Yousif Fadhel - fadhely - 400434747
Ryan DeGroot - degrootr - 400373275
Michael Padeigis - padeigim - 400265497
Imran Chowdhury - chowdi13 - 400470828
Sohail Persaud - persas29 - 400473441

April 09, 2025

Executive Summary

Millions of DeafBlind people are still relying on expensive and hard to acquire equipment to be able to have access to written text. The team's Text-2-Braille device is able to close this gap by creating an affordable and reliable alternative to help grow these individuals' independence.

To answer the question *"what should we do?"*, or in other words, is this something that the team can pursue as undergraduate students successfully? The following has been assessed:

During the design verification, the team performed formal demonstrations and the device consistently finished within 7 to 10 seconds, with a worst case runtime of 15 seconds. This metric outperformed the goal of having a translation speed from text to braille in about 20 seconds. Secondly, the accuracy of displaying accurate Grade 1 braille with the device was 100%, meaning that the device was able to translate every possible combination of words within its limitations, outperforming the original benchmark set at 99% accuracy.

The team analysed the risk management of the device using an FMEA table. Initially the scores flagged a structural failure and an undetected battery depletion for the users as unacceptable risks, however after a further iteration of the risk analysis, they were both deemed ALARP once counter measures were put in place. Embedding these safety changes to the prototype negated any risks that were deemed unacceptable and that would put the user in danger.

Three economical analysis paths were analysed over the course of 5 years. The alternatives were as follows: A fully portable device with a camera and AI integration, a desktop connected text input translation device, and lastly doing nothing with the initial funds and not developing a product. Alternative 1 presented a present worth value of $1.5 million and alternative 2 presented a present worth value of $1.13 million, both analyses for developing these products significantly outperformed the "do nothing" option. With this in mind, the team came to the conclusion that developing the Text-2-Braille device as alternative 2, the desktop connected device, would be the better option given the constraints to build the more sophisticated and idealized version.

The team considered the economic assumptions that were made before finalizing a decision on what to do. Due to the overall developmental timeline, complexity, monetary investments, and the sheer time and effort it would demand from a small team, it would be detrimental for undergraduate students. Therefore, the team has opted to not further pursue development beyond the scope of this project.

Design Controls

Finalized Design Inputs and Outputs

Number	Design Input	Design Output
1	**Translates text into braille at a speed of 20 second(s) per word.**	Software translation module (prototype). Implements a text‑parsing function to convert standard text into a unique numerical braille code. Preliminary bench‑test data indicates throughput meets or is near 20 seconds/word under typical usage. [1]
2	**Translates text into braille to a 99% accuracy.**	Validation procedure (planned tests with standard text samples). Log files and analytics confirming correct braille output on at least 99% of tested words. <u>Pass/Fail:</u> criteria established for basic accuracy checks. [1]
3	Reads text to a 99% accuracy.	Reading/recognition sub‑module integrated in the prototype. Planned software adjustments if accuracy falls below 99%. [1]
4	Device must include all 26 braille letters.	Braille character library embedded in software. Preliminary checklist verifying each letter outputs correct braille. Simple user demo script to verify A–Z coverage. <u>Pass/Fail:</u> device includes all 26 braille letters. [1]
5	Device must be able to withstand 100 N of force.	Basic enclosure CAD and material spec (e.g. PLA). Planned mechanical test to confirm that the housing, when compressed or impacted, does not fracture or deform beyond 100 Newtons (or relevant measure). [1]
6	Device must weigh under 1 kgs.	Bill of Materials (BOM) with approximate mass for each component. Initial weigh‑in of a functional prototype to confirm total mass < 1 kg. [1].

7	Device must be smaller than 200 mm by 80 mm by 80 mm.	Dimensioned mechanical drawings ensuring the device's length, width, and height remain under the specified 200 by 80 by 80 mm. Prototype measurement results confirm compliance in typical assembly. [1]
8	Device must be able to be used for 3 hours before being charged.*	Preliminary power management plan with battery selection (capacity specs). Expected usage profile calculations, showing that the device can run continuously for 3 hours under moderate load. Test logs from early pilot runs. [1]
9	Device must have limited interactions under 7 buttons and commands.	User Interface (UI) specification describing layout and functionality of each button/command. Menu flow diagram (if any) or note that the device is mostly single-function with minimal input. [1]
10	Device must not produce noise louder than 25 dB.	Basic acoustic test method planned to measure operational noise. Shielding or dampening approach documented if needed to keep mechanical/actuator sound below 25 dB. [1]
11.1	Device must be able to withstand a fall of 1m.	Drop-test plan that includes repeated drops from 1 meter onto various surfaces. Physical integrity check to confirm no structural breakage.[1]
11.2	Device must be completely functional after a 1 meter drop.	Pass/Fail criteria: device boots up and operates after the test (no cracks or unresponsive components). Functional test verifying translation, braille cells, and user interface still work post-drop. [1]
12	Device operates under minimum amperage requirements for the let-go threshold 6-25mA.	Electrical design notes specifying circuit limits to ensure typical operating current remains under 25 mA. [2] Power monitoring integration in firmware/hardware for immediate fault detection if current spikes.

Table 1. Design inputs and outputs in tabular form.

NOTE:	Design Input 4. → changed from requiring letters and numbers to only letters. Design Input 8. → newly added or clarified to specify multiple hours. Design Input 9. → reworded again from a previous Milestone (2) "least user interaction" to "limited interactions under X buttons/commands". Design Input 11. → was split into 11.1 (physical durability) and 11.2 (functional recovery) from a single drop requirement. Added measurements to input designs.

Final Prototype (most relevant design outputs)

