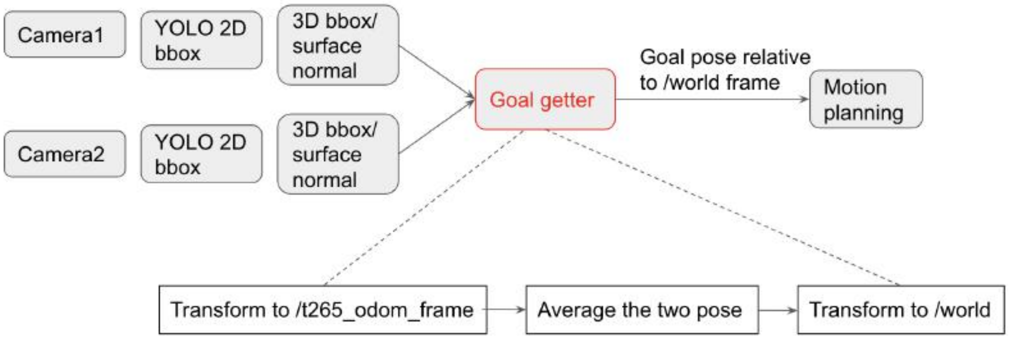


Figure 10 - Vision (Localization) ROS Node Map

The Localization subsystem aims to determine the 3D target part position and surface normal by utilizing a hand detection algorithm combined with point cloud information. Intuitively, the hand position could be inferred as the goal position of the target since users always want to put their hands on the target to hold it overhead. By localizing the hand position on the part, the goal position can be easily retrieved. In the spring semester, we implemented one camera pipeline with 2D hand detection, 3D bounding box extraction, and post-processing: including goal position calculation and surface normal estimation. We added a second camera to deal with the overhead use case to extend the detection range (i.e., Vertical FOV), defined by our system requirements. The main pipeline is shown above in Figure 10. All of the processes discussed are implemented within ROS to ensure solid communication between different subsystems. By using ROS publishers and subscribers, the data is easily transferred between different processes.

The Localization subsystem starts from the 2D hand detection algorithm to localize the hand in the camera frame. The hand detection algorithm being used is the open-source algorithm YOLO v3 (You Only Look Once) [3]. The same pretrained hand detection model is used for both pipelines. An example of hand detection results combined with ROS is shown in Figure 11. Once the 2D bounding boxes are retrieved, the system will start looking for the point cloud information from the depth camera (D435i) to extract the 3D bounding boxes. The size of the 3D bounding boxes remains the same as the 2D bounding boxes, except that extra depth information is provided. The example yellow bounding box generated by this process is also shown in Figure 11. The system first takes the average of the 3D bounding boxes' center points to obtain the goal position. It then searches the nearby points around the goal to estimate the surface normal orientation information for the target.

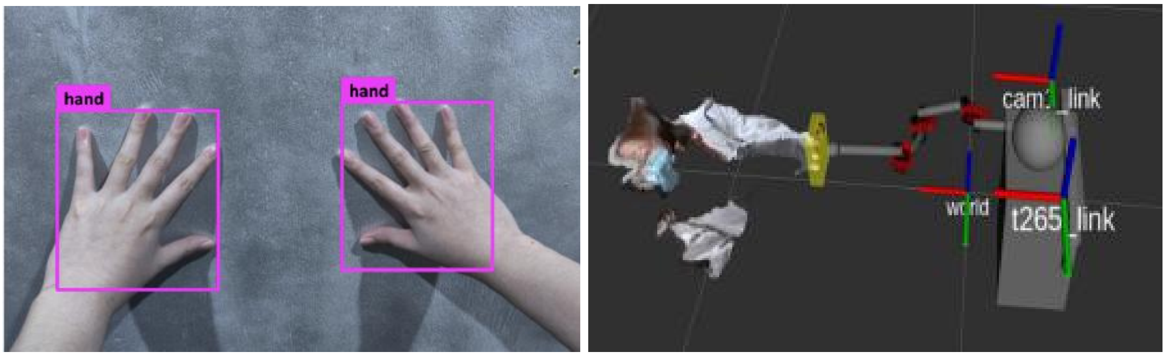


Figure 11 - Sample Outputs for 2D and 3D Hand Detection Results

The upgraded vision system utilizes two depth cameras stacked together to achieve a larger vertical field of view. The two hand detection pipelines also run in parallel with both depth cameras. Each channel will output a goal position and surface normal based on how many hands are shown in the frame. To post-process the outputs from the two pipelines to get a single goal pose, the system first converts the goal positions from the two channels to the same T265 odometry frame. Then, it takes the moving average by looking at the five adjacent positions and surface normals to generate the final goal pose. With the goal pose generated, the system transforms it to the world frame, which can be easily used later in the motion planning system.

Obstacle Detection:

The obstacle detection functionality is powered by Octomap built into Move-It. The purpose of this obstacle detection was to actively identify the user's arm(s) to avoid the user during actuation. The Octomap takes in the point cloud from the two D435i cameras and samples the points to determine obstacles in the space, given specific tunable parameters. These parameters impact the range of obstacle sensing, subsampling of voxels, and padding around the URDF for robot self-recognition. The range was set to 0.6m to detect the user's arm(s) without impeding the goal's actuation and minimizing computational impact. The computational impact was also optimized by increasing the subsampling to 100. This meant that the Octomap samples fewer points on the point cloud (1 in 100) to generate collision voxels. The padding was set to allow slight errors between the URDF and the robot as seen in the vision space so that the Octomap wouldn't mistakenly detect the robot itself as an obstacle. Figure 12 below shows the collision voxels generated in the camera space over the user's arm about the robot's URDF. The image on the left shows the robot arm in this visual space without any collision voxels generated due to the URDF padding parameters in Octomap.

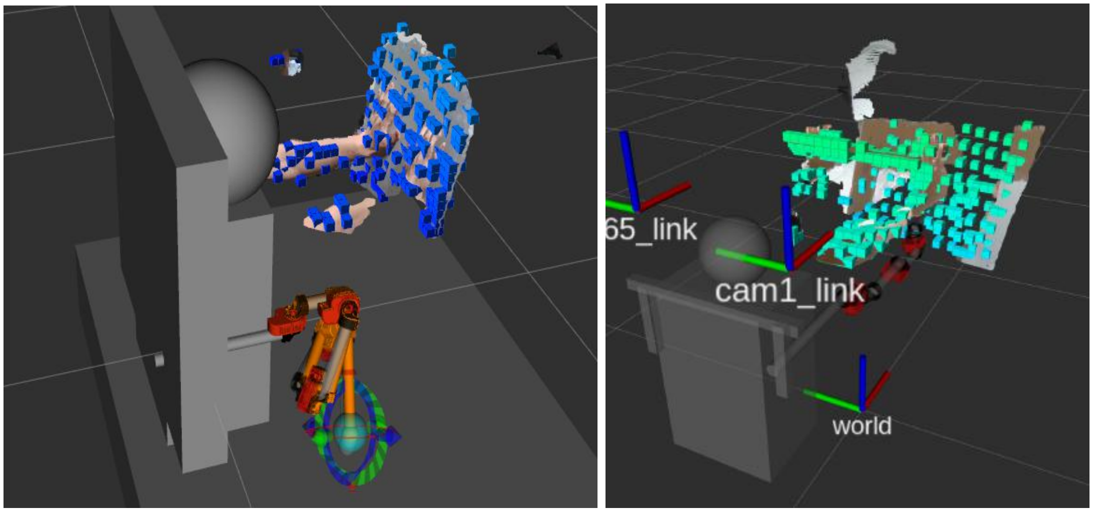


Figure 12 - Collision Voxels Generated on User's Arm

7.2.5. Perception: Voice

The voice recognition subsystem serves as the interface between the user and the COBORG for the user's desired functionality. The voice recognition system uses the open-source PocketSphinx platform for optimal recognition, allowing fully offline/untethered functionality. For implementation on the COBORG system, the dictionary was optimized to recognize commands and the word "COBORG" because COBORG is not in the standard English dictionary. The given model was trained on the new dictionary for roughly a day to optimize the model for improved recognition. The input for the voice recognition system is the audio stream from the microphone mounted on the COBORG backpack strap. The audio output is the speaker mounted on the COBORG. The input audio is processed using pyaudio, and the output is served to the speaker as mp3 files using pydub built on pyaudio. The voice recognition system is built on the voice_recog ROS node. Once a valid command from the user is recognized, the node publishes the command code to the /voice_commands ROS topic. The main_state_machine then subscribes to this topic to process the voice command and dictate to the other subsystems the commanded function.

