LAB-5

date: 11 February, 2020

Lab Title: Magnitude Comparator

Recorded and Submitted By: Bhesaniya Abhisha

Objective: To write program in VHDL & Verilog both for 4 bit magnitude comparator and test it in DE SoC-1.

Implementation:

- Magnitude comparator is logical circuit which has 2 4-4 bit input and 3 output.

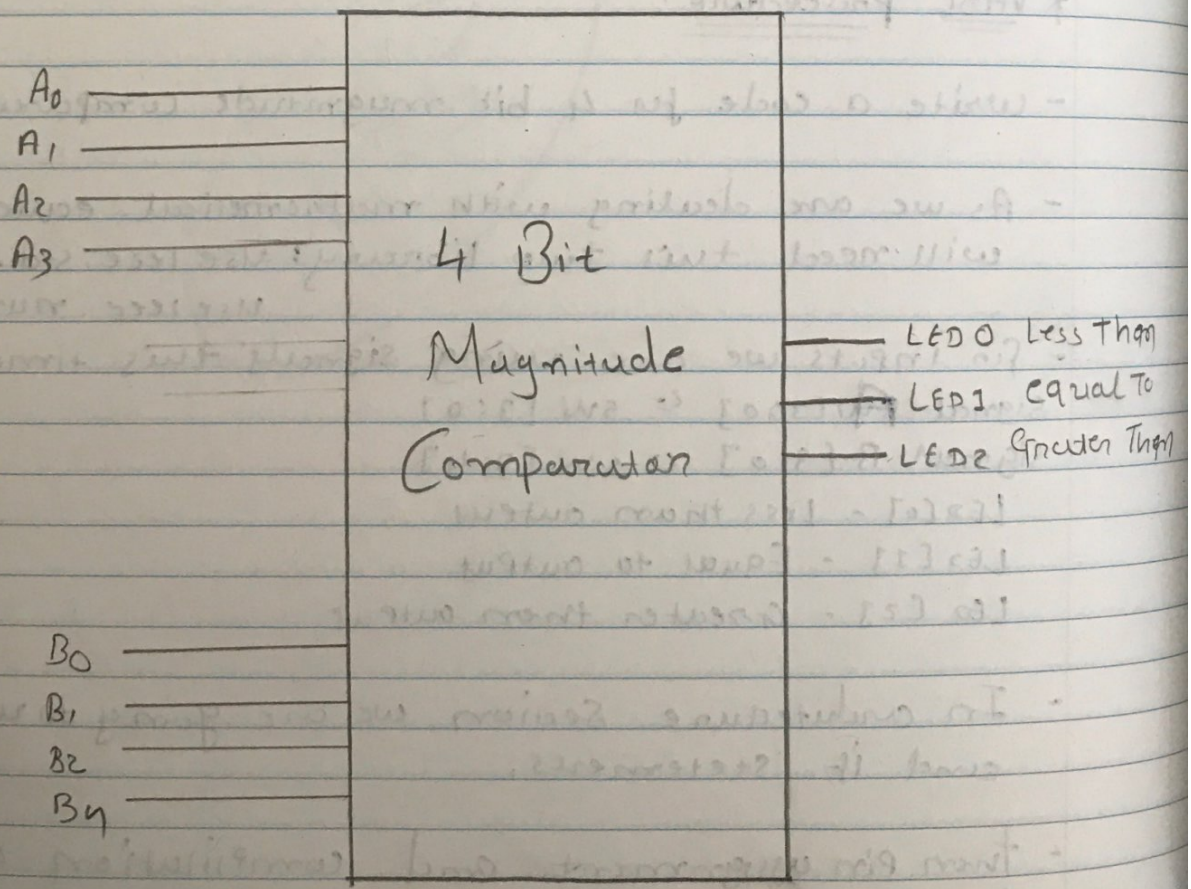
* VHDL Procedure:

- write a code for 4 bit magnitude comparator.
- As we are dealing with mathematical equations we will need two library: use ieee.std_logic_Usigned.all;
  use ieee.numeric_std.all;
- for inputs we are using signals this time.
  signal A[3:0] <= SW[3:0]
  signal B[3:0] <= SW[7:4]
  LED[0] - Less than output
  LED[1] - Equal to output
  LED[2] - Greater then output
- In architecture Section we are going to use Process and if statements.
- then pin assignment and compilation and download the code on board.

* Vlog Procedure.

- Enter the Vlog code for magnitude comparator.
- for library input [7:0]sw;
  output [2:0]LED;
- we are also using reg in this program. reg [3:0]A,B.
  reg [2:0]LED
- we are using always and if stement.
- then pinasignment compilation and then downloading program on board.

