

GNG 2101
Design Project User and Product Manual

Project Deliverable H

Submitted by:

Robogrip C2.2

Ashan Dindyal, 300271949

Jiadong Yu, 300109867

Kyle Leonard, 300244275

Alif Adnan Alif, 300264791

Michael Haddad, 300298328

April 12th 2023

University of Ottawa

Table of Contents

Table of Contents	ii
List of Figures	iv
List of Tables	vi
List of Acronyms and Glossary	vii
1 Introduction.....	1
2 Overview.....	2
2.1 Conventions.....	3
2.2 Cautions & Warnings	3
3 Getting started.....	4
3.1 Configuration Considerations	4
3.2 User Access Considerations	4
3.3 Accessing/setting-up the System.....	4
3.4 System Organization & Navigation	5
3.5 Exiting the System	5
4 Using the Robo Grip	5
4.1 Controlling the Robotic Arm by App.....	5
4.1.1 Locomotion	6
4.1.2 Switching Pages	6
4.1.3 Arm Controlling.....	6
4.1.4 Claw Controlling.....	7
5 Troubleshooting & Support	8

5.1	Error Messages or Behaviors	8
5.2	Special Considerations	8
5.3	Maintenance	8
5.4	Support	8
6	Product Documentation	10
6.1	Final Prototype	10
6.1.1	BOM (Bill of Materials)	10
6.1.2	Equipment list	12
6.1.3	Instructions.....	13
6.2	Testing & Validation.....	26
7	Conclusions and Recommendations for Future Work	26
8	Bibliography	27
	APPENDICES	29
9	APPENDIX I: Design Files	29
10	APPENDIX II: Other Appendices	30

List of Figures

Figure 1 : Final Product

Figure 2 : Two power switches on the back

Figure 3 : “Connect Bluetooth” button on the app

Figure 4 : Two pages of the app interfaces

Figure 5 : Base Design

Figure 6 : Bottom of base

Figure 7 : Creation of metal base

Figure 8 : Ball screw mountain bracket design

Figure 9 : Build of the bracket design

Figure 10 : All components together

Figure 11 : Hand mechanism design

Figure 12 : PVC pipe

Figure 13 : CAD design of the hand attachments

Figure 14 : Final 3D print design

Figure 15 : 3D print of hand

Figure 16 : Arm mounting to ball screw with adhesive

Figure 17 : Design of the circuit board

Figure 18 : Arduino code

Figure 19 : Electronic equipment build

Figure 20 : MIT app inventor

Figure 21 : MIT code blocks

Figure 22 : Bluetooth testing

Figure 23 : Weight limit testing with different objects

Figure 24 : Steering testing

List of Tables

Table 1. Acronyms.....	vii
Table 2. Referenced Documents.....	29

List of Acronyms

Table 1. Acronyms

Acronym	Definition
CAD	Computer Aided Design – Use of software that allows for design of a part (Onshape as example)
JS	JavaScript – A programming language most commonly used for applications

1 Introduction

The purpose of this document is to inform the user of RoboGrip's main functions, in addition to a guide on how to properly use RoboGrip. There will be a brief overview of the product, then a step-by-step guide on how to set up and use RoboGrip and the overall process of creation of all the various subsystems. All the relevant product documentation will also be included such as troubleshooting any errors that may arise.

2 Overview



Figure 1

A need existed for a fully autonomous sturdy and stable robotic arm that is capable of comfortably picking up household items such as cups and containers from a height of a kitchen counter. The arm must be controlled via a simple to use mobile app. Our product consists of an aluminum frame, along with a 3D printed claw mated to a copper extension, an Arduino was used as the microcontroller alongside an android app created using MIT app inventor.

When looking at other solutions on the market, our team found that the other products simply lacked the type of free-range mobility that our client prioritized, the other products would

be mounted to the wheelchair itself. Additionally, these products lacked the control method that the client wished for, as the other products use a separate joystick and screen unit. We made these aspects to be the highest priority when developing and building our product.

2.1 Conventions

We utilized several stylistic and command syntax conventions in order to simplify the use of our product. For instance, instead of mentioning the direction the button will move the robot, we simply indicated so by using arrows (**←↑→↓**). Additionally, we indicated the Stop button with a stop sign for a more stylistic and obvious functionality of the button.

2.2 Cautions & Warnings

The product is not designed for lifting heavy things (> 0.5kg), so not applicable for heavy duty works. RoboGrip is not liable in any way for the risk, danger, injury or damages caused by the other party. A Hold Harmless Agreement will need to be signed with the following information: the name of the party that will not be held liable, name of the party offering protection from liability, date of signature, as well as the province where the parties reside, in the case where both parties reside in different provinces, either address can be chosen.

3 Getting started

3.1 Configuration Considerations

Download and install the .apk file on Android phone. Enable Bluetooth access.

3.2 User Access Considerations

Only Android users may access the app at the moment. IOS users may use the robot through a wired connection, however, downloading the app and connecting via Bluetooth is only function for Android.

3.3 Accessing/setting-up the System

Open two switches on the back of the robotic arm. Lights will turn on three boards for indication of power.

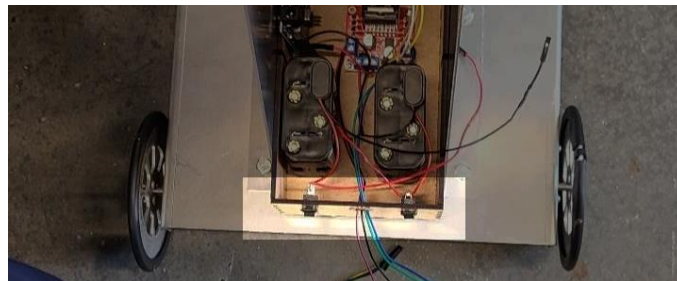


Figure 2

Open the app, and press “connect Bluetooth” button to open the list of possible connections, press to connect “HC-06” showing in the list to connect the phone to the robotic arm. There will be a prompt sound if the connection is established.



