Design Study & Final Project Report

Abigail Figueroa, Charuvi Singh, Derek Desrosiers, Kruthi Gundu, Samhitha Bodangi
May 22nd, 2024
Massachusetts Academy of Math and Science

Table of Contents

An Optical Text-to-Braille Translation Device

Introduction

Motivation and Problem Statement

In a world abundant with information, particularly in the visual format, accessibility remains unequal for the visually impaired. Approximately 40 million individuals in the world are blind, and 250 million have visual impairments (WHO, 2022). Visually Impaired Persons (VIPs) face daily challenges in accessing the written information of their environment, which is crucial as written information pertains to knowledge and education. Braille, a configuration of raised dots representing letters and numbers, provides a means for VIPs to read and write. Despite its significance, only about 10% of legally blind individuals are proficient in Braille, largely due to the bulkiness and cost of Braille materials and the lack of qualified teachers to instruct students in Braille literacy. (Braille Works, 2023). Children and adolescents are particularly affected, as early Braille literacy is essential for their future education, employment, and overall life outcomes.

Facing this challenge, there is a critical need for a cost-effective, portable device that can translate written text into Braille using Optical Character Recognition (OCR) for the purposes of expanding the deficit of Braille literacy. Such a device would enable VIPs, particularly the younger demographic, to access a broader range of written materials independently, enhancing their literacy, autonomy, and quality of life. The device should provide a medium to display the translated Braille, be easy to use, and be engaging, addressing the gap in current assistive technologies for Braille literacy education.

Target Audience

While most of those with visual impairments tend to be older individuals, we inquired through personal interviews and found that such a device would not be extremely useful for them as there is a decreased desire to learn Braille partly due to reduced tactile sensitivity at an older age. However, there is still a relevant younger demographic that would benefit from a text-to-Braille translation device to aid in the expansion of Braille literacy as for them, learning Braille would facilitate the absorption of information needed for future endeavors. Thus, our device is aimed for those on the younger side, particularly in the context of Braille education.

Purpose of document

This design document aims to present the final prototype that our team has developed over the course of 3 months, with multiple rounds of designing and developing. Through working on both the CS and hardware components of our design, we have created the foundations of a prototype that meets a considerable number of our requirements.

Summary of Market Research

Currently, there are existing devices on the market that transmit information to VIPs via Braille. Almost all of the existing Braille devices available do not perform the function of translating text to Braille using Optical Character Recognition (OCR), making it difficult for VIPs to access text in a portable manner. Current designs such as those prototyped by Tactile (Petronzio, 2017) which scan printed text underneath the device and produce a Braille display on top. However Tactile, developed in 2008, now uses outdated technology which could be made more efficient, like for example, expanding from Grade one to Grade two braille. While some devices may already convert text screens on computers to Braille, they are often also costly. For example, the Brailliant B1 20X Braille Display costs about $2200. The portable device connects to a phone or computer via Bluetooth and prints the text in Braille on the surface of the device (Humanware, 2024).

Brailliant helps users read e-books on iOS and Android devices but is financially inaccessible to many. Additionally, the device is unable to read text that is not electronic, making it difficult to function outside of the home or for physical media. Adding a camera element to the device can allow users to understand the text in their surroundings, giving users a better understanding of the written information of their environment.

Preliminary Designs

The following is composed of some initial preliminary designs of the device and the design selection process.

Design #1 - Braille Modules with Piezoelectric Actuators

This design concept also utilizes OCR technology to translate text to Braille in a portable, handheld display. This device, utilizing piezoelectric actuators published in the paper “Touching force response of the piezoelectric Braille cell” by Smithmaitrie et al. in 2008, moves the Braille pins and displays more Braille characters while being simultaneously portable.

Piezoelectric actuators are a popular development in Braille system technology (Sarkar and Das, 2012). Piezoelectric actuators offer highly portable Braille displays using a motor, a mobile element or slider, and a piezoelectric ceramic transducer. Unlike other mechanisms, this motor allows for a light, simple design that emphasizes portability. Because the cells are more compact, a higher number of Braille characters can be added to the device, adhering to the requirements of the device being less than 8 cm tall, 10 cm wide, and 5 cm deep. A driver circuit for controlling piezoelectric actuators will be implemented under the Braille face. A rechargeable lithium-ion battery with sufficient capacity to power the device and actuators will be implemented. The pins being controlled by the Piezoelectric actuators will be purchased separately.

While relatively simple to implement, individual piezoelectric Braille cells are highly costly, with just one priced at $30. The cost for each Braille cell would limit the amount of Braille characters that would be able to be presented, and the cost would contradict the important affordability criteria in our requirements.

Figure 1: A piezoelectric Braille cell (Smithmaitrie et al., 2008).

Design 2 - Cam and Follower Assembly

This design concept from 0LAUK0 PRE2018 3 Group 13 presents a unique, portable, and handheld display (Control Systems Technology Group, 2019). This device places an emphasis on portability and cost-effectiveness, containing 6 Braille cells controlled by 3D printed pins using a cam and follower assembly.

The pin configuration on each Braille cell uses a cam and follower assembly made up of five parts: the groove, cam, follower, spring, and pin, which work together to form a manually toggled system, as shown in Figure 2. As seen in many pens, the cam pushes the follower, rotating it to a resting position that places the pins in an upward and downward configuration. Each Braille pin has an assembly below it powered by a stroke solenoid. The pins will have an adequate lifting force, as the follower on the “ON” state locks in place until pushed by the cam, preventing the pin from being accidentally pushed down.