The user triggers the command recognition by saying the key phrase, “Hey COBORG.” Once “Hey COBORG” is recognized, the COBORG plays a short tone to alert the user that it’s now listening for a command. From here, the user can command the robot to move to the target part established by the perception subsystem by saying “Go Here.” Once the user has secured the panel, the command “Come Back” will bring the COBORG off the target part and back to the compact home position. After the key phrase is said and either of these commands is recognized, the COBORG plays a short, upbeat success tone to confirm that the command was successfully interpreted.

Additionally, the user can trigger the command prompt with “COBORG” and say the command “Stop” to activate the soft emergency stop function. Alternatively, the user can say “Stop Stop Stop” without preceding with the “COBORG” trigger for a more easily accessible and natural command to emergency stop the COBORG’s current function. Once either of these e-stop commands is recognized, the COBORG plays a jarring alert sound. If the “Hey COBORG” keyphrase is recognized but the voice system does not recognize any valid commands in the following phrase, it will play a negative failure tone to alert the user that no command was detected and therefore the COBORG will not act.

7.2.6. Actuated Manipulation

The actuated manipulation subsystem stands at the end of the integration pipeline. It provides the motion planning, motor actuation, and force output needed to aid the user in their intended task. Action is initiated from the main state machine. Goal positioning is derived from the output of the vision subsystem. Given the high-level command interpreted by the voice subsystem/main state machine as well as the end goal pose from the vision subsystem, the actuated manipulation subsystem will move its various joints to direct the end effector. The end effector will move from a standby state to the end goal state within a reasonable amount of time and through a viable pathway that is minimally intrusive to the user and its environment.

The manipulator arm is a serial linkage system with four revolute joints and a flat pad installed on a spherical joint at the end of the arm, serving as the passive end effector. Figure 5 below displays the robot arm’s build at a powered-off state. Each revolute joint is installed at specific angles between each other, relative to the principal axes of the base link. The entirety of the arm is situated on the right side of the user, between their shoulder and elbow, depending on the height and build of the user. The linkages connected to each revolute joint are thin-walled carbon fiber tubes. The revolute joints are HEBI X-Series actuator motors, and the mounting brackets between the joints and linkages were also made by HEBI. Data and power are serially connected between motors where the linkages serve as the conduit between motors. From the proximal to the distal motor, the names of the motors are as follows: base motor, shoulder motor, elbow motor, and wrist motor.

At the standby state of the actuated manipulation subsystem, the manipulator arm is in its compact/home position, as seen in the top-left photo of Figure 13. The high-level action that the system performs is to push the end effector onto a part that is situated within the designated task space, as highlighted in requirements P.M.7.1, P.M.7.2, and P.M.7.3 in Table 1. The goal state is located on the surface of the part, between the user's hands. This goal is constantly being updated and supplied by the Localization subsystem, which publishes the goal to the /goal ROS topic. The extracted goal state is about the /world frame of the robot URDF, a local frame from which the actuated manipulation subsystem plans its trajectories in 3D space.

After picking up the goal state from the vision subsystem, the robot arm will first actuate from the home position to an intermediate ready position. This intermediate position was implemented to prevent the manipulator arm from reaching an undesirable joint configuration while actuating to the goal state. The next position is at an offset distance away from the goal state, along the surface's normal vector. This position allows the manipulator to switch from avoiding obstacles to pushing into the panel. Should the actuated manipulation subsystem be unable to solve for the intermediate position or goal offset position within a set amount of time, the robot arm will alert the user with a beep, return to the home position, and await the next "Go Here" command from the user. Otherwise, once the end effector reaches the goal position offset point, the actuated manipulation subsystem will go into the stabilization state.

In the stabilization state, the actuated manipulation subsystem now maintains its global position, thereby stabilizing the intended part. The robot arm is controlled using resolved rate. This closed-loop cartesian controller translates the error between the robot arm's end effector position and the goal position into joint angle changes for the robot arm to compensate. Stabilization begins by taking the current offset position as the goal and iterating that goal position into the surface by a set distance. This causes the robot arm to incrementally actuate forward until it is in contact with the board and applying significant force in the direction normal to the panel. The global goal position is constantly being updated and published from the vision subsystem into the /goal ROS topic, so the actuated manipulation subsystem constantly updates the goal position over time and takes incremental steps to close the distance between its current and target positions.

Once the user wants the robot arm to return to the home position, they give the "Come back." voice command. This causes the robot arm to iterate back to the original offset position. Then the arm switches from resolved rate control to planning a joint trajectory using RRTConnect and actuates to the intermediate ready position. After reaching the intermediate ready position, the robot arm actuates to its compact home position and waits for the following command to be issued. Figure 13 shows the robot arm at its various stages during an entire use case demonstration of installing an overhead panel. Figure 14 shows the robot arm going through a de-installation process of taking a board out of the wall in front of the user.

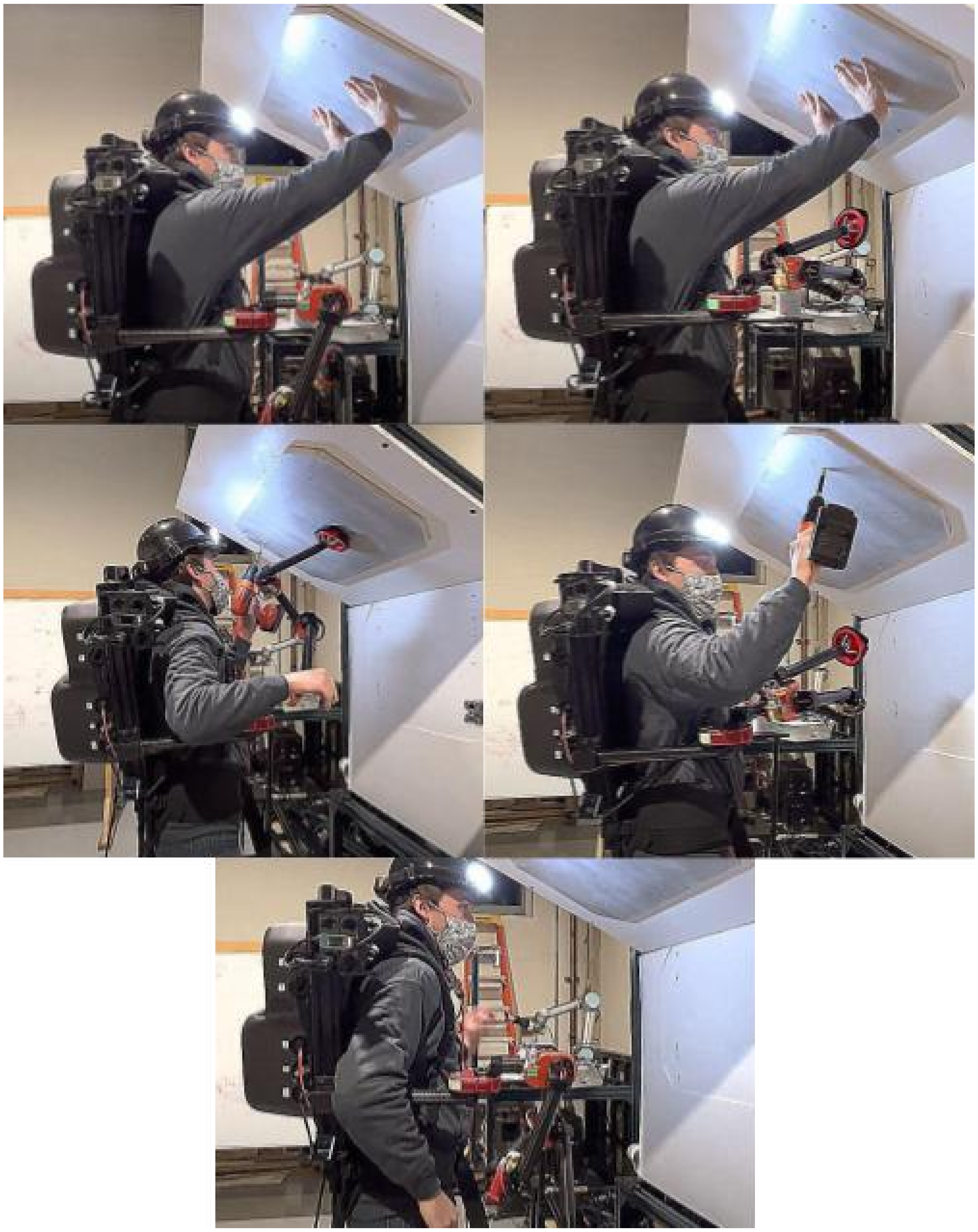


Figure 13 - Full Use Case for Overhead Installation: Compact/Home position (Top Left), Intermediate Ready Pose (Top Right), Push and Stabilization (Middle Left), Naive Pull (Middle Right), Compact/Home Position Again (Bottom Middle)

