A Fingerprint Identification Algorithm for Integration into an Electronic Voting Machine using a Microcontroller.

Jonathan A. Enokela, Ph.D.

Department of Electrical Engineering, University of Agriculture, PMB 2373, Makurdi, Nigeria.

E-mail: jonajapeno@gmail.com

ABSTRACT

Fingerprints, each authenticated into a 64-byte format using a prescribed manufacturer's algorithm, were stored in the flash program memory of a microcontroller. The fingerprints were later captured using a fingerprint module and each fingerprint was compared byte-by–byte with those in memory for matching and identification thus enabling the microcontroller to take decisions regarding the status of the individual whose fingerprint has just been read. The decision taken is conveyed via interrupts to another microcontroller whose program controls the voting process in an Electronic Voting Machine. The use of fingerprints to identify voters in this manner not only guarantees the voters' anonymity but also ensures a one-man one-vote system.

(Keywords: fingerprint identification, microcontroller, authentication, electronic voting)

INTRODUCTION

There are many good fingerprints Software Development Kits (SDK) in the market with very good False Rejection Rate (FRR) and False Acceptance Rate (FAR) [1], [2], [3]. Many of these use PC-based authentication processes to extract and enroll templates as well as to carry out the identification and verification of fingerprints. Because these SDKs are PC-based they are unsuitable for integration into portable battery-powered Electronic Voting Machines (EVM) such as the one being considered here. Battery-powered portable fingerprint modules that can perform identification functions such as [4], [5], can be used together with microcontroller chips to perform identification of voters. The system being considered here will lead to overall cost reduction and can easily be integrated into an existing EVM.

MATERIALS AND METHODS

The configuration of the voting system being considered is depicted in Figure 1. The overall goal of the system is to prevent multiple voting. In other words, a voter using this system can vote but only once. The system can also however be used for the enrollment or storage of fingerprints, fingerprints update and identification. The storage of fingerprints becomes necessary during voters' registration phase of the election process. Fingerprint update can be done during by-elections that may result from the inability of the incumbent office holder to continue to exercise the duties of his office due to a number of reasons including recall by the constituents, resignation, or even death. The identification of voters is carried out during the election proper.

For a voting system, the fingerprints of the voters would have already been captured during the registration of voters. These fingerprints are first transferred to a database in a computer for post registration analysis that seeks to determine if there is any match between fingerprints from different registration localities. The existence of identical fingerprints in different localities implies multiple registrations by the individuals involved and appropriate penalties could be imposed on them. After the analysis the fingerprints for voters for each polling unit are transferred from the computer and stored in the fingerprint processor through a serial port (RS 232).

The fingerprint processor carries out the functions of fingerprints storage, update and identifications. The flow chart that depicts this process is given in Figure 2. This routine shows that three processes are involved: new fingerprints storage in memory, fingerprints update, and fingerprints identification. The identification process is carried out during elections and is the main routine of the program.