Figure 3

3.4 System Organization & Navigation

From hardware aspect, the system can break into three sections: tracks, arm, hand.

Tracks: two dc motors drives wheels at the back, performing basic motions in four directions: forward, backward, left, and right.

Arm: A stepper motor on the top connects to a thread rod, allowing the whole arm to move vertically (up and down).

Hand: The grabbing mechanism is designed to use a single servo motor to control the opening and closing motion of the claw.

From software aspect, the user interface is designed with two separated pages.

Home Screen: Takes the charge of connecting Bluetooth and controlling the locomotion.

Second Screen: Controls the arm vertical motion and claw grabbing motion.

3.5 Exiting the System

It is important to follow the following steps in order to prevent any injury or damages:

- Exit the app or disconnect the Bluetooth from phone settings.
- Close the two switches on the back of the robotic arm. Make sure all indicating lights turns off.

4 Using the Robo Grip

The following sub-sections provide detailed, step-by-step instructions on how to use the various functions of controlling the robotic arm by the App.

4.1 Controlling the Robotic Arm by App

After powering up the robot and establishing the Bluetooth connection (section 3.3), the robotic arm is ready to be used, with simple controlling app on the phone.

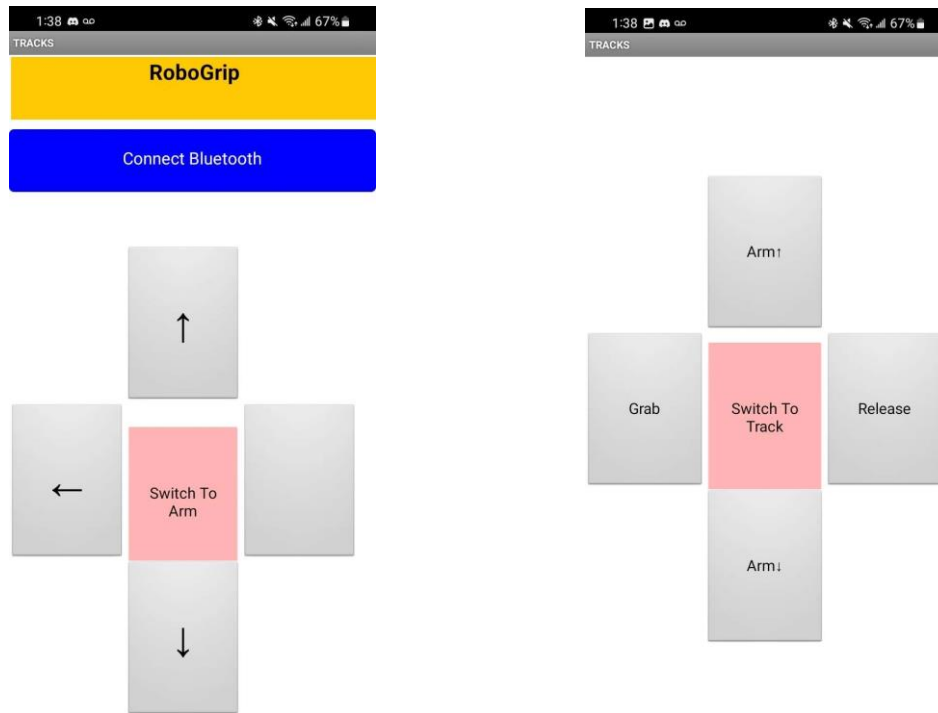


Figure 4

4.1.1 Locomotion

Four buttons on the homepage (first page on the left) control the track motion. Pressing and holding a button will move the track in the corresponding directions (forward, backward, left and right). The motion will stop if the button is released.

4.1.2 Switching Pages

On both pages, there is a button in the middle, which takes charge of switching between different pages. Press on the button and the app will switch to the other interface so that the app can control the other functions of the robot.

4.1.3 Arm Controlling

On the second page, the top button with the text “arm up arrow” will raise the arm, and the one with the text “down arrow” on the bottom will lower the arm. The arm can only be moved in vertical axis. All buttons need to press and hold to perform continuous motion, the motion will stop if the button is released.

4.1.4 Claw Controlling

On the second page, the left button with the text “grab” will close the claw for grabbing something, and the one with the text “release” on the bottom will open the claw, so that it can release the holding item. All buttons need to press and hold to perform continuous motion, the motion will stop if the button is released.

5 Troubleshooting & Support

5.1 Error Messages or Behaviors

Some parts that are prone to breaking include:

- The motor located near the top of the machine is prone to breaking as it does not have enough force to rotate the screws properly for an extensive amount of time.
- The motor, as well as the circuit board can overheat and break due to high heat if used for too long.
- The moving finger on the claw may slightly detach from the servo motor allocated in the stationary finger stopping all opening and closing movement.
- Over time, rubber may begin to peel off wheels and fingers.
- An error message appears in the application if Bluetooth is not connected properly, it is important that the device being used has no restrictions for Bluetooth.

*Refer to **5.4 Support** if the problem cannot be resolved on your own.*

5.2 Special Considerations

Some of the wires may lose connection due to lack of grip in the connectors, this may result in some or all of the lights not turning on.

To resolve this issue, simply ensure all of the wires on every circuit board are connected all the way in and reconnect any wires that may have lost connection.

5.3 Maintenance

- Every two weeks batteries should be replaced (AA batteries)
- After use of robot, ensure on/off switch is **OFF**.
- Rubber grip on wheels will need to be replaced yearly or whenever grip is noticeably low.

5.4 Support

Please allocate the 'troubled' component on the robot and correspond it to the following contact information.

