```
1   library ieee;
2   use ieee.std_logic_1164.all;
3
4
5   entity CLAdder is
6      port(Xin: in std_logic_vector(15 downto 0);
7           Yin: in std_logic_vector(15 downto 0);
8           Cin: in std_logic;
9           Sum: out std_logic_vector (15 downto 0);
10          Cout: out std_logic
11          );
12  end CLAdder;
13
14
15  architecture behaviour of CLAdder is
16
17  signal h_sum : std_logic_vector(15 downto 0);
18  signal G: std_logic_vector(15 downto 0);
19  signal P: std_logic_vector(15 downto 0);
20  signal CarryIn: std_logic_vector(15 downto 1);
21
22  begin
23     h_sum<= Xin xor Yin;
24     G <= Xin and Yin;
25     P <= Xin or Yin;
26
27     process (G,P, CarryIn, h_sum, Cin)
28     begin
29     CarryIn(1) <= G(0) or (P(0) and Cin);
30        inst: for i in 1 to 14 loop
31           CarryIn(i+1) <= G(i) or (P(i) and CarryIn(i));
32           end loop;
33     Cout <= G(15) or (P(15) and CarryIn(15));
34     end process;
35     Sum(0)<= h_sum(0) xor Cin;
36     Sum(15 downto 1)<= h_sum(15 downto 1) xor CarryIn(15 downto 1);
37
38  end behaviour;
39
40
```