**PRE LAB 2**

**Exercise 1:**

library ieee;

use ieee.std\_logic\_1164.all;

entity PreEx1 is

    port(a, b, ci: in  std\_logic;

            co, s: out std\_logic);

end PreEx1;

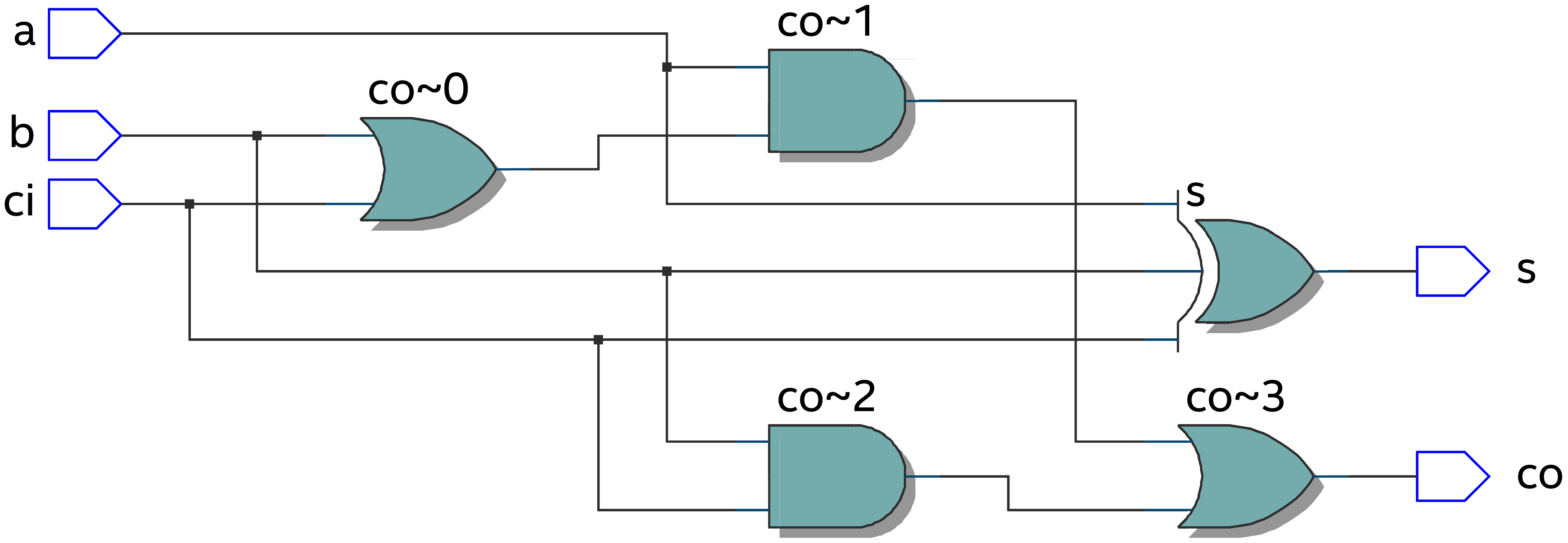
architecture Full\_Adder of PreEx1 is

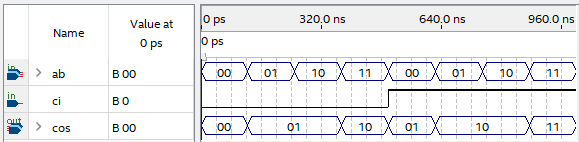
begin

    s <= a xor (b xor ci);

    co <= (a and (b or ci)) or (b and ci);

end Full\_Adder;



****

**Exercise 2:**

library ieee;

use ieee.std\_logic\_1164.all;

entity PreEx2 is

    port(    R,  S, CLK: in     std\_logic;

