Braille Study Aid

ECE 445 Final Report - Spring 2021

Team 36:

Luis Aragon (laragon2@illinois.edu)

Emma Erickson (emmaee2@illinois.edu)

Aris Karnavas (arisnk2@illinois.edu)

TA: Dean Biskup

Abstract

We implemented a five letter Braille study aid, capable of outputting any preloaded five letter word in Braille characters as well as over speakers. The device is designed to give new Braille readers a way to study easily and without the presence of a tutor. The Braille output consists of five 28-sided gears with one braille character per face, as well as a blank reset face. To use, the student hits the "Next Word" button, and a cover lowers over the gears, which spin into the position of a word stored on the device. The cover rises and the user may read the Braille characters, then hit the top "Speaker" button to hear the correct word displayed. Our prototype holds and is capable of cycling through 50 words and audio clips for demonstration purposes.

Contents

1. Introduction

1.1 Problem and Solution Overview

Braille is a tactile written language used by people with visual impairments [1]. Unfortunately, only 10% of blind children learn to read Braille due to a lack of access [2]. Many students are taught through a seeing teacher. The current issue with that process is that most seeing teachers are not familiar with Braille and they are not always available. We want to provide an easier way for a student to learn independently. The study aid will allow for a quick succession of practice at the user's own pace while providing audio confirmation on each word. The aid will cycle between a hundred introductory words, displaying Braille and auditory representation.

There are currently expensive Braille e-readers on the market, but they are targeted more to extended reading and not teaching the language [3]. Our design improves on competing braille teaching methods by offering more flexibility than purely mechanical blocks and books. Though there are more complex Braille readers, they are expensive and specialized for leisure reading, not studying. Our solution offers a flexible learning opportunity without requiring a seeing or Braille familiar tutor.

1.2 Visual Aid

'Figure 1' represents how our Braille Study Aid is intended to be used. A user will run a finger across the character display, reading the embossing. After realizing the word, the user can hit the second button to confirm the word with the speaker. The speaker will output the given word to the user. When the user is ready, they can click the upper button to progress to the next word in the sequence. The protective cover will lower and shield the user from the recalibrating gears. Once the gears configure to their new positions, the cover will rise and the user can repeat the process with a new word.

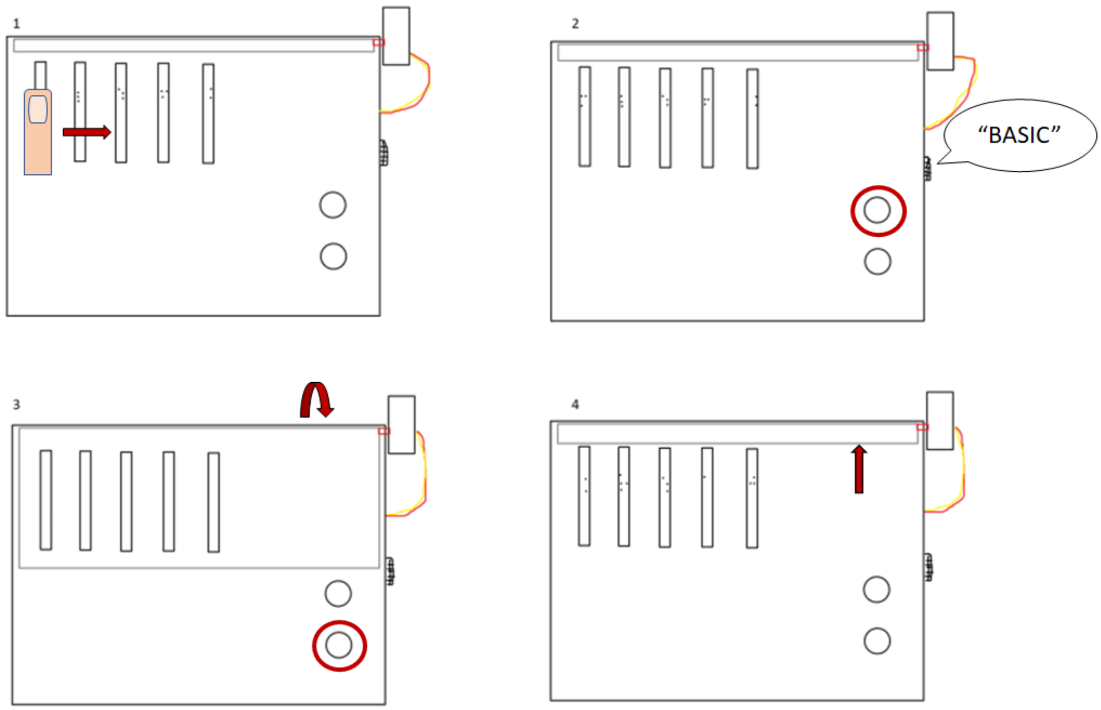

Figure 1: A Visual Representation of how a User will Interact with the Aid

1.3 High-Level Requirements List

1.3.1: The study aid must display five braille characters.
1.3.2: The study aid must provide audio for the word being displayed upon button push.
1.3.3: The study aid must allow the user to switch to the next word within 10 seconds.