The motor needed to operate this device is larger than expected, which would contradict our portability requirement. Also, a cost analysis found this method to also be expensive.

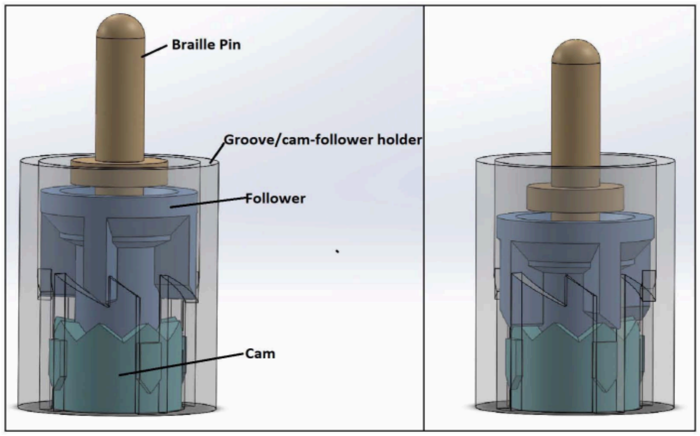

Figure 2: Cam-follower system shown in an ON (left) and OFF (right) state (Control Systems Technology Group, 2019).

Design 3 - Electromechanical Refreshable Braille Module + FireBeetle Camera

This design incorporates a prototype created by Vijay Varada published on Hackaday (Varada, 2023). It uses an affordable alternative to typical refreshable Braille display technology by incorporating rotating electromagnets and 3D-printed Braille Cells connected to PCB boards.

This potential design concept employs electromechanical refreshable Braille modules. The key component of this design is a cam actuator, which consists of an eccentric cam with a magnet embedded in it. This complex is rotated to two stable positions by an electromagnet that changes its polarity. The rotation of the cam causes a Braille dot to be lifted or taken down. A visual of the Braille cell construction using the cam actuator is shown in Figure 3. The frame of this device will be designed in SolidWorks and Onshape CAD and 3D printed using a 3D printer. All electrical design will be constructed using a PCB board and an Arduino circuit. TTS will be implemented using speech application program interface (SAPI) the given text is converted into audio. Batteries will be added to ensure the device can sustain power whilst being portable.

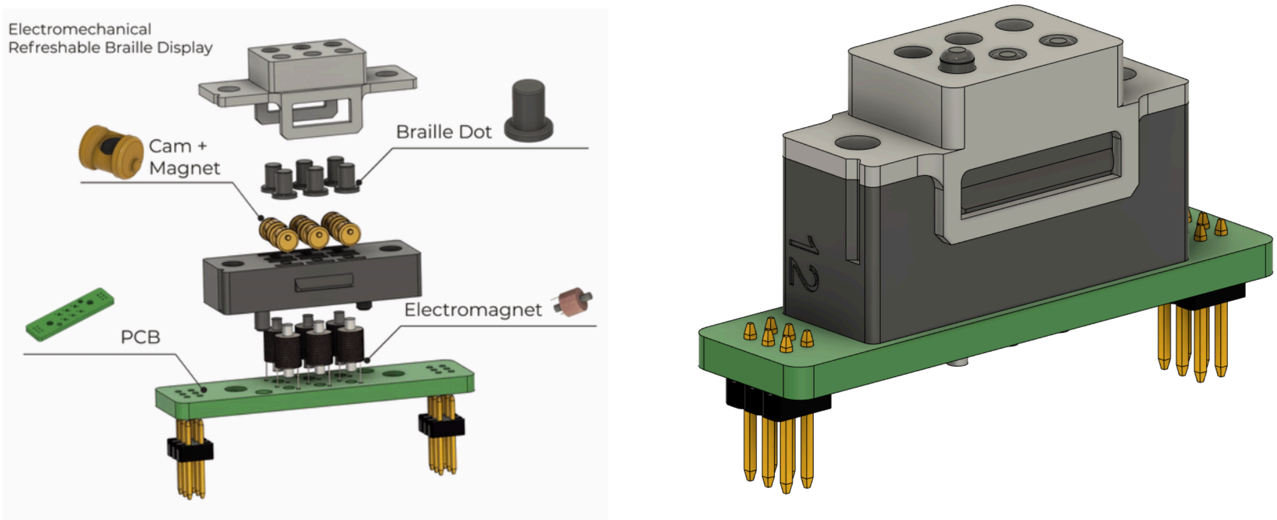

Figure 3: Braille Cell Construction from *Electromechanical Refreshable Braille Module* (Varada, 2023)

Building/Testing

Our team decided to pursue Prelim. design #3 because it promised both portability and affordability, qualities that increase the amount of Braille characters that could be displayed. Not only that, the materials needed were accessible to procure and overall, assembly seemed straight forward.

Build Steps

3D printing

Using an SLA 3D printer, we printed several different scaled up models of the Prelim. design #3 braille cells to both observe and experiment with. We worked with 1x, 2x, 5x, and 10x sized cells. Alongside that, we printed the braille pins that correspond to the design and the cams for holding the miniscule rare-earth magnet. To aid in individually winding solenoids, it's recommended to also print a solenoid winding tool that holds the coil and ferrite core in place.