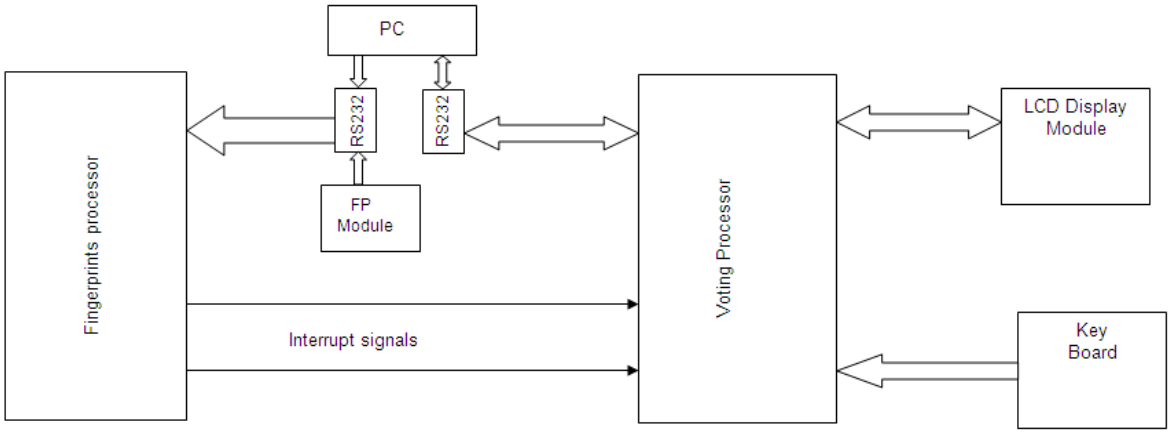

Figure 1: Configuration of Electronic Voting Machine (Note: FP is Fingerprint Module),

The fingerprint of a voter is captured by the fingerprint module (FP Module) and read through the RS 232 port as a string of 64-byte data. This fingerprint is compared with the fingerprints of all voters for the polling station in question which were stored in the program memory during voters' registration exercise. The purpose of this comparison is to ensure that a voter cannot cast a vote if he did not register. An audible alarm is emitted if the intending voter did not register. If the voter is registered for the election his fingerprint is again compared with those of all the voters that have already cast their votes. It is only when the voter has not already cast a vote that he will be allowed to vote. This second comparison process ensures that no one can vote twice in an election. It is only when it has been confirmed that a voter has not already voted that the fingerprints processor will generate an interrupt signal that will send the voting processor into the election mode.

The insecurity of network-based election systems [8], [9], [10] compels us to consider here a system that is not based on any network. Moreover the polling station is the only one channel used by the voter to cast his vote [11].

The new fingerprints storage and fingerprints update processes constitute the interrupt routines for the fingerprints processor. The first four data bytes received from the RS 232 port by this processor enables it to select among three options: If the sequence is "NEWC" the processor enters the new fingerprints storage routine that occurs after the conduct of a fresh voters' registration exercise. If this is so, it erases all formerly stored fingerprints, reads in new fingerprints and stores them in the flash memory of the processor. It also erases the voted positions fingerprints and then generates a high priority interrupt (INT1) signal for the voting processor to erase the votes scored by the candidates as well as the information about the types of election which were formerly stored in the voting processor's EEPROM locations. If the sequence received is "NEWE" it signifies that only the voted positions fingerprints and the data about candidates' scores and types of election need to be erased from memory. This is the situation we might expect during further elections in a chain of elections (for instance governorship election after a presidential election) or during a by- election in a constituency.

If the sequence received is "UPDA" the processor enters the update mode of operation. The voter's fingerprint is read and compared with those of all the registered voters to ensure that double registrations of a voter will not be done. When this has been verified the voter's fingerprint is then appended to the fingerprints memory locations.

The fingerprints identification and update processes are shown in greater details in the flow charts of Figures 3 and 4. A schematic diagram detailing the interconnection between the serial port, the fingerprints processor and interrupt generation is depicted in Figure 5.

