بِسۡمِ ٱللَّهِ ٱلرَّحۡمَٰنِ ٱلرَّحِيمِ

An-Najah National University

Faculty of Engineering and Information Technology

Computer Engineering Department

Hardware Graduation Project

BRAILLE PRINTER WITH WIRELESS PDF TRANSFER

Prepared by:

-Faiq Bakri ***-Sami Haji***

Supervised by:

Dr. Saed Tarapiah

Submitted on:

31 January 2025

Presented in partial fulfillment of the requirements for Bachelor degree in Computer Engineering.

Dedication

Dedication our loving parents, family, friends and for everyone who believed and loved us.

Acknowledgment

We extend our deepest gratitude and appreciation to the individuals who have played a significant role in our graduation project. Their guidance, support, and unwavering belief in our abilities have been invaluable throughout this journey.

We would like to **thank our supervisor Dr. Saed Tarapiah** a lot for his helpful, kind, patience and taking care of us, and for making everything simple. He was always inspiring and encouraging us to move.

We would also like to **thank all the teachers and teacher's assistant in the Department of Computer Engineering**, and we feel proud to be students in it, as this helps us to improve our educational level as well as improve our skills.

Disclaimer

This report was written by students at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

Table of content

List of Figures

List of tables

Abstract

Braille printers play a vital role in providing accessibility to visually impaired individuals, but their high cost and complexity often make them inaccessible. This project focuses on designing a low-cost, portable Braille printer using readily available components such as the ESP32 microcontroller, stepper motors, and a solenoid. The system consists of an (ESP32) bi-directional server that receives PDF files and process them, then sends the processed text via Serial transmission to the Arduino that translates the received and processed text into Braille and embosses it onto A4 paper with precision, adhering to international Braille standards.

Key features include wireless file transfer via Wi-Fi, an intuitive web interface, automated word wrapping to maintain text readability, responsive LCD, keypad for user-interaction within the printing process, and a simple safety-system. The printer achieves a dot spacing accuracy of ±0.2mm. Its midweight design, powered by USB, ensures portability and ease of use.

It bridges the accessibility gap, enabling individuals, schools, and organizations to produce Braille materials independently. This open-source solution has the potential to make significant contributions to education, inclusivity, and accessibility for the visually impaired community.

Chapter 1: Introduction

- **Statement of the problem**

Visually impaired individuals face significant challenges when it comes to printing text in Braille. One of the most critical obstacles is the high cost of Braille printers, which can range from $4,000 to $7,000. This price barrier places such devices beyond the financial reach of many who could benefit from their use.

Furthermore, the availability of Braille printers in the market is extremely limited compared to standard paper printers. This scarcity is caused by the specialized nature of Braille embossing technology, the smaller target market for such devices, and the technical complexities involved in their production.

As a result, visually impaired individuals often encounter difficulties accessing essential printed materials in a readable format. This includes educational content, professional documents, and even everyday communication. Such barriers hinder their progress in education and employment and limit their ability to engage fully in personal and social activities.

Addressing these challenges requires innovative solutions that focus on reducing the cost and increasing the availability of Braille printers. By making these devices more affordable and accessible, this project aims to empower visually impaired individuals, providing them with greater independence and inclusion in society.

▪ Objectives

The objective of this project is to design and implement an affordable and user-friendly Braille printer that overcomes the limitations faced by visually impaired individuals. The specific goals of the project are:

1. **Develop a User-Friendly System**:
   - Create an intuitive Braille printer interface, supported by a Liquid Crystal Display for status update, and a keypad for user-interaction
   - Design a simple web-interface to allow users to upload PDF files and send it to the printer seamlessly.

2. **Automate the Printing Process**:
   - Enable a half-automated workflow where users can load paper, send text via the web interface, and retrieve printed Braille paper without as less of intervention as possible.

