Digital-electronics-1

Labs/02-logic

Dominik Grenčík, 220815

Digital-electronics-1

1. Binary comparator truth table
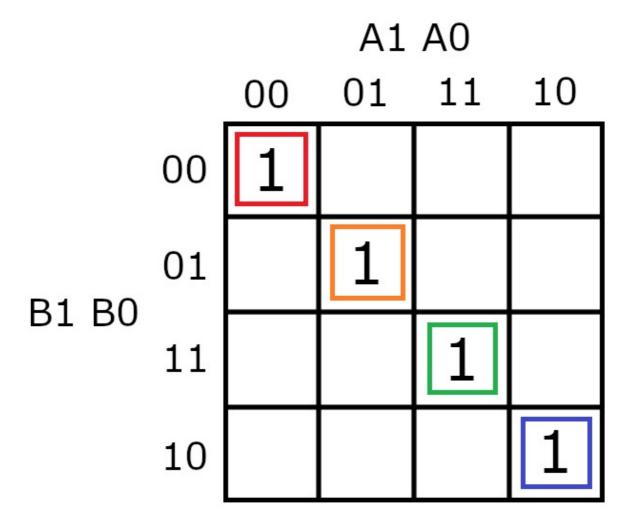
Dec. equivalent	B[1:0]	A[1:0]	B is greater than A	B equals A	B is less than A
0	0 0	0 0	0	1	0
1	0 0	0 1	0	0	1
2	0 0	1 0	0	0	1
3	0 0	1 1	0	0	1
4	0 1	0 0	1	0	0
5	0 1	0 1	0	1	0
6	0 1	1 0	0	0	1
7	0 1	1 1	0	0	1
8	1 0	0 0	1	0	0
9	1 0	0 1	1	0	0
10	1 0	1 0	0	1	0
11	1 0	1 1	0	0	1
12	1 1	0 0	1	0	0
13	1 1	0 1	1	0	0
14	1 1	1 0	1	0	0
15	1 1	1 1	0	1	0

$$equals_{SoP}^{canon} = \left(\overline{B_1} \cdot \overline{B_0} \cdot \overline{A_1} \cdot \overline{A_0}\right) + \left(\overline{B_1} \cdot B_0 \cdot \overline{A_1} \cdot A_0\right) + \left(B_1 \cdot \overline{B_0} \cdot A_1 \cdot \overline{A_0}\right) + \left(B_1 \cdot B_0 \cdot A_1 \cdot A_0\right)$$

$$less_{PoS}^{canon} = (B_1 + B_0 + A_1 + A_0) \cdot \left(B_1 + \overline{B_0} + A_1 + A_0\right) \cdot \left(B_1 + \overline{B_0} + A_1 + \overline{A_0}\right) \cdot \left(\overline{B_1} + B_0 + A_1 + A_0\right) \cdot \left(\overline{B_1} + B_0 + A_1 + \overline{A_0}\right) \cdot$$

$$= \left(\overline{B_1} + B_0 + \overline{A_1} + A_0\right) \cdot \left(\overline{B_1} + \overline{B_0} + A_1 + A_0\right) \cdot \left(\overline{B_1} + \overline{B_0} + A_1 + \overline{A_0}\right) \cdot \left(\overline{B_1} + \overline{B_0} + \overline{A_1} + A_0\right) \cdot \left(\overline{B_1} + \overline{B_0} + \overline{A_1} + \overline{A_0}\right)$$

2. 2-bit comparator

Karnaughove mapy

- Equals SoP

$$A1\ A0$$

	00	01	11	10
00	1			
01		1		
11			1	
10				1

B1 B0

- Greater SoP

A1 A0

	00	01	11	10
00				
01	1			
11	1	1		1
10	1	1		

B1 B0

- Less PoS

$$A1\ A0$$

$$greater_{SoP}^{min} = \left(B_1 \cdot \overline{A_1}\right) + \left(B_1 \cdot B_0 \cdot \overline{A_0}\right) + \left(B_0 \cdot \overline{A_1} \cdot \overline{A_0}\right)$$
$$less_{PoS}^{min} = \left(\overline{B_1} + A_1\right) \cdot \left(\overline{B_0} + A_1\right) \cdot \left(A_1 + A_0\right) \cdot \left(\overline{B_1} + \overline{B_0}\right) \cdot \left(\overline{B_1} + A_0\right)$$

EDA Playgrounds 2-bit comparator

---

3. 4-bit comparator

- **Listing of VHDL architecture from design file (design.vhd)**

```
-------------------------------------------------------------------------
--
-- Example of 4-bit binary comparator using the when/else assignment.
-- EDA Playground
--
-- Copyright (c) 2020-2021 Tomas Fryza
-- Dept. of Radio Electronics, Brno University of Technology, Czechia
-- This work is licensed under the terms of the MIT license.
--
-------------------------------------------------------------------------

library ieee;
use ieee.std_logic_1164.all;
```

```
    --------------------------------------------------------------------
    -- Entity declaration for 2-bit binary comparator
    --------------------------------------------------------------------
    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);


            -- COMPLETE ENTITY DECLARATION
            B_greater_A_o    : out std_logic;
            B_equals_A_o     : out std_logic;
            B_less_A_o       : out std_logic       -- B is less than A
        );
    end entity comparator_4bit;


    --------------------------------------------------------------------
    -- Architecture body for 2-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';


        -- WRITE "GREATER" AND "EQUALS" ASSIGNMENTS HERE


    end architecture Behavioral;
```

- **Listing of VHDL stimulus process from testbench file (testbench.vhd)**

```
    --------------------------------------------------------------------
    --
    -- Testbench for 4-bit binary comparator.
    -- EDA Playground
    --
    -- Copyright (c) 2020-2021 Tomas Fryza
    -- Dept. of Radio Electronics, Brno University of Technology, Czechia
    -- This work is licensed under the terms of the MIT license.
    --
    --------------------------------------------------------------------


    library ieee;
    use ieee.std_logic_1164.all;


    --------------------------------------------------------------------
    -- Entity declaration for testbench
    --------------------------------------------------------------------
    entity tb_comparator_4bit is
```

```vhdl
      -- 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_4bit entity (Unit Under Test)
        uut_comparator_4bit : entity work.comparator_4bit
            port map(
                a_i           => s_a,
                b_i           => s_b,
                B_greater_A_o => s_B_greater_A,
                B_equals_A_o  => s_B_equals_A,
                B_less_A_o    => s_B_less_A
            );


        -------------------------------------------------------------------------
        -- Data generation process
        -------------------------------------------------------------------------
        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 =
'0'))
            -- If false, then report an error
            report "Test failed for input combination: 0000, 0000" severity error;

            -- First test values
            s_b <= "0000"; s_a <= "0001"; wait for 100 ns;
            -- Expected output
            assert ((s_B_greater_A = '0') and (s_B_equals_A = '0') and (s_B_less_A =
'1'))
            -- If false, then report an error
            report "Test failed for input combination: 0000, 0001" severity error;

            -- First test values
            s_b <= "0100"; s_a <= "0010"; wait for 100 ns;
            -- Expected output
```

```vhdl
        assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A =
'0'))
        -- If false, then report an error
        report "Test failed for input combination: 0100, 0010" severity error;

        -- First test values
        s_b <= "0101"; s_a <= "0110"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '0') and (s_B_equals_A = '0') and (s_B_less_A =
'1'))
        -- If false, then report an error
        report "Test failed for input combination: 0101, 0110" severity error;

        -- First test values
        s_b <= "0110"; s_a <= "0110"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '0') and (s_B_equals_A = '1') and (s_B_less_A =
'0'))
        -- If false, then report an error
        report "Test failed for input combination: 0110, 0110" severity error;

        -- First test values
        s_b <= "0110"; 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 =
'1'))
        -- If false, then report an error
        report "Test failed for input combination: 0110, 0111" severity error;

        -- First test values
        s_b <= "0111"; s_a <= "0010"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A =
'0'))
        -- If false, then report an error
        report "Test failed for input combination: 0111, 0010" severity error;

        -- First test values
        s_b <= "0111"; s_a <= "0111"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '0') and (s_B_equals_A = '1') and (s_B_less_A =
'0'))
        -- If false, then report an error
        report "Test failed for input combination: 0111, 0111" severity error;

        -- First test values
        s_b <= "1000"; s_a <= "0011"; wait for 100 ns;
        -- Expected output
        assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A =
'0'))
        -- If false, then report an error
        report "Test failed for input combination: 1000, 0011" severity error;

        -- First test values
```

```vhdl
        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 =
'1'))
        -- If false, then report an error
        report "Test failed for input combination: 1011, 1101" severity error;

        -- First test values
        s_b <= "1110"; 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 =
'1'))
        -- If false, then report an error
        report "Test failed for input combination: 1110, 1000" 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;
```

- **Listing of simulator console output, i.e. with one reported error**

```
[2021-02-20 13:21:53 EST] ghdl -i design.vhd testbench.vhd  && ghdl -m
tb_comparator_4bit && ghdl -r  tb_comparator_4bit    --vcd=dump.vcd && sed -i
's/^U/X/g; s/^-/X/g; s/^H/1/g; s/^L/0/g' dump.vcd
analyze design.vhd
analyze testbench.vhd
elaborate tb_comparator_4bit
testbench.vhd:51:9:@0ms:(report note): Stimulus process started
testbench.vhd:127:9:@1100ns:(assertion error): Test failed for input combination:
1110, 1000
testbench.vhd:133:9:@1100ns:(report note): Stimulus process finished
Finding VCD file...
./dump.vcd
[2021-02-20 13:21:54 EST] Opening EPWave...
Done
```

EDA Playgrounds 4-bit comparator