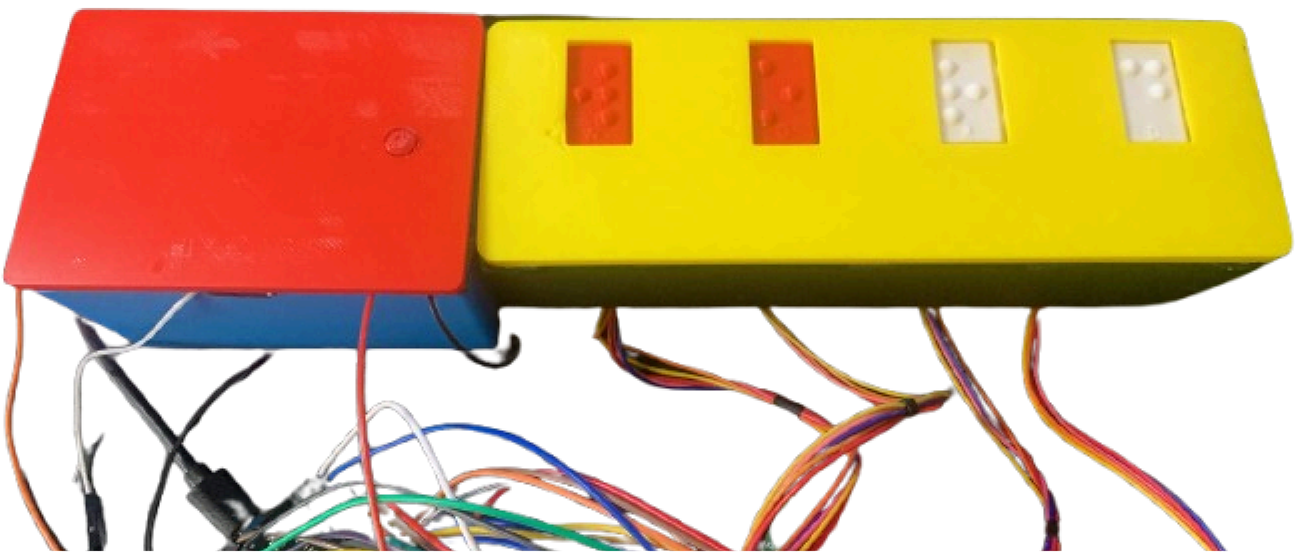


Figure 1. Completed design, the blue & red box on the left contains the "homing" button and its circuit, the yellow and black enclosure on the right contains the braille octagons.

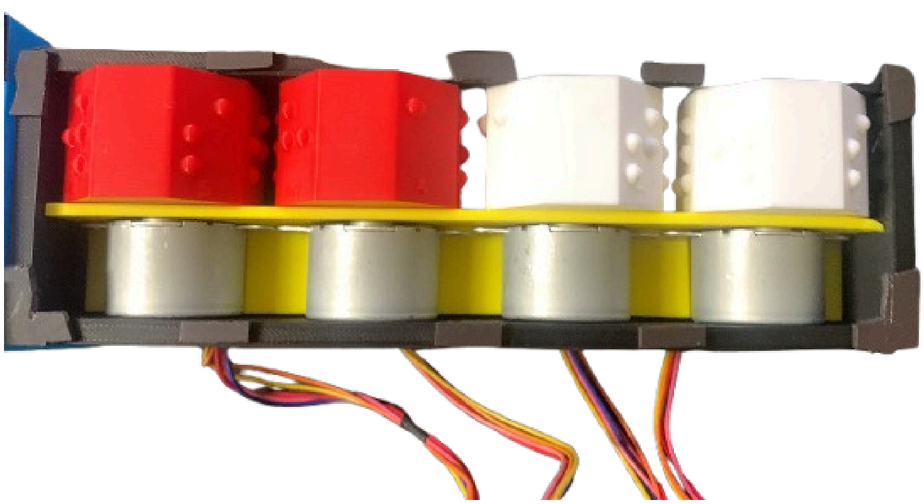


Figure 2. Magnetic Lid removed, showing the inner workings of our device with each octagon connected to a stepper motor

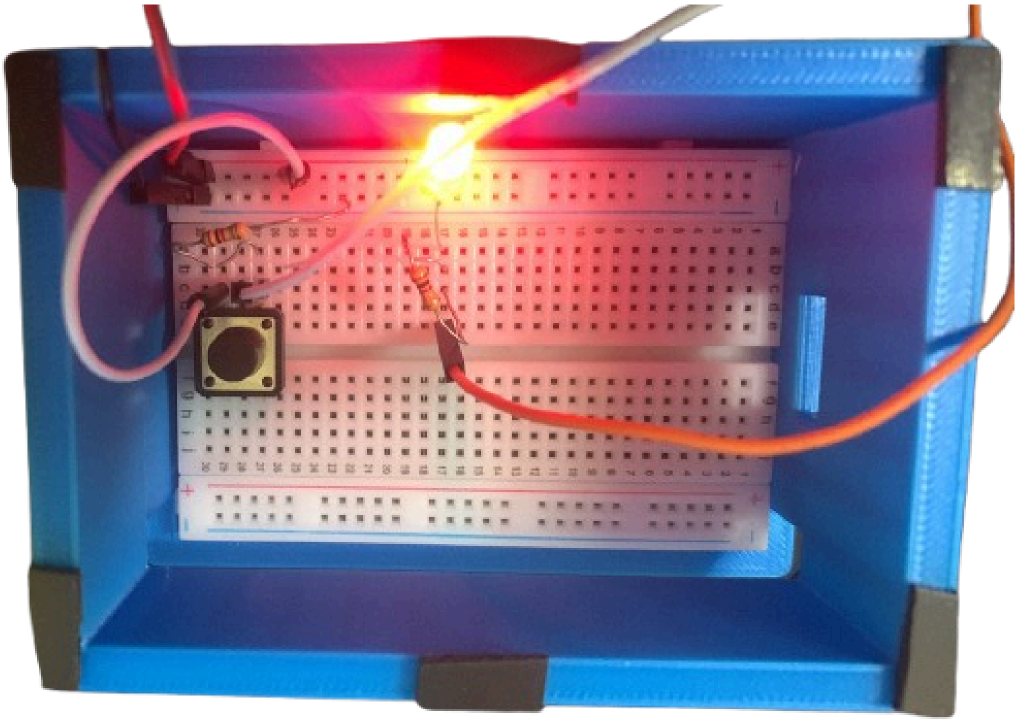


Figure 3. Magnetic lid removed showing the momentary push button, and its circuitry

