

An Architecture Improvement of Electronic Braille Quran Based on Raspberry Pi

Ahmad Syahrin Bin Shahrman, Shaharil bin Mad Saad, Zair Asrar Bin Ahmad, Mohamed Bin Hussein, Mohd Azuwan Bin Mat Dzahir, Mohd Azwarie Bin Mat Dzahir

Abstract: An electronic braille (eBraille) Quran is a braille display that is used to aid the visually impaired and blind Muslims in their journey of understanding and appreciating the Quran in a much more convenient way. It is an invention of University of Technology Malaysia (UTM) since 2009 for the blind people to recite Quran through the Braille system. However, current architecture of eBraille Quran comprises of too many modular components, numerous microcontrollers and complexly designed. Another vital issue is that its architecture still uses old technologies where the firmware updating or data correction process cannot be done remotely. This makes it hard to maintain, repair, costly and time consuming. This study proposed an enhancement of current architecture with the latest technology of single board computer named Raspberry Pi 3B+. The architecture is first designed according to the existing features of the eBraille Quran. Then, a prototype circuit is developed and fabricated for testing purposes of the proposed architecture. Once the circuit is fabricated, an experiment is conducted to test on how well the new architecture manages the intended functionalities, controlling input and output of braille cells, the performance of start-up time and battery consumption. The result shows that the proposed architecture using Raspberry Pi manages to carry out all the proposed functionalities of eBraille Quran. The start-up time is also within tolerable level with only 1 second slower than the current architecture. Battery consumption on the other hand shows that using Raspberry Pi reduces the battery consumption by 1.2% per hour.

Keywords – Braille System, Raspberry Pi, Single-Board Computer, eBraille Al-Quran, Architecture.

I. INTRODUCTION

The blind or visually impaired people uses braille system to communicate and connect – without the need of their eyes. A braille system is a system of raised dots that could be read with the tip of our fingers. It is used by thousands of people world-wide in their own native languages and literacy. Through advancement of technology, the Braille system is used on refreshable devices, such as the eBraille Quran – an invention of Universiti Teknologi Malaysia

Revised Manuscript Received on August 05, 2019.

Ahmad Syahrin Bin Shahrman, School of Mechanical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor Bahru, Malaysia

Shaharil bin Mad Saad, School of Mechanical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor Bahru, Malaysia

Zair Asrar Bin Ahmad, School of Mechanical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor Bahru, Malaysia

Mohamed Bin Hussein, School of Mechanical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor Bahru, Malaysia

Mohd Azuwan Bin Mat Dzahir, School of Mechanical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor Bahru, Malaysia

Mohd Azwarie Bin Mat Dzahir, Faculty of Built Environment & Surveying, Universiti Teknologi Malaysia, 81310 Skudai, Johor Bahru, Malaysia.

(UTM) for blind and visually impaired people to recite Quran [1][2][3]. The eBraille Al-Quran, as seen in Figure 1 [4], does not come without limitations that can be improved.



Figure 1: eBraille Quran [4]

The current eBraille Al-Quran comprised of too many modular components, numerous microcontrollers, extensive wiring and overall complexly designed. This makes it hard to maintain, repair, is costly and time-consuming. Another vital issue in the current architecture is that any form of firmware updating, or data correction process cannot be done remotely and had to be done manually via secure digital (SD) cards. Overall, the whole architecture can be simplified and upgraded further with the implementation of single board computer (SBC).

II. LITERATURE REVIEW

The Braille system is devised in 1824 by a Frenchman, Louis Braille, a fifteen-year-old boy in Paris, France as a sixty-three-character code that indicates the different variations of characters or alphabets by using raised dots [5]. This idea was inspired to him by a former French artillery captain, Charles Barbier who went and visited his school. Charles had shared his invention called “night” writing which is a code comprised of 12 raised dots made for the army that allows for delivery of messages in the dark [6]. As technology progresses, Braille system is no longer used on paper and instead are used by means of electronic braille displays.

