International Journal of
Research in
Advanced Electronics Engineering

E-ISSN: 2708-4566
P-ISSN: 2708-4558
IJRAEE 2022; 3(1): 01-05
© 2022 IJRAEE
www.electrojournal.com
Received: 04-11-2021
Accepted: 05-12-2021

US Akor
Department of Electrical and Electronics Engineering, Kaduna Polytechnic, Kaduna, Nigeria

Peter Makanjuola O
Department of Electrical and Electronics Engineering, Federal Polytechnic, Nekede, Nigeria

M Hamza
Department of Electrical and Electronics Engineering, Kaduna Polytechnic, Kaduna, Nigeria

Mustapha M Kabir
Department of Electrical and Electronics Engineering, Kaduna Polytechnic, Kaduna, Nigeria

Correspondence
US Akor
Department of Electrical and Electronics Engineering, Kaduna Polytechnic, Kaduna, Nigeria

Design and implementation of a combinational lock state system using VHDL

US Akor, Peter Makanjuola O, M Hamza and Mustapha M Kabir

Abstract

A combinational lock access control system is a critical link in a security chain. The paper focuses on the implementation of a coded keyless lock. To gain access to the lock, you must first enter a binary code combination. It will unlock the lock if the correct sequence is entered; otherwise, it will remain locked. The ALTERA Design suite was used to create the combination lock state design. The functional verification was carried out using the Quartus II/MaxPlus II package. The Mealy type of Finite State Machine (FSM) is the subject of this report. The output of a sequential circuit is a combination of both the flip-flops' present state and the circuit's input in a Mealy state machine. One of the key hardware description languages utilized in this research to create the Combination Lock system is VHDL (very high-speed integrated circuit HDL). The security technology for machines is discussed in this report. This can be accomplished using the Quartus II/MaxPlus II package. A combination lock state machine is described in this paper as being designed to activate an "unlock" operation when an acceptable binary code is provided, however, the lock can only be opened when the correct code (password) is typed into the system. Because the input data recorded on X during the previous the UNLK result was 0110111 upon 7 clock strikes is 1 when X is 0 (the desired unlock code).

Keywords: ALTERA Design, Lock, Quartus II/MaxPlus II, VHDL

Introduction

The combination lock state machine is essential for achieving a digital system's control and decision- making logic. A digitized password lock system is basically a protective device that allows any user to unlock the lock by inputting the suitable binary data. The system's goal is to provide improved security features. It also makes it impossible for the lock to be broken. It's a keyless security system that just requires the owner to have the correct code to unlock it (Yeap, *et al.* 2012) [11]. A code lock-based system is the most common type of lock used in electronic safes. It is possible that the code used is numeric or alpha numeric (Singh and Abhilasha, 2013) [9]. The code used in this project is binary in nature.

It is general knowledge that the old-fashioned key and lock method has flaws in terms of effectiveness because the keys are vulnerable to being lost or duplicated (Li *et al*, 2018) [4]. Intruders can readily replicate keys, making systems and property vulnerable to break-ins (Yeap, 2012) [11]. According to statistical analysis, (Schiefer, 2015) [8] asserted that a property without either an access control seems to be more prone to breaking into than a residence with an access control. When an invasion occurs, it generally appears unpredictably, there may be obvious hard evidence of it happening whenever accomplished, but it may slip without notice, based on the invader's intent (Jadaa, 2019) [5]. The report's major goal is to use VHDL to implement an electronic combination lock system that will improve the security of a home, business, or other building. Arabahmadi, 2019 [1] argued that a s a result of urbanization and various types of crime, the efficiency of the archaic key lock has been called into doubt in recent years. This electronic lock system is a protection mechanism that only the landlord any authorized person may use to unlock the door by inserting the special password on the Altera Design suite. By building the password and modifying the text length parameter, the secret code is entered. For instance, the hidden password is made up of seven-digit numbers. As a result, the house owner needed to provide both the exact special password and the length of the password. To unlock the door, the landlord or occupant of the property had to use a combination of the cipher text on the drop keys.