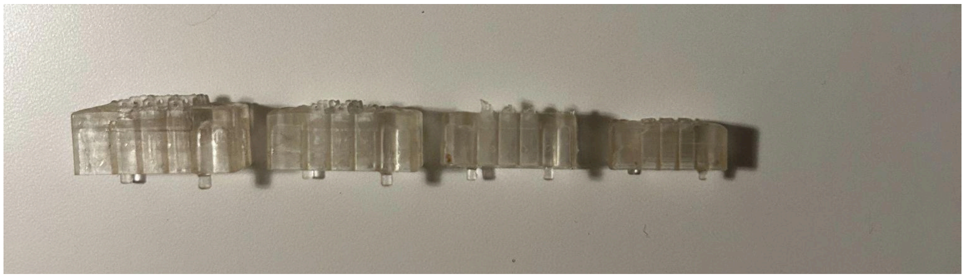

Figure 4: (From left to right) The 10x, 5x, 2x, and 1x sized up braille cells

3D printing proved to be tricky at times as we found that smaller items came out with less accuracy and were very fragile. The l0x amplified Braille cell was the most intact and served as our main one to test functionality, unfortunately, this meant that the PCB board was too small for it as it was designed for the original-sized Braille cell which was much smaller and printed the cam holder inaccurately.

Below is the solenoid winding tool used to expedite the process of making multiple solenoids.

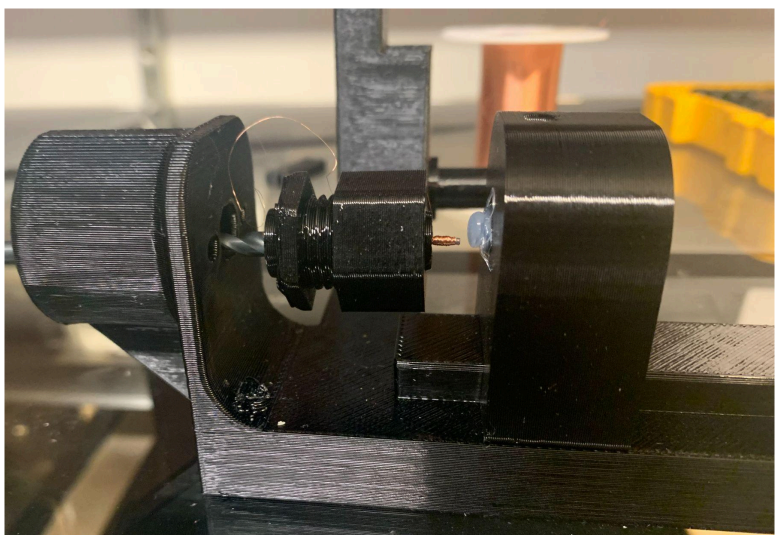

Figure 5: Solenoid winding tool

Electromechanical component

Adding drill chucks to the winding tool made the grasp on the ferrite rod firmer and sped up assembly. The goal was to create several individual solenoids which when connected to a voltage source, would generate an electromagnetic field that would reverse the polarity of the magnet in the cam, making the cam rotate and pushing the braille pin upward. To make a solenoid, use the tool, drill, and drill chucks, to rotate the iron ferrite rod and wrap the copper coil around it several times. Cut the cams from their 3D printed supports and place the rare-earth magnet inside. The cam is supported on a small axle printed out on the top of the braille cell so that it could rotate up and down in place. Finally, place these pieces into a slot for a braille pin and place the braille pin on top. To test functionality, one can perform the “One-Dot” Test by connecting the pin to a 6V voltage source and seeing if raising the pin up and down is successful.

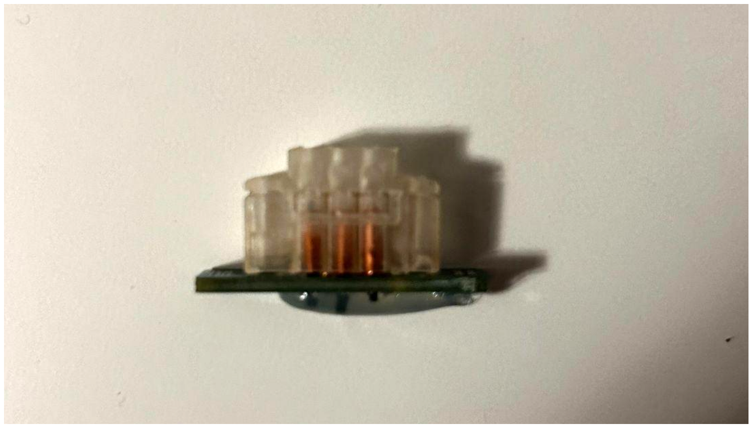

Figure 6: Partially assembled 1x braille cell (Missing cam)

PCB Board

There are many components that were used to make the PCB board, including the files needed and parts (besides the PCB boards themselves), in the spreadsheet linked here. Following the designations of each specific part and where it belongs on the board, you can solder each part in its appropriate place. Make sure when soldering the decoders that they are facing the correct direction. A lot of these parts are relatively small, so a magnifying glass and smaller soldering iron help in the process. Due to the close proximity of some parts on the evaluation board, applying solder flux will help the solder sit in its proper pad. It is important that an Arduino Nano is used as it not only runs the code necessary for the switching of polarity of the Braille cell pins but also regulates the voltage throughout the PCB board.