3. **Enhance Affordability and Accessibility**:
   - Develop a printer that is significantly cheaper than existing models, with a cost range of 800$ - 1200$, making it more accessible to a broader audience.

4. **Ensure Precision and Quality**:
   - Guarantee that the printer produces Braille text that adheres to international standards, with accurate dot spacing and alignment.

Standards, Codes, and Constraints

This project adheres to various industry standards to ensure reliability and compliance:

1. **Braille Standards:** Follows international Braille guidelines for character spacing and dot formation.

2. **Wi-Fi Communication Standards:** Uses IEEE 802.11 standards for wireless file transfers between devices.

3. **Safety and Electrical Standards:** All electrical components comply with standard safety regulations to prevent overvoltage and overheating.

4. **Design Constraints:** The design prioritizes affordability, accuracy, and ease of use, ensuring the printer remains cost-effective without sacrificing functionality.

- ## Scope of the work

This project involves the following components and functionalities to achieve the objectives:

1. **Web Interface Development**:
   - Design Python-based web interface that connects to the printer via the ESP32 Wi-Fi module.
   - Provide a user-friendly interface for uploading PDF to the printer.

2. **Wi-Fi Connectivity**:
   - Establish reliable communication between the web interface and the printer using Wi-Fi to enable remote control and file transfer.

3. **Text-to-Braille Conversion Algorithm**:
   - Develop an efficient algorithm to convert text into Braille patterns, ensuring compliance with Braille language standards.

4. **Paper Feeder Mechanism**:
   - Create a user-friendly mechanism of feeding paper to the printer via simple instructions given by the LCD.

5. **Printing Mechanism**:
   - Implement an X-based coordinate axis-system with a Y-axis for paper enrollment and a single needle solenoid pinned onto the X-axis which operates as a Z-axis for embossing Braille dots onto paper accurately.

6. **System Integration and Testing**:
   - Combine all hardware and software components into a cohesive system.
   - Conduct thorough testing to ensure reliability and accuracy.

- ## Importance of the work

The Braille printer holds immense significance for visually impaired individuals due to its ability to address critical barriers in accessibility.

1. **Precision and Quality**:
    - The printer produces high-quality results that comply with Braille standards, ensuring accurate dot spacing and character alignment.

2. **Time and Effort Savings**:
    - Compared to manual Braille embossing, the automated process significantly reduces the time and effort required for printing Braille text.

3. **Affordability**:
    - Existing printers cost upwards of $4,000, while this project offers a low-cost solution priced between $800 and $1200.

4. **Availability**:
    - The printer can be rapidly assembled and made available on demand, addressing the current scarcity of Braille printing devices.

5. **Customer Satisfaction**:
    - By offering a cost-effective and reliable alternative, this initiative aims to improve customer satisfaction and accessibility for visually impaired users.

▪ Organization of the report

This report is structured to provide a comprehensive overview of the project and its outcomes:

1. **Introduction**: Outlines the problem, objectives, scope, and significance of the project.

2. **Theoretical Background and Previous Work**: Reviews existing research and similar projects to establish the project's context and identify knowledge gaps.

3. **Methodology**: Details the design and implementation process, including the hardware, software, and workflow of the Braille printer.

4. **Results and Analysis**: Presents the project's outcomes, including the performance metrics and testing results.

5. **Discussion**: Analyzes the features, benefits, and limitations of the system, along with recommendations for improvement.

6. **Conclusions and Recommendations**: Summarizes the findings and highlights the impact and potential future directions for the project.

7. **References**: Lists all cited sources to provide credibility and traceability.

8. **Appendices**: Includes supplementary materials, such as schematics, source code, and testing logs.

Chapter 2: Theoretical Background and Previous Work

2.1 Overview of Braille Printing Technology

Braille printing technology has significantly evolved over the years to help visually impaired individuals access written materials. Braille printers, also known as embossers, play a vital role in translating text into tactile dots that can be read through touch.