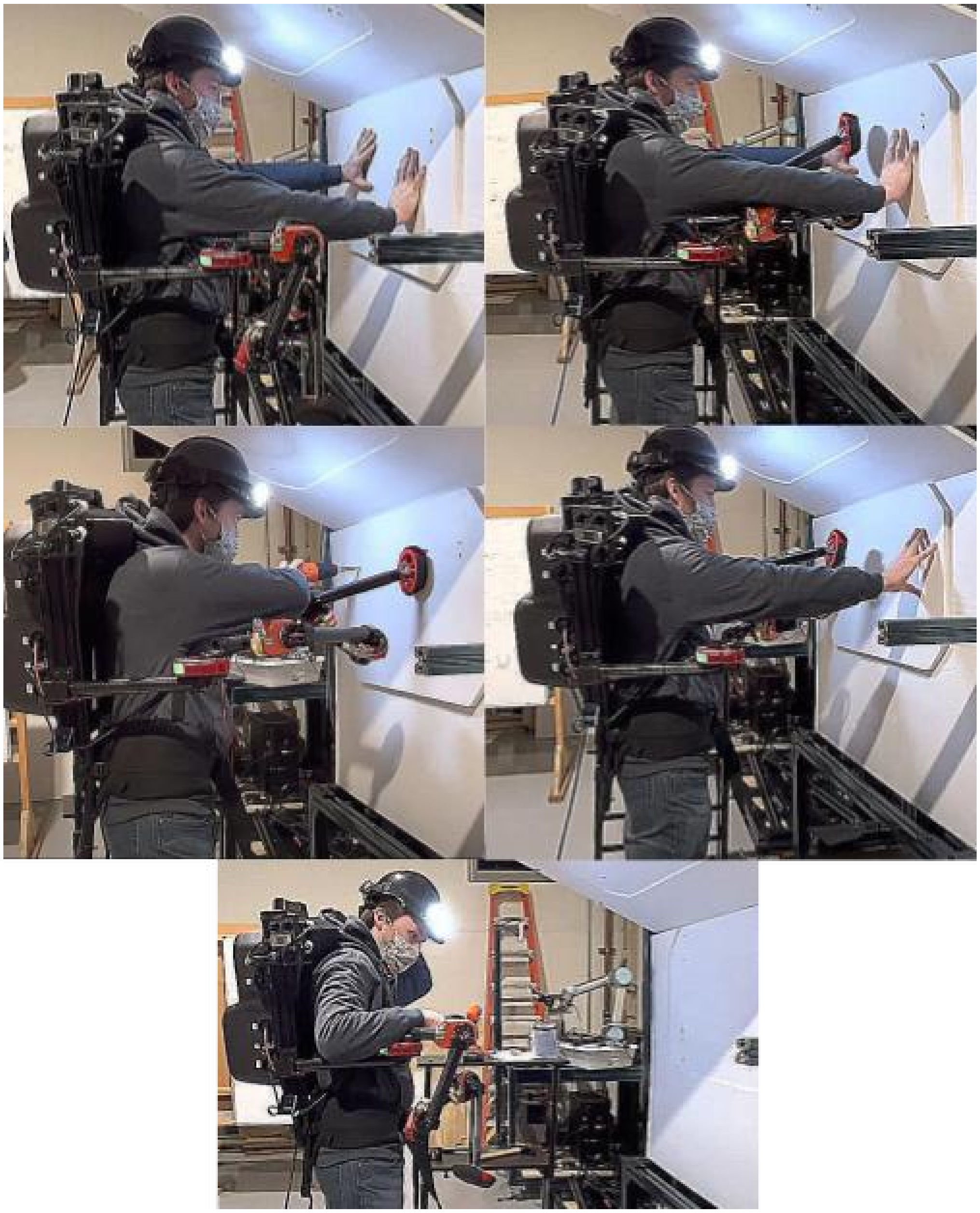


Figure 14 - Full Use Case for In-Front De-installation: Compact/Home Position (Top Left), Offset Along Normal Vector Pose (Top Right), Naive Push and Stabilization (Middle Left), Naive Pull (Middle Right), Compact/Home Position Again (Bottom Middle)

7.3. Modeling, Analysis, and Testing

7.3.1. Vision Subsystem Validation

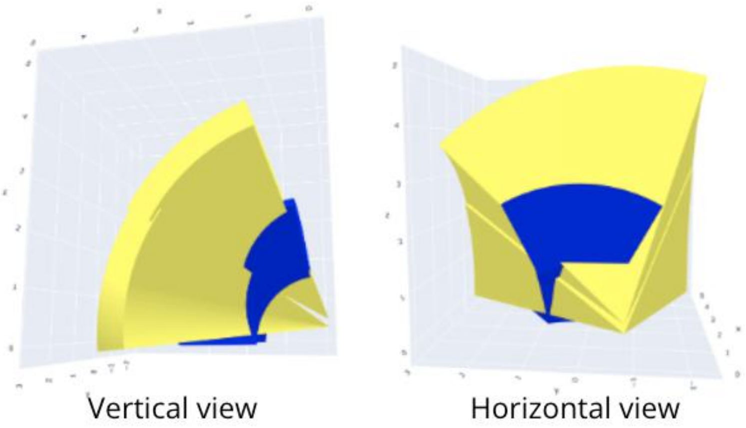


Figure 15 - Vision System FOV Validation (Blue: Task Space, Yellow: Camera Space): Vertical View (Left), Horizontal View (Right)

The upgraded vision system with two cameras was validated using a simulation tool. By putting two cameras together, the vision system's vertical field of view was extended from 42 degrees to about 75 degrees. This gave the capability to achieve both the front and overhead use cases. The above Figure 15 demonstrates the simulation results: the yellow volume represents the space that two depth cameras could capture, and the blue volume represents the task space defined in our requirements. To summarize the results, the upgraded vision system could capture 75 degrees of vertical FOV, and 66 degrees of horizontal FOV. This meets the system requirement for both use cases (70 degrees horizontal, with 10 degrees on either side of the user's chest).

The complete system integration has been tested for 26 trials with the same person holding different boards at different locations. The result is summarized in Table 7. Within all the 26 trials, the use case failure due to the vision system's false detection happened only once. Therefore, the total success rate for the vision system is 96.15%, which is above the 60% accuracy defined in our subsystem requirements.

