ISSN: 0975-766X
CODEN: IJPTFI
Available Online through *Research Article*
www.ijptonline.com

A HANDY APPROACH FOR TEACHING AND LEARNING DIGITAL VLSI DESIGN USING EDA TOOLS

[1]V Elamaran, [2]D Jayapriya, [2]R Hemavathy [3]Har Narayan Upadhyay
[1]Assistant Professor, Dept., of ECE, SEEE, SASTRA University, Thanjavur, Tamil Nadu, India.
[2]Dept., of ECE, SEEE, SASTRA University, Thanjavur, Tamil Nadu, India.
[3]Associate Dean, Dept., of ECE, SEEE, SASTRA University, Thanjavur, Tamil Nadu, India.
Email: jayapriya1993ece@gmail.com

Received on 04-11-2015 *Accepted on 24-12-2015*

Abstract

Very Large Scale Integration (VLSI) has proven instrumental in ushering in the era of powerful and inexpensive integrated circuits packed in small chip sizes. Hence, a foundational course on VLSI Design has become a necessity for any student of Electrical, Electronics or Computer Engineering. Conventionally, courses on Basic VLSI design are taught with an emphasis on the characteristics of CMOS logic, complemented by a laboratory course on HDL programming, which provides a basis for low-level description of both ASICs and FPGAs. However, there is a disconnect between the physical aspects that the student learns in the design theory course and the programming-based laboratory course. This study aim to resolve this discrepancy by introducing an Electronic Design Automation (EDA) tool called Microwind, that allows for both schematic and mask design, apart from a host of other features, including Verilog compilation.

Keywords: CMOS VLSI Design, Layout, Schematic,Simulation, Testing Timing, Layout Timing.

I. Introduction

CMOS logic is unarguably the most preferred logic used in integrated circuits today, due to its inherent property of extremely low static power dissipation. However, as in any logic structure, there are plenty of limitations that the designer has to keep in mind while drawing up the schematic [1].

Some of these limitations are threshold voltage effects, load capacitances, and hence higher RC time constants. Another significant characteristic to be considered by the chip designer is the difference in mobility of the charge carriers in PMOS and NMOS transistors, which have to be taken into account while deciding their (W/L) ratios, in order to ensure comparable rise time and fall time. Finally, there is the ever-present problem of faults. An improper

mask or manufacturing process may short signals to other signals/metal contacts, or worse, to supply or ground rails, which may lead to chip failure [2].

The primary objective of the study is, as stated before, to provide a novel simulation-based approach to understanding the properties of MOS transistors and circuits. This can be done by working on simple circuits that illustrate the above concepts. We first begin by illustrating the threshold voltage problem by way of a "bad buffer". Next, we proceed with explaining the parameters that determine time delay- using fundamental circuit theory concepts to show how load resistances and capacitances affect the overall circuit- while simultaneously conveying the importance of the (W/L) ratio in PMOS and NMOS transistors in determining the rise time and fall time of MOS transistors and circuits. We end by demonstrating simple methods with which the verification of the functioning of basic logic gates can be achieved [3] [4].

This paper is organized as follows. Section II illustrates the effects of threshold voltage in CMOS circuits. Section III describes the importance of loading effects on switching speed. Section IV describes the digital design and verification fundamentals using EDA tools simulation results and Section V gives the conclusion.

II. Thresold Voltage Effects

A. The poorly-designed 'buffer'

The classical example of demonstrating the effects of the threshold voltage present at the gate is taking the fundamental CMOS inverter and switching the positions of the NMOS and PMOS transistors. At first sight, this looks like a circuit that replicates the input logic level at the output [1].

On closer inspection, however, we see that this is not the case- since the NMOS transistor is a poor conductor of logic 1 and the PMOS transistor is a poor conductor of logic 0. This is shown in Fig. 1. We see that when the input is high, the source of the NMOS transistor rises to the lesser of the two potentials (the gate voltage and the drain voltage) - however, the effective gate voltage is now lesser by one factor of the threshold voltage Vth.

Similarly, when the input is low, the PMOS transistor conducts: while the output voltage ought to ideally be zero, it is raised by a factor of one Vth.

