EXPERIMENT - 2

AIM : Design and simulation of full adder circuit using Verilog HDL
Hardware implementation of full adder circuit using logic gates

COMPONENT / SOFTWARE USED :

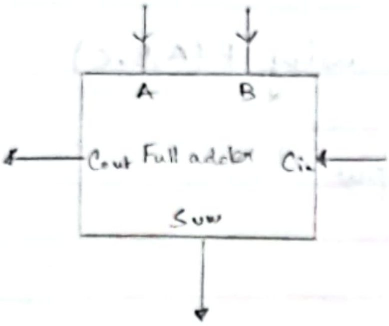
Component / Software	Specification
• IC's	7408 , 7432 , 7486
• Bread board , power supply, LED's, resistors, switches, connecting wires	As per requirement
• Software's used	Vivado 2016.1

THEORY :

Full adder :

A full adder is a combinational circuit that forms the arithmetic sum of three bits. It consists of three inputs and two outputs. Two of the input variables, denoted by A and B represent the two significant bits to be added. The third input Cin represents the carry from the previous lower significant position. Two outputs are required because the arithmetic sum of three binary digits ranges in value from 0 to 3 and the binary representation of 2 or 3 needs two bits. The two outputs are designated by Sum and Cout for output carry.

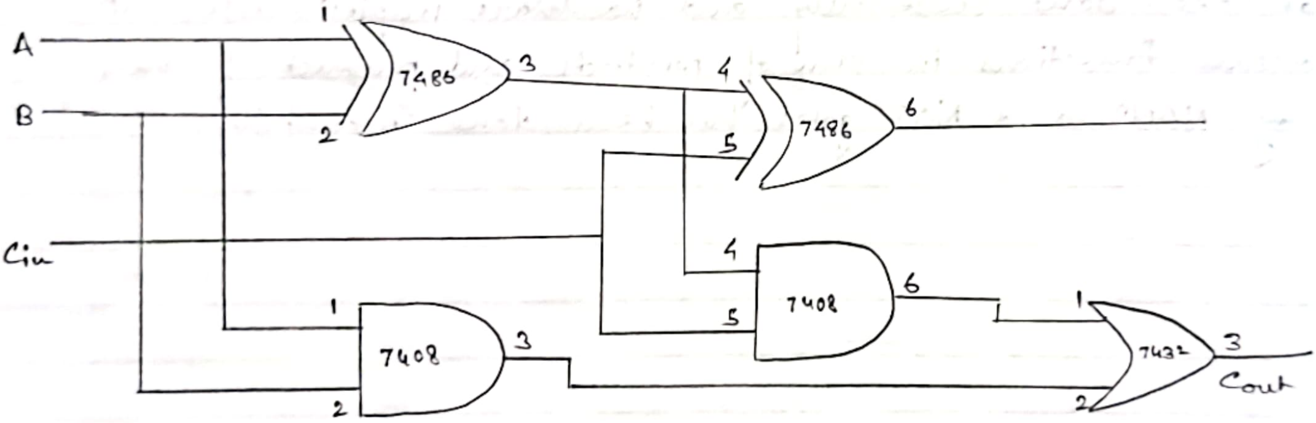
When all input bits are 0, The Sum output is equal to 1 when only one input is equal to 1 or when all three inputs are equal to 1. The Cout output has a carry of 1 if two or three inputs are equal to 1.