         Rg, Sg, Qa, Qb: buffer std\_logic);

end PreEx2;

architecture RS\_Latch of PreEx2 is

begin

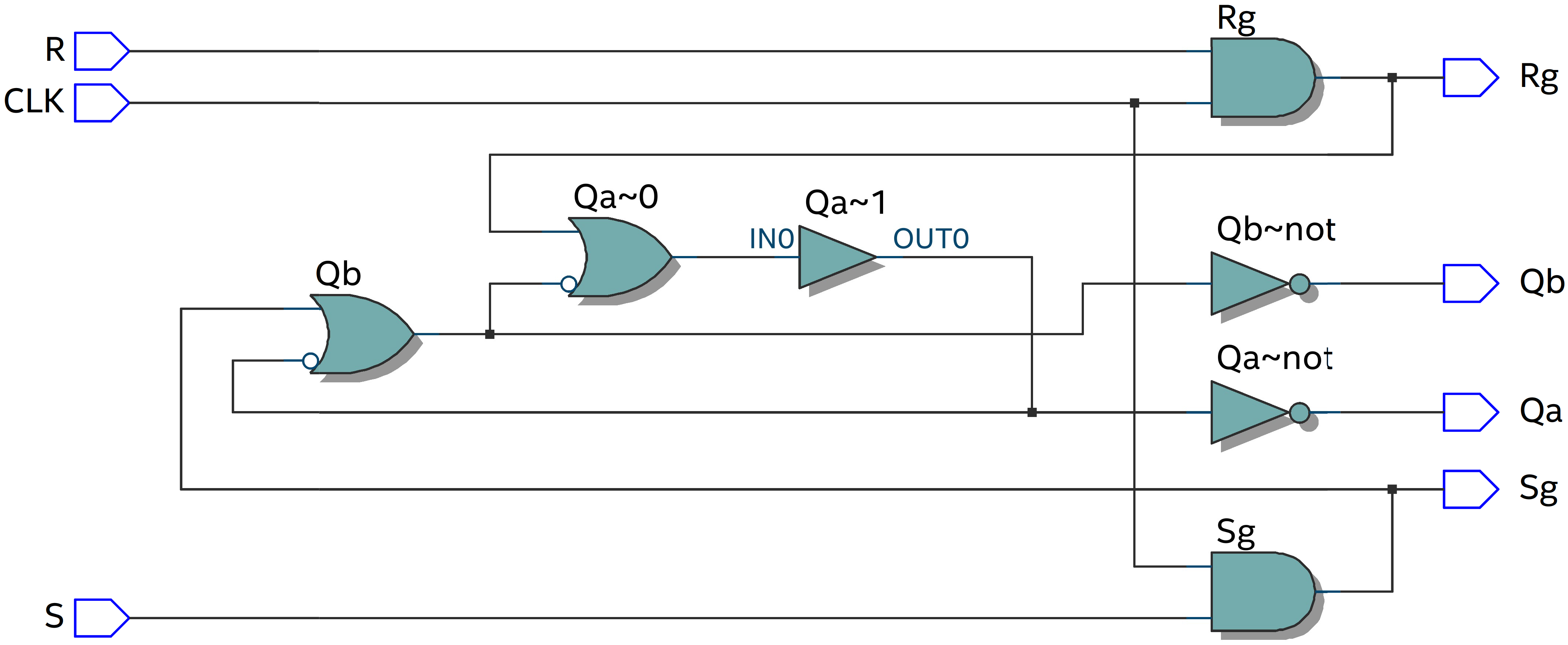
    Rg <= R and CLK;

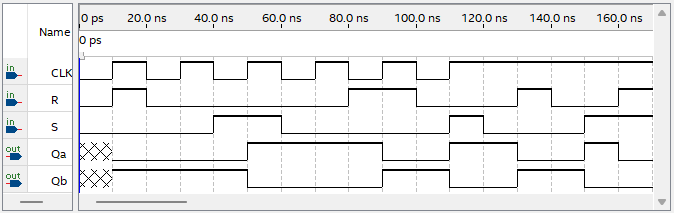
    Sg <= S and CLK;

