```
1  library ieee;
2  use ieee.std_logic_1164.all;
3  use ieee.std_logic_unsigned.all;
4
5  entity count is
6    port (
7      D         : in  std_logic_vector(3 downto 0);
8      Q         : out std_logic_vector(3 downto 0);
9      count, load : in  std_logic;
10     clock       : in  std_logic;
11     carry       : out std_logic := '0');
12 end count;
13
14 architecture behav of count is
15   signal Q_int : std_logic_vector(0 to 3);
16   signal here : std_logic := '0';
17     -- added for testing
18 begin
19   count_process: process (clock)
20   begin
21     if rising_edge(clock) and load='1' then
22       Q_int <= D;
23       carry <= '0';
```

```
24        elsif rising_edge(clock) and count='1' then
25          -- for testing
26          here <= '1', '0' after 5 ns;
27
28          if Q_int = "1111" then
29            carry <= '1';
30          else
31            carry <= '0';
32          end if;
33
34          Q_int <= Q_int + "0001";
35        end if;
36      end process;
37
38      -- Update Q whenever Q_int changes:
39      Q_update: process (Q_int)
40      begin
41        Q <= Q_int after 3 ns;
42      end process Q_update;
43
44    end behav;
```