

```

-- Engineer: Greg Donohoe
--
-- Create Date: 08:59:31 03/06/2008
-- Module Name: add_sub_rev2 - Behavioral
-- Project Name: ECE241 Lab 8 example
--
-- Revision 0.02 – Changed “Additional comments” description, and the line
-- add_sub_out<= result(3 downto 0);
-- Additional Comments:
-- Implements a 4-bit adder-subtractor with carry in and carry out.
-- Creates 5-bit internal signals irega and iregb, padded on the left
-- with '0', to perform the operations.
-- 5-bit internal signal vector "result" holds the result of the operations.
-- Bits 3 down to 0 of "result" are routed to the output, "add_sub_out".
-- Bit 4, the MSB of "result" catches the carry, which is then routed to
-- the c_out bit.
--

```

```

-----
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity add_sub is
    Port ( rega : in STD_LOGIC_VECTOR (3 downto 0);
          regb : in STD_LOGIC_VECTOR (3 downto 0);
          control: in STD_LOGIC;
          c_in : in STD_LOGIC;
          add_sub_out : out STD_LOGIC_VECTOR (3 downto 0);
          c_out : out STD_LOGIC);
end add_sub;

architecture Behavioral of add_sub is
    signal irega, iregb, result: STD_LOGIC_VECTOR(4 downto 0);
begin
    irega <= '0' & rega;
    iregb <= '0' & regb;
    result <= irega + iregb + c_in when control = '0' else
              irega - iregb;
    add_sub_out <= result(3 downto 0);
    c_out <= result(4);
end Behavioral;

```