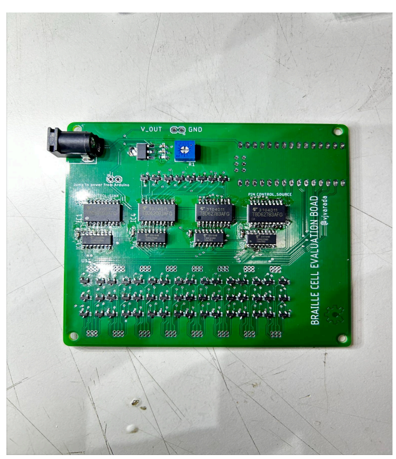

Figure 7: Larger finished soldered PCB board

OCR and Text Conversion

Using a Raspberry Pi, optical character recognition (OCR) can be performed to take image text and convert it into braille. As the Raspberry Pi can run on Python, the Pytesseract library was used to perform OCR. A web camera was connected to the Raspberry Pi and a web server was initialized to display the visual conversion between text and braille. After installing the necessary dependencies and creating a virtual environment in Python, the web server was displayed. For the text-to-braille conversion, the Pybraille library was used. The library can convert the text into Unicode, which is a set of characters within the Unicode standard that represents Braille patterns. Each Unicode Braille character corresponds to a specific Braille pattern, allowing digital text to be translated into tactile Braille through the Python library. In the web server, after the Raspberry Pi takes a snapshot of the camera image, it will run the Braille conversion program and output the respective Braille Unicode characters.

Design Studies

"One-Dot Test" Design Study

Purpose:

This test was designed to determine if the fundamental concept to raise a Braille pin up and down is functional. The "One-Dot Test" assembles a single pin in the Braille cell with a solenoid winding connected directly to a power supply without need for the PCB board. Our team will use the results of this test to determine if this is a viable pathway for the greater construction of the Braille display.

Independent Variable:

The independent variable is the constructed single powered Braille pin, which consists of the completed assembly of the solenoid and rotation of a cam and rare-earth magnet pushing the pin up and down.

Dependent Variable:

The dependent variable is the efficacy of this model, which is measured by

1. Whether or not the Braille pin was able to be raised
2. The time it took for the pin to be raised (which attributes to the responsiveness of the Braille display)

Materials:

- 3D printed Braille cell, Braille pin, and cam
- Iron ferrite core

- Copper wire
- Drill, Drill chucks
- Small rare-earth magnet
- 6V voltage source
- Stopwatch

Methodology:

1. The CAD files for the Braille cell and pins were developed and 3D printed
2. Using a drill and drill chucks, the copper wire was wound around the ferrite core to create a solenoid
3. The magnet was placed inside the 3D printed cam using tweezers
4. These components were placed inside the Braille cell and the pin placed on top
5. The ends of the copper wire were connected to an open wire with a voltage source
6. The voltage source was turned on and off and the responsive time of the pin was measured with a stopwatch

Qualitative Test Results:

	Functional	Responsive time
Trial 1	Yes	< 0.378 sec
Trial 2	Yes	< 0.343 sec
Trial 3	Yes	< 0.395 sec

Analysis:

In the three trials, the One-Dot test was functional each time and had a responsive time of less than 0.4 seconds.

Conclusion:

The One-Dot test proves functional and with a quick responsive time.

Future Work:

From the One-Dot test, the next steps are to test with two pins and then the full array of six Braille pins. The following tests require the PCB board to control which pins are raised.

Optical Character Recognition (OCR) and Text-to-Braille Conversion Design Study

Purpose:

This test was used to determine if the written code (Python) is functional by testing its ability to recognize text from a camera input and successfully convert the extracted text to Braille characters. In this test, a camera will be connected to a Raspberry Pi and this visual input will be analyzed by the Python-tesseract library to output the text. Then, this text will be converted to Braille Unicode characters using the PyBraille also on the Raspberry Pi. This conversion to Unicode is a simple mechanism to map visual text to Braille. In addition to this, the text will also be converted to boolean formats of the Braille cell modules. These booleans will define which Braille dots are present in the character.

Independent Variable:

The independent variable was the text that was being recognized by the camera and converted to Braille Unicode and Boolean representation. All words were captured by a text camera connected to a Raspberry Pi.

Dependent Variable:

The dependent variable is the accuracy of the program in recognizing the text and converting it to Braille Unicode and Boolean representation accurately.

Materials:

- Raspberry Pi
- Test Camera (webcam, Raspberry Pi Cam, etc.)
- Python Code implementing Python-tesseract library
- Whiteboard and whiteboard marker

Methodology:

1. A test word or phrase was written on the whiteboard.
2. The camera was positioned towards the whiteboard and text was extracted from the camera input.
3. The text was converted to Braille Unicode characters.
4. The camera feed, text output, and Braille Unicode characters were displayed on a web server for verification.

5. Steps 1 to 4 were repeated 10 times for different strings.

<u>Results:</u>

Independent Variables		**Results**	
Text	**Number of characters**	**Final State**	**Comments**
“Hello”	5	⠠⠓⠑⠇⠇⠕	Program successfully added the marker for capital letters
“Hello World”	11	⠠⠓⠑⠇⠇⠕ ⠠⠺⠕⠗⠇⠙	Program successfully added the marker for capital letters
“Good morning”	12	⠠⠛⠕⠕⠙ ⠍⠕⠗⠝⠊⠝⠛	Program successfully added the marker for capital letters
“Good night”	10	⠠⠛⠕⠕⠙ ⠝⠊⠛⠓⠞	Program successfully added the marker for capital letters
“The fox jumped over the lazy dog.”	33	⠠⠞⠓⠑ ⠟⠥⠊⠉⠅ ⠃⠗⠕⠺⠝ ⠋⠕⠭ ⠚⠥⠍⠏⠎ ⠕⠧⠑⠗ ⠞⠓⠑ ⠇⠁⠵⠽ ⠙⠕⠛⠲	Program successfully added the marker for capital letters
“lorem ipsum dolor”	17	⠇⠕⠗⠑⠍ ⠊⠏⠎⠥⠍ ⠙⠕⠇⠕⠗	
“1234567890”	10	⠼⠁⠃⠉⠙⠑⠋⠛⠓⠊⠚	Program successfully added the marker for number characters
“13579”	5	⠼⠁⠉⠑⠛⠊	Program successfully added the marker for number characters
““	0	Nothing was Printed	Unclear if the program printed anything, so a conditional was added to make sure.
“ “	1	Space was Printed	Unclear if the program printed a

			space, so a cognitional was added to make sure.