Electronic braille displays are an array of braille cells which are made to be refreshable by using various actuators that drive the dots up or down [7][8][9]. This enables the visually impaired users to read a sequence of characters and even words with ease. It is also common for a braille display to be equipped with Perkin keys and text-to-speech output which grants the users extra accessibilities. One example of

the electronic braille display is the eBraille Quran – an invention of Universiti Teknologi Malaysia since 2009.

eBraille Quran is a project aimed to aid the visually impaired and blind Muslims in their journey of understanding and appreciating the Al-Quran in a much more convenient way. It is comprised of 40 piezoelectric-actuated braille cells that displays Al-Quran verses, allowing users to read and recite line after line of the Quran without relying on heavy and cumbersome printed Braille Quran books as shown in Figure 2 [1]. Although the eBraille Quran provides ease of access for the blind people to recite the Holy Quran, it can be further improved with SBC.



Figure 2: Al-Quran printed in Braille books [4]

SBC is a complete functioning computer built on a single circuit board consisting of microprocessor(s), memory, storage, input and output interface, wireless adapter and various necessary functionalities – all in a single piece of circuit board [10][11]. Although not as strong as their desktop counterparts, they are sufficient for various applications on smaller devices. They cost as low as one-fifth of a normal desktop, making it excellent tools for experimentation, media player, home automation, computing tasks and teaching purposes. Many different types of SBC are available in the market. A comparison of the available SBC in the market is as shown in Table 1.

Table 1: Comparison of available SBC

Manufacturer	Price (USD)	Description	Limitations
Raspberry Pi 3 Model 3B+ (2018) [12]	35	1.4 GHz 64-bit quad-core CPU 1GB SDRAM, build-in HDMI, wireless LAN, Gigabit Ethernet, Extended 40-pin GPIO, micro SD card slot	Power hungry
BeagleBoard-X15 (2016)[13]	239	1.5GHz dual core CPU, 2GB RAM, 157 GPIO Pins, Full HDMI, MicroSD slot, 3 USB 3.0 slots, 2Gb Ethernet.	Power hungry, expensive

UDOO X86 ULTRA (2018)[14]	267	2.56GHz quad-core CPU, 32 GB storage + microSD card slot. 3x USB 3.0 ports, Gigabit Ethernet, Wi-Fi and Bluetooth modules, Up to 20 GPIOs	Expensive and limited amount of on GPIO pins
Banana Pi W2 (2018)[15]	93	1.5GHz quad-core, 2GB RAM, MicroSD slot, SIM card slot, HDMI supported, 3x USB ports, 1x Type-C interface, 28 GPIO pins	On-board Wi-Fi and Bluetooth not supported, limited amount of GPIO pins

Table 1 shows that available SBCs in the market that has some limitations in term of power consumption, cost per unit, features and amount of GPIO pins available. In this study, the Raspberry Pi 3B+ as shown in Figure 3 is the chosen as SBC for a new architecture of eBraille. In comparison to the other available SBC, Raspberry Pi 3B+ is chosen for this project for the following reasons:

- a) Inexpensive for experimentation purposes, to implement and to replace. It is also cheaper compared to the other SBC with almost the same specifications and features.
- b) Provide more than enough features and accessibilities that is perfect for the new architecture of the eBraille Quran.
- c) Provides more solutions for communication and connectivity.
- d) Fully functional and the vendor provides open source drivers which would mean easier configuration and update.

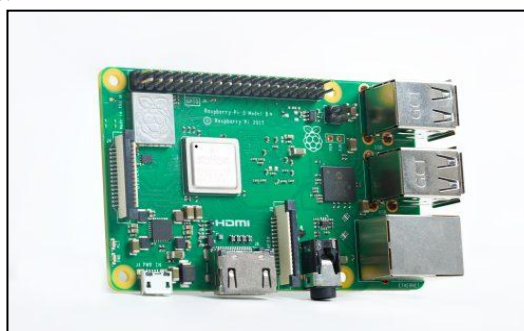


Figure 3: Raspberry Pi 3 Model B+[12]

III. METHODOLOGY

The study is started with identify the existing features that is currently in the eBraille Quran, constructing a proposed architecture based on Raspberry Pi 3B+ and fabricating a prototype circuit for functionality test of each features. The functionalities tests are including text, audio and user data, audio output, output data to shift register and input or Perkin key test. Once



everything is functioning, two tests were carried out which are performance test on start-up time as well as battery consumption. At the same time, the state of the Raspberry Pi's operating system (OS) is checked to ensure no corruption.

