Lab 6-1

date: 19th February, 2020

Lab title: Half Adder

Recorded and Submitted By: Bhesaniya Abhishek

Objective To write VHDL and Vlog Program for half adder and test it using DESoC-I FBGA Board.

Implementation:

Half adder is circuit which process two input and gives two output sum and carry.

VHDL procedure:

- write a code for half adder.

- we will use signal for both input and output.

```
Signal A : Std_logic_Vector (3 downto 0);
Signal B : Std_logic_Vector (7 downto 4);
Signal Sum : Std_logic_Vector (3 downto 0);
Signal CARRY : Std_logic_Vector (7 downto 4);

A <= SW(3 downto 0);
B <= SW(7 downto 4);

Sum(0) <= A(0) XOR B(4);
CARRY(4) <= A(0) AND B(4);
```

and like wise

=) Then use process statment for if else loop in architecture.

=) After writing code compile the program and do Pin assign Then

- after that use programer and load your program into DE SOC-1 board and check the output value.

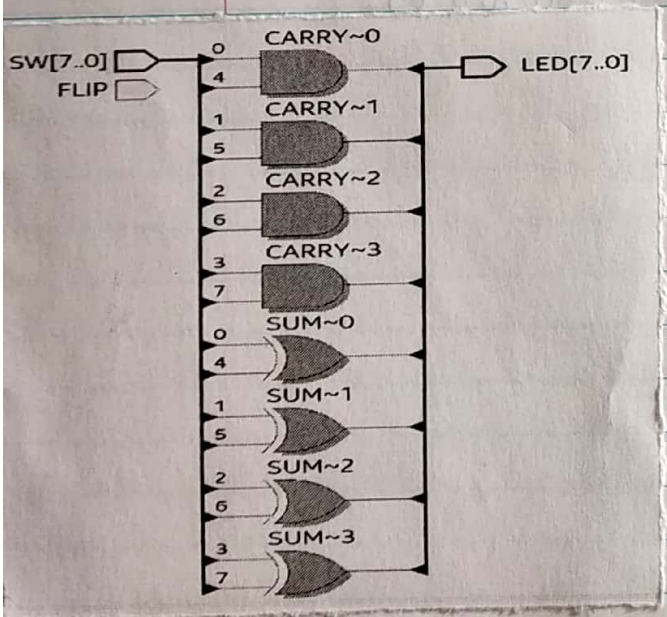
Vlog procedure:

- This time in Vlog we are using wire for input and reg for output.

```
wire [3:0] A = SW[3:0];
wire [7:4] B = SW[7:4];
reg [3:0] SUM;
reg [7:4] CARRY;
```

- use always statment for if else loop.
- after writing code compile and do pinplanner.
- Then use programer to load your program and check output.

Circuit Diagram:



Truth Table

SW7	SW6	SW5	SW4	SW3	SW2	SW1	SW0

Truth Table

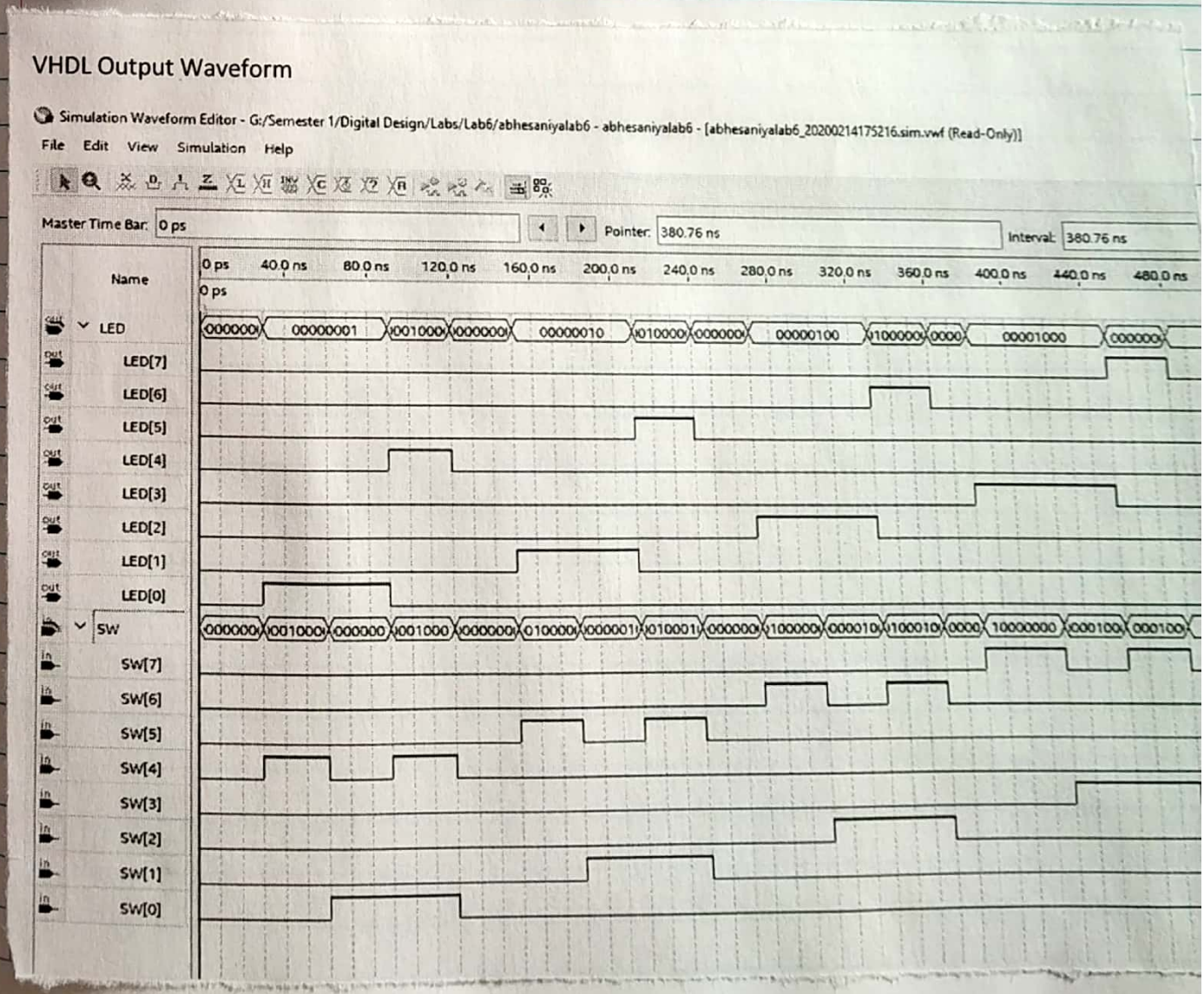
SW_4	SW_0	S_0	C_4
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