The door will be unlocked only when the code provided matches the preset password. Furthermore, if an invalid password is entered three times in a row, the smart key lock system will trigger the rest circuit, signaling that the system is in a holding condition.

Saleh *et al*., 2018 [7], offered a keyless lock system that was almost identical to the project of the application of Field Programmable Gate Arrays (FPGA) toward an intelligent home automation system was carried out. This report describes an electronic combination lock that is used daily. Lock state is a convenient concept for digital circuit with distinct "states" of activities, where logic functions generate responses and the next state as each clock edge approaches a consequence of inputs and current state. The simulation will be executed, and the results will be presented in the report.

Methodology

The state machine is a machine that detects sequences and demands a specified password to advance to the next stage. If the correct key for a state is not entered, the state machine returns to its initial state. A Combination lock can have whatever number of inputs and states. This research looks upon combination locks with three or four inputs and eight or sixteen states. After the last state condition is achieved and the appropriate key is provided for this state, the state machine asserts the "success" code, which is tied to the multiplexer activate password. The key sequence acts as a password for the rest of the circuit. Until reset, which returns the state machine to its starting state, is asserted, the state machine remains in the final state and continues to assert "success" (Bradley, 2016) [2]. This machine is Mealy type of machine, which is a finite state machine whose output values are dictated by its current state as well as the current input values.

Altera Quartus II is a complete multiplatform design environment that easily adapts to your specific design needs. It is a full-fledged system-on-a-programmable-chip (SOPC) design environment.

All steps of FPGA and CPLD design are covered by the Quartus II software. You can also utilize the Quartus II graphical user interface and command-line interface for each phase of the design flow with the Quartus II software. You can use one of these interfaces throughout the process, or you can switch between them at key stages. The Altera Quartus II design software, which includes System interface, is a hardware debugging and able to monitor tool that is configurable. It enables you to easily develop GUI elements and connect them to hardware in order to track efficiency, debug models, and demonstrate design performance. You can test your design at any time, whether it's in the lab or during simulation (Tomar, 2011) [11].

The system was created by transforming the state values of the combination lock into VHDL. Given that the lock's seven states are the alphabets A, B, C, D, E, F, G, and H, the system has two outputs, UNLK and HINT, and one input signal designated X, the value of which determines which state the system moves to next. The unlock (UNLK) output is 1 but only if X is zero and the sequence of inputs received on X during the preceding seven clock ticks was 0110111. The HINT output is 1 if but only if the present value of X is the correct one for bringing the machine closer to the "unlocked" state (with UNLK=1).

