1. Specify the minimum sizes of ROMs (number of words and number of bits per word) that will accommodate the truth table for the following combinational circuits:

   a. A binary multiplier that multiplies two 8-bit binary numbers.
   b. A binary divider that returns the integer part of the quotient of two 4-bit binary numbers (assume that the output is all 1’s in the case of divide-by-zero).
   c. A converter which maps an 8-bit binary number to the corresponding three-digit BCD representation.

2. Provide either a sum of products or an AND-OR-INVERT Boolean expression for each of the functions $F_1$, $F_2$, and $F_3$ computed by the PLA circuit below:
3. Following the instructions in Example 7.2 in the textbook, program a PLA to implement the three Boolean functions

\[ F_1(x, y, z) = \Sigma(1, 3, 5, 6) \]
\[ F_2(x, y, z) = \Sigma(0, 1, 6, 7) \]
\[ F_3(x, y, z) = \Sigma(0, 1, 2, 4, 7) \]

a. Start by making K-maps for each function to write it as both a minimized SOP expression, as well as a minimized AOI expression. In the end, you should have six minimized boolean expressions, each either OR-ing or NOR-ing together several product terms.

b. Try to re-use the largest possible number of product terms by choosing either the SOP or the AOI expression for each function. Although there is also a six term solution, I was able to find one that uses just five unique product terms.

c. Print out a copy of the diagram at http://www.swarthmore.edu/NatSci/mzucker1/e15_f2015/pla_blank.pdf, and mark locations of fuses to program a PLA to implement the SOP/AOI expressions you determined above.