

# BrailleBud: Transitional Learning Tool from Pre-Literacy to Braille Literacy

Tatiana Jaimes, Mechatronics Engineering, Alina Santander Vinokurova, Mechatronics Engineering,  
August Rodriguez, Mechanical Engineering  
Vaughn College of Aeronautics and Technology, NY, USA  
tatiana.jaimes@vaughn.edu, alina.santandervinokurova@vaughn.edu, august.rodriguez@vaughn.edu

**Abstract**– *The present paper displays the design process and characteristics of the BrailleBud. The BrailleBud is a low-cost teaching tool that encourages pre-K through first grade children to learn the Braille alphabet and acquire letter recognition skills through interactive methods that stimulate their tactile and auditory senses. It is programmed with Arduino and includes passive infrared sensors (PIR), a DFPlayer, and 5-volt solenoids. The solenoids act as braille dots to display letters, depending on their pattern, while the speaker informs the child of the letter that is currently shown. The objective is to aid young children to progress from pre-literacy to literacy, in a fun and simple manner.*

**Keywords**-- Braille, solenoid, Arduino, initial learning

## I. INTRODUCTION

The history of Braille dates to the early 1800s, when Charles Barbier developed a unique system known as “night writing.” This system was based on a 12-dot cell, two dots wide and six dots tall, representing a letter or phonetic sound. Louis Braille then created a reading method based on a six-dot cell for the fingertip to take-up the entire unit with one touch [1]. Currently, this system is utilized to touch reading and writing for the blind in which raised dots contain equivalents for punctuation marks and provide symbols to show letter groupings.

According to the 2017 American Community Survey (ACS), approximately 568,202 children have vision difficulty, while 63,657 children, youth, and adult students are legally blind in educational settings. Pre-readers and primary Braille readers compose 24.2 percent of this statistical portion [2]. Pre-literacy plays a critical role for children to process the systematic language and stimulate their tactile and visual cortex. Nevertheless, access to Braille educational material is expensive. However, as of 2016, about a million visually impaired Americans, nearly one in five, live in poverty. This gives them less access to educational resources, leading to a lower education level than the general population [3].

### A. Need statement

There is limited access to educational Braille resources due to high prices, and non-targeted functions. Visually impaired and legally blind students have fewer materials to learn and review Braille, they have a lower educational level [4]. This project's objective is to close the literacy gap by introducing an innovative Braille block that teaches children the alphabet to ensure good fundamentals of pre-literacy; allow visually impaired and legally blind children to have equal access to

educational products and methods without paying shockingly high prices.

### B. Societal Impact

The objective of the project is to give back to the visually impaired community by providing an inexpensive Braille teaching tool that will help children and their families acquire the basic knowledge of Braille.

### C. Distinguishing Element

BrailleBud differs in price, interactive elements, and simplicity compared to other braille learning tools. Although similar devices implement wireless technology to record children’s progress, this feature considerably increases the cost of the device. On the other hand, BrailleBud is almost 1/5 of the price of this type of product while offering an engaging learning experience. Since it has two levels, the learning process is short, concrete, and constant. The focus of the device is to entertain and captivate children’s curiosity, while ensuring a compact housing for the elements. Whereas other products are made up of loose components, having families run the risk of losing important parts. BrailleBud is designed to create a dynamic learning environment through its compact, simple, and interactive design.

## II. MARKET COST

Educational and instruction material is found on “Teaching Students with Visual Impairments.” This site includes material resources for grades 1 to 3 Braille. Static pre-literacy tools, such as Flip-Over Concept Books, Pop-A-Cell, or Early Experience Collection books, cost up to USD 400, while electronic devices, like Brailiant Refreshable Braille Display, average USD 2,600. These tools have limited interaction, which make learning tedious for the child, and mainly focuses on only tactile sensory. In contrast, the BrailleBud aims to motivate children with positive affirmation and a dynamic device to learn Braille while sharpening tactile and auditory senses at the price of approximately USD 250.

## III. AUDIENCE

The BrailleBud is targeted for children from ages two to five as student motivation to accept Braille is inversely proportional to the child’s age when introduced. Children are more interested and less frustrated in learning the earlier they

Digital Object Identifier (DOI):  
<http://dx.doi.org/10.18687/LACCEI2021.1.1.639>  
ISBN: 978-958-52071-8-9 ISSN: 2414-6390

are exposed to Braille. Cognitive abilities, foundational concepts, finger sensitivity, and motor coordination are essential for the child to progress from pre-literacy (i.e., alphabet) to literacy (i.e., recognize words and read simple sentences). The main idea of the device is to encourage children to learn while playing and develop well pre-literacy skills to go onto literacy.

Learning through the Braille block is acquired thanks to repetition. This is a key learning aid that helps children transition a skill from the conscious to the subconscious [5]. Recognizing the alphabet becomes easier with time and is step one to recognize short words. The goal is to aid children and their families beginning to learn Braille.

BrailleBud implements 5-volt - 5V - solenoids whose pins act as braille dots. These solenoids are placed side by side along three rows to simulate a Braille cell. The Arduino Mega controls the outputs of the components, depending on the letter case. Additionally, a PIR sensor is added to keep track of the child's placement on the device. If the child is hovering over the braille cell, the speaker plays an audio congratulating them, and the program goes onto the following case. Otherwise, an audio encouraging the child to try again is played, and the program loops the case until it detects the child's movement over the cell.

#### IV. ENGINEERING REQUIREMENTS

BrailleBud must meet the following requirements to stand out among other Braille teaching tools in the market:

**Compact:** the design must contain all the electrical components (solenoids, resistors, transistors, speaker, sensors, and switch) in one "home" to reduce the probability of losing parts and affecting the functioning of the device.

**Light:** distribute and reduce the weight of the device's components by implementing 3D printed housing.

**Maneuverable:** can be taken anywhere on the go and easy for storage.

#### V. BUDGET

Price is an essential factor when choosing the most convenient resources for blind children, BrailleBud is projected to cost a fifth of the price of electrical learning tools on the market.

The budget breakdown for the project is viewed in table 1. This device will have a net worth of around USD 200. Nevertheless, the team expects that by implementing printed circuit boards, and manufacturing the parts needed, the price decreases by approximately 20 percent, making BrailleBud one of the cheapest electrical teaching tools in the market.

TABLE 1: OVERVIEW BUDGET FOR BRAILLEBUD

Part	Unit cost	Subtotal
3 x Breadboard	5.00	15.00
1 x Arduino Mega	37.95	37.95
18 x TIP120 Transistor	0.85	15.3
18 x 5 V Small Solenoid	5.00	90.00
18 x 220 Ω resistors	0.10	1.80
6 x 1N4001 Diode	0.19	3.42
1 x Speaker	1.95	1.95
1 x DFPlayer SD	3.35	3.35
4 x Battery 1.5 V	1.73	1.73
1 x SD card	1.99	1.99
PLA manufactured parts	10	10
40 pack Cables	2.99	11.96
Enclosed Battery Holder w/switch	3.19	3.19
IR Pyroelectric Infrared PIR Motion Sensor	3.50	10.50
Total	77.79	208.14

Considering that BrailleBud has a complex circuit assembly due to the number of components and wires used, and a high chance of loosening their connection to the breadboard due to drops, version two of this model will include a printed circuit board (PCB). This decreases the probability of malfunction and damage and offers a compact electrical design. Although the manufacturing cost of a PCB depends on the size, layer, components, and assembly, the average cost ranges from USD 10 to USD 50. It decreases the weight of the device, increasing mobility and user satisfaction.

By implementing a PCB in the circuit, the team could reduce the price by more than USD 20, since resistors, diodes and cables will be included in the PCB price.

The price with which the device will be offered in the market can be calculated as following:

$$\text{Unit cost} = \text{variable cost} + \frac{\text{fixed cost}}{\text{sold units}} = 30 + \frac{185.46}{15} = 42.36 \quad (1)$$

$$\text{Overprice} = \frac{\text{unit cost}}{(1 - \text{desired sales performance})} = \frac{42.36}{1 - 0.25} = 56.48 \quad (2)$$

Equation (1) shows the calculations to find the unit cost of the device, which is used in (2) to get the overprice. This overprice is added to the fixed cost of USD 185.46. The market price result is USD 241.94 [6].

The variables used in the equations are hypothetical yet realistic based on work hours and market demand.

#### VI. DESIGN CONCEPT

##### A. Electrical

The BrailleBud consists of 18, 5-volts solenoids, TIP120 transistors, 1N4001 diodes, and 220-ohm resistors; three (3) PIR sensors (passive infrared), large breadboards; one Mega Arduino, slide switch, and Micro SD Card Module. Initially, a

subcircuit with the basic components to control the solenoid was tested (Fig. 1). The final circuit is powered by 4 AA batteries, that sum a total voltage source of 6 volts with an embedded switch.

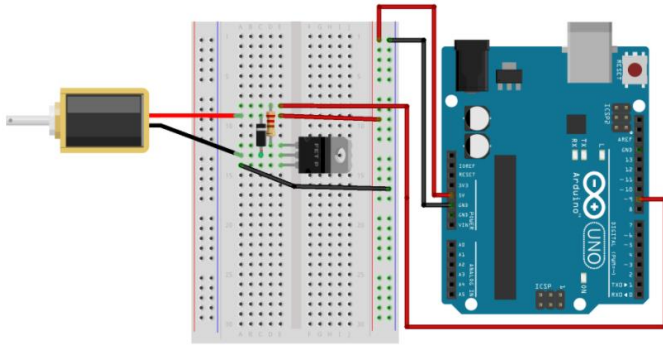


Figure 1: First subcircuit to control the solenoids

The electrical design is divided into three subcircuits. The first subcircuit (Fig. 2, 1), developed in phase one of the project, controls the solenoids. A 2.2kΩ resistor, TIP120 transistor, and 1N4001 diode are needed for each solenoid present to control the current and prevent it to flow backwards when a pulse is sent from the Mega Arduino to the solenoid [7]. The resistor is connected in series to the base and the digital pin in the Arduino. The diode is in series with the collector, voltage, and the positive and negative ends of the solenoid, and the emitter is connected to the ground of the breadboard. The set-up is repeated six times for the braille dots.

The second subcircuit (Fig. 2, 2) consists of the input components - the slide switch and the PIR sensor. The slide switch is connected to the second breadboard (from left to right) to control the executed program. The switch has three terminals, one connected to the ground, another to power, and the common terminal to a digital pin of the Arduino. The same conceptual connection is valid for the sensor. These input components communicate to the microcontroller the user's choice of program and detect the motion of the child's hand above the braille cell to indicate if the program runs continuously or a letter must be looped.

The third subcircuit (Fig. 2, 3) is focused on the output audio of the device. This consisted of a DF MP3 player with an SD reader and a speaker. The sound files were previously uploaded to the 4-gigabyte secure digital – SD - card to use in the code. These files are in wav format and have the following characteristics for the Arduino to read correctly: 8-bit resolution, a sampling rate of 16 000 Hz with a mono audio channel, and an unsigned 8-bit pulse-code modulation (PCM) format.

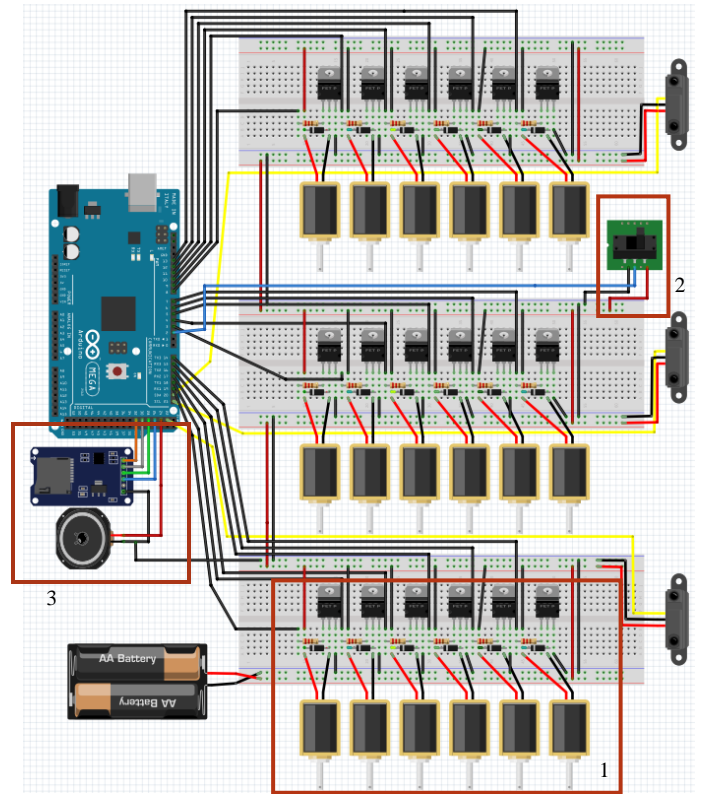


Figure 2: Electrical design of BrailleBud

The three breadboards implemented are connected in series from ground to ground and power to power. Each breadboard has six solenoids, representing a Braille cell and one infrared sensor. The sensor has a sensitivity range of 6 meters, which is limited by the sensor's location on the circuit.

The resistors, diodes, and transistors' values prevent the circuit to overheat and short. Equation (3), derived from Ohm's Law, is applied to ensure the safety of the circuit:

$$I = \frac{V}{R} \quad (3)$$

Where V corresponds to the voltage source of the circuit, and R to the resistance used. The result is compared to 40 mA, which is the maximum current draw in DC current per input and output - I/O - pin of the Mega Arduino. As the voltage source is 5 volts and the resistor has a value of 220 Ω, the total current going through each pin is equivalent to 22.7 mA which is in a safe range.

### B. Simulation

The circuit was built on Tinkercad, a free online resource to build circuits and test Arduino code. During the prototyping phase, the lead programmer tested out a simpler solenoid connection by directly connecting the cathode of each diode to the ground of the breadboard. As the simulation did not work, the programmer decided to keep the solenoid's original circuit design. The first problem encountered was that the software did not have solenoids available. As such a motor was added

to replace the actuator (Fig. 3, 1), as the conceptual application was the same. From there, the diodes, transistors and resistors for each solenoid was put in place, powered by an Arduino UNO. A slide switch (Fig. 3, 2) and PIR sensor (in Fig. 3, 3) was added to write and test the code for the solenoids, sensors, and slide switch to work together. Once this circuit was successful, it was built in real life.

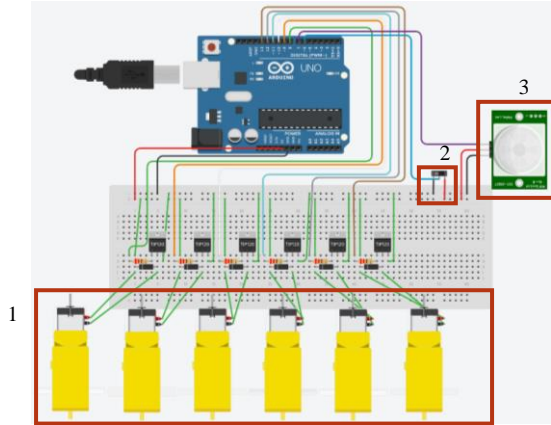


Figure 3: Tinkercad circuit design

### C. Voice

Based on children's television programs and studies of the effects of virtual personal assistants' voices on users, we decided to use a childlike female voice to interact with the device with the child.

The audios are recorded for 29 cases, 26 of them belonging to the alphabet letters (A-Z) and two possibilities for right and wrong manipulation of the device. Additionally, there is one audio to welcome the child at the beginning.

Some examples of the phrases played by the device are: "Welcome! Let us start learning. Please place your hand on the block", "This is A, B, C...", "Good job," "Try placing your hand again."