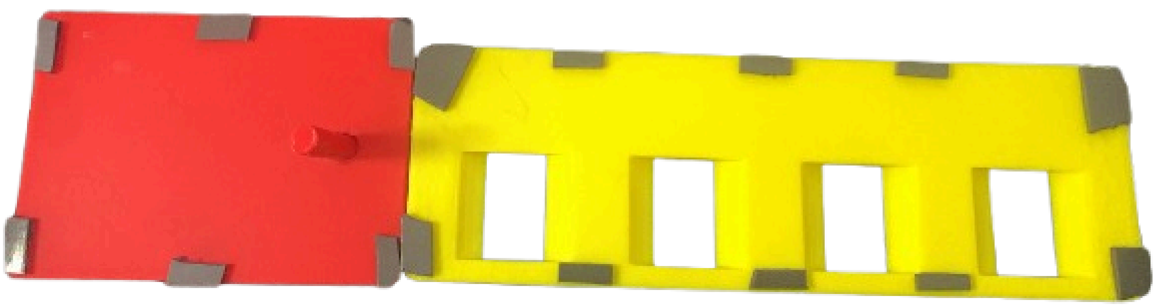


Figure 4. Magnetic lids detached from the main device

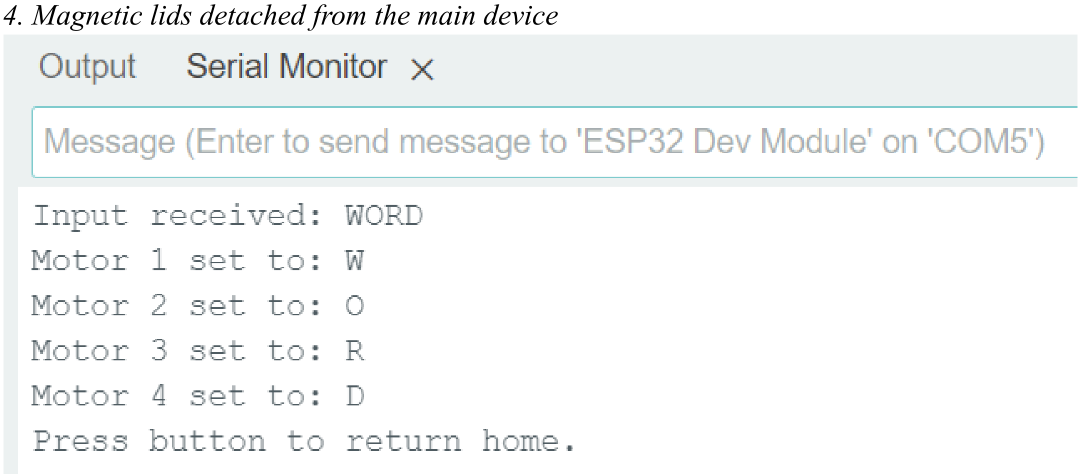


Figure 5. Serial monitor from the Arduino IDE when the program is launched and a 4-letter word is sent

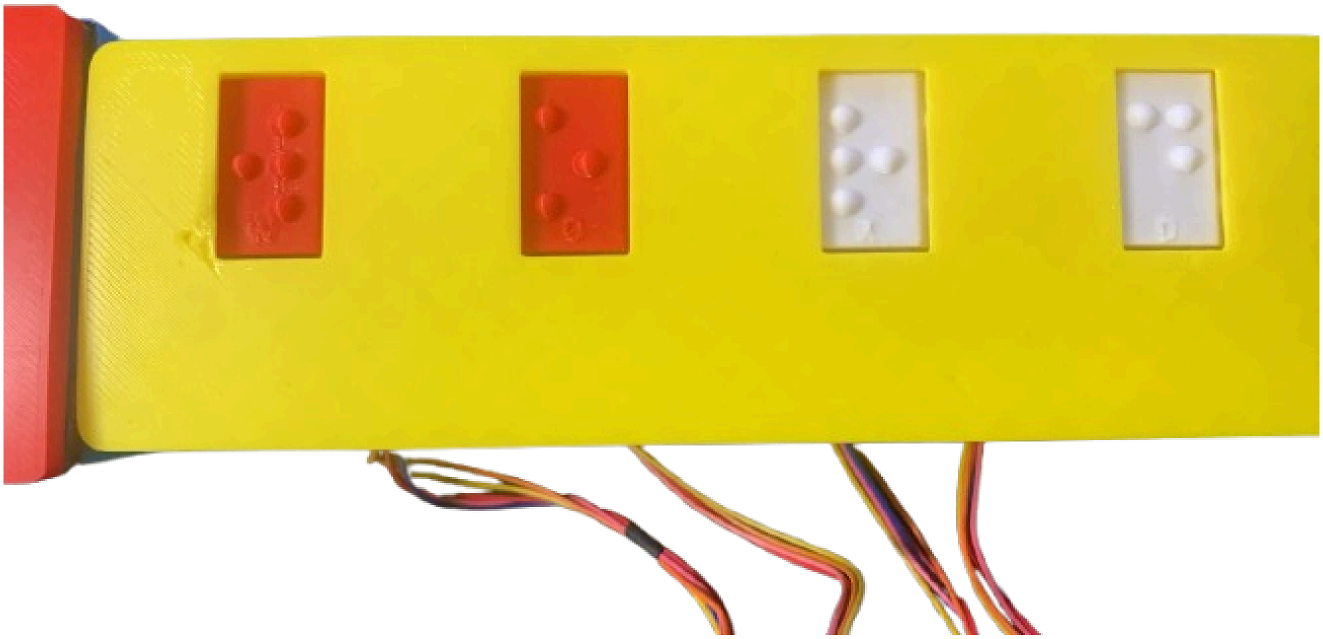


Figure X. Displaying the word "WORD" in braille

```
Input received: PEN
Motor 1 set to: P
Motor 2 set to: E
Motor 3 set to: N
Press button to return home.
```

Figure 6. Serial monitor from the Arduino IDE when the program is launched and a 3-letter word is sent

Figure 7. Displaying the word "PEN" in braille

```
Input received: TO
Motor 1 set to: T
Motor 2 set to: O
Press button to return home.
```

Figure 8. Serial monitor from the Arduino IDE when the program is launched and a 2-letter word is sent

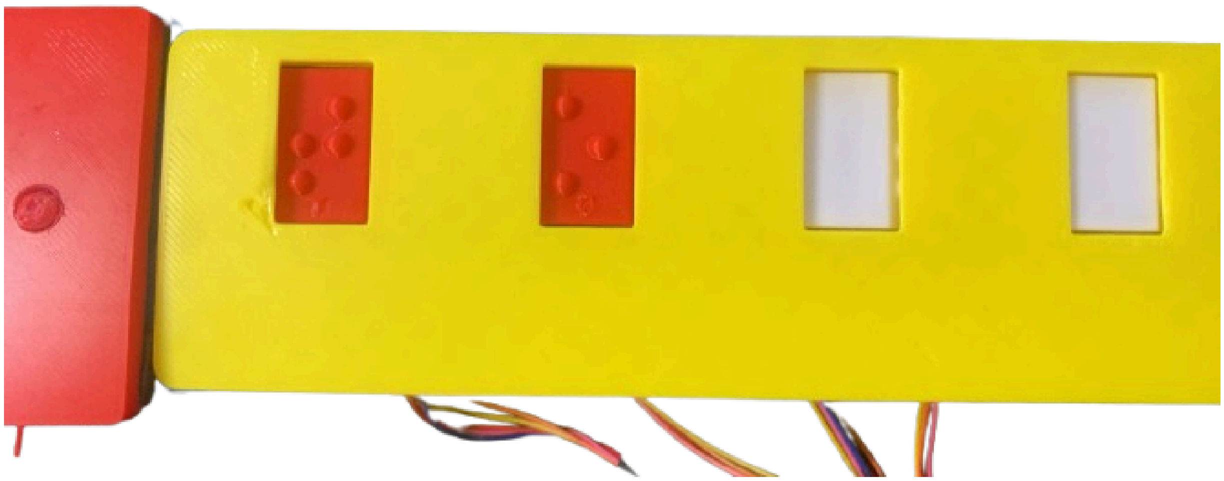


Figure 9. Displaying the word "TO" in braille

Is it correct?

