International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 4– No.4,October 2012 – www.ijais.org

Design and Implementation of an Open Ended Automated Vault Security System

Saunak Mitra Sarit Hati Diganta Sengupta Indraneel Datta Pratim Sinha Roy Soumyanil Banerjee

Future Institute of Engineering and Management
Sonarpur Station Road, Kolkata - 700150,
West Bengal, India

ABSTRACT

It has always been in human nature that he tends to keep his prized possession in a secured place, a place where he thinks it would be safe from the rest of the world. The technical knowledge of man has increased exponentially with time leading to breakthroughs in modern security system. As every domain has witnessed pros and cons, hence the modern technology has also given rise to unethical developments in the arena of technically advanced security systems. The proposed system in this paper has been developed keeping in mind the vulnerabilities projected by the said technological advancement. A complex set of logic enhanced by competing software has been incorporated to design the work in concern. The said system has been termed as Open – Ended as the designated route to the vault is open for all to see, the trusted user as well as the trespassers. The beauty of the work is that even after knowing all the route combinations, only the authorized personnel can gain access to the vault. The novelty of the said work lies in the fact even if the authorized user is compromised by any means; he is bound to be trapped in the mesh structure surrounding the vault. The actual route to the vault is automatically generated by the software in a random manner, hence making it completely unpredictable for anyone, the creator of the system also, to predict the next accessible route to the vault through the mesh.

General Terms

Logic Based Coding, Automated Vault Security System, Intrusion Analysis.

Keywords

Automated Vault Security System, Security System, Logic based Security System, Open Ended Security System.

1. INTRODUCTION

Technology has aided in developing highly encrypted logic for designing security systems raising the implementation values exponentially. Security systems that embed strong logic and implementation criteria demand better expenses. The proposed system is a tradeoff between expensive sophisticated systems and low-cost lesser secured systems in that the system provides utmost security [4] with comparable lower installation and processing costs.

Keeping these in mind, numerous sets of logic have been developed, and only upon satisfying all logics, the door of the SAFE VAULT would be opened [1]. The logic sets proceed as follows:

- There would be a series of doors before the main vault. A person needs to select the correct sequence of doors among a large number of doors to be safely allowed into the vault room.
- Since it is an open–ended system, all the sequences of doors, which lead to the vault room, would be shown outside the secure complex for the eyes of either the trespasser or the authorized personal.
- If the person arriving at the entrance of the secured mesh is authorized, once he validates his identity through biometric sensors [2][5].
- If the person entering the room is authorized but is held at gun-point, there would be an additional security measure to detect it. On the opening door of the mesh itself, there would be another security device, where the personal is required to enter a secret password. If the password is entered in the reverse manner, then the system would automatically trigger a silent alarm.
- An instant message (SMS) would be sent to the cellphone of the authorized person giving him the direction of the path to be covered during the transition to the vault. This sms would be sent irrespective whether he has been held at gunpoint or not, such that the suspicion of the dacoit should not be aroused.
- The correct path changes randomly every time a person validates himself at the biometric sensors.
- Once the employee enters a correct door, the particular door would remain open always until and unless he moves out of the secured mesh structure completely.
- If a the person is held at gunpoint, he would be allowed to find his way into the vault since by entering the password in reverse order, he has already triggered the silent alarm, and security would already be on their way to catch the dacoit.
- In the process of selecting the door, if a person choses any wrong door, all the doors would be locked and an alarm would be triggered and a new random pattern of doors would now be set as the default way to enter. This random path would be

only accessible to the system administrator who, once enters his password, would be able to view it.

- Finally when the person enters the vault room, he has to enter the secret passcode to gain access to the valuables.

Using all this techniques together, a large complex code needs to be satisfied to enter the particular vault. Any break in any of the steps, would lead to automatic triggering of the alarm and the person would be trapped. Also if a person skips any of the steps, the vault won't open even if he reaches the vault room since he has not completed the sequence in the particular order. Thus, it is by far a more reliable and secure security system.