Table 7 - Vision Subsystem Success Statistics

Total Trials	26
Vision Success	25
Success Rate	96.15%

7.3.2. Actuated Manipulation Subsystem Validation

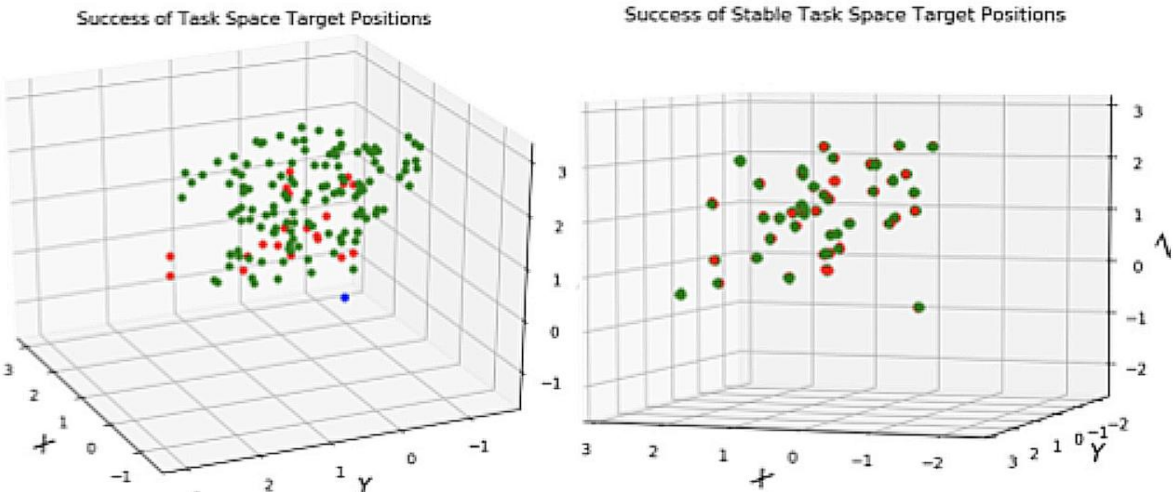


Figure 16 - Position Task Space (Left) and Stabilization Task Space (Right) Success (Green: Success; Red: Failure)

The upgraded robot arm hardware was validated using a simulation tool against multiple robot arm design iterations. By expanding the degrees of freedom of the robot arm, the actuation task space can expand. In addition, allowing additional degrees of freedom will enable the robot to stabilize against more user configurations. The tested task space in the simulation consists of the volume 1.5 to 3.0 ft away from the user's chest and spanning an angle relative to the horizontal plane between -10 and 90 degrees. A set amount of points were sub-sampled from the volume and shown in Figure 16 (left). From those sub-sampled points, more points were sub-sampled for the stabilization space. Two additional points are created for each sample in the stabilization space, one about a 90-degree angle rotation to the left about the user's chest and one 45-degree rotation to the right. For each of the three points, additional points were created for one inch in all cartesian directions (-x, +x, -y, +y, -z, +z) about the point. Success is determined if the MoveIt planning and execution pipeline can solve and actuate the robot arm in simulation to the given points. For the task space simulation, the starting position of the robot arm is at the compact/home position. For the stabilization space simulation, the starting position of the robot arm is at the original sub-sampled point.

Two overall designs were considered and compared to the baseline 3 DoF robot arm: one 4 DoF and one 5 DoF robot arm. Results from the mechanical task space simulation are shown in Table 6. Analyzing the results and specifications of the different designs, the highest performing design was the 5 DoF robot arm. However, it was marginally better than the 4 DoF arm design. Thus, increasing the power and weight to gain a couple of percentages in performance for the 5 DoF arm over the 4 DoF variant did not seem like an optimal decision. In the end, we determined that the 4 DoF arm provided the most performance increase over the 3 DoF with an optimal increase in power and weight.

7.4. Performance Evaluation Against Fall Validation Demonstration (FVD)

For the Fall Validation Demonstration, the COBORG system was required to hold and stabilize a representative part while an operator used a drill and screws to attach the part to a forward-facing and overhead wall. The user also detached the board from these walls. Additionally, the user was allowed no more than five minutes for each task. During FVD, the system successfully attached and detached the panel in both the forward-facing and overhead scenarios; these tasks took the user a little over a minute to complete, well within the allotted five minutes. A human demonstration was also given for comparison, which took only slightly less time than the COBORG system. For FVD Encore, the weight of the panels was increased from 3.5 lbs to 7.5 lbs and even 15 lbs. The 7.5 lb panel was successfully attached and detached in the forward-facing and overhead scenarios. The 15 lb panel was only tested in the forward-facing configuration for safety considerations. It was successfully attached to the wall, but some slipping necessitated the user re-aligning the arm during detachment. More statistics on the system's success rate can be found in Table 8 below.

Table 8 - FVD Success Statistics

Total Trials	26 (14 Front, 12 Overhead)
Use Case Success	22
Success Rate	84.62%
Average Time to Finish	74s

Another requirement for the system was that it would support typical usage for at least 20 minutes. At the end of both FVD and FVD Encore, the total battery life of the system was extrapolated and indicated an entire lifespan of more than 2 hours, well over the required 20 minutes.

The comfort of the system was tested in an informal manner. The system was expected to be worn for 30 consecutive minutes without discomfort. After both FVD and FVD Encore, the user, who had been wearing the backpack for even longer, was asked to report on the device's comfort. The user remarked that the device was surprisingly snug and comfortable, with no issues standing out to them. The user had also been wearing the COBORG backpack for multiple hours consecutively leading up to the demos.

The voice subsystem proved successful by our requirements by the end of spring, so the only changes before FVD were adding final commands and ROS publishers and subscribers for improved interfacing. Because the functionality was the same as the spring, no quantitative tests were used to validate the voice subsystem for FVD or FVD Encore. Instead, the voice subsystem was implicitly tested and validated through successful demonstration of the total integrated use case of the COBORG, with the voice subsystem serving as the control interface for the user to the robot. We tested extreme cases with the voice recognition subsystem during these demonstrations by proving successful functionality in environments with lots of background voices and with loud manufacturing plant background noise during FVD Encore. We faced some challenges during the FVD Encore when our original system microphone became unresponsive as the audience entered the room. Luckily, we had planned for this in risk analysis and had a backup microphone to hot-swap. However, the new microphone was not performing with the same sensitivity and directionality, causing some recognition issues during the demonstration. We also proved voice recognition capabilities during the poster presentation with our original microphone despite the background noise of many other people speaking and asking questions nearby. Through these passive trials, we implicitly validated the success of the voice recognition system integrated in the COBORG.

7.5. FVD Strong/Weak Points

We finished strong in our FVD, but there is always the potential for improvement. Our voice and vision subsystems performed excellently, but the stabilization features of our actuated manipulation subsystem could use refinement. Below is the detailed breakdown of each subsystem, with the pros and cons listed in a digestible format.

7.5.1. Vision Subsystem

- **Strong points:** High detection accuracy, real-time detection (20 FPS after change to tiny version) , accurate octomap for obstacle detection
- **Weak points:** Color-dependent, hand pose variant
- **Future work:** Try different hand detection methods to improve accuracy and inference latency

7.5.2. Voice Subsystem

- **Strong points:** High recognition accuracy, flexible command structure, low/no false-positives, no false commands sent to main_state_machine
- **Weak points:** Sensitive to hot swapped microphones (relies on tuned microphone)
- **Future work:** Add more commands as needed

7.5.3. Actuated Manipulation Subsystem

- **Strong points:** High position accuracy, no collisions with user, tuned torque limits allow for safe backdrivability
- **Weak points:** Inconsistent stabilization, underactuated (4 < 6 DoF), edge case scenarios seize manipulation pipeline
- **Future work:** Implement impedance control (over Resolved Rate), overhaul MoveIt implementation, convert code to a class-based system

8. Project Management

8.1. Work Breakdown Structure

Figure 17 below depicts the updated high-level Work Breakdown Structure (WBS) required to execute the COBORG system. From the high-level breakdown, we were able to derive the necessary tasks to accomplish each of the work packets detailed in the WBS:

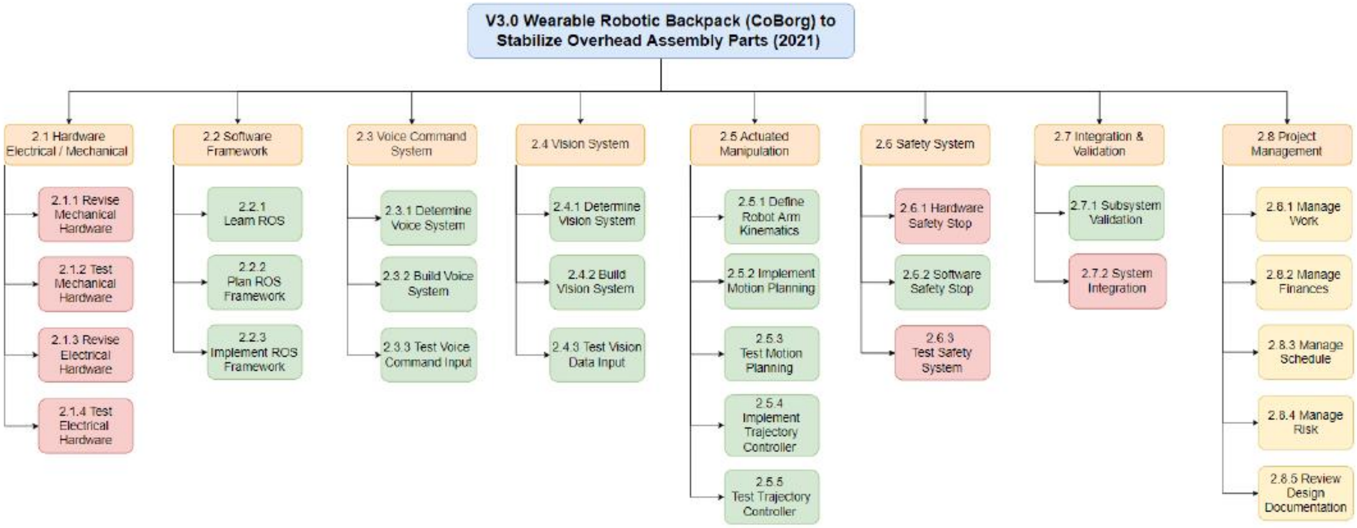


Figure 17 - Current Fall WBS Structure Level 3

A significant difference between our original WBS planned for the Fall and our current WBS is the removal of the "New Task Development." With the overwhelming reality of the Fall semester's progress, we felt that adding a new task on top of what was originally planned was a mistake. With these updated work packets, we were able to refine our timeline regarding our schedule and when key milestones should take place during the Fall.

8.2. Schedule

We were scheduled to begin right on track with the start of the semester. There were three main phases in the Fall development of the COBORG platform:

- Concept Development
- Subsystem Development
- System Integration and Validation

"Concept Development" was a short sprint at the beginning of the semester that focused on creating the design foundations for the hardware framework. After the quick sprint, we commenced with the hardware procurement and developing core subsystem enhancements in parallel. Finally, we had budgeted four weeks of integration and validation to ensure that all components of the COBORG worked and were ready for FVD. The "Fall Work Breakdown Schedule," generated from our WBS, can be found below in Figure 18. Below it, in Table 9, are the primary system development milestones that we achieved during the Fall semester:

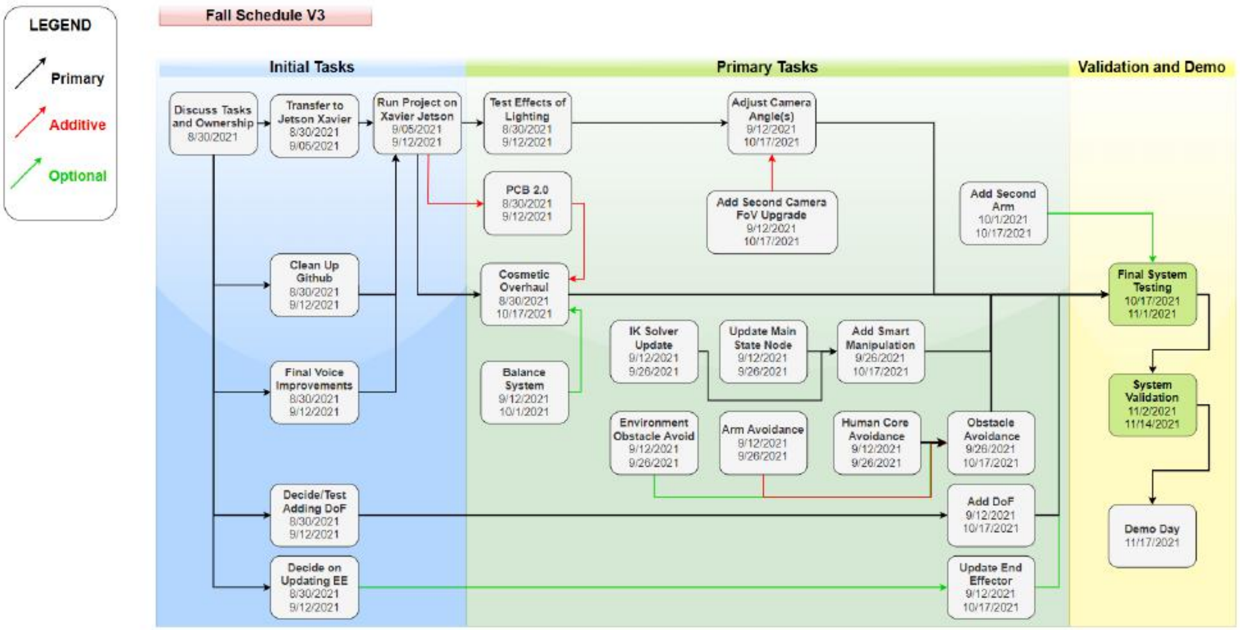


Figure 18 - Fall Work Breakdown Schedule

Table 9 - Major System Development Milestones for Fall

Date	Milestones
08-Sep-2021	Concept Design Completion
01-Oct-2021	New Motor Integration + Carbon Fiber Robot Arm Completion
10-Oct-2021	Hardware Framework Completion
17-Oct-2021	Resolved Rate Stabilization Completion
07-Nov-2021	Full System Integration and Validation
17-Nov-2021	Fall Validation Demonstration

One part of the project that did not meet our original schedule is “Hardware Framework Completion.” This was delayed by over two weeks due to COVID shipping issues that we did not anticipate. Instead of taking two weeks for the parts to arrive, it took over a month for delivery. The hardware framework was completed on October 24th, which placed a massive burden on final integration and validation. We put a ten-day padding in the schedule to account for issues. However, this problem crushed our buffer and caused FVD integration to have zero margins for error. While we successfully demonstrated our use case during FVD, a longer buffer (14 days) would have accounted for impactful delays during hardware development.

8.3. Parts List and Budget

The parts list detailed in Table 10 is actual and covers our expenditures in the Fall semester. Our goal was to utilize our remaining project funds ($3,269) for hardware expansion and risk mitigation. Compared to our estimated expenditure of $1525, our actual expenditure was $3,488.08. Our estimated carbon fiber tubing ended up being double, and purchasing spare components did increase our overall costs. However, most of the additional cost ended up being many small ($10-20) components that were not accounted for in the original budget but were necessary to complete the project. We also decided it would be best to order, test, and fail quickly rather than spend time simulating as our time constraints were much more difficult to deal with than our budgeting constraints. Table 11 details the actual high-level expenditures of the fall semester and includes the borrowed HEBI motors cost that was not factored directly into our budget. The entire parts inventory can be found in Appendix A.

Table 10 - Fall Budget Estimated Expenditure

No.	Part Name	Cost	Quantity	Total Cost
1	Fiberglass Shell Assembly	$400.00	1	$400.00
2	Laser Cut Acrylic Base	$200.00	1	$200.00
3	Carbon Fiber Tubing	$125.00	1	$125.00
4	Electrical Components	$300.00	1	$300.00
5	T-Slotted Aluminum Assembly	$500.00	1	$500.00
Total				**$1,525.00**

Table 11 - Fall Budget Actual Expenditure

No.	Part Name	Cost	Quantity	Total Cost
1	Plastic Sheets	$166.53	1	$166.53
2	Carbon Fiber Tubing	$283.90	1	$283.90
3	Electrical Components	$790.49	1	$790.49
4	T-Slotted Aluminum Assembly + Hardware	$814.94	1	$814.94
5	Misc Items & Risk Mitigation Duplicate Parts	$1,432.22	1	$1,432.22
Total				**$3,488.08**
Borrowed Parts Cost	Hebi X5-9 x 2 Hebi X5-4 x 1 Hebi X8-16 x 1	$12,500	1	$12,500

