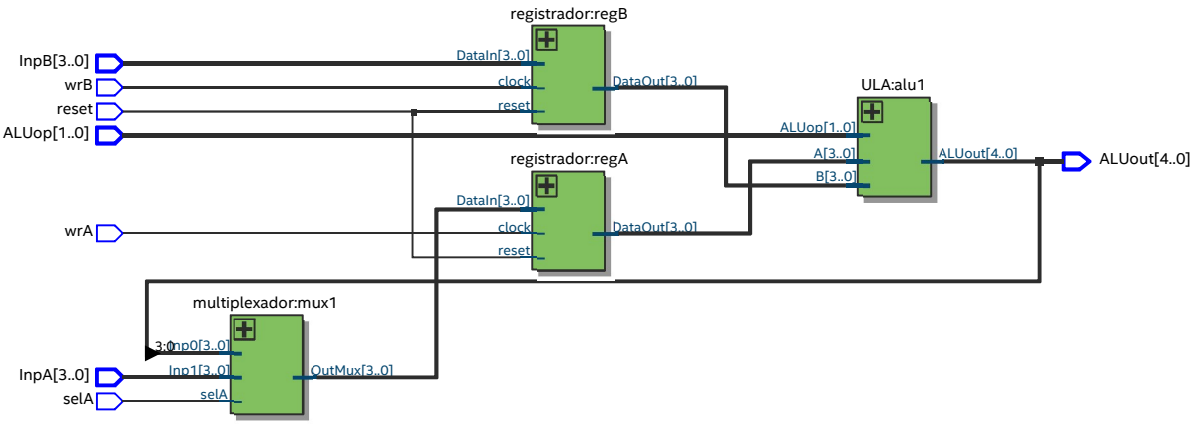
registrador:regB
InpB[3..0]
wrB
reset
ALUop[1..0]
DataIn[3..0]
clock
reset
DataOut[3..0]
ULA:alu1
ALUop[1..0]
A[3..0]
B[3..0]
ALUout[4..0]
ALUout[4..0]
registrador:regA
wrA
DataIn[3..0]
clock
reset
DataOut[3..0]
multiplexador:mux1
3:0
Inp0[3..0]
InpA[3..0]
Inp1[3..0]
selA
OutMux[3..0]

```vhdl
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.numeric_std.all;

entity blocoOperacional is
    port
    (
        InpA,InpB: in unsigned(3 downto 0);
        selA: in std_logic;
        wrA,wrB: in std_logic;
        ALUop: in std_logic_vector(1 downto 0);
        reset: in std_logic;

        ALUout: buffer unsigned(4 downto 0)
    );
end entity;

architecture behavior of blocoOperacional is

    component multiplexador is
        port
        (
            Inp0,Inp1: in unsigned(3 downto 0);
            selA: in std_logic;

            OutMux: out unsigned(3 downto 0)
        );
    end component;
    signal OutMux: unsigned(3 downto 0);

    component registrador is
        port
        (
            DataIn: in unsigned(3 downto 0);
            clock, reset: in std_logic;

            DataOut: out unsigned(3 downto 0)
        );
    end component;
    signal DataOutA,DataOutB: unsigned(3 downto 0);

    component ULA is
        port
        (
            A,B: in unsigned(3 downto 0);
            ALUop: in std_logic_vector(1 downto 0);

            ALUout: out unsigned(4 downto 0)
        );
    end component;

    --signal ALUout4bits: unsigned(3 downto 0);

begin

   --ALUout4bits <= ALUout(3 downto 0);

    mux1: multiplexador port map(ALUout(3 downto 0), InpA, selA, OutMux);
    regA: registrador port map(OutMux, wrA, reset, DataOutA);
    regB: registrador port map(InpB, wrB, reset, DataOutB);
    alu1: ULA port map(DataOutA, DataOutB, ALUop, ALUout);

end architecture;
```