Number Systems

See section 12.1 in Schwarz and Oldham.

The ordinary numbers that we use every day are written in base (or radix) 10, or decimal, because each digit represents a multiple of a power of 10. The decimal number 3211, for example, can be written:

$$3 \times 10^3 \quad + \quad 2 \times 10^2 \quad + \quad 1 \times 10^1 \quad + \quad 1 \times 10^0$$

But we can also write numbers where each digit represents a multiple of a power of 2. Such numbers are said to be written in base 2, or binary. For example, 101011 can represent:

$$1 \times 2^5 \quad + \quad 0 \times 2^4 \quad + \quad 1 \times 2^3 \quad + \quad 0 \times 2^2 \quad + \quad 1 \times 2^1 \quad + \quad 1 \times 2^0$$

which in base 10 sums up to 43 (i.e., $4 \times 10^1 + 3 \times 10^0$). In fact, we can express a number in base 8 (octal), base 16 (hexadecimal), or base anything (as long as it's an integer greater 2). If the number contains digits to the right of a decimal point, remember that those digits represent multiples of negative powers.

We may use a subscript to indicate the base of a number:

$$101011_2 \quad = \quad 43_{10} \quad = \quad 53_8 \quad = \quad 2B_{16}$$

Note that in base 2 we need 2 digits, namely 0 and 1. In base 10 we need 10 digits; in base 16, 16 digits. In base 16, the first ten digits are the same as in base 10, but then the letters A, B, C, D, E, and F are used for the other six.

Examples:

Convert 137_{10} to octal:

64 (= 8^2)	8 (= 8^1)	1 (=8^0)

Convert 111_8 to decimal:

64 (= 8^2)	8 (= 8^1)	1 (=8^0)

Complements

Computers use complements for logical manipulation and subtraction. There are two types of complements, diminished radix complement and radix complement. Although complements are applicable to any base, we'll only need to consider binary (base 2). Since the radix (base) is 2, the diminished radix complement is called the 1's complement and the radix complement is called the 2's complement.

Diminished Radix Complement

In general, given a number N in base r with n digits, the (r – 1)'s complement of N is $(r^n - 1) - N$. In binary, this is called the 1's complement and is very simple to calculate. All we have to do is take the number and change every 1 into a 0 and every 0 into a 1. (Why?)

Example:

What is the 1's complement of 1011000?

Radix Complement

In general, given a number N in base r with n digits, the r's complement of N is $r^n - N$. Of course this is simply 1 greater than the diminished radix complement. In binary, the radix complement is called the 2's complement and is most easily obtained by adding 1 to the 1's complement.

Example:

What is the 2's complement of 1101100?

Binary Subtraction using 2's Complements

By using 2's complements, we can subtract by adding. To perform X – Y:

1. Find the 2's complement of Y.
2. Add the 2's complement of Y to X.
3. If there is an end carry, discard it. Call the result Z.
4. If X > Y, then the answer is Z. If X < Y, then the answer is the 2's complement of Z with a minus sign in front.

Examples:

What is 1101100 – 1011000?

What is 0101110 – 1110100?

Boolean Algebra

Basics

- In Boolean algebra, there are only two values, 0 and 1 (false and true, etc.).
- We have only three basic operations: AND (·), OR (+), and NOT (¯ or ´).
- Operator precedence: parentheses first, then NOT, then AND, lastly OR.
- For convenience, we define XOR ($\oplus$), but note that $x \oplus y = x\bar{y} + \bar{x}y$.

What is 1 + 1?

This depends on what you mean by the + operator. We use it to mean two different things: binary addition and logical OR. When used for ordinary addition, $1 + 1 = 10_2 = 2_{10}$. When used for Boolean algebra, 1 + 1 = 1 OR 1 = 1.

Rules

❖ Zero and one:	$0 + x = x$	$0 \cdot x = 0$
	$1 + x = 1$	$1 \cdot x = x$
❖ Idempotent:	$x + x = x$	$x \cdot x = x$
❖ Complementarity:	$x + \bar{x} = 1$	$x \cdot \bar{x} = 0$
❖ Involution:	$\overline{(\bar{x})} = x$	
❖ Commutative:	$x + y = y + x$	$x \cdot y = y \cdot x$
❖ Associative:	$x + (y + z) = (x + y) + z$	$x \cdot (y \cdot z) = (x \cdot y) \cdot z$
❖ Distributive:	$x \cdot (y + z) = x \cdot y + x \cdot z$	$(x + y) \cdot (x + z) = x + y \cdot z$
❖ Absorption:	$x + x \cdot y = x$, $x + \bar{x} \cdot y = x + y$, and others	

Perfect Induction

Proof by perfect induction refers to proving a statement by listing all possible cases and showing that the statement holds for every case. Do this by constructing a truth table. Proof by perfect induction is possible in Boolean algebra because of the finiteness of input and output states. Proof by perfect induction is as valid as proof by algebraic reduction to basic axioms.

De Morgan's Theorem

Use this to distribute complementation. Remember to change the operation from AND to OR or vice versa.

$$\overline{x_1 + x_2 + x_3 + \dots + x_n} = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \dots \cdot \bar{x}_n$$

$$\overline{x_1 \cdot x_2 \cdot x_3 \cdot \dots \cdot x_n} = \bar{x}_1 + \bar{x}_2 + \bar{x}_3 + \dots + \bar{x}_n$$

Sum of Products and Product of Sums

Use the 1's in a truth table to get the sum of products. Use the 0's in a truth table and then find the complement to get the product of sums. Generally, the sum of products form is more common than the product of sums.