Microcontroller Based Braille Translator for Visually Impaired

Priyanka D. Joshi¹ Prof. R. R. Maggavi² Prof. A. V. Deshmukh³

¹M.Tech Student ²Professor ³Assistant Professor

¹Department of DECS, ¹²Department of Electronics & Communication Engineering
¹²Maratha Mandal Engineering College, Belagavi, India ³Gogte Institute of Technology
Belagavi, India

Abstract— All over the world, persons who are not able to visualize (blind) have used Braille as the primary means of accessing information. Almost all countries have adapted the system ‘Braille’ as a universal approach that works to get the information for visually impaired. This paper describes the hardware implementation of a text to English Braille translator using Arduino UNO. Till now all Braille translators are FPGA based for its parallel programming, whereas in this paper it will be implemented on Arduino UNO. Arduino UNO is a powerful, cost effective and compact disc which occupy very less space. The translator is based on the translating algorithm, proposed by Paul Blenkhorn [1].

Keywords: Braille Translation, Arduino UNO, LCD, Memory Module, Relay Drive, Algorithms, Visually Impaired

I. INTRODUCTION

Blind people are an integral part of the society. However, their disabilities have made them to have less access to computers, the Internet, the e-books also the high quality educational software, than the people with clear vision. Consequently they have not been able to improve on their own knowledge, and have significant influence and impact on the economic, commercial & educational ventures in the society. One way to narrow this widening gap is necessary for them to develop a system, within their economic reach and which will empower them to communicate freely and widely using the softcopy of books or any other information infrastructure.

Braille a linguistic model, was invented in the year 1829 by a blind Frenchmen, Louis Braille. Braille comprises of a rectangular in shape with six-dot cell on it. Using one or more of the six dots it can have up to 64 possible combinations (26 = 64). Braille is embossed by hand or with a machine onto thick paper, and read by the blind people with the fingers moving across on top of the dots. Each letter in Braille script comprises of six dots arranged in three rows and two columns. This six dots arrangement is termed as a “cell” [1]. An example of a Braille letter cell has been shown in Fig.1.1. The main advantage of Braille is that it encourages the visually impaired to be independent in all walks of life, get the information of the world around.

Fig. 1: Braille Cell

![Fig. 1: Braille Cell](image1)

Since Braille become one of the most important ways for the blind to learn and obtain information, translating normal text into Braille became a necessity in today’s life. Today, most Braille translators are computer based and use the American Standard Code for Information Interchange (ASCII). Paul Blenkhon’s proposed a system to convert text into Standard English Braille [1]. Apart from the English alphabets there are various symbols we can use while writing a text and same can be translated in Braille.

II. BLOCK DIAGRAM & WORKING

The block diagram given below shows the basic blocks required to execute the result in Braille language. The Memory (SD card) is the input and the Braille cell which is connected or mounted on the relay mechanism is the output. The basic and most important block of this section is Arduino UNO which is the heart of this project which will convert the ASCII data to binary. We need to interface the memory of the SD card module and LCD (16x2) with Arduino UNO. So when any text file can be saved in the memory it will interact with Arduino UNO and the data (alphabets/word) will be displayed on the LCD. For this we
need to go through with the codes for interfacing and most importantly the conversion of ASCII to binary.

III. PROPOSED SYSTEM

Braille is a method of representing characters through a pattern of raised dots so that the blind can read by the sensation of touching. Written communication between two people is an easy task provided that they can both read and write the same language. So this project explains “To design a translator for converting English text to Braille code and this Braille code into vibration signal through vibration motors & can be read on a Braille cell membrane”. The vibration produces more sensitivity in fingers.

Braille Software Algorithm: When English text is translated to Braille code, the steps are as follows:
1) Read the input character from the text file stored in SD card.
2) Separate the words on the basis of blank space.
3) Break the word into single character.
4) Access the Braille database.
5) Display the character on LCD screen along with the Braille cell as output.

The detail procedure will be explained in the next section. The system has been implemented in an Arduino UNO ATmega 328 evaluation board.

IV. HARDWARE IMPLEMENTATION

Figure 1.4 shows the block diagram of the module for reading text of the system. In this module the e-books and e-documents, blogs etc. is sent to the Graphical User Interface (GUI) on the define code running in the PC. The American Standard Code for Information Interchange (ASCII) value of the character sent from the Braille database is converted to the corresponding Braille code using a conversion algorithm. This conversion program is written in Arduino 1.0 programming language. The microcontroller board used here is the Arduino UNO ATmega328 development board by Open Source Electronics. The output of the microcontroller Arduino UNO board is taken from the general purpose input/output pins of the development board in the form of voltages that is either 0 Volts or 5 Volts.

Since we have used six vibration motors and the Braille cell contains only six dots, only six of the Input/output pins of the development board are used. A six bit number in binary/hexadecimal form is obtained from the output of the microcontroller corresponding to the Braille code of the character. The output from the six Input/output pins is further given to the vibration motor through relay circuit. This driver circuit is used for voltage conversion which will be suitable for the safe operation of the vibration motor. Any visually impaired person can read through touch sensation and understand the English characters through the vibration of the motors. Similarly the whole word or sentence is converted into Braille vibration codes.

V. SOFTWARE IMPLEMENTATION

The Arduino runs a simplified version of the C programming language, with some extensions for accessing the hardware. It is not the complete version of C & Java thus called “hardwire”. All Arduino instructions are one line. The board can hold a program hundreds of lines long and has space for about 1,000 two-byte variables. The Arduino executes programs at about 300,000 source code lines per sec. The following flowcharts in Figure 1.5 and 1.6 explains the execution.

The code written in Arduino 1.0 software is generalized code and all the characters are stored in Braille database. The flowchart in Figure 1.6 gives the complete overview of the code.
The above flowchart shows the algorithm to initialize LCD and SD card. The ‘M.txt’ text file is searched and if the data is available, it is read alphabet by alphabet.

The next flowchart Figure 1.7 explains the function ‘ShowOnBoard’. This function checks for the character and activates the respective motors depending on the Braille code. The ShowOnBoard function checks the character alphabet by alphabet and make the respective pins low or high and generates the output on LED’s and the motors.

At last compile the code and dump or upload the code on hardware. The steps are:
1) Upload or Dump the code.
2) Connect the USB cable to the Arduino UNO board.
3) Go on Tools menu select the Serial Port, Board- Arduino Uno.
4) Check the output on the hardware.

Fig. 7: Flowchart to Read Character & Run the Respective Motors

VI. RESULT

Implemented Circuit for the Design System, the hardware setup is shown along with the vibrating motors and LEDs which will show the Braille equivalent of text read from the memory module. For the use of visually impaired people, the feel of vibration will be considered as a dot and Braille language of the respective alphabet of the word can be read.

The figure 1.8 shows the hardware setup when power supply is provided to the Arduino board. The LCD module connected shows a message “Braille Project”.

The below figures depicts the next step where the memory module is initialized. The Arduino UNO board now starts reading the text of the memory alphabet-by-alphabet. For the execution of the code written in Arduino board, a test message ‘ONE LAST RIDE’ is saved in the memory module.

Fig. 8: LCD Screen Displaying ‘Braille Project’ At Start-Up

The figure two and three LEDs are glowing respectively and corresponding vibration motor is also running. When placed in Braille dots arrangement, the user can feel and understand the word according to Braille language. The objective here which can be placed again is to build an arrangement where
the visually impaired person can read each and every word effectively, one after the other.

VII. CONCLUSION & FUTURE SCOPE

The design and implementation of an Arduino UNO microcontroller based, text-to-Braille translator has been presented. In this paper, we have used Grade 1 Braille conversion. In its current version, the system can be used in embedded and high-performance applications.

However, there are several improvements which will be incorporated in future versions of the hardware translator. For example, the current system is a stand-alone component. Its structure has to be changed for every individual application.

For further future scope the texts to be translated but the translated Braille English text to be stored in a PC as a file format so that it can be connected to Braille printer and the user can get the hard copy of it. Braille printers are easily available in market but with the high cost. Also for further improvement, a multi-language-Braille translator will be considered. Look-up tables for different languages could be stored in flash memory so that when translation of text in a particular language is required, the microcontroller loads the corresponding look-up table. Thus, we hope that the future hi-tech improvement will enlighten their life.

REFERENCES

[7] A System for Fast Text-to-Braille Translation Based on FPGA’s by Xuan Zhang, Cesar Ortega-Sanchez and Iain Murray.
[10] Hardware-Based Text-to-Braille Translator by Xuan Zhang, Cesar Ortega-Sanchez and Iain Murray, ASSETS’06, October 22-25, 2006, Portland, Oregon, USA.