library ieee;

USE ieee.std\_logic\_1164.all;

USE ieee.numeric\_std.all;

entity alu is

port(rx:in std\_logic\_vector(7 downto 0); -- input 1

ry: in std\_logic\_vector(7 downto 0); -- input 2

opcode: in std\_logic\_vector(3 downto 0); -- opcode in

sele: in std\_logic\_vector (3 downto 0); --select line

output: out std\_logic\_vector(7 downto 0) -- output

);

end alu;

architecture logic of alu is

begin

process(rx,ry,opcode,sele)

variable tempA : signed(7 downto 0); -- signed rx

variable tempB : signed(7 downto 0);--signed ry

variable tempout: signed (7 downto 0);--signed output

variable tempAA : unsigned( 7 downto 0);-- unsigned rx

variable tempBB : unsigned(7 downto 0); -- unsigned ry

variable tempoutt: unsigned (7 downto 0); -- unsigned output

begin

tempA := signed(rx);-- signed variables unused

tempB := signed(ry);

tempAA := unsigned(rx);

tempBB := unsigned(ry);

if (tempA(7) = '1' ) then

tempA :=(not tempA) + 1;

end if ;

if (tempB(7) ='1') then

tempB := (not tempB)+ 1 ;

end if;

case (opcode) is

when "0000" => null;

when "0001" => tempoutt := (tempAA + tempBB);

output <= std\_logic\_vector (tempoutt);

when "0010" => if(sele="0000") then

tempoutt := tempAA+tempBB; --additon,subtration of unsigned depending on its sel code

output <= std\_logic\_vector (tempoutt);

end if;

if(sele="0001") then

tempoutt := tempAA-tempBB ;

output <= std\_logic\_vector (tempoutt);

end if;

when "0011" => if(sele = "0000")then

tempoutt:=tempAA + "00000001";

output <= std\_logic\_vector (tempoutt);

end if;

if(sele = "0001") then

tempoutt:= tempAA - "0000001";

output <= std\_logic\_vector (tempoutt);

end if;

when "0100" => if (sele = "0000") then -- shift left by how many bits are in tempBB

tempoutt := tempAA sll to\_integer(tempBB);

output <= std\_logic\_vector (tempoutt);

end if;

if(sele = "0001") then -- shift right by how many bits are in tempBB ( srl means shift right )

tempoutt := tempAA srl to\_integer(tempBB);

output <= std\_logic\_vector (tempoutt);

end if;

when "0101" => -- logical operations

if (sele = "0000") then

output <= not(rx);

end if;

if(sele = "0001") then

output<=rx nor ry;

end if;

if(sele = "0010") then

output <= rx nand ry;

end if;

if(sele = "0011") then

output <= rx xor ry;

end if;

if(sele = "0100") then

output <= rx and ry;

end if;

if(sele = "0101") then

output <= rx or ry;

end if;

if(sele = "0110") then

tempoutt:="00000000";

output <= std\_logic\_vector (tempoutt);

end if;

if(sele = "0111") then

tempoutt := "00000001";

output <= std\_logic\_vector (tempoutt);

end if;

if(sele = "1111") then

if ( rx < ry ) then

tempoutt := "00000001";

output <= std\_logic\_vector (tempoutt);

end if;

end if;

if(sele = "1000") then

tempoutt := tempBB;

output <= std\_logic\_vector (tempoutt);

end if;

when "1000" => null;

when "1001" => null;

when "1010" => null;

when "1011" => null;

when "1100" => null;

when "1101" => null;

when "1110" => null;

when "1111" => null;

when others => tempout := "00000000";

end case ;

-- output <= std\_logic\_vector (tempoutt); -- all unsigned is assigned to output

-- output <= std\_logic\_vector (tempout); -- all signed is assigned to output

end process;

end logic;