2. Design

2.1 Physical Design

The following figures are a physical representation of our design. The design includes five slots for Braille characters to be shifted out to the user. Our gears will have 28 sides, to encompass the 26 Braille characters, a numeric symbol, and a blank tile [1]. Our design also includes an audio jack to output an audible representation of the currently displayed word. Two buttons allow the user to cycle between new words and queue the speaker to provide audio feedback. All moving parts and wiring are contained in a casing. In the casing, our initial design segregated a majority of our parts to a PCB, with wires connecting all parts and motors. Due to PCB failures, we transitioned the interior logic onto a breadboard.

The external user view of the study aid is displayed in ‘Figure 2’. A closeup of our design for our Braille Gears is represented by ‘Figure 3’. Each gear will be 89.3mm in diameter and each tooth that a Braille character will rest on will be 10mm long. In order for our stepper motors to be accurate the angle between gear faces or in our case the angle between adjacent letters is required. In order to calculate this angle we divide 360 by 28 which is the number of faces we have to achieve an angle of 12.85° and is also represented in ‘Figure 4’ through a CAD drawing and measurement.

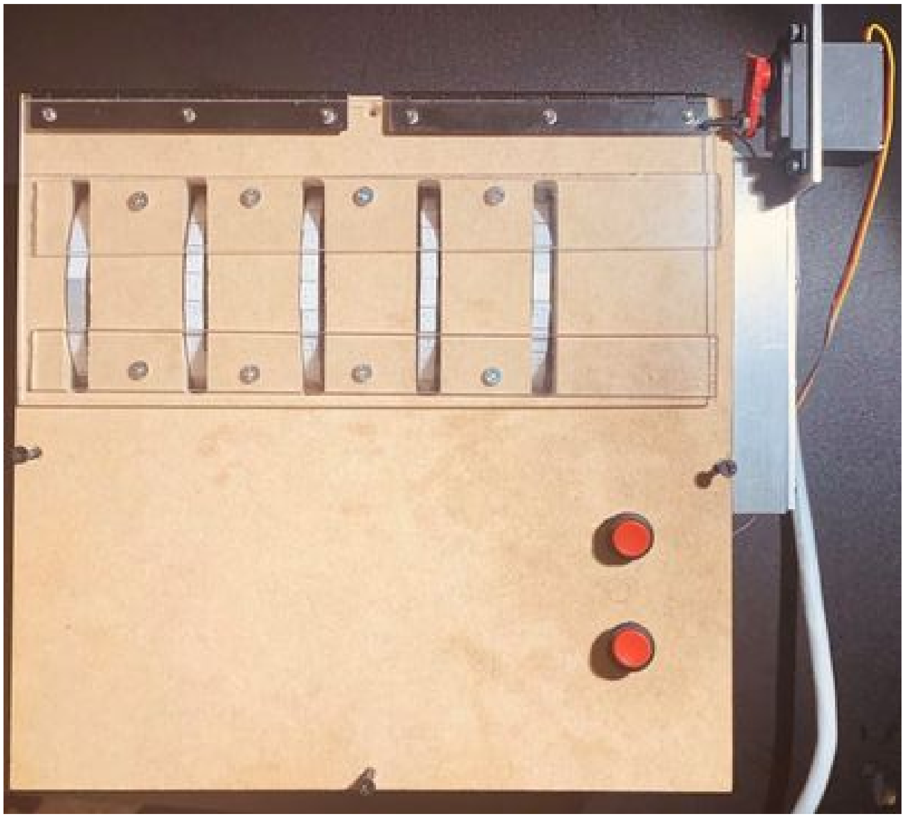

Figure 2: Exterior Design of the Braille Study Aid

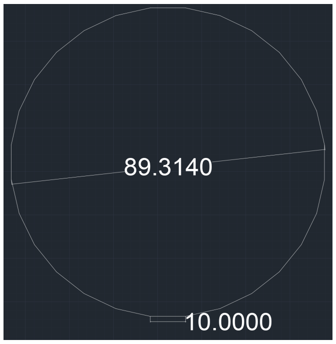

Figure 3: CAD Representation of 28 Sided Character Gear (mm)