In the early days, Braille text was created manually using a Braille stylus to emboss dots onto paper. This method was simple but extremely time-consuming and required considerable effort for even small amounts of text [1].

The invention of mechanical embossers was a major milestone in Braille printing. These devices used punch-and-die mechanisms to automate the embossing process, making it faster and more efficient than manual methods. However, they were bulky, expensive, and limited in print speed, restricting their accessibility to a broader audience [1].

With the rise of digital technology, electronic Braille printers transformed the field. These printers allowed computer-generated text to be converted into Braille and embossed onto paper. This innovation made it possible to produce books, educational materials, and other documents in Braille more easily. However, the complexity of these devices and their high cost remain significant challenges, making them inaccessible to many individuals and organizations [2].

Our project aims to address these limitations by designing a Braille printer that combines affordability with simplicity. By reducing the cost and simplifying the design, this printer ensures that more visually impaired individuals can access written materials independently.

2.2 Previous Work

Over the years, various efforts have been made to improve Braille printing technology. These can be categorized into commercial Braille printers and DIY or open-source solutions.

2.2.1 Commercial Braille Printers:

- **Index Everest-D**: This high-performance Braille printer offers fast and accurate embossing for single sheets of paper. It supports multiple languages and Braille layouts but costs over $5,000, making it inaccessible to most individuals [3].

- **Braillo 300 S2**: This industrial-grade Braille printer is capable of printing large volumes of Braille text with exceptional accuracy. However, its price exceeds $30,000, and it is designed for institutional rather than personal use [4].

- **VersaPoint Duo**: A compact Braille printer suitable for small-scale applications, the VersaPoint Duo supports double-sided printing but has slower print speeds compared to larger models [5].

These commercial solutions set a high standard for quality and reliability but remain unaffordable for most individuals or small organizations.

2.2.2 DIY and Open-Source Solutions:

- **BrailleRAP**: A 3D-printed Braille embosser that uses modified 3D printer components. It is affordable and easy to assemble for users familiar with 3D printing but lacks durability and precision [6].

- **OpenBraille**: A fully open-source Braille printer project that utilizes off-the-shelf components, making it budget-friendly. However, it requires significant technical knowledge for assembly and maintenance [7].

- **Arduino-Based Braille Printer**: A low-cost DIY solution using an Arduino microcontroller and solenoids to emboss Braille dots. While affordable, it has limited features and lacks wireless connectivity [8].

These solutions demonstrate that low-cost Braille printing is possible but often come with trade-offs in durability, precision, or usability.

2.3 Innovations Introduced in This Project

Our design builds on the strengths of previous work while addressing their limitations. The key innovations include:

1. **Affordability**:
    - The printer is designed to cost between $800 and $1200, making it accessible to individuals and small organizations.

2. **Wireless Connectivity**:
    - Unlike most DIY solutions, our printer integrates Wi-Fi functionality, enabling users to upload text from a web application.

3. **Simplified Design**:
    - The system uses an ESP32 microcontroller and off-the-shelf components, ensuring ease of assembly and maintenance.
    - The system eliminates the usage of a Y-axis like the one used in CNC system, and instead approaches moving the solenoid only among the X-axis, while the Y-axis is identified as a roller for rolling the paper forward, printer-like. This compounds the accuracy because the movement is minimized as much possibly made.

4. **Precision and Reliability**:
    - The printer achieves dot spacing accuracy of ±0.2mm, meeting international Braille standards for tactile readability.

By combining these features, this project provides a practical and affordable solution for Braille printing, empowering visually impaired individuals to access written materials independently.

Chapter 3: Methodology

- ## How It Works?

At first you need to turn on the printer by pressing the power button, then the WiFi module start connecting to the WiFi then connect to the server which starts listening and waiting for any upload to happen.

