



WPI

Project Number: RBE.TP1.CPS2

Robolink: Modular Assistive Robot Arm

A Major Qualifying Project Report

Submitted to the faculty of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Bachelor of Science

Advisor:

Taskin Padir

Submitted By:

Cassiopia Hudson

Gabriel Morell-Pacheco

1 Acknowledgments

The team would like to extend our thanks to several people who helped make this project possible:

Taskin Padir, for his help and guidance as our advisor,

Ty Tremblay, for configuring the Robolink and getting us started,

Lillian Walker, for contributing her time and CAD experience,

Joe St. Germain, for his constant and helpful advice,

James Fleming, for his expertise on the Cyber Physical Systems project, and

Thomas Angelotti and Patrick Morrison, for opening the ECE shop to us.

2 Abstract

The goal of this project is to utilize the igus® Robolink arm five degree of freedom modular robot arm, to complete useful tasks for persons with no or limited mobility. These tasks include driving the joystick of a wheelchair, flipping a light switch, and turning the pages of a book. This is done through designing and building a modular interface for mounting the Robolink arm onto an existing wheelchair project and implementing a universal control interface in the software for future expansion of tasks and control methods.

3 Table of Contents

- 1 Acknowledgments 2**
- 2 Abstract..... 3**
- 3 Table of Contents 4**
- 4 Table of Figures..... 5**
- 5 Table of Tables 6**
- 6 Introduction..... 7**
 - 6.1 Explore existing technology..... 7**
 - 6.2 Task Specifications..... 8**
 - 6.3 Design Specifications..... 8**
- 7 Background 10**
 - 7.1 igus® Robolink Applications..... 10**
 - 7.2 Similar Projects 12**
- 8 Methodology 15**
 - 8.1 Review of Previous Work 15**
 - 8.2 Proof of Concept Experiments 15**
 - 8.3 Design Constraints 16**
 - 8.3.1 Mounting the Robolink 16
 - 8.3.2 Passive End Effector 17
 - 8.4 Design Solutions 17**
 - 8.4.1 Mounting Arm 17
 - 8.4.2 Power Supply 19
 - 8.4.3 Controller 19
 - 8.4.4 Wheelchair Joystick 20
 - 8.4.5 End Effector 22
 - 8.5 Completion of Tasks..... 23**
 - 8.5.1 Joystick Code 23
 - 8.5.2 Robolink Controller 25
- 9 Results & Discussion..... 27**
 - 9.1 Mounting Arm 27**

9.2	Power Supply	30
9.3	Controller	30
9.4	Wheelchair Joystick	30
9.5	End Effector	30
9.6	Completion of Tasks & Experimental Results	30
9.6.1	Drive Joystick	30
9.6.2	Flip Light Switch	31
9.6.3	Turn Page of Book	38
10	Conclusion	39
10.1	Goal Completion	39
10.1.1	Task Specifications	39
10.1.2	Design Specifications	40
10.2	Societal Considerations	40
10.3	Future Recommendations	41
11	Bibliography	43
12	Appendix	48
12.1.1	Instructions for Use	48
12.1.2	Repair History / Incident Report	49
12.1.3	Presentation Slides	51

4 Table of Figures

Figure 1:	Wheelchair with Joystick and Desk	9
Figure 2:	Robolink Cable Box	10
Figure 3:	igus Robolink arm	11
Figure 4:	FLASH Humanoid Robot	11
Figure 5:	Robolink Fishtail Robot	12
Figure 6:	Gripper with "Fingers" Tied	16
Figure 7:	Mounting Fixture	17
Figure 8:	Mounting Structure	18
Figure 9:	Raspberry Pi	19

Figure 10: Simple Arcade Joystick	20
Figure 11: CTI Electronics Industrial Joystick	21
Figure 12: Emergency Stop Button.....	22
Figure 13: End Effector	23
Figure 14: End Effector Over Joystick	23
Figure 15: Workspace of Arm – Vertical	28
Figure 16: Workspace of Arm - Joystick Driving	28
Figure 17: Workspace of Arm – Horizontal	29
Figure 18: Workspace of Arm - Over Desk.....	29
Figure 19: Light Switch With Chair Facing Wall.....	32
Figure 20: Light Switch With Chair at 45-Degree Angle.....	33
Figure 21: Forces on a Light Switch.....	35
Figure 22: Light Switch With Chair at 90-Degree Angle.....	37
Figure 23: Rocker Light Switch.....	38
Figure 24: Robolink Typing on Laptop	39
Figure 25: 80/20 Extruded Aluminum.....	40
Figure 26: Malleable End Effector	42
Figure 27: Motor Cable Pulled from Fastener	49
Figure 28: Current Motor Axle Bend.....	50

5 Table of Tables

Table 1: Similar Project Comparison.....	14
Table 2: Returns and Updates Rolling Average	24
Table 3: Logic for Determining Wheelchair Speed.....	24
Table 4: Example of Use.....	26
Table 5: Configuration 1 Necessary Force	36
Table 6: Configuration 2 Necessary Force	37

6 Introduction

Tasks such as drinking from a glass of water, preparing a meal, or doing housework are daily activities most people can perform without much difficulty. For a person with limited mobility, these Instrumental Activities of Daily Living (IADL) can be difficult, or even impossible, without help or supervision. A disability statistics report written in 2000 that focused on the use of mobility devices in the United States outlines that 1.7 of the 6.8 million Americans who need assistive devices, need wheelchairs or scooters to help them with mobility [19]. More than two thirds of these people (68.1 percent) also need assistance with at least one IADL [19]. Wheelchair mounted robotic arms have been developed since the early 1990's to help conquer this problem [19].

The purpose of this project is to establish control of a wheelchair mountable modular robotic arm to complete tasks useful for a person with no or limited mobility. The igus® Robolink is a five degree-of-freedom (DOF) modular robotic arm that is mounted to augment the person's ability to complete tasks. The goals of this project are to utilize the Robolink to drive the joystick of a wheelchair, flip a light switch, and turn the page of a book.

6.1 Explore existing technology

Existing Robolink applications vary in scope from humanoid robots to industrial processes. Research of Robolink applications, including a review of their Youtube playlist “igus® robolink® robot modules applications and references” yielded no other cases of the Robolink's use as a wheelchair mounted assistive robot arm (igus GmbH). This led to research on the history of assistive robot arms. The team learned that work on assistive robotic arms started in the early 1990's, an example of which is the Manus ARM made by Exact Dynamics [39]. The ongoing trend is that robot arms are now smaller and lighter than their earlier counterparts while increasing in functionality and usability. Advantages of the igus® for human assistive projects include it's low cost, approximately \$2,000 as opposed to the approximate \$48,000 JACO arm, and compact size, compared to arms like DORA [20], [30]. The igus® Robolink has a maximum of six DOF, however this configuration utilizes only five. Most wheelchair mounted assistive robotic manipulators utilized 5-7 DOF (Table 1: Similar Project Comparison). As with most existing projects, the Robolink arm used has the extension capability of just above a meter (Table 1: Similar Project Comparison).

6.2 Task Specifications

A set of task specifications served as guidelines for the project. The most important task specification involved driving the wheelchair using the Robolink arm. Specifications for driving revolved around creating smooth motion of the chair for the passenger, so abrupt acceleration and deceleration were to be avoided. The arm needed to move the joystick of the wheelchair front, back, left, and right within a few seconds of receiving the command. The second task was flipping a standard height light switch. The arm needed to reach out to a 48in. high light switch on a wall, taking less than 15 seconds to flip the switch in either direction. The third task involved flipping the pages of a hardcover book, taking less than 30 seconds to open the front cover, and less than 30 seconds to turn a page. The time constraints were chosen arbitrarily based on the team's opinions of reasonable time intervals.

6.3 Design Specifications

Several design specifications were developed to guide the design process. The Robolink arm and its attached control box (12" x 12" x 4") containing the cables, motors, and controllers (referred to as the "cable box") shown in Figure 2: Robolink Cable Box needed to be stably mounted on the wheelchair. This mounting needed to allow the end effector of the Robolink to reach the joystick of the wheelchair at the end of the right armrest and the desk along the left armrest of the wheelchair seen in Figure 1: Wheelchair with Joystick and Desk.

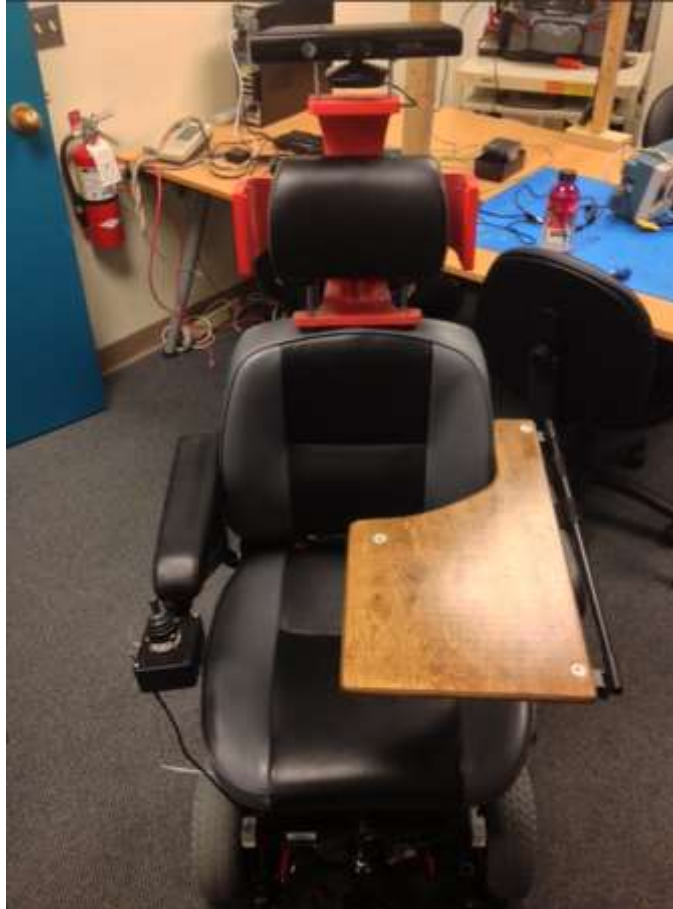


Figure 1: Wheelchair with Joystick and Desk

The arm also needed to be mobile, so an interface needed to be developed to draw power from the wheelchair's power supply, and the Robolink needed to be controlled by a portable small board or laptop.