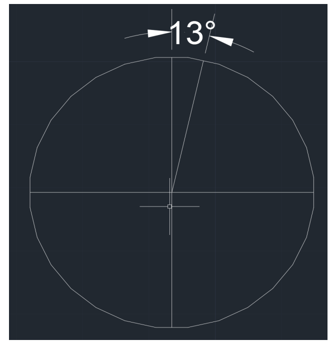

Figure 4: CAD Representation of Angle between Gear Faces

2.2 Block Diagram

In order to implement our Braille teacher to the specifications described, we required four main modules: power, control, audio, and Braille output. The power module converts AC wall power to the DC power specifications of both the controller and motors. The control module converts user input into instructions for the motors and speaker, positioning the motors appropriately and activating the speaker at the user's request. The audio module is responsible for reading out the word being tested. Finally, the Braille output module displays the tested word in Braille using the gear mechanism. It includes a protective cover to prevent confusion or interference as the gears move between words. The interactions between modules are described in 'Figure 5'.

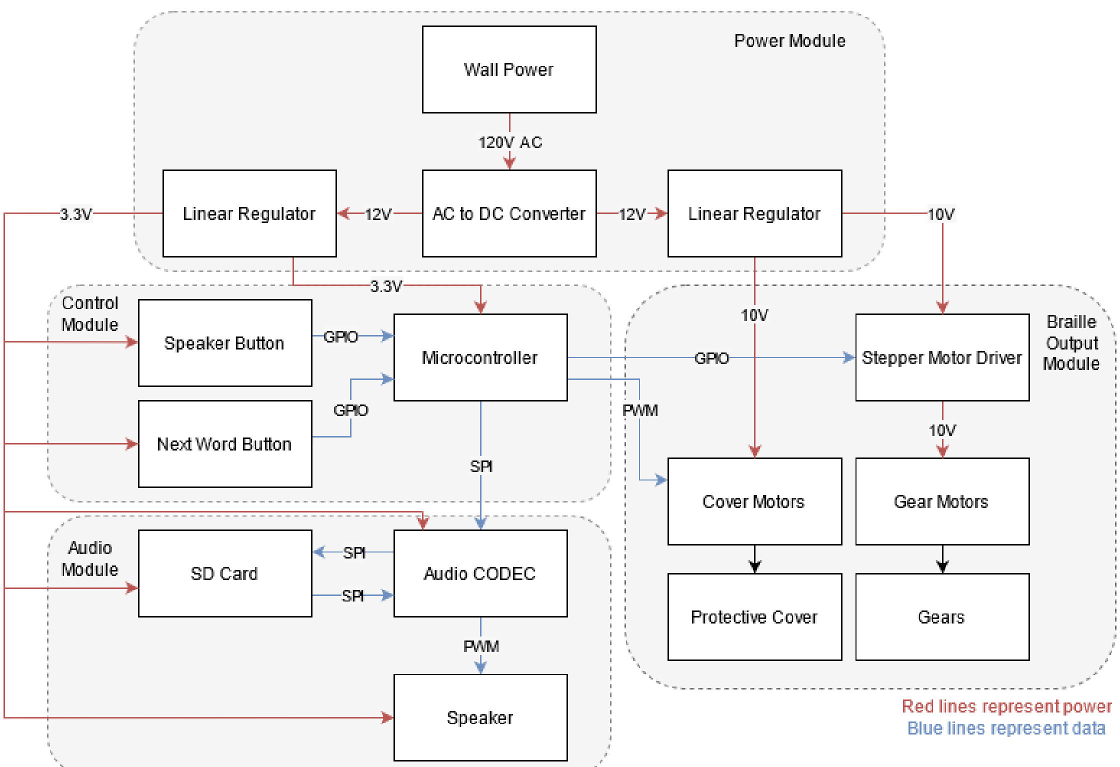

Figure 5: Braille Study Aid block diagram

2.3 Module Designs

2.3.1 Power Module

The power module takes in 120V AC from the wall socket and utilizes an AC-DC converter to bring this down to a usable DC voltage that will supply power to microcontroller, motors, and speaker. This delivered 12V with a small ripple continuously to the PCB which in the original design was stepped down to 3.3V, 5V, and 10V, to supply the microcontroller, cover motors, and motor drivers respectively. The power module circuit schematic is below in 'Figure 6'.

