We can express Boolean functions in Verilog using the `assign` statement and substituting `&` for `∧`, `|` for `∨`, and `~` for NOT. Once expressed in Verilog, we can simulate a Boolean function, or encapsulate it in a `module` that can be used to build more complex functions.

**BIBLIOGRAPHIC NOTES**

George Boole formulated Boolean logic in the mid 1800s. He presents his results in two texts that are now freely available online [12], [13]. De Morgan describes much of his logic formulations in an 1860 text [35].

**Exercises**

3.1 *Prove absorption.* Prove that the absorption property is true by using perfect induction (i.e., enumerate all the possibilities).

3.2 *Prove idempotence.* Prove that the idempotence property is true by using perfect induction.

3.3 *Prove the associative property.* Prove that the associative property is true.

3.4 *Prove the distributive property.* Prove that the distributive property – both `∧` over `∨` and `∨` over `∧` – is true.

3.5 *It’s not + and ×.* Prove the distributive property does not work for distributing `+` over `×` with integers.

3.6 *De Morgan’s theorem, I.* Using perfect induction, prove De Morgan’s theorem with four variables. Specifically

\[ \overline{w \land x \land y \land z} = w \lor x \lor y \lor z \]

and

\[ w \lor x \lor y \lor z = \overline{w \land x \land y \land z}. \]

3.7 *De Morgan’s theorem, II.* Show that Equation (3.9) is indeed true by successively applying De Morgan’s theorem to a logic function in normal form.

3.8 *Simplifying Boolean equations, I.* Reduce the following Boolean expression to a minimum number of literals: \((x \lor y) \land (x \lor \overline{y})\).

3.9 *Simplifying Boolean equations, II.* Reduce the following Boolean expression to a minimum number of literals: 
\((x \land y \land z) \lor (\overline{x} \land y) \lor (x \land y \land \overline{z})\).

3.10 *Simplifying Boolean equations, III.* Reduce the following Boolean expression to a minimum number of literals: \(((y \land \overline{z}) \lor (\overline{x} \land w)) \land ((x \land \overline{y}) \lor (z \land w))\).

3.11 *Simplifying Boolean equations, IV.* Reduce the following Boolean expression to a minimum number of literals: 
\((x \land y) \lor (x \land ((w \land z) \lor (w \land \overline{z})))\).

3.12 *Simplifying Boolean equations, V.* Reduce the following Boolean expression to a minimum number of literals: 
\((w \land \overline{x} \land \overline{y}) \lor (w \land \overline{x} \land \overline{y} \land z) \lor (w \land x \land \overline{y} \land z)\).
3.13 Dual functions, I. Find the dual function of the following function and write it in normal form: \( f(x, y) = (x \land \bar{y}) \lor (\bar{x} \land y) \). 

3.14 Dual functions, II. Find the dual function of the following function and write it in normal form: \( f(x, y, z) = (x \land y) \lor (x \land z) \lor (y \land z) \). 

3.15 Dual functions, III. Find the dual function of the following function and write it in normal form: \( f(x, y, z) = (x \land ((y \land z) \lor (\bar{y} \land \bar{z}))) \lor (\bar{x} \land ((y \land z) \lor (\bar{y} \land z))) \). 

3.16 Normal form, I. Rewrite the following Boolean expression in normal form: 
\( f(x, y, z) = (x \land \bar{y}) \lor (\bar{x} \land z) \).

3.17 Normal form, II. Rewrite the following Boolean expression in normal form:
\( f(x, y, z) = x \).

3.18 Normal form, III. Rewrite the following Boolean expression in normal form:
\( f(x, y, z) = (x \land ((y \land z) \lor (\bar{y} \land \bar{z}))) \lor (\bar{x} \land ((y \land z) \lor (\bar{y} \land z))) \).

3.19 Normal form, IV. Rewrite the following Boolean expression in normal form:
\( f(x, y, z) = 1 \) if exactly 0 or 2 inputs are 1.

3.20 Equation from schematic, I. Write down a simplified Boolean expression for the functions computed by the logic circuit of Figure 3.10(a).

3.21 Equation from schematic, II. Write down a simplified Boolean expression for the functions computed by the logic circuit of Figure 3.10(b).

3.22 Equation from schematic, III. Write down a simplified Boolean expression for the functions computed by the logic circuit of Figure 3.10(c).

3.23 Schematic from equation, I. Draw a schematic for the following unsimplified logic equation: 
\( f(x, y, z) = (\bar{x} \land y \land \bar{z}) \lor (\bar{x} \land \bar{y} \land z) \lor (x \land \bar{y} \land \bar{z}) \).

3.24 Schematic from equation, II. Draw a schematic for the following unsimplified logic equation: 
\( f(x, y, z) = ((x \land y) \lor z) \land (x \land \bar{z}) \).

Figure 3.10. Logic circuits for Exercises 3.20, 3.21, and 3.22.
3.25 *Schematic from equation, III.* Draw a schematic for the following unsimplified logic equation:

\[ f(x, y, z) = \overline{(x \land y)} \lor z. \]

3.26 *Schematic from equation, IV.* Draw a schematic for the following unsimplified logic equation: \( f(x, y, z) = 1 \) if 1 or 2 inputs are 1.

3.27 *Verilog* Write a Verilog module that implements the logic function

\[ f(x, y, z) = (x \land y) \lor (\bar{x} \land z). \]

Write a test script to verify the operation of your module on all eight combinations of \( x, y, \) and \( z \). What function does this circuit realize?

![Logic circuit for Exercise 3.28.](image)

**Figure 3.11.** Logic circuit for Exercise 3.28.

3.28 *Logic equations.*

(a) Write out the unsimplified logic equation for the circuit of Figure 3.11.

(b) Write the dual form with no simplification.

(c) Draw the circuit for the unsimplified dual form.

(d) Simplify the original equation.

(e) Explain how the inverter and the last OR gate in the original circuit work together to allow this simplification.

3.29 *Choosing a representation.* Which representation, (a), (b), or (c), for a playing card requires the fewest gate inputs to check if three playing cards are all of the same suit (i.e., all hearts, all spades, all diamonds, or all clubs). For the purposes of this answer, assume that an XOR gate costs the same as three normal gates, e.g., a two-input XOR costs six gate inputs.

(a) Representing the suit as a four-bit one-hot number. A one-hot number has exactly one bit set to 1. For example, clubs would be represented by 0001, spades by 0010, diamonds by 0100, and hearts by 1000.

(b) Representing the suit as a two-bit Gray-coded number. Gray coding is explained in Exercise 1.9, an example of which is 00 for clubs, 01 for spades, etc.

(c) Representing the suit with three bits that can be either one-hot or zero-hot. This encoding can have zero (000) or one bit set (001, 010, 100).