A. Identify Existing Features of eBraille

As the architecture to be proposed is based on the eBraille Quran, the features currently existing should be identified first as they would determine how the architecture using Raspberry Pi would be. The current existing features of the electronic Braille Quran are as follows:

- a) Menu selection: The feature that allows the selection of surah and ayah. The selection will use Perkin keys to operate.
- b) Database: SD card is used to store text, audio and user data. Text and audio is at separated SD card.
- c) Audio module: Audio propagation from speaker installed.
- d) Perkin keys: Allow user input for controlling the eBraille AI-Quran like menu selection.
- e) Braille cells: Output for the braille characters to be displayed.
- f) Power module: Lithium battery module that is rechargeable.

Based on the features identified from the current existing eBraille AI-Quran, the architecture of the eBraille AI-Quran using Raspberry Pi 3B+ is designed. The architecture proposed will mostly be incorporated inside the Raspberry Pi with minimum additional modules. The architecture will consist of the following existing features while revolving around the SBC:

- a) Input/Perkin keys: Menu selection and user control
- b) Database: All data including text and audio data is saved locally inside the Raspberry Pi.
- c) Audio module: Allows audio propagation from speaker installed.
- d) Braille cells: Output for the braille characters to be displayed.
- e) Remotely update: Data and firmware can be updated remotely.

B. Proposed Architecture Overview

The proposed architecture overview using Raspberry Pi is shown in Figure 4 and Figure 5 shows the architecture of data management, audio and input respectively. Different from the current eBraille AI-Quran architecture, most of the proposed functionalities are incorporated or embedded from within the Raspberry Pi without needing additional modules. Figure 3.2 shows the proposed architecture overview of the eBraille AI-Quran using Raspberry Pi.

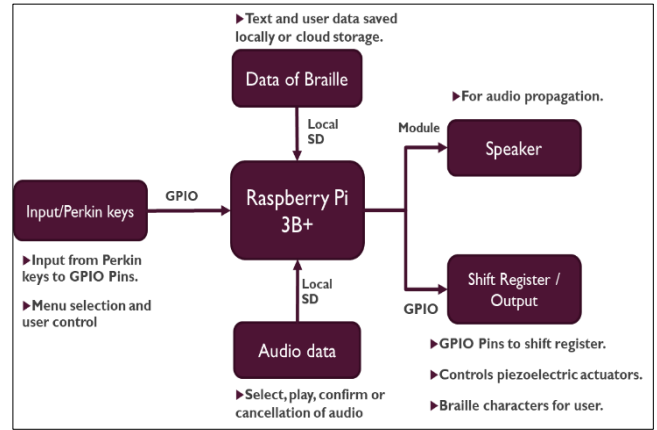


Figure 4: Proposed architecture overview using Raspberry Pi.

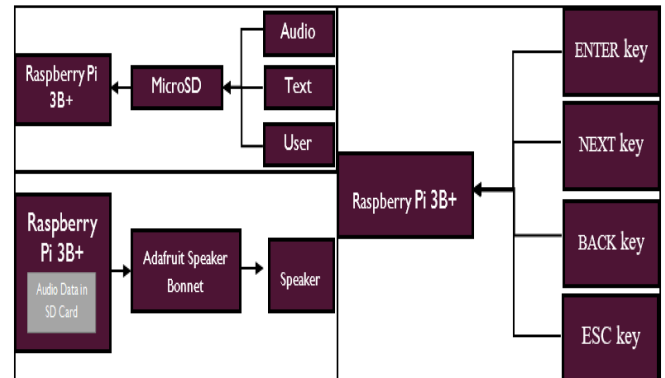
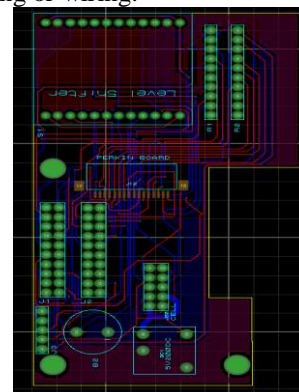