Figure 8: Raspberry Pi and web Cam Assembly

Analysis:

- After conducting the testing on the camera and Python code, the following analysis was conducted. It was observed that all of the text phrases were successfully captured by the camera and picked up by the Tesseract Python package. Then, the text was extracted from the video feed and converted into Braille Unicode. In addition to this, the program was able to account for the specific braille markers that indicate capital letters and number characters.

Conclusion:

- The Raspberry Pi successfully performed OCR and printed the braille Unicode. The program and the Raspberry Pi successfully work together to display braille characters. Additionally, the web server was able to display the braille characters that were output by the conversion function in the pybraille library. The OCR component was able to successfully take into account the extra markers that represent capital letters, numbers, and whitespace characters.

Future Work:

- In the future, the OCR component will be configured to the device for more autonomy. Additionally, the program currently uses Unicode braille. However, in order to control individual braille pins, the Unicode braille characters will be converted into arrays of booleans. The booleans will correspond to each of the pins, where a 1 represents an up pin, and a 0 represents a down pin.

Magnetic Field Design Study

Purpose:

This test was used to determine if the magnetic field powering the Braille cell pins were working with the solenoid design. Different turn densities were analyzed to understand which one produced the strongest resistance of the solenoid.

Independent Variable:

Turn density, or number of turns per inch.

Dependent Variable:

Magnetic field (Teslas) produced per volt, measured by finding resistance of the solenoid.

Materials:

- Drill
- Drill bit
- 44 AWG copper wire
- Ferrite cores
- Ohmmeter
- Lead-based solder (note: wash hands after use)
- Soldering iron

Methodology:

1. Insert the drillbit into the drill and tighten.
2. Insert a ferrite core into the drillbit about ⅙ of the length, and tighten drill bit.

3. Unwind around 5 feet of copper wire. In the middle of the wire, begin to wrap it around the ferrite core. With one hand, hold each side of the wire with two fingers and with your other hand, push the trigger on the drill to begin winding. Wind until the wire runs out.
4. Heat up the soldering iron to around 650˚F. Place a dot of solder on the iron and push the ends of the copper wire into the bead of solder. The ends should turn silver.
5. Connect each end of the ohmmeter to each end of the wire (connecting to the silver ends that were done in step 4). Set the ohmmeter to the smallest setting (ohms) and measure the resistance.
6. Using the equation V = IR, calculate the current for 1 volt, or 1 = IR.
7. Solve for the equation $B_{solenoid} = \frac{\mu_0 NI}{l}$, where μ_0 is $4\pi * 10^{-7}$, N is the number of turns, I is the current per volt, and l is the length of the solenoid. This will give you the magnetic field per volt generated by the solenoid.

Results:

Test #	Solenoid resistance (Ω)	I (Amperes) per V	N/l (l=10mm)	B (Teslas)
1	26.8	37.3 mA	20,000	$9.37 * 10^{-4}$
2	46.6	21.4 mA	45,000	$12.1 * 10^{-4}$
3	37.4	26.7 mA	37,500	$12.6 * 10^{-4}$

Qualitative Tests:

- Using the calculated magnetic field of each solenoid, it can be confirmed by qualitatively testing the strength of the magnet to the ferrite core if the solenoid is charged. This can be done by either connecting it to a battery or a current/voltage generator.

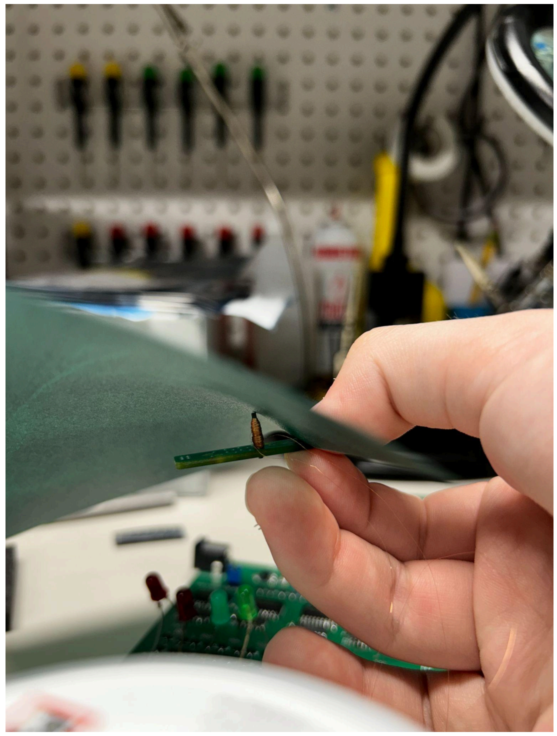

Figure 9: Experimentation with solenoid magnetic field

Analysis:

- The analysis portion of this test was the qualitative test to support the result, as well as the additional calculations done after measuring the resistance of the solenoid. This analysis provides a helpful tool for how to continue with the project and what factors to keep in mind.

