ISSN XXXX XXXX © 2018 IJESC

Research Article Volume 8 Issue No.6

Logic Design and Implementation of Half-Adder and Half Subtractor using NAND Gate Given the VHDL Descriptions

Ayuba John[1], Ngubdo Maigana Aduwamai[2]
Department of Computer Science
Federal University Dutse, Jigawa State, Nigeria[1]
Federal Polytechnic Mubi, Nigeria[2]

Abstract:
This paper described a detail laboratory report of a printed circuit board (PCB) design and implementations of half-adder and half-subtractor as a combinational circuit using NAND logic gate only and we also introduced a flip – flop (latch) in to the design which makes it to have two stable states and is used to store state information. Each of the steps involved in the designed are clearly outline in this paper and the behavior of the combinational circuit are evaluated using VHDL descriptions language. The main focus of this paper is to help students and beginners to understand the basic steps involves in the laboratory for a PCB design from simple to complex logic circuits.

Keywords: VHDL Description, PCB Design, Combinational Circuit, Logic Gate.

1. LITERATURE SURVEY

In literature [1], they mentioned early MOS threshold logic solutions and detail numerous very-large-scale integration (VLSI) implementations including capacitive (switched capacitor and floating gate with their variations), conductance/current (pseudo-nMOS and output-wired-inverters, including a plethora of solutions evolved from them), as well as many differential solutions. This [2] presents the methods required to implement a high speed and high performance parallel complex number multiplier. The system has been designed efficiently using VHDL codes for 8x8-bit signed numbers and successfully simulated and synthesized using Xilinx. In [7], they introduce a unique method that replaces the Binary to Excess-1 converter (BEC) using common Boolean logic. Experimental analysis illustrates that the proposed architecture achieves advantages in terms of speed, area consumption and power. This [8] describes the architecture, design & implementation of 2 bit ternary ALU (T-ALU) slice. The proposed ALU is designed for two-bit operation & can be used for n bit operations by cascading n/2 ALU slices. In this [5], they proposed the design of a new reversible gate called TR gate. Further, they investigated the design of reversible binary subtractor based on the proposed TR gate .In [4] work, the design and implementation of a low power ternary full adder are presented in CMOS technology. In a ternary full adder design, the basic building blocks, the positive ternary inverter (PTI) and negative ternary inverter (NTI) are developed using a CMOS inverter and pass transistors. In [3], they developed half adder and half subtractor operated by the same DNA platform in an enzyme-free system and shares a constant threshold set point. The novel feature of the designed system is that the two required logic gates for the half adder (an AND and an XOR logic gate integrated in parallel) or the half subtractor (an XOR and an INHIBIT logic gate integrated in parallel) are achieved simultaneously with the same platform and are triggered by the same set of inputs. This [9] paper presents the practical details involved in the design and implementation of a contactless battery charger that employs a pair of neighboring printed circuit board (PCB) windings as a contactless energy transfer device. In [6], two versions of fabric-based active electrodes are presented to provide a wearable solution for ECG monitoring clothing. In [10], five hybrid full adder designs are proposed for low power parallel multipliers. The new adders allow NAND gates to generate most of the multiplier partial product bits instead of AND gates. In this paper we presented unique method of design and implementation of half-adder and half subtractor using NAND gate only and have used a VHDL hardware description language to describe the behavior of the circuits. We introduced a flip – flop (latch) in to the design which makes it to have two stable states and is used to store state information.

2. HALF ADDER AND HALF SUBTRACTORS

Aim and Objective:

To implement half adder and half subtractor:

i. Using only NAND gates.
ii. Giving the VHDL descriptions.

List of components:

NAND gates, Resistors, LEDs, Diodes, Voltage regulators, Capacitors and D- Flip Flop

Procedures:

I. The circuits were designed using protus ISIS and exported to PROTUS ARES where PCB layouts were made.
II. The layouts were printed on a photo paper using HP laser jet printer.
III. We cut a copper board to the size of the layout on the printed photo paper.
IV. The copper board was washed with soap and water to remove dirt and rust, and allow it to dry.
V. We placed the photo paper faced down on top of the copper board and pressed it with hot iron for 15 minutes until the toner is completely transferred to the copper board.
VI. We used hydrogen chloride (hydrochloric acid), to etch the copper board to produce the required trace.
VII. We rinsed out the chemical from the copper board.
VIII. We Used Tina to remove the black ink that covered the copper board.
IX. We then drilled the appropriate holes to mounts the ICs on the PCB board and solder the components.