One goal of this project involved experimenting with passive end effectors to avoid using power and pneumatics to control a gripper, so a passive end effector needed to be developed. The passive end effector needed to be modular and easily replaceable, while versatile enough to complete multiple task specifications.

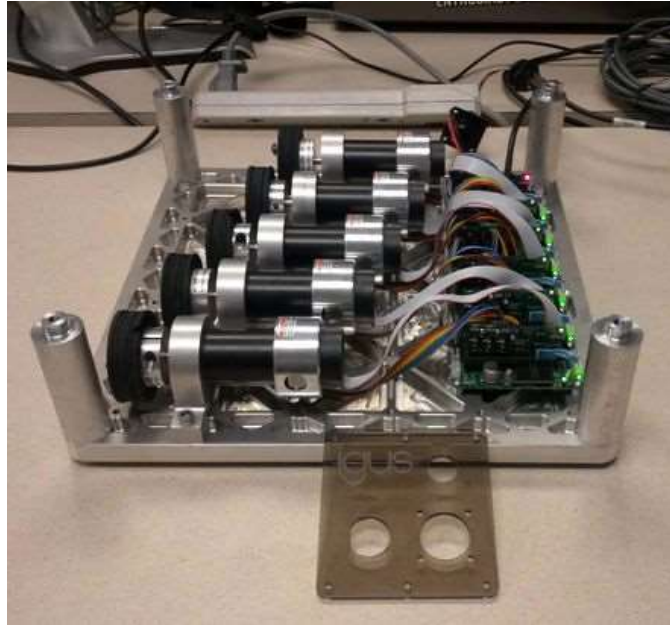


Figure 2: Robolink Cable Box¹

7 Background

7.1 igus® Robolink Applications

The Robolink system was made by igus® and was released in 2011. It is composed of a set of lightweight plastic joints, metal tubes, drive units, cables and end attachments that are meant to work together to produce an infinite variety of low cost and low maintenance robot arms.

The Robolink system (Figure 3: igus Robolink arm) has a large number of example applications ranging from swimming robots, to pick and place industrial robots, to humanoid robots. The team found a few good examples of the Robolink system in use in humanoid robotics, all of which extend the maximum number of DOF afforded by the Robolink via complex grippers and hands.

¹ <http://robot.wpi.edu/wiki/index.php/Robolink>



Figure 3: igus Robolink arm²

The most impressive of these is the Flexible LIREC Autonomous Social Helper (FLASH) from the Wrocław University of Technology, which has seven DOF per arm and incorporates two arms [21]. The Robolink arms on FLASH are “purposed to express emotions by means of ... gesticulation” [21]. It is mounted on a self-balancing platform similar to a Segway, shown in Figure 4: FLASH Humanoid Robot.

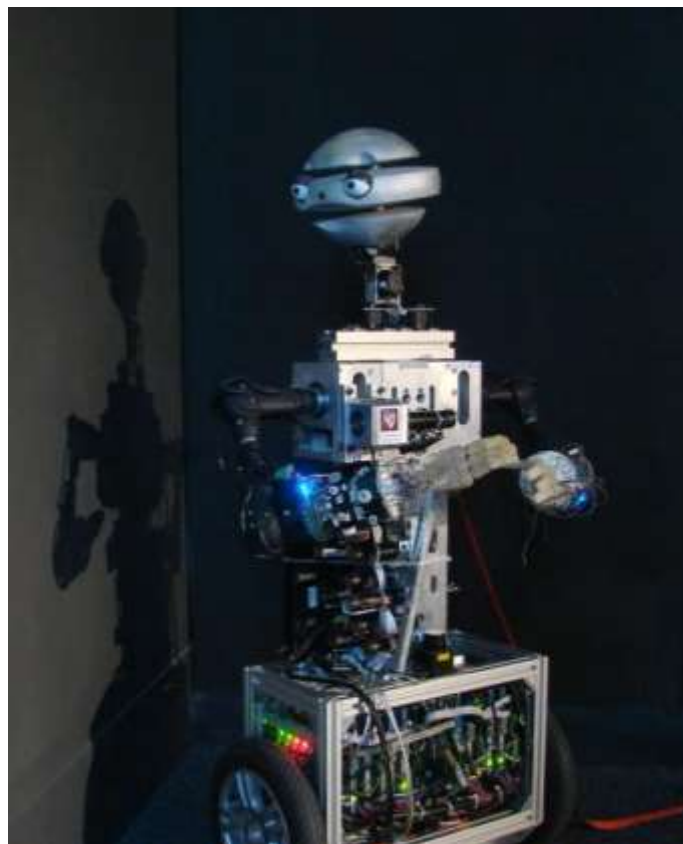


Figure 4: FLASH Humanoid Robot³

² http://www.igus.com.sg/wpck/default.aspx?Pagename=robolink_articulatedarm&C=SG&L=en

³ <http://www.flash.lirec.ict.pwr.wroc.pl/>

Another humanoid, Humanoid Robot A1 (ADAM), has six DOF and two arms as well. Both are some of the more complex examples of Robolink applications. On the other end of the spectrum is a swimming robot from the German Research Center for Artificial Intelligence (DFKI GmbH) that uses a Robolink joint to actuate a mechanical fish tail to propel a robot through water seen below in Figure 5: Robolink Fishtail Robot. The research conducted suggests that the Robolink arm used in our project is the only instance of Robolink currently being used for assistive robotics.

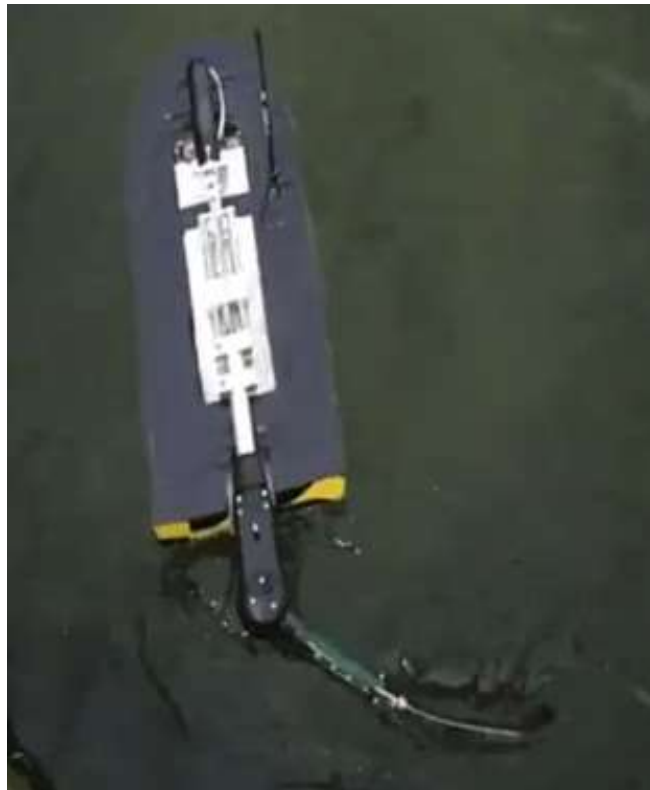


Figure 5: Robolink Fishtail Robot⁴

7.2 Similar Projects

To learn more about assistive robotic arms, the team researched previous endeavors in the field over the past twenty years. Table 1: Similar Project Comparison, organized by year, was developed to compare eight similar projects to our work with the Robolink. The team members based on photos, videos, and other research determined ratings in categories like “Size” and “Number of Tasks”. The table also compares categories such as Control, Degrees of Freedom, Maximum Extension, and Weight of Arm.

⁴ <http://www.youtube.com/watch?v=4ajf4Ob-BGk&list=PL07B939DF7512B3B4>

Many previous assistive wheelchair mounted robotic arms have used joysticks to control the arm, but alternative control methods have been explored. One of the first arms by Exact Dynamics used a combination of a camera and a touch screen interface that allows a human to control the arm [40]. Touch screens have been used in robotic arms but more recently the use of brain signal processing has been a focus of research. This enabled the use of assistive arms that do not require physical input from the user, such as the movement of a hand to utilize a joystick, so that the arm could be used by persons with no mobility. As the field has progressed, arms have been getting smaller and lighter, with the weight of current generation arms being less than half of the first iterations. The Robolink arm being used by the team is recent and weighs 10.3 kg (arm, cable box, and mounting hardware). As well as being light, the Robolink arm the team is using is highly compact even compared to other Robolink implementations. The standard stepper motors used to drive the cables have been replaced with a custom drive module, which is far smaller.

As seen in Table 1: Similar Project Comparison, many existing assistive robotic arms use a gripper, one of the project's goals involved implementing a passive end effector to keep complexity down. This goal is to prove that a well designed and thought out passive end effector is as useful in performing a range of tasks as one that is specialized.

In order for the end effector to be effective, the system needs a workspace that allowed it to perform its task. As seen in Table 1: Similar Project Comparison, the baseline minimum number of degrees of freedom for an assistive robotic arm is five. Any fewer degrees and the reach of the arm is limited and starts to have difficulty performing tasks due to not being able to reach points in space. The Robolink configuration being used for this project has five, which allows for versatile reach and task performance while keeping complexity down.

The reach of existing robotic arms has historically been approximately a meter with variances of $\pm 20\%$. In the wheelchair configuration being used the reach of the Robolink arm is 1.187 meters, which is on the higher end of historical reaches.