- Claw & or Arm → kleon080@uottawa.ca
- App → mhadd033@uottawa.ca
- Movement & or Body of Robot → adind054@uottawa.ca
- Wiring & or Electronics → ju150@uottawa.ca

Things to include in your email:

1. State the broken component. If you cannot see the problem, state N/A
2. Explain when and how the component was broken.
3. How urgent do you need this fixed?
4. Send corresponding pictures.

*Please provide at least 48h for a response. If no response within the timeframe, call or text one of the following numbers: (613) – 914 – 6873 , 905-598-0878

6 Product Documentation

6.1 Final Prototype

6.1.1 BOM (Bill of Materials)

Item Name	Description	Unit of Measure	Quantity	Unit Cost (CAD)
Arduino	The microcontroller used for controlling the arm	N/A	1	\$17.00 - MakerStore
12V DC Motor	Used for moving the tracks	Volts	2	\$16.98 - Amazon
L298N Motor Drive Controller	Drive DC motors.	N/A	1	\$11.99 - Amazon
HC-06 Bluetooth Module	Connect the board to the mobile phone.	N/A	1	\$10.91 - Amazon
A3967 EasyDriver	Drive the stepper motors.	N/A	2	\$15.92 - Amazon
Stepper motor	Used to drive the axis of the arm.	Kg*Cm (Torque)	1	\$9.50 - Amazon
Servo Motor	Used to open/close the claw mechanism	Kg*Cm		\$10.50 - DFRobot
Jumper Wires	Connect the board and the component	inch	1 (around 60 wires in a package)	\$10 - Amazon
Battery Holder (12V)	Used to hold the 12v batteries	N/A	2	\$2.15 - Digikey

AA Battery	Used to combine the 12V external power source	Voltage	8	\$0 – Already own
1200 mm Linear Actuator	Used for vertical motion of the arm (up and down)	mm (height)	1	\$105.99 - Amazon
Hex Nut	Used in the threaded rod system	N/A	1	\$0 – Already own
Aluminum Sheet	Used as the basis for the base of the arm, will contain the wheels, tracks, motors, batteries, drivers and Arduino unit	Inch	1	\$0 – Already have, use a punch press or a laser cutter to cut to dimensions and create holes, a press brake will be used to bend it into a box.
Aluminum Sheet	Used as the main support beam for the linear actuator and ensures the arm stays straight when travelling	Inch	1	\$0 – Use a laser cutter or punch press to dimension and cut holes and a press brake to bend it into shape.
Aluminum Sheet	Used to create the wheels	Inch	1	\$0 – Will be laser cut into shape for the wheels.
M6 x 30 Hex Bolts	Used to fasten various parts to the chassis	N/A	10-20	\$0 – Readily available for the team
Aluminum Shim	Used to prevent the arm from turning with the linear actuator, fastened using double-sided tape.	N/A	1	\$0 – Readily available for the team
M6 Nuts	Used to fasten various components together and to the chassis	N/A	22	\$0 – Readily available for the team
10-32 Screws	Used to fasten arm to ball screw	N/A	10-20	\$0 – Readily available for the team
10-32 Pem Nuts	Used to fasten the two front casters to the base.	N/A	4	\$0 – Readily available for the team

M2 x 20 mm	Used to fasten the 2 12V DC motors to the rear of the chassis	N/A	10	\$0 – Readily available for the team
Hex Nuts and Bolts	Used to fasten the arm base to the moving platform.	N/A	20-30	\$0 – Already in possession of a variety of nuts and bolts.
Casters	Used as the front wheels, allowing to steer on the spot.	N/A	2	\$0 – Had two laying around
Smaller screws	Used to fasten the motors, batteries and Arduino to the arm	N/A	5	\$0 – Already in possession of a variety of screws
ReactJS	JavaScript framework used for user interface.	N/A	1	\$0.00 – Can be installed for free
Integrated Development Environment (Visual Studio Code)	Software application with developer tools used to facilitate the programming process.	N/A	1	\$0.00 - The IDE we will be using is free to install

6.1.2 Equipment list

Equipment Name	Description	Quantity
Soldering Iron Station	Solder the wire and connection pins with solder and heat shrinks	1
3D Printers	Design three-dimensional prototypes and create the product by directly building them using computer aided design (CAD)	1

Amada LC 2415 Industrial Laser	Laser cutter used to cut the sheets of aluminum for the chassis	1
Amada RG 80 80 ton Press Brake	Press brake used for bending all the sheets for the chassis	1
Various sizes of hand tools such as: screwdrivers, wrenches and pliers	Used for assembly of the electronic and mechanical subsystems	-
Amada AP100 Software	Software used to create the parts to be laser cut.	1
Onshape	CAD software used to design every part of the product.	1

6.1.3 Instructions

Creation of final prototype chassis (Mechanical):

The **chassis** was designed in mind to relate to the need of being sturdy and not breaking. This was represented via the use of aluminum, which presented the benefits of sturdiness and being light. Other metals could be substituted in place of it; however this may impact its weight negatively. This was chosen as aluminum was abundant at the time of its creation. The base dimensions of 10 x 16 inches was also chosen as it would provide further stability due to its large surface area and the fact that the sheet of aluminum was convenient for these dimensions.

1. 2 sheets of aluminum were obtained, for creation of the base of which the wheels would be mounted on, the sheet should be of the dimensions that allow the base to be 10 x 16 inches. For the base, an 18 x 18 inch sheet was used. The picture below indicates how shape of the base to be laser cut, additionally, 5 holes on each side are to be created on the flaps that stick out of the rectangle depending on what model of motor is used. Holes were also created in the center of the base to allow for mounting of the ball screw and its corresponding bracket. (Note: CAD picture does not accurately represent actual design used) For all laser cutting work, the software Amada AP100 was used to create the drawings for parts.