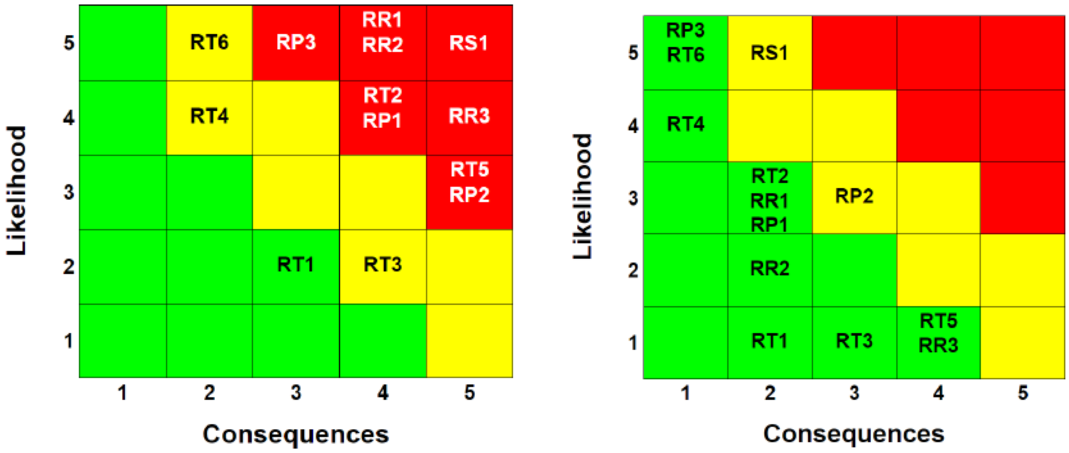
8.4. Risk Management

An essential facet of every project is considering potential risks that may arise as the project progresses forward. These risks span from technical to programmatic and inconsequential to “show-stopping.” To ensure that our project progresses successfully, we have considered significant project risks. We detailed these risks and the tasks taken to mitigate the severity or likelihood of these issues as shown in Table 12 below:

Table 12 - Risk Mitigation Plan

Risk	Label	Mitigation Plans
Hebi motor module dies	RT1	Ensure at least one spare is secured
		Reduce budget to $2,000 to afford spare
Main computer dies or does not perform to our standards	RT2	Work out of Cloud, Github Repository
		Ensure spare workstation is available
		Budget $1300 to purchase spare (Zotac)
Estop devices malfunction	RT3	Inquire Biorobotics lab for spare component
		Inquire John's inventory for spare device(s)
		Ensure Estop device is fail-open
TCP/IP connectivity is lost	RT4	Inquire Biorobotics lab for spare
		Inquire John's inventory for spare device(s)
		If no spare, allocate funds to purchase spare
Water damage to COBORG System	RT5	Seal backpack and perform liquid spill test
		If operating outside, ensure weather is favorable
		Ensure all liquids are 6ft from robot exoskeleton
End effector breaks	RT6	3D print files downloaded into our Google Drive
		3D printed spares on hand
Team lacks ROS fundamentals by start of spring semester	RR1	Execute plan to learn ROS over winter break
		Enforce one week boot camp before Spring
		Hire ROS SME to build framework for project
Unable to work 10hrs/week/member on MRSD project	RR2	Offload work amongst team in certain situations
		Treat MRSD project deadlines as HW deadlines
Hazard occurs on user while wearing robot	RR3	Be prepared to act when the issue arises
		Brief and prep the user on proper procedure
		Perform on-the-rack run through before testing
		Set torque limits and vision obstacle avoidance
Member contracts COVID	RP1	Follow CMU pandemic safety procedures
		Create simulation of system in ROS
MRSD program gets disrupted due to COVID pandemic	RP2	House robot outside of lab environment
		Create simulation of system in ROS
Our sponsor graduates in the spring of 2020	RP3	Confirm Julian can commit time after graduation
		Acquire non-CMU contact information
Hardware components do not arrive on time (2+ week delay)	RS1	Reach out to parts manufacturer and request accelerated production
		Order duplicate components from McMasterCarr

We have ensured that none of the risks found have a critical impact or likelihood as the project is executed. One new risk that significantly impacted our project this semester is “RS1 - Hardware components do not arrive on time”. This greatly affected our ability to integrate the project. Had we not taken steps to reduce the consequences of this risk, our project would not have been completed on time. Figure 19 demonstrates the risks before mitigation and after:



Likelihood	1	2	3	4	5
5	🟩	🟨 RT6	🟥 RP3	🟥 RR1 RR2	🟥 RS1
4	🟩	🟨 RT4	🟨	🟥 RT2 RP1	🟥 RR3
3	🟩	🟩	🟨	🟨	🟥 RT5 RP2
2	🟩	🟩	🟩 RT1	🟨 RT3	🟨
1	🟩	🟩	🟩	🟩	🟨
	Consequences				

Likelihood	1	2	3	4	5
5	🟩 RP3 RT6	🟨 RS1	🟥	🟥	🟥
4	🟩 RT4	🟨	🟨	🟥	🟥
3	🟩	🟩 RT2 RR1 RP1	🟨 RP2	🟨	🟥
2	🟩	🟩 RR2	🟩	🟨	🟨
1	🟩	🟩 RT1	🟩 RT3	🟩 RT5 RR3	🟨
	Consequences				

Figure 19 - Risk Mitigation Before and After

9. Conclusions

9.1. Lessons Learned

While overall, we were able to execute well on the plans we made in the previous semester; there were still several challenging aspects to this project that we encountered. The biggest lesson learned is that hardware delays have cascading effects and that delays in integration can lead to devastating results. As a result, we had to be very thoughtful in how we budgeted time for the project with hardware delays and eventually worked continuously to overcome the bottlenecks. With this strategy, we were able to have five people working 80-120 hours per week to stay on track with project objectives.

Another lesson we learned was that documentation proved to help us remember our thought process as we developed the COBORG system. We took meeting minutes weekly and used a versioning control system to track what was edited in the code. We also made sure to create “readme” guides for every subsystem so that those who were not subject matter experts could quickly gain expertise and support the project.

We had minimal reliance on version control during the spring semester as everyone worked on their subsystems. However, these tools became invaluable during the fall semester as we began integrating our work and making changes, adjustments, and updates to fix problems that arose. Using version control software allowed everyone to work simultaneously from different computers while maintaining our project’s integrity. This became critically important as the semester wore on and the work got progressively more intense.

Also, during the spring semester, parts of the actuated manipulation subsystem were written with an ‘as soon as possible’ mentality instead of using proper software architecture guidelines. This was fine for individual development but created massive headaches during integration. In addition, since it wasn’t written as a class-based system, it required significant effort to adjust functionality while maintaining consistency throughout the program. If it had been written using proper architecture in the Spring, it would have been easier to adjust, handoff to others, possibly saving weeks of development work.

9.2. Future Work

If we were to use our current platform as the basis of a budding startup, the first thing we would do is scrap pure position control for impedance control. After many hours of tweaking and tuning the system to work with position control, it became evident that the entire approach was flawed. Most likely, a simple force/position hybrid control would work much better than a pure position for this use case. We would also strive to make the COBORG as light as possible. Even at 20lbs, the device may be too heavy to be worn all day. We may have to switch to cloud computing and only attach the arm and a thin client to the frame. The frame itself was designed to be robust and easy to modify; however, it is far too bulky as a product. We would have to slim almost every dimension of the COBORG in half to become market-ready.

