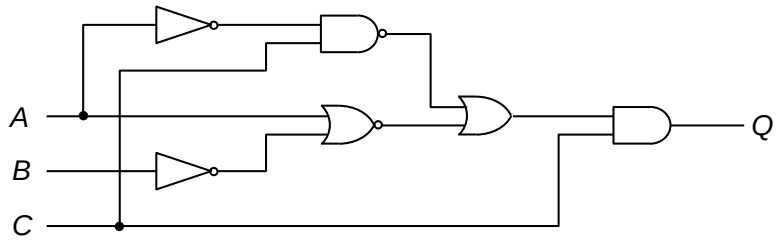
1. Consider the below mess of gates:



   (a) Convert the mess of gates into the equivalent boolean expression.

   (b) Use boolean algebra to reduce your boolean expression.

   (c) Implement your reduced boolean expression in gates.

   (d) On this sheet of paper, label the logic level of each wire for inputs: $(A, B, C) = (1, 1, 1)$.

2. Perform the following conversions:

   (a) Convert the decimal number 36 to binary.

   (b) Convert the decimal number 35 to octal.

   (c) Convert the hexadecimal number 34 to binary.

   (d) Convert the hexadecimal number 33 to decimal.

   (e) Convert the BCD number 0011,0010 to decimal.

   (f) Convert the BCD number 0011,0001 to binary.

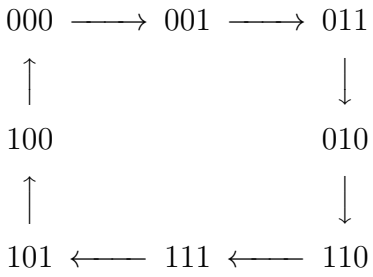
3. Reduce the following expressions:

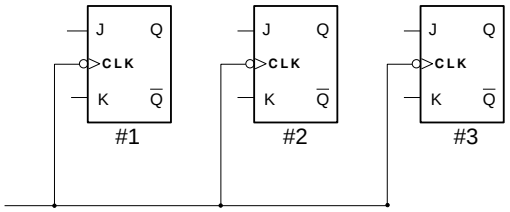
   (a) $A + \overline{(A + \overline{B}\,\overline{C}) \cdot (\overline{A}C + \overline{D})}$

   (b) $\overline{AB + \overline{\overline{C}}} + A\overline{C} + B$

   (c) $ABC + \overline{A}B\overline{C} + A\overline{B}C + \overline{A}C + A$

4. The Gray code state diagram of the three binary digits: $Q_1Q_2Q_3$ is displayed below:





Your job is to design a synchronous circuit built from three edge-triggered JKFFs that follows the above state diagram, where the three binary digits $Q_1Q_2Q_3$ are the outputs of distinct edge-triggered JKFFs. You will need to determine the gate arrangement needed to make this cycle run, i.e., connecting the outputs of the three JKFFs: Q_i to the inputs of the three JKFFs: J_iK_i possibly using the usual (AND, OR,...) gates.

(a) Begin by considering the possible transitions of a single JKFF. What values of JK allow a particular transition? Fill in the below table. Hint: in every row either J or K will be an X for "don't care".

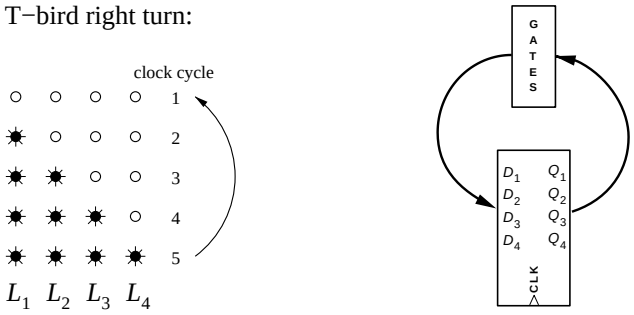
Transition:	J	K
$0 \longrightarrow 0$		
$0 \longrightarrow 1$		
$1 \longrightarrow 0$		
$1 \longrightarrow 1$		

(b) Now fill in the below table which displays the desired cycle.

Q_1	Q_2	Q_3	J_1	K_1	J_2	K_2	J_3	K_3
0	0	0						
0	0	1						
0	1	1						
0	1	0						
1	1	0						
1	1	1						
1	0	1						
1	0	0						

(c) Gates for J_1, K_1, J_2, and K_2 can be fairly easily generated using two input gates and the outputs of the JKFFs. Pick one of the above (J_1, K_1, J_2, or K_2) and find a simple expression for it in terms of the Q_i and $\overline{Q_i}$. Implement (construct) your expression using simple gates.

5. The rear turn indicators on old Ford Thunderbirds had a distinctive pattern in which the four segments of the indicator were illuminated sequentially, "pointing" the direction of the turn. For example, on a right-turning T-bird you would see the four segments which made up the right-rear indicator cycle as shown below:



Implement this cycle using four DFFs and external gates ($Q_i = L_i$).

(a) Draw the state diagram for this process.

(b) Display in the below table the desired cycle

L_1	L_2	L_3	L_4	D_1	D_2	D_3	D_4
0	0	0	0				

(c) Make a Karnaugh map for D_2 in terms of the four logical variables L_1, L_2, L_3, L_4. Note that there will be *lots* of Xs (don't care) entries. Circle appropriate groups to find a boolean expression for D_2. Please carefully label your Karnaugh maps so I know what each row and column of the map represents!

6. Consider the below circuit using pair of JKFF. Directly on this sheet appropriately (i.e., correctly given the circuit diagram) label each JKFF "negative edge triggered" or "positive edge triggered". At the start of this clock sequence Q_1 amd Q_2 are both HIGH; controls J_1 and K_1 are changed as shown in the below plot stack and controls J_2 and K_2 are determined by the circuit. Directly on the below stack of output traces record Q_1 (i.e., output of #1 JKFF) and Q_2 for the given "clock in" stream.