# CS 232 Lab 2 Q1

Arya Amin 200050014

22nd January 2022

**2×1 Muxes**(mux.vhd)

I have designed the 2×1 Mux using behavioural code. i.e. if selector is 1 the output is in1 else the output is in0. Further used this mux to create OR and AND Gates

**8×3 Encoder**(EightbyThreeEncode.vhd)

I have designed the Encoder circuit using only 2×1 Muxes.

The encoder Logical Expression are as follows,

z(0) = en.(i(1) + i(3) + i(5) + i(7))
z(1) = en.(i(2) + i(3) + i(6) + i(7))
z(2) = en.(i(4) + i(5) + i(6) + i(7))

To implement the OR Gate used in the above logical Expression, I have used muxes by keeping selector same as in1.

port map (in0=>i(1), in1=>i(3), sel=>i(3), out0=>p)
p = i(1) + i(3)

port map (in0=>i(5), in1=>i(7), sel=>i(7), out0=>q)
q = i(5) + i(7)

port map (in0=>p, in1=>q, sel=>q, out0=>w1)
w1 = p + q = i(1) + i(3) + i(5) + i(7)

Similarly, w2 = r + s = i(2) + i(3) + i(6) + i(7)
and w3 = t + u = i(4) + i(5) + i(6) + i(7).
Here p,q,r,s,t,u are signals.

Similarly to implement the enabler pin, I have used mux as AND Gate by keeping selector same as in0.

port map (in0=>en, in1=>w1, sel=>en, out0=>z(0))
z(0) = w1.en

Therefore, when Enabler pin en=0 output is 000 else the signal will pass.

OR GATES = 9 and AND GATES = 3
Thus total MUXES = 9 + 3 = 12