Expt no: 2

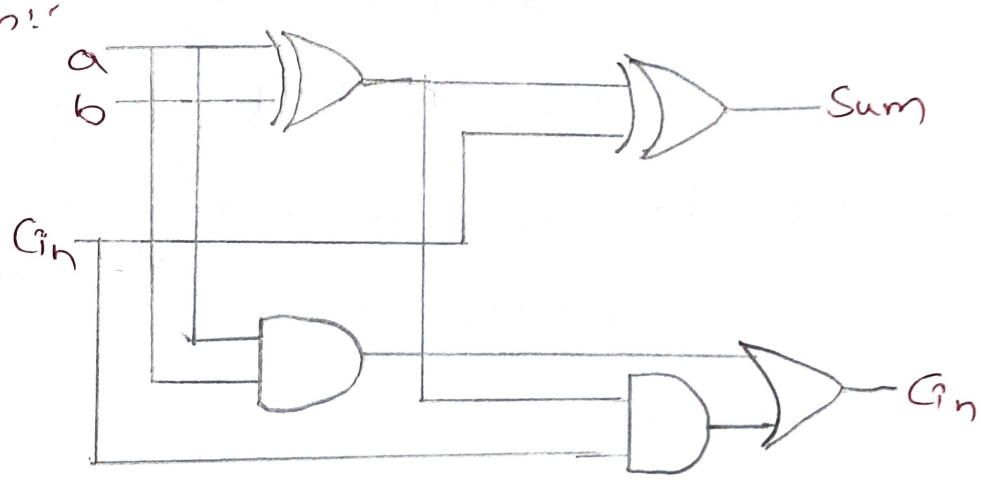
Full Adder

Name: Sumanth Balabhadruni	Date: 13/07/2021
Rollnumber: CB.EN.U4ECE18211	Marks: out of 15

Aim:- Design a full adder using VHDL.

Tools used:- modelsim

Block diagram:-



Truth Table:-

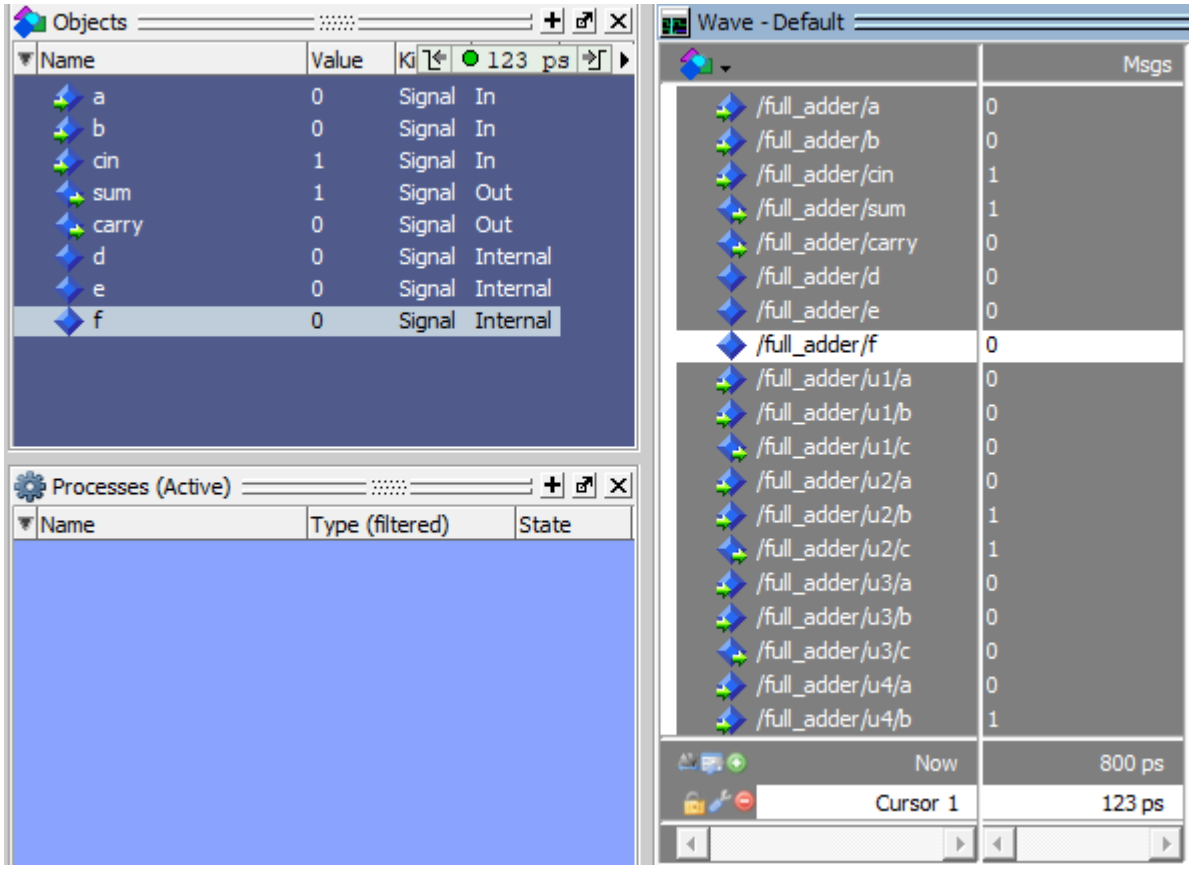
Input			output	
a	b	Cin	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