Table 1: Similar Project Comparison

Robot Arm Name:	Developed By:	Year Developed:	Control:	Weight of Arm:	Size: (Compact (1) to Bulky (5))	End Effector:	Number of Tasks: (Few (1) to Many (5))	Weight can lift (Payload):	Extension:	Degrees of Freedom:	Ref:
Robolink	igus®	2013	Brain Signals, Joystick	10.3 kg	1	Passive	3	N/A	1.187 m	5	[16]
RAPUDA (Robotic Arm for Persons with Upper Limb DisAbilities)	Intelligent Systems Research Institute (AIST) Japan	2010	Joystick	5.897 kg	3	Gripper	3	0.454 kg	0.914 m	5	[13] [48]
Human in Loop Integration of An Arm Mounted Wheelchair Robot Based on RT Middleware	Sugano Lab Waseda University	2010	Joystick, Touch Screen, Brain Interface	5.000 kg	2	Gripper	4	1.000 kg	1.070 m	7	[46] [44]
JACO Rehab Edition	Kinova	2010	Joystick	5.700 kg	1	Gripper	5	1.500 kg	0.900 m	6	[18] [20]
PerMMA (Personal Mobility and Manipulation Appliance)	Carnegie Mellon University, University of Pittsburgh	2009	Joystick, Brain Signals, Touch Screen	28.600 kg	5	2x Gripper	5	2.500 kg/arm	0.830 m/arm	5	[26] [34]
DORA (Door Opening Robotic Arm)	UMass Lowell	2009	Joystick	13.063 kg	3	Gripper	1	N/A	1.245 m	7	[30] [31]
iARM (Assistive Robotic Manipulator)	Exact Dynamics	2009	Keypad, Joystick, or Single Button	9.000 kg	1	Gripper	5	1.500 kg	0.900 m	6	[14] [15]
WMRA (Wheelchair Mounted Robotic Arm)	University of Southern Florida	2008	Brain Signals	13.750 kg	2	Gripper	5	4.500 kg	1.082 m	7	[1]
Manus ARM (Assistive Robotic Manipulator)	Exact Dynamics	~1992	Touch Screen User Interface, Camera	14.300 kg	3	Gripper	5	1.500 kg	0.800 m	7	[40]

8 Methodology

The goal of this project is to utilize the igus® Robolink arm to drive a wheelchair, flip a light switch, and turn the page of a book. The team's approach to the project involved understanding the igus® arm, its capabilities, and constraints. Design solutions for making the arm mobile included mounting the arm on the wheelchair, interfacing with the wheelchair's power supply, investing in a joystick to control the chair, and designing a custom passive end effector for the Robolink.

8.1 Review of Previous Work

The igus® Robolink arm utilized for this project was configured prior to the team's work [39]. The team's first steps in this project involved reviewing and understanding the initial configuration of the cable box, and the documentation of the arm from igus®. The team was also provided with the forward and inverse kinematics of the arm, including the Jacobian matrix. The team reviewed the configuration of the 5 DOF arm and kinematics calculations. Code was provided to run the arm with an Xbox controller. The Xbox controller code was very useful throughout the project.

8.2 Proof of Concept Experiments

Several proof of concept experiments were conducted as the team brainstormed approaches to the design and task specifications of the project. The first task the team experimented with was utilizing the Robolink to move the joystick of a controller. The wheelchair did not have a joystick for the team to work with, so to conduct the test, the team utilized an Xbox controller and a GameCube controller. Since a passive end effector was required but not yet developed the team tied the "fingers" of the igus® gripper together seen below in Figure 6: Gripper with "Fingers" Tied.



Figure 6: Gripper with "Fingers" Tied

The team utilized the Xbox controller code to move the “fingers” over the different joysticks of the controllers to determine if the control was precise enough to move a joystick. The results showed that the arm could be used to control a joystick.

The second proof of concept experiment revolved around finding possible mounting positions for the Robolink on the wheelchair based on the workspace of the arm. Taking into account the design specification that the end effector needed to reach both the wheelchair’s joystick on the end of the right armrest and the desk attached to the left armrest, the team measured the workspace of the arm from different mounting configurations. The conclusion was that the most appropriate place to mount the arm was on the back of the wheelchair, with the arm coming over the right shoulder of the chair.

8.3 Design Constraints

8.3.1 Mounting the Robolink

The team faced many constraints in achieving the design goals. Mounting the cable box and arm on the wheelchair posed the issue of finding suitable space to mount the arm. Since assistive wheelchairs are already loaded with technology, available mounting space is limited. As well as being limited the mounting point needs to support the 10.3 kg weight of the robot arm. The arm also needs to be interfaced with the chair’s mobile power supply and laptop. As

explained in the proof of concept experiment above, the workspace of the arm with respect to both the joystick and desk needed to be accommodated.

8.3.2 Passive End Effector

Another important design constraint involved not utilizing the igus® gripper as the end effector. This decision was made in an attempt to save space by not mounting pneumatics to control it. A simpler solution was necessary, in the form of a passive end effector. This end effector needed to be modular, easy to remove and replace the existing gripper. This end effector also needed to be versatile enough to work for multiple tasks including joystick driving.

8.4 Design Solutions

8.4.1 Mounting Arm

The Robolink arm needed to be mounted such that the end effector could reach both the joystick at the end of the right armrest and the desk of the wheelchair on the left armrest. The preliminary experiment showed that the best place to mount the arm was the back of the chair, with the arm coming over the right shoulder of the chair. Possible mounting solutions involved attaching the cable box to the back of the chair using the headrest, back pocket of the chair, or the unused mounting fixture at the base of the chair seen below in Figure 7: Mounting Fixture.



Figure 7: Mounting Fixture

The first design version involved utilizing the mounting fixture and 1" steel square pipes. The pipes would be bent to an L-shape to attach to the mounting fixture. An L-shaped aluminum plate would be machined to attach to and support the cable box. For extra support, nylon mounting straps would attach the box to the headrest.

This first design was evaluated to be too complex. A second version of the design also utilized the empty mounting fixture at the base of the chair, but consisted of only 80/20 Aluminum parts. Several pieces of 1 in. 80/20 pipe and several brackets were utilized to screw directly into the back plate of the box. The box was mounted several inches off center such that the arm came directly over the right shoulder. The mounting fixture with the back plate of the box attached can be seen below in Figure 8: Mounting Structure.



Figure 8: Mounting Structure

8.4.2 Power Supply

The Robolink drive module takes 24V DC and at least 10A. The arm needed to be mobile, so it needed to interface with the wheelchair's power supply. Once an empty 24V Anderson Power Pole connector was located on the chair, about 5 ft. of wiring was run from the base of the chair to the power connector on top of the box to power the arm.

8.4.3 Controller

To achieve a mobile system, the igus® arm needed to run from a mobile computer. The first mobile computer the team attempted to utilize was a Raspberry Pi, shown below in Figure 9: Raspberry Pi. The Raspberry Pi was small and powerful, which would have been ideal for this application. Issues occurred with building the existing software against the EposManager library on the Raspberry Pi. The issue was that the only provided EPOS driver was available for x86 and not ARMel. EposManager is a necessary Robot Operating System (ROS) package for controlling the arm, making the Raspberry Pi unusable for this application.



Figure 9: Raspberry Pi⁵

The remaining options for controlling the Robolink were laptops. The team investigated purchasing a small laptop to control the system. The small laptop would have been ideal because it would have been lightweight and compact, while supporting the EposManager. This small laptop was ordered used and was found to have issues with the boot drive. The team determined that it was not worth it to devote any more time to it, and moved to the Lenovo ThinkPad laptop

⁵ http://images.bit-tech.net/content_images/2013/03/raspberry-pi-case-competition-update/pi11.jpg

provided by WPI. A ThinkPad labeled *Amazon* was the computer utilized for controlling the wheelchair. The team was assigned a ThinkPad labeled *Nile*. The *Nile* laptop's software installation was 32-bit, which is required to run the EposManager, while the *Amazon* laptop's existing software installation was 64-bit, and could not be used for the Robolink.

8.4.4 Wheelchair Joystick

The highest priority task involved driving the joystick of the wheelchair with the igus® arm. The biggest challenge to this accomplishment was the lack of a joystick to work with, since the original joystick had been disassembled and determined inadequate for the application. The team needed to purchase a joystick with both directional control and a button to be used as an emergency stop. Options included game system controllers, such as Xbox, GameCube, or Wii. Proof of concept experiments showed promise with the GameCube controller's grey joystick.

These game controllers were all potentially clumsy to mount on the chair, however, so arcade joysticks were explored next. A simple arcade joystick held much potential; it was cost efficient and the proper size to mount on the wheelchair. It can be seen in Figure 10: Simple Arcade Joystick. Three concerns were raised with this option, though. The ball on the end of the joystick would be difficult to push around, and the joystick was digital not analog, which was necessary to control the speed of the wheelchair, and there were not buttons available.



Figure 10: Simple Arcade Joystick⁶

⁶ <https://www.sparkfun.com/products/9182>

The team then researched industrial joysticks, which were deemed to provide high compatibility with the team's vision for an end effector. They were an ideal size for being manipulated. They provided analog control, which was necessary for fine tuned wheelchair control, and some were USB compatible, which would make working with Robot Operating System (ROS) straightforward. The main downfall of the industrial joysticks was cost. A CTI Electronics industrial joystick was ordered for the wheelchair. It can be seen in Figure 11: CTI Electronics Industrial Joystick.



Figure 11: CTI Electronics Industrial Joystick⁷

The team soon discovered that an emergency stop button would be necessary. A McMaster-Carr turn-to-reset Emergency Stop button was ordered and mounted on the back of the cable box to cut power to the Robolink, shown in Figure 12: Emergency Stop Button.

⁷ <http://www.ctielectronics.com/OEM-Analog-Industrial-Joysticks/Inductive-Industrial-Joystick.php#N24>



Figure 12: Emergency Stop Button

8.4.5 End Effector

One goal of the project was to utilize a passive end effector, as mentioned in the Constraints section above. A Robotics Engineering Undergraduate student volunteered to assist the team in designing a passive end effector for the arm. While Walker's project was under development, the team tied the "fingers" of the igus® gripper together to test and prove concepts.

Walker's end effector was tailored to work with the CTI Electronics joystick and made to modularly fit the end effector attachment of the Robolink. It consisted of a 1 in. deep cup that fit over both the joystick and a light switch. The custom end effector was 3D printed. Photos of the custom end effector can be seen below in Figure 12: End Effector and Figure 14: End Effector Over Joystick.