Definition of the Test

The verification tests were carried out using a standardized protocol detailed in Appendix A. This protocol defines the testing process beginning from powering up the device to the translation. The testing procedure is refined down to the moment the enter key is pressed word after a word is typed into the Serial Monitor. This triggers the translation cycle, where the system rotates octagonal display faces to present the corresponding braille characters.

The test is considered successful if the device:

1. Completes the braille translation within 20 seconds **(Design Input 1: Translation Speed)**, and
2. Accurately displays the correct braille characters for the entered word **(Design Input 2: Translation Accuracy)**.

This section outlines both the testing process and the outcomes, clearly aligned with the design requirements.

Recommendation

Based on the outcomes of the verification tests conducted for Design Inputs 1 and 2, the Text-2-Braille device meets and exceeds its core performance specifications. Therefore, the design can be confidently verified as correct.

Justification

Design Input 1 [Translation Speed]

To verify compliance with the required translation time of ≤ 20 seconds per word, the system was tested under a variety of conditions during both formal design demonstrations and internal testing. The process began when a user manually entered a word into the Serial Monitor. A timer was started upon pressing enter, and the duration to complete the braille display was recorded.

Outcome

The device consistently completed full word translations well under 20 seconds, with most trials averaging 7-10 seconds, including mechanical rotation and internal processing. Words that required more rotational steps due to their position on the octagon took slightly longer, with an observed maximum of 15 seconds, still comfortably within the required threshold.
This performance gain resulted from a software update that shifted the motor operation from sequential to simultaneous execution, allowing all four motors to rotate at once. While this timing performance is difficult to represent in a static figure, it is demonstrated in the submitted project videos, which visibly confirm the device's responsiveness and timing accuracy.

Design Input 2 [Translation Accuracy]

To verify translation accuracy, the team conducted standardized testing using input words manually entered into the software interface. Each character was translated into a corresponding braille representation and physically displayed via rotating octagonal segments. The accuracy was assessed by comparing each braille output against the official Grade 1 Braille standard.

Outcome

The system achieved 100% translation accuracy in controlled trials. As shown in Figure 10, where the input word was "WORD", each octagonal face rotated to the correct corresponding braille character. Though minor alignment issues are visible in the letters O and R, caused by prototype motor limitations (e.g., stepper motor resolution, torque slip), the raised dot configurations remained correct and readable by touch.

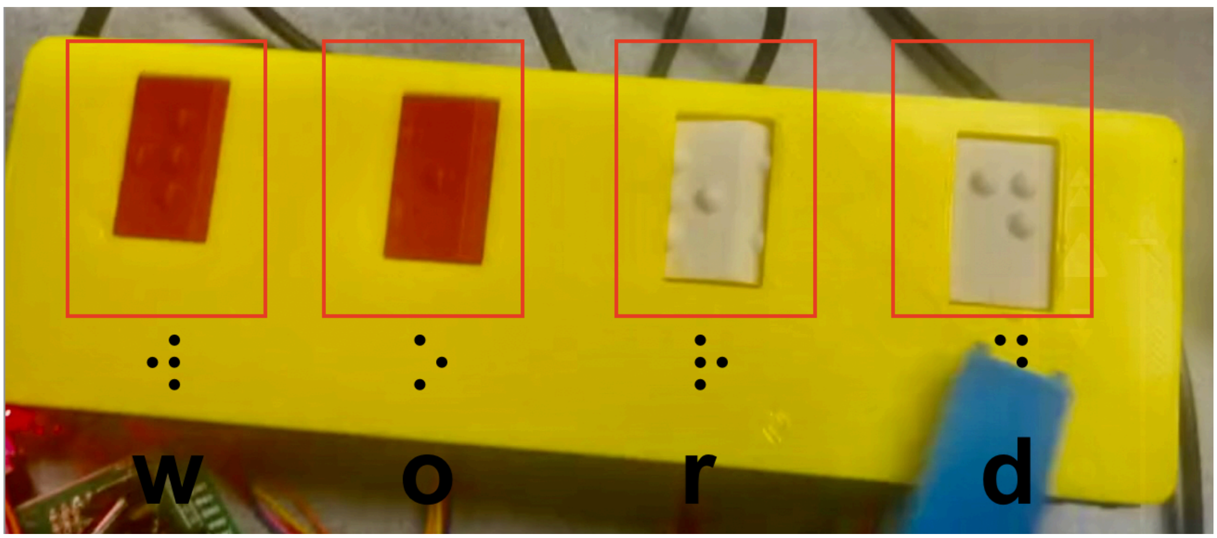


Figure 10. Final braille output from the device for the input text "WORD". From left to right, each octagon represents the braille for W, O, R, and D. Despite slight misalignments in the O and R positions, the displayed braille characters are accurate.

Despite various engineering challenges encountered during the prototyping phase, the system consistently delivered reliable performance. Motor positioning inaccuracies occasionally caused slight misalignments in the display output, particularly in letters requiring longer rotations. Similarly, load-sharing limitations of the ESP32 microcontroller introduced intermittent timing delays and disconnections during early tests, which were later mitigated through the addition of an external power source and software optimization. Additionally, mechanical imperfections stemming from 3D printing tolerances affected the rotational precision and symmetry of the octagonal display pieces.

Yet, in spite of these limitations, the mechanical, electrical, and software subsystems demonstrated strong integration and fault resilience, enabling the device to function consistently and accurately within the defined performance thresholds.

Based on the consistent performance of the device across multiple trials meeting both the translation accuracy and speed requirements, it can be confidently recommended that the Text-2-Braille system is functionally correct and verified. The device is ready for further refinement or usability testing, having successfully met its core design intent under real-world constraints.

Risk Management

Failure Mode	Effect of Failure	Severity, S	Probability, P	Risk Classification	Reasoning
Electrical Failure	Braille not displayed	4	2	Broadly Acceptable	Circuit design and electrical connections were tested and reinforced, with minimal likelihood of failure.
Mechanical Failure	Injury or motor damage	2	2	ALARP	Risk of jamming minimized by design (enclosure shielding), but not zero.
Structural Failure	Enclosure breaks open	5	4	Unacceptable	Drop from common height can easily fracture PLA joints in current iteration.
Battery Depletion	Device stops functioning	4	4	Unacceptable	Observed often in testing; highly likely due to user behavior (e.g., forgetting to recharge).
Battery Explosion	Damage and safety hazard	5	1	ALARP	Proper charging system and lithium-safe designs in place; extremely unlikely.
Comprehension Failure	Misread braille	1	5	Broadly Acceptable	No risk to function; users can re-feel the dots. Issue is non-critical.

