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

entity Complementer is
    Port ( A : in STD_LOGIC_VECTOR (nb-1 downto 0);
           B : in STD_LOGIC_VECTOR (nb-1 downto 0);
           C : in STD_LOGIC;
           Ris : out STD_LOGIC_VECTOR (nb+1 downto 0));
end Complementer;

architecture Behavioral of Complementer is
signal OpB: STD_LOGIC_VECTOR (nb downto 0);
signal p, g: STD_LOGIC_VECTOR (nb+1 downto 0);
signal rip: STD_LOGIC_VECTOR (nb+2 downto 0);
begin
  rip(0)<=C;
  with C select
  OpB<=(B & '0') when '0',
       (not (B & '0')) when '1',
       (others=>'X') when others;
  p<=(A(nb-1)&A&'0') xor (OpB(nb)&OpB);
  g<=(A(nb-1)&A&'0') and (OpB(nb)&OpB);
  rip(nb+2 downto 1)<= g or (p and rip(nb+1 downto 0));
  Ris<=p xor rip(nb+1 downto 0);
end Behavioral;
```