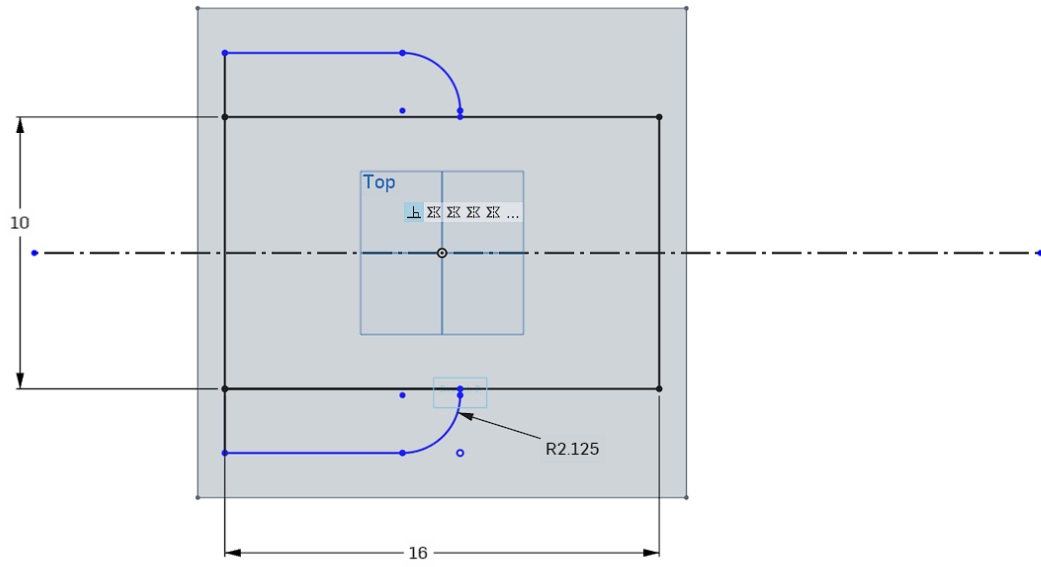


Figure 5



Figure 6

In this photo, all of the necessary holes are present. At the front for mounting of the wheel bracket, the holes in the center all relate to the ball screw mounting and its bracket. As well as the holes on the sides for the motors.

2. From another sheet of any size allowing for 2 wheels of 3 diameter, the mentioned wheels were then laser cut with the only criteria being a hole in the center of the wheel to allow it to fit on the shaft of the motor securely.
3. Then, another sheet was also created and laser cut to act as the mounting bracket for the ball screw. As such, it was required to be at least 4 feet tall. The mounting bracket is also required to have specific holes at the very top to allow for the ball screw to be securely mounted.
4. With the mains parts of the chassis created and laser cut, it is time to bend them into shape with the use of a large press brake. Starting with the base, the flaps sticking out were bended at 90 degrees to act as the side of the base.

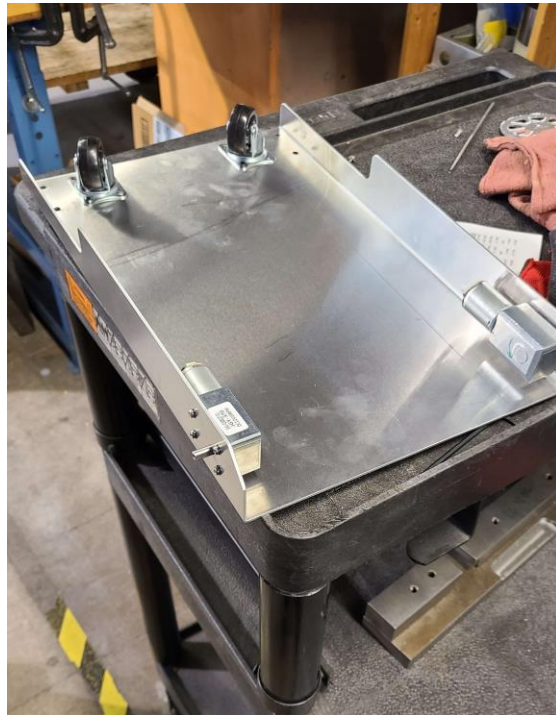


Figure 7

This photo represents the base with the side flaps bended into shape.

5. The ball screw mounting bracket also had to be bended into shape. The sides were bended 90 degrees inwards to create a structural u-shape.

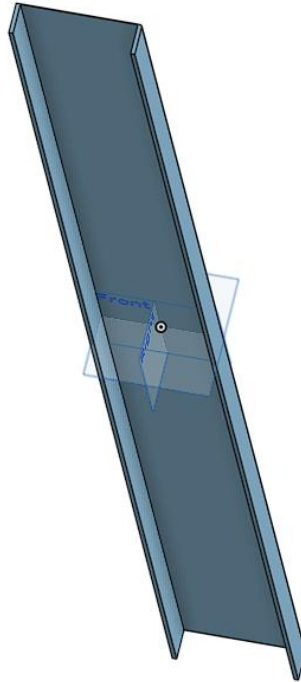


Figure 8

Mounting bracket should resemble this photo, with an additional bended piece at the bottom to allow for mounting to the base. Additionally, holes for mounting are not present here.

6. For the wheels it is noted that another bracket is required for the front wheels where it will mount in the front and be connected to the base using the front holes in the base. The front wheels will then mount to the bottom of the bracket. This was created by use of a laser cutter and the use of a press brake.



Figure 9

In this photo, the wheel bracket is present with all necessary holes on the front and sides.

7. The next steps comprised of assembly, the use of various types of screws and nuts were used to assembly the 4 ft ball screw and its mounting bracket, the motors, the front wheels and its brackets to the base. The final chassis should resemble:



Figure 10

The **arm and hand** were specifically designed to reach, grab, and hold onto household items from the client's kitchen counters. It is important to note that the prototypes were manufactured for strength, sturdiness, safety, and finally reliability.