Circuit Diagram for magnitude Comparator:



Pin Assignment:

Input / Output Variable	Signal Name	FPGA Pin no.
SW[0] / A[0]	SW[0]	PIN - AB12
SW[1] / A[1]	SW[1]	PIN - AC12
SW[2] / A[2]	SW[2]	PIN - AF9
SW[3] / A[3]	SW[3]	PIN - AF10
SW[4] / B[0]	SW[5]	PIN - AD12
SW[5] / B[1]	SW[6]	PIN - AE11
SW[6] / B[2]	SW[7]	PIN - AC9
SW[7] / B[3]	SW[8]	PIN - AD10
LED[0]	LEDR[0]	PIN - V16
LED[1]	LEDR[1]	PIN - W16
LED[2]	LEDR[2]	PIN - V17
SLED[0]	HEX0[0]	PIN - AG26
SLED[1]	HEX0[1]	PIN - AE27
SLED[2]	HEX0[2]	PIN - AE28
SLED[3]	HEX0[3]	PIN - AG27
SLED[4]	HEX0[4]	PIN - AF28
SLED[5]	HEX0[5]	PIN - AG28
SLED[6]	HEX0[6]	PIN - AH28
SLED[7]		

Programs

* VHDL Program

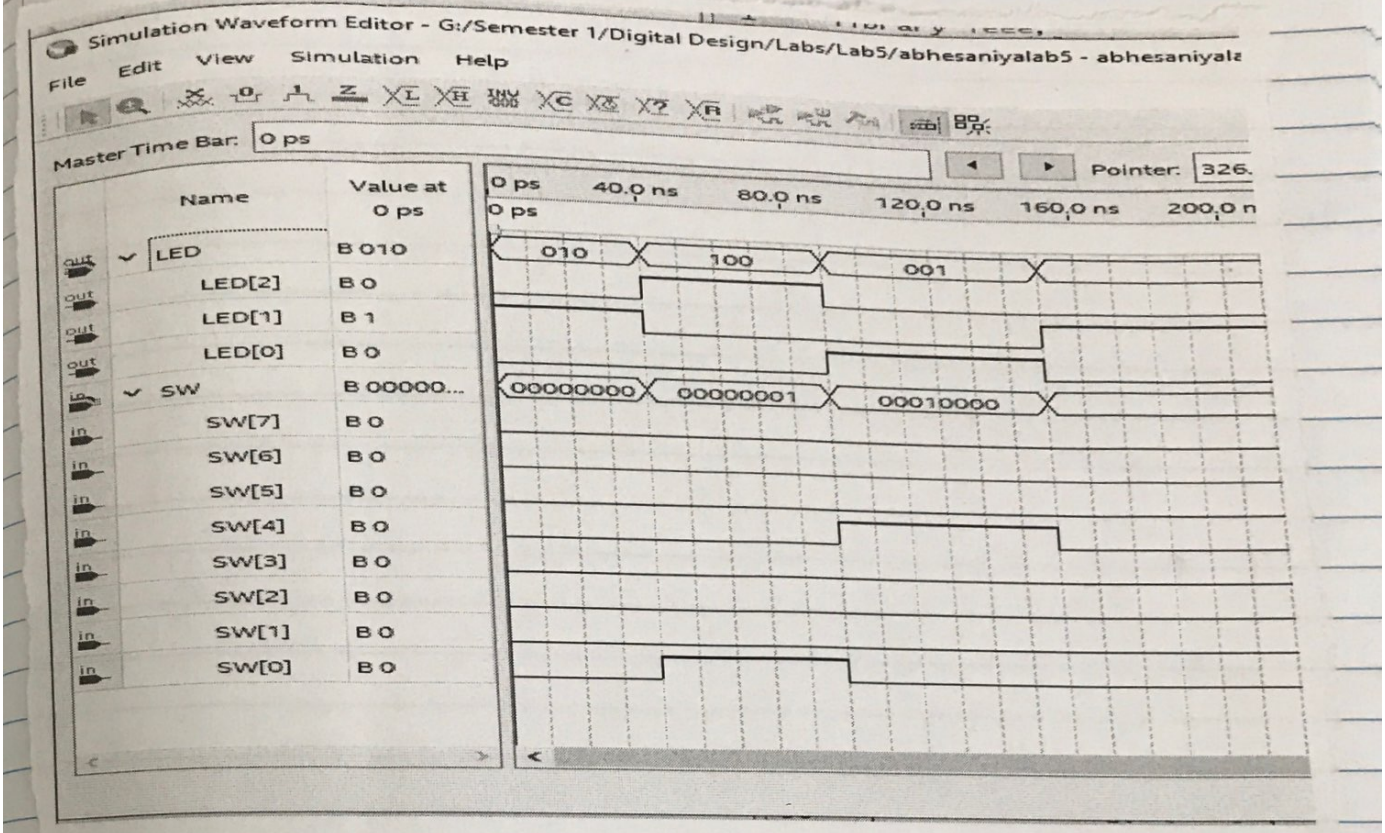
Design/Labs/Lab5/abhesaniyalab5 - abhesaniyalab5

