CS-472: Design Technologies for Integrated Systems

Exercise Problem Set 2 Date: 18/09/2025

Problem 1

A truth table is a complete listing of all points in a Boolean input space and the corresponding output values. For example, the two-input AND function has the following truth table:

x_1	x_0	x_0x_1
0	0	0
0	1	0
1	0	0
1	1	1

Given the following function: $f(a, b, c) = a\bar{b}c + bc + \bar{a}b\bar{c}$

(a) Write the truth table.

(b) Write the function in terms of its minterms.

(c) Write the function using only NAND-2 gates.

Problem 2

To compactly represent a truth table, we can use a bitstring $b_{2^n-1} \dots b_1 b_0$ where $b_x = f(w_1, \dots, w_n)$ when $w = (x_n \dots x_1)_2$. Each bit in the bitstring corresponds to the output of f evaluated at some assignment to its variables. The most-significant bit b_{2^n-1} corresponds to all variables assigned to 1, $f(1, \dots, 1)$ and the least-significant bit b_0 correspond to $f(0, \dots, 0)$. Using this compact representation, the two-input AND function is 1000.

(a) Suppose that on a remote country logicians use the symbol 1 for "false" and 0 for "true". How do our logical operations associate to theirs?

For example, they have a binary operation "or"

$$1 \text{ or } 1 = 1, \quad 1 \text{ or } 0 = 0, \quad 0 \text{ or } 1 = 0, \quad 0 \text{ or } 0 = 0$$

which we associate with AND (1000).

Table 1: The sixteen logical operations on two variables, $f(x, y)$.

Truth table	Notation(s)	Name
0000	$\bot$	Contradiction; antilogy; constant 0
0001	$x \mathbin{\overline{\vee}} y$, $x \downarrow y$	Nondisjunction; NOR
0010		Converse nonimplication
0011	x', $\neg x$, $\bar{x}$	Left complementation
0100		Nonimplication
0101	y', $\neg y$, $\bar{y}$	Right complementation
0110	$x \oplus y$	Exclusive disjunction; nonequivalence; XOR
0111	$x \mathbin{\overline{\wedge}} y$, $\uparrow y$	Nonconjuction; NAND
1000	$x \cdot y$, xy, $x \wedge y$	Conjunction; AND
1001	$x \equiv y$, $x \Leftrightarrow y$	Equivalence
1010	y	Right projection
1011	$x \Rightarrow y$	Implication
1100	x	Left projection
1101	$x \Leftarrow y$	Converse implication
1110	$x + y$, $x \vee y$	Disjunction; OR
1111	$\top$	Affirmation; tautology; constant 1

(b) All operations in Table 1 can be expressed in terms of NAND. For each of the 16 operations in the table find a formula equivalent to the function that uses only two-input NAND. The formula should be as short as possible and not contain any constant.

(c) Similarly, find 16 short formulas when constants are allowed.

(d) For each of the 16 operations in the table try to find a formula equivalent to the function using only the two-input XOR operator ($\oplus$) and without the use of constants 0 or 1. Is it possible? Why?

(e) Consider the previous exercise using the material implication ($\Rightarrow$) as basis operator instead of $\oplus$.

(f) If the use of constants is allowed, how would it affect the answer for (e)?