Table 2. [Table 4b: Risk Classification and Reasoning]

Failure Mode	**Severity, S**	**Probability, P**	**Risk Mitigation Plan**	**Category**	**Residual Risk**
Structural Failure	5	4	Add internal ribbing, round critical corners, increase wall thickness to 2mm+	Design Change	Re-evaluated as ALARP due to reduced likelihood of breakage
Battery Depletion	4	4	Integrate vibration motor to notify users of low battery through tactile feedback. Add simple mechanical reset or re-docking cue post power loss.	Procedural + Design Change	Re-evaluated as ALARP due to fully accessible user alert mechanism

Table 3. [Table 5: Risk Mitigation Plan]

Risk Mitigation Outcomes and Impact on Classification

The risk management process for the Text-2-Braille project evolved significantly from the original FMEA to the present. Initially, the risk analysis suffered from poor differentiation between failure modes, effects, and causes, which were concerns directly flagged for revisions. For example, causes in the original FMEA often restated failure modes or lacked root cause logic (e.g., “User Error” or “Drops” without specificity). Additionally, no outcome-based reasoning or alignment with ISO 14971 terminology was present.

To address this, the risk management section was completely restructured. Each unacceptable risk was mapped back to its underlying mechanism, evaluated for its user impact, and paired with targeted mitigations. A two-stage classification strategy was applied:

- *Initial Classification (Table 4b):* Each failure mode was analyzed for severity and probability using ISO-based Tables 3a and 3b. Unacceptable risks were identified where S * P exceeded acceptable bounds, especially for Structural Failure (5*4 = 20) and Battery Depletion (4*4 = 16).
- *Post-Mitigation Reclassification (Table 5):* Upon implementation of mitigation strategies, residual risk was re-evaluated.

Structural Failure

Initial Classification:	Unacceptable
Outcome:	The original PLA casing failed impact durability requirements (Design Input 11) and was prone to cracking when dropped.
Mitigation:	Structural reinforcement was introduced via internal ribbing, thicker shell walls, and elimination of unsupported overhangs.
Impact:	These physical design changes significantly reduced the probability of failure during drops. As a result, the residual risk was reclassified as ALARP, as further structural risk reduction would compromise portability or printability.

Battery Depletion

Initial Classification:	Unacceptable
Outcome:	Originally, mitigation relied on an LED indicator – inappropriate for BlindDeaf users.
Mitigation (Revised):	Integration of a vibration motor for tactile battery status alert + inclusion of mechanical fail-safes (e.g., a reset position after power loss).
Impact:	This solution directly addressed accessibility constraints and added meaningful user feedback. The residual risk was reclassified as ALARP due to improved alert perception and reduced likelihood of power surprises.

These outcomes not only reduced likelihood and improved user safety but also transformed design decisions. The battery vibration motor was a direct response to user needs. Similarly, casing enhancements affected CAD modeling and 3D print tolerances. These ripple effects ensured that risk controls were not isolated checkboxes, but integral to the evolution of the design.

These changes directly impacted the probability of failure occurring. For Structural Failure, the reinforcement plan significantly reduces likelihood of cracking and makes the casing more resilient under stress. For Battery Depletion, the tactile cue now provides a meaningful, perceivable alert system, satisfying usability requirements for the intended user population. As a result of these actions, both risks were re-evaluated and reclassified from Unacceptable to ALARP, which is consistent with ISO 14971's framework for residual risk.

The What?: Recommendation

Based on the comprehensive risk analysis conducted and the successful implementation of targeted mitigation strategies, it can be recommended that the Text-2-Braille device be deemed sufficiently safe for continued development, user testing, and eventual deployment. All previously unacceptable risks have been re-evaluated following mitigation and are now classified as ALARP, indicating that they have been reduced to a level that is tolerable and cannot be further mitigated without disproportionate effort or compromise to other design goals.

The Why?: Justification

Two risks were originally classified as unacceptable in Table 4b:

- *Structural Failure (S = 5, P = 4):* due to brittle PLA casing prone to fracture during 1-meter drop tests.
- *Battery Depletion (S = 4, P = 4):* due to BlindDeaf users being unaware of low battery states.

Mitigation strategies were designed and executed for both:

- For Structural Failure, the casing underwent major structural redesign. Internal ribbing, shell thickening, and optimized wall transitions were added to absorb impact and reduce stress concentrations. These physical changes directly lowered the probability of casing breakage, bringing the risk classification down to ALARP.
- For Battery Depletion, the originally proposed LED indicator was replaced by a vibration motor that tactically alerts users when battery levels are low. This solution was intentionally selected to match the accessibility needs of BlindDeaf users. Additionally, fallback mechanical cues such as default orientation post power-off were proposed for redundancy.

These mitigation actions did not just exist on paper, they had tangible impacts on the physical and user-centered aspects of the prototype. More importantly, they aligned directly with the design's core philosophy: enabling safe, intuitive, and independent use for a marginalized user group.

Limitations Associated with Risk Management

Risk management does not guarantee elimination of all possible failures. Even after all unacceptable risks have been reduced to ALARP, several inherent limitations remain. These limitations include issues of detection, implementation feasibility, and reliance on user behaviour. Structural reinforcements can reduce the probability of casing breakage, but do not eliminate risk when subjected to unanticipated impact vectors or edge landings. Additionally, while the use of vibration motors provides accessible feedback for battery depletion, such alerts are only effective if the user is physically in contact with the device, which may not always be the case.

Risk mitigation also faces constraints imposed by prototype limitations, including available components, space constraints, and power budgets. Some risks, although technically addressable, cannot be fully resolved due to limitations in manufacturability or integration complexity within the current system design.

Moreover, certain failure modes, such as comprehension failure, inherently depend on the user learning curve and cannot be fully mitigated through design alone. Risk analysis also relies on subjective estimates of severity and probability, particularly in early development phases where data is limited. This introduces variability in risk classification and underscores the importance of conservative evaluation.

The original FMEA tables lacked root cause identification and accessible user-centered mitigation strategies. The current system corrects these omissions but still acknowledges that residual risks exist and that further refinement may be necessary as user testing proceeds. Risk management, therefore, is an iterative process that evolves with the design, and not a static checklist.

Economic Analysis

Description of Alternatives

The idealized version (Option 1) of the Text-2-Braille is a portable and battery-powered device that can be carried around by DeafBlind users. Words can be scanned using an integrated camera and utilizes onboard AI that can run Optical Character Recognition (OCR) to identify printed or digital text. The created software is able to rotate four octagons that display the word in Grade 1 Braille.