X. We verified whether all the components are in good condition.

XI. We set up the half adder and half subtractor circuit; and feed the input bit combinations according to the truth table.

XII. We observed the output corresponding to input combinations and enter it in truth table as shown.

XIII. We notes down the output readings for half adder and half subtractor i.e the sum/difference and the carry/borrow bit for different combinations of inputs.

2.1 THE IMPLEMENTATION OF THE HALF ADDER USING NAND GATE

Theory:

A half adder is a logic circuit used for summing two one bit numbers or basically two bits. It has two inputs which receive the two bits and two outputs; with one generating sum output and other yield carry output. It is recognized that binary sum is normally achieved by Ex-OR gate, but for the first three rules, it does the binary adding and when the two inputs are logic 1, it ensures to not produce any carry. For us to realize the half adder using NAND gates as presented in the figure 1 below; we needed Five NAND gates. They are normally called half adders because is there is no possibility for them to sum the carry bit from preceding bit. This is a main drawback of half adders when used as binary adders particularly in real time situations which includes addition of multiple bits. To overcome this constraint, full adders are established.

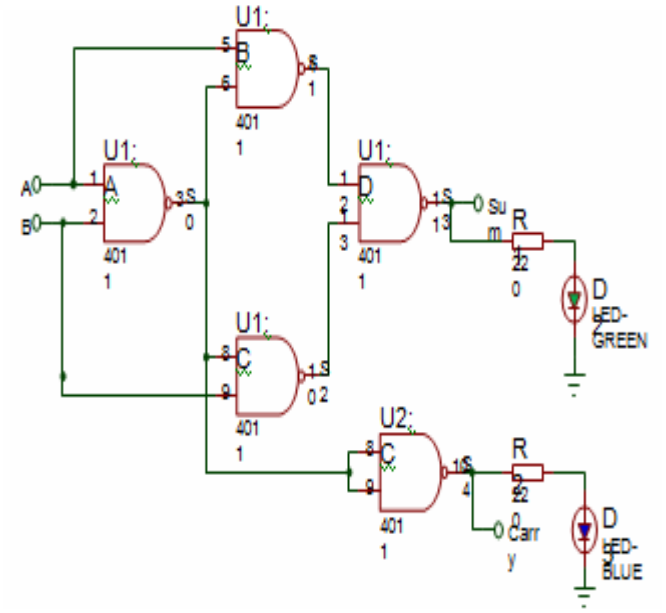


Figure.1. Circuit diagram of half adder using NAND gates

Table.1. Truth table of the half adder

Input		Output	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

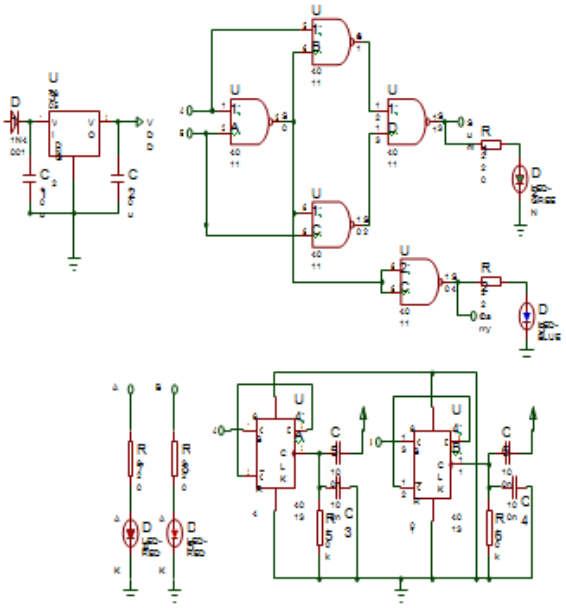


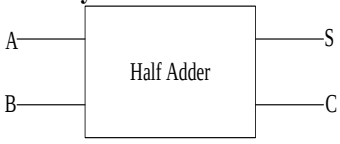
Figure.2. Circuit diagram for simulating the half adder

Figure.2. Fabricated half adder PCB board

The D- flip-flop circuit serves as the switch to toggles the bits in the circuit diagram for simulating the half adder above. The D flip-flop is used to tracts the input, making transitions which match those of the input D when the clock is enabled. The D stands for data, this flip-flop stores the value that is on the data line, it can be thought of as a basic memory cell.