Figure 5: Architecture of database, audio and input/Perkin keys.

C. Fabrication of Prototype Circuit

Once the architecture is proposed, a prototype circuit is designed and fabricated in order to test each functionality. The prototype circuit is designed by using Proteus Design 8. It was then fabricated through outsourcing as a custom-made order to JLCPCB company. The designed and fabricated circuit is shown in Figure 6. The components involved are also connected to the fabricated circuit by means of soldering or wiring.



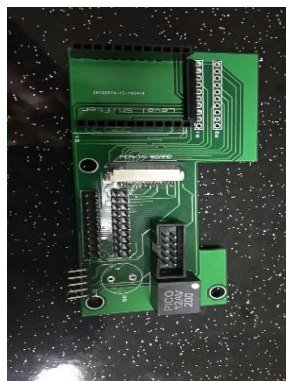


Figure 6: Prototype circuit designed and fabricated.

IV. RESULTS & DISCUSSIONS

This section will explain the result of functionality tests that has been done for each features like input/Perkins keys and data output for the braille characters to be displayed. This section also presented on how ebraille’s data with the proposed architecture can be updated remotely. Finally, the result of performance test on start-up time as well as battery consumption and Raspberry Pi’s operating system (OS) corruption are also presented in this section.

A. Input/Perkin Keys Functionality Test

The aim of this research is to develop a new functioning architecture of eBraille Quran using SBC which is Raspberry Pi. As such, it must be able to carry out the intended functionalities. Figure 8 shows the result of input/Perkin keys functionality test. The navigational keys are functioning where “ENTER” will called the next directory/menu, “ESCAPE” goes back one directory/menu, “NEXT” brings the next item in the list and “BACK” does the opposite.



Figure 7: Result of testing Input/Perkin keys test

B. Executing Program Results

Once that is done, a program is developed that allows for reading of Al-Quran from main menu, surah selection, ayah selection and the verses, as shown in Figure 9(a) to 9(d). All the while testing the navigational keys / input keys to select the available surah, ayah, or verses of Quran. The architecture using Raspberry Pi was successful.



Figure 8(a): Main menu showing “ebraille teaching solutions”.



Figure 8(c): Ayah selection “ayah 1”

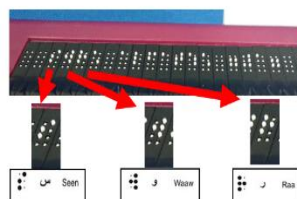


Figure 8(d): First verse of Al-Fatihah.

Figure 8(b): “Surah al-fatehah” shown

Figure 8(a) to 8(d): Result of executing program

C. Additional Feature – Virtual Network Computing (VNC)

One of the main issues of the current eBraille Quran is its inability to be updated remotely. Using the proposed architecture, an additional feature of Virtual Network Computing (VNC) is available. The VNC service uses the Remote Framebuffer (RFB) protocol to allow user to view and control a window system on another computer [4]. VNC Server and VNC Viewer by RealVNC is used to demonstrate the ability for user to connect to the Raspberry Pi remotely. A file transfer process is shown in Figure 9. The main significance of this is that it allows for easy update of data and transferring of files that can be done remotely anytime and anywhere.