This leads to a lessened output voltage swing, and may do more harm than good when used as a 'buffer'. In place, two CMOS inverters in cascade would serve well to restore the voltage at the input [5]. Fig. 2 shows the output voltage swing in a Voltage Vs Time graph. For reference, Vdd is 1.2V while Vth is 0.4V.

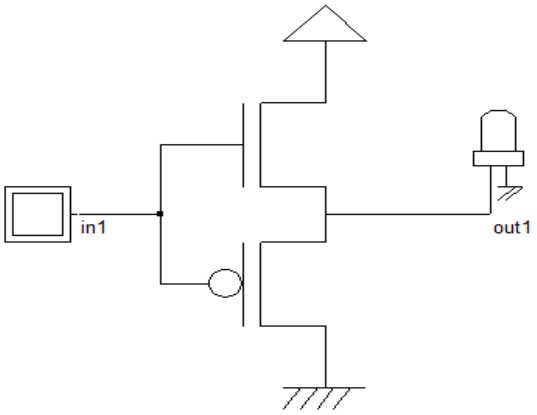

Fig. 1. A buffer circuit.

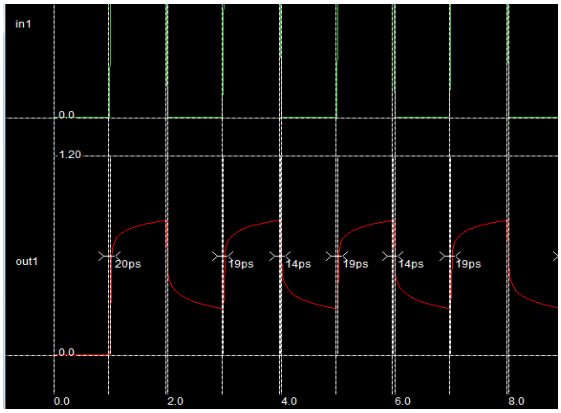

Fig. 2. Demonstrating the decreased voltage swing.

B.The Pass Transistor

While the pass transistor does not implement CMOS logic, it is yet another great example that illustrates the problems caused by threshold voltage effects. The idea behind pass transistor logic is to feed one of the inputs itself to the drain (or source) of an NMOS (or PMOS) transistor, thus effectively reducing the total number of transistors [1] [5]. Such examples can conveniently be demonstrated by means of simulation. The threshold voltage problem caused by pass transistors(especially when they are cascaded) are succinctly explained by Fig. 3. below:

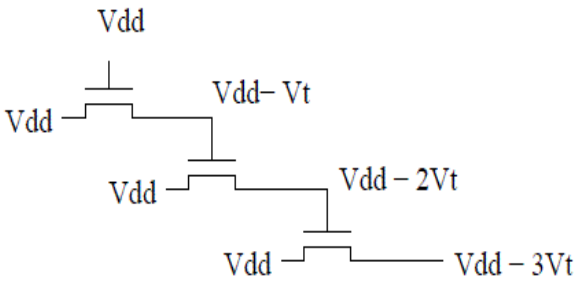

Fig. 3. Demonstrating the threshold-loss problem.

III. Loading Effects on Switching Speed

A. Input Load Capacitances

The load driven by a signal plays a large role in determining the switching speeds. Best conveyed with an example, the reader is asked to consider the circuit shown in below as in Fig. 4.

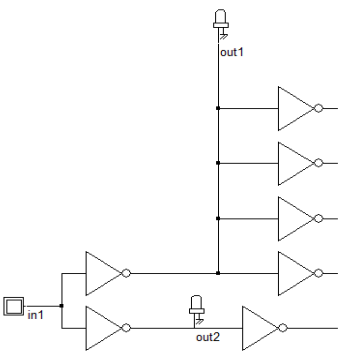

Fig. 4. Emphasizing an inverter driving four loads.

An input driving 2 (CMOS) inverters is shown. The inverter at the top has to drive 4 other inverters, incurring 4 load capacitances that are theoretically in parallel, thus increasing the overall load capacitance. It follows that the RC time constant is increased, thus reducing the switching speed of the top inverter. The bottom inverter, however, faces no such issues [1] [6]. This is aptly understood by simulating a switching input in a time-varying graph, as shown below in Fig. 5. and Fig. 6. Here, notable is the significantly high fall time of 22 ps. The switching times of the inverter described by Fig. 5 approaches nearly twice those of the inverter described by Fig 6. Such an example is ideal to illustrate the pitfalls of large fan-out, i.e., driving large loads [7].

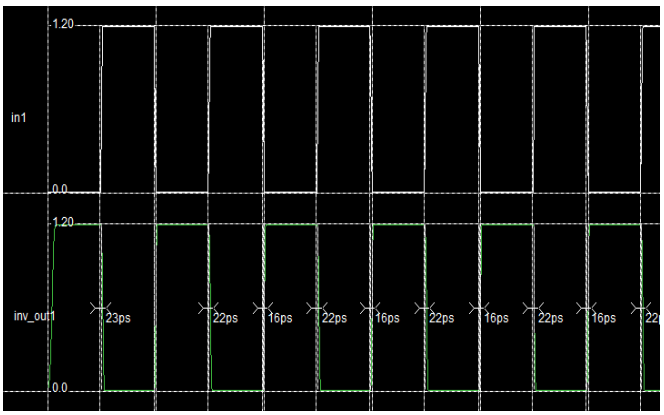

Fig. 5. Switching times of the inverter driving a high capacitive load.

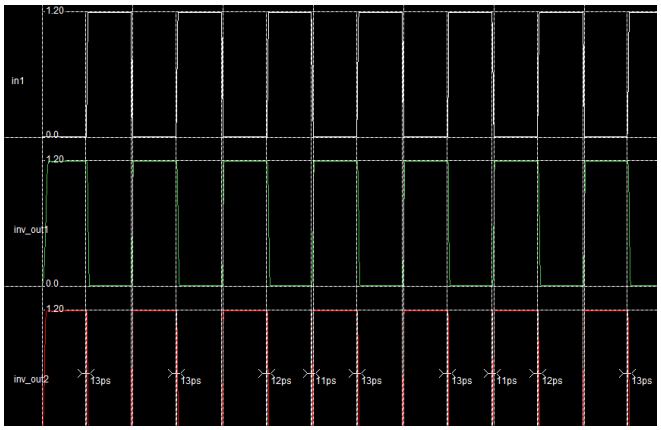

Fig.6. Switching times of the inverter driving a lower capacitive load.

B. Low-to-High and High-to-Low Transitions

The propagation delays like low-to-high (tLH) and high-to-low(tHL) for a circuit are demonstrated with a 3-input NOR gate as shown in Fig. 7. Since three pMOS transistors are connected in series in a pull-up network, the low-to-high propagation delay is more comparatively with a pull-down network in which only one transistors plays a role to obtain the transition from high-to-low [8]. The corresponding simulation results are obtained in Fig. 8. This design need 37 picoseconds for low-to-high transition and 4 picoseconds for high-to-low transition. Fig. 9 shows the schematic diagram of the three-input NAND gate in which the low-to-high propagation delay is low compared with the high-to-low delay [9] [10].

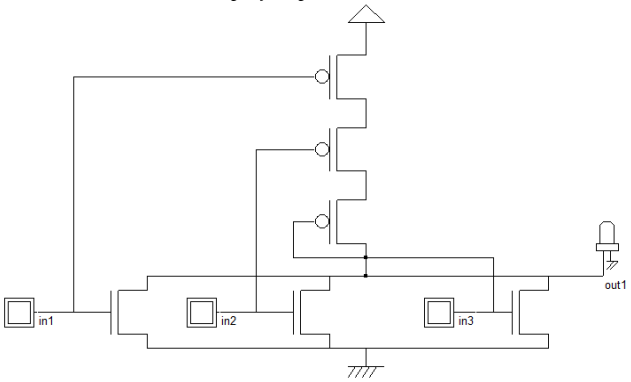

Fig. 7. Three input NOR gate.

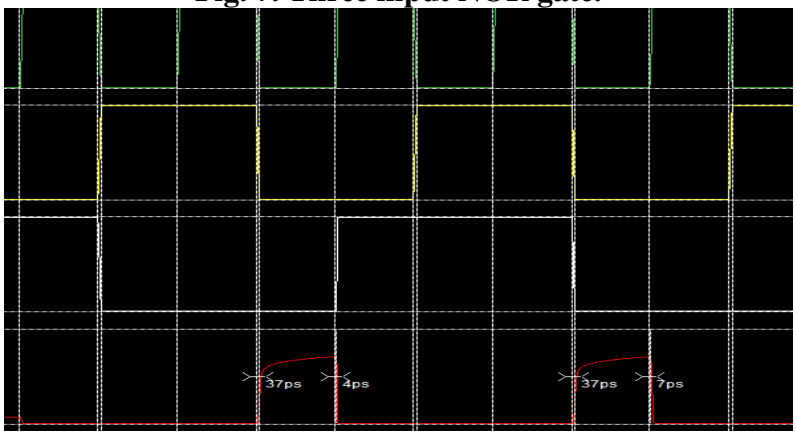

Fig. 8. Low-to-high and high-to-low transition timing results.

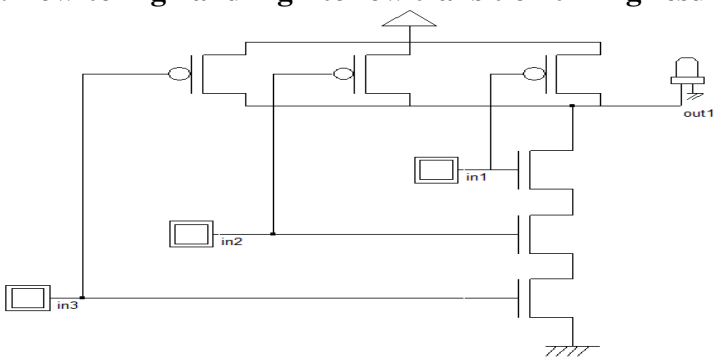

Fig. 9. Three input NAND gate.

C. Trade-offs between delay and power-dissipation

As stated before, CMOS logic is widely used due property of very low static power dissipation. However, this comes at the expense of larger time delay [11]. This is because, either the Pull-up network or the Pull-down network invariably consists of a large number of transistors in series, which contributes to a significant resistance, thus increasing the RC time constant [12]. This is best explained by way of an example, that demonstrates a theoretical 8-input AND gate in Fig. 10. A convenient but expensive solution to increase the speed of the circuit is to implement an alternate logic structure known as pseudo-NMOS logic, as depicted in Fig. 11 below. The biggest pitfall of such a structure is the significantly higher static power dissipation [13]. On inspection, it is seen that the Pull-up network is always on. Therefore, whenever any of the inputs is logic 0, there exists a direct path between supply to ground, with little resistance provided by the two ON transistors. Such a concept can be lucidly explained by means of tools such as DSCH and Microwind [14] [15].

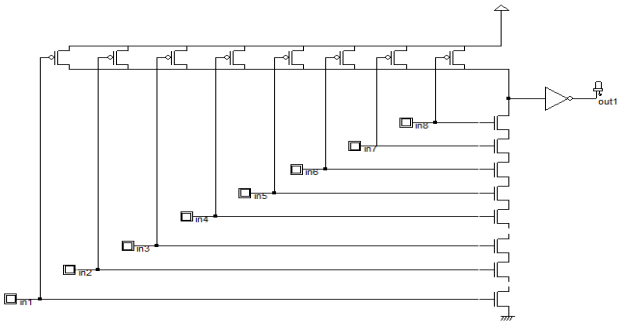

Fig. 10. 8-input AND gate using CMOS logic.

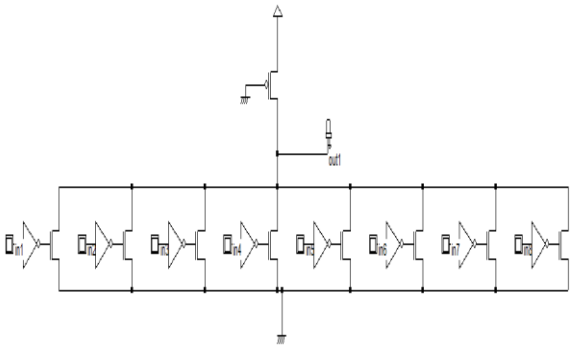

Fig. 11. 8-input AND gate using pseudo-NMOS logic.

IV. Digital Design and Verification

The digital schematic tool DSCH allows the user to work at various levels of abstraction, including transistor-level, gate-level and circuit-level abstraction, which provide a great way for the learner to understand the building blocks of large-scale devices.

A. The Barrel Shifter

One of the fundamental components of the Arithmetic and Logical Unit (ALU) is the Barrel Shifter, which has been widely covered in literature [6]. The barrel shifter can shift multiple bits of the input in a single clock cycle. The most popular implementation of barrel shifters is using multiplexers: the schematic of a 4-bit barrel shifter using four 4:1 multiplexers is shown below in Fig. 12. Such a circuit can be implemented in DSCH by providing buttons at the input and LEDs at the output to verify functionality.

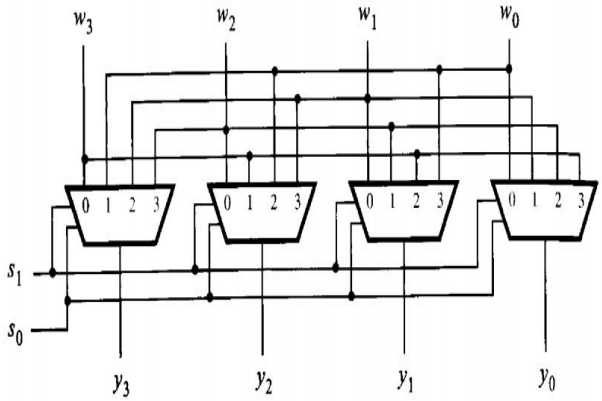

Fig. 12. A 4-bit barrel shifter.

B. Parity Checker

Parity bits are widely used in digital communication to ensure that errors are detected upon reception of the message [1]. At its heart, the XOR is responsible for checking whether there are an even number of signals that are logic 1. Fig. 13 shows a circuit that takes 4 input signals and returns an output that is high when and odd number of inputs are high. The working of the above figure can be aptly verified from Fig. 14 which is obtained from DSCH:

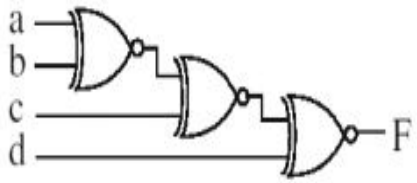

Fig. 13. Parity checker using XOR gates.

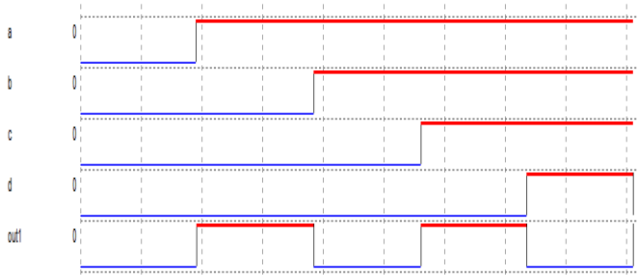

Fig. 14. Verification of the parity checker using timing diagram.

C. Full Adder using Mux

A full adder circuit can also be designed using multiplexer which is shown in Fig. 15. When the inputs 'a' and 'b' are not equal, the C_{out} depends on the input C_{in}. If Cin = 1, the Cout becomes also '1' and vice versa [10]. When the inputs 'a' and 'b' are equal, the C_{out} depends on either 'a' or 'b' irrespective of C_{in}. The sum is calculated as usual with the 'a' xor 'b' xor 'C_{in}'. This is shown in as schematic in Fig. 15. The DSCH tool generate a Verilog Hardware Description Language (HDL) code for the given module, which is shown in Fig. 16. This Verilog code is compiled by using the layout editior tool called Microwind, which generate an automatic layout for the given design. Fig. 17 shows an layout for this full adder module. A functional verification can also be done on the layout to recheck the design [16]. This elegant tool is handy to analyze the design in terms of layout, number of pmos and nmos transistors, and power dissipation analysis [17].

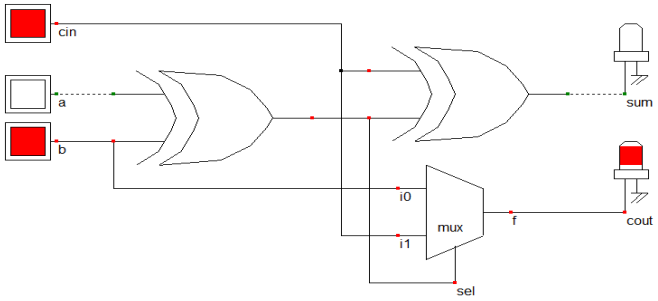

Fig. 15. A full adder circuit using a multiplexer.

```
module fulladder_mux(cin,b,a,sum,cout);
        input cin, b, a;
        output sum, cout;
        mux #(10) mux(cout,b,cin,w3);
        xor #(23) xor2(w3,a,b);
        xor #(16) xor2(sum,cin,w3);
  endmodule
```

Fig. 16. A Verilog HDL script.

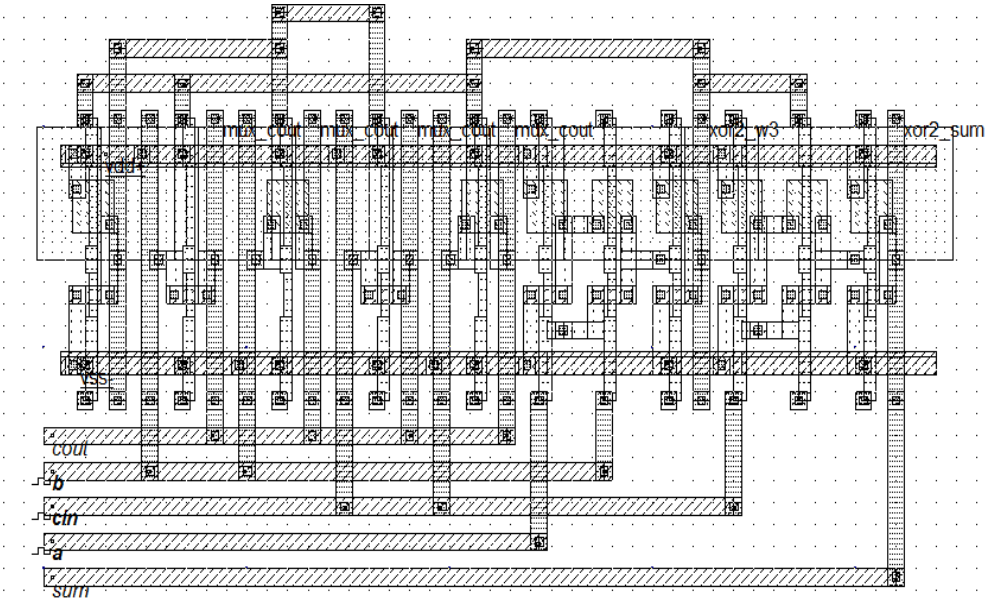

Fig. 17. A layout generated using Microwind.

D. Other Logic Functions

Circuits that are used to implement a plethora of logic functions can be simulated. Here is an example of an alternative circuit that implements the NOR functionality, which is shown in Fig. 18. The wiring of the circuit seems unorthodox, but on further inspection it is indeed seen that the functionality of the 2-input NOR gate is implemented and the truth table is shown in Fig. 16.

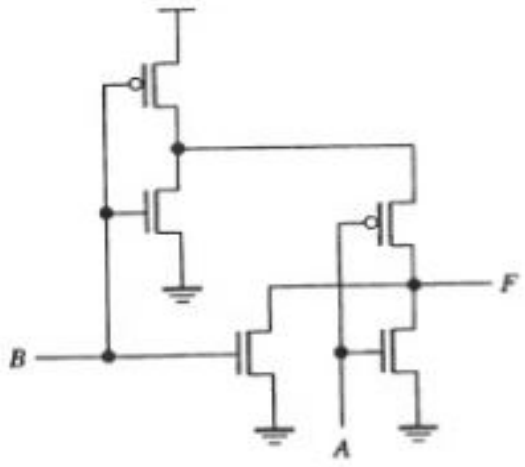

Fig. 18. An unusual NOR implementation.

A	B	F
0	0	V_{DD}
0	V_{DD}	0
V_{DD}	0	0
V_{DD}	V_{DD}	0

Fig. 19. Truth table for the 2-input NOR gate.

V. Conclusions

The best method to learn and study digital VLSI design fundamentals is by practice with demonstrations which helps reducing the gap between the industry and academy. The EDA tools play a vital role to analyze and study the circuits which helps the student and faculty cum research community. In this study, the fundamentals of threshold effect with loss problem, pass-transistors are demonstrated with most clarity. In addition, the concept of fan-in and fan-out is demonstrated with the help of 8-input AND gate. Other forms of combinational logic circuits such as 4-bit barrel shifter, parity checker are verified using the schematic editor tool. Further, the layout editor tool is used to compile the verilog script and to generate the layout for the design. The sole objective of the study to understand the fundamentals of the digital VLSI design theory by doing demonstration, which is fulfilled.

VI. References

1. Neil Weste, David Harris, and Ayan Banerjee, CMOS VLSI Design, Pearson Education, 2008.
2. R.Jacob Baker, Harry W.LI., and David K.Boyce, CMOS Circuit Design, 3rd Edition, Indian reprint, PHI, 2000.
3. Frederick J.Hill, and GR Peterson, Computer Aided Logical Design with Emphasis on VLSI, 4th Edition, John Wiley & Sons.
4. John F.Wakerly, Digital Design: Principles and Practices, 4th Edition, Prentice Hall, 2005.
5. John P.Uyemura, Introduction to VLSI Circuits and Systems, John Wiley & Sons, 2002.
6. Stephen Brown, Zvonko Vranesic, Fundamentals of Digital Logic with VHDL Design, TMH, 2002.
7. Volnei A.Pedroni, Digital Electronics and Design with VHDL, Morgan Kaufmann Publishers, 2008.
8. J.Bhaskar, A VHDL Primer, 3rd edition, B.S.Publications, India, 2010.
9. John P.Uyemura, Chip design for Submicron VLSI: CMOS Layout and Simulation, Thmson, 2005.
10. S Hari Hara Subramani, K S S K Rajesh, and V Elamaran, “Low Energy, Low Power Adder Logic Cells: A CMOS VLSI Implementation”, Asian Journal of Scientific Research, vol.7, no.2, pp.248-255, 2014.

11. John P.Uyemura, Digital Systems Design: An Integrated Approach, Thomson, 2000.
12. S.M.Kang and Y.Leblebici, CMOS Digital Integrated Circuits: Analysis and Design, Third Edition, McGraw-Hill, 2002.
13. J.M.Rabaey, A.Chandrakasan and B.Nikolic, Digital Integrated Circuits: A Design Perspective, Second Edition, Pearson/PH, 2003.
14. V Elamaran, Narredi Bhanu Prakash Reddy, Kalagarla Abhiram, "Low Power Prescaler Implementation in CMOS VLSI," Proceedings of the International Conference on Emerging Trends in Electrical Engineering and Energy Management (ICETEEEM-2011), pp.16-19, Dec 2012.
15. V Elamaran, Har Narayan Upadhyay, "A Case Study of Nanoscale FPGA Programmable Switches with Low Power," International Journal of Engineering and Technology, vol.5, no.2, pp.1512-1519, Apr-may, 2013.
16. C.Saint and J.Saint, IC Layout Basics: A Practical Guide, McGraw-Hill, 2002.
17. K S S K Rajesh, S Hari Hara Subramani, and V Elamaran, "CMOS VLSI Design of Low Power Comparator Logic Circuits", Asian Journal of Scientific Research, vol.7, no.2, 238-247, 2014.

Corresponding Author:

D. Jayapriya*,

Email: *jayapriya1993ece@gmail.com*