CODIA MARK II

C.O.D.I.A. (Computer operated & delightfully intelligent assistant)

2019

Ellie T. Xu
Garth Webb Secondary School
2019

Table of Contents

RATIONALE

81 years. That's the average life expectancy of a Canadian (Disabled World, 2017). It sounds like a lot, but keep in mind that we spend one-third or around 25 years with our mind wandering around in sleep-land. As the average person either works or goes to school and assuming the average person does so 40 hours a week from age 4 - 62 (MacNaughton, 2016), then we would spend around 14 years in our jobs and school. That leaves us with only half of our lives - 42 years - to do things from watching movies to going to the washroom. That's pretty tight considering how we spend 11 hours a day staring at a screen (Fottrell, 2018). To summarize, we don't have a lot of time.

Now, imagine if you had to waste away those precious hours doing boring, mundane, and irritating tasks. Not a good way to spend valuable time, is it?

The goal of this project is to increase the efficiencies of our lives by eliminating certain repetitive tasks that we have to do. Although many approaches can be taken to achieve this goal, the specific way in which this project is taking to tackle this challenge is in the form of a versatile and adaptable system targeted towards manual tasks.

The existence of an adaptable system as such would greatly benefit us in terms of improving the efficiency of our lives. For instance, many of our senior population experience one of the three most common disabilities: pain, mobility or flexibility (Statistics Canada, 2017a). As a result, they experience difficulties performing tasks that require a high degree of motion, such as daily tasks or quickly navigating around their house (Statistics Canada, 2017a). This makes assistance from a capable individual extremely valuable. However, as Statistics Canada indicates, with the increase of age, the likelihood for the individual to be living alone increases as well (Statistics Canada, 2018). As a result of these trends, numerous individuals who develop the need for assistance could not receive any and end up spending a large amount of time doing what the average adult can do in an instant. With a system that aids people in their daily lives, this would be less of a problem than it is now. Joint pain from picking up objects from the floor would no longer be an issue if something else can do the job.

With everything that is going on in our modern society, we usually grit our teeth and force our way through a short but troublesome task like shovelling the snow. But despite how they seemingly take little time out of a day, they end up occupying years of our lives. Time is precious, if we could reduce those years and put it into something else, perhaps we could achieve more and live a little happier. After all, how would you like to spend your years?

CODIA Mark I

To bring the dreams of a robotic assistant in the average household a step closer to reality, Mark I was created as the first generation of an intelligent assistant that helps people with their everyday tasks such as retrieving an object. It contains a claw and arm system and has the mobility to maneuver around a house. The large breadth of accomplishable tasks the claw design can accomplish is created with the goal to adapt to the different environments across populations.

Specifically, it operates with a wireless manual control system through the user's hand gestures, monitored with rings around the fingers' joints, communicating by way of radio frequency at 433 MHz. The robotic arm for which the gesture control controls is to grab an object and lift it up or drop it down. Structurally, it is made out of wood and 0.08" thick acrylic where needed, for acrylic is flexible and strong.

This model is able to travel around on smooth floors and carpets. It is able to pick up items on the floor of up to 110g in weight and is able to be controlled through the user's hand gestures with a radio transmitter and a receiver circuit. Additionally, it is equipped with a fire sensor on board as a safety precaution for alarming the user when it is adventuring near fire.

Problems

There are many problems in Mark I:

- Being made of wood and acrylic that are sometimes hot glued together, it lacks structural integrity. This reduces the efficiency of the system.
- The circuits are breadboarded on. It makes it not so user friendly.
- It only has a claw, meaning it can't do things in which a claw cannot do.
- The wheels are too thin to rotate smoothly. It has trouble navigation tight spaces.
- The gesture controller cannot be used by people who suffer from things like carpal tunnel. It also takes a while to learn how to use it.
- The motors are not powerful enough because they don't get enough power from the Arduino.
- The wheel axis, due to being hot glued, is not straight.
- About a dozen other problems.

INTERVIEWS

Purpose

Through background research, statistics have shown that a need for a robotic assistant exists. Hence C.O.D.I.A. was created. However, this is an inference made from data and the reality may not be as predicted. To confirm a need for such a product exists, interviews were conducted. From which data were collected across populations to asses the real situations. This acts as a form of product validation.

To achieve the purpose (product validation) of the interviews, they were targeted to explore aspects of daily life that are troublesome across different populations. The collected data will hence by qualitative and will be used to analyze the conditions of populations in order to assess their needs.

Procedure

Informed Consent
To ensure all interviews were conducted with the full knowledge and consent of participants, a form with interview details were signed by all participants. If the participant is under the age of 18, then the signature of their legal guardian is also required.

Privacy
Furthermore, names, addresses, phone numbers were not recorded. However, age, any mental or physical illnesses that may interfere with daily life and all responses will be recorded anonymously.

Potential Risks
The major potential risk of the interviews is that questions may trigger an unpleasant memory or touch on a sensitive topic. This risk will be minimized by stopping these questions at any sign of discomfort.

Populations
All individuals may suffer from disabilities and illnesses or be economically disadvantaged. However, interviews will only be conducted on individuals who are able to perform daily tasks

with reasonable ease and those who are not suffering from any form of extreme disabilities/illness.

Sample Questions

"Walk me through a typical day."

- a. Narrow it down to tasks and ask if there are troubles associated with it.
    - i. Examples:
        1. *"Do you have any trouble with your medicine?"*
        2. *"Do you have trouble cooking dinner?"*

Summarized Data Collection

Participant 1: Female, born 1974, aged 43

- Decision fatigue
    - Thinking up a recipe every day for available foods
    - Choosing outfits in the morning (takes 20 minutes every day)
- Many mandatory tasks in everyday life that if possible, would be better off removed
    - Eating meals to sustain the human body
    - Washing up
    - Exercising for health concerns
- Youngest daughter (age 7) takes up lots of time
    - Don't trust nanny for safety and privacy reasons
- Do not like distractions, including music, during tasks

Participant 2: Female, born 2001, aged 17

- Has siblings who get in the way in the morning so things move slowly
- Enjoys a healthy breakfast
- Repetitive tasks are best off eliminated
    - Brushing teeth
    - Not eating, eating is fun
- Concerned about robots and artificial intelligence invading privacy
- Big obstacle: messy room > can't find anything
    - But would prefer to clean up personally as to know where everything is

Participant 3: Female, born 2001, aged 17

- Significant portion of time spent on making decisions
    - 30 minutes in the morning to choose outfit
        - However, would like to keep this because it acts as a wake up period
- Enjoys the daily commute (walking)

- Often forgets to do important things
- Easy to get sidetracked
- Do not like distractions
- Has chores (e.g. vacuum, dishes)

Participant 4: Male, born 1965, aged 54

- Long morning process
    - Do not like interactions with people in the early morning so this period acts as a buffer to get ready to talk with people
- Repetitive tasks are best off eliminated
    - Setting up computers every morning
- Do not like interruptions
    - Big emphasis
- Open to new technology, however, it’s preferred they are proven first

Participant 5: Female, born 2001, aged 17

- Has a sibling who gets in the way in the morning
- Obstacle: school
    - Teaches boring stuff that are easily forgotten
    - Sleep deprived
- Scheduling conflicts
    - Sibling wants to get to school earlier
    - Family asleep when she is still awake and doing work

Common Trends from Collected Data

- Anything mandatory or repetitive is preferably avoided if possible.
    - Meals, morning routine, night routine.
- Suffering from decision fatigue.
    - Obstacles: choosing an outfit, choosing a recipe.
- Wants freedom.
    - Obstacles: taking care of children, household chores.
- Wants efficiency.
    - Obstacles:
        - Siblings getting in the way of morning routine.
        - Messiness that causes things to be lost.
        - Slow at doing something.

Discussion

As the interviews were conducted in Oakville, Canada, within a population that has settled in this city as residents, the data collected in these interviews are subject to bias existing in this community. For instance, the average household income is $170 000 (“Average Household Income”, 2018) annually, while across Canada, this rate is at $70 000 (Statistics Canada, 2017b). The population within Oakville also has a higher education than the rest of Canada. In 2018, 70% of Oakville residents have post-secondary education while the number drops to 57% in just Ontario (“Post-Secondary Education”, 2018). Both of these factors among others contribute to bias within this community. What exists as an issue here could not be as significant elsewhere and vice versa. This bias is only amplified across the globe. As a result, the collected data is best suited for the population of Oakville, and to an extent, the population of developed countries such as Canada than the rest of the world.

Furthermore, it is not to be ignored that the interview participants may not have been comfortable to disclose more personal information, resulting in some missing links in the collected data.

Conclusion

The collected data in product validation reveals a trend in which people generally want efficiency in their lives. The first design produces an assistant to increase efficiency. However, due to the nature of its systems, such as the gesture controller, it may prove as inefficient to the needs of a group of people and as a result, this would defeat its purpose. Despite the common theme of efficiency that people seek for, there still exists a large variation in details to what that efficiency entails. This greatly limits the applications of the gesture-controlled-claw design. By studying the collected data, a new approach can be taken to increase its efficiency in which it is modulated as it would allow the user to customize it in a way that best suits their needs.

NEW APPROACH: a Modulated Design

As gathered from the conducted interviews, a modulated design would better achieve the goals of C.O.D.I.A.

A modulated design would include detachable components that can be exchanged for something else. For instance, the existing claw component can be traded for a simple screen. This would allow people to customize it to their needs. This factor can also reduce the cost for customers as they only need to purchase the base and components specifically for their lives. Irrelevant functions do not have to be purchased.

On the other hand, this design also faces drawbacks and obstacles. Some significant ones are decision fatigue and the development of the modules. Decision fatigue was mentioned by the participants in the interviews. They voiced that due to the repetitive nature of making several minor decision in their daily lives, it becomes hindersome to the quality and efficiency of their lives. With a large selection of modules available for this design, it may backfire and contribute to the fatigue. However, interview participants have also mentioned that they appreciated a certain degree of decision fatigue despite it being negative because it was a form of freedom. The second obstacle is that in order for the modules to be created, a team (i.e. companies), have to be working towards their creation. However, this can cause a new modular design to get stuck in a loop. A loop where no modules are being created because there is no one working towards them due to the lack of an existing customer base, but the customer base is not growing due to the lack of modules. To solve this issue, the design can be released with a few modules to attract an initial base.