2. VAULT DESIGN

This is the pictorial representation of the Secure System. As shown in figure 1, it contains a series of doors from Door 1(D1) to Door 10(D10). The reason behind this particular design is that whenever a person enters any particular door, he has two doors as options to traverse into the vault room, the vault room (marked by 'V') being the one behind door 10 (D10) shown with a rectangular box. In this element of choosing 1 of 2 doors, if a person chooses a wrong one, an alarm is triggered, all the doors are closed and he would be automatically trapped inside the secure location.

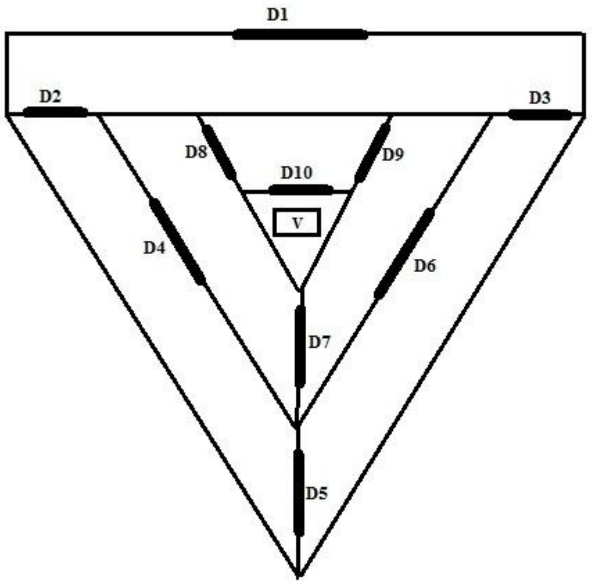

Fig 1: The Vault Design Scheme

3. OVERALL FLOW LOGIC

The entire vault security system works on the flow diagram shown in figure 2. A computer is used here as the main controller which generates the random path every 6 hours. This has been implemented using C programming. The path code so generated from the C program would be then sent serially through the serial port to the path selector board. Sensors are placed at every door to get us the status of any person entering the door or not. This data is fed to the path detector also. The path selector decrypts the serial data received from the controller and the data received from the door sensors and based on the logic built inside it; it sends open/close door signals to the respective gate.

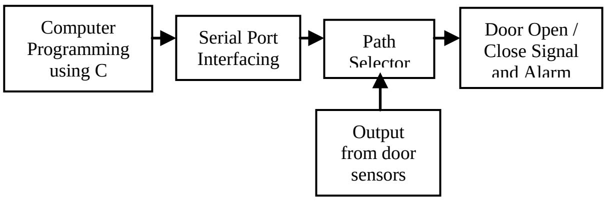
Computer Programming using C
Serial Port Interfacing
Path Selector
Door Open / Close Signal and Alarm
Output from door sensors

Fig 2: The Overall Flow Logic Diagram

3.1 Controller Flow Logic

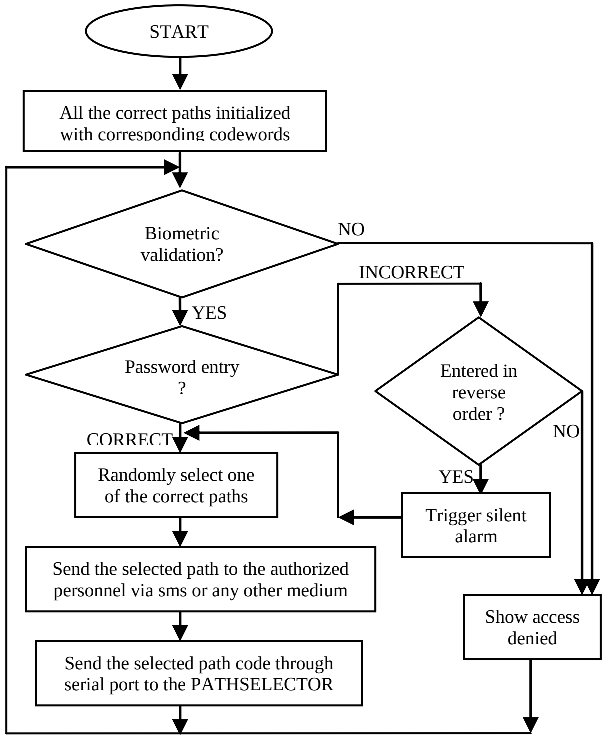
START
All the correct paths initialized with corresponding codewords
Biometric validation?
NO
YES
Password entry ?
INCORRECT
Entered in reverse order ?
NO
CORRECT
YES
Randomly select one of the correct paths
Trigger silent alarm
Send the selected path to the authorized personnel via sms or any other medium
Show access denied
Send the selected path code through serial port to the PATHSELECTOR

