TEXT TO BRAILLE CONVERTER WITH LOW-COST REFRESH DISPLAY

PROJECT REPORT

submitted by

AKSHAI R NAIR

Reg. No: MAC16EE008

BIJU JOSEPH

Reg. No: MAC16EE037

BINILA BINU

Reg. No: MAC16EE039

ELDHO JOHN MENACHERY

Reg. No: MAC16EE048

to

the APJ Abdul Kalam Technological University
in partial fulfillment of the requirements for the award of the Degree

of

Bachelor of Technology
In
Electrical and Electronics Engineering

Department of Electrical and Electronics Engineering

Mar Athanasius College of Engineering
Kothamangalam, Kerala, India 686 666

JULY 2020

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
MAR ATHANASIUS COLLEGE OF ENGINEERING
KOTHAMANGALAM

CERTIFICATE

This is to certify that the report entitled **TEXT TO BRAILLE CONVERTER WITH LOW-COST REFRESH DISPLAY** submitted by **Mr. Eldho John Menachery (MAC16EE048)** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Electrical & Electronics Engineering is a bonafide record of the project carried out by him under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Prof. Neetha John	Prof. Beena M Varghese
Project Guide & Project Coordinator	Head of the Dept.

Dept. Seal

10-06-2020

ACKNOWLEDGEMENT

It is a great pleasure to acknowledge all those who have assisted and supported us for successfully completing the first phase of our project.

First of all, We thank God Almighty for his blessings as it is only through his grace that we were able to complete our project successfully.

We extend our sincere thanks to our project coordinator and project guide Prof. Neetha John, Associate Professor, Department of Electrical and Electronics Engineering and all other members of the Electrical and Electronics Engineering Department for sharing their valuable comments during the preparation of the project.

We are also grateful to Prof.Acy M Kottalil, former Head of Electrical and Electronics Engineering Department and Prof.Beena M Varghese, Head of Electrical and Electronics Engineering Department for their valuable guidance as well as timely advice which helped us a lot during the preparation of the project.

We are deeply indebted to Dr. Mathew K, Principal, Mar Athanasius College of Engineering for his encouragement and support.

We whole - heartedly thank all our classmates, for their valuable suggestions and for the spirit of healthy competition that existed between us.

ABSTRACT

The braille system of encoding letters and numbers into an arrangement of up to 6 raised dots was devised back in the 1800s and has since been the method taught to visually impaired people. Consequently, there has also been an increase in the number of braille books available. These books are printed using special equipment which emboss the braille cells onto the paper. However, its estimated that less then 1 percent of the books have braille version. Also, the printing process being slower and more expensive than traditional printing, naturally raises the price of braille books. An open source text to braille scanner along with a unique low-cost refreshable braille display is proposed. The display generates all Braille characters by simple sliding arrangement of two elven-slot pre protruded sliders utilizing permutations of the slots. The display requires extremely low power just to carry out linear sliding of the sliders without the need of individual slot excitation.

TABLE OF CONTENTS

Contents | Page. No.

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

"Living life easy with eyes closed". It is obvious that this thought which has paved the way for a scrutiny of various technologies that can provide ways of reading for the blind which is often enough for them to visualize things. Of the 37 million people across the globe who are totally blind, over 15 million are from India, a situation that has left us crestfallen. According to this statistic the visually impaired people cannot be ignored and remain solid as they form a major part of our population. Anyway, there is no doubt that most of them are self-sufficient enough and Visual impairments have not imparted much restrictions in their path of learning as we have people like Helen Keller standing forefront in the worldly quest of success. However, the increasing percentage of blind have always left the question of how to improve their mode of accessing information. It had been revealed that Brain circuits which normally handle information from the eye can switch gears to tactile receptibility in visually impaired people which further accelerated the researches behind Braille devices but the fact that braille display technology has not changed significantly for 35 years is astonishing while the interaction paradigms for personal computing is continually subjected to changes. As of this date, number of companies are currently manufacturing a braille display with more than one line. The major throwbacks on the road of marketing braille displays as the primary reading modality for blind are:

1. The increasing cost of braille displays.
2. The decline in support for teaching blind students and drop in their literacy rate.
3. The increasing cost of hard copy of braille textbooks.

We present a low-cost refreshable braille display which evolved out after taking the first two above mentioned causes of concern. We are aware of the fact that developing a Braille Display is no more a new technology but the widening possibilities of a braille display is still a credible factor to be explored further. Even as far back as 1916, a patent was filed for a spring-loaded refreshable braille display. Since then, a tremendous number of actuator technologies have been adapted to create refreshable braille including belt drives and tape drives.

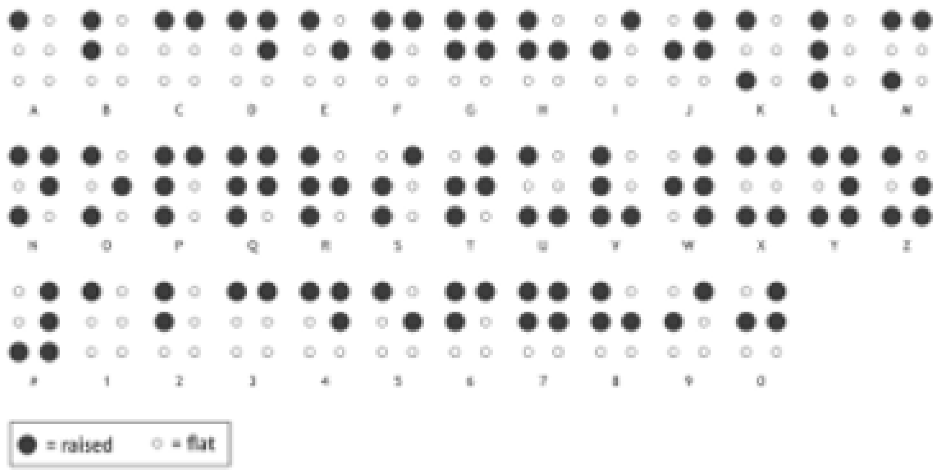


Figure 1.1: Braille cells for the English alphabet

The 3x2 dot array makes up a braille character as shown in figure 1. The tactile acuity is much poorer than the visual acuity and thereby the tactile field of view is much smaller than that of the visual field of view. Hence the patterns that make up the braille characters are much simpler than the globally processed word shapes followed by us which is more akin to vision. Combination of six solenoids can represent 64 braille characters in either up or down position. The dots are sized to fit the space between the fingertip and flat surface of the device. Above all the device has a provision to produce the pronunciation of the word that is being depicted which makes it more advantageous as far as the students are concerned.

The major aspect is that it is cheap and affordable comparing to the existing braille displays because of its reduced manufacturing cost when it comes to bulk production. Currently, a maximum of nineteen Braille printers are available all over India. So it is obvious that with this limited infrastructure, number of braille textbooks made available for the blind people are quite less in amount and hence they are really very expensive. With the advent of this product,it would facilitate the primary education for students at a much higher pace.Thus initiating a substantial amount of public support for the educating the blind population and hence notable increment in their literacy rate can be expected in the near future.Another differentiating feature to pay attention is that it is equipped with Wi-Fi connectivity which helps it to be controlled from anywhere from distant places. This enables the visually impaired people to be able to develop an advanced communication channel with other normal or deaf-blind people.

1.1 OBJECTIVE

The overarching objective is to create a low cost,simple to use,and compact refreshable braille display that would be affordable for people in developing countries and at the same time provide functionality and quality that are appealing to users in developed countries.The cost of braille scanner is very high and cannot afford for every one those who have vision impairments.Refreshable braille display(RBD) are peripheral devices that display braille characters,usually be raising and lowering dots through holes in a flat surface. Young students should be exposed to paper braille as well as refreshable braille.The main objective is to develop refreshable braille display with low cost.The RBD only require a very light touch to produce braille characters. This device can be useful as a teaching tool for primary school students.This can make teaching more effective. There are only eighteen Braille printers available in India till now and scarcity in the number of textbooks had become a major issue confronted by the blind people. Moreover, updating the textbooks has become yet another tedious work with this limited infrastructure. This prevailing situation became the predominant reason behind the development of new technology for the upliftment of education system for blind students. This led to the emergence of Low Cost Refreshable Braille Display which mainly intends to provide primary education for Blind students with ease of teaching. This device can be implemented in a classroom at low cost which requires very little space and it is a portable device due to its compact nature.The solenoids which represent the six dots in the device ultimately give the sensation of the desired Braille character and are capable of getting refreshed between the occurrences of each character. In addition to all these, this device has additional advantages like adjustment of the speed of occurrences of characters and to view the previous character and so on. This highly advanced version of the refreshable Braille display will create an immense progression in the development of the society for the blind individuals.

1.2 OVERVIEW

Chapter 1- Presents an introduction to the project.

Chapter 2- Deals with the literature review of the project.

Chapter 3- This section discusses the construction details of the project.

Chapter 4- Software Simulation

Chapter 5-Hardware components Used

1.3 RELEVANCE

Traditionally Braille is printed on paper through a specially designed impact printer. However the costs of printing Braille textbooks as well as their bulkiness prove to be a major hindrance for their mass production. It so results that the blind have access to very limited resources. An alternative to this drawback is the use of Refreshable Braille Displays, which are electromechanical devices for displaying Braille characters. They are available in various kinds with display ranging from those that display a single character to those which can display a complete sentence. Nevertheless it proves to be an expensive technology affordable to the elite sections of the blind community. This paves the way for the need for a low cost refreshable Braille display which is cost effective for blind students. This device can display many character at a time and hence would be very useful as a teaching tool for primary school students. Apart from displaying braille characters, it can also provide the pronunciation of the character which is displayed. This can make teaching more effective.

1.4 COMPARISON OF DIFFERENT ACTUATORS

In response to the need of cheaper alternatives, there have been many proposed solutions over the years. These solutions use different "actuating" techniques ranging from solenoids to shape memory alloys (SMAs). A comprehensive comparison of different actuating techniques was carried out . A summary has been reproduced in Table 1.1 (rank : 1 = best, 4 = worst).

Actuator	Response Time	Size	Power Consumption	Cost
Piezoelectric	1	1	4	4
Solenoids	2	4	3	1
SMAs	4	2	1	2
Pneumatic	3	3	2	3

Table 1.1: Comparison of different actuators

CHAPTER 2

LITERATURE REVIEW

Shahruk Hossain Asifur Rahman[1] try to propose an affordable refreshable braille display using stepper motor. Their idea is using a specific pattern of dots that can create any braille alphabets using slider mechanism.They point out the limits of current system such as relibility and cost.The prototype can display one letter at a time . The device in the paper published can be made more compact by stocking motors opposite sides than side by side and improved ocr can read and identify letters and paragraphs at a time.

