Assignment-2 (CS252) by Utkarsh Ranjan

8:3 Encoder with enabler

- The Top-Level entity EightbyThreeEncode take as input
    1. A 8-bit logic signal (i_7 to i_0)
    2. enabler signal (en)
- It converts this signal into a 3-bit output signal (z_2 to z_0)
- Based on the 8 inputs one of the 3 outputs is selected.
- The truth table for 8 to 3 encoder is shown in table (1).

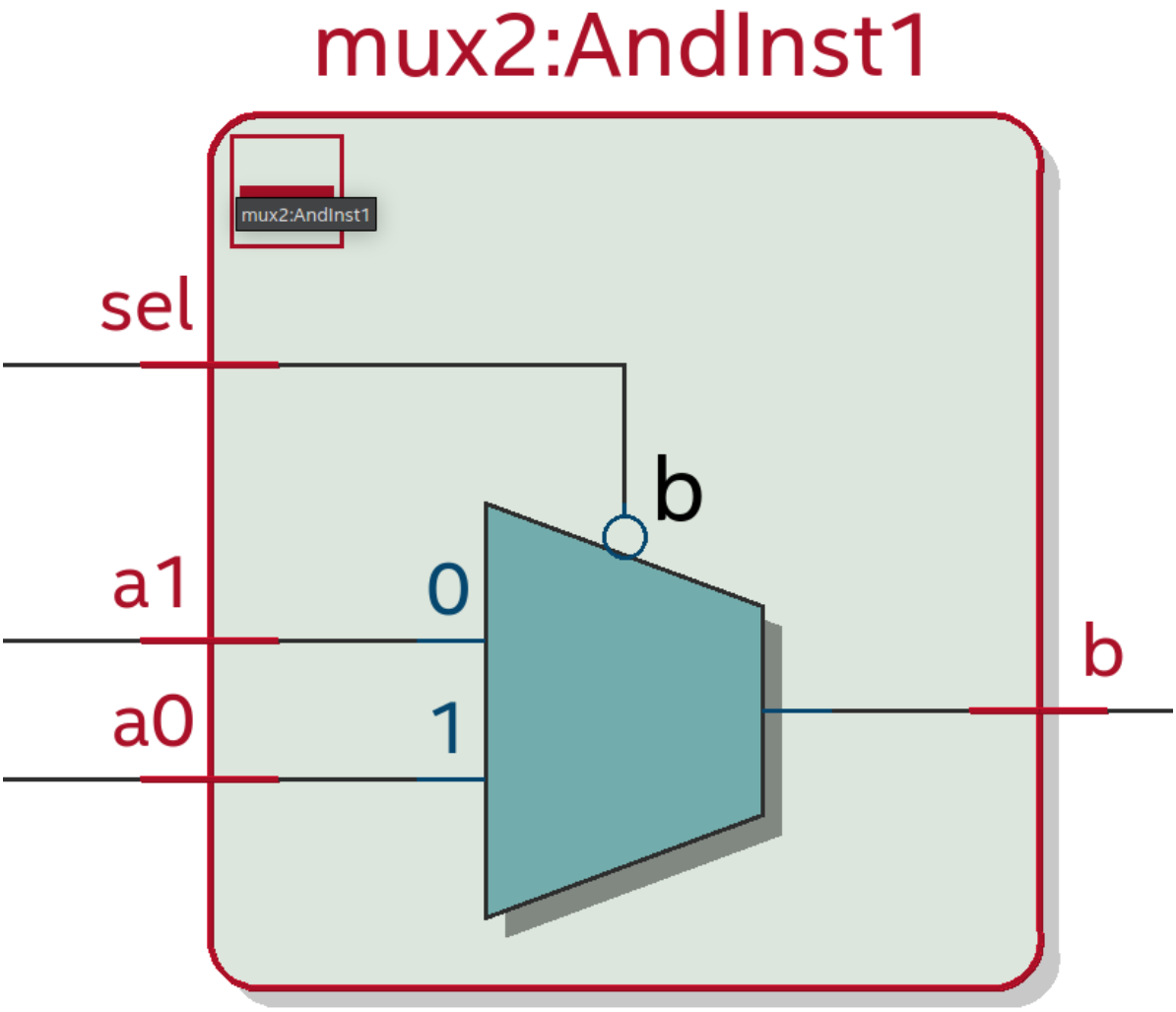
I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	**en**	Z_2	Z_1	Z_0
1	0	0	0	0	0	0	0	1	0	0	0
0	1	0	0	0	0	0	0	1	0	0	1
0	0	1	0	0	0	0	0	1	0	1	0
0	0	0	1	0	0	0	0	1	0	1	1
0	0	0	0	1	0	0	0	1	1	0	0
0	0	0	0	0	1	0	0	1	1	0	1
0	0	0	0	0	0	1	0	1	1	1	0
0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	0	0	0	1	1	1

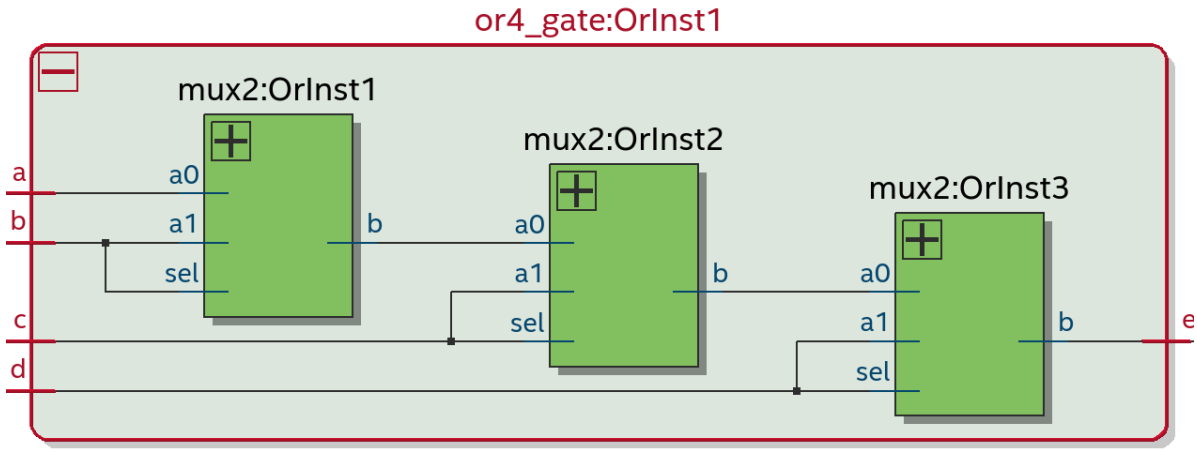
- From the truth table, it is seen that only one of three outputs (z_0 to z_1) is selected based on 8 select inputs.
- From the truth table, the logic expressions for outputs (with en == 1) can be written as follows:

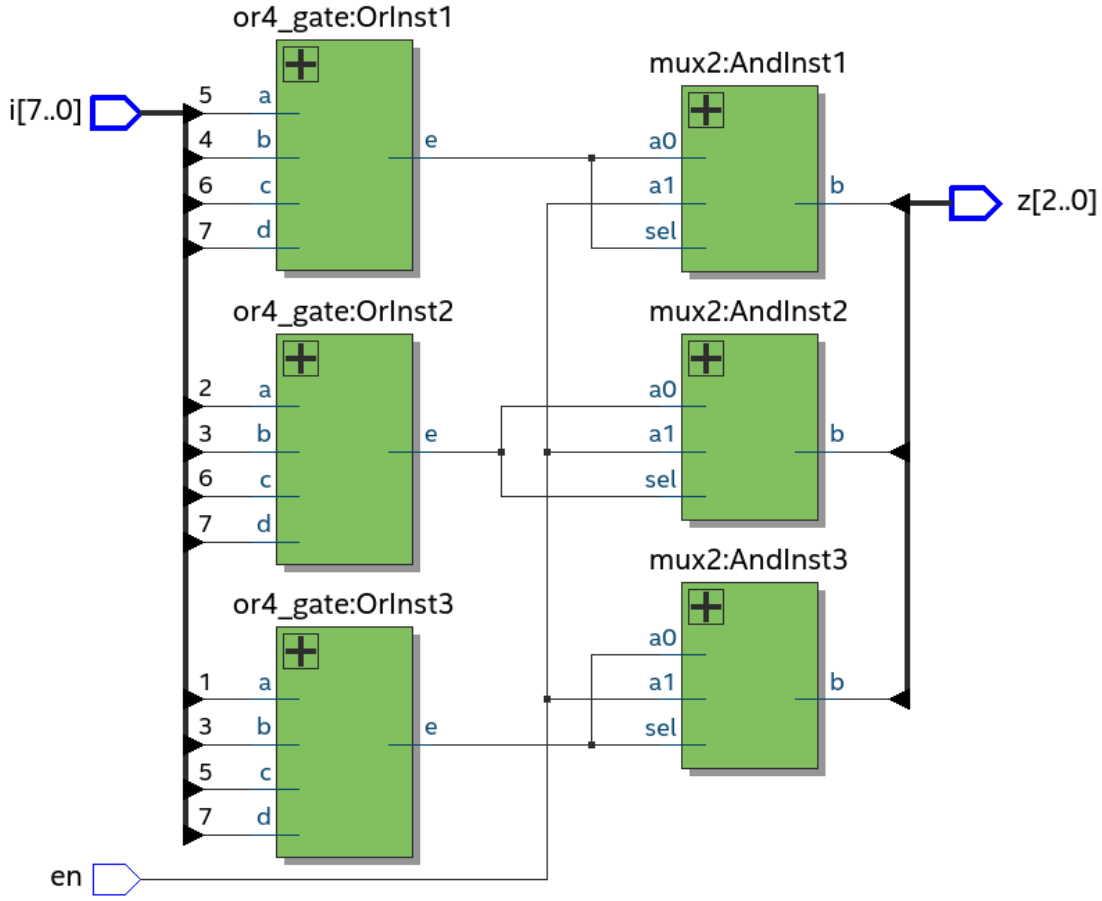
$$z_2 = I_7 + I_6 + I_5 + I_4$$
$$z_1 = I_7 + I_6 + I_3 + I_2$$
$$z_0 = I_7 + I_5 + I_3 + I_1$$

Structural Description

- 3 x 3 = 9 OR GATE:
  1. To perform the three OR operation for each ouput - 3 OR GATES are used
  2. Each OR gate used 1 Multiplexer, which is obtained by making the selector input of the multiplexer equal to one of the two inputs
  3. So for 3 such output, 9 OR GATE and hence 9 MUXES are used
- 3 x 1 = 3 AND GATE:
  1. To And the ouput of the Or gates with the enabler signal - 1 AND GATE was used.
  2. Thus using 3 AND Gates for each ouput, a total of 3 Multiplexer was used (Obtained my making one of the input signal 0 and using selector and other input to get their and as the output).
- Total Number of Mux used = 1 * OR Gate + 1 * AND Gate = 1 * 9 + 1 * 3 = **12 MUXES**




or4_gate:OrInst1
mux2:OrInst1
mux2:OrInst2
mux2:OrInst3
a
b
c
d
a0
a1
sel
b
e


or4_gate:OrInst1
or4_gate:OrInst2
or4_gate:OrInst3
mux2:AndInst1
mux2:AndInst2
mux2:AndInst3
i[7..0]
z[2..0]
en
5
4
6
7
2
3
1
a
b
c
d
e
a0
a1
sel
b