The economic alternative (Option 2) is a stationary device that receives input data directly from a desktop computer and can translate given input text by the user or host software into Grade 1 Braille through the rotation of four octagons. This alternative removes the need for a camera, onboard AI, and a battery due to input text being manually received instead of OCR being needed and the power connection to the desktop computer.

The third option is to do nothing, in which nothing is done with the $20,000.00 original investment.

Documentation and Comparison of Finances

Receipts				
Description	**Type**	**Period (for singular)**	**Interval (for annuity)**	**Amount ($)**
Monthly Units Sold (144)	Annuity	n = 13 - 60	Monthly	$80,000.00
Bank Loan	Single	n = 0		$75,000.00
Gift Money	Single	n = 0		$20,000.00
Disbursements				
Description	**Type**	**Period (for singular)**	**Interval (for annuity)**	**Amount ($)**
Cost of Monthly Units Sold	Annuity	n = 13 - 60	Monthly	$28,800.00
Die Mold Cost	Single	n = 0		$35,000.00
Rent & Utilities	Annuity	n = 0 - 60	Monthly	$1,800.00
AI Software Engineer Salary (6 Months)	Annuity	n = 0 - 6	Monthly	$9,600.00

Software Developer Salary (12 Months)	Annuity	n = 0 - 12	Monthly	$6,400.00
Electrician Salary (12 Months)	Annuity	n = 0 - 12	Monthly	$3,200.00
Physical Labourer/Assembly Salary x2 (54 Months)	Annuity	n = 7 - 60	Monthly	$4,800.00
Futurpreneur First Year Payments	Annuity	n = 0 - 12	Monthly	$177.00
Futurpreneur Loan Payments 4.95% + 3% (With the spread covering to 9%) Using 8.495% As Average	Annuity	n = 13 - 60	Monthly	$615.00
Futurpreneur One Time Loan Management Fee of 1%	Single	n =0		$250.00
BDC First Year Payments	Annuity	n = 0 - 12	Monthly	$290.63
BDC Loan Payment 4.95% + 1.5% (With spread covering to 7.5%) Using 6.975% as Average	Annuity	n = 13 - 60	Monthly	$1,196.00
BDC One Time Disbursement Fee	Single	n = 0		$50.00
Our Wages x5 (60 Months)	Annuity	n = 0 - 60	Monthly	$12,000.00

Table 4. Monthly finances of Option 1 (Idealized) in tabular form.

Receipts				
Description	**Type**	**Period (for singular)**	**Interval (for annuity)**	**Amount ($)**
Monthly Units Sold (144)	Annuity	n = 13 - 60	Monthly	$68,000.00
Bank Loan	Single	n = 0		$75,000.00
Gift Money	Single	n = 0		$20,000.00
Disbursements				
Description	**Type**	**Period (for singular)**	**Interval (for annuity)**	**Amount ($)**
Cost of Monthly Units Sold	Annuity	n = 13 - 60	Monthly	$24,480.00
Die Mold Cost	Single	n = 0		$35,000.00
Rent & Utilities	Annuity	n = 0 - 60	Monthly	$1,800.00
Software Developer Salary (12 Months)	Annuity	n = 0 - 12	Monthly	$6,400.00
Electrician Salary (12 Months)	Annuity	n = 0 - 12	Monthly	$3,200.00
Physical Labourer/Assembly Salary x2 (54 Months)	Annuity	n = 7 - 60	Monthly	$4,800.00
Futurpreneur First Year Payments	Annuity	n = 0 - 12	Monthly	$177.00
Futurpreneur Loan Payments 4.95% + 3% (With the spread covering to 9%) Using 8.495% As Average	Annuity	n = 13 - 60	Monthly	$615.00

Futurpreneur One Time Loan Management Fee of 1%	Single	n =0		$250.00
BDC First Year Payments	Annuity	n = 0 - 12	Monthly	$290.63
BDC Loan Payment 4.95% + 1.5% (With spread covering to 7.5%) Using 6.975% as Average	Annuity	n = 13 - 60	Monthly	$1,196.00
BDC One Time Disbursement Fee	Single	n = 0		$50.00

Table 5. Monthly finances of Option 2 (Alternative) in tabular form.

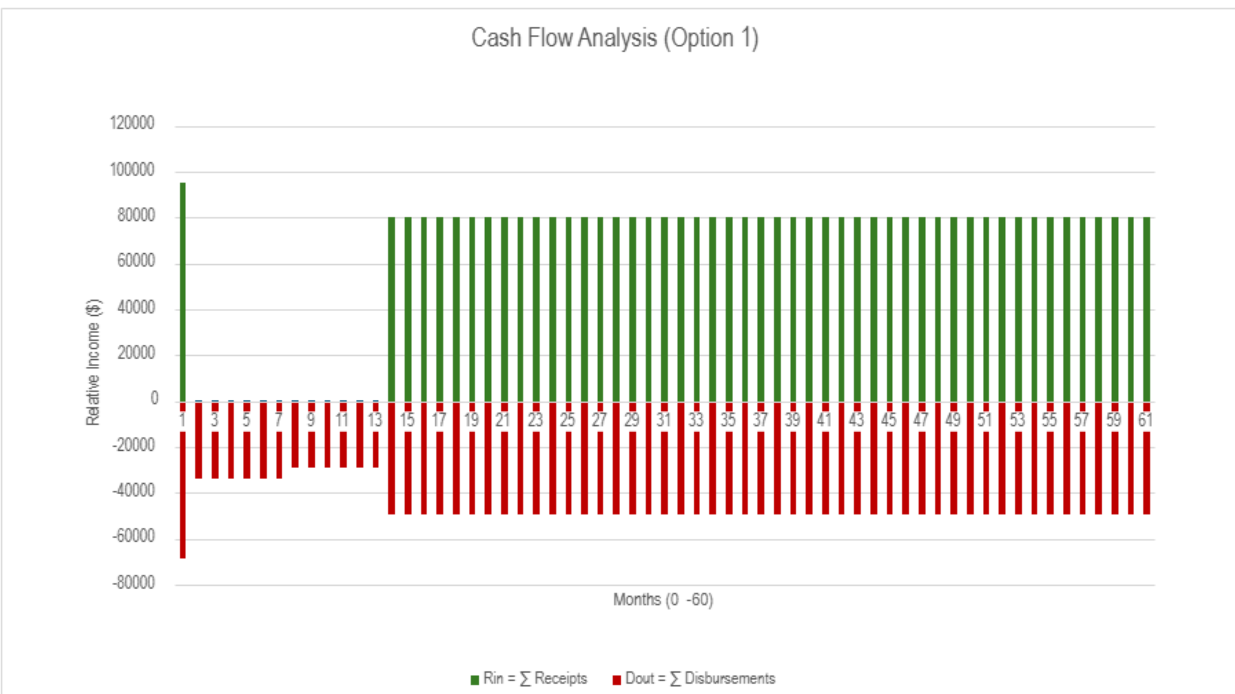


Figure 11. Cash flow diagram for Option 1 (Idealized).