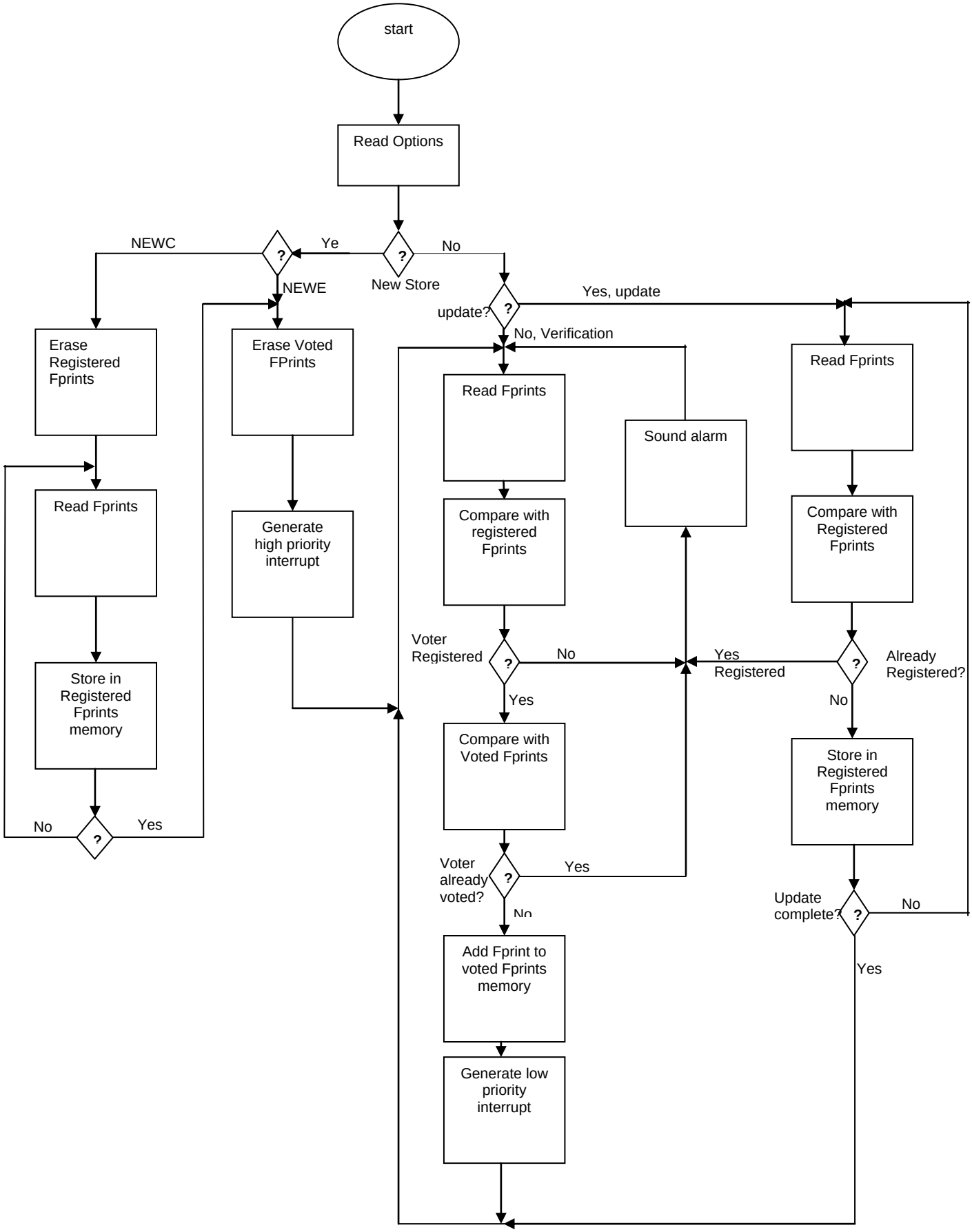

Figure 2: General flow Chart for Fingerprints Storage, Update and Identification.