1. After completing the ideation process it was decided that we would go with a 3D printed servo motor hand design. One finger would be stationary, holding the servo motor in a box-like attachment, while the other finger would move on a 90 degree rotation. In some ways you could imagine it as a claw. See an early picture on the idea below:

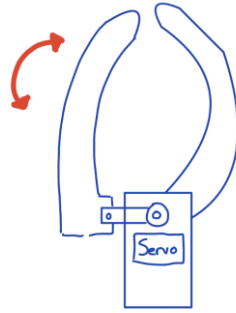


Figure 11

2. When it came to the arm, we originally planned for $\frac{3}{4}$ inch PVC piping to act as the extension; however, after discussion regarding time constraint and availability it was decided we would go forward with $\frac{3}{4}$ inch copper piping. Although it would be heavier, copper would be much stronger and sturdier than the PVC.



Figure 12

3. The hand to arm and arm to hand attachments were adapted off a past groups project found on the maker repo. The piping would be slotted and bolted into their corresponding slots. They would also be 3D printed. See picture below.



Figure 13

4. Moving forward it was time for the CAD design. We used Onshape for the complete design of our hand and its attachments. It is important to note that we carefully noted specific dimensions, of bolt & screw holes, high priority household items to be picked up, as well as the servo motors dimensions. See picture and link to Onshape document below.

<https://cad.onshape.com/documents/9e29c95d812a62d37b69b216/w/29a98689c6b58d0df5a74625/e/c952255bf191f13c5a51082e?renderMode=0&uiState=643762762ba9ed2cb5333196>

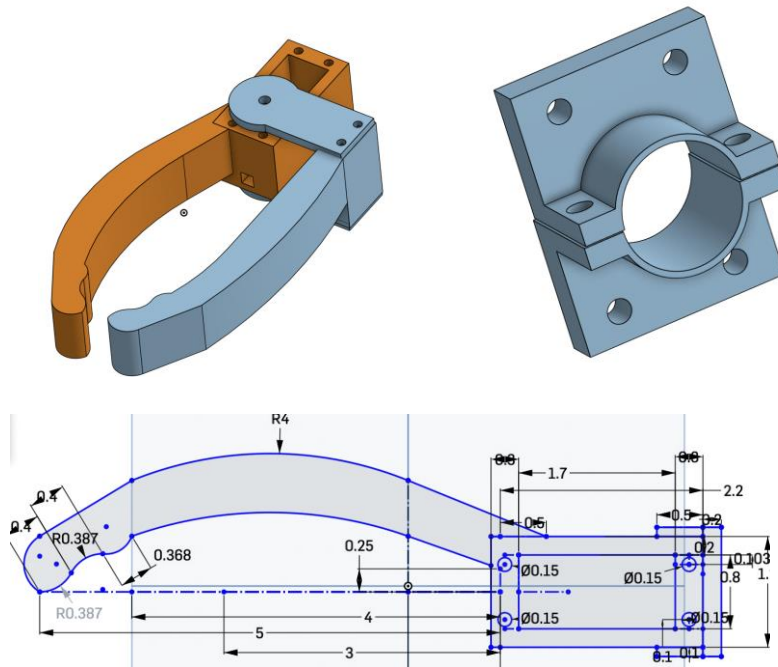


Figure 14

5. When it came time for the 3D printing, we were fortunate enough to have access to the makerspace lab. There we were able to select our aspired 3D printer of either 8mm or 6mm (the bigger mm faster print; however less precise). After loading up a stl file onto a SD card we simply plugged it all in and got it going. It is important to check the print in the early stages as it is most likely to fail something then. See picture from the process below.



Figure 15

6. Once all pieces are printed and dimensions were double checked the building process was able to commence. Using a variety of preassigned bolts and screws the servo motor was able to slide in comfortably and the hand was constructed!
7. Finally, the attachments were screwed into both the back of the claw and the front of the chassis. Then we were able to slot the pipe snug into the holes and bolt the attachments from top to bottom with the piping to pull everything together. Note that as an additional measure you may need to use silicon or construction adhesive to ensure there is no wiggle room. (This aspect will need improving in future designs). See pictures below



Figure 16

Creation of final prototype (Electrical):

For the electrical sub-system, the following circuit diagram demonstrates the pin connections between the controller and other components:

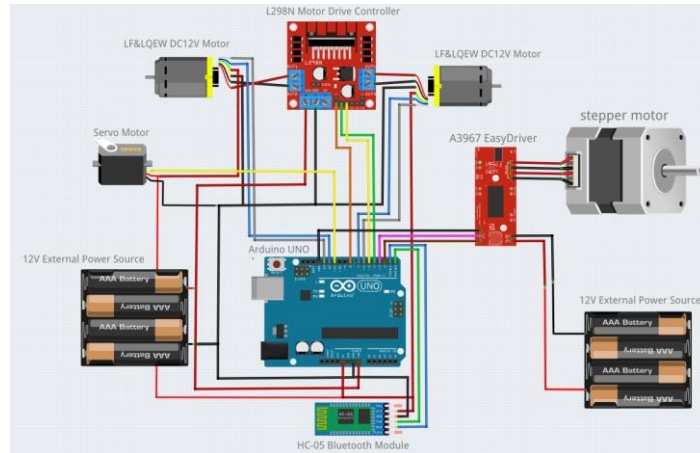


Figure 17

Arduino is chosen for controller because it is simple to code and build connections. Arduino is an open-source electronics platform based on easy-to-use hardware and software. It has a USB interface that allows it to communicate with a computer, making it easy to write and upload code. The development environment for Arduino is free and open-source software, based on the Processing programming language. The software comes with a large library of code examples and functions, making it easy to get started with programming.

```

93 | while(income_val == "L"){ //dc motor left
94 |   Serial.print(income_val);
95 |   Serial.print("run right\n");
96 |   delay(10);
97 |
98 |   digitalWrite(motor1pin1, LOW); //motor 1 backward
99 |   digitalWrite(motor1pin2, HIGH);
100 |
101 |   digitalWrite(motor2pin1, HIGH); //motor 2 forward
102 |   digitalWrite(motor2pin2, LOW);
103 |
104 |   if(Serial.available() > 0){
105 |     income_val = "";
106 |     income_val+=(char)Serial.read();
107 |   }
108 | }
109 |
110 | while(income_val == "D"){ //arm down
111 |   Serial.print(income_val);
112 |   Serial.print("arm up\n");
113 |
114 |   digitalWrite(direction_pin, HIGH); //down
115 |
116 |   digitalWrite(step_pin, HIGH);
117 |   delayMicroseconds(100);
118 |   digitalWrite(step_pin, LOW);
119 |   delayMicroseconds(100);
120 |
121 |   if(Serial.available() > 0){
122 |     income_val = "";
123 |     income_val+=(char)Serial.read();
124 |   }
125 | }
127 | while(income_val == "U"){ //arm up
128 |   Serial.print(income_val);
129 |   Serial.print("arm down\n");
130 |
131 |   digitalWrite(direction_pin, LOW); //up
132 |   digitalWrite(step_pin, HIGH);
133 |   delayMicroseconds(100);
134 |   digitalWrite(step_pin, LOW);
135 |   delayMicroseconds(100);
136 |
137 |   if(Serial.available() > 0){
138 |     income_val = "";
139 |     income_val+=(char)Serial.read();
140 |   }
141 | }
142 |
143 | while(income_val == "C"){ //hand close
144 |   Serial.print(income_val);
145 |   Serial.print("hand close\n");
146 |
147 |   while(pos >= 0){ //lower boundary
148 |     pos -= 1;
149 |     myservo.write(pos);
150 |     if(Serial.available() > 0){
151 |       income_val = "";
152 |       income_val+=(char)Serial.read();
153 |       if(income_val != "C"){
154 |         break;
155 |       }
156 |     }
157 |     delay(50);
158 |   }
159 |   if(Serial.available() > 0){
160 |     income_val = "";
161 |     income_val+=(char)Serial.read();
162 |   }
163 | }

```

Figure 18

For the motors, we use 2 DC motors for locomotion, a stepper motor for vertical arm motion, and a servo motor for claw grabbing and release. Stepper motors and dc motors requires 12V power supplies, so they are controlled by separated drivers.

During the prototype and testing, few decisions are made to improve the reliability and safety of the system. To ensure enough current to drive these motors, we separate the power supply for the stepper motor and the DC motors. We also designed a box that contains all the electronics to prevent them contacting with metal to ensure safety and avoid short circuit.

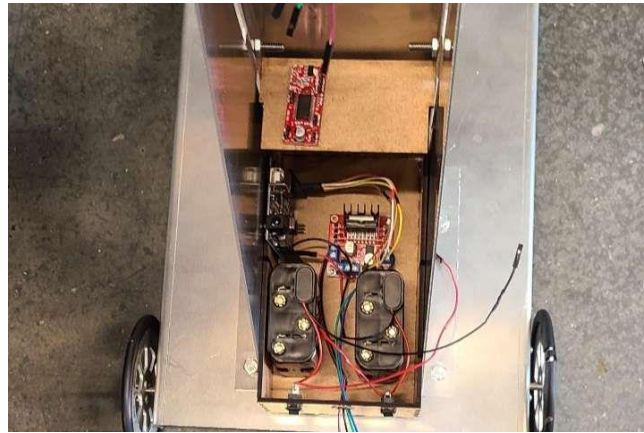


Figure 19

The **Application** was designed so that the client may use the robot from a long distance, that way our client can freely move around without being forced to travel towards the object she wants to grab. Utilizing Bluetooth, a device can control the robot up to 10 feet away.