g Tools Window Help

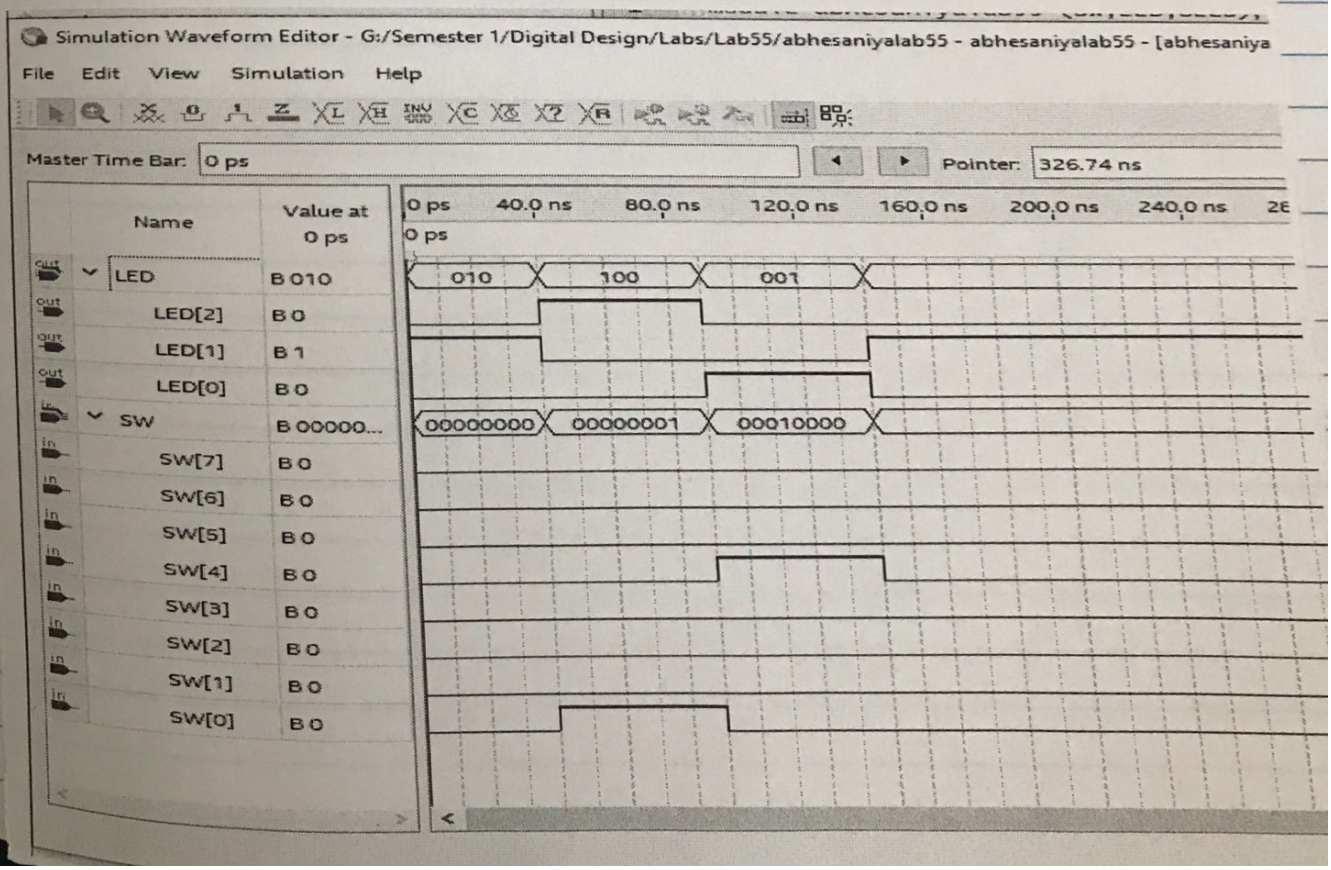
abhesaniyalab5.vhd*

```
1
2     library ieee;
3     use ieee.std_logic_1164.all;
4     use IEEE.STD_LOGIC_UNSIGNED.ALL;
5     use IEEE.numeric_std.ALL;
6
7
8     entity abhesaniyalab5 is
9     port(--EN:in std_logic;
10      SW : in std_logic_vector(7 downto 0);
11    LED : out std_logic_vector(2 downto 0);
12    SLED0: out std_logic_vector(6 downto 0)
13    );
14    end abhesaniyalab5;
15
16    architecture magnitudecomparator of abhesaniyalab5 is
17
18    signal A : std_logic_vector(3 downto 0) ;
19    signal B : std_logic_vector(3 downto 0) ;
20    signal GT : std_logic;
21    signal EQ : std_logic;
22    signal LT : std_logic;
23
24    begin
25
26        A <= SW(3 downto 0);
27        B <= SW(7 downto 4);
28
29
30
31      process(A,B)
32        begin
33                  if (A = B) then
34
35                  EQ <= '1';
36                  LT <= '0';
37                  GT <= '0';
38                  SLED0<="0110000";
39
40                  else if(A > B) then
41
42                  EQ <= '0';
43                  LT <= '0';
44                  GT <= '1';
45                  SLED0<="0100001";
46
47                  else if(A < B) then
48
49                  EQ <= '0';
50                  LT <= '1';
51                  GT <= '0';
52                  SLED0<="1110001";
53
54                  else
55
56                  EQ <= '0';
57                  LT <= '0';
58                  GT <= '0';
59                  SLED0<="1111111";
60
61                  end if;
62                  end if;
63                  end if;
64
65
66    LED(0) <= LT;
67    LED(1) <= EQ;
68    LED(2) <= GT;
69
70    end process;
71
72
73    end magnitudecomparator;
```

Output Wave form for VHDL



Output Wave form for Vlog



Vlog Program:

Design/Labs/Lab55/abhesaniyalab55 - abhesaniyalab55

File Edit View Project Assignments Processing Tools Window Help

abhesaniyalab55.v

```
module abhesaniyalab55 (SW,LED,SLED);

input [7:0]SW;
output[2:0]LED;
output[6:0]SLED;

//assign
reg [3:0] A,B;
reg [2:0] LED;
reg [6:0]SLED;

 always @(A or B)
 begin
        //{B,A} = SW;
        A = SW[3:0];
        B = SW[7:4];

        if (A < B)
          LED = 3'b001;

        else if (A > B)
          LED = 3'b100;

        else if (A == B)
          LED = 3'b010;

       else
          LED = 3'b000;

       end
 always @(A or B)
  begin
       if (A < B)
          SLED = 7'b1110001;
        else if (A > B)
          SLED = 7'b0100001;
        else if (A == B)
          SLED = 7'b0110000;
       else
          SLED = 7'b1111111;
       end

endmodule
```

Discussion

```
Running Quartus Prime Analysis & Synthesis
Command: quartus_map --read_settings_files=on --write_settings_files=off abhesaniyalab55 -c abhesaniyalab55
18236 Number of processors has not been specified which may cause overloading on shared machines.  Set the global assignment NUM_PARALLEL_PROCESSORS in your
20030 Parallel compilation is enabled and will use 6 of the 6 processors detected
10170 Verilog HDL syntax error at abhesaniyalab55.v(19) near text: "'1";  expecting ";". Check for and fix any syntax errors that appear immediately before o
10170 Verilog HDL syntax error at abhesaniyalab55.v(22) near text: "'0";  expecting ";". Check for and fix any syntax errors that appear immediately before o
10170 Verilog HDL syntax error at abhesaniyalab55.v(25) near text: "'0";  expecting ";". Check for and fix any syntax errors that appear immediately before o
10112 Ignored design unit "abhesaniyalab55" at abhesaniyalab55.v(1) due to previous errors
12021 Found 0 design units, including 0 entities, in source file abhesaniyalab55.v
Quartus Prime Analysis & Synthesis was unsuccessful. 4 errors, 1 warning
293001 Quartus Prime Full Compilation was unsuccessful. 6 errors, 1 warning
```

- I am getting an error, if statement in Vlog program is not allowing me to write second statement in side if and else if loop.

- so from next time i am going to use some diferent statement and finding solution for that also.

MSB	LSB	MSB	LSB				
A_n - - -	A_0	B_n - - -	B_0	L_0	L_1	L_2	Hex 0
SW_3	SW_0	SW_4	SW_5	LED0	LED1	LED2	
				LT	EQ	GT	

Demo

A $S_3 - S_0$ L

B $S_8 - S_5$ L

7seg. L_2 L_1 L_0

L > = <

G E L

A	1000	1010	1101
B	0111	1010	1110
	G	E	L
	100	010	001