

alucontrol

```

library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;

entity ALUControl is
  port(
    -- Inputs
    ALUOp : in std_logic_vector( 2 downto 0 );
    funct : in std_logic_vector( 5 downto 0 );

    -- Outputs
    ALUCtrl : out std_logic_vector( 2 downto 0 );
  end ALUControl;

architecture bhv of ALUControl is

begin
  process(ALUOp, funct)
  begin
    case ALUOp is
      -- when 00 or 01, opcode doesn't matter
      when "000"   => ALUCtrl <= "010"; -- add
      when "001"   => ALUCtrl <= "110"; -- sub
      when "110"   => ALUCtrl <= "000"; -- and
      when "101"   => ALUCtrl <= "001"; -- or
      when "100"   => ALUCtrl <= "100"; -- xor

      when others =>

        -- when 10, send different signal depending on opcode
        case funct(5 downto 0) is
          when "100000" => ALUCtrl <= "010"; -- add
          when "100010" => ALUCtrl <= "110"; -- sub
          when "100100" => ALUCtrl <= "000"; -- and
          when "100101" => ALUCtrl <= "001"; -- or
          when "101010" => ALUCtrl <= "111"; -- slt
          when "100110" => ALUCtrl <= "100"; -- xor
          when "000000" => ALUCtrl <= "011"; -- sll
          when others =>
            end case;
        end case;
      end process;
    end bhv;
  end bhv;
end bhv;

```