SW_5	SW_1	S_1	C_5
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

SW_6	SW_2	S_2	C_6
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

SW_7	SW_3	S_3	C_7
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

VHDL Output Waveform



PIN - Assignment

in FLIP	Input	PIN_AE12
out LED[7]	Output	PIN_W20
out LED[6]	Output	PIN_Y19
out LED[5]	Output	PIN_W19
out LED[4]	Output	PIN_W17
out LED[3]	Output	PIN_V18
out LED[2]	Output	PIN_V17
out LED[1]	Output	PIN_W16
out LED[0]	Output	PIN_V16
out SLED0[6]	Output	PIN_AE26
out SLED0[5]	Output	PIN_AE27
out SLED0[4]	Output	PIN_AE28
out SLED0[3]	Output	PIN_AG27
out SLED0[2]	Output	PIN_AF28
out SLED0[1]	Output	PIN_AG28
out SLED0[0]	Output	PIN_AH28
out SLED1[6]	Output	PIN_AJ29
out SLED1[5]	Output	PIN_AH29
out SLED1[4]	Output	PIN_AH30
out SLED1[3]	Output	PIN_AG30
out SLED1[2]	Output	PIN_AF29
out SLED1[1]	Output	PIN_AF30
out SLED1[0]	Output	PIN_AD27
out SLED2[6]	Output	PIN_AB23
out SLED2[5]	Output	PIN_AE29
out SLED2[4]	Output	PIN_AD29
out SLED2[3]	Output	PIN_AC28
out SLED2[2]	Output	PIN_AD30
out SLED2[1]	Output	PIN_AC29
out SLED2[0]	Output	PIN_AC30
out SLED3[6]	Output	PIN_AD26
out SLED3[5]	Output	PIN_AC27
out SLED3[4]	Output	PIN_AD25
out SLED3[3]	Output	PIN_AC25
out SLED3[2]	Output	PIN_AB28
out SLED3[1]	Output	PIN_AB25
out SLED3[0]	Output	PIN_AB22
out SLED4[6]	Output	PIN_AA24
out SLED4[5]	Output	PIN_Y23
out SLED4[4]	Output	PIN_Y24
out SLED4[3]	Output	PIN_W22
out SLED4[2]	Output	PIN_W24
out SLED4[1]	Output	PIN_V23
out SLED4[0]	Output	PIN_W25
in SW[7]	Input	PIN_AD10
in SW[6]	Input	PIN_AC9
in SW[5]	Input	PIN_AE11
in SW[4]	Input	PIN_AD12
in SW[3]	Input	PIN_AF10
in SW[2]	Input	PIN_AF9
in SW[1]	Input	PIN_AC12
in SW[0]	Input	PIN_AB12

VHDL Program:

VHDL program

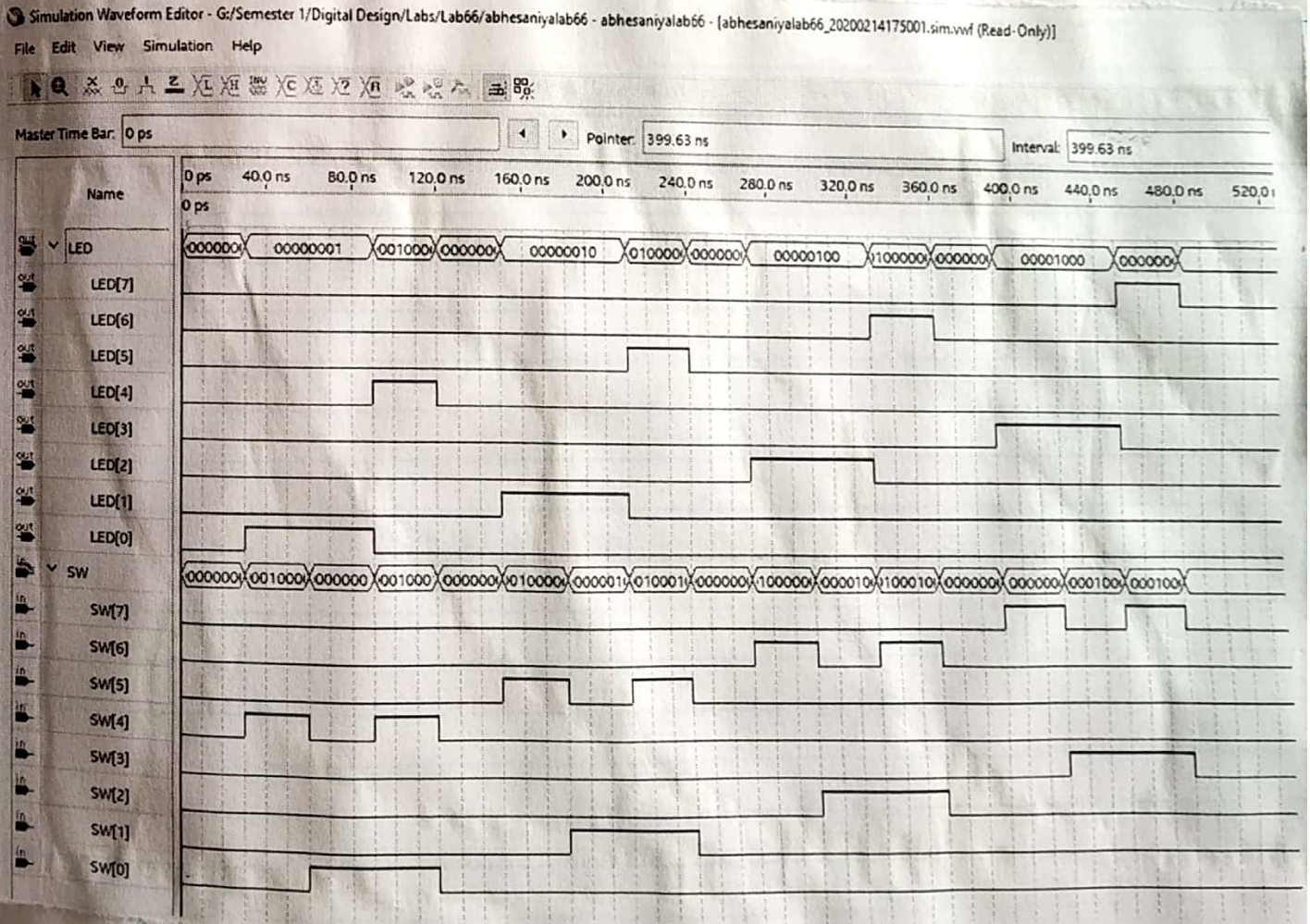
abhesaniyalab6.vhd

```
1   library ieee;
2   use ieee.std_logic_1164.all;
3
4
5   entity abhesaniyalab6 is
6   port(--EN:in std_logic;
7     SW : in std_logic_vector(7 downto 0);
8     FLIP : in std_logic;
9   LED : out std_logic_vector(7 downto 0);
10  SLED0: out std_logic_vector(6 downto 0);
11  SLED1: out std_logic_vector(6 downto 0);
12  SLED2: out std_logic_vector(6 downto 0);
13  SLED3: out std_logic_vector(6 downto 0);
14  SLED4: out std_logic_vector(6 downto 0)
15  --SLED3: out std_logic_vector(6 downto 0)
16  );
17  end abhesaniyalab6;
18
19  architecture halfadder of abhesaniyalab6 is
20
21  signal A : std_logic_vector(3 downto 0) ;
22  signal B : std_logic_vector(7 downto 4) ;
23  signal SUM : std_logic_vector(3 downto 0) ;
24  signal CARRY : std_logic_vector(7 downto 4) ;
25
26
27  begin
28
29      A <= SW(3 downto 0);
30      B <= SW(7 downto 4);
31
32      SUM(0) <= A(0) XOR B(4);
33      CARRY(4) <= A(0) AND B(4);
34
35      SUM(1) <= A(1) XOR B(5);
36      CARRY(5) <= A(1) AND B(5);
37
38      SUM(2) <= A(2) XOR B(6);
39      CARRY(6) <= A(2) AND B(6);
40
41      SUM(3) <= A(3) XOR B(7);
42      CARRY(7) <= A(3) AND B(7);
43
44      LED(0) <= SUM(0);
45      LED(1) <= SUM(1);
46      LED(2) <= SUM(2);
47      LED(3) <= SUM(3);
48      LED(4) <= CARRY(4);
49      LED(5) <= CARRY(5);
50      LED(6) <= CARRY(6);
51      LED(7) <= CARRY(7);
52
```

```
54  process (SUM , CARRY, FLIP)
55  begin
56
57    if ( FLIP = '0') then
58      SLED4 <= "0100100";
59
60        if (SUM(0) = '0') then
61            SLED0 <="0000001";
62        else if (SUM(0) = '1') then
63            SLED0 <="1001111";
64        end if;
65        end if;
66
67        if (SUM(1) = '0') then
68         SLED1 <="0000001";
69        else if (SUM(1) = '1') then
70         SLED1 <="1001111";
71        end if;
72        end if;
73
74        if (SUM(2) = '0') then
75         SLED2 <="0000001";
76        else if (SUM(0) = '1') then
77         SLED2 <="1001111";
78        end if;
79        end if;
80
81        if (SUM(3) = '0') then
82         SLED3 <="0000001";
83        else if (SUM(3) = '1') then
84         SLED3 <="1001111";
85        end if;
86        end if;
87
88
89        else if(FLIP = '1')then
90
91
92          SLED4 <= "0110001";
93          if (CARRY(4) = '0') then
94              SLED0 <="0000001";
95          else if (CARRY(4) = '1') then
96              SLED0 <="1001111";
97          end if;
98          end if;
99
100
101         if (CARRY(5) = '0') then
102             SLED1 <="0000001";
103         else if (CARRY(5) = '1') then
104             SLED1 <="1001111";
105         end if;
106         end if;
107
```

```
108
109             if (CARRY(6) = '0') then
110                 SLED2 <="0000001";
111             else if (CARRY(6) = '1') then
112                 SLED2 <="1001111";
113             end if;
114             end if;
115
116
117             if (CARRY(7) = '0') then
118                 SLED3 <="0000001";
119             else if (CARRY(7) = '1') then
120                 SLED3 <="1001111";
121             end if;
122             end if;
123
124         end if;
125         end if;
126
127     end process;
128
129
130
131 end halfadder;
132
```

