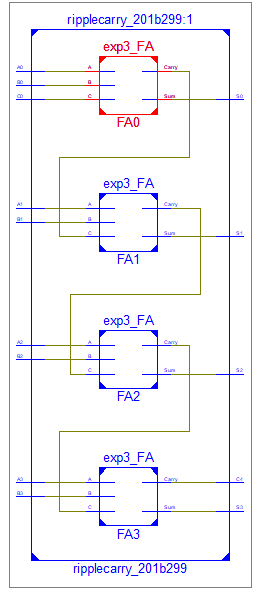
# Experiment 3

**Aim: Aim: Design of 4-bit adder-subtractor circuits.**

**Exercise#1:** Design 4-bit ripple carry adder shown in Fig.1 using

1. **structural style of architecture**

**Design Code:**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity ripplecarry\_201b299 is

Port ( A0, A1, A2,A3 : in STD\_LOGIC;

B0,B1,B2,B3 : in STD\_LOGIC;

C0 : in STD\_LOGIC;

S0, S1,S2,S3 : out STD\_LOGIC;

C4 : out STD\_LOGIC);

end ripplecarry\_201b299;

architecture structural of ripplecarry\_201b299 is

component exp3\_FA

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C : in STD\_LOGIC;

Sum : out STD\_LOGIC;

Carry : out STD\_LOGIC);

end component;

signal C1,C2,C3: STD\_LOGIC;

begin

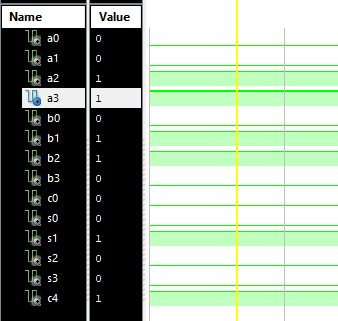
FA0: exp3\_FA port map(A0,B0,C0,S0,C1);

FA1: exp3\_FA port map(A1,B1,C1,S1,C2);

FA2: exp3\_FA port map(A2,B2,C2,S2,C3);

FA3: exp3\_FA port map(A3,B3,C3,S3,C4);

end structural;

****

**Test Bench Code:**

C0 <= '0';

A3<= '1';A2<='1';A1<='0';A0<='0';

B3<= '0';B2<='1';B1<='1';B0<='0';

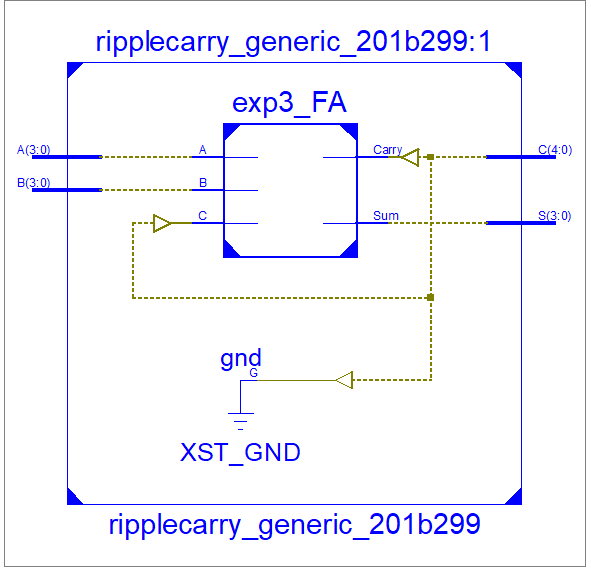
wait for 100 ns;

**(ii) generic and for-generate statements in structural style of architecture**

**Design Code:**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity ripplecarry\_generic\_201b299 is

generic (N: integer := 4);

Port ( A : in STD\_LOGIC\_VECTOR (N-1 downto 0);

B : in STD\_LOGIC\_VECTOR (N-1 downto 0);

C : inout STD\_LOGIC\_VECTOR (N downto 0);

