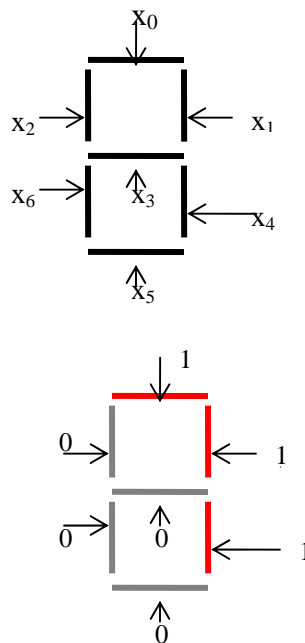


CS 135 Team Assignment 3. Due September 29th, 12noon.

Blackboard Submission.

1. Question 3.13
2. Question 3.23
3. Question 3.22 from textbook
4. In this question you will design the control logic for a 7 element LED display to represent decimal numbers. Each LED element is controlled by one line (bit) – to turn on that element you will have to set the control line to a 1 (to turn off the element you set the line to a 0). Call the 7 control lines ($x_0, x_1, x_2, x_3, x_4, x_5, x_6$). The figure below shows the LED display and the control lines. In the second figure, the number 7 is displayed by setting the lines $x_0=1, x_1=1, x_4=1$ and x_2, x_3, x_5, x_6 are all set to zero.



This LED display is to be controlled by an unsigned four bit binary number (a_3, a_2, a_1, a_0) – this number corresponds to a decimal digit between 0 and 9. For example, to set the LED to display 7 the input four bit number is 0111. If the decimal equivalent of the input is greater than 9, then the LED display is set to display nothing (i.e., all elements are turned off). For example, if the input is 1100 (decimal equivalent 12) then all the LED display control lines are set to 0.

You are required to design a combinational circuit that takes as input the four bit number and displays its decimal equivalent (between 0 and 9).

- (a) First write the truth table for this circuit. The input is the four bit number (a_3, a_2, a_1, a_0) and the output is the values of the control signals $(x_0, x_1, x_2, x_3, x_4, x_5, x_6)$ for each combination of input values.
- (b) Next, implement a circuit for the truth table in part (a) using **only** the basic logic gates – AND, OR, NOT, XOR, NAND, NOR.
- (c) Can you implement the same truth table in (a) using a more complex combinational logic device as the building block. Hint: Read section 3.3.4