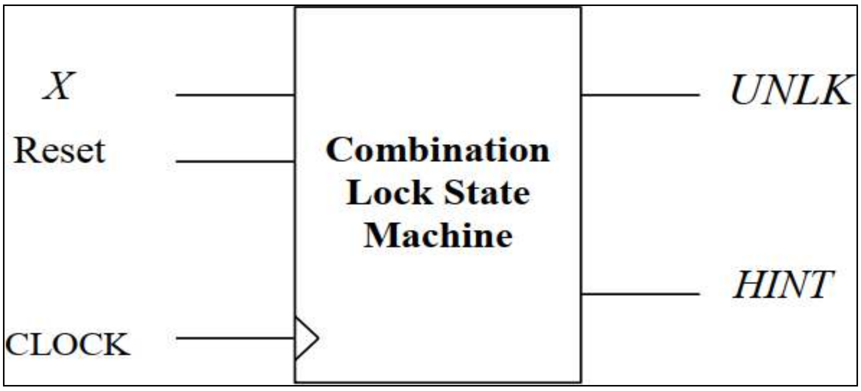


Fig 1: Block diagram of the combination lock state system

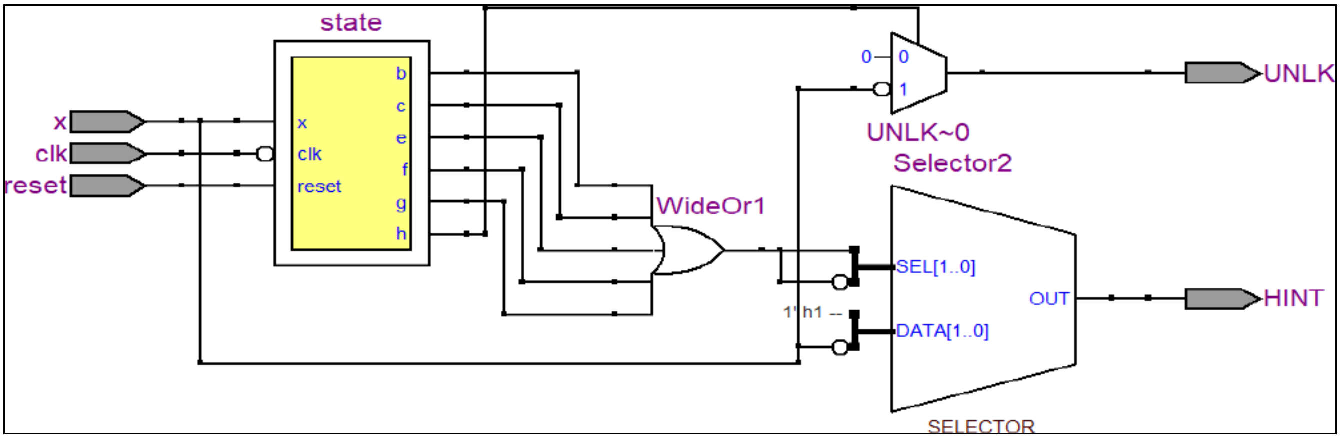


Fig 2: Block diagram of the RTL view

Table 1: State and output table for a combination lock machine (Where S and S* designate the current and next states, correspondingly)

Meaning	S	X	
		0	1
Got zip	A	B, 01	A, 00
Got 0	B	B,00	C,01
Got 01	C	B,00	D,01
Got 011	D	E,01	A,00
Got 0110	E	B,00	F,01
Got 01101	F	B,00	G,01
Got 011011	G	E,00	H,01
Got 0110111	H	B,11	A,00
		S*, UNLK	HINT

2.1 State Diagram

The first stage in constructing a combination lock state machine is determining how many states are required and which transitions between states are permitted. As illustrated in Fig. 3 below, A was chosen as the starting state in this state diagram. Each component (state) is defined by a circle, with arrows interconnecting the circles which reflect the rules for moving from one state to another. If X equals 1, the state machine in this system shifts from state A to state B (else, it remains in state A) and accordingly to H. The output values in each state are represented by the values along the path in the circle.