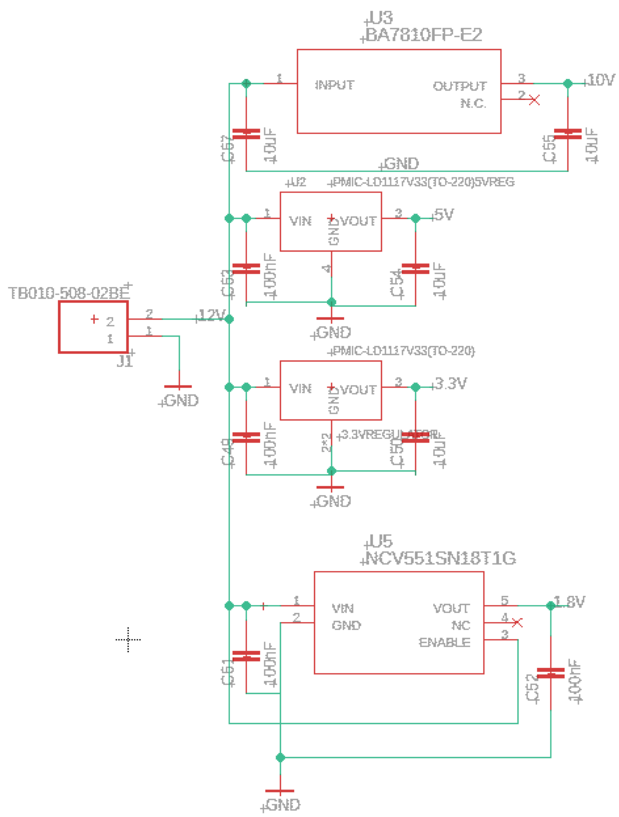

Figure 6: Power Module CIrcuit Schematic

In reality, the PCB failure and an internal short in the motor drivers which would have required us to operate either the driver or gear motors on voltages outside of their specified ranges prevented us from using this full system. Instead, we substituted our drivers for 5V ones matching our motors, and utilized only the 5V voltage regulator in our final design.

2.3.2 Control Module

The microcontroller is responsible for communicating motor orientations necessary to form a Braille word when the next word button is pressed. It also sends the audio CODEC a signal for the current word when the speaker button is pressed. Our design called for a single microcontroller, requiring one such as the ATMEGA328PB-AN, pictured in 'Figure 7', due to the many pins required by each motor driver. The control module also communicated with the speaker through the CODEC to play audio when the speaker button input was pressed [4].

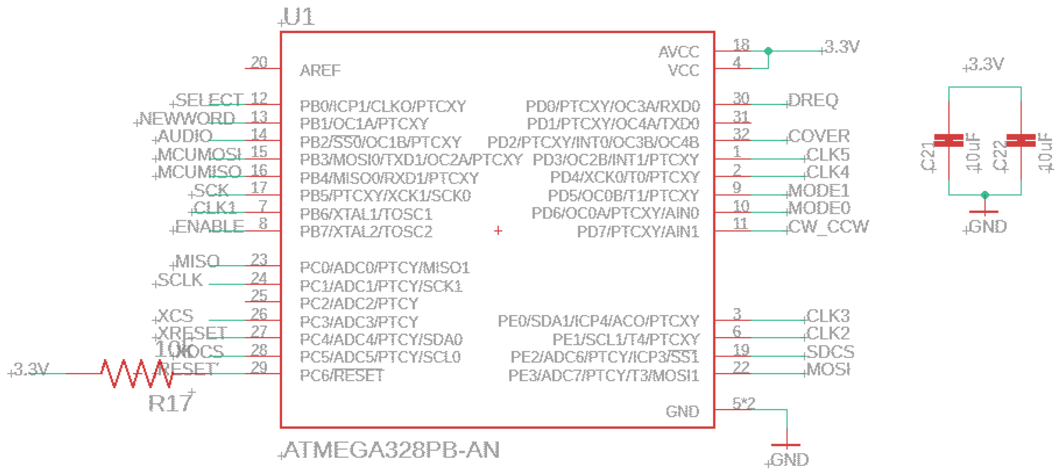

Figure 7: Control Module Circuit Schematic

Though our original microcontroller chip burned out, our code was flexible enough to run on other chips. We eventually settled on the ATmega2560 on the Arduino Mega, which had enough IO pins to communicate with all of our motors and audio at once. The design could also be run on several smaller microcontrollers, each responsible for a single motor. Though this would increase costs, it would have the advantage of parallelizing our gear motors, speeding up Braille display time by a factor of five.

The main outline of our code can be viewed in ‘Figure 8’. The main function begins with the “Next Word” flowchart (left) then moves onto the “Speaker” flowchart (right), looping while the device is plugged in.

