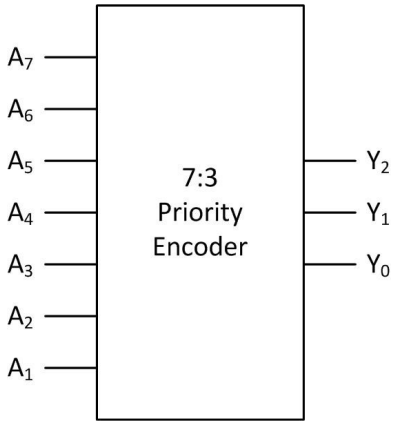
Priority Encoder

Problem Statement

Recall the priority circuit described in class that takes N inputs A[N-1:0] and produces N outputs Y[N-1:0], with at most one output asserted corresponding to the highest priority (most significant) input asserted. If no inputs are asserted, then no output is asserted.

A priority encoder is similar but encodes the output to save bits. Let $N = 2^M$. The N-1:M priority encoder takes N-1 inputs A[N-1:1] and produces an M-bit output Y[M-1:0] indicating the most significant bit asserted in A, or 0 if no bits of A are asserted. For example, a 7:3 priority encoder takes a 7-bit input and produces a 3-bit output. If A=0100110, Y=110 because A[6] is the most significant bit asserted. If A = 0000111, Y = 011 because A[3] is the most significant bit asserted. If A = 0000000, Y = 000 because no input bits are asserted.



The goal of this lab problem is for you to design a gate-level implementation of a 7:3 priority encoder, model it with structural SystemVerilog, and debug your design. Use only AND, OR, and NOT gates in your solution. Simplify into minimal sum-of-products form.

Start with the files below. Modify the priorityencoder module but don't change the testbench because that would mess up the hash. Note that the test vectors are missing most of the expected outputs. Edit the file to add the missing outputs, but do not change the order of the vectors.