Block diagram of full adder

Input			Output	
A	B	Cin	Cout	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Truth table of the full adder



Logic diagram of the full adder

The truth table of the full adder is listed in table 2.1. The eight rows under the input variables designate all possible combinations of the three variables. The block diagram of the full adder is shown in fig.2.1 Then find out the Boolean expression for Sum and Cout using Boolean Algebra. In fig. 2.2, the logic diagram of full adder is shown

$$Sum(A,B,C_{in}) = \sum m(1,2,4,7) = \bar{A}\bar{B}C_{in} + \bar{A}B\bar{C}_{in} + A\bar{B}\bar{C}_{in} + ABC_{in}$$
$$= (A \oplus B) \oplus C_{in}$$

$$C_{out}(A,B,C_{in}) = \sum m(3,5,6,7) = \bar{A}BC_{in} + A\bar{B}C_{in} + AB\bar{C}_{in} + ABC_{in}$$
$$= (\bar{A}B + A\bar{B}) \cdot C_{in} + AB(\bar{C}_{in} + C_{in})$$
$$= (A \oplus B) \cdot C_{in} + (A \cdot B)$$

PROCEDURE

For software simulation

(a) Create a module with required number of variables and mention its input/output.

(b) Write the description of given boolean function using operators or by using built in primitive gates.

(c) Synthesize to create RTL Schematic.

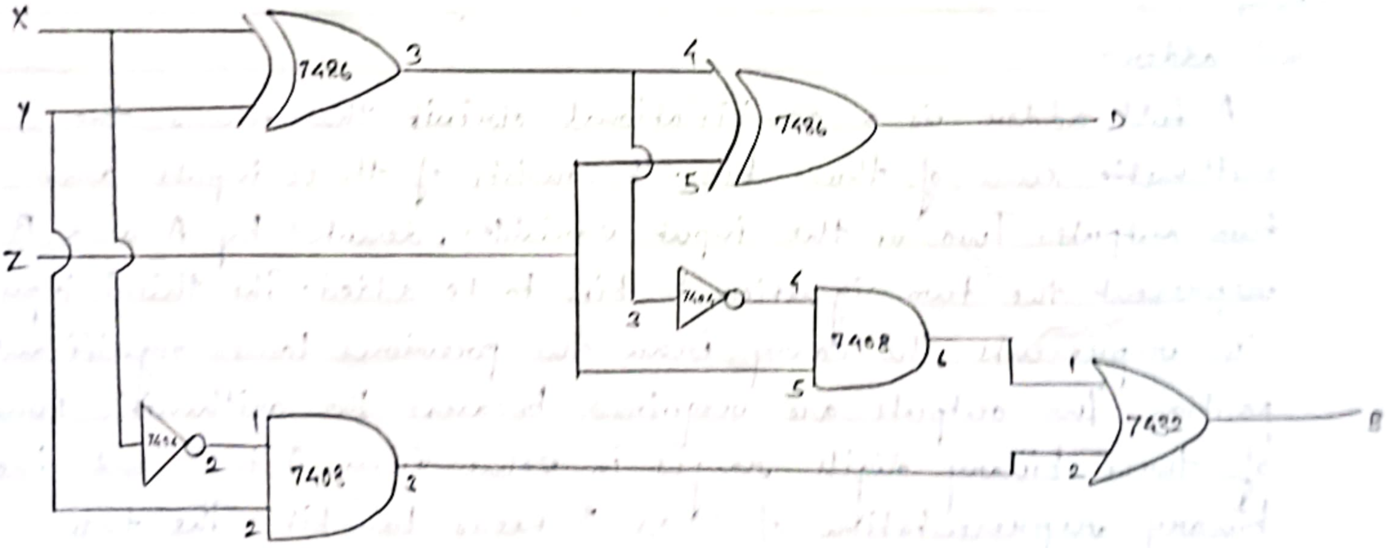
(d) Create another module referred to as test bench to verify the functionality and to obtain the waveforms of input and output.

(e) Follow the steps required to stimulate the design and compare the obtained output with the corresponding truth table.

(f) Take the screenshots of the RTL schematic and the stimulated waveforms.

Inputs			Outputs	
X	Y	Z	D	B
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Truth table of the full subtractor



For hardware implementation

(a) Turn off the power of the trainer kit before constructing any circuit.

(b) Connect power supply (+5V DC) pin and ground pin to respective pins of the trainer kit.

(c) Place the IC's properly on the bread board in trainer kit.

