CSE3015 Digital Logic Design

Study Questions

Q1) Design an odd parity generator circuit for 3-bit data – If the number of 1's in the 3-bit data is odd, then F is 1, otherwise F is 0. (Ex: For input data, 010 F is 1, 110 F is 0). Start from a truth table, then convert to an equation, and finally convert equation to circuit.

Q2) Determine whether the Boolean functions F=ab' and G=(a' + ab)' are equivalent, using **a)** algebraic manipulation and **b)** truth tables.

Q3) Convert F = w AND x AND ((NOT(x) AND y) OR (z AND NOT(y)) OR (x AND y)) into SOP (Sum of Products) form and make simplifications

Q4) Convert F(a,b,c) = (a+c)(a'+ab)(b+c) into sum-of-minterms.

Q5) Use K-Map method to minimize function F = { A'.B'.C'.D', A'.B.C'.D', A'.B.C'.D, A'.B.C.D, A'.B.C.D', A.B.C'.D', A.B.C'.D, A.B.C.D, A.B.C.D', A.B'.C'.D'} with given don't cares A'.B'.C.D', A'.B'.C.D and A.B'.C.D'.

Q6) Use multiple levels to reduce number of transistors for F = A.C + A.B.C + A.B.D

Q7) Implement a 4-to-16 decoder using 2-to-4 decoders and/or 2-input logic gates at minimum cost. A 2-to-4 decoder costs 5 units, and/or gates cost 2 units, inverter cost 1 units.

Q8) Show that any boolean function can be implemented using just NOR gates. (Hint: Show that you can implement AND, OR, and NOT with NOR)