Note: for all of the following problems you can assume someone else is providing you with multiplexors, decoders, and/or flip-flops (you do not have to build them yourself). Thus, if you need to use one of these elements, you can just draw the box symbol for it, with each of the inputs and outputs labeled, like we have done in class for all the state machine and counter problems.

1. Implement the following function in the specified technologies. You may use a minimum number of extra gates where needed:

\[ F = (A + B + D) + BCD + ACD + BC \]

a.) 4:16 decoder:

b.) 16:1 MUX

c.) 8:1 MUX

2. In class we built a 4:2 priority encoder, which required a VALID output signal and allowed multiple inputs to be on at a time. An alternative is to simply define address 0 as the invalid state, and not allow an input wire D0.

Design a 3:2 encoder. It will take inputs D3, D2, and D1, and produce A1 and A0. If exactly one of the inputs is true, the output is the binary number corresponding to the input number (D3 -> 11, D2 -> 10, D1 -> 01). If no input is true, or multiple inputs are true, the output will be 00.