Fig 3: Logic to be followed in the C Program

The C program running inside the controller would be working mainly on the following logic depicted in figure 3. The program would only be activated if an authorized personal is validated using biometric identification techniques. Once done, the controller would generate a random path among the correct paths available to traverse to the vault room. This selected path route is now sent via short message system (SMS) to the authorized person trying to access the vault once his biometric identification is verified. Simultaneously the controller also sends the corresponding path

3.2 Path Code using Logic Based Coding

The Path Generator logically decrypts the data sent serially through the serial port using binary logic. The path code used for selecting the path is generated by the controller (i.e. the computer) itself. One effective way of encrypting the path code is by using Logic Based Coding. In this scheme of coding, the entire logic is transformed into a binary tree. The leftmost bit of each branch of any particular level of the tree is assigned a binary 0 value while the rightmost branch is assigned a binary 1 [3]. Using this scheme, instead of sending numerous bits at a time, one can send few bits of data, but containing same information. In figure 4, let us assume that the correct path is Door1-Door2-Door5-Door6-Door7-Door8-Door 10. If Logic Based Coding Scheme is used, the bits transferred would be 011011. Thus, only 6 bits of data are required to tell carry the entire information using Logic Based Coding Scheme.

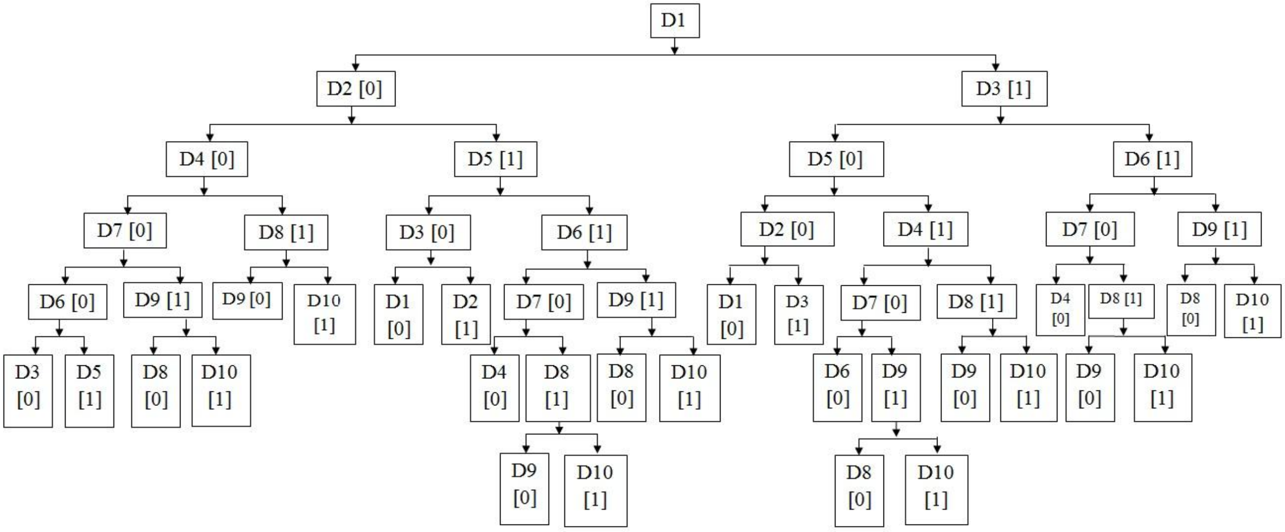

Fig 4: The Binary Logic Tree Using Logic Based Coding

3.3 Path Code Using Binary Logic