Now we can connect to the WiFi module via any smart device, let it be a smart phone, a tablet, a laptop or even a smart TV, then after connecting to it you will be redirected instantly to the web interface which has the option of upload in a user-friendly front, after clicking on the upload, the files folder on your smart device will be opened and the PDF files in it will be filtered, thus you can't upload any non-PDF files, which support the fault tolerance of the project. After picking the PDF that you want to upload you can press the upload button. After pressing it, the server will process the PDF file content then split it into chunks, taking into consideration the maximum serial buffer size of the Arduino, and start sending it, chunk after the other, with a simple hand-shaking protocol between it and the Arduino microcontroller. Each chunk will be received in the Arduino, and then the Arduino will start processing the processed string which consists of two operations: 1- The word-wrapping algorithm which works by checking for the last space index within the single row of braille letters on a paper and start counting how many rows will be printed in braille. 2- translating the word-wrapped string into Braille language. The processing operation will turn an array of characters to an array of word-wrapped Braille letters which is an array of a class called 'Letter' that has a matrix in it, for representing the Braille letter. After the processing is done, we will have an array of Braille letters ready to print. Then the printing will occur.

The printing process works by going through the Braille letters array, element after the other, in that order: ROW-> SUB-ROW-> LETTER-> COLUMN. It checks the first word-wrapped row, then strap it into 3 sub-rows, going from the first row of each letter in a whole paper row to the last. In each sub-row it goes through the letters in the row, letter by letter; skipping it if empty (space), and travel within the letter's columns if else.

- **The Mechanism of conversion text to braille**

When text received in the Arduino stores it in an array of characters

```
for (int j = x_counter; j <= last_space_index; j++) {
          Serial.println(x[j]);
            // Check for small letter
            if (x[j] >= 'a' && x[j] <= 'z') {
                x[j] = x[j] - 32; // Convert to uppercase
                int index = x[j] - 'A';
                letters[j] = BRAILLE_ALPHABET_MIRRORED[index];
                isEnglish=true;
            }
            // Check for capital letter
            else if (x[j] >= 'A' && x[j] <= 'Z') {
                int index = x[j] - 'A';
                letters[j] = BRAILLE_ALPHABET_MIRRORED[index];
                isEnglish=true;
            }
            // Check for Arabic letters
            else if (x[j] >= 0x0627 && x[j] <= 0x064A) { // Arabic Unicode range
              int index = getArabicBrailleIndex(x[j]);
                if (index != -1) {
                  letters[j] = BRAILLE_ARABIC[index];
                  isArabic=true;
                }
            }
            // Check for number
            else if (x[j] >= '0' && x[j] <= '9') {
                int index = x[j] - '0';
                letters[j] = BRAILLE_NUMBERS[index];
            }
            else if (x[j] == ' ') {
                // int index = x[j] - '0';
                letters[j] = BRAILLE_NUMBERS[10];
            }
            else if (x[j] == '\n'){
              letters[j]= BRAILLE_NUMBERS[10];
            }
            else if(x[j] == '*'){
              letters[j]= BRAILLE_NUMBERS[10];
            }
            // Other characters can be handled here if needed
            else if(x[j] == 39){
              letters[j] = BRAILLE_PUNCTUATION[4];
            }
```

```
            else{
              int length = sizeof(x) / sizeof(x[0]);
              uint32_t codePoint = decodeUtf8(x, j, length);  // Decode the UTF-8
character

              // Check if the decoded codePoint is an Arabic character
              if (codePoint >= 0x0627 && codePoint <= 0x064A) {  // Arabic
Unicode range
              int index = getArabicBrailleIndex(codePoint);  // Get Braille index
for Arabic
              if (index != -1) {
              letters[j] = BRAILLE_ARABIC[index];
              isArabic = true;
        }
    }
            }
        }

        Serial.println("Finished Translating");

        // Place an asterisk at the end of the row
          letters[last_space_index+1] = BRAILLE_SIGNS[2]; // '*'


        Serial.println("Chunk processed");
        if (current_cursor_index >= x_size) {
    break;  // Exit the loop if we've reached the end of the string
    }
         i = current_cursor_index;
        counter = 0; // Reset counter for the next chunk
        // Place an asterisk at the end of the row
        // i = current_cursor_index;
        // letters[i] = BRAILLE_SIGNS[2]; // '*'
    } // i loop
      Serial.println("Translation complete");
```

- # Components

Table 1: Hardware components

Hardware Component	Image	Number of times Used	Purpose
Arduino Mega		**1**	Main controller board for overall system functionality.
ESP32		**1**	Used for WiFi communication between the Braille Printer and the Web application application.
Ultrasonic sensor		**2**	Used to detect if there is an obstacle or not.
Stepper Motors Nema 23		**2**	Control to the XY coordinates.
TB6600 Stepper Motor Driver Carrier		**2**	Drives the stepper motor responsible for precise movement in the system

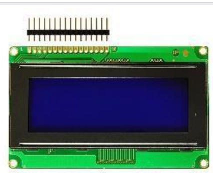
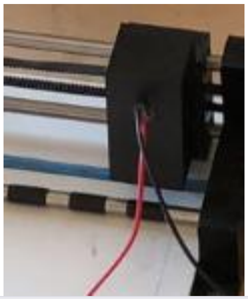
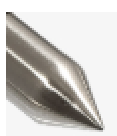
Hardware Component	Image	Number of times Used	Purpose
5V 2-Channels Relay Module Active High/Low		**1**	To controlling to the solenoid.
DC 12V 2.1Kg 10mm Stroke Push Pull Type Solenoid		**1**	To knocking on the paper.
On/Off Switch		**1**	Serves as an on/off switch for the entire system, allowing easy control of the system's power.
PC Power Supply		**1**	Supplies the required power to the system, ensuring proper functioning of the components.
LCD 16*2		**1**	To show the status and update it
4x4 Keypad		**1**	User-interaction with the system

Table 2: Mechanical components

Mechanical Component	Image	Number of times Used	Purpose
Stainless Steel Carbide Metal Rod		**1**	Y coordinate
Stainless Steel Carbide Metal Rod		**2**	X coordinate
Slide Block Carriage Unit (contains the solenoid)		**1**	To move on X-axis
Rigid Coupling		**1**	To put it in the head of the Y-axis stepper motor and connect it to the Metal rod

GT2 16 Teeth Idler Pulley Bore 3mm		**1**	To put it in the corresponding side of the X-axis' stepper motor

Mechanical Component	**Image**	**Number of times Used**	**Purpose**
1meter Rubber GT2-6mm Open Timing Belt ,6mm Width		**1**	To move the X-axis
Braille pen		**1**	To put it on the head of solenoid

▪ Design

▪ Body of machine

1. The goal was to create an affordable, functional Braille printer. Inspired by traditional inkjet and dot matrix printers, but adapted for embossing Braille.
   i. X-Axis: Moves the embossing head (needle + solenoid) left and right.
   ii. Y-Axis: A roller mechanism moves the paper forward after each sub-row.
   iii. Z-Axis: Controls the solenoid and needle for embossing dots.

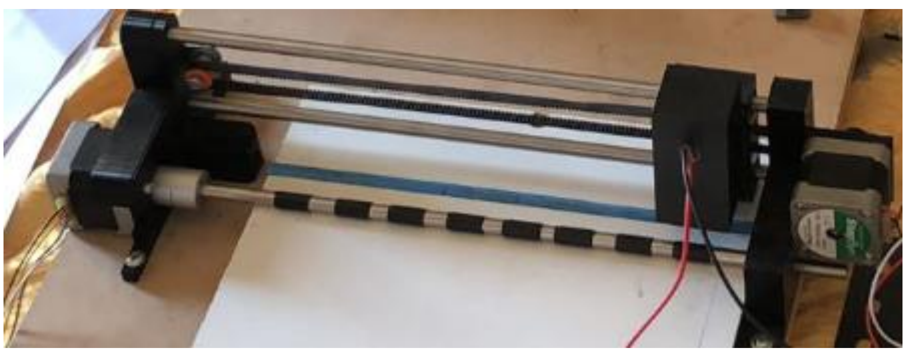


Figure 1: Inner body design of the printer

2. Constructing the base for the Braille printer, pinning the Power supply to it, and connecting its power wires to a switch

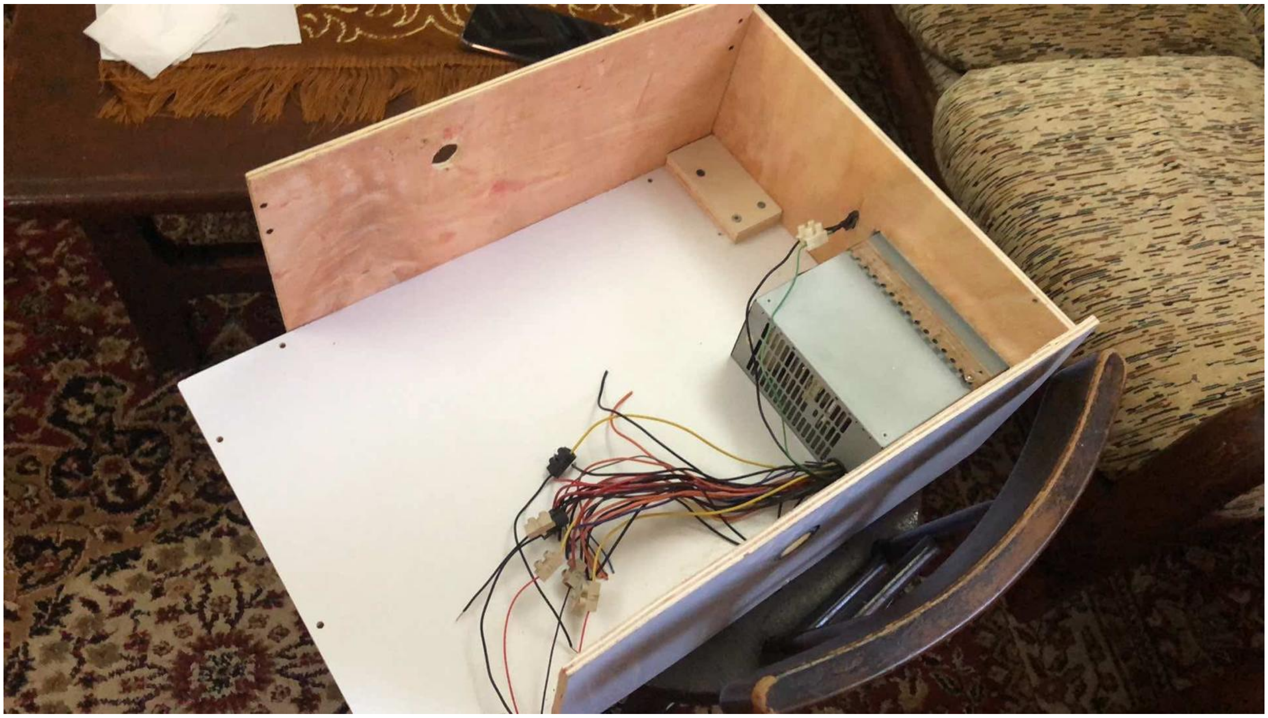


Figure 2: Base

Figure 3: The base from behind

3. We attached the main body of the printer on the base, and designed an upper case the covers the inner body.

Figure 4: Attached the body on the base and designing an uppercase

- **The Final result of the model**

1. The upper cover was now covered with white tape for aesthetic
2. A wooden base was pinned on where the page outputs

Figure 5: Complete model

- **ESP32 model connection (WiFi):**

We established a connection between the ESP32 and the host device (a laptop) using an IP address and port communication. The ESP32 served as a bridge for transferring PDF files from a smart device to the host, where a Python script handled file processing.

The script was responsible for normalizing the text, segmenting it into manageable chunks while accounting for the maximum capacity of Arduino’s serial buffer, and sequentially transmitting these chunks to the Arduino via a serial connection.

This setup enabled seamless data transfer, ensuring efficient processing and reliable communication between the devices.

- ## Challenges Through design

In our journey of designing the Braille printer, we encountered several challenges and successfully addressed them through diligent research and the implementation of best solutions.

1- **The Solenoid not embossing although the relay was being triggered:** The first mechanical relay we have used in the design was a One-channel 5V relay. Sometimes It was "clicking", but the solenoid wasn't being triggered, while it was on other "clicks". We have fixed this by using a Two-Channel Relay.

2- **The Y-axis not taking the bait:** The problem happened when the paper wasn't being pulled forward as it should have, in the beginning, we were using normal black-tape for friction and paper-pulling. We tried to make the black-tapped rolls thicker, which unfortunately did not work either. After thinking it thoroughly, comparing our roller (Y-axis) to the one used in printers, we determined that the usual black-tapped rolls cause as minimum of a friction as possible, and that they were too soft on paper, so we bought special tape with hard texture, measured the circumference of the Y-axis' rod, cut them evenly and equally and replaced them. Which has fixed the problem.

Standards and specifications

1. **The Braille language:** is a tactile writing system that enables individuals with visual impairments to read and write through touch. It consists of raised dots arranged in a grid, with different combinations of dots representing letters, numbers, punctuation, and even whole words. The Braille language has its own standard that governs its usage and ensures consistency in communication for people who are blind or visually impaired.

2. **WiFi technology for communication**: employed as the standard protocol for communication between the User using ESP, a Server with a web-interface, and the printer.

Constraints and design limitations

1. **Build the suitable design:** The process involves working with woods that are inherently tough to manipulate, especially in achieving the precise and intricate shape demanded by the desired design.

2. **Handling paper:** handle paper with very small thickness presents a set of intricate challenges that revolve around the paper's delicate nature and its interactions with various mechanical systems. The difficulties mainly stem from the fact that the paper's thinness makes it less substantial and more susceptible to a range of mechanical and environmental factors.

3. **Head of solenoid:** find the suitable head that knock the paper in correct dimension.

4. **Surface of printing area:** to be able to knocking the paper and the knock appear on the paper clearly should have mushy surface while the current surface is solid wood.

Chapter 4: Results and Analysis

- **Braille Dot Quality**

Our Braille printer produces Braille papers of good quality that adhere to the standards of the Braille language, ensuring correct distances between dots within each character and between characters.

The adherence to Braille language rules signifies the printer's capability to produce readable Braille. This is crucial for ensuring that the tactile reading experience is consistent and reliable for visually impaired users.

- **Precision Challenges**

Achieving absolute distance accuracy between dots can be challenging due to various factors. For instance, the solenoid's precision may not be optimal.

The solenoid's precision is a key factor in the accuracy of Braille dots. Any deviations can affect the clarity of the printed Braille. Exploring higher-precision solenoids or alternative actuation mechanisms could enhance the accuracy.

- **Paper Handling**

Handling the paper presents difficulties due to its extremely small thickness and sensitivity to external factors.

Thin paper can be difficult to manage, leading to potential misalignment or damage during printing. Implementing better paper handling mechanisms, such as improved feed systems or using thicker, more durable paper, could mitigate these issues.

Chapter 5: Discussion

Individuals with visual impairments encounter numerous hurdles when seeking to accuse printed text material in Braille. Among these challenges, a prominent one is the considerable expense associated with purchasing Braille printers, rendering them inaccessible to a substantial portion of the population that could greatly benefit from their utility. Furthermore, the availability of Braille printers presents an additional obstacle.

