EE214 Digital Circuits Laboratory
Wadhwani Electronics Laboratory
Electrical Engineering IIT Bombay

Problem set: 5 Date: September 12, 2025

ALU

Instructions:

1. Use Behavioral and Dataflow modelling for this experiment.
2. Perform RTL and Gate-level simulation using the provided testbench and tracefile.
3. Demonstrate the simulations to your TA.
4. Perform **Scanchain** on the Xenon board and verify with your TA.
5. Submit the entire project files in .zip format in moodle.

Problem Statement

1. Describe the given ALU using VHDL. This ALU circuit performs various functions based on select lines.

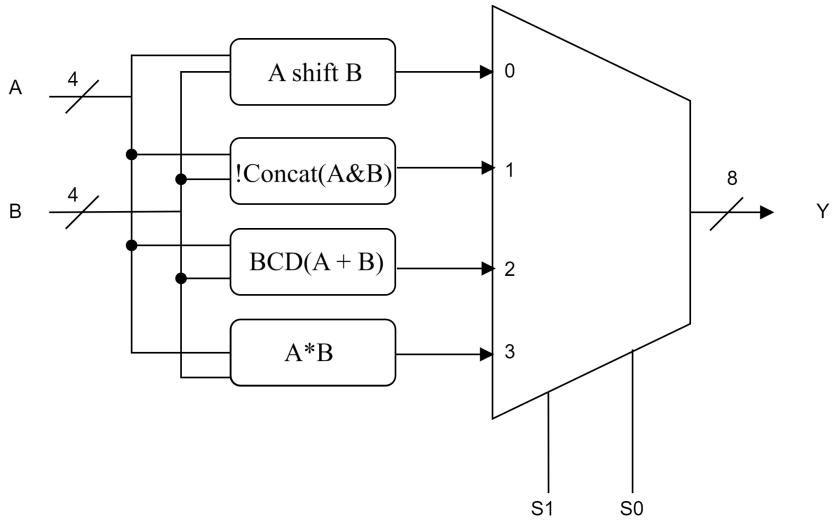

Figure 1: ALU with 4 functions

2. Use Behavioral modeling only for the given functions.

S1	S0	ALU Output
0	0	Performs shift on A by B [0 extend A to 8 bits][B3 - 0(left)/1(right)][B2 B1 B0 - no. of bits to be shifted]
0	1	Performs 2s complement on A concat B [not(concat(A&B))]
1	0	Performs A + B BCD Addition
1	1	Performs A * B Operation using Shift and Add operation [A*B]

3. To perform arithmetic, standard logic vectors may be type casted using numeric_std library (IEEE.NUMERIC_STD.ALL). Refer to the typecasting table below.

From	To	Cast Expression
std_logic_vector	unsigned	unsigned(x)
	signed	signed(x)
unsigned/signed	standard_logic_vector	std_logic_vector(x)
	integer	to_integer(x)
integer	unsigned	to_unsigned(i, N)
	signed	to_signed(i, N)

4. Simulate your design using the generic testbench to confirm the correctness of your description.
5. Tracefile format $< S1\,S0\,A3\,A2\,A1\,A0\,B3\,B2\,B1\,B0 >< Y7\,Y6\,Y5\,Y4\,Y3\,Y2\,Y1\,Y0 >$ 11111111
6. You can use the skeleton code given below:

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

entity alu_beh is
    port (
        S : in std_logic_vector(1 downto 0);
        A : in std_logic_vector(3 downto 0);
        B : in std_logic_vector(3 downto 0);
        Y : out std_logic_vector(7 downto 0)
    );
end entity;

architecture arch of alu_beh is

    function Shift(A, B : std_logic_vector(3 downto 0)) return std_logic_vector is
-- declaring and initializing variables using aggregates
        variable A_extended : std_logic_vector(...) := (others => '0'); -- extend A to 8 bits
        variable shift_amount : integer := to_integer(unsigned(B(...))); -- define shift amount
        variable result : std_logic_vector(...); --return result
    begin
        --use behavioral modeling to do shift operation
    end;

begin

   -- complete VHDL code for various outputs of ALU based on select lines
   -- Hint: use if/else statement
   --
   --   function usage :
   --   signal_name <= shift(A,B)
   --   variable_name := shift(A,B)
    end process;

end architecture;
```