2.2.THE VHDL DESCRIPTION OF THE HALF ADDER:

…The Entity for the half adder:




```
Entity half_adder is
   Port (A: inbit;
         B: inbit;
         S: outbit;
         C: outbit);
endhalf_adder;
```

…The Behavior Model for the half adder:

```
Architecture basic_beh of half_adder is
begin
process (A,B)
begin
      if (A = '1' and B = '1') then
      S <= '0';
            C<= '1';
      elsif (A= '1' and B = '0') then
            S<= '1';
```

```
        C<= '0';
        elsif (A= '0' and B = '1' )then
        S<= '1';
              C<= '0';
        elsif (A = '0' and B = '0') then
        S<= '0';
              C<= '0';
        else
        S<= '0';
              C<= '0';
        end if;
end process;
endbasic_beh;
```

…The Structural Model for the half adder:

```
architecturebasic_struct of half_adder is
begin
        U1.A: entity work1.nand(basic_nand)
        Port map (A,B,S0);
             U1.B: entity work2.nand(basic_nand)
        Port map (A,S0,S1);
        U1.C: entity work3.nand(basic_nand)
        Port map (B,S0,S2);
U1.D: entity work4.nand(basic_nand)
        Port map (S1,S2,S3);
        U2.C: entity work5.nand(basic_nand)
        Port map (S0,S4);
endbasic_struct;
```

…The NAND gates:

```
entity U1.A is
port (A: inbit;
              B: inbit;
              S0: outbit);
end U1.A;
entity U1.B is
port (A: inbit;
              S0: inbit;
              S1: outbit);
end U1.B;
entity U1.C is
port (B: inbit;
              S0: inbit;
              S2: outbit);
end U1.C;
entity U1.D is
port (S1: inbit;
              S2: inbit;
              S3: outbit);
end U1.D;
entity U2.C is
port (S0: inbit;
              S4: outbit);
end U2.C;
endnand gates;
```

… architecturebasic_signals of nand is

```
begin
    S0<= A nand B;
    S1<= A nand S0;
    S2<= B nand S0;
    S3<= S1nand S2;
    S4<= nand S0;
endbasic_signals;
```

3. IMPLEMENTATION OF HALF SUBTRACTOR USING NAND GATES

Theory: A combinational logic circuit which has a multiple output and does the subtraction of two bits of binary data is called a half subtractor. It has four variables which represent two input bits and two output bits correspond to the difference bit and borrow bit. It is also observed that binary subtraction is achieved by the Ex-OR gate with extra circuitry to do the borrow operation. Thus, to design a half subtractor, seven NAND gates is needed as depicted in the figure 3 below. For an instance of multi-digit subtraction, subtraction between the two digits is essentially done beside with borrow of the preceding digit subtraction. Therefore, a subtractor requires having three inputs, and hence, a half subtractor has restricted applications and it is rigorously not used in practice.

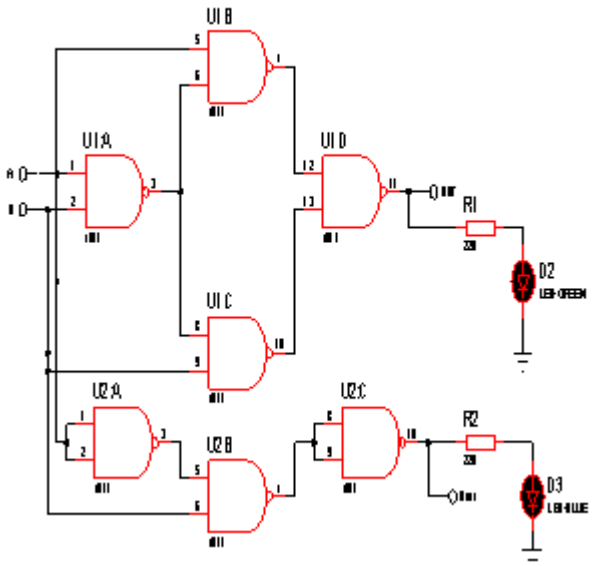


Figure 3: Circuit diagram of half subtractor using NAND gates

Table.2. Truth table of the half subtractor

Truth Table			
Input		Output	
A	B	D	B_0
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