S : out STD\_LOGIC\_VECTOR (N-1 downto 0));

end ripplecarry\_generic\_201b299;

architecture Behavioral of ripplecarry\_generic\_201b299 is

component exp3\_FA

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C : in STD\_LOGIC;

Sum : out STD\_LOGIC;

Carry : out STD\_LOGIC);

end component;

begin

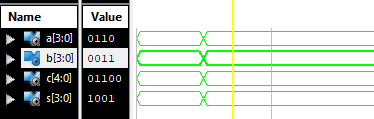
C(0) <= '0';

F: for i in 0 to N-1 generate

FA: exp3\_FA port map (A(i), B(i), C(i), S(i), C(i+1));

end generate;

end Behavioral;

****

**Test Bench Code:**

A <= "1100";

B <= "0110";

wait for 100 ns;

A <= "0110";

B <= "0011";

wait for 100 ns;

**Exercise 2:** Design 4-bit adder-subtractor shown in Fig.2 using structural style of architecture.

**Design Code:**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity adderSub\_201b299 is

generic (N: integer := 4);

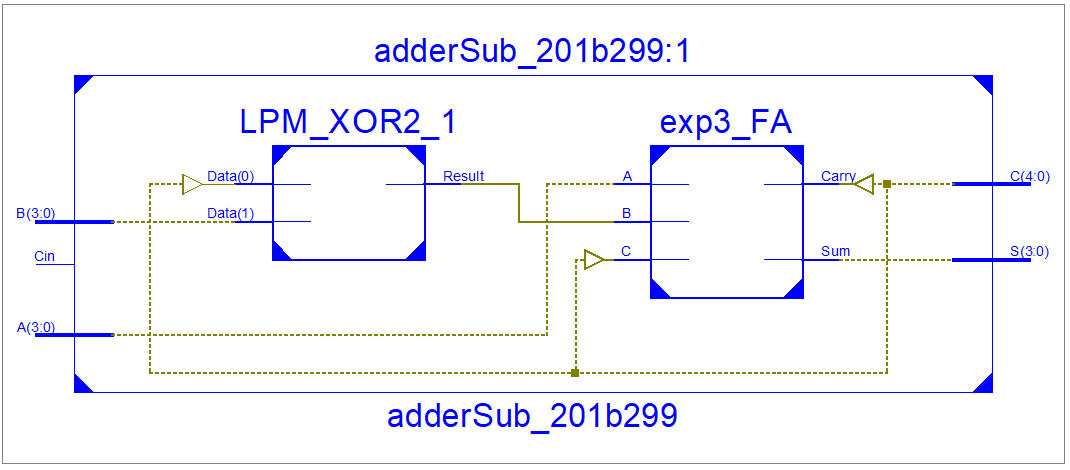
Port ( A : in STD\_LOGIC\_VECTOR (N-1 downto 0);

B : in STD\_LOGIC\_VECTOR (N-1 downto 0);

C : inout STD\_LOGIC\_VECTOR (N downto 0);

S : out STD\_LOGIC\_VECTOR (N-1 downto 0);

Cin : in STD\_LOGIC);

end adderSub\_201b299;

architecture Behavioral of adderSub\_201b299 is

component exp3\_FA

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C : in STD\_LOGIC;

Sum : out STD\_LOGIC;

Carry : out STD\_LOGIC);

end component;

signal Sig : STD\_LOGIC\_VECTOR (N-1 downto 0);

begin

C(0) <= Cin;

F: for i in 0 to N-1 generate

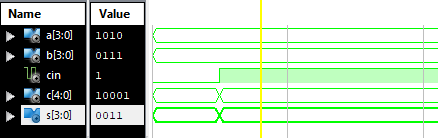
Sig(i) <= B(i) xor C(0);

FA: exp3\_FA port map (A(i), Sig(i), C(i), S(i), C(i+1));

end generate;

end Behavioral;

**Test Bench Code:**

Cin <= '0';

A <= "1010";

B <= "0111";

wait for 100 ns;

Cin <= '1';

A <= "1010";

B <= "0111";

wait for 100 ns;