Figure 5 depicts the paths which a person is likely to take once he enters the vault room. Using simple binary logic codes, we give a binary 1 for a correct door and a binary 0 for a wrong door selection. Similarly, if the correct door pattern is Door1-Door2-Door5-Door6-Door7-Door8-Door 10, then by the binary logic, the path code sent by the C compiler would be 1100111101, where each binary 1 signifies the door which should be opened according to the correct door pattern.

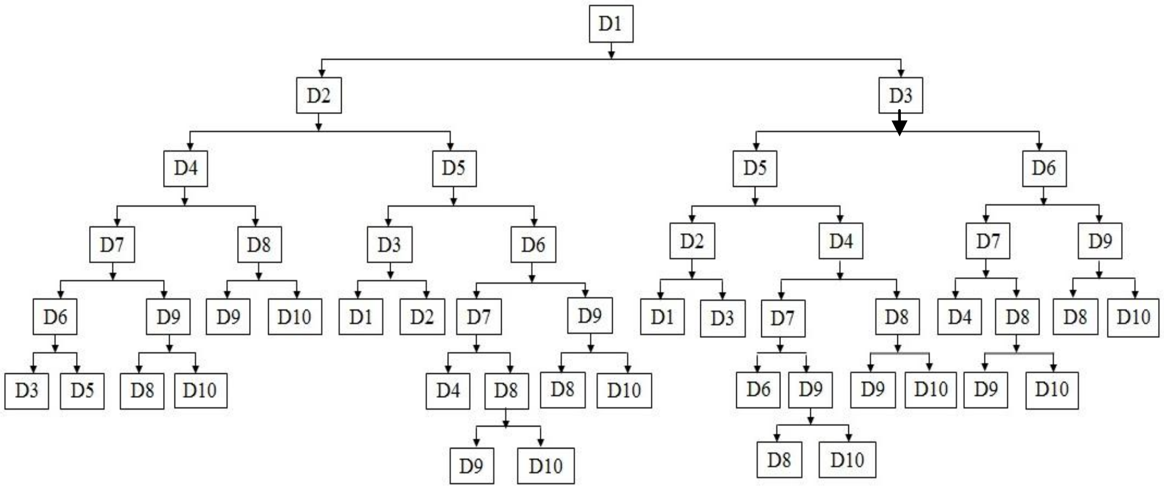

Fig 5: Path Code Using Binary Logic

3.4 Advantages of 3.3 Over 3.2 in the Prescribed Model

Though in real time, Logic Based Coding is far superior to simple binary logic, but in the prescribed model, we intend to keep the cost as low as possible without compromising any security features. Had Logic Based Coding been used, there need to be a decoder for it before the path selector, which would increase the cost. If 10 data bits are transmitted in this case, instead of 6 data bits in the case of Logic Based Coding, then there would be no significant loss, instead, the delay time used for decoding scheme would be saved. Hence, for path coding, all the 10bits of data carrying the individual status of each door are sent serially.

4. SERIAL COMMUNICATION

In figure 6, the MAX 232 is acting as a receiver from the PC and transmitter to the Path selector simultaneously. The C programming which is done in the PC has to be sent to the path selector through this serial port communicator .In this case, as the MAX232 is receiving signal from the PC, so the pin no.3 of the socket will be activated for communication and is connected to the pin no.13 (R_{in}) of MAX232 and pin no.5 of the socket is grounded. The pin no.2 of the socket here does not have any function, as the mode is receiving type. The pin no.16 of MAX232 is supply the $+V_{cc}$ of 5V and the pin no.15 acts as the ground terminal. The data received by pin no. 13 of MAX232 is taken out or transmitted out from the pin no.12.i.e, the Rx pin of the Path selector. As the communication is complete, data can be sent accordingly.

5. IMPLEMENTATION, DESIGN AND WORKING

Once the path is selected by the administrator, the corresponding path code is sent serially from the control room to the block known as PATH SELECTOR as shown in Figure 7 / Figure 8. The path selector is so designed that it can easily distinguish the wrong doors from the correct ones by deciphering the 10 bit serial data received from the control room. If any wrong door is stepped upon by the person trying to access the vault, the path detector would detect it automatically, and would trigger an alarm.

In Figure 7 / Figure 8, we can see an array of T-Flip-flops and an array of AND Gates. Also we find a module as PATH SELECTOR and PC PROGRAMMING.

The controller (i.e. the computer) runs a C program continuously in its background. The main function of this program is to verify from the biometric sensor, whether the person trying to access the vault is authenticated or not, and if so, it would generate a randomly selected path from among its correct path directory and send the corresponding path direction to the authorized personal via text message on his cell phone. Also correspondingly, the controller would send the corresponding path code to the PATH SELECTOR via its serial port.

The data from PC is now sent serially into the PATH SELECTOR. In Figure 7 / Figure 8, we see the internal Architecture of Path Selector. It consists of a 10 bit Serial in Parallel out (SIPO) Shift Register and an array of NOT Gates.

Let us assume that the bit pattern sent from PC Serially is 1101001011, where each bit corresponds to the Gate which should be opened or remained closed for the particular path entered by the administrator. The above code means that Doors No. 1,2,4,7,9,10 should be open and rest should remain closed for that particular path. This serial output from PC is then being converted into parallel data by the 10 bit SIPO Shift Register. Now, from here, this data is being divided into two sequences: one, for the correct path selector and the other for wrong path selector. The data from SIPO is connected directly to the Correct Path Selector and it is fed from a series of NOT Gates into the wrong path selector. Thus the sequence obtained in wrong path selector would be just the opposite of the one obtained in Correct Path Selector. Thus for the above sequence we get 1101001011 in the Correct Path Selector and 0010110100 in the Wrong Path Selector. These two are then fed into arrays of AND Gates.