Nitin Ramesh, Aksha Srivastava, K. Deeba [2] has introduced the concept of OCR (Optical Character Recognition) to read text from images. This involves scanning a document containing text, and converting character by character to their digital form. Thus, it is defined as the process of digitizing a document image into its constituent characters. The system makes use of Tesseract OCR engine while eliminating various drawbacks, hence improving performance. The various preprocessing tools used are Text Region Extraction, Image Resizing, Binarization and Denoise etc.

Kaustubh Bawdekar, Ankit Kumar and Rajkrishna Das[3] propose an algorithm which enables the user to convert the text that normally have in our day to day usage into Braille Script and thus gives impetus for the visually impaired to read that text . The Product that will be very intuitive and simplistic in design that will enable the end user to feel familiar and at home with the product. As all textbooks will not be available in Braille script as well as Audio recordings of all textbooks are not available. Thus take the image of content in the textbook and it will be reproduced as Braille script for persons who are blind. Thus they have successfully created a Low Cost Text to Braille Converter and also added an added functionality of Text to Audio to help the visually impaired in reading books, or listening to them directly

Josephine Stella and Krishna Valsan[4] propose a Rule Based Algorithm for text to braille translator that will help the blind people. Rule-based technique is one of artificial intelligence technique. Rule-based methods, rule discovery or rule extraction from data, are data mining techniques aimed at understanding data structure, providing comprehensible knowledge hidden in data, providing logical justification for drawing conclusions.A software is developed to perform the translation of Text to Braille and support of translation to Braille from many languages with the translation following grade rules. This application is improvements to the existing system in terms of features and abilities in translating the text into Braille code.

Saad D. et. al. [5] presents a design and implementation of Optical Character Recognition with voice and text conversion. Algorithm is implemented which is based on Braille dot position. They discuss about basic modules, i.e. image processing, image acquisition, dot localization, segmentation and dot recognition. By following these steps, Braille Recognition has been successfully completed. The author has also described decimal Braille code generation.And also discuss about concept like image thresholding with beta distribution for the recognition of Braille characters and convert it into simple language text and analysis vision based printing.

CHAPTER 3

METHODOLOGY

The purpose of the project is to enable the visually challenged people belonging to all economic sections of the society to effectively learn and communicate through braille language. In the following chapter the working principle and the algorithm of the project is explained. A basic idea about how the project works can be understood.

3.1 BLOCK DIAGRAM

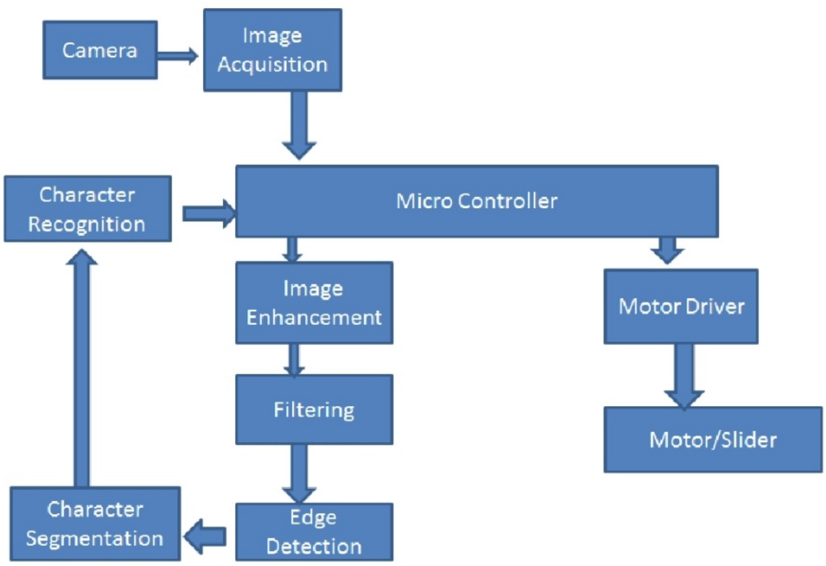


Figure 3.1: The block diagram representing the working of Text to Braille scanner with ultra low cost refreshable Braille

Camera is activated by push-button. Image is then processed in raspberry pi. After noise removal and thresholding, it is converted to text file with the help of tesseract and optical character recognition. First the captured image is enhanced.Then it is filtered for noise. After the noise has been smoothed out, the process of thresholding takes place where each pixel is compared to a threshold value and the pixel is set to maximum colour if its above the threshold else its set to minimum colour. After this the edges are detected and characters are segmented. Segmented characters are recognized by the micro controller which will then give commands to motor drivers to rotate the slider. Sliders are moved to generate desired alphabet. The combination of the sliders is displayed in the slot where the visually impaired person can touch. The slider has 12 sections and each section contains a dot or a blank space in a particular design. Any letter can be created by the combination of the sliders by adjusting their position. The slider is attached to motor and each letter will require two sliders.

3.2 TEXT TO BRAILLE SCANNING

A user can just snap a picture of the page he/she has to convert and the characters gets recognized by Google's open source Optical Character Recognition (OCR) engine, which is a free to use and modify software package that allows the acquisition of text from images. The scanning process is currently only supported for English text, but it can be expanded to other languages as well. Once the text is acquired, it is sent over serial connection to the arduino which triggers the stepper motors of the braille display. The scanning process is simple and easy to initiate as the process begins with a simple click of a button.

3.3 BRAILLE CELL

Braille symbols are formed within units of space known as braille cells. A full braille cell consists of six raised dots arranged in two parallel rows each having three dots. The dot positions are identified by numbers from one through six. Sixty-four combinations are possible using one or more of these six dots. A single cell can be used to represent an alphabet letter, number, punctuation mark, or even a whole word.

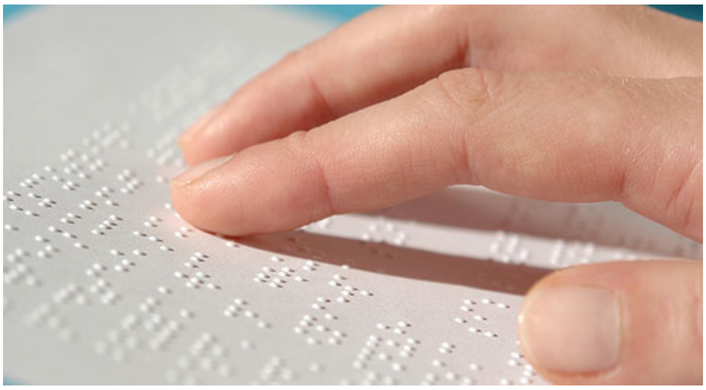


Figure 3.2: Braille cells

3.4 REFRESHABLE BRAILLE DISPLAY

The display is constructed in a modular fashion with a main base module and one or more extension modules. Each extension module can display one braille cell. By cascading more extension modules, the number of characters/braille cells displayed simultaneously can be increased as needed by the user. The base module houses the battery, the main logic unit. A 12V DC adapter is converted to the required 9V using an IC7089 voltage regulator. After the text from the image is obtained from the raspberry pi, the ATMEGA328 micro controller in the arduino maps each character within it to its corresponding braille pattern. It then sends control signals to the extension modules to display the appropriate braille pattern.

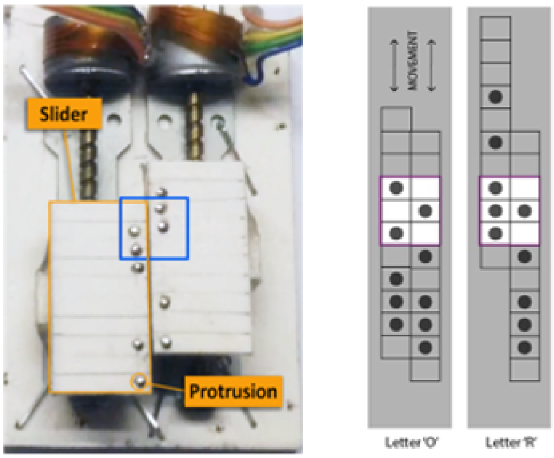


Figure 3.3: Pre-protruded sliders and Generating Patterns

An extension module consists of two small 2-pole, 4-wire stepper motors (typically used for sliding DVD drive trays in and out), and a custom dual stepper motor driver. The latter is designed using BJT transistors (BC547 and BC557) and a serial-in parallel out shift register (74HC595). The shift register allows chaining of several motor drivers by connecting the overflow pin of one register to the input of the next register. This allows several stepper motors to be operated using just 3 control lines (data, clock, and load) from the base module. Each stepper motor in the extension module is attached to a slider which has been pre-protruded with steel pinheads as shown in Fig.3.3. The identical sliders are arranged side by side and different positions of each slider produce the different braille cells under the opening at the center. Each slider has eleven "slots", five of which are protruding (forms the dots) and six are blank.

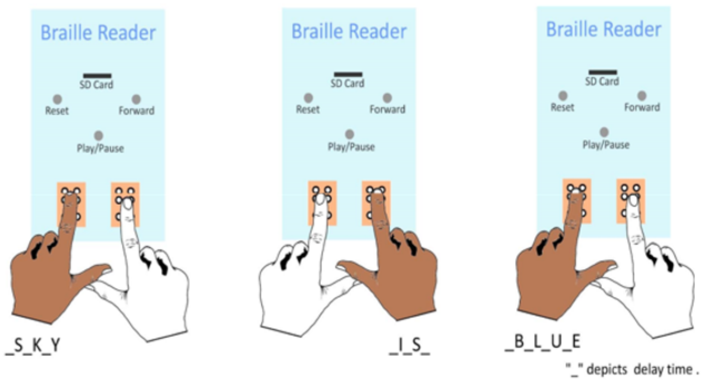


Figure 3.4: Process describing how the text 'SKY IS BLUE' read

Assuming a dot represents a 1 and a blank represents a 0, In order to form the braille cell for letter R , a 1-1-1 is needed on the left slider, and 0-1-0 on the right, with the first digit representing the top of the cell as shown in Fig.3.3. This is part of the processing that takes place in the base module. A button is clicked to display the first character and there is a five second halt for the user to feel the character. The micro controller directs the stepper motors to act accordingly through the motor drivers as described previously. The arrangement of the slots are not random – they have been specifically placed to form the most efficient arrangement (super-permutation) of dots/blanks than can generate any of the 8 possible patterns of 3 dots or blanks in the middle. That is to the say 12 is the minimum number of slots required to represent all 8 combinations with pre-protruded dots.

CHAPTER 4

SOFTWARE SIMULATION

The entire project is coded using python software. Using optical character recognition, the raspberry pi detects the text from the image. After the text is recognised each letter is sent over to the arduino uno which triggers the stepper motors to make the desired combination that represents the particular letter. In the following chapter the working and simulation of the optical character recognition is shown.

4.1 ALGORITHM

The steps involved in the execution of the optical character recognition are:

Step 1: Start

Step 2: Image of the page is captured by the camera module.

Step 3: Image is stored in a known location in the raspberry pi.

