main ▾

Digital-electronics-1 / LAB_02 / **README.md**

Raw   Blame

183 lines (160 sloc) | 7.37 KB

🔗 BPC-DE1 Lab_02

Github Digital-electronic-1 repository

🔗 Preparation task

Truth Table for 2-bit comparator

Dec. equivalent	B[1:0]	A[1:0]	B is greater than A	B equals A	B is less than A
0	0 0	0 0	0	1	0
1	0 0	0 1	0	0	1
2	0 0	1 0	0	0	1
3	0 0	1 1	0	0	1
4	0 1	0 0	1	0	0
5	0 1	0 1	0	1	0

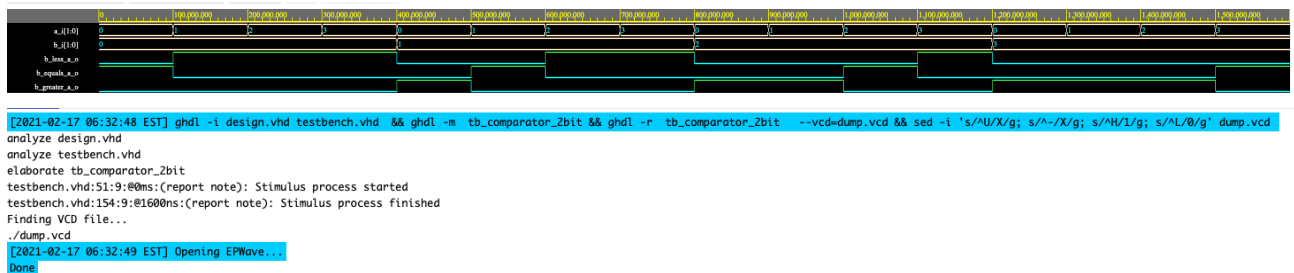| Dec. equivalent	B[1:0]	A[1:0]	B is greater than A	B equals A	B is less than A
6	0 1	1 0	0	0	1
7	0 1	1 1	0	0	1
8	1 0	0 0	1	0	0
9	1 0	0 1	1	0	0
10	1 0	1 0	0	1	0
11	1 0	1 1	0	0	1
12	1 1	0 0	1	0	0
13	1 1	0 1	1	0	0
14	1 1	1 0	1	0	0
15	1 1	1 1	0	1	0

🔗 2-bit comparator

🔗 K-maps and Sop/Pos functions

[🖼️ K_maps](#)

🔗 Simulation screenshots

🔗 Playground link

[Link to playgroundEDA.com 2-bit comparator](#)

🔗 4-bit comparator

🔗 Design listing

```vhdl
entity comparator_4bit is
    port(
        a_i             : in  std_logic_vector(4 - 1 downto 0);
        b_i             : in  std_logic_vector(4 - 1 downto 0);

        B_greater_A_o   : out std_logic;        -- B is greather than A
        B_equals_A_o    : out std_logic;        -- B is equal to A
        B_less_A_o      : out std_logic         -- B is less than A
    );
end entity comparator_4bit;


-------------------------------------------------------------------------
-- Architecture body for 4-bit binary comparator
-------------------------------------------------------------------------
architecture Behavioral of comparator_4bit is
begin
    B_greater_A_o   <= '1' when (b_i = a_i) else '0';
    B_equals_A_o    <= '1' when (b_i = a_i) else '0';
    B_less_A_o      <= '1' when (b_i < a_i) else '0';
end architecture Behavioral;
```

🔗 Test-bench listing

```vhdl
entity tb_comparator_4bit is
    -- Entity of testbench is always empty
end entity tb_comparator_4bit;


-------------------------------------------------------------------------------
-- Architecture body for testbench
-------------------------------------------------------------------------------
architecture testbench of tb_comparator_4bit is

    -- Local signals
    signal s_a          : std_logic_vector(4 - 1 downto 0);
    signal s_b          : std_logic_vector(4 - 1 downto 0);
    signal s_B_greater_A : std_logic;
    signal s_B_equals_A  : std_logic;
    signal s_B_less_A    : std_logic;

begin
    -- Connecting testbench signals with comparator_2bit entity (Unit Under Tes
    uut_comparator_2bit : entity work.comparator_2bit
        port map(
            a_i             => s_a,
            b_i             => s_b,
            B_greater_A_o => s_B_greater_A,
```

```vhdl
            B_equals_A_o  => s_B_equals_A,
            B_less_A_o    => s_B_less_A
        );


    --------------------------------------------------------------------------
    -- Data generation process 0
    --------------------------------------------------------------------------
    p_stimulus : process
    begin
        -- Report a note at the begining of stimulus process
        report "Stimulus process started" severity note;


        -- First test values
        s_b <= "0000"; s_a <= "0000"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '0') and (s_B_equals_A = '1') and (s_B_less_A
        -- If false, then report an error
        report "Test failed for input combination: 00, 00" severity error;

        s_b <= "1100"; s_a <= "0101"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A
        -- If false, then report an error
        report "Test failed for input combination: 00, 01" severity error;

        s_b <= "0010"; s_a <= "0111"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '0') and (s_B_equals_A = '0') and (s_B_less_A
        -- If false, then report an error
        report "Test failed for input combination: 00, 10" severity error;

        s_b <= "1111"; s_a <= "1111"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '0') and (s_B_equals_A = '1') and (s_B_less_A
        -- If false, then report an error
        report "Test failed for input combination: 00, 11" severity error;

        s_b <= "1111"; s_a <= "1010"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A
        -- If false, then report an error
        report "Test failed for input combination: 01, 00" severity error;

        s_b <= "0110"; s_a <= "1000"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '0') and (s_B_equals_A = '0') and (s_B_less_A
        -- If false, then report an error
        report "Test failed for input combination: 01, 01" severity error;

        s_b <= "1011"; s_a <= "1101"; wait for 100 ns;
```

```
            -- Expected output
            assert ((s_B_greater_A = '0') and (s_B_equals_A = '0') and (s_B_less_A
            -- If false, then report an error
            report "Test failed for input combination: 01, 10" severity error;

            s_b <= "1001"; s_a <= "0100"; wait for 100 ns;
            -- Expected output
            assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A
            -- If false, then report an error
            report "Test failed for input combination: 01, 11" severity error;

            s_b <= "0101"; s_a <= "0101"; wait for 100 ns;
            -- Expected output
            assert ((s_B_greater_A = '0') and (s_B_equals_A = '1') and (s_B_less_A
            -- If false, then report an error
            report "Test failed for input combination: 10, 00" severity error;

            s_b <= "1110"; s_a <= "1100"; wait for 100 ns;
            -- Expected output
            assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A
            -- If false, then report an error
            report "Test failed for input combination: 10, 01" severity error;


            -- Report a note at the end of stimulus process
            report "Stimulus process finished" severity note;
            wait;
        end process p_stimulus;

end architecture testbench;
```

🔗 Terminal listing (with error)

```
[2021-02-17 09:02:38 EST] ghdl -i design.vhd testbench.vhd  && ghdl -m  tb_comp
analyze design.vhd
analyze testbench.vhd
elaborate tb_comparator_2bit
testbench.vhd:51:9:@0ms:(report note): Stimulus process started
testbench.vhd:57:9:@100ns:(assertion error): Test failed for input combination:
testbench.vhd:63:9:@200ns:(assertion error): Test failed for input combination:
testbench.vhd:75:9:@400ns:(assertion error): Test failed for input combination:
testbench.vhd:81:9:@500ns:(assertion error): Test failed for input combination:
testbench.vhd:99:9:@800ns:(assertion error): Test failed for input combination:
testbench.vhd:105:9:@900ns:(assertion error): Test failed for input combination
testbench.vhd:111:9:@1us:(assertion error): Test failed for input combination:
testbench.vhd:117:9:@1us:(report note): Stimulus process finished
Done
```

🔗 Playground link

[Link to PlaygroundEDA.com 4-bit comparator](#)