DIGITAL CIRCUITS LAB

EXPERIMENT-7

Introduction to VHDL and CPLD

Shashi prabha

200020043

Aim: To write VHDL code for full adder in all 3 modelling styles i.e

- Dataflow
- Behavioral
- Structural

Summary of the experiment:

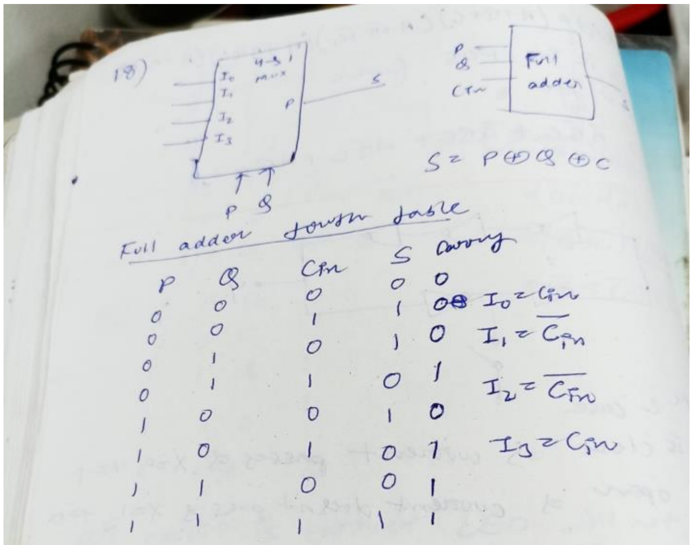
In this experiment we have to write the VHDL code for full adder in all three modelling styles i.e dataflow, behavioral, structural and observe the gate level netlist in the quartus software and emulated the CPLD board as full adder from from the VHDL code written for all three styles.

Components used:

- CPLD board
- Computer
- USB cable etc.

Design Procedure:

Truth Table (Full adder)



$S = P \oplus Q \oplus C$

Full adder truth table

P	Q	Cin	S	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$I_0 = C_{in}$

$I_1 = \overline{C_{in}}$

$I_2 = \overline{C_{in}}$

$I_3 = C_{in}$

HDL code (for CPLD experiments):

Dataflow :-

```
library ieee;

use ieee.std_logic_1164.all;

entity Full_adder is

port   (  a, b, c: in bit;

          s, c : out bit   );

end Full_adder;

architecture dataflow of Full_adder is

begin

s <= a xor b xor c;

c <= (a and b) or ((a xor b) and c);

end dataflow;
```

Behavioral-

```
library ieee;
use ieee.STD_LOGIC_1164.ALL;
use ieee.STD_LOGIC_ARITH.ALL;
use ieee.STD_LOGIC_UNSIGNED.ALL;

entity fulladder_B is
    Port (a,b,c : in std_logic;
s,c : out std_logic;

 end fulladder_B;
architecture Behavioral of fulladder_B is
begin
 process(a,b,c)
 begin
 if(a='0' and b='0' and c='0')
then
         s<='0' ;
         c<='0' ;
 elsif ( a='0' and b='0' and c='1')
then
         s<='1' ;
         c<='0' ;
  elsif ( a='0' and b='1' and c='0')
then
         s<='1' ;
         c<='0' ;
  elsif ( a='0' and b='1' and c='1')
then
         s<='0' ;
         c<='1' ;
 elsif (  a='1' and b='0' and c='0')
```

```
then
         s<='1' ;
         c<='0' ;
   elsif ( a='1' and b='0' and c='1')
 then
         s<='0' ;
         c<='1' ;
   elsif ( a='1' and b='1' and c='0')
then
         s<='0' ;
         c<='1' ;
 else
         s<='1' ;
         c<='1' ;
 end if ;
 end process ;
end Behavioral;
```

