```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 02/04/2026 10:52:24 PM
-- Design Name:
-- Module Name: ARCA_Logic_Array_Test_Bench - 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;
use STD.ENV.FINISH;
use WORK.OPERATIONS_ARRAY_CUSTOM_PACK.ALL;


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

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

architecture Behavioral of ARCA_Logic_Array_Test_Bench is
Constant d_w_c: integer := 3; -- Declaring data width signal
Constant rows: integer := 2; -- Declaring the number of rows
Constant cols: integer := 4; -- Declaring the number of columns

Signal Ais, Bis, Yis: VectorArray_1d (0 to cols - 1)(d_w_c - 1 downto 0); --
Declaring input and output port signals
Signal ADataFlow1is, ADataFlow2is: VectorArray_1d (0 to 1)(0 downto 0); -- Declaring
arithmetic logic block dataflow select signal
Signal CDataFlow1is, CDataFlow2is, RDataFlow1is, RDataFlow2is: STD_LOGIC_VECTOR (1
downto 0); -- Declaring comparison and rotational logic blocks dataflow select
signals
Signal OpSelAis: ALBArray_2d (0 to rows - 1)(0 to 1); -- Declaring arithmetic
operation select signals of each arithmetic logic block
Signal OpSelCis: CLBArray_1d (0 to rows - 1); -- Declaring comparison operation
select signals of each comparison logic block
Signal OpSelRis: RLBArray_1d (0 to rows - 1); -- Declaring rotational operation
select signals of each rotational logic block

begin
ARCA_Logic_Array_Inst: entity work.ARCA_Logic_Array(Structural)
    Generic map (d_w => d_w_c,
                    rows => rows,
                    cols => cols) -- Port-mapping the data width values of the
instatiated computation unit

    Port map (  A => Ais, -- Port-mapping the the input, output and select signals
                B => Bis,
                ADataFlow1 => ADataFlow1is,
                ADataFlow2 => ADataFlow2is,
                CDataFlow1 => CDataFlow1is,
                CDataFlow2 => CDataFlow2is,
                RDataFlow1 => RDataFlow1is,
                RDataFlow2 => RDataFlow2is,
                OpSelA => OpSelAis,
                OpSelC => OpSelCis,
                OpSelR => OpSelRis,
                Y => Yis
                );

stim: process
begin
    Ais <= (("011", "110", "101", "011"));
    Bis <= (("010", "001", "111", "011"));

    OpSelAis <= ((aMult, aADD),
                (aSUB, aSUB));

    OpSelRis <= ((rLSR),
                (rLSL));

    OpSelCis <= ((cLTH),
                (cGTET)); wait for 5ns;

    ADataFlow1is <= (("0", "0")); -- Arithmetic logic block's A inputs are (0, 0)
and (0, 2)
```

```
    ADataFlow2is <= (("1", "1")); -- Arithmetic logic block's B inputs are (0, 1)
and (0, 3)

    CDataFlow1is <= "11"; -- Comparison logic block's A input is (0, 3)
    CDataFlow2is <= "00"; -- Comparison logic block's B input is (0, 0)

    RDataFlow1is <= "10"; -- Rotational logic block's A input is (0, 2)
    RDataFlow2is <= "01"; -- Rotational logic block's B input is (0, 1)
    wait for 10ns;

end process;
end Behavioral;
```