SANTA CLARA UNIVERSITY
Electrical Engineering Department

Homework 3 and 4 solutions

2.71 A given system has four sensors that can produce an output of 0 or 1. The system operates properly when exactly one of the sensors has its output equal to 1. An alarm must be raised when two or more sensors have the output of 1. Design the simplest circuit that can be used to raise the alarm.

2.71. Assuming that the condition where all sensors produce the output of 0 is a don't care, the complement of the desired function is

$$\overline{f} = \overline{x}_1\overline{x}_2\overline{x}_3 + \overline{x}_1\overline{x}_2\overline{x}_4 + \overline{x}_1\overline{x}_3\overline{x}_4 + \overline{x}_2\overline{x}_3\overline{x}_4$$

Then, $f = \overline{\overline{f}}$.

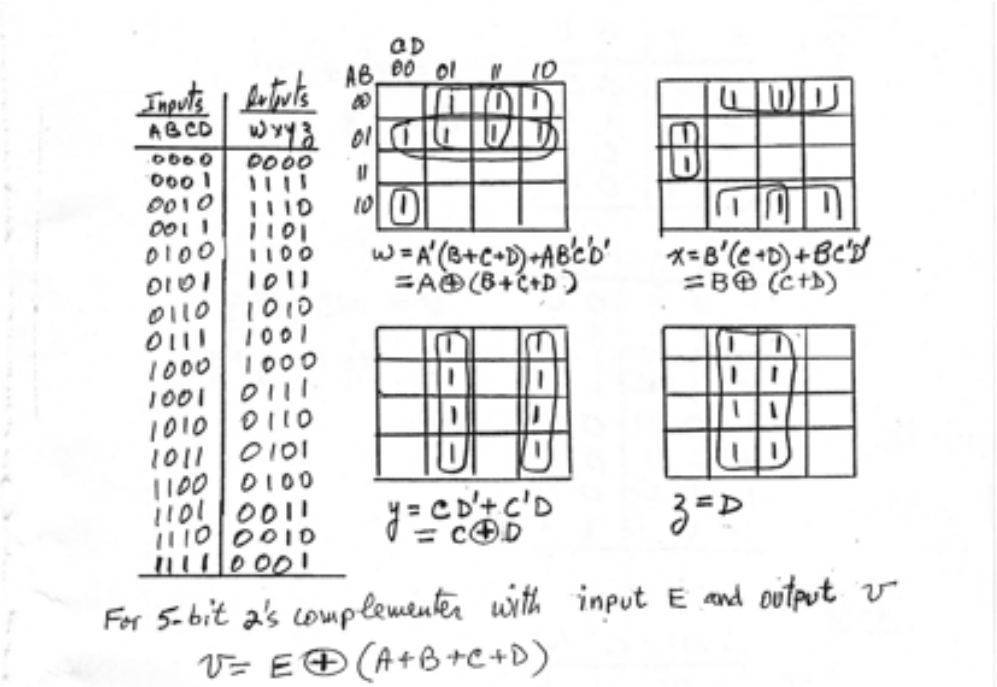
2.73 Find the minimum-cost circuit consisting only of two-input NAND gates for the function f (x1,..., x4) =Σm(0, 1, 2, 3, 4, 6, 8, 9, 12). Assume that the input variables are available in both uncomplemented and complemented forms. (Hint: Consider the complement of the function.)

2.73. Implement first the complement of f as

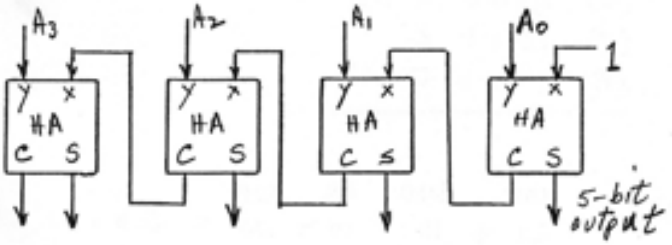
$$\begin{aligned} \overline{f} &= x_1x_3 + x_2x_4 \\ &= (x_1 \uparrow x_3) \uparrow (x_2 \uparrow x_4) \end{aligned}$$

Then $f = \overline{f} \uparrow \overline{f}$.

Problem 3 (4-bit 2's complementer):



Problem 1: 4-bit combinational circuit incrementer



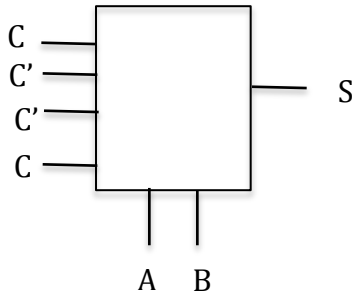
Problem 2: Implement a full adder with two 4-to-1 muxes.

The truth table for a full adder is:

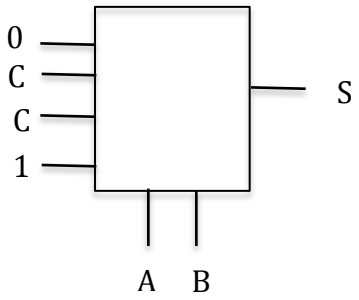
A	B	C	Cout	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0

1	1	0	1	0
1	1	1	1	1

Implementing the Sum with a 4-to-1 mux



Implementing the Carry with a 4-to-1 mux



3.18. Let $Y = y_3y_2y_1y_0$ be the 9's complement of the BCD digit $X = x_3x_2x_1x_0$. Then, Y is defined by the truth table

x_3	x_2	x_1	x_0	y_3	y_2	y_1	y_0
0	0	0	0	1	0	0	1
0	0	0	1	1	0	0	0
0	0	1	0	0	1	1	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	0	0
0	1	1	0	0	0	1	1
0	1	1	1	0	0	1	0
1	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0

This gives

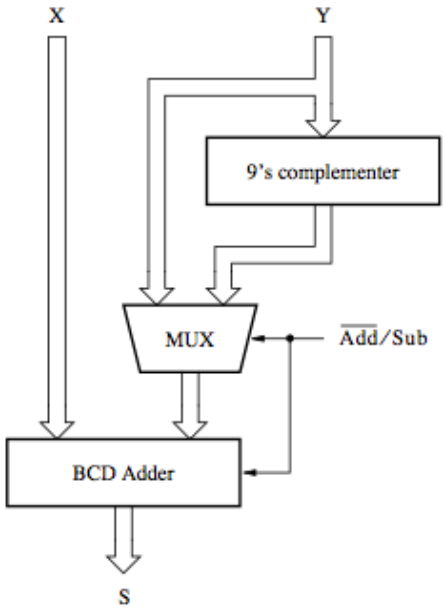
$$
\begin{aligned}
y_0 &= \overline{x}_0 \\
y_1 &= x_1 \\
y_2 &= \overline{x}_2x_1 + x_2\overline{x}_1 \\
y_3 &= \overline{x}_3\overline{x}_2\overline{x}_1
\end{aligned}
$$

3.19. BCD subtraction can be performed using 10's complement representation, using an approach that is similar to 2's complement subtraction. Note that 10's and 2's complements are the radix complements in number systems where the radices are 10 and 2, respectively. Let X and Y be BCD numbers given in 10's complement representation, such that the sign (left-most) BCD digit is 0 for positive numbers and 9 for negative numbers. Then, the subtraction operation $S = X - Y$ is performed by finding the 10's complement of Y and adding it to X, ignoring any carry-out from the sign-digit position.

For example, let $X = 068$ and $Y = 043$. Then, the 10's complement of Y is 957, and $S' = 068 + 957 = 1025$. Dropping the carry-out of 1 from the sign-digit position gives $S = 025$.

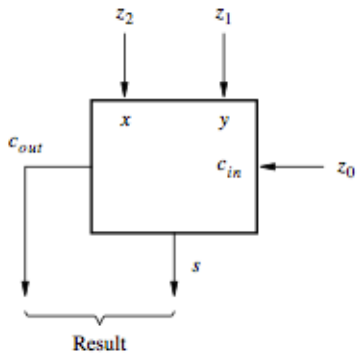
As another example, let $X = 032$ and $Y = 043$. Then, $S = 032 + 957 = 989$, which represents -11_{10}.

The 10's complement of Y can be formed by adding 1 to the 9's complement of Y. Therefore, a circuit that can add and subtract BCD operands can be designed as follows:



For the 9's complementer one can use the circuit designed in problem 3.18. The BCD adder is a circuit based on the approach illustrated in Figure 3.39.

3.21. A full-adder circuit can be used, such that two of the bits of the number are connected as inputs x and y, while the third bit is connected as the carry-in. Then, the carry-out and sum bits will indicate how many input bits are equal to 1.



3.22. Using the approach explained in the solution to problem 3.21, the desired circuit can be built as follows: