

GNG 2101

Design Project User and Product Manual

Remote Braking System for a Pediatric Gait Trainer

Submitted by:

BRAKEThrough Solutions (A08)

Matt White, 300165613

Elsa Lange, 300168393

Tahmeed Khan, 300177400

Brad Cole, 300172151

December 5th, 2021

University of Ottawa

Table of Contents

1	Introduction	5
2	Overview	5
2.1	Conventions	6
3	Getting started	7
3.1	Set-up Considerations	7
3.2	User Access Considerations	7
3.3	Activating the System	7
3.4	System Operation	8
3.5	Exiting the System	8
4	Using the System	9
4.1	Mounting System	9
4.2	Bluetooth Remote and Braking Mechanism	9
5	Troubleshooting & Support	11
5.1	Error Messages or Behaviors	11
5.2	Special Considerations	11
5.3	Maintenance	11
5.3.1	Maintenance of the Components	11
5.3.2	Maintenance of the Cases	12
5.4	Support	12
6	Product Documentation	12
6.1	Remote Control	13
6.1.1	BOM (Bill of Materials)	13
6.1.2	Equipment list	14
6.1.3	Instructions	14
6.2	Linear Actuator Circuit	17
6.2.1	BOM (Bill of Materials)	17
6.2.2	Equipment list	19
6.2.3	Instructions	19
6.3	Mounting System	21
6.3.1	Aluminum Mounting Bracket	21
6.3.2	PLA Circuit Casing	23
6.3.3	Mounting System And Circuitry Case B.O.M	24
6.3.4	Equipment Used	24
6.4	Testing & Validation	24
7	Conclusions and Recommendations for Future Work	25
8	Bibliography	26
9	APPENDIX I: Design Files	26

List of Figures

Figure 1a) Braking and Attachment and 1b) Remote Control	6
Figure 2a) Brake Module On/Off Switch and 2b) Remote On/Off Switch	8
Figure 3) Aluminum Mount	9
Figure 4) Bluetooth Remote Control	10
Figure 5) Block Diagram of Component Relationships	12
Figure 6) Arduino Bluetooth Module Connections	15
Figure 7) Fritzing Diagram of the Remote Control Circuit	15
Figure 8) Arduino Code for Remote Control	16
Figure 9) Finished Remote Control	17
Figure 10) Arduino Motor Controller Pins	19
Figure 11) Arduino Bluetooth Module Connections	20
Figure 12) Linear Actuator Control Code	21
Figure 13. a) and 13. b) Mounting Bracket Blueprint	22
Figure 14) Mounting Bracket	22
Figure 15) Assembled Mount	23
Figure 16) PLA Casing Components	23

List of Tables

Table 1. Acronyms	4
Table 2. Glossary	4
Table 3. BOM for Circuitry Components	13
Table 4. BOM for Braking Module	17
Table 5. BOM for Mounting Subsystem	24
Table 6. Quantitative and Results of the Testing Done on the Final Prototype	24

List of Acronyms and Glossary

Table 1. Acronyms

Acronym	Definition
FSR	Force-Sensitive Resistor: A Pressure-Sensitive Button component.
M-M, F-F, M-F	Male-Male, Female-Female, Male-Female. Describe the connectors of solderless wires.

Table 2. Glossary

Term	Acronym	Definition
Force Sensitive Resistor	FSR	A pressure-sensitive button.
Gait Trainer	Not Applicable	Device similar to a walker, but allows the user to walk independently.
Linear Actuator	Not Applicable	A strong, accurate piston capable of exerting great force.
Bluetooth	Not Applicable	A wireless communication tool used to send data between the remote and the brakes.

1 Introduction

The goal of this document is to instruct future students of GNG 2101 on the process of designing, refining, and eventually manufacturing a remote braking system for a pediatric gait trainer. The following process can be manipulated and changed to fit various models and makes of accessibility devices that work on wheel based movement. The examples and processes which are explained here were designed for an R82 Crocodile Gait Trainer. Furthermore, the initiation, planning, troubleshooting, and documentation procedures involved in the design and manufacturing of this product will be explained in detail. Lastly, a step by step walkthrough of the proper functionality of the device will be discussed. For reasons of security and/or privacy, no images, descriptions, or identifying information of the client is included in this manual, or should be included in any future versions.

2 Overview

The client currently uses the R82 Crocodile gait trainer for his 7-year-old child. The R82 Crocodile only has manual brakes on the back wheels. Consequently, the client must be close to their child at all times so they can engage the brakes needed which limits their independence and could be potentially dangerous in emergency situations. In Canada alone, there are 1.1 million people who use walking aids¹. The same problem can be applied to anyone who needs a guardian or caretaker to manually engage their brakes.

The product is created for users who need a safe and reliable braking system that can be activated remotely so the walking aid user can walk independently. The remote braking system is portable and can be easily detached from the walking aid. Additionally, it can be used in a variety of common walking environments.

Currently, there are no remote braking systems on the market. Furthermore, the force of the linear actuator and the force-sensitive braking are the key aspects of our product that make it better. The linear actuator that is used in the product has a force of 750 N which is much more force than a standard brake can apply. Moreover, the remote has 4 different force levels using a force sensitive resistor. As a result, different force levels can be used in various situations. For instance, the remote control user would not apply a force level of 4 if the walking aid user was going up a hill.

¹<https://www150.statcan.gc.ca/n1/pub/89-654-x/89-654-x2013002-eng.pdf>

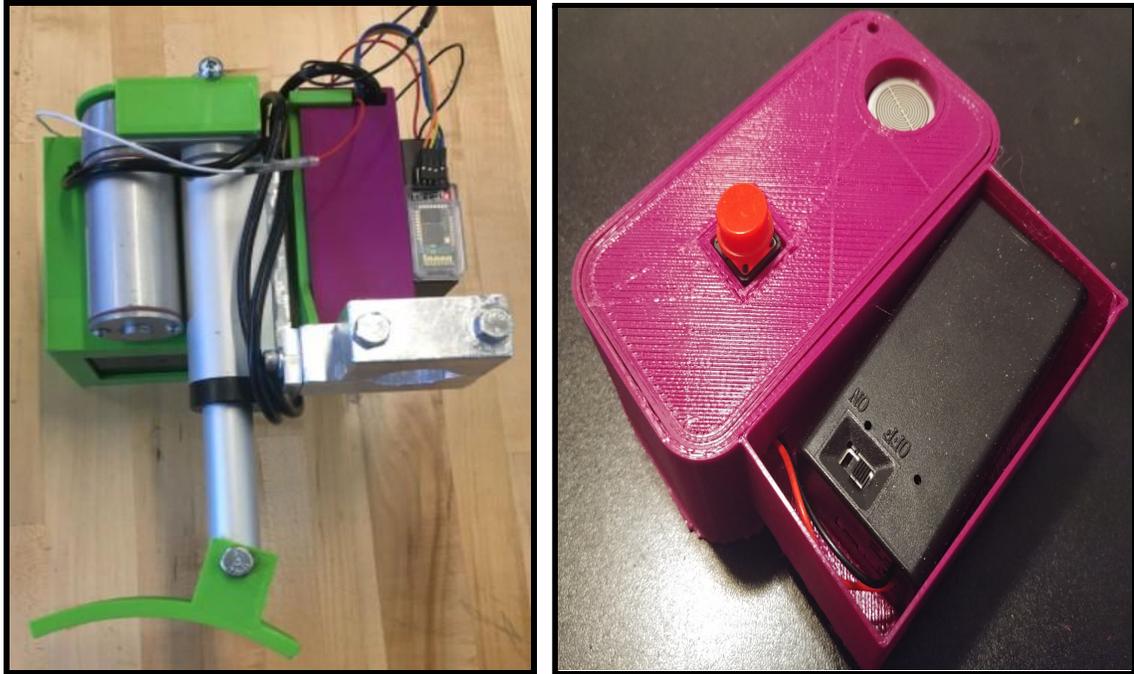


Figure 1a) Braking and Attachment and 1b) Remote Control

The figures 1a and 1b above are the braking and attachment subsystems and the remote control. The machined clamp attaches to the post of the gait trainer using the screws and a wrench. Using the circular button on the remote, the linear actuator and brake shoe will extend until its full stroke. Pressing the big red button will make it return to its original position. Depending on how hard the force sensitive resistor is pressed the linear actuator will extend at 4 different speeds.

The attachment subsystem is an aluminum clamp that has been machined. It also contains a vertical bar which the battery box attaches to. Attached to the linear actuator are 3D printed parts that hold (from left to right) the 12V battery in the back, an Arduino nano circuit (microcontroller and breadboard) that controls the linear actuator with a bluetooth and a 9V battery. Inside the battery there is an Arduino nano and circuit. It also includes the big red button, force sensitive resistor and 9V battery.

2.1 Conventions

None.

2.2 Cautions & Warnings

Currently, there is no indication of the battery power. Before using the system, especially if in emergency situations, ensure that there is enough power for the prototype to work.

Additionally, the prototype would not work in certain weather conditions, during snowstorms for instance.

3 Getting started

3.1 Set-up Considerations

Considerations that must be taken into account before the braking system can be operated are as follows...

Do you have all of the necessary components?: Included in this kit should be a linear actuator powered brake module with an aluminium mount attached, as well as a separate hand held remote. The brake module should be fastened firmly (so it cannot move) to the leg of the gait trainer, and the remote should be operated by a person not using the actual gait trainer.

Do both components have batteries installed?: A 9V battery must be installed in both the battery box of the brake module and the remote control.

3.2 User Access Considerations

The product is intended to have two users while in use: the person operating the gait trainer and the person remotely operating the remote braking system. The user operating the gait trainer will generally have a physical impairment which requires them to use a gait trainer to walk. The user operating the remote braking will generally be the guardian or caretaker of the person operating the gait trainer. Since the system is relatively simplistic, no aspect of the system is explicitly hidden from any user of the product, besides the encapsulation of the electronics and communication devices. In most use cases, the user of the gait trainer does not require access to the remote control, and the gait trainer is separated from the remote control from a distance of up to 20 m. Any user has access to removing and replacing any of the three batteries.

3.3 Activating the System

To activate the remote brake system the user must first power on both the brake module and the remote control. To do this, power switches can be found on both components, one being on the operating face of the controller, and the other being found on the circuit box of the brake module (see images below).

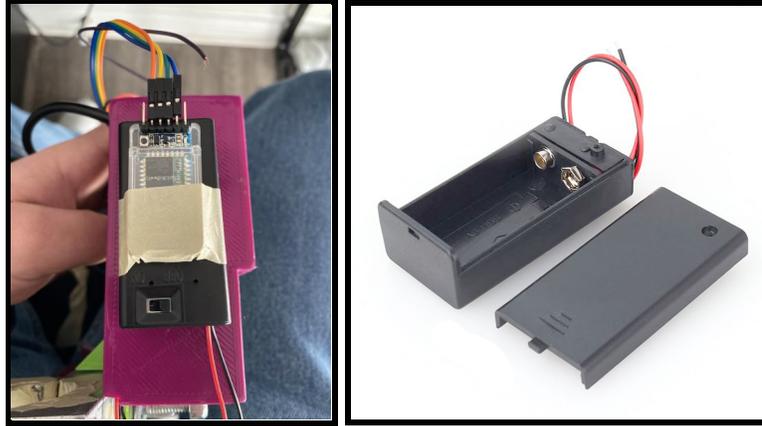


Figure 2a) Brake Module On/Off Switch and 2b) Remote On/Off Switch

Once the remote and brake module are powered on, a moment should be given for the two components to make a bluetooth connection. Before use, it is suggested that a test trial of the brake is done to confirm this connection (see below for operating instructions).

3.4 System Operation

General Operation

As suggested above, before fully utilizing the remote braking system, a functionality test should be done. To do this first power on both the remote and brake module (see above). Secondly, with the trainer immobile, press and hold down on the large circular button on the top of the remote control. This should begin to activate the brakes linear actuator, which has the ability to vary its speed in accordance with how hard the button is pressed and held. The harder the button is pushed, the faster the brake engages.

Once the brake pad has fully engaged with the surface of the wheel, gently attempt to push the gait trainer back and forth. If the wheel of the trainer rotates while the brake is engaged, shut both components off and readjust the brake module closer to the wheel, and repeat the above. When the brake is adjusted such that the wheel is fully immobilized, the brake can now be disengaged. To disengage the brake, the small square button on the bottom of the remote can be pushed, which will fully retract the brake. The gait trainer should now move unhindered by the brake. The system should now be ready for use.

3.5 Exiting the System

To fully exit the system, first switch off the controller. The power switch for the controller is located on the 12V battery box inside of the battery compartment, with clearly indicated ON and OFF positions. Next, switch off the braking mechanism. The power switch for the braking mechanism is located on the 12V battery box, attached to the side of the electronics

housing. If both 9V battery boxes are in the OFF state, and there are no visible lights, sounds, or movements, then the system has been exited and is no longer consuming battery power.

4 Using the System

4.1 Mounting System

The mounting subsystem is the subsystem that attached the braking subsystem to the gait trainer. It is depicted in the **figure** below.



Figure 3) Aluminum Mount

The subsystem contains two aluminum components. The first component is the clamp that will attach to the pole of the gait trainer. The drilled hole is where the hole will be placed so the subsystem goes around the pole. The clamp can attach anywhere on the gait trainer as long as it is secured in place using a wrench. To remove the wrench use a $\frac{7}{8}$ inch wrench and then secure it where the clamp should be on the pole. The vertical bar will attach to the linear actuator and battery box. To attach the linear actuator with the mounting subsystem use two screws and a wrench to secure them.

4.2 Bluetooth Remote and Braking Mechanism

The braking mechanism works by protecting and retracting the linear actuator mounted onto the gait trainer. After the brakes have been mounted, one is ready to use the braking mechanism. Engaging the brakes means protracting the linear actuator downwards until the brake shoe attached to the tip of the linear actuator reaches the tip of the wheel. Disengaging means retracting the linear actuator brakes upwards until it has reached its fully unextended position. Both the bluetooth remote and braking system have an on-off switch, so **both** must be turned on to activate the braking mechanism. The lights on the bluetooth modules (both sender and receiver) will blink twice once they connect to each other, though you only need to look at the light of the bluetooth module attached to the linear actuator casing.



Figure 4) Bluetooth Remote Control

To engage the braking mechanism, press down on the white/black top button **as shown in the following figure**. The system is made to detect four different levels of braking, from a soft brake to a very hard brake. How hard the button is pressed corresponds to one of the four braking levels which correspond to one of the linear actuator protractions. In emergency situations, people are more likely to use the hard braking functions by pressing hard on the button, though a soft brake is possible. Not pressing any button will leave the linear actuator at its same position. Pressing the red button disengages the brake and retracts the linear actuator to the unextended position at linear actuator speed (~10 mm/s). The braking mechanism itself does not offer any manual braking control.

5 Troubleshooting & Support

5.1 Error Messages or Behaviors

The two most common error behaviours are a battery power and connection problem. Firstly, ensure that all the connections are appropriately connected. This error is the most common error behaviour. If all the connections are correct, then it may be a battery problem. There is no indication when the battery power is lost, so try testing the batteries with other electronic components.

5.2 Special Considerations

Firstly, since the two 9V batteries and the 12V battery are not rechargeable, they will have to be replaced every time it runs out of battery. Keeping track of how many hours the batteries have been in use will reduce the chance of failure in an emergency situation.

It is recommended that the circuitry should be checked every time the product is used to ensure that the connections are correct. Another precaution would be to avoid unfavourable weather conditions as the product may fail.

Finally, to ensure that the bluetooths are connected correctly, there is a light that will blink twice, slowly, repeatedly. If the bluetooths are disconnected the light will flash repeatedly and quickly. If connection is lost, the light will flash slowly.

5.3 Maintenance

The maintenance of the prototype can be divided into the maintenance of the components and its cases.

5.3.1 Maintenance of the Components

The electrical and mechanical components must be maintained with regular maintenance. After a certain amount of use, the electrical components such as the microcontrollers will need to be replaced with new ones. The jumper wires used in the circuits should be replaced regularly as the product will not work with misshapen wires. The brake should be filed/deburred to ensure the plastic brakes appropriately. Avoid general contact with water and other unfavourable weather conditions.

5.3.2 Maintenance of the Cases

To maintain the cases, one should ensure that the cases are cleaned regularly. Additionally, reprinting the cases is recommended once the cases become damaged or old. Avoid general contact with water and other unfavourable weather conditions.

5.4 Support

The first step is to try to resolve the problem without outside assistance. In all likelihood, it is a connection and/or battery power problem.

Please refer to the MakerRepo for this project [here](#) where there are more files to search through to find a solution. If the problem persists, email brakethroughsolutions@gmail.com and we will reply at the earliest convenience.

6 Product Documentation

Our remote braking system is split up among two different subsections including the braking mechanism and the remote control. The braking mechanism includes the linear actuator, the aluminum mount, the arduino nano attached to a motor controller and 9v and 12 batteries. The remote control features an arduino nano, pushbutton, 9v battery and a force sensitive resistor. Both subsystems are made under 3D printed encasing so the components are protected. The following block diagram should detail relationships between each remote braking subsystem.

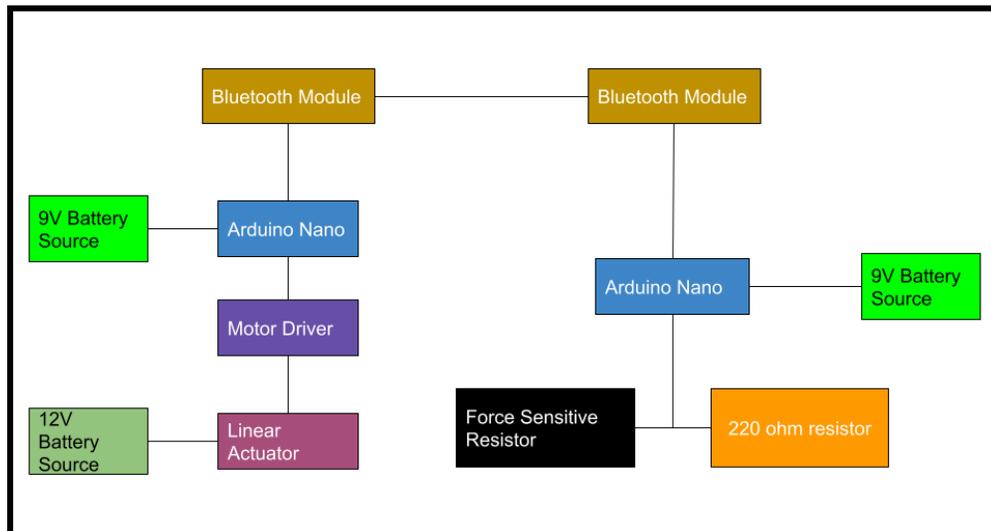


Figure 5) Block Diagram of Component Relationships

Along with the remote control subsystem, a mounting subsystem was developed. This subsystem connected and held together all components of the brake module and circuitry, as well as connected it to the gai trainer itself. The production process of this subsystem can be seen in section 6.4.

6.1 Remote Control

6.1.1 BOM (Bill of Materials)

Table 3. BOM for Circuitry Components

Item	Description	Quantity (Approx.)	Price	Source
9V Battery	It's a battery!	1	\$7.50	Makerstore https://www.canadiantire.ca/en/pdp/duracell-copper-top-alkaline-9v-batteries-4-pk-0650829p.html#srp
Bluetooth HC-05	Allows for bluetooth communication using Arduino	1	\$12.99	MakerLab
Arduino Nano	Used for logic and communication.	1	\$16.00	MakerLab
Pushbutton	Digital Button	1	Free	Owned by multiple Team Members
Pressure Sensitive Button	Analog push button dial for braking force control.	1	\$6.95	SparkFun
Arduino Mini breadboard	Used for the connections to the Arduino Nano	1	\$1.50	https://edu-makerlab2021.odoo.com/shop/product/breadboard-53?search=breadboard#attr=223
Arduino Software	Used for coding the Arduino Nano	1	Free	https://www.arduino.cc/en/software
Arduino Casing	Protects arduino	1	Free	3D Printed in L'Abbe
Controller Casing	Protects arduino in controller	1	Free	3D Printed in L'Abbe
Total			\$44.94	

6.1.2 Equipment list

1. Soldering Iron + Solder
2. Multimeter
3. Electrical Tape
4. Double Sided Tape
5. File
6. Mini USB to USB cable
7. M-M Header Solderless Wires
8. F-F Header Solderless Wires
9. M-F header Solderless Wires
10. 3-D Printing Equipment from MakerSpace

6.1.3 Instructions

Step 1) Electronic Components

First, place the Arduino Nano into the center of the mini-breadboard. There are four (4) main components to consider while assembling the electronics of this subsystem: the Battery Pack, the FSR, the Pushbutton, and the HC-05 Bluetooth Module.

Battery Pack

- Connect the red “power” wire from the battery pack to the V_{in} pin row on the breadboard.
- Connect the black “ground” wire from the battery pack to the GND pin row on the breadboard.

Pushbutton

- Connect one of the pushbutton pins to the 5V pin row on the breadboard.
- Connect the corresponding paired push button pin to the D5 pin row of the breadboard.
- Make a breadboard connection, using a 10 k Ω resistor, from the D5 pin row to the ground pin row of the breadboard.

FSR (Force-Sensitive Resistor)

- Connect the bottom (or left, pins facing toward) FSR pin to the 5V pin row on the breadboard.
- Connect the other FSR pin to the A0 pin row of the breadboard.
 - NOTE: The FSR pins are very small, secure above connections using electrical tape!
- Make a breadboard connection, using a 10 k Ω resistor, from the A0 pin row to the ground pin row of the breadboard.

Bluetooth Module

- Connect the pin labelled VCC to the 5V pin row on the breadboard.
- Connect the pin labelled GND to the GND pin row on the breadboard.
- Connect the pin labelled TXD to the RX0 pin row on the breadboard.
- Using 3 x 1 k Ω resistors, create the voltage divider circuit shown below, to supply less voltage to the Arduino’s TX1 pin. Note that the black wire is the ground wire. Use this to connect the pin labeled RXD to the TX1 pin row on the breadboard.

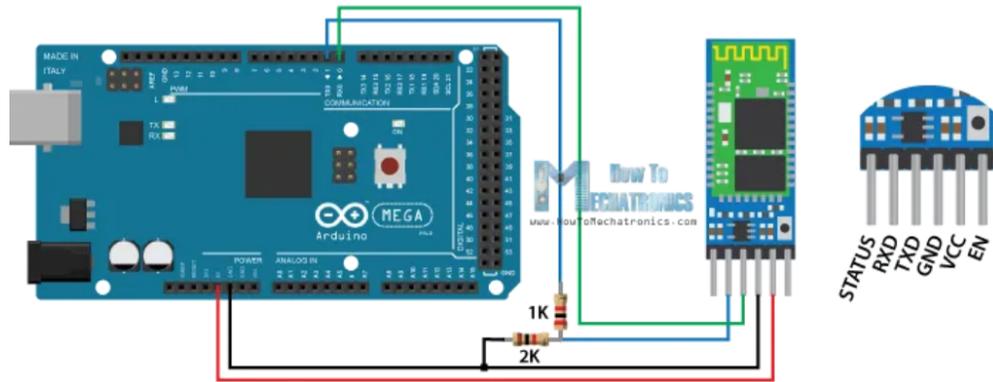


Figure 6) Arduino Bluetooth Module Connections

The electrical component is now finished. To test the circuit, insert a 9V battery into the 9V battery holder, and flick the switch ON. If the Arduino lights up and the Bluetooth module begins to blink quickly, the electronics are functioning correctly. The circuit should look similar to the figure below:

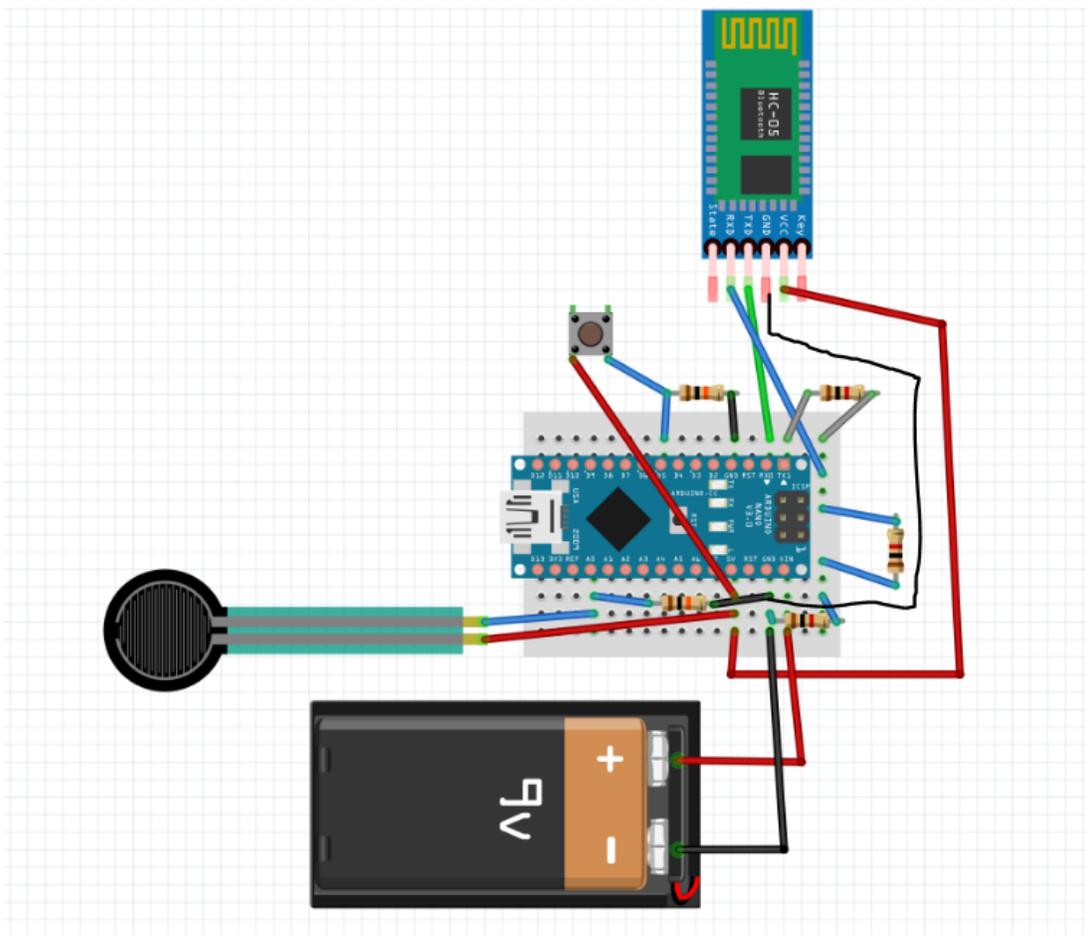


Figure 7) Fritzing Diagram of the Remote Control Circuit

Step 2) Software/Wireless Component

Arduino

- Plug the Arduino into a computer running the Arduino IDE software. Ensure that the system powers up correctly.
- Download the Arduino code off of MakerRepo. The controller code should look as follows:

```
int pressureTier;
int prevPressureTier;
int pressureChart[] = { 500, 900, 950, 975 };

void setup() {
  Serial.begin(38400);
  pinMode(5, INPUT);
  pressureTier = 0;
  prevPressureTier = -1;
}

void loop() {
  int data = analogRead(A0); // The FSR reading.
  if (digitalRead(5) == HIGH) pressureTier = 0;

  if (data >= pressureChart[0] && data < pressureChart[1] && prevPressureTier < 1) {
    pressureTier = 1;
  } else if (data >= pressureChart[1] && data < pressureChart[2] && prevPressureTier < 2) {
    pressureTier = 2;
  } else if (data >= pressureChart[2] && data < pressureChart[3] && prevPressureTier < 3) {
    pressureTier = 3;
  } else if (data >= pressureChart[3] && prevPressureTier < 4) {
    pressureTier = 4;
  }

  if (pressureTier != prevPressureTier) {
    Serial.write(pressureTier);
  }

  prevPressureTier = pressureTier;
}
```

Figure 8) Arduino Code for Remote Control

- Upload the Arduino code to the Nano board.
 - **NOTE: BEFORE UPLOADING ANY CODE, UNPLUG THE RX AND TX CONNECTIONS IN THE ELECTRONICS.**
- Restore any connections that have been severed in the process. Unplug the Arduino from the computer.

Step 3) Physical/3D-Printing Component

- 3D-print both the case and the lid using the CAD files from MakerRepo.

- Check if the lid fits snugly on top of the case. If not, sand or file down the lid until it does.
- Place the circuitry into the case. Ensure that the battery holder's wires pass through the hole in the side of the casing, and that all wires and components fit into the case.
- Using double sided tape, ensure that the FSR is fastened securely to the top of the pillar, and that the push-button is securely attached to the lid divot.
- Place the lid on top of the case. If need be, use tape to hold the lid in place.

The remote control subsystem should now be fully assembled and operational. It should look like the image below.



Figure 9) Finished Remote Control

6.2 Linear Actuator Circuit

6.2.1 BOM (Bill of Materials)

Table 4. BOM for Braking Module

Item	Description	Quantity (Approx.)	Price	Source

Linear Actuator/Push Pull Solenoid	Used to engage and disengage brakes.	1	\$40	Makerstore
9V Battery	It's a battery!	1	\$7.50	Makerstore https://www.canadiantire.ca/en/pdp/duracell-copper-top-alkaline-9v-batteries-4-pk-0650829p.html#srp
12V Battery	Again, it's a battery!	1	\$35.99	Amazon https://www.canadiantire.ca/en/pdp/mk-battery-12-volt-7-2-ah-sla-battery-0102038p.html#srp
Arduino Nano	Used for logic and communication.	1	\$16.00	MakerLab
Stepper Motor Driver	Used for interfacing the nano with the linear actuator	1	\$13.64	https://edu-makerlab2021.odoo.com/shop/product/stepper-motor-driver-13#attr=281
Arduino Mini breadboard	Used for the connections to the Arduino Nano	1	\$1.50	https://edu-makerlab2021.odoo.com/shop/product/breadboard-53?search=breadboard#attr=223
Arduino Software	Used for coding the Arduino Nano	1	Free	https://www.arduino.cc/en/software
Arduino Casing	Protects arduino	1	Free	3D Printed in L'Abbe
Controller Casing	Protects arduino in controller	1	Free	3D Printed in L'Abbe
Total			\$114.63	

6.2.2 Equipment list

1. Soldering Iron + Solder
2. Multimeter
3. Electrical Tape
4. M-M Header Solderless Wires
5. F-F Header Solderless Wires
6. M-F header Solderless Wires
7. 3-D Printing Equipment from MakerSpace
8. Mini-USB Cable

6.2.3 Instructions

This is the circuit that controls the speed of the linear actuator and receives the signal from the remote. This circuit makes extensive use of a motor controller that controls the speed of the linear actuator. A tutorial was mostly used to configure the connections between the linear actuator and the motor board [2]. The circuit can't be remade in Fritzing due to the motor controller not being available on Fritzing. In order for the circuitry to work, there must be a common ground between all of the components, so the ground for both the motor driver and the bluetooth module were connected to the ground pin of the arduino nano. Our implementation rested an arduino nano and all of its connections on a mini-breadboard except the motor driver. We created a casing to protect the motor driver and the arduino nano, which is further explained in the PLA section. The Solidworks file for the casing can be found in the makerepo.

Motor Driver

The setup for the motor driver is illustrated with the **figure below**:

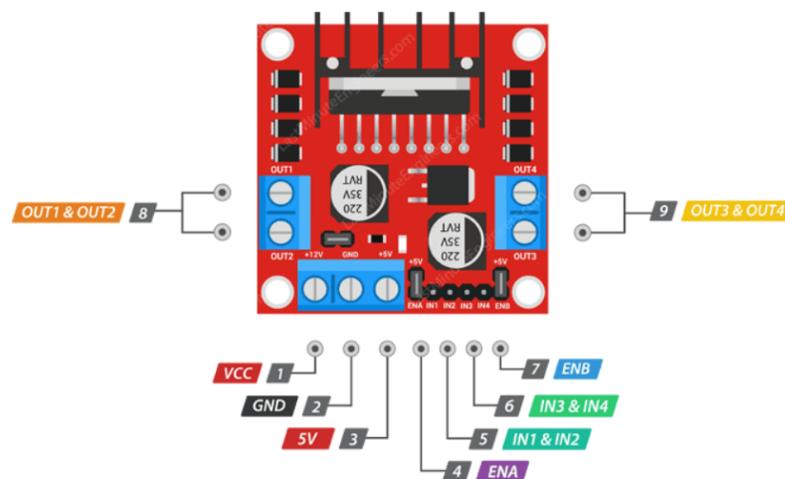


Figure 10) Arduino Motor Controller Pins

1. Vcc is connected to the positive terminal (M-M connection) of the 12v battery. This pin will be used to power the linear actuator. You can secure the free end of this pin to the positive terminal using electrical tape. You will need an extension M-F header for the cord to reach the terminal.
2. The GND is connected to the arduino's ground pin (M-M Connection).
3. The 5V pin is connected to the Arduino's 5V pin (M-M Connection).
4. The ENA pin is used to control the speed of the motor A. In our implementation, we used motor output B, and thus used pin groups 6, 7, and 9 to control the speed and direction of the linear actuator.
5. Unused group of pins
6. IN3 and IN4 control the direction of the linear actuator (M-F connection). These are connected to D5 and D6 pins on the arduino nano.
7. The ENB controls the speed of the linear actuator and should be connected to the D3 Pin on the nano. It is critical that the ENB pin is connected to a PWM signal pin so that the speed of the linear actuator can be mapped to the strength of the PWM signal. D3 works perfectly for our case.
8. The output pins for motor 1. This is not used in our implementation. It is important to note that these pins can be used for the linear actuator, but then pin group 4 must be used for the D3 connection while pin group 5 uses the D5 and D6 connections.
9. The output pins for motor 2. The positive lead of the linear actuator should be attached to OUT4 while the negative lead is attached to OUT 3.

Once all of these connections are finished, the setup for the motor driver is complete.

Bluetooth Module

The bluetooth module can be implemented in a similar fashion to the remote control where the TX of the bluetooth module is connected to the RX of the nano. The RX of the bluetooth module must be connected to the arduino common ground with a 2 k Ω pullup resistor and a 1 k Ω pullup resistor to the TX. The Vcc pin should be connected to the 5V pin and the GND pin should be connected to the common ground. These connections are illustrated with the Arduino Uno below (the connections should be identical to the nano).

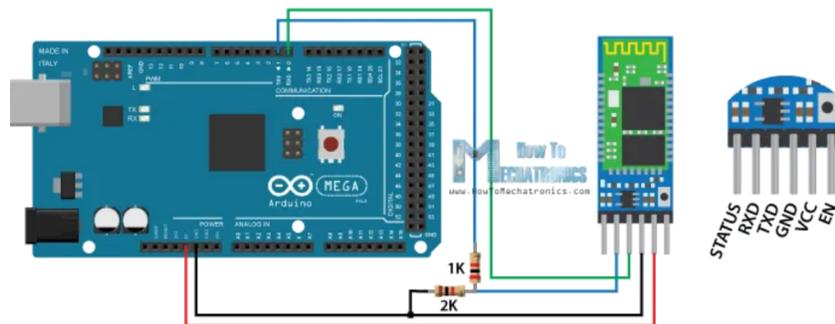


Figure 11) Arduino Bluetooth Module Connections

Arduino Code

```
#define enB 3
#define IN3 5
#define IN4 6

int oldData;

void setup() {
  // put your setup code here, to run once:
  pinMode(enB, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  oldData = 0;
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
  Serial.begin(38400);
}

void forward() {
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}

void backward() {
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}

void disengage() {
  backward();
  analogWrite(enB, 255);
  delay(800);
  analogWrite(enB, 0);
  forward();
}

void loop() {

  if (Serial.available() > 0)
  {
    int newData = Serial.read(); // reading the data received from the bluetooth module
    Serial.println(newData);
    if (newData == 0) {
      if (oldData != 0) disengage();
    } else {
      analogWrite(enB, 63 + (48 * newData));
    }
    oldData = newData;
  }
}
```

Figure 12) Linear Actuator Control Code

6.3 Mounting System

6.3.1 Aluminum Mounting Bracket

The mounting subsystem of our design consisted of aluminum components which fastened the linear actuator brake module to the frame of the gait trainer. The initial steps of the production of this subsystem began with finding a reliable way to connect an aluminum mount to the linear actuator. There were two options to do this, being either use the square mounting bracket on the top of the linear actuator, or using two screws mounted to the side of the actuator. It was decided that two points of contact would

be better than one, so the screw option was chosen. For the mount to fit to the gait trainer a simple two piece square mount, with a circular hole in the centre would suffice (the hole having the same diameter as the leg on the trainer of course). We began drawing up plans for the mount, aiming to keep its dimensions as compact as possible. We thought constructing the blueprint out of aluminum would keep the weight of the mount to a minimum, while still providing ample strength to our brake. Our eventual blueprint can be seen below.

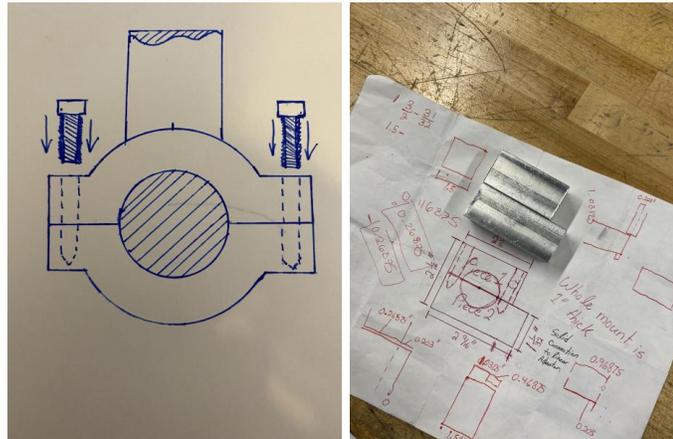


Figure 13. a) and 13. b) Mounting Bracket Blueprint

Next, we acquired scrap pieces of aluminum from the Brunfield machine shop. These were face milled to the desired dimensions of the two blocks seen above in a mill with up to one over one thousands of an inch precision. These would become the square mounting brackets that would fit around the gait trainer leg. The next task was to drill and tap the holes for the bolts. This was again done using the mill, drilling holes for quarter inch threaded bolts. These bolts would extend through the length of the bracket and be bolted with lock nuts on the other side. Furthermore, the one and one sixteenth centre whole in the bracket would be drilled on the mill by starting with a small drill bit, and continuously graduating up to the desired diameter. The finished bracket can be seen in the image below.