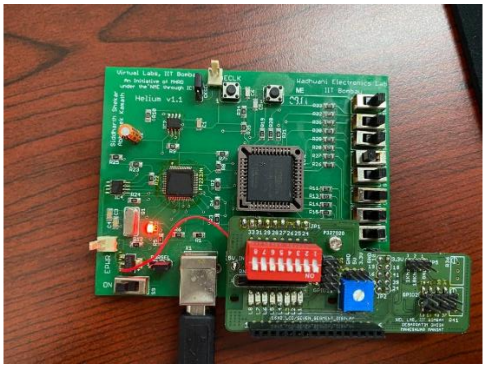
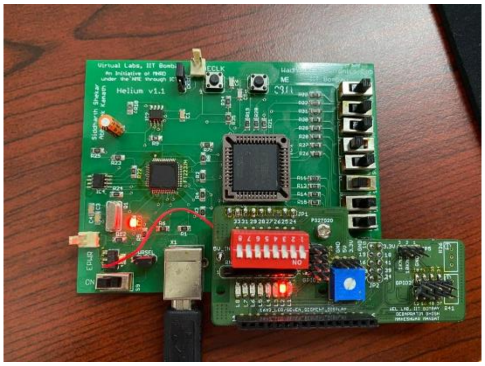
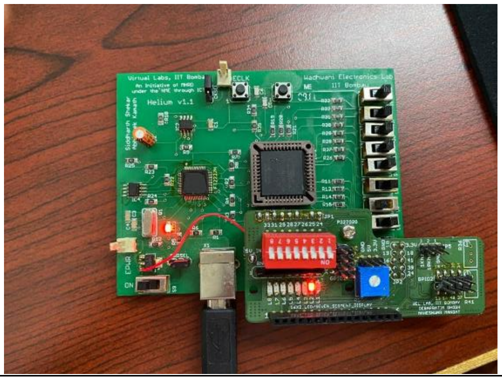
Structural-

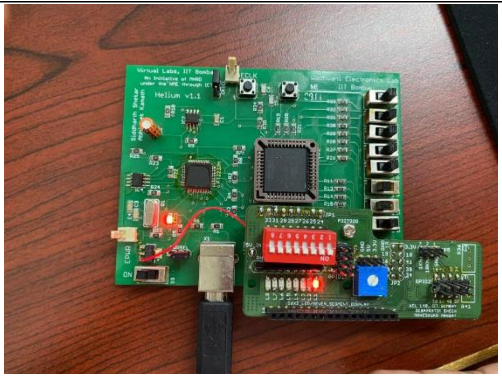
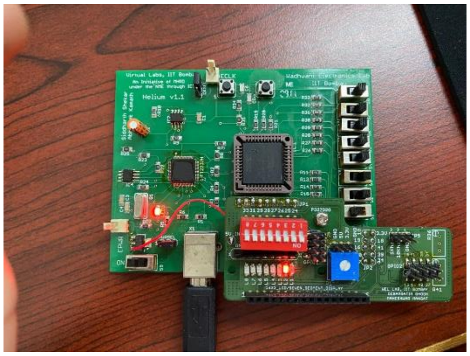
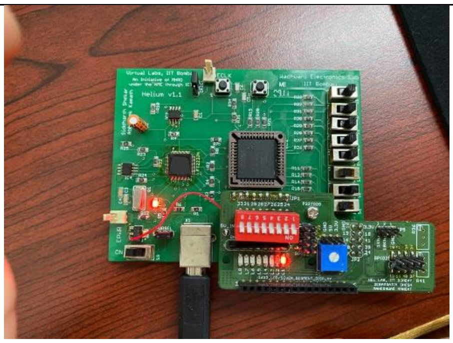
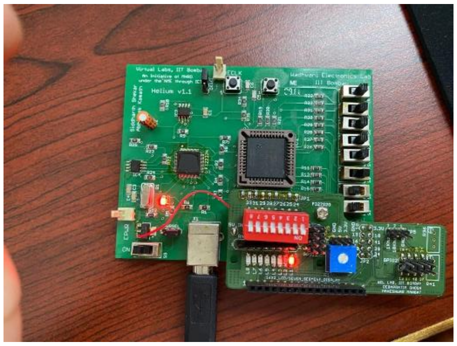
```
library ieee;
use ieee.std_logic_1164.all;
entity fulladder_S is
port(
 X1, X2, Cin : in std_logic;
S, Cout : out std_logic
 );
end fulladder_S;


architecture structural of fulladder_S is
signal a1, a2, a3: std_logic;
begin
a1 <= X1 xor X2;
a2 <= X1 and X2;
```

```
a3 <= a1 and Cin;
Cout <= a2 or a3;
S <= a1 xor Cin;

end structural;
```

Circuit and Simulation Snapshots:

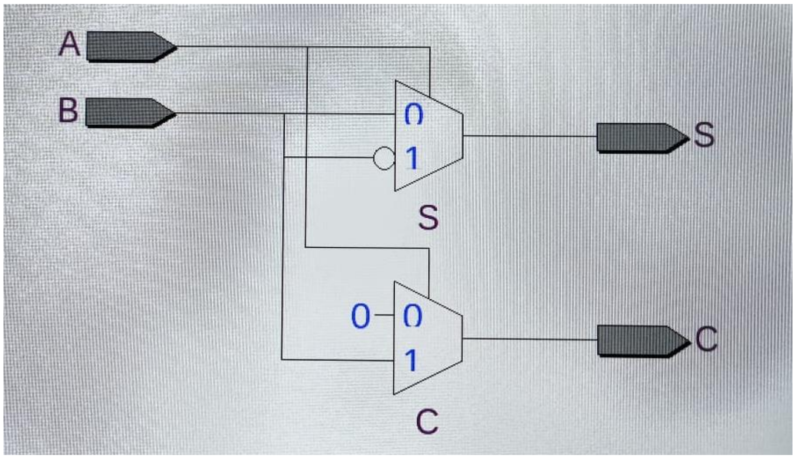
FULL ADDER

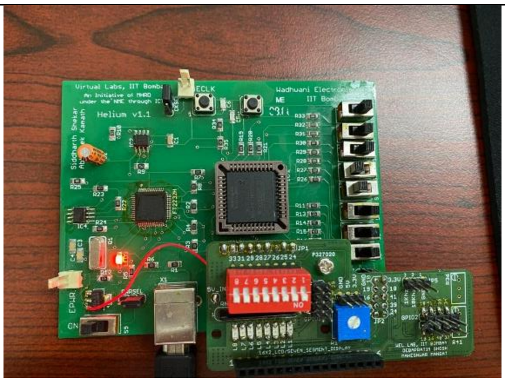
X	Y	Cin	S	Cout or carry	
0	0	0	0	0	
0	0	1	1	0	
0	1	0	1	0	

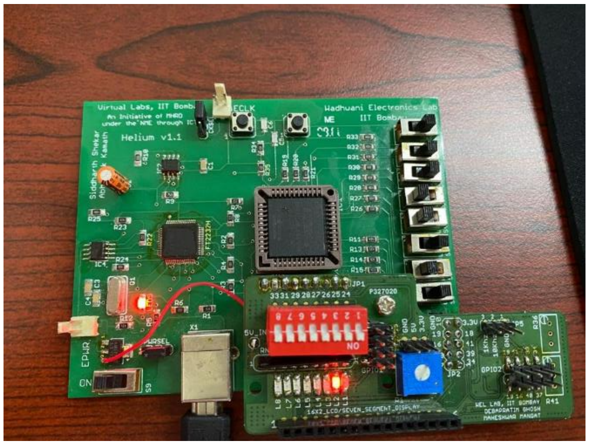
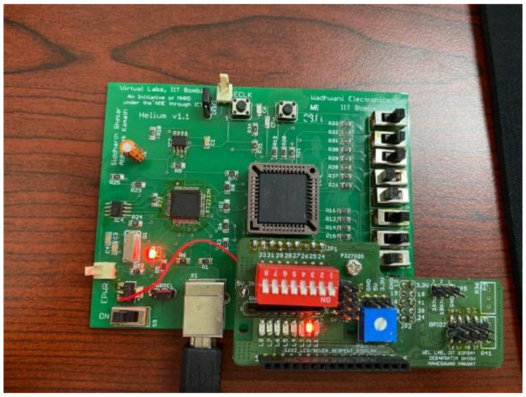
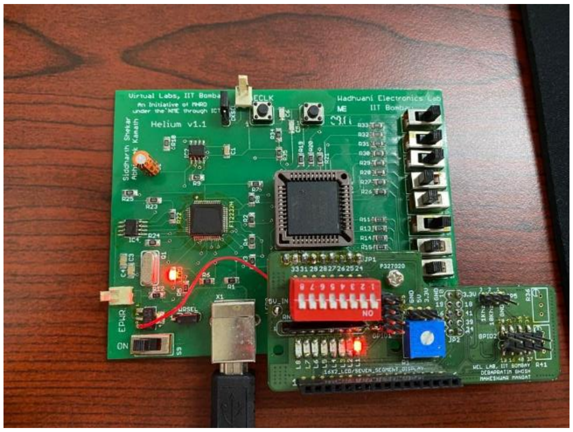
0	1	1	0	1	
1	0	0	1	0	
1	0	1	0	1	
1	1	0	0	1	

1	1	1	1	1	

HALF ADDER



X	Y	C	S	IMAGES
0	0	0	0	

0	1	0	1	
1	0	0	1	
1	1	1	0	

Results and Discussions:

In this lab we were introduced to VHDL code and CPLD board, types of HDL why we use HDL and so on also we learnt VHDL and its coding method through QUARTUS software/IDE.

Conclusion:

In this experiment we wrote the VHDL code for full adder in all three modelling styles i.e dataflow, behavioral, structural and observed the gate level netlist in the quartus software and emulated the CPLD board as full adder from the VHDL code written for all three styles.