FULL ADDER

Name: Vedant Golait Class: BE EnTC - A Roll No.:16

OR Gate

```
----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 21.07.2025 09:46:52

-- Design Name:

-- Module Name: og - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------
```

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity og is
    Port ( x : in STD_LOGIC;
           y : in STD_LOGIC;
           z : out STD_LOGIC);
end og;

architecture Behavioral of og is

begin

z <= x OR y;
end Behavioral;
```

Half Adder

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 21.07.2025 09:28:14
-- Design Name:
-- Module Name: ha - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;


-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;


entity ha is
    Port ( a : in STD_LOGIC;
           b : in STD_LOGIC;
           s : out STD_LOGIC;
           c : out STD_LOGIC);
end ha;


architecture Behavioral of ha is


begin


s <= a XOR b;
c <= a AND b;


end Behavioral;
```

Full Adder

```
----------------------------------------------------------------------------------
```

```
-- Company:
-- Engineer:
--
-- Create Date: 21.07.2025 09:36:03
-- Design Name:
-- Module Name: fa - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
```

```
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity fa is
   Port ( a : in STD_LOGIC;
          b : in STD_LOGIC;
          cin : in STD_LOGIC;
          s : out STD_LOGIC;
          cout : out STD_LOGIC);
end fa;

architecture structural of fa is
component ha is  --half adder
Port ( a,b : in STD_LOGIC;
      s,c : out STD_LOGIC);
end component;

component og is   --or gate
Port ( x,y : in STD_LOGIC;
      z : out STD_LOGIC);
end component;

SIGNAL s0,s1,s2:STD_LOGIC;
```

```
begin

U1: ha PORT MAP(a=>a, b=>b, s=>s0, c=>s1);

U2: ha PORT MAP(a=>s0, b=>cin, s=>s, c=>s2);

U3: og PORT MAP(x=>s2, y=>s1, z=>cout);


end structural;
```

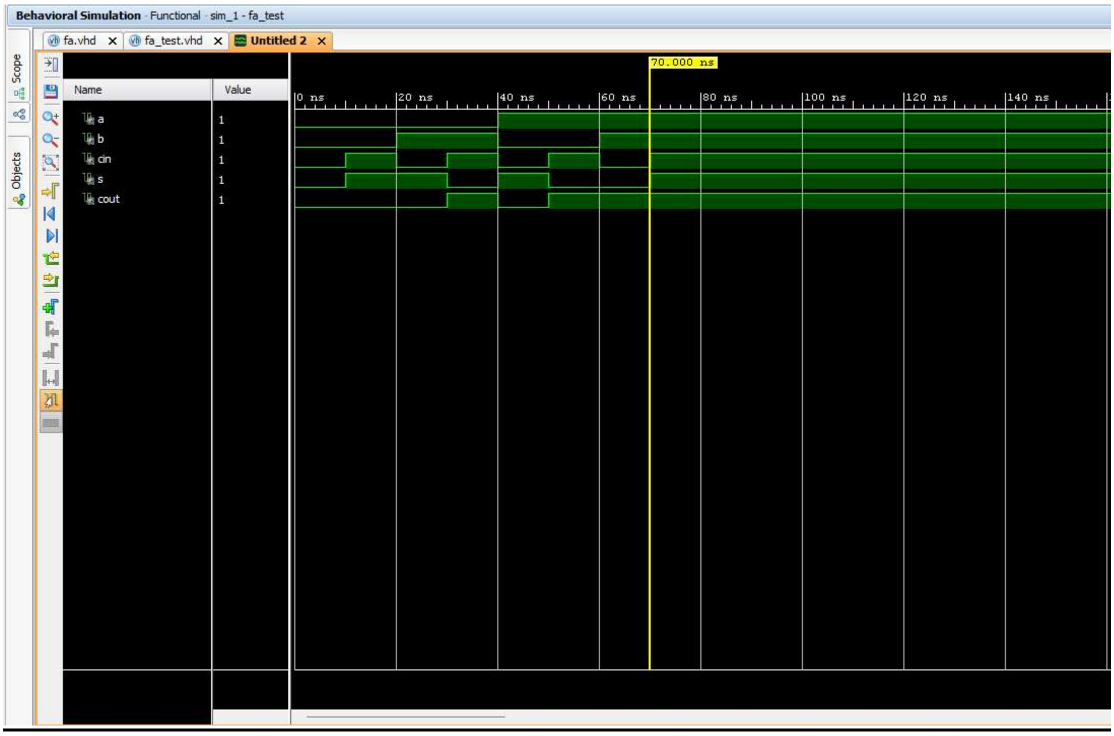
Full Adder (test bench file):

```
----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 21.07.2025 10:10:32

-- Design Name:

-- Module Name: fa_test - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created
-- Additional Comments:

--

----------------------------------------------------------------------------------



library IEEE;

use IEEE.STD_LOGIC_1164.ALL;


-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC_STD.ALL;


-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;


entity fa_test is

--  Port ( );

end fa_test;


architecture Behavioral of fa_test is


component fa is

Port(a : in STD_LOGIC;
    b : in STD_LOGIC;

    cin : in STD_LOGIC;

    s : out STD_LOGIC;

    cout : out STD_LOGIC);

end component;


SIGNAL a, b, cin, s, cout : STD_LOGIC;


begin


uut : fa PORT MAP(a, b, cin, s, cout);


stim : process

begin


a <= '0';

b <= '0';

cin <= '0';

wait for 10ns;


a <= '0';

b <= '0';

cin <= '1';

wait for 10ns;
```

```
a <= '0';
b <= '1';
cin <= '0';
wait for 10ns;

a <= '0';
b <= '1';
cin <= '1';
wait for 10ns;

a <= '1';
b <= '0';
cin <= '0';
wait for 10ns;

a <= '1';
b <= '0';
cin <= '1';
wait for 10ns;

a <= '1';
b <= '1';
cin <= '0';
wait for 10ns;
```

```
a <= '1';

b <= '1';

cin <= '1';

wait for 10ns;

wait;


end process;

end Behavioral;
```

Full Adder



Name	Value
a	1
b	1
cin	1
s	1
cout	1

HALF ADDER

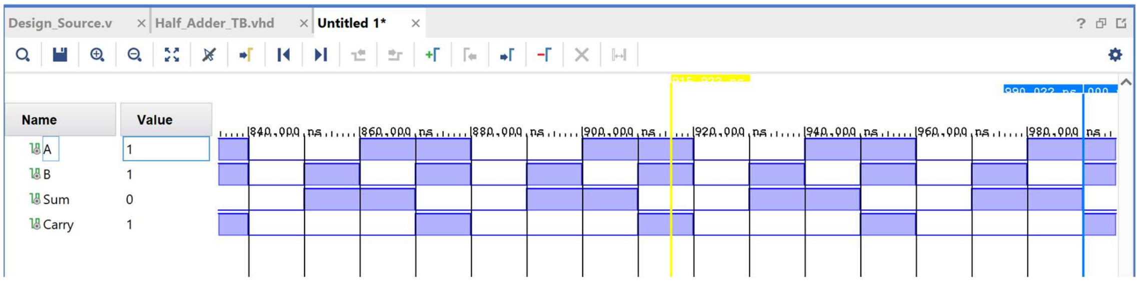
Code:

```
library ieee;
use ieee.std_logic_1164.all;
entity Half_Adder is
    port(
        A, B: in std_logic;
        Sum, Carry: out std_logic
    );
end Half_adder;
architecture df of Half_Adder is
    begin
        Sum <= A XOR B;
        Carry <= A AND B;
end df;
```

TestBench:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity Half_Adder_TB is
end Half_Adder_TB;
architecture df of Half_Adder_TB is
    component Half_Adder is
        port(
            A, B: in std_logic;
            Sum, Carry: out std_logic
        );
    end component;
    signal A, B, Sum, Carry : std_logic := '0';
begin
```

