Fun with Boolean Logic

James E. Stine
Associate Professor
Electrical and Computer Engineering Department
Oklahoma State University
Stillwater, OK 74078, USA

1 Definitions

The following are some definitions that may be useful for Boolean Algebra which is the essential underwire for Digital Logic Design. Boolean algebra theorems and axioms can be used to simplify Boolean equations. As explained in the text, these Boolean algebra rules are very similar to those of ordinary algebra, but many are simpler. This occurs because the number set is significantly reduced to only contain 0 and 1.

- An Axiom is a mathematical statement that serves as a starting point from which other statements are logically derived.
- A Theorem is a type of abstract object which can be derived from the rules of the formal system that is applied to the formal language usually based off or derived from a fixed set of inference rules or axioms without any additional assumptions.
- A Dual of an expression is computed by replacing every + by a ·; every · by a +, every 1 by a 0 and every 0 by a 1 without changing the order of computation.

$$\textit{dual}\ f(x_1, x_2, \ldots, x_n, +, \cdot, 1, 0) = f(x_1, x_2, \ldots, x_n, \cdot, +, 0, 1)$$

- A Lemma is a proven proposition which is used as a stepping stone to a larger result rather than a statement in-and-of itself.

2 Boolean Algebra

This section is meant to give the various Boolean algebra axioms and theorems used in the book along with some more advanced and extremely powerful versions. The more advanced expressions, such as Shannon decomposition, are used in programs, such as espresso, to simplify Boolean logic.

Axiom		Dual		Name
$A1$	$B = 0$ if $B \neq 1$	$A1'$	$B = 1$ if $B \neq 0$	Binary field
$A2$	$\overline{0} = 1$	$A2'$	$\overline{1} = 0$	NOT
$A3$	$0 \cdot 0 = 0$	$A3'$	$1 + 1 = 1$	AND/OR
$A4$	$1 \cdot 1 = 1$	$A4'$	$0 + 0 = 0$	AND/OR
$A5$	$0 \cdot 1 = 1 \cdot 0 = 0$	$A5'$	$1 + 0 = 0 + 1 = 1$	AND/OR

Table 1: Axioms of Boolean algebra.

2.1 Generalized De Morgan's Theorem

$$\begin{aligned} \overline{x_1 + x_2 + \cdots + x_n} &= \overline{x_1} \cdot \overline{x_2} \cdots \overline{x_n} \\ \overline{x_1 \cdot x_2 \cdots x_n} &= \overline{x_1} + \overline{x_2} + \cdots + \overline{x_n} \end{aligned}$$

Theorem		Dual		Name
$T1$	$B \cdot 1 = B$	$T1'$	$B + 0 = B$	Identity
$T2$	$B \cdot 0 = 0$	$T2'$	$B + 1 = B$	Null Element
$T3$	$B \cdot B = B$	$T3'$	$B + B = B$	Idempotency
$T4$	$\overline{\overline{B}}$			Involution
$T5$	$B \cdot \overline{B} = 0$	$T5'$	$B + \overline{B} = 1$	Complements

Table 2: Boolean Theorems of One Variable.

Theorem		Dual		Name
$T6$	$B \cdot C = C \cdot B$	$T6'$	$B + C = C + B$	Commutativity
$T7$	$(B \cdot C) \cdot D = B \cdot (C \cdot D)$	$T7'$	$(B + C) + D = B + (C + D)$	Associativity
$T8$	$(B \cdot C) + (B \cdot D) = B \cdot (C + D)$	$T8'$	$(B + C) \cdot (B + D) = B + (C \cdot D)$	Distributivity
$T9$	$B \cdot (B + C) = B$	$T9'$	$B + (B \cdot C) = B$	Covering
$T10$	$(B \cdot C) + (B \cdot \overline{C}) = B$	$T10'$	$(B + C) \cdot (B + \overline{C}) = B$	Combining
$T11$	$(B \cdot C) + (\overline{B} \cdot D) + (C \cdot D)$ $= B \cdot C + \overline{B} \cdot D$	$T11'$	$(B + C) \cdot (\overline{B} + D) \cdot (C + D)$ $= (B + C) \cdot (\overline{B} + D)$	Consensus
$T12$	$\overline{B_0 \cdot B_1 \cdot B_2} = (\overline{B_0} + \overline{B_1} + \overline{B_2})$	$T12'$	$\overline{B_0 + B_1 + B_2} = (\overline{B_0} \cdot \overline{B_1} \cdot \overline{B_2})$	De Morgan's Theorem

Table 3: Boolean Theorems of Several Variable.

2.2 Shannon's Theorem

$$\overline{f(x_1, x_2, \ldots, x_n, +, \cdot)} = f(\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n}, \cdot, +)$$

2.3 Shannon's Decomposition Theorems

Careful - extremely powerful!

$$\begin{aligned} f(x_1, x_2, \ldots, x_n) &= x_1 \cdot f(1, x_2, \ldots, x_n) + \overline{x_1} \cdot f(0, x_2, \ldots, x_n) \\ f(x_1, x_2, \ldots, x_n) &= (x_1 + f(0, x_2, \ldots, x_n)) \cdot (\overline{x_1} + f(1, x_2, \ldots, x_n)) \end{aligned}$$

The decomposition theorems, which are usually attributed to Shannon (what else isn't?), can be easily generated by comparing truth tables of both sides of the equations when $x_1 = 1$ and when $x_0 = 0$. From these decomposition theorems, a powerful result is formulated that allows many of the synthesis packages, including the ones in your Xilinx software, to simplify many of your schematics and Hardware Descriptive Language (HDL) code (including espresso).

$$\begin{aligned} x_1 \cdot f(x_1, x_2, \ldots, x_n) &= x_1 \cdot f(1, x_2, \ldots, x_n) \\ \overline{x_1} \cdot f(x_1, x_2, \ldots, x_n) &= \overline{x_1} \cdot f(0, x_2, \ldots, x_n) \\ x_1 + f(x_1, x_2, \ldots, x_n) &= x_1 + f(0, x_2, \ldots, x_n) \\ \overline{x_1} + f(x_1, x_2, \ldots, x_n) &= \overline{x_1} + f(1, x_2, \ldots, x_n) \end{aligned}$$