```
library ieee;
use ieee.std_logic_1164.all;


entity CLA_block is
  port(
    a, b : in std_logic_vector(3 downto 0);
    cin  : in std_logic;
    s    : out std_logic_vector(3 downto 0);
    cout : out std_logic
  );
end entity CLA_block;



architecture mixed of cla_block is
   signal p, g : std_logic_vector(3 downto 0);
   signal c : std_logic_vector(4 downto 0);
   signal p30, g30 : std_logic;

    component fulladder is
      port(
            a, b, cin : in std_logic; -- inputs
            s : out std_logic; -- sum
            cout : out std_logic -- carry
        );
     end component;

begin
    p <= a or b;
    g <= a and b;
    p30 <=  p(0) and p(1) and p(2) and p(3);
    g30 <= g(3) or (p(3) and (g(2) or (p(2) and (g(1) or (p(1) and g(0))))));
    cout <= g30 or (p30 and cin);
    c(0) <= cin;

  f1: fulladder port map(
      a => a(0),
      b => b(0),
      cin => c(0),
      s => s(0),
      cout => c(1)
  );
 f2: fulladder port map(
      a => a(1),
      b => b(1),
      cin => c(1),
      s => s(1),
      cout => c(2)
  );
 f3: fulladder port map(
      a => a(2),
      b => b(2),
      cin => c(2),
      s => s(2),
      cout => c(3)
  );
 f4: fulladder port map(
```

```
      a => a(3),
      b => b(3),
      cin => c(3),
      s => s(3),
      cout => c(4)
  );
end architecture mixed;
```