10. References

[1] Owano, N. (2018, June 21). Ford, exoskeleton company address strain in overhead tasks. Retrieved December 18, 2020, from https://techxplore.com/news/2018-06-ford-exoskeleton-company-strain-overhead.html

[2] Farris, Riley & Pitt, LLP. (n.d.). Retrieved December 18, 2020, from https://www.frplegal.com/industrial-accidents/8-most-common-workplace-injuries-for-factory-workers/

[3] Bruggisser, F. (n.d.). Yolo-hand-detection. Retrieved December 18, 2020, from https://github.com/cansik/yolo-hand-detection

11. Appendix

11.1. Appendix A: Bill of Materials

Line Number	Part No.	Quantity	Part Name	Unit Price	Total Price
1	2280	1	Crucial P2 500GB 3D NAND NVMe PCIe M.2 SSD	$52.99	$52.99
2	T265	1	Intel® RealSense™ Tracking Camera T265	$199.00	$199.00
3	SAT32M225	1	StarTech.com M.2 SSD to 2.5in SATA Adapter - M.2 NGFF to SATA Converter - 7mm - Open-Frame Bracket - M2 Hard Drive Adapter (SAT32M225) Green	$19.63	$19.63
4	MN16GSD42666	1	PNY 16GB DDR4 2666MHz Notebook Memory RAM	$67.85	$67.85
5	4051	1	10 Series 180 Degree Right Hand Pivot Bracket Assembly with "L" Handle	$26.25	$26.25
6	3321	**5**	Bolt Assembly: 1/4-20 x .500" Black FBHSCS and Slide-In Economy T-Nut - Centered Thread - Black Zinc	$0.50	$2.50
7	N12	1	USB Powered Computer Stereo Speaker	$19.99	$19.99
8	UL10	1	Flanged Wing Nut Black Zinc Alloy, 6-32 Thread Size, 3/4" Wide	$22.99	$22.99
9	CB-C64	1	AUKEY USB C Hub Ultra Slim USB C Adapter	$11.99	$11.99
13		1	Nvidia Jetson Xavier AGX	$699.00	$699.00
14		1	Tacklife HD60 Classic Laser	$32.27	$32.27
15		1	Digikey PCB Parts List	$379.73	$379.73
16		1	UWS-5/10-Q48P-C	$32.50	$32.50
17		20	1331-BK	$0.12	$2.42
18		20	261G2	$0.12	$2.42
19		20	1327-BK	$0.61	$12.20
20		20	1327G6-BK	$0.61	$12.20
21		1	Dynamixel Starter Kit	$90.95	$90.95
22		1	McMasterCarr Hardware Items	$266.04	$266.04
23		1	80/20 Hardware Items	$311.95	$311.95
24		2	1" Tube End Cap	$8.25	$16.50
25		2	1" Threaded End Connector	$14.25	$28.50
26		2	1" Threaded End Connector with Stud	$14.25	$28.50
27		1	Carbon Fiber Roll Wrapped Twill Tube 1" ID x 72" Tube Options: 1/8" Wall/ Standard Modulus (33 msi)	$236.81	$236.81
28		1	1/4-20 x 3/4" Button Head Socket Cap Screws, Allen Socket Drive, Black Oxide, Alloy Steel Class 10.9, Fully Threaded, Machine Thread, 25 PCS	$8.99	$8.99
29		1	1/4-20 x 1-1/4" Button Head Socket Cap Screws, Allen Socket Drive, Black Oxide, Grade 10.9 Alloy Steel, Fully Threaded, 25 PCS	$9.19	$9.19
30		1	18-8 Stainless Steel Pan Head Machine Screw, Black Oxide Finish, Meets MS-51958, #2 Phillips Drive, #10-32 Thread Size, 3/8" Length, Fully Threaded, USA Made (Pack of 25)	$6.04	$6.04
31		1	Powerwerx Panel Mount Housing for Two Anderson Powerpole Connectors with a Weather Resistant Cover	$24.99	$24.99
32		1	Block Erupter DC Power 12V 5.5mm x 2.5mm Barrel Male Plug Connector 12ft Pigtail 16AWG (1 Pack)	$19.10	$19.10
33		1	InstallGear 12 Gauge AWG 30ft Speaker Wire True Spec and Soft Touch Cable - Red/Black	$11.95	$11.95
34		1	Glarks 10 Pair 30AMP Quick Disconnect Power Terminals Connectors, Red Black Quick Connect Battery Connector Modular Power Connectors Set	$12.99	$12.99
35		1	Metal Deburring Tool Kit Deburring Cutters Set	$11.99	$11.99
36		1	Industrial Grade 160 Watts 2.5 Liters Digital Heated Ultrasonic Cleaner	$79.98	$79.98
37		1	GeeekPi AC8265 Wireless NIC Module with 5db Antennas for Jetson Nano	$24.20	$24.20
38		1	NEW Microsoft Bluetooth Desktop - Matte Black	$36.99	$36.99
39		1	Cable Matters Long USB to USB Extension Cable (USB 3.0 Extension Cable) in Black 10 ft for Oculus Rift, HTC Vive, Playstation VR Headset and More	$8.99	$8.99
40		1	PCI Express Adapter Cable x1 PCIe Adapter Cable PCI-e PCI Express 36PIN	$25.99	$25.99
41		1	Aluminum Threaded Rod 6-32 Thread Size, 3/4" Long	$9.79	$9.79
42		1	Flanged Wing Nut Black Zinc Alloy, 6-32 Thread Size, 3/4" Wide	$9.11	$9.11
43		1	18-8 Stainless Steel Low-Profile Socket Head Screws with Hex Drive, M3 x 0.5 mm Thread, 8 mm Long	$7.55	$7.55
44		1	18-8 Stainless Steel Low-Profile Socket Head Screws with Hex Drive, M3 x 0.5 mm Thread, 5 mm Long	$1.68	$1.68