RISK AND SAFETY

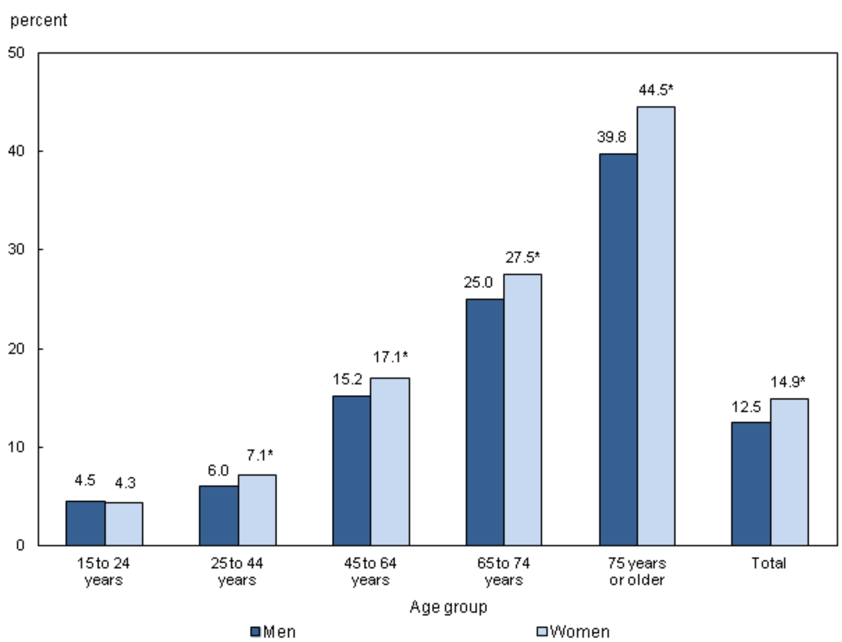
- Any circuit may short-circuit. The danger being that components may smoke and catch on fire. Precaution to take is to double check the circuit before plugging it to a power source, unplug it instantly when a sign of short-circuiting presents itself, and to not work on it when it is connected to power.
- Many workshop machines are dangerous. For example, the soldering rod is extremely hot, the band-saw can easily cut a finger in half. Therefore, safety procedures are needed, which is to say goggles on, hair tied-back, place things back when not being used, and shutting off everything when cleaning up.
- Liquids (e.g. water) can short-circuit electronics. The work area should be dry, and clean.
- Electric shock from high voltages/currents. High voltages or currents will not be used in this circuit. But sources such as the outlet still possess such dangers. To avoid it, do not bring water near the outlet, or to poke it with conductive materials (i.e. forks).

BACKGROUND

Population Statistics

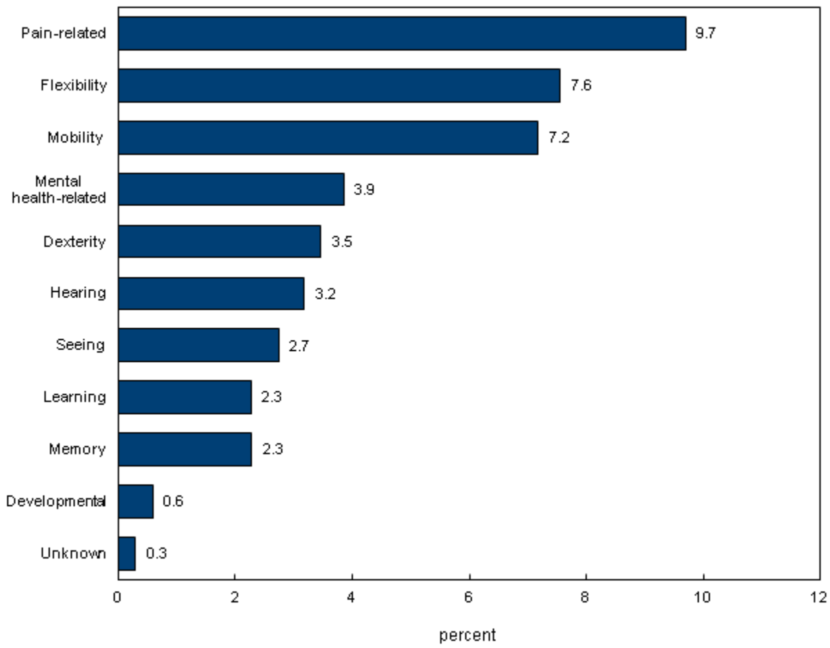
Around 12% of Canadians aged 15 or older (more than three million individuals), experience one of the three most common disabilities: pain, mobility or flexibility (refer to charts 2 and 3) (Statistics Canada, 2017a). The occurrence of these disabilities increases with age (Statistics Canada, 2017a). As a result, the older the population is, the more likely they are to experiences difficulties performing tasks that require a high degree of motion, such as daily tasks or quickly navigating around their house (Statistics Canada, 2017a). This makes assistance from a capable individual extremely valuable. However, as Statistics Canada indicates, with the increase of age, the likelihood for the individual to be living alone increases as well (chart 4) (Statistics Canada, 2018). As a result of these trends, numerous individuals who develop the need for assistance could not receive any.

Chart 2
Prevalence of disability, by age group and sex, aged 15 years or older, Canada, 2012



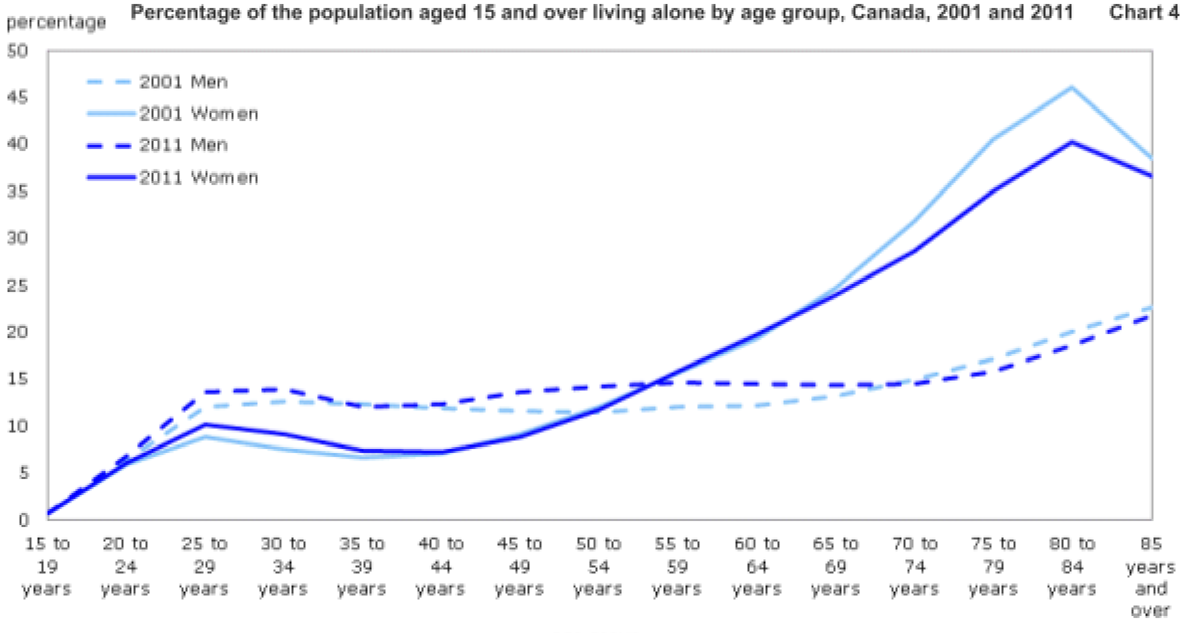
* significantly different from men ($p < 0.05$)
Source: Statistics Canada, Canadian Survey on Disability, 2012.

Chart 3
Prevalence of disabilities, by type, aged 15 years or older, Canada, 2012



Source: Statistics Canada, Canadian Survey on Disability, 2012.

Percentage of the population aged 15 and over living alone by age group, Canada, 2001 and 2011 **Chart 4**



Component Specific Research

Most component specific background research is located under the Background section in the Mark I binder. Only new, non-repetitive research is located here.

Wheels

Omni Wheels

The small wheels around the Omni wheel allow it to have great mobility. Having the small wheels reduce the side traction, making it easy to move in any direction (“Vectoring robots with Omni or Mecanum wheels”, n.d.).

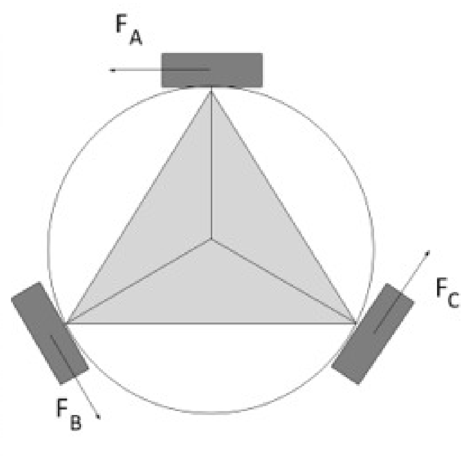


Figure 1: Vector Diagram

Omni wheels can be mounted in many different positions to function. Figure 1 shows them placed in the popular triangle formation (“Vectoring robots with Omni or Mecanum wheels”, n.d.). The force vectors in Figure 1 shows the movements of the wheels for it to rotate.