The isolated circuit for the audio is displayed in Fig. 4. It is not necessary to implement resistors or other components if it is connected in the indicated positions. The circuit for the audio could include sound amplifiers, such as CI TDA7073, that can be used to control the volume and distortion of the tones.

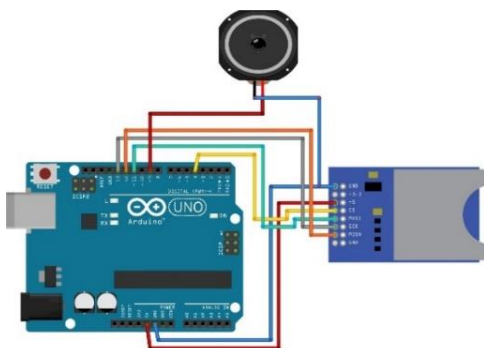


Figure 4: Audio subcircuit

### D. Coding

The code for the BrailleBud was developed on the Arduino software. The setup includes 19 output and four input pins. The 18 solenoids and DF player are set as output pins while the slide switch and the three PIR sensors are the inputs. The code has six void functions to build the leveled programs and set the output pin patterns with the corresponding wav files. Once the device is turned on, the loop, by default, is active. Here, the microcontroller waits for the state of the slide switch. Once it is specified, a level one or level two program is activated.

The slide switch does not implement parameters, as its purpose is to recognize the state of the switch. The input of the switch is determined by a 0 or a 1, meaning LOW or HIGH state, respectively. If it has a low state, the program for level one, learning the alphabet, is triggered. Else level two, word association program, is triggered (Fig. 5). The digitalRead function is added to ensure that the reference system is valid (i.e., to the left, level one is triggered as the switch causes the pin to be connected to ground while to the right, level two is triggered the pin is connected to the voltage source).

```
void loop() {
  int swi = 0;
  swi = digitalRead(switchS);
  if(swi == LOW) {
    programO();
  }
  else { //level 2, letter recognition
    programT();
  }
}
```

Figure 5: Loop to determine program levels

Level one function operates with six solenoids and has a nested switch case in a for loop. The for loop allows the program to run from one case to another automatically if conditions are met (Fig. 8). The aPO function (Fig. 6), also known as activate pin for level one, is implemented to reduce the lines of code to activate the solenoid pins accordingly to the letter. It has 6 integer parameters that correspond to the digitalWrite of solenoidPins 1 through 6 that communicates to the microcontroller which pin is given a HIGH or LOW voltage. Similarly, function aPT (Fig. 7), active pin for level two, is developed where the microcontroller operates with 12 to 18 solenoids at a time.

```
void aPO(int pin1, int pin2, int pin3, int pin4, int pin5, int pin6) {
  digitalWrite(solenoidPin, pin1);
  digitalWrite(solenoidPin2, pin2);
  digitalWrite(solenoidPin3, pin3);
  digitalWrite(solenoidPin4, pin4);
  digitalWrite(solenoidPin5, pin5);
  digitalWrite(solenoidPin6, pin6);
}
```

Figure 6: aPO function for level 1

```

void aPT(int pin1, int pin2, int pin3, int pin4, int pin5, int pin6,
int pin7, int pin8, int pin9, int pin10, int pin11, int pin12, int pin13,
int pin 14, int pin15, int pin16, int pin17, int pin18){
digitalWrite(solenoidPin, pin1);
digitalWrite(solenoidPin2, pin2);
digitalWrite(solenoidPin3, pin3);
digitalWrite(solenoidPin4, pin4);
digitalWrite(solenoidPin5, pin5);
digitalWrite(solenoidPin6, pin6);
digitalWrite(solenoidPin7, pin7);
digitalWrite(solenoidPin8, pin8);
digitalWrite(solenoidPin9, pin9);
digitalWrite(solenoidPin10, pin10);
digitalWrite(solenoidPin11, pin11);
digitalWrite(solenoidPin12, pin12);
digitalWrite(solenoidPin13, pin13);
digitalWrite(solenoidPin14, pin14);
digitalWrite(solenoidPin15, pin15);
digitalWrite(solenoidPin16, pin16);
digitalWrite(solenoidPin17, pin17);
digitalWrite(solenoidPin18, pin18);
}

```

Figure 7: aPT function for level 2

```

void programO (){
for (int i = 0; i < 27; i++) {
switch (i){
case 1: //letter A
aPO(1,0,0,0,0,0);
delay(1000);
break;
case 2://letter B
aPO(1,1,0,0,0,0);
delay(1000);
break;
case 3: //letter C
aPO(1,0,0,1,0,0);
delay(1000);
break;
case 4: //letter D
aPO(1,0,0,1,1,0);
delay(1000);
break;
case 5: //letter E
aPO(1,0,0,0,1,0);
delay(1000);
break;
case 6: //letter F
aPO(1,1,0,1,0,0);
delay(1000);
break;
}}
}

```

Figure 8: Cases for letters displayed by solenoids in level 1

The PIR sensor is continuously activated within the leveled programs and coordinates with the wav audio files. Once the child covers the sensor for the first time, the counter will start running, adding +1 every time the child covers the sensor again. The code will jump to the following letter every time the sensor detects motion in a specified range (Fig. 10).

For this purpose, the team used cases, which will be explained in the following page.

Fig. 9 shows the code to play one audio. An external library was added to the program. This audio library uses OpenAI. It was made to have a simple tool for playing sound/music in a game or another multimedia application.

Then, the program will display messages to indicate whether the code is working properly or not until it plays the audio.

```

#include <sd_10.h>
#include <sd_11.h>
#include <sd_12.h>
#include <SimpleSDAudio.h>
#include <SimpleSDAudioDefs.h>

void setup() {
Serial.begin(9600);
Serial.println("Initializing SD card...");
if (!SdPlay.init(SSDA_MODE_FULLRATE | SSDA_MODE_MONO | SSDA_MODE_AUTOWORKER)) {
Serial.println("initialization failed. Things to check:");
Serial.println("* is a card is inserted?");
Serial.println("* Is your wiring correct?");
Serial.println("* maybe you need to change the chipSelect?");
while(1);
} else {
Serial.println("Wiring is correct and a card is present.");
}
if(!SdPlay.setFile("A.wav")) {
Serial.println("file not found on card! Error code: ");
Serial.println(SdPlay.getLastErrorMessage());
while(1);
} else {
Serial.println("file found.");
}
}

void loop() {
Serial.print("Playing... ");
SdPlay.play();
while(!SdPlay.isStopped()) {
;
}
Serial.println("done.");
delay(3000);
}

```

Figure 9: Audio setup code

This audio code must be combined with the one controlling the solenoids and sensors. For this purpose, the team used cases representing each letter of the alphabet, for example case 1 corresponds to letter A.

```

void loop(){
sensor = digitalRead(pir);
if (sensor == 1){
//digitalWrite(solenoids[0],HIGH);
delay(100);
count ++;
Serial.println(count);
}
solenoid(count);
}

```

Figure 10: Sensor setup code

The counter starts with the first interaction with the sensor. The functions in Fig. 11 shows the code for this purpose.

```

void setup(){
Serial.begin(9600);
for(int i=0; i< n_solenoids; i++){
pinMode(solenoids[i], OUTPUT);
}
pinMode(pir, INPUT);
}

void solenoid(int q){
switch(q){
case 1:{
int states[] = {1,0,0,0,0,0}; //A
aL(solenoids, states, n_solenoids);
Audio.play("this is letter A"); break;
}
break;
}
}

```

Figure 11: Cases for letters displayed by solenoids

For each case, two functions will be executed: one controlling the solenoids and the second controlling the audio.

With every correct child's interaction with the device, the code jumps to the following case. Otherwise, the code will run for the same case.

### E. Mechanical

Braille Bud consists of 4 components. The Braille Box compartment (Fig.15), the switches/speaker (Fig. 16 – 17), the Arduino Mega (Fig. 13, 5), and finally, the tablet casing (Fig. 18). The mechanical designs focus was on housing the breadboards, solenoids, speakers, and the microcontroller in one single tablet. Each Braille Box houses the six solenoids, a large breadboard and a PIR sensor used to detect motion through a 10mm diameter hole. This physically decreases the motion range of the PIR and is only activated once the child hovers their hand above the cells. Where the solenoids are placed creates a Braille cell for preschoolers to learn Braille. The power source for this device are four 6-volt AA batteries that were chosen to ensure the longevity of the tool, and ease to remove and replace the battery.

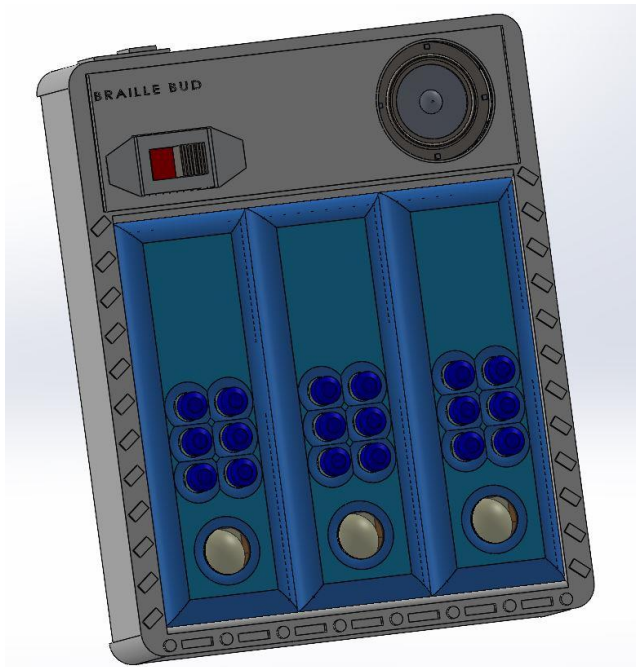


Figure 12: BrailleBud overview

### F. Working Mechanism

This product's design intent is to be used in a pre-school setting and or at home. The BrailleBud's rectangular design was specifically chosen for two main reasons: to distribute any sudden impact caused by most accidents and familiarity for parents and children to comprehend. Its size is slightly above the average of an at-home tablet used by most families but is still much smaller than other products that perform similar functions.

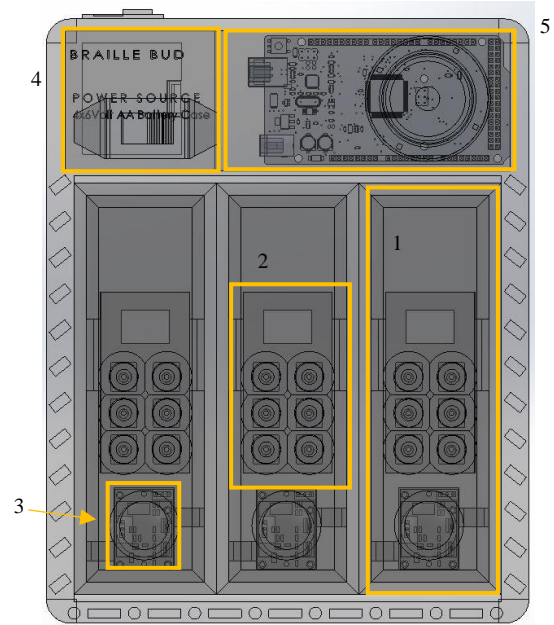


Figure 13: BrailleBud Skeleton (Top View / Spacing)

When creating the design of this device, space was constantly an issue. To make the spacings correct, it was imperative to work from the inside and expand outward. The largest part of BrailleBud, that will always be used is the Braille Box (Fig. 13, 1). The length and width of this compartment is 175mm and 59mm, respectively. This spacing ensures a slight “wobble room” for the wiring into the breadboard. The container that houses the solenoids (Fig. 13, 2) is placed 5.15mm from the bottom of the box, leaving enough room for the breadboard to fit. Finally, a square hole is placed to the rear of the container (Fig. 14) to connect the solenoids with the breadboard.

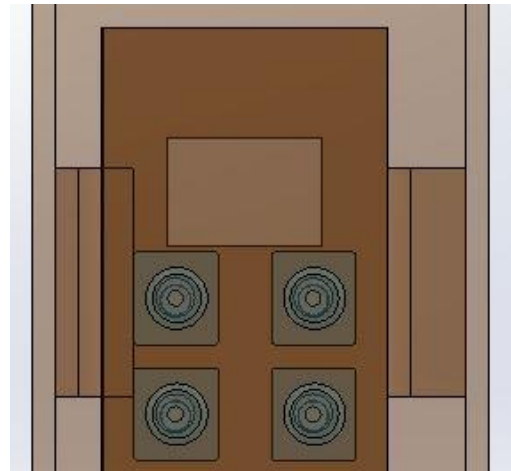


Figure 14: Wire Hole in Braille Box (Top View)

An additional box was created at the towards the front Braille Box to hold the PIR sensor, as viewed in Fig. 13, 3.

Simultaneously, the battery source (Fig. 13, 4), the Arduino Mega (Fig. 13, 5) are placed at the top of the device. Finally, with the speaker and slide switches. The upper interior has 68mm by 184mm of space to, again, leave enough “wobble room” for the wires going into the Arduino’s pins and a proper flow of heat potentially generated from the electrical parts.

The Braille Box has six solenoids, where each solenoid has a 3D printed cover on the actuator with a radius of 4mm. This allows the child to have a better feel for where the cells are when interacting with the Braille cell. The solenoids are connected via magnetic strips and placed in two columns and three rows. These solenoids are apart measuring from the center by 17.90mm and 15mm long. The BrailleBud has three of these Braille Boxes. Setting these solenoids in this formation creates unlimited possibilities for displaying outputs to teach the user, from a single letter to complicated words for a preschooler.

Looking at Fig. 13; when creating the design of this product, the spacing constantly come up as an issue. Starting inward and then moving outward, the central part of BrailleBud that will be interacting all the time is the Braille Box (Fig. 15). The length and width of the compartment are 175mm and 59mm. This spacing amount ensures a slight “wobble room” for the wire management when placing the breadboard inside the Braille Box. This container houses the solenoids (Fig. 13, 1) and placed 5.15mm from the bottom of the box. This leaves enough room for the breadboard to fit. A square hole was cut to the rear of the container holding the solenoids (Fig. 14) to connect them to the breadboard.

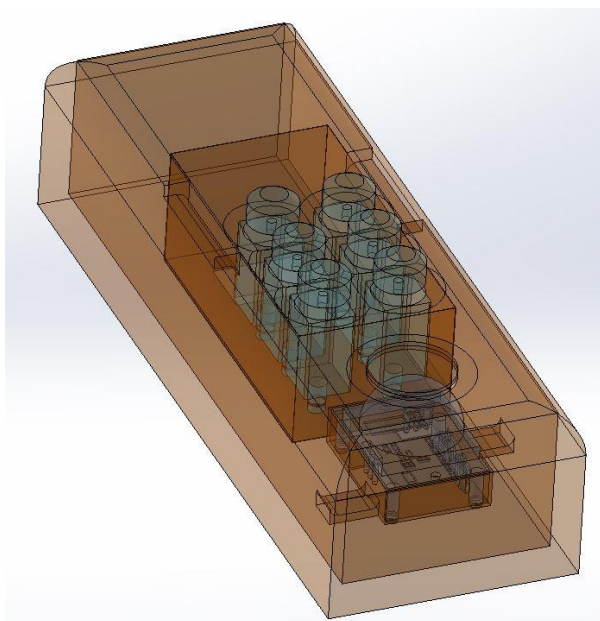


Figure 15: Braille Box (Isometric View)

An additional box was created to hold the PIR is housed (Fig. 13, 3). As soon as Braille Box found the measurements, making spacing for the battery source (Fig. 13,

4), an Arduino (Fig. 13, 5) was placed on top of the BrailleBud, with the speaker and slide switch. The upper interior has 68mm by 184mm of space. Like how the Braille Box components were placed, this leaves enough “wobble room” for wire management and a proper flow of heat generated.

Two switches are displayed in Fig. 16. The level slide switch (Fig. 16,1) and power switch (Fig. 16, 2). The level switch is placed directly over the power supply but close enough to connect to the Arduino. It allows the tablet switch between level 1 and level 2 difficulty. The power switch is placed directly in front of the power source to eliminate the need for any additional wiring and to keep the switch in a safe location, in case of any potential damages. The speaker’s system (Fig. 17) is placed directly above the Arduino, creating a more compact design, and eliminating any unnecessary wiring across the tablet.

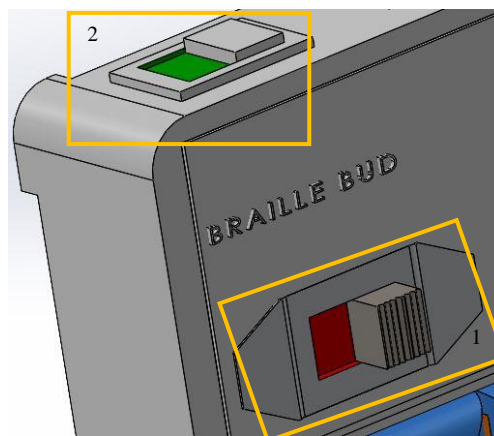


Figure 16: Level Switch (1), Power (2)

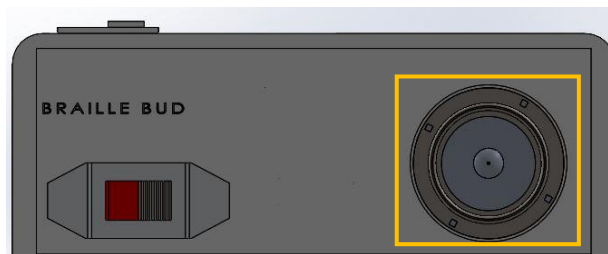


Figure 17: Speaker Location

Using the same logic of working internally to then expand outward, the most important area that was necessary to have the correct measurements was the middle section to hold the 3 Braille Box cases together with no “wobble room”. Measuring the cases side by side, the space needed had to be 170mm in length and 177mm in width (Fig. 18, 1). This provided the exact measurements for the Braille Box with all its components secured inside. Knowing the necessary measurements of the Braille Box’s side by side, it gave more room to play around with the size and space of the device. As the main design intent is to keep BrailleBud compact, light, and easy to travel around, the total size of the device is 260mm long, 200mm wide and

3.82mm thick (Fig. 18, 2). The main design intent was to keep this tablet within a reasonable size of a tablet used by most families.

This size was considered the proper amount of space required for connecting all the necessary wires from the breadboard, Arduino, speaker, and power source. This also left a relative amount of space to ensure proper wire management for the Arduino, speaker, switch, sensors, and breadboard connection. Coincidentally, the size that BrailleBud is planned to take up is a massive improvement from other competitors that are providing the same product. Additionally, the weight of the tablet considering everything put together is 1.067kg. Although this weight is above the average for any modern tablet, it is still below the total weight of other competitors with their products being over 2kg to 3kg.

*E. Pattern*

One aspect that went into creating the Braille Bud was how to make it stand out and encouraging for children to use their sense of touch. On the tablet case, surrounding the edges, are different extruded patterns. These are strategically placed close to where the child will be using this tablet the most, around where the box (Fig. 18, 3). This sense of interaction comes from making the patterns the first distinction of the device that children will interact with. This will make the child curious enough to wonder what the thing in front of them is, and the second is to create memory memorization. With these patterns' lines up, the child will remember the tablet and become excited to utilize it with their parent or teacher.

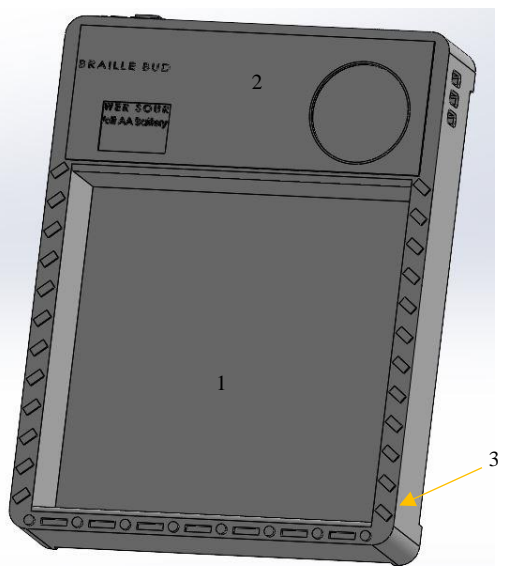


Figure 18: Tablet Case (Isometric View)

A ventilation system was implemented as a safety precaution in areas that had the most thermal activity. The two vent spots selected was on the right face perpendicular to the Arduinos location (Fig. 19, 1). The second location was to the left of the power switch (Fig. 20, 2). These two locations were

selected because of the amount of potential heat generated proportional to the entire tablet, specifically, with the Arduino and power source being the primary electrical components that require a proper amount of cooling. Even with the thermal conductivity of PLA, it was a precaution that was necessary for this tablet's longevity. The battery cover is displayed in Fig. 20, 3.