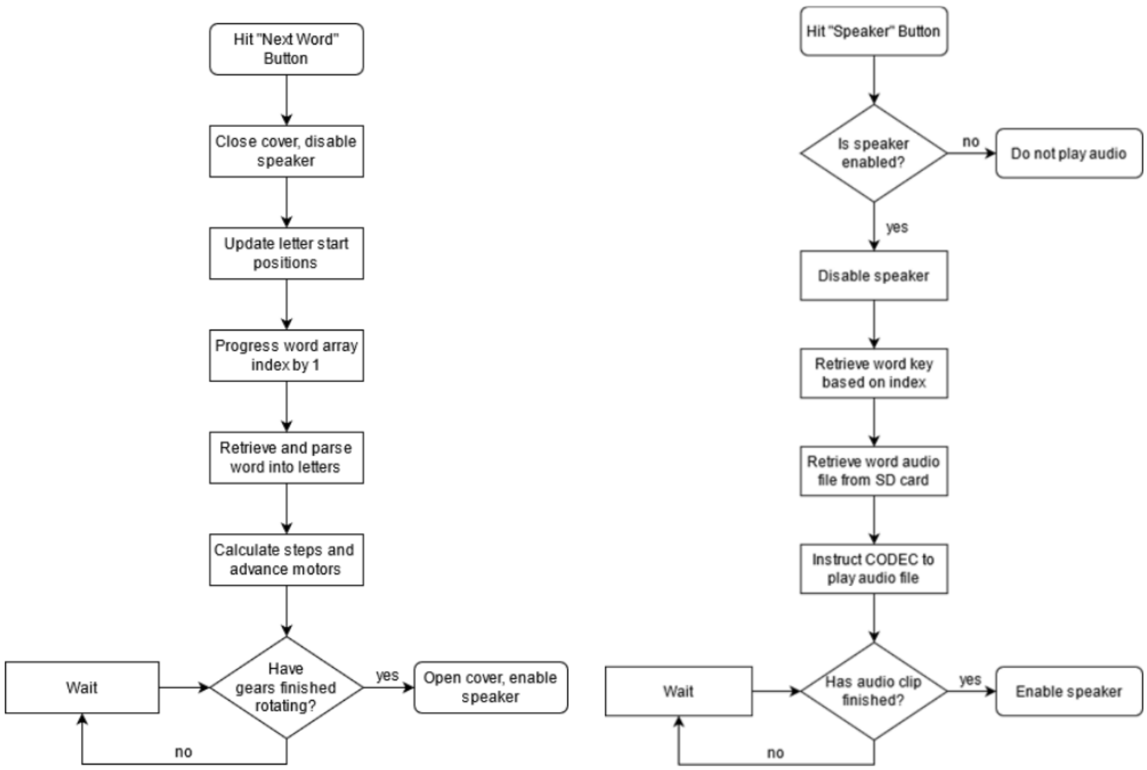

Figure 8: Code Structure Flowcharts

2.3.3 Audio Module

The audio module takes signals from the microcontroller to output what is currently displayed on the gear display audibly. The speaker will be controlled by an audio CODEC that receives messages from the microcontroller for when to output audio that is stored on an external SD card. This must output audio at least 8kHz to encompass the frequency content of human voice [4]. This subsystem will be powered from the 3.3V and 1.8V rail. 'Figure 9' below shows the circuit diagram for the audio module consisting of the CODEC and control/communication signals from the microcontroller. The same generic layout can be used for both speakers and audio jacks, and for our demo we used an audio jack because the sound quality was better.