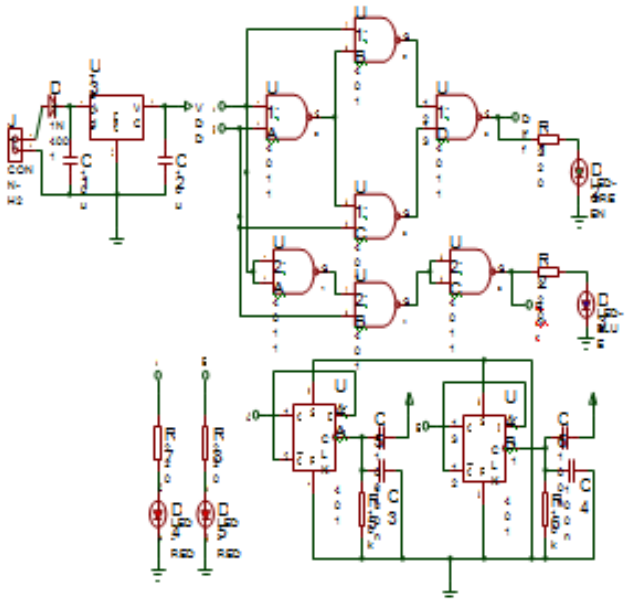


Figure.4. Circuit diagram for simulating the half subtractor

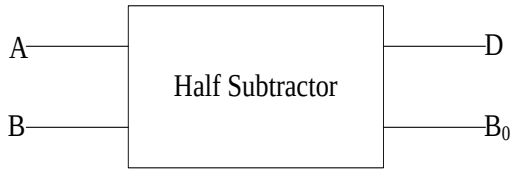
Figure.5. Fabricate half subtractor PCB board

The D flip-flop circuit serves as a switch to toggles the bits in the circuit diagram for simulating the half adder above, the D flip-flop is used to tracts the input, making transitions which match those of the input D when the clock is enabled.The D stands for data, this flip-flop stores the value that is on the data line, it can be thought of as a basic memory cell.

3.1.VHDL DESCRIPTION OF THE HALF SUBTRACTOR

Half Subtractor - Behavioral, Data flow and Structural model:

…Behavioral model of Half Subtractor




```
…entity Halfsubtractor_beh is
   Port ( A,B: inbit;
          B0,D: outbit);
endHalfsubtractor_beh;
…architecture Behavioral of Halfsubtractor_beh is
Begin
process (A,B)
begin
if (A='0')then
       B0<= B;
       D <= B;
else
       B <= '0';
       D <= not B;
end if;
end process;
end Behavioral;
```

…Data Flow model of Half Subtractor

```
…entity Halfsubtractor_df is
   Port (A: inbit;
         B: inbit;
         B0: outbit;
         D: outbit);
endHalfsubtractor_df;
…architecture dataflow of Halfsubtractor_df is
begin
process (A,B)
begin
     S0<= A nand B;
      S1<= nand A;
 S2<= A nand S0;
 S3<= S0nand B;
 S4<= S1nand B;
 D <= S2nand S3;
 B0<= nand S4;
end process;
end dataflow;
```

…Structural model of Half Subtractor

```
…entity Halfsubtractor is
  Port (A: inbit;
        B: inbit;
        B0: outbit;
        D: outbit);
endHalfsubtractor;
```

…NAND gates:

```
entity U1.A is
port (A: inbit;
          B: inbit;
          S0: outbit);
end U1.A;
entity U2.A is
port (A: inbit;
         S1: outbit);
end U2.A;
entity U1.B is
port (A: inbit;
          S0: inbit;
          S2: outbit);
end U1.B;
entity U1.C is
port (B: inbit;
          S0: inbit;
          S3: outbit);
end U1.C;
entity U2.B is
port (B: inbit;
          S1: intbit;
          S4: outbit);
end U2.B;
entity U1.D is
port (S2: inbit;
          S3: intbit;
          S5: outbit);
end U1.D;
entity U2.C is
port (S4: intbit;
          S6: outbit);
end U2.C;
end structural model;
```

…architecture Structural of Halfsubtractor is

```
component U1.A nand gate
port ( A,B: inbit;
         S0: outbit);
end component;
component U2.A nand gate
port (A: inbit;
            S1: outbit);
end component;
component U1.B nand gate
port (A,S0: inbit;
            S2: outbit);
end component;
component U1.C nand gate
port (S0,B: inbit;
            S3: outbit);
end component;
component U2.B nand gate
port (B,S1: inbit;
            S4: outbit);
end component;
```

```
component U1.D nand gate
port (S2,S3: inbit;
          S5: outbit);
end component;
component U2.C nand gate
port (S4: inbit;
          S6: outbit);
end component;
…Signal structural model
signal S: bit= '0';
begin
   S0: nand gate port map(A,B);
   S1: nand gate port map(A);
   S2: nand gate port map(A,S0);
   S3: nand gate port map(B,S0);
   S4: nand gate port map(B,S1);
   S5: nand gate port map(S2,S3);
   S6: nand gate port map(S4);
end Structural;
```

4. CONCLUSION

The technique involved in design of simple combinational circuit to a complex one is acquired perfectly and gate behaviors were observed using VHDL descriptions. It was also found out that, practical results recorded are corresponded to the theoretical results; as depicted in the truth tables in the figures above. However, every practical observation matched its expectations. Thus, the practical results were exactly the same as the theoretical results.

