EE2001-Tutorial 2
Date: 30th January 2018
Gate-Level Minimization and Some Combinational Logic

1) Simplify the following Boolean functions, using Karnaugh maps:
(a) F(x, y, z) = Σ(2, 3, 6, 7)
(b) F(A, B, C, D) = Σ(4, 6, 7, 15)
(c) F(A, B, C, D) = Σ(3, 7, 11, 13, 14, 15)
(d) F(w, x, y, z) = Σ(2, 3, 12, 13, 14, 15)
(e) F(w, x, y, z) = Σ(11, 12, 13, 14, 15)
(f) F(w, x, y, z) = Σ(8, 10, 12, 13, 14)

2) Simplify the following Boolean expressions, using four variable K-maps:
(a) w'z + xz + x'y + wx'z
(b) AD' + B'C'D + BCD' + BC'D
(c) AB'C + B'C'D' + BCD + ACD' + A'B'C + A'BC'D
(d) wxy + xz + wx'z + w'x

3) Find the minterms of the following Boolean expressions:
(a) xy + yz + xy'z
(b) C'D + ABC' + ABD' + A'B'D
(c) wyz + w'x' + wxz'
(d) A'B + A'CD + B'CD + BC'D'

4) Find all the prime implicants for the following Boolean functions, and determine which are essential:
(a) F(w, x, y, z) = Σ (0, 2, 4, 5, 6, 7, 8, 10, 13, 15)
(b) F(A, B, C, D) = Σ (0, 2, 3, 5, 7, 8, 10, 11, 14, 15)
(c) F(A, B, C, D) = Σ (2, 3, 4, 5, 6, 7, 9, 11, 12, 13)
(d) F(w, x, y, z) = Σ (1, 3, 6, 7, 8, 9, 12, 13, 14, 15)
(e) F(A, B, C, D) = Σ (0, 1, 2, 5, 7, 8, 9, 10, 13, 15)
(f) F(w, x, y, z) = Σ (0, 1, 2, 5, 7, 8, 10, 15)

5) (i) Convert the following Boolean function from a sum-of-products form to a simplified product-of-sums form.
F(w, x, y, z) = Σ (0, 1, 2, 5, 8, 10, 13)
(ii) Simplify the following expressions to (1) sum-of-products and (2) products-of-sums:
(a) x'z' + y'z' + yz' + xy
(b) ACD' + C'D + AB' + ABCD
(c) (A + B + D')(A' + B' + C')(A' + B' + C)(B' + C + D')
(d) BCD' + ABC' + ACD

6) Simplify the following Boolean function F, together with the don't-care conditions d, and then express the simplified function in sum-of-products form:

(a) $F(x, y, z) = \Sigma (0, 1, 4, 5, 6)$, $d(x, y, z) = \Sigma (2, 3, 7)$

(b) $F(A,B,C,D) = \Sigma (0, 6, 8, 13, 14)$, $d(A,B, C, D) = \Sigma (2,4,10)$

(c) $F(A,B,C,D) = \Sigma (5, 6, 7, 12, 14, 15)$, $d(A,B,C,D) = \Sigma (3, 9,11)$

(d) $F(A,B,C,D) = \Sigma (4, 12, 7, 2, 10)$, $d(A,B,C,D) = \Sigma (0, 6, 8)$

7) (i) With the use of maps, find the simplest sum-of-products form of the function $F = fg$, where $f = abc' + c'd + a'cd' + b'cd'$ and $g = (a + b + c' + d')(b' + c' + d)(a' + c + d')$

(ii) Implement $F(A, B, C, D) = \Sigma (0, 4, 8, 9, 10, 11, 12, 14)$ using the two-level forms of logic

(a) NAND-AND, (b) AND-NOR, (c) OR-NAND and (d) NOR-OR

8(i) Design a combinational circuit with three inputs and one output.

(a) The output is 1 when the binary value of the inputs is less than 3. The output is 0 otherwise.

(b) The output is 1 when the binary value of the inputs is an even number.

(ii) A majority circuit is a combinational circuit whose output is equal to 1 if the input variables have more 1's than 0's. The output is 0 otherwise. Design a 3-input majority circuit by finding the circuit's truth table, Boolean equation, and a logic diagram.

9(i) Design a combinational circuit that converts a four-bit Gray code to a bit four binary number. Implement the circuit with exclusive-OR gates.

(ii) Design a four-bit combinational circuit 2's complementer. (The output generates the 2's complement of the input binary number.) Show that the circuit can be constructed with exclusive-OR gates. Can you predict what the output functions are for a five-bit 2's complementer?

10) (i) Design a half-subtractor circuit with inputs x and y and outputs D_{iff} and B_{out}. The circuit subtracts the bits $x – y$ and places the difference in D_{iff} and the borrow in B_{out}.

(ii) Design a full-subtractor circuit with three inputs x, y, B_{in} and two outputs D_{iff} and B_{out}. The circuit subtracts $x–y– B_{in}$, where B_{in} is the input borrow, B_{out} is the output borrow, and D_{iff} is the difference.