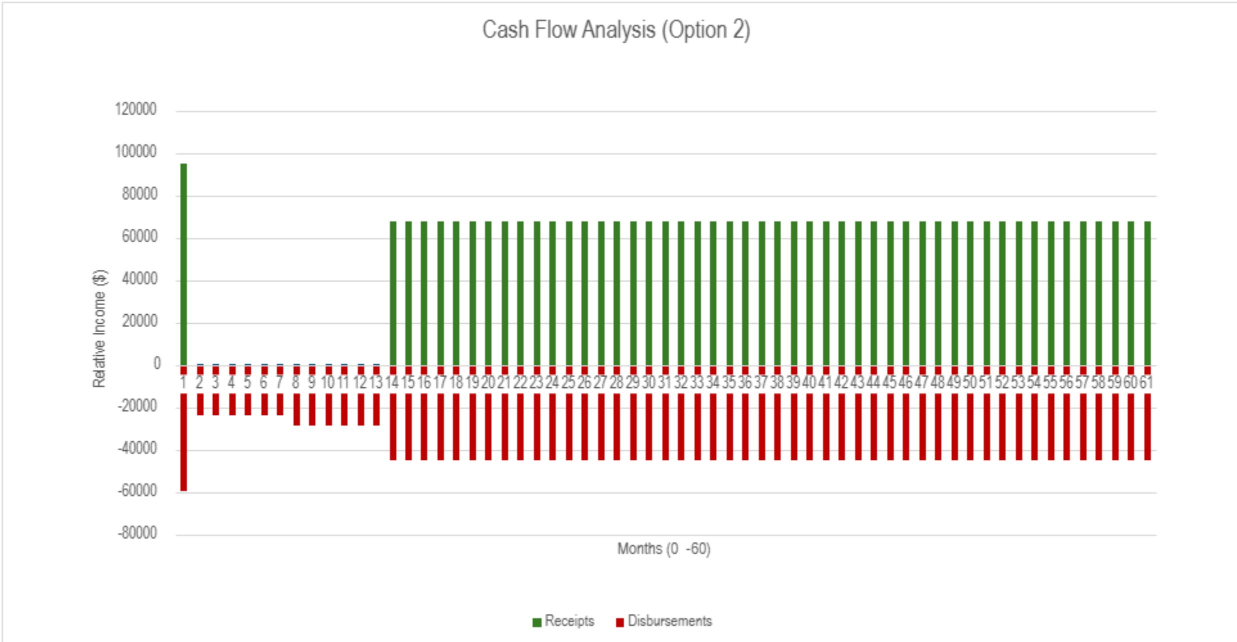


Figure 12. Cash flow diagram for Option 2 (Alternative).

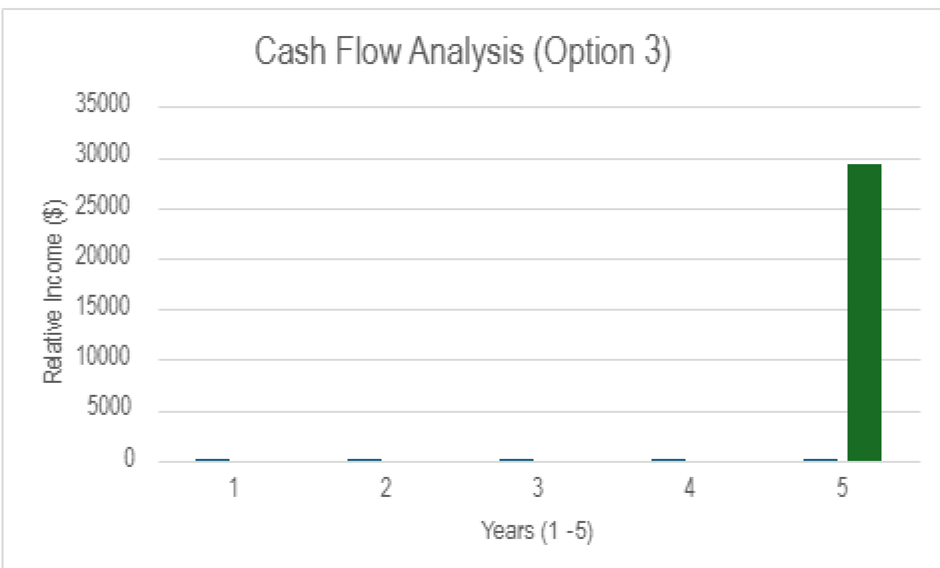


Figure 13. Cash flow diagram for Option 3 (Do Nothing).

As seen statistically in Tables 4 and 5 and visually in Figures 11 and 12 summarizing monthly cash flows for both the idealized and alternative versions of the product, the key economic reduction for the alternative lies in the lower monthly cost of unit production due to the lack of camera, AI and battery, which saves $30.00 per unit or $4,320.00 per month. Additionally, there is no need to hire an AI software engineer for 6 months which saves $9,600.00 per month for the first 6 months or $57,600.00 over the 5-year period.

Figure 13 is a brief cash flow diagram for Option 3 (Do Nothing), in which no cash flows in or out for the first four years. The future value after 5 years then flows in at the Minimum Acceptable Rate of Return (MARR) of 8%.

Explanation of Assumptions

Some assumptions can be found in both Tables 4 and 5 that directly pertain to the monthly cash flow. The wages for software developer, electrician, AI software engineer (exclusive to Option 1), and physical labourer/assembly worker were all set based on average salaries of those jobs accounting for job market localization and demand and assuming no taxes, vacation pay, or bonuses. Both Options assume the Futurpreneur and BDC loans would be accepted, with the loan terms held that only the monthly interest would be required to be paid in the first year during R&D and production phases. The principal amount along with the average interest rates from the listed spreads would then be paid off monthly at an assumed constant rate for the remaining 4 years.

Other assumptions that can be appropriately listed pertain to simplifying both the variable nature of a start-up and profitability comparison calculations. No marketing expenses would be made as it does not fall within the scope of this start-up business. All fabrication equipment needed to die-cast molds for the plastic injection molding would be readily available for use. Vendor quotes stay consistent over the 5-year horizon to simplify production costs and subsequent revenue. It is assumed that no defective units are produced, which is unlikely over the 5-year span but not realistic to account for and balances out when both alternatives make the same assumption.

Profitability Comparison

Present Worth (PW) is the chosen method of profitability comparison used for the two alternatives of Text-2-Braille. It allows for the direct comparison of both alternatives by using the MARR of 8% to give a direct ranking of both alternatives.