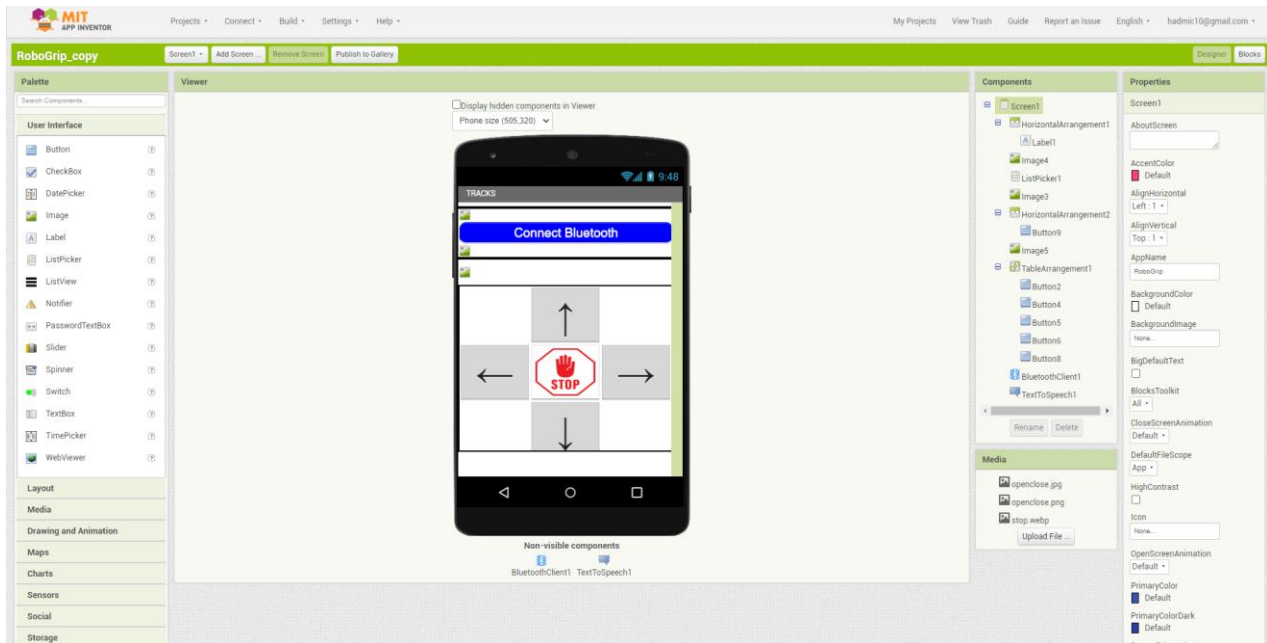


Figure 20

Fortunately, MIT app inventor is very simple to use and requires little to no knowledge in programming. Briefly, first step is to add layout arrangements (horizontal, vertical and table arrangements) in order to have a specific space where everything can be put inside. Then, put buttons, images...etc to your liking. Update the colors, the background, the titles...etc. After following these steps, the main layout is complete. Next, simply add the Bluetooth Client that is integrated into the MIT app inventor. Following up, is the creation of the program, which is super simple.

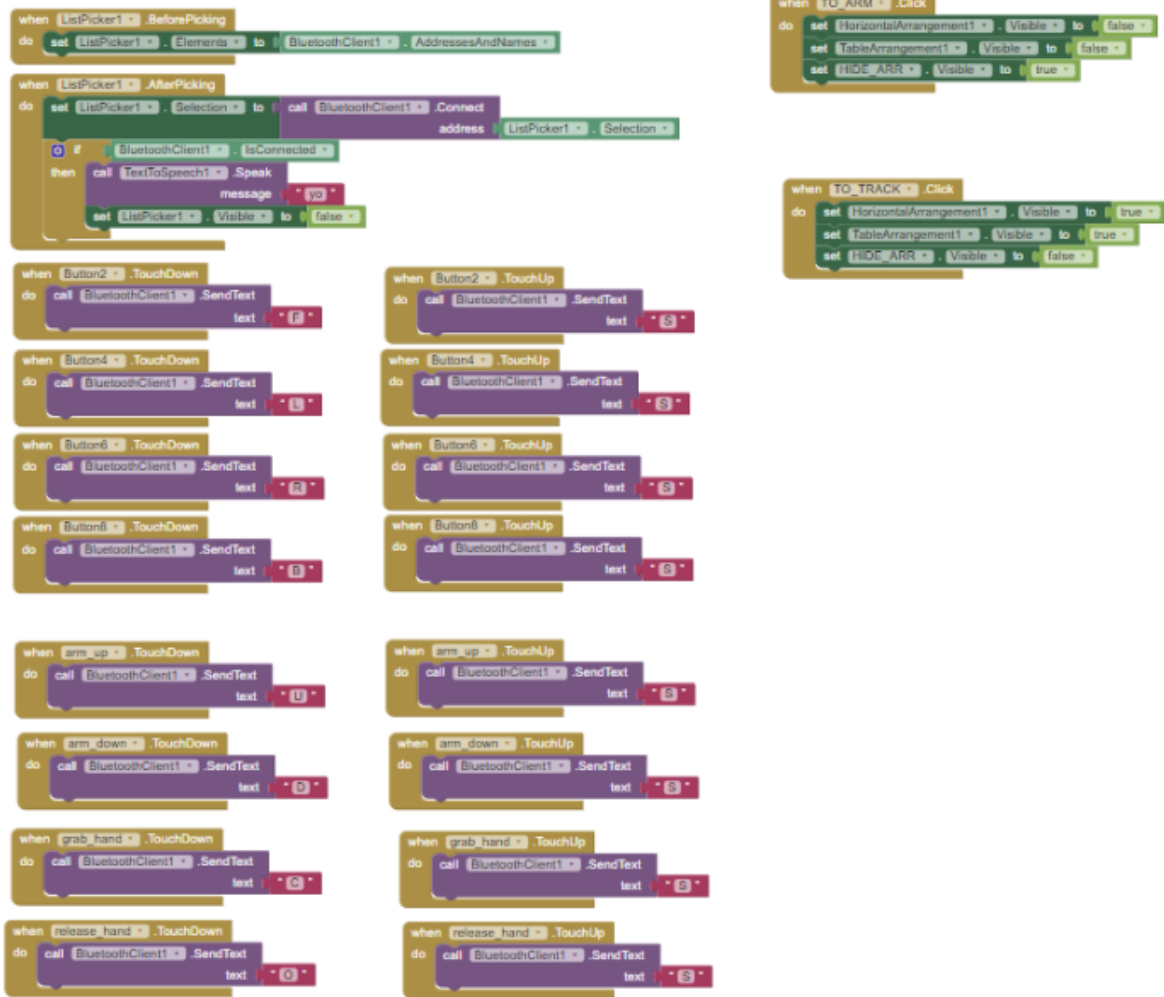


Figure 21

All in all, the main purpose of the code is for each button to return a value, and each value will serve a purpose in the Arduino code. For instance, button 2 returns F, and in the arduino code, we will simply give the value F the action of going Forward (as has been shown previously). This makes it very straight forward to create the code, and it makes it very simple for anyone trying to understand it and/or recreate it. We also need a function for the Bluetooth connection, where we add a texttospeech simply to know when the bluetooth is connected, we listen for the “yo” sound that comes out of the device. Finally, we create onClick functions near the top right, that hide certain arrangements we created earlier in order to have a completely different screen. This is very useful as we can create an app that has multiple “pages”, although it is simply one page, it appears to the user as if there are multiple and makes it an easy and enjoyable experience.

6.2 Testing & Validation

Many tests were done on the prototypes for validation of the final design.

