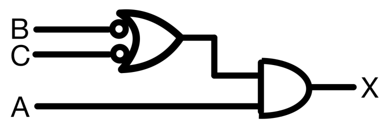
****
EE/CS 119A Homework 1
Edward Speer
October 7, 2024
****

1. **Problem 1**: *Simple Circuit*

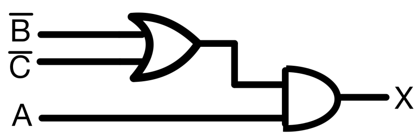
   From the problem description, either B or C must be inactive with A active for X to be active. Clearly, this means that $X = A(B' + C')$.

   Active high circuit:

   $X = A(B' + C')$

   Active low circuit:

   $X = A(B' + C')$

2. **Problem 2**: *Gray Code to Binary Converter*

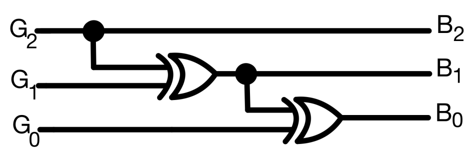
   - From inspection of the truth table, it is immediately apparent that $B_2 = G_2$.
   - For B_1, inspect the cases in which G_2 is active. There are 4 such cases: $(G_2, G_1, G_0) = \{(0,1,1), (0,1,0), (1,0,1), (1,0,0)\}$. It is clear that we may eliminate G_0 leaving the following: $(G_2, G_1) = (0,1), (1,0)$. This clearly shows $B_1 = G_2 \oplus G_1$
   - For B_0, enumerate the active cases as an equation and simplify:

   $$B_0 = (G_2G_1'G_0') + (G_2'G_1'G_0) + (G_2G_1G_0) + (G_2'G_1G_0')$$

   $$B_0 = G_2(G_1'G_0' + G_1G_0) + G_2'(G_1'G_0 + G_1G_0')$$

   $$B_0 = G_2(G_1 \oplus G_0)' + G_2'(G_1 \oplus G_0)$$

   $$B_0 = G_2 \oplus G_1 \oplus G_0$$

3. **Problem 3**: *Squaring Circuit*

First, create the truth table for the circuit. Then consider the active cases for each output bit seperately.

B_3	B_2	B_1	B_0	S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0	0
0	0	1	1	0	0	0	0	1	0	0	1
0	1	0	0	0	0	0	1	0	0	0	0
0	1	0	1	0	0	0	1	1	0	0	1
0	1	1	0	0	0	1	0	0	1	0	0
0	1	1	1	0	0	1	1	0	0	0	1
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	1	0	1	0	1	0	0	0	1
1	0	1	0	0	1	1	0	0	1	0	0
1	0	1	1	0	1	1	1	1	0	0	1
1	1	0	0	1	0	0	1	0	0	0	0
1	1	0	1	1	0	1	0	1	0	0	1
1	1	1	0	1	1	0	0	0	1	0	0
1	1	1	1	1	1	1	0	0	0	0	1

- For S_0 trivially notice from the truth table that $\boxed{S_0 = B_0}$
- For S_1, trivially notice that $\boxed{S_1 = 0}$ always
- For S_2, the cases are 0010, 0110, 010, 1110. This holds all cases for B_3 and B_2 so that they can be eliminated, leaving $\boxed{S_2 = B_1 B_0'}$
- For S_3, the cases are 0011, 0101, 1011, 1101. Clearly B_0 must be active. B_3 may be eliminated, since its value makes no difference. Notice that either B_2 or B_1 are always active but never both. Therefore, $\boxed{S_3 = B_0(B_2 \oplus B_1)}$.
- For S_4, the cases are 0100, 0101, 0111, 1001, 1011, 1100. Notice that for the cases where B_0 is active:

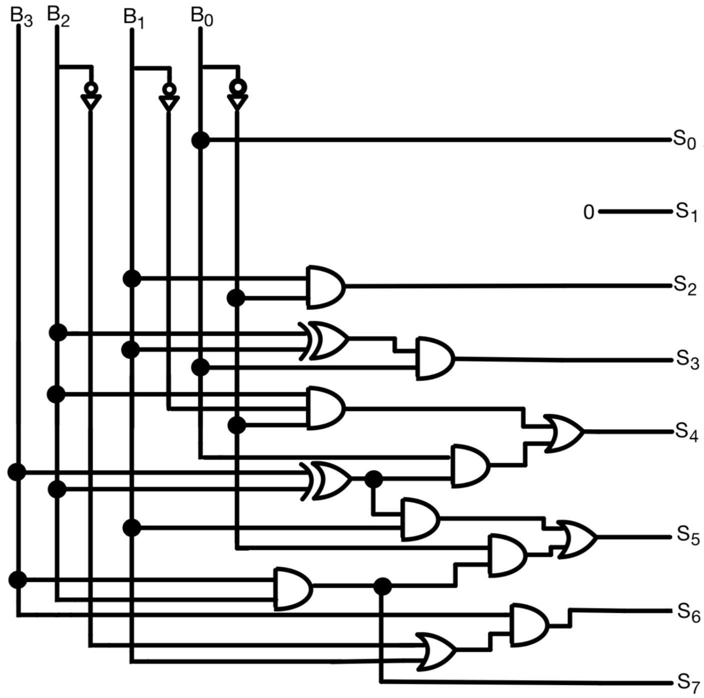
$$S_4 = B_0 B_3' B_2 + B_0 B_3 B_2'$$
$$S_4 = B_0(B_3' B_2 + B_3 B_2')$$
$$S_4 = B_0(B_3 \oplus B_2)$$