Step 4: The image is invoked to the program.

Step 5: Image undergoes resizing such that average character height is not above 20 pixels.

Step 6: The image is filtered for noise.

Step 7: The noise filtered image is taken in for thresholding.

Step 8: OCR is conducted for the processed image using tesseract.

Step 9: Text part of the image is extracted and stored as string.

Step 10: End

4.2 TESSERACT

Tesseract is an Optical Character Recognition (OCR) engine for various operating systems. It is a free software, released under the Apache License, Version 2.0 and development has been sponsored by Google since 2006. It has been considered as one of the most accurate OCR engine.

4.3 DRAWBACKS OF THE TESSERACT ENGINE

Tesseract works just fine if the input image that is given to it is free from any kind of background noise. An example is shown in Fig 4.1

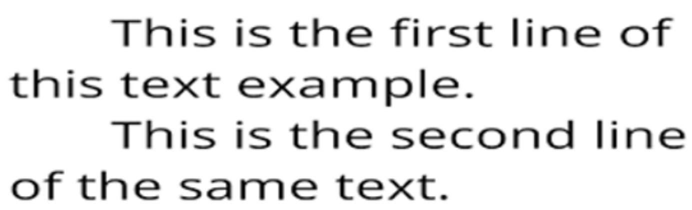


Figure 4.1: First input to Tesseract

This figure shown above is fairly clear and free from any background noise. When we apply Tesseract OCR to Fig 4.1, we get :

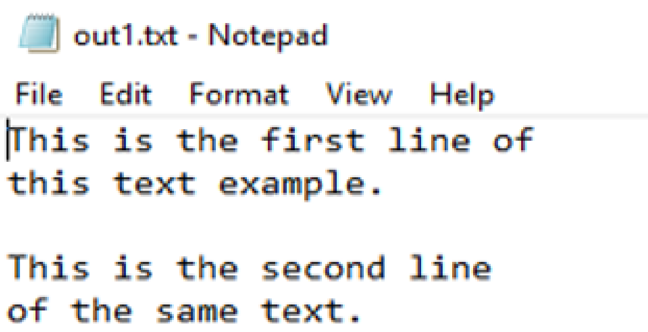


Figure 4.2: Output of the first input

The tesseract engine worked just fine. But when we input an image with a noisy as shown in Fig 4.3:

Figure 4.3: Second input to Tesseract

The output obtained when we apply Tesseract OCR to Fig 4.3 is shown in Fig 4.4 below.

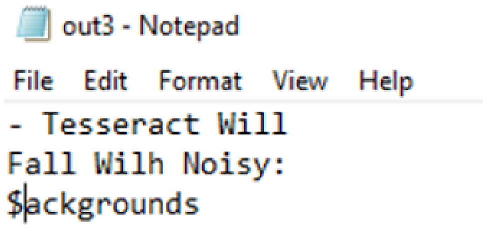


Figure 4.4: Output of the second input

There were very slight errors while conducting OCR on the above image. When an image with background pictures is inputted as shown in figure 4.5

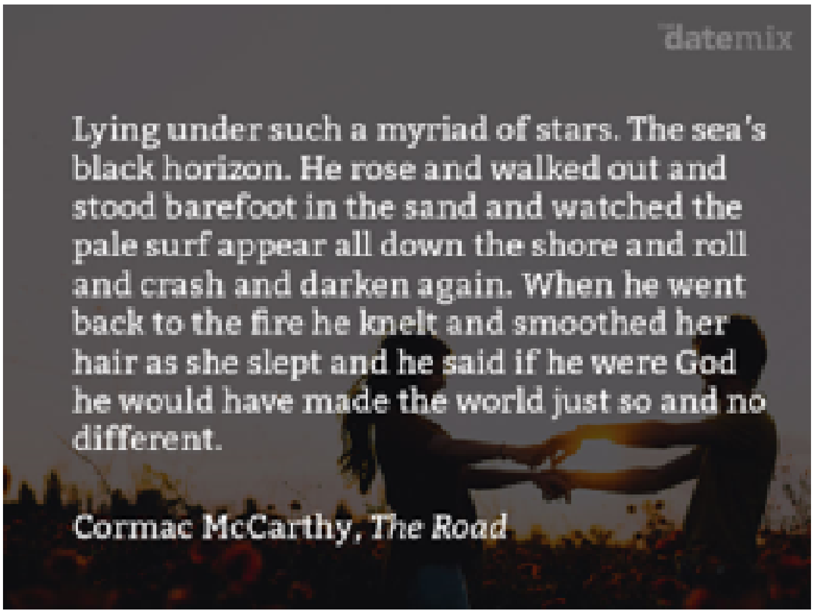


Figure 4.5: Output of the second input

There will be no output at all. To fix up all the issues related to OCR image, processing methods using python are introduced .

4.4 IMAGE PROCESSING USING PYTHON

4.4.1 TEXT REGION EXTRACTION

In order to extract the text regions, a document populated with words is shrunk to the level where the spaces between words become negligible. Thus when finding contours, only one contour encompassing all the words is obtained, and the unnecessary area around the paragraph can be cropped.

4.4.2 IMAGE RESIZING

Since Tesseract scans the image pixel by pixel, reading an image with an average character height above 20 pixels will increase computation time. This step also helps eliminating one major flaw of Tesseract, where the OCR accuracy drops for characters below a height of 20 pixels.

4.4.3 BINARIZATION AND DENOISE

Binarization is the process of conversion of an image to black and white. The high contrast between the background and the characters make OCR more accurate. The process of binarization is done through thresholding the color image. There are two types of thresholding, one is global thresholding and the other being adaptive thresholding. Thresholding changes the pixel color value to a minimum if it is below a threshold or to a maximum if it is above the threshold value.

4.4.4 EXECUTION OF PROGRAM

As seen above fig. 4.5 had no output by just conducting OCR using tesseract. Now after executing OCR using the python program for pre-processing the following output is achieved as shown in fig 4.6 and fig 4.7

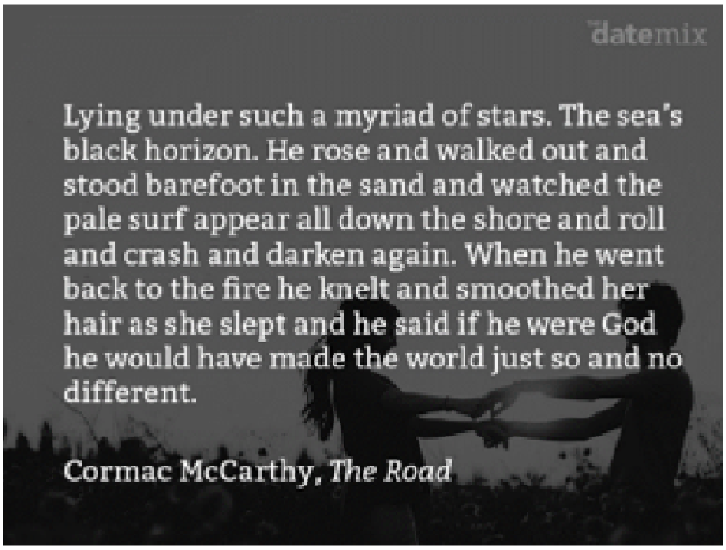


Figure 4.6: Output after reducing noise function is carried out of Fig 4.5

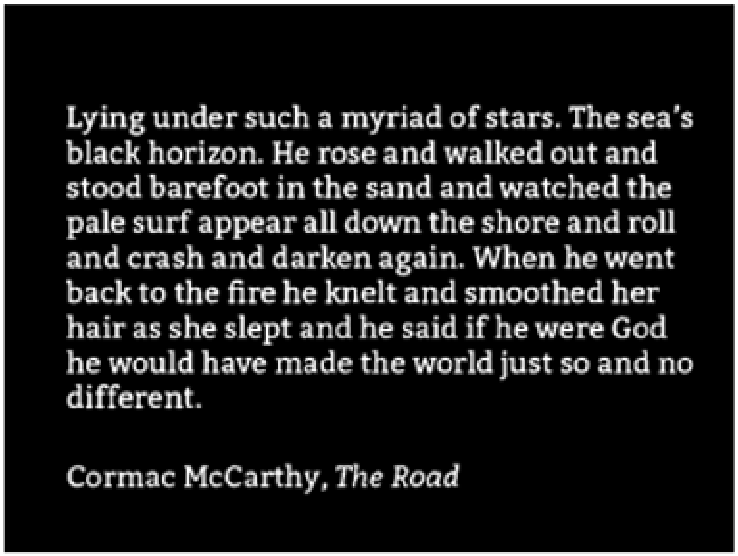


Figure 4.7: Output after thresholding noise function is carried out for Fig 4.6

Thus it is seen that with the help of the image processing techniques we are able to improve the efficiency of Tesseract engine.The final output is shown in fig 4.8

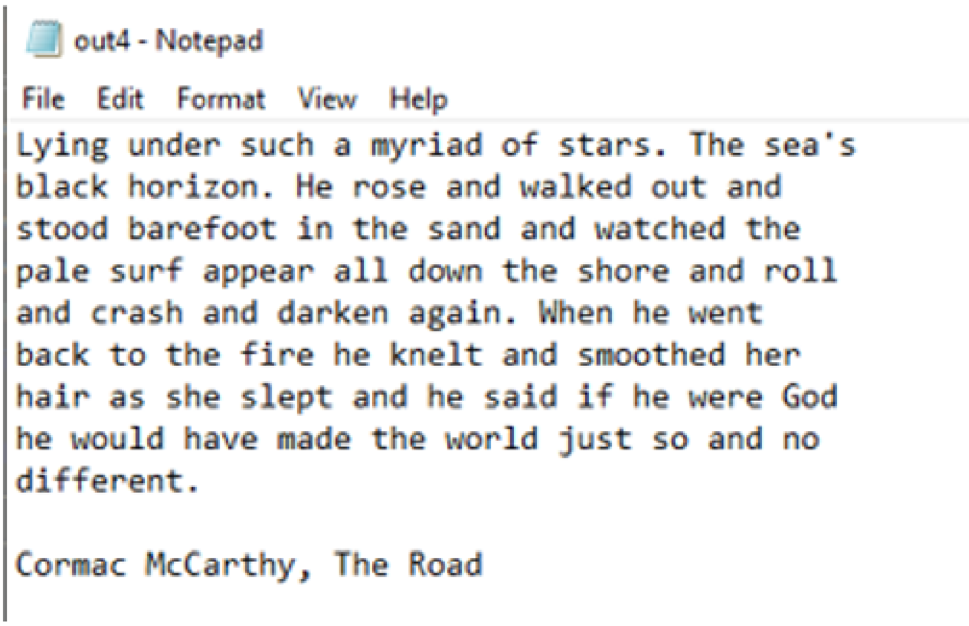


Figure 4.8: OCR carried out of Fig 4.5

CHAPTER 5

HARDWARE COMPONENTS USED

The key attribute of this project is the fact that all hardware components used are inexpensive and have easy maintenance. The various hardware components used in making this project have been explained in the following chapter.