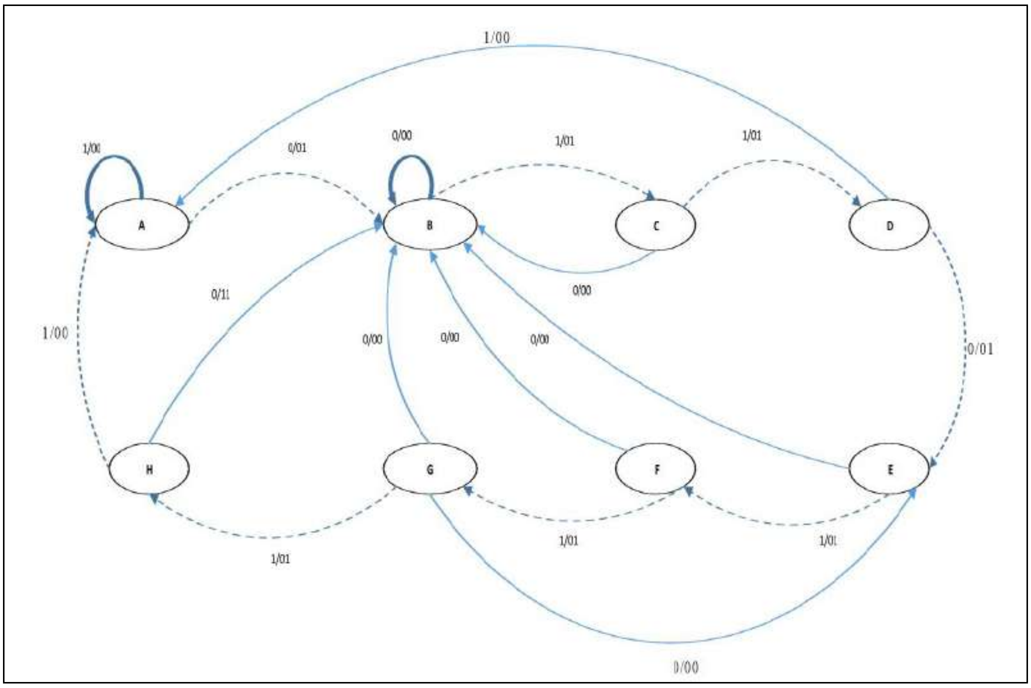


Fig 3: State diagram of the combination lock state machine

2.2 VHDL implementation of the design

VHDL is an abbreviation for very high-speed integrated circuit, while VHSIC is an abbreviation for VHDL Hardware Description Language (Charles, Roth and Lizy, 2016) [12]. VHDL is a programming language for simulating and synthesizing complex digital electrical circuits. The purpose of this project is to develop circuits for the programmable logic implementation of lock state machine using VHDL. In the first phase of designing the VHDL, the header is introduced:

```
library ieee;
use ieee.std_logic_1164.ALL.
```

The description of the system's external interface, including inputs, outputs, and the entity's name is entailed in VHDL entity. This paper uses the following format:

```
entity SmithDesign is port (clk, x, reset in bit;
UNLK, HINT: out bit);
end Smith Design;
```

2.3 Complete VHDL programme of the combination lock state machine

```
Library ieee;
Use ieee.std_logic_1164.all; -- header
entity Smith Design is -- entity declaration port (clk, x, reset
: in bit;
UNLK, HINT: out bit);
end SmithDesign;
architecture machine of SmithDesign is type state_type is
(A, B, C, D, E, F, G, H); -- defines the states signal state:
state_type; -- create a signal that uses different states begin
process (clk, reset) begin
if reset = '1' then state <= A; -- upon reset, set the state to A
elsif clk'event and clk = '0' then --if there is a rising edge of
the clock, then the what instructed below
case state is

When
A=> if x <='0' THEN STATE <= B;
else state <= A;
end if;

When
B=> if x <='1' then state <= C;
else state <= B;
end if;

When
C=> if x <= '1' then state <= D;
else state <= B;
end if;
```

```
When
D=> if x <='0' then state <= E;
else state <= A;
end if;

When
E=> if x <='1' then state <= F;
else state <= B;
end if;

When
 F=> if x <='1' then state <= G;
else state <= B;
end if;

When G=> if x <='1' then state <= H;
else state <= E;
end if;

When H=> if x <='0' then state <= B;
else state <= A;
end if; end case;
end if;
end process;
process (state, x) begin case state is

When
A=> if X <='0' then UNLK <= '0';
HINT <= '1';
else
UNLK <= '0';
HINT <= '0';
end if;

When
B=> if X <='1' then
UNLK <= '0';
HINT <= '1';
else UNLK <='0';
HINT <= '0';
end if;

When C=> if X <='1' then
UNLK <= '0';
HINT <= '1';
else UNLK <='0';
HINT <= '0';
end if;

When D=> if X <='0' then
UNLK <= '0';
HINT <= '1';
else UNLK <='0';
HINT <= '0';
end if;

When E=> if X <='1' then
UNLK <= '0';
HINT <= '1';
else UNLK <='0';
HINT <= '0';
end if;

When F=> if X <='1' then
UNLK <= '0';
HINT <= '1';
else UNLK <='0';
HINT <= '0';
end if;

When
G=> if X <='1' then
UNLK <= '0';
HINT <= '1';
else UNLK <='0';
HINT <= '0';
end if;

When H=> if X <='0' then
UNLK <= '1';
HINT <= '1';
else UNLK <='0';
HINT <= '0';
end if; end case;
end process; end machine;
```

2.4 Design simulation procedure

The goal of simulation is to perform technical evaluation including to precisely reproducing the desired system behaviour. Quartus II/MaxPlus II simulator software was used for the simulation. The simulation's output was presented in waveforms. However, the waveform was edited for the seven clock ticks of 0110111 after compiling the design, and the simulation was run to determine the required output.