Conclusion:

- In conclusion, solenoid #2 is the most promising for providing a strong base for a Braille cell design as it has the strongest magnetic field and would therefore be the most effective in reversing the polarity of the magnet and reversing the direction of the cam.

Future Work:

- Future work for this testing topic could include the investigation of various different materials of wire as well as different gauges to determine the best combination of various factors that will result in the strongest magnetic field.

Light-Emitting Diode (LED) Design Study

Purpose:
The LED test was used to test the driving electronics and ensure that the electrical current can be controlled between specific braille modules.

Independent Variable:
The LED lights, the amount of voltage for each pass

Dependent Variable:
The direction of the current

Materials:
- PCB board
- Arduino Nano Microcontroller
- 4 small LED Lights
- USB connect for the Arduino and Computer

Methodology:
1. Inset the LED lights into the pin openings of the PCB board, spacing them so each LED is in one braille module pin
2. Connect the Arduino Nano pins to the top left area of the big PCB board
3. Connect the USB cable to the computer and the Arduino board. There should be a red light that indicated the microcontroller is powered
4. Using the Arduino IDE, ensure that the Arduino Nano is successfully recognized by the computer and is visible in the Serial Ports tab
5. Using the Arduino IDE, code the program to switch lights between each LED
6. Record the timing and when each LED light is turned on

Results:

	LED 1	LED 2	LED 3	LED 4
Trial 1	0 seconds	1.2 seconds	2.1 seconds	3.2 seconds
Trail 2	0 seconds	1.1 seconds	2.1 seconds	3.1 seconds
Trail 3	0 seconds	1.1 seconds	2.0 seconds	3.2 seconds

Analysis:

- The program was successfully able to channel the current to separate LEDs, which an average 1-second difference between each LED being lit. The LED lights were able to turn on based on their position on the PCB board. The program was made in such a way that the current would be directed to individual LEDs. The clear difference in lighting between LEDs also shows that the program is accurate and efficient.

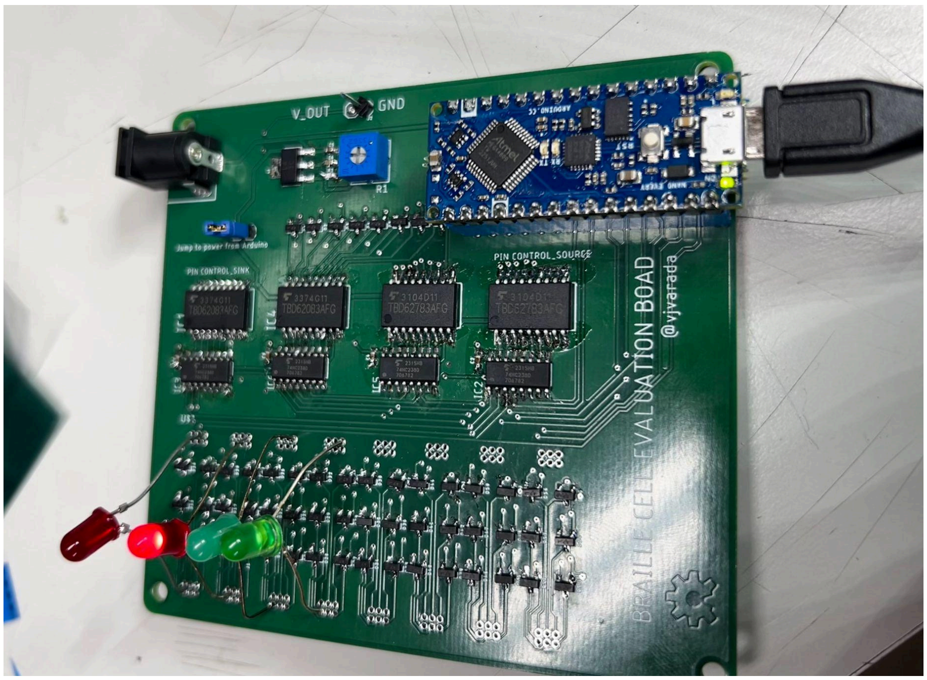

Figure 10: LED Design Study

Conclusion:

- As the LED lights can be successfully controlled individually, it can be said that the Arduino-PCB connection is viable for future alterations. After adding the braille Modules, the code can be modified to control each module separately, which will allow each individual Braille pin to be controlled. This specific control will allow the device to be efficient and simple to implement.

Future Work:

- In the future, Braille cells will be added with their respective small PCB boards onto the bigger PCB board. The program will be expanded upon to control each individual braille pin within the Braille modules

Engineering Matrix

Table 1: The following matrix displays the pass/fail status for each of the requirements and criteria for each of the designs.

Requirement	Type	Level	Version 2 - DOT. (Final Product)	Version 1 - Cardboard Axle Display
The device shall be able to record text from a surface	Functional	1	**Yes**	No
The device shall be able to convert the text into braille characters	Functional	1	**Yes**	No
The device shall be able to present the braille characters to the users	Functional	1	**Yes**	Yes
The device is portable, and can be operated outside of the home	Physical	1	**Yes**	No
The device shall weight less than 400 grams	Physical	1	**Yes**	Yes
The device shall be less than 8 cm tall, 15 cm wide, and 8 cm deep	Physical	1	**Yes**	No
The device shall cost less than $150	Cost	1	**Yes**	Yes
The user has the ability to read braille and/or listen to audio	User	1	**Yes**	Yes
The user has the ability to properly position the device with guidance from the device	User	1	**Yes**	Yes
The device shall include a user guide detailing how to use the device correctly	Documentation	1	**Yes**	No
The device shall be able to convert the text into speech	Functional	2	**No**	No
The device shall be able to present the speech to the user	Functional	2	**No**	No
The device shall have buttons to control the device physically	Functional	2	**No**	No
The device shall be made of durable material	Physical	2	**Yes**	No
The device will connect to an app which can help the user customize their experience	Functional	3	**No**	No
The device can be used with no hands.	Functional	3	**No**	No

The device shall cost less than $100	Cost	3	**Yes**	Yes

Prototyping Iterations

The initial design was the cardboard box axle display. However, it did not satisfy many of the criteria that were deemed crucial for the client, so multiple iterations of the final design were implemented.

After analyzing current market devices, it was clear that a significant issue is the cost of the devices, many of them ranging from thousands of dollars. Additionally, many devices were not portable, which is why we made sure to make the device as small as possible. However, by making the Braille cells small, many of the components of the cells did not print properly. Therefore, the cells were scaled up, and many different iterations were tried. In the end, the x10 braille cell was used as it was printed well with all of the necessary parts, but still small enough to ensure portability.

Additionally, the team initially used an ESP32 Cam to perform OCR. However, it became clear that the implementation of the ESP bard was complex, and the team shifted to using a Raspberry Pi instead. The Raspberry Pi can run on Python, allowing for the use of python libraries to aid in the program.

Final Design Summary

The final device consisted of two developed components. First, was the foundational aspects of the braille display, which couldn't be fully developed to the extent we wished because of consistent misprinting errors of the braille cell. However, our testing and design studies show that the concepts do indeed function and that we have all the necessary parts. The only improvement needed is a more precise 3D printer that can handle printing the small Braille cells in correct resolution.

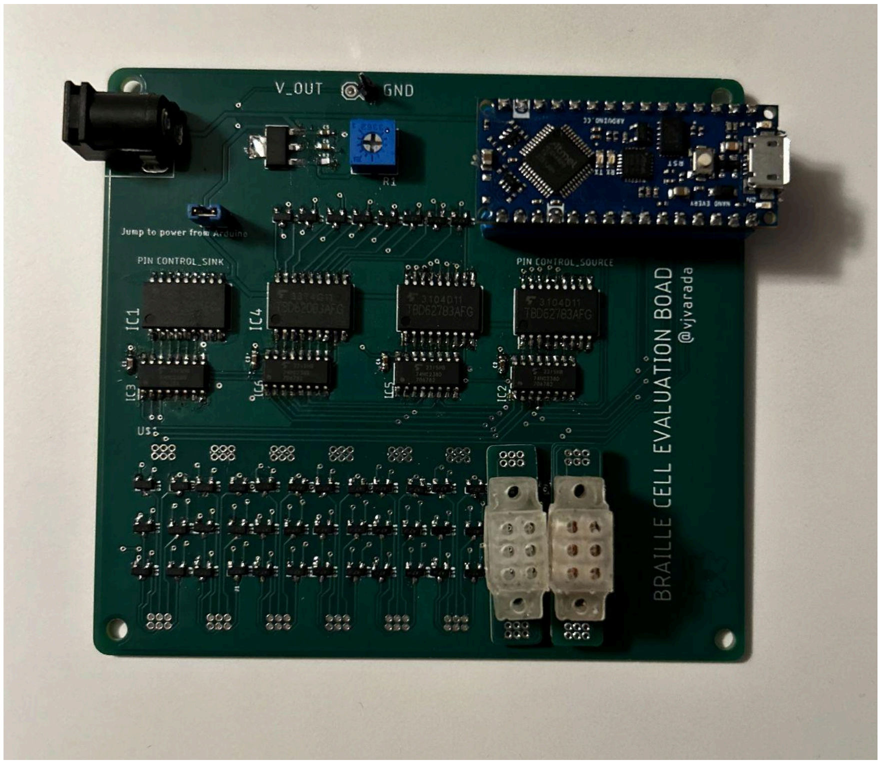

Figure 11: Final Prototype

Second is the text-to-braille configuration conversion aspect, for which we were able to create and refine a computer model for. The Raspberry Pi was used to take camera images and perform Optical Character Recognition using Python's Tesseract library. After converting images into text, the Raspberry Pi uses the pybraille library to convert tests into Unicode braille, which is displayed on the live web server. In the future, both components will be combined to automate the process and connect the OCR to the device itself.

Figure 12 below details what the final, fully incorporated device will look like, combining both the optical and braille display aspects. This is the more complex device that our group will work towards during our further continuation of the project in Lemelson.

