Multiplier LAB

Introduction

- We use complete adders, which are crucial parts of digital logic circuits, to accomplish the design of multipliers in this lab report. Our goal is to use complete adders' capabilities to create a 2x2 multiplier and a 4x4 multiplier.
- With a focus on maximizing speed and functionality, these multipliers will be made to multiply binary values effectively.

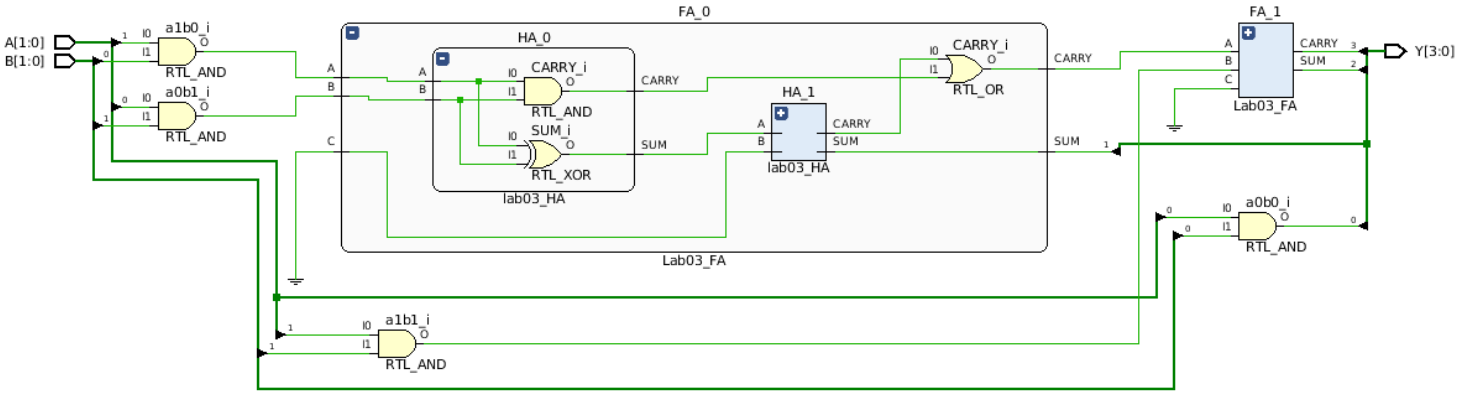
2x2 Multiplier

1. Design Source file

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 03/11/2025 02:04:25 PM
-- Design Name:
-- Module Name: Multiplier_2bit - 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 Multiplier_2bit is
    Port (  A : in STD_LOGIC_VECTOR (1 downto 0);
            B : in STD_LOGIC_VECTOR (1 downto 0);
            Y : out STD_LOGIC_VECTOR (3 downto 0));
end Multiplier_2bit;

architecture Behavioral of Multiplier_2bit is
Component Lab03_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 a0b0, a1b0, a0b1, a1b1 : std_logic ;
signal c1, s1, c2, s2 : std_logic;
```

```
begin

    a0b0 <= a(0) AND b(0) ;
    a1b0 <= a(1) AND b(0) ;
    a0b1 <= a(0) AND b(1) ;
    a1b1 <= a(1) AND b(1) ;


    FA_0 : lab03_FA
    port map(
            A => a1b0,
            B => a0b1,
            C => '0',
            SUM => s1,
            CARRY => c1
    );

    FA_1 : lab03_FA
    port map(
            A => c1,
            B => a1b1,
            C => '0',
            SUM => s2,
            CARRY => c2
    );


    Y(0) <= a0b0 ;
    Y(1) <= s1;
    Y(2) <= s2;
    Y(3) <= c2;



end Behavioral;
```

2. Elaborated design schematic

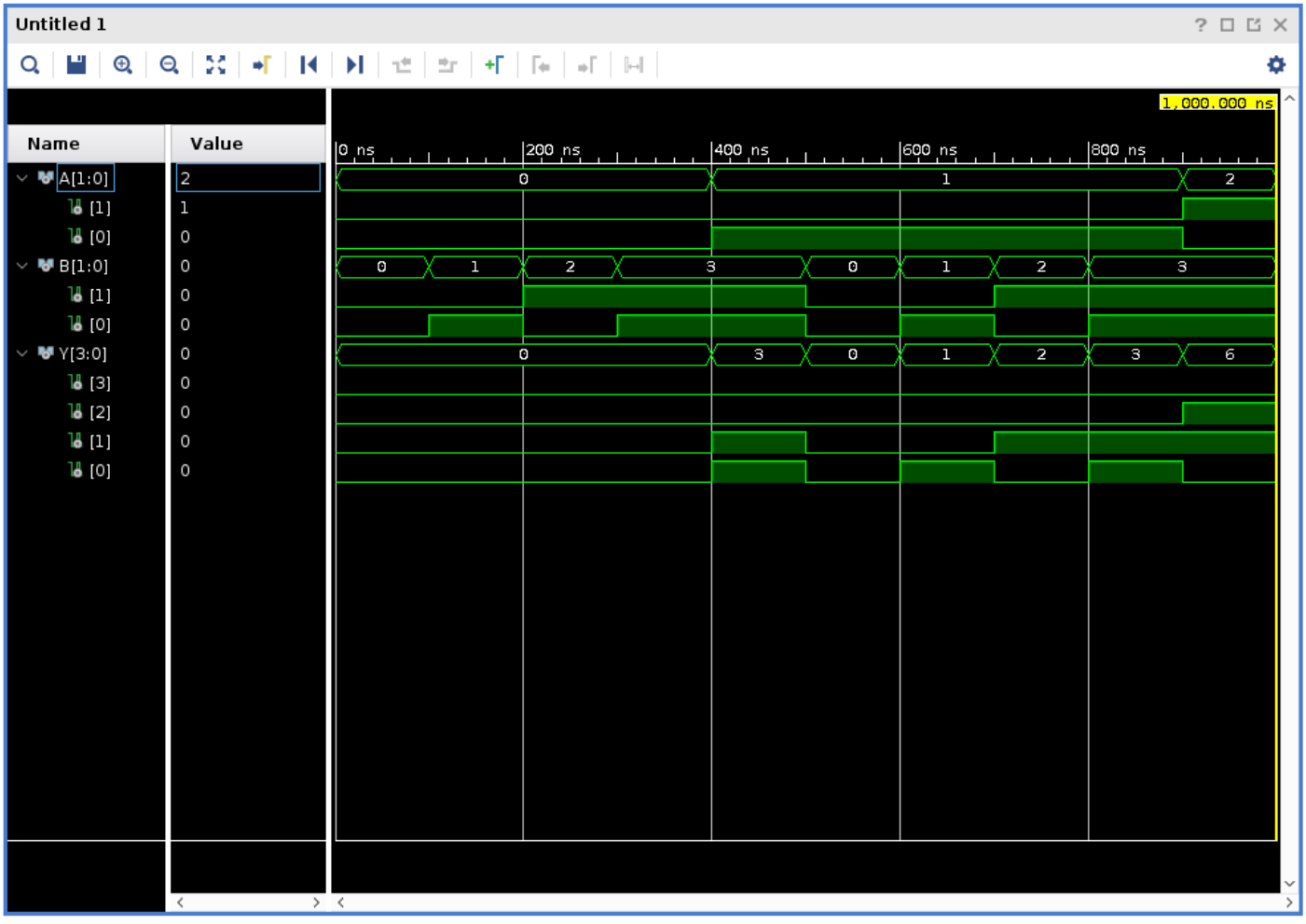


3. Simulation source file

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 03/11/2025 02:27:29 PM
-- Design Name:
-- Module Name: TB_Multiplier_2bit - 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 TB_Multiplier_2bit is
--  Port ( );
end TB_Multiplier_2bit;

architecture Behavioral of TB_Multiplier_2bit is
component Multiplier_2bit
       Port (  A : in STD_LOGIC_VECTOR (1 downto 0);
               B : in STD_LOGIC_VECTOR (1 downto 0);
               Y : out STD_LOGIC_VECTOR (3 downto 0));
end Component;

signal A : std_logic_vector(1 downto 0);
signal B : std_logic_vector(1 downto 0);
signal Y : std_logic_vector(3 downto 0);

begin
UUT  : Multiplier_2bit Port Map(
                        A  =>  A,
                        B =>  B,
                        Y  =>  Y
);

process
begin
          A <= "00";
          B <= "00"; WAIT FOR 100ns;
          B <= "01"; WAIT FOR 100ns;
          B <= "10"; WAIT FOR 100ns;
          B <= "11"; WAIT FOR 100ns;

          A <= "01"; WAIT FOR 100ns;
          B <= "00"; WAIT FOR 100ns;
          B <= "01"; WAIT FOR 100ns;
          B <= "10"; WAIT FOR 100ns;
```

```
        B <= "11"; WAIT FOR 100ns;

        A <= "10"; WAIT FOR 100ns;
        B <= "00"; WAIT FOR 100ns;
        B <= "01"; WAIT FOR 100ns;
        B <= "10"; WAIT FOR 100ns;
        B <= "11"; WAIT FOR 100ns;

        A <= "11"; WAIT FOR 100ns;
        B <= "00"; WAIT FOR 100ns;
        B <= "01"; WAIT FOR 100ns;
        B <= "10"; WAIT FOR 100ns;
        B <= "11"; WAIT FOR 100ns;

        wait;


    end process;

    end Behavioral;
```

4. Timing diagram



4x4 Multiplier

1. Design Source file

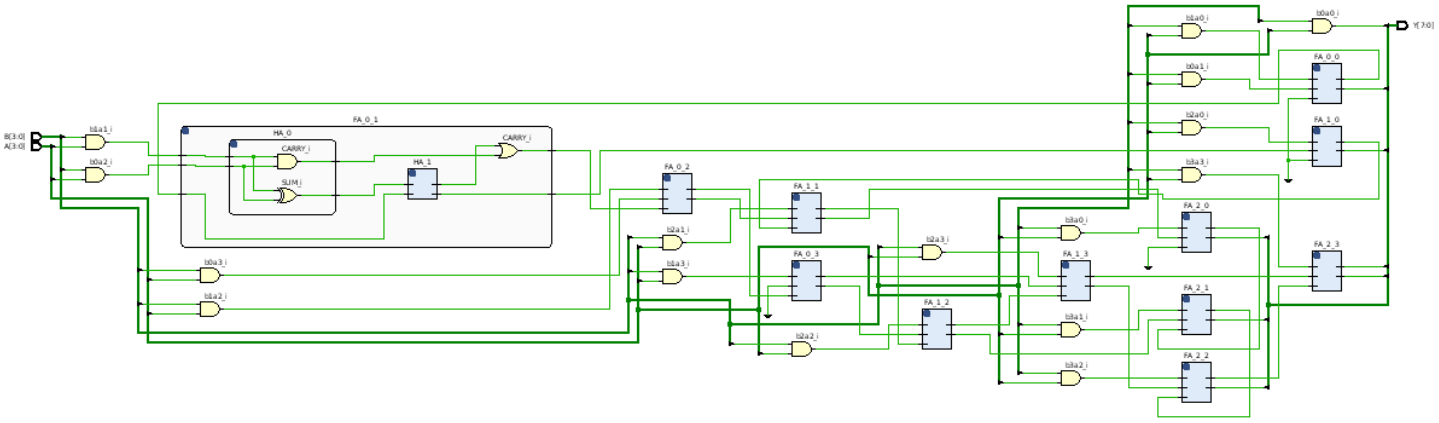
```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 03/11/2025 02:54:00 PM
-- Design Name:
-- Module Name: Multiplier_4bits - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- 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 Multiplier_4bits is
    Port ( A : in STD_LOGIC_VECTOR(3 downto 0);
           B : in STD_LOGIC_VECTOR(3 downto 0);
           Y : out STD_LOGIC_VECTOR(7 downto 0));
end Multiplier_4bits;

architecture Behavioral of Multiplier_4bits is
Component Lab03_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 b0a0, b0a1, b0a2, b0a3 : std_logic;
signal b1a0, b1a1, b1a2, b1a3 : std_logic;
signal b2a0, b2a1, b2a2, b2a3 : std_logic;
signal b3a0, b3a1, b3a2, b3a3 : std_logic;

signal c_0_0, c_0_1, c_0_2, c_0_3 : std_logic;
signal c_1_0, c_1_1, c_1_2, c_1_3 : std_logic;
signal c_2_0, c_2_1, c_2_2, c_2_3 : std_logic;

signal s_0_0, s_0_1, s_0_2, s_0_3 : std_logic;
signal s_1_0, s_1_1, s_1_2, s_1_3 : std_logic;
signal s_2_0, s_2_1, s_2_2, s_2_3 : std_logic;
```

```
begin

	b0a0 <= B(0) and A(0);
	b0a1 <= B(0) and A(1);
	b0a2 <= B(0) and A(2);
	b0a3 <= B(0) and A(3);

	b1a0 <= B(1) and A(0);
	b1a1 <= B(1) and A(1);
	b1a2 <= B(1) and A(2);
	b1a3 <= B(1) and A(3);

	b2a0 <= B(2) and A(0);
	b2a1 <= B(2) and A(1);
	b2a2 <= B(2) and A(2);
	b2a3 <= B(2) and A(3);

	b3a0 <= B(3) and A(0);
	b3a1 <= B(3) and A(1);
	b3a2 <= B(3) and A(2);
	b3a3 <= B(3) and A(3);


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

	FA_0_0 : Lab03_FA
	port map (
	   A => b1a0,
	   B => b0a1,
	   C => '0',
	   SUM => s_0_0,
	   CARRY => c_0_0
	);

	FA_0_1 : Lab03_FA
	port map (
	   A => b1a1,
	   B => b0a2,
	   C => c_0_0,
	   SUM => s_0_1,
	   CARRY => c_0_1
	);

	FA_0_2 : Lab03_FA
	port map (
	   A => b1a2,
	   B => b0a3,
	   C => c_0_1,
	   SUM => s_0_2,
	   CARRY => c_0_2
	);

	FA_0_3 : Lab03_FA
	port map (
	   A => b1a3,
	   B => '0',
	   C => c_0_2,
	   SUM => s_0_3,
	   CARRY => c_0_3
	);

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

	FA_1_0 : Lab03_FA
	port map (
	   A => b2a0,
	   B => s_0_1,
	   C => '0',
	   SUM => s_1_0,
	   CARRY => c_1_0
```

```
    );

    FA_1_1 : Lab03_FA
    port map (
        A => b2a1,
        B => s_0_2,
        C => c_1_0,
        SUM => s_1_1,
        CARRY => c_1_1
    );

    FA_1_2 : Lab03_FA
    port map (
        A => b2a2,
        B => s_0_3,
        C => c_1_1,
        SUM => s_1_2,
        CARRY => c_1_2
    );

    FA_1_3 : Lab03_FA
    port map (
        A => b2a3,
        B => c_0_3,
        C => c_1_2,
        SUM => s_1_3,
        CARRY => c_1_3
    );

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

    FA_2_0 : Lab03_FA
    port map (
        A => b3a0,
        B => s_1_1,
        C => '0',
        SUM => s_2_0,
        CARRY => c_2_0
    );

    FA_2_1 : Lab03_FA
    port map (
        A => b3a1,
        B => s_1_2,
        C => c_2_0,
        SUM => s_2_1,
        CARRY => c_2_1
    );

    FA_2_2 : Lab03_FA
    port map (
        A => b3a2,
        B => s_1_3,
        C => c_2_1,
        SUM => s_2_2,
        CARRY => c_2_2
    );

    FA_2_3 : Lab03_FA
    port map (
        A => b3a3,
        B => c_1_3,
        C => c_2_2,
        SUM => s_2_3,
        CARRY => c_2_3
    );



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

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

Y(0) <= b0a0;
Y(1) <= s_0_0;
Y(2) <= s_1_0;
Y(3) <= s_2_0;
Y(4) <= s_2_1;
Y(5) <= s_2_2;
Y(6) <= s_2_3;
Y(7) <= c_2_3;


end Behavioral;
```