REFERENCE

[1]. V. Beiu, J. M. Quintana, and M. J. Avedillo, "VLSI implementations of threshold logic-a comprehensive survey," *IEEE Transactions on Neural Networks,* vol. 14, pp. 1217-1243, 2003.

[2]. S. S. Basha and B. Jahangir, "Design and implementation of radix-4 based high speed multiplier for alu's using minimal partial products," *International Journal of Advances in Engineering and Technology,* vol. 4, pp. 314-25, 2012.

[3]. Xu, S., Li, H., Miao, Y., Liu, Y., & Wang, E. (2013). Implementation of half adder and half subtractor with a simple and universal DNA-based platform.*NPG Asia Materials, 5*(12), e76.

[4]. A. Srivastava and K. Venkatapathy, "Design and implementation of a low power ternary full adder," *VLSI Design,* vol. 4, pp. 75-81, 1996.

[5]. H. Thapliyal and N. Ranganathan, "Design of efficient reversible binary subtractors based on a new reversible gate," in *VLSI, 2009. ISVLSI'09. IEEE Computer Society Annual Symposium on*, 2009, pp. 229-234.

[6]. C. R. Merritt, H. T. Nagle, and E. Grant, "Fabric-based active electrode design and fabrication for health monitoring clothing," *IEEE Transactions on information technology in biomedicine,* vol. 13, pp. 274-280, 2009.

[7].J. E. Clapten, E. Konguvel, and M. Thangamani, "VLSI implementation of low power area efficient fast Carry Select Adder," *International Journal of Computer Applications,* vol. 115, 2015.

[8]. A. Dhande and V. Ingole, "Design and implementation of 2 bit ternary ALU slice," in *Proc. Int. Conf. IEEE-Sci. Electron., Technol. Inf. Telecommun*, 2005, pp. 17-21.

[9]. B. Choi, J. Nho, H. Cha, T. Ahn, and S. Choi, "Design and implementation of low-profile contactless battery charger using planar printed circuit board windings as energy transfer device," *IEEE Transactions on Industrial Electronics,* vol. 51, pp. 140-147, 2004.

[10]. Z.Abid, H. El-Razouk and D.A. El-Dib, ―Low power multipliers based on new hybrid full adders‖, Microelectronics Journal, Volume 39, Issue 12, Pages 1509-1515, 2005.

[11]. Brown, S.D., *Fundamentals of digital logic with Verilog design*2007: Tata McGraw-Hill Education.

[12]. Lipsett, R., C.F. Schaefer, and C. Ussery, *VHDL: Hardware description and design*2012: Springer Science & Business Media.

[13]. Roth Jr, C.H. and L.K. John, *Digital systems design using VHDL*2016: Nelson Education.

[14]. Rushton, A., *VHDL for logic synthesis*2011: John Wiley & Sons.

[15]. Sjoholm, S. and L. Lindh, *VHDL for Designers*1997: Prentice Hall PTR

Author

Ayuba John[1] received his B.Eng. Engineering Degree in Computer Engineering from University of Maiduguri, Nigeria in 2010, and M.Eng. Computer Engineering Degree from University of Benin, Nigeria, in 2017, He has worked as a transmission Engineer in National Control Centre (NCC); Transmission Company of Nigeria (TCN) in 2014 and he is currently a lecturer from Federal University Dutse, Nigeria. His research interests are in the areas of Embedded System Technologies, wireless sensor network and Network Security. E-mail: ayuba.john@fud.edu.ng

Maigana Aduwamai Ngubdo[2] received his B.Eng.(Hons.) Degree in Electrical and Electronics Engineering from the Federal University of Technology, Yola (now Modibbo Adama University of Technology), Nigeria in 2008, and MSc. Networked Communications from Loughborough University, Leicestershire, United Kingdom in 2014, He has been working as a lecturer in the department of Computer Engineering at the Federal polytechnic Mubi, Nigeria from 2009 to date. He is a Member of the Nigeria Society of Engineers (MNSE), a Member Nigeria Computer Society (MNCS) and Members of the Institute of Electrical and Electronics Engineering (MIEEE). His research interests include telecommunication, wireless sensor network and Network Security.
E-mail: mangadzama@yahoo.com