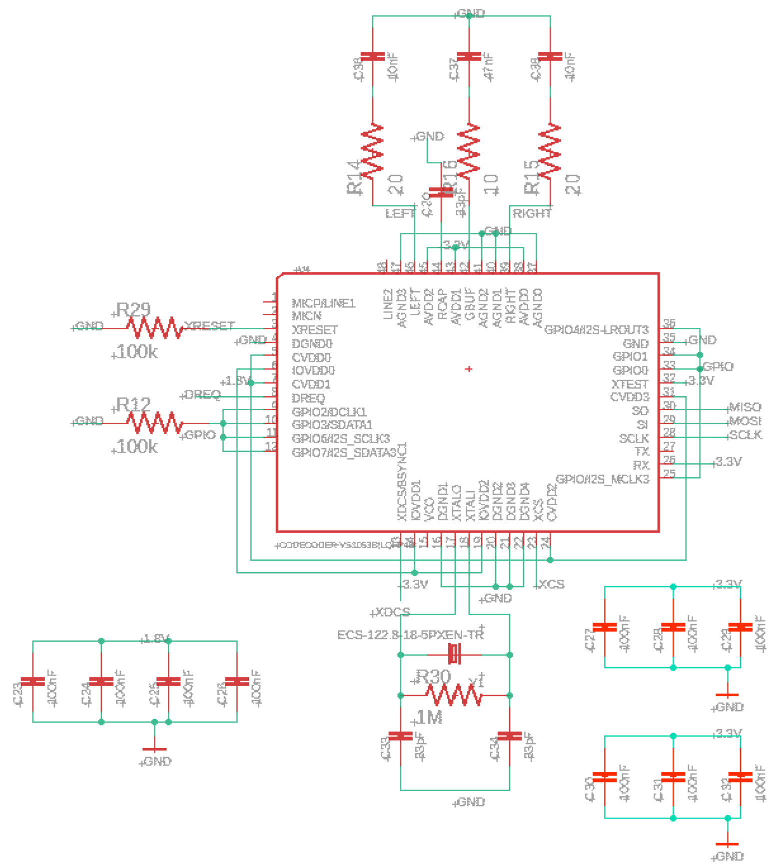

Figure 9: Audio Module CIrcuit Schematic

2.3.4 Braille Output Module

The motors take signals from the microcontroller through a stepper motor driver detailing what position each gear needs to be to display the next word. ‘Figure 10’ below displays 2 out of the 5 motor driver circuit schematics to show connections and control signals needed from the microcontroller. During the design phase it was noticed that we can cut down on IO needed for the drivers by having all the drivers share a majority of the control signals and thus only need a distinct clk signal that determines the number of steps needed for the motor to turn. Upon a new word button press the protective cover motor will move out to prevent the user from accessing the spinning gears and the cover will retract once new word is in position. The stepper motors are capable of rotating 360 degrees and can stop at 28 equally spaced points along this rotation [5].

The 5 gear motors receive signals from the microcontroller and turn the gears an appropriate amount to display the correct Braille symbol. The motors also receive power via the AC/DC power converters from the Power Module. The motors need to operate under relatively precise constraints, given the 28 characters and small radius of gears. The motor rotates in two directions, and the program selects the shortest to the next word to cut down on transition time. This is an improvement from our original design, where motors would reset between each letter.

The character gears accommodate 28 braille cells along their circumference. According to standards [6], each Braille character, or cell, is 7.7mm high, which we will round up to a 10mm side length to accommodate some space above and below the dots. Given the dimension of a single Braille cell, the circumference of the gear came out to approximately 8.9cm.

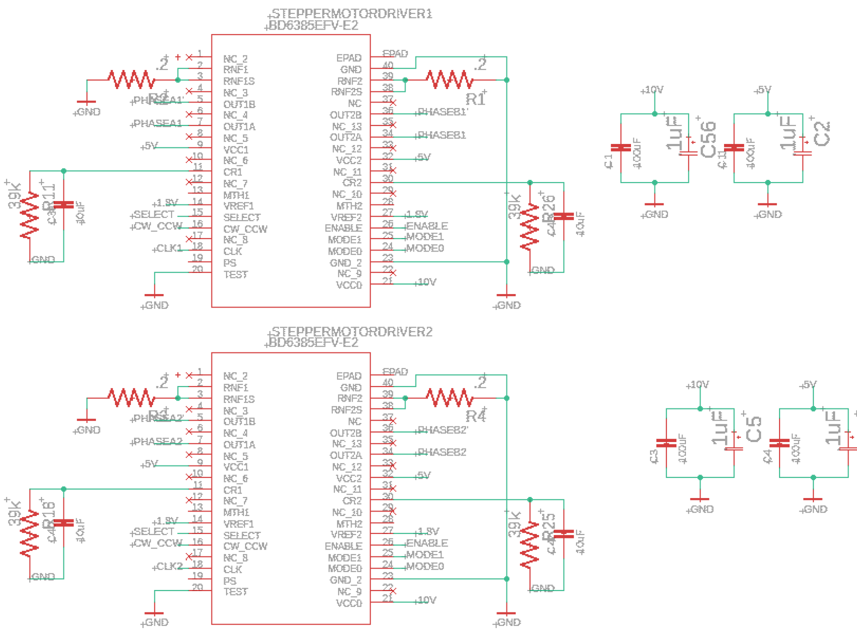

Figure 10: Braille Output Module CIrcuit Schematic

Finally, the protective cover lays over the interactive Braille character space when the gears are in motion. The cover motor is a servo which moves between two positions - perpendicular to and flush with the display area.

3. Design Verification

Our final verification table can be found in appendix A. As our project evolved, visual confirmation over different test cases became extremely important to verification. Because specific part numbers used in our subsystems changed due to blown chips and shipping delays, few of our tests are specific to one any one component. Given more time on the project and in the lab, oscilloscope and multimeter level confirmations of values would have been collected. In the absence of this, we used test cases of different behaviors as our main verification scheme.

3.1 Visual Verification

The following is a list of our visual tests to verify the effectiveness of our project. Though these tests cannot be viewed as images, videos demonstrating the following functionalities are hosted within a public Google Drive folder [7].