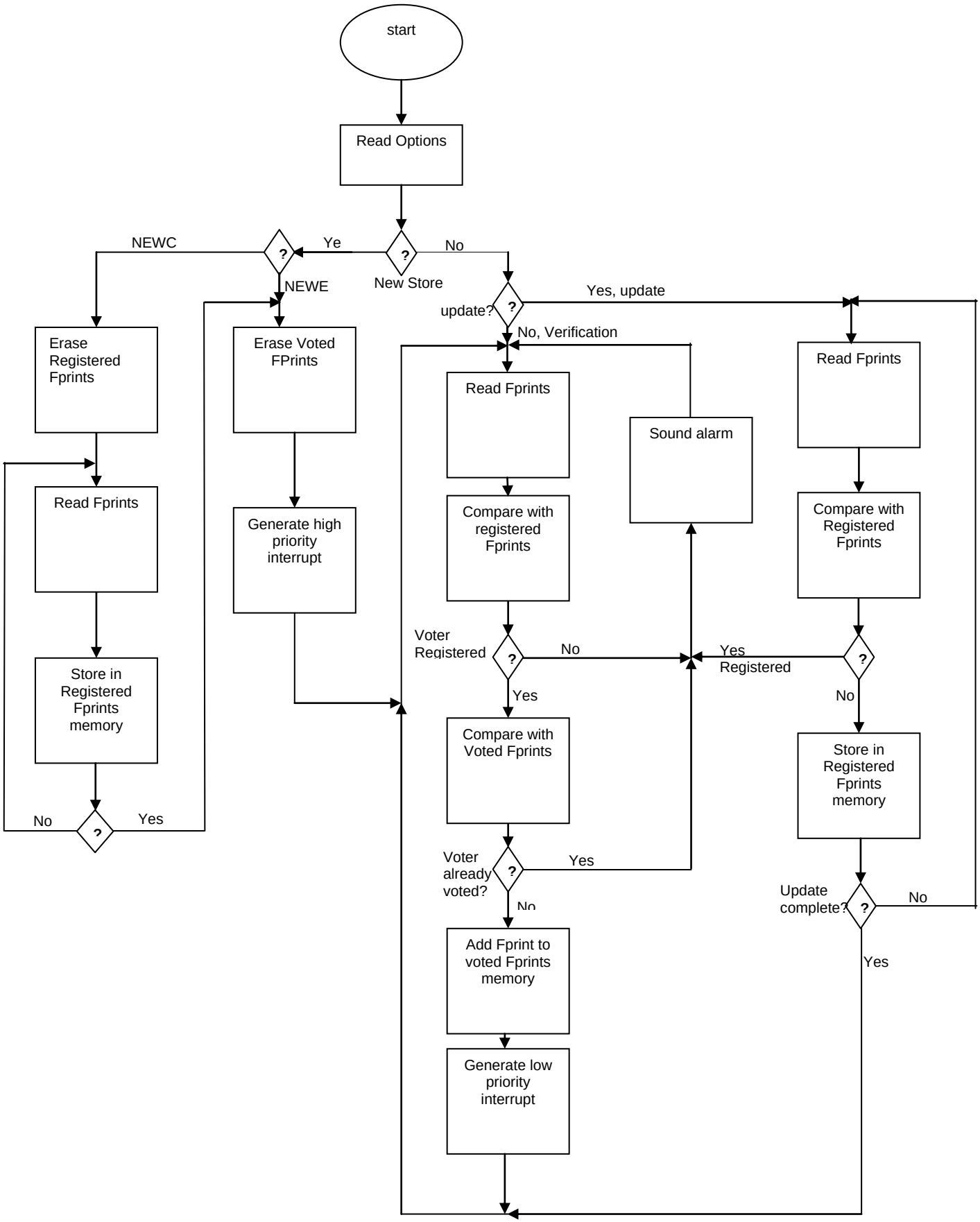

Figure 3: Detailed Flow Chart of Identification Process.