We've developed an affordable Braille printer that's easily accessible whenever required. This printer is made up of readily available basic parts found in the market. This creative solution enables people with visual impairments to conveniently obtain the resources they need.

Chapter 6: Conclusions and Recommendation

We have constructed a Braille printer with a low cost that is readily available when needed. This printer is composed of simple components that are readily available on the market. This innovative solution allows individuals with visual impairments to easily access to the resources.

After studying the project during the design stage, we chose the most efficient approach to its implementation. Our goal was to achieve high performance based on the hardware components and project design, while also ensuring cost-effectiveness and high availability.

We gained valuable insights into effectively working with hardware components by thoroughly studying datasheets and component documentation. This comprehensive approach allowed us to understand the intricate details of each component's functionality and integration. In addition to mastering the technical aspects, we honed our skills in designing a system that ensures stability and reliability.

By delving into the datasheets, we were able to get information about pin configurations, electrical characteristics, and communication protocols.

Furthermore, our journey involved not only understanding individual components but also orchestrating their harmonious collaboration. By thoughtfully addressing these aspects, we were able to create a system that remains good and dependable under varying conditions.

Future Works :

1- Adding another pin for embossment, and dividing the X-axis in half, making both work their half of the row, which doubles speed ideally.
2- Make the printer achieve an automated paper-feed system, and accept different sizes of papers such as A5, A3.
3- Make the printer multilingual.

References

[1] The Editors of Encyclopaedia Britannica, "Braille," Encyclopedia Britannica, Jan. 20, 2025. [Online]. Available: https://www.britannica.com/topic/Braille-writing-system. [Accessed: Feb. 2, 2025].

[2] K. Carey, "The Opportunities and Challenges of the Digital Age: A Blind User's Perspective," Library Trends, vol. 55, no. 4, pp. 767-784, 2007. [Online]. Available: https://dx.doi.org/10.1353/lib.2007.0030. [Accessed: Feb. 2, 2025].

[3] Index Braille, "Everest-D V5 cut-sheet fed Braille printer," Index Braille, [Online]. Available: https://www.indexbraille.com/en-us/braille-embossers/everest-d-v5.[Accessed: Feb. 2, 2025].

[4] Braillo, "Braillo 300 S2 Braille Embosser," Braillo Braille Printers and Braille Embossers, [Online]. Available: https://braillo.com/braillo-300-braille-embosser/.[Accessed: Feb. 2, 2025].

[5] Freedom Scientific, "VersaPoint Duo Interpoint Braille Embosser User's Guide," Freedom Scientific, Sept. 2002. [Online]. Available: https://support.freedomscientific.com/Content/Documents/Manuals/Legacy/VersaPointDocs/VersaPointDuo_UserMan.pdf. [Accessed: Feb. 2, 2025].

[6] BrailleRAP, "BrailleRAP," BrailleRAP, [Online]. Available: https://www.braillerap.org/en/index.html. [Accessed: Feb. 2, 2025].

[7] ccampos7, "OpenBraille, a DIY Braille Embosser," Instructables, [Online]. Available: https://www.instructables.com/OpenBraille-a-DIY-Braille-Embosser/. [Accessed: Feb. 2, 2025].

[8] Hackaday, "Braille Printer," Hackaday, [Online]. Available: https://hackaday.io/project/20239-braille-printer.[Accessed: Feb. 2, 2025].

Appendices

- **Arduino Mega code:** https://github.com/samimassoud/Braille-Printer-GP2/tree/main/Arduino

- **ESP8266 D1 mini code:** https://github.com/samimassoud/Braille-Printer-GP2/tree/main/ESP

- **Web application code:** https://github.com/samimassoud/Braille-Printer-GP2/tree/main/Server/Python-Script