3.1.1 Audio Module

Pressing the speaker button down will play the audio clip matching the displayed Braille output. Clicking the button up does nothing. If the speaker button is pressed while the gears are turning, it produces no audio output. If the speaker button is pressed while the speaker is already playing, audio will not restart.

3.1.2 Braille Output Module

Pressing the next word button will activate the gears, which will realign to the next word in the queue. The cover motor begins in a closed position when plugged in. When the next word button is pressed, it will cover the spinning gears, then open to display them.

3.2 Timing Test

With our high level requirement 1.3.3 binding the study aid to transition between words in under 10 seconds we performed a timing test to measure whether this requirement is met. We conducted this test by randomly selecting a word from our dictionary upon each “new word button” press. This allows us to accurately test our device rather than cycling through our predefined list of words that could relax the time between words. We then measured the time between 15 different word transitions whose results are in ‘Table 1’ below.

Table 1 Timing Test

From	To	Time
Doors	Final	9.13
Final	L	11.16
L	Y	5.69
Y	A	7.74
A	C	3.57
C	Cream	11.12
Cream	Y	14.59
Y	Call	12.24
Call	Final	10.02
Final	Adult	9.82
Adult	Day	11.18
Day	C	8.41
C	Belly	13.18
Belly	Doors	8.56
Doors	Call	11.24

Table 1: The next word timing transition breakdown can be viewed here

The result of this test gave an average transition time of 9.84 seconds. While our high level requirement isn't met for every single transition between words, our average time to transition from one word to the next is just under our requirement.

4. Costs

4.1 Parts and Labor

Going by industry standards, our fixed development costs will be set at $33.65, 10 hours/week, for an estimated 8 work weeks, for three people. This will work out to a labor sum cost of $20,190. Our part costs can be seen in 'Table 2', resulting in $162.37. The total cost will come out to $20,352.73

Table 2 Parts Costs

Part	Manufacturer	Retail Cost ($)	Bulk Amount	Actual Cost ($)
Power Supply	MEAN WELL	$16.00	1	$16.00
Stepper Motor	Pololu Corporation	$13.81	5	$69.05
Servomotor	Adafruit Industries LLC	$5.95	1	$5.95
Speaker	Adafruit	$4.95	1	$4.95
Microcontroller	Microchip Technology	$1.51	1	$1.51
Linear Regulator	STMicroelectronics	$0.59	1	$0.59
Linear Regulator	Rohm Semiconductor	$0.72	1	$0.72
Buttons	NTE Electronics Inc	$2.13	2	$4.26
Terminal Block	CUI Devices	$0.80	1	$0.80
Terminal Block	CUI Devices	$0.95	7	$6.65
Stepper Motor Driver	Monolithic Power Systems Inc.	$2.95	5	$14.75
Audio CODEC	Adafruit Industries LLC	$12.50	1	$12.50
PCBs	PCBWay	$25.00	5	$25.00
Total				**$162.37**

Table 2: The costs for our part orders can be viewed here

5. Conclusion

5.1 Accomplishments

Our final design delivered a prototype capable of meeting our three high level requirements. Unfortunately the prototype didn't include our designed PCB due to board malfunctions leading up to the demo. The prototype is thus controlled by an arduino holding a microcontroller unit similar to our proposed design so any written code can be ported over to PCB if manufactured. Upon each "new word button" press our protection cover shuts to allow our gears to spin to the next word. The "play audio" button successfully outputs auditory feedback to the user of the currently displayed word. Thus requirements 1.3.1 and 1.3.2 are met with this description.

5.2 Uncertainties

Under the speed verification test our high level requirement 1.3.3 isn't met for all word transitions with some taking as long as 14.95 seconds. The reason for the timing requirement not being met is we can only communicate with one motor at a time with our serial communication with one microcontroller. If we wanted to meet our timing requirement we would need to have 2 or more gears moving simultaneously which would either need parallel programming or having multiple microcontrollers to divide the work of the gear control. Another possible solution would be to implement an RTOS into our code which would allow for multiple tasks to run simultaneously, but to account for this a different design choice would be needed to use a microcontroller unit with ARM cortex. This would result in our code simplicity of using Arduino IDE to not be viable since we would need to use an IDE that can program the ARM core of a microcontroller such as Code Composer Studio (CCS).

5.3 Ethical considerations

With a predetermined catalogue of words being imputed to our study aid, we are eliminating any potential ethical issues regarding inappropriate words, such as profanity or harassment, from being taught. This mitigation aligns with IEEE Code of Ethics, 7.8.II: "To treat all persons fairly…" [8].

Safety concerns for the product are also taken into consideration. While testing, the linear regulators of the PCB grew hot to the touch. These are kept enclosed in the interior of the box, as with all electrical and power components. We also implemented a cover system to avoid pinching from the spinning gears.