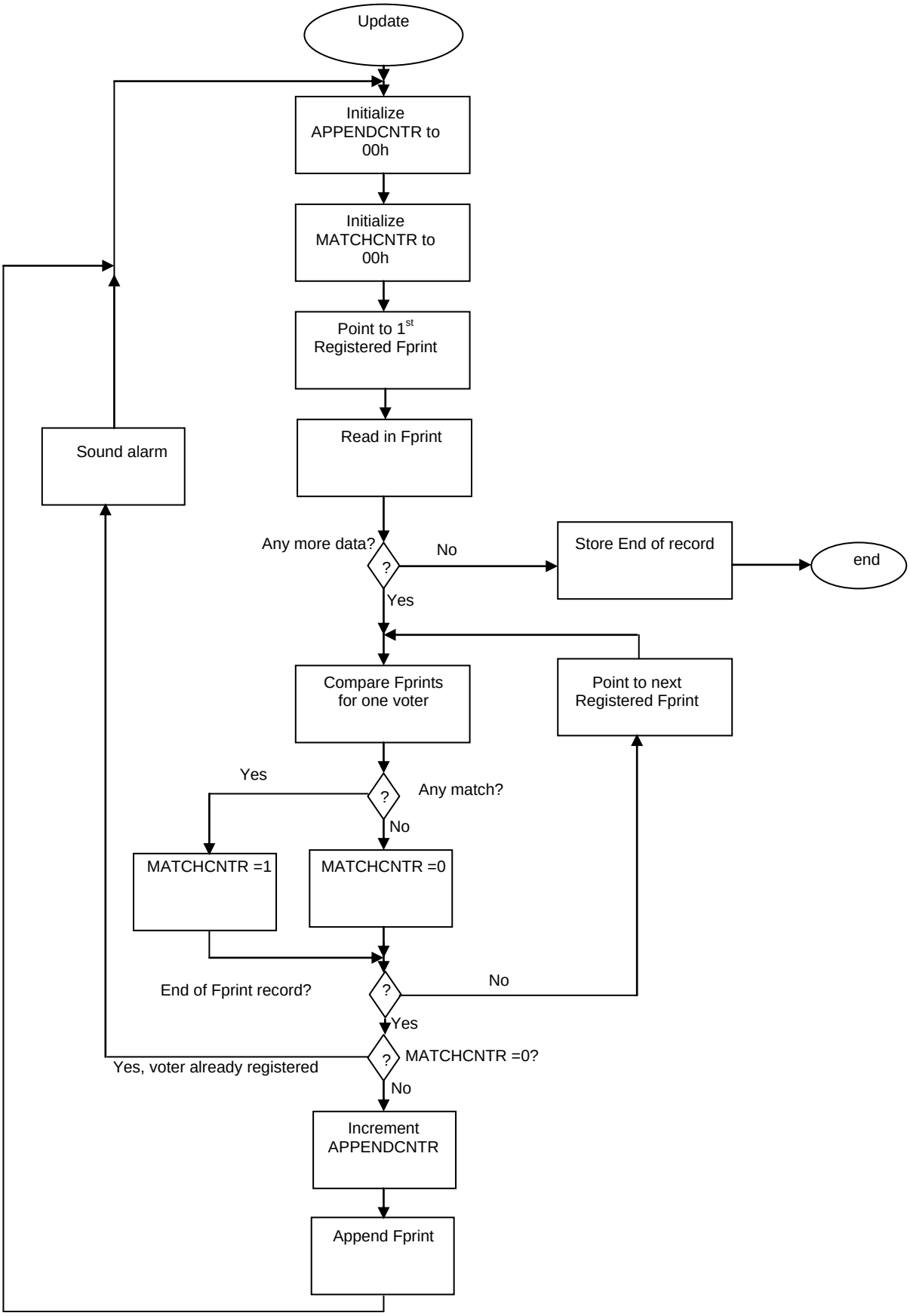

Figure 4: Detailed Flow Chart for Update Process.

