```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
library work;
use work.costanti.all;

entity RegAdder is
    generic (nb:integer:=nbitin);
    Port ( A : in STD_LOGIC_VECTOR (nb-1 downto 0);
           B : in STD_LOGIC_VECTOR (nb-1 downto 0);
           clk : in STD_LOGIC;
           RSum : out STD_LOGIC_VECTOR (nb downto 0));
end RegAdder;

architecture Behavioral of RegAdder is
component Registro is
    generic (nb: integer);
    Port ( D : in STD_LOGIC_VECTOR (nb-1 downto 0);
           clk : in STD_LOGIC;
           Q : out STD_LOGIC_VECTOR (nb-1 downto 0));
end component;
component Adder is
    generic (nb: integer);
    Port ( A : in STD_LOGIC_VECTOR (nb-1 downto 0);
           B : in STD_LOGIC_VECTOR (nb-1 downto 0);
           Sum : out STD_LOGIC_VECTOR (nb downto 0));
end component;
signal RA, RB: STD_LOGIC_VECTOR (nb-1 downto 0);
signal Sum: STD_LOGIC_VECTOR (nb downto 0);
begin
  RegA: Registro generic map (nb) port map (A,clk,RA);
  RegB: Registro generic map (nb) port map (B,clk,RB);
  Add:  Adder generic map (nb) port map (RA,RB,Sum);
  RegS: Registro generic map (nb+1) port map (Sum,clk,RSum);
end Behavioral;
```