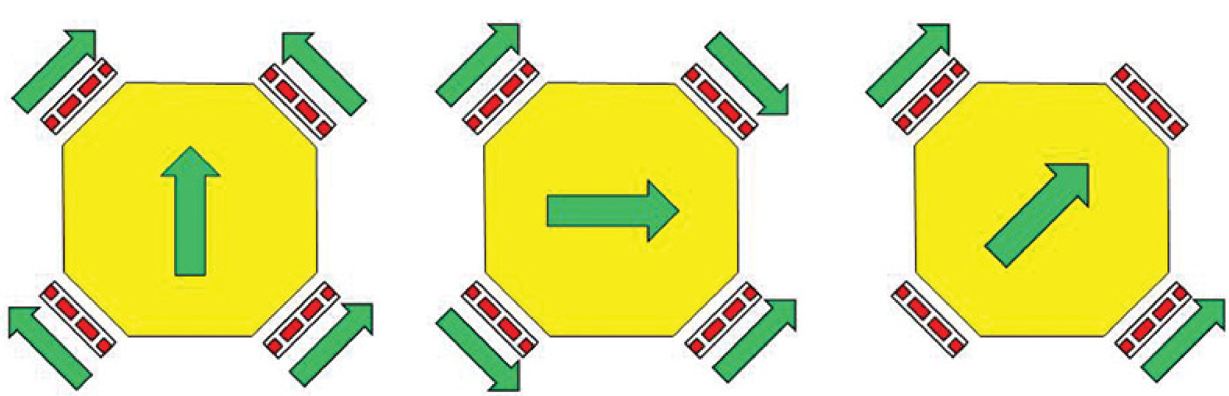
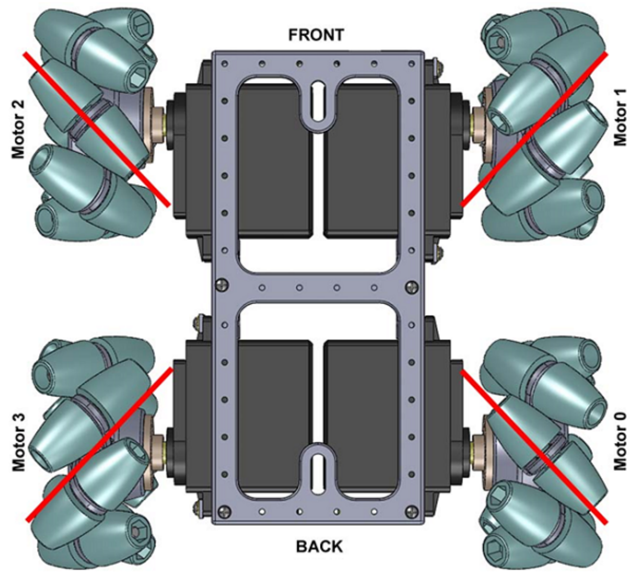


Figure 2: X-drive

However, since Omni wheels are placed in an angle to each other, much of their torque and force is used to cancel each other out and only a fraction of the original is used to move the robot. In the X-drive formation shown in Figure 2, it is only 50% efficient in all directions. This means they don't have a lot of push power and could experience trouble in inclined planes (Technik3k, 2018).

Mecanum Wheels



Similar to Omni wheels, mecanum wheels also have several small wheels on the side to allow easy movements in all directions. The difference is that the small wheels are placed 45° and they can be mounted on the sides like how regular wheels would be ("Vectoring robots with Omni or Mecanum wheels", n.d.). To turn around, or move sideways, the wheels have to be turned in the way shown in Figure 2 (Guest, 2015).

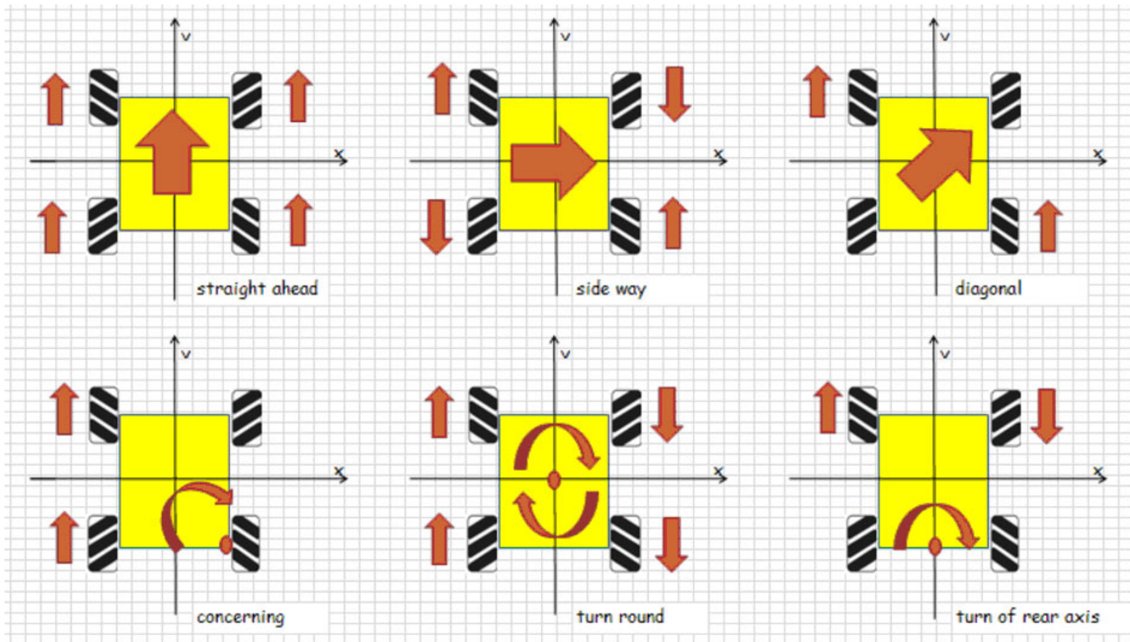


Figure 3

Omni Wheels vs Mecanum Wheels
Both wheels have great mobility and can be moved around in all directions with a fixed axis. The fixed axis means there would be no misalignment that could happen ("The advantages of Omni-directional wheels", 2017).

Table 1: Omni Wheels vs Mecanum Wheels

Wheel type	Advantages	Disadvantages
Omni wheels	• Mobility • Fixed axis • Cheaper	• Easy to get pushed around • Low push power (trouble on inclined planes)
Mecanum wheels	• Mobility • Fixed axis • More push power than Omni	• Slower • Heavier • More expensive

L293D IC

To allow the wheels to move in both the forward and the backward direction, an H-bridge can be implemented for each motor. While a circuit with transistors is able to construct this H-bridge, a single L293D IC can replace 2 of these circuits.

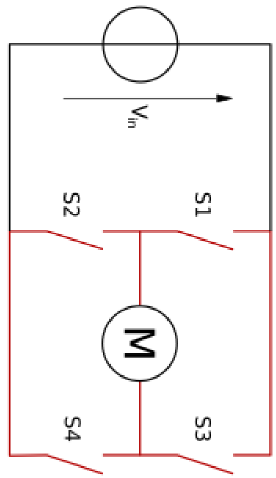


Figure 4: H-bridge

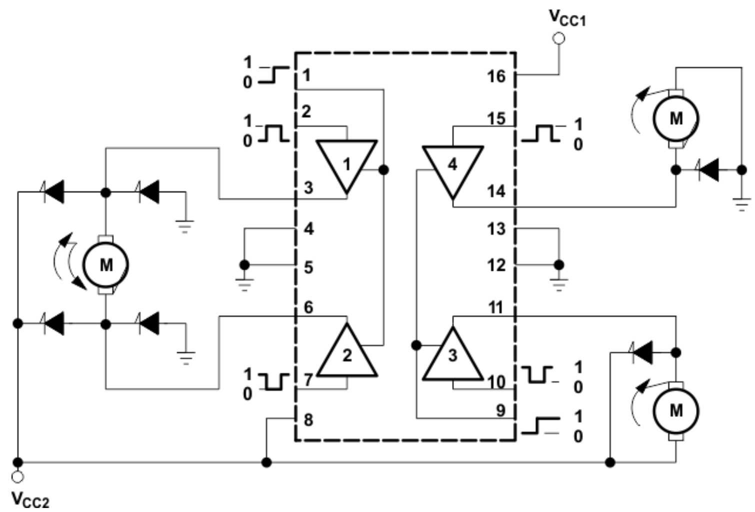


Figure 5: L293D function block

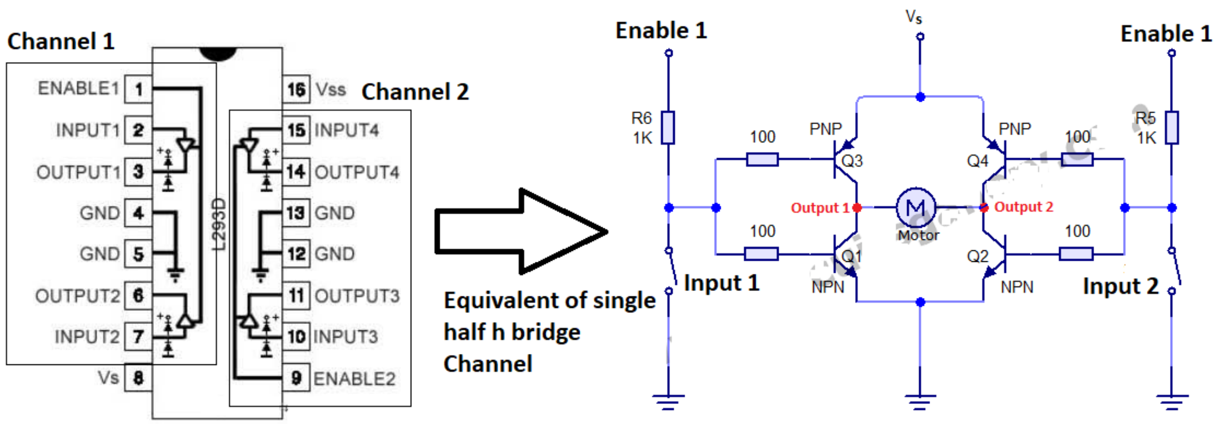


Figure 6: L293D (Microcontroller Projects, n.d.)

As can be seen from Figure 5:
Pins 2, 7, 15, and 10 are inputs that control the direction of the current to the motor.
Pins 3, 6, 11, and 14 are connected directly to the motor.
Pins 4, 5, 13, and 12 have to be grounded.
Pins 1, and 9 are enable pins that allow the channels to be turned on or off.
Pin 8 is connected to an external power source whose maximum voltage can be 36V (Texas Instruments, 2016). This means the maximum voltage of a motor the L293D can support is 36V.
Pin 16 powers the IC.

3D Printing Filaments - Plastics

The most popular 3D printing plastic filaments are ABS and PLA. They can be readily found on the market and easily purchased. As different types of plastics, they have their own advantages and disadvantages.

Other materials include carbon fibre, ASA, PET, Polycarbonate. However, these are more difficult to find and/or more expensive.

Table 2: 3D printing plastics

Plastic Types	Advantages	Disadvantages
ABS	● Strong material ● Can be welded with chemical processes ● Can withstand temperatures between -20℃ to 80℃	● Not biodegradable ● Shrinks in contact with air ● More nanoparticle emission than PLA ● More expensive than PLA
PLA	● Biodegradable ● Easiest material to print ● Cheap	● Deteriorates faster than ABS ● Not as strong as ABS

Table 2 citation: Alexandrea P., 2018.

PROTOTYPES

Module ID Code

To differentiate between the different modules attached, the following Arduino code was written to complete this task.

Prototype 1:
In here, the electrical connections to the modules would include an output pin, an input pin, power, and ground. When things start, the output pin would be set up as an input pin. The module then sends out either a HIGH or LOW output to these input pins and the data would be read as binary numbers.