5.1 STEPPER MOTOR

Stepper motor is an actuator transforming electric pulse into angular displacement. When receiving a pulse signal, the stepper motor will rotate a fixed angle (stepping angle) according to the direction set for the stepper motor.

The angular displacement volume can be controlled by controlling the pulse number to achieve the purpose of positioning accurately. The rotational velocity and acceleration of motor can be controlled by controlling the pulse frequency to achieve the purpose of speed control. .

Figure 5.1: Stepper motor

When the current flows through the stator winding of stepper motor, the stator winding produce a vector magnetic field. The magnetic field drives the rotor to rotate by an angle so that the pair of magnetic fields of the rotor and the magnetic field direction of the stator are consistent.

When the stator's vector magnetic field is rotated by an angle, the rotor also rotates with the magnetic field at an angle. Each time an electrical pulse is input, the motor rotates one degree further.

The angular displacement it outputs is proportional to the number of pulses input and the speed is proportional to the pulse frequency. By changing the order of winding power, the motor will reverse its direction. Therefore, it can control the rotation of the stepping motor by controlling the number of pulses, the frequency and the electrical sequence of each phase winding of the motor.

5.2 RASPBERRY PI

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. The raspberry pi comes in two models, they are model A and model B. The main difference between model A and model B is USB port. Model A board will consume less power and that does not include an Ethernet port. But, the model B board includes an Ethernet port The raspberry pi board comprises a program memory (RAM), processor and graphics chip, CPU, GPU, Ethernet port, GPIO pins, Xbee socket, UART, power source connector. And various interfaces for other external devices. It also requires mass storage, for that we use an SD flash memory card. So that raspberry pi board will boot from this SD card similarly as a PC boots up into windows from its hard disk.

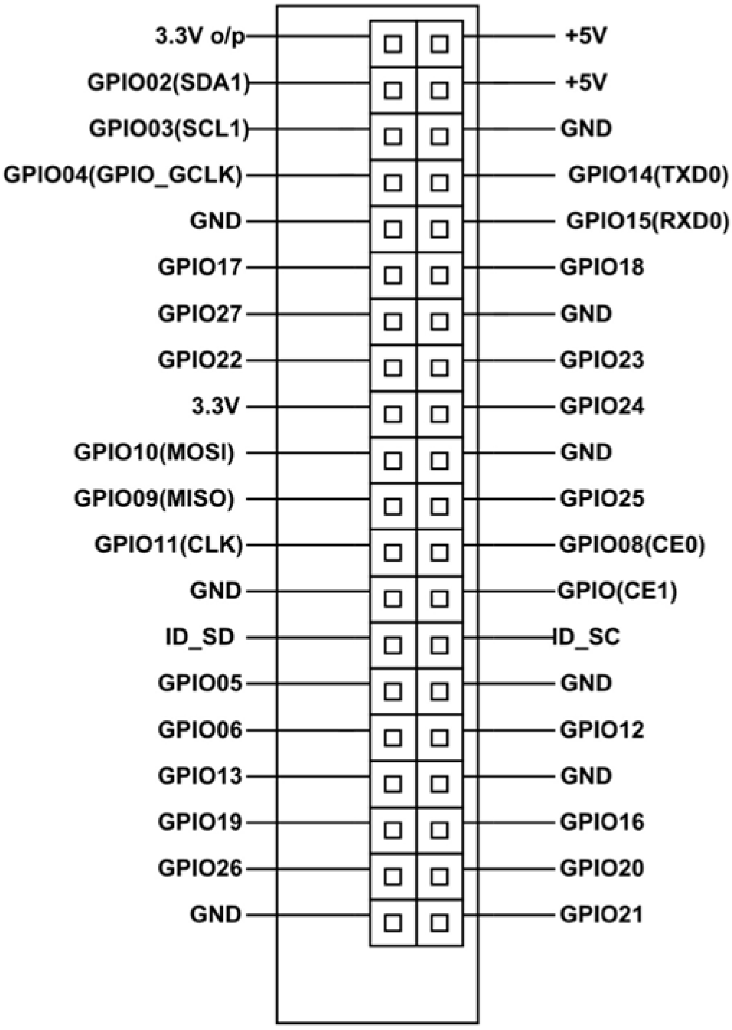


Figure 5.2: Pin diagram

Essential hardware specifications of raspberry pi board mainly include SD card containing Linux OS, US keyboard, monitor, power supply and video cable. Optional hardware specifications include USB mouse, powered USB hub, case, internet connection, the Model A or B: USB WiFi adaptor is used and internet connection to Model B is LAN cable.

5.3 HARDWARE SPECIFICATION OF RASPBERRY PI

- CPU: The CPU is a brain of this tiny computer that helps in carrying out a number of instruction based on the mathematical and logical formulas. It comes with a capacity of 64 bit
- Clock Speed and RAM: It comes with a clock speed of 1.4 GHz Broadcom BCM2837B0 that contains quad-core ARM Cortex-A53 and RAM memory is around 1GB GPU: It stands for graphics processing unit, used for carrying out image calculation. Broadcom video core cable is added in the device that is mainly used for playing video games.
- USB Ports: Two more USB ports are introduced in this new version, setting you free from the hassle of using an external USB hub when you aim to join a number of peripherals with the device.
- MicroUSB Power Source Connector: This connector is used for providing 5V power to the board. It draws 170 to 200mA more power than B model.
- HDMI and Composite Connection: Both audio output socket and video composite now reside in a single 4-pole 3.5mm socket which resides near HDMI. And the power connector is also repositioned in new B+ model and lives next to HDMI socket. All the power and audio video composite socket are now placed on the one side of the PCB, giving it a clean and precise look.
- The USB hard drive is available on the board that is used to boot the device. It is identical to the hard drive of regular computer where windows is used to boot the hard drive of the computer.
- PoE: B+ model comes with a facility of Power over Ethernet (PoE); a new feature added in this device which allows the necessary electrical current using data cables.

5.4 A4988 DRIVER

The A4988 is a complete Microstepping Motor Driver with built-in translator for easy operation. The driver has a maximum output capacity of 35 V and 2 A. It can operate bipolar stepper motors in full-, half-, quarter-, eighth-, and sixteenth-step modes.

A4988 has an inbuilt translator, so only two wires are required to connect it to controller board. The Driver provides five different step resolutions: full-step, half-step, quarter-step, eight-step and sixteenth step. Also, it has a potentiometer for adjusting the current output, over-temperature thermal shutdown and crossover-current protection.

Its logic voltage is from 3 to 5.5 V and the maximum current per phase is 2A if good addition cooling is provided or 1A continuous current per phase without heat sink for cooling.

Before using the motor, there's a small adjustment that we need to make. We need to limit the maximum amount of current flowing through the stepper coils and prevent it from exceeding the motor's rated current. There's a small trimmer potentiometer on the A4988 driver that can be used to set the current limit. You should set the current limit to be at or lower than the current rating of the motor.

Only two pins DIR and STEP of module is connected with Raspberry pi.STEP pin used to control the steps while DIR pin is used to control direction. Micro-step pins (MS1, MS2 and MS3) are used to operate the driver module in different step functions. Stepper Motor wires is connected with output pins (1A, 1B, 2A 2B) of driver module.

The 3 Micro-step pins (MS1, MS2 and MS3) are for selecting one of the five step resolutions according to the truth table given in figure . These pins have internal pull-down resistors so if we leave them disconnected, the board will operate in full step mode.

MS 1	MS 2	MS 3	Resolution
Low	Low	Low	Full Step
High	Low	Low	Half Step
Low	High	Low	Quarter step
High	High	Low	Eighth step
High	High	High	Sixteenth Step

Table 5.1: Truth Table

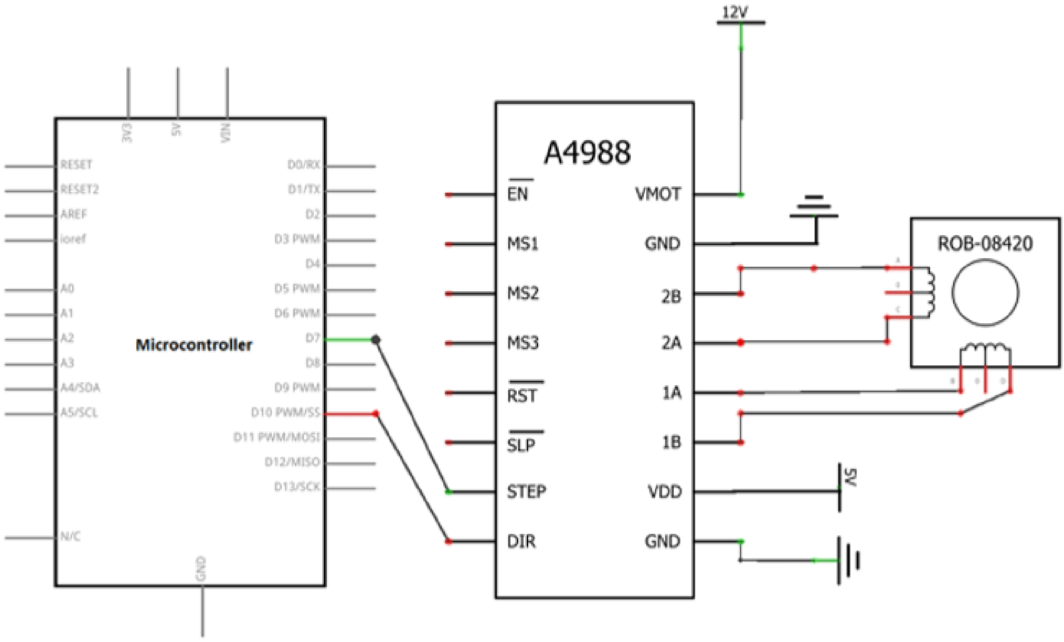


Figure 5.3: Connection diagram

Pin Name	Description
VDD&GND	Connected to 5V and GND of the controller
VMOT &GND	Used to power the motor
1A,1B,2A,2B	Connected to 4 coils of motor
DIRECTION	Motor direction control pin
STEP	Steps control pin
MS1,MS2,MS3	Microstep selection pin
SLEEP	Pin for controlling power states
RESET	

Figure 5.4: Pin Specification oF the Driver

CHAPTER 6

RESULT AND IMPLEMENTATION

To convert English text to braille, paper containing the English text is placed under the camera module. With a simple click of a button the camera will capture the photo of this page. This image is then processed by the program embedded in the raspberry pi. After various image processing techniques like thresholding and noise removal, the device is able to extract the text from the image. The text extracted is stored as a string and hence it is possible to access each and every character individually. This entire process of image capture and processing takes about five seconds. Then the user must click a button and the first letter is analysed and the raspberry pi determines the number of steps the motor must rotate based on predefined python code. This value is passed to the Arduino via serial communication using nanpy technique, where the arduino acts as a slave to the raspberry pi. The A4988 driver and the stepper motors are connected to the Arduino. The A4988 is a complete micro-stepping motor driver with built-in translator for easy operation. Now as the Arduino receives the number of steps information from the raspberry pi, it sends signals to the A4988 driver to turn on the stepper motor so that the necessary steps are completed. As the stepper motor rotates it pushes the slider forward forming the required combination of braille dots. Then there is a five second halt so that the user can touch and feel the combination. After five seconds the motor comes back to its original position. Then to form the next character the button is pressed again and the process repeats. This continues till all characters are read and converted to braille. In Fig. 6.1 the rough connection diagram is shown and Fig. 6.2 and Fig. 6.3 shows the working model of the project that has been developed.