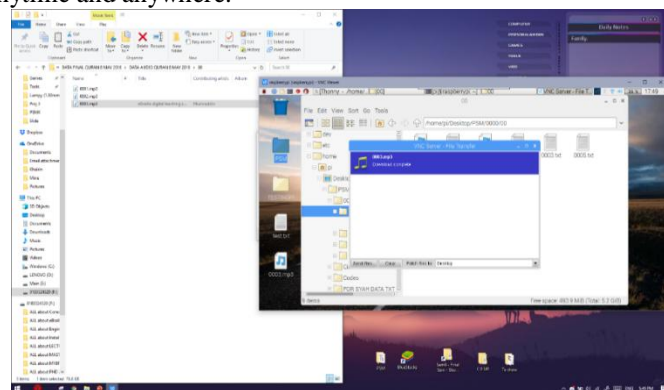


Figure 9: Transferring files process from laptop to Raspberry Pi

D. Result – Start-up Time and OS Corruption

One of the main concerns of using a SBC is the start-up time of the Raspberry Pi in comparison to the current eBraille Quran architecture. As a SBC, it comes with an OS which has a possibility that it could get corrupted if the device is not shut down properly or is turned on and off repetitively. Therefore, a test is conducted by turning both the eBraille Quran and the implemented Raspberry Pi 3B+ on and off. The start-up time is measured from when the device is switched on until the first menu text up and tabulated in Table 1.



Table 1: Start-up time and OS Corruption Test

Run	Start-up time (s)		OS Condition
	eBraille Quran	Raspberry Pi 3B+	
1	5.1	6.3	OK
2	5.0	6.1	OK
3	5.0	5.9	OK
4	5.2	5.4	OK
5	5.3	6.4	OK
6	4.9	6.0	OK
7	5.0	6.1	OK
8	5.0	6.2	OK
9	5.1	6.1	OK
10	5.2	6.0	OK
Mean	5.1	6.1	OK

It can be concluded that although the start-up time of the Raspberry Pi takes a longer time than the eBraille Quran, it is only by an average of 1 second. It is within the tolerable level that the user might not even notice. As for the OS corruption, our findings found out that the state of the OS experienced no corruption even after 50 iterations of turning it on and off repeatedly.

E. Result – Battery Consumption

Another test is conducted with the purpose of checking the battery consumption of the eBraille Quran and the Raspberry Pi. This is especially important for the eBraille Quran application as the device serves as a remote and portable device usable everywhere with battery power. The battery voltage is measured every 10 minutes for 60 minutes. This test is repeated three times at different times and dates. The results are tabulated in Table 2. The summarized result of battery consumption per hour is as shown in Table 3.

Table 2: Measurement at every 10 minutes for 1 hour for 3 repeated tests.

Run	1			2			3											
	E	-	%	R	-	%	E	-	%	R	-	%	E	-	%	R	-	%
1	4.07	0	-	3.64	0	-	5.32	0	-	6.44	0	-	4.14	0	-	5.27	0	-
2	4.02	0.05	1.23	3.58	0.06	1.65	5.26	0.06	1.13	6.37	0.07	1.09	4.1	0.04	0.97	5.2	0.07	1.33
3	3.9	0.12	2.99	3.53	0.05	1.40	5.18	0.08	1.52	6.33	0.04	0.63	4.05	0.05	1.22	5.16	0.04	0.77
4	3.84	0.06	1.54	3.5	0.03	0.85	5.14	0.04	0.77	6.29	0.04	0.63	4.02	0.03	0.74	5.11	0.05	0.97
5	3.81	0.03	0.78	3.47	0.03	0.86	5.11	0.03	0.58	6.26	0.03	0.48	3.99	0.03	0.75	5.07	0.04	0.78
6	3.78	0.03	0.79	3.45	0.02	0.58	5.08	0.03	0.59	6.21	0.05	0.80	3.94	0.05	1.23%	5.02	0.05	0.99
7	3.74	0.04	1.06	3.42	0.03	0.87	5.03	0.05	0.98	6.17	0.04	0.64	3.9	0.04	1.02	4.98	0.04	0.80
MEAN			1.40			1.03			0.93			0.71			0.99			0.94

E = eBraille Al-Quran, R = Raspberry Pi 3B+

Table 3: Summarized battery consumption over an hour.