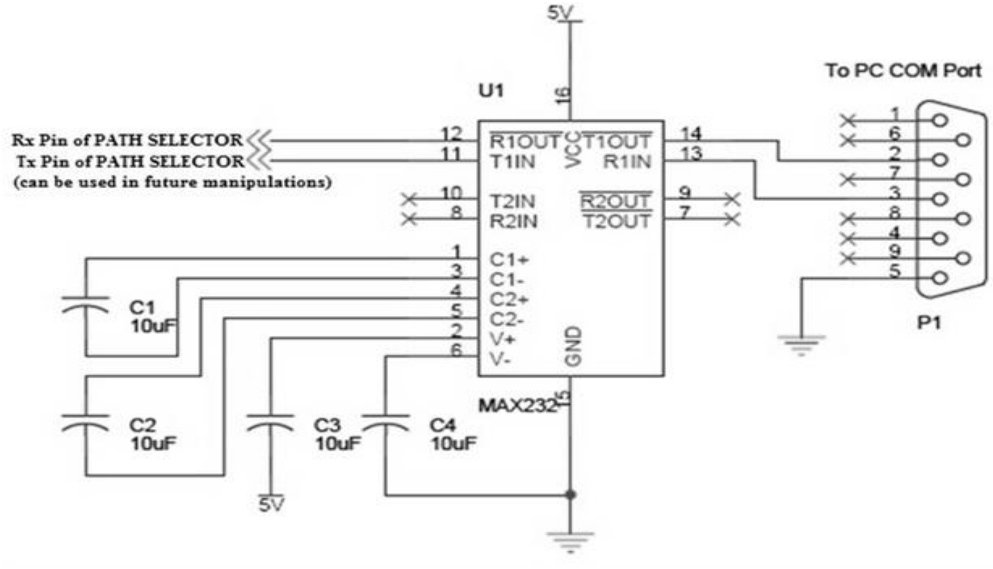

Fig 6: Serial Communication from PC using MAX232

Considering Correct Path Sequence Selector, the correct door sequence is D1-D2-D4-D7-D9-D10. Now when the person is near Door 1, the output from Door Sensor #1 is HIGH and is fed to the AND Gate. The Second input to the AND Gate is from the 1st bit from the sequence i.e. 1. So the output of the AND Gate is HIGH which is fed to the Clock of the Respective T- Flip-Flop. Initially T=1 (from the 5V Supply) and Q_n=0. Thus, on a clock triggering, Q_n=1 and the output is the Open/Close Door Signal of door #1. Now, let as assume that the person has chosen Door D3 instead of Door D2. The Output from 3rd Bit of Wrong Door Sequence Selector is 1 whereas the output from 3rd Bit of Correct Door Sequence Selector is 0. Thus the 3rd AND Gate of Correct Sequence Selector is OFF. Now since the person has entered the 3rd Gate, the output from Door Sensor #3 is HIGH. AND Gate #3 at Correct Sequence Selector Side is OFF already so signal won't pass through the AND Gate. But in Wrong Door Sequence Selector, the AND Gate would be turned ON from the Door Sensor #3 HIGH SIGNAL. This signal then passes through an OR Gate which gives a base current sufficient enough to turn ON the Transistor.