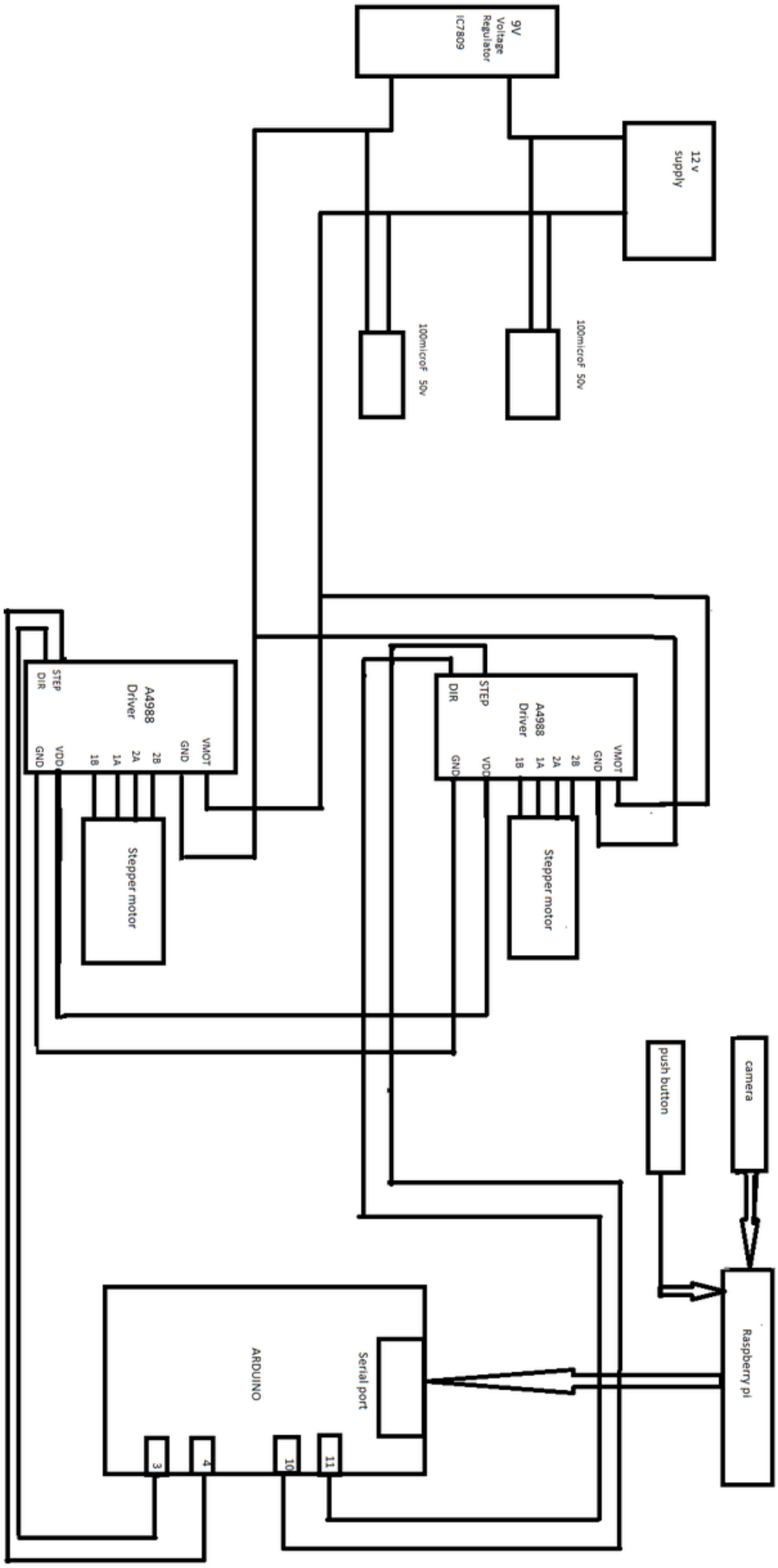


Figure 6.1: connection Diagram

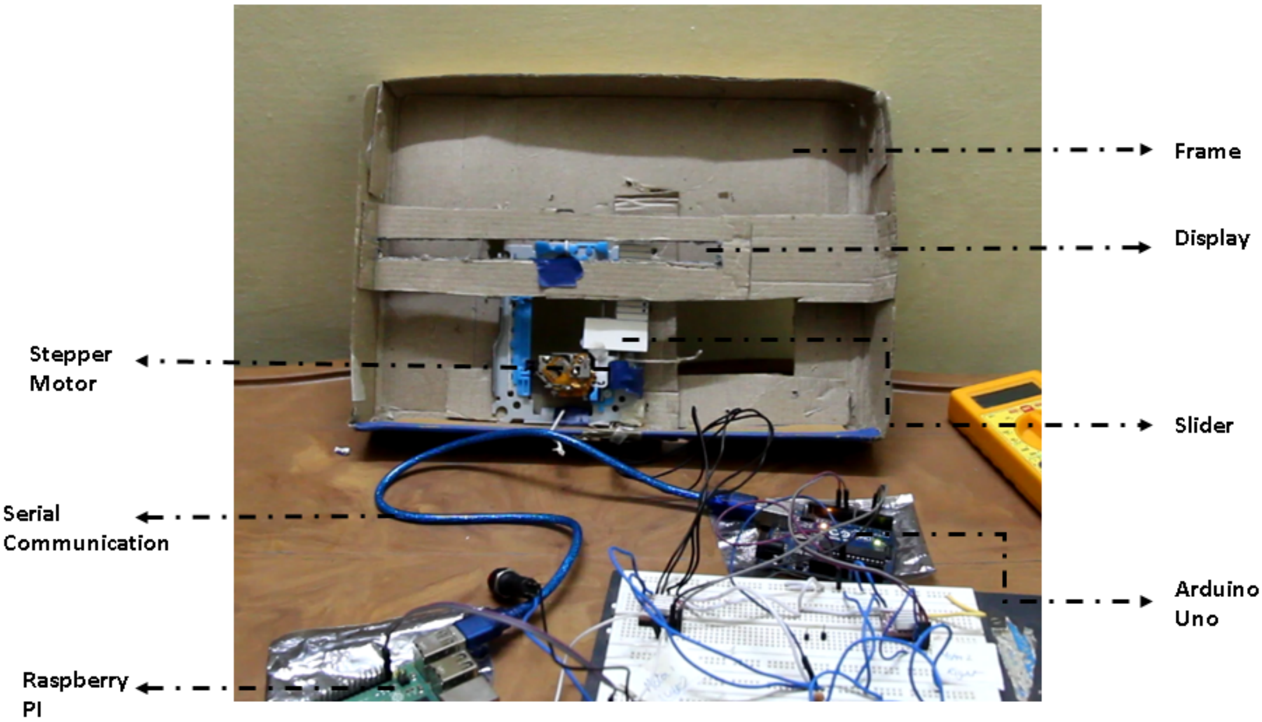


Figure 6.2: Working Model of Braille Scanner

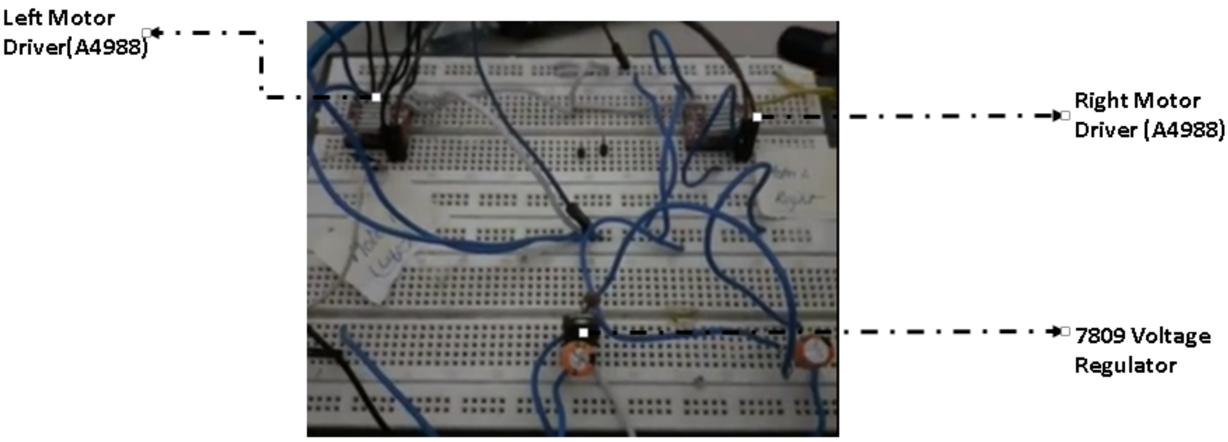


Figure 6.3: Drivers

CHAPTER 7

CONCLUSION

Even if there are many refresh braille display available in the market, they are extremely costly. They use expensive machinery that are difficult to repair and replace. Motor based slider display can effectively provide a solution for this problem. Even though they are little bulkier than piezoelectric based displays, it is more cheaper to manufacture and to maintain thereby making it affordable to all sections of the society. A single cell display has been successfully developed here. This project ensures that the visually challenged person does not have to depend on anyone or anything to read a book he or she desires. A large scale production model can be made with much compact motors and specially designed sliders. This will make design further compact, light and very affordable when compared to other display methods.

REFERENCES

[1] Shahruk Hossain,Asifur Rahman , *Text to Braille Scanner With Low cost Refreshable Braille Display*,IEEE TENCON 2018

[2] Nithin Ramesh,k Deeba, *OCR to Read Text From Image*, IEEE International Conference, 2015

[3] Kaustubh Bawdekar,Ankit Kumar, *Text To Braille Converter*, International Journal of Electronics and Communication Engineering and Technology, 2016.

[4] Josephine Stella, Krishna Valsan., *Text To Braille Conversion:A Survey*, International Journal Of Managenment And Applied Science, 2018.

[5] Saad D, *Braille Recognition and Transcription into Text and Voice*, International Biomedical Engineering Conference,2015.

[6] Nestor Falcon ,Miguel A., *Image processing Technique For Braille Writing Recognition*, 2017

[7] Lisa Wong, *A software Algorithm For Optical Recoginition of Embossed Braille*, IEEE International Conference,2008.

[8] Hardik A, *Optical Gujarathi Braille Regognition*, International Journal of Emerging Technologies, 2014.

[9] P Fieguth , *A Robust Probabilistic Braille Recognition system*, IJDER, 2011.

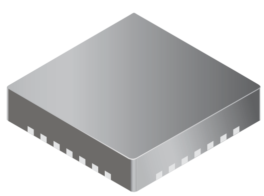
DMOS Microstepping Driver with Translator and Overcurrent Protection

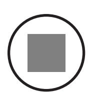
Features and Benefits

- Low $R_{DS(ON)}$ outputs
- Automatic current decay mode detection/selection
- Mixed and Slow current decay modes
- Synchronous rectification for low power dissipation
- Internal UVLO
- Crossover-current protection
- 3.3 and 5 V compatible logic supply
- Thermal shutdown circuitry
- Short-to-ground protection
- Shorted load protection
- Five selectable step modes: full, $^1/_2$, $^1/_4$, $^1/_8$, and $^1/_{16}$

Package:

28-contact QFN
with exposed thermal pad
5 mm × 5 mm × 0.90 mm
(ET package)





Approximate size

Description

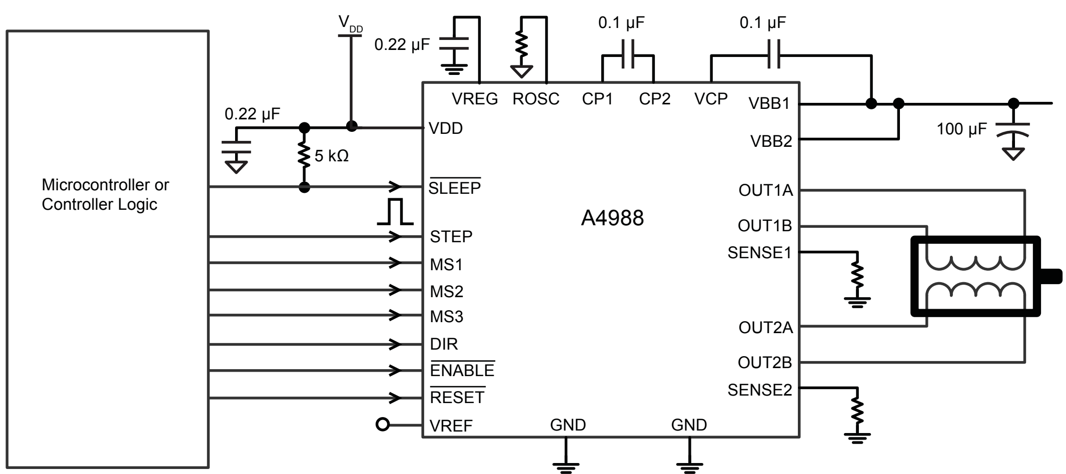
The A4988 is a complete microstepping motor driver with built-in translator for easy operation. It is designed to operate bipolar stepper motors in full-, half-, quarter-, eighth-, and sixteenth-step modes, with an output drive capacity of up to 35 V and ±2 A. The A4988 includes a fixed off-time current regulator which has the ability to operate in Slow or Mixed decay modes.

The translator is the key to the easy implementation of the A4988. Simply inputting one pulse on the STEP input drives the motor one microstep. There are no phase sequence tables, high frequency control lines, or complex interfaces to program. The A4988 interface is an ideal fit for applications where a complex microprocessor is unavailable or is overburdened.

During stepping operation, the chopping control in the A4988 automatically selects the current decay mode, Slow or Mixed. In Mixed decay mode, the device is set initially to a fast decay for a proportion of the fixed off-time, then to a slow decay for the remainder of the off-time. Mixed decay current control results in reduced audible motor noise, increased step accuracy, and reduced power dissipation.

Continued on the next page…

Typical Application Diagram



Description (continued)

Internal synchronous rectification control circuitry is provided to improve power dissipation during PWM operation. Internal circuit protection includes: thermal shutdown with hysteresis, undervoltage lockout (UVLO), and crossover-current protection. Special power-on sequencing is not required.

The A4988 is supplied in a surface mount QFN package (ES), 5 mm × 5 mm, with a nominal overall package height of 0.90 mm and an exposed pad for enhanced thermal dissipation. It is lead (Pb) free (suffix –T), with 100% matte tin plated leadframes.

Selection Guide

Part Number	Package	Packing
A4988SETTR-T	28-contact QFN with exposed thermal pad	1500 pieces per 7-in. reel

Absolute Maximum Ratings

Characteristic	Symbol	Notes	Rating	Units
Load Supply Voltage	V_{BB}		35	V
Output Current	I_{OUT}		±2	A
Logic Input Voltage	V_{IN}		–0.3 to 5.5	V
Logic Supply Voltage	V_{DD}		–0.3 to 5.5	V
VBBx to OUTx			35	V
Sense Voltage	V_{SENSE}		0.5	V
Reference Voltage	V_{REF}		5.5	V
Operating Ambient Temperature	T_A	Range S	–20 to 85	ºC
Maximum Junction	T_J(max)		150	ºC
Storage Temperature	T_{stg}		–55 to 150	ºC

Allegro MicroSystems, Inc.
115 Northeast Cutoff
Worcester, Massachusetts 01615-0036 U.S.A.
1.508.853.5000; www.allegromicro.com