    Qa <= Rg nor Qb;

    Qb <= Sg nor Qa;

end RS\_Latch;



****

**Exercise 3:**

library ieee;

use ieee.std\_logic\_1164.all;

entity PreEx3 is

    port(        D, CLK: in     std\_logic;

         Rg, Sg, Qa, Qb: buffer std\_logic);

end PreEx3;

architecture D\_Latch of PreEx3 is

begin

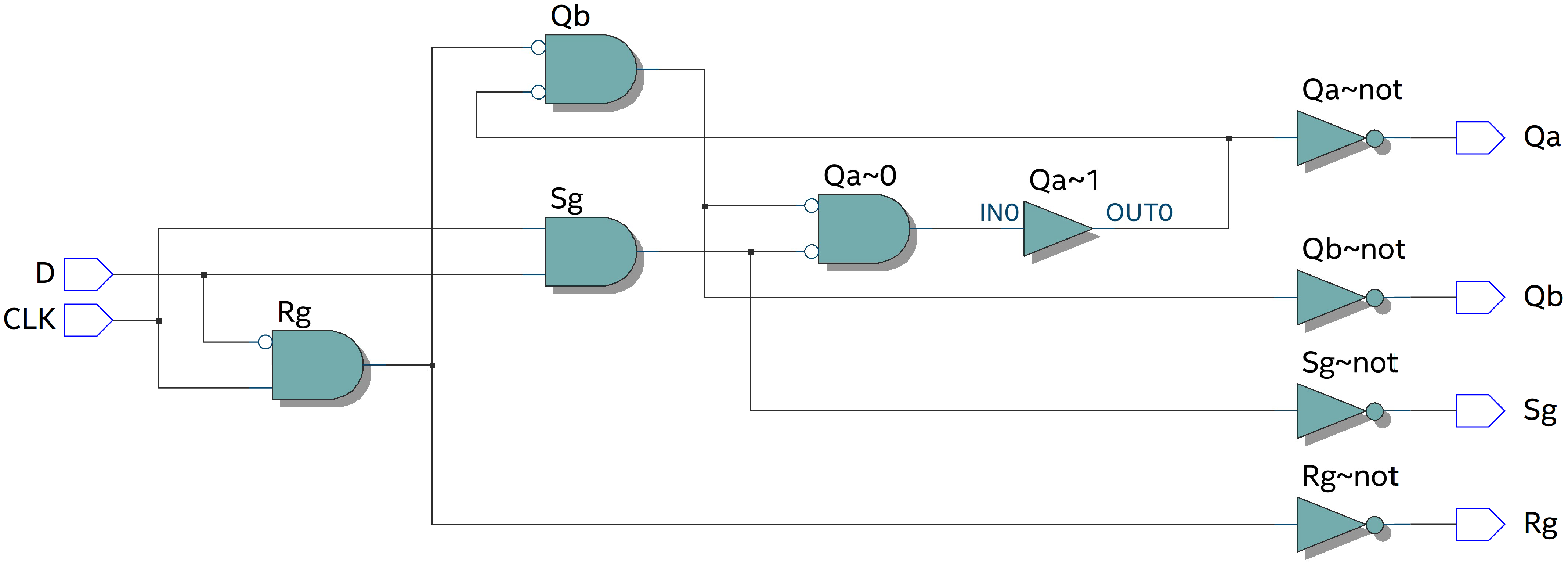
    Rg <= (not D) nand CLK;

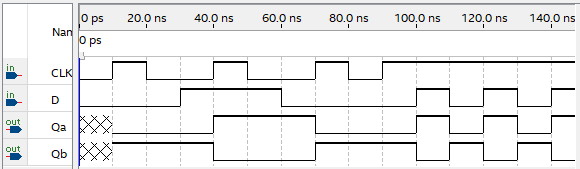
    Sg <= D nand CLK;

    Qa <= Sg nand Qb;

    Qb <= Rg nand Qa;

end D\_Latch;



****