	PW ($)	PW_{net} ($) (accounts for 20% disbursement after 5 years)
Option 1 (Idealized)	$1,528,948.12	$1,223,158.50
Option 2 (Alternative)	$1,125,695.75	$900,556.60

Table 6. Present worth summary of Option 1 and 2 in tabular form.

Table 6 is the results of the PW calculations done on the Excel spreadsheet. Both the Present Worth and Net Present Worth that accounts for the 20% disbursement after 5 years is relatively higher than for the Idealized version of Text-2-Braille compared to the alternative stationary device. This means that at present worth, choosing the portable device that has onboard AI and battery life would mean a net worth difference of $322,601.90 in the present.

This significant difference in Present Worth between the Idealized and Alternative outcomes highlights the long-term value of the more technologically advanced and versatile solution. The portability and AI capabilities may also contribute to broader user accessibility and adoption. Option 1 proves to be more profitable but also potentially more impactful in terms of usability, innovation, and accessibility.

However, this does not account for the higher complexity and development costs that the idealized version would entail. The alternative version is a financially viable and relatively lower-risk option for undergraduate engineering students to have a feasible chance at pursuing. Considering this, Option 2 (Alternative) was chosen as the most appropriate and achievable design path with economics considered.

The What?: Recommendation

Overall, although Text-2-Braille has a great potential for economic benefit and can help the DeafBlind community, there is an immense time commitment required for the product to be truly successful, along with initial investment risks. Based on the economic analysis of different alternatives to the design path but considering monetary risk and time investment, it is recommended that the Text-2-Braille device should not be pursued from an economic analysis standpoint as students.

The Why?: Justification

Overall real world limitations must be considered in addition to the monetary economic analysis represented by the PW calculations. As five undergraduate engineering students with busy schedules and difficult coursework, it may not be feasible to commit the required resources in terms of time and energy to help the business reach its calculated economic projections. Considerations of natural co-op cycles would need to be factored in as well, further increasing the complexities of free time needed to be allocated to various areas of the project timeline, specifically in the initial R&D phases that would help optimize the product and give it the greatest chance for commercial success.

Limitations Associated with Economic Analysis

The assumptions that were made in order to simplify calculations and compare profitability do not fully consider that a start-up business is based on more than just economic profitability. It is realistically unreasonable to assume a completely constant rate of producing and selling of units to provide a direct monthly profit. The assumption of no defective units may be economically viable for a surface-level economic comparison, but should not be expected in a real working environment where manufacturing processes over time will require maintenance. These maintenance costs are variable and were not considered in the economic analysis, becoming another source of unconsidered risk.

With such few employees hired in a small company, there is little room for error which can induce stress within the staff. This can become a contributing factor in overall efficiency within the workplace, a concept that an economic analysis is unable to analyze.

Material availability and the changing prices of electronic components has also recently become an extremely volatile issue as well when considering the trade wars going on between USA and China, especially considering electronic devices as the main US import from China [3]. As the electronic components such as the ESP32 microcontroller or miscellaneous wiring can potentially be outsourced from the USA based on supply needs, this creates yet another source of risk that is temporally sensitive if production were to start in less than 6 months from now.

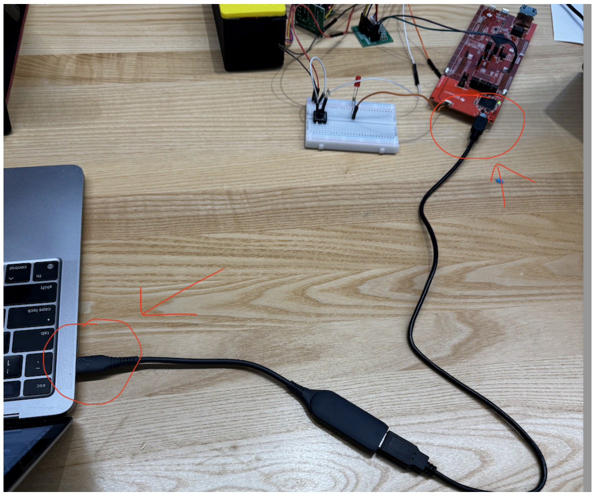
All of these factors show that the economic analysis performed is not able to capture the constantly changing conditions of the real world, as well as the time and dedication required to make a start-up product truly economically viable to a group of five undergraduate engineering students. Economic analysis is able to provide a direct comparison of economic benefit, but is unable to consider both the natural fluctuations caused by the entropy of real life and the intangibles that are required to make this start-up worth pursuing.

References

[1] “Interview with Industry Professional and Blind Person,” Group 32, Mar. 08, 2025.
[2] Cornell University, “16.1 Electrical Safety.” https://ehs.cornell.edu/book/export/html/1441
[3] B. Chu, “What would a US-China trade war do to the world economy?,” BBC, Apr. 08, 2025.

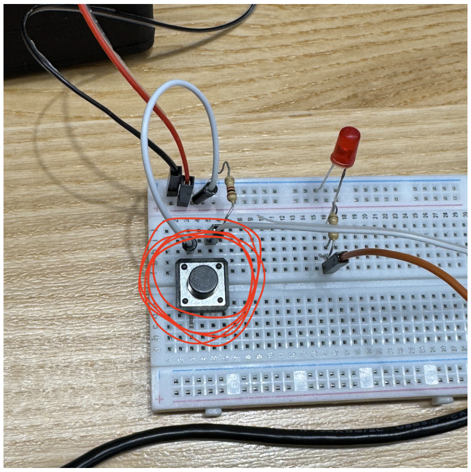
Appendix A: Experimental Protocol

Power Up the Device



   a. Connect the machine to a power source (from devices micro-usb port to your desktop).

2. Open the Arduino IDE
   a. Launch the Arduino IDE on your computer.
   b. Make sure the correct board is selected:
      i. Go to Tools > Board > ESP32 Dev Module.
   c. Make sure the correct port is selected:
      i. Go to Tools > Port and select the one corresponding to your ESP32.
   d. Turn on the Text-2-Braille Machine
      i. Ensure the text-to-braille device is powered on and ready.
   e. Upload the Code to the ESP32
      i. In the Arduino IDE, press the Upload button (right arrow on the top left).
      ii. Wait until the code is successfully uploaded.
3. Open the Serial Monitor in the Arduino IDE
   a. After uploading, open the Serial Monitor.
4. Input a Word
   a. In the Serial Monitor input bar, type a 2-, 3-, or 4-letter word, then press Enter.
   b. The motors will spin and adjust to the Braille representation of each letter in the word.
5. Read the Braille Output
   a. Feel the raised dots to read the translated Braille characters.



6. Reset the Machine
    a. Once you're done reading, press the reset button on the machine.
    b. The motors will rotate back to the default position, with the empty sides of the octagons facing up.
7. Repeat the Process
8. You can now input another word and repeat from Step 6