For instance, if the module sends out a HIGH to pin 1 and another HIGH to pin 2, this would read as 11, making it 3 in binary. If the module sends out a HIGH to pin 1 and a LOW to pin 2, this would read as 10, making it 2 in binary.

Each number corresponds to a module. Once the module is identified, one of the input pins is switched to an output pin so that it can communicate with components such as a servo motor on the module.

However, this method is limited to two binary digits unless more pins are added. This would cap the number of available modules to three.

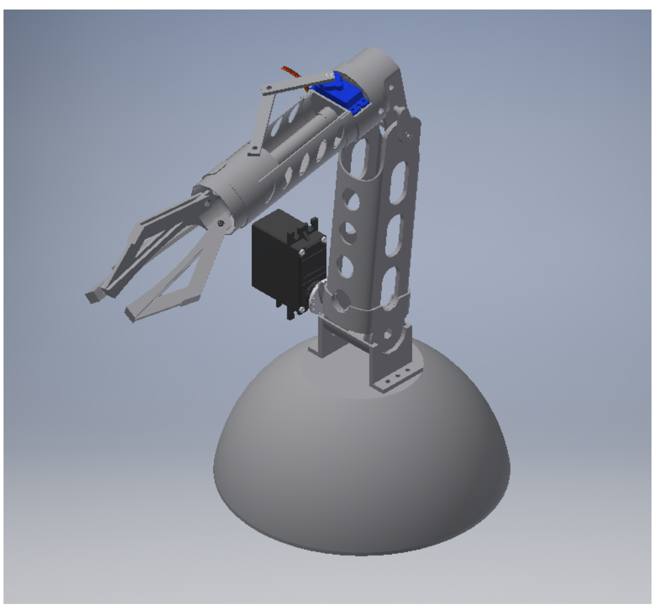
```
//module id
int pin1 = 3;
int pin2 = 4;
boolean start = true;
int id;
void setup() {
  //module id
  pinMode(pin1, INPUT);
  pinMode(pin2, INPUT);
  Serial.begin(9600);
}
```

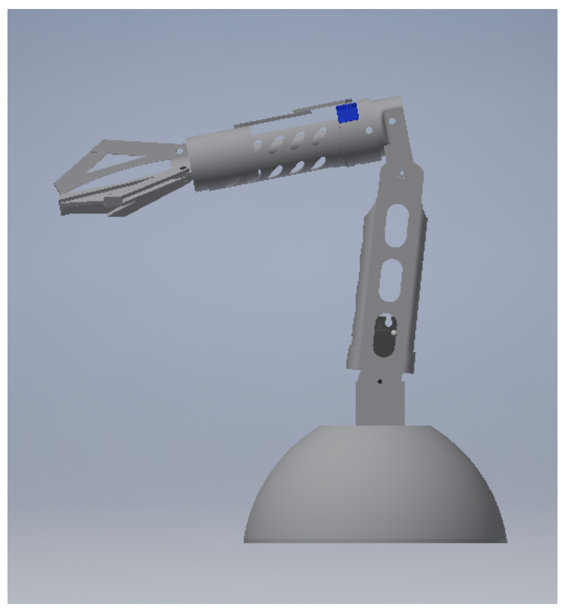
```
void loop() {
  //module id
  if (start == true){
    id = digitalRead(pin1) + digitalRead(pin2);
    pinMode(pin1, OUTPUT);
    start = false;
  }
  Serial.println(id);
  if (id == 0){
    claw();
  }
}
```

Chassis Design

Prototype 1

To make the design more house friendly, a circular approach to the chassis was aimed towards at first.



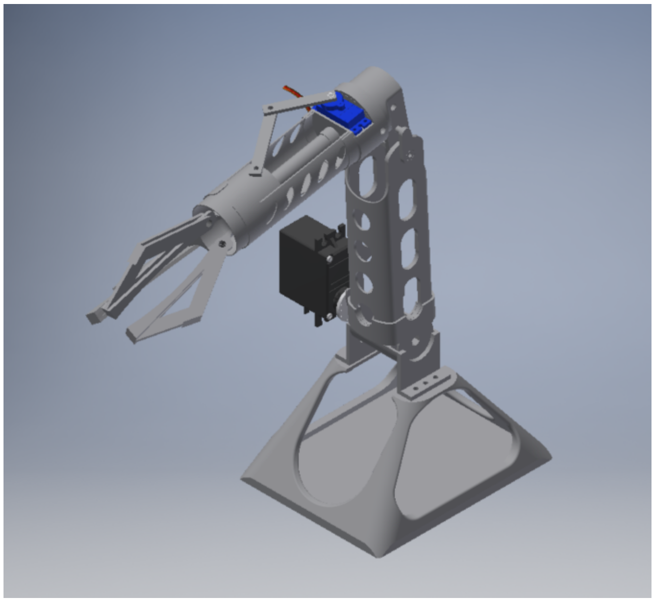


With the circular approach, it would minimize collision damage and the chances of being stuck in a corner.

However, the chassis has to be at least 2.5 inches high to accommodate for the Arduino, wires, batteries, and more inside. With the height of the arm, this makes the design very rectangular and skyscraper-like in shape. The center of gravity could easily be moved if the claw where to grab for something heavier and cause it to lose balance.

Furthermore, the Arduino and components are rectangles, which means a lot of space in the chassis would be unused as nothing can fit in there.

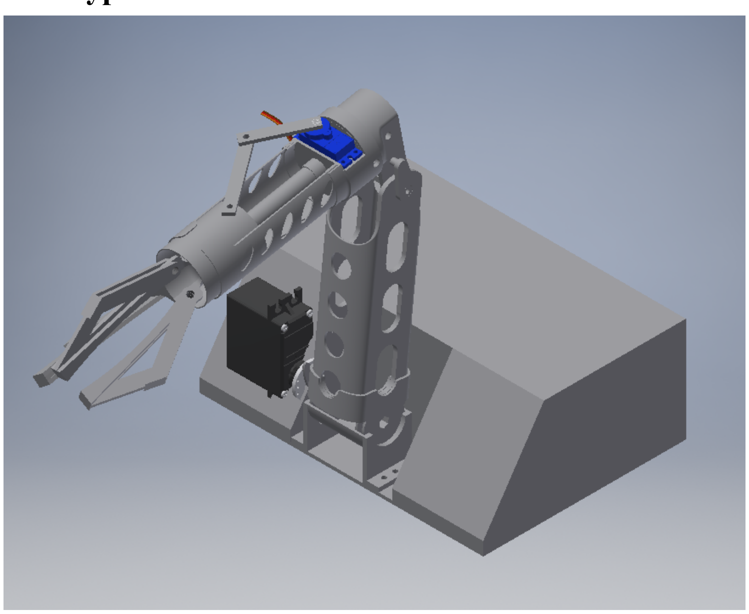
Prototype 2



This design is similar to the prototype 1. Only with a rectangle base so that it would be easier to manufacture and more space would be utilized than the circular design because the Arduino and components are rectangles. These electrical components physically cannot fit in certain areas with the circular base.

However, it has the same problem as prototype 1 in that it is prone to losing balance.

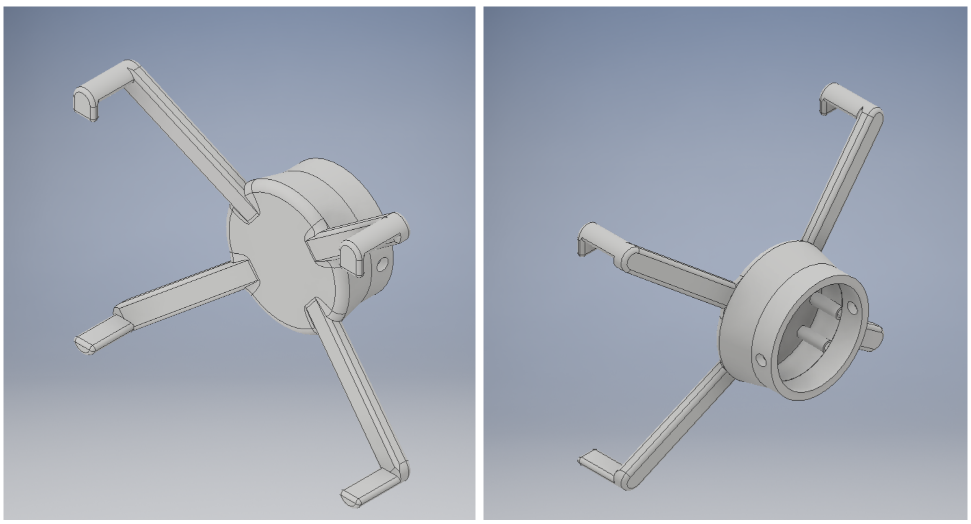
Prototype 3



To make it more stable, prototype 3 was designed. The low attachment of the arm to the base makes the overall structure more square than skyscraper. With the heavy batteries and other components placed in the back, the center of gravity would be more difficult to move outside of the chassis. This makes it more stable.
However, this is a prototype for a reason. The bottom continuous servo motor is in the way of the chassis. To be able to physically fit the motor in the position it is in now, the chamfered fronts would have to be entirely removed. This removes valuable space for electrical wirings.

Module 2

Module 2 was designed to hold a phone. So to achieve that goal, a frame was created on Inventor.



However, as hard plastic from the 3D printer, this would only work for my phone as that is the size it was basing off from. Additionally, there was no guarantee this would work and it is a large print that would take hours.

It then occurred to me that I can just buy a selfie stick and use what I know already 100% works. All I would have to do then is to make a connector that connects to CODIA.

FINAL DESIGN

Module ID Code

To differentiate between the different modules attached, the following design was used to accomplish this task.

To communicate with the modules, four pins are used: output pin, input pin, power (5v), and ground.

By utilizing the analog pins of the Arduino, the idea is that every module is associated with a certain input value. The analog pin would become the input pin that connects with the module.

The analog pins are able to read different input voltages. According to Ohm's law, voltage = resistance * current. Using this law, a different resistor is attached to every module so when 5v from the power pin is passed through it, a different value comes out depending on the resistors.