Functional Block Diagram

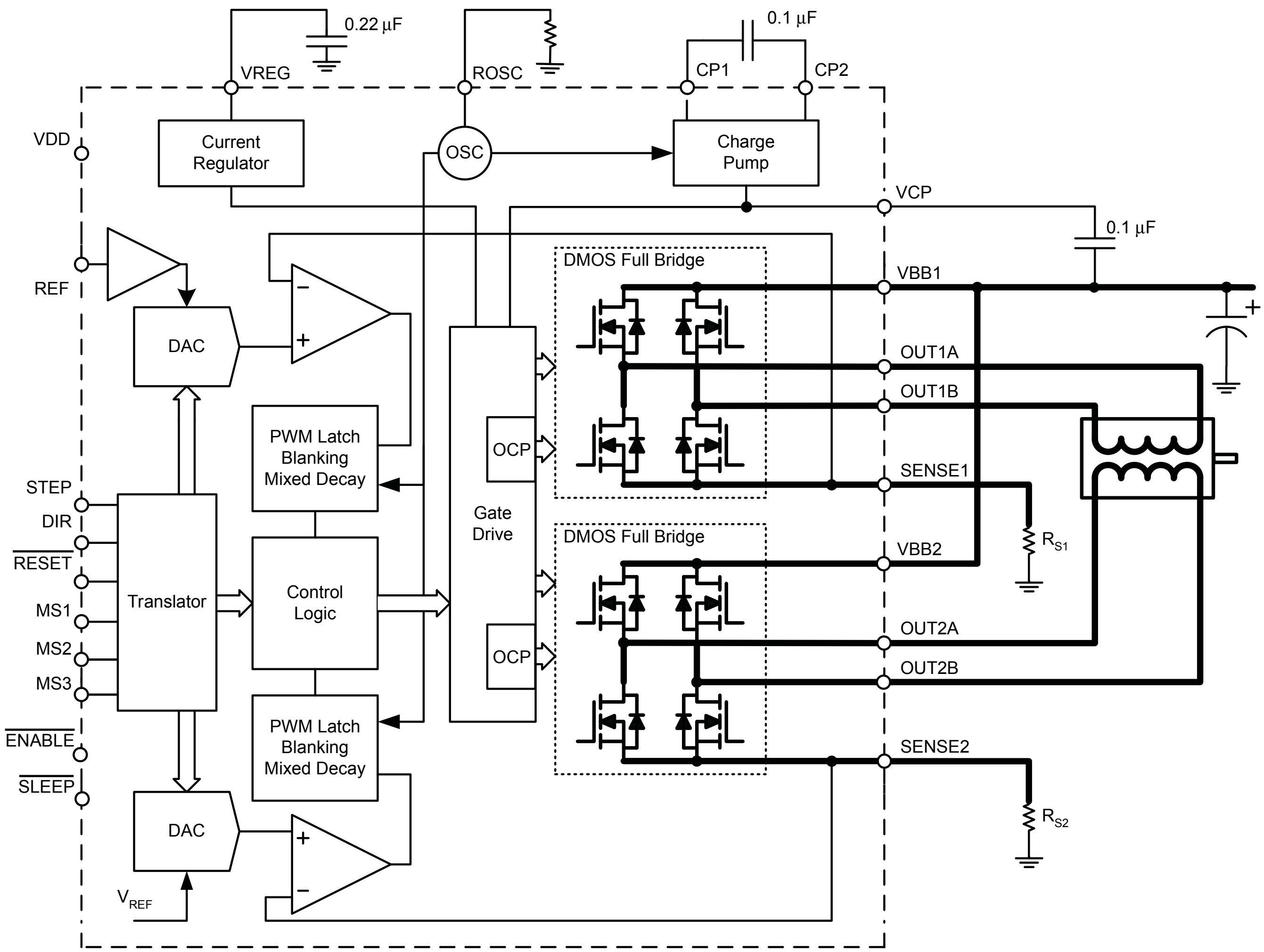


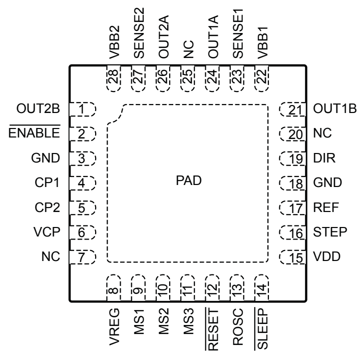
Table 2. Step Sequencing Settings

Home microstep position at Step Angle 45º; DIR = H

Full Step #	Half Step #	1/4 Step #	1/8 Step #	1/16 Step #	Phase 1 Current [% $I_{tripMax}$] (%)	Phase 2 Current [% $I_{tripMax}$] (%)	Step Angle (º)	Full Step #	Half Step #	1/4 Step #	1/8 Step #	1/16 Step #	Phase 1 Current [% $I_{tripMax}$] (%)	Phase 2 Current [% $I_{tripMax}$] (%)	Step Angle (º)
	1	1	2	1	100.00	0.00	0.0		5	9	17	33	–100.00	0.00	180.0
				2	99.52	9.80	5.6					34	–99.52	–9.80	185.6
			2	3	98.08	19.51	11.3				18	35	–98.08	–19.51	191.3
				4	95.69	29.03	16.9					36	–95.69	–29.03	196.9
		2	3	5	92.39	38.27	22.5			10	19	37	–92.39	–38.27	202.5
				6	88.19	47.14	28.1					38	–88.19	–47.14	208.1
			4	7	83.15	55.56	33.8				20	39	–83.15	–55.56	213.8
				8	77.30	63.44	39.4					40	–77.30	–63.44	219.4
1	2	3	5	9	70.71	70.71	45.0	3	6	11	21	41	–70.71	–70.71	225.0
				10	63.44	77.30	50.6					42	–63.44	–77.30	230.6
			6	11	55.56	83.15	56.3				22	43	–55.56	–83.15	236.3
				12	47.14	88.19	61.9					44	–47.14	–88.19	241.9
		4	7	13	38.27	92.39	67.5			12	23	45	–38.27	–92.39	247.5
				14	29.03	95.69	73.1					46	–29.03	–95.69	253.1
			8	15	19.51	98.08	78.8				24	47	–19.51	–98.08	258.8
				16	9.80	99.52	84.4					48	–9.80	–99.52	264.4
	3	5	9	17	0.00	100.00	90.0		7	13	25	49	0.00	–100.00	270.0
				18	–9.80	99.52	95.6					50	9.80	–99.52	275.6
			10	19	–19.51	98.08	101.3				26	51	19.51	–98.08	281.3
				20	–29.03	95.69	106.9					52	29.03	–95.69	286.9
		6	11	21	–38.27	92.39	112.5			14	27	53	38.27	–92.39	292.5
				22	–47.14	88.19	118.1					54	47.14	–88.19	298.1
			12	23	–55.56	83.15	123.8				28	55	55.56	–83.15	303.8
				24	–63.44	77.30	129.4					56	63.44	–77.30	309.4
2	4	7	13	25	–70.71	70.71	135.0	4	8	15	29	57	70.71	–70.71	315.0
				26	–77.30	63.44	140.6					58	77.30	–63.44	320.6
			14	27	–83.15	55.56	146.3				30	59	83.15	–55.56	326.3
				28	–88.19	47.14	151.9					60	88.19	–47.14	331.9
		8	15	29	–92.39	38.27	157.5			16	31	61	92.39	–38.27	337.5
				30	–95.69	29.03	163.1					62	95.69	–29.03	343.1
			16	31	–98.08	19.51	168.8				32	63	98.08	–19.51	348.8
				32	–99.52	9.80	174.4					64	99.52	–9.80	354.4

Allegro MicroSystems, Inc.
115 Northeast Cutoff
Worcester, Massachusetts 01615-0036 U.S.A.
1.508.853.5000; www.allegromicro.com

Pin-out Diagram



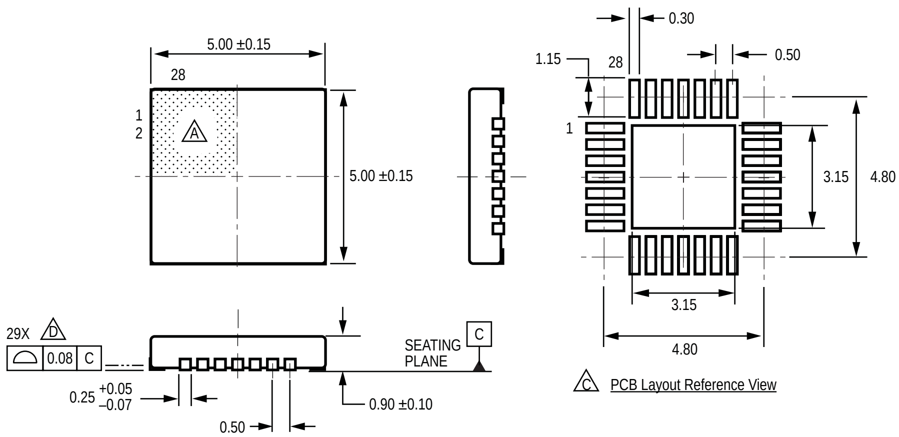
Terminal List Table

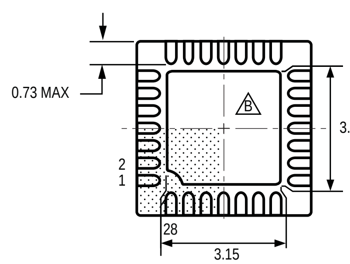
Name	Number	Description
CP1	4	Charge pump capacitor terminal
CP2	5	Charge pump capacitor terminal
VCP	6	Reservoir capacitor terminal
VREG	8	Regulator decoupling terminal
MS1	9	Logic input
MS2	10	Logic input
MS3	11	Logic input
$\overline{\text{RESET}}$	12	Logic input
ROSC	13	Timing set
$\overline{\text{SLEEP}}$	14	Logic input
VDD	15	Logic supply
STEP	16	Logic input
REF	17	G_m reference voltage input
GND	3, 18	Ground*
DIR	19	Logic input
OUT1B	21	DMOS Full Bridge 1 Output B
VBB1	22	Load supply
SENSE1	23	Sense resistor terminal for Bridge 1
OUT1A	24	DMOS Full Bridge 1 Output A
OUT2A	26	DMOS Full Bridge 2 Output A
SENSE2	27	Sense resistor terminal for Bridge 2
VBB2	28	Load supply
OUT2B	1	DMOS Full Bridge 2 Output B
$\overline{\text{ENABLE}}$	2	Logic input
NC	7, 20, 25	No connection
PAD	–	Exposed pad for enhanced thermal dissipation*

*The GND pins must be tied together externally by connecting to the PAD ground plane under the device.

Allegro MicroSystems, Inc.
115 Northeast Cutoff
Worcester, Massachusetts 01615-0036 U.S.A.
1.508.853.5000; www.allegromicro.com

ET Package, 28-Pin QFN with Exposed Thermal Pad





For Reference Only; not for tooling use
(reference JEDEC MO-220VHHD-1)
Dimensions in millimeters
Exact case and lead configuration at supplier discretion within limits shown

A Terminal #1 mark area

B Exposed thermal pad (reference only, terminal #1 identifier appearance at supplier discretion)

C Reference land pattern layout (reference IPC7351 QFN50P500X500X100-29V1M); All pads a minimum of 0.20 mm from all adjacent pads; adjust as necessary to meet application process requirements and PCB layout tolerances; when mounting on a multilayer PCB, thermal vias at the exposed thermal pad land can improve thermal dissipation (reference EIA/JEDEC Standard JESD51-5)

D Coplanarity includes exposed thermal pad and terminals

For the latest version of this document, visit our website:
www.allegromicro.com

Allegro MicroSystems, Inc.
115 Northeast Cutoff
Worcester, Massachusetts 01615-0036 U.S.A.
1.508.853.5000; www.allegromicro.com

ARDUINO UNO REV3

Code: A000066

The UNO is the best board to get started with electronics and coding. If this is your first experience tinkering with the platform, the UNO is the most robust board you can start playing with. The UNO is the most used and documented board of the whole Arduino family.

- **Arduino Uno** is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.. You can tinker with your UNO without worring too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

  "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

  You can find here your board warranty informations.
  https://www.arduino.cc/en/Main/warranty

Getting Started

You can find in the Getting Started section all the information you need to configure your board, use the Arduino Software (IDE), and start tinker with coding and electronics.
https://www.arduino.cc/en/Guide/HomePage

TECH SPECS

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA

Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

- ## OSH: Schematics

Arduino Uno is open-source hardware! You can build your own board using the following files:

EAGLE FILES IN .ZIP
https://www.arduino.cc/en/uploads/Main/arduino_Uno_Rev3-02-TH.zip
SCHEMATICS IN .PDF
https://www.arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf
BOARD SIZE IN .DXF
http://arduino.cc/documents/ArduinoUno.dxf

Programming

The Arduino Uno can be programmed with the (Arduino Software (IDE)). Select "Arduino/Genuino Uno from the Tools > Board menu (according to the microcontroller on your board). For details, see the reference and tutorials.

The ATmega328 on the Arduino Uno comes preprogrammed with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar; see these instructions for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available in the Arduino repository. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then rese ing the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See this user-contributed tutorial for more information.

Warnings

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Differences with other boards

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Power

The Arduino Uno board can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector.

The board can operate on an external supply from 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- Vin. The input voltage to the Arduino/Genuino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

- 5V.This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

- 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

- GND. Ground pins.

- IOREF. This pin on the Arduino/Genuino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.

Memory

The ATmega328 has 32 KB (with 0.5 KB occupied by the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).
https://www.arduino.cc/en/Reference/EEPROM

Input and Output

See the mapping between Arduino pins and ATmega328P ports. The mapping for the Atmega8, 168, and 328 is identical.

PIN MAPPING ATmega328P
https://www.arduino.cc/en/Hacking/PinMapping168

Each of the 14 digital pins on the Uno can be used as an input or output,
using pinMode() https://www.arduino.cc/en/Reference/PinMode,
digitalWrite() https://www.arduino.cc/en/Reference/DigitalWrite,
and digitalRead() https://www.arduino.cc/en/Reference/DigitalRead functions.
They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller.

In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.

- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.

- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

- LED: 13. There is a built-in LED driven by digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

- TWI: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analogReference().

- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

Arduino/Genuino Uno has a number of facilities for communicating with a computer, another Arduino/Genuino board, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows serial communication on any of the Uno's digital pins.
https://www.arduino.cc/en/Reference/SoftwareSerial

The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino Software (IDE) includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.
https://www.arduino.cc/en/Reference/Wire

https://www.arduino.cc/en/Reference/SPI

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino/Genuino Uno board is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino Software (IDE) uses this capability to allow you to upload code by simply pressing the upload button in the interface toolbar. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno board contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

Revisions

Revision 3 of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible with both the board that uses the AVR, which operates with 5V and with the Arduino Due that operates with 3.3V. The second one is a not connected pin, that is reserved for future purposes.

- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

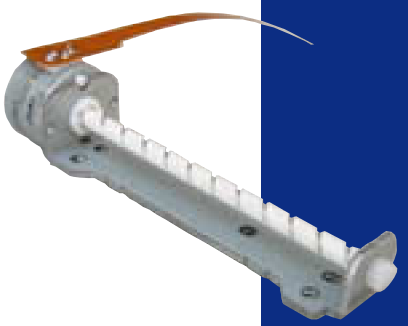
https://store.arduino.cc/usa/arduino-uno-rev3 12-8-17

NMB-MAT

Minebea-Matsushita Motor Corporation

eMINEBEA.COM

PL15S-020



PM Motor

PL Type

Applications

Head Drive for Floppy Disk Drive
Pick Up Drive for CD
Pick Up Drive for DVD

Reference Characteristics

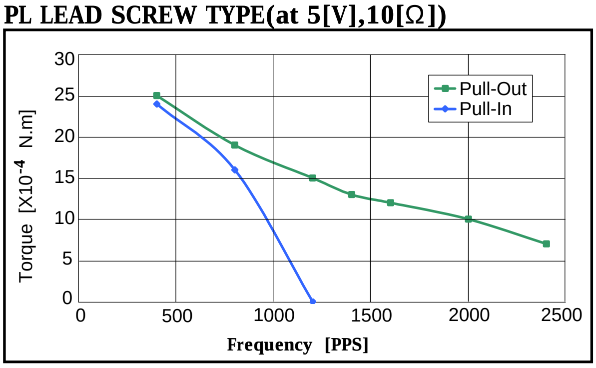
Moter Size	PL15S-020
Number of Steps per Rotation	20
Drive Method	2-2 PHASE
Drive Circuit	BIPOLAR CONST. VOLT.
Drive Voltage	5[V]
Coil Resistance/Phase	10[Ω]
Magnet Material	Nd-Fe-B bonded magnet (MS70)
Holding Torque	30[X10^{-4} N.m]
Maximum PULL-IN Frequency	1450[PPS]

Typical Torque Characteristics

Driver Frequency	400[PPS]	1400[PPS]
PULL-OUT Torque	25[X10^{-4} N.m]	13[X10^{-4} N.m]

Torque Characteristics

PL LEAD SCREW TYPE(at 5[V],10[Ω])



These torque values are reference only.
Heat radiation conditions and temperature rise effect by duty are different on each equipment, therefore please select motors after considering the heat conditions in the actual equipment.

Switching Sequence
(Viewed from Lead Screw Side)

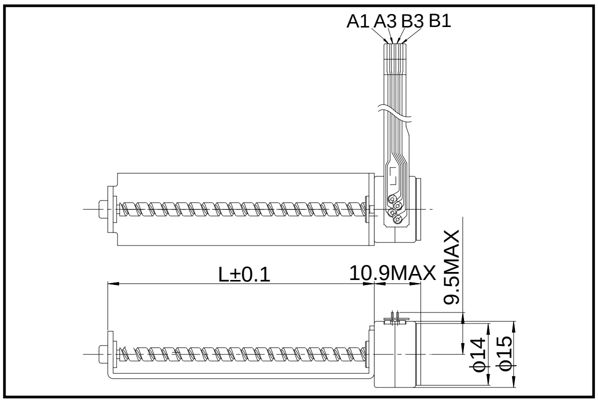
STEP \ TERMINAL	A1	A2	B1	B3
1	−	+	+	−
2	−	+	−	+
3	+	−	−	+
4	+	−	+	−

CW ↓   CCW ↑

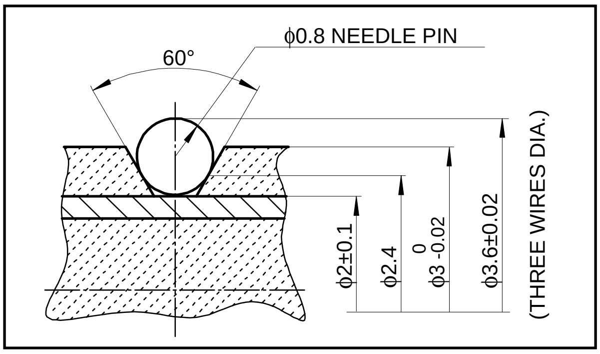
Lead Screw Spec

OUTSIDE DIA.	ϕ3
MINOR DIA.	ϕ2
PITCH DIA.	ϕ2.4
PITCH	3.0
LEAD ANGLE	21°41" 49'
LEAD DIRECTION	RIGHT-HAND SINGLE THREAD
SIMPLE PITCH ERROR	0.020
CUMULATIVE PITCH ERROR	0.030

Dimensions



Detail of Screw



Performance values shown on this website are representative.
Motor design and specifications are subject to change without notice.

1 Introduction

The Raspberry Pi Compute Module 3+ (CM3+) is a range of DDR2-SODIMM-mechanically-compatible System on Modules (SoMs) containing processor, memory, eMMC Flash (on non-Lite variants) and supporting power circuitry. These modules allow a designer to leverage the Raspberry Pi hardware and software stack in their own custom systems and form factors. In addition these modules have extra IO interfaces over and above what is available on the Raspberry Pi model A/B boards, opening up more options for the designer.

The CM3+ contains a BCM2837B0 processor (as used on the Raspberry Pi 3B+), 1Gbyte LPDDR2 RAM and eMMC Flash. The CM3+ is currently available in 4 variants, CM3+/8GB, CM3+/16GB, CM3+/32GB and CM3+ Lite, which have 8, 16 and 32 Gigabytes of eMMC Flash, or no eMMC Flash, respectively.

The CM3+ Lite product is the same as CM3+ except the eMMC Flash is not fitted, and the SD/eMMC interface pins are available for the user to connect their own SD/eMMC device.

Note that the CM3+ is electrically identical and, with the exception of higher CPU z-height, physically identical to the legacy CM3 products.

CM3+ modules require a software/firmware image dated November 2018 or newer to function correctly.

2 Features

2.1 Hardware

- Low cost
- Low power
- High availability
- High reliability
  - Tested over millions of Raspberry Pis Produced to date
  - Module IO pins have 15 micro-inch hard gold plating over 2.5 micron Nickel

2.2 Peripherals

- 48x GPIO
- 2x I2C
- 2x SPI
- 2x UART
- 2x SD/SDIO
- 1x HDMI 1.3a
- 1x USB2 HOST/OTG
- 1x DPI (Parallel RGB Display)
- 1x NAND interface (SMI)
- 1x 4-lane CSI Camera Interface (up to 1Gbps per lane)
- 1x 2-lane CSI Camera Interface (up to 1Gbps per lane)
- 1x 4-lane DSI Display Interface (up to 1Gbps per lane)
- 1x 2-lane DSI Display Interface (up to 1Gbps per lane)

2.3 Software

- ARMv8 Instruction Set
- Mature and stable Linux software stack
  - Latest Linux Kernel support
  - Many drivers upstreamed
  - Stable and well supported userland
  - Full availability of GPU functions using standard APIs

3 Block Diagram

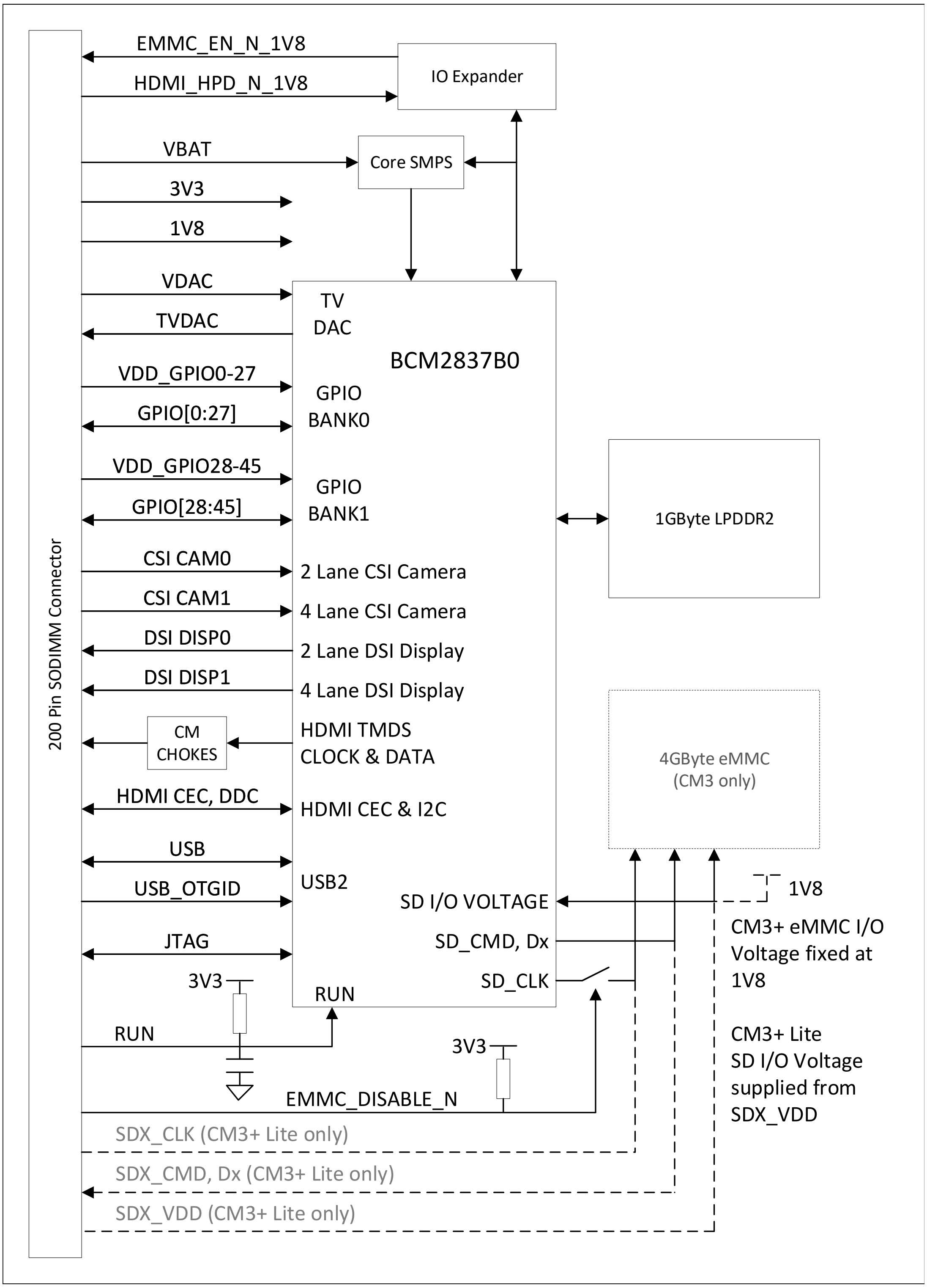


Figure 1: CM3+ Block Diagram

4 Mechanical Specification

The CM3+ modules conform to JEDEC MO-224 mechanical specification for 200 pin DDR2 (1.8V) SODIMM modules and therefore should work with the many DDR2 SODIMM sockets available on the market.(Please note that the pinout of the Compute Module is not the same as a DDR2 SODIMM module; they are not electrically compatible.)

The SODIMM form factor was chosen as a way to provide the 200 pin connections using a standard, readily available and low cost connector compatible with low cost PCB manufacture.

The maximum component height on the underside of the Compute Module is 1.2mm.

The maximum component height on the top side of the Compute Module is 2.5mm.

The Compute Module PCB thickness is 1.0mm +/- 0.1mm.

Note that the location and arrangement of components on the Compute Module may change slightly over time due to revisions for cost and manufacturing considerations; however, maximum component heights and PCB thickness will be kept as specified.

Figure 2 gives the CM3+ mechanical dimensions.

Figure 2: CM3+ Mechanical Dimensions