Figure 13: End Effector



Figure 14: End Effector Over Joystick

8.5 Completion of Tasks

Two distinct pieces of code have been written for this project. The first one interfaces the joystick with the wheelchair's control systems. The other controls the arm mainly to drive the joystick and perform other tasks.

8.5.1 Joystick Code

The joystick control code converts the raw joystick input data from the axes into something that ROS can comprehend. It subclasses some of the methods and overrides others that were written for Xbox controller. Some issues with the joystick itself were overcome in the code. When the joystick nears the edges of its range (in the extremes), all the values go to zero. To counteract this, a weighted rolling average function was written to check against the last few values that the joystick published and output the maximum needed value if the present joystick

values are determined to be in the extremes. The function that returns and updates the rolling average can be seen below in Table 2: Returns and Updates Rolling Average.

Table 2: Returns and Updates Rolling Average

```
def _ij_x_avg(self):  
    # this function returns the rolling weighted average for  
    # the X values of the joystick value timeline  
    return average(self.x_timeline,weights=WEIGHTS)  
  
def _ij_y_avg(self):  
    # this function returns the rolling weighted average for  
    #the Y values of the joystick value timeline  
    return average(self.y_timeline,weights=WEIGHTS)  
  
def _ij_x_update(self,val):  
    # This function updates the Value Timeline of the X values  
    #by removing the Last one and adding the Latest as the first  
    self.x_timeline.pop()  
    self.x_timeline.insert(0,val)  
  
def _ij_y_update(self,val):  
    # This function updates the Value Timeline of the Y values  
    # by removing the Last one and adding the Latest as the first  
    self.y_timeline.pop()  
    self.y_timeline.insert(0,val)
```

The logic for determining the wheelchair's speed based on the joystick values can be seen below in Table 3: Logic for Determining Wheelchair Speed.

Table 3: Logic for Determining Wheelchair Speed

```
if ij_x > POS_MIN_VAR and ij_x < POS_MAX_VAR:  
    # if the value of x isn't in the joystick extremes  
    if ij_x not in extremes:  
        # convert the joystick value to an integer  
        x_map = convert_joy_value_to_int(1.0-ij_x)  
        # update the rolling average  
        ij0._ij_x_update(x_map)
```



```

elif ij_x < NEG_MIN_VAR and ij_x > NEG_MAX_VAR:
    # if the value of x isn't in the joystick extremes part 2
    if ij_x not in extremes:
        # convert the joystick value to something usable
        x_map = convert_joy_value_to_int(-1.0-ij_x)
        # update the rolling average
        ij0._ij_x_update(x_map)

    else:
        # we are in the extremes and it's terrible here
        # so we get the current rolling average
        a = ij0._ij_x_avg()
        print a
        # if the rolling average is above the rolling floor we set it to our max mapped
value
        # if the rolling average is under the rolling floor we set it to our negative max
mapped value
        if a > ROLLING_FLOOR:
            x_map = ROLLING_MAP
        elif a < -ROLLING_FLOOR:
            x_map = -ROLLING_MAP/2

```

8.5.2 Robolink Controller

8.5.2.1 Command Message

The Robolink control code is composed of two classes that speak to existing Robolink driver functionality. The first class is the CommandMessage class, which is initialized with the following arguments:

- *'positions'*: a list of position values
- *'timeinterval'*: a list of time intervals
- *'controller'*: a controller to modify
 - defaulting to the standard 'RobolinkControl' class
- *'reset'*: whether or not to expose a reset parameter
 - defaults to True
- *'steps'*: a list of arbitrary steps to execute
- *'backsteps'*: a list of steps to execute when being run in reverse

- *'startingstep'*: a starting step
 - defaults to the first given
- *'method'*: which base method to modify
 - defaults to pose
 - can be set to None
- *'method_sub_map'*: what properties of the method to modify the values of *'positions'*
 - defaults to the ['position.x','position.y','position.z'] objects
- *'control_mode'*: which control mode to use with EPOS
 - defaulting to pose control
- *'iterative_control'*: whether or not to use iterative control
 - defaults to False
- *'DRO'*: a list of command names not to reset on

The only required argument is the list of positions. It exposes five methods, most importantly those to run the steps, run them forward, and to run through them backward. It also supports executing any arbitrary position that was given to the Command Message either regularly or iteratively

8.5.2.2 Command Acceptor

The second class is the Robolink Command Acceptor, which opens a ROS subscriber to wait for and execute commands. It is initialized with a dictionary of CommandMessage instances. Upon initialization it returns a list of available commands that it is listening for on the subscriber. Upon receiving a command via a publisher it proceeds to run it until completion.

8.5.2.3 Example of Use

A command can be simple or override default behavior as needed. Two examples are shown in Table 4: Example of Use with the command acceptor being initialized after defining the CommandMessages. The first defines a set of points in space to iterate to. The second defines a set of angles that the arm should replay.

Table 4: Example of Use

```
cmddict ={
    'simple': CommandMessage(
        positions=[(0,0.05,0),(0,.1,0)],
```

```

        reset=False
    ),
    'morecomplex': CommandMessage(
        positions = [(-8,5,0,89,0),(-8,4,0,90,0),(-8,2,0,88,0),(-8,0,0,87,0),(-8,-1,0,85,0)],
        method=None,
        reset=True,
        method_sub_map=METHOD_SUB_MAP_JOINTS,
        control_mode=JOINT_VELOCITY,
        DRO=['drive_stop','drive_fwd','drive_back'],
        backsteps=[0,1,2]
    ),
}

cmdacceptor = CommandAcceptor(cmddict)

```

There are a large number of default settings that can be overridden in order to perform tasks beyond the default scope.

9 Results & Discussion

9.1 Mounting Arm

The arm mounting structure consisting of 80/20 Aluminum pipes and brackets worked well for this application. The structure was stable and strong, both in the sense that it did not wiggle or move and that it attached to a very stable fixture on the wheelchair. The design was also simple; it took very few pieces, did not require any custom work, and was easy to put together. This design also achieved modularity, as its attachment to the fixture was straightforward, and easy to put on or remove. The workspace of the Robolink was also very efficient from this mounting position – the desk and joystick could both be reached, as well as a range of vertical and horizontal locations, shown in Figure 15: Workspace of Arm – Vertical, Figure 16: Workspace of Arm – Joystick Driving, Figure 17: Workspace of Arm – Horizontal, and Figure 18: Workspace of Arm – Over Desk.



Figure 15: Workspace of Arm – Vertical

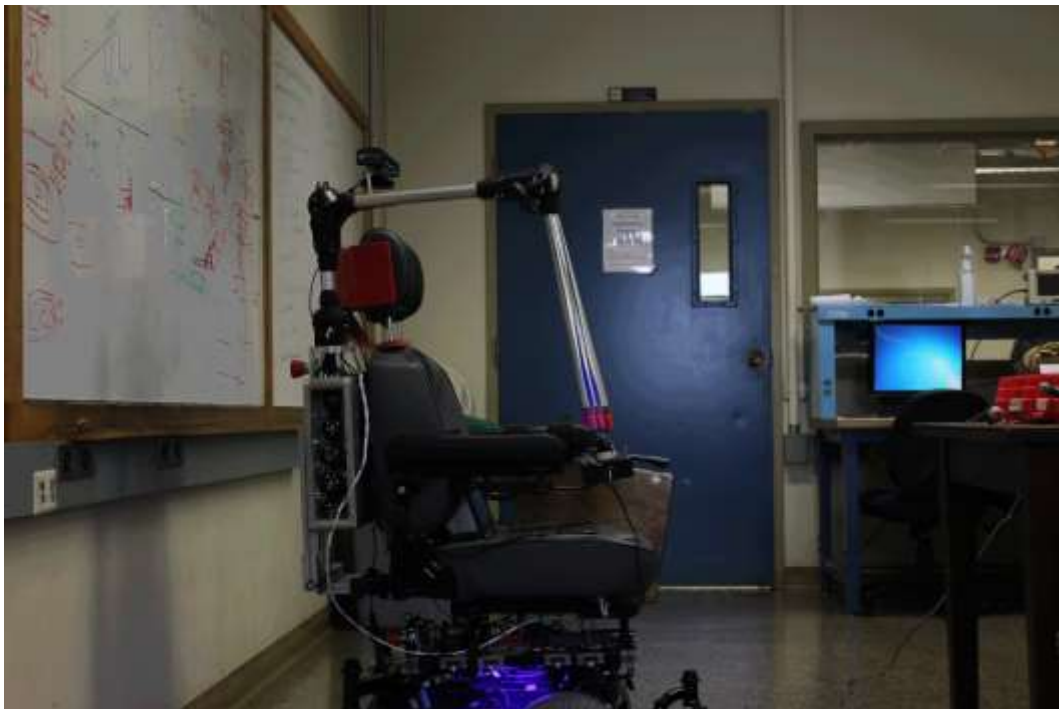


Figure 16: Workspace of Arm - Joystick Driving



Figure 17: Workspace of Arm – Horizontal



Figure 18: Workspace of Arm - Over Desk

9.2 Power Supply

Interfacing with the wheelchair's power supply worked well. While it was difficult to locate an open 24V DC rail with no knowledge of the wheelchair's electronics, once it was found the interfacing was simple. This interfacing also kept within the modular spirit of the project, and an Anderson Power Connector was the only necessary components, and was easy to connect and disconnect from the system.

9.3 Controller

Overall, the Lenovo ThinkPad laptops worked well for this application. Working with both *Amazon* and *Nile* on the mobile wheelchair was clumsy, but given that the EposManager could not work on *Amazon*, it was necessary to use both laptops.

9.4 Wheelchair Joystick

The CTI Electronics joystick was definitely the right tool for the job, despite the cost. It was USB compatible, which worked extremely well for this application. The code developed worked well for wheelchair driving, and can be run simply on the *Amazon* laptop.

9.5 End Effector

The custom 3D printed end effector was optimal for use with the CTI joystick. It kept within the modular theme since it was easy to attach and remove, and therefore worked well with the project. The end effector was passive, not adding bulk to the system and not drawing any power. It also served as an experiment to illustrate the usefulness of passive end effectors. Though an immobile person would not be able to change out end effectors on their own, they can still be useful.