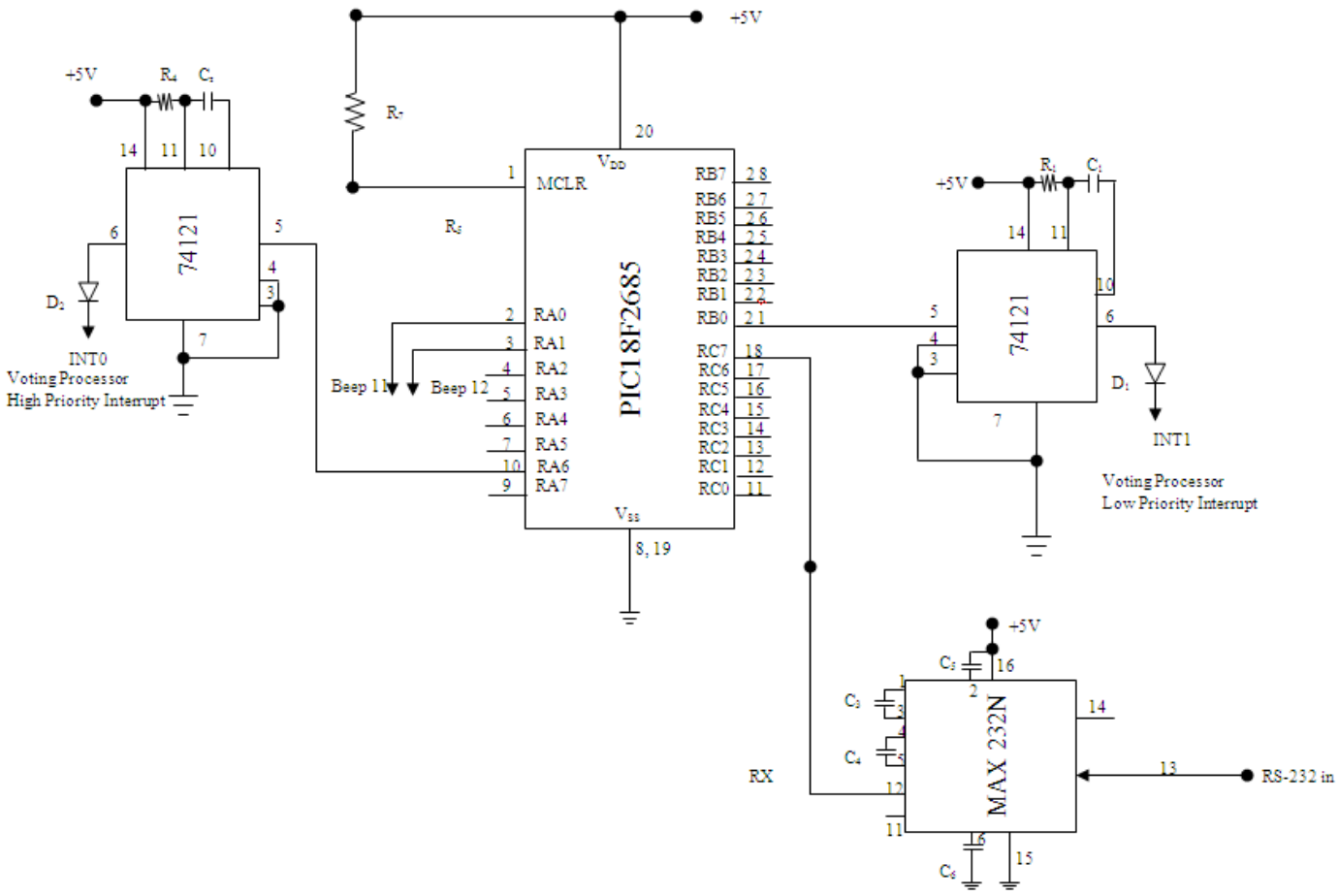

Figure 5: Schematic of Fingerprint storage, Identification, and Interrupt Control Circuit.

PROGRAM CODING, RESULTS, AND DISCUSSIONS

The flow charts were translated into Assembly Language program using the instruction set of the PIC18F2685 microcontroller [6]. The code was then assembled in the MPLAB IDE (Integrated Development Environment) version 7.50 [7]. Software simulation was carried out using MPLAB SIM (embedded in the MPLAB IDE) to verify the accuracy of the code and to debug the program. It was assumed that the fingerprint authentication algorithm to be used with the system encrypted a fingerprint into a 64 -byte data [4].

Various 64-byte data representing fingerprints were stored in the program memory of the microcontroller. The update and identification processes were observed to be correct. The microcontroller was then inserted into the circuit of Figure 5 and the In-circuit Debugger MPLAB ICD2 [8] was used to verify and observe the correctness of the code that the microcontroller executes in the actual circuit. Figures 6 and 7 are screen shots showing sections of the flash program memory of the microcontroller.

CONCLUSION

A PIC microcontroller was used to store fingerprints data in its flash program memory. A code was developed that could be used for the identification of fingerprints. These processes generated interrupts that controlled the operations of another microcontroller that was dedicated to the processing of votes. Although a 64-byte authentication algorithm has been assumed, algorithms that authenticate fingerprints into larger data bytes can also be used.