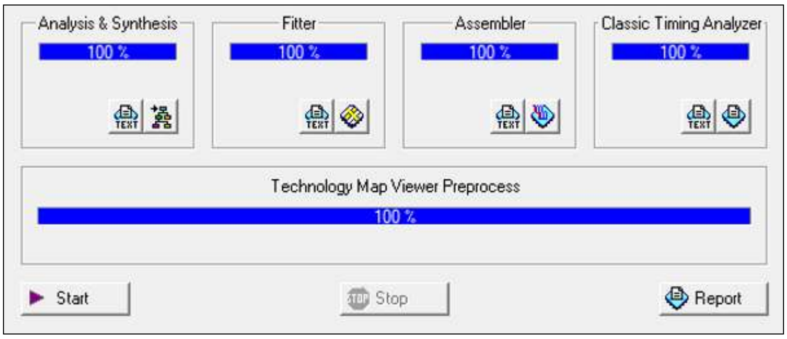
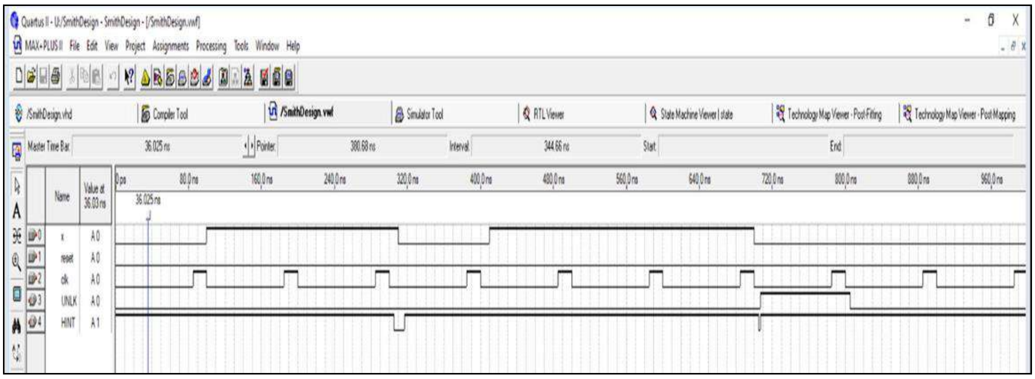


Fig 4: Compiler stage

The waveform was edited for the seven clock ticks of 0110111 after compiling the design, and the simulation was run to determine the required output.

Result

The combination lock state machine design was successfully using VHDL. The system's output waveforms have been confirmed and displayed in Quartus II/MaxPlus II simulator. The output result simulated was comparable to the predefined simulation program codes 0110111.


Quartus II - U:/SmithDesign - SmithDesign - [/SmithDesign.vwf]
MAX+PLUS II File Edit View Project Assignments Processing Tools Window Help
/SmithDesign.vhd
Compiler Tool
/SmithDesign.vwf
Simulator Tool
RTL Viewer
State Machine Viewer | state
Technology Map Viewer - Post-Fitting
Technology Map Viewer - Post-Mapping
0 ps
80.0 ns
160.0 ns
240.0 ns
320.0 ns
400.0 ns
480.0 ns
560.0 ns
640.0 ns
720.0 ns
800.0 ns
880.0 ns
960.0 ns
36.025 ns

Master Time Bar: 36.025 ns Pointer: 380.68 ns Interval: 344.66 ns Start: End:

	Name	Value at 36.03 ns
0	x	A 0
1	reset	A 0
2	clk	A 0
3	UNLK	A 0
4	HINT	A 1

Fig 5: Waveform presentation of the combination lock state machine

Discussion

The state graph was created using the design table, which included transitions between the states of the state lock system. The State of the Combination Lock Machine was described in VHDL using the Quartus II/MaxPlus II package. A state machine's number of states, inputs, and outputs were determined using VHDL. I assumed that the reset is active-low and that the status changes when the clock rises. The Quartus II/MaxPlus II package was used to capture and verify the design. There were no problems in the design, thus it was compiled and determined to be correct. The waveform file (*.vwf) was updated to define all in terms of input and output signals that was examined.