9.6 Completion of Tasks & Experimental Results

9.6.1 Drive Joystick

The highest priority goal for this project involved driving a joystick mounted on the wheelchair. While driving using the arm is achievable by executing the control movements manually, there are a number of issues that the team came across that stand in the way of automating it. To automate the wheelchair driving functions, the team recorded a series of angles to play back and fed them into a `CommandMessage`. Testing yielded that telling the arm to go

back and forth between two points in space would never yield the exact same result. The arm would move to approximately the same point in space but up to a half inch away from its previous location. In an attempt to improve on the repeatability the team ensured that the drive cables were did not have slack and were appropriately tightened. This in turn lowered the margin of error but not enough to be satisfactorily repeatable due to another issue the team noticed; when the power for the arm was cycled, the repeatability of tasks suffered due to the fact that the arm drooped slightly then took the new position on power up as its default position. The team attempted to work around this by accounting for the droop before power down so that the arm would be in its default position even with the droop occurring. However, this wasn't ideal for a deployed setup with an actual person occupying the chair.

The team also had to work around the fact that moving to a position outside of the driving range while the end effector was still over the joystick would force the end effector to be caught on the joystick and put massive strain on the arm as it attempted to move. This was mitigated by adding a flag to the command message and having the arm move to a safe point above the joystick before continuing on to the next task.

Part of the issue with repeatability was the feedback mechanism used by the command acceptor. The Robolink arm feedback topic returned integers for angles instead of other number types. When the end of arm was over forty-five centimeters away from the rotating joint, the resolution for a degree is just under a centimeter, which was not accurate enough.

Circumference of end effector tube:

$$48.6 \text{ cm} * 2 * \pi = 305.3 \text{ cm}$$

Margin of error:

$$305.3 \text{ cm} / 360 \text{ deg} = 0.84 \text{ cm/deg}$$

9.6.2 Flip Light Switch

The second priority goal of this project involved flipping a standard height light switch. This task proved more complex than initial thoughts portrayed. An incident occurred during an experiment when the Robolink was mounted on the chair and run utilizing the Xbox control code rather than the Command Message/Command Acceptor code. The Xbox control code crashed but the motors continued to run. A team member was sitting in the wheelchair at the time, and the arm drove into the team member's leg. Though not enough force was put on the team

member to cause pain or injury, a motor axle for one of the degrees of freedom bent and needed replacement. This incident provided valuable information to the team; the arm was potentially good for an immobile person, as it would break before causing injury, and the force at the end effector was very low.

This made the team wonder if there would be enough force to flip a light switch. To measure the force to flip a light switch, the team utilized a force gauge, hooking the end over the light switch. The team had access to two switches in Atwater Kent; old switches which required a force of 3 lbf to flip, and new switches which required a force of 2 lbf to flip. Flipping a switch on or off took the same amount of force.

The team conducted an initial experiment with the Robolink arm on the wheelchair utilizing the Xbox controller code. The team first placed the wheelchair directly facing a standard height light switch. The first problem presented itself; the arm was not long enough to reach the wall in front of the chair with the footrest of the chair in its down position, seen below in Figure 19: Light Switch With Chair Facing Wall.



Figure 19: Light Switch With Chair Facing Wall

The team corrected this by moving the chair at an approximate 45-degree angle with respect to the wall, and utilizing the rotational DOF at Joint 0 of the arm to place the end effector near the light switch seen in Figure 20: Light Switch with Chair at 45-Degree Angle. The arm was then moved to a position such that Joint 2 was at an approximate 150-degree angle with the end effector on the light switch. This was done in hopes that the weight of the arm would assist in forcing the switch down. Both joints were driven to push downward on the switch simultaneously.



Figure 20: Light Switch With Chair at 45-Degree Angle

This resulted in the end effector simply staying on the light switch. When the wheelchair was pulled back from the wall, the end effector was approximately 7 in. lower than the light switch on the wall. This shows that the arm was attempting to force the switch down but was not exerting enough force to flip the switch. The result was the same for both the old and new switches. The team was concerned, as a repeat of the above incident was not desirable and could potentially result in another broken motor axle.

This prompted the team to perform a calculation of force exerted by the end effector assuming the arm was pointed directly upward. The nominal torque for the motor was multiplied by the gear ratio to get the output torque from the motors. This was then divided by the radius of the motor wheels to find the force exerted after the motor wheels, F_1 . This force was then multiplied by the radius of the rope in the joint divided by the link length to get the force at the end of the joint, F_2 . This process was repeated until the force at the end effector was found.

$$Torque_{Motor} = 82.7mNm = 0.0827Nm$$

$$Gear\ Ratio_{Motor} = 381$$

$$Diameter_{MotorWheel} = 80mm$$

$$Diameter_{Rope\ in\ Joint} = 50mm$$

$$Link_1 = 285.75mm$$

$$Link_2 = 450.85mm$$

$$Link_3 = 450.85mm$$

From motor:

$$Torque_{Motor} * Gear\ Ratio_{Motor} = T_{out}$$

$$0.0827Nm * 318 = 26.2986Nm$$

$$T_{out} / Radius_{Motor\ Wheel} = F_1$$

$$26.298Nm / 0.04m = 657.45N = 147.8lbf$$

End of Link 1:

$$F_1 * (Radius_{Rope\ in\ Joint} / Link_1) = F_2$$

$$657.45N * (25mm / 285.75mm) = 57.52N = 12.931lbf$$

End of Link 2:

$$F_2 * (Radius_{Rope\ in\ Joint} / Link_2) = F_3$$

$$57.72N * (25mm / 450.85mm) = 3.201N = 0.7196lbf$$

End of Link 3 (End Effector):

$$F_3 * (\text{Radius}_{\text{Rope in Joint}} / \text{Link}_3) = F_{\text{End Effector}}$$

$$3.201N * (25\text{mm} / 450.85\text{mm}) = 0.1775N = 0.0399\text{lb}_f$$

Though this calculation only accounted for one position and was therefore not entirely accurate, it showed the minimal force the end effector could exert. This made sense, as it was not enough force to hurt a person in the chair or flip a light switch, but it was enough to move the high quality joystick. The Robolink utilized for this project used small Maxon Motors to make the system more mobile, but were not as strong as the stepper motors igus® generally uses for the Robolink.

The team performed calculations with the Jacobian matrix for the Robolink system to prove that the motors could not provide enough torque in multiple configurations. The force/torque relation, $\tau = J^T F$, where τ is a vector of joint torques and F is a vector of the forces and moments on the end effector, was utilized for these calculations. The team assumed no force in the y-plane and that the moments were considered negligible. The forces in the x and z planes were calculated using trigonometry and can be seen in Figure 21: Forces on a Light Switch.

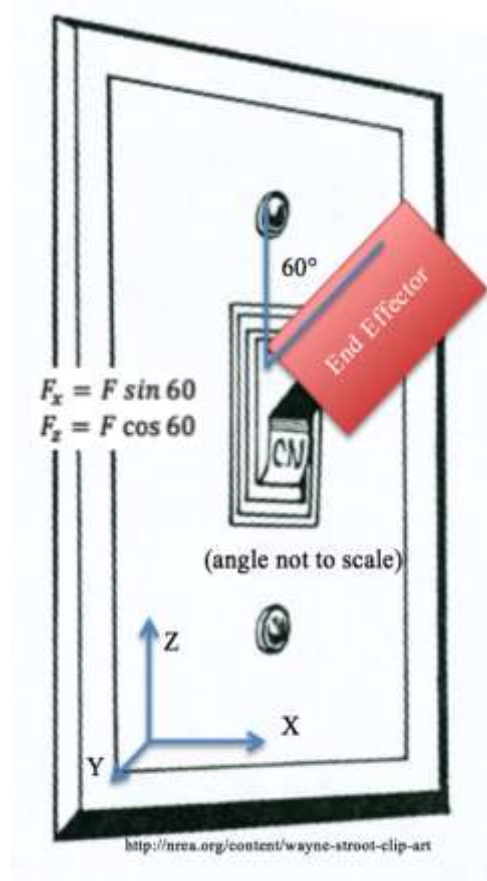


Figure 21: Forces on a Light Switch

$$\tau = J^T F$$

Where:

τ – corresponding vector of joint torques

F – vector of forces and moments externally acting on the end – effector

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \\ \tau_5 \end{bmatrix} \quad \tau_{nominal} = \begin{bmatrix} 0.0827Nm \\ 0.0827Nm \\ 0.0827Nm \\ 0.0827Nm \\ 0.0827Nm \end{bmatrix} \quad \tau_{stall} = \begin{bmatrix} 0.782Nm \\ 0.782Nm \\ 0.782Nm \\ 0.782Nm \\ 0.782Nm \end{bmatrix}$$

$$F = \begin{bmatrix} F_x \\ F_y \\ F_z \\ n_x \\ n_y \\ n_z \end{bmatrix} \quad F = \begin{bmatrix} 3 \sin 60 \text{ lbf} \\ 0 \\ 3 \cos 60 \text{ lbf} \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad F = \begin{bmatrix} 13.344 \sin 60 \text{ N} \\ 0 \\ 13.344 \cos 60 \text{ N} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

For sets of joint angles, the output torque was compared to the normal torque and stall torque for the motors. For the configuration described above, with the arm at a 45-degree angle to the wall Table 5: Configuration 1 Necessary Force was generated, where i , the angle between Link 1 and Link 2, was in range of 40-70 degrees, and j , the angle between Link 2 and Link 3, was in range of 120-150 degrees.

Table 5: Configuration 1 Necessary Force

Configuration: [45, i , 0, j , 0]	2 lbf Force	3 lbf Force
Nominal Torque (0.0827 Nm)	None	None
Stall Torque (0.782 Nm)	None	None

This data illustrated that the team’s suspicions were true; no combination of joint angles would reach and flip the light switch in this configuration. The team decided to work with another arm configuration, seen in Table 6: Configuration 2 Necessary Force, where k , the

rotation from the shoulder, was in range of 60-120 degrees, seen below in Figure 22: Light Switch With Chair at 90-Degree Angle.