  This leaves 2 cases, 0100 and 1100. From these, B_3 is eliminated leaving $S_4 = B_0' B_1' B_2$. Therefore, $\boxed{S_4 = B_0(B_3 \oplus B_2) + B_0' B_1' B_2}$
- For S_5, the cases are 0110, 0111, 1010, 1011, 1101, 1111. Eliminating trivial terms, these cases are summed up as follows:

$$S_5 = B_1 B_2 B_3 + B_1 B_2' B3 + B_0 B_2 B_3$$
$$S_5 = B_1(B_2 B_3' + B_2' B_3) + B_0 B_2 B_3$$

  Then $\boxed{S_5 = B_1(B_2 \oplus B_3) + B_0 B_2 B_3}$
- For S_6, the cases are 1000, 1001, 1010, 1011, 1110, 1111. Immediately notice that B_3 always must be active. When B_0 is low, notice that B_1 and B_0 enumerate all cases and may be eliminated. This leaves only the cases where B_2 is active, in which case B_0 enumerates all of its possibilities but all other inputs are active. Remove the redundancy on B_2 from other cases, and combining all factors, $\boxed{S_6 = B_3(B_2' + B_1)}$.

- For S_7, the cases are the last 4 in the truth table. It is obvious to see that these are the only 4 cases where both B_3 and B_2 are active. Therefore, $\boxed{S_7 = B_2 B_3}$.



4. **Problem 4**: *Binary to BCD Hours Converter*

Notice from the truth table that $\boxed{B_0 = Q_0}$, and that $\boxed{B_2 = Q_2}$. Then solve for the remaining equations using Karnaugh maps.

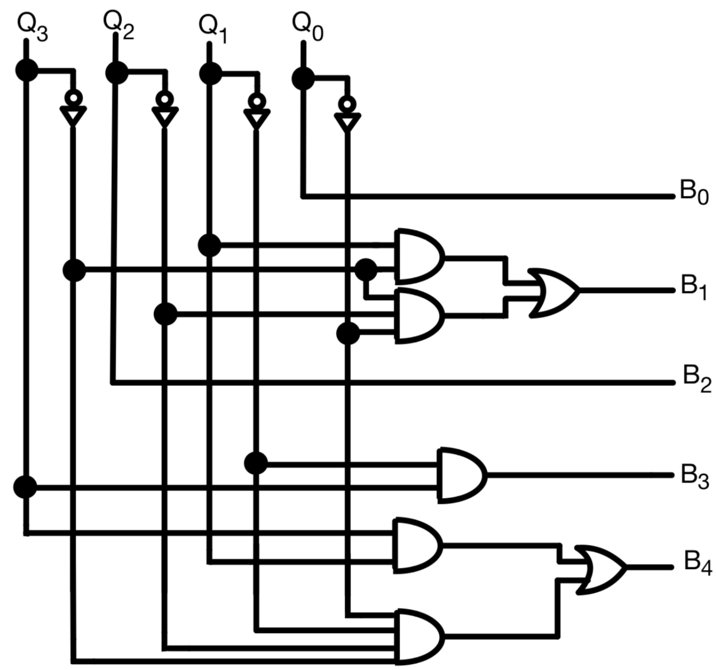
B1	Q1Q0			
Q3Q2	0, 0	0, 1	1, 1	1, 0
0, 0	1	0	1	1
0, 1	0	0	1	1
1, 1	x	x	x	x
1, 0	0	0	0	0
B3	Q1Q0			
Q3Q2	0, 0	0, 1	1, 1	1, 0
0, 0	0	0	0	0
0, 1	0	0	0	0
1, 1	x	x	x	x
1, 0	1	1	0	0
B4	Q1Q0			
Q3Q2	0, 0	0, 1	1, 1	1, 0
0, 0	1	0	0	0
0, 1	0	0	0	0
1, 1	x	x	x	x
1, 0	0	0	1	1

These Karnaugh maps reveal the following:

- $\boxed{B_1 = Q_3' Q_1 + Q_3' Q_2' Q_0'}$

- $\boxed{B_3 = Q_3Q_1'}$
- $\boxed{B_4 = Q_3'Q_2'Q_1'Q_0' + Q_3Q_1}$

AND/OR/INVERT circuit:



NAND/NAND circuit: