```vhdl
----------------------------------------------------------------------------------
-- Engineer: Olasubomi Borishade
--
-- Create Date: 01/30/2026 07:46:57 PM
-- Design Name: Computational Unit
-- Module Name: Comp_Unit_TestBench - Behavioral
-- Project Name: EENG 5560 Homework 1
-- Description: Testing a computational unit (CU) for 4 bit operands with 4 bit
outputs that can compute the following operations:
-- AND, NOR, Addition (ADD), Subtraction (SUB), Multiplication (MULT), Greater than
(GT), Less than (LT) & Equal to (EQ)
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use work.OPERATIONS_ARRAY_CUSTOMPACK.ALL;
use IEEE.NUMERIC_STD.ALL;
use STD.ENV.FINISH;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Comp_Unit_TestBench is
-- Port ( );
end Comp_Unit_TestBench;

architecture Behavioral of Comp_Unit_TestBench is
Constant d_w_c: integer := 4; -- Declaring data width signal
Signal As, Bs, Ys: STD_LOGIC_VECTOR (d_w_c - 1 downto 0); -- Declaring input and
output signals
Signal Opsels: CU_Operation; -- Declaring the operation selection signal as an input
vector

begin
Computation_Unit_Inst: entity work.Computation_Unit(Behavioral)
    Generic map (d_w => d_w_c) -- Port-mapping the data width values of the
instatiated computation unit
    Port map (A => As, -- Port-mapping the the input, output and select signals
              B => Bs,
              Opsel => Opsels,
              Y => Ys
```

```vhdl
            );

stim: process
begin
    As <= "1001"; Bs <= "0101"; -- Assigning first set of test vectors
        Opsels <= lAND; wait for 5 ns;
        Opsels <= lNOR; wait for 5 ns;
        Opsels <= aADD; wait for 5 ns;
        Opsels <= aSUB; wait for 5 ns;
        Opsels <= aMULT; wait for 5 ns;
        Opsels <= cGTH; wait for 5 ns;
        Opsels <= cLTH; wait for 5 ns;
        Opsels <= cEQT; wait for 5 ns;
    finish;
    end process;

end Behavioral;
```