```
1   ----------------------------------------------------------------------------------
2   -- Company:
3   -- Engineer:
4   --
5   -- Create Date:   15:30:50 08/17/2023
6   -- Design Name:
7   -- Module Name:   C:/Users/ETC-VLSI LAB/Downloads/Xilinx_ISE_DS_14.7_1015_1_2/New
    folder/alu_D1/alu_test.vhd
8   -- Project Name:  alu_D1
9   -- Target Device:
10  -- Tool versions:
11  -- Description:
12  --
13  -- VHDL Test Bench Created by ISE for module: modecalci
14  --
15  -- Dependencies:
16  --
17  -- Revision:
18  -- Revision 0.01 - File Created
19  -- Additional Comments:
20  --
21  -- Notes:
22  -- This testbench has been automatically generated using types std_logic and
23  -- std_logic_vector for the ports of the unit under test.  Xilinx recommends
24  -- that these types always be used for the top-level I/O of a design in order
25  -- to guarantee that the testbench will bind correctly to the post-implementation
26  -- simulation model.
27  ----------------------------------------------------------------------------------
28  LIBRARY ieee;
29  USE ieee.std_logic_1164.ALL;
30
31  -- Uncomment the following library declaration if using
32  -- arithmetic functions with Signed or Unsigned values
33  --USE ieee.numeric_std.ALL;
34
35  ENTITY alu_test IS
36  END alu_test;
37
38  ARCHITECTURE behavior OF alu_test IS
39
40      -- Component Declaration for the Unit Under Test (UUT)
41
42      COMPONENT modecalci
43      PORT(
44           a : IN  std_logic_vector(2 downto 0);
45           b : IN  std_logic_vector(2 downto 0);
46           mode : IN  std_logic_vector(2 downto 0);
47           y : OUT  std_logic_vector(2 downto 0)
48          );
49      END COMPONENT;
50
51
52     --Inputs
53     signal a : std_logic_vector(2 downto 0) := (others => '0');
54     signal b : std_logic_vector(2 downto 0) := (others => '0');
55     signal mode : std_logic_vector(2 downto 0) := (others => '0');
56
```

```
57      --Outputs
58      signal y : std_logic_vector(2 downto 0);
59      -- No clocks detected in port list. Replace <clock> below with
60      -- appropriate port name
61
62    BEGIN
63
64      -- Instantiate the Unit Under Test (UUT)
65      uut: modecalci PORT MAP (
66             a => a,
67             b => b,
68             mode => mode,
69             y => y
70           );
71
72       -- Stimulus process
73      stim_proc: process
74      begin
75       a<="001";
76       b<="010";
77       mode<="000";
78       wait for 10 ns;
79       mode<="001";
80       wait for 10 ns;
81       mode<="010";
82       wait for 10 ns;
83       mode<="011";
84       wait for 10 ns;
85       mode<="100";
86       wait for 10 ns;
87       mode<="111";
88       wait for 10 ns;
89
90
91
92         -- hold reset state for 100 ns.
93         wait for 100 ns;
94
95         wait;
96      end process;
97
98    END;
99
```

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date:    15:07:53 08/17/2023
-- Design Name:
-- Module Name:    modecalci - 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 primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity modecalci is
    Port ( a : in  STD_LOGIC_VECTOR (2 downto 0);
           b : in  STD_LOGIC_VECTOR (2 downto 0);
           mode : in  STD_LOGIC_VECTOR (2 downto 0);
           y : out  STD_LOGIC_VECTOR (2 downto 0));
end modecalci;

architecture Behavioral of modecalci is

begin
process(a,b,mode)
   begin
      case mode is
      when"000" =>y<= NOT a;
      when"001" =>y<=a AND b;
      when"010" =>y<=a OR b;
      when"111" =>y<=a NOR b;
      when"011" =>y<=a XOR b;
      when"100" =>y<=a NAND b;
      when others =>null;
      end case;
      end process;
      end Behavioral;


```