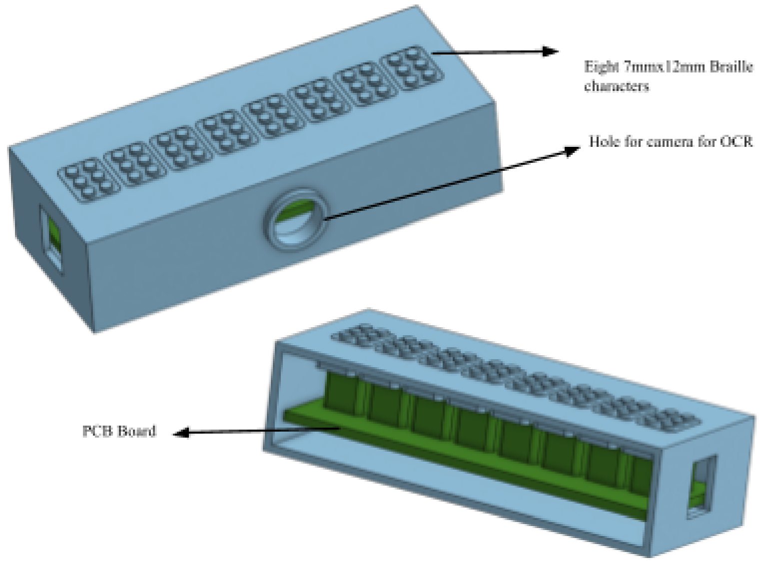

Figure 12: CAD Model of the projected incorporated device (8 cm tall, 15 cm wide, and 8 cm deep)

Table 2: The following matrix shows the final requirements with an assessment for each of the requirements to determine if each of the requirements were met.

Requirement	Type	Level	Version 2 - DOT. (Final Product)
The device shall be able to record text from a surface	Functional	1	**Yes**
The device shall be able to convert the text into braille characters	Functional	1	**Yes**
The device shall be able to present the braille characters to the users	Functional	1	**Yes**
The device is portable, and can be operated outside of the home	Physical	1	**Yes**
The device shall weight less than 400 grams	Physical	1	**Yes**
The device shall be less than 8 cm tall, 15 cm wide, and 8 cm deep	Physical	1	**Yes**
The device shall cost less than $150	Cost	1	**Yes**
The user has the ability to read braille and/or listen to audio	User	1	**No**
The user has the ability to properly position the device with guidance from the device	User	1	**No**
The device shall include a user guide detailing how to	Documen	1	**Yes**

use the device correctly	tation		
The device shall be able to convert the text into speech	Functiona l	2	**No**
The device shall be able to present the speech to the user	Functiona l	2	**No**
The device shall have buttons to control the device physically	Functiona l	2	**No**
The device shall be made of durable material	Physical	2	**Yes**
The device will connect to an app which can help the user customize their experience	Functiona l	3	**No**
The device can be used with no hands.	Functiona l	3	**No**
The device shall cost less than $100	Cost	3	**Maybe**

The only level one requirements not met are those to do with the audio component of the device, which would repeat the text in the text-to-speech format and provide auditory instructions to the user for positioning the camera in the ideal location. In hindsight, perhaps these requirements would have been better off being labeled as level two, as they are not critical to the functionality of the device, but the reason for the failure to meet was that we chose to devote most of our time to ensuring that key aspects of the device perform the way they need to.

Future Work

Our future work is the continued development of this design. Our first priority is to remedy the issue with the 3D printer, perhaps by getting access to more precise ones. High-quality printers cost between $50-$200 for a print, but more information is needed for an accurate estimate. A different course of action would be to keep the 10x braille cell, which was printed intact, and reorder PCBs that would fit that size. Next, would be the implementation of the audio component so that a VIP can receive instructions on positioning but directly translate from text-to-speech if they didn't wish to use the braille function at that moment. Finally, because we aim for portability, we would take the lightweight device and strap a neck cord around it so that a user can carry it without their hands, as we learned from personal interviews that VIPs typically need their hands free when navigating, one to hold on to a sighted person if need be and the other for the walking stick.

Appendix

Appendix A. Table shows all products bought and product details with unit and total price

Item	Supplier	Catalog Number	Quantity	Unit Price	Total price	Link To Product
Ferrite Core	DigiKey	1934-1516-ND	90	$0.18	$8.96	https://www.digikey.com/en/products/detail/fair-rite-products-corp/3067990831/8599513
Pitch Pin Headers	DigiKey	2057-HPH2-B-06-UA-ND	30	$0.50	$13.08	https://www.digikey.com/en/products/detail/adam-tech/HPH2-B-06-UA/9831288?s=N4IgTCBcDa4AwFYDsBaAEgBTWFAhFcAbCgKoCCKAcgCIgC6AvkA
Neodymium Disc Magnets	TheMagnetBaron	N/A	100	$0.19	$18.99	https://themagnetbaron.com/products/100pcs-1-32-x-1-64-rare-earth-disc-magnets
Lithium Ion Polymer Battery - 3.7v 1200mAh	Amazon	B07BTV5Q6F	1	$9.49	$9.49	link
Adafruit Micro-Lipo Charger for LiPoly Battery with USB Type C Jack	PiHut	ADA4410	1	$4.93	$4.93	https://thepihut.com/products/adafruit-micro-lipo-charger-for-lipoly-batt-with-usb-type-c-jack?variant=31257709248574
FireBeetle 2 ESP32-S3 with Camera	PiHut	DFR0975	1	$16.17	$16.17	https://thepihut.com/products/firebeetle-2-esp32-s3-with-camera?variant=42474279600323
PCB boards	PCBWay	N/A	60	N/A	$5	pcbway.com

Solenoid Winding tool						
Drill Chucks	Amazon	B07MCZDRBR	2	$4.49	$8.99	https://www.amazon.com/Ximimark-Electric-Fixture-Adapter-0-5mm-3-2mm/dp/B07MCZDRBR

Shipping and Assembly costs	$6.99 DigiKey, $3.99 TheMagnetBaron, $20 PCBWay, $14 DFRobot, $7.44 PiHut (Intl Standard, 3-10 working days)
Total	**$98.30 for parts, $149.72 including shipping**