5.4 Future work

The most immediate next step necessary on our project is the implementation of our final circuit on a PCB. While this was not possible at demo time, experiences with different parts and failures improved our decision making on necessary components and layout. Implementing this would allow us to greatly reduce the footprint of our device.

Beyond this initial step, a few changes to our design would certainly improve user experience. First, more permanent, for example 3D printed, gears would be necessary for a viable project. Second, changing our gear control strategy to accommodate the motion of multiple gears at a time will reduce latency between words. Finally software improvements, namely making our dictionary of possible words longer or more modular with code that allows users to input their own words.

The foundation of our project, a device capable of outputting paired Braille and audio at a user's request, was a success, and provided a solid base for future improvements to the design.

References

[1] afb.org, “What Is Braille?”, 2016. [Online]. Available: https://www.afb.org/blindness-and-low-vision/braille/what-braille. [Accessed: 3-26-21].

[2] nfb.org, “Blindness Statistics”, 2019. [Online]. Available: https://www.nfb.org/resources/blindness-statistics. [Accessed: 14-Feb-2021].

[3] bbc.com, “Braille e-books: Why can’t you buy a budget e-reader?”, 2014. [Online]. Available: https://www.bbc.com/news/technology-27243376. [Accessed: 3-23-2021].

[4] goodhousekeeping.com, “This sound says a lot about your hearing”, 2016. [Online]. Available: https://www.goodhousekeeping.com/uk/health/health-advice/a559784/hearing-loss-ageing-ears/. [Accessed: 3-24-21].

[5] digikey.com, “Pololu Corporation 1208”. [Online]. Available: https://www.digikey.com/en/products/detail/pololu-corporation/1208/10449951?fbclid=IwAR21ygSSuMg_LOE0TeArDj6VJcryWPvqprZPPXZjN1zq3KNhlpYoLlMl9Cw. [Accessed: 3-24-21]

[6] brailleauthority.org, “Size and Spacing of Braille Characters”, 2008. [Online]. Available: http://www.brailleauthority.org/sizespacingofbraille/sizespacingofbraille.pdf. [Accessed: 28-Feb-2021].

[6] brailleauthority.org, “Size and Spacing of Braille Characters”, 2008. [Online]. Available: http://www.brailleauthority.org/sizespacingofbraille/sizespacingofbraille.pdf. [Accessed: 28-Feb-2021].

[7] Aragon, Erickson, Karnavas, “ECE 445 Sp21 Team 36 Project Videos”, 2017. [Online]. Available: https://drive.google.com/drive/folders/1n3D1Y8xv7j0og_kNh7db3WG7rLFEf58t?usp=sharing

[8] Ieee.org, "IEEE IEEE Code of Ethics", 2021. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 16-Feb-2021].

Appendix A Requirement and Verification Table

Our Requirements and Verification can be found in ‘Table 3’. The descriptions used match our established rubric for our demonstration, highlighting the areas of our project that we wanted to function. All requirements that were laid out were effectively met and verified during our demonstration and remain effective today.

Table 3 System Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. Power supply must step down 120V AC from wall socket to 12V DC up to 8.5A .	1. Power is supplied to the board as shown by LED. Can probe terminals to prove 12V DC.	Y
2. Regulator outputs appropriate voltage to its relative electronics.	2. Can probe terminals to show 5V is properly represented at its regulator.	Y
3. Audio Codec will receive the corresponding audio file from the SD card and relay it to the speaker.	3. Press the "Speaker" button and listen for an auditory response from the speaker. Verify that the spoken word matches the current gear orientation.	Y
4. Speaker will clearly vocalize the expected audio file from the Audio Codec.	4. Speaker will output the word once the Audio Codec reads from the current Braille orientation after the "Speaker" button is pressed.	Y
5. Gear motor will cycle between each Braille character, stopping precisely at the expected orientation. It will also return to its origin position after.	5. Press the "Next Word" button. Gears should spin to represent the correct Braille word. Gears will also spin an equidistant amount backwards to a flush origin. a. Each gear will stop at the expected orientation cleanly upon request. b. Each gear can perform the reset function before moving to the next position. c. All 5 gears can work in succession when requested.	Y

6. Cover Motor will rotate the Protective Cover in under 2 seconds to lie down during gear rotation.	6. Press the "Next Word" button. The Cover Motor will rotate to lower the cover flushly over the gears. Once the gears are oriented, the cover motor will raise the cover to its rest position. a. The cover motor will turn upon request to both static positions. b. The cover motor will operate under 2 seconds between static positions.	Y

Table 3: Our Requirements and Verification Table used during our demonstration