(d) Connect VCC and GND pins on each chip to the power supply and ground bus strips on the bread board.

(e) Connect the input and output pins of chips to the input switches and output LEDs respectively in the trainer kit.

(f) Check the connection before you turn on the power.

(g) Apply various combinations of inputs according to truth table and observe outputs of LEDS.

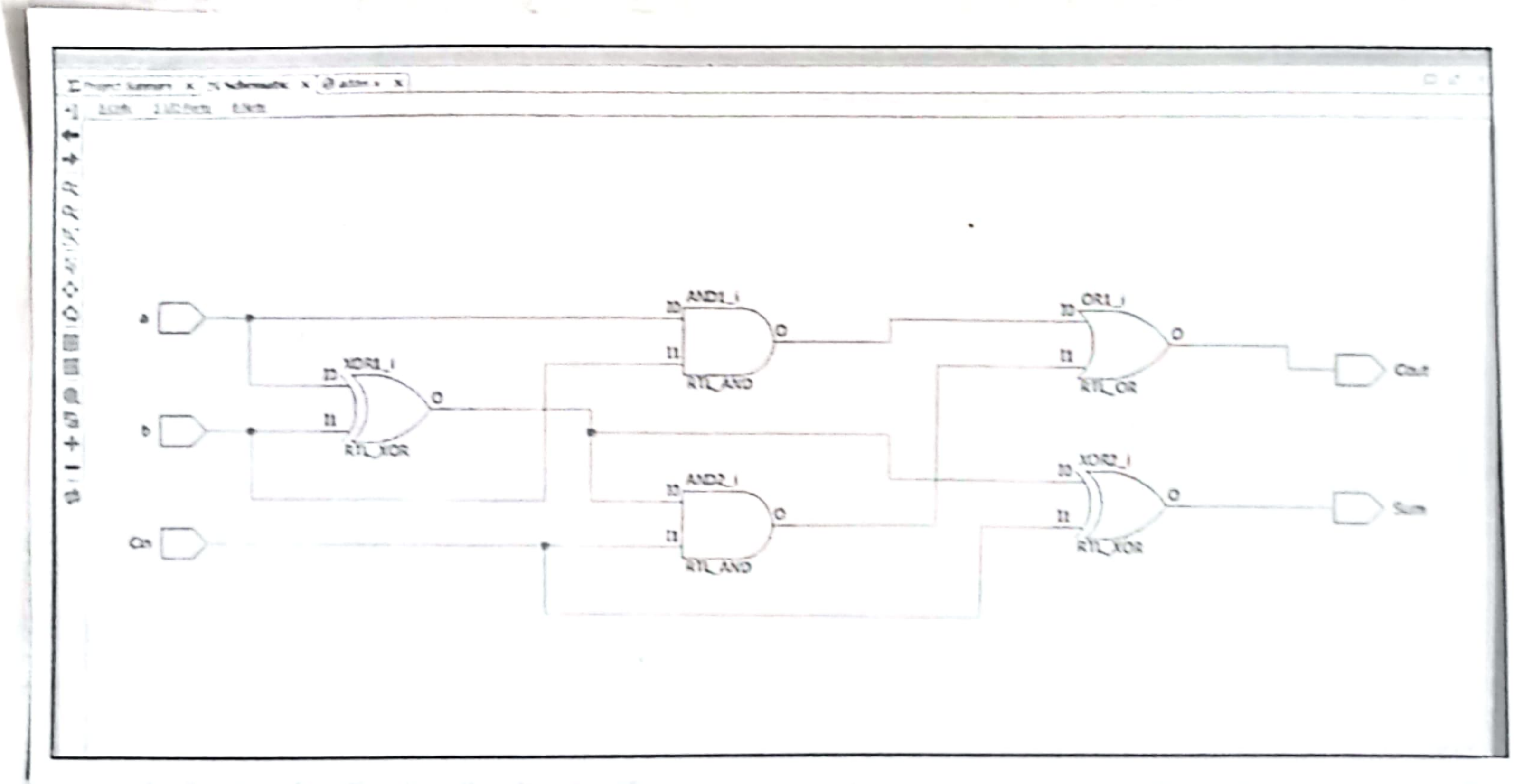
■ HDL Code

```
module Full_Adder (output Cout, output Sum, input A, input B, input Cin);
        wire w0, w2, w3 ;
        xor XOR1 (w0, A, B);
        and AND1 (w1, A, B);
        xor XOR2 (Sum, w0, Cin);
        and AND2 (w2, w0, Cin);
        or OR1 (Cout, w1, w2);
endmodule
```