To find the different values the Arduino's analog pin associated with each voltage, the following code was used to collect the data.

```
void setup() {
pinMode(A1, INPUT);
Serial.begin(9600);
}

void loop() {
Serial.println(analogRead(A1));
delay(100);
}
```

Table 3: analog pin value and their corresponding resistors

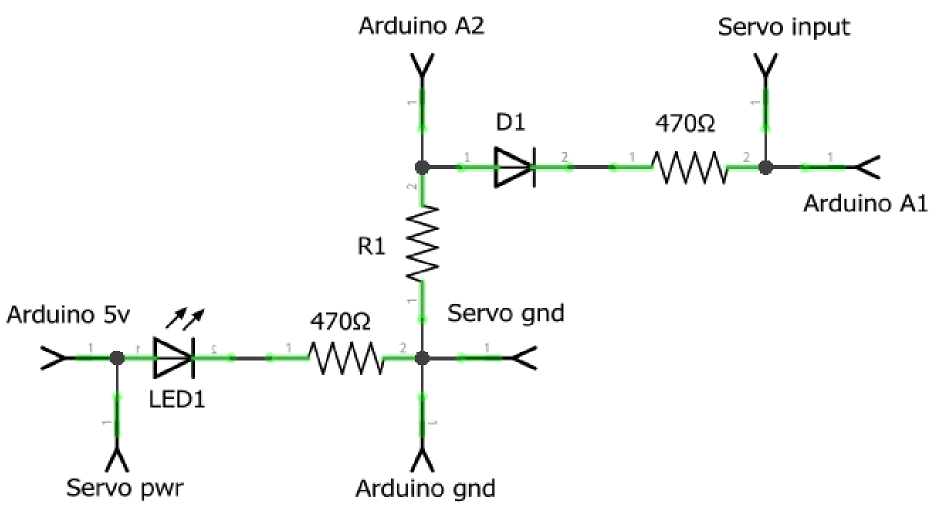
Power (v)	Resistance value (Ω)	Analog value with an uncertainty of ± 1
5	100	N/A - resistance value too low; too much current flows back to the Arduino

5	200	970
5	470	1009
5	1k	1004
5	10k	1023
5	100k	1023

*These values change depending on what other circuits are attached to the Arduino.

200Ω, 470Ω, and 10kΩ were eventually used for the modules. In the future, other values can be used for different modules.

Module-Base Communication



Schematic 1: Electrical connections for the module 1 (claw)

Four pins from the Arduino is used to communicate with the modules.

- Power pin (5V)
- Ground pin
- Analog pin 1 (A1)
- Analog pin 2 (A2)

When the module is connected with the Arduino, A1 is set as an input pin and A2 is set as an output pin. A2 is then set as HIGH so 5v will travel from it to ground and A1. On the path, the current passes through resistors so that when it reaches A1, the voltage would differ depending on the resistance value of the R1 resistor. This voltage value unique to each module tells the Arduino which module it is connected to.

Schematic 1 shows the electrical connections of the claw module. After the Arduino recognizes it is communicating with the claw module, A2 switches to LOW (or becomes an input in), and A1 switches to an output pin. A1 and A2 switch their roles to prevent current backflows and premature commands to pins like the servo input when A2 is set as HIGH during the module identification period.

The following is the code to read the ID of a module and to control the claw module:

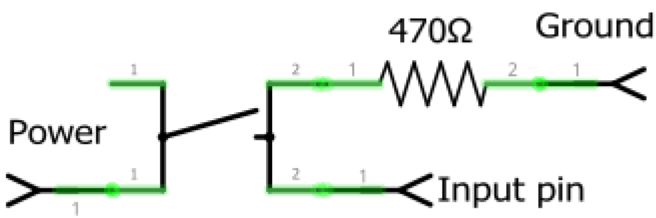
```
#include <Servo.h>
Servo servo;
long id = 0;
int counter = 0;
boolean displayed = false;

void setup() {
 pinMode(A1, INPUT);
 pinMode(A2, OUTPUT);
 digitalWrite(A2, HIGH);
 Serial.begin(9600);
}

void loop() {
 int input = analogRead(A1);
 if (input > 100 && counter < 5) {
  delay(100);
  counter ++;
 }
 else if (input > 100 && counter < 55) {
  id = id + input;
  Serial.print(counter);
  Serial.print(" ");
  Serial.println(id);
  delay(100);
  counter ++;
 }
 else if (displayed == false && counter == 55) {
```

```
    Serial.println(id);
    id = id / 50;
    Serial.print("id ");
    Serial.println(id);
    displayed = true;
    pinMode(A1, OUTPUT);
    digitalWrite(A2, LOW);
    servo.attach(A1);
  }
  if (displayed == true) {
    clawActions();
  }
}
```

Reset Button



Schematic 2: Module reset

This circuit was added as the reset button. This tells the Arduino that a module has been removed and it should start the identification process again.

An if statement under the method loop() is added for this:

```
if (digitalRead(moduleReset) == HIGH) {
  Serial.println("RESET");
  id = 0;
  counter = 0;
  displayed = false;
  pinMode(A1, INPUT);
  pinMode(A2, OUTPUT);
  digitalWrite(A2, LOW);
  delay(1000);
  digitalWrite(A2, HIGH);
}
```

Wheels

Type
CODIA is designed to be used in a house or similar environment, which means it has to be able to move around tight corners on flat floors or carpets. A fast speed is not a necessary component, in fact, it would be more beneficial to move slower as people/pets are moving around at the same time and moving slower would avoid the possibilities of a collision.
To summarize:

Things to have	Mobility around tight corners	Slow speed
Things to avoid / unnecessary things	Don’t have to design for irregular terrain	Fast speed
Nice things to have as an addition	Moving up and down stairs	Sensor to alert people when it’s stuck somewhere

Two types of wheels were used in Mark I:

Wheel 1 Wheel 2

Wheel 1 was made of hard plastic. While it had a large surface area in contact with the floor, it lacked significantly in traction due to the material. This made it ineffective when turning around or moving up inclines.

Wheel 2 had a soft rubber rim that gave it traction. However, the wheels are extremely thin. When it turned around, the large traction plus the thin frame caused the wheels to bend instead of turning CODIA around.

The major setbacks regarding these wheels were the lack of mobility and the lack of traction. To solve these problems, Mecanum wheels were used in Mark II.

Mecanum wheels have small wheels around it attached at a 45° angle. These small wheels allow them to move sideways with less traction than normal wheels would face, resulting in greater mobility. Additionally, these wheels have a fixed axis to the chassis, limiting the chances of misalignment while being able to move around in all directions.

Omni wheels are not used here because they have to be placed at an angle to each other, decreasing their efficiency. For instance, in the X-drive formation, they are only 50% efficient in all directions because half of their power is used to cancel the force of the other wheel out. This would result in trouble moving up inclines or pushing heavier objects.

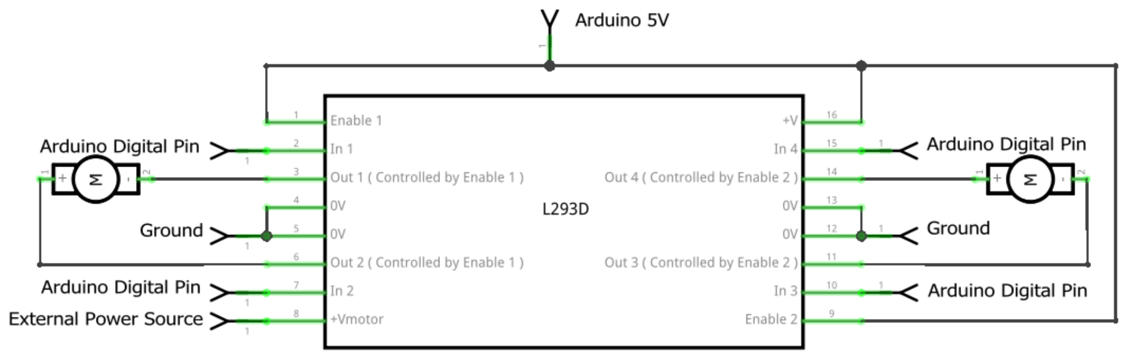
Mecanum wheels, although having their setbacks, doesn't affect their purpose as much in this project. For instance, Mecanum wheels are generally slower, however, speed is not an important factor. In fact, a fast moving robot would increase the likelihood of collision with a pet or children. Mecanum wheels can also be attached to the chassis like how normal wheels would be attached - at the sides. This means they can easily be replaced by normal wheels.

Four Wheel Drive
Four-wheeled drive is used in Mark II instead of two in Mark I. A four-wheel drive allows it to have more power and more control over its movements.

H-bridge
A circuit with transistors was used in Mark I for its two wheels (reference the design category under the Mark I binder). However, for greater mobility, mecanum wheels are being used in Mark II, which would require a four-wheel drive.

While the same circuit that was used in Mark I could be replicated again for the other two wheels, L293D ICs were used instead.

These ICs would greatly simplify the schematic and take away unnecessary cluster. Additionally, with an IC socket, these ICs can easily be taken out and replaced with new ones if they broke down. Whereas, in the transistor circuit of Mark I, replacing components would involve desoldering and resoldering them.



Schematic 3: L293D

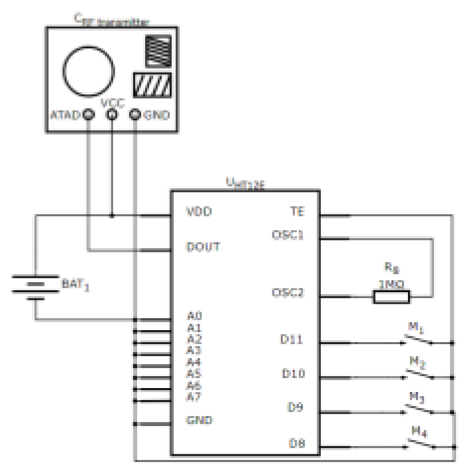
To achieve four-wheel drive, schematic 3 was created. This circuit was made twice as there are four motors. So a total of two L293D was used.

The input pins on the L293D is connected to the Arduino digital pins 4 - 11, inclusive.

Radio Frequency Remote

Mark I was controlled wirelessly through hand gestures. However, I realize that a remote alternative was still important to have. The gesture controller, while being beneficial in many situations, requires great hand dexterity and it takes longer to become adept at it.

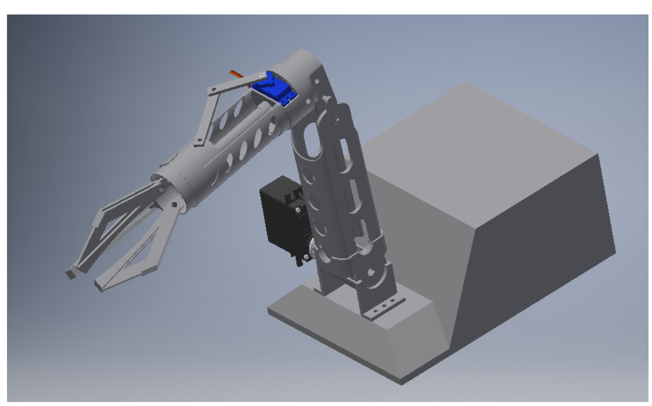
As a result, schematic four was created on a PCB and soldered on. It is powered by a 3V button cell battery and a frame was made for it so it’s more comfortable to hold (reference drawing 23).