2. Elaborated design schematic



3. Simulation source file

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 03/12/2025 07:42:33 PM
-- Design Name:
-- Module Name: TB_Multiplier_4bits - 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 TB_Multiplier_4bits is
--  Port ( );
end TB_Multiplier_4bits;

architecture Behavioral of TB_Multiplier_4bits is
component Multiplier_4bits
       Port (  A : in STD_LOGIC_VECTOR (3 downto 0);
               B : in STD_LOGIC_VECTOR (3 downto 0);
               Y : out STD_LOGIC_VECTOR (7 downto 0));
end Component;

signal A : std_logic_vector(3 downto 0);
signal B : std_logic_vector(3 downto 0);
signal Y : std_logic_vector(7 downto 0);


begin
UUT  : Multiplier_4bits
      Port Map(
               A  =>  A,
               B  =>  B,
               Y  =>  Y
);

process
begin
          A <= "1010";                  --random value
          B <= "0000"; WAIT FOR 100ns;
          B <= "0001"; WAIT FOR 100ns;
          B <= "0010"; WAIT FOR 100ns;
          B <= "0011"; WAIT FOR 100ns;
          B <= "0100"; WAIT FOR 100ns;
          B <= "0101"; WAIT FOR 100ns;
          B <= "0110"; WAIT FOR 100ns;
          B <= "0111"; WAIT FOR 100ns;
          B <= "1000"; WAIT FOR 100ns;
          B <= "1001"; WAIT FOR 100ns;
          B <= "1010"; WAIT FOR 100ns;
          B <= "1011"; WAIT FOR 100ns;
          B <= "1100"; WAIT FOR 100ns;
          B <= "1101"; WAIT FOR 100ns;
          B <= "1110"; WAIT FOR 100ns;
          B <= "1111"; WAIT FOR 100ns;

          A <= "1001"; WAIT FOR 100ns;    --random value
          B <= "0000"; WAIT FOR 100ns;
          B <= "0001"; WAIT FOR 100ns;
          B <= "0010"; WAIT FOR 100ns;
          B <= "0011"; WAIT FOR 100ns;
          B <= "0100"; WAIT FOR 100ns;
          B <= "0101"; WAIT FOR 100ns;
          B <= "0110"; WAIT FOR 100ns;
          B <= "0111"; WAIT FOR 100ns;
          B <= "1000"; WAIT FOR 100ns;
          B <= "1001"; WAIT FOR 100ns;
          B <= "1010"; WAIT FOR 100ns;
          B <= "1011"; WAIT FOR 100ns;
```

```
        B <= "1100"; WAIT FOR 100ns;
        B <= "1101"; WAIT FOR 100ns;
        B <= "1110"; WAIT FOR 100ns;
        B <= "1111"; WAIT FOR 100ns;

        A <= "1110"; WAIT FOR 100ns;    --random value
        B <= "0000"; WAIT FOR 100ns;
        B <= "0001"; WAIT FOR 100ns;
        B <= "0010"; WAIT FOR 100ns;
        B <= "0011"; WAIT FOR 100ns;
        B <= "0100"; WAIT FOR 100ns;
        B <= "0101"; WAIT FOR 100ns;
        B <= "0110"; WAIT FOR 100ns;
        B <= "0111"; WAIT FOR 100ns;
        B <= "1000"; WAIT FOR 100ns;
        B <= "1001"; WAIT FOR 100ns;
        B <= "1010"; WAIT FOR 100ns;
        B <= "1011"; WAIT FOR 100ns;
        B <= "1100"; WAIT FOR 100ns;
        B <= "1101"; WAIT FOR 100ns;
        B <= "1110"; WAIT FOR 100ns;
        B <= "1111"; WAIT FOR 100ns;

        wait;

end process;

end Behavioral;
```

4. Timing diagram