Bluetooth connection: We performed numerous tests, ensuring that Bluetooth connection was successful throughout several different devices. After many failures, it eventually worked on three separate devices. However, there is a special requirement for the Bluetooth to be functional. By utilizing MIT app inventor, it rendered the BluetoothClient to only be operational for Android mobiles.

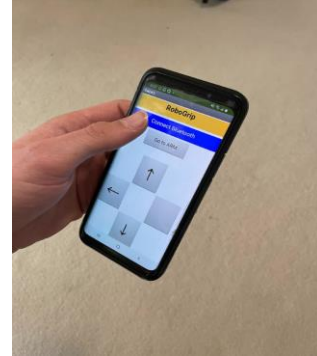


Figure 22



Figure 23

The steering mechanism was also rigorously tested in order to prove our team's theory about the proposed steering mechanism of using 2 powered wheels in the back and two free moving casters in the front. The results of this testing indicated that the steering mechanism proved to be quite effective. It can nearly steer on the spot which satisfied our conditions about good mobility. Additionally, we also found that the motors used for propulsion had more than enough torque to move and steer the nearly 30-pound unit.

Additionally, we did dozens of tests to ensure the robot arm can handle the weight of daily objects used by the client, such as: cups, bowls...etc. The hand mechanism was able to hold around 500 grams, more weight is not recommended and would result in risk of damage.



Figure 24

7 Conclusions and Recommendations for Future Work

There were many lessons learned, whether it was from team members, online sources and even from mistakes made during the long process of making the product itself. An important lesson learned is that it is vital to separate roles for certain subsystems of the project and set dates

for them. In doing so, it brings out the particular talents of each person and forces everyone to work since they are expected to deliver their part by a certain due date. Another lesson learned is that it is necessary to find the most efficient method as quickly as possible, this is a must because of the limited time we have during the semester. Our team managed to successfully accomplish this by assigning tasks for each team member based on their field of study, as that is where they are most knowledgeable. For future work, our suggestion to other groups in order to improve upon our work would be to ask for feedback very often from people with more experience, whether it's from a mechanical engineer, a professor, a software engineer or even the teaching assistants. They have more knowledge than us and experienced exactly what we are undergoing, hence, they understand the most common mistakes, the best options to take and how to deliver high quality work.

As mentioned previously, lack of time was one of the biggest missing pieces in a large project like this, and because of that, there were elements/concepts that had to be relinquished. For instance: our original plan was to develop and application from scratch using the programming language JavaScript and the library React. Although this would have been very useful in terms of usability and a great learning experience, we quickly realised that there was no way we would finish in the time given to us. As a result, we resolved this by using MIT app inventor, which facilitated the process enormously. Unfortunately, this led to the problem where it is not operational with IOS devices and can only be used with Androids. If we had a few more months to work on this project, we would build the App from scratch using JS, HTML, and CSS. As well, we were unable to make the interface pleasing to the eye as it is an avoidable step in the project. However, if we had more time, we would ensure that the app was visually appealing. Finally, if we had a few more months, we would work on polishing the design which includes a greater quality of fastening for connecting the arm to the chassis, in addition to incorporating a camera for greater controllability and the addition of a bigger rechargeable battery.

8 Bibliography

Here are some of the sources we consulted and were inspired by some ideas/concepts.

Open source project: <https://www.instructables.com/MOTUS-Open-Source-3D-Printed-Robotic-Arm/>

Arduino motor guide: <https://forum.arduino.cc/t/simple-code-moving-forward/231759>

MIT app inventor: <https://appinventor.mit.edu/>







Arm grabber : <https://makerepo.com/ZitaiPeng/1140.gng2101c3p2power-grabbereh>

APPENDICES

9 APPENDIX I: Design Files

Link to MakerRepo page containing all files: <https://makerepo.com/kleonard/1492.robogrip>

Table 2. Referenced Documents

Document Name	Document Location and/or URL	Issuance Date
Team Contract	 GNG2101_C22_PDA1_Team Contract (1).pdf	Jan 15, 2023
Problem Definition, Concept Development, Project Plan	 GNG 2101 - C2.2 - Project Deliverable B (5).pdf	Feb 2, 2023
Detailed Design	 PDC (3).pdf	Feb 14, 2023
Prototype 1 and Project Progress Presentation	 PDD (3).pdf	Feb 17, 2023
Design Constraints and Prototype 2	 PDE.pdf	Feb 26, 2023
Business Constraints	 PDF (1).pdf	Mar 22, 2023

10 APPENDIX II: Other Appendices

When using the robotic arm, there are several crucial and significant issues and protocols to keep in mind:

- i. Accuracy: In manufacturing and other sectors where precision is essential, robotic arms are frequently utilised. To obtain the appropriate level of accuracy, it is crucial to calibrate the robotic arm accurately and make sure that it is programmed correctly.
- ii. Maintenance: Regular maintainnace is important to keep the robotic arm functioning properly. This includes cleaning, lubrication and other tasks which is recommended by The RoboArm company.
- iii. Safety: Our robotic arm is capable of doing minor to major harm or damage if not used appropriately. When using a robotic arm, it is crucial to comprehend and adhere to all safety instructions.
- iv. Power supply: The robotic arm will require a reliable power supply (battery) to function properly. It is important to ensure that the power supply (battery) is adequate and that there are backup batteries options available in case the batteries which operates the robotic amr drains out.
- v. Common sense & Understanding: Basic common sense is essential for anyone who will be operating the robotic arm. This includes understanding how to program and operate the arm, as well as how to troubleshoot any issues that may arise.
- vi. Environment factors: The robotic arm's performance may be impacted by the environment in which it will be used. Vibration, humidity and both high and low temperature all have an impact on the arm's precision and dependability by minutest margin.
- vii. Cost: It is important to consider the cost of ownership, including maintenance, repairs and upgrades when deciding whether to buy the robotic arm.