Test	eBraille	Raspberry
1	8.11%	6.04%
2	5.45%	4.19%
3	5.80%	5.50%
AVG	6.45%	5.25%

Based on our findings, it shows that the battery consumption of Raspberry Pi is more desirable as it is

slightly better compared to the current eBraille Quran with a difference of 1.2% less battery consumption over an hour.

CONCLUSION

This study was carried out with the purpose to implement a new architecture to the existing eBraille Quran – an invention of University of Technology Malaysia (UTM) by using a Single-Board Computer. The architecture is proposed based on the existing features of the eBraille Quran. A prototype circuit has also been designed and fabricated for the purpose of testing the architecture. Several tests have been conducted like functionalities test, start-up time and battery consumption test. The results show that it can carry out all the functionalities same with current eBraille Quran architecture. The performance tests also show that proposed architecture consumes less battery level per hour and has a slightly longer start-up time compare to current architecture which is within tolerable level with only 1.4 second.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to Universiti Teknologi Malaysia (UTM) for providing funding and facilities and resources to conduct this research with Project No. 16J04.

REFERENCES

1. M. Hussein et al., "Increasing Braille Literacy : Voice-Assisted Electronic Braille Books (eBraille eBook) for the Visually Impaired 2 Overview of Assistive Devices in Braille Education," *Adv. Comput. Technol. Educ.*, pp. 72–77, 2017.
2. S. Mad Saad, M. Z. Md Zain, M. Hussein, M. S. Yaacob, A. R. Musa, and M. Y. Abdullah, "A system architecture of electronic Braille panel for reciting Al-Quran," in *Proceedings - 2nd International Conference on Computational Intelligence, Modelling and Simulation, CIMSIm 2010*, 2010, pp. 427–430.
3. S. M. Saad et al., "Development of piezoelectric braille cell control system using microcontroller unit (MCU)," *WSEAS Trans. Circuits Syst.*, vol. 9, no. 6, 2010.
4. UTM, "What is eBraille," Weebly, 2014. .
5. M. Marmor, I. Alonso, and J. Jime, "Biography of Louis Braille and Invention of the Braille Alphabet," vol. 54, no. 1, 2009.
6. Y. Bar-cohen, "Electroactive polymers for refreshable Braille displays," pp. 5–7, 2009.
7. G. Frediani, J. Busfield, and F. Carpi, "Enabling portable multiple-line refreshable Braille displays with electroactive elastomers," *Med. Eng. Phys.*, vol. 60, pp. 86–93, 2018.
8. D. Kendrick, "Freedom Scientific Focusing On Braille Part 2: A Review of the Focus Blue 40 Braille Display," pp. 2–3, 2014.
9. A. Russomanno, S. O’Modhrain, R. B. Gillespie, and M. W. M. Rodger, "Refreshing refreshable braille displays," *IEEE Trans. Haptics*, vol. 8, no. 3, pp. 287–297, 2015.
10. H. Nguyen, T. Thi, K. Loan, B. D. Mao, and E. Huh, "Low Cost Real-Time System Monitoring Using Raspberry Pi," 2015 Seventh Int. Conf. Ubiquitous Futur. Networks, pp. 857–859, 2015.
11. N. Kamal and P. Ghosal, "Three Tier Architecture for IoT Driven Health Monitoring System Using Raspberry Pi," 2018 IEEE Int. Symp. Smart Electron. Syst. (Formerly iNiS), pp. 167–170, 2018.
12. "Raspberry Pi 3 Model B." [Online]. Available: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>.
13. "Meet the Beagles: Open Source Computing." [Online]. Available: <http://beagleboard.org/>.
14. "UDOO X86 II." [Online]. Available: <https://www.udoo.org/udoo-x86/>.
15. "BPI-W2." [Online]. Available: <http://www.banana-pi.org/w2.html>.