The design was Simulated the function of the Combination Lock was verified. When the clock period was 100 ns. The VHDL codes of the smart lock system had been installed on the Altera Design Suite and the simulation period was 100 ns with the time range from 0 ps to 1,0 µs. from 0 ps to 100.698 ns X = 0 and from 100.698 ns to 310.826 ns, X=11, from 310.826 ns to 410.36ns, X=0, from 410.36 ns to 699.65 ns, X=111. From 699.65 ns to 707.799 ns, there was a slight propagation delay that triggered the UNLK at 708.381ns to 805.587ns. Therefore, after the preceding seven clock ticked 0110111, the UNLK output at the rising edge triggers to 1 when X is 0 and the sequence of inputs received on X at the preceding seven clock ticks was 0110111. The HINT output is 1 when the current value of X is 0 that moved the machine to the “unlocked” state (with UNLK=1).

Conclusion

One of the project's main advantages is that it provides great security by deceiving someone attempting to hack the lock by the number of input bits required. If a password is hacked, the user has the option to change it, giving them more flexibility. The security difficulty is mitigated to some extent by the design of combinational locks.

References

1. Arabahmadi R. A reversible fluorescence -ON-OFF-ON sensor for sequential detection of F- and Cu2+ ions and its application as a molecular-scale logic device and security keypad lock. Journal of Coordination Chemistry. 2019;72(5-7):1187-202.
2. Bradley DC. Active FPGA security through decoy circuits. MSc dissertation, Air Force Institute of Technology. Wright-Patterson Air Force Base, Ohio, 2006, 49.
3. Sam SO, Sewe S, Kimathi G, Wainaina M. Antecedents of patients Covid-19 management outcomes. International Journal of Statistics and Applied Mathematics. 2021;6:109-117.
4. Li W, Li H, Gong A, Ou Y, Li M. An intelligent electronic lock for remote-control system based on the internet of things. Journal of Physics: conference series. IOP Publishing. 2018, 1-7.
5. Jadaa KJ, Kamarudin LM, Ahmad RB, Hussein WN. Multi objects detection and tracking system for smart home using wireless sensor network. International Journal of Advanced Trends in Computer Science and Engineering. 2019;8(5):2434-41.
6. Muhibul HB. Finite State Machine Based Directional Counter Design Using VHDL, 2007. Availableat: https://www.researchgate.net/publication/287975558_Finite_State_Machine_Based_ Directional_Counter_Design_Using_VHDL (Accessed: 4th April 2022).
7. Saleh SB, Mazlan SB, Hamzah NI, Karim AZ, Zainal MS, Hamzah SA. Smart home security access system using field programmable gate arrays. Indonesian Journal of Electrical Engineering and Computer Science. 2018;11(1):152-60.
8. Schiefer M. Smart home definition and security threats. In ninth International Conference on IT Security Incident Management & IT Forensics, 2015, 114-8.
9. Singh C, Abhilasha S. FPGA Based Implementation of Electronic Safe Lock, IOSR Journal of Electronics and Communication Engineering. 2013;6(1):19-23.
10. Tomar A. Altera: Introduction to the QUARTUS II Software 2011. Available at: https://community.element14.com/products/devtools/technicallibrary/w/documents/9382/altera- introduction-to-the-quartus-ii-software. Accessed: 25th March 2022.
11. Yeap KH, Nisar H, Chong KH, Rizman ZI, Yang CC, Yong YT. *et al*. Design of a Keyless Coded Home Lock System Using Verilog Hardware Description Language, International Journal of Electronics Communication and Computer Technology (IJECCT). 2012;3(1):10-16.
12. Charles H, Roth Jr, Lizy KJ. Digital Systems Design using VHDL. 3rd Edn. Cengage Learning, 2016, 40.