Figure 22: Light Switch With Chair at 90-Degree Angle

Table 6: Configuration 2 Necessary Force

Configuration: [0, 90, k , 90, 0]	2 lbf Force	3 lbf Force
Nominal Torque (0.0827 Nm)	for $0 < k < 61$ deg	for $0 < k < 60$ deg
Stall Torque (0.782 Nm)	for $0 < k < 71$ deg	for $0 < k < 67$ deg

This configuration was much more promising than the first, as it utilized the motion of a stronger joint than the end effector. In the case of a standard height light switch, this

configuration would still not work, since k would need to be much closer to 90 degrees. The arm could potentially have enough force to flip a lower light switch.

The team suspects that the Robolink would be capable of flipping other kinds of light switches, such as rocker switches like the one shown below in Figure 23: Rocker Light Switch.



Figure 23: Rocker Light Switch⁸

9.6.3 Turn Page of Book

Due to unforeseen delays in the project, the team was unable to thoroughly experiment with turning the pages of a book. As a brief experiment, a laptop was placed on the wheelchair's desk, seen in Figure 24: Robolink Typing on Laptop. The laptop was well within the workspace of the Robolink, and the end effector of the Robolink could exert enough force to type on the keys of the laptop. The team speculates that the Robolink would have been capable of turning the pages of a book, given a rubberized end effector. This would have been a useful task to perform, as it would majorly improve the quality of life of a person with no mobility.

⁸ <http://www.homecontrols.com/Leviton-LevNet-Self-Powered-Wireless-Light-Switch-Single-Rocker-LVWSS0SD0x>



Figure 24: Robolink Typing on Laptop

10 Conclusion

10.1 Goal Completion

10.1.1 Task Specifications

Some of the tasks specifications for this project were met. The arm was able to safely drive the wheelchair using the industrial joystick with its custom end effector even with caveats. However, the arm was unable to flip a standard height light switch due to the lack of necessary force at the tip. The team was unable to test the opening of a book on the desk, however the team tested the viability of the workspace that would ultimately be used to open the book and the reach of the arm in that workspace.

10.1.2 Design Specifications

The design specifications for this project were all met. The arm needed to be mounted on the wheelchair. The 80/20 Aluminum structure fulfilled this requirement while keeping in the modular spirit of the project. The whole Robolink system needed to be mobile on the wheelchair. This was achieved by interfacing the arm with the wheelchair's power supply and utilizing laptops to control the system. A passive end effector was 3D printed to interact with the joystick.

10.2 Societal Considerations

Through the course of this project, the team needed to address the societal concerns revolving around wheelchair mounted robotic arms. These concerns involve ethics, safety, and aesthetics of the project.

The ethical concerns of this project involve the safety of the occupant of this wheelchair. The team designed the mounting structure for the arm such that the arm's workspace remains out of the way of the occupant. In addition, it is possible to add software limitations to prevent the arm from entering specific areas of the workspace, adding an additional level of safety. As well as avoiding the occupant, the end effector of the arm can exert very little force (about ½ lbf). This means that the arm will break before it is able to injure an occupant of the wheelchair in case of software failure. An emergency stop button has also been added to the cable box of the arm that cuts power to all motors controlling the arm. Though the team could not find regulations on wheelchair velocity or acceleration, the code that controls the wheelchair's joystick was designed to avoid any sudden acceleration and deceleration in order to provide a smooth ride.



Figure 25: 80/20 Extruded Aluminum

The simple mounting design of the arm conforms to the design qualities of the existing hardware and is pleasing to the eye. The design consists of non-abrasive materials, such as the 80/20 Aluminum shown in Figure 25: 80/20 Extruded Aluminum, with smoothed edges to improve the quality and safety of the arm. All wiring is away from the occupant and moving parts of the chair, keeping the look clean and safe. The passive end effector designed for this project is also pleasing to the eye, made of 3D printed red plastic (which can be recycled), matching the red headrest of the wheelchair.

10.3 Future Recommendations

The team would like to leave suggestions for future work on the Robolink project. The most common problem the team encountered was a lack of sufficient motor bracing. The design of the cable box did not support the cable wheels from both sides, leading to issues with bent motor axles. The cables were very strong, so in a situation where a degree of freedom was forced beyond its limit, the axle of the motor for that DOF would bend. The team recommends finding some design solution to this weakness for future work on the project.

Another recommendation involved adding accuracy to the control system for the arm. Repeatability and precision were difficult with the control code given by to the team. Joint angles were integers of degrees, and were not repeatedly reached between power cycles. The team recommends looking deeper into the control software, and potentially adding a feedback loop to add more precision to the system.

An alternative to editing the control software could involve different passive end effectors. Since the code cannot provide the precision required for the 3D printed end effector, an end effector of a more malleable material could work well, as seen below in Figure 26: Malleable End Effector.



Figure 26: Malleable End Effector⁹

⁹ <http://www.bastiansolutions.com/blog/index.php/2013/06/12/innovative-robotic-end-of-arm-tool/#.UhzpIWRoQx8>

11 Bibliography

1. Alqasemi, Redwan M., and Rajiv V. Dubey. "Chapter 3: A 9-DoF Wheelchair-Mounted Robotic Arm System: Design, Control, Brain-Computer Interfacing, and Testing." *Advances in Robot Manipulators*. Ed. Ernest Hall., 2010. <http://www.Intechopen.com/books/advances-in-Robot-manipulators/a-9-Dof-Wheelchair-Mounted-Robotic-Arm-System-Design-Control-Brain-Computer-Interfacing-and-Testing> Intech. Web. 2013.
2. Alqasemi, Redwan M., et al. "Analysis, Evaluation and Development of Wheelchair-Mounted Robotic Arms." www.smpp.northwestern.edu. July 1 2005.Web. <http://www.smpp.northwestern.edu/ICORR2005/proceedingsCD/pdf/files/papers/ThP01_33.pdf>.
3. Chung, Cheng-Shiu MS, and Rory A. PhD Cooper. "Literature Review of Wheelchair-Mounted Robotic Manipulation: User Interface and End-User Evaluation". *RESNA Annual Conference - 2012*. 2012.<http://resna.org/conference/proceedings/2012/StudentScientific/Robotics/LITERATUREREVIEWOFWHEELCHAIR-MOUNTEDROBOTICMANIPULATION.html>. Print.
4. Deyle, Travis. "Telescoping Robot Arm for Wheelchair - Mounted Object Retrieval." www.hizook.com. Aug 16 2010.Web. <<http://www.hizook.com/blog/2010/08/16/telescoping-robot-arm-wheelchair-mounted-object-retrieval>>.
5. dfkvideo. "Entwicklung eines Antriebskonzepts für das Robotergelenk Robolink." www.youtube.com. Ed. Praxissemester-Arbeit am Robotics Innovation Center des DFKI. Sep 23 2010.Web. <<http://www.youtube.com/watch?v=4ajf4Ob-BGk&list=PL07B939DF7512B3B4>>.
6. E. Chaves, B. S., et al. "Clinical Evaluation of a Wheelchair Mounted Robotic Arm". *Technology & Disability: Research, Design, Practice & Policy*. June 19, Atlanta, GA. Web.

7. Evenchick, Eric. "Toorcamp: MC Hawking Robotic Wheelchair." *hackaday.com*. Aug 14, 2012 2012.Web. <<http://hackaday.com/2012/08/14/toorcamp-mc-hawking-robotic-wheelchair/>>.
8. Fox, Stuart. "Robotic Arm Opens Dorrs for the Wheelchair-Bound." *www.popsoci.com*. Nov 24 2009.Web. <<http://www.popsoci.com/technology/article/2009-11/robotic-arm-opens-doors-wheelchair-bound>>.
9. Gray, Jeremy. "Trussville students build robotic arm to aid quadriplegics (video)." *blog.al.com*. April 23, 2013 2013.Web. <http://blog.al.com/spotnews/2013/04/trussville_students_build_robo.html>.
10. Gutierrez, Jon. "Wheelchair-Mounted Robotic Arm Is A Total Snoozefest." *www.uproxx.com*. Sept 27 2010.Web. <<http://www.uproxx.com/gammasquad/2010/09/wheelchair-mounted-robotic-arm-is-a-total-snoozefest/>>.
11. hairy8. "Robotic Arm for Wheelchair - 2010 SJSU Senior Project Presentation (1/2)." *www.youtube.com*. Ed. Aimy Huynh, et al. May 21, 2010 2010.Web. <http://www.youtube.com/watch?v=_ZqYBd9rTJA>.
12. Hillman, Michael, et al. "A Wheelchair Mounted Assistive Robot". *International Conference on Rehabilitation Robotics*. Stanford, CA. ICORR , 1999. <http://rehabrobotics.org/icorr1999/papers/papers/hillman.pdf>. Print.
13. Hollister, Sean. "Lightweight robot arm connects to your wheelchair, stoops in your stead (video_." *www.engadget.com*. Aug 1 2010.Web. <<http://www.engadget.com/2010/08/01/lightweight-robot-arm-connects-to-your-wheelchair-stoops-in-you/>>.
14. "iArm." *www.exactdynamics.com*. Web. <<http://www.exactdynamics.nl/site/?page=iarm>>.
15. "iARM Product Overview." *iarmrobot.com*. 2012.Web. <<http://iarmrobot.com/overview.shtml>>.
16. "igus® Robolink." *www.igus.com*. Web. <<http://www.igus.com/wpck/default.aspx?pagename=roboLink>>.

