Sets	Propositions
Commutative Laws	
$P \cap Q = Q \cap P$	$p \wedge q = q \wedge p$
$P \cup Q = Q \cup P$	$p \vee q = q \vee p$
Associative Laws	
$(P \cap Q) \cap R = P \cap (Q \cap R)$	$(p \wedge q) \wedge r = p \wedge (q \wedge r)$
$(P \cup Q) \cup R = P \cup (Q \cup R)$	$(p \vee q) \vee r = p \vee (q \vee r)$
Distributive Laws	
$P \cap (Q \cup R) = (P \cap Q) \cup (P \cap R)$	$p \wedge (q \vee r) = (p \wedge q) \vee (p \wedge r)$
$P \cup (Q \cap R) = (P \cup Q) \cap (P \cup R)$	$p \vee (q \wedge r) = (p \vee q) \wedge (p \vee r)$
De Morgan's Laws	
$(P \cap Q)' = P' \cup Q'$	$\neg(p \wedge q) = \neg p \vee \neg q$
$(P \cup Q)' = P' \cap Q'$	$\neg(p \vee q) = \neg p \wedge \neg q$
Identity Laws	
$P \cap \emptyset = \emptyset;\ P \cup \emptyset = P$	$p \wedge F = F;\ p \vee F = p$
$P \cap \mathcal{U} = P;\ P \cup \mathcal{U} = \mathcal{U}$	$p \wedge T = p;\ p \vee T = T$
Absorption and Complement Laws	
$P \cap P = P;\ P \cup P = P$	$p \wedge p = p;\ p \vee p = p$
$P \cap P' = \emptyset;\ P \cup P' = \mathcal{U}$	$p \wedge \neg p = F;\ p \vee \neg p = T$

Example 3.11 We use the laws of logic to prove

$$\neg(p \wedge \neg q) = \neg p \vee q.$$

First, by De Morgan's Law, we have

$$\neg(p \wedge \neg q) = \neg p \vee \neg(\neg q).$$

However, $\neg(\neg q) = q$, by Result 3.3. Thus we have

$$\neg(p \wedge \neg q) = \neg p \vee q,$$

as required. □

3.4 Logic Gates

Learning Objectives

After studying this section, you should be able to:

- draw block diagrams to represent the **NOT-gate**, the **AND-gate** and the **OR-gate**;
- construct a logic network to represent a given symbolic statement;
- obtain an expression for the output from a given logic network.

Introduction

A modern digital computer is often envisaged as a super-fast adding machine. It would be more accurate, however, to think of it as a logic machine, because it performs all the arithmetical operations by means of the logical rules summarized in the previous section. It does this by means of simple electronic circuits called **gates**. These are designed to operate on one or more input