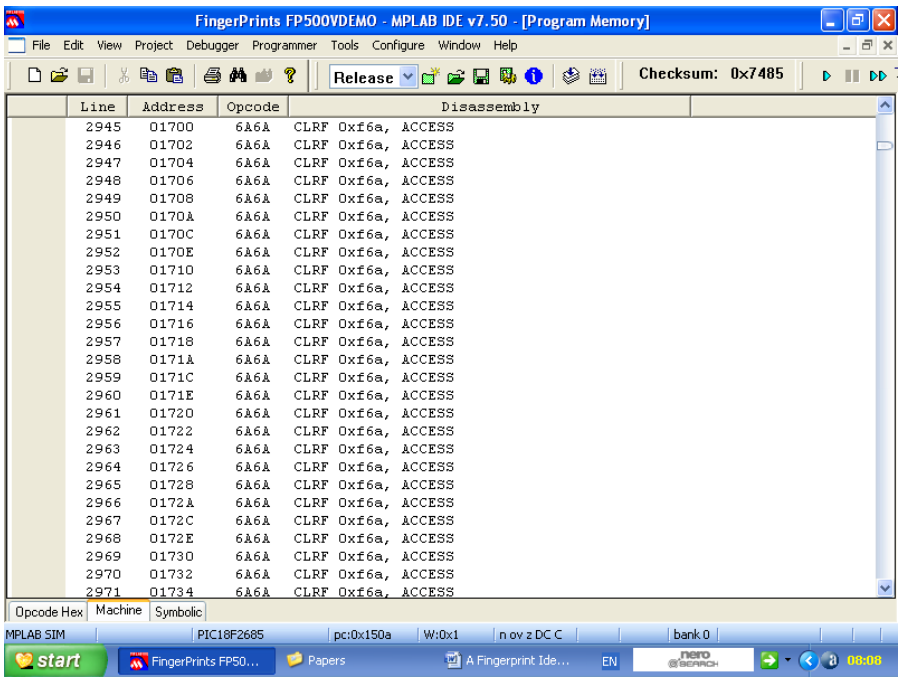

Figure 6: Arbitrary Data Used to Represent Fingerprints During Simulation.

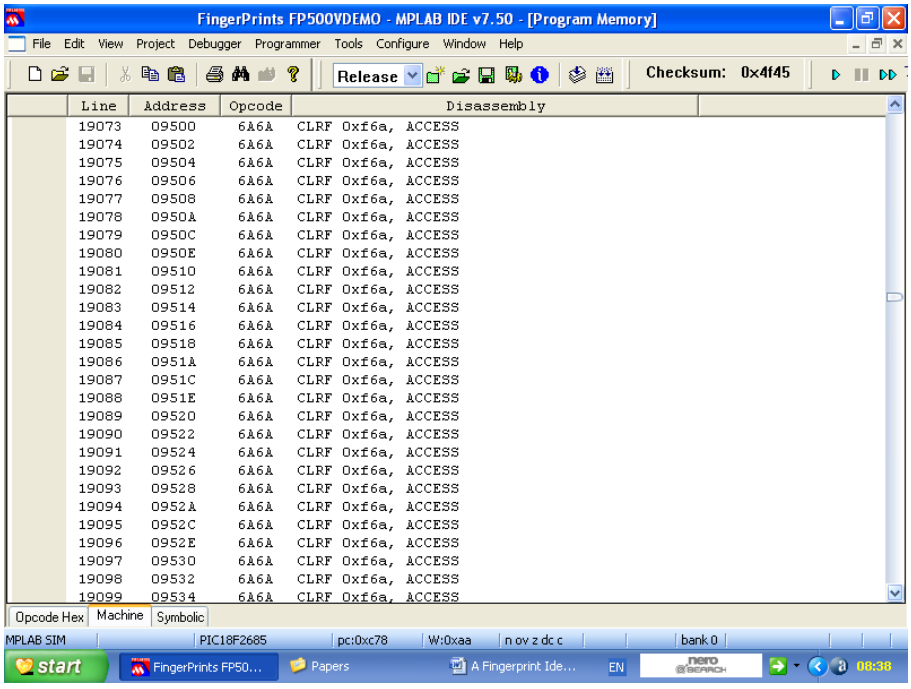

Figure 7: Fingerprint Verified and Stored in Another Memory Location.