Schematic 4: radio frequency remote

Base Design

The base is designed as a rectangle with the arm attached in the front (as shown in the picture on the right). It’s designed this way to maximize stability. The batteries, Arduinos, and circuits are placed in the back to balance the weight of the arm out. The design is also made rectangular for simpler manufacturing.



Arm Design

C-channel

A c-channel is used for the arm. The c-channel is a stronger shape than simply two plates running down the sides. The rounded c-channel is similar to a hollowed tube, with the forces of the module running down the outside rims of the tube.

The c-channel is necessary because of several reasons. One the motor attached the bottom is only attached at one side, so the force distribution is greater on one side than the other. In order to spread it to both sides, both sides have to somehow be connected. While attaching a rod that extends from one side to the other helps, making it a c-channels has more contact points between the left and right sides. The c-channel adds reinforcement that makes sure it does not easily bend out of shape.

Furthermore, as thin plastic, something as heavy as a phone added to the module is a weight factor that is significant enough to cause structural damage. As evident in Mark I, objects heavier than 110g added to the claw module starts to bend the arm out of shape. Hence another reason to add the c-channel.

Holes are added the c-channel as well to reduce its mass. The more plastic the 3D printer has to print, the longer the print.

Motor holder

Motor holders are added to each continuous servo to ensure that they are fully attached to the arm. This makes sure all their torque is being used to rotate the module/arm, not to twist themselves out of shape.

Additionally, an extra motor is added to the bottom of the arm to increase the degree of motion. This did not exist in Mark I.

Module connector

The module connector is designed so that the modules can easily be removed and replaced.

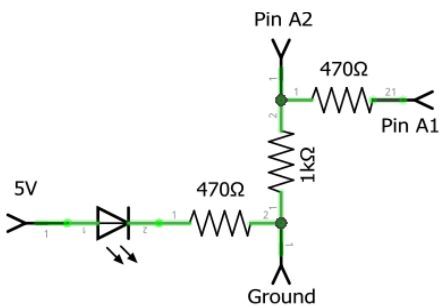
Modules

Module 1: claw

The design of module 1 is largely similar to that in Mark I's. The improvements and changes are that it is modulated. A connector the arm is created on the end and a holder for the mini servo is also added. Many adjustments and changes were made in all areas to improve its efficiency and structural integrity. For specifications, refer to the module drawings in the following pages.

Module 2: phone holder
The head of a selfie stick is removed and attached to a module connector. This is done because while a phone holder can be made, a selfie stick is already proven to work and can be adjusted for different phone types. So, it makes more sense to modify an existing proven product than to create something from scratch.

Overall Arduino Code & Other Schematics



Schematic 4: Module 2

Overall Code

```
#include <Servo.h>
Servo servo;
Servo continous1;
Servo continous2;

long id = 0;
int counter = 0;
boolean displayed = false;
boolean RFConnected = false;
int contServo1Switch = 2;
int servoSwitch = 3;
int moduleReset = 12;

void setup() {
  continous1.attach(A3);
```

```
  continous2.attach(A4);
  pinMode(contServo1Switch, OUTPUT);
  pinMode(servoSwitch, OUTPUT);
  pinMode(moduleReset, INPUT);
  pinMode(A1, INPUT);
  pinMode(A2, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(8, INPUT);
  pinMode(9, INPUT);
  pinMode(10, INPUT);
  pinMode(11, INPUT);

  digitalWrite(4, LOW);
  digitalWrite(5, LOW);
  digitalWrite(6, LOW);
  digitalWrite(7, LOW);
  digitalWrite(A2, HIGH);
  digitalWrite(contServo1Switch, LOW);
  digitalWrite(servoSwitch, LOW);
  Serial.begin(9600);
}

void loop() {
  readRFpin();
  int input = analogRead(A1);
  if (input > 100 && counter < 5) {
    delay(100);
    counter ++;
  }
  else if (input > 100 && counter < 55) {
    id = id + input;
    Serial.println(counter + " " + id);
    delay(100);
    counter ++;
  }
  else if (displayed == false && counter == 55) {
```

```
    id = id / 50;
    Serial.println("id " + id);
    displayed = true;
    if (id < 700 && id > 650) {
      clawSetUp();
      Serial.println("module 1: claw");
    }
    else if (id > 995 && id < 1005) {
      Serial.println("module 2: phone holder");
    }
  }
  if (displayed == true) {
    if (id < 700 && id > 650) {
      clawActions();
    }
  }

  if (digitalRead(moduleReset) == HIGH) {
    Serial.println("RESET");
    id = 0;
    counter = 0;
    displayed = false;
    pinMode(A1, INPUT);
    pinMode(A2, OUTPUT);
    digitalWrite(A2, LOW);
    delay(1000);
    digitalWrite(A2, HIGH);
    digitalWrite(servoSwitch, LOW);
  }
}

void readRFpin() {
  if (digitalRead(8) == HIGH) {
    if (RFConnected == false ) {
      Serial.println("RF remote connected");
    }
    RFConnected = true;
  }
```

```
  if (RFConnected == true) {
    if (digitalRead(8) == LOW && digitalRead(10) == HIGH) {
      Serial.println("8 is on: move foward");
      digitalWrite(4, HIGH);
      digitalWrite(7, HIGH);
    }
    else {
      digitalWrite(4, LOW);
      digitalWrite(7, LOW);
    }
    if (digitalRead(9) == LOW && digitalRead(10) == HIGH) {
      Serial.println("9 is on: turn");
      digitalWrite(4, HIGH);
      digitalWrite(6, HIGH);
    }
    else {
      digitalWrite(4, LOW);
      digitalWrite(6, LOW);
    }
    if (digitalRead(10) == LOW) {
      Serial.println("10 is on: shift");
      digitalWrite(contServo1Switch, HIGH);
      if (digitalRead(9) == LOW){
        Serial.println("10 shift 9: move top servo down");
        continous1.write(130);
      }
      else if (digitalRead(8) == LOW){
        Serial.println("10 shift 8: move top servo up");
        continous1.write(80);
      }
      else if (digitalRead(8) == HIGH && digitalRead(9) == HIGH){
        continous1.write(90);
      }
    }
  }
}

void clawSetUp() {
  pinMode(A1, OUTPUT);
```

```
  digitalWrite(A2, LOW);
  digitalWrite(servoSwitch, HIGH);
  servo.attach(A1);
}

void clawActions() {
  if (digitalRead(11) == LOW) {
    Serial.println("pin 11 is on");
    servo.write(90);
  }
  else {
    servo.write(70);
  }
}
```

DATA COLLECTION METHODS

- Circuits checked for their functionality by using an indicator LED. When a current passes through the correct path, the desired LED will turn on.
- A multimeter is used to check the voltage, current, and resistance of the circuit. Useful when double checking the functionality of a circuit, and its safety. For example, making sure the resistors have the right values to reduce the current as not to burn out the connect module. Additionally, it can be used to check a component's battery usage, and thus its energy efficiency.
- Interactions between the Arduino and the radio frequency modules are checked by the serial monitor on the Arduino and indicator LEDs. When the input from the radio frequency transmitter matches the output of the radio frequency receiver, then they must be communicating correctly.
- A vernier calliper is used to check for correct dimensions.
- When the theoretical output matches the actual output, things are working.
- A timer and a tape measure are used to calculate speed.
- Coding wise, Serial.println() serves to monitor the code's progress and used for debugging.

PRODUCT TESTING & ANALYSIS

Mobility

To measure the efficiency of the wheels, their travel speeds on different surfaces are obtained.

Table 4: wheel speed

Surface	Distance (inches)	Time (seconds)				Speed (distance/average time)
		Trial 1	Trial 2	Trial 3	Average	
Carpet	54	9	8	9	8.667	6.23 m/s
Hardwood	61	10	8	7	8.333	7.32 m/s

Table 5: wheel speed in Mark I

	Time took to cover 2m in seconds on hardwood floor			
Wheel type	Trial 1 (seconds)	Trial 2 (seconds)	Trial 3 (seconds)	Average (seconds)
Thin wheels	10.60	11.79	12.31	11.57
Wide wheels	15.16	17.78	17.85	16.93

As shown in tables 4 and 5, the mecanum 4-wheeled drive in table four is 2 times faster than the drive in table 5.

RF controller

The RF controller is able to accomplish its goal. However, due to the battery being only 3V, a little decrease in its voltage over time would significantly hinder its performance (the controller has to be within a 1cm range of the receiver if the battery is at 2.8V; the controller basically stops working at 2.5V).

There is about a 1-second delay between command and action. This number varies on the battery power.

Continuous Servos on the Arm

When a module is attached to the arm, it visibly hinders the top continuous servo's performance. It appears that the top servo is not powerful enough.

Resolved issues from Mark I

- The servo is no longer twisting the arm out of shape.
- Mark II travels faster and better.
- Wheel axes are not crooked.
- There is enough power to power all motors as an external power supply is attached.
- More LEDs are added to places to monitor their status.
- The gesture controller is not the only control option since a remote is made.
- Parts are not falling off due to being hot glued.
- Simplified schematics by adding ICs.
- More precise parts = better efficiency. (3D printed parts around within 0.005 inches of design; Mark I hand-cut pieces are within 0.020 inches of design.)