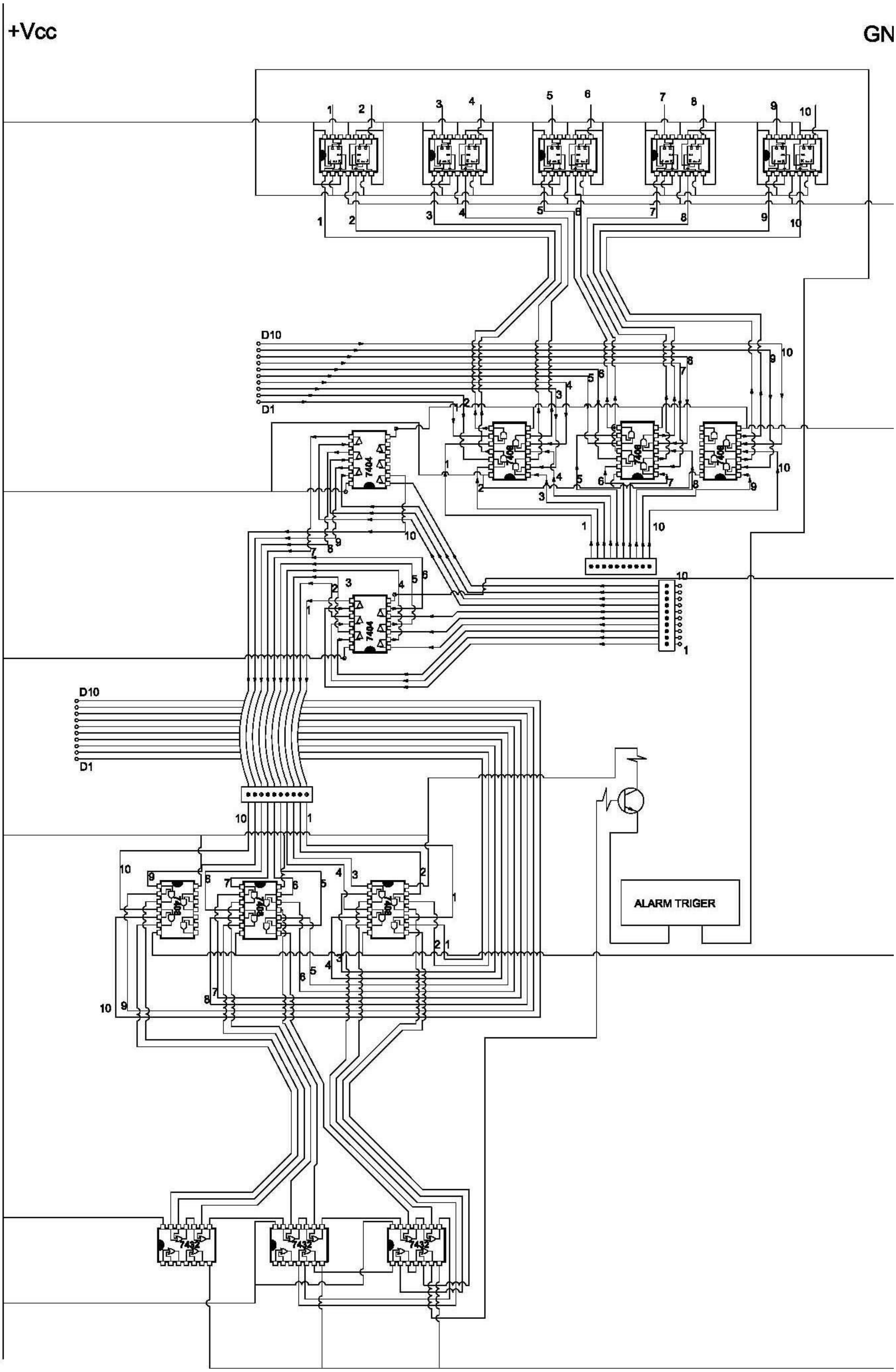

Fig 7: Implemented Pin Diagram of the Hardware Logic

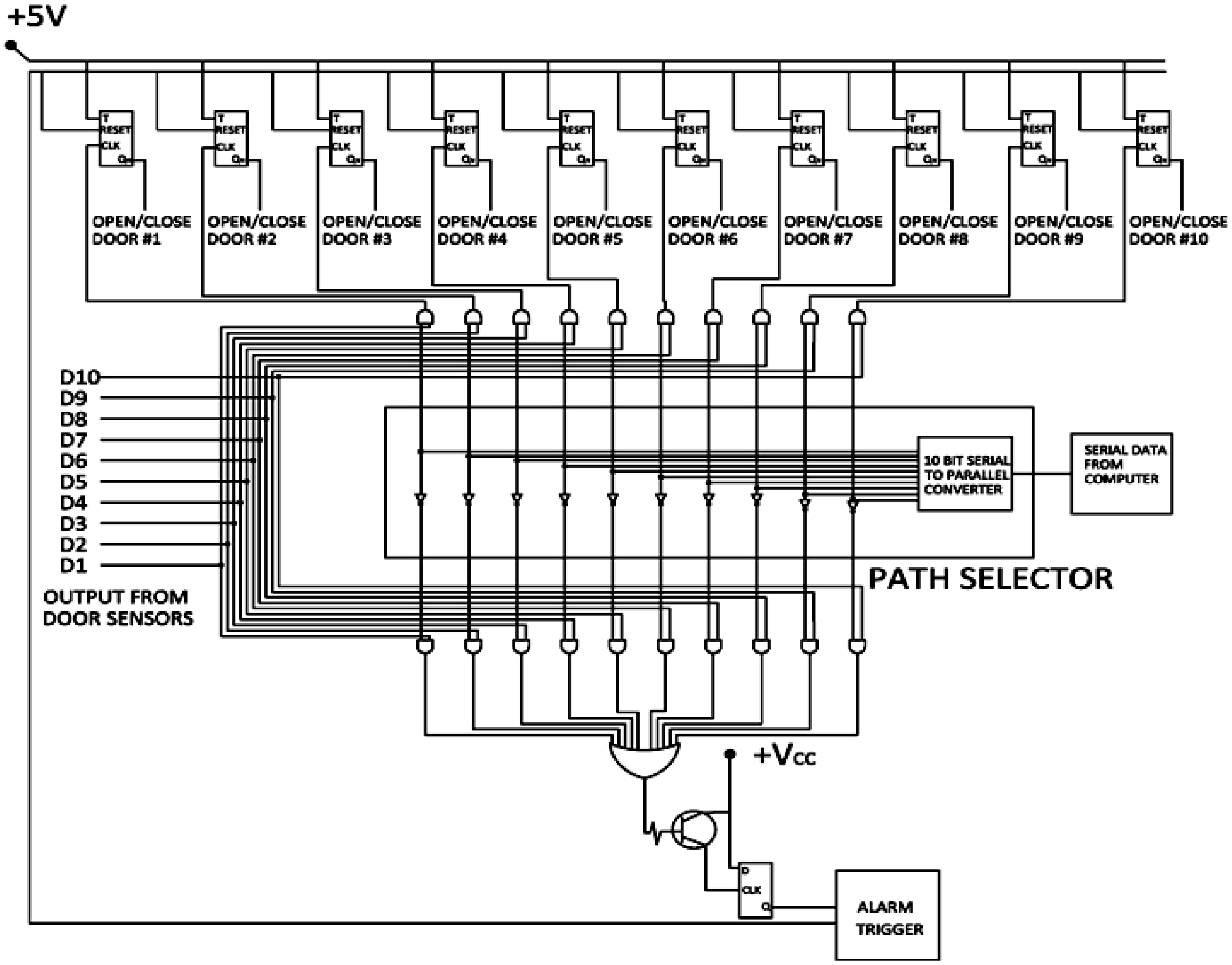

Fig 8: The Complete Design Logic of the Hardware

Thus, current starts flowing from emitter to collector in the transistor. As a result, an alarm is triggered as soon as the person chooses a wrong door. Now this alarm signal is connected to the RESET Pins of each T-Flip-Flops associated with each door. Thus when an alarm signal occurs, the Flip-flops are reset and Qn becomes LOW again and all doors are closed, meaning the person is trapped inside as soon as he chooses the wrong door.

6. CONCLUSION

The Open Ended Automated Vault Security System provides for a highly secured and featured technical system which facilitates the platform on which the security of the entire mesh can further be increased if the following things are implemented in it; Temperature and Pressure Sensors can be added to maintain a constant temperature and pressure when there is no authorized personal inside, Also when an alarm is triggered, a previously recorded message can be sent automatically to the authorities asking for immediate backup. In the present form itself the system values a highly secured vault.

7. ACKNOWLEDGEMENT

We would like to thank our entire faculty members of Applied Electronics and Instrumentation Department, Future Institute of Engineering and Management for helping us thoroughly in this endeavor.

8. REFERENCES

[1] K.NandaKumar, “Multibiometric Systems: Fusion Strategies and Template Security”, PhD Thesis, Department of Computer Science and Engineering, Michigan State University, January 2008.

[2] Ratha, N.K., J.H. Connell, and R.M. Bolle, “Enhancing security and privacy in biometrics-based authentication systems”, IBM Systems Journal, vol. 40, no. 3.Brown, L. D., Hua, H., and Gao, C. 2003.

[3] Jagadish H. Pujar, Lohit M. Kadlashkar, “A New Lossless Method of Image Compression and Decomposition using Huffman Coding”, Journal of Theoretical and Applied Information Technology

[4] Walter J. Scheirer and Terrance E. Boult, “Cracking Fuzzy Vaults and Biometric Encryption”, Securics Inc. and University of Colorado at Colorado Springs

[5] Manvjeet Kaur, Dr. Sanjeev Sofat, Deepak Saraswat, “Template and Database Security in Biometrics System : A Challenging Task”, International Journal of Computer Applications, number 5, article 1, 2010.