This would, however, require a microcontroller with a larger program memory to accommodate the same number of voters. The use of biometrics data such as fingerprints to identify voters will enforce the principle of one-man one-vote since, unlike Personal Identification Numbers (PIN), biometrics characteristics are unique to individuals and are thus neither transferrable nor buyable. Following similar thinking an Automated Fingerprint Identification System (AFIS) was used to identify voters in the Venezuelan presidential elections in 2006 [12].

The Independent National Election Commission of Nigeria (INEC) employed a fingerprint identification system known as the Direct Data Capture machine (DDC) to register voters for the 2011 general elections that were held in April 2011[13]

REFERENCES

1. Griaule Biometrics LTDA. 2007. *Fingerprint SDK 2007 Developer's Manual.* http://www.griaulebiometrics.com

2. Nitgen Co., Ltd.. 2008. "Nitgen Biometrics Solutions". http://www.nitgen.com

3. Beyond LSI, Inc. 2005. "Fingerprint Technology". August, 2005. http://www.beyondlsi.com.

4. Nitgen Co., Ltd. 2008. "Nitgen FIM Module SDK". http:// www.nitgen.com

5. Microchip Technology, Inc. 2004. "PIC18F2585/4585/2680/4680 Data sheet". http://ww1.microchip.com/downloads/en/DeviceDoc/39625C.pdf

6. Microchip Technology, Inc. 2006. "MPLAB IDE User's Guide". Aug., 2006. http://ww1.microchip.com/downloads/en/DeviceDoc/51519B.pdf

7. Microchip Technology, Inc. 2007. "MPLAB ICD 2 User's Guide", Nov., 2007. http://ww1.microchip.com/downloads/en/DeviceDoc/51331C.pdf

8. Jefferson, D., Rubin, A.D., Simon, B., and Wagner, D. 2004. "A Security Analysis of the Secure Electronic Registration and Voting Experiment (SERVE)", Jan. 2004. http://www.servesecurityreport.org/paper.pdf

9. Neumann, P., Mercuri, R. and Weinstein, L. "Internet and Electronic Voting". *The Risks Digest*. ACM Committee on Computers and Public Policy. Dec., 2000. 21(14). http://catless.ncl.ac.uk/Risks/21.14.html

10. Fischer, E.A. 2003. "Election Reform and Electronic Voting Systems (DREs): Analysis of Security Issues". Congressional Research Service Report for Congress. http://epic.org/privacy/voting/crsreport.pdf

11. Xenakis, A. and Macintosh, A. 2004. "Procedural Security Analysis of Electronic Voting". Presented at ICEC'04 Sixth International Conference on Electronic Commerce. March, 2004. 541-546.

12. The Carter Center. 2007. "Developing a Methodology for Observing Electronic Voting". Oct. 2007. http://www.cartercenter.org.

13. Kayode, R.I. 2009. "Voters Registration Exercise on Course". http://www.inecnigeria.org/newsview.php?newsid=534.

ABOUT THE AUTHOR

Engr. Dr. Jonathan A. Enokela, is a Lecturer in the Department of Electrical/Electronic Engineering at the Federal University of Agriculture, Makurdi, Nigeria. He has taught various aspects of analog and digital systems design to both the undergraduate and postgraduate students. He is a registered engineer with the Council for Regulation of Engineering in Nigeria (COREN) and has a wide range of practical experiences. His research interests include embedded systems design and applications.

SUGGESTED CITATION

Enokela, J.A. 2012. "A Fingerprint Identification Algorithm for Integration into an Electronic Voting Machine Using a Microcontroller". *Pacific Journal of Science and Technology*. 13(1):292-299.

Pacific Journal of Science and Technology