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

entity Sim_RegAdd is
 generic (nb:integer:=nbitin);
end Sim_RegAdd;

architecture Behavioral of Sim_RegAdd is
component RegAdder is
    --generic (nb:integer);
    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 component;
signal IA, IB:  STD_LOGIC_VECTOR (nb-1 downto 0);
signal Iclk: STD_LOGIC:='0';
signal ORSum: STD_LOGIC_VECTOR (nb downto 0);
begin
  CUT: RegAdder port map (IA,IB,Iclk,ORSum);
  process
  begin
    wait for 5ns;
    Iclk<=not Iclk;
  end process;
  process
  begin
    --t=0
    wait for 110ns; -- attesa per il global reset
    IA<=(others=>'1');
    IB<=(others=>'1');
    wait for 10 ns;
    IB<=(others=>'0');
    wait for 10 ns;
  end process;
end Behavioral;
```