

Nemecxpetr / Digital-electronics-1

MIT License

0 stars 0 forks

[Star](#)

[Unwatch](#)

Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

[main](#) [...](#)

Digital-electronics-1 / Labs / 02-logic /



Nemecxpetr ...

8 minutes ago



..

[Images](#)

26 minutes ago

[Karnaugh.xlsx](#)

2 days ago

[README.md](#)

8 minutes ago

README.md



02 Logic

Lab assignment

1. Preparation tasks (done before the lab at home). Submit:

- Binary comparator truth table.

2. A 2-bit comparator. Submit:

- Karnaugh maps for all three functions.
- Simplified SoP form of the "greater than" function
- And simplified PoS form of the "less than" function.

3. A 4-bit binary comparator. Submit:

- Listing of VHDL code (design.vhd),

- Listing of VHDL testbench (`testbench.vhd`),
- Listing of simulator console output, i.e. with reported error,
- Link to your public EDA Playground example.

1 Binary coparator 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

2 A 2-bit binary comparator

Karnaugh maps for all free functions

		A0						A0			
		A1				A1					
		00	01	11	10			00	01	11	10
Equals		1							1		
			1							1	
				1							1
					1						
B1		00	01	B0	10	B1		00	01	B0	10
B0		11				B0		11	10	11	
Less		00	01	11	10	Legend		00	01	11	10
			1	1	1			0	1	3	2
				1	1			4	5	7	6
								12	13	15	14
					1			8	9	11	10
B1		00	01	B0	10	B1		00	01	B0	10
B0		11				B0		11	10	11	

Simplified SoP form of the "greater than" function and a PoS form of the "less than" function:

$$\begin{aligned} \text{greater_SoP}_{\min}^{\text{sim}} &= (b_1 \cdot \overline{a_1}) + (b_1 \cdot b_0 \cdot \overline{a_0}) \\ \text{less_PoS}_{\min}^{\text{sim}} &= (\overline{b_1} + a_1) \cdot (\overline{b_0} + a_1) \cdot (\overline{b_1} + \overline{b_0}) \cdot (a_1 + a_0) \cdot (\overline{b_1} + a_0) \end{aligned}$$

3 A 4-bit binary comparator

Public EDA Playground [link](#).

Listing of `design.vhd`

```

architecture Behavioral of comparator_4bit is
begin

    B_less_A_o      <= '1' when (b_i < a_i) else '0';
    B_equals_A_o   <= '1' when (b_i = a_i) else '0';
    B_greater_A_o <= '1' when (b_i > a_i) else '0';

end architecture Behavioral;

```

Listing of testebench.vhd

```

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;

    s_b <= "0011"; s_a <= "0001"; wait for 100 ns;
    assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A = '0'))
    report "Test failed for input combination: 0011, 0001" severity error;

    s_b <= "0111"; s_a <= "0101"; wait for 100 ns;
    assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A = '0'))
    report "Test failed for input combination: 0111, 0101" severity error;

    s_b <= "1111"; s_a <= "1010"; wait for 100 ns;
    assert ((s_B_greater_A = '1') and (s_B_equals_A = '0') and (s_B_less_A = '0'))
    report "Test failed for input combination: 1111, 1010" severity error;

    s_b <= "0000"; s_a <= "1100"; wait for 100 ns;
    assert ((s_B_greater_A = '0') and (s_B_equals_A = '1') and (s_B_less_A = '0'))
    report "Test failed for input combination: 0000, 1100" severity error;

    s_b <= "0011"; s_a <= "1001"; wait for 100 ns;
    assert ((s_B_greater_A = '0') and (s_B_equals_A = '0') and (s_B_less_A = '1'))
    report "Test failed for input combination: 1011, 1001" severity error;

    s_b <= "0111"; s_a <= "1000"; wait for 100 ns;
    assert ((s_B_greater_A = '0') and (s_B_equals_A = '0') and (s_B_less_A = '1'))
    report "Test failed for input combination: 0111, 1000" severity error;

    s_b <= "0111"; s_a <= "0101"; wait for 100 ns;
    assert ((s_B_greater_A = '0') and (s_B_equals_A = '1') and (s_B_less_A = '0'))
    report "Test failed for input combination: 0111, 0101" severity error;

    report "Stimulus process finished" severity note;
    wait;
end process p_stimulus;

```

Listing of simulator console output, i.e. with reported error:

```
[2021-02-21 14:44:21 EST] ghdl -i design.vhd testbench.vhd && ghdl -m  
tb_comparator_4bit && ghdl -r tb_comparator_4bit  
analyze design.vhd  
analyze testbench.vhd  
elaborate tb_comparator_4bit  
testbench.vhd:46:9:@0ms:(report note): Stimulus process started  
testbench.vhd:69:9:@500ns:(assertion error): Test failed for input combination:  
0000, 1100  
testbench.vhd:81:9:@800ns:(assertion error): Test failed for input combination:  
0111, 0101  
testbench.vhd:84:9:@800ns:(report note): Stimulus process finished  
Done
```