17. igus GmbH. "igus® robolink® robot modules applications and references."
www.youtube.com. Web.
<<http://www.youtube.com/playlist?list=PL07B939DF7512B3B4>>.
18. "Jaco | Rehab Edition." www.kinovarobotics.com. 2012. Web.
<<http://kinovarobotics.com/products/jaco-rehab-edition/>>.
19. Kaye, H. S., Kang, T. and LaPlante, M.P. (2000). Mobility Device Use in the United States. Disability Statistics Report, (14). Washington, D.C.: U.S. Department of Education, National Institute on Disability and Rehabilitation Research.
20. "Kinova JACO™ Robot Manipulator (Research Edition)." www.robotshop.com. 2013. Web. <<http://www.robotshop.com/kinova-technology-jaco-robot-manipulator-academic-edition-2.html>>.
21. LIRECWRUT. "FLASH Mobile Dexterous Social Robot, LIREC, WRUT." www.youtube.com. Mar 6 2012. Web. University of Technology, Poland.
<<http://www.youtube.com/watch?v=o8kf14SZkaQ&list=PL07B939DF7512B3B4>>.
22. Malte. "Humanoid robot A1: robolink arms." www.mtahlers.de. July 19 2010. Web.
<<http://www.mtahlers.de/index.php/robotik/humanoider-torso/robolink-arme>>.
23. Natasha. "Robotic Arm Developed By Columbia University, Can Be Controlled By Facial Muscles." webmuch.com. May 2013. Web. <<http://webmuch.com/robotic-arm-developed-by-columbia-university-can-be-controlled-by-facial-muscles/>>.
24. Neergaard, Lauran. "Paralyzed man uses brain-powered robot arm to touch." www.nbcnews.com. Oct 10 2011. Web.
<http://www.nbcnews.com/id/44843896/ns/health-mens_health/t/paralyzed-man-uses-brain-powered-robot-arm-touch/#.UZz1gopDuKk>.
25. Ostrovsky, Gene. "DORA Robot Arm a New Accessory for Wheelchairs." www.medgadget.com. Nov 24 2009. Web.
<http://www.medgadget.com/2009/11/dora_robot_arm_a_new_accessory_for_wheelchairs.html>.
26. "PerMMA - Personal Mobility & Manipulation Appliance." www.cmu.edu. Web.
<<http://www.cmu.edu/qolt/AboutQoLTCenter/PressRoom/ces-2012/permma.html>>.

27. Polito, Joe. "This Week in Research: Wheelchair-mounted robotic arm." www.usforacle.com. April 4 2011.Web. <http://www.usforacle.com/this-week-in-research-wheelchair-mounted-robotic-arm-1.2530642#.Uhe_7WRoQx9>.
28. puzzlebox. "Brain-Controlled Wheelchair Robot." www.instructables.com. Sept 3, 2012 2012.Web. <<http://www.instructables.com/id/Brain-Controlled-Wheelchair-Robot/>>.
29. Raak, Martin. "roboLink® Joint kit." robot.wpi.edu. Jan 2013.Web. <http://robot.wpi.edu/wiki/images/7/7d/RoboLink_Documentation.pdf>.
30. Rapacki, Erin B., Christopher Niezrecki, and Holly A. Yanco. "An Underactuated Gripper to Unlatch Door Knobs and Handles." robotics.cs.uml.edu. 2009.Web. <http://robotics.cs.uml.edu/fileadmin/content/publications/2009/tepraFINAL_id333_rapacki_v2.pdf>.
31. Rapacki, Erin. "The Design of an Underactuated Wheelchair-Mounted Robotic Arm to Unlatch Door Knobs and Handles." 2009.Web. <<http://users.wpi.edu/~nems/Rapacki.pdf>>.
32. "Researchers Develop "Brain-Controlled" Wheelchair Robotic Arm." news.usf.edu. Jan 28 2009.Web. <<http://news.usf.edu/article/templates/?a=1130>>.
33. "Researchers Develop "Brain-Controlled" Wheelchair Robotic Arm." www.singularityhub.com. Feb 11 2009.Web. <<http://singularityhub.com/2009/02/11/researchers-develop-brain-controlled-wheelchair-robotic-arm/>>.
34. "Robotic Arm's Big Flaw: Patients in Wheelchairs Say It's 'Too Easy'." www.sciencedaily.com. Sept 24 2010.Web. <<http://www.sciencedaily.com/releases/2010/09/100923184453.htm>>.
35. "Robotic system may put more tasks within wheelchair users' reach." www.research.va.gov. Feb 3 2010.Web. <http://www.research.va.gov/news/research_highlights/robotics-020310.cfm#.UhvnXmRoQx9>.
36. Santos, Alexis. "Columbia University's low-cost robotic arm is controlled by facial muscles, we go face-on (video)." www.engadget.com. May 4, 2013 2013.Web. <<http://www.engadget.com/2013/05/04/columbia-university-robotic-arm-controlled-by-facial-muscles-hands-on-video/>>.

37. Schwartz, Ariel. "The Jaco Robotic Arm Can Scratch Your Back, Hands Out Drinks." *www.fastcompany.com*. Jan 23 2011.Web.
<<http://www.fastcompany.com/1720283/jaco-robotic-arm-can-scratch-your-back-hand-out-drinks>>.
38. Sherer, Kyle. "Wheelchair-mounted robotic arm controlled by thought alone." *www.gizmag.com*. Feb 10 2009.Web. <<http://www.gizmag.com/brain-control-wheelchair-robotic-arm/10954/>>.
39. Tremblay, Ty and Taskin Padir. Modular Robot Arm Design for Physical Human-Robot Interaction. Systems, Man and Cybernetics (SMC), 2013 International Conference on. Manchester, UK.
40. Tsuia, Katherine M., et al. "'I Want That': Human-in-the-Loop Control of a Wheelchair-Mounted Robotic Arm." *robotics.cs.uml.edu*. 2011.Web.
<<http://robotics.cs.uml.edu/fileadmin/content/publications/2011/2011-JAAB-tsui-et-al-final.pdf>>.
41. TullyFoote. "Documentation - ROS Wiki." *www.ros.org*. July 18, 2013 2013.Web.
<<http://www.ros.org/wiki/>>.
42. Vieru, Tudor. "Robotic Arms Improve Life for Wheelchair Users." *news.softpedia.com*. Nov 26 2009.Web.
<<http://news.softpedia.com/news/Robotic-Arms-Improve-Life-for-Wheelchair-Users-127981.shtml>>.
43. "Visual Control Interface for Manus ARM." *robotics.cs.uml.edu*.Web.
<<http://robotics.cs.uml.edu/research/current-research/visual-control-interface-for-manus-arm/>>.
44. Wang, Wei, and Yuki Suga. "Human in Loop Integration of an Arm Mounted Wheelchair Robot Based on RT Middleware." 2010.Web.
<<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5756822&tag=1>>.
45. Wang, Hongwu, et al. "Performance Evaluation of the Personal Mobility and Manipulation Appliance (PerMMA)." *Medical engineering & physics*.0 Web.
46. "Wheel chair robot project." *www.sugano.mech.waseda.ac.jp*.Web.
<<http://www.sugano.mech.waseda.ac.jp/project/wheelchair/index.html>>.

47. "Woman Users Her Thought to Control Robot to Feed Chocolate - Amazing New Technology news." www.youtube.com. Dec 17 2012.Web.
<http://www.youtube.com/watch?v=C7H_M8-dBHc>.
48. 広報部 産総研. "ロボットアーム RAPUDA – Robotic Arm for Persons with Upper-limb DisAbilities." www.youtube.com. July 27 2010.Web.
<http://www.youtube.com/watch?feature=player_embedded&v=yqOUexGKusg#!>.

12 Appendix

12.1.1 Instructions for Use

Mounting & Startup:

- To mount the arm simply slide the 80/20 square pipes into the mounting fixture at the back of the CPS chair. The masking tape should line up with the fixture
- Plugging in (Note: BLUE is PWR; WHITE is GND):
 - Attach Anderson Pole Connector to open 24V rail on CPS chair
 - Attach the Emergency Stop Button to the open dual lock tape on cable box
 - Attach TE Circular Connector
- Turn on CPS and Robolink
 - If Robolink joints are in their marked startup positions (should point straight out), the software limitations will be correct
 - If not, use Ty's Xbox code (<http://robot.wpi.edu/wiki/index.php/Robolink>) to put it in startup position, then power cycle
- Running code:
 - <https://github.com/robolink>
 - To run the Robolink
 - Run roscore
 - Run roslaunch robolink robolink.launch
 - Run roslaunch robolink CommandAcceptor.launch
 - To run the CTI Joystick
 - Run roscore
 - Run roslaunch rad rad2_indjoy.launch

12.1.2 Repair History / Incident Report

Broken Motors in White Box:

- M121594 002 - Broken Hall Sensor
 - Was the spare (6th) motor, never used due to broken sensor
- M121594 004 - Bent Axle
 - Was the motor for the yellow cable (Joint 3), was found bent on 5/28/13 with cable pulled out of metal fastener

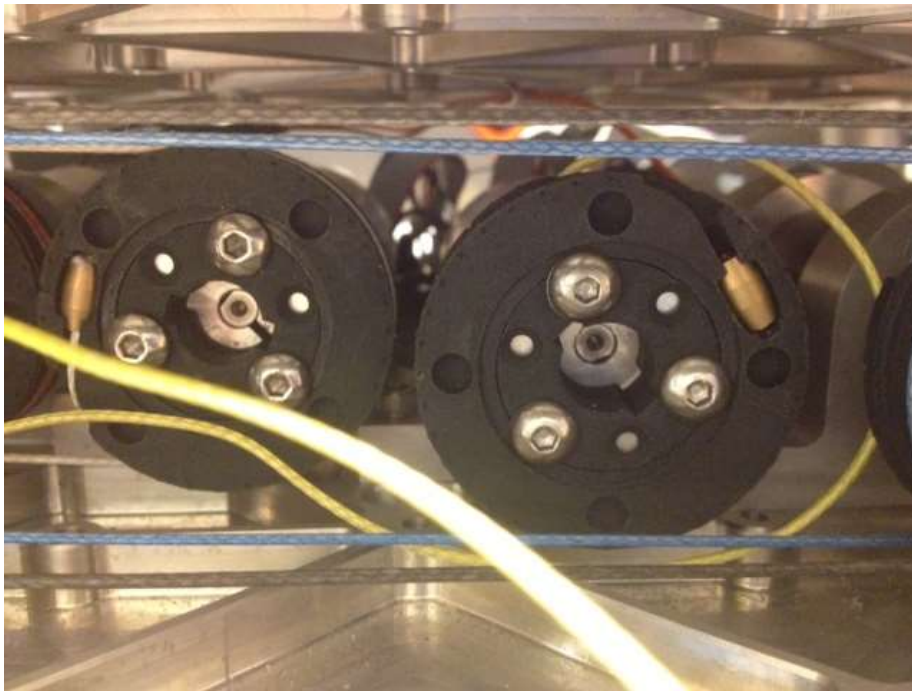


Figure 27: Motor Cable Pulled from Fastener

- M121594 003 - Bent Axle
 - Was the motor for the black cable (Joint 1), axle bent when the code crashed while mounted on the CPS chair - the motors continued to run and end effector pressed on the leg of the teammate seated in CPS chair. While not enough force was exerted to injure the team member, the motor axle bent. After this, the emergency stop button was added

Yellow cable was repaired after the bent axle incident

- Repair consisted of cutting the cable and knotting on a new piece such that the knot remains on the wheel through the range of motion

- The knot was a strong sailor's knot, sewn through several times (green thread), and super-glued


Current status:

- One motor axle slightly bent, still useable:



Figure 28: Current Motor Axle Bend

1



WPI

Robolink: Modular Assistive Robot Arm

**Cassiopia Hudson
Gabriel Morell-Pacheco**

Worcester Polytechnic Institute
Robotics Engineering, Electrical and Computer Engineering
Advisor: Taskin Padir
Major Qualifying Project
August 29, 2013

2

Intro - Goal

To establish control of the five degree of freedom Robolink arm to complete tasks useful for persons with no or limited mobility.

