**VHDL Code for 8-bit ALU**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity alu\_lab8 is

Port (A,B : in STD\_LOGIC\_VECTOR (7 downto 0);

sel : in STD\_LOGIC\_VECTOR (2 downto 0);

Y : out STD\_LOGIC\_VECTOR (7 downto 0));

end alu\_lab8;

architecture Behavioral of alu\_lab8 is

begin

process(A,B,sel)

begin

case sel is

when "000" => Y <=A+B;

when "001" => Y <=A-B;

when "010" => Y <=A AND B;

when "011" => Y <=A NAND B;

when "100" => Y <=A XOR B;

when "101" => Y <=A XNOR B;

when "110" => Y <=A OR B;

when "111" => Y <=A;

when others => Y <="ZZZZZZZZ";

end case;

end process;

end Behavioral;

**VHDL Test Bench**

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

ENTITY alu IS

END alu;

ARCHITECTURE behavior OF alu IS

-- Component Declaration for the Unit Under Test (UUT)

COMPONENT alu\_lab8

PORT(

A : IN std\_logic\_vector(7 downto 0);

B : IN std\_logic\_vector(7 downto 0);

sel : IN std\_logic\_vector(2 downto 0);

Y : OUT std\_logic\_vector(7 downto 0)

);

END COMPONENT;

--Inputs

signal A : std\_logic\_vector(7 downto 0) := (others => '0');

signal B : std\_logic\_vector(7 downto 0) := (others => '0');

signal sel : std\_logic\_vector(2 downto 0) := (others => '0');

--Outputs

signal Y : std\_logic\_vector(7 downto 0);

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: alu\_lab8 PORT MAP (

A => A,

B => B,

sel => sel,

Y => Y

);

PROCESS

BEGIN

A<="10110011";

B<="01000100";

sel<="000";

wait for 20ns;

sel<="001";

wait for 20ns;

sel<="010";

wait for 20ns;

sel<="011";

wait for 20ns;

sel<="100";

wait for 20ns;

sel<="101";

wait for 20ns;

sel<="110";

wait for 20ns;

sel<="111";

wait;

end process;

END;

**Implementation Constraint File**

Net"A<0>"LOC="A10";

Net"A<1>"LOC="D14";

Net"B<0>"LOC="C14";

Net"B<1>"LOC="P15";

Net"SEL<0>"LOC="P12";

Net"SEL<1>"LOC="R5";

Net"SEL<2>"LOC="T5";

Net"Y<0>"LOC="U18";

Net"Y<1>"LOC="M14";