library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use work.Package\_ComponentList.all;

------two bit full adder----------------------------------

entity TwoBitFullAdder is

 Port ( in\_a : in STD\_LOGIC\_VECTOR (1 downto 0);

 in\_b : in STD\_LOGIC\_VECTOR (1 downto 0);

 in\_carry : in STD\_LOGIC;

 out\_carry : out STD\_LOGIC;

 out\_sum : out STD\_LOGIC\_VECTOR (1 downto 0));

end TwoBitFullAdder;

architecture Behavioral of TwoBitFullAdder is

signal signal\_carry :STD\_LOGIC;

begin

 FA1 : fullAdder1bit port map(in\_a(0),in\_b(0),in\_carry,out\_sum(0),signal\_carry );

 FA2 : fullAdder1bit port map(in\_a(1),in\_b(1),signal\_carry,out\_sum(1),out\_carry);

 --out\_carry<=signal\_carry(1);

end Behavioral;

--------------1 bit full adder---------------------

entity fullAdder1bit is

 Port ( in\_a : in STD\_LOGIC;

 in\_b : in STD\_LOGIC;

 in\_carry : in STD\_LOGIC;

 out\_sum : out STD\_LOGIC;

 out\_carry : out STD\_LOGIC);

end fullAdder1bit;

architecture Behavioral of fullAdder1bit is

 signal and\_1,and\_2,and\_3 :STD\_LOGIC ;

begin

 uSUM : threeInputXORgate port map(in\_a,in\_b,in\_carry,out\_sum);

 u1 : twoInputAndGate port map(in\_a,in\_b,and\_1);

 u2 : twoInputAndGate port map(in\_a,in\_carry,and\_2);

 u3 : twoInputAndGate port map(in\_b,in\_carry,and\_3);

 uCarry : threeInputORgate port map(and\_1,and\_2,and\_3,out\_carry);

end Behavioral;

----------------------------------------------two input OR gate----------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity twoInputAndGate is

 Port ( in\_a : in STD\_LOGIC;

 in\_b : in STD\_LOGIC;

 out\_and : out STD\_LOGIC);

end twoInputAndGate;

architecture Behavioral of twoInputAndGate is

begin

 out\_and<= in\_a and in\_b after 20 ns;

end Behavioral;



Expected result

S0 after 20 nsec

S1 after 40 nsec

Output carry after 80 nsec