Improvements from Mark I

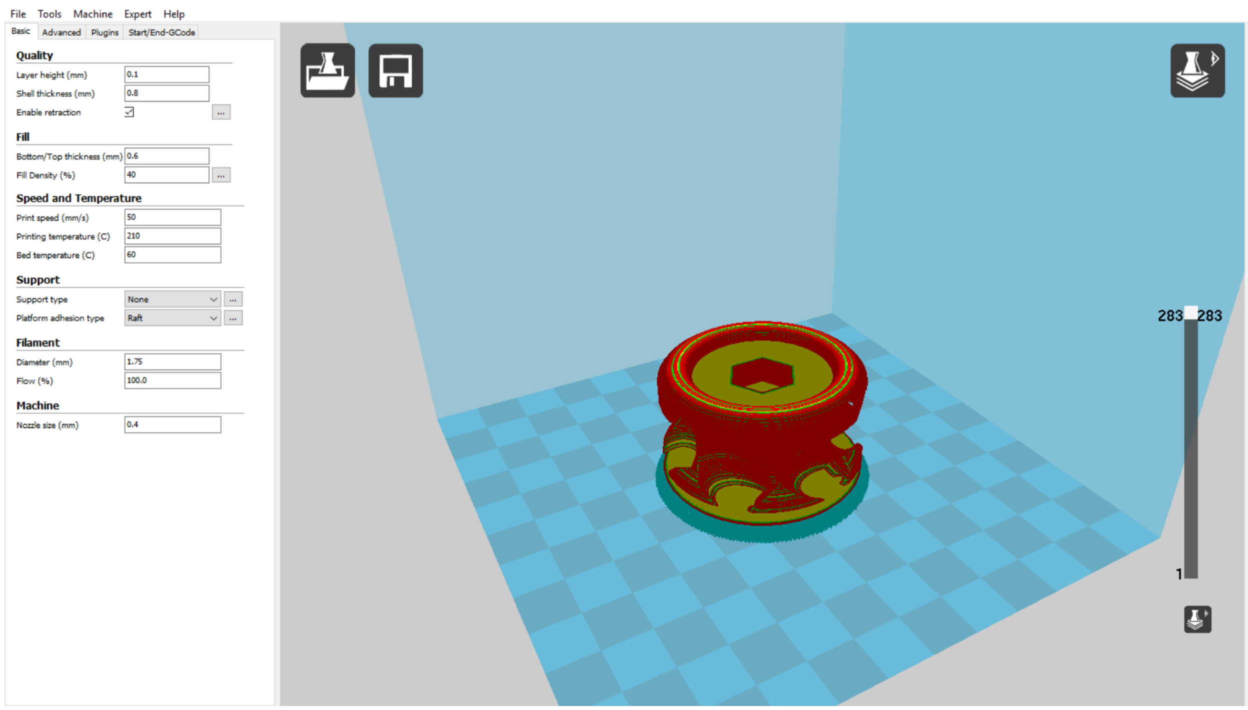
- Made it modulated.
    - Can now have a claw, phone holder, etc.
    - Added corresponding circuits and codes.
- Added servo to the bottom of the arm.
- Added additional joint at the bottom of the arm to increase mobility.
- Soldered on circuits.
- 3D printed precise components.
- Better structural integrity.
- Better wheels for better mobility.
- 4-wheel drive.
- Added casing for circuits.
- Improved overall efficiency.

MANUFACTURING

3D Printing

Most parts are 3D printed. The advantage is that they can be customized and made precisely to what is needed.

For the g-code, I used the program Cura.



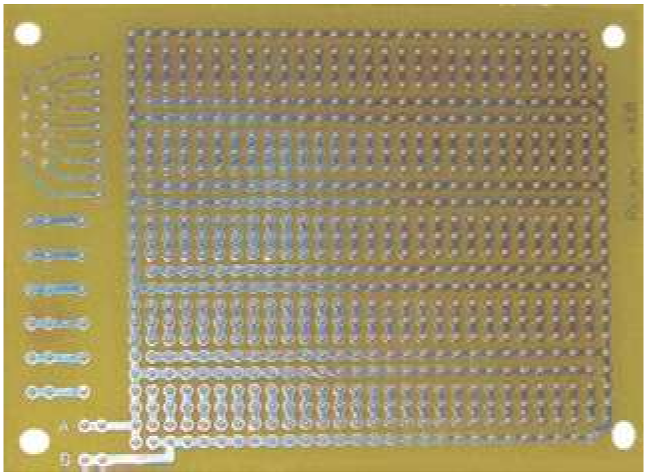
In this screenshot, the mecanum wheel was being in the process of being printed. (For this particular item, it was 3D printed and not bought because it's cheaper to 3D print.)

The obstacle to 3D printing besides being time-consuming is that the parts don't always stick to the bed. They like to detach mid-print, so, many times, the print ends up getting ruined mid-way. To solve this issue, I got Elmer's extra strength glue stick. Actually works really well. Would recommend.

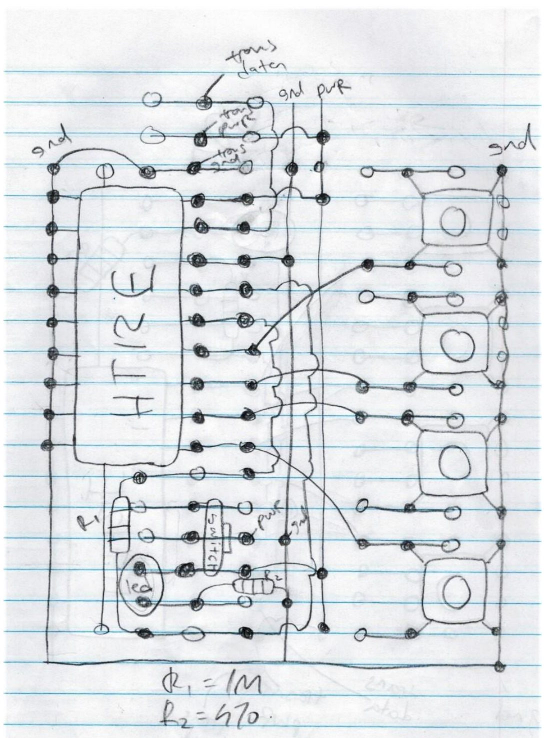
Most prints also have a raft platform and supports. These are important because they help the print to stay on the bed while printing and that it gets printed the way it's designed to.

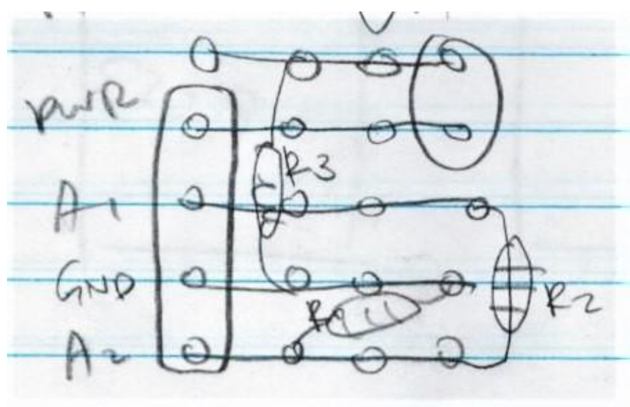
PCB

To solder the circuits onto a board, I bought PCBs to do the job. The template looks like this:



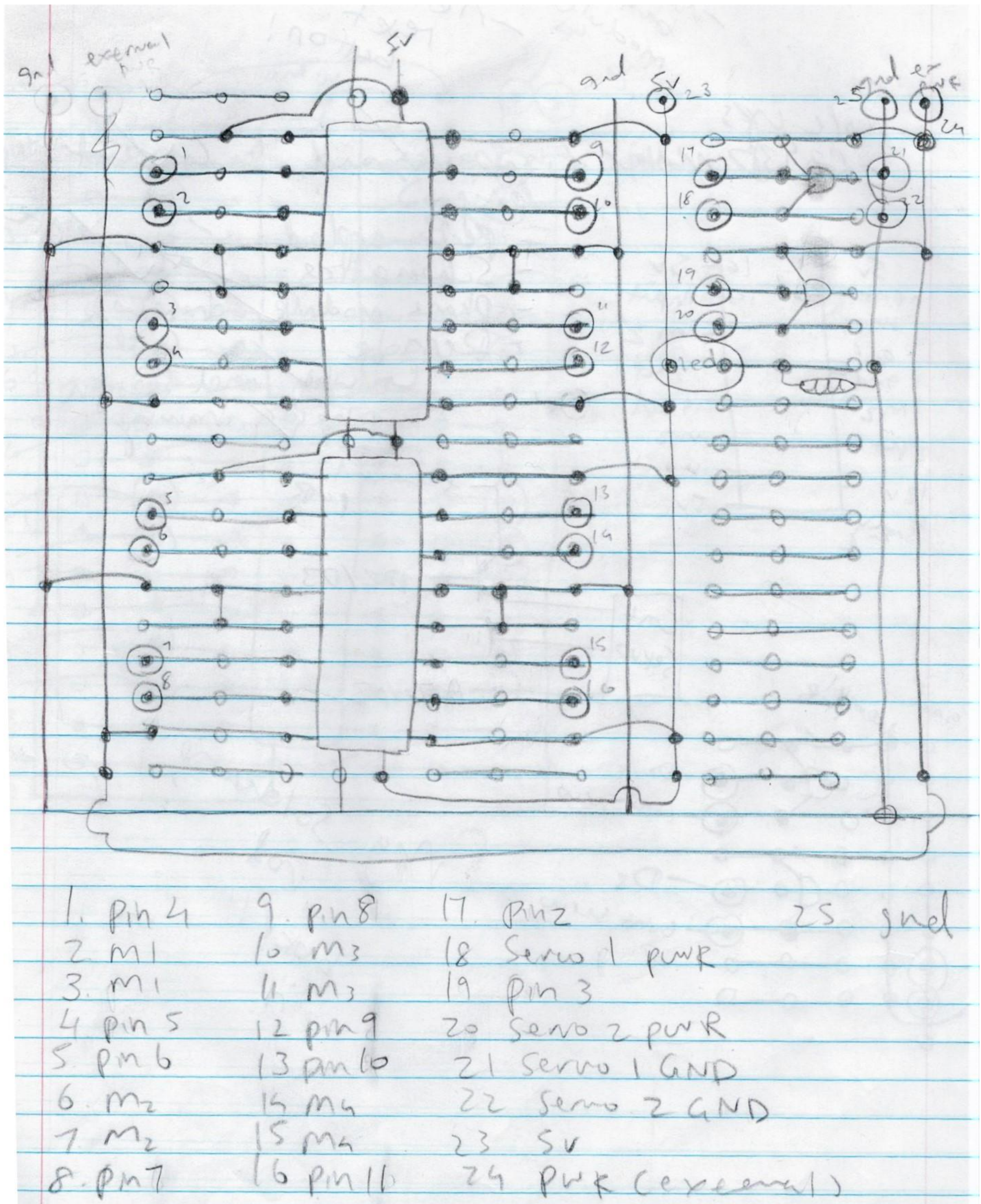
So, I cut pieces of the board big enough for each circuit and made diagrams to convert the schematics to actual soldered on circuits.