45		2	Primex ABS - 1/4" x 20.5 x 17.5	$23.49	$46.98
46		3	Primex ABS - 3/32" x 24" x 48"	$23.60	$70.80
47		1	3M 2216 Epoxy	$55.99	$55.99
48		1	Powerwerx PanelPole1, Panel Mount Housing for a Single Anderson Powerpole Connector with a Weather Tight Cover	$22.99	$22.99
49		1	2 Ports Dual USB 3.0 Male to USB 3.0 Female AUX Flush Mount Car Mount Extension Cable for Car Truck Boat Motorcycle Dashboard Panel -(3 Feet)	$11.50	$11.50
50		1	dkplnt 20A 240W 12v Golf Cart 48V 36V to 12V Converter Voltage Regulator Reducer Transformer with Fuse	$24.99	$24.99
51		1	Hilitchi 40-Pieces 2 3 4 5 Pin 16mm Thread Male Female Panel Metal Aviation Wire Wire Connector Plug Assortment Kit (2 Pin / 3 Pin / 4 Pin / 5Pin)	$19.93	$19.93
52		2	Molex 2 Pin Connector Lot, 6 Matched Sets, w/18-24 AWG w/Pins Mini-Fit Jr	$9.35	$18.70
53		1	NETGEAR 5-Port Gigabit Ethernet Unmanaged Switch (GS305) - Home Network Hub, Office Ethernet Splitter, Plug-and-Play, Silent Operation, Desktop or Wall Mount	$15.99	$15.99
54		1	Heat Shrink Tube 10 Sizes Tubing Wrap Sleeve Set Combo	$7.99	$7.99
55		1	Sysly IDC40 2x20 Pins Male Header Breakout Board Terminal Block Connector with Simple DIN Rail Mounting feet	$23.99	$23.99
56		1	40p to 40p GPIO Ribbon Cable for Raspberry Pi 4/3 / Zero / 2 (8" 20cm)	$8.08	$8.08
57		1	AC Infinity AIRPLATE S3, Quiet Cooling Fan System 6" with Speed Control, for Home Theater AV Cabinets	$28.99	$28.99
58		1	AC Infinity Ventilation Grille, for PC Computer AV Electronic Cabinets, Also mounts one 120mm Fan	$11.99	$11.99
59		1	SparkFun Logic Level Converter - Bi-Directional	$2.95	$2.95
60		2	SparkFun Beefcake Relay Control Kit (Ver. 2.0)	$8.95	$17.90
61		1	NiceCo Treadmill Safety Key, Universal Treadmill Magnet Security Lock, 28mm Magnetic Shell, Replacement Kit for Sole, Weslo, Weider, Epic, Healthrider (Red)	$6.29	$6.29
62		1	RESALET Magnetic Reed Switch Alarm NC Normally Closed Surface Mount Wired Contact Sensor for Gate Garage Window Door Security System Silver Gray MC-52	$12.49	$12.49
64		1	Cable Matters 10-Pack Snagless Short Cat6 Ethernet Cable (Cat6 Cable, Cat 6 Cable) in Black 1 ft	$17.99	$17.99
65		4	Monoprice 0.5FT 24AWG Cat6 550MHz UTP Ethernet Bare Copper Network Cable - Black	$1.60	$6.40
66		1	RJ45 Coupler, Ethernet Coupler, in Line Coupler for Cat7/Cat6/Cat5e/Cat5 Ethernet Cable Extender Adapter Female to Female (6 Pack Black)	$11.99	$11.99
67		1	Black-Oxide Alloy Steel Socket Head Screw M5 x 0.8 mm Thread, 15 mm Long	$11.73	$11.73
68		1	Black-Oxide Alloy Steel Socket Head Screw M5 x 0.8 mm Thread, 10 mm Long	$12.14	$12.14
69		1	50ft PET Expandable Braided Cable Sleeve, Wire Sleeving with 127 Pieces Heat Shrink Tube for Audio Video and Other Home Device Cable Automotive Wire (1/4 Inch, 1/2 Inch, 3/4 Inch, Black)	$13.99	$13.99
70		1	Skog Bands: Heavy Duty Rubber Bands Made from EPDM Rubber - 5col Survival Supply (Big Mix)	$10.99	$10.99
71		1	Sata Male/Female	$6.99	$6.99
72		1	McMaster Carr Hardware Order	$94.62	$94.62
73		1	Cable Matters Snagless Long Cat6 Ultra Thin Ethernet Cable (Thin Cat6 Cable) in Black 50 ft	$11.49	$11.49
74		1	Klein Tools VDV826-703 Pass-Thru Modular Data Plug, RJ45 CAT6, Pass Through Connectors 50-Pack	$24.64	$24.64
75		1	RJ45 Professional Crimp Tool Pass Through Cat5 Cat5e Cat6 Heavy Duty Crimping	$19.99	$19.99
76		1	trueCABLE RJ45 Strain Relief Boots, Small 5.5 to 7.0mm Cut-to-Fit PVC Plug Cover	$7.99	$7.99
77		1	Rust-Oleum 249127 Painter's Touch 2X Ultra Cover, 12 Ounce (Pack of 1), Flat Black	$7.26	$7.26
78		1	20Pcs 20x28 Corner Bracket Right Angle 2020 Series Connector 2 Hole Aluminum Brackets for Aluminum Extrusion Profile with Slot 6mm	11.49	$11.49
80		1	80/20 EQUIVALENT 3313 - 1/4-20 Drop-In T-Nut w/set-screw for 10 Series (25 pcs) (265361401417)	$60.00	$60.00
81		1	Champion Cutting Tool Brute Platinum XL22-1/4-20 Heavy Duty Spiral Point Tap: (Individual Pack)- MADE IN USA	$11.73	$11.73
83		1	Self Adhesive Weather Stripping Rubber Insulation Foam Seal Tape 1/2" Wide X 1/8" Thick Total 50 ft Long（16.5ft x 3 Rolls)	$7.99	$7.99
84		1	3M Super Strength Molding Tape, 03614, 1/2 in x 15 ft	$9.10	$9.10
85		1	BATIGE Single Port USB 3.0 Male to Female AUX Car Mount Flush Cable Waterproof Extension for Car Truck Boat Motorcycle Dashboard Panel - 3ft	$8.90	$8.90
86		1	uni USB C to USB Adapter 2 Pack, Aluminum USB-C [Thunderbolt 3] to USB 3.0 Adapter OTG Cable Compatible for MacBook Pro 2020/2019/2018, MacBook Air/iP	$11.99	$11.99
87		1	SATA Power Cable, CableCreation [2-Pack] 6-Inch SATA 15 Pin Male to 2xSATA 15 Pin Down Angle Female Power Splitter Cable	$6.99	$6.99
88		1	Barbed Inserts for Plastic, Brass, 1/4"-20 Thread Size, 0.3" Installed Length, packs of 10	$9.14	$9.14
89		1	RESALET Magnetic Reed Switch Alarm NC Normally Closed Surface Mount Wired Contact Sensor for Gate Garage Window Door Security System Silver Gray MC-52	$13.49	$13.49
90		1	UGREEN Micro USB 3.0 Cable USB 3.0 Type A Male to Micro B Cord Compatible with Samsung Galaxy S5 Note 3 Camera Hard Drive and More 1.5ft	$5.99	$5.99
91		2	Amazon Basics Fast Charging 60W USB-C3.1 Gen2 to USB-A Cable - 3-Foot, Black	$8.87	$17.74

92		1	4" Anti Slip Furniture Pads	$12.99	$12.99
93	UL10	1	USB Lavalier Microphone	$22.99	$22.99
94		2	Molex 2 Pin Connector Lot, 6 Matched Sets, w/18-24 AWG w/Pins Mini-Fit Jr	$9.35	$18.70
95		1	UGREEN 3ft cable USB 3.0 Type A Male to Micro B Cord	$7.99	$7.99
96		1	Headrest	$19.99	$19.99
97		1	USB Hub	$19.99	$19.99
98		1	Confined-Space Conical Compression Spring, 3" Long, 2.25" x 1.125" OD, 2.05" x 0.925" ID, Packs of 1	$18.38	$18.38
99		1	PoiLee 5pcs Capacitor 10000uf 25v 18X35MM, Radial Electrolytic Capacitor 25v Low ESR	$8.68	$8.68
100		1	PoiLee 5pcs Capacitor 10000uf 25v 18X35MM, Radial Electrolytic Capacitor 25v Low ESR	$8.68	$8.68
101		1	PoiLee 5pcs Capacitor 10000uf 25v 18X35MM, Radial Electrolytic Capacitor 25v Low ESR	$8.68	$8.68
102		3	2PCS 10000uF 25V Rubycon PX 105 Degrees 18x35 mm Filter Capacitor	$38.97	$116.91
103		1	uxcell Rolling Door Contact Magnetic Reed Switch Alarm with 3 Wires for N.O./N.C. Applications MC-52	$17.59	$17.59
104		1	Pyramex Ridgeline Cap Style Hard Hat, 4-Point Ratchet Suspension, Shiny Black Graphite Pattern	$25.63	$25.63
105		1	Led Headlamp Rechargeable, Super Bright 1500 Lumens 220° Wide Beam Headlamp, 3 Modes IPX6 Waterproof Lightweight Headlamp with Taillight & Clips for Camping, Outdoors, Hard Hat Work, Emergency	$34.99	$34.99
106		1	Super Bright Headlamp, USB Rechargeable Led Head Lamp, IPX4 Zoomable Waterproof Headlight with 4 Modes and Adjustable Headband, Hard Hat Light Perfect for Camping, Hiking, Outdoors, Hunting, Running	$15.99	$15.99
107		1	NATGAI Sponge Neoprene with Adhesive Foam Rubber Sheet 1/4" Thick X 12" Wide X 54" Long, Cut to Multiple Dimensions and Lengths - DIY, Gaskets, Cospla	$19.95	$19.95
108		1	Hard-to-Find Fastener 014973458911 Multi-Purpose Wood Screw, 6 x 1, Yellow, 50	$3.75	$3.75
109		1	Hard-to-Find Fastener 014973458959 Wood Screw, 8 x 1-1/2", Yellow, 30	$3.75	$3.75
110		2	DEWALT Magnetic Drive Guide, 1/4-Inch (DW2054)	$11.90	$23.80
111		1	Fasgear USB C to Micro USB Cable 30cm Nylon Braided Type C to Micro USB Cord Compatible with Galaxy S7/S6, HTC One/10 and More (Black, 1ft)	$5.99	$5.99
112		1	GL.iNet GL-MT300N-V2(Mango) Portable Mini Travel Wireless Pocket VPN Router - WiFi Router/Access Point/Extender/WDS | OpenWrt | 2 x Ethernet Ports | OpenVPN/Wireguard VPN | USB 2.0 Port | 128MB RAM	$29.90	$29.90
113		1	E-Z LOK 400-008 Threaded Inserts for Wood, Installation Kit, Brass, Includes 8-32 Knife Thread Inserts (10), Drill, Installation Tool	$13.82	$13.82
114		1	19V 3.42A 65W AC Adapter Replacement for ASUS X45A X550 X550ZA X550LA X551 F555 AD887320 PA-1650-78 A56C A56CA A56CM ADP-65GD B EXA1203YH 5.5/2.5mm Po	$15.88	$15.88
115		1	Home Depot Items	$31.02	$31.02
116		1	595-OPA548T-1G3	$17.11	$17.11
117		1	Home Depot Items	$38.67	$38.67
118		1	USB Computer Microphone with Mute Button, Plug&Play Condenser, Desktop, PC, Laptop, Mac, PS4 Mic -360 Gooseneck Design -Recording, Dictation, YouTube,	$17.99	$17.99