```
uut: Half_Adder PORT MAP(A, B, Sum, Carry); --unit under test
process
   begin
      A <= '0';
      B <= '0';
      wait for 10ns;
      A <= '0';
      B <= '1';
      wait for 10ns;
      A <= '1';
      B <= '0';
      wait for 10ns;
      A <= '1';
      B <= '1';
      wait for 10ns;
end process;
end df;
```



Name	Value
A	1
B	1
Sum	0
Carry	1

RIPPLE CARRY ADDER

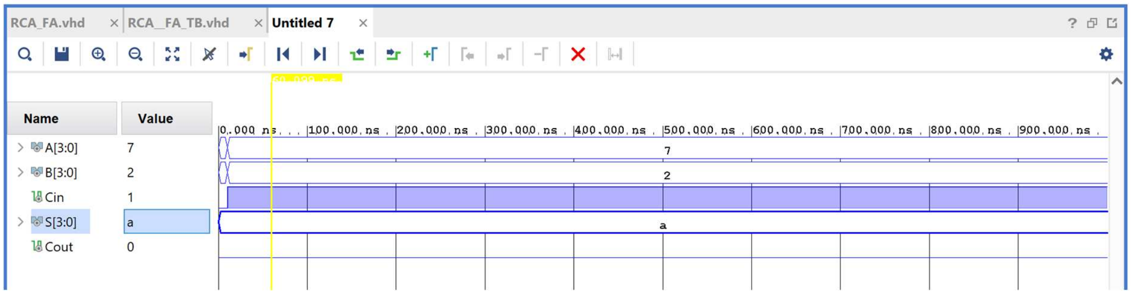
Code:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity RCA_FA is
   Port ( A : in STD_LOGIC_VECTOR (3 DOWNTO 0);
          B : in STD_LOGIC_VECTOR (3 DOWNTO 0);
          Cin : in STD_LOGIC;
          S : out STD_LOGIC_VECTOR (3 DOWNTO 0);
          Cout : out STD_LOGIC);
end RCA_FA;
architecture struct of RCA_FA is
   component Full_Adder is
      port(
         A,B: in std_logic;
         Cin: in std_logic;
         S: out std_logic;
         Cout: out std_logic
      );
   end component;
      signal C: std_logic_vector (2 downto 0);
begin
   U1: Full_Adder PORT MAP(A(0), B(0), Cin, S(0), C(0));
   U2: Full_Adder PORT MAP(A(1), B(1), C(0), S(1), C(1));
   U3: Full_Adder PORT MAP(A(2), B(2), C(1), S(2), C(2));
   U4: Full_Adder PORT MAP(A(3), B(3), C(2), S(3), Cout);
end struct;
```

TestBench:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity RCA_FA_TB is
end RCA_FA_TB;
architecture struct of RCA_FA_TB is
component RCA_FA is
   Port ( A : in STD_LOGIC_VECTOR (3 DOWNTO 0);
        B : in STD_LOGIC_VECTOR (3 DOWNTO 0);
        Cin : in STD_LOGIC;
        S : out STD_LOGIC_VECTOR (3 DOWNTO 0);
        Cout : out STD_LOGIC);
end component;
   signal A, B : STD_LOGIC_VECTOR (3 DOWNTO 0) := (others => '0');
   signal Cin  : STD_LOGIC := '0';
   signal S    : STD_LOGIC_VECTOR (3 DOWNTO 0);
   signal Cout : STD_LOGIC;
begin
   uut: RCA_FA PORT MAP (A, B, Cin, S, Cout);
   process
      begin
         A <= "0101";
         B <= "0101";
         Cin <= '0';
         wait for 10ns;
                  A <= "0111";
         B <= "0010";
         Cin <= '1';
         wait for 10ns;
      wait;
```

```
    end process;
end struct;
```



Name	Value
A[3:0]	7
B[3:0]	2
Cin	1
S[3:0]	a
Cout	0