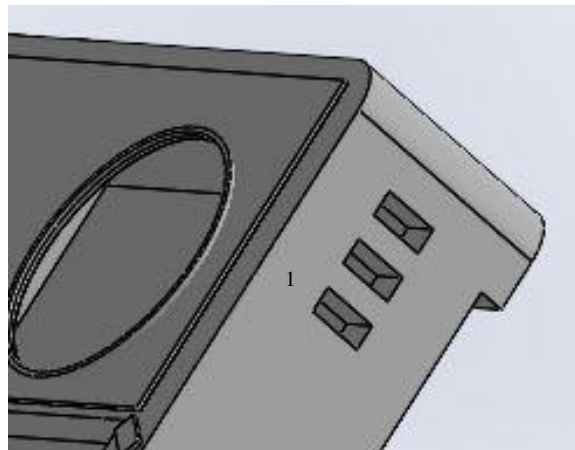


Figure 19: Right Side Vent View

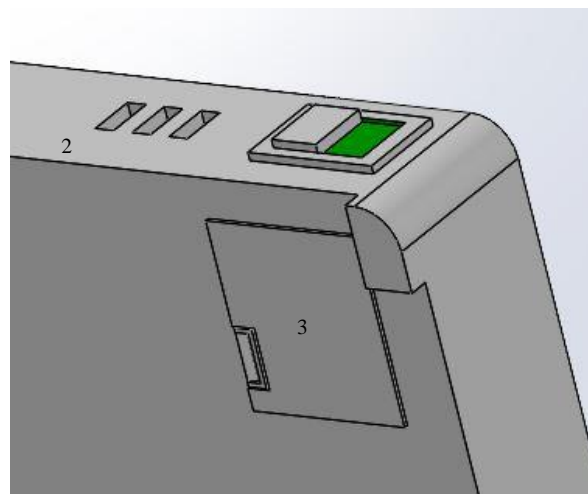


Figure 20: Rear Vent and Battery Case

VII. PRODUCT EVALUATION

Once the prototype of the BrailleBud is assembled, the next step is to evaluate how children interact and ultimately learn from the device. The steps to determine the device's efficiency are as follows: inspection, demonstration, test, and analysis. For this, the team plans to collaborate with the Helen Keller Services for the Blind Children's Program to collect feedback from a teacher and a small group of children. The team will instruct how the device is used and have children trace over a few letters of the Braille alphabet before slowly transitioning into level 2. The team will assess what attracts children the most to the device as well as how long they are interested in playing



with BrailleBud and evaluate how they develop letter recognition skills going from level 1 to level 2. All feedback will be noted for future improvement.

### VIII. CONCLUSION

Considering the convenience of price, the relatively simple construction of the device, and the interaction in the learning process, BrailleBud can make a big difference when choosing a method to learn to read braille from an early age.

Designing and testing the circuit prototype allowed the team to understand the combination of the code with the mechanism of the device. Besides, this allowed us to find future features that can be added to the project.

One of the future improvements the team can consider for future versions is making the device able to connect to smart devices such as smartphones or TVs, so parents can supervise their child's learning remotely.

Other improvements include solutions to ventilate the interior of the device in case of overheating, subsequently decreasing risk factors, such as a fan and slots under the design; implement softer and more durable materials for the solenoid's piston instead of plastic; add more blocks in future versions to create longer and more complex words for older children and even teenagers.

### ACKNOWLEDGMENT

The authors would like to thank the President of Vaughn College of Aeronautics and Technology, Dr. Sharon DeVivo,

along with the Engineering and Technology Department Chair, Dr. Hossein Rahemi and faculty professors Dr. Shouling He and Dr. Mohammed Benalla for their continued support throughout the whole project. Also, a special thank you to Professor Donald Jimmo for his continuous support in putting this project together.

### REFERENCES

- [1] "Description and History of Braille." *Description and History of Braille*. | *American Council of the Blind*. [Online]. Available: <https://www.acb.org/history-of-braille>. [Accessed: March 1, 2021].
- [2] "Statistics About Children and Youth with Vision Loss." *American Foundation for the Blind*. [Online]. Available: <https://www.afb.org/research-and-initiatives/statistics/statistics-blind-children>. [Accessed: February 25, 2021].
- [3] "Blind Adults in America: Their Lives and Challenges." *A Report by the National Center for Policy Research for Women and Families*. Washington, DC. February 2004. [Online]. Available: <http://center4research.org/wp-content/uploads/2010/05/blind02041.pdf>. [Accessed: February 28, 2021].
- [4] "Visual impairment statistics including population, causes, education and employment." Perkins School for the Blind, 2017. [Accessed: June 18, 2021].
- [5] "Focus and Repetition in Learning." *apl next ED*. 23, March 2020. [Online]. Available: <https://aplnexted.com/focus-and-repetition-in-learning>. [Accessed: February 28, 2021].
- [6] Kotler, P., Armstrong, G., & Cunningham, M. H. (2017). *Principles of Marketing*. Toronto: Pearson Prentice Hall.
- [7] Rabbani, S., & D'Arrigo, J. D., & He, S., & Elzawawy, A., & Rahemi, H. (2016, June), *MAKER: A Braille Clock* Paper presented at 2016 ASEE Annual Conference & Exposition, New Orleans, Louisiana. 10.18260/p.2559