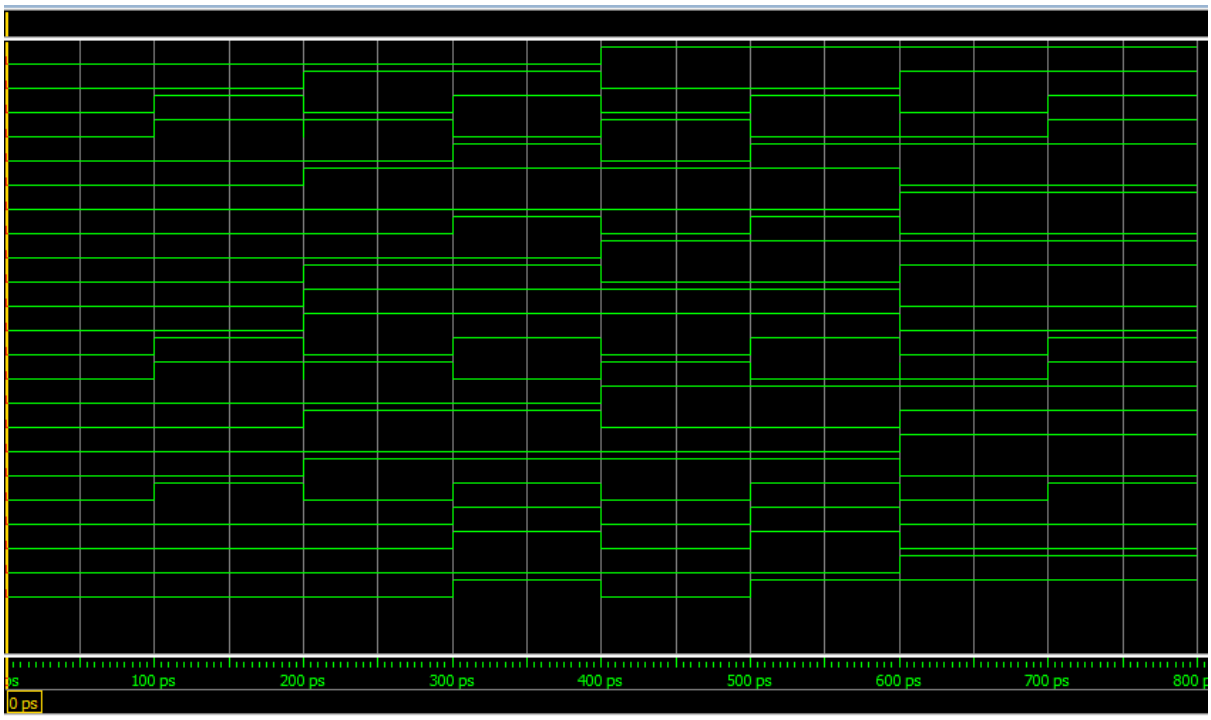
Code:

```
library ieee;
use ieee.std_logic_1164.all;

entity xor is
 Port (a: in std_logic;
       b: in std_logic;
       c: out std_logic
 );
 end xor2;
 architecture xo2_a of xor2_ is
 begin
      c<=a xor b;

 end xor2_a;

library ieee;
use ieee.std_logic_1164_all;
entity and2 is
Port (a: in std_logic;
      b: in std_logic;
      c: out std_logic

);
end and2
```

```
architecture and_2a of and2 is
begin c<a and b;
end and2_a;
library ieee;
use ieee.std_logic_1164.all;
entity or2 is
 Port (a: in std_logic;
       b: in std_logic;
       c: out std_bgic
 );
end or2;
architecture or2_a of or2 is
begin
      c<=a or b;
end or2a;
library ieee;
use ieee.std_logic_1164.al;
entity full_adder is
 Port (a: in std_logic;
       b: in std_logic
       cin: in std_logic;
       Sum: out std_bgic;
       Carry: out std_logic;
 );
```

```
architecture full_adder_a of full_adder is
  Component xor2
  Port (a : in std_logic;
        b : in std_logic;
        c : out std_logic
  );
  end component
    Component or2
    Port (a : in std_logic;
          b : in std_logic;
          c : out std_logic
    );
  End component;
  Signal d : std_logic;
  signal e : std_logic;
  signal f : std_logic;
  begin
  u1 : xor2 port map (a => a, b => b, c => d);
  u2 : xor2 port map (a => d, b => cin, c => Sum);
  u3 : and2 port map (a => a, b => b, c => e);
  u4 : and2 port map (a => d, b => cin, c => f);
  u5 : or2 port map (a => f, b => e, c => Cout)
  end fulladder_a.
```





Order of inputs is corresponding to truth table .

Inference:

Full adder code has been written and compiled using modelsim and the expected output is being observed and matched to truth table.

* * * END * * *