Figure 14) Mounting Bracket

In order to fasten the bracket to the actuator another “L” bracket was required. This would also be made of aluminum, and would mount via two screws to the linear actuator. A larger flat sheet of aluminum was started with, and the “L” shape was cut out with a band saw. The two screw holes were drilled in a drill press for M6 screws. The fully assembled mount can be seen below.



Figure 15) Assembled Mount

6.3.2 PLA Circuit Casing

With all components of the circuitry completed, casings were required to contain them on the brake. The casings, in order to keep them lightweight and low cost, were first designed in CAD modeling software, and then 3D-printed with PLA. The images below show how these were attached to the brake mount.



Figure 16) PLA Casing Components

6.3.3 Mounting System And Circuitry Case B.O.M

Table 5. BOM for Mounting Subsystem

Item	Description	Quantity (Approx.)	Price	Source
Aluminum Bar	Used to make square bracket mount.	2	Free	Brunsfeld Machine Shop Scrap Bin
¼” Hex Bolt	Fastens bracket together.	2	\$6.00	Canadian Tire
Aluminum Sheet	Used to make “L” bracket	1	Free	Brunsfeld Machine Shop Scrap Bin
PLA	Used to make circuitry Containers	3	Free	Makerspace 3D-Printers

6.3.4 Equipment Used

- a. Milling Machine
- b. Drill Press
- c. Band Saw
- d. Various hand tools such as files and scribes

6.4 Testing & Validation

The final prototype design was tested extensively. The table below highlights the most important testing done on the final prototype.

Table 6. Quantitative and Results of the Testing Done on the Final Prototype

Test	Target	Observed
Activation Range	10 meters	18.3 meters
Braking Force	750 Newtons (Given)	Satisfactory
Activation Time Delay	1 second	0.5-1.0 seconds
Mount Stability	Firmly (no budge)	Minute movement
Braking Distance	1 meter	1-2 meters
Total mass (for one brake)	2-3 lbs	5 lbs
Battery Life	2.5 hours	2-3 hours

The activation range was tested by measuring the distance in which the linear actuator would activate. The farthest distance recorded was 18.3 m. Note that the speed of transmission reduces at the distance from the linear actuator increases. Furthermore, the braking force was tested on the gait trainer. The brake shoe was flush with the wheel and then the group tried to walk with the gait trainer. After the brake was applied, the wheel could not turn. It is assumed that if a second brake was applied in tandem, that the gait trainer would stop completely. The action time delay was tested using a timer. The time it took to activate the linear actuator after the remote was activated was about 0.5 - 1.0 second depending on the activation range. The mount was attached to the gait trainer. As the dimensions of the mount were inaccurate to the diameter of the gait trainer's pole, the mount had to be filed. Consequently, the mount was a bit loose and had minute movement. Once the brake was applied, the braking distance was measured to move 1-2 meters after the braking force was applied. Using a scale, the braking subsystem weighed about 5 lbs. The battery life is estimated to be 2-3 hours by calculations using the specifications of the linear actuator and batteries. Additionally, the group kept track of how long the prototype has been in use throughout all of the testing and it has not needed new batteries.

7 Conclusions and Recommendations for Future Work

In this project, our group learned many technical skills. We learned how to do iterative prototyping based on client feedback. This project incorporated knowledge from digital technologies and manufacturing.

Furthermore, if we had more time our group would have created a second brake. Currently, the prototype has only one brake but for the remote braking system to completely stop the gait trainer, a second brake would be needed on the other back leg of the walker. The second brake would work with the same remote and in tandem with the other brake. An identical brake could be created for the other wheel using the same methodology for the first one. Another bluetooth module would be needed on that wheel as well. The bluetooth module would need to connect it to the remote (transmitter). There would be no need to make any changes to the code to get them to work together.

Additionally, the current remote circuitry uses a breadboard and wires which are unnecessarily bulky and prone to disconnecting. Ideally, a custom printed circuit board (PCB) would be created to compactify the circuits. A PCB is much more reliable as the connections would not disconnect or more while in use. This PCB could be created using a PCB machine or ordered online.

A custom battery box would need to be created for the 9V battery. Currently it is attached using hot glue but had we had more time we would have created a 3D printed box for it. Just like the remote control, it would have some sort of hole so the wires could pass through to connect to the circuit.

Moreover, the client requested a forward brake in addition to the back brake. As there is less room than on the back wheels, the same brake on the back wheels could not be used on the front wheels. A different design would need to be created for the front wheels.

Finally, the group would have bought better equipment. The linear actuator is slow and bulky and a more expensive one would have been faster and more compact. Consequently, the design specifications would have been better. For instance, the transmission time could have been decreased from a tenth of a second if the linear actuator was faster. Additionally, the 12V is heavy and large so a smaller and ideally a lithium battery would have been purchased. Then, the device could have been rechargeable and a battery indicator would have been included in the remote.

8 Bibliography

[1] Statistics Canada. *Disability in Canada: Initial findings from the Canadian Survey on Disability*.

<http://www.statcan.gc.ca/pub/89-654-x/89-654-x2013002-eng.pdf>. Published December 2013.

Accessed December 5, 2021.

[2] Last-Minute Engineers. *Interface L298N DC Motor Driver Module with Arduino*.

<https://lastminuteengineers.com/l298n-dc-stepper-driver-arduino-tutorial>. Accessed November 4th,

2021.

9 APPENDIX I: Design Files

Table 7. Referenced Documents

Document Name	Document Location and/or URL	Issuance Date
MakerRepo Link	https://makerepo.com/elang099/924.gng210-1-a8-brakethrough-solutions-	December 5, 2021