Specifications:

- Complete tasks
 - Drive Joystick
 - Flip Light Switch
 - Turn Book Page
- Design a modular addition to CPS
 - Easily mounted/removed with a useful workspace
 - Completely mobile system
 - Passive end effector

Background

- Other Robolink applications include humanoid, industrial, and fishtail robots



<http://www.fishlink.com/robot/robot.html>

Background

- Other Robolink applications include humanoid, industrial, and fishtail robots



<http://www.youtube.com/watch?v=Arty5eRQC78&list=PL07B9390F7512B3B4&index=43>



<http://www.youtube.com/watch?v=4eM0b-BGxM&list=PL07B9390F7512B3B4>

Background

Robot Arm Name:	Developed By:	Year Developed:	Control:	Weight of arm:	Size: (Compact(1) to Bulky(5))	End Effector:	Number of Tasks: (Few(1) to Many(5))	Weight can lift (Payload):	Extension:	Degrees of Freedom:
PerMMA (Personal Mobility and Manipulation Appliance)	Carnegie Mellon University, University of Pittsburgh	2009	Joystick, Brain Signals, Touch Screen	28.600 kg	5	2x Gripper	5	2.500 kg/arm	0.630 m/arm	6
DORA (Door Opening Robotic Arm)	UMass Lowell	2009	Joystick	13.063 kg	3	Gripper	1	N/A	1.245 m	7
IARM (Assistive Robotic Manipulator)	Exact Dynamics	2009	Keypad, Joystick, or Single Button	9.000 kg	1	Gripper	5	1.500 kg	0.900 m	6
WMRA (Wheelchair Mounted Robotic Arm)	University of Southern Florida	2008	Brain Signals	13.750 kg	2	Gripper	5	4.500 kg	1.082 m	7
Menus ARM (Assistive Robotic Manipulator)	Exact Dynamics	~1992	Touch Screen User Interface, Camera	14.300 kg	3	Gripper	5	1.500 kg	0.800 m	7

Background

Robot Arm Name:	Developed By:	Year Developed:	Control:	Weight of arm:	Size: (Compact(1) to Bulky(5))	End Effector:	Number of Tasks: (Few(1) to Many(5))	Weight can lift (Payload):	Extension:	Degrees of Freedom:
RoboLink	Igus	2013	Brain Signals, Joystick	10.3 kg	1	Passive	3	N/A	1.187 m	5
RAPUDA (Robotic Arm for Persons with Upper Limb DisAbilities)	Intelligent Systems Research Institute (AIST) Japan	2010	Joystick	5.897 kg	3	Gripper	3	0.454 kg	0.914 m	5
Human in Loop Integration of An Arm Mounted Wheelchair Robot Based on RT Middleware	Sugano Lab Waseda University	2010	Joystick, Touch Screen, Brain Interface	5.000 kg	2	Gripper	4	1.000 kg	1.070 m	7
JACO Rehab Edition	Knova	2010	Joystick	5.700 kg	1	Gripper	5	1.500 kg	0.900 m	6

Approach - Joystick

- Needed a joystick
 - Analog
 - USB Compatible
 - Wheelchair Mountable
- CTI Electronics Industrial Joystick
 - Interfaced with existing CPS system
 - Subclasses existing driving code



<http://www.ctielectronics.com/DEM-Analog-Industrial-Joysticks.html>
active-industrial-joystick.php#

Approach - End Effector

Needs:

- Passive
 - Unpowered
- Versatile
 - Multiple Tasks
- Modular
 - Easily Removable



Approach - Mounting

Needs:

- Stable mounting
- Low cost
- Modular

Approaches:

- Under armrest
- Behind seat



Approach - Mounting

Solution:

- Attaches to existing wheelchair mounting points
- Attaches to existing box mounting points
- Built with 80/20



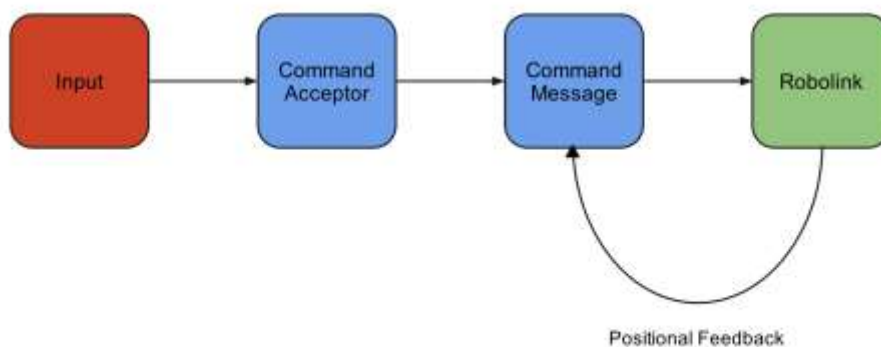
Approach - Mounting

Solution:

- Attaches to existing wheelchair mounting points
- Attaches to existing box mounting points
- Built with 80/20



Approach - Code



Approach - Code

Needs:

- Arbitrary Tasks
- Repeatable

Solution:

- **CommandMessage**
 - Sets values on properties in order to move the arm
 - Reusable and Repeatable

Simple Example

```
| CommandMessage(positions=[(0, .05, 0), (0, .1, 0)], reset=False)
```

Approach - Code

Solution:

- **CommandAcceptor**
 - Creates ROS Subscriber
 - Executes Command Messages that are published

Usage Example

```
| import rospy
| from std_msgs.msg import String
| pub = rospy.Publisher('cmdacceptor', String)
| rospy.init_node('cmdpublisher')
| pub.publish(String('reset'))
| pub.publish(String('drive_neut'))
```

Approach - Code

Command Acceptor Example

```

cmddict = {
  'drive_neut': CommandMessage(
    positions =
    [(-8, 5, 0, 89, 0), (-8, 4, 0, 90, 0), (-8, 2, 0, 88, 0), (-8, 0, 0, 87, 0), (-8, -1, 0, 85, 0), ],
    method=None,
    reset=True,
    method_sub_map=METHOD_SUB_MAP_JOINTS,
    control_mode=JOINT_VELOCITY,
    DRO=['drive_stop', 'drive_fwd', 'drive_back'],
    backsteps=[0, 1, 2]
  ),
}

cmdacceptor = CommandAcceptor(cmddict)

```

Results - Mounting

- Arm mounting is stable
- Efficient workspace



Results - Flipping

Configuration 1: $40 < i < 70, 120 < j < 150$

Configuration 2: $0 < i < 120$

Configuration: [45, i, 0, j, 0]	2 lbf Force	3 lbf Force	Configuration: {0, 90, i, 90, 0}	2 lbf Force	3 lbf Force
Nominal Torque (0.0827 Nm)	None	None	Nominal Torque (0.0827 Nm)	for $0 < i < 61$ deg	for $0 < i < 60$ deg
Stall Torque (0.782 Nm)	None	None	Stall Torque (0.782 Nm)	for $0 < i < 71$ deg	for $0 < i < 67$ deg



Results - Driving & Flipping



Results - Turn Book Page

- Within the workspace
- Capable of exerting enough force
- Some preliminary testing was done with a laptop on a desk
- Rubber end effector would turn page easily

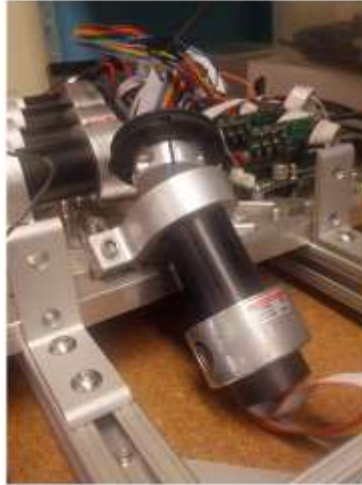


Conclusion

- Goal Completion:
 - Tasks:
 - Drive Joystick
 - Flip Light Switch
 - Turn Book Page
 - Design:
 - Stable Mounting
 - Mobile System
 - Passive End Effector

Recommendations

- Improve motor bracing
 - Axles only supported on one side
 - Motor axles will bend before cable will break



Recommendations

- Issues with repeatability and precision
 - Between power cycles, coordinates are inconsistent due to droop
 - Feedback loop with Kinect or CPS encoders
 - Joint angles are integers of degrees



Recommendations

- Experiment with other passive end effectors
 - Repeatability is required for current one
 - More malleable end effector could work well



<http://www.taskinsolutions.com/blog/index.php/2013/09/12/innovative-robotic-end-of-arm-tool/#h3pWRoQx8>

Thanks To:

Taskin Padir, for his help and guidance as our advisor,

Ty Tremblay, for configuring the Robolink and getting us started,

Lillian Walker, for contributing her time and CAD experience,

Joe St. Germain, for his constant and helpful advice,

James Fleming, for his expertise on the CPS wheelchair, and

Thomas Angelotti and **Patrick Morrison**, for opening the ECE shop to us.



Questions?