Vlog output waveform:



Vlog - Program

Vlog Program

abhesaniyalab66.v

```
1   module abhesaniyalab66 (SW,FLIP,LED,SLED0,SLED1,SLE
2
3   input [7:0]SW;
4   input FLIP;
5   output[7:0]LED;
6   output[6:0]SLED0;
7   output[6:0]SLED1;
8   output[6:0]SLED2;
9   output[6:0]SLED3;
10  output[6:0]SLED4;
11
12  reg [7:0]LED;
13  reg [6:0]SLED0;
14  reg [6:0]SLED1;
15  reg [6:0]SLED2;
16  reg [6:0]SLED3;
17  reg [6:0]SLED4;
18  //assign
19  wire [3:0] A = SW[3:0];
20  wire [7:4] B = SW[7:4];
21  reg [3:0] SUM;
22  reg [7:4]CARRY;
23
24
25  always @(A or B or FLIP)
26  begin
27
28            SUM[0] = A[0] ^ B[4];
29            SUM[1] = A[1] ^ B[5];
30            SUM[2] = A[2] ^ B[6];
31            SUM[3] = A[3] ^ B[7];
32
33            CARRY[4] = A[0] & B[4];
34            CARRY[5] = A[1] & B[5];
35            CARRY[6] = A[2] & B[6];
36            CARRY[7] = A[3] & B[7];
37
38    LED[3:0] = SUM;
39    LED[7:4] = CARRY;
40
41       if (FLIP == 0)
42          begin
43             SLED4 = 7'b0100100;
44             if (SUM[0] == 0)
45               SLED0 = 7'b0000001;
46             else if (SUM[0] == 1)
47               SLED0 = 7'b1001111;
48
49             if (SUM[1] == 0)
50               SLED1 = 7'b0000001;
51             else if (SUM[1] == 1)
52               SLED1 = 7'b1001111;
```

```
        if (SUM[2] == 0)
          SLED2 = 7'b0000001;
        else if (SUM[2] == 1)
          SLED2 = 7'b1001111;

        if (SUM[3] == 0)
          SLED3 = 7'b0000001;
        else if (SUM[3] == 1)
          SLED3 = 7'b1001111;
      end

    else if (FLIP == 1)
    begin

        SLED4 = 7'b0110001;
        if (CARRY[4] == 0)
          SLED0 = 7'b0000001;
        else if (CARRY[4] == 1)
          SLED0 = 7'b1001111;

        if (CARRY[5] == 0)
          SLED1 = 7'b0000001;
        else if (CARRY[5] == 1)
          SLED1 = 7'b1001111;

        if (CARRY[6] == 0)
          SLED2 = 7'b0000001;
        else if (CARRY[6] == 1)
          SLED2 = 7'b1001111;

        if (CARRY[7] == 0)
          SLED3 = 7'b0000001;
        else if (CARRY[7] == 1)
          SLED3 = 7'b1001111;

    end
  end

  endmodule
```

5

Discussion:

	I/P		O/P		
Switch	L	L	L_7 - L_4 L_3 - L_0	H_4	H_3 - H_0
S_9	S_8 - S_5	S_3 - S_0	C_7 - C_4 S_4 - S_0	S C	S_3 - S_0
	B_4 - B_0	A_4 - A_0			

A 1011

B 1101

0110 S

1001 C