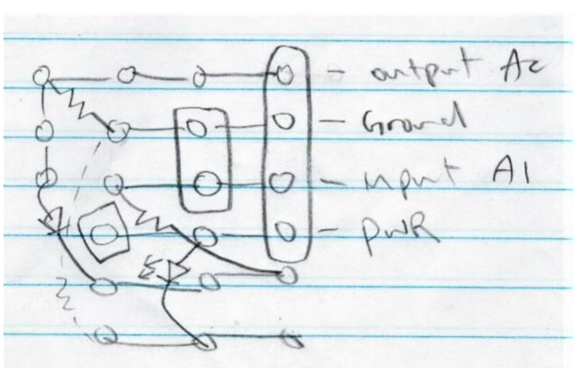


PCB 1: remote control with HT12E

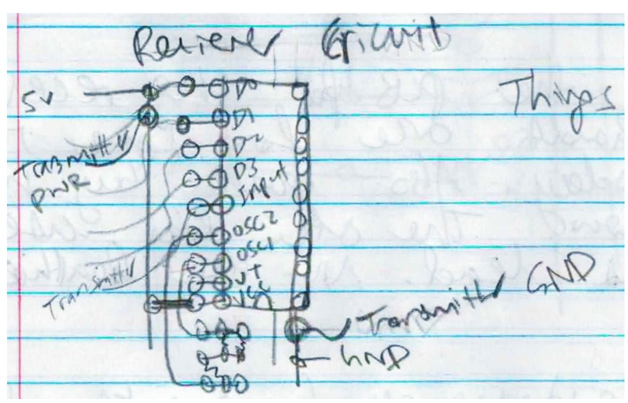
PCB 2: module 2



PCB 3: Motor Driver (L293D)



PCB 4: module 1



PCB 5: Receiver circuit

CONCLUSION

Mark II, in conclusion, is a modulated system that is designed to tackle the challenges of people's varying lives. Two modules are created to prove that the concept is possible and it is.

Module

Imagination is the limit when it comes down to the types of modules that can be created. From the basic claw to something that cleans up tables, these modules can be swapped and attached to the base. This allows Mark II to accomplish a variety of tasks and thus better achieve its goal of improving the efficiencies of our lives.

Remote

Mark II also adds a rectangular remote on top of the gesture controller. This way, people with difficulties such as carpal tunnel or people who simply prefer a rectangular remote are also able to enjoy the same range of control as the gesture controller allows.

Like Mark I, the remote communicates with the base via radio frequency, hence the remote is able to function across large distances and regardless of whether or not there are walls in between.

Structural

Being 3D printed PLA plastic, Mark II has much greater structural integrity than Mark I. Many of Mark I's problems are hence resolved. For instance, in Mark I, the acrylic plates on the arm twists if the claw is lifting up objects greater than 110g. This is no longer an issue in Mark II as the arm is designed in a c-channel shape.

Mark II also has better mobility since the wheels are mecanum wheels that are able to move in all directions. Mark II is also a four-wheel drive, meaning it can travel up ramps better and carry more weight than Mark I.

The circuits in Mark II are also soldered on, making them more user-friendly.

DISCUSSION

Sources of Error

The major set back and a regular in the sources of errors are bad connections. A soldered circuit board could function perfectly for one moment, and the next, it mysteriously stops working. This usually takes anywhere between 1 to 3 hours to find out the problem, and 99% of the time, it's a bad connection. For instance, the printed metal on the PCB could be peeling off, and hence, something is no longer connected. The difficulty with this is that it is invisible to the eye and thus hard to detect. In a large circuit, a single tiny bad connection is incredibly difficult to spot.

While 3D is awesome, there is a slight set-back up to it. The printed product is usually about 0.005 inches thicker than the design. For rougher things, this is no big deal, but for more precise parts, this has to be taken into consideration during the 3D modelling stage. Additionally, 3D printing is slow. It takes a really really long time, to print stuff.

On the bright side, most sources of error that appeared in Mark I was resolved here. For example, Mark I suffered greatly from a lack of structural integrity. The wheel axis in Mark I was not attached perfectly straight. So the wheels were at an angle, meaning they are not travelling at their best efficiency. In Mark II, the axis and a special holder were able to be 3D printed, so things were attached on much more precisely, eliminating many of these sources of error.

Shortcomings in the design

Despite the mecanum wheels being so much better than Mark I, there still cons to it. They are, for one, 3D printed, so as PLA plastic, they lack traction. While the existing traction is enough, if the terrain is a little rougher, it would experience trouble.

The base is a rectangular prism in shape - it can get stuck in corners. If Mark II decides to drive into a 90-degree corner, it can no longer turn. The only option left for it is to reverse. But if it must turn to travel to the destination, it becomes stuck in this spot. A solution to this would be to make the base circular. However, that is also harder to manufacture.

The continuous servo motors should also be replaced with regular servo motors. In Mark II, it is crucial to know the angle the arm is at. But for continuous servos, it doesn't tell you. The only way to know the angle is to know the speed at which the servos are travelling and for how long they have travelled. The only reason they are not regular servos is because I couldn't find any.

The RF remote only has four inputs/outputs. This limits the degree of control the user has. For instance, in Mark II, there is no button to travel sideways because there are not enough buttons. The solution to this would be getting more inputs/outputs. This can be done either with a different IC/RF pair, or by layering an HT12D/E pair to the existing HT12D/E pair. Another solution is to designate one of the buttons as a shift button.

Future Directions

- More modules can be created in the future for tasks.
- Ultrasound sensors can be added so it doesn't bump into things.
- Add a user-friendly interface so that the user, without any programming knowledge can set up a schedule for CODIA.
    - For example charge phone at 8 pm every day.
- Make CODIA's base round.
- Connect CODIA to an AI.
    - Voice command!
- Connect CODIA to the internet.
    - Access via phone or laptop, etc.
- Get CODIA to move towards a signal.
    - Useful if the user has trouble moving around.
    - This can be done through tracking the radio frequency the remote control transmits.
        - CODIA would have a directional receiver mechanism or two listening for the broadcast to determine the direction. The directional receiver can made out of a simple RF receiver by isolating it from all directions by a metal casing except for the direction it is to listen towards (as radio signal cannot pass through solid metal). This will determine the direction of the signal. Once the direction is found, the distance between the user and CODIA can be found by measuring the strength of the transmission.

APPLICATION

With the different modules, CODIA Mark II can be used in a variety of situations. Do you need something to hold your phone? Here's a module for that. Do you need something to pick things up? Here's another module for that. This variety and flexibility in its modules allow it to be implemented in many different situations.

On top of it, being modulated means that the user only has to buy the parts they need. They don't have to spend money on excess things. This might sound counter-productive for most markets as people are trying to get customers to buy as much as possible. However, as technology is already expensive, adding pieces the customers don't need just skyrockets the price. As any reasonable person would think, the product becomes too expensive and with too many unnecessary parts to purchase. It would never make it to the market. Whereas, if the user only has to buy what they want, a) the price becomes more manageable and b) it's money spent towards things the customer wants so it feels like money well spent. If the customer likes the product, they'll come back for more modules.

CODIA Mark II can also be pre-programmed. For example, is there is a complicated pathway CODIA has to take in a house, it can be pre-programmed to know the path so that it will travel through it automatically at the press of a button. This concept can be expanded to activities such as automatically charging your phone.

Last but not least, CODIA is not made for just a household environment. The goal is that it can improve the efficiencies of our lives everywhere. The modulated advantage comes into play again here as CODIA can easily be adapted to function at many different settings. For instance, it can be adapted to deliver items to hotel rooms via a tray module and a pre-programmed path; or it can be adapted to set up the computer lab every morning through a different module.

Imagination is the limit when it comes down to the possibilities and application of the modules.

ACKNOWLEDGEMENT

I would like to thank my parents for their never-failing faith in me, I would like to thank my friends for always being there to be counted on, and I would like to thank my teachers for supporting me all the way through.

BIBLIOGRAPHY

Average Household Income. (2018, November). Retrieved from https://www.oakville.ca/dashboard/economic-growth-household-income.html

Disabled World. (2017, May 12). *Canadian Male and Female Life Expectancy Rates by Province and Territory.* Retrieved from https://www.disabled-world.com/calculators-charts/ca-lifespan.php

Fottrell, Q. (2018, August 4). *People spend most of their waking hours staring at screens.* Retrieved from https://www.marketwatch.com/story/people-are-spending-most-of-their-waking-hours-staring-at-screens-2018-08-01

Guest. (2015, October 27). *Driving Mecanum Wheels Omnidirectional Robots.* Retrieved from https://www.roboteq.com/index.php/applications/applications-blog/entry/driving-mecanum-wheels-omnidirectional-robots

MacNaughton, L. (2016, June 19). *Goodbye Freedom 55: More Canadians spend their retirement years working.* Retrieved from https://www.cbc.ca/news/business/canada-retirement-age-rising-1.3638733

Microcontroller Projects. (n.d.). *L293 and L293D h bridge motor driver Ic Pin out and Working.* Retrieved from https://www.microcontroller-project.com/l293d-pin-description-and-working.html

P. Alexandrea. (2019, February 23). *3D Printer Materials Guide: 3D Printing Plastics.* Retrieved from https://www.3dnatives.com/en/plastics-used-3d-printing110420174/

Post-Secondary Education. (2018, November). Retrieved from https://www.oakville.ca/dashboard/economic-growth-education.html

Statistics Canada. (2017a, February 15). *A profile of persons with disabilities among Canadians aged 15 years or older, 2012.* Retrieved from https://www150.statcan.gc.ca/n1/pub/89-654-x/89-654-x2015001-eng.htm

Statistics Canada. (2017b, September 13). *Household income in Canada: Key results from the 2016 Census.* Retrieved from https://www150.statcan.gc.ca/n1/daily-quotidien/170913/dq170913a-eng.htm

Statistics Canada. (2018, July 23). *Living arrangements of seniors.* Retrieved from https://www12.statcan.gc.ca/census-recensement/2011/as-sa/98-312-x/98-312-x2011003_4-eng.cfm

The advantages of Omni-directional wheels. (2017, January 6). Retrieved from https://www.dynamotousa.com/blogs/news/the-advantages-of-omni-directional-wheels

Technik3k. (2018, May). Omni wheel vs Mecanum. Retrieved from

https://www.vexforum.com/t/omni-wheel-vs-mecanum/48151/12

Texas Instruments. (2016, January). *L293x Quadruple Half-H Drivers*. Retrieved from

http://www.ti.com/lit/ds/symlink/l293.pdf

Vectoring robots with Omni or Mecanum wheels. (n.d.). Retrieved from

https://www.superdroidrobots.com/shop/custom.aspx/vectoring-robots/44/