■ ~~Conclusion~~

~~The truth table of the~~

PTO →


Project Summary
Schematic
a
b
Cin
XOR1_i
I0
I1
O
RTL_XOR
AND1_i
I0
I1
O
RTL_AND
AND2_i
I0
I1
O
RTL_AND
OR1_i
I0
I1
O
RTL_OR
XOR2_i
I0
I1
O
RTL_XOR
Cout
Sum

Full Adder

- Test bench code

```
module full_add_tb ;
      reg a,b,cin ;
      wire sum,cout ;

  full_add f1(.a(a),.b(b),.cin(cin),.sum(sum),.cout(cout));

  initial begin $dumpfile ("full-tb.vcd"); $dumpvars(); end

  initial begin a=1'b1; #1; a=1'b0; #10 $stop(): end
  initial begin b=1'b1 ; forever #2 b=-b ; end
  initial begin cin=1'b1; forever #1 cin=-cin ;#10 $stop(); end

   initial begin $monitor
endmodule
```

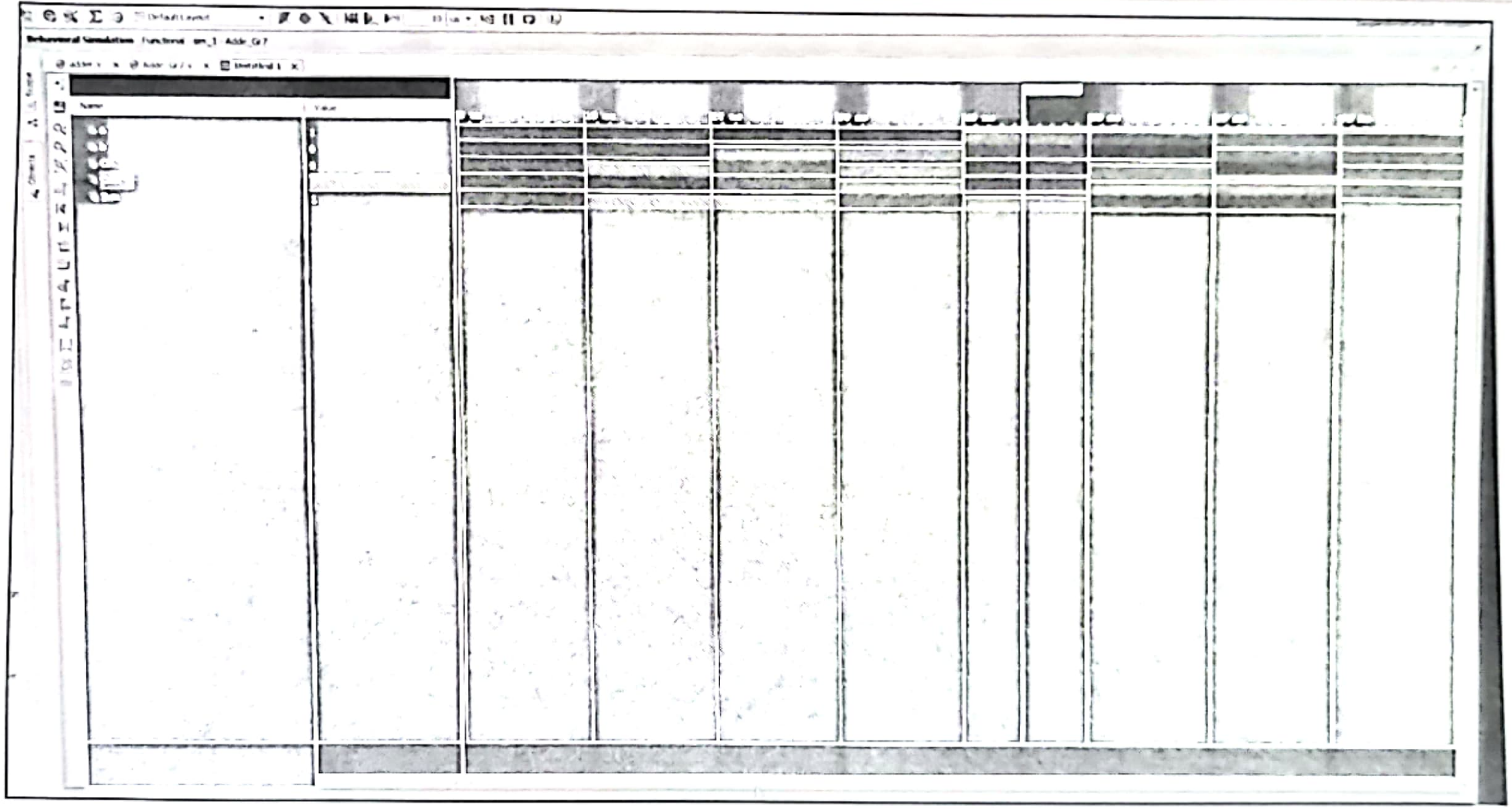
- Design problem

Design and simulation of full subtractor circuit using Verilog HDL.
Hardware implementation of full subtractor circuit.

- Solution:

Full subracter is a combinational circuit that performs a subtraction of two bits, taking into account borrowing from the lower significant stage. The circuit has three inputs named minuend (X), subtrahend (Y), and previous borrow (Z) and two outputs, called different (D) and a borrow (B)

Then find out the boolean expression for difference (D) and a borrow (B) using Boolean algebra



