```
 1: library STD;
 2: library IEEE;
 3: use IEEE.std_logic_1164.all;
 4:
 5: entity Cache_Cell_Valid is
 6:     port(
 7:         s       :   in  std_logic;
 8:         r       :   in  std_logic;
 9:         Rd_En   :   in  std_logic;
10:         Wr_En   :   in  std_logic;
11:         q       :   out std_logic
12:     );
13: end Cache_Cell_Valid;
14:
15: architecture structural of Cache_Cell_Valid is
16:
17:     component SRlatch
18:     port(
19:         s   :   in  std_logic;
20:         r   :   in  std_logic;
21:         q   :   out std_logic
22:     );
23:     end component;
24:
25:     component tx
26:     port(
27:         sel     :   in  std_logic;
28:         selnot  :   in  std_logic;
29:         input   :   in  std_logic;
30:         output  :   out std_logic
31:     );
32:     end component;
33:
34:     component nor2
35:     port(
36:         in1     :   in  std_logic;
37:         in2     :   in  std_logic;
38:         out1    :   out std_logic
39:     );
40:     end component;
41:
42:     component invX1
43:     port(
44:         in1     :   in  std_logic;
45:         out1    :   out std_logic
46:     );
47:     end component;
48:
49:     signal  q1      :   std_logic;
50:     signal  nRd_En  :   std_logic;
51:     signal  out_en  :   std_logic;
52:
53:     for or1 :   nor2 use entity work.nor2(structural);
54:     for inv :   invX1 use entity work.invX1(structural);
55:     for sr  :   SRlatch use entity work.SRlatch(structural);
56:     for t   :   tx  use entity work.tx(structural);
57:
58: begin
59:
60:     or1 :   nor2     port map(Rd_En, Wr_En, nRd_En);
61:     inv :   invX1   port map(nRd_En, out_en);
62:     sr  :   SRlatch port map(s, r, q1);
63:     t   :   tx      port map(out_en, nRd_En, q1, q);
64:
65: end structural;
```