Full Adder Waveform

TEST BENCH CODE

```
include "Full_Subtractor.v"
module Full_Subtractor_tb;
wire D,B;
reg X,Y,Z;
Full_Subtractor_ Instance0 (D,B;X,Y,Z);
    initial begin
          X=0; Y=0; Z=0;
   #1     X=0; Y=0; Z=1;
   #1     X=0; Y=1; Z=0;
   #1     X=0; Y=1; Z=1;
   #1     X=1; Y=0; Z=0;
   #1     X=1; Y=0; Z=1;
   #1     X=1; Y=1; Z=0;
   #1     X=1; Y=1; Z=1;
   end
initial begin
     $monitor ("%t, X = %d | Y = %d | Z = %d | B = %d | D = %d", $time,
                                                            X;Y,Z,B,D);

     $dumpfile ("dump.vcd");
     $dumpvars();
  end
  endmodule
```

$$D(X,Y,Z) = \sum m(1,2,4,7) = \bar{X}\bar{Y}Z + \bar{X}Y\bar{Z} + X\bar{Y}\bar{Z} + XYZ$$

$$= (X \oplus Y) \oplus Z$$

$$B(X,Y,Z) = \sum m(1,2,3,7) = \bar{X}\bar{Y}Z + \bar{X}Y\bar{Z} + \bar{X}YZ + XYZ$$

$$= \bar{X}Y(\bar{Z}+Z) + Z(\bar{X}\bar{Y} + XY)$$

$$= \bar{X}Y + Z(\overline{X \oplus Y})$$

■ CONCLUSION

The truth table of the full adder is verified in all the three modelling styles and its RTL schematic is generated. Design and simulation of full adder and full subtractor has been done successfully.

Teacher's Signature : 31/3/2023